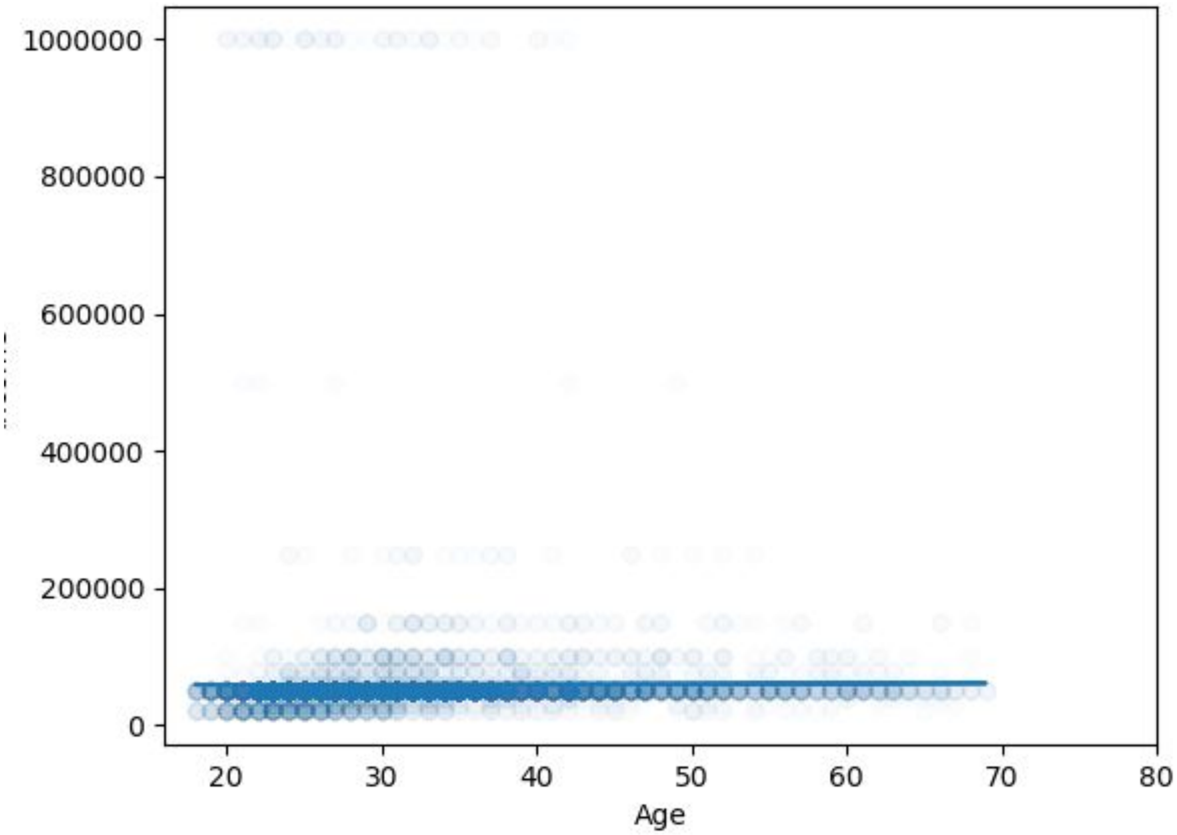
OKCupid: Predicting Income based on other factors

Machine Learning Fundamentals
Eric Stebbins
11/12/2018

Hypothesis: Age vs. Income

One would expect (or at least hope) that as one ages, their income would increase.

In order to test out this hypothesis, I first set out to determine if there was a positive correlation between one’s age and their stated income. I got some surprising results.



In this graph, there does not seem to be much of a correlation between age and income.

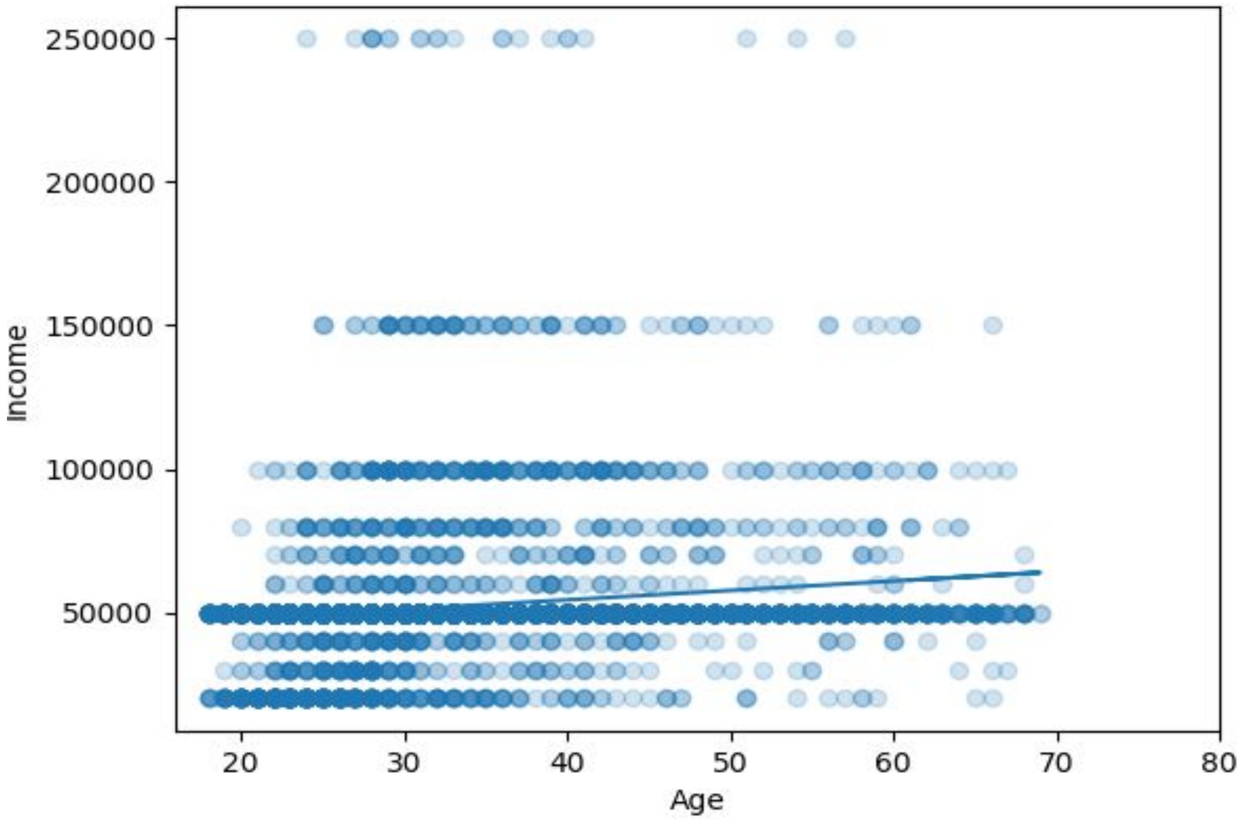
However, I suspect that there may be some issues with the data.

What percentage of 20-40 year-olds make a million dollars per annum?

Issues with that graph:

- Even granted that OKCupid users are not representative of the general population I suspect that the percentage of people that make 1,000,000/an. is less than 0.9%, which is roughly what the data would suggest : 349 / 38688 rows left in our data = 0.00902.
- Getting rid of outlying values will hopefully get rid of this problem. Looking at the data, I think that the income values above $250,000 are a) suspect and b) statistically significant.
- Problem 2: There are many -1 responses in the data for income indicating that people did not respond. Getting rid of these would give us an unacceptably small sample. Solution: set them to the median result.
- Changing these values produces the graph we will see on the next page

Final Age vs. Income Plot



Results of Excluding Outliers:

- The data is no longer squashed close to the x axis and the graph is more readable due to the elimination of the majority of the outliers.
- We are now able to see a positive correlation between age and income in this data.
- The training and testing scores for our simple linear regression model are 0.024 and 0.027, respectively.
- Though there is clearly a positive linear relationship between age and income, it’s not very accurate at all. Perhaps we should add more features and run multiple linear regression.

Multiple Linear Regression

Added features:

- Estimated years of education
- Drinking level
- Tobacco smoking level
- Whether or not they use drugs

Notes:

Years of education were estimated based on the multiple choice response. Mappings were not scientifically exact and were a judgement call.

Drinking level was ranked on a 0-5 scale based on responses (see data/code).

Tobacco smoking level was ranked similarly to drinking.

Drug use was converted to binary yes/no data based on the responses available.

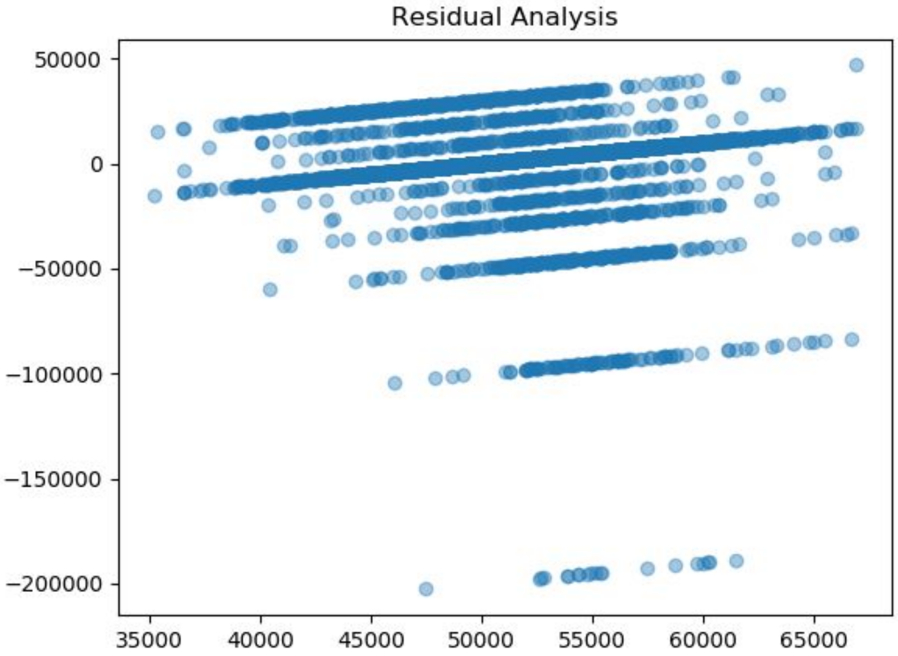
Multiple Linear Regression (MLR) Results:

The MLR training score would suggest that we’ve gotten quite a bit better about our prediction: 0.0484 vs. 0.024 based on age alone.

The MLR test score seems to back that up with 0.0562 vs. 0.027.

Conclusion: the MLR that I created is slightly better than the single variable linear regression, but still pitiful at best.

Residual Analysis Graph and Interpretation



It’s interesting to note the bands that are in this graph. If they seem to reflect the slope of the predicted line on the previous plot, it’s because they do, and the bands exist because there are few discrete values in the responses on income.

If we had more continuous data reflecting actual income value, this would look more like a normal scatter plot.

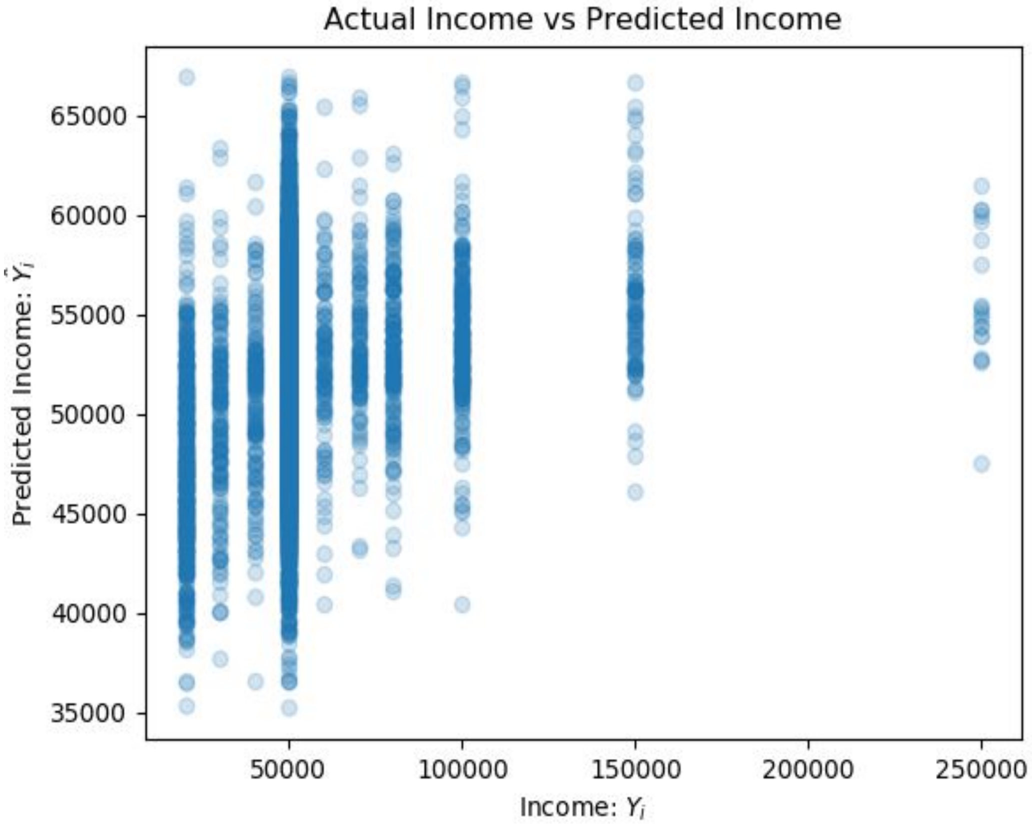
Actual vs. Predicted Income

This graph illuminates the same issue with the data, though this time the bands are vertical rather than following the slope of the linear prediction.

The x axis indicates the actual value for income provided in the dataset, while the y axis indicates the predicted values.

Again this is due to the discrete values for actual data that are available to us.

The main takeaway for me is that it does still seem to indicate a weak but positive correlation of our actual and predicted datasets.



Classification Instead of Regression

Since our actual income data is non-continuous, perhaps we would be better off classifying the income labels and predictions into a series of brackets.

Brackets:

- $0-40000
- $40001-70000
- $70001-100000
- 100001+

K-Nearest Neighbors (KNN) Classifier

I created a KNN classifier model based on the same test data as the multiple linear regression.

KNN Classifier test score results: 0.8078110320762314

Wow, that’s good right? Actually I suspect that it’s because of the preponderance of responses of $50,000. This would cause overfitting to the model.

Time to run the KNN classifier: 0.14462 seconds

Support Vector Machine (SVM) Classifier

I also created an SVM classifier model based on the same test data as the multiple linear regression.

KNN Classifier test score results: 0.8201583680042948

That’s also really good right? It’s because of the same issue as the KNN classifier: overfitting to the model.

Time to run the SVM classifier: 36.63973586344059 seconds

Wow, why the difference in processing time?

SVM takes the individual distances between each point in the dataset which becomes a quadratic problem.

From the sklearn documentation: The fit time complexity is more than quadratic with the number of samples which makes it hard to scale to dataset with more than a couple of 10000 samples.

We are running with 38688 rows, so this would explain the long processing time.

Conclusion:

Unfortunately, we were not able to come up with a very valid predictor for income, other than to say with 80% confidence that someone is likely to land in the 70K - 100K income bracket.

We would likely get some better results if we had continuous income data (i.e. not a multiple choice response) and a more balanced distribution between the brackets we invented to begin with.

We could potentially sculpt our training dataset more to reflect an even distribution between these values, but that would restrict us to a much smaller training set.

We need a more accurate (as far as income) and bigger training sample.