Jimma University

Institute of Technology

Faculty of Computing

Title：-Supervised Learning in Data Mining

Group Members	ID
1. Abdulkerim Jemal	RU0724/08
2. Umer Aliyi	RU0766/08

Jimma, Ethiopia

Feb 2018

1. Introduction

Data mining techniques come in two main forms: supervised (*also known as predictive or directed*) and unsupervised (*also known as descriptive or undirected*). Both categories encompass functions capable of finding different hidden patterns in large data sets.

Supervised learning is a method used to enable machines to classify objects, problems or situations based on related data fed into the machines. Machines are fed with data such as characteristics, patterns, dimensions, color and height of objects, people or situations repetitively until the machines are able to perform accurate classifications. Supervised learning is a popular technology or concept that is applied to real-life scenarios. Supervised learning is used to provide product recommendations, segment customers based on customer data, diagnose disease based on previous symptoms and perform many other tasks. Supervised methods are methods that attempt to discover the relationship between input attributes (sometimes called independent variables) and a target attribute (sometimes referred to as a dependent variable). The relationship discovered is represented in a structure referred to as a model. Usually, models describe and explain phenomena, which are hidden in the dataset and can be used for predicting the value of the target attribute knowing the values of the input attributes. The supervised methods can be implemented in a variety of domains such as marketing, finance, and manufacturing.

During supervised learning, a machine is given data, known as training data in data mining parlance, based on which the machine does classification. For example, if a system is required to classify fruit, it would be given training data such as color, shapes, dimension and size. Based on this data, it would be able to classify fruit. Usually a system requires multiple iterations of such process to be able to perform accurate classification. Since real-life classifications such as credit card fraud detection and disease classification are complex tasks, the machines need appropriate data and several iterations of learning sessions to achieve reasonable abilities.

It is useful to distinguish between two main supervised models: classification models (classifiers) and Regression Models. Regression models map the input space into a real-value domain. For instance, a regressor can predict the demand for a certain product given its characteristics. On the other hand, classifiers map the input space into pre-defined classes.

Classifiers can be used to classify mortgage consumers as good (fully pay back the mortgage on time) and bad (delayed payback). There are many alternatives for representing classifiers, for example, support vector machines, decision trees, probabilistic summaries, algebraic function, etc. Along with regression and probability estimation, classification is one of the most studied models, possibly one with the greatest practical relevance. The potential benefits of progress in classification are immense since the technique has a great impact on other areas, both within Data Mining and in its applications.

In supervised learning or class prediction, knowledge of a particular domain is used to help make distinctions of interest. In life sciences, analyses tend to involve selecting the features most correlated with a phenotypic distinction. The features are then used as the input to a classification algorithm that uses known sample labeling to build a model, so that future unknown samples can be classified. For example, a model could be built to identify which sub-type of cancer a patient has based upon a subset of expressed genes that distinguish the cancer types of interest. Supervised learning classifiers can be very accurate in molecular classification, especially if a large number of high quality samples are used to train the model.

Now let’s clarify that with some specific demonstrations:

1. CLASSIFICATION

Classification of a collection consists of dividing the items that make up the collection into categories or classes. In the context of data mining, classification is done using a model that is built on historical data. The goal of predictive classification is to accurately predict the target class for each record in new data, that is, data that is not in the historical data.

A classification task begins with build data (also known as training data) for which the target values (or class assignments) are known. Different classification algorithms use different techniques for finding relations between the predictor attributes' values and the target attribute's values in the build data. These relations are summarized in a model; the model can then be applied to new cases with unknown target values to predict target values. A classification model can also be applied to data that was held aside from the training data to compare the predictions to the known target values; such data is also known as test data or evaluation data. The comparison technique is called testing a model, which measures the model's predictive accuracy.

The application of a classification model to new data is called applying the model, and the data is called apply data or scoring data. Applying a model to data is often called scoring the data.

Classification is used in customer segmentation, business modeling, credit analysis, and many other applications. For example, a credit card company may wish to predict which customers are likely to default on their payments. Customers are divided into two classes: those who default and those who do not default. Each customer corresponds to a case; data for each case might consist of a number of attributes that describe the customer's spending habits, income, demographic attributes, etc. These are the predictor attributes. The target attribute indicates whether or not the customer has defaulted. The build data is used to build a model that predicts whether new customers are likely to default.

Classification problems can have either binary or multiclass targets. Binary targets are those that take on only two values, for example, good credit risk and poor credit risk. Multiclass targets have more than two values, for example, the product purchased (comb or hair brush or hair pin). Multiclass target values are not assumed to exist in an ordered relation to each other, for example, hair brush is not assumed to be greater or less than comb.

As a supervised data mining method, classification begins with the method described above. Imagine you’re a credit card company and you want to know which customers are likely to default on their payments in the next few years.

You use the data on customers who have and have not defaulted for extended periods of time as build data (or training data) to generate a classification model. You then run that model on the customers you’re curious about. The algorithms will look for customers whose attributes match the attribute patterns of previous defaulters/non-defaulters, and categorize them according to which group they most closely match. You can then use these groupings as indicators of which customers are most likely to default.

Similarly, a classification model can have more than two possible values in the target attribute. The values could be anything from the shirt colors they’re most likely to buy, the promotional methods they’ll respond to (mail, email, phone), or whether or not they’ll use a coupon.

1. REGRESSION

Regression models are similar to classification models. The difference between regression and classification is that regression deals with numerical or continuous target attributes, whereas classification deals with discrete or categorical target attributes. In other words, if the target attribute contains continuous (floating-point) values or integer values that have inherent order, a regression technique can be used. If the target attribute contains categorical values, that is, string or integer values where order has no significance, a classification technique is called for. Note that a continuous target can be turned into a discrete target by binning; this turns a regression problem into a problem that can be solved using classification algorithms. To reuse the credit card example, if you wanted to know what threshold of debt new customers are likely to accumulate on their credit card, you would use a regression model.

Simply supply data from current and past customers with their maximum previous debt level as the target value, and a regression model will be built on that training data. Once run on the new customers, the regression model will match attribute values with predicted maximum debt levels and assign the predictions to each customer accordingly. This could be used to predict the age of customers with demographic and purchasing data, or to predict the frequency of insurance claims.

2. Descriptions of Basic algorithms

Some of the most used algorithms are:

1. K—Nearest Neighbor
2. Decision Trees
3. Naive Bayes
4. Support Vector Machines

K-Nearest Neighbors:

An algorithm is said to be a Lazy Learner if it simply stores the tuples of the training set and waits until the test tuple is given. Only when it sees the test tuple does it perform generalization to classify the tuple based on its similarity to the stored training tuples. K -Nearest Neighbor Classifier is a lazy learner.

KNN is based on learning by analogy, that is, by comparing a given test tuple with training tuples that are similar to it. The training tuples are described by n attributes. Each tuple represents a point in an n-dimensional space. In this way, all training tuples are stored in n-dimensional pattern space. When given an unknown tuple, a k-nearest neighbor classifier searches the pattern space for the k training tuples that are closest to the unknown tuple. These k training tuples are the k "nearest neighbors" of the unknown tuple. "Closeness" is defined regarding a distance metric, such as Euclidean distance. A good value for K is determined experimentally.

In pattern recognition, the k-nearest neighbors algorithm (k-NN) 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 plurality 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 k nearest neighbors. k-NN is a type of instance-based learning, or lazy 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. A peculiarity of the k-NN algorithm is that it is sensitive to the local structure of the data.

Decision Trees

Decision tree learning uses a decision tree (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). It is one of the predictive modeling approaches used in statistics, data mining and machine learning. Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees.

In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. In data mining, a decision tree describes data (but the resulting classification tree can be an input for decision making).

Naive Bayes

Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. There is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness, and diameter features.

For some types of probability models, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without accepting Bayesian probability or using any Bayesian methods.

An advantage of naive Bayes is that it only requires a small number of training data to estimate the parameters necessary for classification.

Support Vector Machines

In machine learning, support-vector machines are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier although methods such as Platt scaling exist to use SVM in a probabilistic classification setting. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces. When data is unlabeled, supervised learning is not possible, and an unsupervised learning approach is required, which attempts to find natural clustering of the data to groups, and then map new data to these formed groups.

3. Application areas in data mining where it can be applied and Used

Supervised learning can be applied on:

- Bioinformatics
- Database marketing
- Handwriting recognition
- Information retrieval
- Learning to rank
- Information extraction
- Object recognition in computer vision
- Optical character recognition
- Spam detection
- Pattern recognition
- Speech recognition

4. Conclusion

Data mining is becoming an essential aspect in the current business world due to increased raw data that organizations need to analyze and process so that they can make sound and reliable decisions. Supervised data mining techniques are appropriate when you have a specific target value you'd like to predict about your data. The targets can have two or more possible outcomes, or even be a continuous numeric value. To use these methods, you ideally have a subset of data points for which this target value is already known. You use that data to build a model of what a typical data point looks like when it has one of the various target values. You then apply that model to data for which that target value is currently unknown. The algorithm identifies the "*new*" data points that match the model of each target value.