CS480 Assignment 4

Nikhil Nanda
20727218

Exercise 1:

$$\rightarrow s_j = \frac{\sum_{i=1}^{n} r_{ik}(x_{ij} - \mu_j)^2}{\sum_{i=1}^{n} r_{ik}}$$

$$= \frac{\sum_{i=1}^{n} r_{ik} x_{ij}^2}{\sum_{i=1}^{n} r_{ik}} - \mu_j^2$$

The above equation comes from the Expectation step of the EM algorithm, where the covariance matrices S_k are constrained to be diagonal

On fixing the responsibility r_{ik},

$$\min_{\theta} \sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \left[-\log \pi_k + \frac{1}{2} \log |S_k| + \frac{1}{2} (x_i - \mu_k)^T \cdot S_k^{-1} (x_i - \mu_k) \right]$$

On taking derivative and setting it to zero, we get that for each K, the j-th diagonal

$$s_j = \frac{\sum_{i=1}^{n} r_{ik}(x_{ij} - \mu_j)^2}{\sum_{i=1}^{n} r_{ik}}$$

→ python3 gl.py

Exercise 2:

$$\begin{aligned}
\text{error}(f) &= P(Y \neq f(X)) \\
&= 1 - P(Y = f(X)) \\
&= 1 - \sum_{m=1}^{c} P(Y=m, f(X)=m) \\
&= 1 - \sum_{m=1}^{c} E\left[P(Y=m, f(X)=m \mid X)\right] \\
&= 1 - \sum_{m=1}^{c} E\left[1_{f(X)=m} P(Y=m \mid X)\right] \\
&= 1 - E \sum_{m=1}^{c} 1_{f(X)=m} P(Y=m \mid X)
\end{aligned}$$

$$\Rightarrow \boxed{\text{error}(f) = E\left[1 - \sum_{m=1}^{c} 1_{f(X)=m} P(Y=m \mid X)\right]}$$

- Thus, the minimum classification

$$\text{error}(f) = E\left[1 - \max_{m=1,\dots,c} P(Y=m \mid X)\right]$$

- This means that Bayes rule (with ties breaking arbitrarily)

$$f^*(X) = \operatorname*{argmax}_{m=1,\dots,c} P(Y=m \mid X)$$

achieves the minimum classification error

Exercise 3:

→ Pseudocode:

thresh$_{:,0}$ ← best threshold values of all features
thresh$_{:,1}$ ← error for all features with threshold value thresh$_{:,0}$
$X_{i,j}$ ← value at i-th row and j-th column of X
n ← number of training examples
Let thresh be a 2D-array of size $(n+1) \times 2$
For every feature j
    Sort the n X_{ij} values
    Set sum be an array of size n+1
    sum[0] ← 0
    sumSq ← 0
    for i ← 1 to n
        sum[i] ← y_i + sum[i−1]
        sumSq ← $(y_i)^2$ + sumSq
    Let error be an array of size n+1
    whose elements are initialized to 0
    for i ← 0 to n
        μ ← sum[i] / i
        ν ← (sum[n] − sum[i]) / (n−i)
        error[i] ← sumSq − (2 * μ * sum[i])
            + (i * (μ)²) + ((n−i) * (ν)²)
            − (2 * ν * (sum[n] − sum[i]))

index ← index of min value of error

if index == 0

thresh[j][0] ← $X_{1,j} - 0.1$

thresh[j][1] ← error[index]

else

thresh[j][0] ← $X_{index,j}$

thresh[j][1] ← error[index]

choose feature j with minimum thresh[j][1] value

→ Runtime:

$O(d*(n\log n + k*(n+1)))+(n+1))$, where k is a constant

∴ $O(d n \log n)$

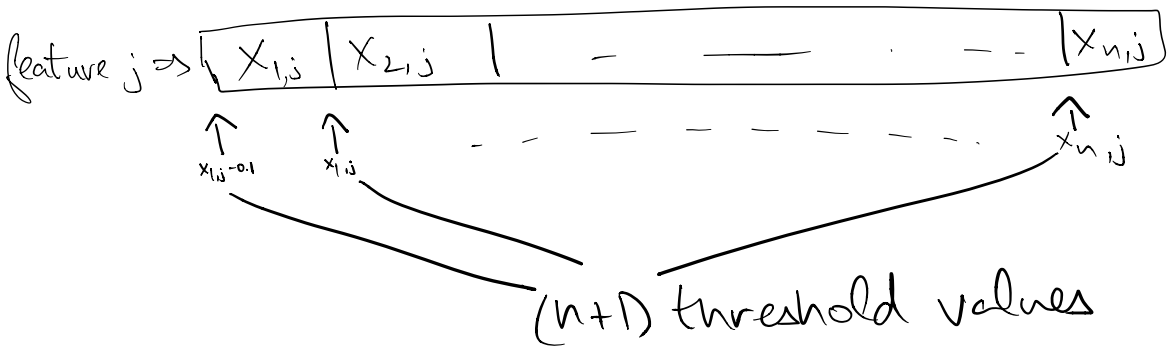
→ Space Complexity:

$O(k.n)$, where k is a constant

⇒ $O(n)$, assuming quicksort is used for sorting

→ The above algorithm goes through every feature in order to decide which feature to split at the root of the regression tree. Each feature is considered independently and a best threshold value (pertaining to minimum error or variance) is calculated for each feature

The best threshold for each feature is the one with minimum error or variance, chosen from (n+1) threshold values.



The error for each (n+1) threshold values of a particular feature is computed in O(n) time by the following pre-computations

$sum[i] \leftarrow \sum_{a=1}^{i} y_a$

$sumSq \leftarrow \sum_{i=1}^{n} (y_i)^2$

μ, ν are set to be the average of all y_i give the constraint on $X:j$

Thus,

$$\sum_{i:X_{ij} \le t_j} (y_i - \mu)^2 + \sum_{i:X_{ij} > t_j} (y_i - \nu)^2$$

$$= \sum_{i:X_{ij} \le t_j} y_i^2 - 2\mu \sum_{i:X_{ij} \le t_j} y_i + \mu^2 \sum_{i:X_{ij} \le t_j} 1$$

$$+ \sum_{i:X_{ij} > t_j} y_i^2 - 2\nu \sum_{i:X_{ij} > t_j} y_i + \nu^2 \sum_{i:X_{ij} > t_j} 1$$

$$= \sum_{i=1}^{n} y_i^2 - 2\mu \sum_{i:X_{ij} \le t_j} y_i + \mu^2 \sum_{i:X_{ij} \le t_j} 1$$

$$-2\nu \sum_{i:X_{ij} > t_j} y_i + \nu^2 \sum_{i:X_{ij} > t_j} 1$$

= error for each of the different (n+1) threshold values for each feature

Thus, the required result returned by the algorithm is the feature with the minimum error (OR) variance as computed above along with the best threshold value (minimum threshold value) for that feature.

Exercise 4:

$$\min_{w \in \mathbb{R}^d} \max_{\forall j, \|z_j\|_2 \le \lambda} \|(X+Z)w - y\|_2$$

$$\Rightarrow \min_{w \in \mathbb{R}^d} \max_{\forall j, \|z_j\|_2 \le \lambda} \max_{\|u\|_2 \le 1} \left[(X+Z)w - y\right]^T u$$

$$\Rightarrow \min_{w \in \mathbb{R}^d} \max_{\|u\|_2 \le 1} \max_{\forall j, \|z_j\|_2 \le \lambda} \left[(Xw - y)^T u + (Zw)^T u\right]$$

$$\Rightarrow \min_{w \in \mathbb{R}^d} \max_{\|u\|_2 \le 1} \left[(Xw-y)^T u + \max_{\forall j, \|z_j\|_2 \le \lambda} (Zw)^T u\right]$$

$$\Rightarrow \min_{w \in \mathbb{R}^d} \left[\|Xw - y\|_2 + \max_{\forall j, \|z_j\|_2 \le \lambda} \|Zw\|_2\right]$$

$$\Rightarrow \min_{w \in \mathbb{R}^d} \left[\|Xw - y\|_2 + \max_{\forall j, \|z_j\| = \lambda} \sqrt{\sum_j z_j w_j}^2\right]$$

$$\Rightarrow \min_{w \in \mathbb{R}^d} \|Xw - y\|_2 + \lambda \sqrt{\sum_j w_j}^2$$

$$\Rightarrow \boxed{\min_{w \in \mathbb{R}^d} \|Xw - y\|_2 + \lambda \|w\|_1}$$