Assignment 2

ADVANCED ECONOMETRICS

Etienne Wijler (Coordinator and Lecturer)

Mariia Artemova (Tutorial instructor)
Gabriele Mingoli (Tutorial instructor)

Georgia Banava (Coding instructor)
Noah Stegehuis (Coding instructor)

Notes and instructions:

1. This assignment is mandatory.
2. The assignment is to be made individually. We have chopped up a large dataset in different bits that you can find `Assignment2_datafiles.zip`. Your dataset is derived from your student ID number. For instance, if your student number is 2601842, take the 5th and 6th digit, in the example 84, and use the dataset `components_84.csv` throughout your case.
3. The deadlines for delivery of this assignment is on Monday, September 26, at 17:00

   There will be no tolerance period for late deliveries. Deliveries after the assigned deadline imply that you have a final grade of zero for the assignment (AG2 = 0).
4. To get the full score for this assignment, the following three things must be done:

   (a) upload your Excel answer sheet in Canvas Assignments using the template file (`AnswerSheetCase2.xlsx`) that we provide to you on Canvas. Only fill in the blue fields and do not modify the column/row structure!! Make sure to also *save* it as `.xlsx`: other template formats (not `.xlsx`) will not receive points. Rename the file **A2solution_2601842.xlsx**, where 2601842 is replaced by your VU student number. Make sure that it contains your name, student number, email address and the assignment number.

   (b) upload a zip file of your runnable R or Python code in Canvas Assignments. The name of the file should be **A2code_2601842_language.zip** , where 2601842 is replaced by your VU student number, and *language* is replaced the language used (python or R), e.g., A2code_2601844_python.zip. The code file(s) should be clear, well commented, and directly runnable, so that it reads the datafile and obtains the results of all questions and prints them. Your initial comments in the file should hold your name and student number.

(c) upload a pdf of your entire code in Canvas Assignments. The name of the file should be **A2code_2601842_language.pdf** , where 2601842 is replaced by your VU student number, and *language* is replaced the language used (python or R), e.g., A2code_2601844_python.pdf . The file should be well readable, with proper indentations and should not contain pictures/photos/screenshots of code snippets.

5. As a standard anti-fraud measure, we will at random select a number of you to explain your code and answers. Failure to explain your answers will result in a deduction of credits for this assignment.

6. For the support for the assignments, carefully read the announcement we put out at the start of the course and consult the discussion boards related to the assignments.

7. Warning: coding might feel as a frustrating exercise at first, certainly if you have not done it often before and if you are making small mistakes every time. But take heart and persevere: completing a coding task also feels very satisfying after all the bits align and you actually made it work completely by yourself!

We wish you success!!

VU VRIJE UNIVERSITEIT AMSTERDAM

Equipment Failure Prediction

1 Background

A key component in many business processes is equipment failure prediction. For example, grocery delivery companies like Picnic have an interest in predicting whether one of their vehicles is in need of repair because one of its components is at risk of failing. If detected too late, the vehicle might not be available at the required time for deliveries. This results in extra costs and consumer dissatisfaction.

You receive a dataset consisting of failure times of specific vehicle components of a vehicle park. The failure times are determined on the basis of regular inspections by company mechanics. If a component is deemed too much at risk of failing before the next inspection, the component is replaced and a failure time of the old component (equal to the repair date) is put into the system. If the new component is failing directly upon installation (which happens sporadically), it is of course directly replaced again and a failure time of zero days is recorded.

You want to study the expected failure times of the different components. A failure time duration (FTD) is defined as the time between the placement and re-placement of the component.

2 Data and descriptives

Each row in your data file corresponds to the k_j-th replacement of some component j, where $k_j = 1,\ldots,n_j$, and n_j is the number of new components of type j installed across all vehicles in the sample over the whole sample period. The dataset holds the following entries:

- the start date, i.e., the installation date of the new component in the vehicle, supplied as a string in the format dd/mm/yyyy;
- the failure date, i.e., date when the component was replaced by the mechanic, supplied as a string in the format dd/mm/yyyy;
- the FTD in years, i.e., the fraction of a year elapsed from the installation date of the component to its failure date, supplied as a float;
- the type of component that failed and was replaced, supplied as a string;

- the average road quality the vehicle drove on during the life of the component that was replaced (from very bad (0) to asphalt concrete quality or better (1)), supplied as a float;
- the average length of a trip in minutes between two customers during the lifetime of the component, or between customer and the depot, supplied as a float.

The dataset is currently sorted on the start date of each component in a particular order. **Do not change this order of the data!!**

The dataset has one peculiar feature. At the end of the sample, it also records all components that have not yet failed by that time. For this particular final time, you thus see many entries in the database. These data[1] also carry information on the lifetime of the components: we know that the component did *not* fail up to that time, so it at least has some reasonable quality. We come back to this point later.

Make some basic plots and descriptive statistics of the data before you start analyzing it. E.g., look at the proportions of the different components, the number of vehicles, the lifetime of the different components (FTD) over time, etc. This will help you interpret your results later on.

3 Static statistical model and inference

To model the FTDs $x_{i,j,k}$, we distinguish two cases: (1) the component failed before the end of sample, and (2) the component survived until the end of the sample.

Case (1): Assume that each FTD is drawn from an exponential distribution with intensity parameter $\lambda_{i,j,k}$ and pdf

$$p(x_{i,j,k};\lambda_{i,j,k}) = \lambda_{i,j,k}\ \exp(-\lambda_{i,j,k}\cdot x_{i,j,k}), \tag{1}$$

where $x_{i,j,k}$ denotes the k-th FTD of vehicle i and component j, with $k = 1,\ldots,n_{i,j}$. Assume that the FTDs are independent conditional on the average road quality and average trip length of the vehicles, and specify the intensity parameter as

$$\log\lambda_{i,j,k} = \beta_{0,j} + \beta_{1,j}RQ_{i,j,k} + \beta_{2,j}ATL_{i,j,k}, \tag{2}$$

where $RQ_{i,j,k}$ and $ATL_{i,j,k}$ denote the road quality and the average trip length for the k-th FTD of vehicle i and component j, respectively. Then if the FTD

[1] Note the word data is always plural in our writing. Also do this in your own work, including your thesis.

corresponds to a component that failed before the end of the sample, its density is given by the pdf above.

Case (2): If, however, the FTD corresponds to a component that survived until the end of the sample (time T), then the probability of observing that FTD equals one minus the probability of failing before T, i.e.,

$$P\left[x_{i,j,k};\lambda_{i,j,k}\right] = 1 - \int_0^{x_{i,j,k}} p(z;\lambda_{i,j,k})\,\mathrm{d}z = \exp\left(-\lambda_{i,j,k}\cdot x_{i,j,k}\right). \tag{3}$$

The mean of an exponential distribution with parameter $\lambda_{i,j,k}$ is equal to $\lambda_{i,j,k}^{-1}$.

Combining the two cases, we obtain the log-likelihood expression

$$\ell_{i,j,k}(\theta) = \begin{cases} \log p(x_{i,j,k};\lambda_{i,j,k}), & \text{if the observation is of type 'Case 1',} \\ \log P(x_{i,j,k};\lambda_{i,j,k}), & \text{if the observation is of type 'Case 2'.} \end{cases} \tag{4}$$

where θ contains all parameters that we want to estimate, such as the betas. The total log likelihood contribution corresponding to a particular component j then equals

$$\ell_j(\theta) = \sum_i \sum_{k=1}^{n_{i,j}} \ell_{i,j,k}(\theta), \tag{5}$$

while the total log likelihood corresponds to $\ell(\theta) = \sum_j \ell_j(\theta)$.

All questions in this case should be answered in the template `AnswerSheetCase2.xlsx` to be uploaded to canvas!

Question 1. Pool all components j, vehicles i, and replacement counts k into one dataset, i.e., use all the data, assuming $\beta_{m,j} \equiv \beta_m$ for $m = 0,1,2$, such that the coefficients are the same across components. Code up the *average* log likelihood corresponding to the above statistical model and compute the value of the *average* log likelihood at the parameters $\beta_0 = 1$, $\beta_1 = -0.1$, $\beta_2 = -0.5$.

NB1: of the log likelihood, *not* the *negative* log likelihood. We compute the average for numerical stability (see also the asymptotic results later in the course). However, note that in a number of questions later we also ask for the *total* rather than the *average* log likelihood value, as the *total* log likelihood is typically what we need for testing and for computing model selection criteria. You obtain the average log likelihood in this particular question by dividing the total log likelihood by the number of observations, in this case $\sum_j \sum_i n_{i,j}$.

NB2: to help you a bit on your way, we provided the value of your average log likelihood at the parameters $\beta_0 = 1.5$, $\beta_1 = 0.1$, $\beta_2 = 0.1$ in the file

`Assignment2_Hint_file.xlsx`. You should be able to obtain the same number (safe some small numerical differences depending on the programming language used).

Now relax the assumption from Question 1 that $\beta_{m,j} \equiv \beta_m$ for $m = 0, 1, 2$ again, and allow the coefficients $\beta_{0,j}$, $\beta_{1,j}$, and $\beta_{2,j}$ to be different across components j.

Question 2. For each component, and using the data for that component only, find the maximum likelihood parameter estimates of $\beta_{0,j}$, $\beta_{1,j}$, and $\beta_{2,j}$ by maximizing the average log likelihood $\ell_j(\theta)/\sum_i n_{i,j}$. Report the results in at least 3 decimals. Also report the value of the maximized *total* log likelihood $\ell_j(\theta)$ at that point (so not the average log likelihood, nor its negative). Do not forget and follow the NBs below.

(challenge part for some extra points, so can be skipped) For the component `tire`, compute the standard errors of the maximum likelihood parameter estimates in at least 4 decimals. Report the outcomes for either the inverse Hessian, inverse OPG, or sandwich standard errors for each of the 3 coefficients as requested in the quiz (but compute all three (inv Hessian, inv OPG, sandwich) of them for yourself to see the differences).[2]

NB1: starting values for the optimizer are always an issue. For this question, we want you to follow the following steps. As starting value for θ, use $\beta_{1,j} = \beta_{2,j} = 0$ and $\beta_{0,j} = -\log \bar{x}_j$, with $\bar{x}_j = \sum_i \sum_{k=1}^{n_{i,j}} x_{i,j,k} / \sum_i n_{i,j}$.
This should provide reasonable starting values, as $\lambda_{i,j,k}^{-1}$ is the expected FTD. Of course, in real life you will have to find reasonable starting values yourself, and the corresponding reasoning. But that is what you build by experience by working more and more with these models.

[2] From your mathematical statistics you know you can compute the standard errors by taking the square root of the diagonal elements of one of the following matrices: $-H_j^{-1}/\sum_i n_{i,j}$ (inverse Hessian based), $G_j^{-1}/\sum_i n_{i,j}$ (OPG based), and $H_j^{-1} G_j H_j^{-1}/\sum_i n_j$ (sandwich), where

$$H_j = H_j(\theta) = \frac{\partial^2 \mathscr{L}_j(\theta)}{\partial\theta\partial\theta^\top},$$

$$G_j = G_j(\theta) = \frac{1}{\sum_i n_{i,j}} \sum_{i=1}^{} \sum_{k=1}^{n_{i,j}} \frac{\partial \log p\left(x_{i,j,k}; \lambda_{i,j,k}(\theta)\right)}{\partial\theta} \cdot \frac{\partial \log p\left(x_{i,j,k}; \lambda_{i,j,k}(\theta)\right)}{\partial\theta^\top}$$

$$\mathscr{L}_j(\theta) = \frac{1}{\sum_i n_{i,j}} \sum_{i}^{} \sum_{k=1}^{n_{i,j}} \log p\left(x_{i,j,k}; \lambda_{i,j,k}(\theta)\right).$$

You can use built-in numerical derivatives packages to compute these derivatives.
Very important: note that the OPG (the outer-product-of-gradients) $G_{i,j}$ is *not* the outer product of the gradient of the likelihood. As the likelihood is maximized, this gradient of the likelihood should be roughly zero in all elements, causing a matrix with only zeros. Instead, the OPG is the variance (or outer product) of the scores of the log likelihood *contributions* $\log p\left(x_{i,j,k}; \lambda_{i,j,k}(\theta)\right)$.

NB2: again, to help you on your way, we have provided the ML estimates and total log likelihood value for the component `buckle`, including standard errors in `Assignment2_Hint_file.xlsx`. You can first try to replicate those. If you are not successful in this, you can try different starting values or different optimizers to make sure you are not stuck at a local optimum. Also make sure your maximum number of iterations is set high enough for the optimizer. Based on elaborate testing from our side, we are confident that *your ML parameter estimates should deviate no more than 0.1 from the values in the hint file, whereas the total log likelihood value should deviate no more than 0.5 from that in the hint file. These error margins also apply to all following questions. We maximized the likelihood in R by using `optim()` and in Python with `scipy.optimize.minimize()`. You are free to use different optimizers, although these two seem to work quite well.*[3]

NB3: it might happen that your optimizer does not converge for the above starting values. Make sure that this is not due to the maximum number of iterations for the optimizer being too low: put the maximum number of iterations to at least 5000. If the problem persists, you can try different starting values or different optimizers. If it still persists, you report the outcomes of the non-converged optimizer. Also later in your program and in your work: *always* check the convergence status of your optimizer (but then do not just report a non-converged estimate). Finally, it could be that the optimizer tolerance is set to rigidly for your series: this would show if the average log likelihood hardly changes, even though the optimizer keeps changing the parameters a bit. You could set the tolerance to a larger value, but always be a careful with this (in other settings it might get you stuck in a local optimum).

4 Filtering and estimation

For one of the components, you have temporarily switched manufacturers during some time of the sample, as this manufacturer was cheaper. Because you suspected the quality of this new supplier was worse, you switched back after some time to the original supplier. For one of the other components, you also suspect that there have been changes in the quality of the supplier, either for better or worse.

To investigate the above two suspicions, you modify your statistical model to have additional time-variation in the intensities $\lambda_{i,j,k}$. You do this by pooling all vehicles for a specific component j, while ensuring to leave the order of the data

[3] If your ML parameter estimates are within a 0.1 margin of the hint answers, but your total log likelihood deviates substantially, please contact your coding instructor.

based on starting dates intact. You re-specify the intensity as

$$\log \lambda_{j,m} = \beta_{0,j,m} + \beta_{1,j} RQ_{j,m} + \beta_{2,j} ATL_{j,m}, \tag{6}$$

$$\beta_{0,j,m+1} = \gamma_{0,j} \cdot (1 - \gamma_{1,j}) + \gamma_{1,j} \beta_{0,j,m} + \gamma_{2,j} \cdot \left(\mathbf{1}_{j,m} - \lambda_{j,m} x_{j,m} \right), \; m \geq 1 \tag{7}$$

where $m = 1, ..., \sum_i n_{i,j}$ "corresponds" to the double index (i,k) which uniquely identifies all points and it is running through all observations in the subset of the data for component j. For initialization consider $\beta_{0,j,1} = \gamma_{0,j}$. In (7), the indicator function $\mathbf{1}_{j,m} = 1$ if the m-th FTD for component j ended *before* the end of the sample, and zero otherwise.

Question 3. For this question, only use the data for the component `tire`. Code up the likelihood corresponding to the dynamic statistical model (with the filter for $\beta_{0,m}$) and compute the value of the *average* log likelihood at the parameters $\beta_{1,j} = -0.1$, $\beta_{2,j} = -0.5$, $\gamma_{0,j} = 1$, $\gamma_{1,j} = 0.8$, $\gamma_{2,j} = 0.025$.

NB1: compute the value of the average log likelihood, *not* the *negative* of it.

NB2: to help you a bit on your way, we provided the value of your average log likelihood at the parameters $\gamma_{0,j} = 1.5$, $\gamma_{1,j} = 0.95$, $\gamma_{2,j} = 0.01$, and $\beta_{1,j} = \beta_{2,j} = 0.1$ in `Assignment2_Hint_file.xlsx` to check yourself further.

Question 4. For each component, and using the data for that component only, find the maximum likelihood parameter estimates of the dynamic model. Report the results in at least 3 decimals. Also report the value of the total maximized log likelihood at that point (so not its negative, and not the average).

NB1: starting values for the optimizer are even more of an issue for dynamic models! Here, you should use the following approach. As starting value for θ, use the parameter estimates from question 2. More specifically, let $\hat{\beta}^*_{0,j}, \hat{\beta}^*_{1,j} \hat{\beta}^*_{2,j}$ be the optimal values for the parameters you obtained for the static model in question 2. Then set the initial values for the dynamic model as $\gamma_{0,j} = \hat{\beta}^*_{0,j}$, $\beta_{1,j} = \hat{\beta}^*_{1,j}$, and $\beta_{2,j} = \hat{\beta}^*_{2,j}$. Finally set the initial values for $\gamma_{1,j} = 1$ and $\gamma_{2,j} = 0.025$.[4] [do you understand why?] Of course, in real life you will have to find reasonable starting values yourself, and the corresponding reasoning.

NB2: again, to help you on your way, we have provided the ML estimates and total log likelihood value for the component `buckle` in `Assignment2_Hint_file.xlsx`.

Question 5. For each of the components, and using the data for that component only, make a plot with on the horizontal axis the starting time, and on the

[4] If you did not manage to get the estimates in question 2, use $\beta_{1,j} = \beta_{2,j} = 0$, $\gamma_{0,j} = -\log \bar{x}_j$, with $\bar{x}_j = \sum_i \sum_{k=1}^{n_{i,j}} x_{i,j,k} / \sum_i n_{i,j}$, $\gamma_{1,j} = 1$ and $\gamma_{2,j} = 0.025$.

vertical axis (1) the maximum likelihood estimate $\hat{\beta}_{0,j}$ from question 2 as a solid black line, and (2) the value of $(\hat{\beta}_{0,m})$ as a red dashed line. Here, $\hat{\beta}_{0,1} = \hat{\gamma}_{0,j}$ from question 4 and $\hat{\beta}_{0,m+1} = \hat{\gamma}_{0,j}(1 - \hat{\gamma}_{1,j}) + \hat{\gamma}_{1,j}\hat{\beta}_{0,m} + \hat{\gamma}_{2,j}(\mathbf{1}_{j,m} - \hat{\lambda}_{j,m}x_{j,m})$, and $\log \hat{\lambda}_{j,m} = \hat{\beta}_{0,m} + \hat{\beta}_{1,j}RQ_{j,m} + \hat{\beta}_{2,j}ATL_{j,m}$. From these figures, which component is most likely to have suffered a trending pattern in the quality? And which component is most likely to have a suffered a temporary break in quality? [Hint: if you think you see a pattern, relate it to the variation of $\hat{\beta}_{0,m}$ on the vertical axis. Small variations are not a convincing signal in general.]

Question 6. Compare the likelihoods at the optimum (of all five components) between questions 2 and 4. Which two components show the strongest evidence of a log likelihood increase (and thus a change in supplier quality)?

Background: the models you work with in this case have many applications. Here you use them to predict component failure times. In the financial industry, they are used to predict bankruptcy of clients or the event of a financial crisis. In marketing they can be used for predicting the time until a client orders your product, or the time spent browsing your website. In climate, they can be used to predict the next hurricane, earthquake, or flood interval. In medicine, one possible use is to predict the time until someone's immune system has relaxed too far to become prone to covid again. Or in public policy, the time until young versus old people find a job in today's market. In short: many, many application areas.