Erasmus University Rotterdam

Business Analytics and Quantitative Marketing

Master Thesis

Robust finite mixture models by

Median-Of-Means in the linear regression context

***Author*:**
Daan Veldhuizen (483082)

***Supervisor*:**
Dr. Andreas Alfons

27th October 2021

Abstract

This paper proposes an estimation method that robustifies the finite mixture model with the use of the Median-Of-Means (MOM) estimator proposed by Lecue & Lerasle (2020). This method replaces expected values in the original estimation algorithm by the MOM-estimator and optimises this new problem. The used data consists of simulations from four data generating processes (DGP) and an empirical data set. Two of the DGP's generate linear or soft-clustered data. The other two DGP's add different types of outliers to these data sets. The results indicate that for uncontaminated data, there is no superior method. For the contaminated data, however, the MOM estimator outperforms maximum likelihood estimation. Nevertheless, this robustification does not hold with certainty for small samples.

Contents

1 Introduction

Not a single person has the exact same preferences. Take for example the learning and personality preferences of students (Durling et al., 1996). In this paper the focus lies on art-based students and the conclusion is that their way of problem-solving differs significantly from other professions. Nevertheless, most econometric models, such as the linear regression model, do not take these differences into account (Heij et al., 2004). The finite mixture model, however, is a model that solves this problem and does include this so-called unobserved heterogeneity. Be that is it may, the problem with this model is that similarly to least squares, the estimations are easily affected by outliers in the data. This is because one uses maximum likelihood to optimise these estimations (G. J. McLachlan & Peel, 2004). Maximum likelihood makes use of the whole data set, including the outliers. This can then result in estimations that are unreliable (Lecue & Lerasle, 2020). Nevertheless, simplifying the model by neglecting unobserved heterogeneity and then using robust linear regression may also lead to weak performance of the proposed model (G. J. McLachlan & Peel, 2004). This is the reason why in this paper, we combine robust statistics with the finite mixture model.

Some research has already been conducted on this problem. Yao et al. (2014) for example, replace the normal distribution that is traditionally used for the finite model by the Student's t-distribution. The reason they do this is because of the heavy tails that a t-distribution contains. In other words, the probabilities corresponding to extreme values are larger compared to that of the normal distribution. In this way one can take outliers into account. This idea can also be used for the replacement of a skew normal distribution (Lin et al., 2007). In this paper they replace the latter by the skewed t-distribution. The problem with both these methods is that there are still extreme values that are not tackled by the t-distribution due to the fact that the tails of this distribution will still converge too zero as the values get more extreme. The focus in our paper therefore lies on a different robust method that does not have this attribute: Median-Of-Means (MOM) estimation (Alon et al., 1999). This method filters the data by creating subsets and then estimate the coefficients of the model for each of these subsets. The subsets with outliers will have other estimations compared to those of the uncontaminated subsets. If one then takes the median over these results, the selected estimation will then likely not be affected by outliers, as the majority of the subsets does not contain outliers. The contaminated estimations will be on the outside of the sequence of estimations and will not be chosen as the median. In this way one takes out the outliers in the data. We combine this method with the finite mixture model. By doing so, we are able to robustify the finite mixture model. This leads to the following research question:

- *"How can one use the Median-Of-Means to effectively combine unobserved heterogeneity and robustness within a linear regression framework?"*

Some papers have already used the MOM estimator for the robustification of the linear regression problem (Lugosi & Mendelson, n.d.; Lecue & Lerasle, n.d.). According to Lecue & Lerasle (2020), however, these researchers have come up with complex methods to use the MOM estimator and robustify the linear regression model. This is why they propose a method where the MOM estimator is used differently. Their proposed method uses the MOM estimator as an optimisation problem instead of an estimator. This means that the expected value that one normally optimises in the linear regression context is replaced by the MOM estimator. The latter is then optimised, resulting in robust estimations. The advantage of this method is that it does not only robustify, but also simplifies the optimisation problem compared to the papers by Lugosi & Mendelson (n.d.); Lecue & Lerasle (n.d.). This is why we use this idea in our paper.

The way our proposed method works is as follows. As mentioned earlier, the optimisation of the finite mixture is traditionally done using maximum likelihood. Nevertheless, due to the multiplicative nature of the model, this cannot be done by optimising the log-likelihood function directly (Muthen & Shedden, 1999). We must use the so-called EM algorithm. This algorithm consists of two steps: Expectation and Maximisation. In the first step an expectation is set up. This expectation is then optimised in the maximisation step. This expectation can be split up into individual expectations similar to the expectation that is optimised in the linear regression problem. (G. J. McLachlan & Peel, 2004). The optimisation of these expected values are easily affected by outliers (Yao et al., 2014). In our case, this means that the maximisation step is prone to contaminated data. We solve this, however, by replacing this expectation with the robust MOM estimator similar to Lecue & Lerasle (2020). By doing so, we obtain several robust loss functions and the estimations of the coefficients are going to be robust as well. For so far as we know, this method has never been done for the finite mixture model.

Our results show that for larger data sets, the MOM estimator either outperforms or performs equally compared to the maximum likelihood estimator for the contaminated and uncontaminated data. For the smaller data sets, however, this is not the case. For these data sets we see that the variance of the MSE increases, which indicates the unreliability of the performance results. So, for small samples the MOM estimator does not robustify the finite mixture model with certainty. The conclusion is confirmed even more when using both the maximum likelihood and MOM estimator for an empirical application: For this small empirical data set, we conclude that the MOM estimator performs less well than the maximum likelihood estimator.

The remainder of the paper is structured as follows. Section 2 discusses relatable papers and research that has previously been conducted. In section 3 we then go through the methodology and estimators used throughout our research. Section 4 discusses the simulation design and corresponding performance results of the proposed estimators. In Section 4 we then analyse the empirical data set and the corresponding results. Lastly, we conclude our research in Section 5 and go through our limitations in Section 6.

2 Literature

2.1 Related studies

In the past years people have conducted research on the subjects of robustness and unobserved heterogeneity in econometric models. The robustness of models ensures that contaminated data, which is data that contains outliers, does not affect the estimation results. Including unobserved heterogeneity into models ensures that we takes personal preferences into account.

In this section we go through to some of this previously conducted research that is relatable to this paper.

2.1.1 Unobserved heterogeneity

When we look at the linear assumption that is made in OLS, we notice that every explanatory variable corresponds to a single coefficient. After the estimation of these coefficients, we use these estimations for forecasts of data points in the test set. This means that for each of these data points, one uses the same estimation of the coefficients. But is this always the case? In practice, this assumption is not often satisfied (Heij et al., 2004). This problem is called unobserved heterogeneity and means that the effect of each explanatory variable on the dependent variable may differ for different data points. G. J. McLachlan & Peel (2004) use a model called the finite mixture model that takes this unobserved heterogeneity into account. The way this model works is by using different clusters, each with different coefficients. A data point then belongs to a certain cluster with a corresponding probability. In other words, the model uses soft-clustering. By doing so, one obtains a linear combination of the different coefficients belonging to the different clusters and therefore every effect that an explanatory variable has on the dependent variable is going to be different for every data point.

There is, however, a disadvantage to the finite mixture model. Due to the clustering within the model, maximum likelihood optimisation becomes difficult and time-consuming. To still find an optimal solution, we must therefore use a method called the EM-algorithm. This algorithm allows for easier optimisation and results in the same estimation as maximum likelihood (Muthen & Shedden, 1999). However, it remains an iterative process, which still makes the usage of this model more time-consuming than other models such as the linear regression model (Veldhuizen et al., n.d.).

2.1.2 Robustness

One of the key definitions of robustness within this context is the breakdown point. This is the proportion of data points that need to be turned into arbitrarily high or low values in order for the estimator to give an extreme arbitrarily estimation (Hampel, 1971). So, to achieve robustness one seeks to find a breakdown point as low as possible. An example of a robust estimator is the MM estimator, which can be used to estimate the linear regression coefficients in a robust and efficient manner (Susanti & Pratiwi, 2014). The MM-estimator combines two estimators: The S-estimator and the M-estimator. The main property of the S-estimator is its robustness. The property of the M-estimator is its efficiency. The MM-estimator combines both, and by doing so, obtains both properties. The result is a robust estimator with high efficiency. Other examples are the papers by Lugosi & Mendelson (n.d.) and Lecue & Lerasle (n.d.). In these papers they use the so-called Median-Of-Means estimation method, which robustly estimates the location parameter of data. This method splits up the data in K subsets. For each of these sub sets the location parameter is estimated, using the mean as the estimator. This then results in a sequence of K different estimations. Finally, the median of this sequence is taken. This method assures that outliers that might appear in the original data set only have an effect on the subset that uses that outlier. Taking the median then makes sure that the contaminated estimator is not chosen as the final estimation. A clear example where this method is used in practice, is in a paper by Pazis et al. (2016). Here they use the MOM estimator for PAC exploration. PAC stands for Probably Approximately Correct, and is a method that seeks to find a solution that has a high probability (Probably) for a low error (Approximately Correct). The main assumption of this method is that there are no errors in the data. In reality however, this can definitely be the case. Pazis et al. (2016) propose to use the MOM estimator in this situation to make PAC exploration more robust. Their conclusion is that the Median-Of-Means estimator make sure there is less dependence on the range of values the optimisation can take, making the process indeed more robust.

Lecue & Lerasle (2020) use the Median Of Means to make linear regression robust. In the general linear regression setting, one optimizes an expected value (Heij et al., 2004). Lecue & Lerasle (2020) replace this expected value by the Median-Of-Means estimator. In this way the optimisation problem becomes completely different. Solving this new optimisation problem then results in robust estimations. Our paper revolves around this idea.

2.1.3 Least Median of Squares regression

Another example that robustifies linear regression is Least Median of Squares regression. Normally, one would use the sum of squared residuals as the loss function and minimise the latter to obtain the least squares solution. However, this method is highly sensitive to outliers (Lecue & Lerasle, 2020). In order to solve this, Rousseeuw (1996) proposes that a different loss function is selected. Instead of taking the sum of the squared residuals, he selects the median of these squared residuals as the chosen loss function. The reason for this is as follows. An outlier generates different squared residuals than other data points. If one then creates an ordered sequence of these squared residuals, the ones belonging to the outliers will lie on the outside of the sequence. If one then takes the median of this ordered sequence, the squared residuals belonging to the outliers will not be chosen. Minimising this median then means that one minimises a loss function that is not affected by outliers, making the result robust. In theory and in practice this works well for outliers in the dependent variable. If there are other types of outliers, this method will not give a robust estimation (Lecue & Lerasle, 2020). Nevertheless, the advantage of this method is that can also be used in a setting different from linear regression. An empirical An example where this method is used is a paper by Mili et al. (1991). In this paper they solve problem of optimising static state estimators in power systems with the help of Least Median of Squares regression.

2.1.4 Robustness and unobserved heterogeneity using the t-distribution

Other research has already been conducted on combining both robustness and heterogeneity (G. McLachlan & Peel, 2000). In this paper, they utilise the t-distribution in the finite mixture model. This method differs from the original use of the finite mixture model in the linear regression context, where they use the normal distribution (G. J. McLachlan & Peel, 2004). The t-distribution has heavier tails, meaning that values that lie further from the mean have a higher probability compared to the normal distribution. Therefore, extreme values cannot be taken into account when assuming normality. Due to heavy tailed property of the t-distribution, this is possible, however. In this way one can take outliers into account when estimating. Lin et al. (2007) extend this idea and use the skew t-distribution as a robust replacement for the skew normal distribution. This idea turns out to be useful for heterogenous data, specifically. An example where this robust method is applied is the medical image segmentation (Nguyen & Wu, 2011).

2.1.5 Regularization

Regularization is a method that is mainly used to make sure that the estimations do not overfit the data. Overfitting means that the estimations are shaped towards the trainig set. When this happens this usually to causes performance problems, because the test set might have other characteristics than the training set (Bradley et al., 2011). To prevent this from happening regularization adds a penalty term to the loss function that accounts for this overfitting problem. Even though this is its main goal, it is also useful in robust statisctics. If your training set contains outliers, these outliers usually affect the estimations. The outliers can simply interfere with the estimation, similarly to all the other data points. However, if one uses regularization, this prevents the model from overfitting. In other words, the influence of the training set on the estimation becomes less. In this way outliers will have less effect on the estimation, making the estimation more robust. A example where this is used is in a paper by Tao & Zhai (2006). In this paper they combine regularisation with the optimisation of the mixture model in order to obtain robust results.

2.2 Contribution

In this paper we propose a new estimator similar to the paper by Lecue & Lerasle (2020) to robustify the finite mixture model. Lecue & Lerasle (2020) give mathematical proof that their method, using the Median-Of-Means estimator as an optimisation problem, indeed robustifies the linear regression problem. This is confirmed even more using simulated results. Moreover, it is mentioned that their method of replacing the original optimisation problem of the linear model by the Median-Of-Means estimator has never been done before. For our research a similar reasoning holds. We show that for the finite mixture model too, optimisation problems can be replaced by the Median-Of-Means estimator. By doing so the estimation will be robust against both outliers in the dependent variables and the explanatory variables. For so far we know, the only way this robustification has been done for the finite mixture model is by using the t-distribution. Our paper will therefore add a new robust method to current literature.

2.3 Outliers in data

There different possible types of outliers in data. First of all, one may have outliers in the dependent variable. This type of outlier is also called a vertical outlier. Secondly, there are bad and good leverage points. In this case the explanatory variables take on extreme values. The difference between good and bad leverage points is the value of the dependent variable. If the dependent variable changes in such a way that the data point lies in line with the linear regression mode, that data point is considered a good leverage point. Even though these data points do not fit the rest of the data set, the corresponding values still fit the general correlation between the variables. Therefore, these data points do not affect the estimation of the coefficients. However, they might affect the estimation of the corresponding standard errors (Verardi & Croux, 2009). A bad leverage point is a point where the explanatory variables are extreme, but value of the dependent variable fits into the rest of the data. In this way the data points will not be in line with the linear regression model.

3 Methodology

In this section we discuss the models, methods and estimators we use throughout our research. It is split up into four parts. The first part discusses the finite mixture model, which is the model that includes unobserved heterogeneity. The second part explains the Median-Of-Means estimator, which is a estimator that robustifies the estimation of an expected value. The third part then explains how we combine part 1 and 2 to robustify the finite mixture model. Lastly, part four elucidates how we train and test the used models.

3.1 Finite mixture model

According to the traditional regression model, the dependent variable y has a linear relation with the explanatory variables (Heij et al., 2004). The assumption that is made here is that the coefficients, or the effect of the explanatory variables on the dependent variable are the same for every data point. In reality, however, this does not have to be the case. The model we propose to include the latter is called the finite mixture model. Instead of using a single relation between the dependent variable y and the explanatory variables $\mathbf{x}$, this model consists of C different clusters. Here $c \in \{1, 2, ...C\}$. Let $i \in \{1, 2, ..., N\}$, where N is the total number of data points. Each of the data points $(y_i, \mathbf{x}_i)$ belongs to one cluster c with probability π_c. Conditional on the cluster, the model follows a specific distribution that is set beforehand. In our case this distribution is set to be the normal distribution. The probability distribution function (PDF) belonging to this distribution equals

$$Pr_n(y_i|\mathbf{x}_i, \beta_c) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(x_i - \mathbf{x}_i'\beta_c)^2}{\sigma}}. \tag{1}$$

Here β_c is the vector of coefficients belonging to cluster c, $\mathbf{x}_i$ are the set of the explanatory variables $\{x_{i1}, x_{i2}, ..., x_{iC}\}$ of index $i \in \{1, 2...N\}$. N is the total number of data points and σ is a scale parameter corresponding to the standard deviation of the normal distribution. Finally, we use the subscript n for clarification purposes that this equation is the PDF of the normal distribution.

Combining the unconditional probabilities π_c for $c \in \{1, 2, ...C\}$ and Equation 1 we obtain the likelihood contribution of data point $(y_i, \mathbf{x}_i)$. This contribution equals

$$L_i(y_i|\mathbf{x}_i, \theta) = \sum_{c=1}^{C} \pi_c Pr_n(y_i|\mathbf{x}_i, \beta_c). \tag{2}$$

The likelihood function then equals the multiplication of all the individual likelihood contribution and becomes

$$L(\theta) = \prod_{i=1}^{N} L_i(y_i|\mathbf{x}_i, \theta). \tag{3}$$

3.1.1 Expectation Maximisation algorithm

Due to the multiplicative nature of the individual likelihood contributions the likelihood function becomes difficult to solve (Heij et al., 2004). Normally, one would use the so-called log-likelihood, the logarithm of the likelihood function, which turns the multiplication in

Equation 3 into a sum. The equation becomes

$$l(\theta) = \sum_{i=1}^{N} ln(L_i(y_i|\mathbf{x}_i, \theta)). \tag{4}$$

Equation 4 is called the log-likelihood function. Solving this new problem entails the same results, but has less computational difficulty (Heij et al., 2004). In this case, however, taking the logarithm does not make the optimisation problem easier, as the multiplication between π_c and Equation 1 remains. This is why in this case, we use the Expectation Maximisation (EM) algorithm. This algorithm makes use of step-wise optimisation, eventually resulting in the same outcome as optimising the log-likelihood using traditional maximum likelihood estimation (Muthen & Shedden, 1999). Logically, it consists of two different stages: Expectation and Maximisation.

Expectation The first step is the Expectation (E) step. In this step, we compose the expected log-likelihood. To do so, we must first define the probability that data point $(y_i, \mathbf{x}_i)$ belongs to cluster c. This is defined as

$$z_{ic} = \frac{\pi_c Pr(y_i|\mathbf{x}_i, \beta_c)}{\sum_{j=1}^{C} \pi_j Pr(y_i|\mathbf{x}_i, \beta_j)},\ i = 1, \ldots, N,\ c = 1, \ldots, C. \tag{5}$$

Plugging in the estimates $\hat{\pi}_c$ and $\hat{\beta}_c$ for all c $\in \{1, 2, ...C\}$, gives the estimate $\hat{z}_{ic}$ for Equation 5 (Melnykov et al., 2010). According to G. J. McLachlan & Peel (2004), the exptected log-likelihood then becomes

$$\mathrm{E}[l(\theta)] = \sum_{i=1}^{N} \sum_{c=1}^{C} \hat{z}_{ic} \ln(Pr(y_i|\mathbf{x}_i, \beta_c)) + \sum_{i=1}^{N} \sum_{c=1}^{C} \hat{z}_{ic} \ln(\pi_c). \tag{6}$$

Maximisation In the Maximisation (M) step, Equation 6 is optimised. Due to the independence of the elements in the sum, this can be done separately for each π_c, β_c and σ^2 Optimising π_c and σ^2 can be done analytically. The optimal solution for π_c equals

$$\hat{\pi}_c = \frac{1}{N}\sum_{i=1}^{N} \hat{z}_{ic},\ c = 1, \ldots, C. \tag{7}$$

For σ^2 the analytical solution becomes

$$\hat{\sigma}^2 = \frac{1}{N}\sum_{i=1}^{N}(y_i - x_i'\hat{\beta})^2 \tag{8}$$

The optimisation of each coefficient β_c equals a weighted least squares problem. This means that an analytical solution is possible. This solution equals

$$\hat{\beta}_c = (X'Z_cX)^{-1}X'Z_cY. \tag{9}$$

Here X is the matrix including all the data corresponding to the explanatory variables and the constant. W_c is a diagonal matrix with weights $\hat{z}_{ic}$ as diagonal matrix elements. $c \in \{1, 2, ...C\}$ corresponds to a specific cluster and i is an index corresponding to data point $(y_i, \mathbf{x}_i)$. Lastly, Y is a vector containing all data $y_i \in \{y_1, y_2...y_N\}$ that corresponds to the dependent variable.

3.2 Median-Of-Means

The Median-Of-Means estimation is a method to robustly estimate the expected value of a cloud of data points that come from an unknown distribution. Given specific assumptions the arithmetic mean is a consistent estimator for the latter (Heij et al., 2004). The Equation of the arithmetic mean equals

$$\hat{\mu}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i = \bar{\mathbf{x}}. \tag{10}$$

Here $\mathbf{x}_i$ is an observation on a set of explanatory variables corresponding to data point i. N is the total number of data points.

Sometimes, however, it might be the case that the assumptions set by Heij et al. (2004) do not hold. For example, their might be an outlier in the data, which is a data point with a value that is different from the rest of the data points. If this is the case, Equation 10 is not going to give correct estimations (Lecue & Lerasle, 2020). In order to minimise the effect of outliers on the estimations, we can use the Median-Of-Means (MOM) estimator. This estimator is calculated in several steps. Firstly, the data points are randomly split up into K different subsets. We create this randomness by first randomly shufflng the original data set. For each of these subsets, Equation 10 is then used to calculate the arithmetic mean. This results in a sequence of K values $\{\hat{\mu}_1, \hat{\mu}_2, ..., \hat{\mu}_K\}$. Secondly, the median of this sequence is taken as the final estimation. This resulting MOM estimator then becomes

$$MOM(\mu) = med(\{\hat{\mu_1}, \hat{\mu_2}, ..., \hat{\mu_K}\}). \tag{11}$$

The split-up of data ensures that outliers can only have an effect on one estimation of the mean. This effect ensures that these affected arithmetic means differ from the uncontaminated estimations. Due to the randomness, these estimations should be similar in terms of value. The affected estimation therefore becomes an extreme value in the sequence of estimations. When the median is taken, the affected estimations will therefore not be chosen. In other words, this method ensures robustness (Alon et al., 1999).

3.3 Using MOM in the the finite mixture model

The MOM method ensures robustness for the estimation of an expected value. But can we also use it in the linear regression setting? Lecue & Lerasle (2020) propose a way to do so by replacing the expected value

$$E((Y - X'\beta)^2), \tag{12}$$

which is the animal the OLS estimator seeks to optimise, by the Median-Of-Means estimator. Let $(Y - X'\beta)^2$ be equal to λ. The Median-Of-Mean estimator for Equation 12 then becomes

$$MOM(\lambda) = med(\{\bar{\lambda_1}, \bar{\lambda_2}, ..., \bar{\lambda_k}\}). \tag{13}$$

Here, each $\bar{\lambda_j}$ for $j \in \{1, 2, 3, .., K\}$ equals the arithmetic mean to estimate Equation 12 for each subset j.

We use this exact method in the finite mixture model setting. To see how this works, one must look at Equation 6 from a different of view. Equation 9 indicates that for each different component in Equation 6 corresponding to a specific cluster c, the optimal outcome is equal to the weighted least squares estimator. Therefore one should not think of this optimisation problem as a maximum likelihood problem. Instead, the estimation of each different component equals the optimisation of a weighted least squares problem, which is a generalisation of a least squares problem. The weighted least squares problem also optimises Equation 12 (Kiers, 1997). This means that conditional on cluster c an expected value equal to Equation 12 is optimised in the finite mixture model. We are therefore able to use the idea proposed by Lecue & Lerasle (2020), and replace these expected values. The difference however, is that the arithmetic means in Equation 13 are replaced by weighted means equalling

$$\mu_{weighted}(\mathbf{x}) = \frac{1}{S_c} \sum_{i=1}^{N_k} z_{ic}(y_i - (\mathbf{x}_i' \beta_c))^2. \tag{14}$$

Here β_c is the vector of coefficients belonging to cluster c, $\mathbf{x}_i$ are the set of the explanatory variables $\{x_i 1, x_i 2, ..., x_i C\}$ of index $i \in \{1, 2...N_k\}$ and y_i is the value of the depending variable corresponding to index i. Here N_k equals the amount of data points in subset k. Lastly, S_c equals $\sum_{i=1}^{N_k} z_{ic}$. Each of the weighted MOM-estimators are then optimised independently for each cluster c. A frequently used method for this optimisation is the gradient descent algorithm. For these kind of algorithms one seeks to find the optimal value. This value is found by setting steps in the direction of the steepest descent. It can be shown that this direction equals the opposite of the gradient, hence the name gradient descent (Ruder, 2016).

3.4 Hold-out sample and performance measure

3.4.1 Training and validation set

In order to be able to test the performance of the different methods on our simulated data set we use a three way hold-out sample. This means that we split up the data into three different subsets, with three different purposes. Let $(N_{tr} + N_v + N_t)$ be the total number of data points in our full data set. The first subset is the training set consisting of N_{tr} data points. We use this set for learning, which equals the estimation of the different parameters in the model. In other words, this is the data set that we use as input for either the EM algorithm or our newly proposed robust MOM estimation method.

The second subset is called the validation set. This set consists of N_v data points. We use this set to optimise the hyperparameters, which are the parameters that are used to control the learning process and performance of a model. In our case there are two hyperparameters: The number of clusters in the finite mixture model and the number of subsets for the MOM estimator. We optimise the latter in the following way. We first select a sequences of values that we want to use for the number of clusters and the number of subsets. Along with the training set, we then use each of these values as an input to train the model. This results in different estimations of the coefficients for the different values of the sequences. One then uses the validation set to test the performance for each value of these estimations The value with the best performance gets selected as the optimal choice. Once the number of clusters is selected, the training set and validation set get combined to obtain a data set consisting of $(N_{tr} + N_v)$ data points. We use this combined set to train the model one final time, resulting in the final estimation of the coefficients.

3.4.2 Test set

The remaining test set is then used to measure the final forecasting performance of the model. For both validation and final testing the performance is measured by the mean squared error. We define this performance measure as

$$MSE(y, \mathbf{x}, \hat{\theta}) = \frac{1}{N_t} \sum_{i=1}^{N_l} (y_i - \hat{f}_i(\mathbf{x}_i' \beta_c)). \tag{15}$$

Here y is the vector that contains all the values of the dependent variable for the test-set and $\mathbf{x}$ is the matrix containing all the values for the vector of explanatory variables. Moreover, N_l is the total number of data points and $\hat{\theta}$ is the set that contains all the estimations for the unknown parameters within the model. The subscript l resembles what data set one uses: the validation set v or the test set t. Lastly we define $\hat{f}_i(\mathbf{x}_i'\beta_c)$ as

$$\hat{f}_i(\mathbf{x}_i'\beta_c) = \sum_{c=1}^{C} \hat{z}_{ic}(\mathbf{x}_i'\beta_c). \tag{16}$$

Here $\hat{z}_{ic}$ corresponds to Equation 5 and $c \in \{1, 2...C\}$. C is the total number of clusters. We use this to optimise the number of clusters and compare across the different methods, but also to see how different data sets change the performance of a specific model.

4 Simulation design and results

In this section we go through the different data generating processes used for our research and analyse the corresponding results. We simulate data using four different data generating processes. Two of the latter are uncontaminated, in the sense that they create data that does not contain any outliers. The other two data generating processes do include outliers, however. The difference between these processes is the type of outliers. For each data generating process small and large data sets are created. We use the small data set to see how well the proposed estimator performs for finite sets. We then use the large for the asymptotic properties. The number of data points for every small data set is set to 300.

For the large data set this number equals 3000. For both the small and large data sets we conduct a total of ten simulations.. All programming is done in Rstudio (Rstudio, 2020).

4.1 Simulation design

4.1.1 Data without outliers

Recall that this research revolves around the linear regression context. Therefore we choose to simulate data that is in line with this model. According to Heij et al. (2004), the general linear regression assumptions correspond to data that have a linear relation between the dependent variable y and explanatory variables $\mathbf{x}$ and an error term that has a normal distribution. Mathematically, this equals

$$y_i = \mathbf{x}_i'\beta + \epsilon \tag{17}$$

Here i is an index that indicates a data point $i \in \{1, 2, ..., N\}$. N is the total number of data points.

In other words, this means that the explanatory variables must come from a normal distribution with a mean equal to $\beta'\mathbf{x}_i$ and a variance σ^2 one can choose. For visualisation purposes, Figure 1 displays an example of a two dimensional simulation. Note that this is not the data set we are using. Nevertheless, it visualises the linearity of the data generating process.

For the simulation of the used data we select σ to be 5. The chosen coefficients for each variable can be seen in Table 1. We simulate the data using three variables and a constant. We then assume that these variables are fixed (Heij et al., 2004). By using Equation 17 we are then able to surmise that the dependent variable indeed comes from a normal distribution with a mean of $\beta'\mathbf{x}$ and a σ equalling 5. Nevertheless, we must simulate the explanatory variables from some distribution. In other words, it is impossible for this assumption to hold. By selecting the normal disrtibution for the simulation design of the explanatory variables, we are able to mimic this assumption, however. When doing this, Equation 17 becomes a sum of normal distributions, which results in the dependent variable y_i being normally distributed (Halfens & Meijers, 2012). In this way the assumption of fixed regressors might not hold, but at least the dependent variable y_i does have the aspects of normality. For the simulation of the explanatory variables we select the variance to be 4 and the mean to be 10. In this way we select the data to lie around 10, with some spread. This is necessary to obtain decent estimations (Heij et al., 2004). We then select this first data generating process as the basis of our research and use it to see how the methods perform in the ideal situation of a linear data.

coefficients	**values**
β_0	1
β_1	7
β_2	4
β_3	5

Table 1: Coefficients used for simulation of linear data.

Discrete data For the second data generating process we add complexity by using the finite mixture model as a basis. This model assumes that the dependent variable y_i may come from different distributions, each with their own probability. For example, y_i may come from distribution 1 with probability π_1, come from distribution with probability π_2 with probability π_2 etc. The number of possible distributions is set beforehand. In literature, the number of possible distributions is defined as the number of clusters. Therefore, this is what we use throughout this paper. We define these clusters to be distributed similarly to the linear regression context. This means that given a cluster, the dependent variable y_i is normally distributed with mean $\beta_c'\mathbf{x}$ and a variance σ^2 that is chosen beforehand. Once again we make the assumption that the regressors $\mathbf{x}$ are fixed. In other words,

$$Pr_n(yi|\mathbf{x}_i, \theta) = \sum_{c=1}^{C} \pi_c Pr_n(yi|xi, \beta_c) \tag{18}$$

Here $c \in \{1, 2, ..., C\}$. C is the total number of clusters. Lastly, $Pr_n(yi|xi, \beta_c)$ is equal to Equation 1.

For visualisation purposes Figure 2 shows the simulation results for 2 dimensions. Note that these figures do not display the true data sets we use but are solely an example of what type of data we generate. We note that the larger the chosen probability of a certain distribution is, the more data points will have that distribution. In this example 0.7 is the largest probability and 0.1 is the smallest probability. In Figure 2 we see that the black data points belong to the largest probability and blue to the smallest.

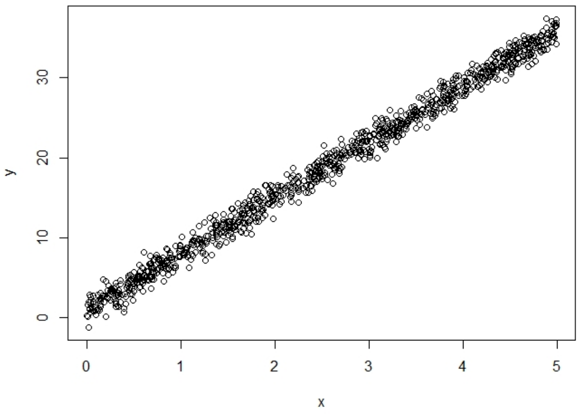

Figure 1: Scatterplot of the linear regression model simulation.

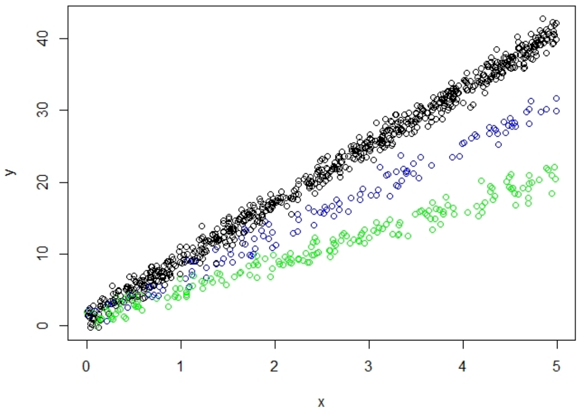

Figure 2: Scatterplot of finite mixture model simulation.

For our true simulation of the data we set each σ_c equal to 5. The coefficients for each cluster are displayed in Table 2. The chosen probabilities can be seen in Table 3. Note that every cluster has its own vector of coefficients β_c. The variables for $\mathbf{x}$ once again come from a normal distribution distribution with a mean of 10 and a variance 4. We select this second simulated set to be the base data set that does contain unobserved heterogeneity.

cluster	**1**	**2**	**3**
β_0	1	1	1
β_1	4	6	8
β_2	2	2	2
β_3	3	2	4

Table 2: The coefficients of each cluster used for simulation of soft-clustered data.

cluster	**1**	**2**	**3**
probability	0.7	0.2	0.1

Table 3: Probabilities of each cluster used for simulation of soft-clustered data

4.1.2 Adding outliers

We use the data set that includes unobserved heterogeneity create to a contaminated data set that contains outliers. There are two outliers that are possible: Outliers in the dependent variable y_i and outliers in the explanatory variables $\mathbf{x}_i$. We separately include both to see what the effect of the different outliers are on the outcomes of the estimation and the performance of the model.

Outliers in dependent variable To be able to include outliers in the dependent variable we first look at Figure 2. Here we see that given a cluster, the model becomes a linear regresssion model, where dependent variable y_i follows a normal distribution according to Equation 17. For the normal distribution we know how to simulate outliers. We simply add values of y_i that lie far in the tails of the distribution. The general rule of thumb for the normal distribution is to use values that lie more than three standard deviations from the mean (Patel & Read, 1996). In this way we assure that this value cannot be seen as a regular data point belonging to this normal distribution. Once this value is chosen, we replace an original data point with an outlier. This happens with probability γ. This means that the higher the value of γ is going to be, the more outliers are added to the contaminated data set. The problem in our case is that we do not have a single linear relation, but three. Nevertheless if we select the steepest linear relation to be the one we use to create outliers, the chosen value must be an outlier for the other linear relations as well. To see why this works, Figure 3 displays an example of a two dimensional simulation. We see that in this finite mixture simulation, the majority of the data looks similar to Figure 2. However we see that the outliers form a separate set, displayed in red. In this simulated example we choose the value of the outliers to be 50. These outliers lie the closest to the steepest subset. For this subset they are regarded as outliers, which means they must be regarded as outliers for the other two subsets as well.

To be more precice, for the outliers in the data we use for our research we choose the value of the outliers to be more than three standard deviations away from the largest simulated value of y_i. By doing so, we ensure that for any smaller values of the dependent variable, this rule of thumb also holds. In our case this value equals 290. Moreover, we select γ to be 0.05. This means that five percent of the original data is replaced by an outlier.

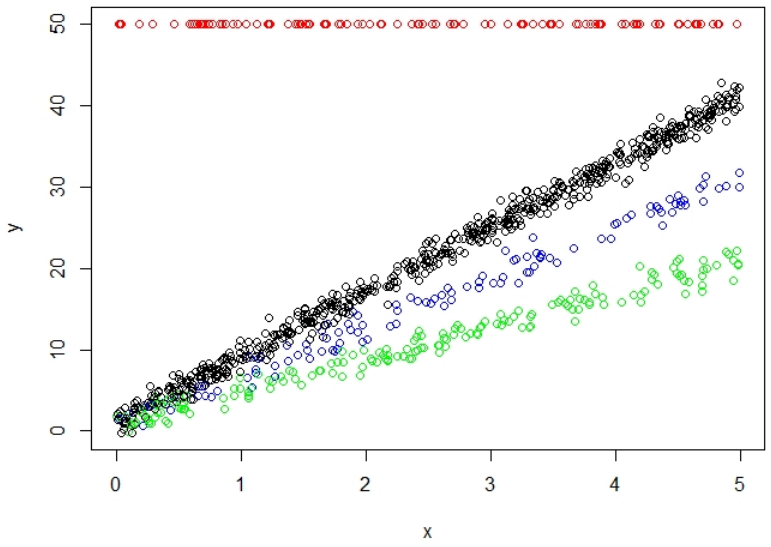

Figure 3: Scatterplot of the finite mixture model simulation including outliers in the dependent variable.

Outliers in explanatory variables To include outliers in the explanatory variables we alter all variables in $\mathbf{x}_i$. Similar to the outliers in the dependent variable, this happens with probability γ. If a data point is selected, one replaces all the values of the corresponding explanatory values with a constant value that lies more than three times the standard deviation away from the chosen mean for each of the explanatory variables. In our case, we select this value to be 20. γ again equals 0.05. The reason this replacement creates an outlier is due to the fact that the value of the dependent variable y, relatively to the new values of the explanatory variables, does not alter and stays small. Figure 6 shows an example the scatter of a two-dimensional simulation. Note that this scatterplot does not correspond to the used data, but is merely an example. The red data points correspond to the outliers.

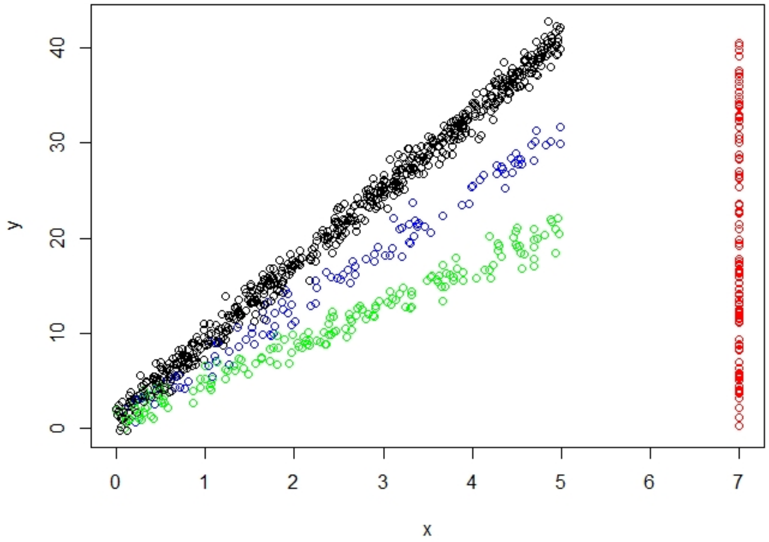

Figure 4: Scatterplot of the finite mixture model simulation including outliers in the explanatory variable.

4.1.3 Results

In this section we go through the main results of for the different data generating processes. The computer we used for the computation of these results is a HP ENVY x360 Convertible 15-bp0xx. Table 4 displays the system details of this computer.

Information type	**Details**
Processor:	Intel®, Core™, i7-7500U CPU @ 2.70 GHz 2.90 GHz
System type:	64-bit Operating System, x64-based processor
Software:	Windows 10
Memory:	8 GB

Table 4: Computer information.

Large sample size Table 5 shows us the results for the mean MSE values over ten simulations for the different data generating processes and estimators for a large sample size. The first thing we notice is that for both estimators this mean increases as the data generating process includes more complexity in terms of unobserved heterogeneity and outliers. For example, looking at the estimates for the maximum likelihood estimator $\hat{\mu}_{ML}$, the lowest value belongs to the linear data, with a value of 25.100. The highest value on the other hand, belongs to the data generating process with outliers in the explanatory variables, with a value of 1034.405. For the MOM estimator a similar reasoning holds. This also means that the mean MSE belonging to the DGP with outliers in the dependent variable is lower than that of the mean MSE belonging to the DGP with outliers in the explanatory variables. This suggests that on average both estimators seem to be more robust against outliers in the dependent variable, which is what was expected theoretically (Lecue & Lerasle, 2020). The second aspect we see in Table 5 is the difference between the two estimates. We note that the average MSE is smaller for the MOM estimator for every data generating process. However, the difference between mean MSE values of the two estimators become smaller as the data generating process becomes more complex. Take for example the discrete data generating process. For this DGP, the difference between the mean MSE values equals 1.192. For the data generating process with outliers in the explanatory variables, this difference equals 204.479. Once again, a similar analysis for the DGP with outliers in the dependent variable In other words, if there are no outliers in the data, the average performance of both estimators are closer together. When outliers are added to the data, we see that on average, the MOM estimator outperforms the maximum likelihood estimator.

	$\hat{\mu}_{ML}$	$\hat{\mu}_{MOM}$
linear	25.100	25.387
discrete	504.474	505.666
outlier Y	902.046	757.741
outlier X	1034.405	829.926

Table 5: Estimations for the mean MSE over ten simulations. N = 3000.

Table 6 visualises the standard errors for the large sample size. Here we see a difference between the data generating processes with and without outliers. For both estimators, the standard errors are smaller when no outliers are included. This suggests that over ten simulations, the performance is more stable for data that does not have outliers. Besides, there is a difference between the data generating process with outliers in the dependent variable and the explanatory variables. For both estimators the standard errors are larger for the DGP with outliers in the dependent variable, with values of 232.573 and 203.101 compared to 151.836 and 56.906 for the DGP with outliers in the explanatory variables. Lastly, we note that when comparing between the two estimators, all the standard errors are smaller for the MOM estimator. This suggests that over ten simulations, the performance of this estimator is more stable than that of the maximum likelihood estimator.

	$\hat{\sigma}_{ML}$	$\hat{\sigma}_{MOM}$
linear	0.937	0.928
discrete	20.652	10.858
outlier Y	232.573	203.101
outlier X	151.836	56.906

Table 6: Estimations for the standard deviation of the MSE over ten simulations. N = 3000.

Lastly, we look at Table 7. This Table visualises the performance of both estimators for all the data generating processes per simulation. The count in this Table indicates how many times the MOM estimator outperformed the maximum likelihood estimator out of ten simulations. We see that for the linear data generatig process this happens one time. For the discrete DGP this is six times, for the DGP with outliers in the dependent variable this number is five and for the outliers in the explanatory variables the MOM estimator outperforms the maximum likelihood estimator nine times. These results can be explained with Table 5 and Table 8. For example, if one looks at the results of the DGP with outliers in the explanatory variables we note that on average, the MOM outperforms the maximum likelihood estimator, while being more stable, with a lower standard error. Even if we would subtract the standard error from the mean MSE, one would not obtain the mean MSE belonging to the MOM estimator. There seems to be little overlap in terms of MSE over the ten simulations. This indicates that the MOM estimator has a better performance. This can also be seen in Table 7. Similar reasoning can be used for the other results in this Table.

DGP	count
linear	1
discrete	6
outlier Y	5
outlier X	9

Table 7: Number of times the MOM estimator outperforms the ML estimator. N = 3000.

Small sample size Table 8 shows the mean MSE values for the different data generating processes and estimators, but this time for a small sample size. Here again we see that the values of the mean MSE increase as the data generating processes get more complex. Nevertheless, for the data generating processes with no outliers we see that on average the performance has decreased for both estimators compared to the large sample size. For the data with outliers this is only the case for the data generating process with outliers in the explanatory variables. Moreover, we note that the MOM estimator outperforms the maximum likelihood estimator for every data generating process but the ones including outliers. For a small sample size the MOM has difficulty with any type of outliers, and therefore has a higher average MSE for the corresponding data generating processes.

	$\hat{\mu}_{ML}$	$\hat{\mu}_{MOM}$
linear	26.882	25.185
discrete	492.230	492.015
outlier Y	666.397	717.837
outlier X	1090.239	1173.653

Table 8: Estimations for the mean MSE over ten simulations. N = 300.

Table 9 visualizes the corresponding standard errors over ten simulations for the different estimators and data generating processes. Here too we see that as the data generating process gets more complex in terms of unobserved heterogeneity and outliers, the standard errors increase. For example, for the linear data generating process the standard error equals 4.390 for the MOM estimator. For the data generating process with outliers in the explanatory variables estimate equals 312.296. If we compare Table 9 with Table 6, we note that for the large sample the estimated standard errors are smaller for every data generating process. Lastly, we note that for a small sample size the difference between standard errors of the

MOM and maximum likelihood estimator are smaller. An example where this is the most obvious is for the DGP with outliers in the explanatory variables. The difference between the standard error is 94.930 for the large sample size, whereas for the small sample size this difference is 21.143. This increase is a result from a larger increase for the standard error of the MOM estimator. For the other data generating processes a similar analysis holds. We can say that the outcome of Table 8 are therefore not reliable because of the large standard errors.

	$\hat{\sigma}_{ML}$	$\hat{\sigma}_{MOM}$
linear	4.427	4.390
discrete	54.773	57.722
outlier Y	389.108	436.855
outlier X	333.439	312.296

Table 9: Estimations for the standard deviation of the MSE over ten simulations. N = 300.

Lastly, we look at Table 10. Due to the larger standard errors, we see that for none of the data generating processes there is an estimation that truly outperforms the other estimator. The highest number of times corresponds to the linear DGP. Here the MOM estimator outperforms the maximum likelihood estimator seven out of ten times. Similar to Table 8, this does not tell us anything. Due to the large standard errors, this Table merely shows results that have to do with chance. Especially over only ten simulations, nothing can be said about these results, apart from the fact that there is no clear estimator that performs better for this sample size.

DGP	count
linear	7
discrete	6
outlier Y	4
outlier X	5

Table 10: number of times the MOM estimator outperforms the ML estimator. N = 300.

4.2 Simulation design for empirical application and results

4.2.1 Empirical data

To illustrate our newly proposed method we train the estimators using empirical data that contains unobserved heterogeneity. In our case we use the blue crab data of Campbell & Mahon (1976). This is data contains unobserved heterogeneity because it both has female and male crab observations. It consists of 200 data points, two categorical and four continuous variables. Similar to Yao et al. (2014), we focus on variates two and three of the data set which are the rear width of the crab (RW) and the length along the midline (CL). Figure 5 shows a scatterplot of these two variates. Using the categorical variable that tells us whether a data point corresponds to a male or female crab, we split up the data sets into two groups. In the figure this results in data points labelled with red, corresponding to the female crab and green, corresponding to a male crab. We see that for both groups there is a different optimal linear line possible, which means there is unobserved heterogeneity. There might also be outliers in this data set. However, Figure 5 shows they are more difficult to distinguish from the correct data, compared to for example Figure 3.

4.2.2 Simulation design

In order to get some more insight on the results of the empirical application, we also simulate data using a data generating process that possesses properties similar to this empirical process. This means that the same number of explanatory variables are going to be used, the number of data points is set to exactly the same number as the empirical data set and we incorporate the exact same amount of clusters. In our case we therefore use one explanatory variable, the number of data points equals 200 and the number of clusters is equal to two. Figure 6 shows the scatterplot of the first simulation. Note than when comparing this Figure to Figure 5, the similarities are visible. The only difference is the value of of the different coefficients and the variance belonging to each of the clusters. This is due to the fact that we do not know the true values belonging the the empirical data set.

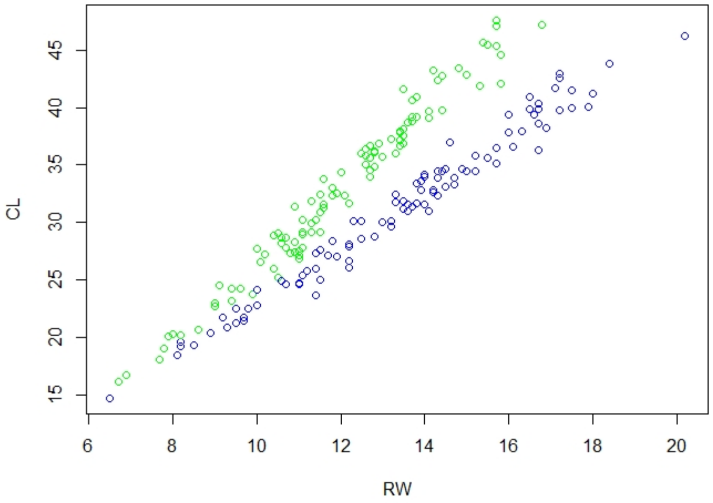

Figure 5: Scatter plot of the second and third variates of the Blue Crab data set with their gender of origin labelled.

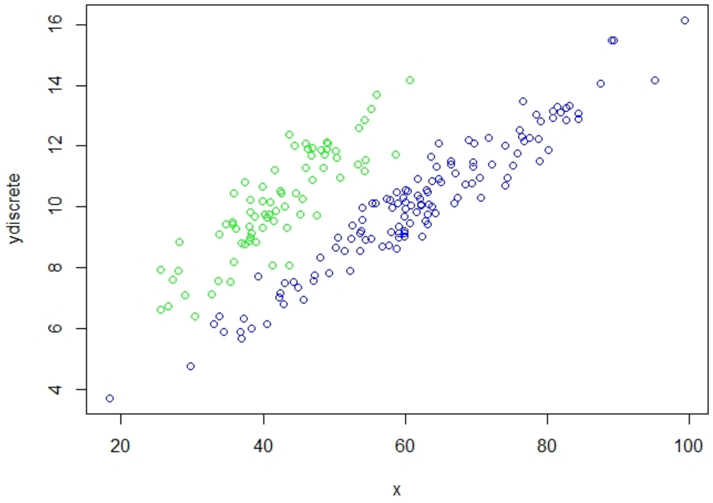

Figure 6: Scatter plot of the simulated data mimicking the Blue Crab data set with their gender of origin labelled.

4.2.3 Results

If we look at the results of the simulated data in Table 11 we note two things. First of all we see that on average, over ten different simulations, the MOM estimator outperforms the traditional maximum likelihood estimator with mean MSE values of 114.246 and 122.088 for the estimators, respectively. Moreover, we see that the estimation of the standard deviation for the MOM estimator is the lowest too, with a value of 10.688 compared to an estimated value of 12.509 for the maximum likelihood estimator. In other words, for ten different simulations the values of the MSE fluctuate less than the MSE of the maximum likelihood estimator and has a lower average. We can say the results of the MOM estimator are more stable in the sense that for different data sets with similar characteristics, the performance of the latter seems to be similar. For the maximum likelihood estimator this performance fluctuates more. All in all, these two measures also explain why we see that the MOM estimator outperforms the maximum likelihood estimator seven out of ten times, but not always. Lastly, for both estimators we see that the number of clusters being chosen correctly equals three out of ten. All in all, looking at these results we would expect that the MOM estimator will outperform the maximum likelihood estimator for this type of data.

	ML	MOM
$\hat{\mu}$	122.088369952486	114.245618938763
$\hat{\sigma}$	12.5094269851191	10.6885742238506
count	3	3

Table 11: Estimations of the mean and standard deviation of the MSE over ten simulations for both the ML and MOM estimator. N = 200.

Results empirical data Table 12 shows the final results of the empirical data set. We note three things when comparing it to the simulation results. First of all, the magnitude of the mean MSE is much higher than the mean MSE values corresponding to both estimators for the empirical data set. Secondly, whereas the simulations only correctly choose the number of clusters three out of ten times, the MOM estimator is able to correctly choose the number of clusters for this empirical application. Lastly, we see that even though the simulations indicate that the MOM estimator outperforms the maximum likelihood estimator, this is not the case. The maximum likelihood estimator has a MSE value of 10.531, whereas the MOM estimator has a MSE value of 10.807. So, whereas the MOM estimator does robustify the finite mixture model for the empirical simulations, it does not for the empirical application.

	ML	MOM
MSE	10.531	10.807
K	3	2

Table 12: MSE and optimal number of cluster for the ML and MOM estimator using the Blue Crab data set. N = 200.

5 Conclusion

Throughout time models have been created to include unobserved heterogeneity. The most commonly used example of such a model is the finite mixture model (Muthen & Shedden, 1999). The problem with this model, however, is that it vulnerable for outliers in data. A recently developed method that takes care of these outliers is the Median-Of-Means (MOM) method. In our research we combine the finite mixture model and the Median-Of-Means method and answer the following research question:

- *“How can one use the Median-Of-Means to effectively combine unobserved heterogeneity and robustness within a linear regression framework?"*

For our research we used four data generating processes. The first process generates linear data with variance that is set beforehand. The second generating process than uses soft-clustering, which means that a data point is simulated from a specific distribution with probability π_c, where $c \in \{1, 2, .., C\}$. C is the total number of different distributions. The other two processes are soft-clustered, but also includes outliers in either the dependent variable or the explanatory variables.

The finite mixture model is a model based on this soft-clustering. This means that the model sets the number of clusters, each with its own corresponding distribution. In our case, this means that every cluster corresponds to a normal distribution with its own mean equalling a linear relation of the explanatory variables and the corresponding coefficients, and a variance that equals the variance of the error term in a standard linear model. Both the coefficients and the variance may differ per cluster and are calculated using the so-called EM-algorithm. This algorithm has a Estimation and Maximisation step. In the estimation step the expectation to be optimised is set up and in the Maximisation step one optimises this expected value.

This Maximisation step is where we combine the MOM estimator with the finite mixture model. By doing so, we make a robust model that includes unobserved heterogeneity. The MOM estimator splits up the data, calculates the arithmetic means for every subset and then takes the median of all these arithmetic means. In this way it makes sure that outliers do not affect the chosen median value, as all arithmetic means influenced by outliers will have an odd value. These will therefore never be selected to be the median. This is what we then use to replace expected value in the Maximisation step. This results in a new problem, which is then optimised to obtain the estimations for the coefficients and variance.

Our results show that for larger data, the MOM estimator indeed robustifies the finite mixture model, but also tends to work for data that does not contain any outliers: For the data generating processes with no outliers both the MOM estimator and the maximum likelihood estimator perform similarly, whereas for the contaminated data generating process the MOM estimator has lower mean MSE value and a smaller variance compared to the maximum likelihood estimator. Moreover, we see that out of the ten simulations, the majority of the times the MSE is smaller for the MOM estimator than for the maximum likelihood estimator, for both data generating processes with outliers.

For the smaller data sets, we are not able to make the same conclusion, because of the large variance over the ten simulations. Even though for some generating processes the MOM estimator outperforms the maximum likelihood in terms of mean MSE, its variance does not make this value reliable. We conclude from that the MOM estimator does not robustify the finite mixture model for small samples with certainty. This conclusion is confirmed using a small empirical data set. For this dat set, containing 200 data points, the ML estimator outperforms the MOM estimator.

6 Discussion

Throughout our research, we have come across some limitations. First of all, the EM algorithm turned out to be very time-consuming. For a data set with 3000 data points it took over half a day to estimate the results. This meant that specific methods of training the model, such as K-fold cross validation, were not possible. This why we had to train the model using a hold-out set. In future research one could make use of K-fold cross validation instead of a hold-out sample and see what this does to the main results.

Secondly, we were only able to do ten simulations due to these time restrictions. Because of this small number, some of the estimates over these ten simulations have become unreliable (Hackshaw, 2008). This also meant that some second degree measures such as the standard error of the MSE were so unreliable that it is debatable if it is useful to even estimate them. In future research more simulations could be done, to see what the asymptotic values of the performance measures are and to be able to calculate second degree estimators with certainty.

In addition, one could compare our proposed method to other robust methods such as the use of the t-distribution instead of the normal distribution (G. McLachlan & Peel, 2000). By doing so, we do not only see how our estimator performs compared to maximum likelihood estimation, which results can be predicted theoretically, but also see how well it performs compared to other robust methods. In this way we get a better view on the estimator's ability to robustify the finite mixture model.

Thirdly, the programming language struggled with the optimisation of both the finite mixture model and the MOM estimator. For many different starting values, optimisation was not possible as an error was given by Rstudio (Rstudio, 2020). This struggle lead to two limitations. First of all we had to run the code many times in order to get results. Due to this struggle the optimisation became even more time consuming. Besides, there were some starting values of the coefficients that could not be used. In this way we cannot be sure whether or not these starting values would result in an optimal solution that is better than our main results, as we could not get an estimation for the latter. In future research one could try to use even more different starting values and see whether it improves the estimations.

This optimisation struggle could have been improved by making the program more efficient. Unfortunately, we did not have the time to do that. The result is that especially for the MOM estimator, where written functions started to play a role, the running tended to go rather slowly, compared to maximum likelihood. If one is able to solve this problem, other problems that have been mentioned might not occur anymore. Take for example the limit on the amount simulations. If the program runs more efficiently, there is more time left to run more than ten simulations. In future research one could look at this.

Furthermore, the MOM-plugin estimator only robustly estimates the coefficients belonging to the explanatory variables. The estimation of the standard deviation σ however, is calculated using Equation 8. This means that this estimation is still vulnerable for outliers and therefore is not robust. In future research one could look at a way to robustify the estimation of the standard deviation in order to make the results more complete.

We also concluded throughout our research that for a larger number of outliers in the dependent variable y, the MOM estimator struggles with the robustification of the finite mixture model. The MOM estimator should be able to counter this problem, however (Lecue & Lerasle, 2020). So, why does it no do what it is supposed to do? The reason is the selected number of subsets for this estimator. This number needs to be at least the double of the amount of outliers in the data (Lecue & Lerasle, 2020). By doing so, one makes sure that the selected median will not be affected by outliers. In our case we used data generating processes that included outliers with a probability of five percent. For 3000 data points, this meant that there would be approximately 150 outliers in our data. This also meant that we had to use at least 300 subsets for the MOM estimator. For each of these subsets the mean would then be calculated. The problem here, however, is that we only have 3000 data points. Dividing this number into 300 different subsets results in 10 data points per subset. Of course, this number is not enough to get a good estimation for the mean. We believe this is the reason the rate of convergence is so low and this is why we believe that the model finds it difficult to cope with this amount of outliers. For the data generating process with outliers in the explanatory variables a similar explanation holds. Nevertheless, in this situation our proposed estimator still outperforms the maximum likelihood estimator under an outlier percentage of five percent. The reason is that this specific type of outliers is more difficult to manage (Verardi & Croux, 2009). The maximum likelihood estimator will therefore be affected even more, whereas the MOM estimator is not. Nevertheless, in future research one could use a smaller percentage of outliers, to see the true performance of our proposed estimator.

Moreover, we were not able to optimise the amount of subsets that we used for the MOM estimator. In our research we set this number equal to a fixed amount. In reality, however, it does not have to be the case that this number fits the data well. For data with no outliers this number can be set to a much lower number. For data with more outliers, the chosen number might not be enough. This can be solved by looking at this number of subsets as a hyperparameter, and tune the latter along with the number of clusters for the finite mixture model. This is necessary for your model to perform optimally (Weerts et al., 2020).

Neverthless, it was not only the optimisation we struggled with but also the type of optimisation. We used the minimisation of a problem, similar to papers by Koltchinskii (2011), Van De Geer (2000) and Vapnnik (2000). Nevertheless, minimisation of a function where the MOM-estimator is plugged in results in a slow minimax rate. In order to improve the minimax rate one could use minimaxisation algorithms proposed in papers by Audibert & Catoni (2011),Baraud & Birge (2016) and Baraud et al. (2017). This method ensures that apart from minimisation of the risk function, the resulting estimators' highest possible risk is the lowest among all other possible estimators. This extra restriction should improve the results of our research and increase the rate of convergence (Lecue & Lerasle, 2020). Therefore one could try and implement this in future work.

Lastly, even though the empirical application was used in a different paper that discusses a similar topic, we feel like we could have chosen a better data set. Looking at Figure 5, there all multiple things that are convenient. First of all, the number of variables is limited, which decreases computation time. Secondly, there are clearly two different subsets, which means there is unobserved heterogeneity. The problems with this data set , however, are as follows. First of all, the data set only has 200 data points. For the most estimators, this is not enough to work well (Heij et al., 2004). This especially counts for the MOM estimator, as we have concluded. This is one of the reasons the MOM estimator does not perform well for this empirical application. The second problem is the lack of outliers. Looking at the final results we note that the MSE values are very low compared to the simulations. This is due to the fact that the chosen variance for the simulation is larger, which is why some data points can be seen as outliers. However, in the empirical application, All data points seem to lie around either one of two regression lines corresponding to finite mixture model. Due to this lack of outliers it does not really test whether or not the MOM estimator robustifies the finite mixture model. It merely shows whether or not the MOM estimator works if there are no outliers. In future research one might look for another empirical application that has a larger sample size, and is more likely to contain outliers.

References

Alon, N., Matias, Y. & Szegedy, M. (1999). the space complexity of approximating the frequency moments. *Twenty-eighth Annual ACM Symposium on the Theory of Computing*, *1*, 137-147.

Audibert, J.-Y. & Catoni, O. (2011). Robust linear least squares regression. *Ann. Stastic*, *5*, 2766-2794.

Baraud, Y. & Birge, L. (2016). Rho-estimators for shape restricted density estimation. *Stochastic, Process. Appl.*, *12*, 3888–3912.

Baraud, Y., Birge, L. & Sart, M. (2017). A new method for estimation and model selection: ρ-estimation. *Invent. Math.*, *2*, 425.517.

Bradley, J., Kyrola, A., Bickson, D. & Guestrin, C. (2011). Parallel coordinate descent for l1-regularized loss minimization. *International Conference on Machine Learning*.

Campbell & Mahon. (1976). Blue crab data on the genus leptograpsus.

Durling, D., Cross, N. & Johnson, J. (1996). Personality and learning preferences of students in design and design-related disciplines. *International Statistical Review*.

Hackshaw, A. (2008). Small studies: strengths and limitations.

Halfens, R. & Meijers, J. (2012). Back to basics: An introduction to statistics. *journal of wound care*.

Hampel, F. R. (1971). A general qualitative definition of robustness. *The Annals of Mathematical Statistics*, *42*(6), 1887-1896.

Heij, C., de Boer, P., Franses, P. H., Kloek, T. & van Dijk, H. K. (2004). *Econometric methods with applications in business and economics.* Oxford University Press.

Kiers, H. (1997). Weighted least squares fitting using ordinary least squares algorithms. *Psychometrika*.

Koltchinskii, V. (2011). Oracle inequalities in empirical risk minimization and sparse recovery problems. *Lecture Notes in Mathematics*, *2033*.

Lecue, G. & Lerasle, M. (n.d.). Learning from mom's principle: Le cam's approach. *Technical report*.

Lecue, G. & Lerasle, M. (2020). Robust machine learning by median-of-means : theory and practice. *Annals of Statistics*.

Lin, T., Lee, J. & Hsieh, W. (2007). Robust mixture modeling using the skew t distribution. *Statistics and computing*.

Lugosi, G. & Mendelson, S. (n.d.). Risk minimization by median-of-means tournaments.

McLachlan, G. & Peel, D. (2000). Robust modelling using the t-distribtion. *Statistics and computing*.

McLachlan, G. J. & Peel, D. (2004). *Finite mixture models.* John Wiley & Sons.

Melnykov, V., Maitra, R. et al. (2010). Finite mixture models and model-based clustering. *Statistics Surveys*, *4*, 80–116.

Mili, L., Phaniraj, V. & Rousseeuw, P. (1991). Least median of squares estimation in power systems. *IEEE Transactions of Power Systems*.

Muthen, B. & Shedden, K. (1999). Finite mixture modeling with mixture outcomes using the em algorithm. *Biomatrics*, *55*, 463–469.

Nguyen, T. & Wu, Q. (2011). Robust student's-t mixture model with spatial constraints and its application in medical image segmentation. *IEEE Transactions on Medical Imaging*.

Patel, J. & Read, C. (1996). Handbook of the normal distribution. , *150*.

Pazis, J., Ronald, P. & How, J., P. (2016). Improving pac exploration using the median of means. *Advances in neural information processing systems*.

Rousseeuw, J., Peter. (1996). Least median of squares regression. *Journal of the American Statistical Association*, *79*(388), 871–880.

Rstudio. (2020). *Rstudio version 4.0.3.* Boston, Massachusetts: The R Foundation for Statistical Computing.

Ruder, S. (2016). An overview of gradient descent optimization algorithms.

Susanti, Y. & Pratiwi, H. (2014). M estimation, s estimation, and mm estimation in robust regression. *International Journal of Pure and Applied Mathematics*, *91*(3).

Tao, T. & Zhai, C. (2006). Regularized estimation of mixture models for robust pseudo-relevance feedback.

Van De Geer, A., Sarah. (2000). Applications of empirical process theory. *Cambridge Series in Statistical and Probabilistic Mathematics*, *6*.

Vapnnik, N., Vladimir. (2000). Adaptive and learning systems for signal processing. *The nature of statistical learning theory*, *2*.

Veldhuizen, D., Borges Soares, J. et al. (n.d.). Predicting the regional potential for restaurants using models with unobserved heterogeneity. , *4*.

Verardi, V. & Croux, C. (2009). Robust regression in stata. *The Stata Journal*, *4*, 439–453.

Weerts, J., Hilde, Müller, C., Andreas & Vanschoren, J. (2020). Importance of tuning hyperparameters of machine learning algorithms.

Yao, W., Y., Wei & Yu, C. (2014). Robust mixture regression using the t-distribution. *Computational Statistics and Data Analysis*, *71*, 116-127.