American Journal of Engineering Research (AJER)
e-ISSN: 2320-0847 p-ISSN : 2320-0936
Volume-7, Issue-5, pp-447-451
www.ajer.org
Research Paper Open Access

A Novel Approach ToBig Data Analysis Through Statistical Learning

Medha Ch[1], Sreevani Y V[2]
[1](Dept. of Computer Science Engineering,Keshav Memorial Institute of Technology/JNTUH,India)
[2](Dept. of IT, Stanley College of Engineering and Technology for Women/ Osmania University, India)
Corresponding Author: Medha Ch

ABSTRACT :*In today's fast growing ever changing economy, huge amounts of data are generated and processed time and again by both small and large scale businesses[1]. These huge amounts of data are called Big Data. It is imperative for these businesses to be able to analyze, process and manage this data efficiently. Statistical learning encompasses the models and algorithms used for the purpose of understanding and wrangling Big data.These humungous volumes of data are either semi structured or unstructured in nature and require extremely sophisticated methodologies and techniques to bring out their true essence. Some very critical technologies such as tensor learningand kernel-based learning have been devised for the purpose of Big data analysis [2].* Tensors *are high dimensional generalizations of matrices. Some of the most recent applications of tensor learning include* the *estimation of parameters of dormant variable models like Hidden Markov Models.Kernel methods encompass algorithms for pattern classification, such as the* support vector *machine (SVM) and principle component analysis(PCA).*

KEYWORDS *–Big Data, Data Wrangling, Kernel Learning, Statistical Learning, Tensor.*

Date of Submission: 12-05-2018 Date of acceptance: 28-05-2018

I. INTRODUCTION

BigData is often characterized by its large volume, velocity, veracity, variety etc. One major issue associated with the volume of data is its dimensionality[1].Problems pertaining to Dimensionality are encountered when working with data spread across a high dimensional space. Dimensionality describes the number of features or attributes present in the dataset.Through the use of PCA as one of the statistical learning approaches such problems could be addressed[6].

Statistical Learning methodologies can be classified into three major heads, namely Supervised learning, Unsupervised learning and Reinforcement Learning. Supervised learning involves the prediction of a response variable based on one or more predictor variables[2][3]. Unsupervised learning is characterized by the process of drawing relationships and then classifying and grouping data into classes. Reinforcement learning is characterized by a learning algorithm that learns from experience and generally uses the trial and error method for the same. Kernel Learning and Tensor Learning are a few examples of some Statistical Learning methods.

Kernel methods owe their name to the use of kernel functions which enable them to operate in a high dimensional, implicit feature space without ever computing the coordinates of the data in that space, but rather by simply computing the inner products between the images of all pairs of data in the feature space. This operation is often computationally cheaper than the explicit computation of the coordinates. This approach is called the kernel trick. Kernel functions have been introduced for sequence data, graphs, text, images, as well as vectors.

Tensors serve as high dimensional representations of matrices. A wide range of real-world data takes the format of matrices and tensors, e.g., recommendation (Karatzoglou et al., 2010), video sequences (Kim et al., 2007), climates (Bahadori et al., 2014), genomes (Sankaranarayanan et al., 2015), and neuro-imaging (Zhou et al., 2013). A naive way to learn from such matrix and tensor data is to vectorize them and apply ordinary regression or classification methods designed for vector data. However, using such a vectorrepresentation would lead to loss in structural information of matrices and tensors such as low-rankness.

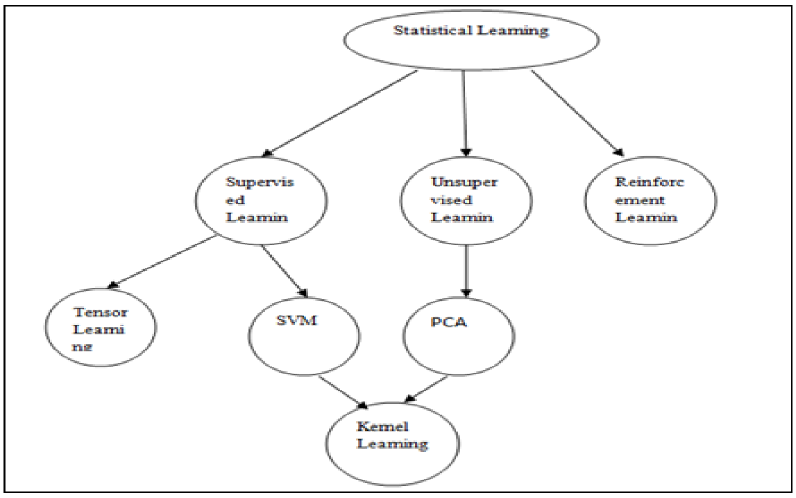

Fig 1: Components of Statistical Learning

II. STATISTICAL LEARNING

Statistical Learning Methodologies are methods used for data analysis post the explanatory data analysis phase that help automate analytical model building. Using sophisticated learning algorithms that iteratively learn from data, statistical learning helps researchers find hidden insights in the datasets under study. Based on the application domain, one could resort to the use of one of the three statistical learning methodologies, namely supervised learning, unsupervised learning and reinforcement learning[4]. A supervised learning algorithm is characterized with the supply of a set of predictor variables(inputs) and response variables(outputs), the algorithm is then supplied with some test data and is expected to generate outputs that are based off the training data. Supervised learning is usually used in scenarios where historical data could be used to predict an event in future. An Unsupervised learning algorithm is only supplied with a set of predictor variables and is expected to draw relations and patterns among the variables of the training dataset. Reinforcement Learning on the other hand encompasses algorithms that learn through the method of trial and error. These algorithms are said to learn from experience. This paper presents two of the most critical statistical learning methods,namely Tensor Learning and Kernel Learning. Tensor Learning is an example of a supervised learning algorithm. Support Vector Machines(SVM's) are an example of a supervised learning method. Principle Component Analysis (PCA) is an example of an unsupervised learning algorithm (Fig 1).

2.1 Tensor Learning

A tensor vector is a one-dimensional or first order tensor and a matrix is a two-dimensional or second order tensor.Tensor notations are much like matrix notations with a capital letter representing a tensor and lowercase letters with subscript integers representing scalar values within the tensor.

t111, t121, t131   t112, t122, t132   t113, t123, t133
T = (t211, t221, t231), (t212, t222, t232), (t213, t223, t233)
 t311, t321, t331   t312, t322, t332   t313, t323, t333

Many of the operations that can be performed with scalars, vectors, and matrices can be reformulated to be performed with tensors. As a tool, tensors and tensor algebra is widely used in the fields of physics and engineering. It is a term and set of techniques known in machine learning in the training and operation of deep learning models can be described in terms of tensors.

2.1.1 Tensors in Python

Like vectors and matrices, tensors can be represented in Python using the N-dimensional array (n-array).

A tensor can be defined in-line to the constructor of array() as a list of lists.

The example below defines a 3x3x3 tensor as a numpy indexed array. Three dimensions is easier to wrap your head around. Here, we first define rows, then a list of rows stacked as columns, then a list of columns stacked as levels in a cube. the tensor is printed as a series of matrices, one for each layer. For this 3D tensor, axis 0 specifies the level, axis 1 specifies the column, and axis 2 specifies the row.

2.1.2 Element-Wise Tensor Operations

As with matrices, we can perform element-wise arithmetic between tensors.The element-wise addition of two tensors with the same dimensions results in a new tensor with the same dimensions where each scalar value is the element-wise addition of the scalars in the parent tensors.

Tensor Addition
a111 ,a121,a131 , a112,a122,a132
A=(a211,a221,a231),(a112,a122,a132)
b111,b121,b131 b112,b122,b132
B=(b211,b221,b231),(b112,b122,b132)
C=A+B
a111 + b111, a121 + b121, a131 + b131 a112 + b112, a122 + b122, a132 + b132
C = (a211 + b211, a221 + b221, a231 + b231), (a112 + b112, a122 + b122, a132 + b132)

2.2 Kernel Learning:

Kernel based learning methods are characterized by a kernel function and an algorithm that takes these kernels as input. A few examples of some kernel functions are RBF, Polynomial, String, Anova etc. Some examples of a few learning algorithms include Support Vector Machines(SVM), Fisher Discrimination Analysis(FDA), Kernel PCA , Kernel CCA , Kernel Regression etc.

2.2.1 Support Vector Machines(SVM):

SVM's are an example of a supervised learning algorithm. These are associated with a set of learning algorithms that analyze a variety of data and recognize patterns that are used for classification and regression analysis[7]. An SVM is an example of a non-probabilistic,binary linear classifier[14]. With regard to its application for classification analysis, the training data fed into the model is classified under one of the two classes or groups that the response variable is expected to belong to. An SVM could be presumed to be a representation of a set of points in space, mapped in such a way that the points are separated by a clear gap that is quite wide[5]. A test data point is then predicted to belong to a class based on which side of the gap it falls in.When a new point is supposed to be classified under one of these groups, a line called the hyper plane will have to be constructed. The gap amidst the two groups or classes can however be inhibited by a number of hyper planes, it is therefore essential for one to choose a hyper plane that best divides the two classes. This could be illustrated through the following graph (Fig 2). A hyper plane must be chosen in such a way that it maximizes the margin between the two classes. (Fig 3)The vector points that the margin lines touch arecalled support vector points. Hence the name Support Vectormachine.

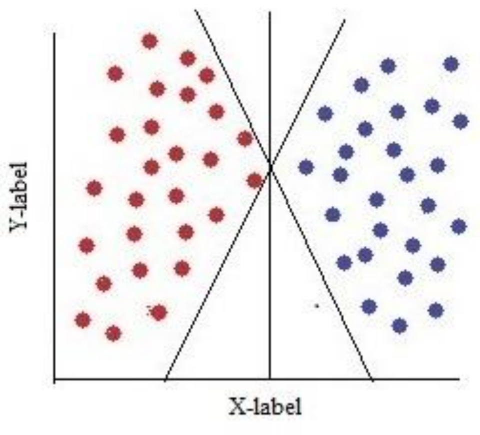

Fig 2

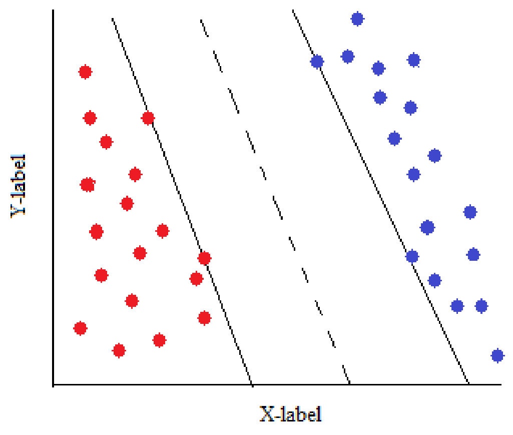

Fig 3

2.2.1.1SVM using python

The scikitlearn package contains all the methods required for analysis using SVM. The data will have to first be imported or read into the development environment. Other important libraries include numpy and pandas for numerical calculations, matplotlib and seaborn for data visualization[10][11]. These libraries can be imported using the following statements[12][13].

```
import numpy as np
import pandas as pd
```

import matplotlib.pyplot as plt
The SVM model can be imported using the following statements[8][9]
from sklearn.svm import svc
The model will later have to be instantiated using the following statements
model=svc()
The model can be trained using the following statements. The fit() is used for the same and it can take two arguments , the training data corresponding to the predictor variable and the training data corresponding to the response variable[8].
model.fit(x_train,y_train)
Data can be predicted using the following statements. The predict() is used for the same and it can take one argument , which is predictor variable corresponding to the test data.
predictions= model.predict(x_test)

2.2.2 Principal Component AnalysisPCA

The PCA method is an unsupervised statistical learning algorithm. It is basically used to probe into the interrelations among a set of variables with the aim of identifying an underlying pattern. This method is also called the General Factor Analysis method. Factor Analysis is used to come up with several orthogonal lines that best fit the data. These lines are perpendicular to each other in an n-dimensional space.The number of dimensions are always equal to the number of variables. For instance, a dataset with 7 variables would be characterized by a 7-dimensional sample space. Components are linear transformations that select a variable system for the dataset under study in such a way that the greatest variance of the dataset lies on the 1st axis, the second greatest variance on the second axis and so on[14]. This process helps bring the number of variables used in the analysis down. Components basically correspond to some common features based off the original features. This method is thus used for the identification of components that help is pattern analysis.

2.2.2.1 PCA using python

The scikitlearn package contains all the methods required for analysis using PCA. The data will have to first be imported or read into the development environment. Other important libraries include numpy and pandas for numerical calculations, matplotlib and seaborn for data visualization[10][11]. These libraries can be imported using the following statements[12][13].
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
Prior to applying PCA, the data will have to first be scaled. The “Standardscaler” model will have to be imported for this purpose. An object of the same will then have to be created. The following statements are used for the same[8][9].
From sklearn.preprocessing import Standardscaler.
scl= Standardscaler()
The principle components can be found using the fit(). Rotation and dimensionality reduction are to be applied using the transform(). Both the fit() and the transform() take the dataset as an argument[8][9].
Scl.fit(dframe)
Scaled_dt=scaler.transform(dframe)
The PCA model can be imported using the following statement
From sklearn.decomposition import PCA
The PCA() takes the number of components as input. In the statement below, n_componentsrefersto the number of components. The range of n_components lies between 1 and the number of dimensions in the scaled data[8][9].
Pc=PCA(n_components=2)
The model can be trained using the following statements. The fit() which takes the scaled data as input.
Pc.fit(scaled_dt)
Transforming this data to its 1st component
x_pca=pc.transform(scaled_dt)
The x_pca is reduced version of the original dataset or rather the scaled data (i.e data adjusted in relation to the two principal components post transformation in this case)The pca_components_ is an array of values that describes the relationship between every feature and each of the components. A heat map for the same can be constructed to enhance readability.

III. CONCLUSION

Statistical Learning techniques are used in the analysis of BigData.BigData is known for its humongous size and heterogeneous nature and the analysis of such data through traditional approaches like relational database management systems is highly difficult and is prone to a number of failures.Statistical Learning methodologies could be used to address critical problems that have been plaguing mankind for quite a long time. A few applications of statistical learning include fraud detection, Network Intrusion detection, Pattern and image recognition etc. Support Vector machines could be used to solve classification problems. Principle Component Analysis is used in dimensionality reduction. Tensor Learning methods could be used to generate low-ranked structural datathat could be utilized in various applications such as missing data imputation, robust principal component analysis and subspace.

REFERENCES

Journal Papers:

[1]. Big Data White Paper.

[2]. M Chen, S Mao, Y Liu, Big data: a survey. Mobile NetwAppl **19**(2), 2014,171–209.

Books:

[3]. Classification, (big) data analysis and statistical learning editorsr.e. moore, *interval analysis* (englewood cliffs, nj: prentice-hall, 1966).

[4]. http://www-bcf.usc.edu/~gareth/ISL/ISLR%20First%20Printing.pdf

[5]. N. Cristianini and J. Shawe-Taylor. An introduction to Support Vector Machines. Cambridge University Press, Cambridge, UK2000.

Proceedings Papers:

[6]. D Che, M Safran, Z Peng, *From big data to big data mining: challenges, issues, and opportunities*, in *Proceedings of the 18thInternational Conference on DASFAA* (Wuhan, 2013), pp. 1–15

Others:

[7]. http://www.cs.upc.edu/~belanche/Docencia/dm2/MATERIAL/7.SVMs-I/mainSVMs.pdf

[8]. Scikit-learn: Machine Learning in Python, Pedregosa *et al.*, JMLR 12, pp. 2825-2830, 2011.

[9]. API design for machine learning software: experiences from the scikit-learn project, Buitinck et al., 2013.

[10]. A guide to Numpy/scikit documentation https://docs.scipy.org/doc/

[11]. A guide to pandas documentation https://pandas.pydata.org/pandas-docs/stable/

[12]. A guide to matplotlib documentation https://matplotlib.org/contents.html

[13]. A guide to seaborn documentation https://seaborn.pydata.org/

[14]. https://www.udemy.com/python-for-data-science-and-machine-learning-bootcamp/