U 8941-5490

Ans to the Que No 1(a)

(1)

Quiz-4

Date : ....

Ans to Que 1 (a)

Posterior-ratio :

$$\frac{P(C_n=1|x_n)}{P(C_n=2|x_n)} \Rightarrow \boxed{\frac{P(x_n|C_n=1)\,P(C_1)}{P(x_n|C_n=2)\,P(C_2)} \underset{C_2}{\overset{C_1}{\gtrless}} 1}$$

if the posterior-ratio is > 1, then C_1

if the ratio is < 1, then C_2

$\Rightarrow$ threshold

$$\frac{P(x_n|C_n=1)}{P(x_n|C_n=2)} \underset{C_2}{\overset{C_1}{\gtrless}} \frac{P(C_2)}{P(C_1)}\cdot 1 \Rightarrow t$$

$$\Rightarrow \log P(x_n|C_n=1) - \log P(x_n|C_n=2) \underset{C_2}{\overset{C_1}{\gtrless}} \log t$$

$$\Rightarrow \log \mathcal{N}(x_n|\mu_1, \Sigma_1) - \log \mathcal{N}(x_n|\mu_2, \Sigma_2) \underset{C_2}{\overset{C_1}{\gtrless}} \log t$$

(i)

2

Date :

$$\log \mathcal{N}(x \mid \mu, \Sigma) = \log \left[\frac{\exp\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\}}{(2\pi)^{d/2} |\Sigma|^{1/2}} \right]$$

$$= -\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu) - \frac{d}{2} \log 2\pi + \frac{1}{2} \log |\Sigma^{-1}|$$

⇓

log of a general multi-variate Gaussian PDF

So, we can re-write eqⁿ (i) as

$$-\frac{1}{2}(x_n - \mu_1)^T \Sigma_1^{-1} (x_n - \mu_1) - \frac{d}{2}\log 2\pi + \frac{1}{2}\log|\Sigma_1^{-1}| + \frac{1}{2}(x_n - \mu_2)^T \Sigma_2^{-1}(x_n - \mu_2) + \frac{d}{2}\log 2\pi - \frac{1}{2}\log|\Sigma_2^{-1}| \underset{c_2}{\overset{c_1}{\gtrless}} \cancel{\log} +$$

~~From here on we will just work with the L.H.S of the above eqⁿ.~~

3

Date :

Again, $-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)$

$$= -\frac{1}{2}\left[x^T \Sigma^{-1} x - x^T \Sigma^{-1} \mu - \mu^T \Sigma^{-1} x + \mu^T \Sigma^{-1} \mu\right]$$

~~Accordingly we can rewrite +~~

$$= -\frac{1}{2}\left[x^T \Sigma^{-1} x - 2x^T \cdot \Sigma^{-1} \mu + \mu^T \Sigma^{-1} \mu\right]$$

So we can rewrite the eqn of the previous page as :

$$\frac{1}{2}\left\{ x_n^T (\Sigma_2^{-1} - \Sigma_1^{-1}) x_n - 2x^T (\Sigma_2^{-1}\mu_2 - \Sigma_1^{-1}\mu_1) + \text{const} \right\} \underset{c_2}{\overset{c_1}{\gtrless}} \log t$$

$$\Rightarrow y(x) = x^T A x + b^T x \underset{c_2}{\overset{c_1}{\gtrless}} 2 \log t$$

if $y > t$ then c_1.

o/w c_2.

4

Date :

So, $A = (\Sigma_2^{-1} - \Sigma_1^{-1})$

$b = (\Sigma_2^{-1}\mu_2 - \Sigma_1^{-1}\mu_1)$

$$t = \frac{P(c_2)}{P(c_1)}$$

* we can ignore the const. as it is same for every data instance X_n.

Ans to the Que No 1(b)

5

Date :

Ans to the Que 1(b)

LDA from QDA :

Assuming, $\Sigma_1 = \Sigma_2 = \Sigma$ that is the covariance matrix of the two classes are same. Which means the ~~Bol~~ two curves have the same ~~varianc~~ variance. The contours of the two gaussian curve on 2d space has the same shape.

Then, $A = (\Sigma_2^{-1} - \Sigma_1^{-1}) = (\Sigma^{-1} - \Sigma^{-1})$

$= 0$

So ~~the class~~ QDA becomes,

$$y(x) = b^T X = X^T b$$

(6)

Date :..............................

where, $b = \Sigma^{-1}(\mu_2 - \mu_1) = \omega$

so, $-2x^T\Sigma^{-1}(\mu_2 - \mu_1) \underset{c_2}{\overset{c_1}{\gtrless}} 2\log +$

$\Rightarrow -2\cancel{x}^T\Sigma^{-1}(\mu_2 - \mu_1) \underset{c_2}{\overset{c_1}{\gtrless}} 2\log\frac{P(c_2)}{P(c_1)}$

$\Rightarrow x^T\Sigma^{-1}(\mu_2 - \mu_1) \underset{c_2}{\overset{c_1}{\lessgtr}} -\log\frac{P(c_2)}{P(c_1)}$

$\Rightarrow x^T\cancel{\beta}\Sigma^{-1}(\mu_2 - \mu_1) \underset{c_2}{\overset{c_1}{\lessgtr}} \log\frac{P(c_1)}{P(c_2)}$

$\therefore \cancel{h} = \frac{P(c_1)}{P(c_2)}$

$b = \Sigma^{-1}(\mu_2 - \mu_1)$

Ans to the Que No 1(c)

7

Date :..............................

Ans to the Que 1(c)

If it is assumed that the covariance matrix is diagonal then from QDA we can obtain Gaussian Naive Bayess classifier. That means, ~~Every~~ the features are independent of eachother. In that case we can assume a univariate model.

$$X_{nd} \sim \mathcal{N}(\mu_d, \sigma_d^2)$$

$$P(X_n | C_n) = \prod_{i=1}^{d} P(X_{ni} | C_n)$$

d = dimensionality of the features.

$X_{ni} \rightarrow$ Refers to the i'th feature of the n'th data instance.

Extra credit:

8

Date : ....

$$\frac{P(c_n=1 \mid x_n)}{P(c_n=2 \mid x_n)} = \frac{\prod_{i=1}^{d} P(x_{ni} \mid c_n=1)\, P(c_n=1)}{\prod_{i=1}^{d} P(x_{ni} \mid c_n=2) \cdot P(c_n=2)}$$

$$\Rightarrow \frac{\log (c_n=1 \mid x_n)}{\log (c_n=2 \mid x_n)} = \frac{\log \prod_{i=1}^{d} P(x_{ni} \mid c_n=1)\, P(c_n=1)}{\log \prod_{i=1}^{d} P(x_{ni} \mid c_n=2)\, P(c_n=2)}$$

$$\Rightarrow \frac{\sum_{i=1}^{d} \log P(x_{ni} \mid c_n=1)}{\sum_{i=1}^{d} \log P(x_{ni} \mid c_n=2)} \underset{c_2}{\overset{c_1}{\gtrless}} \log \frac{P(c_2)}{P(c_1)} \Rightarrow t$$

$$\Rightarrow \cancel{\sum_{i=1}^{d} \log \mathcal{N}(x_{ni} \mid \mu_i^1, \sigma_i^1) - \sum_{i=1}^{d} \log \mathcal{N}(x_{ni} \mid \mu_i^2, \sigma_i^2)}$$

$$\cancel{\underset{c_2}{\overset{c_1}{\gtrless}} \log t}$$

i

$$\cancel{\mathcal{N}(\mu^1, \sigma^1)} \Rightarrow c_1 \qquad \cancel{\mathcal{N}(\mu^2, \sigma^2)} \Rightarrow c_2$$

9

Date :

$$\Rightarrow \sum_{i=1}^{d} \log \mathcal{N}(x_{ni} \mid {}^{1}\mu_i, \sigma^2) - \sum_{i=1}^{d} \log \mathcal{N}(x_{ni} \mid {}^{2}\mu_i, \sigma^2) \underset{c_2}{\overset{c_1}{\gtrless}} \log t \quad (i)$$

$$\mathcal{N}(x \mid \mu_1, \sigma^2) \Rightarrow c_1,$$

$$\mathcal{N}(x \mid \mu_2, \sigma^2) \Rightarrow c_2$$

$$\boxed{\log \mathcal{N}(x \mid \mu, \sigma^2) = -\frac{1}{2\sigma^2}(x_n - \mu)^2 - \frac{1}{2}\ln \sigma^2 - \frac{1}{2}\ln 2\pi}$$

So we can rewrite eqn. (i) as.

$$\sum_{i=1}^{d}\left[-\frac{1}{2\sigma_i^2}(x_{ni} - {}^{1}\mu_i)^2 - \frac{1}{2}\ln \sigma_i^2 - \frac{1}{2}\ln 2\pi\right]$$

$$- \sum_{i=1}^{d}\left[-\frac{1}{2\sigma_i^2}(x_{ni} - {}^{2}\mu_i)^2 - \frac{1}{2}\ln \sigma_i^2 - \frac{1}{2}\ln 2\pi\right] \underset{c_2}{\overset{c_1}{\gtrless}} \log t$$

10

Date :

$$\Rightarrow \sum_{i=1}^{d} \frac{1}{2}\Big[\cancel{\frac{1}{\sigma_i^2} x_{ni}^2} - \frac{1}{\sigma_i^2} 2 x_{ni}\, {}^{1}\mu_i + {}^{1}\mu_i^2 \frac{1}{\sigma_i^2} - \cancel{\frac{1}{\sigma_i^2} x_{ni}^2} + 2 x_{ni}\, {}^{2}\mu_i \frac{1}{\sigma_i^2} - \frac{1}{\sigma_i^2}\, {}^{2}\mu_i^2 \Big] \underset{c_2}{\overset{c_1}{\gtrless}} \log t$$

$$\Rightarrow \sum_{i=1}^{d} \frac{1}{2\sigma_i^2} \left[2 x_{ni} ({}^{2}\mu_i - {}^{1}\mu_i) + ({}^{1}\mu_i^2 - {}^{2}\mu_i^2) \right] \gtrless \log t$$

$$\Rightarrow \underbrace{\sum_{i=1}^{d} \frac{1}{\sigma_i^2} x_{ni} ({}^{2}\mu_i - {}^{1}\mu_i)}_{w^T \bar{x}} + \underbrace{\sum_{i=1}^{d} \frac{1}{2\sigma_i^2} ({}^{1}\mu_i^2 - {}^{2}\mu_i^2)}_{w_0} \gtrless \log t$$

$$\Rightarrow \boxed{w^T \bar{x}_n + w_0 \underset{c_2}{\overset{c_1}{\gtrless}} \log t}$$

$\hookrightarrow$ it is a linear classifier

Ans to the Que No 2

11

Ans to the Que No 2

Objective :

→ Maximize the margin between Decision boundary and data points.

→ It is a decision machine so no posterior probability.

→ $y(x) = w^T \underbrace{\phi(x)}_{\text{non-linear function}} + b$

→ Main objective is to maximize the margin between $\boxed{\text{support vectors}}$ and decision boundary.

→ we want to find w and b that gives maximum margin.

Date :..............................

Kernel trick :

→ $\boxed{f(x) = w^T \phi(x) + b}$ ⇒ it is a linear function. But in order to solve the non-linear problems we have $\phi(x)$. Which is a non-linear feature extranction function. That means using ϕ we are transforming / mapping data from lower to high dimension. On that higher dimensional feature space the ~~data become~~ problem becomes linear.

→ We don't need to know the exact ϕ. Using kernel trick we can achieve this non-linearity.

→ Using the kernel trick SVM avoids solving/optimizing for learning the best ϕ function.

13

Date :

Training :

→ During training ~~SVM~~ procedure SVM considers a particular ϕ choice.

→ So at the begining SVM is restricted to a ϕ choice.

→ We wan't to minimize classification error. ⇒ Maximize # of correctly classified points.

$$\Rightarrow \sum \max \frac{t_n\, y(x_n)}{\|w\|} = \max \frac{t_n\,(w^T\phi(x_n)+b)}{\|w\|}$$

$$\boxed{y(x_n) = w^T\phi(x_n)+b}$$

for correctly classified points

$$\boxed{t_n\, y(x_n) > 0}$$

where, $t_n \begin{cases} +1 \\ -1 \end{cases}$

class label

Date :..............................

→ So our target is to find the w and b that maximizes the ~~bou~~ margin for the support vector. As our target in to find a uniq sol^n for $\boxed{w}$ we have norm$(w) \Rightarrow \|w\|$ in the denominator.

$$\max \frac{t_n\, y(x_n)}{\|w\|} = \max$$

minimum margin is 1

→ So, we now fix $\boxed{t_n\, y(x_n) \geq 1}$ and will use lagrange multiplier to optimize for w and b. So minimize the denominator which is $\|w\|$.

→ using lagrange multiplier we convert this constrained problem to an unconstrained problem.

15

Date :

$$w = \sum_{n=1}^{N} \alpha_n t_n \phi(x_n)$$

lagrange mult.

* now we search ~~function~~ for α_n.

$$\rightarrow y(x_n) = \sum_{n=1}^{N} w^T \phi(x_n) + b$$

$$= \sum_{n=1}^{N} \alpha_n t_n \phi(x_n)^T \phi(x_n) + b$$

kernel trick

$\boxed{\text{Kernel function}}$

$N \Rightarrow$ # number of training samples.

→ In the begining we ~~mat~~ started with the S.V only. Later we see that we are including all the training samples (including S.V.) while considering the Loss function later we will see that eventually only the S.V ~~make~~ ~~has the~~ contributes to maximizing the margin. And also we don't always know what are the S.V beforehand.

16

Testing :

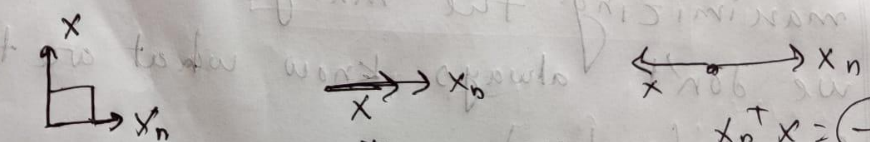
→ $y(x) = \sum_{n=1}^{N} \alpha_n \, t_n \, k(x, x_n)$

test point

training point

→ So to decide for a test poin x we take the inner-product (similarity) betw^n x and all the training points x_n.

→ So the similarity score should be high if x and x_n is similar. So the kernel function works as a weighting factor for the train data label t_n ~~If they are similar~~



$x_n^T x = 0$ $\quad$ $x_n^T x = 1 \to$ max $\quad$ $x_n^T x = -1 \to$ min.

17

→ If x and x_n are similar we count that label t_n in decision making

→ if $\boxed{x_n^T x \to 0}$ then t_n has no contribution

→ if $\boxed{x_n^T x = -1}$ then t_n contributes negatively in decision making.

Loss function:

→ $E(y_n, t_n) = \max\{1 - y_n t_n, 0\}$

$y_n t_n \geq 1$ for every ~~supt~~ training point other than the support vectors so. the error for them is 0.

But the error for support vectors would be non-zero as. $y_n t_n < 1$

18

Date :

for S.V. That means we want to maximize the margin for the S.V The margin is already maximized for other trainig samples.

Ans to the Que No 3

Boosting:

- Instead of using only one model. Data is trained on multiple models.
- Each new model is trained to emphasize the data that was incorrectly classified by the previous model.
- Uses some loss functions to measure the performance of a model.
- Size of trees is a parameter that needs to be tuned.

Bagging:

- Each model has equal weight in voting.
- But each model is trained on a randomly sampled subset of the original train data. This is done to avoid overfitting to train data and to promote model variance. Example: Random Forest => Feature Bagging as RF selects a random subset of features within the random subset of the dataset.
- Number of trees to be built is a parameter that needs tuning.
  Normally uses out of bag error or any loss function to measure the optimal value of number of trees to be built.
- Requires mechanism to de-correlate the trees that are built => bootstrap sampling is one such mechanism.