OKCUPID

Date a Scientist Project
By Teodora

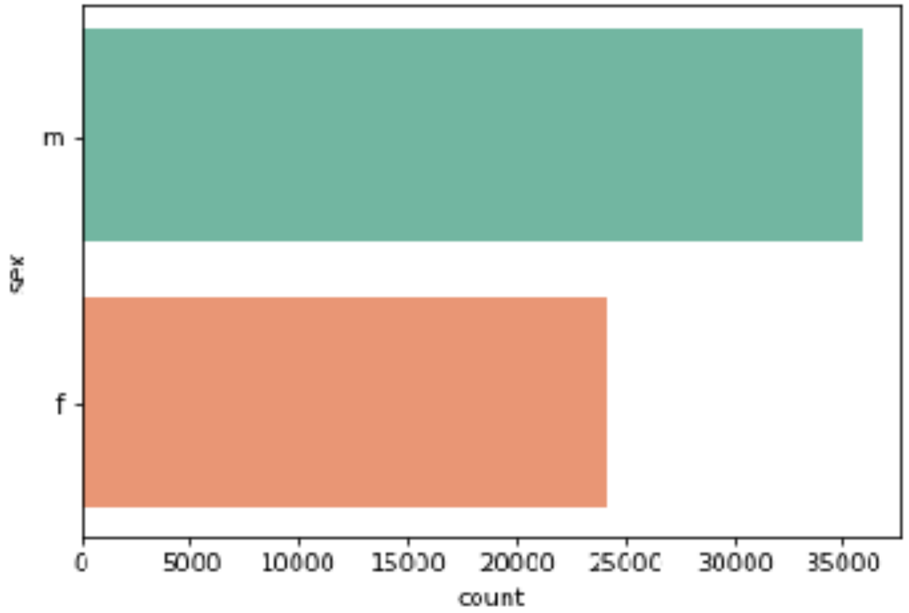
This is a Codecademy capstone project for the data science path, where we have to analyse data from the online dating application called OKCupid.

The objective is to try and predict the astrological sign of the OKCupid users by using the other variables provided, and applying Machine Learning algorithms.

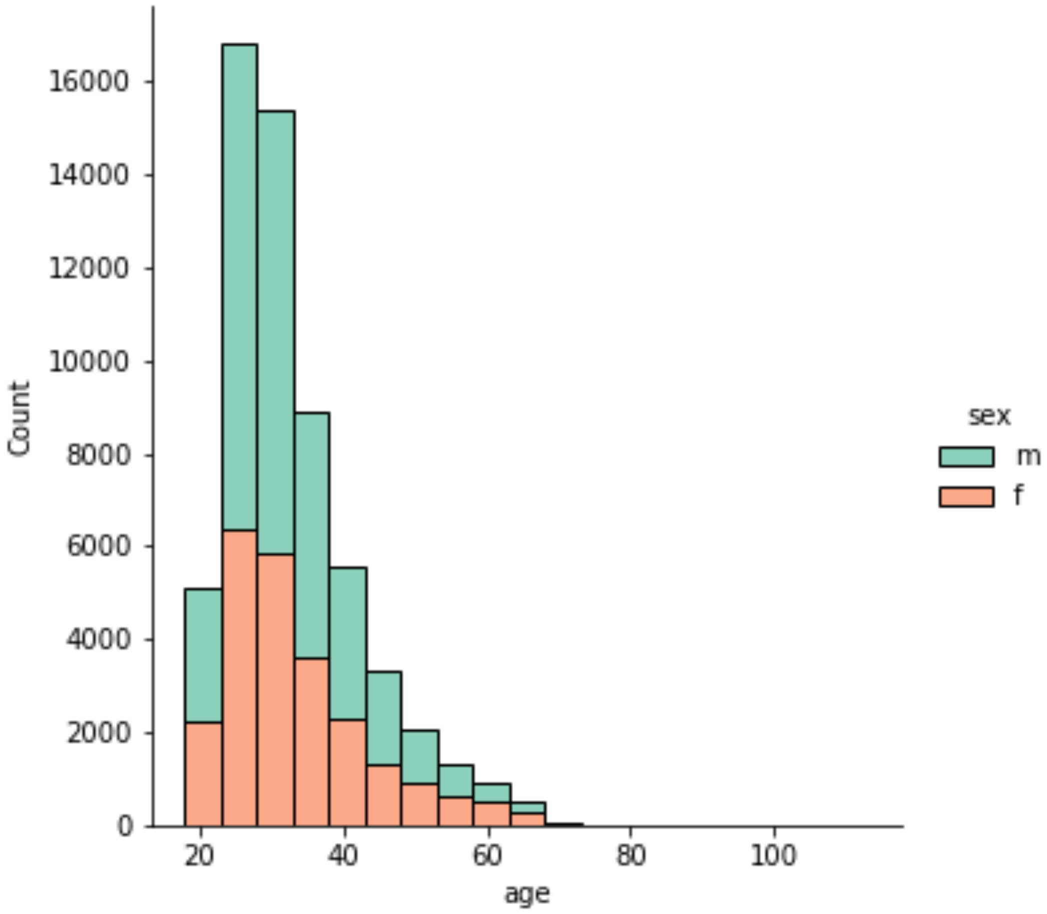
I have used K Nearest Neighbors, Decision Tree Classifier and Support Vector Machine.

But first, let’s have a look at the data.

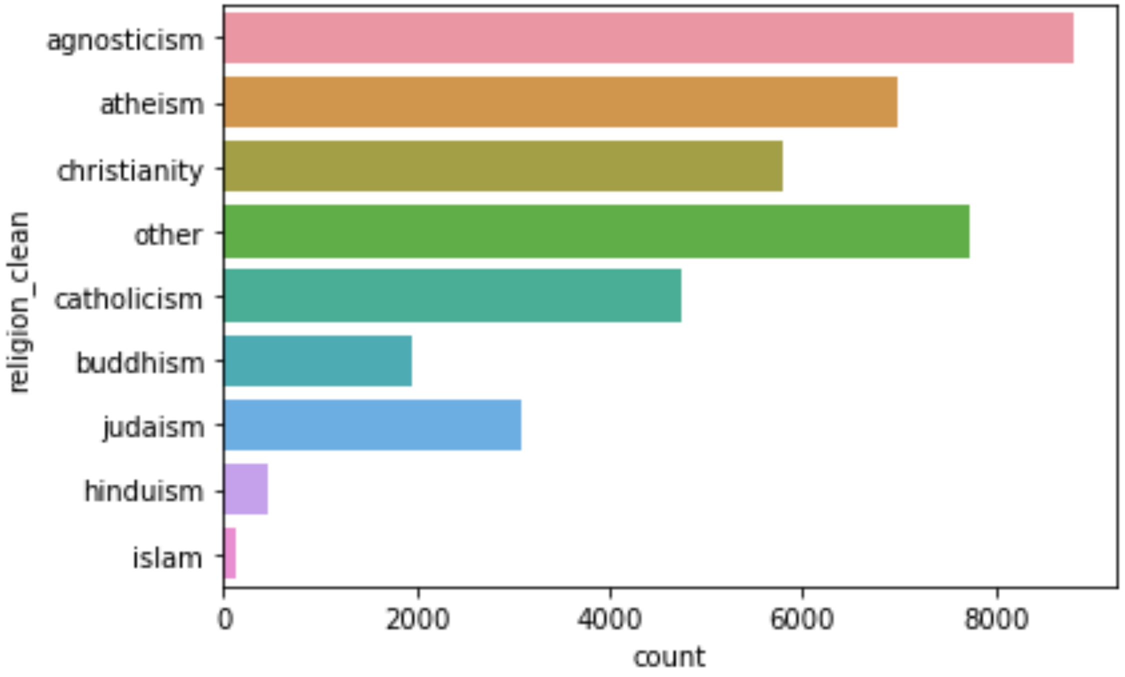
We have about 10.000 more men than women in our dataset.

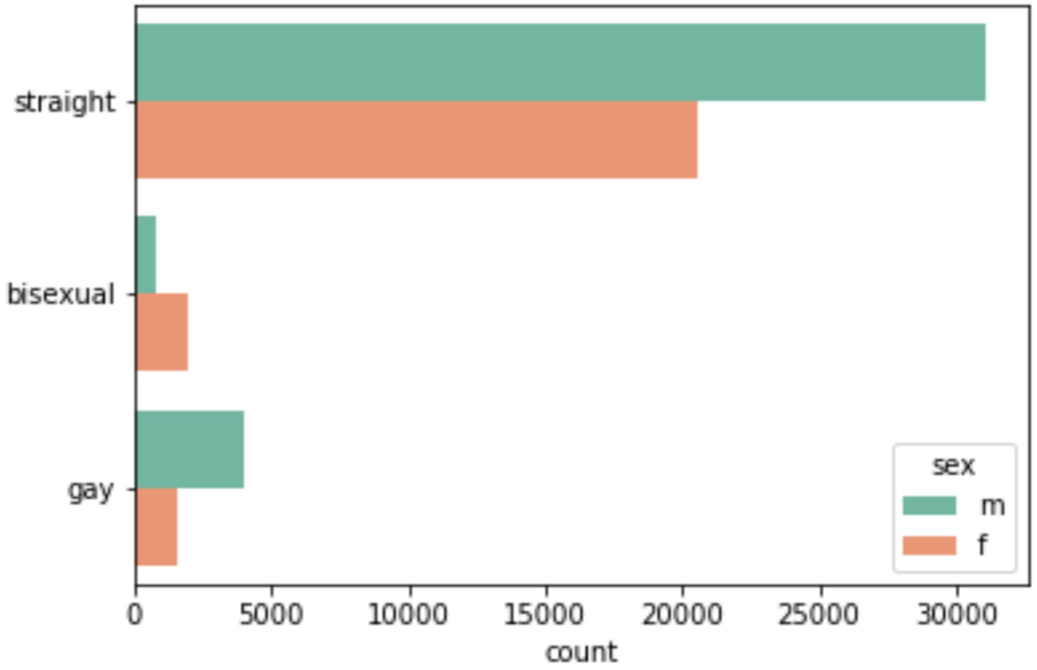


The average age is 32 years, and as we can see, most users are in the 20 - 40 years age range.

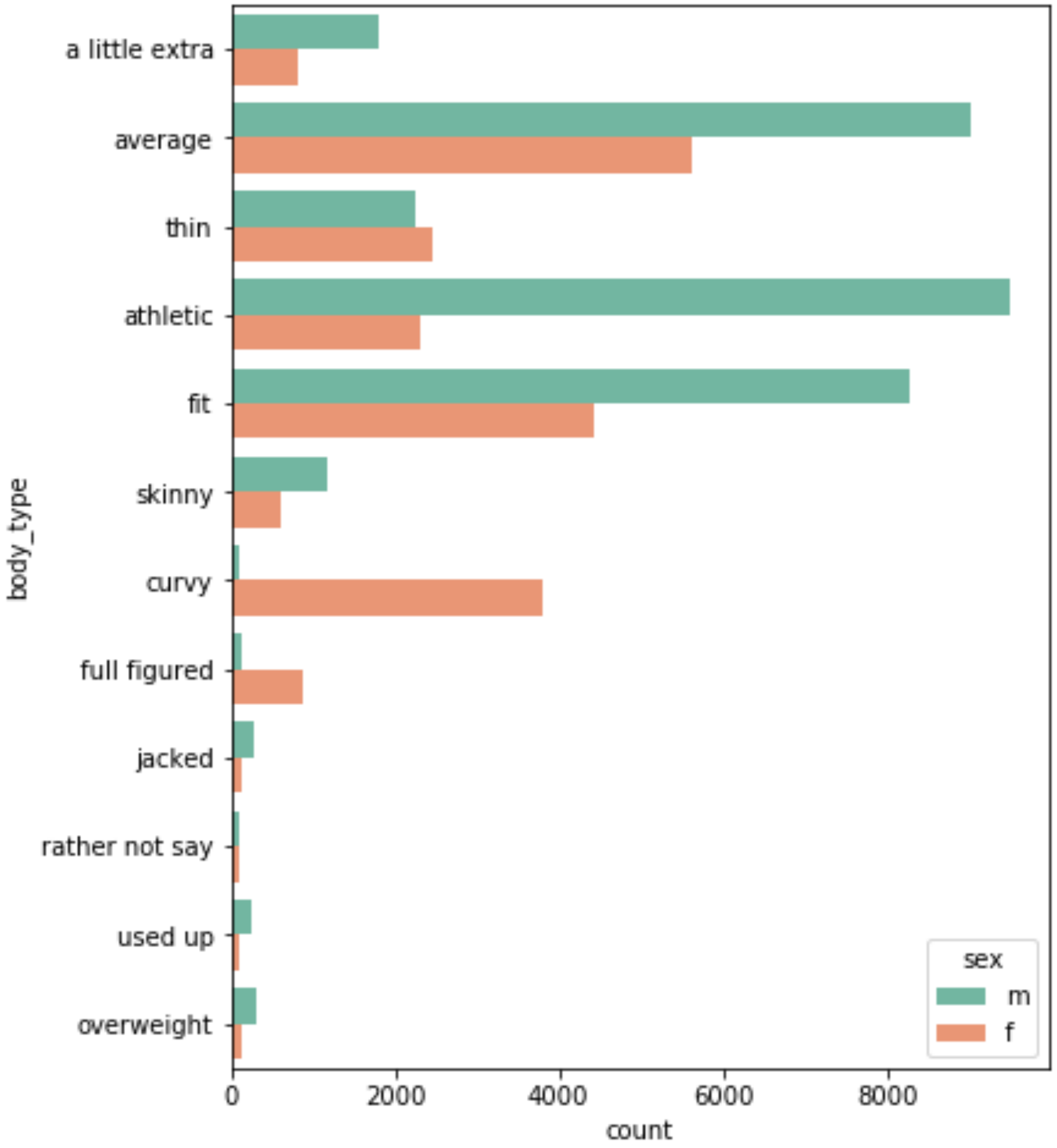


Religions

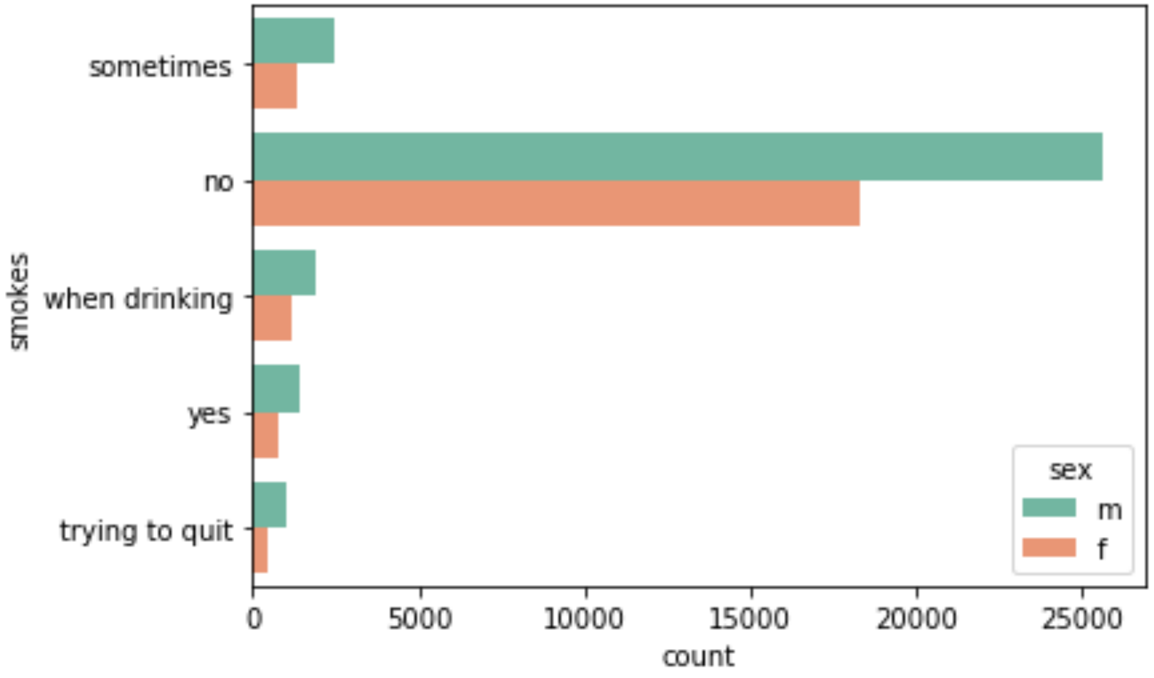




Sexual orientation

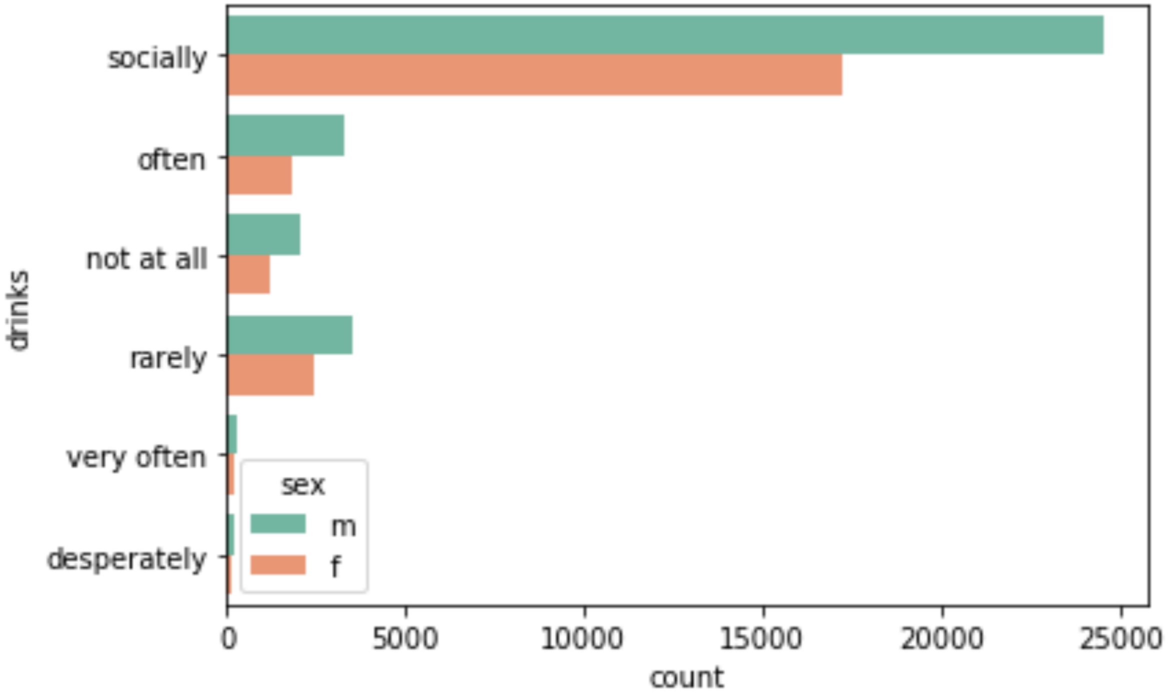


Body type count

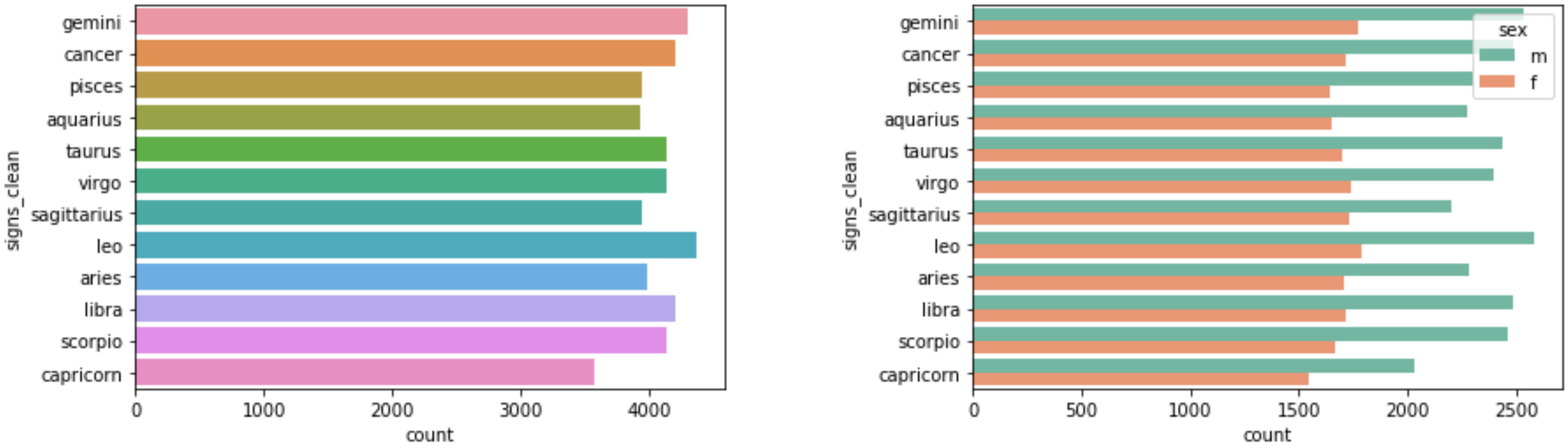


Smoking users

Drinking users

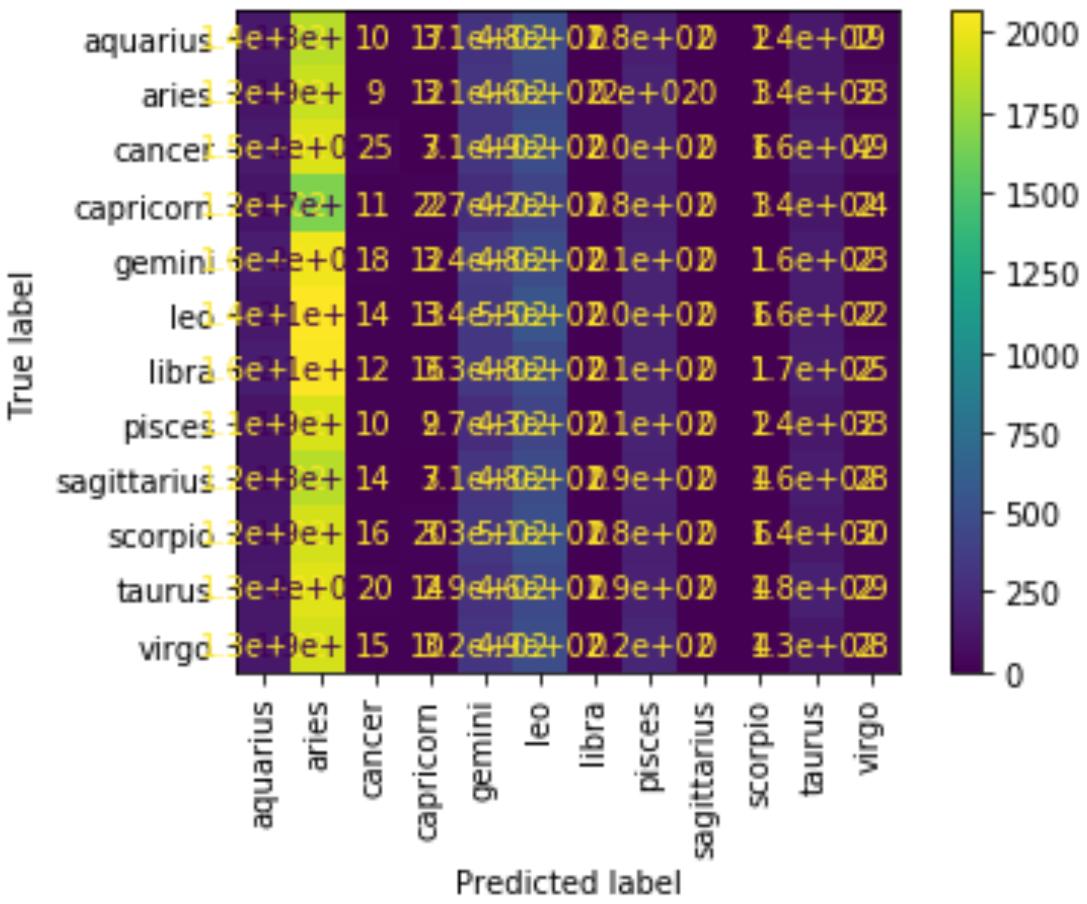


Signs



As we can see, the signs column is the most proportionate in the data set, so next we will try to see if we can predict a user's sign based on other variables. First, I tried to see if we can predict a sign based on someone's smoking and drinking. I mapped the data and used the train_test_split function to train my models.

I used K Nearest Neighbors, Decision Tree Classifier, and Support Vector Machine.



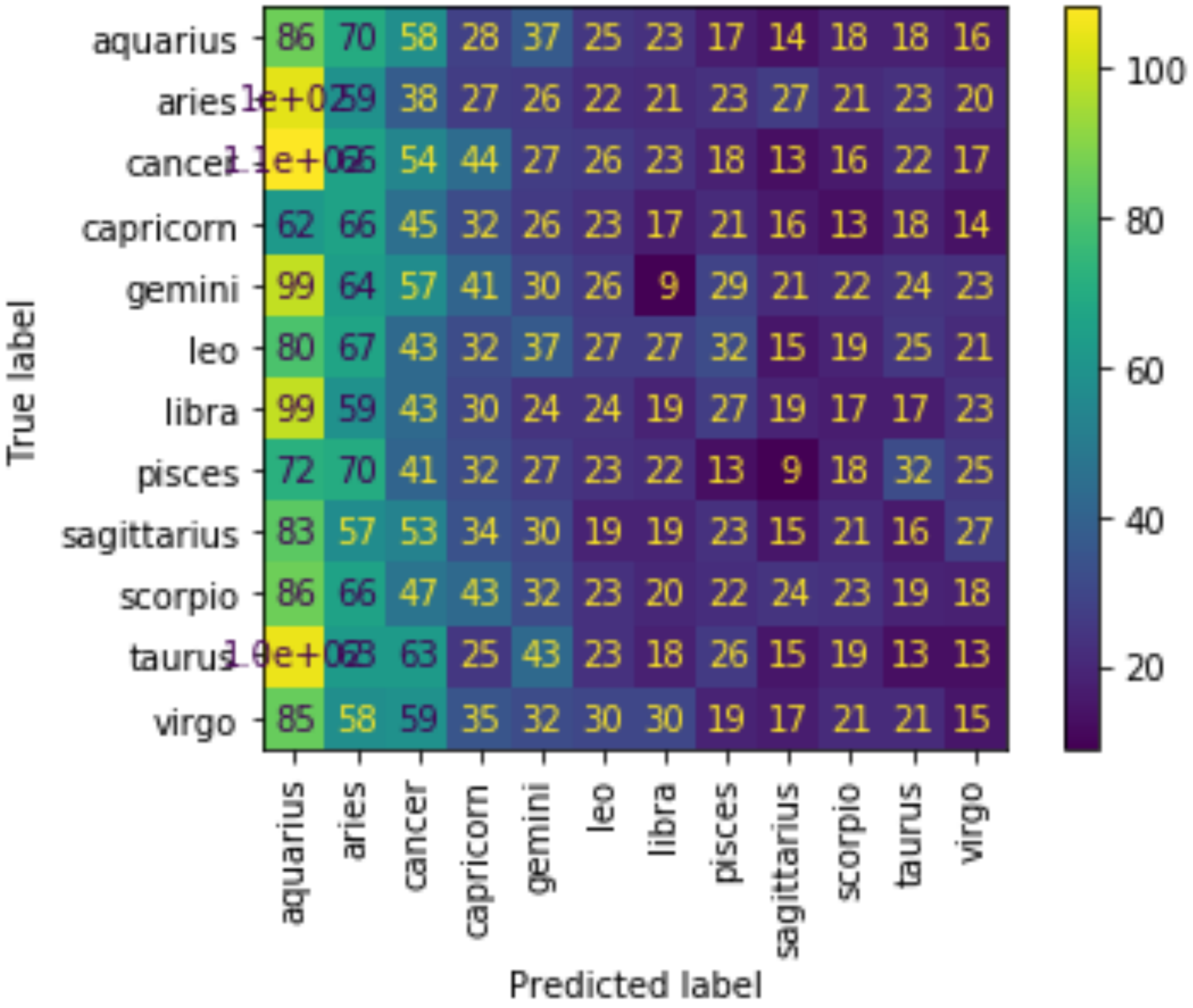
KNN confusion matrix

Mean accuracy for this model 0.08698097770505216, so obviously, what we are trying to achieve is not really possible. The model is only 8% accurate, which means that we should

rethink our strategy and maybe change the variables, as this is not enough data for the algorithm. Just for the sake of it, I went along and tried the other two methods as well. The mean accuracy for both of them was under 10% so I decided they are not relevant.

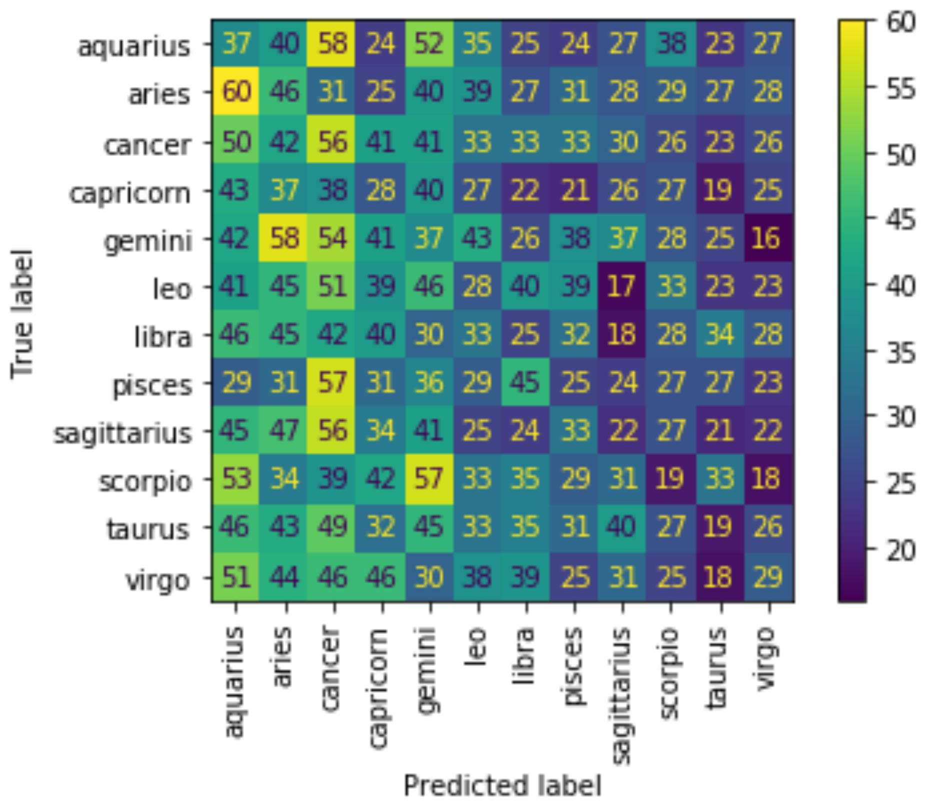
Next step was rethinking the strategy so I chose body type, diet, orientation, education, religion, sex and job columns as variables in trying to predict the sign.

KNN classifier had a mean accuracy of 0.32211180963969444, definitely an improvement from the first try.

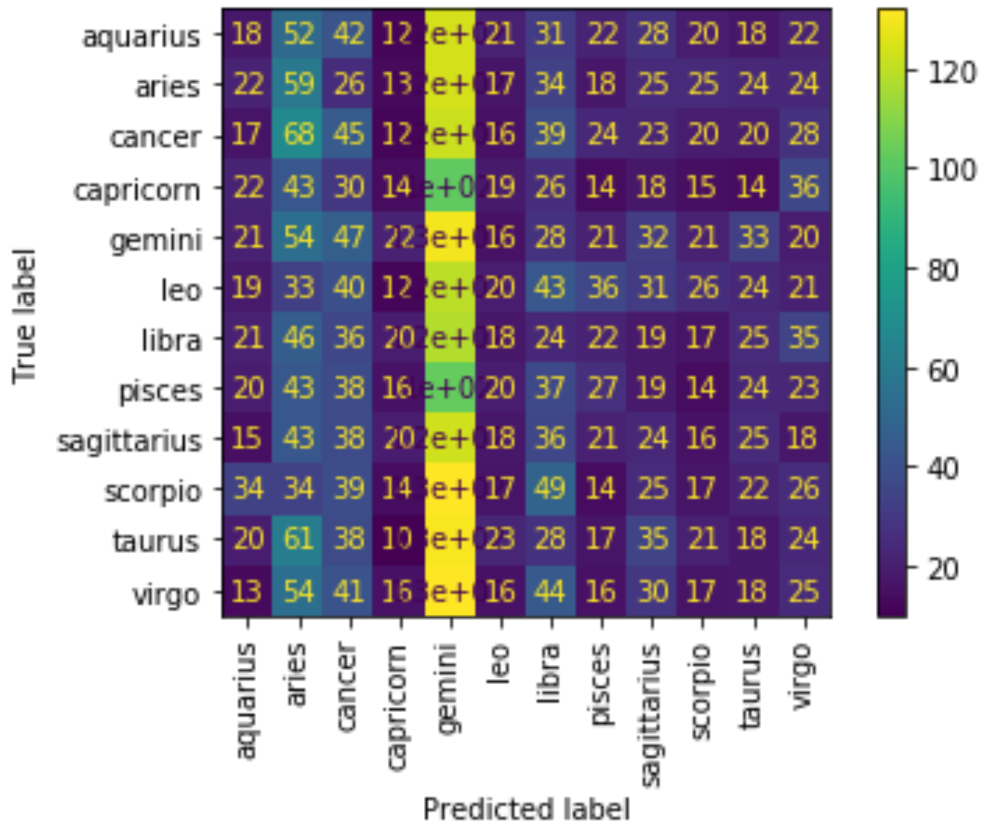


	aquarius	aries	cancer	capricorn	gemini	leo	libra	pisces	sagittarius	scorpio	taurus	virgo
aquarius	86	70	58	28	37	25	23	17	14	18	18	16
aries	1e+02	59	38	27	26	22	21	23	27	21	23	20
cancer	1.1e+02	65	54	44	27	26	23	18	13	16	22	17
capricorn	62	66	45	32	26	23	17	21	16	13	18	14
gemini	99	64	57	41	30	26	9	29	21	22	24	23
leo	80	67	43	32	37	27	27	32	15	19	25	21
libra	99	59	43	30	24	24	19	27	19	17	17	23
pisces	72	70	41	32	27	23	22	13	9	18	32	25
sagittarius	83	57	53	34	30	19	19	23	15	21	16	27
scorpio	86	66	47	43	32	23	20	22	24	23	19	18
taurus	1.0e+02	63	63	25	43	23	18	26	15	19	13	13
virgo	85	58	59	35	32	30	30	19	17	21	21	15

KNN confusion matrix



Decision Tree Classifier confusion matrix



Decision Tree Classifier (25) confusion matrix

The mean accuracy for the Decision Tree Classifier is 0.7323058203204218

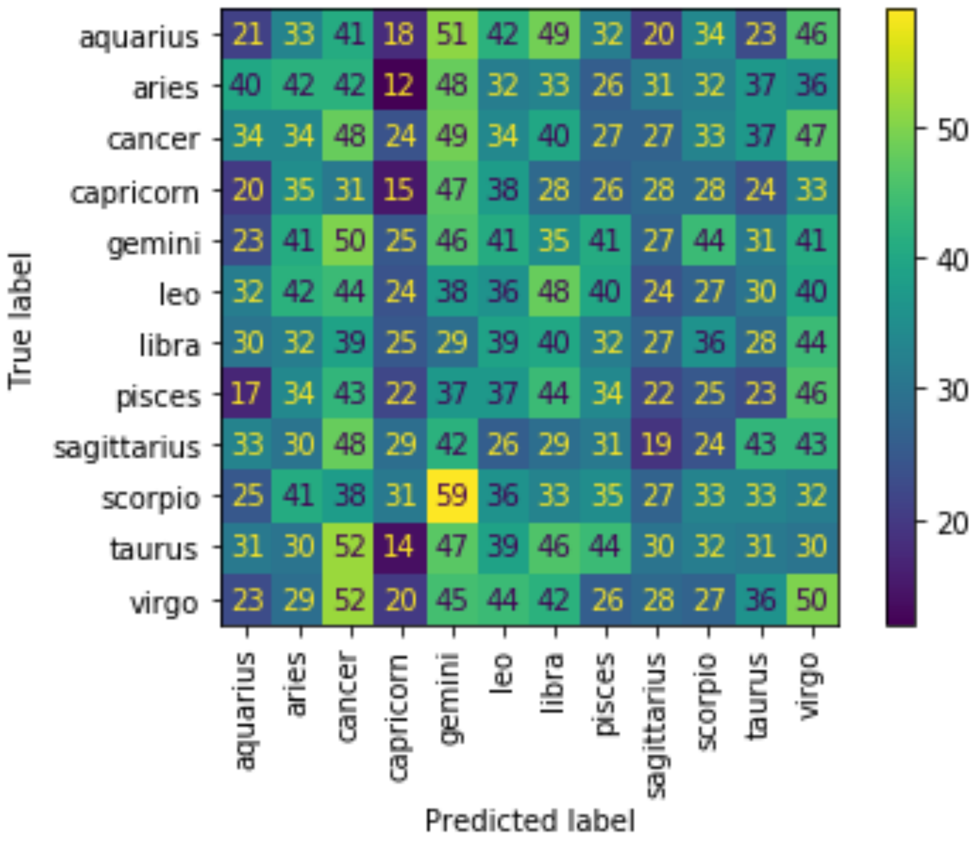
Compared to KNN, this is a better fit to our data, and more reliable… but is it?

Our model's maximum depth is 64, so I tried to set the depth at 25.

After setting the maximum depth at 25, we get a mean accuracy of 0.38058541201919827

It is a bit above KNN, but surely if we set the depth lower, it will get lower than KNN.

Support Vector Machine confusion matrix



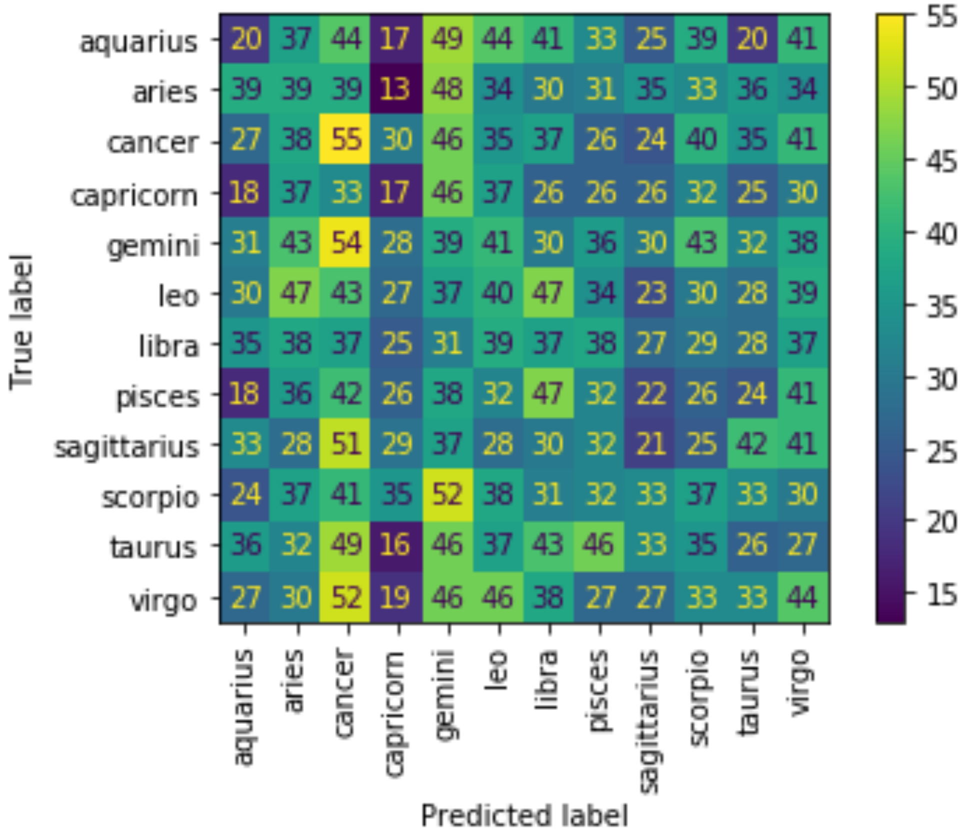
	aquarius	aries	cancer	capricorn	gemini	leo	libra	pisces	sagittarius	scorpio	taurus	virgo
aquarius	21	33	41	18	51	42	49	32	20	34	23	46
aries	40	42	42	12	48	32	33	26	31	32	37	36
cancer	34	34	48	24	49	34	40	27	27	33	37	47
capricorn	20	35	31	15	47	38	28	26	28	28	24	33
gemini	23	41	50	25	46	41	35	41	27	44	31	41
leo	32	42	44	24	38	36	48	40	24	27	30	40
libra	30	32	39	25	29	39	40	32	27	36	28	44
pisces	17	34	43	22	37	37	44	34	22	25	23	46
sagittarius	33	30	48	29	42	26	29	31	19	24	43	43
scorpio	25	41	38	31	59	36	33	35	27	33	33	32
taurus	31	30	52	14	47	39	46	44	30	32	31	30
virgo	23	29	52	20	45	44	42	26	28	27	36	50

For SVM, we got a mean accuracy of 0.719461907659028

Our initial C parameter was set at 2. At C = 0.01 the data is under fit and mean accuracy is 0.09321976610559048.

I also changed C to equal 10, and our mean accuracy went up to 0.7323058203204218.

The second confusion matrix is for C = 10, so we can notice the differences.



	aquarius	aries	cancer	capricorn	gemini	leo	libra	pisces	sagittarius	scorpio	taurus	virgo
aquarius	20	37	44	17	49	44	41	33	25	39	20	41
aries	39	39	39	13	48	34	30	31	35	33	36	34
cancer	27	38	55	30	46	35	37	26	24	40	35	41
capricorn	18	37	33	17	46	37	26	26	26	32	25	30
gemini	31	43	54	28	39	41	30	36	30	43	32	38
leo	30	47	43	27	37	40	47	34	23	30	28	39
libra	35	38	37	25	31	39	37	38	27	29	28	37
pisces	18	36	42	26	38	32	47	32	22	26	24	41
sagittarius	33	28	51	29	37	28	30	32	21	25	42	41
scorpio	24	37	41	35	52	38	31	32	33	37	33	30
taurus	36	32	49	16	46	37	43	46	33	35	26	27
virgo	27	30	52	19	46	46	38	27	27	33	33	44

In conclusion, it seems that Support Vector Machine is the best classifier to use for our data set, as the other classifiers did not perform up to our expectations.

Also, choosing your variables right makes a huge difference on machine learning algorithms performance.

This was a cool project to work on, and I feel like I have expanded my Python knowledge and also trained my skills.