My classification model uses sklearn's DecisionTreeClassifier class. To ensure the decision tree was useful, I only used the features: `'Relationship','Sex','Age','Education_num', 'Capital_gain', 'Hours_per_week'`

Performance

The classification model performs reasonably well with an accuracy of 0.82 and a mean square error of 0.18. The confusion matrix is as follows:

```
[[11714   721]
 [ 2209  1637]]
```

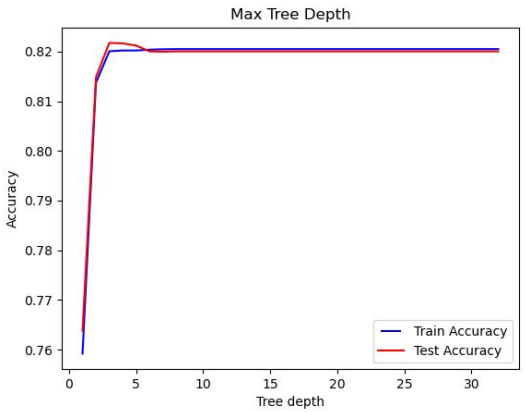
Meaning that the model classified 11714 true negatives, 721 false negatives, 2209 true positives and 1637 false positives.

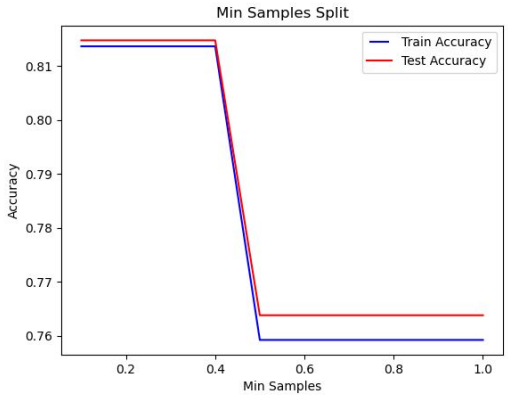
Data Preprocessing

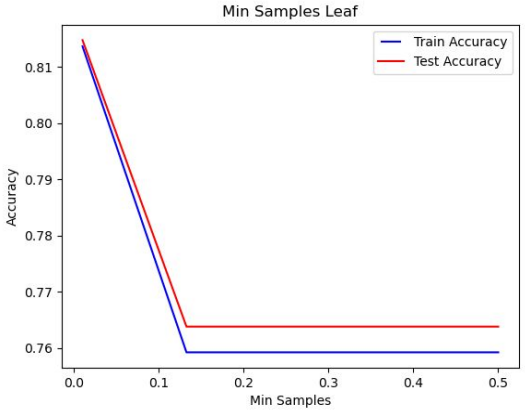
In the data set there were many missing values with the symbol '?'. These values were replaced with either the mean or mode of the feature, depending on whether the values were continuous (numeric) or discrete (categorical). Numeric features were transformed into categorical types by dividing their domain into n bins with the divideIntoNBins function. The number of bins parameter was optimized to give both the best correlation and the simplest decision tree. All categorical features were then converted into numbers so that the decision tree algorithm could process them. This was done using the LabelEncoder class from sklearn's preprocessing library.

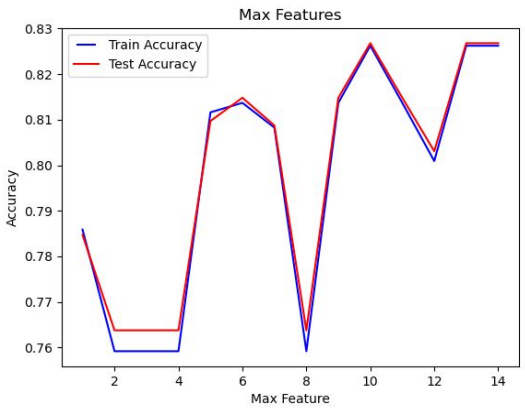
Feature Selection and Optimization

The parameters of the DecisionTreeClassifier class were fine tuned by plotting them against the accuracy score.





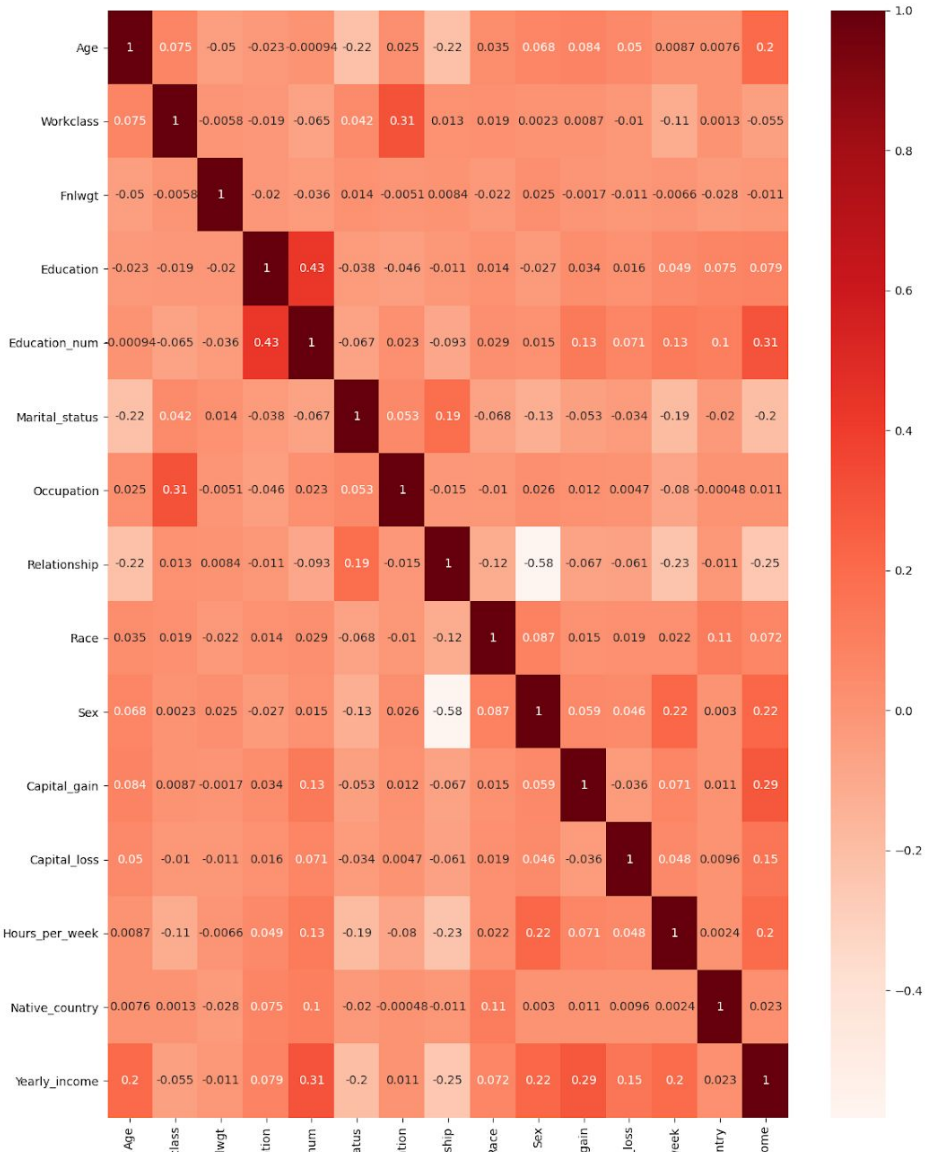




From these observations I chose to set `max_depth=6`, `min_samples_split=0.4`, `min_samples_leaf=0.01` and `max_features=6`.
Fine tuning the n bins parameter for each continuous and numerical feature type was done by trial and error. Optimal parameters yielded a higher correlation with Yearly Income.
The parameter values were as follows:

Feature	Num bins
Education num	4
Age	4
Capital gain	15
Hours per week	8

Features were then selected based on the correlation heatmap in figure below. The 6 best features which were highly correlated with Yearly Income and had fewer discrete values to display on a decision tree were used in the model.



Decision Tree: