Name: William Burke

Assignment: Case Study 3

Github: https://github.com/wgburke310/DS501-CS3

Notes: All questions are answered both in this document, and in the .rmd file that is attached and posted in the github.

1. Dataset

The dataset I used is from Kaggle, titled "Global Country Information Dataset 2023". This dataset contains numerical information for several countries, with attributes related to demongraphics, economy, environmental factors, healthcare metrics, education statistic, and more. Specifically, I chose this dataset because I wanted to use these attributes to predict a country's C02 emissions, which would provide me with insights into which factors lead to higher C02 emissions.

When cleaning the dataset, I selected a subset of columns to utilize, and removed several records (each representing a country) that had multiple missing values. This produced a clean dataset, which is uploaded to github and was used to train the regression model.

Dataset: https://www.kaggle.com/datasets/nelgiriyewithana/countries-of-the-world-2023?resource=download

2. Selecting an algorithm

The algorithm that I selected is a classic regression algorithm. Regression is a relatively simple machine learning algorithm, and I chose it in this circumstance because I'm trying to generate a prediction based on several attributes. Furthermore, that attribute I'm trying to predict (C02 emissions in tons) is represented by a linear number, meaning it has to be a regression problem as opposed to classification. In my case, the regression model is trained with the following code:

```
2.ML Algorithm for Dataset: Regression
```{r echo=TRUE}
# Setting up a linear regression model
df.lm <- lm(formula = co2_emissions ~ density + land_area_m + agricultural_land + birth_rate + fertility_rate + life_expectancy +
proportion_forested + population_m + urban_population_m, data=df)
summary(df.lm)
```
```

3. Explain the mathematical/statistical details of the algorithm

Regression is a statistical method for "training" a machine learning model by computing a set of "weights" or coefficients that can be used to calculate a single prediction number given a set of inputs. Regression models can be increasingly complex, or very simple like in the case of linear regression. In the simplest form, linear regression is about finding the best fitting 'line' for the dataset that the mathematical model was trained on. This best fitting line is defined by an intercept, along with a coefficient for each of the input variables. In the equation below, a is the intercept, and b * X represents the multiplication of a single coefficient with its respective input value, or "explanatory variable". In my case, I used 9 "explanatory variables", meaning that in the prediction stage each of the 9 inputs would be multiplied by some weight value and summed to calculate the final predicted value.

$\boldsymbol{Y} = \boldsymbol{a} + \boldsymbol{bX}$, where $\boldsymbol{X}$ is the explanatory variable and $\boldsymbol{Y}$ is the dependent variable.

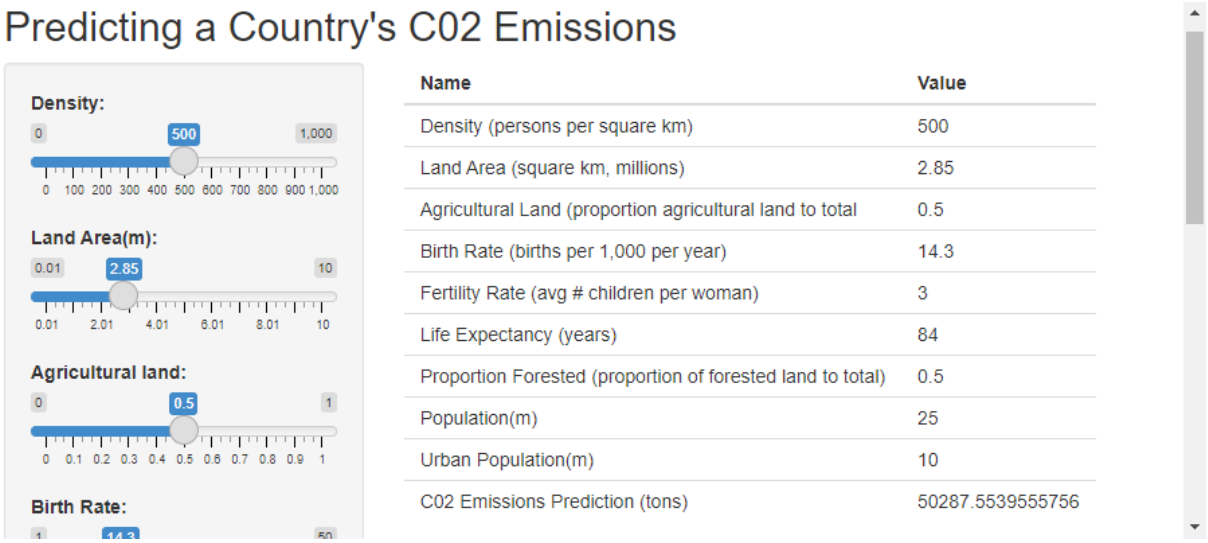
Like any machine learning model, linear regression's statistical methodology is based on trying to minimize some cost function, which in this case is the sum of squares. Pictured below, the sum of squares is the squared difference between the predicted y value, and the true y value. Minimizing this number logically means minimizing the distance between the guess and actual answer.

$$\text{TSS} = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

4. Create a Shiny application

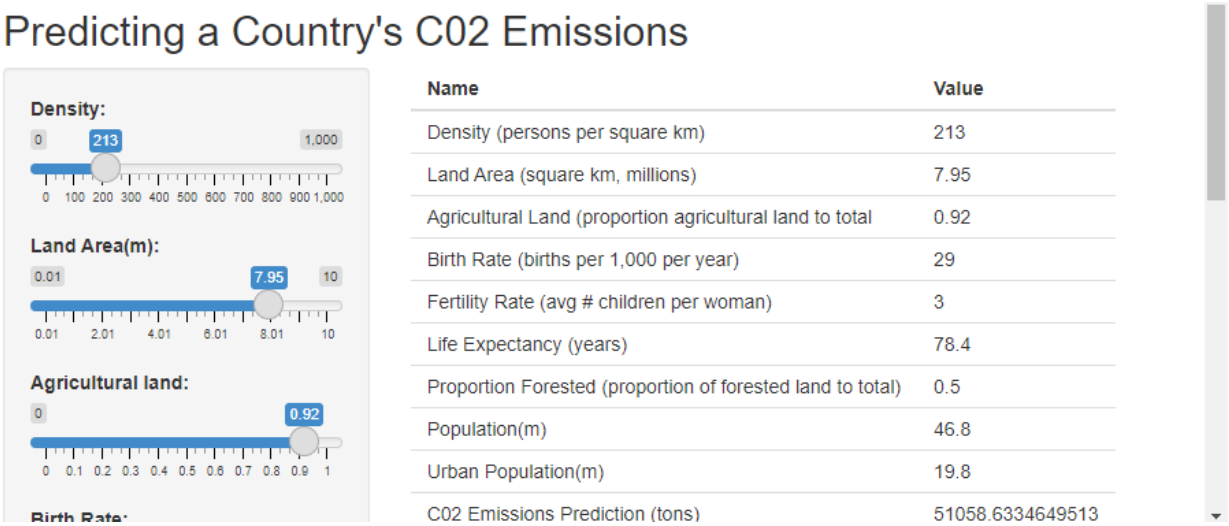
Below is a screenshot of my shiny application, which uses a machine learning model to generate predictions for C02 emissions based on sliding inputs. There are several attributes that are inputted via sliders, which are the same as the ones used to train the model. Since the model is already trained, the user can move the sliders, which changes the input values to the model prediction and generates a new prediction.

Url: https://y946da-william-burke.shinyapps.io/MLapp/



Note: There are 9 sliders total, which can be seen by using the scroll bar on the right.

Here is an additional screenshot showing new values:



5. Additional questions.

What data you collected?

- The dataset I used is from Kaggle, and the dataset contains numerical information for several countries, with attributes related to demographics, economy, environmental factors, healthcare metrics, education statistic, and more. The data was not very clean in its original form, so I had to do some data cleaning to produce good data that I could use to train my model.

Why this topic is interesting or important to you? (Motivation)

- This topic is interesting to me because I think that climate change is one of the biggest problems the world faces today. With C02 emissions being one of the biggest causes of it, any effort to provide scientific information regarding C02 emissions could be valuable. Since climate change is not exactly taken seriously by everyone, providing better data-backed information could help inform progressive policies, or convince the general public of why it's harmful.

How did you analyze the data?

- First, I performed exploratory analysis on the original dataset to gain a better understanding of the dataset, and find inspiration for any data cleaning processes I would have to complete. After that, I cleaned the dataset in preparation for training the regression model. Next, I trained the model using the cleaned dataset, which allowed me to generate predictions given any input. The trained coefficients for each variable provided insights into how each input value affects the final number. The final step of the analysis was to tie everything together in the Shiny app, which essentially allows users to change input variables and visualize how those changes affect the predicted C02 value. The use of this app is analysis in itself, as the process of experimenting with different input variables provides valuable insights.

What did you find in the data?

- One thing I found in the analysis is that a country's population density is not a great predictor of C02 emissions. Regardless of being positive or negative, the scale of a coefficient compared to others tells a lot about the input's impact on the final prediction value. The input values for this

attribute tend to be low to start, and on top of that the coefficient is -1.829, much smaller compared to the others.

```
Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)       268899.814 460682.918   0.584    0.560
density               -1.829     28.239  -0.065    0.948
land_area_m        13612.013  11651.561   1.168    0.244
agricultural_land -92530.826  98198.009  -0.942    0.347
birth_rate         -8791.564  10381.330  -0.847    0.398
fertility_rate     47749.944  73269.127   0.652    0.515
life_expectancy    -3006.398   5194.887  -0.579    0.564
proportion_forested -93029.962 87549.510  -1.063    0.289
population_m        -4525.405    437.740 -10.338   <2e-16 ***
urban_population_m  18443.046    912.328  20.215   <2e-16 ***
---
```

- Another thing I found is that attributes related to agriculture or nature are inversely proportional with C02 emissions. What this means is that a higher proportion of agricultural or forested land means that a country will have a lower carbon emissions value. This is important because many efforts related to combating climate change involve preserving or adding more plants/tree, which is supported b this data.

7. Check in your code into GitHub https://github.com (Links to an external site.) (including data, take a small dataset, few MB)

https://github.com/wgburke310/DS501-CS3