Hw 7

Nikita McClure

12/2/2024

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

Student Answer

chances of hearing "yes" (π) = truthful yeses ("yes" given.yes) + yeses based on command, true or not ("yes" given.command)

$\hat{\pi} = (\text{"}yes\text{"}given.yes) + (\text{"}yes\text{"}given.command)$

"yes"given.yes = likelihood of first flip being heads (θ) * proportion of students who have actually cheated (P) "yes"given.yes = $\theta\hat{P}$

"yes"given.command = likelihood of first flip being tails $(1-\theta)$ * likelihood of second flip being heads (θ) "yes"given.command = $(1-\theta)\theta$

$\hat{\pi} = \theta\hat{P} + (1-\theta)\theta$

to find the proportion of truths: solve for P

$\hat{\pi} = \theta\hat{P} + \theta - \theta^2$

$\theta\hat{P} = \hat{\pi} - \theta + \theta^2$

$\hat{P} = \frac{\hat{\pi}}{\theta} - \frac{\theta}{\theta} + \theta$

$\hat{P} = \frac{\hat{\pi}}{\theta} - 1 + \theta$

$\hat{P} = \frac{\hat{\pi}}{\theta} + \theta - 1$

Next, show that this expression reduces to our result from class in the special case where $\theta = \frac{1}{2}$.

Student Answer *with a fair coin, when $\theta = \frac{1}{2}$, the equation boils down to $\hat{P} = 2\hat{\pi} - \frac{1}{2}$

$\hat{P} = \frac{\hat{\pi}}{\theta} + \theta - 1$ replace θ with $\frac{1}{2}$

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

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

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

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
#nearest_neighbors function

chebychev <- function(a, b) {
  max(abs(a-b))

}

nearest_neighbors = function(x, obs, k, dist_func){
  dist = apply(x, 1, dist_func, obs) #apply along the rows
  distances = sort(dist)[1:k]
  neighbor_list = which(dist %in% sort(dist)[1:k])
  return(list(neighbor_list, distances))
}


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

knn_classifier = function(x,y){

  groups = table(x[,y])
  pred = groups[groups == max(groups)]
```

```
  return(pred)
}

#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
## 71           5.9         3.2          4.8         1.8
## 84           6.0         2.7          5.1         1.6
## 102          5.8         2.7          5.1         1.9
## 127          6.2         2.8          4.8         1.8
## 128          6.1         3.0          4.9         1.8
## 139          6.0         3.0          4.8         1.8
## 143          5.8         2.7          5.1         1.9
```

```
obs[,1:4]
```

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

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

```
## virginica
##         5
```

```
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

It classified the unknown observation as virginica and was correct. It specified that 5 of the closest points were virginica, however it returned a total of 7 points. It did so because there were multiple points the same distance from the observation being classified. The four closest points all had unique distances, but there were 3 distances tied for 5th place so they were all included. While the *5 closest observations* may not have all been virginica, the *majority* of the observations in the *5 nearest distances*, (5 of 7 points) were virginica leading to the correct classification of virginica.

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 Those who should be privy to the sensitive healthcare data are medical professionals treating the patients and those working on the development of the algorithm that will benefit those patients and more. The data should not be available to entities such as insurance companies, even if the software company is subsumed by it. The insurance companies could use the data to better calibrate their actuarial risk which could help some patients by allowing those with less risk to pay lower premiums and could even make allocation of resources more equitable. However, this could hurt those with higher risk. Their premiums could be increased, making it harder, even unreasonable, for them to pay. The insurance companies could even deny care to those who need it most. This potential major drawback greatly outweighs the potential benefits in my opinion.

Even if the intention of the insurance company was good, to make coverage more equitable, that potential outcome is bad, making the whole act bad. This is following the philosophy of consenquentialism, that even if the intention is good, the outcome of the act is what determines if it is right or wrong.

When considering an act with both pros and cons, such as this healthcare app, a subset of consequentialism refered to as utilitarianism must be applied. The overall benefit verses the overall detriments (pleasure vs pain) must be analyzed, to determine the goodness of the act.

If it is given that for the app to be created or maintained the insurance company must have access to the information, the benefits of the app as a whole must be weighed. However, even when accounting for the potential benefits of the app entirely, the consequence of those most in need being refused coverage is still almighty. As the bad outweighs the good, the act is bad according to utilitarianism. However, if the app can be established without the insurance companies being privy to the information, the main consequence to consider becomes lack of privacy, then I believe the benefits would outweigh the detriments and the app would be overall good.

I have described our responsibility to proper interpretation as an *obligation* or *duty*. How might a Kantian Deontologist defend such a claim?

Student Answer A Kantian Deontologist would believe the responsibility we have to proper interpretation of data science is a true obligation or duty of ours. This is because the Categorical Imperative deems so. It describes the Universal Maxim Formulation which dictates that an act is justifiable only if it can be universalized. Proper interpretation of (data) science can and should be universalized. If everyone is interpreting results accurately then everyone can trust the results leading to more trust in, and overall improvement of, science. The alternative, misinterpretation, being universalized would lead to full mistrust undermining science as a whole.

Additionally, the Categorical Imperative includes the Ends-Not-Means Formulation, which says that the act can never treat a "moral agent" strictly as a means to an end. If data science is intentionally misinterpreted, the people who provided the data are being disrespected and used only as means to the interpreter's end-generally profit or gain.