International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 02 | Feb 2019 www.irjet.net p-ISSN: 2395-0072

Machine Learning

Ragini Avhad

Lecturer, Computer Engg.Department, V.P.M's polytechnic, Thane, Maharashtra, India

---

Abstract - *The goal of machine learning is to program computers to use example data or experience to solve a given problem. Many successful applications of machine learning exist already, including systems that analyze past sales data to predict customer behavior, optimize robot behavior so that a task can be completed using minimum resources, and extract knowledge from bioinformatics data. Introduction to Machine Learning is a comprehensive textbook on the subject, covering a broad array of topics not usually included in introductory machine learning texts. To present a unified treatment of machine learning problems and solutions, it discusses many methods from different fields, including statistics, pattern recognition, neural networks, artificial intelligence, signal processing, control, and data mining.*

1. INTRODUCTION

Machine learning is a subfield of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people. Although machine learning is a field within computer science, it differs from traditional computational approaches. In traditional computing, algorithms are sets of explicitly programmed instructions used by computers to calculate or problem solve. Machine learning algorithms instead allow for computers to train on data inputs and use statistical analysis in order to output values that fall within a specific range. Because of this, machine learning facilitates computers in building models from sample data to automate decision-making processes based on data inputs. Any technology user today has benefitted from machine learning. Facial recognition technology allows social media platforms to help users tag and share photos of friends. Optical character recognition (OCR) technology converts images of text into movable type. Recommendation engines, powered by machine learning, suggest what movies or television shows to watch next based on user preferences. Self-driving cars that rely on machine learning to navigate may soon be available to consumers. Machine learning is a continuously developing field. Because of this, there are some considerations to keep in mind as you work with machine learning methodologies or analyze the impact of machine learning processes.

1.1. Mining the history data using AR

One of the shortcomings in traditional underwriting operation is that one may neglect the relationship between the applicants' information items. So the first step of this algorithm is to neaten and mine the insured's personal information and claim data using AR mining. We try to find the potential association and relationship between these items. In insurance practice, the amount of benefit determines the number of personal statement and physical examination items. In general, most small benefit and young applicants need no physical examinations. Thus, the data of these policyholders is not integrated relatively. Before the AR mining, we must neaten the original data. In AR mining we need discrete data. Thus, we must transform the continuous data into discrete data. The transformation method is to divide the definition domain of every continuous item into several intervals, then to distribute every value of the item into corresponding interval, finally to evaluate the new discrete item variable by the sequence number of the interval to which the original value belongs. We adopt the classic priori Algorithm in AR mining. Apriori Algorithm is the basis of all AR algorithms in recent researches. The theory of this algorithm is based on the so-called Apriori Attribute: all non-empty subsets of the frequent itemsets must be frequent. In Apriori Algorithm we repeat two steps: connecting and trimming. In the step of connecting, we produce high dimension itemsets by connecting the low dimension ones. To reduce operation complexity, in the step of trimming we delete the frequent itemsets which cannot exist according to Apriori Attribute. These two steps will be repeated until there is no higher dimension itemsets. Because of the small benefit amount applicants need not have physical examination. So, the insured database lacks some information. We can't directly use the Apriori Algorithm that is designed for the integrated database. But we can estimate the actual support level of one itemset by calculating the minimum possible support ratio (when minimum including) and maximum possible support ratio (when maximum including) We may get some association or relationship between the insured information items in insurance companies' database by means of AR mining. And we may simplify the data and make some preparation for the next classification process. The data mining makes great sense for improving the underwriting algorithm.

1.2. Learning Strategies

Machine learning employs the following two strategies:

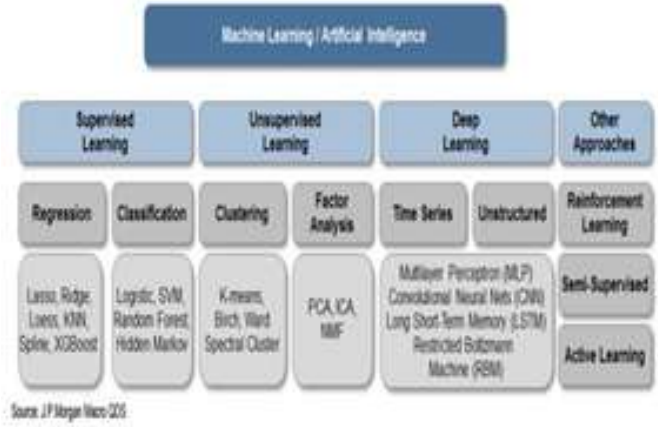
Supervised Learning

In supervised learning, the training set contains data and the correct output of the task with that data. This is like giving a student a set of problems and their solutions and telling that

student to gure out how to solve other problems he or she will have to deal with in the future. Supervised learning includes classification algorithms, which take as input a dataset and the class of each piece of data so that the computer can learn how to classify new data. For example, the input might be a set of past loan applications with an indication of which of them went bad. On the basis of this information, the computer classifies new loan applications. Classification can employ logic regression, classification trees, support vector machines, random forests, artificial neural networks (ANNs), or other algorithms. ANNs are a major topic on their own; we discuss them in more detail later. Regression algorithms predict a value of an entity's attribute ("regression" here has a wider sense than merely statistical regression). Regression algorithms include linear regression, decision trees, Bayesian networks, fuzzy classification, and ANNs.

Unsupervised Learning

In unsupervised learning, the training set contains data but no solutions; the computer must find the solutions on its own. This is like giving a student a set of patterns and asking him or her to figure out the underlying motifs that generated the patterns. Unsupervised learning includes clustering algorithms, which take as input a dataset covering various dimensions and partition it into clusters satisfying certain criteria. A popular algorithm is k-means clustering, which aims to partition the dataset so that each observation lies closest to the mean of its cluster. Other clustering approaches include hierarchical clustering, Gaussian mixture models, genetic algorithms (in which the computer learns the best way for a task through artificial selection), and ANNs. Dimensionality reduction algorithms take the initial dataset covering various dimensions and project the data to fewer dimensions. These fewer dimensions try to better capture the data's fundamental aspects. Dimensionality reduction algorithms include principal component analysis, tensor reduction, multidimensional statistics, random projection, and ANNs.



2. Essential Tools

Machine learning's popularity has brought along a wealth of tools. Most of them are open source, so users can easily experiment with them and learn how to use them. Table 1 compares some popular machine learning tools. The numerical and statistical communities are divided into two camps: one that prefers R and one that prefers Python. Of course, any absolute division makes no sense. For a field as wide as machine learning, no single tool will do. The best a software engineer can do is to become acquainted with many different tools and learn which one is the most appropriate for a given situation. That said, R is more popular with people with a somewhat stronger statistical background. It has a superb collection of machine-learning and statistical-inference libraries. Chances are, if you find a fancy algorithm somewhere and want to try it on your data, an implementation in R exists for it. R boasts the ggplot2 visualization library, which can produce excellent graphs. Python is more popular with people with a computer science background. Although not made specifically for machine learning or statistics, Python has extensive libraries for numerical computing (NumPy), scientific computing (SciPy), statistics (Stats Models), and machine learning (sickie-learn). These are largely wrappers of C code, so you get Python's convenience with C's speed. Although there are fewer machine learning libraries for Python than there are for R, many programmers find working with Python easier. They might already know the language or find it easier to learn than R. They also find Python convenient for preprocessing data: reading it from various sources, cleaning it, and bringing it to the required formats. For visualization, Python relies on matplotlib. You can do pretty much everything on matplotlib, but you might discover you have to put in some effort. The seaborn library is built on top of it, letting you produce elegant visualizations with little code. In general, R and Python work when the dataset fits in the computer's main memory. If that's not possible, you must use a distributed platform. The most well-known is Hadoop, but Hadoop isn't the most convenient for machine learning. Making even simple algorithms run on it can be a struggle. So, many people prefer to work at the higher level of abstraction that Spark offers. Spark leverages Hadoop but looks like a scripting environment. You can interact with it using Scala, Java, Python, or R. Spark has a machine-learning library that implements key algorithms, so for many purposes you don't need to implement anything yourself. H2O is a relatively newer entrant in the machine-learning scene. It's a platform for descriptive and predictive analytics that uses Hadoop and Spark; you can use it with R and Python. It implements supervised- and unsupervised-learning algorithms and a Web interface through which you can organize your workflow. A promising development is the Julia programming language for technical computing, which aims at top performance. Because Julia is new, it doesn't have nearly as many libraries as Python or R. Yet, thanks to its impressive speed, its popularity might grow. Strong commercial players include

MATLAB and SAS, which both have a distinguished history. MATLAB has long offered solid tools for numerical computation, to which it has added machine-learning algorithms and implementations. For engineers familiar with MATLAB, it might be a natural fit. SAS is a software suite for advanced statistical analysis; it also has added machine-learning capabilities and is popular for business intelligence tasks.

3. CONCLUSION

Many machine-learning books have a practical slant, aiming to introduce machine learning on a particular platform. As technologies quickly evolve, it's better to focus on getting a solid grasp of the fundamentals. After all, using a machine-learning platform isn't difficult; knowing when to use a particular algorithm and how to use it well requires quite a bit of background knowledge.

REFERENCES

[1] Machine Learning: (Andrew Ng, Stanford University via Coursera)

[2] Machine Learning: (Professor John W. Paisley, Columbia University via edX)

[3] Andrew Ng's Machine Learning course on Coursera.