Introduction to Machine Learning

ML-Basics: Losses & Risk Minimization

HOW TO EVALUATE MODELS

OVERVIEW

No Free Lunch In machine learning, there’s something called the “No Free Lunch” theorem. In a nutshell, it states that no one algorithm works best for every problem, and it’s especially relevant for supervised learning (i.e. predictive modeling).
For example, you can’t say that neural networks are always better than decision trees or vice-versa. There are many factors at play, such as the size and structure of your dataset.
As a result, you should try many different algorithms for your problem, while using a hold-out “test set” of data to evaluate performance and select the winner. Hypothesisspace + Risk + Optimization

LINEAR MODEL FUNCTIONALITY

General information

- one of the most common algorithms

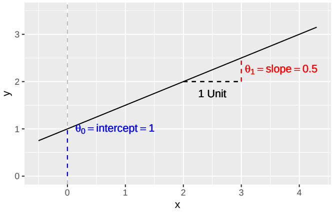
Aim

Aim Find the best line/straight hyperplane through data (LINEAR!)

- Predict continuos, numeric variables

Hypothesisspace

$$\mathcal{H} = \{\theta_0 + \boldsymbol{\theta}^T \mathbf{x} \mid (\theta_0, \boldsymbol{\theta}) \in \mathbb{R}^{p+1}\}$$



LINEAR MODEL FUNCTIONALITY

Risk

- Empirical Risk Minimization with the loss function - normally quadratic loss fucntion

Optimization

for L2-loss analytically; numerical optimization for others

Typical appication

LINEAR MODEL - ADVANTAGES AND DISADVANTAGES

Advantages

- simple implementation and simple to understand
- interpretability: gives information about mean influence of the features –> feature importance
- works good independent of dataset size
- fits linearly separable datasets very good
- cheap computational cost –> fast train and forecaste
- ground for many other ML algorithms
- fast training

Disadvantages

- strong assumptions: data is independent and normal-distributed(multicollinearity must be removed); simplification of real-world problems
- overfitting –> can be reduced by regularization
- sensitve to outliers and noisy data
- not suitable for non-linear data

Conclusion Impressive results on lineare separable datasets with easy

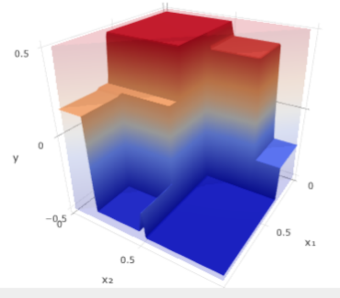
FUNCTIONALITY

General idea Starting from a root node, ***classification & regression trees (CART)*** perform repeated **binary splits** of the data according to feature values, thereby subsequently dividing the input space $\mathcal{X}$ into M **rectangular partitions**.

- $\rightarrow$ Pass observations along until each ends up in exactly one leaf node
- $\rightarrow$ In each step, find the optimal feature-threshold combination to split by
- $\rightarrow$ Assign response c_m to leaf node m

Hypothesis space

$$\mathcal{H} = \left\{ f(\mathbf{x}) : f(\mathbf{x}) = \sum_{m=1}^{M} c_m \mathbb{I}(\mathbf{x} \in Q_m) \right\}$$



Loss functions

Classification: ***Brier score, Bernoulli loss***
Regression: ***quadratic loss***

Optimization

Exhaustive search for optimal splitting criterion

NON-PARAMETRIC | WHITE-BOX | FEATURE SELECTION

RANDOM FOREST - FUNCTIONALITY

- xx
- xx

RANDOM FOREST - ADVANTAGES AND DISADVANTAGES

Advantages

- powerful
- accurate
- also good performance on non-linear problems
- fast execution
- flexible
- can model missing values

Disadvantages

- no interpretability
- can easily overfit
- number of trees must be chosen; small changes in training data changes model
- slow training
- not suitable for small samples
- occasionally too simple for complex problems

SVM - FUNCTIONALITY

Support Vector Machines (SVM)

- Support vector machines (SVM) use a mechanism called kernels, which essentially calculate distance between two observations. The SVM algorithm then finds a decision boundary that maximizes the distance between the closest members of separate classes.
- xx

SVM - ADVANTAGES AND DISADVANTAGES

Advantages

- SVMs can model non-linear boundaries
- robust against overfitting; especially in high-dimensional space
- computational

Disadvantages

- memory intensive
- not easy to tune –> important to choose the right kernel
- does not scale well to larger data sets

GRADIENT BOOSTING - FUNCTIONALITY

- xx
- xx

GRADIENT BOOSTING - ADVANTAGES AND DISADVANTAGES

Advantages

- interpretability
- computational

Disadvantages

- only linear relationship

NEURAL NET - FUNCTIONALITY

- Deep learning refers to multi-layer neural networks that can learn extremely complex patterns. They use "hidden layers" between inputs and outputs in order to model intermediary representations of the data that other algorithms cannot easily learn.
- state-of-the-art for computer vision and speech recognition

NEURAL NET - ADVANTAGES AND DISADVANTAGES

Advantages

- very accuarate
- can solve complex, non-linear or classification problems
- perform very well on unstructured data (image, audio and text data)
- can be easily updated (batch propagation)
- reduce the need for feature engineering

Disadvantages

- very slow to train and forecast
- requires large amount of data
- black-box; hard to interpret
- computationally expensive
- require much expertise for tuning
- tend to overfit

REGULARIZED LINEAR MODEL - FUNCTIONALITY

- xx
- xx

REGULARIZED LINEAR MODEL - ADVANTAGES AND DISADVANTAGES

Advantages

- interpretability
- computational

Disadvantages

- only linear relationship

KNN - FUNCTIONALITY

- Nearest neighbors algorithms are "instance-based," which means that that save each training observation. They then make predictions for new observations by searching for the most similar training observations and pooling their values.
- xx

KNN - ADVANTAGES AND DISADVANTAGES

Advantages

- simple adabtle to problem
- accuarate
- easy to understand
- few parameters to tune

Disadvantages

- memory intensive
- computationally costly –> all training data might be involved in the decision making
- slow performance
- wrong distance measure can lead to inaccurate results
- k must be selected