Summary and Overview of Concepts in Machine Learning

Machine learning is divided into supervised learning and unsupervised learning. Supervised learning is divided into single models and ensemble learning. Single models include linear models, KNN, decision trees, BP neural networks, support vector machines, and naive Bayes. Linear models encompass linear regression, logistic regression, and Lasso. Ensemble learning includes boosting and bagging. Boosting comprises GBDT, adaboost, xgboost, lightgbm, and catboost. Bagging includes random forests and extra trees. Unsupervised learning is divided into clustering and dimensionality reduction. Clustering includes K-Means algorithm and hierarchical clustering, while dimensionality reduction includes PCA (Principal Component Analysis), SVD (Singular Value Decomposition), and LDA (Linear Discriminant Analysis).

1，Supervised Learning

Supervised learning typically involves training a model using labeled data with expert annotations, learning a function mapping from input variable X to output variable Y. Y = f(X). The training data usually takes the form of (n x, y), where n represents the size of the training samples, and x and y are the respective sample values of variables X and Y.

Supervised learning can be categorized into two main types:

Classification problems: Predicting the category or class to which a particular sample belongs (discrete). For instance, determining gender, health status, etc.

Regression problems: Predicting a real-number output associated with a particular sample (continuous). For example, predicting the average height of individuals in a specific region.

Additionally, ensemble learning is another form of supervised learning. It involves combining predictions from multiple different, relatively weaker machine learning models to forecast outcomes for new samples.

1.1 Single Model

1.11 Linear Regression

Linear regression refers to a regression model entirely composed of linear variables. In linear regression analysis, there is only one independent variable and one dependent variable, and their relationship can be approximated by a straight line. This type of regression analysis is called simple linear regression analysis. If the regression analysis involves two or more independent variables, and there exists a linear relationship between the dependent variable and the independent variables, it is referred to as multiple linear regression analysis.

1.12 Logistic Regression

Logistic regression is used to study the relationship between variables X and Y when Y represents categorical data. When Y has two categories, such as 0 and 1 (for instance, 1 indicating willingness and 0 indicating unwillingness, or 1 indicating purchase and 0 indicating non-purchase), it is termed binary logistic regression. If Y has three or more categories, it is referred to as multinomial logistic regression.

The independent variables do not necessarily have to be categorical; they can also be quantitative variables. When X is categorical data, dummy variables need to be created for X in the analysis.

1.13 Lasso

The Lasso method is an alternative regularization technique to the least squares method. Its fundamental concept involves constructing an L1 regularized model. During the model creation process, some coefficients are shrunk and set to zero. Once the model training is completed, these parameters with zero weights can be eliminated, simplifying the model. This process effectively prevents overfitting and is widely used for fitting data with multicollinearity and for variable selection.

1.14 K-Nearest Neighbors (KNN)

The primary difference between KNN in regression and classification lies in the decision-making process during prediction.

For KNN in classification predictions, the typical approach is using a majority voting method. This involves selecting the K nearest samples in the training set to the predicted sample based on their features and assigning the predicted sample to the class that has the highest frequency among these K nearest neighbors.

In KNN regression, the general method used is averaging, where the predicted value is the average of the output values of the K nearest samples.

However, the underlying theory for both classification and regression in KNN remains the same.

1.15 Decision Tree

In a decision tree, every internal node represents a splitting condition. It specifies a test on an attribute of instances, segregating the samples arriving at that node based on a particular property. Each subsequent branch from the node corresponds to a possible value of that attribute.

In classification trees, the mode of the output variable among the samples within leaf nodes determines the classification result. For regression trees, the average of the output variable among the samples within leaf nodes represents the prediction result.

1.16 Backpropagation Neural Network (BPNN)

The backpropagation neural network is a multi-layer feedforward network trained using the error backpropagation algorithm. It's one of the most widely used neural network models. The learning rule of a BP neural network employs the steepest descent method. It continuously adjusts the network's weights and thresholds through backpropagation to minimize the network's classification error rate (minimize the sum of squared errors).

The BP neural network is a multi-layer feedforward neural network characterized by forward signal propagation and backward error propagation. Specifically, for a neural network model containing only one hidden layer:

The process of the BP neural network mainly consists of two stages. The first stage is the forward propagation of signals, starting from the input layer through the hidden layer, and finally reaching the output layer. The second stage is the backward propagation of errors, starting from the output layer, through the hidden layer, and finally reaching the input layer. This process sequentially adjusts the weights and biases from the hidden layer to the output layer, and from the input layer to the hidden layer.

1.17 Support Vector Machine (SVM)

Support Vector Machine Regression (SVR) utilizes non-linear mapping to project data into a high-dimensional feature space, enabling the variables and outcomes to exhibit good linear regression

characteristics in this high-dimensional feature space. After fitting in this feature space, the results are then returned to the original space.

Support Vector Machine Classification (SVM) is a type of generalized linear classifier that performs binary classification on data in a supervised learning manner. Its decision boundary is the maximum margin hyperplane derived from the training samples.

1.18 Naive Bayes

Naive Bayes involves computing the probability of one event occurring given that another event has occurred, using Bayes' theorem. Given prior knowledge (denoted as "d"), to calculate the probability of our hypothesis "h" being true, we apply Bayes' theorem.

$$P(h|d) = \frac{P(d|h) * P(h)}{P(d)}$$

This algorithm assumes that all variables are independent of each other.

1.2 Ensemble Learning

Ensemble learning is a technique that combines the results of different learning models (such as classifiers) to further enhance accuracy through voting or averaging. Typically, voting is used for classification problems, while averaging is employed for regression problems.

Ensemble algorithms primarily consist of three main types: Bagging, Boosting, and Stacking. This discussion will not cover Stacking.

1.21 Gradient Boosting Decision Trees (GBDT)

GBDT is a boosting algorithm that uses CART (Classification and Regression Trees) regression trees as the base learners. It operates as an additive model, sequentially training a set of CART regression trees. Ultimately, it combines the predictions of all the regression trees to form a strong learner. Each new tree is fitted in the negative gradient direction of the current loss function. Finally, the output is the sum of this ensemble of regression trees, providing direct regression results or applying sigmoid or softmax functions to obtain binary or multiclass classification results.

1.22 Adaboost

Adaboost assigns higher weights to the learners with lower error rates and lower weights to learners with higher error rates. It combines weak learners with their corresponding weights to generate a strong learner. The main difference between the algorithms for regression and classification problems lies in the way error rates are calculated. For classification problems, a 0/1 loss function is commonly used, while regression problems generally use a square loss function or a linear loss function.

1.23 XGBoost

XGBoost, short for "Extreme Gradient Boosting," is a type of ensemble algorithm that combines base functions and weights to create a highly effective fitting algorithm for data. Since its introduction in 2015, XGBoost has gained popularity in the fields of statistics, data mining, and machine learning due to its strong generalization ability, high scalability, and fast computational speed.

XGBoost is an efficient implementation of Gradient Boosting Decision Trees (GBDT). However, it differs from GBDT in that XGBoost introduces a regularization term to the loss function. Additionally, since calculating derivatives for some loss functions might be challenging, XGBoost employs the second-order Taylor expansion of the loss function for fitting.

1.24 LightGBM

LightGBM is an efficient implementation similar to XGBoost. Its concept involves discretizing continuous floating-point features into k discrete values and constructing a histogram with a width of k. It then traverses the training data, computing cumulative statistics for each discrete value in the histogram.

During feature selection, it only needs to search for the optimal split point based on the discrete values of the histogram. LightGBM employs a leaf-wise growth strategy with depth constraints, which saves significant time and space overhead. This strategy allows for more efficient computation compared to traditional depth-first growth methods used in other boosting algorithms.

1.25 CatBoost

CatBoost is a GBDT (Gradient Boosting Decision Trees) framework based on symmetric decision tree algorithms. Its primary focus is efficiently handling categorical features and addressing gradient bias and prediction shift issues to enhance the algorithm's accuracy and generalization capabilities.

1.26 Random Forest

In the process of creating numerous decision trees in random forest classification, sampling is conducted randomly on both the observations in the modeling dataset and the feature variables. Each tree is generated based on one of these random samples, producing rules and classification results specific to that tree. The forest then integrates all the decision trees' rules and classification results to achieve the classification (or regression) carried out by the random forest algorithm.

1.27 Extra Trees

Extra Trees, also known as extremely randomized trees, are quite similar to random forests. The term "extremely" random is manifested in the splitting of nodes in decision trees. Extra Trees take an even more random approach by directly using random features and random thresholds for partitioning nodes. Consequently, each decision tree's shape and differences will be more varied and even more random compared to standard random forests.

2，Unsupervised Learning

Unsupervised learning deals with training data that only consists of input variables (X) without corresponding output variables. It models the structure of data using unlabeled training data without expert annotations.

2.1 Clustering

Clustering involves grouping similar samples into clusters. Unlike classification problems, clustering doesn't have prior knowledge of categories, and the training data naturally lacks category labels.

2.11 K-means Algorithm

Clustering analysis, specifically the K-means algorithm, is a centroid-based clustering method. It iteratively allocates samples to K clusters, aiming to minimize the sum of distances between each sample and the mean or centroid of its assigned cluster. Unlike hierarchical clustering and other algorithms that cluster based on attributes, K-means clustering organizes samples into clusters based on sample proximity.

2.12 Hierarchical Clustering

Hierarchical clustering is a clustering method that decomposes a given set of data objects based on a hierarchy determined by the clustering strategy employed. This algorithm creates clusters by building a tree structure with clusters as nodes.

If the hierarchy is constructed from the bottom up, it's called agglomerative hierarchical clustering (e.g., AGNES). Conversely, if it is built from the top down, it's called divisive hierarchical clustering (e.g., DIANA). Agglomerative hierarchical clustering, forming clusters from the bottom up, is commonly used in practice.

2.2 Dimensionality Reduction

Dimensionality reduction refers to reducing the number of features in a dataset while retaining the meaningful information. This is achieved through feature extraction and feature selection methods.

Feature selection involves choosing a subset of the original variables. On the other hand, feature extraction involves transforming data from a high-dimensional space to a lower-dimensional space. Principal Component Analysis (PCA) is a well-known algorithm used for feature extraction.

2.21 PCA (Principal Component Analysis) is a method used for dimensionality reduction by creating linear combinations of correlated indicators. Its goal is to explain as much information as possible from the original data in fewer dimensions. The variables become linearly independent of each other after dimensionality reduction, and the new variables formed are linear combinations of the original variables. Additionally, the further the principal components, the less they contribute to the variance, signifying a weaker overall information representation.

2.22 Singular Value Decomposition (SVD) is a widely used algorithm in the field of machine learning. It is not only employed for feature value decomposition in dimensionality reduction algorithms but also finds applications in recommendation systems, natural language processing, and serves as a cornerstone for many algorithms.

2.23 LDA (Linear Discriminant Analysis)

The principle behind linear discriminant analysis is to project samples onto a straight line, aiming to keep the projection points of similar samples as close together as possible while ensuring that points of different samples are as far apart as possible. When classifying new samples, they are projected onto the same line, and their position along the line is used to determine the category of the new sample.