Dublin, on the 6th of August 2023

CCT College Dublin
Master of Science (MSc) in Data Analytics
Subjects:
Programming for Data Analytics
Statistics for Data Analytics
Machine Learning for Data Analytics
Data Preparation and Visualisation

Master of Science (MSc) in Data Analytics

CA 3 Repeat Programming for Data Analytics

Author: Geomar Munoz Quintero
E-mail: 2014222@student.cct.ie
Student ID: 2014222

ABSTRACT

Using methods like data analysis, statistics, and machine learning, this study aims to provide an analysis of Ireland's Public Sector Employment and Earnings. In connection to other variables like comparing the Public Sector Employment with other countries worldwide.

In addition, a study comparing Public Sector Employment with other countries worldwide was created. A forecasting, sentiment analysis and evidence-based recommendations for the sector as well as a complete rationale of the entire process used to discover your findings with Ireland as your baseline. The conclusion also includes plans for potential future additions to this work.

This CA3 Repeat Report in Programming for Data Analytics will focus primarily to show Programming, Data structures, Documentation, Testing and Optimisation.

INTRODUCTION

The main goal of this study will be to provide a comprehensive overview of the public sector employment landscape in Ireland while using it as the starting point for comparisons with other countries. It will identify the difficulties, chances, and possible growth areas in the industry by carefully reviewing the data and applying cutting-edge analytical methods.

Sentiment analysis will provide insight into sentiments and opinions regarding public sector employment in Ireland beyond the quantitative data. It can discover public sentiments, pinpoint areas of strength and vulnerability, and determine the general degree of satisfaction with the industry by utilising the power of natural language processing.

Python will be the programming language used due to a number of benefits, including its simple coding, which enables solutions to be produced with fewer lines of code than with other programming languages. The ability to automatically identify and associate various types of data with one another, as well as the fact that Python has no information processing restrictions, enable it to be used in a wide range of environments and devices, including desktop and cloud computing environments.

The aggregate functions in Python are crucial for processing and changing data from different data structures. For this project, Numpy and Pandas have been used, which are required libraries for data purification. These are the libraries that are typically employed in accordance with accepted standards in order to carry out mathematical and statistical operations throughout a data analysis process.

To meet assessment requirements and execute appropriate Machine Learning algorithms to produce and convey insights, I have utilised Python as the main programming, as well as, datasets in the CSV and JSON formats.

DATA ANALYSIS REPORT

For this study, the following datasets were analysed:

- EHQ10.20230721095207.csv

(https://data.gov.ie/dataset/ehq10-public-sector-employment-and-earnings/resource/062d0798-b666-407f-b72d-077f989bc057?inner_span=True, 2023)

- HOW_TEMP_SEX_ECO_NB_A-full-2023-07-24.json

(https://www.ilo.org/shinyapps/bulkexplorer57/?lang=en&id=HOW_TEMP_SEX_ECO_NB_A, 2023)

RESULTS AND DISCUSSION

This section's objective is to present an overview of the analysis that was done along with its key points.

Ireland's Public Sector Employment and Earnings

This investigation involved looking through the datasets to gain some preliminary understanding of Ireland's Public Sector Employment and Earnings and to find a viable dataset for future work using machine learning models.

The Jupyter Notebooks **2014222-GeomarMunoz CA-03-Repeat.ipynb** will show more in detail the coding behind the scenes for this project.

Python will be used for the Data Cleaning phase, visualisations, and Machine Learning algorithms implementation.

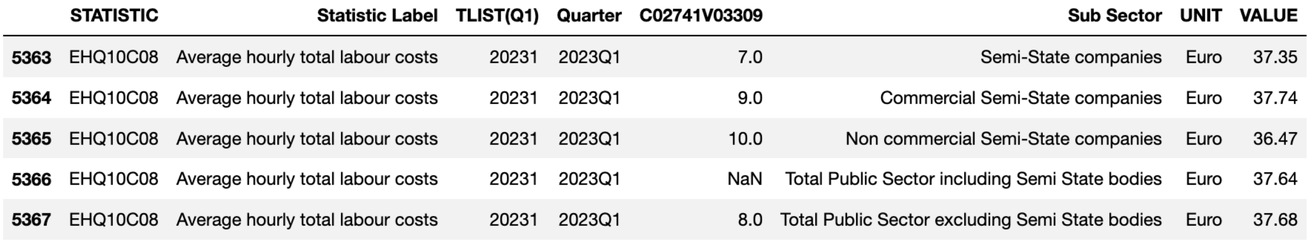
On the other hand, SQL will be utilised during the Exploratory Data Analysis phase. This choice is motivated by SQL being a more efficient language for simple queries and aggregations.

In addition, SQL is a specialised language for data manipulation, meaning it has a broader range of built-in functions for working with data, such as joins, aggregations, and subqueries.

Also, SQL makes it easier to write complex queries, can be faster than Python, and works better with big datasets. Remember that the data set picked for the second part is quite extensive.

Graphs and visualization were implemented using the **Seaborn library. Seaborn has been chosen over Matplotlib because** of its visually appealing style and diverse functions that can be implemented in a much smoother fashion compared to Matplotlib. Furthermore, Seaborn is a lot easier to customise thanks to its built-in themes and styles and it meets the need to display a static chart that is visually appealing while producing an image file that occupies less space compared to Plotly.

DATA CLEANING



	STATISTIC	Statistic Label	TLIST(Q1)	Quarter	C02741V03309	Sub Sector	UNIT	VALUE
5363	EHQ10C08	Average hourly total labour costs	20231	2023Q1	7.0	Semi-State companies	Euro	37.35
5364	EHQ10C08	Average hourly total labour costs	20231	2023Q1	9.0	Commercial Semi-State companies	Euro	37.74
5365	EHQ10C08	Average hourly total labour costs	20231	2023Q1	10.0	Non commercial Semi-State companies	Euro	36.47
5366	EHQ10C08	Average hourly total labour costs	20231	2023Q1	NaN	Total Public Sector including Semi State bodies	Euro	37.64
5367	EHQ10C08	Average hourly total labour costs	20231	2023Q1	8.0	Total Public Sector excluding Semi State bodies	Euro	37.68

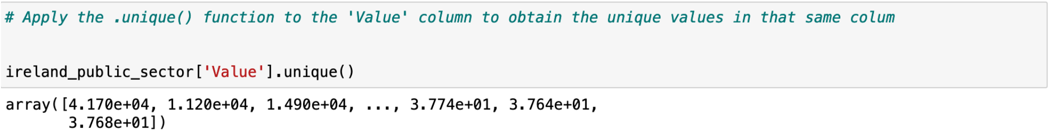
Figure 1. Table - Ireland Public Sector (raw)

It can be observed that:

The column names are almost all capitalised. Therefore, they will be set to title case; the 'STATISTIC', 'TLIST(Q1)', and 'Quarter' columns contain information about years and quarters.

Therefore, they will be explored; the 'Statistic Label' column contains different types of categories that don't fall within the same category, for instance, 'Employment' and 'Average hourly total labour costs'; it is still uncertain what the 'C02741V03309' column contains.

Therefore, it requires further exploration; the 'UNIT' column contains values that don't fall within the same category, for instance, 'Euro' and 'Number', and this could be explained by the values contained in the 'Statistic Label' column; the 'Value' column contains a wide range of values.



```
# Apply the .unique() function to the 'Value' column to obtain the unique values in that same colum


ireland_public_sector['Value'].unique()
```

```
array([4.170e+04, 1.120e+04, 1.490e+04, ..., 3.774e+01, 3.764e+01,
       3.768e+01])
```

Figure 2. Applying the .unique() function to the 'Value' column

The 'Value' column seems to contain a wide range of values. These values don't require any renaming or correction. However, this column will be visually explored to understand the data distribution.

At this stage, it is necessary to separate the years and the quarters contained in the 'Tlist(Q1)' column. To do so, the fastest and most straightforward method that has been found was:

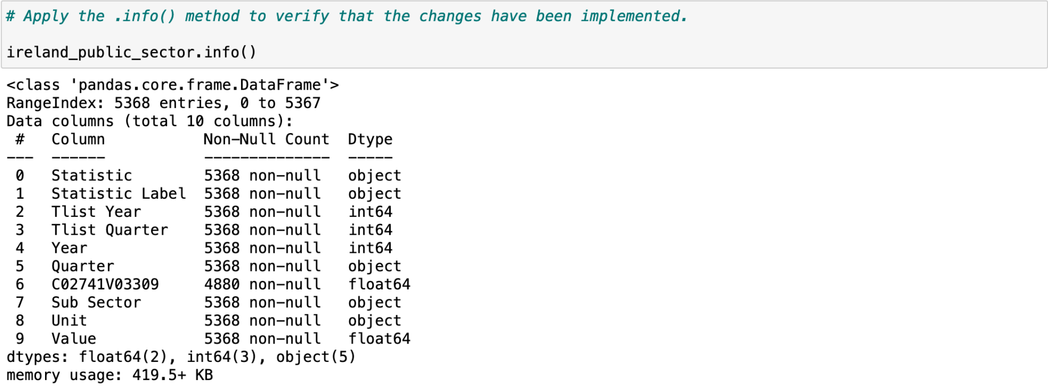
Converting the column from numerical to categorical;

slicing the content of each row to separate years and quarters.

This method has been chosen over a function thanks to its simplicity and effectiveness. Another option could have been using the floor() function, which would have required two steps. Another possible method would have been converting the 'Tlist(Q1)' column to a timestamp.

However, this conversion would have caused the values to be presented containing the number of seconds. Therefore, it would have made the values harder to interpret. Converting to DateTime could also have been an option.

Nevertheless, the values that are going to be used are not going to be used for date and time calculations. These are why, although not as efficient as other methods, the 'str' method has been chosen.



```
# Apply the .info() method to verify that the changes have been implemented.

ireland_public_sector.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5368 entries, 0 to 5367
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   Statistic        5368 non-null   object 
 1   Statistic Label  5368 non-null   object 
 2   Tlist Year       5368 non-null   int64  
 3   Tlist Quarter    5368 non-null   int64  
 4   Year             5368 non-null   int64  
 5   Quarter          5368 non-null   object 
 6   C02741V03309     4880 non-null   float64
 7   Sub Sector       5368 non-null   object 
 8   Unit             5368 non-null   object 
 9   Value            5368 non-null   float64
dtypes: float64(2), int64(3), object(5)
memory usage: 419.5+ KB
```

Figure 3. Applying the .info() method to verify that the changes have been implemented.

It has been observed that both the 'Tlist Year' column and the 'Year' column contain years, while both the 'Tlist Quarter' and the 'Quarter' columns contain quarters. However, it is still unclear whether the years in the 'Tlist Year' and the 'Year' columns match.

It is still unclear whether the quarters in the 'Tlist Quarter' and 'Quarter' columns match. To compare the 'Tlist Year' and the 'Year' column, the most straightforward and effective method found is the .equals() method, given that both columns contain numeric values and might contain the same value in the same row.

EXPLORATORY DATA ANALYSIS

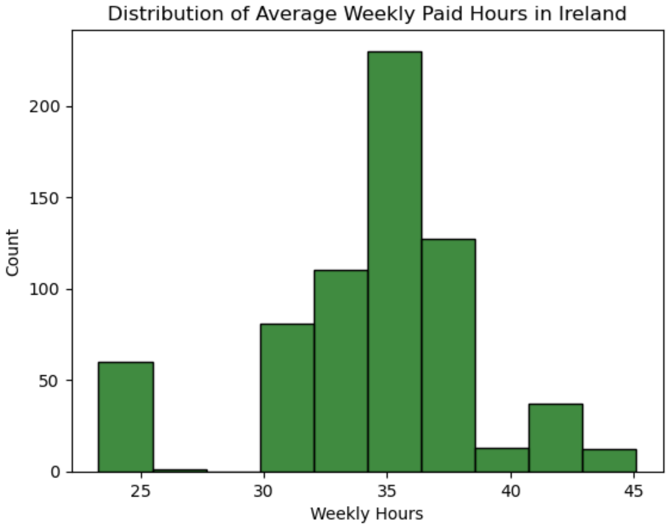


Figure 4. Graph - Distribution of Average Weekly Paid Hours in Ireland

The histogram shows the distribution of the average weekly paid hours in Ireland. The lowest number of paid hours, on average, is around 23, while the highest number of paid hours is around 45. The most common number of paid hours is around 34-36 There are fewer people who work 23-25 hours or 38-45 hours per week.

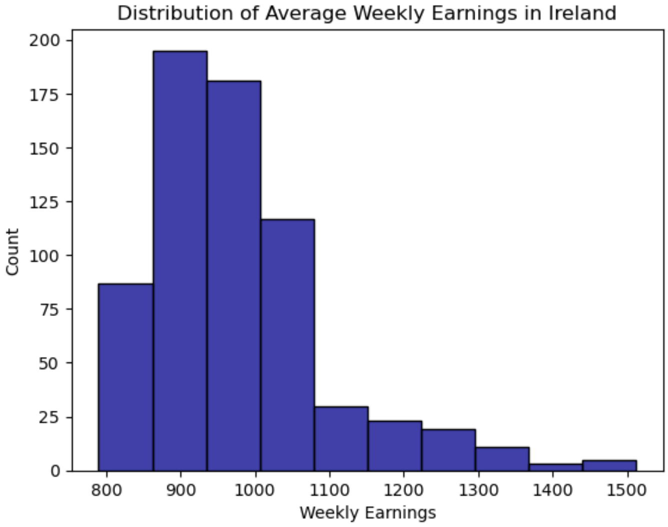


Figure 5. Graph - Distribution of Average Weekly Earnings in Ireland

The histogram shows the distribution of average weekly earnings in Ireland. The distribution is right-skewed and not normally distributed. This suggests that there are a few people who earn significantly more than the average, but definitely a considerable amount of people who earn from approximately less than €900 to approximately €1000 per week.

The mean of the distribution is approximately €1,000. This means that half of the people in the distribution earn less than €1,000 and half of the people earn more than €1,000 per week.

The minimum value in the distribution is approximately €800. This means that there are some people who earn less than €800 per week.

The maximum value in the distribution is approximately €1500. This means that there are some people who earn more than €1500 per week, even though little in number.

Formulate and evaluate testing and optimisation strategy for programmatic solutions.

For the purpose of the next section of this analysis (Machine Learning), an exploration of the weekly earnings per year and sub-sector will be conducted. In order to plot this result, the Plotly library will be utilised. This decision is motivated by the need to examine different earnings per sub-sector each year. Therefore, an animated plot is a solution that best helps in visualising this result, and Plotly is the visualisation library that allows the creation of such a chart. In addition, Plotly is easier to use compared to Matplotlib, it supports a wider range of features, and it's more widely supported.

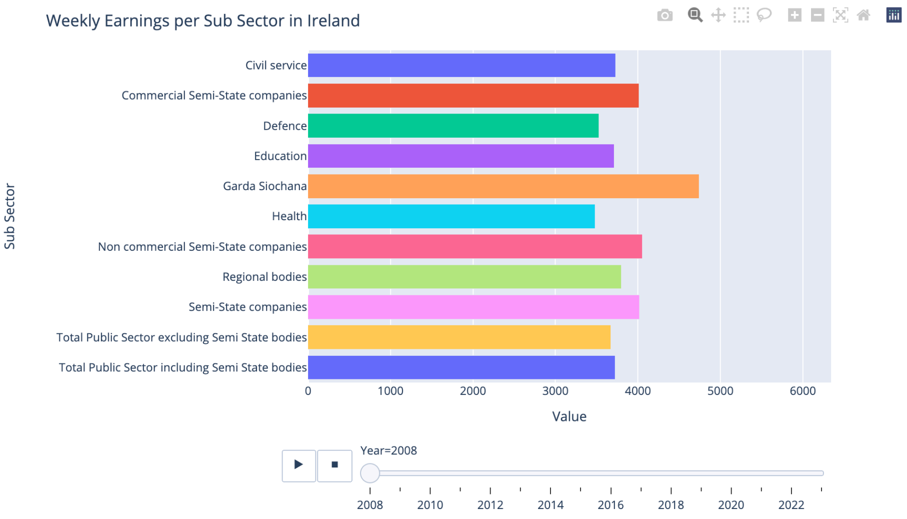


Figure 6. Graph - Weekly Earnings per Sub Sector in Ireland

It appears evident that, throughout the recorded years (2008 - 2023), the Garda Siochana sub-sector recorded the highest amount of earnings. The next step will be to focus on the weekly earnings in this sub-sector.

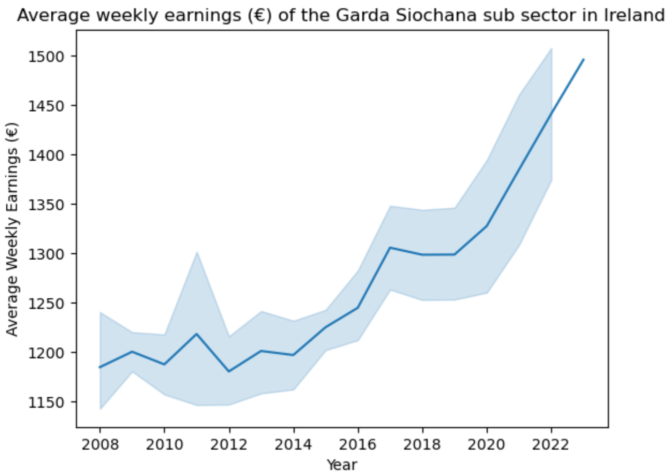


Figure 7. Graph - Average weekly earnings (€) of the Garda Siochana sub-sector in Ireland

The relationship between the average weekly earning and the years is not a perfect linear relationship, however, there seems to be a positive correlation between years passing by and average weekly income in the Garda Siochana sub-sector. This relationship will be used to apply a regression Machine Learning algorithm in the following section of this analysis.

MACHINE LEARNING

In this section, for the purpose of this assignment, the following algorithms have been chosen:

- Random Forest Regression
- Sentiment Analysis

Random Forest Regression has been chosen over another regressor because:

- The relationship between 'Year' and 'Value' is not perfectly linear
- Random Forest is a robust algorithm
- It is more interpretable and faster than other algorithms used for non-linear relationships, such as Support Vector Machine

Sentiment Analysis has been chosen given that it is required by the assignment.

RANDOM FOREST REGRESSION

Plot distributions and correlations by utilising a **Seaborn pairplot**. A pairplot allows to visualise both distribution and correlation of the selected columns, and Seaborn has been chosen over other libraries such as Matplotlib or Plotly Express because Seaborn is easier to utilise, compared to the other presented libraries; it is highly customisable, unlike Matplotlib, which also requires more complex coding, thus making the code harder to interpret. Furthermore, **Plotly Express** would have been a possible option for creating an animated plot, but not as performing as Seaborn when creating a static plot like the pairplot that is needed.

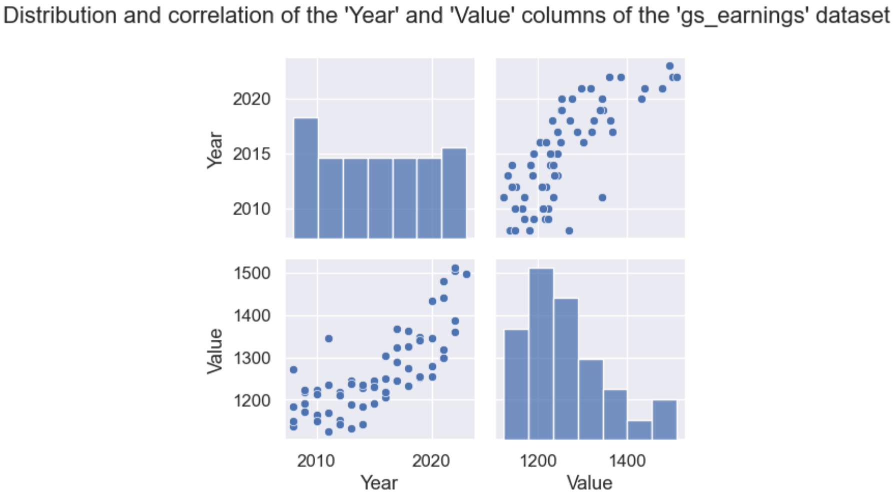


Figure 8. Graph - Average weekly earnings (€) of the Garda Siochana sub-sector in Ireland

It can be observed that the relationship between 'Year' and 'Value' presents a positive correlation but it's not perfectly linear. A correlation matrix will be plotted in order to obtain the correlation coefficients.

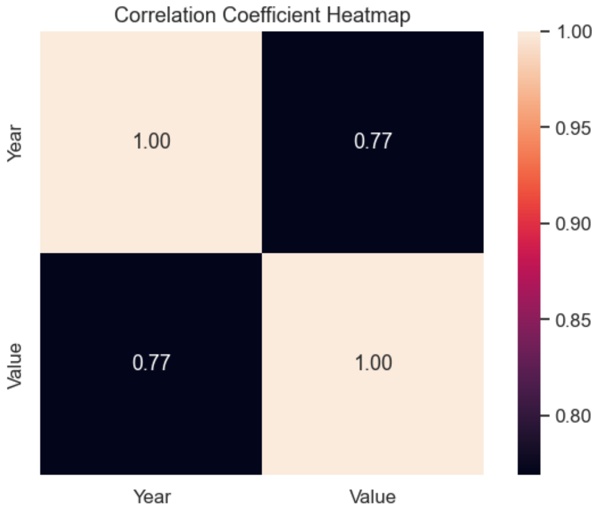


Figure 9. Graph - Correlation Coefficient Heatmap

There's a 0.77 correlation coefficient between the 'Year' column and the 'Value' column, which classifies the correlation as a strong positive correlation.

Creating a scatter plot by applying the sns.scatterplot() function of the Seaborn library. This is a standard function used to plot a scatterplot with Seaborn, which has been chosen due to its straightforwardness and high level of customisation. Pass the 'gs_earnings' as the data frame from which to retrieve the data to plot. Pass the 'Year' column as the 'x' value, and the 'Value' column as the 'y' value

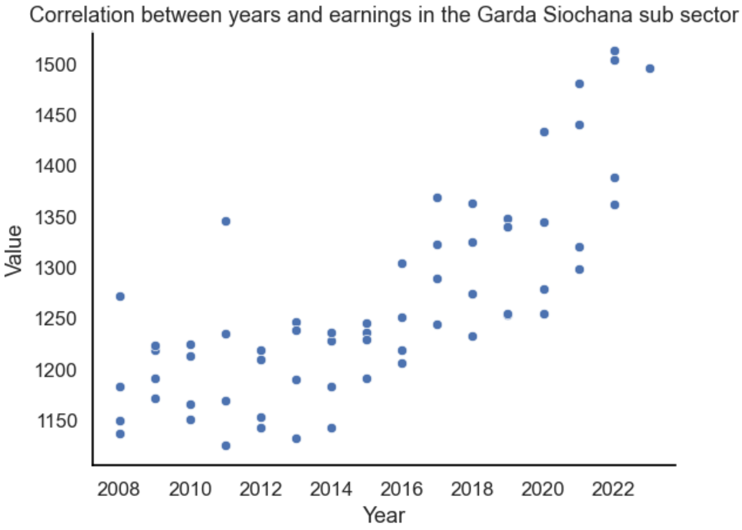


Figure 10. Graph - Correlation between years and earnings in the Garda Siochana sub-sector

Creating a scatter plot to visualize the residuals (the differences between the predicted values and the actual values) for both the training data and the test data. For this purpose, Matplotlib has been chosen because the purpose was to create two static plots in one figure which should ideally occupy little space.

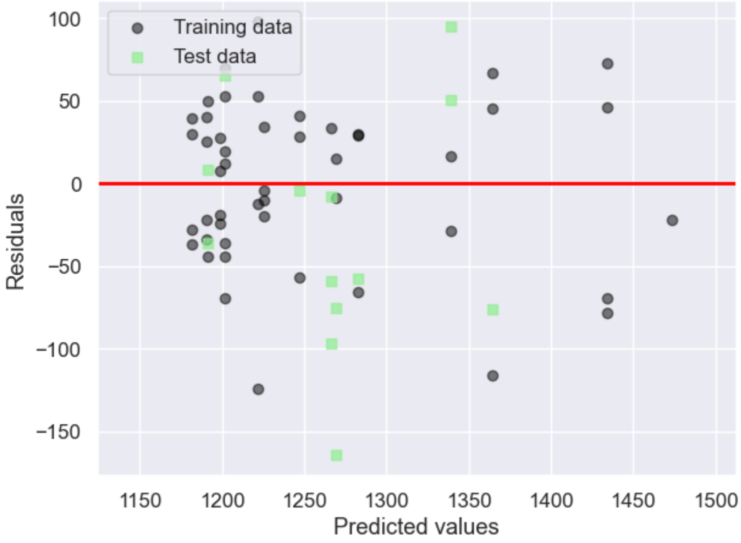


Figure 11. Graph - Scatter plot

SENTIMENT ANALYSIS

For the purpose of this sentiment analysis, the overall average weekly paid hours in 2023 will be analysed.

Creating a histogram using the Seaborn library to visualise the distribution of the hours worked. The number has been set to 10 after several attempts to identify the number that allowed clear and clean visualisation of the distribution

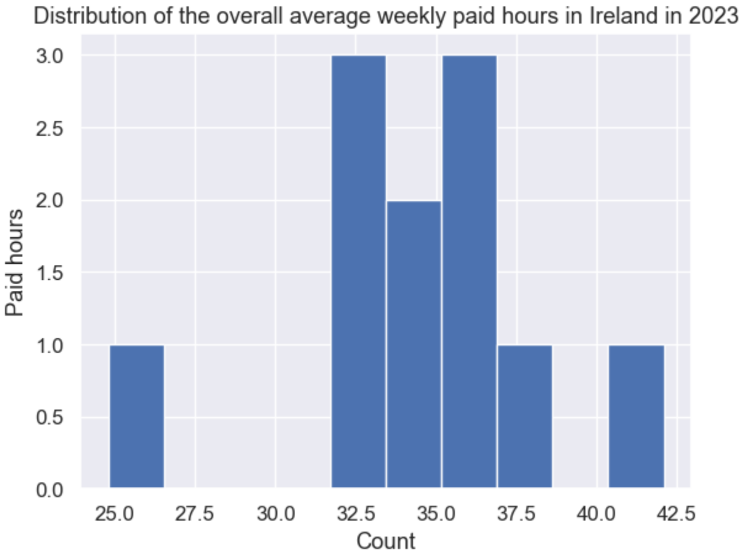
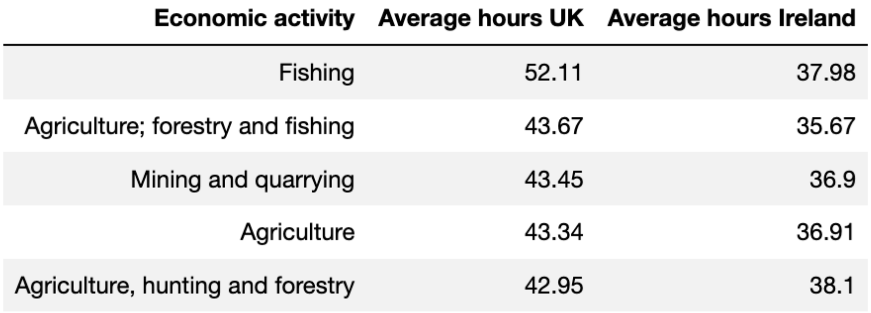


Figure 12. Graph - Distribution of the hours worked

EXPLORATORY DATA ANALYSIS

In the Exploratory Data Analysis phase, the focus will be exclusively on Ireland and the UK in order to compare the average weekly worked hours in both countries.



Economic activity	Average hours UK	Average hours Ireland
Fishing	52.11	37.98
Agriculture; forestry and fishing	43.67	35.67
Mining and quarrying	43.45	36.9
Agriculture	43.34	36.91
Agriculture, hunting and forestry	42.95	38.1

Figure 13. Table - Economic activity between Ireland and UK

The 'Fishing' economic activity is where more hours have been worked. It is visible that the string 'Fishing' appears in more than one row. Therefore, a search will be performed to identify all the rows containing the 'Fishing' string.

The two unique strings containing the 'Fishing' string are 'Agriculture; forestry and fishing' and 'Fishing'. Even though the results won't show details about the Fishing economic activity only, a decision has been made to include both economic activities in the analysis to evaluate the hours worked in the fishing industry indeed.

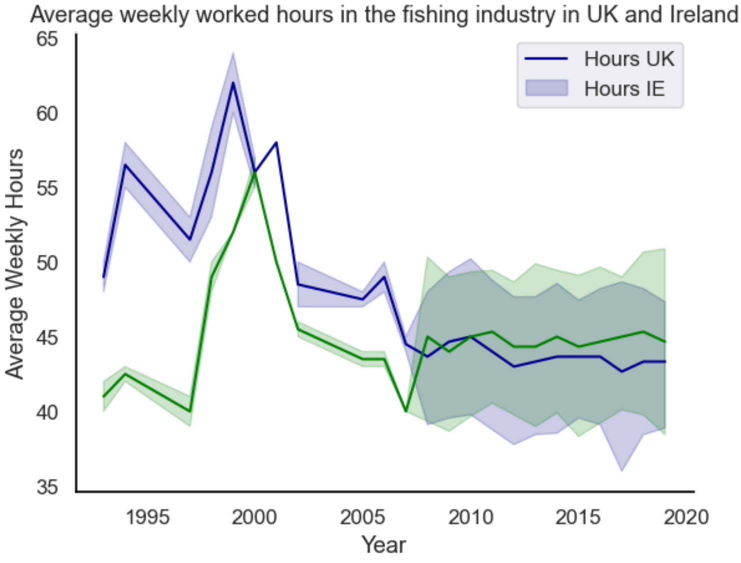


Figure 14. Graph - Number of hours worked in the fishing industry in the UK and Ireland.

Conclusions

In this conclusion, I wanted to explain more in detail the tasks that were required for me to do in this **CA 3 Repeat Programming for Data Analytics.**

1- Programming: The project must be explored programmatically: this means that you must implement suitable Python tools (code and/or libraries) to complete the analysis required. All of this is to be implemented in a Jupyter Notebook. **[0-20]**

In the past **CA2**, I received the following mark and feedback: **"Great use of programming tools. Some code repetition. 13/20".**

This helps me to improve my coding and try to not repeat some code in this **CA3 Repeat.**

I used Python as a programming language due to a variety of advantages, including its straightforward coding, which enables solutions to be developed with fewer lines of code than with other programming languages.

Jupyter Notebook was a great place to put the code, create visualisations, and write explanations/comments all in one document. Also, I have used libraries like **Numpy and Pandas** in order to perform data cleansing. These are the libraries that are conventionally used as a common practice in order to perform mathematical and statistical operations during a data analysis process.

Matplotlib and Seaborn libraries were also used in order to perform data visualisation procedures

Working with Python tools, code, and libraries in a Jupyter Notebook makes it easier to explore, clean, and analyse data. I believe when Python and Jupyter Notebook are used together, they create a powerful and efficient environment for analysing data and coming up with useful ideas and results.

Furthermore, Jupyter Notebook has proven to be an efficient tool for implementing Python and SQL in a single environment. This allows for a more linear and efficient way to implement calculations and visualize the analysis results.

2- Data structures: You are required to gather and process data that has been stored in at least two distinct formats. For example, this can be data in a CSV file, from a MySQL database or from a web API in JSON format. **[0-20]**

In the past **CA2**, I received the following mark: **"Task 2: 14/20"**

The gathering and process data in this **CA3 Repeat** was stored in a CSV file and JSON format.

Note: This JSON format has a considerable size of 71.9 MB. I had to compress it into a zip file to upload to GitHub and Submitted to Moodle.

I had trouble **(the same happened in my CA2)** with the implementation of the MySQL database in my MacBook Pro. Therefore I have decided to implement **SQLite** as a server-less database which means the DB engine runs as a part of the app and not requires a server to run like MySQL.

The requirement to gather and process data from at least two distinct formats, helped me to do more research and learn the importance of ensuring flexibility, adaptability, and efficiency in the process. This diversity showed me to handle real-world data scenarios and extract valuable information for decision-making and problem-solving.

Another challenge I faced was finding a dataset in JSON format, as most datasets come in CSV format. However, having allowed me to work again on a JSON file allowed me to understand the requirements and needs of such a file format.

3- Documentation: The project documentation must include sound justifications and explanations of your code choices. Code quality standards should also be applied. **[0-20]**

In the past **CA2**, I received the following mark and feedback: **"Task 3: Good comments and markdown cells. 13/20".**

This helps me to improve and keep the same structure in this **CA3 Repeat**.

This time I commented more in detail on each step during my coding. Helping me to be more clear in the explanation and for the creation of this report.

Through my comments in the code, I tried my best to apply code quality standards. In this way, anyone can read and understand my thinking behind and easily identify potential issues, suggest improvements, and provide feedback.

I think this practice improves the codebase and makes the standard of the code as a whole better.

Although the process took longer than expected, it allowed me to present my code as efficiently as possible by being concise and clear in writing my procedures.

4- Testing & Optimisation: You are required to document and evaluate a testing and optimisation strategy for your analysis. As part of this, you may want to plan and document how you ensured your code is doing what it is meant to, as well as ensuring that the code is making good use of your resources (eg computing, time etc). Note any trade-offs that you've made in these areas. **[0-20]**

In the past **CA2**, I received the following mark and feedback: **"Task 4: Not discussed 0/20"**

Unfortunately, I have failed in doing this task. Therefore on this **CA3 Repeat,** I have implemented the following:

I applied a unittest to test the effectiveness of a function I implemented and verify that it worked properly. I created four test data frames to check that the function was working as it should have worked, and the result confirmed its functionality on all four data frames. In the Machine Learning section of the CA, I used the time it tests to compare the speed of two functions and choose the fastest one of the two. Later, I tested the effectiveness of two regression algorithms: Random Forest Regression and Support Vector Regression. After training the two models on the train and test data and generating predictions on the test data, I calculated the Mean Squared Error, R-squared, and Mean Absolute Error for each model to choose the most adequate algorithm based on the results. In this case, I chose the Random Forest Regression algorithm.

5- Data manipulation: For each of the different data sources, compare and contrast at least two relevant libraries and techniques for a) processing and b) aggregating the respective data, in order to justify your chosen libraries/techniques. **[0-20]**

In the past **CA2**, I received the following mark and feedback: **"Task 5: Not discussed 0/20"**

Unfortunately, I have failed in doing this task. Therefore on this **CA3 Repeat,** I have implemented the following:

I applied Seaborn for static charts, mainly histograms and line plots. Seaborn was chosen due to its aesthetically appealing visual, achievable with less coding than Matplotlib. I implemented Plotly Express to create animated plots due to the practicality of this library in achieving these results.

I then applied Matplotlib to plot simple charts when applying Machine Learning models. In this last case, I chose Matplotlib because of its immediateness and because a focus on aesthetics was not paramount when plotting these charts. Overall, I have concluded that all libraries are powerful tools to visualise plots with Python. Each can be used for its speciality: Matplotlib for simple and quick plots, Seaborn for simple and appealing plots, and Plotly Express for appealing and animated plots.

References and Data Sources

1- Datasets that were used for this assessment:

- **EHQ10 - Public Sector Employment and Earnings:** The Public Sector Employment and Earnings Status Q1 2008 to Q1 2023. **Accessed July 25, 2023**

Available: https://data.gov.ie/dataset/ehq10-public-sector-employment-and-earnings/resource/062d0798-b666-407f-b72d-077f989bc057?inner_span=True

- **HOW_TEMP_SEX_ECO_NB_A-full-2023-07-24.json:** Mean weekly hours actually worked per employed person by sex and economic activity (Annual). Accessed July 30, 2023

Available: https://www.ilo.org/shinyapps/bulkexplorer57/?lang=en&id=HOW_TEMP_SEX_ECO_NB_A

2- Towards Data Science Inc. Website magazine, where thousands of independent contributors contribute their knowledge and skills. **Accessed July 25, 2023**

Available: https://towardsdatascience.com/a-practical-introduction-to-the-shapiro-wilk-test-for-normality-5675e52cee8f

3- Version control repository in Github.

Available: https://github.com/2014222-student-cct-ie/2014222-GeomarMunoz-CA-03-REPEAT/