Q1 Numerical Check of the least square solution (10 points)

A1 Solution in the attached ".py" file.

Q2 Smokers (10 points)

A2 a) Let f be the fraction of women who do not smoke but get cancer.

⇒ Hence fraction smoking women who get lung cancer is 13 times.

Let A = Woman who smokes

B = women gets lung cancer

$$\Rightarrow P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

$$= \frac{P\left(\text{women gets lung cancer} \,/\, \text{who smokes}\right) P(\text{women who smokes})}{P(\text{Women gets lung cancer})}$$

$P(B) = 0.15 \times 13f + (1-0.15)f$

$P(B) = 2.8f$

$$\Rightarrow \boxed{P(A|B) = \frac{0.15 \times 13f}{2.8f} = 67\%}$$

(b) Given 18% of total adult population who smokes out of which 0.5 are women & 15% of that will be fraction of women smokers. Let $x$ be total population who smokes.

Hence $f = \dfrac{\frac{x}{2} \times 0.15}{0.18 \times x}$

$\Rightarrow f = \dfrac{5}{12}$

Q3 MLE of Poisson's distribution

A3 $P(x;\lambda) = \dfrac{\lambda^{x} e^{-\lambda}}{x!}$.

$\Rightarrow$ Likelihood of data for given parameter $\theta$ is

$$L(\theta) = \prod_{i=1}^{n} f(x_i/\theta)$$

$$MLE = \sum_{i=1}^{n} \log f(x_i/\theta)$$

$$L(\lambda) = \prod_{i=1}^{n} \frac{\lambda^{x_i} e^{-\lambda}}{x_i!}$$

Differentiating it w.r.t. $\lambda$ to find the log likelihood of $L(\lambda)$:

$$LL(\lambda) = \sum_{i=1}^{n} \log\left(\frac{\lambda^{x_i} e^{-\lambda}}{x_i!}\right)$$

$$LL(\lambda) = \sum_{i=1}^{n} \left[\log \lambda^{x_i} + \log e^{-\lambda} - \log x_i!\right]$$

$$LL(\lambda) = \sum_{i=1}^{n} \left[x_i \log \lambda - \lambda - \log(x_i!)\right]$$

$$LL(\lambda) = \log \lambda \sum_{i=1}^{n} x_i - n\lambda - \sum_{i=1}^{n} \log(x_i!)$$

$$\Rightarrow \frac{\delta(LL(\lambda))}{\delta(\lambda)} = 0$$

$$\Rightarrow \frac{1}{\lambda} \sum_{i=1}^{n} x_i - n - 0 = 0$$

$$\Rightarrow \boxed{\hat{\lambda} = \frac{1}{n} \sum_{i=1}^{n} x_i}$$

Q4 Generating Rand. Variable (10 points)

A4 rand(d, 1) gives $x$

$$x \in R^d$$

$$x \sim N(0, I)$$

if $x \sim N(\mu, \Sigma)$

$\Rightarrow Ax + b \sim N(A\mu + b, A\Sigma A^T)$

To transfer $N(0, I)$ to $N(\mu, \Sigma)$

$\mu = A(0) + b = b$

$\Sigma = AIA^T$

$$\boxed{\begin{aligned} \mu &= b \\ \Sigma &= AIA^T \\ Ax + b &\sim N(\mu, \Sigma) \end{aligned}}$$

Q5 Minimize loss fⁿ (10 points)

A5 $$WMSE = \min_{\theta} \frac{1}{n} \sum_{i=1}^{n} r_i (y_i - \phi_i^T \theta)^2$$

⇒ Let r be a matrix where $r_i$ are values at diagonal & zero for the rest

⇒ Let $\Psi$ be matrix of $\phi_i^T$ than WMSE can be written in matrix form as follows

$$WMSE = \frac{1}{n}(Y - \Psi\theta^T) R (Y - \Psi\theta)$$

$$WMSE = \frac{1}{n}\left(Y^T R Y - Y^T R \Psi\theta - \theta^T \Psi^T R Y + \theta^T \Psi^T R \Psi\theta\right)$$

⇒ Differentiating w.r.t $\theta$, we get the gradient as follows:

$$\nabla_\theta WMSE = \frac{2}{n}\left(-\Psi^T R Y + \Psi^T R \Psi \theta\right)$$

$$\Rightarrow \nabla_\theta WMSE = 0$$

$$\Rightarrow \boxed{\hat{\theta} = (\Psi^T R \Psi)^{-1} \Psi^T R Y}$$

Q6 Newsgroups dataset problem (50 points)

A6 All the codes can be found in the attached ".py" file.

a) Naive-Bayes using Bernoulli random variable

⇒ Test Accuracy:

28.409

⇒ Probability Distribution P is given by: $P(x|y) = \prod P(x_j/y)$.

where $x_j$ is the vector containing the information about the present word in the specified vocab. i.e. 1 if word in the document & vocab else 0.

$$\Rightarrow P(X) = P(X=x) = \begin{cases} q = 1-p & x=0 \\ p & x=1 \end{cases}$$

and the Bernoulli Naive Bayes Classifier

$$P(x_j/y) = P(j/y)\, x_j + (1 - P(j/y))(1-x_j)$$

(b) Multinomial Naive Bayes Classifier

⇒ Test Accuracy:

39.021

⇒ Considering the frequency count, we have:

$$x = [x_1, x_2, \ldots, x_m] \text{ where}$$
$$x_m \in \{1, \ldots, k\} \propto \underline{\text{word\_idx}}$$

⇒ Combining this distribution of P with fraction of documents belonging to each class 'j', word 'i' at a word frequency of 'f':

$$P(j) \propto \pi_j \prod_{i=1}^{|V|} P(i|j)^{f_i}$$

⇒ In order to avoid underflow, we will use the sum of log's:

$$P(j) \propto \log\left(\pi_j \prod_{i=1}^{|V|} P(i|j)^{f_i}\right)$$

$$P(j) = \log \pi_j + \sum_{i=1}^{|V|} f_i \log(P(i|j))$$

(c) TF-IDF with LDA

⇒ Test Accuracy:

39.568

⇒ TFIDF is calculation of term-frequency & Inverse Document frequency

$$TF(t,d) = \frac{\text{Count of term 't' in document 'd'}}{\text{Total number of words in 'd'}}$$

$$IDF(t) = \log\left(\frac{\text{Total number of documents 'N'}}{\text{document frequency 'df'} + 1}\right)$$

$$TF\text{-}IDF(t,d) = TF(t,d) * IDF(t)$$

LDA - Linear Discriminant Analysis

⇒ Given the assumption: $\Sigma_1 = \Sigma_2 = \dots = \Sigma_k = \Sigma$

$$f(x) = \arg\min_c (x-\mu_c)^T \Sigma^{-1} (x-\mu_c) + b_c$$

$$f(x) = \arg\min_{c} \; x^T \Sigma^{-1} x + \mu_c^T \Sigma^{-1} \mu_{c_i} - 2\mu_c^T \Sigma^{-1} x + b_c$$

$$f(x) = \arg\max_{c} \; \mu_c^T \Sigma^{-1} x + b_c$$

Decision boundaries become linear ⇒

$$P(y=c/x) = \frac{e^{(\mu_c^T \Sigma^{-1} x + b_c)}}{\sum_j e^{(\mu_j^T \Sigma^{-1} x + b_c)}}$$

⇒ Where '$\mu_j$' is mean of all the documents which belong to class '$j$'.

(d) Least Square Classifier

⇒ Test Accuracy:
40.527

$$\Rightarrow f(x) = \underset{c}{\arg\max}\left(W_c^T x + b_c\right)$$

wher $c = 1, 2, \ldots, 20$

$\Rightarrow$ It can be solved using the following matrix operations.

$$\begin{bmatrix} W \\ b^T \end{bmatrix} = \begin{bmatrix} X^T X & X^T \mathbf{1} \\ \mathbf{1}^T X & n \end{bmatrix}^{-1} \begin{bmatrix} X^T y \\ 1^T y \end{bmatrix}$$

$\Rightarrow$ For solving this eq$^n$ for training data we will get Weight 'w' and bias 'b'.

$$X \longrightarrow X \longrightarrow X \longrightarrow X \longrightarrow X$$