Comparison of Regression Analysis Algorithms

Final Progress Report

Patrick Rocha

Thesis CS4490Z

Department of Computer Science

Western University

April 8th, 2023

Roberto Solis-Oba, Dept. of Computer Science

Nazim Madhavji, Dept. of Computer Science

**Glossary**

Csv: Comma-separated value. A file extension commonly used for spreadsheets.

MAE: Mean absolute error. A metric for evaluating model performance.

**Structured Abstract**

This paper will conduct a comparative analysis of machine learning algorithms using four datasets with different characteristics. The objective is to shed light on the efficacy of these algorithms and investigate how the characteristics of the dataset impact their performance. By doing so, correlations can be discovered and applied to other datasets outside of the ones used here. The algorithms in question include random forest, linear regression, decision tree, k-neighbour, lasso, and ridge. These are all regression algorithms, and as such will be tested on their ability to predict numerical data. Scikit-learn, a Python library, will be used to create the models and fit them into the data. The results will be compared both with and without a method named one-hot encoding. This allows for a comparison not only of the algorithms themselves, but how encoding influences their results. The results will be measured using a metric named MAE (mean absolute error), which measures the difference between the true value and the predicted value. An analysis like this is important to enhance our understanding of the intricacies of different regression analysis algorithms, why these differences exist, and how the world of computer science and, more specifically, data science can take advantage of them.

**Table of Contents**

1.0 Introduction

2.0 Background and Related Work

2.1 Learning the Process

2.2 Analysis and Research Gap

3.0 Research Objectives

4.0 Methodology

5.0 Results

5.1 Dataset 1: Fuel Consumption Rate

5.2 Dataset 2: Wine Quality

5.3 Dataset 3: Individual Income

5.4 Dataset 4: Housing Prices

5.5 Analysis

6.0 Discussion

6.1 Threats to the Validity of the Results

6.2 Implications of the Research Results

6.3 Limitations of the Results

6.4 Generalisability of the Results

7.0 Conclusions

8.0 Future Work and Lessons Learnt

9.0 Acknowledgements

10.0 References

**1.0 Introduction**

Regression analysis helps us make predictions regarding various data and statistics by utilising regression algorithms to build models. The functionality of it expands across many industries to serve a variety of purposes. It can help manufacturers determine how much product they need to produce, give insight on someone’s future health, and aid financial advisors in forecasting stock prices. Overall, regression analysis plays a major role in the modern world. Several aspects of our lives are reliant on it, so a misuse of the algorithms at play, or a misunderstanding of when to use them, can lead to disastrous results. Poor results are worse than no results at all. Therefore, having accurate models is crucial.

Within regression analysis, there exists a wide variety of different algorithms, all of which can act as the baseline for a machine learning model. To create accurate models, it is important to experiment with these algorithms and gather results to discover various patterns and acquire knowledge on their intricacies. This thesis aims to find correlations and make conclusions based on its experiment involving the comparison of regression analysis algorithms. Several regression algorithms will be looked at and their results will be observed over four different datasets, analysing how and why the results came about as they did. Each dataset covers a different topic with different attributes, including number of columns, number of rows, number of missing values, and the ratio between categorical and numerical values. Every regression analysis algorithm performs better or worse under certain conditions. What will be determined is what these conditions are, why that is the case, and what we can do with that information. The analysis is intended to be useful in areas outside of this paper and not simply to assist in the specific four datasets.

**2.0 Background and Related Work**

To prepare for this research project, research was done to ensure the experiment was valid and the biases were minimized. This includes research into how one would go about creating an empirical study involving machine learning and regression analysis and comparisons of similar works. These comparisons both helped to validify the results as well as determine where there are still areas of improvements that this paper can act upon.

**2.1 Learning the Process**

A variety of sources were used to learn how to form the results that act as the base for the experiment. The beginning of the research started with gathering material for the basics of machine learning. This included books, websites, and journals. An emphasis was placed on educational sources, rather than experiments like this one (Bonaccorso, 2017) (Rong et al., 2018). Once a basis of knowledge was formed for machine learning and regression analysis, more specific sources were needed. This means sources that place a bigger emphasis on the methodology of an experiment like the one present in this paper (Massaron et al., 2016; Ogutu et al., 2012). Further details regarding the methodology for this paper can be found in Section 4.0. Any paper that delved into whatever tools, technologies, and methods that were required became essential to the learning process. Any potential issues would also need to be researched. For example, the problems with overfitting and underfitting (Jabbar et al., 2016), as well as improper encoding (Seger, 2018). Finally, to assist with the experiment itself, other experiments of similar nature were analysed, both to understand the methods and to plan the different steps involved with writing this paper (Archontoulis et al., 2015; Domínguez-Almendros et al., 2011; Maulud et al., 2020).

**2.2 Analysis and Research Gap**

Analysing other author’s works involving regression analysis experiments is essential to producing a meaningful experiment. Multiple sources were analysed, each covering a variety of purposes. For evaluating model performance, Massaron and Boschetti encouraged the use of squared errors because it handles negative values and emphasizes large differences. However, it is important to note that their book mainly focuses on one algorithm at a time to emphasize different aspects of machine learning in Python. This paper looks at various datasets under different conditions at once, so absolute errors were favoured over squared errors. For the experiment, some papers like the one from Rong and Bao-wen created scatterplots to analyse the relationships between variables within the dataset (Rong et al., 2018). While scatterplots can assist with analysis, the purpose of this experiment was more centred towards the relationship between the results of different datasets, not one from within the same one. Rong and Bao-wen’s paper utilised Python for their regression analysis. This is something that nearly every paper I found on the topic had in common was the use of Python. As such, it was believed to be the best option for this paper and was thusly used.

By analysing several different sources, some gaps in the research of regression analysis were found. Many of these papers and books were not comparative, but instead a simple test of something’s performance, or a proposal for new methods (Jabbar et al., 2015). Those that were focused on things outside of the algorithms themselves. For example, Seger’s paper compared different categorical variable encoding techniques (Seger, 2018), and Ogutu’s paper focused on Genomic selection (Ogutu et al., 2012). While these sources involved comparative regression analysis, they shifted their focused on another topic. There are some papers that directly compare the performance of algorithms, but only for a specific scenario (Karadag et al., 2007). Overall,

there is some room for additional experimentation in the realm of general algorithm comparison and analysis. That is why this paper has come to be.

**3.0 Research Objectives**

O1: Observe past experiments from other papers.

O2: Form the results of the analytical experiment.

O3: Analyze the results of the experiment.

O4: Determine what conclusions can be made about the results of the experiment.

**4.0 Methodology**

To conduct the experiment, the programming language Python will be used to read the data, build the models, and compare the results. Python brings several advantages that makes it a comfortable choice. “Python has been focused by researchers due to its concise, elegant and clear language” (Rong et al., 2018). Because of this, Python has plenty of documentation and past experiments that can help. Additionally, a couple of Python libraries will be used to incorporate further functionality. These include Pandas and Scikit-learn. Pandas is used for data manipulation, while Scikit-learn provides the core regression-focused methods. To bring in the data, four datasets in the form of csv (comma-separated value) files will be utilized and read from the Python file. These datasets describe the following: individual income, wine quality, housing prices, and vehicular CO2 emissions. They are initially inserted into the code via the Pandas library. From there, the data needs to be standardized. Without standardization, the data would be unfit for a regression model. To accomplish this, Scikit-learn and Pandas are used together to perform a variety of tasks. The first of these is to rename some columns whose name may provide errors. Afterwards, the columns are split between the target variable and the features variables.

The target variable determines which column the models will be making predictions for. The targets used include ‘CO2\_emissions’ (amount of CO2 emissions per vehicle) for the vehicle data, ‘quality’ (overall quality) for the wine data, ‘price’ (price of the house) for the housing data, and ‘value’ (income) for the individual income data. It is crucial that these values have a strong relation with other columns in the dataset. Otherwise, predictions would not be able to be made. Each dataset picked for the experiment contains these strong relationships.

The feature variables are the columns that will be used to make the predictions for the target variable. A common theme among these datasets is that the target variable essentially acts as a central point that every feature variable is contextually related to. For example, with the wine dataset, there is the wine quality variable, and every feature variable is something that helps determine the wine quality. Hence, wine quality is the central point. Datasets where there is no obvious central point would be of no use to this paper, for the lack of a strong relationship between the variables would make the results less reliable.

The hope is that the differences among these datasets, including the number of rows, number of columns, and context within what their values represent will help avoid biases and grant the potential for a deeper analysis.

**5.0 Results**

The final results of the experiment occur within the four different datasets that were used. The best way to analyse these results in a fair fashion is to analyse their results within each dataset individually, then search for any patterns. This will ensure that the conclusions that arise are meaningful for datasets outside of the ones in this paper. For any dataset that contains categorical columns, they will be transformed into numerical columns using one-hot encoding. What this does is transform each individual categorical column into numerous numerical columns filled with 1s and 0s. For example, in the first dataset regarding fuel consumption rate, there is a categorical column named Make, which contains the company that created the vehicle encompassing that row. A numerical column will be created for each company. All of them will contain a 0, except for the column that matches the company that was in the original categorical column. This column will contain a 1. Additionally, any missing values within the feature columns will be filled via imputation using mean. This means that the mean of the non-missing values is calculated, and that value is used to fill in the missing slots. With these strategies in use, the datasets will be prepared to be used by the models.

**5.1 Dataset 1: Fuel Consumption Rate**

This dataset consists of a variety of information relating to different vehicles (Government of Canada, 2017). This includes information such as the engine size, number of cylinders, and the model. The target for this dataset will be the CO2 emissions. The features will be every column besides the target. In total, there are 5431 rows and 15 columns. Every row and column are in use. 5 out of the 15 columns are categorical columns, with the rest being numerical. Two of the feature columns contain missing values, so some imputation will be required. The final results of the models are as follows:

A graph of a graph showing the amount of fuel consumption rate

Description automatically generated

Figure 1: Graphical representation of dataset 1’s MAE scores

The one-hot-encoded value for linear regression is not shown here because it contains an abnormally high value of 4089466964. There is something else of note regarding this dataset: the categorical columns have many unique values. This is most notable in the Model column, which contains 1357 columns. The Make and Transmission columns also have several unique values, at 41 and 27 values respectively. By removing these 3 categorical columns and keeping the other two, another comparison can be made. Linear regression without one-hot-encoding gets an MAE (mean absolute error) of 9.97. With one-hot-encoding, it gets an MAE of 1.86. This new change produces results that are far more similar to the other algorithms than before. It is crucial to note that the number of columns created with one-hot-encoding is proportional to the number of unique values in each column. “In many of these domains categorical features are common and often of high cardinality. Using one-hot encoding in such circumstances lead to very high dimensional vector representations, causing memory and computability concerns for machine learning models” (Seger). This is demonstrated here in the linear regression’s results. Adding a large number of columns can have negative effects on machine-learning algorithms

The random forest algorithm and decision tree algorithm produced the best results here in both categories, both with very similar results. Another pair that is similar is lasso and ridge. Both of them had poor results without one-hot-encoding and found significant improvements with it. This makes sense, as the two share many similarities by utilising regularisation techniques.

**5.2 Dataset 2: Wine Quality**

This dataset centers around an overall score for wine quality given several different attributes (Parmar, 2018). There are 13 columns and 6497 rows. Naturally, the quality is the target column, and everything else is being used as the features. Of the 13 columns, only one is a categorical column. This being the wine type. It only contains two unique values: white and red.

A chart with purple and yellow bars

Description automatically generated

Figure 2: Graphical representation of dataset 2’s MAE scores

It is seen here that the fact that the difference between one-hot-encoding and no encoding is minimal, for the effect that the wine type has on its quality is minimal. Unlike the previous dataset, the differences between the algorithm’s scores are small. Every algorithm produced an MAE of less than 0.7. This shows that the information provided by the features is extremely relevant to the quality.

**5.3 Dataset 3: Individual Income**

This dataset contains the income of different individuals in Canada, alongside some additional information (Government of Canada, 2023). Some trimming of the columns was needed due to repeated or irrelevant information. Once that was done, there ended up being a total of 5 categorical columns, 4 numerical columns, and 1338839 rows. This is by far the largest number of rows among the 4 datasets in this paper. The target variable is the income of the individual, with the remaining 8 columns acting as the features.

A graph of a number of people

Description automatically generated with medium confidence

Figure 3: Graphical representation of dataset 3’s MAE scores

One immediate thing of note is that without one-hot-encoding, every algorithm had similar results. Furthermore, the MAE values are much larger here than the previous 2 datasets. When one-hot-encoding was used, there were drastic changes within the random forest, decision tree, and k-neighbour algorithms, and very little with the others. Because most of the feature columns are categorical, and because of how relevant they are to the final product, drastic changes like these are to be expected.

**5.4 Dataset 4: Housing Prices**

This dataset focuses on the price of houses given certain properties (Badole, 2024). The final price is the target variable, and everything else will be the features. There are 7 categorical columns and 6 numerical columns, including the target, as well as 545 rows. This makes it the smallest dataset in this paper in terms of number of rows. Each categorical column has very few unique values. The final results are as follows:

A graph showing the number of houses in the housing prices

Description automatically generated

Figure 4: Graphical representation of dataset 4’s MAE scores

In each algorithm, there are small improvements when utilising one-hot-encoding. This is the only instance where improvements are both meaningful and consistent. The MAE is at its highest here by far, but the differences between each algorithm is small.

**5.5 Analysis**

Based on the results of the four datasets, some conclusions can be made. The linear, lasso, and ridge regression algorithms produce far worse results than the other two when too many columns are created as a result of one-hot encoding. This is evident in the third and first datasets, where this attribute held true. The same three algorithms did not receive an improvement under one-hot encoding in the third dataset. These algorithms are all linear algorithms. This suggests that there

is a correlation between the performance of linear algorithms and the reliance on categorical values. However, lasso and ridge still performed well in the first dataset under one-hot encoding. This dataset has more numerical columns, and they are more relevant to the target variable. When measuring fuel consumption in the real world, the numerical values we see in the first dataset are very relevant. Given this information, it can be deduced that one-hot encoding becomes more significant to the final MAE for lasso and ridge when relevant numerical data is used alongside it. Without that data, lasso and ridge do not see a significant improvement with one-hot encoding. Another thing we can deduce with the linear algorithms is that the basic linear regression algorithm falters with too many feature columns. This is shown from the high values in the first dataset. Lasso and ridge do not have their results suffer in the same way, for they show significant improvements with the encoding.

The random forest and decision tree algorithms shared similar results in the four datasets. They were both consistently effective, with no notable abnormalities as seen with the linear algorithms. One-hot encoding always provided benefits. This is significant because for every other algorithm, the encoded results were sometimes worse than the original. There are several reasons why this could be the case. A higher volume of features may make it difficult for models to find underlying patterns. It may also make models more susceptible to overfitting. Random forest and decision tree proved to be resistant to these potential faults within this experiment.

The k-neighbour algorithm is in somewhat of a middle ground between the two groups of algorithms listed previously. There are scenarios where one-hot encoding worsens the results. It never encounters any abnormalities. The MAE values are never the lowest and are only the highest by a small amount in dataset 2. K-neighbour’s values aren’t as good as the ones from random forest and decision tree, but it maintains a similar level of consistency and robustness against the dataset attributes that gave the linear algorithms trouble.

Overall, random forest and decision tree saw the best results. However, this does not mean that they are objectively better than the other algorithms. There are other factors that must be considered when trying to make the best possible model for a specific set of data. These include computation time, parallelization compatibility, and memory availability among other things. The results can be used as a start to determine which kind of algorithms one could begin testing with. There are many different algorithms out there, but each algorithm shares qualities with others. For example, if the decision tree algorithm is determined to work well thanks to these results, one could also consider other tree-based algorithms. That is the purpose of this paper.

**6.0 Discussion**

There are some things to consider before making conclusions about these results.

**6.1 Threats to the Validity of the Results**

The result of this experiment falls under a few limitations. A major threat is the core library in use, Scikit-learn. The performance of the models is partially dependent on the performance of Scikit-learn’s functions. While Scikit-learn brings several methods to improve the validity of the models such as tuning different parameters, the library itself is what allows the models to be created in the first place. Scikit-learn’s personal implementation of their models will have an influence on the final results. However, by adjusting the parameters and doing research into Scikit-learn’s implementation, the biases that arise can be lessened, making this less of a threat to the result’s validity. This was done for the experiment within the code in a process called hyperparameter tuning. It involves using a parameter grid which encompasses many potentially optimal parameters and tests the model’s accuracy using those parameters. All six models that were developed using their respective algorithms were given the most optimal parameters that were found .

**6.2 Implications of the Research Results**

The research in this paper brings light to the strengths and weaknesses of the different algorithms that were compared and showcases their variability within different scenarios. Because of the many different factors and variables that differ in each dataset, it is crucial to understand why it is important that different algorithms exist and how computer scientists and data scientists among others can take advantage of this knowledge.

**6.3 Limitations of the Results**

Just as the Python library Scikit-learn brings some threats to the validity of the results, it may also limit the extent at which they can be utilized. Because the results only consider one of many different libraries and software, it can be tricky to extent the knowledge gained into other works. The effects of this limitation can be minimized by analyzing other experiments done on algorithm comparison that used different software. The results of this paper’s experiments can be compared to other works that contain results for the same algorithms that were used here. For example, one of the algorithms this paper looked at was linear regression. Comparing the linear regression results form this paper to other experiments that used the same dataset, but different libraries could help satisfy the claims made.

**6.4 Generalisability of the Results**

While the results of this research come from four different datasets, they are useful beyond that. The intention of this paper is not simply to test the performance of certain algorithms on the specific datasets, but to analyse their performance under different conditions. These conditions being the unique attributes of the datasets that may also exist in others. For example, the results of the algorithms on a dataset with many categorical columns and rows can be applied to other datasets with the same attributes. Additionally, patterns can be found within the analyses of this paper that allow for a broader scope. If an algorithm sees better performance when more rows are added, then we can apply this knowledge to many more datasets, as opposed to just applying them to datasets with a similar number. Composing these patterns is what allows the results to be generalizable.

**7.0 Conclusions**

There were four objectives listed in Section 3.0. Of the three, O4 was the most crucial. The main objective of this research project was to discover new information regarding the relationship between certain dataset qualities and the performance of regression algorithms. Through the comparative experiment on the regression algorithms and the analysis that resulted off of it, this objective was satisfied. There are patterns that can be found within the analysis in Section 5.5 that can help improve the understanding of when it is optimal to use each pattern. This paper does not answer every question, nor does it kickstart the world’s knowledge of machine learning. Instead, it adds onto what already exists within. It is a building block that can further knowledge in regression algorithms.

**8.0 Future Work and Lessons Learnt**

Several conclusions were made in Section 5.5. However, the quantity and certainty of these conclusions can be improved upon. There were six regression algorithms showcased in the experiment, but there are many more algorithms than that. Additionally, only four datasets were used to compare the algorithms, but far more than that exist. Simply put, there are many ways one can expand upon the experiment done here. They could add more algorithms and compare them in more datasets. The addition of more algorithms would allow more conclusions to be made, for with each algorithm added is potential for another pattern to be found. Having more datasets in this experiment that showcase these patterns being consistent would improve the validity of the results. However, this eventually reaches a point where the effort required does not outweigh the results that arise. If a pattern has already been established, bringing in more datasets of a similar nature to the ones previous will not lead to more conclusions. While there

are many different algorithms, they are all grouped together into categories such as linear algorithms, tree algorithms, etc. As established in the ending statement in Section 5.5, this paper is not simply meant to be used for the four algorithms exclusively, but to have these algorithms act as a starting point by using the groups they are a part of. That is not to say that this paper covers everything necessary, however. There would still be plenty of benefit to expand on these experiments a bit further.

This research project has shown that there is still much to be discovered in the field of regression algorithms. While there has been much work in finding new ways to improve the datasets themselves, such as new encoding methods (Seger, 2018) and problem avoidance (Jabbar et al., 2015), algorithm analysis proves to have untapped potential. A combination of desirable data-related strategies and vast algorithm-related knowledge is the best way to create optimal machine learning models.

**9.0 Acknowledgements**

I would like to give my thanks to Roberto Solis-Oba for supervising this thesis and providing support along the way. His recommendations and guidance proved to be of great help.

**10.0 References**

1. Archontoulis, S., & Miguez, F. (2015). *Nonlinear Regression Models and Applications in Agricultural Research. Agronomy Journal,* 107(2), 786–798.
2. Badole, S. (2024, March 13). *Housing price data*. Kaggle. <https://www.kaggle.com/datasets/saurabhbadole/housing-price-data>
3. Bonaccorso, G. (2017). *Machine Learning Algorithms*. Packt Publishing Ltd.
4. Domínguez-Almendros, S., & Benítez-Parejo, N. (2011). *Logistic Regression Models*. Allergologia et Immunopathologia.
5. Government of Canada. (2017). *Fuel consumption ratings* - 2015-2019 fuel consumption ratings. Open Government Portal. <https://open.canada.ca/data/en/dataset/98f1a129-f628-4ce4-b24d-6f16bf24dd64/resource/332be680-9577-42c6-8c47-a0380ef48c5e>
6. Government of Canada. (2023). *Income of individuals by age group, sex and Income Source*, Canada, provinces and selected census metropolitan areas - dataset. Open Government Portal. <https://open.canada.ca/data/en/dataset/a6ca7336-656c-4ca5-98d1-ceb9c9b6bfc9/resource/4b8eaf32-f857-4476-8042-9299024ad89>
7. Jabbar, H. K., & Khan, R. Z. (2015). *METHODS TO AVOID OVER-FITTING AND UNDER-FITTING IN SUPERVISED MACHINE LEARNING (COMPARATIVE STUDY)*
8. Karadag, D., Koc, Y., Turan, M., & Ozturk, M. (2007). *A comparative study of linear and non-linear regression analysis for ammonium exchange by Clinoptilolite Zeolite. Journal of Hazardous Materials*, 144(1–2), 432–437. <https://doi.org/10.1016/j.jhazmat.2006.10.055>
9. Massaron, L., & Boschetti, A. (2016). *Regression analysis with python: Learn the art of regression analysis with python. Packt Publishing Ltd.*
10. Maulud, D., & Abdulazeez, A. M. (2020). *A Review on Linear Regression Comprehensive in Machine Learning. Journal of Applied Science and Technology Trends*, 1(4), 140-147. <https://doi.org/10.38094/jastt1457>
11. Ogutu, J. O., Schulz-Streeck, T., & Piepho, H.-P. (2012). *Genomic Selection Using Regularized Linear Regression Models: Ridge Regression, Lasso, Elastic Net and Their Extensions*.
12. Parmar, R. (2018, July 9). *Wine quality*. Kaggle. <https://www.kaggle.com/datasets/rajyellow46/wine-quality/data>
13. Rong, S., & Bao-wen, Z. (2018). *The research of regression model in Machine Learning Field*. MATEC Web of Conferences, 176, 01033. <https://doi.org/10.1051/matecconf/201817601033>
14. Seger, C. (2018). *An investigation of categorical variable encoding techniques in machine learning: binary versus one-hot and feature hashing*