## SUPERVISED LEARNING

**Classification:** A type of supervised learning where the target is a class. The model learns to produce a class score and to assign each vector of input features to the class with the highest score. A cost can be introduced to penalize one of the classes during class assignment.

**Decision Tree:** Follows the C4.5 decision tree algorithm. These algorithms generate a tree-like model of decisions. The input data is recursively split into subsets based on one of the input features, generating two or more branches. The other output features are made in subsequent nodes until a node is generated where all of almost all of the data belong to the same class.

**Logistic Regression:** A statistical algorithm that models the relationship between the input features and the categorical output classes by maximizing a likelihood function. Originally extended to problems with more than two classes (multinomial logistic regression).

**Naive Bayes:** Based on Bayes theorem and assuming statistical independence between input features. The algorithm estimates the conditional probability of each output class given the vector of input features. The class with the highest conditional probability is assigned to the input.

**Support Vector Machine (SVM):** A supervised learning algorithm for classification or regression. Based on the concept of discriminating hyperplanes in high-dimensional feature spaces. SVMs can perform classification or regression depending on whether a linear or a non-linear mapping of the input space is needed. SVMs implicitly map their inputs into high-dimensional feature spaces, where the classes are linearly separable.

**k-Nearest Neighbor (kNN):** A non-parametric method that assigns the class of the k-th nearest neighbors to the point. The value of k is an adjustable parameter in the range [1, infty]. Class attribution can be weighted by the distance to the k-th point or/and by the class probability.

**Ensemble Learning:** A combination of multiple models from supervised learning algorithms to make more stable and accurate overall model. Most commonly used ensemble techniques are Bagging and Boosting.

### BAGGING

**Bagging:** A method for training multiple classification/regression models on different subsets of data. The final prediction is based on the average of all predictions, which reduces the chance of overfitting.

### BOOSTING

**Boosting:** A method for training a set of classification/regression models iteratively. At each step, a new model is trained on prediction error added to the ensemble to improve the results from the previous model, leading to higher accuracy after each iteration.

**Random Forest of Decision/Regression Trees:** Ensemble model of multiple decision regression trees trained on different subsets of data. Each subset has more samples or less or equal columns are bootstrapped from the original training set. The prediction is based on the majority vote of all trees (averaging all probabilities or numeric predictions) on all involved trees.

### DEPLOYMENT

**Model Loader:** A representation of a classification model with false positive rates on the x-axis and true positive rates on the y-axis. A decision boundary for generalization is obtained for different classification models using a geometrical algorithm.

## UNSUPERVISED LEARNING

**Unsupervised Learning:** A set of machine learning algorithms to discover patterns in the data. A labeled dataset is not required, since data are usually organized and/or transformed based on similarity measures or statistical features.

**Clustering:** A branch of unsupervised learning algorithms that group data together based on similarity measures, without the help of labels. The most dissimilar (distance a priori) data points and clusters, according to a selected distance measure. K-means: clustering around the most distant data point in the cluster.

**Hierarchical Clustering:** Builds a hierarchy of clusters by either choosing the overall best clustering (agglomerative approach) or separating the most dissimilar (divisive approach) data points and clusters, according to a selected distance measure. Sota Learner: Similar to SOTA Learner (agglomerative) or separating the data in different mutually exclusive (divisible).

## RECOMMENDATION ENGINES

**Recommendation Engines:** A set of algorithms that learn user information about user preferences to predict items of interest.

**Association Rules:** The notion of association is widely used in co-occurrences of multiple products in labels, classes or categories. Based on the a-priori algorithm, rules are created that describe the most frequent items in the dataset are used to generate rules. Rule Learner: a type of recommendation system that learns regularities in data.

**Collaborative Filtering:** Based on the idea that users who have similar interests in the past (e.g., based on SVD techniques) will have similar interests in the future. The preferences of users are modeled as preferences of multiple users (collaborating).

**Confusion Matrix:** A representation of a classification model where each cell shows the number of samples that have a true class equal to the row class and a predicted class equal to the column class. The cluster prototype is taken as the average of the points in a leaf and/or by the class probability.

**SARIMA:** A linear regression with time series variables. The model learns to associate one or more numbers with the vector of past values. SARIMA model parameters are estimated concurrently by various algorithms, mostly following the Box-Jenkins approach.

## CLUSTERING

**Decision Trees:** A set of machine learning algorithms to predict the value of a target class or variable. They produce a mapping function that maximizes a likelihood function. Originally extended to problems with more than two classes (multinomial logistic regression).

**Ensemble Learning:** A type of supervised learning where the target is a class. The model learns to produce a class score and to assign each vector of input features to the class with the highest score. A cost can be introduced to penalize one of the classes during class assignment.

**Logistic Regression:** A statistical algorithm that models the relationship between the input features and the categorical output classes by maximizing a likelihood function. Originally extended to problems with more than two classes (multinomial logistic regression).

**Naive Bayes:** Based on Bayes theorem and assuming statistical independence between input features. The algorithm estimates the conditional probability of each output class given the vector of input features. The class with the highest conditional probability is assigned to the input.

**Support Vector Machine (SVM):** A supervised learning algorithm for classification or regression. Based on the concept of discriminating hyperplanes in high-dimensional feature spaces. SVMs can perform classification or regression depending on whether a linear or a non-linear mapping of the input space is needed. SVMs implicitly map their inputs into high-dimensional feature spaces, where the classes are linearly separable.

**k-Nearest Neighbor (kNN):** A non-parametric method that assigns the class of the k-th nearest neighbors to the point. The value of k is an adjustable parameter in the range [1, infty]. Class attribution can be weighted by the distance to the k-th point or/and by the class probability.

**Ensemble Learning:** A combination of multiple models from supervised learning algorithms to make more stable and accurate overall model. Most commonly used ensemble techniques are Bagging and Boosting.

### BAGGING

**Bagging:** A method for training multiple classification/regression models on different subsets of data. The final prediction is based on the average of all predictions, which reduces the chance of overfitting.

### BOOSTING

**Boosting:** A method for training a set of classification/regression models iteratively. At each step, a new model is trained on prediction error added to the ensemble to improve the results from the previous model, leading to higher accuracy after each iteration.

**Random Forest of Decision/Regression Trees:** Ensemble model of multiple decision regression trees trained on different subsets of data. Each subset has more samples or less or equal columns are bootstrapped from the original training set. The prediction is based on the majority vote of all trees (averaging all probabilities or numeric predictions) on all involved trees.

## DEPLOYMENT

**Model Loader:** A representation of a classification model with false positive rates on the x-axis and true positive rates on the y-axis. A decision boundary for generalization is obtained for different classification models using a geometrical algorithm.

**Confusion Matrix:** A representation of a classification model where each cell shows the number of samples that have a true class equal to the row class and a predicted class equal to the column class. The cluster prototype is taken as the average of the points in a leaf and/or by the class probability.

**SARIMA:** A linear regression with time series variables. The model learns to associate one or more numbers with the vector of past values. SARIMA model parameters are estimated concurrently by various algorithms, mostly following the Box-Jenkins approach.

## RECOMMENDATION ENGINES

**Recommendation Engines:** A set of algorithms that learn user information about user preferences to predict items of interest.

**Association Rules:** The notion of association is widely used in co-occurrences of multiple products in labels, classes or categories. Based on the a-priori algorithm, rules are created that describe the most frequent items in the dataset are used to generate rules. Rule Learner: a type of recommendation system that learns regularities in data.

**Collaborative Filtering:** Based on the idea that users who have similar interests in the past (e.g., based on SVD techniques) will have similar interests in the future. The preferences of users are modeled as preferences of multiple users (collaborating).

## CLUSTERING

**Decision Trees:** A set of machine learning algorithms to predict the value of a target class or variable. They produce a mapping function (model) from the input features to the target class/variable. To estimate the model parameters during the training phase, labeled example data are needed in the training set.

**Evaluation:** Various scoring metrics for assessing model quality - in particular, a model’s predictive ability or propensity to error.

**Cross-Validation:** A set of machine learning algorithms to predict the value of a target class or variable. They produce a mapping function (model) from the input features to the target class/variable. To estimate the model parameters during the training phase, labeled example data are needed in the training set.

**Evaluation:** Various scoring metrics for assessing model quality - in particular, a model’s predictive ability or propensity to error.

**Cross-Validation:** A set of machine learning algorithms to predict the value of a target class or variable. They produce a mapping function (model) from the input features to the target class/variable. To estimate the model parameters during the training phase, labeled example data are needed in the training set.

**Evaluation:** Various scoring metrics for assessing model quality - in particular, a model’s predictive ability or propensity to error.

**Cross-Validation:** A set of machine learning algorithms to predict the value of a target class or variable. They produce a mapping function (model) from the input features to the target class/variable. To estimate the model parameters during the training phase, labeled example data are needed in the training set.

**Evaluation:** Various scoring metrics for assessing model quality - in particular, a model’s predictive ability or propensity to error.
Extend your KNIME knowledge with our collection of books from KNIME Press. For beginner and advanced users, through to those interested in specialty topics such as topic detection, data blending, and classic solutions to common use cases using KNIME Analytics Platform - there's something for everyone. Available for download at www.knime.com/knimepress.

A Guide to KNIME Analytics Platform for Beginners

File Reader
Partitioning
Decision Tree
Scorer

File Reader
Partitioning
Decision Tree
Scorer

A Guide to KNIME Analytics Platform for Advanced Users

Data Blending with KNIME

Consume &
Interact
Monitor & Update
Validate & Deploy
Production

Model &
Visualize
Optimize & Capture
Blend & Transform
Production
ProcessCreation

From Modeling to Model Evaluation

KNIME for Life Sciences

Human-in-the-Loop Label Generation with Active Learning & Weak Supervision

Guided Labeling

Written by
Raul Tamagno

from Excel to KNIME

from SAS to KNIME

from Alteryx to KNIME

© 2022 KNIME AG. All rights reserved. The KNIME® trademark and logo and OPEN FOR INNOVATION® trademark are used by KNIME AG under license from KNIME GmbH, and are registered in the United States. KNIME® is also registered in Germany.