CSCI 1052 Problem Set 2

January 18, 2024

Submission Instructions

Please upload your solutions by **5pm Friday January 19, 2024.**

- You are encouraged to discuss ideas and work with your classmates. However, you **must acknowledge** your collaborators at the top of each solution on which you collaborated with others and you **must write** your solutions and code independently.
- Your solutions to theory questions must be typeset in LaTeX or markdown. I strongly recommend uploading the source LaTeX (found here) to Overleaf for editing.
- I recommend that you write your solutions to coding question in a Jupyter notebook using Google Colab.
- You should submit your solutions as a **single PDF** via the assignment on Gradescope. You can enroll in the class using the code GPXX7N.
- Once you uploaded your solution, **mark where you answered each part of each question**.

Problem 1: Normal Distribution from Darts

Figure 1: A dartboard. Intuitively, the likelihood that a dart hits a particular point should only depend on the distance to the center and rotating the dartboard shouldn't change where darts likely land.

In this problem, we will derive the density function of the normal distribution from the example of a dartboard. Let $f : \mathbb{R}^d \to [0, 1]$ be a probability density function that describes the probability a dart lands at the point (x, y). We want our probability density function f to have two properties:

1. **Radial symmetry:** The probability that a dart lands at a point depends only on the distance between the point and the origin.
2. **Independence:** Coordinates are independent e.g., knowing the x-coordinate does not give us information about the y-coordinate.

From the radial symmetry property, we can conclude that

$$f(x, y) = f(r) = f(\sqrt{x^2 + y^2}) \tag{1}$$

where $r = \sqrt{x^2 + y^2}$ is the distance between the origin and (x, y).

From the independence property, we can conclude that $f(x, y) = g(x)h(y)$ for some functions g and h. Further, the radial symmetry property tells us that $f(x, y) = f(y, x) = g(y)h(x)$ so g and h must be the same function. That is,

$$f(x, y) = g(x)g(y). \tag{2}$$

Part 1 (.5 points)

Up to rescaling, we can assume that $g(0) = 1$. Use this fact and plug in the point $(r, 0)$ to conclude that f and g are the same function.

Part 2 (.5 points)

Define the function $h(x) = f(\sqrt{x})$. In particular,

$$f(\sqrt{x^2 + y^2}) = f(x)f(y) \Leftrightarrow h(x^2 + y^2) = h(x^2)h(y^2). \tag{3}$$

Use this property to show that

$$h(x_1 + x_2 + \ldots + x_n) = h(x_1)h(x_2) \cdot \ldots \cdot h(x_n). \tag{4}$$

Then prove that $h(n) = h(1)^n$.

Without loss of generality, let $h(1) = b$.

Part 3 (1 point)

Let p and q be any integers. Prove that

$$h\left(\frac{p}{q}\right) = b^{\frac{p}{q}}. \tag{5}$$

Hint: Consider the expression

$$h\left(\frac{p}{q} \cdot q\right).$$

As long as h is continuous, conclude that $h(x) = b^x$ for any real number since the rational numbers are dense in the real number line.

Part 4 (1 point)

You have shown that $h(x) = b^x$. Further, $h(x) = e^{cx}$ for some real number c. Then $f(x) = h(x^2) = e^{cx^2}$. Recall this implies the density function of x and y is $f(x, y) = f(x)f(y) = e^{c(x^2+y^2)}$.

For f to be a valid probability density function, what constraint do we have on c? If we wanted to make the distribution more concentrated, how should we change c?

Except for the normalization, we have explained every part of the multivariate normal distribution given below

$$f(x, y) = \frac{1}{\sigma^2 \cdot 2\pi} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right).$$

Why is there a normalization?

Problem 2: Johnson-Lindenstrauss for Join Size Estimations

In class, we showed the Johnson-Lindenstrauss Lemma for preserving the norm of differences. Consider vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$. Let $k = O\left(\frac{\log n}{\epsilon^2}\right)$. We showed that a random matrix $\mathbf{\Pi} \in \mathbb{R}^{k \times d}$ satisfies

$$(1-\epsilon)\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 \leq \|\mathbf{\Pi}\mathbf{x}_i - \mathbf{\Pi}\mathbf{x}_j\|_2^2 \leq (1+\epsilon)\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 \tag{6}$$

with probability 9/10.

Part 1 (2 points)

Show that we can use the Johnson-Lindenstrauss Lemma as stated above to show that inner-products are also preserved. In particular, under the same conditions as above,

$$|\langle \mathbf{x}_i, \mathbf{x}_j \rangle - \langle \mathbf{\Pi}\mathbf{x}_i, \mathbf{\Pi}\mathbf{x}_j \rangle| \leq \frac{1}{2}\epsilon(\|\mathbf{x}_i\|_2^2 + \|\mathbf{x}_j\|_2^2) \tag{7}$$

with probability 9/10.

Hint 1: Show that $\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 = \|\mathbf{x}_i\|_2^2 - 2\langle \mathbf{x}_i, \mathbf{x}_j \rangle + \|\mathbf{x}_j\|_2^2$.

Hint 2: Apply the Johnson-Lindenstrauss Lemma and Hint 1 to the term $\|\mathbf{\Pi}\mathbf{x}_i + \mathbf{\Pi}\mathbf{x}_j\|_2^2 - \|\mathbf{\Pi}\mathbf{x}_i - \mathbf{\Pi}\mathbf{x}_j\|_2^2$.

Part 2 (1 point

One powerful application of sketching is in database applications. For example, a common goal is to estimate the *inner join size* of two tables without performing an actual inner join (which is expensive, as it requires enumerating the keys of the tables). Formally, consider two sets of unique keys $X = \{x_1, \ldots, x_m\}$ and $Y = \{y_1, \ldots, y_n\}$ which are subsets of $1, 2, \ldots, U$. Our goal is to estimate $|X \cap Y|$ based on small space compressions of X and Y.

Using your result from Part 1, describe a method based on inner product estimation that constructs independent sketches of X and Y of size $k = O\left(\frac{\log n}{\epsilon^2}\right)$ and from these sketches can return an estimate Z for $|X \cap Y|$ satisfying

$$|Z - |X \cap Y|| \leq \epsilon(|X| + |Y|)$$

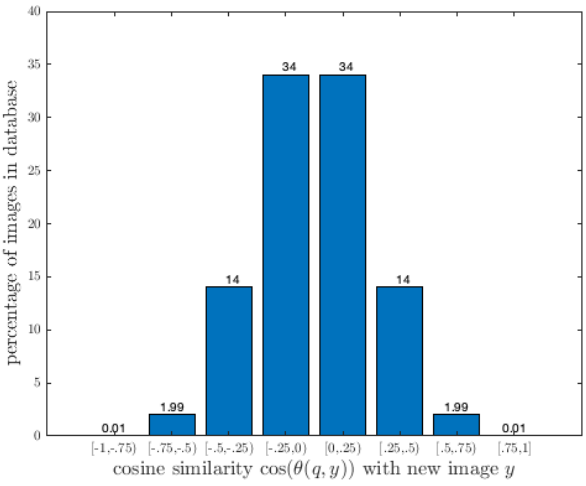
with probability 9/10.

Problem 3: LSH in the Wild

To support its largely visual platform, Pinterest runs a massive image de-duplication operation built on Locality Sensitive Hashing for Cosine Similarity. You can read about the actual system here. All information and numbers below are otherwise purely hypothetical.

Pinterest has a database of $N = \mathbf{1\ billion}$ images. Each image in the database is pre-processed and represented as a vector $\mathbf{q} \in \mathbb{R}^d$. When a new image is pinned, it is also processed to form a vector $\mathbf{y} \in \mathbb{R}^d$. The goal is to check for any existing duplicates or near-duplicates to $\mathbf{y}$ in the database. Specifically, Pinterest would like to flag an image $\mathbf{q}$ as a near-duplicate to $\mathbf{y}$ if $\cos(\theta(\mathbf{q}, \mathbf{y})) \geq .98$. We want to find any near-duplicate with probability $\geq 99\%$.

Given this requirement, your job is to design a multi-table LSH scheme using SimHash to find candidate near-duplicates, which can then be checked directly against $\mathbf{y}$. To support this task, Pinterest has collected data on the empirical distribution of $\cos(\theta(\mathbf{q}, \mathbf{y}))$ for a typical new image $\mathbf{y}$. It roughly follows a bell-curve:



Pinterest wants to consider two possible computational targets for your LSH scheme, which will determine the speed of the de-duplication algorithm:

1. Ensure that no more than 1 million candidate near-duplicates are checked on average when a new image is pinned. "Checked" means that the image's cosine similarity with the new image is computed explicitly, which is a computationally expensive operation.

2. Ensure that no more than $200,000$ candidates are checked on average when a new image is pinned.

Based on the data above, describe how to set parameters for your LSH scheme to minimize the space (i.e., number of tables) used, while achieving each of the above goals. Justify your answers, and any assumptions you make. If you code anything up to help calculate your answer, please attach the code. As in class, you can assume that each hash table has $m = O(N)$ slots and this is large enough to ignore lower order terms depending on $1/m$.