CMPT 353 - Computational Data Science
Final Group Project

Authored by:
Homayoun Banazedeh - 301336653
Kassymkhan Bekbolatov - 301349731
Sergii Padalka – 301333953

Accomplishments:

Homayoun:

- Trained machine learning models such as linear regression and k-nearest neighbours to predict the happiness score of a country given the happiness scores of its neighbours.
- Trained machine learning models to predict the happiness score of a country in a given year based on the data from the previous years with linear regression. 98% accuracy was achieved.
- Performed ETL to transform the yearly data into a new data frame suitable for machine learning.
- Visualized the results of machine learning and compared them with the actual values.
- Extensively contributed to the report.

Kassymkhan:

- Cleaned and tailored data to create and train machine learning models based on linear regression to predict the future happiness rate in countries based on the happiness scores in previous years.
- Trained machine learning model based on linear regression to give 98% score accuracy in prediction of future happiness rates for countries.
- Employed ordinary least squares to get the coefficients and p-value of linear regression to successfully rule out correlation.
- Analyzed the output of the linear regression model and its visualization of residuals.
- Extensively contributed to the report.

Sergii:

- Cleaned initial data by manual and automated means, removing bottlenecking discrepancies and devising methods of extracting and merging data for future parts of the project.
- Aggregated data for happiness scores of countries and their neighbours to determine similarities
- Contributed to the report through addition of information as well as extensive editing for cohesion and grammar.
- Supported other team members during machine learning analysis stage.

Overview

The world can sometimes be a cruel and miserable place. At other times, you might be in luck in live large. This project focuses on the happiness levels of people by country. In particular, we tried to explore the question of whether neighbouring countries affect a country's happiness, and by how much, if at all. Techniques such as data cleaning, ETL, aggregation, machine learning and statistical analysis were performed to help with the research.

Problems attempted:

1. Predicting the happiness score of a country based on the average score of its neighbours over a 3-year period.
2. Predicting the happiness score of a country in a given year based on its neighbours' happiness scores with machine learning. In particular, linear regression and k-nearest neighbours.
3. Dividing the countries by geographical regions and observing whether the prediction score for (2) increases.
4. Predicting the happiness scores of another year given the previous years' results.

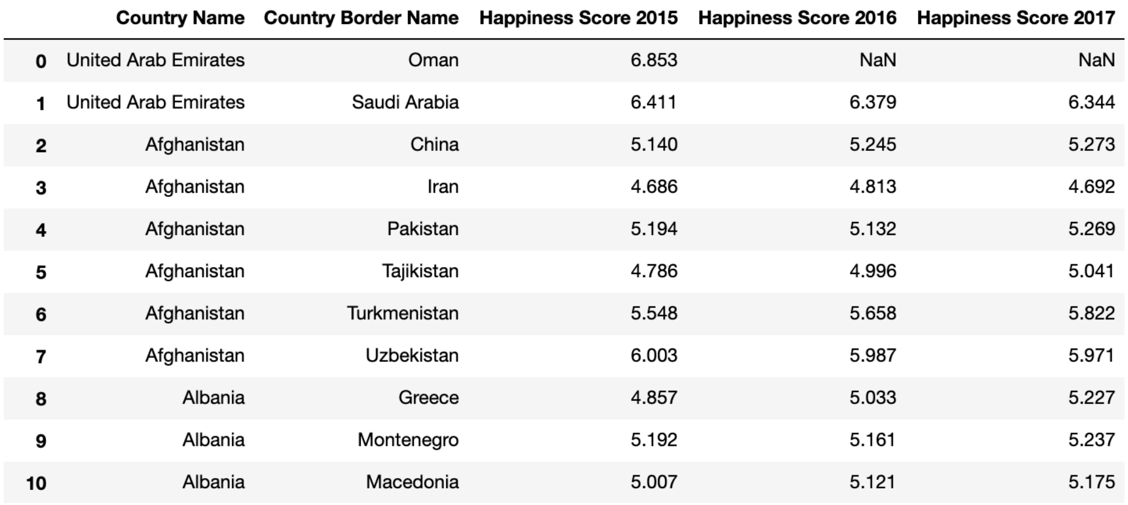
Processing and Analysis

The initial task was to collect the data. Global statistics with calculated happiness scores were ready from Kaggle [1], albeit with other data unnecessary to us and not for every country every year. To deduce the neighbours of a country, another source was necessary [2]. Here it was necessary to do a significant amount of manual labour and pattern matching make sure that the country names match in both the happiness data and border data.

The process of data cleaning and ETL involved the following:

- Country happiness data and border data were loaded from csv files into pandas dataframes.
- Excess columns were dropped from all dataframes, and remaining columns were appropriately renamed.
- Country name data were extracted from happiness data to then match up with the countries in the border data. get_close_matches from the difflib library as well as manual labour were employed to match the names as much as possible.
- Happiness data from all three years were joined with the border data to get the happiness scores for the bordering countries for those three years.

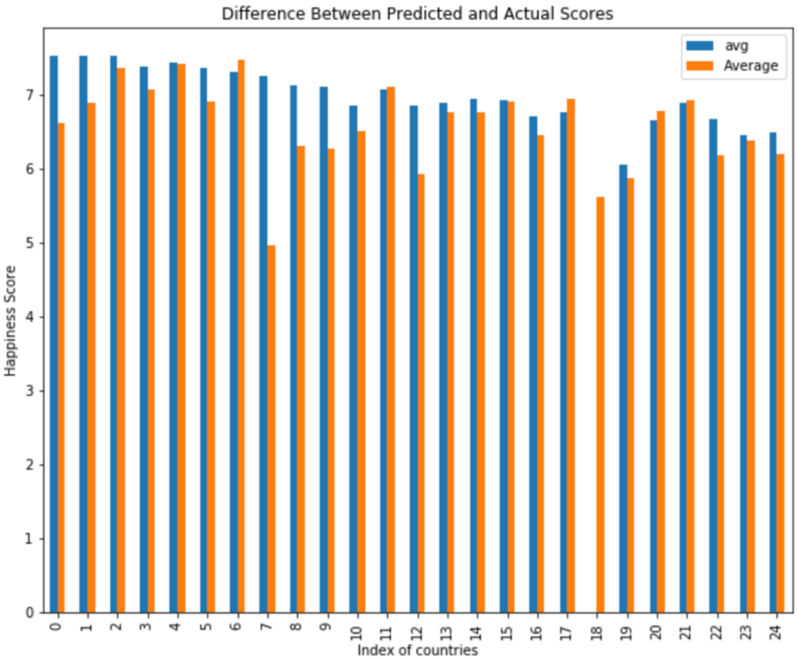
The result can be seen here:



	Country Name	Country Border Name	Happiness Score 2015	Happiness Score 2016	Happiness Score 2017
0	United Arab Emirates	Oman	6.853	NaN	NaN
1	United Arab Emirates	Saudi Arabia	6.411	6.379	6.344
2	Afghanistan	China	5.140	5.245	5.273
3	Afghanistan	Iran	4.686	4.813	4.692
4	Afghanistan	Pakistan	5.194	5.132	5.269
5	Afghanistan	Tajikistan	4.786	4.996	5.041
6	Afghanistan	Turkmenistan	5.548	5.658	5.822
7	Afghanistan	Uzbekistan	6.003	5.987	5.971
8	Albania	Greece	4.857	5.033	5.227
9	Albania	Montenegro	5.192	5.161	5.237
10	Albania	Macedonia	5.007	5.121	5.175

Please refer to the code in Excess Column Removal ETL.py and Three Year Data ETL.py for more information about the code.

In Prediction by Aggregation with Neighbours.py, the mean of happiness scores was found first grouping by year and bordering country, and then just by the bordering country. These were joined with the scores of the countries themselves to compare appropriately.



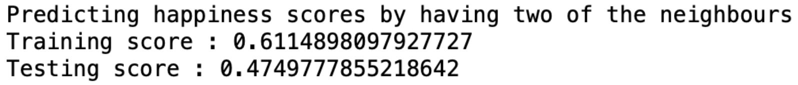
Indeed, we can see that some of the scores are more or less closely aligned with what they should be. Others – not so much. We should look to other methods to see if this prediction may be improved.

Logically we should try another method for prediction. The next choice was to attempt machine learning to determine a country's happiness score based on its neighbours' happiness scores.

We predicted that the happiness score for a country may be correlated to its neighbour's happiness. We observed that, for instance, most of the countries in Scandinavia have very high scores that are similar to each other. We used the 2015 data and performed ETL and data cleaning on the original data to transform it to a format that can be used for machine learning. The ETL was related to what was done previously done but was nonetheless challenging. We wanted a dataframe which contained the name of the country, the happiness score of 3 of its neighbours, and its region.

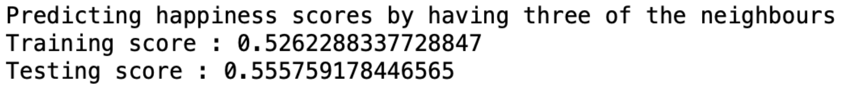
- Countries that have less than 3 neighbours were removed.
- Reformatting the data was challenging because none of us exactly knew how to manipulate them. An SQL approach was used to resolve the issue, in which dataframes were treated as tables.

Upon obtaining the improved format, we trained and tested the linear regression model on 2, 3 and 4 neighbours.



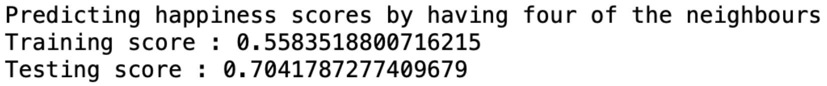
```
Predicting happiness scores by having two of the neighbours
Training score : 0.6114898097927727
Testing score : 0.4749777855218642
```

Two neighbors: The first two neighbours of those countries with at least 2 neighbours (120 countries) were used. Since the training and testing scores were significantly different, we think that underfitting had occurred. It is interesting to note that each time we ran the model we got a very different training/testing score. Sometimes the testing score is greater than the training score which is unusual. In general, we were looking to find for how many neighbours the training and testing scores will be similar.



```
Predicting happiness scores by having three of the neighbours
Training score : 0.5262288337728847
Testing score : 0.555759178446565
```

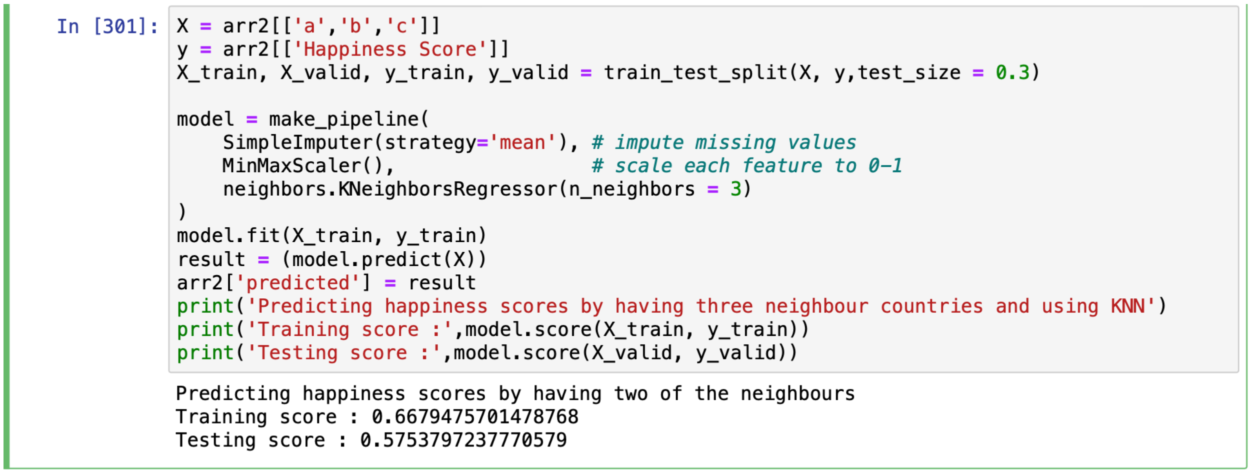
Three neighbours: This seemed to not be overfitting and not underfitting. After several tests, the training and testing scores look to be more stable. It looks like an optimal score we can reliably get by linear regression.



```
Predicting happiness scores by having four of the neighbours
Training score : 0.5583518800716215
Testing score : 0.7041787277409679
```

Four neighbours: The training and testing scores here are greater than with three neighbours but they vary a lot. For instance the testing score is 15% more than the training score. This is rare to because the training score is almost always higher than the testing score. Our final guess is that having four neighbours leads to overfitting.

We tried to use k-nearest neighbours but to no avail. The results are not much different.

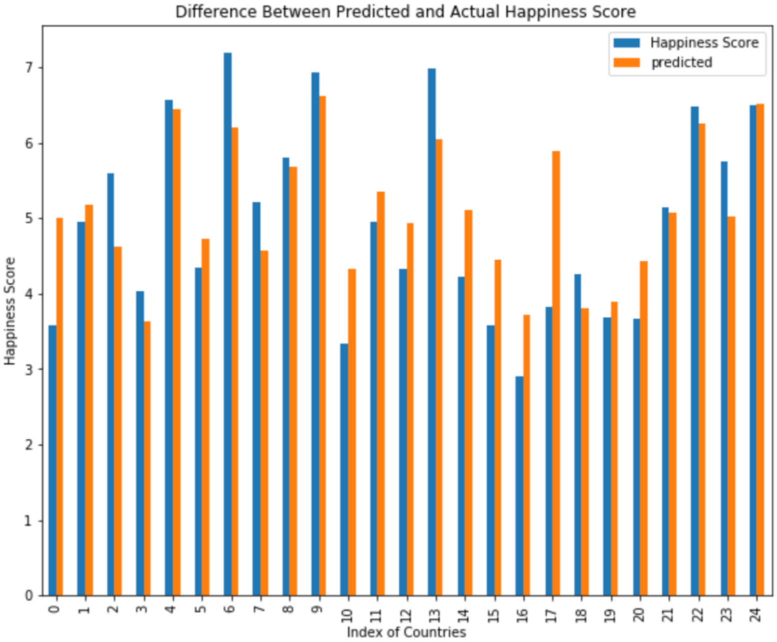


```
In [301]: X = arr2[['a','b','c']]
          y = arr2[['Happiness Score']]
          X_train, X_valid, y_train, y_valid = train_test_split(X, y,test_size = 0.3)

          model = make_pipeline(
              SimpleImputer(strategy='mean'), # impute missing values
              MinMaxScaler(),                 # scale each feature to 0-1
              neighbors.KNeighborsRegressor(n_neighbors = 3)
          )
          model.fit(X_train, y_train)
          result = (model.predict(X))
          arr2['predicted'] = result
          print('Predicting happiness scores by having three neighbour countries and using KNN')
          print('Training score :',model.score(X_train, y_train))
          print('Testing score :',model.score(X_valid, y_valid))
```

```
Predicting happiness scores by having two of the neighbours
Training score : 0.6679475701478768
Testing score : 0.5753797237770579
```

Let us look at a visualization of the data from linear regression. Here is a plot demonstrating the actual happiness scores compared with our predictions.



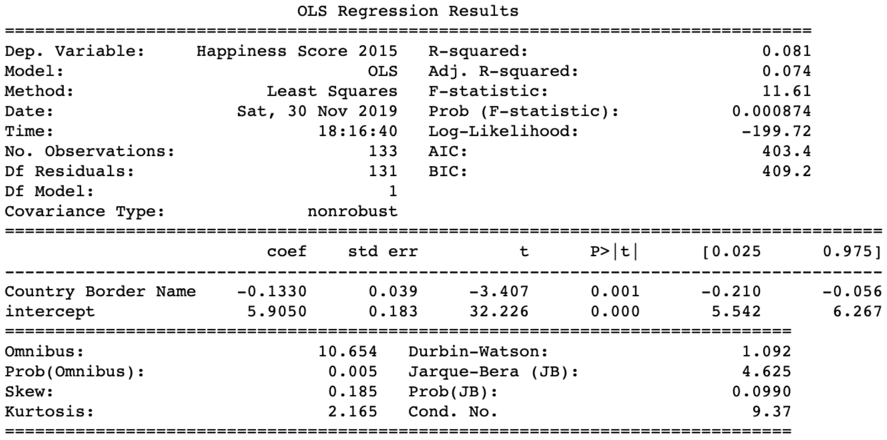
Given the results here, we can say that linear regression was reasonably successful in predicting happiness scores.

An additional task we wanted to explore was to predict happiness score values ahead of the time using machine learning. Once again, dataframes were extracted from csv files and merged like in the first averaging task. Linear regression worked well so we stuck with it. We trained the model using 2015 and 2016 happiness scores as x-values and 2017 ones for the y-value. The validation score for Xtest and ytest was found to be 98% – a significantly high value! A positive result indeed.

Finally, we tried to explore predictions for happiness scores within geographical regions. However, we were unable to obtain the most promising results. Our statistical analysis approach to explain why is detailed below.

Hypothesis: **We want to find how the number of neighbours of a given country affects its happiness rate.**

We created a dataframe to contain the region, country name, number of neighbours and happiness score. We performed the ordinary least square to see what the coefficients are and whether p-value is significant or not.
If p-value is significant (less than 0.05) then we reject null hypothesis, meaning that coefficient is not zero. If p-value is greater than 0.05 then we fail to reject null hypothesis – the coefficient may or may not be zero.
In analyzing the happiness scores for 2015, there was a $p = 0.001 < 0.05$. We therefore chose to reject the null hypothesis. I.e. the coefficient is not zero.
We used OLS with the number of neighbours and happiness scores to help with linear regression.
We then used the summary function to look see the coefficient and p-values (as seen below).



```
                            OLS Regression Results
==============================================================================
Dep. Variable:     Happiness Score 2015   R-squared:                       0.081
Model:                              OLS   Adj. R-squared:                  0.074
Method:                   Least Squares   F-statistic:                     11.61
Date:                  Sat, 30 Nov 2019   Prob (F-statistic):           0.000874
Time:                          18:16:40   Log-Likelihood:                -199.72
No. Observations:                   133   AIC:                             403.4
Df Residuals:                       131   BIC:                             409.2
Df Model:                             1
Covariance Type:              nonrobust
=======================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------------
Country Border Name    -0.1330      0.039     -3.407      0.001      -0.210      -0.056
intercept               5.9050      0.183     32.226      0.000       5.542       6.267
==============================================================================
Omnibus:                       10.654   Durbin-Watson:                   1.092
Prob(Omnibus):                  0.005   Jarque-Bera (JB):                4.625
Skew:                           0.185   Prob(JB):                       0.0990
Kurtosis:                       2.165   Cond. No.                         9.37
==============================================================================
```

It was not very useful to relate the number of neighbours of countries with its happiness scores, but we also wanted to see how it can affect the score of the linear regression model. It turned out that score was a negative number which is not correct. Then we used OLS to look at coefficients and their p-values. The p-value of the coefficient was smaller than alpha (0.05) and so no correlation was found between number of neighbours and a country's happiness rate.

Limitations and Conclusions

Limitations

- Getting the data was challenging because we had access to only 3 years of data and for machine learning tasks we needed more data. Any other data after that would have from other sources and was formatter differently such that even more effort was to be required of ETL.
- The countries were named differently in the happiness data and border data. Therefore, merging them was not working properly and we had to make sure all the country names were in the same format.
- Using the plots and histograms to communicate our results was challenging because we had to identify how plotting works and how to plot our results in a meaningful way.
- At first we tried to see how countries in the same region are affecting each other's happiness score but we could not make it work because
  - the X and y values of the same region were incompatible and
  - we could not train the model based on some of the countries in the region to test on other countries of the same region.

  Then we switched our focus to see how many neighbours each country has and how they are affecting the happiness score of the country. We wanted to prove that there is not much correlation between these two variables and this was indeed the case.
- Had we had more time, we would have liked to look into countries with maritime borders and twitter sentiment analysis to determine the potential causes of happiness levels in a country.

Conclusions

1. Predicting the happiness score of a country based on the average score of its neighbours over a 3-year period was proven to be a reasonable strategy given the similar values observed.
2. Likewise, linear regression was also reasonably successful in predicting the happiness score of a country in a given year based on its neighbours' happiness scores with machine learning. Whether it be linear regression or k-nearest neighbours does not affect the findings much.
3. No correlation was found for countries within a region between their happiness scores and number of neighbours. We therefore cannot use machine learning for score predictions based on regions and number of neighbours.
4. Machine learning was proven to be effective in the case of predicting another year's set of happiness score values. The 98% validation score for linear regression shows that this is indeed something that could be looked into.

References

[1] https://www.kaggle.com/unsdsn/world-happiness?fbclid=IwAR3V1IGM6RDYWqJfegnRrYpeY2kejoy3FIUF9rjPn6Bt_hJIQXmbomiAte8

[2] https://github.com/geodatasource/country-borders