**Research Question**

*Research Question:* Can a regression model be constructed to predict electrical energy output from a combined-cycle power plant? The null hypothesis for this question states a regression model cannot be constructed, while the alternative hypothesis asserts a regression model can be constructed to predict this energy output. A data analysis is beneficial to answering this question because it allows power companies to more accurately predict power output, which reduces issues such as power outages, as well economic and technical difficulties (Siddiqui et al., 2021). Combined Cycle Power Plants (CCPP’s) are becoming more popular around the world for their greater efficiency and lower waste: Ramireddy (2012) found that CCPP’s can generate more than double the amount of electricity as conventional power plants. This study will use a multiple linear regression model to evaluate which, if any, variables can predict the plant’s electrical output. Multiple linear regression models predict the value of a dependent variable at a certain value of two or more independent variables (Bevans, 2020).

**Data Collection**

The data for this study can be found [here](https://archive.ics.uci.edu/ml/datasets/Combined+Cycle+Power+Plant). It is made publicly available by the UC Irvine Machine Learning Repository, a “collection of databases, domain theories, and data generators that are used by the machine learning community for the empirical analysis of machine learning algorithms” (Dua and Graff, 2019). The data set includes predictor variables of Ambient Temperature (AT), Ambient Pressure (AP), Relative Humidity (RH) and Exhaust Vacuum (V), and target variable Electrical Output (PE). The data was collected from a Combined Cycle Power Plant over 6 years (2006-2011) using sensors located around the plant that record these variables every second. There are 9568 rows, and five columns (variables) which are broken down as follows:

|  |  |
| --- | --- |
| **Field** | **Type** |
| Ambient Temperature (AT) | Continuous |
| Ambient Pressure (AP) | Continuous |
| Relative Humidity (RH) | Continuous |
| Exhaust Vacuum (V) | Continuous |
| Net Hourly Electrical Energy Output (PE) | Continuous |

Data will be downloaded as a Microsoft Excel (xlsx) file from the UCI Machine Learning Repository website, which is part of the University of California, Irvine. Although this is a very simple and uncomplicated data collection method, one disadvantage of collecting data this way is that the accuracy of the data is entirely reliable on this organization. Furthermore, there is no way to verify the accuracy. Navigating the organization’s website and dealing with small mistakes (for example, variable “PE” is actually listed as “EP” in the data set description) provided brief challenges, but they were easy enough to work through.

They were also advantages to this data-gathering methodology. Because the data is used specifically for machine learning tasks, it is complete data that does not need cleaning. Also, the University of California is a very reputable and respected university, and it is safe to assume we can trust the validity of the data that they have provided.

**Data Extraction and Preparation**

I used Python programming language in a Jupyter notebook to assist with this data analysis; specifically, the Pandas, NumPy, Matplotlib, Seaborn, Sklearn, SciPy, and Statsmodel libraries were utilized. As a general-purpose language, Python allows for the exploration and complete assessment of the data. The code and syntax in Python and pandas is readable and comprehensible, and is better suited for machine learning than R is (*Python vs. R: What's the Difference*?, 2021). At the same time, R is typically superior for statistical and data analysis (Johnson, 2022) and could potentially provide more in-depth outcomes than Python. However, with the aid of the Statsmodel library, which can conduct statistical tests, estimate a regression model, and produce a list of results for each estimator (Andrade, 2021), Python is perfectly capable of handling this analysis.

I used Python programming language. I started by importing the libraries and packages that I would need.

Text

Description automatically generated

Then, I loaded the data into a data frame and viewed a preview of the first 5 rows.

Table

Description automatically generated

I programmatically and visually inspected the data for accuracy and completeness.

Text

Description automatically generated

Graphical user interface, application, table, Excel

Description automatically generated

Chart, box and whisker chart

Description automatically generated

I ran a Shapiro-Wilk test using the Stats sub-package of the SciPy library to determine whether the data comes from a normal distribution. Based on the low p-value (below 0.05), we have evidence that the sample of the data does not come from a normal distribution. Therefore, we need to normalize the data.

Graphical user interface, text, application, email

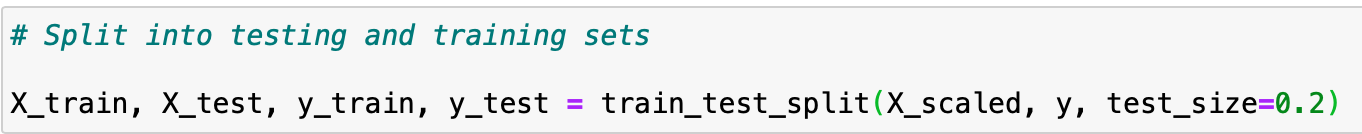
Description automatically generated

The independent variables will be standardized using Z-score normalization (or Standard Scaling) using the Sklearn library (a/k/a StandardScaler). Standardization is a vital preprocessing step when variables which are on different scales interact (Caron, 2020). Based on the variables’ differing ranges seen in the histogram plots, it is clear these variables are using different scales, and standardizing these variables is necessary.

Text

Description automatically generated

After standardizing the data, it was split the into training and test sets, which allows the model to be trained on one set of rows, and then be tested on a separate set. This provides a tool to measure the model’s fit and accuracy.



Finally, I added a constant to the training set and printed a summary of the model, seen on the following page.

Graphical user interface, text, table

Description automatically generated

**Analysis**

The training data set was used to create a regression model using the Statsmodel library, which utilizes the ordinary least squares (OLS) method. This linear regression method estimates the parameters of the model by minimizing the sum of the squared differences between the observed values and the model’s fitted values (Frost, 2017). Linear regression is easy to implement in Python and has a very interpretable output, as will be shown in the next section. However, one disadvantage of any linear regression is that not all systems are linear, and it’s very possible that the subject of our analysis is one of those systems.

One assumption of multiple linear regression models is that no multicollinearity among independent variables exists. We can measure multicollinearity by calculating the variance inflation factors (VIF’s). Generally, variables with VIF’s above 10 indicate high correlation and are considered problematic (*Variance Inflation Factor*, 2020). Fortunately, it appeared no concerning multicollinearity exists among these variables. While this calculation may tell us that our independent variables are truly completely independent from one another, we cannot ignore the fact that air conditions such as humidity, temperature, pressure, etc. often do move together, and therefore may be dependent on each other. This possible embedded multicollinearity is another potential disadvantage to our analysis technique.

The final step in our analysis was to check how accurate our model was. Below, the first ten rows of a data frame of the predicted values and actual values (along with the difference between them, or residual) is shown.

Table

Description automatically generated

Below, a graph showing the electrical output vs. the corresponding residual. This shows that our residuals, or differences between the observed values and the values our model predicted, are relatively small.

Chart, scatter chart

Description automatically generated

Below is a graph of predicted values vs. the observed (actual) values. We can see that our predicted values are close to the observed values, making the mean residual error relatively low.

Chart, scatter chart

Description automatically generated

**Data Summary and Implications**

Based on the above results, I would contend that we have successfully constructed a regression model to predict electrical energy output from a combined-cycle power plant. A R-Squared value of 0.929 implies that 92.9% of electrical output is explained by changes in the dependent variables. In other words, 92.9% of our model fit the actual observed data. That is a very strong model, and we can predict electrical output based on these variables with a high level of confidence.

*Equation of final regression model:*

**Net Hourly Electrical Energy Output** = 454.3551 + Ambient Temperature (\* -14.7106) + Exhaust Vacuum (\* -2.9531) + Ambient Pressure (\*0.3622) + Relative Humidity (\* -2.3112)

This equation tells us that temperature has the biggest effect on energy output (with higher temperatures creating less energy), while ambient pressure has the smallest effect (with higher pressure creating more energy). Exhaust vacuum and relative humidity have moderate effects, with higher vacuum and humidity correlated with lower energy output. Thus, maximum energy output is correlated with conditions of low temperature, exhaust vacuum and relative humidity, and high ambient pressure.

One significant limitation of this analysis is that this data is entirely comprised of rows from one single Combined Cycle Power Plant. This could lead to conclusions that may be true for this specific plant, but not to any others. Accordingly, one course of action I would recommend is to take these measurements at other Combined Cycle Power Plants and attempt to answer the same research question with the newly gathered data. If we see similar trends, we can begin to consider how we can maximize electrical output (i.e., recommending locations for CCPP’s that most often have favorable conditions for electrical output). Regarding this specific data set, there are also other approaches that could be taken. In the future, I would propose using other types of regression techniques such as lasso regression, or an algorithm like k-means clustering to analyze this data. I may also suggest incorporating Tableau into this analysis; specifically, it may be helpful to create a world map that shows averages of the independent variables in this data set (temperature, pressure, exhaust vacuum, humidity).

References

Andrade, F. (2021, October 18). *A Simple Guide to Linear Regression using Python*. Medium. Retrieved May 1, 2022, from <https://towardsdatascience.com/a-simple-guide-to-linear-regression-using-python-7050e8c751c1#0fa8>

Bevans, R. (2020, February 20). *Multiple Linear Regression | A Quick and Simple Guide*. Scribbr. Retrieved March 31, 2022, from <https://www.scribbr.com/statistics/multiple-linear-regression/>

Dua, D. and Graff, C. (2019). *UCI Machine Learning Repository* [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science.

Frost, J. (2017, May 5). *Ordinary Least Squares*. Statistics By Jim. Retrieved March 31, 2022, from <https://statisticsbyjim.com/glossary/ordinary-least-squares/>

IBM Cloud Team. (2021, March 23). *Python vs. R: What's the difference?* IBM. Retrieved April 5, 2022, from <https://www.ibm.com/cloud/blog/python-vs-r>

Johnson, D. (2022, March 8). *R vs Python: What's the difference?* Guru99. Retrieved May 1, 2022, from <https://www.guru99.com/r-vs-python.html>

Ramireddy, V. (2012, August 25). *An Overview of Combined Cycle Power Plant*. EEP - Electrical Engineering Portal. Retrieved May 1, 2022, from <https://electrical-engineering-portal.com/an-overview-of-combined-cycle-power-plant>

Statistics How To. (2020, December 16). *Variance Inflation Factor*. Retrieved May 1, 2022, from <https://www.statisticshowto.com/variance-inflation-factor/>

Siddiqui, R., Anwar, H., Ullah, F., Ullah, R., Rehman, M. A., Jan, N., & Zaman, F. (2021, December 23). *Power Prediction of Combined Cycle Power Plant (CCPP) Using Machine Learning Algorithm-Based Paradigm*. Wireless Communications and Mobile Computing. Retrieved May 1, 2022, from <https://www.hindawi.com/journals/wcmc/2021/9966395/>