Beryl Sin

PeopleSoft ID: 4522433

01/11/2024

Problem Set 1

1. Conceptual question: propose one new regression problem (one that we have not discussed in class) that you can solve with machine learning, and describe how you would go about solving it.

Problem: Predict the number of passengers that take the subway in NYC

Please give a short answer (text) to the following;

* 1. What features (x) would you use?
     1. Fare price
     2. Date
     3. Time
     4. Station name
  2. What would the labels (y) be?
     1. Number of passengers
  3. How would you collect data?
     1. Track the number of swipes/taps on the card readers from a station each day.
  4. Why might the problem turn out to be challenging?
     1. There may be other important factors outside of the features listed that may influence the number of passengers that take the subway
     2. Not enough data may affect results

1. Repeat question 1 for a classification problem.

Problem: Is this an apple?

* 1. What features (x) would you use?
     1. Color
     2. Size
     3. Shape
     4. Texture
  2. What would the labels (y) be?
     1. Apple
     2. Not apple
  3. How would you collect data?
     1. Use an open-source fruit image dataset (here is a link to one: https://www.vicos.si/resources/fids30/)
  4. Why might the problem turn out to be challenging?
     1. Not enough important factors were considered
     2. Outliers can impact results

1. Basic operations
   1. See ps1.py for code.
   2. See ps1.py for code.
   3. See ps1.py for code. See ps1-3-c-1.png and ps1-3-c-2.png for output images.

The histogram for vector x does look like a Gaussian distribution.

The histogram for vector z does look like a uniform distribution.

* 1. See ps1.py for code.



* 1. See ps1.py for code.



Adding a constant to a long vector without using a loop is more efficient.

* 1. See ps1.py for code.

A black screen with white text

Description automatically generated

The number of retrieved elements are different each time the same lines of code is ran because the function (in the code, np.random.uniform) used to generate vector z uses random numbers from a normal distribution. These random numbers change every time the function is called, which affects vector y, which is determined by how many positive elements are less than 1.5 in vector z.

1. Linear Algebra
   1. See ps1.py for code, results:  
      A screen shot of a computer

      Description automatically generated
   2. See ps1.py for code, results:

A black background with white text

Description automatically generated

* 1. By hand:

A math equations on a graph paper

Description automatically generated

See ps1.py for code, results using the norm function in python:

A black background with white text

Description automatically generated

1. Splitting Data
   1. See ps1.py for code, results:

A screenshot of a computer

Description automatically generated

* 1. See ps1.py for code.
  2. See ps1.py for code.
  3. See ps1.py for code, results:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A computer screen shot of a black screen

Description automatically generated

I did not get the same submatrices each time I split X because the shuffle function used to partition the matrices mixes the row indices randomly so that the rows used for the training set/testing set differs every time the function is called.