12/01/2020

3) Data Analysis in Depth

Distance Based Models K-means:

Distance metric uses distance function which provides a relationship metric between each element of the data set.

There are many ways to reach a particular destination.

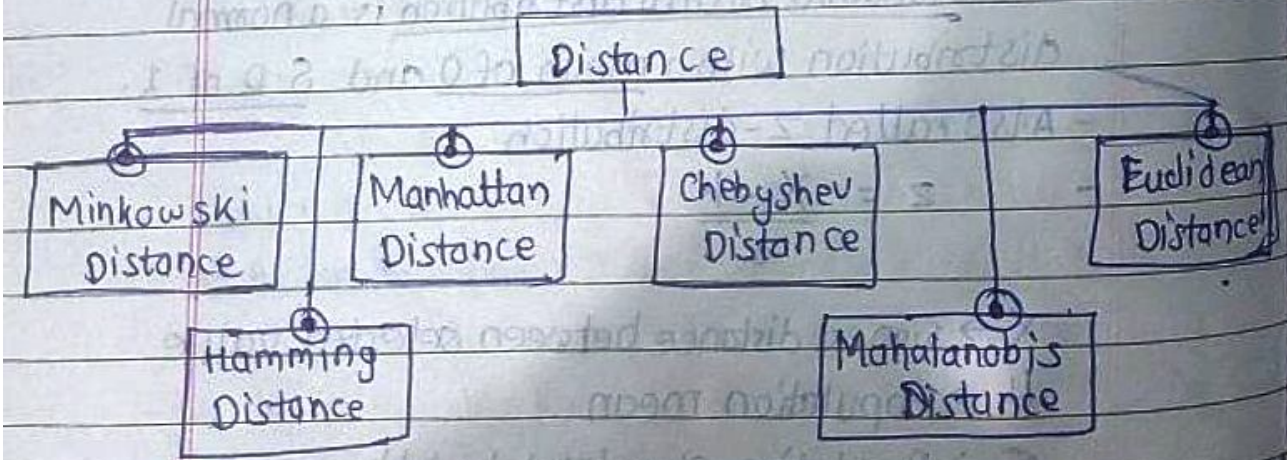
Distances are very useful to find similarities and dissimilarities between two instances.

Distance Measures w.r.t chess:

There are six different pieces to use while playing chess game. Each piece is governed to move by certain set of instructions which restricts its movement in some way. Likewise:

A mountain range can be considered as large detour while using car, train or foot but it is easy to cross it flying.

Distances are used to measure similarity:



Minkowski Distance:

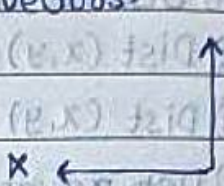
$$d(x,y) = \|x,y\|_m = \left[\sum_{i=1}^{N} (x_i - y_i)^m\right]^{1/m}$$

Norm

If $m=1$, then it becomes Manhattan Distance
$m=2$, —— " —— Euclidean Distance
$m=\infty$, —— " —— Chebyshev Distance

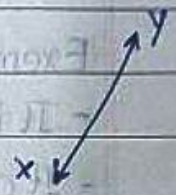
Manhattan Distance:

Manhattan Distance is calculated as the sum of the absolute differences between two vectors.



$$|x_1 - x_2| + |y_1 - y_2|$$

Euclidean Distance:



$$\sqrt{(x_1 - x_2)^2} + \sqrt{(y_1 - y_2)^2}$$

Chebyshev Distance

$$Dis_\infty(x,y) = \max_j |x_j - y_j|$$

Hamming Distance:

- Bits needed to change that is the Hamming distance
- Calculates the distance between two binary vectors, also referred to as binary strings or bits.

red = [1, 0, 0]
green = [0, 1, 0]
blue = [0, 0, 1]

Mahalahobis Distance:

$$d(x,y) = |x-y|^T S_{xy}^{-1} |x-y|$$

- Introduced by Prof. P.C. Mahalanobis in 1936 and is used in many statistical applications ever since.
- It transforms the columns into uncorrelated variables
- Scale the columns ~~into uncorrelated~~ to make their variance equal to 1.
- Finally calculates Euclidean distance.

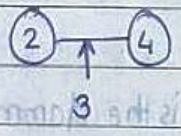
Properties of Distance:

① Distance between point and itself is zero.
② Dist $(x,y) >= 0$.
③ Dist (x,y) = Dist (y,x) are symmetric
④ Detours cannot shorten distance.

Exemplars:

- It is example, sample, instance.
- It can be ideal case for consideration for testing other sample.
- Neighbours :- closest of to exemplar.

① Arithmetic mean:



$\frac{2+4}{2} = \frac{3}{=}$

② Geometric mean:

A term between two terms of geographic sequence is the geometric mean of two terms.

Geometric mean of 2 & 18 $= \sqrt{2 \times 18} = 6$

③ Centroid:

- Centroid is an arithmetic mean

A = [1, 2, 3, 4, 5]

= 1+2+3+4+5/5

= 3

④ Medoid:

- Medoids are representative objects of a data set or a cluster with a data set whose average dissimilarity to all objects in cluster is minimal.

A = [1, 2, 3, 4, 5]

Medoid = 3

- It is always a member of given set.

Clustering:

- Grouping Unlabelled examples is called as clustering.
- If examples are labelled, then it becomes classification

~~Distance Based Clustering~~:

13/01/2020

Given a data Matrix X, the scatter matrix is,

$$S = (X - 1\mu)^T (X - 1\mu) = \sum_{i=1}^{n} (x_i - \mu)^T (x_i - \mu)$$

(Transpose)

μ = row vector containing all columns means of X

Where scatter of X is defined as,

$$Scatt(X) = \sum_{i=1}^{n} \| (x_i - \mu)^2 \|$$

Imagine now that we have partitioned D into K subsets D1 ... Dk = D, and let μj denote the mean of Dj. Let S be the scatter matrix of D, and Sj be the scatter matrices of Dj. These scatter matrices have following relationship:

$$S = \sum_{j=1}^{K} (S_j + B)$$

Here, B is the scatter matrix that results by replacing each point in D with the corresponding μj

K-means Clustering:

The K-means problem is NP complete, which means that there is no efficient solution to find the global minimum and we need to resort to the heuristic algorithm.

The algorithm iterates between partitioning the data using the nearest-centroid decision rule, and recalculating centroids from a partition.

Algorithm:

Input :- Data $d \subseteq R^d$; no. of clusters $K \in N$

Output :- K cluster means $\mu_1, \ldots, \mu_K \in R^d$.

① Randomly initialize K vectors $\mu_1, \ldots, \mu_K \in R^d$.

② Repeat

③ Assign each $x \in D$ to $\arg\min_j Dis2(x, \mu_j)$;

④ for $j = 1$ to K do

⑤ $D_j \leftarrow \{x \in D \mid x \text{ assigned to cluster } j\}$;

⑥ $\mu_j = \frac{1}{D_j} \sum_{x \in D_j} x$

⑦ end
⑧ until no change in $\mu_1, \ldots, \mu_K$,
⑨ return $\mu_1, \ldots, \mu_K$

Ex:- Given Data Points are
(2,9), (1,5), (7,4), (5,8), (6,4), (7,5), (1,2), (4,9)
Let,
$X_1 = (2,9)$ $X_2 = (1,5)$ $X_3 = (7,4)$ $X_4 = (5,8)$
$X_5 = (6,4)$ $X_6 = (7,5)$ $X_7 = (1,2)$ $X_8 = (4,9)$

Let us consider K = 3 – (any number) –
Initial centroids, $C_1 = X_1(2,9)$ $C_2 = X_4(5,8)$
$C_3 = X_7(1,2)$
– (can take any 3 (k) at random) –

Iteration 1

	Distance from C1 (2,9)	Distance from C2 (5,8)	Distance from C3 (1,2)
X_1 (2,9)	$\sqrt{(2-2)^2+(9-9)^2} = 0$	$\sqrt{(2-5)^2+(9-8)^2} = 3.16$	$\sqrt{(2-1)^2+(9-2)^2} = 7.07$
X_2 (1,5)	$\sqrt{(1-2)^2+(5-9)^2} = 4.12$	$\sqrt{(1-5)^2+(5-8)^2} = 5$	$\sqrt{(1-1)^2+(5-2)^2} = 3$
X_3 (7,4)	$\sqrt{(7-2)^2+(4-9)^2} = 7.07$	$\sqrt{(7-5)^2+(4-8)^2} = 4.47$	$\sqrt{(7-1)^2+(4-2)^2} = 6.32$
X_4 (5,8)	$\sqrt{(5-2)^2+(8-9)^2} = 3.16$	$\sqrt{(5-5)^2+(8-8)^2} = 0$	$\sqrt{(5-1)^2+(8-2)^2} = 7.21$
X_5 (6,4)	$\sqrt{(6-2)^2+(4-9)^2} = 6.40$	$\sqrt{(6-5)^2+(4-8)^2} = 4.12$	$\sqrt{(6-1)^2+(4-2)^2} = 5.38$
X_6 (7,5)	$\sqrt{(7-2)^2+(5-9)^2} = 6.40$	$\sqrt{(7-5)^2+(5-8)^2} = 3.60$	$\sqrt{(7-1)^2+(5-2)^2} = 6.70$
X_7 (1,2)	$\sqrt{(1-2)^2+(2-9)^2} = 7.07$	$\sqrt{(1-5)^2+(2-8)^2} = 7.21$	$\sqrt{(1-1)^2+(2-2)^2} = 0$
X_8 (4,9)	$\sqrt{(4-2)^2+(9-9)^2} = 2$	$\sqrt{(4-5)^2+(9-8)^2} = 1.41$	$\sqrt{(4-1)^2+(9-2)^2} = 7.61$

* Closest ones are highlighted.

New Centroids
$C_1 = (3,9)$
$C_2 = (6.25, 5.25)$
$C_3 = (1, 3.5)$

Iteration 2

	Distance from C1(3,9)	Distance from C2(6.25, 5.25)	Distance from C3(1, 3.5)
x_1 (2,9)	$\sqrt{(2-3)^2+(9-9)^2}=1$	$\sqrt{(2-6.25)^2+(9-5.25)^2}=5.73$	$\sqrt{(2-1)^2+(9-3.5)^2}=5.59$
x_2 (1,5)	$\sqrt{(1-3)^2+(5-9)^2}=4.4$	$\sqrt{(1-6.25)^2+(5-5.25)^2}=5.81$	$\sqrt{(1-1)^2+(5-3.5)^2}=1.5$
x_3 (7,4)	$\sqrt{(7-3)^2+(4-9)^2}=6.40$	$\sqrt{(7-6.25)^2+(4-5.25)^2}=1.45$	$\sqrt{(7-1)^2+(4-3.5)^2}=6.02$
x_4 (5,8)	$\sqrt{(5-3)^2+(8-9)^2}=2.23$	$\sqrt{(5-6.25)^2+(8-5.25)^2}=3.02$	$\sqrt{(5-1)^2+(8-3.5)^2}=6.02$
x_5 (6,4)	$\sqrt{(6-3)^2+(4-9)^2}=5.83$	$\sqrt{(6-6.25)^2+(4-5.25)^2}=1.27$	$\sqrt{(6-1)^2+(4-3.5)^2}=5.02$
x_6 (7,5)	$\sqrt{(7-3)^2+(5-9)^2}=5.65$	$\sqrt{(7-6.25)^2+(5-5.25)^2}=0.79$	$\sqrt{(7-1)^2+(5-3.5)^2}=6.18$
x_7 (1,2)	$\sqrt{(1-3)^2+(2-9)^2}=7.28$	$\sqrt{(1-6.25)^2+(2-5.25)^2}=6.17$	$\sqrt{(1-1)^2+(2-3.5)^2}=1.5$
x_8 (4,9)	$\sqrt{(4-3)^2+(9-9)^2}=1$	$\sqrt{(4-6.25)^2+(9-5.25)^2}=4.37$	$\sqrt{(4-1)^2+(9-3.5)^2}=6.26$

New Centroids:

$C_1 = (3,9)$ $C_2 = (6.25, 5.25)$ $C_3 = (1, 3.5)$

Iteration 3 is as same as Iteration 2

Therefore above centroids are required output.

Advantages:

- It is very easy
- It is done in polynomial time.

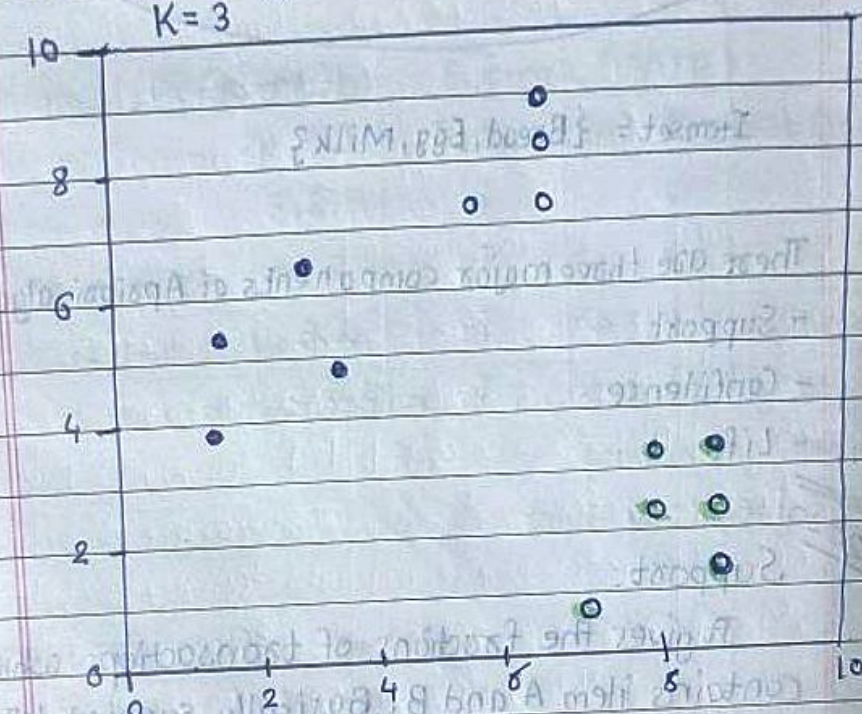
Disadvantages:

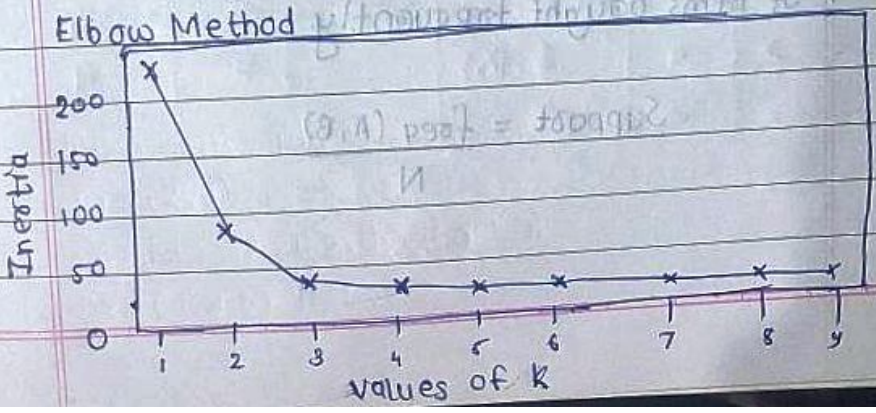
- sensitive to outliers, an object with extremely large value may substantially distort the distribution of data.
- K is not known in advance.
- Cannot handle categorical data.

Determining K in K-means:

- Deceiding K in K-means is difficult as data is large and unlabelled.
- Distortion:- It is calculated as an average of the squared distances from the cluster centers of the respective clusters. Typically, the Euclidean distance metric is used.
- Inertia:- It is the sum of squared distances of samples to their closest cluster center

Scatter plot of given data set

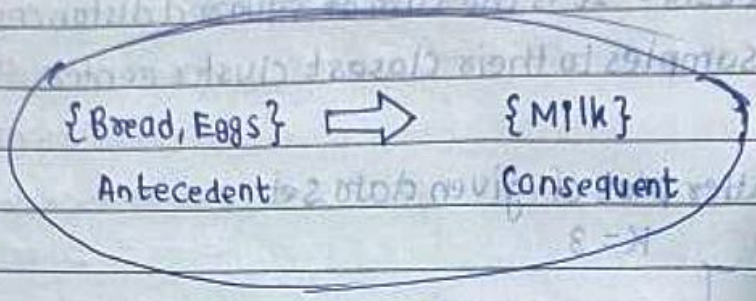




Association rule:

- Association rules analysis is a technique to uncover how items are associated to each other.
- The process of identifying an associations between products is called as association rule mining.

Apriori :-



Itemset = {Bread, Egg, Milk}

There are three major components of Apriori algorithm:

- Support
- Confidence
- Lift

01/20[20]

Support:

It gives the fractions of transactions which contains item A and B. Basically support tells us about the frequently bought items or the combination of items bought frequently.

$$\text{Support} = \frac{\text{freq}(A, B)}{N}$$

Confidence:

It tells us how often the items A and B occurs together, given the number of times A occurs.

$$\text{Confidence} = \frac{\text{freq}(A, B)}{\text{freq}(A)}$$

Lift:

Lift indicates the strength of a rule over the random occurrance of A and B. It is basically tells us the strength of any rule.

$$\text{Lift}(A \rightarrow B) = \frac{\text{Support}(A \cup B)}{\text{Support}(A) \cdot \text{Support}(B)}$$

Support
↳ Freq. of an item in given set of transaction

Confidence
↳ It is conditional probability of purchasing item B given A is already purchased.

Lift
↳ It is same as confidence but also considers the popularity of item B also
Lift (A → B) = Conf (A and B) / supp (B)

Apriori:-
- Set a minimum value for support and confidence.
- Extract all the subsets having higher value of support than minimum threshold.
- Select all rules from the subsets with confidence value higher than minimum threshold.
- Order the rules by descending order of Lift.

Association Rule Mining:-
Ex:-
Given:- Support = 2 Confidence = 60%

Sr. No	Tid	Items
1.	T_1	1, 3, 4
2	T_2	2, 3, 5
3	T_3	1, 2, 3, 5
4	T_4	2, 5
5	T_5	1, 3, 5

- Frequency Count

Item	Frequency
1	3
2	3
3	4
4	1 (X)
5	4

As Support=2 →

⟹

Item	Frequency
1	3
2	3
3	4
5	4

- Two set Items

Item set	Frequency
1,2	1 (X)
1,3	3
1,5	2
2,3	2
2,5	3
3,5	3

s, support=2

⟹

Item set	Frequency
1,3	3
1,5	2
2,3	2
2,5	3
3,5	3

- Three set Items

Item Set	Frequency
1,3,5	2
1,2,3	1
2,3,5	2

⟹

Item Set	Frequency
1,3,5	2
2,3,5	2

- Setting Rules:

① 1,3,5

↳ 1^3 → 5

Confidence = Support (1,3,5) / Support (1,3)
= 2/3
= 66.66%

↳ 3^5 → 1

Confidence = Support (1,3,5) / Support (1,3)
= 2/3
= 66.66%

↳ 1 → 3^5

Confidence = Support (1,3,5) / Support (1,3)
= 2/3
= 66.66%

↳ 3 → 1^5

Confidence = Support (1, 3, 5) / Support (3)
= 2/4
= 50% → (Rejected)

↳ 5 → 1^3

Confidence = Support (1, 3, 5) / Support (5)
= 2/4
= 50% → (Rejected)

② 2, 3, 5

↳ 2^3 → 5

Confidence = Support (2, 3, 5) / Support (2, 3)
= 2/2
= 100%

↳ 2^5 → 3

Confidence = Support (2, 3, 5) / Support (2, 5)
= 2/3
= 66.66%

↳ 3^5 → 2

Confidence = Support (2, 3, 5) / Support (3, 5)
= 2/3
= 66.66%

↳ 2 → 3^5

Confidence = Support (2, 3, 5) / Support (2)
= 2/3
= 66.66%

↳ 3 → 2^5

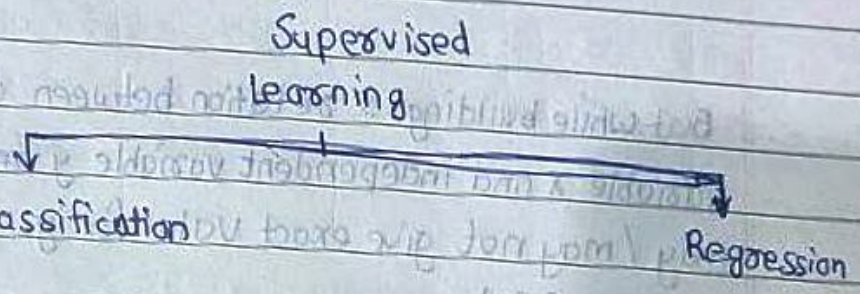
Confidence = Support (2,3,5) / Support (3)
= 2/4
= 50% (Rejected)

↳ 5 → 2^3

Confidence = Support (2,3,5) / Support (5)
= 2/4
= 50% (Rejected)

18/01/2021

Regression:



Regression is used to predict real value or it can also be termed as Estimators function. Regression is used to predict Stock Price, Weather forecast and home prices in a particular area.

Regression analysis:-

A set of statistical processes for estimating the relationships between a dependent variable (outcome variable) and one or more independent variables (predictors). The most common form of regression analysis is linear regression.

- A function estimator, regressor, is mapping $f: X \rightarrow Y$. The regression learning problem is to learn a function estimator from example $(x_i, f(x_i))$. For instance, we might want to learn an estimator for the Dow Jones index, or the FTSE 100 based on selected economic indicators.

- In Regression, our aim is to show relationship between two variable (independent variables) X and Response variable (target/dependent variable) Y.

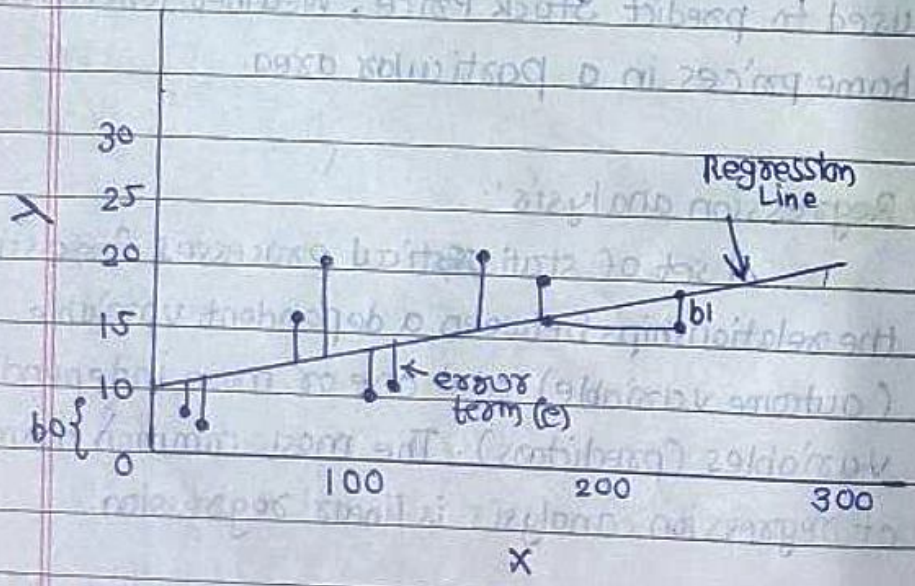
$$Y = f(x)$$

But while building a relation between dependent variable x and independent variable y the function may / may not give exact value for y.

$$Y = f(x)$$

$$\hat{Y} = f(x)$$

Residual error $= Y - \hat{Y}$



$\hat{y}$ → y predicted

①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩
x	y	x-mean	y-mean	$(x-mean)^2$	$\hat{y}$	$y-\hat{y}$	$(y-\hat{y})^2$	$(y-ymean)^2$	$(ymean-y)^2$
1	2	-2	-2	4	2.8	-0.8	0.64	4	1.44
2	4	-1	0	1	3.4	0.6	0.36	0	0.36
3	5	0	1	0	4	1	1	1	0
4	4	1	0	1	4.6	-0.6	0.36	0	0.36
5	5	2	1	4	5.2	-0.2	0.04	1	1.44
[3]	[4]			10			2.4	6	3.6

↑ x-mean ↑ y-mean

Eqn of line : $Y = mx + c$

$y = B_0 + B_1 x$

B_0 - Intercept

B_1 = Slope

B_1 = Sum of all (x-mean) × (y-mean) / sum of all $(x-mean)^2$

B_0 = Ymean - B_1 * Xmean

~~= (-2 x -2)~~

= (-2 x -2) + (-1 x 0) + (0 x 1) + (1 x 0) + (2 x 1) / 10

= 4 + 0 + 0 + 0 + 2 / 10

= 6 / 10

→ = 0.6

B_0 = 4 - ((0.6) * 3) = 2.2

$\hat{y} = 2.2 + 0.6x$

$SSE = \Sigma (y-\hat{y})^2 = 2.4$

$SSR = \Sigma (\hat{y} - ymean)^2 = 3.6$

$SST = SSR + SSE = 2.4 + 3.6 = 6$

$R^2 = SSR / SST = 3.6 / 6 = 0.6$

Implementation using python:

[1]
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

[2]
```
data = pd.DataFrame({'x': [1,2,3,4,5], 'y': [2,4,5,4,5]})
```

[3]
```
print(data)
```

	x	y
0	1	2
1	2	4
2	3	5
3	4	4
4	5	5

[4]
```
x_mean = np.mean(data['x'])
y_mean = np.mean(data['y'])
n = len(data)
```

[5]
```
print(x_mean)
```
3.0

[6]
```
print(y_mean)
```
4.0

[7]
```
x = data['x'].values
y = data['y'].values
```

[8]
```
print(x)
```
[1,2,3,4,5]

[9] print [y]
[2, 4, 5, 4, 5]

[10]
```
num = 0
den = 0
for i in range (n):
    num += ((X[i] - X_mean) * (y[i] - y_mean))
    den += ((X[i] - x_mean)) **2
m = num / den
c = y_mean - (m * x_mean)
```

[11]
```
X = np.linspace (1, 10, 2)
y = (m * X) + c
plt.plot (X, y, color = 'blue', label = 'Regression Line')
plt.scatter (X, Y, color = 'green', label = 'Scatter')
plt.xlabel ('X')
plt.ylabel ('Y')
plt.legend ()
plt.show ()
```
[plots]

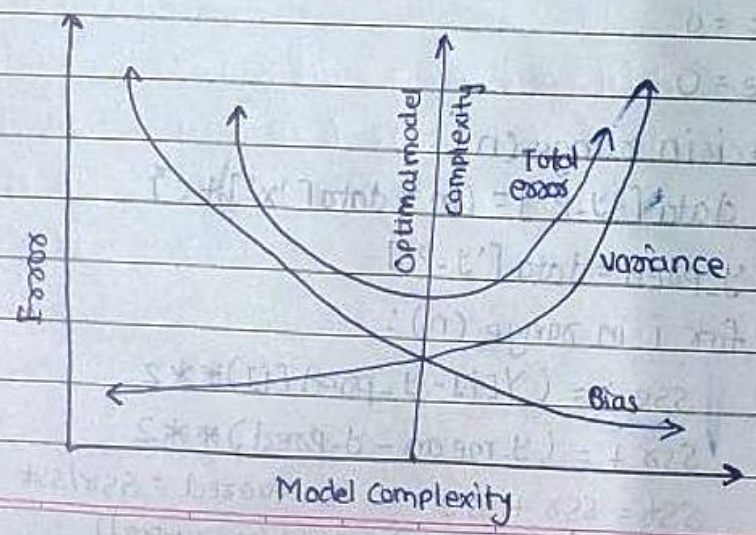
[12]
```
ssr = 0
sst = 0
sse = 0
for i in range (n):
    data ['y_P'] = (m * data ['x']) + c
    y_pred = data ['y_P']
    for i in range (n):
        sse += (Y[i] - y_pred [i]) **2
        ssr += (y_mean - y_pred) **2
        sst = ssr + sse
r_squared = ssr / sst
print (ssr, sse, sst)
print (r_squared)
```

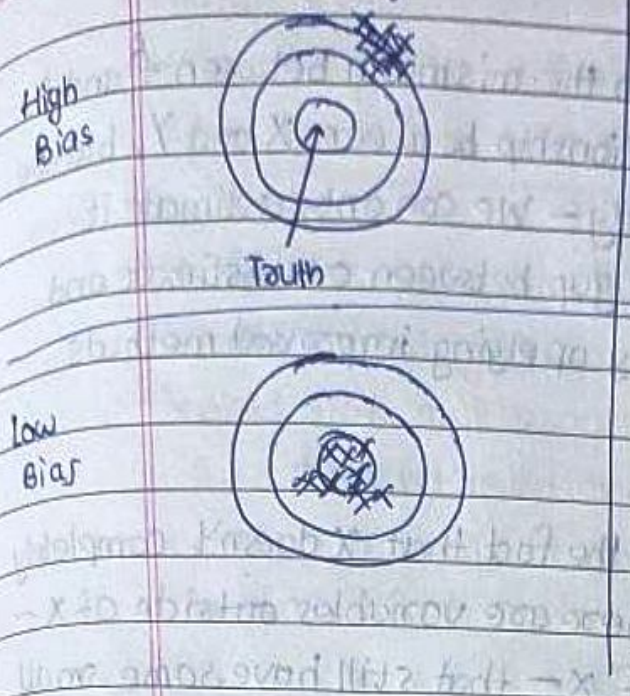
[13] Print data

	x	y	y_pred
0	1	2	2.8
1	2	4	3.4
2	3	5	4.0
3	4	4	4.6
4	5	5	5.2

[14]
```
plt.scatter(X,Y, color='green', label='scatter')
plt.plot(x,y, color='blue', label="Regression Line")
plt.axhline(y=y_mean, color='orange', linestyle='-')
plt.axvline(x=x_mean, color='orange', linestyle='-')
plt.plot([X[i],X[i]], [Y[i], y_pred[i]],
                    color='red', linestyle='-')
plt.xlabel('X')
plt.ylabel('Y')
plt.legend()
plt.show()
```

Bias-Variance Tradeoff :





If our model is too simple and has very few parameters then it may have high bias and low variance.

On the other hand, if our model has large numbers of parameters, then its going to have high variance and low bias.

So, we need to find the right / good balance without overfitting or underfitting data.

This tradeoff in complexity is where there is a tradeoff between bias and variance. An algorithm can't be more complex and less complex at the same time.

Errors:

$Y = f(x) + e$

e = Reducible error + Irreducible error

Reducible error:
- Error arising from the mismatch between $\hat{f}$ and f
- f is the true relationship between X and Y, but we can't see f directly — we can only estimate it.
- We can reduce the gap between our estimate and the true function by applying improved methods.

Irreducible error:
- Error arising from the fact that X doesn't completely determine Y. i.e there are variables outside of X — and independent of X — that still have some small effect on Y.
- The only way to improve prediction error related to irreducible error is to identify these outside influences and incorporate them as predictors.

Reducible error = Bias + Variance

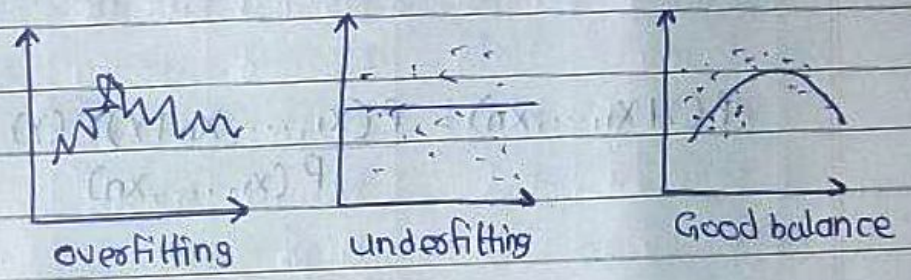
Bias:
- The difference between the average prediction of our model and the correct value which we are trying to predict.
- Model with high bias pays very little attention to the training data and oversimplifies the model.
- It always leads to high error on training and test data.

Variance:

- Variability of model prediction for a given data point or a value which tells us spread of our data.
- Model with high variance pays a lot of attention to the training data and does not generalize data which it hasn't seen before.
- As a result, such models perform very well on training data but has high error rates on test data.

Error in prediction:

$$Err(x) = E\left[(Y - \hat{f}(x))^2\right]$$



Underfitting:

- happens when a model is unable to capture the underlying pattern of the data.
- These models usually have ~~high~~ low variance and high bias.
- It happens when we have very less amount of data to build an accurate model.

Overfitting:

- happens when our model captures the noise along with the underlying pattern of data.
- It happens when we train our model a lot over noisy dataset. These models have low bias and high variance.
- These models are very complex like Decision trees which are prone to overfitting.

22/01/2020

Naïve Bayes:

- An probabilistic Approach which will identify the posterior probability of occurrence of an event

The Bayes Classifier:

$$P(Y|X_1, \ldots, X_n) = \frac{P(X_1, \ldots, X_n|Y)P(Y)}{P(X_1, \ldots, X_n)}$$