A Dictionary of Machine Learning

Comments are warmly welcome at `alex.jung@aalto.fi` and will be acknowledged properly.

Dipl.-Ing. Dr.techn. Alexander Jung[*†]

February 4, 2023

Abstract

*
†

Lists of Symbols

$\mathcal{G} = \big(\mathcal{V}, \mathcal{E}\big)$	Empirical graph whose nodes $i \in \mathcal{V}$ carry local datasets and local models.
$i \in \mathcal{V}$	A node in the empirical graph that represents a local dataset and local model. It might also be useful to think of node i as a small computer that can collect data and execute computations for model training.
$\mathcal{D}^{(i)}$	The local dataset $\mathcal{D}^{(i)}$ at node $i \in \mathcal{V}$.
m_i	The number of data points (sample size) contained in the local dataset $\mathcal{D}^{(i)}$ at node $i \in \mathcal{V}$.
$\mathbf{x}$	The feature vector $\mathbf{x} = \big(x_1, \ldots, x_d\big)^T$ of a generic data point.
$\mathcal{X}$	The feature space $\mathcal{X}$ is the set of all possible values that the features $\mathbf{x}$ of a data point can take on.
$\mathbf{x}^{(i,r)}$	The feature vector of the r-th data point in the local dataset $\mathcal{D}^{(i)}$
$\mathbf{v} \in \mathbb{R}^d$	Some numeric vector (one-dimensional array) of length d.

$\|\mathbf{w}\|_2$	The Euclidean norm $\|\mathbf{w}\|_2 := \sqrt{\sum_{j=1}^{d} w_j^2}$ of the d-dimensional vector $\mathbf{w}$.
h	A hypothesis map $h : \mathcal{X} \to \mathcal{Y}$ that reads in the features $\mathbf{x}$ of a data point and delivers a prediction $h(\mathbf{x})$ for its label y.
$\mathcal{Y}^{\mathcal{X}}$	Given two sets $\mathcal{X}$ and $\mathcal{Y}$, we denote by $\mathcal{Y}^{\mathcal{X}}$ the set of all maps $h : \mathcal{X} \to \mathcal{Y}$.
$\mathcal{H}$	A hypothesis space which consists of hypothesis maps h.

Glossary

0/1 loss The 0/1 loss $L\big(\big(\mathbf{x}, y\big), h\big)$ measures the quality of a classifier $h(\mathbf{x})$ that delivers a prediction $\hat{y}$ for a label y of a data point. It is equal to 0 if the prediction is correct, i.e., $L\big(\big(\mathbf{x}, y\big), h\big) = 0$ when $\hat{y} = y$. It is equal to 1 if the prediction is wrong, $L\big(\big(\mathbf{x}, y\big), h\big) = 0$ when $\hat{y} \neq y$.

k-fold cross-validation (k-fold CV) k-fold cross-validation is a method for learning and validating a hypothesis using a given dataset. This method first divides the dataset evenly into k subsets or "folds" and then executes k repetitions of training and validation. Each repetition uses a different fold as the validation set and the remaining $k-1$ folds as a training set. The final output is the average of the validation errors obtained from the k repetitions.

k-means The k-means algorithm is a hard clustering method which assigns each data points to pecisely one out of k different clusters. The method iteratively updates this assignment in order to minimize the average distance between data points in their nearest cluster mean (centre).

accuracy Consider data points characterized by features $\mathbf{x} \in \mathbf{x}$ and a categorical label y which takes on values from a finite label space $\mathcal{Y}$. The accuracy of a hypothesis $h : \mathcal{X} \to \mathcal{Y}$, when applied to the data points in a dataset $\mathcal{D} = \big\{\big(\mathbf{x}^{(1)}, y^{(1)}\big), \ldots, \big(\mathbf{x}^{(m)}, y^{(m)}\big)\big\}$ is then defined as $1 - (1/m) \sum_{r=1}^{m} L\big(\big(\mathbf{x}^{(r)}, y^{(r)}\big), h\big)$ using the 0/1 loss.

activation function Each artificial neuron within an artificial neural network (ANN) consists of an activation function that maps the inputs of

the neuron to a single output value. In general, an activation function is a non-linear map of the weighted sum of neuron inputs (this weighted sum is the activation of the neuron).

Application Programming Interface (API) An Application Programming Interface (API) is a particular set of rules and specifications that a software program can follow to access and make use of the services and resources provided by another particular software program that implements that API

artificial intelligence Artificial intelligence aims at developing systems that behave rational in the sense of maximizing a long-term reward.

artificial neural network An artificial neural network is a graphical (signal-flow) representation of a map from features of a data point at its input to a predicted label at its output.

backdoor A backdoor (atttack) is obtained by data poisoning such that a trained local model predicts well most data points but anomalously for specific feature values (which unlock the backdoor).

bagging bagging (or "bootstrap aggregation") is a generic technique to improve (the robustness of) a given ML method. The idea is to use the bootstrap to generate perturbed copy of a given training set and then apply the original ML method to learn a separate hypothesis for each perturbed copy of the training set. The resulting set of hypotheses is then used to predict the label of a data point by combining or aggregating the individual predictions of each individual hypothesis.

For hypotheses that deliver numeric label values (regression methods) this aggregation could be implemented by computing the average of individual predictions.

baseline A reference value or benchmark for the average loss incurred by a hypothesis when applied to the data points generated in a specific ML application. Such a reference value might be obtained from human performance (e.g., error rate of dermatologists diagnosing cancer from visual inspection of skin areas) or other ML methods ("competitors")

Bayes estimator A hypothesis h whose Bayes risk is minimal [1].

Bayes risk We use the term Bayes risk as a synonym for the risk or expected loss of a hypothesis. Some authors reserve the term Bayes risk for the risk of a hypothesis that achieves minimum risk, such a hypothesis being referred to as a Bayes estimator [1].

bias Consider some unknown quantity $\bar{w}$, e.g., the true weight in a linear model $y = \bar{w}x + e$ relating feature and label of a data point. We might use an ML method (e.g., based on empirical risk minimization (ERM)) to compute an estimate $\hat{w}$ for the $\bar{w}$ based on a set of data points that are realizations of RVs. The (squared) bias incurred by the estimate $\hat{w}$ is typically defined as $B^2 := \big(\mathbb{E}\{\hat{w}\} - \bar{w}\big)^2$. We extend this definition to vector-valued quantities using the squared Euclidean norm $B^2 := \big\|\mathbb{E}\{\widehat{\mathbf{w}}\} - \overline{\mathbf{w}}\big\|_2^2$.

bootstrap Consider a probabilistic model that interprets a given set of data points $\mathcal{D} = \big\{\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(m)}\big\}$ as realizations of independent and identically

distributed (i.i.d.) RVs with a common probability distribution $p(\mathbf{z})$. The bootstrap uses the histogram of $\mathcal{D}$ as the underlying proability distribution $p(\mathbf{z})$.

classification Classification is the task of determining a discrete-valued label y of a data point based solely on its features $\mathbf{x}$. The label y belongs to a finite set, such as $y \in \{-1, 1\}$, or $y \in \{1, \ldots, 19\}$ and represents a category to which the corresponding data point belongs to.

classifier A classifier is a hypothesis (map) $h(\mathbf{x})$ that is used to predict a discrete-valued label. Strictly speaking, a classifier is a hypothesis $h(\mathbf{x})$ that can take only a finite number of different values. However, we are sometimes sloppy and use the term classifier also for a hypothesis that delivers a real number which is quantized, i.e., compared against a threshold, to obtain the predicted label value. For example, in a binary classification problem with label values $y \in \{-1, 1\}$, we refer to a hypothesis $h(\mathbf{x})$ as classifier if it is used to compute a predicted label according to

$$\hat{y} = 1 \text{ for } h(\mathbf{x}) \geq 0, \text{ and } \hat{y} = -1 \text{ otherwise.} \tag{1}$$

cluster A cluster within an empirical graph is a subset of nodes that carry local datasets with (nearly) identical statistical distributions. Clustered federated learning (FL) aims at identying the clusters and use the pooled local darasets to train a separate model for each cluster.

cluster A cluster is a subset of data points that are more similar to each other than to the data points outside the cluster. The notion and

measure of similarity between data points is a design choice. If data points are characterized by numeric Euclidean feature vectors it might be reasonable to define similarity between two data points using the (inverse of th) Euclidean distance between the corresponding feature vectors

clustering Clustering means to decompose a given set of data points into few subsets, which are referred to as clusters, that consist of similar data points. Different clustering methods use different measures for the similarity between data points and different representation of clusters. The clustering method k-means uses the average feature vector ("cluster means") of a cluster as its representative. A popular soft clustering method based on Gaussian mixture model (GMM) represents a cluster by a Gaussian (multivariate normal) probability distribution.

clustering assumption The clustering assumption postulates that data points form a (small) number of groups or clusters, within which the data points behave similar [2].

computational aspects By computational aspects of a ML method, we refer to resources required to implement the method. For example, if the ML method is based on gradient descent (GD) methods then its computation aspects are (i) how much computation is needed to implement a single GD step and (ii) how many GD steps are needed to learn (the parameters) of a useful hypothesis.

condition number The condition number $\kappa(\mathbf{Q}) \geq 1$ of a positive semi-definite (psd) matrix $\mathbf{Q}$ is the ratio $\lambda_{\max}/\lambda_{\min}$ between the largest $\lambda_{\max}$

and the smallest $\lambda_{\min}$ eigenvalue of $\mathbf{Q}$. One example for such a matrix $\mathbf{Q}$ is obtained from the feature matrix $\mathbf{X}$ of a dataset, $\mathbf{Q} = \mathbf{X}\mathbf{X}^T$. From a computational perspective, we prefer $\mathbf{Q}$ to have a condition number close to 1.

confusion matrix Consider data points characterized by features $\mathbf{x}$ and label y having value $c \in \{1, \ldots, k\}$. The confusion matrix is $k \times k$ matrix with rows representing different values c of the true label of a data point. The columns of a confusion matrix correspond to different values c' delivered by a hypothesis $h(\mathbf{x})$. The (c, c')-th entry of the confusion matrix is the fraction of data points with label $y = c$ and predicted label $\hat{y} = c'$.

convex A set $\mathcal{C}$ in $\mathbb{R}^d$ is convex if it contains the line segment between any two points of that set. A function is called convex if its epigraph is a convex set [3].

data A (indexed) set of data points.

data augmentation Data augmentation methods add synthetic data points to an existing set of data points. These synthetic data points might be obtained by perturbations (adding noise) or transformations (rotations of images) of the original data points.

data point A data point is any object that conveys information [4]. Data points might be students, radio signals, trees, forests, images, RVs, real numbers or proteins. We characterize data points using two types of properties. One type of property is referred to as a feature. Features

are properties of a data point that can be measured or computed in an automated fashion. Another type of property is referred to as a label. The label of a data point represents a higher-level facts or quantities of interest. In contrast to features, determining the label of a data point typically requires human experts (domain experts). Roughly speaking, ML aims at predicting the label of a data point based solely on its features.

data poisoning FL methods allow to leverage the information contained in local datasets generated by other parties to improve the training of a tailored model. Depending on how much we trust the other parties, FL can be compromised by data poisoning. Data poisoning refers to the intentional manipulation (or fabrication) of local datasets to steer the training of a specific local model [5, 6].

dataset With a slight abuse of notation we use the terms "dataset" or "set of data points" to refer to an indexed list of data points $\mathbf{z}^{(1)}, \ldots,$. Thus, there is a first data point $\mathbf{z}^{(1)}$, a second data point $\mathbf{z}^{(2)}$ and so on. Strictly speaking a dataset is a list and not a set [7]. By using indexed lists of data points we avoid some of the challenges arising in concept of an abstract set.

decision region Consider a hypothesis map h that reads in a feature vector $\mathbf{x} \in \mathbb{R}^d$ and delivers a value from a finite set $\mathcal{Y}$. The decision boundary induced by h is the set of vectors $\mathbf{x} \in \mathbb{R}^d$ that lie between differentdecision regions. More precisely, a vector $\mathbf{x}$ belongs to the decision boundary if and only if each neighbourhood $\{\mathbf{x}' : \|\mathbf{x} - \mathbf{x}'\| \leq \varepsilon\}$,

for any $\varepsilon > 0$, contains at least two vectors with different function values.

decision region Consider a hypothesis map h that can only take values from a finite set $\mathcal{Y}$. We refer to the set of features $\mathbf{x} \in \mathcal{X}$ that result in the same output $h(\mathbf{x}) = a$ as a decision region of the hypothesis h.

decision tree A decision tree is a flow-chart like representation of a hypothesis map h. More formally, a decision tree is a directed graph which reads in the feature vector $\mathbf{x}$ of a data point at its root node. The root node then forwards the data point to one of its children nodes based on some elementary test on the features $\mathbf{x}$. If the receiving children node is not a leaf node, i.e., it has itself children nodes, it represents another test. Based on the test result, the data point is further pushed to one of its neighbours. This testing and forwarding of the data point is repeated until the data point ends up in a leaf node (having no children nodes). The leaf nodes represent sets (decision regions) constituted by feature vectors $\mathbf{x}$ that are mapped to the same function value $h(\mathbf{x})$.

deep net We refer to an ANN with a (relatively) large number of hidden layers as a deep ANN or "deep net". Deep nets are used to represent the hypothesis spaces of deep learning methods [8].

degree of belonging A number that indicates the extend by which a data point belongs to a cluster. The degree of belonging can be interpreted as a soft cluster assignment. Soft clustering methods typically represent the degree of belonging by a real number in the interval $[0, 1]$. The boundary values 0 and 1 correspond to hard cluster assignments.

denial-of-service attack A denial-of-service attack uses data poisoning to steer the training of a local (client) model such that it performs poorly for typical data points

density-based spatial clustering of applications with noise (DBSCAN) A clustering algorithm for data points that are characterized by numeric feature vectors. Similar to k-means and soft clustering via GMM also DBSCAN uses the Euclidean distances between feature vectors to determine the clusters. However, in contrast to these other clustering methods, DBSCAN uses a different notion of similarity between data points. In particular, DBSCAN considers two data points as similar if they are "connected" via a sequence (path) of close-by intermediate data points. Thus, DBSCAN might consider two data points as similar (and therefore belonging to the same cluster) even if their feature vectors have a large Euclidean distance.

differentiable A function $f : \mathbb{R}^d \to \mathbb{R}$ is differentiable if it has a gradient $\nabla f(\mathbf{x})$ everywhere (for every $\mathbf{x} \in \mathbb{R}^d$) [9].

differential privacy Consider an algorithm used to compute some output based on a dataset. Differential privacy is a precise measure of this algorithm's privacy leakage. Roughly speaking, the algorithm is differentially private if the probability distribution of its output does not allow it to infer if the dataset contains a specific data point.

effective dimension The effective dimension $d_{\text{eff}}(\mathcal{H})$ of an infinite hypothesis space $\mathcal{H}$ is a measure of its size. Loosely speaking, the effective

dimension is equal to the number of "independent" tunable parameters of the model. These parameters might be the coefficients used in a linear map or the weights and bias terms of an ANN.

eigenvalue We refer to a number $\lambda \in \mathbb{R}$ as eigenvalue of a square matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$ if there is a non-zero vector $\mathbf{x} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$ such that $\mathbf{A}\mathbf{x} = \lambda \mathbf{x}$.

eigenvalue decomposition The task of computing the eigenvalues and corresponding eigenvectors of a matrix.

eigenvector An eigenvector of a matrix $\mathbf{A}$ is a non-zero vector $\mathbf{x} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$ such that $\mathbf{A}\mathbf{x} = \lambda \mathbf{x}$ with some eigenvalue λ.

empirical graph We use an ampirical graph to represent data and models in FL applications. An empirical graph is an undirected weighted empirical graph whose nodes carry local datasets and models. FL methods learn a local hypothesis $h^{(i)}$, for each node $i \in \mathcal{V}$, such that it incurs small loss on the local datasets.

empirical risk The empirical risk of a given hypothesis on a given set of data points is the average loss of the hypothesis computed over all data points in that set.

empirical risk minimization Empirical risk minimization is the optimization problem of finding the hypothesis with minimum average loss (empirical risk) on a given set of data points (the training set). Many ML methods are special cases of empirical risk minimization.

estimation error Consider data points with feature vectors $\mathbf{x}$ and label y. In some applications we can model the relation between features and

label of a data point as $y = \bar{h}(\mathbf{x}) + \varepsilon$. Here we used some true hypothesis $\bar{h}$ and a noise term ε which might represent modelling or labelling errors. The estimation error incurred by a ML method that learns a hypothesis $\widehat{h}$, e.g., using ERM, is defined as $\widehat{h} - \bar{h}$. For a parametrized hypothesis space, consisting of hypothesis maps that are determined by a parameter vector $\mathbf{w}$, we define the estimation error in terms of parameter vectors as $\Delta\mathbf{w} = \widehat{\mathbf{w}} - \overline{\mathbf{w}}$. first

Euclidean space The Euclidean space $\mathbb{R}^d$ of dimension d refers to the space of all vectors $\mathbf{x} = \big(x_1, \ldots, x_d\big)$, with real-valued entries $x_1, \ldots, x_d \in \mathbb{R}$, whose geometry is defined by the inner product $\mathbf{x}^T\mathbf{x}' = \sum_{j=1}^{d} x_j x'_j$ between any two vectors $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^d$ [9].

expectation maximization Expectation maximization is a generic technique for estimating the parameters of a probabilistic model (a parametrized probability distribution) $p(\mathbf{z}; \mathbf{w})$ from data [10–12]. Expectation maximization delivers an approximation to the maximum likelihood estimate for the model parameters $\mathbf{w}$.

expert ML aims at learning a hypothesis h that accurately predicts the label of a data point based on its features. We measure the prediction error using some loss function. Ideally we want to find a hypothesis that incurs minimum loss. One approach to make this goal precise is to use the i.i.d. assumption and use the resulting Bayes risk as the benchmark level for the (average) loss of a hypothesis. Alternatively, we might know a reference or benchmark hypothesis h' which might be obtained by some existing ML method. We can then compare the loss incurred by h

against the loss incurred by h'. Such a reference or baseline hypothesis h' is referred to as an expert. Note that an expert might deliver very poor predictions. We typically compare against many different experts and aim at incurring not much more loss than the best among those experts (this is known as regret minimization) [13, 14]. first

explainability We define the (subjective) explainability of a ML method as the level of predictability (or lack of uncertainty) in its predictions delivered to a specific human user. Quantitative measures for the (subjective) explainability can be obtained via probabilistic models for the data fed into the ML method [15, 16].

explainable empirical risk minimization An instance of structural risk minimization that adds a regularization term to the training error in ERM. The regularization term is chosen to favour hypotheses that are intrinsically explainable for a user.

explainable machine learning Explainable ML methods aim at complementing predictions with explanations for how the prediction has been obtained.

feature A feature of a data point is one of its properties that can be measured or computed in an automated fashion. For example, if a data point is a bitmap image, then we could use the red-green-blue intensities of its pixels as features. Some widely used synonyms for the term feature are "covariate","explanatory variable", "independent variable", "input (variable)", "predictor (variable)" or "regressor" [17–19]. However, this

book makes consequent use of the term features for low-level properties of data points that can be measured easily.

feature map A map that transforms the original features of a data point into new features. The so-obtained new features might be preferable over the oginal features for several reasons. For example, the shape of datasets might become simpler in the new feature space, allowing to use linear models in the new features. Another reason could be that the number of new features is much smaller which is preferable in terms of avoiding overfiiting. If the feature map delivers two new feautres, we can depict data points in a scatterplot.

feature matrix Consider a dataset $\mathcal{D}$ of m data points that are characterized by feature vectors $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}$. It is convenient to collect these feature vectors into a feature matrix $\mathbf{X} := \big(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}\big)$.

feature space The feature space of a given ML application or method is constituted by all potential values that the feature vector of a data point can take on. Within this book the most frequently used choice for the feature space is the Euclidean space $\mathbb{R}^d$ with dimension d being the number of individual features of a data point.

federated averagng (FedAvg) Federated averaging is an iterative FL algorithm that alternates between local model trainings and averaging the resulting local model parameters.Different variants of this algorithm are obtained by different techniques for the model training. The authors of [20] consider federated averaging methods where the local model training is implemented by running several GD steps

federated learning (FL) Federated learning is an umbrella term for ML methods that train models in a collaborative fashion using decentralized data and computation.

Finnish Meteorological Institute The Finnish Meteorological Institute is a government agency responsible for gathering and reporting weather data in Finland.

flow-based clustering Flow-basted clustering groups the nodes of an undirected graph by applying k-means clustering to node-wise feature vectors. These feature vectors are built from flows between carefully selected source and destination nodes [21].

Gaussian mixture model Gaussian mixture models (GMM) are a family of probabilistic models for data points. Within a GMM, the feature vector $\mathbf{x}$ of a data point is interpreted as being drawn from one out of k different multivariate normal (Gaussian) distributions indexed by $c = 1, \ldots, k$. The probability that the feature vector $\mathbf{x}$ is drawn from the c-th Gaussian distribution is denoted p_c. The GMM is parametrized by the probability p_c of $\mathbf{x}$ being drawn from the c-th Gaussian distribution as well as the mean vectors $\boldsymbol{\mu}^{(c)}$ and covariance matrices $\boldsymbol{\Sigma}^{(c)}$ for $c = 1, \ldots, k$.

General Data Protection Regulation The General Data Protection Regulation (GDPR) is a law that has been passed by the European Union (EU) and put into effect on May 25, 2018 `https://gdpr.eu/tag/gdpr/`. The GDPR imposes obligations onto organizations anywhere, so long as they target, collect or in any other way process data related to people (i.e., personal data) in the EU.

generalized total variation Generalized total variation measures the changes of vector-valued node attributes of a graph.

gradient For a real-valued function $f : \mathbb{R}^d \to \mathbb{R} : \mathbf{w} \mapsto f(\mathbf{w})$, a vector $\mathbf{a}$ such that $\lim_{\mathbf{w}\to\mathbf{w}'} \frac{f(\mathbf{w})-\left(f(\mathbf{w}')+\mathbf{a}^T(\mathbf{w}-\mathbf{w}')\right)}{\|\mathbf{w}-\mathbf{w}'\|} = 0$ is referred to as the gradient of f at $\mathbf{w}'$. If such a vector exists it is denoted $\nabla f(\mathbf{w}')$ or $\nabla f(\mathbf{w})\big|_{\mathbf{w}'}$.

gradient descent (GD) Gradient descent is an iterative method for finding the minimum of a differentiable function $f(\mathbf{w})$.

gradient step Given a differentiable real-valued function $f(\mathbf{w})$ and a vector $\mathbf{w}'$, the gradient step updates $\mathbf{w}'$ by adding the scaled negative gradient $\nabla f(\mathbf{w}')$, $\mathbf{w}' \mapsto \mathbf{w}' - \alpha\nabla f(\mathbf{w}')$.

gradient-based method Gradient-based methods are iterative algorithms for finding the minimum (or maximum) of a differentiable objective function of a parameter vector. These algorithms construct a sequence of approximations to an optimal parameter vector whose function value is minimal (or maximal). As their name indicates, gradient-based methods use the gradients of the objective function evaluated during previous iterations to construct a new (hopefully) improved approximation of an optimal parameter vector.

graph A graph $\mathcal{G} = \left(\mathcal{V}, \mathcal{E}\right)$ is a pair that consists of a node set $\mathcal{V}$ and an edge set $\mathcal{E}$. In general, a graph is specified by a map that assigns to each edge $e \in \mathcal{E}$ a pair of nodes [22]. One important family of graphs (simple undirected graphs) is obtained by identifying each edge $e \in \mathcal{E}$ with two different nodes $\{i, i'\}$. Weighted graphs also specify numeric

weights A_e for each edge $e \in \mathcal{E}$.

GTV minimization GTV minimization is an instance of regularized empirical risk minimization (RERM) using the generalized total variation (GTV) of local paramter vectors as regularization term.

hard clustering Hard clustering refers to the task of partitioning a given set of data points into (few) non-overlapping clusters. Each data point is assigned to one specific cluster.

high-dimensional regime A ML method or problem belongs to the high-dimensional regime if the effective dimension of the model is larger than the number of available (labeled) data points. For example, linear regression belongs to the high-dimensional regime whenever the number d of features used to characterize data points is larger than the number of data points in the training set. Another example for the high-dimensional regime are deep learning methods that use a hypothesis space generated by a ANN with much more tunable weights than the number of data points in the training set. The recent field of high-dimensional statistics uses probability theory to analyze ML methods in the high-dimensional regime [23, 24].

Hilbert space A Hilbert space is a linear vector space that is equipped with an inner product between pairs of vectors. One important example for a Hilbert space is the Euclidean spaces $\mathbb{R}^d$, for some dimension d, which consists of Euclidean vectors $\mathbf{u} = \big(u_1, \ldots, u_d\big)^T$ along with the inner product $\mathbf{u}^T\mathbf{v}$.

hinge loss Consider a data point that is characterized by a feature vector $\mathbf{x} \in \mathbb{R}^d$ and a binary label $y \in \{-1, 1\}$. The hinge loss incurred by a specific hypothesis h is defined as

$$L\big((\mathbf{x}, y), h\big) := \max\{0, 1 - yh(\mathbf{x})\}. \tag{2}$$

A regularized variant of the hinge loss is used by the support vector machine (SVM) [25] to learn a linear classifier with maximum margin between the two classes (see Figure 1).

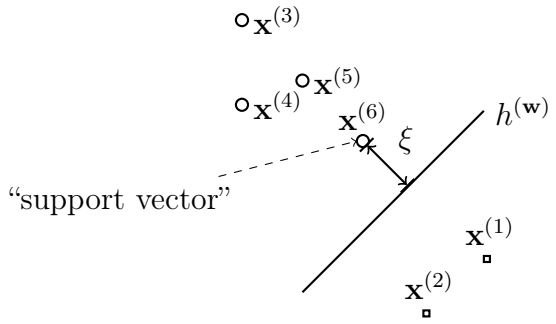

Figure 1: The SVM learns a hypothesis (or classifier) $h^{(\mathbf{w})}$ with minimum average soft-margin hinge loss. Minimizing this loss is equivalent to maximizing the margin ξ between the decision boundary of $h^{(\mathbf{w})}$ and each class of the training set.

histogram Consider a dataset $\mathcal{D}$ consisting of data points $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(m)}$ that belong to some box in $\mathbb{R}^d$. We partition this hyper-rectangle evenly into small elementary boxes. The histogram of $\mathcal{D}$ is the assignment of each elementary box to the corresponding fractions of data points in $\mathcal{D}$ that belong to this elementary box.

Huber loss The Huber loss is a mixture of the squared error loss and the absolute value of the prediction error.

hypothesis A map (or function) $h : \mathcal{X} \to \mathcal{Y}$ from the feature space $\mathcal{X}$ to the label space $\mathcal{Y}$. Given a data point with features $\mathbf{x}$ we use a hypothesis map h to estimate (or approximate) the label y using the predicted label $\hat{y} = h(\mathbf{x})$. ML is about learning (or finding) a hypothesis map h such that $y \approx h(\mathbf{x})$ for any data point.

hypothesis space Every practical ML method uses a specific hypothesis space (or model) $\mathcal{H}$. The hypothesis space of a ML method is a subset of all possible maps from the feature space to label space. The design choice of the hypothesis space should take into account available computational resources and statistical aspects. If the computational infrastructure allows for efficient matrix operations and we expect a linear relation between features and label, a resonable first candidate for the hypothesis space is a linear model.

i.i.d. It can be useful to interpret data points $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(m)}$ as realizations of independent and identically distributed RVs with a common probability distribution. If these RVs are continuous, their joint probability density function (pdf) is $p\big(\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(m)}\big) = \prod_{r=1}^{m} p\big(\mathbf{z}^{(r)}\big)$ with $p(\mathbf{z})$ being the common marginal pdf of the underlying RVs.

i.i.d. assumption The i.i.d. assumption interprets data points of a dataset as the realizations of i.i.d. RVs.

interpretability Within this book, we use the term interpretability as a

synomym for explainability.

kernel Consider data points characterized by features $\mathbf{x} \in \mathcal{X}$ with values in some arbirtrary feature space. A kernel is a map that assignes each pair of features $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ a real number. This number is interpretred as a measure for the similarity between $\mathbf{x}$ and $\mathbf{x}'$. For more details about kernels and the resulting kernel methods, we refer to the literature [25, 26].

Kullback-Leibler divergence The Kullback–Leibler divergence is a quantitive measure for how much one probability distribution is different from another probability distribution [4].

label A higher level fact or quantity of interest associated with a data point. If a data point is an image, its label might be the fact that it shows a cat (or not). Some widely used synonyms for the term label are "response variable", "output variable" or "target" [17–19].

label space Consider a ML application that involves data points characterized by features and labels. The label space of a given ML application or method is constituted by all potential values that the label of a data point can take on. A popular choice for the label space in regression problems (or methods) is $\mathcal{Y} = \mathbb{R}$. Binary classification problems (or methods) use a label space that consists of two different elements, e.g., $\mathcal{Y} = \{-1, 1\}$, $\mathcal{Y} = \{0, 1\}$ or $\mathcal{Y} = \{$"cat image","no cat image"$\}$

labeled datapoint A data point whose label is known or has been determined by some means (might require human experts).

Laplacian matrix The geometry or structure of a $\mathcal{G}$ can be analyzed using the properties of special matrices that are associated with $\mathcal{G}$. One such matrix is the graph Laplacian matrix $\mathbf{L}$ associated with an an undirected and weighted graph [27, 28].

law of large numbers The law of large numbers refers to the convergence of the average of an increasing (large) number of i.i.d. RVs to the mean (or expectation) of their common probability distribution. Different instances of the law of large numbers are obtained using different notions of convergence.

learning rate Consider an iterative method for finding or learning a good choice for a hypothesis. Such an iterative method repeats similar computational (update) steps that adjust or modify the current choice for the hypothesis to obtain an improved hypothesis. A prime example for such an iterative learning method is GD and its variants. We refer by learning rate to any parameter of an iterative learning method that controls the extent by which the current hypothesis might be modified or improved in each iteration. A prime example for such a parameter is the step size used in GD. Some authors use the term learning rate mostly as a synonym for the step size of (a variant of) GD

learning task A learning tasks consists of a specific choice for a collection of data points (e.g., all images stored in a particular database), their features and labels.

least absolute deviation regression Least absolute deviation regression uses the average of the absolute precondition errors to find a linear

hypothesis.

least absolute shrinkage and selection operator (Lasso) The least absolute shrinkage and selection operator (Lasso) is an instance of structural risk minimization (SRM) for learning the weights $\mathbf{w}$ of a linear map $h(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$. The Lasso minimizes the sum consisting of an average squared error loss (as in linear regression) and the scaled ℓ_1 norm of the weight vector $\mathbf{w}$.

linear classifier A classifier $h(\mathbf{x})$ maps the feature vector $\mathbf{x} \in \mathbb{R}^d$ of a data point to a predicted label $\hat{y} \in \mathcal{Y}$ out of a finite set of label values $\mathcal{Y}$. We can characterize such a classifier equivalently by the decision regions $\mathcal{R}_a$, for every possible label value $a \in \mathcal{Y}$. Linear classifiers are such that the boundaries between the regions $\mathcal{R}_a$ are hyperplanes in $\mathbb{R}^d$.

linear model This book uses the term linear model in a very specific sense. In particular, a linear model is the hypothesis space which consists of all linear maps,

$$\mathcal{H}^{(d)} := \left\{ h(\mathbf{x}) = \mathbf{w}^T\mathbf{x} : \mathbf{w} \in \mathbb{R}^d \right\}. \tag{3}$$

Note that (3) defines an entire family of hypothesis spaces, which is indexed by the number d of features that are linearly combined to form the prediction $h(\mathbf{x})$. The design choice of d is guided by computational aspects (smaller d means less computation), statistical aspects (larger d might reduce prediction error) and interpretability (linear models with few features are considered interpretable).

linear regression Linear regression aims at learning a linear hypothesis map to predict a numeric label based on numeric features of a data point. The quality of a linear hypothesis map is typically measured using the average squared error loss incurred on a set of labeled data points (the training set).

local dataset The concept of a local dataset is somewhat half-way between the concept of a data point and a dataset. Similar to the basic notion of a dataset, also a local dataset consists of several individual data points which are characterized by features and label. In constrast to a single dataset used in basic ML methods, a local dataset is also related to other local datasets via different notions of similarities. These similarities might arising from probabilistic models or communcation infrastructure and are encoded in the edges of an empirical graph.

logistic loss Consider a data point that is characterized by the features $\mathbf{x}$ and a binary label $y \in \{-1, 1\}$. We use a real-valued hypothesis h to predict the label y solely from the features $\mathbf{x}$. The logistic loss incurred by a specific hypothesis h is defined as

$$L\big((\mathbf{x}, y), h\big) := \log(1 + \exp(-yh(\mathbf{x}))). \tag{4}$$

. Carefully note that the expression (4) for the logistic loss applies only for the label space $\mathcal{Y} = \{-1, 1\}$ and using the thresholding rule (1).

logistic regression Logistic regression learns a linear hypothesis map (classifier) $h(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$ to predict a binary label y based on numeric features $\mathbf{x}$ of a data point. The quality of a linear hypothesis map is measured

using its average logistic loss on some labeled data points (the training set).

loss With a slight abuse of language, we use the term loss either for loss function itself or for its value for a specific pair of data point and hypothesis.

loss function A loss function is a map

$$L : \mathcal{X} \times \mathcal{Y} \times \mathcal{H} \to \mathbb{R}_+ : \big(\big(\mathbf{x}, y\big), h\big) \mapsto L\big((\mathbf{x}, y), h\big)$$

which assigns a pair consisting of a data point, with features $\mathbf{x}$ and label y, and a hypothesis $h \in \mathcal{H}$ the non-negative real number $L\big((\mathbf{x}, y), h\big)$. The loss value $L\big((\mathbf{x}, y), h\big)$ quantifies the discrepancy between the true label y and the predicted label $h(\mathbf{x})$. Smaller (closer to zero) values $L\big((\mathbf{x}, y), h\big)$ mean a smaller discrepancy between predicted label and true label of a data point. Figure 2 depicts a loss function for a given data point, with features $\mathbf{x}$ and label y, as a function of the hypothesis $h \in \mathcal{H}$.

maximum Given a set of real numbers, the maximum is the largest of those numbers.

maximum likelihood Consider data points $\mathcal{D} = \big\{\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(m)}\big\}$ that are interpreted as realizations of i.i.d. RVs with a common probability distribution $p(\mathbf{z}; \mathbf{w})$ which depends on a parameter vector $\mathbf{w} \in \mathcal{W} \subseteq \mathbb{R}^n$. Maximum likelihood methods aim at finding a parameter vector $\mathbf{w}$ such that the probability (density) $p(\mathcal{D}; \mathbf{w}) = \prod_{r=1}^{m} p(\mathbf{z}^{(r)}; \mathbf{w})$ of observing

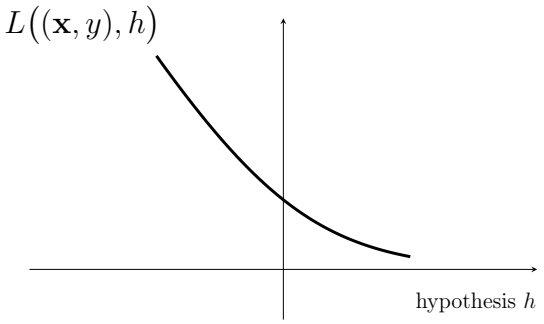

Figure 2: Some loss function $L\big((\mathbf{x}, y), h\big)$ for a fixed data point, with feature vector $\mathbf{x}$ and label y, and varying hypothesis h. ML methods try to find (learn) a hypothesis that incurs minimum loss.

the data is maximized. Thus, the maximum likelihood estimator is obtained as a solution to the optimization problem $\max_{\mathbf{w} \in \mathcal{W}} p(\mathcal{D}; \mathbf{w})$.

mean The expectation of a real-valued random variable.

mean squared estimation error Consider a ML method that uses a parametrized hypothesis space. For a given training set, whose data points are interpreted as realizations of RVs, the ML method learns the parameters incurring the estimation error $\Delta\mathbf{w}$. The mean squared estimation error is defined as the expectation $\mathbb{E}\big\{\big\|\Delta\mathbf{w}\big\|^2\big\}$ of the squared Euclidean norm of the estimation error.

metric A metric refers to a loss function that is used solely for the final performance evaluation of a learnt hypothesis. The metric is typically a loss function that has a "natural" interpretation (such as the 0/1 loss) but is not a good choice to guide the learning process, e.g., via ERM. For ERM, we typically prefer loss functions that depend smoothly on

the (parameters of the) hypothesis. Examples for such smooth loss functions include the squared error loss (**??**) and the logistic loss (**??**).

minimum Given a set of real numbers, the minimum is the smallest of those numbers.

missing data By missing data, we refer to a situation where some feature values of a subset of data points are unknown. Data imputation techniaues aim at estimating (predicting) these missing feature values [29].

model We use the term model as a synonym for hypothesis space

multi-label classification Multi-label classification problems and methods involve data points that are characterized by several individual labels.

nearest neighbour Nearest neighbour methods learn a hypothesis $h : \mathcal{X} \rightarrow \mathcal{Y}$ whose function value $h(\mathbf{x})$ is solely determined by the nearest neighbours in the feature space $\mathcal{X}$

networked data Networked data refers to a collection of local datasets that are related by some notion of pair-wise similarities. We present networked data using an empirical graph whose nodes carry local datasets and edges represent pair-wise similarities betweem them.

networked exponential families A networked collection of exponential families having a separaate parameter vector for each node of the network. These parameter vectors are coupled via the network structure.

networked federated learning Networked federated learning refers to methods that learn personalized models in a distributed fashion from local

datasets that are related by an intrinsic network structure.

networked model A networked model is a collection of local models or hypothesis spaces assigned to the nodes of an empirical graph.

node degree The degree of a node i in an undirected empirical graph is the number of its neighbours.

non i.i.d. data A dataset that cannot be well modelled as realizations of i.i.d. RVs.

non-i.i.d. See non-i.i.d. data.

non-smooth We refer to a function as non-smooth if it is not smooth [30].

norm A norm is a function that maps each element (vector) of a linear vectorspace to a non-negative number. This function must be homogeneous, definite and satisfy the triangle inequality.

objective function An objective function is a map that assigns each possible value of an optimization variable, such as the parameters $\mathbf{w}$ of a hypothesis $h^{(\mathbf{w})}$, to an objective value $f(\mathbf{w})$. The objective value $f(\mathbf{w})$ could by the the risk or the empirical risk of a hypothesis $h^{(\mathbf{w})}$.

outlier Many ML methods are motivated by the i.i.d. assumption which interprets data points as realizations of i.i.d. RVs with a common probability distribution. The i.i.d. assumption is useful for applications where the statistical properties of the data generation process are stationary (time-invariant). However, in some applications the data consists of a majority of "regular" data points that conform with an i.i.d.

assumption and a small number of data points that have fundamentally different statistical properties compared to the regular data points. We refer to a data point that substantially deviates from the statistical properties of the majority of data points as an outlier. Different methods for outlier detection use different measures for this deviation.

overfitting Consider a ML methods that uses ERM to learn a hypothesis with minimum empirical risk on a given training set. Such a method is called overfitting if it learns hypothesis with small empirical risk on the training set but unacceptly large loss outside the training set.

parameters The parameters of a ML model are tunable (learnable or adjustable) quantities that allow to choose between different hypothesis maps. For example, the linear model $\mathcal{H} := \{h : h(x) = w_1 x + w_2\}$ consists of all hypothesis maps $h(x) = w_1 x + w_2$ with a particular choice for the parameters w_1, w_2. Another example of parameters are the weights assigned to the connections of an ANN.

positive semi-definite A symmetric matrix $\mathbf{Q} = \mathbf{Q}^T \in \mathbb{R}^{d \times d}$ is referred to as positive semi-definite if $\mathbf{x}^T \mathbf{Q} \mathbf{x} \geq 0$ for every vector $\mathbf{x} \in \mathbb{R}^d$.

predictor A predictor is a hypothesis whose function values are numeric, such as real numbers. Given a data point with features $\mathbf{x}$, the predictor value $h(\mathbf{x}) \in \mathbb{R}$ is used as a prediction (estimate/guess/approximation) for the true numeric label $y \in \mathbb{R}$ of the data point.

principal component analysis (PCA) Principal component analysis determines a given number of new features that are obtained by a linear

transformation (map) of the raw features.

probabilistic model A probabilistic model interprets data as realizations of RVs with some joint probability distribution.This joint probability distribution typically involves parameters which have to be manually chosen or learnt via staitstical inference methods [1].

probabilistic PCA Probabilistic PCA extends basic principal component analysis (PCA) by using a probabilistic model for data points. Within this probabilistic model, the task of dimensionality reduction becomes an estimation problem that can be solved using expectation maximization (EM) methods.

probability density function (pdf) The probability density function (pdf) $p(x)$ of a real-valued RV $x \in \mathbb{R}$ is a particular representation of its probability distribution. If the pdf exists, it can be used to compute the probability that x takes on a value from a (measurable) set $\mathcal{B} \subseteq \mathbb{R}$ via $p(x \in \mathcal{B}) = \int_{\mathcal{B}} p(x')dx'$ [31, Ch. 3]. The pdf of a vector-valued RV $\mathbf{x} \in \mathbb{R}^d$ (if it exists) allows to compute the probability that $\mathbf{x}$ falls into a (measurable) region $\mathcal{R}$ via $p(\mathbf{x} \in \mathcal{R}) = \int_{\mathcal{R}} p(\mathbf{x}')dx'_1 \ldots dx'_d$ [31, Ch. 3].

probability distribution The data generated in some ML applications can be reasonably well modelled as realizations of a RV. The overall statistical properties (or intrinsic structure) of such data are then governed by the probability distribution of this RV. We use the term probability distribution in a highly informal manner and mean the collection of probabilities assigned to different values or value ranges of a RV. The probability distribution of a binary RV $y \in \{0, 1\}$ is fully

specified by the probabilities $p(y=0)$ and $p(y=1)\big(=1\text{–}p(y=0)\big)$. The probability distribution of a real-valued RV $x \in \mathbb{R}$ might be specified by a probability density function $p(x)$ such that $p(x \in [a,b]) \approx p(a)|b-a|$. In the most general case, a probability distribution is defined by a probability measure [32, 33].

projected GD Projected GD is obtained from basic GD by projecting the result of a gradient step into a given constrain set.

proximity operator Given a convex function and a vector $\mathbf{x}$, we define the proximity operator as

$$\mathbf{prox}_{L_i(\cdot),2\lambda}(\mathbf{w}'') := \operatorname*{argmin}_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}) + (\rho/2)\left\|\mathbf{w}-\mathbf{w}'\right\|_2^2 \text{ with } \rho > 0.$$

Convex functions for which the proximity operator can be computed efficiently are sometimes referred to as "proximable" or "simple" [34].

quadratic function A quadratic function $f(\mathbf{w})$, reading in a vector $\mathbf{w} \in \mathbb{R}^d$ as its argument, is such that

$$f(\mathbf{w}) = \mathbf{w}^T\mathbf{Q}\mathbf{w} + \mathbf{q}^T\mathbf{w} + a,$$

with some matrix $\mathbf{Q} \in \mathbb{R}^{d\times d}$, vector $\mathbf{q} \in \mathbb{R}^d$ and scalar $a \in \mathbb{R}$.

random forest A random forest is a set (ensemble) of different decision trees. Each of these decision trees is obtained by fitting a perturbed copy of the original dataset.

random variable (RV) A random variable is a mapping from a probability space $\mathcal{P}$ to a value space [33]. The probability space, whose element

are elementary events, is equipped with a probability measure that assigns a proability to subsets of $\mathcal{P}$. A binary random variable maps elementary events to a set containing two different value, such as $\{-1, 1\}$ or $\{\text{cat}, \text{no cat}\}$. A real-valued random variable maps elementary events to real numbers $\mathbb{R}$. A vector-valued random variable maps elementary events to the Euclidean space $\mathbb{R}^d$. Probability theory uses the concept of measurable spaces to rigorously define and study the properties of (large) collections of random variables [32, 33].

rectified linear unig (ReLU) The rectified linear unit or "ReLU" is a popular choice for the activation function of a neuron within an ANN. It is defined as $g(z) = \max\{0, z\}$ with z being the weighted input of the neuron.

regret The regret of a hypothesis h relative to another hypothesis h', which serves as a reference (or baseline), is the difference between the loss incurred by h and the loss incurred by h' [13]. The baseline hypothesis h' is also referred to as an expert.

regularization Regularization techniques modify the ERM principle such that the learnt hypothesis performs well also outside the training set which is used in ERM. One specific approach to regularization is by adding a penalty or regularization term to the objective function of ERM (which is the average loss on the training set). This regulazation term can be interpreted as an estimate for the increase in the expected loss (risk) compared to the average loss on the training set.

regularized empirical risk minimization Regularized empirical risk min-

imization is the problem of finding the hypothesis that optimally balances the average loss (empirical risk) on a training set with a regularization term. The regularization term penalizes a hypothesis that is not robust against (small) perturbations of the data points in the training set.

ridge regression Ridge regression aims at learning the weights $\mathbf{w}$ of linear hypothesis map $h^{(\mathbf{w})}(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$. The quality of a particular choice for the weights $\mathbf{w}$ is measured by the sum of two terms (see (**??**)). The first term is the average squared error loss incurred by $h^{(\mathbf{w})}$ on a set of labeled data points (the training set). The second term is the scaled squared Euclidean norm $\lambda\|\mathbf{w}\|_2^2$ with a regularization parameter $\lambda > 0$.

risk Consider a hypothesis h that is used to predict the label y of a data point based on its features $\mathbf{x}$. We measure the quality of a particular prediction using a loss function $L\big((\mathbf{x}, y), h\big)$. If we interpret data points as realizations of i.i.d. RVs, also the $L\big((\mathbf{x}, y), h\big)$ becomes the realization of a RV. Using such an i.i.d. assumption allows to define the risk of a hypothesis as the expected loss $\mathbb{E}\big\{L\big((\mathbf{x}, y), h\big)\big\}$. Note that the risk of h depends on both, the specific choice for the loss function and the common probability distribution of the data points.

sample A finite sequence (list) of data points $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(r)}$ that is obtained or interpreted as the realizations of m i.i.d. RVs with the common probability distribution $p(\mathbf{z})$. The length m of the sequence is referred to as the sample size.

sample size The number of individual data points contained in a dataset that is obtained from realizations of i.i.d. RVs.

scatterplot A visualization technique that depicts data points by markers in a two-dimensional plane.

semi-supervised learning Semi-supervised learning methods use (large amounts of) unlabeled data points to support the learning of a hypothesis from (a small number of) labeled data points [2].

similarity graph Some applications generate data points that are related by a domain-specific notion of similarity. These similarities can be represented conveniently using a similarity graph $\mathcal{G} = \big(\mathcal{V} := \{1, \ldots, m\}, \mathcal{E}\big)$. The node $r \in \mathcal{V}$ represents the r-th data point. Two nodes are connected by an undirected edge if the corresponding data points are similar.

smooth We refer to a real-valued function as smooth if it is differentiable and its gradient is continuous [30, 35]. In particular, a differentiable function $f(\mathbf{w})$ is refered to as β-smooth if the gradient $\nabla f(\mathbf{w})$ is Lipschitz continuous with Lipschitz constant β, i.e., $\|\nabla f(\mathbf{w}) - \nabla f(\mathbf{w}')\| \leq \beta \|\mathbf{w} - \mathbf{w}'\|$.

soft clustering Soft clustering refers to the task of partitioning a given set of data points into (few) overlapping clusters. Each data point is assigned to several different clusters with varying degree of belonging. Soft clustering amounts to determining such a degree of belonging (or soft cluster assignment) for each data point and each cluster.

spectogram The spectogram of a time signal, e.g., an audio recording, characterizes the time-frequency distribution of the signal. Loosely speaking, the spectogram quantities the signal strength at a specific time and frequency.

spectral clustering Spectral clustering groups the nodes of an undirected graph by applying k-means clustering to node-wise feature vectors. These feature vectors are built from the eigenvectors the graph Laplacian matrix [21, 27].

squared error loss The squared error loss is widely used to measure the quality of a hypothesis when predicting a numeric label from the features of a data point.

statistical aspects By statistical aspects of a ML method, we refer to (properties of) the probability distribution of its ouput given a probabilistic model for the data fed into the method.

step size Many ML methods use iterative optimization methods (such as gradient-based methods) to construct a sequence of increasingly accurate hypothesis maps $h^{(1)}, h^{(2)}, \ldots$. The rth iteration of such an algorithm starts from the current hypothesis $h^{(r)}$ and tries to modify it to obtain an improved hypothesis $h^{(r+1)}$. Iterative algorithms often use a step size (hyper-) parameter. The step size controls the amount by which a single iteration can change or modify the current hypothesis. Since the overall goal of such iteration ML methods is to learn a (approximately) optimal hypothesis we refer to a step size parameter also as a learning rate.

stochastic block model The stochastic block model (SBM) is a (family) probabilistic generative model for an undirected graph $\mathcal{G} = \big(\mathcal{V}, \mathcal{E}\big)$ with given nodeset $\mathcal{V}$. In its most basic forms, the SBM amounts to generating a graph by first randomly assigning labels $c_i \in \{1, \ldots, k\}$ to each node $i \in \mathcal{V}$ of the graph. A pair of different nodes in the graph is connected by edge with probabailiy $p_{i,j}$ that depends solely on the labels c_i, c_j.

stochastic gradient descent Stochastic GD is obtained from GD by replacing the gradient of the objective function by a noisy (or stochastic) estimate.

stopping criterion Many ML methods use iterative algorithms that construct a sequence of model parameters (such as the weights of a linear map or the weights of an ANN) that (hopefully) converge to an optimal choice for the model parameters. In practice, given finite computational resources, we need to stop iterating after a finite number of times. A stopping criterion is any well-defined condition required for stopping iterating.

strongly convex A continuously differentiable real-valued function $f(\mathbf{x})$ is strongly convex with coefficient σ if $f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) + (\sigma/2)\left\|\mathbf{y} - \mathbf{x}\right\|_2^2$ [30], [36, Sec. B.1.1.].

structural risk minimization Structural risk minimization is the problem of finding the hypothesis that optimally balances the average loss (empirical risk) on a training set with a regularization term. The regularization term penalizes a hypothesis that is not robust against (small)

perturbations of the data points in the training set.

subgradient For a real-valued function $f : \mathbb{R}^d \to \mathbb{R} : \mathbf{w} \mapsto f(\mathbf{w})$, a vector $\mathbf{a}$ such that $f(\mathbf{w}) \geq f(\mathbf{w}') + \left(\mathbf{w} - \mathbf{w}'\right)^T \mathbf{a}$ is referred to as a subgradient of f at $\mathbf{w}'$ [37, 38].

subgradient descent Subgradient descent is a generalization of GD that is obtained by using sub-gradients (instead of gradients) to construct local approximations of an objective function such as the empirical risk $\widehat{L}\left(h^{(\mathbf{w})} \middle| \mathcal{D}\right)$ as a function of the parameters $\mathbf{w}$ of a hypothesis $h^{(\mathbf{w})}$.

support vector machine A binary classification method for learning a linear map that maximally seperates data points the two classes in the feature space ("maximum margin"). Maximizing this separation is equivalent to minimizing a regularized variant of the hinge loss (**??**).

test set A set of data points that have neither been used in a training set to learn parameters of a model nor in a validation set to choose between different models (by comparing validation errors).

training error The average loss of a hypothesis when predicting the labels of data points in a training set. We sometimes refer by training error also the minimum average loss incurred on the training set by any hypothesis out of a hypothesis space.

training set A set of data points that is used in ERM to learn a hypothesis $\hat{h}$. The average loss of $\hat{h}$ on the training set is referred to as the training error. The comparison between training error and validation error of $\hat{h}$

allows to diagnose ML methods and informs how to improve them (e.g., using a different hypothesis space or collecting more data points).

validation Consider a hypothesis $\hat{h}$ that has been learn via ERM on some training set $\mathcal{D}$. Validation refers to the practice of trying out a hypothesis $\hat{h}$ on a validation set that consists of data points that are not contained in the training set $\mathcal{D}$.

validation error Consider a hypothesis $\widehat{h}$ which is obtained by ERM on a training set. The average loss of $\widehat{h}$ on a validation set, which is different from the training set, is referred to as the validation error.

validation set A set of data points that has not been used as training set in ERM to train a hypothesis $\widehat{h}$. The average loss of $\widehat{h}$ on the validation set is referred to as the validation error and used to diagnose the ML method. The comparison between training and validation error informs adaptations of the ML method (such as using a different hypothesis space).

Vapnik–Chervonenkis (VC) dimension The VC dimension of an infinite hypothesis space is a widely-used measure for its size. We refer to [39] for a precise definition of VC dimension as well as a discussion of its basic properties and use in ML.

variance The variance of a real-valued RV x is defined as the expectation $\mathbb{E}\big\{\big(x - \mathbb{E}\{x\}\big)^2\big\}$ of the squared difference x and its expectation $\mathbb{E}\{x\}$. We extend this definition to vector-valued RVs $\mathbf{x}$ as $\mathbb{E}\big\{\big\|\mathbf{x} - \mathbb{E}\{\mathbf{x}\}\big\|_2^2\big\}$.

weights We use the term weights synonymously for a finite set of parameters within a model. For example, the linear model consists of all linear maps $h(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$ that read in a feature vector $\mathbf{x} = \big(x_1, \ldots, x_d\big)^T$ of a data point. Each specific linear map is characterized by specific choices for the parameters for weights $\mathbf{w} = \big(w_1, \ldots, w_d\big)^T$.

References

[1] E. L. Lehmann and G. Casella, *Theory of Point Estimation*, 2nd ed. New York: Springer, 1998.

[2] O. Chapelle, B. Schölkopf, and A. Zien, Eds., *Semi-Supervised Learning*. Cambridge, Massachusetts: The MIT Press, 2006.

[3] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge Univ. Press, 2004.

[4] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. New Jersey: Wiley, 2006.

[5] X. Liu, H. Li, G. Xu, Z. Chen, X. Huang, and R. Lu, "Privacy-enhanced federated learning against poisoning adversaries," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 4574–4588, 2021.

[6] J. Zhang, B. Chen, X. Cheng, H. T. T. Binh, and S. Yu, "Poisongan: Generative poisoning attacks against federated learning in edge computing systems," *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3310–3322, 2021.

[7] P. Halmos, *Naive set theory*. Springer-Verlag, 1974.

[8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[9] W. Rudin, *Principles of Mathematical Analysis*, 3rd ed. New York: McGraw-Hill, 1976.

[10] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[11] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. New York, NY, USA: Springer, 2001.

[12] M. J. Wainwright and M. I. Jordan, *Graphical Models, Exponential Families, and Variational Inference*, ser. Foundations and Trends in Machine Learning. Hanover, MA: Now Publishers, 2008, vol. 1, no. 1–2.

[13] N. Cesa-Bianchi and G. Lugosi, *Prediction, Learning, and Games*. New York, NY, USA: Cambridge University Press, 2006.

[14] E. Hazan, *Introduction to Online Convex Optimization*. Now Publishers Inc., 2016.

[15] A. Jung and P. Nardelli, "An information-theoretic approach to personalized explainable machine learning," *IEEE Sig. Proc. Lett.*, vol. 27, pp. 825–829, 2020.

[16] A. Jung, "Explainable empiricial risk minimization," *submitted to IEEE Sig. Proc. Letters (preprint: https://arxiv.org/pdf/2009.01492.pdf)*, 2020.

[17] D. Gujarati and D. Porter, *Basic Econometrics*. Mc-Graw Hill, 2009.

[18] Y. Dodge, *The Oxford Dictionary of Statistical Terms*. Oxford University Press, 2003.

[19] B. Everitt, *Cambridge Dictionary of Statistics*. Cambridge University Press, 2002.

[20] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Singh and J. Zhu, Eds., vol. 54. Fort Lauderdale, FL, USA: PMLR, 20–22 Apr 2017, pp. 1273–1282. [Online]. Available: http://proceedings.mlr.press/v54/mcmahan17a.html

[21] Y. SarcheshmehPour, Y. Tian, L. Zhang, and A. Jung, "Flow-based clustering and spectral clustering: A comparison," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, 2021, pp. 1292–1296.

[22] R. T. Rockafellar, *Network Flows and Monotropic Optimization*. Athena Scientific, Jul. 1998.

[23] M. Wainwright, *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge: Cambridge University Press, 2019.

[24] P. Bühlmann and S. van de Geer, *Statistics for High-Dimensional Data*. New York: Springer, 2011.

[25] C. Lampert, "Kernel methods in computer vision," *Foundations and Trends in Computer Graphics and Vision*, 2009.

[26] B. Schölkopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, Dec. 2002.

[27] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, Dec. 2007.

[28] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Adv. Neur. Inf. Proc. Syst.*, 2001.

[29] K. Abayomi, A. Gelman, and M. A. Levy, "Diagnostics for multivariate imputations," *Journal of The Royal Statistical Society Series C-applied Statistics*, vol. 57, pp. 273–291, 2008.

[30] Y. Nesterov, *Introductory lectures on convex optimization*, ser. Applied Optimization. Kluwer Academic Publishers, Boston, MA, 2004, vol. 87, a basic course. [Online]. Available: http://dx.doi.org/10.1007/978-1-4419-8853-9

[31] D. Bertsekas and J. Tsitsiklis, *Introduction to Probability*, 2nd ed. Athena Scientific, 2008.

[32] R. Gray, *Probability, Random Processes, and Ergodic Properties*, 2nd ed. New York: Springer, 2009.

[33] P. Billingsley, *Probability and Measure*, 3rd ed. New York: Wiley, 1995.

[34] L. Condat, "A primal–dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms," *Journal of Opt. Th. and App.*, vol. 158, no. 2, pp. 460–479, Aug. 2013.

[35] S. Bubeck, "Convex optimization. algorithms and complexity." in *Foundations and Trends in Machine Learning*. Now Publishers, 2015, vol. 8.

[36] D. P. Bertsekas, *Convex Optimization Algorithms*. Athena Scientific, 2015.

[37] D. P. Bertsekas, A. Nedic, and A. E. Ozdaglar, *Convex Analysis and Optimization*. Athena Scientific, 2003.

[38] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA: Athena Scientific, June 1999.

[39] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning – from Theory to Algorithms*. Cambridge University Press, 2014.