**NW Data Science Bootcamp Final Project Report**

– Happiness Analysis and Prediction w Machine Learning

**Team Members:** Ivan Choi, Paul Smith, Meng-Yin Lee

**Google Drive:** <https://drive.google.com/drive/folders/19rd_mrhzEEKfVz2Qv6fnOeG0r8fNI9eI?usp=sharing>

GitHub Repository:

<https://github.com/RebelDroid09/final-project.git>

Tableau Public: <https://public.tableau.com/profile/ivan.choi#!/vizhome/FinalProjectMap_16137055299100/Story1?publish=yes>

Did you know countries are ranked each year based on a Happiness Score? This score rates the happiness of countries on a scale from 0 to 10.

**The definition of the Happiness Score is based on survey results, that was first used in the 2012 World Happiness Report. In the survey, the respondents were asked to rate their happiness on a scale from 0 to 10. The Happiness Score is calculated by averaging the survey results of the respondents.**

The World Happiness Report, which sources its data from the Gallup World Poll.

(Ps. Gallup – the organization behind this enormous poll – interviews approximately 1,000 residents per country each year. Each respondent in this happiness survey is asked the same questions in his or her own language to produce statistically comparable results.)

World Happiness Report found a number of key factors that could likely explain the variance in happiness.

1. ***GDP per capita*:**

Gross Domestic Product – or GDP – is simply the value of all the goods and services a country produces on a yearly basis. Divide the GDP by the total population of a country and you have the GDP per capita.

Linear regression observation:

Happy countries are generally also wealthy countries.

**The wealth of a country is significantly correlated to its Happiness Score.**

(There are a handful of outliers (which happen to be some of the smallest countries) that have a very high GDP per capita. This skews the color coding of this chart a lot!)

1. **Social support**

**The correlation between the Happiness Score and the level of Social support is less linear,** but more exponential. The lack of Social support seems to affect the Happiness Score up to a certain level.

### **Healthy life expectancy**

The estimate for the healthy life expectancy represents the average number of “healthy” years a child at birth is estimated to live.

**The correlation between a healthy life and a high Happiness Score is clearly visible.**

### **Freedom to make life choices**

**The correlation between freedom and a high Happiness Score is linear with more outliers**

**(e.g. Algeria – low freedom score with high happiness index)**

### **Generosity**

Question asked: “Have you donated money to a charity in the past month?”

**People don’t necessarily have to donate money to a charity in order to be generous.**

1. **Perceptions of corruption**

**This factor has the smaller correlation to the Happiness Score**. A lot of seemingly corrupt countries still maintain a high Happiness Score.

**Data Set**

1. Kaggle: <https://www.kaggle.com/unsdsn/world-happiness>

2015.csv

2016.csv

2017.csv

2018.csv

2019.csv

1. Transparency International: <https://www.transparency.org/en/cpi/2019/results/table>

### **Machine Learning Workflow**

#### ***1.Collect and Prepare the data***

* + Utilize AWS S3 and RDS
  + Database: PostgreSQL
  + Drop data with missing values or fill in with mean value.

#### ***2.Analyze the data in Python***

* Utilize sklearn, statsmodels, seaborn, pandas, seaborn, numpy, matplotlib
* Data Analysis
* Use scatter plot and find Pearson Correlation coefficient for each factor
* Checking for multicollinearity
* Use seaborn stripplot to further analyze outliers

#### ***3.Create the Machine Learning Model***

* **Multiple Linear Regression**

Based on our data analysis and the nature of the data sets, we have selected multivariate/multiple linear regression to build the model.

##### **When implementing linear regression in a machine learning system, the variables must be continuous in nature**, not categorical.

##### **The variables in our datasets are continuous variables**, so we don't need to encode them at all.

A linear regression carried out on more than one independent variable, comparing the correlations between features for the given number of features.[¶](http://localhost:8888/notebooks/Final_Project_Happiness_ML_Multiple_Regression.ipynb#Select-multivariate/multiple-linear-regression,-a-linear-regression-carried-out-on-more-than-one-independent-variable,-comparing-the-correlations-between-features-for-the-given-number-of-features.)

#### ***4.Train the model***

Data volume: 782 records ( 2015 to 2019 world happiness report datasets from kaggle)

X = [["GDP per capita", "Social support", "Healthy life expectancy",

"Freedom to make life choices", "Generosity", "Perceptions of corruption"]]

y = ["Score"]

We use train\_test\_split to create training and testing data

train\_test\_split is a function in Sklearn model selection for splitting data arrays into two subsets:

for training data and for testing data

**from sklearn.model\_selection import train\_test\_split**

**X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.2, random\_state=9)**

#### ***5.Evaluate the model***

To validate our regression models, we use R2, RMSE and Residual plots to visually confirm the validity of your model.

i. Calculate R2 (R-squared ) and RMSE (Root mean square erro)

**Training R2 Score: 0.7755999127353526**

**Testing R2 Score: 0.7151408209729091**

**Test Set RMSE: 0.6100957246059002**

**Test Set R2: 0.7151408209729091**

# R-squared

# it ranges from zero to one, and one indicating perfect prediction.

# with zero indicating that the proposed model does not improve prediction over the mean model,

# Improvement in the regression model results in proportional increases in R-squared.

###

# RMSE

# Lower values of RMSE indicate better fit.

# RMSE is a good measure of how accurately the model predicts the response,

# and it is the most important criterion for fit if the main purpose of the model is prediction.

# The RMSE is the square root of the variance of the residuals. (Or, the square root of the mean square error (MSE))

# It indicates the absolute fit of the model to the data–how close the observed data points are to the model’s predicted values.

# Whereas R-squared is a relative measure of fit, RMSE is an absolute measure of fit.

# As the square root of a variance, RMSE can be interpreted as the standard deviation of the unexplained variance,

# and has the useful property of being in the same units as the response variable.

ii.Plot the Residuals for the Training and Testing data

A typical residual plot has the residual values on the Y-axis and the independent variable on the x-axis.

The most important assumption of a linear regression model is that the**errors are independent and normally distributed.**

**Characteristics of Good Residual Plots**

A few characteristics of a good residual plot are as follows:

1. It has a high density of points close to the origin and a low density of points away from the origin.
2. It is symmetric about the origin.

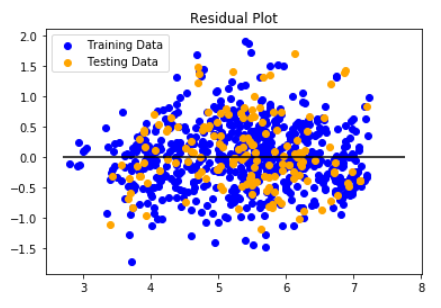
plt.scatter(model.predict(X\_train), model.predict(X\_train) - y\_train, c="blue", label="Training Data")

plt.scatter(model.predict(X\_test), model.predict(X\_test) - y\_test, c="orange", label="Testing Data")

plt.legend()

plt.hlines(y=0, xmin=y.min(), xmax=y.max())

plt.title("Residual Plot")



iii. Evaluate prediction with linear regression and statsmodel

df = data\_2015to2019.copy()

regr = linear\_model.LinearRegression()

regr.fit(X, y)

Intercept:

[2.15766649]

Coefficients:

[[1.21132866 0.58623713 1.00527655 1.70572833 0.74421236]]

# with statsmodels, OLS

X = sm.add\_constant(X) # adding a constant

model = sm.OLS(y, X).fit() # Ordinary Least Square

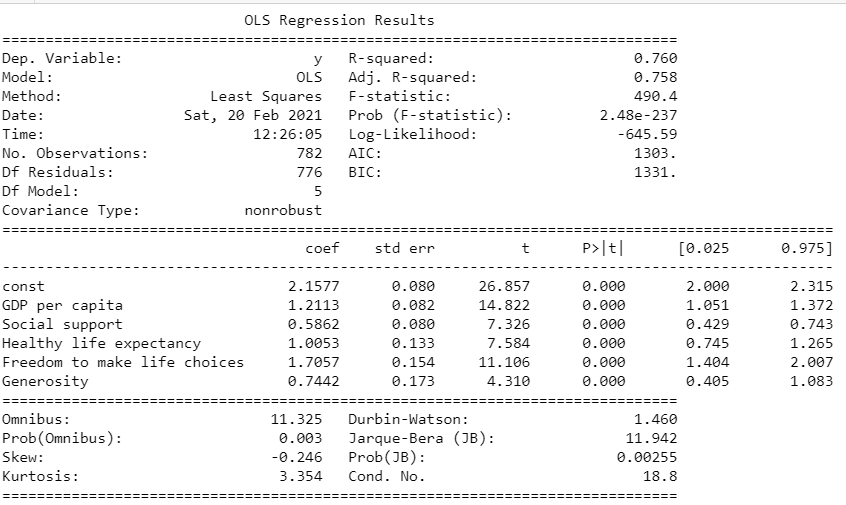
predictions = model.predict(X)

print\_model = model.summary()

**Ordinary least squares,** or linear least squares, [estimates](https://statisticsbyjim.com/glossary/estimator/) the [parameters](https://statisticsbyjim.com/glossary/parameter/) in a [regression](https://statisticsbyjim.com/glossary/regression-analysis/) model by minimizing the sum of the squared [residuals](https://statisticsbyjim.com/glossary/residuals/). This method draws a line through the data points that minimizs the sum of the squared differences between the observed values and the corresponding [fitted values](https://statisticsbyjim.com/glossary/fitted-values/).

\*parameter: unknown value of an entire population, such as mean and standard deviation

\*fitted value: predicted value



iv. Prediction with Input

# Multiple linear regression formula

# The formula for a multiple linear regression is:

#

# y= B0 + B1X1 +...+ BnXn + e

#

# y = the predicted value of the dependent variable

# B0 = the y-intercept (value of y when all other parameters are set to 0)

# B1X1= the regression coefficient (B1) of the first independent variable (X1) (a.k.a. the effect that increasing the value of the independent variable has on the predicted y value)

# … = do the same for however many independent variables you are testing

# BnXn = the regression coefficient of the last independent variable

# e = model error (a.k.a. how much variation there is in our estimate of y)

# "GDP per capita", "Social support", "Healthy life expectancy",

# "Freedom to make life choices", "Generosity", "Perceptions of corruption"

# prediction with sklearn

***# Using 2019 Finland as example to evaluate prediction. Finland score was 7.769, rank 1***

New\_GDP = 1.34

New\_Social\_Support = 1.58

New\_Health = 0.986

New\_Freedom = 0.596

New\_Generosity = 0.153

New\_CDI = 0.393

print ('Predicted Score: \n', regr.predict([[New\_GDP ,New\_Social\_Support, New\_Health, New\_Freedom, New\_Generosity,New\_CDI]]))

**Predicted Score:**

**[[7.02453596]]**

# "GDP per capita", "Social support", "Healthy life expectancy",

# "Freedom to make life choices", "Generosity", "Perceptions of corruption"

# prediction with sklearn

**# Using 2019 South Sudan as example to evaluate prediction. South Sudan score was 2.853, rank 156**

New\_GDP = 0.306

New\_Social\_Support = 0.575

New\_Health = 0.295

New\_Freedom = 0.01

New\_Generosity = 0.202

New\_CDI = 0.091

print ('Predicted Score: \n', regr.predict([[New\_GDP ,New\_Social\_Support, New\_Health, New\_Freedom, New\_Generosity, New\_CDI]]))

**Predicted Score:**

**[[3.40548722]]**

**Conclusion:**

With the individual key factor data analysis and multiple liner regression model we can obtain about 70% accuracy comparing to World Happiness Report happiness score.

We might be able to use this model to predict personal happiness with personal happiness key factors.

However, we have to accept that there is a part of the Happiness Score that cannot be explained by other – more easily measurable – factors.