**Used libraries:**

- numpy
- pandas
- matplotlib
- seaborn
- scikit-learn
- Statsmodels
- datetime
- warnings

This is a report on the ten tasks assigned in the data analytics assignment. It includes the steps, the outcome, and my personal perspective on each task.

**Task1:**

The work in this section consisted of downloading historical daily weather data for France and importing it into my environment. In addition, to fill any gaps in the data, I had to use linear interpolation. I completed this section by performing the following steps:

1. I obtained the daily weather data set via a link provided.
2. I imported it as a Data Frame into the Python web-based interactive development environment Jupyter
3. I examined the columns of this data frame and the number of missing data in each column.
4. To reduce the number of missing values, I removed unnecessary columns and then used the panda's interpolation function, which estimates missing values using the linear interpolation method.

**Results:** The provided data set was successfully imported, and it had 365 rows and 21 columns. The information that this data set provided were the daily measurements of the

- Temperature (high, average, and low temperature) in degree Celsius
- Dew point (the amount of moisture in the air) in degree Celsius
- Humidity (high, average, and low humidity) in percentage
- Sea level pressure (high, average, and low sea level pressure) in hectopascal
- Visibility (high, average, and low visibility) in kilometres
- Wind speed (high, average, and low temperature as well as high gust wind) in kilometres per hour
- Precipitation in millimetres
- Events (whether it rained, snowed or if there was Fog)

After removing the superfluous columns like high gust wind and events, we were left with 19 columns.

**Insight:** We removed these two columns because they had a large amount of missing data, and the available data would not be useful in our future analysis.

**Task 2:**

The main goal of this section was to compute the correlation matrix between all of the weather variables and to create a graph that displayed the correlation matrix as a heat-map. The steps below were followed to bring this section through completion

1. Through the cleaned daily weather data frame in the first part, I used the correlation pandas' function to calculate the correlation matrix and the result was loaded into a variable
2. Then I plotted the created variable as a heat map from the seaborn library

**Result:** The resulted graph shows the relationship between all the weather variables based on the colours and their relationship is shown by a correlation coefficient which is between -1 and 1 that indicates the strength and direction of the linear relationship between these variables.

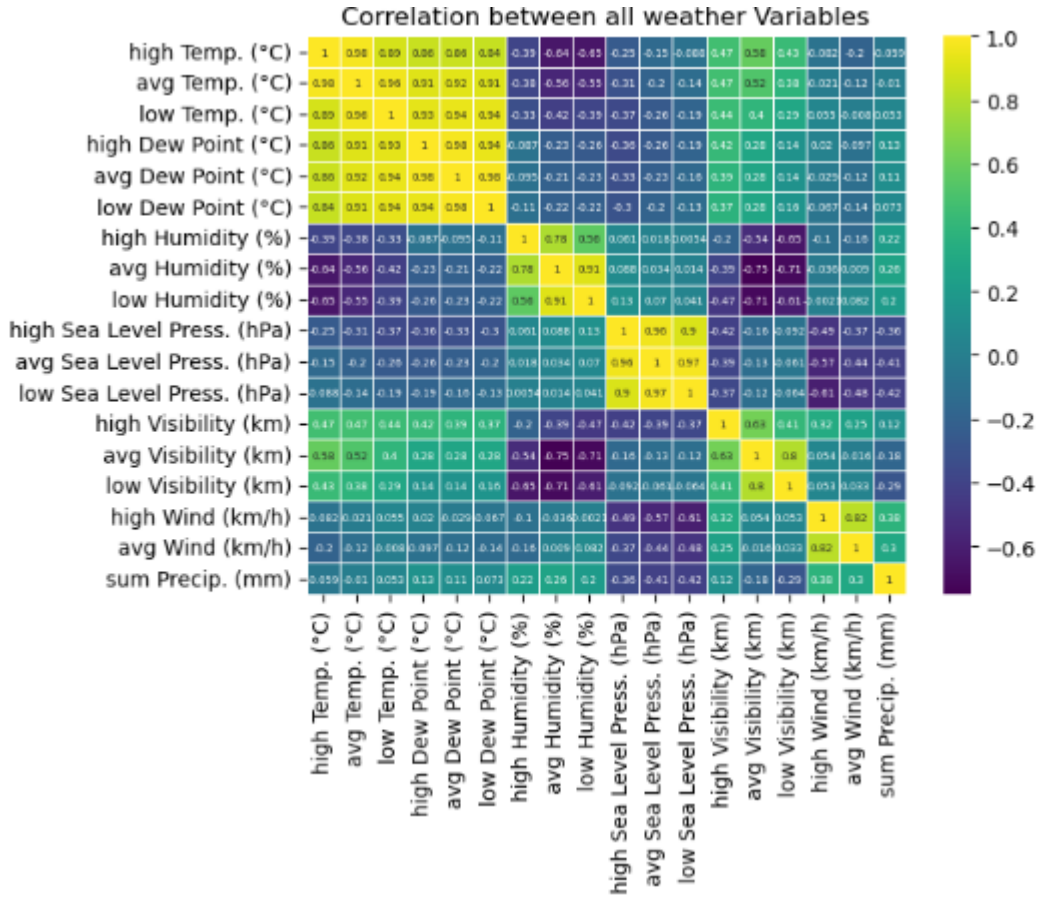


*Figure 1: A heatmap that shows the correlation between 2017 daily weather variables in France*

**Insight:** As shown in figure 1, temperature has a positive relationship with the dew point (the amount of moisture in the air), as well as a small positive relationship with visibility and a negative relationship with humidity. The temperature has almost no relationship with the sea level pressure, wind speed, or precipitation. A positive relationship means directly proportional, as in as the temperature rises, the dew point rises, and a negative relationship means inversely proportional, as in as the temperature rises, the humidity falls.

**Task 3 and 4:**

In this task, I had to download a 2017 historical daily electricity consumption data set for France and load it into my environment so that I could use it alongside the daily weather data set that I had previously loaded. Then I had to synchronize the dates for both time series and create a scatter plot of energy consumption versus mean temperature. This was accomplished by following the steps outlined below.

1. I obtained the daily energy consumption data set via a link provided and I converted it from excel formal to csv to easily manipulate it in my environment
2. I imported it as a Data Frame into the Python web-based interactive development environment Jupyter and cleaned this data frame by skipping unfilled rows and deleting empty columns
3. I also deleted an unnecessary column for our analysis which included the type of data
4. I examined the columns of this data frame and the number of missing data in each column.
5. I gave both data frames the same data format and then merged them together into one data frame
6. Then I plotted a scatter plot of the daily energy consumption against the mean daily temperature

**Results:** The resulting graph shows a negative relationship between daily average temperature and daily electricity consumption. In fact, when the temperature was low, more energy was consumed.

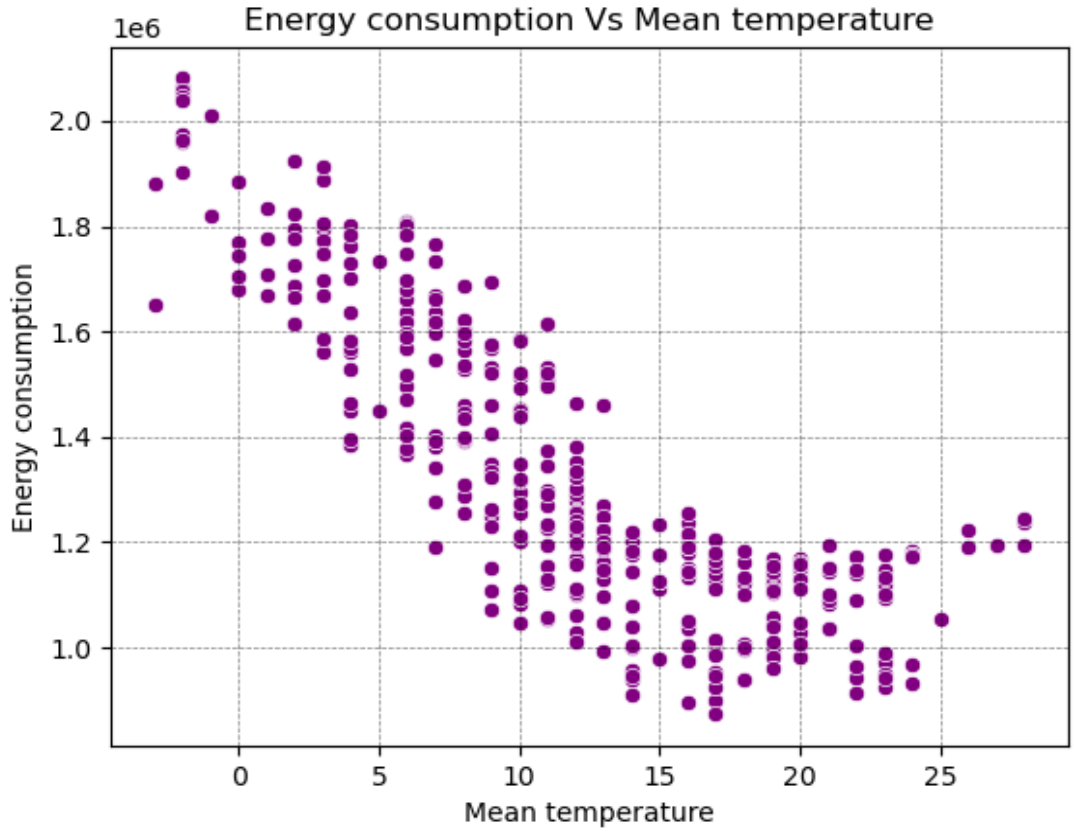


*Figure 2: This graph shows the relationship between the 2017 daily energy consumption and the daily average temperature in France*

**Insight:** The temperature is lower during the winter, and the reason for the high energy consumption during this time is that it was mostly used for indoor heating. Furthermore, most people spend the day indoors during this time, which may result in the use of lights, televisions, and other home appliances that consume when left on for an extended period.

**Task 5:**

The task in this section was to fit a quadratic model to the energy versus temperature data and plot the quadratic fit as a line on top of the scatter plot created in part 4, which was accomplished through the use of the procedures outlined below.

1. I found the constant from this polynomial regression $y = B2x + B1x + B0$ thanks to the numpy function polyfit () and I considered the average temperature to be the independent variables and the daily consumption to be the dependent variable
2. I created a line that ranges from -5 to 30 and I used it to create a polynomial line from the formula $y = B2x + B1x + B0$ by considering x as the created line and the resulted constant in step 1
3. Then plotted the quadratic fit as a line on top of the scatter plot created in part 4

**Result:** The purple balls represent the true measured values of the daily energy consumption against the mean temperature while the blue line represents the values obtained after minimizing errors between the measured values and predictions

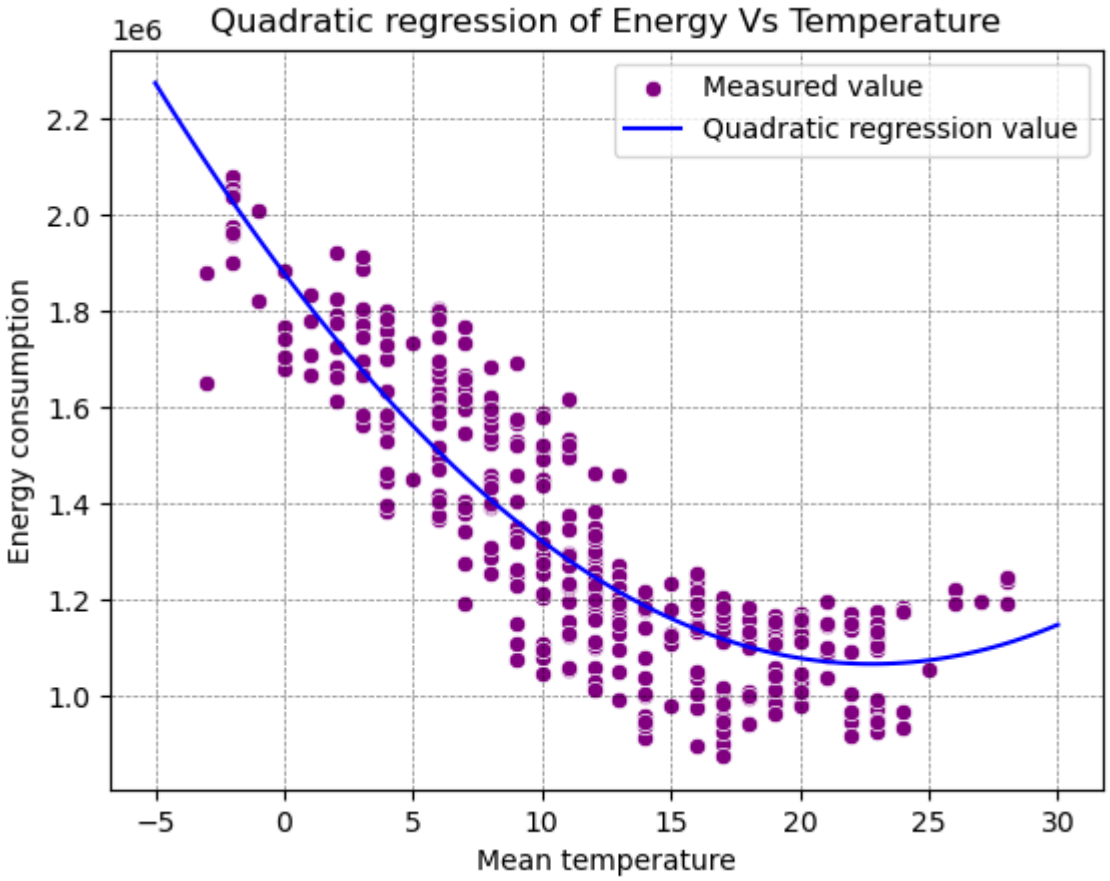


*Figure 3: This graph shows the relationship between the 2017 both the measured value of the daily energy consumption and the daily average temperature in France as well as their quadratic regression predicted value*

**Insight:** This quadratic line has assisted in capturing some trends that were previously difficult to capture; here, I can clearly identify what was the consumption at any temperature I choose which was quite complicated before.

**Task 6:**

The task in this section was to use empirical analysis to determine the optimal temperature that corresponded with the least amount of consumption and to visually check it using the quadratic fit. I completed it by following the steps outlined below.

1. Using the numpy function 'polyfit,' I created a polynomial function of degree 2 that fits the independent variable (mean temperature) and the dependent variable (daily energy consumption). The resulting polynomial function was assigned to the variable.
2. I applied the polynomial function result to the estimated variable (the line I created in the previous task) and obtained the minimum temperature value of 22.9293 degrees Celsius.
3. The result was then plotted by pointing to the optimal temperature that corresponded with the lowest consumption. In fact, I took into account the minimum value of the polynomial function result that I applied to the estimated value and its associated temperature.

**Result:** The results showed that the daily energy consumption was $1065901.948MWh$ and its corresponding optimal minimum temperature is $22.9293^{o}C$ as shown in figure 4 below.

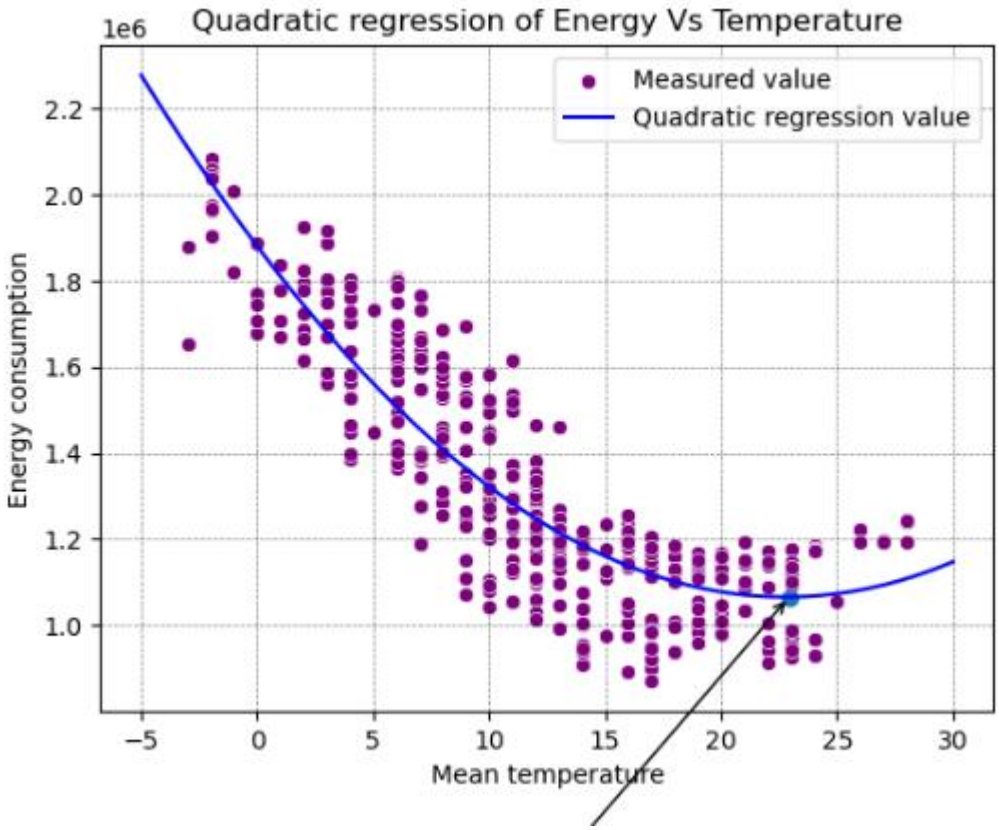


*Figure 4:The minimum optimal temperature that correspond with the daily energy consumption*

**Task 7:**

The task here was to use a stepwise approach to find an optimal multivariate linear regression model using weather variables to forecast consumption and to determine the selected variables as well as the coefficient of determination, which was accomplished through the following steps.

1. I separated the independent variable a (all weather variables) and the dependent variable (daily energy consumption) and then checked for missing data, there was none.
2. To find the predictors that improve my model's fit the most, I used a forward stepwise regression.
3. I found the coefficient of determination ( $R^2$ ) to measure how well the model fits my data. This was found using two methods because I wanted to compare their answers. Firstly, I used '. score ()' method from scikit-learn library then I used'. rsquared ( )' method provided by Statsmodels library. Therefore, the answer I obtained from this method were the same.

**Results:** Seven weather variables were chosen as the best predictors for our model, with a coefficient of determination of 0.75064. All of the variables that were chosen as best predictors are listed below.

- The daily measured high temperature
- The daily measured average temperature
- The daily measured high visibility
- The daily measured high humidity
- The daily measured low humidity
- The daily measured average dew point
- The daily low sea level pressure measured

**Insight:** The number of variables chosen is as small as possible while still adequately explaining our response variable. In fact, the majority of the chosen values have a positive relationship.

**Task 8:**

The goal was to increase the number of explanatory variables by including squared terms for each weather variable, and then to generate a new model using a stepwise approach. Following that, I had to identify the variables that had been chosen, calculate the new R2 value, and provide my opinion. This was accomplished through the following steps:

1. I created a data frame that consist of the squared values of each weather variables and then I combined it to the independent variables used in the previous task
2. To find the predictors that improve my model's fit the most, I used a forward stepwise regression.
3. I found the coefficient of determination ( $R^2$ ) to measure how well the model fits my data using the same two methods I used in the previous task

**Results:** Four weather variables were chosen as the best predictors for our model, with a coefficient of determination of 0.8068. All of the variables that were chosen as best predictors are listed below.

- The value of the daily measured high temperature
- The value of the daily measured high temperature squared
- The value of the daily measured high visibility
- The value of the daily measured high visibility squared

**Insight:** The number of variables chosen is kept as low as possible while still adequately explaining our response variable, and all of the values are related in a positive way.

**Task 9:**

The task was to consider the day of the week effect in the multivariate regression by including dummy variables for the day of the week, determine the days of the week chosen for the new model, as well as the new R2 value, and provide my opinion. To finish this section, I followed the steps outlined below:

1. I created a data frame that consist of all the 2017 dates and each day of the week as the columns. I set the dates as the index and marked down each day of the week and its corresponding day oof the week.
2. I merged the created data frame with the one that consist of the weather variables as well as the squared values of each weather.
3. To find the predictors that improve my model's fit the most, I used a forward stepwise regression as I did in the previous tasks but I considered the data frame created in step1.
4. I found the coefficient of determination ( $R^2$ ) to measure how well the model fits my data using the same two methods I used in the previous tasks

**Results:** Eleven weather variables were chosen as the best predictors for our model, with a coefficient of determination of 0.8945. All of the variables that were chosen as best predictors are listed below.

- The value of the daily measured high temperature
- The value of the daily measured high temperature squared
- Sunday
- Saturday
- Monday
- The daily measured average temperature
- The daily measured low humidity
- The daily measured high wind speed
- The sum of the daily measured precipitation
- The value of the daily measured average temperature squared
- The daily measured high dew point squared

**Insight**: In my opinion, the selected variables listed above stepwise approach thinks they have a strong association with the outcome variable**.**

**Task 10:**

The question asks if I can be certain that this modelling approach is not overfitting and to describe two approaches that could be used to prevent overfitting. In fact, I can't because I can't tell if the model is overfitting based on the coefficient of determination. It is possible to have a high R-squared value and still be overfitting. To avoid this problem, it is critical to determine whether a model is overfitting, which can be accomplished by evaluating its performance with techniques such as cross-validation or regularization.

1. **Regularization**

Regularization is a technique used to reduce overfitting and increase the generalization of a model by adding a penalty term to the loss function. Regularization does require additional bias and a search for optimal penalty hyperparameter is needed to find the best model [1]. Regularization has three main types such as LASSO (L1 regularization) regression, Ridge (L2 regularization) regression and Elastic net (which combines both L1 and L2) [1].

L1 regularization adds a penalty equal to the absolute value of the magnitude of the coefficients, which tends to shrink the coefficients towards zero and eliminates some features altogether. This can lead to sparse models with fewer features [1] [2].

L2 regularization adds a penalty term to the loss function equal to the square of the coefficient magnitude. This tends to shrink all coefficients by the same factor towards zero, but it does not necessarily eliminate any coefficients. L2 regularization helps to prevent overfitting by including a penalty term that discourages the model from placing too much emphasis on any single feature, and it can also help to improve model stability by reducing coefficient variance [2] [3].

Elastic net combines L1 and L2 with the addition of an alpha parameter that determines their ratio. The alpha parameter regulates the balance of L1 and L2 regularization. This enables Elastic Net to overcome some of the limitations of Lasso (L1 regularization) and Ridge (L2 regularization) alone and, in some cases, provide better predictive performance [4].

## 2. Cross validation

Cross-validation is a more sophisticated method for dividing data into training and testing sets. By evaluating a model on previously unseen data, cross-validation can help identify whether it is overfitting the training data or generalizing well to new data, and it can help identify the best set of hyperparameters even when we don't have enough data [1].

# References


[1] S. S. Skiena, The Data Science Design Manual, New York: Springer, 2017.

[2] A. Nagpal, "L1 and L2 Regularization Methods," Towards Data Science, 13 October 2013. [Online]. Available: https://towardsdatascience.com/l1-and-l2-regularization-methods-ce25e7fc831c. [Accessed 20 January 2023].

[3] A. Nagpal, "L1 and L2 Regularization Methods, Explained," Buit in, 05 January 2022. [Online]. Available: https://builtin.com/data-science/l2-regularization. [Accessed 20 January 2023].

[4] B. Giba, "Elastic Net Regression Explained, Step by Step," Machine learning compass, 26 June 2021. [Online]. Available: https://machinelearningcompass.com/machine_learning_models/elastic_net_regression/. [Accessed 20 January 2023].

[5] G. James, D. Witten, T. Hastie and •. R. Tibshirani, An Introduction to Statistical Learning, London: Springer, 2021.