TECHNISCHE
UNIVERSITÄT
WIEN

MASTER'S THESIS

Compressed Sensing Recovery with Bayesian Approximate Message Passing using Empirical Least Squares Estimation without an Explicit Prior

By

Abdulrazak Tajjar

June, 2017

Advisor:

Univ.Prof. Dipl.-Ing. Dr.-Ing. Norbert Görtz

Abstract

Compressed Sensing (CS) is a signal processing technique that allows for high-quality reconstruction of a source signal vector of dimension N from a number M << N of linear measurements (undersampling!), and Approximate Message Passing (AMP) is a recovery technique that works particularly well at very low complexity. It has been shown that CS recovery in the AMP framework can be seen as recovery of the N independent and identically distributed (iid) components of the source signal in a decoupled measurement model, with a Gaussian noise of a variance that is estimated adaptively during the AMP-iterations; this variance contains the measurement noise as well as extra noise due to undersampling (i.e. M/N<<1). Hence, CS recovery in the AMP framework boils down to estimation of a signal in Gaussian noise of known variance.

The Bayesian version of AMP (BAMP), which has the best performance, seems to require knowledge of the probability density function (pdf) of the signal prior; this may be seen as a major drawback in practice. It is known, however, that observations of a signal corrupted by Gaussian noise can be de-noised without knowing the signal prior explicitly, and the performance can be very close to that of the optimal Bayesian estimator knowing the pdf of the signal components.

The contribution of this thesis is the use of kernel density estimator as an Empirical Bayes Least Squares Estimator in the BAMP recovery framework and to compare the performance with other, "semi-blind" approaches that exploit partial knowledge of the signal prior, e.g. schemes that assume a shape of the prior pdf and optimize for its parameters from the observed data.

Acknowledgments

I would like to take a moment to thank my supervisor, Norbet Görtz, for being of great help and support since the very beginning of my journey in the institute of Telecommunications. He has guided and assisted me throughout the whole thesis. Overall, I felt encouraged by his positivity, driven by his high standards, and inspired by his work ethic. Most importantly, his confidence in my potential made me work harder, which led me to where I am today.

I would also like to thank all my colleagues and friends, whom made Austria feel like home, special thanks goes to Kenneth Otalor, a true friend to rely on, no matter the circumstances, also my university friends for their interaction, collaboration and sense of humor, Not forgetting my dear friends all over the world for their support, encouragement, and for believing in me. Special thanks to Raed Tawil for standing by my side through difficult times, and helping to put me on the right track.

Last, but not least, no words can really describe my appreciation and admiration to my parents who did all what could be done and even more for me to make it to Austria and pursue a degree, even though my homeland is experiencing a crisis and my family is facing a hard time, it felt like I'm the one who is struggling, while they are relaxed due to their positivity. Deepest respect to my father, who trusts my ambitions and gave me the work ethic to achieve them; my mother, the biggest source of love and hope, not forgetting the wisdom and experience of both of them. Having you all made my dreams come true.

Table of Contents **Page**

Statement of Originality

I hereby certify that the work reported in this thesis is my own, and the work done by other authors is appropriately cited.

Chapter 1: Introduction

In recent years, high-dimensional datasets brought challenges such as complexity, along with numerous opportunities to be exploited. “Though many signals of interest live in a high-dimensional ambient space, they often have a much smaller inherent dimensionality, which, if leveraged, lead to improved recoveries”[7]. For example, the notion of sparsity is a requisite in the compressive sensing (CS) field, which allows for accurate signal reconstruction from sub-Nyquist sampled measurements given certain conditions.

Compressed sensing works on the concept of undersampling a certain high-dimensional signal, while exploiting their characteristics to accurately reconstruct them. In case of a sufficiently sparse object in a known basis, accurate reconstruction is possible. Currently, convex optimization recovery offers a good sparsity–undersampling tradeoff, but it is expensive in important large-scale applications, so as an alternative, a simple cost-efficient modification to iterative thresholding has been introduced, as the new iterative-thresholding algorithms are inspired by belief propagation in graphical models.

“When recovering a sparse signal from noisy compressive linear measurements, the distribution of the signal’s non-zero coefficients can have a profound effect on recovery” [10]. If the distribution is apriori known, computationally efficient Bayesian approximate message passing (AMP) techniques could be used that yield approximate minimum mean square error (MMSE) estimates or critical points to the maximum a posteriori (MAP) estimation problem. In practice, though, the distribution is unknown, which leads us to perform MMSE estimation without knowledge of signal’s prior using some approximations and applying Bayesian approximate message passing (BAMP) for recovery. A comparison with some other schemes that know the prior will be made to show that even though it is not as good as the other schemes which know the prior, it still gives good results without any assumptions.

This chapter is mainly based on [1] and [7]

1.1 Structured Sparse Recovery

1.1.1 Linear Observation Model

The aim is to infer a signal $x \in R^N$ from noisy measurements $y \in R^M$. In addition, we use signals that share a common low dimensional structure, called sparsity, where only a fraction of the signal elements are non-zero. We call a signal K-sparse if it has K non-zero elements.

We next introduce the sensing matrix A of dimension m x n with m<n, i.e.

$$y= Ax + w \tag{1.1}$$

The matrix A is assumed to have its full possible rank m and its components are independently drawn realizations of a real random variable that, e.g., has a Gaussian distribution. Moreover, the matrix column vectors A_j, j = 1,2,...,n (each of dimensions m x 1) are assumed to have zero mean and be normalized to unit l_2-norm , i.e.,

$$A = \{A_1, A_2,..., A_n\}, \tag{1.2}$$

With

$$||A_j||_2= 1 \ \forall \ j. \tag{1.3}$$

The measurements are assumed to be affected by Gaussian noise that is modeled by the addition of the m–dimensional noise vector $w \doteq \{w_j, j = 1,2,...,m\}$. The components w_j of the noise vector are assumed to be iid Gaussian with variance $\sigma^2 > 0$.

In traditional inverse problems, the signal is first recovered from measurements obtained from Shannon-Nyquist sampling (i.e., $M \geq N$)) via standard tools, e.g., least squares recovery. It can then be subsequently compressed, exploiting its underlying sparsity. Compressive sensing (CS), attempts to recover the sparse signal x using fewer measurements than unknowns, i.e., $M < N$. whereas signal recovery in the Shannon-Nyquist paradigm compresses after sampling, CS attempts to compress while sampling.

The problem is to find the vector x given the measurements y and the matrix A. As the number of measurements m in (1.1) is smaller than the number of unknowns n in x, the problem is underdetermined. However, one tries to form a "good estimate" of x by exploiting extra information about x that may be available. Such "prior knowledge" can be of various types [10]:

1) If x is "sparse" the solution $\hat{x}$ would be supposed to contain "few" non-zero components.
2) The vector x might be known to have a "large" number of components that take the values ∓ 1.
3) The probability distribution $p_X(x)$ of x might be known.

The focus will be on case 3). In practice, the "given" $p_X(x)$ may, however, not be the "true" distribution" but it may be a good model for it or even only be useful to find a good solution to the problem of estimating x from a given observation vector y. the remaining are covered when choosing a suitable distribution $p_X(x)$.

Regarding the design of the measurement matrix A, for simplicity we stick to the "random Gaussian" design but we keep in mind that AMP and BAMP will also work for non-Gaussian designs.

Normalizing the matrix A can simplify notation, it is however, not necessarily required.

1.2 Review of Vector Spaces

Since our signal is of a linear model, we model it as a vector living in an appropriate vector space, which in return preserves the linear structure we are after. Going back to (1.1), the noisy measurement is the addition of both the signal we are to recover and the noise, knowing that both are of linear structure and will in return give a linear measurement vector. As another important feature, vector spaces pose powerful tools, such as lengths, distances and angels, to describe and compare signals of interest irrespective of our signal dimension.

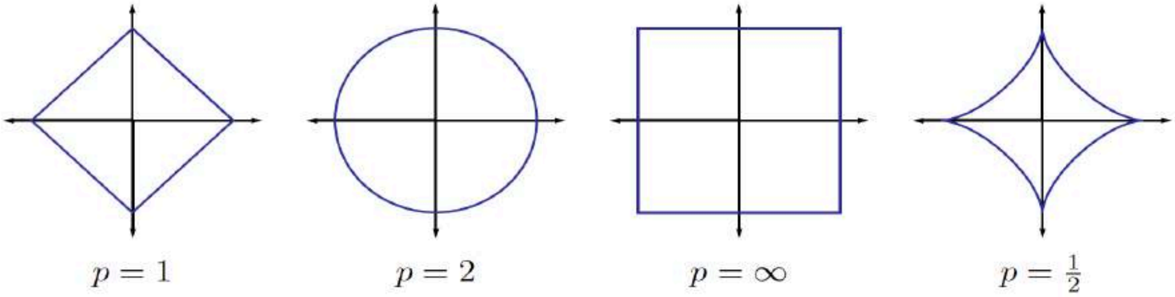

Figure 1.1 Unit spheres in R^2 for the l_p norms with $p = 1; 2; \infty$, and for the l_p quasinorm with $p = {}^1\!/_2$.

1.2.1 Normed vector spaces

The signal we are working on considered as real-valued function having domains that are either continuous or discrete, and either infinite or finite. In our simulations (chapter 5) we normalized the columns of the sensing matrix A using l_2-norm, so we briefly discuss normed vector spaces to make the reader familiar with it. Signals in general can be viewed as vectors in an n-dimensional Euclidean space, denoted by R^n. When dealing with vectors in R^n, frequent use of the l_p norms will be made, which are defined for $p \in [1,\infty]$ as

$$||x||_P = \begin{cases} (\sum_{j=1}^{n} |x_j|^P)^{\frac{1}{P}} \; p \in [1,\infty] \\ \max_i |x_i| \; p = \infty \end{cases} \tag{1.4}$$

The l_p (quasi-)norms have notably different properties for different values of p . To illustrate this, in Figure 1.1 we show the unit sphere, i.e., {x: $||x||_P = 1$ } induced by each of these norms in R^2[1].

It is also good to consider the standard inner product in R^n, which is denoted as

$$\langle x, z\rangle = z^T x = \sum_{i=1}^{n} x_i z_i \qquad (1.5)$$

This inner product leads to the l_2 norm: $||x||_2 = \sqrt{\langle x, x\rangle}$
Some contexts extend the notion of l_p norms to p < 1, which fails to satisfy the triangle inequality in (1.4). Such a norm is called quasi-norm. Also we introduce the notation $||x||_0 := |supp(x)|$, where supp(x)= {i: $x_i \neq 0$ } denotes the support of x and |supp(x)| denotes the cardinality of supp(x) and is frequently used in our Approximate Message Passing (AMP) recovery scheme . It is easily showed that $\lim_{p\to 0}\|x\|_p^p$= |supp(x)|,

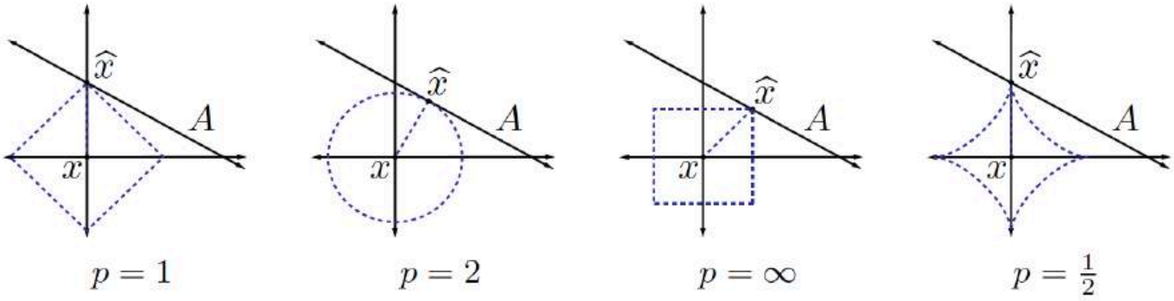

Figure 1.2 Best approximation of a point in R^2 by a one-dimensional subspace using the l_p norms for p = 1, 2, ∞, and the l_p quasinorm with p = 1/2.

Typically norms are used to measure the strength of a signal, or the size of an error. Normally the aim is to minimize the error $||x - \hat{x}||_p$. The choice of p will significantly affect the properties of the resulting approximation error. An example is illustrated in Figure 1.2. [1] to compute the closest point in A to x using each norm. We notice that larger p spread out the error more evenly, while smaller p unevenly distributes the error (sparse).

1.3 Low-Dimensional Signal Models

Finding efficient algorithms is of big importance in signal processing, in order to acquire, process, and extract information from different types of signals.

Accurate models for specific signals are normally needed to design such algorithms. In our work we exploited probabilistic Bayesian model to incorporate the apriori information to recover the signal of interest.

High dimensional signals suffer from small number of degrees of freedom compared to their ambient dimensionality, which led to showing interest in low dimensional signal models.

In the following section we will give a brief overview of the most common low-dimensional structures encountered in the field of CS.

1.3.1 Sparse models

A signal is called sparse if we can approximate it as a linear combination of just few elements from a known basis. Sparse signal models present the fact that high dimensional signals contain relatively little information compared to their dimension.

1.3.1.1 Sparsity and nonlinear approximation

Mathematically, we say that a signal x is k -sparse when it has at most k nonzero [1], i.e., $||x||_0 \leq$ k. We let

$$\sum_k = \{x : \ ||x||_0 \leq \ \mathrm{k}\} \tag{1.6}$$

denote the set of all k-sparse signals. The signal we are dealing with is not sparse, but can be represented as one. So our signal x will be referred to as k-sparse, with x = Φc where $||c||_0 \leq$ k.

Sparsity is already exploited in compression, denoising, and image processing. Figure 1.3. [1] shows an image and its best K-term approximation, as Large coefficients are represented by light pixels, while small coefficients are represented by dark pixels. Note that most of the wavelet coefficients are close to zero.

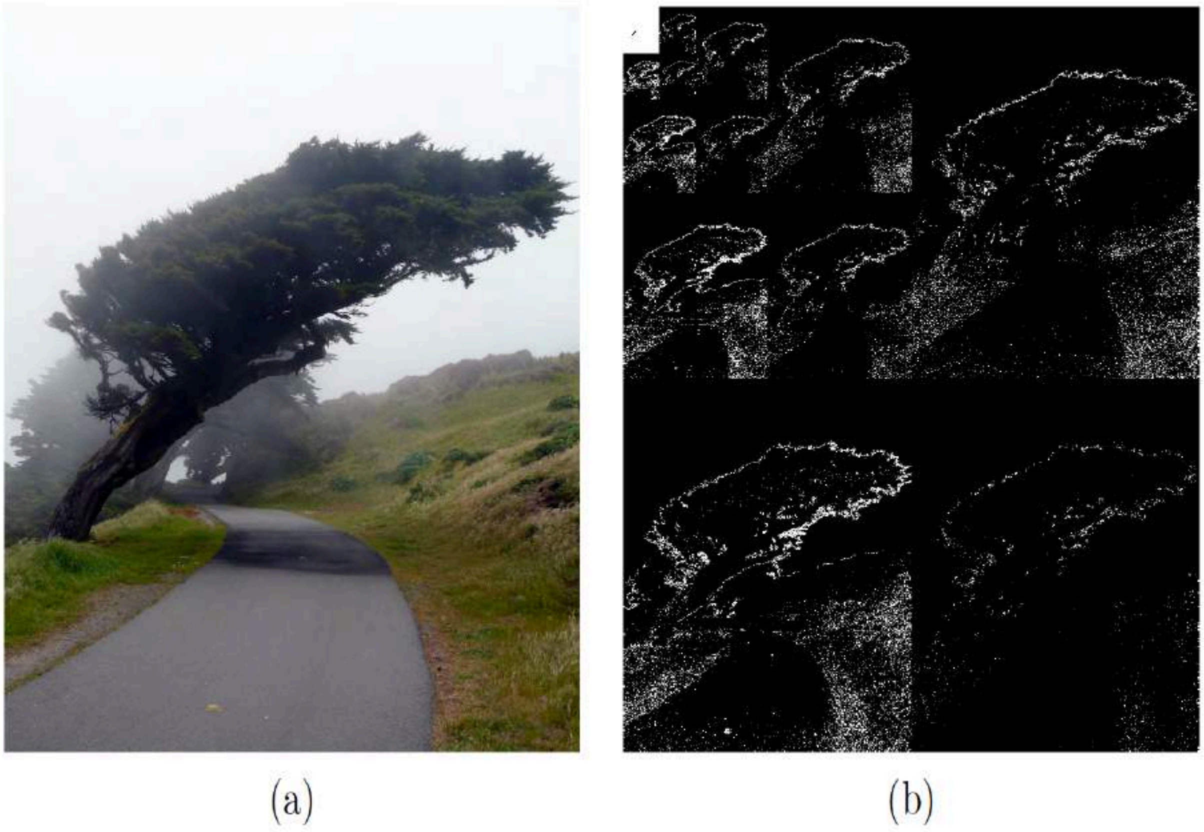

Figure 1.3 Sparse representation of an image. (a) Original image. (b) Wavelet representation.

1.3.1.2 Compressible signals

In practice only few signals are strictly sparse, or can be well approximated by a sparse model. Such signals are often said to be compressible. "The compressibility of a signal can be quantified by calculating the error incurred by approximating a signal” [1] x by some $\hat{\mathrm{x}} \in \sum_{\mathrm{k}}$ as:

$$\sigma_{\mathrm{k}}(\mathrm{x})_{\mathrm{p}} = \min_{\hat{\mathrm{x}} \in \sum_{\mathrm{k}}} ||\mathrm{x} - \hat{\mathrm{x}}||_{\mathrm{p}} \tag{1.7}$$

If $\mathrm{x} \in \sum_{k}$, then clearly $\sigma_{\mathrm{k}}(\mathrm{x})_{\mathrm{p}} = 0$ for any p. Moreover, the term used in (1.7) results in optimal approximation for the thresholding used in Figure 1.3

1.4 Sensing Matrix

As discussed in section 1.1 the sensing matrix A of dimension m x n is used as part of our process to recover the desired signal. The reason for that is to map R^{n} to R^{m}, where m<<n. The measurement is assumed to be non-adaptive, i.e. independent of previous measurements.

Normally, we are interested in designing the sensing matrix A in such a way that it preserves the information in the signal x. To be able to do so, generally several properties are desirable to serve our purpose, such as null space condition, restricted isometry property, and coherence[1].

[1] Properties of matrix A are explained in reference [1]

Chapter 2: Approximate Message Passing

This chapter is excerpted from [2], [3], and [4]

AMP is an iterative algorithm that solves a linear inverse problem by successively converting matrix channel problems into scalar channel denoising problems with additive white Gaussian noise (AWGN). “AMP has received considerable attention, because of its fast convergence and the state evolution (SE) formalism, which offers a precise characterization of the AWGN denoising problem for each iteration”[16].

Let’s assume that a vector y of m measurements is obtained from an unknown n-vector x according to:

$$y = Ax + w, \quad m << n \tag{2.1}$$

Where A = $\{A_1, A_2, \dots, A_n\}$ is the m x n measurement matrix which assumes normalized column vectors $\|A_j\|_2$, and w is the measurement noise vector with dimension m and variance σ_w^2. In practice both the pdf and sparsity are not known, and have to be estimated.

Depending on the measurement noise variance σ_w^2 we have two algorithms

2.1 Approximate Message Passing Algorithms

2.1.1 AMP Algorithm I

The presented math for the discussed algorithms is mainly taken from [4]

In this algorithm we assume that the measurement noise variance σ_w^2 is known for iterations i = 1, 2, ...

$$\hat{x}^{(i)} = \eta\,(\hat{x}^{(i-1)} + A^T z^{(i-1)};\ \sqrt{\beta c^{(i-1)}}); \quad i = 1,2,\dots \tag{2.2}$$

$$b^{(i-1)} = \frac{1}{m}\ ||\hat{x}^{(i)}||_0 \tag{2.3}$$

$$z^{(i)} = y - A\hat{x}^{(i)} + b^{(i-1)}z^{(i-1)} \tag{2.4}$$

$$c^{(i)} = \sigma_w^2 + c^{(i-1)}b^{(i-1)} \tag{2.5}$$

Where β is a regularization parameter.

Initializations at i = 0:

$$\hat{x}^{(0)} = 0_{nx1} \quad \text{(signal vector; dimension n > m)} \tag{2.6}$$

$$z^{(0)} = y \text{ (dimensions: m x 1)} \tag{2.7}$$

$$c^{(0)} = \sigma_w^2 + \frac{1}{m}\ ||z^{(0)}||_2^2 \quad \text{(scalar)} \tag{2.8}$$

We analyse the stability of the recursion in (2.5) Using Unilateral Z-transform (time index k = i – 1) while considering two cases

Case 1: we assume that the scheme recovers the correct s-sparse solution, with $b^{(i-1)} = \frac{1}{m}\ ||\hat{x}^{(i)}||_0 = \frac{S}{m} < 1$, which is treated as a constant

$$c^{(k)} = \sigma_w^2 + c^{(k-1)}b \tag{2.9}$$

with $c^{(-1)} = \sigma_w^2 + \frac{1}{m}\ ||z^{(0)}||_2^2 > \sigma_w^2$ for k= 0,1, …

$$C(z) = \sigma_w^2 \frac{z}{z-1} + b\ z^{-1}C(z) + bc^{-1} \tag{2.10}$$

Z-transform reads:

$$C(z) = \frac{\sigma_w^2}{1-b}\left(\frac{z}{z-1} - b\frac{z}{z-b}\right) + bc^{-1}\frac{z}{z-b} \tag{2.11}$$

and the solution in the time domain is:

$$c^{(k)} = \frac{\sigma_w^2}{1-b}\left(1 - b^{k+1}\right)\sigma[k] + c^{-1}b^{k+1}\sigma[k] \qquad (2.12)$$

$$c^{(k)} = \frac{\sigma_w^2}{1-b}\sigma[k] + \left(c^{-1} + \frac{\sigma_w^2}{1-b}\right)b^{k+1}\sigma[k] \qquad (2.13)$$

Knowing that $\lim_{k\to\infty} b^{k+1}\sigma[k] = 0$, as $|b| < 1$, we are left with:

$$\lim_{k\to\infty} c^{(k)} = \frac{\sigma_w^2}{1-b} > \sigma_w^2 \qquad (2.14)$$

Where $c^{(k)}$ is the noise variance of the iteration k to be used in the thresholding function, meaning that the effective noise variance $c^{(k)}$ in the n signal components will be greater than the measurement noise variance σ_w^2 in the m measurements.

Case2: we assume that the scheme recovers the correct s-sparse solution, with $b^{(i-1)}$ is not a constant.

$$b^{(i-1)} = \frac{1}{m}\ ||\hat{x}^{(i)}||_0\ \geq 0\ ;\ \max||\hat{x}^{(i)}||_0 = n > m \qquad (2.15)$$

If $||\hat{x}^{(i)}||_0$ is close to m we obtain $b^{(i-1)} \to 1$. This means (k= i-1)

$$c^{(k)} = \frac{\sigma_w^2}{1-b}\sigma[k] + \left(c^{-1} + \frac{\sigma_w^2}{1-b}\right)b^{k+1}\sigma[k] \qquad (2.16)$$

When (1-b) is small, the noise variance of the iteration k ($c^{(k)}$) tends to be very large. Mathematically speaking this is valid, as large c means that in the next iteration most components will be zeros by the soft thresholder η() used in (2.2), so (2.5) re-starts with $c^{(i)} = \sigma_w^2$. For implementation purposes we use

$$b^{(i-1)} = \max\left\{b^{(i-1)}, \frac{m-1}{m}\right\} \qquad (2.17)$$

2.1.2 AMP Algorithm II

This algorithm assumes that the measurement noise variance σ_w^2 is unknown.

$$\hat{x}^{(i)} = \eta\,(\hat{x}^{(i-1)} + A^T z^{(i-1)};\ \sqrt{\beta c^{(i-1)}});\ i= 1,2,\ldots.. \quad (2.18)$$

$$z^{(i)} = y - A\hat{x}^{(i)} + z^{(i-1)} \frac{1}{m} \left\|\hat{x}^{(i)}\right\|_0 \quad (2.19)$$

$$c^{(i)} = \frac{1}{m}\ ||z^{(i)}||_2^2 \quad (2.20)$$

We observe that $b^{(i-1)}$ is not present in this algorithm; also the noise variance is depending entirely on the input vector.

Initializations at i = 0:

$$\hat{x}^{(0)} = 0_{nx1} \quad \text{(signal vector; dimension n > m)} \quad (2.21)$$

$$z^{(0)} = y \quad \text{(dimensions: m x 1)} \quad (2.22)$$

$$c^{(0)} = \frac{1}{m}\ ||z^{(0)}||_2^2 \quad \text{(scalar)} \quad (2.23)$$

After introducing the AMP approach we are ready to work on (2.1) to recover our sparse signal vector x.

The task at hand is to find an estimate $\hat{x}$ that minimizes MSE i.e.

$$\hat{x} = \arg\min_{\tilde{x}} E_x\,\{||X - \tilde{x}||_2^2 | Y = y\} \quad (2.24)$$

Using Bayes' rule the solution for the conditional estimation is:

$$\hat{x} = \int x\, p_{X|Y}(x|y)dx = \frac{1}{p_Y(y)} \int x\, p_{Y|X}(y|x)\, p_X(x)dx \quad (2.25)$$

Where the pdf $p_{Y|X}(y|x)$ describes the noisy measurement process. In case of independent Gaussian noise components w_j with variance σ_w^2 we obtain

$$p_{Y|X}(y|x) = \prod_{K=1}^{m} \frac{1}{\sqrt{2\pi}} e^{\frac{(y_k-(Ax)_k)^2}{2\sigma_w^2}} = \frac{1}{(\sqrt{2\pi\sigma_w^2})^m} \exp\left(-\frac{1}{2\sigma_w^2}\|y-Ax\|_2^2\right) \tag{2.26}$$

and the prior pdf for signal vector x in case of independent components is:

$$p_X(x) = \prod_{j=1}^{n} p_{X_j}(x_j) \tag{2.27}$$

Using (2.26) and (2.27), unconditional pdf $p_Y(y)$ of the observations can be computed by the marginalization

$$p_Y(y) = \int x \, p_{Y|X}(y|x) \, p_X(x)dx \tag{2.28}$$

Substituting (2.28) in (2.25) we get:

$$\hat{x} = \frac{1}{p_Y(y)} \int x \frac{1}{(\sqrt{2\pi\sigma_w^2})^m} \exp\left(-\frac{1}{2\sigma_w^2}\|y-Ax\|_2^2\right) \prod_{j=1}^{n} p_{X_j}(x_j) \, dx \tag{2.29}$$

We assume in our study that the prior is not known (as in practice), but with the restriction that x is “somehow sparse”.

2.2 Bayesian Approximate Message Passing (BAMP)

A version of AMP with exploiting the prior of the signal, which is normally, assumed known, but in reality it is most of the time not known. That’s why we will work with BAMP with unknown prior and try to estimate the prior (We will use kernel density estimation to do so) before using the BAMP algorithm to recover our signal.

Here we present two BAMP algorithms, but our work will focus on only one.

2.2.1 BAMP algorithm I (known measurement noise variance σ_w^2)

Start at iteration i = 0 with the initializations

$$\hat{x}^{(0)} = 0_{nx1} \text{ (signal vector; dimension } n > m) \tag{2.30}$$

$$z^{(0)} = y \text{ (dimensions: m x 1)} \tag{2.31}$$

$$c^{(0)} = \frac{1}{m} ||z^{(0)}||_2^2 \text{ (scalar)} \tag{2.32}$$

Then, for iterations i = 1, 2, ... :

$$u^{(i-1)} = \left\{u_1^{(i-1)}, u_2^{(i-1)}, \dots., u_n^{(i-1)}\right\}^T = \hat{x}^{(i-1)} + A^T z^{(i-1)} \tag{2.33}$$

$$\hat{x}_j^{(i)} = F(u_j^{(i-1)};\ c^{(i-1)}) \tag{2.34}$$

$$v_j^{(i)} = G(u_j^{(i-1)};\ c^{(i-1)}) \tag{2.35}$$

$$q_j^{(i)} = F'(u_j^{(i-1)};\ c^{(i-1)}) \tag{2.36}$$

$$z^{(i)} = y - A\hat{x}^{(i)} + z^{(i-1)} \frac{1}{m} \sum_{j=1}^{n} q_j^{(i)} \tag{2.37}$$

$$\text{with } \hat{x}^{(i)} = \left\{\hat{x}_1^{(i)}, \hat{x}_2^{(i)}, \dots., \hat{x}_n^{(i)}\right\} \tag{2.38}$$

$$c^{(i)} = \sigma_w^2 + \frac{1}{m} \sum_{j=1}^{n} v_j^{(i)} \tag{2.39}$$

Stop iterations, if $\left\|\hat{x}^{(i)} - \hat{x}^{(i-1)}\right\|_2 < \in \left\|\hat{x}^{(i-1)}\right\|_2$, e.g., with $\in = 10^{-2}$

To approximately solve the MMSE problem (with the exact but infeasible solution (2.29)), the scalar operators in ((2.34), (2.35), and (2.36)) must be chosen according to:

$$\mathrm{F}(\mathrm{u_j};\mathrm{c}) = \mathrm{E}_{\mathrm{X_j}}\{\mathrm{X_j}|\mathrm{U_j} = \mathrm{u_j}\} \tag{2.40}$$

$$\mathrm{G}(\mathrm{u_j};\mathrm{c}) = \mathrm{Var}_{\mathrm{X_j}}\{\mathrm{X_j}|\mathrm{U_j} = \mathrm{u_j}\} \tag{2.41}$$

$$\mathrm{F}'(\mathrm{u_j};\mathrm{c}) = \frac{d}{d\mathrm{u_j}}\mathrm{F}(\mathrm{u_j};\mathrm{c}) \tag{2.42}$$

A result of the derivation is that the pdf to compute (2.40), (2.41), (2.42) is given by:

$$\mathrm{p}_{\mathrm{X_j|U_j}}(\mathrm{x_j}|\mathrm{u_j};\mathrm{c}) = \frac{\mathrm{p}_{\mathrm{X_j,U_j}}(\mathrm{x_j}|\mathrm{u_j};\mathrm{c})}{\mathrm{p}_{\mathrm{U_j}}(\mathrm{u_j};\mathrm{c})} = \frac{\mathrm{p}_{\mathrm{U_j|X_j}}(\mathrm{u_j}|\mathrm{x_j};\mathrm{c})\mathrm{p}_{\mathrm{X_j}}(\mathrm{x_j})}{\mathrm{p}_{\mathrm{U_j}}(\mathrm{u_j};\mathrm{c})} \quad \text{j=1,2,…., n} \tag{2.43}$$

(with standard Bayes' rule) where

$$\mathrm{p}_{\mathrm{U_j|X_j}}(\mathrm{u_j}|\mathrm{x_j};\mathrm{c}) = \frac{1}{\sqrt{2\pi\mathrm{c}}}\exp\left(-\frac{1}{2\mathrm{c}}(\mathrm{x_j}-\mathrm{u_j})^2\right) \tag{2.44}$$

The variance c of this Gaussian distribution is computed during the BAMP iterations in (2.39), and c is strictly larger than the variance σ_w^2 of the measurement noise.

"The Gaussian pdf in (2.44) applies for j = 1, 2, ..., n, but we have only m < n measurements, which gives a new decoupled measurement model in n dimensions (instead of m)" [4]

$$\mathrm{u_j} = \mathrm{x_j} + \hat{\mathrm{w}}_\mathrm{j}\ \mathrm{j} \ = \ 1,2,\ldots,\mathrm{n}, \text{and}\ \hat{\mathrm{w}}_\mathrm{j} \ \sim \ \mathrm{N}(0,\mathrm{c}) \tag{2.45}$$

BAMP algorithm II (unknown measurement noise variance)

Start at i = 0 with the initializations

$$\hat{\mathrm{x}}^{(0)} = \ 0_{\mathrm{nx1}} \quad (\text{signal vector; dimension n} \ > \ \mathrm{m}) \tag{2.46}$$

$$\mathrm{z}^{(0)} = \mathrm{y}\ (\text{dimensions: m x 1}) \tag{2.47}$$

$$c^{(0)} = \frac{1}{m} ||z^{(0)}||_2^2 \text{ (scalar)} \tag{2.48}$$

Then, for iterations i = 1, 2,.... : (Stopping rule as for BAMP I)

$$u^{(i-1)} = \hat{x}^{(i-1)} + A^T z^{(i-1)} \tag{2.49}$$

$$\hat{x}_j^{(i)} = F(u_j^{(i-1)};\ c^{(i-1)} \tag{2.50}$$

$$z^{(i)} = y - A\hat{x}^{(i)} + z^{(i-1)} \frac{1}{m} \sum_{j=1}^{n} F'(u_j^{(i-1)};\ c^{(i-1)}) \tag{2.51}$$

$$c^{(i)} = \frac{1}{m} ||z^{(i)}||_2^2 \tag{2.52}$$

Note that the scalar operators F (uj; c), F′ (uj; c) defined in (2.40) and (2.42) are applied component-wise.

Chapter 3: MMSE Estimator Construction

In the previous chapter we mentioned the need to estimate the pdf of our sparse signal. We are interested in doing so using minimal mean square error, which will be in return used by the Bayesian Approximate message passing. This chapter is entirely dependent on [6], and [10] to show the computation of the MMSE estimator relying on the measurement vector.

3.1 Computing MMSE estimates

Assume the signal x to be recovered is iid. Without knowing its pdf, it helps to rely on the observed data to form the Bayesian MMSE estimates of the signal components [6] according to

$$\hat{x}(u) = u + c\frac{P'_U(u)}{P_U(u)} \tag{3.1}$$

This is possible, if we have enough realizations u_l, ℓ = 1, 2,...,L. From (3.1) we notice that both the pdf $P_U(u)$ of the observations and the known variance c of the effective Gaussian noise within BAMP are required to form the estimates $\hat{x}(u)$. Having enough realizations, gives the opportunity to estimate the pdf based on a histogram approach. The drawback is the spikiness of the histogram estimates, which leads to particular difficulties when approximating the derivative $P'_U(u)$.

Our aim is to efficiently compute an estimator function (3.1) from the observed data. “We start by splitting the real axis of u into non overlapping intervals according to figure 3.1”[10], where $g_i < g_{i+1}$ for all indices i = 0, 1,.... , next we approximate the actual pdf within each interval by an exponential function as follows[6]:

$$\widetilde{P_U}(u) = \ \widetilde{q_\imath}e^{-a(u-g_i)} \ g_i \le u \ < g_{i+1} \tag{3.2}$$

where g_i marks the left limit of the interval, and is constant, so:

$$\widetilde{P_U}(u) = q_i e^{-au} \quad g_i \leq u < g_{i+1} \tag{3.3}$$

with a different constant q_i (instead of $\widetilde{q_i}$). This choice is based on the need of the derivative of the log-pdf for our estimator, which is computed as:

$$\frac{d}{du} \log \widetilde{P_U}(u) = \frac{P'_U(u)}{P_U(u)} = \frac{-a_i q_i e^{-a_i u}}{q_i e^{-a_i u}} = -a_i \quad g_i \leq u < g_{i+1} \tag{3.4}$$

this estimator is very simple, and (3.4) implies that, “we first need to identify the interval i to find the right value for a_i to use in the resulting approximate estimator”[10] as:

$$\hat{x}(u) = u + c\frac{P'_U(u)}{P_U(u)} = u - ca_i \quad g_i \leq u < g_{i+1} \tag{3.5}$$

For small intervals, we will get a good approximation, e.g. when the binwidths

$$h_i \doteq g_{i+1} - g_i \tag{3.6}$$

a_i is estimated from the observed data points u_i, with having large number of realizations u_i in each interval. The use of a maximum likelihood (ML) estimator [6] is necessary to estimate a_i. The indices of the data points are written in a set as follows:

$$S_i = \{l \in \{1,2, \dots ., L\} : \; g_i \leq u_l < g_{i+1}\} \tag{3.7}$$

We condition the pdf on the fact that the considered values are in the interval $[g_i, g_{i+1})$, while (3.2) is defined on the whole support of u, so the conditional pdf is:

$$P_{U|U\,\epsilon\,[g_i,g_{i+1})}(u|u\,\epsilon\,[g_i,g_{i+1})) = \frac{P_{U,U\,\epsilon\,[g_i,g_{i+1})}(u,u\,\epsilon\,[g_i,g_{i+1}))}{P_{U,U\,\epsilon\,[g_i,g_{i+1})}(\,u\,\epsilon\,[g_i,g_{i+1}))} \tag{3.8}$$

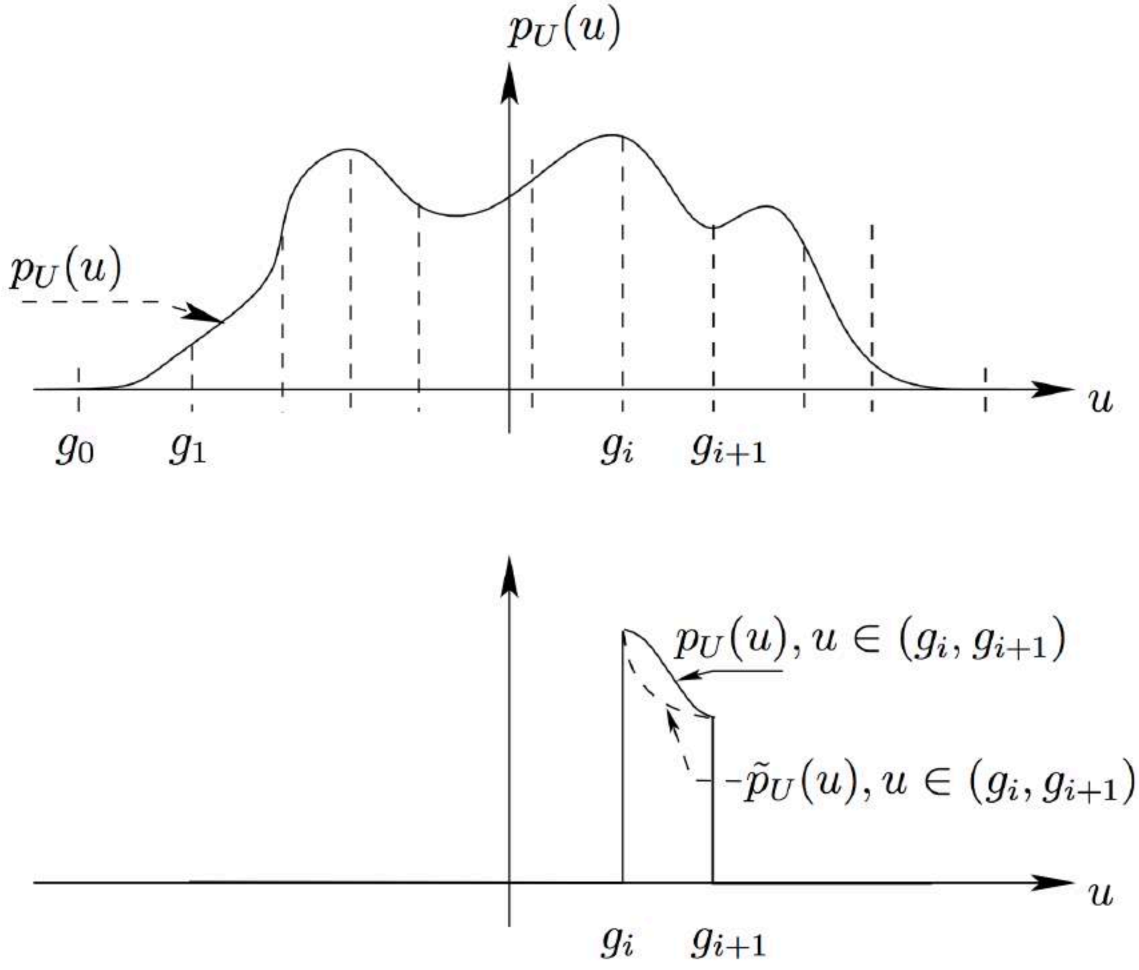
$p_U(u)$
$p_U(u)$
u
g_0
g_1
g_i
g_{i+1}
$p_U(u), u \in (g_i, g_{i+1})$
$\tilde{p}_U(u), u \in (g_i, g_{i+1})$
u
g_i
g_{i+1}

Fig 3.1. Splitting the support of the pdf into non-overlapping intervals and exponential approximation of the pdf.

The denominator is a normalizing probability that can be computed by integration over the interval $[g_i, g_{i+1})$, from the approximate pdf in (3.2), so we get:

$$P_{U|U\,\epsilon\,[g_i,g_{i+1})}(u|u\,\epsilon\,[g_i,g_{i+1})) = \begin{cases} \frac{1}{\int_{q_i}^{q_{i+1}} q_i e^{-a_i u} du} q_i e^{-a_i u} & \text{if } u\ \epsilon\,[g_i,g_{i+1}) \\ 0 & \text{otherwise} \end{cases} \tag{3.9}$$

The upper part can be simplified[2] according to

$$P_{U|U \,\epsilon\, [g_i, g_{i+1})}(u|u \,\epsilon\, [g_i, g_{i+1})) = \quad = \frac{\frac{a_i}{2} e^{-a_i(u-\overline{g_\imath})}}{\sinh a_i h_i/2} \tag{3.10}$$

$$\text{with } \overline{g_\imath} = \frac{1}{2}(g_i + g_{i+1}) \tag{3.11}$$

so the conditional pdf is:

$$P_{U|U \,\epsilon\, [g_i, g_{i+1})}(u|u \,\epsilon\, [g_i, g_{i+1})) = \begin{cases} \frac{\frac{a_i}{2} e^{-a_i(u-\overline{g_\imath})}}{\sinh a_i h_i/2} & \text{if } u \,\epsilon\, [g_i, g_{i+1}) \\ 0 & \text{otherwise} \end{cases} \tag{3.12}$$

It integrates to “one” over the whole support of u, with $\overline{g_\imath}$ defined in (3.11) and h_i defined in (3.6). “Next we try to optimize the interval limits g_i, g_{i+1} (which define $\overline{g_\imath}$ and h_i as well as the constant a_i)”[10].

We assume an interval $[g_i, g_{i+1})$, and try to estimate the exponential parameter a_i using an ML estimator. We consider the joint conditional pdf

$$P(u_{S_i}|a_i) \doteq \prod\nolimits_{\forall l \epsilon S_i} P_{U|U \,\epsilon\, [g_i, g_{i+1})}(u_l|u_l \,\epsilon\, [g_i, g_{i+1})) \tag{3.13}$$

The fact that both the signal and the noise are iid makes it possible to write them as a product. The ML estimator maximizes the value of the joint pdf. Since the log-function is strictly monotone, we maximize as follows:

$$\hat{a}_i = \arg\max\nolimits_{a_i} \log P_{U|U \,\epsilon\, [g_i, g_{i+1})}(u_l|u_l \,\epsilon\, [g_i, g_{i+1})) \tag{3.14}$$

[2] For complete derivation refer to [10]

$$= \arg\max_{a_i} \sum_{\forall l \epsilon S_i} \log P_{U|U\,\epsilon\,[g_i, g_{i+1})}(u_l | u_l \,\epsilon\, [g_i, g_{i+1})) \tag{3.15}$$

$$= \;\; \arg\max_{a_i} \sum_{\forall l \epsilon S_i} \log \frac{\frac{a_i}{2} e^{-a_i(u_l - \bar{g}_\imath)}}{\sinh a_i h_i / 2} \tag{3.16}$$

Using the properties of the log, neglecting the constant ½, and dividing by the constant |Si| we arrive to the following:

$$\hat{a}_i = \arg\max_{a_i} \left\{ \log a_i - a_i\left(\frac{1}{|S_i|} \sum_{\forall l \epsilon S_i} u_l - \bar{g}_\imath\right) - \log \sinh a_i h_i / 2 \right\} \tag{3.17}$$

We also describe the average of the observations uℓ in the interval $[g_i, g_{i+1})$ using the abbreviation:

$$\bar{u}_\imath \doteq \frac{1}{|S_i|} \sum_{\forall l \epsilon S_i} u_l \tag{3.18}$$

To find the maximum, we take the derivative of (3.25):

$$\frac{d}{da_i}\{..\} \; = \frac{1}{a_i} - (\bar{u}_\imath \; - \bar{g}_\imath) - \frac{h_i}{2} \coth\,(a_i h_i / 2) \tag{3.19}$$

then we set the derivative to zero, to find the estimate $\hat{a}_i$, i.e.,

$$\frac{1}{\hat{a}_i} - (\bar{u}_\imath \; - \bar{g}_\imath) - \frac{h_i}{2} \coth\,(\hat{a}_i h_i / 2) = 0 \tag{3.20}$$

so

$$\frac{1}{\hat{a}_i \frac{h_i}{2}} - \coth\,(\hat{a}_i h_i / 2) = \frac{(\bar{u}_\imath - \bar{g}_\imath)}{\frac{h_i}{2}}. \tag{3.21}$$

The difference $(\bar{u}_\imath \; - \bar{g}_\imath)$can at most take the value $\pm h_i/2$. “Hence, the magnitude of the right-hand side of (3.21) can at most take the value of one”[10]. Of course, this also applies to the function $\frac{1}{x} - \coth(x)$ on the left-hand side, which cannot be inverted analytically.

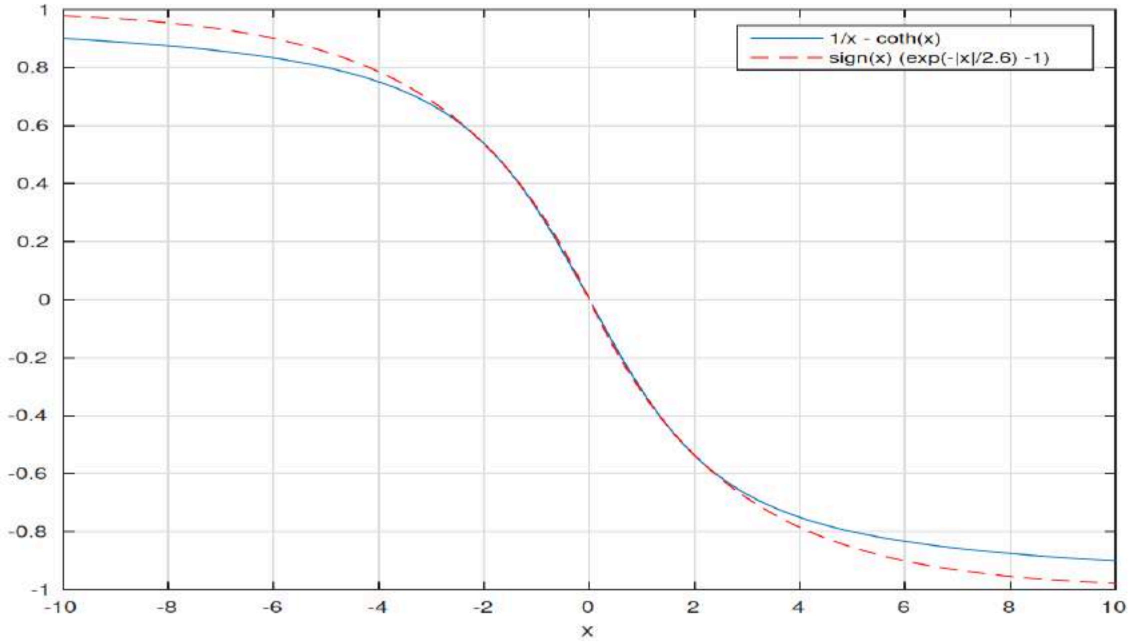

Fig. 3.2. Approximation of $\frac{1}{x}-\coth(x)$ by the invertible function sign(x)(e−|x|/2.6 − 1).

The inversion of (3.21) is difficult to deal with numerically and we look for an approximation. The values of u_i should more or less spread across the whole interval, so we hope $\overline{u_i}$ takes a value close to the center $\overline{g_i}$ of the interval, and then find a good approximation of the odd function $\frac{1}{x}-\coth(x)$ around "zero". "A possible approximation of $\frac{1}{x}-\coth(x)$ by the invertible function sign(x)($e^{-|x|/2.6}$ − 1) is shown in Figure 3.2."[10]. We notice that if right-hand side magnitude of (3.21) is smaller than 0.6, the approximation

$$\frac{1}{x}-\coth(x)=\text{sign}(x)\left(e^{-|x|/2.6}-1\right) \tag{3.22}$$

is indeed very good. Since both the sign-function and the magnitude-function in the exponent don't pose a problem for inversion, as the function is odd, and the negative sign of the argument x and the sign of the result are the same. Hence,

$$\operatorname{sign}\left(\hat{a}_i \frac{h_i}{2}\right)\left(e^{-\left|\hat{a}_i \frac{h_i}{2}\right|/2.6} - 1\right) \approx \frac{1}{\hat{a}_i \frac{h_i}{2}} - \coth\left(\hat{a}_i \frac{h_i}{2}\right) \qquad (3.32)$$

as the function is odd, the approximation simplifies to

$$e^{-\left|\hat{a}_i \frac{h_i}{2}\right|/2.6} - 1 \approx -\left|\frac{(\overline{u_i} - \overline{g_i})}{\frac{h_i}{2}}\right| \qquad (3.23)$$

so

$$-\left|\hat{a}_i \frac{h_i}{2}\right|/2.6 \approx \log\left(1 - \left|\frac{(\overline{u_i} - \overline{g_i})}{\frac{h_i}{2}}\right|\right) \qquad (3.24)$$

Note that, as $\left|\frac{(\overline{u_i} - \overline{g_i})}{\frac{h_i}{2}}\right| < 1$ the log-function will produce a negative value. In the next step we obtain

$$|\hat{a}_i| \approx -\frac{5.2}{h_i}\log\left(1 - \left|\frac{(\overline{u_i} - \overline{g_i})}{\frac{h_i}{2}}\right|\right) \qquad (3.25)$$

taking (3.6), (3.11) and (3.22) into consideration we right the end result as:

$$\hat{a}_i \approx \frac{5.2}{h_i}\ \operatorname{sign}(\overline{u_i} - \overline{g_i})\ \log\left(1 - \frac{2}{h_i}\left|(\overline{u_i} - \overline{g_i})\right|\right) \qquad (3.26)$$

Note that $\overline{u_i}$ is computed from the observed data which falls into the interval $[g_i, g_{i+1})$; the open question is how to choose the intervals.

3.2 Choice of binwidth

To find the optimal binwidth, an MSE estimate must be taken into consideration to figure out how good an estimate is. This estimate can be written as:

$$E\{(\hat{a} - a)^2\} = E\{((\hat{a} - E\{\hat{a}\}) + (E\{\hat{a}\} - a))^2\}$$

$$= \mathrm{Var}\{\hat{a}\} + \ (\mathrm{E}\{\hat{a}\} - a)^2 \qquad (3.27)$$

where â is an estimate of the true value a. The first term is the variance of the estimate and will decrease if we increase the binwidth of the interval. While the second term is the squared bias, which will conversely increase as the interval is increased, thus a bias-variance tradeoff will exist.

For small h (due to large amount of data points N), the number falling in the interval h is approximated as:

$$n \approx \ \widetilde{P_U}(u)Nh \qquad (3.28)$$

Hence,

$$\mathrm{Var}\{\hat{a}\} = \frac{C}{n} \qquad (3.29)$$

where C is an appropriately chosen constant.
Also as h → 0 the bias for the interval will decrease to zero. For small h [6] assumes that

$$(\mathrm{E}\{\hat{a}\} - a)^2 \approx \mathrm{D}h^m \qquad (3.30)$$

where D depends on the shape of $\widetilde{P_U}(u)$. If the density is smooth enough according to [6] the dependence of D on $\widetilde{P_U}(u)$ is ignored, and treated as a constant for all values of u. After theses assumptions the approximation can be written as:

$$\mathrm{E}\{(\hat{a} - a)^2\} \approx \frac{C}{n} + \ \mathrm{D}h^m \qquad (3.31)$$

Deriving the equation and setting it to zero with respect to h gives:

$$h = \left(\frac{C}{Dmn}\right)^{\frac{1}{m-1}} \tag{3.32}$$

(3.32) verifies the assumption that h → 0 as the amount of data (N) increases., and that the MSE, go to zero as N → ∞ . Thus the optimal binwidth is chosen such that the product of the number of points which fall in the interval times some power of the binwidth of the interval is constant [6].

3.3 Simulation Example

This example is taken from [10], where we have 10000 realizations and 47 bins. It shows the histogram of the original data (discrete ±1 source), the noisy measurements, and the recovered data as well as a scatterplot of the same data.

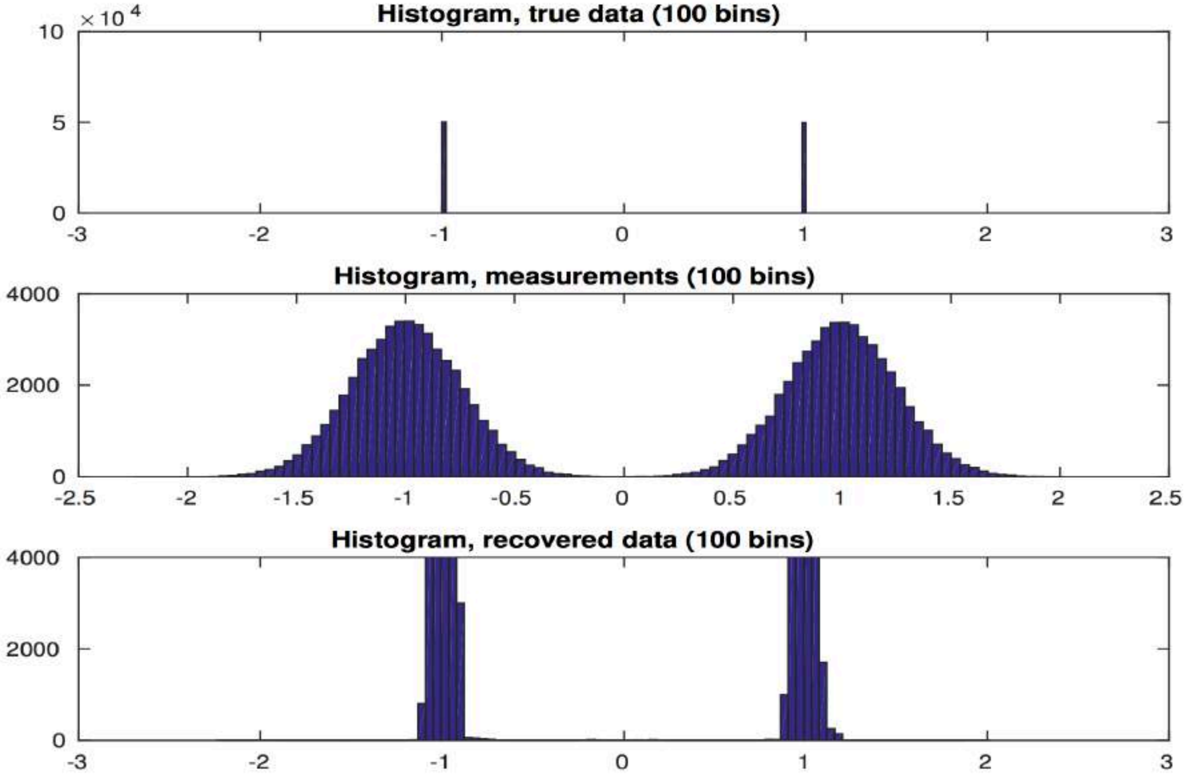

Fig 3.3 Histograms of the original data (discrete ±1 source), the noisy measurements and the recovered data

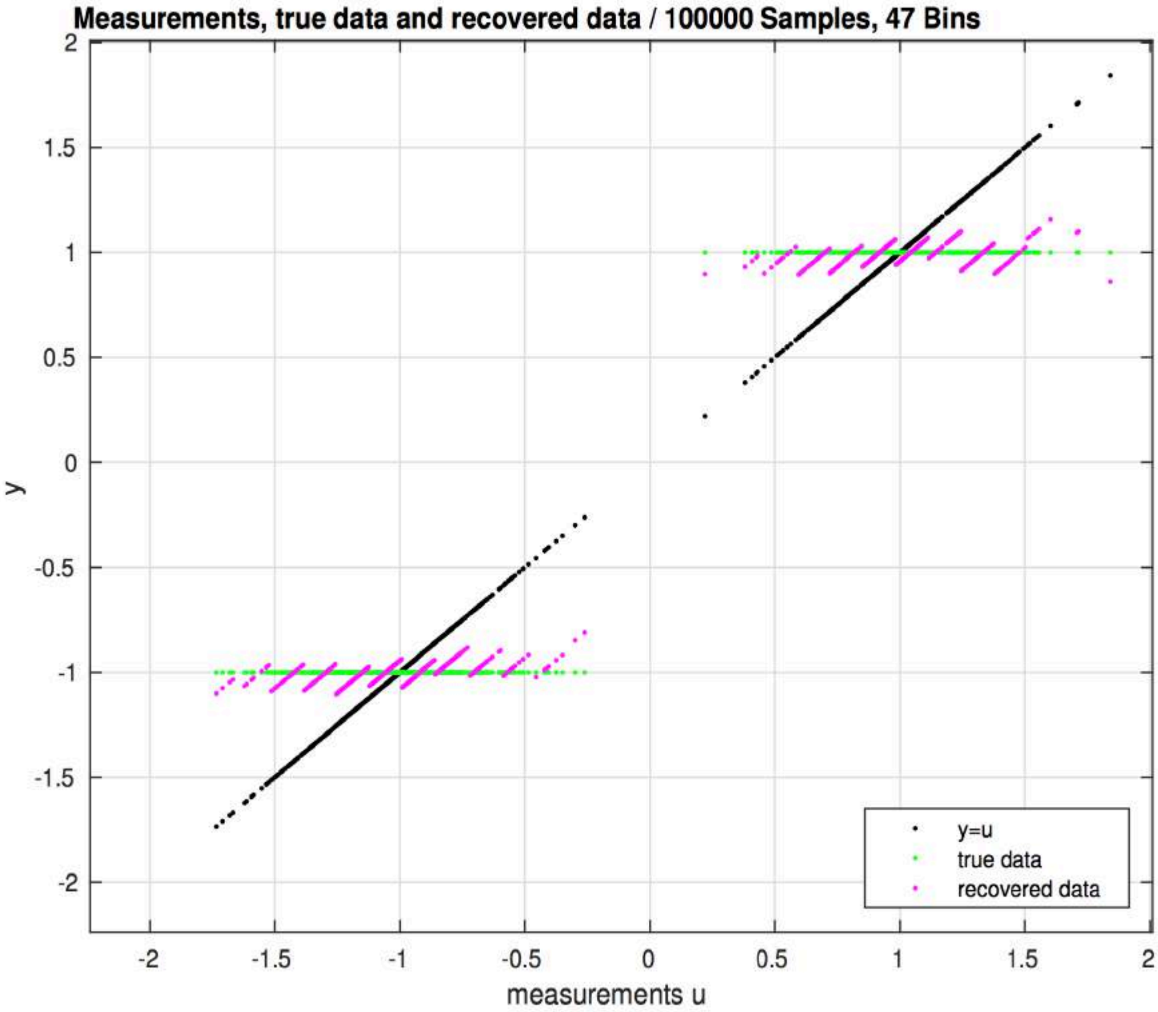

Fig 3.4 scatterplot of the same data as in Fig 3.3

Note that the number of bins is dependent on the number of realizations. It is also worth noticing that Fig 3.4 describes equation (3.5), where subtracting the constant c forms the estimates a_i from the input measurement u, with a_i being computed for each bin according to (3.26). It is a strong simplification, caused by the exponential approximation of the pdf within each interval.

Chapter 4: KERNEL DENSITY ESTIMATION VIA DIFFUSION

4.1 Introduction

Nonparametric density estimation is an important tool in the statistical analysis of data. One example is to estimate the parameters in a specified model via the likelihood principle. The advantage of this approach is the great flexibility in modeling a given dataset. Currently, kernel density estimation is the most popular nonparametric approach for estimation. Despite being an attractive approach for estimation, selecting a technique to estimate the optimal bandwidth is quite dependent on the dataset and it is not always feasible.

Another draw back, is the lack of local adaptivity, which in return results in a large sensitivity to outliers, the presence of spurious bumps, and overall unsatisfactory bias performance- a tendency to flatten the peaks and valleys of the density [8].

Also, in case of nonnegative data, kernel estimators suffer from boundary bias—a phenomenon caused by the fact of not taking specific knowledge about the domain of the data into account. Smoothing the signal we want to recover is of a big concern. One can use kernel density estimation method based on the smoothing properties of linear diffusion processes [8]. The linear diffusion used leads to a kernel estimator with reduced asymptotic bias and mean square error. The proposed estimator deals well with boundary bias [8].

We start by describing the Gaussian kernel density estimator and show that it is a special case of smoothing using a diffusion process. Then, we analyze the asymptotic properties of the general linear diffusion estimator and explain how to compute the asymptotically optimal plug-in bandwidth [8]. The used approach demonstrates an improved bias performance, low computational cost, and a boundary bias improvement.

4.2 Background

Given N independent realizations $\mathcal{X}_N \equiv \{X_1, . . . , X_N\}$ from an unknown continuous probability density function (pdf.) f on $\mathcal{X}$, the Gaussian kernel density estimator is defined as

$$\hat{f}(x; t) = \frac{1}{N}\sum_{1=1}^{N} \emptyset(x, X_i; t) \text{ , } x \in R, \tag{4.1}$$

Where

$$\emptyset(x, X_i; t) = \frac{1}{\sqrt{2\pi t}} e^{-\frac{(x-X_i)^2}{2t}}$$

Is a Gaussian pdf (kernel) with location X_i and scale $\sqrt{t}$. The scale is usually referred to as the bandwidth. Note that the performance of $\hat{f}$ depends crucially on the scale value. The Mean Integrated Squared Error (MISE) is used to determine the optimal t as follows [8]:

$$\text{MISE}\{\hat{f}\}(t) = E_f \int [\hat{f}(x; t) - f(x)]^2 \, dx \text{ ,}$$

The MISE depends on the bandwidth $\sqrt{t}$ and f in a quite complicated way but can be simplified using asymptotic approximation to the MISE (AMISE) "under the consistency requirements that $t = t_N$ depends on the sample size N such that $t_N \downarrow 0$ and N $\sqrt{t_N} \to \infty$ as N $\to \infty$, and f is twice continuously differentiable."[8]

The Gaussian kernel density estimator (4.1) is a unique solution to the diffusion partial differential equation (PDE) [8]

$$\frac{\partial}{\partial t}\hat{f}(x; t) = \frac{1}{2}\frac{\partial^2}{\partial x^2}\hat{f}(x; t), \quad x \in \mathcal{X}, t > 0 \tag{4.2}$$

With X ≡ R and initial condition $\hat{f}(x; 0) = \Delta(x)$where $\Delta(x) \ =\frac{1}{N}\sum_{i=1}^{N}\delta(x-X_i)$ is the empirical density of the data X_N [here δ(x−Xi) is the Dirac measure at Xi] [8]. The Gaussian kernel in (4.1) is the so-called Green's function[3] for the diffusion PDE (4.2). Thus, the Gaussian kernel density estimator $\hat{f}(x; t)$ can be obtained by evolving the solution of the parabolic PDE (4.2) up to time t.

4.3 The diffusion estimator

All the upcoming results are taken from [8][4]. The extension of the simple diffusion model (4.2) is based on the smoothing properties of the linear diffusion PDE [8]:

$$\frac{\partial}{\partial t}g(x;t) = Lg(x;t) \quad x \in \mathcal{X}, t > 0 \tag{4.3}$$

"where the linear differential operator L is of the form $\frac{1}{2}\frac{d}{dx}\left(a(x)\right)\left(\frac{dy}{dx}\left(\frac{\cdot}{p(x)}\right)\right)$, and a and p can be any arbitrary positive functions on X with bounded second derivatives, and the initial condition is g(x, 0) = Δ(x). If the set X is bounded, we add the boundary condition $\frac{\partial}{\partial t}\left(\frac{g(x;t)}{p(x)}\right)$= 0 on ∂X , which ensures that the solution of (4.3) integrates to unity"[8]. The PDE (4.3) describes the pdf of X_t for the Itô diffusion process (X_t, t > 0) [8] given by

$$dX_t = \mu(X_t)\, dt + \sigma(X_t)\, dB_t, \tag{4.4}$$

where the drift coefficient $\mu(x) = \frac{a\prime(x)}{2p(x)}$, the diffusion coefficient $\sigma(x) = \sqrt{\frac{a(x)}{p(x)}}$, the initial state X_0 has distribution Δ(x) , and (B_t, t > 0) is standard Brownian motion [8]. Clearly, if a = 1 and p = 1, we go back to the simpler model (4.2). What makes the solution g(x; t) to (4.3) a plausible kernel density estimator is that g(x; t) is a pdf with the following properties.

[3] You can read more about green's function in [8]

[4] The derivations of the mentioned results can be found in reference [8]

First, g(.; 0) is identical to the initial condition of (4.3), that is, to the empirical density This property is possessed by both the Gaussian kernel density estimator (4.1) and the diffusion estimator (4.3). Second, if p(x) is a pdf on X , then

$$\mathrm{lim}_{t\to\infty}\, g(x;\ t) = p(x)\ , x \in X.$$

In the context of the diffusion process governed by (4.4), p is the limiting and stationary density of the diffusion. Third, similar to the Gaussian kernel density estimator (4.1), the solution of (4.3) can be written as [8]

$$g(x; t) = \frac{1}{N}\sum_{1=1}^{N} \kappa(x, X_i; t) \tag{4.5}$$

Where for each fixed y ∈X the diffusion kernel κ satisfies the PDE

$$\begin{cases} \dfrac{\partial}{\partial x}\kappa(x, y; t) = L\kappa(x, y; t)\ x \in\ \mathcal{X}, t > 0 \\ \kappa(x, y; 0) = \delta(x - y)\ x \in\ \mathcal{X} \end{cases} \tag{4.6}$$

In addition, for each fixed x ∈X the kernel κ satisfies the PDE

$$\begin{cases} \dfrac{\partial}{\partial x}\kappa(x, y; t) = L^{*}\kappa(x, y; t)\ x \in\ \mathcal{X}, t > 0 \\ \kappa(x, y; 0) = \delta(x - y)\ x \in\ \mathcal{X} \end{cases} \tag{4.7}$$

Where L^{*} is of the form $\frac{1}{2p(y)}\frac{d}{dy}(a(y)\frac{dy}{dx}(\,.\,))$ that is, L^{*} is the adjoint operator of L. Note that L^{*} is the infinitesimal generator of the Itô diffusion process in (4.4). If the set $\mathcal{X}$ has boundaries, Neumann boundary condition can be added [8]

$$\frac{\partial}{\partial \mathrm{x}}\left(\frac{\kappa(\mathrm{x},\mathrm{y};\mathrm{t})}{\mathrm{p}(\mathrm{x})}\right)|_{\mathrm{x}\in\partial\mathcal{X}} = 0\ \forall \mathrm{t} > 0 \qquad (4.8)$$

and $\frac{\partial}{\partial \mathrm{y}}\kappa(\mathrm{x},\mathrm{y};\mathrm{t})|_{\mathrm{y}\in\partial\mathcal{X}} = 0$ to (4.6) and (4.7) respectively. These boundary conditions ensure that g(x; t) integrates to unity for all $t \geq 0$.

4.4 Bias and variance analysis

In the following we will examine the asymptotic bias, variance and MISE of the diffusion estimator (4.5). In order to derive the asymptotic properties of the proposed estimator, a small bandwidth behavior of the diffusion kernel satisfying (4.6) is needed [8].

Theorem 1. Let $t = t_N$ be such that lim $N \rightarrow \infty$ $t_N = 0$, lim $N \rightarrow \infty$ $N\sqrt{t_N} = \infty$. Assume that f is twice continuously differentiable and that the domain $X \equiv R$. Then:

1.The pointwise bias has the asymptotic behavior

$$\mathrm{E_F}[\mathrm{g}(.;\mathrm{t})] - \mathrm{f}(\mathrm{x}) = \mathrm{tLf}(\mathrm{x}) + O(\mathrm{t}^2), \quad \mathrm{N} \rightarrow \infty. \qquad (4.9)$$

2. The integrated squared bias has the asymptotic behavior

$$||\mathrm{E_F}[\mathrm{g}(.;\mathrm{t})] - \mathrm{f}||^2 \sim \mathrm{t}^2||\mathrm{Lf}||^2 = \frac{1}{4}\mathrm{t}^2\left\|\mathrm{a}(\left(\frac{\mathrm{f}}{\mathrm{p}}\right)')'\right\|^2, \quad \mathrm{N} \rightarrow \infty. \qquad (4.10)$$

3. The pointwise variance has the asymptotic behavior

$$\mathrm{Var_F}[\mathrm{g}(\mathrm{x};\mathrm{t})] \sim \frac{\mathrm{f}(\mathrm{x})}{2\mathrm{N}\sqrt{\pi \mathrm{t}}\sigma(\mathrm{x})} \quad \mathrm{N} \rightarrow \infty. \qquad (4.11)$$

Where $\sigma^2 = a(x)/p(x)$.

4. The integrated variance has the asymptotic behavior

$$\int \mathrm{Var}_F[g(x;t)]dx \sim \frac{E_f[\sigma^{-1}(x)]}{2N\sqrt{\pi t}} N \ \rightarrow \infty. \qquad (4.12)$$

5. Combining the leading order bias and variance terms gives the asymptotic approximation to the MISE

$$\mathrm{AMISE}\{g\}(t) = \tfrac{1}{4}t^2 \left\| a\left(\left(\tfrac{f}{p}\right)'\right)' \right\|^2 + \tfrac{E_f[\sigma^{-1}(x)]}{2N\sqrt{\pi t}} \qquad (4.13)$$

6.Hence, the square of the asymptotically optimal bandwidth is

$$t^* = {\frac{E_f[\sigma^{-1}(x)]}{2N\sqrt{\pi t}||Lf||^2}}^{2/5} \qquad (4.14)$$

Which gives the minimum

$$\min_t \mathrm{AMISE}\{g\}(t) = N^{-4/5} {\frac{5E_f[\sigma^{-1}(x)]}{2^{14/5}\pi^{2/5}}}^{4/5} ||Lf||^{-2/5} \qquad (4.15)$$

First, if $p \neq f$, the rate of convergence of (4.15) is $O(N^{-4/5})$. According to [8] The multiplicative constant of $N^{-4/5}$, can be made very small by choosing p to be a pilot density estimate of f Preliminary. Second, if $p \equiv f$, then the leading bias term (4.9) is 0. In fact, if f is infinitely smooth, the pointwise bias is exactly zero, as we see from

$$E_F[g(x;t)] = \textstyle\sum_{k=0}^{\infty} \frac{t^k}{k!} L^k f(x), \quad f \in C^{\infty} \qquad (4.16)$$

Where $L^{n+1} = LL^n$ and L^0 is the identity operator. In addition, if a = p ∝ 1, then the bias term (4.9) is equivalent to the bias term (4.19) of the Gaussian kernel density estimator. Third, (4.11) suggests that the ideal variance behavior results when the diffusivity σ(x) behaves inversely proportional to f(x).

4.5 Bandwidth Selection Algorithm

4.5.1 Bandwidth selection for the diffusion estimator

We discuss the bandwidth choice for the diffusion estimator (4.5). “In the following we assume that f is as many times continuously differentiable as needed” [8]. Computation of t^* in (4.14) requires an estimate of $||Lf||^2$ and $E_f[\sigma^{-1}(x)]$. We estimate $E_f[\sigma^{-1}(x)]$ via the unbiased estimator $\frac{1}{N}\sum_{i=1}^{N} \sigma^{-1}(x_i)$. The identity $||Lf||^2 = E_f L^* Lf(X)$ suggests two possible plug-in estimators. The first one is

$$\widehat{E_f L^* Lf} := \frac{1}{N}\sum_{j=1}^{N} L^* Lg(x; t_2)|_{x=X_j} \tag{4.17}$$

where g(x; t2) is the diffusion estimator (4.5) evaluated at t2, and X ≡ R. The second estimator is

$$||\widehat{Lf}||^2 := ||Lg(.; t_2)||^2 \tag{4.18}$$

The optimal t_2^{*}[5] is derived in the same way that $*t_2$[6] is derived for the Gaussian kernel density estimator. That is, t_2^* is such that both estimators $E_f L^* Lf$ and $||f||^2$ have the same asymptotic mean square error.

[5] t^* is the square of the asymptotically optimal bandwidth

[6] $*$ t is the asymptotically optimal value of t which minimizes AMISE

when p(x) = a(x) = 1, t_2^* and $_*t_2$ are identical. Thus the following bandwidth selection and estimation procedure is suggested for the diffusion estimator (4.5). There are three different algorithms for that, but in the following we will present only one [8].

Algorithm

1.Given the data X1, . . . ,X_N, run Algorithm 1[7] to obtain the Gaussian kernel density estimator (4.1) evaluated at $_*\hat{t}$ and the optimal bandwidth $\sqrt{*\,\hat{t}}$ for the estimation of $||f''||^2$. This is the pilot estimation step.

2. Let p(x) be the Gaussian kernel density estimator from step 1, and let a(x) = $p^{\propto}(x)$ for some α ∈ [0, 1].

3. Estimate $||Lf||^2$ via the plug-in estimator (4.18) using $\widehat{t_2^*} = *\,\widehat{t_2}$, where $*\,\widehat{t_2}$ is computed in step 1.

4. Substitute the estimate of $||Lf||^2$ into (4.14) to obtain an estimate for t^*.

5. Deliver the diffusion estimator (4.5) evaluated at $\widehat{t^*}$ as the final density estimate.

The bandwidth selection rule used for the diffusion estimator in the Algorithm is a single stage direct plug-in bandwidth selector, where the bandwidth t_2^* for the estimation of the functional $||Lf||^2$ is approximated by $*\,\widehat{t_2}$, instead of being derived from a normal reference rule.

[7] To know how Algorithm 1 is constructed please refer to [8]

4.6 Conclusion

A kernel density estimator based on a linear diffusion process was presented. The key idea is to consider the most general linear diffusion with a stationary density equal to a pilot density estimate when constructing an adaptive kernel. Also, “the estimator is consistent at boundaries”[8]. The contribution was to implement this kernel estimator as a part of the code used in the simulations conducted in the next chapter. The new scheme provides better smoothness of the recovered signal, with faster run time and less resource consumption compared to the "ksdensity"-estimator implemented in Matlab[8].

[8] Type help ksdensity in Matlab to know how the ksdensity estimator works

Chapter 5: Simulations

In this chapter we will present the simulations that have been done to compare the BAMP scheme w.r.t other recovery schemes. We will briefly discuss each recovery scheme and mention the parameters used, but before we proceed we will give a brief introduction about the modulation schemes used.

In our simulations we compared the performance of different recovery schemes using 3 modulation schemes to be defined shortly in what follows.

5.1 Modulation Schemes

5.1.1 Normal Distribution

The normal distribution or, as it is often called, the Gauss distribution is the most important distribution in statistics. The distribution is given by

$$f(x;\ \mu, \sigma^2\ ) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

where μ is a location parameter, equal to the mean, and σ the standard deviation. For μ = 0 and σ = 1 we refer to this distribution as the standard normal distribution. In many connections it is sufficient to use this simpler form since μ and σ simply may be regarded as a shift and scale parameter, respectively. In figure 5.1 we show the standard normal distribution.

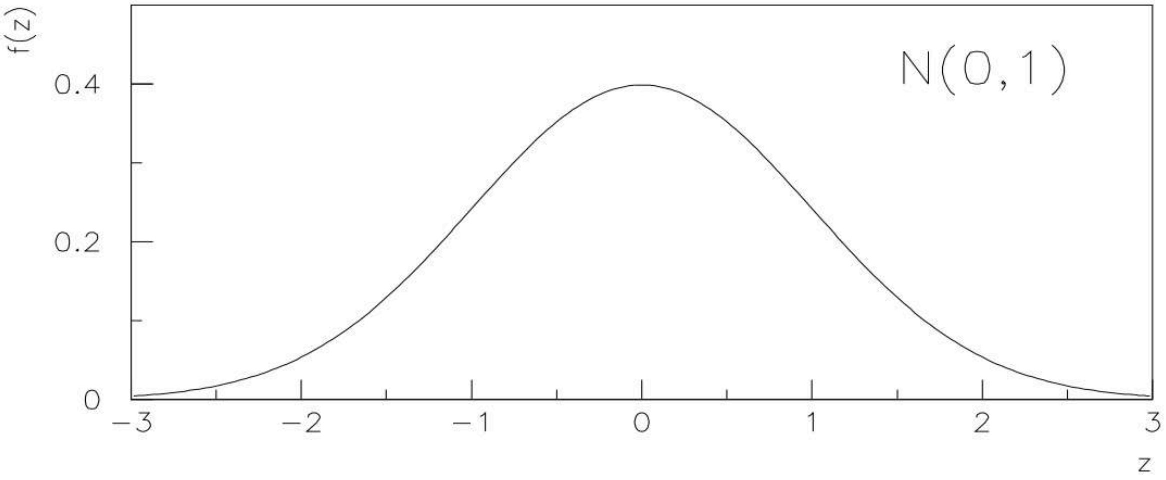

Fig 5.1: Standard normal distribution

5.1.2 Phase Modulation

Given a sinusoidal carrier with frequency: fc, we may express a digitally-modulated passband signal, S(t), as:

$$S(t) = A(t)\cos(2\pi fc\ t + \theta(t)),$$

where A(t) is a time-varying amplitude modulation and θ(t) is a time-varying phase modulation. For digital phase modulation, we only modulate the phase of the carrier, θ(t), leaving the amplitude, A(t), constant.

5.1.2.1 Binary Phase Shift keying (BPSK)

BPSK is the simplest form of digital phase modulation. For BPSK, each symbol consists of a single bit. Accordingly, we must choose two distinct values of θ(t), one to represent 0, and one to represent 1

Since there are 2π radians per cycle of carrier, and since our symbols can only take on two distinct values, we can choose θ(t) as either 0 and π or as $-\frac{\pi}{2}$ and $\frac{\pi}{2}$ referring to θ1(t) and θ0(t). Fig 5.2 shows the BPSK modulation for 0 and π choice [13].

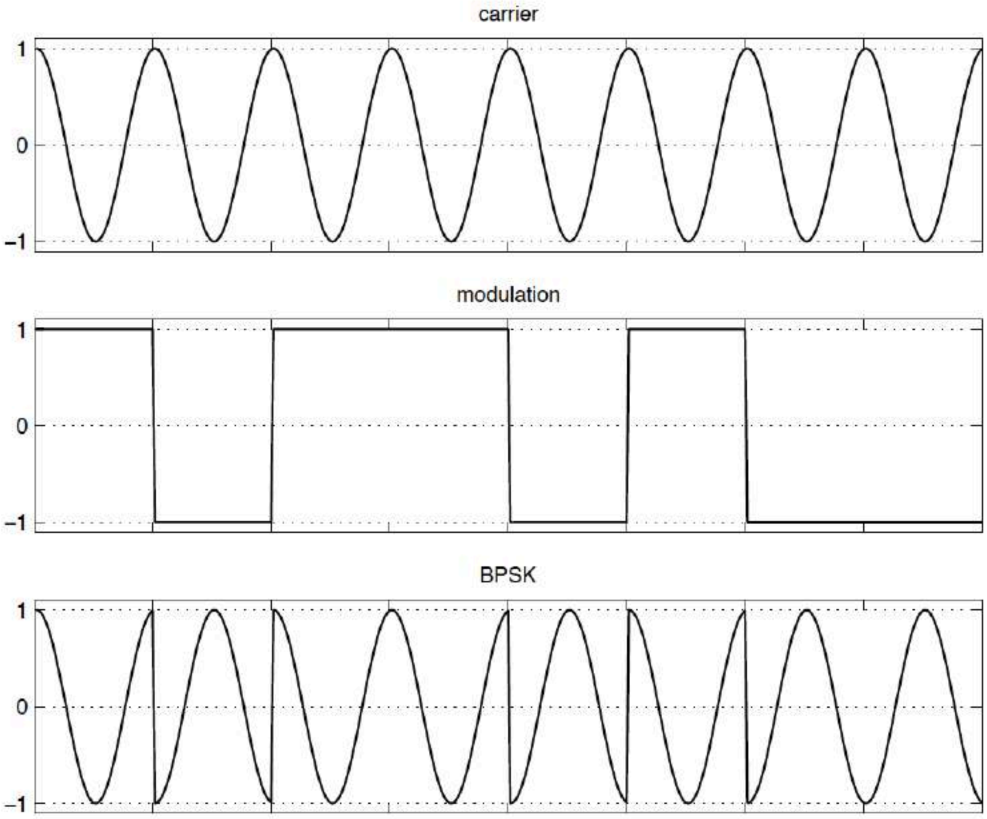

Fig 5.2:BPSK Modulation (case 0 and π)

5.1.3 Amplitude Modulation

5.1.3.1 Amplitude Shift Keying (ASK)

ASK is an amplitude modulation that represents digital data as variations in the amplitude of a carrier wave. In an ASK system, the binary symbol 1 is represented by transmitting a fixed-amplitude carrier wave and fixed frequency for a bit duration of T seconds. If the signal value is 1 then the carrier signal will be transmitted; otherwise, a signal value of 0 will be transmitted.

Any digital modulation scheme uses a finite number of distinct signals to represent digital data. ASK uses a finite number of amplitudes, each assigned a unique pattern of binary digits. Usually, each amplitude encodes an equal number of bits [13].

The simplest and most common form of ASK operates as a switch, using the presence of a carrier wave to indicate a binary one and its absence to indicate a binary zero. This type of modulation is called on-off keying (OOK).

5.2 Sparse Recovery Schemes

We have previously discussed both AMP and BAMP with prior and without prior. In this section we are going to briefly declare the rest of the schemes that are present in our simulations.

5.2.1 Expectation Maximization for Gaussian mixtures

An elegant and powerful method for finding maximum likelihood solutions for models with latent variables is called the expectation-maximization algorithm, or EM algorithm. “It is used to obtain the variational inference framework”[11]. We used this method to estimate the parameters of the BAMP recovery scheme (BAMP/EM).

Given a Gaussian mixture model, the goal is to maximize the likelihood function with respect to the parameters [11] (comprising the means and covariances of the components and the mixing coefficients).

1. Initialize the means μ_k, covariances Σ_k and mixing coefficients π_k, and evaluate the initial value of the log likelihood.

2. E step. Evaluate the responsibilities using the current parameter values

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n|\mu_j, \Sigma_j)}$$

3. M step. Re-estimate the parameters using the current responsibilities

$$\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) x_n$$

$$\Sigma_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(x_n - \mu_k^{new})(x_n - \mu_k^{new})^T$$

$$\pi_k^{new} = \frac{N_k}{N}$$

where

$$N_k = \sum_{n=}^{N} \gamma(z_{nk})$$

4. Evaluate the log likelihood

$$\ln(X|\mu, \Sigma, \pi) = \sum_{n=}^{N} \ln\left\{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)\right\}$$

and check for convergence of either the parameters or the log likelihood. In case the convergence criterion is not satisfied return to step 2.

5.2.2 Orthogonal Matching Pursuit

OMP is greedy algorithm that can reliably recover a signal with m nonzero entries in dimension n given O(m ln n) random linear measurements of that signal. The results for OMP are comparable with results for another algorithm called Basis Pursuit (BP). The OMP algorithm is faster and easier to implement, which makes it an attractive alternative to BP for signal recovery problems. We simulated this scheme with other schemes, as it is often used in practice. Note that it doesn't work well for high dimensions and large number of non-zero components as a pseudo-inverse has to be computed.

5.3 Simulations

In the following section we present the simulations that were done assuming different scenarios, where BAMP/HI (a scheme based on using the histogram model after estimating the pdf of the unknown signal x to be recovered using the kernel density estimator presented in the previous chapter), AMP, and OMP work without knowledge of the prior, and BAMP/EM estimates the prior using EM algorithm. The priors used are for the BAMP (curves in the figure below). We introduce and define some of the parameters to be considered fixed for all simulations while altering the rest.

$$N = 400, \text{ (signal dimension)}$$

Where this chosen value is considered to be relatively big, and constrained by OMP scheme.

$$N_{\mathrm{sims}} = 100,$$

where N_{sims} is the number of blocks simulated. We use large value to reduce the size of the significance interval.

$$S=60$$

Sparsity, as we have in our simulations 60 non-zeros for representing the recovered data. Now we present each simulation with the respective parameters and prior chosen

5.3.1 Simulation 1

First simulation assumes a sparse Gaussian as a prior, and the noise levels for the CS observation vector y as stdwv = [0.075, 0.125, 0.2];

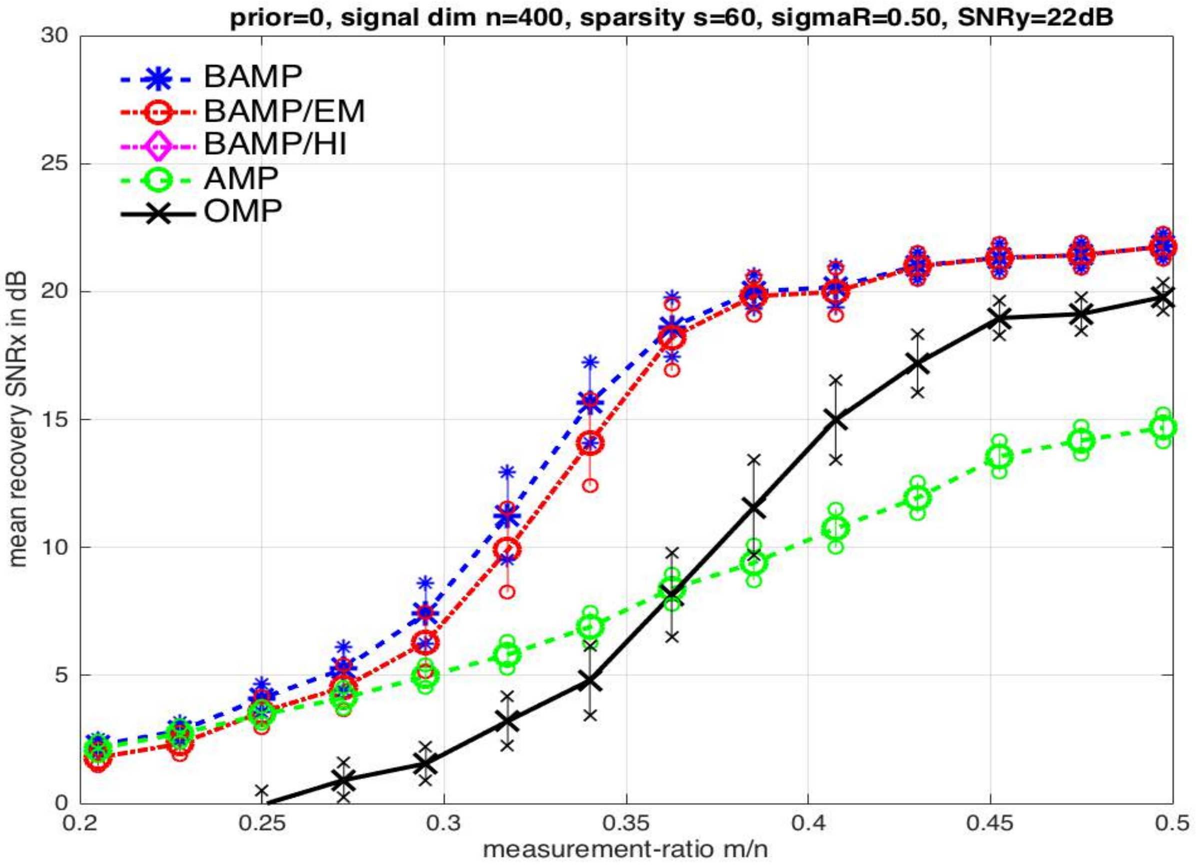

Fig 5.3: Comparison of BAMP without knowledge of the prior with other schemes assuming sparse Gaussian as a known prior

We notice that in general BAMP/HI has the exact same performance as BAMP, which is better than all other schemes.

5.3.2 Simulation 2

Second simulation assumes sparse BPSK as a prior, and the noise levels for CS observation vector y as stdwv = [0.075, 0.125, 0.2];

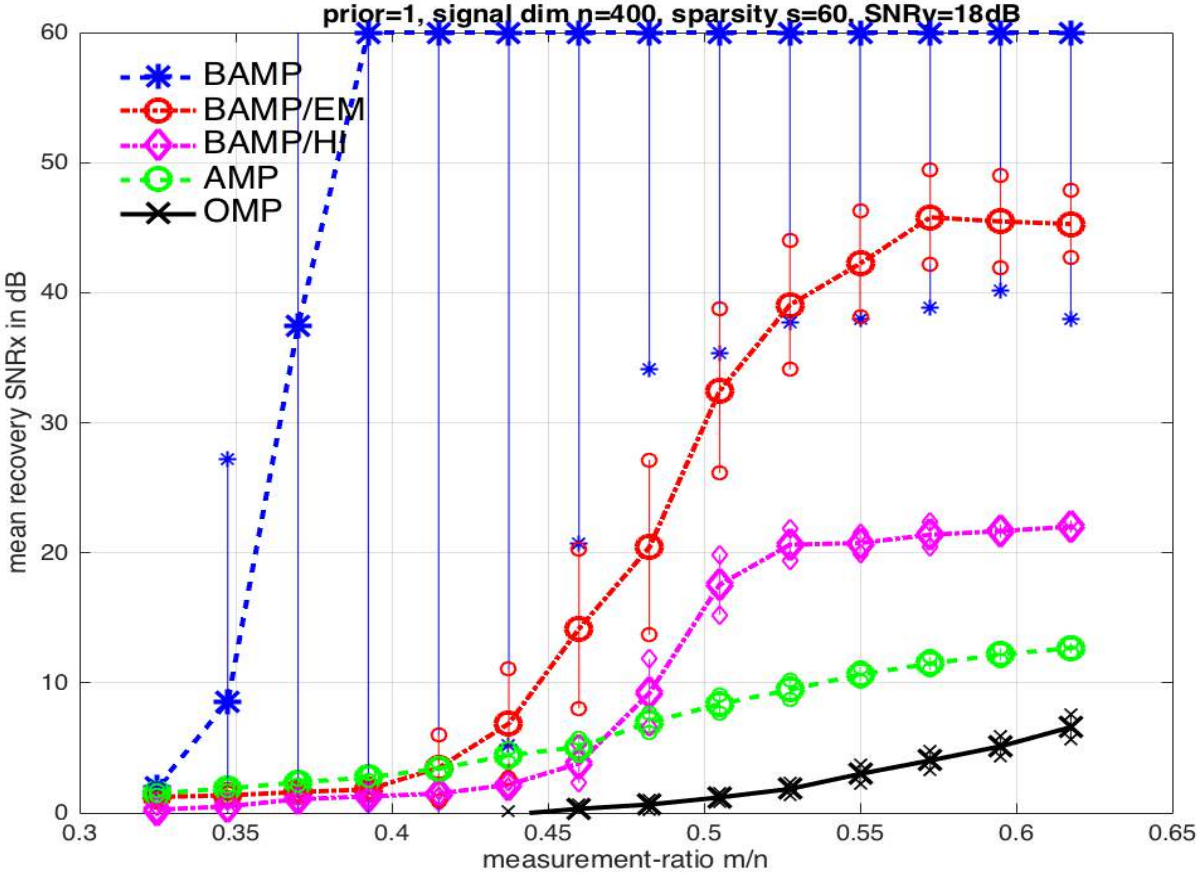

Fig 5.4: Comparison of BAMP without knowledge of the prior with other schemes assuming sparse BPSK as a known prior

BAMP/HI performs better than AMP and OMP in general and gets closer to BAMP/EM for high measurement ratio. Also, note that BAMP can get inf-SNR because it decides for sufficient SNRy correctly, while other schemes don't know that only two BPSK signal levels exist, so an exact match can be found by recovery.

5.3.3 Simulation 3

Third simulation assumes sparse 4-ASK as a prior, and the noise levels for CS observation vector y as stdwv = [0.05, 0.1, 0.15]

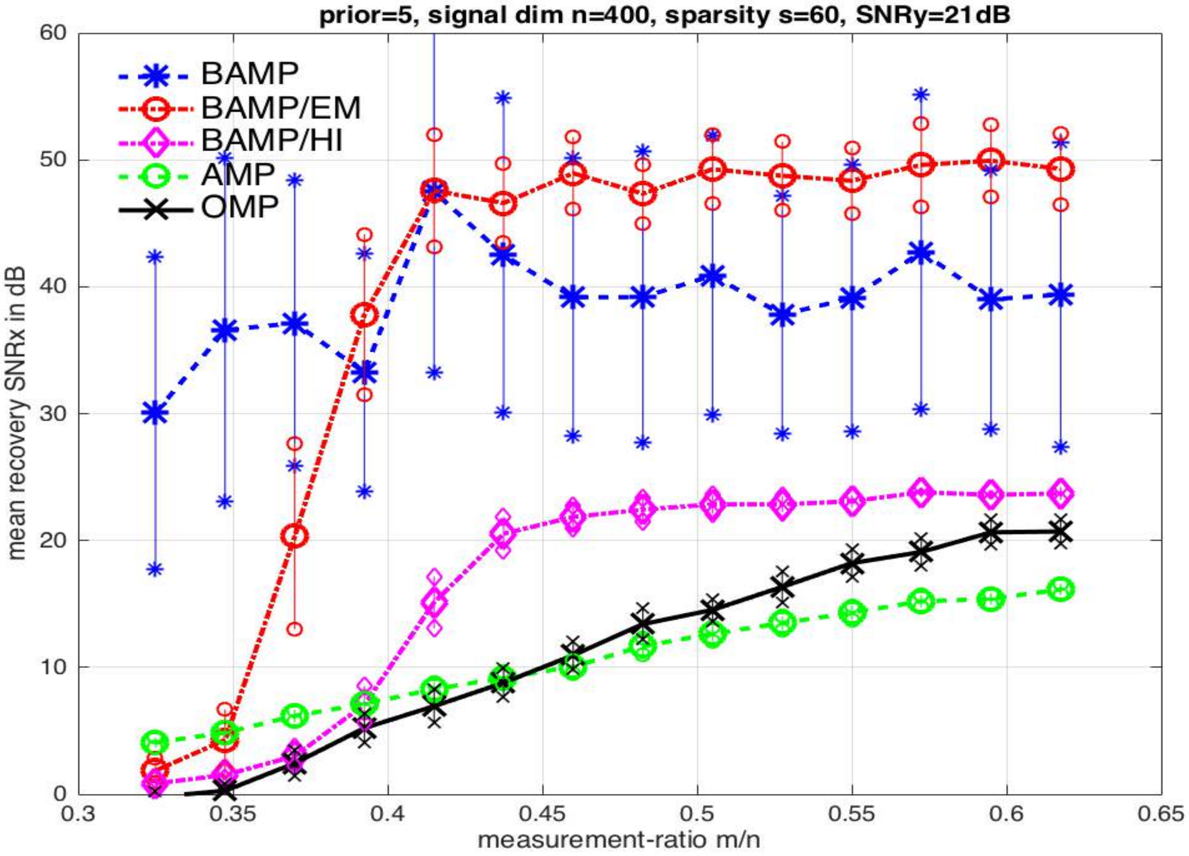

Fig 5.5: Comparison of BAMP without knowledge of the prior with other schemes assuming sparse BPSK with slow fading as a known prior

Also in this case we find that BAMP/HI get closer to BAMP performance and performs better than AMP and OMP. We would like to point out that the number of iterations for the BAMP/HI is higher than other schemes used to be able to learn the prior.

Chapter 6: Conclusion:

BAMP without knowledge of the prior performs better than OMP and AMP knowing that also the OMP and AMP schemes have no knowledge of the prior as well, but when compared to the BAMP schemes which know the prior, it is evident that they will have a better performance since they know the prior, but in practical situations we don't know the prior. On the on hand BAMP/HI has a reduced complexity compared to other schemes that assume knowledge of the prior. Also there is some degradation in the performance, but it is safe to say that it is in the acceptable range of SNR and relatively close to the optimum performance, while AMP and OMP are considered way below the acceptable SNR range. A trade off between complexity and performance is quite dependent on where our interest lies, but the results are quite promising with far less complexity than expected. The contribution in this thesis was substituting the offered kernel density estimator which is built-in in MATLAB (ksdensity) by a kernel density estimator (kde) programmed in a way that requires at most half the time to run the simulations defining similar parameters for both functions, with the ability to estimate the optimum bandwidth for improved smoothness, and even less resource consumption while running the simulations (CPU, memory).

Bibliography

[1] M. A. Davenport, M. F. Duarte, Y. C. Eldar, and Gitta Kutyniok, Introduction to Compressed Sensing, http://statweb.stanford.edu/~markad/publications/ddek-chapter1-2011.pdf

[2] D. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," Proceedings of the National Academy of Sciences, vol. 106, no. 45, pp. 18 914–18 919, 2009. [Online]. Available: http://www.pnas.org/content/106/45/18914. Abstract

[3] N. Goertz, Iterative Recovery in Compressed Sensing Introductory Lectures – SS2016 Seminar "Signal

Processing for Big Data"

[4] N. Goertz, Iterative Schemes and Approximate Message Passing for recovery in Compressed Sensing Lecture "Compressed Sensing", May 2015

[5] M. Mayer, and N. Goertz, Bayesian Optimal Approximate Message Passing to Recover Structured Sparse Signals, IEEE Transactions on Signal Processing, pp 1-4, 2015

[6] M. Raphan and E. Simoncelli, "Empirical Bayes least squares estimation without an explicit prior," Computer Science Technical Report, Courant Inst. of Mathematical Sciences, New York University, Technical Report TR2007-900, May 2007. [Online]. Available: http://www.cns.nyu.edu/pub/eero/raphan07b.pdf

[7] J. P. Vila, Empirical-Bayes Approaches to Recovery of Structured Sparse Signals via Approximate Message Passing, Dissertation, 2015, http://www2.ece.ohio-state.edu/~schniter/pdf/vila_diss.pdf

[8] Z. I. Botev1, J. F. Grotowski and D. P. Kroese, KERNEL DENSITY ESTIMATION VIA DIFFUSION, 2010, Vol. 38, No. 5, 2916–2957, https://arxiv.org/pdf/1011.2602.pdf

[9] Springer, 2006.M. P. Wand, and M. C. Jones (1995). Kernel Smoothing. Monographs on Statistics and Applied Probability Chapman & Hall, 1995, http://compdiag.molgen.mpg.de/docs/talk_05_01_04_stefanie.pdf

[10] N. Goertz, "A Practical Guide to Bayesian Approximate Message Passing in Compressed Sensing", Internal Report, TU Wien, December 2016.

[11] C. Bishop, Pattern Recognition and Machine Learning. Springer, 2006.

[12] Y. Ma, J. Zhu, and D. Baron, Approximate Message Passing Algorithm with Universal Denoising and Gaussian Mixture Learning, in IEEE Transaction on Signal Processing, 2015

[13] J. E. Gilly Digital Phase Modulation, A Review of Basic Concepts, 2003

[14] J. G. Proaksis, and M. Salehi, Digital Communications, fifth edition, "Convergence behavior of iteratively decoded parallel concatenated codes," IEEE Transactions on Communications, vol. 49, pp.1727–1737, 2008

[15] Y. Zai, analysis, algorithms and applications of compressed sensing, 2014, https://repository.ntu.edu.sg/handle/10356/59537

[16] Y. Ma, D. Baron, and D. Needell, Two-Part Reconstruction With Noisy Sudocodes, pp 1-3, 2014

[17] A. Montanari, "Graphical models concepts in compressed sensing," http://arxiv.org/abs/1011.4328, 2011.

[18] A. Maleki, Approximate Message Passing Algorithms for Compressed Sensing, Ph.D. thesis, Stanford University, 2011.

Appendix A: GAUSSIAN KERNEL DENSITY ESTIMATOR PROPERTIES

In this appendix, we present the technical details for the proofs of the properties of the diffusion estimator(chapter 4). In addition, we include a description of our plug-in rule in two dimensions. We use ‖ . ‖ to denote the Euclidean norm on R.

Assume that f'' is a continuous square-integrable function. The integrated squared bias and integrated variance of the Gaussian kernel density estimator (4.1) have asymptotic behavior

$$\left\|E_F\left[\hat{f}(.\,;t)\right] - f\right\|^2 = \frac{1}{4}t^2\left\|f''\right\|^2 + O(t^2), \quad N \to \infty. \tag{4.19}$$

And

$$\int \mathrm{Var}_F\left[\hat{f}(x;t)\right] dx = \frac{1}{2N\sqrt{\pi t}} + O\left((\sqrt{N}t)^{-1}\right), \quad N \to \infty. \tag{4.20}$$

respectively. The first-order asymptotic approximation of MISE, denoted AMISE, is thus given by

$$\mathrm{AMISE}\{\hat{f}\}(t) = \frac{1}{4}t^2\left\|f''\right\|^2 + \frac{1}{2N\sqrt{\pi t}} \tag{4.21}$$

The asymptotically optimal value of t is the minimizer of the AMISE

$$* t = \left(\frac{1}{2N\sqrt{\pi t}\left\|f''\right\|^2}\right)^{2/5} \tag{4.22}$$

giving the minimum value

$$\mathrm{AMISE}\{\hat{f}\}(* t) = N^{-4/5}\,\frac{5\left\|f''\right\|^{2/5}}{4^{7/5}\pi^{2/5}} \tag{4.23}$$

For a simple proof , see [12]