Assignment-2 :-

Q-4:- Proofs :-

① Multinomial naive Bayes $\longrightarrow$ is a linear classifier

→ We know that,
The likelihood of observing x provided class C_K ~~Multinomial naive bayes classifier~~ is given by

$$P(x|C_k) = \frac{(\sum_i x_i)!}{\prod_i x_i!} \prod_i p_{ki}^{x_i} \quad \text{———} ①$$

Where, P_i = Probability that event i occurs
$\mathbf{x}$ = A feature vector $(x_1, \dots, x_n)$
C_k = class k

~~→ Now, if we express ① in log-space,~~

→ Multinomial naive Bayes classifier,

$$P(C_k|x) = P(C_k) \prod_{i=1}^{n} p_{ki}^{x_i} \quad \text{———} ②$$

→ Expressing ① in log-space,

$$\log(P(C_k|x)) \propto \log\left(P(C_k) \prod_{i=1}^{n} p_{ki}^{x_i}\right)$$

$$= \log(P(C_k)) + \sum_{i=1}^{n} x_i \log p_{ki}$$

$$= \boxed{b + W_k^T x} \quad \text{———} ③$$

In —(3),

$b = \log(p(c_k))$ and

$w_{ki} = \log(p_{ki})$

→ As, we can see, —(3) is a linear equation in log-space.

→ Hence, proved

Q-4:-

(2) Logistic regression → linear classifier

→ We know that decision boundary for logistic regression is sigmoid function given by,

$$\boxed{\frac{1}{1+e^{-\theta^T x}} = 0.5} \quad \text{—— (1)}$$

$$\therefore \quad 1 = 0.5 + 0.5\left(e^{-\theta^T \cdot x}\right)$$

$$\therefore \quad \frac{0.5}{0.5} = e^{-\theta^T \cdot x}$$

$$\therefore \quad \boxed{e^{-\theta^T \cdot x} = 1} \quad \text{—— (2)}$$

→ Taking natural log on both sides of —(2)

$$\ln\left(e^{-\theta^T \cdot x}\right) = \ln(1)$$

$$\therefore \quad -\theta^T \cdot x = 0$$

$$\therefore \quad \theta^T \cdot x = 0$$

$$\therefore \quad \boxed{\sum_{i=0}^{n} \theta_i x_i = 0} \quad \text{—— (3)}$$

→ As we proved, for logistic regression decision boundary translates to linear boundary.

→ Hence, required.

2019/3/2 18:04

Q1: Steps:

1. First get data in dataframes. We need convert labels to numeric values so that we can evaluate after predictions.
2. To perform Linear Discriminant Analysis(LDA), I am using Fisher's LDA method.
3. For fisher's LDA, first we need to calculate within class scatter matrix and between class scatter matrix for both given classes.
4. get_within_class_scatter_matrix() and get_between_class_scatter_matrix() functions do that.
5. Next, we need to solve standard eigen value problem and get all eigen values and eigen vectors.
6. From eigen value-eigen vector pairs we can see that top 2 values retain maximum variance. Hence, using those top 2 value-vector pairs and translating 4dimensional data to 2 dimensional data.
7. After calculating weights(W), we can perform "X dot W", which will give required predictions