All of Machine Learning

A summary under eternal construction

Moritz Gück
github.com/MoritzGuck/All_of_ML-under_construction
Last changes: March 17, 2020

Abstract

This is a reference for machine learning approaches and methods. The topics range from basic statistics to complex machine learning models and explanation methods. For each method and model, I have provided the underlying formulas (objective functions, prediction functions, etc.) as well as code snippets from the respective python libraries. I made this reference to quickly look up things I have studied already. I published it to give data scientists a catalogue to find methods for their problem, refresh their knowledge and give references for further reading. If you find errors or unclear explanations in this text, please file an issue under:
`github.com/MoritzGuck/All_of_ML-under_construction`

Contents

1 Probability Theory

A probability is a measure of how frequent or likely an event will take place.

Probability Space The probability space is a triplet space containing a sample/outcome space Ω (containing all possible atomic events), a collection of events S (containing a subset of Ω to which we want to assign probabilities) and the mapping P between Ω and S.

Axioms of Probability The mapping P must fulfill the axioms of probability:

1. $P(a) \geq 0$
2. $P(\Omega) = 1$
3. $a, b \in S$ and $a \cap b = \{\} \Rightarrow P(a \cup b) = P(a) + P(b)$

Random Variable A random variable is a function that maps points from the sample space Ω to some range (e.g. Real numbers or booleans). They are characterized by their distribution function. E.g. for a dice roll:

$$X(\omega) = \begin{cases} 0, & \text{if } \omega = heads \\ 1, & \text{if } \omega = tails. \end{cases}$$

Proposition A Proposition is a conclusion of a statistical inference that can be true or false (e.g. a classification of a datapoint). More formally: A disjunction of events where the logic model holds. An event can be written as a **propositional logic model**:
$A = true, B = false \Rightarrow a \wedge \neg b$. Propositions can be continuous, discrete or boolean.

1.1 Probability distributions

Probability distributions assign probabilities to to all possible points in Ω (e.g. $P(Weather) = \langle 0.3, 0.4, 0.2, 0.1 \rangle$, representing Rain, sunshine, clouds and snow). Joint probability distributions give you a probability for each atomic event of the random variables (e.g. $P(weather, accident)$ gives you a 2×4 matrix.)

Cumulative Distribution Function (CDF) The CDF is defined as $F_X(x) = P(X \leq x)$ (See figure 2).

Probability Density Function (PDF) For continuous functions the PDF is defined by

$$p(x) = \frac{d}{dx} p(X \leq x). \tag{1}$$

The probability of x being in a finite interval is

$$P(a < X \leq b) = \int_a^b p(x) dx \tag{2}$$

A PDF is shown in figure

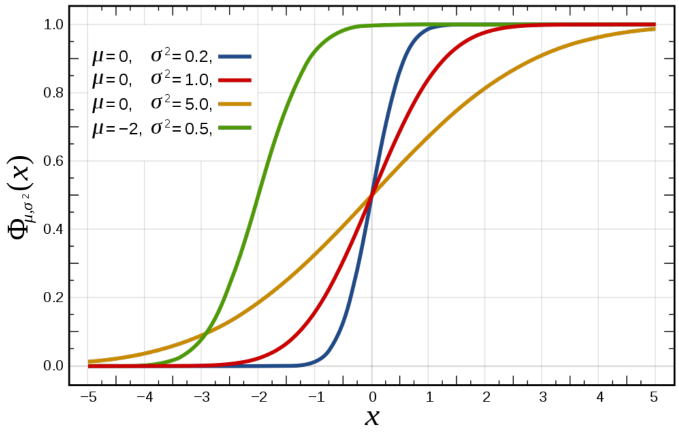

Fig. 1: Cumulative distribution function of a normal distribution for different mean (μ) and variance (σ). *Figure from user Inductiveload on wikimedia.org.*

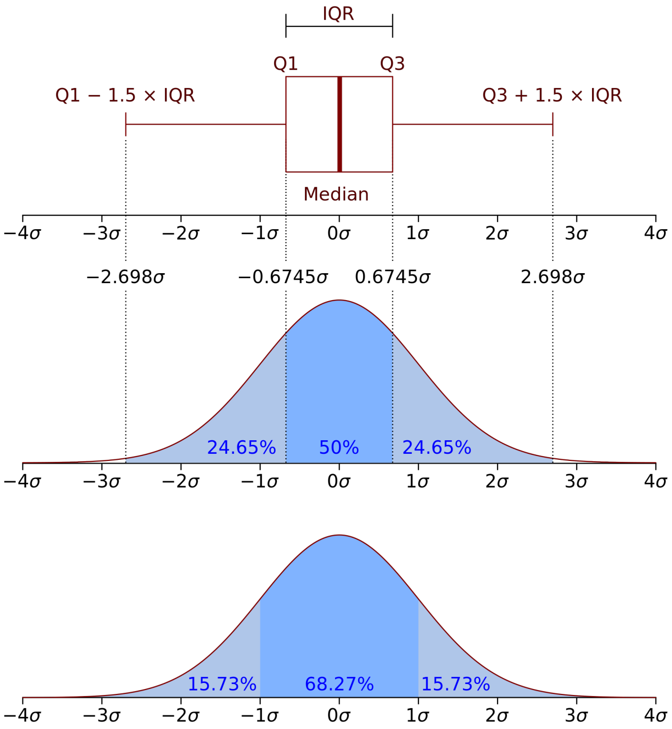

Fig. 2: Probability density function of a normal distribution with variance (σ). In red a range from a Box-plot is shown with quartiles (Q1, Q3) and interquartile range (IQR). For the cutoffs (borders to darker blue regions) the IQR (on top) and σ are chosen. Another common cutoff is the confidence interval with light blue regions having a probability mass of $2 * \alpha/2$. *Figure from user Jhguch on wikimedia.org.*

Uniform distribution The uniform distribution has the same probability throughout a specific interval and is defined as

$$\mathsf{Unif}(a,b) = \frac{1}{b-a}\mathbb{I}(a < x \leq b) = \begin{cases} \frac{1}{b-a}, & \text{if } x \in [a,b] \\ 0, & \text{else} \end{cases}$$

2 Classification Methods

Properties of Distributions Classification is the assignment of objects (data points) to categories (classes). It requires a data set (i.e. training set) of points with known class labels. If the class labels are not known you can instead group the data using clustering algorithms (chapter **??**).

2.1 Linear Classifiers

Linear classifiers use linear decision boundaries to classify points to a respective class.

2.2 Support Vector Classifier (SVC)

SVCs use hyperplanes to separate data points according to their class label with a maximum margin (M) between the separating hyperplane $(x^T\beta + \beta_0 = 0)$ and the points. If points cannot be perfectly separated by the decision boundary, a soft margin SVM is used with a slack variable ξ that punishes points in the margin or on the wrong side of the hyperplane. The optimization problem is given by [**?**] :

$$\begin{aligned} \max_{\beta,\beta_0,\beta=1} M, \\ \text{subject to } y_i(x_i^T\beta + \beta_0) \geq 1 - \xi_i, \quad \forall i, \\ \xi_i \geq 0, \quad \sum \xi_i \leq constant, \quad i = 1, ..., N, \end{aligned} \tag{3}$$

where β are the coefficients and x are the N data points. The support vectors are the points that determine the orientation of the hyperplane (i.e. the closest points). The classification function is given by:

$$G(x) = \text{sign}[x^T\beta + \beta_0] \tag{4}$$

Index