MARMARA UNIVERSITY
FACULTY OF ENGINEERING
*Moving Towards Excellence in Education and Research*

# CSE4062 2020 SPRING PROJECT

# ANALYSIS OF ELECTRIC PRODUCTION OF A WIND TURBINE

**Delivery 4: Predictive Analysis**

**Group: 8**

| Name | Department | ID Number | Mail Address |
|---|---|---|---|
| Furkan Can Ercan | Industrial Engineering | 150316044 | furkanercan98@gmail.com |
| Muhammed Avcı | Mechanical Engineering | 150414007 | muhammedavci96@gmail.com |
| Yasin Gök | Computer Engineering | 150115058 | ygok.96@gmail.com |

## 4) Predictive Analysis

In this part of our project, we established our predictive models. To do that task, we first applied feature selection with subjective judgement, we eliminated some of our useless features such as dates, hours, moonrise, moonset, sunrise and sunset features. We eliminated these features due to our previous analyses on the exploratory data analysis part. We saw that either these features are irrelevant with our target variable or it is not possible to include these variables in our models.

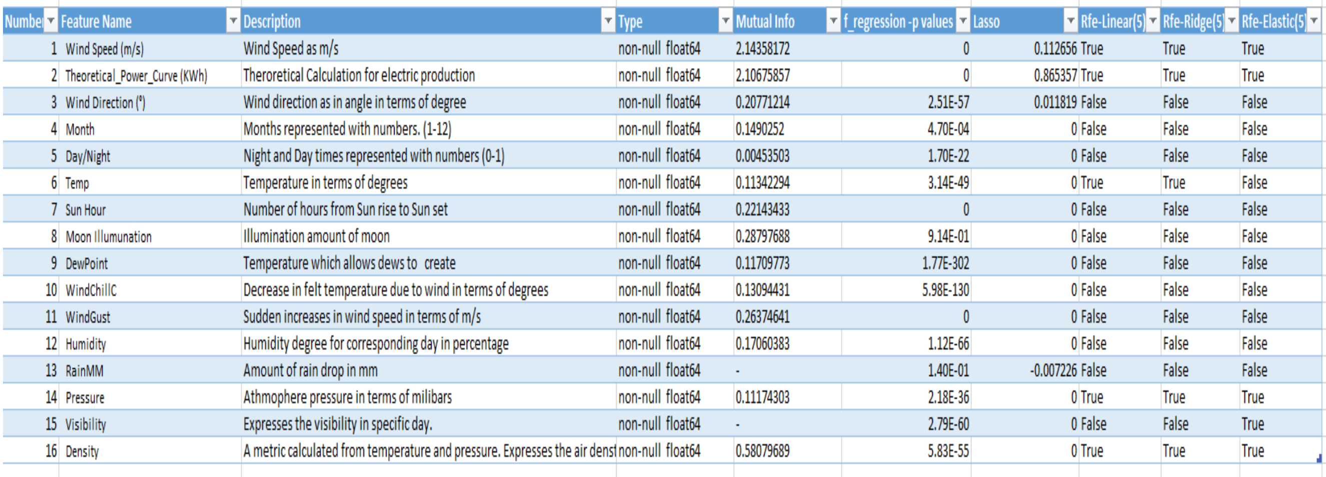
We tried to include date and hour features in our models, by using dummy variables to discretize them. This process created about 1400 new columns or features in our data and we gave up because we were both unable to process this new huge dataset in our computers and we did not believe that this new features are useful enough for our target variable based on our previous judgements in exploratory data analysis part.

After this pre-elimination of our features, we applied well-known feature selection techniques to our features or predictive variables.

We used f-regression to generate p-values about our features, which express whether there is a statistically meaningful relationship between feature variables and target variable. We accepted our confidence interval as 0.95 so for p-values smaller than the 0.05 we accepted that there is a statistically meaningful relationship. We saw that nearly all of our features are related with our target variable according to this method.

We used mutual information and selected the best 3 features with highest scores to use in our experiments. We used lasso regression to select our features. Lasso regression automatically set the useless variables' factors as zero, once fitted to the data. Lasso regression also helped us to reduce our variables to 3.

For our last feature selection method, we used recursive feature elimination, abbreviated as RFE, this method recursively reduces the number of features to a number you specify. It uses a regression algorithm and uses it to reduce number of features. We used different regression algorithms for different experiments. To select features for Ridge regression algorithm we used RFE method with Ridge regression algorithm, and used this methodology for our different algorithms as well.

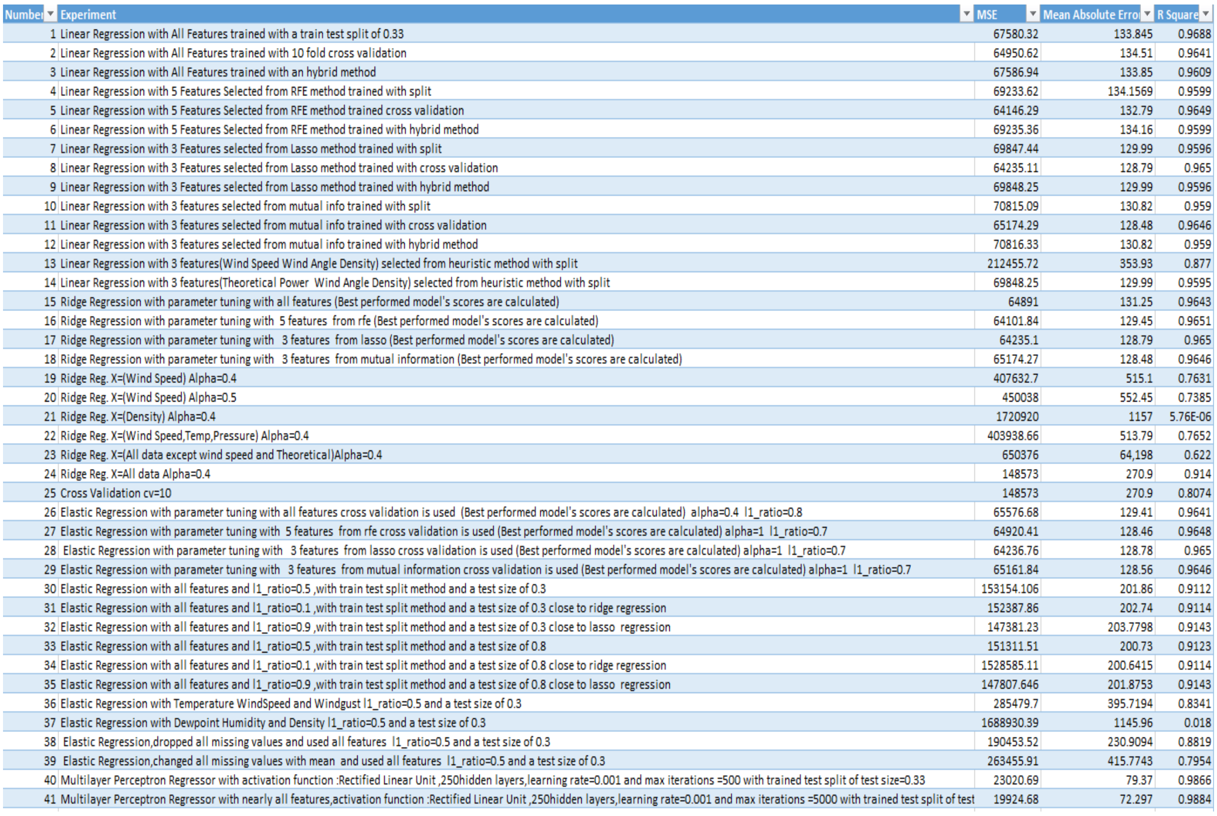


| Numbe | Feature Name | Description | Type | Mutual Info | f_regression -p values | Lasso | Rfe-Linear(5) | Rfe-Ridge(5) | Rfe-Elastic(5) |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Wind Speed (m/s) | Wind Speed as m/s | non-null float64 | 2.14358172 | 0 | 0.112656 | True | True | True |
| 2 | Theoretical_Power_Curve (KWh) | Theroretical Calculation for electric production | non-null float64 | 2.10675857 | 0 | 0.865357 | True | True | True |
| 3 | Wind Direction (°) | Wind direction as in angle in terms of degree | non-null float64 | 0.20771214 | 2.51E-57 | 0.011819 | False | False | False |
| 4 | Month | Months represented with numbers. (1-12) | non-null float64 | 0.1490252 | 4.70E-04 | 0 | False | False | False |
| 5 | Day/Night | Night and Day times represented with numbers (0-1) | non-null float64 | 0.00453503 | 1.70E-22 | 0 | False | False | False |
| 6 | Temp | Temperature in terms of degrees | non-null float64 | 0.11342294 | 3.14E-49 | 0 | True | True | False |
| 7 | Sun Hour | Number of hours from Sun rise to Sun set | non-null float64 | 0.22143433 | 0 | 0 | False | False | False |
| 8 | Moon Illumunation | Illumination amount of moon | non-null float64 | 0.28797688 | 9.14E-01 | 0 | False | False | False |
| 9 | DewPoint | Temperature which allows dews to create | non-null float64 | 0.11709773 | 1.77E-302 | 0 | False | False | False |
| 10 | WindChillC | Decrease in felt temperature due to wind in terms of degrees | non-null float64 | 0.13094431 | 5.98E-130 | 0 | False | False | False |
| 11 | WindGust | Sudden increases in wind speed in terms of m/s | non-null float64 | 0.26374641 | 0 | 0 | False | False | False |
| 12 | Humidity | Humidity degree for corresponding day in percentage | non-null float64 | 0.17060383 | 1.12E-66 | 0 | False | False | False |
| 13 | RainMM | Amount of rain drop in mm | non-null float64 | - | 1.40E-01 | -0.007226 | False | False | False |
| 14 | Pressure | Athmophere pressure in terms of milibars | non-null float64 | 0.11174303 | 2.18E-36 | 0 | True | True | True |
| 15 | Visibility | Expresses the visibility in specific day. | non-null float64 | - | 2.79E-60 | 0 | False | False | True |
| 16 | Density | A metric calculated from temperature and pressure. Expresses the air denst | non-null float64 | 0.58079689 | 5.83E-55 | 0 | True | True | True |

Figure 4.1 Feature Selection Table

After selecting our features, we implemented different scenarios and experimented them. We used Linear Regression with Ordinary Least Squares method, Ridge Regression with a penalized coefficient of alpha, to penalize bigger parameters and prevent overfitting, Elastic Regression with a mixture both L1 and L2 penalization methods and lastly we implemented Multilayer Perceptrons  with Rectified Linear Unit activation function.

We used 3 different approaches for training, 10 fold cross validation, traditional train-test split approach and hold out method which is a hybrid of our first 2 approaches. We calculated R-Scores, mean squared errors and mean absolute errors for our different experiments. We demonstrated our experiments and their results below.



| Numbe | Experiment | MSE | Mean Absolute Erro | R Square |
|---|---|---|---|---|
| 1 | Linear Regression with All Features trained with a train test split of 0.33 | 67580.32 | 133.845 | 0.9688 |
| 2 | Linear Regression with All Features trained with 10 fold cross validation | 64950.62 | 134.51 | 0.9641 |
| 3 | Linear Regression with All Features trained with an hybrid method | 67586.94 | 133.85 | 0.9609 |
| 4 | Linear Regression with 5 Features Selected from RFE method trained with split | 69233.62 | 134.1569 | 0.9599 |
| 5 | Linear Regression with 5 Features Selected from RFE method trained cross validation | 64146.29 | 132.79 | 0.9649 |
| 6 | Linear Regression with 5 Features Selected from RFE method trained with hybrid method | 69235.36 | 134.16 | 0.9599 |
| 7 | Linear Regression with 3 Features selected from Lasso method trained with split | 69847.44 | 129.99 | 0.9596 |
| 8 | Linear Regression with 3 Features selected from Lasso method trained with cross validation | 64235.11 | 128.79 | 0.965 |
| 9 | Linear Regression with 3 Features selected from Lasso method trained with hybrid method | 69848.25 | 129.99 | 0.9596 |
| 10 | Linear Regression with 3 features selected from mutual info trained with split | 70815.09 | 130.82 | 0.959 |
| 11 | Linear Regression with 3 features selected from mutual info trained with cross validation | 65174.29 | 128.48 | 0.9646 |
| 12 | Linear Regression with 3 features selected from mutual info trained with hybrid method | 70816.33 | 130.82 | 0.959 |
| 13 | Linear Regression with 3 features(Wind Speed Wind Angle Density) selected from heuristic method with split | 212455.72 | 353.93 | 0.877 |
| 14 | Linear Regression with 3 features(Theoretical Power  Wind Angle Density) selected from heuristic method with split | 69848.25 | 129.99 | 0.9595 |
| 15 | Ridge Regression with parameter tuning with all features (Best performed model's scores are calculated) | 64891 | 131.25 | 0.9643 |
| 16 | Ridge Regression with parameter tuning with  5 features  from rfe (Best performed model's scores are calculated) | 64101.84 | 129.45 | 0.9651 |
| 17 | Ridge Regression with parameter tuning with   3 features  from lasso (Best performed model's scores are calculated) | 64235.1 | 128.79 | 0.965 |
| 18 | Ridge Regression with parameter tuning with   3 features  from mutual information (Best performed model's scores are calculated) | 65174.27 | 128.48 | 0.9646 |
| 19 | Ridge Reg. X=(Wind Speed) Alpha=0.4 | 407632.7 | 515.1 | 0.7631 |
| 20 | Ridge Reg. X=(Wind Speed) Alpha=0.5 | 450038 | 552.45 | 0.7385 |
| 21 | Ridge Reg. X=(Density) Alpha=0.4 | 1720920 | 1157 | 5.76E-06 |
| 22 | Ridge Reg. X=(Wind Speed,Temp,Pressure) Alpha=0.4 | 403938.66 | 513.79 | 0.7652 |
| 23 | Ridge Reg. X=(All data except wind speed and Theoretical)Alpha=0.4 | 650376 | 64,198 | 0.622 |
| 24 | Ridge Reg. X=All data Alpha=0.4 | 148573 | 270.9 | 0.914 |
| 25 | Cross Validation cv=10 | 148573 | 270.9 | 0.8074 |
| 26 | Elastic Regression with parameter tuning with all features cross validation is used  (Best performed model's scores are calculated)  alpha=0.4  l1_ratio=0.8 | 65576.68 | 129.41 | 0.9641 |
| 27 | Elastic Regression with parameter tuning with  5 features  from rfe cross validation is used (Best performed model's scores are calculated) alpha=1  l1_ratio=0.7 | 64920.41 | 128.46 | 0.9648 |
| 28 | Elastic Regression with parameter tuning with   3 features  from lasso cross validation is used (Best performed model's scores are calculated) alpha=1  l1_ratio=0.7 | 64236.76 | 128.78 | 0.965 |
| 29 | Elastic Regression with parameter tuning with   3 features  from mutual information cross validation is used (Best performed model's scores are calculated) alpha=1  l1_ratio=0.7 | 65161.84 | 128.56 | 0.9646 |
| 30 | Elastic Regression with all features and l1_ratio=0.5 ,with train test split method and a test size of 0.3 | 153154.106 | 201.86 | 0.9112 |
| 31 | Elastic Regression with all features and l1_ratio=0.1 ,with train test split method and a test size of 0.3 close to ridge regression | 152387.86 | 202.74 | 0.9114 |
| 32 | Elastic Regression with all features and l1_ratio=0.9 ,with train test split method and a test size of 0.3 close to lasso  regression | 147381.23 | 203.7798 | 0.9143 |
| 33 | Elastic Regression with all features and l1_ratio=0.5 ,with train test split method and a test size of 0.8 | 151311.51 | 200.73 | 0.9123 |
| 34 | Elastic Regression with all features and l1_ratio=0.1 ,with train test split method and a test size of 0.8 close to ridge regression | 1528585.11 | 200.6415 | 0.9114 |
| 35 | Elastic Regression with all features and l1_ratio=0.9 ,with train test split method and a test size of 0.8 close to lasso  regression | 147807.646 | 201.8753 | 0.9143 |
| 36 | Elastic Regression with Temperature WindSpeed and Windgust l1_ratio=0.5 and a test size of 0.3 | 285479.7 | 395.7194 | 0.8341 |
| 37 | Elastic Regression with Dewpoint Humidity and Density l1_ratio=0.5 and a test size of 0.3 | 1688930.39 | 1145.96 | 0.018 |
| 38 | Elastic Regression,dropped all missing values and used all features  l1_ratio=0.5 and a test size of 0.3 | 190453.52 | 230.9094 | 0.8819 |
| 39 | Elastic Regression,changed all missing values with mean  and used all features  l1_ratio=0.5 and a test size of 0.3 | 263455.91 | 415.7743 | 0.7954 |
| 40 | Multilayer Perceptron Regressor with activation function :Rectified Linear Unit ,250hidden layers,learning rate=0.001 and max iterations =500 with trained test split of test size=0.33 | 23020.69 | 79.37 | 0.9866 |
| 41 | Multilayer Perceptron Regressor with nearly all features,activation function :Rectified Linear Unit ,250hidden layers,learning rate=0.001 and max iterations =5000 with trained test split of test | 19924.68 | 72.297 | 0.9884 |

Figure 4.2 Experiments Table

From our experiments we realized that our best performing model is Multilayer Perceptron with 250 hidden layers and maximum of 5000 iterations but we also saw that although Multilayer Perceptron is the best performing model its computational complexity is really high. Since it has 250 hidden layers and 5000 iterations it is significantly takes much more time than other models. When we try to lower the number of iterations, we got an error which says that the network optimization procedure has not converged yet. This means that we need to decide whether the amount of improvement provided by the Multilayer Perceptron is enough for us to accept this high computation time.

Other than that we experimented with Ridge Regression and Elastic Regression algorithms which are normally better models than Linear Regression, and saw that the hyperparameter tuning is a very important process. We realized that when we do not set our parameters carefully Ridge and Elastic Regression performs very poorly in our cases worse than Linear Regression. After we made our hyperparameter tuning phase, we saw that this time Ridge and Elastic Regression algorithms performs very well. Again we have a decision to make because again we will need more computational time to make our hyperparameter tuning phase for these algorithms.

To see if there is a statistically meaningful difference between models we need to use t test but our best performed model takes too much time to run for a cross validation. For one model it takes nearly 15 minutes to run and for cross validation it needs to run 10 times. After considering this we compared the best 2 models which is other than Multilayer Perceptron.

We compared one of our Linear Regression and Ridge Regression experiments as our 2 of the best competitors. Ridge Regression was slightly better than Linear Regression. We made our test for r score values. In the end we obtained a p-value of 0.231 from our t-test. Our confidence interval was 0.95 and to accept there is a meaningful difference between means , as our hypothesis we needed a value smaller than 0.05 but we obtained a bigger p-value. This mean there is no evidence for our claim.

In these circumstances even though Ridge Regression model performed slightly better, there is not a statically meaning difference and after considering that Ridge Regression computationally more expensive and it also needs a hyper parameter tuning phase we can use our Linear Regression model. Because the performance difference between them is not a meaningful difference and Linear Regression has its computational time advantage.

In the end we saw that for our dataset the further effort we made with Multilayer Perceptron and other regression algorithms does not seem to worth the extra effort(computational complexity and computational time). We can use our best performed Linear Regression Model. However if we neglect the computational costs we should definitely prefer Multilayer Perceptron. Even though there is not much difference, Multilayer Perceptron is our best model.