A REPORT

ON

Aspects of Supervised Dimensionality Reduction Technique

BY

Name of the Student	ID Number
Rachit Agrawal	2016B2A70901P
Daksh Bajaj	2017A5PS1100P

AT

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

APRIL 2020

Supervised Learning

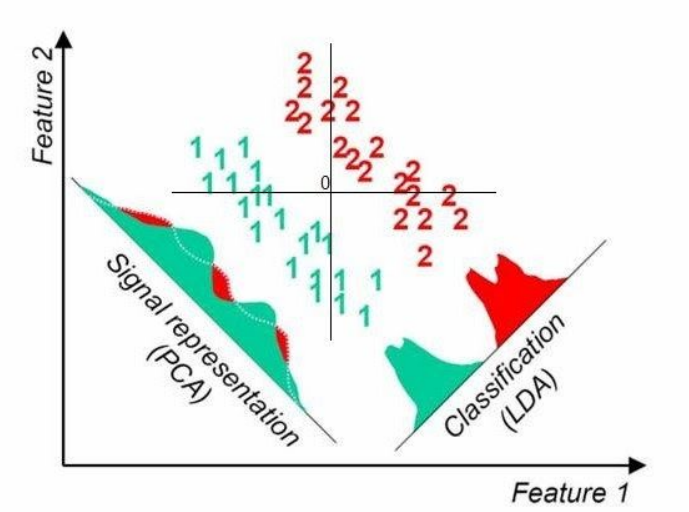
Supervised learning is a model of learning designed to render prediction, provided an unpredictable instance of data.

To learn the regression/classification model, a supervised learning algorithm takes a known set of input datasets and their known responses to the data (output). Afterward, a learning algorithm trains a model to produce a prediction for the answer to new data or test data. The algorithms include linear regression, logistic regression, and neural networks, apart from the decision tree, Support Vector Machine (SVM), random forest, naive Bayes, and k-nearest neighbor.

LDA

Linear discriminant analysis (LDA) is considered the most common linear method of supervised dimensionality reduction.

This is designed to find low-dimensional projections that maximize the separation of classes.



The goal of LDA: Maximize the ratio of the variance between the classes to the variance within the classes.

Multi-class LDA is based on the study of two scatter matrices: Scatter matrix within the class and Scatter matrix between classes

Given a set of samples x1,x2,.....x_n, and their class labels y1,y2,.....y_n:

The within-class scatter matrix is defined as:

$$\mathbf{S}_w = \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu}_{y_i})(\mathbf{x}_i - \boldsymbol{\mu}_{y_i})^T$$

Here, $\boldsymbol{\mu}_k$ is the sample mean of the k^{th} class.

The between-class scatter matrix is defined as:

$$\mathbf{S}_b = \sum_{k=1}^{m} n_k (\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^T$$

Here, m is the number of classes, $\boldsymbol{\mu}$ is the overall sample mean, and n_k is the number of samples in the kth class. Then, multi-class LDA can be formulated as an optimization problem to find a set of linear combinations (with coefficient w) that maximize the ratio of between-class scattering to within-class scattering, as:

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmax}} \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}$$

The solution is given by the generalized eigenvalue problem:

$$\mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w}$$

Generally, at most m-1 generalized eigenvectors are useful to discriminate between m classes.

We have used LDA function for performing Linear Discriminant Analysis, the syntax of which is something like this

Syntax:- lda(formula, data, ..., subset, na.action)

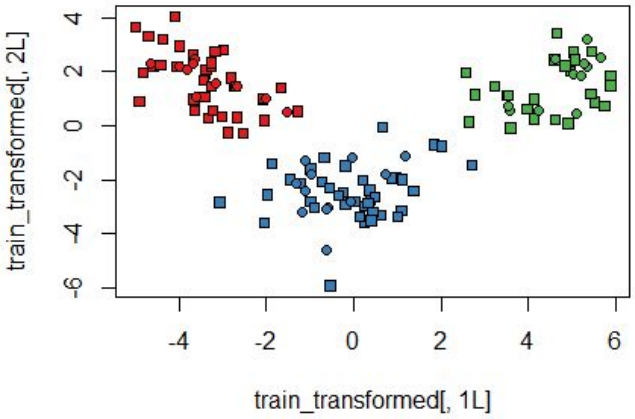


Fig.1 Plot obtained using LDA followed by KNN

Metric Learning:

Distance metric learning (or simply metric learning) aims to construct task-specific distance metrics in a machine learning manner from (weakly) supervised data. The learned distance metric can then be used to perform various tasks (e.g. k-NN classification, clustering, retrieval of information). Metric learning aims at calculating the similarity between samples by using an optimal distance metric for learning tasks.

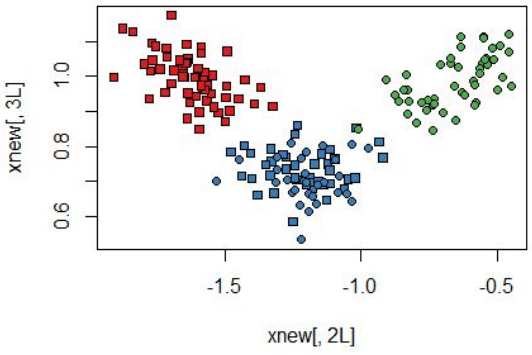


Fig.2 Plot obtained using Metric Learning(RCA) followed by KNN

All currently implemented algorithms learn the so-called Mahalanobis distances in the metric-learn set. The Mahalanobis distance is a Euclidean distance after a linear transformation of the feature space defined by L (recovering the normal Euclidean distance from L as the identity matrix). We have used Relevant Component Analysis(RCA) because it offers a simple yet powerful method to learn this metric.

Syntax:- rca(x, chunks, useD)

Unsupervised Learning:

In unsupervised learning, the learning algorithm is given no labels, leaving it to find meaning in its data on its own. Unsupervised learning can be a goal in itself, (the discovery of hidden patterns in data).

PCA

Principal Component Analysis(PCA) is an unsupervised dimensionality reduction technique used in a dataset to highlight variability and show clear patterns. It is also used to promote the analysis and visualization of data.

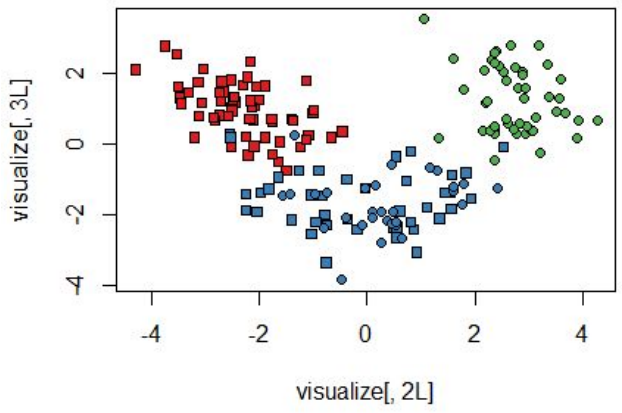


Fig.3 Plot obtained using PCA followed by KNN

We have used the function ***prcomp*** for PCA. This function carries out a principal component analysis on the given data matrix and returns the results as a class object.

Syntax: prcomp(x, scale)

Arguments for prcomp():

- x: a numeric matrix or data frame
- scale: a logical value indicating whether the variables should be scaled to have unit variance before the analysis takes place

Implementation:

We have used a wine dataset from the google machine learning repository. We have divided the entire data into training and test dataset. Using the training dataset, we have first trained the algorithm and then we have tried to predict the test dataset using the trained algorithm. In the case of both supervised and unsupervised dimensionality reduction techniques we have applied K-NN for classification purposes and then we have compared the accuracy of the supervised model to the accuracy of the unsupervised model. We found out that the accuracy obtained in supervised learning was more as compared to the accuracy of unsupervised learning. This leads us to the conclusion that when a model is trained, it leads to better predictions.

The accuracy obtained in the PCA model was close to 95%, whereas the accuracy obtained on the same dataset using FLD was approximately 97%. In the case of metric learning, the accuracy obtained was nearly 100%

IRIS Dataset

Dimensionality reduction was done on IRIS dataset from UCL for both supervised and unsupervised learning and the following visualizations were obtained-

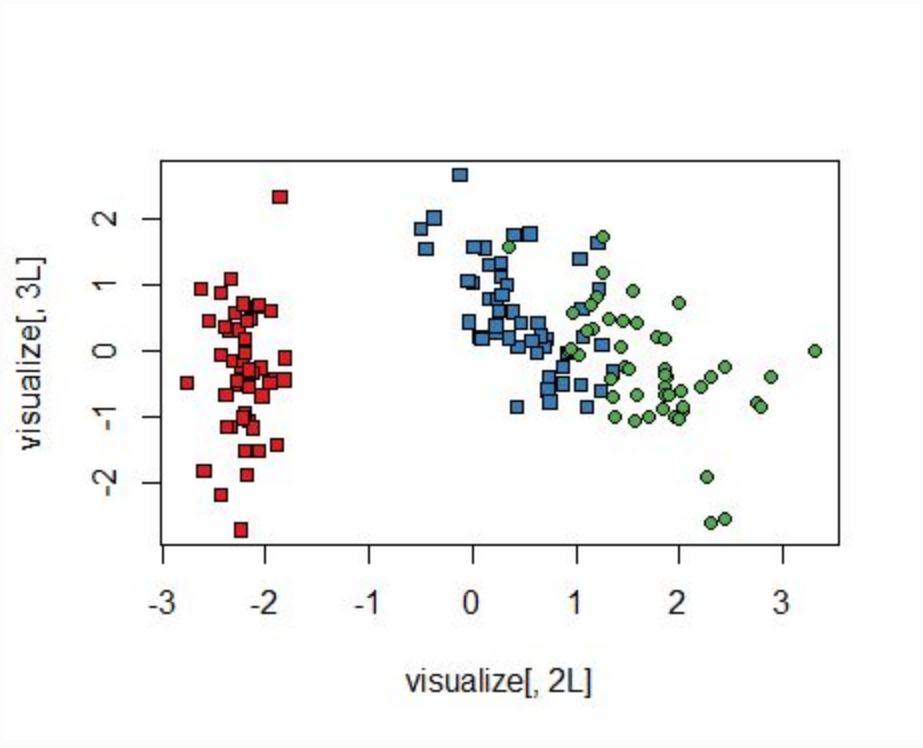


Fig.4 Plot obtained using PCA on IRIS dataset

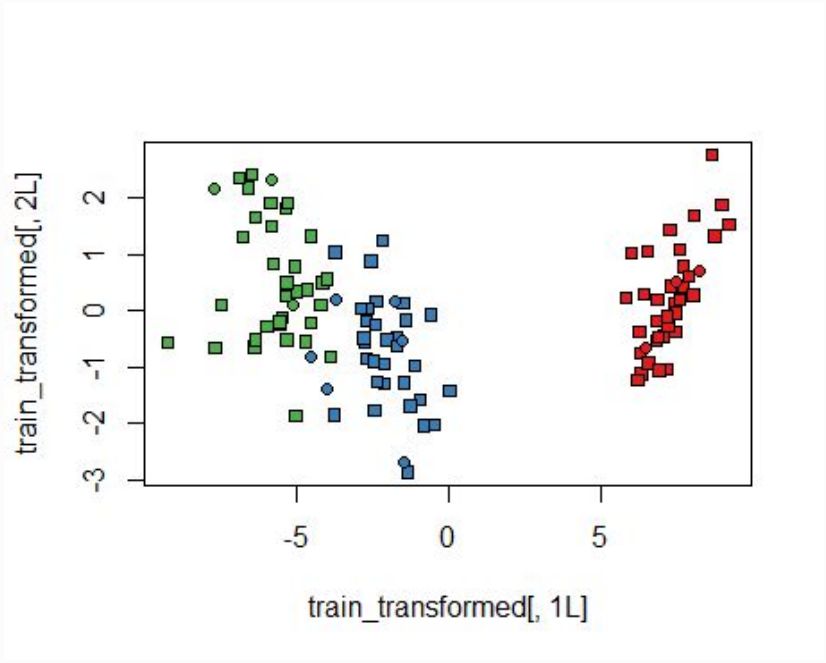


Fig.5 Plot obtained using LDA on IRIS dataset

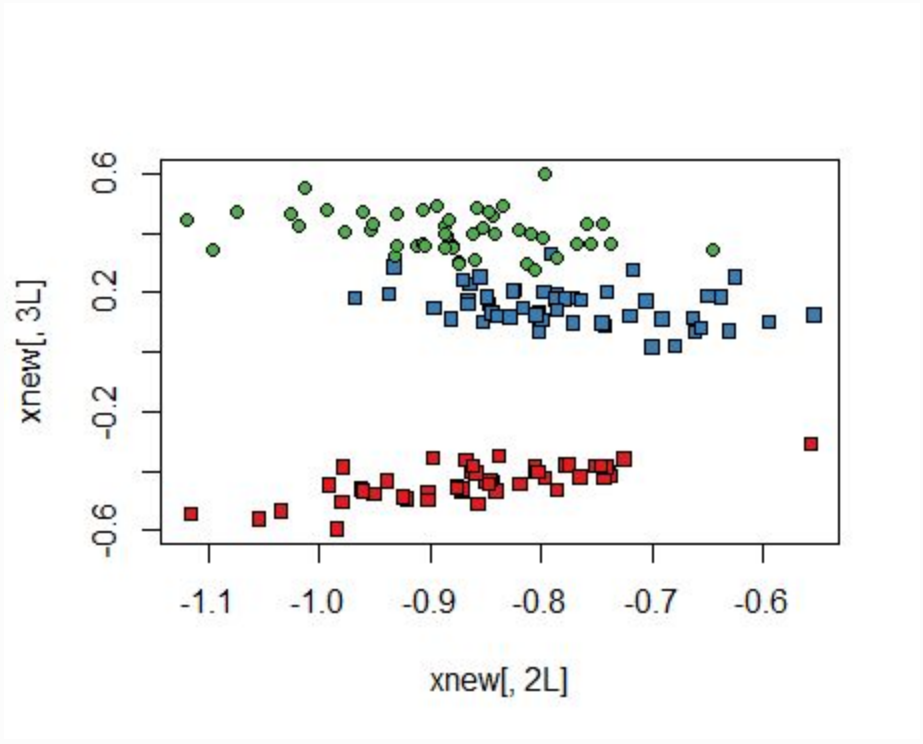


Fig.6 Plot obtained using RCA (metric learning) on IRIS dataset