**Progress Report 1:**

**Estimating a Country’s Mortality Rates Using Linear Regression**

Heron Ziegel

**Current Progress:**

This week I took a look at the [Adult Mortality Rate](https://www.kaggle.com/datasets/mikhail1681/adult-mortality-rate-2019-2021) dataset and began the preprocessing stage. The data is contained in a CSV file, which I downloaded and read into a DataFrame using the Pandas library in Python. The dataset is comprised of seven features: country, continent, average population, average GDP, average GDP per capita, average healthcare expenditure per capita, and development level. There are three ground truth columns: adult female mortality rate, adult male mortality rate, and average crude death rate.

The dataset seems to be very clean and well organized. First I used the DataFrame.info() method from the Pandas library to view information about the columns. There are three categorical features: country, continent, and development level. The rest are numeric values, specifically floats. Then I checked the data for null values by dividing DataFrame.isnull().sum() by DataFrame.shape[0], using the first column to extract the number of rows. There were no null values in the dataset.

I used the LabelEncoder function from the sklearn.preprocessing library in order to covert the categorical features into numeric features. Using the labelencoder.fit\_transform() function, I transformed the categorical features country, continent and development level into integer columns. I chose to use label encoding rather than one-hot encoding in order to keep the data simple and avoid creating additional columns.

Then I created a matrix X using the DataFrame.iloc function to pull out only the attribute columns from the DataFrame. I also created three y vectors: y\_female, y\_male and y\_avg. I will use these as the ground truth vectors for the three different models.

**Remaining Work:**

This project will require a linear regression machine learning model. Over the next several weeks I will need to split, normalize, train and test the data for three different linear regression models. This is because there are three ground truth columns, therefore each will need to be tested independently of the others. There is a clear dependency between adult female mortality rate, adult male mortality rate, and average crude death rate. Training crude death rate using adult female mortality rate and adult male mortality rate as attribute columns, for example, may result in a model which weights those two columns very heavily and ignores the other data. This would not be very informative or useful. So instead I will use all three columns as separate ground truths and see whether the other data can be used to predict any or all of them.

I will use the train\_test\_split() function from the sklearn.model\_selection library in order to separate the data into training and testing data. I will normalize both the training and testing sets using the StandardScaler() normalizer from the sklearn.preprocessing library. Then I will use 5-fold cross-validation in order to find a hyper parameter for the data. There are not many rows in this data, so using more folds could be challenging since it would create a very small testing set and may not be very accurate. I made this decision after reading the article “[Multiple-k: picking the number of folds for cross-validation](https://cran.r-project.org/web/packages/cvms/vignettes/picking_the_number_of_folds_for_cross-validation.html).”

Then I will train the linear regression model. I will use Mean Average Error, Mean Squared Error, and Root Mean Squared Error in order to evaluate the model accuracy. If these numbers are very large, I may need to make changes to my code to increase accuracy. Once the error is more acceptable, I will move on to the next step.

Since I have three ground truths, I will have to repeat these steps of training and testing for each of the three ground truths. Each of the three models will be evaluated for accuracy as well. Finally, once all three models are satisfactory, I will attempt to use these models to predict data from outside of the dataset.

I will need to generate my own CSV file by searching for the individual attribute values for countries which were not included in the original dataset. Then I will use Pandas to import the CSV file as a DataFrame and run the new numbers through each of the three models to make predictions. Finally, I will search for the actual mortality rates in these countries and compare it to the predicted values. I would like to visualize some of my findings for the final project report in order to give a more readable understanding of my findings.

**Upcoming Week Plan:**

My focus for next week will be splitting, training and testing the data. Unlike this week, these steps will need to be performed three times for each of the three ground truth values. After splitting the testing data I will normalize both the training and testing sets. I will also be selecting a hyper parameter for each model using 5-fold cross validation, for reasons previously mentioned. This will be the majority of the programming work for the project, but depending on how well the data tests it could require further work in the following week.

The most important thing to keep in mind for the upcoming week will be to maintain separation of variables. Working with three models and copy/pasting code between them could result in errors which are not caught by Jupyter Notebook. I will also need to review hyper parameters, and may need to ask the TA for advice with this step. If I complete the work early, I might start working ahead of schedule and test the models using MAE, MSE and RMSE. This way if the models are not working well, I can start adjusting the code.

**References**

Dataset: Adult Mortality Rate. <https://www.kaggle.com/datasets/mikhail1681/adult-mortality-rate-2019-2021>

Multiple-k: picking the number of folds for cross-validation. <https://cran.r-project.org/web/packages/cvms/vignettes/picking_the_number_of_folds_for_cross-validation.html>