CCT College Dublin

Assessment Cover Page

Module Title:	Machine Learning (10 ETCS)
Assessment Title:	Machine Learning Project
Lecturer Name:	Dr Muhammad Iqba
Student Full Name:	Arthur Claudino Gomes de Assis Heitor Gomes de Araujo Filho Natalia de Oliveira Rodrigues
Student Number:	2023146 2023098 2023112
Assessment Due Date:	30/04/2023
Date of Submission:	30/04/2023

Declaration

By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

MACHINE LEARNING

Higher Diploma in Science in Data Analytics for Business

Project Title

Develop an energy generation predictive model by using several machine learning methods, hyperparameters, and cross-validation.

Lecturer Name: Dr. Muhammad Iqba

Student Full Name & Number:

Arthur Claudino Gomes de Assis 2023146

Heitor Gomes de Araújo Filho 2023098

Natalia de Oliveira Rodrigues 2023112

Contents

1. Introduction

The idea that energy is the driver of growth in the twenty-first century is confirmed by the global surge in energy consumption. Population growth is another factor supporting the high demand for energy and services associated with it, in addition to urbanization. Most of the energy used to support economic growth that threatens environmental sustainability through CO2 emissions comes from fossil fuels.

A worldwide energy crisis has emerged with environmental concerns in the present conflicts, endangering access to electricity, particularly for people with lower incomes. World Energy Outlook (2022) states that "...the combination of the COVID pandemic and the current energy crisis means that 70 million people who recently gained access to electricity will likely lose the ability to afford that access - and 100 million people may no longer be able to cook with clean fuels, returning to unhealthy and unsafe means of cooking."

Words: 145

2. Business Understanding

2.1 Business Objective

Develop an energy generation predictive model by using several machine learning methods, hyperparameters, and cross-validation to justify the authenticity of the ML results.

2.2 Assess Situation

Population increase has led to a situation where developing sustainable energy sources is both a big financial opportunity and a noble contribution to safer and cleaner electricity infrastructure. Photovoltaic Energy (PV) is one of the most significant and widely used clean energy technologies now. PV is created by PV boards that harness solar energy to produce power. Despite being easily accessible, one of its biggest problems is that it might be unstable depending on the weather.

2.3 Data Mining

After analysing the dataset, this report will split the Data, Fit Data Preparation on Training Dataset, apply Data Preparation to Train and Test Datasets explore models and apply cross-validation to create forecasts capable to estimate power generation expected based on weather conditions.

2.4 Project Plan

The goal is to create and implement ML models in the eld of energy and the environment. The fundamental concept is to use cross-validation and hyperparameters to apply multiple ML. The next step is to select the ideal ML approach values. Based on the preferred explanation, a logical defence is also offered.

2.5 Project Question

Based on the information provided, the problem question of our project is: which regression approach will be used to predict "Generation_Power_kw" as a target variable?

Words: 200

3. Data Understanding

3.1 Data Exploration

The dataset spg.csv can be found on Kaggle, an authentic resource repository. Link: *https://www.kaggle.com/datasets/stucom/solar-energy-power-generation-dataset*.

The dataset has 4213 observations and 21 attributes. The Generation_Power_kw attribute, renamed afterwards as Power, is the target variable. The attributes data type are Numerical data: float64(17), and int64(4). The data dictionary can be found in Appendix I.

	temp	humid	sea_level	precip	snowf	t_cloud	h_cloud	m_cloud	l_cloud	radiat	...	w_direc10	w_speed80	w_direc80	w_speed900	w_direc900	w_gus
0	2.17	31	1035.0	0.0	0.0	0.0	0	0	0	0.00	...	312.71	9.36	22.62	6.62	337.62	24.4
1	2.31	27	1035.1	0.0	0.0	0.0	0	0	0	1.78	...	294.78	5.99	32.74	4.61	321.34	21.9
2	3.65	33	1035.4	0.0	0.0	0.0	0	0	0	108.58	...	270.00	3.89	56.31	3.76	286.70	14.0
3	5.82	30	1035.4	0.0	0.0	0.0	0	0	0	258.10	...	323.13	3.55	23.96	3.08	339.44	19.8
4	7.73	27	1034.4	0.0	0.0	0.0	0	0	0	375.58	...	10.01	6.76	25.20	6.62	22.38	16.5

5 rows × 21 columns

Figure 1 - df.head()

Clear visualization of the renamed columns can be found on Appendix II.

In the table below, we can see the statistical measurements of the dataset. It helps us to check if the spg.csv dataset needs to be normalized and also helps us to identify the presence of outliers.

	temp	humid	sea_level	precip	snowf	t_cloud	h_cloud	m_cloud	l_cloud	radiat	...	w_direc10
count	4213.000000	4213.000000	4213.000000	4213.000000	4213.000000	4213.000000	4213.000000	4213.000000	4213.000000	4213.000000	...	4213.000000
mean	15.068111	51.361025	1019.337812	0.031759	0.002808	34.056990	14.458818	20.023499	21.373368	387.759036	...	195.078452
std	8.853677	23.525864	7.022867	0.170212	0.038015	42.843638	30.711707	36.387948	38.013885	278.459293	...	106.626782
min	-5.350000	7.000000	997.500000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.540000
25%	8.390000	32.000000	1014.500000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	142.400000	...	153.190000
50%	14.750000	48.000000	1018.100000	0.000000	0.000000	8.700000	0.000000	0.000000	0.000000	381.810000	...	191.770000
75%	21.290000	70.000000	1023.600000	0.000000	0.000000	100.000000	9.000000	10.000000	10.000000	599.860000	...	292.070000
max	34.900000	100.000000	1046.800000	3.200000	1.680000	100.000000	100.000000	100.000000	100.000000	952.300000	...	360.000000

8 rows × 21 columns

Figure 2 - df.describe()

Words: 97

3.2 Data Visualization

3.2.1 Correlation among variables

The attributes have a strong correlation, as we can see below, 9 attributes have a positive correlation > 0.9, and 3 attributes have a negative correlation > 0.7. The relevant correlations can be visualized below.

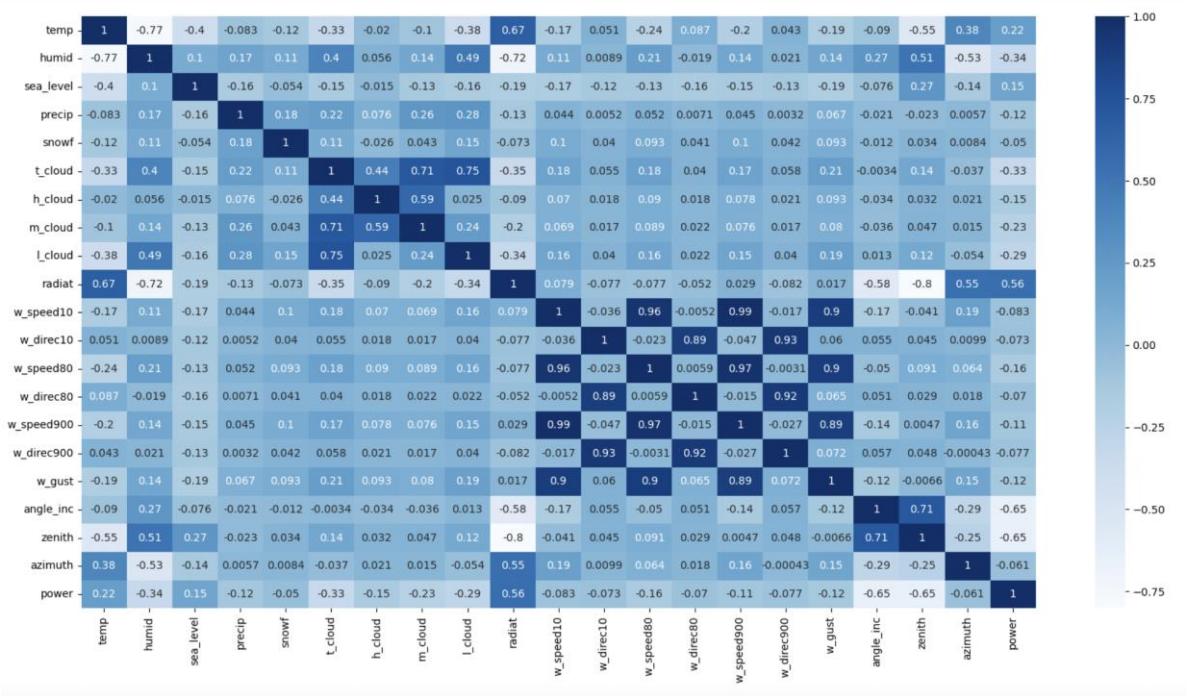


Figure 3 - Correlation Matrix. sns.heatmap(corr_table, cmap="blue", annot=True);

The most relevant correlations among variables and target variable:

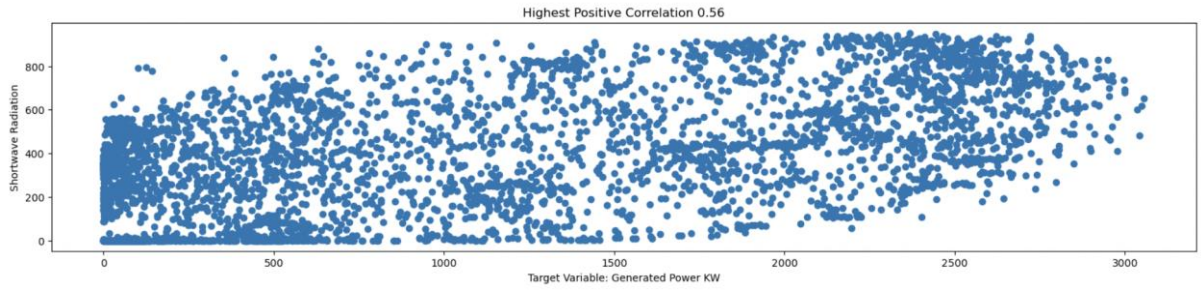


Figure 4 - Correlation between Shortwave Radiation and Generated Power

Words: 52

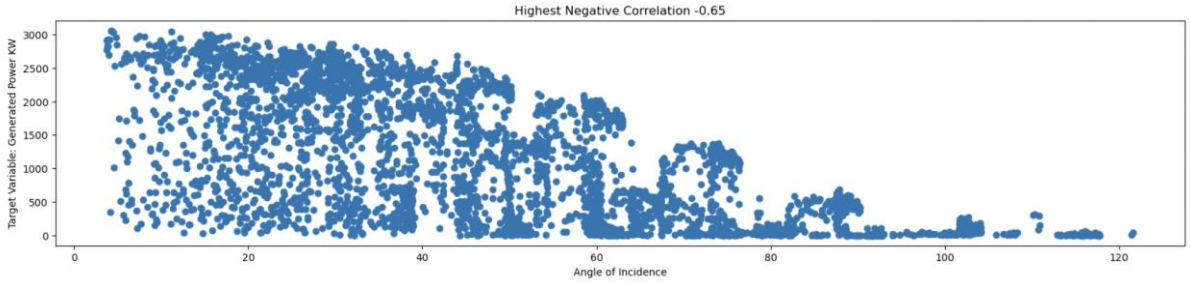


Figure 5 - Correlation between Angle of Incidence and Generated Power

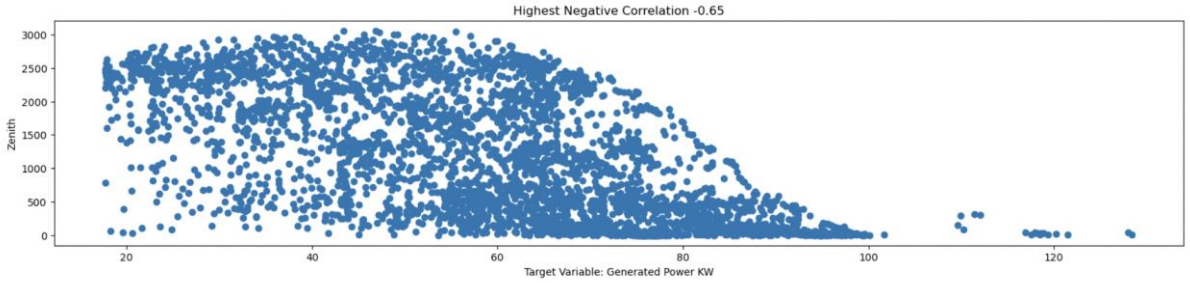


Figure 6 - Correlation between Angle of Incidence and Generated Power

3.2.2 Distribution and Outliers

The skewness will help us to understand how much the distribution of our data deviates from a normal distribution. As we can observe in Figure 7, 7 attributes have skew values > 1.0.

The strong presence of outliers can be noted in the attributes with skewness > 0.5%. Except for t_cloud and angle_inc attributes. Boxplots of the six features with more outliers are presented in Figure 8.

The visualization below is following the same order as the skewness graph above.

The correlation, distribution and outliers visualizations of other attributes not mentioned above in this chapter (Data Visualization), can be found in the appendix III.

Words: 104

```
df_skew = df.skew(axis=0)
df_skew.sort_values(ascending=False)
```

```
snowf          26.278299
precip          8.630336
h_cloud         2.143289
m_cloud         1.537975
l_cloud         1.449282
w_gust          1.111227
w_speed10       1.017751
w_speed900      0.990214
w_speed80       0.931519
t_cloud         0.739499
sea_level       0.517280
angle_inc       0.480446
power           0.324025
humid           0.296871
radiat          0.172694
temp            0.145986
azimuth         0.081150
zenith         -0.111279
w_direc80      -0.240414
w_direc900     -0.265769
w_direc10      -0.300773
dtype: float64
```

Figure 7 - Skewness per variable

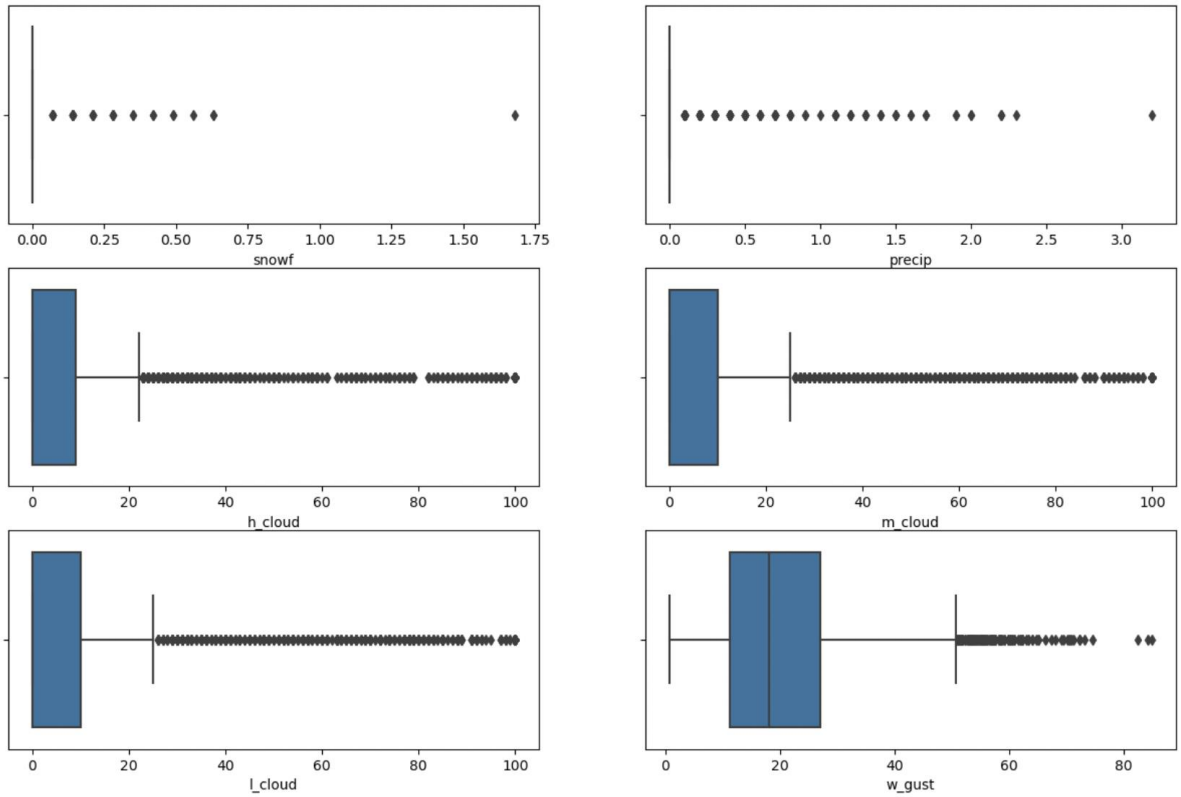


Figure 8 - Outliers in the 6 features with more presence of outliers.

4. Data Preparation

Null values were not found.

```
df.isnull().values.any()
```

```
False
```

Figure 9 - Coding to analyse the presence of null values.

Duplicated values were not found.

```
df_duplicate = df[df.duplicated()]
print("Duplicate Rows :", df_duplicate.sum())
```

```
Duplicate Rows : temp          0.0
humid         0.0
sea_level     0.0
precip        0.0
snowf         0.0
t_cloud       0.0
h_cloud       0.0
m_cloud       0.0
l_cloud       0.0
radiat        0.0
w_speed10     0.0
w_direc10     0.0
w_speed80     0.0
w_direc80     0.0
w_speed900    0.0
w_direc900    0.0
w_gust        0.0
angle_inc     0.0
zenith        0.0
azimuth       0.0
power         0.0
dtype: float64
```

Figure 10 – Coding to analyse the presence of duplicate values.

Words: 10

5. Modelling & Evaluation

5.1 Pre-processing the data

As was previously mentioned, the variables that are most closely related to our target variable, "power," do not exhibit high skewness or a significant number of outliers. As a result, pre-processing techniques that could bring the data closer to a normal distribution or minimize the excessive effects of outliers are not anticipated to significantly enhance models chosen for machine learning. Additionally, KNeighbors Regressor and Randon Forest Regressor were the strategies selected to tackle the regression problem, as will be shown below.

As we can verify by the graphs plotted of the most correlated variables to our target variable, it exists a certain characteristic of linearity between these variables. However, a model that views this association as strictly linear would be underfitting the results since it would be too straightforward to capture the intricacy of the regression studied. To further the presentation of this project, the score for a linear model will be provided in Appendix IV.

5.1.1 Train-Test Split and Random Forest Regressor

The dataset will be divided into 80% training and 20% testing for the project's initial methodology. To ensure reproducibility for this analysis, a random state of 38 will be chosen. Tests will then be run on a dataset that has been divided into proportions of 10% and 30% for training.

5.1.2 Train-Test Split and KNeighbors Regressor

The division of the dataset in this case is quite similar to the Random Forest model, except that in this case, the independent variables will be standardized using the RobustScaler, and subsequently the dataset will be split by the same proportions. The same random state will be set in this case.

After those procedures, we have the datasets needed to apply our Machine Learning methods.

Words: 184

5.2 Random Forest Regressor Model

```
In [27]: print("Training Coefficient of determination :", randomforestmodel.score(X_train, y_train))
         print("Testing Coefficient of determination :", randomforestmodel.score(X_test, y_test))

         Training Coefficient of determination : 0.9719971782461598
         Testing Coefficient of determination : 0.8213418119788982
```

It is possible to confirm that the coefficient of determination for the training set is higher than the testing set (overfitting). Nevertheless, a testing coefficient of determination of 0.8213 is still a good result, and they are not excessively apart from each other, which classifies the model created as useful to make predictions as proposed in the introduction of this project.

After performing the cross-validation analysis (kfolds=10), it was possible to observe that the real R2 to the dataset analysed utilizing the RandomForest is approximately 0.7939. Although it is a significantly lower result, it is more accurate to the real precision of the model. One way of improving the results found in this method is through the hyperparameters utilized by the method.

Tuning is the process of determining the ideal hyperparameters, and it is handled by the GridSearch function. Four hyperparameters provided by scikit-learn will be taken into consideration for the Random Forest Regressor in this project. (n.d.): max_features, n_estimators, min_samples_leaf, and min_samples_split

```
In [30]: # Create the parameter grid based on the results of random search
         param_gridrf = {
             'min_samples_leaf': range(1,50,100),
             'min_samples_split': range(2,50,100),
             'n_estimators': [100, 300,500],
             'max_features': [1, 10, 20]
         }
         # Instantiate the grid search model
         grid_search_rf = GridSearchCV(estimator = RandomForestRegressor(), param_grid = param_gridrf,
                                       cv = k_folds, n_jobs = -1,verbose = 1, scoring="r2")

In [31]: grid_search_rf.fit(X_train, y_train)

         Fitting 10 folds for each of 9 candidates, totalling 90 fits

Out[31]: GridSearchCV(cv=KFold(n_splits=10, random_state=None, shuffle=False),
                      estimator=RandomForestRegressor(), n_jobs=-1,
                      param_grid={'max_features': [1, 10, 20],
                                  'min_samples_leaf': range(1, 50, 100),
                                  'min_samples_split': range(2, 50, 100),
                                  'n_estimators': [100, 300, 500]},
                      scoring='r2', verbose=1)

In [32]: # printing the optimal Coefficient of Determination and hyperparameters
         print('We can get r2 of',grid_search_rf.best_score_,'using',grid_search_rf.best_params_)

         We can get r2 of 0.8003002703559939 using {'max_features': 10, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_e
         stimators': 500}
```

Figure 11 - GridSearch for tunning hyperparameters on Random Forest Model

As can be seen, the strategy produced an improvement of about 0.007. We believe that the result produced is sufficient for the objective purposed, considering the amount of data provided. Further improvement could possibly be reached tunning more parameters.

5.2.1 Fitting the Model with the best hyperparameters

It is feasible to confirm that the model performs with an R2 of approximately 0.799 inside its own training dataset and with R2 approximately of 0.8260 within the testing dataset. These values being close together suggests that the model is not overfitting or underfitting but it is rather functioning optimally.

Words: 202

5.3 KNeighbors Regressor Model

The same procedure will be applied to utilize the machine learning methods of KNeighbors, using at this time the training and testing dataset standardized.

```
In [39]: print("Training Coefficient of determination :", Kneighborsmodel.score(X_train_scaled, y_train_scaled))
         print("Testing Coefficient of determination :", Kneighborsmodel.score(X_test_scaled, y_test_scaled))

         Training Coefficient of determination : 0.7931459458212848
         Testing Coefficient of determination : 0.7270760124842842

In [40]: scores = cross_val_score(KNeighborsRegressor(), X_train_scaled, y_train_scaled, scoring = 'r2', cv = k_folds)
         print(scores, scores.mean())

         [0.61340899 0.70368328 0.6322615  0.69989516 0.6315504  0.70742988
          0.68823704 0.70153025 0.64648563 0.68756625] 0.6712048390098235
```

Figure 12 - Values of coefficient of determination to the training dataset, testing dataset and after cross-validation.

The decrease in the coefficient of determination following cross-validation can be confirmed. That occurs because several dataset subsets are tested, and the outcome is, therefore, less skewed.

In this project, the KNeighbors regressor will consider 3 hyperparameters, provided by scikit- learn. (n.d.): Weight, p: Power parameter and n_neighbors. It is feasible to confirm that the tuning of the hyperparameters produced an even greater impact in the case of the KNeighbors algorithm, raising R2 by about 0.05, which is a significant result.

```
In [41]: # Create the parameter grid based on the results of random search
param_gridkn = {
    'weights': ['uniform','distance'],
    'p': [1,2],
    'n_neighbors' : [5, 8, 11, 15]
}
# Instantiate the grid search model
grid_search_kn = GridSearchCV(estimator = KNeighborsRegressor(), param_grid = param_gridkn,
                              cv = k_folds, n_jobs = -1, verbose = 1, scoring="r2",)

In [42]: grid_search_kn.fit(X_train_scaled, y_train_scaled)

Fitting 10 folds for each of 16 candidates, totalling 160 fits

Out[42]: GridSearchCV(cv=KFold(n_splits=10, random_state=None, shuffle=False),
             estimator=KNeighborsRegressor(), n_jobs=-1,
             param_grid={'n_neighbors': [5, 8, 11, 15], 'p': [1, 2],
                         'weights': ['uniform', 'distance']},
             scoring='r2', verbose=1)

In [43]: # printing the optimal Coefficient of Determination and hyperparameters
print('We can get r2 of',grid_search_kn.best_score_,'using',grid_search_kn.best_params_)

We can get r2 of 0.7237545699582916 using {'n_neighbors': 5, 'p': 1, 'weights': 'distance'}
```

Figure 13 - GridSearch for tunning hyperparameters on K Neighbors

5.3.1 Fitting the Model with the best hyperparameters

The model performs even somewhat better when predictions are made using the ideal model, reaching an R2 of 0.7607.

5.4 Plotting graphs for analysis

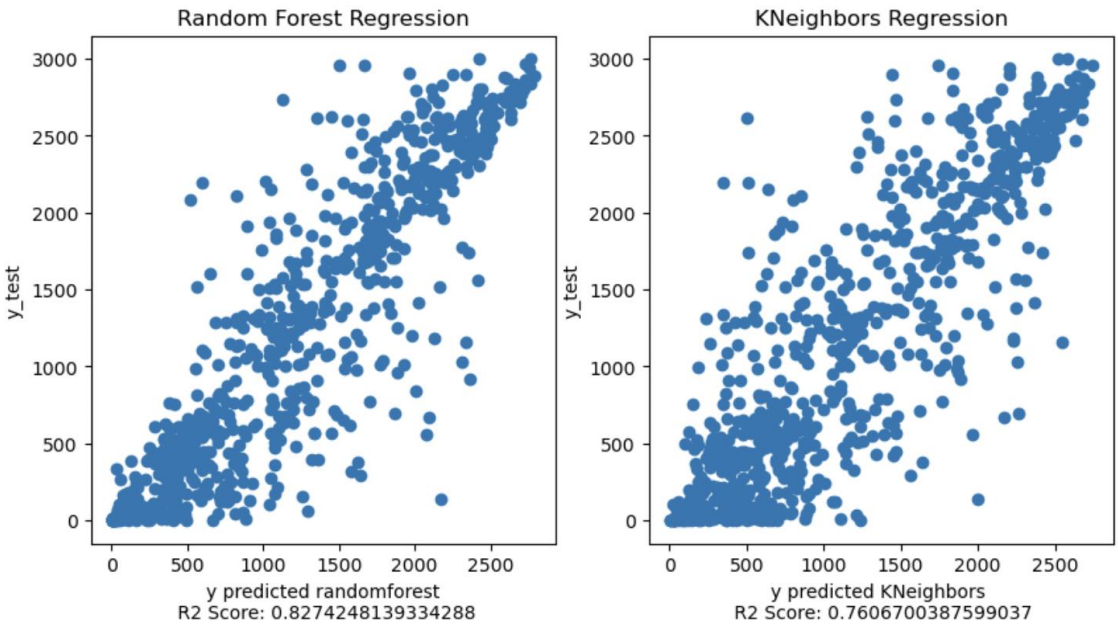


Figure 14 - Comparison between values predicted by Random Forest Regression and K Neighbors Regression.

The fit is better the more the points are concentrated on a diagonal from 0 to the point where x=3,000 and y=3,000. The model that comes closest to this ideal representation, as can be seen, is the random forest regression.

Words: 140

5.5 Train-Test Split (10% and 30%) and Modelling

All the procedures using Train-Test split 10% and 30%, including cross-validation and hyperparameters, were performed and it can be found in the Jupyter Notebook file attached.

The following graph shows the comparison of all results performed.

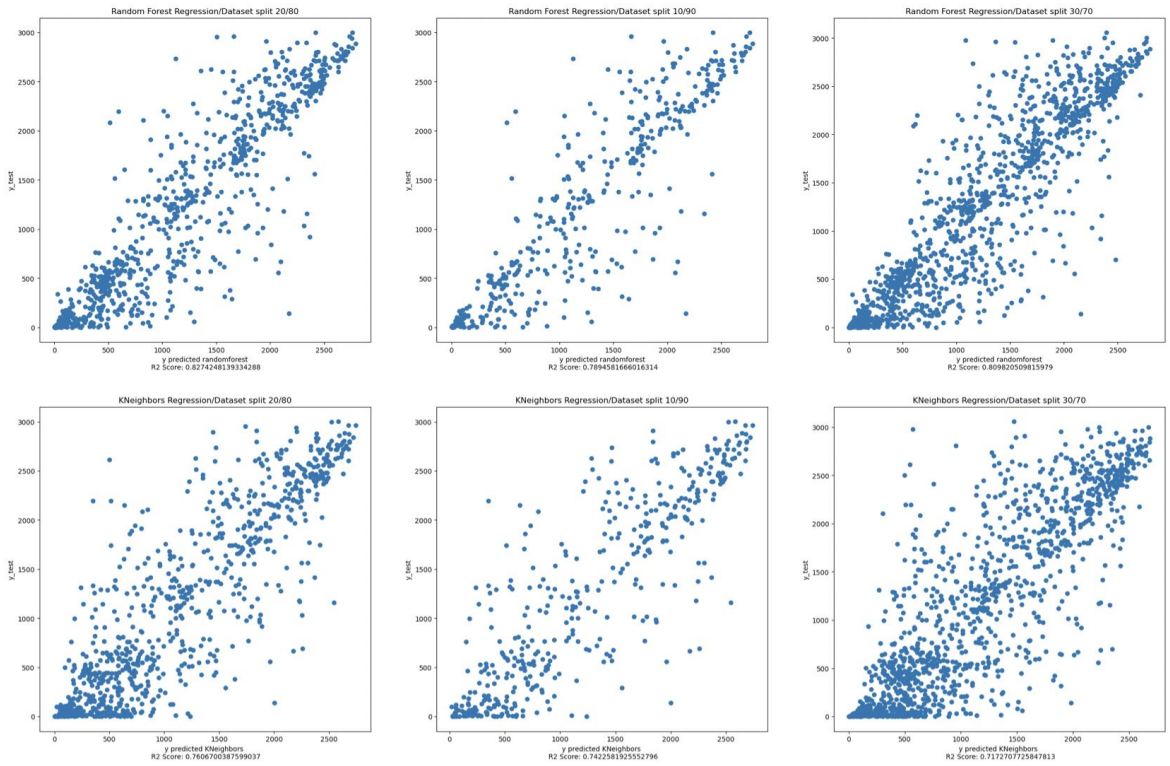


Figure 15 - Comparison between final results found on all performed tests.

Words: 36

6. Conclusion

Skewness values were obtained from this analysis, which demonstrated that the distributional features of all the collected data varied. As the construction of the prediction model continued, the obtained data were examined for the presence of missing data and it was determined that no additional data cleaning was required.

After PCA analysis, it was concluded that there was no need to apply it and the Machine Learning Algorithms Linear Regression, Decision Tree, Random Forest, and KNN models were analyzed to predict the dependent variable. After the results, only Random Forest and KNN were chosen in this project to be processed and analyzed as predictor algorithms.

After applying the two algorithms we obtained an R-squared value of 0.81 for RF and 0.72 for KNN. After that, we applied the hyperparameters to control the learning process and the values presented were 0.82 for RF and 0.76 for KNN.

It is possible to conclude that Random Forest is the best model to predict the studied Data.

Words: 163

7. Reference List

Arya, N. (2022). *Does the Random Forest Algorithm Need Normalization?* [online] KDnuggets. Available at: https://www.kdnuggets.com/2022/07/random-forest-algorithm-need-normalization.html [Accessed 22 Apr. 2023].

Goyal, C. (2021). [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2021/ [Accessed 27 Apr. 2023].

Meshva (2022). *Solar energy power generation dataset*. [online] www.kaggle.com. Available at: https://www.kaggle.com/datasets/stucom/solar-energy-power-generation-dataset.

Nyuytiymbiy, K. (2022). *Medium*. [online] Medium. Available at: https://towardsdatascience.com/parameters-%20and-hyperparameters-%20aa609601a9ac#:~:text=Hyperparameters%20are%20parameters%20whose%20values [Accessed 25 Apr. 2023].

scikit-learn (2018). *3.2.4.3.2. sklearn.ensemble.RandomForestRegressor — scikit-learn 0.20.3 documentation*. [online] Scikit-learn.org. Available at: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html [Accessed 23 Apr. 2023].

Scikit-learn.org. (2019). *sklearn.neighbors.KNeighborsRegressor — scikit-learn 0.22 documentation*. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html [Accessed 21 Apr. 2023].

8. Appendix

Appendix I: Dictionary

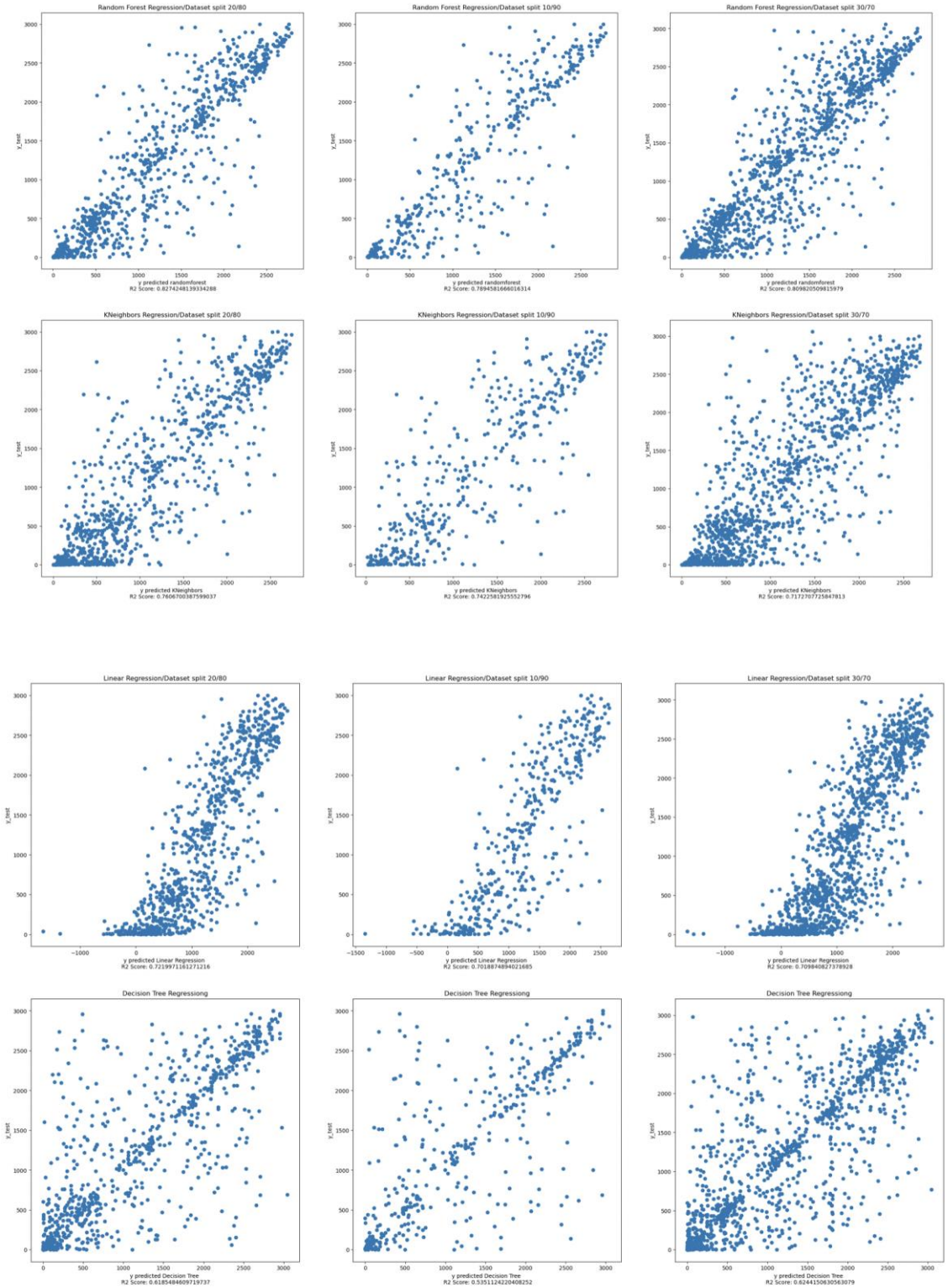
- temperature_2_m_above_gnd - Temperature of the air 2 meters above the level of the ground.
- relative_humidity_2_m_above_gnd - Relative humidity 2 meters above the level of the ground.
- mean_sea_level_pressure_MSL - Mean Sea level pressure
- total_precipitation_sfc - Total precipitation of surface
- snowfall_amount_sfc - Total precipitation of snow of surface.
- total_cloud_cover_sfc - Fraction of the sky covered by all visible clouds.
- High_cloud_cover_high_cld_lay - Covering of clouds over 20,000 ft in altitude.
- medium_cloud_cover_mid_cld_lay - Covering of clouds over 6,000 ft and under 20,000 ft in altitude.
- low_cloud_cover_mid_cld_lay - Corering of clouds under 6.000 ft in altitude.
- shortwave_radiation_backwards_sfc - Radiant energy produced by the sun
- wind_speed_10_m_above_gnd - wind speed 10 meters above the ground
- wind_direction_10_m_above_gnd - Direction of wind 10 meters above the ground (measured in degrees from true north).
- wind_speed_80_m_above_gnd - wind speed 80 meters above the ground
- wind_direction_80_m_above_gnd - direction of wind 10 meters above the ground (measured in degrees from true north).
- wind_speed_900_mb - wind speed 900 meters above the ground.
- wind_direction_900_mb - direction of wind 900 meters above the ground
- wind_gust_10_m_above_gnd - the turbulence of wind. Define the highest wind speed during the past hour, 10 meters above the ground.
- angle_of_incidence - the angle between the sun's ray and a normal vector to the surface of a solar board.
- Zenith - Angle between the sun's ray and a normal vector to the Earth.
- Azimuth - Angle between the sun's ray on the surface of the Earth, and its true North, measured in clockwise way.
- generated_power_kw - Power generated by solar plant.

Appendix II: Renamed Columns Glossary

temp = temperature_2_m_above_gnd
humid = relative_humidity_2_m_above_gnd
sea_level = mean_sea_level_pressure_MSL
precip = total_precipitation_sfc
snowf = snowfall_amount_sfc
t_cloud = total_cloud_cover_sfc
h_cloud = high_cloud_cover_high_cld_lay
m_cloud = medium_cloud_cover_mid_cld_lay
l_cloud = low_cloud_cover_low_cld_lay
radiat = shortwave_radiation_backwards_sfc
w_speed10 = wind_speed_10_m_above_gnd
w_direc10 = wind_direction_10_m_above_gnd
w_speed80 = wind_speed_80_m_above_gnd
w_direc80 = wind_direction_80_m_above_gnd]
w_speed900 = wind_speed_900_mb
w_direc900 = wind_direction_900_mb
w_gust = wind_gust_10_m_above_gnd
angle_inc = angle_of_incidence
zenith = zenith
azimuth = azimuth
power = generated_power_kw

Appendix IIII: Machine Learning Tests Appendix V: Machine Learning Graphs

All the other tests performed can be found on the Jupyter Notebook. The graph below shows a resume of the results reached after modelling.



9. Reflective Journal

9.1 Team Member: Heitor Gomes de Araújo Filho 2023098

I, Heitor, made a number of contributions to this project throughout its duration. Firstly, All the members of the group divided themselves to look for a Dataset regarding energy consumption and the environment. During this phase, 4 different datasets were found, but they didn't fit the criteria established by the professor. Finally, Nathalia found the Dataset `Solar Power Generation` and we all decided that this Dataset would be used for the assignment.

After analyzing the variables in question and understanding the purpose of the Dataset, I worked together with Nathalia and Arthur to create a comprehensive plan that will be used to predict our dependent variable, which in this case is `energy generation`. To make sure our plan worked, we generated concepts, tweaked them, and tested them with focus groups.

Thirdly, I was responsible for creating/validating the introduction and the Business Understanding of the assignment so that the next steps of the work would be easier to understand. Meanwhile, Nathalia started the EDA part and Arthur started the Machine Learning part.

Next, I discussed with Nathalia which part of the EDA would be important to put into our work to try to answer our question. We discussed and decided to address cleaning, outliers, normality of samples, and correlation between variables. Soon after, I discussed with Arthur the cross-validation and hyperparameters of the two algorithms used to predict (Random Forest and KNN), so I agreed with what was done and helped in the comments of each result he approached.

In the next phase, all the members of the group were responsible for validating all the coding used in the assignment, and soon after, we started to format each item using Colab. When all the work was validated, I was responsible for writing the conclusion and, in a concise way, I covered the main findings in the conclusion and answered our main question.

Finally, each member included their references used (Havard) and the individual participation of each member was discussed.

I am pleased with my overall contributions to this project. Working closely together, my team members and I were able to create a machine-learning project that met our objectives and went above and beyond what we had anticipated.

Words: 366

9.2 Team Member: Arthur Claudino Gomes de Assis 2023146

The whole project was split into four main phases, Business Understanding, Data Preparation, Modelling and Evaluation, and Deployment. By the beginning, all groups started an interaction seeking to decide on a dataset that was interesting to all. We found out that each one came from different background and had different personal interests regarding what to study. We concluded that we could work over any kind of dataset as soon as that was something of relevance and importance.

I have a particular interest in the sector of energy, especially speaking of renewable energies, because I aim to make a Master's Degree later in this subject. Therefore, I suggested trying to find something related to this, which included clean energy and the environment. Although I had this idea about working on this, I did not know what kind of data we could work on, and I did not find anything that interested me beforehand. All the group embraced the idea, and Natalia was the member who found the dataset that caught our attention, and that turned out to be this group’s project. We discussed the quality of the dataset, and its possibilities to attend the requirements of the project and decided to explore further the dataset and work on it.

For being the member most interested specifically in this subject in the group, I have been accountable to research the meaning of each feature in the dataset and create the dictionary. In addition, I sketched an introduction that would help everyone to further understand the context of the dataset within business, and that would try to spark interesting insights about the dataset in its context. This was later used to create the ultimate introduction of the project.

All the groups discussed actively the main features of the dataset, created in the exploratory data analysis, and reached conclusions about how to explore the dataset using Machine Learning models. After discussing, I have been responsible for sketching the first coding of machine learning models and explaining thoroughly what

has been done. Afterwards, all the group contributed actively to the discussions of the ML, validated the model that has been created, and decided which model would be kept on the report. I made further analysis of the ML models that have been created and left all the information necessary for a full understanding of the models.
This information was later resumed by Heitor and all the conclusions taken from the ML models were deployed by him.
Overall, I played an important whole in the understanding of the dataset context and how to explore it, and in the construction and understanding of the Machine Learning models explored in the assignment. Nevertheless, all members of the group participated actively and validated all the steps of the project. I feel glad and appreciated for all the team contribution, and I consider that this is an authentic project in which every member contributed significantly in every step.

Words: 486

9.3 Team Member: Natalia de Oliveira Rodrigues 2023112

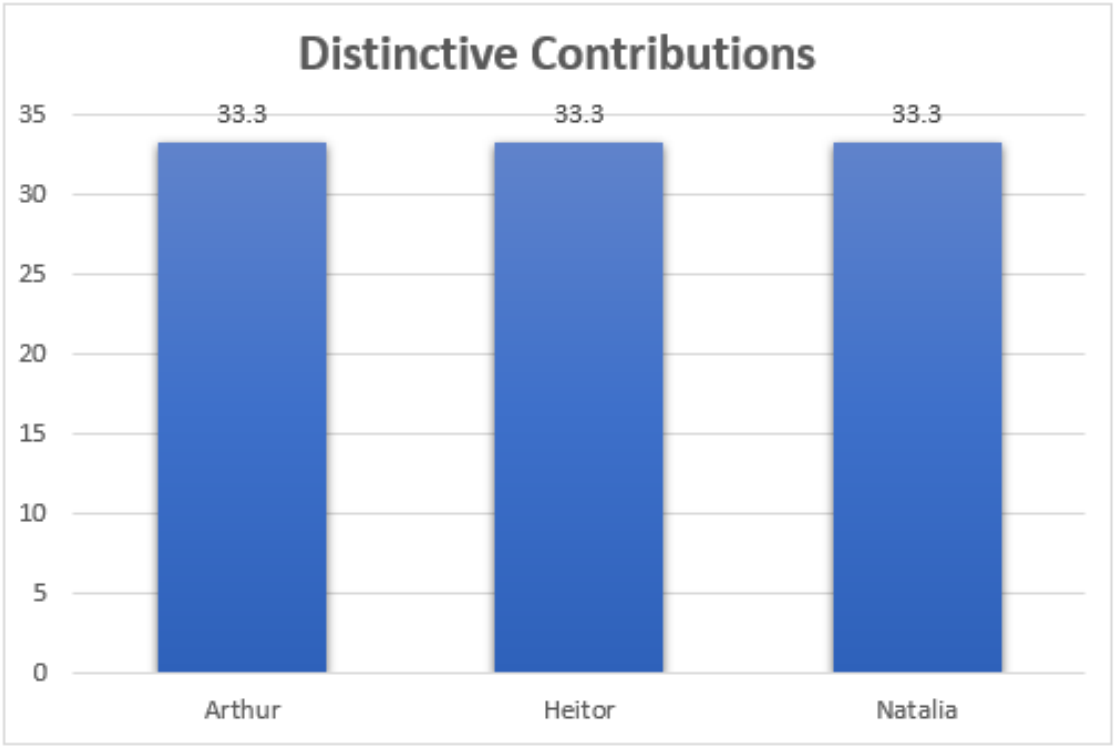
Participating in this group assignment has been an extraordinary experience. It is well-known that group assignments have challenges, such as different points of view, conflicting schedules, and so on. However, learning how to deal with these challenges is part of our professional development. Without a doubt, our group have done that.
Our group were able to recognize our strengths and weakness and move forward with only one goal in mind: To deliver the best report possible, managing our limited knowledge and lack of experience in the application of machine learning models.
I participated in founding the dataset used in the energy and environment area. It was suggested by Arthur, who is a renewable energy passionate. After the group approval, I progressed to the first steps to understand the data, creating EDA visualization such as correlations among variables, distributions, outliers, and variables' skewness, and checking the necessity of data cleaning. I also helped the group track the project's progress using CRISP-DM and made sure the CA requirements were been followed as requested.
Have no words to express my gratitude for my colleagues and their contributions to this project:

-Arthur, who is a valuable member of this group assignment, making an incredible job applying loads of different scenarios and approaches to guarantee that the best accurate result would be presented on the final report.
-Heitor, who has an extraordinary mind, is capable to consolidate our ideas, suggestions, and analysis to create the most amazing report comments and conclusions.
For sure, I grow as a person, as a student and as a professional after this assignment.
Thank you, Dr Muhammed, for giving us this opportunity.

Words: 472

9.4 Individual Contribution Graph

The group agreed that the individual contribution of each member is equal (33%). So, it is represented below:



The contributions also can be found on the CRIP-DM spreadsheet used to track down the project progress.

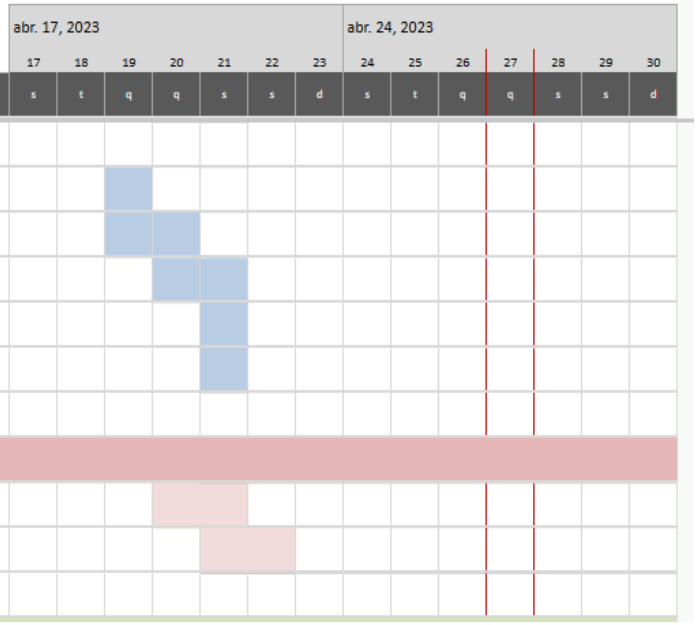
MACHINE LEARNING PROJECT TITLE: SOLAR ENERGY

Company Name
Project Lead

SIMPLE GANTT CHART by Vertex42.com
https://www.vertex42.com/ExcelTemplates/simple-gantt-chart.html

Project Start: qua., 4/19/2023
Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Phase 1 Business Understanding & Data Preparation				
To find a public dataset from a authentic resource repository (at least 3000 rows and 10 columns in the areas suggested).	Natalia	100%	4/19/23	4/19/23
To explore dataset and verify data quality.	Arthur | Heitor | Natalia	100%	4/19/23	4/20/23
To formulate the question that the group is considering for the ML project.	Natalia	100%	4/20/23	4/21/23
To Introduce to problem description and motivation that provide a compact and clear case for the proposed ML project.	Arthur	100%	4/21/23	4/21/23
To create a dictionary of variables explaining their meaning in context	Arthur	100%	4/21/23	4/21/23
Phase 2 Data Preparation				
To characterise and to clean of dataset.	Natalia	100%	4/20/23	4/21/23
To procude outstanding characterization, and highlight all details from source to fields in tables and graphs.	Natalia	100%	4/21/23	04/21/23



MACHINE LEARNING PROJECT TITLE: SOLAR ENERGY

Company Name
Project Lead

SIMPLE GANTT CHART by Vertex42.com
https://www.vertex42.com/ExcelTemplates/simple-gantt-chart.html

Project Start: qua., 4/19/2023
Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END	abr. 17, 2023							abr. 24, 2023						
					17	18	19	20	21	22	23	24	25	26	27	28	29	30
					s	t	q	q	s	s	d	s	t	q	q	s	s	d
Phase 3 Modeling & Evaluation																		
To train and test ML Models using 3 (three) logical splits.	Arthur	100%	4/21/23	4/22/23														
To use cross-validation to test the generalizability of the model & To apply Grid Search to find Optimal Hyperparameters	Arthur	100%	4/21/23	4/22/23														
To interpret and to explain the results obtained, discuss overfitting/underfitting/generalisation.	Arthur	100%	4/23/23	4/24/23														
To provide a rationale for the chose model and use visualization to support your findings.	Arthur	100%	4/23/23	4/24/23														
Phase 4 Deployment																		
To write the report conclusion based on problem specification and objectives.	Heitor		4/25/23	4/25/23														
To review final project: Full CA review by all group members start-to-end.	Arthur | Heitor | Natalia		4/26/23	4/27/23														
To deliver a power point presentation - 5 slides each member (for individual group member)	Individual Task		4/27/23	4/27/23														
To deliver a reflective Journal - Between 500 to 700 words (for individual group member)	Individual Task		4/27/23	4/27/23														
To submit group assigment on Moddle	Natalia		4/27/23	4/27/23														
Insert new rows ABOVE this one																		