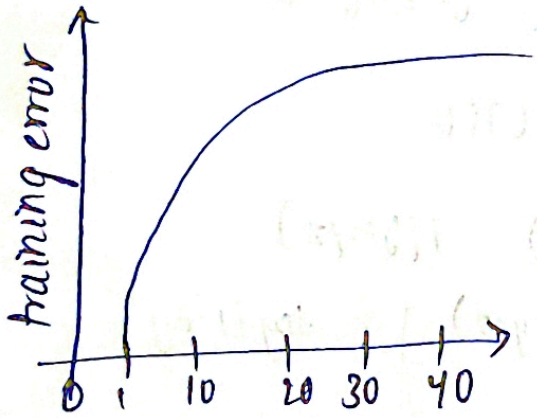
KNN.

(8-1)a For a given dataset; training error rate for $k=1$ is always zero. This is because closest point any training data point is itself. So prediction is always accurate with $k=1$

But increase in value of k, decision boundary becomes smoother with k+n then the point predicts belong to either of class depending on which majority.

The following graph shows training error variation with k varying from 1 to n



Training Error.

Scanned By Scanner Go

Q-1(b)

(i) If network performs well on the training set but generalizes badly, we can say it is overfitting. A network might overfit if the training set contains accidental regularities.

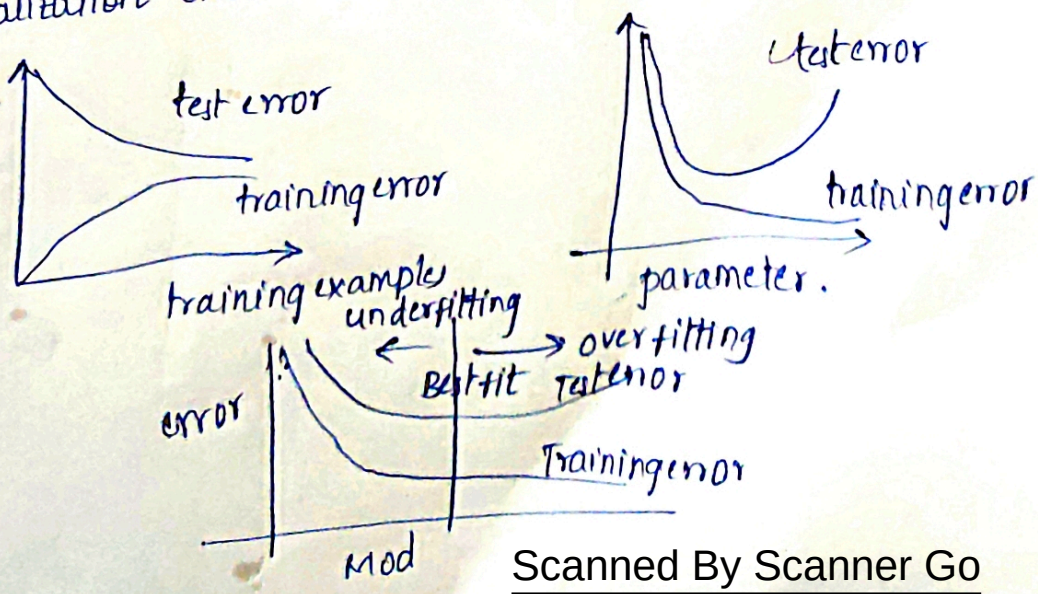
For instance we decided task to be classify handwritten digits it might happen that in the training set, all ~~integers~~ images of 9's have pixel no. 122 on, while all other examples have it off. The network might decide to exploit this accident regularity, thereby correctly classifying all the training examples of 9's, without learning the true regularities. If this property doesn't hold on the test set, the network will generalize badly.

Having more training data should only help generalization; for any particular test example, the larger the training set, the more likely there will be closely related training example. Also, the larger the training set, the fewer the accidental regularities, so the network will be forced to pick up the true regularities.

Generalization error ought to improve as we add more training examples.

If we add more parameters, it becomes easier to fit both the accidental & the true regularities of the training data.

Training error improve as we add more parameters. The effect on generalization error is a bit more subtle.



Scanned By Scanner Go

c. KNN works well with smaller no. of input variables, but struggles when the no. of inputs is very larg.

Each input variable can be considered a dimension of p-dimensional input space For example if you had 2 input variables x_1, x_2 the input space would be 2-dimensional.

As the no. of dimensions increases volume of input spaces increases at an exponential rate.

In high dimensions, points that may be similar may have very large distances. All points will be faraway from each other & our intuition for distances in simple 2 & 3 dimensions spaces 'break down. This is "curse of dimensionality.

d. No, the decision boundaries for 1-NN correspond to cell boundaries of each point & are not necessarily parallel to the co-ordinate axes, The decision tree boundaries would always be parallel to co-ordinate axes based on kinds of questions asked at each node of decision tree. So it is not possible build decision tree that classifies similar to 1NN using Euclidean distance.

Scanned By Scanner Go

2. Bayes classifier:

(a) Given training examples for class 1

$$C_1 = \{0.5, 0.1, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.35, 0.25\}$$

Training examples from class 2 =

$$C_2 = \{0.9, 0.8, 0.75, 1.0\}$$

classification probability as given by

Bayes theorem =

$$P(C_j/x) = \frac{P(x/c_i)\ P(C_j)}{\sum_{k=1}^{K} P(x/C_k)\ P(C_k)}$$

in problem $K = 1, 2$

In this case, using Gaussian likelihood $\rightarrow P(x/C_j) = \frac{1}{\sigma_j \sqrt{2\pi}} e^{-1/2\left(\frac{x-\mu_j}{\sigma_j}\right)^2}$

From problem Given.

$$\sigma_1^2 = 0.0149$$

$$\sigma_2^2 = 0.0092$$

$$\mu_1 = \frac{1}{N_1} \sum_{i=1}^{N_1} x_i \Rightarrow \frac{2.6}{10} = 0.26.$$

Similarly:

$$\mu_2 = \frac{1}{N_2} \sum_{i=1}^{N_2} x_i$$

$$= 0.8625$$

Also for calculating probability of class 1 $\rightarrow P_1 = \frac{N_k}{\sum_{k=1} N_k}$

$$P_1 = \frac{N_1}{N_1 + N_2} = \frac{10}{10+4} = \frac{10}{14} = 0.714.$$

$$P_2 = N_2/N_1 + N_2 = 4/14 = 0.2857$$

Now, to predict test point using Bayes theorem by fitting Gaussian model to likelihood $x = 0.6$;

$$P(C_j/0.6) = \frac{P(0.6/C_j)\ P(C_j)}{\sum_{k=1}^{K} P(0.6/C_j)\ P(C_j)}$$

Now $P(0.6/C_1) = \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-1/2\left(\frac{x-\mu_j}{\sigma_j^2}\right)^2}$

$$= \frac{0.000}{\sqrt{2\pi \times 0.0149}} = 0.0675$$

Scanned By Scanner Go

$P(0.6/c_2) = \frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-1/2\left(\frac{0.6-\mu_2}{\sigma_2}\right)^2} = \frac{0.0236}{0.240} = 0.0981$

$P(c_1/0.6) = \frac{0.0675 \times 0.714}{(0.0675 \times 0.714) + (0.0981 \times 0.285)} = T.$

$P(c_1/0.6) = 0.632.$

Ily for $P(c_2/0.6) = \frac{0.0981 \times 0.285}{T}$

$= 0.367.$

$\therefore P(c_1/0.6) = 0.632$ & $P(c_2/0.6) = 0.367$

Conclusion:-

From above equation & answer acquired it can be concluded that test point $x = 0.6$ belong to class 1

Naive Bayes - Test classifier:-

b) Attempt to classify documents as either sports or politics.

Given $x = \{$Goal, football, golf, defence, offence, wicket, office, strategy$\}$

$x_{politics} = P$

$$\begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

$x_{sport} = (S)$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

To find if document $x = (1, 0, 0, 1, 1, 1, 1, 0)$ is about politics?

Considering; $y = 1 \rightarrow$ politics

$y = 0 \rightarrow$ sports

Scanned By Scanner Go

Ans. Let x_j be presence of jth word among 8 attributes/words. 'y' be classification in politics or sports.

So, $p(y) \approx \phi_y$

$\forall_j \quad p(x_j=1/y=0) \approx \phi_{j/y=0}$

$\forall_j \quad p(x_j=1/y=1) \approx \phi_{j/y=1}$

⇒ Given from politics & sport.

matrices; $m = 6+6 = 12$

n = No. of columns = 8.

⇒ $\phi_y = \frac{\text{No. of rows of (P)}}{\text{No. of rows of (P)} + \text{No. of rows of S.}}$

$= 6/6+6 = 0.5$

So $\boxed{\phi_y = 0.5}$

$\phi_{j/y=1} = \frac{\sum I\{y^i=1, x_j^i=1\}}{\text{No. of rows of (P)}}$

$\therefore \phi_{j/y=1} \Rightarrow [0.33, 0.166, 0.166, 0.833, 0.83, 0.166, 0.66, 0.83]$

Similarly calculating for.

$\phi_{j/y=0} \Rightarrow$

$[0.66, 0.66, 0.166, 0.66, 0.166, 0.166, 0, 0.166]$

Now to predict,

$\arg\max_y P(y/\vec{x}) = \arg\max \frac{P(\vec{x}/y)\,P(y)}{P(\vec{x})}$.

$P(y=1/\vec{x}) = P(\vec{x}/y=1)\,P(y=1)$

$= \prod_{i=1}^{R} \phi_{j/y=1}^{x_j} (1-\phi_{j/y=1})^{1-x_j^i} * \phi_y$

$P(y=0/\vec{x}) = P(\vec{x}/y=0)\,P(y=0)$

$= \prod_{i=1}^{8} \phi_{j/y=0}^{x_j} (1-\phi_{j/y=0})^{1-x_j^i} (1-\phi_y)$

Writing in logs:

$\log P(y=1/\vec{x}) = \sum_{j=1}^{8} x_j^i \log(\phi_{j/y=1}) + (1-x_j) \log(1-\phi_{j/y=1}) + \log \phi_y$

Scanned By Scanner Go

$\log P(y=0/\vec{x})$

$$= \sum_{j=1}^{8} (x_j \log \phi_{j/y=0}) + (1-x_j)\log(1-\phi_{j/y=0}) + \log(1-\phi_y).$$

So Now Given document to be predicted = (1, 0, 0, 1, 1, 1, 1, 0)

Now calculating ! $\log [P(y=1/x))$

j	$x_j \log(\phi_{j/y=1})$	$(1-x_j) \times \log(1-\phi_{j/y=1})$	$\log(\phi_y)$
1	0.401	0	
2	0	-0.0788	
3	0	-0.0788	
4	-0.0793	0	
5	-0.0793	0	
6	-0.7798	0	
7	-0.1804	0	
8	0	-0.777	
Total	=1.1798	-0.9348	-0.301

So $\log P(y=1/x)) = -1.5998 - 0.9348 - 0.301.$

$\log P(y=1/x)) = -2.8356.$

Now calculating $x \rightarrow (1, 0, 0, 1, 1, 1, 0)$ $P(y=0/x)$

j	$x_j \log \phi_{j/y=0}$	$(1-x_j) \times \log(1-\phi_{j/y=0})$	$\log(1-\phi_y)$
1.	-0.1804	0	
2.	0	-0.1804	
3.	0	-0.7798	-0.301
4.	-0.1804	0	
5.	-0.7798	0	
6.	-0.7798	0	
7.	0	0	
8.	0	-0.7798	
Total	-1.9204	-1.74	-0.301

$\log P(y=0/x) = -3.9614$

Scanned By Scanner Go

Conclusions:-

From $\log P(y=1/x) = -2.8356$ d.

$\log P(y=0/x) = -3.9814$

It can be concluded that given document $x = (1,0,0,1,1,1,1,0)$ belong to politics as

$\log P(y=1/x) > \log P(y=0/x)$

Scanned By Scanner Go