Northeastern University
College *of* Professional Studies

ALY 6140 – 21018

Analytics Systems Technology

Richard He

CAPSTONE PROJECT REPORT

UNEMPLOYMENT STATISTICS BY ETHNICITY, GENDER AND REGIONS IN THE UK FROM 2004 TO 2018

By:

HOANG UYEN LE

NORTHEASTERN UNIVERSITY COLLEGE OF PROFESSIONAL STUDIES

February 14th, 2020.

1. Project topic

1.1 Unemployment statistics by ethnicity, gender and regions in UK between 2004 and 2018

All we acknowledge, at the end of January 2020, Britain officially exited Europe. Thousands of politicians, economists and socialist forecasted that this event will create a huge affect (positive or negative) on the social, labor force and economic fields for both sides Britain and Europe.

There are many factors that affect the unemployment rate; in macroscope, it may be the volatility of the economy or the intervention/salvage of the government; in microscope, it may depend on age, gender, education level of the employee and local regional economy. In this capstone, I will determine the unemployment trend of UK over the last 15 years as well as which and how demographic elements have affected unemployment in the UK over the last 15 years.

1.2 Dataset content

- Data source: **Unemployment by ethnicity**
  https://data.gov.uk/dataset/fe6c83aa-62aa-4a8c-94cc-225f47287225/unemployment-by-ethnicity
- Dataset dimension: 35,100 x 15
  Containing statistic on the percentage of working age people (from 16 to 64 years old) who are unemployed, broken down by ethnicity type, gender, age range from thousands investigations in different regions of England from 2004 to 2019.
- Variable contents:

1 - MEASURE	1 unique value: "Unemployed"
2 – MEASURE_TYPE	1 unique value "Percentage of individual unemployed"
3 - ETHNICITY	13 unique ethnicity types
4 – ETHNICITY_TYPE	4 unique ethnicity classifications
5 - TIME	15 unique year values (from 2004 to 2019)
6 – TIME_TYPE	1 unique value "Year"
7 - REGION	12 unique name of region in UK
8 - AGE	5 unique age ranges: 16 – 24, 25 – 49, 50 – 64, 65+, All
9 – AGE_TYPE	1 unique value: " 16+ "
10 - GENDER	3 unique genders: Women, Men, All
11 - VALUE	Percentage of individual unemployed in the investigation
12 – CONFIDENCE_INTERVAL	Confidence interval of "Value" variable
13 - NUMERATOR	Numerator of the equation to calculate unemployed percentage
14 -DENOMINATOR	Denomination of the equation to calculate unemployed percentage
15 – SAMPLE SIZE	Sample size of the investigation

2. Data cleansing stage

2.1 Dropping columns

	feature	number of unique
0	Measure	1
1	Measure_type	1
2	Ethnicity	13
3	Ethnicity_type	4
4	Time	15
5	Time_Type	1
6	Region	12
7	Age	5
8	Age_Type	1
9	Sex	3
10	Value	437
11	confidence_interval	210
12	Numerator	1927
13	denominator	7232

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 35100 entries, 0 to 35099
Data columns (total 11 columns):
Ethnicity              35100 non-null object
Ethnicity_type         35100 non-null object
Time                   35100 non-null int64
Region                 35100 non-null object
Age                    35100 non-null object
Sex                    35100 non-null object
Value                  35100 non-null object
confidence_interval    35100 non-null object
Numerator              35100 non-null object
denominator            35100 non-null object
samp_size              35100 non-null object
dtypes: int64(1), object(10)
memory usage: 2.9+ MB
```

First of all, I count unique values in each column and the results came out with some columns only contain 1 repeated value. These columns provide us more detailed understanding about the dataset, but not useful at all for analytics process, so I drop them via a function. This function will loop over every column of dataset and if it satisfies the condition that there is only one unique values in it, that column will be removed away from the dataset. After applying this function, 4 columns named “Measure”, “Measure_type”, “Time_type” and “Age_type” are not in the dataset anymore, make my data neat and easier to generate with 11 columns remained.

```
def dropping_column(dataset):
    columns = list(dataset)
    for i in columns:
        if len(dataset[i].unique()) == 1:
            dataset.drop(i,inplace=True,axis=1)
```

2.2 Converting data type

In the original dataset, all variable but “Time” are defined in “object” datatype even when some columns are supposed to be numeric, such as “Value” or “samp_size”. Converting data into proper data type is necessary in data cleaning process. With some strings that could not be converted into numerical type such as question marks (?) or sticks (-), they will be replaced by NULL value to be suitable with the converting function. This function apply for numerical variables named: Value, confidence_interval, Numerator, denominator and sample size.

```
def replacing_convert(dataset, column_list):
    for i in column_list:
        dataset[i] = dataset[i].apply(lambda x: np.nan if x == '?' else x)
        dataset[i] = dataset[i].apply(lambda x: np.nan if x == '-' else x)
        data[i] = data[i].astype(str).astype(float)
```

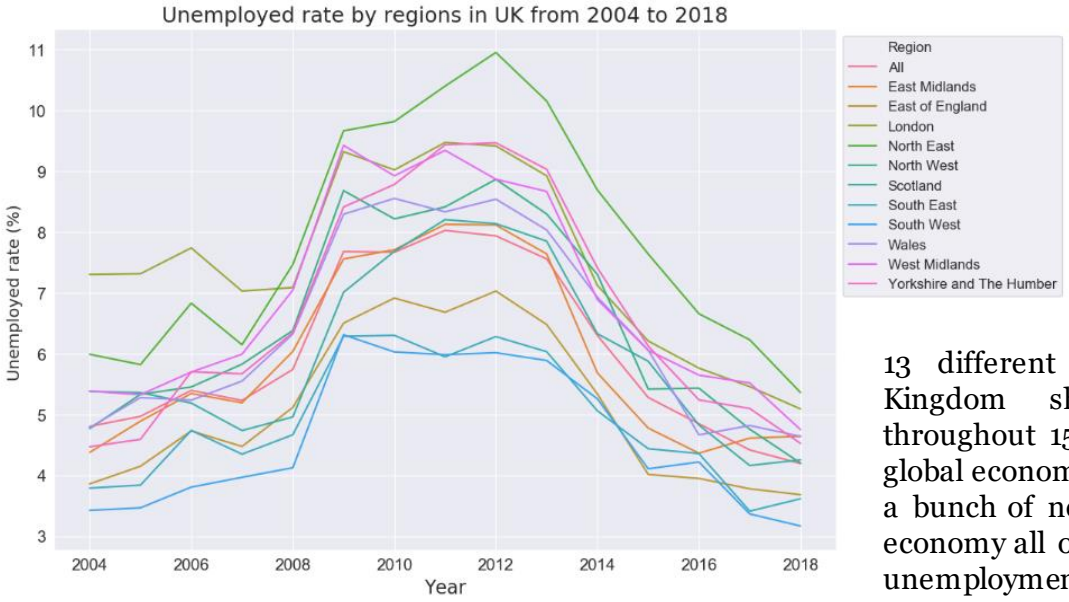
2.3 Missing values.

More than 50% out of the number of rows contains missing values, almost in numeric variable, and it could not be acceptable to delete these rows away, cut off 50% of the dataset will influence the accuracy of analytics result. Therefore, my solution for this case is to replace missing value by average values which are calculated from available numbers in the dataset. However, because the ratio of missing value is significantly huge, it will create a bias in the result if all missing values are replaced by one same average value. In order prevent this scenario occurring, I groupby() some demographic variables (Region, Gender, Age, Year) before calculating the average values. The underneath function of this steps is to create some smaller cluster of demographic and each cluster obtains it own average value, so that when filling these average ones into missing values, the variety of data is keep remained and it keep the data in normal distribution (without bias). I generate this idea on numerical value named: confidence_interval, Numerator, denominator and sample size. The percentage of unemployment is manually calculated by divide Numerator over Denominator variables.

```
def filling_value(input_data, output_data, groupby_list, fill_list):
    for i in list(fill_list):
        average_values = pd.DataFrame(input_data.groupby(list(groupby_list))[i].mean()).reset_index()
        output_data = output_data.merge(average_values, on = groupby_list, how="left")
groupby_list = ('Time', 'Region', 'Age', 'Sex')
fill_list = list(data[['samp_size', 'Numerator', 'denominator']])
filling_value(full, blanks, groupby_list, fill_list)
```

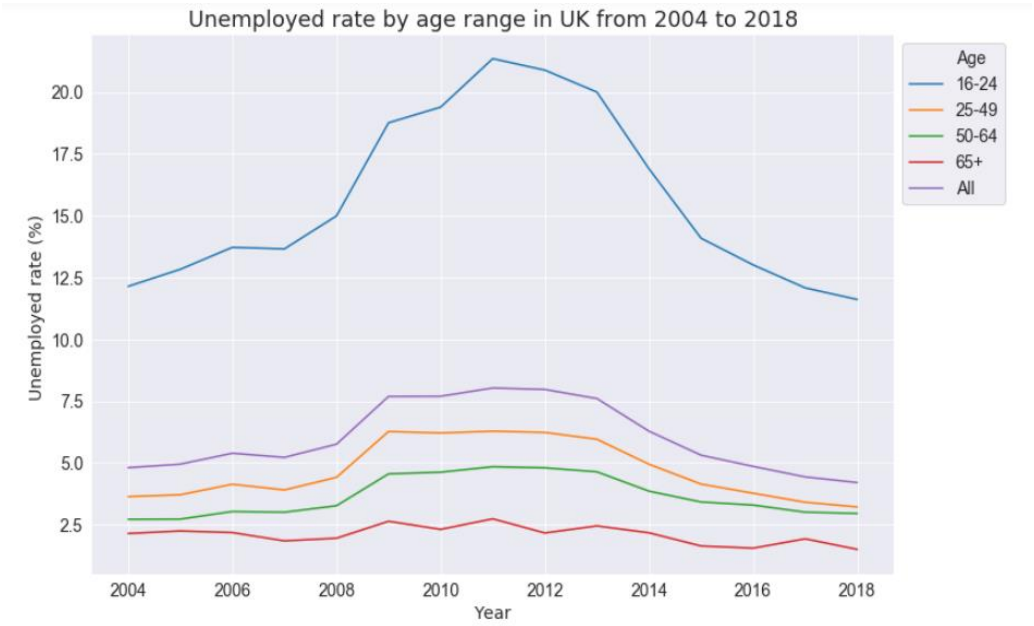
After this step, my dataset now is full, proper and clean to be ready for further analysis steps.

3. Data visualization

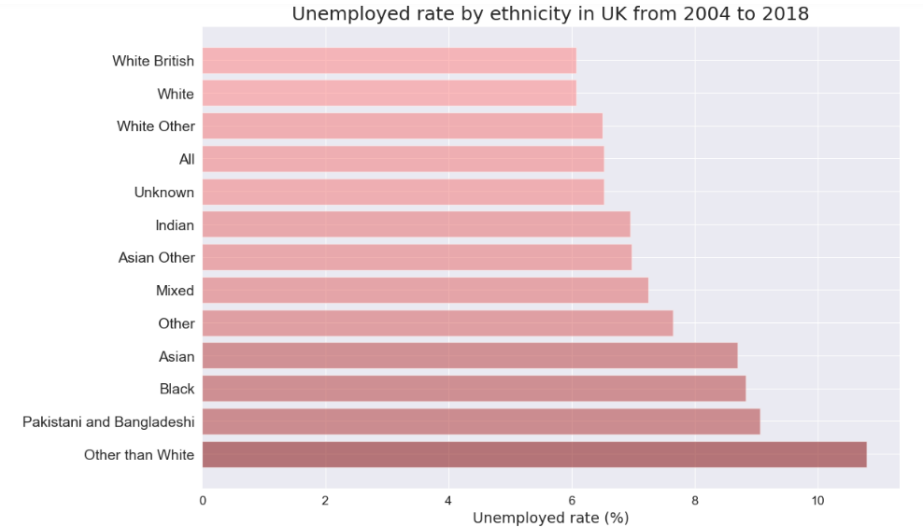


13 different regions in the United Kingdom share similar pattern throughout 15-year period. In 2008, global economic crisis broke out, led to a bunch of negative influences to the economy all over the world, including unemployment rate. This rate was almost doubled immediately after a year and keep hovering around that rate for the next 4 years. Until to 2013, this dark situation in labor field was gradually improved, and the lowest rate of unemployed was recorded at the bottom peak in 2018.

London and North East are witnessed to be regions obtain highest percentage of unemployed over the whole period, while South East and South West got the lowest rate in comparison.



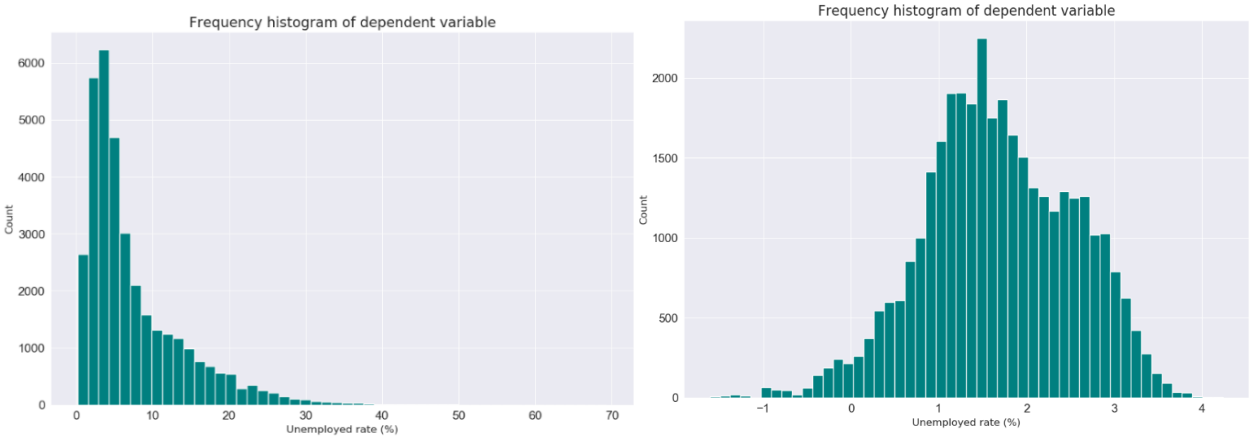
In term of working age, youngest range (16-24) is considered as the most unsuccessful on to get a job. This consequence is caused by some specific reasons, such as they pursue higher education or stick on a part-time job and then travelling instead of getting an official career like other age group. In economy global crisis, this blue line noticeably rocketed, almost 21% at the peak in 2011.



In term of ethnicity, White people get the lower unemployment rate than other group. By contract, Other than White, Pakistani and Bangladeshi is witnessed as highest rate group. However, the gap between bottom and top group in this graph is not significant, so ethnicity is assumed to less affect the percentage of unemployed.

4. Descriptive/ Predictive analysis

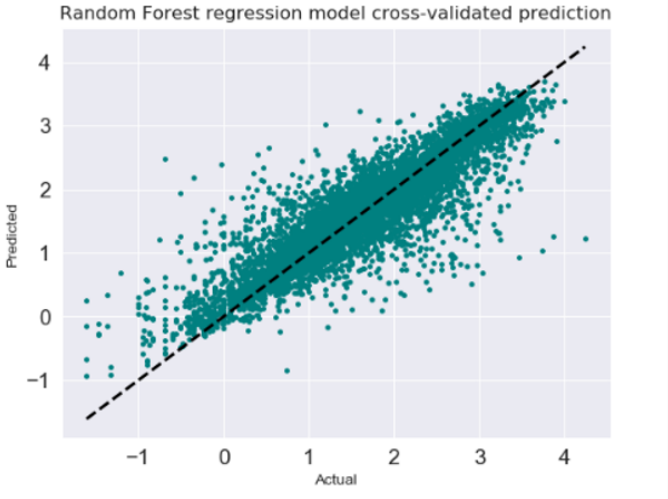
Depending on this dataset, the percentage of unemployed individuals in UK population will be predicted by some categorical varibles: Ethnicity, Age range, Region, Year and Gender.



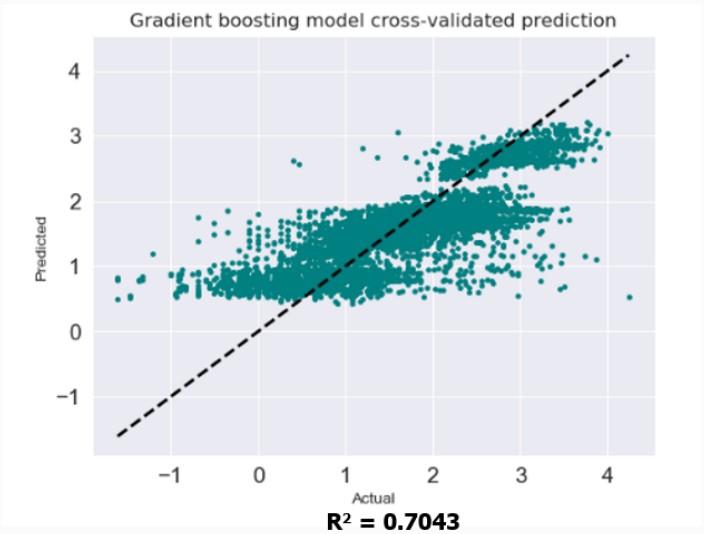
Above are the histogram of distribution of Dependent variable (Value). Graph on the left shows its original distribution, which as be seen clearly that non-normal and highly positive skew with a long tail expending to the right. This feature will effect the accuracy of predictive model, so log10 function is used to normalized this variable (shows in graph on the right).

In addition, all categorical variables are converted into dummies variables with 0/1 value. After this step, Value column is now predicted by 29 other binary variables. I split dataset into train and test set with the ratio 70:30

Regression models are suitable the most for numerical value prediction. This time, I choose Random forest and Gradient boosting to build models.

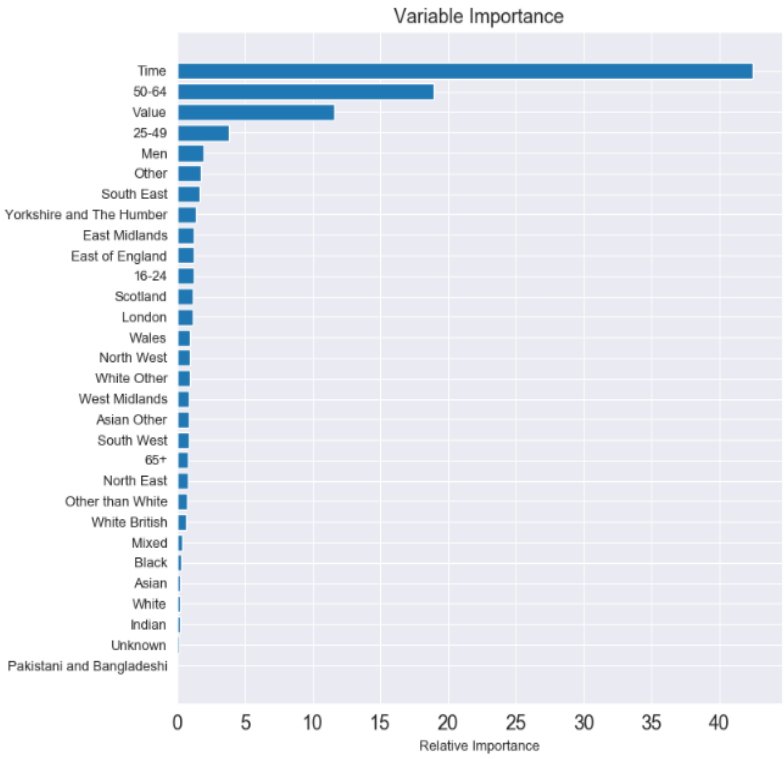


R^2 = 0.8429
MAE = 0.1994
MSE = 0.1146
RMSE = 0.3385



R^2 = 0.7043
MAE = 0.352
MSE = 0.2157
RMSE = 0.4644

In conclusion, Random forest regression model do a better job this time, since all the accuracy scores (R^2, MAE, MSE, RMSE) is higher than those of Gradient boosting.



To explore which predictor variables having the largest influence on the tendency unemployed in UK labor force, I create plot of Variable importance. As can be witnessed clearly, some variables which be considered as affect unemployment rate the most are year, age range and regions. Ethnicity types are ranked down way of the list, it means that the ethnicity individuals belong to does not affect the chance they get a job offer.

Moreover, in this dataset, only basic personal information of employee is mentioned. The unemployment rate also gets influence by academic levels, the condition of global and domestic economy. The variable 'Year ' stands at the first place in variable importance list prove that labor force scores (employed/unemployed and other rates) are highly stick on the global economy changes.