

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, \dots, x_p	Feature
$\mathbf{x} = (x_1, \dots, x_p)^T \in \mathbf{R}^p$	Feature vector
$\mathbf{x} = (1, x_1, \dots, x_p)^T \in \mathbf{R}^{p+1}$, 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} .
$X = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$	Multiset of feature vectors
X	Random variable (randomness regarding feature x of an object o)
\mathbf{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	$\in \{\square_1, \dots, \square_d\}$	Number of layers in a multilayer perceptron
\square_i	$\in \{\square_1, \dots, \square_k\}$	Number of classes Number of folds during cross validation
\square_l	$\in \{\square_1, \dots, \square_m\}$	Number of elements in a domain of a feature Number of hyperparameter values during model selection
\square_i	$\in \{\square_1, \dots, \square_n\}$	Number of elements in a data set D
\square_j	$\in \{\square_1, \dots, \square_p\}$	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

$$f(x) = \dots$$

function

Part :

Notation Overview

Algorithms

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

Classification Approaches Overview

Search in hypothesis space

Taxonomy		Model function	Classification rule	Optimization principle	Optimization objective (loss/cost function [+ regularization])	Optimization approach (algorithm)												
Classification approaches	Discriminative approaches	Linear decision boundary (in inner product space)	Perceptron: $y(\mathbf{x}) = \text{heaviside}(\mathbf{w}^T \mathbf{x})$	Exploit misclassified examples individually: Hebbian learning	No misclassified example	Perceptron training algorithm												
			Linear function: $y(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$				$\mathbf{w}^T \mathbf{x} \begin{cases} \geq 0 \\ < 0 \end{cases}$ $\mathbf{w}^T = (w_0, \dots, w_p)$ $x_0 = 1$	<table border="1"> <tr> <td>Linear regression</td> <td rowspan="2">+ Regularization</td> </tr> <tr> <td>Logistic regression</td> </tr> </table>	Linear regression	+ Regularization	Logistic regression	<table border="1"> <tr> <td>Squared loss (residual sum of squares, RSS)</td> <td rowspan="2">+ L_1 or L_2 norm on $\mathbf{w}_{1, \dots, p}$</td> </tr> <tr> <td>Logistic loss (derived via ML)</td> </tr> </table>	Squared loss (residual sum of squares, RSS)	+ L_1 or L_2 norm on $\mathbf{w}_{1, \dots, p}$	Logistic loss (derived via ML)	<ul style="list-style-type: none"> Gradient descent: <ul style="list-style-type: none"> - batch - incremental - stochastic Newton-Raphson, BFGS 		
			Linear regression						+ Regularization									
			Logistic regression															
			Squared loss (residual sum of squares, RSS)						+ L_1 or L_2 norm on $\mathbf{w}_{1, \dots, p}$									
	Logistic loss (derived via ML)																	
	Logistic function: $y(\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x}}}$	Empirical risk minimization	Regularized hinge loss	<ul style="list-style-type: none"> Quadratic prog., sub-grad. descent 														
	SVM w/o kernel (aka linear kernel)				$\mathbf{w}^T \mathbf{x} - b \begin{cases} \geq 1 \\ \leq -1 \end{cases}$ $\mathbf{w}^T = (w_1, \dots, w_p)$	<table border="1"> <tr> <td>Linear regression (nonlinear in predictors)</td> <td rowspan="2">+ Regularization</td> </tr> <tr> <td>Log. regression (nonlinear in predictors)</td> </tr> </table>	Linear regression (nonlinear in predictors)	+ Regularization	Log. regression (nonlinear in predictors)	<table border="1"> <tr> <td>Squared loss</td> <td rowspan="2">+ L_1 or L_2 norm on $\mathbf{w}_{1, \dots, w }$</td> </tr> <tr> <td>Logistic loss</td> </tr> </table>	Squared loss	+ L_1 or L_2 norm on $\mathbf{w}_{1, \dots, w }$	Logistic loss	<ul style="list-style-type: none"> Gradient descent: <ul style="list-style-type: none"> - batch - incremental - stochastic Newton-Raphson, BFGS 				
	Linear regression (nonlinear in predictors)						+ Regularization											
	Log. regression (nonlinear in predictors)																	
Squared loss	+ L_1 or L_2 norm on $\mathbf{w}_{1, \dots, w }$																	
Logistic loss																		
Nonlinear in input / linear in feature space	Empirical risk minimization	Regularized hinge loss	<ul style="list-style-type: none"> Quadratic prog., sub-grad. descent 															
$y(\mathbf{x}) = \text{sign}(\mathbf{w}^T \phi(\mathbf{x}))$				$\mathbf{w}^T \phi(\mathbf{x}) \begin{cases} \geq 0 \\ < 0 \end{cases}$ $\mathbf{w}^T = (w_0, \dots, w_{ w })$ $\phi_0(\mathbf{x}) = 1$	Regression	Squared loss (residual sum of squares, RSS)	Backpropagation algorithm											
$y(\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T \phi(\mathbf{x})}}$								$\mathbf{w}^T \phi(\mathbf{x}) - b \begin{cases} \geq 1 \\ \leq -1 \end{cases}$ $\mathbf{w} = \sum_{i=1}^n \alpha_i c \phi(\mathbf{x}_i)$	Maximize version space	No misclassified example	Candidate elimination algorithm							
SVM with nonlinear kernel												$\text{argmax}_{c \in C} \{ y_c(\mathbf{x}) \}$	Decision tree: (greedy) feature-wise splitting of example set	+ Regularization	<table border="1"> <tr> <td>0/1 Loss (= number of misclassified examples)</td> <td rowspan="2">+ Tree height, external path length</td> </tr> </table>	0/1 Loss (= number of misclassified examples)	+ Tree height, external path length	<ul style="list-style-type: none"> Algorithms: ID3, C4.5, C5.0, CART (exhaustive) search in space of domain splittings
0/1 Loss (= number of misclassified examples)																+ Tree height, external path length		
Polythetic	<ul style="list-style-type: none"> Test if \mathbf{x} is a model for α (= fulfills α). α is a formula in DNF. 	Maximum a-posteriori hypothesis	Goodness of fit, e.g. according to chi-squared, Kolmogorov-Smirnov															
Unrestricted decision boundary				$\text{argmax}_{c \in C} \{ \text{Naive Bayes probabilities} \}$	$\text{argmax}_{e \in C} \{ P(\mathbf{x} \mu_e, \sigma_e) \}$													
						Monothetic feature analysis	Bayes rule for combined conditional events											
Nominal feat. $\bigwedge_{i=1, \dots, p} x_i = v_i$								<ul style="list-style-type: none"> Bayes rule for combined conditional events $X \sim N(\mu, \sigma^2)$ (or other family) 										
Arbitrary features: DNF ($\bigvee_i \bigwedge_j$) on domain predicates																		
Statistical approaches	<ul style="list-style-type: none"> Bayes rule for combined conditional events $X \sim N(\mu, \sigma^2)$ (or other family) 																	
Generative approaches		<ul style="list-style-type: none"> Bayes rule for combined conditional events $X \sim N(\mu, \sigma^2)$ (or other family) 																
			<ul style="list-style-type: none"> Bayes rule for combined conditional events $X \sim N(\mu, \sigma^2)$ (or other family) 															
				<ul style="list-style-type: none"> Bayes rule for combined conditional events $X \sim N(\mu, \sigma^2)$ (or other family) 														
					<ul style="list-style-type: none"> Bayes rule for combined conditional events $X \sim N(\mu, \sigma^2)$ (or other family) 													
	<ul style="list-style-type: none"> Bayes rule for combined conditional events $X \sim N(\mu, \sigma^2)$ (or other family) 																	