Big Data Machine Learning Algorithms

Gilbert Alipui, Claude Asamoah, Richard Barilla, Leigh Anne Clevenger, Alecia Copeland, Sam Elnagdy, Hugh Eng, Michael Holmes, Saravanan Jayaraman, Kevin Khan, Steven Lindo, Javid Maghsoudi, Mantie Reid, Michael Salé, and Charles C. Tappert

Seidenberg School of CSIS, Pace University, White Plains NY

***Abstract*—Machine Learning is a critical technology in predicting outcomes based on data. Four popular Big Data machine learning algorithms are addressed in this paper: Bayesian Decision Theory Classification, k-Nearest-Neighbor Classification, k-Means Clustering, and Linear Regression. The first two are supervised learning algorithms, the third an unsupervised learning algorithm, and the fourth a correlation algorithm. Computationally simple examples are used to illustrate the four algorithms.**

***Index Terms*— Bayes Decision Theory, K-Nearest-Neighbor, K-Means, Linear Regression, Machine Learning**

I. Introduction

Machine Learning is a branch of computer science that is used in Big Data predictive analytics to infer patterns [6] or to predict future outcomes [7]. It is employed in a range of computing tasks in which the discovery of patterns implicit in the data leads to better decision-making: examples include data mining [21], digital recognition, speech understanding, biometrics, tele medical diagnoses, and anomaly (fraud) detection [20]. Machine learning algorithms can be classified according to the underlying problem classes that they address: classification, regression, clustering, and rule extraction [4]. Alternatively, an algorithm can be classified according to its learning style: supervised, unsupervised, semi-supervised, or reinforcement learning.

The Big Data machine learning algorithms addressed in this paper are: Bayesian Decision Theory Classification, k-Nearest-Neighbor Classification, and k-Means Clustering. In addition, we discuss Simple Linear Regression vs. Classification and its application to problems whose models depend linearly on the unknown parameters, and where the response variable is continuous and numeric, as opposed to classification models where the response variable is binary or is split into two or more non-numeric classes.

Bayesian Decision Theory Classification provides optimal decisions for known probability distributions. Here, probability distributions are assumed to be multivariate Gaussian for all classes, and for simplicity identical class covariance matrices proportional to the identity matrix are assumed. With these simplifying assumptions the Gaussian probability distributions are spherical and the optimal Bayes decision surfaces are hyperplanes separating the classes in feature space. The algorithms derived from Bayesian Decision Theory are called parametric algorithms because they involve parameters, such as the mean vectors and covariance matrices of the classes [17].

The k-Nearest-Neighbor (kNN) algorithm makes no assumptions about the underlying probability distributions, and because there are no underlying parameters it is a non-parametric classification procedure. These two algorithms are therefore at the two extremes of supervised machine learning procedures [15].

The k-Means Clustering algorithm clusters observations into related groups without any prior knowledge of those relationships. The algorithm clusters observations into k groups, where k is provided as an input parameter. It then assigns each observation to clusters based upon the observation's proximity to the mean of the cluster. The cluster means are then recomputed and the process continues until no further changes occur [17].

Simple Linear Regression investigates the relationships between two variables by creating a line through the datapoints which provides the best fit to the data.

II. Bayesian Decision Theory Classification

The classification of objects is an important and integral part of research and application in many different areas and arenas. Some of these areas include artificial intelligence, pattern recognition, healthcare and statistics. When there is extensive and full knowledge of underlying probabilities, Bayes decision theory is well suited for giving optimal error rates. In situations where underlying probability information is not present, leveraging distance and similarities among samples as a means of classification is common for many algorithms. The k-Nearest Neighbors, k-Means, and linear regression algorithms are nonparametric procedures commonly used in these pattern recognitions situations with arbitrary distributions and unknown underlying probability densities. This study starts with a presentation of Bayesian Decision Theory Classification, then discusses the nonparametric pattern recognition procedures.

A. Advantages and Disadvantages of Bayesian Decision Theory Classification for Machine Learning

Bayesian Decision Theory Classification is a fundamental statistical approach to the problem of pattern classification and refers to a decision theory which is informed by Bayesian probability. It is considered the ideal case in which the probability structure underlying the categories is known perfectly. While this situation rarely occurs in practice, studying it allows determination of the optimal (Bayes) classifier against which all other classifiers can be compared. In some problems it enables prediction of the error resulting from generalization to novel patterns. Bayesian Decision Theory is a formal theory for rational inference and decision making.

Application of Bayesian Decision Theory provides methods for:

- thinking about problems of inference under uncertainty
- constructing mathematical models for inference and decision problems
- modelling applications for drawing inferences from data and making decisions

Bayesian Decision Theory Classification is based on quantifying the tradeoffs between various classification decisions using probability and the costs that accompany such decisions. It makes the assumption that the decision problem is posed in probabilistic terms, and that all the relevant probability values are known.

In this theory, an agent operating under such a decision theory uses the concepts of Bayesian statistics to estimate the expected value of its actions, and update its expectations based on new information. These agents can and are usually referred to as estimators.

From the perspective of Bayesian Decision Theory, any kind of probability distribution – such as the distribution for tomorrow's weather – represents a prior distribution. That is, it represents how we expect today what the weather is going to be tomorrow. This contrasts with frequentist inference, the classical probability interpretation, where conclusions about an experiment are drawn from a set of repetitions of such experience, each producing statistically independent results. For a frequentist, a probability function would be a simple distribution function with no special meaning [4].

Suppose we intend to meet a friend tomorrow, and expect a 50% chance of rain. If we are choosing between various options for the meeting, with the pleasantness of some of the options (such as going to the park) being affected by the possibility of rain, we can assign values to the different options with or without rain. We can then pick the option whose expected value is the highest, given the probability of rain.

B. Simplifying Bayes Decision Theory

Bayes Decision Theory is a general parametric algorithm, and in this study assumptions are made which give accuracy comparable to the non-parametric algorithms but with less computational complexity. The general theory is limited in this study with the following assumptions:

- Supervised learning data – a labeled training set of data is available for constructing the classification system
- Identical spherical Gaussian probability distributions for all classes – that is, identical class covariance matrices proportional to the identity matrix
- Equal prior probability distributions

The following are significant equations used to evaluate Bayesian Decision Theory Classification.

Bayes Rule

$$P(w_i|x) = \frac{P\left((x|w_i)\,P(w_i)\right)}{P(x)} \quad (1)$$

- $P(w_i \mid x)$ - probability of instance x being in class w_i, This is the value to be computed.
- $P(x \mid w_j)$ - probability of generating instance x given class w_i, Being in class w_i causes feature x to appear with some probability.
- $P(w_i)$ - probability of occurrence of class w_i, How frequently the class w_i occurs.
- P(x) - probability of instance x occurring. The P(x) term can be eliminated because it is the same for all classes.

Bayes rule can also be viewed as

$$posterior = \frac{likelihood * prior}{evidence}$$

[6, pp.615-616]

Multi-class Logarithmic Discriminant Function

$$g_i(x) = \ln P(x|w_i) + \ln P(w_i) \quad (2)$$

[6, pp.52-53]

Multivariate Normal Density Function

$$P_i(x) = \frac{1}{(2\pi)^{d/2}|\sum_i|^{1/2}} e^{-\frac{1}{2}(x-u_i)^t \sum_i^{-1}(x-u_i)} \quad (3)$$

Where μ is the expected value, mean, or average of x, and $\sum$ is the covariance matrix [6, p.625]

Logarithmic Discriminant Function

$$g_i(x) = \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\sum_i| - \frac{1}{2}(x-u_i)^t \sum_i^{-1}(x-u_i) + \ln P(w_i) \quad (4)$$

Squared Mahalanobis Distance

Assume equal $P(w_i)$ and å :" :

$$\max \; g_i(x) = -(x - m_i)^t å^{-1} (x - m_i) \quad (5)$$

$$\text{or min} \; d_i(x) = (x - m_i)^t å^{-1} (x - m_i) \quad (6)$$

[6, pp.35-36]

Squared Euclidean Distance

From the squared Mahalanobis distance, further assume diagonal Σ

$$d_i(x) = \sum_{k=1}^{d} \left(\frac{x^k - \mu_i^k}{\sigma^k}\right)^2 \quad (7)$$

Further assume $\Sigma \propto I$

$$d_i(x) = \sum_{k=1}^{d} (x^k - \mu_i^k)^2 \quad (8)$$

[6, p.36]

"Pattern Classification" by Duda provides a complete discussion of Bayes Decision Theory Classification [6].

C. Example of Bayesian Decision Theory Classification Boundary Determination

Using the assumptions and equations described in the previous section, this example demonstrates the application of Bayesian Decision Theory Classification. Training samples for a three-class (red, blue, green), 2D (two-feature) problem are shown in the Figure 1. Red = {(1,4), (-3,0), (5,0), (1,-4)}, Green = {(-6,17), (-10,13), (-2,13), (-6,9), Blue = {(10,15), (6,11), (14,11), (10,7)}. Drawn on Figure 1 are the Bayes decision boundaries which provide the boundaries for classifying all new data presented to the system. These are the perpendicular bisectors between the means [15].

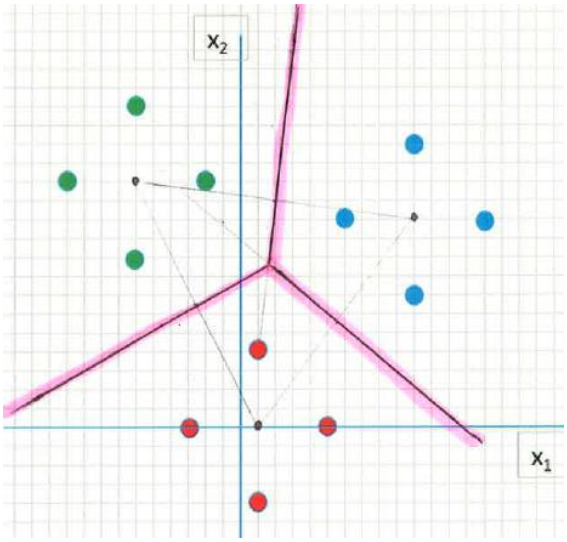

Figure 1. Bayes decision boundaries in a three-class, two-feature problem with simplifying assumptions.

III. K-NEAREST-NEIGHBOR CLASSIFICATION

The proliferation of computers, mobile devices and other computing platforms has resulted in the constant generation of a flood of data. Our lives are awash in ever-increasing amounts of data and there seems to be no end in sight [21]. Scientists and researchers seeking to derive meaning from this data have many tools at their disposal. This section describes the k-Nearest Neighbor (k-NN) algorithm.

The k-Nearest-Neighbor algorithm is a non-parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression:

- In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor.
- In k-NN regression, the output is the property value for the object. This value is the average of the values of its k nearest neighbors.

k-NN is a type of instance-based learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms. Both for classification and regression, it can be useful to weight the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. For example, a common weighting scheme consists in giving each neighbor a weight of 1/d, where d is the distance to the neighbor. The neighbors are taken from a set of objects for which the class (for k-NN classification) or the object property value (for k-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required [2].

The classification of objects is an important and integral part of research and application in many different areas, including artificial intelligence, pattern recognition, healthcare and statistics. When there is extensive and full knowledge of underlying probabilities, Bayesian Decision Theory Classification is well suited for giving optimal error rates. In situations where underlying probability information is not present, leveraging distance and similarities among samples as a means of classification is common for many algorithms. The k-NN algorithm is commonly used in these pattern recognitions situations.

Developed from the need to perform discriminant analysis when reliable parametric estimates of probability densities are unknown or difficult to determine, the U.S. Air Force School of Aviation Medicine introduced k-NN in an unpublished report in 1951 by Fix and Hodges. It was further refined in 1967 by Cover and Hart when more properties of k-NN were formally defined. k-NN is known as lazy learning where the function is only approximated locally and all computation is deferred until classification[18].

The k-NN algorithm usually uses the Euclidean or the Manhattan distance. However, any other distance such as the Chebyshev norm or the Mahalanobis distance can also be used [3]. The common distance functions are [12]:

Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$ (9)

Manhattan $\sum_{i=1}^{k}|x_i - y_i|$ (10)

Minkowski $\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{\frac{1}{q}}$ (11)

A problem that can occur when using k-NN is in cases where even the nearest neighbors are very far away. This is called "the curse of dimensionality," and it makes k-NN a poor algorithm in some cases [4].

A. Comparing Bayesian Decision Theory Classification to k-Nearest-Neighbor

Bayesian Decision Theory Classification offers two pseudocounts or hyperparameters available to the data scientist for data tuning. By contrast, k-NN only has one knob, namely k, the number of neighbors. Bayes is a linear classifier, while k-NN is not. Both are examples of supervised learning, meaning the data is labeled. k-NN requires no training - the dataset can just be loaded. Also, the curse of dimensionality and large feature sets are a problem for k-NN, while under these conditions Bayesian Decision Theory Classification performs well. [12]

The example given previously for Bayesian Decision Theory Classification can also be analyzed using k-NN. As before, training samples for a three-class (red, blue, green), 2D (two-feature) problem are shown in Figure 2. Red = {(1,4), (-3,0), (5,0), (1,-4)}, Green = {(-6,17), (-10,13), (-2,13), (-6,9), Blue = {(10,15), (6,11), (14,11), (10,7)}. No probability distribution is assumed. Drawn on Figure 2 are the k-NN decision boundaries which provide the boundaries for classifying all new data presented to the system. These are similar to those for Bayes Decision Theory Classification [15].

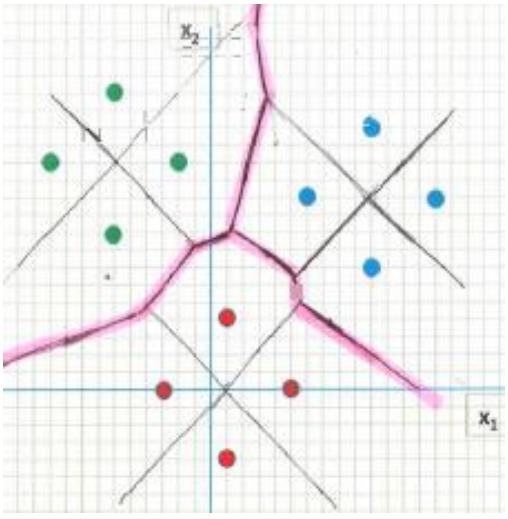

Figure 2. k-Nearest-Neighbor boundaries in a three-class, two-feature problem with no distribution assumed.

IV. K-MEANS CLUSTERING

The k-Means algorithm is arguably one of the most referenced and used clustering algorithms. It is important to understand that k-Means is not a classification algorithm. k-Means is generally used on data sets that contain some possible pattern or grouping, but those groups are not defined. Take for instance a large data set of sales transactions for a major department store chain. In a data warehouse environment, this transaction data may be merged with customer biographic and demographic data. While obvious classes can be formed such as "male shoppers" and "female shoppers" or "west-coast customers" and "east-coast customers," there may be other more effective ways of clustering the data so that effective predictive and prescriptive analytics can be performed. Not only can k-Means clustering identify possible groupings, but it then allocates data points (in this case, transactions) to the appropriate cluster.

k-Means has its roots in traditional statistical analysis. As the name of the algorithm implies, its job is to assign individual data points - a customer, an occurrence of an event, on object, etc. - to a cluster whose center, called a centroid, is nearest. It does this using an iterative algorithm as described here and shown in Figures 3 and 4:

Step 1: Select a value of "k." k is the number of clusters into which to split the data points. Note that the value of k can be adjusted many times and the algorithm repeated in order to find an optimal solution.

For example, you may think your customer transactions are separated into 5 different groupings. Generating k can be somewhat arbitrary but most software packages can also prescribe potential optimal values of k.

Step 2: Randomly assign a center/centroid to each cluster. Generally, extreme opposite points are selected as initial cluster centroids.

Step 3: Assign each point in the data set to the nearest centroid. This is generally done using Euclidean distance. Whichever cluster centroid is closest to each individual point (by way of Euclidean distance) is the cluster to which the point is assigned.

Step 4: Recompute the new cluster centroids. This is done by taking the average of all the points in the cluster; that is, its coordinates are the means for each dimension separately over all the points in the cluster.

Step 5: Repeat steps 3 and 4 until the clusters no longer change; that is to say, the assignment of points to clusters becomes stable.

The example given previously for Bayesian Decision Theory Classification and k-Nearest-Neighbor Classification can also be analyzed using k-Means. As before, training samples for a three-class (red, blue, green), 2D (two-feature) problem are shown in Figure 2. Red = {(1,4), (-3,0), (5,0), (1,-4)}, Green = {(-6,17), (-10,13), (-2,13), (-6,9), Blue = {(10,15), (6,11), (14,11), (10,7)}. No probability distribution is assumed. Shown in Figure 3 are the two steps required for reaching k-Means decision boundaries using three *distant* points, which provide the boundaries for classifying all new data presented to the system.

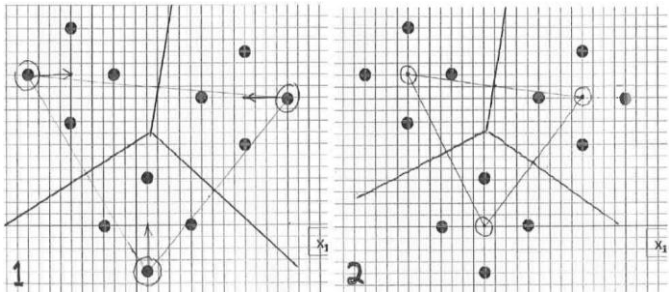

Figure 3. k-Means clustering initialized with three distant points converges in two steps.

Figure 4 shows that five steps are needed to reach the k-Means decision boundaries with three *non-distant* points. These final results are similar to those for Bayes Decision Theory Classification and k-Nearest-Neighbor [15].

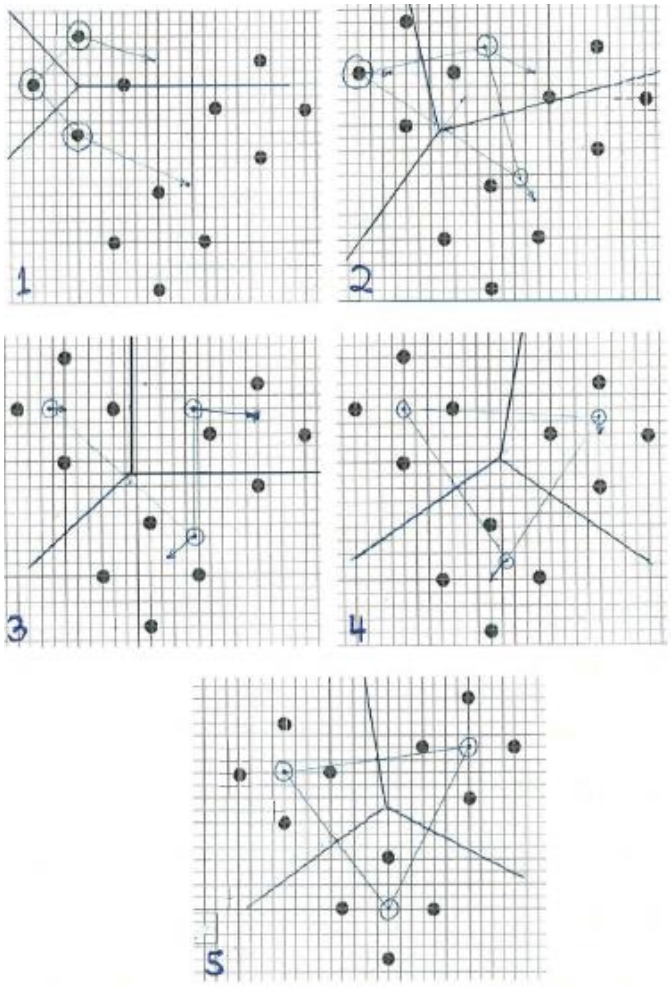

Figure 4. k-Means clustering initialized with three non-distant points converges in five steps.

The k-Means algorithm has its advantages and disadvantages in its data mining and analytic applications. This technique is especially good with a large number of variables. k-Means can be computationally faster than other clustering methods such as hierarchical clustering, especially if k is small. k-Means tends to also produce tighter clusters than other methods. One of the major drawbacks to this algorithm is that it is difficult to judge the quality of clusters produced, as the value of k is a preselected fixed number. Because the true number of optimal partitions is unknown, picking a good value of k can be troublesome. It is helpful to rerun the clustering algorithm using the same as well as different k values to compare the results achieved. It is important to note that k-Means does not work well for non-globular clusters. These are clusters that do not have well-defined centers, but have a more chain-like shape. For these types of data structures, algorithms such as Jarvis-Patrick may be better suited.

V. SIMPLE LINEAR REGRESSION

Regression analysis is a statistical tool for the investigation of relationships between variables. Usually, the investigator seeks to ascertain the causal effect of one variable upon another—the effect of a price increase upon demand, for example, or the effect of changes in the money supply upon the inflation rate. To explore such issues, the investigator assembles data on the underlying variables of interest and employs regression to estimate the quantitative effect of the causal variables upon the variable that they influence. The investigator also typically assesses the statistical significance of the estimated relationships, that is, the degree of confidence that the true relationship is close to the estimated relationship.

A. Use of Continuous or Dichotomous Independent Variables

Another way of exploring regression analysis is to know that it is used when predicting a continuous dependent variable from a number of independent variables. When a researcher wishes to include a categorical variable with more than two levels in a multiple regression prediction model, additional steps are needed to insure that the results are interpretable. The independent variables used in regression can be either continuous or dichotomous. Independent variables with more than two levels can also be used in regression analyses, but they first must be converted into variables that have only two levels.

These steps include recoding the categorical variable into a number of separate, dichotomous variables. If the dependent variable is dichotomous, then logistic regression should be used. If the split between the two levels of the dependent variable is close to 50-50, both logistic and linear regression give similar results.

B. Applications of Regression Analysis

There are many applications of regression analysis. Regression techniques have long been central to the field of economic statistics as econometrics. Increasingly, they have become important to lawyers and legal policy makers. Regression has been offered as evidence of liability under Title VII of the Civil Rights Act of 1964 of racial bias, in death penalty litigation, as evidence of damages in contract actions, as evidence of violations under the Voting Rights Act, and as evidence of damages in antitrust litigation.

C. Practical Use of Regression Analysis

Linear regression was the first type of regression analysis to be studied rigorously, and to be used extensively in practical applications. This is because models which depend linearly on unknown parameters are easier to fit than models which are non-linearly related to parameters, and because the statistical properties of the resulting estimators are easier to determine. Linear regression has many practical uses. Most applications fall into one of the following two broad categories:

- If the goal is prediction, or forecasting, or reduction, linear regression can be used to fit a predictive model to an observed data set of y and X values. After developing such a model, if an additional value of X is then given without its accompanying value of y, the fitted model can be used to make a prediction of the value of y.
- Given a variable y and a number of variables X¬1…X_p that may be related to y, linear regression analysis can be applied to quantify the strength of the relationship between y and the X_j, to assess which X_j may have no relationship with y at all, and to identify which subsets of the X_j contain redundant information about y.

D. Simple Linear Regression vs. Multiple Linear Regression

Linear regression is an approach for modeling the relationship between a scalar dependent variable y and one or more explanatory variables (or independent variables) denoted X. The case of one explanatory variable is called simple linear regression. When we have more than one explanatory variable, the process is called multiple linear regression.

E. Linear models: Using "Conditional Mean" & Median

In linear regression, data are modeled using linear predictor functions, and unknown model parameters are estimated from the data. Such models are called linear models. Most commonly, linear regression refers to a model in which the conditional mean of y given the value of X is an affine function of X. Less commonly, linear regression could refer to a model in which the median, or some other quantile of the conditional distribution of y given X is expressed as a linear function of X. Like all forms of regression analysis, linear regression focuses on the conditional probability distribution of y given X, rather than on the joint probability distribution of y and X, which is the domain of multivariate analysis.

F. Use of the Least Squares Approach in Linear Regression Models

Linear regression models are often fitted using the least squares approach, but they may also be fitted in other ways, such as by minimizing the "lack of fit" in some other norm (as with least absolute deviations regression), or by minimizing a penalized version of the least squares loss function as in ridge regression (L2-norm penalty) and lasso (L1-norm penalty). Conversely, the least squares approach can be used to fit models that are not linear models. Thus, although the terms "least squares" and "linear model" are closely linked, they are not equivalent formulas.

Given a data set

$$\{y_i, x_{i1}, \ldots, x_{ip}\}_{i=1}^{n}$$

of n statistical units, a linear regression model assumes that the relationship between the dependent variable y_i and the p-vector of repressors x_i is linear. This relationship is modeled through a disturbance term or error variable ε_i — an unobserved random variable that adds noise to the linear relationship between the dependent variable and regressors. Thus the model takes the form

$$y_i = \beta_i x_{i1} + \ldots + \beta_p x_{ip} + \varepsilon_i = x_i^T \boldsymbol{\beta} + \varepsilon_i, \quad i = 1, \ldots, n, \tag{12}$$

Where T denotes the transpose, so that $x_i^T \boldsymbol{\beta}$ is the inner product between vectors x_i and **β.**

Often these **n** equations are stacked together and written in vector form as

$$y = X\beta + \varepsilon, \tag{13}$$

Where

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \quad X = \begin{pmatrix} X_1^T \\ X_2^T \\ \vdots \\ X_n^T \end{pmatrix} = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ x_{21} & \cdots & x_{2p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix},$$

$$\beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix}, \quad \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}.$$

G. Example of Simple of Linear Regression

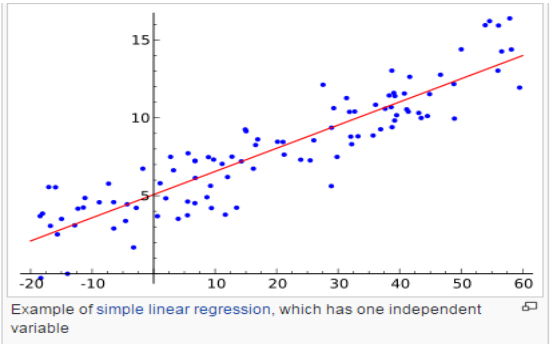

Figure 5. Example of linear regression with one independent variable.

Simple linear regression is a way to describe a relationship between two variables through an equation of a straight line, called line of best fit, that most closely models the relationship. The following linear regression formula provides the least-squares, best-fit equation for the line:

$$y = a + bx$$

where

$$b = \frac{\sum_{i=1}^{n} x_i y_i - n\overline{xy}}{\sum_{i=1}^{n} x_i^2 - n\overline{x^2}} \quad \text{and} \quad a = \bar{y} - b\bar{x} \tag{14}$$

Where x-bar and y-bar are the means of the x's and y's.

This formula can be applied to a simple example. The computed least-squares, best-fit line through the five points {(2,2), (0,0), (-2,-2), (-1,1), (1,-1)} is shown in Figure 6.

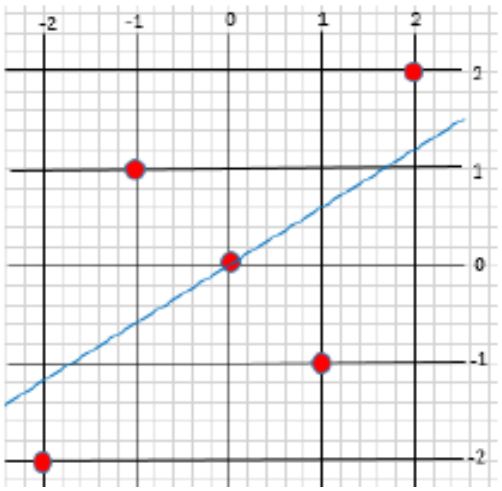

Figure 6. Least-squares, best-fit linear regression.

- The equation of the solution line is y=0.6x
- The least-squares error, which is the sum of the squares of the y-differences between each original point and the best-fit regression line, is

2 * (1.6 * 1.6 + 0.8 * 0.8) = 2 * (2.56 + 0.64) = 2 * 3.20 = 6.40

H. Example of a Cubic Polynomial Regression - a Type of Linear Regression

A polynomial term–a quadratic (squared) or cubic (cubed) term turns a linear regression model into a curve. But because it is X that is squared or cubed, not the Beta coefficient, it still qualifies as a linear model.

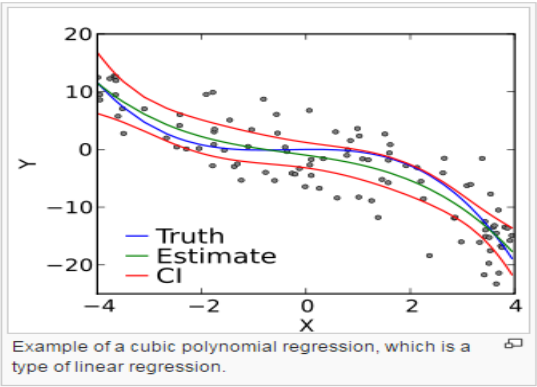

Figure 7. Example of cubic polynomial regression.

VI. Discussion and Conclusions

This study covered four popular Big Data Machine Learning Algorithms critical in predicting outcomes based on data. Simple examples of each type of algorithms were presented.

Supervised pattern classification procedures take in data related to a pattern (object), and make a decision based on the class (category) of the pattern, whereas with unsupervised learning the data is not labeled and there are no target variables. Clustering groups data points, with similar data points in one cluster and dissimilar points in a different group. A number of different measurements can be used to measure similarity.

The study began with Supervised Learning, which means a set of labeled patterns are available to design and train the classification system. Bayesian Decision Theory Classification is one such system. It provides optimal decisions for known probability distributions.

Next the study investigated k-Nearest Neighbor Classification, another Supervised Learning algorithm. Unlike Bayesian Decision Theory Classification, k-Nearest Neighbor makes no assumptions about the underlying probability distributions, and because there are no underlying parameters, it is called a non-parametric classification procedure. These first two algorithms are therefore at the two extremes of supervised machine learning procedures.

The third Big Data Machine Learning Algorithm studied is a clustering technique used in unsupervised learning, called k-Means. k-Means is a clustering algorithm that tries to partition a set of points into k sets (clusters) whereby the points in each cluster lean near each other. The points have no external classification, making k-Means an unsupervised learning method [5].

The final algorithm discussed is Linear Regression. Linear regression is typically used on observed data as a model to fit a linear equation to the relationship between two variables. One

variable is treated as the explanatory variable, and the other the dependent variable. This is done after ascertaining whether or not a relationship actually exists between the variables of interest [10].

Linear regression is considered the first type of regression analysis to be studied rigorously and to be used extensively in practical applications. This is because models that depend linearly on unknown parameters are easier to fit than models, which are non-linearly related to their parameters. Also, the statistical properties of the resulting estimators are easier to determine [19].

The decision boundary results in Figures 1-4 show that for the simple 3-class 2-D problem, the resulting boundaries are similar for Bayesian Decision Theory Classification, k-Nearest-Neighbor, and k-Means Clustering. In practice, the algorithms k-NN and k-Means are good choices for Big Data Machine Learning, because they are easier to use than Bayesian Decision Theory Classification, and in our examples give equivalent decision boundaries.

REFERENCES

[1] D.R. Abrams, "Introduction to Regression," http://dss.princeton.edu/online_help/analysis/regression_intro.htm, accessed March 2015

[2] N.S. Altman, "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression," *The American Statistician*, Vol.46, No.3, 1992, pp.175–185

[3] A. Ashari, I Paryudi, and A.M. Tjoa, "Performance Comparison between Naïve Bayes, Decision Tree and k-Nearest Neighbor in Searching Alternative Design in an Energy Simulation," *International Journal of Advanced Computer Science and Applications*, Vol 4, No. 11, 2013

[4] C.M. Bishop, *Pattern Recognition and Machine Learning*, Vol. 4, No. 4. New York: Springer, 2006, p.12

[5] CrossValidated, "What are the main differences between K-means and K-nearest neighbours?," http://stats.stackexchange.com/questions/56500/what-are-the-main-differences-between-k-means-and-k-nearest-neighbours, accessed April 2015

[6] R.O. Duda, P.E. Hart and D.G. Stork, *Pattern Classification.* New York, NY: Wiley, 2001, pp. 84-102

[7] T. Hastie, R.J. Tibshirani and J.H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer, 2011

[8] P. Langley, W. Iba, and K. Thompson, "An Analysis of Bayesian Classifiers," *Proceedings of the Tenth Conference on Artificial Intelligence*, San Jose, CA, 1992. AAAI Press.

[9] LessWrongWIKI, "Bayesian Decision Theory," 2012, http://wiki.lesswrong.com/wiki/Bayesian_decision_theory, accessed February 2015

[10] "Linear Regression," http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm, accessed March 2015

[11] N.G. Polson, "Bayesian Inference," *International Library of Critical Writings in Econometrics*, Edward Elgar Pub, 1995.

[12] S. Sayad, "An Introduction to Data Mining," 2015, http://www.saedsayad.com/k_nearest_neighbors.htm, accessed February 2015

[13] D. Spiegelhalter, K. Rice, "Bayesian Statistics," 2009, http://www.scholarpedia.org/article/Bayesian_statistics, accessed February 2015

[14] D.W. Stockburger, "Multiple Regression with Categorical Variables," http://www.psychstat.missouristate.edu/multibook/mlt08m.html, accessed March 2015

[15] C. Tappert, Emerging Technologies II, Pace University, February 2015, http://www.csis.pace.edu/~ctappert/dps/d861-15/assign/assign1.pdf, accessed March 15.

[16] The Analysis Factor, "Regression Models: How do you know you need a polynominal," 2015, http://www.theanalysisfactor.com/regression-modelshow-do-you-know-you-need-a-polynomial/, accessed March 2015

[17] Wikipedia, "K-Means Clustering," http://en.wikipedia.org/wiki/K-means_clustering, accessed March 2015

[18]Wikipedia: "K-Nearest Neighbors," http://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm, accessed February 2015

[19]Wikipedia, "Linear Regression," http://en.wikipedia.org/wiki/Linear_regression, accessed March 2015

[20] Wikipedia, "Machine Learning," http://en.wikipedia.org/wiki/Machine_learning, accessed March 2015

[21] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining Practical Machine Learning Tools and Techniques*, Burlington, MA: Morgan Kaufmann, 2011