Australia Next Day Rain Prediction

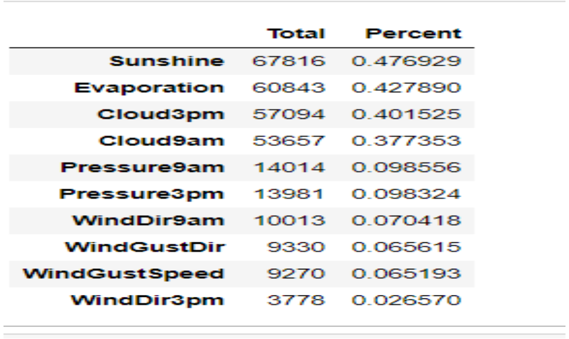
Data Set:

The dataset was obtained from Kaggle, the following is the link to the dataset "https://www.kaggle.com/jsphyg/weather-dataset-rattle-package". The weather dataset contains 142,193 daily weather observations from 49 weather stations across Australia over the period November 2007 to June 2017 and 24 features such as Rain Tomorrow, Min Temp, Max Temp, etc., The dataset excites me because it has lot of missing values and many Categorical features.

Number of Days Rain Tomorrow: 31877

Number of Days No Rain Tomorrow: 110316

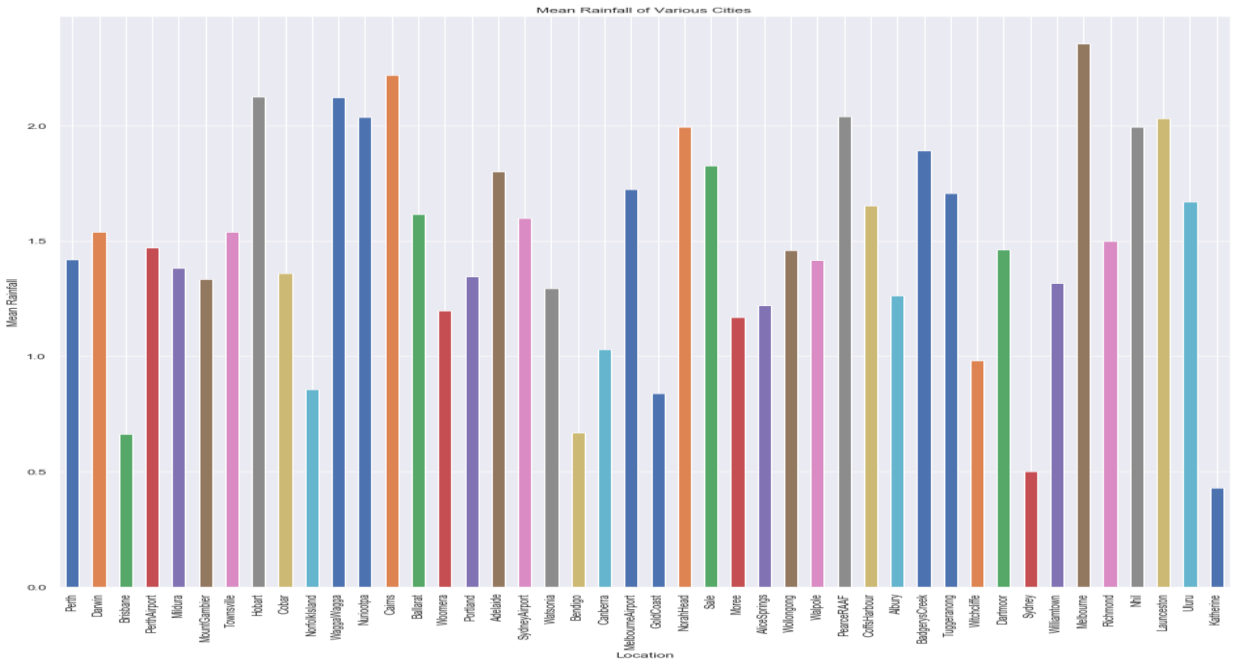
Missing Values:



	Total	Percent
Sunshine	67816	0.476929
Evaporation	60843	0.427890
Cloud3pm	57094	0.401525
Cloud9am	53657	0.377353
Pressure9am	14014	0.098556
Pressure3pm	13981	0.098324
WindDir9am	10013	0.070418
WindGustDir	9330	0.065615
WindGustSpeed	9270	0.065193
WindDir3pm	3778	0.026570

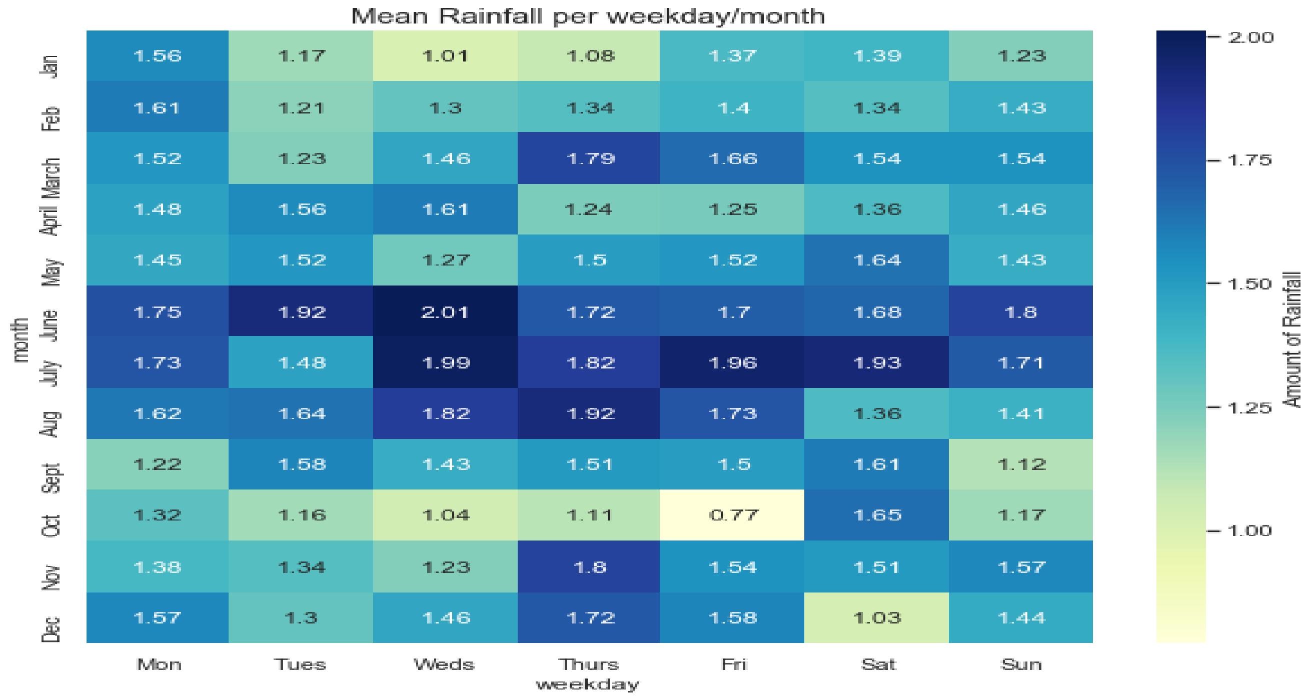
Sunshine has the greatest number of missing values followed by Evaporation, Cloud3am, and Cloud9am.

Average Rainfall of Cities:

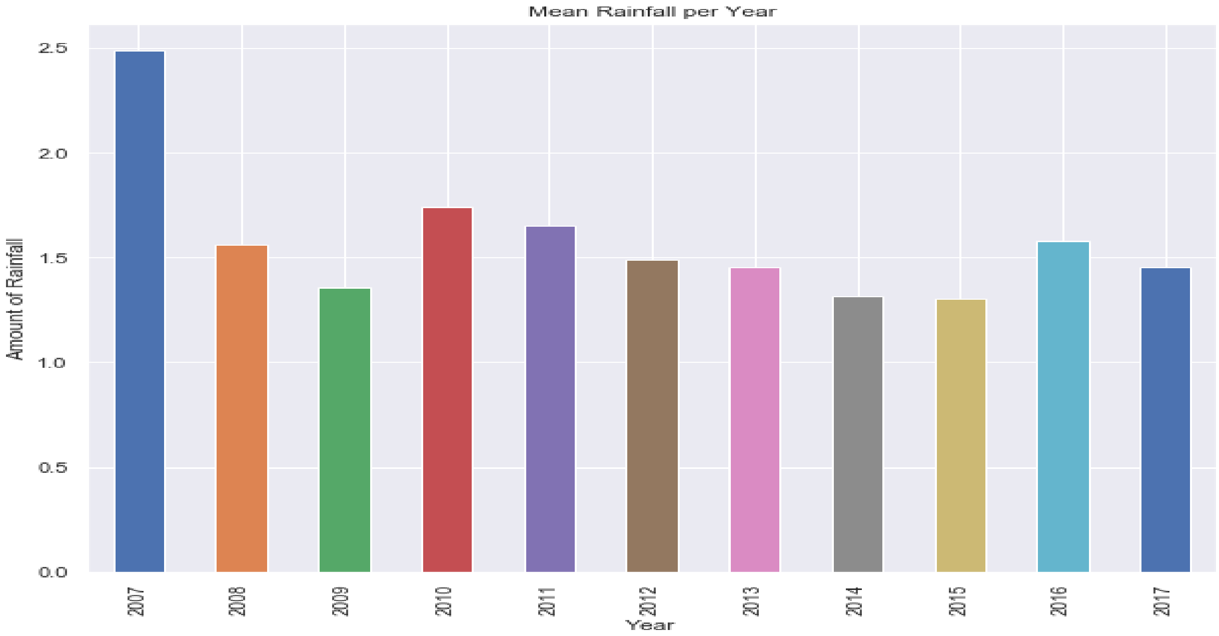


- Melbourne has the greatest Average rainfall.

Monthly Average of Rain: June -Wednesday has the highest average rainfall of 2.01 mm



Yearly Average Rainfall: We can omit 2007 as it has only 2 months data, 2010 has the highest.



Rain Tomorrow Prediction:

Machine learning Algorithms:

1. Support Vector Machine (Linear Kernel) – Hyperparameter tuning using Grid search
2. Support Vector Machine (Gaussian Kernel) - Hyperparameter tuning using Grid search
3. Support Vector Machine (Polynomial Kernel) - Hyperparameter tuning using Grid search

4. Decision Tree (Without hyperparameter tuning) -
5. Pruned Decision Tree - Hyperparameter tuning using Grid search
6. XG Boost
7. XG Boost -Randomized Hyperparameter tuning

Evaluation Metrics:

1. Accuracy
2. Precision
3. Recall
4. F1-Score

Models and Estimation Techniques:

1. **Support Vector Machine (Linear Kernel):**

   K(x, xi) = sum(x * xi)

   Hyperparameters: C= [0.1, 1, 10, 100] and gamma= [1, 0.1, 0.01, 0.001, 0.0001].

I run the SVM linear kernel using the above mentioned hyperparameters using grid search. There are 20 possible combinations, and the found out the best values of C and gamma, which are 1 and 1 respectively. The train and test accuracy are 0.856 and 0.856 respectively. The result is good, and the model is so generalized as both the train and test accuracies are almost same. As shown in fig below (learning Curve). The gap between the training accuracy and test accuracy decreases, the model is generalizing as the training set increases.

2. **Support Vector Machine (Gaussian Kernel/Radial Bias Function):**

The Radial basis function kernel is a popular kernel function commonly used in support vector machine classification. RBF can map an input space in infinite dimensional space.

K(x,xi) = exp(-gamma * sum((x – xi^2))

Hyperparameters: C= [0.1, 1, 10, 100] and gamma= [1, 0.1, 0.01, 0.001, 0.0001]. I run the SVM radial bias kernel using the above mentioned hyperparameters using grid search. There are 20 possible combinations, and the found out the best values of C and gamma, which are 100 and 1 respectively. The train and test accuracy are 0.869 and 0.8572 respectively. The model has got better accuracy compared to the linear SVM. RBF is good at mapping an input space in infinite dimensional space. As shown in fig below. There is a narrow between the training accuracy and test accuracy.

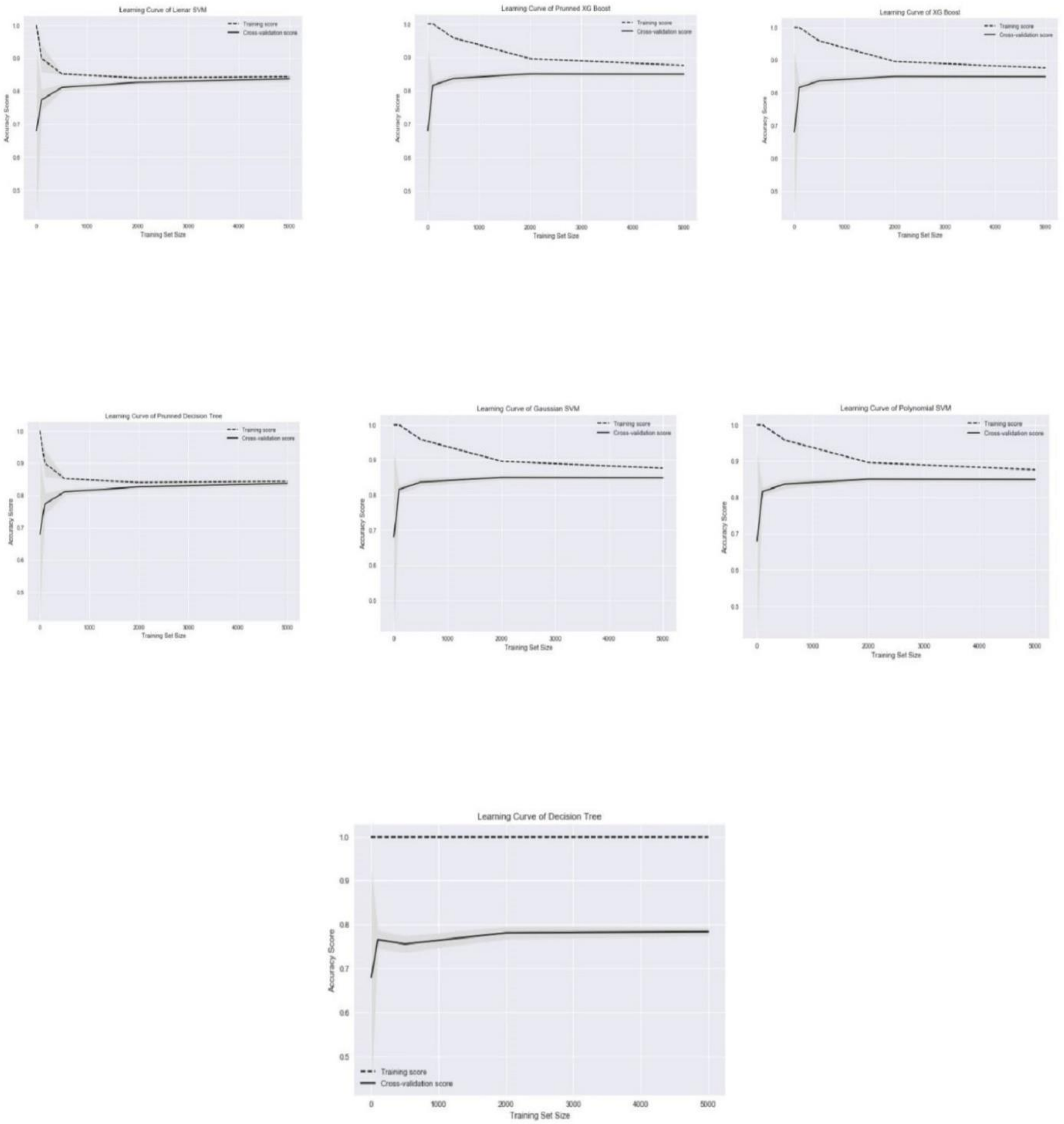
3. **Support Vector Machine (Polynomial Kernel):**

A polynomial kernel is a more generalized form of the linear kernel. The polynomial kernel can distinguish curved or nonlinear input space. **K(x,xi) = 1 + sum(x * xi)^d**

Hyperparameters: C= [0.1, 1, 10, 100] and gamma= [1, 0.1, 0.01, 0.001], Degree = 3, 5. I run the SVM polynomial kernel using the above mentioned hyperparameters using grid search. There are 32 possible combinations, and the found out the best values of degree, C and gamma, which are 3, 10 and 0.1 respectively. The train and test accuracy are 0.891 and 0.85 respectively. The model is the best among SVM, but it is not generalized. As shown in fig below. There is a gap between the training accuracy and test accuracy. The gap is not big. The model is suffering from high variance.

Learning Curve

Training Set Size vs Accuracy

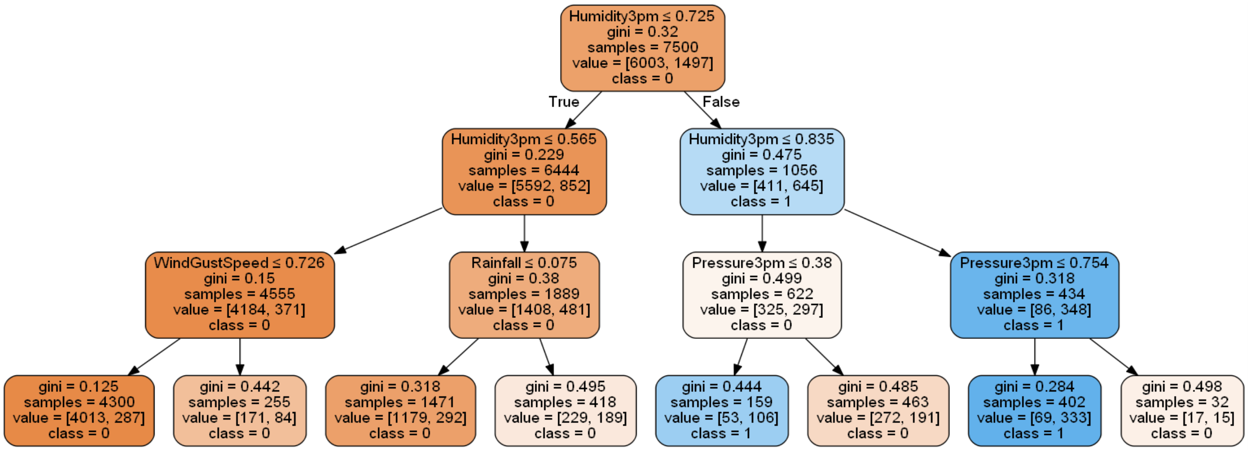


4. Decision Tree:

Decision tree breaks down a data set into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The train and test accuracy are 1 and 0.7952 respectively. There is huge gap between training and test accuracy. The model is suffering from high bias. The training error is zero and the tree used all the features.

5. Pruned Decision Tree:

Parameters: criterion= ["gini", "entropy"], min_samples_split=[2, 10, 20], max_depth= [None, 2, 5, 10], min_samples_leaf=[1, 5, 10], max_leaf_nodes=[None, 5, 10, 20]. The best values are 'criterion': 'gini', 'max_depth': 2, 'max_leaf_nodes': 10, 'min_samples_leaf': 1, 'min_samples_split': 20. The train and test accuracy are 0.843 and 0.8436 respectively. The result is good, and the model is so generalized as both the train and test accuracies are almost same. As shown in fig below. The gap between the training accuracy and test accuracy decreases, the model is generalizing as the training set increases. I used hyperparameter tuning to get the best depth of the tree (2 in this case). Humidity3pm is the root node and has got Gini index of 0.32.



6. XG Boost:

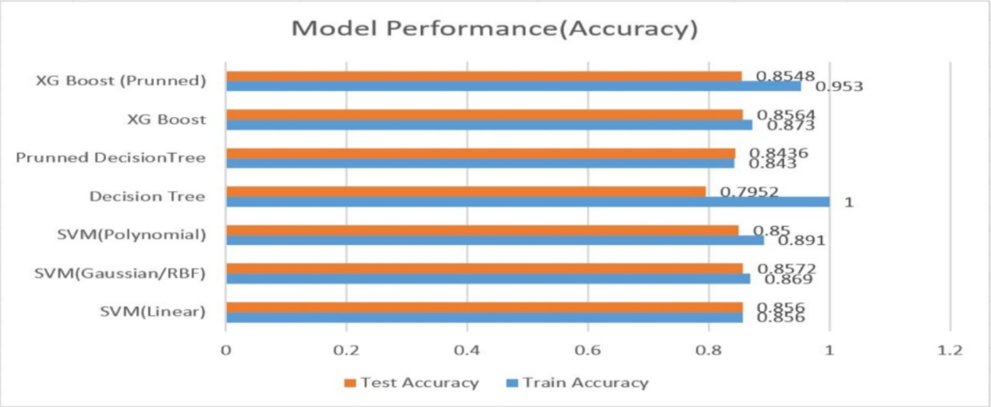
XGBoost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. The train and test accuracy are 0.873 and 0.8564 respectively. XG Boost has performed decently with the default values such as booster: gbtree, gamma :0, max_depth =6.

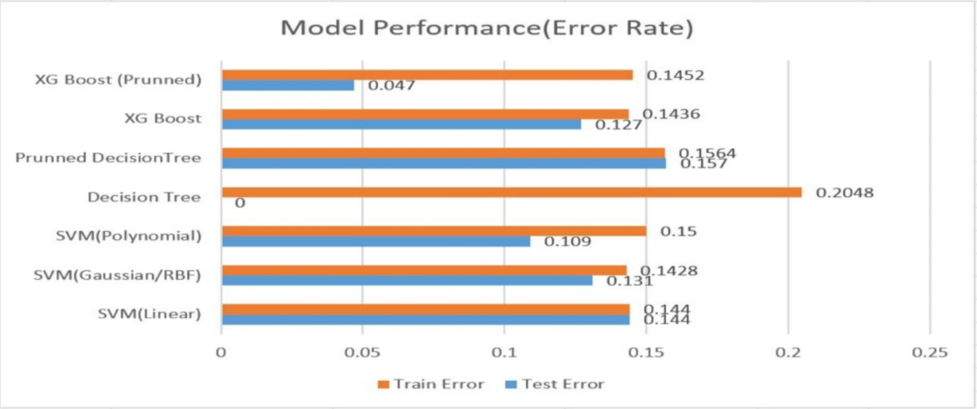
7. XG Boost (Randomized hyperparameter tuning):

Hyperparameters: 'max_depth': [2,3,5,10],'n_estimators': [100,500,1000],'learning_rate': [0.01,0.005,0.001],'min_child_weight': [1,5],'eta':[.3],'gamma': [0,1,5]. The values of the parameters for the best model are “n_estimators': 1000, 'min_child_weight': 1, 'max_depth': 10, 'learning_rate': 0.005, 'gamma': 5, 'eta': 0.3”. The train and test accuracy are 0.953 and 0.8548 respectively. The model is performed well for the training set. There is a small gap between training and test accuracy. The model is suffering from high bias.

Model Performance:

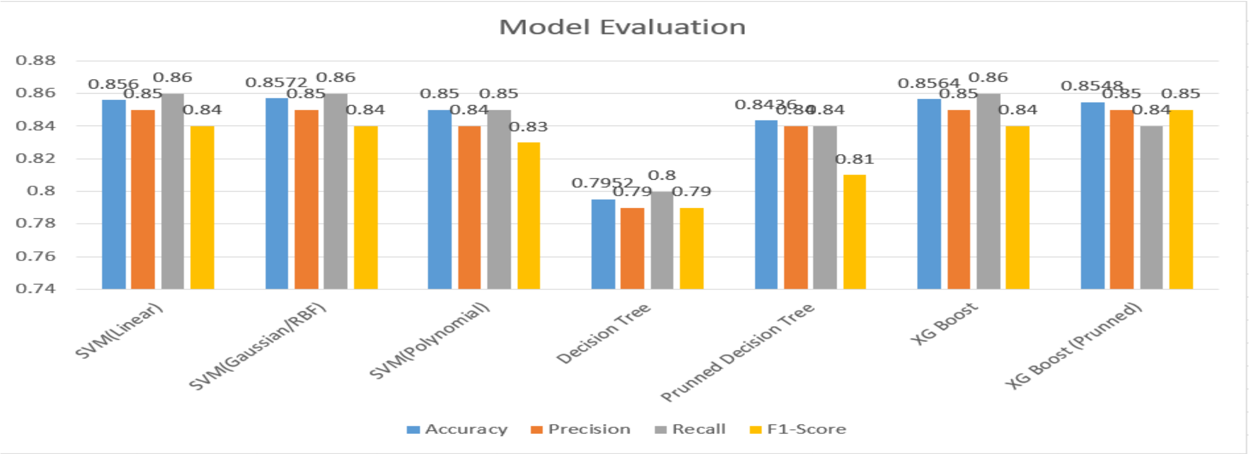
Algorithm	Train Accuracy	Test Accuracy	Test Error	Train Error
SVM(Linear)	0.856	0.856	0.144	0.144
SVM(Gaussian/RBF)	0.869	0.8572	0.131	0.1428
SVM(Polynomial)	0.891	0.85	0.109	0.15
Decision Tree	1	0.7952	0	0.2048
Prunned Tree	0.843	0.8436	0.157	0.1564
XG Boost	0.873	0.8564	0.127	0.1436
XG Boost (Prunned)	0.953	0.8548	0.047	0.1452





Model Evaluation:

Algorithm	Accuracy	Precision	Recall	F1-Score
SVM(Linear)	0.856	0.85	0.86	0.84
SVM(Gaussian/RBF)	0.8572	0.85	0.86	0.84
SVM(Polynomial)	0.85	0.84	0.85	0.83
Decision Tree	0.7952	0.79	0.8	0.79
Prunned Decision Tree	0.8436	0.84	0.84	0.81
XG Boost	0.8564	0.85	0.86	0.84
XG Boost (Prunned)	0.8548	0.85	0.84	0.85



- SVM(Gaussian/RBF) has given the best accuracy, Precision and Recall. XG boost and SVM(Linear) are almost close to SVM (RBF).
- I haven't used the full data set, I have used the sub set of the data (10,000) out of full data set (100,000+). I would get better result if all the instances will be included in model. I omitted some variable cload9am, cloud3pm. Including those variables and filling the missing values with mean or median will yield better results.
- We can extract a variable called month from date column, that will give us better prediction.
- **Cross Validation:** This significantly reduces bias as we are using most of the data for fitting, and significantly reduces variance as most of the data is also being used in validation set.
- SVM(RBF) kernel is the best model in terms of test accuracy, precision and recall. Gamma determines the rbf kernel. As the value of 'γ' increases the model gets overfits. And the value of 'γ' decreases the model underfits. In this case gamma is 1. Best model is better accuracy and generalized not only for train data but for all data. In such scenario, SVM(linear) is the best model.

Conclusion:

1. SVM(Gaussian/RBF) performed better in both the data sets. It has better accuracy, precision and recall.
2. Pruned Decision tree worked well for Australian Rain but very badly worked for appliance energy.
3. Better results can be obtained if all the data points will be used in the model and adding relevant variables, extract new variables and implement hyperparameter tuning with wide range and 10-fold cross validation with 3 repeats.