I. Introduction

Over the duration of the semester, my Machine Learning course covered model types of topics including Regression and Classification. This project deep-dives the use of regression where the highest performing models produced a R2 score of 0.9. Meanwhile, classification of fraudulent charges were created in order to predict a fraudulent charge with an accuracy of 0.999.

II. First Dataset: Energy Efficiency

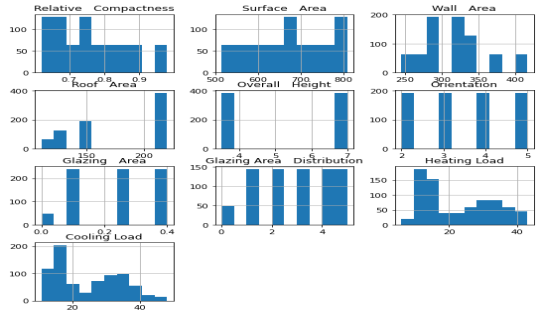
Everyday use of energy and resources have skyrocketed in recent years. With the regression data models it helps to predict and approximate the heating and cooling load needed of different types of buildings in order to help reduce and save the amount of energy being used.

A. Data Set

This dataset contains twelve different types of buildings with eight different types of feature elements in order to use the dataset for regression. These feature elements for the building consist of characteristics of the building such as height and wall area. Meanwhile the target variables for the dataset are the cooling and heating load. Fine tuning the prediction of this dataset results in proper energy analysis through machine learning regression models.

B. Data Exploration

Visualizing the building data and all variables of the dataset shows the following image.



The use of histograms to visualize the data helps to view the frequency of values being used throughout the dataset. For example, the histograms help to identify that the orientation of the buildings is separated into four different values rather than split up over hundreds of datapoints. Unlike the credit card dataset, there doesn't seem to be any overall issues with the data concerning balance or obscure, outlying data points. Checking for empty values or missing values in the dataset resulted in none being found.

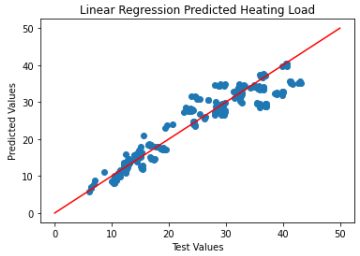
C. Preprocessing Techniques

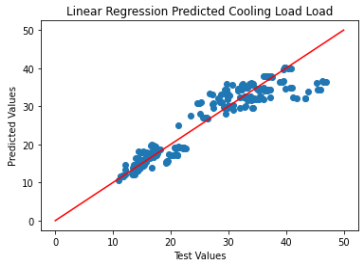
As it pertains to the data just explored, there is no Gausian distribution of the data we just looked at. In order to create a cleaner and more accurate model, I chose to normalize the feature training and testing set. Normalization is a scaling technique that creates a common scale amongst all values of the data, without distorting the range in minimum and maximum values in the dataset. Due to the variety of ranges

amongst the features variables, I believe Normalization was the correct way to create a common path amongst all the data. With regards to splitting the testing data, I used the Sckit-learn train_test_split function in order to create a testing set that was one-third of the overall data provided in the dataset. The scaling and splitting of the dataset leads into through regression tactics I used in order to discover and create meaning from energy efficiency data.

D. Linear Regression

In order to perform energy efficiency analysis on the dataset, I thought an import technique to use was Linear Regression. This regression technique is a well-known supervised machine learning problem known for its simplicity. Unlike other machine learning models, there aren't many parameters needed to be fine-tuned in order to create a well-ordered machine learning model. In regards to my Linear Regression application, the only parameter that was implemented was setting the Normalization of the data to false because previously in the preprocessing section we had already achieved this. The result of our linear regression model being applied to the testing data resulted in a R2 Score of a 0.90 score, which is very good considering the top score is 1. The following graphs show the deviation of the prediction values from the target values.





The Cooling Load, visually did not seem to perform as good as the heating load but it is still a fairly accurate prediction.

E. Ridge Regression

Ridge Regression is known as a regularized linear regression model in which a regularizer term is added to the cost function. The Sckit-learn Ridge Regression model required the strength parameter to be adjusted to 0.2 in order to reach a peak R2 Score of 0.899 which performed slightly worse than my linear regression model. The following is a table of the ridge regression model evaluations.

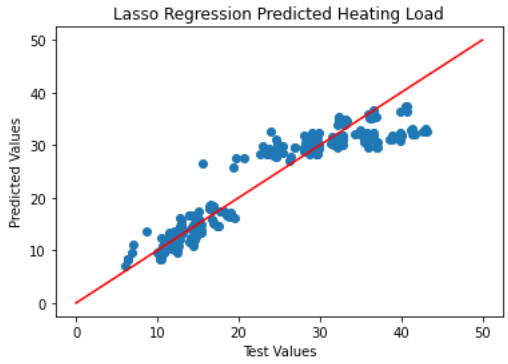
Table I: Ridge Regression Evaluation

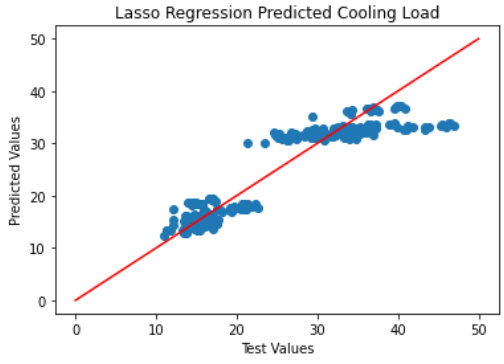
Mean Absolute Error	2.262865661831279
Root Mean Squared Error	3.130251468619064
Mean Squared Absolute Error	9.79847425679180
R2 Score	0.8995346011629067

Mean absolute error being 2.26 shows the many outliers in the data set where handled effectively but could be improved.

F. Lasso Regression

Lasso Regression functions very similarly to Ridge besides the fact that it uses the full norm of the weight vector. This Lasso Regression algorithm performed the worst on the data due to the fact that it was undefitting as seen in the graphs there were two clusters of data and it created a bigger variance between the predictions vs the actual data.

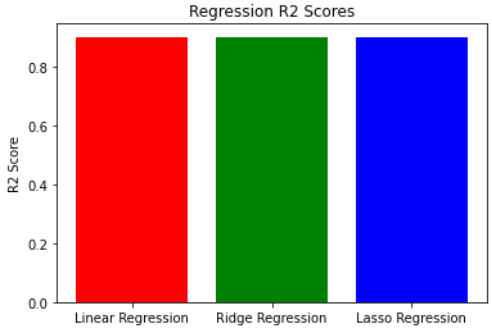




The data points lying far away from the red line show the lesser performance of this model. This model only finished with a score of 0.87 R2 Score.

G. Comparing Models

Overall, out of the three regression models, Linear Regression performed the best.



As shown in the regression bar graphs, Linear Regression had the highest R2 score followed by Ridge, than Lasso. I believe the models are a little under fitted compared to the data but that is expected regression models such as these. The models performed well by there are still different techniques available such as other types of scaling in order to see improvements with the predictions.

III. Classification Dataset: Credit Card Transactions

Credit card companies are faced with the challenge of deciding which customers' purchases are normal everyday purchases vs which are fraudulent. This process of labeling transitions helps prevent crime and save their card users thousands of dollars.

H. Data Set

The fraudulent credit card dataset contains transactions made by credit cards in September 2013 by European cardholders. The data set consists of 30 features and 284, 807 transitions for each feature making this dataset fairly large. Of the large number of transitions in the dataset, only 492 are verified fraudulent purchases.

I. Data Exploration

I used a histogram of the data in order to identify the frequency of values in the dataset as well as to help locate our target variable. This variable is clearly identified as the Class column which is a binary variable holding only values of 1 and 0. The following pie chart of these demonstrates the distribution of the data.

As seen in the visual, there is a substantial difference between the number of fraudulent purchases(Class 1) and non-fraudulent purchases(Class=0). This considerable discrepancy in the data results in imbalanced data. In order to achieve proper and accurate results a few techniques will need to be used in order to alter the data accordingly.

J. Pre-processing Techniques

The class imbalance problem is the distribution of values in a classification problem is biased or skewed. This problem poses a threat to our predictions because most classification predictive models are based upon the assumption that there is an almost equal amount of distribution for each value in a variable. So, in order to fix this issue, I used an imbalanced-learn library in order to alter the dataset. The SMOTE function was applied to effectively balance the data. What the SMOTE function does on this data is it over-samples the fraudulent charges in our target variable which in this case is the minority class. After the function was applied on the dataset, the data resulted in almost equal share of values between the data.

Next, to preprocess the data I scaled the data. In order to properly scale the data, I used the Scikit-learn StandardScaler method

which normalizes each individual feature of the X columns of the dataset. The process of scaling resulted in better accuracy scores for almost every model performed on the dataset.

K. Logistic Regression

One of the key attributes to obtain a high level of accuracy for machine learning is to be able to fine tune a model hyperparameters. Due to the size and amount of parameters that need changing this would become a very tedious task. In this case, I used the GridSearchCV Scikit-learn function to automate the process of optimizing a model. Logistic Regression uses a logistic function in order to model a binary dependent variable. In the case of this Logistic Regression model, the GridSearchCV returned the best parameters for the logistic regression model as 'C': 10 and the 'solver': 'lbfgs'. When the model is run on these optimized parameters it results in an accuracy score of approximately 98.1%. As well as a recall score for fraudulent cases of around 99%.

Table II

Logistic Regression Classification Report

	precision	recall	f1-score
class=0	0.97	0.99	0.98
class=1	0.99	0.97	0.98

Having a high precision for fraudulent cases is important because the model would be unreliable if many fraudulent purchases went unidentified. It is better to find a False-Positive scenario than a misidentified fraudulent charge.

L. Decision Tree Classifier

A decision tree classifier is a supervised machine learning algorithm that uses a set of rules in order to make a decision on data. One of the hardest goals to accomplish with a decision tree classifier is building the perfect tree, but it handles all types of data easily so it is widely used. My attempt to build a highly accurate model first starts with using the GridSearchCV scikit-learn function again. This time we search through the parameters of criterion and max_depth. Criterion is the function to measure the quality of the split while max_depth is the maximum depth of the tree. The best parameters for the model ended up being ‘entropy’ and 20. When the model was run on the scaled data with these parameters the accuracy score reached a high of 99.5%.

Table III

Decision Tree Classification Report

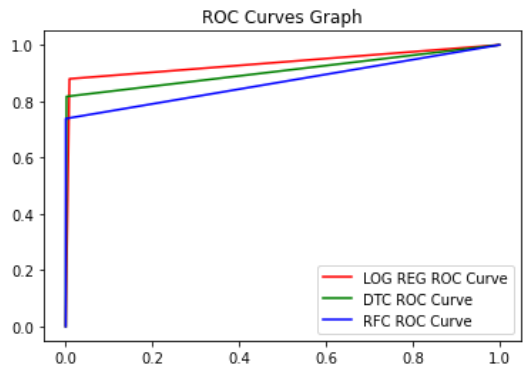
	precision	recall	f1-score
Non Fraudulent	0.97	0.99	0.98
Fraudulent	0.99	0.97	0.98

M. Random Forest Classifier

Random Forest Classifier fits a number of decision trees on the dataset splitting it into subsections of data. The same GridSearchCV technique was also used on this Random Forest Model. The best parameters for the model produced a hyper-parameter of max depth equal to 11 and max features equal to 3. These features and parameter tweaking resulted in an impressive accuracy score of 0.999 which is almost perfect.

N. Comparing Classification Models

The best performing Classification model was easily the Random Forest Classification Model. It produced the highest accuracy of 3 and had a fairly high precision and recall score. Shown is a comparative graph of the 3 Classification models recall and precision plots.



The models are highly accurate and may overfit the data, but accuracy and precision are very important when it comes to predicting fraudulent charges. As it pertains to classification, it has become clear to me that a high accuracy score is not the only score needed to understand a model's performance and the high performing Logistic Regression precision and recall scores help me come to this realization.

O. Conclusion

In the future, I would look at the effects of scaling on the data, as well as using a different approach to fine-tuning the hyper-parameters rather than the GridSearchCV function. Perhaps in future work I could incorporate RandomSearchCV instead. Another aspect of the project I would look into is applying Neural Networks to both dataset. The more complex machine learning models would seem to fit the data well and could produce fairly accurate results. In the meantime, a regression R2 score of 0.9 and classification highest accuracy score of 0.999 is very well performing and this project could help make an impact on energy efficiency and fraudulent crimes.