

# CS 7641 Machine Learning

## Assignment 3

Philip Bale  
pbale3

Due Sunday April 1st, 2018 11:59pm

### Introduction

This assignment explores unsupervised learning and dimensionality reduction. It begins by examining clustering algorithms, specifically k-means and expectation maximization. It then proceeds to cover four dimensionality reduction algorithms: principal components analysis, individual components analysis, randomized projections, and random forests. After running these six algorithms on the original datasets and observing the results, the results are then piped into a neural network learner for further examination.

### Datasets chosen

The datasets chosen were the same datasets chosen for assignment 1. The first dataset is the US permanent visa dataset. This dataset is interesting due to its potential to aid in the visa application process from a cost and time savings potential. It could also enable confidence in those interested in applying for a US permanent visa but doubting their chances of acceptance. At the end of the day, the goal is it to try to determine the application result before time, money, and other resources are spent.

The second dataset is a home sale price prediction dataset taken from an ongoing Kaggle competition. This dataset is interesting for two primary reasons: real-world applicability and participating in a Kaggle challenge. First, modeling home prices is both a difficult and lucrative task. If one can successfully model home sale prices on large sets of data, he/she can make large amounts of money investing in real estate when he/she detects outliers in listed price vs. what it is expected to sell for. This applies to flipping, investing, and remodeling. Second, the dataset is part of an ongoing Kaggle competition that does not have a winning solution yet. By taking part of the competition, the dataset presents the opportunity to work towards a winning solution and advance one's algorithms over time.

## Part 1: Clustering Algorithms

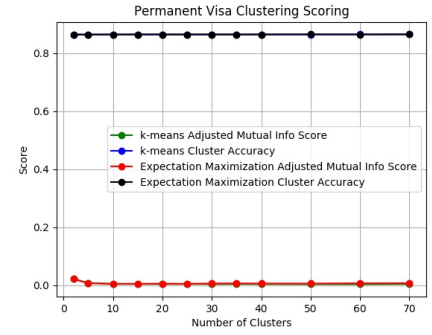
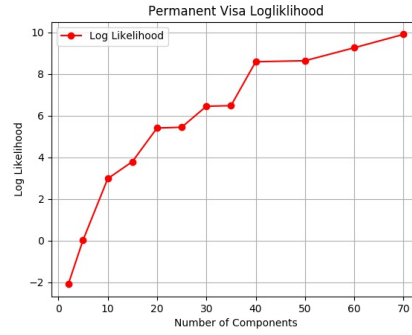
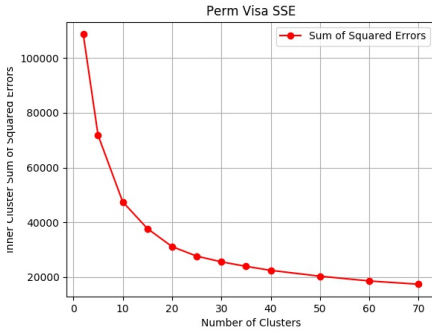
### Introduction

K-means clustering is the first algorithm applied to the datasets and expectation maximization is the second. Both algorithms work by clustering: gathering groups of instances together based upon their features. The rationale is that similar instances will likely be labeled the same way—such as identical visa applications obtaining the same outcome.

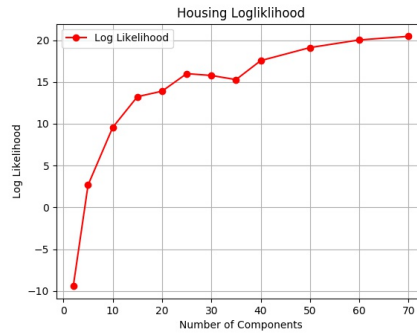
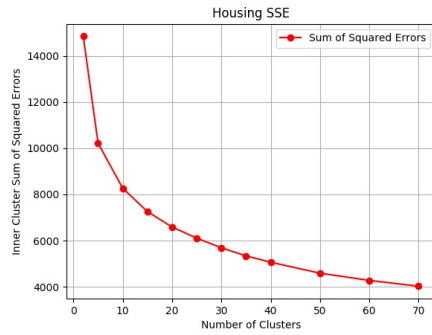
### 1) k-means clustering

#### Overview

K-means works by clustering  $n$  instances into  $k$ -clusters of similarity using least-squares Euclidean distance between the instances. In practice, the algorithm converges on 'mean' for each cluster that is representative of the members of that cluster. A variety of cluster sizes were tested to find the best parameters possible.



Perm Visa Sum of Square Errors for Clus- Perm Visa Log Likelihood vs. # Compo- Perm Visa Scoring for k-means and expectation maximization  
ters vs. # Clusters nents



Housing Sum of Square Errors for Clus- Housing Log Likelihood vs. # Compo- Housing Scoring for k-means and expectation maximization  
ters vs. # Clusters nents

## k-Means Analysis

Text

| Clusters              | 2      | 5     | 10    | 15    | 20    | 25    | 30    | 35    | 40    | 50    | 60    | 70    |
|-----------------------|--------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>PERM VISA</b>      |        |       |       |       |       |       |       |       |       |       |       |       |
| <b>SSE</b>            | 108717 | 71834 | 47453 | 37701 | 31090 | 27611 | 25517 | 23874 | 22410 | 20267 | 18532 | 17331 |
| <b>Log Likelihood</b> | -9.44  | 2.67  | 9.57  | 13.25 | 13.90 | 16.01 | 15.78 | 15.29 | 17.56 | 19.12 | 20.04 | 20.47 |
| <b>k-Means AMI</b>    | 0.022  | 0.008 | 0.005 | 0.005 | 0.004 | 0.005 | 0.004 | 0.004 | 0.004 | 0.004 | 0.004 | 0.004 |
| <b>k-Means ACC</b>    | 0.865  | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 |
| <b>EM AMI</b>         | 0.022  | 0.007 | 0.005 | 0.005 | 0.006 | 0.005 | 0.006 | 0.007 | 0.006 | 0.006 | 0.007 | 0.007 |
| <b>EM ACC</b>         | 0.865  | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 | 0.865 |
| <b>HOUSING</b>        |        |       |       |       |       |       |       |       |       |       |       |       |
| <b>SSE</b>            | 14840  | 10217 | 8265  | 7256  | 6589  | 6106  | 5687  | 5339  | 5063  | 4589  | 4276  | 4024  |
| <b>k-Means AMI</b>    | 0.105  | 0.134 | 0.120 | 0.103 | 0.091 | 0.080 | 0.086 | 0.090 | 0.086 | 0.081 | 0.083 | 0.081 |
| <b>k-Means ACC</b>    | 0.628  | 0.634 | 0.695 | 0.702 | 0.694 | 0.688 | 0.695 | 0.704 | 0.705 | 0.715 | 0.719 | 0.726 |
| <b>EM AMI</b>         | 0.010  | 0.065 | 0.073 | 0.073 | 0.078 | 0.073 | 0.081 | 0.085 | 0.077 | 0.078 | 0.065 | 0.064 |
| <b>EM ACC</b>         | 0.628  | 0.631 | 0.631 | 0.644 | 0.657 | 0.666 | 0.682 | 0.688 | 0.686 | 0.677 | 0.673 | 0.679 |

Table of Housing Data Results for Cluster

## 2) Expectation Maximization

### Overview

Expectation Maximization is the second algorithm applied to the datasets and, similar to k-means, is a clustering algorithm. Expectation Maximization works by iteratively finding the maximum likelihood of parameters leading to a labeling of an instance despite possibly not having all data or parameters. For our examples, we used Scikit-learn's Gaussian mixture

models to implement the Expectation Maximization algorithm. A varying number of mixture components (or number of distributions) were used to determine the best possible parameters for the clustering.

## Expectation Maximization Analysis

Text

# Part 2: Dimensionality Reduction Algorithms

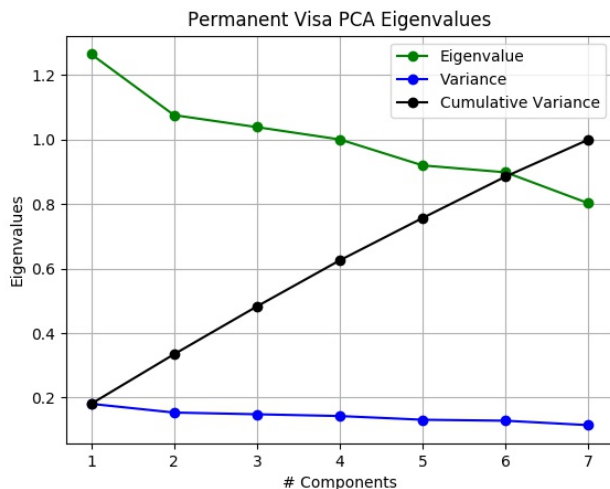
## Introduction

Part 2 deals with dimensionality reduction algorithms. The four algorithms used are principal components analysis, individual components analysis, randomized projections, and random forests. After running the algorithms on both datasets, an analysis is provided on the results.

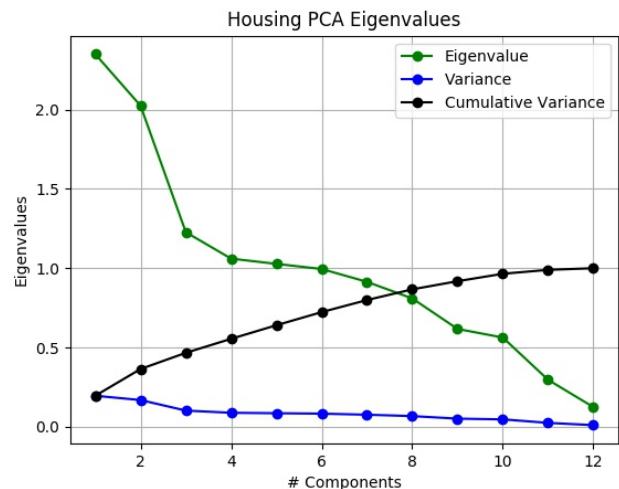
## 1) Principal Components Analysis (PCA)

### Overview

The first dimensionality reduction algorithm, Principal component analysis is a statistics approach to finding vectors that maximize variance and thus help to determine components that are correlated. Each subsequent component is found with the intent to be orthogonal to the preceding component. The resulting eigenvalue matrix from PCA is therefore maximized for covariance.



Permanent Visa Principal Components Analysis



Housing Principal Components Analysis

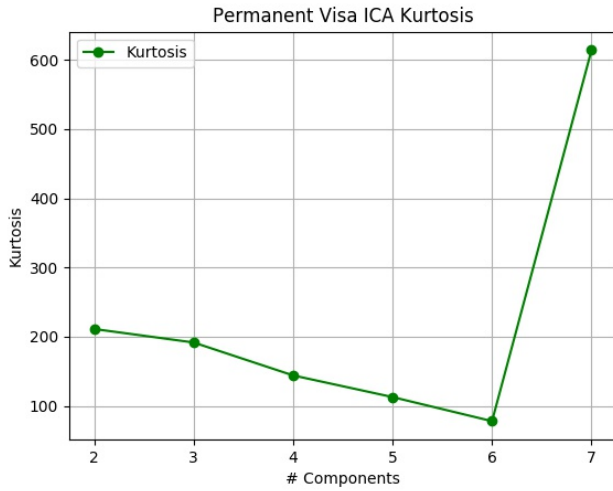
## Analysis

IText

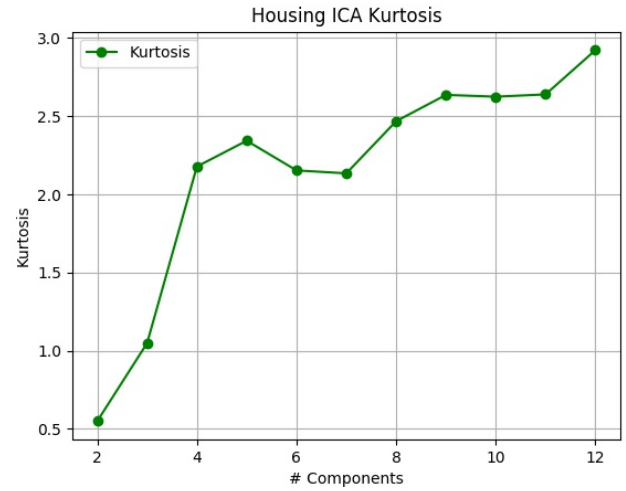
## 2) Independent Components Analysis (ICA)

### Overview

The second dimensionality reduction algorithm, independent components analysis, is an approach to separating a mixture of a data into appropriate subcomponents. As discussed in lecture, a good example of what ICA is used for is the cocktail problem; where one needs to separate various sounds into their sources: a tv show, humans, car noises, etc. Kurtosis is used as a measurement of how gaussian the derived components are.



Permanent Visa Independent Components Analysis



Housing Independent Components Analysis

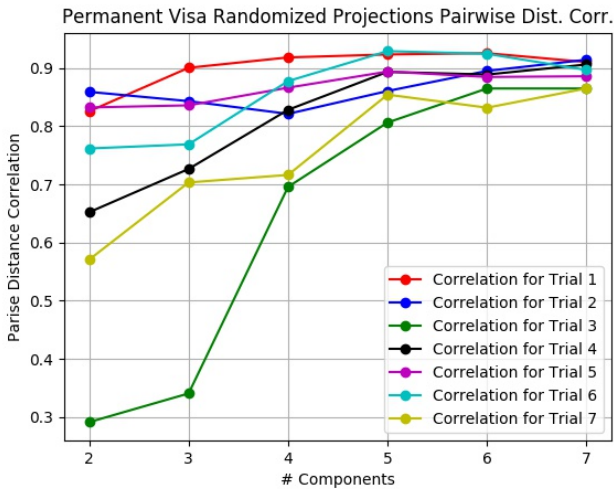
## Analysis

IText

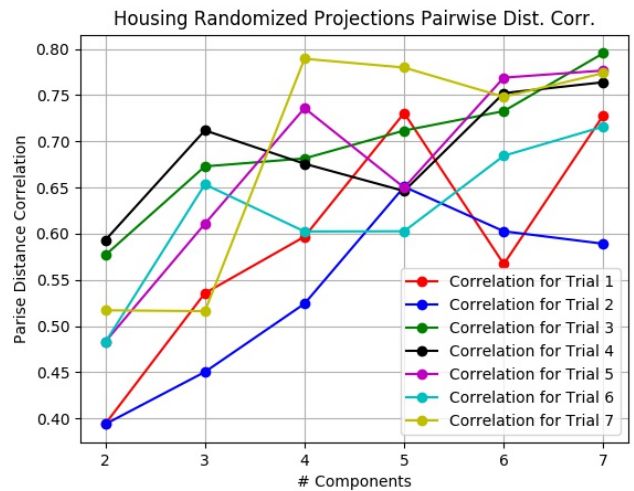
### 3) Randomized Projections

#### Overview

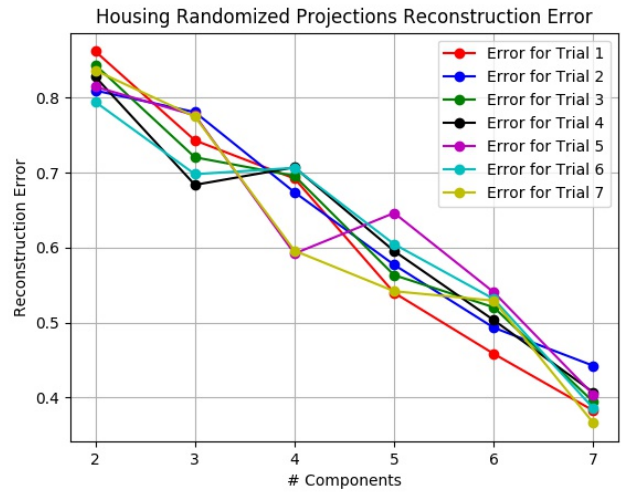
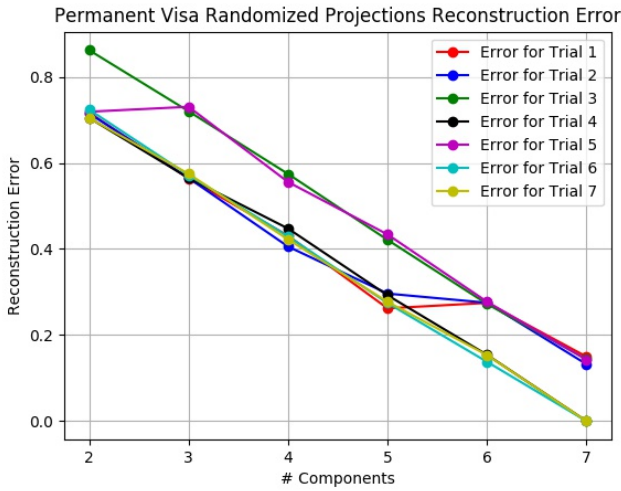
The third dimensionality reduction algorithm, randomized projections, is an approach that randomly generates a projection matrix that attempts to create a lower dimension representation of the data that is approximately accurate to its original state. By varying the number of components to project, we can run various tests on how well the lower dimension data captures the original.



Permanent Visa Randomized Projections Pairwise Correlation



Housing Randomized Projections Pairwise Correlation



Permanent Visa Randomized Projections Reconstruction Error

Housing Randomized Projections Reconstruction Error

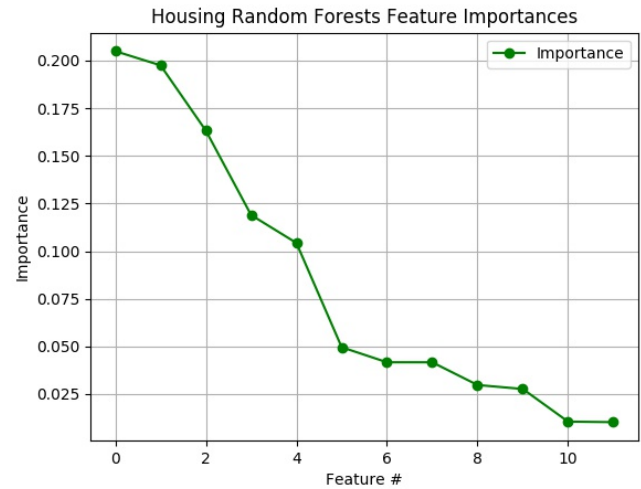
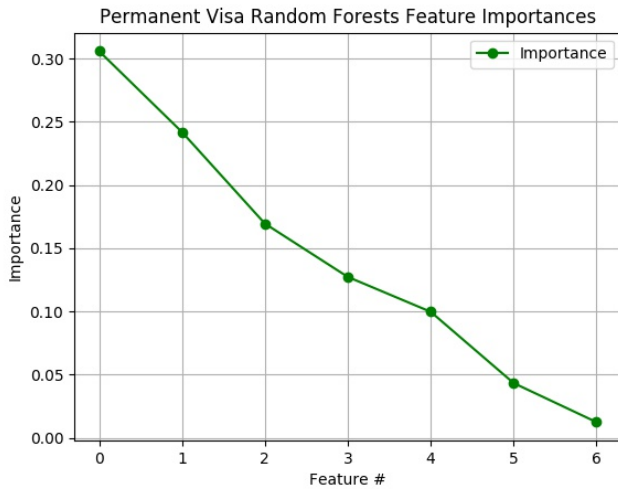
## Analysis

IText

## 4) Random Forest Feature Selection

### Overview

The fourth, and last, dimensionality reduction algorithm, random forest feature selection, is an approach that uses an ensemble of decision trees conditioned on different features. By training the decision tree and observing the impact of each feature by its ability to classify data correctly, we can select the most important features and disregard unimportant features.



Permanent Visa Random Forest Feature Importances (Descending Order)

Housing Random Forest Feature Importances (Descending Order)

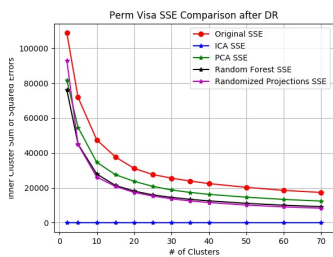
## Analysis

IText

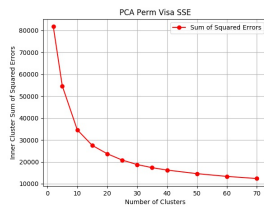
# Part 3: Dimensionality Reduction and Clustering

## Overview

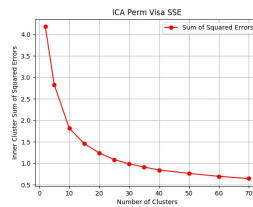
### k-Means after Dimensionality Reduction



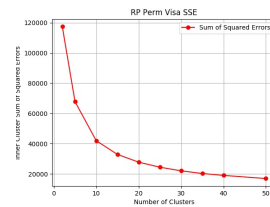
Original Visa Data



PCA



ICA

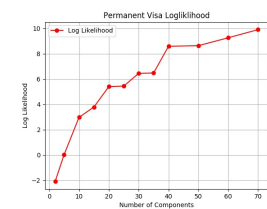


Randomized Projections

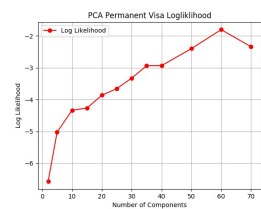
## SSE Analysis

Text

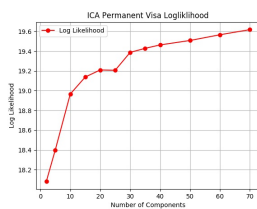
### Expectation Maximization after Dimensionality Reduction



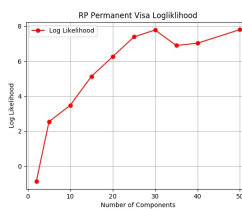
Original Visa Data



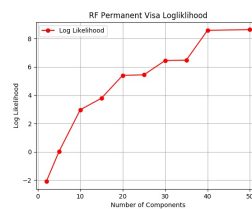
PCA



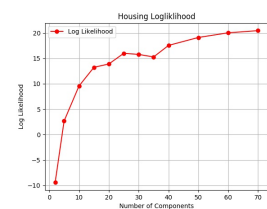
ICA



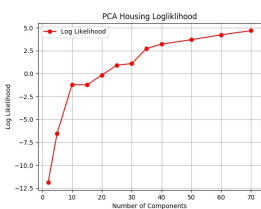
Randomized Projections



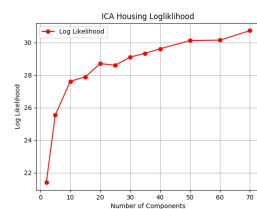
Random Forest



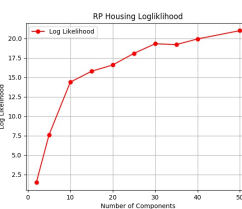
Original Housing Data



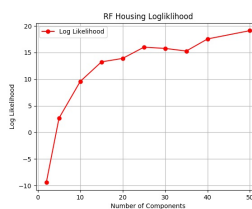
PCA



ICA



Randomized Projections



Random Forest

## PCA Analysis

Text

## Conclusion

Todo conclusion