Categorical based feature modeling on a zero inflated performance measure

V.S. Berghuis

Technische Universiteit Delft

TU Delft

Categorical based feature modeling on a zero inflated performance measure

V.S. Berghuis

June 30, 2017

in partial fulfillment of the requirements for the degree of

Master of Science in Applied Mathematics

at the Delft University of Technology,
to be defended publicly on Thursday July 6, 2017 at 10.00 AM.

Supervisor:	Dr. J. Söhl	TU Delft
Thesis committee:	Dr. ir. B. Van Hoof	ASML
	Prof. dr. ir. G. Jongbloed	TU Delft
	Dr. D. Kurowicka	TU Delft
	Dr. M. Voncken	ASML

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

TU Delft Delft University of Technology

ASML

Preface

In the last nine months, I worked on this graduation project carried out as part of the Master Applied Mathematics at the Delft University of Technology.

Within this master, I have chosen to go for the specialization of Probability, Risk and Statistics because of my interests in data analysis. The main reason why I am interested in data analysis is because of the many concrete applications of statistics on real world problems. Moreover, with my background in technology, policy and management I hope to apply the detailed, substantiate statistical analysis on high-level decision making in my further career.

When choosing my graduation project, different subjects came along. However, when the data-project of ASML passed by, I knew I had to apply for it. Even though I had to move from Delft to Eindhoven, I had to take the challenge. It is a unique chance to take a look in such a complex high-tech company, belonging to the top of the Dutch' exports. As the initial research objective was quite broad, I had the chance to scope it in the direction I preferred.

I would not have been able to achieve this study without the help of some people.
Jakob Söhl, thank you for advising and guiding me throughout the process. You helped me in doing the analysis just a bit further and stood always open for new suggestions. Thank you for making always time if I had questions.
Special thanks go to my supervisor at ASML, Bram van Hoof. You were glad to help and answer any of my questions and took the time to guide me. Both on professional level as on the content you had some very useful tips which I will definitely use in my further career.
Of course, I would also like to thank Maarten Voncken for the monthly meeting. You guided me on a more abstract level and was interested in how the project was going and if I were doing okay.
Geurt, thank you for your enthusiasm and small chats on the hallway. Although you might not have seen my project that often I am sure you kept an eye on it and would have been willing to spend time if I needed to.
Finally, I would like to thank my family and my friends. Rikkert thank you for your chats, jokes, discovering Eindhoven and keeping me motivated throughout the process. Anton and Dennis I would like to thank you for the funny (and too long) coffee breaks, being happy because of sparkling water and for the jokes of Dennis' pronunciation of the wafers.

V.S. Berghuis
Delft, June 30, 2017

Abstract

ASML produces TwinScan NXT machines that are used for the production of microchips. The machines ensure that an accurate pattern of DUV-light passes a lens and that it is projected as accurate as possible on the wafer. To ensure that the focal point of the converged DUV-light falls exactly onto the wafer, the leveling functionality is of great importance. That is, placing the wafer in the correct depth of focus by rotating the wafer and moving the wafer up or down during exposure.

In order to meet the imaging requirements, the performance is investigated by analyzing errors of the machines at customers' site, considering one-year data. The most important errors are A, B, C and D. To reduce the total unscheduled down (USD) time of those errors, we should focus on reducing the mean USD time for errors A and C; and focus on reducing the frequency for errors B andD where these last two errors are likely to be solved together.

Different nominal customer-related variables are considered as possible causes of USD times such as location, system type or type of sensors. After applying hierarchical clustering and multidimensional scaling, the variable set is reduced. This set is used to model the USD time of one error: B. Significant differences in USD times are found, showed by the robust and distribution free rank tests: Wilcoxon and Kruskal-Wallis test.

To discover interesting patterns among variables, regression models are applied. The linear regression model and generalized linear model not seem to be the right model to the data. The zero adjusted exponential model seems to be the correct model and show that AG type, location and FSM flex package are the most important explanatory variables. This directs to a potential root cause where ASML is working further upon.

Contents

1

Introduction to ASML

ASML is the world's leading supplier and manufacturer of lithography using machines. The companies headquarter is stationed in Veldhoven, the Netherlands. ASML was initially a spin-off from Philips, a world known electronics company. The machines ASML produces are used for the production of micro chips, or integrated circuits, for all kinds of electronic devices. The machine type that is evaluated in this research is a frequently used machine by the customer such as Samsung or Intel. This machine is called the TWINSCAN NXT. The TWINSCAN NXT ensures an accurate pattern that is projected on the *wafer*, a rounded thin slice made of silicon.

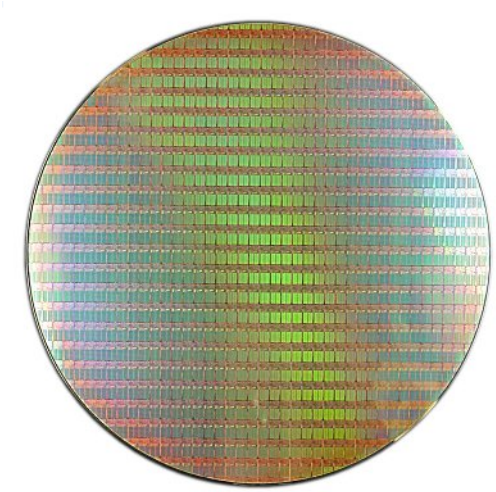


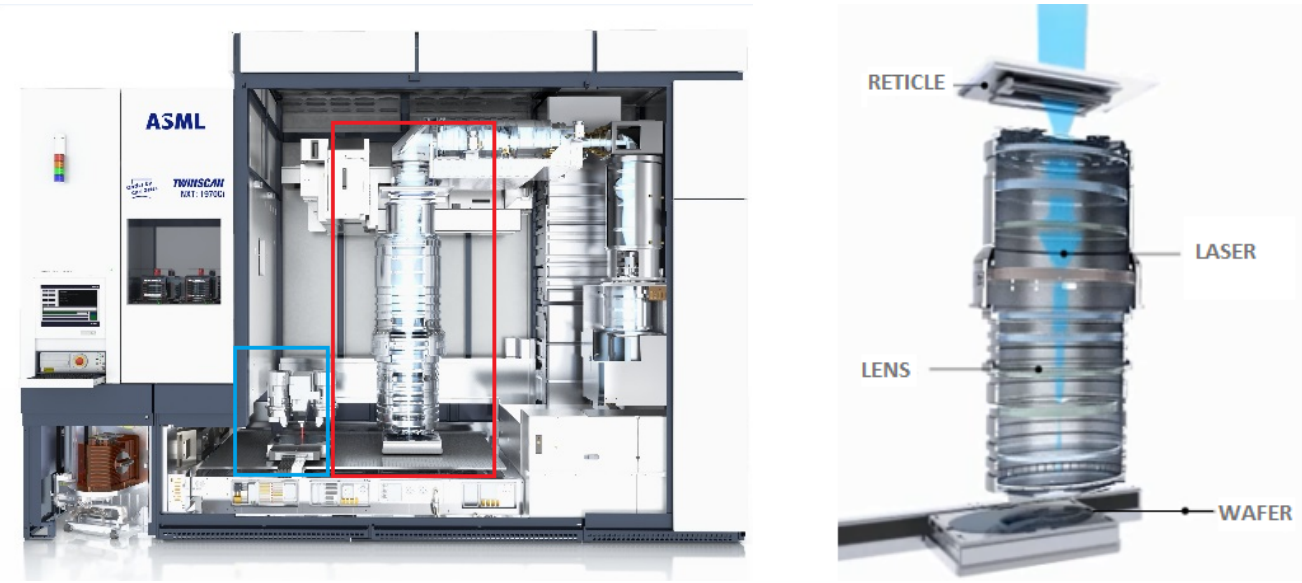
Figure 1.1: Top view of a wafer

Out of the wafer, hundreds of *dies* (small squares) can be obtained. These dies become individual integrated circuits. The pattern on the wafer determines eventually the capacity of the chips such as the type of memory capability or type of processor. One of the challenges of ASML's customers is to increase the capacity of the dies but also to make the dies as small as possible. To accomplish this, the pattern or so called *image* needs to be as small as possible and as accurate as possible on a scale of nanometers. To ensure accuracy, *leveling* is important. To understand the function of leveling, the technique of the TWINSCAN NXT machine needs to be explained. Thereafter the task of leveling is discussed.

1.1. Task of leveling within the TWINSCAN NXT

The name TWINSCAN is derived from the fact that the machine uses a dual-stage design: measuring one wafer on the *measure side* while imaging another on the *expose side*. On the measure side, all kind of characteristics are measured, such as topography, the position and contamination of the wafer. This

information is used as input on the expose side, where the wafer is exposed to the laser, which uses Deep Ultra Violet light (DUV light). On the expose side light shines through a mask, called *reticle*, such that at some places the light is blocked and the rest of the light passes through. In this way a pattern of light and shadow is obtained. This pattern goes through the lens and falls onto the wafer such that the pattern is four times reduced in size. In figure 1.2a a TWINSCAN NXT is pictured. In the red box the light goes from reticle through lens onto the wafer, which is more clear in figure 1.2b. The blue box indicates the location of the measure side.



(a) Measure side (blue) and expose side (red)

(b) Light passing reticle, trough lens onto wafer

Figure 1.2: A TWINSCAN NXT machine

When the DUV light passes through the lens, the light beams converge and cross each other in exactly one point, called the focal point. After this point, the light beams diverge in the opposite direction. See figure 1.3.

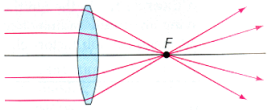


Figure 1.3: Converging of light after passing the lens

When projecting a pattern on the wafer, the wafer should be placed exactly in the focal point. In this way the sharpest image on the wafer is obtained. If the image of the pattern is not sharp, the wafer is useless. When looking at a wafer, it looks flat from a human eye but when zooming in on nanometer scale, the wafer is not flat: it contains differences in heights in a range of a few 100 nanometers to microns. For every specific height, the wafer has to be moved up or down or rotated to minimize defocus. Therefore leveling is needed to place the wafer in the correct height and rotation such that the wafer is in the correct depth of focus during exposure.

To ensure the depth of focus, the leveling group has the core responsibility to create a wafer map. This map shows the height at each x, y coordinate of the wafer. The height is measured by the level sensor (LS). Obtaining the height measurements from the LS is not the responsibility of the leveling department, but the department transforms the raw measurements into the wafer map.

The wafer map is segmented according to the fields defined on the wafer. Afterwards, the optimal leveling trajectory, or exposure profile, is determined for each field by taking into account: continuous best focus within the exposure slit during scanning of the field and wafer stage servo performance limits.

That is, the wafer stage moves the wafer and has limits such as maximum acceleration.

Besides creating a wafer map and calculating the exposure profile the leveling functionality has other functionalities:

- Immersion/improved vertical stage align (iVSA): providing the vertical position of the wafer stage which is used for a correct alignment with respect to the reticle. This is done on the measure side.
- Extended vertical stage align (xVSA): controlling the drift and tilt between the LS and the measurement of the position of the wafer stage (SPM).
- Process dependency control: correcting the measured height by the LS for errors which are caused by the interaction between the level sensor beams and layers and patterns.
- Contamination detection: detecting contamination on the wafer and chuck, and potentially cleaning. The chuck is the part of the wafer stage where the wafer is located upon. This contamination leads to overlay and focus errors.
- Measure sequence: determining the sequence of measurement actions on the wafer before exposure.

Then the exposure set points for minimizing defocus and other relevant parameters that influence the height-measurements, such as limitations of the sensors or contamination particles, are sent to the expose side. In this way the wafer is placed in the correct height and is handled correctly.

1.2. Research description

As discussed in the chapter above, it is important to image the pattern in the available depth of focus during exposure to ensure the quality of the wafer. This is called *imaging performance*. Besides imaging performance, the customer has also other interests which are the *throughput*, i.e. the number of wafers processed per hour and the *yield*, i.e. the number of usable dies out of the wafer.
The leveling department would like to optimize these performances. One way to measure performance is by analyzing and quantifying *errors* which occur at the customer. While a wafer is processed all kind of measurements are logged. When a measurement value is below or above a specific threshold, an error can occur. This error can have different consequences: some measurements have to be redone, a wafer or a lot (i.e. set of wafers) can be rejected or the machine can go down (i.e. unscheduled down time (USD time)). These four consequences result in delay and hence in a lower throughput and a loss in yield.

To minimize these errors, insight needs to be gained first in their appearance, such as frequency or mean USD time. This also gives insight in which errors are the most important ones, which defines question one: 'Which errors have the greatest impact on the leveling-performance?' Moreover, it could be that errors occur not-frequently but when it occurs it causes a high USD time. Or it can be the other way around: the errors occur frequently but only cause a small delay. In both situations throughput can be optimised but each situation should be improved differently. These situations are handled by question two. The next question which arises is, what causes those errors and how? Is it customer dependent? Is it sensor dependent? For example, some customers may abandon the clean room protocol or may use extra sensors to improve the yield. Some customers may use extra sensors or packages. This may ensure accuracy, but it takes more time during production than when using not those extra sensors or packages are not used. However, this extra time may outweigh the number of lost wafers when not using those extra sensors. Therefore, the third question focuses on finding out which of these customer related variables could have an influence on the USD time.
With these variables it is investigated how they behave jointly and relate among each other. It could be that a particular combination of variables leads to a strong decrease of USD time while separately they do not. In the last sub question a possible causality is explained which resulted from the analysis. Summarizing, the following research goal is investigated:

Predict and decrease the USD time of leveling-errors in TwinScan NXT machines in order to increase the performance of leveling at customers' site

To give an answer to this goal the following research question is investigated: 'What are the predicted USD times of leveling-errors on TwinScan NXT given some specific customer-related variables?' This question is divided by the following sub questions:

1. Which errors have the greatest impact on the leveling-performance?
2. What is the behavior of those errors with respect to their frequency and USD time?
3. For which variables exist a significant difference in the USD time of one of the most important errors?
4. Do there exist patterns or findings which indicate to a root cause?
5. What is the root cause and what can a customer do?

1.3. Outline

First data need to be collected. The data belonging to the errors are collected where some initial analysis is done. Afterwards, the data belonging to the configurations, such as type of sensor or location, are collected. Then each configuration is assessed on having a significant impact on the USD time or not. The chapter thereafter handles sub question four where some patterns are tried to be discovered. In the last question a concrete and physical example of causality is investigated where some hypotheses are gathered and compiled. Finally, a conclusion is drawn. Herein, each question is answered and we show an interesting pattern that is discovered which points towards a potential root cause of one specific error.

2

The error logging data set

In this chapter the data corresponding to the errors are gathered and sorted. Then descriptive analysis is done to their USD times and frequencies. Afterwards an exploratory analysis is performed where their dependencies are investigated.

2.1. Data collection

Every time an error occurs on a machine, it is logged in the database called performance, monitoring & analysis (PMA).

As we can recall, an error can cause zero USD time, USD time, wafer rejection or even lot abort. With zero USD time we mean that an error occurred but the machine did not go down but still could have caused delay if some measurements had to be redone. In order to answer the first sub question 'Which errors have the greatest impact on the leveling-performance?', the consequences of each error need to be gathered. Two types of data are required: log data of the machines during production to obtain the USD time and data which show wafer rejections and lot aborts. The data set consisting of the wafer rejections and lot aborts appears not to be consistent and is therefore not considered. The data set consisting of the USD data is obtained through OBI (Oracle Business Intelligence), which is an interface of PMA. The link and path to get there can be found in appendix A.1.

In the first chapter, the chosen options in OBI are defined to collect the logged data. After collecting these data, the *leveling-errors*, i.e. errors which belong to the leveling functionality, are extracted and sorted in order to define the most important errors. In the third chapter, a short description is given of those.

2.1.1. Options chosen in OBI

The following options are chosen in OBI:

Machine types	NXT1960Bi, NXT1970Ci, NXT1980Ci
Time span	365 days with end week 201643, i.e. from 01-11-2015 to 30-10-2016
Customer sites only	Yes
Error codes	Errors which starts with specific letters that are confidential.
Accountability	All, i.e. both ASML as non-ASML
Interrupt type	All, i.e. availability, reliability, auto recovery and repetitive

Table 2.1: Options chosen to collect data from OBI, using PMA

These error codes are selected such that the data consist of at least the leveling-errors. In the later stage these particular leveling-errors are subtracted.
The errors of type 'auto recovery' or 'reliability' will not cause USD time and the errors of type 'availability' or 'reliability' do. When an error did not cause USD time, the machine did not go down but

still could have caused a delay in the process. The delay time is not known but is still interesting, since decreasing delay will increase the throughput for a machine.

2.1.2. Most important errors

The data belonging to these leveling-errors are subtracted from the OBI-data set. Such that 36 leveling-errors remain. These errors are listed with a short description in appendix A.3. In the following table these errors are quantified and sorted by he total USD time of that error, that influences the *availability* of a machine.

This table is confidential.

The following quantifications are shown: mean USD time (hrs), frequency, total USD time caused (hrs), median USD time (hrs) and 3 times standard deviation of USD time. We use 3 times standard deviation instead of standard deviation since this is an often used measure within ASML.

2.2. Descriptive analysis

To get insight in the structure of the data set and to provide some understanding, descriptive statistics are applied. Firstly, the distribution of the four errors are considered for both the number of occurrences per day and the USD time. Secondly, outliers will be addressed and handled.

2.2.1. Distribution of selected errors

Visualization of the distribution of the USD times and the frequency per day, shows the behavior of each error and its differences in a quick way. However, data were only gathered when an error occurred. No information is available when an error did not occur. To overcome this problem the data are transformed to daily data, i.e. the total number of error occurrences per day per error is counted.

Distribution of the frequency per day

The following estimations of the probability density function are obtained:

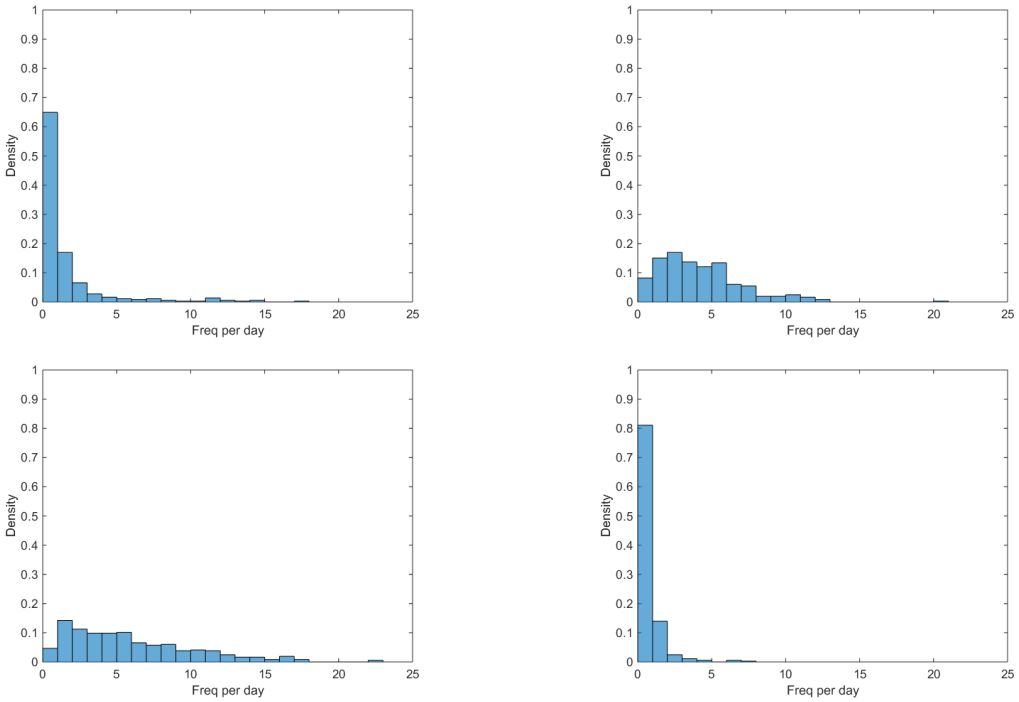


Figure 2.1: Estimated probability density function of frequency per day of the four most important errors. From left to right and top left to bottom right: A, B, D, C.

The errors A and C both have a high number of days where no error occurs: a high peak at zero and then a steep slope towards the right side. The histogram of B and D are more spread with the highest peaks at two times a day and one time a day respectively.

Probability of at least one error occurs on a day on a machine

The frequency of an error occurrence per day is a discrete random variable. To calculate the probability of having x frequencies on a day, we could use the Poisson distribution, given a constant mean λ. That is the mean number of occurrences on a day and is equal to the variance. However, applying on the data, the mean and variance differ too much for each error, indicating that the mean λ may not be constant and differ per day. Therefore, the Poisson distribution may not be the correct model. Moreover, using the GoF test, it appears that the four errors are significantly not Poisson distributed. Instead of determining the probability having x frequencies per day, we look at the probability of having at least one frequency ($x \geq 1$) per day and the probability of having zero frequencies ($x = 0$) per day. For that we use the binomial distribution where we assume that the observations are independent. This is a distribution of obtaining k successes out of a sequence of n independent yes/no experiments where each experiment has a probability p of success. Applying on the data, the sequence n is the integer number of observed days (365) where on each day at least one error can occur ($x \geq 1$) or not occur ($x = 0$). Then p is the probability that at least one error occurs on a day. In this case, success means that an error occurs.

For each error an estimation of the parameter p is calculated by maximum likelihood estimation (MLE) which is derived in appendix A.4.

$$\hat{p} = \frac{X}{n} = \frac{\sum_{i=1}^{n} x_i}{n}$$

Where X_i for $i = 1, .., n$ is a 1 or 0 indicating that error occurred or not. For this parameter a 95% confidence interval can be constructed, i.e. finding an L and U such that $P(L < \hat{p} < U) = 0.95$, using the 'Wilson method' that is possible for large n (Dekking, Kraaikamp, Lopuhaä & Meester, 2005, ch. 24, pp. 361-362). This method makes use of the fact that for large n, as a consequence of the central limit theorem, the distribution of X is approximately normal, and $\frac{X-np}{\sqrt{np(1-p)}} = \frac{\frac{X}{n}-p}{\sqrt{\frac{p(1-p)}{n}}} \sim N(0,1)$. So for large n:

$$P\left(-z_{\alpha/2} < \frac{\frac{X}{n} - p}{\sqrt{\frac{p(1-p)}{n}}} < z_{\alpha/2}\right) \approx 1 - \alpha \tag{2.1}$$

Using $\alpha = 0.05$ results in $z_{\alpha/2} = 1.96$. Subtstituting $z_{\alpha/2}$, X and n in equation 2.1 and solving for p results in a lower and an upper bound for p. Agresti and Coull suggest a more convervative method, especially for p nearby 0 or 1 (Dekking et.al. 2005, ch. 24, p. 364) . Hence, this could be suitable because of the excess of zeros. Define

$$\tilde{X} = X + \frac{z_{\alpha/2}^2}{2}, \quad \tilde{n} = n + z_{\alpha/2}^2, \quad \tilde{p} = \frac{\tilde{X}}{\tilde{n}}$$

with the confidence interval (CI):

$$\left(\tilde{p} - z_{\alpha/2}\sqrt{\frac{\tilde{p}(1-\tilde{p})}{n}}, \tilde{p} + z_{\alpha/2}\sqrt{\frac{\tilde{p}(1-\tilde{p})}{n}}\right)$$

For each error both confidence intervals are determined and collected in table 2.2.

Discussion

The confidence intervals of the two methods do not differ a lot. From the table it follows that the probability of occurring error B or D on a day on a machine is almost one. The probability A happens

Error	$\hat{p}$	CI Wilson	CI Agresti and Coull
A	0.3507	$(0.3035, 0.4010)$	$(0.3035, 0.4010)$
B	0.9178	$(0.8851, 0.9418)$	$(0.8848, 0.9422)$
D	0.9534	$(0.9267, 0.9707)$	$(0.9262, 0.9712)$
C	0.1890	$(0.1522, 0.2324)$	$(0.1521, 0.2325)$

Table 2.2: Estimation of the binomial parameter with a 95% confidence interval, calculated using Wilson and Agresti and Coull method

is much lower, namely 0.3507. The probability that C happens on a day on a machine is even lower: 0.1890. These are low values but still appear to cause a high total USD time, which could indicate to outliers of these two errors. In order to decrease the total USD time, these two errors should be handled differently than the other two: for A and C the focus could lay on decreasing the USD time for some specific occurrences while for D and B the focus could lay on decreasing the frequency.

Distribution of USD time

The USD distribution of each erroris right skewed. For example, for error D we get the density estimate shown in figure 2.2. The density estimates of the other errors can be found in appendix A.5 and

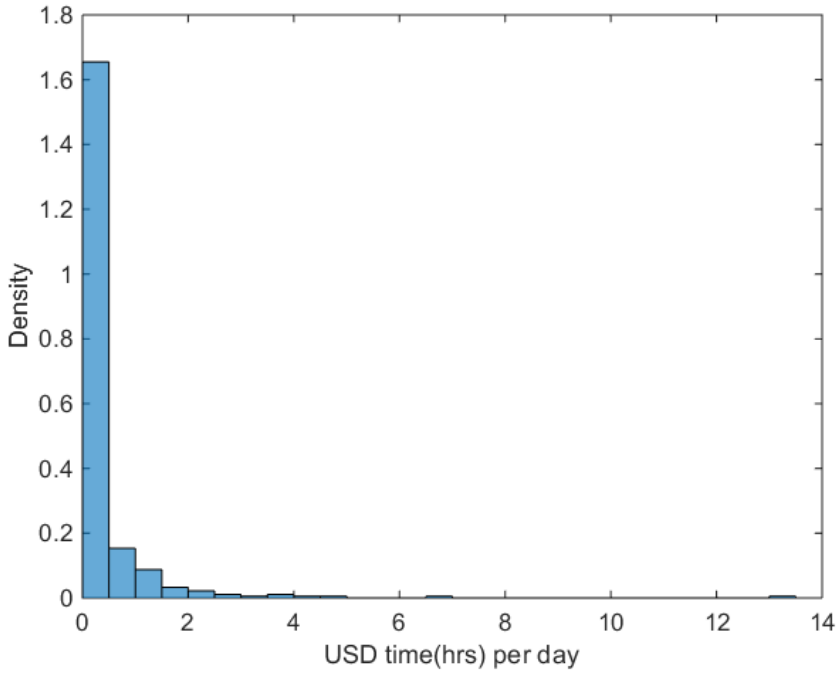


Figure 2.2: Estimated probability density function of the USD times of error D

also seem to be right-skewed distributed because of the inflated amount of zeros. As a consequence no parametric distribution is a suited fit. Therefore, the data are split into $USD = 0$ and $USD > 0$.

Probability of USD = 0

For $USD = 0$ the point mass can be estimated by the number of days where $USD = 0$ divided by the total number of 365 days. The estimates and their corresponding confidence intervals are observed with the 95% confidence interval and shown in table 2.3.

Distribution of USD>0

Error	Prob. of $USD = 0$	CI using Agresti & Coull
A	0.7123	(0.6638,0.7564)
B	0.1343	(0.1029,0.1733)
D	0.6712	(0.6214,0.7175)
C	0.8904	(0.8540,0.9187)

Table 2.3: Estimation of the probability of $USD = 0$ on a day with a 95% confidence interval

Looking at the $USD > 0$ the density estimates are obtained, shown in figure 2.3.

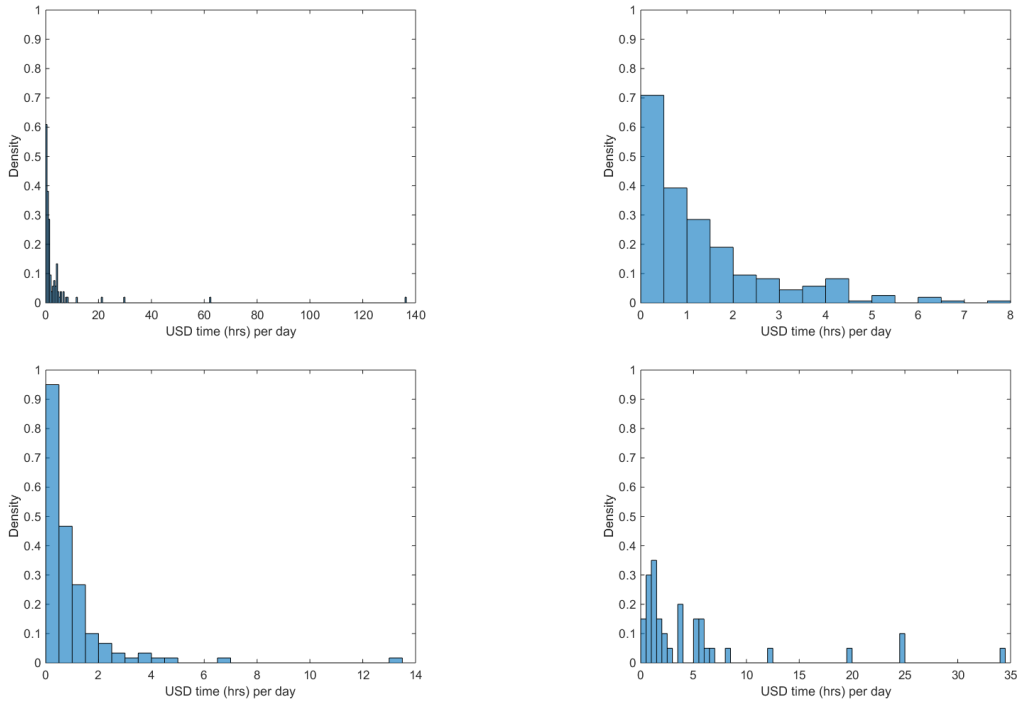


Figure 2.3: Estimated probability density function of $USD > 0$ of the four most important errors. From left to right and top left to bottom right: A, B, D, C.

Note, the limits of x-axes of A and C are higher compared to B and D since they contain some high USD times.

Discussion

Table 2.3 shows the highest probability for having USD=0 is for error C with 0.8904. This errors also has the lowest probability to occur. Note, the data of $USD = 0$ are unconditioned on the fact if the error happened or not. Hence, a low probability of occurrence may influence the high probability of $USD = 0$, since when no error occurred, $USD = 0$. B has a low probability of $USD = 0$ with 0.1342 and has a high probability that the error occurs according to table 2.2. In addition, D and A are also quite high in probability of having $USD = 0$, namely 0.6712 and 0.7123 respectively. However, it is interesting that the probability of occurrence of D is a bit high (0.9534). Hence, when the error occurs the probability of having $USD = 0$ is high.
Looking at USD>0, we see that all errors are still right-skewed distributed. This indicates that the data consist of a high amount of USD times nearby 0.

Conclusion

Errors A and C have a similar behavior: the highest peak of frequency per day is at 0 and they contain the highest probability of $USD = 0$ on a day, namely a probability of 0.71 and 0.89 respec-

tively. Still they have a high total USD time, indicating that these two errors contain extremely high USD times. These are investigated in the next section. D has a more spread frequency distribution and a lower probability of $USD = 0$ with 0.6712. This indicates that the high total USD time is caused by the high frequency. Just like D, error B also has a less skewed frequency distribution. The highest peak is around two: mostly two errors occur on a day on a machine. Further, it has the lowest chance of $USD = 0$ of 0.1343. This indicates that both frequency and USD times > 0 cause the high total USD time.
Hence, when we would like to reduce the total USD time of each error, we should solve them differently. For error A and C the frequency is not that high, but when the error occurs the USD time can be high. Hence, for these errors we should reduce the mean time to repair.
The errors B and D have a high frequency but not that high USD time if the error occurs. Hence, when reducing these two errors we should focus on preventing that the error occurs and so reduce the frequency.

2.2.2. Outliers

As the density figures already suggest, the error logging data set consists of some extreme high values. In figure 2.4 the USD times across the time are plotted.

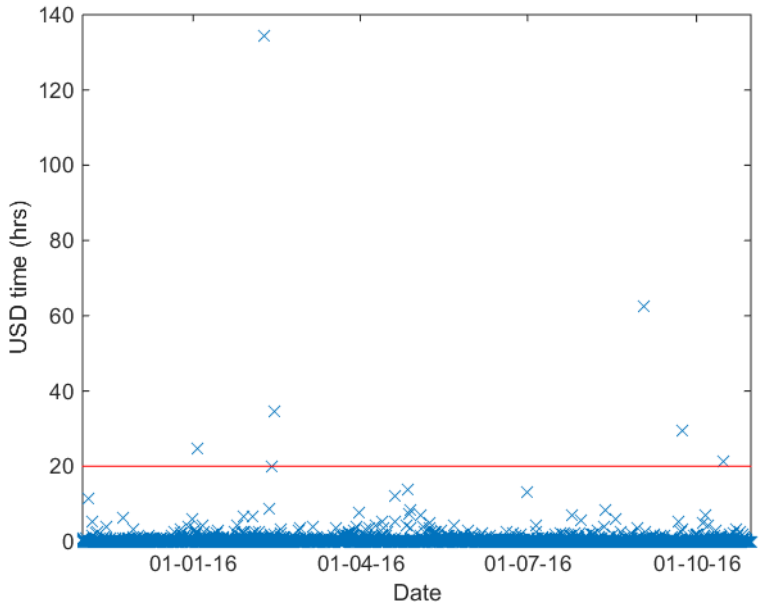


Figure 2.4: USD times of leveling errors between 01-11-2015 and 30-10-2016

From the figure can be concluded that six out of 4548 USD time values are higher than 20 hours. These six values correspond to the data shown in table 2.4.

Date	Error code	Root error ID	Machine number	USD time (hrs)	Context
03-01-2016	C	123	M1	24.86	Generic root error
08-02-2016	A	123	M2	134.45	Technical error
13-02-2016	C	123	M3	34.45	Earthquake
01-09-2016	A	123	M4	62.42	Technical error
23-09-2016	A	123	M5	29.51	Technical error
15-10-2016	A	123	M6	21.25	Generic root error

Table 2.4: Data corresponding to the outliers

From the table, it follows that four out of six outliers are errors of the type A and the other two from C. Looking at the context of those errors on that particular machine using the System Diagnostic Tool (SDT) gives the information displayed in tables 2.5 and 2.6.

The outliers do not appear during production time, but they occur while the machine was tested,

Outliers A	
08-02-16	Not available from SDT.
01-09-16	Symptom. The error occurred from testing.
23-09-16	Symptom. The error occurred while testing.
15-10-16	The error occurred while testing. Whole day alarms appeared and machine was rebooted.

Table 2.5: Information on machine corresponding to outlier A

Outliers C	
03-01-16	Not available from SDT.
13-02-16	Down due to earthquake

Table 2.6: Information on machine corresponding to outlier C

rebooted or something similar. These observations should not be included in the analysis since they do not contain any useful information and possibly affect the results negatively. This information is not eliminated since this will lead to a smaller data set. Therefore, these USD values are replaced by the median of the corresponding error. Ideally, all observations should be removed when the machine was tested, rebooted or any other form other than during production such that we would only use observations during production time. However, this information is only available for half of the current observations. Moreover, the information for every observation should be looked up manually which costs a lot of time. Therefore, we do not investigate for each observation if it occurred during production time or not, but assume that they all occurred during production time.

2.3. Initial data analysis

In this section an initial data analysis is done to the interplay between the four errors by investigating their dependency. When a pair of errors is dependent, it is likely that these errors are triggered for the same reason and hence we could solve them together. Both the chi-square test of independence and the Fisher's exact test are applied. Besides seeking for dependency between two errors, dependency between more errors is interesting. A pair of errors can be independent or dependent but can be independent or dependent given another error occurred. However, as we see in the upcoming section, we do not have enough data to test for this.

2.3.1. Independence test

Chi square test of independence

The chi-square test is a hypothesis tests in the analysis of categorical data. Given two categorical random variables X and Y, the following null and alternative hypothesis are tested:

$$H_0 : f(x, y) = g(x)h(y) \text{ i.e. X and Y are independent}$$
$$H_1 : f(x, y) \neq g(x)h(y) \text{ i.e. X and Y are not indepedent}$$

Where X takes values in set S with k elements and Y takes values in set T with m elements. The joint probability density function of (X, Y) is denoted as $f(x, y) = P(X = x, Y = y)$ for $x \in S$ and $y \in T$. The marginal probability density functions are

$$g(x) = \sum_{y \in T} f(x, y), \ x \in S$$
$$h(y) = \sum_{x \in S} f(x, y), \ y \in T$$

Now, the k and m elements represent both 2 with the sets S and T representing 'Yes' and 'No'. Let O_{xy} denote the number of days that the pair (x, y) occurs on the same day, on the same machine for every $(x, y) \in S\ X\ T$. Where

		Error Y occurred		
		Yes	No	Total
Error X occurred	Yes	$O_{11} = 0$	$O_{12} = 2$	$N_{yes} = 2$
	No	$O_{21} = 5$	$O_{22} = 339$	$N_{no} = 344$
	Total	$M_{yes} = 5$	$M_{no} = 341$	$n = 346$

Table 2.7: Example of frequency table for pair of errors

$$N_x = \sum_{y \in T} O_{x,y} \quad \text{i.e. the number of times that x occurs in sample X}$$
$$M_y = \sum_{x \in S} O_{x,y} \quad \text{i.e. the number of times that y occurs in sample Y}$$

This can be done for every possible pair of the four errors, which gives in total $\binom{4!}{(4-2)!2!} = 6$ frequency tables for each machine number. For the test statistic the estimated expected frequency is needed under H_0 for each cell. The best estimate for the density functions $g(x)$ and $h(y)$ is $\frac{1}{n}N_x$ and the same for for $h(y) = \frac{1}{n}M_y$. The expected frequency under H_0 is then given by

$$E_{x,y} = n * \frac{1}{n}N_x\frac{1}{n}M_y = \frac{1}{n}N_xM_y \tag{2.2}$$

Using these expected frequencies, the test statistic is calculated by

$$V = \sum_{x \in S}\sum_{y \in T} \frac{(O_{x,y} - E_{x,y})^2}{E_{x,y}} \tag{2.3}$$

The distribution of V converges to a chi-square distribution as $n \to \infty$ with $(k-1)(m-1)$ degrees of freedom. H_0 is rejected if and only if $V > \chi^2_{(k-1)(m-1)}(1-\alpha)$.

Fisher's exact test

For small n the chi-square distribution can not be assumed. Therefore we use also another technique: *Fisher's exact test*. This test uses the exact probability distribution of the observed frequencies where the row and column sums are fixed. Then one cell frequency can be picked that also determines the rest of the cell frequencies. The probability of obtaining a particular arrangement of cell frequencies $\{O_{11}, O_{12}, O_{21}, O_{22}\}$ is given by a hypergeometric distribution, conditioned on fixed row and column totals and assuming the two variables are independent (Everitt, 1977, ch.2, p.15). That is,

$$P = \frac{(O_{11}+O_{12})!(O_{21}+O_{22})!(O_{11}+O_{21})!(O_{12}+O_{22})!}{O_{11}!O_{12}!O_{21}!O_{22}!n!}$$

Then the p-value is calculated by summing the probability of the obtained observed frequencies and the probabilities of the frequencies which would represent more extreme discrepancies between the proportions of error y occurred yes or no with respect to error x occurred yes or no. For example, for the top left frequency table in table 2.8, the more extreme case would be for $4 < O_{11} \leq 163$. Then for each of these extreme cases the probability of having this particular arrangement can be calculated. Whereas the chi-square test is two-tailed, is the Fisher's exact test one-tailed (Everitt, 1977, ch.2 p.17-18). That is, the Fisher's exact test decides whether the proportions $\frac{O_{11}}{O_{21}}$ and $\frac{O_{12}}{O_{22}}$ are equal or whether $\frac{O_{11}}{O_{21}}$ is *greater* than $\frac{O_{12}}{O_{22}}$.

Now we can determine the p-value for each frequency table. In the next section we apply both the chi-square test of independence and Fisher's exact test and compare the results.

2.3.2. Application

Table 2.8 shows the frequency tables for each possible pair. Using these tables, we can apply both tests.

		B		
		Yes	No	Total
A	Yes	4	159	163
	No	979	44608	45587
	Total	983	44767	45750

		C		
		Yes	No	Total
A	Yes	1	162	163
	No	82	45505	45587
	Total	83	45667	45750

		C		
		Yes	No	Total
B	Yes	4	979	983
	No	79	44688	44767
	Total	83	45667	45750

		D		
		Yes	No	Total
A	Yes	2	161	163
	No	1364	44223	45587
	Total	1366	44384	45750

		D		
		Yes	No	Total
B	Yes	49	934	983
	No	1317	43450	44767
	Total	1366	44384	45750

		C		
		Yes	No	Total
D	Yes	0	1366	1366
	No	83	44301	44384
	Total	83	45667	45750

Table 2.8: Frequency table for each pair of errors

Application chi-square test of independence

The estimated frequency under the null hypothesis is calculated, using equation 2.2. Then we get the p-values shown in table 2.9 The pairs A & C and B & D are significantly dependent. The rest are

Error X	Error Y	P-value
A	B	1.00
A	D	0.17
A	C	0.00
B	D	$1.39 \cdot 10^{-4}$
B	C	0.15
D	C	0.15

Table 2.9: P-values after applying the chi-square test for testing independence

likely to be independent. However, to recall, the expected cell frequencies should not be too small, since then the p-values may not be reliable. Typically this is interpreted by an expected cell frequency of less than five. But as Everitt (1992, p. 39) pointed out, no mathematical or empirical evidence is found for this rule. While no clear rule is defined, for skewed data the chi-square distribution may not provide an accurate estimate of the underlying sampling distribution and therefore the p-values should be used by caution. To be more certain about the dependence, we calculate the exact probability distribution by the Fisher's exact test.

Application Fisher's exact test

Applying this test, we get the p-value shown in table 2.10. From the table it follows that five out

Error X	Error Y	P-value
A	B	0.47
A	D	0.96
A	C	0.26
B	D	$3.95 \cdot 10^{-4}$
B	C	0.10
D	C	1.00

Table 2.10: P-values after applying the Fisher's exact test for testing independence

of six pairs are likely to be independent. One pair is likely to be dependent. That is the pair B and D. The strength of dependency can be measured by the odds ratio (OR).

$$\text{OR} = \frac{\frac{O_{11}}{O_{21}}}{\frac{O_{12}}{O_{22}}} = \frac{O_{11}O_{22}}{O_{21}O_{12}} = 1.73$$

This means, that the proportion of the occurrence of error B is 1.73 higher given error D occurs compared that error D not occurs.

Conditional dependence or independence

Now we can assume dependence or independence between each specific pair. We would like to know if each pair is conditional independence or conditional dependence. To test for this, we could make the same frequency table as in table **??** only now we count the number of occurrences given that the conditioned error also occurred. However, as we noted before the cell frequencies of O_{11} in table 2.8 are already small. When we would like to condition on the fact another error occurred, these cell frequencies become even smaller. Namely, then $O_{11} \leq 4$, since now three errors should have occurred together instead of two. Moreover, O_{12} and O_{21} are small since for these two cell frequencies still two errors should occur together (instead of one as in the non-conditional case). Therefore, we expect that we do not have enough data to test for the conditional dependencies or conditional independencies and that more data should be needed.

2.3.3. Conclusion

Comparing the results of the chi-square test and the Fisher's exact test we see that in both cases the pair B & D is likely to be dependent. The chi-square test also shows that the pair A & C is significantly dependent whereas the Fisher's exact test shows this is not significant. Since we assume that the Fisher's exact test is more reliable, we assume that the pair A & C is independent. Now, when solving one of the two errors of the dependent pair it is likely that the other error also will be solved since it is likely that those two errors are triggered by the same cause. As we also see in section 6, both errors could occur because of the same reason.

3

The leveling configuration data set

In this chapter are data gathered consisting of the leveling configurations for each machine. Afterwards, these number of leveling configurations is reduced by dimensionality reduction techniques.

3.1. Data collection

The data of *leveling configurations* are collected in order to investigate sub research question three i.e. investigating how the leveling configurations influence the USD time of the most important errors. Each leveling configuration provides *options*, for which a customer can choose from during wafer measurements or wafer exposure. The most recent leveling configurations with their options are extracted using the PMA Data Extractor. Each configuration is displayed per machine number. In the further investigation the term leveling configurations is abbreviated to *configurations*. These configurations are considered as the *features* or *explanatory variables* in the further analysis. The options a customer can choose from within the feature are considered as the different *levels* of a feature.

Irrelevant features

The table in appendix B.1 shows a total of 23 features. Ideally, all of the 23 features should contribute to provide information about the way the machine is used and what are the differences in usage between all the machines. However, some features are missing and are therefore removed. Further, some of the features contain the same options across all observations. These features will not provide information in discriminating the usage of the machines. These irrelevant features are eliminated and can be found in the table of appendix B.2.

Feature set 1

In total 12 features remain. This is named as *feature set 1*. In table 3.1 the configurations are listed with their corresponding abbreviations.

Levels of each feature

Most features can take two levels, for example 'Enabled' or 'Disabled'. However, the features `system type` and `location` contain 57 and 19 levels respectively. As a consequence, we have not enough data per level. Therefore we would like to merge certain levels, even though this leads to information loss. By merging we have enough data per level so that we can apply tests on it. Based on domain knowledge, levels within each of the two features are merged. Originally, `location` gives information about what customer at what country or state at what factory. This information is merged such that only the information of customer is present. For the customers C1,C2 and C3 no merging is done, since it is expected that enough data are present per factory. Moreover, the customers C4 and C5 are merged, and C7 with C8 as well. The feature `system type` contains information about product type (NXT), the

Feature	Abbreviation
System type	MachType
Location	Loc
Field width optimised leveling	FieldW
Leveling Field Extensions Algorithm	FieldE
Leveling Setpoint Smoothing	Smooth
Leveling on single LS Spots	Single
Leveling with LS Spot Weight Update Algorithm	Spot
Air Gauge	AG
Air Gauge Improved Leveling	AGILE
Type of Air Gauge	AGT
FSM Flexibility package	FSMFlex
Improved FSM algorithm. Part of FIP-1 commercial package	iFSM

Table 3.1: Leveling configurations types with their abbreviations which form feature set 1

different type of lenses (19xx), throughput-level (B,C,D) and having immersionhood or not (i). These levels are merged into the product type (NXT) and the 19xx information. For the newer machines, NXT1980, the throughput-level is kept since it is not sure if those can be considered as the same, based on domain knowledge. After merging those levels, the feature consists of 48 levels for location and 7 levels for system type. The features `leveling setpoint smoothing` and `type of air gauge` contain both a level with a small amount of data and are similarly named to another level that contain more data. Therefore for these features these levels are merged as well. The tables in appendix B.3 show a list of the merged levels and from what original levels they are derived. In the table of appendix B.4 each feature with its corresponding levels is shown. Each of the features are considered as *nominal variables*, i.e. no ordering between the levels is assumed.

3.2. Dimensionality reduction

Features become unnecessary if they contain highly similarly information. One way to identify highly similar groups of features is by applying *clustering*. A cluster is a group of variables which are similar to each other and dissimilar to other variables. Another way to look for dependency is by calculating the linear correlation between each pair of variables. Also by correlation, groups of similar variables can be discovered when all the variables within a group have a high correlation with each other and a low correlation with the other variables. Besides clustering and calculating correlation, dependency can be calculated by the test of independence using frequency tables like done in chapter 2.3.2. Moreover, with multi-way tables also conditional dependence or conditional independence can be identified. Although, this is interesting for identifying structures between variables it will not add information in identifying similar groups and dissimilarity with other groups. It will only add information in identifying similar variables or dissimilar variables given another variable. However, that the other variable is given can not be guaranteed for every machine. Hence, this will not lead to dimensionality reduction. Moreover, we do not have enough data to condition on an other variable.

Since clustering has the advantage of visualization, this method is used for finding similar groups. Then to check, the correlation is calculated for every pair to see if the similar groups are also high correlated with each other. A cluster represents a group of variables which share common information. Ideally, some clusters are found and so the feature set can be reduced, that is called *dimensionality reduction*. Summarizing the data in a smaller amount of variables gives a brief description of the patterns and differences in the data.

Different methods exist to form clusters. In general, clustering methods can be split into two parts: hierarchical clustering and partitioning clustering, from which K-means clustering is a well known method. For categorical data, hierarchical clustering and K-means clustering are both applicable (Everitt, Landau, Leese & Stahl, 2011, ch. 9, p. 258). However, the result of K-means clustering is less stable since it depends on the given input k (Singh, Malik & Sharma, 2011). Therefore, hierarchical clustering is applied. As a comparison and because of the nice visualization aspect, multidimensional scaling (MDS)

is studied as well.

3.2.1. Hierarchical clustering

Hierarchical clustering can be done agglomerative or divisive. In the *agglomerative* version each variable is seen as a separate cluster. Then the two 'closest' (most similar) clusters are combined into a new cluster. This repeats until one single cluster is formed. This method is also called a 'bottom-up' method. The *divisive* method works the other way around: all variables belong to one cluster and is then partitioned into two clusters which are least similar. This repeats until there is one cluster for each variable. In this report agglomerative clustering is done. The algorithm is as follows:

Algorithm 1 Agglomerative hierarchical clustering algorithm

Input: A set of variables $\{x_1, ..., x_J\}$
Output: Dendrogram
1: Place each variable j into its own singleton cluster c_j which results in $C = \{c_1, ..., c_J\}$
2: **for** $k = 1 : J - 1$ **do**
3: **for** some *dissimilarity measure* $d(c_i, c_j)$ **do**
Calculate the dissimilarity between each possible pair of clusters
Find the closest cluster pair c_r and c_s by $d(c_r, c_s) = \min\{d(c_i, c_j)\}_{i,j \in J, i \neq j}$
4: **end for**
5: Remove c_r and c_s from C
6: Add $c_k = \{c_r, c_s\}$ to C
7: **end for**

Once a variable belongs to a cluster, then that variable cannot be removed from that cluster anymore. The result of hierarchical clustering is called a *dendrogram*. This is a hierarchical tree where each node or leaf represents a variable and where the length of the stems represents the distance or dissimilarity at which a variable or cluster is joined.

As can be seen in the algorithm, a *dissimilarity measure* is required. This measure is calculated using a similarity measure $s(.)$ by

$$d(c_i, c_j) = 1 - s(c_i, c_j)$$

In order to calculate the similarity between two clusters, two aspects need to be defined: the definition of similarity between two variables and a definition of similarity between two clusters from which at least one consists of more than one variable. This is called the *inter-group* measure.

First, defining the similarity measure between two variables is considered. Everitt et al. (2011, ch. 3, p. 47) propose different similarity measures for categorical data by transforming these into binary data and using a frequency table:

		Dummy variable i		
		1	0	Total
Dummy variable j	1	a	b	$a + b$
	0	c	d	$c + d$
	Total	$a + c$	$b + d$	$p = a + b + c + d$

Table 3.2: Example of frequency table

Each of the nominal variables with k levels are transformed into k dummy variables, which can take value 1 or 0 describing the presence or absence of that attribute. Note, the data are not transformed into $(k - 1)$ dummies, since each level needs to be tested for dependency and since there is no reference dummy needed. For example, the variable `AG type` has three levels, so this variable is transformed into 3 dummy variables: 'Type 1 Initial AG (-25.5 mm)', 'Type2 Shifted AG' and 'No AG device present'.

When selecting a similarity measure it needs to be considered what will be defined as similar: are

zero-zero matches (d in the frequency table) just as important as one-one matches (a)? In this study one-one matches are considered to be more important than zero-zero matches. Since the interest in finding when two variables are both present and not when they are both absent, the focus lies on one-one matches. Moreover, some nominal variables consist of more than two levels, which can lead to a large amount of 'negative' matches. Therefore, d is not considered and the following measures are left over according to Gower & Legendre (1986) and Everitt et al. (2011, ch.3, p. 47):

Jaccard(i,j)	$\frac{a}{a+b+c}$
Dice(i,j)	$\frac{2a}{2a+b+c}$
Sneath and Sokal(i,j)	$\frac{a}{a+2(b+c)}$
Gower and Legendre(i,j)	$\frac{a}{a+\frac{1}{2}(b+c)}$
S5(i,j)	$\frac{1}{2}\left(\frac{a}{a+b}+\frac{a}{a+c}\right)$
S6(i,j)	$\frac{a}{\sqrt{(a+b)(a+c)}}$

Table 3.3: Different similarity measures for binary data where 'negative' matches are excluded

The differences in these similarities lie in weighing the single presences differently or by normalizing the co-presences (a) by their own variable frequencies (S5 and S6). Since there is no interest for correcting rare frequencies and because of the straightforward interpretation, the Jaccard similarity is chosen.

Secondly, the inter-group measure needs to be defined. This is a measure of the dissimilarity between two clusters from which at least one is not a singleton cluster. Everitt et al. (2011, ch. 3, p. 61) consider 3 types: *single linkage, complete linkage or group average clustering*. Single linkage takes the dissimilarity between the two closest individuals, one from each cluster. Complete linkage is the opposite: it takes the largest dissimilarity between any two individuals, one from each cluster. The last method stays in between the two extremes of above: the average dissimilarity between the individuals of both groups is taken. For this research *single linkage* method is used, since this shows the variables which are nearest to each other and leads to a clearer interpretation.

3.2.2. Multidimensional scaling

Besides hierarchical clustering, multidimensional scaling (MDS) is a method to get insight in possible dependencies. This technique gives sign of the presence of clusters that are useful for dimension reduction. MDS is considered in order to be more sure about the data structure and because of the attractive visualization aspect. The technique represents variables as a spot in a lower-dimensional space where the original distances between the variables are tried to be preserved. Thus, two variables which are originally close to each other, should also be relatively close in the lower-dimensional space. The closer the variables, the more similar they are.

Different variations of MDS techniques exist such as *classical MDS*, which assumes the dissimilarities to be Euclidean distances (Borg, Groenen & Mair, 2013, ch. 8, p.83). Further *metric* and *nonmetric* MDS exist. Nonmetric MDS is an ordinal method where the rank-order of the dissimilarities is considered. In metric MDS the observed interval scaled dissimilarities are considered (Borg & Groenen, 1997, ch.9, p. 200-203). The classical MDS is not considered here since the data do not represent Euclidean distances. Nonmetric MDS is less preferred compared to the metric variant since the interest lies in the actual dissimilarities and not in the sorted ones. However, since it is not sure if the distances between the binary variables are preserved by the metric MDS, also nonmetric MDS is considered. Applying both will help in deciding which method is more convenient and if the possible gaining of fit in nonmetric MDS can compensate for the lack of interpretation of the plot.

The algorithm of nonmetric MDS is as follows:
The stress function used is the Kruskal's nonmetric stress criterion (Kruskal, 1964):

$$\text{STRESS}_{nonmetric} = \sqrt{\frac{\sum_{i\neq j}^{J}(d_{ij}^{*} - f(\delta_{ij}))^2}{\sum_{i\neq j}^{J} d_{ij}^{*2}}}$$

Algorithm 2 Nonmetric multidimensional scaling algorithm

Input: $J \times n$-dimensional matrix X_J representing J variables and n observations
Output: $m-$dimensional plot of the variables X_j such that the pairwise dissimilarity is presented as best as possible with $m < J$

1: **for** the Jaccard similarity measure **do**
2: Calculate dissimilarities $\delta_{ij} \ \forall$ variables $i, j \in J, \ i \neq j$
3: **end for**
4: For some starting configuration find a mapping in the $m-$dimensional space of the points $x_1, ..., x_J$
5: Calculate the pairwise distances $||x_i - x_j||_2 = d^*_{ij}, \ i, j \in J$
6: Calculate the stress function
7: **while** Stress function is larger than some criterion **do**
8: Find a new mapping configuration f of $x_1, ..x_J$ s.t. the order of δ_{ij} is presevered, i.e.
9: whenever $\delta_{ij} < \delta_{kl} \Leftrightarrow f(\delta_{ij}) < (\delta_{kl}) \Leftrightarrow \hat{d}_{ij} \leq \hat{d}_{kl} \ \forall i, j, k, l \in J$
10: Recalculate the pairwise distances $||x_i - x_j||_2$
11: Recalculate the stress function
12: **end while**

where d^*_{ij} is Euclidean distance measure between the two variables x_i and x_j in the $m-$dimensional space.

$$d^*_{ij} = \left(\sum_{a=1}^{m} (x_{ia} - x_{ja})^2 \right)^{1/2}$$

And where $\hat{d_{ij}}$ is called a *disparity* which is a transformed dissimilarity such that only the order is preserved. A perfect MDS solution has a stress value of zero. If $\delta_{ij} = \delta_{kl}$, called a *tie*, the primary approach to ties is handled. That is, when $\delta_{ij} = \delta_{kl}$ it is not of importance which d_{ij} or d_{kl} is larger, nor if they are equal or not (Kruskal, 1964).

The algorithm for metric MDS is almost the same, only step eight and nine will be different: the mapping function f can take other forms than monotonic (Borg & Groenen, 2005, ch.9, p.201-202). This gives the algorithm stated in algorithm 3.

Algorithm 3 Metric multidimensional scaling algorithm

Input: $J \times n$ dimensional matrix X_J representing J variables and n observations
Output: $m-$dimensional plot of the variables X_j such that the pairwise dissimilarity is presented as best as possible with $m < J$

1: **for** Jaccard similarity measure **do**
2: Calculate dissimilarities $\delta_{ij} \ \forall i, j \in J, \ i \neq j$
3: **end for**
4: For some starting configuration find a mapping in the $m-$dimensional space of the points $x_1, ..., x_J$
5: Calculate the pairwise distances $||x_i - x_j||_2 = d^*_{ij}, \ i, j \in J$
6: Calculate the stress function
7: **while** Stress function is larger than some criterion **do**
8: Find a new configuration of $x_1, ..x_J$
9: Recalculate the pairwise distances $||x_i - x_j||_2$
10: Recalculate the stress function
11: **end while**

The stress function is now defined as:

$$\text{STRESS}_{metric} = \sqrt{\frac{\sum_{i \neq j}^{J} (d_{ij}^{*} - f(\delta_{ij}))^2}{\sum_{i \neq j}^{J} \delta_{ij}^2}}$$

Both the MDS techniques need a *starting configuration*. Then the MDS distances are found by a sequence of little replacements of the spots in the plot such that the stress value decreases. As a consequence, the computing algorithms find local minimum solutions, meaning that any little replacements of the spots lead to a higher stress value. Different starting configurations lead to different local minima (assuming they exist). To be sure that MDS gives the smallest possible stress value, which is the global minimum, the starting configuration is important. Borg, Groenen and Mair (2013, ch.7, p.62) recommend to use the solution of the classical MDS as starting configuration if there is not a prior theory about the locations of each spot, which is also called Torgerson solution.

Note, for both metric as nonmetric Jaccard is chosen as dissimilarity measure, for the reasons explained in the hierarchical clustering chapter.

3.3. Application on the configuration data set

The hierarchical clustering method and multidimensional scaling method is applied on machine data. In total we have 241 machines. Every machine number with their configurations is one observation. In here the configurations are transformed to binary data as explained in chapter 'Hierarchical clustering'.

3.3.1. Hierarchical clustering application

The configurations considered are *feature set 1*. `System type` and `location` are not included, since these two variables contain a lot of levels (7 and 48 respectively), which results in a low similarity measure for each level. As a consequence, the result is a *chaining* effect and an unclear tree structure, since lots of leafs are added one by one at the end of the tree. Moreover, because of the low similarity values no strong correlations are expected hence this does not add value for dimension reduction. One could solve this by for example, using another similarity measure which is described in Everitt et al. (2011, ch. 3, pp. 47-49). This measure can handle the high amount of negative matches. Merging the levels even further can be another solution. Both solutions are considered as out of scope for this study.

Figure 3.1 shows the dendrogram.
On the x-axis, the leaf nodes are represented, which contain each level within a configuration. The y-axis represents the distance or dissimilarity. The length of each link represents the dissimilarity with the other cluster. From the dendrogram it follows that the following pairs of settings are highly similar:

Configurations		Dissimilarity value
FieldE-Avg	Single-NotUse	0.0071
FieldW-En	iFSM-En	0.0085
AG-Present	Agile-2	0.0079
FieldE-Local	Single-Use	0.0099

Table 3.4: Pair of configurations which are highly similar

The first and the last pair can be extended to a broader cluster by adding the configurations `spot` and `Smooth`. Another somewhat clear cluster is AG-absent with Agile-No. FieldW-Dis with iFSM-Dis can also be defined as a cluster but already have some higher dissimilarity.

In order to verify the dendrogram, we check how well the height of the link represents the original dissimilarity between each pair of clusters. For this the *cophenet correlation* can be used. This is a measure of how well the dendrogram reflects the dissimilarities (Everitt et al., 2011, ch. 4, p. 91). The closes the linear correlation is to one, the more accurately the tree reflects the original dissimilarities. The corresponding cophenet value is 0.7826. To be certain about the clusters found and to visualize clusters in an other way, multidimensional scaling is used.

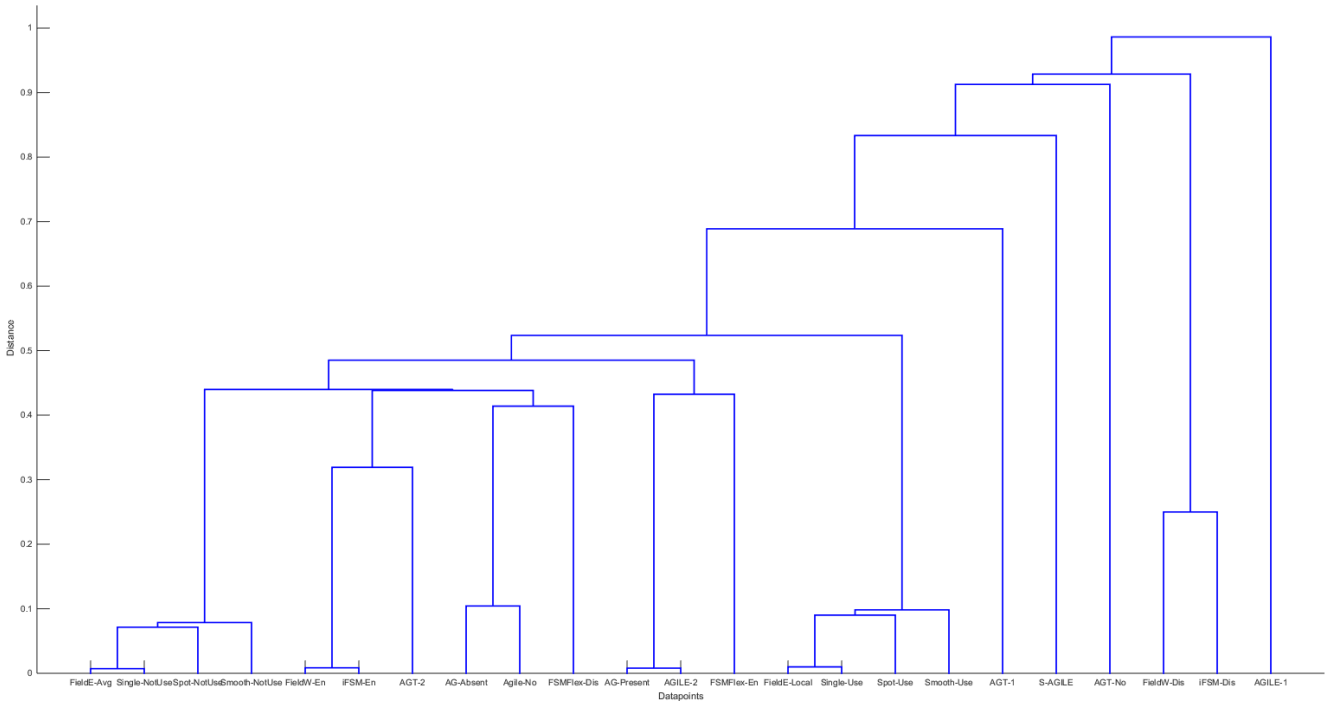


Figure 3.1: Dendrogram of feature set 1 except for `System Type` and `Location` using Jaccard similarity measure and single linkage

3.3.2. Multidimensional scaling application

The same data set as for hierarchical clustering is used where each machine number with its configurations is one observation. The configurations used are the same for hierarchical clustering, that is *feature set 1* except for **`system Type`** and **`location`** since they cause an unclear plot because of the inflation of number of spots in the plot and because a lot of points are co-located such that the labels are not readable. Solutions to overcome this problem are suggested in the previous chapter.

Before getting the $m-$dimensional plot using metric or nonmetric dimensional scaling, the value of m needs to be determined for each methods. For every different m value, a different minimum stress function is obtained. In the following figure both stress values are plotted against the number of dimensions for both metric as nonmetric MDS. For metric MDS 1500 iterations are enough such that the stress value converges. For nonmetric MDS 1500 iterations are enough up to $m = 5$, after that 10000 iterations are used.

In the figure both stress values are plotted in the same graph. Since both values are calculated differently, they cannot be compared directly. Nonmetric MDS stabilizes after six dimensions and metric MDS stabilizes after eight dimensions. Kruskal (1964) states rough guidelines for the goodness-of-fit, shown in table 3.5.

$\text{STRESS}_{nonmetric}$	Goodness of fit
0.20	Poor
0.10	Fair
0.05	Good
0.025	Excellent
0.00	'Perfect'

Table 3.5: Rough guidelines according to Kruskal for nonmetric MDS

$\text{STRESS}_{nonmetric}$	m
0.1406	2
0.0870	3

Table 3.6: Stress values for the nonmetric MDS for an m-dimensional plot

The stress value for nonmetric MDS is shown in the table above for $m = 2$ and $m = 3$. where a

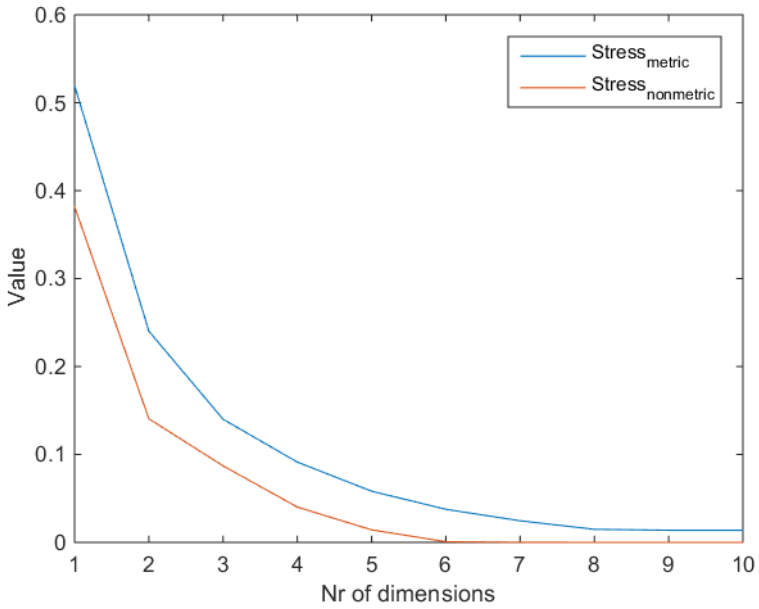


Figure 3.2: Plot of both stress values against the number of dimensions for nonmetric and metric MDS

$\text{stress}_{nonmetric}$ of 0.2 is rather 'poor' and of 0.1 is 'fair'. So for $m = 2$ or $m = 3$ the nonmetric MDS does not perform exceptionally well and hence will not compensate for the lack of interpretation. Ideally, a two-dimensional plot would be best interpreted. However, because of the high decrease in value for metric MDS, a value of $m = 3$ is chosen which corresponds with a stress value of 0.1400.
Figure 3.3 shows the 3−dimensional plot after applying metric MDS. The third dimension is presented

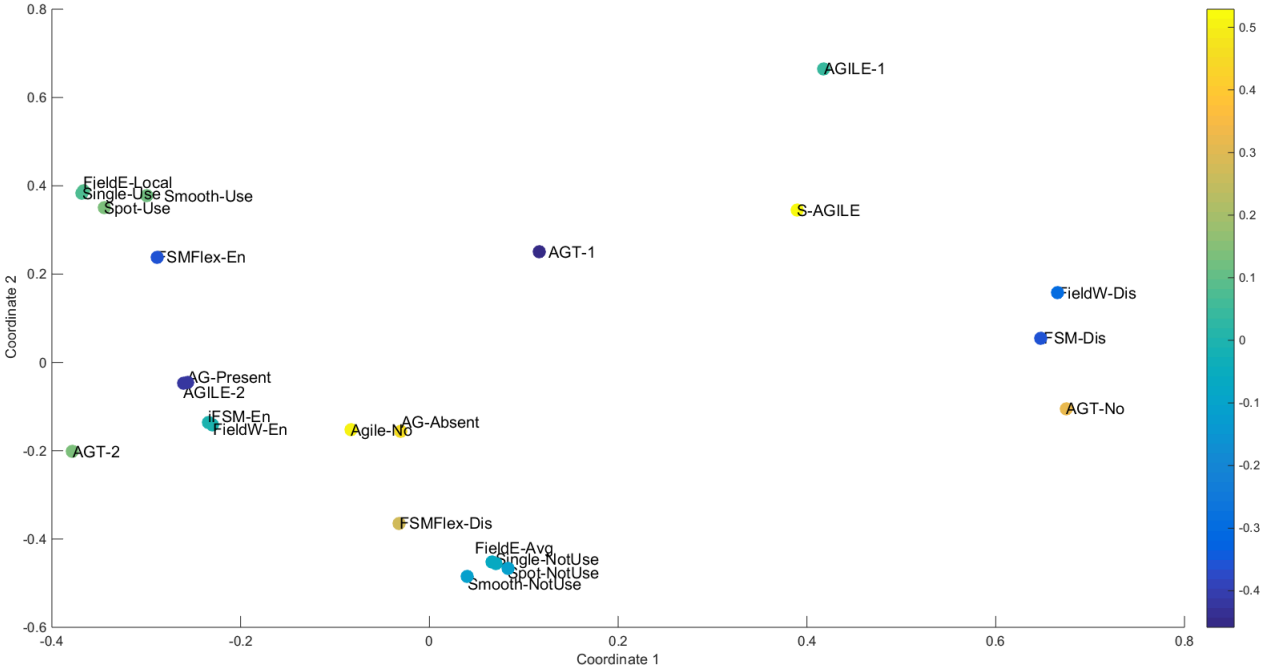


Figure 3.3: The 3-dimensional metric MDS plot using the Jaccard similarity measure applied on feature set 1 except for `system type` and `location`

as a color indicating the height. The axes are the principal axes of the solution space. This dimension

system can be arbitrarily rotated and reflected. Hence, the interpretation of the dimensions is not possible. What one can interpret are the relative distances; which point is close to which others and far from other points? Some sign of clustering can be seen from the plot where the same highly similar pairs can be discovered as in the dendrogram.

3.4. Results

Both methods show strong sign of the presence of clusters: some variables are highly similar and dissimilar from the rest. In table 3.7 the groups of similar variables are listed, revealed by the dendrogram and the MDS-plot. Note, in order to apply hierarchical clustering and MDS, these categorical features had to be changed to binary data as explained in the chapter 'Hierarchical clustering'. In the table below, the binary data are transformed back to the original variables. The remaining variables from the plot,

Clusters as found in dendrogram & MDS-plot	**Binary variables**	**Original variables**
Cluster 1	FieldE-Avg Single-NotUse Spot-NotUse Smooth-NotUse	FieldE Single Spot Smooth
Cluster 2	FieldE-Local Single-Use Spot-Use Smooth-Use	
Cluster 3	AG-Present Agile-2	AG AGILE
Cluster 4	AG-Absent Agile-No	
Cluster 5	FieldW-En iFSM-En	FieldW iFSM
Cluster 6	FieldW-Dis iFSM-Dis	

Table 3.7: Clusters which are visual from the dendrogram and MDS plot

`FSMFlex disabeld and enabled`, `AG type 1,2 and no`, and `AGILE 1 and S-AGILE` are not highly similar to the others and therefore stay separate. To check if the variables found in the clusters are indeed dependent, the Pearson's correlation is also calculated. We get the correlation matrix shown in appendix B.5. From this matrix, it follows that the absolute value of the correlations for each pair *within* cluster 1 and 2 are significant and higher than 0.89. Within cluster 3 and 4, the absolute value of the correlation is significant and higher than 0.90, for cluster 5 and 6 we have an significant absolute value of the correlation higher than 0.86. Hence within each cluster the pairs have a significant and high correlation coefficient.
The correlation *between* the clusters can be calculated by looking at every possible pair where one variable is from one cluster and the other variable from the other cluster. The variables considered are the ones from the original variables in each cluster. The highest absolute value of correlation between clusters 1&2 with respect to 3&4 is 0.09 and with respect to 5&6 is 0.17. The highest absolute value of correlation between clusters 3&4 with respect to 5 & 6 is 0.15. Hence, the correlation between clusters is low.
Now looking at the correlation between the *singleton clusters* and the other variables we see that the highest absolute correlation value are all lower than 0.48. Hence, the correlation values between the singleton clusters and the other variables are low.
Therefore, the correlation values confirm the founded clusters.

3.5. Conclusion

When variables are similar or dependent, there is no need to include them all since they will add no extra information. For example, when a machine is in the presence of `fieldE-local` it is almost certain that it also has `single-use`. Thus this variable will not add extra information in finding the discrepancies

in usage of the machines. As a consequence, some variables can be removed from the data set and can be represented by one of the variables from the cluster such that correlation is counteracted. Table 3.8 shows an overview of the clusters and the variable that represents the cluster.

Clusters	Represented by	
FieldE-Avg, Single-NotUse, Spot-NotUse, Smooth-NotUse	FieldE-Avg	FieldE
FieldE-Local, Single-Use, Spot-Use, Smooth-Use	FieldE-Local	
FieldW-En, iFSM-En	iFSM-En	iFSM
FieldW-Dis, iFSM-Dis	iFSM-Dis	
AG-Present, AGILE-2	AG-Present	AG
AG-Absent, Agile-No	AG-Absent	

Table 3.8: Clusters and the corresponding variable which represents the cluster

Which variable to choose to represent the cluster can be done in different ways. One way would be performing a principal component analysis (PCA) based on the variables in the clusters. However, Mori, Kuroda and Makino (2016, ch.2, p.7) state that for categorical data non-linear PCA is required. Another way is choosing the 'central' variable in the plot. Though there is no central variable in the plot when two variables represent a cluster. Another approach would be choosing the best regressor. The disadvantage of this method is that the selected variables depend on the response variable. Consequently, these variables can be different for every other data set having different USD times. Moreover, we select the variables which suits the model most and we may get a better model than we actually have. Therefore, for a cluster consisting of two variables we pick one arbitrary. For the cluster consisting of four, we pick the variable which is closest to one of the others. That is, we need to choose between FieldE or Single. Picking one arbitrary leads to FieldE. A limitation of representing the cluster by one of the variables is that we lose the information of the other variables. However, as the correlations within each cluster are high, we expect not to loose too much information. Another limitation is that we may choose the wrong representative variable. As a consequence, important information from one variable is not captured by the representative variable. However, I expect that this will not be the case because of the high correlations. In the chapter 'Future Research' we also see that choosing other representatives are not expected to effect the predictive models too much.

Now considering the clusters and dependencies we can summarize the features. Note, the separate clusters **`AG type`** and **`FSMFlex`** remain in the model. Moreover, **`location`** and **`system type`** as well since these are not tested for dependency. *Feature set 1* can now be reduced from 12 to 7 nominal variables, this data set will be called *feature set 2* in the further analysis:

Feature	Abbreviation
System type	MachType
Location	Loc
Leveling field extensions algorithm	FieldE
FSM flexibility package	FSMFlex
Improved FSM algorithm. Part of FIP-1 commercial package	iFSM
Air Gauge	AG
Type of air gauge	AGT

Table 3.9: Feature set 2 after applying hierarchical clustering and MDS on feature set 1 (except for location and system type) remains 7 out of 12 variables

4

Assessing features based on rank differences

Now the reduced feature set is determined, it is of interest how the configurations are related with the errors, in particular with the USD time. To find differences between groups, the *Wilcoxon rank sum test* and the *Kruskal-Wallis test* are used. Using these techniques, relevant features to the USD time contribution can be discovered and can be distinguished from features which are less relevant on its own. Both techniques are distribution-free and as a consequence, is suitable for the non-parametric USD distribution as is discussed in chapter 2.2.1 (Hollander, Wolfe & Chicken, 2014, ch.4, p. 115 and ch. 6, p. 204). Another reason for considering those two methods is the robustness. Both rely on ranking the data. This results in an analysis and an outcome that is less sensitive for high USD times. The difference between the two tests lies in the number of groups it can handle. The Wilcoxon rank sum test considers a variable of two levels. Whereas the Kruskal-Wallis test considers a variable of more than two levels.

In this analysis the focus lies on one error: B. The other errors are recommended for further analysis and fall out of the scope of this research. Ideally, differences are found between groups such that for one group the USD times are significantly higher or lower than for the other group. This could lead to an indication of improvement of usage for the machine and customer.

Just like in the previous chapter, machine data are considered, which are in total 241 machines. Each observation corresponds to a machine number with its configurations, where the dependent variable is the total USD time of the 1-year data. Recall, the configurations considered are the ones in *feature set 2*.

4.1. Wilcoxon rank sum test

4.1.1. Theory Wilcoxon rank sum test

This test considers data of two mutual independent samples $X_1, ...X_m$ and $Y_1, ..., Y_n$ where $n + m = N$ and N is the total number of observations. Applied on this data we have, $X_i = (\text{USD}_i|\ \text{var} = \text{level 1})$ for $i = 1, ..., m$ and $Y_j = (\text{USD}_j|\ \text{var} = \text{level 2})$ for $j = 1, ..., n$. The method assumes that the two samples are drawn from two continuous populations and that the two populations are equivalent under the null hypothesis (Pratt & Gibbons, 1981, ch.4, p.249). The alternative hypothesis can be one or two tailed. Since it is of interest to find out if a specific sample has a tendency to be smaller than the other sample, a one tailed alternative hypothesis is considered. Let F be the distribution corresponding to sample X, i.e. $X \sim F$ and G the distribution corresponding to sample Y, i.e. $Y \sim G$. The following hypotheses are tested by the wilcoxon rank sum test:

$$H_0 : F(x) = G(x)\ \forall x$$
$$H_1 : F(x) \geq G(x)\ \forall x \text{ and strict inequality for at least one } x$$

That is, the random variable X is stochastically smaller than the random variable Y. It can also be written as $P(X > Y) \leq P(X < Y)$ (Gibbons & Chakraborti, 2003, ch.6, p. 234).
The procedure for retrieving the rank sum test statistic is described by algorithm 4.

Algorithm 4 Wilcoxon - retrieving rank sum test statistic algorithm

Input: Two continuous sample sizes $X = \{X_1, ..., X_m\}$ and $Y = \{Y_1, ..., Y_n\}$.
Output: Test-statistic R_x or R_y

1: Combine X and Y in one group: $\{X_1, ..., X_m, Y_1, ..., Y_n\}$ with total size $m + n = N$
2: Rank the combined observations by $\{R_1, ..., R_N\}$ from smallest to largest
3: Sum the ranks of the observations of X resulting in R_x
4: Sum the ranks of the observations of Y resulting in R_y

Both R_x or R_y could be a test statistic, which is called the Wilcoxon rank sum statistic (Pratt & Gibbons, 1981, ch.4, p.250). The well-known Mann-Whitney statistic U is equivalent to the rank sum statistic. U can be interpreted as the number of (X_i, Y_j) pairs such that $X > Y$, and reversing the test statistic gives U' indicating the number of pairs (X_i, Y_j) such that $X < Y$. The relationship between the two statistics is as follows (Pratt & Gibbons, 1981, ch.4, p.251):

$$U = R_x - \frac{m(m+1)}{2} \tag{4.1}$$

$$U' = R_y - \frac{n(n+1)}{2} \tag{4.2}$$

Small values for R_x indicate a small total ranking of sample X, indicating also a low U value. Same as for R_y; small values lead to small values for U'. In the considered data set retrieving the rank sum statistic is more straightforward than the Mann-Whitney U test statistic. Consequently, the Wilcoxon rank sum statistics are considered for further analysis.

The null distribution of R_x needs to be determined. According to Gibbons & Chakraborti (2003, p. 299) for m, n large and under the continuous assumption, R_x follows an approximately normal distribution under H_0 (called, large sample approximation) with mean and variance :

$$E[R_x] = \frac{m(N+1)}{2} \tag{4.3}$$

$$Var(R_x) = \left(\frac{m(N-m)}{N-1}\right)\left(\frac{N^2-1}{12}\right) = \frac{mn(N+1)}{12} \tag{4.4}$$

Given the null distribution and the given rank sum statistic the p-value can be calculated. However, the assumption of a normal null distribution is not guaranteed by the considered data set. In some cases m, n is not necessarily large. Moreover, the data set suffers from ties, caused by the excessive amount of zeros. As a solution, the null distribution is simulated which is explained in the next chapter.

4.1.2. Application of Wilcoxon rank sum test

For each configuration that consists of two levels, the rank sums are calculated according to algorithm 4. Where the USD times of one level correspond to sample X and the USD times of the other level correspond to sample Y. The two samples X and Y are assumed to be mutually independent, that is the observations between groups and within groups should be independent. This is assumed to be the case, i.e. assuming that each machine number is unique and has its own properties. Table 4.1 shows the obtained rank sums.

To decide whether the R_x is small enough, the null distribution needs to be determined. According to the theory, R_x follows a normal distribution since m and n are large with the expectation and variance given in equations 4.3 and 4.4. However, this is under the continuous assumption. The data considered are not continuous because of the presence of *ties*. These are caused by the high amount of zeros. As a consequence, the null distribution of R_x is affected. Hollander, Wolfe and Chicken (2014, ch.4, p.118) suggest a modified variance and modified test statistic to overcome the problem of ties. Another direct method is suggested by simulating the exact distribution while using average ranking (Pratt & Gibbons, 1981, p.259). This last method will be applied, since then the exact distribution is simulated which is considered as more precisely. For the ace of simplicity only R_x is considered.

Feature	Levels	Ranksums
FieldE	FieldE-Avg	$R_x = 16189$
	FieldE-Local	$R_y = 12973$
AG	AG-present	$R_x = 12610$
	AG-absent	$R_y = 16551$
FSMFlex	FSMFlex-En	$R_x = 12173$
	FSMFlex-Disabled	$R_y = 16989$
iFSM	iFSM-En	$R_x = 28430$
	iFSM-Dis	$R_y = 731.5$

Table 4.1: The rank sums of the USD times per group for each configuration

The ties are handled using the method of *average ranking*. This technique assigns the rank of each of the tied values to their average ranking per tied subset. Now, under the null hypothesis, all arrangements of the rankings of m X's and n Y's are equally likely. This gives in total $\binom{N}{m}$ possible arrangements each with probability $\frac{1}{\binom{N}{m}}$ to occur. Then obtaining the null distribution is to list all possible arrangements and calculate for each arrangement the rank sum R_x. The chance of that R_x is then given by

$$P(R_x = z) = \frac{v(z)}{\binom{N}{m}} \tag{4.5}$$

Where $v(z)$ is the number of arrangements and where $R_x = z$. Since the number of all possible arrangements is very large, permutation is used. Permutating $10000 \cdot N$ times results in a null distribution looking like the normal distribution. In case the test statistic R_y is preferred, the null distribution can be obtained in the same way where m is replaced by n.

For each configuration the null distribution of R_x is simulated. In appendix C.1 the null distribution for each configuration is shown. This helps in deciding if R_x is significantly small enough (or high enough) for $\alpha = 0.05$. The following configurations show significant differences between levels:

Configuration	Level with significantly lower total USD time	P-value
FieldE	Averaged	0.03
AG	Present	0.00
FSM Flex	Enabled	0.05
iFSM		0.08

Table 4.2: Two-level configurations and their p-value after applying the Wilcoxon rank sum test

The configuration with no significant difference between groups is iFSM. Although in table 4.1 the ranksum of iFSM disabled is clearly much lower, there is no significance for $\alpha = 0.05$. Even though the p-value is quite low (0.08). This can be explained by the high differences between the two sample sizes. Namely, $m = 233$ and $n = 8$. This is also the reason why there is no smooth null distribution, which is already predicted by Pratt & Gibbons (1981, ch.4, p. 260).

So we can reject H_0 for the significant configurations and we can conclude that the settings from table 4.2 are stochastically smaller than the opposite setting. For the feature iFSM no significant difference is found.

4.2. Kruskal-Wallis test

4.2.1. Theory Kruskal-Wallis test

The Kruskal-Wallis test is an extension of the Wilcoxon rank sum test but compares $k > 2$ levels. It assumes the samples to be mutually independent coming from continuous populations. The following

hypothesis is tested (Gibbons & Chakraborti, 2003, ch.10, p.353):

$$H_0 : F_1(x) = F_2(X) = ... = F_k(x) \ \forall x \tag{4.6}$$

$$H_1 : F_i(x) \neq F_j(x) \text{ for at least one i,j} \in \{1, ..., k\} \text{ for some } x \tag{4.7}$$

That is, at least two populations differ. Just like the rank sum test, the test statistic is based on the rank sum for each sample. The procedure for retrieving the test statistic is shown in algorithm 5 (Gibbons & Chakraborti, 2003, ch. 10, p.364-365, 368).

Algorithm 5 Kruskal-Wallis - test statistic algorithm

Input: K continuous sample sizes $X_i = \{X_1, ..., X_{n_i}\}$ for $i \in \{1, ..., K\}$
Output: Test-statistic H

1: Combine all X_i in one group getting total size $n_1 + n_2 + ... + n_K = N$
2: Rank the combined observations from smallest to largest resulting $\{R_1, ..., R_N\}$
3: Sum the ranks of the observations of each sample X_i resulting in R_i
4: Calculate test statistic H by

$$H = \frac{12}{N(N+1)} \sum_{i=1}^{k} \frac{R_i^2}{n_i} - 3(N+1) \tag{4.8}$$

For all n_i not very small, the test statistic H is approximately chi-square distributed with $k-1$ degrees of freedom under H_0. Then H_0 is rejected for $H \geq \chi^2_{\alpha,k=1}$ (Gibbons & Chakraborti, 2003, ch. 10, p. 366).

When the null hypothesis is rejected, it is of interest what pairs of samples are different. *Multiple comparisons procedure* can be used to compare any possible pair of groups $i, j \in \{1, ..., k\},\ i \neq j$ (Gibbons & Chakraborti, 2003, ch. 10, p. 367). Groups i and j are different when the difference in mean rank is high enough. That is

$$|\overline{R}_i - \overline{R}_j| \geq z^* \sqrt{\frac{N(N+1)}{12} \left(\frac{1}{n_i} + \frac{1}{n_j} \right)} \tag{4.9}$$

Where $z^* = z_\alpha$ is the upper quantile of the standard normal distribution and $\overline{R}_i = \frac{1}{n_i} R_i$ with R_i the rank sum (Hochberg & Tamhane, 1987, ch. 9, p. 245). Usually, α is the probability of rejecting falsely the null hypothesis for a test. However, performing multiple comparisons, namely $\binom{k}{2}$ comparisons, the type 1 error increases. Therefore the Dunn-Sidak correction is used. This uses a more strict α:

$$\alpha^* = \frac{1}{2} \left(1 - (1 - \alpha)^{\frac{1}{\binom{k}{2}}} \right) \tag{4.10}$$

This reduces the so called *familywise error rate*, which is the probability of making wrong inferences that a multiple comparison makes (Hochberg & Tamhane, 1987, ch. 1, p. 7). This problem arises since $\binom{k}{2}$ comparisons are done. The more comparisons are done, the more likely it is that a group differs from at least another group. For example, testing at a 5% significance level for one test, has a 5% chance of incorrectly rejecting H_0. Whereas performing 1000 tests, it is expected that 5% of 1000 will incorrectly reject H_0, i.e. rejecting falsely 50 tests. Therefore a more strict α level is needed which is done by using equation 4.10.

The multiple comparison procedure can also be graphically displayed. This consists of displaying the mean rank with the comparison interval for each sample which is based on the covariances between samples. The exact calculation is done by $\overline{R}_i \pm z^* \cdot W_i$ where W_i is calculated according to Hochberg & Tamhane (1987, ch.3, p. 98). Then the two mean ranks are considered significantly different from each other if and only if their comparison intervals are disjoint (Hochberg & Tamhane, 1987, ch.3, p. 96).

4.2.2. Application of Kruskal-Wallis test

The Kruskal-Wallis test is applied on the features that have more than two levels from *feature set 2*. These are `location`, `system type` and `type of air gauge`. For each of these configurations the null and alternative hypothesis are tested by calculating the test statistic. However, the assumption of continuous populations is not met due to the presence of ties. As a consequence, the test statistic H changes. Gibbons and Chakraborti (2003, ch. 10, p. 367) state that the correction for ties is done by dividing H through a correction factor and by using the method of average ranking. The corrected test statistic is now calculated by

$$H_c = \frac{H}{1 - \frac{\sum_{l=1}^{L} t_l(t_l^2-1)}{N(N^2-1)}} \tag{4.11}$$

Where the sum is over the L sets over the t tied values in the population. The considered data have one tied subset, namely the zeros, so $L = 1$ with $t = 152$.
Just like H, for all n_i not small H_c follows a chi-square distribution with $k - 1$ degrees of freedom (Lehmann, D'Abrera, 1975, ch.5, p.201). The following test statistics with corresponding p-values are obtained:

Feature	H_c	D.f.	P-value
System type	120.70	6	$3.48 \cdot 10^{-16}$
Location	84.92	47	$2.10 \cdot 10^{-8}$
Air gauge type	49.66	2	$1.65 \cdot 10^{-11}$

Table 4.3: Configurations tested on Kruskal-Wallis assuming the test statistic follows a chi-square distribution

From the table it follows that in all three features at least one pair of group is significantly different. Now it is of interest what groups significantly differ in mean rank. Using equation 4.9 a significant difference is found among the following groups:

System type	NXT1950	-	NXT1965
	NXT1950	-	NXT1970
	NXT1960	-	NXT1965
	NXT1960	-	NXT1970
	NXT1965	-	NXT2
	NXT1970	-	NXT1980Di
	NXT1970	-	NXT2

Confidential table

AG type	Type 1	-	Type 2

Table 4.4: The pair of groups which are significantly different in ranked USD time

Note, the null distribution of H_c assumes that the sample size n_i is not small. However, this cannot be guaranteed. A possible solution can be simulating the distribution, like this is done in the Wilcoxon test. However, the method which displays the mean ranks graphically does not assume any distribution and is therefore also considered as a solution. This is done in the upcoming text.

Now a graphical display is applied where the mean ranks with comparison intervals are shown. Figure

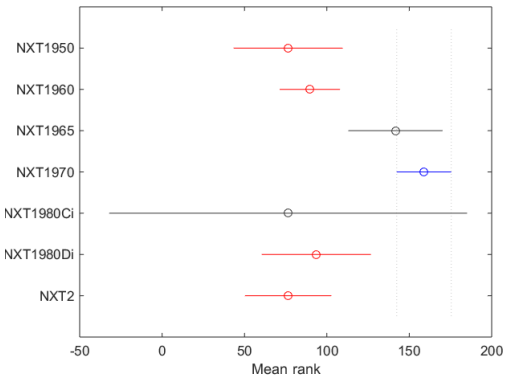


Figure 4.1: Graphically multiple comparison of **`system type`** displaying the mean rank and comparison interval for each level

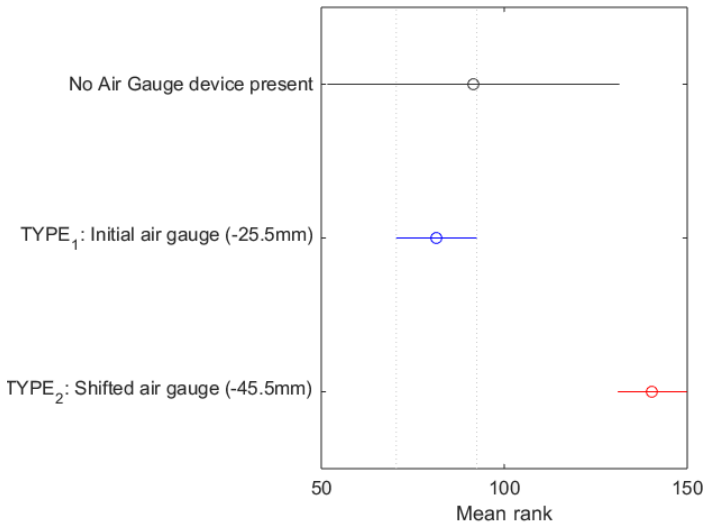


Figure 4.2: Graphically multiple comparison of **`AG type`** displaying the mean rank and comparison interval for each level

4.1 and table 4.4 corresponding to **`system type`** show the same significantly different groups. The graph shows that NXT1965 and NXT1970 have a higher mean ranking than the others. A long bar, which is shown for NXT1980Ci, indicates that little data are available for this specific system type.
About **`AG type`**, the table and graph show the same significant groups: type 1 and type 2, from which type 2 has a high mean rank.

4.3. Results

Combining the results of both the Wilcoxon rank sum test and Kruskal-Wallis test show the features listed in table 4.5 where at least one pair of group has significant difference in USD time of error B.
The configuration **`iFSM`** does not appear in the table because of the lack of significance. Furthermore, from these configurations it appears that **`fieldE-Avg`**, **`FSMFlex-Enabled`** and **`AG-Enabled`** show a lower USD time compared to their counterparts. Moreover, **`AG type 1`** seems to have a lower USD time than

Significant feature
System type
Location
FieldE
AG
AG Type
FSMFlex

Table 4.5: Configurations where at least one pair of levels within that configuration differs from each other with respect to the ranked USD time

`type 2`. By domain knowledge this could be hypothesized by the introduction of the new UV Level Sensor (UVLS) which caused some difficulties in the beginning. Namely, type 1 is applied on the VISLS while type 2 is used for UVLS. The higher mean rank of the `NXT1970` with respect to the others could be explained by the same hypothesis. The `NXT1980` update had solved some issues already.

4.4. Conclusion

The features that are significant appear to be a relevant contributor on their own to the discrepancy in USD time. The feature iFSM seems to be a less important contributor. Although, it is possible that it will become an important contributor when combining it with other features. However, this is not expected because of the small amount of discrepancy it can make: only nine out of 241 observations contain the level 'Disabled'. Therefore this feature is removed from the feature set. The following features are left over and will be called *feature set 3*.

Feature	Abbreviation
System type	MachType
Location	Loc
Leveling Field Extensions Algorithm	FieldE
Air Gauge	AG
Type of Air Gauge	AG Type
FSM Flexibility package	FSMFlex

Table 4.6: Feature set 3

5

Feature selection using predictive models

In the previous chapter each leveling configuration is tested separately on coming from the same USD population or not based on rankings. This indicates that it matters what level a machine has. However, no statements were made about the strength of how much it would matter.
In order to understand in what way and to what extend a specific level relate to the USD time, *regression analysis* is used, where the USD time is the *dependent variable* and the configurations are the *independent variables*. Besides discovering the strength, regression analysis gives insight in how the options relate to each other and how they act combined. This helps understanding the relationship between the features and the USD time of error B. In this way the most influencing configurations can be discovered. Besides the advantage of joint modeling, regression analysis can be used for prediction. Using prediction and joint modeling, a crucial combination of configurations is attempted to be found.

In the next chapter, data consisting of 241 data points, are pre-processed such that the models can be applied. Then the often used linear regression model with least squares is discussed. As an improvement of the linear regression model, a generalized linear model is considered in the chapter thereafter. Finally, a zero adjusted model is applied to overcome the problem of the inflation of zeros. For each of the three models a model selection is done and a best combination of features is chosen and evaluated. In the last chapter all three models are compared.

5.1. Pre-processing data

Before applying regression models, the independent categorical variables need to be transformed into dummy variables, i.e. $X_j = I(G = j)$. Each feature of feature set 3 with k levels is transformed into $(k-1)$ dummies where the last level is the reference variable. In total p dummy variables are left.

Since by domain knowledge it is expected that both system type and location could be relevant, these should also be taken into the model. However, the multiple levels they contain cause that we only have a few data points per level. Therefore the levels are merged even further. Although this merging leads to a loss of information, we would like to have enough data per level to apply tests on and to draw conclusion. They are merged as follows: `NXT1950, NXT1960, NXT1965, NXT2` are merged into `machtype-Old` and `NXT1980Ci` and `NXT1980Di` into `machtype-New`; `NXT1970` stays the same.
For location the following levels are merged: all factories belonging to C1 are merged to the level `loc-C1`, the same applies for C2 and C3. All the other locations belong to the variable `loc-Rest`.

In table 5.1 a list is shown of the used dummy variables and the corresponding reference variable.

Dummy variable	Reference variable
FieldE-Average	FieldE-Local
AG-Absent	AG-Present
AG type-1 AG type-2	AG type-No
FSMFlex-Disabled	FSMFlex-Enabled
Machine type-Old Machine type-1970	Machine type-New
Loc-C1 Loc-C2 Loc-C3	Loc-Rest

Table 5.1: Transformed dummy variables and reference variable for each feature from feature set 3 that is used for the regression models

5.2. Linear model and least squares

Usually, a lot of people apply the linear regression model. This model assumes that each observation $\{y_i, x_i\}_{i=1}^n$ can be modeled by a linear function of predictors $X^T = (X_1, X_2, ..., X_p)$ as follows:

$$y_i = \beta_0 + \sum_{j=1}^{p} X_{i_j}\beta_j + \epsilon_i \tag{5.1}$$

Where β_0 the intercept, β_j the estimated coefficient for parameter j and ϵ_i the unobserved term for observation i. In the current data we do not expect that the variables are linearly related to the dependent variable. However, we are going to run it anyway as a comparison for the other models in the further sections and to see the limitations of the model. Equation 5.1 can also be written in matrix form by:

$$Y = \mathbf{X}\beta + \epsilon \tag{5.2}$$

Where Y is an n-vector of outputs, $\mathbf{X}$ a $n \times (p+1)$ matrix where the 241 rows are the observations and each column a parameter p. A column of 1 is added, representing the intercept.

The covariates $X_1, .., X_p$ are the leveling configurations which are dummy coded. The desired response variable Y represents the USD time. The linear model assumes that the response variable is normally distributed. However, the USD times are highly skewed distributed and the GoF test shows that the USD times are not likely to be normally distributed with a $p-$value of $3.50 \cdot 10^{-12}$. Therefore a log-transformation is done in order to spread out the low values more and the high values less.

5.2.1. Data transformation

A constant c is added to the USD time to ensure a positive input for the log function:

$$Y = \ln(\underbrace{c + USD}_{Z}) \tag{5.3}$$

Now a c should be chosen. For $0 < Z \leq 1$ the ln function has a higher slope than for $Z > 1$. So for a small increase of Z where $Z \in (0, 1]$ leads to a higher increase in Y. Since it is desired to spread out the distribution most; are the values of $c \leq 1$ considered. The histograms are shown in figure 5.1.
The figures show that the peak of zeros and right-skewness is still present for every c. Moreover, the GoF-test shows that none of the log-transformation is likely to be normal. However, since we would like to perform the linear regression model we need to choose a transformation. Hence, quantifications such as QQ-plots, kurtosis and skewness are compared for every c value in order to choose a value c which comes closest to normality. The kurtosis is a measure of 'peakness' and skewness is a measure for asymmetry. They have the values shown table 5.2.

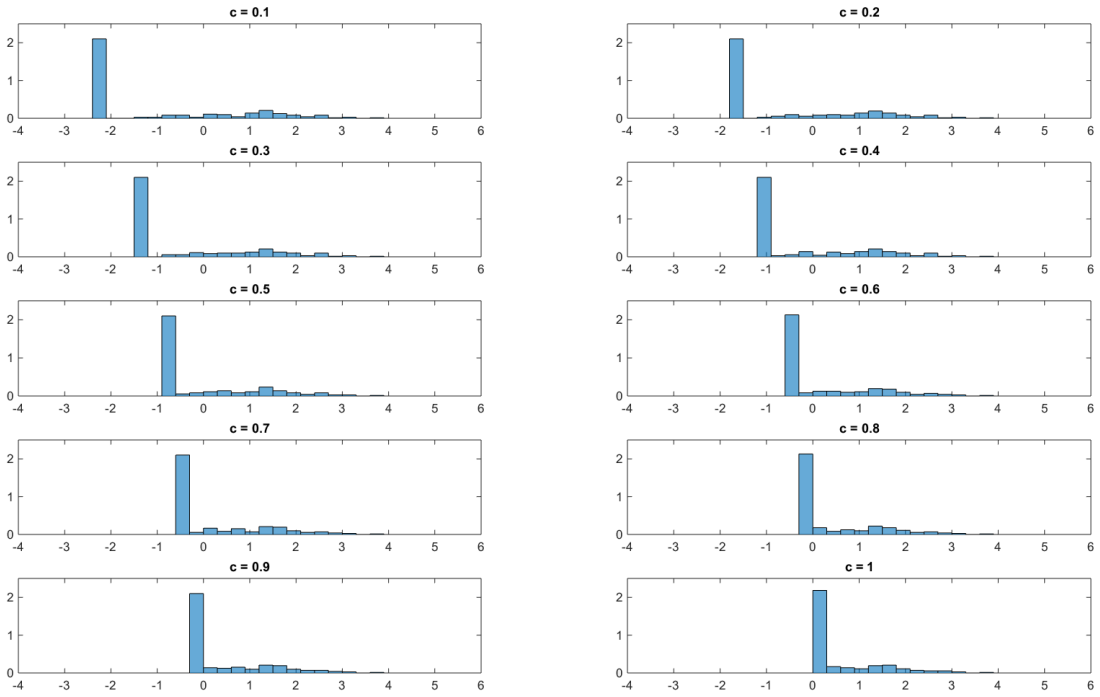


Figure 5.1: Histograms belonging to the log transformed USD times for each constant c

$$k = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^4}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2\right)^2}$$

$$s = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^3}{\left(\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2}\right)^3}$$

c	Kurtosis	Skewness
0.1	2.41	0.98
0.2	2.78	1.10
0.3	3.07	1.19
0.4	3.33	1.26
0.5	3.56	1.32
0.6	3.77	1.38
0.7	3.97	1.43
0.8	4.16	1.47
0.9	4.34	1.51
1	4.51	1.55

Table 5.2: For each c-value the kurtosis and skewness is calculated

The kurtosis increases as c increases. A normal distribution has a kurtosis of 3, where $c = 0.3$ is the closest to that. Considering the skewness, a value of 0 is desirable, indicating no asymmetry. The lower c, the lower the skewness. In figure 5.2 the QQ-plots are shown. In the QQ-plots are the Y values standardized to mean 0 and variance of 1. Then the quantiles of these values are compared with the standard normal quantiles. The red line corresponds to the positions of the data if they followed a standard normal distribution. Now quantifying the QQ-plot by looking at the sum of squared residuals (SSR): $\sum_i^n (Y_{standardized}(x_i) - Y_{redline}(x_i))^2$ it follows that $c = 0.1$ has the lowest SSR.

Looking at the QQ-plot and the skewness, $c = 0.1$ comes closer to normality than the other c-values. However, using the GoF-test, this transformation is not normal. Now choosing $c = 0.1$, the Y is calculated by equation 5.3 and is applied in the regression model for estimating the coefficients.

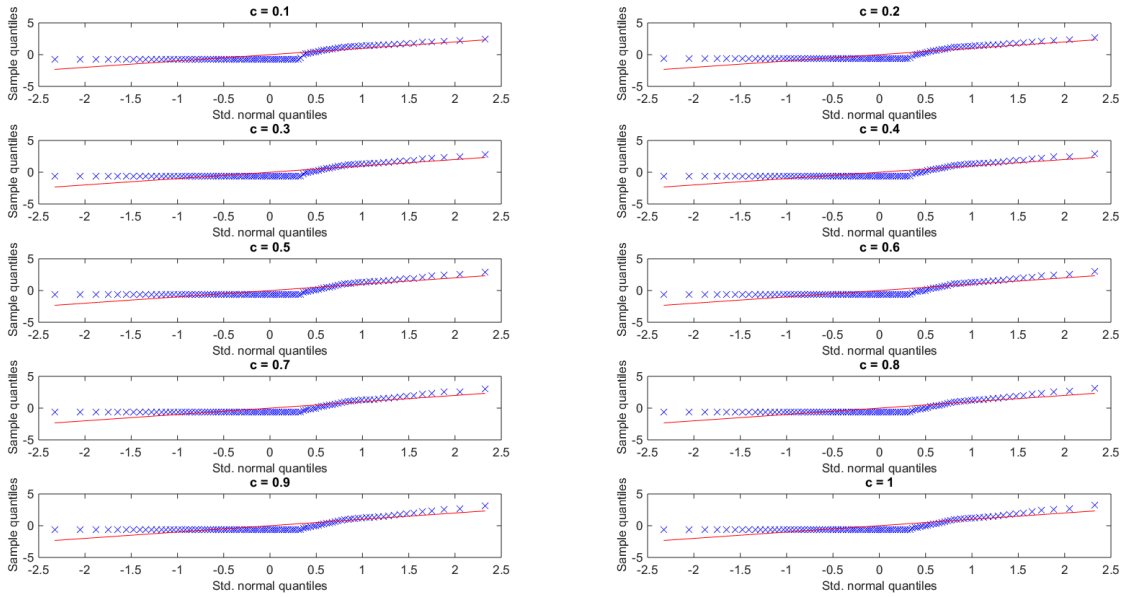


Figure 5.2: QQ-plots of the standard normal distribution and the distribution of the log transformed data for each constant c

5.2.2. Estimation

The coefficients β_j for $j = 0, ..., p$ are estimated by the *least squares* method, i.e. the residual sum of squares (RSS) is minimized. The residual sum of squares (RSS) is defined by

$$RSS(\beta) = \sum_{i=1}^{N} \epsilon_i^2 \tag{5.4}$$

$$= \epsilon^T \epsilon \tag{5.5}$$

$$= (Y - \mathbf{X}\beta)^T (Y - \mathbf{X}\beta) \tag{5.6}$$

$$= Y^T Y - 2\beta^T \mathbf{X}^T Y + \beta^T \mathbf{X}^T \mathbf{X}\beta \tag{5.7}$$

$$\tag{5.8}$$

Differentiating with respect to β gives:

$$\frac{\delta RSS}{\delta \beta} = -2\mathbf{X}^T y + 2\mathbf{X}^T \mathbf{X}\beta$$

$$\frac{\delta^2 RSS}{\delta \beta \delta \beta^T} = 2\mathbf{X}^T \mathbf{X}$$

The second order derivative should be positive as a condition for a minimum, this is the case if $\mathbf{X}$ has full column rank and hence $\mathbf{X}^T\mathbf{X}$ is positive definite (Hastie, Tibshirani & Friedman, 2008, ch. 3, p. 45). The β's can be found by differentating and by setting it equal to 0:

$$\frac{\delta RSS}{\delta \beta} = -2\mathbf{X}^T y + 2\mathbf{X}^T \mathbf{X}\beta = 0$$

$$\Leftrightarrow \mathbf{X}^T \mathbf{X}\beta = \mathbf{X}^T Y$$

$$\Leftrightarrow \beta = \left(\mathbf{X}^T \mathbf{X}\right)^{-1} \mathbf{X}^T Y$$

Its minimum always exists and is unique in case the matrix has full rank, meaning that the columns of X are linearly independent (Hastie, Tibshirani & Friedman, 2008, ch. 3, p. 46). In the used data, the design matrix $\mathbf{X}$ has no column dependencies and has full column rank, so a unique minimum is found. Now the set of observations $(x_1, y_1), ..., (x_n, y_n)$ are used to estimate the parameters $\hat{\beta}$, where

each $x_i = (x_{i1}, x_{i2}, ..., x_{ip})^T$is a vector of measurements for the p parameters for $i \in \{1, .., n\}$.
According to Hastie et. al (2008, ch.3, p.44) RSS is a plausible criterion if the data (x_i, y_i) are independent random draws or if the y_i's given the x_i's are, even though the x_i's are not independent. Assuming the observations are independent, the coefficients can be estimated from the input variables using RSS. Then the following model is estimated:

$$\hat{Y}_i = \beta_0 + \sum_{j=1}^{n} X_{ij}\beta_j \tag{5.9}$$

Besides the single variables, also interaction terms can be included. However, as discussed in later stage, this leads to an high amount of insignificant p-values or p-values which were not found since that interaction did not exist in the data. Now assuming, the Y follows normality and has a linear relationship described in equation 5.1, we get the following coefficient estimates where all dummy variables are used as input.

Term	Coefficient	Std. Error	P-value
Intercept	-3.34	0.62	$1.97 \cdot 10^{-7}$
FieldE-Average	0.09	0.19	0.96
AG-Absent	0.14	0.25	0.58
AG type-1	0.27	0.46	0.56
AG type-2	1.85	0.47	$1.22 \cdot 10^{-4}$
FSMFlex-Disabled	0.82	0.26	$1.77 \cdot 10^{-3}$
Machine type - Old	1.01	0.38	0.01
Machine type - 1970	1.58	0.33	$3.56 \cdot 10^{-6}$
Loc - C1	-2.78	0.35	$7.78 \cdot 10^{-14}$
Loc - C2	-1.00	0.24	$3.16 \cdot 10^{-5}$
Loc - C3	-0.47	0.30	0.12

Table 5.3: Full linear regression model using least squares method

The intercept is the value where the fitted linear line crosses the y-axis. Intercept, **`AG type-2`**, **`FSMFlex-Disabled`**, **`machine type - Old`**, **`machine type - 1970`**, **`location C1`** and **`location C2`** are significant variables. From those variables, having **`AG type-2`**, **`FSMFlex-Disabled`**, **`machine type - Old`**, **`machine type - 1970`**, leads each to an increase to the fitted values $\hat{Y}$ and $\hat{USD}$, since

$$\hat{Y}_i = \ln(\hat{USD}_i + 0.1) \Leftrightarrow \hat{USD}_i = e^{\hat{Y}_i} - 0.1 \tag{5.10}$$

The intercept has a negative estimate. The same counts for **`C1`** and **`C2`**. **`FieldE-Average`**, **`AG-Absent`**, **`AG type 1`**, and **`loc-C3`** have insignificant coefficients. This can be the case since the variable will not add significantly much information and that one of the variables is correlated with one of the significant variables, and consequently do not add significantly information comparing to the other correlated variable. **`FieldE-Avg`** has no Pearson correlation higher than 0.50 with the other variables. **`AG-Abs`** has two correlations of 0.50 with **`loc-C2`** and -0.50 with **`loc-C3`**. Hence both variables are not strongly correlated with the significant ones. **`AG type 1`** however has a strong correlation of -0.92 with **`AG-type 2`**. This explains why **`AG-type 1`** is insignificant. Moreover, **`loc-C3`** has a significant correlation of -0.59 with **`FSMFlex-Dis`** which can explain the slight insignificance of C3 of 0.12.

The proportion of variance that is explained by the model is $R^2 = 0.524$. This means that 52.4% of the variance of the USD times can be explained by the data. Hence, the data has not enough prediction capability.
In the above model (the full model) all variables are used. However, because of the well-known bias-variance trade off, a subset of variable may be more convenient.

5.2.3. Model selection

The expected squared prediction error of the model is decomposed by bias and a variance.

$$E\left[\left(y - \hat{f}(x)\right)^2\right] = \text{Bias}\left[\hat{f}(x)\right]^2 + \text{Var}\left[\hat{f}(x)\right] + \sigma^2$$

In minimizing both, a trade off is made. As model complexity grows, and more and more parameters are added to the model, bias is reduced and the variance is increased. In the full model, all parameters are used such that the bias is minimized. However, it is also desired to minimize the variance. Therefore a subset of variables is preferred to improve prediction accuracy. Moreover, it will improve the interpretation of the model. Namely with a smaller amount of covariates the strongest effects are modeled at the cost of smaller details (Hastie, Tibshirani & Friedman, 2008, ch. 3, p. 57).

To find the best subset, the *best-subset selection* approach is used. This method fits a linear regression model for every possible subset of variables (Hastie, Tibshirani & Friedman, 2008, ch. 3, p. 57). For every model both the complexity and the bias are assessed using *Akaike Information criterion (AIC)*. This criterion considers the likelihood and gives a penalty of two on the number of estimated parameters:

$$AIC = 2a - 2ln(\hat{l}) \tag{5.11}$$

Where $\hat{l}$ is the maximized likelihood, $a = p + 2$, i.e. the number of estimated parameters which is the number of covariates plus the intercept and the variance of the error term.

The AIC has a penalty of two, however this penalty could also be chosen differently. Setting a heavier penalty, could result in a selected model with less parameters, and hence in a simpler model. Setting a less heavier penalty, could result in a selected model with more parameters. Note, the well-known Bayesian information criterion (BIC) has a penalty of $\log(n)$. This is a heavier penalty than two for $n = 241$. Hastie et. al. (2008, ch. 7, p. 235) state that BIC often chooses too simple models when having finite samples. Therefore, we choose the classical AIC with a penalty of two.

As can be seen in appendix D, the AIC can be defined as:

$$AIC = 2a + n \ln \frac{RSS}{n} \tag{5.12}$$

So the RSS should be as low as possible with a penalty on the number of parameters.

Assuming we met the normality and linearity assumption of the model, a best subset p is found where AIC is minimized. All possible subsets are fitted, where only all $(k-1)$ dummies per categorical variable are included or all $(k-1)$ dummies are excluded. For each model the AIC is calculated and a line is drawn through the lowest AIC for each subset p. We get the values shown in figure 5.3.

Also for the models including interactions a best subset selection is done. However, the lowest AIC value is 1319 with $R^2 = 0.42$. Moreover, in this model are 17 out of 25 coefficients insignificant or NaN. A possible reason would be the small amount of data: not every possible combination exist in the data or a combination occurs not-frequently. Hence when mostly $X_i \cdot X_j = X_i$ for $i, j \in 1, .., p,\ i \neq j$, we get dependencies between the interaction variables and the singular variables. Besides the increasing number of coefficients of insignificance, the overview will get worse. Still an amount of 25 coefficients are present which gives not a straightforward and quickly overview. For those two reasons, the interactions are not considered.

The model with $p = 8$ has the lowest AIC value of 788.14. After $p = 8$ the AIC value is slowly increasing again. A horizontal line can be seen between $p = 3$ and $p = 4$. In this last model the variable **`FSMFlex-Disabled`** is added to **`AG-Absent`** and **`system type`**. Apparently, adding this single variable does not contribute to a much lower RSS that it outweighs the penalty of the extra parameter. In $p = 5$ a high decrease can be derived from the figure. In here **`FSMFlex-Disabled`** and **`AG-Absent`** are replaced by **`location`**, where **`machine type`** is still present. It appears that **`location`** is a more important complement to **`system type`** in reducing RSS than **`AG`** and **`FSMFlex`** are. Looking at all models, it seems that **`location`**, **`system type`** and **`AG type`** are the most important contributors in predicting USD time. **`FieldE`** mostly occurs in the models with the highest AIC, and consequently has less predictive capability. **`AG`** and **`FSMFlex`** seem to be in the middle.

The coefficient estimates for model with $p = 8$ are shown in table 5.4. Note, recall that the linearity assumption is not met and hence the slopes and intercepts may not be reliable. However, for

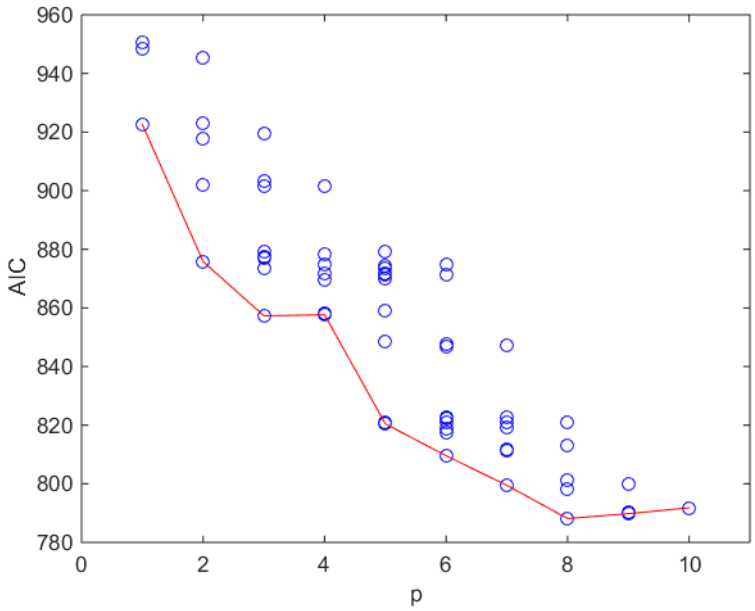


Figure 5.3: AIC against the number of variables p

Term	Coefficient	Std. error	P-value
Intercept	-3.23	0.58	$8.52 \cdot 10^{-8}$
AG type-1	0.22	0.45	0.63
AG type-2	1.81	0.46	$1.28 \cdot 10^{-4}$
FSMFlex-Disabled	0.87	0.24	$3.49 \cdot 10^{-4}$
Machine type-Old	1.00	0.37	$7.87 \cdot 10^{-3}$
Machine type-1970	1.58	0.33	$2.98 \cdot 10^{-6}$
Loc-C1	-2.88	0.29	$2.24 \cdot 10^{-19}$
Loc-C2	-0.99	0.23	$3.02 \cdot 10^{-5}$
Loc-C3	-0.52	0.28	0.06

Table 5.4: Selected model with $p = 8$ using AIC

comparison reasons we run the model.

The variables **`AG type`**, **`FSMFlex`**, **`machine type`** and **`location`** are the variables with the strongest effects, since they appear in the model with lowest AIC. **`FieldE`** and **`AG`** will add not that more extra information. **`AG type 1`** is insignificant, which could be explained by the correlation between type 1 and type 2 as shown in table E.1 in appendix E. **`Location-C3`** is close to significance at the 5% level. From the significant terms, **`AG type-2`**, **`FSMFlex-Disabled`**, **`machine type-Old`** and **`1970`** will let the response variable $\hat{Y}$ increase.

5.2.4. Model assessment

Now a model is selected. We expect that the relationship of the variables is not linearly related to the dependent variable. Moreover, the dependent variable is not likely to be normal. Let us look how these two violations of the assumptions is reflected in the residuals.

Using the GoF-test, the distribution of the residuals is significantly not normally distributed with a p-value of $1.01 \cdot 10^{-21}$. This effects the calculation of the p-value and standard error of the estimates (Hastie et al., 2008, ch. 3, p. 47-48) and may be incorrect.
Further, the residual values should be randomly distributed around zero and should not contain any predictive information. Figure 5.5a shows how the residuals would look like if Y$\sim$ $N(X^T\beta, \hat{\sigma})$. In that figure the residuals are randomly distributed and some vertical lines can be seen. However, looking

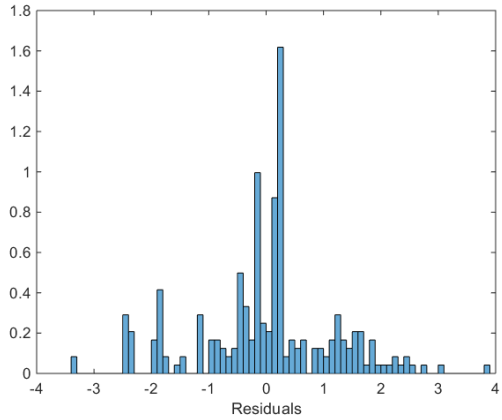
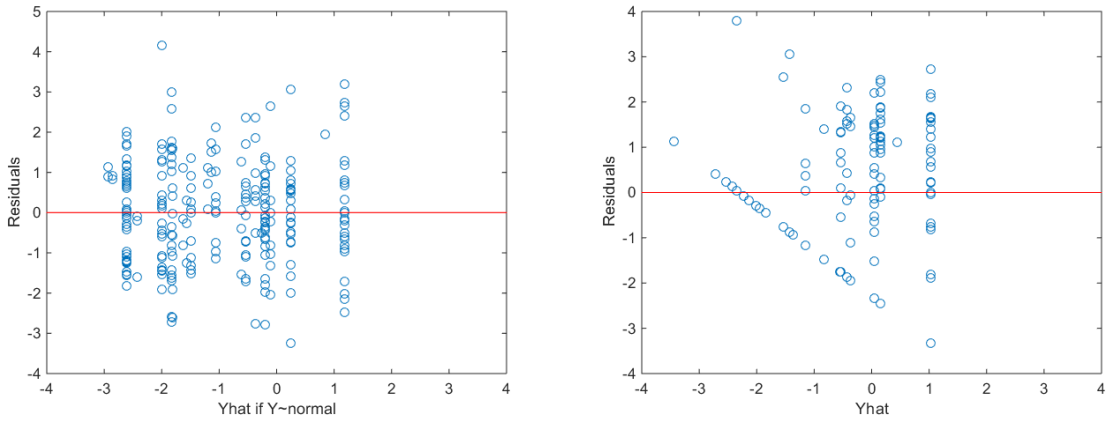


Figure 5.4: Distribution of the residuals



(a) Residuals against $\hat{Y}$ if $Y \sim N(X^T\beta, \hat{\sigma}^2)$ **(b)** Residuals against $\hat{Y}$

Figure 5.5: Residual plots

at figure 5.5b, a systematic pattern can be derived from the plot; the residual value increases as the fitted value increases. This could indicate *heteroscedasticity*, meaning that the variance differs across the observations and that it is not constant, as the linear model assumes. This can be explained when a specific combination of dummy variables has some high USD times but also a lot of zeros. Then for that specific combination a too high value is estimated (for negative residuals) or a too low value is estimated (for positive residuals). A reason for this pattern can be a missing variable which captures this pattern. Another reason is that a linear relationship is not the correct link between the response and predictors, which was already expected in this data. As a consequence, just like in the non-normality case, the underlying hypothesis tests cannot be relied on and standard errors and p-values may not be correct.

Although we see that this linear model is not a correct model for the log transformed data and the estimates and p-values are likely not to be reliable, we would like to check the predictive properties of the model in order to compare this with the models in the upcoming sections. The model has a value of $R^2 = 0.523$. The R^2 value shows that only about the half of the total variation of Y is explained by the above variables, assuming the model would be correct. This indicates that these variables alone may not be suited for prediction.

Further, $U\hat{S}D$ is not higher than 2.7. The possible combinations of variables are limited hence, the possible predicted values of $U\hat{S}D$ are also limited. Moreover, some $U\hat{S}D$ values are negative, this happens 62 times out of 241. Lastly, the residual sum of squares of the USD and estimated USD is RSS

= 4122.88.

5.2.5. Conclusion

In conclusion, the data do not meet the assumptions of linearity, normality and constant variance. These effect the performance. The intercept and coefficient estimates with the corresponding p-values may be incorrect and hence the prediction is not right. Assuming we would meet these assumptions, then **`AG type`**, **`FSM Flex`**, **`machine type`** and **`location`** would be the variables with the strongest effects. However, we would still not have sufficient predictive power; negative values are predicted and only half of the total variation is explained by the model by 52.3%. In the next chapter the generalized linear model is considered, which is an extension of the linear model and that can account for heteroskedasticity.

5.3. Generalized Linear Model

The advantage of the generalized linear model (GLM) with respect to the linear model (LM) is that it allows the error distribution to be different than normal. The distribution can be any type of the exponential family:

$$f(y;\theta,\phi) = exp\left\{\frac{y\theta - b(\theta)}{a(\phi) + c(y,\phi)}\right\}$$

For some specific functions a(.), b(.) and c(.), dependent on which specific exponential family distribution, ϕ the dispersion parameter and θ the important parameter, which is also called the *canonical parameter* (McCullagh & Nelder, 1989, ch.2, p. 28). The regression model takes the form (McCullagh & Nelder, 1989, ch.2, p.27):

$$g(E(Y_i)) = \eta_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{j_i} \Leftrightarrow E(Y_i) = g^{-1}\left(\beta_0 + \sum_{j=1}^{p} \beta_j x_{j_i}\right)$$

Where g is called the *link function*.
First, the distribution of the response variable Y needs to be specified (De Jong & Heller, 2008, ch.5, p.65). Looking at the USD time distribution, where a small constant is added to ensure strictly positivity, the Gamma or inverse Gaussian would be possible. Nevertheless, they both suffer from a lack of fit caused by the excessive amount of zeros which can be seen in figure F.1, appendix F. Therefore the USD times are log-transformed again, where a small constant is added to the USD time. Possible fitted distributions are Gamma or inverse Gaussian. Now a value of $c = 0.1$ is not possible since Y should be greater than 0. Therefore $c = 1.1$:

$$Y = \log(USD + 1.1)$$

Testing all three distributions on the data using the GoF-test, none of the distributions fits the data significantly well. However, since applying the GLM is a well-known solution for non-normality and heteroskedasticy and since we would like to compare this model with the LM and upcoming model, we are going to apply it anyway. Therefore, we need to choose one distribution. Looking at figure 5.6, it looks like the inverse Gaussian distribution comes closest to the distribution of Y. While fitting all three distributions, it also appears that the inverse Gaussian has the highest log-likelihood compared to the others. Therefore, we choose the inverse Gaussian, even though it is not significantly a good fit to the data.

Secondly, a *link function* $g(.)$ needs to be defined. The link function describes the relationship between the linear combination of the covariates and the mean. For this the inverse square is used, which is the canonical link of the inverse Gaussian (De Jong & Heller, 2008, ch.5, p. 67).

$$g(E(Y)) = g(\mu) = \frac{1}{\mu^2}$$

A link is canonical if the function expresses θ in μ, i.e. $g(\mu) = \theta = \mathbf{X}^T\beta$.

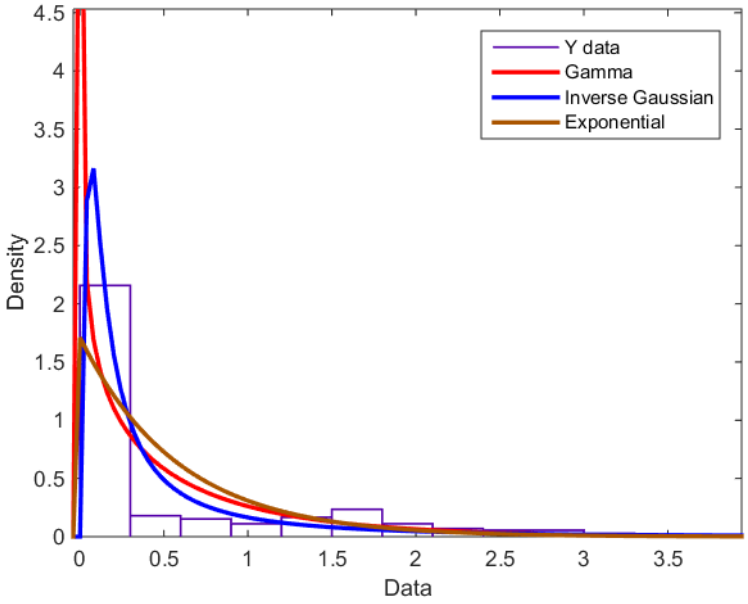


Figure 5.6: The distribution of Y is fitted for Gamma, inverse Gaussian and exponential distribution

5.3.1. Estimation

The parameters β_j are solved by maximum likelihood estimation (McCullagh & Nelder, 1989, ch.2, p.23-24). The likelihood and log-likelihood are defined by

$$l(\theta, \phi; y_1, y_2, ..., y_n) = \prod_{i=1}^{n} f(y_i; \theta, \phi) \quad \text{assuming independence} \tag{5.13}$$

$$= \prod_{i=1}^{n} e^{z_i} \quad \text{where } z_i = \frac{y_i\theta_i - b(\theta_i)}{a(\phi_i)} + c(y_i, \phi_i) \tag{5.14}$$

$$= e^{\sum z_i} \tag{5.15}$$

$$ll(\theta, \phi; y_i) = \sum_{i=1}^{n} z_i \tag{5.16}$$

$$= \sum_{i=1}^{n} \frac{y_i\theta_i - b(\theta_i)}{a(\phi_i)} + c(y_i, \phi_i) \tag{5.17}$$

Where we assume that the second derivative < 0. Now this is differentiated to β_j and is set equal to zero which then is solved by iterative weighted least squares method (Dobson & Barnett, 2008, ch. 4, p. 66). In appendix F.2 the derivative is obtained.

5.3.2. Model selection

Considering both bias and variance when selecting a model, the same principle applies as to the LM where AIC is defined in the same way as in equation 5.11 (De Jong & Heller, 2008, ch.5, p.80).
In order to calculate AIC, the log-likelihood needs to be determined first.
Equation 5.17 is used where $a(.), b(.), \theta$ are filled in according to the inverse Gaussian distribution. In appendix F.3 these functions and parameters are obtained. The following log-likelihood is obtained.

$$ll(\mu, \phi; y) = \sum_{i=1}^{n} \frac{y_i \left(\frac{1}{2}\mu_i^2\right) - \frac{1}{\mu_i}}{-\phi}$$

Where y_i are the observed response variable, μ_i are the fitted values and ϕ is the dispersion parameter. Note, the $c(y, \phi_i)$ value is not considered here. Now the AIC can be calculated for each possible subset

of variables, where for each categorical variable all $(k-1)$ dummies are included or all excluded, like is done in the LS method. Further, $a = 2 + p$ where p is the number of covariates and 2 for the intercept and the error term. Assuming that Y would follow an inverse Gaussian distribution, we obtain the following AIC values:

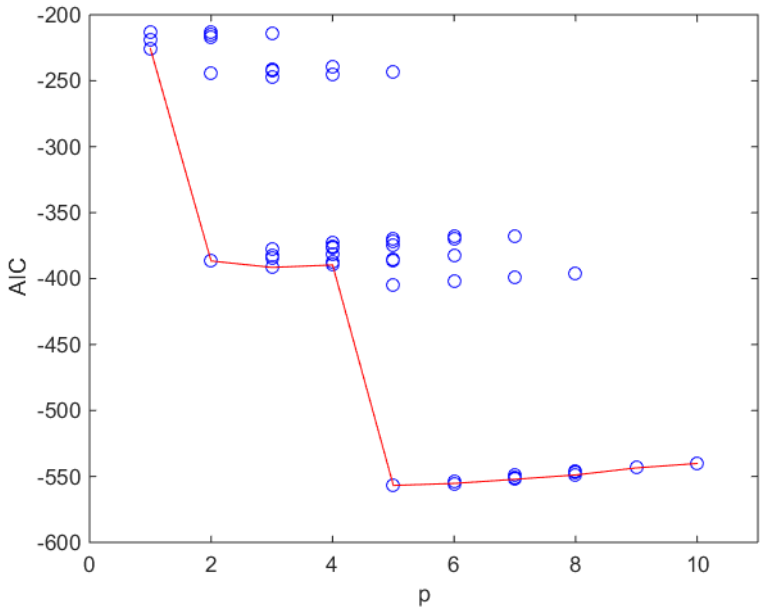


Figure 5.7: AIC against the number of variables p applying GLM

The line of AIC looks non-smooth. A similar horizontal line can be seen between $p = 3$ and $p = 4$ like in the LM, but now this line is extended to $p = 2$. Further, a high decrease can be seen between $p = 4$ and $p = 5$ just like in the LM.
What also strikes are the 2 gaps: one appears after $p \geq 2$ and the other appears for $p \geq 5$. It looks like 3 'rows' are present in the figure. In the first gap after $p \geq 2$, **`AG type`** can be added as well as **`system type`**. This was not possible for $p = 1$ since those consist of two levels and only all the levels are included or none of them. It appears that, **`AG type`** contributes to an increase of the log-likelihood, compared to **`fieldE`**, **`AG`**, **`FSMFlex`** or **`system type`** alone. For $p = 3$ **`location`** alone is used and appears to have an increase to the log-likelihood compared to **`fieldE`**, **`AG`**, **`FSMFlex`** or **`system type`**. All in all, the second 'row' is caused by the variables **`location`** or **`AG type`** which are added to **`fieldE`**, **`AG`**, **`FSMFlex`** or **`system type`**. The third 'row', for $p \geq 5$, is caused since now **`location`** can be combined with **`AG type`**. In the subsequent subsets p **`FSMFlex`**, **`fieldE`**, **`AG`** and/or **`system type`** are added to the combination of **`AG type`** and **`location`**.
Summarizing, if Y would follow an inverse Gaussian distribution, **`fieldE`**, **`AG`**, **`FSMFlex`** and **`system type`** seem to be less important contributors for reducing the likelihood. In contrast, **`AG type`** and **`location`** are the most important contributors. Especially, when using them in combination.

The model with the lowest AIC has the coefficient estimates shown in table 5.5. Note, since Y is not significantly inverse Gaussian distributed, the intercept and coefficient estimates may not be reliable. But since we would like to compare all the models, we still apply it.
As in the LM, **`AG type`** and **`location`** exist in the selected model and thereby representing the strongest effects. **`AG type-1`**, **`Loc-C1`** and **`Loc-C3`** have significant coefficient estimates. From these variables, all let the response variable $\hat{Y}$ decrease. Note, the coefficients do not have a direct result on the estimated

Term	Coefficient	Std. Error	P-value
Intercept	13.43	9.63	0.16
AG type-1	47.18	15.16	$2.1 \cdot 10^{-3}$
AG type-2	-12.83	9.63	0.18
Loc-C1	109.49	23.57	$5.64 \cdot 10^{-6}$
Loc-C2	1.03	0.99	0.30
Loc-C3	15.01	7.45	0.05

Table 5.5: Selected GLM with $p = 5$ using AIC

USD time. Since, $\hat{Y}$ and $\hat{USD}$ are calculated by:

$$\hat{Y}_i = \frac{1}{\sqrt{\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}}} \tag{5.18}$$

$$\hat{Y}_i = \ln(\hat{USD}_i + 1.1) \Leftrightarrow \hat{USD}_i = e^{\hat{Y}_i} - 1.1 \tag{5.19}$$

Where it is used that the inverse of $\frac{1}{\mu^2}$ is equal to $\frac{1}{\sqrt{\mu}}$. So one unit of change in the explanatory variable leads to an increase by the coefficient of $\ln(1.1 + USD)$ as can be seen in equation 5.19.

5.3.3. Model assessment

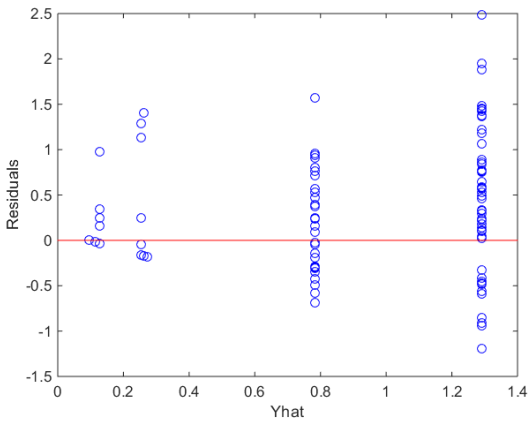
Assessing the GLM is a bit different than for the LM. It does not assume linearity, normality or a constant variance. It assumes the chosen response distribution, i.e. inverse Gaussian, and linearity between the independent variables and the transformed dependent variable. However as we have seen in chapter 5.3., we do not fulfill the assumption of the inverse Gaussian distribution. As a consequence the standard errors and corresponding p-values of the coefficients may be incorrect. Let us look how this is reflected in the residuals. McCullagh and Nelder (1989, ch.2, p.37) stated that an extended definition of residuals is needed. One type is the *deviance residual* δ_i.

$$\delta_i = \text{sign}(Y - \hat{Y}) \sqrt{\frac{Y_i - \hat{Y}_i}{\hat{Y}_i^2 Y}}$$

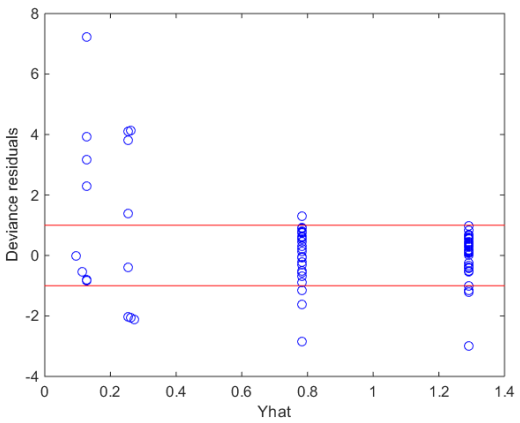
This assesses how much each residual contributes to the *deviance* of the model. The deviance is a measure of discrepancy: it shows how well the estimated $\hat{Y}$ fits the observed Y (McNullagh & Nelder, 1989, ch.2, p.39). A high value of deviance residual means that the corresponding fitted $\hat{Y}_i$ value does not fit the model well. De Jong & Heller (2008, ch.5, p.78) state that the model has a lack of fit for $|\delta_i| > 1$ and for n large. The value is based on the assumption that for n large, the deviance follows a chi-square distribution with $n - p$ d.f. and with expected value $(n - p)$. Each case is then expected to contribute $(n-p)/n \approx 1$ to the deviance. In figure **??** the absolute residuals and the deviance residuals are plotted.

From the deviance residual plot it follows that some fitted values suffer from a lack of fit: 64 out of 241 deviance residual values have an absolute value greater than one. This indicates that for those data points the inverse Gaussian GLM may not be the correct model. If $Y \sim IG(\mu_x, \lambda)$ with $\lambda = \frac{1}{\phi}$, then we would get the residual plots shown in figure 5.9.

This plot shows already a less spread of residuals of the lower predicted values, although the model would still not be correct, since 40 deviance residuals are greater or lower than 1 or -1, respectively. This indicates that we also violate another assumption; that there is no linearity between the independent variables and the transformed dependent variable. As a consequence, the intercept and coefficient estimates may be incorrect. Although we see that this GLM is not a correct model for the dependent variable Y and as a consequence, the estimates and corresponding p-values are not reliable, we would like to check the predictive properties for comparison with the LM and the upcoming model. The GLM model has a value of $R^2 = 0.407$. So less than the half of the total variation is explained by the model. This is 0.116 lower than for the LM. However, an improvement with respect to the LM are the number of negative predicted values: none are negative with respect to 62 in the LM. The residual sum of squares of USD and the estimated USD is RSS = 4011.24, which is an improvement of 111.64 with respect to the LM.

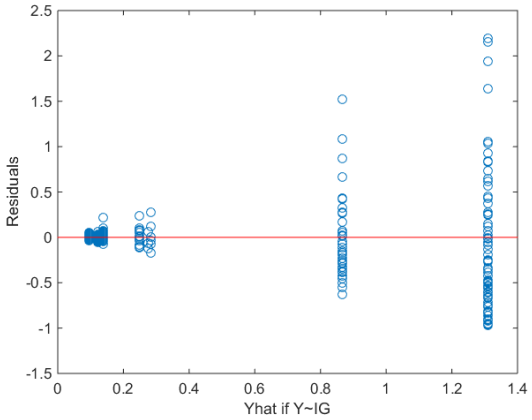


(a) Residuals against $\hat{Y}$ for the GLM

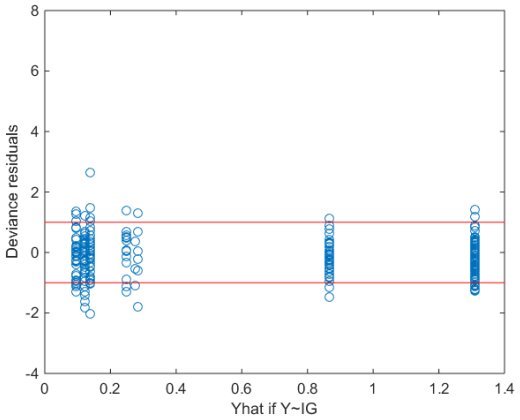


(b) Deviance residuals against $\hat{Y}$ for the GLM

Figure 5.8: Residual plots for GLM



(a) Residuals against $\hat{Y}$ if $Y \sim IG(\mu_x, \lambda)$



(b) Deviance residuals against $\hat{Y}$ if $Y \sim IG(\mu_x, \lambda)$

Figure 5.9: Residual plots if $Y \sim IG(\mu_x, \lambda)$

5.3.4. Conclusion

In conclusion, just like the LM the GLM model may not be the correct model for prediction. The data do not follow the inverse Gaussian distribution and there is no linearity between the independent variables and the transformed dependent variable. However, in this model we have no negative estimated USD times, which seems to be an improvement with respect to the LM. Another improvement is that this model accounts for heteroskedasticity.

Assuming we would have meet these assumptions, **`AG type`** and **`location`** seem to model the strongest effects. **`FieldE`**, **`AG`**, **`FSMFlex`** and **`system type`** represent the smaller effects. Especially, **`AG type`** together with **`location`** appear to be the most important contributor for modeling the USD time.

The excessive amount of zeros was modeled by the inverse Gaussian distribution, but appear not be correct. This indicates that the model can be improved. Therefore, a mixed distribution model is considered which is discussed in the next chapter.

5.4. Zero adjusted exponential model

The skewed distribution and inflation of zeros distort fitting a parametric distribution even after log transformation. Both the normality and inverse Gaussian distribution are not significantly a good fit to the data. Another way of dealing with this excessive amount is by applying a mixed distribution model.

We assume that the USD times can be divided into two groups: the first group has zero USD time and the second group has non-zero USD time which follow a continuous distribution. For $i = 1, .., n$ and response variable $y_i = USD_i$ has the mixed distribution model the following form:

$$f(y) = \begin{cases} p_0, & \text{if } \ y = 0 \\ (1 - p_0)g(y), & \text{if } \ y > 0 \end{cases} \tag{5.20}$$

Where p_0 is the probability of zero USD time and $g(y)$ the continuous density function for $y > 0$. Where $g(y)$ is chosen which fits the data best. In the following figure different possibilities of a suitable right-skewed density functions are displayed:

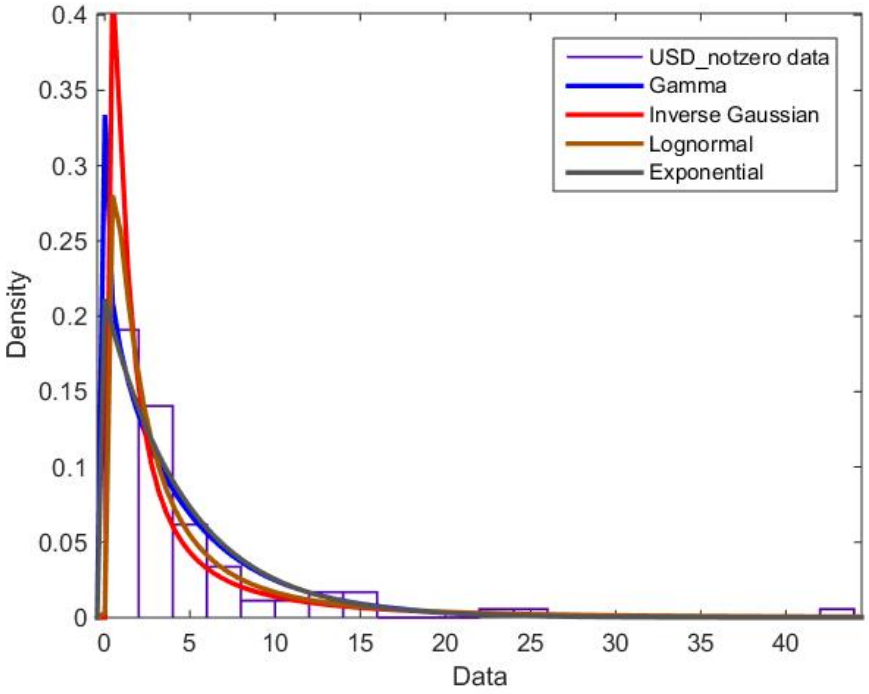


Figure 5.10: Fitting density functions through the non-zero USD times

Four right-skewed distributions are tested: Gamma, inverse Gaussian, log normal and exponential. Applying the GoF-test to all distributions, it appears that only the inverse Gaussian is significant on not coming from that distribution. The log normal, gamma and exponential distribution have p-values of 0.533, 0.1431 and 0.217, respectively. Therefore, the exponential density function is chosen to model $g(y)$, since it has the highest p-value. For modeling the point mass distribution p_0, the binomial distribution is used. The zero adjusted exponential model (ZAExp) takes now the following form:

$$f(y; \lambda, p_0) = \begin{cases} p_0, & \text{if } \ y = 0 \\ (1 - p_0)\lambda e^{-\lambda y}, & \text{if } \ y > 0 \end{cases}$$

For $y \in [0, \infty), p_0 \in (0, 1)$ and $\lambda > 0$

With

$$E_f[Y] = (1-p_0)E_g[Y] = (1-p_0)\frac{1}{\lambda} \text{ , for } y > 0$$
$$Var_f(Y) = (1-p_0)\frac{1}{\lambda^2} \text{ , for } y > 0$$

The model is implemented using the Generalized Additive Models for Location, Scale and Shape (GAMLSS) structure which is introduced by Stasinopoulos and Rigby (Stasinopoulus & Riby, 2007). The advantage of GAMLSS with respect to GAM or GLM is that they do not assume to be from the exponential distribution family. GAMLSS is extended to have a general distribution, including skewed distributions by allowing to model also other parameters than the mean (Stasinopoulus & Riby, 2007). Hence, in this case the mean and the p_0 can be modeled differently. Their relationship with the predictors are the following:

$$\log(\mu) = \eta_1 = X_1^T\beta_1$$
$$\text{logit}(p_0) = \eta_2 = X_2^T\beta_2$$

Where X_t and β_t for $t = 1, 2$ are the design matrices and corresponding coefficients respectively. These can be different for each parameter t.

5.4.1. Estimation

Both parameters are estimated through maximum likelihood estimation. The probability function of $f(y)$ can be given by $f(y; \lambda, p_0) = f(p_0)f(y|(1-p_0))$ Then the log-likelihood of $f(y)$ is the sum of the log of probability having zero USD time and log of the probability of USD time given that $USD > 0$ (Tong, Mues & Thomas, 2013).

$$\log f(y) = \log f(p_0) + \log f(y|(1-p_0))$$

The log-likelihood is then maximized separately into two components. Firstly, each response variable is transformed into $Y_i = I(Y_i = 0)$; the binomial model is fitted; and p_0 is found by MLE, as done in appendix A.4. In this way the probability of having $Y_i = 0$ can be determined. Secondly, for the values $Y_i > 0$ the λ can be estimated by MLE using the exponential distribution (Stasinopoulus, Enea & Rigby, 2017), as done in appendix G.1.

5.4.2. Model Selection

Like in the LM and in the GLM, the bias-variance trade off is assessed using best-subset selection comparing each AIC value. In AIC the log-likelihood is used with a penalty on the numbers of estimated parameters p. Since for each parameter estimation a different subset of covariates can be used, the number of estimated coefficient estimates p is much higher than in the LM or GLM case.

For estimating the parameter $\mu = \frac{1}{\lambda}$ all possible subsets of dummy variables are fitted such that for each categorical variable all $(k-1)$ dummies are included or all are excluded, like is done in the LM or GLM. For estimating the parameter p_0 all dummy variables are used for estimation. Note, for this parameter all possible subsets can be tested as well. However, this is left for further studies. The AIC values for each possible subset is shown in the following figure:

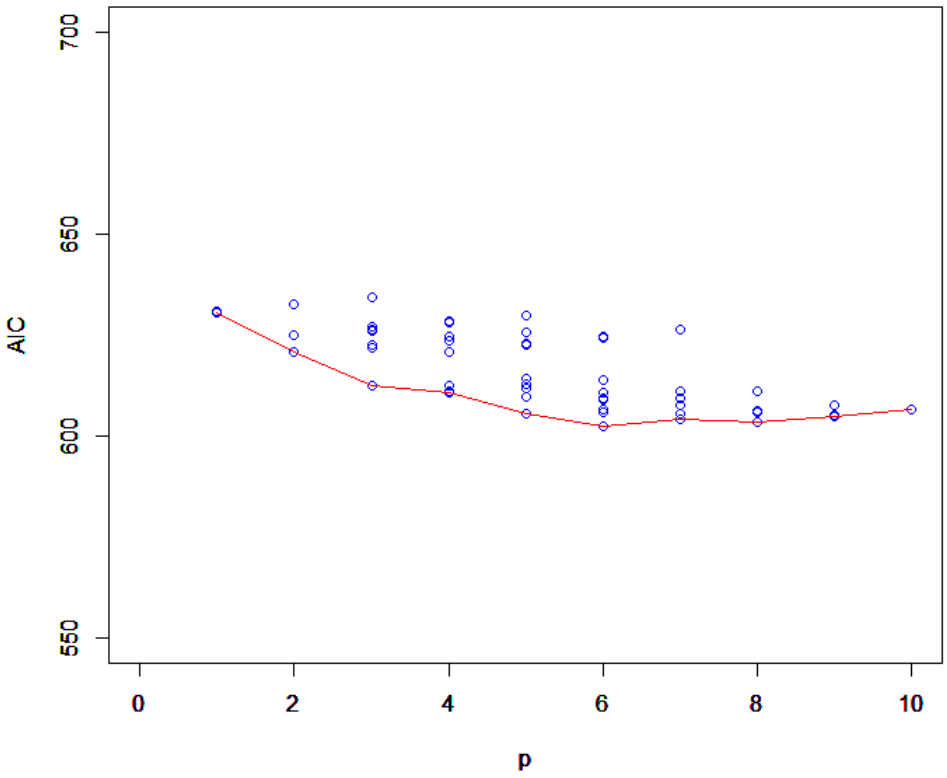


Figure 5.11: AIC against number of variables p using GAMLSS

The model with $p = 6$ has the lowest AIC value of 602.28 and contains the following coefficients:

	Coefficient	Std. Error	P-value
logit(p_0)			
Intercept	5.82	1.63	$4.33 \cdot 10^{-4}$
FieldE-Avg	-0.12	0.50	0.81
AG-Absent	-0.21	0.56	0.71
AG type-1	-0.21	1.24	0.87
AG type-2	-3.78	1.26	$2.88 \cdot 10^{-3}$
FSMFlex-Disabled	-2.04	0.70	$4.18 \cdot 10^{-3}$
Machine type-Old	-2.02	1.02	0.05
Machine type-1970	-2.74	0.89	$2.33 \cdot 10^{-3}$
Loc-C1	15.87	88.07	0.86
Loc-C2	1.38	0.61	0.03
Loc-C3	1.06	0.76	0.16

Table 5.6: Coefficients for estimating logit(p_0) of the selected GAMLSS model

	Term	Std. Error	P-value
log(μ)			
Intercept	2.24	1.04	0.03
AG type-1	-3.14	1.15	$6.72 \cdot 10^{-3}$
AG type-2	$-6.76 \cdot 10^{-1}$	1.02	0.51
FSMFlex-Disabled	$6.23 \cdot 10^{-1}$	$2.71 \cdot 10^{-1}$	0.02
Loc-C1	-1.58	$3.13 \cdot 10^{5}$	1.00
Loc-C2	-1.43	$2.68 \cdot 10^{-1}$	$2.16 \cdot 10^{-7}$
Loc-C3	$-9.85 \cdot 10^{-1}$	$5.35 \cdot 10^{-1}$	0.07

Table 5.7: Coefficients for estimating log(μ) of the selected GAMLSS model

Note, $\text{logit}(p_0) = \log\left(\frac{p_0}{1-p_0}\right) = X^T\beta \Leftrightarrow p_0 = \frac{e^{X^T\beta}}{1+e^{X^T\beta}}$ for $X^T\beta \in \mathbb{R}$. For estimating p_0, the **`intercept`**,

AG type-2, **FSMFlex-Disabled**, **machine type-Old**, **machine type-1970** and **loc-C2** are significant. **AG type-2**, **FSMFlex-Disabled** and the two **machine types** lead to a lower probability of having 0 USD time. **Location-C2** leads to a increase in probability of having 0 USD time.

Looking at the mean in case the USD time > 0: **AG type**, **FSMFlex** and **location** model the strongest effects compared to the variables which are not in the model. Compared to the GLM, **FSMFlex** is added to the model. Compared to the LM, **system type** is excluded. The variable that contributes in an increase in USD time is **FSMFlex-Disabled**. All the others lead to a decrease in mean USD time. Note, the coefficient estimates do not directly add to the expected USD time but by:

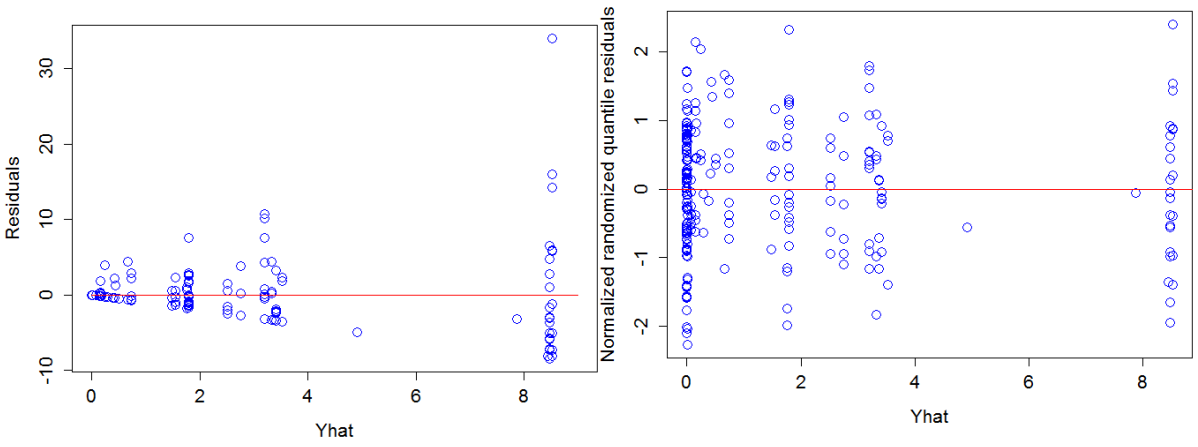
$$\log(\mu) = \mathbf{X}^T\beta \Leftrightarrow \mu = e^{\mathbf{X}^T\beta}$$

5.4.3. Model assessment

Firstly, the adequacy of the model needs to be checked. Just like in the LM and GLM we look at the residuals. Rigby & Stasinopoulos (2005) suggest an extended form of the residuals, in particular the *normalized randomized quantile residuals*:

$r_i = \Phi^{-1}(u_i)$
Where $u_i = F(y_i|\hat{p_0}, \hat{\mu})$ for y_i continuous,
and a random variable $u_i \in [F(y_i - 1|\hat{p_0}, \hat{\mu}), F(y_i|\hat{p_0}, \hat{\mu})]$ for y_i discrete response

If the residuals r_i have a standard normal distribution, the model is correct (Rigby & Stasinopoulos, 2005). In figure 5.12 the absolute and normalized randomized quantile residuals are plotted.



(a) Residuals against $\hat{Y}$ for the ZAExp

(b) Normalized randomized quantile residuals against $\hat{Y}$ for the ZAExp

Figure 5.12: Residual plots of the ZAExp

Looking at figure 5.12, the normalized randomized quantile residuals look randomly and normally distributed. The vertical lines are still visible but are inevitable because of the discrete predictors. Testing for normality using the KS test also shows that these residuals are likely to be normally distributed with a p-value of 0.44.

Summarizing, the model assumes that for USD> 0 it follows an exponential distribution, which is supported by the GoF-test. For these USD times a linear relationship is assumed between the exponent of the independent variables and the dependent variable and could be the right model to this data. This is also supported by the residuals since no specific structures can be seen from the normalized

randomized quantile plot. This model seems to be the right model to this data and hence the coefficient estimates and p-values can be interpreted.

Looking at the predictive properties, we have a value of $R^2 = 0.647$. This is an increase of 0.124 with respect to the LM. Moreover, no negative USD times are predicted. Calculating the RSS on the USD and estimated USD times, gives RSS = 3124.24. This is another improvement with respect to LM of 998.64 and GLM with 887.00.

5.4.4. Conclusion

The mixed distribution model seems to be a suitable model for the distribution of the USD time. The model has a $R^2 = 0.647$, hence the prediction capability may not be high enough and can be increased, but then other and/or more independent variables are needed.

Considering the interpretation, the model gives insight in three ways: the influence of the variables on the probability of having 0 USD time; what leveling options model the strongest effect given USD> 0 and how they influence the USD time given USD > 0.

For predicting the probability of having USD $= 0$ are **`AG type-2`**, **`FSMFlex-Disabled`**, both **`system types`** and **`location-C2`** significant. Interesting is that they all decrease the probability of having zero USD time, except for **`location-C2`**.

`AG type`, **`FSMFlex`** and **`location`** appear to model the strongest effects in predicting USD time given that USD> 0, whereas **`fieldE`**, **`AG`** and **`system type`** are left out. They all decrease the predicted USD time, except for **`FSMFlex-Disabled`**.
When we calculate the residual sum of squares of USD and $\hat{USD}$ we get RSS= 3124.24. This is an improvement of 998.64 compared to the LM and 887 to the GLM model.

5.5. Comparison & Conclusion

Regression can both be useful for prediction and interpretation, i.e. how the leveling options relate among each other and which variables model the strongest effect. Three types of regression models are applied in this chapter: the linear regression model, the generalized linear model and finally a mixed distribution model.

The often used linear regression model is not the correct model for this data. We do not meet the assumptions and as a consequence, the coefficients and p-values are not reliable. As an improvement, the generalized linear model is applied where we assume that the USD times come from an exponential family. However, this model does not meet the assumptions either and hence we can not interpret the model. Finally, the mixed distribution model is applied. The data do meet the assumptions and no undesired structures can be revealed from the residual plots. Therefore, this model is considered to be the right model for the data. Hence, from this model the coefficients and p-values can be interpreted. It follows that **`AG type`**, **`FSMFlex`** and **`location`** are the variables with the strongest effects on the USD time given USD> 0. **`C2`** and **`AG type-1`** both leads to a decrease of USD time of 1.43 and 3.14, respectively. **`FSMFlex-disabled`** increases the USD time of 0.623. The chance of having zero USD time is decreased by **`AG type-2`**, **`FSMFlex-disabled`**, **`machine type-old`**, **`machine type-1970`** where **`AG type-2`** has the highest effect of -3.78 and **`machine type-old`** the lowest effect of -2.02. The location **`C2`** increases the chance of having 0 USD time with an effect of 1.38.

However, the predictive capability of this model is not high enough. We see that 64.7% of the total variance is explained by the model. Moreover, we try to predict a continuous response with discrete predictors which results only in a limited number of predicted values. These vertical lines can be seen in the residuals plot of the mixed model.

In conclusion, now the right model is found, we need to improve the predictive power. From this model we can see which variables are the most important ones and how they relate among each other. However, to do predictions we need more and/or other variables where also continuous variables can be

considered. In this way we expect that the predictive power can be increased.

6

Root cause analysis

The variables **location** and **AG type** appear to be relevant parameters for predicting USD time of error B. However, a more surprisingly variable which appears in the zero adjusted model, is **FSM Flex package**. This relationship was less expected and is therefore interesting for further research.

The linear models show dependency but do not imply causality. To investigate if there is a causation and what this causation looks like, more expertise should be known in the engineering point of view of NXT machines. In the upcoming chapter a broad description is given about error B and FSM flexibility package. Afterwards a hypothesis is formed about the physical relationship between the two. Another hypothesis is added which is formed by expert knowledge. In the chapter thereafter insight is given in the usage of FSM thresholds using data of available machine diagnostics logging (MDLs). Finally, recommendations are done for further research.

6.1. Description error B

Error B is triggered when there are not enough valid points (250) for a global wafer wedge calculation Global wafer wedge (GWW) is needed for coarse wafer alignment (COWA) and fine wafer alignment (FIWA). These are needed to align the wafer and determine the horizontal position on the expose side. GWW can be calculated using the center spot data from global level contour (GLC) or by a sub selection of the wafer z-map (WZM) data, depending on the type of calculation of WZM. All the investigated machines have the scan in scan out (SOSI) based WZMs. Meaning that the GWW is calculated by the WZM data. However, when SOSI fails a fallback is done to GLC. As a consequence, the GWW is calculated by GLC data.

The investigation to error B shows that often SOSI fails before B occurs. Consequently, a fallback to GLC is done and GWW is calculated using GLC data by a plane fit (without AGILE correction)

Hence, GWW fails because GLC fails and as a consequence GLC has too less valid points. In the next chapter the functionality of GLC is explained and reasons why GLC could fail.

6.1.1. Functionality of GLC

GLC is the method of measuring the height Z and rotation R_y across the contour of the wafer using the level sensor (LS). This is done to determine the scan-in set points for the wafer map strokes measurements.

Before the GLC starts, the level sensor needs to know in which height it should start measuring. This done by using a *capture*. The capture determines the Z set point for GLC and set R_y to 0 by using the capture spot of the level sensor. Then the GLC starts and goes counter clockwise. Not the outermost contour is measured, but the focus edge clearance (FEC). This is the contour of the wafer slightly inwards. The measurement is done by the level sensor which consists of 35 rectangles, where the

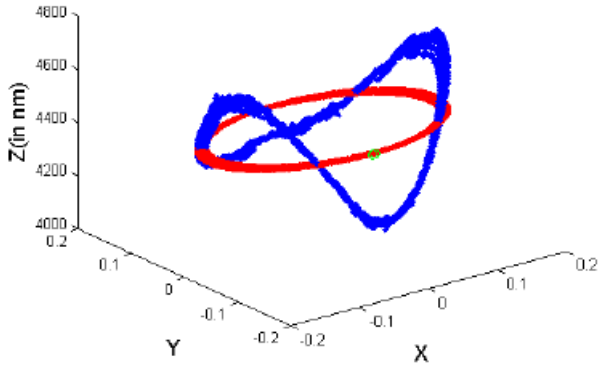


Figure 6.1: GLC data (blue) to calculate the plane fit (red)

central location of each rectangle is called *level sensor spot* (LS spots). The center of the level sensor spots follows the FEC. The data of the LS spots which fall within the FEC are collected.

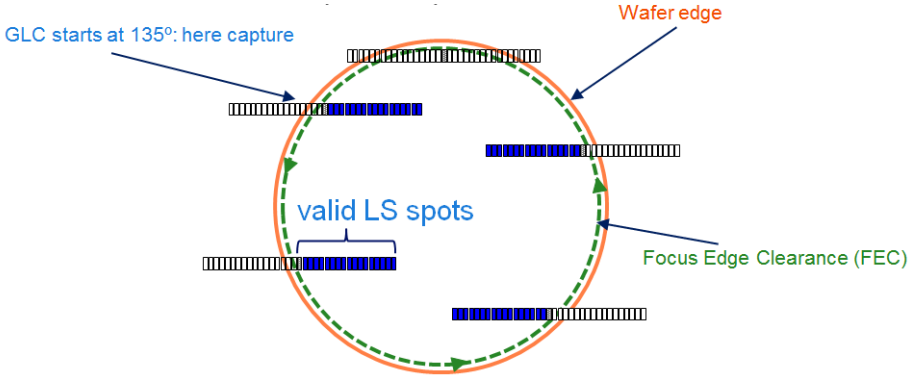


Figure 6.2: GLC measurement

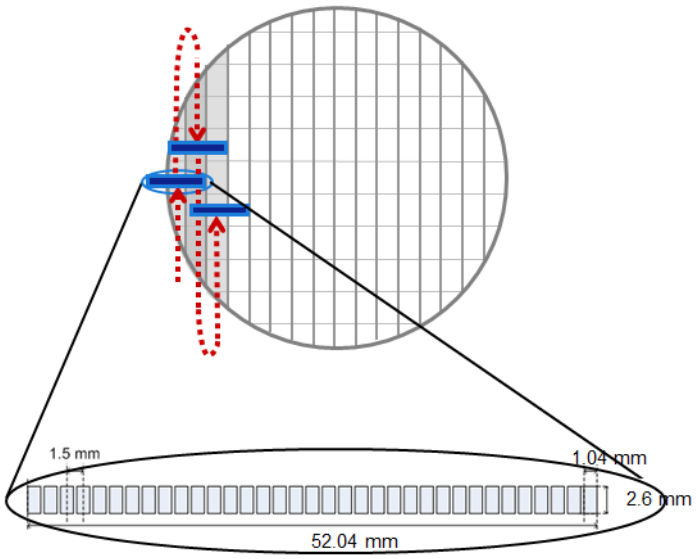


Figure 6.3: The WZM strokes

The main purpose of these Z and R_y values around the FEC is to determine the scan in set points for the WZM strokes. Side purposes are global wafer wedge (GWW) consisting of coarse wafer alignment (COWA) focus set points, fine wafer alignment (FIWA) focus set points and exposure fallbacks. Another side functionality is estimating the wafer height extremes.
In this chapter the focus is on the functionality of GWW calculation.

6.1.2. When GLC is triggered

GLC can be triggered by the PDxC sequence which is a sequence when PDGC or PDOC (AG) are executed in order to calibrate the LS on specific wafers. GLC can also be triggered by SOSI fallbacks. Moreover, GLC is performed during the approval sequence. As a previous investigation shows, error B mostly occurs after a SOSI-failing and occassionaly on the subsequent approval sequence. Therefore, the focus is on these last two instances.

- By a SOSI (systematic or event driven) fallback. That is, first SOSI is tried but when this fails GLC need to be done. This happens when there are large set point errors caused by unflat or contaminated wafers. This fallback leads to to a so called robust sequence or robust fallback sequence. The triggering of GLC using SOSI can be done by two ways:

  1. Systematic fallback to GLC: when the GLC output can not be derived from the WZM or when there is a high chance that skipping GLC not works.
  2. Event driven fallback to GLC: WZM without GLC is not measured correctly because of a technical error.

- During the approval sequence in which the wafers are approved for immersion and exposures. Only Chuck Temperature Conditioning (CTC) wafers and rejected wafers undergo this sequence. The minimum and maximum height at the wafer edge is calculated to determine if there is a risk for mechanical contact with immersion hood. These heights are determined by GLC.

6.1.3. Failing of GLC

GWW fails when no GLC data are available, so when the x, y, z coordinates are not available. Typically this is caused by contamination. High spots will let GLC fail ($> 2\mu m$). An example is shown in figure 6.4. GLC starts at the capture and goes counter clockwise. Contamination was present at $x \approx 25$ and $y \approx -140$ .After this contamination GLC loses track (dark blue) causing the absence of data.

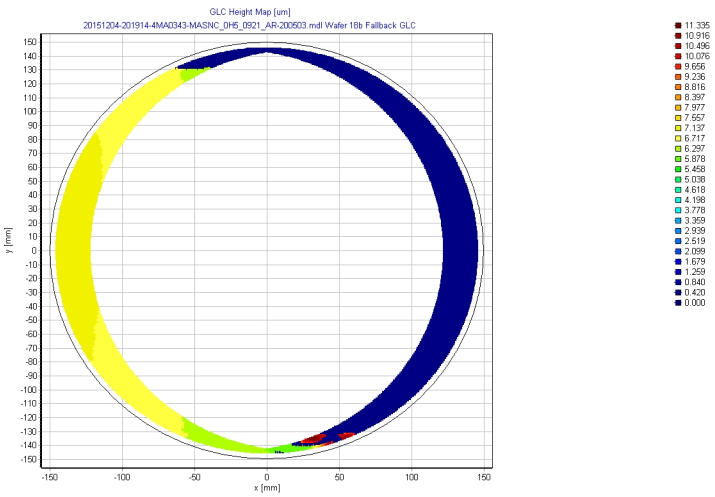


Figure 6.4: Example of performing GLC on wafer with contamination

6.2. Description FSM flexibility package

FSM flexibility package belongs to the sub function of contamination and control.
Contamination on a wafer results in focus and overlay errors. Therefore, contamination is undesired.

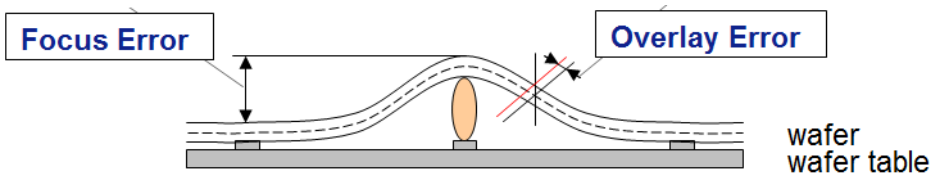


Figure 6.5: Contamination causes focus and overlay errors

To detect contamination, focus spot monitoring (FSM) is done. FSM is done on the measure side, using the WZM data. From these data high spots can be detected if present. These high spots indicate to contamination. When the detected height of the contamination spot, $z \geq$ threshold t, the contamination

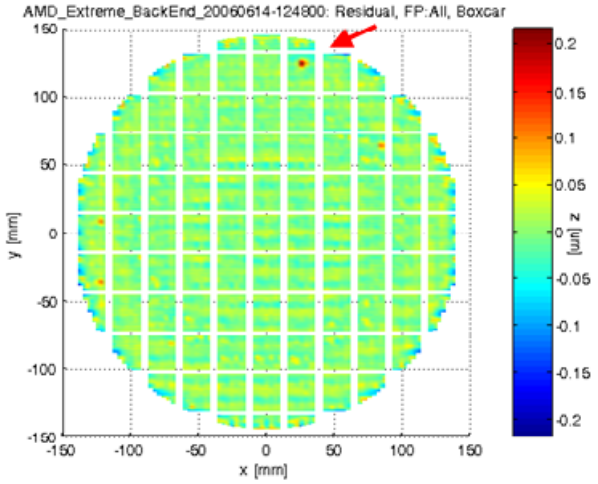


Figure 6.6: Example of a FSM residual map with a contamination spot

is detected. This is called a *focus spot*. The spot detection threshold is advised to be 40 nm. Thresholds can also be set up based on the standard deviation of the topography, this is called a *relative threshold*. When this detected focus spot is present on a consecutive series of wafers on the same chuck, it will be called *chuck contamination* or *chuck spot*.

When a chuck spot or focus spot is detected, *exception handling* is performed. That is, the wafer is rejected or accepted depending on the thresholds of the customers. When a number of focus spots detected > threshold, the wafer is labeled as *rejected*. When a number of chuck spots detected > threshold, the chuck is going to be cleaned and the lot is stopped or continued. FSM is improved by the commercial option *FSM flexibility package*. The main functionalities are:

- Allowing different detection thresholds for different radial zones (RTZ): four different thresholds are allowed for five different zones. For each radius a specific threshold is defined. For example, the following wafer has 3 radial zones:

  · Ignore the outer 5 mm
  · Less tight FSM spot detection threshold, e.g. 135-145 mm

  Using this, the false alerts can be ignored: detecting contamination while there is not. These radial zones are driven by edge roll off which causes false alerts on the edge.

Figure 6.7: FSM residual map with (false) spots on the edge and a wafer with radial zones

- Set wafer map exclusion areas (REA): exclude specific areas for contamination detection. For example, exclude the area that contains laser marks.

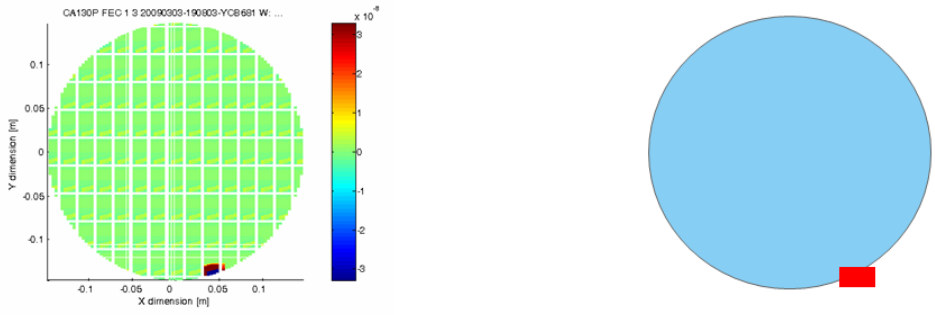


Figure 6.8: FSM residual map with a (false) spot and wafer with an excluded rectangular area

Both functionalities support flexibility in minimizing false alerts.

6.3. Possible relationships

The cause of error B and FSM flexibility package (flex pack) have a subject in common: contamination. Error B arises due to contamination on the edge. FSM and flex pack have the ability detecting contamination and minimizing false alerts. Therefore, a causality relationship is conceivable. Both a direct causality and a latent causality theory is formed. Both theories are stated in the next sections for which assumptions are made. A suggestion is done in how these assumptions could be investigated.

These sections are left out for confidential reasons.

7

Summary and conclusions

Investigation is done to the leveling-performance by analyzing errors. Errors can cause delay or unscheduled down time (USD time) which result in throughput loss. Therefore errors are not desired. Based on the total USD time in the considered time frame, we can answer sub question one: the most relevant leveling-errors are A, B, C and D. To answer sub question two: A and C show similar behavior and B and D as well. The first pair of errors does not occur very often compared to the other two (403 times and 109 times, respectively); however when they occur, the USD times can be high (mean is 1.08 h and 1.95 h, respectively). The second pair shows the opposite behavior: the mean USD time is lower (0.31 h and 0.06 h) but their frequency is higher (1354 and 1991 times). As a consequence, both types of errors should be handled differently in order to reduce the total USD time. The total USD time of errors A and C can be reduced by improving the mean time to repair. For B and D it can be reduced by decreasing the frequency.

Customer related variables are considered as possible causes of USD times. For example, location, system type and options customers can choose from can be of influence. These variables are clustered such that variables are removed that do not add extra information in the discrepancy of the usage. The remaining variables are used to model the USD time, where the study focuses on error B. To answer sub question three, we found significant differences in USD times between pairs of options of some variables. **`Leveling field extensions algorithm using the field averaged values`**, having **`air gauge present`** and having **`FSMFlex enabled`** have a lower USD time than their opposite level. Moreover, **`NXT1970`** compared to **`NXT1950`**, **`NXT1960`** and **`NXT1980Di`** and **`air guage type 2`** compared to **`air gauge type 1`** have a higher USD time. These last two significant differences can be explained by the introduction of the new level sensor using UV-light. Also within **`location`** significant differences are found. After testing the variables separately, it is of interest how they behave jointly. This is done by using the zero adjusted exponential model that is considered as the correct model to this data. To answer sub question four, **`location`** and **`air gauge type`** combined are the most important contributors in predicting USD times. A relevant supplement to these two is the **`FSMFlex package`**. Assuming USD time > 0, **`air gauge type 1`** leads to an increase of 0.04 USD time; **`FSMFlex disabled`** leads to an increase of 1.06 USD time; and a specific **`location`** leads to an increase of 0.24 USD time. **`System type`** is a less important contributor. Moreover, **`leveling field extensions algorithm`** and having **`air gauge`**, model the even smaller details and may be the least important supplement to **`location`** and **`air gauge type`** to predict USD time.

An interesting result came out of the regression analysis. Theories are formed and recommendations are done to investigate these theories on which ASML is working further on such that question five is answered.

8 Discussion

The main research question for this study is how to predict and decrease the USD times for the NXT machines such that the leveling-performance is increased at the customer. First suggestions are done in order to improve the study where different facets of the drawn conclusions are discussed. Afterwards different topics for future research are suggested.

8.1. Model improvements

Most important errors

All the logged USD times of the errors are considered, however a distinguishing can be made between the USD times which occurred during production and USD times which occurred not during production, e.g. during testing, rebooting, installing etc. This will lead to a more direct measure for performance.
Further, research can be done to the joint model of the USD times of the other errors; A, D and C and eventually all errors combined. The most important variables for each error can be compared together with their estimated coefficients. In this way it is discovered if some specific variables may be more important for one error compared to the other error. This comparison is also important for determining the total effect of one variable. For example, FSM flex disabled may increase the predicted USD time for error B, but may decrease the USD time of an other error.

Predictive model

One variable of each cluster is now used as input variable for testing on the USD time. However, the formed clusters and corresponding representatives can also be chosen differently. For example, by defining more clusters or by representing the cluster by another variable. However, the most important variables which model the USD time are variables which are not clustered. Therefore, it is expected that changing those clusters will not have a relevant impact on the prediction of USD time.
What may lead to an improvement for prediction is incorporating other variables then the ones considered now, in particular adding interval scaled variables, not necessarily customer-related. The zero adjusted model seems to be promising as a correct fit to the data and is an initial step in the direction of determining the crucial combination of variables, however the prediction capability is not high enough yet. For example, the 'age' of the latest upgrade to a new system type can be incorporated into the model. The advantage of continuous data is that it can take multiple forms compared to dummy variables.
The zero adjusted model has so far been applied on one data set. Variables can be found which have a decrease in USD time. To be more certain about the conclusion of the joint model, the mixed model should be applied on a new data set. Where possibly another continuous distribution function is chosen. Applying a mixed model on other data sets gives a more robust conclusion about which variables may be the most important ones in decreasing the USD time.
Further, another model could be thought of to try to predict the USD time, such as classification meth-

ods on the machines. That is, try to identify different styles of usage of machines. Investigate how the USD times look like for each style and what the differences are between the styles and USD times. A new observation could be classified to one of the styles such that the USD time can be predicted. However, more machines are needed to apply this model.

8.2. Future research

Besides the suggestions for improving the current model, recommendations related to the leveling department are done for a follow-up research.

- Investigate the root cause analysis of FSM flexibility package more into detail by answering the open questions in section 6.5.
- A more general proposal would be improving the storage of configurations, i.e. CM-options.
- A software related follow-up project would be making an automated tool where the input are data of (an) error(s) and explanatory variables. The output shows the ordering and behavior of those errors, such as frequency, mean, most frequent location etc. This can be expanded by fitting a zero adjusted model (possibly gamma, exponential or inverse Gaussian, dependent on the distribution of the USD time of the input error). Such an automated tool can also be useful for other departments within ASML.

Bibliography

[1] Borg, I., Groenen, P.J.F. (2005). *Modern Multidimensional scaling: Theory and Applications* (2nd ed.). New York, NY: Springer.

[2] Borg, I., Groenen, P.J.F., Mair, P. (2013). *Applied Multidimensional Scaling*. Springer Heidelberg New York Dordrecht London.

[3] Dekking, F.M., Kraaikamp, C., Lopuhaä, H.P., Meester, L.E. (2005). *A Modern Introduction to Probability and statistics, understanding Why and How*. Spring-Verlag London Limited.

[4] Dobson, A.J., & Barnett, A.G. (2008). *An Introduction to Generalized Linear Model* (3rd ed.). Boca Raton, Florida: Chapman & Hall.

[5] Everitt, B.S. (1977). *The Analysis of Contingency Tables*. Springer.

[6] Everitt, B.S. (1992). *The Analysis of Contingency Tables* (2nd ed.). Boca Raton, FL: Chapman & Hall/CRC.

[7] Everitt, B.S., Landau, S., Leese, M., Stahl, D. (2011). *Cluster Analysis* (5th ed.). West Sussex, United Kingdom: John Wiley & Sons Ltd.

[8] Gibbons, J.D., Chakraborti, S. *Nonparametric Statistical Inference*, Marcel Dekker, Inc. New York Basel, 2003, fourth edition, page 294

[9] Gower, J.C., Legendre P. (1986). Metric and Euclidean Properties of Dissimilarity Coefficients. *Journal of Classification*, 3:5-48, 34. Retrieved from `http://fitelson.org/coherence/gower_legendre.pdf`.

[10] Hastie, T., Tibshirani, R., Friedman, J. (2008). *The elements of statistical learning: Data Mining, Inference and Prediction* (2nd ed.). Springer.

[11] Hochberg, Y., Tamhane, A.C. (1987). *Multiple comparison procedures*. John Wiley & Sons, Inc, 1987.

[12] Hollander M., Wolfe, D.A., Chicken, E. (2014). *Nonparametric Statistical Methods* (3rd ed.). Hoboken, New Jersey: Wiley & Sons, Inc.

[13] De Jong, P. & Heller, G. Z. (2008). *Generalized Linear Models for Insurance Data*. New York: Cambridge University Press.

[14] Kruskal, J.B. (1964). *Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis*, Psychometrika, 29(1), 3-27.

[15] Lehmann, E.L, D'Abrera H.J.M. *Nonparametrics Statistical Methods Based on Ranks*. Holdan-Day, Inc, San-Franciso, 1975. Page 209.

[16] McCullagh, P., & Nelder, J.A. (1989). *Generalized Linear Models* (2nd ed.). London, New York: Chapman and Hall.

[17] Mori, Y., Kuroda, M., Makino, N. (2016) *Nonlinear principal component analysis and its applications*. Springer.

[18] Pratt, J.W., Gibbons, J.D. (1981). *Concepts of Nonparametric Theory*, New York, NY: Springer-Verlag.

[19] Rigby, R.A., & Stasinopoulos D.M. (2005). Generalized additive models for location, scale and shape. *Series C, Applied Statistics*, 54, 507-554. doi: 10.1111/j.1467-9876.2005.00510.x

[20] Singh, K., Malik, D., Sharma, N. (2011). Evolving limitations in K-means algorithm in data mining and their removal. *International Journal of Computational Engineering & Management*, 12, 107.

[21] Stasinopoulus, M., Enea, M., & Rigby, R.A. (2017). *Zero adjusted distributions on the positive real line*. Retrieved from `https://cran.r-project.org/web/packages/gamlss.inf/vignettes/ZeroAdjustedDistributions.pdf`

[22] Stasinopoulos, D.M., & Rigby, R.A. (2007). Generalized Additive Models for Location Scale and Shape (GAMLSS) in R. *Journal of Statistical Software*, 23(7), 1-42. Retrieved from `https://www.jstatsoft.org/article/view/v023i07/v23i07.pdf`.

[23] Tong, E.N.C., Mues, C. & Thomas, L. (2013). A zero-adjusted gamma model for mortgage loan loss given default. *International Journal of Forecasting*, 29(4), 548-562. Retrieved from `http://www.sciencedirect.com/science/article/pii/S0169207013000447`

A

Data set of leveling-errors

A.1. Confidential

This appendix is confidential

A.2. Confidential

This appendix is confidential

A.3. Confidential

This appendix is confidential

A.4. Maximum likelihood estimation of binomial distribution

$$
\begin{aligned}
ll(p; n, x_i, ..., x_n) &= \log\left(\prod_{i=1}^{n} f(x_i; p)\right) \text{ assuming independence} \\
&= \sum_{i=1}^{n} \log\left(\binom{n}{x_i} p^{x_i}(1-p)^{n-x_i}\right) \\
&= \sum_{i=1}^{n} \log\left(\binom{n}{x_i}\right) + x_i \log(p) + (n - x_i)\log((1-p)) \\
\frac{\delta ll}{\delta p} &= \sum_{i=1}^{n} \frac{x_i}{p} + \frac{n - x_i}{1-p} = 0 \\
&\Leftrightarrow p = \sum_{i=1}^{n} \frac{x_i}{n}
\end{aligned}
$$

Assuming that the second derivative < 0.

A.5. Probability density estimates of USD times on daily data

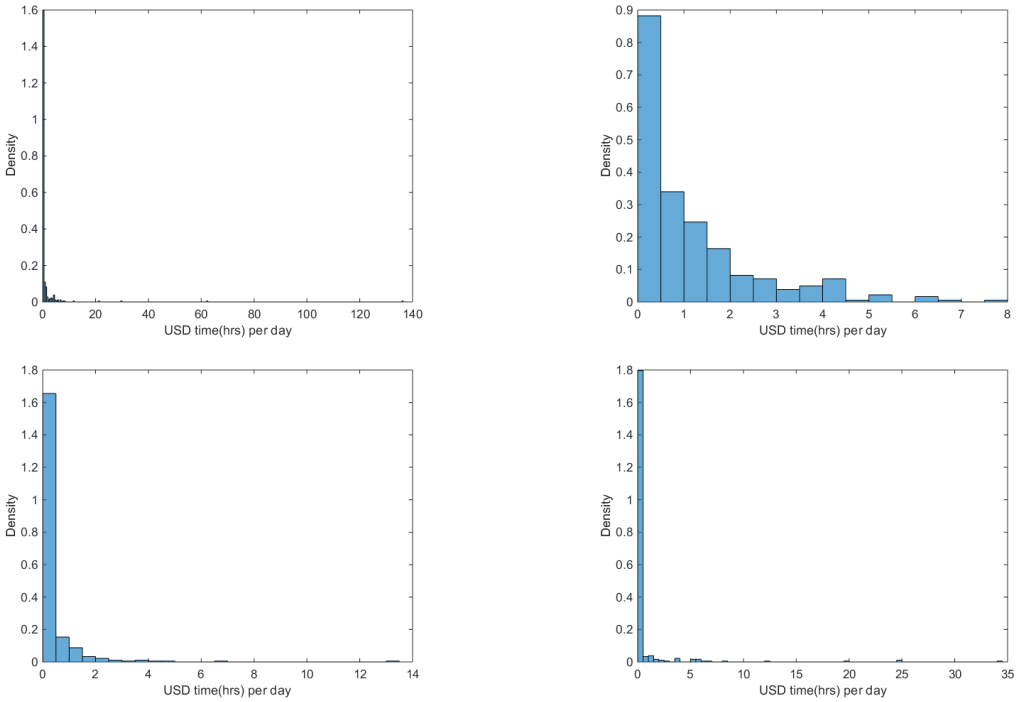


Figure A.1: Estimated probability density function of the USD times of the four most important errors. From left to right and top left to bottom right: A, B, D, C.

Note, the limits of the x-axes between the four errors differ.

B

Data set of leveling-configurations

B.1. Confidential

This appendix is confidential

B.2. Confidential

This appendix is confidential

B.3. Merged levels

System type	NXT2:1950BI NXT2:1950i	NXT1950
	AT:NXT1960Bi NXT2:1960BI NXT2:1960Bi NXT2:1960i NXT:1960BI NXT:1960Bi	NXT1960
	NXT3:1965CI NXT3:1965Ci NXT:1965CI NXT:1965Ci	NXT1965
	NXT3:1970CI NXT3:1970Ci NXT3:1970i NXT:1970Ci	NXT1970
	NXT3:1980CI	NXT1980Ci
	NXT3:1980DI	NXT1980Di
	NXT2	NXT2

Table B.1: The original levels and merged levels for the feature `system type`

Note, a confidential table is left out.

Leveling Setpoint Smoothing	Use LS spot fading on edge dies	Use LS spot fading on edge dies
	Do not use LS spot fading on edge dies Do not use LS spot fading on dies(Overrules use LS spot fading on edge dies)	Do not use LS spot fading on edge dies

Table B.2: The originally and merged levels for feature `Leveling Setpoint Smoothing`

Type of Air Gauge	TYPE_1: Initial version TYPE_1: Initial air gauge (-25.5mm)	TYPE_1: Initial air gauge (-25.5mm)
	TYPE_2: Shifted air gauge (-45.5mm)	TYPE_2: Shifted air gauge (-45.5mm)
	No Air Gauge device present	No Air Gauge device present

Table B.3: The originally and merged levels for feature `type of Air Gauge`

B.4. Features with their corresponding levels

Feature	**Levels**	**Abbreviation**
System type	NXT1950, NXT1960, NXT1965, NXT1970, NXT1980Ci, NXT1980Di, NXT2	Does not apply
Location	See table **??**	Does not apply
Field with optimised leveling	Disabled Enabled	FieldW-Dis FieldW-En
Leveling field extensions algorithm	Confidential Confidential	FieldE-Local FieldE-Avg
Leveling setpoint smoothing	Confidential Confidential	Smooth-NotUse Smooth-Use
Leveling on single LS spots	Do not use single spot leveling Use single spot leveling	Single-NotUse Single-Use
Leveling with LS spot weight update algorithm	Selection of LS Spot Weight Update algorithm is allowed Selection of LS Spot Weight Update algorithm is not allowed	Spot-Use Spot-NotUse
Air Gauge	Absent Present	AG-Absent AG-Present
Air Gauge improved leveling	Agile1 Agile2 S-Agile Absent	Agile-1 Agile-2 S-Agile Agile-No
Type of Air Gauge	Type_1: Initial air gauge (-25.5mm) Type_2: Shifted air gauge (-45.5mm) No air gauge device present	AGT-1 AGT-2 AGT-No
FSM Flexibility package	Disabled Enabled	FSMFlex-Dis FSMFlex-En
Improved FSM algorithm Part of FIP-1 commercial package	Disabled Enabled	iFSM-Dis iFSM-En

Table B.4: Features with the corresponding levels and their abbreviations

B.5. Correlation matrix of the features

	FieldE-Avg	Smooth-NotUse	Single-NotUse	Spot-NotUse	FieldE-Local	Smooth-Use	Single-Use	Spot-Use
FieldE-Avg	1.00	0.90	0.99	0.91	-1.00	-0.90	-0.99	-0.91
Smooth-NotUse		1.00	0.91	0.89	-0.90	-1.00	-0.91	-0.89
Single-NotUse			1.00	0.92	-0.99	-0.91	-1.00	-0.92
Spot-NotUse				1.00	-0.91	-0.89	-0.92	-1.00
FieldE-Local					1.00	0.90	0.99	0.91
Smooth-Use						1.00	0.91	0.89
Single-Use							1.00	0.92
Spot-Use								1.00

Table B.5: Correlations between first two clusters

	AGPresent	AGILE2	AGAbsent	AGILE-No
AGPresent	1.00	0.99	-1.00	-0.90
AGILE2		1.00	-0.99	-0.90
AGAbsent			1.00	0.90
AGILE-No				1.00

Table B.6: Correlations between third and fourth cluster

	FieldW-En	iFSM-En	FieldW-Dis	iFSM-Dis
FieldW-En	1.00	0.86	-1.00	-0.86
iFSM-En		1.00	-0.86	-1.00
FieldW-Dis			1.00	0.86
iFSM-Dis				1.00

Table B.7: Correlations between fifth and sixth cluster

	AGPresent	AGILE2	AGAbsent	AGILE-No	FieldW-En	iFSM-En	FieldW-Dis	iFSM-Dis
FieldE-Avg	-0.08	-0.09	0.08	0.00	-0.13	-0.16	0.13	0.16
Smooth-NotUse	-0.01	0.00	0.01	0.03	-0.10	-0.13	0.10	0.13
Single-NotUse	-0.09	-0.09	0.09	0.00	-0.14	-0.16	0.14	0.16
Spot-NotUse	-0.02	-0.01	0.02	0.04	-0.15	-0.17	0.15	0.17
FieldE-Local	0.08	0.09	-0.08	0.00	0.13	0.16	-0.13	-0.16
Smooth-Use	0.01	0.00	-0.01	-0.03	0.10	0.13	-0.10	-0.13
Single-Use	0.09	0.09	-0.09	0.00	0.14	0.16	-0.14	-0.16
Spot-Use	0.02	0.01	-0.02	-0.04	0.15	0.17	-0.15	-0.17

Table B.8: Correlation matrix of first two clusters with variables outside these clusters

	FSMFlex-Dis	FSMFlex-En	AGT1	AGT2	AGTNo	AGILE1	S-AGILE
FieldE-Avg	0.32	-0.32	0.03	-0.10	0.17	0.05	0.19
Smooth-NotUse	0.24	-0.24	-0.10	0.02	0.18	-0.07	-0.05
Single-NotUse	0.32	-0.32	0.04	-0.11	0.17	0.05	0.19
Spot-NotUse	0.26	-0.26	-0.05	-0.02	0.18	-0.07	-0.06
FieldE-Local	-0.32	0.32	-0.03	0.10	-0.17	-0.05	-0.19
Smooth-Use	-0.24	0.24	0.10	-0.02	-0.18	0.07	0.05
Single-Use	-0.32	0.32	-0.04	0.11	-0.17	-0.05	-0.19
Spot-Use	-0.26	0.26	0.05	0.02	-0.18	0.07	0.06

Table B.9: Continue of correlation matrix of first two clusters with variables outside these clusters

	FieldW-En	iFSM-En	FieldW-Dis	iFSM-Dis	Flex-Dis	Flex-En	AGT1	AGT2	AGTNo	AGILE1	S-AGILE
AGPresent	-0.05	0.01	0.05	-0.01	-0.47	0.47	0.17	-0.08	-0.21	0.06	-0.24
AGILE2	-0.05	0.01	0.05	-0.01	-0.48	0.48	0.16	-0.07	-0.20	-0.07	-0.24
AGAbsent	0.05	-0.01	-0.05	0.01	0.47	-0.47	-0.17	0.08	0.21	-0.06	0.24
AGILE-No	0.03	-0.03	-0.03	0.03	0.38	-0.38	-0.33	0.22	0.23	-0.06	-0.20

Table B.10: Correlations of third and fourth cluster with variables outside these clusters

	FSMFlex-Dis	FSMFlex-En	AGT1	AGT2	AGTNo	AGILE1	S-AGILE
FieldW-En	0.07	-0.07	-0.19	0.22	-0.11	0.01	0.04
iFSM-En	0.11	-0.11	-0.13	0.16	-0.09	0.01	0.04
FieldW-Dis	-0.07	0.07	0.19	-0.22	0.11	-0.01	-0.04
iFSM-Dis	-0.11	0.11	0.13	-0.16	0.09	-0.01	-0.04

Table B.11: Correlations of fifth and sixth clusters with variables outside these clusters

C

Rank based tests

C.1. Simulated null-distributions of the USD times for Wilcoxon rank sum test

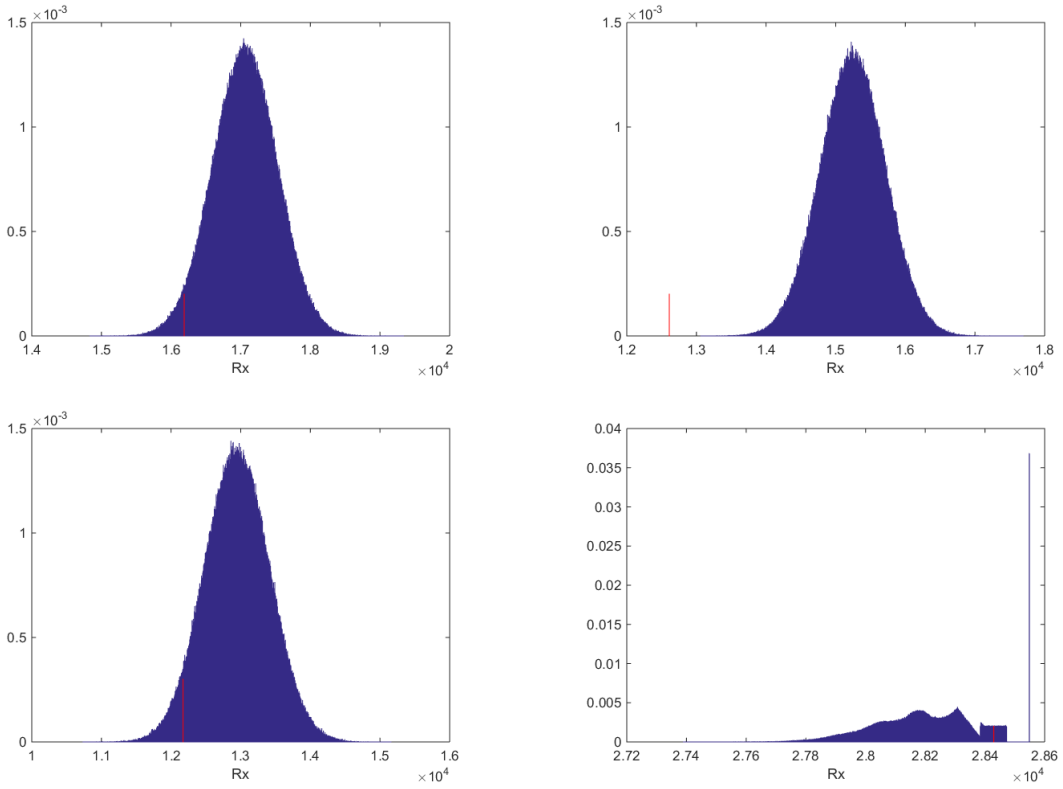


Figure C.1: The null distribution of R_x of each of the features: from top left to bottom right: `fieldE`, `AG`, `FSMFlex`, `iFSM`

Note, the observed R_x for `AG` is so small it is not even present in the null distribution. A possible explanation is that for simulating the null distribution not all possible combinations are simulated but only a limited amount of permutations. Further, the null distribution of `iFSM` is lumpy. This can be explained by the highly unequal sample sizes: m consists of 233 observations and n of 8 observations. So when applying permutation, not much different combinations are possible anymore.

D

AIC value of least squares method

In the linear regression model the dependent variable y is assumed to be normal with mean $x_i\beta$ and variance σ^2. We get the following likelihood and loglikelihood, assuming the y_i's are independent:

$$\begin{aligned} l(\beta, \sigma^2, y, X) &= \prod_{i=1}^{n} f(y_i; \beta, \sigma) \\ &= \prod_{i=1}^{n} \left(2\pi\sigma^2\right)^{-1/2} \exp\left(-\frac{1}{2}\frac{(y_i - x_i\beta)^2}{\sigma^2}\right) \\ &= \left(2\pi\sigma^2\right)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - x_i\beta)^2\right) \\ ll(\beta, \sigma^2; y) &= -\frac{n}{2}\ln(2\pi) - \frac{n}{2}\ln(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - x_i\beta)^2 \end{aligned}$$

Then maximizing the likelihood for σ^2, we get

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i\beta)^2$$

Then the maximized likelihood is given by

$$\begin{aligned} \hat{ll}(\beta, \sigma^2; y) &= -\frac{n}{2}\ln\left(\frac{RSS}{n}\right) - \frac{1}{2\frac{RSS}{n}}RSS \\ &= \frac{-n}{2}\ln\left(\frac{RSS}{n}\right) + C \end{aligned}$$

E

Correlation matrix of the dummy variables

	FieldE-Avg	AG-Abs	AGT-1	AGT-2	Flex-Dis	MType-Old	MType-1970	Loc-C1	Loc-C2	Loc-C3
FieldE-Avg	1.000	0.080	0.035	-0.100	0.315	0.321	-0.206	0.269	0.182	-0.325
AG-Abs		1.000	-0.170	0.082	0.469	-0.161	0.199	-0.339	0.498	-0.507
AGT-1			1.000	-0.917	-0.147	0.639	-0.554	-0.265	0.059	0.463
AGT-2				1.000	0.107	-0.697	0.604	0.288	-0.069	-0.407
Flex-Dis					1.000	-0.023	0.082	0.314	0.481	-0.594
MType-Old						1.000	-0.867	0.015	0.025	0.199
MType-1970							1.000	-0.006	0.003	-0.289
Loc-C1								1.000	-0.244	-0.215
Loc-C2									1.000	-0.319
Loc-C3										1.000

Table E.1: Correlation matrix of the dummy variables

	FieldE-Avg	AG-Abs	AGT-1	AGT-2	Flex-Dis	Mtype-Old	Mtype-1970	Loc-C1	Loc-C2	Loc-C3
FieldE-Avg	1.000	0.219	0.594	0.122	0.000	0.000	0.001	0.000	0.005	0.000
AG-Abs		1.000	0.008	0.206	0.000	0.012	0.002	0.000	0.000	0.000
AGT-1			1.000	0.000	0.023	0.000	0.000	0.000	0.361	0.000
AGT-2				1.000	0.098	0.000	0.000	0.000	0.283	0.000
Flex-Dis					1.000	0.720	0.205	0.000	0.000	0.000
Mtype-Old						1.000	0.000	0.812	0.698	0.002
Mtype-1970							1.000	0.926	0.958	0.000
Loc-C1								1.000	0.000	0.001
Loc-C2									1.000	0.000
Loc-C3										1.000

Table E.2: P-values corresponding to the correlation matrix

F

Generalized Linear Model

F.1. Gamma and Inverse Gaussian distribution fit through USD with a small constant

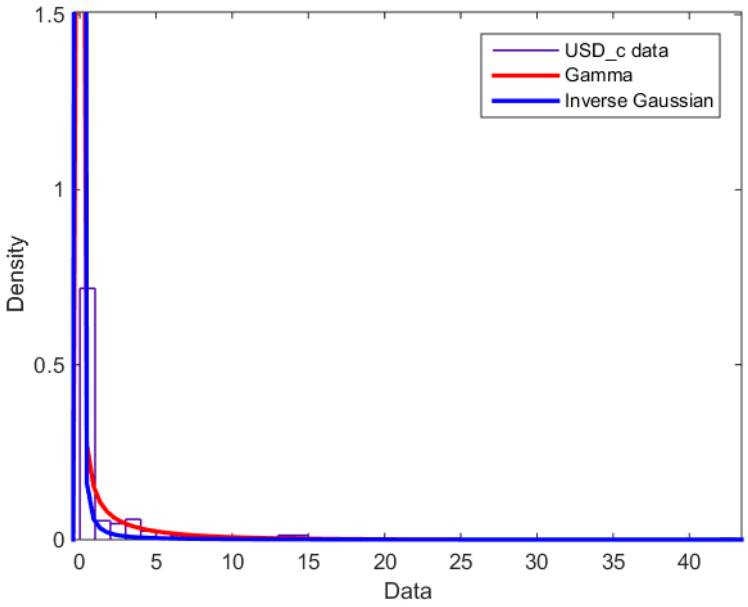


Figure F.1: A fitted Gamma and Inverse Gaussian are plotted through the distribution of USD+ϵ with ϵ small

A small constant to USD is added to ensure values > 0. Further, the binwidth of the histogram are chosen by the Freedman-Diaconis decision rule.

F.2. Obtaining maximum likelihood estimates for GLM

$$
\begin{aligned}
\frac{\delta ll}{\delta \beta_j} &= \sum_{i=1}^{n} \frac{\delta ll}{\delta \theta_i} \frac{\delta \theta_i}{\delta \beta_j} = 0 \\
&= \sum_{i=1}^{n} \frac{(y_i - b'(\theta_i))\, a(\phi_i) - (y_i\theta_i - b(\theta_i)) \cdot 0}{a(\phi_i)^2} \frac{\delta \theta_i}{\delta \beta_j} = 0 \\
&= \sum_{i=1}^{n} \frac{y_i - b'(\theta_i)}{a(\phi_i)} \frac{\delta \theta_i}{\delta \beta_j} = 0 \\
&= \sum_{i=1}^{n} \frac{y_i - b'(\theta_i)}{a(\phi_i)} \frac{\delta \theta_i}{\delta \eta_i} \frac{\delta \eta_i}{\delta \beta_j} = 0 \\
&= \sum_{i=1}^{n} \frac{y_i - b'(\theta_i)}{a(\phi_i)} \frac{\delta \theta_i}{\delta \eta_i} x_{ij} = 0 \\
&= \sum_{i=1}^{n} \frac{y_i - \mu_i}{a(\phi_i)} \frac{\delta \theta_i}{\delta \eta_i} x_{ij} = 0 \\
&= \sum_{i=1}^{n} \frac{y_i - \mu_i}{a(\phi_i)} \left(\frac{\delta \eta_i}{\delta \theta_i} \right)^{-1} x_{ij} = 0 \\
&= \sum_{i=1}^{n} \frac{y_i - \mu_i}{a(\phi_i)} \left(\frac{\delta \eta_i}{\delta \mu_i} \frac{\delta \mu_i}{\delta \theta_i} \right)^{-1} x_{ij} = 0 \\
&= \sum_{i=1}^{n} \frac{y_i - \mu_i}{a(\phi_i)} \left(g'(\mu_i) b''(\theta_i) \right)^{-1} x_{ij} = 0 \\
&= \sum_{i=1}^{n} (y_i - \mu_i) \left(g'(\mu_i) \mathrm{Var}(\mu_i) \right)^{-1} x_{ij} = 0
\end{aligned}
$$

Where it is used that $b'(\theta_i) = \mu_i = E(Y_i)$ and $b''(\theta_i) = Var(\mu_i)$ (McNullagh & Nelder, 1989).
The last equation can be written in matrix form by (De Jong & Heller, 2008, ch.5, p.68), :

$$\mathbf{X}^T D(y - \mu) = 0$$

Where D is a diagonal matrix with the entries $(g'(\mu_i)\mathrm{Var}(\mu_i))^{-1}$. Now splitting the matrix D into two matrices W, G where W has entries $\left(g'(\mu_i)^2 \mathrm{Var}(\mu_i)\right)^{-1}$ and G has entries $g'(\mu_i)$ then $D = WG$ and hence $\mathbf{X}^T WG(y - \mu) = 0$. Linking this equation to a Taylor approximation and using $\mathbf{X}\beta = \mu$ leads to the equation

$$\hat{\beta} = \left(\mathbf{X}^T W \mathbf{X} \right)^{-1} \mathbf{X}^T W y$$

where the diagonal entries of W are $1/\mathrm{Var}(\mu_i)$ and $\hat{\beta}$ is the weighted least squares estimator (De Jong & Heller, 2008, ch.5, p.68). The result is difficult to solve directly. Therefore the *iterative weighted least squares* method is used (Dobson & Barnett, 2008, ch. 4, p. 66). This can be written as:

$$\hat{\beta}^{(m+1)} = \left(\mathbf{X}^T W^{(m)} \mathbf{X} \right)^{-1} \mathbf{X}^T W^{(m)} y$$

In here μ replaced by $\mu^{(m)}$ which is estimated by $g(\mu^{(m)}) = \mathbf{X}\beta^{(m)}$ where an initial approximation of $\beta^{(0)}$ is needed. Then the variance can be calculated where $g(\theta_i)$ is needed in order to define $b''(\theta_i))$ so that $\beta^{(m+1)}$ can be determined.

F.3. Parameters of IG for the exponential family density function

Exponential family is given by:

$$f(y;\theta,\phi) = exp\left\{\frac{y\theta - b(\theta)}{a(\phi)} + c(y,\phi)\right\}$$

Now the Inverse Gaussian can be transformed to the exponential family density function by:

$$\begin{aligned}
f(y;\mu,\lambda) &= \sqrt{\frac{\lambda}{2\pi\lambda^3}}\exp\left(\frac{-\lambda(y-\mu)^2}{2\mu^2 y}\right) \quad \text{for } \mu,\lambda>0 \text{ and } 0<y<\infty \\
&= \exp\left(\frac{-\lambda(y-\mu)^2}{2\mu^2 y} + \ln\left(\frac{\lambda}{2\pi y^3}\right)^{1/2}\right) \\
&= \exp\left(\frac{-\lambda(y-\mu)^2}{2\mu^2 y} + \frac{1}{2}\ln\left(\frac{\lambda}{2\pi y^3}\right)\right) \\
&= \exp\left(\frac{-\lambda(y-\mu)^2}{2\mu^2 y} + \underbrace{\frac{1}{2}\left[\ln(\lambda) - \ln(2\pi y^3)\right]}_{S}\right) \\
&= \exp\left(\frac{-\lambda(y^2-2\mu y+\mu^2)}{2\mu^2 y} + S\right) \\
&= \exp\left(\frac{-\lambda y^2 + 2\mu y\lambda - \lambda\mu^2}{2\mu^2 y} + S\right) \\
&= \exp\left(\frac{-\lambda y^2}{2\mu^2 y} + \frac{2\mu y\lambda}{2\mu^2 y} - \frac{\lambda\mu^2}{2\mu^2 y} + S\right) \\
&= \exp\left(\lambda\left[\left(y\frac{-1}{2\mu^2}\right) + \frac{1}{\mu}\right] - \frac{\lambda}{2y} + S\right) \\
&= \exp\left(\frac{y\frac{-1}{2\mu^2} + \frac{1}{\mu}}{1/\lambda} - \frac{\lambda}{2y} + S\right) \text{ , where the square brackes are multiplied by } \frac{1/\lambda}{1/\lambda}
\end{aligned}$$

Then

$$\theta = \frac{-1}{2\mu^2}, \quad b(\theta) = \frac{-1}{\mu} = -\sqrt{-2\theta}, \quad \phi = \frac{1}{\lambda}$$

and

$$\begin{aligned}
c(y,\phi) &= \frac{-\lambda}{2y} + S \\
&= \frac{-\lambda}{2y} + \frac{1}{2}\ln(\lambda) - \frac{1}{2}log(2\pi) - \frac{3}{2}\ln(y) \\
&= \frac{-1}{2y\phi} - \frac{1}{2}\ln(2\pi/\phi) - \frac{3}{2}\ln(y)
\end{aligned}$$

G

Zero adjusted exponential model

G.1. Maximum likelihood estimation of $f(y|(1-p_0))$

Let $y_i > 0$ for $i = 1, ..., n$ and $y_i \sim Exp(\lambda)$. Then the likelihood and log-likelihood are given by:

$$\begin{aligned} l(\lambda; y_1, ..., y_n, p_0) &= \prod_{i=1}^{n} f(y_i; \lambda, p_0) \text{ , assuming } y_i \text{ are independent} \\ &= \prod_{i=1}^{n} (1-p_0)\lambda e^{-\lambda y_i} \\ &= (1-p_0)^n \lambda^n e^{-\lambda \sum y_i} \\ ll(\lambda; y_1, ..., y_n, p_0) &= n\log(1-p_0) + n\log(\lambda) - \lambda \sum y_i \end{aligned}$$

Differentiating to λ and set equal to zero gives:

$$\begin{aligned} \frac{\delta ll}{\delta \lambda} &= \frac{n}{\lambda} - \sum y_i = 0 \\ &\Leftrightarrow \hat{\lambda} = \frac{n}{\sum y_i} \end{aligned}$$

To ensure it is the maximum the second derivative should be < 0.

$$\frac{\delta ll^2}{\delta^2 \lambda} = -n\lambda^{-2}$$

which is < 0 since $n, \lambda > 0$.