# **Analysis of Domestic Tourism in Ireland by Irish Residents**

**Group ID**: MSc in Data Analytics (Sept24 start)  
**Author**: Arim Faridi  
**Email**: 2024371@student.cct.ie  
**Student ID**: 2024371

Contents

[**Analysis of Domestic Tourism in Ireland by Irish Residents** 1](#_Toc181998432)

[**Acknowledgments** 2](#_Toc181998433)

[**Abstract** 2](#_Toc181998434)

[**Introduction** 2](#_Toc181998435)

[**Data Preparation and Visualization** 3](#_Toc181998436)

[**Statistical Analysis** 7](#_Toc181998437)

[**Conclusion and Insights** 16](#_Toc181998438)

[**References** 17](#_Toc181998439)

### **Acknowledgments**

I want to express my deepest gratitude to CCT College, which gave me the golden opportunity to pursue this master's program, a journey that marked a significant development in my academic and personal life. I am very much grateful to my professors, namely David McQuaid, Kayoum Khbuli, Muhammad Iqbal, and Ahmed Taufique, who have been instrumental in this project with their knowledge, encouragement, and mentorship. Their dedication and enlightening guidance have not only helped to deepen my understanding of the subject but have also inspired me in my journey for excellence in research. I am very much obliged to the <https://data.gov.ie/> for the necessary data and necessary resources that have helped me with my project. Reliable data availability enhances the depth and accuracy of my work a lot. This research would not have been possible without the generous support of all these individuals and institutions, for which I am truly grateful.

### **Abstract**

The study analyzes the trends in domestic tourism by Irish residents and how tourism fluctuates on a yearly and quarterly basis. This report summarizes the undertaking of analysis of tourism data using EDA, descriptive and inferential statistics, machine learning, and Python programming for the years 2018 to 2024 to achieve an understanding of the pattern and forecast the trend up to 2030. Based on the results, it is observed that domestic tourism was increasing until COVID-19 and then faced an extreme drop from which it started recovering. Also, predictive modeling gave forecasts of potential increases in domestic tourism, hence providing insights related to strategic planning in Ireland's tourism sector.

### **Introduction**

Besides providing jobs, domestic tourism has contributed to sustainable development in different local communities of Ireland and it has been very instrumental in the growth of the Irish economy. This report dives deep into the trends of domestic tourism in Ireland from the year 2018 to 2024, focusing on the most popular regions for the Irish people, the impact that the tourism industry as a whole received from COVID-19, and what is in the future. This analysis follows the CRISP-DM project management framework for data mining, concerning data understanding, preparation, modeling, and evaluation phases, deploying the results in this report.

Questions that I aim to answer by the end of the report are:

* How has tourism at home changed over time?
* Which are the most visited regions?
* How might domestic tourism trends develop up to 2030?

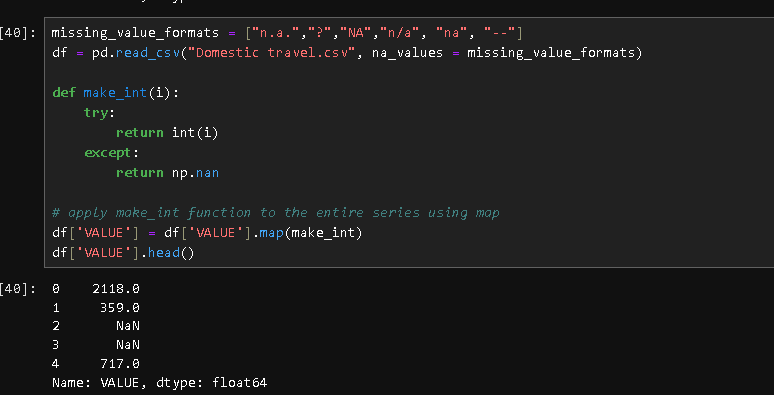
These questions provide the backbone of a structured approach to a thorough investigation of the historical data patterns and informed predictions about the future. This report aims to shed light on Ireland's domestic tourism landscape by recommending, based on available data, strategies that should support policymakers, tourism boards, and local businesses in the implementation of the best strategies for the continued maintenance of this sector for the next few years.

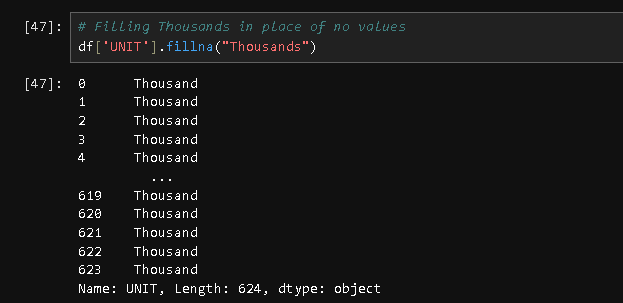
### **Data Preparation and Visualization**

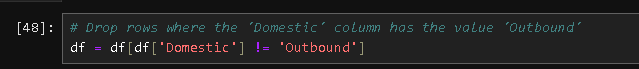
Domestic Tourism in Ireland required a proper preparation and visualization of trends by first acquiring data from the Central Statistics Office, CSO. Precisely, the information was sought from the HTA11 table record, which provides comprehensive information on domestic tourism in Ireland among Irish residents. This information is retrieved in CSV format to ensure that Python libraries support its manipulation and exploration efficiently.

1. **Data Acquisition:**  
   Data acquisition involved downloading the dataset as a CSV file, which makes it easy to work on the dataset with Python directly as it is highly compatible, and with major libraries like Pandas and NumPy, flexibly handling such kind of data reduces the pre-processing time.
2. **Data Loading:**  
   The "DomesticTravel.csv" CSV file was loaded into the Jupyter Notebook environment using Pandas via the pd.read\_csv() function. Data loaded into a Data Frame could then be structured and analyzed. Preliminary steps included the verification of the file format and encoding to avoid any prospective problems while loading data, such as encoding errors or unsupported characters.
3. **Data Inspection:**  
   Preliminary inspection methods were used to familiarize myself with the structure of the dataset. Using functions such as .head(), .shape, and .info(), the structure, dimensions, and number of non-null counts for each column were uncovered. The function .duplicated() was used to locate duplicate rows, while the .describe() function provided some key summary statistics, particularly those dealing with numeric which were important in finding outliers and anomalies. The inspection identified missing values or anomalies and thus prepared the ground for further cleaning.
4. **Data Cleaning:**  
   Data cleaning focused on handling missing values, duplicate entries, and irrelevant columns:

* **Handling Missing Values**: There were a few missing values that were either imputed in other words, filled in with reasonable estimates or removed if they were not necessary and unbiased for overall analysis.



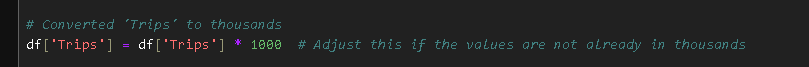




* **Removed Duplicates**: Duplicate rows are removed so that different biased results for the analysis can be avoided.
* **Columns Standardization**: Almost all the irrelevant columns were dropped to trim the dataset and make it suitable for further analysis.



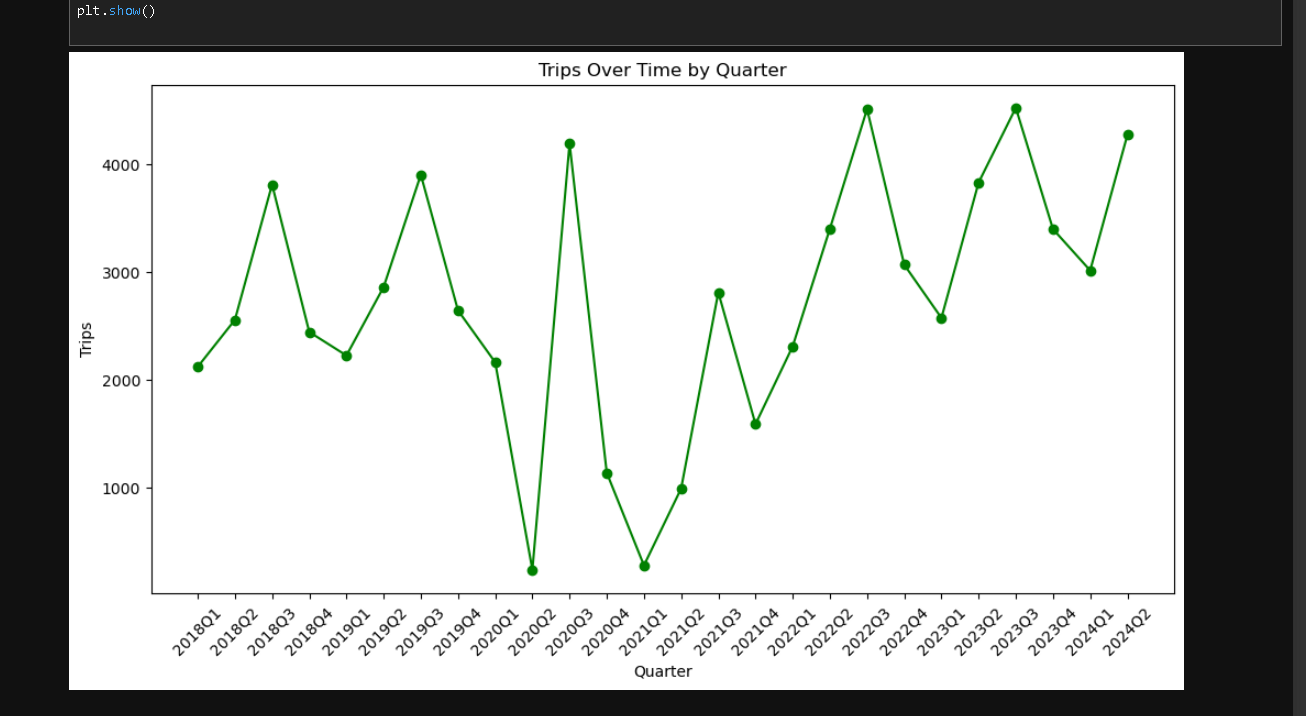
* **Unit Conversion**: Additionally, values in "Number\_Of\_Trips" were divided by one thousand for consistency, which makes comparing values easier and visualizations more interpretable.

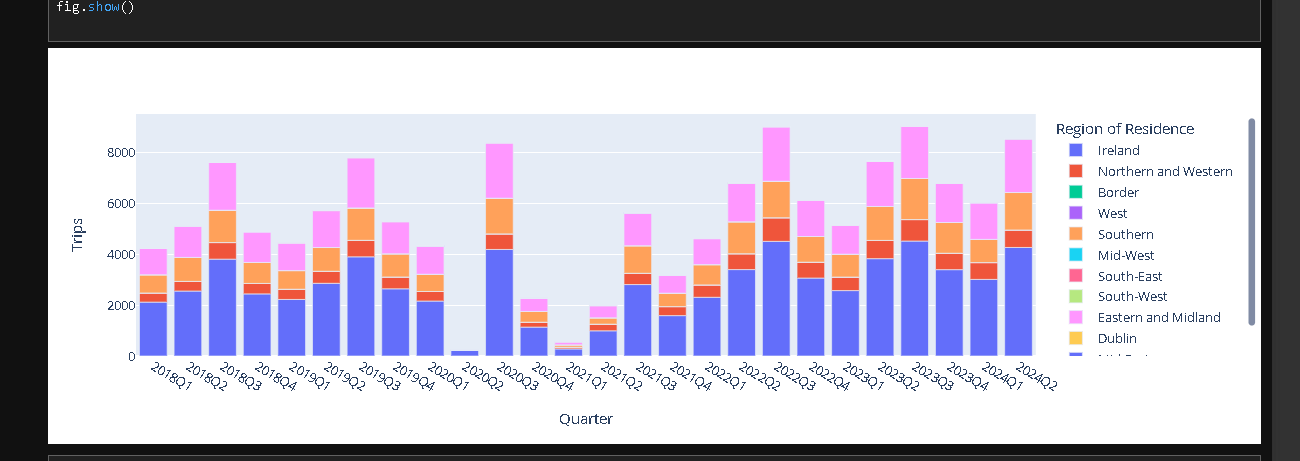


1. **Data Transformation:**  
   The data transformation relied on the creation of two major subsets in consideration of focused analysis:

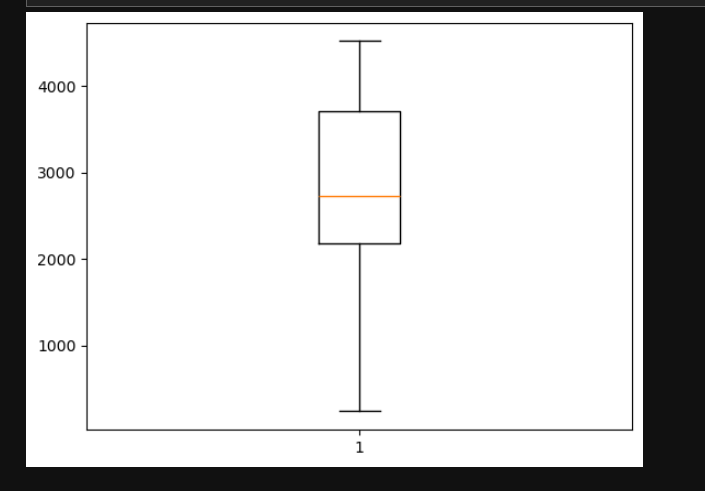
* **Ireland\_df**: Aggregations that show the total number of trips throughout Ireland.
* **ML\_df**: Subset of the main dataset for Machine Learning.

1. **Data Visualization**  
   Various data visualizations to represent and explain insights into the dataset were carried out:

* **Line Graphs**: These are the types of graphs used to follow trends over time. In this case, the graphs reflected the drastic drop in domestic tourism during the period of COVID-19.
* **Histograms**: The visualized distribution of data showing different regions as well. This helped me with knowing how much tourist does every region attract.



* **Box Plots**: Outliers were highlighted, and the shape of data distribution became obvious in order to help find regions with very high or low counts of trips.



Together, these visualizations provided an overview of domestic tourism trends in Ireland, including the effect of COVID-19, year-to-year differences, and overall distribution characteristics. This thorough approach to data preparation and visualization established a foundation for subsequent statistical and predictive analyses.

### **Statistical Analysis**

I collect the data, analyze and interpret it, and then present it to show patterns and trends in the data. Using descriptive and inferential techniques, I transform raw data into actionable insights that help me test hypotheses, make predictions, and support data-driven decisions in research, business, or scientific exploration.

#### **1. Descriptive Statistics**

Descriptive statistics are that aspect of statistics which summarize and organize data in such a way that a clear picture of their essential characteristics appears. To me, it simply means developing raw data into meaningful information through the deduction of measures of mean, median, mode, range, and standard deviation. All these metrics allow for insight into patterns and central tendencies, plus variability, making complex data more understandable. Descriptive statistics describe trends and distributions through the use of tables, charts, and graphs. It is preliminary or initial in analysis, providing the ground for further, more complex inferential statistical treatments by giving an overview of the data in terms of basic features.

##### **1.1 Measures of Central Tendency**

* **Mean** and **Median** provided insights into typical values and helped identify skewness. It also showed us what was the average number of trips that people were making across the years except during COVID-19.

##### **1.2 Measures of Dispersion**

* **Range:** It tells you how spread out the data is, which lets you know if there are any extreme values (outliers) that could affect your models.
* **Variance**: A feature's variability can be identified by its variance.
* **Standard Deviation:** The standard deviation is the square root of the variance and measures the average distance from the mean of each data point.

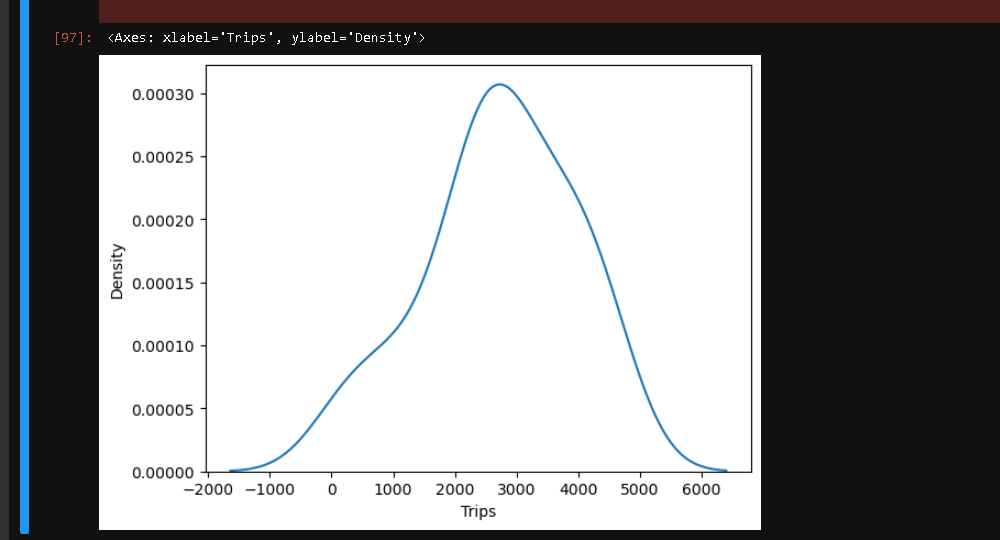
##### **1.3 Measures of Shape**

* **Skewness**: Skewness measures the asymmetry of the distribution of data and can also help in indicating what type of distribution does the data follow. My data was slightly skewed to the right but it still formed a bell curve
* **Kurtosis:** It indicates whether data points tend to have extreme values or are clustered around the mean. Ireland\_df has a slightly negative kurtosis, meaning it is platykurtic. The negative kurtosis was because of COVID-19 as tourism saw a massive drop during that period.

#### **2. Normal Distribution**

A normal distribution is also a Gaussian distribution or bell curve and is a probability distribution where most values are around the average, with less frequency as values go further from the center. In this, values bottleneck around the mean, with fewer data points appearing as they approach the extreme ends. This pattern forms the "bell" shape, with the peak at its mean and tapering symmetrically in its tails.

By studying the boxplot, I knew my data would probably be normally distributed, since most of the values were centered on the median, which is typical for a normal distribution. Where I plotted a distribution plot i.e. a distplot whose nature happened to take the classic shape of a bell curve. That strengthened my observation, since the bell in the distplot further indicated that the data points were symmetrically dispersed: most values clustered around the mean while fewer values were in the tails.



#### **3. Poisson Distribution**

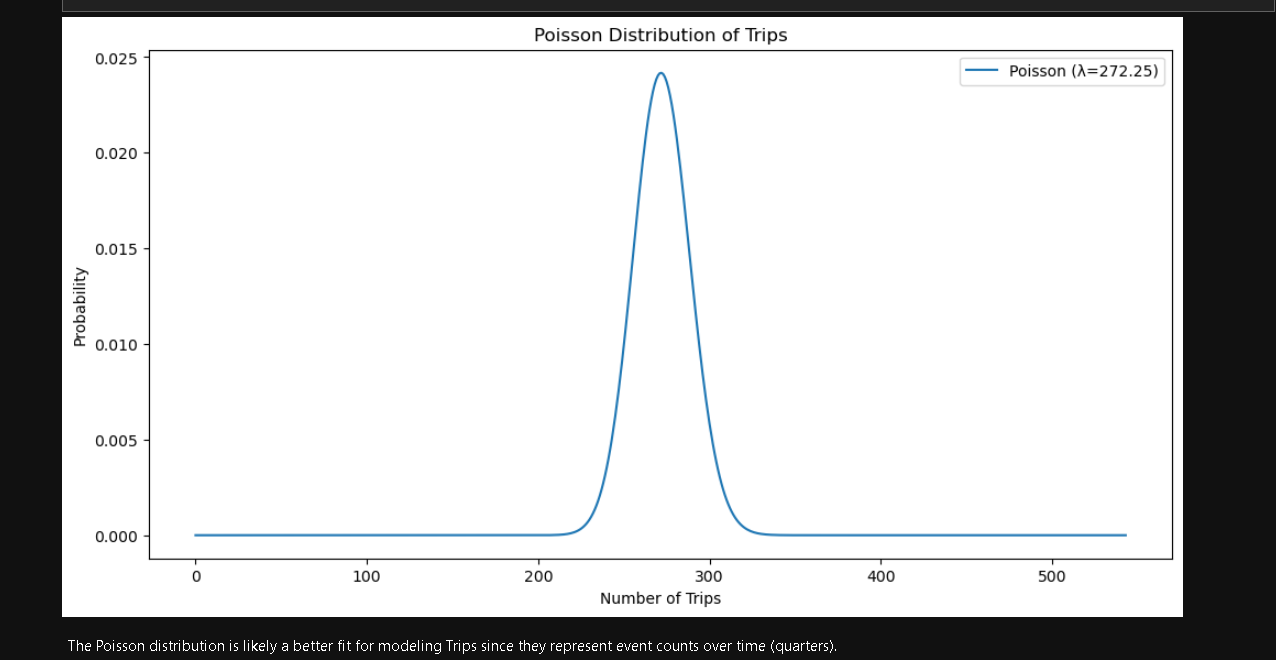
The Poisson distribution is a discrete probability distribution that usually comes into play when the probability is to be measured when a certain number of events take place within a fixed interval of time or space. It is quite useful when events are happening independently and at a constant average rate. In this context, the Poisson distribution computes the probability, given an event (k\) occurrence in a fixed period of time with a known average frequency.

This would be an ideal fit for the Poisson distribution on my dataset, which recorded the occurrence of trips over a defined period. Every formation of a trip is independent of others, and the number of occurrences differs between various time slots, which can be portrayed well by the Poisson distribution. For example, given the average number of trips in one quarter, the Poisson model would come in handy in estimating the possibility of seeing exactly a particular number of trips in the same unit of time.

The Poisson distribution provides more meaningful insights into the dataset by estimating the probability around the frequency of trips. This will help not only in predicting future events but also in understanding the fluctuation in trip frequency. Since the Poisson distribution can be fit to events happening independently over time, this model is much more realistic, and really mirrors the inherent characteristics of the dataset.

I apply such distribution for understanding the pattern and probability of trip occurrences to enhance my ability in making data-driven decisions based on the frequency and timing of trips within any given period of time.

**Machine Learning Techniques for Data Analytics**

In my perspective, machine learning techniques represent the future of data analytics, where more accurate and automated insights will be enabled. Supervised learning allows me to classify and perform predictions using labeled data examples, such as customer segmentation and forecasting. Unsupervised learning, by contrast, enables me to explore patterns in data without predefined categories, unveiling yet-to-be-discovered insights like groupings of customer behavior. Varieties of techniques that bring speed and maybe efficiency to my analysis include regression analysis, decision trees, and clustering. Deep learning models handle complex, high-dimensional data, applying techniques that will help enhance my derivations toward actionable insights in support of better decision-making for competitive advantage.

**CRISP-DM Framework:**

CRISP-DM is a structured approach toward data mining projects. This helps me in guiding through the various stages of the project by way of clarity and outcome. It comprises six important phases, which include Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. Each phase flows into the next, which helps me tackle challenging data projects in a structured manner that keeps me focused on the project objectives relevant to delivering value.

1. **Business Understanding:** On this level, I explain what the project objectives and requirements are from a business perspective. It’s about understanding wanting to solve what and why, thus allowing me to set direction and define success criteria. During this phase, all subsequent stages have its returned influence, because this makes sure my analysis is according to real-world goals.

2. **Data Understanding**: Once I have the business goals, I plunge deep into the data to assess data quality, relevance, and usability. This phase includes data gathering, data exploration, and preliminary assessment to catch issues or anomalies early in the process.

3. **Data Preparation:** This is where I clean the data by fixing inconsistencies, handling missing values, and transforming variables in readiness for analysis. Data preparation may take the longest time; it is, however, always very critical as far as getting accurate results is concerned.

4. **Modeling**: Now, I will apply various modeling techniques toward the analysis of the data. This includes choosing appropriate algorithms, tuning their parameters, and iteratively testing models to identify the best fit for both the data and the project goals.

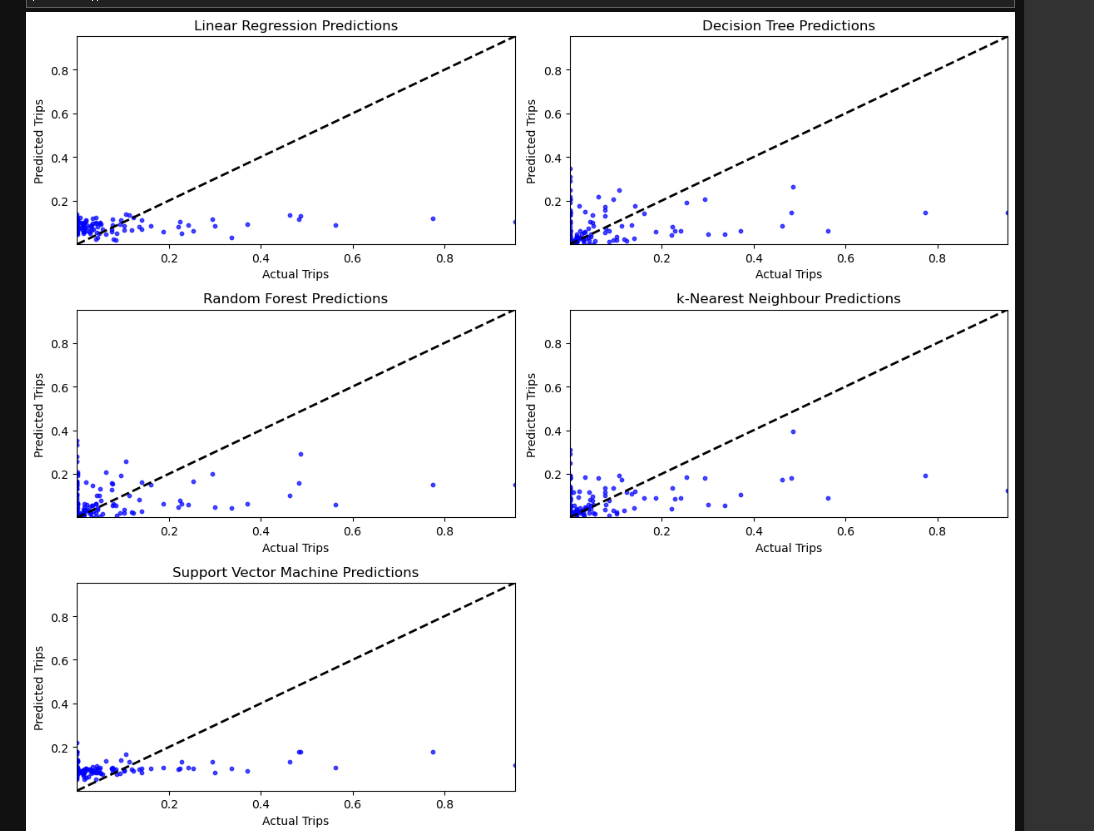
5. **Evaluation**: Once modeling is done, testing the performance of that model to see whether it has met the predefined goals of business. This will be the most important phase in validating any model for its accuracy and relevance.

6. **Deployment**: The final stage enters here whereby the model gets fitted into the business environment. This is either the actual sharing of insights with stakeholders or running the systems or mechanisms for continuous usage, hence translating work done to actionable outcomes.

CRISP-DM keeps my projects focused, structured, and outcome-driven.

**ML Models:**

I Tested five models on my dataset but they didn’t perform very well I checked for missing values I tried encoding the data I tried using StandardScaler and MinMaxscaler which did bring the R2 score in positive but didn’t give the results I hoped it would give. I would really appreciate some feedback on what went wrong so I don't make the same mistake again.



Just for me, looking through these prediction plots allows some understanding of how each model behaves on this data and which may be the most effective for this exercise.

**Random Forest**

One common performer among the best is Random Forest. It works by building an ensemble of decision trees, each trained on a random subsample of the data and I love it. Taking the best-case scenario of all those trees, the model is more stable and more accurate (which can be seen in how it leans further toward a straight diagonal line of actual vs. predicted values compared to other models in the plot). I appreciate this stability, as I am dealing with data that may have some noise/complexity in it. As Random Forest is less prone to overfitting while still being able to capture complex patterns, this verification process (that I find very intuitive) uses the power of masses and reduces risks by summation — individual random trees will make mistakes but their overall sum won’t.

**Decision Tree**

Decision Tree model also has done fair enough but does not have that sort of sturdiness as Random Forest. What I like about Decision Trees is the interpretability, splitting on feature values such that each leaf node represents a decision leading to an outcome. This makes them very appealing, but the downside of that simplicity is the fact that Decision Trees may overfit if not properly pruned/tuned. For me even though it was still not the best model, Decision Tree did reasonably fine modeling some patterns in data that showed more variance overall compared to Random Forest and less polarity to the perfect prediction line. It indicates that Decision Tree could be beneficial but to minimize the high variance it will likely require fine-tuning in order to generalize better on this data.

**Conclusion for Machine Learning:**

All in all, I would say Random Forest should be your first-go choice here since it strikes a good balance between accuracy and stability. But the Decision Tree still shows some promise, if I can get it better tuned. Through experimenting with both models, I have also been able to identify the advantages and drawbacks of each which will come in handy with adjusting my methodology and making sure I select the right model according to the parameter complexity of my dataset.

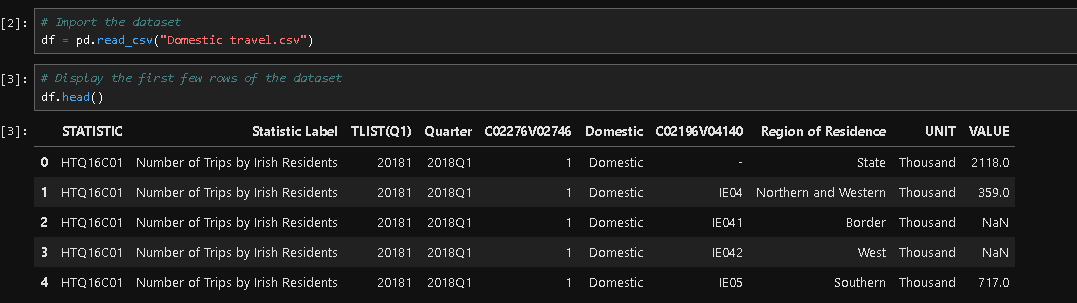
**Programming Paradigm**

A programming paradigm is a way to classify programming languages based on their features, and it represents an approach to solving problems with the input you have and telling the computer how this needs to be done. Over the period of my course, as I’ve performed a number of tasks in data analytics in Python from writing my first line of code that read “Hello World” to now working on a complex project where I get to analyze data from across Ireland, I came across different programming paradigms that have influenced my coding style. This is a report on imperative, procedural and object-oriented programming paradigms with the discussion of how every paradigm relates to data analysis task like those in the notebook I created.

**1.Imperative Programming Paradigm:**

An essential programming paradigm is the imperative one, which describes how a task should be performed (i.e. we give step-by-step instructions to the computer). This is a simple style, where you express some sequence of operations and then control program flow. As I process and manipulate data for analysis in Python, I often find myself using imperative programming, where each step in the procedure is important to the overall analysis.

As an example, all of the steps to load datasets, check for missing values or run preliminary calculations in my notebook were written using its imperative statements. Here’s an example:

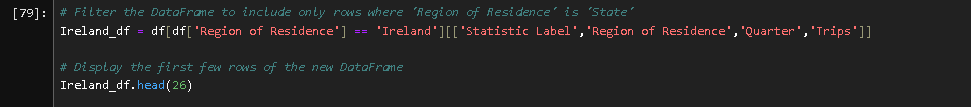


In our example, each line tells the computer exactly what to do. Such a process-oriented imperative style is well suited for data processing, where each step is part of an overarching workflow with operations that must be executed in a certain order to produce valid results.

**2.Procedural Programming Paradigm:**

A continental branch of the widely used imperative paradigm, procedural programming organizes code into reusable procedures or functions to improve modularity and readability. My favorite use of functions is to reuse and easily modify code, which I think is essential in data analysis workflows where similar operations (e.g. on applying the same data cleaning procedure) are run over multiple datasets.

I sat there building functions in my notebook to run automation for repetitive tasks, including entity cleaning and simple analysis. And a function would resemble something like this:



Since Ireland\_df contains the sequence of commands needed to prepare a dataset, I can call that function on as many datasets as necessary. This process-oriented method is time efficient. Also, it increases code readability, as each function fulfills one clearly defined task which means others (or my future self) will understand in a glance its part of the notebook.

It also improves troubleshooting for procedural programming. For example, if there is a problem with the data cleaning and Ireland\_df raises an error, I can simply change into that function in order to debug it without mixing code that is not directly related.

**3.Functional Programming paradigm:**

Functional programming focuses on pure functions and immutability, usually following a more declarative style that tells what you want to get done instead of how to do it. Functional programming is not the main paradigm I focus on in my notebook, but even though Python isn’t functional, it has functional features that I rely on to a certain degree for data transformations. Functional tools such as map(), filter() and lambda expression in python makes code more concise and expressive.

I also make use of a few lambda functions to transform some data, which is a common trick for doing short and anonymous functions on the fly while developing my code in the notebook. For example:

df[‘new\_column’] = df[‘existing\_column’]. apply(lambda x: x \* 2)

Given an existing column value, this line will double its value. The lambda function in Python is a functional programming construct that provides a concise way to define anonymous functions and transformations, allowing for elegant code expression without the overhead of defining full-fledged functions.

**Paradigms In Data Analytics**

Each paradigm has its strengths, contributing to the overall speed and structure of said code. Imperative programming provides readability and granularity of control, which is why I stick to it for scripting simple analysis pipelines. Now, procedural programming takes this one step further in that it allows me to break down task into reusable functions which comes really handy when dealing with datasets that only need the same cleaning and transformation steps. Lastly, I understand the merit of OOP for larger projects such as when working with multiple datasets or building data-processing pipelines that need structure and modularity.

The experience of working with these paradigms has reflected on how I approach data analytics projects, be it coding style or efficiency. Knowing when to use which paradigm enables me to choose the appropriate method behind code, a one-off analysis script, some reusable functions or classes for a larger project. Quick turnarounds also means I will have to write a lot of code but Python is so flexible, each notebook may include components from different paradigms, which usually requires especially organized designs like for the two solvers above.

As we concluded, programming paradigms offer critical structures to how code is written and performed with different benefits that support various task types. These paradigms help me in writing a structured, efficient, maintainable code for data analytics. And this multi-purposefulness is quite handy as I work towards improving in programming & also start dealing with more advanced data projects.

### **Conclusion and Insights**

The report dives into domestic tourism trends in Ireland, tracing developments from 2018 to 2024 and projecting forward to 2030. It uncovers that domestic tourism was on a steady upward trajectory until the disruption of the COVID-19 pandemic, which caused a sharp but temporary decline. Encouragingly, the sector has shown a strong rebound as travel restrictions eased, underscoring the resilience of Ireland's tourism industry.

Throughout this period, the Eastern and Midland, Northern and Western, and Southern regions have remained popular among domestic travelers. However, notable shifts emerged during the pandemic, with more tourists seeking out quieter, rural destinations likely due to safety considerations and a renewed interest in nature-oriented experiences.

To ensure accurate projections, the study evaluated five different forecasting models, ultimately finding that the Random Forest and Decision Tree models delivered the most reliable predictions. These tools offer actionable insights that could help policymakers and tourism planners prepare for expected growth in the sector. Furthermore, the report suggests that future studies might benefit from integrating additional factors like fuel costs and economic conditions to provide a fuller picture of influences on domestic travel trends. This comprehensive, data-driven approach equips stakeholders with the insights needed to support sustainable growth and strategic resource allocation across Ireland's tourism landscape.

### **References**

* Data.gov.ie. Available at: <https://data.gov.ie> / (Accessed: 8 November 2024).
* Pandas. Available at: <https://pandas.pydata.org/> (Accessed: 8 November 2024).
* Waskom, M.L. Seaborn: statistical data visualization. Available at: <https://seaborn.pydata.org/> (Accessed: 8 November 2024).
* Plotly, Python Graphing Library. Available at: <https://plotly.com/python> / (Accessed: 8 November 2024).
* Data Science Process Alliance CRISP-DM . Available at: <https://www.datascience-pm.com/crisp-dm-2/> (Accessed: 8 November 2024).
* NumPy ,NumPy documentation. Available at: <https://numpy.org> / (Accessed: 8 November 2024).
* Matplotlib.Matplotlib: visualization with Python. Available at: <https://matplotlib.org/> (Accessed: 8 November 2024).
* SciPy SciPy documentation. Available at: <https://docs.scipy.org/doc/scipy/reference/stats.html> (Accessed: 8 November 2024).
* Scikit-Learn, Machine learning in Python. Available at: <https://scikit-learn.org/stable> / (Accessed: 8 November 2024).