Dissertation

STATISTICAL UNSUPERVISED IMAGE CLUSTERING FOR POSITRON EMISSION TOMOGRAPHY IN RADIONUCLIDE THERAPY

Thomas Layer
(thomas.layer@yandex.com)

Institute of Telecommunications

Vienna University of Technology

DISSERTATION

STATISTICAL UNSUPERVISED IMAGE CLUSTERING FOR POSITRON EMISSION TOMOGRAPHY IN RADIONUCLIDE THERAPY

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines
Doktors der technischen Wissenschaften

unter der Leitung von
Ao. Univ.-Prof. Dipl.-Ing. Dr. Gerald Matz
Institute of Telecommunications

eingereicht an der Technischen Universität Wien
Fakultät für Elektrotechnik

von
Thomas Layer
Schillerstraße 13/B/1
2351 Wiener Neudorf

Wien, im November 2014 _____

Die Begutachtung dieser Arbeit erfolgte durch:

1. Ao. Univ.-Prof. Dipl.-Ing. Dr. G. Matz

   Institute of Telecommunications
   Technische Universität Wien

2. Assoc. Prof. Dipl.-Ing. Dr. G. Langs

   Department of Biomedical Imaging and Image-guided Therapy
   Medizinische Universität Wien

Abstract

This thesis introduces statistical classification algorithms for positron emission tomography (PET) images to support computational treatment planning in radiotherapy. Common clinical practice is based on manual delineation and on threshold methods. These methods suffer from certain shortcomings in connection with the following problems with PET images: (1) the images are very noisy; (2) the low resolution of the images leads to partial volume effects (PVE) with discrete clusterings.

To improve the clinical state of the art, we consider probabilistic models that are capable of describing partial membership of the image entities (voxels). As we are not given data sets to train learning models, our approaches have to estimate the labeling as well as the parameters describing the models. We first study classical methods like the expectation-maximization procedure to fit Gaussian models to the data (EMGMM). Due to bad statistical ensembles of small clusters, we study whether a Bayesian treatment of the parameters is beneficial for our task. To countersteer the image distortions which arise due to point spread effects, graphical models are considered next. With graphical models we can easily capture dependencies among voxels. The price to pay is a more complex optimization procedure for the parameters as well as for the labellings. The parameter estimation becomes a convex optimization problem which is solved by adapting the probability distributions to the corresponding empirical statistics of some intermediate labelled image. To solve the labelling problem we are either sampling label configurations according to local probability distributions or apply marginalization procedures like belief propagation.

The proposed algorithms are numerically assessed using PET images of a modified NEMA sphere phantom. These were acquired at the General Hospital of Vienna using general clinical settings at a Siemens Biograph True Point 64 PET/CT scanner. Multiple measurements have been performed using different signal-to-background ratios (SBR). To test the algorithms in tough conditions, a small sphere of 8mm diameter not used in previous investigations, has been added to the NEMA phantom. Moreover a measurement with SBR of 2.06 has not been used in previous works.

The small statistical ensembles of the NEMA spheres result in unreliable parameter estimates of the Gaussian distributions modeling the sphere voxels. Therefore the volume of the spheres get overestimated including also many outliers located in background regions. A Bayesian treatment of the parameters could not resolve this problem. Hence the mean and standard deviation of the sphere clusters are assigned ad hoc using information gained from the image, e.g. assigning the maximum intensity value to the average of the sphere voxels. This way, the EMGMM outperforms the clinical

state of the art methods regarding their volume predictability as well as regarding their capability of detecting spheres comprised in the noisy background reservoir of the NEMA phantom. Nevertheless Bayesian models yield powerful detection algorithms improving the detection statistic of the EMGMM.

Using Markov random fields (MRF), the overestimation of small clusters can be reduced more selectively with the drawback of influencing also the volume estimates of larger ensembles. Nevertheless the results are competitive with those of the EMGMM algorithm. Moreover with graphical models, reasonabe results are obtained by employing the parameter updates derived from the model. Finally, by applying MRFs only during defined correction steps, more stable results regarding the size of the analyzed image region are obtained.

Kurzfassung

Diese Arbeit wurde durchgeführt, um die Standardmethoden der Tumorsegmentierung in Bildern aus Positron Emissions Tomographen (PET) für die computerunterstützte Behandlungsplanung in der Radionuklidtherapie zu verbessern. Bei den zur Anwendung kommenden Verfahren handelt es sich um manuelle oder schwellwertbasierte Bestimmung von Lesionen. Diese haben folgende Nachteile: (1) hohes Verhältnis von Signal zu Hintegrund (SBR) in den Bildern; (2) geringe Auflösung der Bilder wodurch es an Objektgrenzen zu Problemen der Objektzuordung durch diskrete Segmentiervefahren (PVE) kommt.

Um die klinischen Standardverfahren zu verbessern, kommen Wahrscheinlichkeitsmodelle zur Anwendung. Diese besitzen die inherente Eigenschaft, Zugehörigkeiten von Bildelementen (Voxeln) in kontinuierlicher Weise zu bewerten und damit das Potential mit dem oben genannten PVE-Effekt umzugehen. Da mit der zur Verfügung stehenden Menge an Datensätze keine Lernmodelle trainiert werden können, müssen neben den Segmentierungen auch die Modellparameter bestimmt werden. Als erstes Modell kommen Gausssche Mischungsverteilungen zur Anwendung die mit Hilfe des Erwartungswert-Maximierung Algorithmus (EMGMM) optimiert werden. Da die Bestimmung der Modellparameter für kleine Objekte keine verlässlichen Werte liefert, wird weiters untersucht ob in diesem Fall Bayes-Schätzer vorteilhaft sind. Um der Unschärfe in den Bilder (verursacht durch die Übertragungsfunktion) entgengen zu wirken, werden graphische Modelle angewandt. Mit diesen können auf einfache Weise Abhängigkeiten unter den Voxeln modelliert werden. Dadurch werden jedoch die Optimierungsverfahren der Parameter als auch der Segmentierung erheblich verkomplizierter. Die Bestimmung der Paramter, ein konvexes Problem, wird durch Anpassung der Wahrscheinlichkeitsverteilungen an die entsprechenden empirisch gewonnen Statistiken aus zwischenzeitlich segmentierten Bildern bewerkstelligt. Ein sehr einfach zu implementierendes Verfahren zur Segmentierung der Bilder ist der Metropolisalgorithmus. Als alternatives Verfahren können Wahrscheinlichkeiten einzelner Bildelemente marginalisiert werden.

Die Verifizierung der vorgeschlagenen Algorithmen wird anhand von Messungen mit Hilfe eines NEMA-Phantoms durchgeführt. Dazu wurden mit dem Siemens Biograph True Point PET/CT des Allgemeinen Krankenhauses Wien Aufnahmen zu verschiedenen SBRs durchgeführt. Um die Leistungsfähigkeit der Methoden auch in Grenzfällen abschätzen zu können, wurde neben einem geringen SBR von 2.06 auch eine Kugel mit 8mm Durchmesser gemessen. Diese Verhältnisse wurden in keiner anderen Arbeit bis jetzt untersucht.

Wie erwähnt sind die Schätzungen der Modellparameter für die Kugeln (welche aus nur wenigen Voxeln bestehen) nicht zuverlässig, wodurch deren Volumina überbestimmt werden. Die Einführung von Bayes-Schätzer ist dabei nicht hilfreich. Darum werden sowohl die Mittelwerte als auch die Standardabweichungen der entsprechenden Wahrscheinlichkeitsverteilungen ad hoc zugewiesen. Z.B. wird an Stelle des Mittelwertes der Kugelsegmente der Maximalwert in diesen Bildregionen verwendet. Durch diese Vorgehensweise können mit Hilfe des EMGMM-Algorithmus Verbesserungen gegenüber der klinischen Standardmethoden erzielt werden. Diese beziehen sich nicht nur auf eine bessere Schätzung der Volumina, sondern auch auf eine höhere Detektionsrate. Dennoch verdient die Anwendung von Bayes-Schätzern Erwähnung als leistungsstärkste Detektionsvariante.

Die Verwendung von Markovschen Zufallsfeldern (MRF) führt zur besseren Schätzungen der kleinen Volumina mit dem Nachteil, dass grössere Volumina dadurch unterschätzt werden. Sie bieten dennoch Vorteile gegenüber klinischen Verfahren und stellen den Ausgangspunkt für zukünftige Untersuchungen, da die Optimierungsverfahren auch ohne ad hoc Parameterzuweisung valide Resultate liefern. Durch Anwendung von MRFs als Korrekturalgorithmen werden die Resultate stabiler bezüglich ihrer Abhängigkeit von der Grösse der zu untersuchened Bildregionen.

Acknowledgements

This thesis is dedicated to my parents, Ernestine and Alfred, because without their help my studies would not have been possible. Moreover i want to express my thanks to Matthias Blaickner and Gerald Matz for technical input and correction of this thesis.

Thomas Layer
Wien
Nov 2015

Contents

1

Introduction

COMPUTATIONAL treatment planning for radiotherapy relies on multi-modal imaging where the anatomical information of the organ containing the tumor origins from CT-scans whereas information about the metabolism gets delivered by emission tomography procedures like Positron-Emission-Tomography (PET) or Single-Photon-Emission-Computer-Tomography (SPECT). A subsequent dose calculation in the tumor and the surrounding tissue is done via numerical or analytical simulation of the radiation transport.

Oncological PET tracers like the analogue of glucose labeled with ^{18}F, called Fludeoxyglucose ^{18}F-FDG characteristically show an increased tracer uptake in lesions and therefore serve as an indicator whether a voxel[1] of the reconstructed tomography scans (PET or/and CT) belongs to the tumor or to healthy tissue. This classification therewith determines the volume and the mass of the tumor.

The determination of the tumor volume is one of the main causes for uncertainties in dosimetry [19]. When trying to assess the volume of a tumour for the sake of treatment planning in external beam radiation therapy (EBRT) or radionuclide therapy a very common practice is to have an expert manually draw a volume of interest (VOI) on the PET- or SPECT-image. The resulting and inevitable inter observer variations have been reported well enough for different types of cancer [6,17,37]. Another prevalent approach is the application of a threshold. The simplest choice for the threshold is a fixed percentage of the maximum activity concentration value [11, 12]. This thresholding method has been shown to predict well for big volumes but yields large errors in case of small volumes which is attributed to partial volume effects (PVE) and moreover depends on the signal to background ratio (SBR). Despite its questionable scientific meaningfulness it is still a widespread method even recommended by an international experts report [33]. Extensions of this method are automatic [5, 9] and iterative threshold approaches [23, 45, 46]. Apart from being sensitive to noise and SBR, standard thresholding methods need to be adjusted to every specific imaging system by performing phantom measurements

[1] A voxel is considered as a three-dimensional pendant to a pixel in two-dimensional images.

in order to guarantee useful regression curves [30]. Alternative methods such as watershed and edge detection are also sensitive to noise and different SBR [40, 41].

As mentioned above, PVE is a significant error source leading to biases in the order of those caused by attenuation effects [42]. In PET attenuation and scatter corrections are implemented into the respective firmware whereas this is not state of the art with PVE. Basically two effects give rise to PVE. First a three-dimensional image blurring occurs which is related to the finite spatial resolution of the PET scanner arising from the limitations in detector design, the reconstruction process and the free path length of the positrons. This blurring causes a spill-over between regions, spreading activity from small isolated peaks into their neighborhood (see figure 1.1 (rigth)). Secondly the continuous activity distribution is sampled on a voxel grid (see figure 1.1 (left)). Naturally the contours of the voxels do not match the actual contours of the tracer distribution, causing a voxel to contain more than one object and therefore carry potential errors when solving discrete labeling problems (see figure 1.1 (middle)). With this in mind it is obvious that smaller lesions will appear bigger as they are but show a lower intensity and thus enhance the possibility of misinterpreting the presence of cancerous tissue in patients. Also tumors necrotic in their centers suffer from intensity loss and therefore further aggravate accurate diagnosis. The main parameters affecting PVE are the tumor size, the image resolution and the SBR which is responsible for the spill-in and -out relation.

To overcome those drawbacks a few PVE correction methods have been applied to PET imaging. One approach is to incorporate measured point spread functions (PSF) of the scanner into the reconstruction process as it is done by the Siemens algorithm called TrueX. These PSFs are measured with box sources that have the same dimensions as the image resolution and are included into an Ordered Subset Expectation Maximization reconstruction. The accuracy of the resulting images has

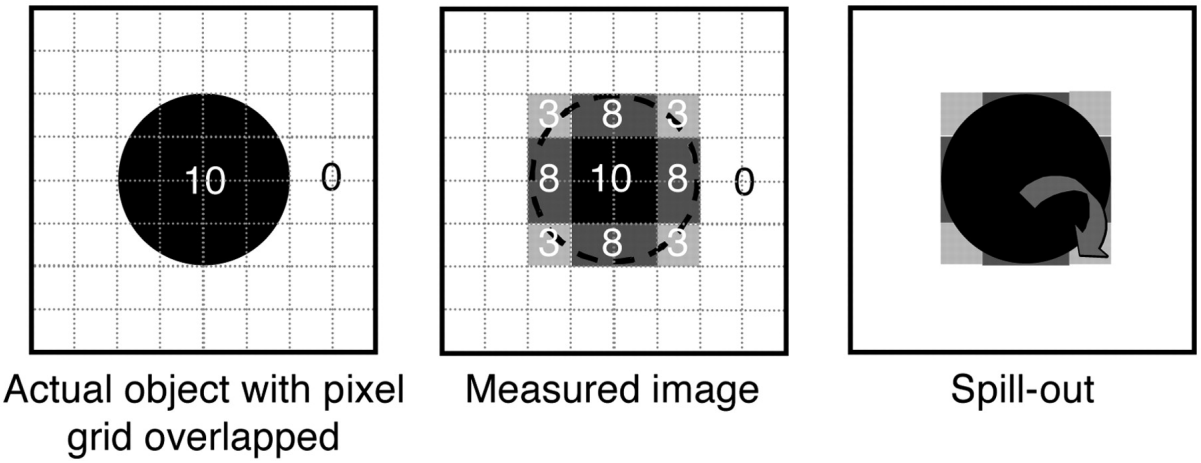


Figure 1.1: Influence of image sampling on PVE. Pixels on edges of source include both source and background tissues. Signal intensity in these pixels is mean of signal intensities of underlying tissues. Part of signal emanating from source is seen outside actual object and therefore is described as spilling out. The figure was taken from [42]

been called into question by Knäusl and co-workers because of overestimations of activity levels for small NEMA [2] spheres [26]. The authors state that the measurement to determine the PSF of the detector is accomplished without background activity and therefore yields overestimations for objects embedded in a background reservoir.

In contrast to predefined methods influencing the reconstruction process post processing steps concerning partial memberships of voxels can be established. As an improvement to the afore mentioned state of the art methods in clinical practice, statistical methods combined with fuzzy logic[3] have been proposed by Hatt and co-workers [20, 21]. In statistical estimation noise is used to determine an appropriate probability distribution describing measurement circumstances of each voxel of the PET-image. In their first work they use a Gaussian noise description whereas in the second they chose a system comprising 8 different probability distributions called the Pearson system. The inclusion of fuzzy membership levels to a statistical model to describe PVE voxels yields better segmentation results regarding voxels close to object boundaries. A statistical approach without fuzzy members was published by Gribben et al [18]. Local spatial correlations are taken into concern using a Markov Random Field to describe an unsupervised segmentation map. In contrast to [21] and [20] a preceding deconvolution step with a Gaussian filter is needed. The results show further improvement compared to fuzzy approaches. All algorithms stated above [18, 20, 21] are evaluated using a NEMA phantom with spheres ranging from 10mm to 37mm. The applied SBR in [21] and [20] are 1:8 and 1:4, whereas the algorithm in [18] is tested with an S/B of 1:9.

In general, probabilistic pattern recognition has some advantages. The solution is expressed in terms of the probability that a voxel is member of a distinct class of objects which enables a natural treatment of PVE. Moreover local neighborhood information can be incorporated making it an optimal tool for pattern recognition in image analysis.

This work is part of a greater software package to support the daily routine in clinical diagnosis for PET and SPECT. For the purpose of automatic registration and segmentation two main work packages are projected, whereof the work presented in this writing is one of them. The first work package uses anatomical atlases of humans to register organs and determine activity distributions inside organs. On PET-scans, which carry the metabolic information, the boarders between organs are hardly to not revealable. Therefore human atlases are mapped to the CT by imposing logical constraints, to localize all organs. Having adapted an atlas to a specific CT-scan, the atlas can further be used to localize the organs on the corresponding PET-scan. Therewith a coarse dose estimation for each organ can be given. Moreover, the registration of organs offers the ability to feed accurate clustering algorithms to search for cancerous tissue. The advantage of such proceeding gets obvious from considering a human whole body PET in detail. Different organs accumulate different doses and therewith a search for tumors in whole body PET-scans is hard to accomplish. In contradiction, working just with information gained from single organs change the circumstances crucial. For a

[2] The NEMA IEC Body Phantom Set (Model PET/IEC-BODY/P) is a phantom modelling the circumstances inside humans, see chapter 5.

[3] The theory of fuzzy sets was developed by [50].

huge amount of healthy organs not attacked by cancerous tissue, the activity distribution can be approximated as a constant signal overlayed with noise. In case of presence of a small tumor (not necrotic in its center) inside an organ, which as mentioned shows increased tracer uptake, the image data can be approximately illustrated as two constant signals overlayed with noise.

The aim of this work is to analyze probabilistic pattern recognition methods for unsupervised PET image clustering in human organs, i.e. without any prior knowledge to train model parameters as in supervised segmentation. Moreover the models under consideration are constructed by solely using parametrized probability distributions. As stated in [25] efficient estimations for probability distribution parameters are attained for $N \to \infty$ with N representing the amount of data. So if small objects comprising just a few voxels are under consideration statistical estimation methods inherit the problem of working on poor statistical ensembles.

It is forecasted that the combination of poor statistical ensembles and PVE will also give rise to wrong estimation of the volume of small objects. Consider the problem of estimating volumes of small spherical foreground objects (FG) filled with constant activity concentration A_{FG} in large background (BG) filled with constant activity concentration A_{BG} and $A_{\mathrm{FG}} > A_{\mathrm{BG}}$. A histogram of a 11.49 ml NEMA sphere in the surrounding 114.9 ml BG is given in figure 1.2. The Gaussian curve on the left is an example of a good statistical ensemble and represents the background. In contrast the few Gaussian distributed FG voxels expected at a higher activity concentration values disappear in an ensemble of uniquely distributed partial volume voxels. As a consequence of the BG's clear delineation it is expected that the partial volume voxels get included into the FG object, thus yielding huge overestimation of the FG object's volume.

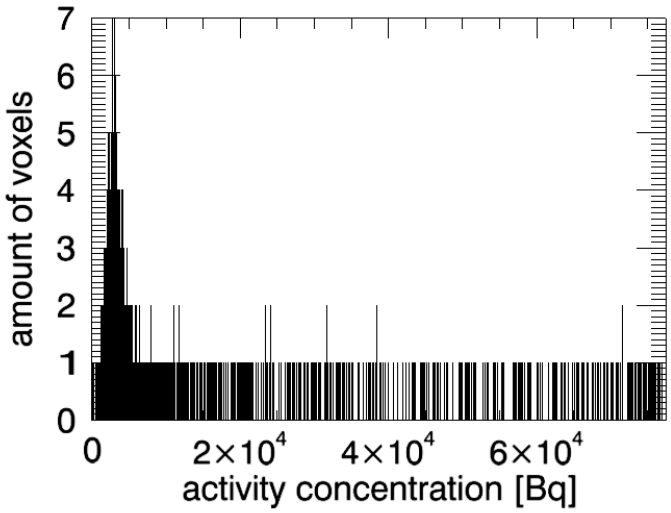


Figure 1.2: Histogram of NEMA sphere filled with constant activity in constant BG. Volume-ratio(FG/BG) = 1/10, SBR = 9.39.

2

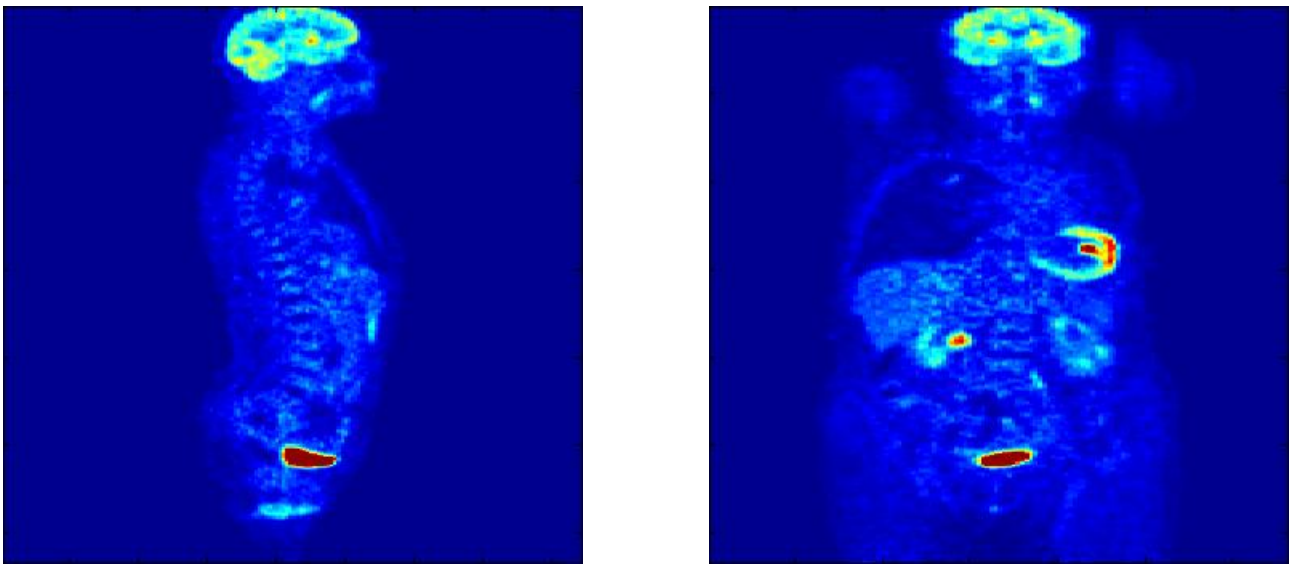
Nuclear Medicine - Positron Emission Tomography

THE application of nuclear- and particle physics in medicine has grown in the last decades involving diagnosis as well as treatment of diseases. Typical examples are the usage of radioactive labeled substances as in PET or SPECT, the interactions of atomic nuclei with external fields as in magnetic resonance imaging (MRI) or making use of nuclear reactions (e.g. Hadron Therapy). In particular the first two examples mentioned above provide possibilities to image either anatomical information (MRI) or functional information (PET, SPECT).

Tracers are molecules, i.e. carrier substances that participate in specific biological functions inside organisms and get labeled by a radionuclide [1] which does not change the biokinetics of the carrier. So the tracer gets distributed via the metabolism and can be monitored, enabling the visualization of pharmacokinetics in human bodies. An important field of application of such radiotracers is radionuclide therapy, a cancer treatment modality where the tracer enriches preferably in cancerous tissue and thus enables the application of doses of ionizing radiation from a very close distance. A possibility of generating such isotopes is using radionuclides with a long half live from nuclear reactors which further decay to daughter nuclides having a short half live. Accelerators are more easily established at hospitals and produce radiopharmaceuticals via nuclear capture or exchange reactions. The administration to patients can be performed via injection, ingestion or inhalation. For the purpose of diagnosis isotopes with a gamma line of suitable energy for the detector as well as a low to intermediate energy of the beta emission are preferred since in this case the dose applied in tissue is comparable low in contrast to high energy beta or alpha particles. Likewise cell damaging properties are desired for cancer treatment and therefore the latter ones get employed there.

The functional imaging principle is to capture the photons emerging due to radioactive decay. Using data from various projections, tomographic reconstruction techniques can be used to derive

[1] A radionuclide or radioisotope is a radioactive, i.e. instable atomic nucleus.



(a) (b)

Figure 2.1: ^{18}F-FDG PET image of (a) human coronal plane and (b) human sagittal plane (b)

the spatial distribution of the tracer. Most of the modern systems are hybrid machines comprising a PET or SPECT combined with a Computer Tomography (CT) to produce functional information co-registered with anatomical information. This way not only diagnosis is enhanced but also tomographic reconstruction processes benefits from the attenuation correction performed by the CT.

Figure 2.1(a) and (b) show a human coronal plane respectively a human sagittal plane of a PET scan after applying ^{18}F-FDG (see section 2.1.1). Since the tracer (see section 2.1.1) is a glucose analogue a brain uptake is unavoidable. Moreover the urinary bladder with it's higher uptake is visible at the bottom of the picture.

2.1 Physical Basics

2.1.1 Positron Emission

Positron emission was first discovered in 1934 by Frédéric and Irène Joliot-Curie due to shooting alpha particles at aluminum yielding a neutron and $^{30}_{15}$P, with the phosphorus decaying further via positron emission. The fundamental interaction guiding this kind of decay is termed the weak interaction and leads to the transformation of a proton to a neutron by changing an up quark to a down quark. To conserve the electric charge a weak interaction particle (W^+-Boson) gets emitted which further decays into a positron, the positively charged antiparticle of an electron, and a neutrino. The whole reaction can be written as

$$\beta^+\text{-decay:} \qquad p \longrightarrow n + e^+ + \nu. \tag{2.1}$$

Since the rest mass of a proton is smaller than the sum of neutron and positron (the mass of the neutrino is negligible) no decay of a free proton can occur. Subsequently a nucleus has to provide the necessary energy to generate the new particles. In the case of the β^+-decay: this is a nucleus with a surplus of protons.

In competition to this decay there is electron capture or K-capture. Thereby an electron from the inner most shell (K-shell) decays to a neutrino and a W^--Boson with subsequent fusion of the W^- and a proton of the nucleus to a neutron resulting in

$$e^-\text{-capture:} \qquad p + e^- \longrightarrow n + \nu. \tag{2.2}$$

The thereby generated energy is transmitted to the neutrino. As a secondary process of the electron capture the emission of an Auger electron or characteristic X-rays will occur when the K-shell gets filled up by an electron transition from an outermost shell.

As mentioned in the introduction the radiotracer is a chemical compound made of a carrier substance and an attached radionuclide. The commonly used radio-nuclei have short have lives (τ) as e.g. Carbon ^{11}C ($\tau = 20$ min.), Nitrogen ^{13}N ($\tau = 9.97$ min.), Oxygen ^{15}O ($\tau = 122$ sec.) and Fluorine ^{18}F ($\tau = 109$ min.). The carrier substance is chosen according to its pharmacokinetics, e.g. its binding to specific receptors. The most widely used radiotracer for PET is an analogue of glucose labeled with ^{18}F called Fludeoxyglucose ^{18}F-FDG.

2.1.2 Positron Electron Annihilation

In case of a collision between a positron and an electron both particles vanish which is called annihilation. Due to the conservation of the energy, electric charge and momentum such elimination process will create other particles. In cases of low energy just photons are created. In order to conserve electric charge and linear momentum at least two of those have to emerge. The generation of more photons is also possible but with decreased probability. Transforming into the frame of zero central momentum, the most common case is the emission of two photons with an angle of 180 degree and an energy of 511 keV each according to the rest energy of the electron/positron.

$$e^+e^-\text{-annihilation:} \qquad e^+ + e^- \longrightarrow \gamma + \gamma. \tag{2.3}$$

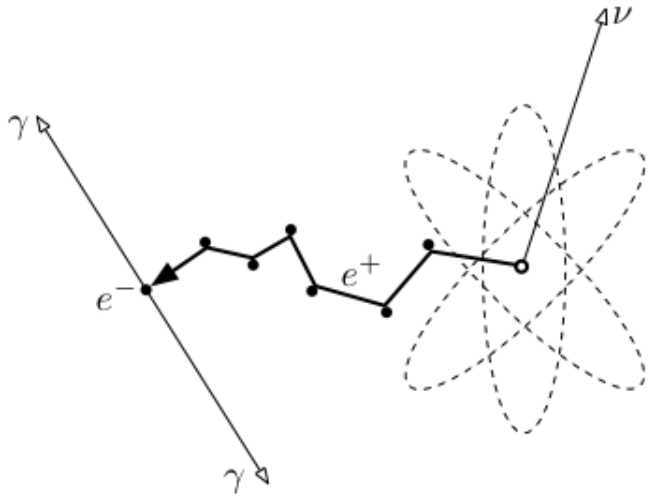


Figure 2.2: β^+ decay with outgoing neutrino (ν) and subsequent annihilation of the positron (e^+) with an electron (e^-) resulting in an emission of two photons (γ) with 511 keV in an angle of 180°.

In cases of kinetic energy at the level of the rest energy of heavier particles, the generation of those is also possible.

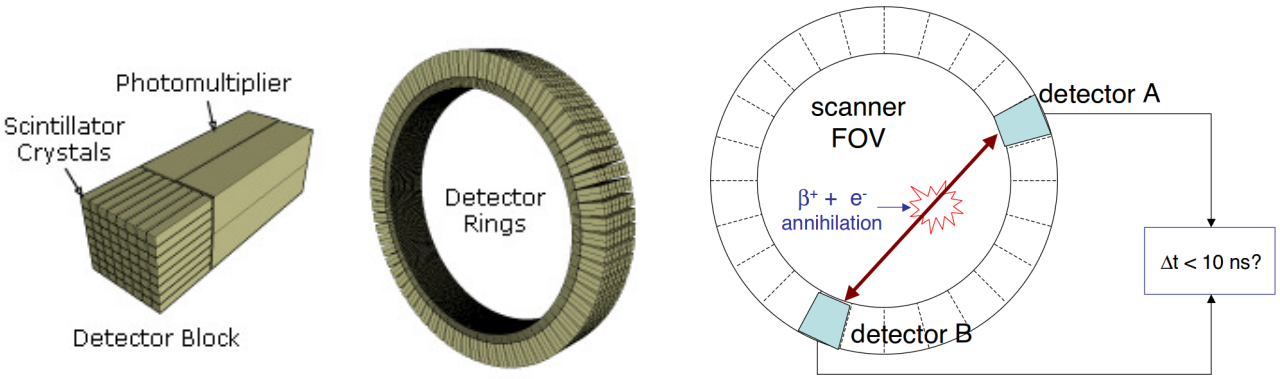
Assuming a beta decay inside humans, the β-particle gets scattered in tissue loosing kinetic energy till it can interact with an electron figure 2.2. The so called mean travel distance of positrons in tissue ranges from 1 to 2mm. The subsequent annihilation and the two photons emitted at an angle of 180° form the so called Line of Response (LOR).

2.2 Positron Emission Tomography System - PET

As mentioned in the introduction the radiotracer is administered to humans by injection, ingestion or inhalation. In order to measure the emitted coincidence photons a PET scanner typically is constituted as an cylindrical entity built up from scintillator crystals as can be seen in figure 2.3 (a). The scintillator crystals detect the 511keV annihilation photons and release optical photons of lower energy (~ 1eV). These pulses gets amplified by downstream photomultiplier tubes (PMT). As shown in figure 2.3 (b) two photons are accepted as signal (i.e. originating from the same annihilation event) if they get registered at both sides at the according angular position in a certain time interval. The respective angular range is determined by the so called field of view (FOW) which can be described as the spatial area where annihilations take place.

2.2.1 Noise Equivalent Count Rate

Due to the non-zero momentum of the positron at the moment of i annihilation the coincidence photons are not emitted exactly in a 180° angle which causes a smearing of ,measured activity. Additionally misclassification's and losses of photons occur and therefore lead to artefacts and a decrease of detected



(a) (b)

Figure 2.3: (a) left: detector block consisting of scintillator crystals and photomultiplier tubes. (a) right: detector blocks constituting a detector ring. (b): visualization of β decay with subsequent photon detection and registration due to the coincidence circuit.

activity.

- **true events (T)**: A true event occurs if both photons of a positron decay reach the two detectors without scattering.
- **single events**: If only one of the photon is detected it is called a single event. This effect leads to decreased detection of activity.
- **random events (R)**: If two single events also happen to pass the spatial and temporal coincidence criteria and therefore falsely get detected as originating from the same annihilation one speaks of a random event.
- **scatter events (S)**: In a scatter event at least one of the detected photons has undergone scattering prior to detection. Since scattering changers the direction of the photon the resulting coincidence event will most likely be assigned to the wrong LOR. Scatter events add a background to the true events, decreasing the contrast.

Figure 2.4 displays the counts for the true, scatter and random events as a function of activity concentration. At lower activities the true ones dominate. Due to dead time effects of the coincident circuit the curve then flattens at higher activities and even gets overtaken by random events due to increased misclassification. This non-linear behavior restricts the range of administered activity for PET. To determine this the Noise Equivalent Counting Rate curve (NECR) is calculated from true (T), scatter (S) and random (R) events as follows

$$\text{NECR} = \frac{T^2}{T + S + 2fR} \tag{2.4}$$

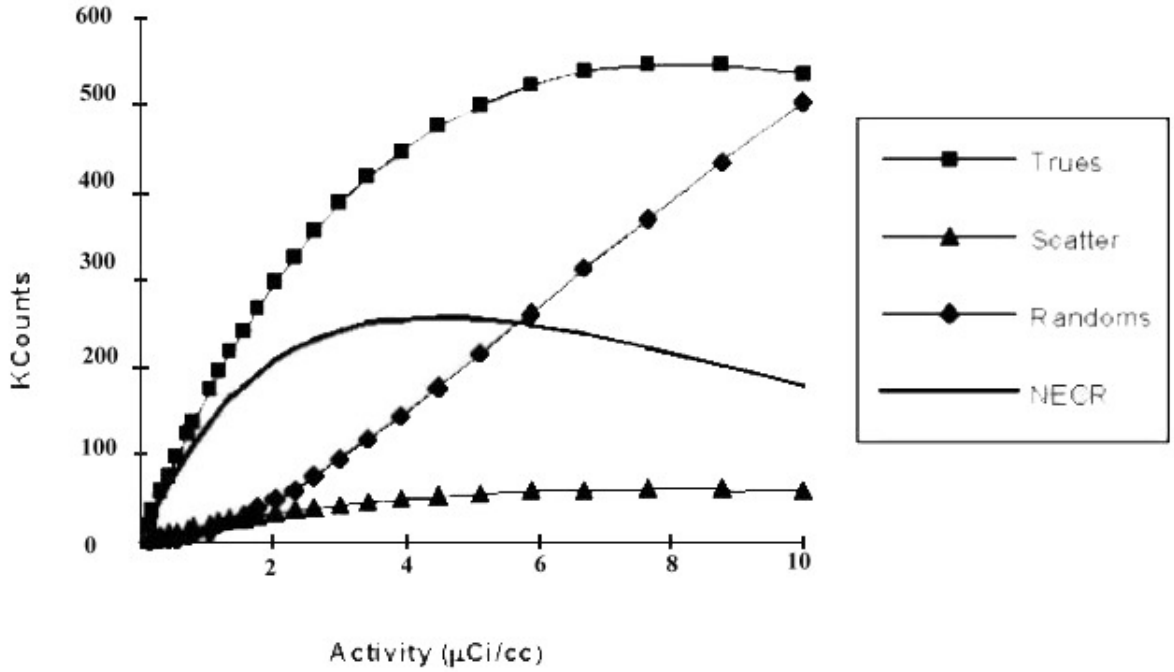


Figure 2.4: Example of true, random and scatter events of a PET device and the resulting Noise Equivalent Noise Rate curve (NECR).

with f being part of the object area which is projected to the projection plane. This curve is also shown in figure 2.4 which is used to determine the range of activity values which are linearly related.

3

Statistical Background

BASICALLY the problem of pattern recognition is to label some given data (e.g., PET voxels) according to specific membership rules. This amounts to find a mapping $f : \mathcal{X} \to \mathcal{Z}$ for some input vector $X \in \mathcal{X}$ to some output labels $Z \in \mathcal{Z}$. While there exist labeling concepts that are not based on probability theory, probabilistic algorithms have many advantages. E.g. not only a specific labeling can be guessed. With statistical methods the probabilities for each labeling of an input value is calculated grounded on mathematical and physical assumptions. Thereby a statistical analysis allows to incorporate naturally the treatment of PVE. Moreover the uncertainty which arises through noisy measurements can be modeled via an appropriate probability distribution.

Probability theory provides a mathematical framework for the quantification and manipulation of uncertainty and forms one of the central foundations for pattern recognition in this work. When combined with estimation theory and decision theory, it allows us to make predictions given all the information available to us, even though that information may be incomplete or ambiguous.

Bringing information theory into the picture, a distance measure for probability distributions can be established which provides a variational framework for approximating probability distributions. Combined with a Bayesian treatment of the distribution parameters, these can be controlled to avoid inaccurate parameter estimates for bad statistical ensembles.

To build more complex connections, graphical models combine probability theory with graph theory, thereby making dependencies among variables more explicit. In case of tree-structured graphs exact inference (e.g., belief propagation) can be applied to calculate marginal probabilities of a variable via integrating out the remaining variables in the tree. If the graph is not a tree, loopy belief propagation is a common method for solving pattern recognition problems. Using an information theoretical perspective, e.g. trying to maximize the Shannon entropy of a graphical model, a variational proceeding can serve to find lower bounds for probability distributions.

Probabilistic pattern recognition can be classified into supervised and unsupervised clustering. As mentioned in chapter 1, the aim of this work is to establish fully automatic algorithms without the need

for human interaction or training data, solely using parametric probability distributions. Therefore the various parameters determining the corresponding probability distributions have to be estimated from data. This can be done in an iterative way, alternating between a labeling step (i.e., estimating Z) and a parameter adjustment step.

3.1 Probability Theory

The mathematical theory of probability constitutes the basis for formulating the clustering processes considered here. Therefore we briefly discuss the basic concepts to deal with single random variables section 3.1.1. The expression is given only for discrete random variables. The formulation for the continuous case is obtained by simply substituting integrations for summations.

The concept of conditioning some variables on others requires more than one random variable, which is discussed in section 3.1.2. Thee formulations can be easily extended to the case of N random variables using the vector representation $X = (x_0, ..., x_N)^T$ simplifying notation.

For a detailed description of probability theory see [1, 13, 14, 27, 38].

3.1.1 Single Random Variable

Consider a random experiment with various outcomes ω, e.g., a die roll. The set of all possible outcomes is called sample space and denoted by Ω, which in case of a die roll is $\Omega = \{1, 2, 3, 4, 5, 6\}$. A specific event is given by a subset of the sample space (for die rolls e.g., $\{2\}, \{1, 6\}, \{1, 3, 5\}, ...$) which will be called Ω_S.

Moreover, there is a mapping which assigns to each event Ω_S a probability $\mathrm{P}\{\Omega_S\}$. This mapping fulfill the Kolmogorov axioms [cite]:

- $\mathrm{P}\{\Omega_S\}$ is real positive number in the interval [0,1]: $0 \leq \mathrm{P}\{\Omega_S\} \leq 1$;
- The probability of the certain event is: $\mathrm{P}\{\Omega\} = 1$;
- If the events Ω_{S1} and Ω_{S2} are mutually exclusive, then $\mathrm{P}\{\Omega_{S1} \cup \Omega_{S2}\} = \mathrm{P}\{\Omega_{S1}\} + \mathrm{P}\{\Omega_{S2}\}$.

A discrete random variable $x(\omega)$ is defined as a function of the experimental outcomes. For the die example, the random variable is given by $x(\omega) = \omega$ which yields $\Omega = \mathcal{X}$, with $\mathcal{X}$ meant to be the domain of the random variable. Subsets of the domain $\mathcal{X}$ are denoted by $\mathcal{S}$. A mapping function $p(\mathcal{S})$, mapping possible events to the probability space, will be called probability mass function (pmf). The pmf is constrained by the normalization condition (2nd Kolmogorov axion) and positivity requirement (1st Kolmogorov axiom) as:

$$\sum_{x \in \mathcal{X}} p(x) = 1, \qquad p(x) \geq 0. \tag{3.1}$$

Using the third Kolmogorov Axiom it is easily seen that the probability of the event $\mathcal{S}$ can be written as

$$p(\mathcal{S}) = \sum_{x \in \mathcal{S}} p(x) = \mathrm{P}\{x \in \mathcal{S}\}. \tag{3.2}$$

In case of continuous random variables, the pendant to the pmf is the probability density function (pdf). For further considerations both, pmf and pdf will be denoted by $p(x)$. Moreover, most definitions are presented for discrete random variables; however, the corresponding expressions for the continuous case can be derived by changing summation with integration.

One of the most important operations in the field of probability theory is that of finding weighted averages of functions. The average value of some function $\phi(x)$ under a probability distribution $p(x)$ is called the expectation of $\phi(x)$ and will be denoted by $\mathbb{E}\{\phi(x)\}$. It is given by

$$\mathbb{E}\{\phi(x)\} = \sum_{x\in\mathcal{X}} \phi(x)p(x). \tag{3.3}$$

Special cases are the moments and the central moments,

$$\mathbb{E}\{x^k\} = \sum_{x\in\mathcal{X}} x^k p(x), \qquad \mathbb{E}\{(x-\mathbb{E}\{x\})^k\} = \sum_{x\in\mathcal{X}} (x-m_x^1)^k p(x). \tag{3.4}$$

Using $k = 1$ within the moment relations yields the mean value μ, whereas using $k = 2$ with the central moment relations gives the variance σ^2.

In cases where the pdf or pmf is unknown, the moments and central moments can be estimated from random experiments. If an experiment is repeated J times yielding samples $x^{(1)}, x^{(2)}, ..., x^{(J)}$, the so called sample moments or empirical expectations are given via

$$\hat{m}_x^k = \frac{1}{J}\sum_{j=1}^{J} x^{(j)k}, \qquad \hat{m}_{x-m_x^1}^k = \frac{1}{J}\sum_{j=1}^{J} (x^{(j)} - \hat{m}_x^1)^k. \tag{3.5}$$

3.1.2 Two Random Variables - Conditioning

To introduce the concepts of statistical dependence and inference a second random variable is assumed. Two random variables x_1 and x_2 can equivalently be viewed as a two-dimensional random vector $(x_1, x_2)^T$. For these random variables, a joint probability distribution is defined by $p(x_1, x_2)$. Assuming same domain for x_1, x_2, the Kolmogorov Axioms are written as

$$1 = \sum_{(x_1,x_2)\in\mathcal{X}^2} p(x_1, x_2), \qquad p(x_1, x_2) \geq 0. \tag{3.6}$$

Furthermore, the expectation of some function of two random variables gets

$$\mathbb{E}\{\phi(x_1, x_2)\} = \sum_{(x_1,x_2)\in\mathcal{X}^2} \phi(x_1, x_2)p(x_1, x_2). \tag{3.7}$$

From (3.7), all moment and central moment relations can be obtained. Integrating out one of the random variables yields the probability distribution of the respective other one-dimensional distribution,

$$\sum_{x_1\in\mathcal{X}} p(x_1, x_2) = p(x_2), \tag{3.8}$$

which is then called a marginal distribution of the joint distribution $p(x_1, x_2)$. (3.8) is termed the summation rule.

Furthermore, consider two random experiments with possible outcomes $\omega \in \Omega$ and $\xi \in \Xi$. A conditional probability measure $P\{\omega \mid \xi\}$ is defined as the probability that some outcome ω occurs assuming that some other outcome ξ already has occurred. An axiomatic definition is given via the

joint and prior probability $P\{\omega \cap \xi\} = P\{\omega \mid \xi\}P\{\xi\}$. As an example consider the die roll from the beginning of this section and assume two dice per toss. With this the number of possible outcomes of the experiment raises to 6×6. With a faked die that yields the same outcome ξ^* for any toss, the unique distribution of this die roll has changed to be a fixed (deterministic) quantity with $P\{\xi = \xi^*\} = 1$. Therefore the probability space of the conditional probability for ω given ξ^* reduces to 6. If again considering ξ as random and trying to calculate the joint probability $P\{\omega \cap \xi\}$ from the conditional $P\{\omega \mid \xi\}$, the increase of the sample space has to be accommodated by multiplying the conditionals with the respective prior probability, in the case above with $P\{\xi\}$.

With this the product rule for two random variables $x_1(\omega)$ and $x_2(\xi)$ can be defined according to

$$p(x_1, x_2) = p(x_1 \mid x_2)p(x_2) = p(x_2 \mid x_1)p(x_1). \tag{3.9}$$

Using (3.9), one conditional pdf can be expressed in terms of the other conditional pdf and the two one-dimensional marginal pdfs:

$$p(x_1 \mid x_2) = p(x_2 \mid x_1)\frac{p(x_1)}{p(x_2)}, \tag{3.10}$$

which is called the Bayesian theorem and plays a central role in probabilistic pattern recognition. Expressing the denominator via $p(x_2 \mid x_1)$ and $p(x_1)$ using the summation rule (3.8), the Bayesian theorem can be written as

$$p(x_1 \mid x_2) = \frac{p(x_2 \mid x_1)p(x_1)}{\sum_{x_1} p(x_2 \mid x_1)p(x_1)}. \tag{3.11}$$

Finally, having defined conditional probabilities, the conditional expectation of a random variable x_1 given an other random variable x_2 can be calculated as

$$\mathbb{E}\{\phi(x_1) \mid x_2\} = \sum_{x_1 \epsilon \mathcal{X}} \phi(x_1)p(x_1 \mid x_2). \tag{3.12}$$

As mentioned in the introduction of section 3.1, the formulation is extended to N-dimensional random variables. So introducing a N-dimensional random vector as

$$X = (x_1, x_2, ..., x_N)^T, \tag{3.13}$$

all concepts introduced so far can easily be extended to the N-dimensional case.

3.2 Information Theory

With respect to the scope of this work, information theory is useful for providing a measure of dissimilarity between true and estimated probability distributions. Moreover the concept of entropy will

The central definition in information theory is that of the entropy. A physical definition of entropy is the one of order, where a system which is in a highly ordered state (e.g., a box with a separating plane enclosing gas in one chamber and vacuum in the other) has low entropy, whereby a chaotic state of the system (e.g., removing the separating plane of the afore-mentioned box, the gas will expand to

fill the entire box uniquely distributed) has high entropy. So in a physical sense, the lower the entropy, the higher the order/energy contained in the system. But as one can see considering the Boltzmann distribution [1]

$$p(X) = \frac{1}{Z} e^{-\beta E}, \tag{3.14}$$

with E being the energy, the higher the energy of a system, the less probable it gets. To define the entropy, let X be a random vector as in (3.13) with probability mass function $p(X)$. The set of all possible outcomes/states of X is the alphabet $\mathcal{X} = \{X^{(1)}, X^{(2)}, ..., X^{(M)}\}$ with cardinality $|\mathcal{X}| = M$ (e.g., for a die roll $\mathcal{X} = \{1, 2, 3, 4, 5, 6\}$ and $|\mathcal{X}| = 6$).

Assuming a realization of X, the information received by observing a sample $X^{(i)}$ is expressed through

$$h(X^{(i)}) = \log \frac{1}{p(X^{(i)})}. \tag{3.15}$$

With this definition, rare symbols having low probability carry more information (physically, improbable states would have more energy) than the more probable ones. The entropy is then defined as the expected information,

$$\mathrm{H}(X) = \mathbb{E}\{h(X)\} = -\sum_{X \in \mathcal{X}} p(X) \log p(X) = -\sum_{m=1}^{M} p(X^{(m)}) \log p(X^{(m)}). \tag{3.16}$$

Considering the die roll and assuming equal probability (maximal randomness) for each outcome, the entropy is calculated as $-6 \ln \frac{1}{6} = 10.75$ whereas if one outcome is determined (deterministic case - no uncertainty), the entropy is calculated as $-\ln \frac{1}{1} = 0$. This means that in a deterministic case no information is gained by drawing a sample from a random experiment (a particle system without the possibility of differing the system states missing external energy is in equilibrium - state of minimal energy). In case of two random variables $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$ with joint probability $p(X, Z)$, the joint entropy is defined as

$$\mathrm{H}(X, Z) = -\sum_{X \in \mathcal{X}} \sum_{Z \in \mathcal{Z}} p(X, Z) \log p(X, Z), \tag{3.17}$$

which is a symmetric quantity $\mathrm{H}(X,Z) = \mathrm{H}(Z,X)$.

The mutual information that random variable (X, Z) carry about each other, likewise a symmetric quantity, is defined as

$$\mathrm{I}(X, Z) = \sum_{X \in \mathcal{X}} \sum_{Z \in \mathcal{Z}} p(X, Z) \log \frac{p(X, Z)}{p(X)p(Z)}, \tag{3.18}$$

Now to introduce the aforementioned dissimilarity measure between a true and an estimated probability distribution, consider some unknown distribution $p(x)$, and suppose that we have modelled this using an approximating distribution $q(x)$. If we use $q(x)$ to construct a coding scheme for the purpose of transmitting values X to a receiver, then the average additional amount of information required to specify the value of X as a result of using $q(x)$ instead of the true distribution $p(x)$ is given by

$$\mathrm{KL}(q \parallel p) = -\sum_{X \in \mathcal{X}} p(X) \log \frac{q(X)}{p(X)} \tag{3.19}$$

[1] The Bolzmann distribution is a probability measure for the various states of a systems known from statistical mechanics.

$$= \; -\sum_{X\in\mathcal{X}} p(X)\log q(X) - \Big(-\sum_{X\in\mathcal{X}} p(X)\log p(X)\Big).$$

This is an asymmetric quantity also termed relative or cross entropy. The term measure as written in the beginning of section 3.2 have to be used carefully because in general the triangle inequality is not applicable. However the positivity constraint

$$\mathrm{KL}(q \parallel p) \geq 0 \tag{3.20}$$

is fulfilled.

For a detailed description of information theory see [1, 8].

3.3 Estimation Theory

In principle, estimation theory is concerned with the problem of determining various parameters of data models, collectively denoted as Θ, from a set of observed data X. A variety of applications are known from radar, sonar, speech recognition, image analysis, biomedicine, communication and more. Typically, parameter estimation methods are placed within a statistical framework, requiring different types of prior knowledge.

In determining good estimators for the parameters, the first step is to mathematically model the data. Because the data are inherently random, they will be described by their pdf or pmf, see section 3.1. Using parametrized probability measures (e.g., a Gaussian distribution has the parameter vector $\Theta = (\mu, \sigma)$), the respective unknown parameters should be inferred from the observations fitting the probability measures to the data. Typically such problems are formulated as optimization approaches minimizing or maximizing an objective function. The domains of the parameters as well as dependencies among them can be encompassed using Lagrange multipliers.

During the further procedure the estimator of Θ is called $\hat{\Theta}$. With this, the deviation in using the estimator $\hat{\Theta}$ instead of the true value is given by $E = \hat{\Theta} - \Theta$, which is called the estimation error. In trying to find good parameter estimators, the estimation error can be minimized so that the estimator is close to the true parameter.

For an unbiased estimator of a deterministic parameter Θ, the estimation error has zero average,

$$\mathbb{E}\{E\} = 0 \quad \longrightarrow \quad \mathbb{E}\{\hat{\Theta}\} = \Theta, \tag{3.21}$$

meaning that the estimator will attain the true parameter on average (as the sample size grows). If Θ is a random parameter, unbiasedness means $\mathbb{E}\{\hat{\Theta}\} = \mathbb{E}\{\Theta\}$. A natural optimization criterion in searching good estimators is the mean square error (MSE), which measures the average mean squared deviation of the estimator from the true value,

$$\mathrm{MSE}\{\hat{\Theta}\} = \mathbb{E}\{E^2\} = \mathbb{E}\{(\hat{\Theta} - \Theta)^2\}. \tag{3.22}$$

For an unbiased estimator as defined in (3.21) the MSE is equal to the variance of the estimation error

$$\mathrm{MSE}\{\hat{\Theta}\} = \mathbb{E}\{(E - \underbrace{\mathbb{E}\{E\}}_{=0})^2\} = \mathrm{var}\{E\}. \tag{3.23}$$

The variance, $\mathrm{var}\{E\}$, corresponds to the second order central moment (3.4). Augmenting (3.22) with $\pm\mathbb{E}\{(\hat{\Theta})\}^2$, the MSE of $\hat{\Theta}$ can be decomposed like

$$\mathrm{MSE}\{\hat{\Theta}\} = \underbrace{(\mathbb{E}\{\hat{\Theta}\} - \Theta)^2}_{\mathrm{bias}\{\hat{\Theta},\Theta\}} + \underbrace{\mathbb{E}\{(\hat{\Theta} - \mathbb{E}\{\hat{\Theta}\})^2\}}_{\mathrm{var}\{\hat{\Theta}\}} \tag{3.24}$$

It is recognized that the MSE is built up from errors due to variance as well as due to a term called bias (i.e., systematic error), which vanishes if the estimated parameter is equal the true parameter. For a detailed description of estimation theory see [25].

3.3.1 Bayesian Estimation

In classical estimation the data X is assumed as random but the parameter vector Θ as deterministic. There again in bayesian estimation both, the data X and the parameter Θ are of random nature. It is therefore assumed that there exists some prior information about the parameters established via prior probabilities $p(\Theta)$. Moreover the dependency of the data X on the parameter Θ and vice versa is given by the conditional $p(x \mid \Theta)$ and posterior probability $p(\Theta \mid X)$. Repeating the product rule (3.9) and the bayesian theorem (3.10), this quantities are directly related as

$$p(X, \Theta) = p(x \mid \Theta)p(\Theta) = p(\Theta \mid X)p(X) \tag{3.25}$$

and

$$p(\Theta \mid X) = p(X \mid \Theta)\frac{p(\Theta)}{p(X)}. \tag{3.26}$$

With Bayesian estimation, the objective function of the optimization problem is constructed by averaging over some cost $\mathrm{C}(E)$ on the estimation error, which should be as small as possible so that the estimator $\hat{\Theta}$ is closest to the true value Θ, i.e.

$$\hat{\Theta}_{\mathrm{B}}(X) = \arg\min_{\hat{\Theta}} \mathbb{E}\big\{ \underbrace{\mathrm{C}(\hat{\Theta} - \Theta)}_{\mathrm{Cost}\ \mathrm{C}(E)} \big\}. \tag{3.27}$$

Using the product rule (3.25) and substituting for the joint probability $p(X, \Theta)$ emerging due to averaging, since $p(X)$ is positive, the average cost is minimized by minimizing $\int_\Theta \mathrm{C}(E)p(\Theta \mid X)d\Theta$ for each X separately. So the Bayesian estimator can be formulated as the one minimizing the conditional expectation of the cost function $\mathrm{C}(E)$ given X

$$\hat{\Theta}_{\mathrm{B}}(X) = \arg\min_{\hat{\Theta}} \mathbb{E}\{\mathrm{C}(E) \mid X\}. \tag{3.28}$$

Using again (3.25) and calculating the conditional expectation via $p(X \mid \Theta)p(\Theta)$ it is obvious that if X and Θ are statistically independent (worst case in statistical estimation), X caries no information about Θ and so $\hat{\Theta}_{\mathrm{B}}(X)$ gets dummy.

An important cost function is the mean square error (MSE) which is given by $\mathrm{C}(E) = \mathbb{E}\{E^2\}$. Using (3.27) the minimum mean square error (MMSE) gets

$$\hat{\Theta}_{\mathrm{MMSE}}(X) = \arg\min_{\hat{\theta}} \mathrm{MSE}_{\hat{\Theta}} = \mathbb{E}\{\Theta \mid X\}. \tag{3.29}$$

In case where the parameter Θ and the data X are jointly Gaussian, $Z = (X, \Theta)^T \approx \mathcal{N}(\mu_Z, \Sigma_Z)$, the MMSE estimator $\Theta_{\text{MMSE}} = \mathbb{E}\{\Theta \mid X\}$ is given by the conditional mean (A.21) as

$$\hat{\Theta}_{\text{MMSE}} = \mathbb{E}\{\Theta \mid X\} = \mathbb{E}\{\Theta\} + \Sigma_{X,\Theta}^{\text{T}} \Sigma_X^{-1}(X - \mathbb{E}\{X\}). \tag{3.30}$$

Here $\Sigma_{X,\Theta}$ and Σ_X are block matrices which constitutes the covariance matrix Σ_Z.

Another important cost function is the uniform cost function (case of no prior knowledge $p(\Theta)$), which assigns equal cost to all error components whose magnitude is a above a certain threshold $\delta > 0$. $\delta \to 0$, the estimates $\hat{\Theta}$ are called maximum a-posterior (MAP) estimates which can be written in various forms. A nice interpretation can be given to the following representation

$$\hat{\Theta}_{\text{MAP}}(X) = \arg\max_{\Theta} \big\{ \ln p(X \mid \Theta) + \ln p(\Theta) \big\}. \tag{3.31}$$

This equation separates the influence of the data X via $p(X \mid \Theta)$ and the influence of the prior $p(\Theta)$. If the prior $p(\Theta)$ is a flat function, it will not greatly influence the position of the maximum in (3.31) and the optimization simplifies according to

$$\hat{\Theta}_{\text{MAP}}(X) \approx \arg\max_{\Theta} \ln p(X \mid \Theta). \tag{3.32}$$

It is mentioned that the formulation (3.32) of the MAP estimator is approximately equal to the maximum likelihood estimator (3.40), $\hat{\Theta}_{\text{MAP}}(X) \approx \hat{\Theta}_{\text{ML}}(X)$, which will be presented in section 3.3.2.

3.3.1.1 Efficient Bayesian Estimators

As mentioned above, an estimator $\hat{\Theta}$ is said to be unbiased if on average the expectation of the estimation error equals zero, $\mathbb{E}\{E\} = 0$, meaning that the expectation of the estimator is equal to the expectation of the true parameter, $\mathbb{E}\{\hat{\Theta}\} = \mathbb{E}\{\Theta\}$, cf. (3.21). As can be seen from (3.23), this relates directly the MSE with the error variance of the estimation error as

$$\text{MSE}_{\hat{\Theta}} = \text{var}\{E\}. \tag{3.33}$$

Assume that the first and second derivatives of the joint probability $p(X, \Theta)$ exist and are absolutely integrable. Then it can be shown that an unbiased efficient estimator exist if its MSE/error variance attain[2] the Cramer-Rao lower bound (CRLB) which is defined by the inverse of the bayesian information L (a pendant to the Fisher information known from classical estimation):

$$\text{MSE}_{\hat{\Theta}} = \text{var}\{E\} \geq \mathbb{E}\left\{ \Big(\underbrace{\frac{\partial}{\partial \Theta} \ln p(X, \Theta)}_{\text{score}} \Big)^2 \right\}^{-1} = \frac{1}{L}, \tag{3.34}$$

The score introduced above is a quantity which describes how strong $p(\Theta \mid X)$ depends on Θ. The score is assumed to be related to the accuracy with which the parameter Θ can be estimated. With this it can be stated that if an unbiased efficient estimator exists, its derivative can be written as

$$\frac{\partial}{\partial \Theta} \ln p(X, \Theta) = K[\text{g}(X) - \Theta], \tag{3.35}$$

[2]The attainment of the Cramer-Rao lower bound by the MSE/error variance means equality in (3.34).

with some constant factor $K \geq 0$ and some function g. It follows that the estimator is given by

$$\hat{\Theta}(X) = g(X), \tag{3.36}$$

and the MSE/error variance reads

$$\mathrm{MSE}_{\hat{\Theta}} = \mathrm{var}\{E\} = \frac{1}{L} = \frac{1}{K} \tag{3.37}$$

As shown by [], an efficient estimator exists just in cases where the conditional probability $p(X \mid \Theta)$ is Gaussian. Except for pathological cases, as the data size grows $N \to \infty$, L goes to infinity $L \to \infty$ whereby the MSE/error variance vanishes and the unbiased efficient estimator converges to the true value.

Moreover if an unbiased efficient estimator exists, it is the MMSE and simultaneously the MAP estimator.

3.3.2 Classical Estimation

As mentioned, in classical estimation Θ is assumed to be deterministic, which represents the case of no prior knowledge about Θ. As can be seen from (3.24) the first term building up the MSE is dependent on the true parameter Θ. In Bayesian estimation, where the parameters were treated as random variables, Θ can be integrated out. In classical estimation this is no longer possible. Therefore classical estimation suffers from the fundamental difficulty that everything depends on the unknown (true) parameter Θ, which can not be integrated out as in bayesian estimation.

Some workaround for this problem is suggested by a decomposition of the mean square error (3.22), which is only valid in a classical sense with the parameter Θ assumed deterministic. Due to Θ being deterministic,

$$\mathrm{var}\{\hat{\Theta}\} = \mathrm{var}\{\hat{\Theta} - \Theta\} = \mathrm{var}\{E\} \tag{3.38}$$

and so the variance of $\hat{\Theta}$ is equal the variance of the estimation error. But as shown in (3.23), if the estimator is unbiased the error variance is directly related to the MSE which leads to relating the variance of the parameters directly to their MSE

$$\mathrm{MSE}_{\hat{\Theta}} = \mathrm{var}\{\hat{\Theta}\}. \tag{3.39}$$

So instead of minimizing the MSE or some other measure of the estimation error $\mathrm{C}(E)$, it can be tried to find the unbiased estimator that has minimum variance among the class of all unbiased estimators. Such an unbiased estimator, if it exists, is called the minimum variance unbiased (MVU). If an efficient estimator does not exist it is still possible that an MVU estimator exists. Using the concept of complete sufficient statistics and the Rao-Blackwell-Lehmann-Steffe theorem, it is sometimes possible to determine the MVU by simply inspection of the pdf. Further, a CRLB as in (3.35) can be derived in an identical manner as for Bayesian estimation.

A turn the crank method for finding good parameter estimates is using maximum likelihood (ML) estimation which is not necessarily optimal (MVU) for finite datasets but asymptotically optimal

(optimal as the size of the dataset grows). It's simply defined as the θ which maximizes the likelihood (or log-likelihood) function

$$\hat{\Theta}_{\mathrm{ML}}(X) = \arg\max_{\Theta} \big\{p(X;\Theta)\big\} = \arg\max_{\Theta} \big\{\ln p(X;\Theta)\big\}. \tag{3.40}$$

If the likelihood function is differentiable with respect to Θ for all X, the ML estimator is obtained by solving the likelihood equation

$$\frac{\partial}{\partial\Theta} \ln p(X;\Theta) = 0. \tag{3.41}$$

In general, this optimization problem can only be solved by numerical techniques (e.g., in cases of modeling the data using normalization constants being convex functions as in section 3.4).

3.3.2.1 Expectation Maximization (EM) Algorithm

A slightly different estimation problem is given by the expectation maximization algorithm. Actually, the iterative structure of this procedure is a common property of the algorithms proposed in this work. For a detailed description of the EM algorithm and its extensions see [1, 2, 10, 15, 16, 32, 34, 36].

Consider an estimation problem where not only the parameter Θ has to be derived from the observed data X, but also a labeling problem has to be solved for an unobserved (hidden) label matrix $\mathbf{Z}$. Moreover, the label matrix $\mathbf{Z}$ to be estimated is considered as ground truth of which the observed data X are a distorted version. Therefore X is said to be the incomplete data whereas the set $\{\mathbf{Z},X\}$ is termed the complete data.

The idea now is to formulate a ML estimation problem (3.40) for the parameter Θ via the complete data, $\hat{\Theta}^{ML} = \arg\max_{\Theta} \ln p(\mathbf{Z}, X;\Theta)$, instead of using $p(X;\Theta)$. But unfortunately, $\mathbf{Z}$ is not observed. Therefore, instead of maximizing $\ln p(\mathbf{Z}, X;\Theta)$ with respect to Θ directly, $\ln p(\mathbf{Z}, X;\Theta)$ is estimated from X and this estimate is maximized regarding Θ. To estimate $\ln p(\mathbf{Z}, X;\Theta)$ from X, the MMSE[3] (3.29) is used with which an iterative algorithm can be formulated:

- Expectation (E) step: Calculate the MMSE (3.29) estimator of $\ln p(\mathbf{Z}, X;\Theta)$ by using the previous parameter value $\Theta^{(i)}$:

$$Q(\Theta, \Theta^{(i)}) \equiv \mathbb{E}\{\ln p(\mathbf{Z}, X;\Theta) \mid X; \Theta^{(i)}\}. \tag{3.42}$$

- Maximization (M) step: Compute the ML estimator $\Theta_{\mathrm{ML}}^{(i+1)}$ (3.40) by maximizing the expectation $Q(\Theta, \Theta^{(i)})$ with respect to Θ

$$\Theta^{(i+1)} \equiv \arg\max_{\Theta} Q(\Theta, \Theta^{(i)}). \tag{3.43}$$

These steps get repeated till convergence. Proofs of convergence can be found in [34, 49].

Hence a first parameter estimate is done filling in initial values for the label matrix $\mathbf{Z}$. The latter are then updated by their predicted values using the initial estimate of the parameter Θ. The parameter is then re-estimated, and so on.

[3] Note that both, the hidden variable $\mathbf{Z}$ and the data X, are considered as random.

3.3.3 Bayesian Expectation Maximization

Instead of calculating the ML estimator of Θ from the complete data (3.43), we can formulate the EM procedure using MAP estimation (3.31). Using the complete data $\{X, \mathbf{Z}\}$, the posterior distribution of Θ can be approximated according to

$$p(\Theta \mid X, Z) \propto p(X, Z \mid \Theta)p(\Theta). \tag{3.44}$$

Therewith the expectation step for a MAP estimator can be formulated as

$$\mathbb{E}\{\ln p(\mathbf{Z}, X \mid \Theta) + \ln p(\Theta) \mid X; \Theta^{(i)}\} = Q(\Theta, \Theta^{(i)}) + \ln p(\Theta). \tag{3.45}$$

with $Q(\Theta, \Theta^{(i)})$ defined by (3.42). Hence the estimation step is calculated as with classical EM algorithm. However the maximization step differs in that the objective function of the maximization process is augmented by the log prior density, $\log p(\Theta)$.

If the logarithmic prior of Θ can be written as

$$\log p(\Theta) = -\xi K(\Theta), \tag{3.46}$$

the ML estimation problem is called Maximum Penalized Likelihood estimation (MPLE). Hence K is the penalty function and ξ is an additional (smoothing) parameter which have to be estimated along with Θ.

3.4 Graphical Models

Combining probability theory with graph theory more complex models become handy which are called graphical models. With graphical models large scale statistical systems can be represented by the construction of graphs, with random variables as nodes and their interactions as links (edges) between those nodes. According to the interactions defined between the nodes/random variables, factorization properties are given to the joint probability of the whole system making it amenable for inference of local marginal or conditional probabilities.

As mentioned in chapter 1, PVE is caused due to the finite resolution of the image as well as due to the spill in/spill out relation. Therefore it is assumed that modeling the dependencies among neighbouring voxels will help to deal with PVE and moreover provides a framework for more complex patient data sets.

For a detailed description of graphical models see [1, 4, 24, 28, 31, 32, 35, 39, 47, 48]

3.4.1 Graph Theory

In general a graph consists of a set of vertices $\mathcal{V}$ which are connected by edges constituting the set $\mathcal{E}$. Directed graphs and undirected graphs are distinct in that they have directed and undirected edges, respectively. An example of an undirected graph is shown in figure 3.1 (left) with 9 vertices

$\mathcal{V} = \{v_1, v_2, ..., v_9\}$ and 12 edges $\mathcal{E} = \{e_{12}, e_{23}, ..., e_{89}\}$, connecting unordered pairs of vertices in $\mathcal{V}$. Due to indirectness of the edges, $e_{12} = \{v_1, v_2\}$ is equal $e_{21} = \{v_2, v_1\}$.

For undirected graphs, the neighbour set $\mathcal{N}(v_n)$ of a vertex v_n is written as

$$\mathcal{N}(v_n) = \{v_m \mid e_{nm} \in \mathcal{E} \cup n \neq m\}, \tag{3.47}$$

addressing vertices v_m connected with vertex v_n by an edge $e_{nm} \in E$. Considering the vertex v_5 from figure 3.1, the neighbour set is given by $\mathcal{N}(v_5) = \{v_2, v_4, v_6, v_8\}$.

Moreover for an undirected graph, cliques $\mathcal{C}_l$ are defined as subsets of $\mathcal{V}$ consisting of vertices $v_l \in \{v_1, ..., v_L\} \subset \mathcal{V}$ which are all mutual neighbours

$$\mathcal{C}_l = \{v_l \mid v_l \in \mathcal{V} \cup \forall(v_{l_1} \neq v_{l_2}) \rightarrow e_{l_1 l_2} \in \mathcal{E}\}. \tag{3.48}$$

For example the clique sets in figure 3.1 comprises the 12 ordered pairs contained in the edge set of this graph $\mathcal{C}_{l_1} \cup ... \cup \mathcal{C}_{l_{12}} = \mathcal{E}$.

Mathematically, an undirected graph $G(\mathcal{V}, \mathcal{E})$ is defined as a pair of sets $\mathcal{E}$ and $\mathcal{V}$ such that $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. Here the edge set $\mathcal{E}$ comprises unordered pairs of elements of the vertex set $\mathcal{V}$. There again, a directed graph $G(\mathcal{V}, \mathcal{E})$ is defined as a pair of sets $\mathcal{E}$ and $\mathcal{V}$ such that $\mathcal{E}$ contains ordered pairs $e_{mn} = [v_m, v_n]$ which is highlighted by drawing arrows on the edge from vertex v_m to v_n. In this context, v_m is said to be a parent of the child v_n defining the parent set of the vertex v_n as $\pi(v_n) = \{v_m \mid e_{mn} = [v_m, v_n] \in \mathcal{E}\}$.

Further a path in a graph G is defined as an ordered sequence of vertices $v_1, ..., v_P$ such that each successive elements form an edge. If $v_1 = v_P$, the path is called a cycle.

Moreover, a graph is said to be connected if for any partition $\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2, \mathcal{V}_1 \cap \mathcal{V}_2 = 0$, there exists at least one vertex $v_1 \in \mathcal{V}_1$ and $v_2 \in \mathcal{V}_2$ such that v_1 and v_2 form an edge. With other words it has to

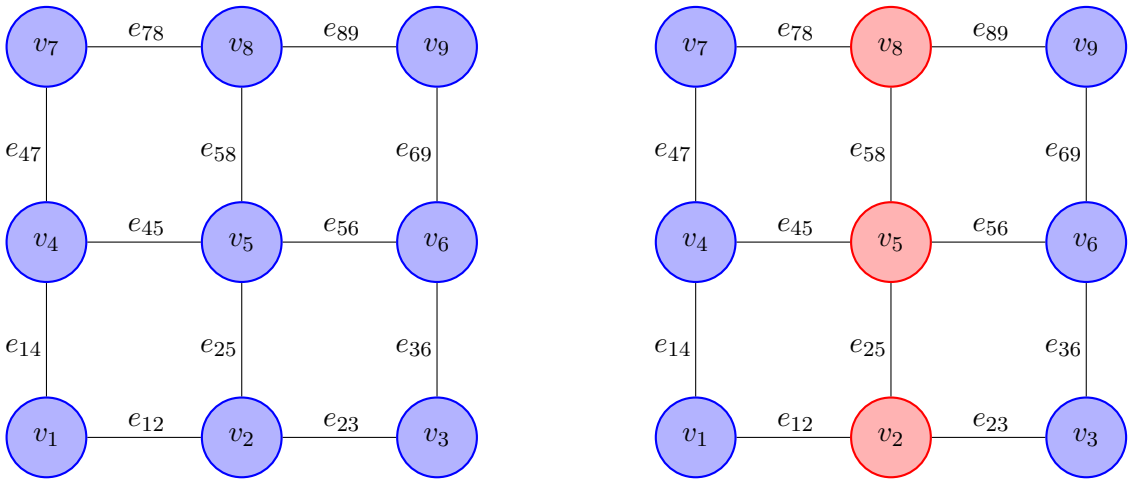


Figure 3.1: (a) Vertex set $\mathcal{V} = \{v_1, v_2, ..., v_9\}$ with edge set $\mathcal{E} = \{e_{12}, e_{23}, ..., e_{89}\}$. (b) Vertex set $\mathcal{V}$ divided into three disjoint subsets $\mathcal{V}_1, \mathcal{V}_2$ and $\mathcal{S}$ with $\mathcal{S}$ separating $\mathcal{V}_1$ and $\mathcal{V}_2$.

be possible to connect any two vertices in $\mathcal{V}$ by a path in $\mathcal{V}$. A separator $\mathcal{S}$ is a set of vertices such that deleting $\mathcal{S}$ from G renders the graph disconnected. In figure 3.1(b), a separator set $\mathcal{S}$ is defined by the red vertices; when deleted, $\mathcal{S}$ cuts all ways from $\mathcal{V}_1$ to $\mathcal{V}_2$.

A connected graph without cycles (loops) is called a tree.

3.4.2 Probability Distributions on Graphs

An important concept in the context of graphical models is the one of statistical independence and conditional independence. Two random variables, x_1 and x_2, are said to be statistical independent if their joint probability factorizes according to

$$p(x_1, x_2) = p(x_1)p(x_2). \tag{3.49}$$

Using Bayes theorem (3.10), this can also be written as

$$p(x_1 \mid x_2) = p(x_1); \quad \text{and} \quad p(x_2 \mid x_1) = p(x_2). \tag{3.50}$$

Further more x_1 and x_2 are conditional independent given a third random variable x_3 if

$$p(x_1, x_2 \mid x_3) = p(x_1 \mid x_3)p(x_2 \mid x_3). \tag{3.51}$$

With undirected graphs, the independent statements are directly related to connectivity properties of the underlying graph as discussed in the last paragraph of section 3.4.1. Assuming the vertex set $\mathcal{V}$ consists of random variables v_n, without getting concrete about a specific probability distribution, we impose following relation on the graph shown in figure 3.1 (b)

$$p(v_1, v_4, v_7, v_3, v_6, v_9 \mid v_2, v_5, v_8) = p(v_1, v_4, v_7 \mid v_2, v_5, v_8)p(v_3, v_6, v_9 \mid v_2, v_5, v_8). \tag{3.52}$$

So, given the separator set $\mathcal{S}$, the disconnected parts of the graph in figure 3.1 (b) (i.e. $\mathcal{V}_1$ and $\mathcal{V}_2$) are considered as conditional independent. Applying again Bayesian theorem (3.10), equivalent conditions for (3.52) are

$$p(v_1, v_4, v_7 \mid v_3, v_6, v_9, v_2, v_5, v_8) = p(v_1, v_4, v_7 \mid v_2, v_5, v_8) \tag{3.53}$$

and

$$p(v_3, v_6, v_9 \mid v_1, v_4, v_7, v_2, v_5, v_8) = p(v_3, v_6, v_9 \mid v_2, v_5, v_8). \tag{3.54}$$

A short hand notation for the statistical independence statements (3.49) and (3.50) reads $x_1 \perp\!\!\!\perp x_2$, respectively for the conditional independence statement (3.51) i.e. $x_1 \perp\!\!\!\perp x_2 \mid x_3$. Hence the conditional independence statement for the graph in figure 3.1 (b), (3.52), correspond to $(v_1, v_4, v_7) \perp\!\!\!\perp (v_3, v_6, v_9) \mid (v_2, v_5, v_8)$.

3.4.2.1 Markov Random Fields

Bringing probability theory into the picture, the vertex set $\mathcal{V} = \{x_1, x_2, ..., x_N\}$ is defined to contain the random variables X whereas the edge set describes the statistical (in-)dependencies of these variables, making the statistical behaviour of systems more obvious. A probability distribution is said to be graphical over the graph G if it satisfies all conditional independence statements described by G. In case of undirected graphs, also called Markov Random Fields (MRF), it can be shown that the joint probability of the whole data vector factorizes over the cliques of the graph (3.48) yielding

$$p(X) = \frac{1}{Z(\phi(\mathcal{C}_l))} \prod_l \phi_l(\mathcal{C}_l), \quad Z(\phi(\mathcal{C}_l)) = \sum_X \prod_l \phi_l(\mathcal{C}_l). \tag{3.55}$$

Here, ϕ_l are called the (clique) potential functions and Z is a normalization constant named partition function. If $\phi > 0$, $p(X)$ satisfies all conditional independence statements described by the underlying graph G (Hammersley-Clifford Theorem). E.g., as discussed in section 3.4.2, for the graph in figure 3.1 (b) we demand the probability distribution p to fulfil (3.52), (3.53) and (3.54).

A convenient representation for the clique potentials in (3.55) is obtained by employing energy functions like $\phi_l(\mathcal{C}_l) = \exp(-E_l(\mathcal{C}_l))$, yielding a Boltzmann distribution with the energy defined as $E_l(\mathcal{C}_l) = -\log \phi_l(\mathcal{C}_l)$. If this can be written in terms of sufficient statistics and certain parameters

$$p(X) = e^{\sum_t \Theta_t T_t - A(\Theta)}, \tag{3.56}$$

$p(X)$ belongs to an exponential family with $A(\Theta) = \log Z(\Theta)$ termed the cumulant generating function or log partition function, which is known from statistical physics as the negative free energy. The cumulant generating function, a logarithmic sum of exponentials, is convex in Θ and infinitely often differentiable on the set $\{\Theta \mid A(\Theta) < \infty\}$. The first derivative can be expressed via the expectation of the sufficient statistics as

$$\frac{\partial A(\Theta)}{\partial \Theta_t} = \mathbb{E}_{\Theta_t}\{T_t\}. \tag{3.57}$$

Assuming the nodes $v_n \in \mathcal{V}$ with $n \in \{1, 2 \dots N\}$ shown in figure 3.1 are representing discrete random variables $v_{nk} \in \{1, 2 \dots K\}$. A multinomial MRF associated with $G(\mathcal{V}, \mathcal{E})$ can be written as

$$p(V) = \prod_k \prod_{k'} \exp\Big\{ \sum_{n \in \mathcal{V}} \Theta_k \delta(v_{nk} - k) + \sum_{\{n,m\} \in \mathcal{E}} \delta(v_{nk} - k) \tilde{\Theta}_{kk'} \delta(v_{mk'} - k') - A(\Theta, \tilde{\Theta}) \Big\}. \tag{3.58}$$

3.4.2.2 Factor Graphs

Both directed and undirected graphs model global probability distributions of multiple variables as product of factors which just depend on subsets of these variables. A problem we will encounter in the context of unsupervised image segmentation is the one of finding marginal probabilities as shown in (3.8). Here, we can either directly evaluate the marginalization (summation) defined in (3.8) (see section 3.4.3) or we can employ approximations like sampling algorithms (see section 3.4.6). To perform analytical solutions like the sum product algorithm described in section 3.4.3, we consider a further variant of graphical models called factor graphs which emphasize the algorithmic queue of the sum product procedure.

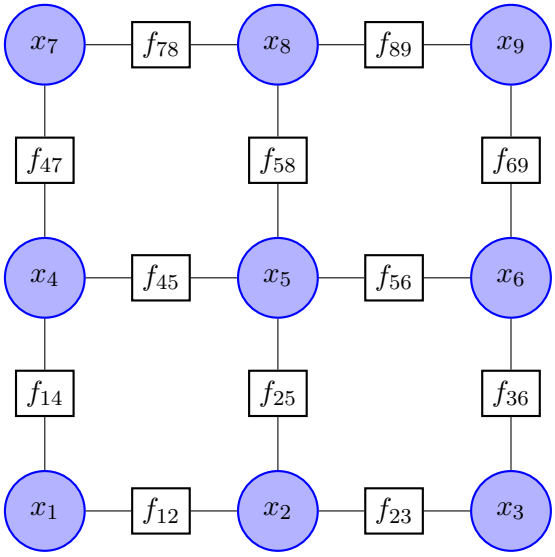


Figure 3.2: One possible factor graph representation of the MRF shown in figure 3.1.

With factor graphs, the factorization property of a graphical model is made more explicit by introducing extra nodes for each factor. A factor graph equivalent to the graph shown in figure 3.1 is presented in figure 3.2. Therewith we define a probability distribution p over the random variables X as product of factors

$$p(X) = \prod_{l=1}^{L} f_l(\tilde{X}_l), \tag{3.59}$$

with $\tilde{X}_l \subset X$. In general the various representations of graphical models (directed model, undirected model and factor graphs) cannot be transformed unambiguously into each other. The factors in (3.59) do not necessarily have to be defined via probability distributions. A normalization always can be established by summing over the possible states of the random variables as with MRFs. The factor graph shown in figure 3.2 has factors involving only two variables and hence can be associated with the distribution

$$p(X) = \prod_{\{n,m\}\epsilon\mathcal{E}} f_{n,m}(x_n, x_m). \tag{3.60}$$

3.4.3 Exact Inference on Graphs

Exact inference on graphs is concerned with e.g. the calculation of local marginal probabilities, i.e., given a probability distribution over a set of random variables $p(x_1, \ldots, x_n, \ldots x_N)$, calculate $p(x_n)$ via the summation rule (3.8)

$$\sum_{X \sim x_n} p(x_1, \ldots, x_n, \ldots x_N) = p(x_n). \tag{3.61}$$

Here the notation $X \sim x_n$ indicates that summation is performed over all elements of X but x_n. Moreover we assume that the random variables X are of discrete nature having an alphabet size equal K.

In case of large-scale graphical models, brute force evaluation of (3.61) is infeasible. The main idea behind the method discussed below is to use the distributive law to reduce the computational load.

3.4.4 Message Passing on Tree-Structured Factor Graphs

The tree structure of a graph is important since any node is a separator that decomposes the graph into disjoint subsets. A general formulation of message passing on tree-structured factor graphs (see figures 3.3 and 3.4), is called the sum-product algorithm. Consider the basic problem formulation (3.61) written for factor graphs

$$p(x_n) = \sum_{X \sim x_n} \prod_{l=1}^{L} f_l(\tilde{X}_l). \tag{3.62}$$

Given a multiple tree-structured factor graph and isolating out the node x_n, we are left with multiple separated graphs which are again trees and whose root is a factor node. Thus, we may rewrite (3.62) as

$$p(x_n) = \sum_{X \sim x_n} \prod_{l \in \mathcal{N}(x_n)}^{L} f'_l(x_n, \tilde{X}'_l), \tag{3.63}$$

where the component trees corresponding to all neighbouring factor nodes are captured by the factors $f'_l(x_n, \tilde{X}'_l)$ that depends on x_n and all other variables $\tilde{X}'_l$ contained in that component (in figure 3.3, the factor for the subtree connected via factor node f_4 is shown as $f'_4(x_n, \tilde{X}'_4)$). Applying the distributive

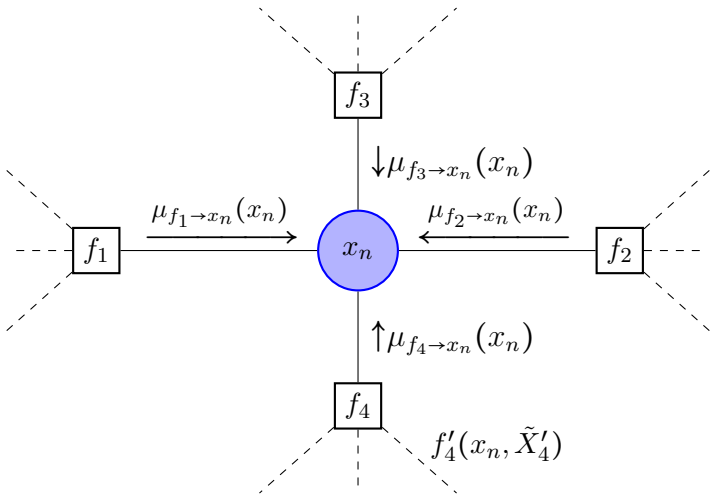


Figure 3.3: Part of a tree-structured factor graph centered about a variable node x_n. Each neighbouring factor node of x_n, i.e. $\mathcal{N}(x_n) = \{f_1, f_2, f_3, f_4\}$, is spanning a new subtree (depicted via dashed lines) having a factor representation $f'_l(x_n, \tilde{X}'_l)$ as shown in (3.63). The marginal probability of x_n can be calculated by multiplying all incoming messages from its neighbouring nodes according to (3.65), i.e., $p(x_n) = \mu_{f_{l_1} \to x_n}(x_n)\mu_{f_{l_2} \to x_n}(x_n)\mu_{f_{l_3} \to x_n}(x_n)\mu_{f_{l_4} \to x_n}(x_n)$.

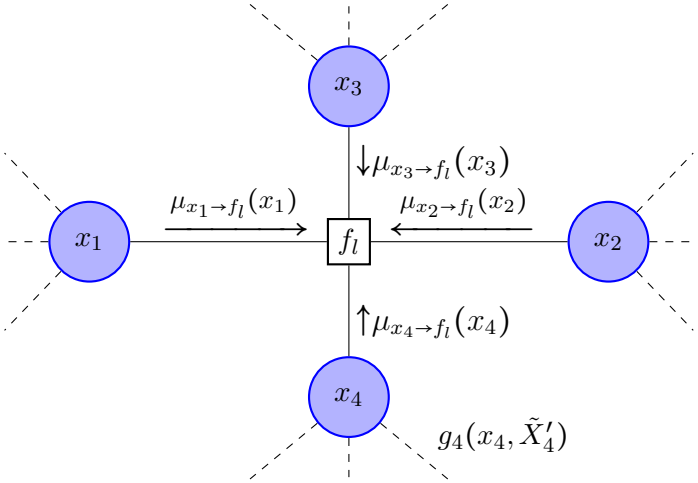


Figure 3.4: Part of a tree-structured factor graph centered about a factor node f_l. Each neighbouring variable node of f_n, i.e., $\mathcal{N}(f_l) = \{x_1, x_2, x_3, x_4\}$, is spanning a new subtree (depicted via dashed lines) having a factor representation $g_i'(x_i, \tilde{X}_i')$ as shown in (3.66).

law to (3.63), summation and multiplication can be exchanged,

$$p(x_n) = \prod_{l\in\mathcal{N}(x_n)}^{L} \underbrace{\sum_{\tilde{X}_l'} f_l'(x_n, \tilde{X}_l')}_{\mu_{f_l\to x_n}(x_n)}. \tag{3.64}$$

Thus $\sum_{\tilde{X}_l'} f_l'(x_n, \tilde{X}_l')$ can be viewed as message from the factor node f_l to the variable node x_n. With this, the marginal probability of x_n becomes a product of incoming messages from all its neighbouring factor nodes

$$p(x_n) = \prod_{l\in\mathcal{N}(x_n)} \mu_{f_l\to x_n}(x_n). \tag{3.65}$$

To further exploit the tree structure of the graph we observe that each subtree can again be decomposed into smaller subtrees, one for each variable node neighbouring f_l (except for x_n). This amounts to the refined factorization

$$f_l'(x_n, \tilde{X}_l') = f_l(\tilde{X}_l) \prod_{x_i\in\mathcal{N}(f_l)\setminus x_n} g_i(x_i, \tilde{X}_i') \tag{3.66}$$

Therefore the message from f_l to x_n can be computed as

$$\begin{aligned}\mu_{f_l\to x_n}(x_n) &= \sum_{\tilde{X}_l'} f_l(\tilde{X}_l) \prod_{x_i\in\mathcal{N}(f_l)\setminus x_n} g_i(x_i, \tilde{X}_i') \\ &= \sum_{\mathcal{N}(f_l)\setminus x_n} f_l(\tilde{X}_l) \prod_{x_i\in\mathcal{N}(f_l)\setminus x_n} \underbrace{\sum_{\tilde{X}_i'} g_i(x_i, \tilde{X}_i')}_{\mu_{x_i\to f_l(x_i)}}.\end{aligned} \tag{3.67}$$

The final observation is that calculating $\mu_{x_i \to f_l} = \sum_{\tilde{X}_i'} g_i(x_i, \tilde{X}_i')$ is just an other instance of the marginalization problem we started with. Hence we can apply the above message passing ideas recursively according to

$$\mu_{x_i \to f_l}(x_i) = \prod_{j \in \mathcal{N}(x_i) \setminus f_j} \mu_{f_j \to x_i(x_i)} \tag{3.68}$$

until we have reached the leave nodes of the factor graph.

3.4.4.1 Sum-Product Algorithm

To obtain a single marginal, we start at the leave nodes and apply the message passing operations recursively until we reach the desired node. If a leave node is a variable node as shown in figure 3.5 (a), the associated g-factor (3.66) and therewith the message passed to its only neighbouring factor node is equal one.
If a leave node is a factor node, then the associated f-factor depends only on the corresponding single neighbouring variable node and hence the outgoing message is just the factor itself (see figure 3.5) (b).

Following the procedure defined in section 3.4.4 we further propagate messages until all messages have been sent along all edges in the graph. A node can send a message as soon as it has received incoming messages on all other links. The main difference between the two types of messages is, that messages sent from variable nodes are built up by multiplying messages delivered by the neighbouring factors (3.68). Messages sent from factor nodes are marginalizing over all variables which are delivering messages to the considered factor node (3.67). This fact is visible in that $\mu_{x_n \to f_m}(x_n)$ is still a function of the variable which is yielding the message. On the other hand, $\mu_{f_m \to x_i}(x_i)$ is already governed by the variable to which it is aimed to be sent but not on variables involved in past message passing steps.

The main observation is that there is nothing special about any single node. Hence the messages that are sent between the nodes are actually independent of the variable we want to marginalize.

(a) (b)

Figure 3.5: Initializations of a message passing procedure in tree-structured factor graph: (a) if a leaf node is a variable node, the message to its neighbouring factor node is just given by 1; (b) if a leaf node is a factor node, the message sent to its neighbouring variable node is the factor as function of the corresponding neighbouring variable.

Therefore it is easy to extend the procedure to obtain all marginals at once without repeating the algorithm for each and every variable.

As mentioned above, with tree-structured graphs every node divides the tree into disconnected subgraphs. Hence the message passing procedure can be solved in an exact manner. Unfortunately many graphs of real world applications do not exhibit a tree structure. Nevertheless loopy versions of the sum-product algorithm are a popular and powerful technique.

3.4.5 Empirical Mean - Maximum Entropy Principle

The empirical expectation of a set of sufficient statistics T_t for a scalar random variable with J samples is defined according to (3.5) as

$$\hat{\mu}_t = \frac{1}{J} \sum_{j=1}^{J} T_t(x_j). \tag{3.69}$$

Based on this vector of empirical expectations, the goal is to infer a probability distribution for the random variable X. A distribution is said to be consistent with the data if

$$\hat{\mu}_t = \mathbb{E}_p\{T_t(X)\} = \int_{\mathcal{X}} T_t(x)p(x)dx \qquad \forall t. \tag{3.70}$$

In other words, the expectation $\mathbb{E}_p\{T_t(X)\}$ under the distribution p is matched to the expectation under the empirical distribution. Because there are many distributions which are consistent with the data, a mechanism is searched to choose among them. With the maximum entropy principle, the distribution p^* with maximum entropy is searched among all distributions which are consistent with the data. This principle is formulated as optimization problem as follows:

$$p^* = \arg\max_{p \in \mathcal{P}} \mathrm{H}(p) \qquad \text{subject to} \qquad \mathbb{E}_p\{T_t(X)\} = \hat{\mu}_t, \tag{3.71}$$

with $\mathcal{P}$ be the set of all probability distributions.

3.4.6 Monte Carlo Methods

A problem arising due to the usage of MRFs including dependencies among random variables is, that the calculation of probabilities for single voxels (marginals as well as conditionals) are no longer analytical feasible. Therefore we have to resort to approximate inference methods. A simple but powerful method to derive expectations under a specific probability distribution is to draw numeric samples, also known as Monte Carlo methods [1, 7, 28, 32].

Having obtained a set of samples $X^{(r)}$ from a probability distribution $p(X)$, the goal is approximate the expectation of some function $f(X)$ by a finite sum as

$$\mathbb{E}\{f(X)\} = \int f(X)p(X)dX \approx \frac{1}{R} \sum_{r}^{R} f(X^{(r)}). \tag{3.72}$$

Basic sampling algorithms are importance sampling and rejection sampling, which does not produce samples from the distribution $p(X)$ but rather uses a proposal distribution $q(X)$. This methods in general suffer from limitations dealing with high dimensional data sets.

A more powerful framework to sample from a large class of probability distributions is given by Markov Chain Monte Carlo (MCMC). This method has its origin in statistical physics where it was desired to generate states of a physical system according to the Boltzmann distribution (see also (3.55))

$$p_B(X) \propto \exp\Big\{ - \frac{E}{kT} \Big\}, \tag{3.73}$$

with E describing the energy of the system, k being the Boltzmann constant and T the thermodynamic temperature. Metropolis solution to this problem works as follows. Change the state of the system randomly in some predefined search space dX. Calculate the energy of the old state X^{old} as well as the energy of the new state X^{new}. The new state is then accepted with probability

$$p_A = \min\left(1, \frac{p_B(X^{new})}{p_B(X^{old})}\right). \tag{3.74}$$

Therefore a uniformly distributed random number u is drawn and the new state is accepted if $p_A \geq u$. Hence if the new state of the physical system has higher probability p_B than the old state, the new state is certainly accepted.

A more generalized procedure was proposed by Hasting. The so called Metropolis-Hasting algorithm is again assuming a proposal distribution. But with Metropolis-Hasting algorithm the proposal distribution $q(X \mid X^{old})$ is dependent on the current state of the system configuration and so the sequence forms a Markov chain. Therewith one can sample from arbitrary probability distributions p according to

$$p_A = \min\left(1, \frac{p(X^{new})q(X^{old} \mid X^{new})}{p(X^{old})q(X^{new} \mid X^{old})}\right). \tag{3.75}$$

It is emphasized that in case of symmetric proposal distributions q the Metropolis-Hasting algorithm (3.75) reduces to the basic Metropolis algorithm (3.74).

A widely applicable Markov chain Monte Carlo algorithm is Gibbs sampling. It can be viewed as Metropolis method with proposal distributions equals the conditional distributions $p(x_n \mid x_1, ..., x_{n-1}, x_{n+1}, ..., x_N)$. In combination with graphical models this procedure is of practical interest. Due to statistical independent statements, which come along with the conditioning of variables (voxels), the conditional distributions involve only a small subset of variables.

4

Image Clustering in PET

TO make things more precise, the problem formulation have to be focused on image labeling for PET. The data under consideration as mentioned in chapter 2, are volumetric images with finite resolution emerging from radioactive decays. Each spatial discrete entity (voxel) is a positive real-valued deterministic variable ([Bq/ml]) which gets random due to measurement and reconstruction processes. Examples of PET images (slices) can be found in figure 2.1 and figure 5.2 (b), showing an instance of a human respectively a phantom PET image.

For mathematical formulations of the problem, an observed PET image comprising N voxels is therefore assumed to be a real and positive realization X of a random vector

$$X = (x_1, ..., x_N)^T. \tag{4.1}$$

Note that PET images are three-dimensional matrices and therefore having three indices which for convenience are collectively denoted by n.

In order to label each voxel according to its membership to K different clusters of the image, an unobserved K-dimensional binary (label-) vector z_{nk} $(k = 1, ..., K)$ for each voxel is introduced. If a voxel is assumed originating from the kth cluster, 1 is assigned to the kth component of z_{nk} whereas the remaining components are set to zero. This representation is called a 1-of-K scheme as used for Generalized Bernoulli random variables, see section A.2. With this assumption, an unobserved multinomial label matrix for the entire image is written as

$$\mathbf{Z} = \begin{pmatrix} z_{11} & \cdots & z_{1K} \\ \vdots & \ddots & \vdots \\ z_{N1} & \cdots & z_{NK} \end{pmatrix}. \tag{4.2}$$

To formulate a clustering problem, i.e., to label each voxel x_n of a PET image according to healthy tissue or cancerous tissue (i.e., estimate $\mathbf{Z}$), we first seek to build statistical models describing relations (e.g., physical behaviour) between the PET data X and the labeling matrix $\mathbf{Z}$. Building statistical

models means to determine some probability measure incorporating the known data X as well as the unknown label matrix $\mathbf{Z}$. A desirable measure for estimating a labeling is e.g. the posterior distribution of $\mathbf{Z}$ given the data X, $p(\mathbf{Z} \mid X, \Theta)$, which is defined in section 3.3.1. Thus, the parameter vector Θ governing the posterior distribution is defined by

$$\Theta = \begin{pmatrix} \Theta_X \\ \Theta_{\mathbf{Z}} \end{pmatrix}, \tag{4.3}$$

But often all what we assume to know is the conditional distribution of X given $\mathbf{Z}$ (3.25), see also section 3.1.2. If establishing a conditional distribution $p(X \mid \mathbf{Z}, \Theta)$, the introduction of an additional prior term for the label matrix $p(\mathbf{Z} \mid \Theta_{\mathbf{Z}})$ enables the calculation of the posterior distribution by applying Bayesian theorem in the form (3.11). In case of no prior knowledge about $\mathbf{Z}$, the prior distribution can be considered as a uniform distribution $p(\mathbf{Z} \mid \Theta_{\mathbf{Z}}) \propto \mathcal{U}$ (see section 3.3.1).

For completely naive models, disregarding prior information, a labeling $\mathbf{Z}$ have to be estimated directly from the conditional distribution $p(X \mid \mathbf{Z}, \Theta)$. As missing prior information is considered to be described by a flat probability distribution (i.e., a uniform distribution), again the application of Bayesian theorem (3.11) is feasible. Hence the optimal labeling is simply calculated by evaluating and normalizing the conditional probabilities for each voxel label and assign these values $\mathbf{Z}$. Due to approximating the required quantity (the posterior distribution $p(\mathbf{Z} \mid X, \Theta)$) via the conditional distribution $p(X \mid \mathbf{Z}, \Theta)$, this strategy actually corresponds to a ML estimation problem (3.40).

Incorporating a prior probability distribution moreover offers a second opportunity of formulating statistical models using the product rule (3.9). Hence also the joint probability distribution (section 3.1.2) of the labeling matrix and the data vector $p(X, \mathbf{Z} \mid \Theta)$ serves for the optimization of $\mathbf{Z}$. Models where $\mathbf{Z}$ governs some mixing coefficients (e.g., GMM from section A.4) as used for generalized Bernouli distributions (A.6) are called mixture models. The set of variables $\{X, \mathbf{Z}\}$ is termed the complete data whereas solely the known image data vector X is called the incomplete data according to section 3.3.2.1

As presented by [44], those two formulations are called generative models and discriminative models. Generative models attempt to model a joint distribution $p(X, \mathbf{Z})$ over the known data X and the unknown data $\mathbf{Z}$ (Directed models are often used as generative models), which factorizes as $p(\mathbf{Z}, X) = p(\mathbf{Z})p(X \mid \mathbf{Z})$, see also (4.75). Figure 4.1 (a) depicts a generative model as a directed graph where the link point from the label node z_k to the data node x describing the model $p(\mathbf{Z})p(X \mid \mathbf{Z})$. Moreover figure 4.1 (b) shows a generative model as a MRF, having an undirected link and describing the model $p(\mathbf{Z}, X)$. There again with discriminative models the conditional probability distribution $p(\mathbf{Z} \mid X)$ is modeled directly, which is all that we need for classification. Generative models are models that describe how a label vector $\mathbf{Z}$ can probabilistically generate a PET image vector X. Discriminative models work in the reverse direction, describing directly how to take an image vector X and assign it a label $\mathbf{Z}$. In figure 4.1 (c), a discriminative model is presented as directed graph where the link is pointing from the data node x to the label node z_k.

The main conceptual difference between discriminative and generative models is that a conditional distribution does not include a model of $p(X)$, which is not needed for classification anyway. Discrim-

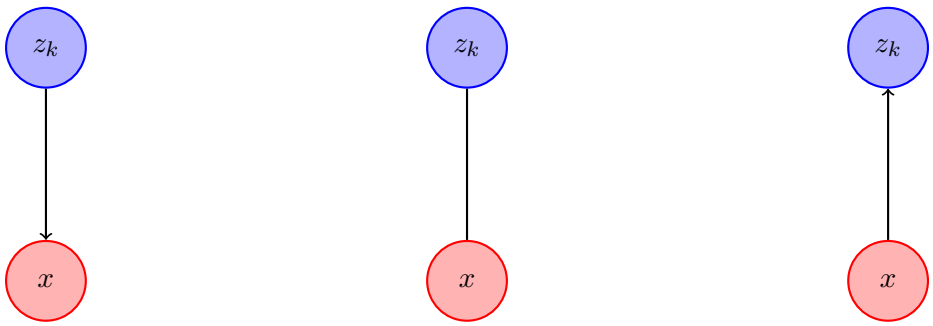


Figure 4.1: Graphical representations of generative and discriminative models, i.e. $p(x, z_k)$ respectively $p(z_k \mid x)$, for a voxel x and its label vector z_k. (a) directed graph representation of a generative model $p(x \mid z_k)p(z_k)$, (b) MRF representation of a generative model $p(x, z_k)$ and (c) directed graph representation of a discriminative model $p(z_k \mid x)$.

inative models make conditional independent assumptions among $\mathbf{Z}$ (more complex graphical models will have local neighbourhood interactions defined among the labels of $\mathbf{Z}$) and assumptions about how the $\mathbf{Z}$ can depend on X, but do not make conditional independence assumption among X. The difficulty in modeling $p(X)$ is that it often contains highly dependent features that are difficult to model. If we construct a graph for the conditional distribution $p(\mathbf{Z} \mid X)$, any factor that depend only on X vanish from the graphical structure for the conditional distribution. They are irrelevant to the conditional because they are constant with respect to $\mathbf{Z}$.

Having defined a statistical model by defining appropriate probability measures, the aim is to estimate the optimal labeling $\mathbf{Z}$ given the measured image data X. However due to missing prior information about Θ as demanded in chapter 1, the best approximating probability distribution is not necessarily represented via some first choice (guess) of parameter settings. Therefore, the parameters also have to be optimized to fit the probability distributions to the various clusters given by $\mathbf{Z}$. For this purpose the ML estimator (3.40) can be calculated from the conditional distribution $p(X \mid \mathbf{Z}, \Theta)$ which, if considered as a function of Θ, is called the likelihood function. If, despite missing prior information, a prior probability $p(\Theta)$ for the parameters is introduced[1], the MMSE estimator (3.29) can be formulated via Bayesian theorem using the conditional probability $p(\Theta \mid X, \mathbf{Z}) \propto p(X \mid \mathbf{Z}, \Theta)p(\Theta)$. If more complex graphical models are under consideration incorporating neighbourhood relations among various labels, the parameter optimization problem is no longer analytical feasible. Working with MRFs using Boltzmann distributions as shown in section 4.4.1, the parameter optimization problem can be viewed as a convex optimization problem. Such problems on its own need to be solved via iterative techniques.

Combining the labeling step with the parameter estimation step, iterative procedures are employed alternating them. Considering the EM algorithm in section 3.3.2.1, a procedure is presented including

[1] A prior probability for the parameters Θ is used in later chapters to influence wrong parameter estimates.

both steps in one mathematical framework. Each step can be chosen as initial step to start with depending on the information we start with, initial parameter estimates θ^{init} or initial label estimates $\mathbf{Z}^{\text{init}}$.

As mentioned in chapter 1 a predefined atlas mapping extracts the activity distribution for single organs. The clinical routine for obtaining VOIs to process on with segmentation algorithms is drawing them manually onto the considered PET image. As further stated in chapter 1, the activity distributions in healthy organs are approximated as noisy constant signals. In case of tumor appearance, where the tumor shows increased tracer uptake in contrast to the surrounding healthy tissue, a coarse description of the underlying problem is to assume two constant signals in Gaussian noise. For this the amount of clusters is two, rendering (4.2) a $N \times 2$ matrix with $K = 2$.

4.1 ML Labeling for a Gaussian Model

4.1.1 Naive MLGM

As mentioned in chapter 1 and during the introduction of this chapter, the data we obtain from a predefined atlas mapping or manual delineation are parts of human organs. The activity concentration in healthy tissue is considered to be constant with additive noise. It was further emphasized that cancerous tissue shows an increased FDG uptake against the surrounding healthy tissue on PET images, see figure 2.1. Hence we propose to approximate both voxel clusters, i.e., voxels being members of healthy tissue and voxels being members of cancerous tissue, to be represented by constant activity levels with additive noise. Although the voxels x_n of an observed image X (resulting from radioactive decays) are counts of events and therefore never negative (see chapter 2), we assume that they are Gaussian distributed (A.12) around their cluster mean. Therefore the voxels of both clusters get distinguished by their means (A.13) and standard deviations (A.14), $\theta_{\text{tum}} = (\mu_{\text{tum}}, \sigma^2_{\text{tum}})$ and $\theta_{\text{hea}} = (\mu_{\text{hea}}, \sigma^2_{\text{hea}})$, which are indexed as θ_k.

Hence as naive basic model, omitting neighbourhood interactions and prior informations of $\mathbf{Z}$, we further simply propose a conditional probability distribution for the data vector X conditioned on a certain labeling $\mathbf{Z}$ to be written as product of all individual (independent) Gaussian voxel probabilities (A.18)

$$p(X \mid \mathbf{Z}, \Theta) = \prod_{k=1}^{K} \prod_{n=1}^{N} p(x_n \mid z_{nk}, \theta_k) = \prod_{k=1}^{K} \prod_{n=1}^{N} \mathcal{N}(x_n \mid \mu_k, \sigma_k^2)^{z_{nk}} \tag{4.4}$$

$$= \prod_{k=1}^{K} \prod_{n=1}^{N} \left[\frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\Big(-\frac{1}{2\sigma_k^2}(x_n - \mu_k)^2 \Big) \right]^{z_{nk}} . \tag{4.5}$$

This representation follows directly from the derivation of a GMM A.30.

In figure 4.2, a two-dimensional pendant of our three-dimensional image clustering problem modeled via (4.4) and (4.5) is depicted. The only interactions incorporated in (4.4) and (4.5) are the conditional dependencies of the observed data from its labels which is shown in figure 4.2 by drawing directed edges pointing from each voxels label to its according observed data point. So this model can be viewed as a generative model in form of a directed graph, with a uniform prior distribution for the label matrix $\mathbf{Z}$.

The parameters μ_k and σ_k are considered to be constant for each cluster (during a labeling step). Moreover, as cancerous and healthy tissue get searched, $K = 2$. With this probability measure for the entire data vector X, each voxel x_n of an observed sample is contributing according to its labeling z_{nk}.

To formulate an iterative classification algorithm with alternating label estimation step and parameter estimation steps, the distribution (4.5) have to serve as objective function for the optimization of Θ and $\mathbf{Z}$. Note that (4.5) do not incorporate prior probabilities. It is solely a product of conditional Gaussian distributions. As mentioned in the introduction of chapter 4, the usage of Bayesian theorem (3.10) relates the conditional distribution with the posterior distribution $p(\mathbf{Z} \mid X, \Theta)$. Assuming a

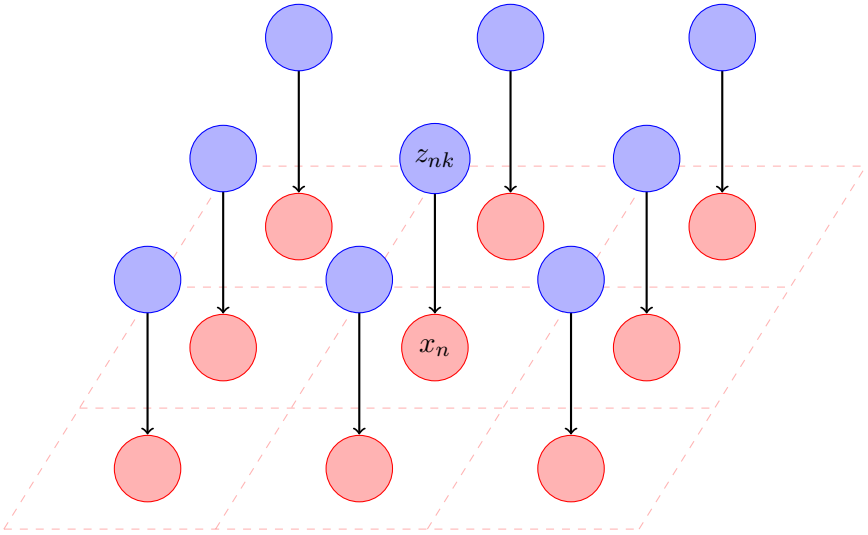


Figure 4.2: two-dimensional Bayesian network representation of the naive MLGM model for an image clustering problem shown in (4.4). The only interactions defined by (4.4) are the one for each voxels label and its according observed data point.

uniform distribution $p(\mathbf{Z} \mid \Theta_{\mathbf{Z}})$ (case of no prior knowledge), the constant prior appears as a factor in (3.11) which can be pulled out leaving the conditional distribution the only quantity involved in determining the posterior distribution (3.11). As $p(\mathbf{Z} \mid X, \Theta)$ is the actually desired optimization function, their interchanging with $p(X \mid \mathbf{Z}, \Theta)$ is hence reasonable. Therefore we apply ML estimation, as given by (3.40) in classical parameter estimation, to estimate the parameter as well as the label matrix

$$\Theta_{\mathrm{ML}}(X) = \arg\max_{\Theta}\{\ln p(X \mid \mathbf{Z}; \Theta)\} \tag{4.6}$$

$$\mathbf{Z}_{\mathrm{ML}}(X) = \arg\max_{\mathbf{Z}}\{p(X \mid \mathbf{Z}; \Theta)\}. \tag{4.7}$$

Alternating these optimization problems by respectively using the updates from the preceding step results in the following iterative scheme (which we will denote maximum likelihood estimation for a Gaussian Model (MLGM)):

- Label estimation: With some initial/previous estimate of the parameter $\Theta^{(i)}$, a labeling is calculated as the solution of (4.7) using (4.5). This is performed by directly evaluating the conditional probability $p(X \mid \mathbf{Z}; \Theta^{(i)})$ (4.5). As the random variables of X and $\mathbf{Z}$ are independent among themselves, the optimization problem (4.7) decomposes into subproblems for each voxel according to

$$z_{nk,\mathrm{ML}}^{(i+1)} = \arg\max_{z_{nk}}\{p(x_n \mid z_{nk}, \Theta^{(i)})\}. \tag{4.8}$$

  Hence the actually desired posterior probabilities of a voxel labeling z_{nk} get approximated according to Bayesian theorem (3.11) using flat priors as

$$p(z_{nk} = 1 \mid x_n) \propto \mathcal{N}(x_n \mid \mu_k^{(i)}, \sigma_k^{2(i)}). \tag{4.9}$$

Comparing the various membership probabilities against each other and go for the largest, a binary labeling is assigned according to

$$z_{nk,\mathrm{ML}}^{(i+1)} = \begin{cases} 1 & p(z_{nk} = 1 \mid x_n) \geq p(z_{nl} = 1 \mid x_n) \quad \forall k \neq l \\ 0 & \text{else} \end{cases} \tag{4.10}$$

- Parameter estimation: Using some initial/previous estimate of the label matrix $\mathbf{Z}^{(i)}$, the Gaussian mean and standard deviation are calculated as the solutions of problem (4.6). This is accomplished by derivate the logarithm of (4.5) regarding μ_k and σ_k and setting the outcome equal zero (see ML estimation (3.41)). Solving these equations shows that the cluster mean and cluster standard deviation get updated as

$$\mu_{k,\mathrm{ML}}^{(i+1)} = \frac{\sum_{n=1}^{N} x_n z_{nk}^{(i)}}{\sum_{n=1}^{N} z_{nk}^{(i)}} \tag{4.11}$$

$$\sigma_{k,\mathrm{ML}}^{(i+1)} = \sqrt{\frac{\sum_{n=1}^{N} (x_n - \mu_k^{(i+1)})^2 z_{nk}^{(i)}}{\sum_{n=1}^{N} z_{nk}^{(i)}}}. \tag{4.12}$$

  Hence (4.11) and (4.12) get the empirical cluster mean and the empirical standard deviation as these entities are calculated from the observed data X.

Those two steps get alternated till the difference of two succeeding parameter estimates deceeds a lower bound ,i.e., $|\Theta^{(i)} - \Theta^{(i+1)}|^2 < \epsilon$.

In figure 4.3 a further Bayesian network representation of the underlying problem is depicted making the dependencies of the data X from the model parameters μ and σ more obvious. Because dependencies among various label vectors z_{nk} or data entries x_n are not taken into concern, figure 4.3 restrict the graphical representation to the case of just one voxel. As the parameters μ and σ are

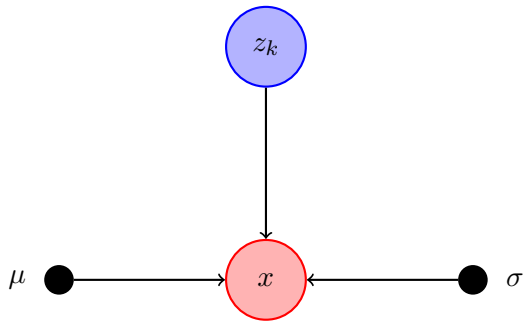


Figure 4.3: Bayesian network representation for only one label-data-pair (x, z) showing also the parameters μ and σ governing the probability distribution (4.4) from which the probability of x is derived. The drawing style for the parameters, little black nodes with labels located outside the node, account for the fact that they are considered as deterministic variables.

considered as deterministic, they are not drawn using the node style for random variables. Instead they are depicted via little black nodes having their label placed outside the node.

Note that the MLGM approach estimates a discrete labeling of the image (4.10) calculated using the conditional distribution (4.9). Hence the labels used to calculate the empirical statistics as shown in (4.11) and (4.12) are of binary nature. But as mentioned in chapter 1 we are interested in resolving problems emerging due to PVE and hence we want to produce continuous labels.

A first step towards a continuous treatment of voxels is to use a final decision step. This calculates the posterior probability of the voxels being member of each cluster, applying Bayesian theorem (3.11) with flat prior probabilities for $\mathbf{Z}$ as

$$z_{nk,\mathrm{ML}}^{\mathrm{final}} = p(z_{nk} = 1 \mid x_n) \propto \frac{p(x_n \mid z_{nk}, \theta_{k,\mathrm{ML}}^{\mathrm{final}})}{\sum_{k=1}^{K} p(x_n \mid z_{nk}, \theta_{k,\mathrm{ML}}^{\mathrm{final}})}. \tag{4.13}$$

In this case the values of $z_{nk,\mathrm{ML}}^{\mathrm{final}}$ for each voxel are decimals in the interval $[1, 0]$ enabling to account for partial volume voxels. The denominator in (4.13) ensures that the probabilities for each voxel sums to one.

Due to the assumption of independent random variables (elements of X and $\mathbf{Z}$) dependencies between neighbouring voxels are not considered. For this we need a model incorporating correlations among voxels having the same label, which is introduced in the next subsection.

4.1.2 MLGM with Correlations

In section 4.1.1, the elements of the observed data vector X were assumed to be independent of each other. To incorporate local correlations of the observed data and hence to account for PVE, a more advanced form of the MLGM approach is proposed. Therefore the data X is considered as a jointly Gaussian random vector (A.15) with a covariance matrix (A.17) relating each voxel of the data vector X. Using a precision matrix rather than a covariance matrix as mentioned in section A.3, the general form of a probability distribution for jointly Gaussian (dependent) random variables X conditioned on $\mathbf{Z}$ is established by

$$p(X \mid \mathbf{Z}, \Theta) \propto \prod_{k=1}^{K} \prod_{n=1}^{N} \prod_{m=1}^{M} \exp\Big(-\frac{1}{2} \Lambda_{nmk} (x_n - \mu_{nk})(x_m - \mu_{mk}) \Big)^{z_{nk} z_{mk}}. \tag{4.14}$$

In (4.14) the $n \times m$ precision matrix Λ is also known as correlation matrix (4.15), i.e., the inverse of the covariance matrix (A.17). The parameter $\vec{\mu}$ denotes a N-dimensional vector of means (A.16).

Because we are not performed with multiple PET images of the same situation (even during time scans just a few PET images are acquired), we posses no sufficient statistics of each single voxel x_n. Therefore the mean vector is restricted to have just differing values for differing clusters. Hence it actually can be considered as a scalar value for each cluster, μ_k, as it was done in section 4.1.1.

Using the same argument as above some further simplifications regarding the structure of the precision matrix have been made. The diagonal elements of Λ in (4.14) are requested to be equal for each cluster, λ, and correspond to the inverse of the squared standard deviations as defined in (A.13).

Moreover the off diagonal elements ν_k, corresponding to correlations among neighbouring voxels, are also considered to be equal for each cluster. Thus the precision matrix can be decomposed according to

$$\Lambda = \begin{pmatrix} \lambda & \cdots & \nu \\ \vdots & \ddots & \vdots \\ \nu & \cdots & \lambda \end{pmatrix} = \lambda \mathbb{1} + \nu \otimes \underbrace{\begin{pmatrix} 0 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 0 \end{pmatrix}}_{\mathbf{S}}. \tag{4.15}$$

The decomposition (4.15) splits the mean power and the correlations. This enables us to write (4.14) via summations over single voxels plus summations over pairs of voxels. Hence inserting (4.15) in (4.14) and using the definitions of vertex sets $\mathcal{V}$ and edge sets $\mathcal{E}$ (see section 3.4.1), (4.14) is simplified for multiple clusters as

$$\begin{aligned} p(X \mid \mathbf{Z}, \Theta) \quad \propto \quad & \prod_{k=1}^{K} \exp\Big(-\frac{\lambda_k}{2} \sum_{n \in \mathcal{V}} (x_n - \mu_k)^2 \Big)^{z_{nk}} \\ & \exp\Big(-\frac{\nu_k}{2} \sum_{\{n,m\} \in \mathcal{E}} (x_n - \mu_k)(x_m - \mu_k) \Big)^{z_{nk} z_{mk}}. \end{aligned} \tag{4.16}$$

To incorporate global correlations for each cluster, meaning that all pairs of voxels comprised in a specified cluster are given an entry in $\mathbf{S}$ (see (4.15)), $\mathbf{S}$ is zero just on the diagonal.

To introduce just local covariances, $\mathbf{S}$ is further restricted according to wether voxels in the three-dimensional PET image are neighbours or not. Actually the matrix $\mathbf{S}$, which describes the correlations among distinct voxels, is becoming a band structure. To illustrate this, we assume a one-dimensional random vector specifying 4 successive points on a chain in space $X = (x_1, x_2, x_3, x_4)^{\mathrm{T}}$, where every pair of successive points is assumed to build a neighbourhood (member of the edge set $\mathcal{E}$). The set of neighbouring pairs therefore consists of $\{(x_1, x_2), (x_2, x_3), (x_3, x_4)\}$. For such system, $\mathbf{S}$ can be written according to

$$\mathbf{S} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{4.17}$$

These circumstances are visualised in figure 4.4 for the case of a two-dimensional image with 5×5 voxels. First it is mentioned, that nodes are just connected by a link if they are direct neighbours as demanded by the local interactions constraint. Further, direct neighbouring nodes are connected only if they belong to the same cluster. Note that in figure 4.4 the graph is clustered into two partitions, the green voxels and the blue voxels. In three dimensions, as it is the case for PET data, a neighbourhood consists of 6 voxels.

Although the correlation matrix is simplified, the voxels of the PET image X are no longer independent random variables and therefore the labeling problem (4.7) does not subdivide into problems concerning just single voxels. To approximate the estimation problem (4.7), local conditional probability distributions get indented.

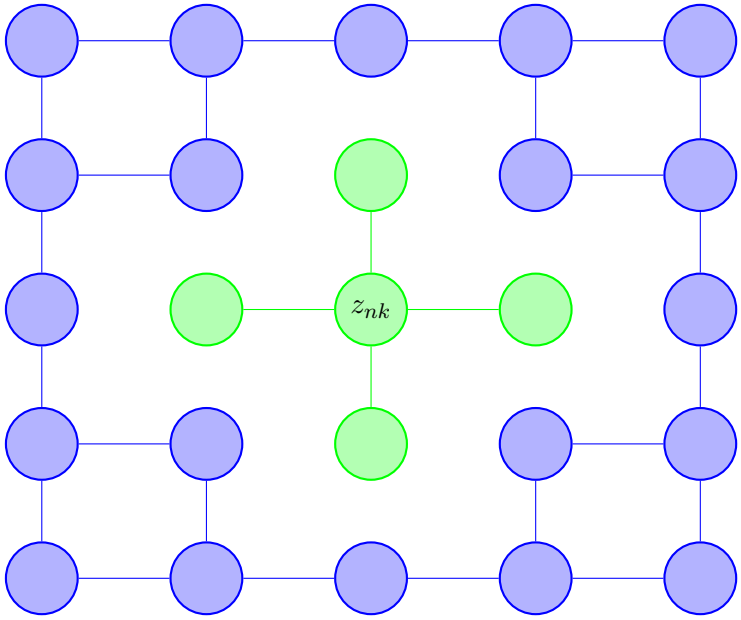


Figure 4.4: 5×5 grid voxels which are labeled according to two clusters (i.e., green and blue). Correlations between voxels are just assumed for neighbouring pairs tagged with the same label resulting in the green and blue edges indicating dependence of the respective voxels.

As with graphical models discussed in section 3.4.1, the central green voxel in figure 4.4 is independent of the blue voxels if its neighbourhood (i.e., all green voxels except the central one) is known. Hence we define a local voxel vector $X_{loc} = (x_n, \vec{x}_m^T)^T \in X$, with x_n corresponding to the central green voxel in figure 4.4 and with $\vec{x}_m$ corresponding to x_ns neighbourhood. For such local voxel vector we further define a partitioning as shown in (A.19)

$$X_{loc} = \begin{pmatrix} x_n \\ \vec{x}_m \end{pmatrix}, \qquad \vec{\mu}_{loc,k} = \begin{pmatrix} \mu_k \\ \vec{\mu}_k \end{pmatrix}, \qquad \Lambda_{loc,k} = \begin{pmatrix} \lambda_k & \tilde{\Lambda}_{mnk}^T \\ \tilde{\Lambda}_{nmk} & \tilde{\Lambda}_{mmk} \end{pmatrix}, \tag{4.18}$$

with

$$\tilde{\Lambda}_{nmk} = \begin{pmatrix} \nu_k \\ \vdots \\ \nu_k \end{pmatrix}, \qquad \tilde{\Lambda}_{mmk} = \begin{pmatrix} \lambda_k & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_k \end{pmatrix}. \tag{4.19}$$

Note that M is used to tag the amount of voxels constituting a neighbourhood. Since the data vector X is a known quantity and moreover jointly Gaussian, we are always able to compute the local conditional probability for a voxel x_n given its neighbourhood $\vec{x}_m$ as (A.20)

$$p(x_n \mid z_{nk} = 1, \vec{x}_m) \approx \mathcal{N}(x_n \mid \mu_{x_n|\vec{x}_m,k}, \Lambda_{x_n|\vec{x}_m,k}^{-1})^{z_{nk}}, \tag{4.20}$$

with conditional mean (A.25) and conditional precision (A.26) given by

$$\mu_{x_n|\vec{x}_m,k} = \mu_k + \tilde{\Lambda}_{mnk}^{\mathrm{T}} \tilde{\Lambda}_{mmk}^{-1} (\vec{x}_m - \vec{\mu}_k) \tag{4.21}$$

$$\Lambda_{x_n|\vec{x}_m,k} = \lambda_k. \tag{4.22}$$

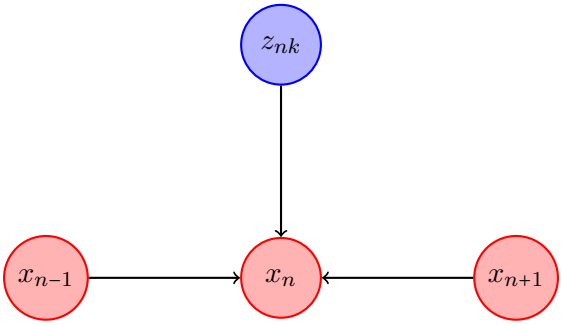


Figure 4.5: Bayesian network representation of the model presented by (4.20). Given the local neighbourhood of node x_n, x_{n-1} and x_{n+1}, and the according label z_{nk}, node x_n is statistical independent from the rest of the graph as also shown in section 3.4.2.

Thus the conditional mean (4.21) separates the cluster mean and a term governed by the neighbourhood. Furthermore, assuming such local dependencies for each voxel in X changes the scalar mean value used with MLGM into a vector valuable quantity with different mean values for each voxel x_n. Instead the precision is independent of the neighbourhood and degenerates to a scalar value. Hence we can proceed as before and solve the labeling step by directly calculating the conditional probability $p(X \mid \mathbf{Z}, \Theta)$ to approximate the desired posterior distribution for $\mathbf{Z}$, $p(\mathbf{Z} \mid X, \Theta)$.

The circumstances described above are depicted in figure 4.5 for a one-dimensional pendant of the underlying problem. The graph is represented by a generative directed model where we have omitted to draw the corresponding parameters. It is shown by the directed edges, that the local conditional probability of x_n depends only on the according label z_{nk} and the neighbourhood of x_n. So the local neighbourhood of a voxel and its label is a separator set, rendering the voxel conditional independent regarding the rest of the graph (see section 3.4.2).

Formulating again an optimization procedure via ML labeling and ML parameter estimation as in section 4.1.1 to fit the model to the data, we again employ (4.6) (4.7) to obtain an iterative update scheme as:

- Label estimation: With some initial/previous estimate of the parameter $\Theta^{(i)}$ a labeling is calculated as the solution of (4.7) having inserted (4.16). This is performed by calculating the local conditional probability of x_n given its neighbouring voxels x_m (4.20), assuming that this local voxels are labeled equally. Hence the problem (4.7) decomposes into subproblems as

$$z_{nk,\mathrm{ML}}^{(i+1)} = \arg\max_{z_{nk}} \{p(x_n \mid z_{nk}, \vec{x}_m, \Theta_k^{(i)})\}. \tag{4.23}$$

  Because all the voxels are assumed to be jointly Gaussian, the local conditional probability is also Gaussian (see section A.3) for each cluster which gets calculated due to

$$p(z_{nk} = 1 \mid x_n) \approx \mathcal{N}(x_n \mid \mu_{x_n|\vec{x}_m,k}^{(i)}, \Lambda_{x_n|\vec{x}_m,k}^{(i)})^{z_{nk}}. \tag{4.24}$$

Thereby $\mu_{x_n|\vec{x}_m}$ and $\Lambda_{x_n|\vec{x}_m}$ are the conditional mean and the conditional precision for a voxel given by (4.21) respectively by (4.22). $\mu_{x_n|\vec{x}_m}$ is the MMSE (3.29) of x_n given $\vec{x}_m$.

Note that $\Lambda_{x_n|\vec{x}_m}$ is independent of $\vec{x}_m$ and therefore a quantity that is equal for each voxel in a cluster, whereby $\mu_{x_n|\vec{x}_m}$ is an N-dimensional vector with differing values for each voxel offering more flexibility for real data. Again to obtain a labeling for the subsequent parameter estimation, we compute

$$z_{nk,\mathrm{ML}}^{(i+1)} = \begin{cases} 1 & p(z_{nk} = 1 \mid x_n) \geq p(z_{nl} = 1 \mid x_n) \quad \forall k \neq l \\ 0 & \text{else} \end{cases} \tag{4.25}$$

- Parameter estimation: Using initial/previous estimates of the labellings $\mathbf{Z}^{(i)}$, the Gaussian mean and correlation matrix per cluster are calculated as the solution of problem (4.6). To calculate the ML estimator of the parameters, we are differentiating the logarithm of the likelihood function (4.16) regarding μ_k, λ_k and ν_k and setting the solutions equal zero

$$\begin{aligned} &\frac{\partial}{\partial \Theta_k} \ln p(X \mid \mathbf{Z}^{(i)}, \Theta) = 0 = \\ &\frac{\partial}{\partial \Theta_k} \sum_{k=1}^{K} \Bigg[\sum_{n \in \mathcal{V}} z_{nk}^{(i)} \Big(\frac{1}{2} \ln \lambda_k - \frac{1}{2} \lambda_k (x_n - \mu_k)^2 \Big) + \\ &\sum_{\{n,m\} \in \mathcal{E}} z_{nk}^{(i)} z_{mk}^{(i)} \Big(\frac{1}{2} \ln \nu_k - \nu_k (x_n - \mu_k)(x_m - \mu_k) \Big) + \text{const.} \Bigg]. \end{aligned} \tag{4.26}$$

Accomplishing the derivation regarding μ_k we are faced with

$$\sum_{n \in \mathcal{V}} z_{nk}^{(i)} \lambda_k (x_n - \mu_k) + \sum_{\{n,m\} \in \mathcal{E}} z_{nk}^{(i)} z_{mk}^{(i)} \nu_k (x_n - \mu_k) + \sum_{\{n,m\} \in \mathcal{E}} z_{mk}^{(i)} z_{nk}^{(i)} \nu_k (x_m - \mu_k) = 0. \tag{4.27}$$

Because the precision matrix is a symmetric quantity (i.e., $\Lambda_{nm} = \Lambda_{mn}$) and moreover we do not assume correlations between differently labeled voxels (note that both voxels of a neighbourhood, $z_{nk}^{(i)}$ and $z_{mk}^{(i)}$, are indexed by k), the second and third summation in (4.27) can be summarized yielding

$$\sum_{n \in \mathcal{V}} z_{nk}^{(i)} \lambda_k (x_n - \mu_k) + 2 \sum_{\{n,m\} \in \mathcal{E}} z_{nk}^{(i)} z_{mk}^{(i)} \nu_k (x_n - \mu_k) = 0. \tag{4.28}$$

Splitting the summation over the edges $\{x_n, x_m\} \in \mathcal{E}$ into a summation over all voxels x_n and a summation over each neighbourhood of x_n, i.e. $x_m \in \mathcal{N}_{x_n}$, (4.28) gets

$$\sum_{n=1}^{N} z_{nk}^{(i)} \lambda_k (x_n - \mu_k) + 2 \sum_{n=1}^{N} z_{nk}^{(i)} \nu_k (x_n - \mu_k) \sum_{m=1}^{M} z_{mk}^{(i)} = 0. \tag{4.29}$$

Pulling out the common factors we are left with

$$\sum_{n=1}^{N} z_{nk}^{(i)} (x_n - \mu_k) \Big(\lambda_k + 2 \nu_k (x_n - \mu_k) \sum_{m=1}^{M} z_{mk}^{(i)} \Big) = 0, \tag{4.30}$$

leading to the same result achieved with MLGM

$$\sum_{n=1}^{N} z_{nk}^{(i)}(x_n - \mu_k) = 0. \tag{4.31}$$

Therefore the cluster means get calculated according to

$$\mu_{k,\mathrm{ML}}^{(i+1)} = \frac{\sum_{n=1}^{N} x_n z_{nk}^{(i)}}{\sum_{n=1}^{N} z_{nk}^{(i)}}. \tag{4.32}$$

Derivate (4.26) regarding λ and ν_k is strait forward and results in the following two update steps

$$\frac{1}{\lambda_{k,\mathrm{ML}}^{(i+1)}} = \frac{\sum_{n\in\mathcal{V}} z_{nk}^{(i)}(x_n - \mu_k)^2}{\sum_{n\in\mathcal{V}} z_{nk}^{(i)}} \qquad \text{and} \qquad \frac{1}{\nu_{k,\mathrm{ML}}^{(i+1)}} = \frac{\sum_{\{n,m\}\in\mathcal{E}} z_{nk}^{(i)} z_{mk}^{(i)}(x_n - \mu_k)(x_m - \mu_k)}{\sum_{\{n,m\}\in\mathcal{E}} z_{nk}^{(i)} z_{mk}^{(i)}}. \tag{4.33}$$

As with (4.11) and (4.12) we are left with empirical statistics which are calculated from the image. Note that the actual labeling is calculated using the conditional distribution (4.24). Hence the elements of the precision matrix (4.33), $\lambda_{k,\mathrm{ML}}^{(i+1)}$ and $\nu_{k,\mathrm{ML}}^{(i+1)}$, get transformed to obtain the scalar value $\Lambda_{x_n|\vec{x}_m,k}^{(i+1)}$. Moreover the mean values calculated from (4.32) get subjected by the affine transformation (4.21) yielding the N-dimensional vector $\vec{\mu}_{x_n|\vec{x}_m,k}^{(i+1)}$. The local varying mean vector offers more flexibility to fit real data and here is aimed to compensate PVE in PET images.

Those two steps get alternated till the difference of two succeeding parameter estimates deceeds a lower bound, i.e., $|\Theta^{(i)} - \Theta^{(i+1)}|^2 < \epsilon$.

Due to incorporating local correlations, this algorithm is called MLGMLC. As with MLGM, the MLGMLC approach estimates a discrete labeling of the image (4.10), and hence calculates the empirical statistics as shown in (4.32) and (4.33). Furthermore a final labeling is defined as the probability of the voxels being member of each cluster (see also section 4.1.1) as

$$z_{nk,\mathrm{ML}}^{\mathrm{final}} = p(z_{nk} = 1 \mid x_n) \approx \frac{p(x_n \mid z_{nk}, \theta_{k,\mathrm{ML}}^{\mathrm{final}})}{\sum_{k=1}^{K} p(x_n \mid z_{nk}, \theta_{k,\mathrm{ML}}^{\mathrm{final}})}, \tag{4.34}$$

enabling to account for partial volume voxels. The denominator in (4.34) ensures that the probabilities for each voxel sums to one.

4.2 EM Labeling for a Gaussian Mixture Model

4.2.1 Naive EMGMM

As shown in section 3.3.2.1, the EM algorithm can be employed to formulate a procedure that simultaneously estimate both, the parameters Θ and the unobserved label matrix $\mathbf{Z}$ resulting automatically

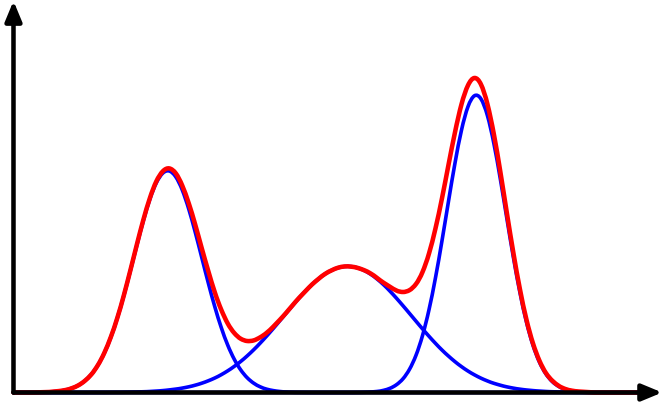


Figure 4.6: This is an example of a Gaussian mixture distribution in one dimension showing three Gaussians (each scaled by a coefficient) in blue and their sum in red. This image was taken from [1].

in an iterative update scheme. The EM algorithm in contradiction to the MLGM approach (which directly estimates the unknown labeling $\mathbf{Z}$) aims to estimate (calculate the MMSE) the joint distribution of the data and the label matrix $p(X, \mathbf{Z} \mid \Theta)$ with X assumed given (3.42).

To establish a joint probability distribution for the data and the label matrix, the product rule (3.9) can be used to combine the conditional distribution $p(X \mid \mathbf{Z}, \Theta)$ with a prior distribution $p(\mathbf{Z} \mid \Theta_{\mathbf{Z}})$ as mentioned in the introduction of this chapter. Proposing again the voxels to originate from two constant signals in Gaussian noise, a Gaussian mixture model (GMM) as formulated in appendix A.4 is exploited. A GMM introduces a generalized Bernoulli distribution (A.6) as prior for each label z_{nk} to establish a superposition of multiple Gaussian densities for each voxel. Hence the whole PET image is written as product of individual GMMs according to (A.32) as

$$p(X, \mathbf{Z} \mid \Theta) = \prod_{n=1}^{N} \prod_{k=1}^{K} \left[\tau_k \mathcal{N}(x_n \mid \mu_k, \sigma_k^2)\right]^{z_{nk}}. \tag{4.35}$$

Comparing (4.35) with the conditional distribution for the MLGM approach (4.14), the only difference is given due to the usage of the weighting factors τ_k in (4.35) which has to sum to one $\sum_k \tau_k = 1$. τ_k shows up, following the derivation in section A.4, via incorporating the generalized Bernoulli distribution for the labels in $\mathbf{Z}$ as given by (A.6). Multiplying the generalized Bernoulli prior with the conditional distribution in (4.5) and hence applying the product rule (3.9), yields (4.35). Such weighted sum of normal distributions is shown in figure 4.6, where adding the three (blue) Gaussian bell shaped curves results in the superposition depicted in red.

The generative directed graph structure of the GMM is shown in figure (4.7). Again to make the influence from the deterministic parameters μ, σ and τ more obvious, they are included by drawing little black nodes with their labels being located outside the node. This Bayesian network differs from the one shown in figure (4.3) for MLGM in adding a new parameter for the prior of z, τ.

The EM derivation of the GMM is well known [1] and leads to analytical solutions. Following the

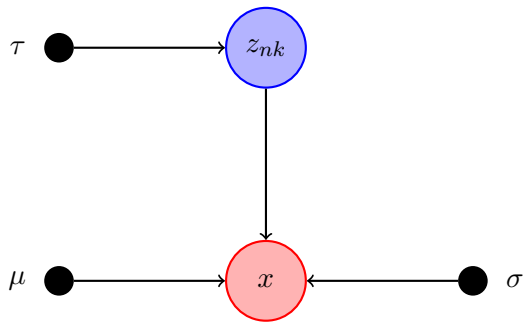


Figure 4.7: Bayesian network representation for only one label-data-pair (x, z) showing also the parameters μ, σ and τ governing the probability distribution (4.35) from which the probability of x is derived. The drawing style for the parameters, little black nodes with labels located outside the node, account for the fact that they are considered as deterministic variables.

calculus in appendix C.1, the parameter estimation step (e-step) and the labeling estimation step (m-step) can be iterated as follows:

- Label estimation (e-step): As with the MLGM approach, the data vector X is considered to comprise independent random variables (no voxel interactions). Hence the estimation problem for the labeling matrix $\mathbf{Z}$ breaks up into subproblems for each label z_{nk}. Employing some estimate of the parameter $\Theta^{(i)}$, the posterior expectation of a labeling is calculated using Bayesian theorem given by (3.11). With $\mathcal{N}_k(x_n; \mu_k^{(i)}, \sigma_k^{(i)}) = p(x_n \mid z_{nk})$ being the kth Gaussian component (see section A.3) and $\tau_k^{(i)} = p(z_{nk})$ its weighting factor, this equates as given by (C.4) to

$$\mathbb{E}\{z_{nk}^{(i+1)} \mid x_n, \theta_k^{(i)}\} = \frac{\mathcal{N}(x_n; \mu_k^{(i)}, \sigma_k^{(i)})\tau_k^{(i)}}{\sum_{k=1}^{K} \mathcal{N}(x_n; \mu_k^{(i)}, \sigma_k^{(i)})\tau_k^{(i)}} = p(z_{nk} = 1 \mid x_n). \tag{4.36}$$

  Due to the labels are of binary nature, the posterior expectation gets equivalent to the posterior probability for z_{nk} given the voxel value x_n (4.36). Note that in this case no second step determines a discrete labeling as done in (4.10) and (4.25). In case of EMGMM, the expectation calculated in (4.36) serves directly for the following parameter estimation step having values in the interval of $[0, 1]$. Thus we calculate the MMSE estimator for each label

$$z_{nk,\mathrm{MMSE}}^{(i+1)} = \mathbb{E}\{z_{nk} \mid x_n, \theta_k^{(i)}\}, \tag{4.37}$$

  which should not be confused with the binary labels z_{nk}.

- Parameter estimation (m-step): Using some initial/previous estimate of the expected labeling and following the derivation in section C.1, the Gaussian mean and standard deviation get updated via

$$\mu_k^{(i+1)} = \frac{\sum_{n=1}^{N} x_n z_{nk,\mathrm{MMSE}}^{(i)}}{\sum_{n=1}^{N} z_{nk,\mathrm{MMSE}}^{(i)}} \tag{4.38}$$

$$\sigma_k^{(i+1)} = \sqrt{\frac{\sum_{n=1}^{N}(x_n - \mu_k^{(i+1)})^2 z_{nk,\mathrm{MMSE}}^{(i)}}{\sum_{n=1}^{N} z_{nk,\mathrm{MMSE}}^{(i)}}}. \tag{4.39}$$

As in (4.11) and (4.12), these parameters represent the empirical cluster mean and the empirical standard deviation. But in (4.38) and (4.39) we use expectations of labels rather than labels themselves. This results in updating the parameters as weighted averages.

Finally solving (C.5) for τ_k, the prior probabilities get updated via

$$\tau_k^{(i+1)} = \sum_{n=1}^{N} \frac{z_{nk,\mathrm{MMSE}}^{(i)}}{N}. \tag{4.40}$$

Although $\mathbf{Z}$ are further discrete labellings, the quantities used during the EM procedure are their expectations $\mathbb{E}\{z_{nk} \mid x_n; \theta_k\}$. These are real numbers that lie in the interval $[0,1]$ and sum to one regarding the index k. With (4.38), (4.39)) and (4.40), weighted averages get calculated differing from the approach discussed in section 4.1.1 and section 4.1.2 where the averages where built up from discrete labellings. Iterating the EM algorithm till convergence, i.e., $|\Theta^{(i)} - \Theta^{(i+1)}|^2 < \epsilon.$, the resulting expected labellings can again be used to decide for partial memberships and so to overcome PVE. The final labeling is thus calculated according to

$$z_{nk,\mathrm{MMSE}}^{\mathrm{final}} = \mathbb{E}\{z_{nk} \mid x_n; \theta_k^{\mathrm{final}}\}, \tag{4.41}$$

4.3 Bayesian Inference

As mentioned in chapter 1, the smaller the volumes to be detected get the worse the statistical ensembles of the resulting clusters get, which in turn leads to bad ML estimates for the cluster parameters (μ, σ, τ). Bayesian treatment potentially offers a way to prevent the parameters estimates from getting unreliable. Bayesian statistics assumes that the parameter under consideration are of random nature (see section 3.3.1) having prior probability distributions which are further governed by so called hyperparameters. These hyperparameters can be used to dominate main parameters which are to be inferred from bad statistical ensembles as shown by the examples in appendix B.

For complex models including various prior distributions for the parameters, solutions are no longer feasible in an analytical manner. To simplify the analysis, the use of conjugate priors is introduced in section 4.3.1. Establishing posterior distributions for the parameters using well-known probability distributions $p(\Theta \mid X, \mathbf{Z})$, the parameter estimators $\hat{\Theta}$ can be written as MMSE estimators (3.29) resulting in closed-form solutions. With conjugate priors, the maximization step of the EM procedure shown in section 4.2.1 can be augmented with penalty terms for the parameters, see section 3.3.3. A so called Bayesian EM is introduced in section 4.3.4. Moreover restricting the class of probability distributions to ones which factorize between the parameters (see section C.3) gives access to highly sophisticated combinations of distributions. Thus a variational approach is presented in section 4.3.5.

4.3.1 Conjugate Priors

As shown in section 3.3.1, an efficient estimator for the parameter of a probability distribution p exists in a Bayesian context if the derivative of $\ln p$ becomes a special functional form (3.35). An example for a Gaussian prior distribution incorporated for a Gaussian mean parameter μ is discussed in appendix B.1. Moreover in appendix B.2, an example is prepared using a Gamma prior for the Gaussian precision (i.e., the inverse variance $\lambda = \frac{1}{\sigma^2}$, see section A.3).

Equivalently this problem can be solved by writing the posterior probability (3.26) as

$$p(\Theta \mid X) = p(X \mid \Theta)\frac{p(\Theta)}{p(X)} \propto p(X \mid \Theta)p(\Theta). \tag{4.42}$$

Using a prior distribution for the according parameter $p(\Theta)$ which has the same functional dependency on Θ as the conditional probability $p(X \mid \Theta)$, simple rearrangement shows that the posterior $p(\Theta \mid X)$ and prior distribution are also having the same functional form. Therefore the posterior expectation of the according parameter is given by a well known equation, i.e., the expectation of the parameter of the prior distribution which is given by the MMSE (3.29). This leads to the definition of conjugate prior distributions. E.g., for the univariate Gaussian mean parameter μ the conjugate prior is given by a Gaussian (see also the example in appendix B.1) whereas for the univariate Gaussian precision λ the conjugate prior is a Gamma distribution (see appendix B.2).

Employing conjugate priors, the MMSE estimator of Θ can be written as the posterior expectation of Θ according to (3.29) as follows

$$\hat{\Theta}_{\mathrm{MMSE}} = \mathbb{E}_p\{\Theta \mid X\}. \tag{4.43}$$

Hence $\mathbb{E}_p\{\Theta \mid X\}$ is the mean value according to the distribution $p(\Theta \mid X)$, simplifying the analysis of the parameter updates.

In the following two sections, we show that the derivations in the examples of appendix B.1 and appendix B.2 do indeed yield the same results as the approach using conjugate priors.

4.3.2 Gaussian Prior for the Mean

To demonstrate the inclusion of a conjugate prior for the mean of a Gaussian, the data X is assumed to be a Gaussian distributed vector with known variance σ^2 and conditional distribution written as $p(X \mid \Theta) = p(X \mid \mu) = \mathcal{N}(X \mid \mu, \sigma^2)$. As already mentioned in section 4.3.1, the prior distribution for

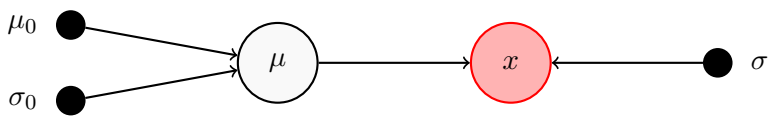


Figure 4.8: Bayesian network representation for a voxel x and the random parameter μ as used in (4.44). The parameter σ is assumed deterministic and known. Moreover the hyperparameters μ_0 and σ_0, governing the prior probability distribution of μ, are depicted. The deterministic parameters and hyperparameters are shown by little black nodes with labels located outside the node.

the mean parameter μ is also Gaussian given by $p(\Theta) = p(\mu) = \mathcal{N}(\mu \mid \mu_0, \sigma_0^2)$. Hence the posterior distribution $p(\mu \mid X)$ can be computed as

$$\begin{aligned} p(\mu \mid X) &\propto p(X \mid \mu)p(\mu) \\ &\propto \exp\left\{ -\frac{1}{2\sigma^2} \sum_{n=1}^{N} (x_n - \mu)^2 - \frac{1}{2\sigma_0^2}(\mu - \mu_0)^2 \right\}. \end{aligned} \tag{4.44}$$

The model defined by to (4.44) is depicted in figure 4.8. The fact that the parameter μ has become a random variable is accounted for by representing it via a random variable node. The hyperparameters μ_0 and σ_0 are assumed deterministic and are hence depicted as little black nodes with their label located outside the node.
Keeping just terms involving μ and reorganizing them, the posterior distribution can be written as Gaussian distribution $\mathcal{N}(\mu \mid \mu_N, \sigma_N^2)$ with μ_N and σ_N given by

$$\mu_N = \frac{\sigma^2}{N\sigma_0^2 + \sigma^2}\mu_0 + \frac{N\sigma_0^2}{N\sigma_0^2 + \sigma^2}\hat{\mu}_{\mathrm{ML}} \tag{4.45}$$

and

$$\frac{1}{\sigma_N^2} = \frac{1}{\sigma_0^2} + \frac{N}{\sigma^2}. \tag{4.46}$$

Here,

$$\hat{\mu}_{\mathrm{ML}} = \frac{1}{N} \sum_{n=1}^{N} x_n, \tag{4.47}$$

is the ML estimate of the mean parameter μ. Comparing these results with (B.3) and (B.4) show the equivalence of both methods.

We next consider two extreme cases of (4.45). The first one occurs if large data ensembles with $N \to \infty$ are under consideration. In this case the first term in (4.45) is negligible and the conditional mean is dominated by the ML estimator, $\mu_N \approx \hat{\mu}_{\mathrm{ML}}$, which is calculated from the data according to (4.47). The second case corresponds to very informative prior distributions, i.e., $\sigma_0^2 \to 0$. In this case, the second term of (4.45) becomes small and hence $\mu_N \approx \mu_0$. Therefore if the data set is small and the estimate $\hat{\mu}_{\mathrm{ML}}$ is poor, the parameter of the prior distribution determines the estimate μ_N.

Figure 4.9 visualizes the behaviour of the posterior distribution (4.44) as N grows. The variances of both, conditional and prior distribution are assumed to be equal 0.1. The mean of the conditional is chosen 0.8 and the mean of the prior is zero. With this, the black bell curve in figure 4.9 emerges from drawing the posterior distribution with $N = 0$ equal to prior distribution. Hence the part arising from the conditional distribution is vanishing resulting in a Gaussian of zero mean and variance equal 0.1. As N increase, the contribution of the conditional distribution is raised and the mean of the resulting bell curve is shifted towards a center of 0.8.

4.3.3 Gamma Prior for the Precision

To provide a closed form expression for the MMSE of the variance of a Gaussian distribution, it is easier to work with the precision λ, which is defined as the inverse of the variance $\lambda = \frac{1}{\sigma^2}$. We

therefore search for a conjugate prior distribution which is proportional to a product of a power of λ and the exponential of a linear function of λ. As mentioned in section 4.3.1 this is valid for the gamma distribution (cf. appendix A.6),

$$\mathrm{Gam}(\lambda \mid a_0, b_0) = \frac{1}{\Gamma(a_0)} b^{a_0} \lambda^{a_0 - 1} e^{-b_0 \lambda}. \tag{4.48}$$

We again assume the data X to be a Gaussian distributed vector with known mean μ and conditional distribution written as $p(X \mid \Theta) = p(X \mid \lambda) = \mathcal{N}(X \mid \mu, \lambda^{-1})$. Applying the product rule using the conditional distribution with the prior distribution (4.48), the posterior distribution $p(\lambda \mid X)$ can be written as

$$\begin{aligned} p(\lambda \mid X) &\propto \mathcal{N}(X \mid \lambda)\mathrm{Gam}(\lambda \mid a_0, b_0) \\ &\propto \lambda^{\frac{N}{2}} \lambda^{a_0 - 1} \exp\left\{ -\lambda b_0 + \frac{\lambda}{2} \sum_{n=1}^{N} (x_n - \mu)^2 \right\}. \end{aligned} \tag{4.49}$$

The model associated with (4.49) describes a model which is depicted in figure 4.10 where we have introduced the deterministic hyperparameters a_0 and b_0 for the Gamma prior. Again the parameter λ, now described as a random variable, is drawn as random variable node. The deterministic parameters are depicted as little black nodes with their labels located outside the node. Rearranging (4.49) and keeping just terms involving λ shows that the posterior distribution is again a Gamma distribution with parameters a_N and b_N given by

$$a_N = \frac{N}{2} + a_0 \tag{4.50}$$

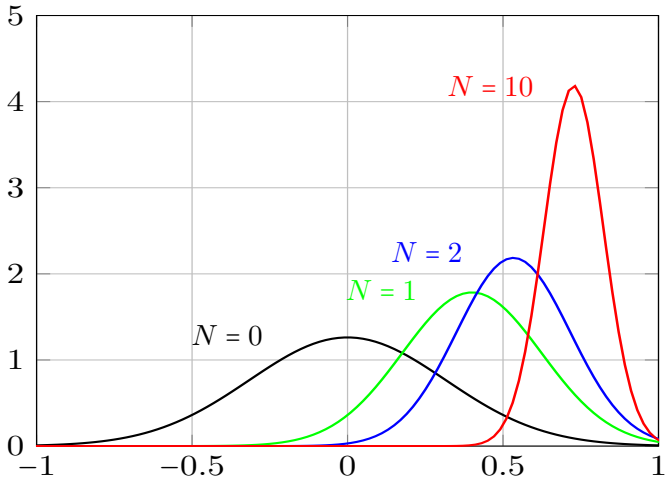


Figure 4.9: Illustration of Bayesian inference for the mean μ of a Gaussian distribution. The various curves show the posterior distribution given by (4.44) with different N. The standard deviation σ of the conditional distribution is chosen 0.1 and the mean $\mu = 0.8$. The standard deviation σ_0 of the prior is also chosen 0.1 but with vanishing mean μ_0.

$$b_N = b_0 + \frac{1}{2}\sum_{n=1}^{N}(x_n - \mu)^2 = b_0 + \frac{N}{2}\sigma^2_{\mathbf{ML}} \tag{4.51}$$

where σ^2_{ML} is the maximum likelihood estimator of the variance.

To better understand this result, we recall that the mean of the gamma distribution is given by $\mathbb{E}\{\lambda\} = \frac{a_N}{b_N}$, see (A.41). In case of no data, $N = 0$, all that is left from (4.50) and (4.51) is a_0 respectively b_0 and therefore the MMSE estimator of the precision is solely determined via the hyperparameters, $\lambda_{\mathrm{MMSE}} = \frac{a_0}{b_0}$. In contrast, the case $N \to \infty$ leads to negligible hyperparameters a_0 and b_0 and hence the classical ML estimator of the precision if obtained $\lambda_{\mathrm{MMSE}} = \lambda_{\mathrm{ML}}$.

Summing up the Bayesian treatment of the parameters, it is seen that it influences the parameter estimation noticeably only in case of small statistical data ensembles which was aimed to be one of the characteristics of the segmentation algorithm. It seems to be a good mechanism to correct for uncertainty of the parameter estimation in case of vanishing data samples and offers a way to incorporate prior information to the estimation procedure.

4.3.4 Bayesian EM for a GMM

The aim of this section is to include prior probabilities into the clustering process for the control of parameters which have to be inferred from bad statistical ensembles. To obtain an iterative algorithm as in the previous sections we again employ an EM framework as done for the naive GMM model in section 4.2.1.

As shown in section 3.3.3, the basic EM procedure discussed in section 3.3.2.1 can be expanded by including prior probabilities via calculation of MAP estimators rather than the ML estimators. It is shown in (3.45), that in such cases the expectation step does not include averaging over the priors. So if we are just augmenting the naive GMM from section 4.2.1 with priors for the mean and precision, the expectation step for a Bayesian EMGMM (BEMGMM) stays exactly the same as for the naive EMGMM (4.36).

Using a Gaussian prior for the mean $p(\mu) = \mathcal{N}(\mu \mid \mu_0, \sigma_0^2)$ and a Gamma prior for the precision $p(\lambda) = \mathrm{Gam}(\lambda \mid a_0, b_0)$, we define the joint probability of X, $\mathbf{Z}$, μ and λ to be written as

$$p(X, \mathbf{Z}, \mu, \lambda) = \mathcal{N}(X \mid \mathbf{Z}, \mu, \lambda^{-1})\mathrm{GBer}(\mathbf{Z} \mid \tau)\mathcal{N}(\mu \mid \mu_0, \sigma_0^2)\mathrm{Gam}(\lambda \mid a_0, b_0), \tag{4.52}$$

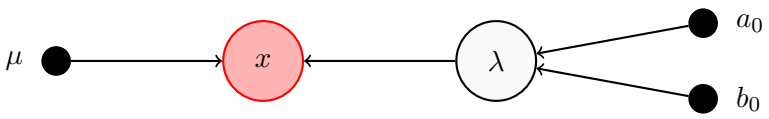


Figure 4.10: Bayesian network representation for a voxel x and the random parameter λ as used in (4.49). The parameter μ is assumed deterministic and known. Moreover the hyperparameters a_0 and b_0, governing the prior probability distribution of λ, are depicted. The deterministic parameters and hyperparameters are shown by little black nodes with labels located outside the node.

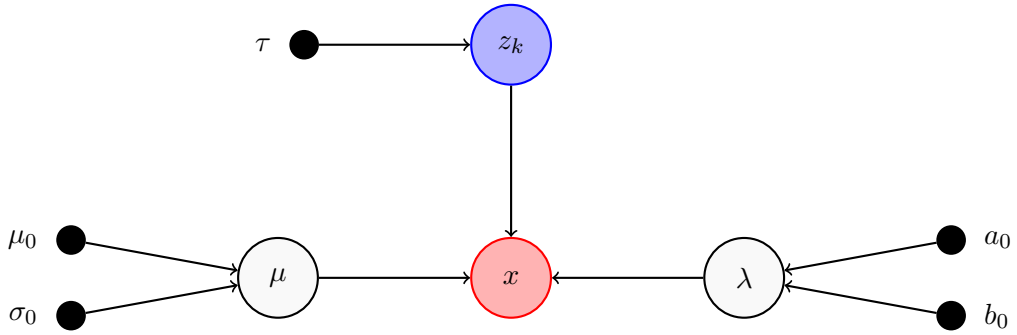


Figure 4.11: Bayesian network representation for a label-data-pair $\{x, z\}$ and the random parameters μ and λ as used in (4.52). The hyperparameters μ_0 and σ_0 govern the prior distribution over μ. The hyperparameters a_0 and b_0 govern the prior distributions over λ. Moreover the prior distribution for the label matrix $\mathbf{Z}$ is governed by the deterministic parameter τ.

The model in (4.52) is depicted in figure 4.11 for a single label-data-pair $\{x, z\}$. Note that again the various voxels are considered statistically independent Gaussian distributed random variables in (4.52). The only difference to the naive EMGMM from section 4.2.1 is that the mean μ and the precision λ are also considered as random variables and are therefore drawn as random variable nodes in figure 4.11. The according hyperparameters governing the prior distributions, μ_0, σ_0, a_0 and b_0, are assumed deterministic. Also the parameter τ, which define the family of generalized Bernoulli distributions over the labels z is a deterministic quantity.

Inserting the model (4.52) into the Bayesian expectation step (3.45) we get

$$\begin{aligned}\mathbb{E}\{\ln[\mathcal{N}(X \mid \mathbf{Z}, \mu, \lambda^{-1})\mathrm{GBer}(\mathbf{Z} \mid \tau)] \mid X; \Theta^{(i)}\} + \ln[\mathcal{N}(\mu \mid \mu_0, \sigma_0^2)\mathrm{Gam}(\lambda \mid a_0, b_0)] \\ = Q(\Theta, \Theta^{(i)}) + \ln p(\Theta)\end{aligned} \tag{4.53}$$

From (4.53) we see that the expectation step stays the same as with naive EMGMM (4.36). In contrast, the maximization step takes the prior distributions into account.

Hence an iterative BEMGMM algorithm can be formulated as follows:

- Label estimation (e-step): As mentioned, the expectation step is equivalent to the expectation step of the naive EMGMM (4.36). With some initial/previous estimate of the parameters $\Theta^{(i)}$, the posterior expectation of a labeling is calculated using Bayesian theorem (3.11) according to

$$\mathbb{E}\{z_{nk}^{(i+1)} \mid x_n, \theta_k^{(i)}\} = \frac{\mathcal{N}(x_n; \mu_{N,k}^{(i)}, \mathrm{var}_{N,k}^{(i)})\tau_k^{(i)}}{\sum_{k=1}^{K} \mathcal{N}(x_n; \mu_{N,k}^{(i)}, \mathrm{var}_{N,k}^{(i)})\tau_k^{(i)}}. \tag{4.54}$$

  with $\mathrm{var}_{N,k}^{(i)} = \frac{1}{\lambda_{N,k}^{(i)}}$.

  Hence a final labeling can be established using the MMSE estimator according to (4.37).

- Parameter estimation (m-step): Using some initial/previous estimate of the expected labeling, instead of differentiating (4.53) regarding each parameter and evaluating the MAP estimator we follow the procedure in section 4.3.2 and section 4.3.3 to evaluate the MMSE estimator.

  As derived in (4.45), the mean values for each cluster k are calculated as

$$\mu_{k,N} = \frac{\hat{\mathrm{var}}_{k,\mathrm{ML}}^{(i+1)}}{N_k^{(i+1)}\sigma_{k,0}^2 + \hat{\mathrm{var}}_{k,\mathrm{ML}}^{(i+1)}}\mu_{k,0} + \frac{N_k^{(i+1)}\sigma_{k,0}^2}{N_k^{(i+1)}\sigma_{k,0}^2 + \hat{\mathrm{var}}_{k,\mathrm{ML}}^{(i+1)}}\hat{\mu}_{k,\mathrm{ML}}^{(i+1)}. \tag{4.55}$$

  According to (4.50) and (4.51), the parameters governing the posterior distribution of λ can be given for each cluster k according to

$$a_{N,k} = \frac{N_k^{(i+1)}}{2} + a_{k,0} \tag{4.56}$$

$$b_{N,k} = b_{k,0} + \frac{1}{2}\sum_{n=1}^{N}(x_n - \hat{\mu}_{k,\mathrm{ML}}^{(i+1)})^2 = b_{k,0} + \frac{N_k^{(i+1)}}{2}\hat{\mathrm{var}}_{k,\mathrm{ML}}^{(i+1)}, \tag{4.57}$$

  with

$$N_k^{(i+1)} = \sum_{n=1}^{N} z_{nk,\mathrm{MMSE}}^{(i)} \tag{4.58}$$

  and

$$\mu_{k,\mathrm{ML}}^{(i+1)} = \frac{1}{N_k^{(i+1)}}\sum_{n=1}^{N} x_n z_{nk,\mathrm{MMSE}}^{(i)}, \qquad \hat{\mathrm{var}}_{k,\mathrm{ML}}^{(i+1)} = \frac{1}{N_k^{(i+1)}}\sum_{n=1}^{N}(x_n - \mu_{k,\mathrm{ML}}^{(i+1)})^2 z_{nk,\mathrm{MMSE}}^{(i)}. \tag{4.59}$$

  Hence the precision is given by

$$\lambda_{N,k}^{(i+1)} = \frac{a_{N,k}}{b_{N,k}}. \tag{4.60}$$

  Finally the parameter τ is updated for each cluster k as done by (C.5) via ML estimation as

$$\tau_k^{(i+1)} = \sum_{n=1}^{N}\frac{z_{nk,\mathrm{MMSE}}^{(i)}}{N}. \tag{4.61}$$

The BEMGMM procedure listed above corresponds to an EMGMM procedure with differing parameter updates for the Gaussian parameters. Hence in case of small tumor lesions having bad statistical ensembles $N_k \to 0$, μ_k and σ_k will be dominated by the hyperparameters μ_0, σ_0, a_0 and b_0.

4.3.5 Variational Bayesian Inference for a GMM

To incorporate a Bayesian treatment described in section 3.3.1 to all of the parameters of the GMM model employed in section 4.2.1, the basic GMM as introduced in appendix A.4 is considered with the whole parameter vector $\Theta = (\tau, \mu, \sigma)$ defined as a random vector. For each of those quantities a conjugate prior distribution as described in (section 4.3.1) is imposed.

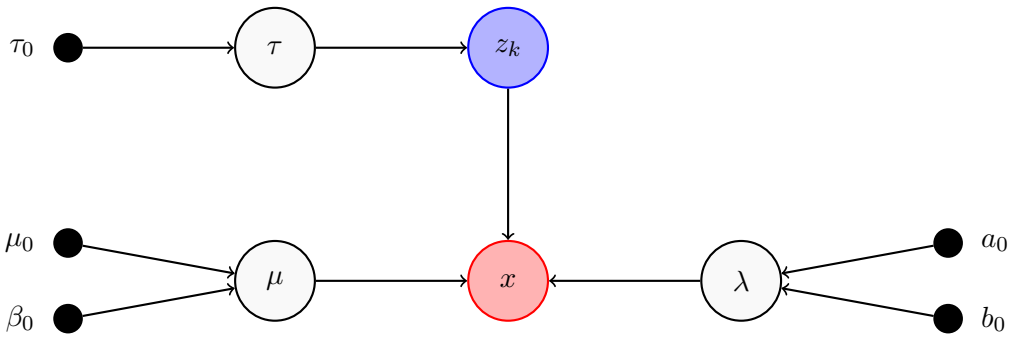


Figure 4.12: Bayesian network representation for a label-data-pair $\{x, z\}$ and the random parameters μ, λ and τ as used in (4.62). The hyperparameters μ_0 and β_0 govern the prior distribution over μ. The hyperparameters a_0, b_0 as well as β_0 govern the prior distributions over λ. Moreover the hyperparameter τ_0 govern the prior distribution of τ.

As shown in section 4.3.1 the conjugate priors for the mean parameters of the GMM are again Gaussian. Using the precisions rather than the standard deviations (see appendix A.3), the conjugate prior is given by a Gamma distribution (cf. appendix A.6). A specific choice is to introduce a constant β_0 and let the precision of the prior for μ be a linear function of the precision of the conditional distribution for X, see (4.62) below. Lastly the prior distribution for the weighting factor τ can be written in form of a Dirichlet distribution (cf. appendix A.7).

With this choice the joint probability reads

$$p(X, \mathbf{Z}, \mu, \lambda, \tau) = \mathcal{N}(X \mid \mathbf{Z}, \mu, \lambda^{-1})\mathrm{GBer}(\mathbf{Z} \mid \tau)\mathcal{N}(\mu \mid \mu_0, (\beta_0\lambda)^{-1})\mathrm{Gam}(\lambda \mid a_0, b_0)\mathrm{Dir}(\tau \mid \alpha_0), \quad (4.62)$$

Introducing these prior probabilities gives rise to a new vector of hyperparameters $\Theta_0 = (\mu_0, \beta_0, a_0, b_0, \alpha_0)$ which can be chosen to help with the problem of bad ensembles. Following the derivation in appendix C.4 and recall section 4.3.1 we can state, that in case of poor statistical ensembles the prior distributions of the parameters Θ carry more information than the conditional distribution $p(X \mid \Theta)$. Moreover, the prior distributions of each parameter comprised in Θ are governed by hyperparameters, which enables interference of the estimation process in case small tumor size. Figure 1.2 shows a histogram of a highly radiating spherical object (28mm diameter) in low background activity measured with PET. This graph highlights the poorness of the statistical ensembles for objects with even 11.49cm^3. The low resolution of the PET scanners (see chapter 2) coupled with PVE are responsible for this fact.

As mentioned, a direct inference of parameter estimates or labeling configurations $\mathbf{Z}$ for a complex model like (4.62) are not traceable. A first simplification is done using conjugate prior distributions as shown in the previous sections. Moreover as stated in appendix C.2, which illuminates the behaviour of the EM procedure summarized in section 3.3.2.1, instead of optimizing some probability measure, the KL divergence can serve as objective function. Assuming further (see section C.3) that the joint probability distribution factorizes among the parameters as well as among the labels $\mathbf{Z}$ (C.16), a variational approach can solve the optimization problems. A comprehensive derivation of the variational

version of the EMGMM algorithm using the model presented in 4.62 is given in appendix C.4. The e-step and m-step equivalents are summarized in the following paragraph.

- E-step: With some initial guess of the parameters $\Theta^{(i)}$ and $\Theta_0^{(i)}$, the expectation of a labeling is calculated according to

$$z_{nk}^{(i+1)} = \mathbb{E}[z_{nk}] = \prod_n \prod_k \frac{\rho_{nk}}{\sum_k \rho_{nk}} \tag{4.63}$$

with

$$\begin{aligned} \ln \rho_{nk} &= \mathbb{E}_\tau\{\ln \tau_k\} - \frac{1}{2}\ln(2\pi) + \mathbb{E}_\lambda\{\ln \lambda_k\} + \\ &\quad \frac{1}{2}\mathbb{E}_\lambda\{\lambda_k\}\Big[x_n^2 - 2x_n\mathbb{E}_{\mu|\lambda}\{\mu_k\} + \mathbb{E}_{\mu|\lambda}\{\mu_k^2\})\Big] \end{aligned} \tag{4.64}$$

- M-step: Using the expected labeling from the E-step, the expectations needed in (4.64) to calculate the next labeling get updated via

$$\mathbb{E}_\tau\{\ln \tau_k\} = \psi(\alpha_{N,k}) - \psi\Big(\sum_k \alpha_{N,k}\Big) \tag{4.65}$$

$$\mathbb{E}_{\mu|\lambda}\{\mu_k\} = \mu_{N,k} \tag{4.66}$$

$$\mathbb{E}_{\mu|\lambda}\{\mu_k^2\} = \mu_{N,k}^2 + (\beta_{0,k}\lambda_k)^{-1} \tag{4.67}$$

$$\mathbb{E}_\lambda\{\lambda_k\} = \frac{a_{N,k}}{b_{N,k}} \tag{4.68}$$

$$\mathbb{E}_\lambda\{\ln \lambda_k\} = \frac{d}{da_{N,k}} \ln \Gamma(a_{N,k}) - \ln b_{N,k} \tag{4.69}$$

with MMSE estimators of the corresponding parameters given by

$$\alpha_{N,k} = \alpha_{0,k} + \sum_n z_{nk}^{(i)} \tag{4.70}$$

$$\mu_{N,k} = \Big(\beta_{0,k}\mu_{0,k} + \sum_n x_n z_{nk}^{(i)}\Big)\beta_{N,k}^{-1} \tag{4.71}$$

$$\beta_{N,k} = \sum_n z_{nk}^{(i)} + \beta_{0,k} \tag{4.72}$$

$$a_{N,k} = \sum_n \frac{z_{nk}^{(i)} + 1}{2} + a_{0,k} \tag{4.73}$$

$$b_{N,k} = b_{0,k} + \frac{1}{2}\Big[\sum_{n=1}^{N} z_{nk}^{(i)}(x_n + \mu_k)^2 + \beta_{0,k}(\mu_k - \mu_{0,k})^2\Big]. \tag{4.74}$$

See appendix C.4 for further details.

4.4 Graphical Models

In the previous discussion, only interactions among the data x_n have been modeled via local precision matrices, see section 4.1.2. Hence no interactions among voxels of different clusters were considered. Note that in (4.16) the labels of neighbouring voxels z_{nk} and z_{mk} carry the same cluster index k.

Moreover, a framework for the refinement of the parameter estimation was established, which is not involving any dependencies among the data X or the latent labeling matrix $\mathbf{Z}$. Here again it is emphasized that the Bayesian framework provides an interface for incorporating information learned from labeled data. To follow the ideas of incorporating dependencies among voxels and to counteract distortions of the images via PVE, graphical models offer a way for a mathematical description of neighbourhood relations. As discussed in section 3.4, the use of graphical models provides such a framework accounting for interactions of random variables. With MRFs, a Potts like model (see section 3.4.2) can be formulated where the probabilities are written via potential functions depending on the various vertex/voxel connections of the underlying graph.

4.4.1 Markov Random Fields - Potts Model

According to the definitions in section 3.4, the vertex set of an MRF for our PET image segmentation task should comprise the data vector X and the labeling matrix $\mathbf{Z}$. Moreover the edge set $\mathcal{E}$ consists of voxel pairs which are connected according to certain neighbourhood relations (probability distributions), see figure 4.13. Note that the edge set $\mathcal{E}$ consists of links connecting the labels of $\mathbf{Z}$ rather than the data X as it was the case for the MLGMC model in section 4.1.2.

Following the ideas from the previous sections, the relations induced by the edges connecting the data X with the labels of $\mathbf{Z}$ are given by probability distributions which are equivalent to the conditional probabilities of a GMM $p(X \mid \mathbf{Z})$, see (A.30). To incorporate local dependencies between neighbouring voxels, these Gaussian terms are combined with a Potts model known from statistical physics. Using the exponential description from section 3.4 a Gaussian MRF (GMRF) with pairwise interactions (edges which establish connections among the labels of $\mathbf{Z}$ as shown in figure 4.13) can be formulated like an Ising model via

$$p(X,\mathbf{Z}) = p(X \mid \mathbf{Z})p(\mathbf{Z}) = \prod_k \prod_l \exp\Big\{ \underbrace{\sum_{n\in\mathcal{V}} z_{nk}[\gamma_k x_n - \gamma_k' x_n^2 - A(\gamma_k,\gamma_k')]}_{\approx \mathcal{N}(X|\mu,\sigma^2)} \Big\}$$
$$\exp\Big\{ \underbrace{\sum_{n\in\mathcal{V}} z_{nk}\alpha_k}_{\approx \mathrm{GBer}(\mathbf{Z}|\tau)} + \sum_{(n,m)\in\mathcal{E}} z_{nk}\tilde{\alpha}_{kl} z_{ml} - A(\alpha,\tilde{\alpha}) \Big\}, \tag{4.75}$$

which is also termed a Gibbs random field (GRF). The two underbraces indicate the parts which can respectively be related to the Gaussian conditional probability (A.30) for the data and another part which is related to the generalized Bernoulli prior probability (A.28). The remaining term is responsible for the interactions of voxels and further comprises the partition function $A(\alpha,\tilde{\alpha})$. It remains to define the edge set $\mathcal{E}$ of our graph. One possible Gibbs model (4.75) is given by the graph visualized in figure 4.13. In an image segmentation context this graph structure can be understood as an image in two dimensions, where each horizontally and vertically neighbouring pixel pair is sharing an interaction term in (4.75). Extending this graph to our three-dimensional PET image problem is accomplished by adding one more dimension and drawing three-dimensional grids for the data X as

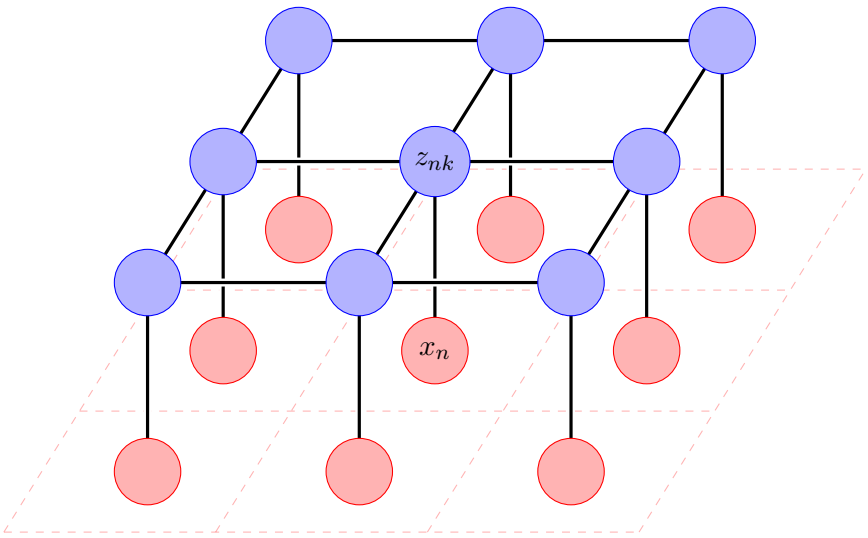


Figure 4.13: One possible graph structure visualizing the neighbourhood relations described by the probability distribution presented in (4.75).

well as for the label matrix $\mathbf{Z}$. The neighbourhood of a voxel x_n in three dimensions would be spanned by the six neighbouring voxels which share a plane with x_n. [2]

In contradiction to the models described in earlier sections incorporating correlations among the data X (section 4.1.2), the neighbourhood dependencies of the GMRF are imposed among the labels of $\mathbf{Z}$. Thus not only dependencies among labels of the same cluster but also dependencies among labels of different clusters are accounted for.

We will see in section 4.4.2.2, when it is about calculating local marginal probabilities of $\mathbf{Z}$ to achieve a labeling, that local conditional probabilities are needed. As discussed in section 3.4.1 and section 3.4.2, conditioning a voxel x_n on all its neighbouring voxels renders x_n disconnected from the rest of the graph. Hence all that is required to know for the calculation of the marginal probability of x_n is the neighbourhood of x_n (which is then a separator set). Therefore we use the Hammersley-Clifford theorem (see section 3.4.2) to calculate the local probability of voxels conditioned on their neighbourhood as (see also [31])

$$p(x_n \mid z_{nk})p(z_{nk} \mid \mathcal{N}(z_n)) = \prod_k \exp\left\{z_{nk}[\gamma x_n - \gamma' x_n^2 - A(\gamma_k, \gamma_k')]\right\}$$
$$\exp\left\{z_{nk}\alpha_k + \sum_{m \in \mathcal{N}(z_n)} z_{nk}\tilde{\alpha}_{kl} z_{ml} - A_n(\alpha, \tilde{\alpha})\right\}, \tag{4.76}$$

making the system amenable for the sampling of labeling probabilities (see section 3.4.6). The part of the terms concerning the data x_n and the labeling z_{nk} are known as sufficient statistics. In (4.76)

[2] Physically, the voxels in three-dimensional PET image can be viewed as small cubes arranged in an three-dimensional grid. In this sense, each voxel is surrounded by six voxel (except the voxel lying on the outer hull). This view is different to the description where the nodes (voxels) are located at the corners of a three-dimensional grid.

they are given by

$$T_\gamma = z_{nk}x_n, \quad T_{\gamma'} = z_{nk}x_n^2, \quad T_\alpha = z_{nk}, \quad T_{\tilde{\alpha}} = z_{nk}z_{ml}. \tag{4.77}$$

The partition function for each Gaussian component is easily found via completing the square in the sum and comparing coefficients with (A.39). This leads to the equation

$$A(\gamma_k, \gamma_k') = \frac{1}{2}\ln\Big(\frac{\gamma_k'}{\pi}\Big) - \frac{\gamma_k^2}{4\gamma_k'}. \tag{4.78}$$

The relation between the Gaussian parameters and the new parameter pair (γ, γ') is

$$(\gamma_k, \gamma_k') = \Big(\frac{\mu_k}{\sigma_k^2}, \frac{1}{2\sigma_k^2}\Big). \tag{4.79}$$

This allows us to estimate the Gaussian parameters as before, transforming mean and standard deviation to fit the Gaussian MRF via (4.79).

The partition function for the Potts model distribution being part of the joint distribution (4.75), i.e. $A(\alpha, \tilde{\alpha})$, has to be calculated via summation over all labeling configurations as defined by (3.55)

$$A(\alpha, \tilde{\alpha}) = \ln \sum_{\mathbf{Z}} \exp\Big\{ \sum_{n\in\mathcal{V}} z_{nk}\alpha_k + \sum_{(n,m)\in\mathcal{E}} z_{nk}\tilde{\alpha}_{kl}z_{ml}\Big\}. \tag{4.80}$$

In contrast the Potts term contributing to the local conditional distribution (4.76) has to be normalized according to

$$A_n(\alpha, \tilde{\alpha}) = \ln \sum_{k} \exp\Big\{z_{nk}\alpha_k + \sum_{m\in\mathcal{N}(z_n)} z_{nk}\tilde{\alpha}_{kl}z_{ml}\Big\}, \tag{4.81}$$

where z_{ml} is assumed given.

With this setting the iterative procedures, as employed in the previous algorithms alternating between a labeling step and parameter estimation step, does no longer yield analytical formulations. Trying to perform a labeling step, a problem arises due to the local interactions of the label matrix which does not permit to calculate the marginal or posterior probability of a simple label z_{nk} directly. Also the parameter estimation step for the parameters α and $\tilde{\alpha}$ becomes infeasible. In previous sections this was achieved by differentiating the logarithmic likelihood function. Using a GMRF as (4.75) normalized by a partition function shown in (4.80), this would require to evaluate the derivative of a sum over all labeling configurations.

Section 4.4.2 and section 4.4.3 discuss how to circumvent these problems.

4.4.2 Labeling

To go for a labeling, the MLGM procedure as well as the EMGMM procedure are calculating the conditional probabilities for each z_{nk} by simply using Bayes theorem, see (4.34) and (4.36). Since the GMM used in section 4.2.1 does not include dependencies between different voxels, this can be done in an analytical manner. Even for the MLGMC approach shown in section 4.1.2, which incorporates local

correlations among the data X, an equation was established to approximate conditional probabilities for single voxel labels (4.13).

When incorporating local interactions among the labels of $\mathbf{Z}$ via a MRF, the calculation of each marginal of $\mathbf{Z}$ is no longer feasible analytically. As mentioned above a bottleneck therewith is the presence of a partition function as shown in (4.80) which includes the summation over all labeling configurations of $\mathbf{Z}$. Possible ways to resolve these difficulties are given by either using sampling techniques (see section 3.4.6) or by doing marginalization via message passing (see section 3.4.3).

However, since the underlying graph exhibits cycles, it is not amenable to exact inference. The actual problem we are dealing with is far from having a tree structure. In fact, an image segmentation problem with local dependencies assumed among local neighbouring pixels/voxels is involving many small cycles. Nevertheless, loopy belief propagation can be performed.

4.4.2.1 Loopy Belief Propagation

As shown in section 3.4.3, the marginals of all variables can be obtained by a summation procedure called message passing or belief propagation. The only requirement for such procedures is that the graph has tree-structure. Therewith it was possible to start the summation procedure at the leaf nodes and propagate the massages through the tree. Hence all messages can be obtained efficiently by following the sum product algorithm in section 3.4.4.1.

To determine the sum product algorithm (actually a loopy pendant) for the Potts model given in section 4.4.1, we first consider the graph shown in figure 4.14 which is a one-dimensional pendant of our three-dimensional image clustering problem. Also with regard to the procedure in two dimensions respectively in three dimensions we start marginalizing at the factor nodes h_n and the variable nodes x_n. This can be done in a single calculation step. In higher dimensions, the graph is no longer tree-structured and we have to apply loopy belief propagation, an iterative procedure. But since the data X is known, the messages have to be sent only among labels of $\mathbf{Z}$, i.e. $\mu_{Z_n \to g_{nm}}$ and $\mu_{g_{nm} \to Z_m}$, with $\mu_{h_n \to Z_n}(Z_n)$, $\mu_{x_n \to f_n}$ and $\mu_{f_n \to Z_n}$ remaining fixed. So we start calculating the messages which are to be calculated just once in any case of dimensionality, $\mu_{h_n \to Z_n}(Z_n)$, $\mu_{x_n \to f_n}$ and $\mu_{f_n \to Z_n}$.

The messages get identified using the joint probability (4.75). Following the recipe in figure 3.5 (b), for $\mu_{h_n \to Z_n}(Z_n)$ we identify the term corresponding to the generalized Bernoulli distribution

$$\mu_{h_n \to z_{nk}}(z_{nk}) = e^{z_{nk}\alpha_k}. \tag{4.82}$$

For simplicity, we introduce vector notation. In fact, a message $\mu(Z_n)$ can be considered as a vector function $\mu_k(Z_n)$. In this sense we may write instead of (4.82) the vector equation

$$\mu_{h_n \to Z_n}(Z_n) = \begin{pmatrix} e^{z_{n1}\alpha_1} \\ e^{z_{n2}\alpha_2} \end{pmatrix}. \tag{4.83}$$

Using the Hadamard-product $\circ$, two messages can be multiplied element-wise as

$$\mu_n \circ \mu_m = \begin{pmatrix} \mu_{n1}\mu_{m1} \\ \mu_{n2}\mu_{m2} \end{pmatrix}. \tag{4.84}$$

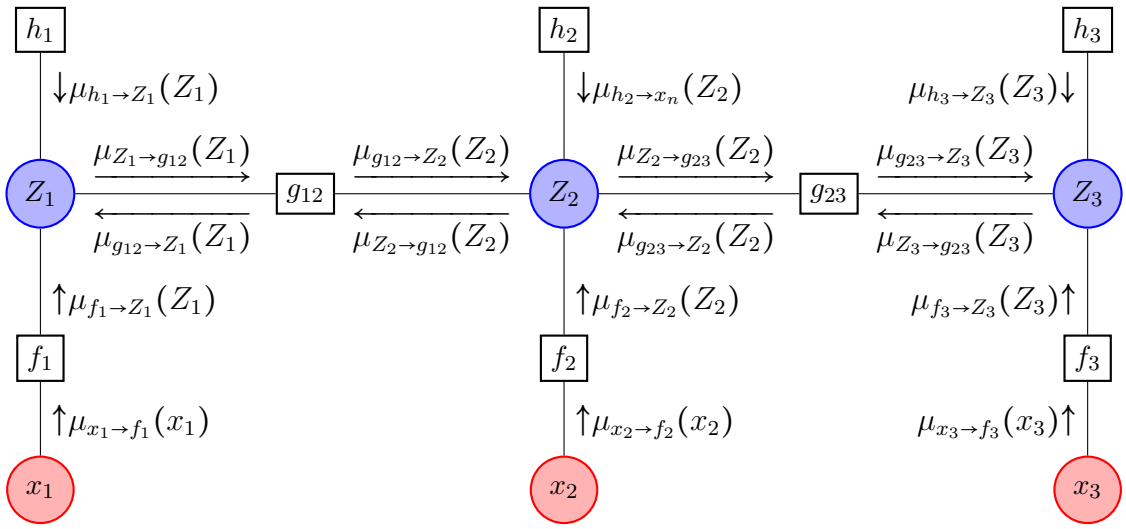


Figure 4.14: One-dimensional factor graph representation of the GMRF presented in (4.75). In this case the graph is tree-structured. Increasing the grid dimensionality of the labels $\mathbf{Z}$ and the data X according to our image segmentation task (two- or three-dimensional regular grid), the graph will be no longer tree-structured.

Moreover, according to figure 3.5 (a) the messages sent from the variable nodes x_n are simply given by

$$\mu_{x_n\to f_n}(x_n) = \begin{pmatrix}1\\1\end{pmatrix}. \tag{4.85}$$

The message from f_n to Z_n is given by

$$\mu_{f_n\to Z_n}(Z_n) = \begin{pmatrix}\mathcal{N}(x_n \mid \mu_1, \sigma_1^2)\\ \mathcal{N}(x_n \mid \mu_2, \sigma_2^2)\end{pmatrix}. \tag{4.86}$$

Finally, the messages exchanged between labels of $\mathbf{Z}$ have to be gained. Those have to be built from the interaction terms in (4.75). The message from the variable nodes Z_n to the factor nodes g_{nm} are further multiplied by the according messages from h_n and f_n to give

$$\mu_{Z_n\to g_{n,n+1}}(Z_n) = \mu_{g_{n-1,n}\to Z_n}(Z_n) \circ \begin{pmatrix}e^{z_{n1}\alpha_1}\\ e^{z_{n2}\alpha_2}\end{pmatrix} \circ \begin{pmatrix}\mathcal{N}(x_n \mid \mu_1, \sigma_1^2)\\ \mathcal{N}(x_n \mid \mu_2, \sigma_2^2)\end{pmatrix}. \tag{4.87}$$

The messages sent from the factor nodes $g_{n-1,n}$ to the variable nodes Z_n have to sum over the variables from which they get messages themselves. Using again the vector notation as in the previous equations we can express the summation as multiplication of a message vector with an interaction matrix resulting in

$$\mu_{g_{n-1,n}\to Z_n}(Z_n) = \underbrace{\begin{pmatrix}e^{z_{n1}\tilde{\alpha}_{11}} & e^{z_{n1}\tilde{\alpha}_{12}}\\ e^{z_{n2}\tilde{\alpha}_{21}} & e^{z_{n2}\tilde{\alpha}_{22}}\end{pmatrix}}_{\mathbf{G}\text{...interaction matrix}} \begin{pmatrix}\mu_{z_{n-1,l}\to g_{n-1,l}}(z_{n-1,l})\\ \mu_{z_{n-1,l}\to g_{n-1,l}}(z_{n-1,l})\end{pmatrix}. \tag{4.88}$$

Expanding the model as mentioned above by increasing the dimensionality of the voxel grid, the messages delivered among the labels Z_n and the interaction factors g_{nm} are no longer propagating on a path. Rather, each variable node is having 6 edges (in three dimensions) which are organized to induce a huge amount of small loops to the graph structure.

In figure 4.15 a grid only for the labels $\mathbf{Z}$ in two dimensions is drawn. Moreover all possible messages are depicted. A common practice is to update all messages in parallel. These procedures are called flooding algorithms. With flooding on the graph in figure 4.15, after the fourth iteration of the algorithm the message sent from each variable node is returned by all neighbours which induce a strong feedback loop. Thus it have to be proven in every special case if such proceeding is working as desired.

In case of a three-dimensional label grid and data grid (PET image with a 6-neighbourhood), the messages from a label node to a factor node (4.87) multiply all incoming messages from the remaining 5 factor nodes,

$$\mu_{Z_n \to g_{ni}}(Z_n) = \sum_{m \in \mathcal{N}(Z_n) \setminus g_{ni}} \mu_{g_{mn} \to Z_n}(Z_n) \circ \mu_{f_n \to Z_n}(Z_n) \circ \mu_{h_n \to Z_n}(Z_n). \tag{4.89}$$

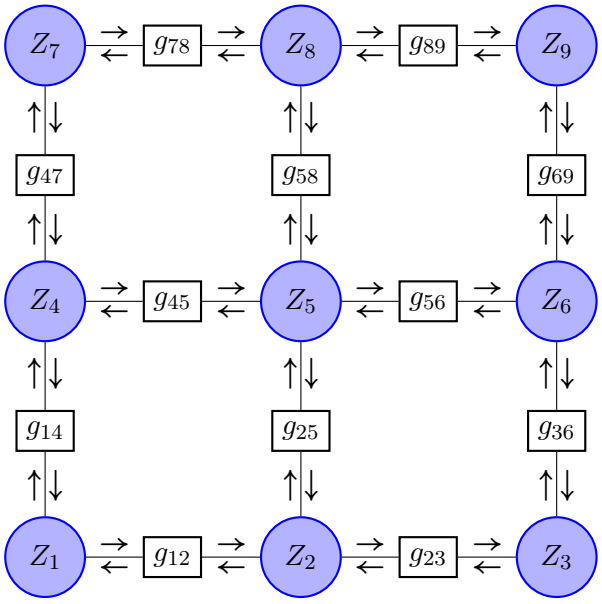


Figure 4.15: Factor graph representation of a two-dimensional label matrix $\mathbf{Z}$ used in the GMRF presented in (4.75). All messages which have to be sent are drawn. But since there are no root nodes (no tree-structured graph) it is not clear from which node to start from and which queue to follow during belief propagation. Sending all messages simultaneously in an iterative fashion is called flooding. After 4 steps of flooding (one step corresponds to sending one message from each factor node and one message from each variable node), the message each node get is influenced by a message this node was sending before 4 steps. So we are dealing with a huge amount of small loops generating feedbacks.

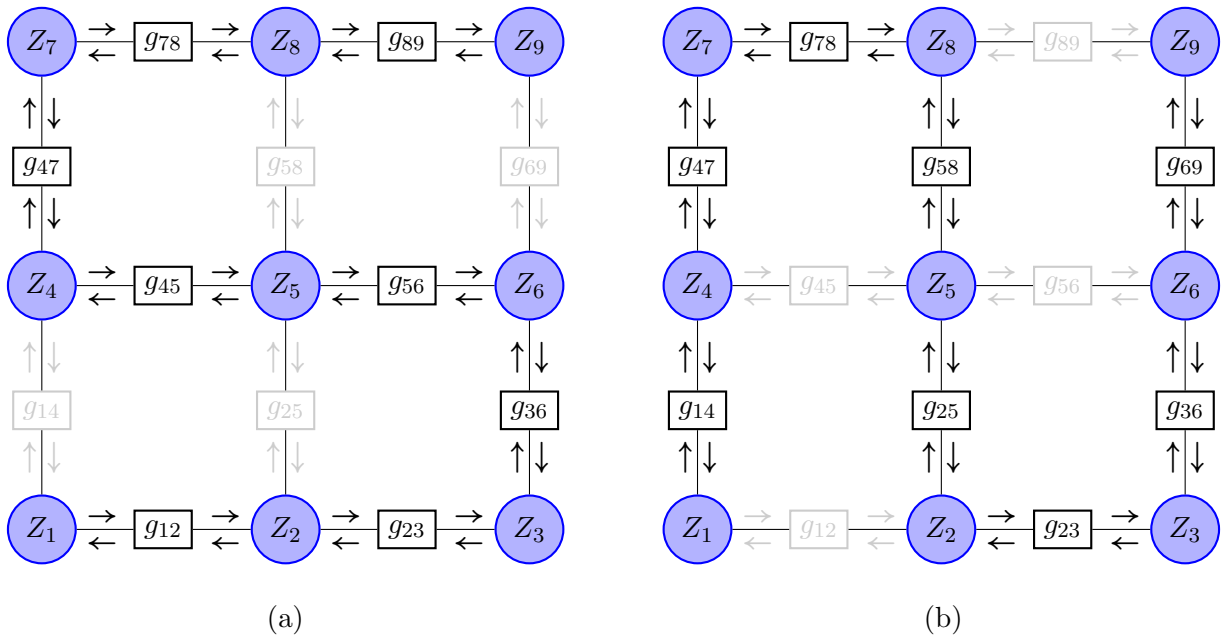


Figure 4.16: Factor graph representation of a two-dimensional label matrix $\mathbf{Z}$ used in the GMRF presented in (4.75). All messages which have to be sent are drawn. Two different paths are highlighted called zig-zag paths. A zig-zag algorithm is updating messages just along a certain path. After a defined amount of update steps the path is changed, e.g. iterating between the paths shown in (a) and (b). Hence we can circumvent short feedback loops.

A common problem with loopy belief propagation is that the cycles are acting as feedback loops. To avoid small loops in graphs having a huge amount of them as the regular grids we are dealing with, one iteration step is split to update messages just along certain path's. In this sense, the flooding algorithms are presenting an extreme case in that all messages are updated simultaneously. Another extreme case is given by using just paths which build linear chains as shown in figure 4.16. Figure 4.16 shows two update schemes which we will call zig-zag schedule. First we will update just messages along the path shown in figure 4.16 (a). After some steps we update just messages along the path shown in figure 4.16 (b). This way it is avoided that a message sent from a certain node returnes via a short loop.

4.4.2.2 Monte Carlo

As discussed in section 3.4.6, sampling methods like MCMC can be used to approximate expectations of random variables by averaging over samples which have been drawn from the respective probability distribution or some proposal distribution, see (3.72). With the Metropolis method, we simply have to change the labeling configuration locally (e.g. one label z_{nk} of the entire label matrix $\mathbf{Z}$) and calculate the ratio of the probabilities of the new and the old label configuration. Hence the fundamental problem of finding adequate search steps and search radii are not present due to the binary nature

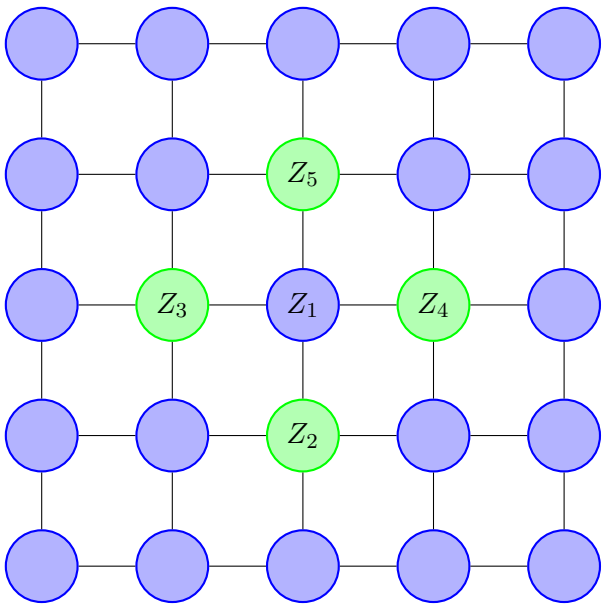


Figure 4.17: Vertex set $\mathbf{Z}$ corresponding to a 5×5 grid. Green vertices are considered to be known whereas the blue vertices are unknown. 4-neighbourhood $\mathcal{M}_1 = \{z_2, z_3, z_4, z_5\}$ separating z_1 from the rest of the graph.

of the configurations of $\mathbf{Z}$ and due to the fact that we are assuming just two Gaussian components. Moreover having defined local conditional probabilities in (4.76), a Gibbs sampler volunteers for the generation of labeling configurations according to (4.76) which is defined by the current estimates of the parameters γ. γ', α and $\tilde{\alpha}$. To get more precise about the underlying problem of image labelling, the graph structure of a three-dimensional PET image has to be exploited in more detail. For simplicity, the edge set is assumed to connect only neighbouring voxels having a face in common[3]. The corresponding graph for a two-dimensional slice of the PET data of size 5×5 is shown in figure 4.17. As can be seen from the left picture in figure 4.17 the subset $\{Z_2, Z_3, Z_4, Z_5\}$ is a separator set which renders Z_1 independent from the rest of the image voxels. So obeying the neighbourhood of Z_1, its conditional probability can be calculated using (4.76). Hence a Gibbs sampler can be employed to update a voxel state by calculating the probabilities for the various labellings of the voxel and decide according the rules given by (3.74).

To update all vertices/voxels of the image, various procedures have been proposed. Updating each label sequentially by looping over all voxels of the entire image is a time wasting strategy. Another approach would be to partition the voxels according to figure 4.18 (b) into two disjoint subsets $\mathcal{V}_g$ and $\mathcal{V}_r$ ($\mathcal{V}_g \cup \mathcal{V}_r = \mathcal{V}$, $\mathcal{V}_g \cap \mathcal{V}_r = \varnothing$). With this partitioning the graph is 2-partite or bipartite. This means that the two subsets, $\mathcal{V}_g$ and $\mathcal{V}_r$, partition the graph so that voxels in the same subset are never connected by an edge. Hence, having observed one subset renders each voxel of the other subset independent from the rest of the graph. In this case one can proceed to update all voxels in a subset

[3]In three dimensions this yields a 6-neighbourhood.

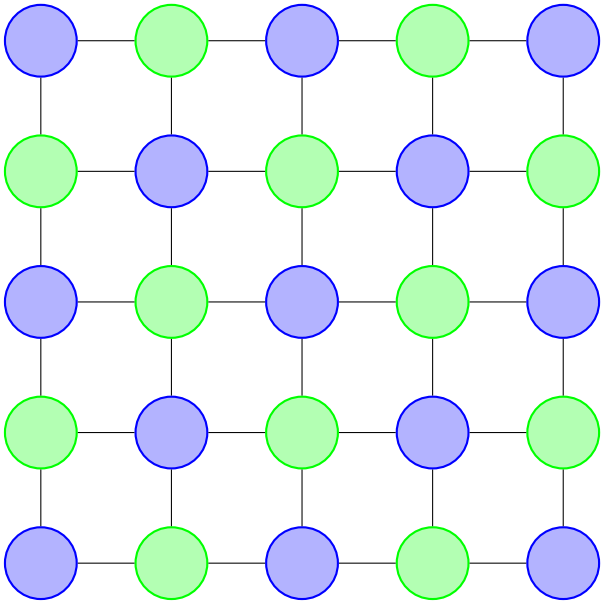


Figure 4.18: Two step sampling allocation.

in common, alternating between the subsets. This procedure even works if a copy of the current state of the labels are stored and the updates are done for both subsets in parallel.

A graph as shown in figure 4.18 (b) leads naturally to a two-stage update procedure. By denoting the variables as $\mathbf{Z}_g \in \mathcal{V}_g$ and $\mathbf{Z}_r \in \mathcal{V}_r$, the labeling problem (4.6) can be approximated using the conditional probability (4.76) according to

$$\arg\max_{\mathbf{Z}_g}\{p(X \mid \mathbf{Z}_g)p(\mathbf{Z}_g \mid \mathbf{Z}_r)\}, \quad \arg\max_{\mathbf{Z}_r}\{p(X \mid \mathbf{Z}_r)p(\mathbf{Z}_r \mid \mathbf{Z}_g)\}. \tag{4.90}$$

In this study, due to large datasets X a fast strategy is pursued. Therefore the updates are done for both subsets ($\mathbf{Z}_g$ and $\mathbf{Z}_r$) of labels in parallel. To calculate an update for each label z_n, the Metropolis algorithm from (3.74) is used. The distribution we draw samples from is the conditional distribution (4.76). The following update procedure can easily be implemented using matrix operations:

- Initialize $\mathbf{Z}$ by using an EMGMM procedure.
- Store the current label matrix $\mathbf{Z}$ and generate a new one, $\mathbf{Z}^{new}$, by flipping all states of the binary matrix.
- For all voxels, calculate
$$a_n = \frac{p(x_n \mid z_n^{new})p(z_n^{new} \mid \mathcal{N}(z_n))}{p(x_n \mid z_n)p(z_n \mid, \mathcal{N}(z_n))} \tag{4.91}$$
- For each voxel, sample a uniformly distributed random variable $q_n \in [0, 1]$. If $q_n < a_n$, accept the new value for $\mathbf{Z}$, otherwise reject it.

After reaching labeling configurations at equilibrium, meaning that the mean energy of each labelling configuration is equal on average, the labeling configurations are stored to calculate the empirical expectations of $\mathbf{Z}$ according to (3.5), i.e.,

$$\hat{\mathbf{Z}} = \frac{1}{J}\sum_{j=1}^{J}\mathbf{Z}^{(j)}. \tag{4.92}$$

As can be seen from (5.31), the partition function $A(\alpha, \tilde{\alpha})$ cancels in the numerator and denominator and therefore needs not to be calculated.

Formulating an iterative algorithm for a GMRF, the e-step can be supplemented by a Metropolis sampler which of coarse can be used with any graph structure.

4.4.3 Parameter Estimation

The algorithms employed till now have in common that their parameter estimation steps were given by closed form solutions. This was either performed by calculating the ML estimator via the derivative of the likelihood function or by calculating the MMSE estimator via a coefficient comparison. The parameters in use were given by the two Gaussian parameters μ and σ and in case of EM procedures additionally by the parameter of the generalized Bernouli distribution τ (each parameter occurs k times with k being the number of clusters). For models incorporating local interactions among voxels the parameter set was moreover augmented with the correlation coefficients ν.

Applying the GMRF presented in (4.75) we have to deal with the Gaussian parameters γ and γ' as well as with the Potts model parameters [4] α and $\tilde{\alpha}$. Since the calculation of an optimal Θ based on $p(\mathbf{Z}, X \mid \Theta)$ is equivalent to calculate the optimal Θ of $\ln p(\mathbf{Z}, X \mid \Theta)$, the problem of estimating the Gaussian parameters can be decoupled from the problem of estimating α and $\tilde{\alpha}$, see (4.75). As mentioned in section 4.4.1, γ and γ' are related to the Gaussian mean μ and the standard deviation σ via (4.79). These are again closed form solutions which can be calculated easily by calculating the sample mean and the sample standard deviation according to (4.38) and (4.39) and using (4.79). So we are left with the problem of estimating α and $\tilde{\alpha}$.

As the model shown in (4.75) does not include prior probabilities for the parameters, the MMSE would be hard to accomplish. But calculating the derivative of the likelihood function and therewith the ML estimator just demands that the likelihood function is differentiable. As can be seen from (4.75) and as is stated in section 4.4.1, this constraint is fulfilled. But the partition function is convex as also mentioned in section 4.4.1. In case of $k = 2$ the partition function is shown in figure 4.4.3. Calculating the derivative of (4.75) and trying to express the parameters fails. Thus one has to resort to numerical methods, i.e, convex optimization algorithms (see appendix D).

The ML problem for the two Potts model parameters α and $\tilde{\alpha}$ reads

$$\arg\max_{\alpha_k}\left\{\alpha_k \sum_{n\in\mathcal{V}}^{N} z_{nk} - A(\alpha, \tilde{\alpha})\right\} \tag{4.93}$$

[4] The term in (4.75) governed by the parameter α can be considered as counterpart to the prior term included during EM procedures as suggested by the underbrace in (4.75).

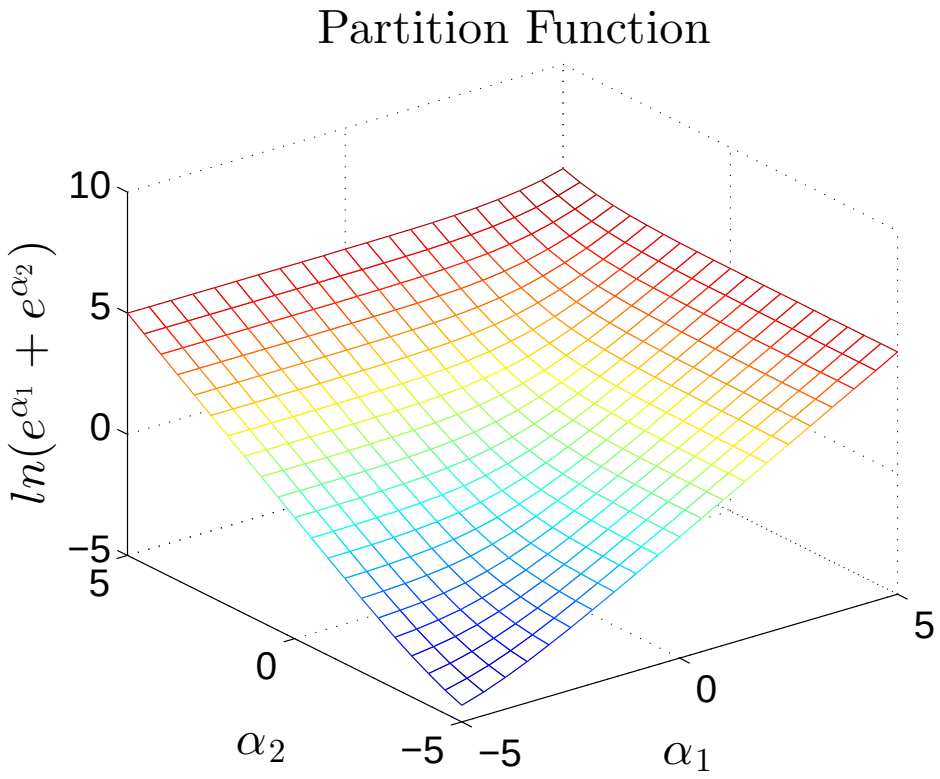


Figure 4.19: Graph of the partition function $A(\alpha_k)$ defined in (4.4.3.1) for the case of $k = 2$.

and

$$\arg\max_{\tilde{\alpha}_{kl}} \Big\{ \tilde{\alpha}_{kl} \sum_{\{n,m\}\epsilon\mathcal{E}} z_{nk} z_{mk} - A(\alpha, \tilde{\alpha}) \Big\}, \tag{4.94}$$

with $\mathcal{E}$ being the set of edges connecting the various labels of the label matrix $\mathbf{Z}$. Differentiating the equations in curly parentheses in (4.93) and (4.94) regarding the parameters α_k respectively $\tilde{\alpha}_{kl}$ yields

$$\sum_{n\epsilon\mathcal{V}} z_n = \mathbb{E}_{\alpha,\tilde{\alpha}} \Big\{ \sum_{n\epsilon\mathcal{V}} z_n \Big\} \triangleq \partial_\alpha A(\alpha, \tilde{\alpha}), \tag{4.95}$$

$$\sum_{(n,m)\epsilon\mathcal{E}} z_n z_m = \mathbb{E}_{\alpha,\tilde{\alpha}} \Big\{ \sum_{(n,m)\epsilon\mathcal{E}} z_n z_m \Big\} \triangleq \partial_{\tilde{\alpha}} A(\alpha, \tilde{\alpha}), \tag{4.96}$$

which reduces to the moment matching conditions shown in (3.71). Specifically, moving the summation outside of the expectation yields

$$\hat{\mu}_\alpha \triangleq \frac{1}{|\mathcal{V}|} \sum_{n\epsilon\mathcal{V}} z_n = \mathbb{E}_{\alpha,\tilde{\alpha}}\{z_n\}, \tag{4.97}$$

$$\hat{\mu}_{\tilde{\alpha}} \triangleq \frac{1}{|\mathcal{E}|} \sum_{(n,m)\epsilon\mathcal{E}} z_n z_m = \mathbb{E}_{\alpha,\tilde{\alpha}}\{z_n z_m\}, \tag{4.98}$$

whereby the expectation $\mathbb{E}_{\alpha,\tilde{\alpha}}$ have to be calculated using $p_{\alpha,\tilde{\alpha}}(\mathbf{Z})$ from (4.75).

Seeking a solution of this problem is equivalent to finding the probability distribution with maximum entropy, see section 3.4.5. Technically this requires to adjust the parameters α and $\tilde{\alpha}$ so that

the corresponding expectations under the probability distribution $p_{\alpha,\tilde{\alpha}}(\mathbf{Z})$ equal the empirical expectations $\hat{\mu}_\alpha$ and $\hat{\mu}_{\tilde{\alpha}}$, see (4.97) and (4.98). Because the partition function $A(\alpha,\tilde{\alpha})$ is a convex function, $p_{\alpha,\tilde{\alpha}}(\mathbf{Z})$ is concave and so this can be achieved using iterative methods described in the following two sections.

4.4.3.1 Local Estimation - Mean Field

It is recognized that the partition function $A(\alpha,\tilde{\alpha})$ depends on both parameters, α and $\tilde{\alpha}$. So optimizing the logarithm of the likelihood functions (4.93) and (4.94) regarding α respectively $\tilde{\alpha}$, we obtain solutions which comprise both the parameter α and the parameter $\tilde{\alpha}$. This would demand to update the parameters simultaneously (alternating the update steps for both parameters). Although this is possible, some further simplification imposes a factorization property to the partition function $A(\alpha,\tilde{\alpha})$, i.e., $A(\alpha,\tilde{\alpha}) = A(\alpha)A(\tilde{\alpha})$. This trick is known from statistical physics where it is called a mean field approach. Hence the moment matching conditions (4.97) and (4.98) get

$$\hat{\mu}_\alpha = \mathbb{E}_\alpha\{z_n\} \triangleq \partial_\alpha A(\alpha), \tag{4.99}$$

$$\hat{\mu}_{\tilde{\alpha}} = \mathbb{E}_{\tilde{\alpha}}\{z_n z_m\} \triangleq \partial_{\tilde{\alpha}} A(\tilde{\alpha}). \tag{4.100}$$

This implies that the marginal probability distribution for a single voxel can be written as

$$p_\alpha(z_n) = \exp\Big\{z_{nk}\alpha_k - A(\alpha)\Big\} \tag{4.101}$$

$$A(\alpha) = \ln\sum_k \exp\Big\{z_{nk}\alpha_k\Big\}. \tag{4.102}$$

For two neighbouring voxels the joint probability distribution gets

$$p_{\tilde{\alpha}}(z_n, z_m) = \exp\Big\{z_{nk}\tilde{\alpha}_{kl}z_{ml} - A(\tilde{\alpha})\Big\} \tag{4.103}$$

$$A(\tilde{\alpha}) = \ln\sum_k\sum_l \exp\Big\{z_{nk}\tilde{\alpha}_{kl}z_{ml}\Big\}. \tag{4.104}$$

The summation constraints $\sum_k p_\alpha(z_n) = 1$ and $\sum_k\sum_l p_{\tilde{\alpha}}(z_n, z_m) = 1$ are implicitly satisfied. Note that and are defined just locally.

Having defined local probability distributions we attempt to approximate the estimators for α and $\tilde{\alpha}$ by optimizing these distributions. Hence (4.93) and (4.94) are reformulated according to

$$\arg\max_{\alpha_k} \Big\{\alpha_k z_{nk} - A(\alpha,\tilde{\alpha})\Big\} \tag{4.105}$$

$$\arg\max_{\tilde{\alpha}_{kl}} \Big\{\tilde{\alpha}_{kl}z_{nk}z_{mk} - A(\tilde{\alpha})\Big\}. \tag{4.106}$$

For $A(\alpha)$ and $A(\tilde{\alpha})$ are convex functions there will be no analytical solutions to these problems which are convex. Therefore this unconstrained program in concave form [3] has to be solved by iterative methods (see appendix D). Since there are no side constraints, one can resort to a simple gradient ascent[5] method [3]

$$\theta^{(i+1)} = \theta^{(i)} + t^{(i)}\nabla[\ln p_{\theta^{(i)}}], \tag{4.107}$$

[5] In case of convex functions a descent method would be employed.

with i denoting the iteration number, $t^{(i)}$ the step size and $\nabla[\ln p_{\theta^{(i)}}]$ the search direction given by the gradient of the objective function. With $\theta \triangleq (\alpha, \tilde{\alpha})$ the gradient terms in (4.107) for the considered problem are given by

$$\nabla_{\alpha_k}[\ln p_{\alpha_k}] = \hat{\mu}_{\alpha_k} - \mathbb{E}_{\alpha_k}\{z_{nk}\} \tag{4.108}$$

$$\nabla_{\tilde{\alpha}_{kl}}[\ln p_{\tilde{\alpha}_{kl}}] = \hat{\mu}_{\tilde{\alpha}_{kl}} - \mathbb{E}_{\tilde{\alpha}_{kl}}\{z_{nk}z_{ml}\} \tag{4.109}$$

Thus a backtracking line search (cf. section D.2) is applicable:

- Count the empirical expectation of the sufficient statistics (4.77) from the current labeling and choose an appropriate initial estimate for the according parameter θ.
- With $a \in (0, 0.5)$ and $b \in (0, 1)$, employ a backtracking line search [3] and iterate the following two steps

  while: $\ln p(\theta + t\nabla \ln p(\theta)) < \ln p(\theta) + at|\nabla \ln p(\theta)|^2$:

  - calculate $t = bt$;
  - update θ according to (4.107) using (4.108) and (4.109);
- Terminate if $|\nabla[\ln p_\theta]| < \epsilon$, otherwise go to step one.

Note that GMRF accounts for interactions among voxels having opposite labels. This enables us to influence the membership probabilities of voxels at the border of different objects.

4.4.3.2 Local Estimation - Pseudo Likelihood

Having established a mean field approach as in section 4.4.3.1, the interaction parameters for different tissues $\tilde{\alpha}_{kl}$ with $k \neq l$ are subject to the constraint

$$\tilde{\alpha}_{kl} = \tilde{\alpha}_{lk} \qquad \forall k \neq l. \tag{4.110}$$

This follows from optimizing local joint probabilities for pair voxels (see (4.103)) having a symmetric matrix $\tilde{\alpha}_{kl}$. With this, the probability of labeling a voxel as cancerous tissue in case of a neighbouring voxel being healthy tissue is equal to the probability of labeling a voxel as healthy tissue in case of a neighbouring voxel being cancerous tissue.

As mentioned in the introduction, PET images are very noisy. Therefore problems arise if considering lesions with low uptake rates in contrast to the surrounding tissue (low SBR). Cancerous tissue in liver is an example where this happens. Moreover small objects are often hard to detect due to PVE (spreading activity to their neighbourhood). To enhance the detectability of objects having low contrast it is desireable to enhance the probability of voxels being part of cancerous tissue.

First of all, to break the constraint (4.110) so that in general

$$\tilde{\alpha}_{kl} \neq \tilde{\alpha}_{lk} \qquad \forall k \neq l, \tag{4.111}$$

we propose a slightly different approach by using local conditional probability distributions (4.76) rather then joint distributions (4.103). As stated in [31], the joint distribution for the entire voxels (4.75) can be approximated by multiplying the local conditionals (4.76), i.e.,

$$p(X, \mathbf{Z}) \approx \prod_{n \in \mathcal{V}} p(x_n \mid z_n) p(z_n \mid \mathcal{N}(z_n)) \tag{4.112}$$

which is termed a pseudo likelihood function [31]. To get separated problems for the parameters α and $\tilde{\alpha}$ (see section 4.4.3.1), the partition function again is subject to the factorization $A(\alpha, \tilde{\alpha}) = A(\alpha)A(\tilde{\alpha})$. Hence the estimation of the parameter α is performed as before. To derive an update procedure for the parameter $\tilde{\alpha}$ we start from the approximation of the overall joint probability (4.112). Thus the ML problem for the model parameter $\tilde{\alpha}$ reads

$$\arg\max_{\tilde{\alpha}_k} \Big\{ \tilde{\alpha}_{kl} \sum_{n \in \mathcal{V}} \sum_{m \in \mathcal{N}(z_n)} z_{nk} z_{mk} - \sum_{n \in \mathcal{V}} A_n(\tilde{\alpha}) \Big\}. \tag{4.113}$$

Note the difference to the optimization problem (4.94). In (4.113) we split the summation over the edge set $\mathcal{E}$ into a summation over each voxel and into a summation over their neighbourhood. Solving for the parameters as done in section 4.4.3 yields

$$\sum_{n \in \mathcal{V}} \sum_{m \in \mathcal{N}(z_n)} z_n z_m = \sum_{n \in \mathcal{V}} \mathbb{E}_{\tilde{\alpha}} \Big\{ \sum_{m \in \mathcal{N}(z_n)} z_n z_m \mid z_m \Big\} \tag{4.114}$$

which again reduces to the moment matching conditions (cf. (3.71), (4.97) and (4.98))

$$\hat{\mu}_{\tilde{\alpha}} \triangleq \frac{1}{|\mathcal{V}||\mathcal{N}(z_n)|} \sum_{n \in \mathcal{V}} \sum_{m \in \mathcal{N}(z_n)} z_n z_m = \mathbb{E}_{\tilde{\alpha}} \{ z_n \mid z_m \}. \tag{4.115}$$

Hence the expectation has to be calculated using the conditional probability $p(z_n \mid z_m)$

$$p(z_{nk} \mid z_{ml}) = \exp \Big\{ z_{nk} \tilde{\alpha}_{kl} z_{ml} - A_n(\tilde{\alpha}) \Big\}, \tag{4.116}$$

with z_{ml} assumed given and with partition function including just the summation over the labellings of the voxel z_{nk} as

$$A_n(\tilde{\alpha}) = \ln \sum_k \exp \Big\{ z_{nk} \tilde{\alpha}_{kl} z_{ml} \Big\}. \tag{4.117}$$

Because conditional distributions are considered, in fact (4.116) can be viewed as describing two distinct functions $p(z_{nk} \mid z_{m0})$ and $p(z_{nk} \mid z_{m1})$ with their own parameter vector $\tilde{\alpha}_{k0}$ and $\tilde{\alpha}_{k1}$ and their own summation constraints according to

$$\sum_k p(z_{nk} \mid z_{ml}) = 1, \qquad l \in \{0, 1\}. \tag{4.118}$$

This was our actual goal as discussed in the beginning of this subsection.

With two objects to be detected ($k = 1, 2$) an iterative procedure can be formulated as in section 4.4.3.1. In this case the two objective functions are given by $p(z_{nk} \mid z_{m0}; \tilde{\alpha}_{k0})$ and $p(z_{nk} \mid z_{m1}; \tilde{\alpha}_{k1})$ whereby the parameters $\tilde{\alpha}_{k0}$ and $\tilde{\alpha}_{k1}$ are two-dimensional vectors in contrast to the predefined case where $\tilde{\alpha}$ was a 2×2 matrix. Hence the counterpart to the gradient equation (4.109) is

$$\nabla_{\tilde{\alpha}_{k0}} [\ln p_{\tilde{\alpha}_{k0}}] \quad = \quad \hat{\mu}_{\tilde{\alpha}_{k0}} - \mathbb{E}_{\tilde{\alpha}} \{ z_{nk} \mid z_{m,0} \} \tag{4.119}$$

$$\nabla_{\tilde{\alpha}_{k1}}[\ln p_{\tilde{\alpha}_{k1}}] \quad = \quad \hat{\mu}_{\tilde{\alpha}_{k1}} - \mathbb{E}_{\tilde{\alpha}}\{z_{nk} \mid z_{m,1}\} \tag{4.120}$$

With these expressions the backtracking line search of section D.2 is applicable, assuring that the empirical expectations follow the summation constraints $\sum_k \hat{\mu}_{\tilde{\alpha}_{k0}} = 1$ and $\sum_k \hat{\mu}_{\tilde{\alpha}_{k1}} = 1$.

5

Results

A common problem with verifying segmentation procedures for human PET scans is that the ground truth is never known. Even in cases where patients have underwent a surgery it is hard to determine the exact size and location of cancerous tissue in three-dimensional images. Moreover some removed tissue probably contains healthy cells, as well as cancerous cells will maybe left inside diseased humans. It is therefore necessary to acquire PET images of some human equivalents where the activity levels and the geometry are given. Various phantoms have already been established for the sake of quality assurance in radio therapy, e.g. for determining the NECR curves shown in section 2.2.1.

The standards was released by the National Electrical Manufacturers Association (NEMA) which further control the DICOM standard (Digital Imaging and Communications in Medicine). The DICOM standard regulates the storage and exchange of medical image data. A DICOM dataset consists of a data block storing the image data (based on the TIFF and JPEG norm) and a block of meta data. The meta data comprise data fields regarding the patient or the study under concern as well as information about the device and the reconstruction method in use (devices and software have to fulfill the DICOM Conformance Statement). Moreover, parameters which are needed for post processing the images are given, e.g., a scale s and an intercept i to be applied to the stored image data X^{stored} as

$$x_n^{\text{real}} = i + s x_n^{\text{stored}}, \tag{5.1}$$

with n tagging a discrete entity of the three-dimensional image. To locate each discrete entity of the images, a three-dimensional coordinate system is spanned by means of an origin $[x, y, z]$ and a voxel spacing [dx,dy,dz] which is also given to the DICOM file header. Based on this geometrical information, coregistered PET/CT scans can be overlayed. In case of PET images informations on decay corrections, which have to be applied before analysis, are stored among the meta data.

In this chapter we present the measurement process, including the PET scanner and the NEMA phantom in use, as well as the application of the segmentation algorithms presented in chapter 4.

Section 5.1.2.1 summarize statistical properties of the image data which will help to understand and advance the clustering techniques. In section 5.2 we introduce some meaningful constraints and definitions to capture the algorithmic outcomes without much notational overhead using graphs.

5.1 PET Measurement

5.1.1 PET Scanner

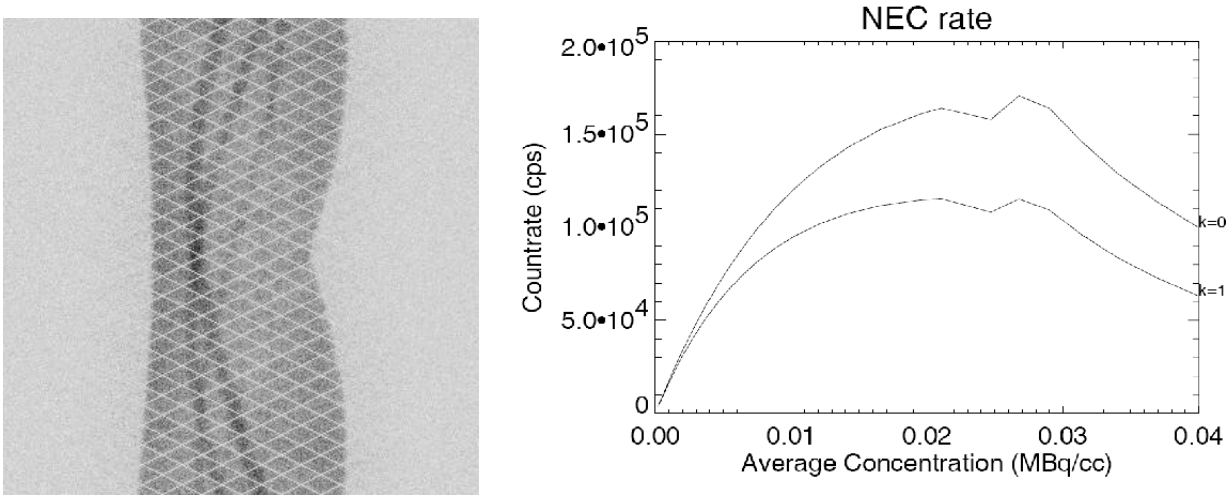
To evaluate the proposed algorithms, we used measurements taken with the PET device at General Hospital of Vienna. The scanner in use is a Siemens Biograph True Point 64 slice PET/CT scanner which in fact is having a maximum of 52 rings (with so called "TrueV option" - see below) leading to 103 image planes. The Lutetium Oxyorthosilicate detector crystals with a fast scintillation decay of 40ns offers a coincidence window of 4.5 ns.

specifications	no TrueV	TrueV
Crystal Material	LSO	-
Crystal Dimensions	4×4×20mm	-
Crystals per Block	169 (13×13)	-
PMT's per Block	4	-
Blocks	144	192
Rings	39	52
Crystals per Ring	624	-
Ring Diameter	842mm	-
Transaxial FOV	605mm	-
Axial FOV	162mm	216mm
Image Planes	81	109
Plane Spacing	2mm	-
Coincidence Win.	4.5ns	-
Count Rate Peak	96kcps@35kBq/cc	165kcps@32kBq/cc

Table 5.1: Specifications of the Siemens Biograph True Point 64 slice PET scanner as listed by Siemens.

This detector crystals of 4×4×20mm are organized in quadratic detector blocks with length 13×13 crystals and repeated cylindrically, constituting an entity with 13 rings and 48 detector blocks. For each detector block 4 Photomultiplier Tubes (PMT) are responsible (see section 2.2 for detailed description of PET systems).

This scanner can be used in two different modes, differing in the amount of axial repetitions of the afore mentioned cylindrical entity which is 3 in normal mode and 4 in TrueV mode. Hence in TrueV mode 52 rings are available. As can be seen from table 5.1, which shows the scanner specifications as given by Siemens documentations, the number of image planes do not follow up from the amount of rings. The amount of image planes, calculated from the amount of rings in table 5.1, should be 77 or 103 with TrueV option. The reason is that the gaps between the detector blocks are treated as additional crystals. Hence the scanner behaves as having detector blocks of 14×14 crystals which



(a) (b)

Figure 5.1: (a) Sinogram of a NEMA phantom (section 5.1.2) measurement yielded from the Siemens Biograph True Point 64 slice scanner in TrueV option with 52 rings, a span of 11 (axial compression) and a maximum ring difference (RD) of 38. For Siemens treats the gaps between detector block as additional crystals, the number of rings increases to 55 crystals which leads to 109 image planes. (b) NECR of the Siemens Biograph True Point 64 scanner at measurement time with k=0 tags the curve with TrueV option.

leads to the number of image planes mentioned in the Siemens specifications. This fact becomes visible when considering sinogram data of the scanner, see figure 5.1(a). The bright diagonal lines crossing this image arise due to missing detector crystals.

A routine inspection measurement shows the NECR (see section 2.2.1) of the detector at measurement time, figure 5.1(b), where $k = 0$ tags the curve with TrueV option. Beneath a value of 10[kBq/ml] the counting rate is linearly related to the average activity concentration as mentioned in section 2.2.1. In order to do comparison on various measurements for different SBR it is therefore necessary not to pass over an average activity concentration of 10[kBq/ml] for each measurement.

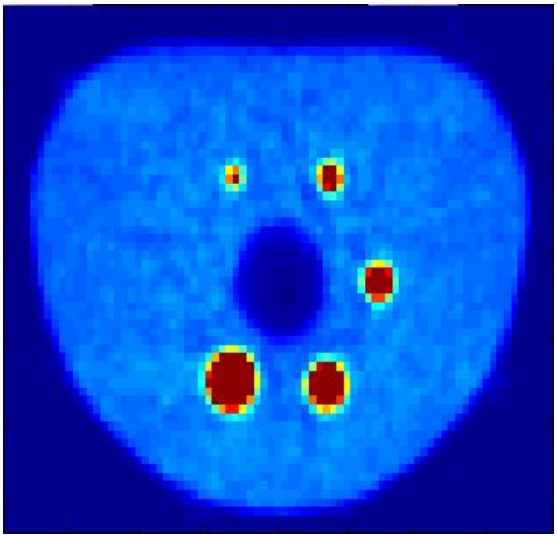
5.1.2 Phantom Measurements

To simulate cancerous tissue in humans which typically shows increased radio-tracer uptake, a NEMA IEC Body Phantom figure 5.2(a) was modified (built in-house at the Medical University of Vienna).

The original NEMA phantom, an acrylic glass construction, consists of a cylindrical outer body with a cylindrical lung insert (in figure 5.2(a) the lung insert is filled with Polystyrol) and six differently sized spheres. The spheres are attached to the outer body via capillary tubes (to fill them with radioactive solutions) having their origins at the same transversal slice.

The modified phantom differs from the original NEMA IEC Body Phantom only in the substitution of the largest sphere (37 mm) by a sphere of 8 mm diameter. Therefore the NEMA phantom in

(a)



(b)

Figure 5.2: (a) Original NEMA IEC Body phantom with six spheres and a lung insert filled with Polystyrol. (b) Transversal slice of modified NEMA phantom showing 5 spheres filled with a water-^{18}F-FDG solution of high activity concentration surrounded by the cylindrical outer body filled with a water-^{18}F-FDG solution of low activity concentration. Since the replaced sphere was not centered in the same transversal slice it can not be seen on this image.

use consists of a cylindrical outer body that simulates healthy tissue and six spherical inlays which represent tumor lesions with higher tracer uptake. The cylindrical body was homogeneously filled with a water-^{18}F-FDG solution of low activity concentration whereas the spherical inlays were homogeneously filled with a water-^{18}F-FDG solution of high activity concentration. The cylindrical inlay which models the lung was filled with air. Figure 5.2 (b) shows a transversal PET image slice of the modified NEMA phantom. Due to substituting the largest sphere by a sphere of 8 mm diameter but not substituting their capillary tube, it can not bee seen on this image which is taken from the transversal slice comprising the sphere centers of the original phantom.

For the remainder of this text the spherical inlays are tagged by SPH and the cylindrical outer body by CYL. The dimensions of the spheres were (diameter[mm]/volume[ml]): 8/0.27, 10/0.52, 13/1.15, 17/2.57, 22/5.58, and 28/11.49. Measurements with different SBRs have been performed and are summarized in table 5.2. The device in use was a Siemens Biograph 64 TruePoint PET/CT scanner. In accordance with the conditions for NEMA phantom quality assurance measurements in Nuclear Medicine [22] the average activity concentration should never exceed 10 kBq/ml. This way the linearity of the NECR (see section 2.2.1) is preserved and the measurements of different SBRs can be compared. The acquisition was performed using emission scans of 10 minutes. The images were reconstructed with a conventional Backprojection (BP) and an iterative Ordered Subset Expectation Maximization (OSEM2D) algorithm (4 iterations on 21 subsets). A preprocessing step was performed by a 5 mm Gaussian Filter. Dimension and volume of the voxels are 4 mm×4 mm×3 mm and 0.048 ml

respectively.

The more advanced iterative reconstruction algorithm for the Siemens scanner, TrueX (PSF), was not taken into account since recent studies recommended cautiousness with regard to its quantitative meaningfulness [26]. The chosen settings correspond to the clinical routine settings at the Medical University of Vienna. An image showing the sagittal plane of the modified NEMA phantom is shown in figure 5.2(b).

A_{SPH}	A_{CYL}	SBR
10.94	5.30	2.06
20.37	5.30	3.84
26.13	5.30	4.90
66.56	9.90	6.72
90.90	9.68	9.39

Table 5.2: Measurements of the modified NEMA phantom: activity concentration A for the spheres (1st column) and for the cylinder (2nd column) in kBq/ml. The resulting SBR is shown in the 3rd column.

5.1.2.1 Statistical Image Properties

To evaluate the image data presented in DICOM format, a graphical user interface was built up using the visualization tools of MATLAB. MATLAB is able to read the header information of DICOM files and so automatic decay correction can be implemented. The GUI is organized to depict the sagittal, coronal, and transversal plane. It permits direct readout of activity concentration per mouse click and provides a rectangle selection tool to mark and store the coordinates of any desired regions of interest (VOI) in the image. Hence statistical evaluations can be performed on multiple VOIs, including direct measurement of statistical measures or acting with classification algorithms.

In order to get a suggestion on how the various activity levels used in the NEMA phantom are reflected by the reconstructed images and what impact of the Point Spread Effects can be detected a priori, statistical pre evaluations (mean values, standard deviations, correlation coefficients) have been done. We delineated the accurate sphere volume by drawing VOIs around each sphere comprising cylinder voxels as well. In this sense each VOI consists of sphere voxels and surrounding cylinder voxels which get tagged by SPH and CYL to avoid notational overhead in figures and tables.

In assuming higher values for the sphere voxels, the basic algorithm for the pre-evaluations is searching for the accurate sphere volumes by increasing a threshold from zero upwards to do a simple cutoff. For the calculations of sphere volume statistics, the voxels with values beyond the threshold get analysed. For calculating the statistics of the surrounding cylinder volume, voxels having values beneath the threshold are considered. This way the most accurate volume in a discrete sense (no

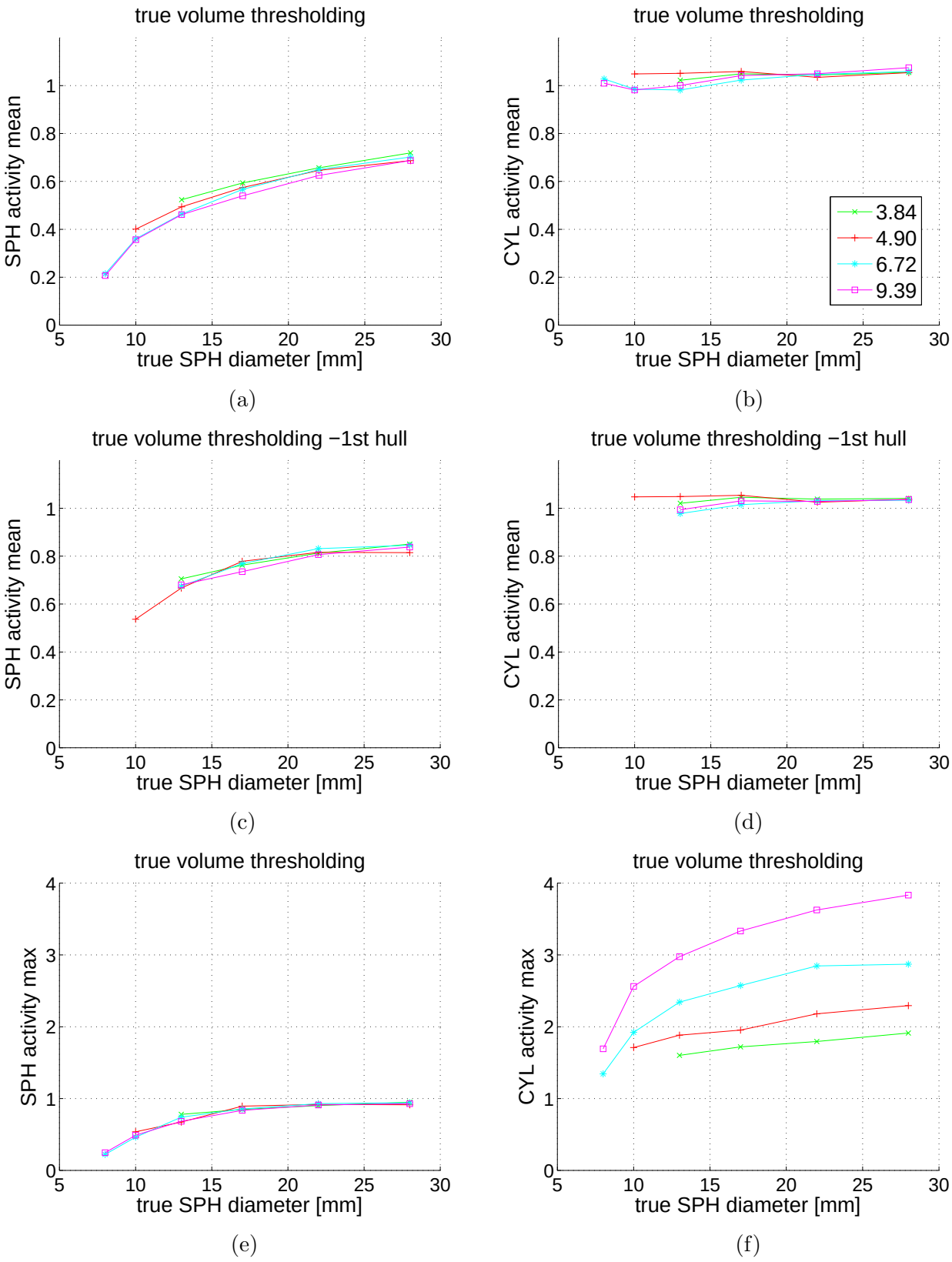


Figure 5.3: Relative mean activity concentration for (a) spheres and (b) surrounding cylinder; relative mean activity concentration for volumes with outer hull removed of (c) spheres and (d) surrounding cylinder; relative maximum activity concentration for (e) spheres and (f) surrounding cylinder. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

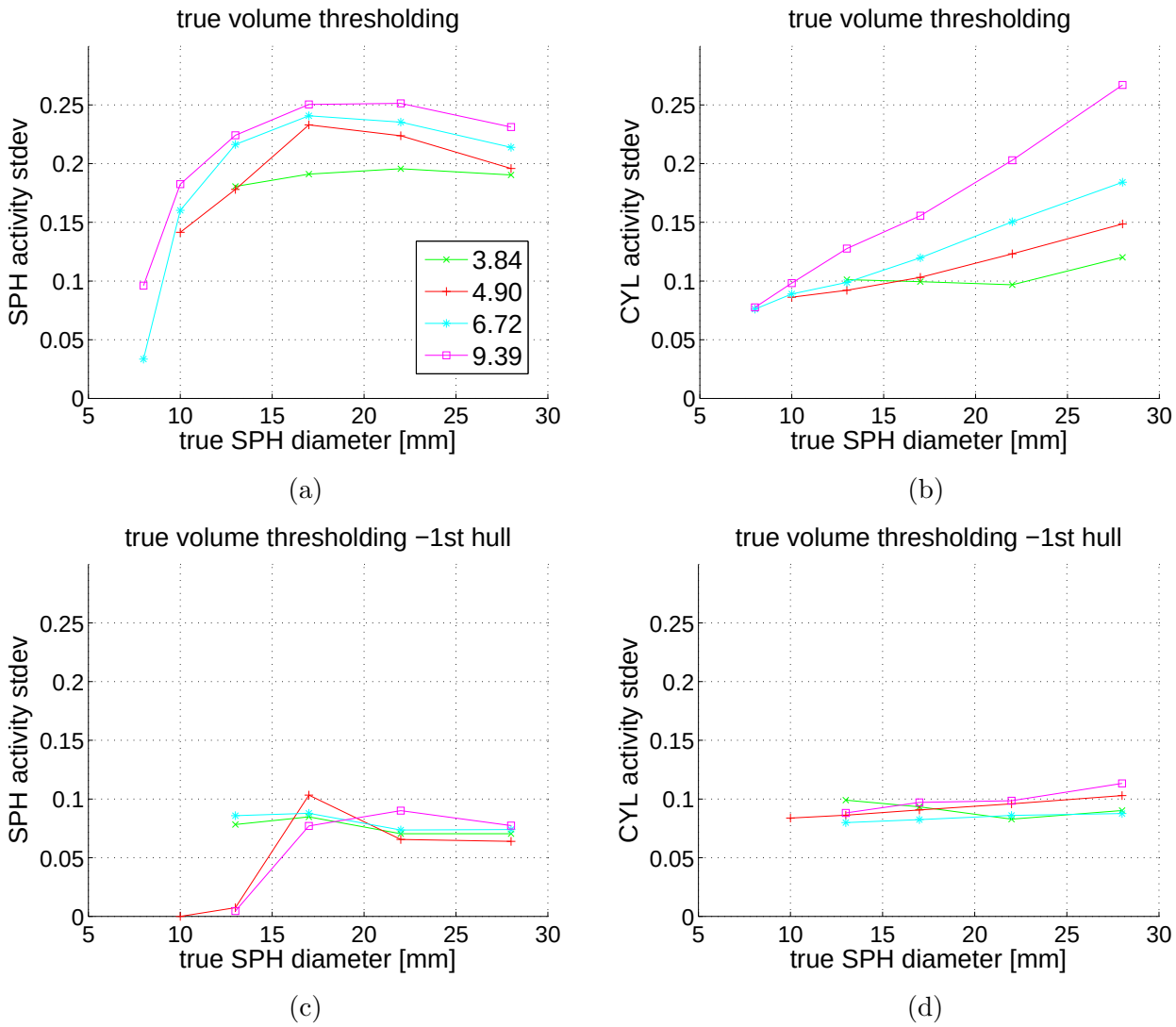


Figure 5.4: Relative standard deviation of the activity concentration for (a) spheres and (b) cylinder; relative stdev of the activity concentration for volumes with subtracted outer hull of (c) spheres (d) cylinder. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

inclusion of partial voxels) is achieved. As mentioned in chapter 1, PVE, which is the main cause of wrong image reconstruction, affects the boarder areas between spheres and surrounding cylinder volumes. To visualize this effect a second approach is established using the same threshold method but acting on the resulting volumes (sphere volumes as well as cylinder volumes) with morphological shrinking operators (erosion) to remove the voxels mainly affected by PVE from the volume borders. Hence statistical analysis is done on reduced clusters.

To attain good statistical ensembles for the cylinder volumes, the VOI's are chosen to contain a huge amount of surrounding cylinder voxels. The results of such thresholding are shown after application on the OSEM2d reconstructed PET images in figure 5.3. In figure 5.3 the relative mean activity

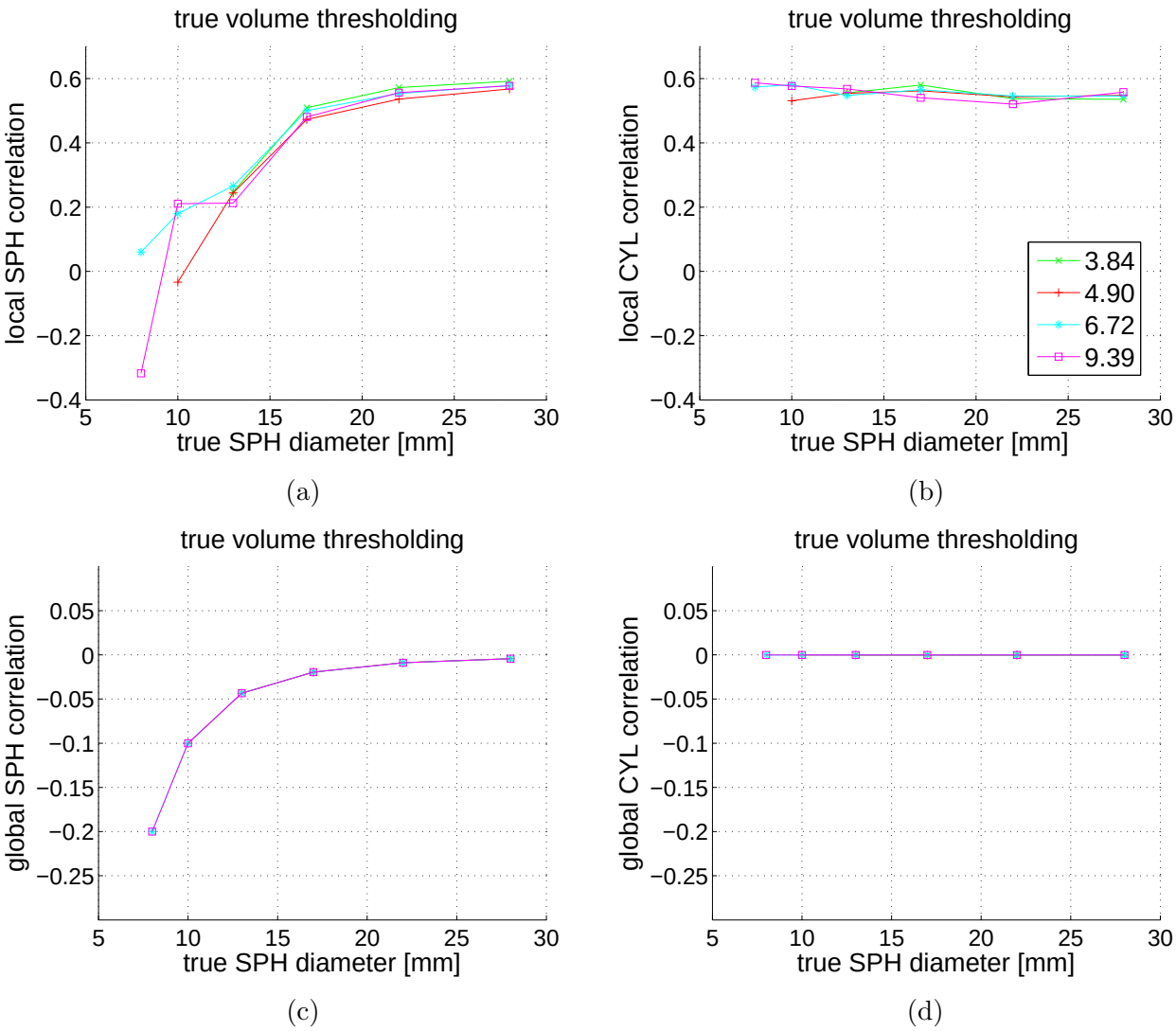


Figure 5.5: Local correlation coefficient of the activity concentration of (a) spheres and (b) cylinder; global correlation coefficient of the activity concentration of (c) spheres and (d) cylinder. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

concentration $\frac{\mu_k}{A_k}$ and the relative maximum activity concentration $\frac{\max_n\{x_n z_{nk}\}}{A_k}$, with k indexing the sphere volumes respectively the cylinder volumes, are presented. Removing the outer hull of voxels from every solution achieved with the threshold method by using morphological shrinking operators with a three-dimensional 6-neighbourhood, the two graphs in the middle of figure 5.3 are obtained. Figure 5.3 (c) and (d) shows the relative mean activity concentration of the remaining sphere volume and cylinder volumes. Every graph depicts curves for different SBR measurements connecting points for each sphere diameter for better visualization.

At first it is recognized that the mean activity concentration for the spheres in figure 5.3(a) under-

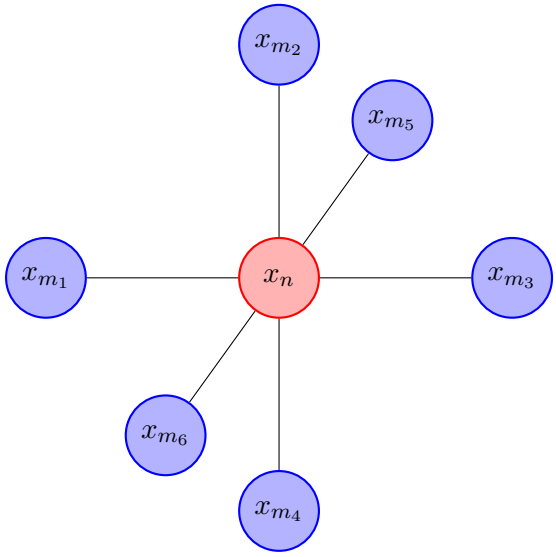


Figure 5.6: Neighbourhood $\{x_{m_1},...,x_{m_6}\}$ of a voxel x_n for the calculation of local correlation coefficients.

estimates the true activity concentration even for the largest sphere. Moreover (as direct consequence of Point Spread Effects spilling out more activity from high uptake regions to low uptake regions) the smaller the spheres get, the more we under-estimate the true activity concentration. Even after removing the outer hull of voxels from the thresholded sphere volume, figure 5.3(c), the predictability of the true value stay worse. Reliable results for the set of depicted SBRs are just achieved with maximum activity concentration values at higher sphere diameters figure 5.3(e).

Figure 5.3(b) reveals the relative mean activity concentration of the cylinder volumes comprised in the VOIs. Even after subtracting the outer hull of the thresholded cylinder voxels, the mean activity concentration for the remaining cylinder volume is stable as shown by figure 5.3(d). There again the maximum activity concentration of the surrounding cylinder volume figure 5.3(f) shows SBR-dependent overestimation of the activity concentration which is attributed to the spill in from sphere voxels showing again the effect of PVE.

Taking a look at the relative standard deviation (i.e., coefficient of variation $\frac{\sigma_k}{\mu_k}$) for the spheres (figure 5.4(a)) and the surrounding cylinder volumes (figure 5.4(b)), the standard deviation of the cylinder activity concentration shows again SBR-dependent behaviour. The larger the SBR and the sphere diameter get, the larger the standard deviation is. This is intuitive having the relative maximum activity concentration of the cylinder volumes in mind, figure 5.3(f). Due to incorporation of voxels having larger activity concentration (due to spill in from sphere voxels) to the cylinder volume, the standard deviation is raised. Reducing the thresholded volumes of the sphere and according cylinder volumes by it's outer hull and calculating again the standard deviation of the sphere volumes (figure 5.4(c)) and cylinder volumes (figure 5.4(d)), the partial volume voxels are assumed to vanish. As figure 5.4(c) and (d) show, the standard deviation in general is stabilized to a value of ~ 10% except

for small spheres.

Lastly in figure 5.5 the local and global correlation coefficients

$$\frac{\sum_{\{n,m\}\epsilon\mathcal{E}}(x_n-\mu_k)(x_m-\mu_k)z_{nk}z_{mk}}{\sum_{\{n,m\}\epsilon\mathcal{E}}z_{nk}z_{mk}}\sigma_k^{-2} \tag{5.2}$$

are depicted. The global correlation coefficients are calculated among all voxels of each cluster (sphere or cylinder)

$$\mathcal{E}=\big\{\{n,m\}|x_n\in\mathcal{V}_k\cap x_m\in\mathcal{V}_k\big\}, \tag{5.3}$$

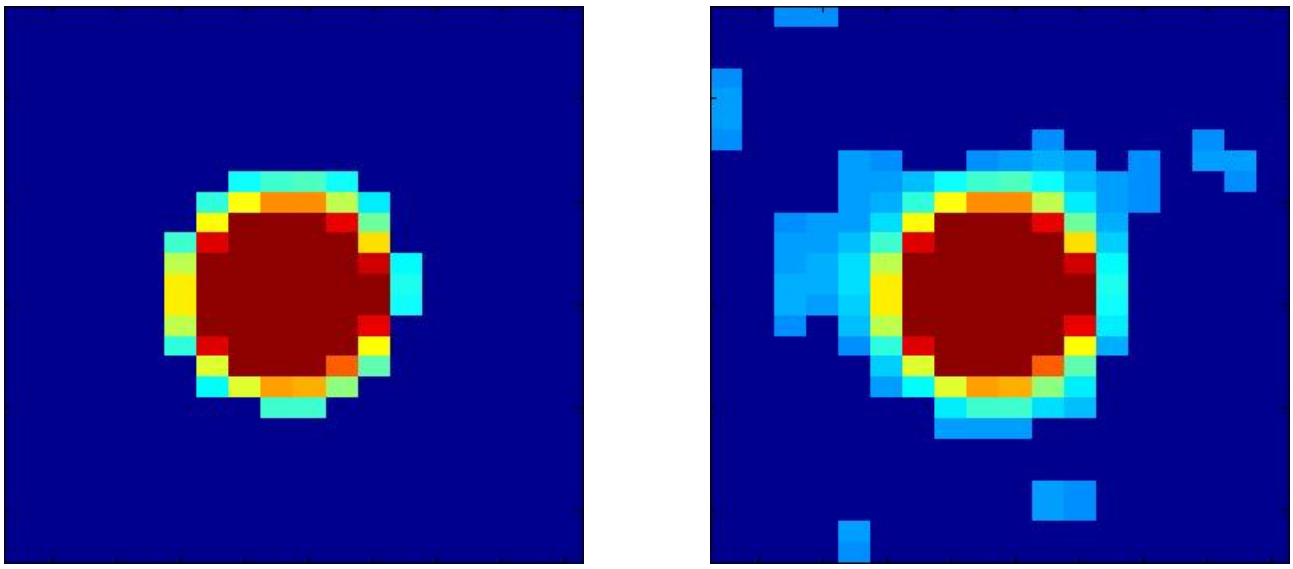
whereby the local correlation coefficients are calculated among cluster voxels which are neighbours

$$\mathcal{E}=\big\{\{n,m\}|x_n\in\mathcal{V}_k\cap x_m\in\mathcal{V}_k\cap x_m\in\mathcal{N}(x_n)\big\}. \tag{5.4}$$

The corresponding neighbourhood is shown in figure 5.6. It gets obvious, that the data is less correlated globally. Local correlations are registered for CYL and large SPH to be around a value of 0.6. As we will see in section 5.4.1, the small statistical ensembles of the spheres (specially of small spheres) leads to unreliable estimates of their statistics. To deal with this problem we will exploit the above analysis.

5.2 Analytical Definitions

To run the algorithms suggested in chapter 4, around each sphere separate VOIs are drawn containing a huge amount of cylinder voxels to yield good statistical ensembles (as done during the statistical



(a) (b)

Figure 5.7: (a) segmentation result which accounts as "detected" due to 2 connected objects are achieved, sphere and cylinder. (b) segmentation result which accounts as "not detected" due to 7 connected object are achieved.

analysis in section 5.1.2.1). In fact to reveal potential dependencies of the algorithms regarding the size of the chosen VOI, two VOIs of different volume are analysed. The small one comprises $14 \times 14 \times 20$ voxels whereas the larger one is comprising $14 \times 14 \times 40$ voxels.

As there is a huge amount of VOIs to be processed with several algorithms, it is not feasible to check on every solution regarding their meaningfulness. Therefore, spheres are considered as "detected" (only acceptable solutions), if and only if the result from segmenting a VOI yields two morphologically connected objects, sphere and surrounding cylinder. E.g. this is the case for the clustering in figure 5.7 (a) but not for the clustering in figure 5.7 (b). The only exception from this rule is labeling the entire image as sphere which is obviously wrong and counts as "not detected". Lastly the solution with every voxel being labelled as cylinder is clearly a "not detected" sphere.

This approach seems to be pessimistic. In current clinical settings, each clustering result would be controlled by a doctor which is either very well able to distinguish some outliers or, nevertheless, is responsible to go for a decision. So our practice is missing solutions which can be found maybe in clinical surroundings. A different proceeding is presented in chapter 6.

To distinguish the wrong clustering results from accurate clusterings in graphs, morphological operators are used to determine the number of objects found in the resulting PET image clusters after applying the algorithms. MATLAB offers a way to determine connected components in binary images via the function bwconncomp().

In general the results are presented showing the relative volume error. This is given by

$$\frac{\mathrm{V}_k - \mathrm{V}_{true}}{\mathrm{V}_k}, \tag{5.5}$$

which get collectively denoted as "SPH volume error". To calculate the volume of the segmentation results V_k from the label matrix $\mathbf{Z}$ we use

$$\mathrm{V}_k = \sum_n \mathbb{E}\{z_{nk}^{\text{final}}\}\mathrm{V}_{VOX}, \tag{5.6}$$

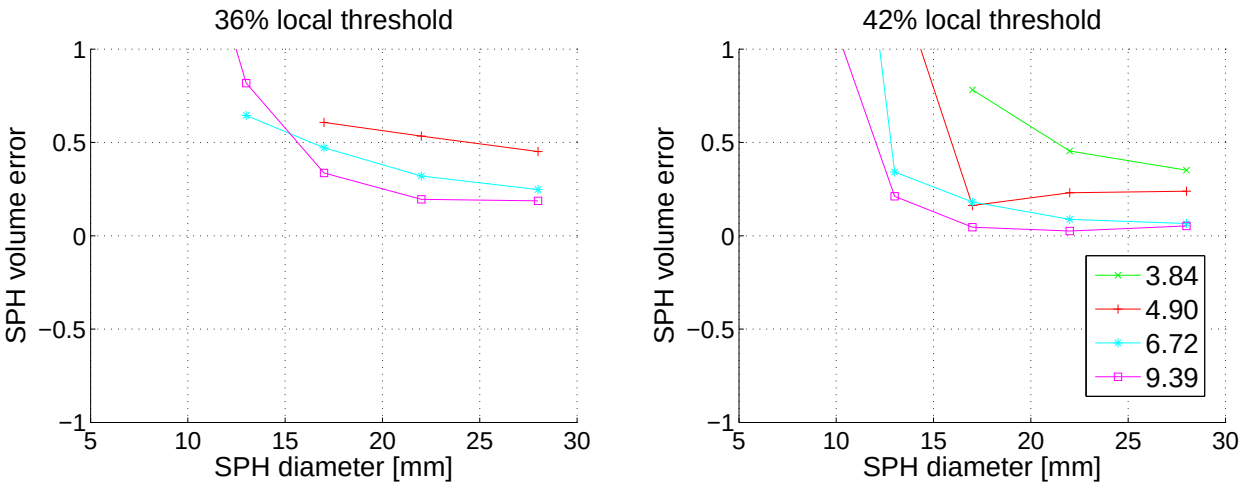
with V_{VOX} tagging the volume of a voxel.

5.3 Thresholding

As mentioned in chapter 1 the state of the art methods which get employed (if at all) for doing segmentation is thresholding. The simplest choice is local percentage thresholding where a fixed percentage of the maximum activity concentration value is used to do a simple cutoff [11, 12].

A more advanced approach is established by doing regression on the accurate threshold curve of some given phantom measurements, using specified parameters which can be read from the image, and adjusting the threshold to be applied to an image during an iterative scheme [23, 45, 46].

To show results of those methods for comparison purpose, regressions and evaluations have been performed using the phantom study discussed in section 5.1.2.



(a) (b)

Figure 5.8: SPH volume error estimated in VOIs comprising $14\times14\times20$ voxels by (a) 36% thresholding and (b) 42% thresholding. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

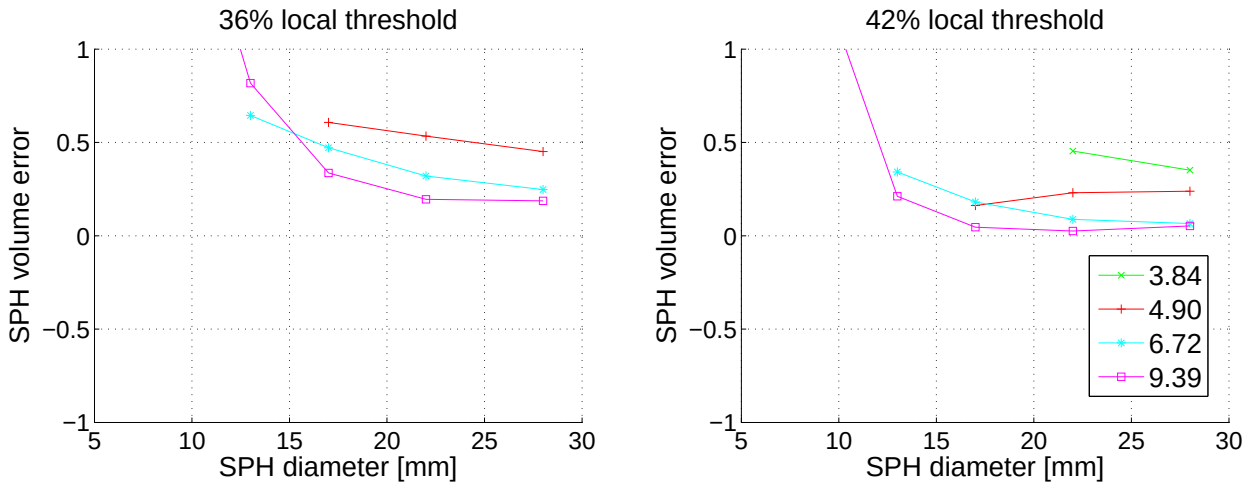
5.3.1 Percentage Thresholding

To compare the algorithms from chapter 4 regarding their volume predictability with clinical state of the art methods, a 36% and 42% local threshold method is evaluated on OSEM2D reconstructions of the NEMA phantom PET images presented in section 5.1.2. In this context, local means that the threshold is calculated as a fixed percentage of the maximum voxel value inside the VOI under consideration and not inside the whole image.

Figure 5.8 and figure 5.9 shows the relative volume estimates achieved by applying percentage thresholding to the VOIs of $14\times14\times20$ voxels respectively to the VOIs of $14\times14\times40$ voxels. Hence figure 5.8(a) and figure 5.9(a) depicts the 36% thresholding and figure 5.8(b) and figure 5.9(b) depicts the 42% thresholding. The relative volume is drawn over the sphere diameter in millimeter. For each SBR the data points are connected by straight lines for better visibility.

As can be seen, both percentage threshold methods are not able to detect the smallest sphere of 8mm diameter at any SBR. Moreover spheres measured at an SBR of 2.06 do not show up in the detection statistic table 5.3. In case of low SBR the entire image is classified as being sphere volume. From the fact that the lowest SBR is about two but the threshold is less than 50% this has to be expected.

The comparison of the VOI containing $14\times14\times20$ voxels and the VOI containing $14\times14\times40$ voxels is further showing VOI dependency of the 42% threshold method. Such huge cutoff value is prone to include outliers enhancing the possibility of misclassification in larger VOIs. This moreover confirms the noisiness of the considered PET images.



(a) (b)

Figure 5.9: SPH volume error estimated in VOIs comprising 14×14×40 voxels by (a) 36% thresholding and (b) 42% thresholding. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

Beyond the fact that spheres of diameter of 8mm are not detected, the results show that such methods are highly sensitive to SBR. The smaller the SBR gets, the larger the overestimation becomes and the more spheres are missclassified as not detected due to outliers (or even the whole VOI gets detected as sphere volume). This sensitivity is a direct consequence of PVE.

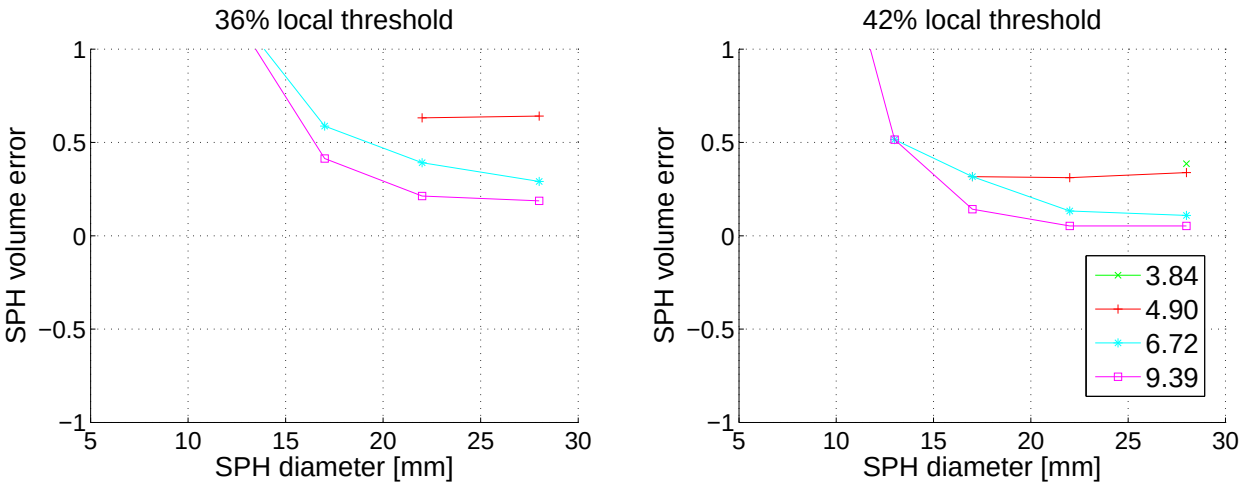
As can be seen from figure 5.8 and figure 5.9, the sphere of 10mm diameter is detected with tremendous overestimation of the sphere volume. This effect is also contributed to PVE which is the larger the smaller the sphere gets, enhancing the spill in and spill out of boarder voxels[1] as discussed in chapter 1.

Table 5.3 summarizes the number of spheres detected by the two threshold methods within the OSEM2D reconstructions of the NEMA phantom for all measured SBR. In summary of the 30 diameter/SBR configurations, the 36% threshold method correctly detects 12 spheres whereas the 42%threshold method detects 17. Both methods fail in detecting spheres of 8mm diameter and at SBRs lower than 3.84.

As mentioned in section 5.1.2 two image reconstruction algorithms, OSEM2D and BP, are used to calculate the three-dimensional images from the projection data. Although the state of the art is using OSEM2D reconstructions as they are incorporating less artefacts, the analysis of BPs is included to this work. Applying the two local percentage threshold methods to the VOIs comprising $14 \times 14 \times 20$ voxels of BP reconstructed images, figure 5.10 shows the relative volume estimates.

Comparing this results with the solutions after applying local thresholding to the OSEM2D reconstructions figure 5.8, it can be seen that this methods yield different estimates regarding low SBRs

[1] The smaller the sphere gets, the more neighbouring voxels are faced by the boarder voxels of the sphere.



(a) (b)

Figure 5.10: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by (a) 36% thresholding and (b) 42% thresholding. All data plotted versus sphere diameter for the NEMA phantom reconstructed with BP. Each graph comprises curves for various SBR, connecting measurements for each sphere.

and small diameters.

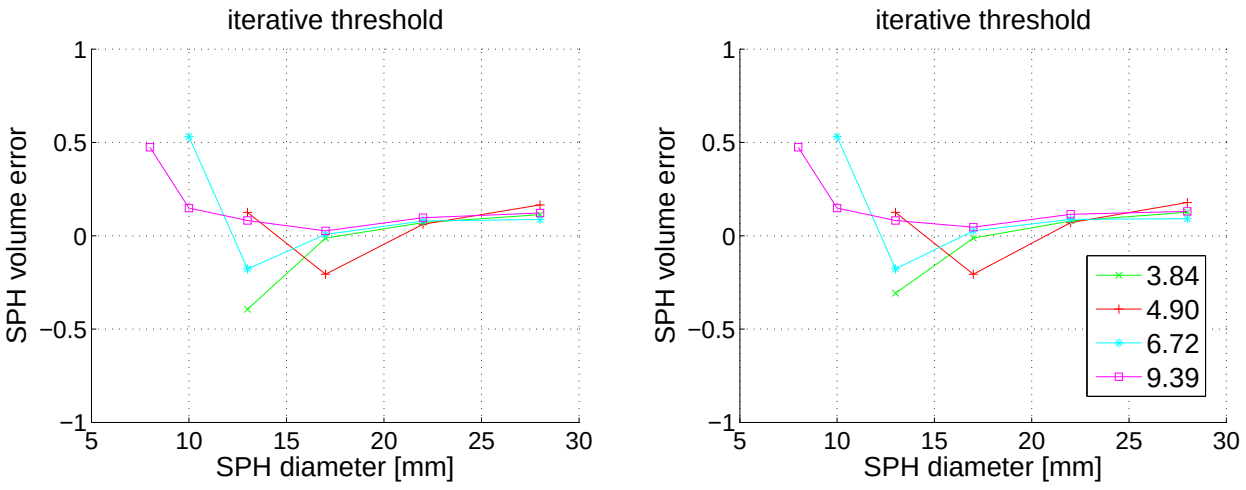
Summing up the results using BP, more spheres are lost than with OSEM2D, which is due to enhanced noisyness of these reconstructions. With the 42% thresholding method calculated on BP reconstructions, all spheres measured at SBR=2.06 are lost. In addition, all spheres but the largest one cannot be detected at SBR=3.84. As with images achieved by the statistical iterations (OSEM) no spheres of 8mm diameter can be detected.

5.3.2 Iterative Thresholding

SBR	2.06	3.84	4.9	6.72	9.39
36%	0	0	3	4	5
42%	0	3	4	5	5
iter.	0	4	4	5	6

Table 5.3: Number of spheres detected by a 36% thresholding (first row), a 42% thresholding (second row) and an iterative thresholding (third row) for the OSEM2d reconstructed images of the NEMA sphere phantom. Each column corresponds to a measured SBR. The VOI size is $14 \times 14 \times 20$ voxels.

As mentioned in the introduction of section 5.3 the results can be improved by using an iterative



(a) (b)

Figure 5.11: SPH volume error estimated in VOIs comprising (a) 14×14×20 and (b) 14×14×40 voxels by iterative thresholding. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

thresholding method (ITM). A necessary procedure to perform ITM is to generate regression curves requiring measurements of phantoms as the ones given in section 5.1.2. The percentage threshold, which clusters the true sphere volume, is calculated as function of e.g. the SBR or the sphere volume.

To perform regression, various models have been suggested, e.g. [23, 45, 46]. [23] uses the inverse sphere volume V_{SPH} and the inverse SBR as regression variables with linear parameters according to

$$\mathrm{Thr} = A_1 + \frac{A_2}{V_{\mathrm{SPH}}} + \frac{A_3}{\mathrm{SBR}}. \tag{5.7}$$

Employing this form of regression model to fit the data from section 5.1.2, the linear parameters are calculated as $A_1 = 29.917322$, $A_2 = 6.7475587$ and $A_3 = 45.357633$.

Using ITM for automatic segmentation, an initial threshold is applied to a VOI followed by the determination of the resulting volume (V_{SPH}) and SBR. With those values the threshold is updated according to (5.7) and applied to the VOI, resulting in an iterative update scheme. The algorithm stops when the deviation of the estimated volume between two iterations is $\leq 0.1\%$.

Using the regression parameters for the model in (5.7) as shown above and performing an ITM on OSEM2D reconstructed images, results are shown in figure 5.11 for VOIs of $14 \times 14 \times 20$ voxels (a) respectively for VOIs of $14 \times 14 \times 40$ voxels (b). The solutions for larger spheres are more accurate than solutions from local percentage thresholding showing the tendency for increasing volume overestimation for increasing sphere diameter. The volume overestimation of the sphere with 28mm diameter is approximately $\approx 13\%$. For the small spheres the overestimation is reduced to a maximum of $\approx 50\%$ The volume predictability is stable concerning different VOI size (disregarding the sphere of 13mm diameter at SBR=3.84).

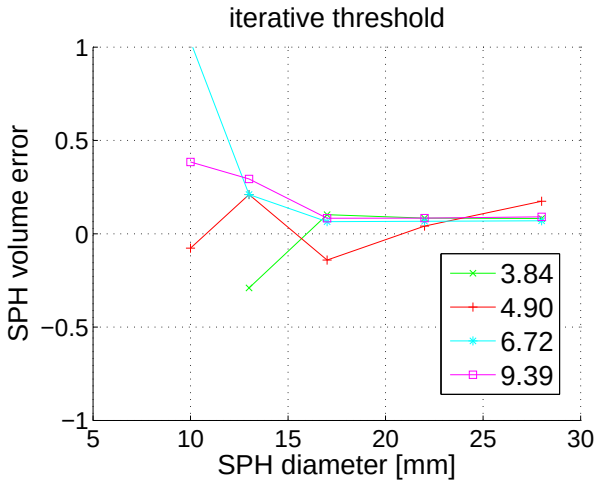


Figure 5.12: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by iterative thresholding. All data plotted versus sphere diameter for the NEMA phantom reconstructed with BP. Each graph comprises curves for various SBR, connecting measurements for each sphere.

Nevertheless, if using ITM one has to perform phantom measurement for any device where images are desired to be processed with ITM. Moreover spheres of the lowest SBR are not detected as can also be seen from table 5.3. In general the detectability of small spheres is enhanced compared to percentage thresholding. Anyway the sphere of 8mm diameter is detected just once and the one with 10mm just twice.

Trying to circumvent the effort of having to evaluate phantom measurements and calculate regression parameters to be able to perform an ITM, which anyway does not a perfect job, statistical methods are aimed to increase detectability and produce more accurate clustering results.

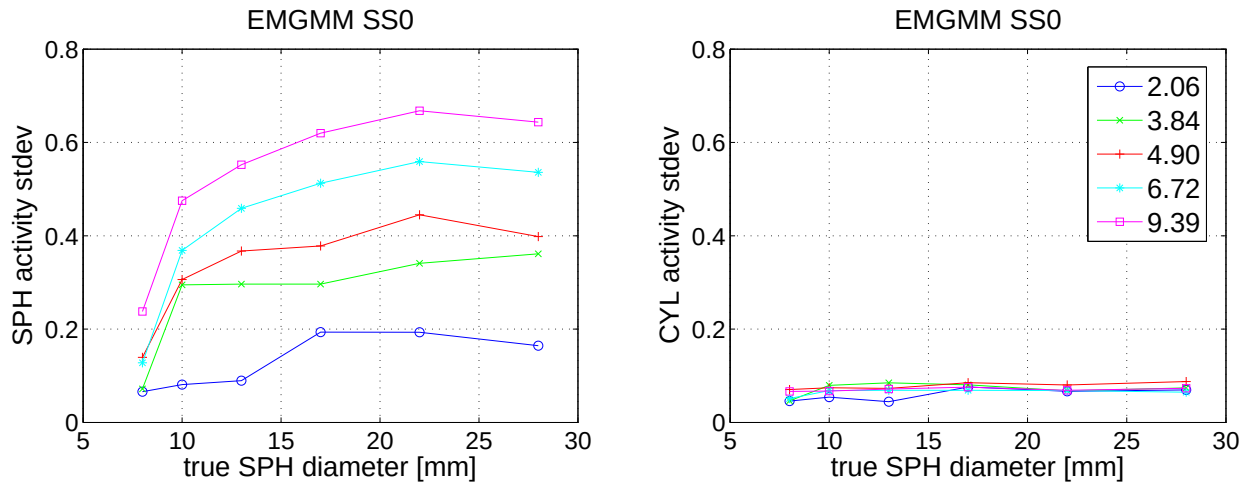
To analyse the behaviour of the iterative thresholding applied to BP reconstructed images of the NEMA phantom (using VOIs of $14 \times 14 \times 20$ voxels), figure 5.12 is obtained. Because BP anyway is not state of the art in supporting the clinical analysis, no regression was performed for the BP images. Instead the regression parameters calculated with the OSEM2D reconstructed datasets are used to show the deviations obtained in such case.

5.4 EMGMM

5.4.1 EMGMM - Two Clusters

With the definition of connectivity in section 5.2, a classical EMGMM (section 4.2) with $k = 2$ clusters is detecting just a few spheres (1 of 30 with OSEM2D) with tremendous overestimation of their volumes. Most of the solutions are including outliers and therefore would predict cancerous tissue in healthy regions.

Violating the constraint of connectivity and analysing the results in detail by plotting estimates



(a) (b)

Figure 5.13: Standard deviation of (a) the estimated sphere volumes and (b) the according cylinder volumes after applying classical EMGMM with $k = 2$ clusters. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere. These curves are drawn violating the detectability constraint made in section 5.2.

of the standard deviation for the spheres and cylinder volumes obtained by the EMGMM algorithm, figure 5.13(a) respectively figure 5.13(b) is achieved.

Contrary to the standard deviation yielded for true volume threshold estimates with and without subtracting the outer hull (figure 5.4(a)-(d)), the EMGMM algorithm causes a strong SBR diversification of the standard deviation of the sphere volumes. In case of the cylinder volume, the standard deviation results in $\sim$ 10% as it is the case for the true volume estimate with subtracted outer hull (figure 5.4(d)).

This shows, that the good statistical ensembles given by the cylinder volume leads to good parameter estimates in contradiction to the parameters of the bad sphere ensembles (as predicted by the statistical theory). With this, the PVE voxels having activity concentration values in between of those of both objects (sphere and cylinder) get included to the sphere clusters leading to an increased standard deviation for the spheres. Moreover this means, decreased estimates of the mean activity concentration.

This shows that even for the large sphere of 28mm diameter, the parameter estimates gets doubtful. A first step towards better solutions is to supplement the update steps for the sphere parameter vector $\Theta_{\text{SPH}} = (\tau_{\text{SPH}}, \mu_{\text{SPH}}, \sigma_{\text{SPH}})$, shown in (4.38)–(4.40), by assigning values extracted from the PET images. E.g., the maximum activity concentration value as naturally suggested by figure 5.3(e). In figure 5.3 it is seen that the VOI maximum would be a better estimate to the true value of the measurement, as the mean is.

5.4.2 EMGMM - Sequential Updates

As argued above, bad statistical ensembles of the spheres lead to the inclusion of partial volume voxels into the sphere volume which yields bad estimates for the mean and the standard deviation of the spheres. This distortion of sphere statistics is a direct consequence of using ML updates for the parameters which are suboptimal in case of bad statistical ensembles. The estimates of the sphere parameters should therefore be gained from other sources of the PET images or assumed to be related to information about the cylinder volume (to counterbalance increased estimate of the sphere standard deviation).

Due to increased variance of the sphere voxels, the gauss curves for the sphere volumes get very broad and therefore no significant differences are expected by just influencing the mean value μ. Inspired by section 5.1.2.1, where it has been shown (by excluding the boarder voxels from accurate segmentation results) that the relative standard deviation is about 10% (see figure 5.4(c)), an update procedure is assumed which assigns 10% of the estimated sphere mean value to the sphere standard deviation. Proceeding this way, most of the outliers obtained with classical EMGMM can be removed. Nevertheless the volume estimates are differing for the various SBR measurements also for the larger spheres.

A first modification we pay attention aims at connecting the two standard deviations, which are related to the presented activity levels and therefore to the corresponding SBR measurement, see figure 5.13. This is achieved by estimating the standard deviation for the cylinder volume as it is done with classical EMGMM (ML estimator) and assigning this value also to the standard deviation of the spheres:

$$\sigma_{\mathrm{SPH}} \longleftarrow \sigma_{\mathrm{CYL}}. \tag{5.8}$$

With this parameter assignment the algorithm evolves as follows. During an iteration where the labeling step causes overestimation of the sphere volume, the standard deviation of the cylinder volume will decrease. Assigning this value as standard deviation of the sphere clusters, their volumes in turn will be underestimated in the subsequent iteration and further the standard deviation of the cylinder will be increase.

Naming this procedure EMGMM-2, plots are shown in figure 5.14 (a) for VOIs comprising $14 \times 14 \times 20$ voxels and in figure 5.14 (b) for VOIs comprising $14 \times 14 \times 40$ voxels of OSEM2D reconstructed images.

With this parameter assignment the SBR dependence of the volume error for all sphere diameters is heavily reduced and moreover becomes independent of the VOI size. The classification error for spheres with 22mm diameter and larger and with SBR larger than 2.06 stays beneath a value 10%. Also the VOI-dependent loss of spheres regards just spheres of low SBR and small diameter. Increasing the VOI size the possibility for including outliers is raised which leads to a loss of more spheres than it is the case with smaller VOIs. This fact is dedicated to the noisiness of the images. In general, the volume overestimation for small spheres is not corrected with this subsequent parameter assignment.

To address the overestimation of small spheres we remember that figure 5.3 (e) exhibit the maxi-

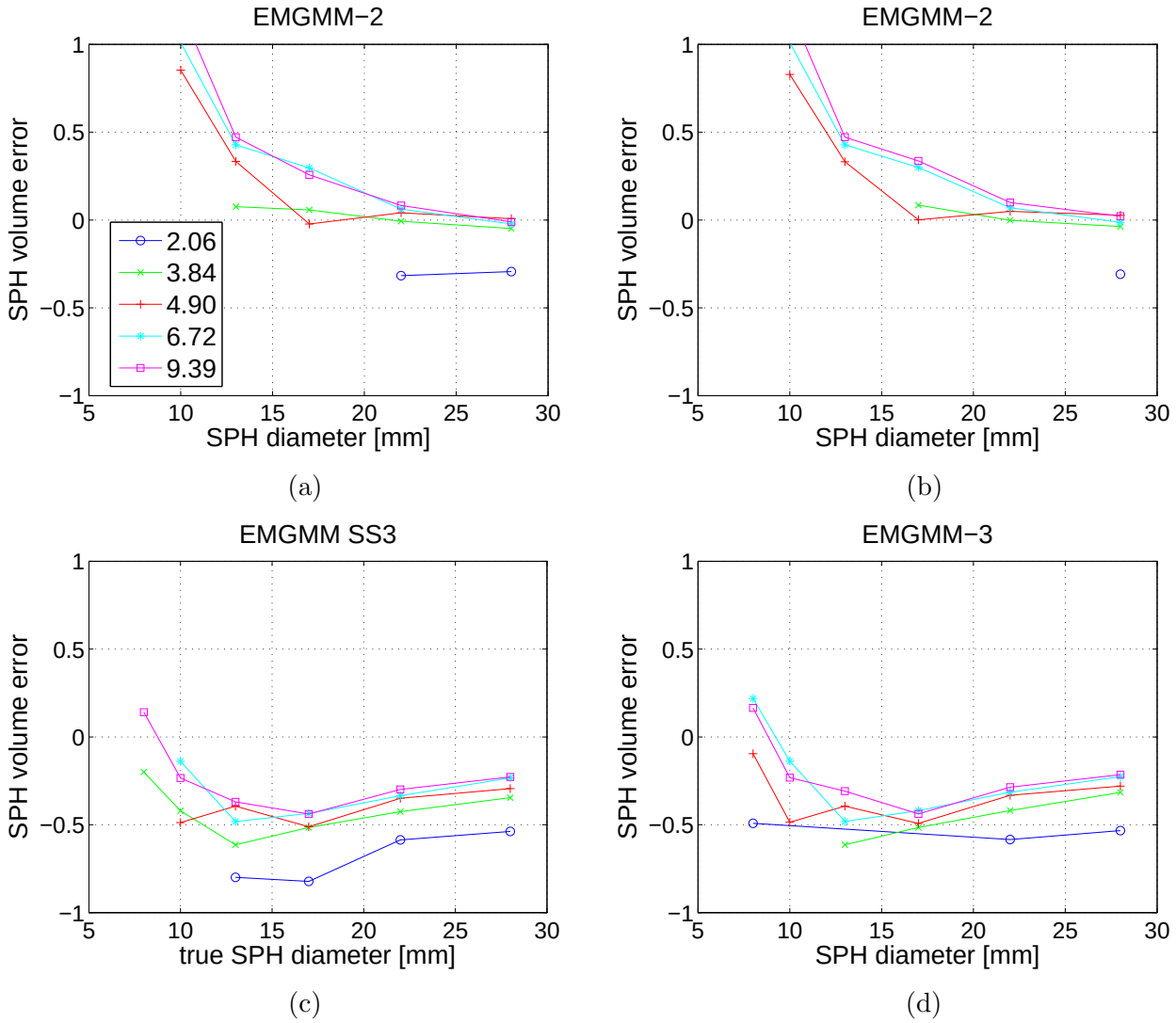


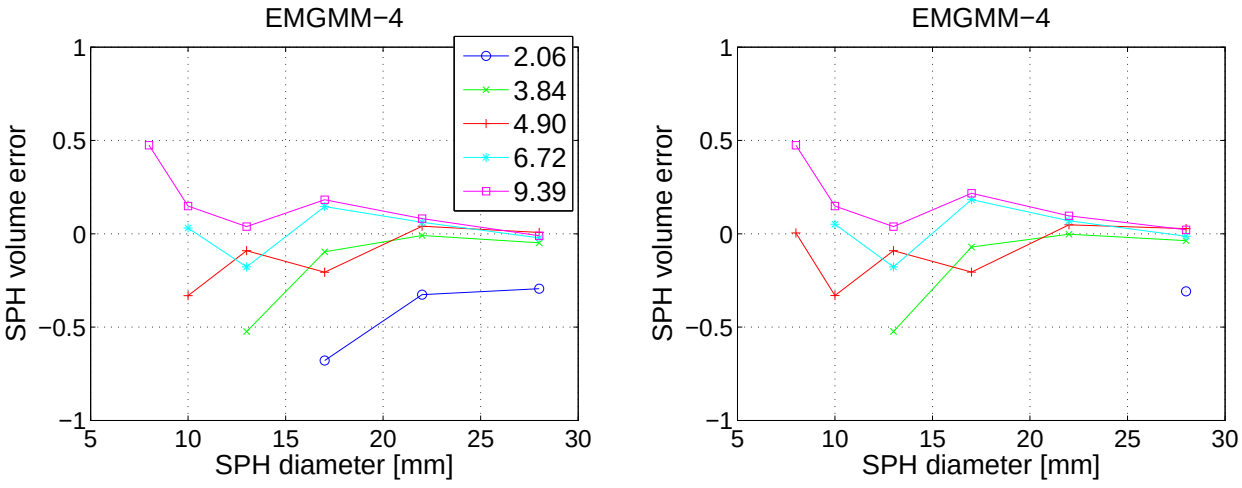
Figure 5.14: SPH volume error estimated in VOIs comprising $14\times14\times20$ voxels by (a) EMGMM-2 and (c) EMGMM-3; SPH volume error estimated in VOIs comprising $14\times14\times40$ voxels by (b) EMGMM-2 and (d) EMGMM-3. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

mum activity concentration as a better estimator for the sphere mean than the estimated mean value. Using the assignment from EMGMM-2 and employing the maximum voxel value as pendant for the sphere mean, a second modification named EMGMM-3 uses the assignment

$$\mu_{\mathrm{SPH}} = \max_n\{x_n z_{\mathrm{n,SPH}}\} \tag{5.9}$$

$$\sigma_{\mathrm{SPH}} \longleftarrow \sigma_{\mathrm{CYL}}. \tag{5.10}$$

Results of the EMGMM-3 procedure are drawn in figure 5.14(c) for VOIs comprising $14\times14\times20$ voxels and in figure 5.14 (d) for VOIs comprising $14\times14\times40$ voxels of OSEM2D reconstructed images. This



(a) (b)

Figure 5.15: SPH volume error estimated in VOIs comprising (a) $14\times14\times20$ and (b) $14\times14\times40$ voxels by EMGMM-4. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

choice of parameter updates leads to underestimation of all sphere volumes except for the smallest one with higher SBR. Again the sphere measurements with SBR lower than 3.84 and diameter lower than 10mm do yield a VOI-dependent detection behaviour. Also the volume errors achieved with this algorithm are stable w.r.t. the number of voxels comprised in the chosen VOI.

Having a look at these results, it is obvious that for larger spheres the EMGMM-2 approach yields good results whereas for smaller spheres EMGMM-3 reduces the overestimation and increases the detectability against EMGMM-2. It is thus desireable, to switch between EMGMM-2 and EMGMM-3 in a continuous manner. This can be achieved using the function

$$f(V_{\mathrm{SPH}}) = 1 - \exp\left(-V_{\mathrm{SPH}} + V_{\mathrm{VOX}}\right) \in (-\infty, 1). \tag{5.11}$$

The function yields 1 if the sphere volume is much larger than the voxel volume. If V_{SPH} is in the order of the voxel size $V_{\mathrm{SPH}} \to V_{\mathrm{VOX}}$, $f(V_{\mathrm{SPH}})$ approaches zero. With this, the parameter updates according to

$$\sigma_{\mathrm{SPH}} \longleftarrow \sigma_{\mathrm{CYL}} \tag{5.12}$$

$$\mu_{\mathrm{SPH}} \longleftarrow \max_n\{x_n z_{\mathrm{n,SPH}}\}[1 - f(V_{\mathrm{SPH}})] + \mu_{\mathrm{SPH}} f(V_{\mathrm{SPH}}). \tag{5.13}$$

Hence the estimator for μ_{SPH} becomes a weighted trade-off that depends on the sphere volume.

The behaviour of this algorithm is illustrated in figure 5.15 (a) for VOIs comprising $14\times14\times20$ voxels and in figure 5.15 (b) for VOIs comprising $14\times14\times40$ voxels of OSEM2D reconstructed images and is called EMGMM-4. It can be seen that the volume estimates as well as the detectability is stabilized against VOI sizes except for the measurement with SBR=2.06. Moreover the volume overestimation

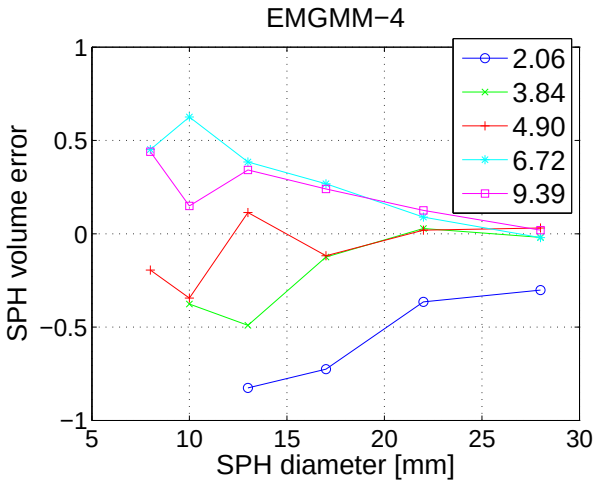


Figure 5.16: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by EMGMM-4. All data plotted versus sphere diameter for the NEMA phantom reconstructed with BP. Each graph comprises curves for various SBR, connecting measurements for each sphere.

of small spheres is reduced when comparing the results with EMGMM-2. Table. 5.4 summarizes the various parameter assignment strategies.

To investigate the stability regarding different reconstruction algorithms of the PET images, the EMGMM-4 procedure is applied to VOIs of $14\times14\times20$ voxels of the BP reconstructed NEMA phantom images. The result is shown in figure 5.16. Compared to solutions with OSEM2d reconstructions, the volume estimates are stable especially for the large spheres. At smaller sphere diameters, the detection behaviour within BP reconstructions is raised. Differences in volume errors are not exceeding 50%.

tag:	parameter assignment
2:	$\sigma_{\mathrm{SPH}} \longleftarrow \sigma_{\mathrm{CYL}}$
3:	$\begin{cases} \mu_{\mathrm{SPH}} & = \max\limits_{n}\{x_n z_{\mathrm{n,SPH}}\} \\ \sigma_{\mathrm{SPH}} & \longleftarrow \sigma_{\mathrm{CYL}} \end{cases}$
4:	$\begin{cases} \mu_{\mathrm{SPH}} & \longleftarrow \max\limits_{n}\{x_n z_{\mathrm{n,SPH}}\}[1 - f(V_{\mathrm{SPH}})] + \mu_{\mathrm{SPH}} f(V_{\mathrm{SPH}}) \\ \sigma_{\mathrm{SPH}} & \longleftarrow \sigma_{\mathrm{CYL}} \end{cases}$

Table 5.4: Table summing the various subsequent parameter assignments and tags.

5.5 MLGM

5.5.1 MLGM - Two Clusters

As one can see from section 5.4.2, the SBR-dependent missclassification is not corrected with EMGMM for small spheres with a diameter beneath 22mm. Moreover in general, the smaller the SBR gets the fewer spheres are detected. Using ad hoc parameter assignment as shown in table 5.4, solutions which are overestimating the sphere volumes are attained as well as solutions which are underestimating the sphere volumes. Combining some of them, the clustering results can be enhanced towards smaller missclassification and better detection statistics.

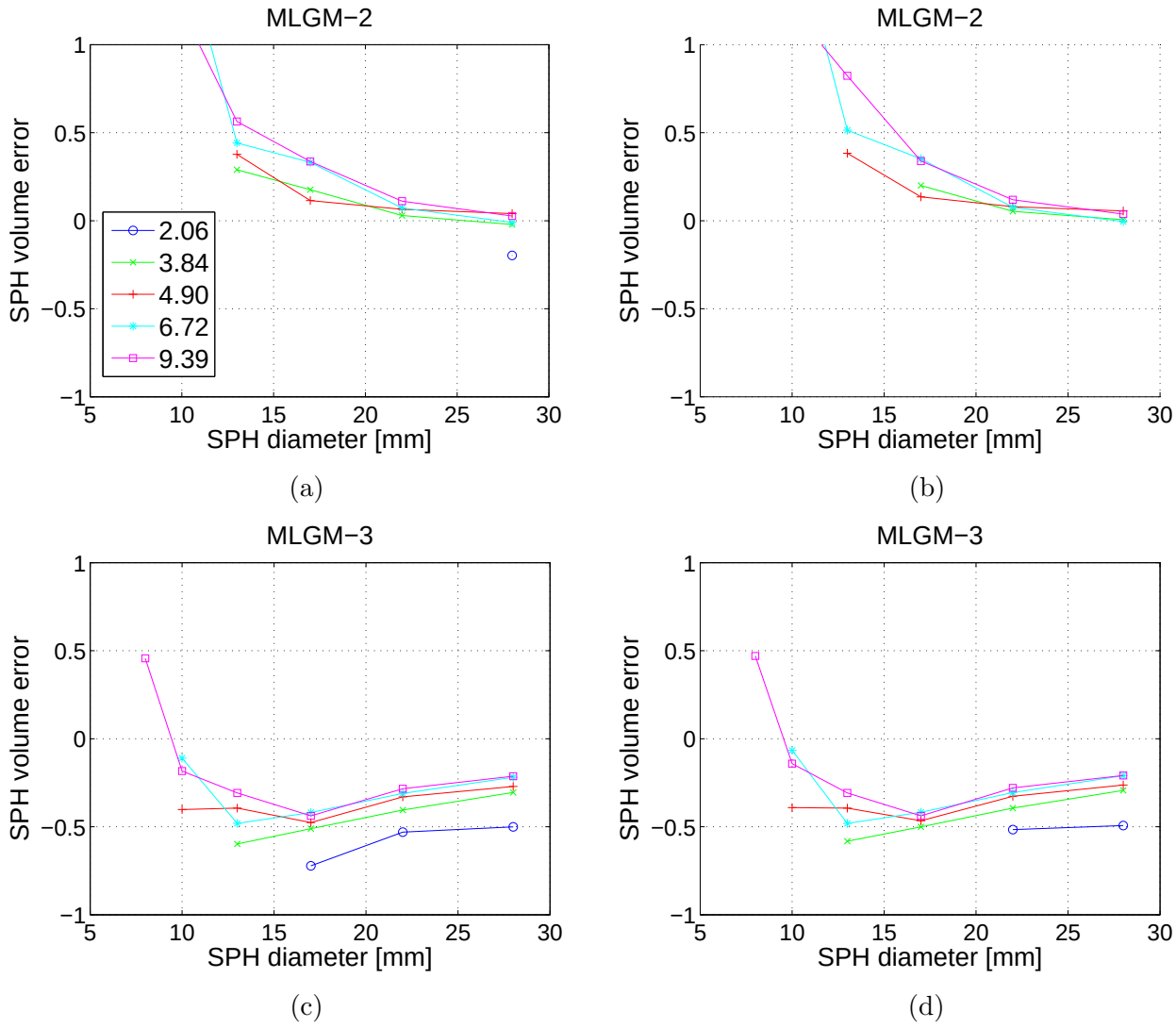
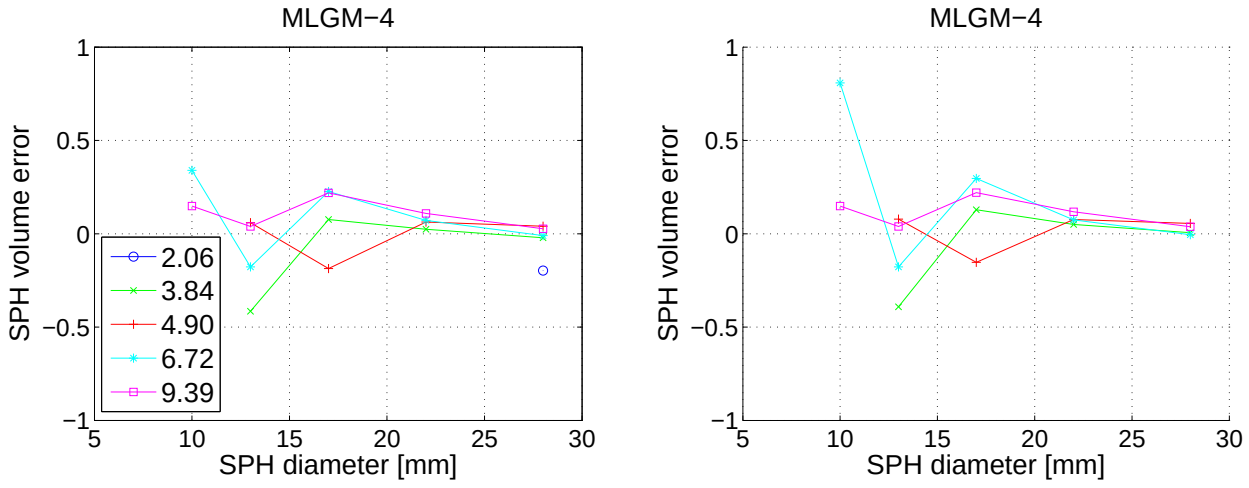


Figure 5.17: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by (a) MLGM-2 and (c) MLGM-3; SPH volume error estimated in VOIs comprising $14 \times 14 \times 40$ voxels by (b) MLGM-2 and (d) MLGM-3. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.



(a) (b)

Figure 5.18: SPH volume error estimated in VOIs comprising (a) $14 \times 14 \times 20$ and (b) $14 \times 14 \times 40$ voxels by MLGM-4. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

To check whether the multinomial prior distribution for the labeling matrix $p(\mathbf{Z})$ (A.9) incorporated to the model during an EMGMM procedure is beneficial in clustering NEMA spheres, the MLGM algorithm as discussed in section 4.1.1 is applied to the phantom measurements. Because small spheres are aimed to be detected, the VOIs under consideration do in general posses more cylinder voxels than sphere voxels. Moreover to get good statistical ensembles the VOIs are chosen large. It is therefore assumed that some prior distribution over $\mathbf{Z}$ would deteriorate the segmentation results as the probability for sphere voxels is significantly lower than the probability for cylinder voxels. A further difference to the EMGMM algorithm is, although the final labeling is done from a continuous probability masks, that the parameter estimation step is using discrete labellings.

As with EMGMM, an application of the MLGM framework with classical parameter updates does not capture the sphere statistics $\Theta_{SPH} = (\mu_{\mathrm{SPH}}, \sigma_{\mathrm{SPH}})$. Therefore many outliers are incorporated to the sphere clusters, violating the connectivity constraint as stated in section 5.2.

Employing again the two basic ad hoc parameter assignments from table 5.4 as done with EMGMM in section 5.4.2, figure 5.17 (a)-(d) is achieved. Figure 5.17 (a) and (c) depicts the relative measured volumes yielded by MLGM-2 and MLGM-3, estimated from VOIs containing $14 \times 14 \times 20$ voxels of OSEM2D reconstructed images. With exception to the detectability, which is decreased in comparison to the EMGMM method, the performance stays the same including a small shift to overestimations. To check for VOI dependencies, figure 5.17 (b) and (d) shows the MLGM-2 and MLGM-3 results of the relative measured volume estimated in VOIs containing $14 \times 14 \times 40$ voxels of OSEM2D reconstructed images. The results seem to be more stabilized but at the price of loosing more spheres in extreme cases than EMGMM.

It is mentioned that the spheres get lost due to incorporation of outliers to the sphere cluster. This confirms the assumption about the prior distribution for the label matrix $\mathbf{Z}$. MLGM is more sensitive to noise then EMGMM is, making the usage of the prior distribution $p(\mathbf{Z})$ for the label matrix a fine tuning. The prior probability further reduces the probability for being a sphere voxel.

For comparison purpose with the algorithms presented in the next subsection, the ad hoc parameter assignment MLGM-4 have been applied to OSEM2D reconstructed images for both VOIs, $14 \times 14 \times 20$ and $14 \times 14 \times 40$ voxels. The results are presented in figure 5.18. There it is seen that the best volume estimates are again achieved using the MLGM4 parameter assignment.

5.5.2 MLGMC - Covariances and Local Conditionals

An advanced application of the basic MLGM approach is to use correlations between voxels as discussed in section 4.1.2. With this algorithm, the parameter set is extended to include the correlations within a cluster k; these are estimated from the likelihood function (4.16). As already mentioned, the neighbourhood of small spheres differs from the neighbourhood of their larger pendants. Hence local interactions are assumed to change the behaviour of the pure MLGM approach by affecting differently sized spheres differently.

As discussed in section 5.1.2.1 the global correlations (GC) among voxels of the image are nearly zero. So it is expected that the solutions achieved due to incorporating global dependencies to the presented clustering problem are not greatly differing from versions without the inclusion of correlations. Instead using local correlations (LC) with correlation coefficients on the order of 0.6 should affect the clustering.

Incorporating local correlations was approved from analysis, but using local dependencies as shown in (4.17) results in estimating the whole VOI as sphere volume (with any choice of parameter settings shown in table 5.4). Obviously incorporating local correlations affects the basic problem heavily, overshooting the mark.

As discussed in section 5.1.2.1 the data is less correlated globally. If there is any global correlation then it occurs for small spheres and moreover gets negative, therefore having the opposite characteristics as the local correlations. So a modification of MLGM was implemented calculating global correlations as shown in (4.15) (among all voxels of a cluster) during the m-step of the iteration procedure and impose them for evaluating the labeling probabilities as before via local conditional probabilities. To account for the global character of the correlations, the local conditional distribution considers not only interactions of the main voxel with its neighbours. Instead the neighbours are also modeled as beeing correlated with a local covariance matrix (contrary to (4.18))

$$\Lambda_{\text{loc}} = \begin{pmatrix} \lambda & \nu & \cdots & \nu \\ \nu & \lambda & \cdots & \nu \\ \vdots & \vdots & \ddots & \vdots \\ \nu & \nu & \cdots & \lambda \end{pmatrix}. \tag{5.14}$$

The correlation matrix in (5.14) is governed by the two parameters, λ and ν, capturing voxel power

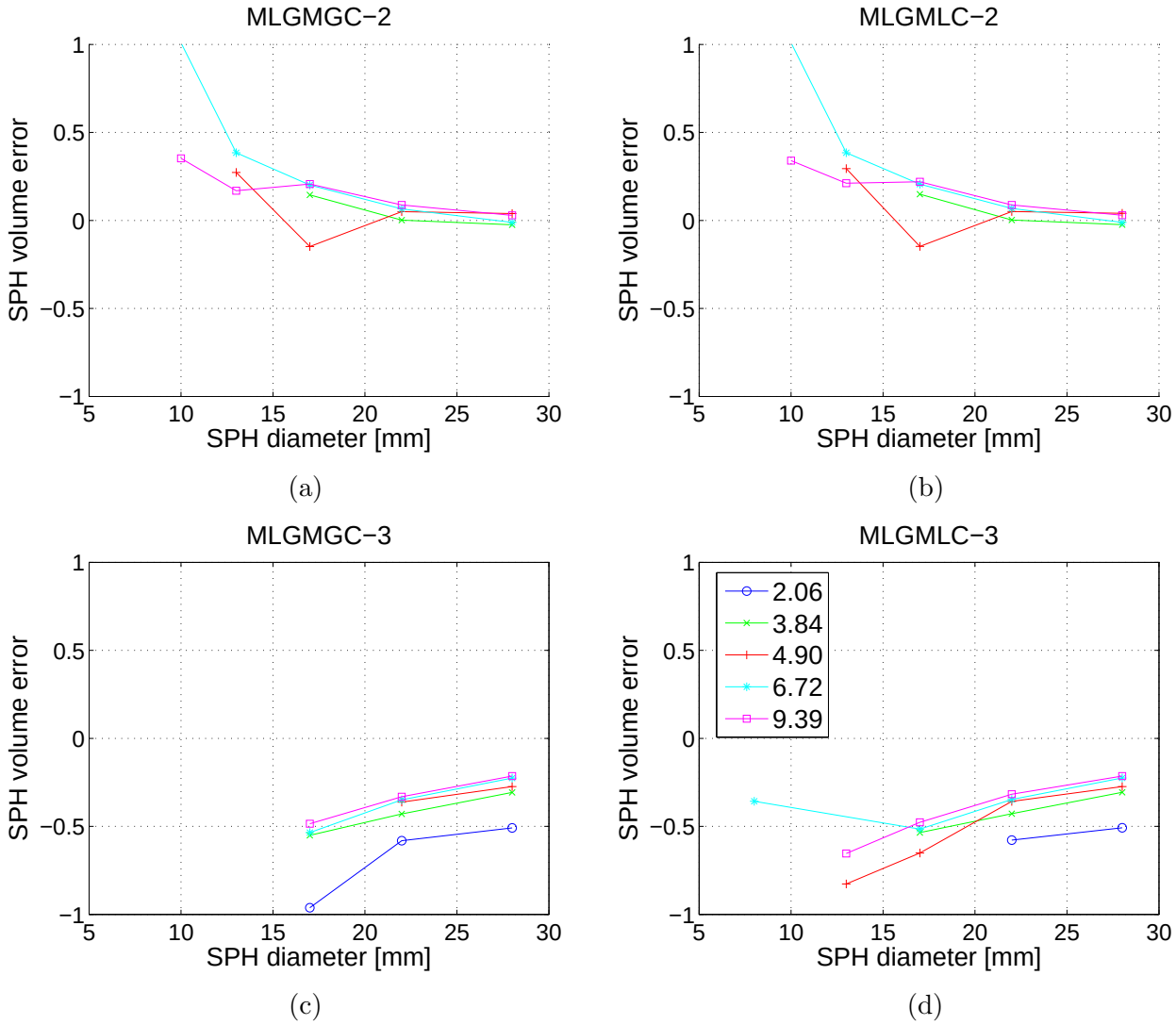
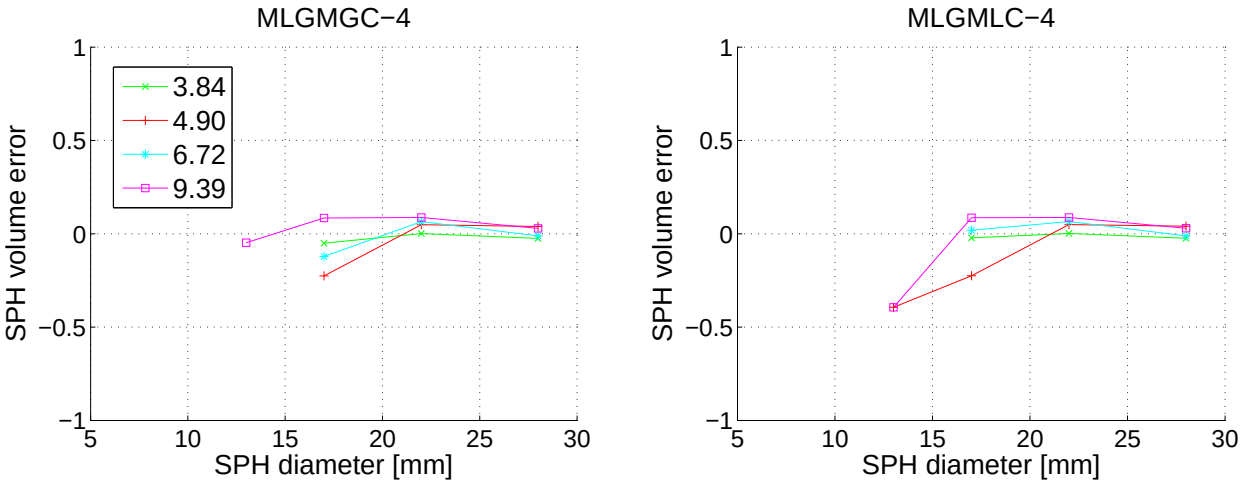


Figure 5.19: SPH volume error estimated in VOIs comprising $14 \times 14 \times 40$ voxels by (a) MLGMGC-2, (b) MLGMLC-2, (c) MLGMGC-3 and (d) MLGMLC-3. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

and correlation respectively. Hence the correlation ν is distance independent. So to perform a labeling for local values z_{nk} of $\mathbf{Z}$, the local conditional is calculated conditioned on the neighbourhood of x_n using the cluster correlations as local neighbourhood information which is also shared among the neighbourhood voxels.

Results are presented for VOIs comprising $14 \times 14 \times 40$ voxels in figure 5.19 (a) and (c) and in figure 5.20 (a) named MLGMGC (MLGM with global correlations), derived using OSEM2D reconstructed images of the NEMA phantom. Beside the loss of a sphere, the results stay constant regarding some volume variation of the VOIs and therefore the results from the smaller VOIs are omitted. Moreover using no ad hoc parameter assignment, no spheres are detected and therefore these results are



(a) (b)

Figure 5.20: SPH volume error estimated in VOIs comprising $14 \times 14 \times 40$ voxels by (a) MLGMGC-4 and (b) MLGMLC-4. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

not depicted.

At first it has to be mentioned that the parameter updates from table 5.4 are again done only for the mean and standard deviation. In general it can be seen that the estimation of global correlations included as local information is indeed reducing the overestimation of small spheres if comparing the results to those of the MLGM algorithm. Concentrating on the solutions of MLGMGC-2, the 10mm spheres of the two largest SBRs are detected as with MLGM-2. Moreover the volume estimates of MLGMGC are reduced to be more accurate than MLGM. But the remaining procedures, MLGMGC-3 and MLGMGC-4, are shooting over the target. Unfortunately this method is just doing a good job at larger spheres but loosing most of the small spheres, which are detected by EMGMM methods. Nevertheless the accuracy of the volume estimates of spheres larger than 13mm and for SBRs over 2.06 is not attained with any other method. This parameter assignment leads to a heavy reduction of misclassification for large spheres.

A last attempt to further improve the method is to reduce the interactions among neighbouring voxels and use a correlation matrix as defined in (4.17). With this choice, the correlations among the neighbouring voxels are ignored and just the interactions with the main voxel are considered. This means, estimating the global correlations in each cluster but including this information via local correlations of each voxel. Results of this approach are shown in figure 5.19 (b) and (d) and in figure 5.20 (b) named MLGMLC. Again without the usage of ad hoc parameter settings this method leads to tremendous overestimation for all spheres under consideration. Although in general more spheres are found, the detection yield of small spheres stays poor missing the smallest sphere of 8mm diameter at all SBRs.

As before, the method is constant for VOIs of different size and therefore a comparison is omitted. Although more spheres are detected with MLGMLC than with MLGMGC, detecting spheres with diameter smaller than 13mm is not ensured.

5.6 Bayesian Inference

As seen from the results in section 5.4.1, bad statistical ensembles of the sphere volumes lead to the inclusion of PVE voxels to the sphere volumes and therefore to tremendous overestimation of the spheres. This is not only the case for small spheres but also for the largest one of 28mm diameter. Therefore ad hoc parameter updates or parameter assignment of values extracted from the PET image to parameters of the sphere distributions was necessary to yield connected segmentation results and to enable a reduction of the volume overestimation of the spheres during EMGMM like procedures. Without those settings mostly no spheres were detected because not only overestimation is emerging but various outliers happen.

As we have seen by the examples shown in section 4.3.2 and section 4.3.3, treating certain parameters as random variables and therewith augmenting the probability distribution of the system with prior distributions, changes the update steps in case of bad statistical ensembles.

A step towards a Bayesian treatment was done via implementing an EMGMM framework incorporating a prior distribution for the parameter μ. Hence μ was also considered to be Gaussian distributed. The hyperparameter of the Gaussian prior mean was estimated from the mean of its neighbouring voxels $\mu_0(x_n) \propto \sum_{m \epsilon \mathcal{N}(n)} x_m$. As standard deviations, the ones from the Gaussian distribution describing the data X were employed as

$$p(X \mid \mathbf{Z}, \mu, \sigma^2) p(\mu; \mu_0) \approx \prod_k \exp\left\{ \frac{1}{2\sigma_k^2}(x_n - \mu_k)^2 + \frac{1}{2\sigma_k^2}(\mu_k - \mu_0)^2 \right\}.$$

This enables analytical solutions to the estimation problem (m-step). But as one can imagine, due to calculating the mean of the local neighbourhood of a voxel to assign for the local varying hyperparameter μ_0, this algorithm evolves like an EMGMM with a post processing average filtering. Therefore, the already existing point spread function gets smeared out and the volume overestimation is increased.

To go further on a Bayesian way we investigate the algorithms described in section 4.3.4 and section 4.3.5. The results are presented in the following two sections.

5.6.1 Bayesian EMGMM

The examples in section 4.3.2 and section 4.3.3 have shown that with a Bayesian treatment the update steps for the parameters consist of a term proportional to the classical ML estimator and an additional term governed by the hyperparameters. In case of small clusters the term involving the ML estimator become negligible, giving rise to the term governed by the hyperparameters. This effect is assumed to correct the unreliable parameter estimates due to bad statistical ensembles.

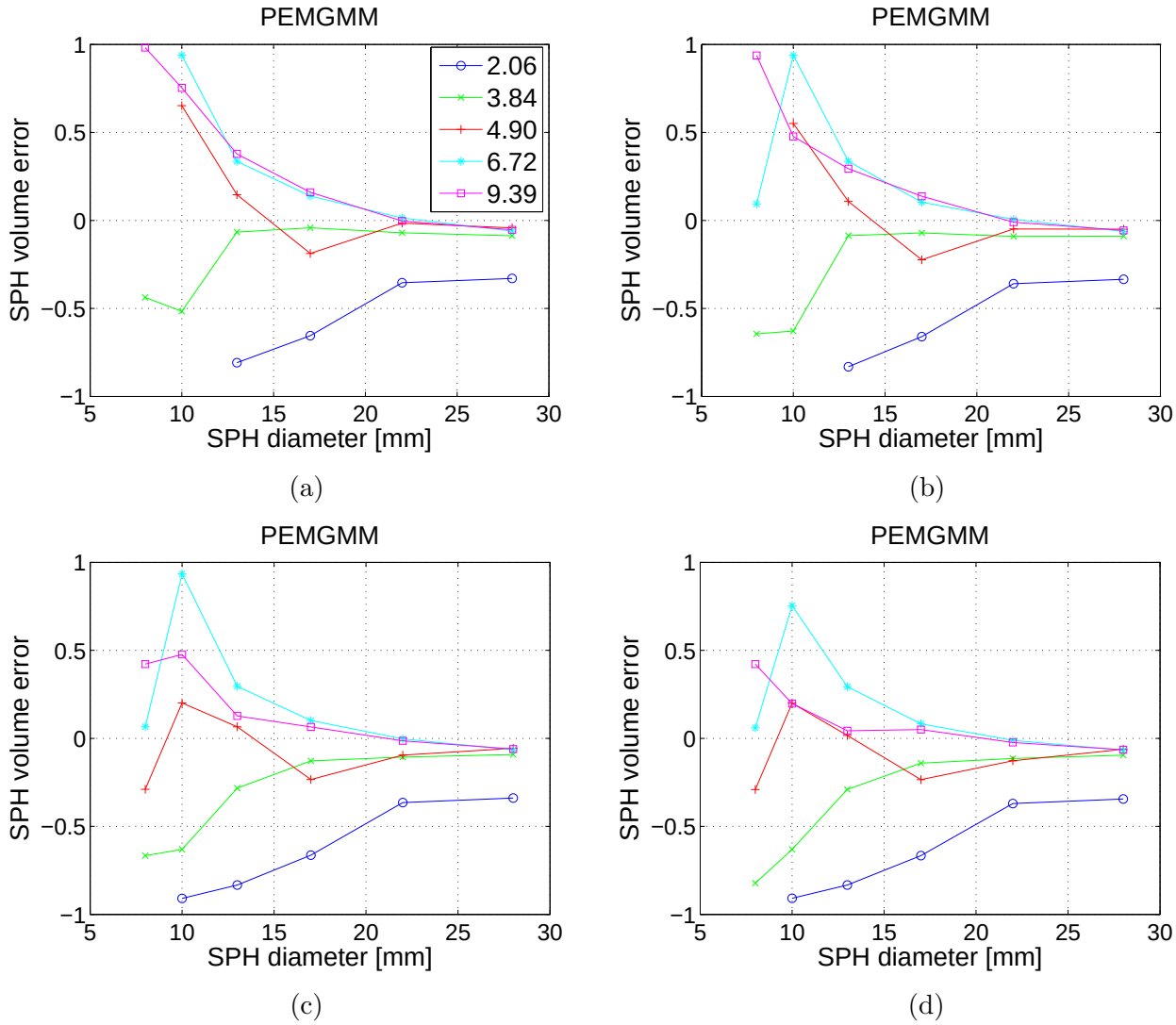
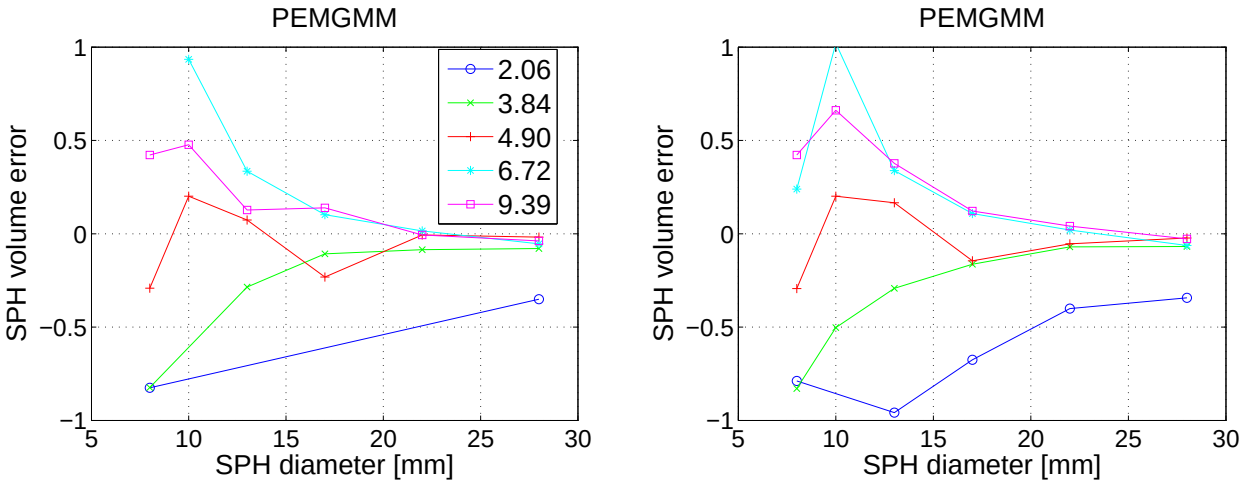


Figure 5.21: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by PEMGMM with (a) $a_{0,\mathrm{SPH}} = 2$, (b) $a_{0,\mathrm{SPH}} = 3$, (c) $a_{0,\mathrm{SPH}} = 4$ and (d) $a_{0,\mathrm{SPH}} = 5$. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

Assuming both Gaussian parameters, μ and σ, as random variables, a Bayesian EMGMM procedure was defined in section 4.3.4. As mentioned in section 3.3.3 and section 4.3.4, the labeling step stay the same as with classical EMGMM see (4.54). In contradiction the parameter update steps given by (4.55)–(4.61) are augmented with terms governed by the hyperparameters μ_0, σ_0, a_0 and b_0.

It is emphasized that the hyperparameters constitute a framework permitting to include information learned from data. Hence in this section we will study the behaviour of the PEMGMM, for certain hyperparameters are assigned by their true values. To achieve the desired statistics from the images we proceed as in section 5.1.2.1, where we have increased a threshold from zero upwards until the true volume of the corresponding sphere was detected. This way, we calculate the mean value of



(a) (b)

Figure 5.22: SPH volume error estimated in VOIs comprising $14 \times 14 \times 40$ voxels by PEMGMM with $a_{0,\mathrm{SPH}} = 4$ in (a) OSEM2D and (b) BP reconstructed NEMA phantom images. All data plotted versus sphere diameter. Each graph comprises curves for various SBR, connecting measurements for each sphere.

the true cylinder volume to define the parameter $\mu_{0,\mathrm{CYL}}$. According to our results in section 5.1.2.1 the maximum of the sphere volumes is assigned to the prior parameter $\mu_{0,\mathrm{SPH}}$. The standard deviation of the prior distribution for μ, σ_0, is set to the true standard deviation of each cluster of X.

Attempts to proceed with equivalent settings for the hyperparameters of the precision (setting b_0 to one and a_0 equal to the true precision in each cluster [2]) did not yield useful results. Searching good estimates for a_0 and b_0, we follow the idea that the standard deviation for spheres is tremendously overestimated with classical Gaussian models, see figure 5.13. As (4.56) and (4.57) show, the estimate of the standard deviation for a certain cluster k would be raised if the parameter $b_{0,k}$ is increased. There again increasing the parameter $a_{0,k}$ would correspond to decreased estimates of the standard deviation for the cluster k. So to decrease the overestimated standard deviation we proceeded by setting the parameter $b_{0,\mathrm{CYL}} = b_{0,\mathrm{CYL}} = a_{0,\mathrm{CYL}} = 0$ but the parameter $a_{0,\mathrm{SPH}}$ to a positive integer. Summing the parameter settings we get

$$\mu_{0,\mathrm{CYL}} = \mu_{\mathrm{CYL}}^{\mathrm{real}} \tag{5.15}$$

$$\mu_{0,\mathrm{SPH}} = \max_n \{x_n z_{\mathrm{n,SPH}}\} \tag{5.16}$$

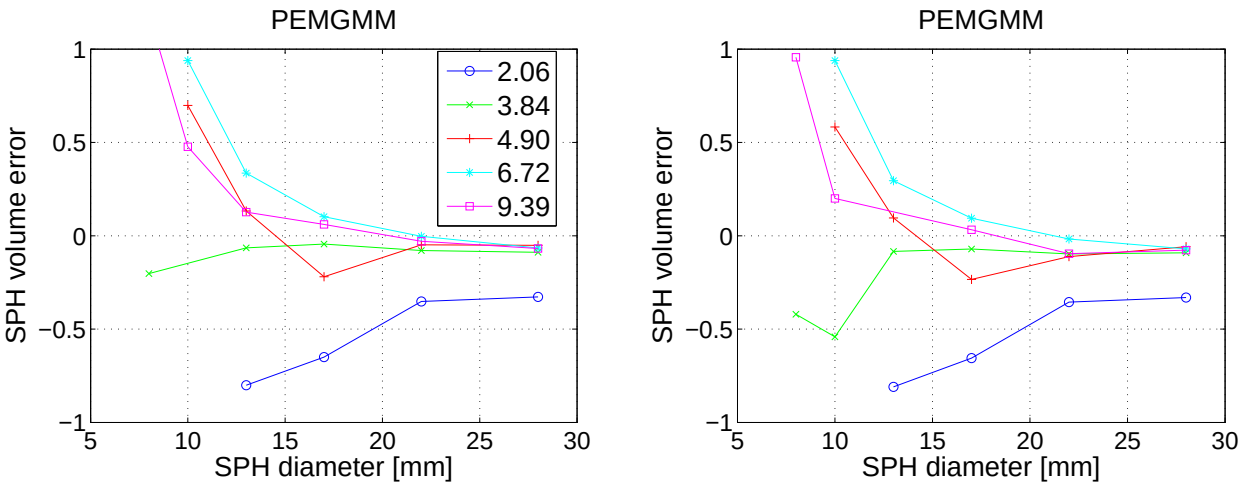
$$\sigma_{0,\mathrm{CYL}} = \sigma_{\mathrm{CYL}}^{\mathrm{real}} \tag{5.17}$$

$$\sigma_{0,\mathrm{SPH}} = \sigma_{\mathrm{CYL}}^{\mathrm{real}} \tag{5.18}$$

$$b_{0,\mathrm{CYL}} = b_{0,\mathrm{CYL}} = a_{0,\mathrm{CYL}} = 0 \tag{5.19}$$

$$a_{0,\mathrm{SPH}} = i \in \mathbb{N}^+, \tag{5.20}$$

[2] If the amount of voxels is zero, the precision is given by $\frac{a_0}{b_0}$.



(a) (b)

Figure 5.23: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by PEMGMM with (a) $a = \widehat{\mathrm{SBR}}$ and (b) $a = 1.4\widehat{\mathrm{SBR}}$. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

Applying the PEMGMM algorithm using solely the parameter settings (5.15)–(5.20) and comparing the solutions with those of the classic EMGMM without ad hoc parameter assignment, the detection statistic is raised just for the smallest spheres. Although this procedure seems to act as desired (reducing the volume overestimation of small spheres), in general no connected objects are detected.

Hence we further employ the subsequent parameter assignment as shown in table 5.4. But we do not assign the corresponding values before estimating the labels. The assignment is done for the ML estimators which get further merged with the terms governed by the hyperparameters. Results of the PEMGMM assignment are shown, applying the algorithm to OSEM2D reconstructed VOIs of size $14 \times 14 \times 20$ voxels, in figure 5.21. The subgraphs of each figure are obtained by setting the parameter a_0 for the spheres as: (a) $a_{0,\mathrm{SPH}} = 2$, (b) $a_{0,\mathrm{SPH}} = 3$, (c) $a_{0,\mathrm{SPH}} = 4$ and (d) $a_{0,\mathrm{SPH}} = 5$. The figure show that indeed the Bayesian treatment can help to reduce overestimation selectively on small spheres leaving the larger ones unchanged. Moreover it is seen that this behaviour is not SBR sensitive. Nevertheless the PEMGMM updates enhance the detectability of the spheres. Just the sphere with diameter 8mm at lowest SBR is not recognized. We note that the volume estimates with higher values ($a = 4$ and $a = 5$) are comparable to those achieved with EMGMM-4, see figure 5.15.

In addition, figure 5.22 (a) depict results obtained after applying the PEMGMM to OSEM2D reconstructed VOIs of $14 \times 14 \times 40$ voxels and results after applying the PEMGMM to BP reconstructed VOIs of $14 \times 14 \times 40$ voxels. This reveals the PEMGMM to be stable concerning the VOI size (ignoring the lowest SBR measurements and small spheres). Moreover the differences of the volume estimates in OSEM2D and BP reconstructed images is negligible for larger spheres and do not exceed 20% for

the small spheres.

A last attempt to remove the strong SBR diversification regarding small spheres was made by assigning

$$a_{0,\mathrm{SPH}} = i \cdot \widehat{\mathrm{SBR}}, \tag{5.21}$$

with $i \in \mathbb{R}^+$ and $\widehat{\mathrm{SBR}} = \frac{\mu_{\mathrm{N,SPH}}}{\mu_{\mathrm{N,CYL}}}$. Figure 5.23 shows the results of this assignment for PEMGMM. Thereby the positive real value i in (5.21) was set to (a) $i = 1$ and (b) $i = 1.4$. Regarding the settings incorporating $\widehat{\mathrm{SBR}}$ in $a_{0,\mathrm{SPH}}$, no great benefits are attained.

5.6.2 Variational Bayesian Inference

With a fully Bayesian treatment of the image clustering problem, section 4.3.5, all parameters τ, μ and σ are assumed to be random variables having prior distributions. The prior distributions are again parametrized probability distributions having so called hyperparameters, see (4.71)–(4.74). Considering the model from section 4.3.5 in more detail reveals that due to the usage of the constant β_0 with values greater than one, the prior terms for μ_N and λ_N (the inverse precision) get more significant in contrast to the ML term even if the sphere is not as small, see (4.71)–(4.74).

To define the hyperparameters appropriately for this problem the settings are partially oriented on the parameter assignments from table 5.4. Again various settings have been tested. A common setting which is done in every run is assigning the hyperparameters μ_0 in (4.66), a_0 in (4.68) and b_0 in (4.69). Likewise some settings in table 5.4 the hyperparameter μ_0 for the MMSE of the mean μ_{N} is assigned by

$$\begin{aligned} \mu_{0,\mathrm{SPH}} &= \max_n \{x_n z_{\mathrm{n,SPH}}\} \\ \mu_{0,\mathrm{CYL}} &= \mu_{\mathrm{N}}. \end{aligned} \tag{5.22}$$

Further the hyperparameters a_0 and b_0, whose ratio euqals standard deviation respectively the precision ($\frac{1}{\sigma_N} = \lambda_N = \frac{a_N}{b_N}$), are constrained according to the settings in table 5.4 as

$$a_{0,\mathrm{SPH}} = a_{\mathrm{N,CYL}} \tag{5.23}$$

$$b_{0,\mathrm{SPH}} = b_{\mathrm{N,CYL}} \tag{5.24}$$

$$a_{0,\mathrm{CYL}} = a_{\mathrm{N,CYL}} \tag{5.25}$$

$$b_{0,\mathrm{CYL}} = b_{\mathrm{N,CYL}}, \tag{5.26}$$

whereby (5.23) and (5.24) results in the parameter assignment $\sigma_{\mathrm{N,CYL}} \rightarrow \sigma_{\mathrm{N,SPH}}$ as done with the ad hoc 2 settings in table 5.4.

β_0 emerge as factor of μ_0 for the calculation of μ_N (4.66) and for the calculation of λ_N (4.69) coupling the MMSE of the mean and the MMSE of the precision. Setting β_0 equal to zero would lead to vanishing prior terms. A value of 0.5 for $\beta_{0,\mathrm{SPH}}$ and $\beta_{0,\mathrm{CYL}}$ have shown to yield comparable results to those of EMGMM procedures.

Solely using these settings the procedure is termed the VBGMM-1 which is shown in figure 5.24 (a) for OSEM2D reconstructed phantom images covering VOIs of $14 \times 14 \times 20$ voxels. These settings

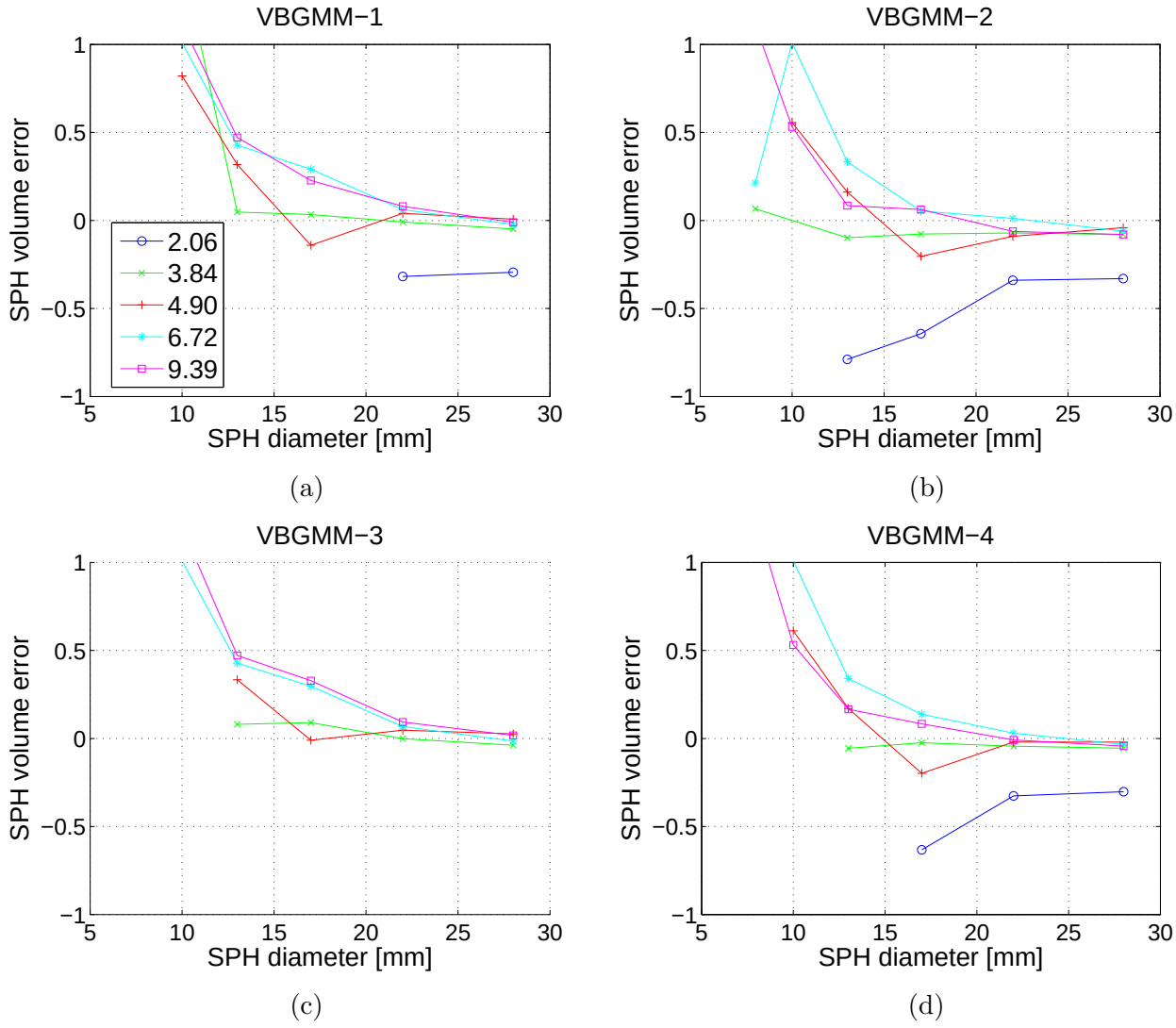


Figure 5.24: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by (a) VBGMM-1, (b) VBGMM-2, (c) VBGMM-3 and (d) VBGMM-4. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

already yield segmentation results as with ad hoc parameter assignment. The results are comparable to those achieved by EMGMM-2 with overestimation of small spheres (which moreover show reduced diversification among various SBR measurements). This is not surprising as the hyperparameters are chosen to fulfil equivalent constraints as the main parameter for the spheres does with EMGMM-2 settings from table 5.4.

Trying to diminish the volume overestimation of small spheres obtained using the hyperparameter settings for VBGMM-1, the parameter $\beta_{0,\mathrm{SPH}}$ is further exploited. Increasing $\beta_{0,\mathrm{SPH}}$ leads to an enhanced weighting of the prior terms emerging due to calculation of the MMSE. Therefore $\beta_{0,\mathrm{SPH}}$ has to be chosen greater than one. In contradiction it does not matter which value $\beta_{0,\mathrm{CYL}}$ is given,

because via (5.22), (5.25), and (5.26) the hyperparameters for the cylinder volumes are set to their ML estimate and therefore do not yield new information for the calculation of the MMSE.

Defining the $\widehat{\text{SBR}}$ as the ratio of the maximum voxel value to the mean value of the cylinder voxels $\mu_{\text{N,CYL}}$ according to $\widehat{\text{SBR}} = \frac{\max_n\{x_n z_{\text{n,SPH}}\}}{\mu_{\text{N,CYL}}}$, the parameter setting is done via

$$\begin{aligned} \text{VBGMM-2} \quad : \quad \beta_{0,\text{CYL}} &= \widehat{\text{SBR}}^{-1} \\ \beta_{0,\text{SPH}} &= \widehat{\text{SBR}}. \end{aligned}$$

Having evaluated these assignments on VOIs of $14 \times 14 \times 20$ voxels of the OSEM2D image reconstructions, the results are presented in figure 5.24 (b). It can be stated that indeed the volume estimates are shrunk for all spheres, having more effect on the smallest ones. Having defined the $\widehat{\text{SBR}}$ as an entity which is extracted from each image separately during each iteration, the parameters of larger spheres get more affected as their predictability of the true activity level (represented by the maximum activity value) is better than for small spheres resulting in larger $\widehat{\text{SBR}}$, see figure 5.3. Although the $\widehat{\text{SBR}}$ for small spheres is significantly lower than for large spheres, for small spheres the effect is more visible. Moreover it can be seen from figure 5.24 (b) that compared to VBGMM-1 the detectability for small spheres is increased as well as the detectability of spheres measured with SBR of 2.06.

A further attempt has been made employing the hyperparameters α_0 for the prior distributions of the parameters τ see (4.70). The first idea behind the following setting is to raise the parameter $\alpha_{0,\text{SPH}}$ for cancerous tissue to raise the detectability of small spheres. Using again the definition of $\widehat{\text{SBR}}$ as above and defining the amount of voxels being members of the two objects as $N_{k,\text{CYL}}$ (healthy tissue) and $N_{k,\text{SPH}}$ (cancerous tissue), the following assignments are tagged VBGMM-3

$$\begin{aligned} \text{VBGMM-3} \quad : \quad \alpha_{0,\text{CYL}} &= N_{k,\text{CYL}}\widehat{\text{SBR}}^{-1} \\ \alpha_{0,\text{SPH}} &= N_{k,\text{SPH}}\widehat{\text{SBR}}. \end{aligned}$$

Results of this approach are shown in figure 5.24 (c) with no benefit compared to the previous approaches. Combining further the updates from VBGMM-2 and VBGMM-3, the hyperparameter assignments are implemented as

$$\begin{aligned} \text{VBGMM-4} \quad : \quad \beta_{0,\text{CYL}} &= \widehat{\text{SBR}}^{-1} \\ \beta_{0,\text{SPH}} &= \widehat{\text{SBR}} \\ \alpha_{0,\text{CYL}} &= N_{k,\text{CYL}}\widehat{\text{SBR}}^{-1} \\ \alpha_{0,\text{SPH}} &= N_{k,\text{SPH}}\widehat{\text{SBR}} \end{aligned}$$

having solutions presented in figure 5.24 (d). The results of the methods including the hyperparameters α_0 (figure 5.24 (c) and (d)) show slightly larger estimates of the sphere volumes ranging from small to big ones but with no general differences to the case without using α_0, disregarding the missdetection of small spheres recognized during both methods.

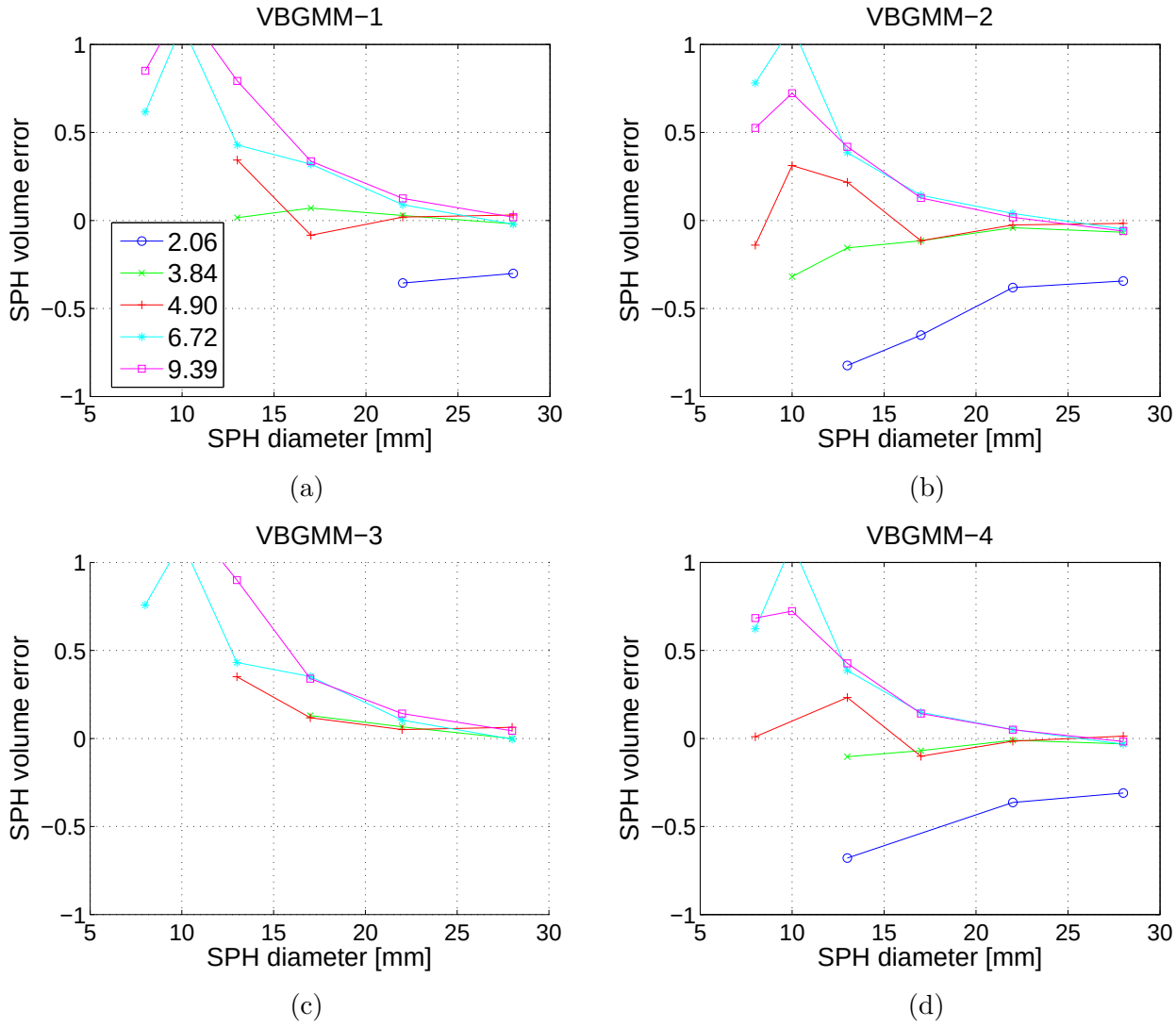


Figure 5.25: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by (a) VBGMM-1, (b) VBGMM-2, (c) VBGMM-3 and (d) VBGMM-4. All data plotted versus sphere diameter for the NEMA phantom reconstructed with BP. Each graph comprises curves for various SBR, connecting measurements for each sphere.

To summarize the behaviour of the Bayesian treatment introduced in this section, minor benefits are obtained in comparison with the ad hoc assignments done in previous sections see table 5.4.

To finally compare again the results with reconstructions achieved analysing other image reconstructions, the VBGMM algorithm is applied to the BP reconstructed NEMA phantom images. The VOI size was chosen $14 \times 14 \times 20$ voxels. The segmentation outcomes are presented in figure 5.25 for the various hyperparameter settings as done with the OSEM2D reconstructions. In general the volume estimates for larger spheres are stable. Differences regarding the SBR are not visible. For the small VOIs, the detection statistic is not equal. Moreover their volume estimates differ, but again the largest deviation from results obtained with OSEM2D is $\approx 50\%$. But in case of using VBGMM,

just spheres of 8mm diameter are misclassified with such error. The volume error difference of 10mm sphere is mostly around 20%.

5.7 Graphical Models

So far the incorporation of correlations among voxels x_n, see section 5.5.2, have not resulted in correcting the missclassification for small spheres. Using the MLGMGC procedure and the MLGMLC procedure (assuming global respectively local correlations) from section 5.5.2, the volume estimates are accurate but their sensitivity regarding noise have led to the loss of small spheres.

Furthermore the basic model, a mixture of two Gaussians, does not accurately fit to the measurements. This matter was compensated by constraining specific parameter values of the sphere cluster see table 5.4. A more mathematical attempt to correct for the problem of bad ML estimators due to bad statistical ensembles was used in section 5.6, treating the model parameters as random variables and including prior distributions governed by hyperparameters.

As an advanced attempt to overcome the drawbacks, we apply models as described in section 3.4. In contradiction to the algorithm provided in section 5.5.2, where correlations for the data matrix X were incorporated, the graphical model from section 4.4 is introducing correlations just among the labels of the label matrix $\mathbf{Z}$. Moreover, instead of the prior parameter τ used within the GMM model (see section A.4), the Potts model parameters α and $\tilde{\alpha}$ have been introduced in (4.75). Actually the part of the Potts model distribution governed by the parameter α can be considered as a pendant to the generalized Bernoulli distribution (A.2) governed by the parameter τ. So the innovation of this model is brought by the mixing term in (4.75) governed by the parameter $\tilde{\alpha}$.

5.7.1 Neighbourhood Systems

As defined by (4.76) the local conditional probability of z_n takes interactions among its neighbourhood into account via the term

$$\exp\left\{\sum_{m\in\mathcal{N}(z_n)} z_{nk}\tilde{\alpha}_{kl}z_{ml}\right\}, \tag{5.27}$$

with sufficient statistics according to $T_{\tilde{\alpha}} = z_{nk}z_{ml}$. (4.76) is the main equation we need during sampling algorithms. As mentioned in section 3.4.2 and section 4.4.2.2, conditioning a variable on its neighbours renders the variable independent from the remaining variables offering the possibility of calculating local marginal probabilities for labels z_{nk}.

Because there are various neighbourhood systems which can be considered (see figure 5.26), it is assumed that each of them is having its own sufficient statistics and hence with its own interaction parameter $\tilde{\alpha}$. In two dimensions the graphical visualization of the neighbourhoods is straightforward which is no longer the case in three dimensions. Therefore in figure 5.26 the neighbourhood system of a node (red) is shown in two dimensions. The edges are left out to cover all systems in one picture by changed colours. If it is assumed that the red node is connected with a first order neighbourhood

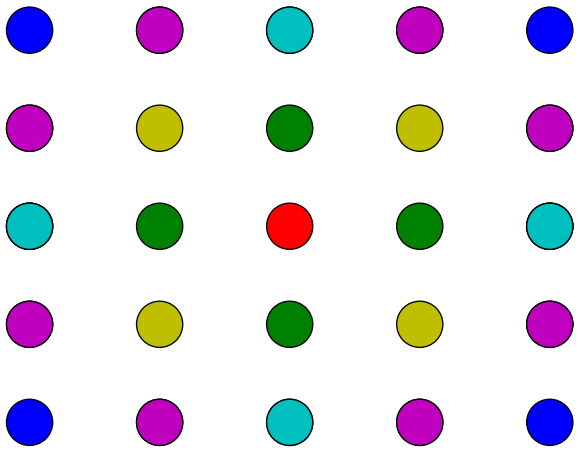


Figure 5.26: Various neighbourhood systems of a main node (red) drawn by different colours for a two-dimensional 5 times 5 graph of an image. The edges which in general are indexing the interactions of vertices/voxels are left out to not overload the graphical representation.

then it is connected just to the dark green nodes. Taking a second order neighbourhood into account then the red node is connected to the light green nodes and so on.

With the three-dimensional approach instead, a node is having six first order neighbouring voxels. The second order neighbourhood comprises 12 voxels and so on. Table 5.5 summarize the neighbourhood systems which we use with the GMRF approach. With this the interaction term in (4.76) gets a sum over all neighbourhood systems as

$$\exp\left\{\sum_{j}\sum_{m\in\mathcal{N}_j(z_n)} z_{nk}\tilde{\alpha}_{kl,j}z_{ml}\right\}. \tag{5.28}$$

Note that the parameters of each neighbourhood $\tilde{\alpha}_{kl,j}$ are estimated separately to give more flexibility in fitting the model to the data X.

neigh. order	$\mathcal{N}_1$	$\mathcal{N}_2$	$\mathcal{N}_3$	$\mathcal{N}_4$	$\mathcal{N}_5$	$\mathcal{N}_6$
voxels	6	12	8	6	24	24

Table 5.5: Table of neighbourhood systems used with GMRF. The first row determines the labels whereby the second row depicts the size of the neighbourhood.

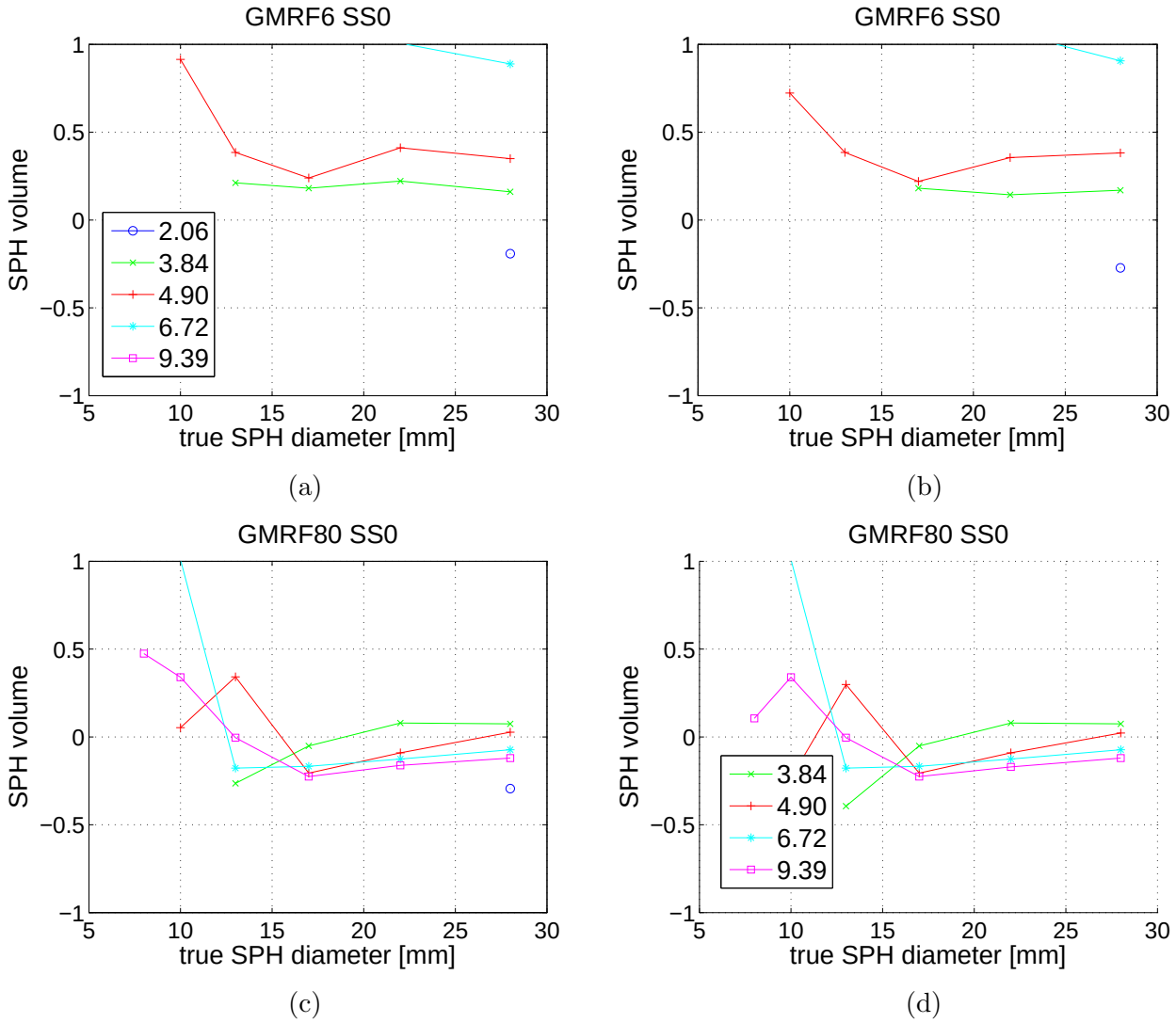


Figure 5.27: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by (a) GMRF6 and (c) GMRF80; SPH volume error estimated in VOIs comprising $14 \times 14 \times 40$ voxels by (b) GMRF6 and (d) GMRF80. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

5.7.2 Monte Carlo Labeling - Mean Field Approach

Employing a GMRF as discussed in section 4.4, two possible parameter update procedures have been presented. Here we start with the mean field approach described in section 4.4.3.1. Hence the conditional probability for labeling a voxel as sphere given a neighbour voxel which is member of the cylinder cluster, is equal to the opposite case. The labeling is done using the MCMC sampling algorithm from section 4.4.2.2.

The neighbourhood systems under consideration are built up as follows. The basic neighbourhood consists of the $\mathcal{N}_1$ as shown in table 5.5 which is tagged GMRF6. Further calculations extend this

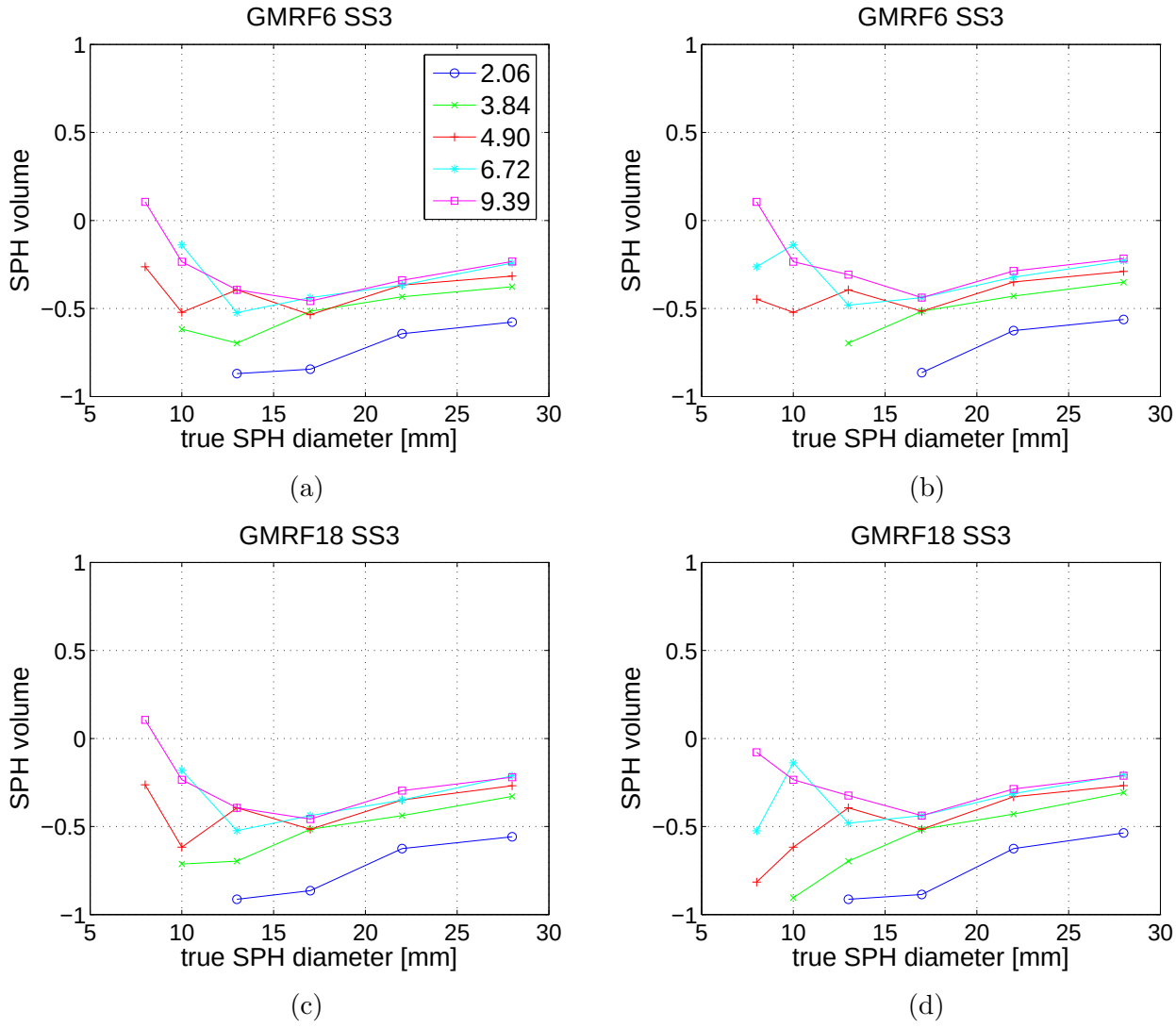


Figure 5.28: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by (a) GMRF6-3 and (c) GMRF18-3; SPH volume error estimated in VOIs comprising $14 \times 14 \times 40$ voxels by (b) GMRF6-3 and (d) GMRF18-3. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

basic neighbourhood system by adding a second and third order neighbourhood to $\mathcal{N}_1$. In this sense, GMRF18 calculates interactions due to the neighbourhood systems $\mathcal{N}_1$ and $\mathcal{N}_2$. GMRF26 is additionally concerning the third order system $\mathcal{N}_3$ and GMRF80 incorporates all of the neighbourhoods shown in table 5.5, $\mathcal{N}_1$,$\mathcal{N}_2$, $\mathcal{N}_3$,$\mathcal{N}_4$,$\mathcal{N}_5$ and $\mathcal{N}_6$. Table 5.6 summarizes the various neighbourhood systems used during section 5.7.

A major difference to the algorithms discussed before is revealed by analysing the clustering results which arise due to applying the basic ML parameter updates for the mean and standard deviation of the spheres. Employing the GMRF6 and GMRF80 algorithms on VOIs comprising $14 \times 14 \times 20$

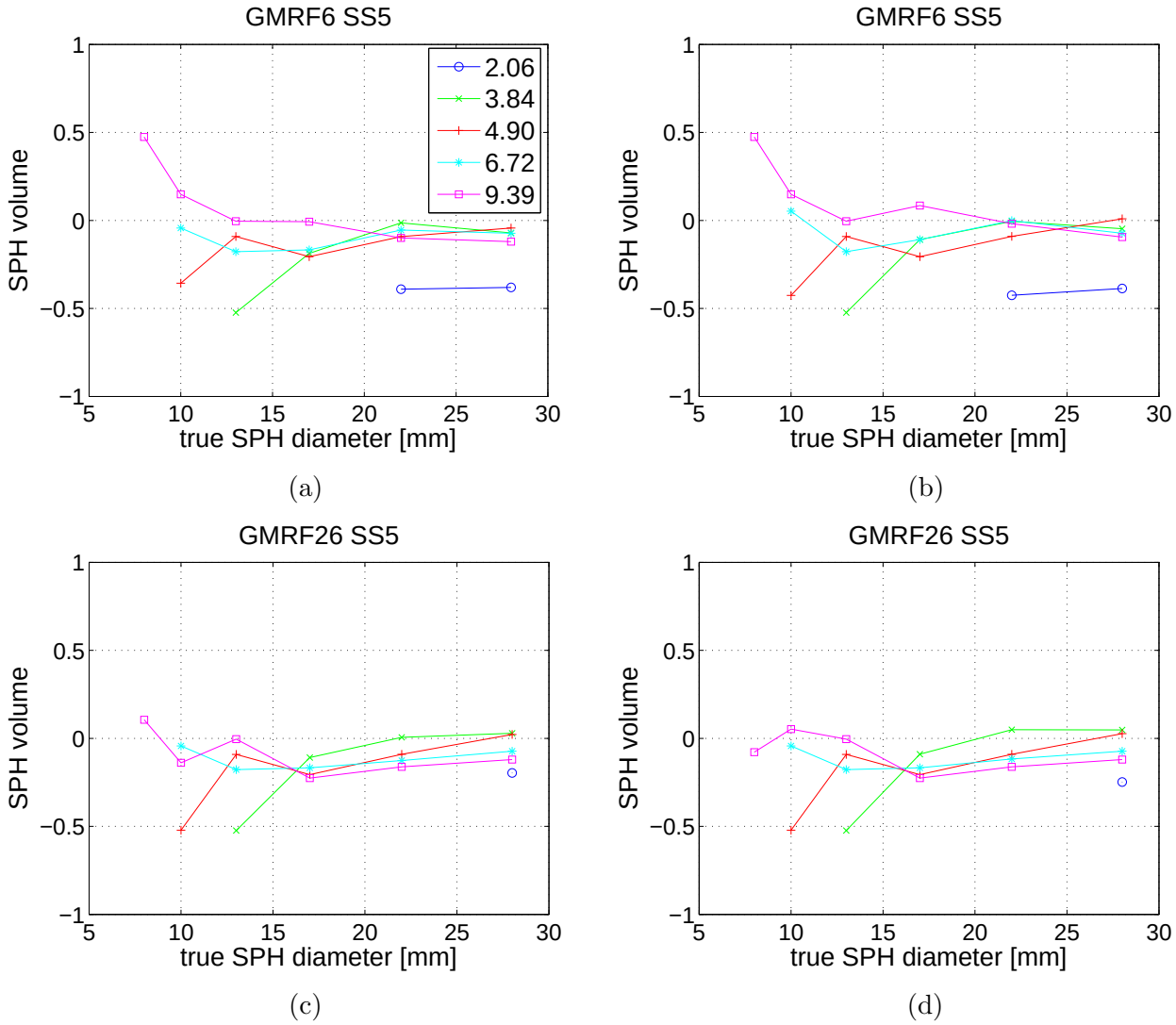


Figure 5.29: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by (a) GMRF6-4 and (c) GMRF26-4; SPH volume error estimated in VOIs comprising $14 \times 14 \times 40$ voxels by (b) GMRF6-4 and (d) GMRF26-4. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

voxels of OSEM2D reconstructed images, figure 5.27(a) respectively figure 5.27(c) shows the volume estimation error of the spheres.

Important to note is that even with the smallest neighbourhood $\mathcal{N}_1$ (figure 5.27 (a)), the GMRF6 with ML parameter updates for μ_{SPH} and μ_{CYL} is detecting most of the spheres. The volume overestimation indeed is mostly beneath 300%, but with EMGMM and MLGM procedures the spheres get overestimated due to various outliers and are therefore accounting as not detected. The detectability is comparable to those of EMGMM and MLGM procedures with subsequent parameter assignment.

The same algorithms processed on VOIs comprising $14 \times 14 \times 40$ voxels in OSEM2d reconstructions

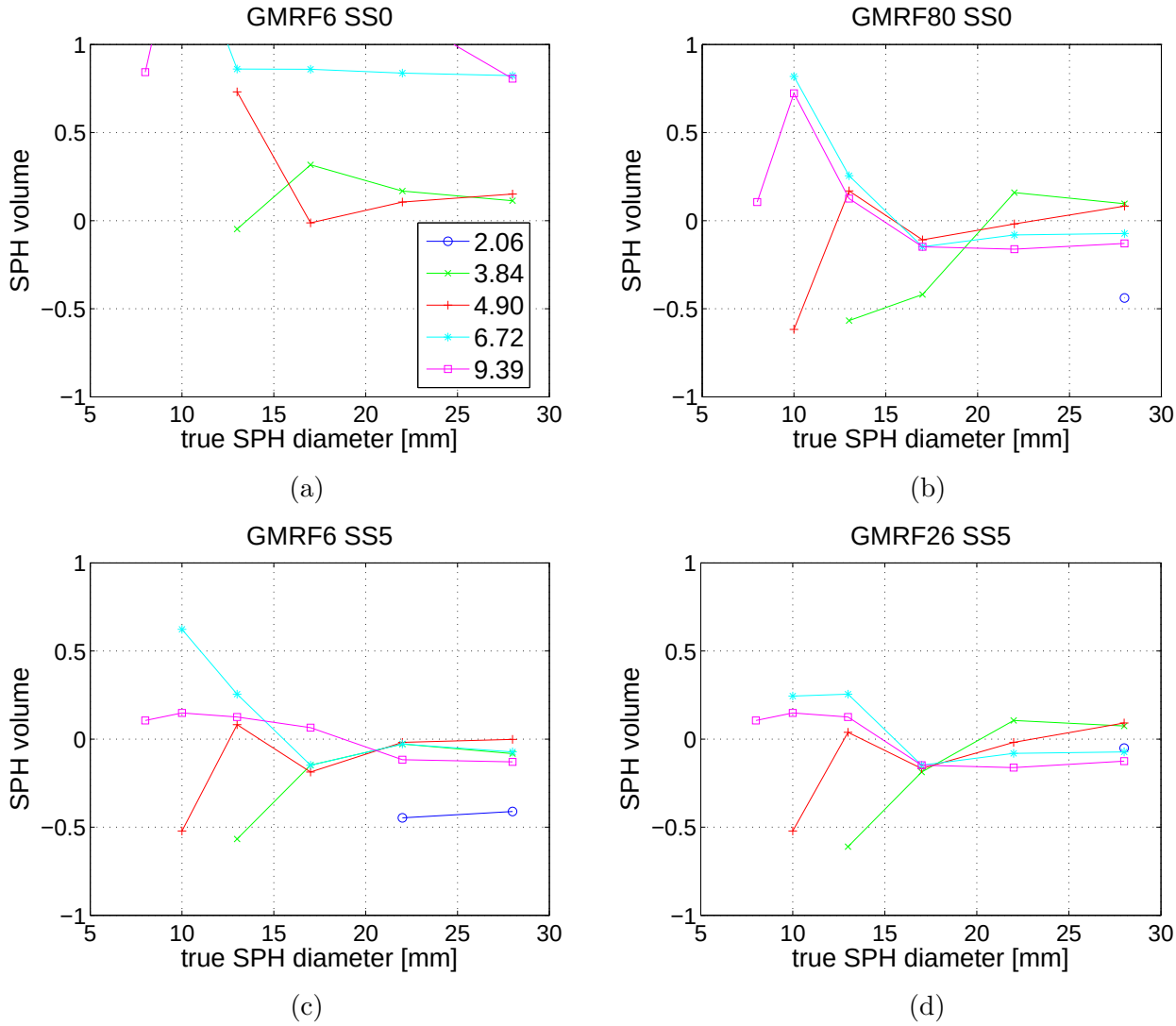


Figure 5.30: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by (a) GMRF6, (b) GMRF80, (c) GMRF6-4 and (d) GMRF26-4. All data plotted versus sphere diameter for the NEMA phantom reconstructed with BP. Each graph comprises curves for various SBR, connecting measurements for each sphere.

(figure 5.27(b) and figure 5.27(d)) shows that this solutions are more ore less stable regarding different sized VOIs (just one sphere lost at SBR=3.84).

At this point the usage of local interactions among the labels $\mathbf{Z}$ gets visible. An explanation for this is given by remembering the way parameter optimization is done. The statistics for marginal and pairwise labellings are calculated from the image and are mapped to the distribution parameters so that the expectations under the probability distribution are equal the empirical statistics (calculated from the image). In case of using the optimization procedures from section 4.4.3.1 where local joint distributions for pair voxels get analysed, the interaction parameters for an opposite labelling $\tilde{\alpha}_{\mathrm{CYL,SPH}} = \tilde{\alpha}_{\mathrm{SPH,CYL}}$ are equal. Trying to cluster small objects in background volume, however, the

Label	neighbourhood system
GMRF6	$\mathcal{N}_1$
GMRF18	$\mathcal{N}_1 + \mathcal{N}_2$
GMRF26	$\mathcal{N}_1 + \mathcal{N}_2 + \mathcal{N}_3$
GMRF80	$\mathcal{N}_1 + \mathcal{N}_2 + \mathcal{N}_3 + \mathcal{N}_4 + \mathcal{N}_5 + \mathcal{N}_6$

Table 5.6: Table of neighbourhood systems used with GMRF. The first column determines the tags whereby the second column depicts the neighbourhood systems in use.

surface of the sphere voxels will always show less voxels than its outer hull surface. Therefore their probability being a member of the sphere cluster get diminished leading to reduced volume overestimation and moreover leads to vanishing outliers. The elimination of outliers is responsible for achieving solutions accounting as detected even for small neighbourhood systems without subsequent parameter updates.

Increasing the neighbourhood as discussed in the previous subsection, the GMRF18 and GMRF26 algorithm show equivalent behaviour (detectability as well as volume estimates) as the GMRF80. Therefore they are not depicted. Having raised the neighbourhood to 80 voxels, GMRF80 shows that the model assumption of a GMRF do better fit to the data why no subsequent parameter steps are needed to fix problems of bad statistical ensembles. The results are comparable to those of the EMGMM and MAPMLGM procedure.

Applying subsequent parameter updates as during the previous algorithms, figure 5.28 and figure 5.29 are achieved for a subsequent parameter assignment GMRF-3 respectively GMRF-4. In both figures the left column presents solutions to VOIs of $14 \times 14 \times 20$ whereas the right column depicts the results estimated from $14 \times 14 \times 40$ to compare the dependencies on VOI sizes. It can be seen that the larger the neighbourhoods get the lower the volume estimates get. Moreover the results, volume estimation as well as detectability, are stable regarding the VOI size. It is emphasized that in case of GMRF-3 the four largest spheres are detected in small and large VOIs. Also two spheres of 8mm diameter are detected in both VOIs (3 spheres at larger VOIs). Therewith the GMRF has shown to be less sensitive to noise. Concerning the GMRF-4 update procedure figure 5.29, the solutions are comparable to those of EMGMM-4. In case of the GMRF approach however, the volume of larger spheres get underestimated.

Finally the mean field application of the GMRF is processed on BP reconstructed images of the NEMA phantom. The VOI size is chosen $14 \times 14 \times 20$. The results are shown in figure 5.30. Figures 5.30(a) and (b) depict the volume estimates of the smallest respectively the largest neighbourhood without parameter tuning. The same behaviour is achieved as for the OSEM2D reconstructions. Figure 5.30(c) and figure 5.30(d) visualize the GMRF-4 parameter assignment. As can be seen the volume estimates are stable regarding the image reconstruction. Moreover the difference of the estimation er-

ror is mostly far below 50%.

5.7.3 Post processing with GMRF

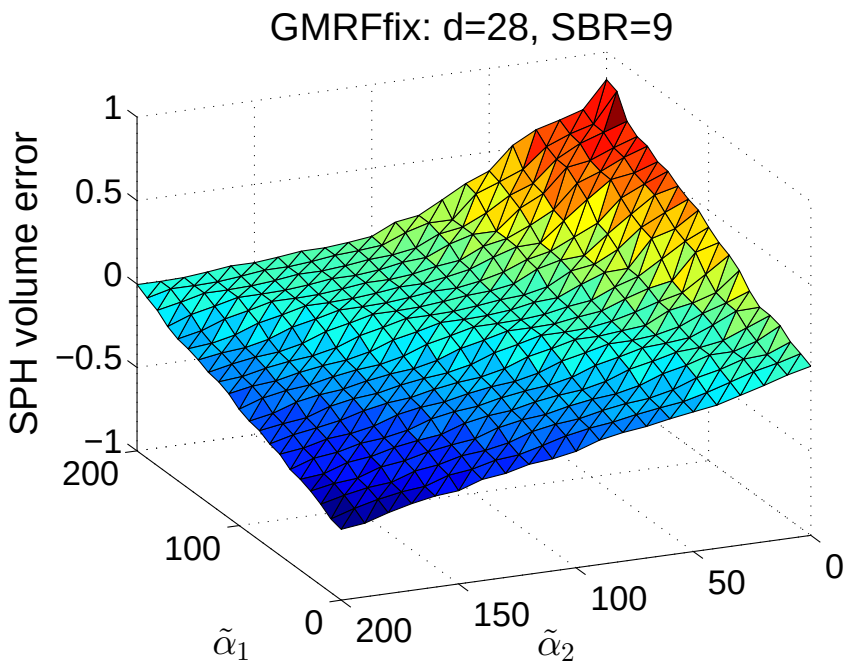


Figure 5.31: SPH volume error estimated with GMRFfix in a VOI comprising 14 voxels of an OSEM2D reconstructed image including the sphere with 28mm diameter at SBR=9, drawn versus the parameters $\tilde{\alpha}_1$ and $\tilde{\alpha}_2$.

A last approach using MRFs was published in [29]. The proposed algorithm consists of two consecutive steps: the coarse estimation step fits a basic model, yielding fairly good initial estimates. These estimates are then refined in the correction step.

With EMGMM-2 we already obtained good volume estimates with volume overestimation of small spheres. As has been seen in section 5.7.2, MRFs are reducing outliers and are further reducing the overestimation. Applying a mean field approach, the parameters $\tilde{\alpha}_{kl} = \tilde{\alpha}_{lk}, \forall k \neq l$ are considered to be equal. Clearly for spherical objects, having fewer voxels at the object boundary as its enclosing voxel cluster, the parameter $\tilde{\alpha}_{kl}$ as defined above leads to lowering the membership probabilities for boundary voxels of the sphere.

Hence the coarse estimation step is done using the EMGMM-2 procedure (see section 5.4.2). To improve the volume estimation of small spheres the correction step applies a corrective sampling (using the final labeling from EMGMM-2 as initial labeling) without re-estimating the parameters any further. We will call this algorithm GMRFfix. The MRF model for the correction step applies just the interaction parameters $\tilde{\boldsymbol{\alpha}}$ and omits the singleton term governed by the parameter α. We further restrict the matrix $\tilde{\boldsymbol{\alpha}}$ to exclude the interactions for equally labeled voxels $\tilde{\alpha}_{kl} = 0, \forall k = l$

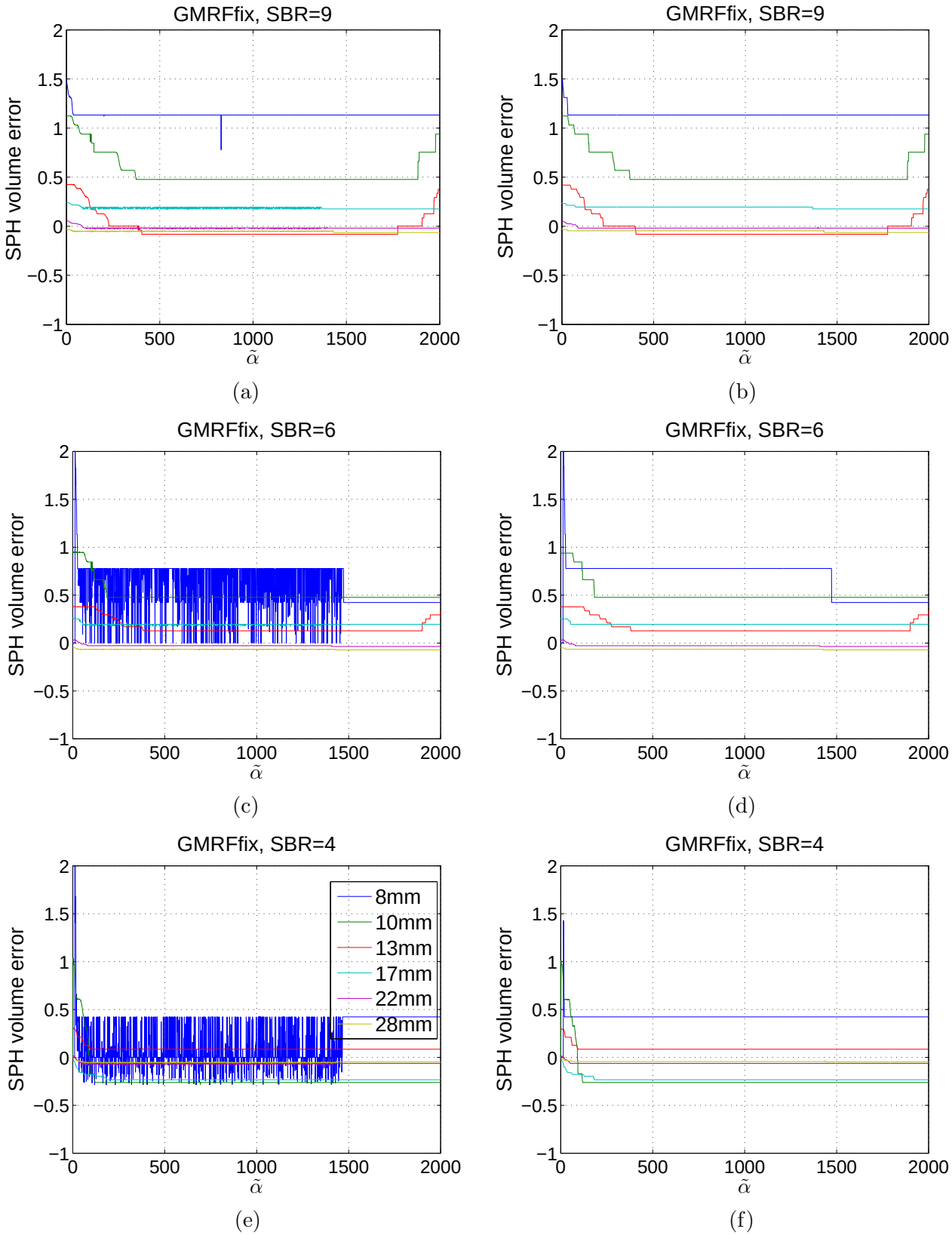


Figure 5.32: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels: (a) uniquely sampled SBR 9, (b) exponentially sampled SBR 9, (c) uniquely sampled SBR 6, (d) exponentially sampled SBR 6, (e) uniquely sampled SBR 4 and (f) exponentially sampled SBR 4. All data plotted versus parameter $|\tilde{\alpha}|$ for the NEMA phantom reconstructed with OSEM2d.

and hence acting just on boarder voxels. The MRF model for the correction step of the GMRFfix procedure can be written as

$$p(X,\mathbf{Z}) = \prod_k \prod_l \exp\left\{ \sum_{n\in\mathcal{V}} z_{nk}[\gamma_k x_n - \gamma'_k x_n^2 - A_n(\gamma,\gamma')] \right\} \exp\left\{ \sum_{(n,m)\in\mathcal{E}} z_{nk}\tilde{\alpha}_{kl} z_{ml} - A(\tilde{\alpha}) \right\}, \tag{5.29}$$

with

$$\tilde{\alpha}_{kl} = \tilde{\alpha}[1-\delta_{kl}]. \tag{5.30}$$

The parameters γ_k and γ'_k are adopted from the coarse estimation step by transforming the Gaussian parameters μ_k and σ_k via (4.79).

As already mentioned, the mean field approach assumes that $\tilde{\alpha}_{kl} = \tilde{\alpha}_{lk}, \forall k \neq l$. To verify that this assumption is indeed yielding good solutions, we have calculated volume estimates by sampling labellings for various parameter settings of $\tilde{\alpha}_{\mathrm{CYL,SPH}} \hat{=} \tilde{\alpha}_1$ and $\tilde{\alpha}_{\mathrm{SPH,SPH}} \hat{=} \tilde{\alpha}_2$. Figure 5.31 shows the relative volume estimates of the sphere with 28mm diameter at an SBR of 9 drawn over $\tilde{\alpha}_1$ and $\tilde{\alpha}_2$ for $\tilde{\alpha}_1, \tilde{\alpha}_2 \in [0, 200]$. It is seen that in a broad range the results are constant for $\tilde{\alpha}_1 = \tilde{\alpha}_2 + c$. Accurate solutions are obtained in case of $\tilde{\alpha}_1 = \tilde{\alpha}_2$. Hence the last term in (5.30) is justified and we can proceed to investigate the scalar $\tilde{\alpha}$.

For this purpose we again sample labeling configurations for various settings of $\tilde{\alpha}$. Figure 5.32 (a), (c) and (e) show the relative volume estimates of the NEMA spheres drawn over $\tilde{\alpha} \in [0, 2000]$ obtained with GMRFfix.

The sampling rate in contrary to the calculations done in figure 5.31 is increased from 10 to 1. Hence we can see that the volume overestimation is indeed decreasing, especially for small spheres, but is fluctuating and so there is no unique optimal value. In case of small SBRs and small volumes the fluctuations are increased changing between detecting and not detecting spheres.

To fix this problem, we implemented some changes to the Metropolis. As shown in section 4.4.2.2 the sampling procedure calculates the probability ratio of the current label and the changed label and compares this value against a uniquely distributed random number in the interval of zero and one, q. Hence not only labeling configurations are attained which are more probable. If a small random number q is generated it is possible that a label which is less probable than the current one is accepted. Now the idea is to decrease the acceptance barrier by generating random numbers which are not uniformly distributed to enhance the outcome of probable labellings.

The modified Metropolis sampler now proceeds as follows:
Initialize $\mathbf{Z}$ by using the EMGMM-2 procedure. With this initial labeling, feed the Metropolis sampler:

- Store the current label matrix $\mathbf{Z}$ and generate a new one, $\mathbf{Z}^{new}$, by flipping all states of the binary matrix.

- For all voxels, calculate

$$a_n = \frac{p(x_n \mid z_n^{new}) p(z_n^{new} \mid \mathcal{N}(z_n))}{p(x_n \mid z_n) p(z_n \mid \mathcal{N}(z_n))} \tag{5.31}$$

- For each voxel, sample a uniformly distributed random variable $q_n \in [0,1]$. If

$$q_n' = e^{-q_n} < a_n, \tag{5.32}$$

  accept the new value for $\mathbf{Z}$, otherwise reject it.

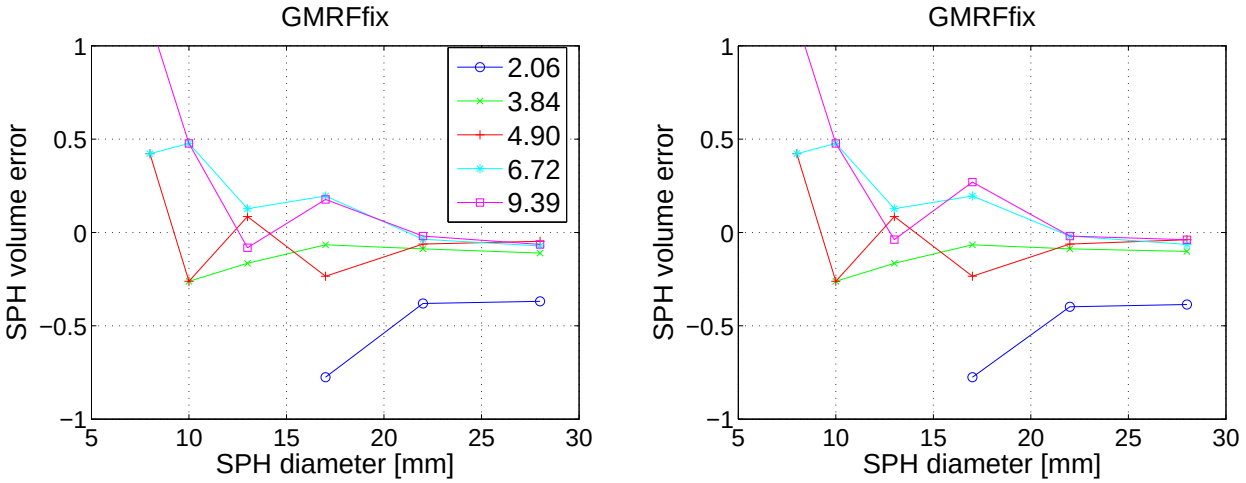
With (5.32), the mean of the rejection threshold $\mathbb{E}\{q'\}$ is raised from 0.5 to 0.6321 because

$$\mathbb{E}\{q'\} = \int_0^1 e^{-q} dq = 1 - \frac{1}{e}. \tag{5.33}$$

Moreover the minimum rejection threshold generated this way stays above $e^{-1} = 0.3679$. Hence we do not accept configurations which are less probable than 36% and in general raise the acceptance threshold for new labellings.

Results of the GMRFfix procedure with the modified Metropolis sampler are depicted in figure 5.32 (b), (d) and (f) for all NEMA spheres at the SBR of 4, 6 and 9. Especially for the smallest sphere of 8mm diameter the uncertainty about the existence of a sphere is reduced and yields stable results. Moreover it is seen that the optimal value for the interaction parameter $\tilde{\alpha}$ is 1500.

Using these considerations about the parameter setting for $\tilde{\alpha}$, the GMRFfix procedure is applied to VOIs of $14 \times 14 \times 20$ voxels (figure 5.33) (a) and to VOIs of $14 \times 14 \times 40$ voxels (figure 5.33) (a) of OSEM2D reconstructed NEMA phantom images. Beyond a diameter of 8mm the algorithm yields stable results. Moreover the smallest sphere at the SBRs of 9, 6 and 4 are detected as well as the three largest spheres at the smallest SBR.



(a) (b)

Figure 5.33: SPH volume error estimated in VOIs comprising (a) $14 \times 14 \times 20$ and (b) $14 \times 14 \times 40$ voxels by GMRFfix. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

5.7.4 Loopy Belief Propagation

The labeling procedure discussed in section 4.4.2.1 called beliefe propagation yields an unambiguous schedule in case of tree structured graphs. If we are faced with loopy graphs, various schedules can be used. Two extreme cases are presented in figure 4.15 and figure 4.16 which are called a flooding schedule respectively zig-zag schedule. Either way, messages are sent all over the graph propagating information from each voxel to all the others. The basic model for which the marginalization procedure is applied is again given by (4.75).

At first we investigate in alternating between a parameter estimation step (m-step), as described in section 4.4.1 for the Gaussian parameters and section 4.4.3 for the Potts model parameters, and a label estimation step (e-step) as shown in section 4.4.2.1. Disregarding the labeling queue the algorithm performs bad with tremendous underestimation and loss of spheres volumes.

Two refinement steps do not correct for this misclassification. The first method includes a clipping value for the parameter estimation, meaning that the mean and the standard deviation of X is calculated using just voxels having a membership probability greater or equal 0.99 for a special cluster. This approach should enhance the accuracy of the parameter estimates due to excluding voxels which are not unambiguous identifiable with a certain object. Moreover, assuming μ and σ as variable nodes in the graph having edges with every data node, many short loops are obtained. To counteract these feedbacks, for every node $x_i \forall i \in 1, ..., N$ the mean and the standard deviation get recalculated as

$$\mu_{ik} = \frac{\mu_{k,\mathrm{ML}} \sum_n z_{nk} - x_i z_{ik}}{\sum_{n \sim i} z_{nk}} \qquad \sigma^2_{ik} = \frac{\sigma^2_{k,\mathrm{ML}} \sum_n z_{nk} - (x_i - \mu_{k,\mathrm{ML}})^2 z_{ik}}{\sum_{n \sim i} z_{nk}} \tag{5.34}$$

with $\mu_{k,\mathrm{ML}}$ and $\sigma_{k,\mathrm{ML}}$ labeling the maximum likelihood estimates of the mean and variance of the whole data.

The last attempt to benefit from marginalization procedures is to break up the main queue iterating an m-step and an e-step and using belief propagation in a post correction step as done in the previous section with GMRFfix. Even in this case the algorithm is very sensitive regarding the parameters α

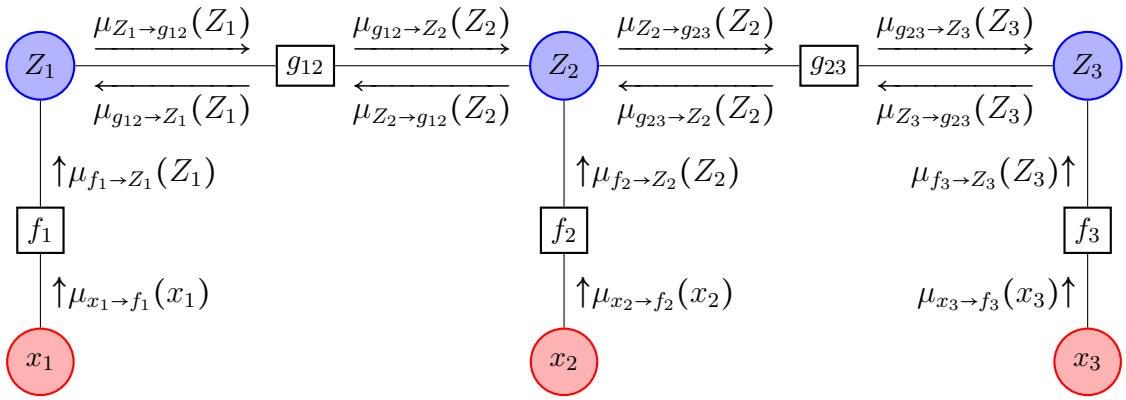


Figure 5.34: One-dimensional factor graph representation of the GMRF presented in (4.75) without the use of the singleton parameter α.

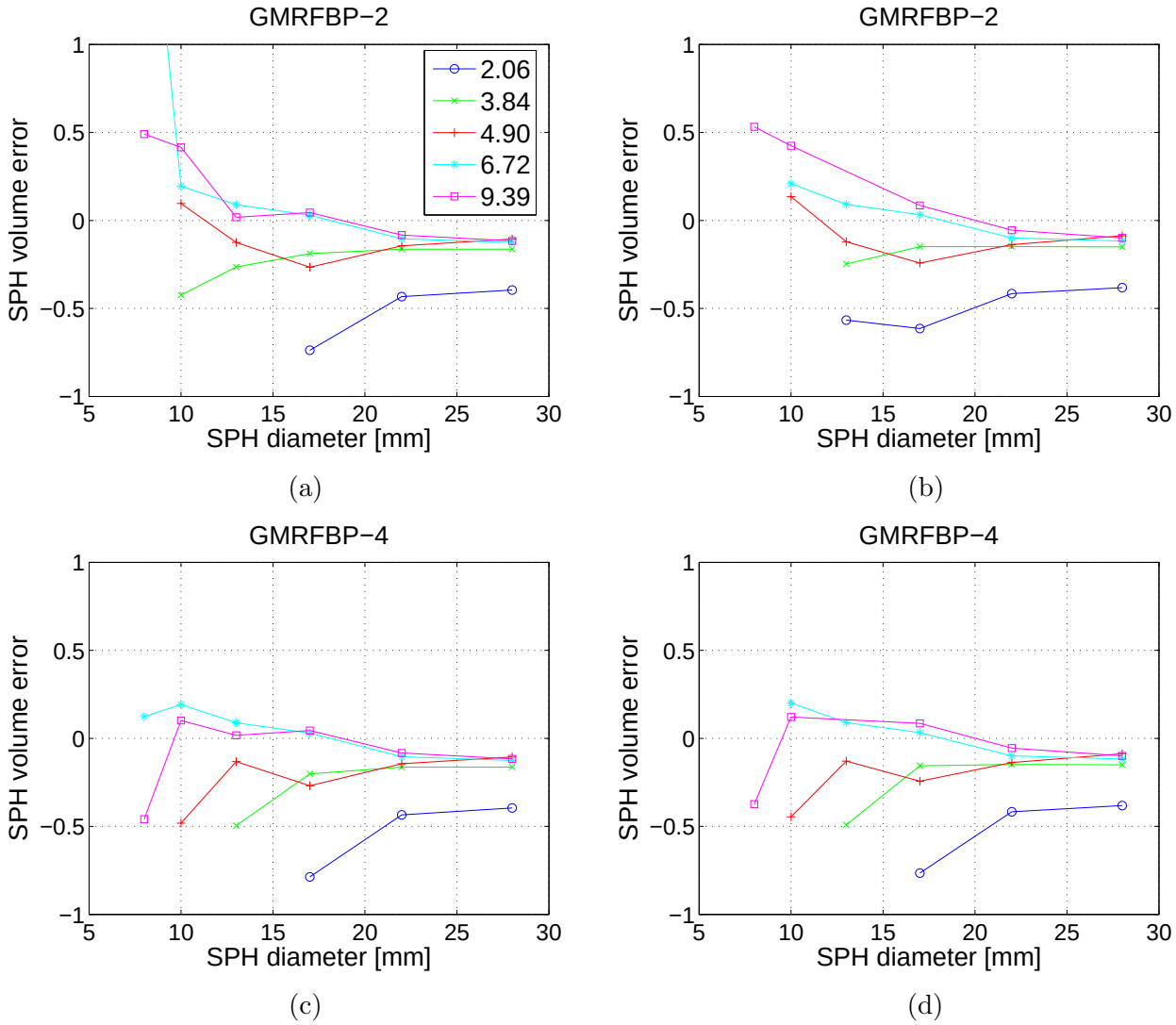


Figure 5.35: SPH volume error estimated in VOIs comprising $14 \times 14 \times 20$ voxels by (a) GMRFBP-2 and (c) GMRFBP-4; SPH volume error estimated in VOIs comprising $14 \times 14 \times 40$ voxels by (b) GMRFBP-2 and (d) GMRFBP-4. All data plotted versus sphere diameter for the NEMA phantom reconstructed with OSEM2d. Each graph comprises curves for various SBR, connecting measurements for each sphere.

and $\tilde{\alpha}$. Initializing the procedure using the EMGMM-2 and applying the mean field approach (see section 4.4.3.1) for the estimation of α as well as $\tilde{\alpha}$ in a post defined correction step, the spheres get overestimated until they vanish.

In contrast using the pseudo likelihood optimization from section 4.4.3.2 (for which it is assumed that the basic model do not incorporate singleton terms for the label nodes) and hence reduce the factor graph pendant described in figure 4.14 to the factor graph pendant shown in figure 5.34, results are drawn for VOIs of $14 \times 14 \times 20$ voxels of OSEM2D reconstructed NEMA phantom images.

6

Conclusions

THIS chapter is devoted to summarizing the benefits of this work and to giving an outlook of what could be potential next steps.

The aim of this work was to investigate methods which are able to improve clinical state-of-the-art methods for image segmentation for positron emission tomography. Although iterative threshold methods seem to perform reasonably well (for SBR from 3.84 upwards and for spheres greater than 10mm) as shown in section 5.3.2 one has to keep in mind that they have to be adjusted to every device. The inclusion of a 8mm sphere and a measurement of 2.06 SBR to the analysis was not found in literature and seems to be a limiting case for the presented problem of image clustering in PET, where a scanner with resolution of $4 \times 4 \times 3$mm is in use and a diameter of 8mm is just double the voxel size.

First we give an overview of the ability to detect the given spheres (section 6.1.1), followed by a discussion of the volume estimation in section 6.1.2. In section 6.1.3 the clustering results are analysed in more detail showing labeling matrices estimated by the algorithms. To compare the clinical state-of-the-art methods to the proposed algorithms, the results are averaged either over the sphere diameter or the SBR. Therewith, various algorithms can be depicted inside the same graph enhancing the visibility of specific benefits or shortcomings.

As already mentioned in section 5.2, discarding solutions which comprise outliers seems a bit pessimistic. Moreover, for different circumstances/questions different algorithms perform well. So currently, a software framework should comprise various algorithms which have to be applied in a specific order to extract different informations.

6.1 Discussion

6.1.1 Detectability of Spheres

As mentioned in the introduction of this chapter we initially want to give an overview of the detection behaviour of various algorithms. The results have shown (see chapter 5) that the detection statistics can be improved if we process small VOIs of $14 \times 14 \times 20$ voxels. The spheres measured at low SBR are excluded from the detection statistics if we process larger VOIs. Exceptions from this perception

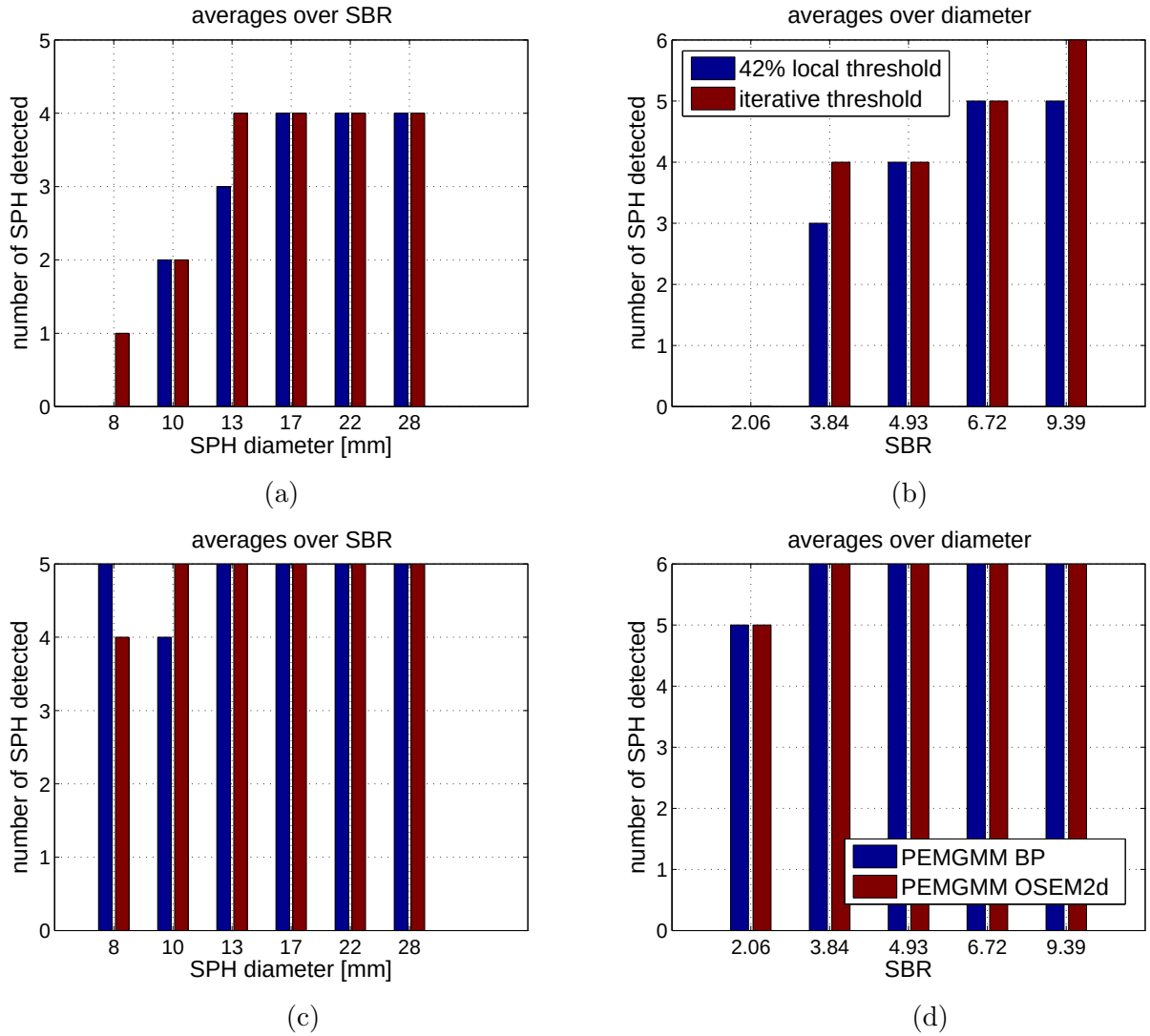
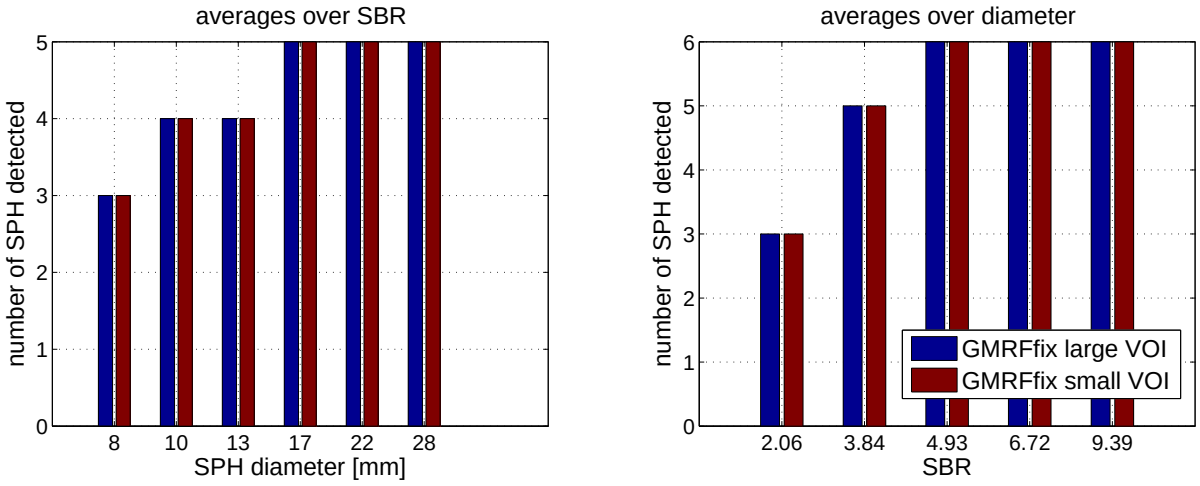


Figure 6.1: Detected spheres averaged over the SBR (a) and (c); detected spheres averaged over the diameter (b) and (d). Results obtained by threshold methods (a) and (b); results obtained by PEMGMM (c) and (d). The results for the threshold methods are achieved using OSEM2D reconstructed NEMA phantom VOIs of size $14 \times 14 \times 20$. The results for PEMGMM are achieved using OSEM2D and BP reconstructed NEMA phantom VOIs of size $14 \times 14 \times 20$.



(a) (b)

Figure 6.2: Detected spheres averaged over the (a) SBR and (b) diameter. Results obtained by GMRFfix in OSEM2D reconstructed NEMA phantom VOIs of size $14 \times 14 \times 20$ and $14 \times 14 \times 40$.

are given by the GMRF-3 methods shown in section 5.7.2 and especially by the method shown in section 5.7.3, which apply the MRF just via a post defined correction step.

According to the results from section 5.3.1, local percentage threshold methods are doing a bad job as can be seen from figure 6.1 (a) and (b). Their ability of detecting spheres do not include spheres of SBRs lower than 3.84 and diameters lower than 10mm. Even the sphere of 10 mm diameter is mostly detected just at an SBR of 9.39. Employing the iterative thresholding method (see figure 6.1 (a) and (b)), the detectability is increased by two spheres (see also table 5.3) including the one of 8mm diameter at SBR=9.39 and the sphere of 10mm diameter at a SBR of 6.72, but no solutions to the SBR measurement lower than 3.84 can be given. Solutions to the lowest contrast measurements can not be provided due to the noisiness of the images which, if using cutoff values, includes various outliers.

Applying Bayesian methods including prior probabilities is most beneficial for detecting spheres see figure 6.1 (c) and (d). Applying the PEMGMM algorithm to small VOIs ($14 \times 14 \times 20$) leads to detecting all spheres but one of the smallest at the lowest SBR. It is therefore recommended to initially run one of them to detect spheres in a first step and determine the volume in a post-processing step. As seen in figure 6.1 (c) and (d), the results are stable regarding the reconstruction algorithm

To emphasize the ability of dealing with low SBRs consider a human example for which the algorithms are aimed to be developed. The liver is one of the organs responsible for the reduction of various substances and is accumulating much of the radioactive tracers before they get rejected. Therefore PET scans of humans with cancerous tissue located inside the liver show low contrasts.

Moreover inspecting figure 6.2 (a) and (b) it is shown that the GMRFfix algorithm, which apply a MRF just in a post-defined correction step yields stable detection statistics for small and large VOIs.

6.1.2 Estimation Error

In contradiction to the detection behaviour presented in the last section, the volume estimates obtained in the larger VOIs of 14×14×40 voxels are more accurate than the volume estimates achieved in smaller VOIs of $14 \times 14 \times 20$ voxels.

In figure 6.3 the averages of the volume error are depicted against the SBRs respectively against the sphere diameter for the 42% threshold method and the iterative threshold method as well as for the EMGMM-4 and the MLGMGC-4 approach in large VOIs. As known from chapter 5, an ad hoc parameter assignment according to 4 (see table 5.4) yield the most accurate solutions.

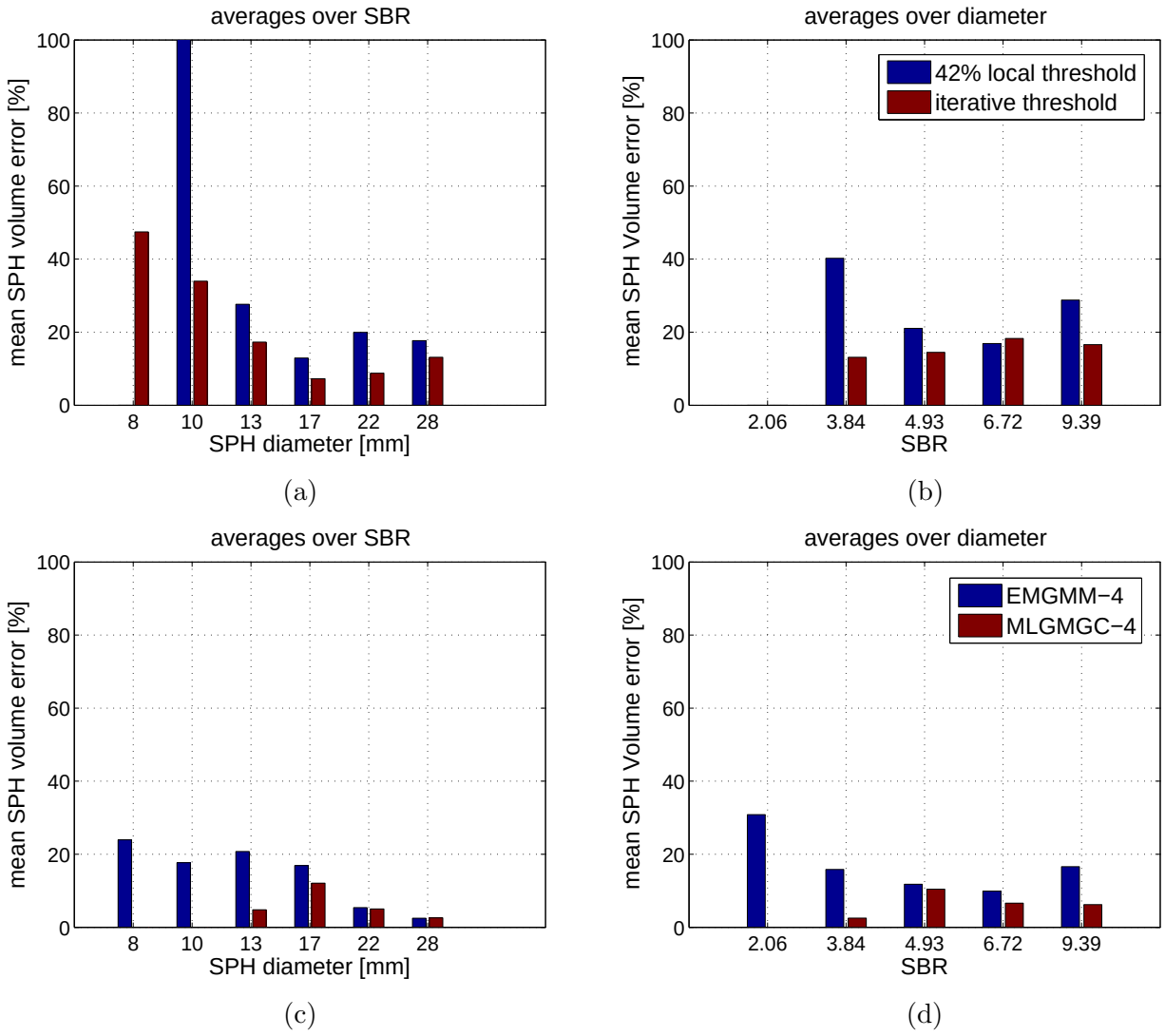
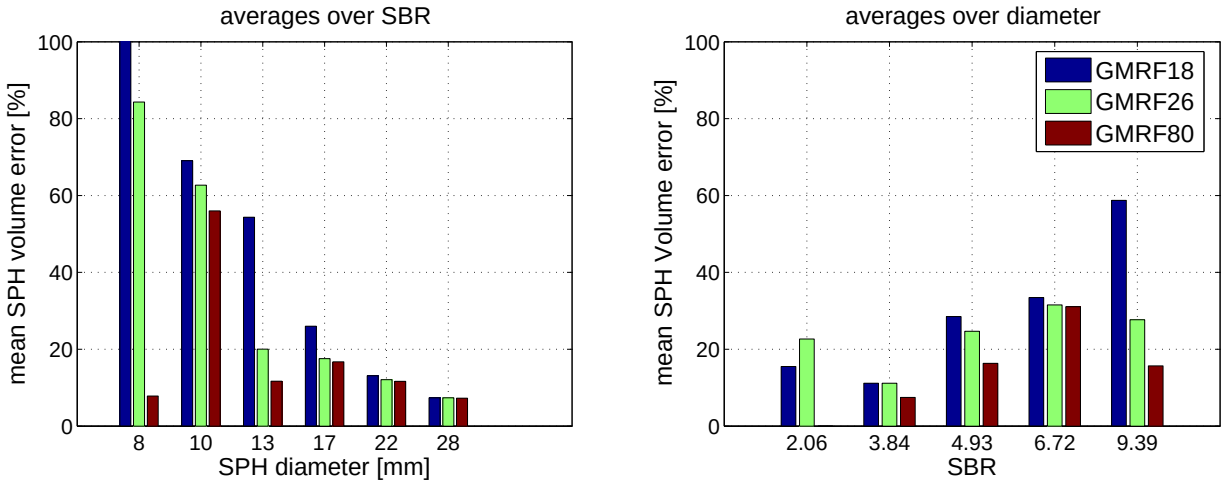


Figure 6.3: SPH volume error averaged over the SBR (a) and (c); SPH volume error averaged over the diameter (b) and (d). Results obtained by threshold methods (a) and (b); results obtained by EMGMM-4 and MLGMGC-4 (c) and (d). The results are achieved using OSEM2D reconstructed NEMA phantom VOIs of size $14 \times 14 \times 40$.



(a) (b)

Figure 6.4: SPH Volume error averaged over (a) SBR and (b) diameter. Results obtained by GMRF18, GMRF26 and GMRF80 using OSEM2d reconstructed NEMA phantom VOIs of $14 \times 14 \times 40$.

As with the ability of detecting spheres, local percentage threshold methods are doing a bad job regarding their volume predictability. Their SBR dependent diversification of the volume overestimation is huge. The accuracy of the volume estimates given by iterative thresholding can bee improved using EMGMM-4. This fact is revealed by the graphs showing averages over the SBR (figure 6.3(c)) as well as by graphs showing averages over the sphere diameter (figure 6.3(d)). Considering the volume estimates of the MLGMGC-4 algorithm, the accuracy can further be improved at the price of loosing the two smallest spheres of 8mm and 10mm diameter. Moreover including correlations among the image data leads to a larger sensitivity regarding noise and therefore to undetected spheres at SBRs of 2.06.

To address the comparison of the threshold methods with the GMRF approaches, figure 6.4 and figure 6.5 depicts the algorithms without parameter tuning respectively with ad hoc 4 substitution. Starting with the analysis of the ML parameter estimation (see figure 6.4), meaning that mean and standard deviation get estimated from the according clusters, it is seen that GMRF80 (largest neighbourhood) yields already good clustering results. Although it is not outperforming the EMGMM-4 results, it can perfectly compete with iterative thresholding. Due to no need for regularizing the parameters it is assumed that the model assumptions of the specified GMRF are more realistic than the GM and GMM used during the previous procedures. Therefore it is further assumed that it offers the potential to be corrected in further investigations (see section 6.2).

Considering the volume estimates done by the GMRF26-4 algorithm (see figure 6.5) it is seen, that it topples the EMGMM-4 approach regarding the volume errors in limiting cases of small spheres and low SBRs. Moreover the GMRF26-4 approach is not only stable regarding the VOI size (see figure 6.4 (a) and (b)), its variablity regarding the reconstruction algorithm stays beneath 10% (see figure 6.4 (c) and (d)).

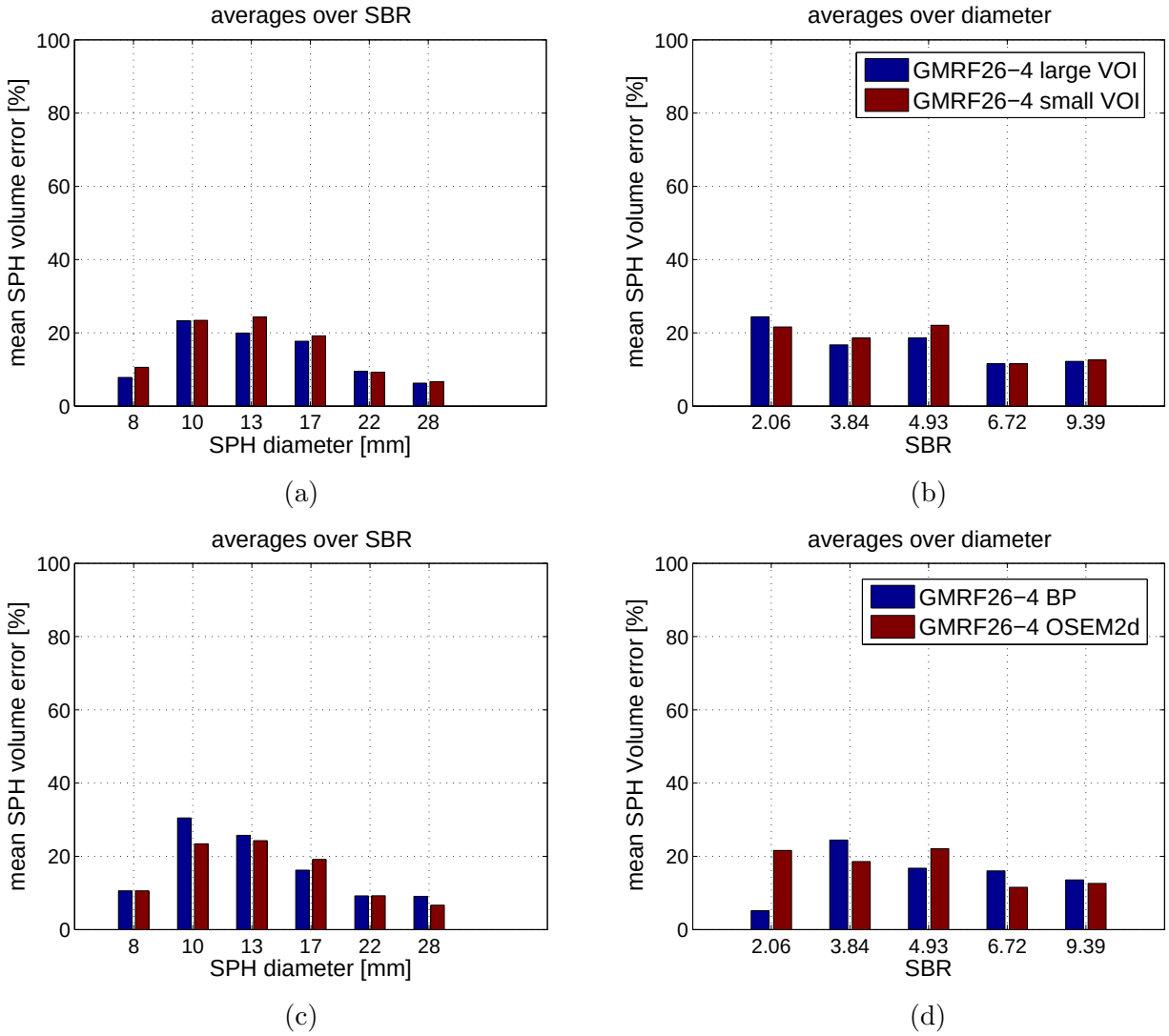


Figure 6.5: SPH Volume error averaged over the SBR (a) and (c); SPH Volume error averaged over the diameter (b) and (d). Results obtained by GMRF26-4 using OSEM2d reconstructed NEMA phantom VOIs of $14 \times 14 \times 20$ and $14 \times 14 \times 40$ voxels (a) and (b); results obtained by GMRF26-4 using OSEM2d and BP reconstructed NEMA phantom VOIs of $14 \times 14 \times 20$ voxels (c) and (d).

Including correlations among the voxels of the images was not beneficial for the aim of PET clustering. A better way to incorporate dependencies among voxels was given by influencing the label matrix. This procedures have shown to be more stable regarding SBR variations than the solutions achieved without inclusion of correlations.

Finally, some remarkable results are obtained analysing the methods which uses MRFs just as a post-defined correction step. As shown in chapter 6.1.1, the GMRFfix algorithm has stable sphere detectability regarding the VOI size. This is almost true for the GMRFBP-4 which employs belief propagation. Similar results are obtained for the volume errors which are shown in figure 6.6. Beside the fact that they are yielding good volume estimates, their solutions are stable regarding the VOI

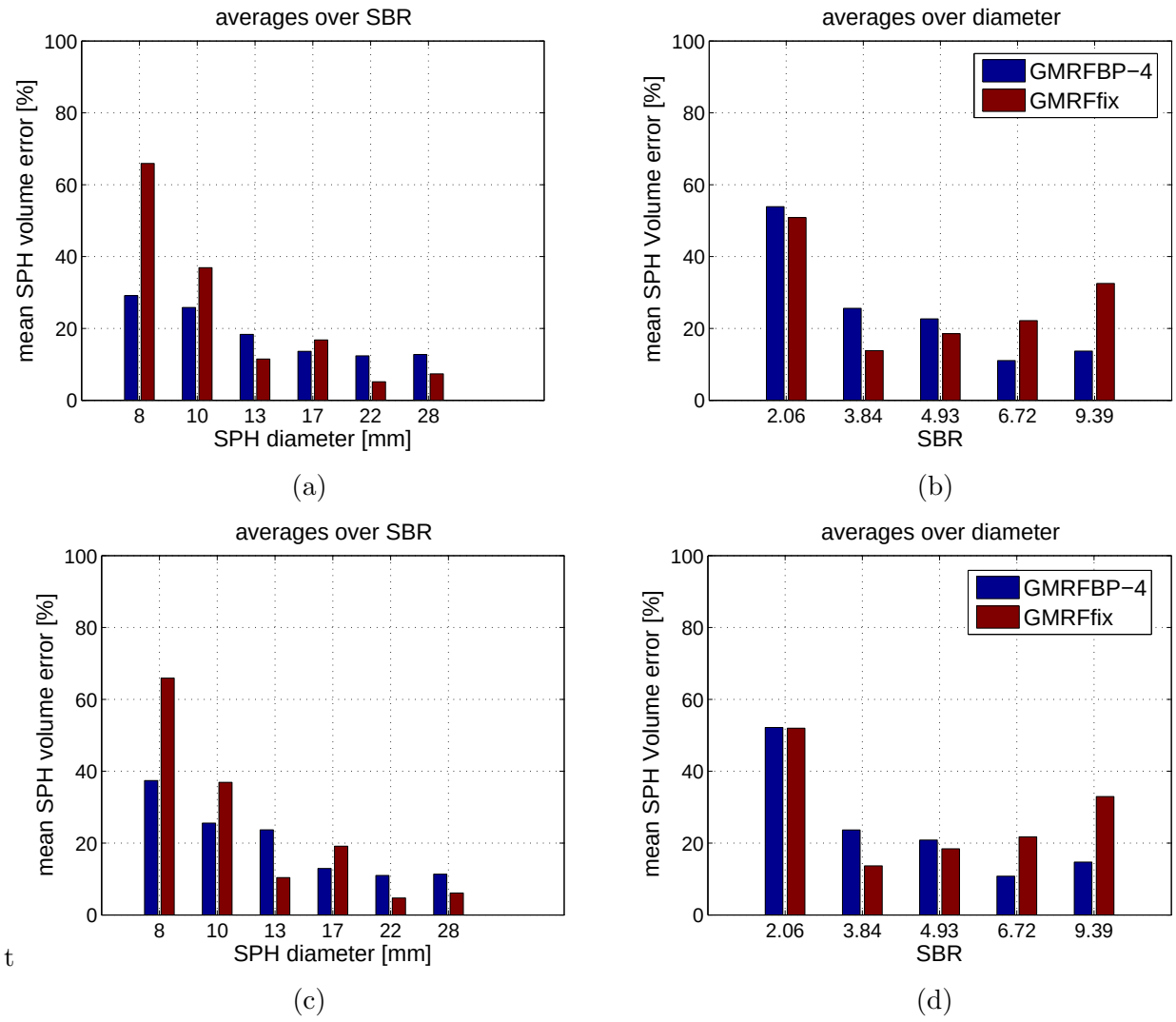
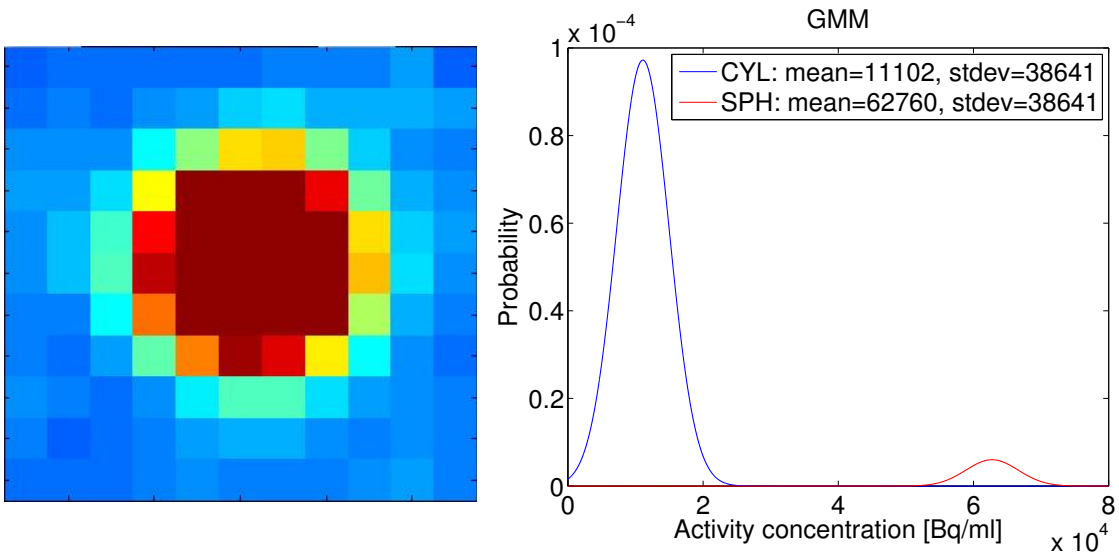


Figure 6.6: SPH Volume error averaged over the SBR (a) and (c); SPH volume error averaged over the diameter (b) and (d). Results obtained by GMRFBP-4 and GMRFfix using OSEM2D reconstructed NEMA phantom VOIs of size $14 \times 14 \times 20$ (a) and (b). Results obtained by the GMRFBP-4 and GMRFfix using OSEM2D reconstructed NEMA phantom VOIs of size $14 \times 14 \times 40$ (c) and (d).

size.

6.1.3 Labellings

One of the main causes for using probabilistic models is the inherent treatment of PVE due to the formulation of membership probabilities for each cluster (sphere or cylinder). So every voxel gets assigned a probability vector which in general is allowed to carry values in the interval between zero and one. As argued already, threshold methods in general are producing a discrete labeling. In case of using a label matrix as done during the proposed probabilistic models this values are binary (0 and



(a) (b)

Figure 6.7: (a) transversal slice of the NEMA phantom reconstructions showing the 28mm sphere at a SBR of 9.39 of 14×14 voxels. (b) probability distributions of a two component GMM. The blue bell curve describes the probability distribution of the cylinder cluster and red one for the sphere cluster.

1). From this fact it is clear that accurate segmentation results which have to account for PVE are not possible even if the predicted volume is equal to the true volume of the object under consideration.

To illustrate the labelling values of certain segmentation results, $\mathbf{Z}_{\text{SPH}}$ will be shown for the OSEM2D reconstruction of the NEMA phantom shown in figure 6.7 (a). Figure 6.7 (a) shows a slice of the 28mm sphere measured at SBR of 9.39. Applying iterative thresholding to this slice the following label matrix is achieved

$$\mathrm{Z}^{(\text{iter Thresh})} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.1}$$

It is obvious that this discrete segmentation result is not representative for a circle and does not deal with PVE. Applying the EMGMM-4 procedure the label matrix in (6.2) is obtained. This matrix shows that the same voxels are involved labeling the circle in figure 6.7. But with EMGMM– 5 the

result is a better fit to the form of a circle. Four voxels at the border are not assigned binary values but do instead carry values between zero and one. Important to note is that the expected smooth transition at the sphere border is not given in such extent as it would be desired. Just 4 boarder voxels are labelled continuous and 8 boarder voxels are labelled discrete. In fact there are slices where this relation is even worse.

$$\mathrm{Z}^{(\text{EMGMM-5})} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0.4630 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.2772 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.9993 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0.9426 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.2}$$

To understand this, one has to take a look at figure 6.7 (b). This graph shows the Gaussian mixtured distributions using the final mean and standard deviation achieved with EMGMM-5. Each Gaussian component is multiplied by the according prior probability τ_k.

The assignment of the cylinder standard deviation to the sphere standard deviation generates two bell curves with same width. But due to large SBR a gap exists where both distributions are nearly zero. Only voxel values (activity concentration) residing within this gap lead to nonbinary labelling values. But with activity concentrations located at the rise of one bell curve, the labelling values snaps to the binary values zero or one. This phenomenon of overconfident labellings even remains at lower SBR. Even approaches incorporating covariances (EMGMMC and MLGMC) are not able to circumvent binary labelling at the clustering boarders.

A label matrix of the slice depicted in figure 6.7 (a) achieved with the variational Bayesian approach VBGMM is shown in (6.3).

$$\mathrm{Z}^{(\text{VBGMM-5})} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0.0289 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0136 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.9703 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.9990 & 0.3261 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.3}$$

Here the behaviour is even worse because the nonbinary values are closer to zero or one. With GMRF methods the discretization is further enhanced leading moreover to the loss of border voxels which where still contributing to the volume estimates of EMGMM and MLGM like procedures, see (6.4). This explains the volume estimation achieved during the application of MRFs to the presented problem.

$$\mathrm{Z}^{(\text{GMRF6-5})} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.4}$$

6.2 Outlook

There is a number of possibilties to further advance the image clustering procedures for PET presented in this work. Some ideas have already been initially studied. Recall that we are discarding all solutions comprising outliers as discussed in section 5.2. An improvement here would be to apply the corresponding algorithm in a second step to all objects that where found initially, by considering each object in a separate VOI. This method was seen to yield better detection behaviour, but with the disadvantage of detecting more than the six spheres in case of high SBR. To address the shortcomings discussed in section 6.1.3, we propose in section 6.2.1 conditional random fields (CRF) which are discriminative models offering a way of dealing with more complex models. To further overcome the already mentioned lack of knowledge of ground truth in real data sets, section 6.2.2 is presenting a software package enabling the numerical simulations of medical imaging.

6.2.1 Conditional Random Fields

As mentioned in the introduction of chapter 4, there are two possibilities of formulating an unsupervised statistical clustering problem (using parametrized probability distributions) for a given data vector X and its label matrix $\mathbf{Z}$. Either we establish a generative model represented by a joint probability $p(\mathbf{Z}, X)$ or we facilitate a discriminative model, i.e., a posterior distribution $p(\mathbf{Z} \mid X)$. Generative models have been presented in section 4.1 by establishing a conditional probability distribution $p(X \mid \mathbf{Z})$ which was reformulated via Bayesian theorem to gain a discriminative model. All the remaining models were formulated as generative models.

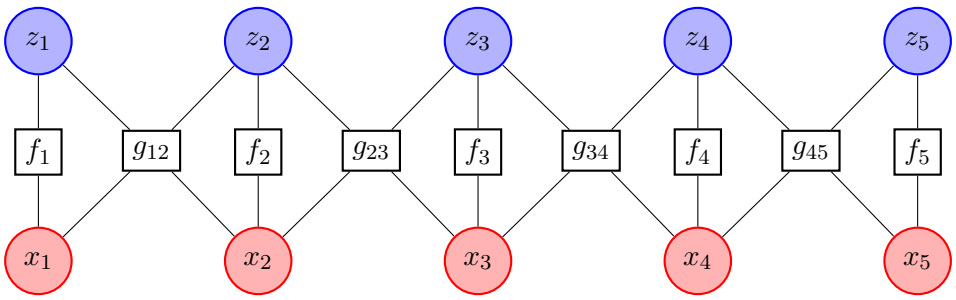


Figure 6.8: Linearized factor graph representation of the CRF presented in (6.6).

With generative models we run into troubles at least if we want to incorporate correlations among the data. E.g., a disadvantage of the MRF presented in the chapter 4.4 is that it is difficult to incorporate features over regions of the data vector X because $p(X \mid \mathbf{Z})$ would have complex structure. Specially the partition function of such model would require the integration regarding the image data X. For the partition function for the Gaussian part of the MRF in (4.75), we circumvented to calculate the normalization in a direct fashion by employing a factor comparison. For a more sophisticated model this is no longer feasible. A discriminative approach, which directly models the posterior distribution of $\mathbf{Z}$ given X, provides a way to address this difficulty.

A general definition of a CRF was given by [44] (page 291). Let G be a factor graph over X and $\mathbf{Z}$. Then (X,$\mathbf{Z}$) is a conditional random field if the distribution $p(\mathbf{Z} \mid X)$ factorizes according to G. With this definition the probability distribution of a CRF can be written as

$$p(\mathbf{Z} \mid X) = \frac{1}{Z(X)} \prod_{l=1}^{L} \phi_l(\mathcal{C}_l(X_l', \mathbf{Z}_l')), \tag{6.5}$$

where the functions ϕ_l correspond to the factors in the factor graph G (which are comparable to the potential functions introduced earlier in (3.55)) depending on cliques, X_l' and $\mathbf{Z}_l'$, as defined in section 3.4.1. But in contrast to a generative model, where the joint probability distribution is a function of X and $\mathbf{Z}$, the posterior distribution of the CRF shown in (6.5) is only a function of the label matrix and hence no integration has to be performed to evaluate the partition function. The advantage thus is that the partition function of the CRF may be computable whereas the partition function of a similar MRF may not.

Investigating in a model useful for our image segmentation task we want to address the problems discussed in section 6.1.3 where it was stated that the segmentation results obtained with GMRF are of discrete nature. To weaken the characteristics of the neighbourhood interactions we use a CRF as

$$p(\mathbf{Z} \mid X) \propto \prod_k \prod_l \exp\Big\{ \sum_{n \in \mathcal{V}} \Theta_k f_k(\mathbf{z}_n, x_n) + \sum_{(n,m) \in \mathcal{E}} \tilde{\alpha}_{kl} g_{kl}(\mathbf{z}_n, \mathbf{z}_m, x_n, x_m) \Big\}. \tag{6.6}$$

Contrary to (4.75), the second term in the exponent of (6.6) is incorporating interactions among voxels not only due to the label matrix but also due to the data vector. A factor graph representation of (6.6) is depicted in figure 6.8. Getting more precise about the structure of the function g_{kl}, we want

to incorporate the data vector to smooth the discrete behaviour of the interaction term,

$$\tilde{\alpha}_{kl} g_{kl}(\mathbf{z}_n, \mathbf{z}_m, x_n, x_m) = \tilde{\alpha}_{kl} z_{nk} z_{ml} \exp\big\{ -\beta (x_n - x_m)^2 \big\}. \tag{6.7}$$

Assuming again that the measured activity is basically Gaussian distributed, a Gaussian CRF is established according to

$$p(\mathbf{Z} \mid X) \propto \prod_k \prod_l \exp\Big\{ \sum_{n \in \mathcal{V}} z_{nk} [\gamma_k x_n - \gamma'_k x_n^2] + \sum_{(n,m) \in \mathcal{E}} \tilde{\alpha}_{kl} z_{nk} z_{ml} \exp\big\{ -\beta (x_n - x_m)^2 \big\} \Big\}. \tag{6.8}$$

To normalize this CRF would demand to evaluate the cumulant generating function $A(X, \gamma, \gamma', \alpha, \beta)$. An approximation can be implemented by normalizing each term in (6.8) individually by $A(X, \gamma, \gamma')$ and $A(X, \tilde{\alpha})$. Hence the Gaussian part can again be normalized by a factor comparison as shown in section 4.4.1. The parameter β can be chosen as the mean squared difference of neighbouring voxel data

$$\beta = \frac{\sum_{(n,m) \in \mathcal{E}} (x_n - x_m)^2}{|\mathcal{E}|}. \tag{6.9}$$

Although we have circumvented the need for integrating out the data X, the main difficulty with such approaches is to evaluate the partition function for the interaction term. At least for the parameter update step this is necessary. As shown in section 4.4.3.1 and section 4.4.3.2 we solved this problem by concerning just local probability functions and adapt them to fulfil global empirical statistics. This is no longer feasible using a model as (6.8).

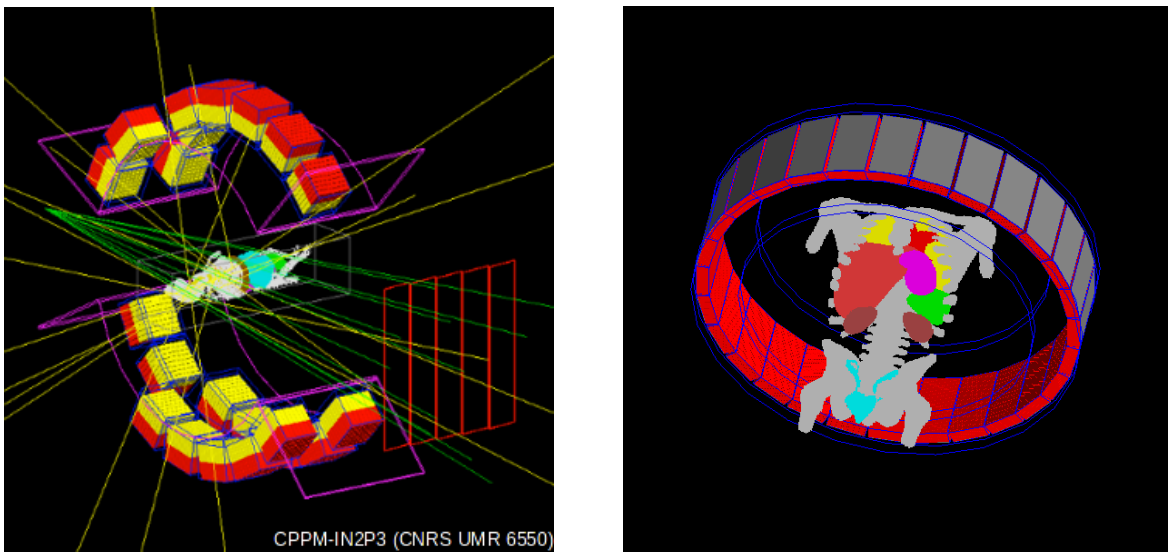
A way out of this difficulties are stochastic approximations such as the Mont Carlo sampler used for labeling MRFs. Instead of evaluating the probability functions and its derivatives analytically, we can resort sampling labeling configurations and adjust the parameters to end up with labels corresponding to the actual empirical statistics.

6.2.2 OpenGATE Simulations

A main problem we have to deal with is that the proposed algorithms cannot be verified using real patient data due to the lack of knowledge of the true size of the tumors. As mentioned in the introduction of chapter 5, it is not feasible to gain this information accurately via post-surgery investigations. Therefore we have performed NEMA phantom measurements where the geometrical ground truth as well as the accurate activity distributions are well known. Unfortunately our data set is not comprehensive and therefore the application of learning models is not possible.

Moreover a discussion was raised about the meaningfulness of the NEMA phantom measurements because the spheres are made up by acrylic glass of 1mm and hence the sphere borders represent areas of no accumulated activity. In the context of modelling humans with cancerous disease this is clearly not an optimal equivalent.

To address the problem of missing data with known characteristics, the international OpenGATE collaboration provides an open source software framework for numerical simulations in medical imaging. It is based on the more general framework GEANT4 which was developed by CERN for the



(a) (b)

Figure 6.9: PET systems modeled with GATE software. The pictures are taken from the OpenGATE homepage http://www.opengatecollaboration.org.

simulation of particle transitions through matter. It allows a user to model detectors as well as phantoms and to define certain sophisticated arrangements of experiments via a simple scripting language. It further provides various options for visualizing the geometry of experiments and is also able to the track particles. In figure 6.9 (a) and (b) two detector geometries are depicted including phantoms modeled with the OpenGATE software. The pictures are taken from the home page of OpenGATE http://www.opengatecollaboration.org.

Appendices

A

Probability Distributions

A.1 Bernoulli Distribution

In probability theory and statistics a single binary random variable is described by the Bernoulli distribution. It is a discrete probability distribution, which takes value 1 with success probability μ and value 0 with failure probability $1-\mu$ (e.g. a coin toss). With $0 \leq \mu \leq 1$ it is $p(z=1 \mid \mu) = \mu$ and $p(z=0 \mid \mu) = 1-\mu$ which can also be expressed in the form

$$p(z \mid \mu) = \mu^z (1-\mu)^{1-z} \qquad \forall z \in \{0,1\}, \tag{A.1}$$

with mean and variance (see (3.4) in section 3.1.1)

$$\mathbb{E}\{z\} = \mu \tag{A.2}$$

$$\operatorname{var}\{z\} = \mu(1-\mu). \tag{A.3}$$

Let Z be a binary random vector comprising N identical and independent Bernoulli distributed observations of a Bernoulli experiment (Bernoulli process) $Z = z_n \mid \forall n \in \{0,N\}, \forall z_n \in \{0,1\}$. Due to the independence of the observations the likelihood function can be constructed as

$$p(Z \mid \mu) = \prod_{n=1}^{N} p(z_n \mid \mu) = \prod_{n=1}^{N} \mu^{z_n} (1-\mu)^{1-z_n}. \tag{A.4}$$

Setting the derivative of the likelihood function with respect to μ equal zero, the maximum likelihood estimator is obtained

$$\mu_{\mathrm{ML}} = \frac{1}{N} \sum_{n=1}^{N} z_n, \tag{A.5}$$

which is known as the sample mean. Observing m times the value 1 this gets $\mu_{\mathrm{ML}} = \frac{m}{N}$.

A.2 Generalized Bernoulli Distribution

Generalizing the Bernoulli experiment (see section A.1) and considering discrete random variables taking values on one of K possible mutually exclusive states, a convenient representation is the 1-of-K scheme. The random variable is represented by a K-dimensional vector z_k in which one of the elements equals 1, and all remaining elements equal 0. E.g. for the case of Z should represent the value "4" in a 1-of-6 scheme, the vector reads $(0,0,0,1,0,0)^T$ fulfilling the relation $\sum_{k=1}^{K} z_k = 1$. Describing the probability of $z_k = 1$ via the parameter μ_k, the distribution of Z reads

$$\mathrm{GBer}(z_k \mid \mu_k) = \prod_{k=1}^{K} \mu_k^{z_k}. \tag{A.6}$$

The mean value μ_k and the variance (see (3.4) in section 3.1.1) are calculated according to

$$\mathbb{E}\{z_k\} = \mu_k \tag{A.7}$$

$$\mathrm{var}\{z_k\} = \mu_k(1-\mu_k). \tag{A.8}$$

Let $\mathbf{Z}$ be a matrix comprising N identical and independent multinomial distributed observations $\mathbf{Z} = z_{nk} \mid \forall n \in \{0,N\}, \forall k \in \{0,K\}, \forall z_{nk} \in \{0,1\}$. The corresponding likelihood function then takes the form

$$\mathrm{GBer}(z_{nk} \mid \mu_k) = \prod_{n=1}^{N}\prod_{k=1}^{K} \mu_k^{z_{nk}} = \prod_{k=1}^{K} \mu_k^{\sum_n z_{nk}} = \prod_{k=1}^{K} \mu_k^{m_k}, \tag{A.9}$$

depending on the data points through the K quantities m_k which are called the sufficient statistics of the distribution. In order to find the maximum likelihood estimate of μ_k, we derivate the logarithm of the generalized Bernoulli distribution $\ln(\mathrm{GBer}(z_{nk} \mid \mu_k))$ with respect to the each mean value μ_k. Using the Lagrange multiplier λ to incorporate the summation constraint $\sum_{k=1}^{K} z_k = 1$, the maximization reads

$$\sum_{k=1}^{K} m_k \ln(\mu_k) + \lambda\Big(\sum_{k=1}^{K} \mu_k - 1\Big), \tag{A.10}$$

which results in

$$\mu_k^{ML} = \frac{\sum_n z_n}{N}. \tag{A.11}$$

A.3 Gauss Distribution

The Gaussian distribution, also called the Normal distribution, is a continuous probability distribution of a random variable $x \in (-\infty, \infty)$ widely used in natural science, e.g. to describe random measurement error. It is written as

$$\mathcal{N}(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}, \tag{A.12}$$

with mean and variance (see (3.4) in section 3.1.1)

$$\mathbb{E}\{x\} = \mu \tag{A.13}$$

$$\text{var}\{x\} \quad = \quad \sigma^2. \tag{A.14}$$

In some cases a convenient representation of (A.12) employs the inverse of the variance which straightway is named the precision λ, $\mathcal{N}(x \mid \mu, \lambda^{-1})$. The square root of the variance is called standard deviation (stdev) which is tagged σ. The Gauss distribution is the conjugate prior (section 4.3.1) of the Gaussian mean parameter.

For a random vector X of N jointly (dependent among each other) Gaussian random variables, the joint probability distribution get

$$\mathcal{N}(X \mid \vec{\mu}, \Sigma) = -\frac{1}{\sqrt{(2\pi)^N \det \Sigma}} e^{-\frac{1}{2}(X-\vec{\mu})^{\mathrm{T}} \Sigma^{-1} (X-\vec{\mu})}, \tag{A.15}$$

with mean and covariance matrix defined as

$$\mathbb{E}\{X\} \quad = \quad \vec{\mu} = (\mu_1, \cdots, \mu_n)^T \tag{A.16}$$

$$\text{covar}\{X\} \quad = \quad \Sigma = (X - \vec{\mu})^{\mathrm{T}} (X - \vec{\mu}) = \begin{pmatrix} \sigma_{11}^2 & \cdots & \Sigma_{1n} \\ \vdots & \ddots & \vdots \\ \Sigma_{n1} & \cdots & \sigma_{nn}^2 \end{pmatrix}. \tag{A.17}$$

If the random variables X are independent of each other, the covariance Σ is a diagonal matrix. In this case the joint probability (A.15) reduces to a product of (A.12)

$$\mathcal{N}(X \mid \vec{\mu}, \Sigma) = \prod_{n=1}^{N} \frac{1}{\sqrt{2\pi\sigma_{nn}^2}} e^{-\frac{1}{2\sigma_{nn}^2}(x-\mu_n)^2}. \tag{A.18}$$

Assume the jointly Gaussian random vector X is divided into two disjoint subsets of random variables X_1 and X_2

$$X = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}, \qquad \vec{\mu} = \begin{pmatrix} \vec{\mu}_1 \\ \vec{\mu}_2 \end{pmatrix}, \qquad \Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}. \tag{A.19}$$

Then it can be shown [1, 13, 14, 38] that the conditional probability of X_1 given X_2 has Gaussian structure as well which reads

$$p(X_1 \mid X_2) = \mathcal{N}(\vec{\mu}_{X_1|X_2}, \Sigma_{X_1|X_2}). \tag{A.20}$$

Hence the conditional mean and the conditional covariance matrix are given by

$$\vec{\mu}_{X_1|X_2} \quad = \quad \vec{\mu}_1 + \Sigma_{12}\Sigma_{22}^{-1}(X_2 - \mu_2) \tag{A.21}$$

$$\Sigma_{X_1|X_2} \quad = \quad \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^{\mathrm{T}}. \tag{A.22}$$

Using a precision matrix

$$\Lambda = \begin{pmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{pmatrix} \tag{A.23}$$

instead of the covariance matrix, the conditional probability distribution of X_1 given X_2 equates to

$$p(X_1 \mid X_2) = \mathcal{N}(\vec{\mu}_{X_1|X_2}, \Lambda_{X_1|X_2}^{-1}) \tag{A.24}$$

with

$$\vec{\mu}_{X_1|X_2} \quad = \quad \vec{\mu}_1 - \Lambda_{11}^{-1}\Lambda_{12}(X_2 - \mu_2) \tag{A.25}$$

$$\Lambda_{X_1|X_2} \quad = \quad \Lambda_{11}. \tag{A.26}$$

A.4 GMM

When it comes to model real datasets, the expressive power of a Gaussian distribution shown in (A.12) is limited. Therefore we assuming a superposition (mixture) of K Gaussian distributions (A.12) for the random variable x. For this to be normalized accurately, weighting factors τ_k (mixing coefficients) have to be introduced resulting in

$$p(x) = \sum_{k=1}^{K} \tau_k \mathcal{N}(x \mid \mu_k, \sigma_k^2). \tag{A.27}$$

Thus the mixing coefficients τ_k have to fulfill the Kolmogorov Axioms $\sum_{k\in K} \tau_k = 1$ and $1 \geq \tau \geq 0$.

Employing a K-dimensional binary random variable z_k having a 1-of-K scheme as mentioned in section A.2, a joint distribution $p(x, Z)$ can be formulated in terms of a marginal distribution $p(Z)$ and a conditional distribution $p(x \mid Z)$ according to the product rule (3.9).The marginal distribution of Z is written using (A.6) as

$$p(Z) \equiv= \prod_{k=1}^{K} \tau_k^{z_k} \tag{A.28}$$

$$\longrightarrow \qquad p(z_{k'} = 1) = \tau_{k'}, \tag{A.29}$$

yielding $\tau_{k'}$ for a specific choice of $z_{k'} = 1$. The conditional distribution of X given a specific choice of $z_{k'} = 1$ should result in a Gaussian distribution, which is achieved by using

$$p(x \mid Z) = \prod_{k=1}^{K} \mathcal{N}(X \mid \mu_k, \sigma_k^2)^{z_k} \tag{A.30}$$

$$\longrightarrow \qquad p(x \mid z_k = 1) = \mathcal{N}(x \mid \mu_k, \sigma_k^2). \tag{A.31}$$

With this the joint probability can be written according to (A.6) as

$$p(x, Z) = p(x \mid Z)p(Z) = \prod_{k=1}^{K} \left[\tau_k \mathcal{N}(x \mid \mu_k, \sigma_k^2)\right]^{z_k} \tag{A.32}$$

To simply proof this equation, the marginal distribution $p(x)$ given by (A.27) should be described by a superposition of Gaussian distributions $\mathcal{N}(x \mid \mu_k, \sigma_k^2)$ weighted by τ_k. This is easily seen by summing the joint probability regarding Z (marginalizing)

$$p(x) = \sum_{Z\epsilon\mathcal{Z}^{\mathcal{K}}} p(x, Z) = \sum_{k=1}^{K} \tau_k \mathcal{N}(x \mid \mu_k, \sigma_k^2). \tag{A.33}$$

A.5 Gauss Normalization

Calculating the normalization constant for Gaussian densities including the temperature parameter C, following normalization equation must hold

$$C \int_{-\infty}^{\infty} e^{-\frac{(x-a)^2}{b}} dx = 1. \tag{A.34}$$

With the Substitution

$$\frac{x-a}{\sqrt{b}} = z \quad \rightarrow \quad dx = \sqrt{b}dz, \tag{A.35}$$

this reads

$$C\sqrt{b}\int_{-\infty}^{\infty} e^{-z^2}dz = 1. \tag{A.36}$$

Via squaring the above integral it can be written in two dimensions as

$$\Big(\int_{-\infty}^{\infty} e^{-z^2}dz\Big)^2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-y^2}e^{-z^2}dydz. \tag{A.37}$$

Introducing polar coordinates $y = r\cos\varphi, z = r\sin\varphi$ with $r^2 = y^2 + z^2$ and calculating the determinant of the Jakobi-matrix $\det J = \det\frac{\partial(x,y)}{\partial(r,\varphi)} = r$, this can be rewritten as

$$\int_0^{\infty}\int_0^{2\pi} re^{-r^2}drd\varphi = 2\pi\int_0^{\infty} re^{-r^2}dr. \tag{A.38}$$

Representing the function under the integral on the right side of (A.38) as $\frac{d}{dr}(-\frac{1}{2}e^{-r^2})$ and evaluation it at $r = 0$ and $r = \infty$, the normalization get

$$C\sqrt{b\pi} = 1 \longrightarrow C = \frac{1}{\sqrt{b\pi}}. \tag{A.39}$$

A.6 Gamma Distribution

The Gamma distribution is a continuous probability distribution over the positive real numbers $x \in (0, \infty)$. It is the conjugate prior for the precision parameter of the Gaussian distribution and is written as

$$\mathrm{Gam}(x \mid a, b) = \frac{1}{\Gamma(a)}b^a x^{a-1}e^{-bx} \qquad \Gamma(a) = (a-1)!. \tag{A.40}$$

With $a > 0$ and $b > 0$ it is

$$\mathbb{E}\{x\} = \frac{a}{b} \tag{A.41}$$

$$\mathrm{var}\{x\} = \frac{a}{b^2} \tag{A.42}$$

$$\begin{aligned}\mathbb{E}\{\ln x\} &= \frac{d}{da}\ln\Gamma(a) - \ln(b) \\ &= \psi(a) - \ln(b).\end{aligned} \tag{A.43}$$

Here $\psi(a)$ is denoting the digamma function of a which equates to

$$\psi(a) = \frac{d}{da}\ln\Gamma(a). \tag{A.44}$$

A.7 Dirichlet Distribution

The Dirichlet distribution is a multivariate distribution for K random variables τ_k in the interval [0,1]. With the positive real values $\alpha_k > 0$, it is written according to

$$\mathrm{Dir}(\tau_k \mid \alpha_k) = C(\alpha) \prod_k \tau_k^{\alpha_k - 1}. \tag{A.45}$$

Thus the function $C(\alpha)$ reads

$$C(\alpha) = \frac{\Gamma(\hat{\alpha})}{\sum_k \Gamma(\alpha_k)} \qquad \hat{\alpha} = \sum_k \alpha_k. \tag{A.46}$$

The following relations can be established

$$\mathbb{E}\{\tau_k\} = \frac{\alpha_k}{\hat{\alpha}} \tag{A.47}$$

$$\mathbb{E}\{\ln \tau_k\} = \psi(\alpha_k) - \ln(\hat{\alpha}) \tag{A.48}$$

denoting $\psi(\alpha_k)$ the digamma function shown in .

B

Estimation Theory

B.1 Example: Gaussian prior for the mean

For this example a Gaussian distributed data vector X is defined with known variance σ and conditional probability written as $p(X \mid \Theta) = p(X \mid \mu) = \mathcal{N}(X \mid \mu, \sigma^2)$. The mean parameter μ as statistical quantity is assumed to have a Gaussian prior given by $p(\Theta) = p(\mu) = \mathcal{N}(\mu \mid \mu_0, \sigma_0^2)$, with μ_0 and σ_0 called the hyper parameters. Searching the functional from of (3.35), first the derivative of the joint probability $p(X, \mu) = p(X \mid \mu)p(\mu)$ is calculated as

$$\frac{\partial}{\partial \mu} \ln[p(X \mid \mu)p(\mu)] = \underbrace{\left(\frac{N\sigma_0^2 + \sigma^2}{\sigma^2\sigma_0^2}\right)}_{K} \Bigg[\underbrace{\left(\frac{N\bar{X}\sigma_0^2 + \mu_0\sigma^2}{N\sigma_0^2 + \sigma^2}\right)}_{g(X)=\hat{\mu}(X)} - \mu \Bigg], \tag{B.1}$$

where $\bar{X}$ is the sample mean

$$\bar{X} = \frac{1}{N} \sum_{n=1}^{N} x_n. \tag{B.2}$$

Identifying the function $g(X)$ on the right hand side of (B.1) and decoupling the sample mean and the prior parameter μ_0, the MMSE estimator of the mean parameter μ is given by

$$\hat{\mu}_{\mathrm{MMSE}}(X) = \mathbb{E}\{\mu \mid X\} = \hat{\mu}_{\mathrm{ML}} \frac{N\sigma_0}{N\sigma_0^2 + \sigma^2} + \mu_0 \frac{\sigma^2}{N\sigma_0^2 + \sigma^2}. \tag{B.3}$$

Here $\hat{\mu}_{\mathrm{ML}}$ is the maximum likelihood estimator defined in (3.40) which in this case is equal the sample mean $\bar{X}$. The MSE/error variance is moreover related according to (3.37) as

$$\frac{1}{\mathrm{MSE}_{\hat{\mu}}} = \frac{1}{\mathrm{var}\{E\}} = \frac{N}{\sigma^2} + \frac{1}{\sigma_0^2}. \tag{B.4}$$

As can be seen from (B.3), the smaller the sample size N gets (bad statistical ensembles) the more relevant the second term in (B.3) becomes which is governed by the prior parameter (further explanation see section 4.3.1). Moreover if the sample size N increase and we get more evidence due to measurement data, the MSE/error variance decline toward zero.

B.2 Example: Gamma prior for the precision

For this example again a Gaussian distributed data vector X is defined but with known mean μ and conditional probability written as $p(X \mid \Theta) = p(X \mid \lambda) = \mathcal{N}(X \mid \mu, \lambda^{-1})$. The statistical quantity λ, the inverse variance, is assumed to have a Gamma prior (A.40) given by $p(\Theta) = p(\lambda) = \mathrm{Gam}(\lambda \mid a_0, b_0)$ with the hyper parameters a_0 and b_0. As in section B.1, we calculate the derivative of the logarithmic joint probability $p(X, \lambda) = p(X \mid \lambda)p(\lambda)$ and rearrange the obtained equation to equalize (3.35)

$$\frac{\partial}{\partial \mu} \ln[p(X \mid \lambda)p(\lambda)] = \underbrace{\left(1 - a_0 - \frac{N}{2}\right)}_{K} \left[\underbrace{\left(\frac{b_0 + \frac{N}{2}\mathrm{var}\{X\}}{\frac{N}{2} + a_0 - 1} \right)}_{g(X) = \frac{1}{\hat{\lambda}(X)}} - \frac{1}{\lambda} \right], \tag{B.5}$$

where $\mathrm{var}\{X\}$ is the sample variance

$$\mathrm{var}\{X\} = \frac{1}{N} \sum_n (x_n - \mu)^2. \tag{B.6}$$

Identifying the function $g(X)$ on the right hand side of (B.5) as the MMSE estimator of the inverse precision, we may write

$$\hat{\lambda}_{\mathrm{MMSE}}(X) = \mathbb{E}\{\lambda \mid X\} = \frac{a_{\mathrm{MMSE}}}{b_{\mathrm{MMSE}}} \tag{B.7}$$

with

$$a_{\mathrm{MMSE}} = \frac{N}{2} + a_0 - 1 \tag{B.8}$$

$$b_{\mathrm{MMSE}} = b_0 + \frac{N}{2}\hat{\sigma}_{\mathrm{ML}}. \tag{B.9}$$

Hence $\hat{\sigma}_{\mathrm{ML}}$ is the maximum likelihood estimator defined in (3.40) which in this case is equal the sample variance $\mathrm{var}\{X\}$. The MSE/error variance is moreover related according to (3.37) as

$$\frac{1}{\mathrm{MSE}_{\hat{\lambda}}} = \frac{1}{\mathrm{var}\{E\}} = 1 - a_0 - \frac{N}{2} \tag{B.10}$$

C

Clustering Algorithms

ALL clustering algorithms studied in this work consider a data matrix X assumed known. The unknown quantities are given by a parameter vector governing probability distributions and by an unknown label matrix $\mathbf{Z}$. The label matrix $\mathbf{Z}$ is introduced to manage multiple data clusters X_k with k tagging the cluster index. The usage of binary label values z_{nk} is simplifying the completion of the algorithms programmatically employing a free available c++ compiler or python interpreter.

C.1 EM Clustering for a GMM

Using the EM procedure defined in section 3.3.2.1, a clustering problem for a GMM section A.4 can be formulated (see also [1, 32]). Hence inserting the logarithm of the joint distribution for a GMM (A.32) in (3.43) we get

$$\Theta^{(i+1)} \propto \arg\max_{\Theta} \left\{ \mathbb{E}\Big\{ z_{nk} \ln \tau_k - z_{nk} \ln \sigma - z_{nk} \frac{(x_n - \mu_{nk})^2}{\sigma_{nk}^2} \mid X; \Theta^{(i)} \Big\} \right\} \tag{C.1}$$

$$= \arg\max_{\Theta} \left\{ \prod_{k=1}^{K} \mathbb{E}_{X|X;\Theta^{(i)}}\{z_k\} \ln \tau_k + \mathbb{E}_{X|X;\Theta^{(i)}}\{z_k\} \ln \mathcal{N}(x \mid \mu_k, \sigma_k^2) \right\}. \tag{C.2}$$

With some initial guess of the parameter vector $\Theta^{(i)}$ the conditional probability of $\mathbf{Z}$ given X is calculated using Bayes rule (3.11). With (A.33) one gets

$$p(z_{nk} \mid x_n; \Theta) = \frac{\prod_{k=1}^{K} \mathcal{N}_k(x_n; \mu_k^{(i)}, \sigma_k^{(i)})^{z_{nk}} \tau_k^{(i) z_{nk}}}{\sum_{k=1}^{K} \mathcal{N}_k(x_n; \mu_k^{(i)}, \sigma_k^{(i)}) \tau_k^{(i)}}. \tag{C.3}$$

For z_{nk} being of binary nature, this also corresponds to the expectation of z_{nk} conditioned on x_n and so the E-step (3.42) gets written as

$$\mathbb{E}\{z_{nk} \mid x_n; \theta_k\} = \frac{\mathcal{N}_k(x_n; \mu_k^{(i)}, \sigma_k^{(i)}) \tau_k^{(i)}}{\sum_{k=1}^{K} \mathcal{N}_k(x_n; \mu_k^{(i)}, \sigma_k^{(i)}) \tau_k^{(i)}}. \tag{C.4}$$

Using the joint probability of $\mathbf{Z}$ and X (A.32), the maximum likelihood problem called M-step (3.43) gets

$$\theta_k^{(i+1)} = \arg\max_{\theta_k} \mathbb{E}_{Z|X;\theta^{(i)}} \left\{ \log \left[\prod_{k=1}^{K} \prod_{n=1}^{N} (\mathcal{N}_k(x_n; \mu_k, \sigma_k) \tau_k)^{z_{nk}} \right] \right\}$$

$$= \arg\max_{\theta_k} \sum_{k=1}^{K} \sum_{n=1}^{N} \mathbb{E}\{z_{nk} | x_n; \theta_k^{(i)}\} [\log \mathcal{N}_k(x_n; \mu_k, \sigma_k) + \log \tau_k]. \tag{C.5}$$

To solve the ML estimation, the normalization condition of τ_k has to be taken into concern. Therefore a Lagrange multiplier is introduced yielding the following optimization problem (for convenience $\mathbb{E}\{z_{nk} \mid x_n; \theta_k\}$ is written as $\mathbb{E}\{z_{nk}\}$)

$$0 = \frac{\partial}{\partial \theta_k} \Big[\sum_{k=1}^{K} \sum_{n=1}^{N} \mathbb{E}\{z_{nk}\} [\log \mathcal{N}_k(x_n; \mu_k, \sigma_k) + \log \tau_k] + \lambda (\sum_{k=1}^{K} \tau_k - 1) \Big]$$

$$= \frac{\partial}{\partial \theta_k} \Big[\sum_{n=1}^{N} \mathbb{E}\{z_{nk}\} [\ln \tau_k - \ln \sigma_k + \frac{(x_n - \mu_k)^2}{2\sigma_k^2}]$$

$$+ \text{const} + \lambda (\sum_{k=1}^{K} \tau_k - 1) \Big]. \tag{C.6}$$

Solving this equation for all θ_k yields the parameter updates of the M-step

$$\tau_k^{(i+1)} = \sum_{n=1}^{N} \frac{E\{z_{nk}\}}{N}, \tag{C.7}$$

$$\mu_k^{(i+1)} = \frac{\sum_{n=1}^{N} x_n E\{z_{nk}\}}{\sum_{n=1}^{N} E\{z_{nk}\}} \tag{C.8}$$

and

$$\sigma_k^{(i+1)} = \sqrt{\frac{\sum_{n=1}^{N} (x_n - \mu_k^{(i+1)})^2 E\{z_{nk}\}}{\sum_{n=1}^{N} E\{z_{nk}\}}}. \tag{C.9}$$

C.2 General EM

To motivate the use of variational methods to solve complex Bayesian models (section 4.3.5) as done in the next section, a different view on the EM process (section 3.3.2.1) is given which can be found in more detail in [1]. The goal is to maximize the logarithm of the likelihood function $p(X \mid \Theta)$ by introducing latent variables $\mathbf{Z}$ as

$$p(X \mid \Theta) = \sum_{\mathbf{Z}} p(X, \mathbf{Z} \mid \Theta), \tag{C.10}$$

because it is assumed that the optimization of $p(X \mid \Theta)$ is too difficult. Moreover an approximating distribution $q(\mathbf{Z})$ is introduced for the latent variables, as have been done in section 3.2.

For any choice of the distribution $q(\mathbf{Z})$, the decomposition

$$\ln p(X \mid \Theta) = \mathcal{L}(q, \Theta) + \mathrm{KL}(q \parallel p) \tag{C.11}$$

holds, where $\mathcal{L}(q, \Theta)$ is called a lower bound on $\ln p(X \mid \Theta)$ (see next paragraph) given by

$$\mathcal{L}(q, \Theta) = \sum_{\mathbf{Z}} \ln \left\{ \frac{p(X, \mathbf{Z} \mid \Theta)}{q(\mathbf{Z})} \right\} q(\mathbf{Z}). \tag{C.12}$$

and $\mathrm{KL}(q \parallel p)$ terms the Kullback Leiber divergence (3.19)

$$\mathrm{KL}(q \parallel p) = -\sum_{\mathbf{Z}} \ln \left\{ \frac{p(\mathbf{Z} \mid X, \Theta)}{q(\mathbf{Z})} \right\} q(\mathbf{Z}). \tag{C.13}$$

First it is stated that $\mathcal{L}(q, \Theta)$ is lower than $\ln p(X \mid \Theta)$, because the Kullback Leiber divergence satisfies $\mathrm{KL}(q \parallel p) \geq 0$ see (3.20). For this $\mathcal{L}(q, \Theta)$ is said to be a lower bound on $\ln p(X \mid \Theta)$ as stated above. But Kullback Leiber divergence vanishes only if $q(\mathbf{Z}) = p(\mathbf{Z} \mid X, \Theta)$ see (C.13), meaning that the approximating distribution gets equal the posterior distribution, which in turn yields

$$\mathcal{L}(q, \Theta) = \sum_{\mathbf{Z}} \ln p(\mathbf{Z}, X \mid \Theta) p(\mathbf{Z} \mid X, \Theta) - \sum_{\mathbf{Z}} \ln p(\mathbf{Z} \mid X, \Theta) p(\mathbf{Z} \mid X, \Theta), \tag{C.14}$$

where the second term is the entropy of $q(\mathbf{Z}) = p(\mathbf{Z} \mid X, \Theta)$, see (3.16).

To illustrate the iterative EM-process as given in section 3.3.2.1, first the E-step is considered with Θ^{old} is held fixed. Therewith, $\mathcal{L}(q, \Theta^{\mathrm{old}})$ is maximized regarding $q(\mathbf{Z})$ whereby $\ln p(X \mid \Theta^{\mathrm{old}})$ does not depend on $q(\mathbf{Z})$ and so this is achieved when Kullback Leiber divergence vanishes. As stated in section 3.2 if Kullback Leibler divergence vanishes the approximating probability gets equal the posterior distribution $q(\mathbf{Z}) = p(\mathbf{Z} \mid X, \Theta^{\mathrm{old}})$ as can be seen from (C.13).

In the subsequent M-step, $q(\mathbf{Z})$ is held fixed and $\mathcal{L}(q,\Theta)$ is maximized regarding Θ. Using $q(\mathbf{Z}) = p(\mathbf{Z} \mid X,\Theta)$ in (C.12), the maximization problem is written as

$$\Theta^{\text{new}} = \arg\max_{\Theta} \Big[\sum_{\mathbf{Z}} p(\mathbf{Z} \mid X,\Theta^{\text{old}}) \ln p(X,\mathbf{Z} \mid \Theta) - \text{const} \Big], \tag{C.15}$$

which is exactly the ML problem problem emerging inside the EM procedure (3.42).

With this in mind and in an Bayesian setting where the parameters Θ are of random nature as the latent variables $\mathbf{Z}$ are, instead of optimizing the conditional probability (C.10) one can approximate the posterior distribution of Θ and $\mathbf{Z}$ by $q(\mathbf{Z})$ and $q(\Theta)$ and directly maximize the lower bound (C.12) or minimize the Kulback Leibler divergence (C.13). To receive traceable algorithms, beside the usage of conjugate prior distributions section 4.3.1 a further restriction to the approximating distribution q is done via the factorization probability section C.3.

C.3 Factorized Distributions

Assuming a fully Bayesian model in which all parameters are of random nature section 3.3.1. With this, for further convenience during this subsection, Θ and the latent variables $\mathbf{Z}$ gets absorbed into the random vector $\mathbf{Z}$ ($\mathbf{Z}_i = \{\mathbf{Z}, \mu, \sigma, ...\}$). Next consider a partitioning i of $\mathbf{Z}$ into M subsets for which the family of distributions $q(\mathbf{Z})$ factorizes according to

$$q(\mathbf{Z}) = \prod_{i=1}^{m} q_i(\mathbf{Z}_i). \tag{C.16}$$

Considering the optimization problem (C.10) with data vector X. Again it is aimed to maximize the lower bound (C.12) as this minimizes the Kulback Leibler divergence and hence the dissimilarity between the approximating distribution and posterior distribution, but restricted to approximating distributions introduced in (C.16). Inserting (C.16) into (C.12) reads

$$\begin{aligned}
\mathcal{L}(q) &= \int \prod_i q_i \Big\{ \ln p(X,\mathbf{Z}) - \sum_i \ln q_i \Big\} d\mathbf{Z} \\
&= \int q_j \Big\{ \int \ln p(X,\mathbf{Z}) \prod_{i \neq j} q_i d\mathbf{Z}_i \Big\} d\mathbf{Z}_j - \int q_j \ln q_j d\mathbf{Z}_j + \text{const} \\
&= \int q_j \ln \tilde{p}(X,\mathbf{Z}_j) d\mathbf{Z} - \int q_j \ln q_j d\mathbf{Z}_j + \text{const},
\end{aligned} \tag{C.17}$$

where $\tilde{p}(X,\mathbf{Z}_j)$ is defined by

$$\ln \tilde{p}(X,\mathbf{Z}_j) = \mathbb{E}_{i \neq j}\{\ln p(X,\mathbf{Z})\} + \text{const}.. \tag{C.18}$$

Here the notation $\mathbb{E}_{i \neq j}\{...\}$ denotes an expectation with respect to the q distributions over all variables z_i for $i \neq j$, so that

$$\mathbb{E}_{i \neq j}\{\ln p(X,\mathbf{Z})\} = \int \ln p(X,\mathbf{Z}) \prod_{i \neq j} q_i d\mathbf{Z}_i. \tag{C.19}$$

Using (C.19) in (C.17) and realizing that (C.17) is a negative Kullback Leibler divergence between $q_j(\mathbf{Z})$ and $\tilde{p}(X, \mathbf{Z}_j)$, maximizing (C.17) is equal to minimize the Kullback Leibler divergence and one gets the optimal solution as

$$\ln q_j^*(\mathbf{Z}) = \mathbb{E}_{i \neq j}\{\ln p(X, \mathbf{Z})\} + \text{const.}. \tag{C.20}$$

C.4 Variational Inference for Bayesian GMM

To derive a fully Bayesian GMM [1] (assuming a basic GMM model defined in section A.4) with prior distributions defined for all main parameters (μ, σ or λ and τ), it is assumed that the true joint probability distribution factorizes (see section C.3) among the main parameters and latent variables $\mathbf{Z}$ as

$$p(X, \mathbf{Z}, \mu, \lambda, \tau) = p(x \mid \mathbf{Z}, \mu, \lambda) \underbrace{p(\mu \mid \lambda)p(\lambda)}_{p(\mu,\lambda)} p(\mathbf{Z} \mid \tau)p(\tau). \tag{C.21}$$

As mentioned in section 4.3.1, choosing specific prior distributions from the exponential family yields closed form solutions to the parameter estimation problem just by inspection of the according terms in the log likelihood.

First of all, the basic GMM from section A.4 get employed using the precision λ rather than the squared standard deviation $\mathcal{N}(X \mid \mathbf{Z}, \mu, \lambda^{-1})$, see section A.3. As further mentioned in section A.3, an appropriate conjugate prior for the Gaussian mean is a Gaussian distribution whereas for the precision it is the Gamma distribution, see section A.6. A specific choice is to indent a constant β_0 and let the precision of the prior for μ be a linear function of the precision of conditional probability for X, $\mathcal{N}(X \mid \mathbf{Z}, \mu, \lambda^{-1})$, yielding a Gauss-Gamma distribution

$$p(\mu, \lambda \mid \mu_0, (\beta_0\lambda)^{-1}, a_0, b_0) = \mathcal{N}(\mu \mid \mu_0, (\beta_0\lambda)^{-1})\text{Gam}(\lambda \mid a_0, b_0). \tag{C.22}$$

For the label matrix $\mathbf{Z}$, a Generalized Bernoulli distribution $\text{GBer}(\mathbf{Z} \mid \tau)$ (see section A.2) with a Dirichlet prior $\text{Dir}(\tau \mid \alpha_0)$ (see section A.7) is employed.

Next consider the variational distribution (section C.3)

$$q(\mathbf{Z}, \mu, \lambda, \tau) = q(\mathbf{Z})q(\mu, \lambda, \tau). \tag{C.23}$$

Using (C.20), the optimal solution for p($\mathbf{Z}$) calculates as

$$\begin{aligned} \ln q^*(\mathbf{Z}) &= \mathbb{E}_{\mu,\lambda}\{\ln p(X \mid \mathbf{Z}, \mu, \lambda)\} + \mathbb{E}_\tau\{\ln p(\mathbf{Z} \mid \tau)\} + \text{const.} \\ &= \sum_{n=1}^{N}\sum_{k=1}^{K} z_{nk}\Big[\underbrace{\mathbb{E}_\tau\{\ln \tau_k\} - \frac{1}{2}\ln(2\pi) + \mathbb{E}_\lambda\{\ln \lambda_k\} + \frac{1}{2}\mathbb{E}_{\mu,\lambda}\{\lambda_k(x_n - \mu_k)^2\}}_{\ln \rho_{nk}}\Big] \\ &\quad + \text{const.}. \end{aligned} \tag{C.24}$$

With the definition of ρ_{nk}, the optimal solution for $q^*(\mathbf{Z})$ is proportional to a Generalized Bernoulli distribution

$$q^*(\mathbf{Z}) \approx \prod_{n=1}^{N}\prod_{k=1}^{K} \rho_{nk}^{z_{nk}}. \tag{C.25}$$

For this distribution to be normalized appropriately and to achieve equality in (C.25), the probability for the binary random variable $q^*(z_{nk})$ must sum to one

$$q^*(\mathbf{Z}) = \prod_{n=1}^{N}\prod_{k=1}^{K}\frac{\rho_{nk}^{z_{nk}}}{\sum_k \rho_{nk}^{z_{nk}}}. \tag{C.26}$$

Applying the same procedure to the second term in (C.23) yields

$$\begin{aligned}\ln q(\mu,\lambda,\tau) &= \mathbb{E}_{\mathbf{Z}}\{\ln p(X \mid \mathbf{Z},\mu,\lambda) + \ln p(\mu \mid \lambda) + \ln p(\lambda) \\ &\quad + \ln p(\mathbf{Z} \mid \tau_k) + \ln p(\tau)\} + \text{const.}.\end{aligned} \tag{C.27}$$

It is seen that $q(\mu,\lambda,\tau)$ factorization further as

$$q(\mu,\lambda,\tau) = q(\tau)\prod_k q(\mu,\lambda). \tag{C.28}$$

Identifying the terms that depends on τ reads

$$\begin{aligned}\ln q^*(\tau) &= \mathbb{E}_{\mathbf{Z}}\{\ln p(\mathbf{Z} \mid \tau_k) + \ln p(\tau)\} + \text{const.} \\ &= \sum_{n=1}^{N}\sum_{k=1}^{K}\mathbb{E}_{\mathbf{Z}}\{z_{nk}\}\ln\tau_k + \sum_{k=1}^{K}(\alpha_{0,k}-1)\ln\tau_k + \text{const.}.\end{aligned}$$

A factor analysis reveals immediately a Dirac distribution for the prior parameter τ

$$q^*(\tau) = \mathrm{Dir}(\tau \mid \alpha), \quad \text{with} \quad \alpha_{N,k} = \alpha_{0,k} + \sum_n \mathbb{E}_{\mathbf{Z}}\{z_{nk}\}. \tag{C.29}$$

Lastly, $q(\mu,\lambda)$ factorizes according to

$$q(\mu,\lambda) = q(\mu \mid \lambda)q(\lambda). \tag{C.30}$$

Trying to find updates for μ just two terms are of interest,

$$\begin{aligned}\ln q^*(\mu_k) &= -\frac{\lambda_k}{2}\Big[\mu_k^2(\sum_n \mathbb{E}_{\mathbf{Z}}\{z_{nk}\} + \beta_{0,k}) - 2\mu_k(\beta_{0,k}\mu_{0,k} + \sum_{n=1}^{N} x_n\mathbb{E}_{\mathbf{Z}}\{z_{nk}\}) \\ &\quad + \text{const.},\end{aligned}$$

which yields a Gaussian distribution after completing the square

$$q^*(\mu_k) = \mathcal{N}(\mu_k \mid \mu_{N,k}, (\beta_{N,k}\lambda_k)^{-1}). \tag{C.31}$$

The mean value μ_N and the factor β_N hence are calculated as

$$\mu_{N,k} = (\beta_{0,k}\mu_{0,k} + \sum_{n=1}^{N} x_n\mathbb{E}_{\mathbf{Z}}\{z_{nk}\})\beta_{N,k}^{-1} \tag{C.32}$$

$$\beta_{N,k} = \sum_n \mathbb{E}_{\mathbf{Z}}\{z_{nk}\} + \beta_{0,k}. \tag{C.33}$$

Finally considering the parameter λ one gets

$$\ln q^*(\lambda_k) = \ln\lambda_k\Big[\sum_n \frac{\mathbb{E}_{\mathbf{Z}}\{z_{nk}\}}{2} + \frac{1}{2} + (a_{0,k}-1)\Big]$$

$$-\lambda_k\Big[\frac{1}{2}\sum_{n=1}^{N}(x_n-\mu_k)^2\mathbb{E}_{\mathbf{Z}}\{z_{nk}\}-\frac{\beta_{0,k}}{2}(\mu_k-\mu_{0,k})^2-b_{0,k}\Big]$$
$$+\text{const.},$$

resulting in a Gamma distribution

$$q^*(\lambda_k)=\text{Gam}(\lambda_k\mid a_{N,k},b_{N,k}), \tag{C.34}$$

having parameters $a_{N,k}$ and $b_{N,k}$

$$a_{N,k} = \sum_n\frac{\mathbb{E}_{\mathbf{Z}}\{z_{nk}\}+1}{2}+a_{0,k} \tag{C.35}$$

$$b_{N,k} = b_{0,k}+\frac{1}{2}\Big[\sum_{n=1}^{N}\mathbb{E}_{\mathbf{Z}}\{z_{nk}\}(x_n+\mu_k)^2+\beta_{0,k}(\mu_k-\mu_{0,k})^2\Big]. \tag{C.36}$$

With this, an iterative algorithm can be formulated. Thus, the E-step pendant is given by (C.26) with

$$\ln\rho_{nk}=\mathbb{E}_\tau\{\ln\tau_k\}-\frac{1}{2}\ln(2\pi)+\mathbb{E}_\lambda\{\ln\lambda_k\}+\frac{1}{2}\mathbb{E}_{\mu,\lambda}\{\lambda_k(x_n-\mu_k)^2\}. \tag{C.37}$$

The last term in (C.37) gives rise to some further simplifications. As seen from (C.30) the variational joint distribution of μ and λ factorizes and so does the joint expectation. With this one gets

$$\frac{1}{2}\mathbb{E}_{\mu,\lambda}\{\lambda_k(x_n-\mu_k)^2\}=\frac{1}{2}\mathbb{E}_\lambda\{\lambda_k\}\big[x_n^2-2x_n\mathbb{E}_{\mu|\lambda}\{\mu\}+\mathbb{E}_{\mu|\lambda}\{\mu^2\}\big]. \tag{C.38}$$

Performing a variational E-step, the desired moments are calculated according to the expectations under the posterior distributions

$$\mathbb{E}_\tau[\ln\tau_k]=\psi(\alpha_{N,k})-\psi(\sum_k\alpha_{N,k}) \tag{C.39}$$

$$\mathbb{E}_{\mu|\lambda}\{\mu_k\}=\mu_{N,k} \tag{C.40}$$

$$\mathbb{E}_{\mu|\lambda}\{\mu_k^2\}=\mu_{N,k}^2+(\beta_{0,k}\lambda_k)^{-1} \tag{C.41}$$

$$\mathbb{E}_\lambda\{\lambda_k\}=\frac{a_{N,k}}{b_{N,k}} \tag{C.42}$$

$$\mathbb{E}_\lambda\{\ln\lambda_k\}=\frac{d}{da_{N,k}}\ln\Gamma(a_{N,k})-\ln b_{N,k} \tag{C.43}$$

and hence

$$\mathbb{E}_{\mathbf{Z}}[z_{nk}]=\prod_n\prod_k\frac{\rho_{nk}}{\sum_k\rho_{nk}} \tag{C.44}$$

with ρ_{nk} defined as

$$\rho_{nk}=\mathbb{E}_\tau\{\ln\tau_k\}-\frac{1}{2}\ln(2\pi)+\mathbb{E}_\lambda\{\ln\lambda_k\}+\frac{1}{2}\mathbb{E}_{\mu,\lambda}\{\lambda_k(x_n-\mu_k)^2\}. \tag{C.45}$$

Again it is emphasized that the effect is just effective in case of bad statistical ensembles.

D

Convex Optimization

This chapter is aimed just to give a brief overview of convex optimization algorithms demanded in the context of this work. Specially unconstrained optimization of geometric programs in convex forms are discussed. For a deeper insight we revere to [3].

A function $f : \mathbb{R}^k \to \mathbb{R}$ is convex if

$$f(\theta X_1 + (1-\theta)X_2) \leq \theta f(X_1) + (1-\theta) f(X_2). \tag{D.1}$$

Examples of convex functions are given by the various partition functions (4.80), (4.81), (4.4.3.1), (4.4.3.1) and (4.117). Note that these equations are functions of the parameter α and $\tilde{\alpha}$ rather than functions of X as (D.1). The afore mentioned partition functions are logarithmic sums of exponentials

$$f(\alpha) = \log \sum_k e^{\alpha_k}, \tag{D.2}$$

which are convex on $\alpha \in \mathbb{R}^k$.

As mentioned in section 3.4.2 the partition function is infinitely often differentiable. Moreover $f(\alpha)$ fulfills the first order condition for convexity

$$f(\alpha') \geq f(\alpha) + \nabla f(\alpha)(\alpha' - \alpha) \tag{D.3}$$

and the second order condition for convexity

$$\nabla^2 f(\alpha) \geq 0, \tag{D.4}$$

for each $\alpha, \alpha' \in \mathbf{dom} f$ with the domain of f ($\mathbf{dom} f \subseteq \mathbb{R}^k$) being the input set of $f(\alpha)$.

Convex optimization concerns the minimization of a convex function f_0 under some constraints which get expressed due to the constraint functions f_i and g_j. Convex optimization problems are presented in standard form as

$$\begin{aligned} \text{minimize} \quad & f_0(\alpha) && && \text{(D.5)} \\ \text{subject to} \quad & f_i(\alpha) && \leq 0, \quad i = 1...I && \text{(D.6)} \\ & g_j(\alpha) && = 0, \quad j = 1...J. && \text{(D.7)} \\ & && && \text{(D.8)} \end{aligned}$$

Hence the functions f_i are assumed to be convex $\forall i = 0...I$ whereby the functions g_j are considered to be affine $\forall j = 0...J$. If the optimization function f_0 is concave, the minimization appearing in (D.5) has to be interchanged by a maximization.

The optimization problem, concerning the objective function (D.2)

$$\text{minimize} \quad f(\alpha) = \log \sum_k e^{\alpha_k} \tag{D.9}$$

is called an unconstrained geometric program in convex form which in general has no analytic solution. Since $f(\alpha)$ is differentiable and convex, a necessary and sufficient condition for a point α^* to be optimal, i.e.

$$\inf_\alpha f(\alpha) = f(\alpha^*), \tag{D.10}$$

is given by

$$\nabla f(\alpha^*) = 0. \tag{D.11}$$

As this is not achievable analytically as mentioned above, the idea is to generate a minimizing sequence

$$\alpha^{(r+1)} = \alpha^{(r)} + t^{(r)} \Delta\alpha^{(r)}, \tag{D.12}$$

with $\alpha^{(r)} \in \mathrm{dom} f$, so that

$$\lim_{r\to\infty} f(\alpha^{(r)}) = f(\alpha^*). \tag{D.13}$$

The parameter $t^{(r)}$ is called the step size which is greater zero (except $\alpha^{(r)}$ is optimal) and $\Delta\alpha^{(r)}$, the search direction, is a K-dimensional vector.

D.1 Gradient Descent Methods

With descent methods the samples are chosen according to

$$f(\alpha^{(r+1)}) < f(\alpha^{(r)}). \tag{D.14}$$

But from (D.3) it follows that α is optimal if and only if $\alpha \in \mathbf{dom} f$ and

$$\nabla f(\alpha)(\alpha' - \alpha) \geq 0 \quad \forall \alpha' \in \mathbf{dom} f, \tag{D.15}$$

which implies $f(\alpha') \geq f(\alpha)$ and hence a descent method has to satisfy

$$\nabla f(\alpha^{(r)})\Delta\alpha^{(r)} < 0. \tag{D.16}$$

With this a general descent method iterates as follows

Given some initial guess of $\alpha \in \mathbf{dom} f$, repeat until stopping criterion is fulfilled:

- Choose a search direction $\Delta\alpha^{(r)}$.
- Choose a step length $t^{(r)} > 0$.
- Update α according to (D.12).

The stopping criteria are free to be chosen. A measure which is naturally volunteering is the gradient of $f(\alpha)$, leading to a stopping criteria $\|\nabla f(\alpha)\|_2 \leq \epsilon$. The second step of this iterative procedure is called the line search which can be accomplished exact or also in an iterative manner.

D.2 Backtracking Line Search

One way to do line search is exact line search. Thus the parameter t is chosen to minimize $f(\alpha)$ along the line $\{\alpha + t\Delta\alpha \mid t \geq 0\}$

$$t = \arg\min_s \{f(\alpha + s\Delta\alpha)\}. \tag{D.17}$$

But most line searches in practical use are inexact. A very simple method is called backtracking line search which uses two constants γ and γ' with $0 < \gamma < 0.5$ and $0 < \gamma' < 1$. It is iterated as follows Starting with $t = 1$ given a descent direction $\Delta\alpha$ for f at $\alpha \in \mathbf{dom} f$, backtracking line search iterates as follows

while: $f(\alpha + t\Delta\alpha) > f(\alpha) + \gamma t \nabla f(\alpha)\Delta\alpha$

- $t = \gamma' t$.
- Update α.

Bibliography

[1] C. M. Bishop, in Pattern Recognition and Machine Learning, M. Jordan, J. Kleinberg, and B. Schlkopf, eds., Springer, New York, 1st ed., 2006, pp. 383–393.

[2] K. Blekas, A. Likas, N. P. Galatsanos, and I. E. Lagaris, *A spatially constrained mixture model for image segmentation*, Neural Networks, IEEE Transactions on, 16 (2005), pp. 494–498.

[3] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, March 2004.

[4] B. Braathen and W. Pieczyński, *Global and local methods of unsupervised bayesian segmentation of images*, Machine Graphics and Vision, 2 (1993), pp. 39–52.

[5] M. Brambilla, R. Matheoud, C. Secco, G. Loi, M. Krengli, and E. Inglese, *Threshold segmentation for pet target volume delineation in radiation treatment planning: The role of target-to-background ratio and target size*, Medical physics, 35 (2008), pp. 1207–1213.

[6] S. L. Breen, J. Publicover, S. D. Silva, G. Pond, K. Brock, B. OSullivan, B. Cummings, L. Dawson, A. Keller, J. Kim, J. Ringash, E. Yu, A. Hendler, and J. Waldron, *Intraobserver and interobserver variability in gtv delineation on fdg-pet-ct images of head and neck cancers*, International Journal of Radiation Oncology*Biology*Physics, 68 (2007), pp. 763 – 770.

[7] S. Chib and E. Greenberg, *Understanding the metropolis-hastings algorithm*, The American Statistician, 49 (1995), pp. 327–335.

[8] T. M. Cover and J. Thomas, *Elements of Information Theory*, Wiley, 1991.

[9] J.-F. Daisne, M. Sibomana, A. Bol, T. Doumont, M. Lonneux, and V. Grgoire, *Tri-dimensional automatic segmentation of pet volumes based on measured source-to-background ratios: influence of reconstruction algorithms*, Radiotherapy and Oncology, 69 (2003), pp. 247 – 250.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin, *Maximum likelihood from incomplete data via the em algorithm*, JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B, 39 (1977), pp. 1–38.

[11] Y. E. Erdi, O. Mawlawi, S. M. Larson, M. Imbriaco, H. Yeung, R. Finn, and J. L. Humm, *Segmentation of lung lesion volume by adaptive positron emission tomography image thresholding*, Cancer, 80 (1997), pp. 2505–2509.

[12] Y. E. Erdi, K. Rosenzweig, A. K. Erdi, H. A. Macapinlac, Y.-C. Hu, L. E. Braban, J. L. Humm, O. D. Squire, C.-S. Chui, S. M. Larson, and E. D. Yorke, *Radiotherapy treatment planning for patients with non-small cell lung cancer using positron emission tomography (pet)*, Radiotherapy and Oncology, 62 (2002), pp. 51 – 60.

[13] W. Feller, *An Introduction to Probability Theory and Its Applications*, vol. 1, Wiley, January 1968.

[14] ——, *An introduction to probability theory and its applications. Vol. II.*, Second edition, John Wiley & Sons Inc., New York, 1971.

[15] J. A. Fessler and A. O. Hero, *Space-alternating generalized expectation-maximization algorithm*, IEEE Trans Signal Process, 42 (1994), pp. 2664–2677.

[16] ——, *Penalized maximum-likelihood image reconstruction using space-alternating generalized em algorithms*, IEEE Tr. Im. Proc, 4 (1995), pp. 1417–1429.

[17] C. Greco, K. Rosenzweig, G. L. Cascini, and O. Tamburrini, *Current status of pet/ct for tumour volume definition in radiotherapy treatment planning for non-small cell lung cancer (nsclc)*, Lung Cancer, 57 (2007), pp. 125–134.

[18] H. Gribben, P. Miller, H. Wang, K. Carson, A. Hounsell, and A. Zatari, *Automated map-mrf em labelling for volume determination in pet*, in 5th IEEE Int Symp Biomedical Imaging: From Nano Macro, 2008, pp. 1–4.

[19] C. S. Hamilton and M. A. Ebert, *Volumetric uncertainty in radiotherapy*, Clinical oncology (Royal College of Radiologists(Great Britain)), 17 (2005), pp. 456–464.

[20] M. Hatt, C. Cheze le Rest, A. Turzo, C. Roux, and D. Visvikis, *A fuzzy locally adaptive bayesian segmentation approach for volume determination in pet*, IEEE Trans Med Imag, 28 (2009), pp. 881–893.

[21] M. Hatt, F. Lamare, N. Boussion, C. Roux, A. Turzo, C. Cheze Lerest, P. Jarritt, K. Carson, F. Salzenstein, C. Collet, and D. Visvikis, *Fuzzy hidden markov chains segmentation for volume detemination and quantitation in pet*, Phys Med Biol, 52 (2007), pp. 3467–3491.

[22] H. Herzog, L. Tellmann, C. Hocke, U. Pietrzyk, M. E. Casey, and T. Kuwert, *Nemanu2-2001 guided performance evaluation of four siemens ecat pet-scanners*, IEEE Trans Nucl Sci, 51 (2004), pp. 2662–2669.

[23] W. Jentzen, L. Freudenberg, E. G. Eising, M. Heinze, W. Brandau, and A. Bockisch, *Segmentation of pet volumes by iterative image thresholding*, Journal of nuclear medicine, 48 (2007), pp. 108–114.

[24] Z. Kato, J. Zerubia, M. Berthold, and W. Pieczyński, *Unsupervised adaptive image segmentation*, in IEEE Int Conf Acoustics, Speech and Signal Processing, vol. 4, 1995, pp. 2399–2402.

[25] S. M. Kay, *Fundamentals of statistical signal processing. [Volume I]. , Estimation theory*, Prentice Hall signal processing series, Prentice Hall, Upper Saddle River (N.J.), 1993. Autre rimpr. 2013.

[26] B. Knäusl, A. Hirtl, G. Dobrozemsky, H. Bergmann, K. Kletter, R. Dudczak, and D. Georg, *Pet based volume segmentation with emphasis on the iterative truex algorithm*, Zeitschrift für medizinische Physik, 22 (2011), pp. 29–39.

[27] A. N. Kolmogorov, *Foundations of the Theory of Probability*, Chelsea Pub Co, 2 ed., June 1960.

[28] S. Lakshmanan and H. Derin, *Simultaneous parameter estimation and segmentation of gibbs random fields using simulated annealing*, IEEE Trans Pattern Anal Mach Intell, 11 (1989), pp. 799–813.

[29] T. Layer, M. Blaickner, B. Knausl, D. Georg, J. Neuwirth, R. Baum, C. Schuchardt, S. Wiessalla, and G. Matz, *Pet image segmentation using a gaussian mixture model and markov random fields*, EJNMMI Physics, 2 (2015), p. 9.

[30] J. A. Lee, *Segmentation of positron emission tomography images: some recommendations for target delineation in radiation oncology*, Radiother Oncol, 96 (2010), pp. 302–307.

[31] S. Z. Li, *Markov Random Field Modeling in Image Analysis*, Springer, Beijing, 3th ed., 2009.

[32] D. MacKay, in Information Theory, Inference, and Learning Algorithms, Cambridge University Press, Cambridge, 4th ed., 2005, pp. 300–310.

[33] M. MacManus, U. Nestle, K. E. Rosenzweig, I. Carrio, C. Messa, O. Belohlavek, M. Danna, T. Inoue, E. Deniaud-Alexandre, S. Schipani, N. Watanabe, M. Dondi, and B. Jeremic, *Use of pet and pet/ct for radiation therapy planning: Iaea expert report 20062007*, Radiotherapy and Oncology, 91 (2009), pp. 85 – 94.

[34] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*, Wiley, New York, 1997.

[35] D. W. G. Montgomery, A. Amira, and H. Zaidi, *Fully automated segmentation of oncological pet volumes using a combined multiscale and statistical model*, Med Phys, 34 (2007), pp. 722–736.

[36] T. K. Moon, *The expectation-maximization algorithm*, IEEE Signal Processing Mag, 13 (1996), pp. 47–60.

[37] K. Morarji, A. Fowler, S. K. Vinod, I. Ho Shon, and J. M. Laurence, *Impact of fdg-pet on lung cancer delineation for radiotherapy*, journal of medical imaging and radiation oncology, 56 (2012), pp. 195–203.

[38] A. Papoulis, *Probability & Statistics*, Prentice-Hall international editions, Prentice Hall, 1990.

[39] W. Pieczyński, *Statistical image segmentation*, Machine Graphics and Vision, 1 (1992), pp. 261–268.

[40] B. Reutter, G. J. Klein, and R. H. Huesman, *Automated 3-d segmentation of respiratory-gated pet transmission images*, IEEE Trans Nucl Sci, 44 (1997), pp. 2473–2476.

[41] C. Riddell, P. Brigger, R. E. Carson, and S. L. Bacharach, *The watershed algorithm: a method to segment noisy pet transmission images*, IEEE Trans Nucl Sci, 46 (1999), pp. 713–719.

[42] M. Soret, S. Bacharach, and B. I., *Partial-volume effect in pet tumor imaging*, Journal of nuclear medicine, 48 (2007), pp. 932–45.

[43] D. C. Stanford and A. E. Raftery, *Approximate bayes factors for image segmentation: The pseudo-likelihood information criterion (plic)*, 2002.

[44] C. Sutton and A. Mccallum, *An introduction to conditional random fields*, in Foundations and Trends in Machine Learning, 2012.

[45] P. Tylski, S. Stute, N. Grotus, K. Doyeux, S. Hapdey, I. Gardin, B. Vanderlinden, and I. Buvat, *Comparative assessment of methods for estimating tumor volume and standardized uptake value in (18)f-fdg pet*, Journal of nuclear medicine, 51 (2010), pp. 268–276.

[46] S. Vauclin, K. Doyeux, S. Hapdey, A. Edet-Sanson, P. Vera, and I. Gardin, *Development of a generic thresholding algorithm for the delineation of 18fdg-pet-positive tissue: application to the comparison of three thresholding models*, Physics in medicine and biology, 54 (2009), pp. 6901–6916.

[47] M. Wainwright and M. Jordan, *Graphical Models, Exponential Families, and Variational Inference*, Foundations and Trends in Machine Learning, 1 (2008), pp. 1–305.

[48] M. J. Wainwright and M. I. Jordan, *A variational principle for graphical models*, 2005.

[49] L. Xu and M. I. Jordan, *On convergence properties of the em algorithm for gaussian mixtures*, Neural Computation, 8 (1995), pp. 129–151.

[50] L. A. Zadeh, *Fuzzy sets*, Information and Control, 8 (1965), pp. 338–353.

List of Abbreviations

BG	background
CYL	cylindrical outer body of the NEMA IEC body phantom
DICOM	digital imaging and communications in medicine
EBRT	external beam radiation therapy
EM	expectation maximization
EMGMM	expectation maximization for a Gaussian mixture model
FG	foreground
GMM	Gaussian mixture model
GMRF	Gaussian Markov random field
GMRFP	Gaussian Markov random field with pseudo likelihood
GRF	Gibbs random field
MAP	maximum a posteriori
ML	maximum likelihood estimator
MLGM	maximum likelihood estimation for a Gaussian Model
MLGMGC	maximum likelihood estimation for a Gaussian Model with global covariances
MLGMLC	maximum likelihood estimation for a Gaussian Model with local covariances
MMSE	minimum mean sqare error
MRF	Markov random field
NECR	noise equivalent counting rate
NEMA	national electrical manufacturers association
PEMGMM	penalized expectation maximization for a Gaussian mixture model
PET	positron emission tomography
PSF	point spread function
PVE	partial volume effect
SBR	signal to background ratio
SPECT	single photon emission computer tomography
SPH	spehrical inlays of the NEMA IEC body phantom
VBGMM	variational Bayesian Gaussian mixture model
VOI	volume of interest