Exercise Collection

September 11, 2024

Contents

1 Basic Mathemathics and Python for Machine Learning

1.1 Linear Algebra, Calculus and Probability

1. **Linear Algebra: Vectors and Matrices.** *Pacients, symptoms, and treatments:*[1] Assume you receive a dataset from a hospital with m patients, n symptoms, and p treatments. Your data is organized in two matrices $S \in \mathbb{R}^{m \times n}$, and $T \in \mathbb{R}^{m \times p}$ such that

$$S_{ij} = \begin{cases} 1 & \text{if patient } i \text{ has symptom } j \\ 0 & \text{otherwise} \end{cases}$$

$$T_{ik} = \begin{cases} 1 & \text{if patient } i \text{ was treated with } k \\ 0 & \text{otherwise.} \end{cases}$$

   (a) What is the meaning of the second column of S? And what about row 500 of T?

   (b) We define the vector of all ones as $\mathbf{1}$ and the transpose of a matrix A as A^T. Describe in plain English the following quantities, and, further, mention dimensions and entrywise expressions:

      i. $S\mathbf{1}$.

      ii. $S^T\mathbf{1}$.

      iii. S^TS.

      iv. SS^T.

   (c) Consider matrix $P \in \mathbb{R}^{n \times p}$ where P_{jk} is the total number of patients with symptom j that received treatment k. Express P in matrix notation as a function of matrices S and T.

   (d) How would your conclusions change if the encoding of the binary variables changed from $\{0, 1\}$ to $\{-1, 1\}$?

2. **Calculus.**[2]

   (a) *Derivative computation:* Compute the derivative of

      i. $f(x) = 3x$;

      ii. $f(x) = x^x$.

   (b) *Meaning of zero derivatives:* What is the meaning of $f'(x) = 0$ for some x? Give an example of a function f and a location x for which this might hold.

   (c) *Geometrical interpretation:* In Python, plot the function $y = f(x) = x^3 - \frac{1}{x}$ and plot its tangent line at $x = 1$ and at $x = 2$.

[1] Exercise from Boyd, S., Vandenberghe, L., Introduction to Applied Linear Algebra, available here.

[2] Exercises partially from the Dive into Deep Learning book.

3. **Probability.**

   (a) The joint probability of two events, E1 and E2, is the probability of occurrence of event E1 or event E2.

   ○ True
   ○ False

   (b) Consider the joint probability density of two continuous variables x, y, denoted by $p(x, y)$. The marginal probability distribution $p(x)$ is obtained by dividing $p(x, y)$ by $p(y)$.

   ○ True
   ○ False

   (c) A probability density function (PDF) can take on values greater than 1.

   ○ True
   ○ False

   (d) In probability, if two events A and B are mutually exclusive, $A \cap B = \emptyset$, then the probability of either event occurring is the product of their individual probabilities.

   ○ True
   ○ False

   (e) *Gaming a coin flip in Python.*[3] Suppose that we stumbled upon a real coin for which we did not know the true $P(\text{heads})$. To investigate this quantity with statistical methods, we need to (i) collect some data; and (ii) design an estimator. Data acquisition here is easy; we can toss the coin many times and record all of the outcomes.

   i. Write a function in Python that given a value of $p = P(\text{heads})$ and the number of tosses N, returns a vector with the outcomes of the tosses. *Hint: try out* `random.random()`.

   ii. As you might have guessed, one natural estimator is the fraction between the number of observed heads by the total number of tosses, e.g., $\hat{p} = \frac{\#n_H}{\#N}$, where $N = n_H + n_T$. Try a $p = 0.6$ and test estimates for $\hat{p}$ while varying N in $\{2, 5, 8, 20, 100, 1000, 10,000\}$.

   iii. Each time you run this sampling process, you will receive a new random value that may differ from the previous outcome. Dividing by the number of tosses gives us the frequency of each outcome in our data. Note that these frequencies, like the probabilities that they are intended to estimate, sum to 1. How is $\hat{p}$ varying as you vary N? What is the relation with p?

   iv. For what N are you sure you should bet on heads in this coin flip?

   (f) *Bounds:* Given two events A and B, and their probabilities $P(A)$ and $P(B)$, compute upper and lower bounds on $P(A \cup B)$ and $P(A \cap B)$. *Hint: Graph the problem with Venn diagrams*.

1.2 Python basics

1. **Python basics.**

[3] Exercises from the Dive into Deep Learning book.

(a) *List comprehension:* Write a Python function that receives a list of integers and returns a list with the squares of the even numbers in the input list.

(b) *Dictionary comprehension:* Write a Python function that receives a list of strings and returns a dictionary with the length of each string in the input list.

(c) *Lambda functions:* Write a Python function that receives a list of integers and a function and returns a list with the result of applying the function to each element of the input list.

(d) *Map and filter:* Write a Python function that receives a list of integers and returns a list with the squares of the even numbers in the input list using `map` and `filter`.

(e) *Reduce:* Write a Python function that receives a list of integers and returns the product of the elements in the list using `reduce`.

1.3 Python for Machine Learning

1. **Python for Machine Learning.**

(a) *Numpy:* Write a Python function that receives a list of integers and returns a numpy array with the squares of the even numbers in the input list.

(b) *Read csv file with Numpy:* Write a Python function that receives the path to a csv file and returns a numpy array with the data in the file.

(c) *Write csv from Numpy:* Write a Python function that receives a numpy array and a path and writes the data in the numpy array to a csv file.

(d) *Matplotlib:* Write a Python function that receives a list of integers and plots the squares of the even numbers in the input list.

(e) *Matplotlib advanced features:* Write a Python function that receives a list of integers and plots the list of even numbers and the squares of the even numbers in the input list with a gray and a red dashed line, and star markers at each data point. Add a title to the plot and label the x and y axes. Add a legend to the plot with the labels "Even numbers" and "Squares of even numbers". Save the plot to a file named `plot.pdf`.

(f) *Pandas:* Write a Python function that receives a list of strings and returns a pandas DataFrame with the length of each string in the input list.

(g) *Read a csv file with Pandas:* Write a Python function that receives the path to a csv file and returns a pandas DataFrame with the data in the file.

(h) *Read a spreadsheet file with Pandas:* Write a Python function that receives the path to a file, identifies if it is a csv, tsv, or xls file and returns a pandas DataFrame with the data in the file.

(i) *Scikit-learn:* Write a Python function that receives a list of integers and returns a numpy array with the squares of the even numbers in the input list using scikit-learn.

(j) *Working with datasets with Pandas:* Browse the UCI Machine Learning Repository. Look for the Iris dataset and write a Python function that loads the dataset using Pandas. Note: Install the `ucimlrepo` package with conda or pip to access the UCI datasets programmatically. Access the Iris dataset with the code below.

Exercises

```
from ucimlrepo.datasets import load_dataset
iris = load_dataset('iris')
```

i. Using Pandas, write a function that receives the Iris dataset and returns the number of samples and features in the dataset.

ii. Write a function that receives the Iris dataset and returns the mean, standard deviation, and quartiles of the sepal length.

iii. Write a function that returns the data points where the sepal length is greater than the mean sepal length in the Iris dataset.