

Project Development Phase Performance Testing

Date	17 February 2026
Team ID	LTVIP2026TMIDS90282
Project Name	Weather-Based Prediction of Wind Turbine Energy Output: A Next-Generation Approach to Renewable Energy Management
Maximum Marks	



















Model Performance Testing:

After data is cleaned and it's time to build the model. We can train our data on different algorithms. for this project we are applying four classification algorithms. The best model is saved based on its performance.

Activity 1: Random forest model

A function named randomForest is created and train and test data are passed as the parameters. Inside the function, RandomForestClassifier algorithm is initialized and training data is passed to the model with .fit() function. Test data is predicted with .predict() function and saved in new variable. For evaluating the model, confusion matrix and classification report is done.

```
# Model
forest_model = RandomForestRegressor(n_estimators=300, random_state=1)
forest_model.fit(train_X, train_y)
```

RandomForestRegressor ⓘ ⓘ		
Parameters		
 <code>n_estimators</code>		300
 <code>criterion</code>		'squared_error'
 <code>max_depth</code>		None
 <code>min_samples_split</code>		2
 <code>min_samples_leaf</code>		1
 <code>min_weight_fraction_leaf</code>		0.0
 <code>max_features</code>		1.0
 <code>max_leaf_nodes</code>		None
 <code>min_impurity_decrease</code>		0.0
 <code>bootstrap</code>		True
 <code>oob_score</code>		False
 <code>n_jobs</code>		None
 <code>random_state</code>		1
 <code>verbose</code>		0
 <code>warm_start</code>		False
 <code>ccp_alpha</code>		0.0
 <code>max_samples</code>		None
 <code>monotonic_cst</code>		None

Activity 2: Evaluating performance of the model and saving the model

Here we will be evaluating the model built. We will be using the test set for evaluation. The test set is given to the model for prediction and prediction values are stored in another variable called `y_pred`. The `r2` score of the model is calculated and its performance is estimated.

Note: To understand cross validation, refer this link. <https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85>.

```
[10] # Prediction
      power_preds = forest_model.predict(val_x)

      print("MAE:", mean_absolute_error(val_y, power_preds))
      print("R2:", r2_score(val_y, power_preds))

... MAE: 198.272716823753
      R2: 0.8711607797474977
```

```
[11] # Save model
      joblib.dump(forest_model, "power_prediction.sav")

      print("Model saved successfully")

... Model saved successfully
```