INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

AIRBUS | FHS
Flight Hour Services

Institut National des Sciences Appliquées de Toulouse
Département Génie Mathématique et Modélisation

GRADUATION PROJECT REPORT

o0o

Flight Data Analysis

Forecast of aircraft parts failures and optimization of spare parts stock management

Ngo Nghi Truyen Huynh
Ngoc Bao Nguyen

January 2021

Theoretical advisor:
Prof. Jean-Yves Dauxois

Supervisors:
Mr. Erwann Cloteaux
Mr. Mohammed Dergham

ACADEMIC YEAR 2020-2021

Abstract

Nowadays, lifetime analysis plays an important role in multiple aspects of a business. This project, under supervision of Airbus FHS and INSA Toulouse, is an analysis of aircraft component reliability. The main challenge is to predict future failure of aircraft parts and provide useful information for stock management procedure. In this paper, our methodology is to utilize some concepts in reliability analysis to build a predictive parametric model, such as exponential, Weibull, log-normal, as well as non-parametric model, namely Kaplan-Meier estimator. We then develop several simulation algorithms to predict and extract those useful information. At the end, we conclude that there would be three future improvements to be done: deployment of regression models, improvement of out-of-stock date prediction, and accounting for the existence of two or more technologies in a transitional period.

Keywords: Lifetime analysis, reliability, parametric model, non-parametric model, Kaplan-Meier estimator, regression models.

Contents

Introduction

This project deals with “Aircraft components reliability”, which is a key concept in the aeronautical industry. The main purpose of this project is to utilize some statistical models in order to predict future failure of aircraft parts, and then to optimize the management of spare parts stock.

This project is carried out in favour of Airbus Flight Hour Services (FHS), an Airbus department that provides stocking and repairing services to airline companies all around the world to minimize the time and cost of replacement of aircraft failed components. At airlines, inoperative units on airplanes are removed and then are replaced with serviceable units that are provided from their on-site stock. Airbus FHS here plays two important roles: providing serviceable units for the site stock and repairing unserviceable units at repair stations (the OEMs).

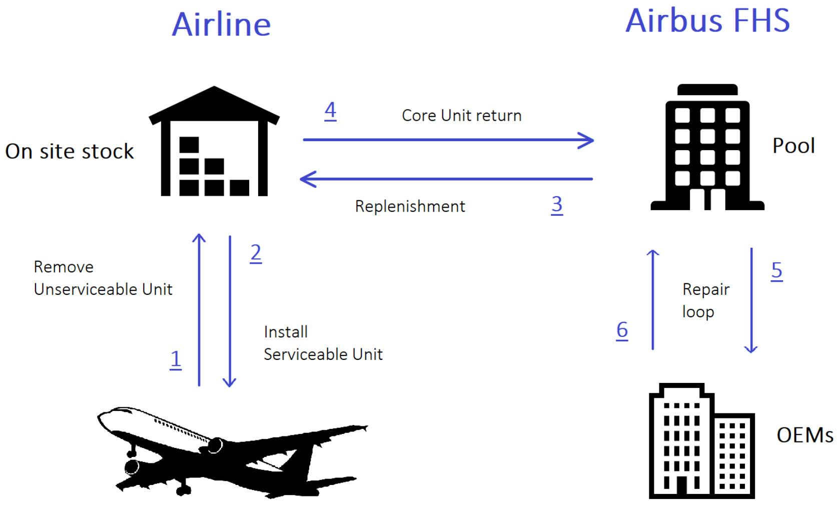

Figure 1: Airbus FHS principal activities

The main objectives of the project is to:

• define component lifetime forecast models and then by using these models,

• determine if the current stock of spare parts is enough to full fill all the needs until the end of the contracts. If not, determine when the stock will be empty and will thus require to be replenished with new or repaired parts.

In the new situation due to the pandemic, the airlines have less aircraft flying and more contracts ending. Consequently, FHS need to fine-tune strategy “repair or stock unserviceable” to optimize the stock management. So the failure forecast of aircraft parts is a significant underlying contributing factor in the proposals of such strategy.

Part I

Data presentation

1 Dataset

The 3 main input data sheets are:

- "Removals": records from 2012 until today of unscheduled removals, with PN, SN, Airline, removal date, TSI & TSN information.
- "SN list": status as of today of parts, with PN, SN, status (on aircraft or stored), Airline (if on aircraft), TSI and TSN information.
- "Airlines": normalized name of airlines, number of aircraft, flight hour per aircraft per month and end of contract of each airline.

Now, let us introduce several technical words in the data set that is used in this report:

- PN (Part number): Identifier of a particular part design of an aircraft component (for example, we have to deal with two data sets, one with the part numbers "A", "B", "C" and "C-new", the other is with "D", "E" and "F").
- SN (Serial number): Unique identifier of a particular physical instantiating of a Part Number across its production.
- TSI (Time Since Installation) is cumulative flight hours by a part between the date of its installation on aircraft and a later date: the date of its removal for the "Removals" sheet, today's date for the "SN list" sheet.
- TSN (Time Since New): cumulative flight hours by a part between the date of its production and a later date: the date of its removal for the "Removals" sheet, today's date for the "SN list" sheet.

Remark. *It is possible that a SN can be present many times in removals list, that means it has been removed and repairs several times. In "Removals" sheet, TSI=TSN means that it's the first removal. In "SN list sheet", for parts on aircraft, TSI=TSN means that the part has never been removed so far, otherwise if TSN>TSI its means that the part has been repaired at least once.*

2 Data pre-processing

The original data sheets are not ready to be used as input data for the `lifelines` python package. These data is indeed need a pre-processing.

Firstly, we want to combine the two data sheets "Removals" and "SN list" to form a population which contains every observed lifetimes (removals) and every healthy individual. Each individual of the combined sheet is identified by a combination of "Serial number", "Time since installation", and "Time since new".

Secondly, in order for `lifelines` to work, we need to create a new variable "failed_or_not" which tell us whether an aircraft part is broken or not yet. Precisely, every individual from "Removals" sheet is considered broken, all the individuals from "Serial Number" sheet which are in service are considered still healthy. All the parts that are not in service nor in "Removals" sheet are already dropped out of the combined sheet due to the unique combination of "Serial number", "Time since installation", and "Time since new", these are just repaired or waiting-to-be-repaired parts and do not contribute any information to the lifetime analysis.

Finally, some correction to blank data cells or duplicated data is carried out as final touch.

Remark. *This pre-processing procedure is coded based on the name space of original data file. In order for this to work with new data file, the new data should be named as presented in Appendix B.*

Part II

Lifetime models

In this project, we consider TSI as the observed lifetime and "failed_or_not" [1] as censoring indicator, denote (T, δ) (see more in 3.2).

3 Basic concepts and notions of reliability

We first introduce several key concepts and notions used in this project, which is related to reliability and survival analysis.

3.1 Reliability and related functions

Definition 3.1. *The survival function (or reliability function) is a function that gives the probability that a patient, device, or other object of interest will survive beyond any specified time. Denote:*

$$\forall t \geqslant 0, R(t) = \mathbb{P}(T > t)$$

where T is a random variable considered as a lifetime which is positive $T > 0$.

Now we denote $F(.)$ is the cumulative distribution function (c.d.f.) of a random variable T ($T > 0$), so the survival function of T: $R(.) = 1 - F(.)$. Both functions characterize the distribution of T.

Considering an object which has survived at age x, we are interested in calculating the probability that it will survive in next y years. Let us define the conditional reliability.

Definition 3.2. *Let X be a random variable $X > 0$, $x \geqslant 0$. So, the conditional reliability at age x of X $R(.|x)$ (or denoted $R_x(.)$ given by:*

$$\forall y > 0, R(y|x) = P(X - x > y|X > x).$$

In addition, we can demonstrate that: $R(y|x) = \dfrac{R(x+y)}{R(x)}$.

The hazard rate at point x represents the instantaneous probability of failure (or death) at time x given that failure (or death) did not occur before.

Definition 3.3. *If X is a continuous random variable, the hazard rate function $h(.)$ is defined by:*

$$\forall x \geqslant 0, h(x) = \lim_{\Delta t \longrightarrow 0^+} \frac{1}{\Delta t} \mathbb{P}(X \in [x, x + \Delta t)|X \geqslant x).$$

And we can also show that:

$$\forall x \geqslant 0, h(x) = \frac{f(x)}{R(x)}$$

where $f(.)$ is the probability density function (p.d.f.).

[1] A new variable created by data prepossessing (see in Section 2).

Definition 3.4. *If X is a continuous random variable, the cumulative hazard rate function $\Lambda(.)$ is defined by:*

$$\forall x \in \mathbb{R}, \Lambda(x) = \int_0^x h(s)\, ds.$$

3.2 Failure and censoring

In survival data analysis field, we often have to study in the condition in which the value of a observation is only partially known, we call this censoring. In this project, we are in the case of right censoring data in which a data point is above a certain value but it is unknown by how much. Concretely, we have some aircraft parts that have not yet failed and that do not correspond to failures.

Definition 3.5. *The lifetime X is said to be right censored by a random variable C if, instead of observing X, one observe $min(X, C)$.*

In our case study, the censoring time C is known and fixed (the same for all items). Concretely, we take $C = c$ the end of study (here is December 2020). One only observes the random variable $(T_i, \delta_i)_{i=1,\dots,n}$ defined by:

- $T_i = min(X_i, c)$,
- $\delta_i = \mathbb{1}_{\{X_i \leqslant c\}}$.

4 Usual parametric models

This section presents all parametric models implemented in our codes for modeling the reliability function.

4.1 Exponential

Exponential distribution is one of the best known models for approximating reliability function of real life products. It has several ideal properties that will facilitate the calculation on paper. However, the chance that the reliability function of a particular product matches the modelisation of this distribution is quite rare since exponential distribution has a very specific shape in order for it to have those special properties.

An exponential distribution $E(\lambda)$ is characterized by its parameter λ. The function that describes the distribution are defined by:

Survival function:

$$R(t) = exp\left(\frac{-t}{\lambda}\right).$$

Cumulative hazard rate:

$$\Lambda(t) = \frac{t}{\lambda}.$$

Hazard rate:

$$h(t) = \frac{1}{\lambda}.$$

In exponential models, parameter λ can be interpreted as inverse of mean residual lifetime.

4.2 Weibull

Weibull distribution is also a well known model in reliability study. This model has a more flexible shape and form in comparison to exponential distribution while still keeping some of the special properties which simplify calculation on paper.

A Weibull distribution $W(\lambda, \rho)$ is characterized by its two parameters λ and ρ. The reliability function is then defined by:

$$R(t) = exp\left(-\left(\frac{t}{\lambda}\right)^{\rho}\right), \lambda > 0, \rho > 0.$$

Cumulative hazard rate:

$$\Lambda(t) = \left(\frac{t}{\lambda}\right)^{\rho}.$$

Hazard rate:

$$h(t) = \frac{\rho}{\lambda}\left(\frac{t}{\lambda}\right)^{\rho-1}.$$

The λ is called scale parameter, which has an usable interpretation: it represents the time when 63.2% of the population has died. The ρ is called shape parameter, which controls whether the shape of cumulative hazard rate is convex or concave, representing accelerating or decelerating hazards.

4.3 Log-Logistic

Log-Logistic is also a usual distribution. Its advantages and disadvantages are similar to Weibull distribution.

A log-logistic $\mathcal{LL}(\alpha, \beta)$ is characterized by its two parameters α and β. The reliability function is defined by:

$$R(t) = \left(1 + \left(\frac{t}{\alpha}\right)^{\beta}\right)^{-1}.$$

Hazard rate:

$$h(t) = \frac{\left(\frac{\beta}{\alpha}\right)\left(\frac{t}{\alpha}\right)^{\beta-1}}{1 + \left(\frac{t}{\alpha}\right)^{\beta}}.$$

Cumulative hazard rate:

$$\Lambda(t) = log\left(\left(\frac{t}{\alpha}\right)^{\beta} + 1\right).$$

In this distribution, the α parameter is interpreted as median lifetime. The β parameter, on the other hand, alter the shape of hazard rate function.

4.4 Log-normal

Log-normal distribution is a modified version of standard normal distribution. In real life, most random variable follow a standard normal distribution with a mean μ and a standard deviation σ, however, random variable lifetime T can rarely match a standard normal distribution. This is where the log-normal model come to be useful.

A log-normal distribution $\mathcal{LN}(\mu, \sigma)$ is characterized by its two parameters μ and σ. The reliability function is defined by:

$$R(t) = 1 - \Phi\left(\frac{log(t) - \mu}{\sigma}\right)$$

where σ, and Φ is the c.d.f. of normal random variable.

Cumulative hazard rate:

$$\Lambda(t) = -log\left(1 - \Phi\left(\frac{log(t) - \mu}{\sigma}\right)\right)$$

μ and σ are interpreted respectively as mean and standard deviation of $log(T)$.

4.5 Piecewise Exponential model

Piecewise exponential model is multiple exponential models defined on each interval of timeline. In `lifelines` Python package, these intervals are predefined by users or defined automatically by `lifelines.uti-ls.find_best_parametric_model` function.

This model could give a lot of flexibility to the estimated reliability function, which helps finding the best model for unusual shape of survival function. Nevertheless, by using this model, we sacrifice the explicit expression of survival function $R(t)$ and cumulative hazard rate $\Lambda(t)$.

It hazard rate function is defined by:

$$h(t) = \begin{cases} 1/\lambda_0 \text{ if } t \leq \tau_0 \\ 1/\lambda_1 \text{ if } \tau_0 < t \leq \tau_1 \\ 1/\lambda_2 \text{ if } \tau_1 < t \leq \tau_2 \\ ... \end{cases}$$

where λ_i is the best parameter for exponential model in each interval.

4.6 Natural cubic splines

Natural cubic splines method is one of the best approximation method in finding a continuous function which passes through a set of points in a Descartes 2 dimensional space. The idea is to approximate a cubic function (a polynomial function of degree 3) between in the interval between each two consecutive points using a predefined underlying relation between f, f', f", and f'''. To keep this report short and concentrated on the reliability analysis, we only include this principle but do not include the exact definition and calculation of this method.

Let v_j are our cubic basis functions which approximate the cumulative hazard rate in function of $log(t)$ at predetermined knots:

$$R(t) = exp\left(\phi_0 + \phi_1 \log t + \sum_{j=2}^{N} \phi_j v_j(\log t)\right).$$

This model offer a great advantage of flexibility and smoothness, however, the explicit expression of some the p.d.f. and the hazard rate remains unknown.

Remark. *The parametric models are not able to estimate the reliability function with high precision comparing with the observed population in several cases (we can verify whether a parametric model can fit the data in Section 6) but it allows us to estimate the survival probability for any individual in or beyond the population.*

5 Non-parametric models

Here we do not assume any parametric model for the lifetime. The aim is to estimate its distribution (or other functions of interest) using only the data and without any assumption.

Let us suppose that there is *no ex-aequo* in the sample, as it is the case for the Competing Risks data set. Let $t_{(1)} \leqslant ... \leqslant t_{(n)}$ be the n observed and ordered times and $\delta_{(1)}, ..., \delta_{(n)}$ their corresponding indicators. One can estimate the hazard rate function $\lambda(.)$ by:

$$\hat{\lambda}(x) = \begin{cases} \dfrac{\delta_i}{n-i+1}, \text{ when } x = t_{(i)} \text{ for } i = 1, ..., n \\ 0 \text{ , elsewhere.} \end{cases}$$

Now we are interested in giving an estimator for reliability function. The Kaplan–Meier estimator is a non-parametric estimator which is used to estimate the reliability function from censored data. This method appeared on a paper given by Kaplan and Meier [3] in 1958 and it has become one of the key statistical methods for analyzing censored survival data in recent decades. The Kaplan-Meier estimator of reliability function $R(.)$ given by:

$$\hat{R_n}(x) = \prod_{i:T_{(i)} \leqslant x} (1 - \hat{\lambda}(T_{(i)})) = \prod_{i:T_{(i)} \leqslant x} (1 - \frac{\delta_{(i)}}{n-i+1}).$$

Figure 2 shows the Kaplan-Meier estimator for each part number A, B, C and C-new.

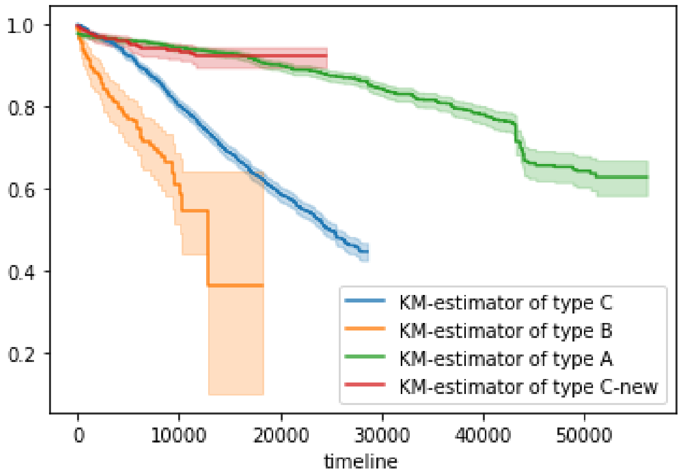

Figure 2: Kaplan estimator of PN "A", "B", "C" and "C-new".

The `lifelines` [2] Python library allows us to find a best parametric model of the population by comparing the score based on AIC, BIC, etc. criterion to Kaplan-Meier model. We can also compare the best parametric model for part number B (Figure 3) and part number A (Figure 4) to the Kaplan-Meier estimators.

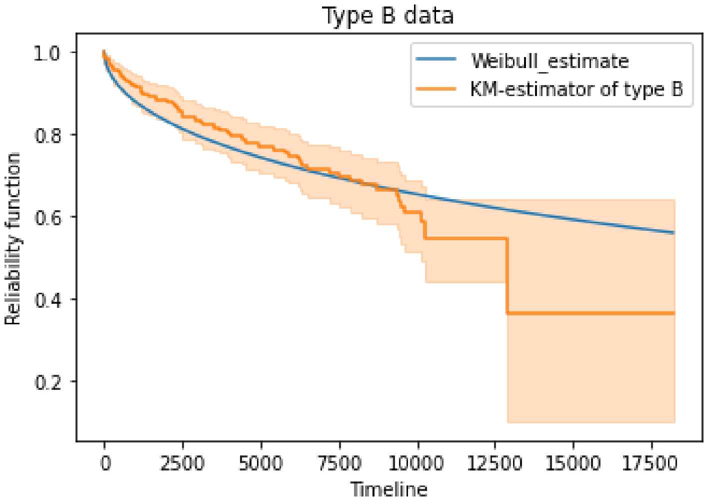

Figure 3: Comparing best parametric model and Kaplan-Meier estimator PN "B".

The best parametric model of PN "B" is Weibull estimator.

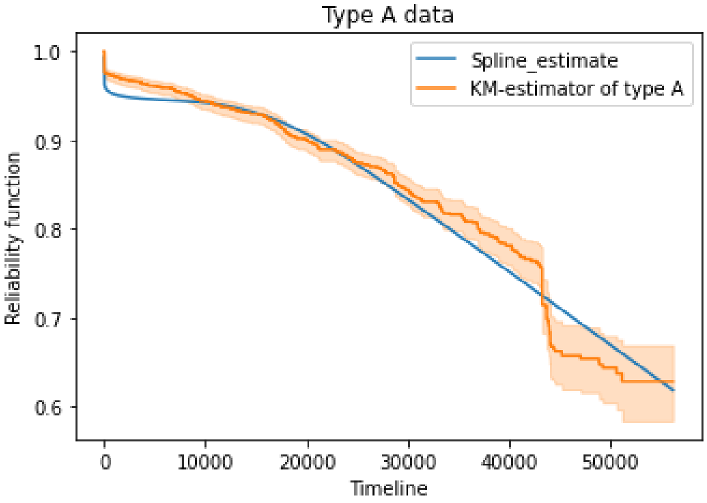

Figure 4: Comparing best parametric model and Kaplan-Meier estimator of PN "A".

The best parametric model of PN "A" is Spline estimator.

Remark. *Kaplan-Meier model allows to estimate the reliability function with high precision comparing with the observed population, especially, for a large number of population size but it is not able to estimate the survival probability of an individual beyond the population.*

6 Homogeneity test

This section focus on comparing the distributions of k different samples, with respective reliability functions $R_1, ..., R_k$. The questions are:

- Is the distribution the same among all the different sub-populations?
- Do the different types of material have the same behavior along time?

In respond to these questions, we could use the homogeneity test. The null hypothesis says that the distribution of the categorical variable is the same for each subgroup or population.

Suppose that we want to compare the distributions of K different samples, with respective c.d.f. $F_1, ..., F_K$. Thus, the test hypothesis given by:

$$\mathcal{H}_0 : F_1 = F_2 = \; = F_K \text{ against } \mathcal{H}_1 : \text{at least one of the distributions differs from the others.}$$

Let $t_1 < t_2 < \; < t_D$ be the distinct observed times in the pooled sample. For $j = 1, ..., K$ and $i = 1, ..., D$, we denote:

- d_{ij} the number of failures (i.e. not censored) observed at time t_i in population j,
- y_{ij} the number of individuals at risk just before time t_i in population j,
- $d_i = \sum_{j=1}^{K} d_{ij}$ the number of failures (i.e. not censored) observed at time t_i in the pooled sample,
- $y_i = \sum_{j=1}^{K} y_{ij}$ the number of individuals at risk just before time t_i in the pooled sample.

Now let consider the weight functions of the form:

$$W_j(t_i) = y_{ij} W(t_i)$$

where $W(.)$ is a common weight function of choice [2]. Now we write:

$$Z_j = \sum_{i=1}^{D} W(t_i)(d_{ij} - y_{ij}\frac{d_i}{y_i}), \forall j = 1, ..., K$$

and note that, under $\mathcal{H}_0$, Z_j are asymptotically close to 0. Further, since $\sum_{j=1}^{K} Z_j = 0$, we can consider the test statistic:

$$S_K = (Z_1, ..., Z_{K-1})\Sigma^{-1}(Z_1, ..., Z_{K-1})'$$

where Σ is the covariance matrix of the random vector $(Z_1, ..., Z_{K-1})$. We can show that this statistical test is asymptotically close to a chi-square distribution with $K-1$ degrees of freedom $\chi^2(K-1)$. Finally, the test reject $\mathcal{H}_0$ for a large values of S_K.

For example, in our code, we try to apply the homogeneity test for the distribution of the old part and new part of PN "A". The output is:

- The test statistic $S_K = 419.61$,

[2] Many choices are possible for the weight function. For example, with $W(.) = 1$, we obtain the Log-rank test, which is used in our code.

- The p value $p \approx 2.97e - 93 << 0.05$.

So we reject $\mathcal{H}_0$. In other words, we can not assume that the distribution of PN "A" is the same for each subgroup (here is old part and new part). Figure 5 allows us to comparing the reliability function of the old part to the new part of PN "A".

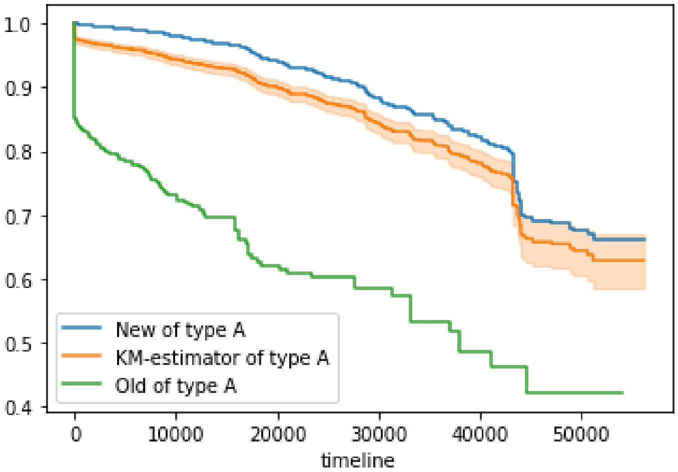

Figure 5: Reliability functions of old part comparing to new part of PN "A".

Part III

Simulation and stock optimization

In this part, we want to simulate the failure number at a determined time in the future and then predict a date at which products are out of stock for immediately replacement (this day is called out-of-stock date). These simulations are based on the law of each part number that we have considered in the previous section. The main idea is to utilize "inverse transform sampling" method in order to simulate the lifetime of each serial number of a specific part number knowing its statistical model as well as its TSI.

7 Inverse transform sampling

We denote T the lifetime of aircraft part, F_T its cumulative distribution function (c.d.f.), and $R_T = 1 - F_T$ its reliability function. We consider the following theorem:

Theorem 7.1. *Let T be a random variable with c.d.f. F_T and reliability function R_T. Thus, the random variables T, $F_T^{-1}(U)$ and $R_T^{-1}(U)$ have a same distribution, where U is a random variable of standard uniform distribution $U \sim U([0,1])$.*

For instance, we are able to simulate the lifetime T by computing the inverse of survival function at a random variable of standard uniform distribution $R_T^{-1}(U)$. In order to compute the inverse of reliability function R_T^{-1} (which could be parametric or non-parametric models), we could use the *piece-wise linear approximation* method given in [1].

Since the lifetime model of a part number of aircraft part is characterized by its survival function $R(.)$, in our code, we consider a lifetime model with two characterizations: the timeline $t = (t_1, t_2, ..., t_n)$ (with respect to $0 < t_1 < t_2 < ... < t_n$) and $p = (R(t_1), R(t_2), ..., R(t_n))$ (with respect to $p_1 \geqslant p_2 \geqslant ... \geqslant p_n > 0$) a set of corresponding survival probability at each point of timeline. Now we want to compute R^{-1} using the piece-wise linear approximation. The main idea of this algorithm is by assuming that R is a linear function on each interval $[t_i, t_{i+1}]$ for $i = 1, 2, ..., n-1$, we have:

$$\forall i = 1, 2, ..., n, \forall x \in [t_i, t_{i+1}], R(x) = r_i(x)$$

with r_i is a linear function, therefore r_i^{-1} can be easily obtained. As a matter of fact, for all value of u such that $p_n \leqslant u \leqslant p_1$, the inverse of R at u given by:

$$R^{-1}(u) = r_i^{-1}(u) \text{ such that } p_{i+1} \leqslant u \leqslant p_i.$$

Hence, the lifetime T is simulated by computing the inverse of reliability function at a random variable of standard uniform distribution $R_T^{-1}(U)$, where $U \sim U([0,1])$.

Remark. *There are two important notes:*

- *For all parts that failed as soon as the installation (TSI=0), we consider its TSI is equal to 10^{-6} (hour) to avoid division by zero in predefined codes.*

- *As we can see, this algorithm allow the inversion of survival function at any point in $[p_n, p_1]$ but not beyond this interval. Since the inverse function at any point on $[0,1]$ ($R_T^{-1}(U)$ with $U \sim U([0,1])$) is needed, we can only say that the simulated sample $T^{(1)}$ of the lifetime T is greater t_n (or smaller t_1) if $U < p_n$ (or $U > p_1$).*

8 Simulation process

This report is linked with our project available on github, which allows to simulate the failure number of aircraft parts, evaluate this failure number as a function of time and forecast a out-of-stock date. A tutorial of this github project is presented in Appendix A.

8.1 Predicting number of failures

The main idea of our algorithm is: for each individual SN on SN list (which is in working on aircraft), we count the number of time it will fail till the end of contract (or a determined time for which we want to forecast in the future) by simulating the lifetime of this part till that time (if it fail before the end time, we replace that part at the failing time then repeat the simulation). After that, for large number of SN, we repeat that simulation for each individual in order to calculate the failure number in total.

In the "Airlines" data sheet, we find the average flight time of each company which gives a relation between the lifetime and the real time. For example, a part of PN "B" from company "3" has a TSI of $18,000$ (hour) and the average fight time of this company is 350 hours per month, that means we can assume that this part has survived for $18,000/350 = 51.43$ months in real time. Therefore, the fact that we want to predict whether it will fail in 2 years comes to the fact that we compute the probability that this part can survive $18,000 + 24 \times 350 = 26,400$ hours conditioning it has survived for $18,000$ hours $\mathbb{P}(T \geqslant 26400 | T \geqslant 18000)$.

As we remarked, the piece-wise linear approximation method does not allow to return a sample beyond the interval $[t_1, t_n]$. For instance, to estimate how many times a part will fail in next $20,000$ hours while the maximum observed TSI is $18,500$, so we are not able to say if a simulated sample $T^{(j)} > 18500$ will fail in next $1,500$ hours or not with Kaplan-Meier model (the timeline is limited in $[t_1, t_n]$). Fortunately this problem can be solved with parametric model, which give the survival probability at any point of time.

Now we consider a rate r in Airbus FHS repairing policy, which is introduced as the probability of replacing a failed product with a repaired product instead of new one. We can see that if new and repaired parts do not follow the same distribution (can be tested by homogeneity test in 6), so replacement by a new part or an old part has a significant impact on the result. Thus, we consider two algorithms, one in the case that we don't have the difference between new parts and old parts (algorithm 1), the other is in the case of having difference distributions (algorithm 2). More precisely, the aircraft parts that have a TSN greater than its TSI are considered as old parts, otherwise, the rest which has a same value of TSI and TSN are considered as new parts.

```
Algorithm 1: Forecast of failure number (without difference distribution)
Input:
$s_1, ..., s_k$ is a list of TSIs of aircraft parts which are in service ($k$ is the number of parts of considered
  type installed on considered company);
$m$ is an integer number;
$FHperMonth$ is average flight hour per month of considering company;
Output:
$N$ is the number of failure we want to forecast in next $m$ months;
Algorithm:
$FH := m \times FHperMonth$;
if $FH \leqslant t_n$ then
    Considering $R(.)$ as Kaplan-Meier model;
else
    Considering $R(.)$ as best parametric model;
end
$N := 0$;
$i := 0$;
while individual $i < k$ do
    $i := i + 1$;
    The lifetime $S^{(i)}$ is simulated by the conditional reliability $R(.|s_i)$ (or denoted $R_{s_i}(.)$):
      $S^{(i)} := R_{s_i}^{-1}(U)$, where $U \sim U([0,1])$;
    $CumTime := S^{(i)}$;
    while $CumTime < FH$ do
        $N := N + 1$;
        $CumTime := CumTime + R_0^{-1}(U)(= CumTime + R^{-1}(U))$, where $U \sim U([0,1])$;
    end
end
```

Once the distribution of old parts and new parts is tested different, we denote $R\_old(.)$ is the reliability function of the old parts with a set of time line $(t\_old_1, ..., t\_old_n)$ and $R\_new(.)$ of the new part with time line $(t\_new_1, ..., t\_new_n)$. We remind that r is a percentage to which failed products are replaced by repaired products instead of replaced by new. In other words, a broken part can be replaced by a new product with probability $1-r$ while it can be replaced by an old part with probability r.

Algorithm 2: Forecast of failure number (with difference distribution)

```
Input:
$s_1, ..., s_k$ and $w_1, ..., w_k$ are the sets of TSI and TSN of aircraft parts which are in service ($k$ is the
  number of parts of considered type installed on considered company);
$m$ is an integer number;
$FHperMonth$ is average flight hours per month of the considered company;
Output:
$N$ is the number of failures we want to forecast in next $m$ months;
Algorithm:
$FH := m \times FHperMonth$;
if $FH \leqslant t\_old_n$ then
    Considering $R\_old(.)$ as Kaplan-Meier model;
else
    Considering $R\_old(.)$ as best parametric model;
end
if $FH \leqslant t\_new_n$ then
    Considering $R\_new(.)$ as Kaplan-Meier model;
else
    Considering $R\_new(.)$ as best parametric model;
end
$N := 0$;
$i := 0$;
while individual $i < k$ do
    $i := i + 1$;
    if $s_i < w_i$ then
        $S^{(i)} := R\_old_{s_i}^{-1}(U)$, where $U \sim U([0,1])$;
    else
        $S^{(i)} := R\_new_{s_i}^{-1}(U)$;
    end
    $CumTime := S^{(i)}$;
    while $CumTime < FH$ do
        $N := N + 1$;
        $CumTime := CumTime + R\_old^{-1}(U)$, where $U \sim U([0,1])$ with probability $r$;
        or:
        $CumTime := CumTime + R\_new^{-1}(U)$ with probability $1-r$;
    end
end
```

8.2 Predicting out-of-stock date

For a given PN and a given number of spare parts of that PN available in stocks at time t, Airbus FHS want to predict whether this stock will be sufficient to meet all future demands (i.e. the replacement of a failed part by a spare part) until the end of contracts. Otherwise, they hope to estimate a date from which the stock will become insufficient, and then when and how many parts must then be sent for reparation (taking in to account the reparation and transport time, which is called Turn Around Time or TAT). More precisely, knowing that Airbus FHS are aiming for a service level of 90% (out of 100 requests, they can respond favorably (i.e. one part is immediately available in stock) to 90), if they have x spare parts, they first want to estimate a date that they are no longer able to maintain this service level.

We first have to turn a date into a numerical feature. In other words, we consider a date as a number (by month) $date = year \times 12 + month + day/30$ instead of a vector $date = (year, month, day)$. The idea base on both algorithms 1 and 2 but this time, we are interested in a set of date that the failure could be observed instead of the number of failure.

Algorithm 3: Forecast of out-of-stock date

```
Input:
$comp_1, ..., comp_m$ is a list of company;
$end_1, ..., end_m$ is a corresponding end of contract (by month);
$x$ is the number of spare parts;
$los$ is the level of service ;
Output:
$oos$ is estimated out-of-stock date;
Algorithm:
$P = int(x/los)$ ($P$ is an integer part of $x/los$);
$j = 0$;
while $j < m$ do
    $j = j + 1$;
    We simulate the moments that the failures of $comp_j$ occur before $end_j$: $T_j = (T_{1j}, ..., T_{k_j j})$;
end
$T = (T_1, ..., T_m)$ (note that each $T_i$ is a vector with the size of $k_i$);
We sort $T$ in ascending order: $T = (T^{(1)}, ..., T^{(k)})$, where $k = \sum_{i=1}^{m} k_i$;
if $P \leqslant k$ then
    $oos = T^{(P)}$: the $P^{th}$ element of $T$;
else
    $oos = T^{(k)}$: the last element of $T$;
end
```

9 Confidence intervals

Now we want to run this simulation process for a large number of times. For example, Figure 6 presents the distribution of failure number of part number B from December 2020 to June 2025 of company 3 for 1000 Monte-Carlo iterations.

Let X be a random sample (here is the number of failure or the out-of-stock date). We remind that, an interval $CI_{1-\alpha}(X)$ is called the confidence interval of X at confidence level $1 - \alpha$ if we have:

$$\mathbb{P}(X \in IC_{1-\alpha}(X)) = 1 - \alpha.$$

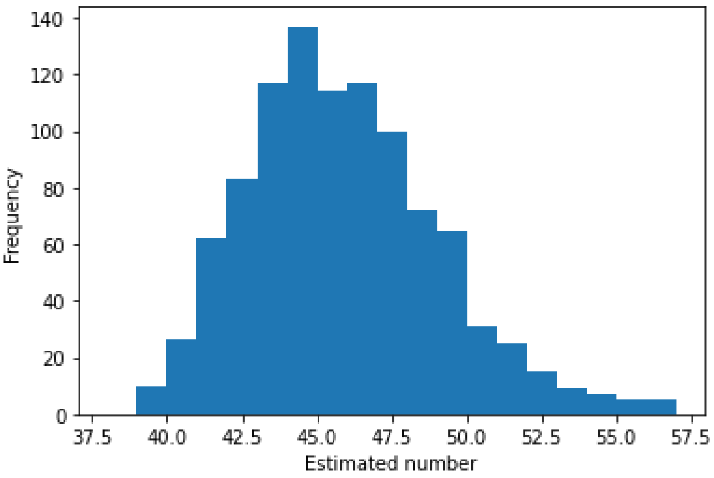

Figure 6: Histogram of estimated failure number of PN "B" in 1000 simulations.

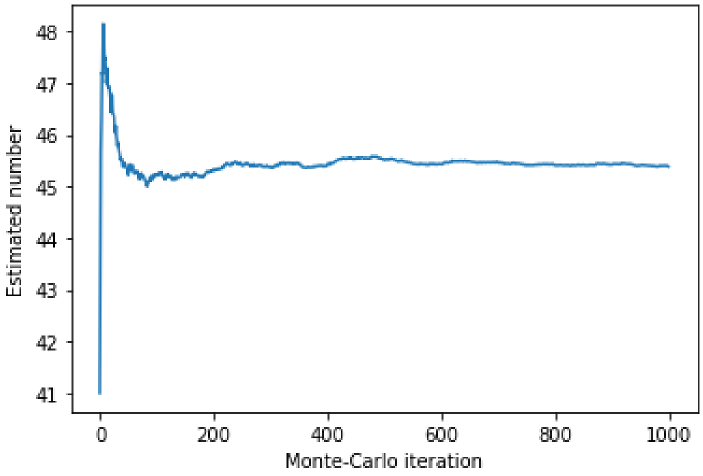

Figure 7: Estimated failure number of PN "B" as a function of number of iterations.

Thus, we want to give the confidence intervals for X and for the mean of X by two ways below.

9.1 Central Limit Theorem

This way focus on the confidence interval of the sample mean $\bar{X}$. We remind the central limit theorem (CLT):

Theorem 9.1. *Let $X_1, X_2, ..., X_n$ be random samples each of size n taken from a population with overall mean μ and finite variance σ^2. Denote $\bar{X}$ is the sample mean. Then, we have:*

$$Z = \sqrt{n}\frac{\bar{X} - \mu}{\sigma} \overset{n \to +\infty}{\sim} \mathcal{N}(0, 1).$$

So we want to find a confidence interval for $\bar{X}$ at confidence level $1 - \alpha$. Since:

$$\sqrt{n}\frac{\bar{X} - \mu}{\sigma} \overset{n \to +\infty}{\sim} \mathcal{N}(0, 1),$$

we have:

$$\mathbb{P}(\sqrt{n}\frac{|\bar{X}-\mu|}{\sigma} \leqslant q_{1-\alpha/2}) = 1-\alpha$$

with $q_{1-\alpha/2}$ is $1-\alpha/2$ quantile of standard normal distribution. Then:

$$\mathbb{P}(\mu - q_{1-\alpha/2}\frac{\sigma}{\sqrt{n}} \leqslant \bar{X} \leqslant \mu + q_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}) = 1-\alpha,$$

and finally:

$$CI_{1-\alpha}(\bar{X}) = \mu \pm q_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}.$$

9.2 Empirical Confidence Interval

This way focus on the empirical confidence interval of the sample X. The idea is to find an interval $CI_{1-\alpha}$ such that:

$$\mathbb{P}(X \in CI_{1-\alpha}) = 1-\alpha.$$

For example, the $\alpha/2$ quantile and $1-\alpha/2$ of X can be chosen in order to define this interval, knowing that the function `quantile` of numpy Python library allows the estimation of quantiles from a population.

Taking an example of out-of-stock date simulation process, we denote T a random variable corresponding to the estimated out-of-stock date for PN "A" with 80 spare parts. By running the simulation process over 1000 Monte-Carlo iterations, we obtain the empirical distribution of T given by Figure 8.

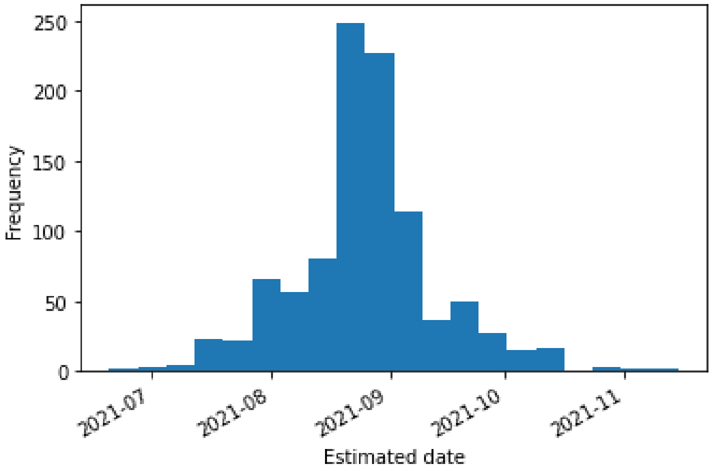

Figure 8: Histogram of estimated out-of-stock date of PN "A" in case of number-of-spare-parts=80.

In this case, we can observe the whole distribution, so the confidence interval is given by:

$$CI_{1-\alpha}(T) = [t_{\alpha/2}, t_{1-\alpha/2}]$$

where $t_{\alpha/2}$ and $t_{1-\alpha/2}$ are $\alpha/2$ and $1-\alpha/2$ quantile of T.

Now, we consider another example where the simulation is carried out on 650 parts, a considerable number of simulated T are censored by the end of contract, this censoring is due to that fact that our algorithm requires an end time to shorten execution time and avoid infinite loop while keeping most of useful information in the simulation till the end of contract. In this case, our simulation predicts that there is a

49.6% chance that the stock is not sufficient, and accordingly 50.4% chance that we will be able to maintain the 90% service level till the end of the last contract. So the confidence interval in this case is given by:

$$CI_{1-\alpha}(T) = [t_\alpha, +\infty)$$

where t_α is α quantile of T.

Part IV

Results and validation

In this section, we try to apply our algorithms on the data in December 2018 and verify if the simulation prediction is coherent with real data between December 2018 and December 2020. Precisely, we estimate the failure number of part number C from December 2018 to December 2020. The results are given in Figure 9 and Table 1.

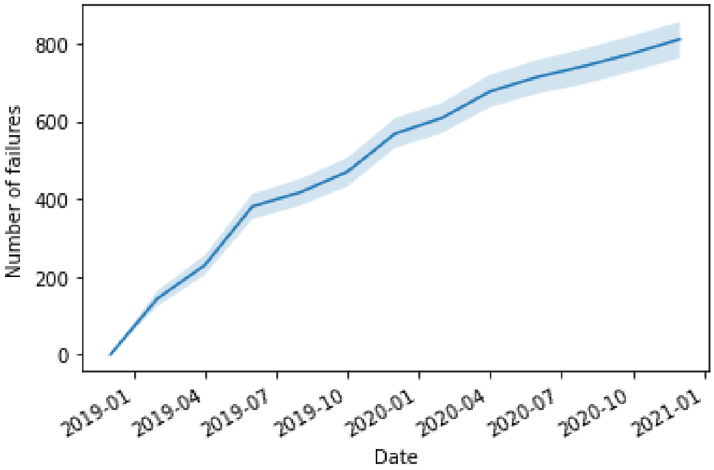

Figure 9: Evolution of failure number of PN "C" from 2018 to 2020 by simulation.

We see that the failure number of PN "C" from December 2018 to December2020 obtained by simulation is 812 (in average) by comparing with 322 in reality. We can explain that:

- Firstly, a part of this different came from the fact that the airline has less aircraft flying in the period 2019-2020 due to the pandemic. Consequently, there was less failure part in that time.
- Secondly, between 2015 and 2020, we have replaced many part number "C" by a new technology, which allows to increase the life time of this part number, so the expected number of failure has decreased over time.

Date	**Number of failure**
2018-12-01	0.00
2019-02-01	143.96
2019-04-01	230.87
2019-06-01	380.33
2019-08-01	417.35
2019-10-01	470.71
2019-12-01	568.86
2020-02-01	609.28
2020-04-01	677.40
2020-06-01	714.54
2020-08-01	745.30
2020-10-01	774.33
2020-12-01	812.22

Table 1: Failure number of PN "C" from Dec 2018 to Dec 2020 by simulation.

Conclusion and Future Work

In this report, we tried to choose and apply the parametric models as well as the non-parametric models in order to estimate the survival probability of each number part on the data set. Then, we validated the best models by using some homogeneity tests. By using these models, we could write the algorithms which allow to forecast the number of failure in the future, the out-of-stock date and evaluate the failure number as a function of time, which could be a significant underlying factor in the proposals of stock optimization strategy of Airbus FHS.

In the future, there would be three major improvements to this project:

- We have not make much use of the "TSN" variable, except from identifying whether a part is new or used. In future work, this would be transform to a categorical variable to be used in a regression model, which hopefully gives a better approximation of survival function in accounting for the aging effect.
- The existence of two technologies ("C" and "C-new") of the same type of parts had inserted a perturbation to our prediction and validation of model. This could be improved by introducing a progressive probability which describes the transition between two technologies.
- Our third algorithm return a population of predicted out-of-stock time T, and a confident interval for that variable. However, the outputs are still based on a naive method which we did not improved because of time constraint of the project. In future, this could be improve by using a reliability model on that simulated population.

References

[1] G. Capodaglio and M. Gunzburger, *Piecewise polynomial approximation of probability density functions with application to uncertainty quantification for stochastic PDEs*, Quantification of Uncertainty: Improving Efficiency and Technology, springer, June 2019, pp. 101–127.

[2] C. Davidson-Pilon, J. Kalderstam, and N. Jacobson, *Camdavidsonpilon/lifelines: v0.25.7.* https://doi.org/10.5281/zenodo.4313838, Dec. 2020.

[3] E. Kaplan and P. Meier, *Non-parametric estimation from incomplete observations*, vol. 53 of Journal of the American Statistical Association, Taylor & Francis, Ltd., 1958, pp. 457–481, 562–563.

Appendices

A Presentation of Python code on github

The github project *Flight Data Analysis* available on:

https://github.com/nghitruyen/Flight_Data_Analysis

We note that the version of `lifelines` package used in our project is not optimized on Windows, so we offer to run this code on Ubuntu Linux in order to optimize performance and execution time.

The notebook file `/doc/FlightDataAnalysis.ipynb` includes all of codes for the simulation process as well as the graphs to visualize.

The `input.txt` file is to calibrate the parameters of simulations:

- `<Data file name>`: name of data file which must be string.
- `<Confidence level>`: confidence level coefficient which must be in $(0, 1)$.
- `<Company name>`: company that we want to forecast, if `company==0` so we want to forecast for all companies. This parameter is only useful for `Number_Failures_Forecasting()` function.
- `<Unit type>`: part number we want to forecast for.
- `<Month>` and `<Year>`: the moment that we want to forecast in the future. This is not useful for `Time_Forecasting()` function.
- `<Number of type unit actually on stock>`: number of product of part number `<Unit type>` available on stock at the moment. This only useful for `Time_Forecasting()` function.
- `<Service level>`: percentage to which customers' need should be met. Only useful for `Time_Forecasting()` function.
- `<Repair rate>`: percentage to which failed products are replaced by repaired products instead of replaced by new.

For estimating of the number of failures in the future, we run `Number_Failures_Forecasting.py` file. For evaluating the failure number as a function of time, we run `Evolution_Number_Failures_Forecasting.py` file. And for estimating of the moment that we can not immediately afford the need of customers, we run `Time_Forecasting.py` file.

Finally, the `/output` directory contains simulation output results.

B Data set format required to execute the code of project

In order for the data prepossessing to work with new data file:

- The new data name should be named without spaces.
- The new data file should contain 3 sheets named: "Removals", "SN list" and "Airlines"
- The clone name of "Removals" should be named as:

  Customer

  Removal date

  P/N

  Description

  S/N

  Maintenance Type

  TSI (Flight Hours) at removal

  TSN (Flight Hours) at Removal
- The clone name of "SN list" should be named as:

  Part Number

  Description

  Serial Number

  Current SN Status Description

  Company

  Hour ageing Since Installation

  Hour ageing Since New

  Since New Date
- The clone name of "Airline" should be named as:

  Company

  Number of aircraft

  FH per aircraft per month

  End of contract