AirBit Consultancy

KLM Business Case

Forecasting removals for a component pool

Authors:	*Email:*
Sebastiaan Berendsen (2537690)	s.p.berendsen@student.vu.nl
Sabine van Breugel (2549431)	s2.van.breugel@student.vu.nl
Joop van der Hulst (2539115)	j.t.vander.hulst@student.vu.nl
Kirsten Kniep (2553118)	k.p.e.kniep@student.vu.nl
Joshua Touati (2540273)	j.touati@student.vu.nl

Supervisor VU: Rob van der Mei
Contact person KLM: Joost Jorritsma

3 July, 2017

1 Management Summary

Predictive modeling is the process of creating, testing and validating a model to predict the probability of an outcome. KLM Engineering & Maintenance, in short KLM E&M, requires this type of modeling. KLM E&M provides components for the Boeing 787 to numerous aircraft carriers using a so-called 'component pool'. This component pool consists of many different components, which are installed on a plane or waiting in stock. When a broken unit is removed, KLM E&M provides a new unit from the pool and restores the broken unit. Then this unit will return to the pool.

It is important for KLM E&M to know what the expected number of component removals in the pool will be. On the one hand, they have to have enough spare components in the pool, because the delivery time of new components is long and a broken component should be replaced as soon as possible. A delay in delivering a new component means a delay of the aircraft, which can cost a lot of money. On the other hand, a component costs a lot of money, which means that a minimal idle cost is preferred. Therefore, it is important that the stock levels are not too high and not too low.

The practical relevance of this study is to create a forecasting model that predicts the amount of expected component removals. When the right model fits the problem, KLM E&M can use this model to reduce storage costs for all components. Furthermore, a comparison between the usage of components of aircraft carriers is made, because carriers who need more components have to pay more for the contract.

Concerning the model, this study concludes that the Weibull, Exponential and the Log-normal distributions are all plausible distributions to resemble the data. A model has been created and it has predicted the expected amount of removals for the period 01-04-2017 until 01-07-2017. However, it is possible to use the model for any defined period. Because of the reliability at which KLM has to be able to provide the maintenance, the model is enriched with a 95% confidence level. This means that in 95% of the cases a less or equal amount of removals occurs and in 5% of the cases more removals are occurring. This confidence level can be adjusted to any level between 50% and 100%. In the appendix of this study, a manual for the model is given. The model itself is a separate Python file.

The conclusion about which aircraft carriers perform well or badly is hard to make, due to missing values. Using the available data, the carriers C03, C04 and C07 seem to perform the best, while carriers C05, C11 and C13 perform the worst. However, this conclusion cannot be made with certainty, because of the missing values.

Several methods were analyzed in this study. The main obstacle that kept coming up is the fact that the data contains a too small time horizon. Due to the fact that the dataset is small, it is important to update the dataset each month and use the model on the updated dataset. By doing this, the prediction of the expected number of removals becomes increasingly more accurate.

Contents

2 Introduction

Predictive modeling is the process of creating, testing and validating a model to best predict the probability of an outcome. Numerous studies have shown that predictive modeling is a useful tool for determining whether future predictions can be made. Predictive modeling is synonymous with the field of machine learning. Steyerberg, Ewout W. (2010) showed that predictive models can be used directly to estimate an output given a defined set of input variables.[1] Using such a model can give benefits, because of the future knowledge it provides. For example, if it is known how much the future demand will be, one will be able to set perfect stock levels for a product. This will reduce the cost of storing additional products.

Based on historical data it is possible to use predictive modeling for determining how often a certain event will occur. A company that could use such models is KLM Engineering & Maintenance, in short, KLM E&M. This company provides components for the Boeing 787 to numerous airline customers and has what they call a 'component pool'. This component pool consists of about a thousand different components. These are installed on a plane or waiting in stock to be installed. When a broken unit is removed, KLM E&M provides a new unit from the 'pool' and restores the broken unit. Then this unit will return to the 'pool'. The number of removals depends on the failure rate of a component, which is defined as the amount of flying hours a certain component endures before it breaks down. It would be profitable for KLM E&M to be able to determine the failure rate of a component since in that case, they are able to determine a more accurate number of components needed. This would reduce the purchase and storage costs of components. The failure rates and the amount of removals can be predicted with a predictive model.

Predictive modeling can be done by using numerous models. Each model has its own properties and is not always applicable to each problem. Therefore, for each specific problem, literature research is needed to determine which predictive models can be used. Also, it is often the case that multiple models can be used for predicting a certain response variable. It is, in fact, a good idea to use multiple models on a problem since a comparison between the results will be possible. Based on these results, the model with the best predicting ability will be chosen. Unfortunately, a significant difference between the results is not always present. This often encountered phenomenon makes it hard to choose a model. Nevertheless, combining different models or iteratively using the same model can overcome this obstacle. Considering these points, it is important to spend enough time on finding the right model or models for a specific problem.

For determining what model or what models can be used by KLM E&M, it is necessary to make a difference between regression and classification. A model that uses a form of regression predicts a response variable that takes continuous values, whereas a model based on classification predicts an output variable that takes class labels or discrete values. KLM E&M needs to predict failure rates, which are continuous values. Therefore, a form of a regression model is wanted here.

The practical relevance of this study concerns two aspects. Firstly, a forecasting model must be found that predicts the amount of expected component removals. When the right model fits the problem, KLM E&M can use this model to reduce storage costs for all components. Secondly, the component usage of aircraft carriers is compared to see whose components last the longest.[2]

3 Research Question

For KLM Engineering & Maintenance it is important to know what the expected number of component removals will be. On the one hand, if KLM has not enough spare components it could mean that a broken component cannot be replaced instantly and would fail to meet the contract requirements. Because of failing contract requirements, aircraft carriers could end their contracts, which can cost KLM E&M a lot of money. Also, the delivery time of new components is long, therefore it is important to have enough spare components. On the other hand, components cost a lot of money, which means KLM would like to minimize the idle cost. When components are waiting to be placed in an aircraft, it means that KLM could have less inventory. This also reduces the cost. Therefore, it is important that the stock levels are not too high and not too low.[3] This results in the main objective of this study:

Develop a method for forecasting the number of removals

Furthermore, it is interesting for KLM to see how the numerous airlines take care of the components that they use. Therefore, a second research question is considered in this paper:

Make a ranking between aircraft carriers based on the component lifetimes

4 Data Inspection and Preparation

4.1 The Data

KLM E&M provided five data files regarding six component types:

- Summary Data 787 Pool v2
- Hours 787 Pool
- General Data 787 Pool
- Removal History 787 Pool
- Expected Pool Growth 787

The summary of the raw data, available for analysing the removals of the Boeing 787, has a time horizon of 17 months. This data file contains general information about the component pool. For instance, this file contains for each month the amount of aircraft in the pool, the number of contract hours, the number of components in the pool, and the number of removals. Also, the mean time between removals, as Boeing expects, is given for each component.

The next file, 'Hours 787 Pool', contains information about the aircraft of operators. For each month, the contract hours for each aircraft are given. This file resembles the 'General Data 787 Pool' file, except that the latter file does not contain contract hours, but the aircraft and engine type for every operator.

The only file containing raw data is 'Removal History 787 Pool'. This file contains the removals of components from aircraft, where for example information about date, component type, aircraft number and reason of removal are given. Since this is raw data, cleaning of the data will be necessary.

To predict the number of components in the pool in the future, the expected pool growth is useful information. This information is given in the 'Expected Pool Growth 787' file. The expected pool growth for each year is given in the range of 2017 until the year of 2025.

4.2 Data Inspection

During the data inspection and preparation, some bottlenecks of the data were discovered. First of all, the given data does not have a lot of data points. For example, the summary data contains 17 points for each component. This will result in a lower reliability of predictive models. It is hard to predict on a monthly basis but even harder to predict on a quarterly or yearly basis. Moreover, the amount of data points could make it harder to discover a connection between variables, which consequently makes it more difficult to create a predictive model. Also, it will be difficult to create a test and training set to test such a model.

Additionally, after investigating the 'Removal History 787 pool' file, it turned out that the number of removals is not consistent with the summary file. For example, the number of removals of component '870047' is in both files equal, whereas for component '870052' different amounts of removals are given. The decision is made to lay focus on the 'Removal History 787 pool' file and to ignore the summary file since the 'Removal History 787 pool' file provides more information. Furthermore, the assumption is made that a component is as good as new when re-entering in

the pool. This way there is no problem with the fact that interconnection between components is missing.

Finally, the data of the 'Removal History 787 pool' file can be split into two categories, because some components are broken down and removed, while others are still in use. The group 'uncensored data' contains all components for which a lifetime in flight hours is known. These lifetimes give information over the expected lifetime per component type and can be used immediately. The 'censored data' contains all components for which it is known when the component is installed in an aircraft, but unknown when the component will be removed. These components are still in use, so it is unclear how long these components will live. Two histograms are plotted in figure 1 to visualize the lifetimes of these groups for component '871765'. The other figures can be seen in the appendix in section A.1.

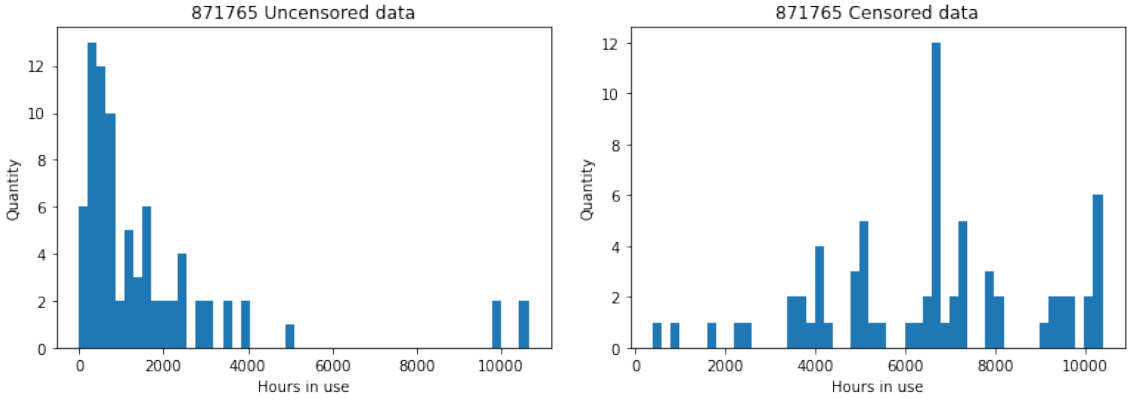


Figure 1: The censored data compared to the uncensored data

As figure 1 shows, the uncensored data contains a lot of components with a relatively short lifetime. The censored data contains longer lifetimes and could become even longer, considering that these components are still in use. It is plausible to say that the uncensored data does not reflect the underlying distribution. So it is possible that a method is needed that combines the uncensored data and the censored data.

It is also visible in figure 1 that there are zero events. That means that some components fail immediately after they are installed in the aircraft. Those events will not be taken into account, because they are unrelated to the failure rate. When these events are used, the distribution will have more weight to the left and the expected amount of removals will be too high with respect to the true amount of removals.

4.3 Data Cleaning

A lot of missing values are encountered in the 'Removal History 787 pool' file. The rows where the column 'DATIS' is unknown are removed. This column contains the date when a specific component was installed on an aircraft. Without these values a row of data becomes useless, since it is impossible to know how long the component has been installed on the aircraft.

Furthermore, a lot of missing values were found in the column 'DATUS'. This column contains the date when a specific component was removed from an aircraft. Because some components are still in the aircraft this value is missing, so the missing values are updated and set on 2017-04-01 ('yyyy-mm-dd'). This date is appropriate, since it is the last date of which it is known that the components are still in use. The difference between an unknown and known removal date results in two types of groups as mentioned before: the censored and the uncensored data.

4.4 Data Preparation

To forecast the amount of removals during a period, the lifetime of components should be estimated. This lifetime is defined as the amount of hours a component has flown. The amount of hours can be calculated using two data files: the 'Removal History 787 pool' file and the 'Hours 787 Pool' file. The hours are calculated by multiplying the number of days that a component has been installed in a specific month in a specific aircraft with the average contract hours a day in that month. By taking the sum of this for the whole time horizon, a reliable estimate of the hours flown is computed.

The hours flown are calculated based on the provided aircraft information. However, the aircraft information data is not complete. For example, there is no information in the given data about October 2016 and a specific aircraft has no data about January 2016. Moreover, the removal date in the 'Removal History 787 Pool' file starts often earlier than the given hours flown in a month for an aircraft in the 'Hours 787 Pool' file. Therefore, the method has to manually recognise these special cases with unique statements.

There could be more or less special cases in the future, if other data values are missing or missing values are added. Therefore, the 'Hours 787 Pool' file has to be inspected to discover the special cases before the hours flown are calculated. That is because when data is inserted with information about October 2016, the current method would still assume no information exists about this month. Thus, to use the method one needs to carefully look which special cases no longer exist or begin to exist and rewrite the method accordingly.

5 Methodologies

It is assumed that the amount of removals occurring is following a distribution. Using this assumption it is possible to create a forecasting model based on that distribution. This model will be made using the programming language Python. KLM can use the correct model in the future if a distribution between the amount of removals is discovered. For now it is hard to choose the correct model since little data is known. The following paragraphs describe methods and distributions that will be analysed to see if the data follows a certain distribution.

5.1 Likelihood estimators

Censored data still contains information. The only question is how to extract this information. To extract this information, likelihood estimators can be used. A likelihood estimator is a method that estimates the parameters of a certain distribution. An assumption is made for this distribution. In this study the exponential distribution, the Weibull distribution and the log-normal distribution are investigated. These are discussed respectively in section 5.2, 5.4 and 5.3. For the exponential distribution the maximum likelihood estimator is chosen, since it has an explicit solution. This will be further explained in section 5.2. For the other two distributions a log-likelihood estimator is chosen, since that is easier to implement. Since the log-likelihood estimator covers two distributions, it will be explained in this section.

First, the log-likelihood function for both the uncensored and censored data has to be determined. Since the uncensored data represents the full observations, the log-likelihood function can be determined as:

$$\log(L(\theta; x_1, \cdots, x_n)) = \sum_{i=1}^{n} \log(f(x_i \mid \theta)) \tag{1}$$

Where θ is the set of parameters, x_i an uncensored observation and f the probability density function of the underlying distribution. The censored data, denoted as Y, can be written as $P(Y > t)$. That is because it is the probability that a component survives longer than t, where t is the time stamp on which the two datasets are divided. Therefore the log-likelihood function of the censored data can be written as:

$$\log(L(\theta; y_1, \cdots, y_m)) = \sum_{i=1}^{m} \log(1 - F(y_i \mid \theta)) \tag{2}$$

Where y_i is the current lifetime of a censored observation. These functions (1) and (2) can be combined by simply taking the summation of both the likelihood functions. The combination of these functions is solved numerically, by setting up a possible parameter space. A parameter space contains all possible combinations of values for all the different parameters of the function. This space is set as broad as possible with the smallest difference between each number, to compare as many combinations as possible and therefore to be more precise. However, the higher the amount of possible parameters are, the longer the duration of the algorithm will be. By choosing a parameter space, it is important that the parameter space is not too narrow, otherwise the answer will not be accurate. On the other hand, the parameter space should also not be too broad, since that will increase the running time enormously.[4][5]

5.2 Exponential distribution

The exponential distribution is a continuous probability distribution which has a lot of useful characteristics, for example its memorylessness. Furthermore the distribution only has one parameter μ, which is simpler to estimate. The distribution reflects lifetime probabilities of cases where there is no wear. This is probably not the case, but the simple implementation and the useful characteristics are still worth an investigation of this distribution.

The same logic is applied as in section 5.1, where the likelihood function is obtained, only now a maximum likelihood estimator is used. The likelihood functions then become:

$$L(\mu; x_1, \cdots, x_n) = \prod_{i=1}^{n} f(x_i \mid \mu) \tag{3}$$

$$L(\mu; y_1, \cdots, y_m) = \prod_{i=1}^{m} f(y_i \mid \mu) \tag{4}$$

Where μ is the mean. The functions (3) and (4) can be combined by multiplication. By solving this function, the maximum of this function is needed. Therefore, the function needs to be differentiated en set equal to zero. The function for an estimation of μ eventually becomes:

$$\hat{\mu} = \frac{x_1 + \cdots + x_n + y_1 + \cdots + y_m}{n} \tag{5}$$

5.3 Log-normal distribution

A log-normal distribution is a continuous probability distribution of a random variable whose logarithm is normally distributed. Thus, if the random variable X is log-normally distributed, then $Y = \ln(X)$ has a normal distribution. The probability density function is given by:

$$f_X(x) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-} \frac{(\ln x - \mu)^2}{2\sigma^2} \tag{6}$$

Where σ and μ are the parameters.The cumulative distribution function is:

$$F_X(x) = \Phi(\frac{\ln(x) - \mu}{\sigma}) \tag{7}$$

Where Φ is the cumulative density function of the standard normal distribution, that is with $\Phi \sim N(0, 1)$. The distribution is commonly used in reliability engineering. It is mostly used for microelectronic wear-out failures. [7] Although the KLM E&M components are mechanic and not microelectronic, the components do wear. Therefore, the distribution is worth an investigation.

5.4 Weibull distribution

The Weibull distribution is a continuous probability distribution. The parameters λ>0 and k>0 provide the scale and shape respectively. The estimation for k is that k<1, which means that the failure rate decreases over time. This would mean that if a component is older than a certain age,

it will probably not be broken for a long time. The probability density function of the Weibull distribution is:

$$f(x;\lambda,k) = \begin{cases} \frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} e^{-(x/\lambda)^k} & x \geq 0, \\ 0 & x < 0, \end{cases} \tag{8}$$

In both this formula (8) and the following formula (9), x is the location parameter. The cumulative distribution function for the Weibull distribution is:

$$F(x;\lambda,k) = \begin{cases} 1 - e^{-(x/\lambda)^k} & x \geq 0, \\ 0 & x < 0, \end{cases} \tag{9}$$

The Weibull distribution is a widely used lifetime distribution in reliability engineering. It is a versatile distribution that can take on characteristics of other types of distributions depending on the values of its parameters.[6] Due to this property and the fact that it can describe components that wear, the Weibull distribution is often used to model component lifetimes.

5.5 Best Fit

When using the best fit method, the data is compared with a list of numerous distributions. A distribution is fitted as close as possible to the data with its parameters. After this, the sum squared error is calculated. The sum of squared errors of prediction (SSE) is used, which measures the overall difference between the data and the values predicted by the distribution with its parameters.

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{10}$$

Here n is the number of observations, y_i is i^{th} observed value of the variable and $\hat{y}_i$ is the predicted value of y_i. The SSE represents unexplained variation, therefore the distribution with the smallest sum squared error is taken as the best fit. This results in a probability density function. After that, future values can be predicted.[8]

The input for the best fit method is the uncensored data, since the real lifetimes of the components in the censored data are unknown. However, the future values which need to be predicted are in the censored data. It might become a problem that the censored data seems to have a significantly larger lifetime than the uncensored data, as shown earlier in figure 1.

5.6 Aircraft carriers comparison

For KLM E&M it is important to know which aircraft carrier needs more replacements than other carriers. If KLM E&M knows for instance that some carriers need replacement more often than others, they can increase the cost for this carrier to be a member of the component pool. To investigate which carrier uses its components longer in comparison to other carriers, the average of the hours flown is calculated for each component per carrier.

Some carriers have zero hours flown with a component. It is unknown whether this information is missing or that this carrier is not using this component in the KLM E&M pool. Since this remains unknown, two averages need to be calculated: one normal average and one average where all zero values are removed.

5.7 Predicting removals

In the previous sections, methods are discussed that are useful to discover a distribution that follows the underlying data. The second part of this research is distribution-independent, because it uses such a distribution as a parameter. Further on, the probability density function is denoted as f and the cumulative density function as F. In this section a model for predicting the amount of removals is discussed.

On time stamp T, where T is the future date where to a prediction is demanded, a component can be in one of two different states. Those two states are:

$$X_i = \begin{cases} 1 & \text{broken at time T with } p_i \\ 0 & \text{works at time T with } 1 - p_i \end{cases} \tag{11}$$

X_i is the state of component i, with $i = 1, \cdots, n$. Here n is the number of components. It is clear that the X_i follows a Bernoulli distribution. So, the failure events have a density function. The probability p_i, needed for the Bernoulli distribution, is defined as:

$$p_i = P(\text{component i is broken at time T}) = P(L_i < T) = P(X_i = 1) \tag{12}$$

Where L_i is defined as the lifetime in flight hours of component i, with $i = 1, \cdots, n$, and $L_i \geq 0$. To predict the amount of removals, the residual probability needs to be calculated. To do this, another variable is needed. The current lifetime is known for the working components, therefore we define a new variable t_i. This is the time that component i has already flown. After conditioning on the event $L_i > t_i$, the residual probability can be written as:

$$\begin{aligned} q_i &= P(\text{component } i \text{ is broken at time } T \mid \text{component } i \text{ works at time } t_i) \\ &= P(L_i < T | L_i > t_i) = \frac{F_i(T) - F_i(t_i)}{1 - F_i(t_i)} \end{aligned} \tag{13}$$

The probabilities of function (13) can be calculated with the cumulative density function F corresponding to component i.

At this moment only the probability that one component breaks down is known. With the central limit theorem it is possible to determine the expected value and the standard value. The central limit theorem states that as summation of identically independent distributed (i.i.d.) random variables and normalising them accordingly, than the limit follows a normal distribution. However, special cases exist where the variables are not required to be i.i.d.. One such sufficient condition is called the Lyapunov condition (14). Here s_n is denoted as the sample standard deviation.

$$For\ every\ \epsilon > 0,\ \frac{1}{s_n^2} \sum_{k=1}^{n} \mathbf{E}(Y_{nk}^2 I\{|Y_{nk}| \geq \epsilon s_n\}) \to 0\ as\ n\ \to \infty \tag{14}$$

Given the example provided by source [9], the data satisfies the Lyapunov condition.
$X_i \sim Bernoulli(p_i)$ with $X_1, X_2, ...$ independent. Let $Y_{nk} = X_k - p_k$. Then for any $\delta > 0$:

$$1 \geq p_k(1 - p_k) = \mathbf{E}\, Y_{nk}^2 \geq \mathbf{E}|Y_{nk}|^{2+\delta} \tag{15}$$

From conditions (14) and (15) follows:

$$\frac{1}{s_n^{2+\delta}} \sum_{k=1}^{n} \mathbf{E}|Y_{nk}|^{2+\delta} \leq \frac{1}{s_n^{2+\delta}} \sum_{k=1}^{n} Var\, Y_{nk} = \frac{1}{s_n^{\delta}} \tag{16}$$

Therefore, the Lyapunov condition (14) is satisfied if $s_n \to \infty$ which is true if p_k is not equal to either 0 or 1. Which means that the condition is satisfied, therefore the CLT can be used. Thus, $\sum_{k=1}^{n} Y_{nk}/S_n \to N(0,1)$.[9] The CLT gains strength as the size n increases. This is viable for $n \geq 30$, which is the case for each component.[10] The expected value can be calculated as follows:

$$E[X_1 + \cdots + X_n] = E[X_1] + \cdots + E[X_n] \tag{17}$$

Where $E[X_i]$ is equal to the expectation of a Bernoulli distribution, in this case $1 - p_i$. A confidence interval can be calculated if the variance is known. The variance can be calculated as follows:

$$Var[X_1 + \cdots + X_n] = Var[X_1] + \cdots + Var[X_n] \tag{18}$$

Where $Var[X_i]$ is equal to $p_i(1 - p_i)$.[11] With the expected value and the variance it is possible to calculate the confidence interval for a certain confidence level α.

The confidence level α is slightly different than the desired confidence level. The desired confidence interval is the upper bound. It is important to know how much components are needed to survive. A confidence interval with $\alpha = 0.90$ gives the expected removals between the lowest and the highest 5%. Therefore, it gives an upper bound for a confidence level of 0.95, since the lowest 5% are also covered in this case.

6 Results

To check which distribution fits the data best, an expected range of the amount of removals is calculated using the given data. First the average over the summary data is calculated per component. Next, the difference between the given amount of failures per month and the average failures is taken and squared. For the standard deviation the square-root of the sum of all the differences is calculated.

$$Standarddeviation = \sqrt{\frac{\sum(X_i - X)^2}{N_x}} * 3 \tag{19}$$

Where X_i is the repaired number of components in the summary file, X is the average of repaired components, and N_x is the number of data points in the summary file. To get a three-month estimation the standard deviation is multiplied by 3.

To give an estimation for the amount of failures, the average of 3 months minus the standard deviation is taken as lower bound and the average of 3 months plus the standard deviation is taken as upper bound. The values are rounded to the nearest integer.

6.1 Discarded Method: Best Fit

The best fit method does not provide a good estimate, since the provided dataset is of a too small time horizon. The method will choose a distribution that fits the data best and due to this, the distribution will be overfitting the data.

6.2 Likelihood estimators

In the table 1 below, the expected number of removals of each considered distribution is posted for the time frame of 3 months.

Component	**Range of expected removals**	**Exponential Distribution**	**Log-normal Distribution**	**Weibull Distribution**
870047	[0.00 - 14.41]	6.48	4.47	5.34
870052	[1.40 - 13.42]	7.55	6.22	7.06
870180	[17.57 - 29.72]	18.10	8.18	17.74
871765	[0.00 - 20.53]	10.30	5.36	6.08
888003	[12.05 - 36.65]	19.85	10.53	15.60
888004	[4.08 - 9.34]	4.92	2.96	4.62

Table 1: The predicted removals using different underlying distributions compared, from 01-04-2017 until 01-07-2017.

The range of expected removals is calculated based on a 3 month average from the summary file plus and minus the standard deviation.

As said before, the accuracy of the model depends on the size of the parameter space, since the estimation of the parameters depends on the parameter space. For these results, the size of the parameter space is 100.

Because the distributions are close or in the range of calculated expected removals, the assumption is made that the data follows either a Weibull, a Log-normal or an Exponential distribution. The expected number of removals resemble the data well, see table 1. Also, the failure rate functions that they produce, resemble the data well. Figure 2 shows the failure rate functions of the distributions, and the uncensored data. It seems that the distributions are plausible. In addition, the Weibull distribution is an often used distribution for predicting failure rates. In the figure only the uncensored data is used, since the censored data is unusable for this kind of comparison.

The censored data has a higher density on the right side of the plot, while the density of the uncensored data is higher on the left side. The parameters of all three distributions are estimated using both data sets. Therefore, all the distributions contain a more significant tail on the right side than the uncensored data. The figures of other component types can be found in the appendix in section A.2.

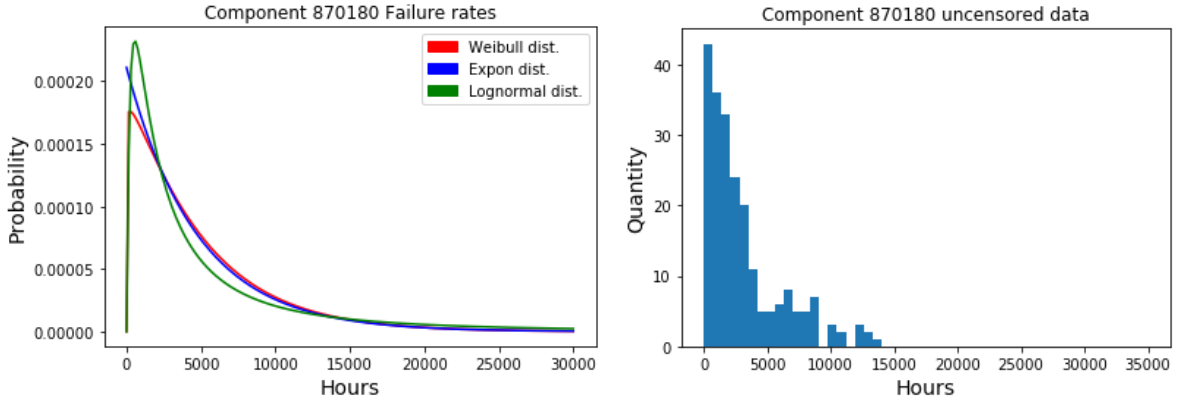


Figure 2: Calculated failure rate functions of distributions and the uncensored data.

6.3 Expected removals for 3 months

In the next subsections the results from the Weibull, Exponential and the Log-normal distribution are displayed.

6.3.1 Weibull distribution

Table 2 shows the results for the expected removals for three months for each component, for the period of 01-04-2017 to 01-07-2017. For these results the size of the parameter space is 1000. The parameters for the Weibull distribution are determined by calculating the maximum likelihood, see section 5.1.

Component	**Mean Removals**	**95% upper bound CI**	**Rounding Down**		**Rounding Up**	
			Removals	Conf. level	Removals	Conf. level
870047	5.34	8.97	8	88.6%	9	95.2%
870052	7.07	11.25	11	93.9%	12	97.4%
870180	17.74	23.86	23	92.1%	24	95.4%
871765	6.09	9.96	9	89.2%	10	95.2%
888003	15.53	21.45	21	93.6%	22	96.4%
888004	4.65	7.89	7	88.4%	8	95.6%

Table 2: Expected removals from 01-04-2017 until 01-07-2017.

Table 3 also shows the results for the expected removals for three months for each component, for the period of 01-04-2017 to 01-07-2017. However, this time the parameter space is 100.

Component	**Mean Removals**	**95% upper bound CI**	**Rounding Down**		**Rounding Up**	
			Removals	Conf. level	Removals	Conf. level
870047	5.34	8.97	8	88.6%	9	95.1%
870052	7.06	11.24	11	94.0%	12	97.4%
870180	17.74	23.86	23	92.1%	24	95.4%
871765	6.08	9.94	9	89.3%	10	95.3%
888003	15.60	21.53	21	93.3%	22	96.2%
888004	4.65	7.89	7	88.4%	8	95.6%

Table 3: Expected removals from 01-04-2017 until 01-07-2017.

As visible in tables 2 and 3, the difference between the amount of removals using a parameter space of respectively 1000 and 100 is small. However, the run time differs a lot. The model is executed in approximately 6376 seconds (106 minutes) with a parameter space of 1000, while the run time with parameter space 100 runs in approximately 81 seconds. Therefore, a parameter space of 100 is recommended.

The mean removals column shows the expected amount of removals in the time period. The third column shows the upper bound of a confidence interval, which is the confidence level. The amount

of removals in the stated period of time will be equal or less than this upper bound in 95% of the cases. Since it is impossible to have a partially removal, the number of removals needs to be integer. Table 3 also shows the confidence level in case when the amount of removals is rounded up or down. This can be helpful to determine the number of spare components needed.

The amount of removals for other confidence levels can also be determined. Table 4 shows the amount of removals per component type for different confidence levels. It is clear that the higher the confidence level, the higher the amount of removals.

Component	**90% Confidence level**	**92.5% Confidence level**	**95% Confidence level**	**97.5% Confidence level**	**99% Confidence level**
870047	8.17	8.52	8.97	9.67	10.48
870052	10.32	10.72	11.24	12.04	12.97
870180	22.51	23.1	23.86	25.03	26.4
871765	9.09	9.46	9.94	10.68	11.54
888003	20.22	20.79	21.53	22.67	23.99
888004	7.14	7.45	7.85	8.47	9.19

Table 4: Amount of removals for different confidence levels from 01-04-2017 until 01-07-2017.

It is also possible to determine the amount of removals for every confidence level, using plots. Figure 3 shows the amount of removals against the confidence level, for component type 870180. It is visible that the higher the confidence level is, the higher the amount of removals predicted. This seems accurate, because a confidence level is the percentage of cases lower or equal to the amount of removals corresponding to that confidence level. For example, the figure shows that a confidence level of 0.95 (95%) corresponds to roughly 24 removals. This means that 95% of the cases is lower or equal to 24 removals. Therefore, a higher percentage than 95% will correspond to more removals than 24. A confidence level of 100% will result in an enormous amount of removals, since it is theoretically impossible to cover 100% of the cases.

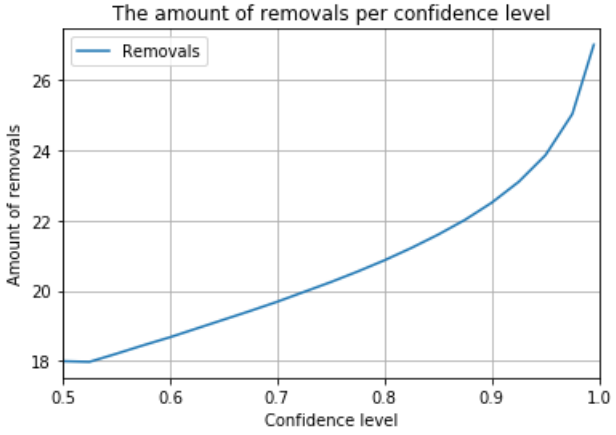


Figure 3: Amount of removals against confidence levels for component type 870180.

In the appendix, paragraph A.3, the figures of the other component types are added.

6.3.2 Exponential distribution

The same analysis done for the results from the Weibull distribution, has been done for the results from the Exponential distribution. Table 5 shows the results for the expected removals for three months for each component, for the period of 01-04-2017 to 01-07-2017 and parameter space 100. The method for determining the parameter for the exponential distribution is shown in section 5.2.

Component	**Mean Removals**	**95% upper bound CI**	**Rounding Down**		**Rounding Up**	
			Removals	Conf. level	Removals	Conf. level
870047	6.48	10.43	10	92.9%	11	97.0%
870052	7.55	11.85	11	90.6%	12	95.5%
870180	18.10	24.27	24	94.2%	25	96.7%
871765	10.30	15.16	15	94.4%	16	97.3%
888003	19.85	26.38	26	93.9%	27	96.4%
888004	4.92	8.24	8	93.7%	9	97.8%

Table 5: Expected removals from 01-04-2017 until 01-07-2017.

Again, the amount of removals for other confidence levels can also be determined. Table 6 shows the amount of removals per component type for different confidence levels. It is clear that the higher the confidence level, the higher the amount of removals.

Component	**90% Confidence level**	**92.5% Confidence level**	**95% Confidence level**	**97.5% Confidence level**	**99% Confidence level**
870047	9.56	9.94	10.43	11.19	12.07
870052	10.90	11.32	11.85	12.68	13.64
870180	22.90	23.50	24.27	25.45	26.82
871765	14.09	14.55	15.16	16.09	17.18
888003	24.94	25.57	26.38	27.63	29.09
888004	7.51	7.82	8.24	8.87	9.61

Table 6: Amount of removals for different confidence levels from 01-04-2017 until 01-07-2017.

It is also possible to plot the amount of removals for every confidence level. Figure 4 shows the amount of removals against the confidence level, for component type 870180.

In the appendix, paragraph A.4, the figures of the other component types for the Exponential distribution are added.

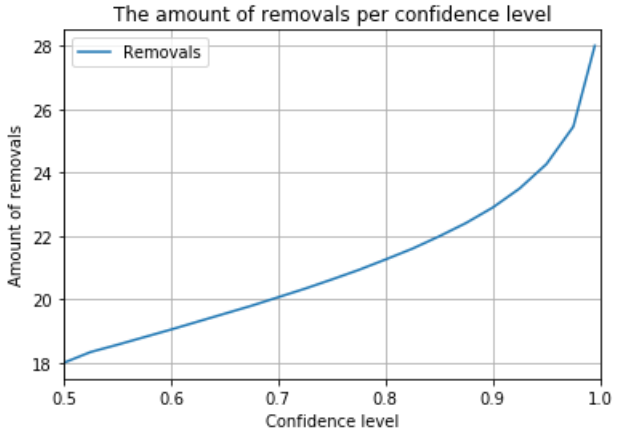


Figure 4: Amount of removals against confidence levels for component type 870180.

6.3.3 Log-normal distribution

And the same analysis has also been done for the log-normal distribution. Table 7 shows the results for the expected removals for three months for each component, for the period of 01-04-2017 to 01-07-2017 and parameter space 100. The parameters for the log-normal distribution are determined by calculating the maximum likelihood, see section 5.1.

Component	**Mean Removals**	**95% upper bound CI**	**Rounding Down**		**Rounding Up**	
			Removals	Conf. level	Removals	Conf. level
870047	4.47	7.81	7	89.3%	8	95.9%
870052	6.22	10.14	10	94.4%	11	97.8%
870180	8.18	12.62	12	92.2%	13	96.3%
871765	5.36	9.01	9	94.9%	10	98.2%
888003	10.53	15.56	15	92.8%	16	96.3%
888004	4.92	8.24	8	93.7%	9	97.8%

Table 7: Expected removals from 01-04-2017 until 01-07-2017.

The amount of removals for other confidence levels can also be determined. Table 8 shows the amount of removals per component type for different confidence levels. It is clear that the higher the confidence level, the higher the amount of removals.

Component	90% Confidence level	92.5% Confidence level	95% Confidence level	97.5% Confidence level	99% Confidence level
870047	7.08	7.40	7.81	8.45	9.2
870052	9.27	9.65	10.14	10.89	11.76
870180	11.64	12.06	12.62	13.47	14.46
871765	8.21	8.56	9.01	9.71	10.53
888003	14.45	14.93	15.56	16.52	17.64
888004	5.04	5.29	5.63	6.14	6.73

Table 8: Amount of removals for different confidence levels from 01-04-2017 until 01-07-2017.

It is also possible to plot the amount of removals for every confidence level. Figure 5 shows the amount of removals against the confidence level, for component type 870180.

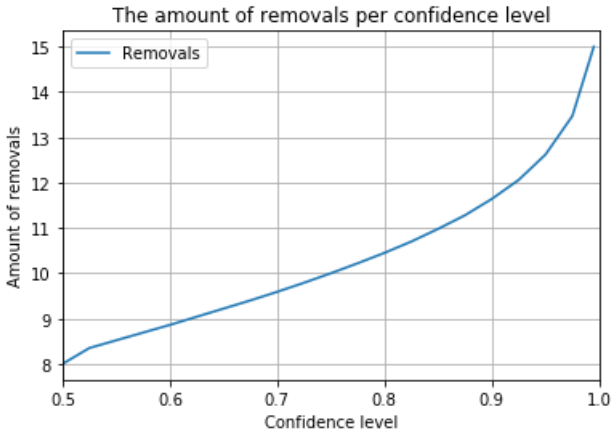


Figure 5: Amount of removals against confidence levels for component type 870180.

In the appendix, paragraph A.5, the figures of the other component types for the Log-normal distribution are added.

6.3.4 Aircraft carriers comparison

As stated in 5.6, it is useful to know which aircraft carrier needs more component replacements than other carriers for KLM E&M. The replacements depend on the lifetime of components. Two averages are calculated: one normal average (red in the figures) and one average where all zero values are removed (green in the figures). The plotted values are the average hours flown per component per carrier. When a carrier performs worse than the average, the carrier will be on the left side of the line.

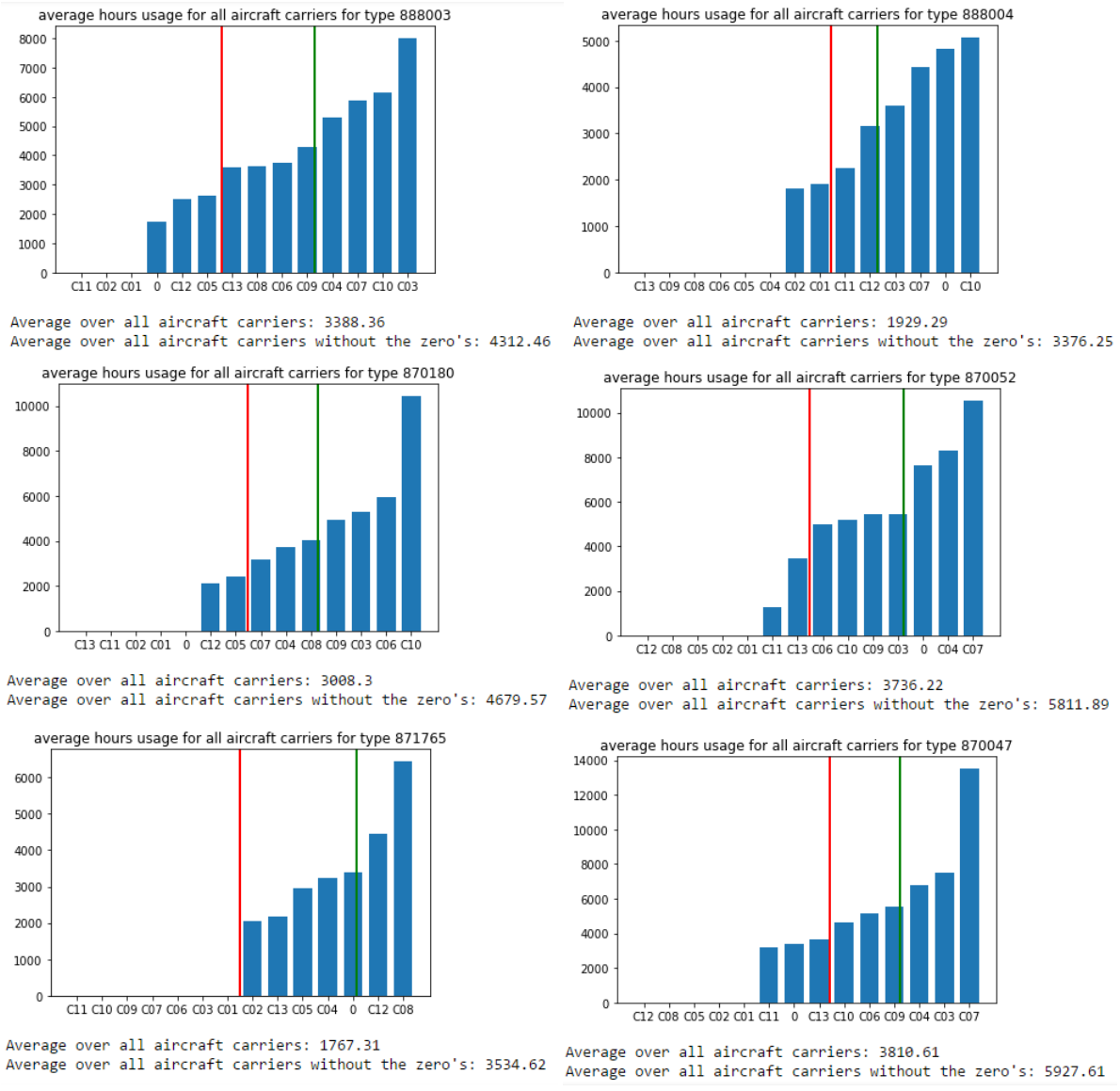


Figure 6: The aircraft carriers compared to each other.

The figures show one remarkable thing: some carriers have zero hours for a component. Therefore, the average (the red line) is not consistent when these values are not inserted but do exist. This is why there is also a green line, the average when all the zero values are removed. The values of the averages are given underneath the figures.

When the amount of times a carrier is better than 'the average without zeros' is calculated, the following results were found:

Aircraft Carrier	**Amount of times under average**	**Amount of times above average**	**Amount of times 'zero'**	**Ranking**
0	3	2	1	-
C01	1	0	5	9
C02	2	0	4	10
C03	1	4	1	1
C04	2	3	1	3
C05	3	0	3	11
C06	3	1	2	5
C07	1	4	1	2
C08	2	1	3	8
C09	3	1	2	6
C10	2	3	1	4
C11	3	0	3	12
C12	3	1	2	7
C13	4	0	2	13

Table 9: Performance per carrier

One point of consideration should be mentioned. There is a carrier called '0'. When no carrier was given, it obtained the code '0', because the value was missing. It is in our believe that removing this data would give more harm, then to give it the value '0'. It could mean that the average from one of the carriers is higher (or lower) because this value is not included. Some carriers, like C11, have a lack of information about three components, which could mean that these missing values are now calculated as the '0' carrier. When more than one carrier was not entered correctly for a component, the value '0' was assigned more than once. Then '0' is a combination of multiple carriers. To see which aircraft carrier performs better, a ranking is made. In this ranking '1' is the best and '13' is the worst. The aircraft carriers are first rated on the amount of times a carrier performed above average, next the carriers are ranked on the amount of times a carrier is under average. As a last ranking the amount of zero's is used. This order is chosen because the zero's are unknown whether or not they are good, bad or do not exist.

Based on the problem above, it is difficult to say which aircraft carrier is better than other carriers. It could be that in reality the average of the hours flown per carrier are different. Therefore, a conclusion cannot be made with certainty. While inspecting the results in table 9, it seems that aircraft carrier C03, C04 and C07 have a good performance. They have the highest ranking, since they have for the most components a lifetime above the average. Also, it seems that for instance carrier C05, C11 and C13 are performing the worst; they score at least three times below the average. However, due to the incomplete data, this cannot be concluded with certainty.

7 Conclusion

In this conclusion the following two questions will be answered:

- *Develop a method for forecasting the number of removals*
- *Make a ranking between aircraft carriers based on the component lifetimes*

7.1 The model

A model is created to forecast the number of removals in a certain time period. This is done by finding an underlying distribution with fitting parameters. After that, the amount of removals is predicted by calculating the residual probability using conditional probabilities. Then the mean and variance are determined using a special variant of the central limit theorem: the Lyapunov condition. For this condition the variables do not need to be identically independent distributed. Finally, the mean and variance can be used to predict the amount of removals for a given confidence level. A plot can be made for each confidence level and each component type to show the relation between the confidence level and the amount of removals.

7.2 Expected Removals

The expected removals for any defined period can be predicted, using the created model. A parameter space of 100 is recommended when using the model, since that provides accurate results but a short run time. Based on the removals from the provided data, the expected removals for the period 01-04-2017 until 01-07-2017 are predicted for the Weibull, Log-normal and Exponential distribution, since all three distributions resemble the data well. Also, the amount of removals for different confidence levels are calculated.

7.3 Aircraft carriers

The conclusion about which aircraft carriers perform well or badly is hard to make, due to missing values. Ranking the carriers from 1 to 13 helps with this, 1 being the best and 13 being the worst. Using the available data, the carriers C03, C04 and C07 seem to perform the best, while carriers C05, C11 and C13 perform the worst, see table 10. However, this conclusion cannot be made with certainty, because of the missing values.

Aircraft Carrier	**Ranking**
C03	1
C07	2
C04	3
C10	4
C06	5
C09	6
C12	7
C08	8
C01	9
C02	10
C05	11
C11	12
C13	13

Table 10: Carrier performance ranking

8 Discussion

A lot of time was spend on finding an underlying distribution of the data, since the data is of a very small time horizon. In the future, this will no longer be an obstacle, since more data will be available then.

As explained in the Data Inspection & Preparation sections, a lot of mistakes were found in the data. For example, the date for when a component was installed on an aircraft is missing multiple times. To optimize a future model, it is important that the data on which the model is based, is reliable. So a solid data collection by KLM Engineering & Management is an important factor. A way to do this is making the data complete before it is used as input for the model. It will differ a lot, since the model used in this study has to make a lot of exceptions in order to deal with incompleteness in the data.

Also, the model that is used can predict what the expected number of removals will be for each component. It can predict stock levels for a short time horizon if repair times are long. It is unable to predict stock levels over a longer time horizon. The reason for this is that the model assumes that when a component is removed, it does not return to the component pool. For this, the average repair time is needed, which was unavailable during this research. Therefore, to make a prediction for a longer time horizon, an adjustment is needed to predict the stock level for each component.

To know for sure whether or not the model gives a good estimate, a test set is needed. To make a test set the data should be split, which does not seem like a good idea with so little data. The model would make a forecast of the training set which is even smaller than the original small set. What could work to test the model is for KLM to generate the data for the months April, May and June and make this the test set.

Additionally, due to the fact that the dataset is small, it is important to update the dataset each month and use the model on the updated dataset. By doing this, the prediction of the expected number of removals becomes increasingly more accurate.

The size of the pool will increase over time. The model will take this into account automatically, since the model needs the *Removal history file*. It is not possible to take into account components that have not been placed yet, since the time a component has already flown is necessary. The file 'Expected Pool Growth' was provided and could be useful for this, however this data is an expectation. Using an expectation to make an expectation could make the output unreliable and inaccurate. As a result of the points mentioned above, the choice was made to make the model as it currently is.

As seen in the results, the expected removals for each distribution do not differ a lot. They all fall in the range of the expected outcome. Nevertheless, it is necessary that the prediction is accurate. Any statistical test should use both data types, otherwise such test could possibly lead to wrong conclusions. No statistical test is performed on each of the used distribution. The reason for this is that the estimated distribution is estimated on two different types of data, the uncensored data and the censored data. When more uncensored data is available, statistical tests can confirm or reject a certain distribution. The use of a test set could also solve this problem, but as explained before this was also not possible.

In the results, an interesting observation is done in the figures where the amount of removals is plotted against the confidence level. For unknown reasons, sometimes the amount of removals is higher for a confidence level of 0.5, than it is for a confidence level of 0.525, which is unexpected.

In a future study the reason of a removal can be investigated and what is the influence of such a reason on its lifetime. For this future study new data is necessary, since all the reasons of a removal are needed. With this information it could be possible to prevent a removal if some reasons are errors from humans.

Additionally another possible future study would be to alter the model such that components could break down multiple times instead of one time in the to be predicted time frame.

Also, the hours flown as some other methods could be more efficiently computed. Some methods in preprocessing the data could be rewritten to functions which the apply function in numpy could use. Furthermore the if statements in the method which calculates the hours flown for a component removal could be bypassed if a complete aircraft information DataFrame would exist. Therefore, preprocessing the aircraft information DataFrame such that this DataFrame is complete would make the preprocessing more time efficient.

Finally, on the topic comparing the aircraft carriers, the conclusion is inconclusive. It is unknown whether some carriers do not fly with certain components, or that data is missing. Therefore, the given data on which conclusions are drawn need to be analysed on completeness.

9 Literature and References

[1] Steyerberg, Ewout W. (October 21, 2010). Clinical Prediction Models. New York: Springer. p. 313. ISBN 1441926488

[2] Touati, Joshua. "Introduction KLM case", (2017). Scientific Writing in English

[3] 'VU Business case Air France KLM Component Services 787 Pool Forecasting', (2017), from https://bb.vu.nl/bbcswebdav/pid-3004947-dt-content-rid-6880057_2/courses/FEW_XB_41000_2016_142/VUBusinesscaseAirFranceKLM2017.pdf.

[4] Log likelihood, from http://webpages.cs.luc.edu/~jdg/w3teaching/stat_304/f05/pdf/likelihood.pdf

[5] Mitra, Debanjan, 'Likelihood inference for left truncated and right censored lifetime data', from https://pdfs.semanticscholar.org/5217/3279bd7a53427577664ed906d54def07563a.pdf

[6] Weibull distribution, http://www.weibull.com/hotwire/issue14/relbasics14.htm

[7] Log-normal or Weibull test, http://www.itl.nist.gov/div898/handbook/apr/section3/apr312.htm

[8] Linear regression, (March 28, 2013), http://www.stat.ufl.edu/~winner/statnotescomp/regression.pdf

[9] Fergurson, Lehman. CLT, Part II: Independent but not identically distributed http://sites.stat.psu.edu/~dhunter/asymp/fall2002/lectures/ln04.pdf

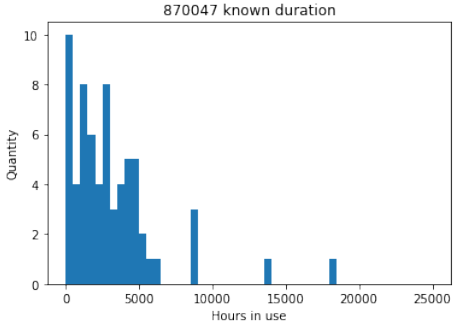
[10] Filmus, Yuval. 'Two Proofs of the Central Limit Theorem', (2010), from http://www.cs.toronto.edu/~yuvalf/CLT.pdf

[11] Johnson, Norman L., Kemp, Adrienne W., Kotz, Samuel, 'Univariate Discrete Distribution' (third edition), from http://ftp.yazd.ac.ir/FTP/E-Book/Statistical%20books/Distribution%20Theory/Univariate%20Discrete%20Distributions%20(Third%20Edition,%20Wiley%202005).pdf

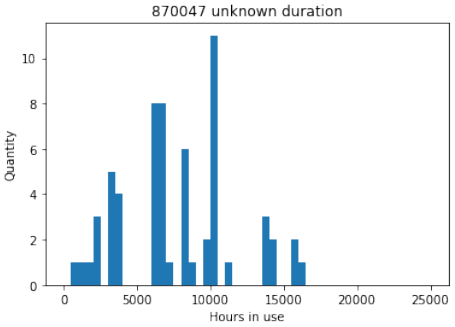
A Appendix

A.1 Censored & uncensored data

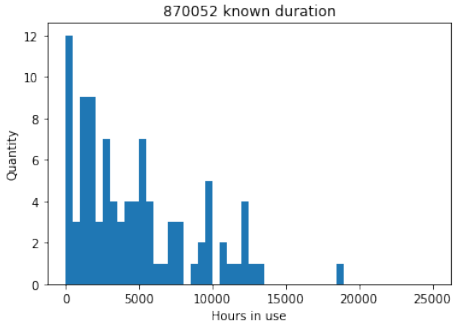
Here the censored data and the uncensored data are shown in histograms for each component type.



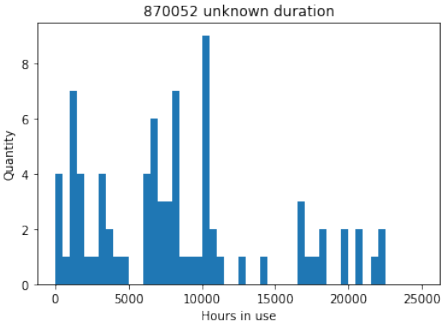
(a) Uncensored Data 870047



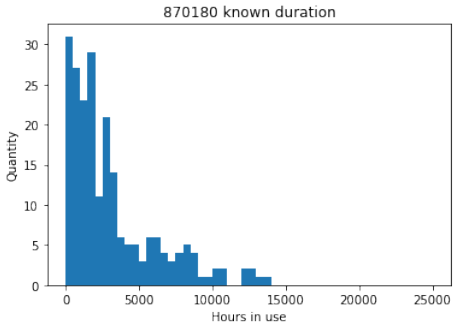
(b) Censored Data 870047



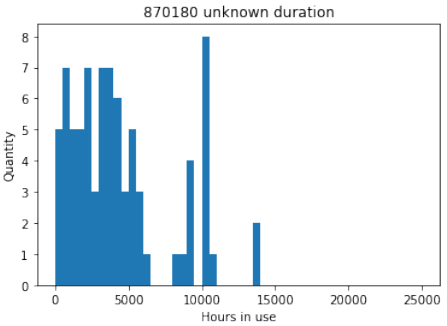
(c) Uncensored Data 870052



(d) Censored Data 870052

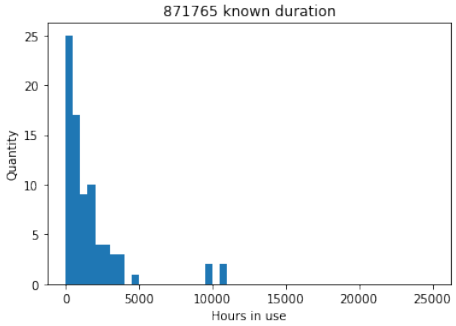


(e) Uncensored Data 870180

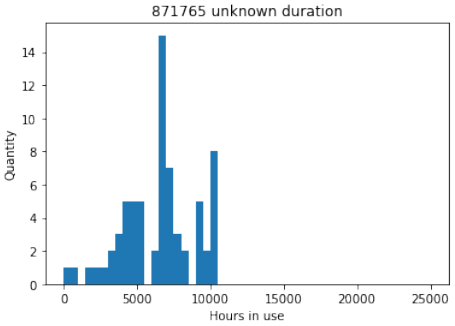


(f) Censored Data 870180

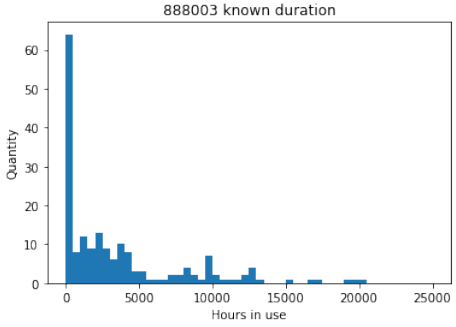
Figure 7: The hours flown for each component.



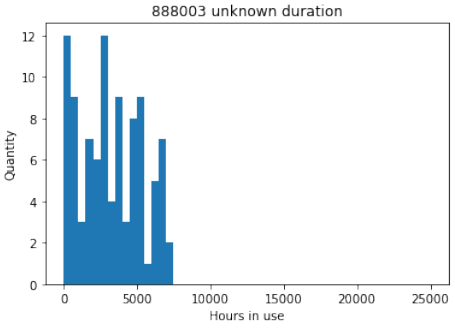
(a) Uncensored Data 871765



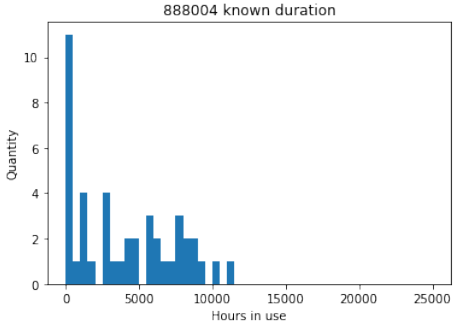
(b) Censored Data 871765



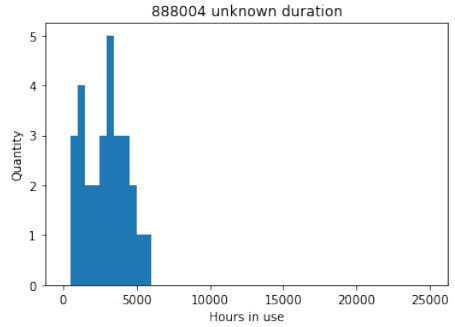
(c) Uncensored Data 888003



(d) Censored Data 888003



(e) Uncensored Data 888004



(f) Censored Data 888004

Figure 8: The hours flown for each component.

A.2 Distribution estimations & uncensored data

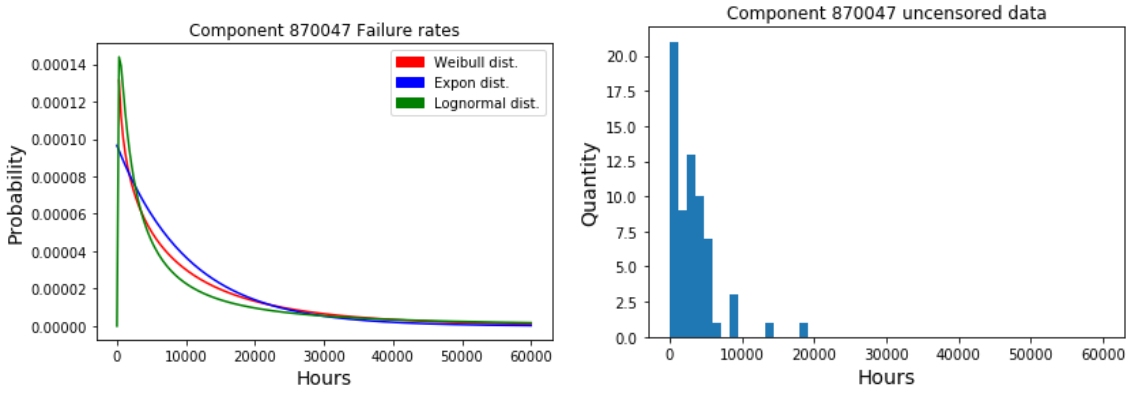


Figure 9: Calculated failure rate functions of the distributions versus the uncensored data.

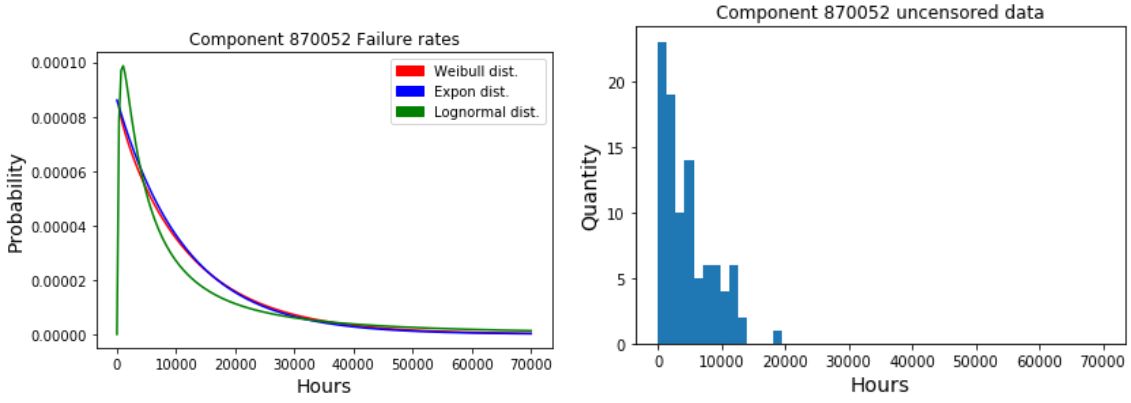


Figure 10: Calculated failure rate functions of the distributions versus the uncensored data.

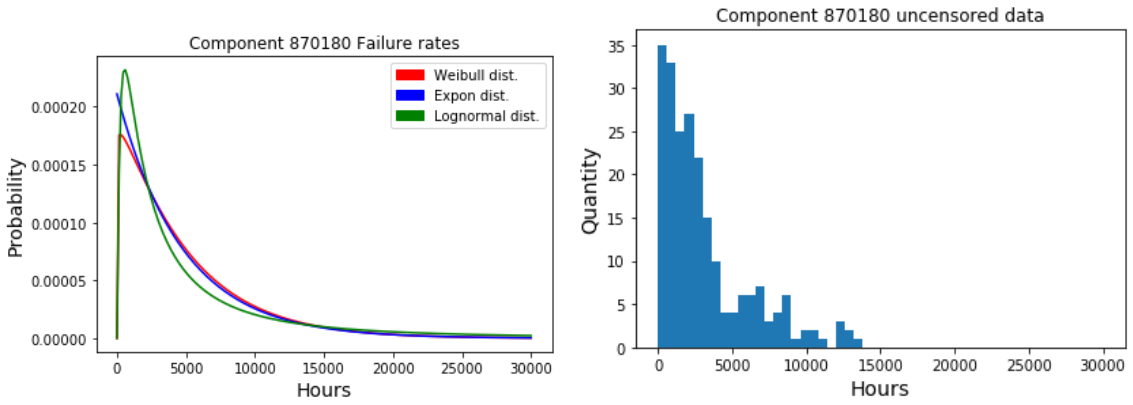


Figure 11: Calculated failure rate functions of the distributions versus the uncensored data.

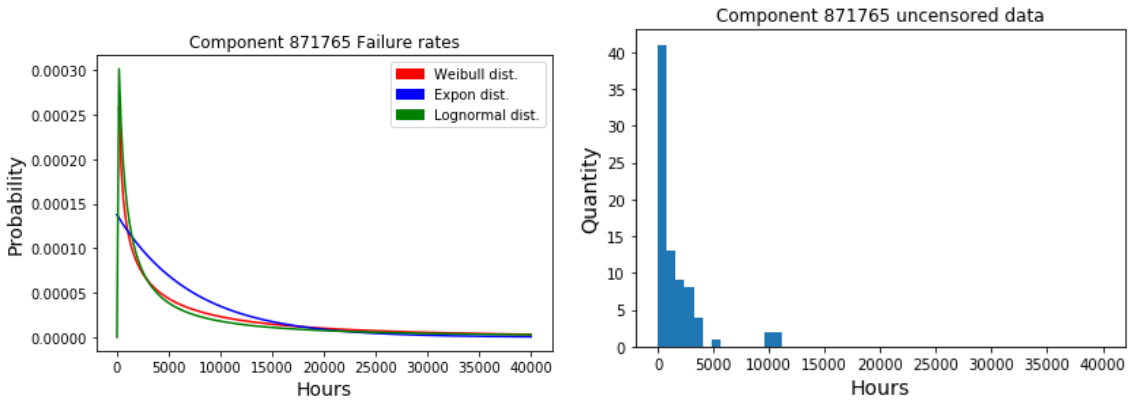


Figure 12: Calculated failure rate functions of the distributions versus the uncensored data.

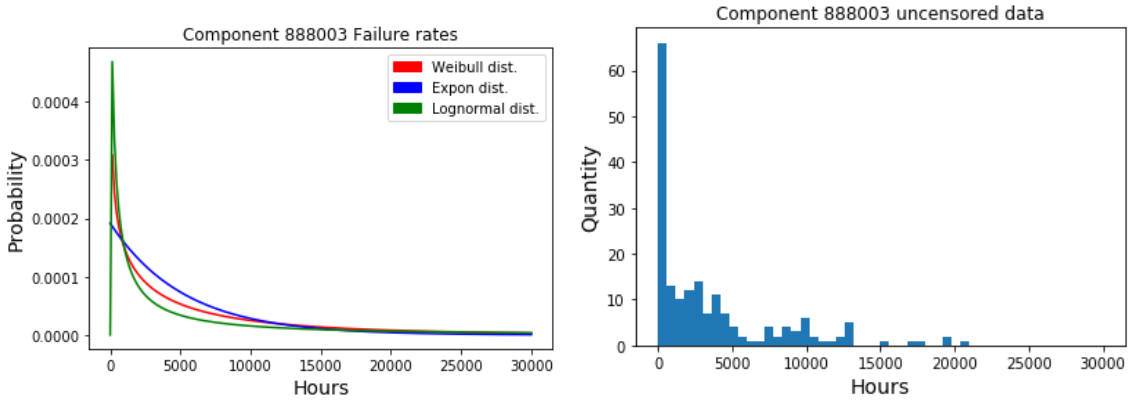


Figure 13: Calculated failure rate functions of the distributions versus the uncensored data.

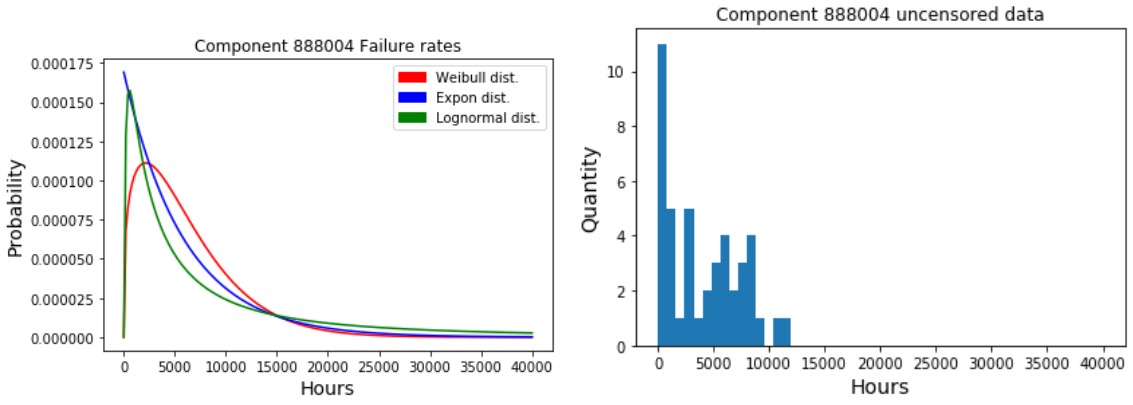
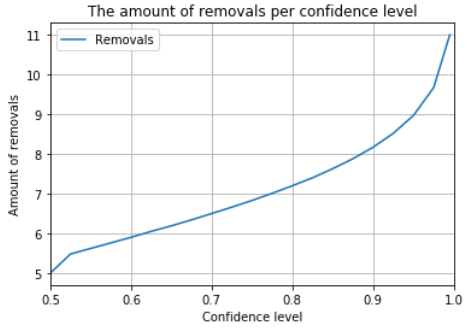


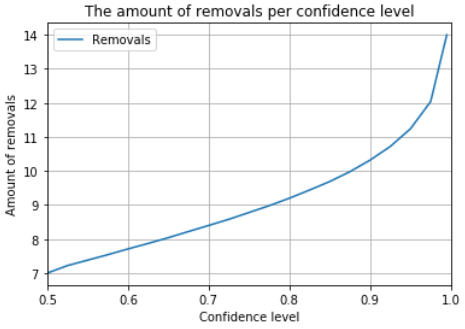
Figure 14: Calculated failure rate functions of the distributions versus the uncensored data.

A.3 Removals against confidence levels Weibull

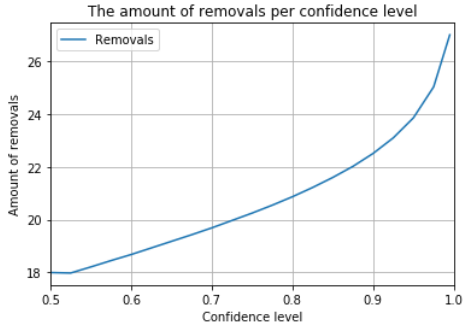
These plots show the amount of removals against the confidence levels for each component type.



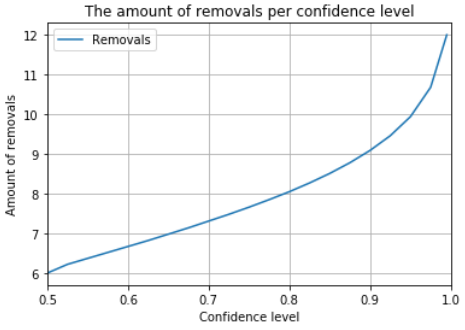
(a) Component type 870047.



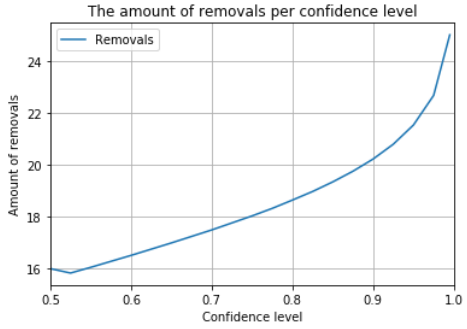
(b) Component type 870052.



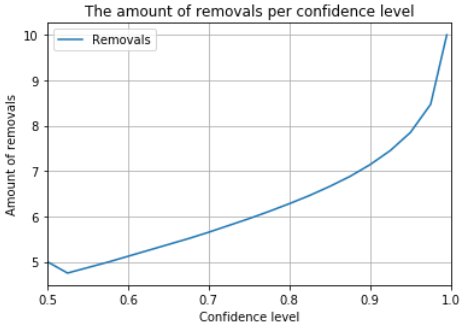
(c) Component type 870180.



(d) Component type 871765.



(e) Component type 888003.

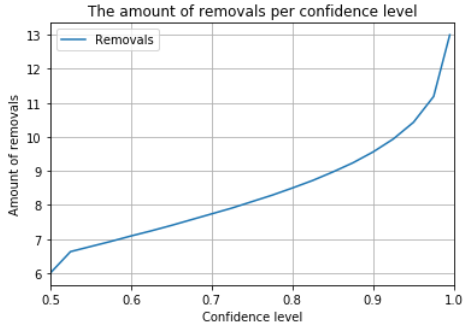


(f) Component type 888004.

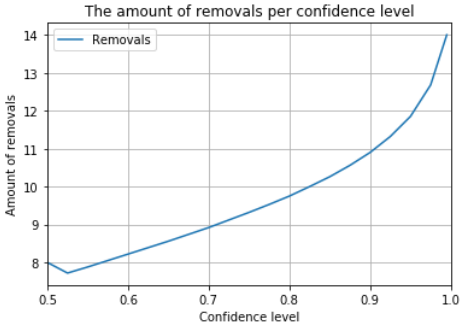
Figure 15: The amount of removals against the confidence level.

A.4 Removals against confidence levels Exponential

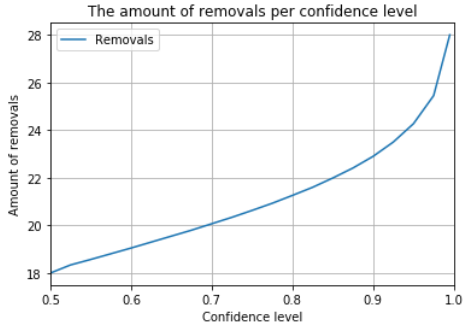
These plots show the amount of removals against the confidence levels for each component type.



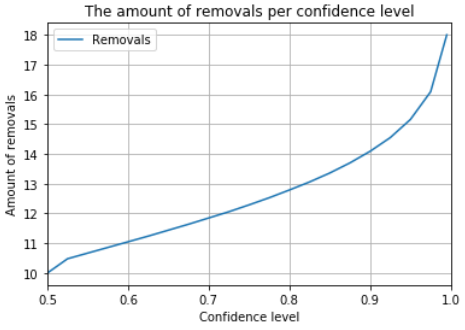
(a) Component type 870047.



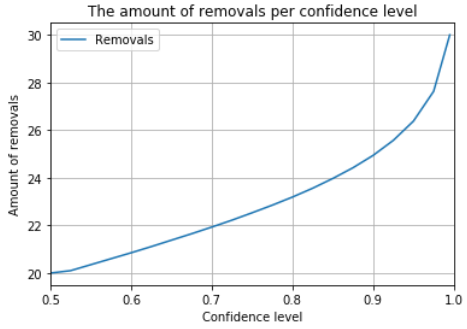
(b) Component type 870052.



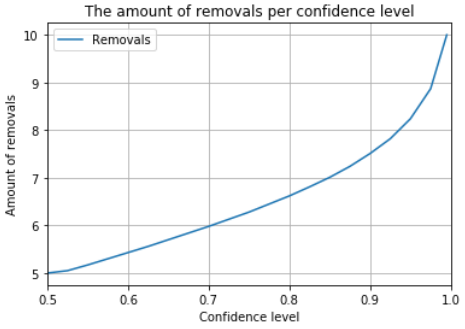
(c) Component type 870180.



(d) Component type 871765.



(e) Component type 888003.

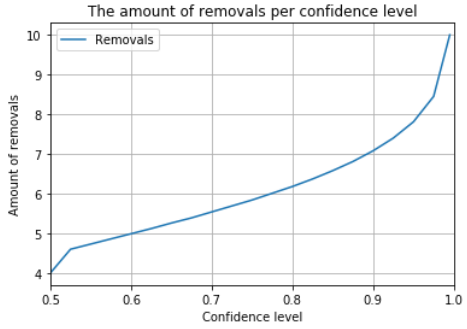


(f) Component type 888004.

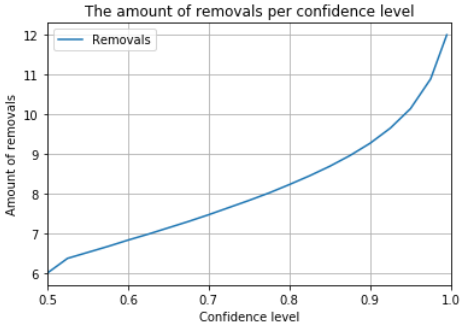
Figure 16: The amount of removals against the confidence level.

A.5 Removals against confidence levels Log-normal

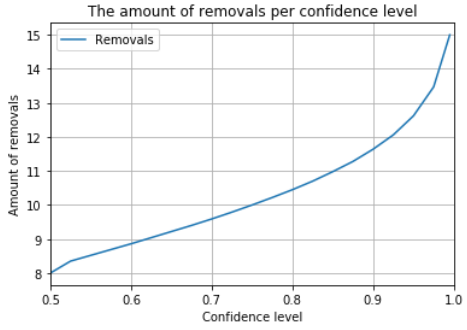
These plots show the amount of removals against the confidence levels for each component type.



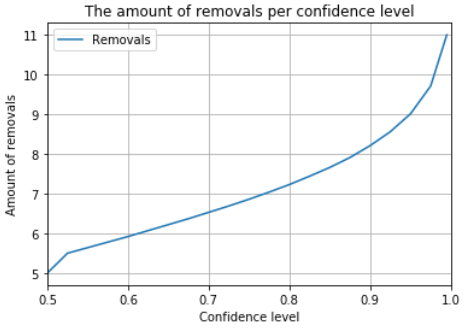
(a) Component type 870047.



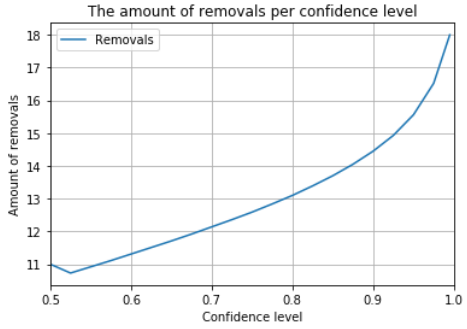
(b) Component type 870052.



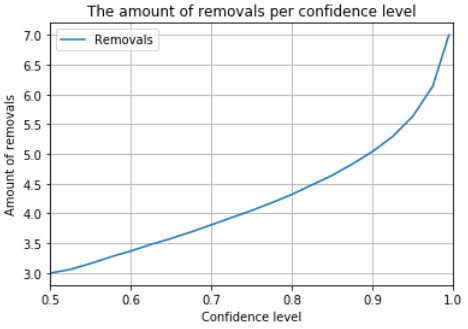
(c) Component type 870180.



(d) Component type 871765.



(e) Component type 888003.



(f) Component type 888004.

Figure 17: The amount of removals against the confidence level.

A.6 Estimator for Exponential distribution

The Python code contains the methods to estimate parameters for the Weibull distributions. This is the Python code to determine the estimator for the Exponential distribution.

```
def estimator_exp(dfDict_censored, dfDict_uncensored, componentType):
'''
The methods return the best parameter that fits to the data according to the Exponential distribution.
'''
estimator[componentType] = (sum(dfDict_censored[componentType]['Hours Flown']) +
sum(dfDict_uncensored[componentType]['Hours Flown'])) / len(dfDict_uncensored[componentType]['Hours Flown'])
```

A.7 Estimators for Log-normal distribution

This is the Python code to determine the estimators for the Log-normal distribution. Please mind that indents should be placed when using the code.

```
def estimator_log(uncensored, censored, iterations_mu, iterations_sigma):
'''
The methods return the best parameters that fit to the data according to the Log-normal distribution.
'''
#Define the space of possible k and lambda
mu = np.linspace(1500, 9400, num = iterations_mu)
sigma = np.linspace(0.000001, 5, num = iterations_sigma)
best_mu = 0
best_sigma = 0
best_estimator = -np.inf


#Loop over all possible k and lambda in order to determine the optimal estimators
for x in mu:
for y in sigma:
estimator = est_log_x(uncensored, x, y) + est_log_y(censored, x, y)
#Check whether the estimator is the best estimator
if estimator > best_estimator:
#Adjust the parameters to the best parameters
best_mu = x
best_sigma = y
best_estimator = estimator
params = [best_sigma, best_mu]
print(params)
return params


def est_log_y(unknown, mu, sigma):
solution = 0
for id in unknown:
solution += np.log( 1- st.lognorm.cdf(id, sigma, loc = 0, scale = mu))
return solution


def est_log_x(known, mu, sigma):
solution = 0
for id in known:
if id != 0:
solution += np.log(st.lognorm.pdf(id, sigma, loc=0, scale = mu))
return solution
```

A.8 Model manual

This is a manual to show how the program should be used. The model itself is an .ipynb file, which is written in the programming language Python. To open and use the model properly, it is recommended to use Jupyter Notebook and have Python version 3.6. The input cell of the model is shown in figure 18.

```
#Enter the dates and files here

#Optional: Enter a date until when a prediction is wanted as 'yyyy-mm-dd', else: enter 'None'. Then the date of today is used.
date_of_prediction = '2017-07-01'
#Enter the name of the removal history csv-file, including .csv, and make sure that it is ';' separated.
removal_history = 'Removal History 787 Pool.csv'
#Enter the date of the day after the last date the removal_history file was updated as 'yyyy-mm-dd'.
date_known = '2017-04-01'
#Enter the name of the general csv-file, including .csv, and make sure that it is ',' separated.
general = 'General Data 787 Pool.csv'
#Enter the name of the hours csv-file, including .csv, and make sure that it is ';' separated.
hours = 'Hours 787 Pool.csv'
#Enter a confidence in the range of [0.5, 1).
conf = 0.95
#Enter the parameter space.
param_space = 100
```

Figure 18: The input cell of the model.

To use the model, some data files and dates are needed:

- *Date of prediction*
  The date of prediction is the date until when a prediction is wanted. This date should be entered in the format 'yyyy-mm-dd' and is optional. When no date of prediction is wanted, 'None' should be entered. In that case the date the model is called will be used, which means the current date.
- *Removal history file*
  Here the name of the removal history 787 pool file should be entered. Make sure that the file is in .csv format and semi-colon separated. The name of the file should be entered as '<name of the file>.csv'.
- *Date known*
  The date known depends on the removal history file, because this is the first date not mentioned in the removal history file. For example, if the removal history file is up-to-date until the 31th of March 2017, the date known should be the 1th of April in that year. The date should be entered in the format 'yyyy-mm-dd'.
- *General data file*
  Here the name of the general data 787 pool file should be entered in .csv format. KLM E&M delivered this file using a comma-separation, therefore this csv-file should always be comma-separated. The name of the file should be entered as '<name of the file>.csv'.

- *Hours file*
  The last file necessary for this model, is the hours 787 pool data file. This file should be entered as a .csv file as well, and should be semi-colon separated. KLM E&M delivered this file as an excel file containing two worksheet tabs. Python does not support excel files and certainly not multiple tabs. Therefore, the tab containing the (pivot)table should be deleted and the other tab should be saved as a semi-colon separated file in csv format. The name of the file should be entered as '<name of the file>.csv'.
- *Confidence level*
  Conf. is the confidence of the confidence interval, meaning that the number of removals in the results have a probability of 'conf' to be equal or lower than this number of removals. Conf. should be entered as a float in the range of [0.5, 1), for example '0.95' if a confidence level of 95% is preferred.
- *Parameter space*
  A parameter space contains all possible combinations of values for all the different parameters, and is an integer value. The higher the number of the parameter space, the longer the duration of the algorithm will be. However, a higher parameter space results in a more accurate answer. Therefore, the parameter space is set at 100 at first. Then accurate results will be obtained, but also a short running time.

These files and dates should be entered at the top of the program. In case it is unclear where to place them, search for the *'Enter the variables and files here'* in the program. To run the program, select 'Cell' in the toolbar and select 'Run All'. The results will appear underneath the cells of the different methods.

N.B. Please note that the used files should be uploaded to the correct Jupyter Notebook folder, that is, the folder used. The format of the data files delivered by KLM E&M are considered as standard. Also, the date of prediction should be later than the date known.

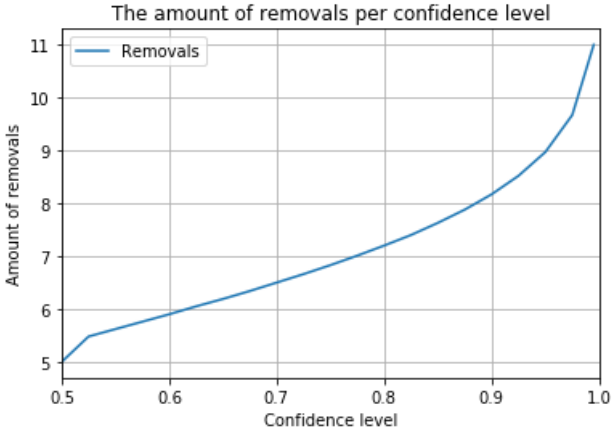
A.9 Model output

The input of the model:
date_of_prediction = '2017-07-01'
removal_history = 'Removal History 787 Pool.csv'
date_known = '2017-04-01'
general = 'General Data 787 Pool.csv'
hours = 'Hours 787 Pool.csv'
conf = 0.95
param_space = 100

The output of the model:

Prediction between 2017-04-01 and 2017-07-01:

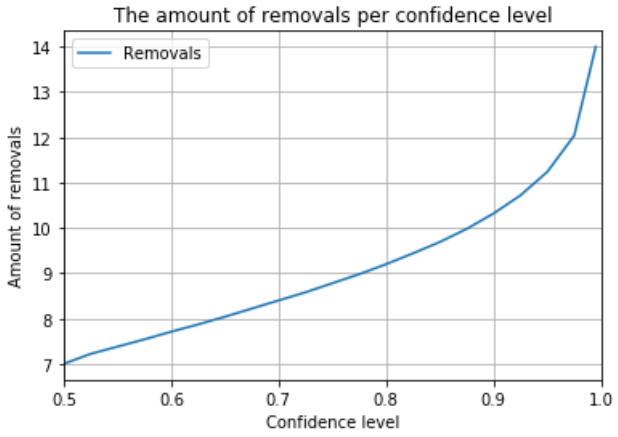
> Component 870047



(a) Component type 870047.

The mean amount of removals = 5.34
With a confidence of 95.0%, the number of removals will be equal or less than 8.97
Rounding the removals down to 8, the confidence will be 88.6 %
Rounding the removals up to 9, the confidence will be 95.1 %

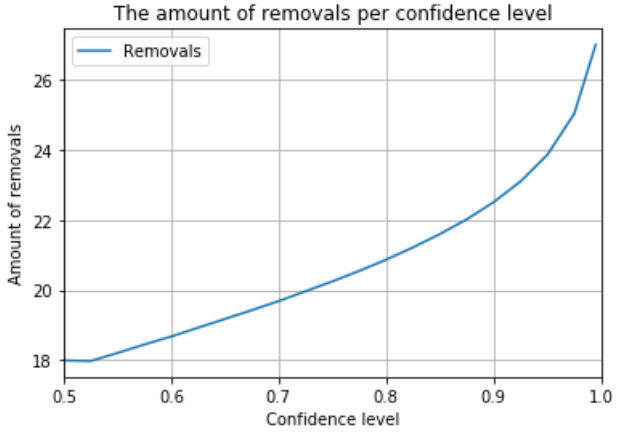
> Component 870052



(b) Component type 870052.

The mean amount of removals = 7.06
With a confidence of 95.0%, the number of removals will be equal or less than 11.24
Rounding the removals down to 11, the confidence will be 94.0 %
Rounding the removals up to 12, the confidence will be 97.4 %

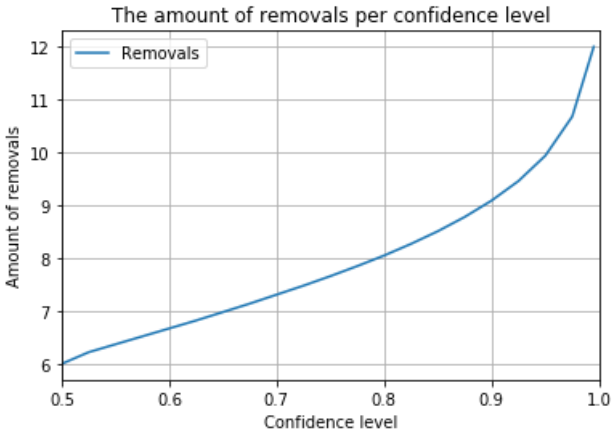
> Component 870180



(c) Component type 870180.

The mean amount of removals = 17.74
With a confidence of 95.0%, the number of removals will be equal or less than 23.86
Rounding the removals down to 23, the confidence will be 92.1 %
Rounding the removals up to 24, the confidence will be 95.4 %

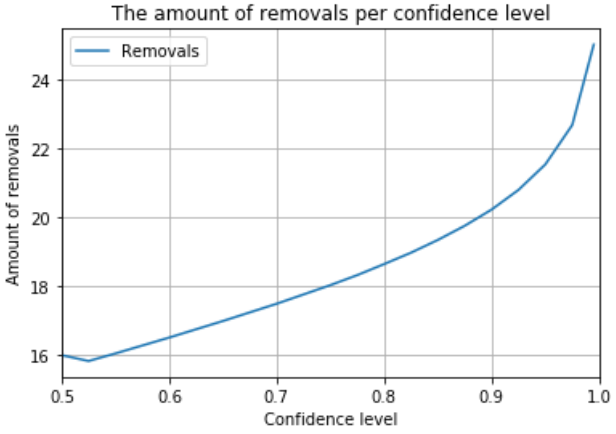
> Component 871765



(d) Component type 871765.

The mean amount of removals = 6.08
With a confidence of 95.0%, the number of removals will be equal or less than 9.94
Rounding the removals down to 9, the confidence will be 89.3 %
Rounding the removals up to 10, the confidence will be 95.3 %

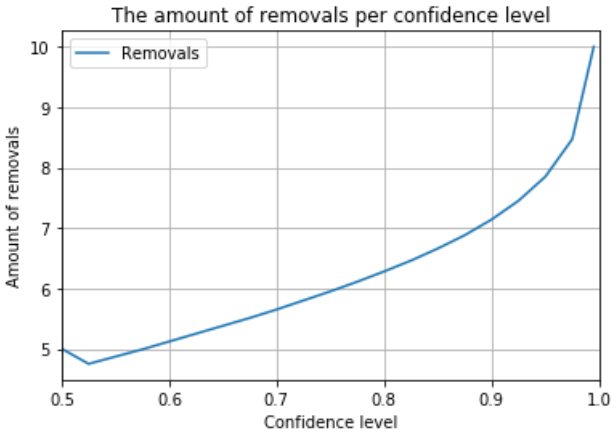
> Component 888003



(e) Component type 888003.

The mean amount of removals = 15.60
With a confidence of 95.0%, the number of removals will be equal or less than 21.53
Rounding the removals down to 21, the confidence will be 93.3 %
Rounding the removals up to 22, the confidence will be 96.2 %

> Component 888004



(f) Component type 888004.

The mean amount of removals = 4.62
With a confidence of 95.0%, the number of removals will be equal or less than 7.85
Rounding the removals down to 7, the confidence will be 88.7 %
Rounding the removals up to 8, the confidence will be 95.7 %

— Run time 80.53 seconds —