CS561 HW8

November 1, 2023

1 Problem 1

$f : \mathbb{R} \to \mathbb{R}$ is a function that satisfies the following property: - (i) f inputs a value corresponding to a probability $\mathbb{P}(X = x)$, which means $f : [0, 1] \to \mathbb{R}$ - (ii) For independent random variables X_1 and X_2, $f(\mathbb{P}(X_1 = x_1, X_2 = x_2)) = f(\mathbb{P}(X_1 = x_1)) + f(\mathbb{P}(X_2 = x_2))$ - (iii) f is non-negative, which means $f : [0, 1] \to \mathbb{R}_0^+$ - (iv) f is a decreasing function - (v) $f(1) = 0$

(a) Properties (i) and (iii) combined ensures that f takes in probability (since information is measured from probability) and always never returns a negative value (we only gain information from knowing more probability). Properties (v) suggests the information gained from a knowing an event that always happen is 0, which is reasonable because we technically did not gain further information from certain events. Property (iv) follows that the less probability, the greater the information, which is justified as knowledge of rarer event means a greater deal of information regarding said event. Lastly, (ii) suggests that we can add information from independent events. This is reasonable because we can visualize the combined information of two independent events as them happening separately, one at a time.

(b) The function $f(\mathbb{P}(X = x)) = \log_2 \left(\frac{1}{\mathbb{P}(X=x)} \right)$ can be simplified as

$$f(x) = -\log_2(x)$$

that takes in value of $x \in [0, 1]$ (since $\mathbb{P}(X = x) \in [0, 1]$, thus satisfying (i)). The function $f(x) = -\log_2(x)$ is non-negative (since values of x in that $[0, 1]$ outputs $\log_2(x) \leq 0$, thus $-\log_2(x) \geq 0$, satisfying (iii)). Moreover, we know $\log_2(1) = 0$ thus $f(1) = 0$ (satisfying (v)) and since log is an increasing function, f is decreasing, satisfying (ii). Since $\mathbb{P}(X_1 = x_1, X_2 = x_2) = \mathbb{P}(X_1 = x_1)\mathbb{P}(X_2 = x_2)$, property (iv) can be written as follows:

$$f(xy) = f(x)f(y)$$

for all $x, y \in [0, 1]$. We can see that $f(x) = -\log_2(x)$ satisfies this property from the log of product property.

(c) Suppose f is twice differentiable (implying f is continuous). The properties can be re-written such that: $f : [0, 1] \to \mathbb{R}^+$ is a decreasing function with $f(1) = 0$ and

$$\forall x, y \in [0, 1], f(xy) = f(x)f(y)$$

Suppose $F(x) = f(e^{-x})$ is a composite function. We can see that F has a domain of $\mathbb{R}_0^-$, is continuous, increasing, satisfies

$$\forall x, y \in \mathbb{R}_0^-, F(x + y) = F(x) + F(y)$$

and has $F(0) = 0$. For $n \in \mathbb{Z}^-$, consider

$$\begin{aligned} F(-1) &= F\left(\frac{n+1}{n}\right) + F\left(\frac{1}{n}\right) \\ &= F\left(\frac{n+2}{n}\right) + F\left(\frac{1}{n}\right) + F\left(\frac{1}{n}\right) \\ &= ... \\ &= -n\left(F\left(\frac{1}{n}\right)\right) \end{aligned}$$

Therefore, for any $q \in \mathbb{Q}_0^-$, write $q = a/n$, $a \in \mathbb{Z}_0^+$, $n \in \mathbb{Z}^-$,

$$\begin{aligned} F(q) &= F\left(\frac{a}{n}\right) \\ &= F\left(\frac{a-1}{n}\right) + F\left(\frac{1}{n}\right) \\ &= F\left(\frac{a-2}{n}\right) + F\left(\frac{1}{n}\right) + F\left(\frac{1}{n}\right) \\ &= ... \\ &= a\left(F\left(\frac{-1}{n}\right)\right) \\ &= \frac{-a(F(-1))}{n} \\ &= -qF(-1) \end{aligned}$$

Therefore, $F(q) = cq$ for all $q \in \mathbb{Q}_0^-$. Recall that $\mathbb{R}_0^-$ is a closed set under a complete metric space $\mathbb{R}$. Since $\mathbb{Q}$ is dense in $\mathbb{R}$, we can construct a cauchy sequence of non-positive rationals $(q_i)_{i \geq 0}$ that converges to r given any $r \in \mathbb{R}_0^-$. By the continuity of F, $(F(q_i))_{i \geq 0}$ converges to $F(r)$. However, we've shown that $(F(q_i))_{i \geq 0} = (cq_i)_{i \geq 0}$, which means $F(r) = cr$ since the point of convergence is unique. Therefore, $f(e^{-x}) = cx$, in which we can substitute $r = e^{-x}$ to obtain

$$f(r) = -c\log(r) = \log_k\left(\frac{1}{r}\right)$$

for some positive base k as desired.

2 Problem 2

Suppose $\mathbf{x} \in \mathcal{X}^n$ is a feature vector over $\mathcal{X}$ with a classifier $\hat{y} = f(x)$. We refer to the property

$$H(\mathbf{x}, \hat{y}) = H(\mathbf{x}) + H(\hat{y}|\mathbf{x}) = H(\hat{y}) + H(\mathbf{x}|\hat{y})$$

Since $\hat{y}$ is a function on $\mathbf{x}$, $\hat{y}|\mathbf{x}$ is a constant random variable that always return $f(\mathbf{x})$ with probability 1, which means its entropy is 0 ($H(\hat{y}|\mathbf{x}) = 0$). Moreover, $H(\mathbf{x}|\hat{y})$ is a value of an entropy function which is nonnegative. Therefore,

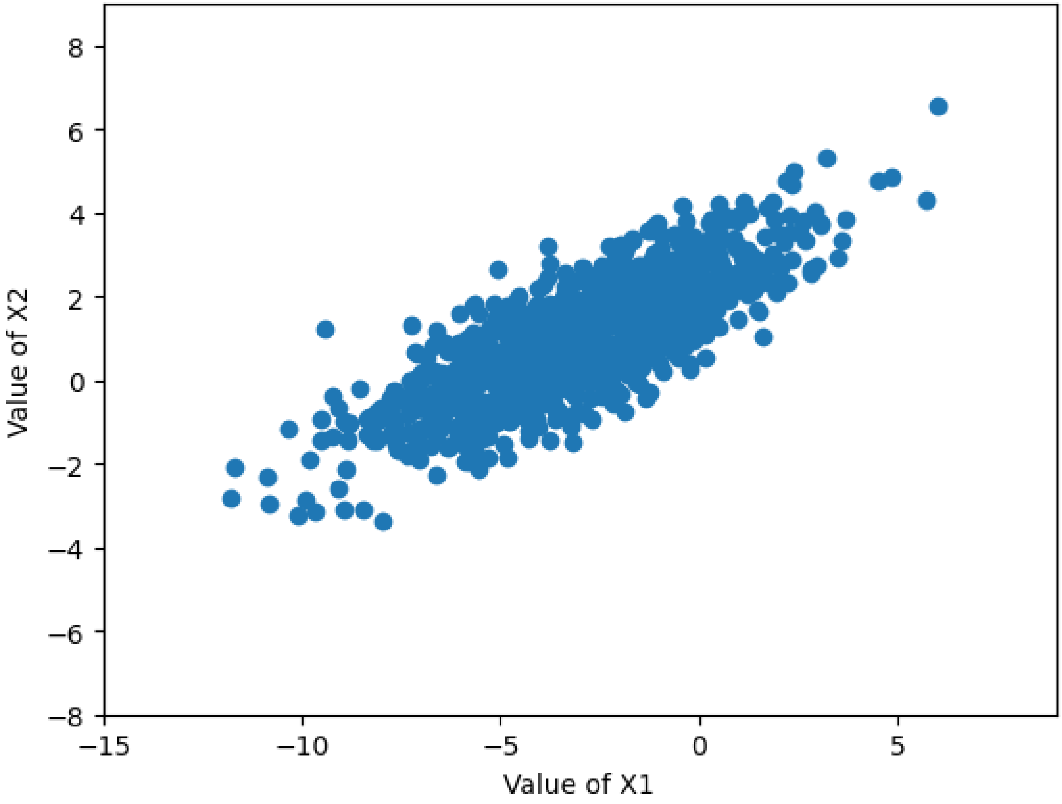
$$H(\mathbf{x}) = H(\mathbf{x}) + H(\hat{y}|\mathbf{x}) = H(\hat{y}) + H(\mathbf{x}|\hat{y}) \geq H(\hat{y})$$

as desired

3 Problem 3

```
[1]: # 3a
     import numpy as np
     import matplotlib.pyplot as plt
     cov = np.array([[7,3],[3,2]])
     scaling_factor = np.linalg.cholesky(cov)
     mean = np.array([-3,1])

     points = np.array([(scaling_factor@np.random.randn(2,1)).reshape(-1)+mean for _␣
      ↪in range(1000)])
     plt.scatter(points[:,0],points[:,1])
     plt.xlim([-15,9]) ; plt.ylim([-8,9])
     plt.xlabel("Value of X1"); plt.ylabel("Value of X2")
     plt.show()
```



(b) We refer to the property discussed in class:

$$X_1|X_2 \sim \mathcal{N}(\Sigma_{12}\Sigma_{22}^{-1}(X_2 - \mu_2) + \mu_1, \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})$$

Where if we condition on $X_2 = 0$ with the given μ and Σ,

$$X_1|(X_2 = 0) \sim \mathcal{N}((3)(2)^{-1}(0 - 1) + (-3), (7) - (3)(2)^{-1}(3))$$

in which we can simplify

$$X_1|(X_2 = 0) \sim \mathcal{N}(-4.5, 2.5)$$

(c) We refer to the property $Ax \sim \mathcal{N}(A\mu, A\Sigma A^T)$. Put $A = \begin{bmatrix} 0 & 1 \end{bmatrix}$ to take the marginal of X_2. With the given μ and Σ, this results in

$$X_2 = Ax \sim \mathcal{N}(1, 2)$$

4 Problem 4

(a) Suppose we define a quadratic discriminant classifier as such: we'd like to find a condition such that the distribution

$$x|Y = 0 \sim \mathcal{N}(\mu_0, \Sigma_0)$$

and

$$x|Y = 1 \sim \mathcal{N}(\mu_1, \Sigma_1)$$

where our classifier outputs

$$\frac{p(x|Y = 0)}{p(x|Y = 1)} \geq 1$$

as class 0 and 1 otherwise. In other words

$$\frac{\frac{1}{(2\pi)^{1/2}|\Sigma_0|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_0)^T \Sigma_0^{-1}(x - \mu_0)\right)}{\frac{1}{(2\pi)^{1/2}|\Sigma_1|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_1)^T \Sigma_1^{-1}(x - \mu_1)\right)} \geq 1$$

taking the logarithm on both sides, we obtain

$$-\frac{1}{2}\log(|\Sigma_0|) - \frac{1}{2}(x - \mu_0)^T \Sigma_0^{-1}(x - \mu_0) \geq -\frac{1}{2}\log(|\Sigma_1|) - \frac{1}{2}(x - \mu_1)^T \Sigma_1^{-1}(x - \mu_1)$$

we can simplify the expression as follows:

$$x^T(\Sigma_1^{-1} - \Sigma_0^{-1})x + \left(\mu_0^T(\Sigma_0^{-1} + (\Sigma_0^{-1})^T) - \mu_1^T(\Sigma_1^{-1} + (\Sigma_1^{-1})^T)\right)x \geq \log(|\Sigma_0|) - \log(|\Sigma_1|) + \mu_0^T \Sigma_0^{-1}\mu_0 - \mu_1^T \Sigma_1^{-1}\mu_1$$

therefore, the threshold for $y = 0$ can be written in the form

$$x^T B x + W^T x \geq c$$

where

$$\begin{aligned} B &= \Sigma_1^{-1} - \Sigma_0^{-1} \\ W^T &= \mu_0^T(\Sigma_0^{-1} + (\Sigma_0^{-1})^T) - \mu_1^T(\Sigma_1^{-1} + (\Sigma_1^{-1})^T) \\ c &= \log(|\Sigma_0|) - \log(|\Sigma_1|) + \mu_0^T \Sigma_0^{-1}\mu_0 - \mu_1^T \Sigma_1^{-1}\mu_1 \end{aligned}$$

[9]:

```
#4a, numerical answer
mu0 = np.array([1,0])
mu1 = np.array([0,2])
sigma0 = np.array([[8,3],[3,2]])
sigma1 = np.array([[1,0.1],[0.1,1]])

B = np.linalg.inv(sigma1)-np.linalg.inv(sigma0)
WT = mu0.T@(np.linalg.inv(sigma0)+np.linalg.inv(sigma0).T)-mu1.T@(np.linalg.
 ↪inv(sigma1)+np.linalg.inv(sigma1).T)
```

```
c = np.log(np.linalg.det(sigma0))-np.log(np.linalg.det(sigma1))+mu0.T@np.linalg.
 ↪inv(sigma0)@mu0-mu1.T@np.linalg.inv(sigma1)@mu1

print(f"B = {B}")
print(f"W^T = {WT}")
print(f"c = {c}")

x2_coeff = B[0,0]
xy_coeff = B[0,1]+B[1,0]
y2_coeff = B[1,1]
x_coeff = WT[0]
y_coeff = WT[1]
const = c

print('Equation (conic section) : ax1^2+bx2^2+cx1x2+dx1+ex2>=const, when')
print(f"a = {x2_coeff}")
print(f"b = {y2_coeff}")
print(f"c = {xy_coeff}")
print(f"d = {x_coeff}")
print(f"e = {y_coeff}")
print(f"const = {const}")
```

```
B = [[ 0.72438672  0.32756133]
 [ 0.32756133 -0.13275613]]
W^T = [ 0.97546898 -4.8975469 ]
c = -1.7987292697809405
Equation (conic section) : ax1^2+bx2^2+cx1x2+dx1+ex2>=const, when
a = 0.7243867243867245
b = -0.13275613275613263
c = 0.655122655122655
d = 0.9754689754689755
e = -4.897546897546897
const = -1.7987292697809405
```

[10]:

```
# 4b and c
import numpy as np

n = 1000
class_0 = np.random.multivariate_normal(np.array([1,0]), np.
 ↪array([[8,3],[3,2]]), n//2)
class_1 = np.random.multivariate_normal(np.array([0,2]), np.array([[1,0.1],[0.
 ↪1,1]]), n//2)
plt.scatter(class_0[:,0],class_0[:,1], label="y=0")
plt.scatter(class_1[:,0],class_1[:,1], label="y=1")

x1min = np.min([np.min(class_0[:,0]),np.min(class_1[:,0])])
x1max = np.max([np.max(class_0[:,0]),np.max(class_1[:,0])])
```

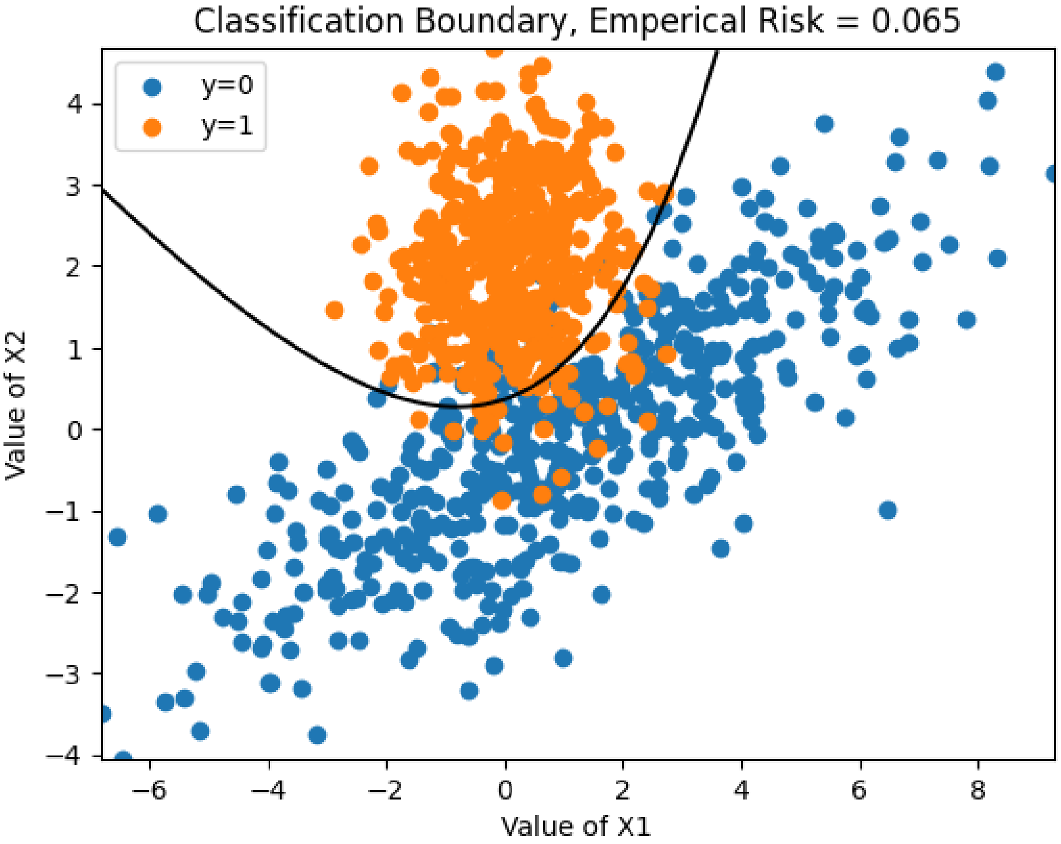
```
x2min = np.min([np.min(class_0[:,1]),np.min(class_1[:,1])])
x2max = np.max([np.max(class_0[:,1]),np.max(class_1[:,1])])

x1 = np.arange(x1min, x1max, 0.1)
x2 = np.arange(x2min, x2max, 0.1)
x1, x2 = np.meshgrid(x1, x2)

plt.contour(x1, x2,(x2_coeff*x1**2 + xy_coeff*x1*x2 + y2_coeff*x2**2 +␣
 ↪x_coeff*x1 + y_coeff*x2 - const), [0], colors='k')
plt.xlim(x1min, x1max);plt.ylim(x2min, x2max)
plt.xlabel("Value of X1"); plt.ylabel("Value of X2")

plt.legend()

def classifier(x):
    if x.T@B@x+WT@x>=c:
        return 0
    else:
        return 1
pred0 = np.array([classifier(v) for v in class_0])
pred1 = np.array([classifier(v) for v in class_1])
error = (np.sum(pred0==1)+np.sum(pred1==0))/n
plt.title(f"Classification Boundary, Emperical Risk = {error}")
plt.show()
```


Classification Boundary, Emperical Risk = 0.065
y=0
y=1
Value of X2
4
3
2
1
0
−1
−2
−3
−4
−6
−4
−2
0
2
4
6
8
Value of X1

[ ]: