Just as note i have changed my project since the progress report since you guy told me my data set was too small

Jordan Smart
Video presentation: https://youtu.be/FmsSPddrAH8
** all visualizations and graph are made and stored in the jupyter notebook

Final Project

1. Project topic: benchmark 5 pts / max 8 pts

My project is focused on using census data to predict if a person's income is greater or less than 50K. So this is a binary classification problem. It's interesting to me to see if we can predict their income in a binary scene from race, age, education and more.

2. Data: 5 pts

LINK: https://archive.ics.uci.edu/ml/datasets/Adult

The data set was obtained from the UCI machine learning repository, it contains data from the 1994 census. There are 15 features(6 numeric features, 9 categorical features)including the income which I am predicting and 32561 samples.

A few feature descriptions:
Income: >50K and <=50K if they make less that 50,000
age: age of the person
workclass: what sector the work in, fed gov or state gov for example
education: highest level of education
marital-status: if they are married, divorced, and so on
occupation: what job they have
race: what race the person is
hours-per-week: hours per week they work
Native-country: what country they originally came from

This data is from one source(1994 census) and not multi-table

3. Data cleaning: benchmark 10 pts / max 15 pts

This data was relatively clean, nor major work was done to get it running through a model. Since a lot of the data was catigoralical i had to use one hot encoding to separate their fields into columns with a binary value of does belong to that category or not. We clean the data set to get a reliable data set with no errors or duplicate data. If we dirty data in the model we will get bad results. I can also identify that the data is imbalanced as we have more data for people we make below 50k than above. In conclusion data is imbalanced but is decently clean.

4. Exploratory Data Analysis (EDA): benchmark 15 pts / max 20 pts

EDA is performed because it gives us good insights of how the data is correlated and what some of the outliers are. After printing box-plots for each feature I found so outliers in capital gain and loss so these were removed. I also found correlation between some features. Transport-moving occupation and an unknown workclass have a very high correlation as well as Transport-moving occupation and Married-civ-spouse. There is around a 0.4 correlation between education level and education num (which make sense because they both are measuring education), race and country of origin(also make sense because if you are from an asian country your race will be asian), and native country of Laos and a low education level. Now there is a very negative correlation between male and female sex, and between race and the other races.

5. Models: benchmark 20 pts / max 30 pts

I chose to try many different classification models, svc, linear svc, sgd,knn, and many decision tree models. After trying all of these I confirmed what I thought, the decision tree classifier i'll work the best. I ran basic models and found adaboost, GradientBoosting, and random forest all performed about the same. So I performed hyperparameter tuning on all of these to find the best results. I found that n_estimators = 460 and min_samples_leaf= 8 is the best fit. I also used F1 scoring with text since the data is imbalanced and F1 is focused on the false negatives and false positives which is the best in this case. I tried feature selection from feature importance after the final best hyper parameters but i didn't get a better f1 score with any combination so i decided to go with the model after hyper parameter. This gave me a f1 score of 0.7178 And a accuracy of 0.8762.

6. Results and Analysis: benchmark 20 pts / max 30 pts

So with the model I have a f1 score of 0.7178 And an accuracy of 0.8762. This is pretty good. I used F1 scoring with text since the data is imbalanced and F1 is focused on the false negatives and false positives which is the best in this case. The tuning process bumped the f1 score from 0.6686 to 0.7177 which definitely increased its accuracy a lot. This was the GradientBoosting model that i was working with but i did have some success with other models as well AdaBoost and RandomForest each of these scoreing about 0.1 and 0.3 less than GradientBoosting. I did try feature selection with very little luck, it seem that i got up to the just under the score of the

GradientBoosting with less features but having all the feature gave me the highest results

7. Discussion and Conclusion: benchmark 10 pts / max 20 pts

So I ended up using GradientBoosting to get the best results, thinking that i could have gotten some better result with trying more with feature selection, for some reason when I did feature selection I did not get any better scores. My takeaway is that yes you can predict income on some sort of level from census data. I think there could be some more work done on the EDA but with limited time and other final i did what i could

** all visualizations and graphs are in the jupyter notebook