Problem 1: Gradient

$$g(A, \beta, x) = \sum_{i=1}^{p} \beta_{(i)} \left(\sum_{j=1}^{k} A_{(ij)} x_{(j)} \right)^3$$

$\beta \in R^p$, $A \in R^{p \times k}$, $x \in R^k$, compute $\nabla_A g$, $\nabla_x g$, $\nabla_\beta g$

$g \in R$ $\quad g(A, \beta, x) = \beta^T (Ax)^3$

$$\begin{cases} \nabla_A g = 3\,[\beta \circ (Ax)^2]\, x^T \\ \nabla_x g = 3 A^T [\beta \circ (Ax)^2] \\ \nabla_\beta g = (Ax)^3 \end{cases}$$

Proof:

$$g(x)$$

$$\uparrow$$

$$\boxed{\beta^T a^{(2)}} \longrightarrow \frac{\partial g}{\partial \beta} = a^{(2)} = (Ax)^3$$

$$\uparrow$$

$$a^{(2)}: \frac{\partial g}{\partial a^{(2)}} = \beta \qquad \frac{\partial g}{\partial a^{(2)}_{(j)}} = \beta_{(j)}$$

$$\uparrow$$

$$\boxed{(z^{(2)})^3}$$

$$\uparrow$$

$$z^{(2)}: \frac{\partial g}{\partial z^{(2)}} = \sum_j \frac{\partial g}{\partial a^{(2)}_{(j)}} \frac{\partial a^{(2)}_{(j)}}{\partial z^{(2)}} = \sum_j \beta_{(j)} \frac{\partial e_j^T (z^{(2)})^3}{\partial z^{(2)}} = \sum_j \beta_{(j)} 3 (z^{(2)})^2 e_j$$

$$= 3\,\beta \circ (Ax)^2$$

$$\frac{\partial g}{\partial A} = \sum_j \frac{\partial g}{\partial z^{(2)}_{(j)}} \frac{\partial z^{(2)}_{(j)}}{\partial A} \longleftarrow \boxed{Ax}$$

$$\uparrow$$

$$= \sum_j 3 [\beta \circ (Ax)^2]_{(j)} e_j x^T$$

$$= 3[\beta \circ (Ax)^2]\, x^T$$

$$x: \frac{\partial g}{\partial x} = \sum_j \frac{\partial g}{\partial z^{(2)}_{(j)}} \frac{\partial z^{(2)}_{(j)}}{\partial x} = \sum_j 3 [\beta \circ (Ax)^2]_{(j)} \frac{\partial e_j^T A x}{\partial x}$$

$$= 3 \sum_j [\beta \circ (Ax)^2]_{(j)} (e_j^T A)^T$$

$$= 3 \sum_j [\beta \circ (Ax)^2]_{(j)} A^T e_j$$

$$= 3 A^T [\beta \circ (Ax)^2]$$

$$\boxed{\frac{\partial z^{(2)}_{(j)}}{\partial A}} = \frac{\partial e_j^T A x}{\partial A} = \frac{\partial\, tr(e_j^T A x)}{\partial A}$$

$$= \frac{\partial\, tr(A x e_j^T)}{\partial A} = \frac{\partial\, tr[A (e_j x^T)^T]}{\partial A}$$

$$= e_j x^T$$

Problem 2: Weighted Logistic Regression

suppose $x_i \in R^d$ $\theta \in R^d$

Since $L(\theta)$ is a function of a vector $\in R^d$

the gradient is also a vector $\in R^d$

the j th entry of gradient is

$$[\nabla L(\theta)]_{(j)} = \frac{\partial L(\theta)}{\partial \theta_{(j)}}$$

$$= \frac{\partial \sum_{i=1}^{n} d_i \left[\log(1+\exp(x_i^T\theta)) - y_i x_i^T\theta\right]}{\partial \theta_{(j)}}$$

$$= \sum_{i=1}^{n} d_i \frac{\partial\left[\log(1+\exp(x_i^T\theta)) - y_i x_i^T\theta\right]}{\partial \theta_{(j)}} \quad \text{linearity of derivative}$$

$$= \sum_{i=1}^{n} d_i \frac{\partial \log(1+\exp(x_i^T\theta))}{\partial \theta_{(j)}} - \frac{\partial y_i x_i^T\theta}{\partial \theta_{(j)}}$$

$$= \sum_{i=1}^{n} d_i \left[\frac{\exp(x_i^T\theta)[x_i]_{(j)}}{1+\exp(x_i^T\theta)} - y_i[x_i]_{(j)}\right]$$

$$= \sum_{i=1}^{n} \left[d_i \left(\frac{\exp(x_i^T\theta)}{1+\exp(x_i^T\theta)} - y_i\right)[x_i]_{(j)}\right]$$

Then $\nabla L(\theta) = \sum_{i=1}^{n} \left[d_i \left(\frac{\exp(x_i^T\theta)}{1+\exp(x_i^T\theta)} - y_i\right) x_i\right]$

Problem 3 Weighted Linear Regression

$$y_i = x_i^T \beta^* + w_i \qquad w_i \sim N(0, \sigma_i^2)$$

$$y_i \sim N(x_i^T \beta^*, \sigma_i^2)$$

$$L(y; x, \beta^*, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left[-\frac{(y_i - x_i^T\beta^*)^2}{2\sigma_i^2}\right]$$

$$NLL = -\log L = -\log\left(\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left[-\frac{(y_i - x_i^T\beta^*)^2}{2\sigma_i^2}\right]\right)$$

$$= -\sum_{i=1}^{n} \log\left(\frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left[-\frac{(y_i - x_i^T\beta^*)^2}{2\sigma_i^2}\right]\right)$$

$$= -\sum_{i=1}^{n} \left[\log \frac{1}{\sqrt{2\pi\sigma_i^2}} - \frac{(y_i - x_i^T\beta^*)^2}{2\sigma_i^2}\right]$$

$$= \frac{n}{2} \log 2\pi + \sum_{i=1}^{n} \log \sigma_i + \frac{1}{2\sigma_i^2} \sum_{i=1}^{n} (y_i - x_i^T\beta^*)^2$$

The effect of adding a varible $w_i \sim N(0, \sigma_i^2)$ to a Gaussian compared to adding a constant $w \sim N(0, \sigma^2)$ to a Gaussian in standard linear regression is that when we try to find the optimal β by solving $\underset{\beta}{\operatorname{argmin}}\, NLL(\beta)$ by setting $\nabla_\beta NLL(\beta) = 0$, every sample will contribute a $\frac{1}{2\sigma_i^2}$ ratio which means every sample is weighted

P4: Regularization

$$L(\theta) = f(\theta) + \frac{\lambda}{2}\theta^T\theta$$

$$\nabla_\theta L(\theta) = \nabla f(\theta) + \lambda\theta$$

$$\theta_{k+1} = \theta_k - \eta_k \nabla_\theta L(\theta_k)$$

$$\boxed{\theta_{k+1} = \theta_k - \eta_k\left(\nabla f(\theta_k) + \lambda\theta_k\right)}$$ where η_k is the learning rate

P5 Concepts

a) $X \in R^{100\times4}$, $x_i \in R^4$ is the i^{th} flower sample in the training set.

$$y_i = \begin{cases} 1 & \text{species A} \\ 0 & \text{species B} \end{cases}$$

For a new test case $v \in R^4$, we calculate its distance between each flower sample in the set, so we have 100 distances. then we select the top 5 flower sample data points with shortest distances

Then we look at the species of these 5 data points.

we choose the most common species to be the label of the new test case.

---

b) $k=1$ will lead to an overfitted model, label of a new test case will be the species of the 1^{st} nearest data point, without capturing comprehensive features of the data set. Test error will be very large.

c) Because the data set only has 100 data points. Set $k=100$. then the label of a new test case will always be the species of majority of the data set.

P6. Gradient Descent

① if $|y-y'|<1$ $\quad l(y,y') = \frac{1}{2}(y-y')^2$

$$\nabla_\theta l(x_i^T\theta, y_i) = l'(x_i^T\theta, y_i)\,\nabla_\theta(x_i^T\theta)$$

$$= \left[(y-y')\Big|_{(x_i^T\theta,\, y_i)}\right]\cdot x_i$$

$$= (y_i - x_i^T\theta)\, x_i$$

② if $|y-y'| \geq 1$ $\quad l(y,y') = |y-y'| - \frac{1}{2}$

$$\nabla_\theta l(x_i^T\theta, y_i) = l'(x_i^T\theta, y_i)\,\nabla_\theta(x_i^T\theta)$$

$$= \left[\mathrm{sign}(y-y')\Big|_{(x_i^T\theta,\, y_i)}\right]\cdot x_i$$

$$= \mathrm{sign}(y_i - x_i^T\theta)\, x_i$$

$$\Rightarrow \boxed{\nabla_\theta l(x_i^T\theta, y_i) = \begin{cases} (y_i - x_i^T\theta)\, x_i & \text{if } |y-y'|<1 \\ \mathrm{sign}(y_i - x_i^T\theta)\, x_i & \text{otherwise} \end{cases}}$$

P7 MLE

Likelihood $L(\theta^*) = \prod_{i=1}^{n} f_\theta(y_i) = \begin{cases} \prod_{i=1}^{n} \exp[-(y_i - x_i^T\theta^*)] & y_i - x_i^T\theta^* \geq 0 \\ 0 & y_i - x_i^T\theta^* < 0 \end{cases}$

$$\text{Log } L(\theta^*) = \log L(\theta^*) = \log \prod_{i=1}^{n} \exp[-(y_i - x_i^T\theta^*)]$$

$$= \sum_{i=1}^{n} (x_i^T\theta^* - y_i)$$

$$\Rightarrow \boxed{\text{Log } L(\theta^*) = \sum_{i=1}^{n} (x_i^T\theta^* - y_i) \quad \text{for } y_i - x_i^T\theta^* \geq 0.}$$

Problem 8 MLE

$x_i \sim N(0, \sigma^2)$ i.i.d

$$p(x_i|\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x_i^2}{2\sigma^2}\right)$$

$$L(x;\sigma^2) = \prod_{i=1}^{n}\left[\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x_i^2}{2\sigma^2}\right)\right]$$

$$NLL(\sigma^2) = -\log L(x;\sigma^2) = -\sum_{i=1}^{n} \log\left[\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x_i^2}{2\sigma^2}\right)\right]$$

$$= \frac{n}{2}\log 2\pi + \frac{n}{2}\log\sigma^2 + \frac{1}{2\sigma^2}\sum_{i=1}^{n} x_i^2$$

$$\frac{\partial NLL(\sigma^2)}{\partial\sigma^2} = \frac{\partial\left(\frac{n}{2}\log\sigma^2 + \frac{1}{2\sigma^2}\sum_{i=1}^{n} x_i^2\right)}{\partial\sigma^2}$$

$$= \frac{n}{2}\frac{1}{\sigma^2} - \frac{\sum_{i=1}^{n} x_i^2}{2}\frac{1}{(\sigma^2)^2}$$

set $\frac{\partial NLL(\sigma^2)}{\partial\sigma^2} = 0$

$$\Rightarrow \boxed{\sigma^2 = \frac{\sum_{i=1}^{n} x_i^2}{n}}$$

The MLE estimate for σ^2 is $\frac{\sum_{i=1}^{n} x_i^2}{n}$

Problem 9 MLE

$$\boxed{\hat{\theta}_j = \log \frac{N_j}{n} \quad N_j \text{ is the number of } [y_i]_{(j)} = 2 \text{ in } n \text{ samples}}$$

$y_i \in R^k$

$$y_1 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} \begin{array}{l} \\ \to 2^{th} \text{ row} \\ \\ \\ \end{array} \quad [y_1]_{(2)} = 1 \Rightarrow Z_{(1)} = 2$$

$$\text{vector } Z = \begin{bmatrix} 2 \\ \vdots \\ j \\ \vdots \end{bmatrix} \begin{array}{l} \to 1^{th} \text{ row} \\ \\ \to i^{th} \text{ row} \\ \\ \end{array}$$

$$y_i = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \begin{array}{l} \\ \\ \to j^{th} \text{ row} \\ \\ \\ \end{array} \quad [y_i]_{(j)} = 1 \Rightarrow Z_{(i)} = j$$

For convenience, denote vector $Z \in R^n$ $\quad Z_{(i)} = k$ if $[y_i]_{(k)} = 1$

define $N_{(j)}$ = number of $Z_{(i)} = j$ $\quad N \in R^k \quad \sum_{j=1}^{k} N_{(j)} = n$

$$L(y;\theta) = L(Z;\theta) = \prod_{i=1}^{n} P(y) = \prod_{j=1}^{k} (p_j)^{N_{(j)}}$$

$$\begin{aligned} NLL &= -\log \prod_{j=1}^{k} (p_j)^{N_{(j)}} \\ &= -\sum_{j=1}^{k} \log (p_j)^{N_{(j)}} \\ &= -\sum_{j=1}^{k} N_{(j)} \log (p_j) \\ &= -\left[\sum_{j=1}^{k} N_{(j)} \log(p_j) + \lambda \left(1 - \sum_{j=1}^{k} p_j\right)\right] \qquad \sum_{j=1}^{k} p_j = 1 \end{aligned}$$

$$\begin{cases} \dfrac{\partial NLL(p_j,\lambda)}{\partial p_j} = \dfrac{N_{(j)}}{p_j} - \lambda = 0 \\[2ex] \dfrac{\partial NLL(p_j,\lambda)}{\partial \lambda} = 1 - \sum_{j=1}^{k} p_j = 0 \end{cases} \Rightarrow \begin{cases} \hat{p}_j = \dfrac{N_{(j)}}{\lambda} = \dfrac{N_{(j)}}{n} = \exp(\hat{\theta}_j) \\[2ex] \lambda = n \end{cases}$$

$$\Downarrow$$

$$\hat{\theta}_j = \log \frac{N_{(j)}}{n}$$

Also we could verify:

$$\boxed{\sum_{j=1}^{k} \exp(\hat{\theta}_j) = \sum_{j=1}^{k} \exp\left(\log \frac{N_{(j)}}{n}\right) = \sum_{j=1}^{k} \frac{N_{(j)}}{n} = \frac{1}{n} \cdot n = 1}$$

Problem 10 Concept

First I will standardize the new data point by the same way training set has done $x' = \frac{x-m}{s}$ (where m is mean, s is standard deviation of training set). Then apply function h to x' to predict its label.

Because the nearness of NN is based on distance, without standordize columns, if one variable is on the scale of billions while another is on tens. then scale of billions will contributes more to error. Thus the prediction will bias towards samples that are close in billions scale feature and ignore the other. Since the training set was standardized, the test set should also be standardized to the same scale.