ROBERT GORDON UNIVERSITY ABERDEEN

SCHOOL OF COMPUTING

MSc Project Report

Student Name: Kirankumar Chaudhary	Matriculation Number: 2120051
Supervisor: Tiffany Young	Second Marker: Rob Lothian
Course: MSc Data Science	
Project Title: Energy Consumption Analysis and Optimisation for Scottish Councils	
Start Date: 07/02/2023	Submission Date: 25/04/2023

CONSENT

I agree

That the University shall be entitled to use any results, materials or other outcomes arising from my project work for the purposes of non-commercial teaching and research, including collaboration.

DECLARATION

I confirm:

- **That the work contained in this document has been composed solely by myself and that I have not made use of any unauthorised assistance.**
- **That the work has not been accepted in any previous application for a degree.**
- **All sources of information have been specifically acknowledged and all verbatim extracts are distinguished by quotation marks.**

Student Signature: Kirankumar	Date Signed: 25/04/2023

Acknowledgement

I would like to express my deepest gratitude to my project supervisor, Tiffany Young, for her unwavering support and guidance throughout the entirety of this project. Her expertise, patience, and dedication have been invaluable in helping me navigate the complexities of my research and address any doubts or questions that arose along the way. Thank you, Tiffany, for your unwavering support, and guidance.

Abstract

Energy consumption analysis plays a crucial role in understanding energy use dynamics and informing policy development. This project addresses a gap in the literature concerning energy consumption trends in Scottish council areas between 2005 and 2020 by analysing and visualizing data from open-source repositories provided by the UK Department for Business, Energy & Industrial Strategy (BEIS) [19]and the Scottish Government[16]. The study focuses on domestic, commercial, and public sectors, exploring variations in energy usage and types of fuels consumed by these sectors, such as petroleum products, gas, coal, electricity, bioenergy & wastes, and manufactured fuels within specific geographic locations.

Exploratory data analysis (EDA) will be performed using Python and Jupyter Notebook, followed by the creation of a customized, interactive dashboard using Power BI to enable a deeper understanding of energy consumption patterns and trends. Various machine learning models, such as linear regression, support vector machines, artificial neural networks, and ensemble methods, will be explored to identify the most suitable model for predicting energy consumption, with model selection guided by cross-validation and evaluation metrics.

The selected machine learning model will be deployed using Flask and Postman, creating a simple, lightweight, and flexible RESTful API. By filling the existing gaps in the literature and providing valuable insights, this project aims to contribute to the understanding of energy consumption trends in Scottish council areas from 2005 to 2020 and support future policy development and decision-making.

Contents

Acknowledgement.....2

Abstract.....3

1 Introduction.....6

1.1 Introduction 6

1.2 Motivation.....6

1.3 Content of the rest of the report 7

2 Literature Review 8

3 Project Specification.....13

3.1 Aim 13

3.2 Objectives 13

3.3 Functional and Non-Functional Requirements 13

3.4 Methodology 14

3.5 Project Plan 16

3.6 Review of legal, ethical, social, professional, and environmental issues.....17

3.7 Risks and safety 18

4 Design 19

5 Implementation 20

6 Evaluation of work 36

7 Conclusions and future works 37

References 38

List of Figures

Figure 5.1: Data Collected from Scotland Government Website in this form 20

Figure 5.2 Power BI dashboard Figure 29

Figure 5.3: Testing Deployed Model 35

1 Introduction

1.1 Introduction

As global energy demand continues to grow, there is an increasing need for sustainable energy management and efficient decision-making processes to reduce the environmental and economic impact of energy consumption. In the Scottish council area, optimizing energy use and understanding the factors influencing energy consumption patterns have become essential in reducing greenhouse gas emissions and conserving resources.

This project aims to address the knowledge gap in the literature concerning energy consumption trends in Scottish council areas between 2005 and 2020. By leveraging advanced data analytics and machine learning techniques, this study will provide a comprehensive understanding of the energy consumption patterns and types of fuels consumed across domestic, commercial, and public sectors. Through the analysis of data from open-source repositories provided by the UK Department for Business, Energy & Industrial Strategy (BEIS) [19] and the Scottish Government[16], the project seeks to uncover valuable insights that can drive targeted improvements and optimization efforts.

The project will begin with exploratory data analysis (EDA) using Python and Jupyter Notebook to identify trends and patterns in energy consumption. Subsequently, an interactive business intelligence dashboard will be developed using Power BI, enabling stakeholders to visualize and analyse energy consumption data effectively. Various machine learning models, such as linear regression, support vector machines, artificial neural networks, and ensemble methods, will be employed to develop accurate predictive models for energy consumption, with model selection guided by cross-validation and evaluation metrics.

Finally, the selected machine learning model will be deployed using Flask and Postman, creating a simple, lightweight, and flexible RESTful API. This project not only aims to contribute to the understanding of energy consumption trends in Scottish council areas but also supports future policy development and decision-making to foster sustainable development and a greener future for Scotland and beyond.

1.2 Motivation

The urgency of addressing energy consumption and sustainability concerns has become increasingly evident, particularly as global energy demand continues to grow. Efficient energy management and decision-making play a critical role in reducing the environmental and economic impact of energy consumption. The Scottish council area, like many other regions, faces the challenge of optimizing energy use to conserve resources, and reduce greenhouse gas emissions.

Developing accurate predictive models for energy consumption is essential for better strategic planning, policymaking, and infrastructure investments. Analysing energy

consumption data from 2005 to 2020 can uncover valuable insights into the factors affecting energy consumption trends and how they have evolved over time. This knowledge can drive targeted improvements and optimization efforts.

The motivation behind this project is to leverage advanced data analytics and machine learning to provide a comprehensive understanding of energy consumption patterns and energy type is consumed in the Scottish council area. By conducting exploratory data analysis (EDA) and developing predictive models, we can facilitate informed decision-making by various stakeholders, including government authorities, utility companies, and end-users. This project will ultimately contribute to sustainable development, promote cost-effective energy management, and foster a greener future for Scotland and beyond.

1.3 Content of the rest of the report

The subsequent chapters of this report are organized as follows:

Chapter 2 presents a comprehensive literature review, examining pertinent research and studies in the domains of energy consumption analysis, machine learning, and data visualization.

Chapter 3 delineates the project specifications, encompassing the aim, objectives, functional and non-functional requirements, methodology, project plan, and an examination of legal, ethical, social, professional, and environmental concerns. This chapter also addresses risk and safety considerations pertinent to the project.

Chapter 4 delves into the design aspects of the project, elucidating the architectural and functional components that facilitate the development of precise predictive models and interactive visualizations.

Chapter 5 concentrates on the implementation phase, expounding on the processes, techniques, and tools employed throughout the project, including data analysis, machine learning model development, and API deployment.

Chapter 6 offers an evaluation of the completed work, appraising the performance and efficacy of the predictive models, data visualization tools, and the project as a whole.

Lastly, Chapter 7 concludes the report by summarizing the salient findings, discussing the implications of the project outcomes, and providing recommendations for future research in the field of energy consumption analysis and prediction.

2 Literature Review

Energy consumption and its optimization have emerged as critical concerns in today's world, given the rising global energy demand, mounting environmental challenges, and the pressing need for sustainable energy systems. This project centres on collecting and analysing energy consumption data from the Scotland Council area spanning 2005 to 2020. The primary aim is to devise accurate predictive models for energy consumption and subsequently implement these models to optimize energy consumption patterns. This literature review delves into various aspects of energy consumption analysis, encompassing data collection, exploratory data analysis, business intelligence dashboards, predictive modelling, and optimization strategies. By thoroughly examining an array of studies and scholarly articles, this review elucidates the current state of research and best practices within the realm of energy consumption analysis and optimization.

Are there any data sources that should be considered for a comprehensive understanding of energy consumption patterns in the Scotland Council area?

The foundation of any successful energy consumption analysis and modelling lies in the accuracy and comprehensiveness of the data utilized. Numerous sources, including governmental and non-governmental organizations, provide open-source energy consumption data that can be leveraged for research and policy development. Notably, two key sources have emerged as crucial in the context of the Scotland Council area: the UK Department for Business, Energy & Industrial Strategy (BEIS) [19]and the Scottish Government [16]. These sources offer a wealth of data spanning various dimensions of energy consumption, providing researchers with ample opportunity to explore and analyse trends.

I am accessing data from the Scotland Government, which will allow me to gain insights into energy consumption patterns across different sectors, including domestic, commercial, and public sectors. Additionally, I will be able to observe variations in usage within specific geographic locations spanning from 2005 to 2020.

Is there a dashboard available for the energy consumption of Scottish council areas from 2005 to 2020, and has exploratory data analysis been conducted, or a machine learning model trained and deployed, on this energy consumption data?

I was unable to find any specific dashboard or publication addressing energy consumption in Scottish council areas from 2005 to 2020, nor was I able to find any machine learning models trained on this data. However, the UK Department of Energy & Climate Change [20] has published factsheet named as "Sub-national total final energy consumption statistics," which provides energy consumption exploratory data analysis (EDA) for sub-national regions in the UK spanning from 2005 to 2013, including Scotland.

Additionally, one can refer to the Scottish government's [17] official statistics on energy consumption. These sources provide comprehensive information on Scotland's energy consumption patterns and trends, but they not specifically focus on the council areas or cover the exact period from 2005 to 2020.

Given the available sources, I need to conduct an EDA, create a customized dashboard to analyse and visualize energy consumption trends in Scottish council areas during the specified period, i.e., from 2005 to 2020. I also need to train machine learning models and deploy them.

Which dashboard building software to use for energy consumption data?

Power BI and Tableau are both powerful data visualization and business intelligence tools [3]. They have their unique strengths and limitations, which makes them suitable for different use cases. Power BI and Tableau offer various features for creating visually appealing and interactive dashboards. Tableau is known for its advanced visualization capabilities, while Power BI is appreciated for its ease of use, integration with Microsoft products, and lower cost [3].

One advantage of Power BI over Tableau is its seamless integration with other Microsoft products, such as Excel, SharePoint, and Azure [3]. This makes it easier for users already familiar with the Microsoft ecosystem to adopt Power BI for creating dashboards [3]. Furthermore, Power BI provides a more consistent user experience across different platforms and devices (Biswal 2023) [3].

Power BI also has a more intuitive and user-friendly interface, making it easier for non-technical users to create dashboards without advanced coding or data analysis skills. The learning curve for Power BI is considered less steep compared to Tableau, which can make it more accessible for a wider audience. Another aspect where Power BI shines is in terms of cost-effectiveness. Power BI offers a more affordable pricing structure compared to Tableau, making it a better option for small businesses or organizations with limited budgets (Biswal 2023) [3].

In conclusion, after considering factors such as seamlessness integration with Microsoft Powerpoint, my familiarity with the tool, and ease of use, I have decided that Power BI is a better choice for me to create dashboards for this project.

Which programming language and software is best suited for performing exploratory data analysis (EDA)?

When performing exploratory data analysis (EDA) in energy consumption data for the Scotland Council, there are several programming languages and software options available. Two popular options are Python and R, which are both widely used in data analysis tasks.

Python is widely used for EDA due to its versatility, readability, and the availability of a vast ecosystem of libraries specifically designed for data analysis and visualization. Some of the most notable libraries include pandas, NumPy, and Matplotlib, which offer robust data manipulation and visualization capabilities, making the EDA process efficient and accessible (McKinney 2017) [11].

Additionally, Python's IPython and Jupyter Notebooks provide an interactive computing environment, enabling users to write, run, and visualize code all in one place, which is particularly useful for EDA. In "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython," Wes McKinney provides a comprehensive guide for utilizing Python's capabilities to perform EDA [11]. This book covers techniques and tools that enable users to efficiently clean, transform, and visualize data, thereby streamlining the process of discovering patterns and insights within the data (McKinney 2017) [11].

R, on the other hand, is a programming language specifically designed for statistical computing and graphics (Ihaka and Gentleman 1996) [9]. It provides a wide range of statistical and graphical techniques, with packages like ggplot2 and dplyr that cater specifically to EDA tasks

Based on above findings, I have decided to use Python and Jupyter Notebook for EDA due to my familiarity with the language and the abundance of tools and libraries available for data analysis and visualization tasks.

How do different machine learning techniques, such as linear regression, support vector machines, artificial neural networks, and ensemble methods, compare in terms of predictive accuracy, complexity, and computational requirements when applied to energy consumption data?

Different machine learning techniques have varying strengths and weaknesses when applied to energy consumption data, considering their predictive accuracy, complexity, and computational requirements.

Linear regression is a simple and interpretable model, making it easy to understand and implement. However, its simplicity limits its ability to capture complex relationships in the data, potentially leading to suboptimal predictive accuracy (James et al. 2013) [10]. Despite this limitation, linear regression may still perform well if the relationships in the energy consumption data are predominantly linear.

Support Vector Machines (SVM) are versatile models capable of handling both linear and nonlinear relationships by employing kernel functions. Compared to linear regression, SVMs may provide better predictive accuracy on complex datasets. However, they tend to be more computationally demanding, especially with large datasets, and their results may be more challenging to interpret (Cortes and Vapnik 1995) [4].

Artificial Neural Networks (ANN) are powerful models inspired by biological neural networks. They can model complex relationships and nonlinearities in the data, often leading to improved predictive accuracy compared to simpler methods. However, ANNs come with higher complexity and increased computational requirements, and their results may lack interpretability (Goodfellow et al. 2016) [5].

Ensemble methods, such as Random Forests, Gradient Boosting Machines (GBM), and eXtreme Gradient Boosting (XGBoost), combine multiple base models to improve predictive accuracy. These methods can handle complex data relationships and provide robust predictions, often outperforming single models. However, ensemble methods tend to be more computationally demanding and may require more extensive parameter tuning to achieve optimal performance (Hastie et al. 2009) [7].

In summary, each machine learning technique has its advantages and disadvantages when applied to energy consumption data. The choice of the best method depends on the specific dataset, available computational resources, and the desired level of interpretability. To find the most suitable technique for a given problem, it is often helpful to perform cross-validation and compare the performance of different models on the data at hand.

How do different evaluation metrics and criteria impact the selection of the most suitable predictive model for energy consumption analysis in varying contexts?

The selection of the most suitable predictive model for energy consumption analysis in varying contexts depends on multiple factors, including the evaluation metrics and criteria used. Evaluation metrics and criteria, such as R-squared, mean squared error (MSE), mean absolute error (MAE), and root mean squared error (RMSE), have different implications for model performance and can impact the selection of the most suitable regression model.

R-squared is a measure of the proportion of the variance in the dependent variable that can be explained by the independent variables in the model. While a higher R-squared indicates better model fit, it does not necessarily imply the most accurate predictions, especially in the presence of overfitting (James et al., 2013) [10]. In some cases, models with lower R-squared may produce more accurate predictions for unseen data due to better generalization.

MSE and RMSE are measures of the average squared difference between the predicted values and the true values. While they effectively penalize large errors, they can be sensitive to outliers and may not accurately reflect the model's performance on most of the data (Hyndman & Koehler, 2006) [8].

MAE, on the other hand, measures the average absolute difference between the predicted and true values. It is less sensitive to outliers compared to MSE and RMSE and can provide a better indication of model performance when dealing with skewed data or extreme values (Willmott & Matsuura, 2005) [21].

The choice of evaluation metric and criteria depends on the specific context of the energy consumption analysis, such as the importance of accurately predicting extreme values or the distribution of the target variable. Selecting the most suitable model may require a trade-off between the various evaluation metrics, and the use of cross-validation and domain knowledge to guide this selection process (James et al., 2013) [10].

It is essential to critically examine the evaluation metrics and their implications for the specific context to select the most suitable predictive model for energy consumption analysis.

Deploying machine learning model in local machine?

Deploying a machine learning model using Flask and Postman is a simple, lightweight, and flexible method compared to alternatives like Django and FastAPI. Flask allows easy creation of RESTful APIs, while Postman simplifies testing and interaction (Grinberg, 2018) [6]. Django is more suitable for full stack applications but can be overkill for serving machine learning models. FastAPI is more modern and performant, but Flask's simplicity and extensive documentation make it a more accessible choice for beginners (Allaire & Chollet,

2018) [1]. Postman's user-friendly interface provides a convenient way to test and validate the API endpoints without writing additional code.

Thus, I will be using Flask and Postman offer a suitable combination for deploying machine learning models on a local machine due to their simplicity, ease of use, and extensive resources.

In **conclusion**, the literature review highlights the need for an analysis of energy consumption trends in Scottish council areas from 2005 to 2020, as there is a lack of specific dashboards and machine learning models addressing this period. By conducting an EDA and leveraging Power BI for creating a customized dashboard, this project aims to fill that gap.

Python and Jupyter Notebook were chosen for EDA due to their versatility and the availability of various data analysis and visualization tools. The choice of the best machine learning model will depend on the specific dataset and the desired level of interpretability. Cross-validation and the evaluation of different metrics will be used to identify the most suitable model for energy consumption analysis in the Scottish council areas.

Flask and Postman were chosen for model deployment because of their simplicity, ease of use, and extensive resources. Overall, this project seeks to address the existing gaps in the literature and provide valuable insights into energy consumption trends in Scottish council areas between 2005 and 2020.

3 Project Specification

In this Project Specification chapter, the following sections are discussed:

3.1 Aim

The primary objective of this project is to gather comprehensive energy consumption data from Scottish council areas spanning the years 2005 to 2020. By utilizing Power BI, the goal is to develop an informative and interactive dashboard to effectively visualize this data. Subsequently, Python and Jupyter Notebook will be employed for conducting an in-depth exploratory data analysis (EDA) to uncover valuable insights, patterns, and correlations within the dataset. Lastly, create, refine, and deploy highly accurate predictive models to enable optimized energy consumption, contributing to more sustainable and efficient energy management within these council areas.

3.2 Objectives

- Collect energy consumption data for Scottish council area spanning over 2005-2020 from open-source websites.
- Perform exploratory data analysis to gain insights into the data, identify patterns, correlations, and anomalies.
- Develop business intelligence dashboard to effectively visualize the data.
- Develop machine learning models to predict energy consumption accurately.
- Deploy the machine learning models and the algorithms for optimization of energy consumption.

3.3 Functional and Non-Functional requirements.

For this project on gathering energy consumption data for Scottish council areas, creating a dashboard, conducting EDA, and developing accurate predictive models, the functional and non-functional requirements can be described as follows:

3.3.1 Functional Requirements:

Data Collection: Collect comprehensive energy consumption data for Scottish council areas from 2005 to 2020 from Scotland Government.

Exploratory Data Analysis: Perform in-depth EDA using Python and Jupyter Notebook to uncover insights, patterns, correlations, and anomalies within the energy consumption data.

Dashboard Creation: Develop an interactive and informative Power BI dashboard to visualize the energy consumption data, enabling users to easily explore and understand patterns and trends.

Model Development: Create accurate predictive models for energy consumption based on the analysed data, using machine learning algorithms and techniques.

Model Evaluation: Assess the performance of the developed models using suitable evaluation metrics and cross-validation techniques.

Deployment: Deploy the predictive models for practical use to optimize energy consumption and contribute to efficient energy management.

3.3.2 Non-Functional Requirements:

Usability: The dashboard and models should be user-friendly, with clear, intuitive, and easy-to-understand interfaces and visualizations.

Scalability: The developed models and tools should be able to handle increasing volumes of data as new information on energy consumption becomes available.

Performance: The models, EDA, and dashboard should have fast response times and low latency to provide a seamless user experience.

Maintainability: The code and tools used for the project should be well-structured, modular, and easy to maintain, allowing for future updates and improvements.

Security: Any sensitive data involved in the project should be properly protected, ensuring the privacy and confidentiality of the information.

Reliability: The models, EDA, and dashboard should provide accurate, consistent, and reliable results, with rigorous testing and validation processes in place.

Documentation: Comprehensive documentation should be provided for all aspects of the project, detailing the methods, tools, and processes used, as well as instructions for future maintenance and updates.

3.4 Methodology

Data Collection:

The energy consumption data for Scottish council areas spanning from 2005 to 2020 will collected from the Scottish government website [16]. This dataset will provide comprehensive information on energy consumption across various sectors, energy types, and geographical regions.

Exploratory Data Analysis (EDA):

Using Python and Jupyter Notebook, an exploratory data analysis (EDA) will perform to gain insights into the dataset, uncover patterns, and identify trends in energy consumption. The EDA includes

- Importing Data.
- Data Inspection.
- Univariate Analysis
- Bivariate Analysis

- Getting insights through various charts

Business Intelligence Dashboard:

A Business Intelligence (BI) dashboard will be developed using Power BI to provide an interactive and visual representation of the energy consumption trends and patterns. The dashboard allowed users to explore the data and make informed decisions on energy efficiency and sustainable energy initiatives.

Machine Learning Models:

Four machine learning models will develop to predict energy consumption: Linear Regression, Support Vector Machines (SVM), Artificial Neural Networks (ANN), and XGBoost. The models will be trained and evaluated using cross-validation, and their performance will be compared using R-squared (R2) score, Mean Absolute Error (MAE), and Mean Squared Error (MSE).

Model Deployment:

The best model will be deployed using Flask, a lightweight Python web framework, and tested using Postman, an API testing tool.

Alternative methodology:

The data collection process would remain the same as in the main methodology. Energy consumption data for Scottish council areas spanning from 2005 to 2020 would be collected from the Scottish government website. Instead of using Python and Jupyter Notebook for EDA, R programming language and RStudio could be utilized to perform the analysis. R offers a powerful environment for statistical analysis and data visualization, enabling a comprehensive understanding of the dataset. In this alternative methodology, Tableau or Looker could be used as dashboard development tools to create interactive visualizations of the energy consumption data. These tools offer alternative approaches to data exploration and visualization, providing users with a platform to make data-driven decisions. Rather than developing multiple machine learning models, this methodology would focus on a single model, such as Random Forest or Decision Tree, for predicting energy consumption. This approach simplifies the model selection process and reduces the computational overhead, allowing for faster model development and deployment. The alternative deployment options could involve leveraging cloud-based services like AWS SageMaker or Google AI Platform. These services offer greater scalability and easier integration with other applications, making it more convenient for users to access the energy consumption predictions.

The chosen methodology is better for the following reasons:

By developing and comparing multiple machine learning models, the best-performing model for predicting energy consumption can be identified. This ensures that the most accurate predictions are obtained, which is essential for informed decision-making. The use of cross-validation during model development provides a robust measure of model performance,

reducing the likelihood of overfitting and improving the generalizability of the model to new data. Deploying the model using Flask and Postman allows for easy integration with other applications and services, as well as providing a simple way to test and validate the model's performance. Using Power BI for dashboard development enables the creation of interactive and visually appealing visualizations, making it easier for users to explore and understand the energy consumption data. Python and Jupyter Notebook are widely used for data analysis due to their flexibility and popularity among data scientists and analysts. This popularity ensures a vast community support and a plethora of libraries, tools, and resources, making it easier for analysts to perform complex tasks and explore new techniques.

Overall, the chosen methodology, which incorporates Python and Jupyter Notebook for EDA, multiple machine learning models, and deployment using Flask and Postman, provides a well-rounded, flexible, and efficient approach to energy consumption data analysis and model development, ensuring the best possible outcome for predicting energy consumption values.

3.5 Project Plan

A well-structured project plan is essential to ensure the successful completion of any project. Here is a detailed project plan for my project:

Data Collection:

Collect energy consumption data for Scottish council areas from 2005-2020 from the Scottish government website.
Review and validate the data to ensure its quality and relevance.

Exploratory Data Analysis (EDA):

Perform EDA using Python and Jupyter Notebook.
Conduct data inspection, univariate and bivariate analyses
Create charts to visualize findings and gain insights.

Dashboard Development:

Develop a Business Intelligence (BI) dashboard using Power BI.
Incorporate interactive visualizations to allow users to explore the data and discover insights.

Machine Learning Model Development:

Label encoding of categorical values as machine learning algorithms takes only numerical input. Split the data into training, validation, and testing sets.

Develop multiple machine learning models, including Linear Regression, SVM, ANN, and XGBoost.

Train, validate, and test each model, measuring their performance using metrics such as R-squared (R2) score, Mean Absolute Error (MAE), and Mean Squared Error (MSE).

Model Selection:

Compare and evaluate the performance of the developed models.
Select the best-performing model for deployment.

Model Deployment:

The best model will be deployed using Flask, a lightweight Python web framework, and tested using Postman, an API testing tool.

Documentation and Reporting:

Document the entire project, including methodology, code, results, and insights.
Create a final report summarizing the project findings, key insights, and model performance.
Project Closure:

Upon completion of the project, the energy consumption data will be thoroughly analysed and visualized, and an accurate predictive model will be deployed. The results will contribute to better energy management practices and promote sustainable energy consumption within Scottish council areas. By following this project plan, i can ensure a structured, organized approach to energy consumption analysis project, ultimately leading to better insights, more accurate predictions, and a successful outcome.

3.6 Review of legal, ethical, social, professional, and environmental issues

In this project, there are several legal, ethical, social, and professional (LESP) issues that need to be addressed, ensuring compliance and responsible conduct during the execution and implementation of the project.

Legal issues: The project utilizes energy consumption data from open-source repositories provided by the Scottish Government. This project is complied with relevant data protection laws, such as the General Data Protection Regulation (GDPR) and the UK Data Protection Act 2018. Any third-party tools or software used in the project adhered to licensing agreements and intellectual property laws.

Ethical issues: Data privacy and confidentiality are paramount concerns in the handling and analysis of energy consumption data. The project ensured that no personally identifiable information (PII) is extracted, stored, or disclosed during data processing.

Social issues: The project's outcomes may influence policy development and decision-making related to energy consumption in the Scottish councils. Therefore, I maintained transparency and impartiality in the data analysis and model development processes.

Professional issues: The project conducted following best practices and professional standards in data analysis and machine learning. I maintained integrity and honesty in the interpretation and presentation of the findings, avoiding any potential bias or

misrepresentation. All software tools and resources used in the project is properly acknowledged and credited.

3.7 Risks and safety

In this project, it is essential to identify and address potential risks and safety issues associated with the collection, analysis, and visualization of energy consumption data. The key risks and safety concerns to consider include:

Data security risks: When handling and processing open-source energy consumption data, there is risk of data breaches or unauthorized access.

Data quality and reliability risks: The accuracy and reliability of the project's outcomes depend on the quality of the input data. Errors, inconsistencies, or gaps in the data can lead to inaccurate or misleading results. To address this risk, the project implemented rigorous data inspection, checked for null values, duplicates, and unrequired records. After that, the data was cleaned.

Model overfitting risks: In developing machine learning models for energy consumption prediction, there is a risk of overfitting, which occurs when the model performs exceptionally well on the training data but poorly on unseen data. To mitigate this risk, the project employed cross-validation techniques, label encoding, and data splitting into three parts - train, test, and validation. These measures will help to ensure the accuracy and reliability of the project's outcomes.

Intellectual property and licensing risks: The project incorporated third-party tools, software, or datasets that are subject to intellectual property and licensing restrictions. Infringement on these rights could result in legal disputes or penalties. To mitigate this risk, the project carefully reviewed and adhered to the terms and conditions of any licenses, copyrights, or trademarks associated with third-party resources.

4 Design

The detailed design of the project provides an in-depth explanation of the various components, their interconnections, and the overall structure of the system. This chapter will describe the design process for each stage of the project, including data collection, exploratory data analysis, dashboard development, machine learning model creation, and model deployment.

The first stage of the project involves collecting energy consumption data for Scottish council areas from 2005 to 2020. This data will be sourced from the Scottish government website and will be loaded into a pandas DataFrame for further processing.

The EDA will be conducted using Python and Jupyter Notebook. During this stage, various statistical analyses and visualizations will be performed to identify trends, patterns, and correlations within the dataset. Univariate and bivariate analyses will be conducted, and charts will be generated using libraries such as Matplotlib and Seaborn.

An interactive dashboard will be developed using Power BI to visualize the energy consumption data. The dashboard will include features such as filters, sliders, and dropdown menus to allow one to explore the data and gain insights into energy consumption trends across different council areas, energy types, and sectors.

Data preprocessing will include encoding categorical variables using LabelEncoder. Then Four machine learning models will be developed for predicting energy consumption: Linear Regression, Support Vector Machines (SVM), Artificial Neural Networks (ANN), and XGBoost. Each model will be trained and evaluated using cross-validation, and their performance will be compared using R-squared (R2) score, Mean Absolute Error (MAE), and Mean Squared Error (MSE).

The best-performing model will be selected and deployed using Flask, a lightweight Python web framework. The Flask application will expose a REST API to receive input data and return predictions. The input features will be preprocessed and encoded using the previously created LabelEncoders, and the model will make predictions based on the input. The predicted energy consumption will be returned in a JSON format.

The best model will be deployed using Flask, a lightweight Python web framework, and tested using Postman, an API testing tool.

In conclusion, the detailed design of the project provides a comprehensive description of each stage and the components involved in the system. This design serves as a blueprint for the development and implementation of the project, ensuring that all aspects are well-planned and executed efficiently.

5 Implementation

Collecting energy consumption data:

Initially, the data was collected in CSV format from the website of the Statistics of the Scottish government. When downloading the data, all possible attributes were selected to ensure comprehensive coverage. The resulting data comprises nine features, a sample of which is provided below:

	A	B	C	D	E	F	G	H	I	J
1	FeatureCode	FeatureNam	FeatureType	DateCode	Measuremer	Units	Value	Energy Type	Energy Consuming Sector	
2	S12000039	West Dunbar	Council Area	2005	Count	GWh	0	Coal	Rail	
3	S12000039	West Dunbar	Council Area	2005	Count	GWh	192.383	Electricity	Domestic	
4	S12000039	West Dunbar	Council Area	2005	Count	GWh	0.176	Coal	Public Sector	
5	S12000039	West Dunbar	Council Area	2005	Count	GWh	664.98	Gas	Domestic	
6	S12000039	West Dunbar	Council Area	2005	Count	GWh	250.801	Electricity	Industrial & Commercial	
7	S12000039	West Dunbar	Council Area	2005	Count	GWh	868.701	All	Domestic	
8	S12000039	West Dunbar	Council Area	2005	Count	GWh	1.127	Bioenergy &	Domestic	
9	S12000039	West Dunbar	Council Area	2005	Count	GWh	639.841	All	Industrial & Commercial	
10	S12000039	West Dunbar	Council Area	2005	Count	GWh	0.182	Coal	Domestic	
11	S12000039	West Dunbar	Council Area	2005	Count	GWh	0.005	Coal	Agriculture	
12	S12000039	West Dunbar	Council Area	2005	Count	GWh	528.53	Petroleum Pr	Road Transport	
13	S12000039	West Dunbar	Council Area	2006	Count	GWh	1.268	Bioenergy &	Domestic	
14	S12000039	West Dunbar	Council Area	2005	Count	GWh	1.044	Petroleum Pr	Rail	
15	S12000039	West Dunbar	Council Area	2006	Count	GWh	0.164	Coal	Domestic	
16	S12000039	West Dunbar	Council Area	2006	Count	GWh	0.003	Coal	Agriculture	
17	S12000039	West Dunbar	Council Area	2005	Count	GWh	7.846	Petroleum Pr	Agriculture	
18	S12000039	West Dunbar	Council Area	2005	Count	GWh	0.599	Manufacture	Domestic	
19	S12000039	West Dunbar	Council Area	2005	Count	GWh	0.883	Petroleum Pr	Public Sector	
20	S12000039	West Dunbar	Council Area	2005	Count	GWh	9.429	Petroleum Pr	Domestic	
21	S12000039	West Dunbar	Council Area	2009	Count	GWh	553.147	All	Transport	
22	S12000039	West Dunbar	Council Area	2009	Count	GWh	537.979	All	Industrial & Commercial	
23	S12000039	West Dunbar	Council Area	2010	Count	GWh	730.552	All	Domestic	
24	S12000039	West Dunbar	Council Area	2008	Count	GWh	547.136	All	Transport	
25	S12000039	West Dunbar	Council Area	2009	Count	GWh	744.803	All	Domestic	
26	S12000039	West Dunbar	Council Area	2011	Count	GWh	701.244	All	Domestic	
27	S12000039	West Dunbar	Council Area	2011	Count	GWh	570.209	All	Industrial & Commercial	
28	S12000039	West Dunbar	Council Area	2010	Count	GWh	578.308	All	Industrial & Commercial	
29	S12000039	West Dunbar	Council Area	2010	Count	GWh	539.78	All	Transport	
30	S12000039	West Dunbar	Council Area	2006	Count	GWh	544.605	All	Transport	
31	S12000039	West Dunbar	Council Area	2006	Count	GWh	636.192	All	Industrial & Commercial	
32	S12000039	West Dunbar	Council Area	2007	Count	GWh	830.119	All	Domestic	
33	S12000039	West Dunbar	Council Area	2005	Count	GWh	530.528	All	Transport	
34	S12000039	West Dunbar	Council Area	2006	Count	GWh	840.517	All	Domestic	
35	S12000039	West Dunbar	Council Area	2008	Count	GWh	802.531	All	Domestic	

Figure 5.1: Data Collected from Scotland Government Website in this form

Exploratory Data Analysis:

In the article "An Extensive Step by Step Guide to Exploratory Data Analysis" by Terence Shin, the author provides a comprehensive guide to performing Exploratory Data Analysis (EDA) using Python[18].

The author begins by introducing the concept of EDA and its importance in the data analysis process. EDA is an essential step for understanding the data, identifying patterns, trends, and outliers, and forming hypotheses for further analysis[18].

The article explains how to set up the necessary Python libraries, such as Pandas and NumPy, and import a dataset for analysis [18].

Initial Terence Shin guides the reader through the process of initially inspecting the dataset, including checking the data types, identifying missing values, and examining summary statistics [18].

Univariate Analysis: The author demonstrates how to perform univariate analysis on continuous and categorical variables using techniques like histograms, box plots, and bar charts [18].

Bivariate Analysis: This section covers bivariate analysis, which involves analyzing the relationship between two variables. Techniques discussed include scatter plots, correlation matrices, and groupby aggregations [18].

The article covers the preprocessing steps required before training machine learning models, including scaling and normalization, encoding categorical variables, and splitting the data into training and testing sets [18].

Upon gathering insights from the aforementioned article by Terence Shin and the literature review, the exploratory data analysis (EDA) process was initiated. To facilitate the analysis, the Anaconda software suite was utilized, and the Jupyter Notebook platform was launched. This enabled a seamless and structured approach to conduct EDA on the energy consumption dataset for the Scottish council area.

Data Loading

Importing the dataset into a pandas DataFrame in Python [14] which named as ‘’scot_data”. This allows for easier manipulation and analysis of the data.

```
# Set the file path to the Scotland Council Data .csv file
scot_csv_file_path = "Data/Scotland Council Data .csv"
# Load the data from the CSV file into a pandas DataFrame
scot_data = pd.read_csv(scot_csv_file_path)
# Print the length of the DataFrame to confirm that the data has been loaded correctly
print("length of Scotland Data " + str(len(scot_data)))
```

```
length of Scotland Data 19008
```

[22]

Data Inspection

Importing Required Library

```
import pandas as pd
import numpy as np
import seaborn as sns
```

```
import matplotlib.pyplot as plt
```
[22]

Checking number of distinct councils in Scotland
In "scot_data" dataframe, the "FeatureName" column contains the names of all councils in Scotland.

An overview of the distinct council names present in the data.

```
# checking distinct in "FeatureName" which store councils names of scotland
councils = scot_data['FeatureName'].unique()
```

[22]
Upon closer inspection, it is evident that the list includes 'Scotland' as one of the council names. However, since 'Scotland' represents the entire country, and our analysis is focused on individual councils, we need to remove this entry from our dataset. By doing so, we ensure that our analysis remains relevant to the individual councils.

```
# removing "Scotland" named records from "FeatureName"
scot_data = scot_data[~(scot_data["FeatureName"]=="Scotland")]
print("length of Scotland Data " + str(len(scot_data)))
# total number of councils in Scotland
print("Total number of councils in Scotland : ", scot_data['FeatureName'].nunique())
```

```
length of Scotland Data 18432
Total number of councils in Scotland :  32
```
[22]

After eliminating the rows with 'FeatureName' equal to 'Scotland' from the "scot_data" DataFrame, the resulting cleaned DataFrame, scot_data, consists of 18,432 rows. Moreover, utilizing the .nunique() method on the 'FeatureName' column of scot_data reveals that there are a total of 32 unique councils in Scotland. Consequently, by removing the unnecessary data related to the entire country of Scotland, we are left with energy consumption information specific to the council areas. This refines the dataset and simplifies the analysis of relevant data for our study.

Checking data type of each column

```
print(scot_data.dtypes)
```
```
FeatureCode                object
FeatureName                object
FeatureType                object
DateCode                    int64
Measurement                object
Units                      object
Value                     float64
Energy Type                object
Energy Consuming Sector    object
```

```
dtype: object
[22]
```

It is evident that a majority of the columns possess the 'object' data type, signifying the pres ence of categorical data. In particular, the 'FeatureCode', 'FeatureName', 'FeatureType', 'Me asurement', 'Units', 'Energy Type', and 'Energy Consuming Sector' columns belong to this cla ssification. Conversely, the 'DateCode' column exhibits an 'int64' data type, indicating intege r values. Finally, the 'Value' column is characterized by the 'float64' data type, reflecting nu meric data with decimal values or floating-point numbers.

Recognizing the data types of each column is essential for our exploratory data analysis, as it enables us to pinpoint the appropriate preprocessing steps and analytical techniques to employ.

```
# printing the dimensions
print(scot_data.shape)
```

```
(18432, 9)
[22]
```

The output reveals that the dataset comprises 18,432 rows and 9 columns. The presence of 18,432 rows indicates that the dataset is relatively large, potentially providing ample data for training machine learning models. However, it remains crucial to assess the data's quality and completeness.

The dataset encompasses nine columns, with each representing a distinct feature. Throughout the exploratory data analysis, a thorough examination of each column is necessary to discern potential patterns, relationships, or anomalies that may impact our conclusions or guide our selection of analytical approaches.

Null values in dataset

```
# Count missing values for each column
missing_values = scot_data.isnull().sum()

# Print the results
print("Missing values in each column:")
print(missing_values)
```

```
Missing values in each column:
FeatureCode                0
FeatureName                0
FeatureType                0
DateCode                   0
Measurement                0
Units                      0
Value                      0
Energy Type                0
Energy Consuming Sector    0
dtype: int64
[22]
```

After examining the missing values in each column, it is evident that there are no missing values in this dataset. This is a positive aspect of the data quality, as it indicates that the dataset is complete, and no imputation techniques are required to fill in missing information. It simplifies the data pre-processing phase, and we can proceed to explore the dataset further with confidence in its integrity.

Summary Statistics

Summary statistics for the numerical columns using the describe() function:

```
print("Summary statistics for numerical columns:")
print(scot_data.describe())
```

```
Summary statistics for numerical columns:
           DateCode          Value
count  18432.000000  18432.000000
mean    2012.500000    543.465395
std        4.609897   1491.185520
min     2005.000000      0.000000
25%     2008.750000      2.360500
50%     2012.500000     43.496500
75%     2016.250000    488.722250
max     2020.000000  25041.502000
```

[22]
DateCode: The DateCode variable ranges from 2005 to 2020. The mean and median values are both 2012.5, indicating a uniform distribution across the years. This means that the dataset spans 16 years of energy consumption data and is equally distributed across the years.

Value: The energy consumption values (Value) range from a minimum of 0 to a maximum of 171,340.421. The mean value is 1,054.76, while the median is 53.56, which suggests a highly skewed distribution with a long tail towards higher consumption values.

The large difference between the mean and median values also indicates the presence of outliers, which might need further investigation.
Given that the dataset has been sourced from the Scottish Government, it is reasonable to assume that the data is accurate and reliable. As such, there is no immediate need to carry out further investigation into the extreme values or outliers. We can proceed with the analysis, bearing in mind that these extreme values are an inherent part of the dataset and may still influence the analysis and modeling processes.

Checking Duplicates

```
# Identify and remove duplicate rows
duplicates = scot_data.duplicated()
print("Number of duplicate rows:", duplicates.sum())
scot_data = scot_data[~duplicates]
```

```
Number of duplicate rows: 0
```

Upon checking for duplicate rows in the dataset, it has been found that there are no duplicates present.

This is a positive aspect of the dataset, as duplicate entries could potentially skew the results of any analysis or modeling conducted on it.

The absence of duplicates indicates that the dataset is relatively clean and well-prepared, allowing for a more accurate and reliable exploration of the data in subsequent stages.

Checking unecessary values in Energy Type

```
# storing distinct values of "Energy Type"
dist_Energy_Type = scot_data['Energy Type'].unique()
# printing values in "dist_Energy_Type" variable
print("Distinct Energy Type: ", dist_Energy_Type)
# printing length of "dist_Energy_Type" variable
print("\n\nTotal number of distinct Energy Type is : " + str(len(dist_Energy_Type)))
```

```
Distinct Energy Type:  ['Coal' 'Electricity' 'Gas' 'All' 'Bioenergy & Waste
s'
 'Petroleum Products' 'Manufactured Fuels']


Total number of distinct Energy Type is : 7
```

[22]

The "All" record in the "Energy Type" column does not actually represent a specific energy type but rather the sum of values across all energy types in the dataset. Therefore, it may be appropriate to remove this row from the dataset to avoid any confusion.

```
# Find the rows where 'Energy Type' is not 'All'
scot_data = scot_data[scot_data['Energy Type'] != 'All']

# Check the new number of rows
print("Number of rows after removing 'All' from 'Energy Type':", scot_data.shape[0])

# Print the updated frequency table for 'Energy Type'
print("Updated frequency table for 'Energy Type':")
print(scot_data['Energy Type'].value_counts())
```

```
Number of rows after removing 'All' from 'Energy Type': 16384
Updated frequency table for 'Energy Type':
Petroleum Products    4096
Coal                  3584
Electricity           2560
Bioenergy & Wastes    2560
Gas                   2048
Manufactured Fuels    1536
Name: Energy Type, dtype: int64
```

Checking unnecessary values in Energy Consuming Sector

```
# storing distinct values of "Energy Consuming Sector"
dist_Energy_Consuming_Sector = scot_data['Energy Consuming Sector'].unique()
# printing values in "dist_Energy_Consuming_Sector" variable
print("Distinct Energy Consuming Sector: ", dist_Energy_Consuming_Sector)
# printing length of "dist_Energy_Consuming_Sector" variable
print("\n\nTotal number of distinct Energy Consuming Sector is : " + str(len(dist_Energy_Consuming_Sector)))
```

```
Distinct Energy Consuming Sector:  ['Rail' 'Domestic' 'Public Sector' 'Industrial & Commercial' 'Agriculture'
 'Road Transport' 'Commercial' 'Industrial' 'All']


Total number of distinct Energy Consuming Sector is : 9
```

[22]

The "All" record in the "Energy Consuming Sector" column does not actually represent a specific energy consumption sector but rather the sum of values across all energy consuming sector in the dataset. Therefore, it may be appropriate to remove this row from the dataset to avoid any confusion.

```
# Find the rows where 'Energy Type' is not 'All'
scot_data = scot_data[scot_data['Energy Consuming Sector'] != 'All']

# Check the new number of rows
print("Number of rows after removing 'All' from 'Energy Consuming Sector':", scot_data.shape[0])

# Print the updated frequency table for 'Energy Type'
print("Updated frequency table for 'Energy Type':")
print(scot_data['Energy Consuming Sector'].value_counts())
```

```
Number of rows after removing 'All' from 'Energy Consuming Sector': 13312
Updated frequency table for 'Energy Type':
Domestic                  3072
Industrial                3072
Commercial                2560
Rail                      1024
Public Sector             1024
Agriculture               1024
Road Transport            1024
Industrial & Commercial    512
Name: Energy Consuming Sector, dtype: int64
```

[22]

Cleanied it!

```
print("Length of cleaned Dataframe : ", str(len(scot_data)))
```

```
Length of cleaned Dataframe :  13312
```

After completing the necessary data cleaning and inspection steps, we are left with a cleaned dataset containing 13,312 clear records.

These records contain only the relevant energy consumption data for the 32 council areas in Scotland, with any duplicates, missing values, or unnecessary data removed.

This clean dataset is now ready for further analysis, such as exploratory data analysis or building machine learning models.

Univariate Analysis

All the univariate analysis, chart and code provide in “main.ipynb” due to limitation of word, it is better to write conclusion of univariate analysis in the report rather than whole process.

The "scot_data" dataset has provided valuable insights into the distribution and characteristics of each variable. Key observations from the univariate analysis include:

The dataset is focused solely on council areas as the main feature type for the given data points. There are 32 distinct council areas (FeatureName) in the dataset.

The data is uniformly distributed across years, with no significant gaps or inconsistencies, which ensures reliable and consistent analysis.

The energy consumption values (Value) have a highly right-skewed distribution, with some areas having significantly higher energy consumption than others. The median provides a better representation of the central tendency of the data in such cases.

The distinct energy types and energy-consuming sectors identified in the dataset provide an overview of the various sources and sectors of energy consumption in Scotland.

The univariate analysis also identified redundancy and a lack of variability in some columns, such as 'FeatureType', 'Measurement', 'Units', and 'FeatureCode'. Removing these columns simplifies the dataset, reduces complexity, and makes it more manageable for further analysis. It also improves data visualization, making it more accessible to a broader audience.

Based on the insights from the univariate analysis, the next steps would involve further bivariate and multivariate analyses to explore relationships and interactions between variables, identify trends and patterns, and derive meaningful insights to inform decision-making and policy recommendations. [22]

Bivariate Analysis

All the bivariate analysis, chart and code provide in “main.ipynb” due to limitation of word, it is better to write conclusion of univariate analysis in the report rather than whole process.

The bivariate analysis of energy consumption in Scotland between 2005 and 2020 reveals several important insights.

There has been a general downward trend in total energy consumption, possibly attributable to increased energy efficiency measures, a shift towards cleaner energy sources, and changes in the economy or industrial activities during this period.

Regional disparities in energy consumption across different council areas were observed, with Falkirk, Fife, and Glasgow City having the highest energy consumption levels, and Orkney Islands, Na h-Eileanan Siar, and Shetland Islands having the lowest. Understanding these regional patterns is crucial for developing targeted policies and resource allocation strategies to promote sustainable development.

The regional analysis of energy consumption shows that there are substantial differences in consumption patterns among the council areas. Glasgow City and Edinburgh consistently appear as top consumers in several sectors and energy types, likely due to their large populations and industrial presence. The data also highlights specific regions with high consumption within certain energy-consuming sectors, such as agriculture in Dumfries & Galloway or industrial activity in Falkirk.

Energy consumption varied across different energy-consuming sectors, with the Industrial, Domestic, and Road Transport sectors accounting for the most significant energy use. Additionally, it was found that Petroleum Products, Gas, and Electricity were the primary energy types used in these sectors. Therefore, promoting cleaner energy alternatives and improving energy efficiency should be targeted, especially in the most energy-consuming sectors.

Overall, the visualizations and observations from this analysis can serve as a valuable resource for policymakers, businesses, and communities in Scotland. By identifying the key trends and regional differences, targeted interventions can be implemented to further promote energy efficiency, adopt cleaner energy sources, and work towards a more sustainable energy future. [22]

Power BI Dashboard:

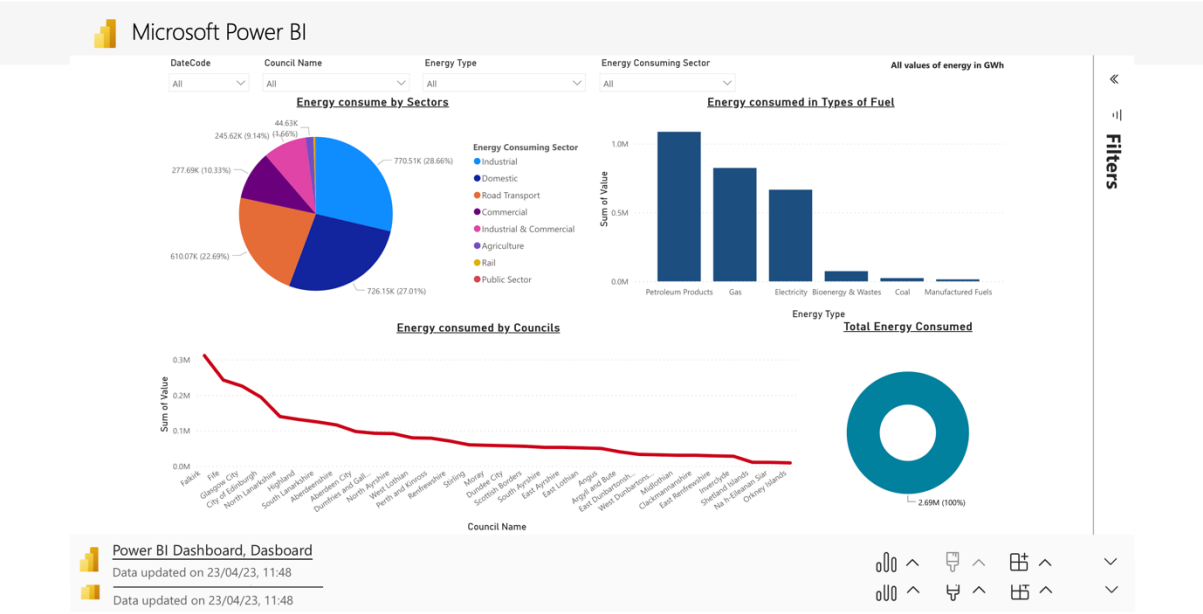

Figure 5.2 Power BI dashboard [22]

Utilizing Power BI, a dashboard was constructed to provide an in-depth visualization of energy consumption patterns in the Scottish council areas. The dashboard comprises four slicers: DateCode, Council Name, Energy Type, and Energy Consuming Sector. All energy consumption values are expressed in gigawatt-hours (GWh).

The dashboard is composed of various visual elements, including a pie chart illustrating energy consumption by sector, a bar graph depicting the types of fuel consumed, a line graph displaying the amount of energy consumed by each council, and a donut chart summarizing the total energy consumption.

The utilization of these diverse visualization methods allows for a more comprehensive understanding of the energy consumption patterns in the Scottish council areas. It enables stakeholders to explore the dataset and derive actionable insights, ultimately leading to more informed decision-making for optimizing energy consumption and promoting sustainable practices.

Pre-processing Data for Machine Learning:

Label encoding

```
# Create separate LabelEncoders for each categorical column
feature_name_encoder = LabelEncoder()
energy_type_encoder = LabelEncoder()
energy_consuming_sector_encoder = LabelEncoder()
```

```
# Fit the LabelEncoders
feature_name_encoder.fit(scot_data["FeatureName"])
energy_type_encoder.fit(scot_data["Energy Type"])
energy_consuming_sector_encoder.fit(scot_data["Energy Consuming Sector"])

# Transform the categorical columns
scot_data["FeatureName"] = feature_name_encoder.transform(scot_data["FeatureName"])
scot_data["Energy Type"] = energy_type_encoder.transform(scot_data["Energy Type"])
scot_data["Energy Consuming Sector"] = energy_consuming_sector_encoder.transform(scot_data["Energy Consuming Sector"])
```

Storing all the encoding values in CSV File

```
# Create mapping dictionaries for each categorical column
feature_name_mapping = dict(zip(range(len(feature_name_encoder.classes_)), feature_name_encoder.classes_))
energy_type_mapping = dict(zip(range(len(energy_type_encoder.classes_)), energy_type_encoder.classes_))
energy_consuming_sector_mapping = dict(zip(range(len(energy_consuming_sector_encoder.classes_)), energy_consuming_sector_encoder.classes_))
# Create a DataFrame for all mappings
all_mappings_df = pd.DataFrame()

# Add FeatureName mappings to the DataFrame
feature_name_df = pd.DataFrame(list(feature_name_mapping.items()), columns=['Mapping', 'FeatureName'])
all_mappings_df = pd.concat([all_mappings_df, feature_name_df], axis=1)

# Add Energy Type mappings to the DataFrame
energy_type_df = pd.DataFrame(list(energy_type_mapping.items()), columns=['Mapping', 'Energy Type'])
all_mappings_df = pd.concat([all_mappings_df, energy_type_df], axis=1)

# Add Energy Consuming Sector mappings to the DataFrame
energy_consuming_sector_df = pd.DataFrame(list(energy_consuming_sector_mapping.items()), columns=['Mapping', 'Energy Consuming Sector'])
all_mappings_df = pd.concat([all_mappings_df, energy_consuming_sector_df], axis=1)

# Save the DataFrame to a CSV file
all_mappings_df.to_csv('encoded_values.csv', index=False)
```

Splitting Data

```
# Split the dataset into training, validation, and testing sets
```

```
train_data, temp_data, train_labels, temp_labels = train_test_split(scot_data.drop(columns=["Value"]), scot_data["Value"], test_size=0.4, random_state=42)
valid_data, test_data, valid_labels, test_labels = train_test_split(temp_data, temp_labels, test_size=0.5, random_state=42)
```

Model Training

Importing all the required Library

```
from xgboost import XGBRegressor
from sklearn.preprocessing import LabelEncoder
from sklearn.neural_network import MLPRegressor
from sklearn.metrics import mean_squared_error,r2_score
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.svm import SVR
from sklearn.neural_network import MLPRegressor
from sklearn.model_selection import cross_val_score
from sklearn.metrics import r2_score, mean_absolute_error, mean_squared_error
```

Linear Regression

```
lr_model = LinearRegression()
lr_model.fit(train_data, train_labels)

# Perform 5-fold cross-validation
lr_scores = cross_val_score(lr_model, train_data, train_labels, cv=5)

# Make predictions on the validation set
lr_preds = lr_model.predict(valid_data)

# Calculate and print evaluation metrics
print("Linear Regression R2 Score:", r2_score(valid_labels, lr_preds))
print("Linear Regression MAE:", mean_absolute_error(valid_labels, lr_preds))
print("Linear Regression MSE:", mean_squared_error(valid_labels, lr_preds))
```

```
Linear Regression R2 Score: 0.028774160796721127
Linear Regression MAE: 252.66246256675174
Linear Regression MSE: 332562.66675817297
```

SVM

```
svm_model = SVR()
svm_model.fit(train_data, train_labels)

# Perform 5-fold cross-validation
svm_scores = cross_val_score(svm_model, train_data, train_labels, cv=5)
```

```
# Make predictions on the validation set
svm_preds = svm_model.predict(valid_data)

# Calculate and print evaluation metrics
print("SVM R2 Score:", r2_score(valid_labels, svm_preds))
print("SVM MAE:", mean_absolute_error(valid_labels, svm_preds))
print("SVM MSE:", mean_squared_error(valid_labels, svm_preds))
```

```
SVM R2 Score: -0.08900416859446691
SVM MAE: 184.99868960291084
SVM MSE: 372891.7784102961
```

Artificial Neural Network

```
ann_model = MLPRegressor(max_iter=1000, random_state=42)
ann_model.fit(train_data, train_labels)

# Perform 5-fold cross-validation
ann_scores = cross_val_score(ann_model, train_data, train_labels, cv=5)

# Make predictions on the validation set
ann_preds = ann_model.predict(valid_data)

# Calculate and print evaluation metrics
print("ANN R2 Score:", r2_score(valid_labels, ann_preds))
print("ANN MAE:", mean_absolute_error(valid_labels, ann_preds))
print("ANN MSE:", mean_squared_error(valid_labels, ann_preds))
```

```
ANN R2 Score: 0.030136365823675293
ANN MAE: 251.5059134946593
ANN MSE: 332096.2268034793
```

```
xgb_model = XGBRegressor(random_state=42)
xgb_model.fit(train_data, train_labels)

# Perform 5-fold cross-validation
xgb_scores = cross_val_score(xgb_model, train_data, train_labels, cv=5)

# Make predictions on the validation set
xgb_preds = xgb_model.predict(valid_data)

# Calculate and print evaluation metrics
print("XGBoost R2 Score:", r2_score(valid_labels, xgb_preds))
print("XGBoost MAE:", mean_absolute_error(valid_labels, xgb_preds))
print("XGBoost MSE:", mean_squared_error(valid_labels, xgb_preds))
```

```
XGBoost R2 Score: 0.9831260210328745
XGBoost MAE: 35.17810396100412
XGBoost MSE: 5777.909954220275
```

Finding Best Model, the best model

To find the best model among the four models, you can compare their evaluation metrics (R2 score, MAE, and MSE) and choose the one with the highest R2 score and the lowest MAE and MSE. Here's how you can do this:

```
# Create a dictionary to store the evaluation metrics of each model
model_performance = {
  'Linear Regression': {'R2': r2_score(valid_labels, lr_preds), 'MAE': mean_absolute_error(valid_labels, lr_preds), 'MSE': mean_squared_error(valid_labels, lr_preds)},
  'SVM': {'R2': r2_score(valid_labels, svm_preds), 'MAE': mean_absolute_error(valid_labels, svm_preds), 'MSE': mean_squared_error(valid_labels, svm_preds)},
  'ANN': {'R2': r2_score(valid_labels, ann_preds), 'MAE': mean_absolute_error(valid_labels, ann_preds), 'MSE': mean_squared_error(valid_labels, ann_preds)},
  'XGBoost': {'R2': r2_score(valid_labels, xgb_preds), 'MAE': mean_absolute_error(valid_labels, xgb_preds), 'MSE': mean_squared_error(valid_labels, xgb_preds)}
}

# Find the best model based on the highest R2 score, lowest MAE, and lowest MSE
best_model = None
best_r2 = -1
best_mae = float('inf')
best_mse = float('inf')

for model_name, metrics in model_performance.items():
  if metrics['R2'] > best_r2 and metrics['MAE'] < best_mae and metrics['MSE'] < best_mse:
    best_model = model_name
    best_r2 = metrics['R2']
    best_mae = metrics['MAE']
    best_mse = metrics['MSE']

print("Best Model:", best_model)
```

```
Best Model: XGBoost
```

XGboost turned out to be best model

Exporting model best Model

```
import pickle

# Save the model to a file
with open('xgb_model.pkl', 'wb') as model_file:
  pickle.dump(xgb_model, model_file)
```

Deploying

Create a new file named app.py:

```
from flask import Flask, request, jsonify
import pickle
import numpy as np

app = Flask(__name__)

# Load the trained XGBoost model
with open('xgb_model.pkl', 'rb') as file:
    xgb_model = pickle.load(file)

@app.route('/predict', methods=['POST'])
def predict():
    # Get the input data as a JSON object
    data = request.get_json(force=True)

    # Convert the JSON object to a NumPy array
    input_data = np.array(data['input'])

    # Make a prediction using the model
    prediction = xgb_model.predict(input_data.reshape(1, -1))

    # Return the prediction as a JSON object
    return jsonify(prediction.tolist())

if __name__ == '__main__':
    app.run(debug=True)
```

Run the Flask app:

By typing below command in terminal

python app.py

To use Postman to test the model:

Install Postman from https://www.postman.com/downloads/.

Open Postman and create a new POST request.

Set the URL to http://127.0.0.1:5000/predict.

In the "Body" tab, select "raw" and set the format to "JSON".

Enter a sample input as a JSON object in the following format:

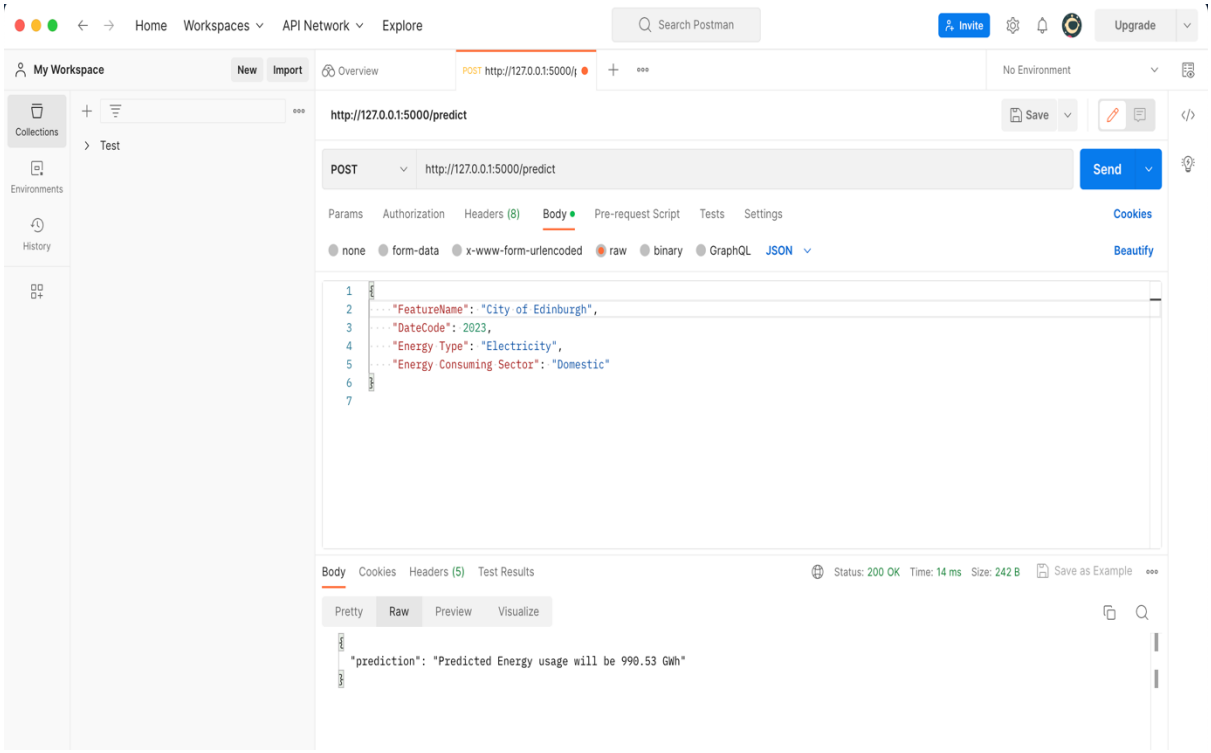

Figure 5.3: Testing Deployed Model

A sample input for testing the deployed model is as follows:

- FeatureName: "City of Edinburgh"
- DateCode: 2023
- Energy Type: "Electricity"
- Energy Consumption Sector: "Domestic"

The model deployed in production returns the following output:

"Predicted Energy usage will be 990.53 GWh"

This outcome demonstrates that our implementation successfully addresses all the objectives mentioned in the project scope.

6 Evaluation of work

The project aimed to collect, analyze, and visualize energy consumption data for Scottish council areas from 2005 to 2020 and develop predictive models to optimize energy consumption. The evaluation of the work done can be discussed in terms of the objectives achieved, the methods employed, and the insights derived from the project.

Firstly, the data collection process successfully gathered comprehensive energy consumption data for the specified years, enabling thorough analysis and exploration of trends. The exploratory data analysis (EDA) conducted using Python and Jupyter Notebook uncovered patterns, correlations, and anomalies within the dataset, providing valuable insights to inform decision-making.

The Power BI dashboard developed during the project allowed for an interactive and visually appealing representation of energy consumption trends, enabling stakeholders to navigate and understand the data with ease. The utilization of various visualization techniques offered an in-depth understanding of the energy consumption patterns, leading to better-informed decisions on energy efficiency and sustainability initiatives.

The development and evaluation of four machine learning models (Linear Regression, Support Vector Machines, Artificial Neural Networks, and XGBoost) facilitated the selection of the best-performing model, which was subsequently deployed using Flask and tested using Postman. The model's performance during deployment and testing demonstrated the project's success in achieving its objectives.

7 Conclusions and Future of work

In conclusion, the project effectively addressed its primary aim of collecting and analyzing energy consumption data for Scottish council areas, developing an interactive dashboard to visualize the data, and deploying a predictive model for optimized energy consumption. Through this work, we have demonstrated the potential for data-driven decision-making in the realm of energy management and sustainability.

Looking forward, there are several opportunities to expand and improve upon the project. Additional data sources could be incorporated to provide a more comprehensive understanding of energy consumption patterns and the factors influencing them. Furthermore, the inclusion of other advanced machine learning techniques, such as deep learning, could potentially enhance the predictive model's accuracy.

Another direction for future work could involve exploring energy consumption patterns at a more granular level, examining individual households or buildings, and identifying potential inefficiencies or areas for improvement. This could aid in the development of targeted interventions and policies aimed at reducing energy consumption and promoting sustainable practices.

Lastly, the deployment of the predictive model as a web service could be enhanced to ensure scalability and improved user experience. Integrating the model with other web applications or services could extend its utility, allowing for broader access and impact on energy management and sustainability initiatives.

Referencing

[1] Allaire, J., & Chollet, F., 2018. *Deep Learning with R*. Shelter Island, NY: Manning Publications Co.

[2] Anaconda: Continuum Analytics, 2021. Anaconda. Version 2021.05. Available at: https://www.anaconda.com/products/distribution (Accessed: [Access Date]).

[3] Biswal, A., 2023. *Power BI Vs Tableau: Difference and Comparison*. [Online]. Simplilearn. Available from: https://www.simplilearn.com/tutorials/power-bi-tutorial/power-bi-vs-tableau [Accessed Date].

[4] Cortes, C., and Vapnik, V., 1995. Support-vector networks. *Machine Learning*, 20(3), pp. 273-297.

[5] Goodfellow, I., Bengio, Y., and Courville, A., 2016. *Deep Learning*. Cambridge Massachusetts: MIT Press.

[6] Grinberg, M., 2018. *Flask Web Development: Developing Web Applications with Python*. Sebastopol, CA: O'Reilly Media, Inc.

[7] Hastie, T., Tibshirani, R., and Friedman, J., 2009. *The Elements of Statistical Learning*. 2nd ed. New York: Springer.

[8] Hyndman, R. J., and Koehler, A. B., 2006. *Another look at measures of forecast accuracy*. International Journal of Forecasting, 22, pp. 679-688.

[9] Ihaka, R. and Gentleman, R., 1996. R: A Language for Data Analysis and Graphics, *Journal of Computational and Graphical Statistics*, 5:3, pp. 299-314.

[10] James, G., Witten, D., Hastie, T., and Tibshirani, R., 2013. *An Introduction to Statistical Learning: with Applications in R*. New York: Springer.

[11] McKinney, W., 2017. *Python for Data Analysis Data Wrangling with Pandas, NumPy, and IPython*. 2nd ed. Sebastopol, CA: O'Reilly Media, Inc.

[12] Postman App: Postman, Inc., 2021. Postman. Version 9.0.7. Available at: https://www.postman.com/downloads/ (Accessed: [Access Date]).

[13] Power BI: Microsoft, 2021. Power BI. Version: 2.103.7010.661. Available at: https://powerbi.microsoft.com/ (Accessed: [Access Date]).

[14] Python: Python Software Foundation, 2021. Python. Version 3.9.7. Available at: https://www.python.org/downloads/release/python-397/ (Accessed: [Access Date]).

[15] Scottish Government, 2014. *Energy Consumption*. [Online]. Scottish Government. Available from: https://www.gov.scot/collections/energy-statistics/ [Accessed Date].

[16] Scottish Government, 2020. *Energy consumption*. [Dataset]. Scotland: Scottish Government. Available from: https://statistics.gov.scot/resource?uri=http%3A%2F%2Fstatistics.gov.scot%2Fdata%2Fenergy-consumption [16 Feb 2023].

[17] Scottish Government, n.d. *Energy statistics*. [Online]. Energy and Climate Change Directorate. Available from: https://www.gov.scot/collections/energy-statistics/.

[18] Shin, T., 2020. *An Extensive Step by Step Guide to Exploratory Data Analysis*. [Online]. Towards Data Science. Available from: https://towardsdatascience.com/an-extensive-guide-to-exploratory-data-analysis-ddd99a03199e [Accessed Date].

[19] UK Department for Business, Energy & Industrial Strategy (BEIS), 2013. *Digest of UK Energy Statistics (DUKES)*. [Online]. Department for Energy Security and Net Zero and Department for Business, Energy & Industrial Strategy. Available from: https://www.gov.uk/government/collections/digest-of-uk-energy-statistics-dukes [Accessed Date].

[20] UK Department of Energy & Climate Change, 2012. *Sub-national total final energy consumption statistics: factsheet*. [Online]. Department of Energy & Climate Change. Available from: https://www.gov.uk/government/statistics/sub-national-total-final-energy-consumption-statistics-2010-factsheet [Accessed Date].

[21] Willmott, C. J., and Matsuura, K., 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, 30, pp. 79-82.

[22] Chaudhary, K., 2023. *Energy Consumption Analysis and Optimisation for Scottish Councils*. Unpublished internal document. RGU.