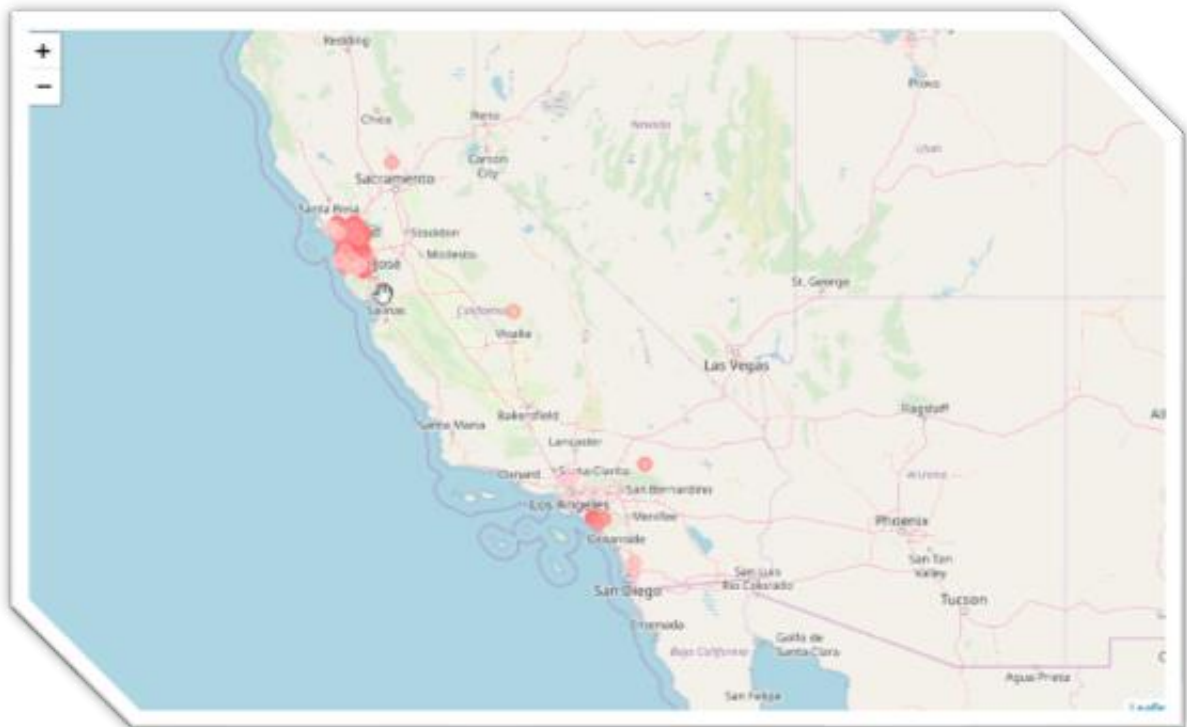


# Machine Learning Fundamentals

DATE A SCIENTIST FINAL

Anthony Graves | Machine Learning with Python | 1-3-2019



## **Table of Contents**

1. Does income play a role in how a dating profile is viewed by the opposite sex?
2. Geolocation columns classify and pinpoint where the dating profiles are located
3. Pairwise Plot data visualization
4. Correlation matrix between quantifiable features of the dataset
5. Distribution of reported and non-reported income
6. Logistic Regression, Precision, Accuracy, ROC Curve, Logit Regression results, P-values and K-nearest values
7. Conclusion
8. Next steps

## **1.Does income play a role in how a dating profile is viewed by the opposite sex?**

The fact that most people do not report their income is unsurprising given its private nature and our culture (in many cultures people are much more open about their income), but it is also quite intriguing to consider whether other quantifiable factors are related to the reporting of income or not, especially after viewing the many correlations with reported income that exist in the data.

So, my research question came to be:

Can reported income be accurately classified and what are the factors that contribute to whether it was reported or not?

Unfortunately, the dataset is entirely self-reported, so our confidence in the results in any question posed will be undermined by self-reporting bias, but there is nothing to be done about that.

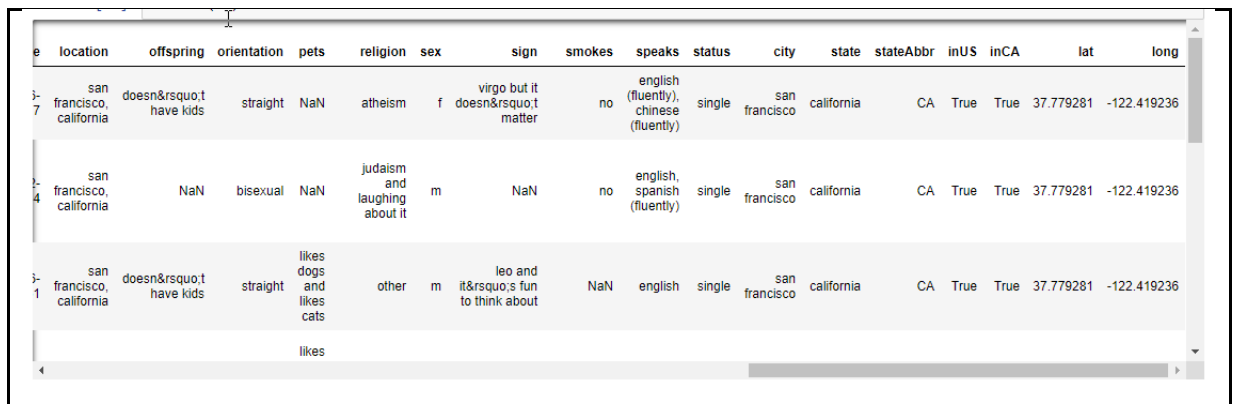
Since we are required to not only do classification but regression, I have decided to use KNN and SVM for classification, and logistic regression and multiple linear regression for regression.

Since the data does not need to be normalized for logistic or multiple linear regression and interpretation will be clearer without normalization, we will start there.

## 2. Geolocation data columns: State Abbrev, InUS, Latitude and Longitude

I wanted to add geolocation data for geo-mapping, but along the way found out that almost all the data was in California. Because of this I just got geolocation data for California cities.

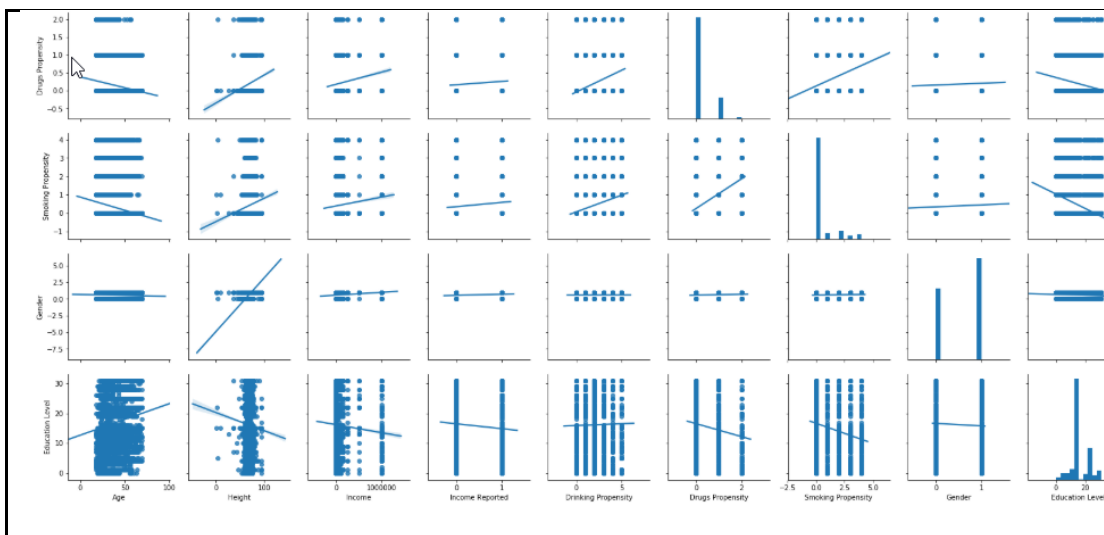
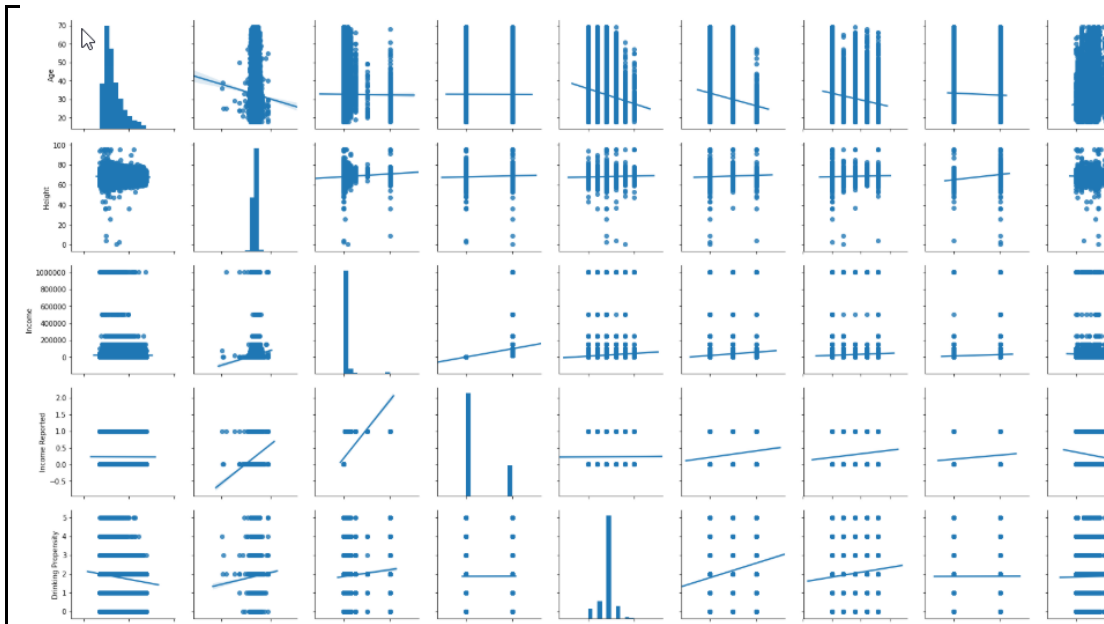
It wasn't possible using the api I found to get geolocation data for all cities, but I was unable to for almost all of them.

A screenshot of a data table with 18 columns. The columns are: 'e', 'location', 'offspring', 'orientation', 'pets', 'religion', 'sex', 'sign', 'smokes', 'speaks', 'status', 'city', 'state', 'stateAbbr', 'inUS', 'inCA', 'lat', and 'long'. The table contains three rows of data, all for 'san francisco, california'. The first row has values: '7', 'san francisco, california', 'doesn't have kids', 'straight', 'NaN', 'atheism', 'f', 'virgo but it doesn't matter', 'no', 'english (fluently), chinese (fluently)', 'single', 'san francisco', 'california', 'CA', 'True', 'True', '37.779281', '-122.419236'. The second row has values: '4', 'san francisco, california', 'NaN', 'bisexual', 'NaN', 'judaism and laughing about it', 'm', 'NaN', 'no', 'english, spanish (fluently)', 'single', 'san francisco', 'california', 'CA', 'True', 'True', '37.779281', '-122.419236'. The third row has values: '1', 'san francisco, california', 'doesn't have kids', 'straight', 'likes dogs and likes cats', 'other', 'm', 'leo and it's fun to think about', 'NaN', 'english', 'single', 'san francisco', 'california', 'CA', 'True', 'True', '37.779281', '-122.419236'. There is a fourth row with a single value 'likes' in the 'pets' column. The table has a scrollbar on the right side.

e	location	offspring	orientation	pets	religion	sex	sign	smokes	speaks	status	city	state	stateAbbr	inUS	inCA	lat	long
7	san francisco, california	doesn't have kids	straight	NaN	atheism	f	virgo but it doesn't matter	no	english (fluently), chinese (fluently)	single	san francisco	california	CA	True	True	37.779281	-122.419236
4	san francisco, california	NaN	bisexual	NaN	judaism and laughing about it	m	NaN	no	english, spanish (fluently)	single	san francisco	california	CA	True	True	37.779281	-122.419236
1	san francisco, california	doesn't have kids	straight	likes dogs and likes cats	other	m	leo and it's fun to think about	NaN	english	single	san francisco	california	CA	True	True	37.779281	-122.419236
				likes													

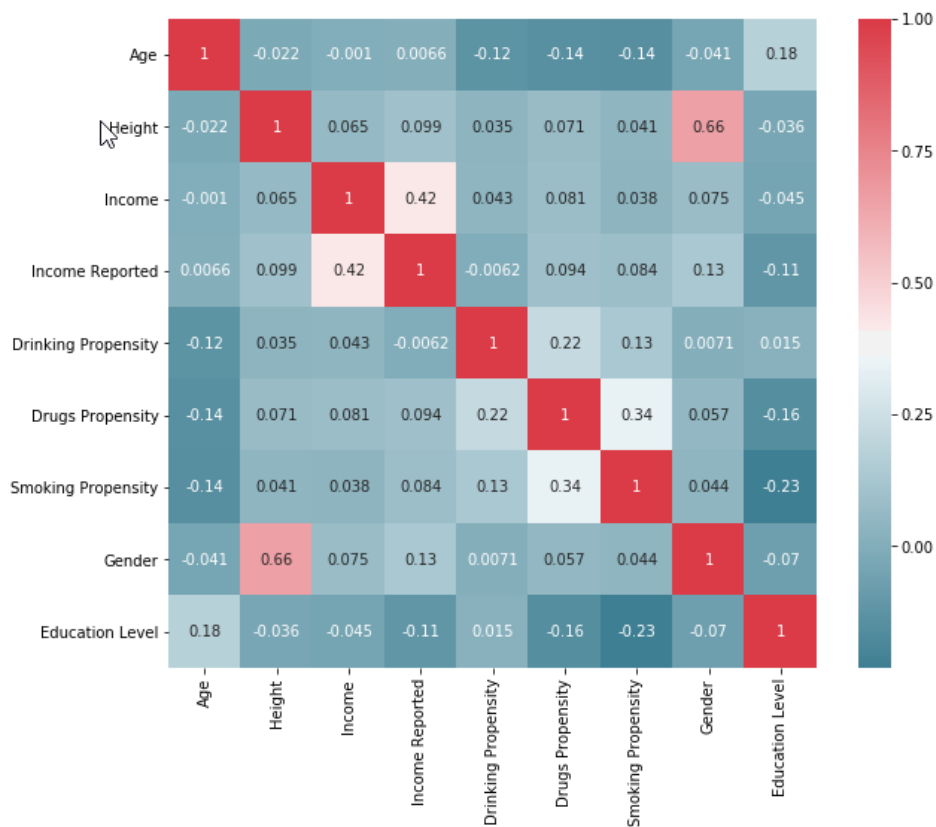
Figure 1: Table shows the addition of latitude and longitude columns for geolocation

### 3. Pairwise Plot Data Visualization



In summary we see some positive correlations as well as some negative correlations.

#### 4. Correlation matrix between quantifiable features of the dataset



Only a few obvious correlations stand out – education and age, drug use and education (lower with higher education), smoking and education (lower with higher education), and gender and height (obviously)

Some weaker correlations are gender and income, height and income, and drug use and smoking for males, as well as drug use and height.

And then there are some correlations between reported income and drug use, smoking, height, gender, and education! None of them are particularly strong, but there seem to be quite a few things correlated with reported income.

Overall, the time to run each model was relatively fast except the SVM model.

## 5. Distribution of reported and non-reported income approaches

### a. Logistic Regression

Logistic regression has the benefit of being very simple to perform, so we will start there.

Confusion matrix for Logistic Regression

	0	1
True label 0	5919	43
1	1747	29
Predicted label		

Precision was: 0.4027777777777778

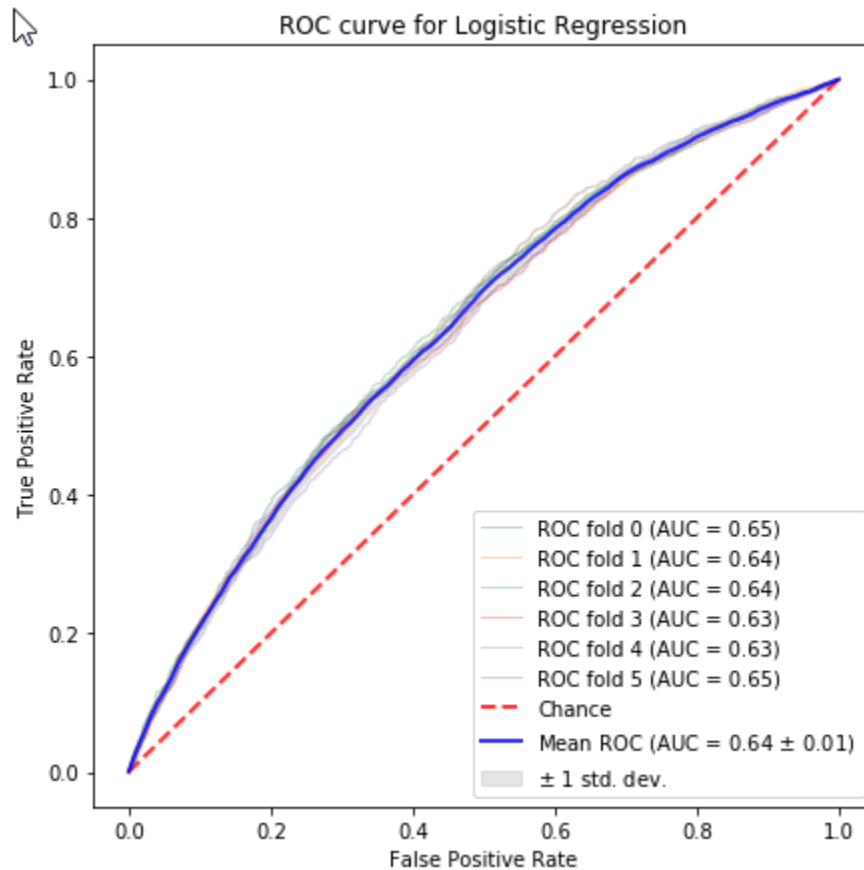
Accuracy was: 0.7686740759886276

Recall was: 0.01632882882882883

The model is accurate for the very few observations it classified as reported, but it chose to guess unreported the vast majority of cases, so it is extremely imprecise.

## b. ROC Curve

For a lot more information, let's look at the ROC curve:



**Figure: shows the model does have some predictive power! Its not minimal which is to be expected given what we are trying to predict.**



## c. Logit Regression Results

Out[186]:  Logit Regression Results

Dep. Variable:	Income Reported	No. Observations:	38688			
Model:	Logit	Df Residuals:	38681			
Method:	MLE	Df Model:	6			
Date:	Wed, 02 Jan 2019	Pseudo R-squ.:	0.03725			
Time:	12:08:40	Log-Likelihood:	-19988.			
converged:	True	LL-Null:	-20761.			
		LLR p-value:	0.000			
	coef	std err	z	P> z	[0.025	0.975]
Age	0.0074	0.001	5.776	0.000	0.005	0.010
Height	-0.0189	0.001	-19.017	0.000	-0.021	-0.017
Drinking Propensity	-0.0547	0.017	-3.231	0.001	-0.088	-0.021
Drugs Propensity	0.3371	0.030	11.146	0.000	0.278	0.396
Smoking Propensity	0.1141	0.012	9.165	0.000	0.090	0.138
Gender	0.7684	0.029	26.902	0.000	0.712	0.824
Education Level	-0.0426	0.002	-18.636	0.000	-0.047	-0.038

From the actual regression, we can see that the p-values for every included feature (all numerical features in general) are all highly significant -p values all less than 0.001 indicate that every feature influence whether someone reported income or not.

The magnitude is strongest for education level (since the scale of education is 0-31, whereas the scale for Drugs propensity is only 0-2. If you normalize according to this then you will see that education has a greater effect than drugs), followed by gender and drug use, and then height, then age, then smoking.

It turns out that this is an interesting research question with predictors that are all significant, but where actual prediction of the outcome is not easy under logistic regression (probably under all models, but we shall see...)

## d. Multiple Linear Regression

Let's compare old ordinary least squares multiple linear regression and if logistic regression does a better job or not.

The commentary above explains a lot of what was done for logistic regression, and much of it will not be repeated here.

```
from sklearn.linear_model import LinearRegression  
  
linear = LinearRegression()  
showConfusionMatrixAndMore(linear, 'Linear Regression', X, y)
```

Confusion matrix for Linear Regression

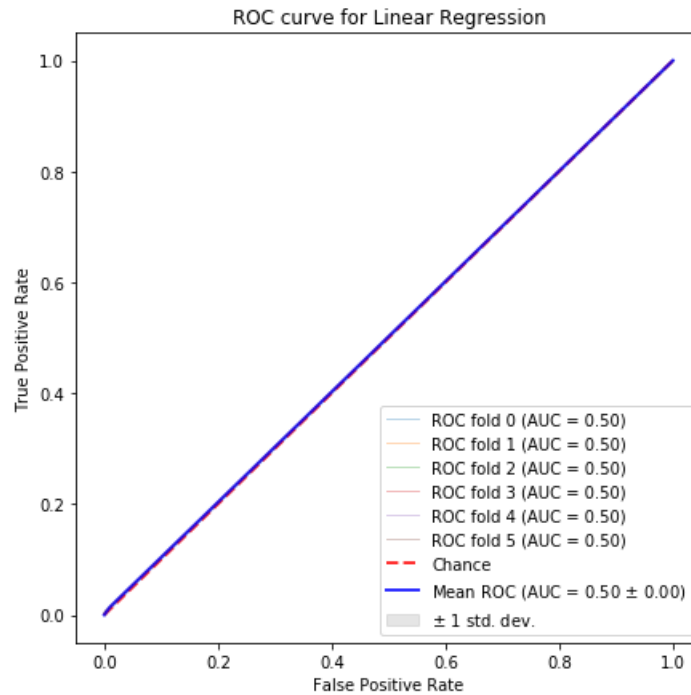
	0	1
True label 0	5950	12
1	1765	11
Predicted label		

Precision was: 0.4782608695652174  
Accuracy was: 0.7703540966658051  
Recall was: 0.006193693693693694

Both linear and logistic regression make too few guesses for income reported to make any meaningful commentary about the precision of each model. Linear regression has a higher number, but a lower amount of positive predictions.

## e. ROC Curve for Linear Regression

```
showROC(linear, 'Linear Regression', X,y)
```



I guess that means multiple linear regression is horrible as a classifier! Luckily, we're not doing classification, but regression. Let's see how it did on that front.

## f. OLS Regression Results

```
In [191]: import statsmodels.discrete.discrete_model as sm  
  
linear_reg = sm.OLS(y, X)  
linear_reg.fit().summary()
```

Out[191]: OLS Regression Results

Dep. Variable:	Income Reported	R-squared:	0.260			
Model:	OLS	Adj. R-squared:	0.260			
Method:	Least Squares	F-statistic:	1943.			
Date:	Wed, 02 Jan 2019	Prob (F-statistic):	0.00			
Time:	12:09:12	Log-Likelihood:	-20453.			
No. Observations:	38688	AIC:	4.092e+04			
Df Residuals:	38681	BIC:	4.098e+04			
Df Model:	7					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
Age	0.0016	0.000	7.337	0.000	0.001	0.002
Height	0.0033	0.000	19.373	0.000	0.003	0.004
Drinking Propensity	-0.0073	0.003	-2.483	0.013	-0.013	-0.002
Drugs Propensity	0.0624	0.006	11.315	0.000	0.052	0.073
Smoking Propensity	0.0241	0.002	10.402	0.000	0.020	0.029
Gender	0.0953	0.005	20.866	0.000	0.086	0.104
Education Level	-0.0068	0.000	-18.049	0.000	-0.008	-0.006
Omnibus:	6257.150	Durbin-Watson:	1.993			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	9670.447			
Skew:	1.216	Prob(JB):	0.00			
Kurtosis:	2.718	Cond. No.	208.			

Next up, KNN, Our first 100% classification approach!

But first, we need to scale all the predictors because the following models need to be normalized to work optimally.

We can see the importance and significance of each feature take on about the same values as in the logistic regression. Remember, the coefficients on OLS are different than the odds ratios in logistic regression. Nonetheless, the ratios between the different features, and their direction of prediction, are all about the same.

This means that regular OLS is good at looking for the predictive features of whether income was reported or not but does a horrible job at classification.

## g.K-Nearest Neighbors

Next up, KNN. Our first 100% classification approach!

But first, we need to scale all the predictors because the following models need to be normalized to work optimally.

```
normalized_X=(X-X.mean())/X.std()
```

```
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors=2)
showConfusionMatrixAndMore(knn, 'K-Nearest Neighbors', normalized_X, y)
```

Confusion matrix for K-Nearest Neighbors

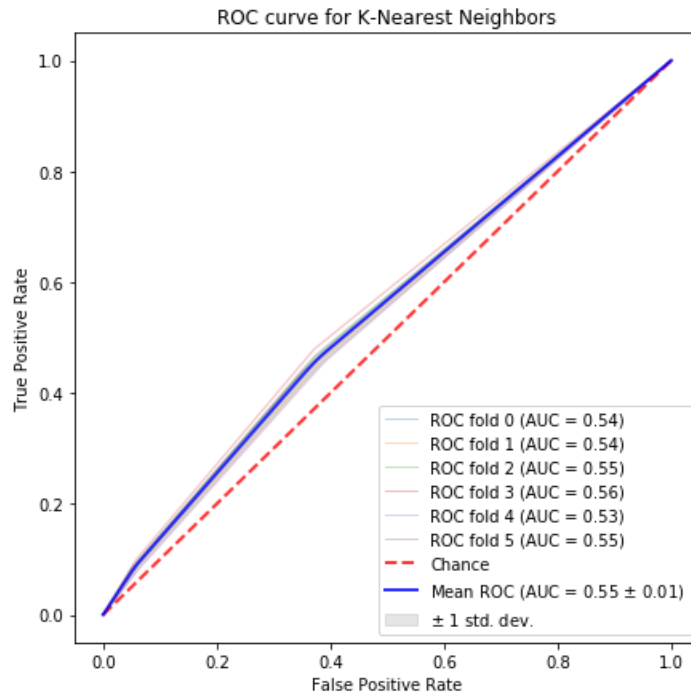
	0	1
True label 0	5653	309
1	1653	123
Predicted label		

Precision was: 0.2847222222222222  
Accuracy was: 0.7464461101059705  
Recall was: 0.06925675675675676

Precision is down from the regression models, but the amount of positive predictions is higher, so the models cannot be directly compared.

## g. ROC curve for K-Nearest Neighbors

```
showROC(knn, 'K-Nearest Neighbors', normalized_X,y)
```



K-nearest Neighbors does have some predictive power, but it seems like logistic regression is in the lead. It's unsurprising given that this is a binary classification problem!

Finally, let's do Support Vector Machines!

## g. Support Vector Machines (SVM)

Confusion matrix for Support Vector Machines

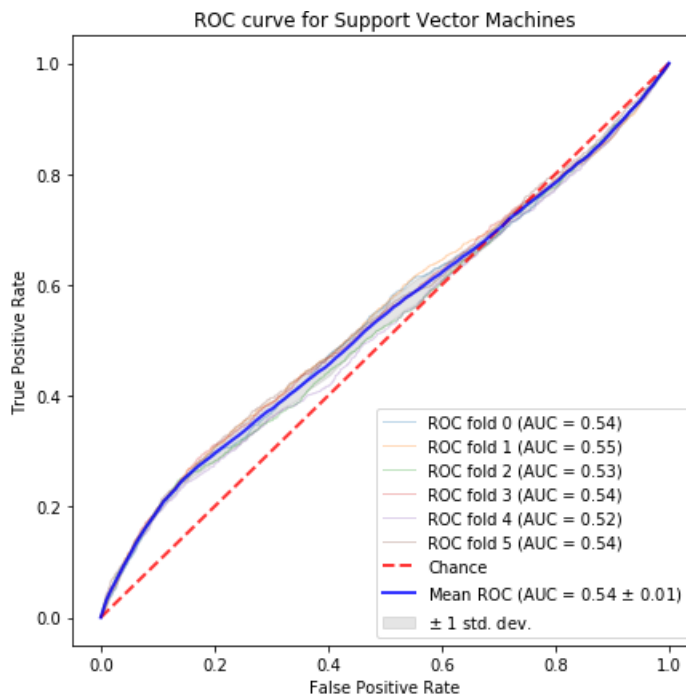
True label	0	1
0	5944	18
1	1757	19
Predicted label		

Precision was: 0.5135135135135135

Accuracy was: 0.7706125613853709

Recall was: 0.010698198198198198

Well, the precision is slightly better than the regression models, and the positive predictions are about the same, so perhaps SVM is the best model so far, or perhaps it is just a result of very few positive predictions in general. Let's see what the ROC curve can show.



SVM is the most complex to run on this data. It also takes by far the most time to run, about an hour for a dataset that is small (less than 40,000 observations)

However, it was still beaten by both logistic regression and k-nearest neighbors.



## Conclusion

The regressions performed clearly show that the numerical features of the dataset all help predict whether someone will report their own income or not, in the following order of importance (the direction, + meaning more likely to report and - meaning less likely, is in parenthesis):

- Education (-)
- Drug use (+)
- Gender (+ -> males)
- Height (-)
- Age (+)
- Smoking (+)

The actual classification part of the problem did not go as well. The ROC curves, especially of logistic regression, show that there is some predictive power to the models (outside of linear regression), but it probably is not strong enough to be of practical use.

### Next steps

In depth text analysis of the essays of dating profiles could be done; this could be transformed into a proxy for writing skill which might not perfectly correlate with education level (you don't necessarily have to get a higher degree to be well educated). This might give more predictive power to the model.

Also, although the data seems linear, it might be worth exploring the linearity of the relationship for each feature in turn. This also might improve the predictive power of the model a little bit.

### Other data

If there were a way to get the actual incomes of respondents and verify their other reported characteristics, we could see the amount of bias in various reported features and adjust for it accordingly. This might increase the predictive power of the various models slightly.

Other quantitative predictors (more in-depth profiles, perhaps asking quantifiable questions such as number of sexual partners,) could add further predictive power to the models if it were available.

Finally, the results might just be specific to San Francisco, where almost all the data in the dataset comes from. It would be interesting to compare whether people reported income based on the type of place they lived (not possible in this dataset because almost all data comes from San Francisco), and a much larger dataset could help with that.