TRABAJO TEMA 3 APR

1.- Demostrar que en cualquier problema de clasificación en C clases, la estimación de máxima verosimilitud de la probabilidad a priori de cada clase c, $\boldsymbol{1 \leq c \leq C}$, es $\boldsymbol{\hat{p}_c = {}^{n_c}/_N}$ donde $\boldsymbol{N = n_1 + \ldots + n_C}$ es el número total de datos observados y $\boldsymbol{n_c}$ es el número de datos de la clase c (ver el último ejemplo de aplicación de la técnica de los multiplicadores de Langrange, transparencias 3.17 y 3.18).

Modelo:

$$P(C = c) = \hat{p}_c$$

$$\sum_{c=1}^{C} \hat{p}_c = 1$$

$$\boldsymbol{\theta} = (\hat{p}_1, \ldots, \hat{p}_C)^t$$

Verosimilitud y logaritmo de la verosimilitud:

$$P(S|\boldsymbol{\theta}) = \prod_{i=1}^{n_1} \hat{p}_1 \cdots \prod_{j=1}^{n_C} \hat{p}_C = \hat{p}_1{}^{n_1} \ldots \hat{p}_C{}^{n_C}$$

$$q_S(\boldsymbol{\theta}) = L_S(\boldsymbol{\theta}) = \log P(S|\boldsymbol{\theta}) = n_1 * \log \hat{p}_1 + \cdots + n_C * \log \hat{p}_C$$

Estimación de máxima verosimilitud:

$$\boldsymbol{\theta}^* = \underset{\substack{\hat{p}_1, \ldots, \hat{p}_C \\ \hat{p}_1 + \cdots + \hat{p}_C = 1}}{\operatorname{argmax}} \ (n_1 * \log \hat{p}_1 + \cdots + n_C * \log \hat{p}_C)$$

Lagrangiana:

$$\Lambda(\hat{p}_1, \ldots, \hat{p}_C, \beta) = n_1 * \log \hat{p}_1 + \cdots + n_C * \log \hat{p}_C + \beta(1\text{-}\hat{p}_1 - \cdots - \hat{p}_C)$$

Soluciones óptimas en función del multiplicador de Lagrange:

$$\begin{matrix} \dfrac{\partial \Lambda}{\partial \hat{p}_1} = \dfrac{n_1}{\hat{p}_1} - \beta = 0 \\ \cdots \\ \dfrac{\partial \Lambda}{\partial \hat{p}_C} = \dfrac{n_C}{\hat{p}_C} - \beta = 0 \end{matrix} \Rightarrow \begin{matrix} \hat{p}_1^*(\beta) = \dfrac{n_1}{\beta} \\ \cdots \\ \hat{p}_C^*(\beta) = \dfrac{n_C}{\beta} \end{matrix}$$

Función dual de Lagrange

$$\Lambda_D(\beta) = n_1 log \frac{n_1}{\beta} + \cdots + n_C log \frac{n_C}{\beta} + \beta \left(1 - \frac{n_1}{\beta} - \cdots - \frac{n_C}{\beta}\right)$$

$$= \beta - N * \log \beta - N + \sum_{c=1}^{C} n_c \log n_c$$

Valor óptimo del multiplicador de Lagrange:

$$\frac{d\Lambda_D}{d\beta} = 1 - \frac{N}{\beta} = 0 \Rightarrow \beta^* = N$$

Solución final:

$$\hat{p}_1^* = \hat{p}_1^*(\beta) = \frac{n_1}{N} \dots \hat{p}_C^* = \hat{p}_C^*(\beta) = \frac{n_C}{N}$$

2.- Aplicar la técnica de descenso por gradiente a la búsqueda del mínimo de la función: $\boldsymbol{q(\theta) = (\theta_1 - 1)^2 + (\theta_2 - 2)^2 + \theta_1\theta_2}$ teniendo en cuenta que $\boldsymbol{\rho_k = {}^1/_{2k}}$ y $\boldsymbol{\theta(1) = (-1, +1)}$ y hacer una traza de las 3 primeras iteraciones.

Algoritmo general de descenso por gradiente:

$$\theta(1) = arbitrario$$

$$\theta(k+1) = \theta(k) - p_k \nabla \mathrm{q}(\theta)|_{\theta=\theta(k)}$$

Derivadas parciales:

$$\frac{dq}{d\theta_1} = 2\theta_1 + \theta_2 - 2$$

$$\frac{dq}{d\theta_2} = 2\theta_2 + \theta_1 - 4$$

Sustituimos en el algoritmo:

$$\theta(1) = \begin{pmatrix} -1 \\ +1 \end{pmatrix}$$

$$\theta(k+1) = \theta(k) - {}^1/_{2\mathrm{k}} \begin{pmatrix} 2\theta_1 + \theta_2 - 2 \\ 2\theta_2 + \theta_1 - 4 \end{pmatrix}$$

Primera iteración:

$$\theta(2) = \begin{pmatrix} -1 \\ +1 \end{pmatrix} - \frac{1}{2*1} \begin{pmatrix} 2*(-1) + 1 - 2 \\ 2*1 + (-1) - 4 \end{pmatrix} = \begin{pmatrix} {}^1/_2 \\ {}^5/_2 \end{pmatrix}$$

Segunda iteración:

$$\theta(3) = \begin{pmatrix} {}^1/_2 \\ {}^5/_2 \end{pmatrix} - \frac{1}{2*2} \begin{pmatrix} 2*({}^1/_2) + {}^5/_2 - 2 \\ 2*{}^5/_2 + ({}^1/_2) - 4 \end{pmatrix} = \begin{pmatrix} {}^1/_8 \\ {}^{17}/_8 \end{pmatrix}$$

Tercera iteración:

$$\theta(4) = \begin{pmatrix} {}^1/_8 \\ {}^{17}/_8 \end{pmatrix} - \frac{1}{2*3} \begin{pmatrix} 2*({}^1/_8) + {}^{17}/_8 - 2 \\ 2*{}^{17}/_8 + ({}^1/_8) - 4 \end{pmatrix} = \begin{pmatrix} {}^1/_{16} \\ {}^{33}/_{16} \end{pmatrix}$$

3.- Existe una variante de la función de Widrow-Hoff que incluye un término de regularización con el objetivo de que los pesos no se hagan demasiado grandes:

$$\boldsymbol{q_S(\theta) = \frac{1}{2}\sum_{n=1}^{N}(\theta^t x_n - y_n)^2 + \frac{\theta^t\theta}{2}}$$

Aplicando la técnica de descenso por gradiente, obtener la correspondiente variante del algoritmo de Widrow-Hoff y la correspondiente versión muestra a muestra.

Algoritmo general de descenso por gradiente:

$$\theta(1) = arbitrario$$

$$\theta(k+1) = \theta(k) - p_k \nabla \mathrm{q}(\theta)|_{\theta=\theta(k)}$$

Calculamos el gradiente:

$$\nabla q_S(\boldsymbol{\theta}) = \nabla \frac{1}{2}\sum_{n=1}^{N}(\boldsymbol{\theta}^t x_n - y_n)^2 + \frac{\boldsymbol{\theta^t\theta}}{2} = \sum_{n=1}^{N}(\boldsymbol{\theta}^t x_n - y_n)x_n + \boldsymbol{\theta}$$

Versión del algoritmo:

$$\theta(1) = arbitrario$$

$$\theta(k+1) = \theta(k) - p_k\left(\sum_{n=1}^{N}(\theta(\mathrm{k})^t x_n - y_n)x_n + \theta(\mathrm{k})\right)$$

$$= \theta(k) + p_k \textstyle\sum_{n=1}^{N}(y_n - \theta(\mathrm{k})^t x_n)x_n - p_k\theta(\mathrm{k})$$

Versión del algoritmo muestra a muestra:

$$\theta(1) = arbitrario$$

$$\theta(k+1) = \theta(k) + p_k(y(k) - \theta(\mathrm{k})^t x(k))x(k) - p_k\theta(\mathrm{k})$$