INTERNATIONAL
HELLENIC
UNIVERSITY

Building CO_2 emissions prediction using Machine Learning/Data Mining

Avramidou Alexia

SID: 3308190001

SCHOOL OF SCIENCE & TECHNOLOGY

A thesis submitted for the degree of

Master of Science (MSc) in Data Science

JANUARY 2021

THESSALONIKI – GREECE

Building CO_2 emissions prediction using Machine Learning/Data Mining

Avramidou Alexia

SID: 3308190001

Supervisor:	Assoc. Prof. Christos Tjortjis
Supervising Committee Members:	Prof. Panagiotis Bozanis Dr. Agamemnon Baltagiannis

SCHOOL OF SCIENCE & TECHNOLOGY

A thesis submitted for the degree of

Master of Science (MSc) in Data Science

JANUARY 2021

THESSALONIKI – GREECE

Acknowledgments

Firstly, I would like to thank my supervisor, Professor C. Tjortjis for his guidance and for encouraging me to work efficiently throughout the whole process by providing his valuable advice and knowledge. Also, I dedicate this work to my family and friends who supported me from the beginning in every way.

Abstract

This dissertation was written as a part of the MSc in Data Science at the International Hellenic University.

Predictive analytics have become a necessity in most sectors of everyday human activities. It is true that the exploitation of data has raised an increasing interest for extraction of useful information concerning energy consuming behaviors for buildings. Although the concept of smart cities is present for more than two decades, it is still an expanding knowledge domain. Smart Buildings aim to prioritize occupants' comfort along with reduced energy waste and emissions.

This dissertation focuses on the development of machine learning algorithms to predict greenhouse gas emissions caused by the building sector and identify key building characteristics which lead to excessive emissions. More specifically, two problems are discussed: the prediction of metric tons of CO_2 emitted annually by a building and building compliance to environmental laws according to its physical characteristics, energy, fuel, and water consumption. The outcomes prove that energy use intensity and natural gas use are significant factors for decarbonizing the building sector.

Avramidou Alexia

4 January 2021

Contents

1 Introduction

It is widely known that climate change is a global threat and immediate actions need to be taken to limit its most important effects. The operation of buildings account for approximately 40% of primary energy consumption globally, drawing the attention of governments to act instantly by implying energy policies and carbon emission measures [1]. Given this reality, countries and cities have already set strict long-term energy efficiency and carbon reduction goals for existing and new buildings. Indeed, New York City aims to reduce its carbon footprint by 80% by 2050. Also, during the Paris Agreement the objective of achieving a climate-neutral EU by 2050 has been endorsed [2], setting high goals for all sectors of human activity. These actions inevitably focus on buildings, as a high proportion of emissions derives from energy and fuel consumption from both residential and non-residential existing buildings.

To support global and city-scale decarbonization goals, energy disclosure directives are a significant policy tool to accelerate the transition towards climate neutrality [3]. The number of cities and local governments adopting energy disclosure legislations has increased the past few years and more building owners are required to report their property's energy consumption. Energy benchmarking allows decision makers to assess the energy performance of buildings and evaluate the energy profile of a whole city or region.

Additionally, there are multiple benefits of having building energy data available both for citizens and decision makers. For tenants and building owners, by reporting energy consumption data annually might help understand their own behaviors and lead to changes that will mitigate excessive energy waste. From decision makers perspective, monitoring energy and emissions data will allow them to have an outlook on how energy is consumed within a city scale and detect any progress over the years concerning decarbonization goals.

NYC has been collecting energy disclosure data since 2010, through the implementation of Local Law 84 (LL84) for large buildings. LL84 requires building owners to report their properties every year. The properties covered by this legislation are of size 50000 square feet at least [4]. In addition, owners are obliged to fulfill the requirements of Local Law 97 for their property's carbon footprint. Thus, an emissions intensity report needs to be submitted annually starting in 2025 or pay substantial fines [5]. However, there is a big financial and political concern that constrain the implementation of these laws to smaller buildings, mainly driven by the potential costs to building owners. Given this legitimate concern, it is essential for policy- makers to have alternative but reliable methods to assess and understand energy consumption patterns across different spatial scales.

This study evaluates several machine learning algorithms, including Random Forest, XGBoost, CatBoost and Artificial Neural Networks, to predict the annual greenhouse gas emissions from existing properties reported at energy disclosure records. More specifically, the first part focuses on predicting the actual number of metric tons of CO_2 emitted from buildings through regression, while the second part examines if the properties fall into acceptable emission boundaries and thus comply to LL97 law, through classification.

For this purpose, actual building and energy data from NYC's Local Law 84 (LL84) for the calendar year 2017 are used to train and evaluate our predictions, combined with LL97 emission limits for each building type. LL84 datasets are publicly available including records from 2010 to 2017. The goal of this research is to predict the environmental footprint of buildings and aid decision makers to understand the factors that contribute to excessive emissions and take actions for decarbonizing the building sector.

The structure of the thesis is the following. After a short introduction in Chapter 1, Chapter 2 includes the literature review, Chapter 3 contains the problem definition along with a brief description of the datasets used. Chapter 4 includes the preprocessing steps and some exploratory data analysis results, while in Chapter 5 the predictions are presented. In Chapter 6 results are discussed and evaluated and in Chapter 7 conclusions and future directions are presented.

All the experiments on this dissertation were executed in Python 3.6. Also, the algorithms and their implementation come from the *Scikit-Learn* package and for Neural Networks the *Keras* package has been used.

2 Literature Review

In this section, relative works are analyzed that approach our problem. At first, the concept of smart cities is presented, mainly focusing on smart buildings. Then, several studies are listed in which data-driven predictive models have been used for buildings.

2.1 Smart City

The definition of a smart city is still complicated, though the concept of smart technologies in cities around the world has gained the attention of many researchers the past few decades. In this chapter a deeper explanation of the term is presented along with some examples of everyday life domains in which smart city concepts are utilized.

2.1.1 The concept of a smart city

A smart city can be defined as a sustainable and efficient urban center that provides a high quality of life to its inhabitants through optimal management of its resources [6]. The transition towards smart cities has accelerated the past few decades because of the impact of new technologies in everyday lives and the daily human- device interaction [7]. However, it is still a complex concept possibly caused by the perception of 'smartness', which varies from city to city and depends on the existing local infrastructure and culture [8].

The 'smartness' of the city incorporates technologies that can be used into commercial applications by implying them on intelligent products and services [9]. Smart homes, communities, transportation and health care systems are equipped with embedded devices and sensors to interact with their environment. Internet of things (IoT) and cloud computing are very significant technologies for connectivity. Also, open public data enable real time decisions, but the production of large amounts of high frequency data within smart networks arise the need for a reliable Big Data platform for storage and processing [10].

2.1.2 Applications

Smart cities use multiple technologies to improve sectors of human activity such as health, transportation, energy, buildings, education, and tourism aiming to improve the quality of life without sacrificing the comfort of their citizens. Some of these applications are analyzed below. Figure 1 illustrates applications of smart city technologies.

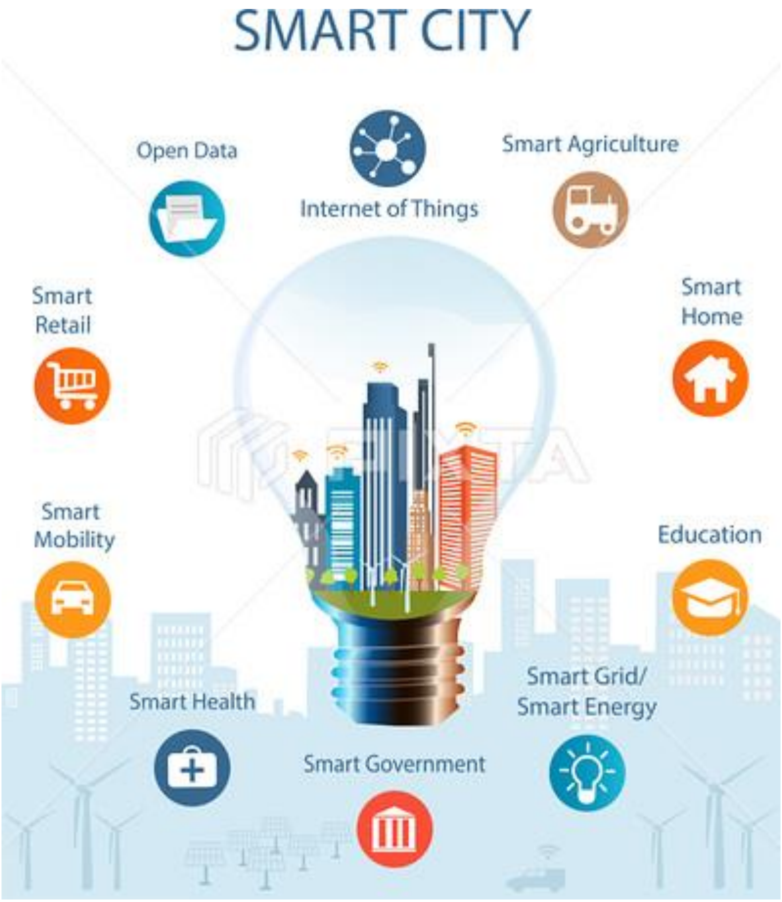


Figure 1: Smart city domains and applications

Transportation: Transportation and mobility are considered one of the key challenges for most of the cities globally. Smart traffic routing and smart parking solutions are two of the most known applications of smart city concepts. Traffic routing uses smart sensors placed in different areas and rows to detect traffic flows. In [7] it is emphasized that traffic prediction is a multidimensional problem affected by numerous factors, like accidents or social events at specific areas and mainly depends on weather conditions.

Health care: Smart healthcare projects enable easy access to patients' files, containing multiple diagnoses, details, tests, etc. Data will be available to doctors, laboratories, and other health experts. This way, waiting time for patients will be drastically reduced, paperwork will be eliminated and more importantly a complete image of patient's history will give insights in numerous health issues.

Buildings: Smart buildings could be defined as buildings that have been retrofitted and automated to reduce their excessive energy consumption and CO_2 emissions without compromising the comfort of the occupants [11,[12]. Buildings can become smarter in two ways; by implementing ICT solutions, or by focusing on retrofits aiming at energy efficiency [13]. Within smart buildings, the automation plays a major role both in commercial and residential buildings. Given that most of the total energy consumption is caused by HVAC systems, many IoT devices and sensors are connected with them for thermal comfort and energy efficiency enhancement.

Education: An intelligent education system uses technologies and learning tools to improve teaching techniques and students' learning experience. Some of the benefits are time saving, less paperwork and improves the connectivity between students, teachers, parents by introducing e-platforms [14].

2.1.3 Smart Buildings

There is a clear confusion concerning the differentiation between Smart and Intelligent Buildings. Although there is an increasing amount of academic literature and research focusing on defining this emerging concept, the answer is still not obvious on how the transition towards smart buildings can be achieved.

2.1.3.1 The meaning of intelligence

Evolving definitions of Intelligent Buildings have been developed since the early 1980s and continue to change and adapt using the latest knowledge and experience. In 1995 the Conseil International du Batiment Working Groups defined an Intelligent Building as:

"A dynamic and responsive architecture that provides every occupant with productive, cost effective and environmentally approved conditions through continuous interaction among its four basic elements: places (fabric; structure; facilities); processes (automation; control systems) people (services; users) and management (maintenance; performance) and the interrelation between them."

Later, it was suggested that Intelligent Buildings are equivalent to the Building Management Systems (BMS) within them [Brooks 2011]. However, BMS is usually one of the components within an intelligent building and not the entire system [15].

Trying to explain intelligence in buildings in a more understandable way, after traditional buildings, the automated buildings have taken their place, in which timers and central controls set a schedule for switching on and off lighting and heating. The next step, the intelligent buildings combined automation systems with sensors which allowed the building to adjust to user needs in real time [16].

2.1.3.2 What are Smart Buildings?

Smart buildings take it a step further from intelligent ones. That means that things are not just turned on and off, but the building collects data about how and when its systems and components are used and provides a real-time picture of its behavior. Networks, cameras and sensors are some of the technologies used to aid this procedure. Then several interesting trends are produced, such as peak hours, occupancy levels, different people's behavior at different times of day etc. In [15], it is mentioned that adaptability and integration between all aspects of the building will differentiate smart buildings from previous generations. Some examples of adaptability are:

- Different choices of occupants to enhance comfort at different times of day and different seasons of the year
- Changes in how occupants use the building
- Different occupancy data characteristics
- Varying yearly average external weather conditions

Figure 2 is an illustration of how the terms intelligent and smart differentiate, and which are the components of both building technologies. It is important to mention that smart building technologies mainly aim to maintain or even increase energy efficiency, minimize its environmental footprint and at the same time provide high satisfaction levels for its occupants. Improved materials and control systems will allow designers and engineers to create buildings that are nearly or completely energy independent or "(nearly) Zero- Energy Buildings", which is a goal to achieve in the next few years for most countries. Thus, smart buildings have integrated renewables such as solar arrays, photovoltaics and geothermal heating systems to produce their own heating and electric power. These technologies may initially produce high costs, but payback periods are sustainable and energy savings are huge. Figure 3 shows some common smart building components and technologies.

In addition, cloud computing and Internet of Things (IoT) play a significant role on developing digital services on buildings [17]. Common IoT applications in smart buildings are energy saving procedures, security enhancement, automations, and maintenance improvements. IoT enables operational systems to deliver more accurate information, as well as improves operations whilst providing the best conditions for occupants [18].

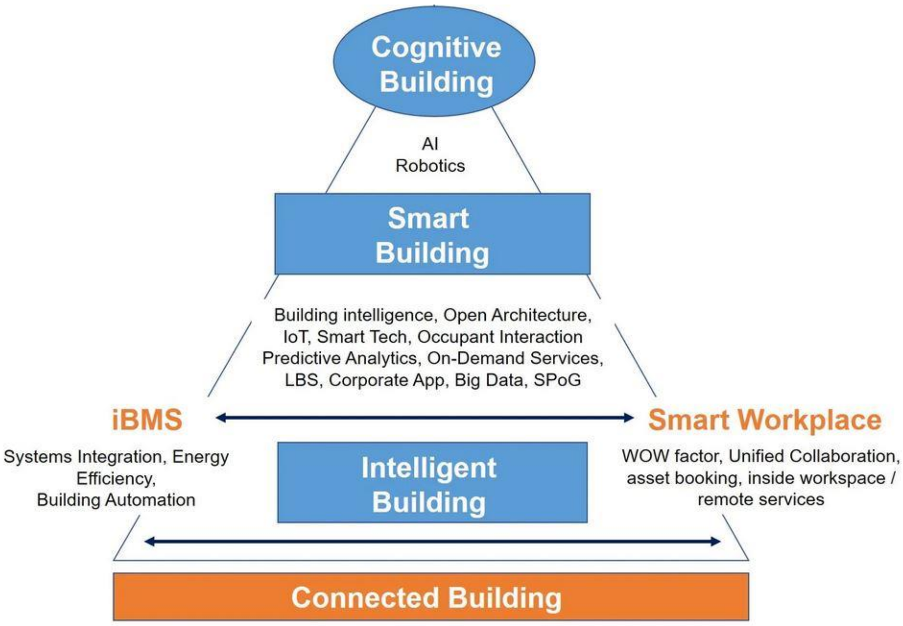

Cognitive Building
AI
Robotics
Smart Building
Building intelligence, Open Architecture, IoT, Smart Tech, Occupant Interaction Predictive Analytics, On-Demand Services, LBS, Corporate App, Big Data, SPoG
iBMS
Systems Integration, Energy Efficiency, Building Automation
Smart Workplace
WOW factor, Unified Collaboration, asset booking, inside workspace / remote services
Intelligent Building
Connected Building

Figure 2: The difference between smart and intelligent buildings.

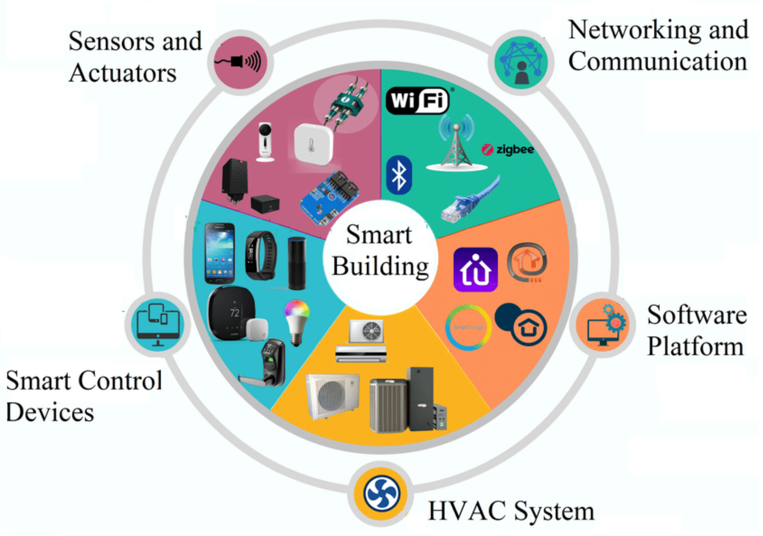


Figure 3: Smart building technologies

2.2 Predictive models for buildings

Predictive analytics are applied in almost any field, while machine learning and data mining techniques continue to grow and provide with interesting and accurate results. Building performance and energy consumption has been the subject of numerous academic research, driven by the urgent need of a “greener” building sector.

2.2.1 Energy consumption and performance prediction

In the following studies the aim is to make predictions about energy consumption and performance for different types of buildings by applying machine learning and data mining techniques.

This paper [19] presents a review of ML approaches including ANN, SVM, Gaussian based regression and clustering which have been applied in forecasting and enhancing energy performance for buildings. ANN is a powerful predictor in building energy forecasting, but several hyperparameters have to be adjusted and be selected properly. In contrast with ANN, SVM and GP are supervised using few parameters and provide satisfactory results. SVM surpasses ANN in load forecasting and the model can be built with less samples.

Another study [20] presents a review on unsupervised data analytics in mining big amounts of building operational data. The purpose is to improve operational performance of buildings. The authors state that unsupervised analytics are more practical and promising in discovering knowledge given limited prior knowledge, so they should be applied more in building operational and consumption data. It is mentioned that effective post-mining methods for knowledge selection should be studied more, and the development of semi-automated or fully automated post-mining methods could reduce complexity in such problems.

A lot of research has been conducted focusing on clustering techniques which could give insights about energy consumption patterns. In this work [21] a k-shape clustering technique has been applied to cluster building energy consumption patterns and then fitted into an SVR model to improve its forecasting accuracy. For this purpose, 10 institutional buildings have been examined and the k-shape clustering technique is compared to dynamic time warping (DTW) clustering. The results have indicated that k-shape clustering performs better than DTW clustering. The k-shape algorithm is then used to identify daily base energy consumption patterns for ten buildings. It is observed that the implementation of k-shape clustering helped improving forecasting accuracy, by comparing the model with another one without clustering.

Another study [22] analyzes time series data to identify buildings with similar temporal energy performance patterns. Using K-means clustering algorithm, two clusters are formed, “improving” and “declining” energy performance for both commercial and residential buildings in NYC. The authors discovered that larger and newer office buildings are more likely to perform energy reduction measures and have shown a significant improvement in terms of EUI between the examined years. Also, office buildings that participate in the NYC Carbon Challenge program are 138% more likely to have improved their performance over this period. Residential buildings that use heavy oil boilers are more likely to have increasing EUI over time.

Next-day energy consumption prediction and peak power demand have also gained interest the past few decades, given the fact that energy efficiency and savings have concerned both building owners and governments. In [23] ensemble methods are used combining eight base models. The energy consumption data used for this purpose are collected from the tallest building in Hong Kong. The results show that the accuracy of the ensemble method is significantly better than those of base models. The best performing base models appeared to be the Random Forests and support vector regression, which are assigned with the largest weights in the ensemble model. Also, by applying clustering analysis, performing feature extraction and generalized extreme studentized deviate (GESD), abnormalities concerning daily energy consumption profiles are detected successfully. Most of the outliers seem to come from public holidays, when the number of occupants in the building is very limited compared to normal days. The authors conclude that developing the models for this purpose can be time consuming, but prediction time can be very short for new inputs once the models are ready.

In addition, tenants’ behavior is proven to be a very affecting factor for energy consumption and therefore buildings’ emissions. In [24], the authors tried to discover patterns of usage in different types of residential buildings that could affect energy consumption and lead to increased carbon emissions. For this purpose, they performed a questionnaire survey to collect annual gas and electricity data for different types of dwelling. Clusters of higher and lower energy consumers were formed, and these clusters were related to indicators of energy consumption. The results have shown that there is a strong relationship between the number of bedrooms and energy consumption, as well as home working. Indeed, working from home appeared to be the strongest indicator of differences in gas and electricity consumption.

Until now, many researchers have applied Support Vector Machines for building energy consumption and performance prediction. In [25], SVM has been used for forecasting building energy consumption in the tropical region. Four office buildings in Singapore were selected which are located around the Central Business District. The data used for this study were utility bills of these four buildings, which have been collected through surveys, as well as weather data gathered from weather stations in Singapore. Weather data include information about dry-bulb temperature, relative humidity, and global solar radiation. The performance of SVM was explored adjusting two hyperparameters, C and ε, by applying stepwise searching method based on radial basis function (RBF) kernel. The results have shown that SVM performs better than neural networks and genetic programming algorithms. The reasons could be the small data pool used in this study, thus abnormal data were not so frequent. Also, when applying SVM for prediction someone needs less hyperparameters to optimize compared to neural networks.

In [26], a sensor-based forecasting model is developed to a multi-family residential building in New York using support vector regression. The authors aim to discover the impact of temporal (daily, hourly, 10 min intervals) and spatial (whole building, by floor, by unit) data have on the predictive power of the model. Results indicate that the optimal model is built with hourly consumption at the floor level. They conclude that more detailed data (by floor and by unit) produce better results.

In [27], the aim is to improve energy efficiency of the HVAC system using data from a skyscraper. To achieve this goal a Support Vector Machine Regression (SVMR) model was built based only on historical data of the building, which include information about its size, heating and cooling systems and other physical properties. The fact that the model relies only on historical data, it makes it easily applicable on different types of buildings.

In [8] a citizen-centric approach is presented for electricity consumption prediction in dwellings. Two problems are examined; produce building energy consumption patterns for each building examined and observe the behavior of grid through aggregated consumption data of all available properties. Three dwellings are selected for analysis, using both weather and consumption data collected for three consecutive years. The regression model was transformed into a binary classification problem, predicting two labels ‘high’ and ‘low’ describing the levels of consumption. The algorithms tested were Support Vector Machines (SVM), Random Forest, Stochastic Gradient Descent (SGD) and Logistic Regression. The results have indicated that the size of a building affects its consumer behavior, as it was more difficult to predict consumption levels for bigger properties. Also, turning the regression problem to a binary one, achieves better results when the point of separation is the mean of all instances.

Most of the studies conducted for energy consumption prediction conclude that is difficult to decide which ML algorithm is the best, as the nature of data and the application differs, as well as the purpose of each research. Artificial Neural Networks are very commonly used in these types of problems because of their high predictive power. In [28], an ANN model has been developed for electric demand prediction of a solar energy research center, the CIESOL building. In this study the authors tried to identify the most influential factors on electricity demand and discover interesting patterns. The most significant factors are outdoor temperature, solar radiation, and information about the solar cooling installation, especially the state of the heat pump. Two approaches have been developed, one considering the solar cooling installation and a simplified one without it. Various tests have been conducted including different types of days and consumption patterns, and different prediction horizons have been evaluated. The results were good for both approaches, but the first more complex approach gave better results in dynamic modeling (prediction horizon tends to infinity).

Another work [29] focuses on improving the performance of ANNs concerning the prediction of electric loads by conducting hypothesis test, information criteria and cross validation. The authors used one hidden layer in order to avoid over-parameterization. It is stated that the input variables are very important for this problem. The results indicate that some environmental variables such as ambient temperature and solar radiation are significant, while others such as wind velocity or humidity can be omitted. The day and time variables and the occupancy variables are very important, regardless of the dataset. Also, short time predictions are very accurate, but they are not the most used. Instead, next day predictors perform well in terms of accuracy and can be applied more easily.

Other studies have compared neural networks with ensemble methods to test their forecasting performance. In [30], artificial neural networks are compared with random forest for predicting the hourly HVAC energy consumption of a hotel in Madrid. The results indicate that incorporating social parameters, such as the number of guests increased prediction accuracy for both algorithms tested. Overall, ANN performed marginally better than RF. Nevertheless, ensemble-based methods, like RF could deal with multi-dimensional data better, for complex data like building data. The authors conclude that both algorithms have strong predictive power and could be applied in building energy applications.

The results of another study [31] have shown that fuzzy systems and neural networks using occupancy data are the best models to describe how energy is consumed in a building. Occupancy data were collected via Wi-Fi network connections and weather data from a weather station placed at the roof of the building. In these cases where historical data have been used, a big amount of data is needed for robust predictions. Also, schedules and events happening in the building affect significantly the predictions and need to be considered.

2.2.2 CO_2 emissions prediction

Several researchers have observed the more affecting factors for CO_2 emissions in the building sector and proposed methods for predicting buildings' environmental footprint.

In [32], a Back Propagation (BP) neural network model is presented for predicting CO_2 emissions caused by the Chinese commercial sector. This model is based on the index quantization ability of Random Forest (RF) and the performance optimization ability of PSO (particle swarm optimization). The authors state that other studies have only focused on algorithm optimization and model mixing, ignoring the selection of important indicators, thus this paper tries to fill this gap and construct a hybrid model for forecasting CO_2 emissions for the commercial sector. The data used for this study are national statistics of China's commercial sector from 1997 to 2017 and include 17 social, financial and energy indicators. By using RF to evaluate the significance of indicators the authors conclude that there are 7 of them that have a large linear or non-linear relationship with CO_2 emissions. These indicators are energy intensity, coal consumption, second industry GDP, education level, total population, business sector GDP and imports. Also, the use of RF can significantly improve prediction accuracy and the best performing model proposed for this purpose is the RF-DPSO (double particle swarm optimization)-BP.

In [33], the goal is to estimate indirect building carbon emissions within the boundaries of various types of Local Climate Zones (LCZs). This research aims to discover interesting patterns and help improving energy management in specific regions. The model used random forest algorithm to make predictions and the results show the linkage between emission coefficients and different LCZ categories in Shanghai. The authors conclude that it is necessary to include not only morphological parameters which are used in this study, but also information about occupancy, HVAC systems, building use, materials and more. Thus, there is a need to modify the Local Climate Zones and develop the Local Energy Zones.

2.2.3 Retrofits and energy efficiency measures

Minimizing greenhouse gas emissions from the building sector requires energy efficiency measures to be applied for both new and existing buildings. Some works include estimations about potential energy savings using multiple machine learning and data mining techniques. This kind of predictions will help targeting poorly-performing buildings and provide with appropriate retrofit scenarios in order to enhance their energy performance and limit their emissions.

In [34], cluster analysis is used to estimate potential energy savings in lighting systems in different types of buildings. The clustering approach is compared with the general averaging one, in which a mean value is used for predictions, averaged by a sample of buildings. The results have shown that the estimated energy savings by using EM algorithm analyzing data from an energy audit database have much smaller errors than those from the traditional approach described above. The building types examined were categorized in three main groups: hotels and hospitals, offices and schools, department stores. The clustering technique has given more accurate results for all three categories. The authors conclude that the proposed clustering method could be applied to estimate energy savings for lighting systems, for HVAC systems and help to a more precise estimation of CO_2 emissions for the building sector.

In [35], the aim is to generalize self-reported energy data from a small sample of buildings to a city-scale level. Three different machine learning algorithms are used (Linear Regression, Random Forest, Support Vector Regression) and feature selection techniques to make predictions from the LL84 data, which is self-reported energy disclosure data for large buildings. The results have shown that Linear Regression performs best when predicting total building energy consumption at the zip code-level for the entire city, while Support Vector Regression performs better in terms of accuracy when estimating energy use within the sample of LL84 buildings. Also, building size, use and morphology seem to be significant attributes for energy use prediction at the building and zip code levels. Larger buildings are found to have smaller EUI (energy use intensity) while taller ones are more intensive. The authors mention that it was more challenging to predict natural gas usage because of the bimodal distribution of gas consumption (some buildings use it only for cooking fuel, while others use natural gas for heating and hot water) and the lack of information about natural gas distribution infrastructure.

In [36], the goal is to classify educational buildings according to their energy performance for space heating and evaluate energy savings in the school sector. The data set included 1100 school buildings school buildings in Greece and contain information about annual consumption for space heating and lighting, the area of the building, number of students and professors, the power of the boiler, the schedule of operation and the manufacturing year of the building. The clustering technique applied was K-means and five balanced energy classes were formed. In these energy-balanced classes the purpose was to find a typical school building as a representative for each group. To achieve that, the authors used PCA to perform analysis on the potential energy savings for each group of school buildings. This classification could be beneficial for decision makers in to implement energy saving measures and set energy performance goals.

In [37], the aim was to classify existing buildings and identify a limited number of representative buildings to be examined for refurbishment. The sample buildings were almost 60 in the province of Treviso in Italy and they were clustered applying a modified K-means approach. For grouping schools into clusters, the authors used real consumption data which were correlated to buildings characteristics. The parameters with the highest correlation with energy consumption levels were used to form the clusters. After identifying the representative schools, it would be easier to classify them according to a priority list to apply retrofit measures.

Some studies such as [38], have proposed energy conservation measures (ECMs) by analyzing energy audit data. The authors state that ECM implementation could be encouraged by policies and legislations which require energy audits, consultants and recommendations which can be costly and time consuming. Thus, the purpose of this study is to accelerate the adoption of building ECMs with reduced costs and complexity. The user-facing falling rule classifier (FRL) classifier performs well for cooling system prediction, distribution system, domestic hot water, fuel switching, lighting, and motors conservation measures. The authors conclude that this work may aid providing effective low-cost retrofit options for building owners to improve energy efficiency and reduce GHG emissions.

Similarly, in [39], two families of ANNs are created, by using energy simulation outcomes as targets for training and testing the models. The first family of ANNs aims to assess the energy performance of the existing building stock, while the second one aims to estimate the impact of energy retrofit measures (ERMs). The case study included office buildings built in South Italy in the period 1920-1970. The results have shown that the outcomes of this study are satisfactory, as they obtained values of relative errors comparable to those of previous studies in which ANNs have been used for energy performance predictions. The authors state that in this study they achieved to perform predictions for any member of an established building stock, unlike other research which refer to single buildings or global behavior of building clusters.

2.2.4 Fuel consumption, thermal comfort, and occupancy

Several machine learning algorithms and data mining techniques have been used to predict fuel consumption, heating and cooling demand which contribute significantly to greenhouse gas emissions.

In [40], machine learning algorithms are used to make long-term predictions (i.e. one year ahead) at one-hour resolution for fuel consumption in several commercial buildings and various climate zones. A feature selection method has been applied to select the best input variables and the machine learning algorithms used for the problem were Neural Networks, Gaussian Process Regression, and multivariate linear regression. NNs and GPR seemed to perform better than linear regression and thus they were included as part of the model which was developed. The results can be used to estimate on-site fuel consumption and emissions from buildings and enhance decision making and decarbonization strategies implementation.

In [41], the purpose of the study was to take advantage of the large number of energy certificates for buildings which are available online and make predictions about heat demand. Artificial Neural Network was used for predictions and this methodology aims to detect anomalies in building energy certificates, thus it would facilitate to check and correct suspicious entries. Also, sensors and other smart technologies in several buildings provide useful data about occupancy and thermal comfort.

As it was mentioned above, smart technologies help gathering useful information about buildings' and occupants' behavior. In [42], paper a Wi-Fi sensing platform is introduced to provide information about occupancy in smart buildings in a privacy-preserving manner. The authors used deep learning methods such as LSTM and Convolutional Neural Network to identify several human activities. This study could contribute to achieving higher levels of energy efficiency in buildings and reducing CO_2 emissions while preserving a good air quality and occupant comfort.

In [43], 36 machine learning algorithms are compared to select the best one for indoor temperature prediction in a smart building. It is found that the Extra Trees regressor gives the best results. The aim of this study is to incorporate these predictions into building management systems to improve energy efficiency.

Similarly, [44] compares several machine learning models to predict heating and cooling loads in residential buildings. The algorithms used for this purpose were artificial neural network, generalized regression neural network, radial basis neural network, support vector machines and others. Among all the models used, the radial basis function network gave the best results, comparing MAPE, MAE and RMSE scores.

2.2.5 Fault and anomalies detection

Fault detection and diagnosis has been the subject of many studies and has been proven to be very beneficial in buildings and control systems. Detecting anomalies could save big amounts of wasted energy and unnecessary CO_2 emissions and could enhance thermal comfort as well.

In [45], a system is presented which is capable of automating detection and diagnosis of faults in commercial building HVAC systems. This system detects faults in real time using data from two sources. The first dataset includes information about an office building in Australia and the second one is obtained from the ASHRAE-1020 FDD project. For this project, Hidden Markov Models (HMM) has been used to learn relationships between groups of points during both normal and faulty operation. Also, parallel models and clustering techniques were used to overcome local optima issues and Data Fusion was applied to resolve conflicting diagnoses from different models. Furthermore, the authors tried to detect any interrelationships and correlations between several group of sensors to improve model accuracy.

Another study [46], deals with two problems; predict energy consumption of a residential building taking into consideration other buildings in its neighborhood and detect faults in building sub-systems which lead to excessive energy consumption. For energy consumption prediction, environmental parameters are not taken into account because similar buildings in the same neighborhood have the same weather conditions, so environmental uncertainties do not affect the results of the forecasting process. The parameters used are locations of internal walls, ceiling heights, minimum and maximum temperatures allowed by control, heating/cooling air temperature, lighting, density of people, fiberglass insulation thickness for exterior walls, roof insulation thickness, window thickness and roof solar absorbance. The authors evaluate their approach using machine learning algorithms such as ANN, SVR, multilinear regression. The experiments are performed four different days, one for each season. They observe that their predictions are robust to small changes in building structures. However, if the buildings are significantly different, the results are not reliable. They propose that future works should try using an optimization technique in combination with energy prediction to improve the energy efficiency of similar buildings in the same neighborhood. The second problem they deal with in this study is fault diagnosis, and their goal is to detect that a fault occurs, as well as locate and identify the fault. For this purpose, they used a decision tree model to diagnose the fault type, which was trained from labeled observations using the software RapidMiner. They concluded that many of the sensors used, contribute minimally to diagnosis and 3 sensors give almost the same results as 12. The authors state that if some faults have other undesirable effects and have no impact on energy consumption, then an alternative method for detection is needed.

In [47], a support vector machine (SVM) model is used to predict and diagnose anomalies in public buildings energy consumption. For this study 11 input parameters were used such as historical energy data, climatic and time-cycle factors. The authors focus only on predicting electricity consumption for the air conditioning during the summer months. The authors state that future work should include different end-use building energy sources like heating, cooling, lighting, cooking etc. This research provides theoretical guidance and a practical data reference for building operations management.

In [48], the authors propose a generic collective contextual anomaly detection framework (CCAD). The CCAD framework uses a sliding window approach, as well as historic sensor data and generated features. The aim is to identify abnormal behaviors. The results show that the CCAD framework can successfully detect anomalies in energy consumption related to HVAC systems.

In [49] problems related to Building Management Systems (BMS) components are discussed and how these problems affect the buildings energy performance. Also, several methods are presented that can help diagnose these types of issues. The authors conclude that Internet of Things (IoT) could be used in diagnostic processes of smart buildings, as building intelligent solutions could help reducing energy waste and carbon emissions.

2.2.6 Benchmarking and energy rating

Energy benchmarking is often used to evaluate the energy performance of buildings and is a crucial step towards reducing emissions. Comparability is a vital element to the success of a benchmarking system and has been the subject of many studies.

In [50], the aim is to improve the comparability of benchmarking the energy performance of English schools assessing the impact of various features, such as built form or occupancy. Energy performance data from 465 schools were analyzed using ANNs. The results indicate that for a 4-year period, electricity consumption has increased, and heating consumption has decreased in both primary and secondary schools. Also, secondary schools appeared to be significantly more energy intensive than primary schools and natural and mechanical ventilated schools differed in terms of electricity consumption. By analyzing the dataset using ANNs, the floor area and the number of pupils seemed to be very important determinants of schools' energy use. In addition, parameters such as built form and exposure ratios appeared to be significant too. The authors state that the differences spotted between primary and secondary schools indicate that there is a need to reexamine the way that non-residential buildings are classified and benchmarked.

In [51], a method for energy classification and rating of school buildings is presented. This method is based on fuzzy clustering techniques and it is compared with frequency rating techniques. The fuzzy clustering method forms more robust classes avoiding imbalanced classes and classifies the buildings more precisely according to their common characteristics and similarities. The data used for this study included energy consumption information of school buildings in Greece from almost all geographic departments of the country and for a three-year period. Energy consumption data has been obtained from energy bills and information about operational periods, number of students, construction characteristics, installed equipment have been used for the purposes of the study as well. The results indicated that school buildings should improve their energy consumption and environmental quality considerably. Also, this clustering method could be applied easily to classify other building types as well.

In [52], a new methodology for buildings energy benchmarking is discussed. The methodology contains feature selection, clustering algorithm adaptation, results validation, and interpretation. The dataset used contains information for 5215 commercial buildings such as building size, year of construction, types of energy used, energy consumption and equipment. In comparison with the energy star approach, it has been shown that the proposed methodology was able to provide a more comprehensive benchmarking approach. This is because the clustering approach incorporates various building characteristics which affect energy usage, while the Energy Star approach classifies the buildings according to their use type.

3 Problem Definition

Buildings are responsible for significant percentage of energy use and carbon emissions. Energy consumption can be reduced by implementing several energy efficiency measures concerning heating, cooling, lighting, and renewable energy systems. Several studies have been conducted to predict energy consumption patterns and evaluate the factors that affect energy waste both in existing buildings and new constructions. Despite the significance of the afore mentioned studies, there is limited research focusing on forecasting carbon emissions caused by the building sector and which factors contribute most to the environmental footprint of a building.

3.1 Problem

Since buildings account for most of the primary use and CO_2 emissions in dense urban areas, countries are increasingly adopting long-term decarbonization and sustainability plans designed to reduce carbon emissions and mitigate the negative effects of climate change [35]. These long-term policies and legislations focus on “greening” existing buildings and constructing new nearly zero or zero energy buildings. Building owners are required to report every year their energy consumption levels, as well as fuel consumption for heating and cooling and comply with the carbon intensity limits. Hence, there is an urgent need to understand the factors that lead to excessive emissions and be aware of which types of buildings are less “green”. In this way, the decarbonization of the building sector will be achieved faster and in a more targeted way.

This work analyzes an energy disclosure dataset with the primary purpose of predicting the total greenhouse gas emissions of a building and focused on discovering any useful information about factors causing excessive emissions. Also, this work can give insights to building owners and decision makers on whether a building complies or not to the specific requirements of decarbonization legislations. Forecasting carbon emissions is not only conducted for building owners and citizens, but also can help governments and city planners to reform strategies and regulations in order to achieve decarbonization goals.

3.2 Dataset description

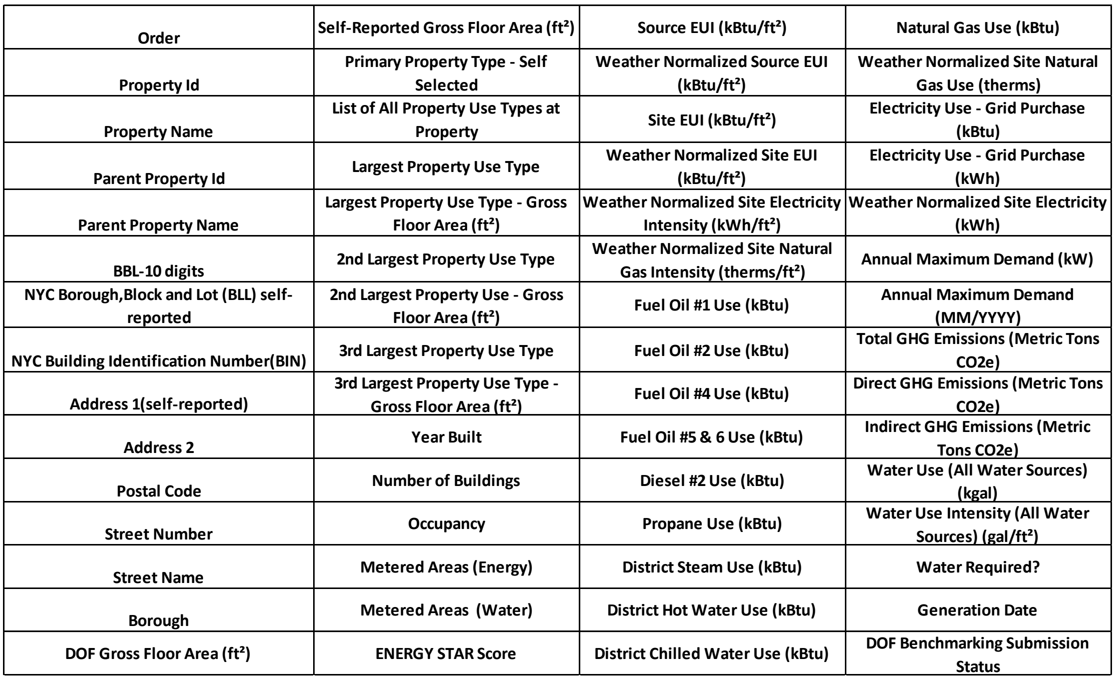
This section contains information about the datasets used for the purposes of the thesis. Two data sources were used which are analyzed in detail below.

3.2.1 Local Law 84 energy disclosure data

Local Law 84, or the NYC Benchmarking Law requires annual benchmarking and disclosure of energy and water usage information. LL84 covers properties with a single building with a gross floor area greater than 50000 square feet and lots having more than one building with a gross floor area greater than 100000 square feet. This dataset includes information about energy use by fuel type, physical descriptors as well as information concerning occupancy, water use and greenhouse gas emissions.

Metrics are calculated by the Environmental Protection Agency's tool ENERGY STAR Portfolio Manager and data is self-reporting by building owners. A public version of the dataset is released annually on the NYC Open Data portal containing a subset of collected data. For this study we chose data for Calendar Year 2017, which is the latest version publicly available. Table 1 below lists the data fields contained in our dataset.

Table 1: Data fields in LL84 energy disclosure dataset



Order	Self-Reported Gross Floor Area (ft²)	Source EUI (kBtu/ft²)	Natural Gas Use (kBtu)
Property Id	Primary Property Type - Self Selected	Weather Normalized Source EUI (kBtu/ft²)	Weather Normalized Site Natural Gas Use (therms)
Property Name	List of All Property Use Types at Property	Site EUI (kBtu/ft²)	Electricity Use - Grid Purchase (kBtu)
Parent Property Id	Largest Property Use Type	Weather Normalized Site EUI (kBtu/ft²)	Electricity Use - Grid Purchase (kWh)
Parent Property Name	Largest Property Use Type - Gross Floor Area (ft²)	Weather Normalized Site Electricity Intensity (kWh/ft²)	Weather Normalized Site Electricity (kWh)
BBL-10 digits	2nd Largest Property Use Type	Weather Normalized Site Natural Gas Intensity (therms/ft²)	Annual Maximum Demand (kW)
NYC Borough,Block and Lot (BLL) self-reported	2nd Largest Property Use - Gross Floor Area (ft²)	Fuel Oil #1 Use (kBtu)	Annual Maximum Demand (MM/YYYY)
NYC Building Identification Number(BIN)	3rd Largest Property Use Type	Fuel Oil #2 Use (kBtu)	Total GHG Emissions (Metric Tons CO2e)
Address 1(self-reported)	3rd Largest Property Use Type - Gross Floor Area (ft²)	Fuel Oil #4 Use (kBtu)	Direct GHG Emissions (Metric Tons CO2e)
Address 2	Year Built	Fuel Oil #5 & 6 Use (kBtu)	Indirect GHG Emissions (Metric Tons CO2e)
Postal Code	Number of Buildings	Diesel #2 Use (kBtu)	Water Use (All Water Sources) (kgal)
Street Number	Occupancy	Propane Use (kBtu)	Water Use Intensity (All Water Sources) (gal/ft²)
Street Name	Metered Areas (Energy)	District Steam Use (kBtu)	Water Required?
Borough	Metered Areas (Water)	District Hot Water Use (kBtu)	Generation Date
DOF Gross Floor Area (ft²)	ENERGY STAR Score	District Chilled Water Use (kBtu)	DOF Benchmarking Submission Status

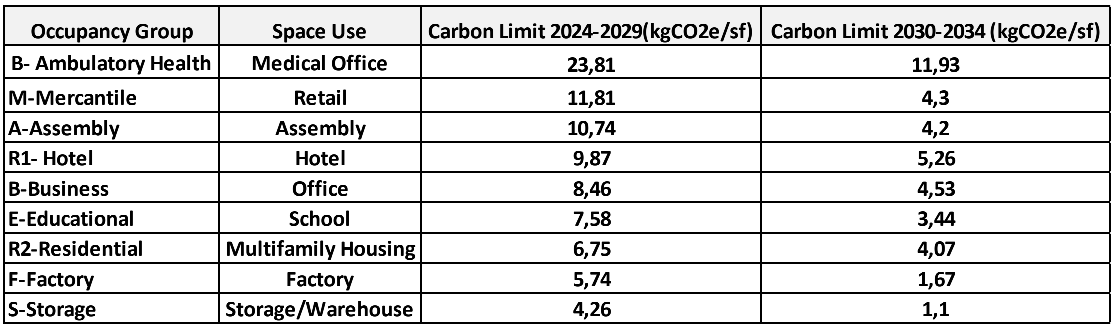
As observed on Table 1 above, the dataset contains some spatial and physical information about reported buildings such as property name, property ID, address, postal code, street number, borough etc. BBL number is a 10-digit property borough, block, and lot identifier. The first digit represents the borough where 1 is for Manhattan, 2 is Bronx, 3 is Brooklyn, 4 is Queens, 5 is Staten Island. The following digits represent the tax block and the tax lot number. BBL number is a unique identifier for each building reported. For the property type fields, several options are available in Portfolio Manager and can be either residential (multifamily building) or non-residential (hotel, restaurant, hospital, office, warehouse etc.). Year build stands for the year in which the property was constructed. The occupancy field contains a percentage of the property's gross floor area which is occupied and operational. Metered Areas for energy and water is a designation of what areas within the building are covered by energy and water meters accordingly. Energy Star score is a percentile ranking calculated in Portfolio Manager based on self-reported energy usage of the reporting year. Information about energy usage are provided from various columns, such as site or source EUI (energy use intensity) and their weather normalized values. Fuel oil use is a summary of the annual consumption of an individual type of energy.

Also, information about natural gas use, diesel, steam, and water use are provided. Greenhouse gas emissions are calculated for the reported year in metric tons of carbon dioxide equivalent. Total greenhouse gas emissions include both direct and indirect emissions. Release date contains the date of submission for a specific property and water required indicates if the property was eligible to use water benchmarking data.

3.2.2 Local Law 97

Local Law 97 sets detailed requirements for two initial compliance periods: 2024-2029 and 2030-2034. Buildings over 25000 square feet are required to meet annual carbon intensity limits during each compliance period based on building type. To comply, building owners must submit an emissions intensity report every year or pay substantial fines. In this work, we try to predict whether a building complies or not to the according compliance period using the LL84 dataset combined with the carbon emissions intensity limits provided by LL97. The emissions intensity limits are listed below in Table 2.

Table 2: Carbon emissions intensity limits by property type and period.



Occupancy Group	Space Use	Carbon Limit 2024-2029(kgCO2e/sf)	Carbon Limit 2030-2034 (kgCO2e/sf)
B- Ambulatory Health	Medical Office	23,81	11,93
M-Mercantile	Retail	11,81	4,3
A-Assembly	Assembly	10,74	4,2
R1- Hotel	Hotel	9,87	5,26
B-Business	Office	8,46	4,53
E-Educational	School	7,58	3,44
R2-Residential	Multifamily Housing	6,75	4,07
F-Factory	Factory	5,74	1,67
S-Storage	Storage/Warehouse	4,26	1,1

In this work, LL97 was used to solve a binary classification problem: whether a building complies or not to the specific requirements and it was combined with LL84 to determine the limits of acceptance according to building type. Both periods were used to make predictions by solving the same classification problem with different acceptable limits. These two identical problems are solved separately in order to make comparisons and draw some conclusions.

4 Greenhouse gas emissions

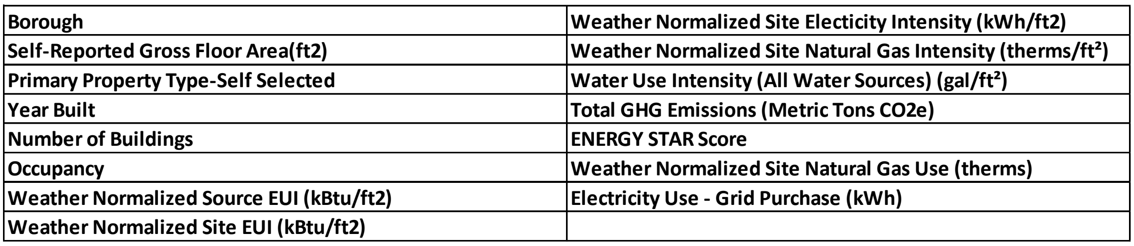
Most of the times, datasets need a preparation in order to be ready for analysis and prediction. Preparation includes handling missing or redundant values, removing misreported or anomalous entries and maybe making some corrections if needed. LL84 data is self-reported, therefore many data fields suffer from missing values and outliers.

4.1 Pre-processing

First, we remove entries with duplicate or missing Borough, Block and Lot (BBL) number, because as we have mentioned BBL is a unique spatial identifier for properties in NYC. Then we remove observations with zero or missing values in their reported weather normalized source EUI and we do the same for total GHG emissions.

Subsequently, we observe that some features are useless for our study because they contain in all rows a “Not Found” entry. Thus, we dropped the following columns: Water Required, DOF Gross Floor Area and DOF Benchmarking Submission Status. Our final dataset consists of 15 features which are listed at Table 3. Fields were removed because they either suffered from a high percentage of missing values or they were not affecting our predictions.

Table 3: Fields of LL84 kept for analysis



Borough	**Weather Normalized Site Electicity Intensity (kWh/ft2)**
Self-Reported Gross Floor Area(ft2)	**Weather Normalized Site Natural Gas Intensity (therms/ft²)**
Primary Property Type-Self Selected	**Water Use Intensity (All Water Sources) (gal/ft²)**
Year Built	**Total GHG Emissions (Metric Tons CO2e)**
Number of Buildings	**ENERGY STAR Score**
Occupancy	**Weather Normalized Site Natural Gas Use (therms)**
Weather Normalized Source EUI (kBtu/ft2)	**Electricity Use - Grid Purchase (kWh)**
Weather Normalized Site EUI (kBtu/ft2)	

Specifically, features like Property ID, Parent Property ID, NYC Building Identification Number(BIN), Address 1, Address 2, Postal Code and Order were excluded because the BBL number provided all the information we needed to identify a specific building. Also, Fuel Oil Use (from number 1 to number 6), Diesel Use, Propane Use, District Hot Water Use, District Chilled Water Use, District Steam Use, and Annual Maximum Demand columns were dropped because they were almost blank. Figure 4 illustrates the process of feature selection.

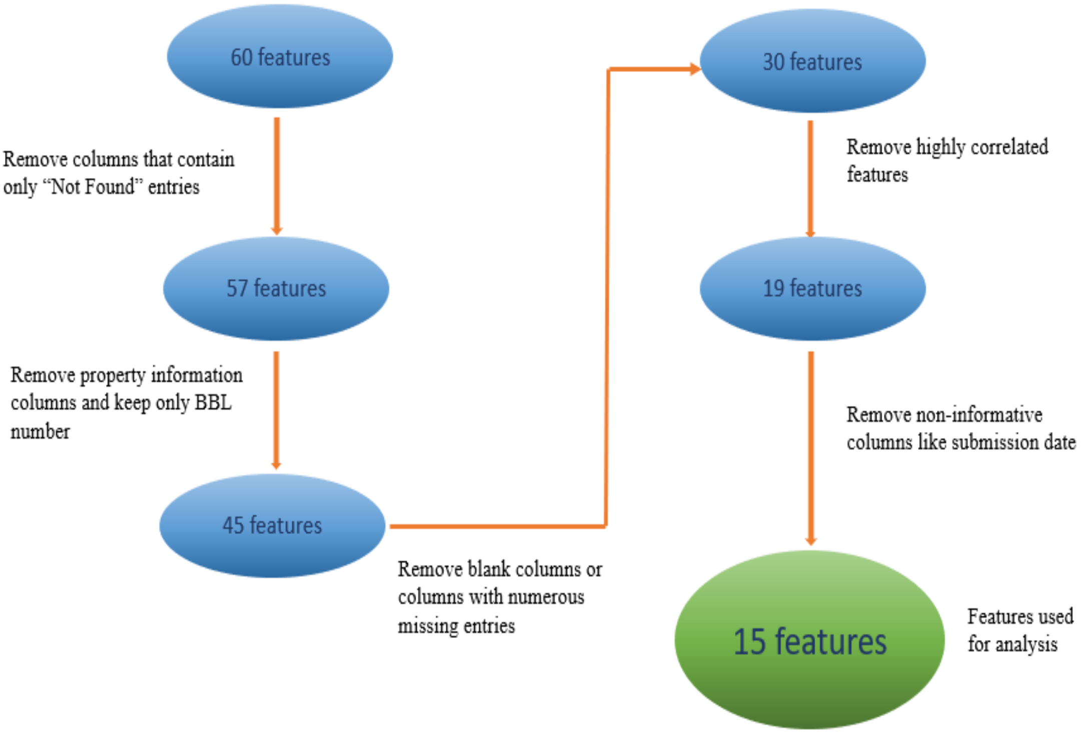

60 features
Remove columns that contain only "Not Found" entries
57 features
Remove property information columns and keep only BBL number
45 features
Remove blank columns or columns with numerous missing entries
30 features
Remove highly correlated features
19 features
Remove non-informative columns like submission date
15 features
Features used for analysis

Figure 4: Diagram of feature selection process

The next step was to group building type values in order to be compatible with the building types listed on Local Law 97. More specifically, the building types were clustered into 9 main categories: Office, Educational, Hotel, Residential, Warehouse, Public Building, Retail, Hospital, and Other. Below at Figure 5 we can see the proportion of each building type in our dataset. As we can see, residential buildings (multifamily housing) appear 68.3% of the times in the energy disclosure dataset and non-residential buildings appear only 31.7% of the times. This imbalance between dwellings and non-residential buildings will make it more difficult to draw conclusions and understand the behavior of several building types.

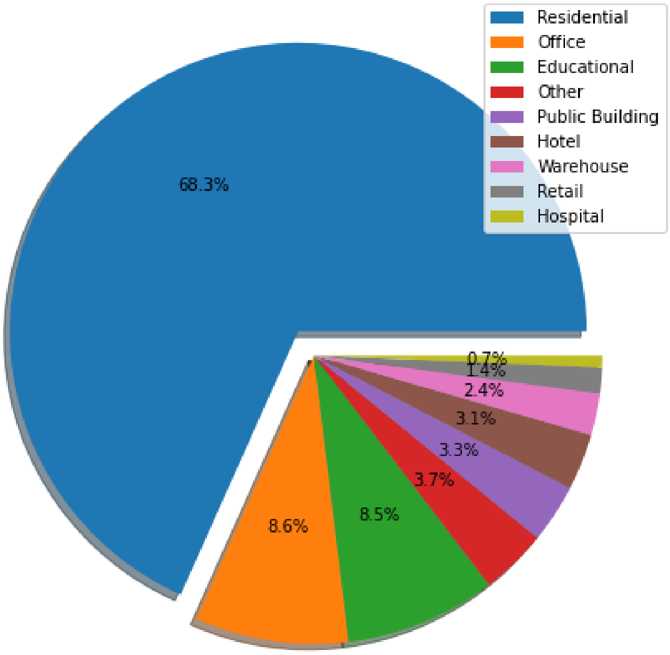


Figure 5: Building types after grouping.

To filter our data from misreported or anomalous entries, we apply for each building type a logarithmic transformation to the Total GHG values. By doing so, we try to approximate the normal distribution given that we have observed a log-normal distribution in the raw data, as shown in Figure 6. Observations were excluded from the analysis if they were falling outside the threshold of two standard deviations from the logged mean.

Figure 6 and Figure 7 show an illustration of the outlier detection process. Observing Figure 7, the range of CO_2 metric tons emitted annually has been bounded between 123 tons and 805.3 tons, while the mean value for greenhouse gas emissions is 361.02 metric tons of CO_2. As we have mentioned previously, the LL84 dataset is self-reported so many values were misreported. As a result, filtering outliers is essential to ensure that our predictions are robust and avoid large errors. Figure 8 shows how the building type percentages are affected after removing outliers. As we observe, educational buildings outnumber offices after dropping anomalous observations, while residential buildings remain at the first place of buildings reported. This means that most misreported or anomalous values tend to appear in non-residential building types indicating a lack of effective BMS systems.

Then, for the rest data fields we replace missing entries with the mean value of the respective column. Finally, we perform one hot encoding for the feature Primary Property Type to fit our data in several machine learning algorithms and make our predictions.

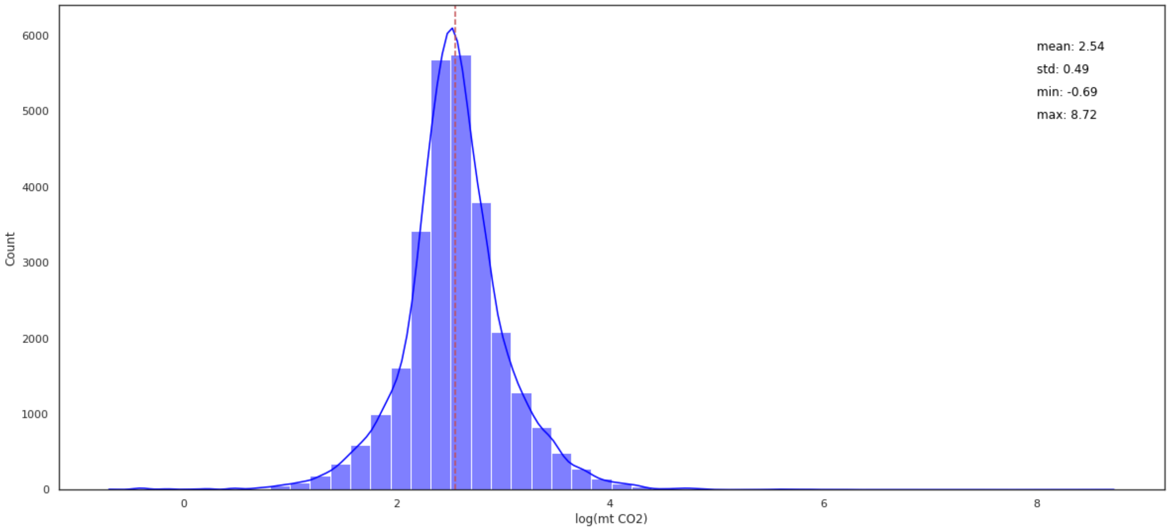


Figure 6: Histogram of log transformed GHG emissions. The red line shows the log sample mean.

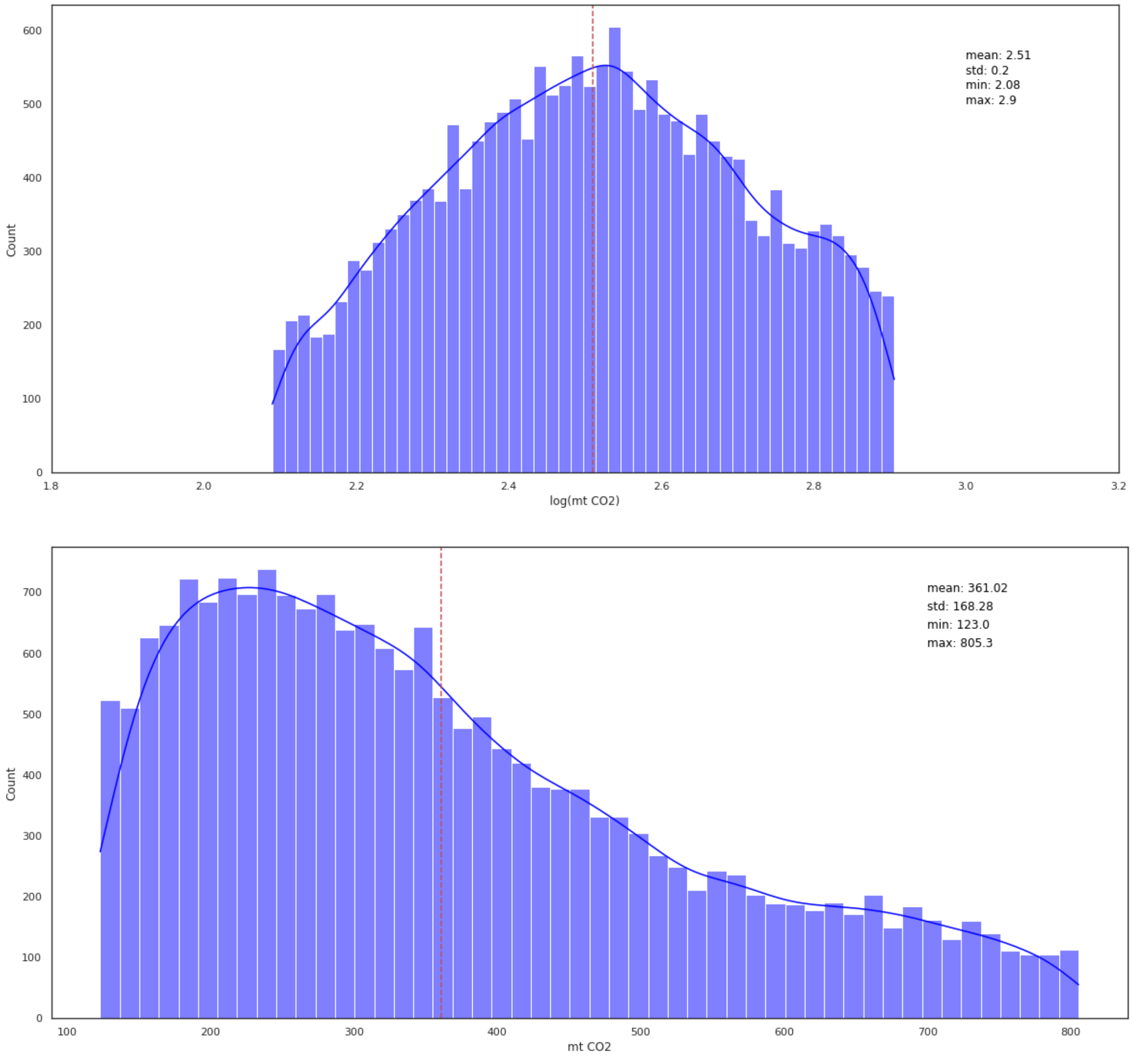


Figure 7: Top: Histogram of log GHG emissions after removing entries falling out of two standard deviations from the logged mean for each building type. Bottom: Histogram of the original values of GHG emissions after eliminating outliers.

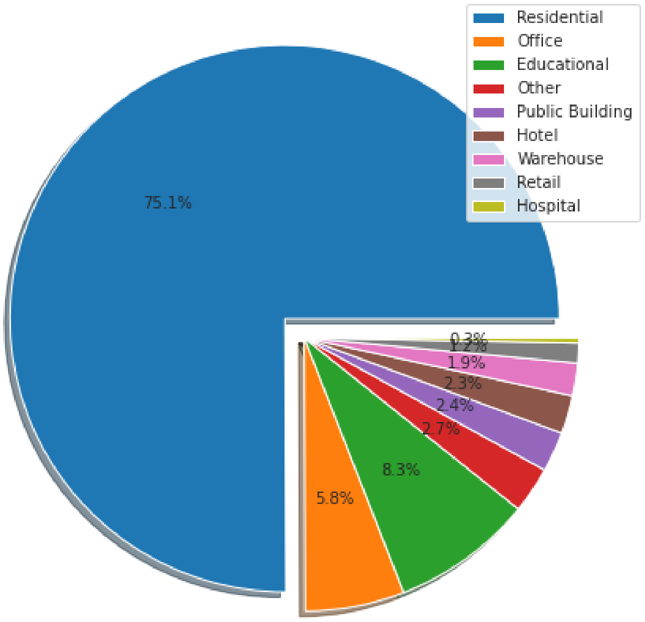


Figure 8: The percentage of building types contained in the dataset after removing outliers.

4.2 Patterns

After performing the initial preprocessing, it is essential to understand the profile of buildings reported and discover some useful patterns about annual emissions. Figure 9 shows which building types are less “green” based on their submitted greenhouse gas emissions. As we can see, retail properties tend to emit almost 450 metric tons of CO_2 every year. Hotels seem to be at the second place of the less “green” building types, while offices and educational buildings emit approximately 400 metric tons annually. Actually, this pattern revealed is expected, as non-residential buildings consume more energy on HVAC and lighting systems and indicate high occupancy rates.

Then, we tried to discover any correlation between different boroughs and their building emissions. Indeed, buildings in some boroughs in NYC show lower emission values compared to others. By observing Figure 10, it is interesting that within each borough we find the same pattern we discovered in Figure 9. More specifically, retail buildings show high emissions in all boroughs, especially in Bronx. Also, Manhattan and Queens do not indicate excessive emissions compared with Bronx or Staten Island.

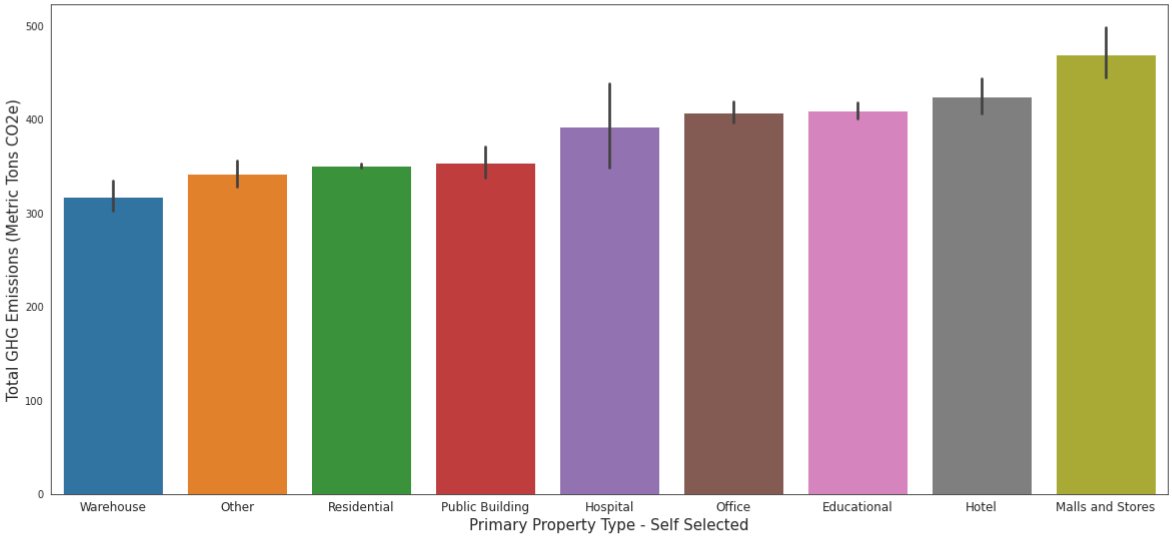


Figure 9: Annual greenhouse gas emissions according to property type

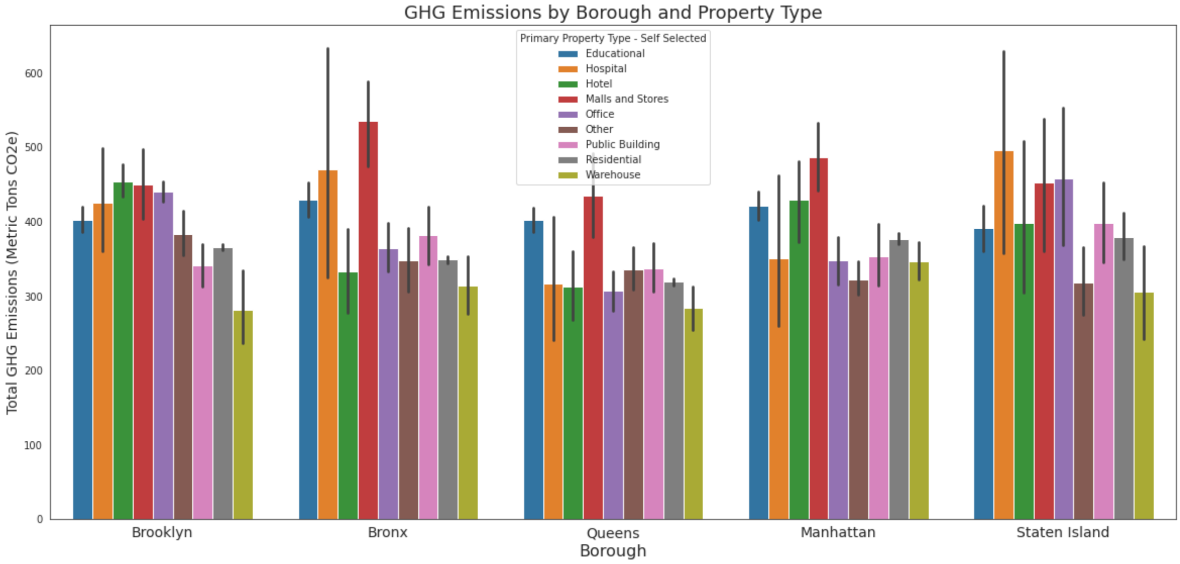


Figure 10: Greenhouse gas emissions by borough and property type.

Next, buildings were grouped according to their recorded year of built in five groups; buildings constructed before 1900, between 1900 and 1950, 1950 to 1970, 1970 to 2000 and after 2000. As shown in Figure 11 new constructions tend to be less energy intensive than older buildings. This illustration indicates that new or refurbished buildings are more energy efficient, using greener materials, smart devices and less energy consuming heating and ventilation systems.

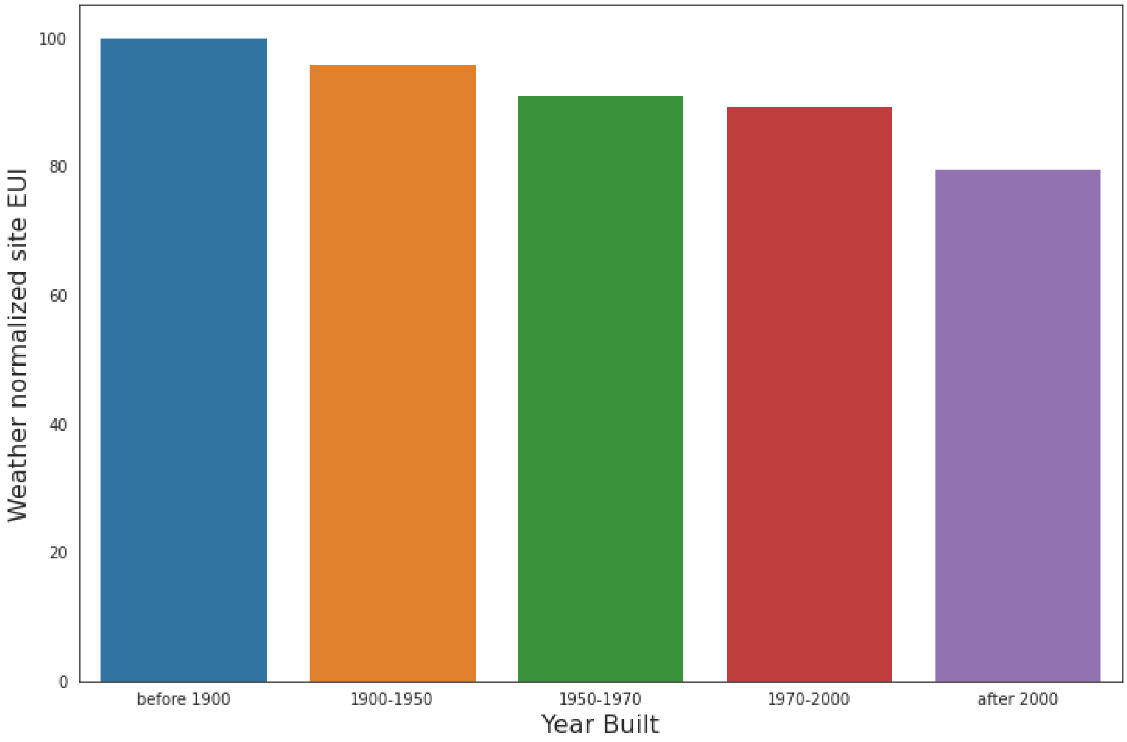


Figure 11: Energy intensity of buildings which were constructed at different time periods.

Exploratory data analysis is a very significant step of predictive analytics, which aims to give insights about feature relationships, correlations, useful patterns and help understand the data prior to predictions. A heatmap illustrates the correlations between all features as color in two dimensions. Diagonal values are always equal to 1, as they represent the correlation between two identical attributes, thus values above the diagonal are omitted.

Figure 12 shows how features are related to each other. As we observe, weather normalized site and source electricity intensities are highly correlated, which is not a surprise. In addition, natural gas intensity seems to relate to source and site EUI. Also, the logged GHG emissions are extremely correlated with total GHG emissions, as this field represents the logarithm of annual emissions. Except from the logarithmic column, emissions are correlated with electricity use and gross floor area, while some features do not appear that relevant, such as the building type, energy star score, water use, year build and occupancy. In this point, it is crucial to remind that some of these features did not contain adequate information, so they were replaced with mean values or maybe suffered from misreported values.

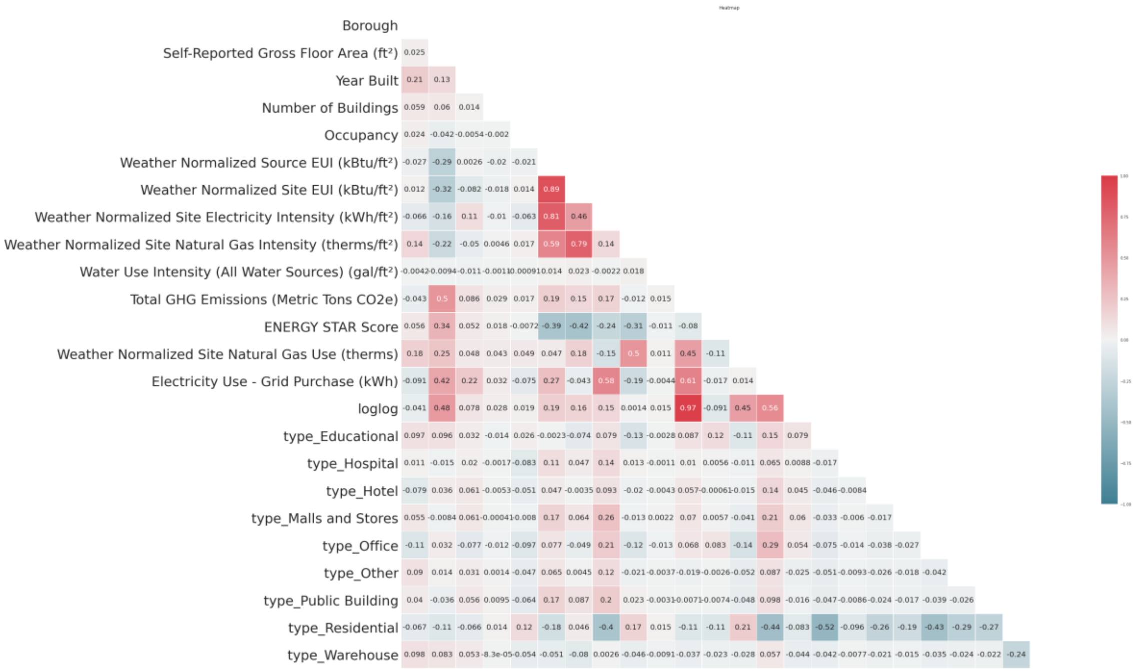


Figure 12: Heatmap illustrating correlations between all features.

5 Experimental Results

This chapter includes the results of the predictions performed for the two problems; the first one is about predicting the annual CO_2 emissions (in metric tons emitted) using regression and the second one determines which buildings comply to emissions limits via classification.

5.1 Predictions for annual GHG emissions

Using regression on our data fields that were kept for analysis, the following results are achieved (Figure 13). Before applying any machine learning algorithm, a train test split was performed, keeping 25% of our data for testing to evaluate our predictions. Also, data was standardized using Standard Scaler from Scikit- Learn package. The first algorithm to be examined is Random Forest, which is one of the most powerful algorithms across the Decision Trees family.

In addition, XG Boost Regressor is used which is a tree based boosting algorithm as well as Cat Boost Regressor which is a gradient boosting algorithm. Additionally, Artificial Neural Networks are used, as they are considered one of the most powerful predictors in the bibliography [19]. Three evaluation metrics were used, Mean Absolute Error (MAE), Mean Squared Error (MSE) and Root Mean Squared Error (RMSE). For the purpose of this study, we mainly focused on minimizing RMSE, because RMSE can be more useful when large errors are undesirable. Figure 13 illustrates the RMSE scores after performing a 5-fold cross validation to evaluate our models.

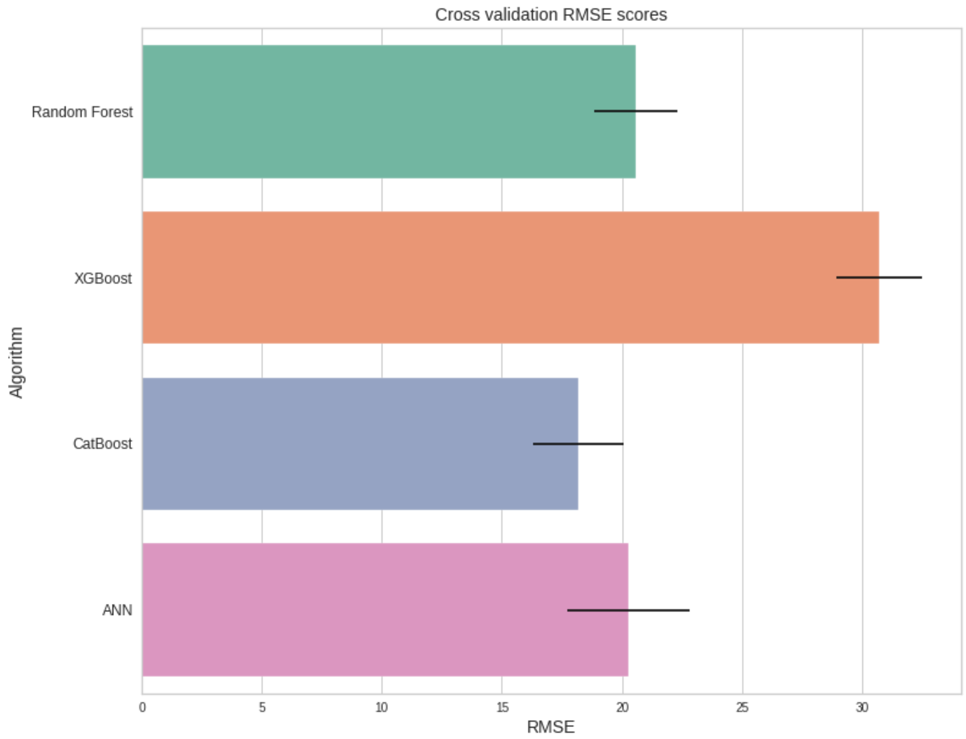


Figure 13: Cross validation RMSE scores.

As we observe, CatBoost gives the best score among all algorithms used. ANNs outperform Random Forest and XGBoost, while XGBoost presents a significant difference on all metric scores. The ANN model used in the study is a feed-forward MLP, composed of 5 layers in total and thus 3 hidden layers. The number of hidden neurons was chosen to be 45 by ‘trial-and-error’ (see Regression Appendix). Also, a Rectified Linear Unit (ReLu) activation function is applied on the hidden layers and a linear function for the output. Training is stopped when the maximum number of epochs is reached, which is 100.

Figure 14 illustrates feature importance using Random Forest. The most significant feature for CO_2 emissions is gross floor area which indicates that the intuition that building size affects its energy consumption and emissions is confirmed. Indeed, in [1] where the LL84 dataset was used to predict electricity and natural gas use, log building size which is the natural logarithm of gross floor area of each property is one of the six more powerful predictors chosen after applying the stepwise feature selection process.

In our case, the four most important attributes after gross floor area are: Weather normalized site EUI, Normalized Natural Gas Use, Electricity Use and Weather normalized source EUI. As we can see, year of construction and building type do not affect greenhouse gas emissions, neither the Energy Star score. At this point, it is essential to clarify that the dataset used is self-reported, as mentioned in the previous section, so many entries may be misreported, or missing and replaced by a mean value. So, these attributes could have been more powerful to predict CO_2 emissions if they contained actual values for each property. Also, almost 75% of the buildings examined are residential, so this imbalance may affect the importance of property type.

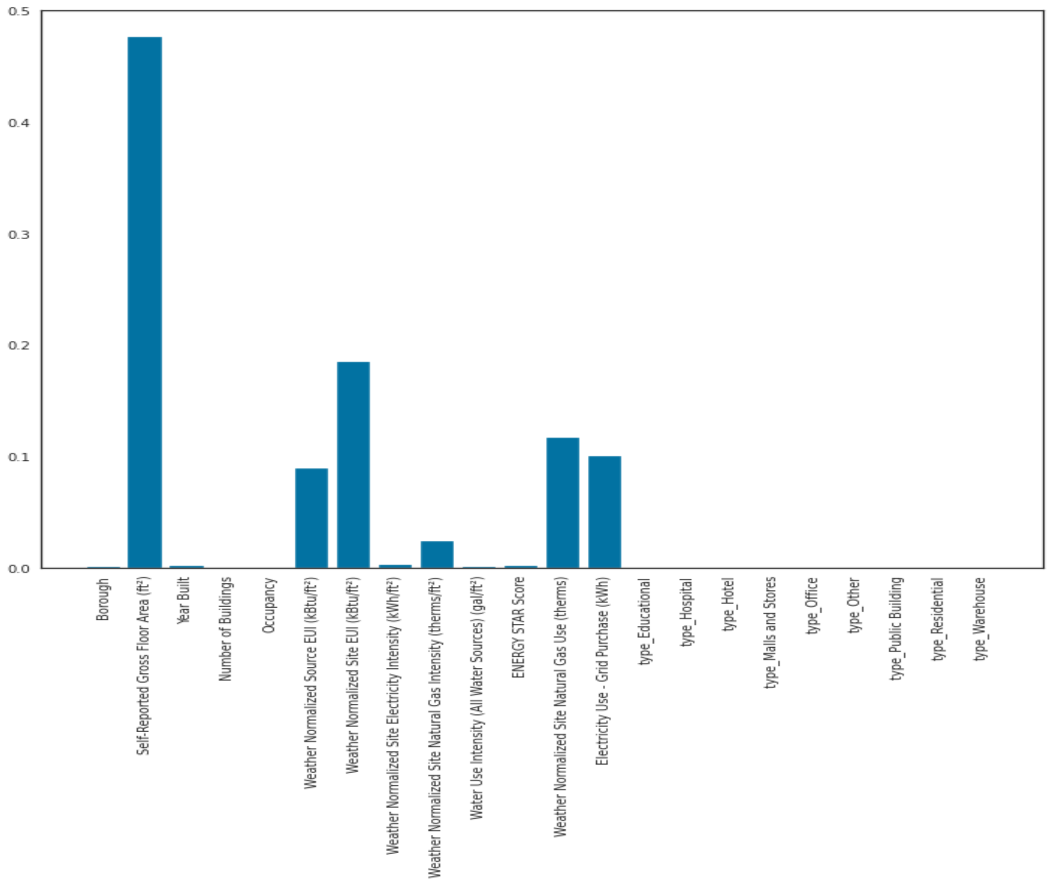


Figure 14: Feature importance using Random Forest

In predictive analytics, hyperparameter tuning is a step that is rarely omitted from a complete analysis procedure. Generally, default values perform well, but hyper tuning can lead to more accurate results. By reviewing the documentation of each algorithm but also the bibliography, we tried to find the right parameter grid in order to improve our models. Table 4 shows which parameters were chosen to be examined and which were the final selections after performing a grid search using GridSearchCV from the model selection Scikit-Learn package.

Table 4: The initial grid and final selections for each of the examined hyperparameters

Model		**Hyperparameters**		
RF		**n_estimators**	**min_samples_leaf**	**min_samples_split**
	Initial grid	100,200,300,500,1000	1, 2	1, 2
	Final selection	1000	1	2
XGBoost		**n_estimators**	**min_samples_leaf**	**min_samples_split**
	Initial grid	100,200,300,500,1000	1, 2	1, 2
	Final selection	1000	1	1
CatBoost		**n_estimators**	**depth**	**12_leaf_reg**
	Initial grid	100,200,300,500,1000	2, 4, 6, 8, 10	1, 2, 3
	Final selection	1000	10	1
ANN		**epochs**	**batch_size**	**optimizer**
	Initial grid	100,500,1000	10,25,32	adam, rmsprop
	Final selection	1000	10	adam

By observing the final selections for all algorithms, we can easily detect several similarities among them. More precisely, for all tree- based algorithms 1000 number of trees (estimators) were chosen which indicates that the default value of 100 estimators was not adequate to produce satisfactory results. Also, the minimum number of samples required for leaf nodes is equal to 1 for both Random Forest and XGB. Then, a very time-consuming grid search was conducted to discover the best hyperparameters for ANNs. The results have shown that the ideal number of epochs is 1000, the batch size equal to 10 and an ‘adam’ optimizer. Adam optimization is a stochastic gradient decent method based on adaptive estimation of first and second order moments [4]. Figure 15 illustrates the progress in MSE validation scores by increasing the number of epochs of ANNs training phase. The final RMSE scores are shown in Table 5 below. The improvement for all models is obvious, while ANNs gave the best RMSE score.

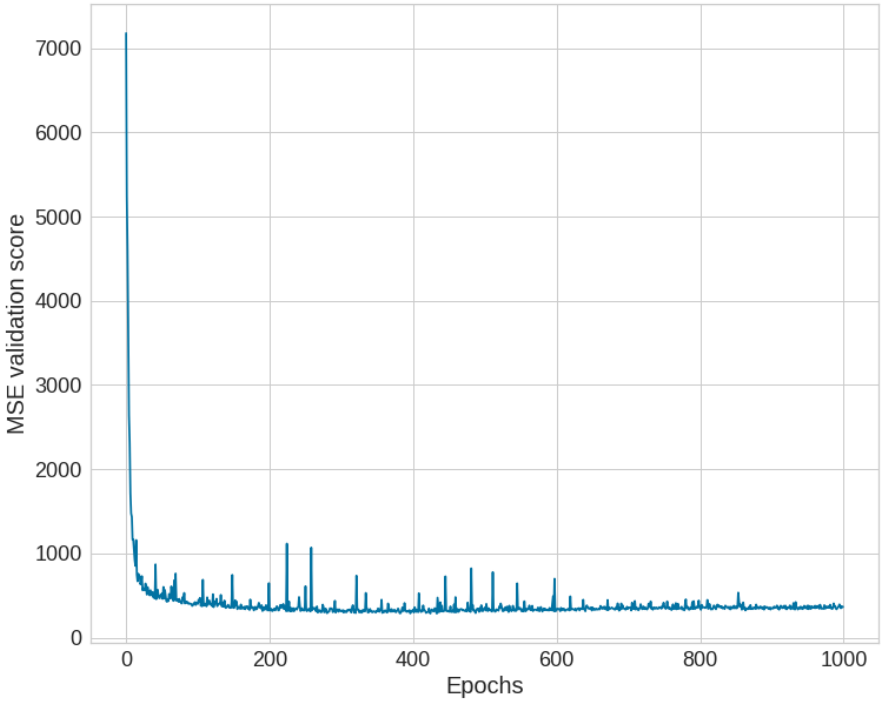


Figure 15: Cross validation MSE scores according to number of epochs

Table 5: Regression results after the selection of the optimal hyperparameters

	Random Forest	**XGBoost**	**Catboost**	**ANN**
RMSE	19,67	19,82	17,25	**15,69**

5.2 Predicting compliance

After forecasting the metric tons of CO_2 emitted from buildings annually, there is a need to predict if their environmental footprint exceeds the acceptable limit. Environmental regulations not only impact building owners, who must pay a fine per excessive metric ton, but also aid achieving global goals which are crucial for reducing greenhouse gas emissions. The goal here, is to predict compliance for properties contained in the LL84 dataset, using the LL97 carbon limits for each building type.

So far, the constructed models aimed to predict the metric tons of greenhouse gas emitted annually through regression. Now, the possible results of our predictions are two values: yes or no; yes for compliance and no for the opposite, thus a binary classification problem is examined. The preprocessing phase does not differ from the one we performed for the regression problem. A new data field was constructed, which stores the total GHG emissions in kilograms CO_2 in order to be compatible with LL97 compliance limits. Then, the limits for all building types were calculated by multiplying the limits per square feet with self-reported gross floor area. If total CO_2 emissions are less or equal to this boundary, this property complies to LL97 legislation, so the target label created is equal to 1, elsewhere the target label is set to 0. Figure 16 shows a diagram of this procedure.

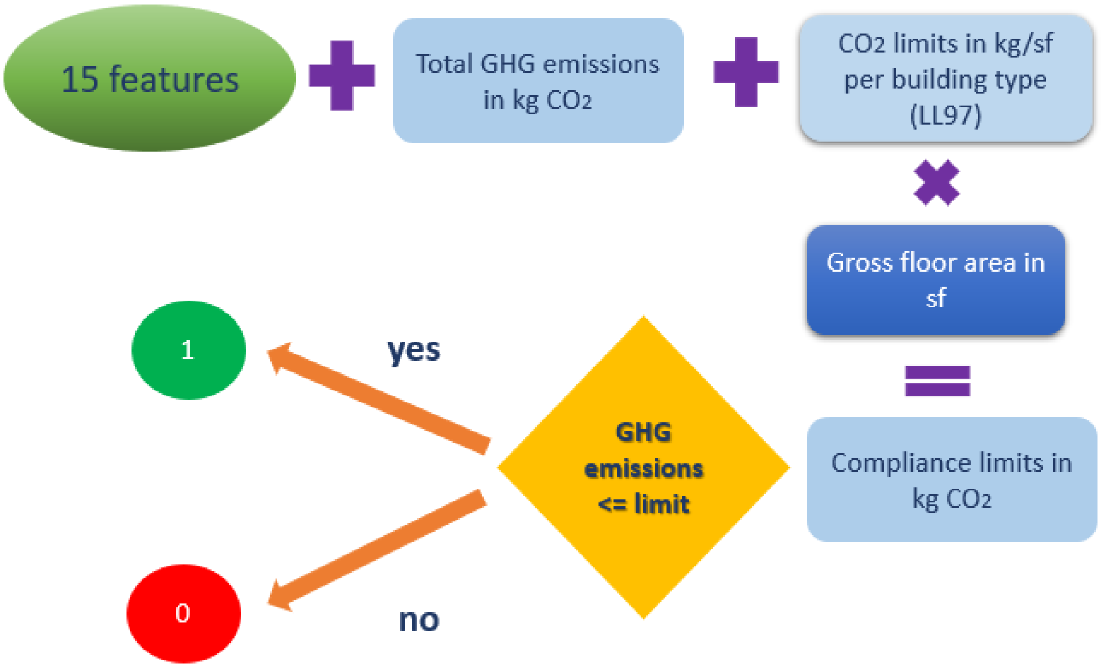

15 features
Total GHG emissions in kg CO2
CO2 limits in kg/sf per building type (LL97)
Gross floor area in sf
Compliance limits in kg CO2
GHG emissions <= limit
yes
no
1
0

Figure 16: Features used for the classification problems

Figure 17 illustrates the percentage of buildings belonging to class “yes” and to class “no”. We can easily observe that the imbalance is significant, as almost 80% of properties do not exceed the acceptable boundaries for greenhouse gas emissions, concerning the regulation for the period 2024-2029. The same imbalance is observed for years 2030-2034 in Figure 18, but this time almost 80% of buildings do not comply to environmental standards. Not surprisingly, in the future environmental regulations will become stricter, as the need to decarbonize all sectors of human activity will grow.

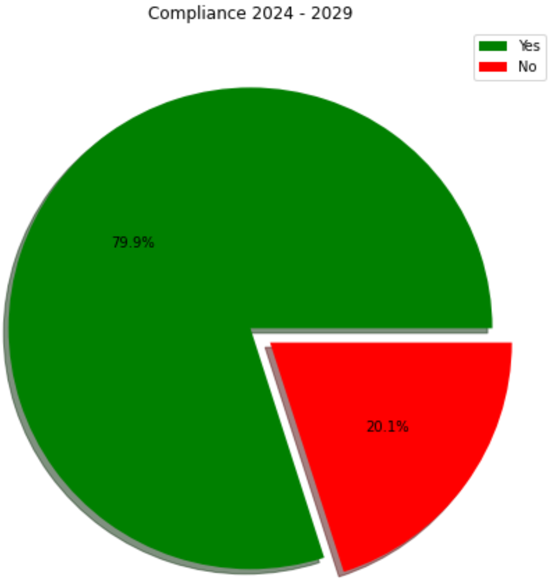


Figure 17: Target labels for compliance in LL97 for the period 2024-2029.

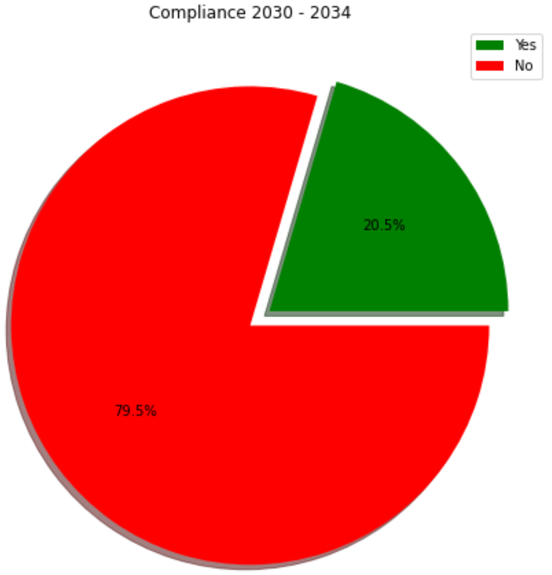


Figure 18: Target labels for compliance in LL97 for the period 2030-2034.

5.2.1 Predictions for 2024-2029

For this binary classification problem, we exclude the feature “self-reported gross floor area” because it was used to calculate the limits for compliance and thus it will affect our predictions. It is essential to note that the features kept for prediction are the same that have been used for the regression problem, minus one which is the gross floor area of the property that was explained above. So, the goal here is to predict compliance for buildings according to their characteristics and their energy use. Table 6 shows the fields used for predictions.

Table 6: Features used for analysis for the classification problems

Borough	**Weather Normalized Site Electicity Intensity (kWh/ft^2)**
Primary Property Type-Self Selected	**Weather Normalized Site Natural Gas Intensity (therms/ft^2)**
Year Built	**Water Use Intensity (All Water Sources) (gal/ft^2)**
Number of Buildings	**Total GHG Emissions (Metric Tons CO2e)**
Occupancy	**ENERGY STAR Score**
Weather Normalized Source EUI (kBtu/ft^2)	**Weather Normalized Site Natural Gas Use (therms)**
Weather Normalized Site EUI (kBtu/ft^2)	**Electricity Use - Grid Purchase (kWh)**

As it was mentioned before, the target variable takes two possible values 1 or 0 for properties which do or do not comply for this specific period, respectively. A train/test split was conducted keeping 25% of our data to evaluate the predictions. In addition, standardization was performed both on the train and test set to enhance our predictions' accuracy. We used the accuracy score from the Scikit-Learn package to evaluate our predictions.

A stratified 5-fold cross validation was performed to evaluate the predictions quality. Stratified cross validation was used because a significant class imbalance was observed and mentioned previously. Three algorithms were used for prediction; Random Forest, XGBoost and CatBoost, while ANNs were left out of the analysis to reduce complexity and execution time. Also, by looking at Figure 19, all algorithms performed well, and cross-validation accuracy scores were impressively high. CatBoost and Random Forest results seem almost identical, while XGBoost gives the lowest accuracy score.

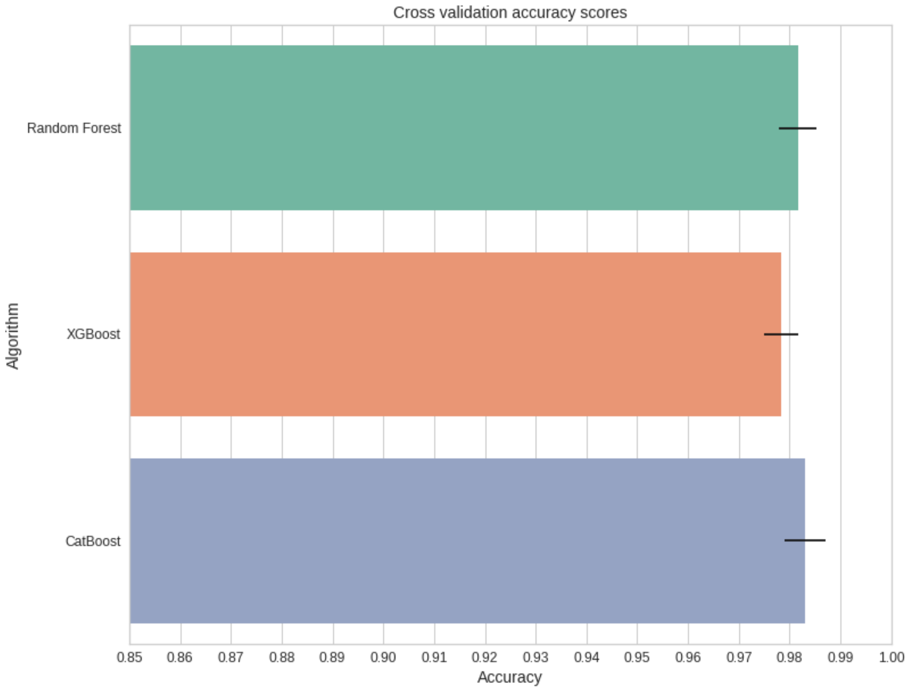


Figure 19: Cross validation accuracy scores for Random Forest, XGBoost and CatBoost.

Figure 20 shows which features are more important for the classification problem, using as base predictor the Random Forest classifier. The most significant features are Weather normalized Site EUI and Weather normalized Source EUI. Also, natural gas intensity and energy star seem to be important for our predictions.

Comparing this figure with the feature importance for the regression problem analyzed previously, source and site EUI were significant predictors in both regression and classification problem. As it was mentioned, gross floor area, which was the most important attribute for predicting CO_2 emissions through regression was eliminated from predicting compliance because it was used to calculate the limits and thus it would distort our predictions.

Surprisingly, energy star score tends to be more important here than previously and the opposite happens for electricity use. It is obvious that site EUI importance score is by far higher than source EUI in both cases. An explanation for that could be the fact that site energy use intensity is the amount of heat and electricity consumed by a building as reflected in utility bills. On the contrary, source EUI represents the total amount of raw fuel that is required to operate the building and it incorporates all transmission, delivery, and production losses.

Consequently, it is more likely that building owners are more aware of their site energy use by looking at their bills, but more unlikely to have calculated their source energy use properly. So, site EUI tends to be more reliable for our predictions, because it has a lower possibility to be misrecorded in the LL84 dataset. However, the Environmental Protection Agency (EPA) suggested that source energy is the most equitable unit of evaluation and provides a complete assessment of energy efficiency in a building [3].

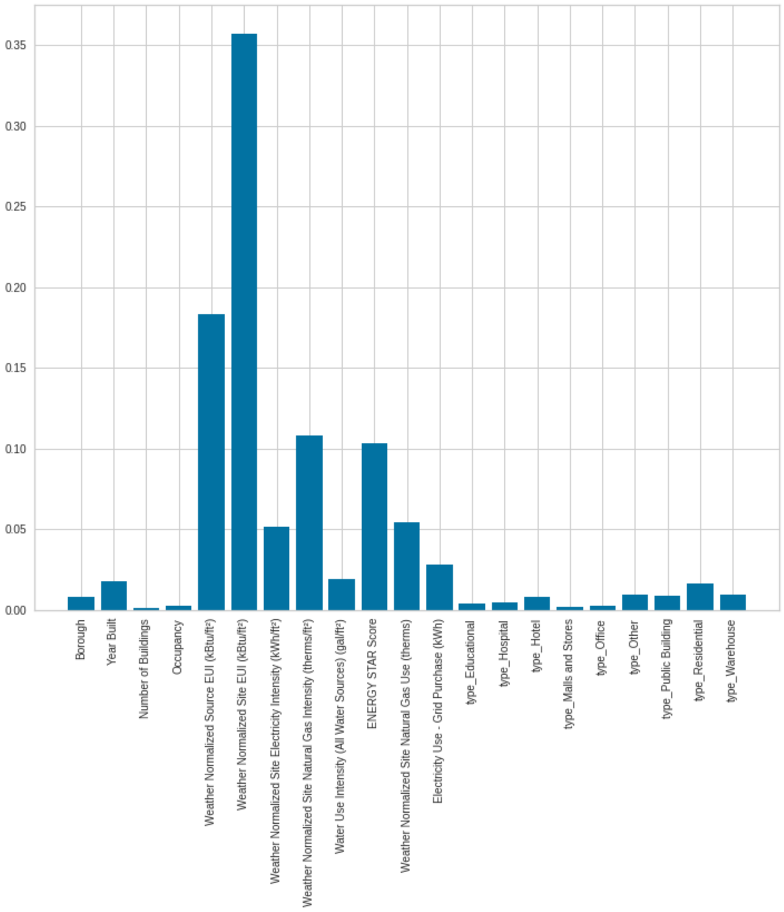


Figure 20: Feature importance using Random Forest classifier

After testing the models with their default values, a grid search was conducted to determine if there are any hyperparameters which will enhance model accuracy. Usually default values perform well, but improvement is always desirable. For Random Forest and XGBoost we focused on tuning the number of trees (‘n_estimators’), ‘min_samples_leaf’ and ‘min_samples_split’. For CatBoost classifier, ‘n_estimators’, ‘depth’ and ‘12_leaf_reg’ were examined. The starting values and final selections for hyperparameters of the models are shown in Table 7 below. Any hyperparameters not displayed were not changed and thus left at the defaults.

Table 7: The initial grid and final selections for each of the chosen hyperparameters

Model		**Hyperparameters**		
RF		**n_estimators**	**min_samples_leaf**	**min_samples_split**
	Initial grid	50, 80, 100, 120, 140, 200	1, 2	1, 2
	Final selection	50	1	2
XGBoost		**n_estimators**	**min_samples_leaf**	**min_samples_split**
	Initial grid	50, 80, 100, 120, 140, 200	1, 2	1, 2
	Final selection	200	1	1
CatBoost		**n_estimators**	**depth**	**12_leaf_reg**
	Initial grid	50, 80, 100, 120, 140, 200	2, 4, 6, 8, 10	1, 2, 3
	Final selection	200	10	1

Table 8 illustrates the models' prediction performance tested on unknown data. Random Forest is the most powerful predictor with accuracy score 98,7%. XGB and CatBoost have also performed well with small differences between them. Random Forest has shown a significant improvement after choosing the appropriate hyperparameters, while CatBoost score is slightly higher than before.

Table 8: Classification results for the period 2024-2029

2024-2029	**Random Forest**	**XGBoost**	**Catboost**
Accuracy	**0,9870**	0,9835	0,9857

5.2.2 Predictions for 2030-2034

The acceptable CO_2 limits emitted from buildings for the period 2030-2034 are much lower than the limits of the previous period examined, but the procedure is almost identical. Specifically, gross floor area was excluded from the analysis for the same reason mentioned before and the same algorithms were tested to compare the accuracies between these different periods. As previously, a stratified 5-fold cross validation was performed to evaluate the models. The mean accuracy score obtained is illustrated in Figure 21. Observing models' accuracy CatBoost and Random Forest perform almost identically, but XGBoost is slightly worse. However, the difference between all algorithms is less than 1% which is not significant. In Figure 22, we notice the same pattern concerning feature importance, as expected.

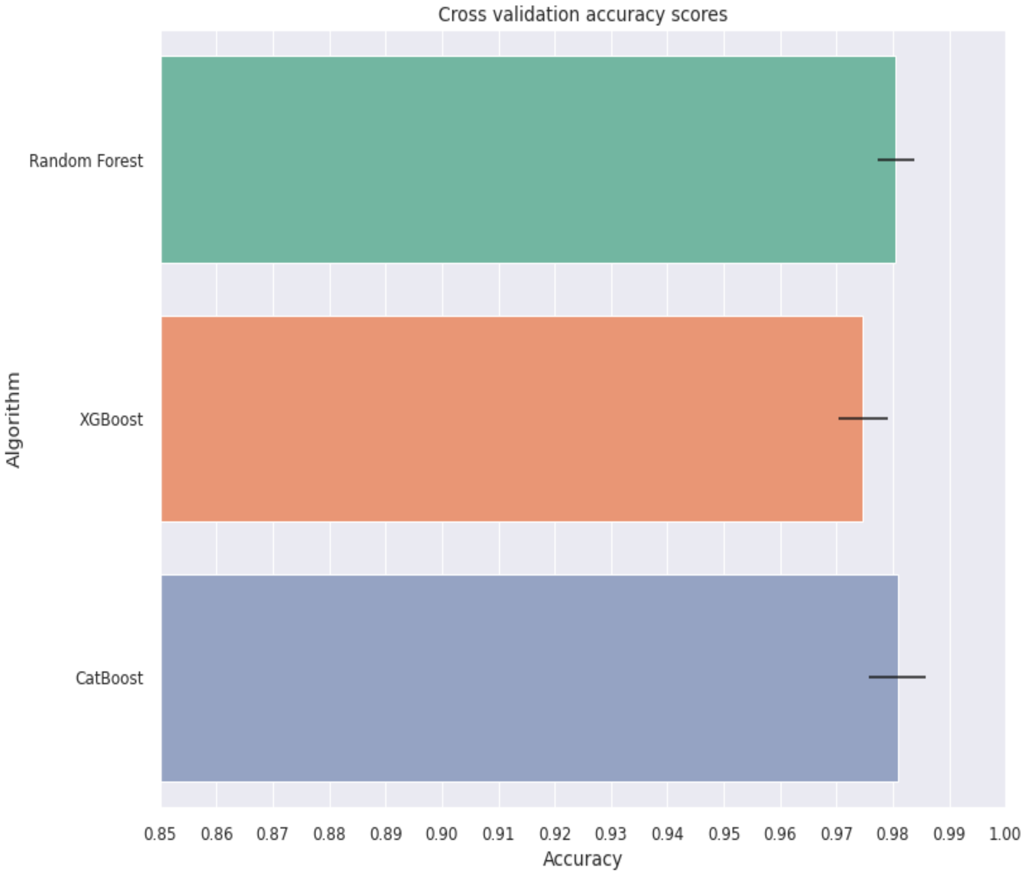


Figure 21: Cross validation accuracy scores for Random Forest, XGBoost and CatBoost.

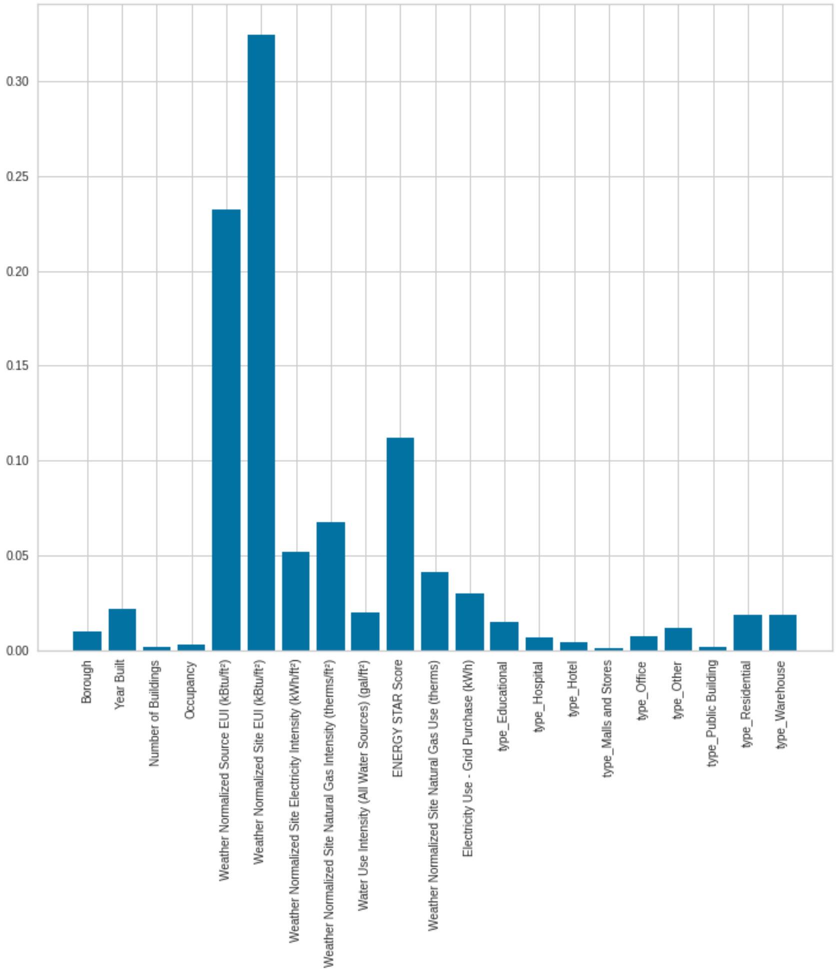


Figure 22: Feature importance using Random Forest classifier

Then, a grid search was conducted to discover the best hyperparameters for our problem. Again, we examine 'n_estimators', 'min_samples_leaf' and 'min_samples_split' for Random Forest and XGB, and the parameters 'depth', '12_leaf_reg', 'n_estimators' for CatBoost classifier, respectively. The initial values tested and the final selections for the hyperparameters are shown in Table 9.

The final results are shown in Table 10 below. As we can see, the accuracy scores have improved after choosing the right hyperparameters and CatBoost slightly outperforms the other two algorithms tested. The displayed results indicate the accuracy scores obtained by making predictions to our test set.

Table 9: The initial grid and final selections for each of the chosen hyperparameter

Model		**Hyperparameters**		
RF		**n_estimators**	**min_samples_leaf**	**min_samples_split**
	Initial grid	50, 80, 100, 120, 140, 200	1, 2	1, 2
	Final selection	80	1	2
XGBoost		**n_estimators**	**min_samples_leaf**	**min_samples_split**
	Initial grid	50, 80, 100, 120, 140, 200	1, 2	1, 2
	Final selection	200	1	1
CatBoost		**n_estimators**	**depth**	**12_leaf_reg**
	Initial grid	50, 80, 100, 120, 140, 200	2, 4, 6, 8, 10	1, 2, 3
	Final selection	200	10	1

Table 10: Classification results for the period 2030-2034

2030-2034	**Random Forest**	**XGBoost**	**Catboost**
Accuracy	0,9810	0,9814	**0,9818**

6 Evaluation and Discussion

Emissions forecasting for the building sector is a complex problem, which requires detailed information about building characteristics and technologies, as well as a deep understanding of the domain. Although domain knowledge is not considered compulsory in predictive analytics, problems like the one that is approached in this dissertation could help give insights and explanations to the observations that arise from the whole process. The bibliography around building emissions is still limited, as most of the studies examine electrical load prediction. So, the purpose of this work was to fill this gap and aims to give useful information about buildings' environmental footprint.

This work aims to forecast annual greenhouse gas emissions from several large buildings reported and also predict if a certain building complies to environmental regulations. These two problems can be useful for different reasons, as the result of a regression problem is a continuous number, while a binary classification problem answers with yes or no. Therefore, predicting the environmental footprint through regression could help governments, engineers and decision makers have a clear picture of the amounts of CO_2 emitted from the building sector and thus take actions like retrofits, or energy efficiency measures, renewables etc.

The results from the classification problem for the two periods (2024-2029, 2030-2034) give insights about the percentage of buildings in New York City which falls into the acceptable boundaries and thus is considered as environmentally friendly. Most importantly, it is a tool for informing building owners about whether their property is ready to meet the environmental requirements and prepare them for any potential measures they need to take. Indeed, this is a zero-cost way to be aware of whether a building complies to the legislations or not, which in other cases would require energy benchmarking, which holds several limitations and can be time consuming.

6.1 Pre-processing and ML algorithm selection

As analyzed in detail in 4.1, a pre-processing step was performed before making any predictions. This step was very crucial for the analysis procedure because the dataset suffered from numerous missing values, as well as outliers. Therefore, a selection was made to keep only attributes that contained useful information and would contribute to our predictions. Most of the input variables are related to energy data for each property, like energy use intensity and electricity use, as well as data about fuel consumption and water use. Afterwards, buildings were grouped in nine building types to agree with the types given in LL97 tables. According to building type, an outlier detection process has been applied.

This approach aimed to detect any outliers by type to ensure that representative values from each building category are preserved. Then, categorical fields were encoded to fit into multiple machine learning algorithms. Finally, a brief exploratory data analysis revealed some interesting patterns about CO_2 emissions. Non-residential buildings have shown higher levels of CO_2, especially high-occupied buildings like hospitals, schools and universities or malls and retail stores. The main reason of increased emissions in these types of buildings is their need to adapt to their occupants' needs quickly, as well as their higher energy requirements. Also, aging buildings are less friendly to the environment than new constructions, because of their lack in energy efficient technologies and materials.

In this study, four different machine learning algorithms were examined, which exhibit different strengths and weaknesses. Generally, tree-based boosting or bagging algorithms perform remarkably well, even without choosing the best hyperparameters. In fact, Random Forest and CatBoost gave very good results both in regression and classification. Also, their execution time is only a few seconds or couple minutes when tuning. On the other hand, ANNs are strong predictors according to bibliography and it was obvious that outperformed in terms of RMSE scores. However, their execution time is longer compared with the other models used and trying to find the best hyperparameters required many hours even with a small parameter grid. The basic conclusion about the selection of the best algorithm is that the complexity of the model was not so large as to let any of the algorithms the chance to stand out. As it is clear from the results, the difference between them for the classification problem is lower than 1%.

6.2 Extracted knowledge

The most important results of this work are briefly listed below:

- The most important predictors for the regression problem are the gross floor area, source and site energy use intensity, electricity and natural gas use. That means that these characteristics could be the key in decarbonizing buildings, while building type and use do not play such significant role.
- ANNs outperformed all other algorithms tested, reaching the lowest RMSE score.
- Number of epochs affects RMSE results the most for ANNs. Indeed, the RMSE score for 1000 epochs of training was almost 5 tons of CO_2 lower than the score resulting from 100 epochs.
- Tree based algorithms' performance is affected from the number of trees. It is clearly noticed that all algorithms performed better when the number of estimators increased from the default value of 100 to 1000.
- For the classification problems, for the first period (2024-2029) almost 80% of the buildings contained in our dataset comply with LL97, while for the second period (2030-2034) only 20% of the buildings fulfill the requirements. This indicates the need to make a transition towards greener technologies and energy efficiency refurbishments in the next few years.
- For both periods examined, the most significant features were source and site EUI, reaching a higher importance score than the one observed at the regression problem. Also, electricity and natural gas use were significant attributes as before, and this time energy score was one of the strongest predictors for the two binary classification problems. As a result, building owners should try to improve their energy score for achieving emission compliance.
- Random Forest presented the highest accuracy score among all for the first period, while at the second period the best algorithm was CatBoost. However, the difference between all algorithms tested was marginal, so the performance was overall good.

6.3 Threats to validity

Most of the times, scientific works suffer from several threats that might affect the validity of the results. These threats may be caused by either the data acquisition procedure, the selection of specific methods or algorithms, or even by the nature of data itself. In our case, the dataset used was LL84 which was self- reported, thus the accuracy of the data contained is questionable, especially in fields where building owners are not familiar with, like source energy use intensity or occupation percentage.

Although a preprocessing and data cleaning procedure has been followed, still it was difficult to understand if all entries are correct or detect all anomalous ones. In addition, as it was mentioned in 4.1, several features have been eliminated from the analysis because they contained too many missing or non-valid values. The fields excluded would probably help the analysis and give more insights about buildings behavior and characteristics.

Also, the majority of the buildings examined were residential and the commercial buildings were very limited. This imbalance does not favor the results and makes it harder to draw conclusions about specific property types and their environmental footprint.

Finally, the selected metric for the classification problems was accuracy, which is not always the most appropriate metric especially when class imbalance is present. As it was observed, in both periods the ones were way more numerous than zeros or the opposite, so maybe another metric would be more appropriate like precision or recall.

7 Conclusions and Future Directions

This chapter summarizes the results and presents ideas for future work. This research focused on predicting greenhouse gas emissions caused by the building sector, focusing on single buildings and their characteristics. The purpose of the study was to analyze the factors affecting building emissions and fill the gap in the bibliography, which mainly focuses on energy performance and energy load forecasting.

7.1 Conclusions

Understanding building environmental footprint is a crucial component of improving urban sustainability plans, reach carbon reduction goals, as well as achieve higher levels of energy efficiency and comfort. Numerous cities and countries all over the world have already adopted energy use and carbon reduction plans, mainly focused on advancing energy efficiency for the existing building stock. To support these goals and comply to the regulations, the implementation of energy disclosure policies requiring buildings of a certain size or type to report their energy consumption, has created new streams of data. These data can provide useful information about urban energy dynamics, give insights about greenhouse gas emissions and aid growing data-driven strategies for achieving more efficient buildings with less or nearly zero carbon emissions.

The analysis presented here aims to predict the total greenhouse gas emissions of buildings using the LL84 self-reported energy disclosure data from properties in New York. Using the acceptable limits of carbon by building category, which are provided by a carbon reduction legislation applied in NYC, we tried to predict whether a building complies to the carbon law for two periods. In this problem we used the same dataset (LL84), which includes information about the building and its energy and water use. Using four machine learning algorithms for the regression problem and three for the classification problems, the results suggest that the data from LL84 sample can produce reasonably accurate predictions of carbon emissions across the city at a building scale.

Overall, we found little differences depending on the machine learning methods used, based on the resultant RMSE values for the regression part and the accuracy scores for the classification of the two periods examined. ANNs provide the most accurate predictions reaching the lowest RMSE score, while Random Forest and CatBoost are the best algorithms chosen for the classification problems, respectively. It is also observed that building size and energy use intensity play a major role in its environmental footprint.

Most samples represented multifamily housing buildings which made it difficult to conclude which building type or use contributes more to excessive emissions. However, the pattern revealed was that buildings that are highly occupied and consume more energy per square feet, such as malls, hospitals, schools or universities, and offices, tend to be less green than other building types. Also, in order to comply to carbon reduction regulations building owners will be obliged to take action to decarbonize their properties in the next few years.

The findings presented in this work can create new opportunities for data-driven environmental policies in cities and give more insights of how decarbonization goals can be achieved.

7.2 Future research directions

This work focuses on a complex problem which requires time-consuming analysis and multiple approaches. This dissertation tried to cover as many aspects as possible given the time and data available for such purpose. However, there is always space for improvement and need to explore more aspects in most studies.

Future studies should collect more energy disclosure data from previous years and maybe incorporate new data which will be publicly available by the end of the year. It is obvious that more data always lead to more accurate and reliable results. Also, data from different regions or cities along with weather information would provide a more comprehensive view of carbon emissions caused by the urban building stock. In this work, most of the buildings examined were residential, so it was difficult to draw conclusions about commercial properties, which need to be examined more, as their emissions were proved to be higher than dwellings.

In addition, future works could try more machine learning algorithms, as well as feature selection techniques, which could improve performance. Regarding ANNs implementation, a more detailed selection of hyperparameters is desirable to explore their dynamic in these types of problems.

Finally, it is essential to mention the importance of focusing on forecasting emissions, as there is little research on this specific field. Combining building with transportation data could also be an idea for future research, as the transportation sector accounts for a significant amount of urban carbon emissions and could be beneficial for city-scale level sustainability plans.

References

[1] Zhao, Hai-xiang, and Frédéric Magoulès. "A review on the prediction of building energy consumption." Renewable and Sustainable Energy Reviews 16.6 (2012): 3586-3592.

[2] 2050 long-term strategy - Climate Action - European Commission. Climate Action - European Commission, https://ec.europa.eu/clima/policies/strategies/2050_en

[3] Kontokosta, Constantine E. "Energy disclosure, market behavior, and the building data ecosystem." *Annals of the New York Academy of Sciences* 1295.1 (2013): 34-43.

[4] GBEE - Greener, Greater Buildings Plan - LL84: Benchmarking - About LL84, https://htt1.nyc.gov/html/gbee/html/plan/ll84_about.shtml

[5] NYC Carbon Emissions Bill Passed into Law - "Local Law 97" - What it means for commercial building owners.CodeGreen Solutions, https://codegreensolutions.com/nyc-carbon-emissions-bill-passed-into-law-local-law-97-what-it-means-for-commercial-building-owners/

[6] Calvillo, Christian F., Alvaro Sánchez-Miralles, and Jose Villar. "Energy management and planning in smart cities." *Renewable and Sustainable Energy Reviews* 55 (2016): 273-287.

[7] K. Christantonis, C. Tjortjis, A. Manos, D Filippidou and E. Christelis, 'Smart Cities Data Classification for Electricity Consumption & Traffic Prediction', Automatics & Software Enginery, 31(1), 2020

[8] Christantonis, Konstantinos, and Christos Tjortjis. "Data mining for smart cities: predicting electricity consumption by classification." *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*. IEEE, 2019.

[9] Moser, Mary Anne. "What is smart about the smart communities movement." *EJournal, 10* 11.1 (2001): 1-11.

[10] Zekić-Sušac, Marijana, Saša Mitrović, and Adela Has. "Machine learning based system for managing energy efficiency of public sector as an approach towards smart cities." *International Journal of Information Management* (2020): 102074.

[11] Snoonian, Deborah. "Smart buildings." *IEEE spectrum* 40.8 (2003): 18-23.

[12] Weng, Thomas, and Yuvraj Agarwal. "From buildings to smart buildings—sensing and actuation to improve energy efficiency." *IEEE Design & Test of Computers* 29.4 (2012): 36-44.

[13] Lazarova-Molnar, Sanja, Hamid Reza Shaker, and Nader Mohamed. "Fault detection and diagnosis for smart buildings: State of the art, trends and challenges." *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*. IEEE, 2016.

[14] Kaur, Maninder Jeet, and Piyush Maheshwari. "Building smart cities applications using IoT and cloud-based architectures." *2016 International Conference on Industrial Informatics and Computer Systems (CIICS)*. IEEE, 2016.

[15] Buckman, Alex H., Martin Mayfield, and Stephen BM Beck. "What is a smart building?" *Smart and Sustainable Built Environment* (2014).

[16] Hoy, Matthew B. "Smart buildings: an introduction to the library of the future." *Medical reference services quarterly* 35.3 (2016): 326-331.

[17] Borgia, Eleonora. "The Internet of Things vision: Key features, applications and open issues." *Computer Communications* 54 (2014): 1-31.

[18] Ferrández-Pastor, Francisco-Javier, et al. "Deployment of IoT edge and fog computing technologies to develop smart building services." *Sustainability* 10.11 (2018): 3832.

[19] Seyedzadeh, Saleh, et al. "Machine learning for estimation of building energy consumption and performance: a review." *Visualization in Engineering* 6.1 (2018): 5.

[20] Fan, Cheng, et al. "Unsupervised data analytics in mining big building operational data for energy efficiency enhancement: A review." *Energy and Buildings* 159 (2018): 296-308.

[21] Yang, Junjing, et al. "k-Shape clustering algorithm for building energy usage patterns analysis and forecasting model accuracy improvement." *Energy and Buildings* 146 (2017): 27-37.

[22] Papadopoulos, Sokratis, Bartosz Bonczak, and Constantine E. Kontokosta. "Pattern recognition in building energy performance over time using energy benchmarking data." *Applied Energy* 221 (2018): 576-586.

[23] Fan, Cheng, Fu Xiao, and Shengwei Wang. "Development of prediction models for next-day building energy consumption and peak power demand using data mining techniques." *Applied Energy* 127 (2014): 1-10.

[24] Baker, Keith J., and R. Mark Rylatt. "Improving the prediction of UK domestic energy-demand using annual consumption-data." *Applied Energy* 85.6 (2008): 475-482.

[25] Dong, Bing, Cheng Cao, and Siew Eang Lee. "Applying support vector machines to predict building energy consumption in tropical region." *Energy and Buildings* 37.5 (2005): 545-553.

[26] Jain, Rishee K., et al. "Forecasting energy consumption of multi-family residential buildings using support vector regression: Investigating the impact of temporal and spatial monitoring granularity on performance accuracy." *Applied Energy* 123 (2014): 168-178.

[27] Solomon, David M., et al. "Forecasting energy demand in large commercial buildings using support vector machine regression." (2011).

[28] Mena, R., et al. "A prediction model based on neural networks for the energy consumption of a bioclimatic building." *Energy and Buildings* 82 (2014): 142-155.

[29] Karatasou, S., M. Santamouris, and V. Geros. "Modeling and predicting building's energy use with artificial neural networks: Methods and results." *Energy and buildings* 38.8 (2006): 949-958.

[30] Ahmad, Muhammad Waseem, Monjur Mourshed, and Yacine Rezgui. "Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption." *Energy and Buildings* 147 (2017): 77-89.

[31] Pombeiro, Henrique, et al. "Comparative assessment of low-complexity models to predict electricity consumption in an institutional building: Linear regression vs. fuzzy modeling vs. neural networks." *Energy and Buildings* 146 (2017): 141-151.

[32] Wen, Lei, and Xiaoyu Yuan. "Forecasting CO_2 emissions in Chinas commercial department, through BP neural network based on random forest and PSO." *Science of The Total Environment* 718 (2020): 137194.

[33] Wu, Yihan, et al. "Mapping building carbon emissions within local climate zones in Shanghai." *Energy Procedia* 152 (2018): 815-822.

[34] Petcharat, Siriwarin, Supachart Chungpaibulpatana, and Pattana Rakkwamsuk. "Assessment of potential energy saving using cluster analysis: A case study of lighting systems in buildings." *Energy and Buildings* 52 (2012): 145-152.

[35] Kontokosta, Constantine E., and Christopher Tull. "A data-driven predictive model of city-scale energy use in buildings." *Applied energy* 197 (2017): 303-317.

[36] Gaitani, N., et al. "Using principal component and cluster analysis in the heating evaluation of the school building sector." *Applied Energy* 87.6 (2010): 2079-2086.

[37] Lara, Rigoberto Arambula, et al. "Energy audit of schools by means of cluster analysis." *Energy and Buildings* 95 (2015): 160-171.

[38] Marasco, Daniel E., and Constantine E. Kontokosta. "Applications of machine learning methods to identifying and predicting building retrofit opportunities." *Energy and Buildings* 128 (2016): 431-441.

[39] Ascione, Fabrizio, et al. "Artificial neural networks to predict energy performance and retrofit scenarios for any member of a building category: A novel approach." *Energy* 118 (2017): 999-1017.

[40] Rahman, Aowabin, and Amanda D. Smith. "Predicting fuel consumption for commercial buildings with machine learning algorithms." *Energy and Buildings* 152 (2017): 341-358.

[41] Khayatian, Fazel, and Luca Sarto. "Application of neural networks for evaluating energy performance certificates of residential buildings." Energy and Buildings 125 (2016): 45-54.

[42] Zou, Han, et al. "Machine learning empowered occupancy sensing for smart buildings." (2019).

[43] Alawadi, Sadi, et al. "A comparison of machine learning algorithms for forecasting indoor temperature in smart buildings." *Energy Systems* (2020): 1-17.

[44] Abdelkader, E., A. Al-Sakkaf, and R. Ahmed. "A comprehensive comparative analysis of machine learning models for predicting heating and cooling loads." *Decision Science Letters* 9.3 (2020): 409-420.

[45] West, Samuel R., et al. "Automated fault detection and diagnosis of HVAC subsystems using statistical machine learning." *12th International Conference of the International Building Performance Simulation Association*. 2011.

[46] Holcomb, Daniel, Wenchao Li, and Sanjit A. Seshia. "Algorithms for green buildings: Learning-based techniques for energy prediction and fault diagnosis." *Google Scholar, UCB/EECS-2009-138* (2009).

[47] Liu, Yang, et al. "Energy consumption prediction and diagnosis of public buildings based on support vector machine learning: A case study in China." *Journal of Cleaner Production* 272 (2020): 122542.

[48] Araya, Daniel B., et al. "Collective contextual anomaly detection framework for smart buildings." *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016.

[49] Lazarova-Molnar, Sanja, Hamid Reza Shaker, and Nader Mohamed. "Fault detection and diagnosis for smart buildings: State of the art, trends and challenges." *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*. IEEE, 2016.

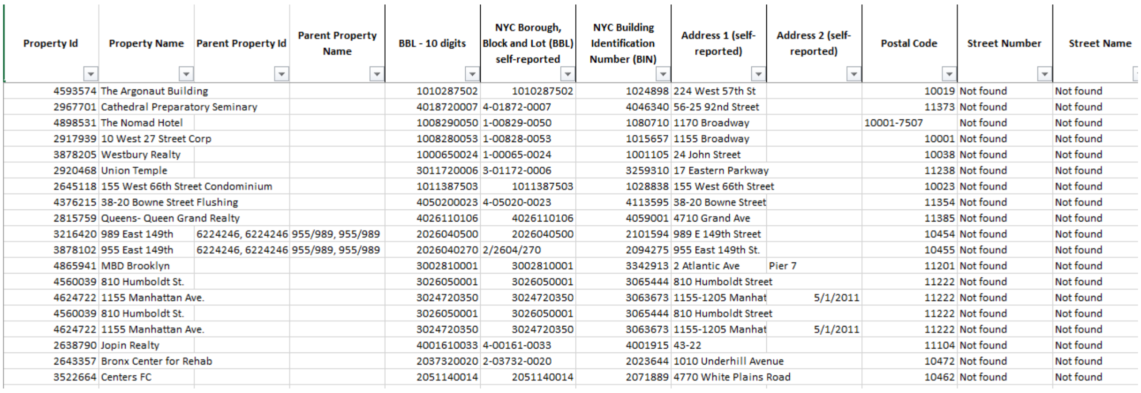
[50] Hong, Sung-Min, et al. "Improved benchmarking comparability for energy consumption in schools." *Building Research & Information* 42.1 (2014): 47-61.

[51] Santamouris, M., et al. "Using intelligent clustering techniques to classify the energy performance of school buildings." *Energy and buildings* 39.1 (2007): 45-51.

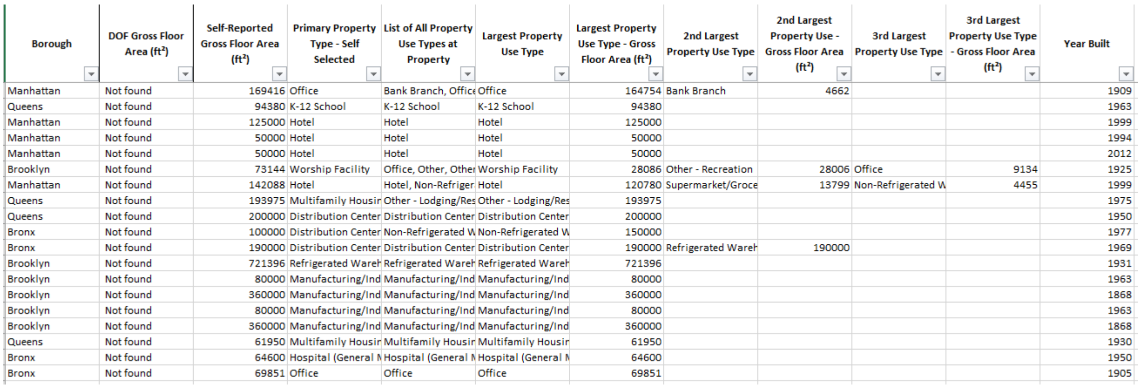
[52] Gao, Xuefeng, and Ali Malkawi. "A new methodology for building energy performance benchmarking: An approach based on intelligent clustering algorithm." *Energy and Buildings* 84 (2014): 607-616.

Appendices

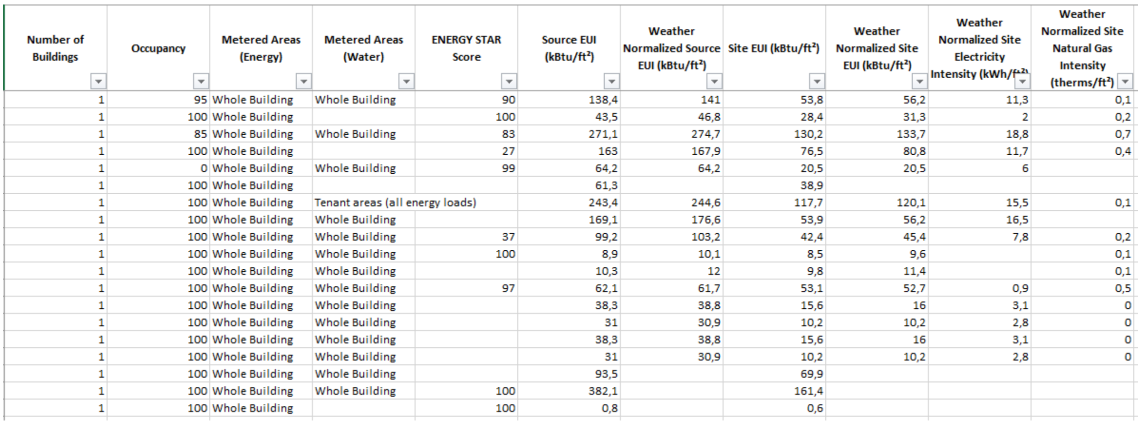
DATA SAMPLES



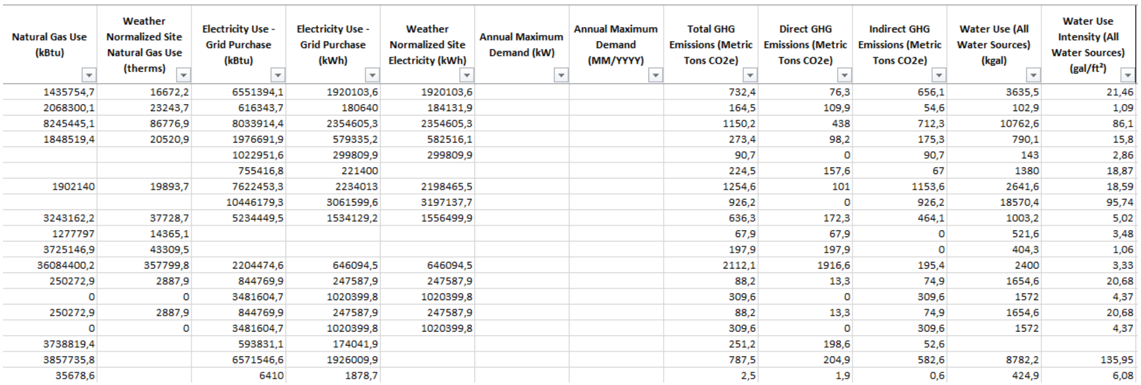
Property Id	Property Name	Parent Property Id	Parent Property Name	BBL - 10 digits	NYC Borough, Block and Lot (BBL) self-reported	NYC Building Identification Number (BIN)	Address 1 (self-reported)	Address 2 (self-reported)	Postal Code	Street Number	Street Name
4593574	The Argonaut Building			1010287502	1010287502	1024898	224 West 57th St		10019	Not found	Not found
2967701	Cathedral Preparatory Seminary			4018720007	4-01872-0007	4046340	56-25 92nd Street		11373	Not found	Not found
4898531	The Nomad Hotel			1008290050	1-00829-0050	1080710	1170 Broadway		10001-7507	Not found	Not found
2917939	10 West 27 Street Corp			1008280053	1-00828-0053	1015657	1155 Broadway		10001	Not found	Not found
3878205	Westbury Realty			1000650024	1-00065-0024	1001105	24 John Street		10038	Not found	Not found
2920468	Union Temple			3011720006	3-01172-0006	3259310	17 Eastern Parkway		11238	Not found	Not found
2645118	155 West 66th Street Condominium			1011387503	1011387503	1028838	155 West 66th Street		10023	Not found	Not found
4376215	38-20 Bowne Street Flushing			4050200023	4-05020-0023	4113595	38-20 Bowne Street		11354	Not found	Not found
2815759	Queens- Queen Grand Realty			4026110106	4026110106	4059001	4710 Grand Ave		11385	Not found	Not found
3216420	989 East 149th	6224246, 6224246	955/989, 955/989	2026040500	2026040500	2101594	989 E 149th Street		10454	Not found	Not found
3878102	955 East 149th	6224246, 6224246	955/989, 955/989	2026040270	2/2604/270	2094275	955 East 149th St.		10455	Not found	Not found
4865941	MBD Brooklyn			3002810001	3002810001	3342913	2 Atlantic Ave	Pier 7	11201	Not found	Not found
4560039	810 Humboldt St.			3026050001	3026050001	3065444	810 Humboldt Street		11222	Not found	Not found
4624722	1155 Manhattan Ave.			3024720350	3024720350	3063673	1155-1205 Manhat	5/1/2011	11222	Not found	Not found
4560039	810 Humboldt St.			3026050001	3026050001	3065444	810 Humboldt Street		11222	Not found	Not found
4624722	1155 Manhattan Ave.			3024720350	3024720350	3063673	1155-1205 Manhat	5/1/2011	11222	Not found	Not found
2638790	Jopin Realty			4001610033	4-00161-0033	4001915	43-22		11104	Not found	Not found
2643357	Bronx Center for Rehab			2037320020	2-03732-0020	2023644	1010 Underhill Avenue		10472	Not found	Not found
3522664	Centers FC			2051140014	2051140014	2071889	4770 White Plains Road		10462	Not found	Not found



Borough	DOF Gross Floor Area (ft^2)	Self-Reported Gross Floor Area (ft^2)	Primary Property Type - Self Selected	List of All Property Use Types at Property	Largest Property Use Type	Largest Property Use Type - Gross Floor Area (ft^2)	2nd Largest Property Use Type	2nd Largest Property Use - Gross Floor Area (ft^2)	3rd Largest Property Use Type	3rd Largest Property Use Type - Gross Floor Area (ft^2)	Year Built
Manhattan	Not found	169416	Office	Bank Branch, Office	Office	164754	Bank Branch	4662			1909
Queens	Not found	94380	K-12 School	K-12 School	K-12 School	94380					1963
Manhattan	Not found	125000	Hotel	Hotel	Hotel	125000					1999
Manhattan	Not found	50000	Hotel	Hotel	Hotel	50000					1994
Manhattan	Not found	50000	Hotel	Hotel	Hotel	50000					2012
Brooklyn	Not found	73144	Worship Facility	Office, Other, Other	Worship Facility	28086	Other - Recreation	28006	Office	9134	1925
Manhattan	Not found	142088	Hotel	Hotel, Non-Refriger.	Hotel	120780	Supermarket/Groce	13799	Non-Refrigerated W	4455	1999
Queens	Not found	193975	Multifamily Housir	Other - Lodging/Res	Other - Lodging/Res	193975					1975
Queens	Not found	200000	Distribution Center	Distribution Center	Distribution Center	200000					1950
Bronx	Not found	100000	Distribution Center	Non-Refrigerated W	Non-Refrigerated W	150000					1977
Bronx	Not found	190000	Distribution Center	Distribution Center	Distribution Center	190000	Refrigerated Wareh	190000			1969
Brooklyn	Not found	721396	Refrigerated Wareh	Refrigerated Wareh	Refrigerated Wareh	721396					1931
Brooklyn	Not found	80000	Manufacturing/Ind	Manufacturing/Ind	Manufacturing/Ind	80000					1963
Brooklyn	Not found	360000	Manufacturing/Ind	Manufacturing/Ind	Manufacturing/Ind	360000					1868
Brooklyn	Not found	80000	Manufacturing/Ind	Manufacturing/Ind	Manufacturing/Ind	80000					1963
Brooklyn	Not found	360000	Manufacturing/Ind	Manufacturing/Ind	Manufacturing/Ind	360000					1868
Queens	Not found	61950	Multifamily Housir	Multifamily Housir	Multifamily Housir	61950					1930
Bronx	Not found	64600	Hospital (General M	Hospital (General M	Hospital (General M	64600					1950
Bronx	Not found	69851	Office	Office	Office	69851					1905



Number of Buildings	Occupancy	Metered Areas (Energy)	Metered Areas (Water)	ENERGY STAR Score	Source EUI (kBtu/ft^2)	Weather Normalized Source EUI (kBtu/ft^2)	Site EUI (kBtu/ft^2)	Weather Normalized Site EUI (kBtu/ft^2)	Weather Normalized Site Electricity Intensity (kWh/ft^2)	Weather Normalized Site Natural Gas Intensity (therms/ft^2)
1	95	Whole Building	Whole Building	90	138,4	141	53,8	56,2	11,3	0,1
1	100	Whole Building		100	43,5	46,8	28,4	31,3	2	0,2
1	85	Whole Building	Whole Building	83	271,1	274,7	130,2	133,7	18,8	0,7
1	100	Whole Building		27	163	167,9	76,5	80,8	11,7	0,4
1	0	Whole Building	Whole Building	99	64,2	64,2	20,5	20,5	6	
1	100	Whole Building			61,3		38,9			
1	100	Whole Building	Tenant areas (all energy loads)		243,4	244,6	117,7	120,1	15,5	0,1
1	100	Whole Building	Whole Building		169,1	176,6	53,9	56,2	16,5	
1	100	Whole Building	Whole Building	37	99,2	103,2	42,4	45,4	7,8	0,2
1	100	Whole Building	Whole Building	100	8,9	10,1	8,5	9,6		0,1
1	100	Whole Building	Whole Building		10,3	12	9,8	11,4		0,1
1	100	Whole Building	Whole Building	97	62,1	61,7	53,1	52,7	0,9	0,5
1	100	Whole Building	Whole Building		38,3	38,8	15,6	16	3,1	0
1	100	Whole Building	Whole Building		31	30,9	10,2	10,2	2,8	0
1	100	Whole Building	Whole Building		38,3	38,8	15,6	16	3,1	0
1	100	Whole Building	Whole Building		31	30,9	10,2	10,2	2,8	0
1	100	Whole Building	Whole Building		93,5		69,9			
1	100	Whole Building	Whole Building	100	382,1		161,4			
1	100	Whole Building		100	0,8		0,6			



Natural Gas Use (kBtu)	Weather Normalized Site Natural Gas Use (therms)	Electricity Use - Grid Purchase (kBtu)	Electricity Use - Grid Purchase (kWh)	Weather Normalized Site Electricity (kWh)	Annual Maximum Demand (kW)	Annual Maximum Demand (MM/YYYY)	Total GHG Emissions (Metric Tons CO2e)	Direct GHG Emissions (Metric Tons CO2e)	Indirect GHG Emissions (Metric Tons CO2e)	Water Use (All Water Sources) (kgal)	Water Use Intensity (All Water Sources) (gal/ft^2)
1435754,7	16672,2	6551394,1	1920103,6	1920103,6			732,4	76,3	656,1	3635,5	21,46
2068300,1	23243,7	616343,7	180640	184131,9			164,5	109,9	54,6	102,9	1,09
8245445,1	86776,9	8033914,4	2354605,3	2354605,3			1150,2	438	712,3	10762,6	86,1
1848519,4	20520,9	1976691,9	579335,2	582516,1			273,4	98,2	175,3	790,1	15,8
		1022951,6	299809,9	299809,9			90,7	0	90,7	143	2,86
		755416,8	221400				224,5	157,6	67	1380	18,87
1902140	19893,7	7622453,3	2234013	2198465,5			1254,6	101	1153,6	2641,6	18,59
		10446179,3	3061599,6	3197137,7			926,2	0	926,2	18570,4	95,74
3243162,2	37728,7	5234449,5	1534129,2	1556499,9			636,3	172,3	464,1	1003,2	5,02
1277797	14365,1						67,9	67,9	0	521,6	3,48
3725146,9	43309,5						197,9	197,9	0	404,3	1,06
36084400,2	357799,8	2204474,6	646094,5	646094,5			2112,1	1916,6	195,4	2400	3,33
250272,9	2887,9	844769,9	247587,9	247587,9			88,2	13,3	74,9	1654,6	20,68
0	0	3481604,7	1020399,8	1020399,8			309,6	0	309,6	1572	4,37
250272,9	2887,9	844769,9	247587,9	247587,9			88,2	13,3	74,9	1654,6	20,68
0	0	3481604,7	1020399,8	1020399,8			309,6	0	309,6	1572	4,37
3738819,4		593831,1	174041,9				251,2	198,6	52,6		
3857735,8		6571546,6	1926009,9				787,5	204,9	582,6	8782,2	135,95
35678,6		6410	1878,7				2,5	1,9	0,6	424,9	6,08

DATA PREPROCESSING

```
#import libraries and dataset
import numpy as np
import pandas as pd
from nltk.stem.snowball import SnowballStemmer
import seaborn as sns

stemmer = SnowballStemmer('english')
df_train = pd.read_excel('/content/drive/My Drive/dissertation/nyc.xlsx', sheet_name='Informati
on and Metrics')
# encoding="ISO-8859-1"

‘#### DROPS’

df_train = df_train[df_train['BBL - 10 digits'].notna()] #drop rows where BBL is nan
df_train = df_train[df_train['Weather Normalized Source EUI (kBtu/ft²)'].notna()] #drop rows w
here source eui is nan
df_train = df_train[(df_train['Weather Normalized Source EUI (kBtu/ft²)'] != 0)] #drop rows wh
ere source eui is zero
df_train = df_train[(df_train['Total GHG Emissions (Metric Tons CO2e)'] != 0)] #drop rows wh
ere ghg is zero
df_train = df_train[df_train['Total GHG Emissions (Metric Tons CO2e)'].notna()] #drop rows w
here ghg is nan

df_train['BBL - 10 digits'] = df_train['BBL - 10 digits'].astype(int)
df_train['SelfReported Gross Floor Area (ft²)'] = df_train['Self-
Reported Gross Floor Area (ft²)'].astype(int)
df_train.drop_duplicates(['BBL - 10 digits'])

‘#### New dataframe containing only the features kept for analysis’

new_df = pd.DataFrame(df_train[['Borough','Self-
Reported Gross Floor Area (ft²)','Primary Property Type -
 Self Selected','Year Built','Number of Buildings', 'Occupancy','Weather Normalized Source EU
I (kBtu/ft²)','Weather Normalized Site EUI (kBtu/ft²)','Weather Normalized Site Electricity Inten
sity (kWh/ft²)','Weather Normalized Site Natural Gas Intensity (therms/ft²)','Water Use Intensity
 (All Water Sources) (gal/ft²)','Total GHG Emissions (Metric Tons CO2e)','Metered Areas (Ener
gy)','Metered Areas  (Water)','ENERGY STAR Score','Weather Normalized Site Natural Gas U
se (therms)','Electricity Use - Grid Purchase (kWh)']])


# encode borough
new_df['Borough'] = new_df['Borough'].replace({"Manhattan": 1 ,"Bronx": 2,"Brooklyn" : 3,"Q
ueens" : 4,"Staten Island" : 5,"Pine Hill" : 6})

#group building types
```

```
new_df['Primary Property Type - Self Selected'] = new_df['Primary Property Type -
 Self Selected'].replace(['K-12 School','College/University','Pre-
school/Daycare','Library','Other - Education','Adult Education'],'Educational')
new_df['Primary Property Type - Self Selected'] = new_df['Primary Property Type -
 Self Selected'].replace(['Hotel','Residence Hall/Dormitory'],'Hotel')
new_df['Primary Property Type - Self Selected'] = new_df['Primary Property Type -
 Self Selected'].replace(['Office','Medical Office','Financial Office','Veterinary Office','Mailing
Center/Post Office'],'Office')
new_df['Primary Property Type - Self Selected'] = new_df['Primary Property Type -
 Self Selected'].replace(['Multifamily Housing','Other -
 Lodging/Residential','Residential Care Facility'],'Residential')
new_df['Primary Property Type - Self Selected'] = new_df['Primary Property Type -
 Self Selected'].replace(['Urgent Care/Clinic/Other Outpatient','Other -
 Special-
ty Hospital','Hospital (General Medical & Surgical)','Outpatient Rehabilitation/Physical Therapy
'],'Hospital')
new_df['Primary Property Type - Self Selected'] = new_df['Primary Property Type -
 Self Selected'].replace(['Retail Store','Supermarket/Grocery Store','Other -
 Mall','Strip Mall','Enclosed Mall'],'Malls and Stores')
new_df['Primary Property Type - Self Selected'] = new_df['Primary Property Type -
 Self Selected'].replace(['Non-Refrigerated Warehouse','Self-
Storage Facility','Refrigerated Warehouse'],'Warehouse')
new_df['Primary Property Type - Self Selected'] = new_df['Primary Property Type -
 Self Selected'].replace(['Senior Care Community','Worship Facility','Police Station','Fire Station
','Parking','Other - Public Services','Other -
 Entertain-
ment/Public Assembly','Fitness Center/Health Club/Gym','Performing Arts','Social/Meeting Hal
l','Movie Theater','Courthouse','Museum','Bank Branch'],'Public Building')


for item in new_df['Primary Property Type - Self Selected']:
  if item not in ['Educational','Hotel','Office','Residential','Hospital','Malls and Stores','Warehous
e','Public Building']:
    new_df['Primary Property Type - Self Selected'] = new_df['Primary Property Type -
 Self Selected'].replace(item,'Other')
‘#### OUTLIER DETECTION’

#sort values

new_df.sort_values(by='Primary Property Type - Self Selected',inplace=True)
new_df.reset_index(drop=True, inplace=True)

#index for each building type
new_df[new_df['Primary Property Type - Self Selected'] == 'Educational' ].index
new_df[new_df['Primary Property Type - Self Selected'] == 'Hospital' ].index
new_df[new_df['Primary Property Type - Self Selected'] == 'Hotel' ].index
new_df[new_df['Primary Property Type - Self Selected'] == 'Office' ].index
new_df[new_df['Primary Property Type - Self Selected'] == 'Other' ].index
new_df[new_df['Primary Property Type - Self Selected'] == 'Public Building' ].index
new_df[new_df['Primary Property Type - Self Selected'] == 'Residential' ].index
```

```
new_df[new_df['Primary Property Type - Self Selected'] == 'Malls and Stores' ].index
new_df[new_df['Primary Property Type - Self Selected'] == 'Warehouse' ].index

#LOG10 of Total GHG emissions
new_df['loglog'] = np.log10(new_df['Total GHG Emissions (Metric Tons CO2e)'])

#educational
ed1 = 2 * new_df['loglog'][0:2361].std() + new_df['loglog'][0:2361].mean()
ed2 =   new_df['loglog'][0:2361].mean() - 2 * new_df['loglog'][0:2361].std()
# drop entries which are above or below the boundary
new_df.drop(list(new_df[new_df['loglog']> ed1].index),inplace=True)
new_df.drop(list(new_df[new_df['loglog']< ed2].index),inplace=True)

#hospital
hos1 = 2 * new_df['loglog'][2362:2548].std() + new_df['loglog'][2362:2548].mean()
hos2 =   new_df['loglog'][2362:2548].mean() - 2 * new_df['loglog'][2362:2548].std()
new_df.drop(list(new_df[new_df['loglog']> hos1].index),inplace=True)
new_df.drop(list(new_df[new_df['loglog']< hos2].index),inplace=True)

#hotel
hot1 = 2 * new_df['loglog'][2549:3426].std() + new_df['loglog'][2549:3426].mean()
hot2 =   new_df['loglog'][2549:3426].mean() - 2 * new_df['loglog'][2549:3426].std()
new_df.drop(list(new_df[new_df['loglog']> hot1].index),inplace=True)
new_df.drop(list(new_df[new_df['loglog']< hot2].index),inplace=True)

#retail
mall1 = 2 * new_df['loglog'][3427:3825].std() + new_df['loglog'][3427:3825].mean()
mall2 =   new_df['loglog'][3427:3825].mean() - 2 * new_df['loglog'][3427:3825].std()
new_df.drop(list(new_df[new_df['loglog']> mall1].index),inplace=True)
new_df.drop(list(new_df[new_df['loglog']< mall2].index),inplace=True)

#office
off1 = 2 * new_df['loglog'][3826:6229].std() + new_df['loglog'][3826:6229].mean()
off2 =   new_df['loglog'][3826:6229].mean() - 2 * new_df['loglog'][3826:6229].std()
new_df.drop(list(new_df[new_df['loglog']> off1].index),inplace=True)
new_df.drop(list(new_df[new_df['loglog']< off2].index),inplace=True)

#other
oth1 = 2 * new_df['loglog'][6230:7269].std() + new_df['loglog'][6230:7269].mean()
oth2 =   new_df['loglog'][6230:7269].mean() - 2 * new_df['loglog'][6230:7269].std()
new_df.drop(list(new_df[new_df['loglog']> oth1].index),inplace=True)
new_df.drop(list(new_df[new_df['loglog']< oth2].index),inplace=True)

#public
pub1 = 2 * new_df['loglog'][7270:8189].std() + new_df['loglog'][7270:8189].mean()
pub2 =   new_df['loglog'][7270:8189].mean() - 2 * new_df['loglog'][7270:8189].std()
new_df.drop(list(new_df[new_df['loglog']> pub1].index),inplace=True)
new_df.drop(list(new_df[new_df['loglog']< pub2].index),inplace=True)
```

```
#residential
res1 = 2 * new_df['loglog'][8190:27265].std() + new_df['loglog'][8190:27265].mean()
res2 =   new_df['loglog'][8190:27265].mean() - 2 * new_df['loglog'][8190:27265].std()
new_df.drop(list(new_df[new_df['loglog']> res1].index),inplace=True)
new_df.drop(list(new_df[new_df['loglog']< res2].index),inplace=True)

#warehouse
war1 = 2 * new_df['loglog'][27266:27922].std() + new_df['loglog'][27266:27922].mean()
war2 =   new_df['loglog'][27266:27922].mean() - 2 * new_df['loglog'][27266:27922].std()
new_df.drop(list(new_df[new_df['loglog']> war1].index),inplace=True)
new_df.drop(list(new_df[new_df['loglog']< war2].index),inplace=True)

#replace nan values with the mean value of the respective column

mean_water = new_df['Water Use Intensity (All Water Sources) (gal/ft²)'].mean()
new_df['Water Use Intensity (All Water Sources) (gal/ft²)'] = new_df['Water Use Intensity (All
Water Sources) (gal/ft²)'].fillna(mean_water)
mean_el =  new_df['Weather Normalized Site Electricity Intensity (kWh/ft²)'].mean()
new_df['Weather Normalized Site Electricity Intensity (kWh/ft²)'] = new_df['Weather Normaliz
ed Site Electricity Intensity (kWh/ft²)'].fillna(mean_el)
mean_es =  new_df['ENERGY STAR Score'].mean()
new_df['ENERGY STAR Score'] = new_df['ENERGY STAR Score'].fillna(mean_es)
mean_therms =  new_df['Weather Normalized Site Natural Gas Use (therms)'].mean()
new_df['Weather Normalized Site Natural Gas Use (therms)'] = new_df['Weather Normalized S
ite Natural Gas Use (therms)'].fillna(mean_therms)
mean_grid =  new_df['Electricity Use - Grid Purchase (kWh)'].mean()
new_df['Electricity Use - Grid Purchase (kWh)'] = new_df['Electricity Use -
 Grid Purchase (kWh)'].fillna(mean_grid)


#one hot encoding for the property type column
df_dummies = pd.get_dummies(new_df['Primary Property Type - Self Selected'], prefix='type')
new_df = pd.concat([new_df, df_dummies], axis=1)
new_df.head()
```

REGRESSION

```
#TRAIN-TEST SPLIT

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix,accuracy_score,classification_report
from sklearn.metrics import roc_auc_score,roc_curve,scorer
from sklearn.metrics import f1_score
import statsmodels.api as sm
from sklearn.metrics import precision_score,recall_score
from yellowbrick.classifier import DiscriminationThreshold
from sklearn import metrics
from sklearn.preprocessing import scale


cols    = [i for i in new_df.columns if i not in 'Primary Property Type -
 Self Selected'+'loglog'+'Total GHG Emissions (Metric Tons CO2e)'+'Metered Areas (Energy)'
+ 'Metered Areas  (Water)' ]
target_col = ["Total GHG Emissions (Metric Tons CO2e)"]

# features
X = new_df[cols]

# Target variable
y = new_df['Total GHG Emissions (Metric Tons CO2e)']

# split X and y into training and testing sets
train_X,test_X,train_Y,test_Y=train_test_split(X,y,test_size=0.25,random_state=0)

#SCALING
from sklearn import preprocessing
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler().fit(train_X)
train_X = scaler.transform(train_X)
test_X = scaler.transform(test_X)


#CROSS VALIDATION SCORES FOR ALL ALGORITHMS

from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import GridSearchCV, cross_val_score, StratifiedKFold, learning
_curve, cross_validate
from sklearn.model_selection import StratifiedKFold
from xgboost import XGBRegressor
from catboost import CatBoostRegressor
import seaborn as sns
import matplotlib.pyplot as plt
from keras.wrappers.scikit_learn import KerasRegressor
from keras.layers import Dense, Activation
from keras.models import Sequential
from keras.callbacks import History

def build_ann():
    model = Sequential()
    # Adding the input layer and the first hidden layer
    model.add(Dense(41, activation = 'relu', input_dim = 20))

    # Adding the second hidden layer
    model.add(Dense(units = 41, activation = 'relu'))
    # Adding the third hidden layer
    model.add(Dense(units = 41, activation = 'relu'))

    # Adding the output layer
    model.add(Dense(units = 1))
    # Compiling the ANN
    model.compile(optimizer = 'adam', loss = 'mean_squared_error', metrics = ['mse'])

    return model

model  =  KerasRegressor(build_fn=build_ann, batch_size=10, epochs=100)
accuracies = cross_val_score(estimator= model, X=train_X, y=train_Y, cv=5, scoring="neg_root_mean_squared_error")

mean=accuracies.mean()
print(mean)

random_state = 42
skfold = StratifiedKFold(5)
regressors = []
regressors.append(RandomForestRegressor(random_state=random_state))
regressors.append(XGBRegressor(random_state=random_state))
regressors.append(CatBoostRegressor(random_state=random_state))


cv_results = []
for regressor in regressors :
    cv_results.append(cross_val_score(regressor, train_X, train_Y, scoring = "neg_root_mean_squared_error", cv = 5, n_jobs=-1, verbose=2))

cv_means = []
cv_std = []
for cv_result in cv_results:
    cv_means.append(abs(cv_result.mean()))
    cv_std.append(abs(cv_result.std()))

cv_means.append(abs(mean))
cv_std.append(abs(accuracies.std()))

cv_res = pd.DataFrame({"CrossVal_RMSE":cv_means,"CrossValerrors": cv_std,"Algorithm":["Random Forest", "XGBoost", "CatBoost","ANN"]})
%matplotlib inline

plt.figure(dpi = 1200)
plt.figure(figsize=(12,10))
# sns.set(font_scale=3)
```

```
g = sns.barplot("CrossVal_RMSE","Algorithm",data = cv_res, palette="Set2",orient = "h",**{'x
err':cv_std})
g.set_xlabel("RMSE", fontsize = 14)
g.set_ylabel("Algorithm",fontsize = 14)

# plt.xlim(0.85,1)
# plt.xticks(np.arange(0.85, 1, 0.01), fontsize = 12)
plt.yticks(fontsize = 12)
g = g.set_title("Cross validation RMSE scores",fontsize = 14)
# plt.savefig('compare_classifiers.jpg', format='jpg', dpi=600)
# plt.savefig('filename.png', dpi=600, bbox_inches='tight')


# GRID SEARCH FOR ALL ALGORITHMS

#Random Forest

RF = RandomForestRegressor(random_state=42)

ex_param_grid = {'n_estimators': [100,200,300,500,1000],
        'min_samples_leaf': [1,2],
        'min_samples_split': [1,2]}

gsRF = GridSearchCV(RF,param_grid = ex_param_grid, cv=5, scoring="neg_root_mean_squar
ed_error", n_jobs= -1, verbose = 1)

gsRF.fit(train_X,train_Y)

y_pred = gsRF.predict(test_X)
print('Mean Absolute Error (MAE):', metrics.mean_absolute_error(test_Y, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(test_Y, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(test_Y, y_pred)))

RF_best = gsRF.best_estimator_
print("Best Score: ",gsRF.best_score_)
print("Best estimator: ",RF_best)

#XGBOOST

XGB = XGBRegressor(random_state=42)

ex_param_grid = {'n_estimators': [100,200,300,500,1000],
        'min_samples_leaf': [1,2],
        'min_samples_split': [1,2]}

gsXGB = GridSearchCV(XGB,param_grid = ex_param_grid, cv=5, scoring="neg_root_mean_s
quared_error", n_jobs= -1, verbose = 1)

gsXGB.fit(train_X,train_Y)
```

```
y_pred = gsXGB.predict(test_X)
print('Mean Absolute Error (MAE):', metrics.mean_absolute_error(test_Y, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(test_Y, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(test_Y, y_pred)))

XGB_best = gsXGB.best_estimator_
print("Best Score: ",gsXGB.best_score_)
print("Best estimator: ",XGB_best)

#CATBOOST
CAT = CatBoostRegressor(random_state=42)

ex_param_grid = {'n_estimators': [100,200,300,500,1000],
        'depth':[2,4,6,8,10],
        'l2_leaf_reg':[1,2,3]}

gsCAT = GridSearchCV(CAT,param_grid = ex_param_grid, cv=5, scoring="neg_root_mean_squared_error", n_jobs= -1, verbose = 1)

gsCAT.fit(train_X,train_Y)

y_pred = gsCAT.predict(test_X)
print('Mean Absolute Error (MAE):', metrics.mean_absolute_error(test_Y, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(test_Y, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(test_Y, y_pred)))

CAT_best = gsCAT.best_params_
print("Best Score: ",gsCAT.best_score_)
print("Best estimator: ",CAT_best)

##TUNING ANNs##
def build_ann():  #(optimizer) otan kanw grid search
    model = Sequential()
    # Adding the input layer and the first hidden layer
    model.add(Dense(45, activation = 'relu', input_dim = 22))

    # Adding the second hidden layer
    model.add(Dense(units = 45, activation = 'relu'))

    # Adding the third hidden layer
    model.add(Dense(units = 45, activation = 'relu'))

    # Adding the output layer
    model.add(Dense(units = 1))
    # Compiling the ANN
    model.compile(optimizer = 'adam', loss = 'mean_squared_error', metrics = ['mse'])

    return model
```

```
model  =  KerasRegressor(build_fn=build_ann, batch_size = 25, epochs = 1000)
model.fit(train_X,train_Y)

y_pred = model.predict(test_X)
print('Mean Absolute Error (MAE):', metrics.mean_absolute_error(test_Y, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(test_Y, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(test_Y, y_pred)))
```

Number of hidden neurons	RMSE
32	23,164836
40	21,15863
45	**20,473213**
50	21,491717

LIMITS FOR 2024-2029

```
#convert tons to kilograms
new_df['Total GHG Emissions (kg CO2e)'] = new_df['Total GHG Emissions (Metric Tons CO2e)']*1000


limits = {"medical office":23.81,
        "retail": 11.81,
        "assembly": 10.74,
        "hotel": 9.87,
        "office": 8.46,
        "school": 7.58,
        "multifamily housing": 6.75,
        "factory": 5.74,
        "storage/warehouse": 4.26}
new_df['limits'] = np.nan


new_df['limits'][0:1617].fillna(limits['school'],inplace = True) #educational
new_df['limits'][1617:1676].fillna(limits['medical office'],inplace = True) #hospital
new_df['limits'][1676:2115].fillna(limits['hotel'],inplace = True) #hotel
new_df['limits'][2115:2342].fillna(limits['retail'],inplace = True) #malls and stores
new_df['limits'][2342:3477].fillna(limits['office'],inplace = True) #office
new_df['limits'][3477:4008].fillna(limits['factory'],inplace = True) #other
new_df['limits'][4008:4468].fillna(limits['assembly'],inplace = True) #public building
new_df['limits'][4468:19042].fillna(limits['multifamily housing'],inplace = True) #residential
new_df['limits'][19042:19417].fillna(limits['storage/warehouse'],inplace = True) #warehouse


#limits
```

```
new_df['compliance limit'] = new_df['limits']*new_df['Self-Reported Gross Floor Area (ft²)']
new_df['comply'] = 0

for i in range(len(new_df)):
  if new_df['Total GHG Emissions (kg CO2e)'][i] <= new_df['compliance limit'][i]:
    new_df.loc[i,'comply'] = 1
```

LIMITS FOR 2030-2034

```
limits2 = {"medical office":11.93,
          "retail": 4.3,
          "assembly": 4.2,
          "hotel": 5.26,
          "office": 4.53,
          "school": 3.44,
          "multifamily housing": 4.07,
          "factory": 1.67,
          "storage/warehouse": 1.1}
new_df['limits2'] = np.nan


new_df['limits2'][0:1617].fillna(limits2['school'],inplace = True) #educational
new_df['limits2'][1617:1676].fillna(limits2['medical office'],inplace = True) #hospital
new_df['limits2'][1676:2115].fillna(limits2['hotel'],inplace = True) #hotel
new_df['limits2'][2115:2342].fillna(limits2['retail'],inplace = True) #malls and stores
new_df['limits2'][2342:3477].fillna(limits2['office'],inplace = True) #office
new_df['limits2'][3477:4008].fillna(limits2['factory'],inplace = True) #other
new_df['limits2'][4008:4468].fillna(limits2['assembly'],inplace = True) #public building
new_df['limits2'][4468:19042].fillna(limits2['multifamily housing'],inplace = True) #residential
new_df['limits2'][19042:19417].fillna(limits2['storage/warehouse'],inplace = True) #warehouse


new_df['compliance limit2'] = new_df['limits2']*new_df['Self-Reported Gross Floor Area (ft²)']
new_df['comply2'] = 0


for i in range(len(new_df)):
  if new_df['Total GHG Emissions (kg CO2e)'][i] <= new_df['compliance limit2'][i]:
    new_df.loc[i,'comply2'] = 1
```

CLASSIFICATION

```
#TRAIN- TEST SPLIT
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix,accuracy_score,classification_report
from sklearn.metrics import roc_auc_score,roc_curve,scorer
from sklearn.metrics import f1_score
import statsmodels.api as sm
```

```
from sklearn.metrics import precision_score,recall_score
from yellowbrick.classifier import DiscriminationThreshold
from sklearn import metrics
from sklearn.preprocessing import scale


cols    = [i for i in new_df.columns if i not in 'Primary Property Type -
 Self Selected'+'loglog' + 'comply'+ 'comply2' +'Self-
Report-
ed Gross Floor Area (ft²)'+ 'Total GHG Emissions (kg CO2e)' +'limits' +'limits2' +'compliance li
mit'+ 'compliance limit2' + 'Total GHG Emissions (Metric Tons CO2e)'+'Metered Areas (Energ
y)' + 'Metered Areas  (Water)']

# features
X = new_df[cols]

# Target variable
y = new_df['comply'] # “comply” for the first period and “comply2” for the second period

# split X and y into training and testing sets
train_X,test_X,train_Y,test_Y=train_test_split(X,y,test_size=0.25,random_state=0)


#SCALING
from sklearn import preprocessing
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler().fit(train_X)
train_X = scaler.transform(train_X)
test_X = scaler.transform(test_X)


# CROSS VALIDATION SCORES FOR ALL ALGORITHMS
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import GridSearchCV, cross_val_score, StratifiedKFold, learning
_curve, cross_validate
from sklearn.model_selection import StratifiedKFold
from xgboost import XGBClassifier
from catboost import CatBoostClassifier
import seaborn as sns
import matplotlib.pyplot as plt


random_state = 42
skfold = StratifiedKFold(5)
classifiers = []
classifiers.append(RandomForestClassifier(random_state=random_state))
classifiers.append(XGBClassifier(random_state=random_state))
classifiers.append(CatBoostClassifier(random_state=random_state))
```

```
cv_results = []
for classifier in classifiers :
    cv_results.append(cross_val_score(classifier, train_X, train_Y, scoring = "accuracy", cv = skf
old, n_jobs=-1, verbose=2))

cv_means = []
cv_std = []
for cv_result in cv_results:
    cv_means.append(cv_result.mean())
    cv_std.append(cv_result.std())


cv_res = pd.DataFrame({"CrossVal_accuracy":cv_means,"CrossValerrors": cv_std,"Algorithm"
:["Random Forest", "XGBoost", "CatBoost"]})
%matplotlib inline

plt.figure(dpi = 1200)
plt.figure(figsize=(12,10))
# sns.set(font_scale=3)

g = sns.barplot("CrossVal_accuracy","Algorithm",data = cv_res, palette="Set2",orient = "h",**
{'xerr':cv_std})
g.set_xlabel("Accuracy", fontsize = 14)
g.set_ylabel("Algorithm",fontsize = 14)

plt.xlim(0.85,1)
plt.xticks(np.arange(0.85, 1, 0.01), fontsize = 12)
plt.yticks(fontsize = 12)
g = g.set_title("Cross validation accuracy scores",fontsize = 14)
# plt.savefig('compare_classifiers.jpg', format='jpg', dpi=600)
# plt.savefig('filename.png', dpi=600, bbox_inches='tight')


#GRID SEARCH FOR ALL ALGORITHMS

#RANDOM FOREST
RF = RandomForestClassifier(random_state=42)

ex_param_grid = {'n_estimators': [50,80,100,120,140,200],
        'min_samples_leaf': [1,2],
        'min_samples_split': [1,2]}

gsRF = GridSearchCV(RF,param_grid = ex_param_grid, cv=skfold, scoring="accuracy", n_job
s= -1, verbose = 1)

gsRF.fit(train_X,train_Y)

y_pred = gsRF.predict(test_X)
print("Accuracy: ",metrics.accuracy_score(test_Y,y_pred))
```

```
RF_best = gsRF.best_estimator_
print("Best Score: ",gsRF.best_score_)
print("Best estimator: ",RF_best)

#XGBOOST
XGB = XGBClassifier(random_state=42)

ex_param_grid = {'n_estimators': [50,80,100,120,140,200],
        'min_samples_leaf': [1,2],
        'min_samples_split': [1,2]}

gsXGB = GridSearchCV(XGB,param_grid = ex_param_grid, cv=skfold, scoring="accuracy", n_jobs= -1, verbose = 1)

gsXGB.fit(train_X,train_Y)

y_pred = gsXGB.predict(test_X)
print("Accuracy: ",metrics.accuracy_score(test_Y,y_pred))

XGB_best = gsXGB.best_estimator_
print("Best Score: ",gsXGB.best_score_)
print("Best estimator: ",XGB_best)

#CATBOOST
CAT = CatBoostClassifier(random_state=42)

ex_param_grid = {'n_estimators': [50,80,100,120,140,200],
        'depth':[2,4,6,8,10],
        'l2_leaf_reg':[1,2,3]}

gsCAT = GridSearchCV(CAT,param_grid = ex_param_grid, cv=skfold, scoring="accuracy", n_jobs= -1, verbose = 1)

gsCAT.fit(train_X,train_Y)

y_pred = gsCAT.predict(test_X)
print("Accuracy: ",metrics.accuracy_score(test_Y,y_pred))

CAT_best = gsCAT.best_params_
print("Best Score: ",gsCAT.best_score_)
print("Best estimator: ",CAT_best)
```

PLOTS

```
#HEATMAP
corr = new_df.corr()
mask = np.zeros_like(corr, dtype=np.bool)
mask[np.triu_indices_from(mask)] = True
```

```
f, ax = plt.subplots(figsize=(40, 25), tight_layout=True)
cmap = sns.diverging_palette(220, 10, as_cmap=True)
sns.heatmap(corr, mask=mask, cmap=cmap, vmax=1, vmin=-
1, center=0, square=True, linewidths=.5, cbar_kws={"shrink": .5}, annot = True, annot_kws={"
size": 20})
ax.axes.xaxis.set_ticklabels([])
plt.yticks(rotation=0, size = 36)
plt.xticks(size = 36)
plt.title('Heatmap')
plt.show()


# TOTAL GHG EMISSIONS VS PROPERTY TYPE
plt.figure(dpi = 1200)
plt.figure(figsize = (20,9))
result = new_df.groupby(["Primary Property Type -
 Self Selected"])['Total GHG Emissions (Metric Tons CO2e)'].mean().reset_index().sort_values(
'Total GHG Emissions (Metric Tons CO2e)')
sns.barplot(x='Primary Property Type -
 Self Selected', y="Total GHG Emissions (Metric Tons CO2e)", data=new_df, order=result['Pri
mary Property Type -
 Self Selected']) #formerly: sns.barplot(x='Id', y="Speed", data=df, palette=colors, order=result['
Id'])
# plt.ylim(0,300000)

# la-
bels4 = ['Warehouse','Other','Residential','Public Building','Hospital','Office','Educational','Hotel
','Retail']
plt.xticks(fontsize = 12)
plt.xlabel('Primary Property Type - Self Selected',fontsize = 15)
plt.ylabel('Total GHG Emissions (Metric Tons CO2e)', fontsize = 15)
plt.show(sns)


# TOTAL GHG VS TYPE AND BOROUGH
plt.figure(dpi = 1200)
plt.figure(figsize = (20,9))

sns.barplot(x="Borough", y="Total GHG Emissions (Metric Tons CO2e)", hue="Primary Prope
rty Type - Self Selected", data=new_df)
plt.xticks(np.arange(5),['Brooklyn','Bronx','Queens','Manhattan','Staten Island'],fontsize = 14)
plt.ylabel("Total GHG Emissions (Metric Tons CO2e)", size=14)
plt.xlabel("Borough", size=16)
plt.title("GHG Emissions by Borough and Property Type", size=18)
# plt.savefig("grouped_barplot_Seaborn_barplot_Python.png")


#YEAR OF BUILT VS SOURCE EUI
rslt_df1 = new_df[new_df['Year Built'] < 1900]
rslt_df2 = new_df[(new_df['Year Built'] >=1900) & (new_df['Year Built']<1950)]
```

```
rslt_df3 = new_df[(new_df['Year Built'] >=1950) & (new_df['Year Built']<1970)]
rslt_df4 = new_df[(new_df['Year Built'] >=1970) & (new_df['Year Built']<2000)]
rslt_df5 = new_df[new_df['Year Built'] >=2000]


y1 = rslt_df1['Weather Normalized Site EUI (kBtu/ft²)'].mean()
y2 = rslt_df2['Weather Normalized Site EUI (kBtu/ft²)'].mean()
y3 =rslt_df3['Weather Normalized Site EUI (kBtu/ft²)'].mean()
y4 = rslt_df4['Weather Normalized Site EUI (kBtu/ft²)'].mean()
y5 = rslt_df5['Weather Normalized Site EUI (kBtu/ft²)'].mean()
years2 = [y1,y2,y3,y4,y5]
xl2 = ['before 1900','1900-1950','1950-1970','1970-2000','after 2000']


plt.figure(dpi = 1200)
plt.figure(figsize = (12,8))
sns.barplot(x=xl2, y=years2)  #formerly: sns.barplot(x='Id', y="Speed", data=df, palette=colors,
order=result['Id'])
# plt.ylim(0,300000)
# plt.xticks(np.arange(5),['Brooklyn','Bronx','Queens','Manhattan','Staten Island'],fontsize = 16)

plt.xlabel('Year Built',fontsize = 16)
plt.ylabel('Weather normalized site EUI', fontsize = 16)

plt.show(sns)


# PIE CHART WITH BUILDING TYPE PROPORTIONS
new_df['Primary Property Type - Self Selected'].value_counts()
jj = {"Residential":19076, "Office": 2404, "Educational": 2362, "Other":1040, "Public Building
": 920,
    "Hotel": 878, "Warehouse": 657, "Retail": 399, "Hospital": 187}

data = [19076,2404,2362,1040,920,878,657,399,187]
labels = ["Residential","Office","Educational","Other","Public Building","Hotel","Warehouse","
Retail","Hospital"]
# plot = new_df.plot.pie(y=j, figsize=(10, 10))
my_explode = (0.1, 0, 0,0,0,0,0,0,0)
plt.figure(figsize=(10,8))
plt.pie(data,autopct='%1.1f%%', shadow = True, explode=my_explode)
plt.legend(labels, loc = "best")
plt.show()


#LOGGED GHG EMISSIONS DISTRIBUTION BEFORE DROPPING OUTLIERS
new_df['loglog'].describe()
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

```
plt.figure(dpi = 1200)
plt.figure(figsize = (20,9))
sns.histplot(new_df['loglog'], kde=True, color = 'blue', bins = 50)
sns.set_style("white")
plt.xlabel("log(mt CO2)")
plt.ylabel("Count")
# plt.xlim(2,3)
# plt.title("GHG")
plt.axvline(np.mean(new_df['loglog']),color='r', linestyle='--')
plt.text(8,5800,"mean: 2.54" ,color = 'black',size = 12)
plt.text(8,5500,"std: 0.49" ,color = 'black',size = 12)
plt.text(8,5200,"min: -0.69" ,color = 'black',size = 12)
plt.text(8,4900,"max: 8.72" ,color = 'black',size = 12)


#LOGGED GHG EMISSIONS DISTRIBUTION AFTER DROPPING OUTLIERS


new_df['loglog'].describe()
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
plt.figure(dpi = 1200)
plt.figure(figsize = (20,9))
sns.histplot(new_df['loglog'], kde=True, color = 'blue', bins = 50)
sns.set_style("white")
plt.xlabel("log(mt CO2)")
plt.ylabel("Count")
plt.xlim(1.8,3.2)
# plt.title("GHG")
plt.axvline(np.mean(new_df['loglog']),color='r', linestyle='--')

plt.text(3,560,"mean: 2.51" ,color = 'black',size = 12)
plt.text(3,540,"std: 0.2" ,color = 'black',size = 12)
plt.text(3,520,"min: 2.08" ,color = 'black',size = 12)
plt.text(3,500,"max: 2.9" ,color = 'black',size = 12)


#PLOT ORIGINAL TOTAL GHG EMISSIONS DISTRIBUTION AFTER DROPPING OUTLIERS


new_df['Total GHG Emissions (Metric Tons CO2e)'].describe()
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
plt.figure(dpi = 1200)
plt.figure(figsize = (20,9))
sns.histplot(new_df['Total GHG Emissions (Metric Tons CO2e)'], kde=True, color = 'blue', bins = 50)
sns.set_style("white")
plt.xlabel("mt CO2")
plt.ylabel("Count")
# plt.xlim(0,500000)
# plt.title("GHG")
plt.axvline(np.mean(new_df['Total GHG Emissions (Metric Tons CO2e)']),color='r', linestyle='--')

plt.text(700,700,"mean: 361.02" ,color = 'black',size = 12)
plt.text(700,670,"std: 168.28" ,color = 'black',size = 12)
plt.text(700,640,"min: 123.0" ,color = 'black',size = 12)
plt.text(700,610,"max: 805.3" ,color = 'black',size = 12)


#FEATURE IMPORTANCE PLOT

importance = rfr.feature_importances_
for i,v in enumerate(importance):
  print('Feature: %0d, Score: %.5f' % (i,v))
# plot feature importance
plt.figure(dpi = 1200)
plt.figure(figsize=(12,10))
labels1 = np.asarray([i for i in range(len(importance))])
labels2 = cols
plt.bar([x for x in range(len(importance))], importance, tick_label = labels2)
plt.xticks(rotation = 90)

# ax.set_xticks(idx)
plt.show()

#MSE VALIDATION SCORE VS NUMBER OF EPOCHS FOR ANNs

print(history.history['val_mse'])
plt.figure(dpi = 1200)
plt.figure(figsize=(12,10))
plt.plot(history.history['val_mse'])
plt.ylabel("MSE validation score", fontsize = 20)
plt.xlabel("Epochs", fontsize = 20)
plt.xticks(fontsize = 18)
plt.yticks(fontsize = 18)
plt.show()
```