Homework#1

Due on Feb 29th, 2016

11:00pm

Submit Via NYUClasses

Submit a Microsoft word document that documents in details all your work and submit the code associated with this homework with a readme file on how to run your code and the dataset you used).

Learning Outcomes

- Most widely used Data pre-processing and dimensionality reduction algorithms
  - Replace Missing Values (min, max, or a specific value)
  - Missing Values Ratio
  - Low Variance Filter
  - High Correlation Filter
  - Random Forests Ensemble Trees
  - Principle Component Analysis
  - Singular Value Decomposition
  - Backward Feature Elimination
  - Forward Feature Selection
- Hands on real world datasets
- Understanding Principal Component Analysis and its applications
- Understanding Singular Value Decomposition and its applications

1. Widely Used Data Pre-processing Techniques (60pts)

1.1 Step 1: Read the following article published recently at kdnuggets:

http://www.kdnuggets.com/2015/05/7-methods-data-dimensionality-reduction.html

1.2 Step 2: Read the following white paper

https://www.knime.org/files/knime_seventechniquesdatadimreduction.pdf

1.3 Step 3: pick a dataset of your choice.
You may use datasets from the following sites (you are not restricted to these sources):

- https://www.kaggle.com/
- http://kdd.org/kdd-cup/view/kdd-cup-2009/Data
- https://aws.amazon.com/datasets/
- http://archive.ics.uci.edu/ml/

1.4 Step 4: Briefly describe in your own words the 9 methods mentioned below in no more than one paragraph per method. (in your own words).
Apply the following (5 of 9) pre-processing and data reduction methods to your dataset If you do more than 5, they will be counted as bonus for 5pts for each additional method.

1. Replace Missing Values (min, max, or a specific value)
2. Missing Values Ratio
3. Low Variance Filter
4. High Correlation Filter
5. Random Forests Ensemble Trees
6. Principle Component Analysis
7. Singular Value Decomposition
8. Backward Feature Elimination
9. Forward Feature Selection

You might use one or a combination of the following tools and API:

Knime (https://www.knime.org/downloads/overview), **RapidMiner**, **Python Libraries** (see appendix), **Weka**, **Java Libraries** (http://java-ml.sourceforge.net/), **Hadoop/Mahout** or combination of some or all to perform the data pre-processing techniques mentioned above on your dataset.

The problem that you will be trying to solve using the dataset your select should be a data science problem (e.g. data classification, data clustering, recommendation engine or mining association rules).
You should be able to evaluate the performance of the model you apply in order to record the effect on performance on each data preprocessing technique. Please stop by my office hours if you have any questions.

1.4 Once you finish your analysis on the pre-processing methods, you will need to document your analysis as a summary in a similar tabular format as the table below:

Pre-processing Method used	Data Reduction Rate	Best threshold / Parameters	Model Accuracy	Notes
Method#1				
Method#2				
….				
….				

…				

2. Understanding PCA and SVD (40pts)

2.1 Explain the concept of PCA in your own words in no more than one paragraph. Research and summarize in half a page other applications of PCA other than dimensionality reduction (cite all your sources).

2.2 Consider the following matrix that represent four users (rows) and their ratings (columns) to products they bought on an e-commerce site.

2 3 1 1
4 4 1 1
0 1 4 4
0 1 2 2

2.3 Apply PCA to reduce the matrix (user/ratings) a lower dimension. Show all your steps.

2.4 Explain the concept of SVD in no more than one paragraph. Briefly list other applications of SVD other than dimensionality reduction.

2.5 Apply SVD process to the same matrix. Explain how you can use the results from SVD to reduce dimensionality of the matrix. Explain the number of concepts learned from this matrix.

2.6 Consider a new User ***X*** who rated four products, his/her query will be [5,0,0,1]

2.7 Use SVD's results you computed previously to find similar users to user **X**.

Refer to ***Chapter 11*** from Mining Massive Datasets (the chapter is posted on NYUclasses).

Appendix

If you decide to use Python, please find attached references: (you don't have to use python library)

Dimensionality Reduction with Scikit-learn

1.Imputation of missing values with example

http://scikit-learn.org/stable/modules/preprocessing.html#imputation-of-missing-values

http://scikit-learn.org/stable/auto_examples/missing_values.html

2.Feature agglomeration with Digit dataset

http://scikit-learn.org/stable/modules/clustering.html#hierarchical-clustering

http://scikit-learn.org/stable/auto_examples/cluster/plot_digits_agglomeration.html#example-cluster-plot-digits-agglomeration-py

3.Feature Selection with Iris dataset (Chi2, tree-based)

http://scikit-learn.org/stable/modules/feature_selection.html#univariate-feature-selection

4.Principal component analysis (PCA)

http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html#sklearn.decomposition.PCA

http://nbviewer.jupyter.org/github/jakevdp/sklearn_pycon2014/blob/master/notebooks/03_basic_principles.ipynb

5. PCA compare with LDA with Iris Dataset

http://scikit-learn.org/stable/auto_examples/decomposition/plot_pca_vs_lda.html#example-decomposition-plot-pca-vs-lda-py

http://scikit-learn.org/stable/modules/generated/sklearn.discriminant_analysis.LinearDiscriminantAnalysis.html#sklearn.discriminant_analysis.LinearDiscriminantAnalysis

6.Singular value decomposition (SVD)

http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html#sklearn.decomposition.TruncatedSVD

7.Random projection

http://scikit-learn.org/stable/modules/classes.html#module-sklearn.random_projection

http://scikit-learn.org/stable/modules/random_projection.html#random-projection

Recommended Environment Setup for Mac:

1. Install pip: https://pip.pypa.io/en/stable/installing/

2. Install Python "NumPy", "SciPy" modules in Terminal: (Example for NumPy)

3. Install Scikit-learn

http://scikit-learn.org/stable/install.html