Analytica Chimica Acta 1279 (2023) 341762

Contents lists available at ScienceDirect

Analytica Chimica Acta

journal homepage: www.elsevier.com/locate/aca

Robust statistical methods for high-dimensional data, with applications in tribology

Pia Pfeiffer *, Peter Filzmoser

Institute of Statistics and Mathematical Methods in Economics, TU Wien, Wiedner Hauptstraße 8–10, 1040, Vienna, Austria

HIGHLIGHTS

- Robustness in context of chemometrics: Overview of selected robust statistical methods and implementations.
- Application of robust regression to predict engine oil degradation based on FTIR spectra.
- Two-step approach to classification for high-dimensional data.
- Robust regression with image features: sampling strategies for imbalanced data, robust data cleaning.
- Outlier explanation: Strategies for investigating why an observation is outlying.

ARTICLE INFO

Handling Editor: Prof. L. Buydens

Keywords:
Robust regression
Robust classification
High-dimensional data analysis
Chemometrics
FTIR spectra

ABSTRACT

Data sets derived from practical experiments often pose challenges for (robust) statistical methods. In high-dimensional data sets, more variables than observations are recorded and often, there are also data present that do not follow the structure of the data majority. In order to handle such data with outlying observations, a variety of robust regression and classification methods have been developed for low-dimensional data. The high-dimensional case, however, is more challenging, and the variety of robust methods is much more limited. The choice of the method depends on the specific data structure, and numerical problems are more likely to occur. We give an overview of selected robust methods as well as implementations and demonstrate the application with two high-dimensional data sets from tribology. We show that robust statistical methods combined with appropriate pre-processing and sampling strategies yield increased prediction performance and insight into data differing from the majority.

1. Introduction

The advance of digital technologies has transformed the way data are collected and analyzed. In tribology, these developments have motivated the use of data-driven methods for the design and validation of tribological systems. For oil condition monitoring, for example, several authors investigate the application of spectroscopic methods to monitor the lubricant's degradation process over time: FTIR (Fourier-transform infrared) spectra can be used to predict oil attributes [1–3]. Other modeling objectives include the comparison of oil degradation in different laboratory alterations and field settings [4,5]. Another aspect linked to oil condition is lubrication performance, i.e. friction and wear behavior. To investigate lubrication performance, SRV® (Schwing-Reib-Verschleiß) tribometer experiments (a steel ball sliding against a steel disk with the lubricant of interest in between) are carried out, resulting in a collection of several types of data for one oil, including functions of the coefficient of friction and optical data of wear scar areas.

However, in data produced from experiments, there may also be observations present that behave differently from the majority of data points. Those observations are called *outliers* in statistics and the data set is said to be *contaminated*. While for traditional methods one outlying observation can have a huge impact on the resulting model, robust methods aim to identify and downweight unusual data points. This way, observations that do not follow the majority of the data can be uncovered and further investigated.

In addition, high numbers of measured variables make the application of classical statistical methods difficult. A given data set is called *high-dimensional* if the number of variables p exceeds the number of

* Corresponding author.
E-mail address: pia.pfeiffer@tuwien.ac.at (P. Pfeiffer).

https://doi.org/10.1016/j.aca.2023.341762
Received 13 January 2023; Received in revised form 8 August 2023; Accepted 28 August 2023
Available online 5 September 2023

observations n. In this setting, both the classical as well as robust regression and classification estimators are not well-defined and run into numerical problems. These can be handled by dimension reduction, using PCR (Principal Component Regression) or PLS (Partial Least Squares), for example. Other approaches for high-dimensional data are penalized regression or classification estimators such as Ridge [6], LASSO [7] or Elastic Net [8] regression or penalized discriminant analysis [9], as well as sparse logistic regression, available in the R package `glmnet` [10]. These approaches are suitable for high-dimensional data, however, they are not robust in the presence of outliers.

While there are many robust methods available for the low-dimensional case, the portfolio of robust methods for a high-dimensional setting is not that rich. In the following we will mention some of these approaches, and also put emphasis on sparse methods, which are based on the underlying assumption that only a few variables of the high-dimensional data contribute to explaining the response. This work does not aim to give an exhaustive review of available methods, but rather demonstrate the application of selected robust statistical methods for practitioners.

The remainder of the paper is organized as follows: In Section 2, an overview of selected sparse and robust methods as well as available implementations for regression and classification tasks is given. Section 3 illustrates the application of these statistical methods using two data sets from lubricant analysis and tribological experiments: FTIR spectra and image data of wear scar areas resulting from a tribometrical experiment, and Section 4 concludes with recommendations for the application of robust statistical methods in practice.

2. Robust statistical methods

First, selected robust regression and classification estimators are introduced for the low-dimensional setting. Then, approaches to extend robust methods to the high-dimensional case are discussed. For all mentioned methods, the availability of implementations in R software packages is indicated.

2.1. Robust linear regression

Consider n samples $(\boldsymbol{x}_i, y_i)$ with $i = 1, \ldots, n$, where $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})$ contains information about the measurements on p variables. In a regression setting, the values y_i are collected in the vector $\boldsymbol{y}$, which is our response, and the information $(1, \boldsymbol{x}_i)$ is collected as rows of the predictor matrix $\boldsymbol{X}$. The linear regression model is given as $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\epsilon}$, where $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)^{'}$ are the regression coefficients, with the intercept term β_0, and $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^{'}$ are the error terms. Let $\widehat{\boldsymbol{\beta}}$ denote an estimate for the unknown regression coefficients. Then the *residuals* $\boldsymbol{r} \in \mathbb{R}^n$ are given as $\boldsymbol{r}(\widehat{\boldsymbol{\beta}}) = (r_1(\widehat{\boldsymbol{\beta}}), \ldots, r_n(\widehat{\boldsymbol{\beta}}))^{'} = \boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}$.

The well-known Least Squares (LS) estimator is then defined as

$$\widehat{\boldsymbol{\beta}}_{LS} = argmin_{\beta} \sum_{i=1}^{n} r_i(\boldsymbol{\beta})^2. \tag{1}$$

The solution can be easily computed in explicit form: $\widehat{\boldsymbol{\beta}}_{LS} = (\boldsymbol{X}^{'}\boldsymbol{X})^{-1}\boldsymbol{X}^{'}\boldsymbol{y}$. However, this only holds if the matrix $\boldsymbol{X}^{'}\boldsymbol{X}$ is invertible, which would not be the case for high-dimensional settings ($n < p$).

As the LS estimator is based on the squared residuals, the influence of potential outliers is not bounded and therefore even one unusual observation can distort the estimation. One important step towards robustness is to introduce observation weights $\omega_i \in [0, 1]$, for $i \in \{1, \ldots, n\}$. Outlying observations will receive a small weight. Outliers in regression are observations with large residuals, and they could either be outliers in the space of the x-variables (bad leverage points), or they could be in the normal x-range (vertical outliers). Observations with abnormal x-values but small residuals are often called good leverage points, because they could stabilize the regression fit. On the other hand, they could lead to underestimating the residual scale. For a robust estimator, the objective function (1) can be generalized as

$$\widehat{\boldsymbol{\beta}}_{M} = argmin_{\beta} \sum_{i=1}^{n} \rho\left(\frac{r_i(\boldsymbol{\beta})}{\widehat{\sigma}}\right), \tag{2}$$

where ρ denotes an appropriate (bounded) function applied to the residuals and $\widehat{\sigma}$ the residual scale estimate [11]. The resulting estimator is called M-estimator of regression and is computed by solving the system of estimating equations $\sum_{i=1}^{n} \omega_i(r_i(\boldsymbol{\beta}))\boldsymbol{x}_i = 0$ with a weight function $\omega(u) = \rho^{'}(u)/u$ that determines the robustness of the estimator. This can be accomplished by using an iterative reweighted LS algorithm: For a given $\widehat{\boldsymbol{\beta}}_t$ in iteration step t, the residuals and weights can be computed and the estimating equations solved for $\widehat{\boldsymbol{\beta}}_{t+1}$. The starting value $\widehat{\boldsymbol{\beta}}_0$ and the residual scale $\widehat{\sigma}$ need to be estimated robustly. A popular choice is the *M-estimator of scale* or *S-estimator*, given as the solution $\widehat{\sigma}$ of

$$\frac{1}{n}\sum_{i=1}^{n} \rho\left(\frac{r_i}{\widehat{\sigma}}\right) = b \tag{3}$$

where ρ denotes an appropriate (bounded) function and b is a constant. Using the S-estimator (Equation (3)) as the initial estimator for the M-estimator leads to the *MM-estimator*, leading to a compromise between good efficiency and robustness [11]. It is implemented as `lmrob()` in the R package `robustbase` [12].

An intuitive and computationally efficient alternative is given by the Least Trimmed Squares (LTS) estimator [13,14]. It is defined as

$$\widehat{\boldsymbol{\beta}}_{LTS} = argmin_{\beta} \sum_{i=1}^{h} r_{i:n}(\boldsymbol{\beta})^2 \tag{4}$$

with the order statistics of the squared residuals $r_{1:n}(\boldsymbol{\beta})^2 \leq \cdots \leq r_{n:n}(\boldsymbol{\beta})^2$. The efficiency and robustness of the estimator are determined by the parameter h, which is typically chosen as half or 3/4 of the number of observations. As for the above regression estimators, a limitation is that they can only be applied to settings with $n > p$, here, depending on the choice of h, even $n > 2p$. The LTS estimator is also available in the R package `robustbase` as `ltsReg()`.

2.2. Robust regression for high-dimensional data

For the case $p > n$, the PLS estimator is often chosen in chemometrics [15]. Several proposals exist to make this estimator robust against outliers: They are based on robust covariance estimation [16] or replace LS regression by a robust estimator [17–21]. A discussion of robust PLS approaches and respective advantages and disadvantages can be found in Ref. [22]. In the following, we describe the Partial Robust M (PRM) estimator [20]. As for the M-estimator (Equation (2)), observation weights $\omega_i \in [0, 1]$, for $i \in \{1, \ldots, n\}$ are introduced to downweight outlying observations. The weights are collected in the diagonal of the diagonal matrix $\boldsymbol{\Omega} = \mathrm{Diag}(\omega_1, \ldots, \omega_n)$, and the weighted data information is obtained as $\widetilde{\boldsymbol{X}} = \boldsymbol{\Omega X}$ and $\widetilde{\boldsymbol{y}} = \boldsymbol{\Omega y}$. In PLS regression we construct latent components (or *scores*), which are linear combinations of the original variables with *weighting vectors*. The weighting vectors $\boldsymbol{a}_h$ for $h \in \{1, \ldots, h_{max}\}$ are obtained by the maximization problem

$$\boldsymbol{a}_h = \underset{\boldsymbol{a}}{\mathrm{argmax}} \mathrm{cov}^2(\boldsymbol{y}, \boldsymbol{Xa}), \tag{5}$$

for $h \in \{1, \ldots, h_{max}\}$ under the constraints that

$$\|\boldsymbol{a}_h\| = 1 \quad \text{and} \quad \boldsymbol{a}_h^{'}\boldsymbol{X}^{'}\boldsymbol{X}\boldsymbol{a}_i = 0 \text{ for } 1 \leq i < h. \tag{6}$$

Here, h_{max} is the maximum number of components we want to retrieve, and it is assumed that the response, as well as the predictor variables, are mean-centered. In PRM regression, the centering is done robustly, e.g.

by the column-wise median. For estimating the covariance in Equation (5), the sample covariance matrix with the weighted observations has been proposed, and thus we maximize

$$\mathrm{cov}^2\left(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{X}}\boldsymbol{a}\right) = \frac{1}{(n-1)^2}\boldsymbol{a}'\tilde{\boldsymbol{X}}'\tilde{\boldsymbol{y}}\tilde{\boldsymbol{y}}'\tilde{\boldsymbol{X}}\boldsymbol{a}, \tag{7}$$

and the constraints (6) are also based on weighted predictors. The resulting weighting vectors are collected as columns in the matrix $\boldsymbol{A}$, and thus the matrix of scores is $\tilde{\boldsymbol{T}} = \tilde{\boldsymbol{X}}\boldsymbol{A}$, with rows $\tilde{\boldsymbol{t}}_i$, for $i = 1, \ldots, n$. The crucial point is to obtain the weights. As the name already suggests, the PRM regression estimator makes use of the concept of the robust M-estimator, see Equation (2), by regressing the weighted response on the robustified scores,

$$\hat{\boldsymbol{\gamma}} = \underset{\gamma}{argmin} \sum_{i=1}^{n} \rho\left(\tilde{y}_i - \tilde{\boldsymbol{t}}_i'\boldsymbol{\gamma}\right), \tag{8}$$

where $\tilde{y}_i$ are the elements in $\tilde{\boldsymbol{y}}$. This yields robust residuals $\tilde{r}_i = \tilde{y}_i - \tilde{\boldsymbol{t}}_i'\hat{\boldsymbol{\gamma}}$, and by employing a robust scale estimator, such as the MAD, a robustly estimated residual scale $\hat{\sigma}$ can be obtained. The weights are defined by

$$\omega_i^2 = \omega_R\left(\frac{\tilde{r}_i}{\hat{\sigma}}\right)\omega_T\left(\frac{\|\tilde{\boldsymbol{t}}_i - \mathrm{med}_j(\tilde{\boldsymbol{t}}_j)\|}{\mathrm{med}_i\|\tilde{\boldsymbol{t}}_i - \mathrm{med}_j(\tilde{\boldsymbol{t}}_j)\|}\right), \tag{9}$$

where $\tilde{\boldsymbol{t}}_j$ is the jth column of $\tilde{\boldsymbol{T}}$, for $j = 1, \ldots, h_{max}$. The weight function $\omega_R(u)$ takes care about downweighting large (scaled) residuals, whereas the weight function $\omega_T(u)$ downweights leverage points. The specific choice of appropriate weight functions, as well as initial weights to start the iterative algorithm, are discussed in Ref. [20]. More recently, the effects of different weight functions have been studied in Ref. [23], and better guidance has been offered to select the most appropriate one.

The PRM method has been extended to a sparse PRM regression procedure in Ref. [24] which, similar to LASSO regression, yields zeros in the regression coefficient vector, and thus, in fact, performs variable selection. In the R package `sprm` [25], both PRM and SPRM regression are available via the functions `prms()` and `sprms()`. In Python, the package `direpack` [26] provides robust dimensionality reduction techniques for high-dimensional data.

Combining the LTS estimator with L1 regularization yields the *sparse LTS* estimator [27], a robust version of the LASSO. It is given by

$$\hat{\boldsymbol{\beta}}_{sparseLTS} = argmin_{\beta} \sum_{i=1}^{h} r_{i:n}(\boldsymbol{\beta})^2 + n{\cdot}\lambda P(\boldsymbol{\beta}), \tag{10}$$

where $P(\boldsymbol{\beta}) = \sum_{j=1}^{p}|\beta_j|$.

Least Angle Regression (LARS) was proposed by Ref. [28] and is closely related to the LASSO [7]. LARS provides an ordered sequence in which the variables enter the regression model. While this sequence is the same as for the LASSO, it is derived in a computationally more efficient way from the correlation matrix of the data. Based on this property, the authors of [29] propose a robustification of LARS by replacing mean, variance and correlation with robust location, scatter and correlation estimators.

Both methods are available in the R package `robustHD` [30] as `sparseLTS()` and `rlars()`.

2.3. Robust classification

It is assumed that a training set of multivariate data observations is available, together with information about their group membership. The task is to train a classifier which reliably assigns test set observations to the groups. For linear discriminant analysis consider g multivariate normally distributed populations π_i, $i = 1, \ldots, g$ with means $\boldsymbol{\mu}_i$ and the same covariance $\boldsymbol{\Sigma}$. Let p_i denote the prior probabilities that an observation belongs to group i. Then the discriminant values for an observation $\boldsymbol{x}$ are given by

$$d_i(\boldsymbol{x}) = \boldsymbol{\mu}_i'\boldsymbol{\Sigma}^{-1}\boldsymbol{x} - \frac{1}{2}\boldsymbol{\mu}_i'\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_i + \ln p_i, \tag{11}$$

for $i = 1, \ldots, g$ [31]. An observation $\boldsymbol{x}$ is assigned to group k, if

$$d_k(\boldsymbol{x}) = \max_i d_i(\boldsymbol{x}). \tag{12}$$

The discriminant values (11) depend on the group means and the joint covariance matrix. In the classical case, the arithmetic means of the data groups and a pooled sample covariance can be used as estimators [31]. In order to achieve a robust classifier in presence of outliers, these estimators can be substituted with robust location and scatter estimates. In Ref. [32], for example, the S-estimator is proposed, while in Ref. [33] the FastMCD estimator is used. The authors of [34] provide a comparative study between different robust covariance estimators. An implementation is available as Linda() in the R package `rrcov` [35].

Robust classification can also be performed by applying robust regression estimators for a logistic regression model, where the posterior class probabilities with the group variable G are modeled by linear functions,

$$\log\frac{P(G=k|\boldsymbol{x})}{P(G=g|\boldsymbol{x})} = \beta_{k0} + \boldsymbol{\beta}_k'\boldsymbol{x}, \quad \text{for} \quad i = 1, \ldots, g-1, \tag{13}$$

with the constraint that the probabilities remain in the interval [0, 1] and that they sum up to 1. The model parameters are commonly estimated using the maximum likelihood (ML) method. In the classical case, this corresponds to an iteratively reweighted LS algorithm. In the R package `robustbase` [12], several different algorithms for a robust estimator are implemented in the function `glmrob()`.

2.4. Robust classification for high-dimensional data

As discussed in Section 2.2, several proposals for robust regression in high dimensions have been developed. For classification purposes, however, fewer methods are available. One approach to robust discriminant analysis is by directly plugging in a regularized version of a robust covariance estimator to compute the discriminant values in (11). This can be done for example by applying the Minimum Regularized Covariance Determinant (MRCD) estimator from Ref. [36] to the group-wise robustly centered observations. Another approach is based on applying robust regression estimators in logistic or multinomial regression. The authors of [37] combine a trimmed estimator with the Elastic Net penalty to achieve a robust estimator suitable for high-dimensional data. An implementation is available in the R package `enetLTS` [38]. Another strategy to perform robust classification for high-dimensional data is to first reduce the dimensionality before applying a robust classification method. If the resulting classifier should be adjusted to a response variable, this can be done by constructing latent variables based on PCR or PLS, or selecting variables based on a robust and sparse regression method like sparseLTS. An example for this two-step approach will be given in Section 3.1.2.

The robust methods discussed in the above sections downweight potentially outlying rows $\boldsymbol{x_i}$ of a given data set $\boldsymbol{X}$. Especially in the high-dimensional case, however, it might be desirable to consider the concept of *cellwise* robustness: In contrast to rowwise robustness, outlying cells x_{ij}, not rows $\boldsymbol{x_i}$, are flagged. Rather recent proposals for cellwise robust estimators have been made by Refs. [39,40], though unfortunately, their algorithms are not available in R packages yet.

3. Examples

3.1. Sparse robust regression and classification with FTIR spectra

Some of the methods above are illustrated on a data set consisting of

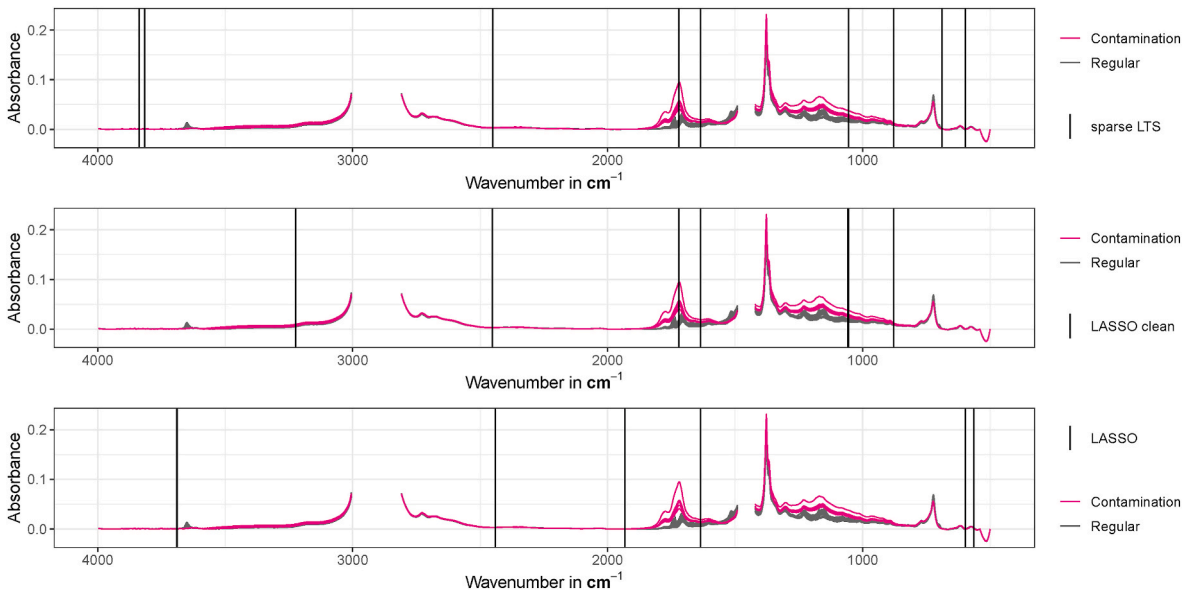


Fig. 1. FTIR spectra of the training set of Group A, for robust and non-robust sparse regression on the contaminated data. The vertical lines indicate the wavenumbers selected by the models. For *sparse LTS*, the selected wavenumbers are 3836.62, 3815.40, 2449.73, 1720.60, 1635.72, 1055.12, 877.66, 688.62, and 597.96 cm^{-1}. For *LASSO* on clean data, the selected wavenumbers are 3223.22, 2451.66, 1720.60, 1635.72, 1057.05, 1055.12, and 877.66 cm^{-1}, and for *LASSO* on contaminated data, the selected wavenumbers are 3690.02, 3688.09, 2440.08, 1932.78, 1635.72, 597.96, and 563.24 cm^{-1}.

FTIR spectra of ten automotive engine oils. The underlying engine oils are commercially available SAE 5W-30 and SAE 0W-20 engine oils. FTIR spectra and other conventional analyses indicate the application of additives commonly used in automotive engine oils, like ZDDP (zinc dialkyldithiophosphates), antioxidants, detergents with a base reserve, and dispersants.The fresh oils were subjected to an artificial small-scale alteration as described in Ref. [41], once with a temperature of 180 °C, and once with 160 °C, denoted by *Group A* and *Group B*, respectively. For both groups, samples were taken regularly during the total duration of 96 h, yielding a data set of in total 50 samples per group.

For all of these samples, FTIR spectra were recorded, each consisting of the absorbance at 1814 wavenumbers. The resulting data set contains $p = 1814$ explanatory variables and $n = 100$ observations and includes two types of response: a grouping variable denoting the membership to Group A or B, respectively, and a numeric response referring to the alteration duration in hours. Hence, the statistical tasks at hand are classification according to group membership and regression on the alteration duration for each group separately. In this application, the interpretability of those models is also of interest: A sparse model with only few non-zero coefficients corresponding to specific wavenumbers can help to understand the underlying chemical processes distinguishing the groups or contributing to oil degradation.

As there are only 50 samples per group (the same ten oils in each temperature group, with varying levels of alteration duration in the groups), we have a high-dimensional setting with low sample sizes. In order to make the tasks even more challenging, we added 6 samples from a *large-scale* artificial alteration series according to Ref. [42] to Group A (same temperature of 180 °C). We will refer to these data as "contaminated" samples. This will call for robust methods, and their performance will be compared to non-robust counterparts.

The wavenumbers between 3030 and 2770 cm^{-1} and 1480-1430 cm^{-1} are areas of high or total absorption, i.e. are not reliable measurements. This is caused by vibrations of hydrocarbons that are always present in engine oils. As a result, these regions not only exhibit total absorption but also do not provide any useful information and are generally disregarded during evaluation. These sections are sometimes removed manually by domain experts but can also be identified as uninformative variable ranges by statistical methods, as proposed by Ref. [5]. After the filtering process applied in Ref. [5], the spectra consist

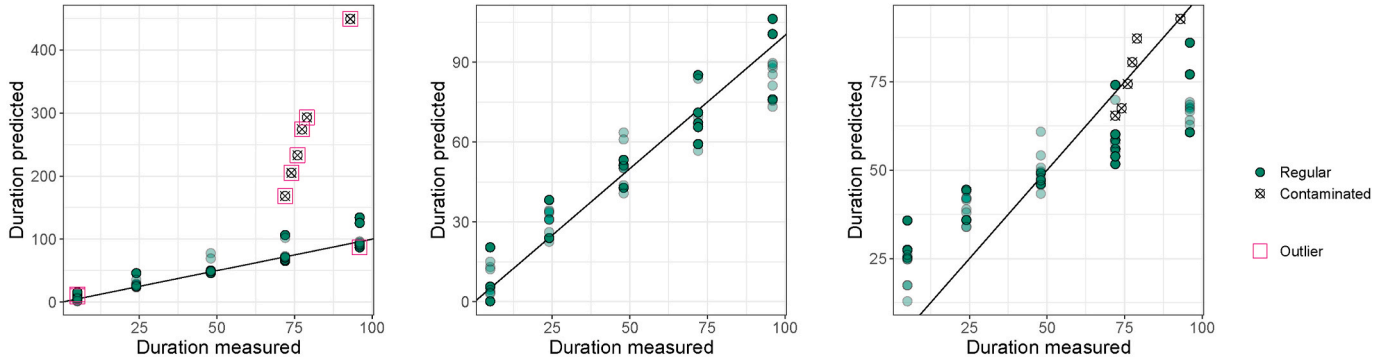


Fig. 2. Measured versus predicted response for training (half transparent) and test set, where the prediction is based on the selected variables from Fig. 1. The robust method (Fig. 2a) fully downweights the contaminated samples, while the classical method (Fig. 2c) is severely influenced by those samples.

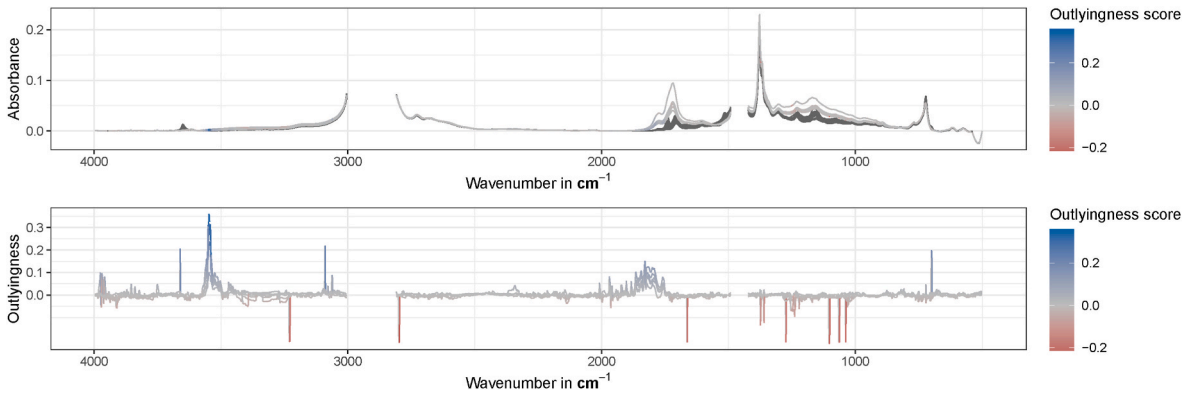


Fig. 3. FTIR spectra of the training set of Group A, as well as the outlyingness scores for each variable resulting from the SPADIMO algorithm [44]. The lines have been colored according to the outlyingness direction and strength.

of 1668 out of 1814 wavenumbers.

3.1.1. *Sparse regression*

Due to the nature of FTIR data, neighboring variables are highly correlated and we can expect that only a few wavenumbers are sufficient for a reasonable prediction accuracy. We use the LASSO estimator [7] to perform sparse regression with high-dimensional data, separately for the Group A and the Group B measurements.

Since Group A has been contaminated by 6 observations, we also fit a LASSO model to the uncontaminated Group A measurements for comparison. As a robust counterpart, the sparse LTS estimator, see Equation (10), is used separately for the contaminated Group A and for Group B. All methods are applied on randomly selected training sets: When fitting a model to the uncontaminated Group A and to Group B, about 2/3 of the samples were selected; when fitting the contaminated Group A, all 6 *large-scale* samples were added to the training set. The test sets consist of all remaining samples from both data sets.

Fig. 1 shows the (selected) FTIR spectra of the training data, here for Group A, together with vertical lines indicating the selected variables from the three approaches. All methods yield only very few variables. The variable selection by the robust method should not be influenced by the contamination. For the LASSO, however, a rather big difference on the clean and contaminated training data can be observed: not only the number, but also the position of selected variables is different.

Fig. 2 shows the measured (horizontal axes) versus the predicted (vertical axes) response of Group A, for the three models, with the selected variables shown in Fig. 1. The colors correspond to training (half transparent) or test dataset and the symbols show whether an observation is regular, contaminated, or identified as an outlier by the robust procedure. The solid line refers to the equality $y = \hat{y}$. The plot for the robust method (Fig. 2a) reveals that the model does not follow the contaminated samples. This can also be verified by inspecting the observations flagged as outliers (encoded as squares): the contaminated samples in the training set, and also some additional observations, were fully downweighted when fitting the model. In addition to the contaminated samples, two additional observations are flagged as outlying. These two outliers consist of atypical x-information, but their prediction is still in a normal range (good leverage points). Since the procedure also yields a robustly estimated standard deviation, this outlying information can also be computed for the test set data: here, outliers are defined as observations with standardized absolute residuals larger than 2.

Fig. 2b and c reveal the effect of contamination on the non-robust LASSO estimator: When contamination is present, the LASSO fit changes significantly, as the model also tries to accommodate the contaminated samples. This implies that the selected variables are also influenced by these samples. In Fig. 2b and c this difference between a fit on clean and contaminated data is illustrated.

Table 1
Misclassification errors based on sparse (robust) logistic regression.

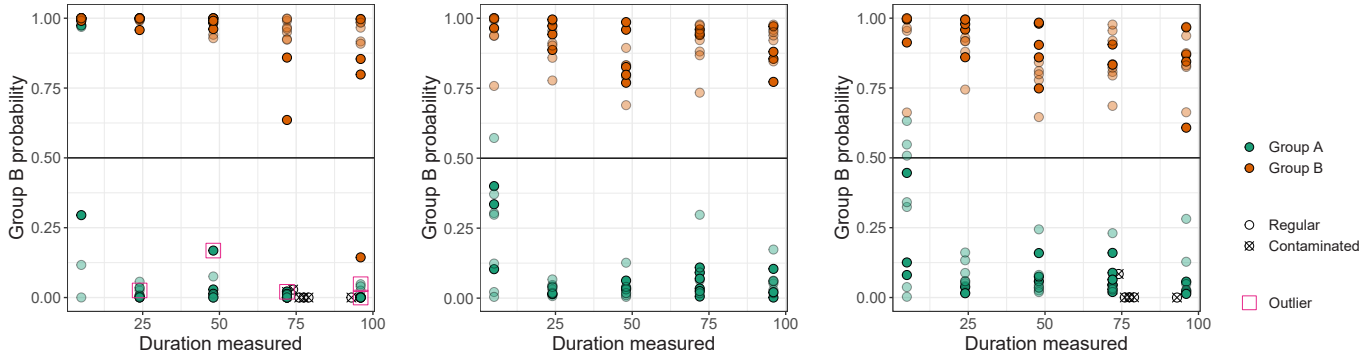
Misclassification error in %	training set	test set
enetLTS for contaminated data	7.46	9.68
Sparse logistic regression for clean data	1.53	0
Sparse logistic regression for contaminated data	4.35	0

The presented methods can identify outliers, and if there is the need to further investigate *why* an observation is outlying, i.e. which variable (s) contribute most to the outlyingness, some recently developed algorithms can be applied: In Ref. [43], the outlyingness is decomposed using Shapley values, and in Ref. [44], the outlyingness is regarded as a regression problem. In the latter approach, it is possible to use the weights, that are output of a robust linear regression fit, as input to the SPADIMO algorithm, which is implanted in the R package `crmReg` [45]. We use the function `spadimo` together with the weights from sparse LTS regression and show the resulting outlyingness scores for each observation that has been identified as outlying in Fig. 3. The lines have been colored according to the outlyingness scores, and upon inspection of the figures the usefulness of robust methods becomes apparent: As several variables, that are selected for the resulting sparse model, also correspond to variables with high outlyingness scores, it is crucial to apply a statistical method that can deal with outliers.

3.1.2. *Sparse classification*

The second statistical task is to predict the group membership of the samples based on the wavenumbers. In our high-dimensional setting, a penalized estimator such as sparse logistic regression, see Ref. [46], with a LASSO penalty on the negative log-likelihood is applied. This yields again variable selection among the wavenumbers. As a robust counterpart, the robust version of the Elastic Net estimator for logistic regression (enetLTS) is used [37]. We use again the same training data as in Section 3.1.1, and evaluate based on the test data; for the non-robust method, the estimator is also applied to the clean data set. The resulting misclassification errors are given in Table 1. For computing the misclassification rates in Table 1 we used the cutoff value 0.5 for the probabilities.

In contrast to the results from the regression, outliers do not seem to have a negative influence on the classical estimators. When inspecting the corresponding plots of group probability over *duration* in Fig. 4, however, it becomes apparent that the misclassification error is not evenly distributed over the different values of duration. In Fig. 4, the cutoff value is displayed as a horizontal line, the colors refer to group membership and the symbols distinguish regular, contaminated and observations identified as outliers by the robust procedure (encoded as a square). The training data are again shown as half transparent points. While the clean data is classified almost perfectly (Fig. 4b), the



(a) direct enetLTS for contaminated data (b) direct sparse logistic regression for clean data (c) direct sparse logistic regression for contaminated data

Fig. 4. Response variable *duration*, used in the regression step, against the posterior probability for Group B, with the cutoff at 0.5. Misclassified observations from the training (half transparent) and test set are shown "on the wrong side" of this cutoff.

Table 2
Misclassification errors based on sparse logistic regression with the pre-selected variables from Section 3.1.1.

Misclassification error in %	training set	test set
sparseLTS for contaminated data	2.86	3.03
LASSO for clean data	5.88	3.03
LASSO for contaminated data	7.46	6.25

prediction for the sparse logistic regression model for contaminated data is worst for observations with duration zero (Fig. 4c). The resulting plot for enetLTS is given in Fig. 4a. While the robust method yields more confident predictions, it fails to detect the contaminated samples correctly. The misclassification error also seems biased and is worse at both minimum and maximum duration. This might be due to the enetLTS algorithm that evaluates all possible subsets of a given size of the explanatory variables. This process can become unstable in the presence of many correlated predictors.

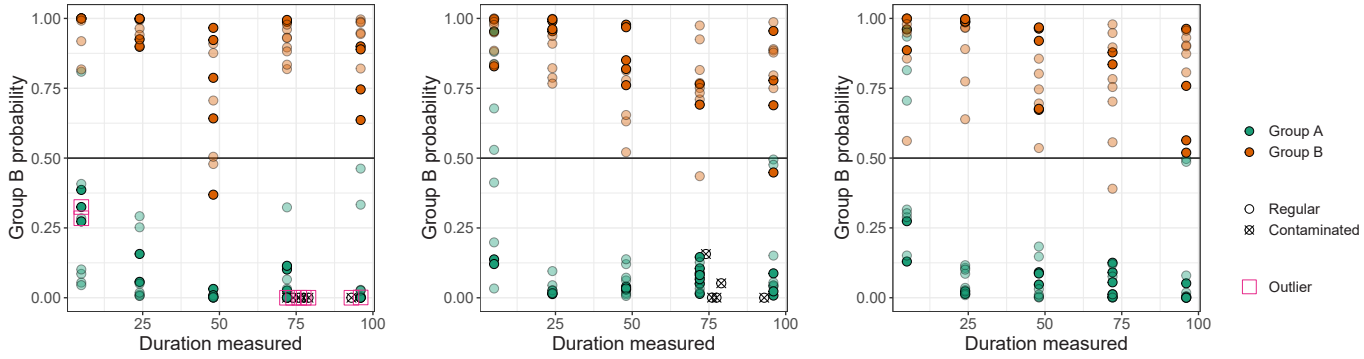
In order to better adjust the classifier to the response *duration*, modeling the respective duration for Group A and Group B can be used as a variable screening step. Using the set of selected variables resulting from this first step, a classification model can now be fitted to discriminate the two groups. Again, the same training and test data as for step 1 are used. Here we did not employ a robust procedure for classification, as the first step as described in Section 3.1.1 already protected against a variable selection bias due to contamination. Moreover, a unified framework in this second step makes the effect of robust estimation in the first step easier identifiable.

The misclassification errors resulting from the different approaches are presented in Table 2. While the robust procedure yields low errors for both training and test data, the errors for the classical procedure with the contaminated data are much higher.

Sparse logistic regression yields estimated posterior probabilities for each sample. Fig. 5 shows the posterior probabilities of the samples to belong to Group B, again based on the models from the robust (Fig. 5a) and the classical approach (Fig. 5c for clean, Fig. 5b for contaminated data). The horizontal axis in the plots is the response variable *duration*, which has been used in the screening step. Again, the colors correspond to group membership and the symbols refer to regular, contaminated, and identified outlying observations. One can see that the non-robust models suffer from bias: for the smallest value of *duration*, only Group A observations are misclassified. In fact, this bias can already be seen in Fig. 2c. For LASSO on the clean data, the model seems to be too much adjusted to the training data, as several test set observations are wrongly classified for small values of *duration*, see also Table 2. The robust procedure seems much more balanced for the two groups (Fig. 5a). Here, the outliers as identified in the first step (see also Fig. 2a) are indicated by pink squares. All these outliers, including the contaminated samples, are clearly assigned to the correct groups by the classifier, indicating an appropriate pre-selection of the variables.

3.2. Robust regression with image data

The performance of lubricants is measured in terms of friction and



(a) sparseLTS for contaminated data (b) LASSO for contaminated data (c) LASSO for clean data

Fig. 5. Response variable *duration*, used in the regression step, against the posterior probability for Group B, with the cutoff at 0.5. Misclassified observations from the training and test set are shown "on the wrong side" of this cutoff.

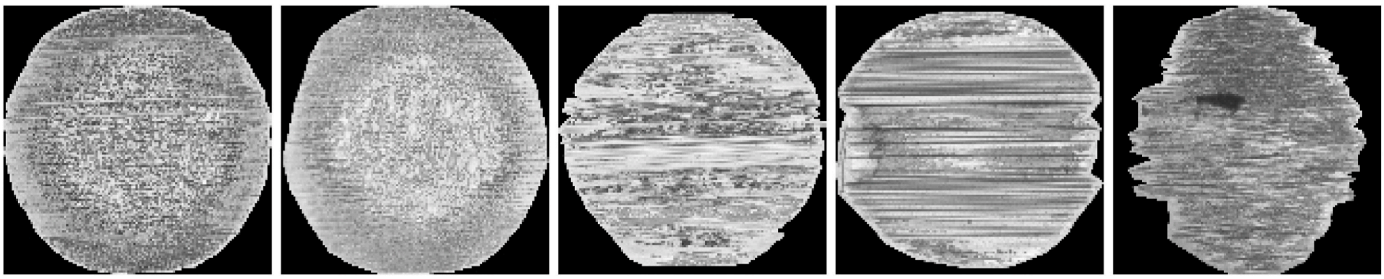


Fig. 6. Example images of the wear experiments with varying oil condition. From left to right, the duration the oil was used is 0 min, 20 min, 10 h, 50 h, and 100 h.

wear, and a reliable model associating the degradation stage of the engine oil with wear would therefore be useful for practitioners. Wear properties under laboratory conditions can be evaluated in a tribometrical experiment, where a steel ball and disc with the oil of interest in between are sliding against each other in a reciprocating contact on an SRV® tribometer (see Ref. [47] for a more detailed description of the experiment). For the present data set, wear scars were created from experiments with oil samples that were used on an engine test rig (according [48]) for up to 100 h, with 38 samples taken at 0 min, 20 min, 20 h, 50 h and 100 h. Then, images of the wear scars were recorded with an optical microscope. The statistical task is to predict the duration the engine oil was used based on the image data of the wear scar areas. For this analysis, only the wear scar images of the balls were used. The original image data are recorded as RGB images in high resolution (2600×2000 pixels), and have to be pre-processed before training a regression model. In a first step, the images were converted to greyscale based on brightness. Then, the images were annotated to segment two classes: the wear scars in the foreground and the background, which was discarded. Next, the images were scaled to size 128×128 pixels and pixel values were normalized to the same range using the *minimax* method, before Histogram of Gradients (HoG) features (see Ref. [49], for example) were extracted using the Python version of `opencv` [50]. HoG features can encode the texture of an image and therefore seem to be especially suitable for the presented case. The input image is first devided into cells, then the gradient magnitude and orientation for each pixel are computed, before they are normalized and collected in histograms. Depending on the cell and bin size for the histogram, a certain number of features is extracted from the input image. Note that there is a variety of textural image features available, with features based on Deep Learning being the most powerful ones [51]. However, training such models on as few as 38 images would only be feasible in combination with a suitable pre-trained model, while HoG features in combination with robust statistical models already lead to a good predictive performance.

Fig. 6 shows example images of the ball as a result of wear experiments with oil at different degradation stages: from left to right, the duration the oil was used on the engine test rig is 0 min, 20 min, 10 h, 50 h, and 100 h. It can be observed that with a longer duration on the engine test rig (and therefore worse oil condition) there are more and more artifacts in the images. The visible lines correspond to ridges along the direction the balls were moved in. Moreover, the shape of the wear scar gets more and more distorted with the degradation of the oil and the image. The 100 h experiment also shows a dark spot, which might be due to soot or other small particles on the wear scar area.

Since the distortion caused by the experiments is heavily varying (shape of the wear scar, type of striation, appearance of spots, etc.), the model needs to be robust against such effects. The method of choice here is linear regression, but there are some challenges for a robust approach:

- Our data set consists of only $n = 38$ images, and every image is encoded by $p = 7730$ variables. With a cell size of 8, bin size of 9, and block size of 2, the HoG feature extraction initially results in 8100 variables. After the removal of columns with only zeros due to the border, the final dimension is reached. This number is already much lower than unfolding the image pixels to 16384 variables, but still, the number of variables exceeds the number of observations by far. For this "flat" data set, sparse regression methods such as LASSO regression yield very poor models, because they can select at most n variables, which seems to be far too low in order to describe the rather complex information of the images. Robust estimators using the Elastic Net penalty, like implemented in the enetLTS package, could be a compromise. However, due to the very high number of variables, the robust algorithm is not computationally feasible anymore.
- 26 out of the 38 images are taken at the beginning of the experiments (duration 0), and for the remaining durations (20 min, 10 h, 50 h, and 100 h) we only have 3 images per duration time. Since this response variable y is extremely skewed, we will work with the transformed variable $y^{1/3}$. Still, robust regression methods either lead to very poor models, or the procedures even stop with an error. The reason is the imbalancedness of the response: The robust methods try to fit the data majority, which is for the group $y = 0$, and data with duration larger than zero are treated as outliers. A regression model only for the zero-group is of course useless.

In contrast to robust procedures, non-robust methods such as PLS regression work without any problem. Thus, the question is whether robustness can still be employed, and whether it leads to any advantage.

A first naive attempt is to exclude outliers in the x-space, i.e. we perform outlier detection only for the image data information. However, since we do not want to exclude images for the small groups with positive values of duration, outlier detection is only applied for the 26 images where the duration is zero. We use the method *pcout* as described in Ref. [52], implemented as function `pcout()` in the R package `mvoutlier`, which also works for very high-dimensional data. The algorithm identified 6 out of the 26 observations as multivariate outliers. PLS regression, as well as PRM regression, can then be applied to the cleaned data.

In order to evaluate and compare the different strategies, we randomly select around 2/3 of the observations (once for the complete and once for the cleaned data), and fit a model. We compare PLS regression with PRM, however, for PRM the internal weights are only used for the group with duration zero, and otherwise the weights are set to 1 in order to avoid downweighting of the observations for these small groups. The models are evaluated with the remaining test set observations by the RMSE as the measure of prediction quality. Here another issue occurs: As the discrete values of the response are very unevenly distributed, a pure random selection of observations could lead to training data where data groups are underrepresented or even absent. Therefore, we also compare the results for a stratified sampling approach: the training sets will consist of about 2/3 of the observations for the group with duration zero, and 2 out of the 3 randomly selected samples from every of the other groups. Each experiment is repeated 50 times.

The resulting RMSE values are presented as boxplots in Fig. 7. There is not much difference between using all observations or the cleaned

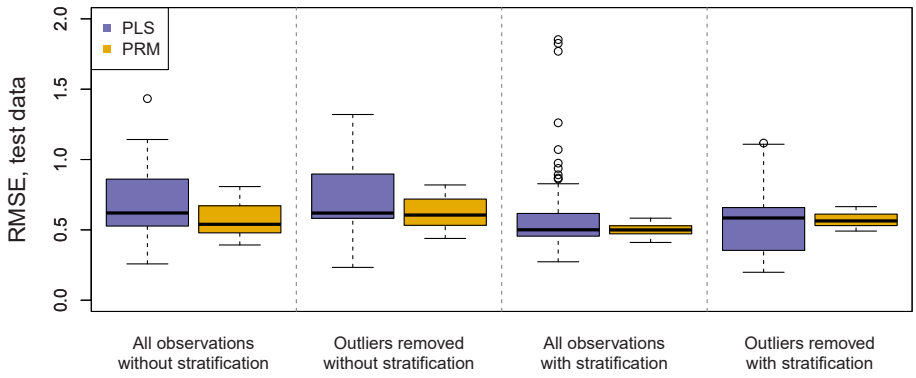
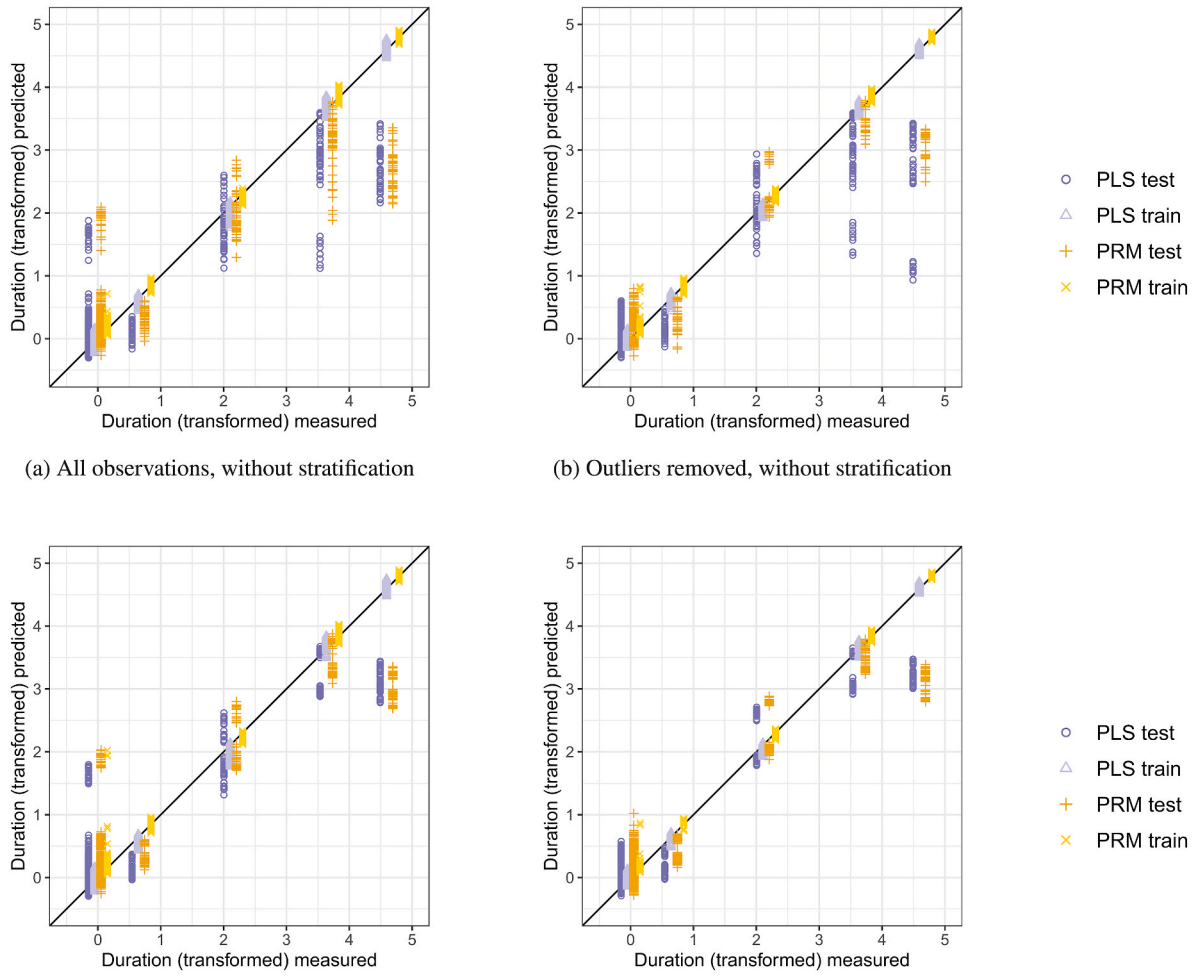


Fig. 7. Prediction errors for classical (PLS) and robust (PRM) estimation, following different strategies for data cleaning and training/test sample selection.



(a) All observations, without stratification

(b) Outliers removed, without stratification

(c) All observations, with stratification

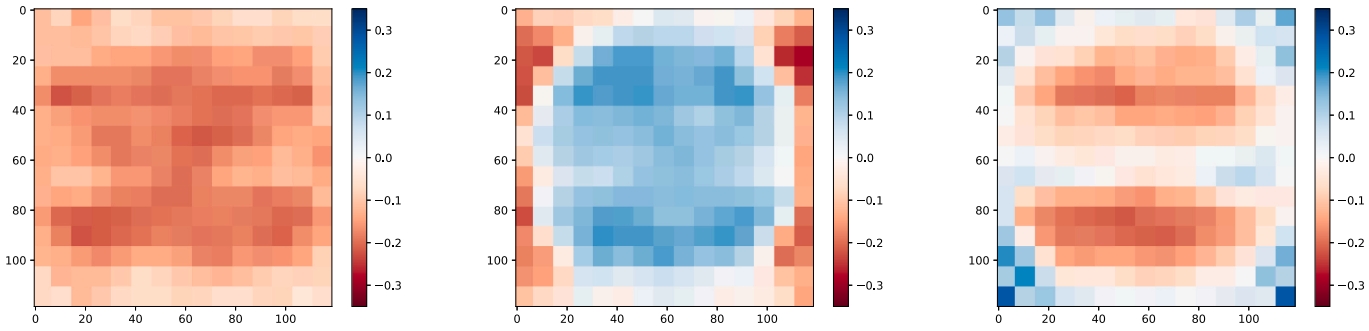
(d) Outliers removed, with stratification

Fig. 8. Training and test set predictions for all 50 PLS and PRM models, based on different strategies for data cleaning and selection.

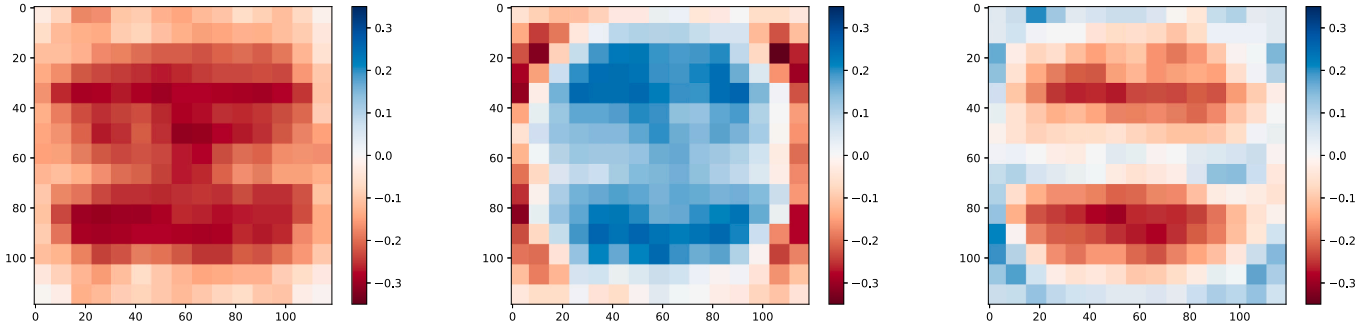
data if no stratification is used. PRM (modified) performs a bit better than PLS. Stratification clearly improves the results, and for PLS based on the uncleaned data, there are several outliers in the predictions. PRM on the uncleaned data gives very stable and good results, and the internal weighting seems to be better than first removing outliers.

More insights can be gained by the plots of the measured versus predicted (transformed) response, shown in Fig. 8, for the different strategies and the two estimators. The predictions are separately shown for the training and test set observations, and for PLS and PRM. In order to avoid overplotting, the values on the horizontal axis have been slightly changed in the plots. Overall, all models fit the training data quite well, but the test set prediction is rather poor, especially for higher values of duration. For all models 3 components were used, but the picture is the same when using e.g. 5 components, and it would become worse for a higher number.

For the approach with sample stratification we can see a reduction of the variability of the test set predictions for higher values of duration. This means that the main problem for these poor predictions is the



(a) PLS: Loading 1 (b) PLS: Loading 2 (c) PLS: Loading 3



(d) PRM: Loading 1 (e) PRM: Loading 2 (f) PRM: Loading 3

Fig. 9. The loadings traced back to the image domain for both PLS (top row) and PRM (bottom row) loadings. The loadings that are shown are those from the best-case scenario when the extracted features were robustly cleaned before applying stratified sampling and estimating the model. The color scale ranges from red (negative) to blue (positive) and the intensity corresponds to the respective section's contribution. It can be seen that while both methods seem to rely on similar areas in the original images, PRM is more precise and confident in its choice. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

imbalancedness of the data set. In the groups with duration zero there is a clear difference whether outliers are removed or not; in the latter case, outliers are visible in the test set predictions, for PLS as well as for PRM. However, as for PRM one also obtains a robust scale estimate of the residuals, these observations would be reliably identified as outliers. The same applies to the deviating predictions for higher values of duration.

To gain a better understanding of what the three latent components represent, the PRM and PLS loadings can be shown in the image domain (up to a normalization factor). In Fig. 9, the loadings from an exemplary train-test split are shown as sections of the original images. The color scale ranges from red (negative) to blue (positive), and the intensity can be interpreted as the importance of the respective sections. While both PLS and PRM rely on similar features of the images, it can be observed that PRM is more confident, yielding more intense colorings. This is especially visible in the first, and most important, loading. Also, the first loading clearly corresponds to wear marks such as horizontal scratches on the ball's surface, while higher-order loadings also have contributions from the border of the wear scars. This emphasizes the usefulness of the presented methods to analyze the given data, also in terms of interpretation.

Overall, we can conclude that the high-dimensional image information yields good models for predicting the duration of the experiments, with the exception of the trials with duration 100 h. A main difficulty here was the imbalancedness of the values of the response, but outliers also had a negative effect. Outlier cleaning before fitting the models makes almost no difference, for PLS as well as for PRM, but stratification clearly improves the results. The robust PRM method performs best in general; only for the stratified version on (robustly!) cleaned data, PLS can compete.

4. Conclusions

Imbalanced and flat data sets with fewer observations than variables pose challenges for statistical methods, especially for robust estimators, where outlying observations are downweighted. In this paper it was demonstrated how robust methods can still be applied, and that they lead to an improved performance. To handle difficulties in model estimation, approaches that split the task in two or more steps have been shown to be successful. Especially for very imbalanced data sets, an appropriate sampling strategy was found to be crucial for the derivation of a good model as well. For a data set consisting of FTIR spectra of engine oils, a robust and sparse regression estimator was applied for the prediction of oil degradation, measured in the duration the oil was subjected to alteration. The resulting model was also demonstrated to be useful as a variable screening procedure: The selected variables, now adjusted to the different degradation stages, were used as input for a classification model. For very high-dimensional data like textural image features, sparse estimators like LASSO were found to yield very poor results, as they cannot select enough variables to represent the image information. PRM, a robust PLS method, could however be applied in combination with a stratified sampling strategy.

The given examples illustrate that, even when the direct application of robust methods is not possible, combined approaches with appropriate pre-processing and sampling methods yield improved results when compared to traditional methods. What is more, they can identify observations that do not follow the majority of the data and therefore offer additional insights.

CRediT authorship contribution statement

Pia Pfeiffer: Formal analysis, Software, Visualization, Writing – original draft. **Peter Filzmoser:** Conceptualization, Writing – original draft, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work was funded by the Austrian COMET-Program (project InTribology1, no. 872176) via the Austrian Research Promotion Agency (FFG) and the federal states of Niederösterreich and Vorarlberg and was carried out at the Austrian Excellence Centre of Tribology (AC2T research GmbH) and the TU Wien. The authors acknowledge TU Wien Bibliothek for financial support through its Open Access Funding Programme. The authors thank Bettina Ronai (AC2T research GmbH) for data acquisition and providing insights about oil chemistry and Christoph Eder (TU Wien) for performing image annotations.

References

[1] Mohammad Ahamd Al-Ghouti, Yahya Salim Al-Degs, Mohammad Amer, Application of chemometrics and FTIR for determination of viscosity index and base number of motor oils, Talanta 81 (3) (2010) 1096–1101.
[2] Yulia Felkel, Nicole Dörr, Florian Glatz, Varmuza Kurt, Determination of the total acid number (TAN) of used gas engine oils by IR and chemometrics applying a combined strategy for variable selection, Chemometr. Intell. Lab. Syst. 101 (1) (2010) 14–22.
[3] Diego Rivera-Barrera, Hoover Rueda-Chacón, V. Daniel Molina, Prediction of the total acid number (TAN) of colombian crude oils via ATR–FTIR spectroscopy and chemometric methods, July 2019, Talanta 206 (2020), 120186.
[4] Charlotte Besser, Nicole Dörr, Franz Novotny-Farkas, Varmuza Kurt, Guenter Allmaier, Comparison of engine oil degradation observed in laboratory alteration and in the engine by chemometric data evaluation, Tribol. Int. 65 (37–47) (2013).
[5] Pia Pfeiffer, Bettina Ronai, Georg Vorlaufer, Nicole Dörr, Peter Filzmoser, Weighted lasso variable selection for the analysis of FTIR spectra applied to the prediction of engine oil degradation, Chemometr. Intell. Lab. Syst. 228 (2022), 104617.
[6] Arthur E. Hoerl, Robert W. Kennard, Ridge regression: biased estimation for nonorthogonal problems, Technometrics 12 (1970) 55–67.
[7] Robert Tibshirani, Regression shrinkage and selection via the lasso, J. Roy. Stat. Soc. 58 (1) (1996) 267–288.
[8] Hui Zou, Trevor Hastie, Regularization and variable selection via the elastic net, J. Roy. Stat. Soc. B 67 (2) (2005) 301–320.
[9] Danielle Witten, Robert Tibshirani, Penalized classification using Fisher's linear discriminant, J. Roy. Stat. Soc. B 73 (2011).
[10] Jerome Friedman, Trevor Hastie, Robert Tibshirani, Regularization paths for generalized linear models via coordinate descent, J. Stat. Software 33 (1) (2010) 1–22.
[11] Ricardo Maronna, Douglas Martin, Victor Yohai, Robust Statistics: Theory and Methods, Wiley, New York, 2006.
[12] Martin Maechler, Peter Rousseeuw, Christophe Croux, Valentin Todorov, Andreas Ruckstuhl, Matias Salibian-Barrera, Tobias Verbeke, Manuel Koller, L. T. Eduardo Conceicao, Maria Anna di Palma, robustbase: Basic Robust Statistics, 2022. R package version 0.95-0.
[13] Peter J. Rousseeuw, Least median of squares regression, J. Am. Stat. Assoc. 79 (388) (1984) 871–880.
[14] Peter J. Rousseeuw, Katrien Van Driessen, Computing LTS regression for large data sets, Data Min. Knowl. Discov. 12 (1) (2006) 29–45.
[15] Varmuza Kurt, Peter Filzmoser, Introduction to Multivariate Statistical Analysis in Chemometrics, 1 edition, CRC Press, 2009.
[16] Juan A. Gil, Rosario Romera, On robust partial least squares (pls) methods, J. Chemometr. 12 (1998).
[17] Ian N. Wakeling, H.J.H. Macfie, A robust pls procedure, J. Chemometr. 6 (4) (1992) 189–198.
[18] David J. Cummins, C. Webster Andrews, Iteratively reweighted partial least squares: a performance analysis by Monte Carlo simulation, J. Chemometr. 9 (6) (1995) 489–507.
[19] Mia Hubert, Karlien Vanden Branden, Robust methods for partial least squares regression, J. Chemometr. 17 (10) (2003) 537–549.
[20] Sven Serneels, Christophe Croux, Peter Filzmoser, J. Pierre, Van Espen, Partial robust M-regression, Chemometr. Intell. Lab. Syst. 79 (1–2) (2005) 55–64.
[21] Zhonghao Xie, Xi'an Feng, Xiaojing Chen, Partial least trimmed squares regression, Chemometr. Intell. Lab. Syst. 221 (2022), 104486.
[22] Peter Filzmoser, Sven Serneels, Ricardo Maronna, Christophe Croux, Robust multivariate methods in chemometrics, second edition edition, in: Steven Brown, Romà Tauler, Beata Walczak (Eds.), Comprehensive Chemometrics, second ed., Elsevier, Oxford, 2020, pp. 393–430.
[23] Esra Polat, The effects of different weight functions on partial robust m-regression performance: a simulation study, Commun. Stat. Simulat. Comput. 49 (4) (2020) 1089–1104.
[24] Irene Hoffmann, Sven Serneels, Peter Filzmoser, Christophe Croux, Sparse partial robust m regression, Chemometr. Intell. Lab. Syst. 149 (2015) 50–59.
[25] Sven Serneels, Irene Hoffmann, Sprm: Sparse and Non-sparse Partial Robust M Regression and Classification, 2015. R package version 1.2.
[26] Emmanuel Jordy Menvouta, Sven Serneels, Tim Verdonck, direpack: A python 3 package for state-of-the-art statistical dimensionality reduction methods, SoftwareX 21 (2023), 101282.
[27] Andreas Alfons, Christophe Croux, Sarah Gelper, Sparse least trimmed squares regression for analyzing high-dimensional large data sets, Ann. Appl. Stat. 7 (1) (2013) 226–248.
[28] Efron Bradley, Trevor Hastie, Iain Johnstone, Robert Tibshirani, Least angle regression, Ann. Stat. 32 (2) (2004) 407–499.
[29] Jafar A. Khan, Stefan Van Aelst, Ruben H. Zamar, Robust linear model selection based on least angle regression, J. Am. Stat. Assoc. 102 (480) (2007) 1289–1299.
[30] Alfons Alfons, *robustHD*: Robust Methods for High Dimensional Data, R Foundation for Statistical Computing, Vienna, Austria, 2016. R package version 0.4.0.
[31] Richard A. Johnson, Dean W. Wichern, Applied Multivariate Statistical Analysis, sixth ed., Prentice Hall, Upper Saddle River, 2007.
[32] Christophe Croux, Catherine Dehon, Robust linear discriminant analysis using s-estimators, Can. J. Stat. 29 (2) (2001).
[33] Mia Hubert, Katrien Van Driessen, Fast and robust discriminant analysis, Comput. Stat. Data Anal. 45 (2) (2004) 301–320.
[34] Valentin Todorov, Ana Pires, Comparative performance of several robust linear discriminant analysis methods, Revstat - Statistical Journal 5 (63–83) (2007).
[35] Valentin Todorov, Peter Filzmoser, An object-oriented framework for robust multivariate analysis, J. Stat. Software 32 (3) (2009) 1–47.
[36] Kris Boudt, Peter J. Rousseeuw, Steven Vanduffel, Tim Verdonck, The minimum regularized covariance determinant estimator, Stat. Comput. 30 (1) (2020) 113–128.
[37] Fatma Sevinç Kurnaz, Irene Hoffmann, Peter Filzmoser, Robust and sparse estimation methods for high-dimensional linear and logistic regression, Chemometr. Intell. Lab. Syst. 172 (2017) 211–222.
[38] Fatma Sevinç Kurnaz, Irene Hoffmann, Peter Filzmoser enetLTS, Robust and Sparse Methods for High Dimensional Linear and Binary and Multinomial Regression, 2022. R package version 1.1.0.
[39] Jasin Machkour, Michael Muma, Bastian Alt, M. Abdelhak, Zoubir, A robust adaptive lasso estimator for the independent contamination model, Signal Process. 174 (2020), 107608.
[40] Lea Bottmer, Christophe Croux, Ines Wilms, Sparse regression for large data sets with outliers, Eur. J. Oper. Res. 297 (2) (2022) 782–794.
[41] Nicole Dörr, Josef Brenner, Andjelka Ristic, Bettina Ronai, Charlotte Besser, Vladimir Pejaković, Marcella Frauscher, Correlation between engine oil degradation, tribochemistry, and tribological behavior with focus on ZDDP deterioration, Tribol. Lett. 67 (2019).
[42] Charlotte Besser, Agocs Adam, Bettina Ronai, Andjelka Ristic, Martin Repka, Erik Jankes, Colin McAleese, Nicole Dörr, Generation of engine oils with defined degree of degradation by means of a large scale artificial alteration method, Tribol. Int. 132 (2019) 39–49.
[43] Marcus Mayrhofer, Peter Filzmoser, Multivariate Outlier Explanations Using Shapley Values and Mahalanobis Distances, Econometrics and Statistics, 2023.
[44] Michiel Debruyne, Sebastiaan Höppner, Sven Serneels, Tim Verdonck, Outlyingness: which variables contribute most? Stat. Comput. 29 (2019) 707–723.
[45] Peter Filzmoser, Sebastiaan Hoppner, Irene Ortner, Sven Serneels, Tim Verdonck, crmReg: Cellwise Robust M-Regression and SPADIMO, 2020. R package version 1.0.2.
[46] Jerome Friedman, Trevor Hastie, Robert Tibshirani, Regularization paths for generalized linear models via coordinate descent, J. Stat. Software 33 (1) (2010) 1–22.
[47] Agocs Adam, Charlotte Besser, Josef Brenner, Serhiy Budnyk, Marcella Frauscher, Nicole Dörr, Engine oils in the field: a comprehensive tribological assessment of engine oil degradation in a passenger car, 3, Tribol. Lett. 70 (2022).
[48] ASTM D7484, Standard Test Method for Evaluation of Automotive Engine Oils for Valve-Train Wear Performance in Cummins ISB Medium-Duty Diesel Engine, ASTM International, West Conshohocken, PA, USA, 2021.
[49] Simon J.D. Prince, Computer Vision: Models Learning and Inference, Cambridge University Press, 2012.

[50] G. Bradski, The OpenCV Library, Dr. Dobb's Journal of Software Tools, 2000.
[51] Anne Humeau-Heurtier, Texture Feature Extraction Methods: A Survey, IEEE Access, PP:1–1, 2019.
[52] Peter Filzmoser, Ricardo Maronna, Mark Werner, Outlier identification in high dimensions, Comput. Stat. Data Anal. 52 (3) (2008) 1694–1711.