QUANTITY PREDICTION MODELS

* All the Regression models that predict a quantity , a numerical feature(price, size, salary, other domain specific features) come under quantity prediction models.
* The Regression models can be classified into 2 categories based on the nature of the data :-

1. Linear Models - Linear regression
2. Non-linear Models - Decision Tree regressor, Random Forest, Bagging regressors, Boosting regressors (AdaBoost,GradientBoost,XgBoost)

Use **Linear regression** when you prefer **interpretability** and of course when you have linear data in hand(predicting house **prices** based on its features and explaining the real estate company why they have to **invest** in a particular house). **Evaluation Metrics** is much important otherwise the Investor might loose his money!!!

But what if the real estate company gave you the data that’s non-linear, you still have to predict house prices , but the data this time has qualitative features, company appointed brokers and gave them questionnaire(survey) sheet to fill and collect the data ,like

“is the house far from Airpot? yes/no”,”is there a school near by?”,”Is this a recent construction?,some numerical features Sqft, no. of bedrooms and finally house price. In this case the Linear Regression model won’t be an ideal choice , go for **DECISION TREE**.

**Decision Tree** does the necessary prediction based on splitting conditions (using features) and impurity indices (gini, entropy) predicts a price (average of all the records’ prices in the corresponding leaf), **but what’s the guarantee that this is an accurate prediction?** As **DT** has the habit of getting **greedy** (only using best features for splitting)and also very easily **overfits** (as it’ll not give up till it eliminates impurity), **NOW WHAT ?**

**RANDOM FOREST** comes to our rescue ! As Random forest is an ensemble of DTs and because of data being distributed to all the DTs using BootStrap sampling, each DT will get data of different distributions and each DT uses therefore different features to split, The Random forest ensemble therefore considers more features than DT and captures info from all the variance zones (1sigma,2sigma,3sigma) of the data and gives an average output , the Price , and predictions will be **much more accurate than DT.**

In this process Random Forest might have reduced the overfitting(high variance situation) drastically but still the **greediness (high bias)** might continue, still RF doesn’t use all the features, this high bias situation can be efficiently solved by a **Boosting(AdaBoost,GradientBoost,XgBoost) model.**

* AdaBoost - is an ensemble which learns Sequentially, first model/regressor learns from the data (that’s again bootstrap sampled), next model learns from the misclassifications done by the previous one and so on..
* Gradient Boost - Again sequential learning , but this time models learn from residuals(not data points like in the previous case), residuals are the difference between actual and predicted values.

NOTE - In the **stage after decision tree** where we are going for Random Forest we can also go for a **BAGGING CLASSIFIER** that eliminates overfitting problem of DT.