Team Mini Project: CAPO and Time to Clinical Stability
DS450-01
Data Science Senior Capstone

Project Goal: To generate a survival analysis curve and develop predictive models using Linear/Logistic Regression for a real-life data set.

Scenario: You are a data analyst working for a health research company. One of the researchers in your organization is interested in the outcomes for parents hospitalized with Community-Acquired Pneumonia. The researcher was made aware of a multi-year retrospective cohort study on CAP conducted by a physician in Louisville, Kentucky who specializes in infectious diseases. Starting in 2001, this study collected data from hospitals around the world on CAP. You have been provided with a snapshot of the data from 2017 and asked to perform several linear/logistic analyses on the data. The researcher is interested in building linear and/or logistic models to predict **me to clinical stability (TCS).** TCS is a measure used by physicians treating CAP parents to determine when they can be discharged and continue treatment at home. In this study, four criteria were used:

a. Improvement in cough/shortness of breath.
b. Temperature of less than 37.8C or 100F.
c. White blood cell counts of 10% or more lower than the previous day.
d. Whether the patient can eat or take pills orally.

These values are reflected in the data with a 0=no, 1=yes for each criterion for up to seven (7) days. Once a parent has a 1 for all four criteria for a particular day, they are clinically stable, and that day is their TCS. For example, if a patient reaches all four criteria on day 4, their TCS is 4. The data are only collected for seven (7) days. If a patient has not reached clinical stability by the end of day seven, their TCS is automatically calculated as 8 (called right-censoring).

The researcher wants to see if TCS can be predicted using basic demographic data (age, sex) and examination data collected at presentation which is the first me the patient is examined by a physician. These data include items such as height, weight, temperature, white blood cell count, etc. In the data set demographic data starts with “dem_” and examination data starts with “exam_”. They are not totally sure linear regression will work, so they also want to see if they can predict whether a patient will meet clinical stability by 7 days or not or if they will have early or late clinical stability which would be defined as clinical stability reached before 4 days (early TCS) or at/after 4 days (late TCS). This will involve running two models.

The researcher is also interested in developing a survival curve that will show the percentage of patients who have reached TCS each day. To the right is an example of a survival curve for veterans with lung cancer. On the left you see that all veterans are alive at the beginning of the study. The percentage drops as veterans die until all veterans have passed away. The light blue shading is the calculated confidence interval. Both Python and R have libraries to calculate survival curves.

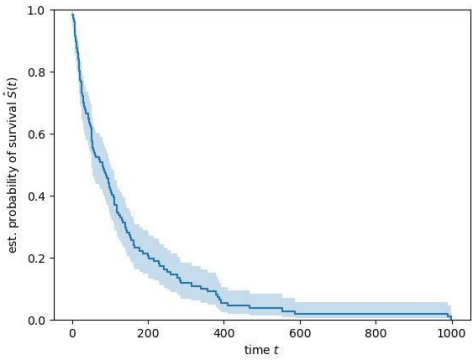

Figure 1: Survival Curve - Veterans with Lung

Cancer

Instructions:

Using the attached data set, your team will clean, prepare, and run multiple models as outlined below:

1. Process the data to get only demographic (dem_), exam (exam_) and chest x-ray (cx_) data. There is a lot of missing data for this data set. You need to consider which records to drop. You'll also need to consider how to impute data when appropriate. Python and R both have libraries for imputation. The TCS is not given in the data set; you will need to calculate it in your code.
2. Generate a survival analysis curve like the veteran lung cancer data. **You are only looking at outcomes for TCS for up to 8 days; your curve will only have 8 steps in comparison to the example curve.**
3. Use the demographic, examination, and chest x-ray data to build a linear regression model to predict me to clinical stability. Evaluate the model using the R^2 value.
4. Build logistic regression models for stable/not stable and early TCS/late TCS and evaluate each model using the same columns in your regression model.
5. For your logistic regression models, check to see if they get better if you use any of the lab data which include columns that start with "lab_".

The output for this project should be:

- Your **thoroughly documented** notebook/script with your analyses using Markdown. Explain what columns/rows of data you dropped, why you selected them, what imputation methods you used and how you decided what to impute versus what to drop, etc. Once you run your models, interpret them, and explain how well or poorly they perform.
- The cleaned data set that you used for your analyses.
- A 5-7 minute presentation of analyses. You want to present all the major steps of your project. All members of your team must be present. You may use either PowerPoint or an alternative format (ex: Jupyter Notebook) for your presentation's backdrop.

Submission: Create a new directory in your git repository for this project and put all your project files (including the data set) in that directory, alongside the chosen presentation format. Upload a link to your GitHub repository in the area provided on Moodle by the deadline specified.