Title: Statistical learning theory

URL: https://en.wikipedia.org/wiki/Statistical_learning_theory

PageID: 1053303

Categories: Category:Estimation theory, Category:Machine learning

---

Supervised learning

Unsupervised learning

Semi-supervised learning

Self-supervised learning

Reinforcement learning

Meta-learning

Online learning

Batch learning

Curriculum learning

Rule-based learning

Neuro-symbolic AI

Neuromorphic engineering

Quantum machine learning

Classification

Generative modeling

Regression

Clustering

Dimensionality reduction

Density estimation

Anomaly detection

Data cleaning

AutoML

Association rules

Semantic analysis

Structured prediction

Feature engineering

Feature learning

Learning to rank

Grammar induction

Ontology learning

Multimodal learning

Apprenticeship learning

Decision trees

Ensembles Bagging Boosting Random forest

Bagging

Boosting

Random forest

k -NN

Linear regression

Naive Bayes

Artificial neural networks

Logistic regression

Perceptron

Relevance vector machine (RVM)

Support vector machine (SVM)

BIRCH

CURE

Hierarchical

k -means

Fuzzy

Expectation–maximization (EM)

DBSCAN

OPTICS

Mean shift

Factor analysis

CCA

ICA

LDA

NMF

PCA

PGD

t-SNE

SDL

Graphical models Bayes net Conditional random field Hidden Markov

Bayes net

Conditional random field

Hidden Markov

RANSAC

k -NN

Local outlier factor

Isolation forest

Autoencoder

Deep learning

Feedforward neural network

Recurrent neural network LSTM GRU ESN reservoir computing

LSTM

GRU

ESN

reservoir computing

Boltzmann machine Restricted

Restricted

GAN

Diffusion model

SOM

Convolutional neural network U-Net LeNet AlexNet DeepDream

U-Net

LeNet

AlexNet

DeepDream

Neural field Neural radiance field Physics-informed neural networks

Neural radiance field

Physics-informed neural networks

Transformer Vision

Vision

Mamba

Spiking neural network

Memtransistor

Electrochemical RAM (ECRAM)

Q-learning

Policy gradient

SARSA

Temporal difference (TD)

Multi-agent Self-play

Self-play

Active learning

Crowdsourcing

Human-in-the-loop

Mechanistic interpretability

RLHF

Coefficient of determination

Confusion matrix

Learning curve

ROC curve

Kernel machines

Bias–variance tradeoff

Computational learning theory

Empirical risk minimization

Occam learning

PAC learning

Statistical learning

VC theory

Topological deep learning

AAAI

ECML PKDD

NeurIPS

ICML

ICLR

IJCAI

ML

JMLR

Glossary of artificial intelligence

List of datasets for machine-learning research List of datasets in computer vision and image processing

List of datasets in computer vision and image processing

Outline of machine learning

v

t

e

Statistical learning theory is a framework for machine learning drawing from the fields of statistics and functional analysis . [1] [2] [3] Statistical learning theory deals with the statistical inference problem of finding a predictive function based on data. Statistical learning theory has led to successful applications in fields such as computer vision , speech recognition , and bioinformatics .

Introduction

The goals of learning are understanding and prediction. Learning falls into many categories, including supervised learning , unsupervised learning , online learning , and reinforcement learning . From the perspective of statistical learning theory, supervised learning is best understood. [4] Supervised learning involves learning from a training set of data. Every point in the training is an input–output pair, where the input maps to an output. The learning problem consists of inferring the

function that maps between the input and the output, such that the learned function can be used to predict the output from future input.

Depending on the type of output, supervised learning problems are either problems of regression or problems of classification . If the output takes a continuous range of values, it is a regression problem. Using Ohm's law as an example, a regression could be performed with voltage as input and current as an output. The regression would find the functional relationship between voltage and current to be R {\displaystyle R} , such that V = I R {\displaystyle V=IR} Classification problems are those for which the output will be an element from a discrete set of labels. Classification is very common for machine learning applications. In facial recognition , for instance, a picture of a person's face would be the input, and the output label would be that person's name. The input would be represented by a large multidimensional vector whose elements represent pixels in the picture.

After learning a function based on the training set data, that function is validated on a test set of data, data that did not appear in the training set.

Formal description

Take X {\displaystyle X} to be the vector space of all possible inputs, and Y {\displaystyle Y} to be the vector space of all possible outputs. Statistical learning theory takes the perspective that there is some unknown probability distribution over the product space Z = X × Y {\displaystyle Z=X\times Y} , i.e. there exists some unknown p (z) = p (x , y) {\displaystyle p(z)=p(\mathbf {x} ,y)} . The training set is made up of n {\displaystyle n} samples from this probability distribution, and is notated S = { (x 1 , y 1) , … , (x n , y n) } = { z 1 , … , z n } {\displaystyle S=\{(\mathbf {x} _{1},y_{1}),\dots ,(\mathbf {x} _{n},y_{n})\}=\{\mathbf {z} _{1},\dots ,\mathbf {z} _{n}\}} Every x i {\displaystyle \mathbf {x} _{i}} is an input vector from the training data, and y i {\displaystyle y_{i}} is the output that corresponds to it.

In this formalism, the inference problem consists of finding a function f : X → Y {\displaystyle f:X\to Y} such that f (x) ∼ y {\displaystyle f(\mathbf {x})\sim y} . Let H {\displaystyle {\mathcal {H}}} be a space of functions f : X → Y {\displaystyle f:X\to Y} called the hypothesis space. The hypothesis space is the space of functions the algorithm will search through. Let V (f (x) , y) {\displaystyle V(f(\mathbf {x}),y)} be the loss function , a metric for the difference between the predicted value f (x) {\displaystyle f(\mathbf {x})} and the actual value y {\displaystyle y} . The expected risk is defined to be I [f] = ∫ X × Y V (f (x) , y) p (x , y) d x d y {\displaystyle I[f]=\int _{X\times Y}V(f(\mathbf {x}),y)\,p(\mathbf {x} ,y)\,d\mathbf {x} \,dy} The target function, the best possible function f {\displaystyle f} that can be chosen, is given by the f {\displaystyle f} that satisfies f = argmin h ∈ H ■ I [h] {\displaystyle f=\mathop {\operatorname {argmin} } _{h\in {\mathcal {H}}}I[h]}

Because the probability distribution p (x , y) {\displaystyle p(\mathbf {x} ,y)} is unknown, a proxy measure for the expected risk must be used. This measure is based on the training set, a sample from this unknown probability distribution. It is called the empirical risk I S [f] = 1 n ∑ i = 1 n V (f (x i) , y i) {\displaystyle I_{S}[f]={\frac {1}{n}}\sum _{i=1}^{n}V(f(\mathbf {x} _{i}),y_{i})} A learning algorithm that chooses the function f S {\displaystyle f_{S}} that minimizes the empirical risk is called empirical risk minimization .

Loss functions

The choice of loss function is a determining factor on the function f S {\displaystyle f_{S}} that will be chosen by the learning algorithm. The loss function also affects the convergence rate for an algorithm. It is important for the loss function to be convex . [5]

Different loss functions are used depending on whether the problem is one of regression or one of classification.

Regression

The most common loss function for regression is the square loss function (also known as the L2-norm). This familiar loss function is used in Ordinary Least Squares regression . The form is: V (f (x) , y) = (y − f (x)) 2 {\displaystyle V(f(\mathbf {x}),y)=(y-f(\mathbf {x}))^{2}}

The absolute value loss (also known as the L1-norm) is also sometimes used: V (f (x) , y) = | y – f (x) | {\displaystyle V(f(\mathbf {x}),y)=|y-f(\mathbf {x})|}

Classification

In some sense the 0-1 indicator function is the most natural loss function for classification. It takes the value 0 if the predicted output is the same as the actual output, and it takes the value 1 if the predicted output is different from the actual output. For binary classification with Y = { – 1 , 1 } {\displaystyle Y=\{-1,1\}} , this is: V (f (x) , y) = θ (– y f (x)) {\displaystyle V(f(\mathbf {x}),y)=\theta (-yf(\mathbf {x}))} where θ {\displaystyle \theta } is the Heaviside step function .

Regularization

In machine learning problems, a major problem that arises is that of overfitting . Because learning is a prediction problem, the goal is not to find a function that most closely fits the (previously observed) data, but to find one that will most accurately predict output from future input. Empirical risk minimization runs this risk of overfitting: finding a function that matches the data exactly but does not predict future output well.

Overfitting is symptomatic of unstable solutions; a small perturbation in the training set data would cause a large variation in the learned function. It can be shown that if the stability for the solution can be guaranteed, generalization and consistency are guaranteed as well. [6] [7] Regularization can solve the overfitting problem and give the problem stability.

Regularization can be accomplished by restricting the hypothesis space H {\displaystyle {\mathcal {H}}} . A common example would be restricting H {\displaystyle {\mathcal {H}}} to linear functions: this can be seen as a reduction to the standard problem of linear regression . H {\displaystyle {\mathcal {H}}} could also be restricted to polynomial of degree p {\displaystyle p} , exponentials, or bounded functions on L1 . Restriction of the hypothesis space avoids overfitting because the form of the potential functions are limited, and so does not allow for the choice of a function that gives empirical risk arbitrarily close to zero.

One example of regularization is Tikhonov regularization . This consists of minimizing 1 n ∑ i = 1 n V (f (x i) , y i) + γ ■ f ■ H 2 {\displaystyle {\frac {1}{n}}\sum _{i=1}^{n}V(f(\mathbf {x} _{i}),y_{i})+\gamma \left\|f\right\|_{\mathcal {H}}^{2}} where γ {\displaystyle \gamma } is a fixed and positive parameter, the regularization parameter. Tikhonov regularization ensures existence, uniqueness, and stability of the solution. [8]

Bounding empirical risk

Consider a binary classifier f : X → { 0 , 1 } {\displaystyle f:{\mathcal {X}}\to \{0,1\}} . We can apply Hoeffding's inequality to bound the probability that the empirical risk deviates from the true risk to be a Sub-Gaussian distribution . P (| R ^ (f) – R (f) | ≥ ■) ≤ 2 e – 2 n ■ 2 {\displaystyle \mathbb {P} (|{\hat {R}}(f)-R(f)|\geq \epsilon)\leq 2e^{-2n\epsilon ^{2}}} But generally, when we do empirical risk minimization, we are not given a classifier; we must choose it. Therefore, a more useful result is to bound the probability of the supremum of the difference over the whole class. P (sup f ∈ F | R ^ (f) – R (f) | ≥ ■) ≤ 2 S (F , n) e – n ■ 2 / 8 ≈ n d e – n ■ 2 / 8 {\displaystyle \mathbb {P} {\bigg (}\sup _{f\in {\mathcal {F}}}|{\hat {R}}(f)-R(f)|\geq \epsilon {\bigg)}\leq 2S({\mathcal {F}},n)e^{-n\epsilon ^{2}/8}\approx n^{d}e^{-n\epsilon ^{2}/8}} where S (F , n) {\displaystyle S({\mathcal {F}},n)} is the shattering number and n {\displaystyle n} is the number of samples in your dataset. The exponential term comes from Hoeffding but there is an extra cost of taking the supremum over the whole class, which is the shattering number.

See also

Reproducing kernel Hilbert spaces are a useful choice for H {\displaystyle {\mathcal {H}}} .

Proximal gradient methods for learning

Rademacher complexity

Vapnik–Chervonenkis dimension

References

v

t

e

Differentiable programming

Information geometry

Statistical manifold

Automatic differentiation

Neuromorphic computing

Pattern recognition

Ricci calculus

Computational learning theory

Inductive bias

IPU

TPU

VPU

Memristor

SpiNNaker

TensorFlow

PyTorch

Keras

scikit-learn

Theano

JAX

Flux.jl

MindSpore

Portals Computer programming Technology

Computer programming

Technology