Neural Networks

Predictive modeling

ITESO, Universidad Jesuita de Guadalajara

Gregorio Alvarez

Introduction

This report aims to evaluate the predictive power and benefits of using a neural network model in a regression problem that requires a complex solution. The primary objective is to analyze the input variables statistically, test multiple linear models as benchmarks, compare their performance with a non-optimized neural network model, and ultimately utilize a grid search to identify the optimal hyperparameters for the optimizer function. The technical procedures involved in finding a regression model that can effectively fit the "Appliances Energy Prediction" dataset will be outlined.

Dataset description

The data used for this analysis is the Appliances Energy Prediction dataset from the UC Irvine Machine Learning Repository. The dataset contains 19,735 observations and 29 variables, including the target variable. The target variable is a continuous variable representing the energy consumption in Wh of appliances in a low energy building. The remaining variables are a mix of continuous and categorical variables, and are described in the table below.

Variable	Description	Type
date	date in format "yyyy-mm-dd hh:mm:ss"	categorical
Appliances	energy consumption in Wh of appliances	continuous
lights	energy consumption in Wh of light fixtures	continuous
T1	temperature in kitchen area in Celsius	continuous
RH_1	humidity in kitchen area, in percentage	continuous
T2	temperature in living room area in Celsius	continuous
RH_2	humidity in living room area, in percentage	continuous
T3	temperature in laundry room area in Celsius	continuous
RH_3	humidity in laundry room area, in percentage	continuous
T4	temperature in office room in Celsius	continuous
RH_4	humidity in office room, in percentage	continuous
T5	temperature in bathroom in Celsius	continuous
RH_5	humidity in bathroom, in percentage	continuous
T6	temperature outside the building (north side) in Celsius	continuous
RH_6	humidity outside the building (north side), in percentage	continuous
T7	temperature in ironing room in Celsius	continuous
RH_7	humidity in ironing room, in percentage	continuous
T8	temperature in teenager room 2 in Celsius	continuous
RH_8	humidity in teenager room 2, in percentage	continuous
T9	temperature in parents room in Celsius	continuous
RH_9	humidity in parents room, in percentage	continuous
T_out	temperature outside (from Chievres weather station) in Celsius	continuous

Variable	Description	Type
Pressmmhg	pressure (from Chievres weather station), in mm Hg	continuous
RH_out	humidity outside (from Chievres weather station), in percentage	continuous
Windspeed	wind speed (from Chievres weather station), in m/s	continuous
Visibility	visibility (from Chievres weather station), in km	continuous
Tdewpoint	dew point temperature (from Chievres weather station) in Celsius	continuous
rv1	random variable 1, unrelated to other variables	continuous
rv2	random variable 2, unrelated to other variables	continuous

For this report, the date and random variables were removed from the dataset. The target variable was constructed from the addition of the Appliances and lights variables. The remaining variables were used as input variables for the models.

Methods

Analysis

A pairplot was generated to examine the distribution of variables and their relationships with each other. The pairplot revealed that the majority of the variables had a unimodal distribution, several independent variables exhibited high correlation, while the dependent variable was highly skewed and showed little to no correlation with the input variable.

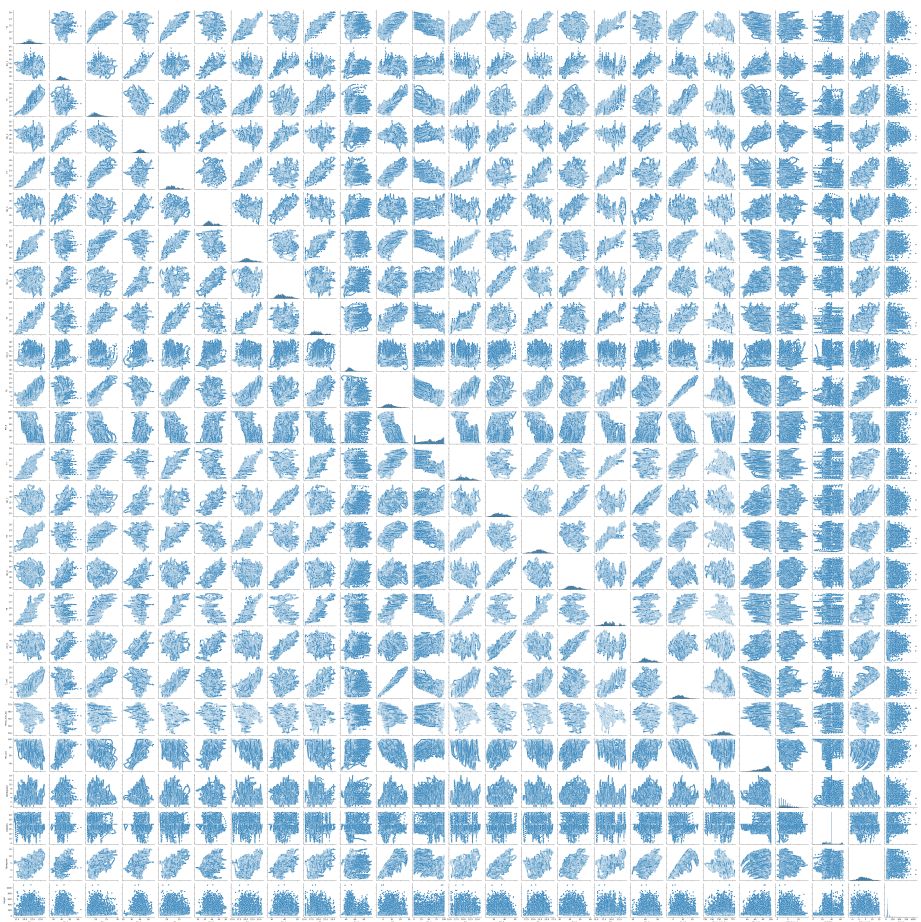

Figure 1: Pairplot of all variables in the dataset

To gain a more accurate understanding of the degree of correlation between the variables, an absolute correlation plot was also obtained. This plot further confirmed the information obtained from the pairplot.

Figure 2: Correlation plot

Benchmark models

Seven linear models were trained to check their predictive power.

- Linear regression, with and without standardization:

A linear regression model was fitted with the predictors and no transformation. It was found that the model was slower to train, than the one with standardization transformation, but the predictive power remained unchanged. Therefore, this transformation was applied to the rest of the models.

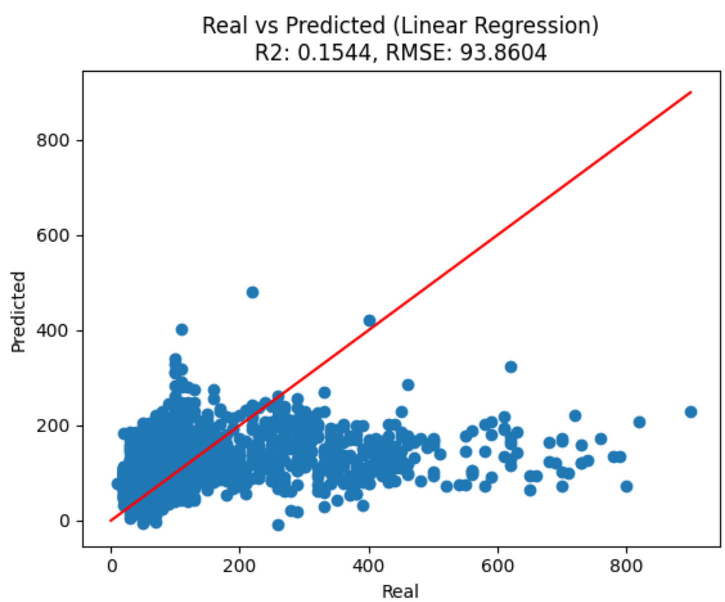

Figure 3: Linear model. It can be observed that the large skewness present in the output variable leads to large errors for large values of the variable

- Linear regression, with variable selection:

A variable selection by highest correlation between pairs, with a threshold of 0.7, was applied to the data. Which decreased the number of features by a factor of 3. The selected variables were: `RH_2, RH_5, T8, RH_9, Press_mm_hg, RH_out, Windspeed, Visibility, Tdewpoint`. As can be seen in the correlation_plot, the majority of these variables, had low linear correlation with respect to the dependent variable, indicating a possible drop of predictive power from the remaining variables.

- Partial Least Squares (PLS) Regression:

The data was fitted using PLS regression. An iterative process was employed to determine the optimal number of components. It was determined that 12 components provided the best results, indicating the "sweet spot" for this analysis.

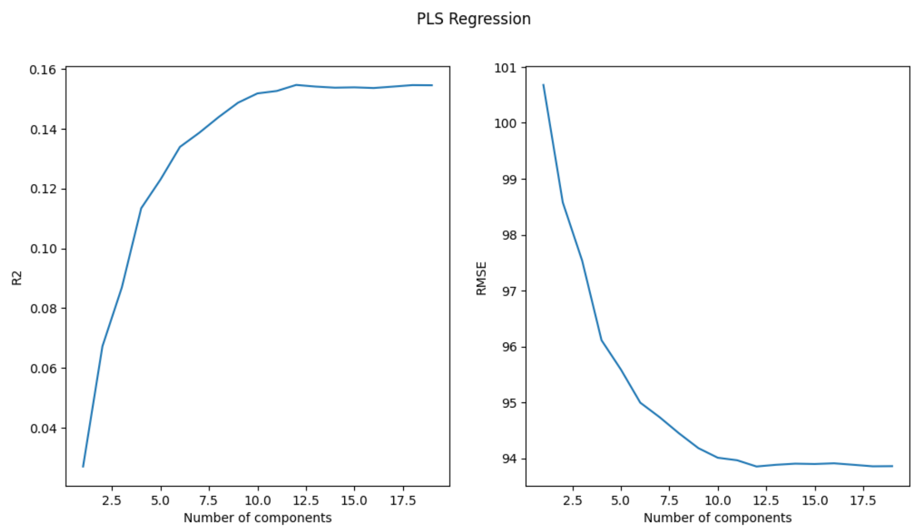

Figure 4: Score vs number of components for the PLS regression

- Transformation of target variable:

To address the high skewness of the target variable, both a logarithmic and square transformation were applied. The following procedure was followed:

1. The target variable was transformed after partitioning the data.
2. A linear model was then applied to the transformed variable.
3. The resulting predictions were transformed back to their original scale using the inverse transformation.
4. The obtained variable was then compared against the actual values for evaluation.

- Lasso Regression:

A lasso regression, with default α parameter, was used to fit the data.

- Results

Model	Test R2	Train R2	Test RMSE	Train RMSE
Regression original data	0.1544	0.1537	93.8604	96.5442
Regression normalized data	0.1544	0.1537	93.8604	96.5442
Regression with selected features	0.0325	0.0298	100.3980	103.3671
PLS regression	0.1546	0.1521	93.8516	96.6372
Regression log transformed target	0.0790	0.0968	97.9600	99.7386
Regression root square transformed target	0.1427	0.1383	94.5070	97.4197
Lasso regression	0.1533	0.1523	93.9259	96.6244

As can be seen from the previous table, the regression with the original data, the normalized data and the PLS regression have the highest accuracy, being the PLS regression the one with the highest interpretability thanks to the reduce number of components and the possibily of interpreting the components' coefficients.

Neural Network models:

The neural network architecture consisted of 500 hidden neurons, employing the hyperbolic tangent activation function. The Adam optimizer was utilized, with a learning rate of 0.01, and the mean squared error served as the loss function. To prevent saturation of the activation function, the data was standardized through normalization. Additionally, to mitigate overfitting, an early stop condition was implemented, whereby training would stop if the validation metric failed to improve within a span of 10 epochs. The table below provides a visual representation of the neural network's architecture.

Neural Network Architecture:

Layer (type)	Output Shape	Param #
dense_1 (Dense)	multiple	12500
dense_2 (Dense)	multiple	501
Total params: 13,501		
Trainable params: 13,501		
Non-trainable params: 0		

Table with number of parameters and architecture

The following table shows the results obtained from the neural network model:

```
Model: "nn_regressor_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_2 (Dense)             multiple                  12500

 dense_3 (Dense)             multiple                  501

=================================================================
Total params: 13,001
Trainable params: 13,001
Non-trainable params: 0
_________________________________________________________________
```

Figure 5: Table with number of parameters and architecture

Model	Test R2	Train R2	Test RMSE	Train RMSE	Iterations
Neural network	0.5276	0.7663	71.0212	51.0517	124
Neural network + PLS	0.5045	0.7737	72.7451	49.8580	161

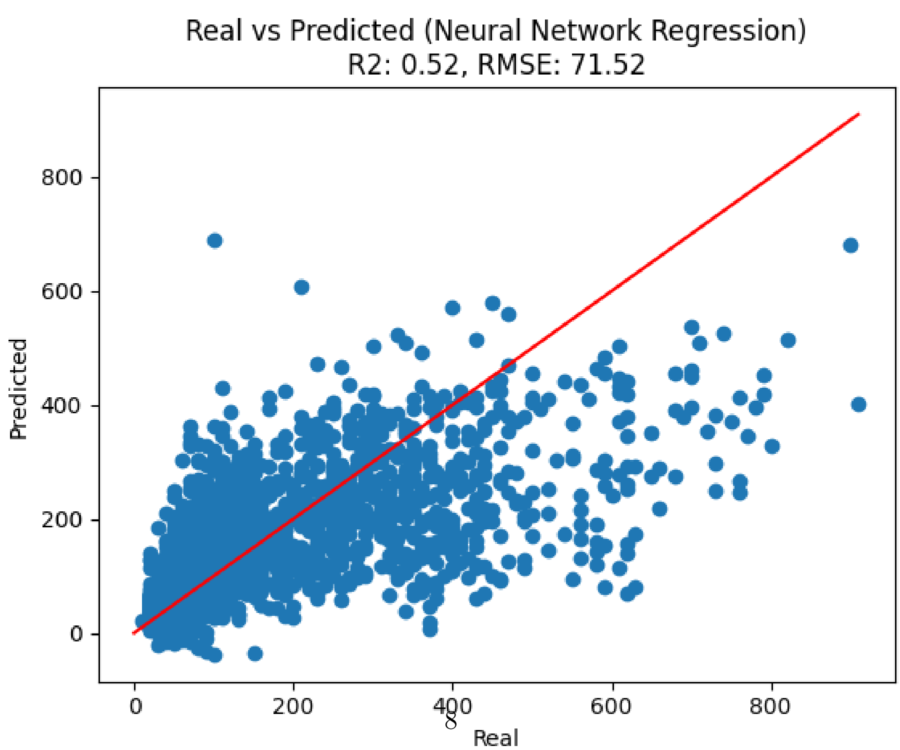

Figure 6: Linear model. As can be seen, the trend was partially captured, even with high skewness present in the output variable

Since the train and test scores are further apart a second experiment with a train, test and validation set was performed. The results are shown in the following

Model	Test R2	Train R2	Validation R2	Test RMSE	Train RMSE	Validation RMSE	Iterations
Neural network + PLS	0.4546	0.7138	0.4865	69.652	56.0751	79.8688	105

Based on the previous tables, it is evident that both the neural network with Partial Least Squares (PLS) transformation and the neural network without any transformation yield similar scores. However, the neural network without any transformation exhibits the best performance. This implies that the variable selection process after the PLS transformation eliminates at least one variable that possesses a non-linear correlation with the dependent variable.

- Grid search

A grid search, with a ten fold, was performed to find the optimal hyperparameters for the neural network. The following table shows the results obtained from the grid search:

Mean R2	Mean RMSE	Standard Deviation	Learning Rate	Epsilon
0.21	8663.86	0.001898	0.0001	1e-08
0.21	8657.54	0.002443	0.0001	1e-07
0.21	8659.28	0.00185	0.0001	1e-06
0.42	6415.83	0.001537	0.001	1e-08
0.42	6395.18	0.003956	0.001	1e-07
0.42	6395.20	0.002816	0.001	1e-06
0.35	7250.48	0.008419	0.01	1e-08
0.35	7148.44	0.01947	0.01	1e-07
0.35	7103.33	0.003568	0.01	1e-06

The results indicate that the optimal learning Rate found was 1e-3, while the value of epsilon does not seem to have a significant effect on the performance of the model.

Based on the results, it was determined that the optimal learning rate for the model was found to be 1e-3. Furthermore, it was observed that the value of epsilon did not have a significant impact on the model's performance, indicating that the range of values tested was sufficient to maintain the training stability, Further research is needed to determine if these values are sufficient for another problem.

Discussion

The results demonstrate that the neural network models outperformed the linear models by a significant margin. This can be attributed to several advantages of neural networks, including their ability to handle high dimensionality, capture non-linear relationships between variables, and effectively tackle complex problems. Additionally, the high skewness of the output variable further emphasizes the need for a more flexible and powerful modeling approach like neural networks. It was observed that the neural network models required high variability in the model in order to improve performance on the test and evaluation sets. This could be attributed to the inherent complexity of the problem being addressed. The grid search process conducted prior to this analysis indicates that the hyperparameters used in the models were either optimal or very close to being optimal.

Conclusions

It was proven that the Neural Networks can handle the high dimensionality of the data, and highly skewed output. In future studies, it would be interesting to study the output variable as a classification-regresssion problem, to handle the high skewness of the output variable. It would also be interesting to study the effect of the variable selection with PLS and logistic regression applied to these variables, as it is expected that the predictive power of the model would increase, due to the reduction of the dimensionality of the problem.