Hw 7

Sam Pell

1/5/2024

Recall that in class we showed that for randomized response differential privacy based on a fair coin (that is a coin that lands heads up with probability 0.5), the estimated proportion of incriminating observations $\hat{P}$ [1] was given by $\hat{P} = 2\pi - \frac{1}{2}$ where π is the proportion of people answering affirmative to the incriminating question.

I want you to generalize this result for a potentially biased coin. That is, for a differentially private mechanism that uses a coin landing heads up with probability $0 \leq \theta \leq 1$, find an estimate $\hat{P}$ for the proportion of incriminating observations. This expression should be in terms of θ and π.

Student Answer

We first isolate both variables from our equation:

$$\hat{P} = 2\pi - \frac{1}{2}$$

$$\pi = \frac{1}{2}\hat{P} + \frac{1}{4}$$

Then, it is important to return to the base table, which defines how we will return values based on what the truth is:

$$f(Truth) = \begin{cases} f_1(Truth), & 'Truth' \\ f_2(Truth), & 'Truth' \\ f_3(Truth), & 'Yes' \\ f_4(Truth), & 'No' \end{cases}$$

$$f(Yes) = \begin{cases} f_1(Yes), & 'Yes' \\ f_2(Yes), & 'Yes' \\ f_3(Yes), & 'Yes' \\ f_4(Yes), & 'No' \end{cases}$$

[1] in class this was the estimated proportion of students having actually cheated

$$f(No) = \begin{cases} f_1(No), & 'No' \\ f_2(No), & 'No' \\ f_3(No), & 'Yes' \\ f_4(No), & 'No' \end{cases}$$

And from here we can calculate probabilities of each outcome:

$$P(f_1) = P(H, H) = \theta^2$$

$$P(f_2) = P(H, T) = \theta * (1 - \theta) = \theta - \theta^2$$

$$P(f_3) = P(T, H) = (1 - \theta) * \theta = \theta - \theta^2$$

$$P(f_4) = P(T, T) = (1 - \theta)^2$$

From these outcomes, we can summarize the odds of receiving a given response in terms of θ

$$\pi = P('Yes'|Yes) = P(f_1) + P(f_2) + P(f_3)$$

$$\pi = \theta^2 + \theta - \theta^2 + \theta - \theta^2 = 2\theta - \theta^2$$

$$P('Yes'|No) = P(f_4) = (1 - \theta^2)$$

Where in the special case where our coin is fair, we consistently find: $P\left('Yes'|Yes, \theta = \frac{1}{2}\right) = \pi = 2\frac{1}{2} - \frac{1^2}{2} = 1 - \frac{1}{4} = \frac{3}{4}$, which aligns with classwork.

Thus, to estimate $\hat{P}$, we understand that there is a θ chance of them immediately telling the truth, rendering $\theta\hat{P}$. Further, there is a θ chance that their response will still align with the truth, giving θ^2. Thus, adding these odds together, we find that $\pi = \theta\hat{P} + \theta^2$. From this, $\hat{P} = \frac{1}{\theta}(\pi - \theta^2)$

To this end, when $\theta = \frac{1}{2}$, $\hat{P} = 2\left(\pi - \frac{1^2}{2}\right) = 2\pi - \frac{1}{2}$.

Part of having an explainable model is being able to implement the algorithm from scratch. Let's try and do this with `KNN`. Write a function entitled `chebychev` that takes in two vectors and outputs the Chebychev or L^∞ distance between said vectors. I will test your function on two vectors below. Then, write a `nearest_neighbors` function that finds the user specified k nearest neighbors according to a user specified distance function (in this case L^∞) to a user specified data point observation.

```
#student input
#chebychev function

chebychev <- function(x, y){
```

```
  return(max(abs(x - y)))
}
#nearest_neighbors function
nearest_neighbors <- function(data, obs = row(data), k = 5, dist_function =
chebychev){
  dists <- apply(data, 1, function(row) dist_function(row, obs))
  indices <- order(dists)[1:k]
  return(indices)
}

x<- c(3,4,5)
y<-c(7,10,1)
chebychev(x,y)

## [1] 6
```

Finally create a `knn_classifier` function that takes the nearest neighbors specified from the above functions and assigns a class label based on the mode class label within these nearest neighbors. I will then test your functions by finding the five nearest neighbors to the very last observation in the `iris` dataset according to the `chebychev` distance and classifying this function accordingly.

```
library(class)
df <- data(iris)
#student input
mode <- function(x){
  u <- unique(x)
  u[which.max(tabulate(match(x, u)))]
}

knn_classifier <- function(nearest_indices, classes) {
  return(nearest_indices$classes)
}

#data less last observation
x = iris[1:(nrow(iris)-1),]
#observation to be classified
obs = iris[nrow(iris),]

#find nearest neighbors
ind = nearest_neighbors(x[,1:4], obs[,1:4],5, chebychev)[[1]]
as.matrix(x[ind,1:4])

##     Sepal.Length Sepal.Width Petal.Length Petal.Width
## 128          6.1           3          4.9         1.8

obs[,1:4]
```

```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width
## 150          5.9           3          5.1         1.8
```

```
knn_classifier(x[ind,], 'Species')
```

```
## NULL
```

```
obs[,'Species']
```

```
## [1] virginica
## Levels: setosa versicolor virginica
```

Interpret this output. Did you get the correct classification? Also, if you specified $K = 5$, why do you have 7 observations included in the output dataframe?

Student Answer

I am desperately misunderstanding this question. I think my function should be returning just one observation, since it is only being fed one, with x[ind,], but am I supposed to be getting 7? Why would I get that? My function also keeps returning NULL, even though I think I've tested just about every possible combination for it manually. I simply have to give up on this one in favor of other pursuits.

Earlier in this unit we learned about Google's DeepMind assisting in the management of acute kidney injury. Assistance in the health care sector is always welcome, particularly if it benefits the well-being of the patient. Even so, algorithmic assistance necessitates the acquisition and retention of sensitive health care data. With this in mind, who should be privy to this sensitive information? In particular, is data transfer allowed if the company managing the software is subsumed? Should the data be made available to insurance companies who could use this to better calibrate their actuarial risk but also deny care? Stake a position and defend it using principles discussed from the class.

Student Answer

When discussing sensitive information, it is important to prioritize data privacy as a first and foremost priority. While it is often helpful or even necessary to share personal information with others, the people whose sensitive information it is should be the primary decision makers on who is enabled access to their data. With that said, supposing it is medical data, these patients should be involved in deciding how far their data is allowed to propogate. Whether or not a department of analysts at the hospital can access their data should be their informed decision, and whether said data can continue to be used by insurance companies, other hospitals, or an acquiring corporation. To act otherwise would be to subsume continued consent into an initial agreement, which certainly does not hold

up to scrutiny. Furthermore, the reduction in privacy could cause more harm to the person than good, and may not even be justifiable in a utilitarian sense, such as if data were shared with an insurance agency that ultimately raised its prices because of it, leading to less affordable healthcare and lower utility in the long run.