**MSc in Data Analytics (SB+) - Sept 2023 - 2023 - YR1**

Author: Jose Maria Rico Leal

e-mail: sba23021@student.cct.ie

Student ID: sba23021

GitHub Link: https://github.com/JoseRicoCct/CA2\_Integrated\_Assesment\_MSc\_Data\_Analytics\_CCT\_Semester\_1.git

Irish transport sector

## **Abstract**

*Empty*

[**Abstract** 1](#_Toc154779035)

[**1.** **Introduction** 3](#_Toc154779036)

[**2.** **Programming** 4](#_Toc154779037)

[2.1. Programming 4](#_Toc154779038)

[2.2. Data Structures 4](#_Toc154779039)

[2.3. Documentation 6](#_Toc154779040)

[2.4. Testing and Optimisation 6](#_Toc154779041)

[2.5. Data Manipulation 6](#_Toc154779042)

[*2.5.1.* *Data Processing* 6](#_Toc154779043)

[*2.5.2.* *Data Aggregation* 7](#_Toc154779044)

[**3.** **Statistical Analysis** 7](#_Toc154779045)

[3.1. Descriptive Statistics 7](#_Toc154779046)

[*3.1.1.* *Dataset for Confidence Interval* 7](#_Toc154779047)

[*3.1.2.* *Dataset for Hypothesis Test Two Populations* 8](#_Toc154779048)

[*3.1.3.* *Dataset for ANOVA One-Way* 10](#_Toc154779049)

[*3.1.4.* *Dataset for Chi-Squared Test* 11](#_Toc154779050)

[*3.1.5.* *Dataset for Kruskal-Wallis* 12](#_Toc154779051)

[*3.1.6.* *Dataset for U-Mann Whitney* 14](#_Toc154779052)

[3.2. Confidence Interval 15](#_Toc154779053)

[3.3. Inferential Statistics 17](#_Toc154779054)

[*3.3.1.* *Parametric* 17](#_Toc154779055)

[*3.3.1.1.* *T-test Two Populations* 17](#_Toc154779056)

[*3.3.1.2.* *ANOVA One-Way* 19](#_Toc154779057)

[*3.3.1.3.* *Chi-squared test* 21](#_Toc154779058)

[*3.3.2.* *Non-parametric* 22](#_Toc154779059)

[*3.3.2.1.* *Kruskal-Wallis* 22](#_Toc154779060)

[*3.3.2.2.* *U-Mann Whitman* 24](#_Toc154779061)

[3.4. Further Research and Challenges Faced 26](#_Toc154779062)

[**4.** **ML** 27](#_Toc154779063)

[4.1. Supervised Learning, GridSearchCV Hyperparameter Tunning. 28](#_Toc154779064)

[4.2. Cross Validation and Feature Reduction. 29](#_Toc154779065)

[4.3. Unsupervised Learning. 30](#_Toc154779066)

[4.4. Sentiment Analysis. 30](#_Toc154779067)

[4.5. Table and Conclusions 32](#_Toc154779068)

[*4.5.1.* *Supervised Learning* 32](#_Toc154779069)

[*4.5.2.* *Cross Validation and Feature Reduction* 35](#_Toc154779070)

[*4.5.3.* *Unsupervised Learning* 37](#_Toc154779071)

[**5.** **Data Preparation and Visualisation** 42](#_Toc154779072)

[5.1. Data acquistion 42](#_Toc154779073)

[5.2. EDA methodology 42](#_Toc154779074)

[5.3. Visualisations 42](#_Toc154779075)

[5.4. Dashboard 42](#_Toc154779076)

[**6.** **Conclusion** 42](#_Toc154779077)

[**7.** **References** 42](#_Toc154779078)

[**8.** **Annex** 43](#_Toc154779079)

[8.1. MySQL Setup 43](#_Toc154779080)

# **Introduction**

# **Programming**

# Programming

I organized the project into five Jupyter Notebooks:  
*CA2\_ML\_Code.ipynb*, *CA2\_Programming\_Code.ipynb*, *CA2\_Statistics\_Code.ipynb* and *CA2\_Data\_Visualisation.ipynb*. The project is programmatically explored using Python, emphasizing modularity and narrative-style programming.

# Data Structures

For this project I gathered and process data from:

CSV file:

A screenshot of a computer

Description automatically generated

Web API in CSV format:

A screenshot of a computer screen

Description automatically generated

Eurostat module:

A screenshot of a computer

Description automatically generated

Also, I explored gathering data from a:

Web API in JSON format:



Entire parsing process is documented in *“CA2\_Programming\_Code\_sba23021.ipynb”* from cell 8 to 17.

MySQL database:

A screenshot of a computer

Description automatically generated

As I could not find an open MySQL database to connect, I will be demonstrating how to extract data from a local MySQL database. I created a table named *“tran\_hv\_psmod”* within a schema called *“eurostat”* and imported the previously fetched file from Eurostat, *“TRAN\_HV\_PSMOD.csv”*:

A screenshot of a computer

Description automatically generated

Once I had the records in the table, I was able to retrieve them using the Python package *“sqlalchemy”*:

A screenshot of a computer

Description automatically generated

Complete instructions on creating the table, pushing, and fetching records are provided in the annex.

# Documentation

In terms of documentation, I endeavoured to provide comments explaining the rationale behind each line of code. This includes details about the purpose of each line and the workflow for data manipulation and visualization.

I maintained code quality standards by adhering to the *“PEP 8”* styling guidelines. Given the project's nature, which primarily involves data manipulation and visualization, there was no need for the introduction of complex programming constructs or advanced *“OOP”* principles. My objective was to ensure clean and modularized code.

# Testing and Optimisation

# Data Manipulation

In this section, we will explore the libraries and techniques employed for the three data sources used in this project: *CSV* files, web *API CSV*, and the *Eurostat* module.

# *Data Processing*

*Pandas* library, function *pd.read\_csv*:

A screenshot of a computer

Description automatically generated

Figure X used for Web API CSV

A screenshot of a computer

Description automatically generated

*Eurostat* library, function *eurostat.get\_data\_df*:

A screenshot of a computer screen

Description automatically generated

*Scipy stats* library, method *st.t.interval* and *Numpy* and *np.mean()*:

A screenshot of a computer

Description automatically generated

# *Data Aggregation*

# **Statistical Analysis**

# Descriptive Statistics

# *Dataset for Confidence Interval*

Dataset used is *“tii03-passenger-journeys-by-luas”.* We aim to estimate the range of potential values for the parameter *“LUAS average passenger number”.*Below are the descriptive statistics for the total LUAS passenger numbers (green and red lines) for the years 2019, 2020, 2021 and 2022:

A table with numbers and a number on it

Description automatically generated

We will be examining the mean values to establish the confidence intervals.  
It is interesting to note that both lines are quite balanced in terms of usage.

A group of pie charts with numbers

Description automatically generated

In Section 3.2, a detailed analysis of confidence intervals will be conducted for this dataset.

# *Dataset for Hypothesis Test Two Populations*

In this section we are comparing Ireland with some European countries, formulating hypothesis to assess if there are statistically significant differences in the *“percentage average passenger-kilometres based on type of transport (Bus, Car and Train”.* Since we will be using Ireland mean against other countries mean, this plot will assist us in constructing the hypothesis:

A group of colorful bars

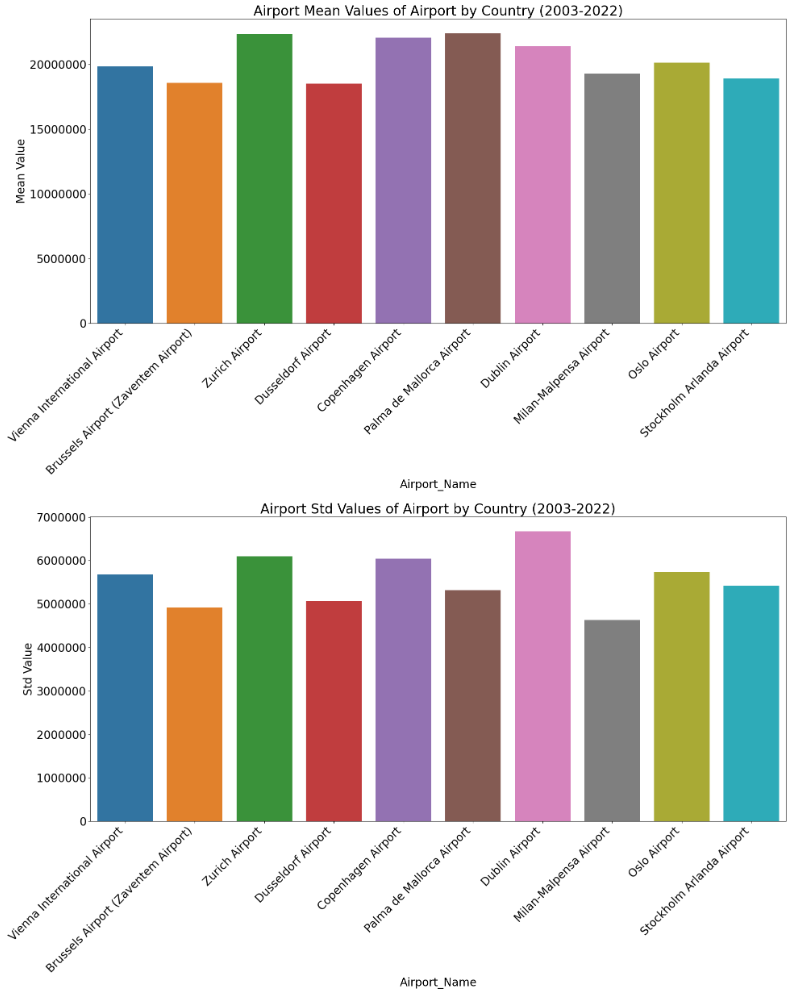
Description automatically generated with medium confidence

The dataset used is *“Modal split of inland passenger transport”* withthe Eurostat code *“TRAN\_HV\_PSMOD”.*

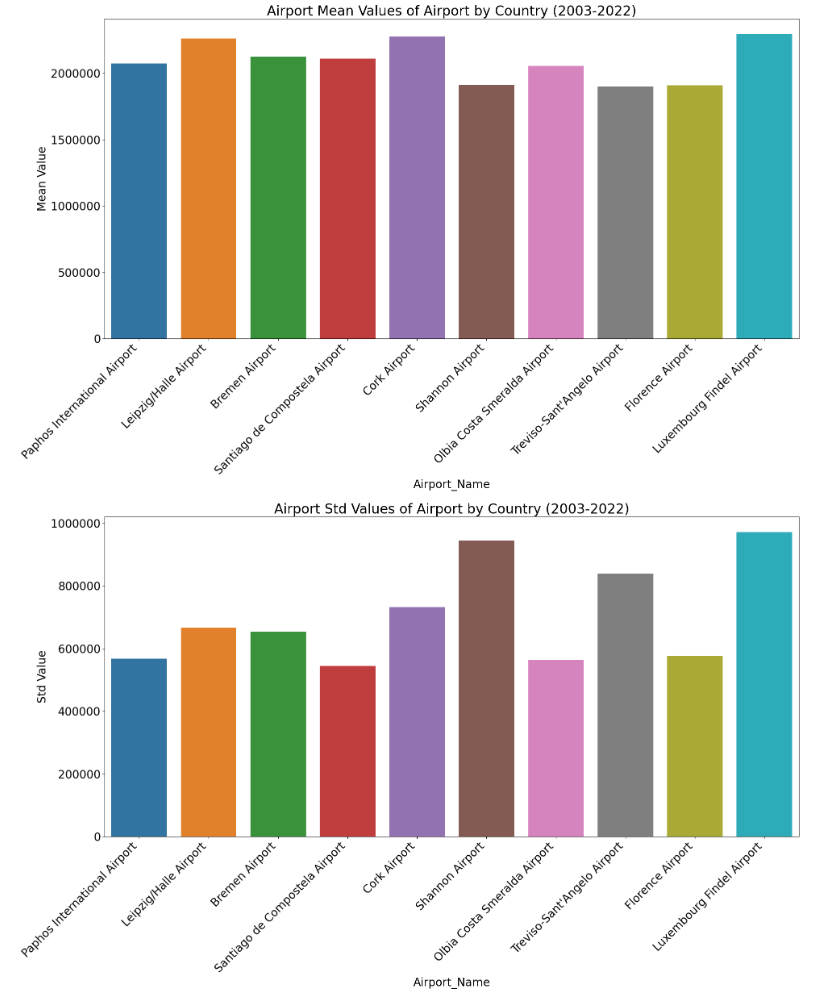
# *Dataset for ANOVA One-Way*

Using Irish airports as a reference, an ANOVA will be conducted to test whether there are any statistically significant differences in the means compared to other European airports. The dataset used is *“Air passenger transport by main airports in each reporting country”* with the Eurostat code *“AVIA\_PAOA”.*

In the first scenario Dublin Airport is considered, and below a graph of the subset of airports selected for this case:



In the second scenario, Shannon Airport is considered, and below is a graph showing the subset of airports selected for this case:



# *Dataset for Chi-Squared Test*

This test will be performed to examine the association between the categorical variable *“Motor\_energy\_type”* for Ireland and Austria. The dataset used is *“New passenger cars by type of motor energy”* with the Eurostat code *“road\_eqr\_carpda”.*

This model is highly sensitive to the frequency of the variables. I have presented two cases: one where we reject H0, and the second one where I manually changed values to accept H0. I will now illustrate how the categorical variable numbers will look for each case, with further analysis to follow in *“3.3.1.3. Chi-squared test”.*

Scenario 1: Rejecting H0.

A graph of a number of cars

Description automatically generated with medium confidence

Scenario 2: Accepting H0.

A graph of different types of cars

Description automatically generated

# *Dataset for Kruskal-Wallis*

For this test, I utilized the same dataset as for ANOVA. Some of the airports violated the assumptions of normality required for ANOVA. The advantage of using the Kruskal-Wallis test is that it does not require normality to perform the test. I will present two scenarios: one to accept H0 and another to reject H0.

Scenario 1: Accepting H0.

A screenshot of a graph

Description automatically generated

Scenario 2: Rejecting H0.

A screenshot of a graph

Description automatically generated

# *Dataset for U-Mann Whitney*

I used a new dataset for this test, *“Passengers transported (Railway transport)”,* with the Eurostat code *“rail\_pa\_total”* because the data did not follow a normal distribution. This choice allowed me to demonstrate the test's capability to handle non-normally distributed data.

Scenario 1: Accepting H0.

A number on a white background

Description automatically generated

Scenario 1: Rejecting H0.

A number on a white background

Description automatically generated

# Confidence Interval

The task is to determine the weekly LUAS average for the total number of passengers in the years 2019, 2020, 2021, and 2022. It is important to note that both LUAS lines (red and green) are in scope. The analysis will be conducted with a 90% confidence level, and here are the results:

A table with numbers and a few black text

Description automatically generated

E.g. At a 90% confidence level, for 2019 the weekly LUAS number of passengers average is between 453K and 472K. Subsequently the same formulation for the rest of the years.

Plotting the confidence intervals:

A group of graphs showing different sizes of data

Description automatically generated with medium confidence

After performing confidence intervals, the next natural step is to verify if the weekly averages are the same for both the red and green LUAS lines. We will use a t-test.

A table with numbers and letters

Description automatically generated

Hypothesis:

H0: μ green line = μ red line.  
H1: μ green line != μ red line.

Results:

A group of graphs with numbers and symbols

Description automatically generated with medium confidence

A table with numbers and text

Description automatically generated

At a 5% significance level, we accept the Null Hypothesis; there is not enough evidence to conclude that the weekly mean values for the LUAS green line are different from those of the red line.

# Inferential Statistics

# *Parametric*

# *T-test Two Populations*

To perform this test, we will compare Ireland with three different countries, each having a distinct transportation method. The first scenario involves comparing Ireland with Slovenia for cars, the second compares Ireland with Denmark for buses, and the third compares Ireland with Slovenia again, but this time for trains.

Hypothesis:

H0: μ Ireland = μ EU Country.

H1: μ Ireland != μ EU Country.

Results:

A graph of a normal distribution

Description automatically generated

Since the p-value is greater than alpha, we accept the null hypothesis (H0). There is not enough evidence to conclude that there is a significant difference between the percentage average of passenger-kilometres for the BUS vehicle in Ireland and Slovenia.

A graph of a normal distribution

Description automatically generated

As the p-value is less than alpha, we reject the null hypothesis (H0), providing sufficient evidence to conclude a significant difference in the percentage average of passenger-kilometres for the Car vehicle between Ireland and Denmark.

A graph of a normal distribution

Description automatically generated

As the p-value is less than alpha, we reject the null hypothesis (H0), indicating sufficient evidence to conclude a significant difference in the percentage average of passenger-kilometres for the Train vehicle between Ireland and Slovenia.

A screenshot of a calculator

Description automatically generated

# *ANOVA One-Way*

In the first scenario, we examine Dublin, Zurich, and Copenhagen airports to verify whether the yearly average passenger numbers for the period from 2003 to 2022 are the same or not.

To perform ANOVA, we need both Shapiro-Wilk and Levene tests to have p-values greater than 5% alpha.

A screenshot of a test results

Description automatically generated

Now we can perform ANOVA:

H0: μ IE\_EIDW = μ CH\_LSZH = μ DK\_EKCH.

H1: there are at least 2 μ that are different one to another.

Result p-value = 0.889

A graph of a flight

Description automatically generated

There is no reason to reject the Null Hypothesis; therefore, we can conclude that, with a 5% alpha, the mean of annual passengers carried (2003-2022) for Dublin, Zurich and Copenhagen airports is quite similar.

Second scenario: Shannon, Billund, and Treviso airports. We want to verify if the yearly average passenger numbers for the period 2003 to 2022 are the same or not.

Shapiro-Wilk and Levene tests:

A screenshot of a test

Description automatically generated

ANOVA hypothesis:

H0: μ IE\_EINN= μ IT\_LIPH= μ DK\_EKBI.

H1: there are at least 2 μ that are different one to another.

Result p-value = 0.04

A graph of a flight

Description automatically generated

We fail to accept the Null Hypothesis; therefore, we can state that with a 5% alpha the mean of annual passengers carried (2003-2022) for Shannon, Treviso-Sant'Angelo and Billund airports is different.

# *Chi-squared test*

First Scenario:

Hypothesis:

H0: There is no significant difference between the observed and expected frequencies. Ireland and Austria are independent with no association or relationship.

A screenshot of a number of numbers

Description automatically generated

H1: There is a significant difference between the observed and expected frequencies, indicating a non-independent relationship between Ireland and Austria

Result p-value = 1.64e-14, we fail to accept H0.

A graph of a car

Description automatically generated

In the second scenario, I manually adjusted values for Austria to align frequencies more closely with those of Ireland; this model is highly sensitive to substantial differences between categorical variables.

A screenshot of a number

Description automatically generated

After applying the Chi-Square test, we obtained a p-value of 0.59, leading us to accept H0.

A graph of a car

Description automatically generated

# *Non-parametric*

# *Kruskal-Wallis*

Scenario 1:

We do not have normality for our samples:

A screenshot of a computer

Description automatically generated

Hypothesis:

H0: μ IE\_EIKN= μ FR\_LFBP= μ SE\_ESGP.

H1: there are at least 2 μ that are different one to another.

Result, p-value = 0.23511.

A graph of a line

Description automatically generated

We accept H0; the means of annual passengers carried (2003-2022) for Ireland West Knock, Pau Pyrenees, and Goteborg airports are quite similar.

Scenario 2:

We do not have normality for our samples:

A screenshot of a computer

Description automatically generated

Hypothesis:

H0: μ IE\_EIKY= μ DE\_EDSB= μ PL\_EPRZ.

H1: there are at least 2 μ that are different one to another.

Result, p-value = 8.48e-08

A graph of a passenger carrier

Description automatically generated

We fail to accept H0; the means of annual passengers carried (2003-2022) for Kerry, Karlsruhe/Baden, and Rzeszow-Jasionka airports are different.

# *U-Mann Whitman*

Scenario 1:

Checking normality:

**A white rectangular object with black text

Description automatically generated**

Hypothesis:

H0: μ Ireland = μ Croatia.

H1: μ Ireland != μ Croatia.

Result p-value = 0.3068

A graph of a train passenger

Description automatically generated

As the p-value is greater than alpha, we accept H0, indicating that there is no significant difference between the average number of train passengers in Ireland and Croatia.

Scenario 2:

Checking normality:

A white background with black text

Description automatically generated

Hypothesis:

H0: μ Ireland = μ Slovakia.

H1: μ Ireland != μ Slovakia.

Result p-value = 3.45e-06

A graph of a train passenger

Description automatically generated

We reject H0 as p-value is lower than alpha, there is a significant difference between the average number of train passengers between Ireland and Slovakia.

# Further Research and Challenges Faced

The tests I have conducted above reveal interesting findings:

*“Number of public transport journeys at highest level since the beginning of the pandemic”* (BreakingNews.ie, 2022), based on the confidence intervals, we can confirm that the number of passengers is recovering:

A graph of a number of people

Description automatically generated with medium confidence

*“Paris Charles De Gaulle recorded the highest number of air passengers”* (ec.europa.eu, n.d.). *“Dublin Airport Was EU’s 11th Largest Airport in 2018”* (DublinAirport, n.d.):

A graph of different colored bars

Description automatically generated

Analising means in the ANOVA section we can see that those headings are highly correlated with *“FR\_LFPG”* and *“IE\_EIDW”.*

The challenges faced included gathering the data and establishing the scenarios to perform the tests.

# **ML**

The methodology for ML part can be seen as follows:

A diagram of a data processing process

Description automatically generated

# Supervised Learning, GridSearchCV Hyperparameter Tunning.

The choice of the dataset *“TRAN\_HV\_PSMOD”* and the selection of supervised ML models (Decision Tree, Random Forest, K-Nearest Neighbours, and Support Vector Machine) are purely matters of modelling. After numerous attempts, this combination has proven effective.

We will be modelling the dataset to see how models react having these classes:

A screenshot of a computer

Description automatically generated

I paired Ireland with Hungary because it showed the best performance association for selected ML models, here are the results:

A screenshot of a computer

Description automatically generated

Overall model is performing well however for Gamma = 0.0001, models seem to be overfitted as this score is lower compared to Gamma 0.01 and 0.001:

A graph of a graph with numbers

Description automatically generated with medium confidence

To address the issue of overfitting, I enriched the dataset by quarterly weighting yearly values:

A close-up of a number

Description automatically generated

A screenshot of a graph

Description automatically generated

The results are as follows:

A screenshot of a computer

Description automatically generated

A graph of a number of numbers

Description automatically generated with medium confidence

Now we have a better fit, with higher scores observed at Gamma = 0.0001.

# Cross Validation and Feature Reduction.

Ireland will be our target variable for each transportation method. In the first attempt with yearly data, we obtained the following results:

A screenshot of a number of features

Description automatically generated

After quarterly enrichment, increasing the number of rows from 32 to 128, we obtained the following results:

A screenshot of a computer

Description automatically generated

Linear regression estimation has performed much better with enriched data.

# Unsupervised Learning.

Dataset it is split into each mode of transportation having Ireland as a reference.

First attempt with quarterly data we get the following results:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

To improve the results, we are going to enhance the dataset by breaking down each year into monthly values. The results are as follows:

A screenshot of a computer

Description automatically generated

A screen shot of a number

Description automatically generated

*KMeans* and *PCA* have performed better with more data.

# Sentiment Analysis.

We are going to conduct sentiment analysis using Ryanair reviews and a set of tweets related to USA airlines. Results:

A comparison of a number of squares

Description automatically generated with medium confidence

*Tweets* dataset is larger than *Ryanair* one that explains more correctly classified inputs:

A screenshot of a graph

Description automatically generated

Ryanair figure X

A screenshot of a computer

Description automatically generated

USA Airlines

Model accuracy for *Ryanair* is 73% and 75% for *USA Airlines.* ROC results as it follows:

A graph of a curve

Description automatically generated

*USA Airlines* seem to be classifying bad inputs better, while the other two are closer to each other. Let us test the classifier by adding reviews:

A screenshot of a computer program

Description automatically generated

Result:

A graph of different sizes and colors

Description automatically generated with medium confidence

*Ryanair's* sentiment analysis performs well even though it has fewer inputs than *USA Airlines.* This difference could be attributed to the collection of reviews from *Tripadvisor,* where I ensured capturing opinions across good, bad, and neutral categories.

# Table and Conclusions

# *Supervised Learning*

Let us compare each model after enriching the dataset.

Decision Trees CM:

A screenshot of a diagram

Description automatically generated

Random Forrest:

A screenshot of a graph

Description automatically generated

KNN:

A graph of different colored lines

Description automatically generated with medium confidence

GridSearchCV CM:

A screenshot of a graph

Description automatically generated

Gamma Accuracy:

A graph of different values

Description automatically generated

Accuracy Table:

A table with numbers and text

Description automatically generated

By adding more values, models have performed better, with the exception of RF. However, we solved overfitting in the hyperparameter tuning phase, achieving higher accuracy at a lower level of gamma.

# *Cross Validation and Feature Reduction*

Same approach as we followed previously, we increased dataset row count by breaking down years into quarters, here the results:

A group of graphs showing the value of a number of objects

Description automatically generated with medium confidence

Summary table:

A screen shot of a computer

Description automatically generated

Optimal feature selection and improved accuracy after enriching the dataset, undoubtedly, this method requires a larger dataset to perform well.

# *Unsupervised Learning*

PCA Variance:

A collage of blue and red graph

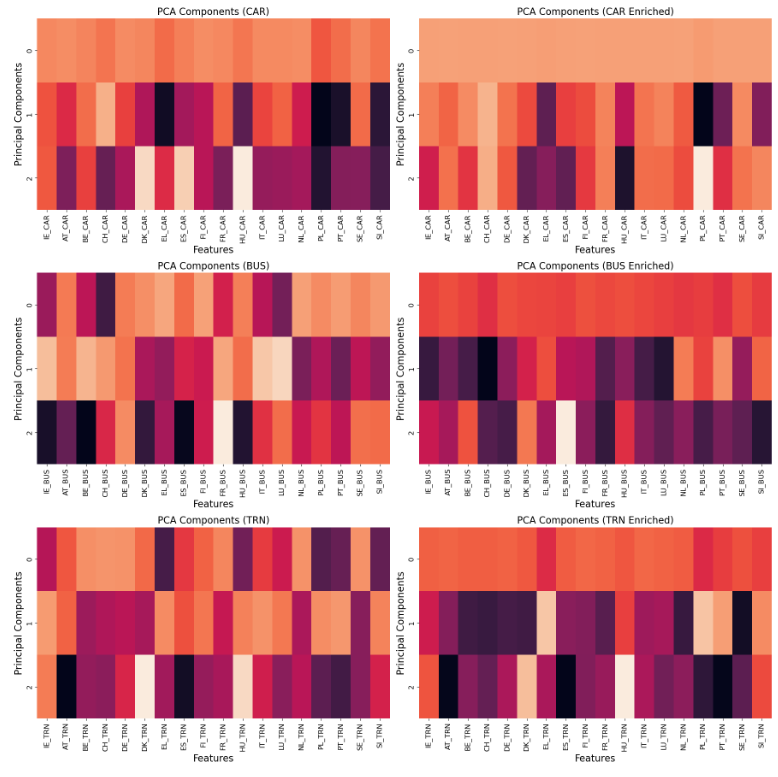
Description automatically generated

PCA Scatter Plot:

A group of colored dots

Description automatically generated

PCA Heatmap:



Elbow Method:

A graph of a number of different numbers

Description automatically generated with medium confidence

Silhouette Score:

A group of graphs showing different sizes of clusters

Description automatically generated

Tables PCA and Silhouette Score:

A screenshot of a computer screen

Description automatically generated

A screenshot of a graph

Description automatically generated

After enriching the data, the first principal component explains almost 97% of the variability for the CAR and BUS datasets, while the TRN dataset exhibits 76.86% variability on the first component. This demonstrates a clear improvement.  
The silhouette score has improved, indicating that the clusters are now closer to being well-defined (closer to +1). Additionally, all plots demonstrate a clear improvement after enlarging the dataset.

# **Data Preparation and Visualisation**

# Data acquistion

# EDA methodology

# Visualisations

# Dashboard

# **Conclusion**

# **References**

Statistics:

BreakingNews.ie. (2022). *Number of public transport journeys at highest level since the beginning of the pandemic.* [online] Available at: https://www.breakingnews.ie/ireland/number-of-public-transport-journeys-at-highest-level-since-the-beginning-of-the-pandemic-1304260.html [Accessed 11 Dec. 2023].

ec.europa.eu. (n.d.). *Air transport statistics.* [online] Available at: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Air\_transport\_statistics#:~:text=In%202022%2C%20820%20million%20people%20in%20the%20EU%20travelled%20by%20air.&text=In%202022%2C%20Paris%20Charles%20De [Accessed 11 Dec. 2023].

DublinAirport. (n.d.). *Dublin Airport Was EU’s 11th Largest Airport in 2018*. [online] Available at: https://www.dublinairport.com/latest-news/2019/05/31/dublin-airport-was-eu-s-11th-largest-airport-in-2018.

DublinAirport. (n.d.). *Dublin Airport Was EU’s 11th Largest Airport in 2018*. [online] Available at: <https://www.dublinairport.com/latest-news/2019/05/31/dublin-airport-was-eu-s-11th-largest-airport-in-2018>.

# **Annex**

# MySQL Setup

I used *MySQL Workbench* to simulate the process of gathering data from a MySQL database.

Steps:

1. For schema creation, click on the database icon, name it *eurostat* in this case, and then click on *Apply*:

A screenshot of a computer

Description automatically generated

1. Click *Apply*:

A screenshot of a computer

Description automatically generated

1. Click *finish*:

A screenshot of a computer

Description automatically generated

1. Our schema will appear:

A screenshot of a computer

Description automatically generated

1. To create the table within the schema for inserting *“TRAN\_HV\_PSMOD.csv”,* load the *“TRAN\_HV\_PSMOD\_table\_creation.sql”* file. Click on File and select *Open SQL Script:*

A screenshot of a computer program

Description automatically generated

1. Execute the code from *“TRAN\_HV\_PSMOD\_table\_creation.sql”:*

A screenshot of a computer

Description automatically generated

1. After successful execution, the table will appear under the *“eurostat”* schema. Please refresh to view the table:

A screenshot of a computer

Description automatically generated

1. Hover over the table, right-click, and select *Table Data Import Wizard*:

A screenshot of a computer

Description automatically generated

1. Browse and locate *“TRAN\_HV\_PSMOD.csv”*, the click on *open* and *next*:

A screenshot of a computer

Description automatically generated

1. *Next:*

A screenshot of a computer

Description automatically generated

1. *Next:*

A screenshot of a computer

Description automatically generated

1. *Next:*

A screenshot of a computer

Description automatically generated

1. *Next:*

A screenshot of a computer

Description automatically generated

1. *Finish:*

A screenshot of a computer

Description automatically generated

1. Checking the results, we should have same number of columns and row count:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Connecting to MySQL and fetching the table *tran\_hv\_psmod*:

A screenshot of a computer

Description automatically generated

This is how we can connect to a MySQL database and pull data from there.