Machine Learning Framework for *Spatial Daylight Autonomy*

- One of Many Routes of Understanding an assessment of any Data is by Performing Exploratory Data analysis , Predictions Through Machine Learning Algorithms & Sensitivity Analysis , here in this study we will apply these tasks in order to get a better understanding of our Provided Data Generaly and the *Spatial Daylight Autonomy* Data Specifically

Study GuideLine :

- Checking and Cleaning Data Format
- Exploratory Data Analysis (Getting to Know out Data Throught Visualizations and code)
- Preparing & Applying Machine Learning Algorithms
- Evaluating Model
- Sensitivity Analysis

Code Source : " Spatial Daylight Autonomy Analysis Through Machine Learning.html " (NoteBook Markdown)

By : Madjid Erroukrma , **Contact** : madjidmain@gmail.com , other Contact information at the end of this file

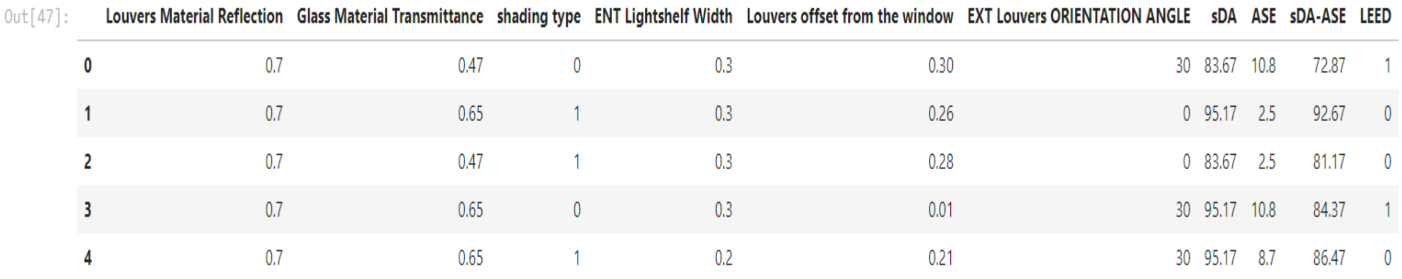
Domain Knowledge :

Data Consists of 10 columns : among them data that represents How Louvers Reflect Light , Values of Glass Material Transmittance of light , the shading type in the space , Characteristic of Louvers , Spatial Daylight Autonomy (sDA) , Annual Sunlight Exposure (ASE) & LEED Rating System

```
Louvers Material Reflection
Glass Material Transmittance
shading type
ENT Lightshelf Width
Louvers offset from the window
EXT Louvers ORIENTATION ANGLE
sDA
ASE
sDA-ASE
LEED
```

I. Checking and Cleaning Data Format : Using Python Through Jupyter Notebook

We import and View Data



Out[47]:

	Louvers Material Reflection	Glass Material Transmittance	shading type	ENT Lightshelf Width	Louvers offset from the window	EXT Louvers ORIENTATION ANGLE	sDA	ASE	sDA-ASE	LEED
0	0.7	0.47	0	0.3	0.30	30	83.67	10.8	72.87	1
1	0.7	0.65	1	0.3	0.26	0	95.17	2.5	92.67	0
2	0.7	0.47	1	0.3	0.28	0	83.67	2.5	81.17	0
3	0.7	0.65	0	0.3	0.01	30	95.17	10.8	84.37	1
4	0.7	0.65	1	0.2	0.21	30	95.17	8.7	86.47	0

Checking The Shape of Data which is 722 observation and 10 column

We need to make Sure that The data is in proper format through :

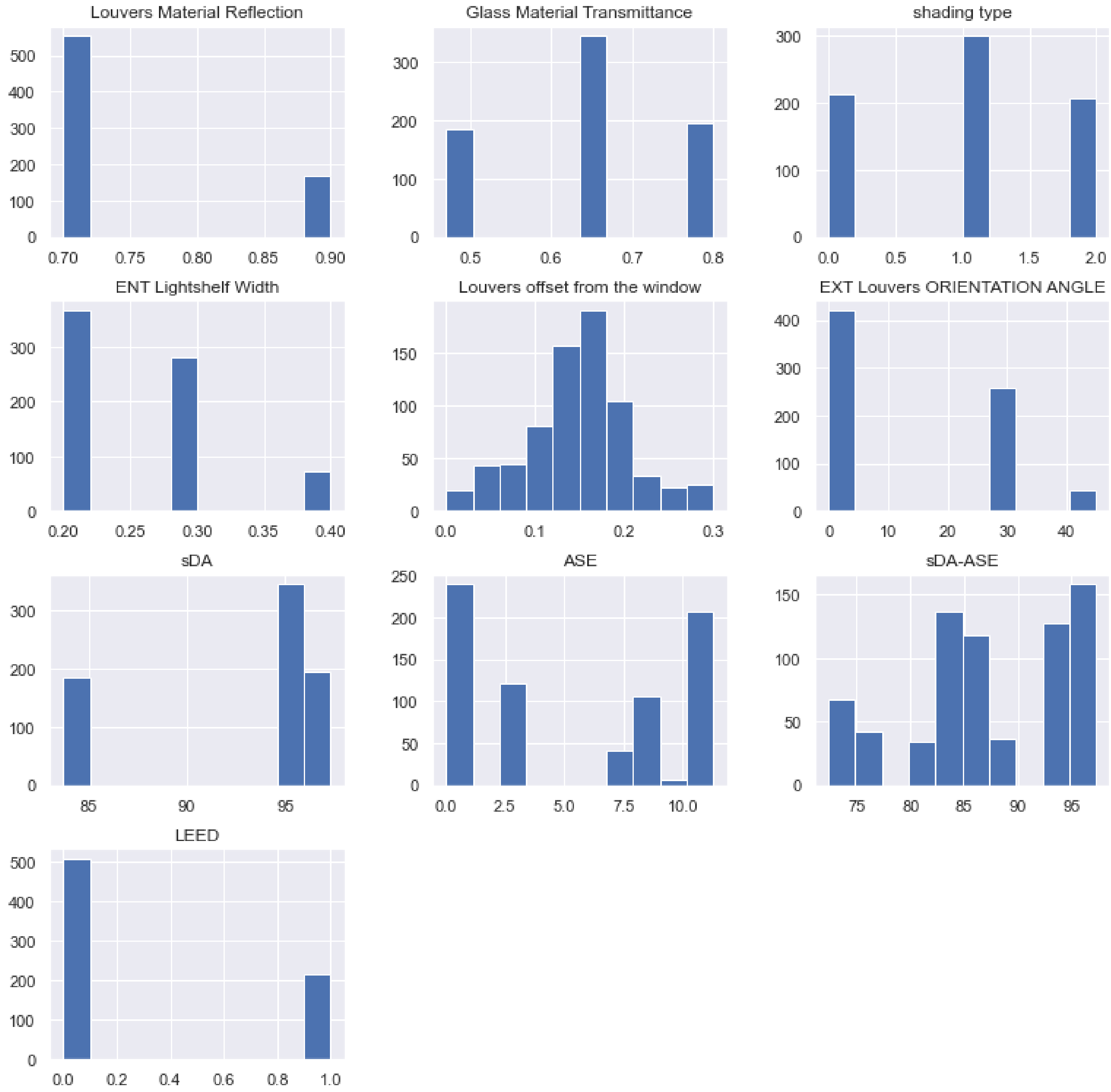
- Checking Missing Values
- Checking Column Correct Format
- Checking Outliers and unique values

We make sure that our data is clean and ready for Next Step Which is :

II. **Exploratory Data Analysis** :

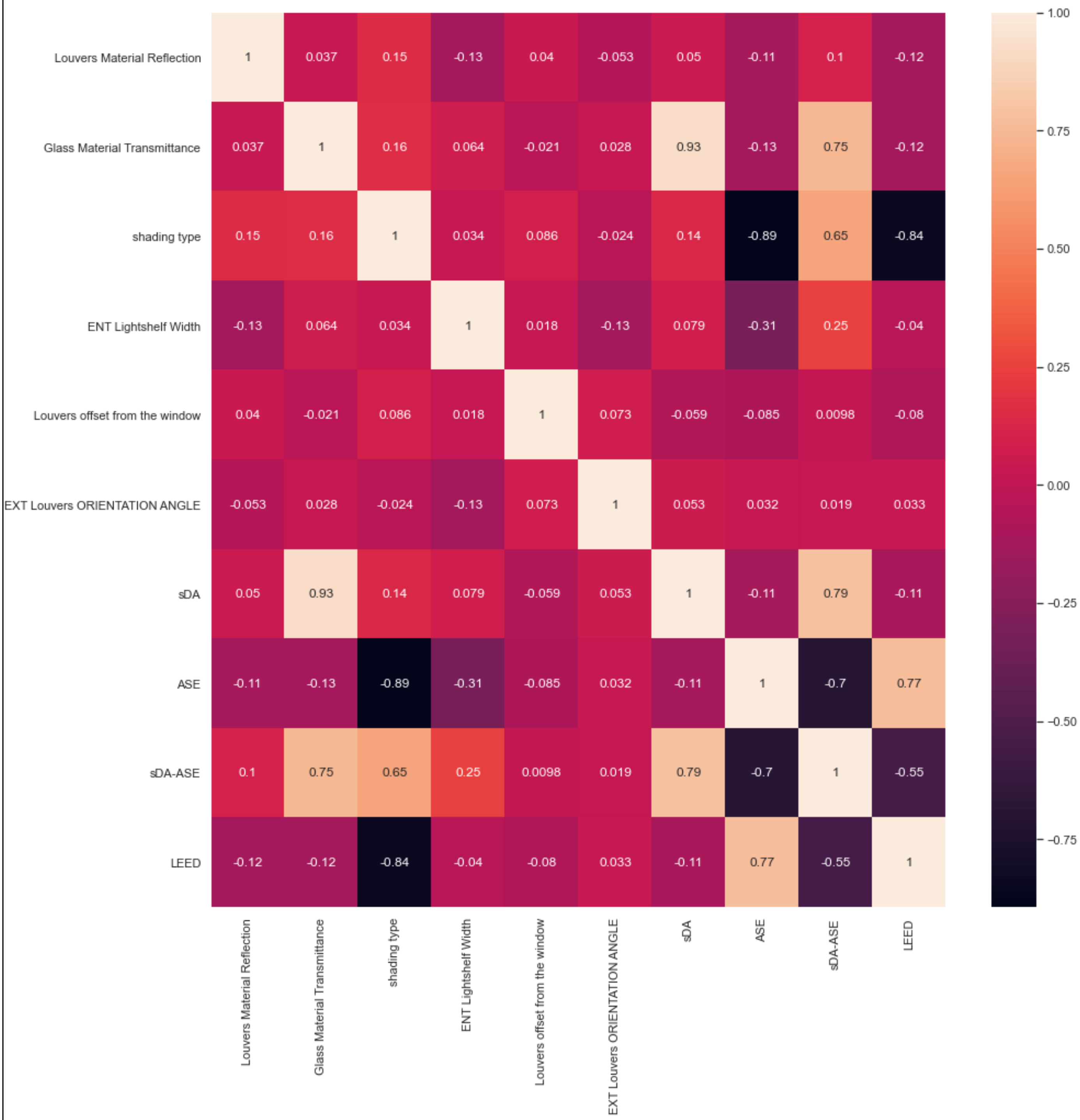
(Getting to Know out Data Throught Visualizations and code)

Next is to check the data Range Through Histograms , checking the data range will help us understand data scale and scale it if we need to for the machine learning algorithms :



Our data have different ranges which point the need to scale the data later on

Now lets check the data correlation to see variable correlation to the target Column (sDA) :



We can notice 2 strong negative & positive correlation :

- sDA-ASE : Positive Correlation
- Glass Material Transmittance : Positive Correlation

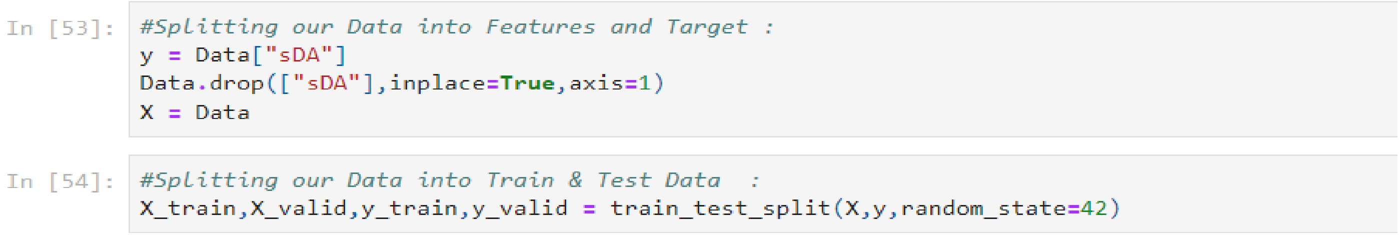
We can later on see if they have any impact on the Prediction

III. Preparing & Applying Machine Learning Algorithms :

Now into the ML Process , we start this phase by splitting data into Features and target column and since we want to see the impact of all columns in predicting Spacial Daylight (target)

After we split X ,y into Train and Validation Data, we don't want to use the same data for training and evaluation

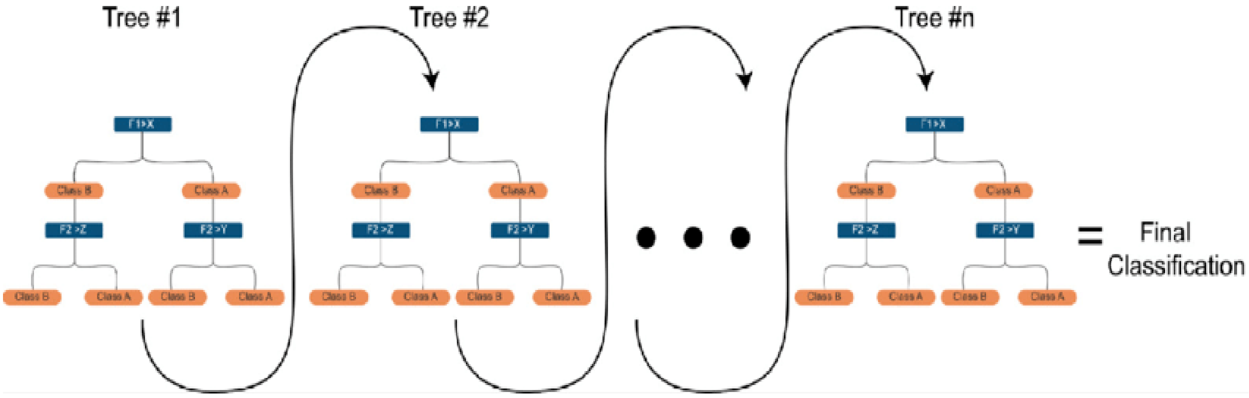
All these steps using Bunch of Pandas , SKlearn Metrics



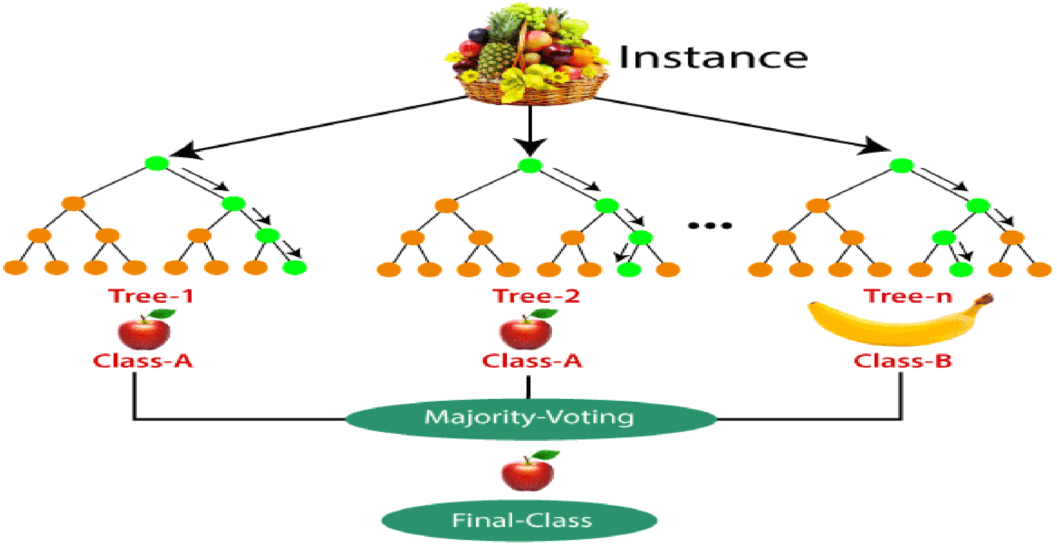
```
In [53]: #Splitting our Data into Features and Target :
         y = Data["sDA"]
         Data.drop(["sDA"],inplace=True,axis=1)
         X = Data

In [54]: #Splitting our Data into Train & Test Data  :
         X_train,X_valid,y_train,y_valid = train_test_split(X,y,random_state=42)
```

We are using 2 main algorithms for this problem :

- XGBOOST : the concept of this algorithm rolls around Multiple Decision Trees that build on top of each other to correct the errors of the Previous Decision Trees



- Random Forest Regressor : This one concept revolves around combining prediction from multiple trees and vote for the majority

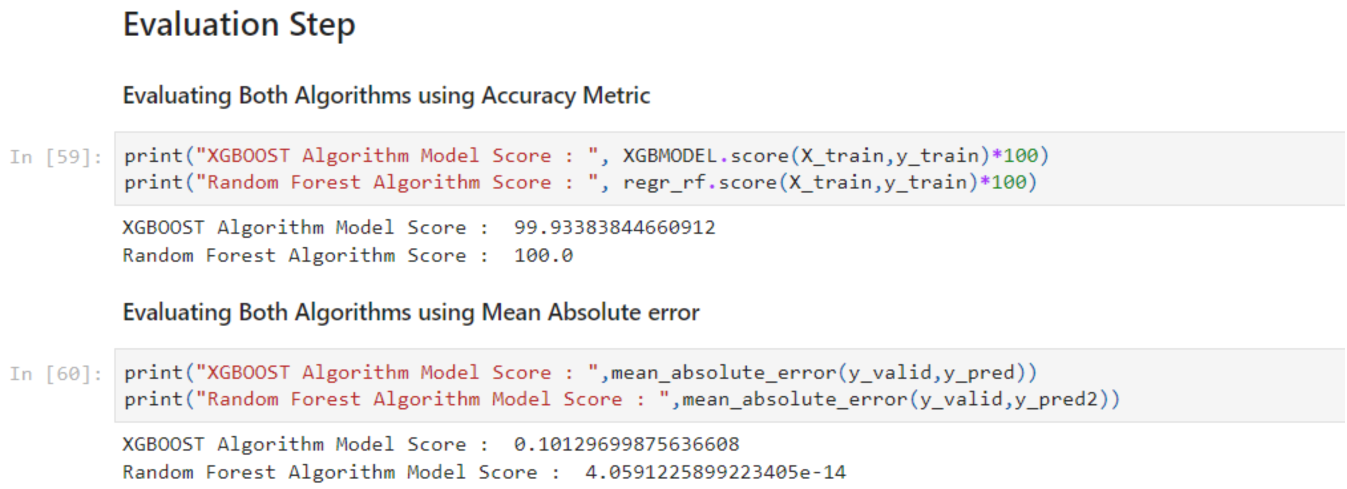


We tune Algorithm Parameters and Train Both algorithm and Predict while storing the prediction into a variable , we're using 2 algorithm so we can compare each algorithm performenace

Before we move to next phase we need to address a roadblock :

- The problem with this Data is that the data has little amounts of Observations (722) and by splitting the data it makes it even smaller and makes the score a bit unbelievable and we can't see the actual effect of columns on the prediction except for the top ones

IV. EVALUATING MODEL :



Evaluation Step

Evaluating Both Algorithms using Accuracy Metric

```
In [59]: print("XGBOOST Algorithm Model Score : ", XGBMODEL.score(X_train,y_train)*100)
         print("Random Forest Algorithm Score : ", regr_rf.score(X_train,y_train)*100)

XGBOOST Algorithm Model Score :  99.93383844660912
Random Forest Algorithm Score :  100.0
```

Evaluating Both Algorithms using Mean Absolute error

```
In [60]: print("XGBOOST Algorithm Model Score : ",mean_absolute_error(y_valid,y_pred))
         print("Random Forest Algorithm Model Score : ",mean_absolute_error(y_valid,y_pred2))

XGBOOST Algorithm Model Score :  0.10129699875636608
Random Forest Algorithm Model Score :  4.0591225899223405e-14
```

We use 2 metrics for evaluating the **Accuracy Metric** & **Mean Absolute error** , the accuracy metric range from 0 % to 100 % and as we can see XGBOOST achieved 99.93 % accuracy while Random Forest Regressor Algorithm scores 100 % , which is a little bit of odd and that returns to the fact that our data is small

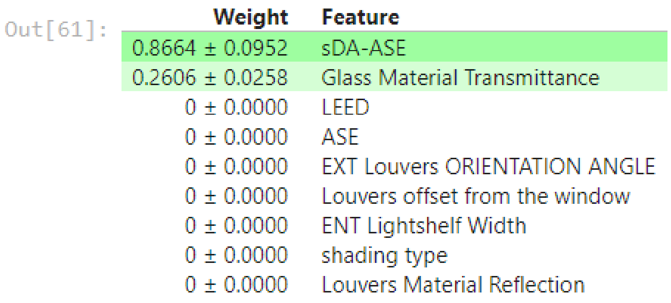
The mean absolute error the closest value to 0 the better is and as we can see XGBOOST has MAE of 0.10 while RGF 4.059x10^-14 which as I said before it all goes back to the lack of data

Next and Last Phase is to see how each variable affected the prediction and how using **Sensitivity Analysis**

V. Sensitivity Analysis :

First , we will check which variables effected the prediction in both algorithms :

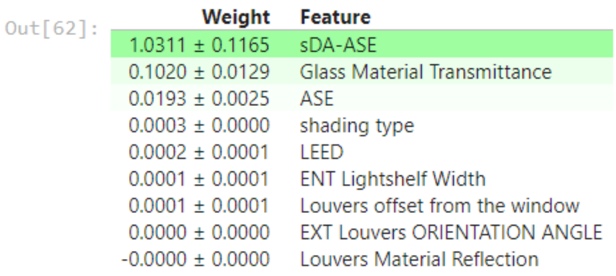
Random Forest Regressor :



Out[61]:

Weight	Feature
0.8664 ± 0.0952	sDA-ASE
0.2606 ± 0.0258	Glass Material Transmittance
0 ± 0.0000	LEED
0 ± 0.0000	ASE
0 ± 0.0000	EXT Louvers ORIENTATION ANGLE
0 ± 0.0000	Louvers offset from the window
0 ± 0.0000	ENT Lightshelf Width
0 ± 0.0000	shading type
0 ± 0.0000	Louvers Material Reflection

XGBOOST :



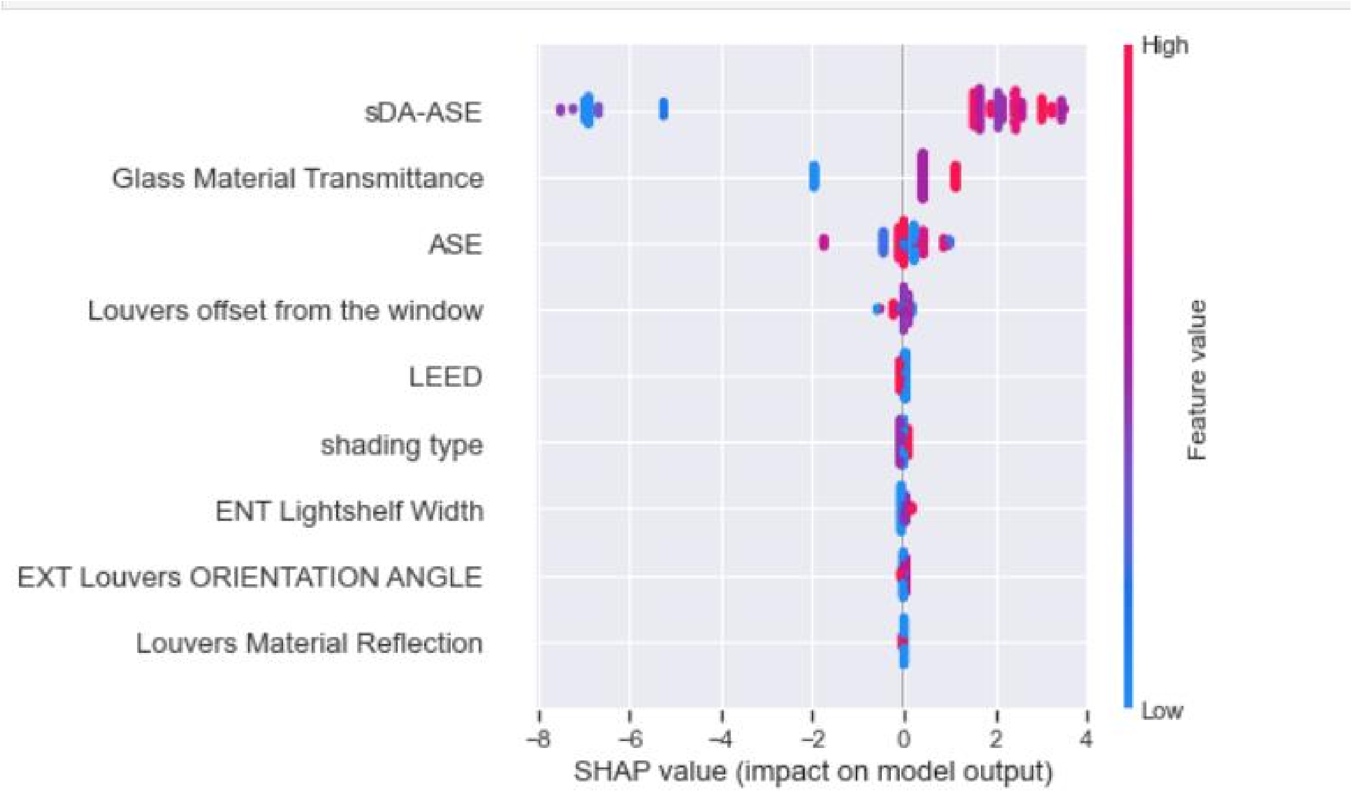
Out[62]:

Weight	Feature
1.0311 ± 0.1165	sDA-ASE
0.1020 ± 0.0129	Glass Material Transmittance
0.0193 ± 0.0025	ASE
0.0003 ± 0.0000	shading type
0.0002 ± 0.0001	LEED
0.0001 ± 0.0001	ENT Lightshelf Width
0.0001 ± 0.0001	Louvers offset from the window
0.0000 ± 0.0000	EXT Louvers ORIENTATION ANGLE
-0.0000 ± 0.0000	Louvers Material Reflection

This Method is called the Permutation Importance the concept of it is it shaffles values in each column a lot of times and some times deletes the column and see the effect of that shuffle on the performance of algorithm , the lower the performance gets the more important the variable is

The first number in each row in the Weight column represents how much the column impacted the algorithm while the second number represents how performance varied from one-shuffling to another

Another methode :

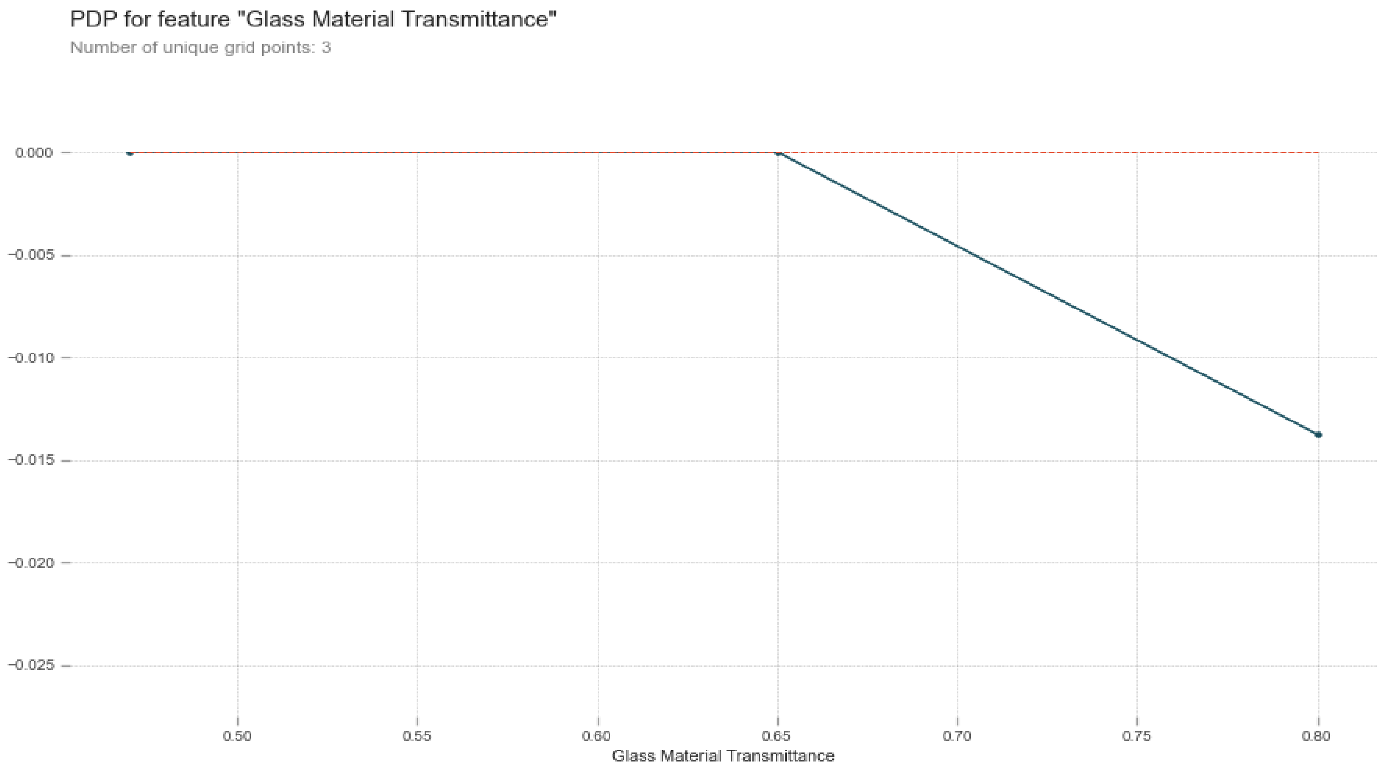


Because random forest regressor is only showing 2 columns we will perform The **SHAP VALUE** method only on the XGBOOST algorithm and as we can see from both 2 methods the top important in order are :

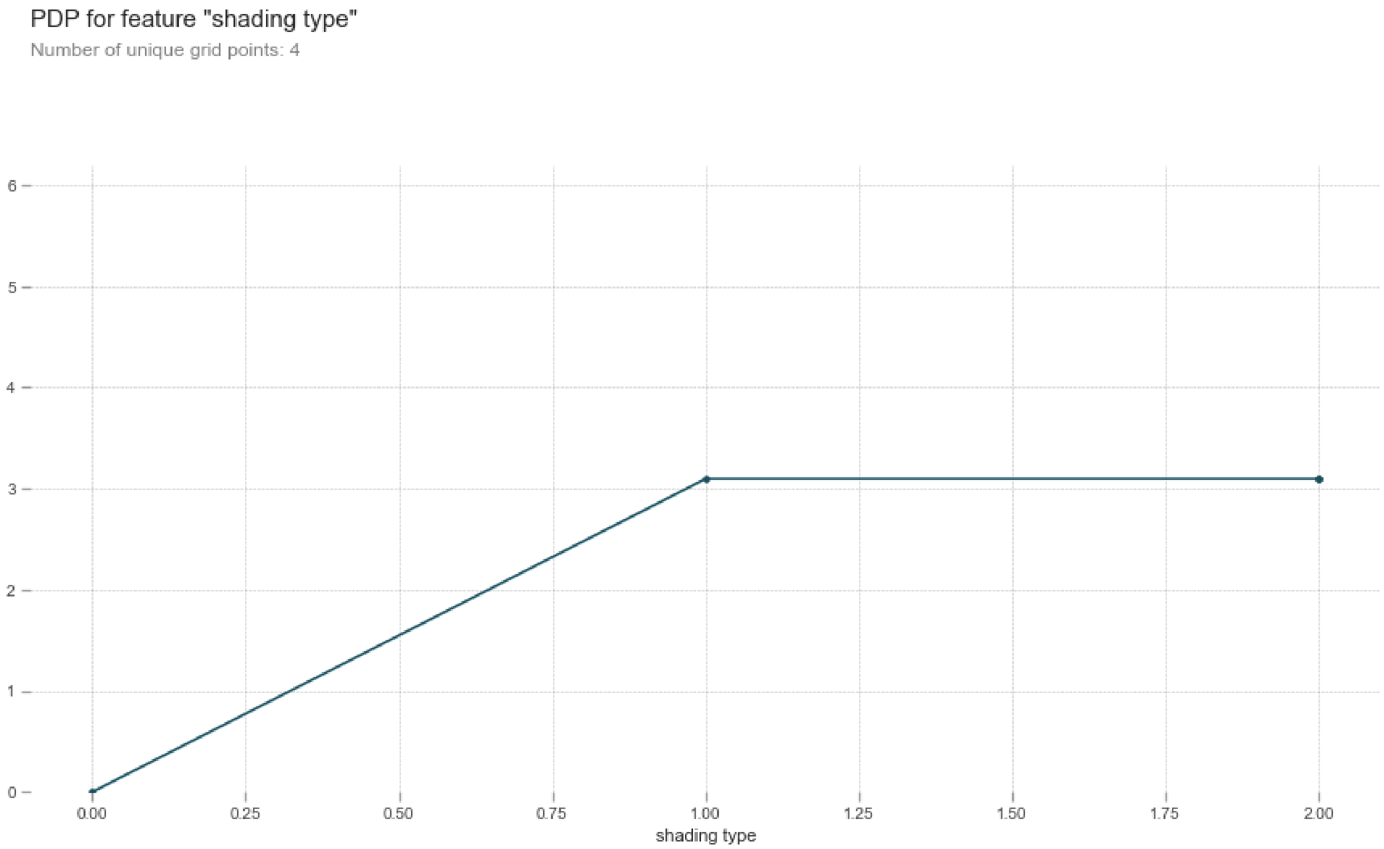
- sDA-ASE
- Glass Material Transmittance
- ASE
- Louvers offset from the window
- LEED
- Shading type

Now we will move into seeing how each of these columns affected the algorithm using Partial Plots

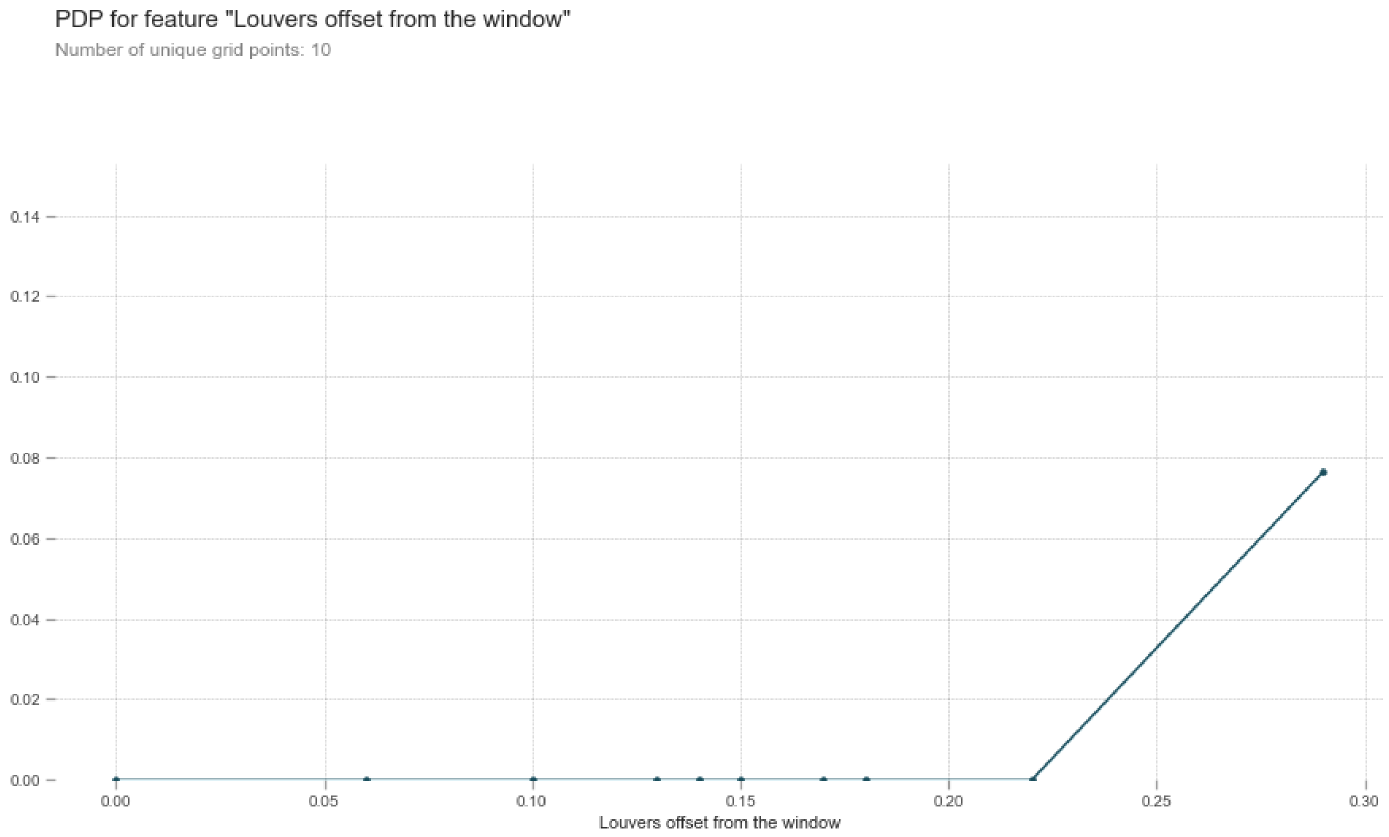
PARTIAL PLOTS : are bunch of graphs and plots that shows the change of predictions at the baseline or given value , they separate each effect from each other and from the target value , they are so useful to see the actual effects of variables on the prediction process



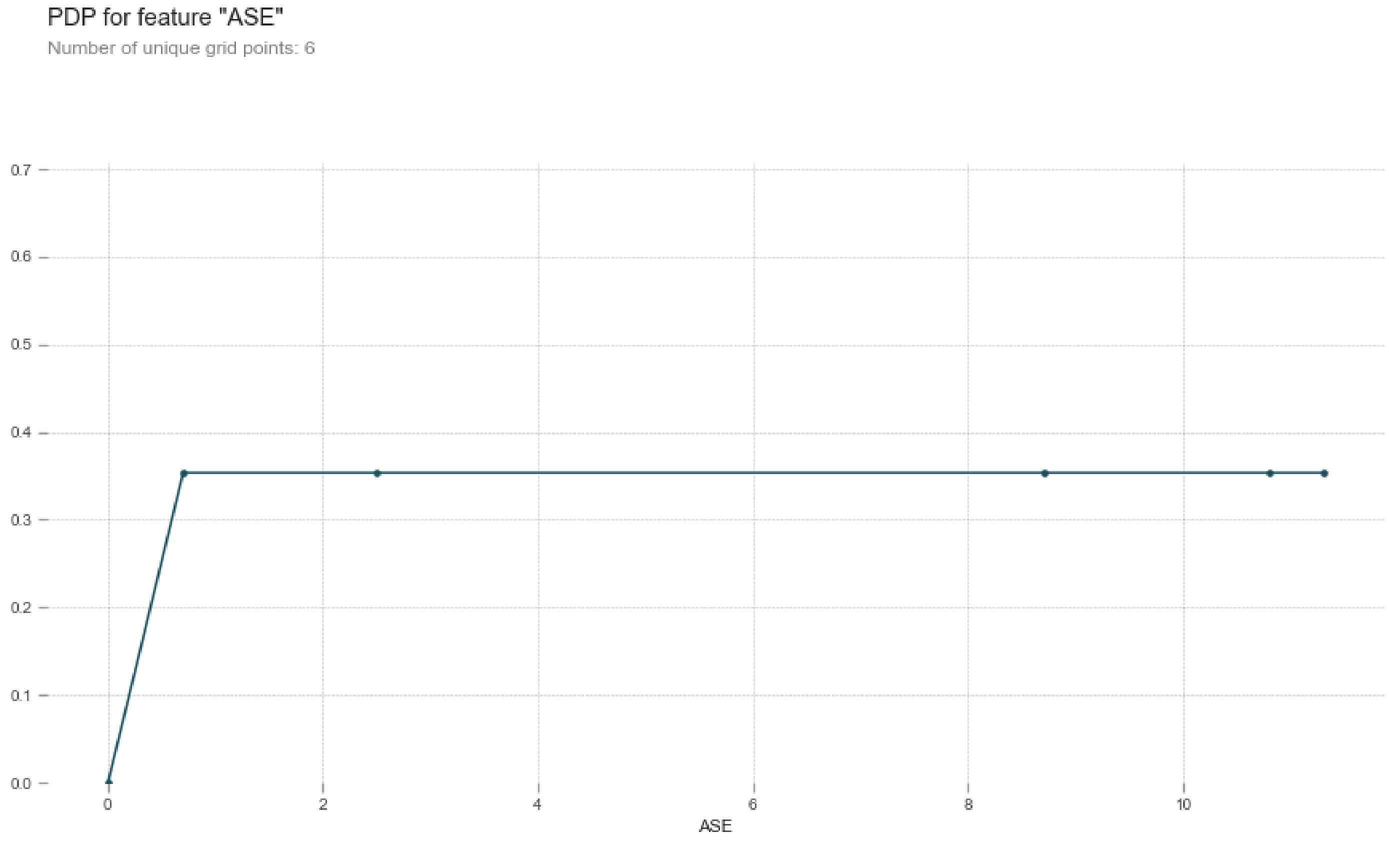
We can see that the **Glass Material Transmittance** doesn't affect the prediction of sDA until its 0.65 it starts affecting sDA negatively



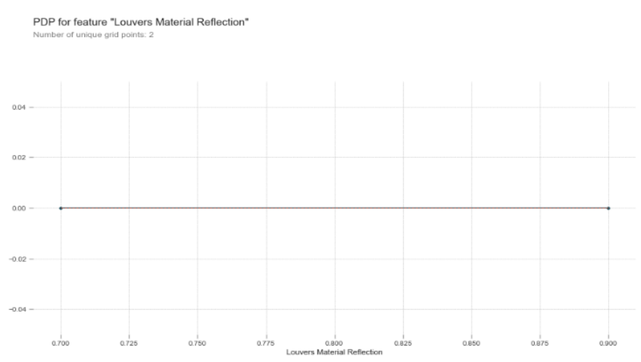
Shading type consist of 0 and 1 value meaning True and false , here shading type don't effect prediction until its 1

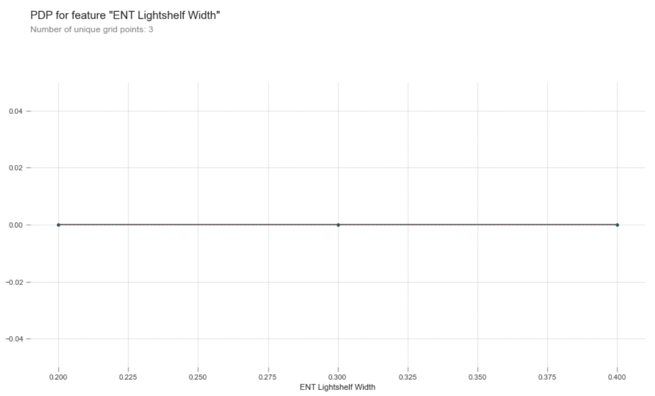

PDP for feature "Louvers offset from the window"

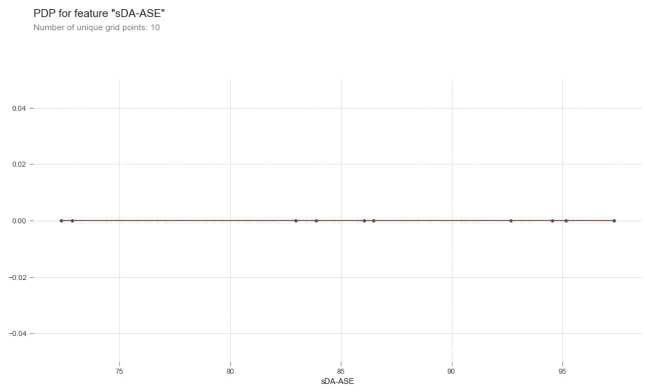
Number of unique grid points: 10

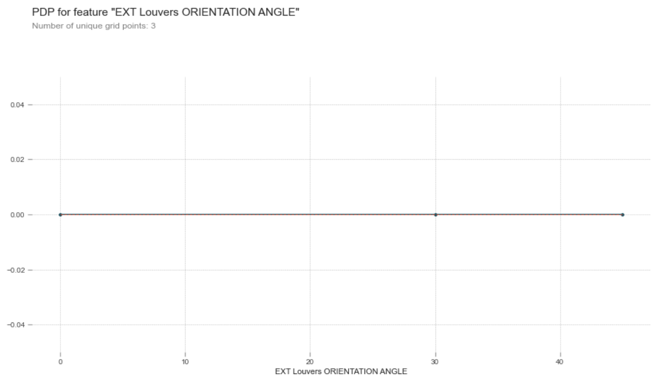

PDP for feature "ASE"

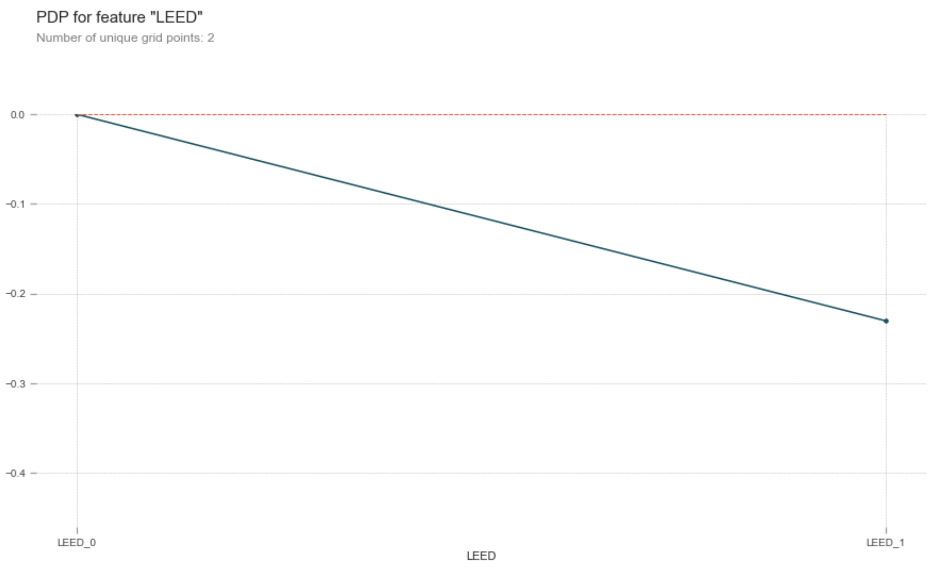
Number of unique grid points: 6











Plots interpretation and Conclusion :

- We can see that the **Glass Material Transmittance** doesn't affect the prediction of sDA until its 0.65 it starts affecting sDA negatively which makes since since the more light the glass allows to pass the less Spatial daylight autonomy in a space
- **Shading type** consist of 0 and 1 value meaning True and false , here shading type don't effect prediction until its 1
- **Louvers offset from the window** effects the prediction until its bigger then 0.22
- **ASE** until its close or 1
- **LEED** consists of 0 & 1 effect sDA negatively when its 1
- **sDA – ASE** don't effect the prediction that's because partial plots separate the effect of sDA-ASE from the target which is sDA and since both column consists of sDA the method showed no effect whats ever on the Prediction

- **Louvers Material Reflection ,ENT Lightshelf Width,EXT Louvers ORIENTATION ANGLE** are the columns that has no effect on the prediction which was proved by all methods

BY : Madjid Erroukrma

Contact :

- **email** : madjidmain@gmail.com
- **Phone** : +213672511865
- **Telegram** : https://t.me/SantoryuuZ