Chapter ML:I

I. Introduction

- Examples of Learning Tasks
- Specification of Learning Tasks
- Elements of Machine Learning

- Notation Overview
- Classification Approaches Overview

Notation Overview

Data, Sets, and Distributions

Symbol	Semantics
$x,\ x_i,\ x_1, \ldots, x_p$	Feature
$\mathbf{x} = (x_1, \ldots, x_p)^T \in \mathbf{R}^p$	Feature vector
$\mathbf{x} = (1, x_1, \ldots, x_p)^T \in \mathbf{R}^{p+1}$, i.e., $x_0 = 1$	Extended feature vector
$\mathbf{X}$	Feature space, Cartesian product of the domains of the p dimensions of a feature vector $\mathbf{x}$.
$X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$	Multiset of feature vectors
X	Random variable (randomness regarding feature x of an object o)
X	Multivariate random variable, random vector (randomness regarding feature vector $\mathbf{x}$ of an object o)

Notation Overview

Indexing

Running	Sequence	Semantics of maximum
$\square_s$	$\in \{\square_1, \ldots, \square_d\}$	Number of layers in a multilayer perceptron
$\square_i$	$\in \{\square_1, \ldots, \square_k\}$	Number of classes Number of folds during cross validation
$\square_l$	$\in \{\square_1, \ldots, \square_m\}$	Number of elements in a domain of a feature Number of hyperparameter values during model selection
$\square_i$	$\in \{\square_1, \ldots, \square_n\}$	Number of elements in a data set D
$\square_j$	$\in \{\square_1, \ldots, \square_p\}$	Dimension of a feature space or a feature vector

Notation Overview

Functions

Function definition	Function name	Occurrence
$I_{\neq}(a,b) = \begin{cases} 0 & a = b \\ 1 & a \neq b \end{cases}$	Indicator function	Part II: Machine Learning Basics Part III: Linear Models
$f(x) = \ldots$	function	Part :

Notation Overview

Algorithms

Signature	Algorithm name	Occurrence	
LMS(D, η)	Least Mean Squares	Part I:	Introduction Examples of Learning Tasks
ALG$(\ldots)$	algorithm	Part :	$\ldots$

Classification Approaches Overview

Taxonomy	Model function	Classification rule	Optimization principle	Optimization objective (loss/cost function [+ regularization])	Optimization approach (algorithm)
			Search in hypothesis space		
Classification approaches / Discriminative approaches / Linear decision boundary (in inner product space) / Linear decision boundary in input space	Perceptron: $y(\mathbf{x}) =$ *heaviside*$(\mathbf{w}^T\mathbf{x})$	$\mathbf{w}^T\mathbf{x} \begin{cases} \geq 0 \\ < 0 \end{cases}$ $\mathbf{w}^T=(w_0,\ldots,w_p)$ $x_0=1$	Exploit misclassified examples individually: Hebbian learning	No misclassified example	Perceptron training algorithm
	Linear function: $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$		Linear regression + Regularization	Squared loss (residual sum of squares, RSS) + L_1 or L_2 norm on $\mathbf{w}\vert_{1,\ldots,p}$	Gradient descent: – batch – incremental – stochastic; Newton-Raphson, BFGS
	Logistic function: $y(\mathbf{x}) = \frac{1}{1+e^{-\mathbf{w}^T\mathbf{x}}}$		Logistic regression + Regularization	Logistic loss (derived via ML) + L_1 or L_2 norm on $\mathbf{w}\vert_{1,\ldots,p}$	
	SVM w/o kernel (aka linear kernel)	$\mathbf{w}^T\mathbf{x}-b \begin{cases} \geq 1 \\ \leq -1 \end{cases}$ $\mathbf{w}^T=(w_1,\ldots,w_p)$	Empirical risk minimization	Regularized hinge loss	Quadratic prog., sub-grad. descent
Nonlinear in input / linear in feature space	$y(\mathbf{x}) = \text{sign}(\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}))$	$\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}) \begin{cases} \geq 0 \\ < 0 \end{cases}$ $\mathbf{w}^T = (w_0,\ldots,w_{\vert\mathbf{w}\vert})$ $\phi_0(\mathbf{x})=1$	Linear regression (nonlinear in predictors) + Regularization	Squared loss + L_1 or L_2 norm on $\mathbf{w}\vert_{1,\ldots,\vert\mathbf{w}\vert}$	Gradient descent: – batch – incremental – stochastic; Newton-Raphson, BFGS
	$y(\mathbf{x}) = \frac{1}{1+e^{-\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x})}}$		Log. regression (nonlinear in predictors) + Regularization	Logistic loss + L_1 or L_2 norm on $\mathbf{w}\vert_{1,\ldots,\vert\mathbf{w}\vert}$	
	SVM with nonlinear kernel	$\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x})-b \begin{cases} \geq 1 \\ \leq -1 \end{cases}$ $\mathbf{w}=\sum_{i=1}^n \alpha_i c \phi(\mathbf{x}_i)$	Empirical risk minimization	Regularized hinge loss	Quadratic prog., sub-grad. descent
Unrestricted decision boundary / Poly-thetic	Multilayer percep.: $\mathbf{y}(\mathbf{x}) = \boldsymbol{\sigma}(W^o\binom{1}{\boldsymbol{\sigma}(W^h\mathbf{x})})$	$\text{argmax}_{c\in C}$ $\{\, y_c(\mathbf{x}) \,\}$	Regression	Squared loss (residual sum of squares, RSS)	Backpropagation algorithm
Monothetic feature analysis	Nominal feat.: $\bigwedge_i x_i=v_i$, $i=1,\ldots,p$	Test if $\mathbf{x}$ is a model for α (= fulfills α). α is a formula in DNF.	Maximize version space	No misclassified example	Candidate elimination algorithm
	Nominal feat.: $\bigvee_i \bigwedge_j x_{ij}=v_{ij}$, $i=1,\ldots,\vert\text{leafs}\vert$, $j=1,\ldots,\text{depth}(l_i)$		Decision tree: (greedy) feature-wise splitting of example set + Regularization	0/1 Loss (= number of misclassified examples) + Tree height, external path length	Algorithms: ID3, C4.5, C5.0, CART
	Arbitrary features: DNF ($\vee_i \wedge_j$) on domain predicates				(exhaustive) search in space of domain splittings
Generative approaches / Statistical approaches	Bayes rule for combined conditional events	$\text{argmax}_{c\in C}$ { Naive Bayes probabilites }	Maximum a-posteriori hypothesis		
	$X \sim N(\mu,\sigma^2)$ (or other family)	$\text{argmax}_{c\in C}$ $\{P(\mathbf{x}\vert\mu_c,\sigma_c)\}$		Goodness of fit, e.g. according to chi-squared, Kolmogorov-Smirnov	