1 Bias → It is the amount that a model prediction differ from the target value, compared to the training dataset.
(A high level bias can lead to underfitting.)
(Bias always with training data)
A linear algorithm often has a high bias, which make them learn fast.

~~Ques~~ ⇒
Low Bias ML Algorithm → Decision Tree, K-NN, SVM

High Bias ML Algorithm → Linear Regression, Logistic Regression

2 Variance error → It is the amount that the estimate of the target function will change if different training data was used.
(Variance can lead to overfit)
(Variance always with testing data)
A model with high Variance may reflect random noise in the training dataset instead of target function. A model with low variance mean sample data is closed to prediction

Low-Variance ML algo - Linear Regression, Logistic Regression

High-Variance ML algo - Decision Tree, KNN SVM

Note - The goal of any Supervised ML algorithm is to achieve low bias & low Variance

Note - increasing the bias will decrease the Variance, increasing the Variance will decrease the bias

3 Total error ⇒ It is the Sum of the bias error and Variance error.
The total goal is to balance bias & Variance, So the the model does not underfit & overfit the data.

Q When Variance Occur?

Ans If the ML model Perform well with the training dataset, but does not perform well with the test dataset, then Variance Occur.

$$\text{Variance } \sigma^2 = \frac{\sum_{i=1}^{N} (x - \mu)^2}{N} \quad \text{or} \quad \sigma^2$$

(data point)

Variance → are the predicted value how scattered they from each other.

Actually mean of variance is how the data value are away from the mean of the data.

Overfitting & Underfitting

4 Underfitting — When ML is underfitting, it means it isn't learning much from the training data. (It didn't have enough information on the target variable or (it perform bad on training data as well as perform poor on testing data.

— Reason of Underfitting —

- High bias & low Variance.
- The size of training data set used is not enough.
- The model is too simple. So used hyperparmets
- Training data is not clear & also contain noise in it.

— Technique to reduce underfitting —

- increases model Complexity by using hyperparmet
- Remove noise from the data.
- increase the duration of training to get better result.

5 Overfitting — Model is said to be overfitted when the model does not accurate prediction on testing data.
When model get trained with so much data, it start learning from the noise & inaccurate data entries in our data set.

Ex- Overfitting might cause our ~~AB~~ model to predict that every person coming to our site will purchase.

— Reason of Overfitting —

- High Variance & low bias
- The model is too complex
- ~~Small Size of training data~~

Note : Overfitting ~~is~~ give good ~~accuracy~~ accuracy in training dataset but bad accuracy in testing data set

Whereas

Underfitting give bad accuracy in testing as well as training dataset

→ Static model

```
>>> import statsmodels.formula.api as smf
>>> lm = smf.ols(formula='sales ~ TV+radio', data=df).
                .fit()
>> lm.summary()
```

Residual = actual − predicted

$$r = y - \hat{y}$$

R-Squared & Adjusted R-Squared for Regression

Residual Sum Square − RSS $= \sum (y_i - \hat{y}_i)^2$ = Loss

∴ The lower RSS, the better is the model prediction.

Total Sum Square: TSS $= \sum (y_i - \bar{y}_i)^2$

$$R^2 = 1 - \frac{RSS}{TSS}$$

$$\text{Adjusted } R^2 = \left(1 - \frac{(1-R^2)(N-1)}{N-P-1}\right)$$

n = no. of data point

P = no. of independent variable

∴ Adjusted R^2 says that he will give more reliable result as compare to R^2.

∴ for Linear Regression or the linear Regression algorithm it is better to check Adjusted R^2 instead of R^2 as in Adjusted R^2 it take n (no. of data point) as well as p (no. independ variable) to control "overfitting".

V.Imp

Note :- If we add useless data or column in our dataset then also R^2 is going to increase by small amount which is not necessary so we use or see adjusted R^2 accuracy score as it divided & Substract, no. of data point (N) & the no. of independent variable (p)

6 VIF ⇒ Variance inflation factor to the multi-colinearity among the features of the dataset.

VIF = Variance inflation factor =

$$\Rightarrow \left(\frac{1}{1-R^2}\right)$$

(If VIF > 5 (greater than 5) then the features are multi-colinearity)