Hw 7

Tianrui Ye

1/5/2024

1

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

Probability of Answering "Yes"

Let π be the true proportion of people who have the incriminating attribute (e.g., have cheated). The probability $P(Y = \text{"Yes"})$ that a person answers "Yes" can be broken down as follows: - The probability of answering "Yes" truthfully (coin lands heads and they are guilty): $\theta \cdot \pi$. - The probability of answering "Yes" falsely (coin lands tails, they answer "Yes" by the mechanism's design): $1 - \theta$.

So, the total probability of a "Yes" answer is:

$$P(Y = \text{"Yes"}) = \theta\pi + (1 - \theta).$$

Estimating π from $\hat{P}$

Given $P(Y = \text{"Yes"})$ from the collected data, let's denote it as $\hat{P}$. Our goal is to express π in terms of $\hat{P}$ and θ. Rearranging the equation:

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

to solve for π, we get:

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

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

This equation provides the estimate of π using the observed proportion $\hat{P}$ of "Yes" answers, adjusting for the bias introduced by the coin's probability θ.

2

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

Student Answer

When $\theta = 0.5$ (unbiased coin), this formula simplifies to:

$$\pi = \frac{\hat{P} - 0.5}{0.5} = 2\hat{P} - 1$$

which matches the original expression $\hat{P} = 2\pi - \frac{1}{2}$ when re-arranged to solve for $\hat{P}$ from π. This verifies that the generalization is correct.

3

Consider the additive feature attribution model: $g(x') = \phi_0 + \sum_{i=1}^{M} \phi_i x'_i$ where we are aiming to explain prediction f with model g around input x with simplified input x'. Moreover, M is the number of input features.

Give an expression for the explanation model g in the case where all attributes are meaningless, and interpret this expression. Secondly, give an expression for the relative contribution of feature i to the explanation model.

Student Answer

If all attributes are meaningless, the attributions ϕ_i for all i would effectively be zero since they do not contribute to explaining the variation in f. Thus, the model simplifies to:

$$g(x') = \phi_0$$

Interpretation: This expression means that the prediction does not depend on any of the input features. The prediction $g(x')$ is constant and equals ϕ_0, which could be interpreted as the expected value of f when no information from the features is available.

The relative contribution of feature i to the explanation model g can be given by:

$$\text{Contribution of } i = \phi_i x'_i$$

Interpretation: This term shows how much feature i, scaled by its simplified input x'_i, contributes to the overall prediction made by g. The coefficient ϕ_i captures the effect size of feature i, and multiplying it by x'_i adjusts this effect by the actual (simplified) value of the feature in the specific instance being explained.

4

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.

```
chebychev <- function(x, y) {
  max(abs(x - y))
}

nearest_neighbors <- function(data, point, k, distance_func) {
  distances <- apply(data, 1, function(row) distance_func(row, point))

  data[order(distances), ][1:k, ]
}

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

chebychev(x,y)
```

5

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(neighbors, class_column) {
  class_labels <- neighbors[[class_column]]
  mode <- function(x) {
    count <- unique(x)
    count[which.max(tabulate(match(x, count)))]
  }

  mode(class_labels)
}

#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])
obs[,1:4]
knn_classifier(x[ind,], 'Species')
obs[,'Species']
```

6

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

The output matrix displays the sepal length, sepal width, petal length, and petal width for the nearest neighbors of our target sample. Notably, the matrix shows seven neighbors rather than the expected five. This discrepancy arises because our function returns all points that tie for the five smallest Chebychev distances to the target, leading to extra entries in the case of a tie at the fifth position. Consequently, seven neighbors are listed. The output also includes a dataframe that describes the attributes of the sample being classified. Although the classification process predominantly labels these neighbors as ‘virginica’, it fails to accurately confirm that the target observation belongs to the ‘virginica’ species, as indicated by the final output.

7

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

Privacy and Confidentiality: The primary principle from our discussions is the inviolability of patient confidentiality. Health data is highly sensitive, and access should be strictly limited to individuals directly involved in patient care. This includes healthcare providers, and possibly specific technicians or data scientists working under strict confidentiality agreements to improve the health outcomes using such technologies.

Data Transfer in Corporate Changes: If the company managing the software, such as Google’s DeepMind, is acquired or changes hands, the transfer of sensitive health data to the new entity should only occur under stringent regulatory oversight. This oversight should ensure that the new entity adheres to the same or higher standards of data privacy and usage as originally agreed upon. The patient’s consent must also be revisited, as they have the right to know and decide if and how their data will be handled by the new entity.

Access by Insurance Companies: While providing access to insurance companies could theoretically allow for better calibration of actuarial risks, this raises significant concerns about potential misuse of the data, such as denying care based on risk profiles. The ethical principle of non-maleficence, or “do no harm,” must guide decisions. Therefore, health data should not be made available to insurance companies for the purpose of adjusting actuarial tables if there is a risk that it could lead to denial of care or discrimination against patients based on their health data.

Position: Sensitive health data, like that used by Google’s DeepMind for managing kidney issues, should be governed by strict regulations that prioritize patient privacy and consent, limit access strictly to those involved in care and improvements directly benefiting patient outcomes, and prohibit uses that could harm patients, such as denial of care based on data-derived risk profiles. Enhanced transparency and patient involvement in consent processes, continuous monitoring of data use, and stringent legal frameworks to protect data misuse are crucial. This approach aligns with the ethical principles of beneficence, non-maleficence, and justice discussed in our classes.

---

1. in class this was the estimated proportion of students having actually cheated↩