**Dissertation**

**Abstract:**

The worldwide widespread of smart meters means more understandable & workable data will be produced for every specific household. In the UK the smart meter bill was passed on the 28th of November 2017 and the implementation of which will lead to raw electricity consumption data being produced which can then be used to measure usage more efficiently (Gov.uk, 2013). This data can be mined and analysed for more than just producing efficient bills and smarter readings. This project intends to provide an achievable, affordable and a minimal change or resource implementation required solution for people by analysing their electricity data and learning specific trends and electricity consumption footprints. Every person has different routine and lifestyles which impact and reflect a different electricity usage trend. Learning those trends can be useful to make a judgement or sense if something is unusual. This will be done by using various data mining techniques both supervised and unsupervised learning. Using as inputs electricity consumption data and active occupancy, to data analysis algorithms such as KNN regression and classification, Linear Regression, Kmeans clustering and Neural Network models in splits of training and testing data to predict the occupancy of the house based on just power consumption. The said algorithms were used for machine learning, prediction and ultimately evaluated on their performance using rigorous scrutinising statistical analysis. Towards the end this research a deduction of the data mining algorithm best suited to the given data set and performing data analysis that closely resemble the proposed situation is presented. All the data mining techniques, listed above, will aim to identify the presence of people at home by processing the electricity consumption data. The dataset used for this model was short term, recorded over a period of 2 months, in the year of 2018, from a home in the city of Coventry. To push limits and stress the nominal requirements of the system, crucial assets like electricity recording time and dataset variables were kept basic and to a minimum to maintain the intended austere nature of the project. The results of this research are predestined to be used for investigating if a deviation from norm or any conspicuous circumstance has been experienced. The norm can be defined as a customized trend for every household. As an extension, the program will choose to notify someone if any unusual electricity consumption behaviour occurs that is a deviation enough from norm to be considered suspicious and anomalous. The furtherment of this research work anticipates laying forth minimalist, resourceful and economical alternative solution to assisted living with the aspiration of helping independent living senior and disabled citizens.

**Introduction (1000 words):**

There is a problem. – The first step towards any solution is to first recognise there’s a problem.

People living on their own – old – can’t afford nurses and luxuries – can’t afford care homes – don’t want care homes – percentage of people living on their own – their average income – disabilities with people – stats of unfortunate events – no one notified until days later –

This dissertation talks about and compares different data mining algorithms based on their ability to correctly predict the number of occupants of a household or simply just the presence of people using the homes electricity usage data. It comments on their accuracy and precision and concludes what specific algorithm best translates itself within the given dataset and type of computational environment. The findings are furthered by providing an application area where seniors or disabled residents can benefit by the implementation of such a system and continue living independently and abdicate assisted living or other equivalent unreasonably priced solutions to the challenges that come with age, disabilities and living by one’s self.

In a research survey carried out by the NHS it was revealed that around one in three adults over the age of 65 are susceptible to fall over at least once a year (NHS, 2015). Around half of whom are likely to fall more frequently. Another similar extended research concluded that two thirds of those who have fallen once are more susceptible to fall again in the next 6 months (Senior Health & Wellness Blog, 2018). In the UK alone, the cause of death amongst seniors aged 75 and over is the most common and unfortunate occurrence (Age UK, 2017).

The government and private institutions are trying their best to get a hold on the situation and minimise these forlorn happenings. (1) Care homes, government allotted nurses, private carers and volunteer organisations are all viable solutions, but extortionate and according to a survey, unfavourable to the elderly as they fear losing their independence more than they fear illness and death.

<http://casas.wsu.edu/news/aging-at-home>

An article published by the Washington state university briefly outlined the average cost of assisted living being somewhere around 70,000 dollars per annum which equates to more than £49,000 per year.

There needs to be an affordable solution for seniors and citizens who, not only, want to keep their independence, but also need to be able to afford it, on their subordinate pension and income.

The application and motivation of this research project is aimed at the elderly and disabled people who reside on their own. As mentioned above, problems like getting around the house and doing daily chores etc can become a struggle at their fragile age and accidents often happen.

Many propositions and steps have already been taken to prevent such events from happening but most of the times they require additional equipment like infrared or heat sensors in homes to monitor movement of the residents which, yet again by being expensive, violates the conditions of a well-suited solution for people who want to be able to afford independence.

Electricity usage data is readily available for, if not all then, most households and the required data is being recorded but going to waste. The imminent implementation of smart meters will, more sooner than later, ensure the recording and storage of the consumption data for every household in the United Kingdom. Leaving us with workable and copious amounts of data that can mined for trends and patterns which in turn can be a less expensive, reasonable and economical alternative solution to the said problem.

Every human being on the planet has a contribution, a footprint. A footprint is described as the physical impression left sand or floor by a foot or shoe. There is another, deeper surrogate meaning of the word “Footprint”. It doesn’t relate to the physical aspects of impressions, but the ecological footprint of a human being. It is elementally defined as the demand of humans on nature. This specific footprint, much like fingerprints, varies from human to human. Every one of us leads a different lifestyle, works different jobs, has different hobbies and uses electricity differently. It’s only fair to have a just and customised judgement for a varying variable as such.

The Electricity consumption data of a household collected over a specific period can be analysed to learn trends. These trends, or “electricity footprint” if you may, once learned, can be used to perform additional actions such as prediction, plotting and forecasting using data science and data mining methodologies. This electricity footprint and trends in the data can, in the scope of this research, be compared with everyday usage to then, based on the electricity usage of the minute, predict the availability of a person at home.

Abnormalities in the data can range from between anything like an electricity spike not being shown roughly at around 7pm when the inhabitant uses the kettle for their evening tea. Or the system predicting the person not being home when the resident is in fact home.

When abnormalities as such are visible, the system can choose to notify a user of the curious occurrence and inconsistency in the predicted data.

This is where the application of this research comes in. An affordable solution to independent living amongst the elderly and disabled.

In simple terms, this project aspires to implement different data mining algorithms that are capable of noticing abnormalities and deviations from normal trends, not directly and in a conventional straight-forward method but, by the uncomplicated comparison amongst the predicted data of occupancy against actual occupancy in electricity consumption entries. This will be made possible through the rigorous prospecting of data and uncompromising statistical tests on the algorithms used followed by a conclusion affirming which algorithm comparatively does the job with admirable accuracy.

try and overcome issues such as mobility and getting around the house that come with old age and disabilities and provide a secondary method of alerting emergency services if a problem arises. The only access to the problem being the persons electricity usage data.

There is a reasonable understanding of the problem, as mentioned above, and the motivation, drive and need for such a system. The findings of this research not only partially implement a system that can be furthered to help in a real-world scenario by making use of real-time power readings, but also counsels which machine learning algorithm works best from a handful of chosen and implemented ones.

The incessant need of the project to be implemented with limited resources and yet produce a capable solution dictated many paths in deciding aspects that were imperative to the project. For instance, the implementation of the algorithms. The algorithms were drafted in the programming language Python using Spyder as the Integrated Development Environment (IDE). The machine learning library “Scikit-learn” was used in the implementation of the algorithms, namely, k-NN regression and classification, Linear regression, Linear classification and Neural networks.

**Literature Review (2500 words):**

<https://books.google.co.uk/books?hl=en&lr=&id=7f5bBAAAQBAJ&oi=fnd&pg=PR5&dq=machine+learning&ots=C45E1q7bJp&sig=FC5Iqs94GWx3ZNclmyqYeBEnKZE#v=onepage&q=machine%20learning&f=false>

Machine learning is a vital field in not only computers, but in every day professions such as retail, . In layman terms, it’s the ability to make a computer system learn and work intelligently using statistical techniques that are performed on data. It is believed that it has originated from the evolution of pattern recognition, computational learning theory and artificial intelligence. Machine learning is the technological modification and automation of a previously unreachable problem. Calculations and trends, patterns and behaviour of data that was once innumerable and computable by the naked eye, was made possible by astute algorithms that learn by reaching into depths and relations between seemingly one-dimensional data to extract meaningful results and outputs that prove to be extremely helpful in various application areas.

Data is constantly being produced and consumed.

**Supervised and unsupervised learning**

<https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/>

Supervised learning is a machine learning approach which involves two separate attribute sets of data, an input set and a derivate output set. The job of this type of learning technique is to reckon the mapping between the input and output pairs. The aim for this approach is to learn and acquire this “mapping function” with such success that the system is then equipped to generate a probable output for a given input instance from the same data set. Examples of supervised learning methods include regression and classification algorithms.

Unsupervised learning, just by then name alone, displays a contradictory nature to its dissimilar sibling, the supervised learning technique. This approach, unlike supervised learning, considers input data that has no reciprocal or corresponding output data.

The goal of this technique is to deduce and extract features from the input unlabelled data. Its structure and elemental patterns and trends are learnt. This methodology is often referred to as “exploratory data analysis”. Due to the one-dimensional personality of this practice and the unavailability of a reference or labelled desired output, there is no way of telling if the results and findings from an unsupervised learning algorithm are, in fact, correct.

Its use is limited to areas where patterns, progressions and tendencies of results are to be analysed, predicted or foreseen.

Examples of unsupervised learning include Clustering, anomaly detection and some branches of neural networks.

<https://books.google.co.uk/books?id=Np9SDQAAQBAJ&printsec=frontcover&dq=machine+learning&hl=en&sa=X&ved=0ahUKEwi3ssmVvMXaAhVCK1AKHQNIDe4Q6wEIKjAA#v=onepage&q=supervised&f=false>

There is plethora of research work that contributes to the area of computer science that deals with data mining and analysis, but only a handful of work that directly associates with my research could be found. The chosen literature’s, much like this research work, adapt different methods of data analysis to interpret and analyse the fitting of their work in a specific application. They also exhibit similarities ranging from the data collection methodologies to statistical analysis of algorithms.

<https://irjet.net/archives/V5/i3/IRJET-V5I3579.pdf>

The unpopularity, lack of full fledge implementation and availability of smart meters has forced many to take it upon themselves to create a homemade smart-meter. A paper published by the International Research Journal of Engineering and Technology (IRJET) implemented such a system using a microcontroller based design and implementation of an energy meter primarily to replace human interference in meter readings which makes the collections susceptible to error hence having an adverse effect on the economy. In addition to that, services such as monitoring energy consumption from a web page through device IP address and electricity theft detection by monitoring meter tampering were also made available. Due to these applications, a homemade smart meter was built based on the Internet of Things (IOT) technology in collaboration with existing electricity meters. The physical aspects of the sysem included an Arduino Nano Board, ESP 8266 Wi-Fi module (Node MCU), 16\*2 LCD display, buzzer, power supply and The Allegro® ACS712 for current sensing. The software aspects included the webpage. The webpage was used to display load energy usage reading in terms of Watts and present device information in a more detailed analysis in both description and visualization.

Coming towards algorithm based related work, a research extract from a book, that consisted of 29 contributions from researches all thoroughly reviewed and validated, analysed electricity data of a household to extract its specific trends which were then, along with the number of inhabitants of the household, used as input variables to a fuzzy model, to forecast how probable it was for a house apparatus to be used or started within the next 60 seconds (Zhu, 2016). The electricity data set collection for this research was long term and collected over a period of 12 months. A case-study on fuzzy tools along with neural network based algorithm for efficient energy management was presented. That was written to predict the electricity usage 24 hours ahead of its time. Lastly, a cost analysis was presented to signify the importance of energy management and its implications. The findings of this specific research in this book, heavily relates to the “Electricity data analysis” area of this dissertation.

Subsequently, a research specifically relating to electricity and anomaly detection with extensive results in the form of figures, diagrams and tables carried out data analysis on electricity consumption data using 2 different approaches namely statistical approach and clustering (Jakkula and Cook, 2010). The two methods were used with the aim of mining the electricity data and count the outliers using both techniques and compare them against the known outliers to see the efficiency of either methodologies. The dataset for this research was power consumption collected using a smart CASAS (Centre for Advanced Studies in Adaptive Systems) environment. It concluded that the 2 outlier detection mechanisms presented wrongly identified an outlier with a ratio of 1:43 in Clustering to Statistical approach. In layman terms, this means the possibility of wrongly identifying an outlier in clustering as opposed to statistical analysis was 1 in 43. The results and finding of this research paper showed that clustering would be inaccurate in its given task 2.3% of the times whereas Statistical approach would wrongly identify an outlier 97% of the time. The application of this research was, like in the previous related work researches, to ultimately conserve energy by identifying abnormal power consumption and hence taking a step towards energy efficiency and conservation, predict abnormalities earlier on, like in the case of too much power being consumed in a smart home the possibility to warn residents on the failure of appliances due etc.

<https://pdfs.semanticscholar.org/b9ba/4b0cfa5cf78d8d4f68da1882f2bb12489c7f.pdf>

Another similar research worked towards forecasting electricity demand in the UK. This research made use of existing deterministic calendar information of weekdays, weekends, bank and national holidays to create a data set of binary data, to distinguish between weekdays and weekends, In collaboration with a KNN- regression model for load forecast and prediction for the upcoming 24 hours. The existing calendar data was used as a base to compare input data against to sort what kind of day the load was being forecasted for (weekday, bank holiday, or weekends). The dataset used for this research dated from November 2001 to November 2008. Almost 8 years with values of power on an hourly basis making that up to be more than 66,500 instances. The results of this research were compared to existing benchmark models (Not created in the research) to the written Multivariate k-Nearest Neighbour Regression model. It concluded that amongst other benchmarks models. The method of choosing best algorithm being mean absolute percentage deviation value. The multivariate system, created within the research took the crown amongst other contenders including univariate k-NN regression model and other statistical benchmark models such as MOVAV (7)24, MOVAV (7)168, RW24 and RW168. This research coupled with the Clustering and statistical approach comparison research paper have made it credible that a benchmark methodology or linear statistical approach is not a strong method to go by when it comes to electricity data. The extensive and exhaustive and variating nature of the electricity data makes it rather impossible to fuse well and it takes complex much intelligent algorithms to learn the ways and motifs.

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.727.486&rep=rep1&type=pdf>

Neural networks happen to be immensely popular with electricity data in arguably one application. Electricity price predictions. The complex multiple layer model and nature of the artificial neural networks makes it possible to work with much convoluted scenarios with more than one variable and attributes. A comparable research that made use of a 3 layered ANN model forecasting price, based on the electricity usage data, for the next 168 hours. This research, however, used the Levenberg-Marquardt algorithm for training purposes. The accuracy and efficiency of the systems predictions were computed using a real-world case study established on the electricity market.

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.193.1235&rep=rep1&type=pdf>

A comprehensive research, in the field of Neural Networks, read and compared published papers dating from years 1990-1996 and 2000-2005 listing 5 areas having attracted most attention amidst which was Load forecasting. A reasonable and in-demand application of electricity data seems to be load forecasting and outlier detection in order to battle energy inefficiencies. Even load forecasting can further be broken down into three sub categories of prediction areas based on periods of forecasts: Long Term (LTLF), Mid Term, and Short-Term Load forecasting. The forecasts lasting anywhere 5 – 20 years for long term, a month to 5 years and short term being defined as anywhere between an hourly to weekly forecasting system. All three of which have different applications and scenarios that call for different forecasting periods.

The conclusion of this research is conflicting yet informative. Despite the many benefits of using Neural networks for power based training and predictions like its ability to deal with stochastic pattern of variating increasing data and fast classification and implied filtering of results, it also stated how and when it would be discouraged to use ANN model. Problems such as training time, training vector and “noisy” data (data containing errors such as missing elements due to faults in recording etc) can cause complications in the algorithm. It also stated that such an algorithm should be used as an “additional tool” not as a “replacement” to other AI techniques.

Anomaly detection is another area that hugely relates to this project. The application of this dissertation is to detect, effectively, occurrences and instances of data that can be considered as anomalous and swiftly acting upon those said occurrences. A survey-based journal article spoke in depth about the different kinds of outliers in data analysis and the challenges faced in isolating a given anomaly from normal regions in a traditional 2-dimensional dataset (Chandola, Banerjee and Kumar, 2009). This survey particularly proved helpful in narrowing down concepts and techniques most effective in successful outlier isolation and detection. In addition to that, it also spoke about techniques for Anomaly detection such as Neural networks, rule based, Bayesian Network, Support vector, Clustering and Nearest neighbour based techniques etc. Most of the mentioned techniques in this survey have been used for this dissertation for not outlier detection, but prediction based on a training dataset.

Another research relating to anomaly detection proved to be more practical than survey based and aimed to detect abnormalities between an aged and a new transformer in the UK based on data and variables from the transformers such as temperature, vibration, moisture, load current etc in the 2 transformers (Catterson, McArthur and Moss, 2010). The data readings were calculated using sensors. This paper concluded that the Conditional Anomaly Detection (CAD) technique can be used in the online monitoring of the transformers.

<https://pdfs.semanticscholar.org/ff0b/d954443338708279f97feb05d6b29e41382c.pdf>

<https://link.springer.com/chapter/10.1007/11941439_114#citeas>

The final deliverable of this research, in addition to implementation of assorted predictive machine learning algorithms, covers a section for statistical analysis and comparison of the enforced algorithms based on their predictive performance and success using various arithmetic formulary. However, it primarily relied on F Score to calculate, prove and thereupon crown one algorithm victor.

One such research study, published by the Australasian Joint Conference on Artificial Intelligence, discussed evaluation measures for different characteristics of machine learning algorithms. It speaks of commonly-accepted performance measuring techniques and touches base with and proposes new unconventional approaches for measures derived from “fault avoidance” such as Youden’s index γ, the likelihood values ρ−, ρ+, and Discriminant Power DP. It is argued in this paper that performance measures other than accuracy, F-score, precision, recall or ROC do apply and can be beneficial.

Although the work Ms. Sokolova was aimed towards providing an unprecedented approach to algorithm measure, the areas of her research that resonated with this research work was the discussion of conventional measures such as precision, recall, sensitivity, specificity, accuracy and F measure. There is substantial evidenced work done using these statistical methodologies and a “fault avoidance” approach was unbefitting.

**The research rudimentarily defined the measures as follows (**For extensive and apprehensible definitions with corresponding formulae refer to the methodology section**):**

* **Receiver Operating Characteristic (ROC):**

Shows a relation between the sensitivity and the specificity of the algorithm

* **Accuracy:**

Approximation of the effectiveness the algorithm displays by showing the probability of the true value of the class label; in other words, it assesses the overall effectiveness of the algorithm;

* **Specificity:**

Specificity approximates the probability of the negative label being true; in other words, it assesses the effectiveness of the algorithm on a single class;

* **Sensitivity:**

Sensitivity approximates the probability of the positive label being true; in other words, it assesses the effectiveness of the algorithm on a single class;

* **F score:**

A composite measure which benefits algorithms with higher sensitivity and challenges algorithms with higher specificity.

For said measures, quality of classification is measured and built from a confusion matrix which records correctly and incorrectly recognized examples for each class. These classes are determined by the nature of the prediction.

A prediction instance can be classified as one of the following:

* **True positives (TP)**

The algorithm correctly classifies the subject of prediction in a binary environment, as ‘1’

* **False Positives (FP)**

The algorithm incorrectly classifies the subject of prediction in a binary environment, as ‘1’

* **True Negatives (TN)**

The algorithm correctly classifies the subject of prediction in a binary environment, as ‘0’

* **False Negatives (FN)**

The algorithm incorrectly classifies the subject of prediction in a binary environment, as ‘0’

The confusion matrix below outlines the elemental method of categorization and determination of the prediction type based on the above-mentioned counts.

|  |  |  |
| --- | --- | --- |
|  | **Recognised Positive** | **Recognised negative** |
| **Class Positive** | **TP** | **FN** |
| **Class Negative** | **FP** | **TN** |

Most of the existing work done in relation to this project takes things as far as analysing electricity usage data to extract trends, test different data analytical techniques and anomaly detection. The major application of those finding being the need to make electricity consumption more efficient and conserve energy where possible. The Anomaly Survey research discussed entirely different application areas of anomaly detection but none of them came close to the intended work and application of this project which makes me confident that the deliverable is new and hence will contribute to a novel application of occupancy and outlier detection.

The Multivariant k-NN regression model research showed sweeping results in the success of a nearest neighbour regression model in the forecasting of load data which stands to show that a k-NN regression model might be a viable technique for triumphant results while working with electricity consumption data.

This project intends to take things in a different direction, it aims to predict the occupancy of a household in not just a Boolean value, but number of occupants as well. The intended application of this project is to identify an anomaly in data and alert emergency services to get the independent senior or disabled resident the help they need. The abnormality detection will root as an extension to the proposed occupancy predicative system. If the system produces a false negative, that is, no inhabitant is predicted against the known data telling us that there is in fact someone home, it’s clear that something is wrong. The proposed application of this project successfully stands out and is derived from personal motivation and empathy towards the human race.

Despite there being many such algorithms capable of successful prediction of occupancy based on electricity data, much like load forecasting, not a lot of research is done in this domain of predicting occupancy based on electricity consumption let alone a statistical comparison of such algorithms and the intended application of this project.

Most of the existing work seems to be aimed at crowning technology victor against the existing linear benchmark and statistical methods. It is a given that an expert learnable algorithm will perform well than a system with a linear standard point of reference. Some might even say there is no comparison there. But even amongst the various learnable algorithms there must be one that performs better than the other. In addition to the many ways listed above, this specific reason of a rigorous statistical analysis and comparison between different algorithms is, also, what makes this dissertation project novel and different from its related existing work.

**Methodology (1500 words):**

What exactly you are doing about the problem in question? Why and how is it different and novel in comparison to everything in the literature review above?

This section of the research focuses on the methods applied and implemented to set the objectives into motion. The dataset, algorithms, testing and statistical analysis methodologies implemented.

**Dataset:**

WHY HAVE A DATASET IN THE FIRST PLACE?

Landing on a dataset for this project was challenging since there are numerous open-source and public data sets available online. It was important to keep in mind that the goal of this project was to provide a solution to the problem that had minimal pre-requisites and requirements for it to be implemented. This helped narrow down the search. Naturally, datasets that had too many additional variables that had to be recorded manually or using sensors, didn’t fit the application of this project. If the resident where to install sensors, it threw the existence of this project into question as the sensors being installed themselves could do the job of recording or picking up any unusual activity.

One such dataset that was taken into consideration was an open-source Electricity consumption dataset of 19 households in London over the course of 2 years. The data set consisted of separate csv files for household information, half-hourly electricity consumption, daily electricity consumption, acorn details, half hourly weather and daily weather. These separate 6 csv files contained information such as tariff, what block the house is in, its acorn group, a smart meter measurement of power on a half hourly and daily basis, power measures, minimum, maximum, mean, median, sum and std. Details of the acorn group, their profile and the people in the group. In addition to that it had 2 other files containing weather readings, again, on a half hourly and a daily basis.

Despite the data being publicly available, clean, workable and with the added benefit of having data from 19 different households and exhaustive readings, I didn’t want to use the data set as the project aimed to display no threat or necessity of peoples personal and sensitive information. Under no circumstance did I want people to believe that minimum requirement for the application and implementation of a system as such requires innumerable readings and a convoluted combination of attributes. I wanted to keep it unambiguous for people yet competent in its goal.

The dataset that was ultimately chosen for the project was not a publicly available one. It was a personal household electricity dataset that was recorded and belonged to my supervisor.

The dataset was a singular csv file containing three attributes, namely, Date and time, Power and “at home”. The power was calculated and recorded for every minute of each over the period of 2 months (January and February 2018) adding up to 37270 instances.

Since there is a reading for every minute of each day, there are roughly 1448 occurrences of the attributes for a single day. This makes the data highly exhaustive and broken down hence very specific and precise.

|  |  |
| --- | --- |
| Attribute | Description |
| datetime | The date and time |
| power | The number of watts consumed in kw/h |
| unit | An additional field added to serialise the datetime variable to represent minutes. |
| athome | A variable denoting the presence and availability of people inside the household with 0, 1 and 2 (indicating number of people) |

The benefits of this data set were the additional variable “athome” which can be used to and linked to patterns and behaviours in the electricity usage and how the power goes up or down based on occupancy, which will make the system more learnable and proficient.

**Data collection Methodology:**

Given that the use of smart meters is not popular in every household thus far, this specific dataset was collected using manual techniques which, soon enough in the future, won’t be necessary once smart meters installation is enforced by law.

A homemade smart meter was created for the effective collection of the electricity data which included the following components:

**Node MCU:** <http://ijireeice.com/upload/2017/november-17/IJIREEICE%204.pdf>  
A Lua (scripting language) based ESP8266-E12 Wi-Fi SOC (System on Chip) module used commonly for the implementation of internet of things. It has 16 GPIO (General Purpose Input Output) and works on 3.3v. The term "NodeMCU" by default refers to the firmware rather than the dev kits. The firmware uses the Lua scripting language

**Raspberry Pi:** <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>“An inexpensive credit card-sized computer that plugs into a computer monitor or TV, uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It’s capable of doing everything you’d expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.”

**SQL Lite Database (For storage of electricity data):**

**Light Sensor:**

simple Nodemcu based circuit consisting of a power supply and a light sensor was used to calculate the power consumption. The sensor would register the number of times the light on the electricity meter would blink. It was known that the meter light would blink once 10watts of usage was noted. Hence the number of blink registers were multiplied with 10 to get the power used per minute.

The “athome” variable was calculated using “pings”. Defined as “Query (another computer on a network) to determine whether there is a connection to it” (CITE - <https://en.oxforddictionaries.com/definition/ping>) which would administer how many specified mobile phones were connected to the internet at a given time. To make this calculation more accurate the use of push notifications was also used.

This dataset was chosen because of its simple precise nature. It consisted of limited but necessary variables. Another reason of preference for this this dataset was its temporal proximity. The lifestyles of people today largely differ from the way of living of people as recent as 5-10 years ago. Factors such as weather, routine, job, lifestyle, activity, insulation and occupancy etc are all factors that influence the consumption of electricity. They are also factors that tend to differ which makes older datasets, if not outdated then at least, irrelevant. Leaving recent data being more relatable and accurate in its ability to determine anomalies in the modern-day electricity consumption patterns.

**Method:**

**Types of algorithms**

**Algorithms:**

The chosen algorithms that were implemented in this research are listed below:

* k-NN Regression
* Linear Regression
* k-NN Classification
* Artificial Neural Network Classification

There was a large pool of algorithms to choose from whilst deciding for a reasonable approach to the proposed problem but the intended statistical analysis and comparison of the algorithms highly influenced their reduction.

**Tools:**

* **Programming Language:** Python 3.6
* **Integrated Development Environment (IDE):** Spyder 3.2.6

Spyder is an open source IDE designed to solemnly work with Python. It enhances the language’s ability given its many beneficial characteristics like syntax highlighting, debugging support, variable exploration, GUI based editor for most data structures, step-by-step execution, ability to run parts of code at a time, code prediction and completion, built-in editor and various other productivity tools. Its IPython (Qt) console makes for an able prompt data visualisation and visual communication area. in the form of graphs, tables and the display of regular outputs. CITE - <https://books.google.co.uk/books?hl=en&lr=&id=N1InCgAAQBAJ&oi=fnd&pg=PP3&dq=spyder+development+environment+python&ots=9GL4id3oTs&sig=JW7RFBACriK53R_dzatWwRIZZto#v=onepage&q=spyder%20development%20environment%20python&f=false>

Like most IDE’s for python, it tackles the linear nature and inflexibility of the regular Python programme interface and provides a more participating candidate as opposed to the stand-alone Python programme.

The IDE “Rodeo” was initially considered, but it was found to be inflexible and caused conflicts in the existing programmes. The library import function was unable to detect to the location of the module where they were downloaded.

<http://www.southampton.ac.uk/~fangohr/blog/installation-of-python-spyder-numpy-sympy-scipy-pytest-matplotlib-via-anaconda.html#what-is-what-python-python-packages-spyder-anaconda>

* **Dataset (.csv file):** Microsoft Excel
* **Stats computation:** Manual

**Essential Libraries:** <https://www.frontiersin.org/articles/10.3389/fninf.2014.00014/full>

* **Scikit learn**

Scikit-learn is a Python module that includes a wide range of state-of-the-art machine learning algorithms for supervised and unsupervised problems. This package emphasises ease of use, performance and documentation while focusing on bringing machine learning to non-experts using a general-purpose high-level language. It has minimal dependencies and is distributed under the simplified BSD license, encouraging its use in both academic and commercial settings. <https://hal.inria.fr/hal-00650905v1/document>

* **Numpy**

In the Python world, NumPy arrays are the standard representation for numerical data and enable efficient implementation of numerical computations in a high-level language. Performance of computations are improved through techniques such as vectorizing calculations, avoiding copying data in memory, and minimizing operation counts.

<https://ieeexplore.ieee.org/document/5725236/>

* **Pandas**

An open source, BSD-licensed library for the Python Language which provides rich data structures and functions which makes working with structured data fast, east and expressive. It enables Python to be a powerful and productive data analysis environment.

<https://books.google.co.uk/books?hl=en&lr=&id=v3n4_AK8vu0C&oi=fnd&pg=PR3&dq=python+pandas&ots=rgFI3qxxqA&sig=5c6fMIuXR_1PqxahUdn_MDPfLDE#v=onepage&q=pandas&f=false>

* **Matplotlib**

A 2D graphics package used by Python for application development, interactive scripting and publication-quality image generation across user interfaces and operating systems.

<https://ieeexplore.ieee.org/document/4160265/>

**Algorithm Implementation**

**k-NN Regression and Classifier model:**

Two of the implemented algorithms in this research are two different renditions of k Nearest Neighbour algorithms. Both use the same groundwork when it comes to the analysis and prediction of data. Both algorithms work in such a way that they store all available cases from the given dataset (Training dataset) and predict a target input dataset (testing dataset) based on a similarity function. The similarity function used in this research was distance. The size of ‘k’ (nearest neighbour to be taken into consideration) in both the models was assigned the value of 3 and kept constant for both the algorithms to maintain a fair equilibrium.

In a nearest neighbour algorithm, the nearest neighbours are calculated using a distance function. Amongst the most popular distance functions are Euclidean, Manhattan and Minkowski.

There is a controversy when it comes to choosing an optimum value for the number of nearest neighbours. A research carried out by the Mu’tah University – Jordan tried out various values of n ranging from 1 to the square root of the sum of samples in the training dataset and proposed a solution to the selection of the value of ‘k’ depending on ensemble learning (CITE - <https://arxiv.org/ftp/arxiv/papers/1409/1409.0919.pdf>).

An archaic yet substantial research paper proposed that “large values of K smoothen the results of classification” (CITE - <http://cs.du.edu/~mitchell/mario_books/Neural_Networks_for_Pattern_Recognition_-_Christopher_Bishop.pdf>). With insufficient evidence as to why, but for it is also considered that the square root of the sum of samples in the training dataset is an optimal assigned value for ‘k’.

All in all, it makes sense that a small value for ‘k’ might be too influenced by individual cases whereas on the other hand, larger values may include too much of a deviation and complexity in a non-complex situation ultimately producing a prediction that is in either case not the most reliable and accurate as it can be.

**Accuracy Percentage Formula:**

Drawing from experience, it was observed that highest accuracy percentage was achieved when k was kept to a small odd digit.

**The accuracy variation for different values of k as follows (based on both distance and uniform):**

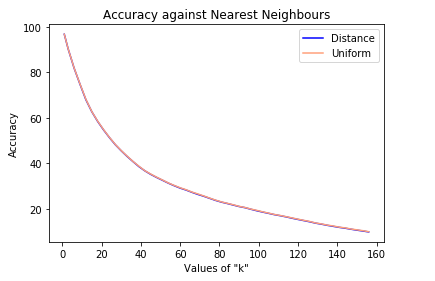
|  |  |  |  |
| --- | --- | --- | --- |
| **Values of ‘k’** | **Accuracy in result (%)** | **Precision score** | **Mean Square Error** |
| 1 | 96.75 | 0.968 | 0.0345 |
| 3 | 90.21 | 0.966 | 0.0246 |
| 9 | 74.77 | 0.964 | 0.0287 |
| 99 | 19.18 | 0.938 | 0.0571 |
| 156 - Square root of training data set | 9.87 | 0.930 | 0.0662 |

Code was adapted to run the k-NN regression algorithm for a total of 53 different Values for “k” ranging between 1 and 156 (Square root of training data set) with increments of 3. The code would, as a result, calculate the precision of the predicted data, accuracy of the result based on correct and incorrect predictions and mean square error for every Value of “k”.

The values of “k” and their resulting accuracies, precision and Mean Square Error (MSE) values were stored in lists.

The said data was plotted in graph format to visually display the findings for a superior understanding of the effect of the values of “k”:

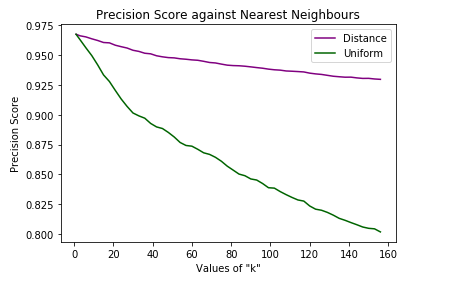
**The Graph for Values of “k” against Accuracies (Based on distance and uniform) is as follows:**



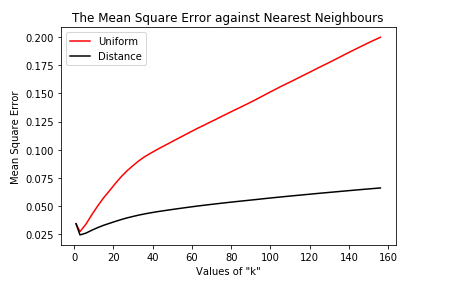
There is a visible decrease in the accuracy of prediction with increasing members taken into account during the categorization and placement of a test dataset instance.

Although the highest percentage of accuracy visibly being achieved at K = 1, the realisation that a number so small may and is capable of being influenced by noise and hence increasing the probability of inaccuracy incorrect prediction, it was decided to find middle ground and settle for K = 3. Because it seemed like a reasonable number and it seemed to be better suited to the given dataset.

**The Graph for Values of “k” against Precision score (Based on distance and uniform) is as follows:**



**The Graph for Values of “k” against Mean Square Error (Based on distance and uniform) is as follows:**



Despite the many extensive research that has been carried out for this sole purpose of figuring out a suitable value for the number of nearest neighbours taken into consideration, during the course of this research through personal experience it was felt that the value of ‘k’ is dependent on the type of dataset and shouldn’t be merged with existing notions of it working best when it is equal to the square root of the training dataset or some constant number. There is a reason that the value of nearest neighbours is not predefined and remains flexible as different types of datasets might call for different considerations. And the random assigning of different values for it reveals which one works best and gives a acceptable result.

However, there is obvious sense in the fact that it shouldn’t be too small, but then again there is a pattern that shows that keeping it too large can cause deviations and the majority vote can get influenced given the data, being worked with, is noisy. Having said that, increasing the value of K also saw a visible constant being reached, although the overall accuracy of the system came down arguably significantly, at a point increasing the value of ‘k’ would produce accuracy very close to that for a closer k value and the difference would not be substantial. This showed that whilst accuracy was being compromised, a large value for the number of nearest neighbours would also make the system less prone to be persuaded by noise and irregularities in data.

**Difference between the regression and classifier model:**

The basic difference between the said algorithm is that the outputs and behaviour of the algorithms. The regression model finds the similarity of the input testing dataset to predict an attribute. The attribute prediction is “measured” based on the similarity function. In the regression model the output is an actual value. The value is calculated based on the average of its given nearest neighbours.

The classifier model, as the name implies, “classifies” or “categorizes” its predicted output in a defined set of possible outputs based on the similarity function. The output for a classification algorithm takes the value of the nearest neighbour most popular in quantity and in majority. This is known as majority vote.

**Example:**

**Classification method:**

For instance, in the case of classification, in a k-NN algorithm with a defined training dataset of 10 instances and 4 possible outputs: either ‘Blue’, ‘Red’ , ‘Green’ or ‘Yellow’, If the family of a testing instance (Bigger, black dot) needs to be determined in a situation where k = 3, It is visible, in the example plot below, that out of the 3 closest points on the graph, 2 are green and one is red hence the testing instance will be “classified” into the “Green” family.

**Regression Method:**

When it comes to regression, it becomes a little more complex than that. Regression will also produce a result that ultimately crowns the testing instance to belong to the green family but how does it do it. First the algorithm will assign the 3 possible instances (Blue, green, yellow and red) numeric values:

|  |  |  |
| --- | --- | --- |
| **Colour** | | **Assigned Numeric Value** |
| Blue |  | 1 |
| Red |  | 2 |
| Green |  | 3 |
| Yellow |  | 4 |
| Black |  | To be predicted |

Given that k = 3, and that as visible in the graph out of the 3 closest points are 2 green and a red dot the algorithm will place the testing instance using the average of the nearest neighbours using their numeric values.

Nearest neighbours = [Green, Green, Red]

Numeric Values = [3, 3, 2]

Testing instance:

The produced value is closest to the numeric Value 3 and hence the testing instance is placed in the green family.

Although the “numeric” and actual execution of the algorithms are far more complex and outcomes are assigned values rather the similarity function uses values based on the Euclidean distance of the testing instance from the training data and computes its illegibility in every possible family.

**Actual case:**

Keeping in mind the example for the nearest neighbour models above, in the case of the electricity dataset being used for this project, the training and input data set being ‘kwh’ (power) and ‘athome’ (Number of occupants of the household), the possible classes for classification for output data becomes 0, 1 and 2 from the “athome” variable. Based on the similarity function, the algorithm ‘classifies’ every instance of the testing dataset into the three possible classes.

The regression model, calculates the average of the 3 closest neighbours to the testing instances and produces a float value which can be rounded off to produce a prediction of the class the testing instance might belong to.

ELABORATE^^^^

**Linear Regression model:**

**How it works. How does it determine nearest neighbour? Euclidean distance. Implementation, pseudocode.**

**Neural networks:**

**Classification and regression**

**Refining results a huge part of this**

**Gradient descent back propagation huge part**

**Initial vs final results**

**Pseudocode, implementation problems encountered**

<https://pdfs.semanticscholar.org/ef87/199c014a18b8b58e05822c5c05ae743447f4.pdf>

**Statistical Analysis Methodology:**

There are many procedures at hand to determine the success of algorithms in reaching a given target. The algorithms can be tested on their ability using markers such as speed of computability, solution quality, simulation, run time, errors encountered, optimisation etc. Many of these units are computable within the scope of code and the language being worked with. With an import of a library and the administration of parameters to a function and running the said code, these units of measure can be determined with the click of a button.

Such methods of algorithm analysis can be computed in one line of code. They are in most cases considered sufficient but it can be argued that they tend to be incomprehensive and in some cases, rendered meagre. In a professional environment however, there is a call for a far more encyclopaedic analysis especially in areas where a heavy part of research is formulated of a comparative statistical analysis.

Upon meeting with statistical expertise for mentoring regarding the most viable possible method that could be devoted to the investigation, Chi squared error was initially considered but on further description of the nature of the system and its outputs, it was rendered unprofitable and precision recall and F score was suggested instead.

Hence, this dissertation makes use of many notable techniques and manual mathematical calculations in collaboration with the results acquired from the algorithms and code to put forward a reasonable and thorough analysis of the algorithms that were worked with.

The statistical techniques and calculations applied to the results from each algorithm are as follows:

* F1 score
* True positive rate
* False negative rate
* True negative rate
* False positive rate
* Prevalence
* Positive predictive value precision
* False discovery rate
* Accuracy
* False omission rate
* Negative Predictive Value
* Positive likelihood ratio
* Negative likelihood ratio

The mentioned calculations make use of computed variables based on the results from the prediction algorithms. The variables used are:

* **True positives (TP)**

The algorithm correctly predicts the presence of a person as positive. Meaning predicting that a person is home when they in fact are.

* **False Positives (FP)**

The algorithm incorrectly predicts the presence of a person as positive. Meaning predicting that a person is home when they in fact aren’t.

* **True Negatives (TN)**

The algorithm correctly predicts the presence of a person as negative. Meaning predicting that a person is not home when they in fact aren’t.

* **False Negatives (FN)**

The algorithm incorrectly predicts the presence of a person as negative. Meaning predicting that a person is not home when they in fact are.

* **Condition Positives (CP)**

The total instances in the data where the person is home, where person occupancy or presence is positive.

* **Condition Negatives (CN)**

The total instances in the data where the person is not home, where person occupancy or presence is negative.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Positive | Negative |
|  |  | Actual data (+)  (Person(s) home) | Actual data (-)  (Person(s) not home) |
| Positive | Predicted Occupancy (+)  (Person(s) home) | **TP** | **FP** |
| Negative | Predicted Occupancy (-)  (Person(s) not home) | **FN** | **TN** |

**Following is a brief explanation and formulas of the above-mentioned methods to test the predicted results:**

**F1 score - https://adamyedidia.files.wordpress.com/2014/11/f\_score.pdf**

Also known as “F score” or “F measure”, this calculation measures the accuracy of a subject. It can be explained as the weighted average between the precision and recall of the subject. Its best possible value is 1, denoting a highly efficient and accurate subject and worst possible outcome is 0, signifying a subject without the possibility of having performed more poorly.

The actual formula based on the definition of F1 score is as follows:

But through methods of substitution, a more direct and applicable formula for it can be derived:

**True positive rate (Sensitivity) https://en.wikipedia.org/wiki/Sensitivity\_and\_specificity**

Also referred to as the sensitivity or “hit rate” of a subject. As the name suggests, measures the proportion of positives identified correctly. Example the proportion of times the algorithm predicted occupancy of the household as positive (Someone home) when there was someone home.

Formula:

Since False Negatives are ultimately all the instances that are true, total positives = True Positives + False Negatives

**False positive rate https://en.wikipedia.org/wiki/False\_positive\_rate**

Is defined as the proportion of negatives that return positive outcomes. The probability of a positive prediction result produced given that the condition being looked for is not present (negative). the expectancy of the false positive ratio. It is calculated as the ratio between number of negative events incorrectly classified as positive to the total number of negative events.

Formula:

Since False Positives are ultimately all the instances that are negatives, total negatives = True Negatives + False Positives

**False negative rate https://en.wikipedia.org/wiki/False\_positives\_and\_false\_negatives#false\_negative\_rate**

Clearly complimentary to “False Positive Rate”. Defined as the proportion of positives that return negative outcomes. The probability of a negative prediction result produced given that the condition being looked for is present (positive).

Formula:

Since False Negatives are ultimately all the instances that are true, total positives = True Positives + False Negatives

**True negative rate (Specificity) https://en.wikipedia.org/wiki/Sensitivity\_and\_specificity**

Also referred to as the specificity or “miss rate” of a subject. As the name suggests, measures the proportion of negatives identified correctly. Example the proportion of times the algorithm predicted occupancy of the household as negative (No one home) when there was no one home.

Formula:

Since False Positives are ultimately all the instances that are negatives, total negatives = True Negatives + False Positives

**Prevalence**

Is defined as the proportion of a specific condition to be true. In our case of household occupancy, it will be defined as the fraction of times when a person is truly home to the total number of readings. Although not a unit of measurement of the accuracy of an algorithm, this measurement will not remain same for every algorithm and only signifies the popularity of occupancy.

Formula:

**Positive Predictive Value (PPV) http://sphweb.bumc.bu.edu/otlt/MPH-Modules/EP/EP713\_Screening/EP713\_Screening5.html**

A second more broken down name for precision which makes the formula more understandable. Defined as the probability that instances where the positive was predicted (person is home) is true.

Formula:

**Negative Predictive Value (NPV) http://sphweb.bumc.bu.edu/otlt/MPH-Modules/EP/EP713\_Screening/EP713\_Screening5.html**

Defined as the probability that instances where the negative was predicted (person is not home) is true.

Formula:

**False discovery rate https://www.mailman.columbia.edu/research/population-health-methods/false-discovery-rate**

Defined as the percentage of false predictions in a given set of predictions.

Formula:

**Accuracy**

Generally defined as the calibre to which the outcome of an assessment, calculation or measurement is precise or correct. In a statistical environment, error is taken into consideration. In order to achieve a high accuracy both precision and correctness of a measurement need to be high.

Formula:

**False omission rate**

False Omission Rate a measure of False Negatives that are incorrectly rejected. It is a compliment of the Negative Predictive Value.

Formula:

**Positive likelihood ratio**

Asdvsav

Formula:

**Negative likelihood ratio**asdvsav

Formula:

**Testing/Findings/Results (500 words):**

Trying to prove my proposed solution works

**Analysis (3000 words):**

Did it work?

**Conclusion (1000 words):**

Hurray!!, My proposed solution worked. Here’s what I could have done to make it better or ways I could have enhanced the solution or added to it.

• What: restate your principal findings.

• Why: The importance of your findings in the academic field.

• How: Explain how your research question/s have been answered by the methods you employed and the evidence you have found.

**Reflection:**

The hardest part about the dissertation was understanding the dissertation itself. The sub-sections such as literature review, methodology, analysis, results and conclusions had straight forward names and explanations that seemed to be even more understandable. We were told what was expected of us in each section and their implementations but it was the actual implementation that was challenging. Related work was to be read and analysed in the literature review but exactly what areas where crucial to discuss, what information had to be in the summary and what was unnecessary. Although the initial understanding was quite clear it wasn’t until I got to typing did I realise I had so many questions and concerns. They were handled and rectified in due time. One very important thing that helped with this would be the early submission of the Project Proposal which had a mini Literature Review, abstract and client motivation which helped getting us in the mindset of understanding what and how the dissertation worked. Although not much from the initial project proposal was made use of in the final project hand in, it was still of immense importance when it came to grasp the idea of the professionalism and formal nature of the report.

Dataset:

Projects that involve data mining often struggle with finding the perfect data set. I was lucky in terms that the I wanted a minimalist data set. An extremely simple one with very little and necessary attributes only. The struggle came in choosing between the two datasets that I narrowed it down to. One dataset was a public one from Kaggle. It included the required data but had other additional data and that too of 30 different households in a neighbourhood. This data was comprehensive yet a bit too intricate for the intentions of my project. Another aspect of this data set was that power reading from the household were taken every 30 minutes which made it undesirable to be worked with given the urgency and demand of my project application.

The other dataset that I chose came from my supervisor’s personal household. It changed the course of my ethics application and some formalities and consent forms had to be fulfilled but the dataset was perfect when it came to the requisites of the project. This data contained 3 attributes in total, datetime, power and a numeric variable to denote the number of people at home. Minimum being no people and a maximum of two people. A significant problem that was faced during the importing and using of data within python was the “datetime” attribute. Since there were power readings for every minute of every day, that meant there were 24 x 60 = 1440 readings for a single day and the data set was collected over a period of 2 months (approximately 60 days). The splitting and usage of the “datetime” attribute was not impossible but laborious. And given a situation where data had to be well broken down, the “datetime” field was left as it is as a reference and another serialised attribute by the name “unit” was added to the dataset. This field was used as a unit of time in minutes to denote the first minute of the 2nd of January 2018 at 15:26 and the 36270th value representing the last minute of the data set at the 23rd of February 2018 at 11:12.

The importance of the occupancy prediction weighed out over the patterns of electricity data usage as per the nature of days and the types of days (for example Mondays, weekdays, weekends etc). It is crucial when you think of it, it is crucial to help refine the system, a false negative works just as well to indicate there is room for concern.

The split for the training and testing data was an additional area that drew some time. It was advised in the end of term presentation feedback that the split wasn’t right down the middle or that the training included all the dates of January and some of February and the testing data of just February rather make the split more random.

The percentage of the split was also a concern and there is debate when it comes to a fair value. Some say an 80:20 is a fair split, whereas in some cases and scenarios even a split of 70:30 is justified. But then again, these split ratios are also highly dependent on the size of the data set.

I personally landed at using a 67:33 training to testing ratio. Given 36210 instances, it left me with a seemingly healthy split of 24300 learnable instances to 11970 predicting/testing instances. The split was also ensured to be random. The python built-in data split had a function of splitting the data randomly, not right down the middle and as shown in the figure below, the first 20 instances of both the testing and the training data was random.

|  |  |
| --- | --- |
| **Training dataset (67%)** | **Testing dataset (33%)** |
|  |  |

\*The numbers on the right-hand side is the additional field that was added to the data set to serialise the headings of the power and ‘athome’ variable readings. Although the first 20 values only aren’t sufficient to display the randomness in the collection and split of the data it is still apparent that the data was not collected serially.

This part of the code is responsible for the split. The split ratio and the randomness of the split. The sklearn library has a function “model\_selection**.**train\_test\_split” which takes in attributes such as the output variables, the size of the testing data and the random state.

X\_train**,** X\_test**,** Y\_train**,** Y\_test **=** sklearn**.**model\_selection**.**train\_test\_split**(**X**,** y**,** test\_size **=** 0.33**,** random\_state **=** 4**)**

Implementation/Algorithms:

The implementation of the machine learning algorithms seemed to be frightening. And it was at first. There was a learning curve for the understanding of how they work. Small things such the training set, the testing set, supervised and unsupervised learning, different types and techniques to do the same thing. The availability of so many different techniques is what encouraged me to try out different ones and statistically analyse which one worked best. The coding and actual implementation of the code for the k- NN regression and linear regression didn’t take as long as understanding their concepts did.

**Bibliography:**

References

1 - <https://www.ageuk.org.uk/information-advice/care/find-care-support/how-to-find-help-at-home/>

2 - <https://www.telegraph.co.uk/news/health/elder/6836648/More-people-fear-losing-independence-in-old-age-than-death-survey-finds.html>

3 - <https://www.telegraph.co.uk/news/2048794/Britains-ageing-population-as-big-a-threat-as-climate-change.html>

**Appendix:**

Evidence/paperwork, that does not fit in elsewhere or in any other sub-section of the dissertation.