Survival Analysis
Report File
Group 4

Submitted by:

Kanak Rana AU759379,
Purbak Sengupta AU759326

26 Apr. 2024

Contents

1 Introduction

Survival analysis is like a crystal ball for engineers—it helps us predict when things will happen, like how long a machine will work or how long customers will stick with a business. In this project, we're diving deep into survival analysis algorithms for employee turnover. Our goal? To create and compare different ways of predicting these time-to-event scenarios, focusing on two main things: accuracy and clarity.

We want our predictions to be spot-on, but we also want them to make sense to regular end users—not just experts. So, we're testing out different algorithms to see which ones give us the best of both worlds. We'll use simple metrics to measure how good they are, and we'll compare them to the best methods out there.

In this report, we'll try to figure out which algorithm works best, and hopefully, making a meaningful contribution to how we understand and predict time-based events in engineering and beyond.

2 Methods and Materials

In this study, we adopted a structured approach to explore turnover prediction using survival analysis techniques. Survival analysis, a branch of statistics, is specifically designed to analyze time-to-event data, making it well-suited for predicting events like turnover in organizational settings.

Our methodology involved several key steps. First, we meticulously prepared the data, performed Exploratory Data Analysis (EDA) which helped us to gain insights into the structure, quality, and potential relationships within the data. We then selected appropriate algorithms for our task, leveraging the lifelines library in Python, which offers a comprehensive suite of survival analysis tools.[1]

One of the primary methods we employed was the Kaplan-Meier estimator, which allowed us to estimate the survival function of our dataset. This method provides valuable insights into the probability of turnover over time, enabling us to understand the underlying dynamics of employee retention.

Additionally, we utilized the Cox proportional hazards model, a powerful tool for predicting turnover based on various covariates. Unlike traditional regression models, the Cox model can handle censored data, where the event of interest (i.e., turnover) may not have occurred for all individuals by the end of the study period. This flexibility makes it particularly well-suited for analyzing longitudinal data common in organizational research.

Throughout our analysis, we emphasized the importance of interpretability and explainability. By visualizing the results of our analyses using techniques such as coefficient plots and partial effects plots, we aimed to provide clear and intuitive insights into the factors influencing turnover prediction.

Furthermore, we conducted thorough evaluations of our models, assessing their performance on both training and testing datasets. This allowed us to

quantify the accuracy of our predictions and identify areas for improvement.

3 Experiment Setup and Results

3.1 Installation of Required Libraries

We utilized the Python programming language along with several libraries including lifelines, numpy, pandas, matplotlib, plotly, and plotly.graph_objects.

3.2 Data Acquisition

- The dataset named "turnover.csv" was obtained, containing information relevant to turnover prediction. It included features such as age, gender, industry, profession, and various personality traits.

```
In [5]: data = pd.read_csv("Downloads/turnover.csv", encoding = "ISO-8859-1")
        data.head(7)
```

Figure 1: Data acquisition for further processing

- Initial inspection of the dataset was performed to understand its structure and contents, ensuring data integrity and relevance to the research question.

3.3 Exploratory Data Analysis (EDA)

- Exploratory Data Analysis (EDA) was conducted to gain insights into the distribution and relationships of variables within the dataset.

```
In [5]: data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1129 entries, 0 to 1128
Data columns (total 16 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   experience    1129 non-null   float64
 1   event         1129 non-null   int64
 2   gender        1129 non-null   object
 3   age           1129 non-null   float64
 4   industry      1129 non-null   object
 5   profession    1129 non-null   object
 6   traffic       1129 non-null   object
 7   coach         1129 non-null   object
 8   head_gender   1129 non-null   object
 9   greywage      1129 non-null   object
 10  way           1129 non-null   object
 11  extraversion  1129 non-null   float64
 12  independ      1129 non-null   float64
 13  selfcontrol   1129 non-null   float64
 14  anxiety       1129 non-null   float64
 15  novator       1129 non-null   float64
dtypes: float64(7), int64(1), object(8)
memory usage: 141.3+ KB
```

Figure 2: Data Info

- Basic statistics such as mean, median, and standard deviation were examined to understand the distribution of variables.

In [6]:
```
data.describe()
```

Out[6]:

	experience	event	age	extraversion	independ	selfcontrol	anxiety	novator
count	1129.000000	1129.000000	1129.000000	1129.000000	1129.000000	1129.000000	1129.000000	1129.000000
mean	36.627526	0.505757	31.066965	5.592383	5.478034	5.597254	5.665633	5.879628
std	34.096597	0.500188	6.996147	1.851637	1.703312	1.980101	1.709176	1.904016
min	0.394251	0.000000	18.000000	1.000000	1.000000	1.000000	1.700000	1.000000
25%	11.728953	0.000000	26.000000	4.600000	4.100000	4.100000	4.800000	4.400000
50%	24.344969	1.000000	30.000000	5.400000	5.500000	5.700000	5.600000	6.000000
75%	51.318275	1.000000	36.000000	7.000000	6.900000	7.200000	7.100000	7.500000
max	179.449692	1.000000	58.000000	10.000000	10.000000	10.000000	10.000000	10.000000

Figure 3: Basic statistics of the Data

3.4 Visualization

- Visualizations were created using matplotlib and plotly libraries to explore relationships between different variables in the dataset.
- Bar plots, box plots, and violin plots were generated to visualize the distribution of categorical and continuous variables across different event categories.
- The shape of the violin plots will show the distribution of trait scores, with wider areas indicating more data points clustered around that score. The box plot inside each violin plot will provide additional information about the quartiles and median of the distribution

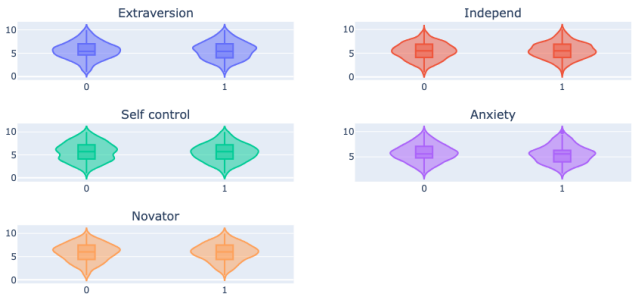


Figure 4: Violin plots showing the distribution of trait scores

```
In [9]: fig = px.box(data, x="event", y="age", color="gender")
        fig.show()
```

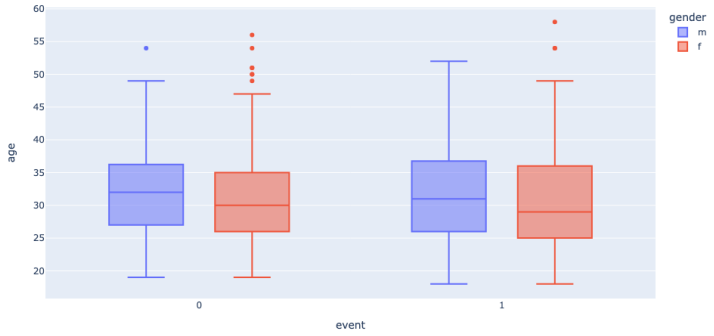
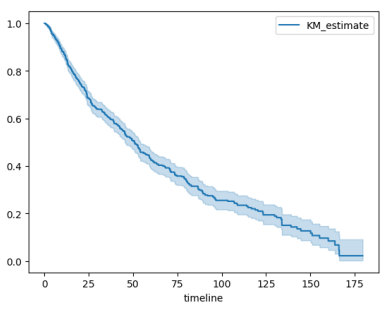


Figure 5: Age distributions among genders for each event

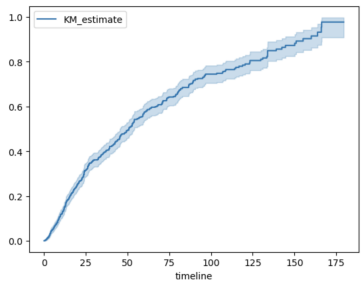
- Similarly, we have generated plots for distribution of professions, industries and events in the dataset as well.

3.5 Survival Analysis with Kaplan-Meier Estimator

- The Kaplan-Meier estimator from the lifelines library was utilized to estimate the survival function of the dataset.
- Survival curves were plotted to visualize the probability of turnover over time, providing initial insights into the event of interest.[2][3]
- The CDF represents the probability that the event of interest (i.e, turnover) occurs before or at a specific time point.



(a) KM Curve



(b) Estimated cumulative density function (CDF)

Figure 6: Kaplan-Meier Estimator

```
In [21]: print(f"Median survival time = {kmf.median_survival_time_}")

         Median survival time = 50.72689938
```

Figure 7: Median survival time

- It suggests that after around 50.73 units of time, half of the individuals are expected to have experienced the event.

3.6 Data Processing

- Data preprocessing steps were undertaken to prepare the dataset for Cox proportional hazards modeling.
- Categorical variables were one-hot encoded to convert them into a suitable format for modeling.

```
In [23]: data_enc = pd.get_dummies(data, columns = ["industry", "profession", "traffic", "coach", "way"], drop_first=True)
         data_enc.head()
```

Out[23]:

	experience	event	gender	age	head_gender	greywage	extraversion	independ	selfcontrol	anxiety	...	traffic_empjs	traffic_friends	traffic_rabrecNErab
0	7.030801	1	0	35.0	1	0	6.2	4.1	5.7	7.1	...	False	False	True
1	22.965092	1	0	33.0	0	0	6.2	4.1	5.7	7.1	...	True	False	False
2	15.934292	1	1	35.0	0	0	6.2	6.2	2.6	4.8	...	False	False	True
3	15.934292	1	1	35.0	0	0	5.4	7.6	4.9	2.5	...	False	False	True
4	8.410678	1	0	32.0	1	0	3.0	4.1	8.0	7.1	...	False	False	False

5 rows × 51 columns

Figure 8: One-Hot Encoding

3.7 Modelling

- Finally, The Cox proportional hazards model is implemented using the lifelines library to predict turnover based on various covariates.[2]
- The model was trained on the preprocessed dataset, and a summary of the model's coefficients was generated to understand the impact of each covariate on turnover prediction.

```
In [25]: #Cox-proportional Hazard Models
cox = lifelines.CoxPHFitter()
cox.fit(data_enc, duration_col="experience", event_col="event")
cox.print_summary()
```

model	lifelines.CoxPHFitter
duration col	'experience'
event col	'event'
baseline estimation	breslow
number of observations	1129
number of events observed	571
partial log-likelihood	-3385.27
time fit was run	2024-04-26 06:58:57 UTC

	coef	exp(coef)	se(coef)	coef lower 95%	coef upper 95%	exp(coef) lower 95%	exp(coef) upper 95%	cmp to	z	p	-log2(p)
gender	0.11	1.12	0.13	-0.14	0.36	0.87	1.43	0.00	0.87	0.38	1.39
age	0.02	1.02	0.01	0.01	0.04	1.01	1.04	0.00	3.26	<0.005	9.81
head_gender	-0.07	0.94	0.10	-0.27	0.13	0.77	1.14	0.00	-0.66	0.51	0.97
greywage	0.49	1.63	0.13	0.23	0.75	1.25	2.12	0.00	3.65	<0.005	11.92
extraversion	0.02	1.02	0.04	-0.05	0.09	0.95	1.09	0.00	0.46	0.65	0.63
independ	-0.01	0.99	0.04	-0.08	0.06	0.92	1.06	0.00	-0.39	0.69	0.53
selfcontrol	-0.05	0.95	0.04	-0.12	0.02	0.89	1.02	0.00	-1.40	0.16	2.62
anxiety	-0.05	0.95	0.03	-0.12	0.01	0.89	1.01	0.00	-1.59	0.11	3.16

Figure 9: Summary of the model's coefficients

- Visualizations such as coefficient plots and partial effects plots were analyzed to interpret the results of the model and understand the influence of different covariates on turnover prediction.

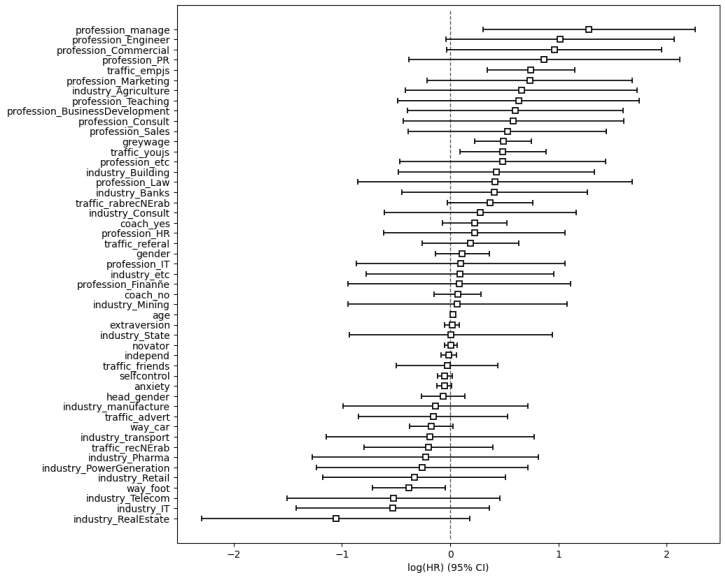


Figure 10: Coefficient Plot

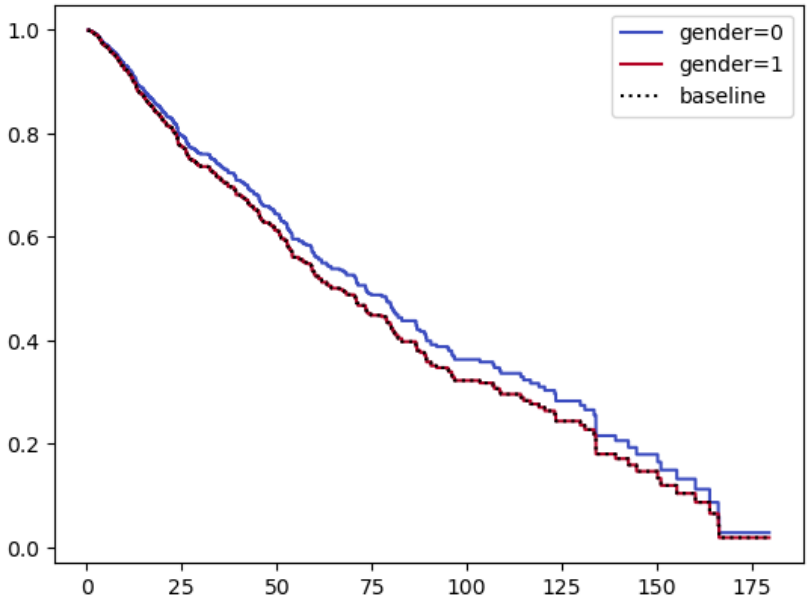


Figure 11: Partial Effects Plot

3.8 Accuracy

- Kaplan-Meier (KM) Concordance Index: 0.0011611030478955006
  This is a very low C-Index, suggesting that the Kaplan-Meier model's predictions are not significantly better than random chance i.e, the model does not effectively predict the survival times of individuals in the dataset.
- Cox Proportional Hazards (CoxPH) Concordance Index: 0.6899129172714078
  This is a much higher C-Index, indicating that the CoxPH model's predictions are significantly better than random chance. The model appears to be effective at predicting the survival times of individuals in the dataset, with a concordance index of 0.6899, which is quite high.

4 Discussion

Our study found that the Kaplan-Meier estimator revealed a median survival time of approximately 50.73 units, indicating the average duration until turnover within our dataset. This metric serves as a key measure, providing valuable insight into employee retention dynamics. Understanding this median survival time is vital for developing effective strategies to retain employees and improve organizational processes.

In our comparison between the Cox model and the Kaplan-Meier estimator, we noticed some big differences in how well they predict survival times. The Cox model turned out to be much better at making predictions because it can look at lots of factors at once, giving us more accurate results. On the other hand, the Kaplan-Meier estimator, while simpler to understand, didn't predict as accurately as the Cox model. So, when we're using these methods in real-life situations, we need to think about finding a balance between accuracy and simplicity.

5 Conclusion and Perspectives

We found out how survival analysis techniques are really versatile and handy when it comes to dealing with data about how long things take to happen, especially when we don't know everything that might affect the outcome. This helped us get a really good grasp of what factors might be affecting turnover in organizations.

As we look ahead, there's a wealth of opportunities to deepen our understanding of turnover dynamics. One avenue for improvement lies in enhancing our Cox models by incorporating additional factors such as company culture and work-life balance. These elements are known to influence employee retention but were not fully explored in our current analysis. By integrating them into our models, we can uncover new insights into how organizational factors impact turnover rates.

Additionally, validating our findings on external datasets from diverse companies can provide valuable validation and enhance the scalabilty of our conclusions. This approach allows us to test the robustness of our predictive models across different organizational contexts and validate the effectiveness of our proposed strategies for reducing turnover.

To sum it all up, our project has given us some really interesting insights into how we can predict turnover using survival analysis techniques. By using these methods, we've found some valuable information that can help organizations make better decisions and keep their employees happy. It's been a really cool journey, and there's still so much more we can explore in the future.

References

[1] B. Kent, "Applications of Modern Survival Modeling with Python", Open Data Science, Talk, 2022.

[2] A. Pandey, "Survival Analysis: Intuition and Implementation in Python", TDS, Tutorial, 2019. .

[3] L.L. Johnson, "Conceptual Approach to Survival Analysis", IPPCR, Talk, 2015.