Introducing Machine Learning

Jan-Philipp Kolb

22 Mai, 2019

What Is Machine Learning?

- Machine learning allows the user to feed a computer algorithm an immense amount of data and have the computer analyze and make data-driven recommendations and decisions based on only the input data.
- If any corrections are identified, the algorithm can incorporate that information to improve its future decision making.

Categorizing machine learning algorithms...

- ... is tricky, and there are several approaches;
- they can be grouped into generative/discriminative, parametric/non-parametric, supervised/unsupervised, and so on.

What is supervised learning?

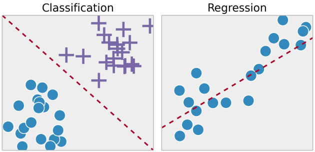
Supervised learning includes tasks for “labeled” data (i.e. you have a target variable).

- In practice, it’s often used as an advanced form of predictive modeling.
- Each observation must be labeled with a “correct answer.”
- Only then can you build a predictive model because you must tell the algorithm what’s “correct” while training it (hence, “supervising” it).
- Regression is the task for modeling continuous target variables.
- Classification is the task for modeling categorical (a.k.a. “class”) target variables.

Supervised vs unsupervised learning

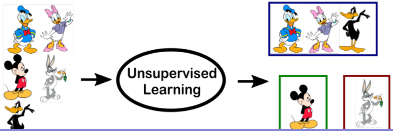
Supervised learning

- Prior knowledge of what output values for samples should be.



Unsupervised learning

- Here the most common tasks are clustering, representation learning, and density estimation - we wish to learn the inherent structure of our data without using explicitly-provided labels.



Machine Learning - Components

- Feature Extraction + Domain knowledge
- Feature Selection
- Choice of Algorithm (Regression or classification, regularization, decision trees, k-Means clustering, . . .)
- Training
- Choice of Metrics/Evaluation Criteria
- Testing

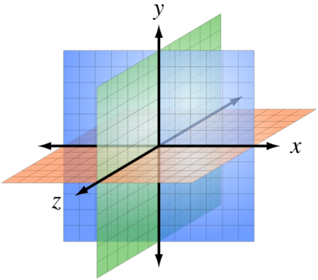
Feature selection in machine learning,. . .

- . . . is the process of selecting a subset of relevant features (variables, predictors) for use in model construction.

Four reasons for feature selection:

1.) simplification of models to make them easier to interpret by researchers/users,
2.) shorter training times,
3.) to avoid the curse of dimensionality,
4.) enhanced generalization by reducing overfitting (formally, reduction of variance)

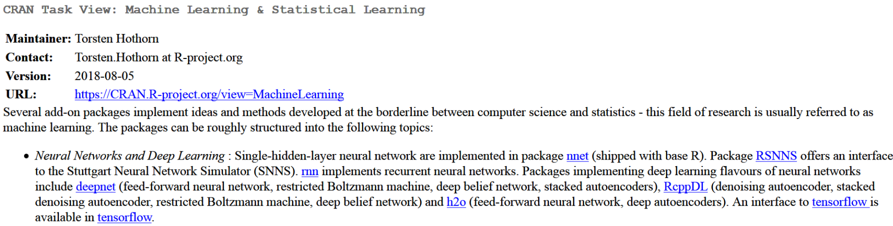
The Curse of Dimensionality



In machine learning, “dimensionality” simply refers to the number of features (i.e. input variables) in your dataset.

- When the number of features is very large relative to the number of observations, certain algorithms struggle to train effective models.
- This is called the “Curse of Dimensionality,” and it’s especially relevant for clustering algorithms that rely on distance calculations.

Task View Machine Learning



CRAN Task View: Machine Learning & Statistical Learning

Maintainer:	Torsten Hothorn
Contact:	Torsten.Hothorn at R-project.org
Version:	2018-08-05
URL:	https://CRAN.R-project.org/view=MachineLearning

Several add-on packages implement ideas and methods developed at the borderline between computer science and statistics - this field of research is usually referred to as machine learning. The packages can be roughly structured into the following topics:

- *Neural Networks and Deep Learning* : Single-hidden-layer neural network are implemented in package nnet (shipped with base R). Package RSNNS offers an interface to the Stuttgart Neural Network Simulator (SNNS). rnn implements recurrent neural networks. Packages implementing deep learning flavours of neural networks include deepnet (feed-forward neural network, restricted Boltzmann machine, deep belief network, stacked autoencoders), RcppDL (denoising autoencoder, stacked denoising autoencoder, restricted Boltzmann machine, deep belief network) and h2o (feed-forward neural network, deep autoencoders). An interface to tensorflow is available in tensorflow.

Install all packages of a task view

```r
install.packages("ctv")
ctv::install.views("MachineLearning")
```

Task: Find R-packages

Go to https://cran.r-project.org/ and search for packages that can be used:

- to reduce overfitting
- for random forests
- for gradient boosting
- for neural networks
- for clustering

Ensembling

Ensembles are machine learning methods for combining predictions from multiple separate models.

Bagging

attempts to reduce the chance overfitting complex models.

- It trains a large number of “strong” learners in parallel.
- A strong learner is a model that’s relatively unconstrained.
- Bagging then combines all the strong learners together in order to “smooth out” their predictions.

Boosting

attempts to improve the predictive flexibility of simple models.

- It trains a large number of “weak” learners in sequence.
- A weak learner is a constrained model (limit for max depth of tree).
- Each one in the sequence focuses on learning from the mistakes of the one before it.
- Boosting combines all the weak learners into a single strong learner.

Bagging and boosting

While bagging and boosting are both ensemble methods, they approach the problem from opposite directions.

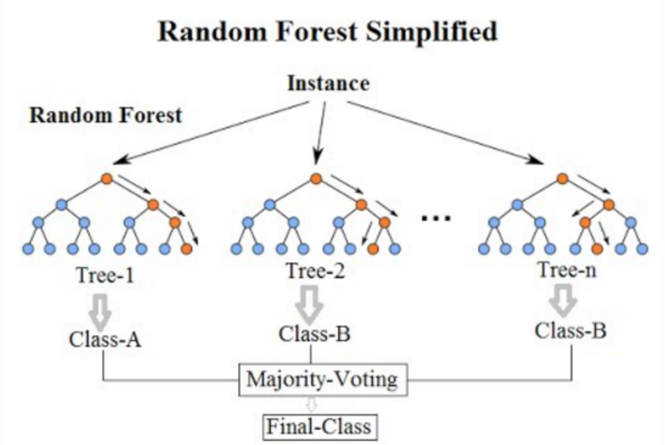
Bagging uses complex base models and tries to “smooth out” their predictions, while boosting uses simple base models and tries to “boost” their aggregate complexity.

Random Forest

Random forest aims to reduce the previously mentioned correlation issue by choosing only a subsample of the feature space at each split. Essentially, it aims to make the trees de-correlated and prune the trees by setting a stopping criteria for node splits, which I will cover in more detail later.

Random forest

- Ensemble learning method - multitude of decision trees
- Random forests correct for decision trees' habit of overfitting to their training set.



What are the advantages and disadvantages of decision trees?

Advantages: Decision trees are easy to interpret, nonparametric (which means they are robust to outliers), and there are relatively few parameters to tune.

Disadvantages: Decision trees are prone to be overfit.

- This can be addressed by ensemble methods like random forests or boosted trees.

Gradient boosting

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function.

The idea of gradient boosting originated in the observation by Leo Breiman that boosting can be interpreted as an optimization algorithm on a suitable cost function.

Breiman, L. (1997). "Arcing The Edge". Technical Report 486. Statistics Department, University of California, Berkeley.

Advantages of gradient boosting

- Often provides predictive accuracy that cannot be beat.
- Lots of flexibility - can optimize on different loss functions and provides several hyperparameter tuning options that make the function fit very flexible.
- No data pre-processing required - often works great with categorical and numerical values as is.
- Handles missing data - imputation not required.

Disadvantages of gradient boosting

- GBMs will continue improving to minimize all errors. This can overemphasize outliers and cause overfitting. Must use cross-validation to neutralize.
- Computationally expensive - GBMs often require many trees ($>$1000) which can be time and memory exhaustive.
- The high flexibility results in many parameters that interact and influence heavily the behavior of the approach (number of iterations, tree depth, regularization parameters, etc.). This requires a large grid search during tuning.
- Less interpretable although this is easily addressed with various tools (variable importance, partial dependence plots, LIME, etc.).

Two types of errors for tree methods

Bias related errors

- Adaptive boosting
- Gradient boosting

Variance related errors

- Bagging
- Random forest

Links

- Presentations on ‘Elements of Neural Networks & Deep Learning’
- Understanding the Magic of Neural Networks
- Neural Text Modelling with R package ruimtehol
- Feature Selection using Genetic Algorithms in R
- Lecture slides: Real-World Data Science (Fraud Detection, Customer Churn & Predictive Maintenance)
- Automated Dashboard for Credit Modelling with Decision trees and Random forests in R
- Looking Back at Google’s Research Efforts in 2018
- Selecting ‘special’ photos on your phone
- Open Source AI, ML & Data Science News
- Google‘s Machine Learning Crash Course
- A prelude to machine learning
- caret webinar by Max Kuhn - on youtube
- learn-math-for-data-science
- learn-statistics-for-data-science
- machine-learning-projects-for-beginners
- An Introduction to machine learning

Annex - Prediction vs. Causation in Regression Analysis

Paul Allison

> *There are two main uses of multiple regression: prediction and causal analysis. In a prediction study, the goal is to develop a formula for making predictions about the dependent variable, based on the observed values of the independent variables. . . .In a causal analysis, the independent variables are regarded as causes of the dependent variable. The aim of the study is to determine whether a particular independent variable really affects the dependent variable, and to estimate the magnitude of that effect, if any*