

**SWINBURNE UNIVERSITY OF TECHNOLOGY**

COS40007 – Artificial Intelligence for Engineering

Portfolio Assessment-3: “Let’s develop AI model by your own decision'”

Student Name: Nguyen Cuong Nhat

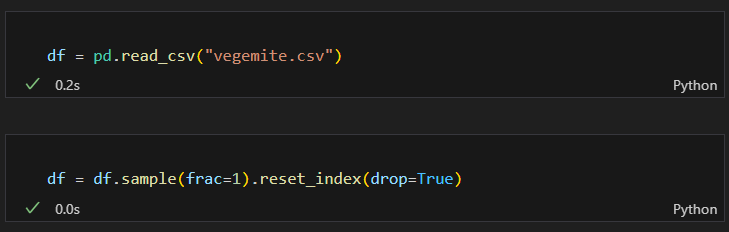
Student ID: 104178590

Instructor: Dr. Pham Thi Kim Dung

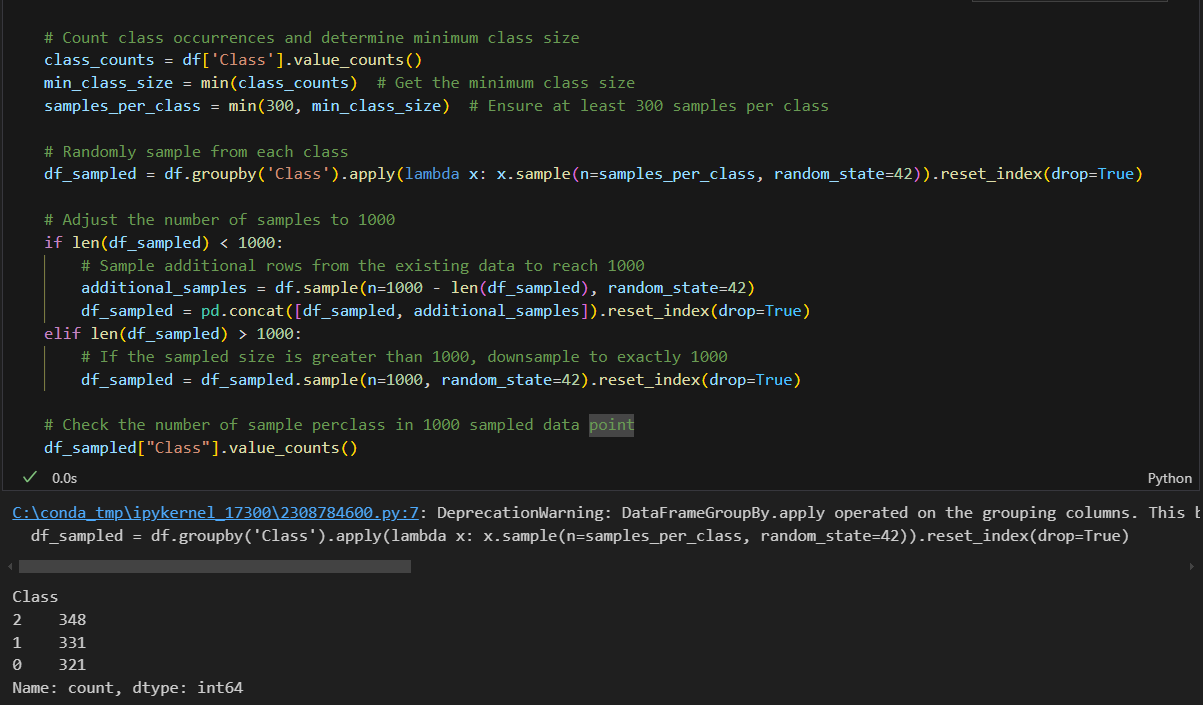
Submission Date: 13/10/2024

# Step 1: Data Preparation

* Shuffle the dataset

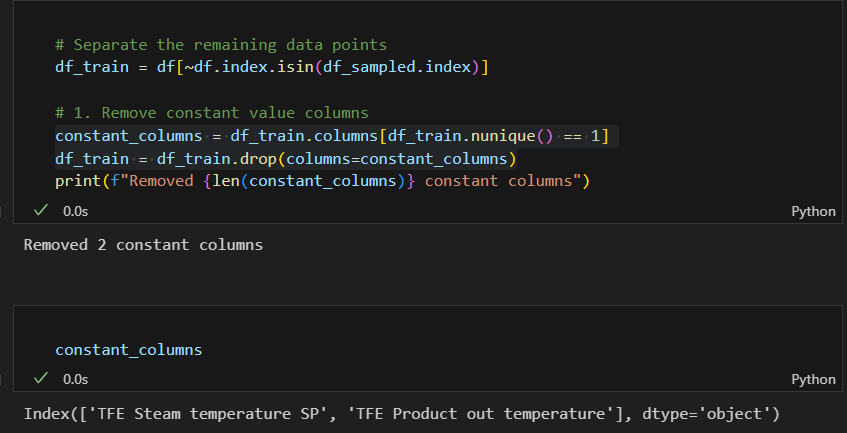


* Randomly take out 1000 data points (rows) such as way that each class in those 1000 samples has near equal distribution (e.g. at least 300 samples from each class)

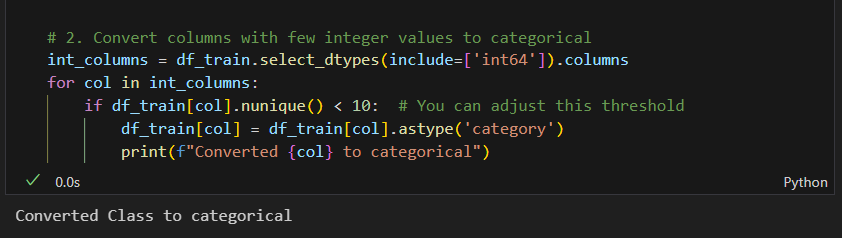


## 1) Does the dataset have any constant value column. If yes, then remove them

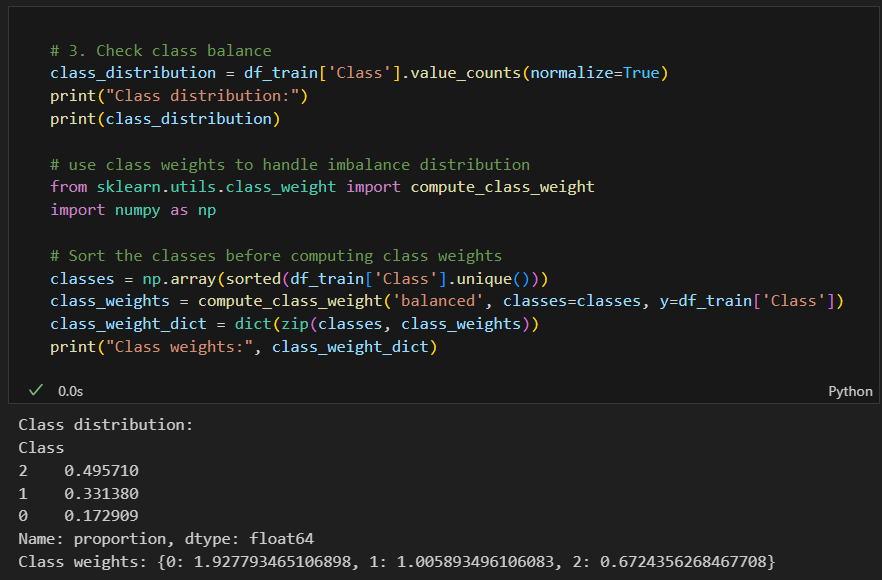
**Answer:** Yes, the dataset has 2 columns with constant value which is 'TFE Steam temperature SP' and 'TFE Product out temperature' => We will remove these 2 columns.

  
2) Does the dataset have any column with few integer values? If yes, then convert them to categorial feature.

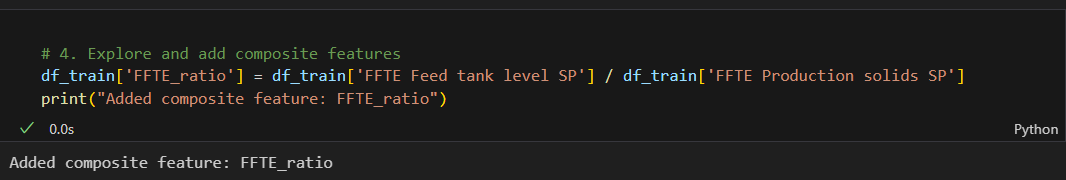
**Answer:** Yes, the “Class” column has few integer values, and it also the target variable. Since we are using classification for this dataset, we will convert the “Class” column to categorical variable

  
3) Does the class have a balanced distribution? If not then perform necessary under sampling and oversampling or adjust class weights.

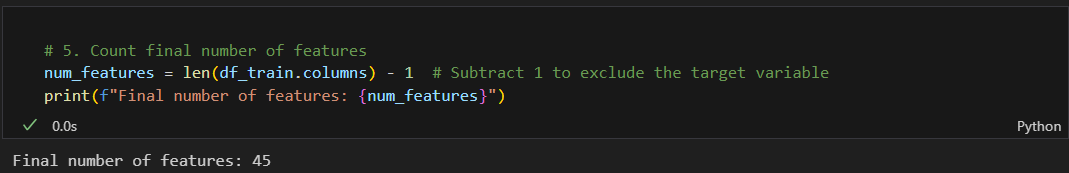
**Answer:** No, the class does not have a balanced distribution. The majority of the samples is from class 2 and class 1 has the least proportion among the 3 classes. So, to handle the imbalance in the data, we will apply class weight for each class.

  
4) Do you find any composite feature through exploration? If so, then add some composite feature in the dataset.

**Answer:** Yes, but it is quite domain specific. After researching, the 'FFTE Feed tank level SP' and 'FFTE Production solids SP' has strong relationship with each other. Therefore, we will create a composite feature based on these two features.

  
5) Finally, how many features you have in your final dataset?

**Answer:** The final dataset has 45 features.

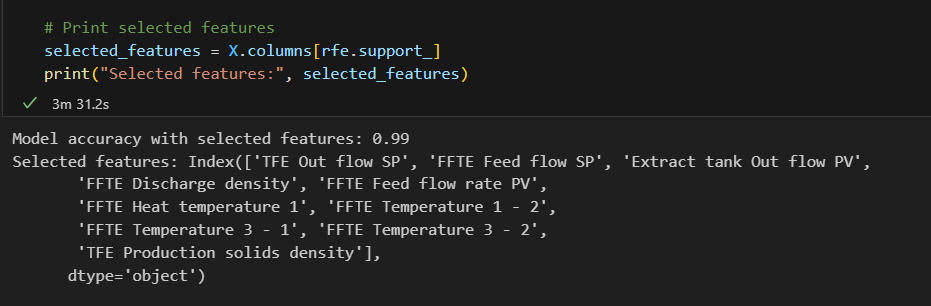


# Step 2: Feature selection, Model Training and Evaluation

## 6) Does the training process need all features? If not, can you apply some feature selection technique to remove some features? Justify your reason of feature selection

**Answer:** No, not all features are always necessary for the training process. Redundant or irrelevant features can negatively impact the performance of a machine learning model by increasing its complexity and potentially leading to overfitting. Feature selection techniques help in identifying and removing such features, thereby improving the model's performance and interpretability.

**=>** We will use Recursive Feature Elimination (RFE) with a Random Forest Classifier to select the top 10 features.

  
*The selected features*

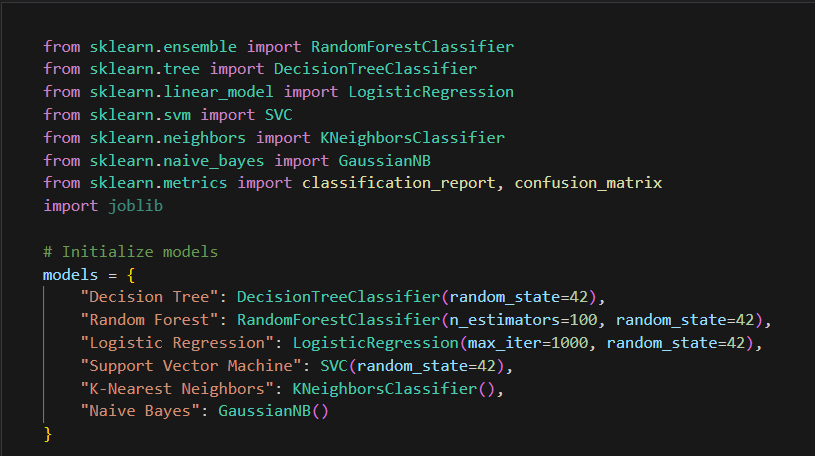
**Justification for Feature Selection**

* Improved Model Performance: By removing irrelevant or redundant features, the model can generalize better to unseen data.
* Reduced Overfitting: Fewer features reduce the risk of the model capturing noise in the training data.
* Enhanced Interpretability: A simpler model with fewer features is easier to understand and interpret.
* Reduced Computational Cost: Training and inference times are reduced with fewer features.

7) Train multiple ML models (at least 5 including DecisionTreeClassifier) with your selected features.

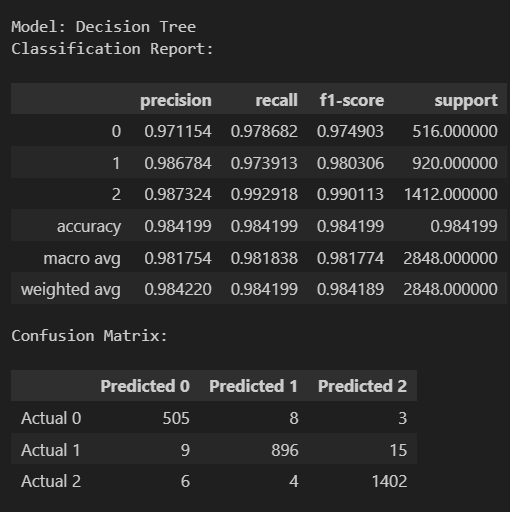
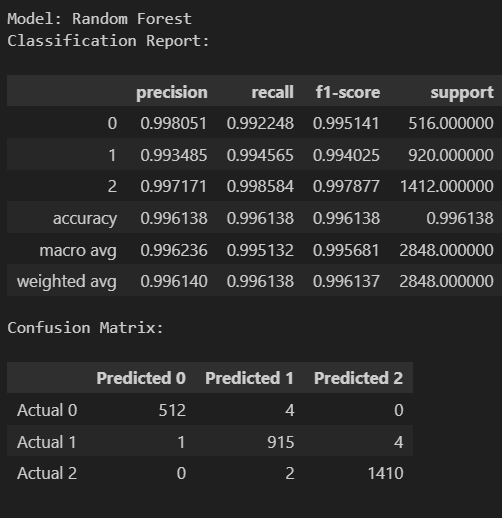
I use the dataset to train 6 machine learning models, including:

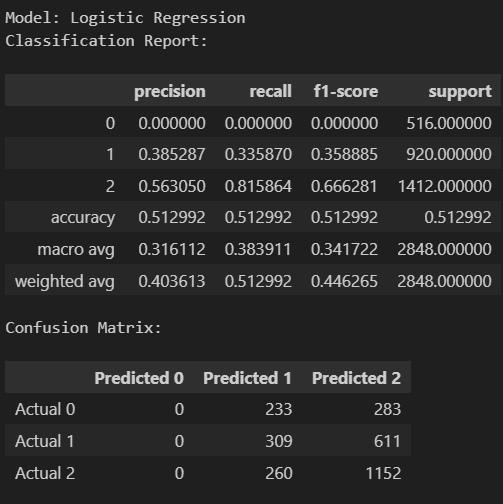
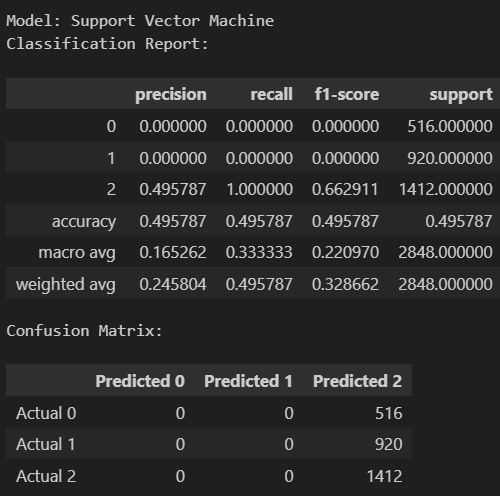
* **Decision Tree**: A tree-like model that splits data into branches based on feature values to make decisions.
* **Random Forest**: An ensemble of decision trees that improves prediction accuracy by averaging multiple tree outputs.
* **Logistic Regression**: A linear model used for binary classification that predicts probabilities of classes using a sigmoid function.
* **Support Vector Machine**: A model that finds the optimal boundary (hyperplane) to separate different classes in feature space.
* **K-Nearest Neighbors**: A model that classifies data points based on the majority class of the nearest data points.
* **Naïve Bayes**: A probabilistic model that uses Bayes' theorem assuming independence between features for classification.

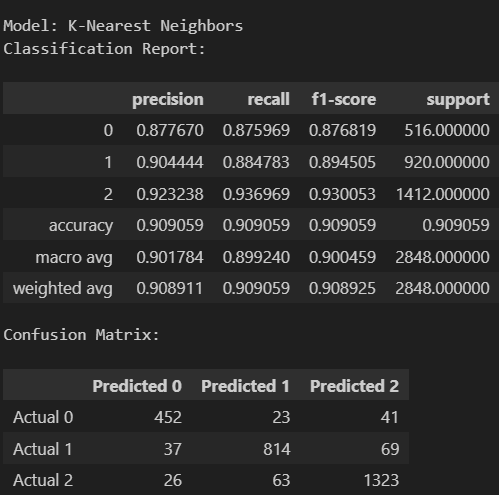
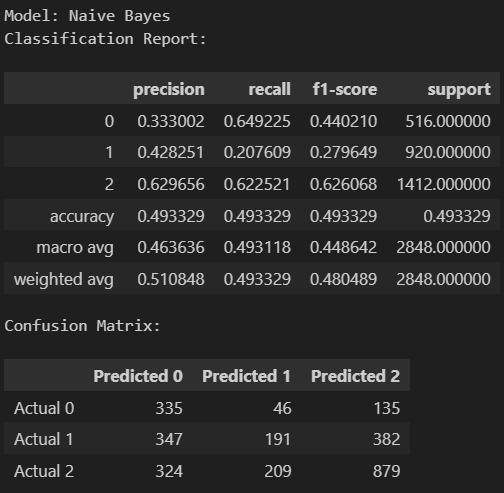


8) Evaluate each model with classification report and confusion matrix

Display the classification report and confusion matrix of each model.

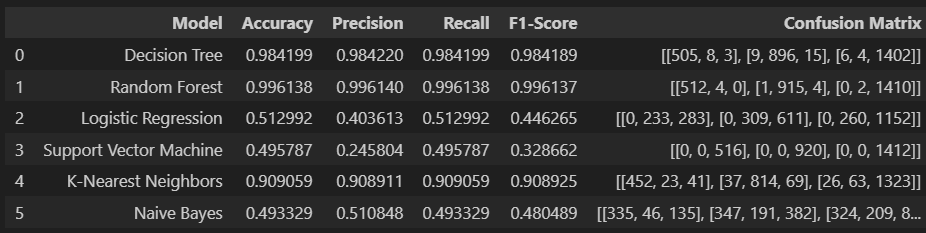
 

## 9) Compare all the models across different evaluation measures and generate a comparison table.

After training, I created the comparison table to compare the performance of 6 models using some common metrics in classification tasks, including:

* **Accuracy**: The proportion of correct predictions out of the total predictions.
* **Precision**: The proportion of true positive predictions out of all positive predictions made.
* **Recall**: The proportion of true positives correctly identified out of all actual positives.
* **F1-Score**: The harmonic mean of precision and recall, balancing the two metrics.



10) Now select your best performing model to use that as AI. Justify the reason of your selection

Based on the classification reports and confusion matrices provided above, the **Random Forest** model appears to be the best-performing model.

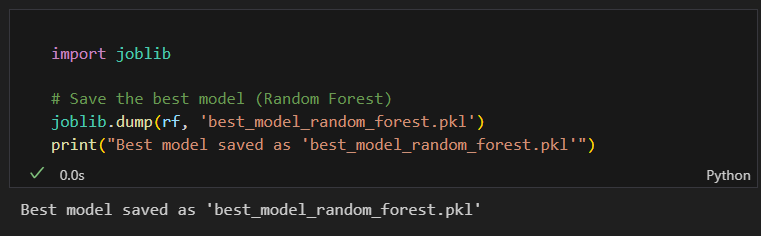
**Justification**:

* **Highest Accuracy**: The Random Forest model achieved the highest accuracy (99.63%) across all models, indicating that it performs well on the overall dataset.
* **Precision and Recall**: It has excellent precision and recall for all classes:
  + Class 0: Precision and recall of 0.998 and 0.993, respectively.
  + Class 1: Precision of 0.997 and recall of 0.994.
  + Class 2: Precision of 0.998 and recall of 1.0.

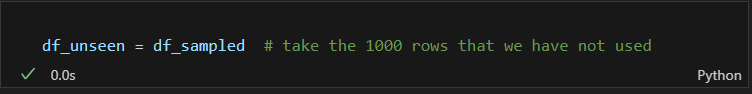
This shows that the model consistently performs well across different classes, reducing the chances of misclassifications.

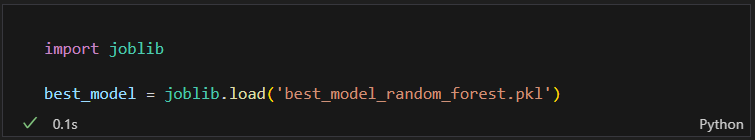
* **Balanced Performance**: The Random Forest model has balanced precision, recall, and F1-scores across all classes, unlike some other models (e.g., Logistic Regression, SVM) which show poor performance for certain classes.
* **Confusion Matrix**: The confusion matrix shows very few misclassifications. For example, the majority of the instances are correctly classified, with negligible errors across the three classes.
* The Random Forest model's combination of high accuracy, well-balanced precision/recall across all classes, and minimal misclassifications makes it the best choice for the AI model in this case.

## 11) Now save your selected model

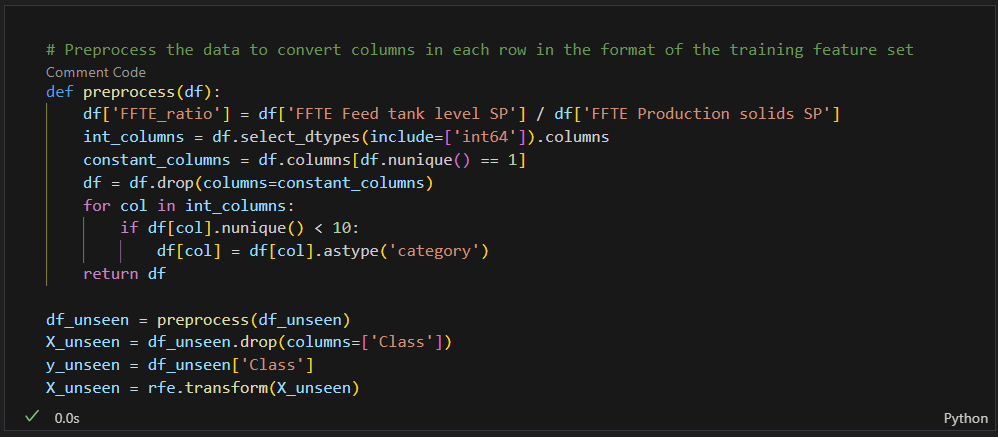


Step 3: ML to AI   
12) Now take the 1000 rows that you have not used (we put aside at the beginning)

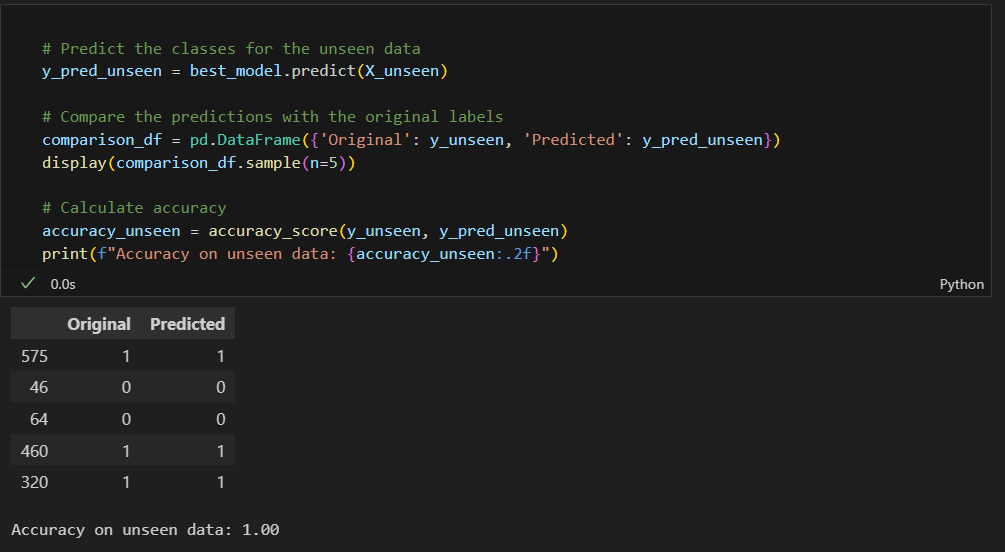
  
13) Load the model



14) Iteratively convert columns in each row in the format of your training feature set

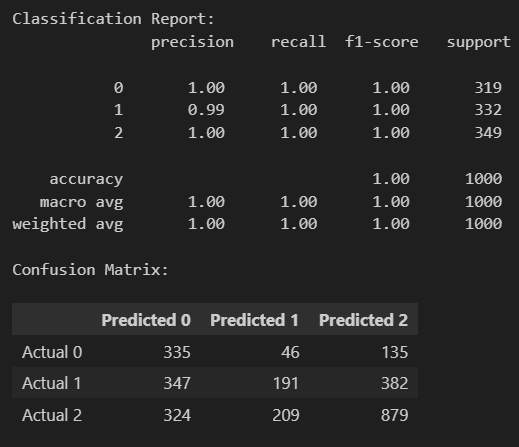
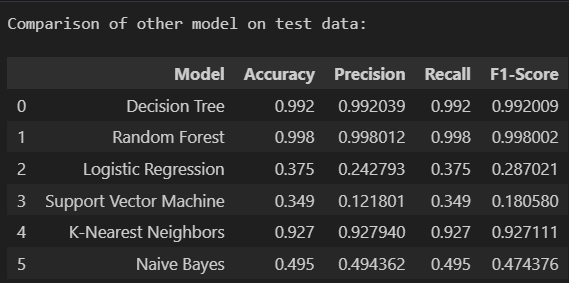


15) Find class prediction using the loaded model and compare with the original label



As we can see, the model correctly predicted the label for all the classes.

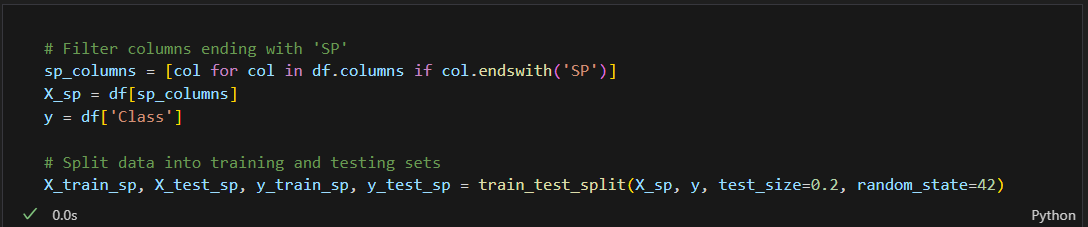
16) Measure the performance of your best model for 1000 unseen data points.

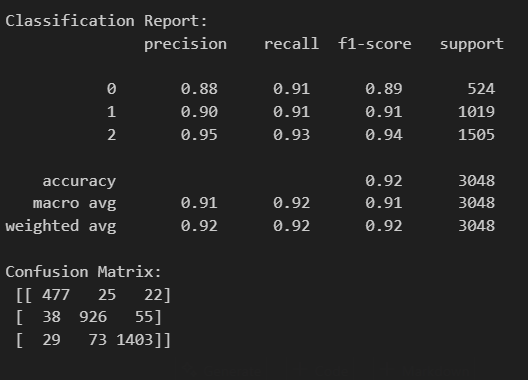
  
17) Now measure the performance of other model using these 1000 data points. Have you observed same result of model selection that you identified through evaluation?  


**Answer**: Yes, the Random Forest Classifier is still the best model in this case. It achieved the highest accuracy, precision, recall and F1-score on the test data

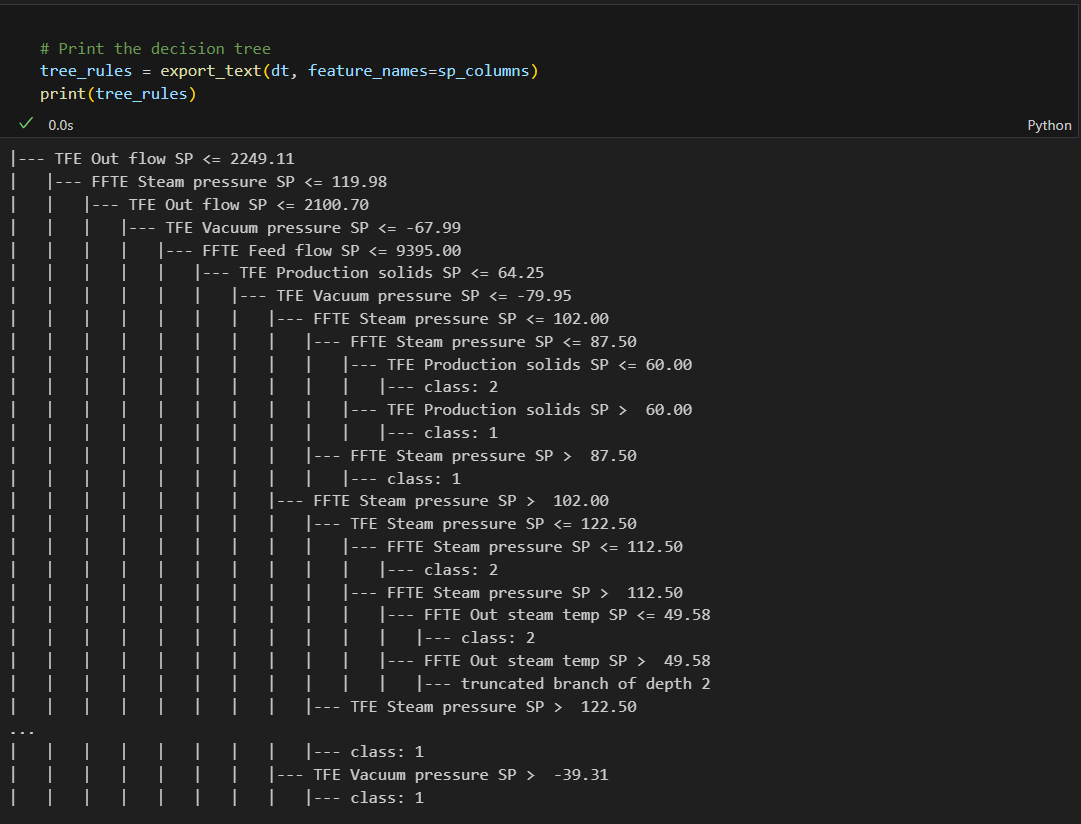
Step 4: Develop rules from ML model

## • Using only SP features generate a DecisionTree model





• Print the tree using export\_text



• Can you now define some rules of SP values for each class?

**Answer**: Yes, here are some set points ranges for each class:

**- Class 1:**

+ FFTE Production solids SP <= 39.50 and FFTE Feed tank level SP <= 42.00

+ FFTE Production solids SP > 39.50 and FFTE Production solids SP <= 42.00

**- Class 2:**

+ FFTE Production solids SP <= 39.50 and FFTE Feed tank level SP > 42.00

**- Class 3:**

+ FFTE Production solids SP > 42.00