An Artificial Intelligence Approach to Predict Household Energy Efficiency

Oisín Brannock (20235671)
School of Computer Science
National University of Ireland Galway

Supervisors
Dr. Karl Mason

In partial fulfillment of the requirements for the degree of
MSc in Computer Science (Artificial Intelligence)
August 18, 2022

DECLARATION

I, Oisín Brannock, do hereby declare that this thesis entitled "An Artificial Intelligence Approach to Predict Household Energy Efficiency" is a bona fide record of research work done by me for the award of MSc in Computer Science (Artificial Intelligence) from National University of Ireland, Galway. It has not been previously submitted, in part or whole, to any university or institution for any degree, diploma, or other qualification.

Signature: ____

Abstract

Many studies have reported successful machine learning models using sensor data. However, generation of a successful predictive model using accessible home owner data, like yearly energy consumption and location, remains to be accomplished. Here, I report that a combination of 13 features can be inserted into a pipeline that can: impute missing values in features, encode categorical features, and under/over sample the data before prediction. The result post optimisation is a new data pipeline, simply named Building Energy Rating Predictor (BERP), that encapsulates the principles of CRISP-DM and MLOPs in the delivery of a user-friendly predictive model, that is able to successfully predict building energy ratings to an F1 score of 0.81. Predictions of this nature can be used to feed information into an application interface, that home owners can make use of to find energy efficiency ratings, without on site inspection. If energy spend costs are becoming problematic for a household, this can provide context into why and help guide mitigation of cost. Likewise, it can support government level initiatives to identify the areas of Republic of Ireland (ROI) that are at most in need of home improvements, in order to reach the 2030 B2 household target set. BERP has the ability to meet both of these criteria, with excellent accuracy.

Keywords: Energy Efficiency, BER, F1, Accessibility, Cost Savings

Contents

List of Figures

List of Tables

List of Acronyms

AI Artificial Intelligence. 5, 34, 35, 65

ANN Artificial Neural Network. 23, 32, 48, 49, 54, 63

AUC Area Under the Curve. 49, 55, 58, 79, 80

AUROC Area Under the Receiver Operating Characteristics. v, 58

BER Building Energy Rating. ii, 1–3, 17, 28, 33–36, 46, 54, 55, 58, 59, 61–63, 66–68, 79, 80

BERP Building Energy Rating Predictor. ii, vi, 30, 33, 47, 57, 58, 61, 63–66, 78

CART Classification and Regression Tree. 19

CNN Convolutional Neural Network. 32, 33

CPU Central Processing Unit. 65

CRISP-DM Cross-Industry Standard Process for Data-Mining. ii, iii, 7, 8

DNN Deep Neural Network. 63

ELM Extreme Learning Networks. 32

FLNN Functional Link Neural Networks. 32

GA Genetic Algorithm. 32

GDPR General Data Protection Regulation. 36

IEEE Institute of Electrical and Electronics Engineers. 26, 27

kNN K-Nearest Neighbours. iii, 2, 9, 17, 18, 48, 54, 63

LSSVM Least Squares Support Vector Machine. 32

ML Machine Learning. 7

MLOPs Machine Learning Operations. ii

MSE Mean Squared Error. 23

PCA Principal Component Analysis. 19, 29

PLS Partial Least Squares. 29

PR Precision Recall. 60

ReLU Rectified Linear Unit. 23, 24

RGS Regression Gradient Guided Feature Selection. 32

RMSE Root Mean Squared Error. 39

ROC Receiver Operating Curve. 58

ROI Republic of Ireland. ii, 34, 36, 66

SEAI Sustainable Energy Authority Of Ireland. 36, 77

sklearn Scikit-Learn. 49

SMOTE Synthetic Minority Oversampling Technique. 14, 16

STL SMOTETomekLinks. 16, 45, 55, 63, 65

Chapter 1

Introduction

1.1 Motivation

Hypothesis: The BER of households can be predicted using general household characteristics to an accuracy of 80%.

Energy efficiency is one of the most prevalent topics in our society today, from conferences on global warming to everyday energy savings. The move towards environmentally friendly practices has seen a dramatic rise in popularity, in business sectors, as well as within the general public. Companies like Disney use a strict net 0 carbon emissions policy in all projects they undertake, or car companies like ford using geothermal cooling systems in factories, as well as developing electric vehicles that are practical for use compared to fuel based engines. Biodegradable products have seen a rise in popularity to reduce plastic usage, such as bamboo toothbrushes or recycled toilet paper. Journals published as early as the 1980s have broached different approaches in search of the ideal predictive model [3]. The scope of this topic governs every industry in the world, and is imperative to a sustainable future for humankind. Initially, studies in this field focused on efforts to improve the efficiency of buildings during their construction period [4]. This has evolved in the age of information through the use of computational models to predict what can impact energy usage in a building, whether it be a school, office or water treatment plant [5; 6; 7; 8].

The goal of this thesis is to predict energy usage for households, using generally accessible household data. Furthermore, to highlight the uses for this predictive model, not just for the home owners, but also general sustainability bodies in government, in the focus on driving to a net zero target by 2050 [9]. The ideal visual representation would first predict BER and show general statistics on what led to this prediction. Home owners would gain an awareness of how much they could save on energy costs and what the most effective route would be to lead to these savings, due to the correlation of BER to energy spend. This analysis can help home owners save money, and most importantly help raise awareness and inspire action towards a sustainable environment. A raise in BER also improves home valuations which is desirable.

In addition, this thesis seeks to determine the optimal method of prediction, and apply this to energy efficiency of Irish households. Methods such as neural networks, decision trees and kNNs have been shown to be effective in this space using sensor data to predict energy usage, but how will they fare using generally accessible household data, like type of structure, insulation type, dwelling type and so on? Sensor data is far more reliable, but not practical for home owners to simply gather on their own. Every feature used in this predictive model is something people can have access to. This predictive model also seeks to add a layer with the BER being used to talk about potential savings as opposed to prediction of energy usage, which can be accessed through energy providers. Therefore this method is more applicable to home owners to predict BER and show ways in which it can be improved in parallel to energy usage.

1.2 Thesis Structure

Chapter 1 outlines the motivation for this paper.

Chapter 2 focuses on the methods used to perform research and analysis of the data.

Chapter 3 is an in depth literature review of the work done in the space of energy and sustainability to get an idea of the work done to date in this field.

Chapter 4 delves into the data used for analysis; where it comes from, how it is processed for use and what considerations had to be taken into account in its usage. The analysis portion deals with the prediction of the BER; what algorithms for prediction were chosen and why, measuring evaluation metrics and fine-tuning to enhance the chosen metrics through a variety of techniques.

Chapter 5 highlights the experimental method from start to finish; steps on how analysis was done, and settings of cross validation.

Chapter 6 examines the prediction results, and goes into depth on post analysis to outline the results and what they mean.

Chapter 7 provides an in depth discussion of the results presented in the Chapter 6; what significance they have in real world use cases? Are they as expected, or is there some flaw highlighted?

Chapter 8 presents a summary of the thesis, outlines the contributions of the research, provides answers to the research questions posed in Chapter 1, and finally discusses the implications and impact of this work.

All diagrams used in this thesis have been created by the author and are only referenced when required.

1.3 Research Questions

This research will answer the following research questions (RQs):

RQ_1 - What are the data requirements for predicting energy usage for households?

RQ_2 - How should machine learning be applied to best predict household energy efficiency?

RQ_3 - How can visualisation aid in delivering insight about the homes from this predictive model?

Chapter 2

Background

This chapter gives context to the methods used in this thesis for analysis. It also ties into how each one is relevant to the idea of energy saving.

2.1 Machine Learning

Machine learning is a branch of AI that allows machines to solve a vast range of problems faster, and more often than not, more accurately than a human can. For example, algorithms have been created to better identify tumours in patients that doctors can. Machine learning is concerned with designing programs that can learn rules and patterns from data, and adapt to new scenarios based on this training [10]. This allows machines to infer answers rather than having to be explicitly programmed. Many of the challenges we wish to overcome in today's world are not straightforward and cannot be simply programmed for a computer to solve in a binary manner. There are a plethora of techniques machines can use to solve problems, which are constantly being improved upon through research and analysis. Machine learning consists of 3 sub-categories: supervised learning, unsupervised learning and reinforcement learning. This thesis focuses on the use of supervised learning methods. In Fig. 2.1, the general flow of a machine learning process is outlined. It starts by first gathering, cleaning and splitting a dataset into training and testing datasets, and normalising/transforming data to suits the means of the task at hand. The predictive model is then fitted on the training data. The optimal model parameters of the algorithm are identified using the training data. The predictive model is then evaluated using statistical tests to check if the results are statistically significant, and if they make sense in reference to the hypothesis. The predictive model can then be deployed for use and, improved using new incoming data. Section 2.2 expands on this for a more cohesive structure.

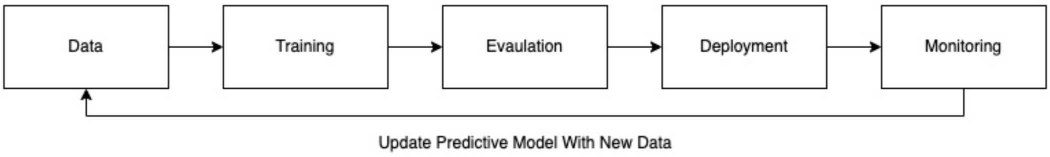


Figure 2.1: Simple Machine Learning Workflow

The availability of data today has led to a huge uptake of machine learning by new practitioners, expanding beyond the realms of scientific research into everyday life. This has included the creation of predictive models that cover the topics such as fraud detection in the financial industry, allows cars to be autonomous on the road, or facial recognition that is used in phone and security technology [11; 12; 13]. The use cases are abundant, and constantly being expanded upon, as noted above.

2.2 CRISP-DM

The Cross-Industry Standard Process for Data-Mining (CRISP-DM) is a methodology that seeks to standardise how predictive models are developed and maintained across industries. This used is to standardise an approach to ML to improve results from start to finish in a project [14; 15]. CRISP-DM is important in relation to the analysis of this thesis as it serves as a foundational layer for model design. This thesis has been structured in a way that follows this format, ranging from the "business understanding" aspect in the first couple of chapters, all the way to prediction and deployment in latter chapters.

Business understanding involves developing a hypothesis for the problem at hand. Is this really a problem that could benefit from machine learning? Is the problem well defined? Break down the project into phases, like chapters of a thesis?

Data understanding deals with the assessment of data collected. What kind of data do we have? Is it structured in the form of comma separated values? Does it have one row per observation? Is it in wide or long format [16] etc?

Data preparation involves making sure the data is in an appropriate format for the algorithm chosen. Poor data impacts the quality of predictive models. Therefore it is vital to prepare the data in a correct manner. Does the data need to be normalised? Do we need to fill in null values? How do we fill in these nulls if they exist; with 0, with the mode or with the mean? Could we create a model to predict what these nulls may be? Could we engineer new features based on the ones we already have at our disposal?

Making predictions through machine learning involves choosing and creating predictive models, and determining the one that best suits the data. The candidate predictive model at first may not be the optimal model after cross validation. If performance is lower than desired in a chosen evaluation metric such as accuracy or F1 score, one may need to go back a step or two to determine if the data has been understood correctly or has been prepared for analysis correctly.

Evaluation takes the optimal predictive model design, evaluated through a chosen metric, and determines if it meets the needs of the project to an acceptable threshold. How are the results gathered statistically assessed? Do they meet the minimum thresholds set by the chosen tests? The key question in this phase is: **Have we solved the problem we outlined in the initial phase of this project?**. Evaluation is all about creating quality.

Deployment involves putting a predictive model into a state where it can be accessed by people outside of the project to solve the problem at hand. How does one plan on achieving this deployment? What resources will be used? Who will maintain the predictive model for incoming data? When questions like these have been addressed, ongoing prediction monitoring can be established and constant iterations of improvement can be made.

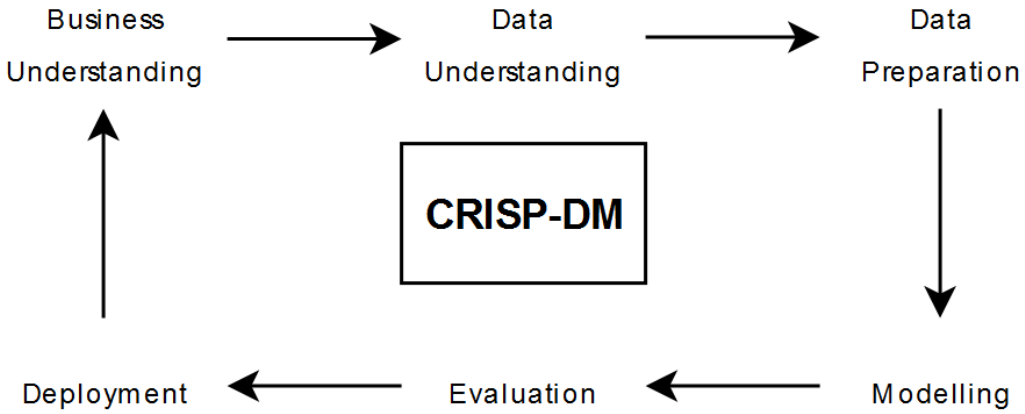


Figure 2.2: CRISP-DM Methodology

2.3 Missing Data - Imputation

Missing data is present in many real world datasets. There is very little that can be done to avoid it in most situations. It arises from cases where manual entry is left blank, entered incorrectly, or is poorly ingested by a data engineering pipeline. Whatever the reason, it is imperative to find a way to deal with this kind of data gap issue.

One such method of dealing with data gaps is imputation. Imputation is the process of handling missing values in data by using statistical techniques to approximate a representative value for the missing data. This can be as simple as filling the missing values in a dataset with the mean, median or mode of that column, depending on the datatype. Lin et al performed a review of all imputation techniques across literature between 2006 and 2017 [17]. They concluded that random forest imputation and kNN imputation were the superior techniques to use in extreme missing data cases ($< 30\%$ for kNN and $> 50\%$ for missing forest).

2.3.1 Missing Forest Imputation

The missing forest algorithm takes data in its raw form, without any preprocessing of categorical variables into numerical variables. It starts off by imputing each column with its respective mean/mode. It then uses all the other features in the dataset to build a random forest model on the present datapoints, to predict the values of the missing datapoints. It iterates on the dataset, each time starting with a better base of data to work with until the difference in imputes calculated are negligible. Stekhoven and Buhlmann [18] created the algorithm, missing forest, in 2011. Their goal was to overcome the limitations of kNN imputation, which is very sensitive to the curse of dimensionality. The curse of dimensionality involves the exponential increase in the dimensions of data and the growth of necessary computing power that results from this. As dimensions increase, points appear to be equidistant and as a result kNN finds it difficult to classify them accordingly.

2.4 Data Encoding

Machine learning algorithms generally do not handle categorical variables very well. Categorical data consists of variables that can be assigned to groups e.g. age, rating, sex etc. Categorical variables generally can either be ordinal, in that they have a clear hierarchy, or simply just labels with no ordinality. In all cases, we must discern the value of each label in order for it to be useful in a machine learning setting. There are a variety of ways this can be done depending on the type of categorical data one has.

Label encoding is one such technique. One simply maps each unique category to a numerical value. If we had 3 countries; Ireland, USA and Japan, this would convert to 1,2,3 for example. The downfall of label encoding is that we now have a situation where a predictive model could interpret an order to the labels. It may capture that Ireland > USA for example, which is not true in this setting.

One hot encoding, or dummy encoding, is a technique that looks to overcome the limitations of label encoding by creating dummy variables in the data. It creates additional dummy features, one for each category in the variable. It then labels with a 1 if a row of data is in that category and 0 elsewhere.

Dummy encoding is a popular technique but it has limitations. For one, the new dummy variables are extremely correlated with one another, which can lead to collinearity issues. The influence of these features over one another may interfere with fitting a predictive model to data. This can be overcome through analysis of the data, and dropping certain dummy columns. The other major

Table 2.1: Original Data

BER
A1
A2
A3
B1

⇒

Table 2.2: One Hot Encoded Data

A1	**A2**	**A3**	**B1**
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

downside of one hot encoding is that it can dramatically inflate a dataset if the number of categories to encode is large. For example, if we wanted to encode years ranging from 1900 to 2000, this is an addition of 100 columns. In big datasets, this can cause memory issues very quickly, as well as decrease model performance and increase training time. We do not have infinite resources so must optimise how we use what we have. Bigger datasets make it harder for the predictive model to fit the data, leading to poorer accuracy, especially when the inflation is due to dummy encoding as it adds no information, but simply transforms.

The last method in scope for this thesis is known as category boosting encoding, or CatBoost Encoding. Prokhorenkova et al developed the algorithm in 2018 in order to combat increasing problems in research related to prediction shift in production grade predictive models over time [19]. This occurs when data changes over time due to changing trends, changing how the model predicts values. Production grade refers to use in commercial settings. It works by first randomly permuting the dataset. This is done to ensure that the dataset is not ordered based on the target variable (what we want to predict). Next, the categorical variables are converted to labels, like with label encoding. The following formula is then used to convert the numerical labels to a floating point numerical interpretation:

$$target = \frac{countInClass + prior}{totalCount + 1} \tag{2.1}$$

Where $countInClass$ is how many times the label value was equal to 1 for objects with the current categorical feature value. $prior$ is the starting value for the label during label encoding.

$totalCount$ is the total number of datapoints (up to the current one) that have a categorical feature value that matches the current one [20].

$Feature$	BER	$countInClass$
Apartment	A1	$\frac{0+\frac{1}{1000000}}{0+1} = 1 \times 10^{-6}$
Apartment	A2	$\frac{0+\frac{1}{1000000}}{1+1} = 5 \times 10^{-7}$
Detached House	A2	$\frac{0+\frac{1}{1000000}}{0+1} = 1 \times 10^{-6}$
Detached House	A2	$\frac{1+\frac{1}{1000000}}{1+1} = 0.5$
Apartment	A1	$\frac{1+\frac{1}{1000000}}{1+1} = 0.5$

Table 2.3: CatBoost Example

This method has a big advantage in that it eliminates both the limitations of label encoding, by going beyond ordering and one hot encoding, keeping the same number of columns in the dataset as there were at the outset. In Table 2.3, we use previous examples to update the new encoding. For every apartment value with A1, this will update based on previous examples. In the end this will converge to a value and be used for all A1 apartments in the data. This eliminates the sparse 0 matrices of one hot encoding.

2.5 Sampling

Imbalanced datasets are abundant, and a big issue in data science. They occur when we have one class that dominates, and another that perhaps only covers 1% of what the majority class does. This leads to deceptive prediction performance, as a predictive model would discriminate on the minority class, and generally choose the majority class. Therefore, we need a technique to combat this problem.

The predictive model may have 95 datapoints for label Non-Cancerous and 5 for label Cancerous. If the model always predicted Non-Cancerous it would have a 95% accuracy, even though it has no ability to distinguish between classes, and lead to misdiagnosis. This is where F1 score becomes our primary metric of evaluation.

Oversampling is a technique that seeks to duplicate samples in the minority class, so that they are inflated to match the size of the majority class, negating the imbalance. It does not add any new information to the data, copying previous datapoints.

Undersampling on the other hand seeks to do the opposite, by removing samples from the majority class at random, until it matches the minority class in volume. The disadvantage of this technique is that we are losing valuable information in our data.

Synthetic Minority Oversampling Technique (SMOTE) was introduced in 2002 by Nitesh Chawla, et al [21]. This algorithm seeks to oversample the minority classes by creating new synthetic datapoints. Instead of simply creating exact duplicates of the minority class, SMOTE seeks to create new samples from the minority class. It does this by selecting samples close together, finding the average/modal values between the samples, and creating a brand new synthetic sample. It will repeat this process until the minority class(es) are balanced with the majority class in volume. The new samples are very similar to the original ones, but the slight difference allows more information to be gained from our dataset for predictive modelling. They are better than repeated oversampling of the same copies, by providing more feature variance.

Tomek developed a technique to deal with undersampling in his 1976 paper on Condensed Nearest Neighbors [22]. This method of undersampling tries to acquire the datapoints from the dataset that limit the loss in information that a model can use to predict. It does this by taking every sample from the minority class in the data, and selecting only points in the majority class that cannot be classified correctly by the predictive model, due to them being very similar to another class.

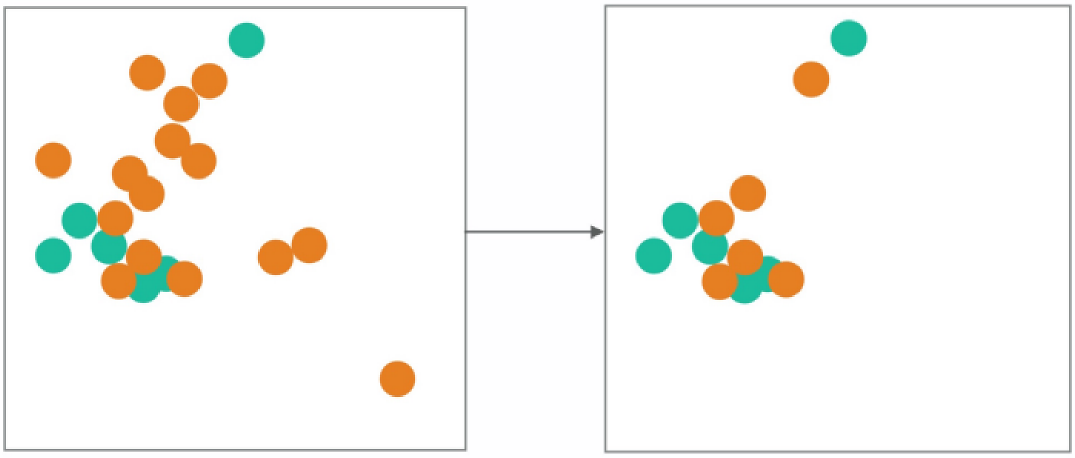


Figure 2.3: CNN [1]

However, this technique suffers in practice as it selects data at random, which leads to unwanted values being left in the training data, as shown in Figure 2.3. The adaption used to overcome this limitation is called TomekLinks. It states that two points form a Tomek link if they are from separate classes, and each of them is the nearest neighbor of one another. This tells us that both of these points are near a decision boundary that determines class separation in the data. It also tells us one of these data points could be an outlier. Therefore, we want to remove all of these identified TomekLink pairs, in order to make the lines that separate classes into two more distinct and clear. Drawing this line in Figure 2.3 would lead to poor accuracy as the points are so close together.

Batista et al combined the methods of SMOTE and TomekLinks together to form SMOTETomekLinks (STL) [23]. The technique simply performs SMOTE as before, to oversample the data with synthetic datapoints, and is then undersampled by finding the TomekLinks in the data to make a clear decision boundary between classes. This technique is utilised in this report.

2.6 Supervised Learning

Supervised learning is a sub-category of machine learning. It involves the use of labelled data to train a computational predictive model. In training the predictive model, the machine is shown the correct outcome for each training example. The basis for this technique is that we can predict what will happen in similar future scenarios, given what has already happened in the past [24]. This assumes that the factors that initially led to these outcomes have not changed, which may not always be the case, depending on the problem. Supervised learning can be split into classification and regression tasks. Classification deals with categorical, discrete and Boolean predictions, where only a finite number of values are possible e.g. Yes/No, 1/0, Red/Green/Blue. Regression deals with the estimation of continuous values, numbers with infinite values between two points e.g. $11.43m^2$ for room area. The focus in this thesis will be on classification, as the goal is to predict BER, which is a multi-class (15-class) ordinal variable.

2.6.1 Classification

2.6.1.1 K-Nearest Neighbours (kNN)

kNN works under a simple premise; any datapoints that have similar characteristics to other datapoints in a dataset will have similar outcomes [25]. This is known as feature similarity. In kNN classification, the algorithm starts by calculating the distance of the new datapoint to each training datapoint. This is done either by calculating the Euclidian, Manhattan or Minkowski distance.

The Euclidian distance is calculated by taking the square root of the sum of the difference of squares between the new datapoint and training datapoint [Eq. 2.2].

The Manhattan distance is calculated by taking the shortest distance between the two vectors that represent the new data and training data, and getting the sum of their absolute difference [Eq. 2.3].

The Minkowski distance is a generalisation of the Euclidian and Manhattan distance formulae. The formula relies on the constant p. If $p = 1$, then we have the Manhattan formula. If $p = 2$, we have the Euclidian formula [Eq. 2.4].

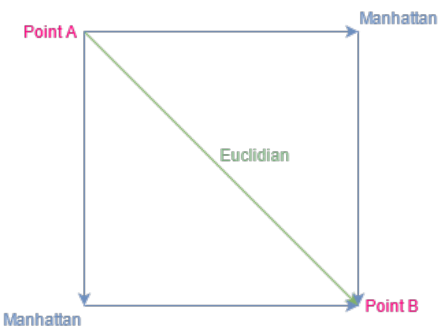


Figure 2.4: Distance Formulae

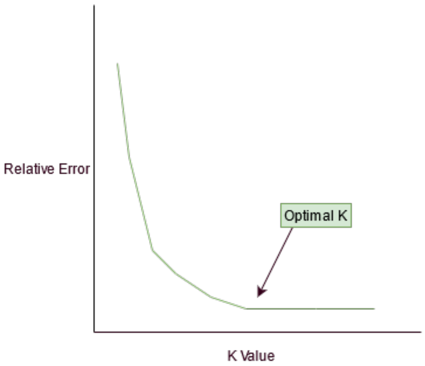


Figure 2.5: Example Elbow Plot

If a new datapoint is close to two or more training datapoints, the mode of the training datapoints close by are taken to be the value of the new datapoint. This is subjective relative to the value chosen for k. The optimal k value can be found by predicting for a range of values of k, and plotting the relative error of calculations against the k values. This is known as an elbow plot (Fig 2.5). A rule of thumb used is $k = \sqrt{N}$, where N is the number of training datapoints. So if we had 400 training datapoints, $k = \sqrt{400} = 20$.

kNN is a useful prediction approach as it is simple, intuitive, and does not assume anything about the data it analyses. However, it does suffer from the curse of dimensionality, wherein the more independent variables that are used to predict a dependent variable, the number of dimensions increases [26]. This causes confusion in N-dimensional space for the predictive model in its perception of distance between points. This can be mitigated using PCA, which is a dimensionality reduction technique that can bring the number of dimensions back to 2.

$$D_{Euclidian} = \sqrt{\sum_{j=1}^{k}(x_j - y_j)^2} \tag{2.2}$$

$$D_{Manhattan} = \sum_{j=1}^{k} |x_j - y_j| \tag{2.3}$$

$$D_{Minkowski} = \left[\sum_{j=1}^{k}(|x_j - y_j|)^p\right]^{\frac{1}{p}} \tag{2.4}$$

2.6.1.2 Decision Trees

Decision trees are useful predictive models for both classification and regression analysis. Classification trees are developed generally using the CART methodology, which takes the independent variables and uses them to split nodes up, with the goal of predicting a dependent variable. A decision node is one that splits into another decision node and leaf nodes, or just leaf nodes. Fig. 2.6 shows a very basic example of a classification tree workflow to illustrate this. We have two classes here: Malignant or not malignant. Each node in the decision tree is decided using a feature from the data gathered. In this case, 3 features are used, and split until we reach our final classes. This tree has also been pruned to have a maximum depth of 4 so as to not overfit the data, so it will be generally good on new unseen data.

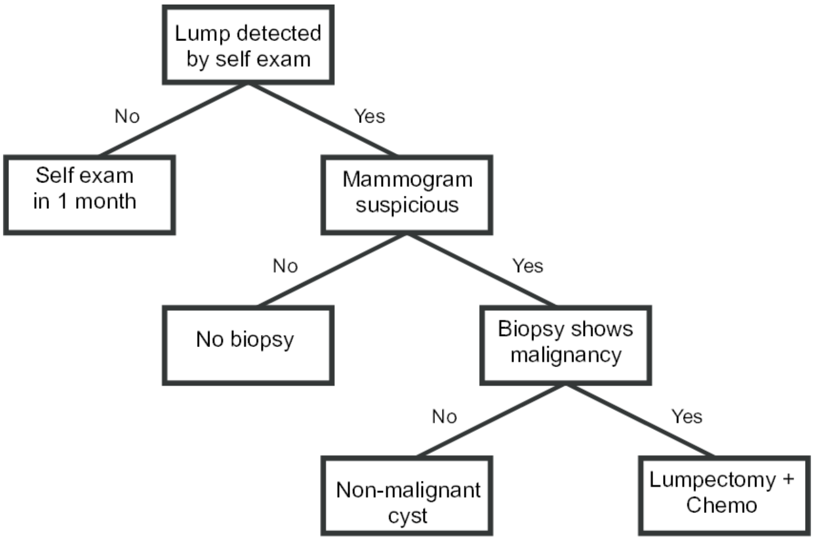


Figure 2.6: Classification Tree Flow Example (Taken from [2])

The splits of decision nodes use loss functions that will determine the best split to make, based on a metric called purity. This can be determined in classification using one of two methods.

One such method is called Gini Impurity [27]. This measures the likelihood that the tree would misclassify a test example. It can also be described as a quantification of the variance across our classes in the dataset.

$$G = \sum_{i=1}^{j}(p_i(1 - p_i)) \tag{2.5}$$

Where p_i is the probability of picking a point from class i.

Another method is known as entropy calculation. Entropy is a measure of the proportion of class spread we have within a given node in our tree.

$$E = -\sum_{i=1}^{N}(p_i log_2 p_i) \tag{2.6}$$

Where p_i is the same as the Gini calculation. Entropy is calculated for splits, and a split is only performed if the entropy of the child node is lower than that of the parent node. We can think of this in terms of information gain.

$$IG = E_p - E_c \tag{2.7}$$

The more entropy removed from the parent node in splitting into the child node/s, the more information we can say this feature gives us about the target class.

We want a child node that generates the least variance when splitting a parent node. This calculation is done for the feature variables each time the parent nodes need to be split, and the child node with maximum information gain is chosen to proceed further along the tree for additional splitting.. The tree will look at each feature variable, and check to see which results in the lowest variance when split and choose this variable to split on, and so forth until the tree ends. Classification trees need to be pre-pruned generally, which means we set a max depth for the tree so it doesn't overfit on the training data. Another solution would be to increase the number of trees used and use the majority voting of these trees to make the best predictive model. This is known as a random forest [28].

2.6.1.3 Random Forests

Random forests make use of an ensemble method called bagging to use many decision tree learners in order to enhance prediction performance. This involves majority voting to come to the best outcome. Random forests allow an individual tree in the forest to grow very large without needing to be pruned, as we are not as concerned anymore about the high variance of a single tree. A method known as bootstrapping [29] is used to pick random samples from the dataset to train each tree with. This negates overfitting. While a single tree may overfit, as a collective, the forest will not be biased to any specific training data.

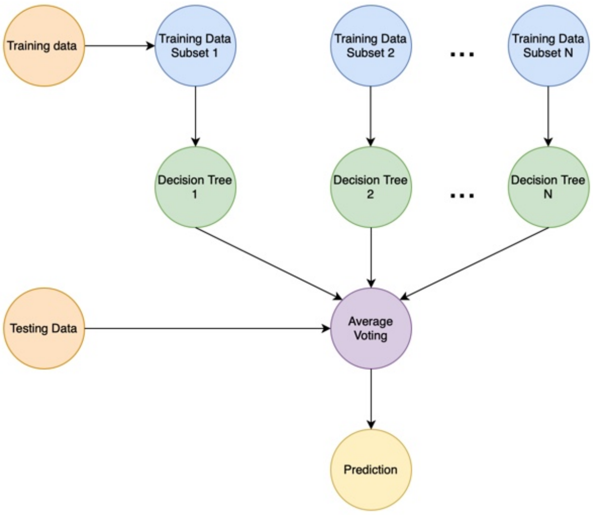


Figure 2.7: Random Forest Flow

When constructing each tree, the variables for each split are chosen randomly, as opposed to using reduction of variance. This again is to ensure that each tree is unique and they do not correlate with one another to lead to a large bias. Hence, we have low variance.

Finally, we repeat this process for n trees. The average prediction of the trees is taken as the prediction for a specific test datapoint.

2.7 Neural Networks

Neural Networks are algorithms that are modelled on the structure of the human brain, and try to replicate the process of information retrieval and deduction. They consist of nodes called neurons, which are points that data flows through and is processed. In it's most basic form, a neural network takes in data, passes it through to subsequent layers of neurons via weighted connections that perform mathematical operation on the data, and finally returns this new processed data as a decision on the instances of data it was provided wit [30]. Neural networks have become very popular in recent years, due to the every increasing availability of large quantities of data, and increased computational power. Outlined below is a form of neural network that has been selected as a possible framework for this thesis.

2.7.1 Artificial Neural Networks

ANNs are not a new concept in machine learning. A multitude of machine learning methods can be represented as neural networks. For example, simple linear regression can be viewed as two input neurons that are multiplied by a weight and bias (slope and y-intercept here) and added together to reach a single output layer as shown in Fig 2.8.

In more complex machine learning problems, we need more layers that perform more operations on the input data, before being output as a prediction, so more complex algorithms can be computed. The hidden layers would make use of a linear or Rectified Linear Unit (ReLU) activation function [31] by convention, but any activation function can be used. Finally when the neural network outputs a generation from the output layer, this can be used to make a prediction. Metrics such as Mean Squared Error (MSE) or categorical-crossentropy can be used to

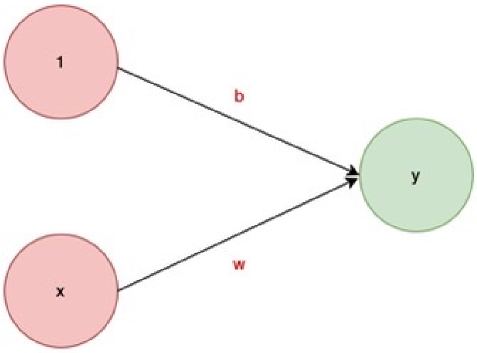


Figure 2.8: Linear Regression Network: $y = wx + b$

measure the error of the prediction, depending on the type of problem. This can be fed back to the network for backpropogation, using gradient descent, to update the weights and biases in order to improve the predictions [32; 33]. This is illustrated in Fig 2.9.

A ReLU activation function is more desirable than a linear one, as we can get derivatives of the ReLU function in order to backpropagate the network to improve the results.

$$f(x) = \begin{cases} 0 \text{ if } x < 0 \\ x \text{ if } x \geqslant 0 \end{cases} \tag{2.8}$$

The ReLU function above has derivatives 0 and 1. Any negative value will end up with an output of 0. The ReLU function is used as it diminishes the occurrence of vanishing gradients during predictive model training. Vanishing gradients occur when the gradients of the activation function get smaller as they reach a global minimum. This can get to the point where the gradients become exponentially small, and hence the gradient descent algorithm will never reach the global minimum. ReLU solves this issue, but can also lead to some of the neurons giving an output of 0.

The use of a neural network as opposed to one of the earlier classification methods outlined avoids manual feature extraction, while also providing detailed predictive modelling of the dataset that may not be as refined in a simple classification algorithm.

We can use more than one hidden layer. They allow us to build more complex function approximations than simple methods like linear or logistic regression regression, by moving away from trying to fit a linear line through data.

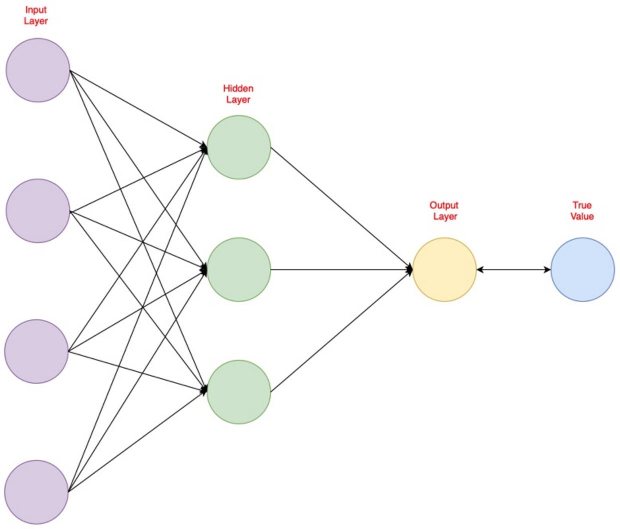


Figure 2.9: Feed Forward Neural Network

Chapter 3

Literature Review

3.1 Research Method

The search was conducted from Scopus within the NUIG Library system, IEEE Xplore and ScienceDirect, which comprised terms such as “Energy Cost Saving Artificial Intelligence” and “Home Energy Saving”. Based on these searches, the most relevant titles were found and scanned for relevant abstracts. If an article was deemed relevant to this thesis’ hypothesis, the papers cited by the article were examined to get more background.

The second set of searches revolved around similar terms, input to Google Scholar. Older papers (pre-2010) were found and scanned for relevance. These papers lay the foundation for machine learning and AI techniques in the field of energy analysis and optimisation. The papers found provide an excellent level of insight that helped guide this review.

Overall, the search proved a success and provided the basis for 3 research questions for this thesis:

RQ_1 - What are the data requirements for predicting energy usage for households?

RQ_2 - How should machine learning be applied to best predict household energy efficiency?

RQ_3 - How can visualisation aid in delivering insight about the household base from this predictive model?

These research questions serve as a foundation for the literature review.

Notes

The references included are based on the quality of the publication, all of which are peer reviewed, as well as the quality of the abstract. Books were also analysed for relevance and new techniques (both books cited are new editions published in 2021). Older research papers were preferentially selected to provide the historical development of work carried out in the field of energy analytics. As the field developed more recent works were chosen to support this thesis based on their relevance to the research questions posed. The references provide a varied sample set to work with in terms of methodology and approaches. The references are in IEEE style.

3.2 Literary Review

Energy efficiency has been at the forefront of many industries across the world, particularly in the last 50 years, due to the oil crisis of the 1970s. Both in the private and public sectors, people are looking for ways to incorporate methods of maximising energy efficiency for many reasons, but all avenues lead to one end result; a sustainable future for the human race by reducing our carbon footprint. Narciso et al presented an overview of 42 of (what they deemed) the most reputable papers in the area of energy efficiency over the last 20 years, outlining the algorithms chosen, along with input variables, pre-processing techniques etc [34]. It highlights that despite thorough research, there are still areas that are lacking in their real world application.

The field of energy efficiency is vast and, as such, not all of it is in scope for this review. The focus will be on the technical approaches, such as data processing and algorithm choice. The literature review is broken down by the research questions that needed to be addressed. From there, the research is collated and examined as a whole to gauge what is most relevant to the scope of this project.

RQ_1 : **What are the data requirements for predicting energy usage for households?**

Machine Learning in Buildings

Before devising any predictive model plan, data needs to be the forefront of analysis. There may be a lot of data available, but is it all relevant? Edwards et al attempted to solve the issue of needing many input variables in order to make energy prediction models viable in residential buildings [35]. Their data was collected from sensors attached to houses, with 140 measurements taken every 15 minutes. The sensors collected data on temperature and time, as well as previous energy consumption readings.

Ambrose et al explored all aspects of their data for energy efficiency, and how useful each one was for predictive modelling [36]. For example, energy billing data, can make it hard to determine daily energy patterns if it is calculated on a quarterly basis. Household characteristics like location, BER, size, as well as income and number of residents can be used as strong input variables for a predictive model. Attributes like indoor temperature, smart sensors readings, and what kind of lighting appliances are used are impossible to determine without survey on site, and therefore are beyond the scope of this analysis.

Mason et al took the approach of using monthly data readings to predict the next month's energy usage, which is more applicable to the data that will be explored in this thesis [37; 38; 39]. The analysis of Satre-Meloy found that household electricity use is best described through socio-demographic and physical dwelling variables, like size of the home, and ownership of an electric vehicle, for example [40]. Finally, it was found that the conversion of research from commercial to residential buildings is not straight forward, as the usage patterns can vary quite dramatically.

RQ_2 : **How should machine learning be applied to best predict household energy efficiency?**

The data needs to be processed before any predictive modelling can be done effectively. Xiao et al examined the effect of splitting data according to days [41]. For example, Monday energy usage data was used only to predict the following Monday's usage, taking into account holidays where electricity usage would be sporadic. Interestingly, they also used forecasting to predict historical data, in order to prove its validity.

Zekić-Sušac et al made use of variable reduction using χ^2 tests of independence for the factor variables and correlations for the numeric variables [42]. Based on the initial 47 variables chosen in the sample, this process determined 10 relevant input variables. The factor variables were mapped to binary categories prior to predictive modelling. Zhu et al used the method of Partial Least Squares (PLS) to find the most relevant input variables [43], while in a lot of cases other simpler methods like normalisation and outlier removal were utilised [44; 45]. Zhang et al made use of Principal Component Analysis (PCA) alongside a neural network in order to encapsulate the most important inputs in the most concise and efficient form possible [46].

Machine Learning in Financial Aid

Machine learning is only useful if it has a clear purpose. In the context of household energy efficiency and sustainability, this purpose needs to be clearly defined. The home owners will more than likely need a loan in order to carry out upgrades. BERP can tell them the benefits of in relation to how much they stand to save, and how long it will take to recoup on investment. Using this as a basis, the purpose of this analysis is to help customers in their life journey, while also making it as efficient and sustainable as possible. If someone is considering buying a new home, then a financial institution like a credit union or a bank can offer them a green specific loan that may give them a good rate of interest.

Predictive models need a platform in which they can be utilised for their purpose. Home upgrades are not a simple matter most of the time. Therefore it is about finding the best way to use the information gathered from the predictions made by the model thereafter, in a real world scenario.

Liang et al explored the effect of sustainable action on financial institution cost efficiency, and found that financial institutions who embrace sustainable actions like adoption of green products, and reducing their carbon footprint, outperform those that do not embrace these actions [47]. From a business standpoint, it is therefore in their best interests to aid home owners to move to a more sustainable lifestyle.

Following from this, Taneja et al explored customer sentiment towards financial institutions taking up sustainable products [48]. By being transparent and showing their intent for action to aid green initiatives, financial institutions gain approval from their customers. Financial institutions can do this by creating products like the ones mentioned previously, and marketing them in a personalised manner. This can be done using predictive models to determine which customers would be most open to such a product. It also helps businesses integrate with individuals in both striving to achieve a common goal while also being realistic about the money driven world we live in. This will lead to a clear definition of energy efficiency in the context of this thesis in a commercial lens as well as residential.

Feature Engineering and Extraction

Not all data is relevant for predictive modelling of the dependent variable, and must be narrowed down. Chou et al proposed a framework in which only 4 input variables (outdoor temperature in $^\circ C$, day of the week, hour of the day and the times at which these values were recorded) were taken to create a number of prediction models, using smart sensors in residential housing [49]. Their results indicated that a hybrid predictive model significantly outperforms single models (by 64%), such as support vector machines and linear regression, when using limited data.

Zekić-Sušac et al in contrast proposed a predictive model structure with 47 input variables (34 continuous and 13 categorical) [42]. They devised a preprocessing technique and gathered a sample of ≈ 1500 buildings to train a predictive model, created using a random forest integrated with the Boruta algorithm for variable reduction. They concluded that even with the optimal predicitve model found after cross validation, the accuracy was too low using this small a data source, and required a big data framework in order to get promising results. They determined that the most important input variables for their successful predictive model were: household characteristics like area, heating and time data (tying into RQ_1).

Zhao et al used the correlation coefficient between each variable in their data with the dependent variable and chose which were relevant [50]. They also performed Regression Gradient Guided Feature Selection (RGS), which was developed by Navot et al for use in study the study of brain neural activities [51].

Predictive Model Selection

Predictive model selection is crucial for success, and is unique to the dataset at hand. Edwards et al found, using environmental sensors, a LSSVM algorithm performed best on residential properties [35]. McLoughlain et al instead used time series forecasting, specifically Fourier transforms and Gaussian processes, and found both performed very well with the variable nature of electricity usage [52]. Mason et al explored ANNs, as they assume less about a dataset than other predictive models do [37]. This combined with the monthly data format proved very effective. Similarly, CNNs, ELMs, FLNNs and GAs were all applied in use cases for commercial properties, but could be altered to suit new data [53; 54; 55; 46].

RQ_3 : **How can visualisation aid in delivering insight about the household base from this predictive model?**

Visualisation

Sacha et al explored integrating data visualisation techniques with machine learning models [56]. They designed a framework in which human interactions were interwoven with machine learning models, so that results could be made more coherent, as well as allowing the changing of parameters to explore what happens on a dynamic basis.

Hadley Wickham's book, Mastering Shiny, provides a comprehensive insight into how to produce interactive dashboards, and how to deploy them for external usage, which would be relevant to any predictive model generated information produced by BERP [57]. The ability for someone to easily interact with data that is directly linked to them is beneficial in the adoption and comprehension of findings from the analysis [58].

Ruff et al explored the use of a shiny dashboard (R language) to visualise the results of a CNN, to make the predictions generated accessible to colleagues with less experience in the field of predictive modelling [59].

Data Processing: What methods can be used to convert spend data to energy data?

Shibano et al devised a predictive model to convert household income into energy usage data, by linking income to the number of electrical appliances owned and hence, the demand of these appliances, which they deduced to have followed a gamma distribution for simplification [60]. Greer provides case studies in energy cost prediction, that mimic real world predictive modelling scenarios [61]. Geo-spacial data could prove useful when mixed with cost data in determining energy usage by regional means, to give a larger view than a single household. Both can be used in conjunction to provide dual perspectives for home owners [62].

Note

The review work done by Narciso et al is the perfect foundation for any paper on this topic, and will serve as a guide for a lot of the work to come in this thesis [34].

This thesis will make a unique contribution to sustainability in that (to the best of my knowledge), it will be the first of its kind to predict BER in relation to home owner accessible information, such as household area, location, energy used etc, as well as governing bodies who seek to drive sustainability initiatives, through the use of artificial intelligence. This is the first application of machine learning to predict BER in Ireland using this type of data. The work of Shibano et al also should be noted as a key area of conversion for feature engineering [60].

3.3 Review Conclusions

Overall, this review concludes that there is an abundance of evidence to support my hypothesis that energy usage for residential households across ROI can be predicted using generally accessible user household data. The caveats however are that usage patterns are a big factor, and therefore the same strategy cannot be used for both types of properties, unless something is done to mitigate this. AI presents us with the opportunity to allow home owners to see what their current BER is, how it could impact their cost savings if it were elevated, and compare this to averages for households similar to their own, or on a regional basis. In addition, this has the potential to showcase the minimum energy a household requires/needs for a given BER. $500,000$ households are expected to be brought up to a minimum BER B2 in the Republic of Ireland by 2030 [63], and this predictive model could be used to find the minimum improvements needed to reach such a rating, whether the household is already B2 or not (in the eyes of the governing bodies rolling out the changes) and finally the impact to energy spend.

The challenges surrounding the collection and processing of data for predictive modelling are numerous, but, using techniques outlined in this review, can be minimised. For example, the data collected will be household data, which will be in the form of annual review, such as year of construction, insulation characteristics and location, and therefore sensor measurements, while providing a good level of background on predictive model input variables and techniques, will be of no benefit to my analysis.

AI can automate tasks that would take an incalculable amount of time for governing staff to complete. The predictive model and dashboard it hypothetically could feed, could be maintained and updated for new data incoming with little effort. The dashboard would promote environmental sustainability as well as saving the home owner money in an approachable, simple and user friendly way.

In conclusion, household data can be used to create a predictive model (using AI) and an interactive dashboard (using Shiny [57]). This predictive model can predict energy efficiency (BER), show this rating to the relevant bodies (home owner or government sustainability department), what their efficiency is versus other households similar to their own, and finally tips on how to reduce energy usage, to reduce cost and improve BER but also benefit the environment. There are no studies in the literature reviewed that do this.

Chapter 4

Data Processing & Analysis

4.1 Feature Selection

The dataset chosen for this analysis was taken from the Sustainable Energy Authority Of Ireland (SEAI) with permission [64]. This dataset contains information on 1.04 million private dwelling households across ROI, with 211 features recorded for each household. The data does not identify any particular individual and complies with GDPR regulations. The data [64] also contains a user guide, which is a data dictionary that was used to determine the common features that would be easily accessible by home owners without an on-site inspection, such as the year of construction or the area of the ground floor. The target variable is the BER of the household. There are 15 different classes and therefore makes this a multi-classification problem, ranging from A1 to G (best to worst).

13 features out of 211 were deemed acceptable for prediction of BER, as the other features were not easily discernible by the household occupants without outside assistance, such as C02 Rating or U-value calculations. The features are described in Table 4.1.

Feature	*Description*	*Example*	*Datatype*
CountyName	County where the household resides	Dublin	Categorical
DwellingTypeDescr	The type of household the domicile is	Detached House	Categorical
YearofConstruction	Year in which the household was built	1984	Ordinal
GroundFloorArea(sq m)	Area of the ground floor in metres squared	171.23	Continuous
MainSpaceHeatingFuel	Main type of fuel source used to heat the household living area	Electrical	Categorical
MainWaterHeatingFuel	Main type of fuel source used to heat water	Mains Gas	Categorical
VentilationMethod	The ventilation type of the household	Natural Vent	Categorical
StructureType	The type of construction used to build the house	Masonry	Categorical
NoOfSidesSheltered	The number of sheltered sides that the household has	Two	Ordinal
InsulationType	The type of insulation used in the household	Loose Jacket	Categorical
InsulationThickness	The thickness of this insulation in millimetres	20.00	Continuous
TotalDeliveredEnergy	The total amount of energy delivered to the household in the last year in kWh	25474.89	Continuous
EnergyRating	The BER of the household.	C2	Ordinal

Table 4.1: Feature Report

4.2 Feature Cleaning

The data was cleaned by evaluating each feature, and determining, through graphical and statistical analysis, if the values made sense practically. For example, any households that were built after the year 2022 (the time of writing this report) were dropped, removing 6 datapoints. Duplicate rows were dropped to ensure uniqueness throughout the dataset, having one row per household. This brought the dataset from 1,043,880 rows to 1,005,348 unique rows.

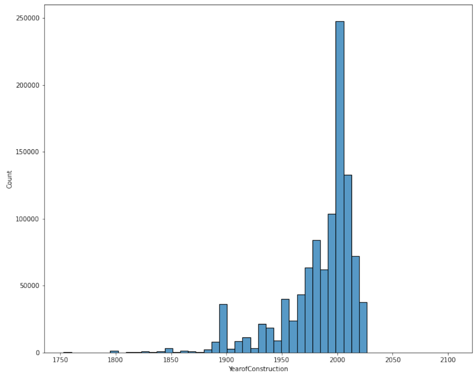


Figure 4.1: Original YearofConstruction Distribution

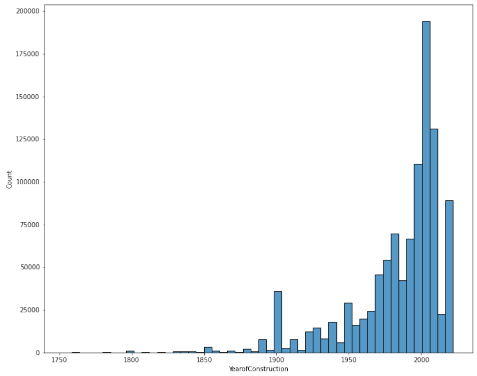


Figure 4.2: Post Removal of > 2022 built households

4.3 Feature Imputation

A gap analysis was conducted on the cleaned data, which showed a missing value issue with 8 of the 13 features.

Feature	*Number of Non Null Values*	*Number of Null Values*	*% of Null Values*
CountyName	$1,043,910$	0	0
DwellingTypeDescr	$1,043,910$	0	0
YearofConstruction	$1,043,910$	0	0
GroundFloorArea(sq m)	$1,043,910$	0	0
MainSpaceHeatingFuel	$1,028,262$	$15,648$	1.50
MainWaterHeatingFuel	$1,028,262$	$15,648$	1.50
VentilationMethod	$1,040,297$	$3,613$	0.35
StructureType	$1,040,297$	$3,613$	0.35
NoOfSidesSheltered	$1,040,297$	$3,613$	0.35
InsulationType	$811,123$	$232,787$	22.30
InsulationThickness	$811,123$	$232,787$	22.30
TotalDeliveredEnergy	$445,800$	$598,100$	57.30
EnergyRating	$1,043,910$	0	0

Table 4.2: Null Analysis Report

Three separate techniques were used to handle this using imputation as a method to fill in the gaps in the data, and compared. Firstly, the null values were dropped, leaving $390,425$ data points, 38.83% of the original cleaned data. However this left a huge class imbalance for the higher rated households (only 4 A1 rated homes left in dataset) and was therefore not deemed fit for use, as oversampling on such few datapoints is not an additional source of information for the predictive model to take advantage of. The second method was mean/mode imputation. This involved grouping the data by the fully populated features, such as YearofConstruction, CountyName and GroundFloorArea(sq m), obtaining the mean/mode for the feature based on this grouping and imputing this value where any nulls existed. This proved more successful, but again was not sufficient for use as the data did not make real world sense as a result. For example, The final and chosen method was using a random forest to impute the missing values. This was achieved using both a random forest classifier for categorical datatypes and a random forest regressor for numerical datatypes. The effectiveness of this imputer was tested on a stratified sample of the dataset that was fully populated with no null values. A copy of this data was created for comparison after imputation. Synthetic null values were made for the TotalDeliveredEnergy column at random. The data was then imputed using the missing forest algorithm. The copied data was used to compare the generated energy delivery to the original. Accuracy was determined by looking at Root Mean Squared Error (RMSE) and R^2, two regression model evaluation techniques. RMSE tells us how close the actual datapoints are to the predicted datapoints. R^2 (also known as the coefficient of determination) measures how well a predictive model can explain deviations in the target variable, and measures how well the model fits the data [65]. In Equation 4.3, y_i represents the i^{th} actual datapoint, $\hat{y_i}$ represents the i^{th} predicted datapoint and $\bar{y}$ represents the mean value of y

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - x_i)^2} \tag{4.1}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(\hat{y_i} - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{4.2}$$

Missing Forest gave an R^2 score of 0.83 for experiment, and therefore deemed the best approach for use on the entire dataset. This was also done for the grouped mean/mode method, to an R^2 of 0.64, proving its inferiority. Table 4.3 shows the missing forest imputer produces lower values to the original than the mean/mode imputer. This is because the missing forest produces duplicates if there are few supporting value for creation, which need to be dropped. The imputation comparison is shown in Figure 4.3, where it is clear missing forest models the behaviour of the original data much better than a simple mean/mode imputer. The mean/mode imputer overestimates the span of TotalDeliveredEnergy and leads to inaccuracy, especially past the original limit of 60,000kWh where it ends past 80,000kWh. In the case of A1 values, the mean/mode imputer could not impute 500 of the A1 values without duplicating them, and therefore they had to be dropped.

BER	*Number of Non Nulls Original Data*	*Dropped Nulls Method*	*Grouped Mean/Mode Impute Method*	*Missing Forest Impute Method*	*Total Original Datapoints*
A1	33	3	757	981	1258
A2	169	71	43,019	25,973	43,242
A3	1,672	865	51,168	34,284	51,194
B1	3,359	2,608	15,248	13,866	15,261
B2	8,753	7,385	32,804	31,155	32,814
B3	22,759	19,648	77,952	75,200	77,970
C1	40,083	34,849	113,773	110,384	113,782
C2	54,909	47,825	124,525	121,622	124,530
C3	61,017	53,588	118,181	116,075	118,185
D1	62,238	55,284	114,352	112,599	114,352
D2	55,155	48,914	98,186	96,922	98,187
E1	32,601	28,575	56,631	55,989	56,631
E2	27,280	23,701	44,780	44,367	44,780
F	29,081	24,954	46,346	45,956	46347
G	35,791	30,239	66,103	66,448	66815

Table 4.3: Imputation Analysis Report

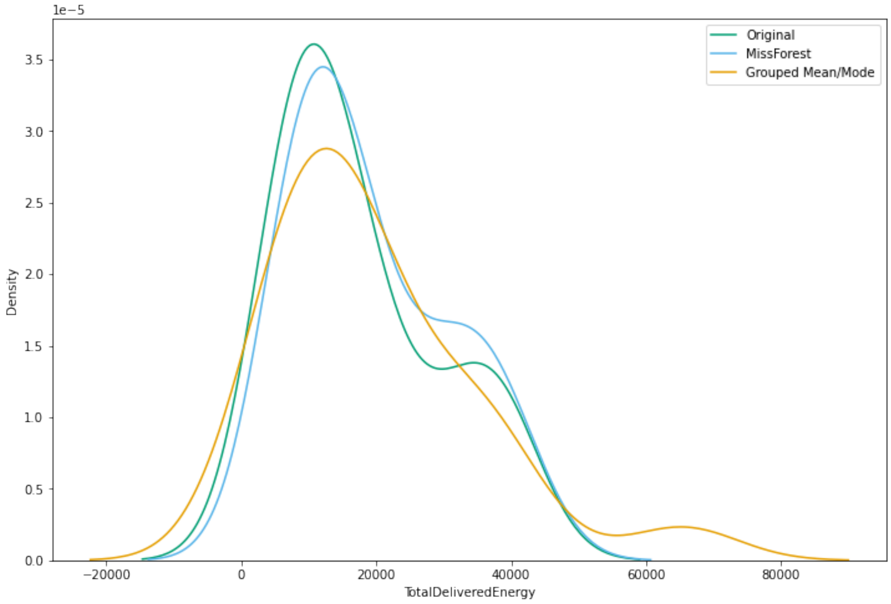


Figure 4.3: Kernel Density Plot of each method compared to original sample distribution.

This left a complete dataset with no missing or nonsensical values, without

Method	*TotalDeliveredEnergy Mean Original*	*TotalDeliveredEnergy Mean Imputed*	*Root Mean Squared Error*	R^2
Missing Forest	18,604.81	17,661.12	6313.69	0.83
Grouped Mean/Mode	18,604.81	29,693.12	9096.61	0.64

Table 4.4: Imputation Results

having to drop the majority of our dataset and lose valuable information. The imputer created duplicate rows which were dropped, leaving 951k datapoints.

4.4 Feature Engineering

Feature engineering was conducted on the cleaned, imputed data. This allowed the extraction of the cost of energy for each household per year, based on the total energy used and the average costs of electricity and gas being applied. The type of fuel was determined using the MainSpaceHeatingFuel feature. [66]. The feature was constructed using the table 4.5 and equation 4.3. The values in equation 4.3 are divided by 100 to convert from cent into euro value

$$EnergyCost(euro) = \begin{cases} \frac{TotalDeliveredEnergy(28.23)}{100}, & \text{if } MainSpaceHeatingFuel = Electricity \\ \frac{TotalDeliveredEnergy(9.51)}{100}, & \text{if } MainSpaceHeatingFuel = Sod\ Peat \\ & \vdots \\ \frac{TotalDeliveredEnergy(12.50)}{100}, & \text{Otherwise} \end{cases} \tag{4.3}$$

Note the values in table 4.5 used to supply equation 4.3 were supplied by the SEAI Domestic Fuel Cost Report published in July 2022 [67].

MainSpaceHeatingFuel Type	*Average Cost per kWh (cent)*
Heating Oil	15.57
Mains Gas	10.00
Electricity	7.93
Solid Multi-Fuel	9.51
Sod Peat	9.51
Bulk LPG (propane or butane)	15.00
House Coal	8.75
Wood Logs	13.50
Bottled LPG	24.21
Peat Briquettes	9.51
Wood Pellets (bulk supply)	9.07
Electricity - Standard Domestic	7.93
Wood Pellets (in bags)	9.52
Manufactured Smokeless Fuel	8.80
Anthracite	8.40
Wood Chips	5.92
Electricity - Off-peak Night-Rate	11.54
Electricity - On-peak Night-Rate	11.54
Bioethanol from renewable source	12.50
Biodiesel from renewable source	12.50
Other	12.50

Table 4.5: Energy Cost Mapping Table

4.5 Feature Sampling & Scaling

The dataset was split into two sets; a training set and a test set. This was in the ratio of 80:20 for the train and test set respectively. The target variable was separated from the independent predictor features. The test set ensures real world evaluation of predictive model performance after training. This split was a stratified sample, meaning the same proportion of classes were kept between the two sets. This left 761k datapoints in the training set and 190k in the test dataset

The categorical features were converted to numerical, using CatBoost encoding in both the train and test datasets, as explored in Chapter 2. The test data was not fitted and only transformed using the fitting learned in the training data to avoid any bias.

$$x^{'} = \frac{x - min(x)}{max(x) - min(x)} \tag{4.4}$$

The features, now all in numeric format, were scaled using a min-max scaler, as shown in equation 4.4, so that no one feature would dominate the others due to scale. This meant every feature had values between 0 and 1. Similar to the CatBoost Encoder, the scaler was fitted on the training data and then used to transform the training and test data, to ensure no patterns were learned in the test data. An example of a CatBoost data value is shown below.

$$Meath \rightarrow 8.129275 \tag{4.5}$$

4.6 Feature Correlation

Multi-collinearity occurs when two independent features in a dataset have a high level of influence over one another. This is measured using the correlation coefficient, which describes the relationship as shown in equation 4.6.

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{4.6}$$

Generally it is best practice to discard one of the features if there is a correlation above 0.8, which would indicate a string positive correlation between two independent features. Correlations between features were plotted as in Fig 4.4, and subsequently analyzed, to determine whether data outliers or multi-collinearity would be an issue. Any features with a score above 0.8 were dropped, but this did not occur with the chosen features. There is a high correlation of 0.75 between TotalDeliveredEnergy and EnergyCost, but these are not independent variables due to one being used to engineer the other. MainSpaceHeatingFuel and MainWaterHeatingFuel also have a high correlation with one another, but again did not meet the discard threshold of 0.8.

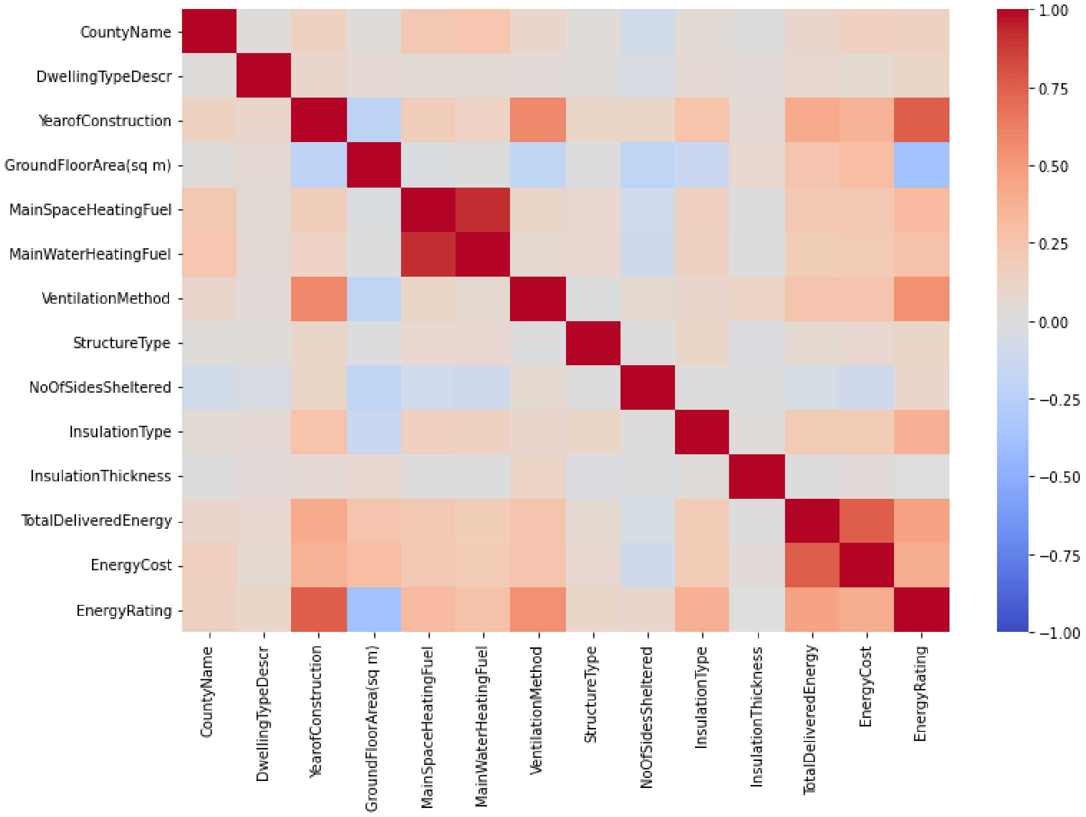


Figure 4.4: Collinearity Matrix

4.7 Feature Re-balancing

The final preprocessing step of the data was re-balancing of the training data. The test data was not sampled as this would lead to a fall real world view. Due to the A1 rated homes being by far a minority class in the dataset, STL was used to balance the dataset, using the nearest 6 datapoints to build out new synthetic data. Tomek Links then undersamples the data on all of the classes to create better boundaries between classes. In table 4.6, we can see the elevation of minority class to balance the dataset. TomekLinks influence can be seen in the case of classes such as C2, where it has been under-sampled to lead to clearer decision boundaries between the classes.

BER	$Count\ Pre\ SMOTETomek$	$Count\ Post\ SMOTETomek$
A1	981	97,490
A2	25,973	102,095
A3	34,284	103,484
B1	13,866	99,237
B2	31,155	101,049
B3	75,200	101,952
C1	110,384	97,342
C2	121,622	92,745
C3	116,075	92,590
D1	112,599	95,841
D2	96,922	100,617
E1	55,989	101,710
E2	44,367	102,063
F	45,956	103,328
G	66,448	108,597

Table 4.6: SMOTETomek Comparison

Chapter 5

Experimental Setup

5.1 ML Pipeline

The machine learning pipeline is developed as shown in the green cylinder in Fig. 5.1. The training data is fitted and transformed on the pre-processing steps outlined in Chapter 4. Finally, this training data is then used to fit a random forest model. The testing data is put through a similar pipeline process, except it is never fitted, but rather transformed, based on what each encoder/scaler learned from the training data. As mentioned in 4, the testing data is not sampled at all, as this would lead to over-inflation of our real world scenario. The training data post SMOTETomek is taken and a stratified sample is used to conduct a grid search evaluation over a parameter grid and cross validated to ensure consistent performance across folds (explored in Table 5.1). This is fed back into the pipeline to create a training model that provides the optimal model weights. The weights are used to predict the test set values. Different algorithms can be put through the pipeline.

The optimal model is saved, and can be fed as a back-end into a user friendly dashboard, where household owners can input the features and be told their BER.

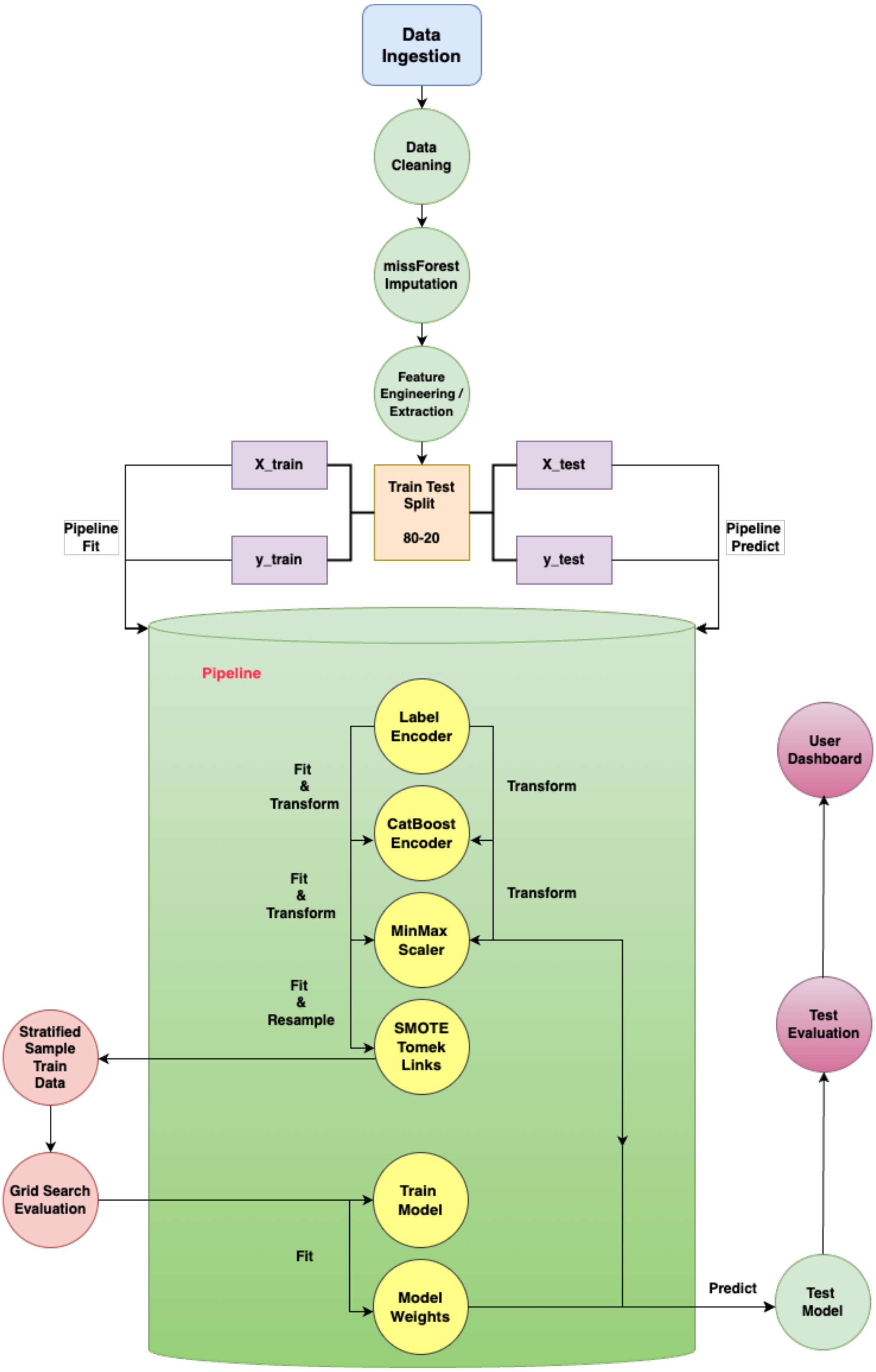


Figure 5.1: BERP Pipeline

5.2 Analysis

As discussed in Chapter 2, the methods deemed most appropriate for this analysis through background research are:

- K-Nearest Neighbours (kNN) Classification
- Decision Tree Classification
- Random Forest Classification
- Artificial Neural Network Classification

These methods were chosen prior to seeing the data. Hence there is more evidence available in the analysis phase as to why these algorithms would still be deemed suitable to work or not.

kNN was chosen because it is simple to understand and easy to interpret. However upon testing the predictive model, it performed very poorly. This is in large part to do with the scale and span of the dataset. kNN does not do well with high numbers of dimensions. Even without using one hot encoding, 13 dimensions means the distance between points becomes unclear for the algorithm to discern, and hence is fails.

Decision trees were chosen for interpretability and ease of implementation. When tested, a single tree performed well.

Hence, random forests became the next point of focus to create the optimal predictive model. This algorithm performed the best out of the 4 choices above and was chosen as the final predictive model. Random forests excel with large datasets and high dimensions of data, as it subsets the data into trees as discussed in Chapter 2.

Finally, an ANN was constructed and tested on the data. The ANN was constructed using 2 dense layers, with 15 neurons in the input later, and 15 in the hidden layer, using a linear activation function in each. These layers ended in a 15 neuron, softmax activation function layer, which was used to determine which class each belonged to. The ANN ultimately could not perform as well as the random forest. It also requires a huge amount of computational power to train, which gives the random forest model a clear advantage. The network structure is outlined in Figure 5.2.

All predictive models were tested using their base variants in sklearn using python, with a random seed set to make the analysis replicable. They were each then trained using grid search over a parameter grid, containing permutations of criteria for each model to test, and to come back with the best arrangement of arguments, outlined in table 5.1. This was also combined with k-fold stratified sampling, which shuffled the dataset into different test folds without replacement, to ensure average performance across the folds was consistent with a single test set. This ensures model stability. Each optimised model was trained and metrics such as precision, recall, accuracy and AUC were utilised to determine the best model.

Method	*Parameter*	*Range*	*Optimal Value*
kNN	Number of Neighbours	1 → 50	27
	Leaf Size	20 → 40	20
	Weights	Uniform/Distance	Distance
	Algorithm	Auto/Ball Tree/KD Tree/Brute	Auto
	Metric	Minkwoski	Minkowski
Decision Tree	Criterion	Gini/Entropy/Log Loss	Entropy
	Max Depth	10 → 100	80
	Min Samples per Split	2 → 10	6
	Min Samples per Leaf	2 → 10	4
	Splitter	Best/Random	Best
Random Forest	Criterion	Gini/Entropy/Log Loss	Log Loss
	Max Depth	40 → 50	50
	Min Samples per Split	2 → 4	2
	Min Samples per Leaf	2 → 4	4
	Number of Trees	100 → 1000	1000
ANN	Batch Size	128 → 256	256
	Weight Initialisation	Uniform/Lecun Uniform/Normal/Zero/Glorot Normal/Glorot Uniform/He Normal/He Uniform	He Uniform
	Neuron Activation Function	Softmax/Softplus/Softsign/ReLU/Tanh/Sigmoid/Hard Sigmoid/Linear	Linear
	Number of Neurons	5 → 30	15
	Dropout Regularisation Rate	0.1 → 0.9	0.1
	Weight Constraint	1.0 → 5.0	2.0

Table 5.1: GridSearch Parameters

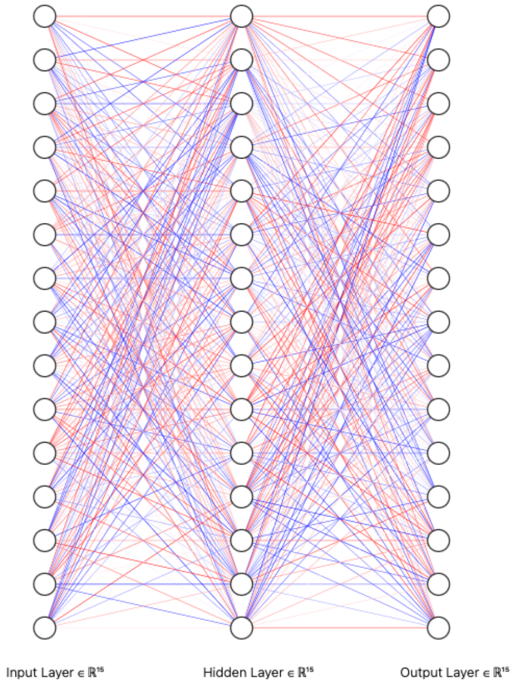

Input Layer $\in \mathbb{R}^{15}$
Hidden Layer $\in \mathbb{R}^{15}$
Output Layer $\in \mathbb{R}^{15}$

Figure 5.2: Neural Network Architecture

5.3 Metrics of Evaluation

5.3.1 Micro Average Scoring

Micro averaging works by calculating a global score across all classes, so each classes contribution is accounted for. It gives a clear view of overall performance in a model when the dataset is balanced, which in this case is true post SMOTE-Tomek sampling.

We use micro average scoring when we are required to weight each datapoint or prediction with equal weight. This is calculated across precision, recall, accuracy and f1 scores by determining the True/False Positive and Negative rates in the model.

A true positive in a classification model is a prediction the correctly finds a class in the test data. If the model predicts a household has a BER of A3, and the actual rating of this house is A3, then this is a true positive. Conversely, a false positive is a prediction that differs from the actual. It occurs when another class is predicted to be A3 in this case for example.

A true negative in a multi-class model is when all other classes are identified correctly. A false negative is the reverse of our false positive, where we would have an A3 rated home in actuality is predicted as any other class.

The true positive rate then

The problem is evaluated as if it is a binary problem for each class, and rolled up for each class when there are more than 2 classes, as is the case with BER.

Recall (also called true positive rate or sensitivity) seeks to find the proportion of how many true positive values were identified by the model correctly

$$Recall = \frac{TP}{TP + FN} \tag{5.1}$$

For example, if our model has a recall of 0.7 for C2 rated homes, this would mean it can accurately identify 70% of all C2 rated homes in the data.

Precision follows on from recall, and seeks to find out of all true positives identified, what proportion of this are actually correct.

$$Precision = \frac{TP}{TP + FP} \tag{5.2}$$

For example, if the C2 class had a precision of 0.4 in our model, this means when we predict a home being rated as C2, the model is correct in this assessment 40% of the time.

F1 is a metric that combines precision and recall by taking the harmonic mean of the two. The harmonic mean is used to ensure when precision and/or recall are low, f1 is also low.

$$F1 = \frac{2PR}{P + R} \tag{5.3}$$

AUC is a metric that calculates how well a model can distinguish between classes. This is found by calculating the area under an ROC curve, which plots the true positive rate vs false positive rate. The false positive rate is given by:

$$FPR = \frac{FP}{FP + TN} \tag{5.4}$$

$$AUC = \int_{x=0}^{1} TPR(FPR^{-1}(x))\, dx \tag{5.5}$$

Finally accuracy is the total number of predictions correctly identified over the total number of predictions

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.6}$$

Accuracy in an imbalanced dataset should not be relied upon solely, as it gives a false sense of model performance.

To find the micro scores for the above metrics, we first sum the 4 true/false positive and negative values across all 15 classes, and then use them as inputs for F1 scoring This produces a micro F1 score. Micro weighting is used to assign more weight to common values, so the model performance does not suffer due to class imbalance. It gives an agnostic view regardless of class. Despite efforts to overcome the class imbalance, this dataset still must be viewed from a micro scoring lens, as the imbalance has been fixed using synthetic data modelling and can only add so much information to our dataset when learning from a small minority of datapoints.

5.3.2 Sub Categories

The performance of the metrics explained previously can be assigned to each of the 15 classes in our dataset, to assess where the model fails in a more specified view. This can help diagnose issues with the data or model and allow hypothesis formation on how to fix the issues if they exist.

5.3.3 Generalized Performance

The generalised performance of a model can be assessed by looking at the evaluation metrics from a train and test data point of view. If the model performs well on the training data but poorly on the testing data, it may have overfit the training data. Conversely it may perform well on the test data but poorer on the training data, which would indicate it has under-fit the training data.

Chapter 6

Results

6.1 Model Evaluation

6.1.1 Micro Scoring

Model	*Precision Micro*	*Recall Micro*	*F1 Micro*	*Accuracy*	*AUC Micro*
kNN	0.33	0.33	0.33	0.33	0.59
Decision Tree	0.76	0.76	0.76	0.76	0.86
Random Forest	0.81	0.81	0.81	0.81	0.99
ANN	0.51	0.51	0.51	0.51	0.74

Table 6.1: Model Comparison on Test Set

The optimal model seeks to determine the BER of a given household. The scores reported are on the best fold score during k-fold stratified cross sampling. Through the experimental method outlined in Chapter 5, the optimal model was determined to be a Random Forest Classifier. The model was able to overcome the dimensionality limitations where kNN fails. The highest optimised kNN model was only able to reach a general F1 score of 0.33, as shown in table 6.1. A single decision tree performed well with an F1 score of 0.76, comparable to the random forest. Finally, the ANN performed poorly with an F1 score of 0.51. It also took an order of magnitude longer to train and optimise over the random forest model, which is why it was not chosen as the final model. The random forest model was strongest across all measures of evaluation, and therefore chosen as the final model.

6.1.2 Sub Categories Performance

In Table 6.2, the evaluation metrics chosen for the optimal model are shown. The same format can be found for the other models tested in Appendix A. The model is balanced across the classes after STL, and performs well. It struggles to classify and correctly predict A1 rated homes, due to the lack of information present in the data to distinguish this class enough from other A classes. The overall model has a balanced accuracy score of 0.81, with a very high AUC of 0.99. The model excels at correctly classifying actual classes, and precise in being accurate in this prediction (high precision and recall). The predictions are ranked very well according to the AUC scores, regardless of any thresholds within the model.

BER	*Precision*	*Recall*	*F1 Score*	*Accuracy*	*AUC*
A1	0.50	0.36	0.42	0.36	0.99
A2	0.89	0.88	0.88	0.88	1.0
A3	0.87	0.87	0.87	0.87	1.0
B1	0.64	0.75	0.69	0.75	0.99
B2	0.75	0.78	0.77	0.78	0.99
B3	0.82	0.84	0.83	0.84	0.99
C1	0.82	0.83	0.85	0.83	0.98
C2	0.84	0.80	0.82	0.80	0.97
C3	0.81	0.77	0.79	0.77	0.97
D1	0.80	0.77	0.79	0.77	0.97
D2	0.79	0.78	0.79	0.79	0.98
E1	0.70	0.77	0.73	0.77	0.98
E2	0.72	0.77	0.75	0.77	0.99
F	0.77	0.84	0.80	0.84	0.99
G	0.95	0.94	0.94	0.93	1.0
Micro Score	**0.81**	**0.81**	**0.81**	**0.81**	**0.99**

Table 6.2: Optimal Random Forest Model Results on Test Data

6.1.3 Generalized Performance

Method	*Split*	*Precision*	*Recall*	*F1*	*Accuracy*	*AUC*
kNN	Train Set	1.0	1.0	1.0	1.0	1.0
	Test Set	0.33	0.33	0.33	0.33	0.59
Decision Tree	Train Set	0.92	0.92	0.92	0.92	1.0
	Test Set	0.76	0.76	0.76	0.76	0.86
Random Forest	Train Set	0.92	0.92	0.92	0.92	1.0
	Test Set	0.81	0.81	0.81	0.81	0.99
ANN	Train Set	0.58	0.58	0.58	0.58	0.78
	Test Set	0.51	0.51	0.51	0.51	0.74

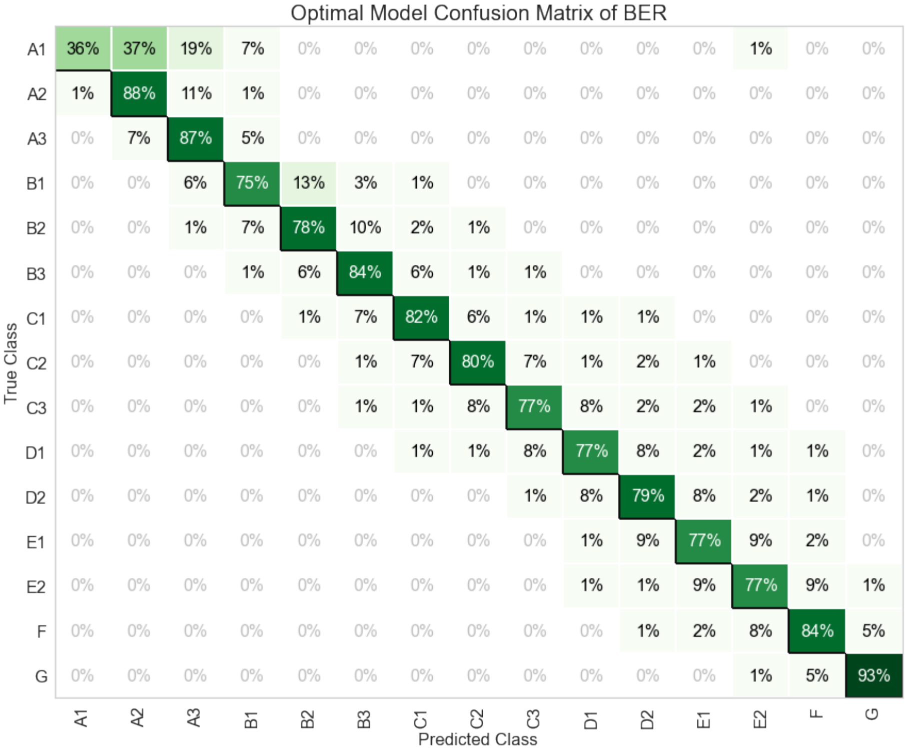
Table 6.3: Train vs Test Split Evaluation

The train and test scores for each model were found to examine model performance in a training scenario versus a real world scenario. The kNN model clearly over-fits the training data and cannot generalise well. The decision tree does comparably well to the random forest, doing well on both the training and test data. The random forest performs the best all round with a very balanced score between the train and test data. Similarly the ANN is balanced.

The random forest model can generalise very well to new unseen data and is fit for real world use.

6.1.4 Confusion Matrix

The confusion matrix shown in Fig 6.1 shows the percentage of accuracy across each class. For example, A1 rated homes are being misclassified as A2 and A3 rated homes quite a lot of the time. In all other cases, BERP is generally correct in it's classifications. The misclassifications centre around a rating one up or down from the class. For example, D2 rated homes are being misclassified as being D1 or E1 rated homes, due to the feature space not being able to differentiate them as clearly. Overall, the model has an acceptable level of misclassification, with the exception of A1 rated homes.



	A1	A2	A3	B1	B2	B3	C1	C2	C3	D1	D2	E1	E2	F	G
A1	36%	37%	19%	7%	0%	0%	0%	0%	0%	0%	0%	0%	1%	0%	0%
A2	1%	88%	11%	1%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
A3	0%	7%	87%	5%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
B1	0%	0%	6%	75%	13%	3%	1%	0%	0%	0%	0%	0%	0%	0%	0%
B2	0%	0%	1%	7%	78%	10%	2%	1%	0%	0%	0%	0%	0%	0%	0%
B3	0%	0%	0%	1%	6%	84%	6%	1%	1%	0%	0%	0%	0%	0%	0%
C1	0%	0%	0%	0%	1%	7%	82%	6%	1%	1%	1%	0%	0%	0%	0%
C2	0%	0%	0%	0%	0%	1%	7%	80%	7%	1%	2%	1%	0%	0%	0%
C3	0%	0%	0%	0%	0%	1%	1%	8%	77%	8%	2%	2%	1%	0%	0%
D1	0%	0%	0%	0%	0%	0%	1%	1%	8%	77%	8%	2%	1%	1%	0%
D2	0%	0%	0%	0%	0%	0%	0%	0%	1%	8%	79%	8%	2%	1%	0%
E1	0%	0%	0%	0%	0%	0%	0%	0%	0%	1%	9%	77%	9%	2%	0%
E2	0%	0%	0%	0%	0%	0%	0%	0%	0%	1%	1%	9%	77%	9%	1%
F	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	1%	2%	8%	84%	5%
G	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	1%	5%	93%

Figure 6.1: Confusion Matrix

6.1.5 Area Under the Receiver Operating Characteristics (AUROC)

The Receiver Operating Curve (ROC) is a probabilistic graph, that plots the true positive rate vs. the false positive rate of the BER predictions. It maps our confusion matrix into one plot, that can evaluate the model performance. It uses different classification thresholds for the predictions scores BERP generates, and converts them to class labels, in order to plot the curve. The area under the curve, AUC, is a measure of our models ability to correctly separate classes. As shown in Fig 6.2, the AUC is 0.99 for the macro, and 0.98 for the micro averages of the class space, indicating that the model can distinguish the different target classes from each other.

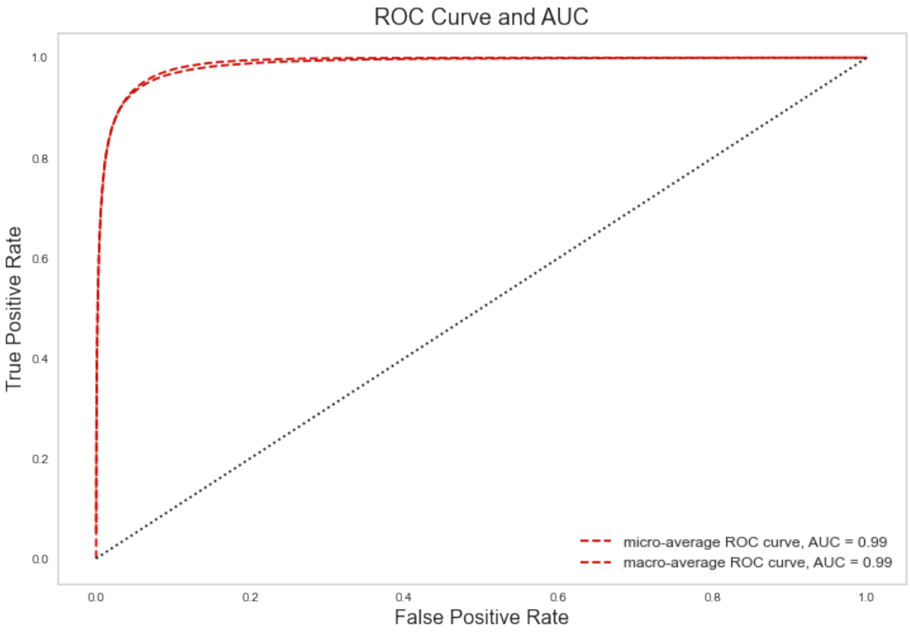


Figure 6.2: ROC Curve

6.1.6 Feature Importance

Fig 6.3 shows which features of the predictive model are most useful in determining the BER. The highest importance lies with the ground floor area of a household, contributing just over 25% of the importance alone. 4 features fall above 10% prediction contribution, while the rest fall under 5%. However, it is important to note that when the features under 5% are removed, and the model is retrained on the remaining features, the A1 scores suffer dramatically, and therefore they are required for that class alone to be acceptable. All chosen features in the model are generally available information for household owners, and can be chosen from a drop-down in our ideal scenario dashboard.

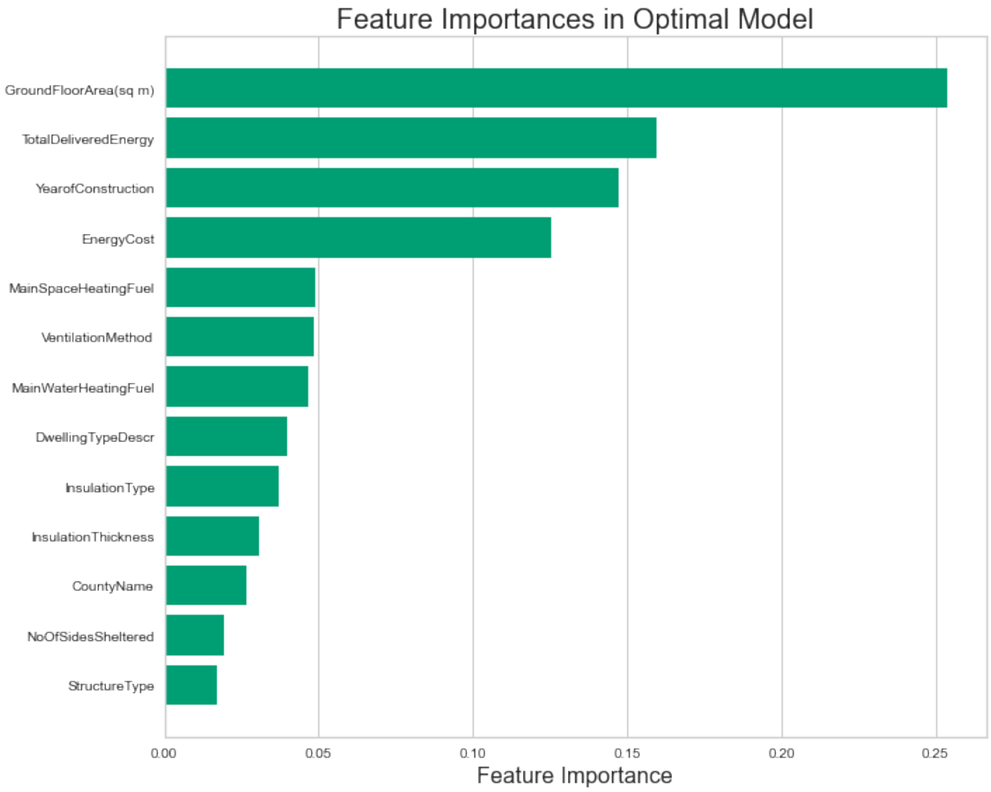


Figure 6.3: Feature Importance

6.1.7 Precision Recall Evaluation

The Precision Recall (PR) curve shows a tradeoff between precision and recall. The curve shows at different classification thresholds how precision and recall are affected. According to Saito et al, PR curves are more descriptive when evaluating imbalanced datasets after sampling [68]. In Fig 6.4, we wish to find how precision affects recall. We seek to optimise recall in our model, as it is more important in this example to find samples as opposed to extreme precision. The gradual curvature of the plot shows that the model is an excellent classifier, in comparison to the decision tree plot as shown in A.8.

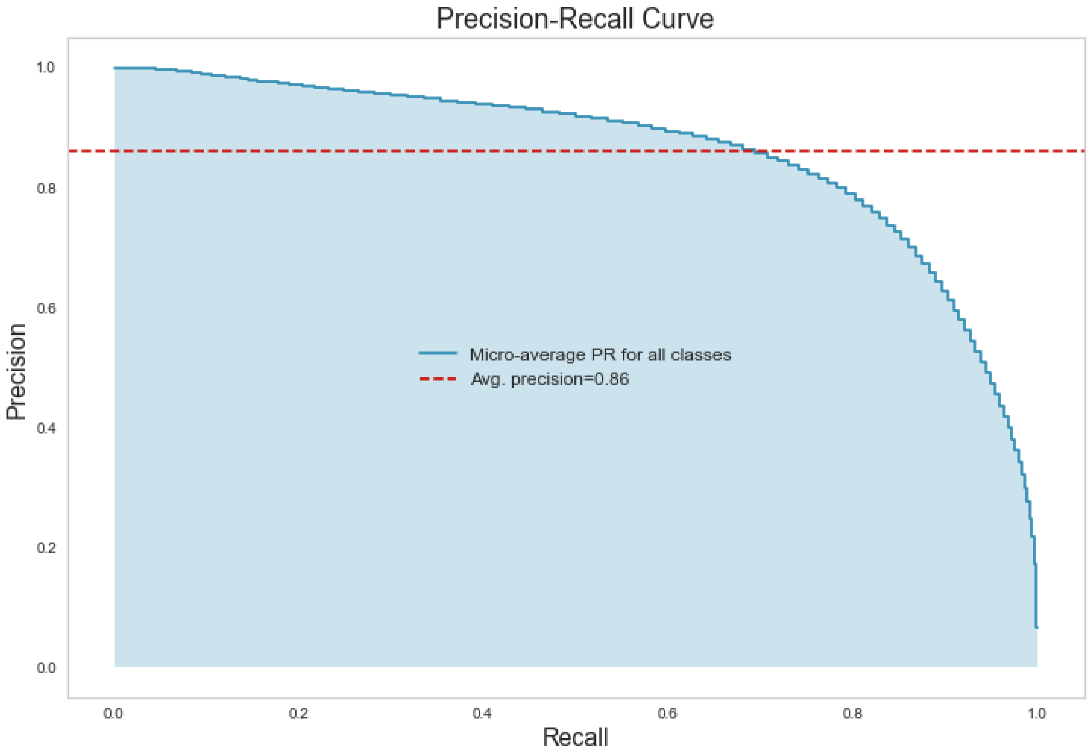


Figure 6.4: Precision-Recall Curve

Chapter 7

Discussion

Overall I feel BERP was a success in proving the hypothesis set for this thesis.

7.1 Model Usage

The fundamental question of any model is: where will it be applied, and is it able to model the full breadth of complexity in the problem it is trying to solve? If a home owner is using BERP, the F1 metric is a good baseline to use, as we want to be able to correctly identify their BER, while also making sure that it is precise, so they are informed on what actions they would need to take to increase it to an acceptable level. If it is a governing sustainability body who seek to determine the BER spread across the country and where their focus must lie, then we care most about recall, as we want to be able to identify as many households as possible for each class. However, F1 would still be acceptable here, as we also need them to be precise to ensure that we are targeting the right areas. Therefore, the general micro-F1 score of 0.81 across the classes proves and exceeds the hypothesis in both cases.

7.1.1 Interpretation of Confusion Matrix Importances

While A1 is the weakest class in our dataset due to class imbalance, it is not necessarily the most important one to classify in this use case. If an A1 household is miss-classified as an A2 for example, it does little to tell either a home owner, or a governing body, anything different about the property. Home owners with an A rated home will not likely seek to modify it any further, as they most likely already have improved their BER previously to an A rating, or their house was only recently built. The difference in cost will be negligible for the home owner, and the governing body will not be concerned with the difference A1 between A2 households. They are only interested in the general A case, as these households do not affect the drive for more B2 rated homes. Therefore, the most important classification for the governing model use case would be the B3 classification. This is the prior rating to the B2 threshold desired. If a house is miss-classified as a B2 (or higher) instead of B3, it means it has been overlooked for improvement. With 12% (with 2% of this coming from C1 being misclassed as B2) of the values being misclassified according the Fig. 6.1, this is an acceptable margin of error with the caveat that it can be improved with more B2 data. This is also true of B2 ratings being misclassed with a lower rating than B2, with 6% of B2 values being misclassed in this fashion. This is not as crucial however, as these households are already at the minimum threshold for BER.

7.1.2 Expectations

At the outset of this thesis, I was unsure of how well a model could capture the information used, given the limitations I had set upon it to include only generally accessible information to home owners, to promote usage. The class imbalances also subverted any expectations of excellent model results at the outset during data preparation. However, through iteration and transformation, the data was able to get to a state where it was perfect for a modelling scenario. The model evaluation solely lies on the real world performance of BERP on the test dataset. This dataset was transformed, but only using the transforms learned by fitting the training dataset. The imputation results were also extremely positive, and is something I will be bringing into any future work.

I had expected STL to be a substantial help in the improvement of class imbalance scores in BERP, and after implementing this technique on the data, it demonstrated how it can positively overcome the limits of simple random over/under sampling techniques.

The ANN approach fell short of expectation. The network was setup to be simple with 3 layers overall, including a dropout layer to ensure no zero weight occurrences. However, the model found it very difficult to discern between classes, even after optimisation. This is due to poor model optimisation and can be improved upon given more specified research into DNNs.

The shortcomings of the kNN model lie primarily in the dimensionality and density of the data. There are certain outliers in the dataset that are important to keep in cases of class imbalance, that clearly have affected the distance calculations. The 13 dimensions also make it very hard for distance metrics to be calculated correctly, and lead to substantial model degradation. The decision tree is easier to unravel, in that it performed well, but it was not capable and capturing the full extent of the relationships in a single tree, even when optimised.

The higher classification metrics of lower BER households is a very positive result, as these are the most important to identify, and have the most to gain from home improvements, in terms of energy savings and campaign results by governing bodies.

The year of construction feature, while not being useful for home owners in that they cannot improve it, allows generalisation. Households in a certain estate/area can generally be assumed to have similar characteristics if built by the same construction companies, and therefore one datapoint could inform 10, 20, or even 100. This is granted no development has been done on any of the houses in the area, but it gives a clear additional geo-spatial data source for BERP in construction agencies. Any future work in this area would excel from inclusion of this kind of data.

The importance of household area over energy usage is surprising. I would have expected the opposite to be the case, but both contribute highly to model predictions. The insulation type or thickness was expected to be of higher importance feature. It would be impractical for households to decrease ground floor area naturally, but again this characteristic can link households of similar size together when other data is unavailable.

Chapter 8

Conclusion

In conclusion, this work has been successful in proving and exceeding the hypothesis set. It has contributed valuable insight into the field of energy research at a residential level, by incorporation of imputation, encoding and sampling techniques from research in AI, that are generally not used together. The A1 rated households are at an inadequate evaluation metric score. Even with STL providing a clear uplift in the model, particularly in the A1 class, it still does not provide enough information for BERP to be able to discern between other A rated homes. As discussed previously, this is the ideal place for the model to fail, as this is the least important class for us to identify.

Where things have gone better than expected is in the optimisation and speed of the model pipeline. In experimentation, from ingestion to optimisation, the dataset takes about 90 minutes to complete a training run. This is on a baseline Macbook Air, so in a production environment, on a cluster of high end CPUs, this would be greatly reduced. Looking back to the research questions included at the outset of this thesis, we can examine where they have been answered and where one could direct their focus in future work.

The data requirements for determining household energy efficiency are not as specific as thought at the outset of this work. Previously, research has been dominated by modelling using sensor measurements and survey data. Using common information about a household shows that a model can be very successful without costly measures. Machine learning is best applied in a tree based decisioning approach, in terms of household energy efficiency. BERP clearly excels at differentiating classes from one another, especially at a lower threshold of poor energy efficiency, due to the majority of households in the training data having a poor efficiency rating. Distance based approaches do not work, due to the high dimensionality of the data, as well as the scale and span.

Creating a dashboard, the effective user interaction with BERP would be best suited to a visual domain. Simple statistical tables and bar charts can be created to show the energy spend one household currently consumes and how it would change if the BER was improved. Financial aid resources for this kind of upgrade to a home can be included through the SEAI schemes, or other providers.

Likewise, a separate interface could be used by governing bodies to access BERP to find poorly rated homes in certain areas, certain household sizings or energy usage points. This could add a hierarchy of importance to choose which $500,000$ households would be best suited to target for a BER improvement to B2 by 2030 across ROI [63].

This work impacts the area of residential energy modelling, and makes it more accessible to users by inclusion of simple input features. The techniques used in this thesis to fill data gaps and encode data are also combined in a pipeline, that can be simply downloaded and altered from a repository (with permission and citation) [69].

BERP was tested on my childhood home as another point of evaluation, as the property has recently been assessed by a BER site inspector and gave the correct result.

In future work in this area, I will be developing a user dashboard, trying to incorporate more abundant data sources from construction companies, or public datasets, and finally create a business property variant for commercial use.

8.1 Research Question Assessment

The data requirements for predicting energy usage for households can be found within general household data. The optimal model feature importance's outlined in Table 6.3 show that each feature was able to give a statistically significant contribution to prediction. Thus, the data requirements are volume based, as with 700 A1 rated homes, this was still not enough information to inform a difference between other classes to an acceptable level.

Machine learning should be applied in an ensemble based methodology to best predict household energy efficiency. This method clearly outperforms the other tested algorithms. The tree based prediction structure clearly works looking at the performance of a single decision tree. The random forest variant simply provides more processing power to this. A suitable deep neural network structure could outperform this but was not possible to test in this work.

Visualisation can clearly aid in delivery of insight about the household base through tabular and graphical analysis. This report has made use of many tables and graphs to inform upon the household base. It provides a clear basis for the ability for a dashboard to make use of a similar structure but in a different manner. Users do not need to see ROC curves or feature importance's, but rather insights on how their energy usage could differ with a change in BER. It can also let people know what changes would be wasted for them, like a change in insulation type that may not necessarily make their household go up a rating score. A simple indicator outperforms complex graphics in getting the point across in my experience. The simplest solution is often the best in this regard. Likewise for governing body use, a variant with a macro location view can be used to show proportions of BER distributions across Ireland based on county.

References

[1] A. More, “Survey of resampling techniques for improving classification performance in unbalanced datasets,” 2016. vi, 15

[2] J. Cruz and D. Wishart, “Applications of machine learning in cancer prediction and prognosis,” *Cancer informatics*, vol. 2, pp. 59–77, 02 2007. vi, 20

[3] A. Kaya and M. Keyes, “Energy management technology in pulp, paper and allied industries,” *IFAC Proceedings Volumes*, vol. 13, no. 4, pp. 609–622, 1980, 4th IFAC Conference on Instrumentation and Automation in the Paper, Rubber, Plastics and Polymerisation Industries, Ghent, Belgium, 3-5 June 1980. 1

[4] H. Kang, M. Lee, T. Hong, and J.-K. Choi, “Determining the optimal occupancy density for reducing the energy consumption of public office buildings: A statistical approach,” *Building and Environment*, vol. 127, pp. 173–186, 2018. 2

[5] A. Roslizar, M. A. Alghoul, B. Bakhtyar, N. Asim, and K. Sopian, “Annual energy usage reduction and cost savings of a school: End-use energy analysis,” *The Scientific World Journal*, vol. 2014, pp. 310–539, Nov 2014. 2

[6] J. A. Rodger, “A fuzzy nearest neighbor neural network statistical model for predicting demand for natural gas and energy cost savings in public buildings,” *Expert Systems with Applications*, vol. 41, no. 4, Part 2, pp. 1813–1829, 2014. 2

[7] M. Santamouris, G. Mihalakakou, P. Patargias, N. Gaitani, K. Sfakianaki, M. Papaglastra, C. Pavlou, P. Doukas, E. Primikiri, V. Geros, M. Assimakopoulos, R. Mitoula, and S. Zerefos, “Using intelligent clustering techniques to classify the energy performance of school buildings,” *Energy and Buildings*, vol. 39, no. 1, pp. 45–51, 2007. 2

[8] D. Torregrossa, U. Leopold, F. Hernández-Sancho, and J. Hansen, “Machine learning for energy cost modelling in wastewater treatment plants,” *Journal of Environmental Management*, vol. 223, pp. 1061–1067, 2018. 2

[9] C. Department of the Environment and Communications, “2050 Net-Zero Act,” 08 2021. [Online]. Available: https://www.gov.ie/en/press-release/9336b-irelands-ambitious-climate-act-signed-into-law/ 2

[10] T. W. Edgar and D. O. Manz, “Chapter 6 - machine learning,” in *Research Methods for Cyber Security*, T. W. Edgar and D. O. Manz, Eds. Syngress, 2017, pp. 153–173. 5

[11] V. N. Dornadula and S. Geetha, “Credit card fraud detection using machine learning algorithms,” *Procedia Computer Science*, vol. 165, pp. 631–641, 2019, 2nd International Conference on Recent Trends in Advanced Computing ICRTAC -DISRUP - TIV INNOVATION , 2019 November 11-12, 2019. 6

[12] S. Lee, Y. Kim, H. Kahng, S.-K. Lee, S. Chung, T. Cheong, K. Shin, J. Park, and S. B. Kim, “Intelligent traffic control for autonomous vehicle systems

based on machine learning," *Expert systems with applications*, vol. 144, p. 113074, 2020. 6

[13] I. P. Adegun and H. B. Vadapalli, "Facial micro-expression recognition: A machine learning approach," *Scientific African*, vol. 8, p. e00465, 2020. 6

[14] C. Schröer, F. Kruse, and J. M. Gómez, "A systematic literature review on applying crisp-dm process model," *Procedia Computer Science*, vol. 181, pp. 526–534, 2021, cENTERIS 2020 - International Conference on ENTERprise Information Systems / ProjMAN 2020 - International Conference on Project MANagement / HCist 2020 - International Conference on Health and Social Care Information Systems and Technologies 2020, CENTERIS/ProjMAN/HCist 2020. 7

[15] R. Wirth and J. Hipp, "Crisp-dm: towards a standard process modell for data mining," 2000. 7

[16] H. Wickham, "Tidy data," *The American Statistician*, vol. 14, 09 2014. 7

[17] W.-C. Lin and C.-F. Tsai, "Missing value imputation: a review and analysis of the literature (2006–2017)," *Artificial Intelligence Review*, vol. 53, no. 2, pp. 1487–1509, Feb 2020. 9

[18] D. J. Stekhoven and P. Bühlmann, "MissForest—non-parametric missing value imputation for mixed-type data," *Bioinformatics*, vol. 28, no. 1, pp. 112–118, 10 2011. 9

[19] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "Catboost: unbiased boosting with categorical features," *Advances in neural information processing systems*, vol. 31, 2018. 11

[20] Yandex, "Catboost Yandex," 06 2022. [Online]. Available: https://catboost.ai/en/docs/concepts/algorithm-main-stages_cat-to-numberic 12

[21] K. W. Bowyer, N. V. Chawla, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *CoRR*, vol. abs/1106.1813, 2011. 14

[22] I. Tomek, "Two modifications of cnn," 1976. 15

[23] G. E. A. P. A. Batista, A. L. C. Bazzan, and M. C. Monard, "Balancing training data for automated annotation of keywords: a case study," in *WOB*, 2003. 16

[24] M. Alloghani, D. Al-Jumeily Obe, J. Mustafina, A. Hussain, and A. Aljaaf, *A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science*, 01 2020, pp. 3–21. 17

[25] J. S. Richman, "Chapter thirteen - multivariate neighborhood sample entropy: A method for data reduction and prediction of complex data," in *Computer Methods, Part C*, ser. Methods in Enzymology, M. L. Johnson and L. Brand, Eds. Academic Press, 2011, vol. 487, pp. 397–408. 17

[26] R. Bellman, "Dynamic programming," *Science*, vol. 153, no. 3731, pp. 34–37, 1966. 18

[27] W. L. Dunn and J. K. Shultis, "5 - variance reduction techniques," in *Exploring Monte Carlo Methods*, W. L. Dunn and J. K. Shultis, Eds. Amsterdam: Elsevier, 2012, pp. 97–132. 20

[28] W.-Y. Loh, "Classification and regression trees," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, pp. 14 – 23, 01 2011. 21

[29] X. Chen, Z. Y. Dong, K. Meng, Y. Xu, K. P. Wong, and H. W. Ngan, “Electricity price forecasting with extreme learning machine and bootstrapping,” *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 2055–2062, 2012. 22

[30] A. Prieto, B. Prieto, E. M. Ortigosa, E. Ros, F. Pelayo, J. Ortega, and I. Rojas, “Neural networks: An overview of early research, current frameworks and new challenges,” *Neurocomputing*, vol. 214, pp. 242–268, 2016. 23

[31] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, “Activation functions: Comparison of trends in practice and research for deep learning,” 2018. 23

[32] E. Guresen and G. Kayakutlu, “Definition of artificial neural networks with comparison to other networks,” *Procedia Computer Science*, vol. 3, pp. 426–433, 2011, world Conference on Information Technology. 24

[33] S. Ruder, “An overview of gradient descent optimization algorithms,” 2017. 24

[34] D. A. Narciso and F. Martins, “Application of machine learning tools for energy efficiency in industry: A review,” *Energy Reports*, vol. 6, pp. 1181–1199, 2020. 28, 33

[35] R. E. Edwards, J. New, and L. E. Parker, “Predicting future hourly residential electrical consumption: A machine learning case study,” *Energy and Buildings*, vol. 49, pp. 591–603, 2012. 28, 32

[36] M. Ambrose and M. James, “Dealing with energy efficiency data,” *Energy Procedia*, vol. 121, pp. 158–165, 2017, improving Residential Energy Efficiency International Conference, IREE 2017. 28

[37] K. Mason, J. Duggan, and E. Howley, “Forecasting energy demand, wind generation and carbon dioxide emissions in ireland using evolutionary neural networks,” *Energy*, vol. 155, pp. 705–720, 2018. 29, 32

[38] J. Lu, P. Mannion, and K. Mason, “A multi-objective multi-agent deep reinforcement learning approach to residential appliance scheduling,” *IET Smart Grid*, vol. 5, pp. n/a–n/a, 05 2022. 29

[39] K. Mason and S. Grijalva, “Building hvac control via neural networks and natural evolution strategies,” in *2021 IEEE Congress on Evolutionary Computation (CEC).* IEEE Press, 2021, p. 2483–2490. 29

[40] A. Satre-Meloy, “Investigating structural and occupant drivers of annual residential electricity consumption using regularization in regression models,” *Energy*, vol. 174, pp. 148–168, 2019. 29

[41] L. Xiao, J. Wang, X. Yang, and L. Xiao, “A hybrid model based on data preprocessing for electrical power forecasting,” *International Journal of Electrical Power Energy Systems*, vol. 64, pp. 311–327, 2015. 29

[42] M. Zekić-Sušac, A. Has, and M. Knežević, “Predicting energy cost of public buildings by artificial neural networks, cart, and random forest,” *Neurocomputing*, vol. 439, pp. 223–233, 2021. 29, 31

[43] L. Zhu and J. Chen, “Energy efficiency evaluation and prediction of large-scale chemical plants using partial least squares analysis integrated with gaussian process models,” *Energy Conversion and Management*, vol. 195, pp. 690–700, 2019. 29

[44] Y. Han, C. Fan, M. Xu, Z. Geng, and Y. Zhong, “Production capacity analysis and energy saving of complex chemical processes using lstm based

on attention mechanism," *Applied Thermal Engineering*, vol. 160, p. 114072, 2019. 29

[45] B. Beisheim, K. Rahimi-Adli, S. Krämer, and S. Engell, "Energy performance analysis of continuous processes using surrogate models," *Energy*, vol. 183, pp. 776–787, 2019. 29

[46] X.-H. Zhang, Q.-X. Zhu, Y.-L. He, and Y. Xu, "Energy modeling using an effective latent variable based functional link learning machine," *Energy*, vol. 162, pp. 883–891, 2018. 30, 32

[47] L.-W. Liang, H.-Y. Chang, and H.-L. Shao, "Does sustainability make banks more cost efficient?" *Global Finance Journal*, vol. 38, pp. 13–23, 2018, special Issue on Corporate Social Responsibility and Ethics in Financial Markets. 30

[48] S. Taneja and L. Ali, "Determinants of customers' intentions towards environmentally sustainable banking: Testing the structural model," *Journal of Retailing and Consumer Services*, vol. 59, p. 102418, 2021. 30

[49] J.-S. Chou and D.-S. Tran, "Forecasting energy consumption time series using machine learning techniques based on usage patterns of residential householders," *Energy*, vol. 165, pp. 709–726, 2018. 31

[50] H. Zhao and F. Magoulès, "Feature selection for predicting building energy consumption based on statistical learning method," *Journal of Algorithms and Computational Technology*, vol. 6, pp. 59 – 77, 2012. 32

[51] A. Navot, L. Shpigelman, N. Tishby, and E. Vaadia, "Nearest neighbor based feature selection for regression and its application to neural activity," vol. 18, 01 2005. 32

[52] F. McLoughlin, A. Duffy, and M. Conlon, “Evaluation of time series techniques to characterise domestic electricity demand,” *Energy*, vol. 50, pp. 120–130, 2013. 32

[53] Z. Geng, Y. Zhang, C. Li, Y. Han, Y. Cui, and B. Yu, “Energy optimization and prediction modeling of petrochemical industries: An improved convolutional neural network based on cross-feature,” *Energy*, vol. 194, p. 116851, 2020. 32

[54] Y.-L. He, P.-J. Wang, M.-Q. Zhang, Q.-X. Zhu, and Y. Xu, “A novel and effective nonlinear interpolation virtual sample generation method for enhancing energy prediction and analysis on small data problem: A case study of ethylene industry,” *Energy*, vol. 147, pp. 418–427, 2018. 32

[55] M. Kovačič and B. Šarler, “Genetic programming prediction of the natural gas consumption in a steel plant,” *Energy*, vol. 66, pp. 273–284, 2014. 32

[56] D. Sacha, M. Sedlmair, L. Zhang, J. A. Lee, J. Peltonen, D. Weiskopf, S. C. North, and D. A. Keim, “What you see is what you can change: Human-centered machine learning by interactive visualization,” *Neurocomputing*, vol. 268, pp. 164–175, 2017. 32

[57] H. Wickham, *Mastering Shiny.* O’Reilly, 2021. 33, 35

[58] Y.-S. Kim, K. Reinecke, and J. Hullman, “Explaining the gap: Visualizing one’s predictions improves recall and comprehension of data,” in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, ser. CHI ’17. New York, NY, USA: Association for Computing Machinery, 2017, p. 1375–1386. 33

[59] Z. J. Ruff, D. B. Lesmeister, C. L. Appel, and C. M. Sullivan, “Workflow and

convolutional neural network for automated identification of animal sounds," *Ecological Indicators*, vol. 124, p. 107419, 2021. 33

[60] K. Shibano and G. Mogi, "Electricity consumption forecast model using household income: Case study in tanzania," *Energies*, vol. 13, no. 10, 2020. 33, 34

[61] M. Greer, *Electricity Cost Modelling Calculations - 2nd Edition.* Academic Press, 2021. 33

[62] M. V. Rocco, E. Fumagalli, C. Vigone, A. Miserocchi, and E. Colombo, "Enhancing energy models with geo-spatial data for the analysis of future electrification pathways: The case of tanzania," *Energy Strategy Reviews*, vol. 34, p. 100614, 2021. 33

[63] E. G. on Future Skills Needs, "2030 BER B2 Target," 11 2021. [Online]. Available: https://enterprise.gov.ie/en/publications/skills-for-zero-carbon.html 34, 66

[64] SEAI, "SEAI Data," 06 2022. [Online]. Available: https://ndber.seai.ie/BERResearchTool/ber/search.aspx 36

[65] P. Ali and A. Younas, "Understanding and interpreting regression analysis," *Evidence-Based Nursing*, vol. 24, no. 4, pp. 116–118, 2021. 39

[66] Sustainable Energy Authority Of Ireland (SEAI), "Average Electricity & Gas prices Ireland 2022," 04 2022. [Online]. Available: https://www.seai.ie/data-and-insights/seai-statistics/key-statistics/prices/ 41

[67] Sustainable Energy Authority Of Ireland (SEAI) Department, "Domestic Fuel Cost Comparison," 07 2022. [Online]. Available: https://www.seai.ie/publications/Domestic-Fuel-Cost-Comparison.pdf 41

[68] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets," *PLOS ONE*, vol. 10, no. 3, pp. 1–21, 03 2015. 60

[69] O. Brannock, "Building Energy Rating Predictor (BERP) Repository," 08 2022. [Online]. Available: https://github.com/OisinB-2814/masters_thesis_ob2814 66

Appendix A

Appendix

BER	*Precision*	*Recall*	*F1 Score*	*Accuracy*	*AUC*
A1	0.10	0.35	0.15	0.35	0.67
A2	0.74	0.75	0.74	0.75	0.87
A3	0.72	0.67	0.70	0.67	0.83
B1	0.20	0.47	0.28	0.47	0.72
B2	0.22	0.41	0.29	0.41	0.68
B3	0.29	0.35	0.32	0.35	0.64
C1	0.34	0.28	0.31	0.28	0.61
C2	0.34	0.22	0.27	0.22	0.58
C3	0.31	0.23	0.27	0.23	0.58
D1	0.30	0.23	0.26	0.23	0.58
D2	0.29	0.25	0.26	0.79	0.59
E1	0.19	0.29	0.23	0.29	0.61
E2	0.18	0.30	0.22	0.30	0.62
F	0.25	0.35	0.29	0.35	0.65
G	0.69	0.64	0.66	0.64	0.81
Micro Score	**0.81**	**0.81**	**0.81**	**0.81**	**0.59**

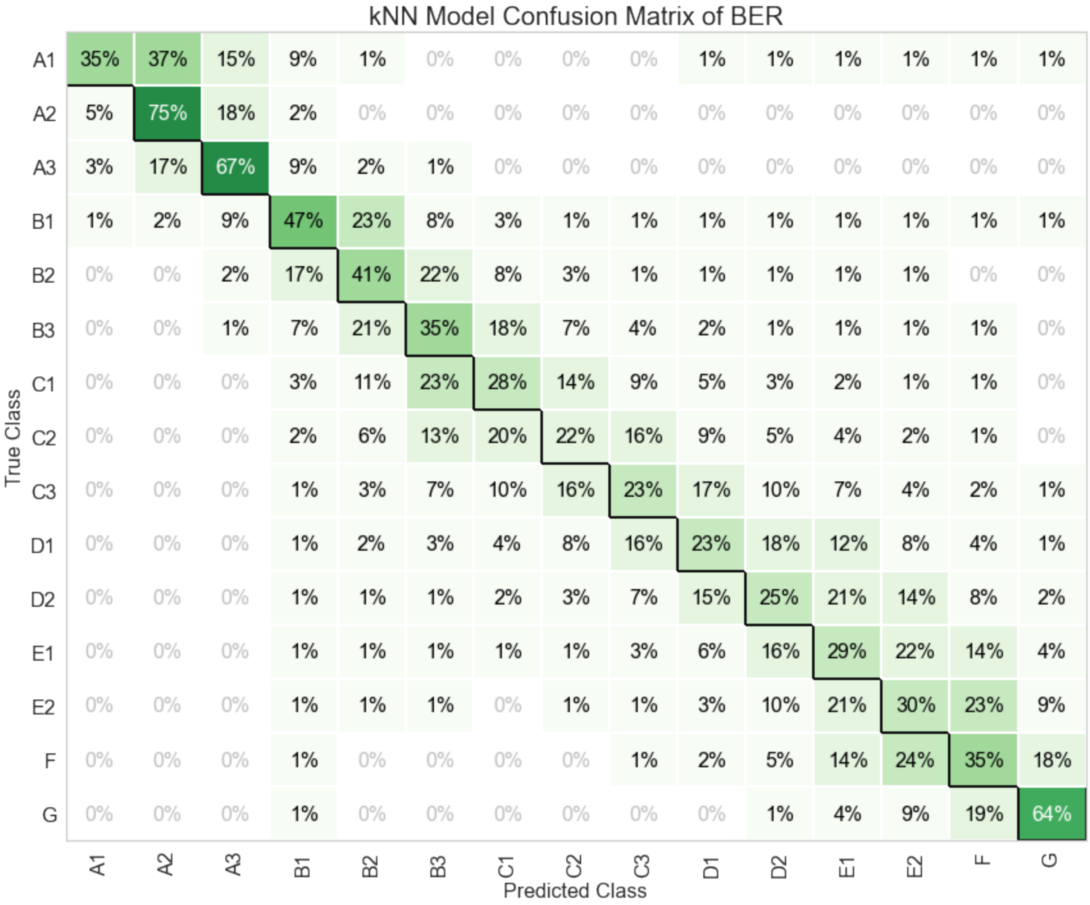
Table A.1: kNN Model Results on Test Data

BER	*Precision*	*Recall*	*F1 Score*	*Accuracy*	*AUC*
A1	0.25	0.38	0.30	0.38	0.69
A2	0.85	0.82	0.84	0.82	0.92
A3	0.85	0.79	0.82	0.79	0.90
B1	0.57	0.67	0.62	0.67	0.84
B2	0.67	0.71	0.69	0.71	0.86
B3	0.80	0.73	0.76	0.73	0.87
C1	0.82	0.73	0.77	0.73	0.87
C2	0.81	0.72	0.76	0.72	0.86
C3	0.78	0.69	0.73	0.69	0.85
D1	0.78	0.69	0.73	0.69	0.85
D2	0.79	0.70	0.74	0.70	0.85
E1	0.71	0.68	0.70	0.68	0.85
E2	0.72	0.71	0.72	0.71	0.86
F	0.79	0.78	0.78	0.78	0.89
G	0.94	0.91	0.93	0.91	0.95
Micro Score	**0.76**	**0.76**	**0.76**	**0.76**	**0.86**

Table A.2: Decision Tree Model Results on Test Data

BER	*Precision*	*Recall*	*F1 Score*	*Accuracy*	*AUC*
A1	0.04	0.45	0.07	0.45	0.74
A2	0.68	0.56	0.61	0.56	0.75
A3	0.71	0.61	0.65	0.61	0.79
B1	0.38	0.57	0.46	0.57	0.77
B2	0.40	0.68	0.50	0.68	0.83
B3	0.58	0.57	0.58	0.57	0.78
C1	0.52	0.57	0.54	0.57	0.77
C2	0.49	0.38	0.42	0.38	0.68
C3	0.48	0.33	0.39	0.33	0.65
D1	0.46	0.40	0.43	0.40	0.69
D2	0.44	0.43	0.44	0.43	0.70
E1	0.35	0.54	0.42	0.54	0.76
E2	0.52	0.62	0.57	0.62	0.81
F	0.70	0.72	0.71	0.72	0.87
G	0.92	0.89	0.90	0.89	0.95
Micro Score	**0.52**	**0.52**	**0.52**	**0.52**	**0.74**

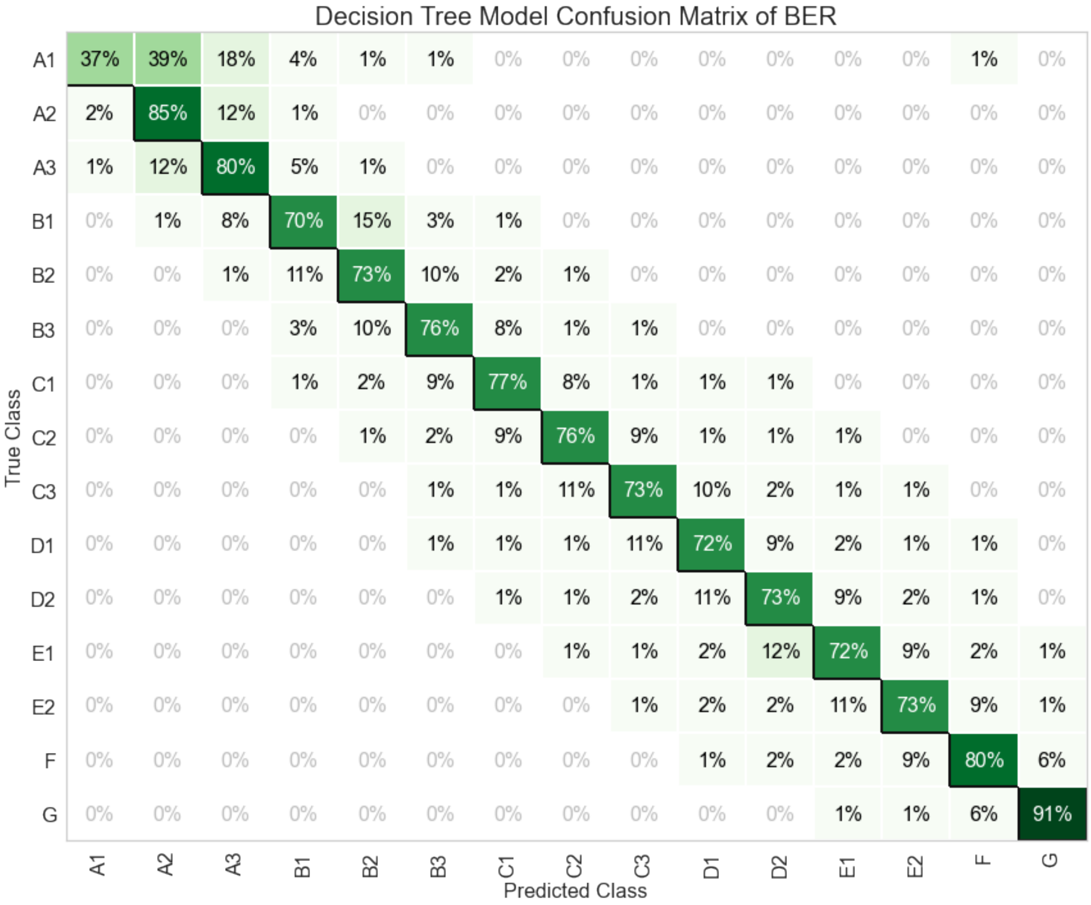
Table A.3: ANN Model Results on Test Data



kNN Model Confusion Matrix of BER

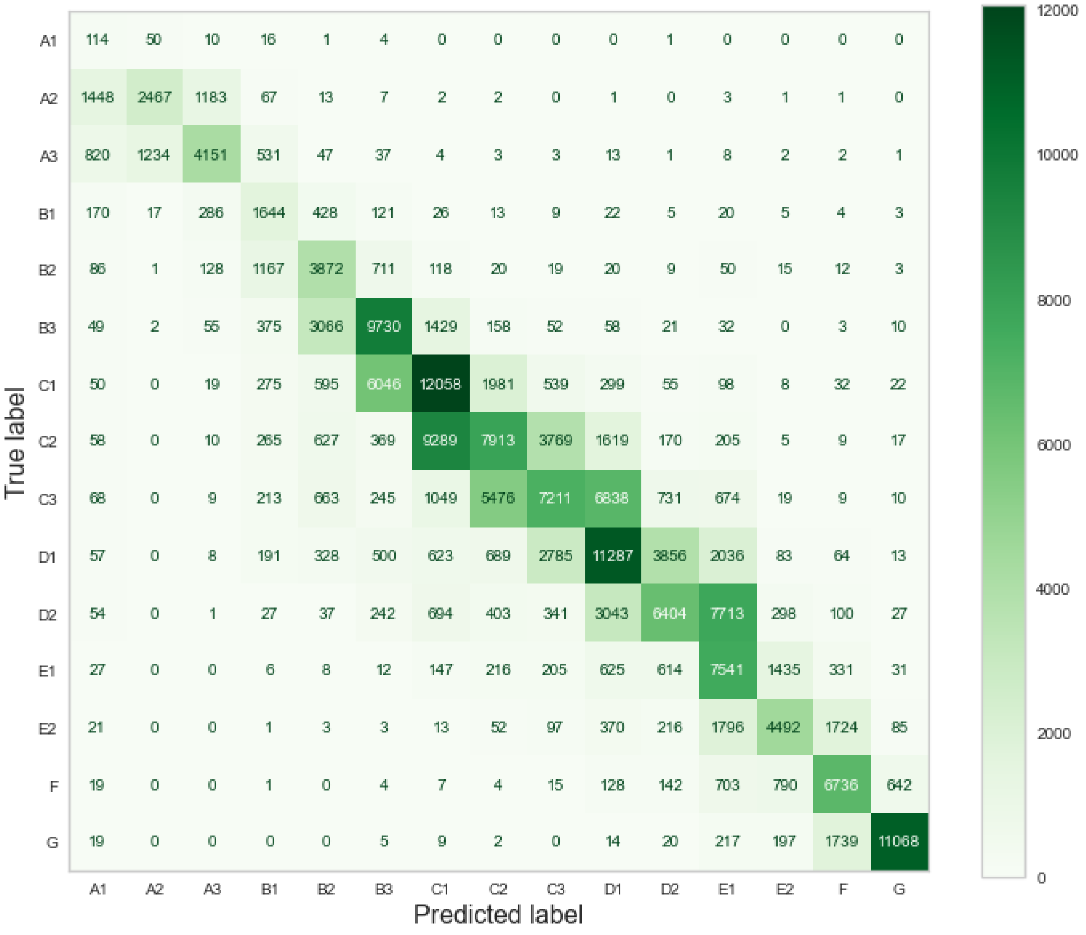
True Class \ Predicted Class	A1	A2	A3	B1	B2	B3	C1	C2	C3	D1	D2	E1	E2	F	G
A1	35%	37%	15%	9%	1%	0%	0%	0%	0%	1%	1%	1%	1%	1%	1%
A2	5%	75%	18%	2%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
A3	3%	17%	67%	9%	2%	1%	0%	0%	0%	0%	0%	0%	0%	0%	0%
B1	1%	2%	9%	47%	23%	8%	3%	1%	1%	1%	1%	1%	1%	1%	1%
B2	0%	0%	2%	17%	41%	22%	8%	3%	1%	1%	1%	1%	1%	0%	0%
B3	0%	0%	1%	7%	21%	35%	18%	7%	4%	2%	1%	1%	1%	1%	0%
C1	0%	0%	0%	3%	11%	23%	28%	14%	9%	5%	3%	2%	1%	1%	0%
C2	0%	0%	0%	2%	6%	13%	20%	22%	16%	9%	5%	4%	2%	1%	0%
C3	0%	0%	0%	1%	3%	7%	10%	16%	23%	17%	10%	7%	4%	2%	1%
D1	0%	0%	0%	1%	2%	3%	4%	8%	16%	23%	18%	12%	8%	4%	1%
D2	0%	0%	0%	1%	1%	1%	2%	3%	7%	15%	25%	21%	14%	8%	2%
E1	0%	0%	0%	1%	1%	1%	1%	1%	3%	6%	16%	29%	22%	14%	4%
E2	0%	0%	0%	1%	1%	1%	0%	1%	1%	3%	10%	21%	30%	23%	9%
F	0%	0%	0%	1%	0%	0%	0%	0%	1%	2%	5%	14%	24%	35%	18%
G	0%	0%	0%	1%	0%	0%	0%	0%	0%	0%	1%	4%	9%	19%	64%

Figure A.1: kNN Confusion Matrix



	A1	A2	A3	B1	B2	B3	C1	C2	C3	D1	D2	E1	E2	F	G
A1	37%	39%	18%	4%	1%	1%	0%	0%	0%	0%	0%	0%	0%	1%	0%
A2	2%	85%	12%	1%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
A3	1%	12%	80%	5%	1%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
B1	0%	1%	8%	70%	15%	3%	1%	0%	0%	0%	0%	0%	0%	0%	0%
B2	0%	0%	1%	11%	73%	10%	2%	1%	0%	0%	0%	0%	0%	0%	0%
B3	0%	0%	0%	3%	10%	76%	8%	1%	1%	0%	0%	0%	0%	0%	0%
C1	0%	0%	0%	1%	2%	9%	77%	8%	1%	1%	1%	0%	0%	0%	0%
C2	0%	0%	0%	0%	1%	2%	9%	76%	9%	1%	1%	1%	0%	0%	0%
C3	0%	0%	0%	0%	0%	1%	1%	11%	73%	10%	2%	1%	1%	0%	0%
D1	0%	0%	0%	0%	0%	1%	1%	1%	11%	72%	9%	2%	1%	1%	0%
D2	0%	0%	0%	0%	0%	0%	1%	1%	2%	11%	73%	9%	2%	1%	0%
E1	0%	0%	0%	0%	0%	0%	0%	1%	1%	2%	12%	72%	9%	2%	1%
E2	0%	0%	0%	0%	0%	0%	0%	0%	1%	2%	2%	11%	73%	9%	1%
F	0%	0%	0%	0%	0%	0%	0%	0%	0%	1%	2%	2%	9%	80%	6%
G	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	1%	1%	6%	91%

Figure A.2: Decision Tree Confusion Matrix



True label \ Predicted label	A1	A2	A3	B1	B2	B3	C1	C2	C3	D1	D2	E1	E2	F	G
A1	114	50	10	16	1	4	0	0	0	0	1	0	0	0	0
A2	1448	2467	1183	67	13	7	2	2	0	1	0	3	1	1	0
A3	820	1234	4151	531	47	37	4	3	3	13	1	8	2	2	1
B1	170	17	286	1644	428	121	26	13	9	22	5	20	5	4	3
B2	86	1	128	1167	3872	711	118	20	19	20	9	50	15	12	3
B3	49	2	55	375	3066	9730	1429	158	52	58	21	32	0	3	10
C1	50	0	19	275	595	6046	12058	1981	539	299	55	98	8	32	22
C2	58	0	10	265	627	369	9289	7913	3769	1619	170	205	5	9	17
C3	68	0	9	213	663	245	1049	5476	7211	6838	731	674	19	9	10
D1	57	0	8	191	328	500	623	689	2785	11287	3856	2036	83	64	13
D2	54	0	1	27	37	242	694	403	341	3043	6404	7713	298	100	27
E1	27	0	0	6	8	12	147	216	205	625	614	7541	1435	331	31
E2	21	0	0	1	3	3	13	52	97	370	216	1796	4492	1724	85
F	19	0	0	1	0	4	7	4	15	128	142	703	790	6736	642
G	19	0	0	0	0	5	9	2	0	14	20	217	197	1739	11068

Figure A.3: Neural Network Confusion Matrix

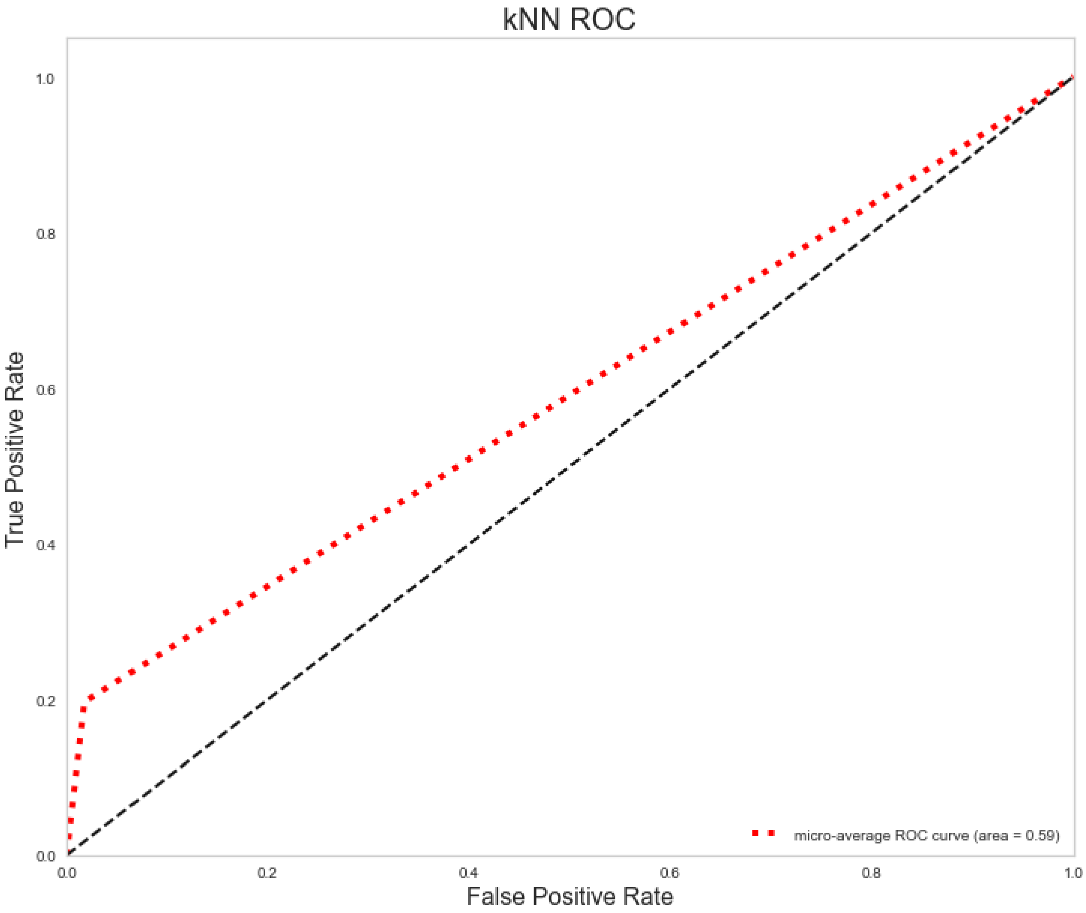


Figure A.4: kNN ROC

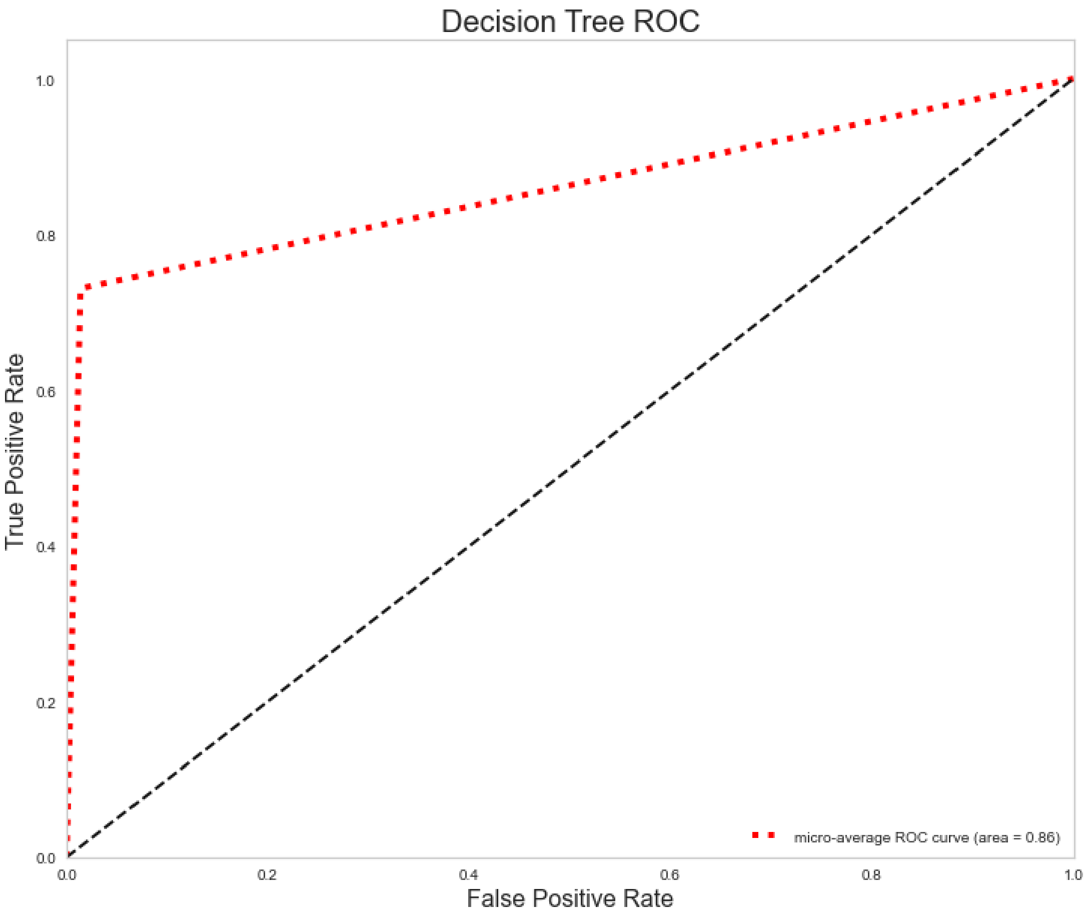


Figure A.5: Decision Tree ROC

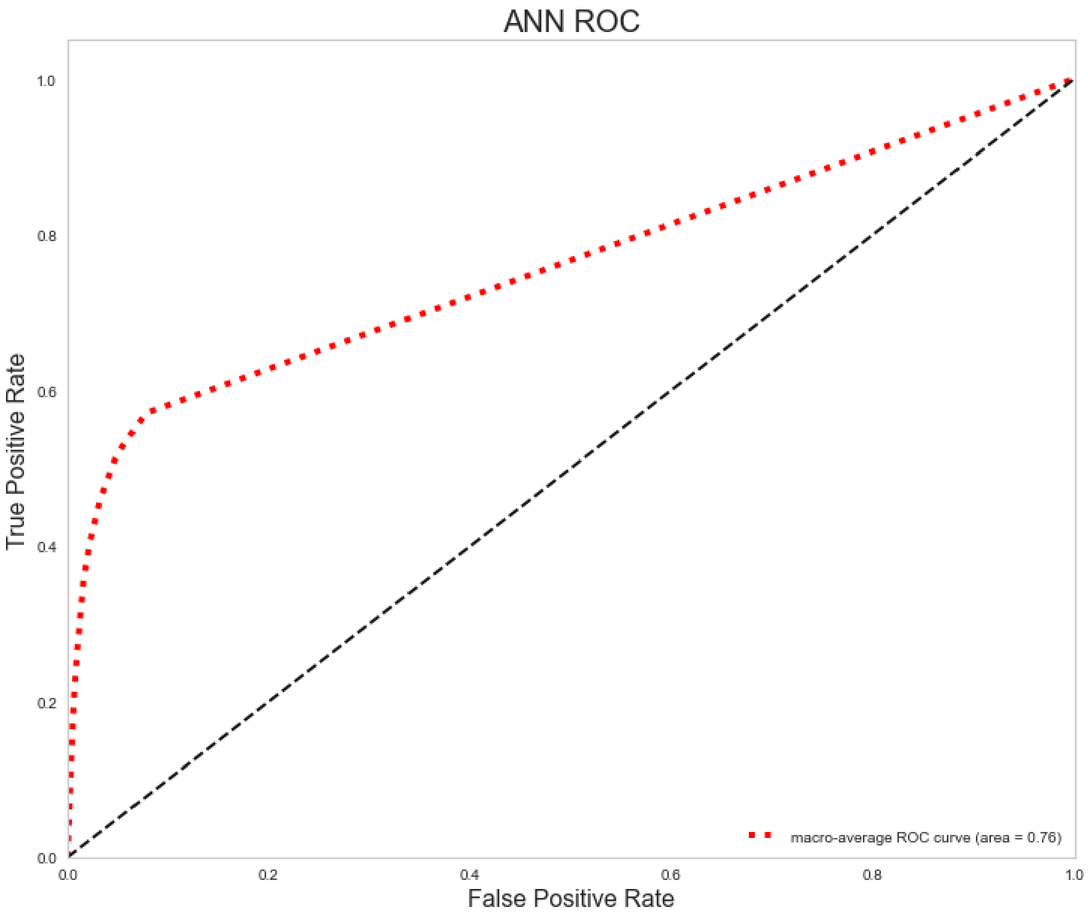


Figure A.6: Neural Network ROC

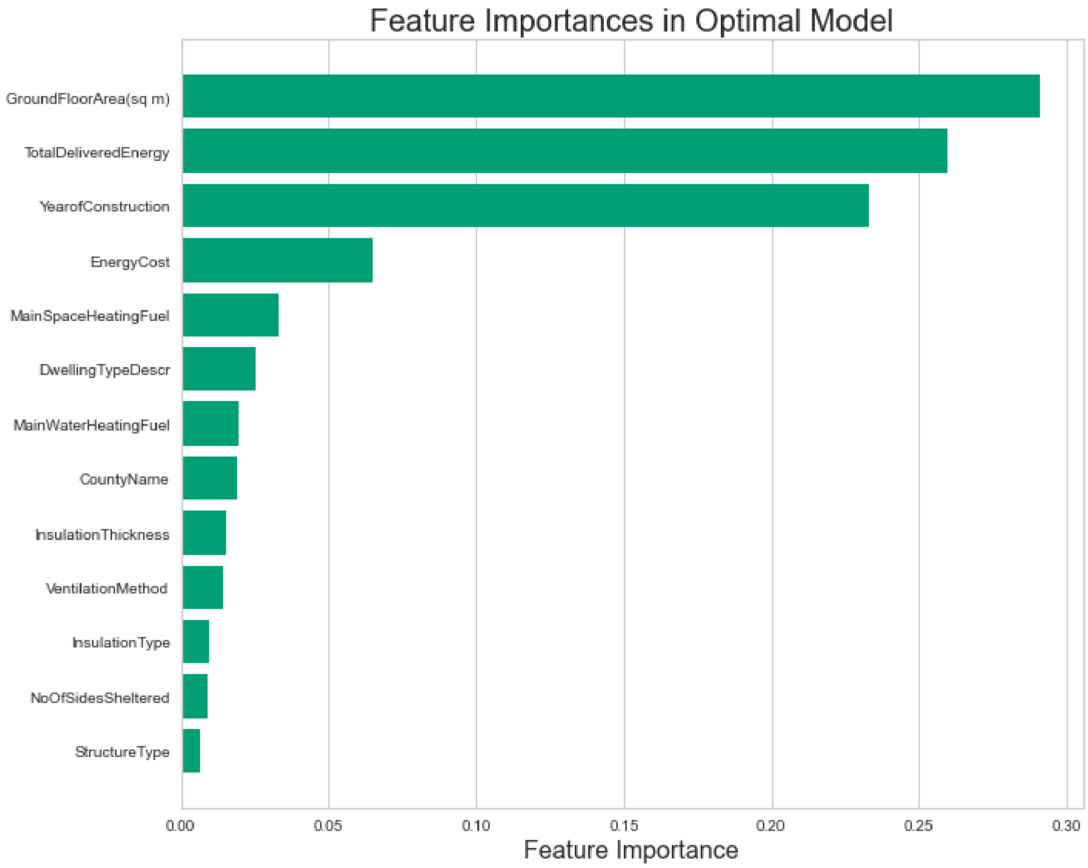


Figure A.7: Decision Tree Feature Importance

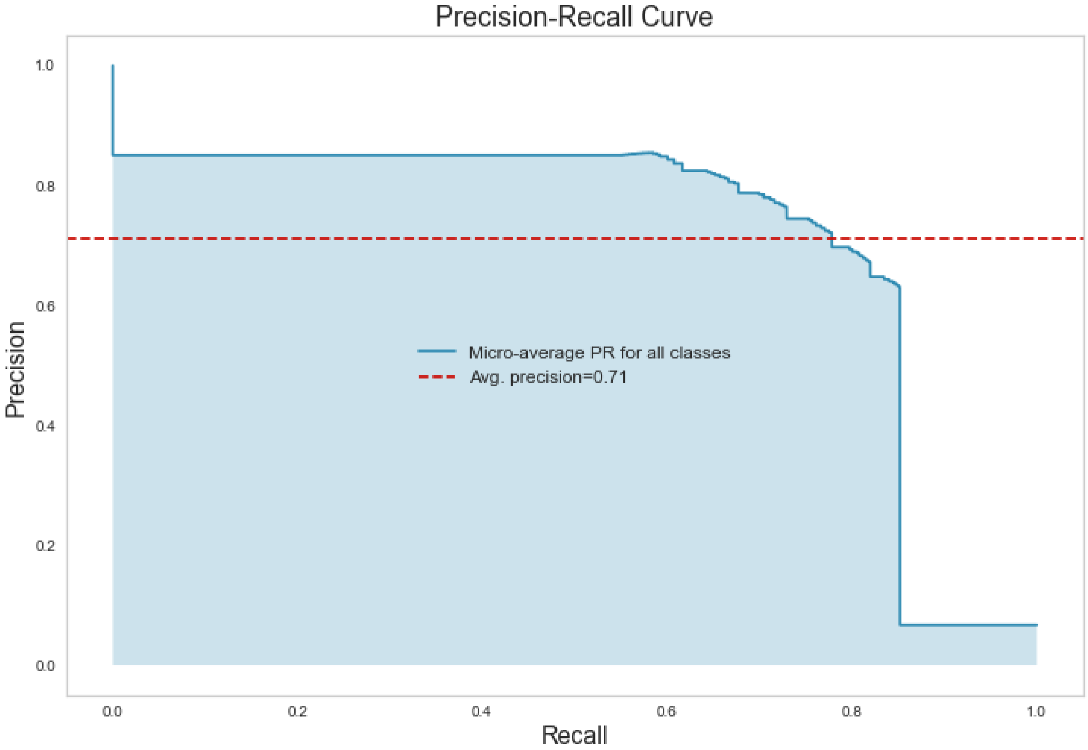


Figure A.8: Decision Tree Precision Recall Curve