Exercise Sheet: Practical Data Collection

Data, Data Storage, Data Collection

Exercise 1 - Designing a Sampling Algorithm

You are tasked with collecting data on student study habits at your university. The target population is all 10,000 students, but you only have resources to survey 500. The university has 4 faculties with the following enrollments:

Faculty	Number of Students
Science	3000
Humanities	2000
Engineering	4000
Business	1000

(1) Systematic Sampling:

You are given a list of all 10,000 students.

(a) Write an algorithm to select every k-th student from the list to achieve a sample of 500. Shift the index of the first taken by a random value. The input should be the list of students *studentList*, and the size of the desired sample *sampleSize* (here 500), and the output should be a systematic sample of the associated size.

(b) What potential bias could this method introduce? How would you mitigate it?

(a) Pseudocode:

```
Algorithm 1: Systematic Sampling
  Input: The list of students studentList, and the size of the
         desired sample sampleSize (here 500)
  Output: Sample of students computed systematically
k ← size (studentList) / sampleSize ;                  // (=20)
start ← randomInt (1, k);
sample ← empty list;
;
for i ← 0 to sampleSize do
    sample ← sample ∪ studentList[start + i * k];
end
return sample;
```

(b) Potential Bias: If the student list is ordered in a way that correlates with the variable of interest (e.g., students sorted by faculty), systematic sampling could introduce periodicity bias by over- or under-representing certain groups. A mitigation strategy is to shuffle the student list before applying systematic sampling, or use stratified sampling instead.

(2) Stratified sampling

Stratified sampling is a technique where the population is divided into subgroups called *strata*, which share a common characteristic (here, the faculty). Instead of drawing a simple random sample from the entire population, we sample from each stratum proportionally to its size in the population.

This approach ensures that important subgroups are represented in the sample, reducing the risk that some groups are over- or underrepresented. It is particularly useful when the variable of interest may differ systematically between strata. In our example, study habits might vary across faculties, so sampling proportionally ensures that the collected data better reflects the diversity of the full student population.

(a) Calculate the number of students to sample from each faculty to ensure proportional representation in a sample of 500.

(b) Write an algorithm to generate a stratified random sample of 500 students. The input should be the list of students grouped by faculties as a list of pairs $faculties = [(facultyName, studentList)]$, and the size of the desired sample $sampleSize$ (here 500), and the output should be a stratified sample of the associated size. You can reuse the prevous algorithm and call it as a function $systematicSampling(studentsList, strataSampleSize)$.

(c) Why is this approach better than simple random sampling for ensuring representation across faculties?

(a) Sample sizes:

Faculty	Sample Size
Science	150
Humanities	100
Engineering	200
Business	50

(b) Pseudocode:

```
Algorithm 2: Stratified Sampling (Proportional)
Input: The list of students grouped by faculty:
       $faculties = [(facultyName, studentList)]$, and the total
       sample size $sampleSize$ (here 500)
Output: Sample of students of size sampleSize, stratified
        proportionally across faculties
sample ← empty list;
foreach facultyName, populationSize ∈ faculties do
    proportion ← populationSize / sum of all populations;
    strataSampleSize ← round(proportion * sampleSize);
    studentsList ← getStudents(facultyName);
    facultySample ← systematicSampling (studentsList,
      strataSampleSize);
    sample ← sample ∪ facultySample;
end
return sample;
```

(c) Stratified sampling ensures that each faculty is proportionally represented in the sample. Simple random sampling might accidentally over- or under-represent certain faculties.

(3) Non-Response Bias

(a) Suppose only 60% of the sampled students respond. Propose a method to adjust your analysis to account for non-response bias.

> To mitigate the bias that might be induced from missing answers, we can compare the demographic characteristics (e.g., faculty, year of study) of respondents and non-respondents. Use post-stratification to weight the responses of underrepresented groups more heavily in the analysis, or conduct follow-up surveys targeting non-respondents.

Exercise 2 - Data Collection and the Curse of Dimensionality

In this exercise, we explore the challenges of high-dimensional data collection and propose strategies to address them. The issue is known as the "curse of dimensionality": as the number of dimensions (variables) increases, the data becomes increasingly sparse, making it difficult to:

- Find meaningful patterns or relationships.
- Generalize results to new data (risk of overfitting).
- Apply traditional statistical or machine learning methods effectively.

We consider the following scenario: A research team wants to collect data to predict student academic performance based on a wide range of factors. They propose measuring 50 variables, including:

- Demographic information (e.g., age, gender, socioeconomic status)
- Behavioral data (e.g., library visits, extracurricular activities)
- Digital footprints (e.g., online study platform usage, social media activity)
- Psychometric scores (e.g., stress levels, motivation)

The team plans to collect this data for 1,000 students.

(1) Explain why collecting 50 variables for 1,000 students might lead to the "curse of dimensionality." Use a simple calculation to illustrate the sparsity of the data. How could this affect the reliability of any conclusions drawn from the data?
Hint: what happens if every variable is binary (True or False)?

Therefore, the goal is to reduce the dimensionality of this dataset while preserving the pairwise distances between students as much as possible. Indeed, when analyzing high-dimensional data, the relationships between data points (e.g., students) are often captured by how "close" or "far" they are from each other in the original space. For instance, two students with similar study habits, socioeconomic backgrounds, and academic performance will be "close" in the 50-dimensional space, while two students with very different profiles will be "far" apart. If we reduce the dimensionality (e.g., from 50 to 10 variables), we want to ensure that these relationships are preserved.

The *Johnson-Lindenstrauss Lemma* states that a set of points in a high-dimensional space can be embedded into a lower-dimensional space in such a way that distances between the points are nearly preserved. Specifically, given $0 < \epsilon < 1$, an integer n, a set X of n points in $\mathbb{R}^D$, and an integer $d > \frac{8 \ln n}{\epsilon^2}$, there exists a (linear) map $f : \mathbb{R}^D \to \mathbb{R}^d$ such that for all pairs of points u, v in X,

$$(1-\epsilon)\|u-v\|^2 \leq \|f(u)-f(v)\|^2 \leq (1+\epsilon)\|u-v\|^2$$

where $\|\cdot\|$ is the Euclidean norms (or L_2 norm) in the appropriate space.

(2) Suppose you want to preserve the pairwise distances between students with an error margin of $\epsilon = 0.3$. Calculate the minimum number of dimensions d required to embed the dataset according to the Johnson-Lindenstrauss Lemma. Use the approximation $d \approx \frac{8 \ln n}{\epsilon^2}$, you can also appropriate $\ln n$ as $\frac{2 \log n}{3}$.

(3) If you reduce the dataset to $d = 10$ dimensions, what is the maximum error margin ϵ you can guarantee for the 1,000 students? Use the same approximation.

Assume you have implemented a dimensionality reduction technique and reduced the dataset to 10 dimensions. You now want to perform a k-nearest neighbors (k-NN) search on this reduced dataset. Let us recall how the k-NN algorithm works. Given a dataset and a query point, the algorithm identifies the k closest points (neighbors) to the query in the dataset, based on a distance metric (usually Euclidean distance). The label or value of the query point is then determined by the majority vote (for classification) or the average (for regression) of its k neighbors.

(4) How might the error margin ϵ affect the results of your k-NN search? Discuss the implications for the accuracy of your search.

(5) Discuss one trade-off between collecting more variables and collecting more samples (e.g., more students). How would you balance this trade-off in practice?

(1) With 50 variables and 1,000 students, the data space is sparse. For example, if each variable is binned into 2 categories, the total possible combinations are $2^{50} \simeq 10^{15}$. This means that the proportion of the space covered by the sample is $\frac{10^3}{10^{15}} = 10^{-12}$. The sparsity makes it difficult to find meaningful patterns or generalize results.

(2) For $n = 1000$ students and $\epsilon = 0.3$, the minimum number of dimensions d is:

$$d \approx \frac{8 \ln 1000}{(0.3)^2} \approx 614$$

Thus, you need at least $d = 614$ dimensions to guarantee the error margin.

(3) For $d = 10$ dimensions and $n = 1000$ students, rearrange the formula to solve for ϵ:

$$\epsilon \approx \sqrt{\frac{8 \ln n}{d}} = \sqrt{\frac{8 \ln 1000}{10}} \approx 2.35$$

This means that with $d = 10$, you can only guarantee an error margin of $\epsilon \approx 2.35$, which is not very precise.

(4) The error margin ϵ affects the k-NN search by distorting the distances between points in the reduced space. If ϵ is large (e.g., 2.35), the distances between students in the 10-dimensional space may not accurately reflect their true distances in the original 50-dimensional space. This could lead to incorrect neighbor assignments. A student who is truly far from a query student might appear close in the reduced space, leading to incorrect neighbor selection. Conversely, a student who is truly close might appear far, causing the algorithm to miss relevant neighbors. As a result, the accuracy of the k-NN search could be significantly reduced, leading to incorrect predictions or classifications.

(5) Collecting more variables increases the risk of sparsity and noise, while collecting more samples improves statistical power but may be costly. A balanced approach is to prioritize variables with strong theoretical or empirical relevance and ensure a sufficient sample size for reliable analysis.