
Problem assignment 1

Due: Thursday, January 24, 2019

Problem 1.

Install Matlab on your computer or access it in one of the CSSD labs.

Problem 2. Exploratory data analysis

In this problem we will explore and analyze the dataset *pima.txt* provided on the course web page. To do the analysis you will need to write short programs. Keep the code you write for future problem sets.

The *pima.txt* is described in the file *pima_desc.txt* The dataset consists of 8 attributes and a binary attribute defining the class label, the presence of diabetes. Data entries are organized in rows such that attributes come first and the class label is last. Answer the following questions with the help of Matlab:

- (a) What is the range of values (minimum and maximum value) for each of the attributes?
- (b) What are the means and variances of every attribute.
- (c) Split *pima.txt* data into two data subsets - one that includes only examples with class label "0", the other one with class "1" labels. Hint: use Matlab's function *find* to split the data. Calculate and report the mean and standard deviations of each attribute (columns 1-8) on these two subsets. Analyze the means and standard deviations of attribute values and select the attribute you think should be most helpful in discriminating the two classes. Include the attribute name in the report and explain why you think the attribute is the best for discriminating the two classes.

While the analysis using basic statistics as performed above conveys a lot of information about the data and lets us make some conclusions about the importance of attributes for prediction or their mutual relation, it is often very useful to inspect the data also visually and get more insight into various shapes and patterns they hide. In the following we will inspect the data using histograms and 2D scatter plots.

- (d) **Histogram** gives us more information about the distribution of attribute values. Write (and submit) a Matlab function *histogram_analysis* that takes the data for an attribute (as a vector) and plots a histogram with 20 bins using Matlab's hist function.
- (e) Analyze attributes in the data using the new histogram function. Answer the following questions: Which histogram resembles most the normal distribution? In your report show at least two histograms, including the choice you picked as the most normally distributed attribute.
- (f) Histogram analysis function you wrote in part (d) lets you plot the distribution of values for any input data. So we can use it to look at attribute distributions for class 1 and class 0 individually and compare them. Similarly to part (c) divide pima dataset into two datasets, one with instances corresponding to class 0 and the other one corresponding to class 1. For each attribute in columns 1 and 8 plot two histograms of the attribute values, one for class 1 and the other one for class 0. Compare the two histograms for each attribute. Based on the pairs of histograms choose an attribute you think should be most helpful in discriminating the two classes. Include the attribute name and the histograms for that attribute for class 1 and class 0 in the report. Explain why you think the attribute is the best.
- (g) **2D Scatter plots** let us inspect the relations between pairs of attributes. Write (and submit) a function *scatter_plot* that takes pairs of values for two attributes and plots them as points in 2D (use Matlab function scatter to do the plot). Analyze the pairwise relations between 8 nontarget attributes in the pima dataset using the function. Answer the following questions. What scatter plot would you expect to see for the two dimensional space if the two attributes are independent and random? Do you see any interesting non-random patterns among the pairs? Include two scatter graphs you think show some interesting dependences or patters. Explain why you think these are interesting? Do not forget to include with every plot the corresponding attribute names.

Problem 3. Data preprocessing

Before applying learning algorithms some data preprocessing may be necessary. In this problem we explore three preprocessing methods: transformation of categorical values to (safe) numerical representation, normalization of of continuous values, and discretization of continuous values.

- (a) Assume you have an attribute with 8 categorical values {brown, blue, white, red, yellow, orange, green, black}. Devise one-hot encoding of the values and explain in the report how values are mapped. Use the mappings to convert the following vector of attribute values to one hot representation and include the results in the report:

  - red
  - black
  - yellow
  - red
  - green
  - blue
  - blue

- (b) Write (and submit) function *normalize* that takes an unnormalized vector of attribute values and returns the vector of values normalized according to the data mean and standard deviation. One way to calculate the normalized value is to apply the following formula:

$$x_{\text{norm}} = \frac{x - \mu_x}{\sigma_x}.$$

  where x is an unnormalized value, μ_x is the mean value of the attribute in the data and σ_x its standard deviation. Test your function on attribute 3 of the pima dataset. Report normalized values of the attribute 3 for the first five entries in the dataset.

- (c) Write (and submit) a function *discretize_attribute* that takes a vector of attribute values, and number k (number of bins) as inputs and assigns each value to one of the k bins. Bins are of equal length and should cover the full range of values that are determined by the min and the max operations on the vector. Every bin is given a numerical label such that the smallest value is in bin 1 and the largest attribute value is in bin k. The bin label represents the result of discretization. Test your function on attribute 3 of the pima dataset. Assume we use 10 bins. Report new (discretized) values of the attribute 3 for the first five entries in the dataset.

Problem 4. Splitting data into training and testing sets

In this problem we write a function supporting the splitting of the dataset into the training and testing sets.

Write (and submit) function *divideset* that takes the full dataset (represented as a matrix) and the probability p_{train} reflecting the proportion of examples (rows in the matrix) to be assigned to the training set and returns the training and testing sets. Everytime the function is run it should return training and testing sets with randomdomly assigned rows such that sizes of the train sets and the test sets match respectively. Hint: use function randperm(n) that lets one randomly shuffle order of numbers from 1 to n. Please verify the performance of the function on the pima dataset. More specifically, every time you run the function the sizes of the training and testing datasets should match, but the content should be random and different.

Problem 5: Matrix operations practice problems

Let us assume:

$$A = \begin{bmatrix} 1 & 2 & 5 \\ 3 & 4 & 6 \end{bmatrix}$$

and

$$B = \begin{bmatrix} 7 & 1 & 9 \\ 2 & 2 & 3 \\ 4 & 8 & 6 \end{bmatrix}$$

and

$$C = \begin{bmatrix} 8 & 6 & 5 \\ 1 & -3 & 4 \\ -1 & -2 & 4 \end{bmatrix}$$

Please calculate:

- (a) A^T
- (b) B^{-1}
- (c) $B + C$
- (d) $B - C$
- (e) $A * B$
- (c) $B * C$
- (c) $B * A$