# Income Prediction

## Final Capstone

By Edgar Garcia

# Income Prediction

# Why Predict a Person’s Income?

- For this project, my goal  is to create a prediction model that can accurately classify if a person makes less or more than $50,000 a year.
- The US Census bureau collects income data annually in order to analyze it and determine the profiles of people that make a certain amount of money each year.
- This solution could prove valuable because the Income Census bureau may want to better understand the data they have and use it in different ways.
- By analyzing it and creating an accurate classification model, we could use it to determine what are the key factors that determine a person's income amount and can apply this model to newly collected  annual income data.

# The Data

- I will use a data set from Kaggle that was taken from the 1994 Census bureau website.

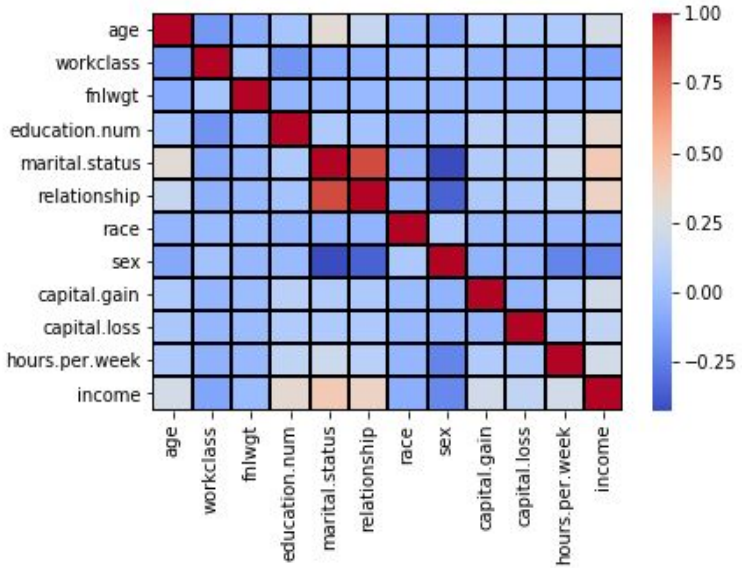
- I first analyzed the data and changed categorical values to numeric. I also dropped some columns such as education.num, native.country, and occupation as they had too many unique categorical values.

- The target variable 'income' will be changed to binary values. 0 will represent <= 50,000 and 1 will represent > 50,0000.

- I changed work.class to numeric values, grouped the marital.status categories by number, and changed all other categorical values to numeric.

# The Data

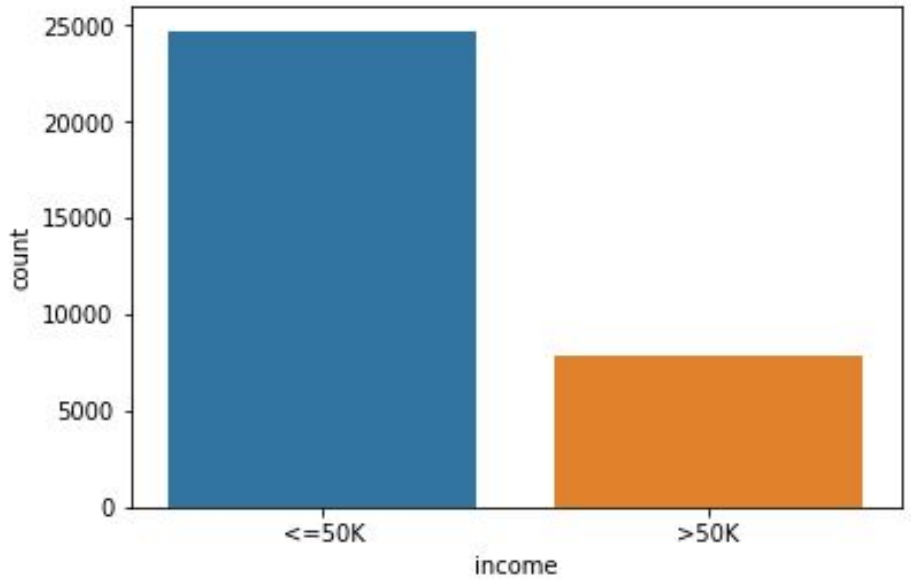
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32561 entries, 0 to 32560
Data columns (total 12 columns):
age               32561 non-null int64
workclass         32561 non-null int32
fnlwgt            32561 non-null int64
education.num     32561 non-null int64
marital.status    32561 non-null int32
relationship      32561 non-null int32
race              32561 non-null int32
sex               32561 non-null int32
capital.gain      32561 non-null int64
capital.loss      32561 non-null int64
hours.per.week    32561 non-null int64
income            32561 non-null int32
dtypes: int32(6), int64(6)
memory usage: 2.2 MB
```



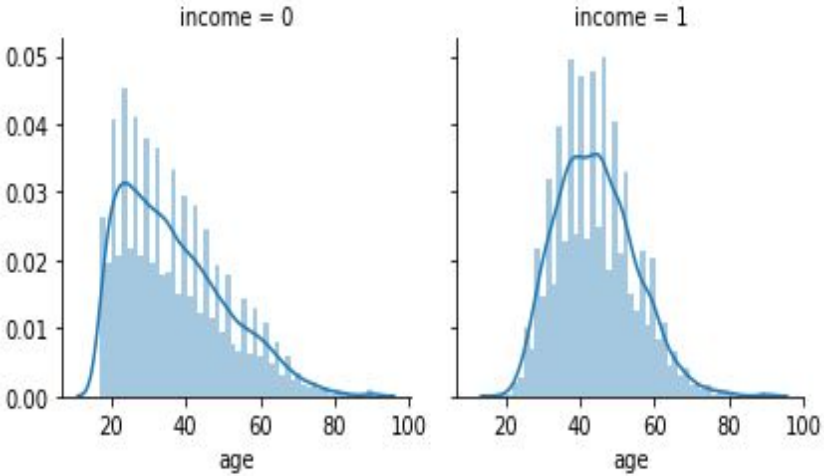

- We can see that values are all numeric.
- From the heatmap above, we can see that there is no strong correlation between the features. Multicollinearity should not be an issue.

# Data Visualizations

Below, we can see that the ratio of people that make less than $50,000 to people that make more that $50,000 is about 7:3.

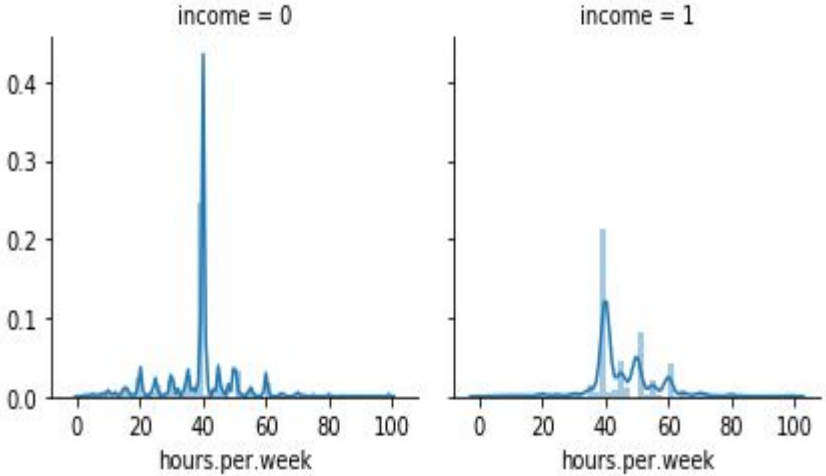


# Data Visualizations

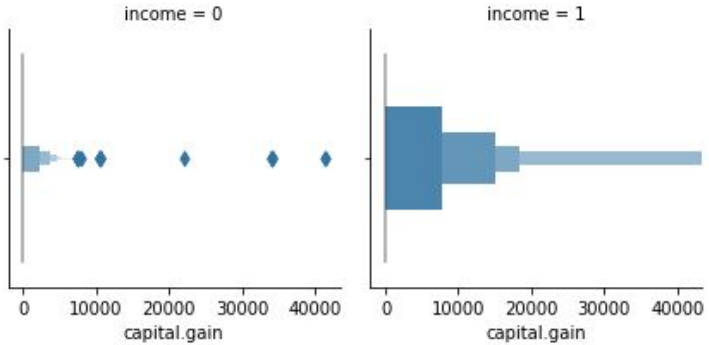



- The distribution for age in relation to income is skewed to the right for people with income below $50,000. (income = 0) and approximately normal for the people with income above $50,000. (income =1)

- The distributions for hours.per.week in relation to income in approximately normal for both groups with the distributions being centered at 40.
- This makes sense since most people work about 40 hours per week.

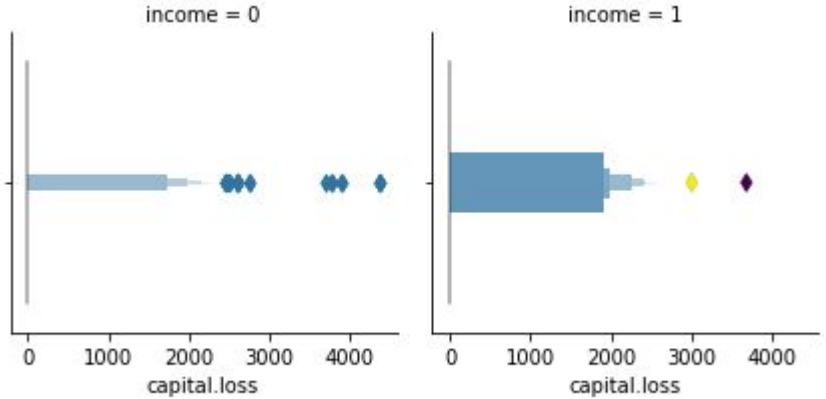


# Data Visualizations

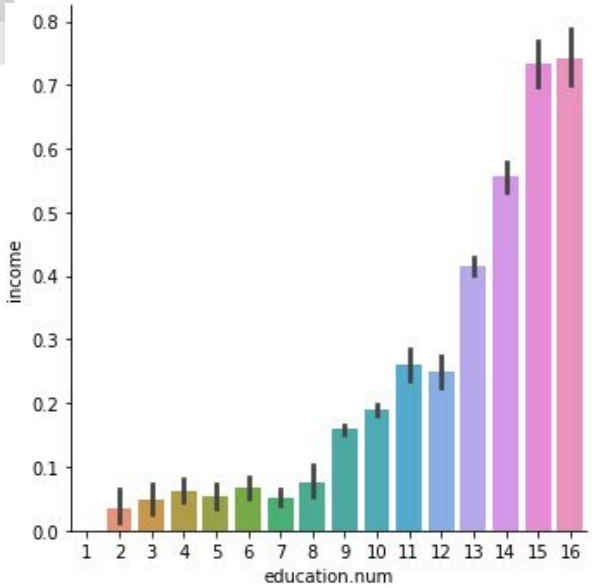



- Above, we can see that people with income less than $50,000 did not have a lot of capital gain.
- People with income amounts over $50,000 had higher capital gain amounts on average.

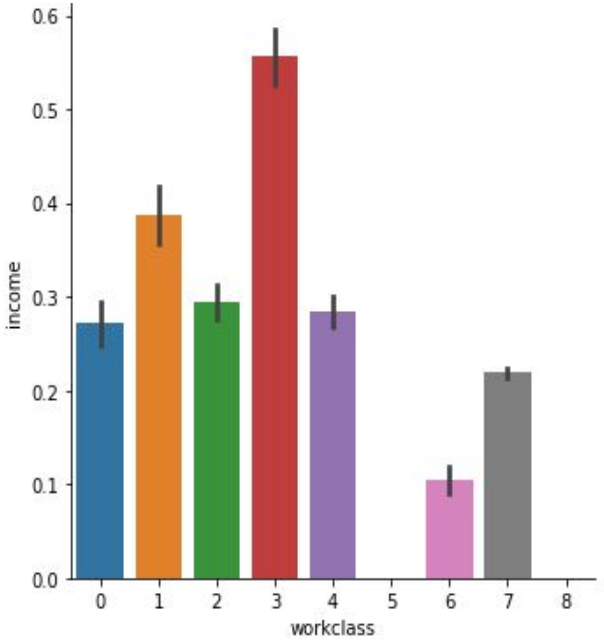
- Below, we can see similarly to capital loss, gain, that there was not a lot of capital loss among people with income amounts less than $50,000.
- People with income amounts above $50,000 had capital loss amounts between 0 and 2000.



# Data Visualizations

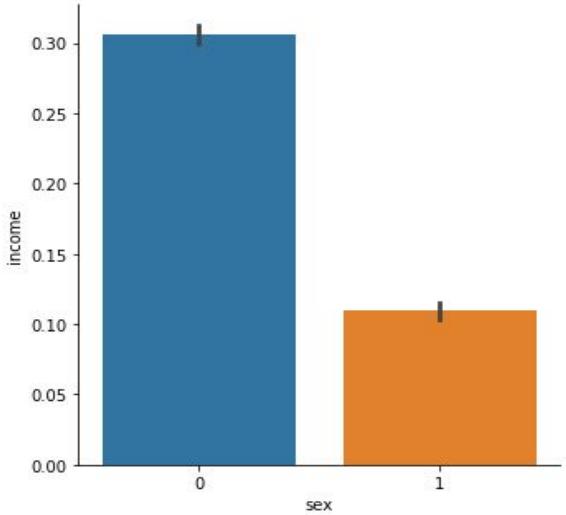



'Preschool': **1**, '1st-4th': **2**, '5th-6th' :**3**, '7th-8th': **4**, '9th': **5**, '10th': **6**, '11th': **7**, '12th': **8**, 'HS-grad': **9**, 'Some-college': **10**, 'Assoc-voc': **11**, 'Assoc-acdm': **12**, 'Bachelors': **13**, 'Masters': **14**, 'Prof-school': **15**, 'Doctorate': **16**



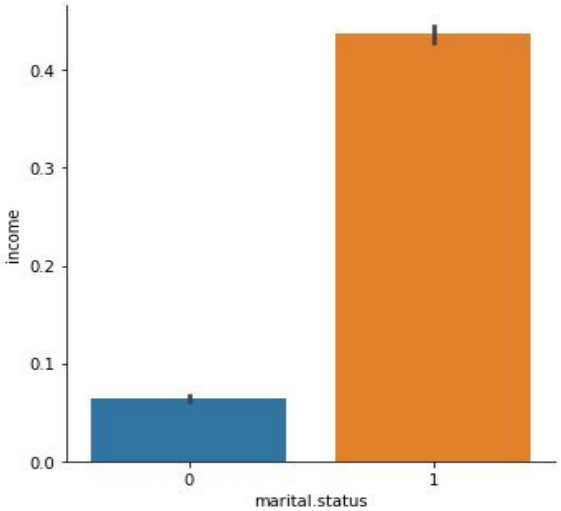

'State-gov': **0**, 'Federal-gov': **1**, 'Local-gov':**2**, 'Self-emp-inc': **3**, 'Self-emp-not-inc': **4**, 'Never-worked': **5**, '?': **6**, 'Private': **7**, 'Without-pay': **8**

# Data Visualizations




- In this bar chart, we can see that sex ( Male:0, Female:1) has a significant on income amount. This bar chart tell us that on average, males have higher income amounts than females.

- In the bar chart below, we can see that single people have a lower income than people that are married.
- This makes sense since people that are married tend to have a shared income amount.



# Prediction Model

- I created six different classification models using Logistic Regression, KNN, Decision Trees, Random Forest, Gradient Boosting, and XG Boosting.
- After comparing the models, the XG boosting model seemed to have performed the best.

In [54]:
```
xgb = XGBClassifier()
xgb.fit(X_train, y_train)
cv = cross_val_score(xgb, X_test, y_test, cv = 5, scoring = 'accuracy')
print("%s: %f (%f)" % ('XG Boosting', cv.mean(), cv.std()))
```

```
XG Boosting: 0.866420 (0.011390)
```

In [55]:
```
print("Accuracy: %s%%" % (100*accuracy_score(y_test, xgb.predict(X_test))))
print(confusion_matrix(y_test, xgb.predict(X_test)))
print(classification_report(y_test, xgb.predict(X_test)))
```

```
Accuracy: 86.8724090281%
[[4708  238]
 [ 617  950]]
             precision    recall  f1-score   support

          0       0.88      0.95      0.92      4946
          1       0.80      0.61      0.69      1567

avg / total       0.86      0.87      0.86      6513
```

# Other Models

## Keras NN Model

```
model = Sequential()

model.add(Dense(64, input_dim=11, init='uniform', activation='relu'))
model.add(Dense(1, init='uniform', activation='relu'))
model.add(Dense(1, init='uniform', activation='sigmoid'))
```

```
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
```

```
model.fit(X_train, y_train, epochs=20, batch_size=10, validation_data= (X_test, y_test) )
```

```
scores = model.evaluate(X_test, y_test)
print("\n%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))
```

```
6513/6513 [==============================] - 0s 36us/step

acc: 75.94%
```

# Other Models

## Random Forest Classfier

```
rf = RandomForestClassifier(n_estimators= 100, max_features= 10)
rf.fit(X_train, y_train)
cv = cross_val_score(rf, X_test, y_test, cv = 5, scoring = 'accuracy')

print("%s: %f (%f)" % ('Random Forest', cv.mean(), cv.std()))
```

```
Random Forest: 0.839400 (0.007349)
```

## Gradient Boosting Classifier

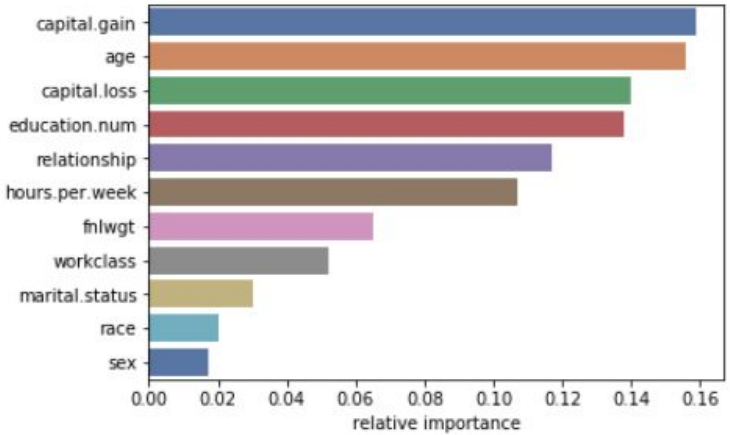
```
gb = GradientBoostingClassifier()
gb.fit(X_train, y_train)
cv = cross_val_score(gb, X_test, y_test, cv = 5, scoring = 'accuracy')
print("%s: %f (%f)" % ('Gradient Boosting', cv.mean(), cv.std()))
```

```
Gradient Boosting: 0.860125 (0.005157)
```

# Most Important Features

- Among the 11 features, it seems that the top three most important features were capital.gain, education.num, and capital.loss.
- This makes sense as capital gains and losses can affect one's annual income. Education is also an important factor as it leads to having a better job.

```
Best Model is XG Boosting
==============================
Feature rank among 11 features:
capital.gain      0.170
education.num     0.148
capital.loss      0.147
age               0.141
relationship      0.126
hours.per.week    0.095
fnlwgt            0.066
workclass         0.046
marital.status    0.024
sex               0.020
race              0.016
dtype: float64
```



# Conclusion

- After comparing different income prediction models, it seems that best model was the XG boosting model. This model had an accuracy score of 86% and a recall score of 87%.

- Other models that were used such as logistic regression, KNN, decision trees, random forest, decision trees, and Keras Neural Network performed well, but did not perform as well as the XG boosting model.

- Looking at our features, the top 3 most important features were capital.loss, capital.gain, and education. So it seems that capital, education, age and hours per week are the most significant factors.

- This capstone project helped me to further understand to explore data, implement feature engineering, and create different types of prediction models..

# Next Steps

- Even though the accuracy score of my model was 86%, I would still like to improve upon it. Perhaps I could create additional features in order to raise the accuracy score.

- I would like to conduct more research on the features in order to learn about them and understand why they are so important and how the Income Census bureau can use this data to benefit them.

- Also, I would like to test my model on a similar data set from a different year to see how it performs on a different data set.

- I believe this model could prove useful for the Census bureau in classifying future income data.