CS 6140: Machine Learning	**Samuel Steiner steiner.s@husky.neu.edu**

Homework Assignment #2

Assigned : 02/16/2021 — *Due: 03/01/2021, 11:59pm, through Canvas*

Three problems, 100 points in total. Good luck!
Prof. Predrag Radivojac, Northeastern University

Problem 1. (20 points) Naive Bayes classifier. Consider the following binary classification problem where there are 8 data points in the training set. That is,

$$\mathcal{D} = \{(-1,-1,-1,-),(-1,-1,1,+),(-1,1,-1,+),(-1,1,1,-),(1,-1,-1,+),(1,-1,1,-),(1,1,-1,-),(1,1,1,+)\},$$

where each tuple (x_1,x_2,x_3,y) represents a training example with input vector (x_1,x_2,x_3) and class label y.

a) (10 points) Construct a naive Bayes classifier for this problem and evaluate its accuracy on the training set. Measure accuracy as the fraction of correctly classified examples.

Stated in Radivojac & White "let $\mathcal{X}$ and $\mathcal{Y}$ be an input and output space respectively with $\mathcal{Y}$ being discrete ... the decision rule for labeling a data point is

$$\begin{aligned}\hat{y} &= \arg\max_{y\in\mathcal{Y}} p(y \mid x)\\ &= \arg\max_{y\in\mathcal{Y}} \{p(x \mid y)p(y)\}\end{aligned}$$

... assuming d-dimensional inputs we can write

$$p(x \mid y) = \prod_{j=1}^{d} p(x_j \mid y).$$

"

Table 1: Probability table for y

y	-	+	p(-)	p(+)
	4	4	1/2	1/2

Table 2: Probability table for x_1

x_1	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

Table 3: Probability table for x_2

x_2	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

Table 4: Probability table for x_3

x_3	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

For our $\mathcal{D}$ we have a problem where all values are the same and the conditional probabilities are the same so in the end our values are the same. The classifier will 'randomly' select a class to give the data set. For all possible $\mathcal{X}$ and for each $y \in \mathcal{Y}$ meaning the accuracy for this classifier is undetermined:

$$p(x_1|y) * p(x_1|y) * p(x_1|y) * p(y)$$
$$\frac{1}{2} * \frac{1}{2} * \frac{1}{2} * \frac{1}{2} = 1/16$$

b) (10 points) Transform the input space into a higher-dimensional space

$$(x1,x2,x3,x_1x_2,x_1x_3,x_2x_3,x_1x_2x_3,x_1^2,x_2^2,x_3^2,x_1^2x_2,x_1x_2^2,x_1x_3^2,x_2^2x_3,x_2x_3^2)$$

and repeat the previous step.

Carry out all steps manually and show all your calculations. Discuss your main observations. Since the conditional probability for all $p(x|y)$ are the same not counting $x_1x_2x_3$ the only consideration should

Table 5: Probability table for x_1, x_2, x_3

$x_1x_2x_3$	-	+	p(-)	p(+)
1	0	4	0	1
-1	4	0	1	0
total	4	4	1	1

Table 6: Probability table for x_1^2

x_1^2	-	+	p(-)	p(+)
1	4	4	1	1

Table 7: Probability table for x_2^2

x_2^2	-	+	p(-)	p(+)
1	4	4	1	1

Table 8: Probability table for x_3^2

x_3^2	-	+	p(-)	p(+)
1	4	4	1	1

Table 9: Probability table for x_1x_2

x_1x_2	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

Table 10: Probability table for x_1x_3

x_1x_3	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

Table 11: Probability table for x_2x_3

x_2x_3	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

Table 12: Probability table for $x_1x_2^2$

$x_1x_2^2$	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

Table 13: Probability table for $x_1x_3^2$

$x_1x_3^2$	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

Table 14: Probability table for $x_2^2x_3$

$x_2^2x_3$	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

Table 15: Probability table for $x_1^2x_2$

$x_1^2x_2$	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

Table 16: Probability table for $x_2x_3^2$

$x_2x_3^2$	-	+	p(-)	p(+)
1	2	2	1/2	1/2
-1	2	2	1/2	1/2

be $x_1x_2x_3$. Here are the calculations for all 8 data points

$$\mathcal{D}_{1(Y=-)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*1*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 1/2048$$
$$\mathcal{D}_{1(Y=+)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*0*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 0$$

$$\mathcal{D}_{2(Y=-)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*0*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 0$$
$$\mathcal{D}_{2(Y=+)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*1*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 1/2048$$

$$\mathcal{D}_{3(Y=-)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*0*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 0$$
$$\mathcal{D}_{3(Y=+)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*1*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 1/2048$$

$$\mathcal{D}_{4(Y=-)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*1*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 1/2048$$
$$\mathcal{D}_{4(Y=+)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*0*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 0$$

$$\mathcal{D}_{5(Y=-)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*0*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 0$$
$$\mathcal{D}_{5(Y=+)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*1*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 1/2048$$

$$\mathcal{D}_{6(Y=-)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*1*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 1/2048$$
$$\mathcal{D}_{6(Y=+)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*0*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 0$$

$$\mathcal{D}_{7(Y=-)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*1*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 1/2048$$
$$\mathcal{D}_{7(Y=+)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*0*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 0$$

$$\mathcal{D}_{8(Y=-)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*0*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 0$$
$$\mathcal{D}_{8(Y=+)} = \frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*1*1*1*1*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2} = 1/2048$$

This classifier has a 100% accuracy. as stated earlier the classifier is wholly reliant on the $x_1x_2x_3$ data point in the input space. The reason why this happens is because our data is distributed in a way where $p(x \mid y)$ of any value in the data set of part a is the same.

Problem 2. (25 points) Consider a binary classification problem in which we want to determine the optimal decision surface. A point **x** is on the decision surface if $P(Y=1 \mid x) = P(Y=0 \mid x)$.

a) (10 points) Find the optimal decision suface assuming that each class-conditional distribution is defined as a two-dimensional Gaussian distribution.

$$p(x \mid Y=i) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} * e^{-\frac{1}{2}(x-m_i)^T\Sigma_i^{-1}(x-m_i)}$$

where $i \in \{0,1\}$, $m_0 = (1,2), m_1 = (6,3), \Sigma_0 = \Sigma_1 = \mathbf{I}_2, P(Y=0) = P(Y=1) = 1/2, \mathbf{I}_d$ is the d-dimensional identity matrix, and $|\Sigma_i|$ is the determinant of Σ_i.

$$P(Y=1|x) = P(Y=0|x)$$
$$\frac{P(x|Y=1)P(Y=1)}{P(x)} = \frac{P(x|Y=0)P(Y=0)}{P(x)}$$
$$P(x|Y=1)P(Y=1) = P(x|Y=0)P(Y=0)$$
$$P(Y=1) = P(Y=0) \implies P(x|Y=1) = P(x|Y=0)$$
$$d = 2$$
$$\frac{1}{(2\pi)|\Sigma_1|^{1/2}} * e^{-\frac{1}{2}(x-m_1)^T\Sigma_1^{-1}(x-m_1)} = \frac{1}{(2\pi)|\Sigma_0|^{1/2}} * e^{-\frac{1}{2}(x-m_0)^T\Sigma_0^{-1}(x-m_0)}$$
$$\Sigma_1 = \Sigma_2 \implies$$
$$e^{-\frac{1}{2}(x-m_1)^T\Sigma^{-1}(x-m_1)} = e^{-\frac{1}{2}(x-m_0)^T\Sigma^{-1}(x-m_0)}$$
log both sides to simplify
$$-\frac{1}{2}(x-m_1)^T\Sigma^{-1}(x-m_1) = -\frac{1}{2}(x-m_0)^T\Sigma^{-1}(x-m_0)$$
simplify both sides
$$x^T\Sigma^{-1}m_1 - \frac{1}{2}m_1^T\Sigma^{-1}m_1 - \frac{1}{2}x^T\Sigma^{-1} = x^T\Sigma^{-1}m_0 - \frac{1}{2}m_0^T\Sigma^{-1}m_0 - \frac{1}{2}x^T\Sigma^{-1}$$
$$x^T\Sigma^{-1}m_1 - \frac{1}{2}m_1^T\Sigma^{-1}m_1 = x^T\Sigma^{-1}m_0 - \frac{1}{2}m_0^T\Sigma^{-1}m_0$$
$$-\frac{1}{2}(m_1+m_0)^T\Sigma^{-1}(m_1-m_0) + x^T\Sigma^{-1}(m_1-m_0) = 0$$
$$-\frac{1}{2}(\begin{bmatrix}6 & 3\end{bmatrix} + \begin{bmatrix}1 & 2\end{bmatrix})\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}(\begin{bmatrix}6\\3\end{bmatrix} - \begin{bmatrix}1\\2\end{bmatrix}) + \begin{bmatrix}x_0 & 0\\0 & x_1\end{bmatrix}\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}(\begin{bmatrix}6\\3\end{bmatrix} - \begin{bmatrix}1\\2\end{bmatrix}) = 0$$
$$-\frac{1}{2}\begin{bmatrix}7 & 5\end{bmatrix}\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\begin{bmatrix}5\\1\end{bmatrix} + \begin{bmatrix}x_0 & 0\\0 & x_1\end{bmatrix}\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\begin{bmatrix}5\\1\end{bmatrix} = 0$$
$$-20 + 5x_0 + x_1 = 0$$

b) (5 points) Generalize the solution from part (a) using $m_0 = (m_{01}, m_{02}), m_1 = (m_{11}, m_{12}), \Sigma_0 = \Sigma_1 = \sigma^2 \mathbf{I}_2$ and $P(Y = 0) \neq P(Y = 1)$. For $P(Y = 0) \neq P(Y = 1)$ lets say $p_y = P(Y = y)$ instead of canceling out they would be logged turned into addition then in the last steps when combined would be turned into subtraction or simplified to $\log \frac{p_0}{p_1}$ making the final equation

$$\log \frac{p_0}{p_1} - \frac{1}{2}(m_1 + m_0)^T \Sigma^{-1}(m_1 - m_0) + x^T \Sigma^{-1}(m_1 - m_0) = 0$$

for $\sigma^2 \mathbf{I}_2$ the Σ^{-1} would become

$$\begin{bmatrix} 1/\sigma^2 & 0 \\ 0 & 1/\sigma^2 \end{bmatrix}$$

for $(m_1 +, m_0)^T$ the matrix would become

$$\begin{bmatrix} m_{11} + m_{01} & m_{12} + m_{02} \end{bmatrix}$$

for $m_1 +, m_0$ the matrix would become

$$\begin{bmatrix} m_{11} - m_{01} \\ m_{12} - m_{02} \end{bmatrix}$$

The final equation would look like this

$$\log \frac{p_0}{p_1} - \frac{1}{2}(\frac{1}{\sigma^2}(m_{11}^2 - m_{01}^2 + m_{12}^2 - m_{02}^2)) + x_0 \frac{1}{\sigma^2}(m_{11} - m_{01}) + x_1(\frac{1}{\sigma^2}(m_{12} + m_{02})) = 0$$

c) (10 points) Generalize the solution from part (b) to arbitrary co-variance matrices Σ_0 and Σ_1. Discuss the shape of the optimal decision surface.

We would not be able to remove the $\frac{1}{(2\pi)|\Sigma_i|^{1/2}}$, when we log both sides this would transform into

$$\log \frac{1}{(2\pi)} + \log \frac{1}{|\Sigma_i|^{1/2}} = \log \frac{1}{|\Sigma_i|^{1/2}} = -\frac{1}{2} \log |\Sigma_i|$$

lets describe a function $G_i(x)$ as $log(P(Y = y)) - \frac{1}{2} \log |\Sigma_i| - \frac{1}{2}(x - m_i)^T \Sigma_i^{-1}(x - m_i)$ then from this we would need to solve this for each Σ_i this would make the problem quadratic in nature. The optimal decision surface would be a quadratic curve.

Problem 3. (55 points) Consider a multivariate linear regression problem of mapping $\mathbb{R}^d$ to $\mathbb{R}$ with two different objective functions. The first objective function is the sum of squared errors, as presented in class; i.e., $\sum_{i=1}^n e_i^2$ where $e_i = w_0 + \sum_{j=1}^d w_j x_{ij} - y_i$. The second objective function is the sum of square Euclidean distances to the hyperplane; i.e., $\sum_{i=1}^n r_i^2$, where r_i is the Euclidean distance between point (x_i, y_i) to the hyperplane $f(x) = w_0 + \sum_{j=1}^d w_j x_j$.

a) (10 points) Derive a gradient descent algorithm to find the parameters of the model that minimizes the sum of squared errors.

$$\text{Let } W = \{w_0, w_1, w_2, \ldots w_d\}$$

$$\text{Intialize } W^{(t=0)} \text{ with random values or 0s}$$

$$\text{Let } \alpha \in (0, 1)$$

$$\text{Set } \frac{\delta}{\delta w_j} \sum_{i=1}^n e_i^2 \text{ as the cost function} \quad j = 0, 1, \ldots d$$

$$-\frac{\delta}{\delta w_j} \sum_{i=1}^n e_i^2 = \begin{cases} \frac{\delta}{\delta w_0} & = 2\ y_i - \sum_{j=1}^d (w_j x_{ij})(x_{i0}) \\ \frac{\delta}{\delta w_1} & = 2\ y_i - \sum_{j=1}^d (w_j x_{ij})(x_{i1}) \\ \vdots & \\ \frac{\delta}{\delta w_d} & = 2\ y_i - \sum_{j=1}^d (w_j x_{ij})(x_{id}) \end{cases}$$

repeat until convergence: {

$$W^{t+1} = W^t - \alpha(-\frac{\delta}{\delta w_j} \sum_{i=1}^n e_i^2)$$

$$t = t + 1$$

}

b) (20 points) Derive a gradient descent algorithm to find the parameters of the model that minimizes the sum of squared distances.

$$r_i = \frac{f(x_i) - y_i}{||w||}$$

which can be written as

$$r_i = \frac{w^T x_i - y_i}{||w||}$$

$$r_i^2 = \frac{(w^T x_i - y_i)^2}{||w||^2}$$

for sum of r_i^2 assume X is the matrix of all x_i assume Y is a column vector and W is a row vector.

$$\sum r_i^2 = (\frac{(W^T X - Y)}{||W||})^2$$

$$\nabla_W \sum r^2 = 2(\frac{(W^T X - Y)}{||W||})((1/\|w\| \cdot X)^T - 1/\|w\|^3 \cdot (X^T w - y) \cdot w^T)$$

our algorithm follows:

Let $W = \{w_0, w_1, w_2, \dots w_d\}$

Intialize $W^{(t=0)}$ with random values

Let $\alpha \in (0,1)$

repeat until convergence: $\{$

$$W^{t+1} = W^t - \alpha(\nabla_W \sum_{i=1}^{n} r^2)$$

$$t = t+1$$

$\}$

c) (20 pooints) Implement both algorithms and test them on 3 datasets. Datasets can be randomly generated, as in class, or obtained from resources such as UCI Machine Learning Repository. Compare the solutions to the closed-form (maximum likelihood) solution derived in class and find the R^2 in all cases on the same dataset used to fit the parameters; i.e., do not implement cross-validation. Briefly describe the data you use and discuss your results.

item c answered with item d

d) (5 points) Normalize every feature and target using a linear transform such that the minimum value for each feature and the target is 0 and the maximum value is 1. The new value for feature j of data point i can be found as

$$x_{ij}^{\text{new}} = \frac{x_{ij} - \min_{k\in\{1,2,3,\cdots,n\}} x_{kj}}{\max_{k\in\{1,2,3,\cdots,n\}} x_{kj} - \min_{k\in\{1,2,3,\cdots,n\}} x_{kj}}$$

where n is the dataset size. the new value for the target i can be found as

$$y_{ij}^{\text{new}} = \frac{y_i - \min_{k\in\{1,2,3,\cdots,n\}} y_k}{\max_{k\in\{1,2,3,\cdots,n\}} y_k - \min_{k\in\{1,2,3,\cdots,n\}} y_k}$$

Measure the number of steps towards convergence and compare with the results from part (c). Briefly discuss your results.

I selected 3 datasets form UCI, which include ratings for red wine, data on concrete and data on airfoil noise. Each of these dataset had a different number of input data and only 1 output column which made them good candidates for regression. The full results for each data set for this experiment and each method is included in the data folder in a csv titled final_data.csv some trends were that for *SSE* the normalization returned results which were closer to the maximum likeliness and the step count was way reduced before hitting convergence. Interestingly the *SED* the step cap of 100000 4 times, twice on normalized data and twice without. This makes me question the validity of using euclidean distance to the hyperplane as an objective function for gradient descent. Overall R^2 were better after normalization, showing that normalization of data as pre-processing with greatly affect the overall performance of gradient descent.