Milestone 2

Feature Engineering

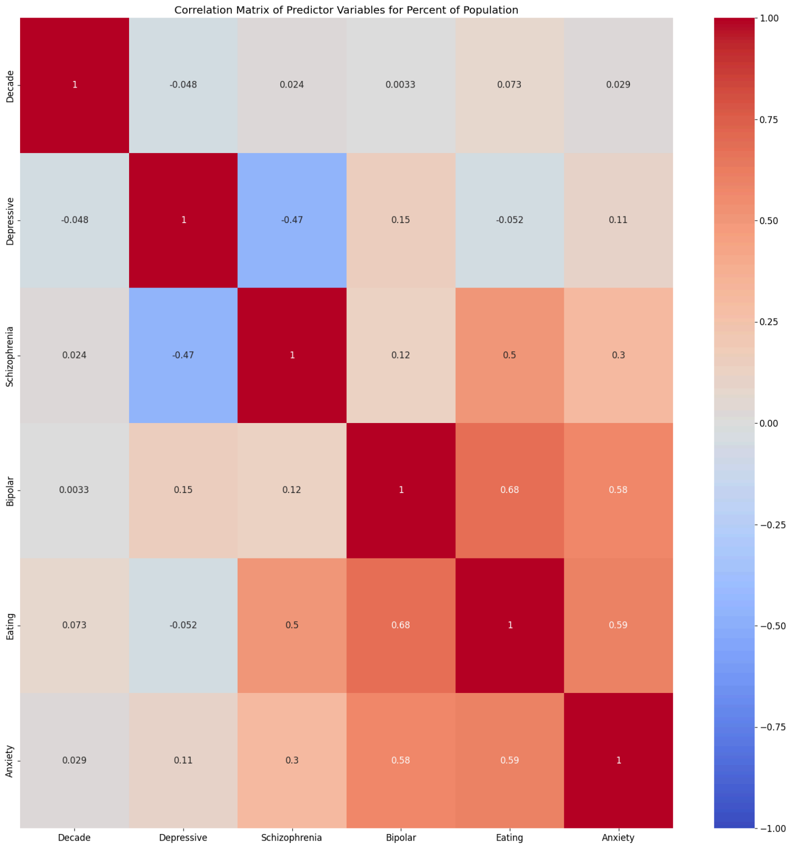
Features were added to refined csv using feature.py. Features were further refined and cleaned for pre-one hot encoding with data_cleaner.py

For the first two csv's of Data 1, one of the features that was created were averages for DALYs and percent population afflicted with the mental illness per world country. In addition to averages, a temporal feature that was added was decade. These csv's were combined with averager.py. Decade was added to see if there was a temporal relation relative to percent or DALYs.
While it would have been very interesting and helpful to create a geographical region based feature to group the world countries to see if location contributed to percent of the population afflicted, a decade and region, and the DALYs per region, which again, would have been amazing, had no feasible way to be done due to the difficulty mapping the countries to regions. These regions pop up again in csv 4, and would have been a great addition, but due to time constraints and no efficient way to map the countries to regions I was not able to implement it. I still wanted to mention this as a possible future feature if time allows it.
For Data 1 Csv 4, the disease impact scores are completely unrelated to one another, and creating an average impact score per country wouldn't make sense to connect to the other data to implement in a model. Thus no features needed to be engineered.
For Data 1 Csv 5, a potential feature created was a combination of treatments, so we could compare treated vs. untreated directly, however, upon closer inspection, trying to connect this to the other data would be too small of a data pool to train a model, as there are only a couple years represented.
Feature engineering for csv6 doesn't make much sense. I initially thought about creating a symptoms vs. no symptoms, however, because this is only for the year 2014, trying to utilize this new feature in a model would be impossible because there simply is just not enough data to connect it to DALYs, percent, or impact score.
For Data 1 csv 7, similar to Data 1 csv 6, feature engineering here doesn't make much sense. I considered making a new feature that would be the average number of countries with data on mental illnesses, however there is only one column and this would result in only one number, which would be useless to train a model and has no connection to the other csv's.
For Data 2, the feature engineered was an aggregate of the number of symptoms that each patient experienced. This will be useful for a model to see if the number of symptoms implies gender or treatment in conjunction with the other features.
For Data 3, the feature engineered was again the number of symptoms. This could be useful to see if number of symptoms is correlated with specific illnesses, or if number of symptoms, along with other features, imply gender.

Feature Importance

Correlation Matrices

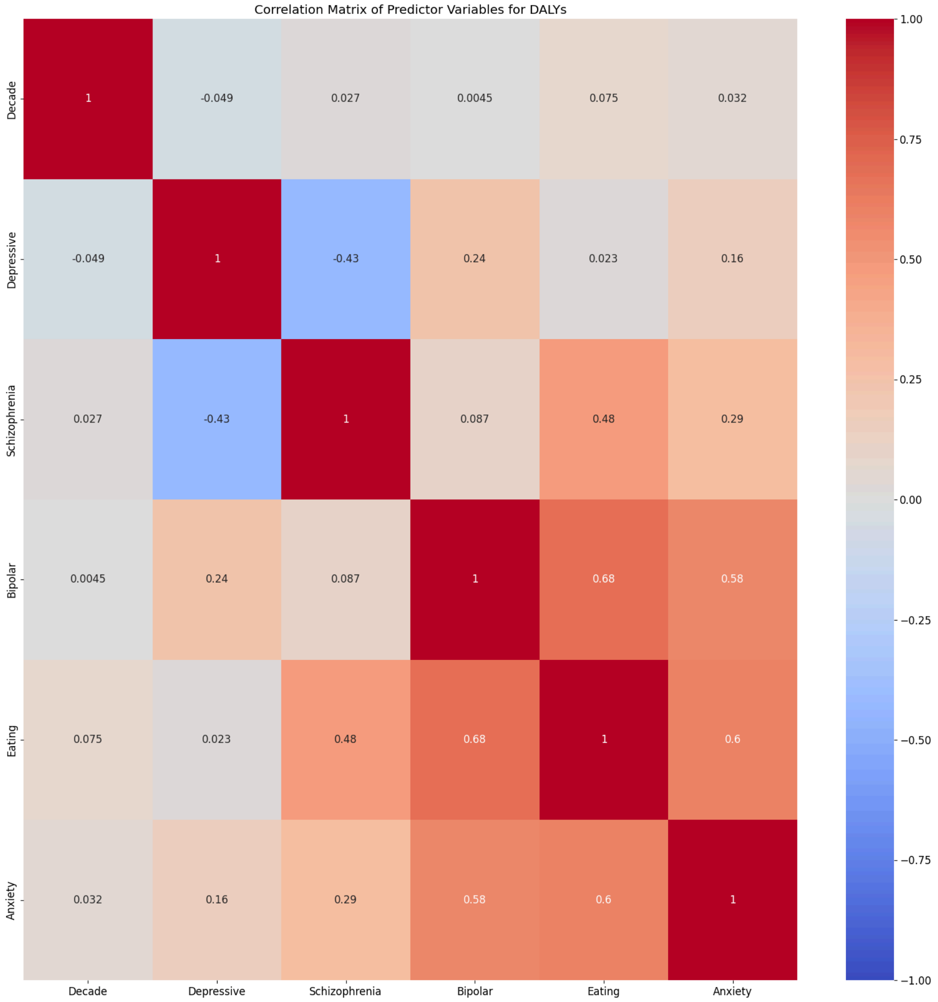
The correlation matrices were created with feature_importance.py



VIF Data

```
         feature     VIF
0          const  164.26
1         Decade   1.014
2     Depressive    1.44
3  Schizophrenia    2.04
4        Bipolar    2.34
5         Eating    2.93
6        Anxiety    1.78
```

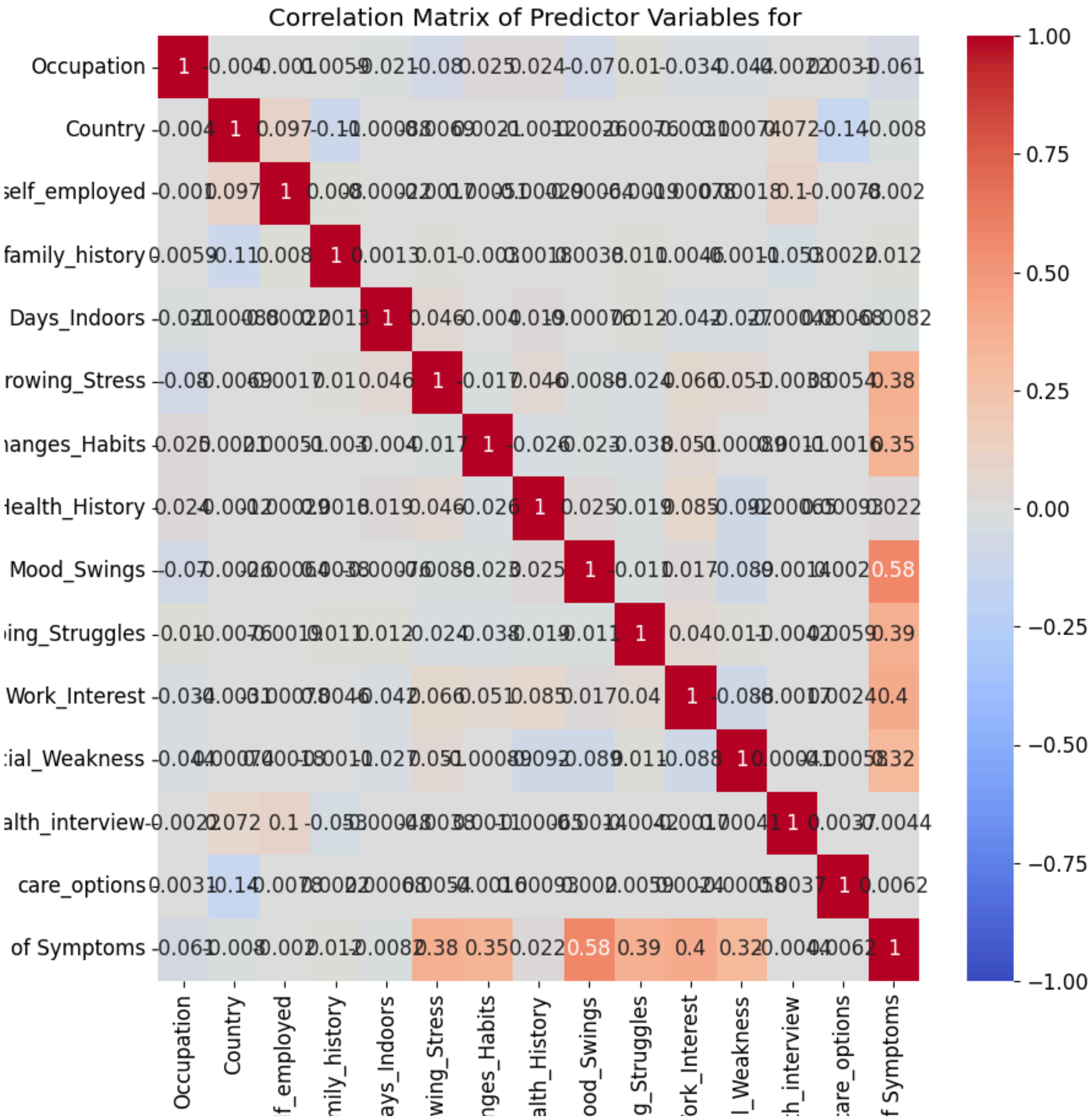
1) Given that the diseases under consideration (Schizophrenia, Depression, Anxiety, Bipolar, Eating Disorder) are known to have distinct etiologies and underlying mechanisms, we do not expect significant multicollinearity among these variables. This expectation is confirmed by the correlation matrix below, which shows low correlations between all disease pairs, and the VIF values listed below, all of which are well below the commonly used threshold of 5, indicating minimal multicollinearity. It does not seem that Decade is correlated in any meaningful way either.



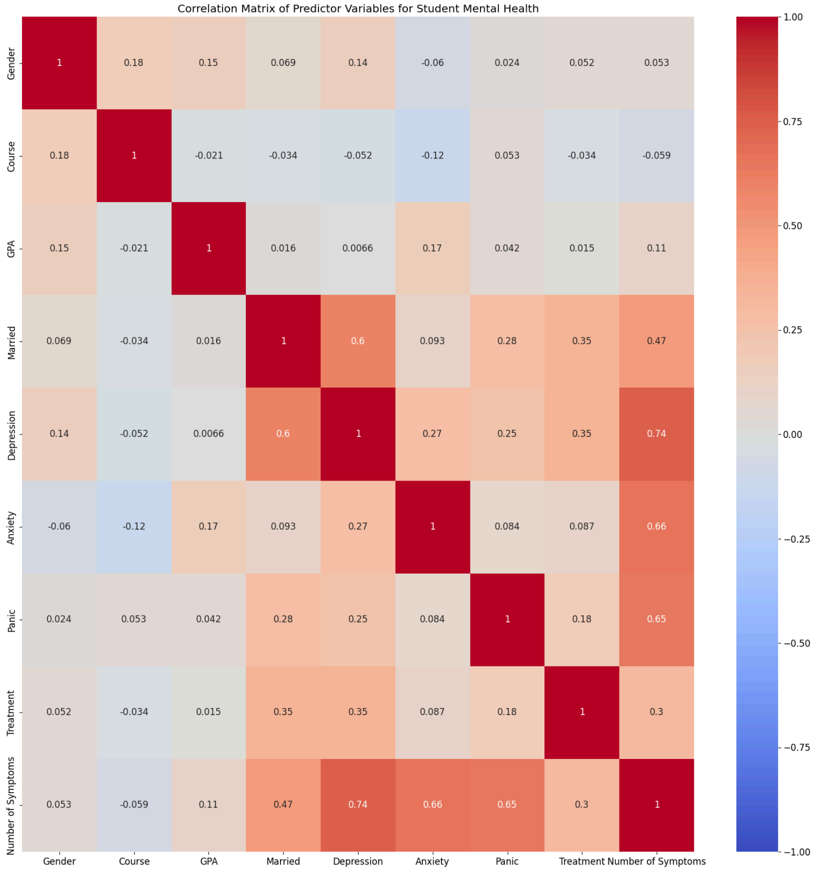
```
VIF Data
       feature      VIF
0        const  137.59
```

```
1         Decade    1.01
2     Depressive    1.42
3  Schizophrenia    1.94
4        Bipolar    2.42
5         Eating    2.93
6        Anxiety    1.80
```

Given that the diseases under consideration (Schizophrenia, Depression, Anxiety, Bipolar, Eating Disorder) are known to have distinct etiologies and underlying mechanisms, we do not expect significant multicollinearity among these variables. This expectation is confirmed by the correlation matrix below, which shows low correlations between all disease pairs, and the VIF values listed below, all of which are well below the commonly used threshold of 5, indicating minimal multicollinearity. It does not seem that Decade is correlated in any meaningful way either.



VIF Data for this correlation matrix kept running into a Nan error I could not fix, but I was able to generate the correlation matrix.



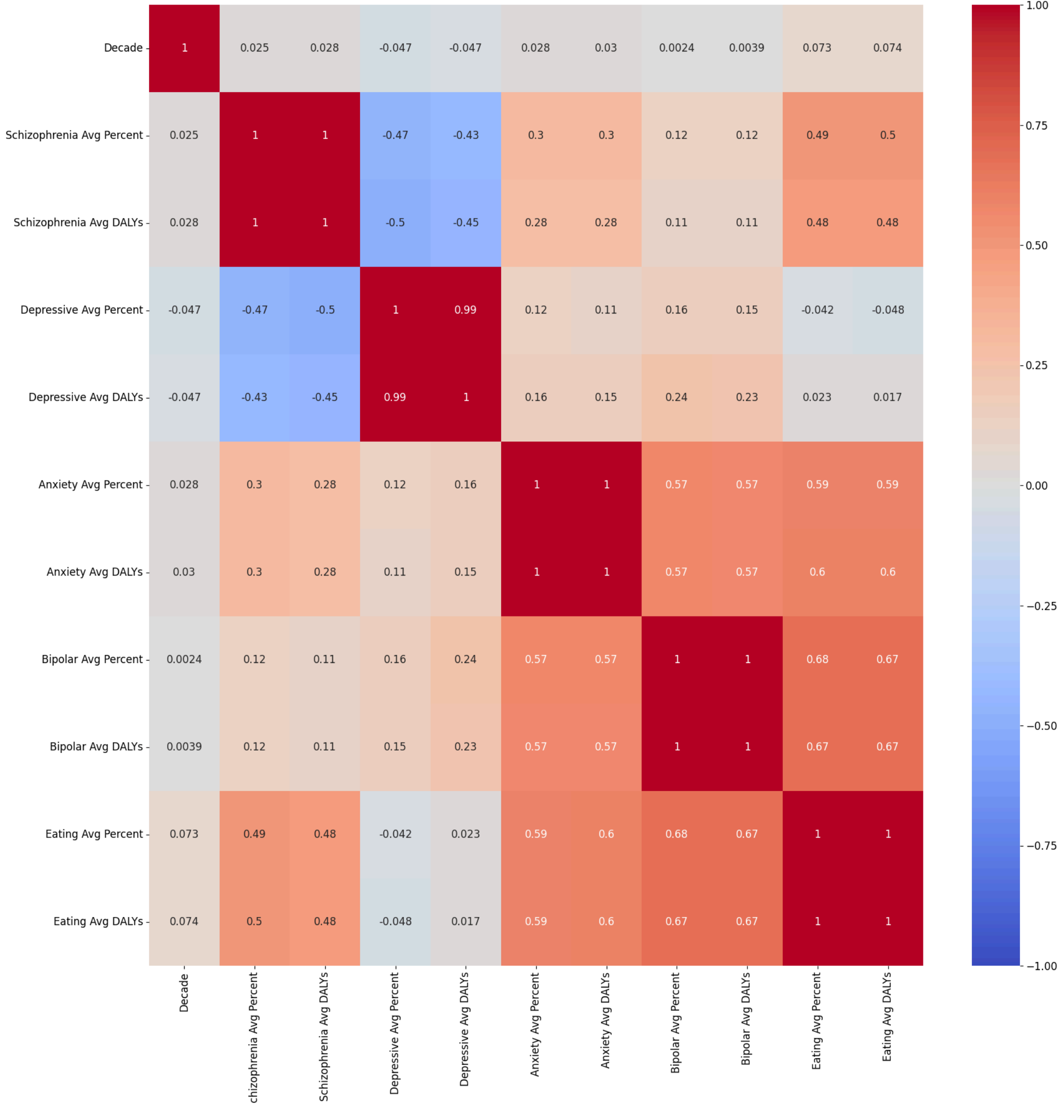
Correlation Matrix of Predictor Variables for Student Mental Health

	Gender	Course	GPA	Married	Depression	Anxiety	Panic	Treatment	Number of Symptoms
Gender	1	0.18	0.15	0.069	0.14	-0.06	0.024	0.052	0.053
Course	0.18	1	-0.021	-0.034	-0.052	-0.12	0.053	-0.034	-0.059
GPA	0.15	-0.021	1	0.016	0.0066	0.17	0.042	0.015	0.11
Married	0.069	-0.034	0.016	1	0.6	0.093	0.28	0.35	0.47
Depression	0.14	-0.052	0.0066	0.6	1	0.27	0.25	0.35	0.74
Anxiety	-0.06	-0.12	0.17	0.093	0.27	1	0.084	0.087	0.66
Panic	0.024	0.053	0.042	0.28	0.25	0.084	1	0.18	0.65
Treatment	0.052	-0.034	0.015	0.35	0.35	0.087	0.18	1	0.3
Number of Symptoms	0.053	-0.059	0.11	0.47	0.74	0.66	0.65	0.3	1

```
VIF Data
            feature      VIF
0             const  16.00
1            Gender   1.102
2            Course   1.057
3               GPA   1.067
4           Married   1.66
5        Depression        inf
6           Anxiety        inf
7             Panic        inf
8         Treatment   1.18
9  Number of Symptoms        inf
```

No VIF score is above 5, indicating minimal multicollinearity and minimal correlation between features. One thing to note, despite all VIF scores being below 5, it seems that depression is closely tied to having all 3 symptoms, when compared to the other symptoms.

Correlation Matrix of Predictor Variables for



```
0                   const    564.740567
1                  Decade      1.040205
2  Schizophrenia Avg Percent   3170.07
3    Schizophrenia Avg DALYs   3250.40
4     Depressive Avg Percent    311.16
5       Depressive Avg DALYs    294.74
6        Anxiety Avg Percent  10824.36
7          Anxiety Avg DALYs  10781.82
8        Bipolar Avg Percent  26697.69
9          Bipolar Avg DALYs  26574.03
```

```
10        Eating Avg Percent   7834.62
11          Eating Avg DALYs   7767.10
```

Because the percents are directly tied to their respective diseases, we would expect the VIF scores to be highly correlated, hence why all are well over 5. This could indicate when utilizing these inputs in a model, they will give worthwhile and accurate output for each disease.

Chi Squared Test

For the Data 2 and Data 3 datasets, these were largely categorical. In order to run the chi squared test on both datasets, the data was converted into binary representation using the pandas one hot encoder. For both datasets, the chi stats and p values were saved to the respective files: Data_2_chi_stats.csv, Data_2_p_values.csv, Data_3_chi_stats.csv, Data_3_p_value.csv.

PCA/LASSO

For the first dataset, PCA or LASSO makes no sense to use. The continuous features cannot be reduced. They are all relevant pieces of information.
For the second and third datasets are categorial, each of the features are necessary and important, and trying to reduce the dimensionality of categorical data would not make sense here.

Data Modeling

Dataset 2

For all the models listed below, they are trained and contained in data_2_models.py

SVM:
For Dataset 2, a SVM model was made to see if the categorical data indicated gender or if the person was in treatment. To split the data into test and train sets, I used an equal split of males and females. For predicting gender, due to the similarity of male and female data, the output was largely female. I used 60% of the dataset for train and 40% for test.

For gender, the model scores are as follows:

Accuracy: 0.49
Precision: 0.49
f1score: 0.65
recallscore: 0.98

For treatment, the output was decently accurate, which is consistent with the model for gender, as the output was largely female, and the bar chart from Milestone one indicates the majority of females get treatment.

Accuracy: 0.77
Precision: 0.71
f1score: 0.83
recallscore: 1.0

Because the dataset was so large, I had to take a much smaller sample set. It could also be that SVM is not a good model for this dataset, as the male-female data is too similar to one another and thus is non-separable, hence largely female output. For treatment, the model performed much better, indicating on this feature, the data is separable.

Decision Tree:
For Dataset 2, a Decision Tree model was made to see if categorical data indicated gender or if the person was in treatment. To split the data into test and train sets, I used an equal split of males and females. For predicting gender, due to the similarity of male and female data, the output was largely female.

For gender, the model scores were as follows:

Accuracy: 0.44
Precision: 0.46
f1score: 0.56
recallscore: 0.72

For treatment, the output perfectly matched the y_test, with an accuracy of 100%

Accuracy: 1.0
Precision: 1.0
f1score: 1.0
recallscore: 1.0

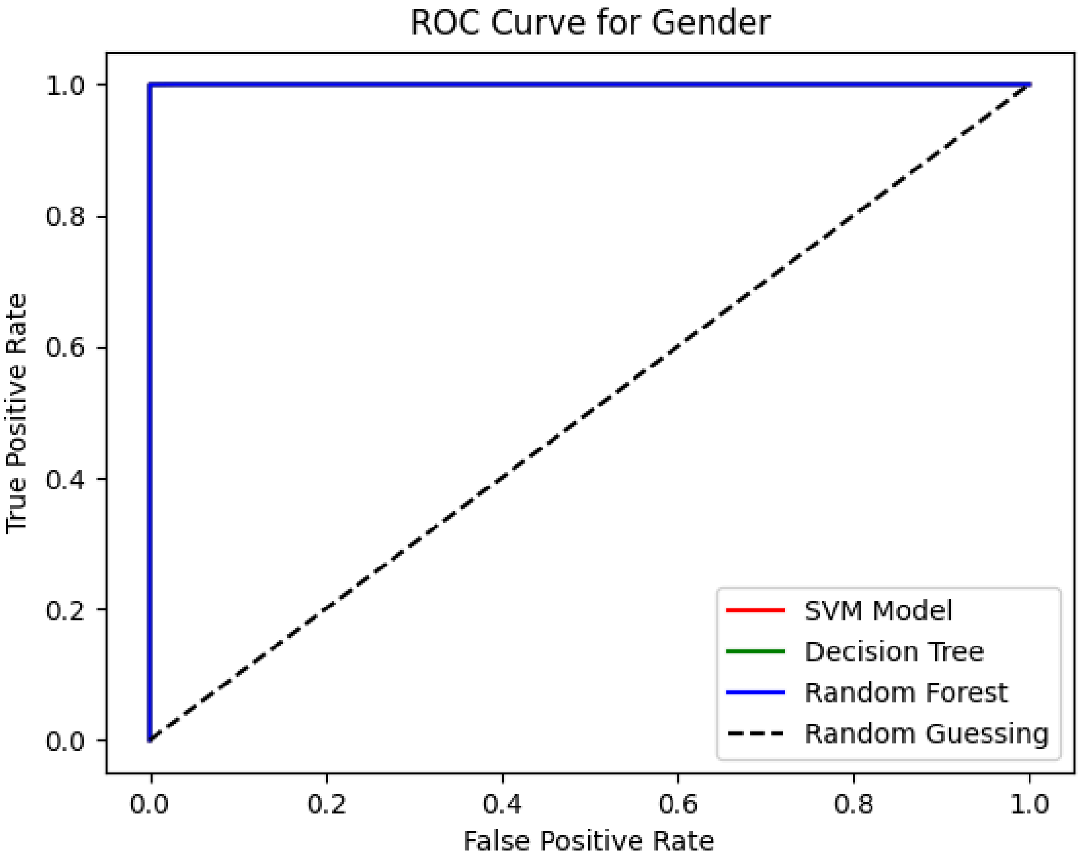
Random Forest Classifier:
For Dataset 2, a Random Forest model was made to see if categorical data indicated gender or if the person was in treatment. To split the data into test and train sets, I used an equal split of males and females. For predicting gender, due to the similarity of male and female data, the output was largely female.

For gender, the model performed as follows:

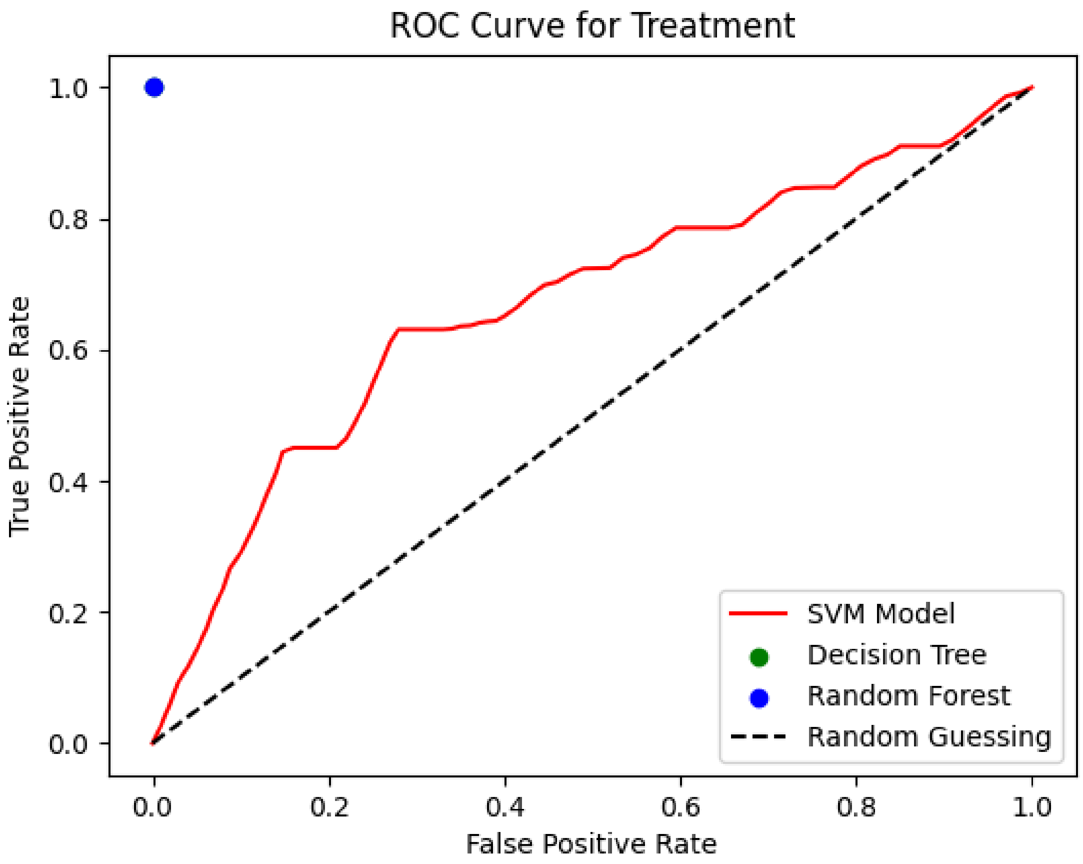
Accuracy: 0.55
Precision: 0.52
f1score: 0.67
recallscore: 0.91

For treatment, the model perfectly classified all the test data.

Accuracy: 1.0
Precision: 1.0
f1score: 1.0
recallscore: 1.0



The ROC curve for gender is an artifact of the way the data is ordered. The data is 3000 Females followed by 3000 males. Assuming the model predicted ~100% females (with some wiggle room based on the accuracy score), this is consistent, as it would be correct for the first 3000, hence the true positive rate increasing vertically, and then wrong for the next 3000, hence the horizontal line across False positive. In order for this ROC to make sense, I inverted the axis, but this could be an error.



The ROC curve for treatment makes complete sense with the scores reported. The SVM model performed with an accuracy score of 0.771125, which would be above random guessing by a decent margin, as shown in red. Both the tree based models gave completely perfect classifications, which would make sense as their True positive rate remains at 1. A perfect model would be a dot in the upper lefthand corner,which is consistent with the ROC curve. In order to make the ROC curve make sense, I had to invert the x and y axis, which could be an error. The scores make sense.

Dataset 3

For all the models listed below, they are trained and contained in data_3_models.py.

SVM:
For Dataset 3, two SVM models were trained, one for Depression and one for Gender.
To split the data into test and train sets, I used 60% of the dataset for train and 40% for test.

For Depression, the scores output are as follows:

Accuracy: 0.90
Precision: 1.0
f1score: 0.81
recallscore: 0.69

For Gender, the scores output are as follows:

Accuracy: 0.90
Precision: 0.88
f1score: 0.94
recallscore: 1.0

Decision Tree:
For Dataset 3, two Decision Tree models were trained, one for Depression and one for Gender.
To split the data into test and train sets, I used 60% of the dataset for train and 40% for test.

For Depression, the scores are as follows:

Accuracy: 0.90
Precision: 0.90
f1score: 0.83
recallscore: 0.76

For Gender, the scores are as follows:

Accuracy: 1.0
Precision: 1.0
f1score: 1.0
recallscore: 1.0

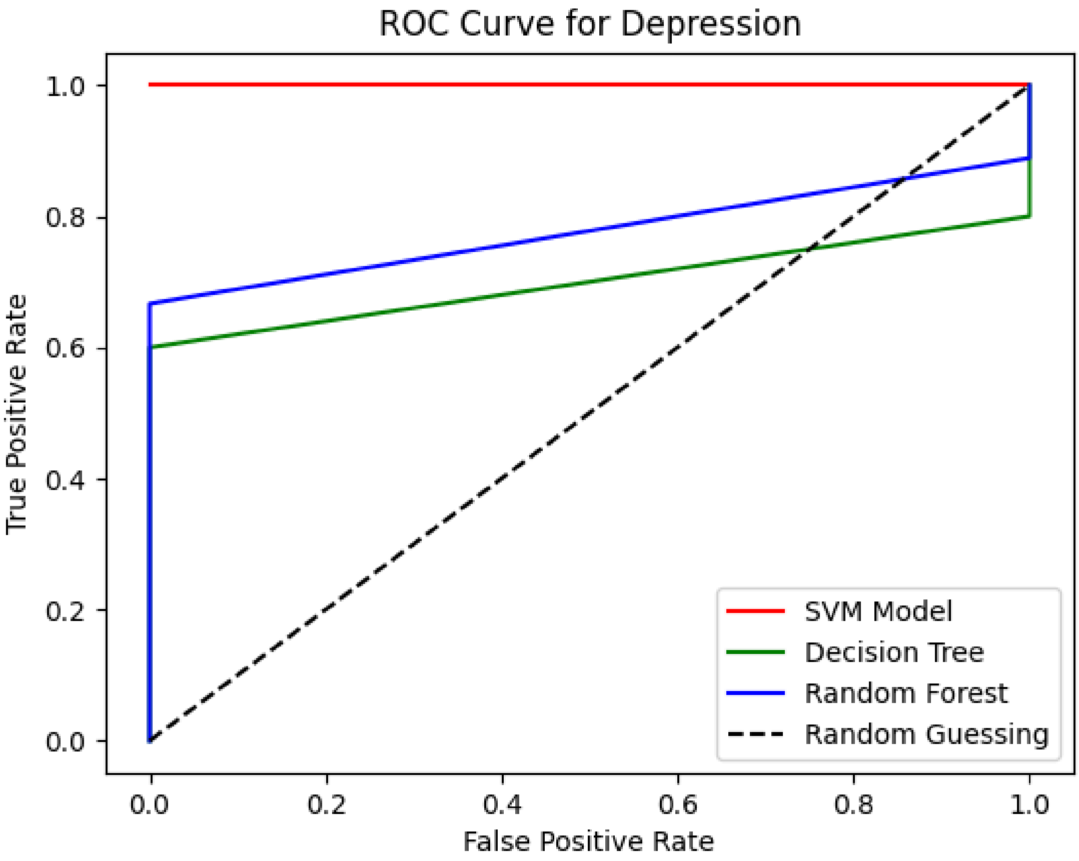
Random Forest:
For Dataset 3, two Random Forest models were trained, one for Depression and one for Gender.
To split the data into test and train sets, I used 60% of the dataset for train and 40% for test.

For Depression, the scores are as follows:

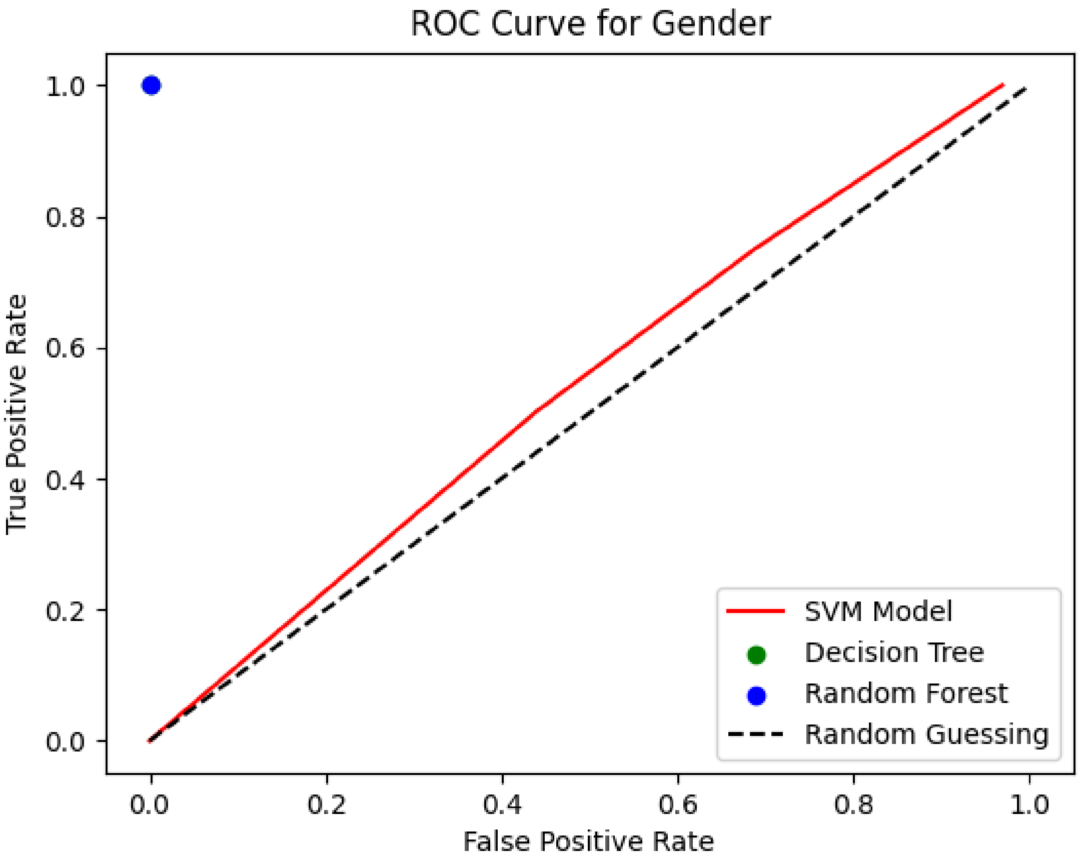
Accuracy: 0.87
Precision: 0.9
f1score: 0.78
recallscore: 0.69

For Gender, the scores are as follows:

Accuracy: 1.0
Precision: 1.0
f1score: 1.0
recallscore: 1.0



The ROC curve is consistent with the scores reported above. The SVM model had an accuracy score of 0.90, which is consistent with a high performance. The decision tree and random forest also performed with an accuracy score of .90 and .87 respectively. The axis on this ROC curve made sense non-inverted.



The ROC curve is consistent with the scores reported above. The SVM model had an accuracy score of 0.90, which is consistent with a high performance. The decision tree and random forest also performed perfectly. The x and y axis were inverted to make this ROC curve make sense, which could be an error.

Dataset 1

For this dataset, due to the continuous nature of the dataset, simple linear regression models were used. To convert Decade into a usable form, one hot encoding was used.

All models for dataset 1 are contained in data_1_models.py

20 linear regression models were created, with the x input for each being a one-hot encoding of decade. The first 5 were models of the percent of population affected by mental illness over the years 1990 - 2019 across world countries. The mean absolute error and mean squared error for the 5 models are below. These 5 models performed well with minimal error.

Schizophrenia percent

Mean Absolute Error 0.030
Mean Squared Error 0.0017

Depressive percent

Mean Absolute Error 0.78
Mean Squared Error 0.93

Anxiety percent

Mean Absolute Error 0.77
Mean Squared Error 1.17

Bipolar percent percent

Mean Absolute Error 0.21
Mean Squared Error 0.065

Eating Disorder percent

Mean Absolute Error 0.10
Mean Squared Error 0.015

The next 5 were models of the DALYs of population affected by mental illness over the years 1990 - 2019 across world countries. The mean absolute error and mean squared error for the 5 models are below. The models performed poorly, with significant error.

Schizophrenia DALYs

Mean Absolute Error 21.22
Mean Squared Error 846.91

Depressive DALYs

Mean Absolute Error 149.02
Mean Squared Error 35241.67

Anxiety DALYs

Mean Absolute Error 78.41
Mean Squared Error 11602.19

Bipolar DALYs

Mean Absolute Error 47.36
Mean Squared Error 3147.43

Eating Disorder DALYs

Mean Absolute Error 22.92
Mean Squared Error 760.18

The next 5 were models of the avg DALYs of population affected by mental illness over the years 1990 - 2019 across world countries. The mean absolute error and mean squared error for the 5 models are below. The models all performed well with low error.

Schizophrenia Avg Percent

Mean Absolute Error 0.030
Mean Squared Error 0.0017

Depressive Avg Percent

Mean Absolute Error 0.76
Mean Squared Error 0.90

Anxiety Avg Percent

Mean Absolute Error 0.79
Mean Squared Error 1.22

Bipolar Avg Percent

Mean Absolute Error 0.21
Mean Squared Error 0.066

Eating Disorder Avg Percent

Mean Absolute Error 0.10
Mean Squared Error 0.015

The next 5 were models of the avg percent of population affected by mental illness over the years 1990 - 2019 across world countries. The mean absolute error and mean squared error for the 5 models are below. The models performed poorly with significant error.

Schizophrenia Avg DALYs

Mean Absolute Error 20.22
Mean Squared Error 735.48

Depressive Avg DALYs

Mean Absolute Error 156.73
Mean Squared Error 38054.16

Anxiety Avg DAYLs

Mean Absolute Error 76.036
Mean Squared Error 11154.11

Bipolar Avg DALYs

Mean Absolute Error 46.78
Mean Squared Error 3101.95

Eating Disorder Avg DALYs

Mean Absolute Error 21.78
Mean Squared Error 717.24