Assigned: 2/1/2017
Due: Fri. 2/17/2017, 11:59pm

Instructions: This project will cover some questions related to topics of data preprocessing, dimensionality reduction, clustering, and model evaluation.

Submission Requirements: Your answers must be computer generated (including text and diagrams). Your final document submission should include text responses to questions and description of your efforts, tables, R/Matlab/Python code used to calculate answers, and figures. As well as the code to carry out the work.

Formatting of submissions: The following methods are acceptable ways to submit your assignment:

- {Word + code}, {Open Office + code} → PDF
  This option may require taking screenshots or printing figures created in R/MATLAB/Python and importing them into the word processing software. Additional code and results should also be inserted into the word documents.
- If you are using R consider:
  - Rmd → PDF, Rmd → HTML
    Use `knitr` or `rmarkdown` to collect all text responses, figures, tables, and code in the Rmarkdown file and process it to produce a PDF or HTML file.
  - Snw → PDF, Stex → PDF
    Use `Sweave` to collect all text responses, figures, tables, and code in the Snw file and process it to produce a PDF.
- If you are using MATLAB consider:
  - .m file + markup, publishing matlab code → HTML
    Incorporate your answers directly into your MATLAB code (code, comments, results), publish the code creating an HTML file.
  - .m files + *Your favorite document editor*
    Answer your questions in your text editor, embedding code and results from matlab .m file
- If you are using Python consider:
  - iPython → HTML
  - LaTeX + Sphinx → PDF / HTML

I highly recommend following the ideas of reproducible research and embed the code, images, and results directly into the text using packages like `knitr/rmarkdown`, `Sweave`, `publish`, or `iPython` (other packages follow this practice using Latex and reStructureText as well and are open for you to use).

If you want to follow the style of the R introduction documents on Canvas (e.g., introA.html, introB.html, etc.), please use the provided CSS style file `min.css`), and follow instructions provided in R Studio documentation and the `rmarkdown` package. There are also a number of style and code highlighting styles available using `Bootstrap` themes.

For this project you are allowed to work in **groups of up to 3**. Please sign-up on Canvas into your group (e.g., GroupRho, GroupTau, etc.). If you are working individually on a project, please place yourself in a group by yourself.

Name your main submission files as *P2_GroupName*, create a zip-file called *Project2_GroupName.zip* and submit on Canvas. For example, if I was using R, and part of GroupChi, I would submit either:

- *P2_GroupChi.Rmd*, *P2_GroupChi.pdf*, or
- *P2_GroupChi.Rmd*, *P2_GroupChi.html*, or
- *P2_GroupChi.Snw*, *P2_GroupChi.pdf*

along with any other supplemental `.R` files I created in *Project2_GroupChi.zip*.

Questions:

List your project GroupName and all group members names.

1. (20 points) Write your own general-purpose functions to perform min-max normalization and z-score normalization (using standard deviation or mean absolute deviation). Do not just use functions available in R, PYTHON or MATLAB. *Think about how to ensure it generalizes. For example, in* R *will it work on vectors, matrices and data frames. In* MATLAB*, will it work on vectors and matrices. For min-max normalization, typically you want to scale to* $[0, 1]$ *(defaults), but you should be able to pass in other limits.*

   The function `minmaxNorm` should take four arguments

   - trData- the training data (use to establish the data properties for normalizaiton)
   - teData- the testing data (if supplied) to also be normalized according the the same data properties
   - minV- minimum value of new range
   - maxV- maximum value of new range

   The function `zscoreNorm` should take three arguments

   - trData- the training data (use to establish the data properties for normalizaiton)
   - teData- the testing data (if supplied) to also be normalized according the the same data properties
   - madFlag- boolean flag if positive use mean absolute deviation instead of standard deviation.

2. (8 points) Data Mining Book: 3.6(a-c)
   Report the normalized values in a table.

3. Load the IRIS data set available from the UCI Machine Learning data repository, http://archive.ics.uci.edu/ml/.

   Using the data for *petal length*, answer the following questions:

   (a) (4 points) Use your min-max normalization function with a range $[-1.0, 1.0]$, to what values would $\{1.95, 3.1, 5.68$ and $6.2\}$ transform?

(b) (4 points) Use your z-score normalization function to determine what values {1.95, 3.1, 5.68, and 6.2} would transform to?

(c) (2 points) Comment on which method is preferred for this data, and why?

4. Consider the following data set of with 5 samples and 3 variables:

	A	B	C
x_1	1.4	1.3	2.9
x_2	1.8	1.4	3.2
x_3	1.3	1.2	2.9
x_4	0.9	3.5	3.1
x_5	1.5	2.1	3.3

You have a new data point $x = (1.25, 1.78, 3.01)$.

(a) (5 points) Calculate and present the distance between the new data point and each of the points in the data set using Manhattan distance, Euclidean distance, Minkowski distance ($\lambda = 3$), supremum distance, and cosine similarity.

(b) (5 points) Normalize the data using min-max normalization to be between 0 and 1. What is the Euclidean distance between the new data point and $x_1, \ldots, x_5$.

5. K-means Clustering
Perform k-means clustering **manually** with k=2 on the example data given below of $n = 8$ samples over $p = 2$ features.

Sample	X_1	X_2	Initial Groups
1	1	4	1
2	1	3	1
3	0	4	2
4	2	5	2
5	5	1	1
6	6	2	2
7	4	0	1
8	5	2	2

(a) (2 points) Plot the sample data.

(b) (4 points) Assign samples to be the initial groupings given in the table. Compute and report the centroid for each cluster.

(c) (4 points) Assign each sample to the centroid to which it is closest (Euclidean distance). Report the cluster labels for each observation.

(d) (20 points) Repeat (b) and (c) until the clusters remain stable.

(e) (2 points) Plot the sample data colored by cluster labeling and adding centroid points.

6. Hierarchical Clustering

Suppose you have 5 samples, for which the dissimilarity matrix is shown below:

$$D = \begin{bmatrix} -- & 0.3 & 0.4 & 0.7 & 0.6 \\ 0.3 & -- & 0.5 & 0.8 & 0.2 \\ 0.4 & 0.5 & -- & 0.45 & 0.4 \\ 0.7 & 0.8 & 0.45 & -- & 0.35 \\ 0.6 & 0.2 & 0.4 & 0.35 & -- \end{bmatrix}$$

That is, the distance between the first and second sample is 0.3; the distance between the first and fourth sample is 0.7.

(a) (12 points) Trace running through hierarchical clustering **manually** with complete linkage and sketch the dendrogram. Estimate the heights in the dendrogram from the dissimilarity distances.

(b) (12 points) Repeat (a), with single linkage clustering

(c) (6 points) Use the dendrogram from (a) and (b), cut the dendrograms to form three clusters. Which samples are in each cluster?