Chapter ML: (continued)

I. Introduction

- □ Examples of Learning Tasks
- □ Specification of Learning Tasks
- □ Elements of Machine Learning
- Notation Overview
- Classification Approaches Overview

ML:I-71 Introduction ©STEIN 2023

Data, Sets, and Distributions

| Symbol | Semantics |
|---|---|
| x, x_i, x_1, \ldots, x_p | Feature |
| $\mathbf{x} = (x_1, \dots, x_p)^T \in \mathbf{R}^p$ | Feature vector |
| $\mathbf{x} = \left(1, x_1, \dots, x_p\right)^T \in \mathbf{R}^{p+1}$, i.e., $x_0 = 1$ | Extended feature vector |
| X | Feature space, Cartesian product of the domains of the $\it p$ dimensions of a feature vector $\bf x$. |
| $X = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ | Multiset of feature vectors |
| Χ | 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) |

ML:I-72 Introduction © STEIN 2023

Indexing

| Running | | Sequence | Semantics of maximum |
|-------------|-------|----------------------------|---|
| \square_s | \in | $\{\Box_1,\dots,\Box_d\}$ | Number of layers in a multilayer perceptron |
| \Box_i | \in | $\{\Box_1,\ldots,\Box_k\}$ | Number of classes Number of folds during cross validation |
| \Box_l | \in | $\{\Box_1,\ldots,\Box_m\}$ | Number of elements in a domain of a feature Number of hyperparameter values during model selection |
| \Box_i | \in | $\{\Box_1,\ldots,\Box_n\}$ | Number of elements in a data set D |
| \Box_j | \in | $\{\Box_1,\ldots,\Box_p\}$ | Dimension of a feature space or a feature vector |

ML:I-73 Introduction © STEIN 2023

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 : |

ML:I-74 Introduction © STEIN 2023

Algorithms

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

ML:I-75 Introduction © STEIN 2023

Classification Approaches Overview

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

ML:I-76 Introduction © STEIN 2023