Chapter 2
Big Data Essentials

Abstract The main objective of this chapter is to organize the big data essentials that contribute to the analytics of big data systematically. It includes their presentations in a simple form that can help readers conceptualize and summarize the classification objectives easily. The topics are organized into three sections: big data analytics, big data classification, and big data scalability. In the big data analytics section, the big data controllers that play major roles in data representation and knowledge extraction will be presented and discussed in detail. These controllers, the problems and challenges that they bring to big data analytics, and the solutions to address these problems and challenges will also be discussed. In the big data classification section, the machine learning processes, the classification modeling that is characterized by the big data controllers, and the classification algorithms that can manage the effect of big data controllers will be discussed. In the big data scalability section, the importance of the low-dimensional structures that can be extracted from a high-dimensional system for addressing scalability issues will be discussed as well.

2.1 Big Data Analytics

In [1], Philip Russom defined the term "Big data analytics" by dividing it into two keywords "big data" and "analytics," and he described them individually based on their combined influence on business intelligence. Business intelligence is one of the applications that can benefit from the big data techniques and technologies. Big data analytics also has a scientific significance in real-world applications; therefore, it is appropriate to define it based on class characteristics, feature characteristics, and observation characteristics—the three important controllers of big data. This chapter discusses these big data controllers in detail. The understanding of big data controllers, the analysis of the problems that the controllers create in a big data environment, the confrontation of the challenges for solving these problems efficiently,

S. Suthaharan, *Machine Learning Models and Algorithms for Big Data Classification*, Integrated Series in Information Systems 36,
DOI 10.1007/978-1-4899-7641-3_2

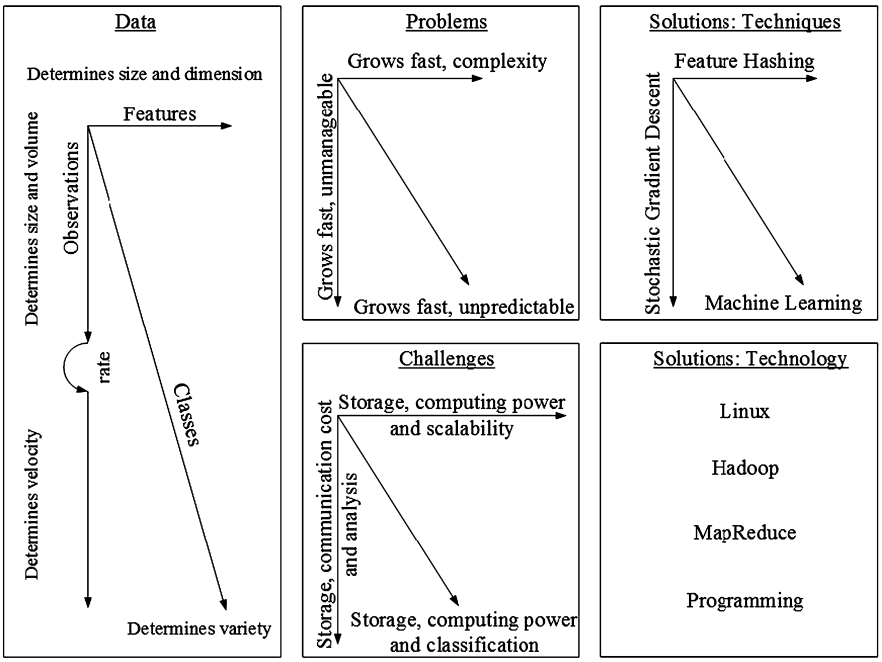

Fig. 2.1 The contributors to the analytics of big data: the controllers, problems, challenges, and solutions (techniques and technologies)

and the development of techniques and technologies to address big data classification are the important contributors in big data analytics. The definition of big data analytics based on the contributors would help the classification of the structured and unstructured data significantly. Whether a data set is structured or unstructured may be determined by proper understanding of the controllers.

2.1.1 Big Data Controllers

The main goal of this book is to address big data classification problems, challenges, and solutions. In [2], these topics are presented focusing on network intrusion detection, which is considered a big data application. A complete understanding of the class characteristics, feature characteristics, and observation characteristics can help address these issues. These three controllers are illustrated in Fig. 2.1. Let us first understand the information presented in the first column of this figure. It presents a 3D representation of a set of data. The observations (the vertical axis) represent the events that are recorded or observed by a system, and they describe the big data's term volume. The number of observations, n, states that the size of the data set is n.

They may also describe the big data's term velocity which may be defined by the availability of data on demand. Hence, the observation controls the classification issues that resulted from the volume and velocity of the big data. The features (the horizontal axis) represent the independent variables that generate the events (or responses), and hence they determine the volume and the dimensionality of the data. The number of features, p, means the data set has p-dimensions. They control the scalability of the data, and the parameters, n and p, together define the characteristics of dimensionality. For example, if $n < p$, then the data set is said to be high dimensional. The third big data controller, the classes (the diagonal axis), represents the types of the events and determines the variety term of big data. It helps to group the data and creates the need for dividing the data domain robustly.

2.1.2 Big Data Problems

The individualization of the controllers and their uncoordinated efforts can create problems in the big data realm. Each controller defines its own contribution to big data, and it affects the individualization of the other controllers orthogonally, and hence we define the controllers' problems using a three-dimensional space as shown in the second column of Fig. 2.1. As it is defined, the controller class contributes to the unpredictability of big data. It means that the detection (or classification) of classes with the growth in big data is very difficult and unpredictable. The growth in the class types is system dependent, and it is independent to the users' knowledge and the experience. Hence, the big data classification becomes unpredictable, and the application of machine-learning models and algorithms becomes difficult.

Similarly, the controller feature contributes to the complexity of big data. It makes the classification of patterns difficult by increasing the dimensionality of data. It is one of the major contributors to the scalability problems in a big data paradigm. The third controller observation contributes to the difficulties of managing, processing, and analyzing the data. Its growth increases the volume of data and makes the processing difficult with the current technologies. Therefore, if we understand the individuality of these controllers and their uncoordinated efforts clearly, then we should be able to confront the challenges that they bring to big data classification.

2.1.3 Big Data Challenges

The individualization and orthogonality problems reported in the previous section create several challenges to the current techniques and technologies. The bottom figure in the second column of Fig. 2.1 illustrates the challenges. The challenges associated with the techniques may be categorized as classification, scalability, and analysis. The challenges associated with the technologies may be categorized as computation, communication, and storage. In addition, the problems can bring

security challenges as reported in the papers [2, 3] as well. Let us now connect these challenges with the corresponding big data controllers.

The problems caused by the controller class can impact the performance degradation of the classification techniques, while imposing challenges on the choice of computing power and storage requirements. The problems caused by the controller feature challenge the reduction of dimensionality and the storage and computing power. Similarly the controller observation brings challenges in the data processing, storage requirements, and communication issues when the data are distributed as demonstrated in [2] to solve intrusion detection problems.

2.1.4 Big Data Solutions

The big data solutions are illustrated in the third column of Fig. 2.1. The big data solutions are divided into techniques and technologies as illustrated in these figures. The techniques involve solving problems by addressing the challenges associated with the big data controllers with respect to their speed, complexity, unpredictability, (un)manageability, and scalability. The techniques may be divided into modeling and algorithms whereas the technologies may be divided into systems and framework. The modeling and algorithms may be described more specifically with supervised machine learning (related to classes) [4], feature hashing (related to features) [5, 6], and stochastic gradient descent (related to observations) [7, 8]. Similarly the systems and framework may be described more specifically with the modern distributed file systems like the Hadoop distributed file system [9, 10] and modern programming frameworks like the MapReduce programming model [11, 12].

2.2 Big Data Classification

The main focus of this book is on big data classification [2, 13], which is one of the important and difficult problems in big data analytics. In simple terms, big data classification is a process of classifying big data under the problems and challenges introduced by the controllers of big data. The steps involved in the big data classification objectives are presented in the top figure of Fig. 2.2, which shows the processes involved with the management of big data, the configuration of big data technology, and the development of machine-learning techniques.

In the top figure of Fig. 2.2, the steps involved in the classification process are clearly presented: collecting the input data, understanding the data, shaping up the data (e.g., data cleaning and representation learning), and understanding the big data environment based on the hardware requirements and constraints of controllers. Finally, the understanding of the modeling and algorithms is also required for the success of big data classifications. The first bottom figure of Fig. 2.2 shows the specific parameters that influence the management of the big data controllers and lead

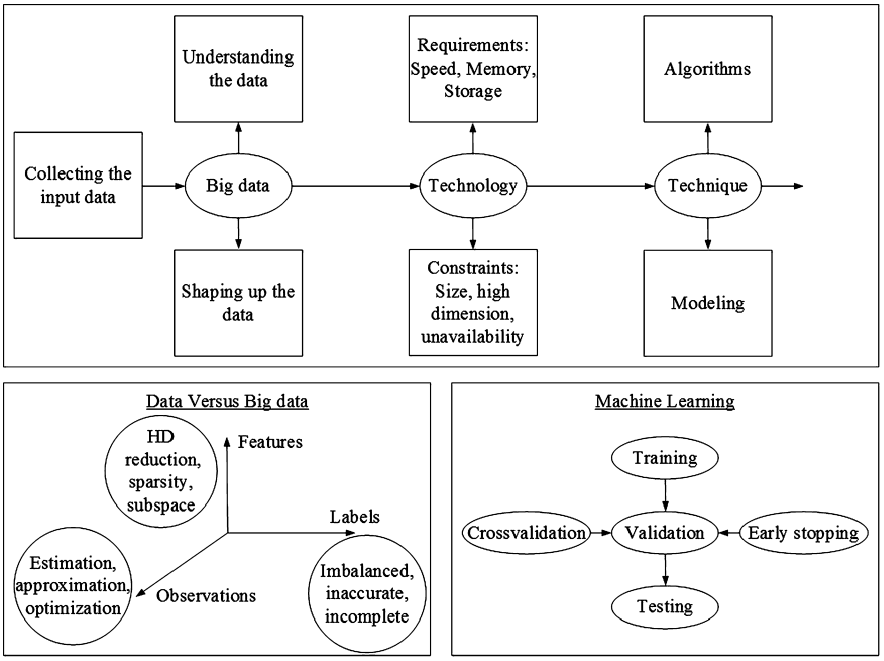

Fig. 2.2 *Top*: the classification processes of big data are illustrated. *Bottom left*: the classification modeling of big data is illustrated. *Bottom right*: the classification algorithms of big data are illustrated

to challenges in the development of learning models. In the second bottom figure of Fig. 2.2, the steps involved in learning algorithms are presented. It shows the flow from training phase to validation phase and then to testing phase. In the validation phase, cross-validation techniques can be applied and an early stopping decision may be made to avoid a so-called overfitting problem.

2.2.1 Representation Learning

The representation learning techniques [14, 15] are useful for understanding and shaping the data. These techniques require statistical measures and processes. Statistical measures like the mean, standard deviation, and covariance can help detect the patterns numerically. Similarly, the graphical tools like pie charts, histograms, and scatter plots can help in understanding the patterns. Statistical processes like normalization and standardization can manipulate data to extract and understand patterns. Representation learning mainly focuses on the big data controller feature, and its goal is the feature selection. Hence it contributes to the dimensionality

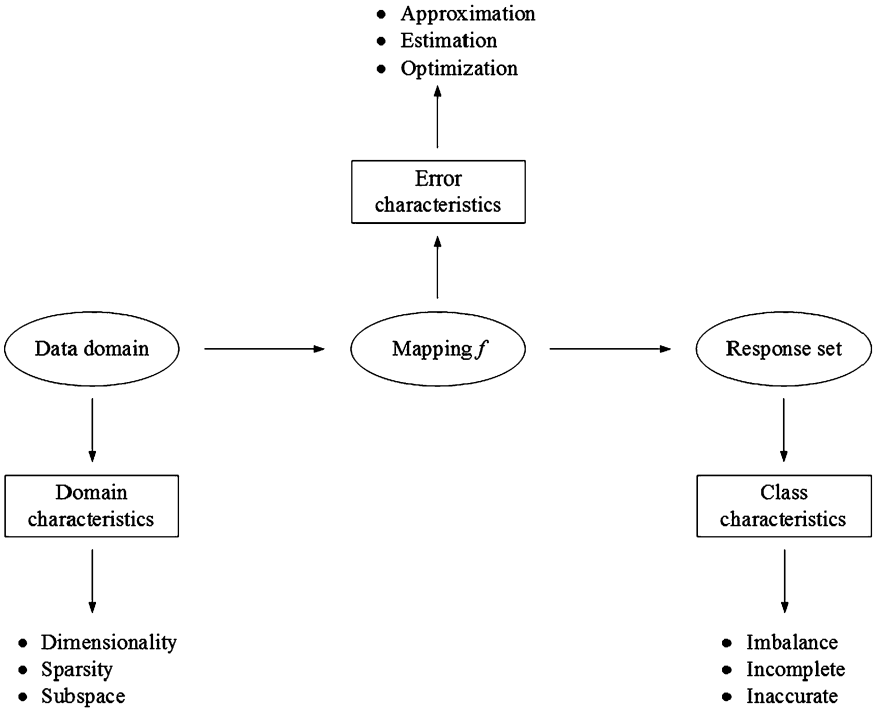

Fig. 2.3 Characteristics problems with the modeling

reduction objectives in machine learning. In big data analytics, the data sets grow dynamically; therefore, the representation learning techniques take the dynamically changing data characteristics into consideration. In general, representation learning techniques have been applied to understand the data, but it does not incorporate the domain division (class-separate) objectives. The recent cross-domain representation learning framework proposed by Tu and Sun [16] may be useful to understand the data for big data analytics.

2.2.2 Distributed File Systems

Distributed file systems are suitable for big data management, processing, and analysis [17]. They may be customized to satisfy the hardware requirements and remove computing environmental constraints. They must be configured to handle a large volume of data (big data storage), real-time data (big data on demand), and large varieties of computations associated with the data types (computer memory). The modern Hadoop distributed file system can be configured to meet these requirements, thus eliminating the constraints that arise from the size, dimensionality, and data unavailability for on demand applications.

2.2.3 Classification Modeling

Classification modeling was illustrated in Fig. 1.1 and discussed in Chap. 1. As we recall, it defines a map f between a data domain and a knowledge (or response) set. This definition may be extended to the analysis of class labels in order to describe the class characteristics defined by imbalanced [18], incomplete [19], and inaccurate data [20]; to the analysis of the observations in order to describe the error characteristics defined by the approximation, the estimation, and the optimization [21, 22] errors; and to the analysis of the features in order to describe the domain characteristics defined by the degree of dimensionality, the sparsity, and the subspace. This extended definition is illustrated in Fig. 2.3. It also shows the relationships between the three characteristics. It states that the approximation, estimation, and optimization issues must be taken into consideration, when the mapping f is defined, the dimensionality, sparsity, and subspace must be taken into consideration when the data domain is divided, and the imbalance, incomplete, and inaccurate class characteristics must be considered when the subdomains are mapped to the responses.

2.2.3.1 Class Characteristics

Imbalanced, incomplete, and inaccurate class characteristics can be defined in the response set portion of the modeling objective (see Fig. 2.3). These characteristics are influenced by the big data controllers: classes, features, and observations. We can describe these characteristics using simple examples.

Let us take two classes: $\{(1,5),(1.5,5.2),(2,4.6)\}$ and $\{(5,1),(6,0.5),(6.5,1.4)\}$. This is balanced data, the observations are represented by (x_1,x_2) with two features and the number of observations in each class is 3. If we assume these are the true observations, but a system generates $\{(1,5),(1.5,5.2),(2,4.6)\}$ and $\{(6,0.5),(6.5,1.4)\}$, then we can call this imbalanced data [18]. You will see a detailed explanation in Chap. 3. That is, if we have more observations in one class than in the other class, then we can say the data is imbalanced, and the smaller class is the minority class, and the other class is the majority class.

If the system generates $\{(1,5),(1.5,5.2),(2,4.6)\}$ and $\{(5,1),(6,0.5),(6.5,-)\}$ then we can call this incomplete data, because a class has missing information and is not complete. Similarly, if the system generates $\{(1,5),(1.5,5.2),(6.5,1.4)\}$ and $\{(5,1),(6,0.5),(2,4.6)\}$, then we can call it inaccurate data because class observations are labeled incorrectly.

This scenario is illustrated visually in Fig. 2.4. The top row shows two classes "class 1" and "class 2" with the balanced, complete, and accurate data. In the first image of the second row, an imbalanced data example is shown; in the second image of this row, an incomplete data example (data are missing) is illustrated; and in the third image, an example of inaccurate data (labels are switched) is shown. In [19], Little and Rubin provide different examples for explaining the patterns that are expected when the data is incomplete. These three class characteristics are the major players for the deformation of patterns.

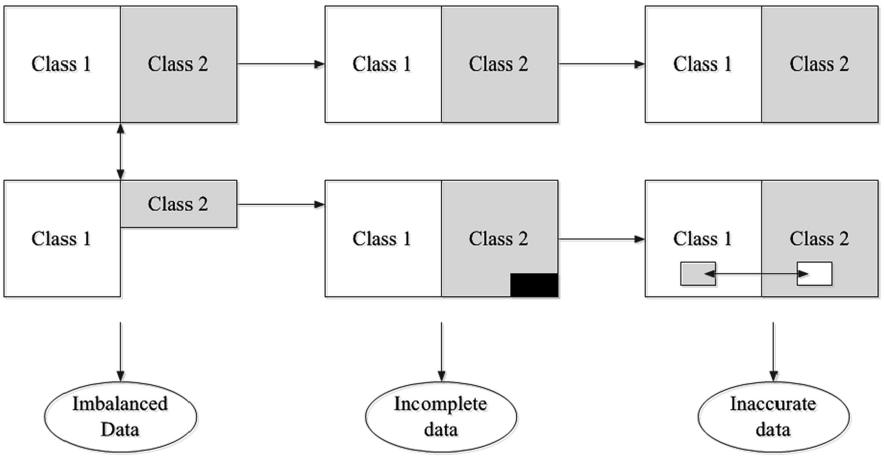

Fig. 2.4 Examples of imbalanced data, incomplete data, and inaccurate data based on the class labels and information abnormalities

2.2.3.2 Error Characteristics

The error characteristics can be defined in the mapping portion of the modeling process. This is shown in Fig. 2.3. Bottou and Bousquet [21] discuss the decomposition of classification errors using estimation, approximation, and optimization errors, and then Dalessandro [22] simplifies it in his paper. Based on these references, the estimation error may be defined as the differences in the models derived from the data sets of different sizes (e.g., finite and infinite sizes); the approximation error may be defined as the differences in the models derived from the parametrized models assumed (e.g., linear and nonlinear models); and the optimization error may be defined as the differences in the algorithms used to derive the models (e.g., efficient and inefficient algorithms). It is described in Fig. 2.5.

Suppose there is a true model, and we don't know that model; hence, we assume a model and develop a classification technique. Then the error between the true model and the model that we assumed will impact the accuracy of the classification model that we developed. This error is called the approximation error. Similarly, suppose there is a best algorithm, but we don't know that, and we develop an algorithm and use it for our classification. This error will impact the classification accuracy as well. This is called the optimization error. In the third error, suppose we use the true model and the best algorithm, in that case, we would get the actual results, but if we use our assumed model and the algorithm, we produce a different result. This error is called the estimation error. The task of big data classification is to minimize these three errors in the modeling and the algorithms. These three error characteristics are the major players for the classification errors.

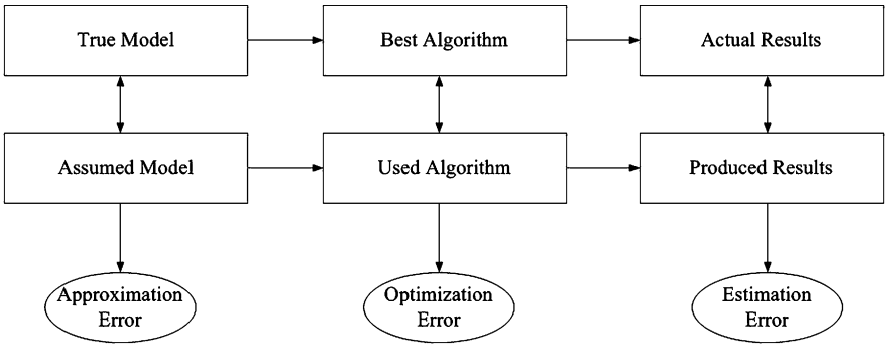

Fig. 2.5 Differences in modeling, algorithm and results can define approximation, optimization and estimation errors respectively

2.2.3.3 Domain Characteristics

Dimensionality, sparsity, and subspace can be defined based on the information of the features (one of the big data controllers). The number of features determines the dimensionality of the data. However, some of the features may not be relevant features, and they may not contribute to the patterns in the data. This can lead to dimensionality reduction, and the new space with fewer features is called the subspace. For example, take three observations with three features: (2,6,1.1), (4,8,2.2), and (3,7,0.5); they are drawn from two classes, "even" and "odd." They form a three-dimensional data domain. We can easily classify them with only the first two features, so it means new data points are (2,6), (4,8), and (3,7), and they now form a two-dimensional subspace. The first two points represent the "even" class and the last one represents "odd" class. Therefore, the subspaces can have low-dimensional structures that are useful for classification. Let us now define sparsity using a modified example. In this example, let us take the following three observations with three features: (2,0,0), (0,8,0), and (0,0,3). Here the features are sparse and they create the sparsity problem of big data classification.

2.2.4 Classification Algorithms

The classification algorithms mainly involve machine-learning processes, which are training, validation, and testing [4]. However, cross-validation and early stopping processes must be incorporated in the validation step.

2.2.4.1 Training

The training phase provides an algorithm to train the model. In other words, we can say that the parameters of a machine-learning model are estimated, approximated, and optimized using a labeled data set, the domain characteristics (dimensionality, sparsity, and subspace), and the class characteristics (imbalanced, incomplete, and inaccurate). The data set used in this phase is called the training set, and when a data set is called a training set then we can assume that it is a labeled data set. That is when the class labels are known.

2.2.4.2 Validation

The validation phase provides an algorithm to validate the effectiveness of the model using another data set, which was not used in the training phase. In this case, the data set is called the validation set, and it is also labeled. The validation phase helps to show that the parameters derived in the training phase work based on a quantitative measure. Hence, the quantitative measure plays a major role in this validation process. Some of these measures are entropy and root mean-squared error.

If the results are not satisfactory, then the model must be trained again to obtain better parameter values. This is the phase where the effects of the problems (the selection of an incorrect model, or the use of an inefficient algorithm) reported in Fig. 2.5 can be seen and corrected. The validation phase can also help to correct the over-training problem, which leads to the overfitting problem. The main technique used for this purpose is called the cross-validation [23].

2.2.4.3 Testing

This is a simple phase, and provides an algorithm to test if the trained and cross-validated model works using another data set, which was not used in the training or validation phases. In this algorithm, the labeled data set is used only to compare the results produced by the final model in terms of classification accuracy and computational time. Several measures are available for this purpose, and they are called qualitative measure as they are used to measure the performance of the model. Some of these measures are listed in [24], and they are: accuracy, sensitivity, specificity, and precision. These measures are used later in this book for performance analysis.

2.3 Big Data Scalability

Scalability is an unavoidable problem in big data applications [25]. Uncontrollable and continuous growth in the features create the scalability problem. In simple terms, the classification results obtained with a set of features expires instantaneously

because of the new additions in the features. Scalability occurs in high-dimensional systems, and it may be addressed using efficient representation learning and feature hashing algorithms.

2.3.1 High-Dimensional Systems

A large number of feature does not mean the data is high dimensional. A data set is high dimensional only if the number of features (p) of the data set are larger than the number of observations (n) in the data set. In big data analytics, the problem, challenges, and solution related to scalability have been treated separately, as it forms a separate problem space and gives significant challenges to different applications like text processing (spam filter) and forensic linguistics. The features are the main controller that contribute to this scalability problem and associated challenges. Hence, the scalable machine learning topic emerged into the big data paradigm. In this problem space, the features dynamically grow, and the system becomes high dimensional and difficult to manage. Therefore, one solution is to understand the patterns in low dimensions to develop efficient big data classification models and algorithms. Removal of irrelevant features can bring the number of features to less than the number of observations and, hence, help define low-dimensional structures.

2.3.2 Low-Dimensional Structures

In this section, two approaches are discussed: representation learning and feature hashing. Representation learning [14] provides models and algorithms to represent data at the preprocessing stage and help learn data characteristics through understanding of the roles of controllers and extracting geometrical and statistical structures. In [15], a simple representation learning technique, called a single-domain, representation-learning model, has been proposed, and its objective is to separate two classes over two-dimensional subspaces. It adopts the concept of unit-circle algorithm proposed in [26]. The main big data controller that may be manipulated to extract low-dimensional structures by generating subspaces in the representation-learning algorithms is the features.

Therefore, the techniques called *hashing techniques* have been proposed in the field to create low-dimensional subspaces. The feature hashing techniques provide dimensionality reduction to the data through the mapping of entire feature space to subspaces that are formed by subsets of features. Hence, it is sensitive to the inferiority of the algorithm used to generate such mappings.

Problems

2.1. Select several data sets from the University of California, Irvine, Machine Learning repository and explore: What is the purpose of the data sets? How many observations are there? How many features are there? How many classes are there? Is it useful to analyze the data sets? Will they evolve into big data sets?

2.2. Data Analytics

(a) Identify two data sets based on the answers that you found for the question in the first problem.
(b) Apply statistical analysis tools (such as scatter plots, histograms, pie charts, statistical distributions, etc.) and determine if the data sets are imbalanced, incomplete, or inaccurate.
(c) Make the data sets balanced, complete, and correct through randomization. Assume Gaussian properties for each feature, or conduct a distribution test to find a suitable distribution. Have you succeeded?

Acknowledgements I would like to thank Bin Yu and Richard Smith for the opportunities they gave me to attend several big data related workshops at their respective institutions. The knowledge gained from these workshops helped me write some of the topics presented in this book.

References

1. P. Russom, "Big data analytics," TDWI Best Practices Report, Fourth Quarter, Cosponsored by IBM, pp. 1–38, 2011.
2. S. Suthaharan. 2014. "Big Data Classification: Problems and challenges in network intrusion prediction with machine learning," ACM SIGMETRICS Performance Evaluation Review, vol. 41, no. 4, pp. 70–73.
3. J. Whitworth and S. Suthaharan. 2014. "Security problems and challenges in a machine learning-based hybrid big data processing network systems," ACM SIGMETRICS Performance Evaluation Review, vol. 41, no. 4, pp. 82–85.
4. S. B. Kotsiantis. "Supervised machine learning: A review of classification techniques," Informatica 31, pp. 249–268, 2007.
5. K. Weinberger, A. Dasgupta, J. Langford, A. Smola, and J. Attenberg. "Feature hashing for large scale multitask learning." In Proceedings of the 26th Annual International Conference on Machine Learning, pp. 1113–1120. ACM, 2009.
6. Q. Shi, J. Petterson, G. Dror, J. Langford, A. Smola, and V. Vishwanathan. "Hash kernels for structured data." The Journal of Machine Learning Research 10, pp. 2615–2637, 2009.
7. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition." Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.
8. T. Zhang. "Solving large scale linear prediction problems using stochastic gradient descent algorithms." In Proceedings of the International Conference on Machine learning, pp. 919–926, 2004.
9. P. Zikopoulos, C. Eaton, et al. Understanding big data: Analytics for enterprise class hadoop and streaming data. McGraw-Hill Osborne Media, 2011.
10. T. White. Hadoop: the definitive guide. O'Reilly, 2012.

11. J. Dean, and S. Ghemawat, S. “MapReduce: simplified data processing on large clusters.” Communications of the ACM, vol. 51, no. 1, pp. 107–113, 2008.
12. J. Dean, and S. Ghemawat. “MapReduce: a flexible data processing tool.” Communications of the ACM, vol. 53, no. 1, pp. 72–77, 2010.
13. H. Tong. “Big data classification,” Data Classification: Algorithms and Applications. Chapter 10. (Eds.) C.C. Aggarwal. Taylor and Francis Group, LLC. pp. 275–286. 2015.
14. Y. Bengio, A. Courville, and P. Vincent. “Representation learning: A review and new perspectives,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 8, pp. 1798–1828, 2013.
15. S. Suthaharan. “A single-domain, representation-learning model for big data classification of network intrusion,” Machine Learning and Data Mining in Pattern Recognition, Lecture Notes in Computer Science Volume 7988, pp. 296–310, 2013.
16. W. Tu, and S. Sun, “Cross-domain representation-learning framework with combination of class-separate and domain-merge objectives,” In: Proc. of the CDKD 2012 Conference, pp. 18–25, 2012.
17. K. Shvachko, H. Kuang, S. Radia, and R. Chansler. “The hadoop distributed file system,” In Proc. of the IEEE 26th Symposium on Mass Storage Systems and Technologies, pp. 1–10, 2010.
18. K. Kotipalli and S. Suthaharan. 2014. “Modeling of class imbalance using an empirical approach with spambase data set and random forest classification,” in Proceedings of the 3rd Annual Conference on Research in Information Technology, ACM, pp. 75–80.
19. R.J.A. Little and D.B. Rubin. “Statistical analysis with missing data,” Wiley Series in Probability and Statistics, John Wiley and Sons, Inc. second edition, 2002.
20. B. Frenay and M. Verleysen, “Classification in the presence of label noise: a survey,” IEEE Transactions on Neural Networks and Learning Systems, vol. 25, no. 5, pp. 845–869, 2014.
21. L. Bottou, and O. Bousquet. “The tradeoffs of large scale learning.” In Proceedings of NIPS, vol 4., p. 8, 2007.
22. B. Dalessandro. “Bring the noise: Embracing randomness is the key to scaling-up machine learning algorithms.” Big Data vol. 1, no. 2, pp. 110–112, 2013.
23. S. Arlot, and A. Celisse. “A survey of cross-validation procedures for model selection,” Statistics surveys, vol. 4, pp. 40–79, 2010.
24. Machine Learning Corner (Design models that learn from data), “Evaluation of Classifier’s Performance,” https://mlcorner.wordpress.com/tag/specificity/, Posted on April 30, 2013.
25. P. Domingos, and G. Hulten. “A general method for scaling-up machine learning algorithms and its application to clustering.” In Proceedings of the International Conference on Machine Learning, pp. 106–113. 2001.
26. S. Suthaharan. 2012. “A unit-circle classification algorithm to characterize back attack and normal traffic for network intrusion detection systems,” in Proceedings of the IEEE International Conference on Intelligence and Security Informatics, pp. 150–152.

http://www.springer.com/978-1-4899-7640-6

Machine Learning Models and Algorithms for Big Data Classification
Thinking with Examples for Effective Learning
Suthaharan, S.
2016, XIX, 359 p. 149 illus., 82 illus. in color., Hardcover
ISBN: 978-1-4899-7640-6