INSY 5378 DATA SCIENCE PROGRAMMING APPROACH

GROUP 7

POKEMON GO

**Team Members:**

Salman Mohammed-1001398053 [salman.mohammed@mavs.uta.edu](mailto:salman.mohammed@mavs.uta.edu) Information Systems

Sujatha Sivakumar-1001359237 [sujatha.sivakumar@mavs.uta.edu](mailto:sujatha.sivakumar@mavs.uta.edu) Information Systems

Tapan Patel-1001450804 [tapansaileshbha.patel@mavs.uta.edu](mailto:tapansaileshbha.patel@mavs.uta.edu) Information Systems

Table of Contents .

**Introduction …………………………………………………………………………………….. 3**

**Web Scraping …………………………………………………………………………………… 3**

**Data Organization ……………………………………………………………………………. 3**

**Data Exploration ………………………………………………………………………………. 4**

**Prediction and Modelling …………………………………………………………………. 7**

**Deep Learning …………………………………………………………………………………… 10**

**Inferences ………………………………………………………………………………………… 13**

# Introduction

Pokemon Go is a location based augmented reality game. It was developed for IOS and Android devices by the collaborative efforts of Niantic and Nintendo in July 2016. The App is only for selective countries. The game deploys the usage of GPS to capture, battle and train virtual creatures.

In Our project, we have extracted the ratings for the app in App store and play store (IOS and Android) from HTML links, created pandas data frames, explored the data and performed prediction modelling using Linear Regression, Ridge, Lasso and Deep learning using Tensor Flow.

# Web Scraping

We have scrapped web data from HTML links using the Python module Beautiful Soup. The HTML links were obtained from 3 month’s data of the App’s App store and Play store web pages. The links were downloaded every 10 minutes. However, while extracting the data we found that few of the links were corrupted. In such cases, we replaced the unknown values with zeros for convenience.

The data consists of 12 parameters inclusive of date. They are:

* Date
* Android Average Ratings
* Android File Size
* Android Ratings 1
* Android Ratings 2
* Android Ratings 3
* Android Ratings 4
* Android Ratings 5
* Android Total Ratings
* IOS All Ratings
* IOS Current Ratings
* IOS File Size

# Data Organization

The Data organization step involves three steps:

* The extracted data from the HTML links were converted into a nested dictionary with Date and Time as key and a dictionary of parameters as value.
* The dictionary was then converted into a pandas data frame for the ease of exploration.
* The data frame was finally saved into 3 file formats viz – JSON, CSV and XLSX

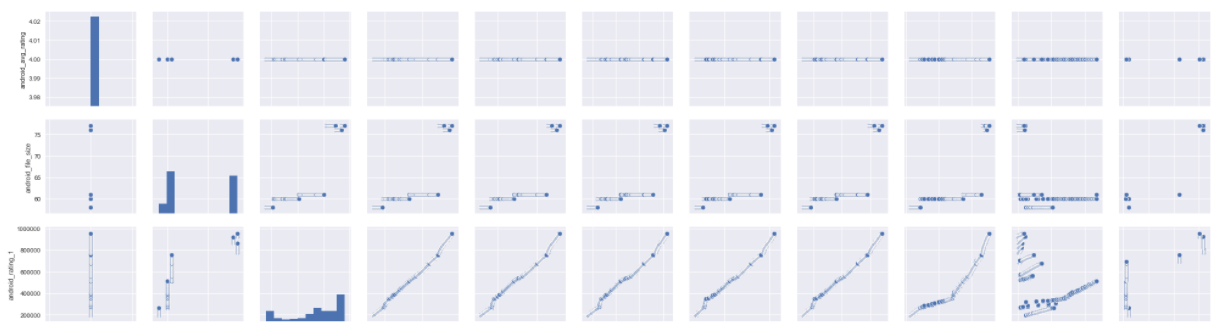
# Data Exploration

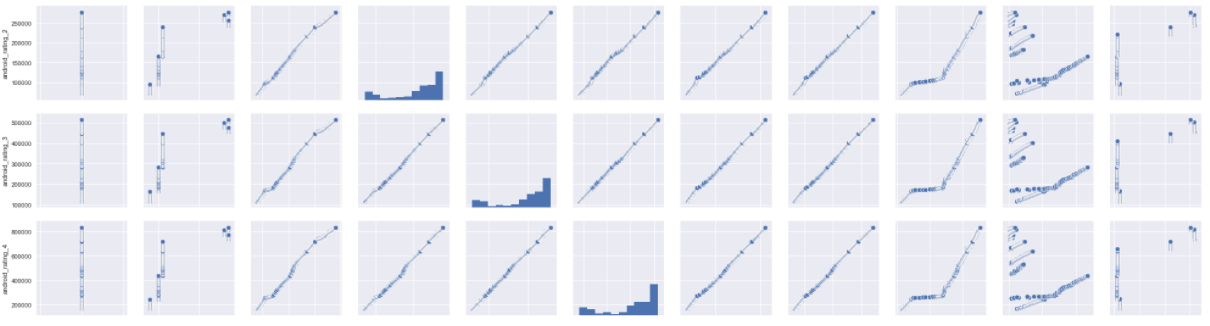
We removed the zero values and explored the data using data using various pandas methods.

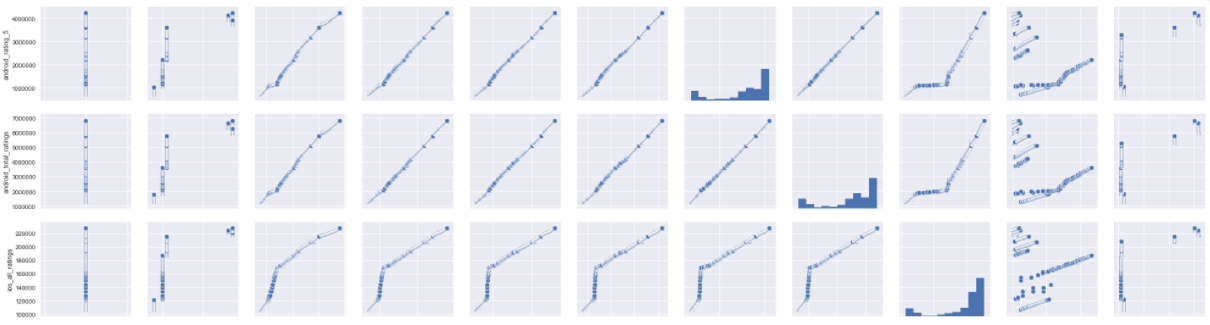
Firstly We used describe() method to compute various statistical coefficients like count, mean, std, min, 25%, 50%, 75%,max values.

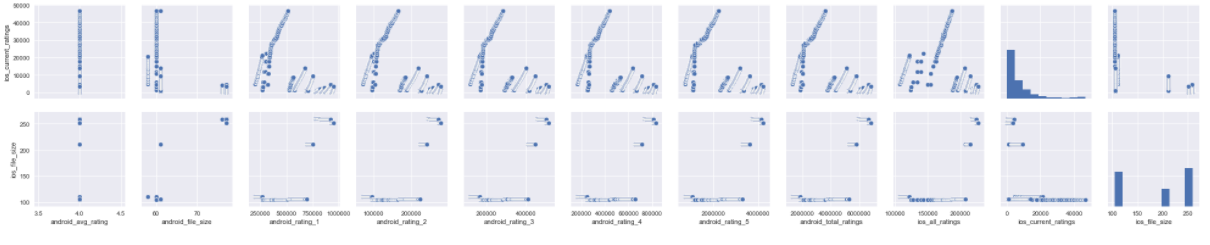
|  | **android\_avg\_rating** | **android\_file\_size** | **android\_rating\_1** | **android\_rating\_2** | **android\_rating\_3** | **android\_rating\_4** | **android\_rating\_5** | **android\_total\_ratings** | **ios\_all\_ratings** | **ios\_current\_ratings** | **ios\_file\_size** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **count** | 11927.0 | 11927.000000 | 11927.000000 | 11927.000000 | 11927.000000 | 11927.000000 | 1.192700e+04 | 1.192700e+04 | 11927.000000 | 11927.000000 | 11927.000000 |
| **mean** | 4.0 | 67.147648 | 693680.590341 | 214149.918169 | 392990.852016 | 628392.217741 | 3.157121e+06 | 5.086335e+06 | 199759.679970 | 7687.831894 | 190.211034 |
| **std** | 0.0 | 8.095310 | 227892.070886 | 62327.847486 | 121505.917883 | 205541.703790 | 1.101980e+06 | 1.717850e+06 | 34338.721179 | 9400.666163 | 67.834794 |
| **min** | 4.0 | 58.000000 | 199974.000000 | 71521.000000 | 117754.000000 | 165956.000000 | 7.265970e+05 | 1.281802e+06 | 106508.000000 | 29.000000 | 104.000000 |
| **25%** | 4.0 | 61.000000 | 590206.000000 | 192919.000000 | 350605.000000 | 562881.000000 | 2.797710e+06 | 4.494321e+06 | 198261.000000 | 1945.000000 | 106.000000 |
| **50%** | 4.0 | 61.000000 | 735380.000000 | 234783.000000 | 436180.000000 | 697004.000000 | 3.521712e+06 | 5.625059e+06 | 213115.000000 | 3810.000000 | 211.000000 |
| **75%** | 4.0 | 76.000000 | 893558.000000 | 263895.000000 | 489883.000000 | 795867.000000 | 4.050571e+06 | 6.493774e+06 | 222378.000000 | 9166.000000 | 258.000000 |
| **max** | 4.0 | 77.000000 | 952604.000000 | 277695.000000 | 514286.000000 | 832416.000000 | 4.239138e+06 | 6.816139e+06 | 227388.000000 | 46692.000000 | 260.000000 |

We plotted a scatter matrix using the seaborn module to analyze the correlation among the variables.









We used the numpy.corrcoef() method to compute the Pearson correlation Coefficient for the most correlated variable pairs.

**ANDROID PAIRS**

* The pearson correlation coefficient between android\_total\_ratings and ios\_all\_ratings is
* 0.9604755216952423
* The pearson correlation coefficient between android\_total\_ratings and android\_rating\_1 is
* 0.9951398570710508
* The pearson correlation coefficient between android\_total\_ratings and android\_rating\_2 is
* 0.9996293142432378
* The pearson correlation coefficient between android\_total\_ratings and android\_rating\_3 is
* 0.9995424019937938
* The pearson correlation coefficient between android\_total\_ratings and android\_rating\_4 is
* 0.9996991316272775
* The pearson correlation coefficient between android\_total\_ratings and android\_rating\_5 is
* 0.9998643381812596

**IOS PAIRS:**

* The pearson correlation coefficient between android\_rating\_1 and ios\_all\_ratings is
* 0.9496420792168682
* The pearson correlation coefficient between android\_rating\_2 and ios\_all\_ratings is
* 0.9660995431490733
* The pearson correlation coefficient between android\_rating\_3 and ios\_all\_ratings is
* 0.9605838249874092
* The pearson correlation coefficient between android\_rating\_4 and ios\_all\_ratings is
* 0.9597710879182769
* The pearson correlation coefficient between android\_rating\_5 and ios\_all\_ratings is
* 0.9612990969745802

**MULTI COLLINEARITY**

* The pearson correlation coefficient between android\_rating\_2 and android\_rating\_1 is
* 0.9943617553504197
* The pearson correlation coefficient between android\_rating\_3 and android\_rating\_1 is
* 0.9927147838563694
* The pearson correlation coefficient between android\_rating\_4 and android\_rating\_1 is
* 0.9934178134778145
* The pearson correlation coefficient between android\_rating\_5 and android\_rating\_1 is
* 0.9935052395308394
* The pearson correlation coefficient between android\_rating\_1 and android\_rating\_2 is
* 0.9943617553504197
* The pearson correlation coefficient between android\_rating\_3 and android\_rating\_2 is
* 0.9994770661066952
* The pearson correlation coefficient between android\_rating\_4 and android\_rating\_2 is
* 0.9993796712015057
* The pearson correlation coefficient between android\_rating\_5 and android\_rating\_2 is
* 0.9994932223811953
* The pearson correlation coefficient between android\_rating\_1 and android\_rating\_3 is
* 0.9927147838563695
* The pearson correlation coefficient between android\_rating\_2 and android\_rating\_3 is
* 0.9994770661066952
* The pearson correlation coefficient between android\_rating\_4 and android\_rating\_3 is
* 0.9998846110151156
* The pearson correlation coefficient between android\_rating\_5 and android\_rating\_3 is
* 0.9995760706359129
* The pearson correlation coefficient between android\_rating\_1 and android\_rating\_4 is
* 0.9934178134778144
* The pearson correlation coefficient between android\_rating\_2 and android\_rating\_4 is
* 0.9993796712015056
* The pearson correlation coefficient between android\_rating\_3 and android\_rating\_4 is
* 0.9998846110151156
* The pearson correlation coefficient between android\_rating\_5 and android\_rating\_4 is
* 0.9996717136848663
* The pearson correlation coefficient between android\_rating\_1 and android\_rating\_5 is
* 0.9935052395308392
* The pearson correlation coefficient between android\_rating\_2 and android\_rating\_5 is
* 0.9994932223811953
* The pearson correlation coefficient between android\_rating\_3 and android\_rating\_5 is
* 0.9995760706359128
* The pearson correlation coefficient between android\_rating\_4 and android\_rating\_5 is
* 0.9996717136848663

Finally, Time series for the variables was plotted using Matplotlib. (Code is Erroneous for this part)

# Prediction And Modelling

We developed two prediction models for the android and ios parameters. For this we have used Linear Regression, Ridge and Lasso from Scikit Learn to predict the values of android\_total\_ratings and ios\_all\_ratings. We have used cross validation for splitting the data into training and testing sets.

**FEATURE EXTRACTION:**

For Feature extraction the key point to note down is the correlation between the value to be

predicted and the regressors.

For linear regression we have considered android\_rating\_5 as feature to predict the

android\_total\_ratings and android\_rating\_2 to predict ios\_all\_ratings as these were the most correlated pairs.

The Results are as follows:

**SIMPLE LINEAR REGRESSION**

**Android:**

* The model coefficient for android model is [ 1.55866423]
* The model intercept for android model is 165442.74062972516
* The model's prediction for android\_all\_ratings for the given feature values is [ 1297963.49428286 1307875.04012255 6771055.51345835 6772835.50800921]
* The R squared value for the model is 0.9997286947666499

**IOS:**

* The model coefficient for ios model is [ 0.02995497]
* The model intercept for ios model is 105188.19836224435
* The model's prediction for ios\_all\_ratings for the given feature values is [ 126953.39029279 232171.45344044]
* The R squared value for the model is 0.9240959538441424

**MULTIPLE LINEAR REGRESSION:**

**ANDROID:**

* The model intercept for android multiple linear regression model is -46958.326166865416
* The model coefficient for android multple linear regression is [ 0.83604164 5.55977744 -7.75402276 9.56093211]
* The model's prediction for android\_total\_ratings for the given feature values is [ 6608956.6316948 6182336.75157377 5676723.20942161 ...,6375911.77307799 6656344.56644049 4530851.05393698]
* The R squared value for the model is 0.9998765572266578
* The root mean sqaured error is 19114.36602890759

**IOS:**

* The model intercept for ios multiple linear regression model is 17316.572834887833
* The model coefficient for ios multple linear regression is [-0.04495736 -0.36734551 -0.49361975 3.49848113]
* The model's prediction for ios\_all\_ratings for the given feature values is [ 224394.33163055 120316.63862326 198959.51737507 ..., 190529.38554138 214948.46912919

203417.27513769]

* The R squared value for the model is 0.9607756756129163
* The root mean sqaured error is 6788.435658973464

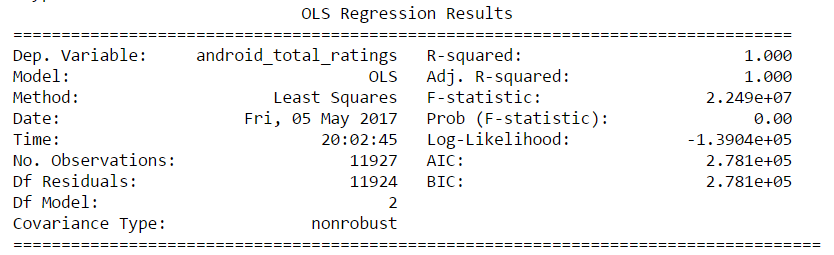
**REGRESSION USING STATS MODEL:**

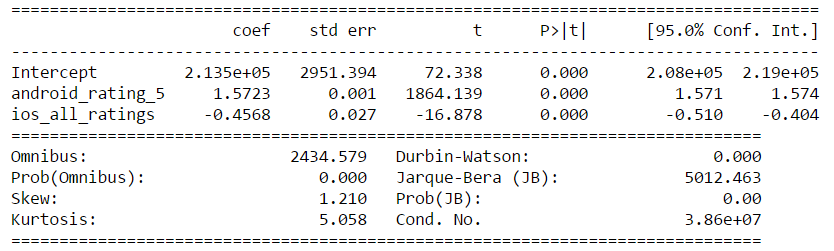
Intercept 213497.789317

android\_rating\_5 1.572349

ios\_all\_ratings - 0.456848

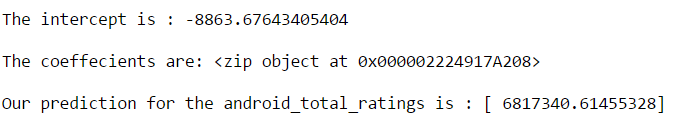
dtype: float64



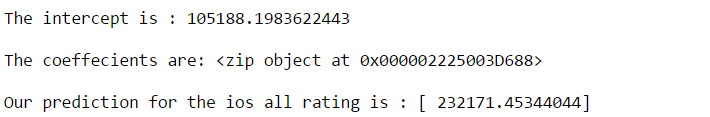


From the above results we predict the values android\_total\_ratings and ios\_all\_ratings of for the date ios\_all\_ratings and android\_total\_ratings for 2016/11/01 11:50 PM.

**ANDROID:**



**IOS:**



# Deep Learning

We have used beautiful soup to collect the snapshot URLs from the HTML files. There are 5 URLs in each file. The URL’s were collected and appended to 2 separate lists (Android and IOS) in order to

Identify the unique snapshot URLs. Finally We, have pickled the lists to for two separate files.

We have used the URLLIB module to extract the top 5 unique Snapshots from the App store and

Play store pages.

The Results are as follows:

**ANDROID:**







**IOS:**









# Inferences

From Our analysis we have computed the values of android total ratings and ios all ratings for 1st November 2016 at 11.50pm.

* Android - 6817340.61455328
* IOS - 232171.45344044

From Deep Learning we found that Charmander is the most popular demon of the pokemon go app.