Title: Kernel method

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

PageID: 3424576

Categories: Category:Classification algorithms, Category:Geostatistics, Category:Kernel methods for machine learning, Category:Pattern recognition

---

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

In machine learning , kernel machines are a class of algorithms for pattern analysis , whose best known member is the support-vector machine (SVM). These methods involve using linear classifiers to solve nonlinear problems. [1] The general task of pattern analysis is to find and study general types of relations (for example clusters , rankings , principal components , correlations , classifications) in datasets. For many algorithms that solve these tasks, the data in raw representation have to be explicitly transformed into feature vector representations via a user-specified feature map : in contrast, kernel methods require only a user-specified kernel , i.e., a similarity function over all pairs of data points computed using inner products . The feature map in kernel machines is infinite dimensional but only requires a finite dimensional matrix from user-input according to the representer theorem . Kernel machines are slow to compute for datasets larger

than a couple of thousand examples without parallel processing.

Kernel methods owe their name to the use of kernel functions , which enable them to operate in a high-dimensional, implicit feature space without ever computing the coordinates of the data in that space, but rather by simply computing the inner products between the images of all pairs of data in the feature space. This operation is often computationally cheaper than the explicit computation of the coordinates. This approach is called the " kernel trick ". [2] Kernel functions have been introduced for sequence data, graphs , text, images, as well as vectors.

Algorithms capable of operating with kernels include the kernel perceptron , support-vector machines (SVM), Gaussian processes , principal components analysis (PCA), canonical correlation analysis , ridge regression , spectral clustering , linear adaptive filters and many others.

Most kernel algorithms are based on convex optimization or eigenproblems and are statistically well-founded. Typically, their statistical properties are analyzed using statistical learning theory (for example, using Rademacher complexity).

Motivation and informal explanation

Kernel methods can be thought of as instance-based learners : rather than learning some fixed set of parameters corresponding to the features of their inputs, they instead "remember" the i {\displaystyle i} -th training example (x i , y i) {\displaystyle (\mathbf {x} _{i},y_{i})} and learn for it a corresponding weight w i {\displaystyle w_{i}} . Prediction for unlabeled inputs, i.e., those not in the training set, is treated by the application of a similarity function k {\displaystyle k} , called a kernel , between the unlabeled input x ′ {\displaystyle \mathbf {x'} } and each of the training inputs x i {\displaystyle \mathbf {x} _{i}} . For instance, a kernelized binary classifier typically computes a weighted sum of similarities y ^ = sgn ⁡ ∑ i = 1 n w i y i k (x i , x ′) , {\displaystyle {\hat {y}}=\operatorname {sgn} \sum _{i=1}^{n}w_{i}y_{i}k(\mathbf {x} _{i},\mathbf {x'}),} where

y ^ ∈ { − 1 , + 1 } {\displaystyle {\hat {y}}\in \{-1,+1\}} is the kernelized binary classifier's predicted label for the unlabeled input x ′ {\displaystyle \mathbf {x'} } whose hidden true label y {\displaystyle y} is of interest;

k : X × X → R {\displaystyle k\colon {\mathcal {X}}\times {\mathcal {X}}\to \mathbb {R} } is the kernel function that measures similarity between any pair of inputs x , x ′ ∈ X {\displaystyle \mathbf {x} ,\mathbf {x'} \in {\mathcal {X}}} ;

the sum ranges over the n labeled examples { (x i , y i) } i = 1 n {\displaystyle \{(\mathbf {x} _{i},y_{i})\}_{i=1}^{n}} in the classifier's training set, with y i ∈ { − 1 , + 1 } {\displaystyle y_{i}\in \{-1,+1\}} ;

the w i ∈ R {\displaystyle w_{i}\in \mathbb {R} } are the weights for the training examples, as determined by the learning algorithm;

the sign function sgn ⁡ {\displaystyle \operatorname {sgn} } determines whether the predicted classification y ^ {\displaystyle {\hat {y}}} comes out positive or negative.

Kernel classifiers were described as early as the 1960s, with the invention of the kernel perceptron . [3] They rose to great prominence with the popularity of the support-vector machine (SVM) in the 1990s, when the SVM was found to be competitive with neural networks on tasks such as handwriting recognition .

Mathematics: the kernel trick

The kernel trick avoids the explicit mapping that is needed to get linear learning algorithms to learn a nonlinear function or decision boundary . For all x {\displaystyle \mathbf {x} } and x ′ {\displaystyle \mathbf {x'} } in the input space X {\displaystyle {\mathcal {X}}} , certain functions k (x , x ′) {\displaystyle k(\mathbf {x} ,\mathbf {x'})} can be expressed as an inner product in another space V {\displaystyle {\mathcal {V}}} . The function k : X × X → R {\displaystyle k\colon {\mathcal {X}}\times {\mathcal {X}}\to \mathbb {R} } is often referred to as a kernel or a kernel function . The word "kernel" is used in mathematics to denote a weighting function for a weighted sum or integral .

Certain problems in machine learning have more structure than an arbitrary weighting function k {\displaystyle k} . The computation is made much simpler if the kernel can be written in the form of a

"feature map" ϕ : X → V {\displaystyle \varphi \colon {\mathcal {X}}\to {\mathcal {V}}} which satisfies k (x , x ′) = ■ ϕ (x) , ϕ (x ′) ■ V . {\displaystyle k(\mathbf {x} ,\mathbf {x'})=\langle \varphi (\mathbf {x}),\varphi (\mathbf {x'})\rangle _{\mathcal {V}}.} The key restriction is that ■ ⋅ , ⋅ ■ V {\displaystyle \langle \cdot ,\cdot \rangle _{\mathcal {V}}} must be a proper inner product. On the other hand, an explicit representation for ϕ {\displaystyle \varphi } is not necessary, as long as V {\displaystyle {\mathcal {V}}} is an inner product space . The alternative follows from Mercer's theorem : an implicitly defined function ϕ {\displaystyle \varphi } exists whenever the space X {\displaystyle {\mathcal {X}}} can be equipped with a suitable measure ensuring the function k {\displaystyle k} satisfies Mercer's condition .

Mercer's theorem is similar to a generalization of the result from linear algebra that associates an inner product to any positive-definite matrix . In fact, Mercer's condition can be reduced to this simpler case. If we choose as our measure the counting measure μ (T) = | T | {\displaystyle \mu (T)=|T|} for all T ⊂ X {\displaystyle T\subset X} , which counts the number of points inside the set T {\displaystyle T} , then the integral in Mercer's theorem reduces to a summation ∑ i = 1 n ∑ j = 1 n k (x i , x j) c i c j ≥ 0. {\displaystyle \sum _{i=1}^{n}\sum _{j=1}^{n}k(\mathbf {x} _{i},\mathbf {x} _{j})c_{i}c_{j}\geq 0.} If this summation holds for all finite sequences of points (x 1 , … , x n) {\displaystyle (\mathbf {x} _{1},\dotsc ,\mathbf {x} _{n})} in X {\displaystyle {\mathcal {X}}} and all choices of n {\displaystyle n} real-valued coefficients (c 1 , … , c n) {\displaystyle (c_{1},\dots ,c_{n})} (cf. positive definite kernel), then the function k {\displaystyle k} satisfies Mercer's condition.

Some algorithms that depend on arbitrary relationships in the native space X {\displaystyle {\mathcal {X}}} would, in fact, have a linear interpretation in a different setting: the range space of ϕ {\displaystyle \varphi } . The linear interpretation gives us insight about the algorithm. Furthermore, there is often no need to compute ϕ {\displaystyle \varphi } directly during computation, as is the case with support-vector machines . Some cite this running time shortcut as the primary benefit. Researchers also use it to justify the meanings and properties of existing algorithms.

Theoretically, a Gram matrix K ∈ R n × n {\displaystyle \mathbf {K} \in \mathbb {R} ^{n\times n}} with respect to { x 1 , … , x n } {\displaystyle \{\mathbf {x} _{1},\dotsc ,\mathbf {x} _{n}\}} (sometimes also called a "kernel matrix" [4]), where K i j = k (x i , x j) {\displaystyle K_{ij}=k(\mathbf {x} _{i},\mathbf {x} _{j})} , must be positive semi-definite (PSD) . [5] Empirically, for machine learning heuristics, choices of a function k {\displaystyle k} that do not satisfy Mercer's condition may still perform reasonably if k {\displaystyle k} at least approximates the intuitive idea of similarity. [6] Regardless of whether k {\displaystyle k} is a Mercer kernel, k {\displaystyle k} may still be referred to as a "kernel".

If the kernel function k {\displaystyle k} is also a covariance function as used in Gaussian processes , then the Gram matrix K {\displaystyle \mathbf {K} } can also be called a covariance matrix . [7]

Applications

Application areas of kernel methods are diverse and include geostatistics , [8] kriging , inverse distance weighting , 3D reconstruction , bioinformatics , cheminformatics , information extraction and handwriting recognition .

Popular kernels

Fisher kernel

Graph kernels

Kernel smoother

Polynomial kernel

Radial basis function kernel (RBF)

String kernels

Neural tangent kernel

Neural network Gaussian process (NNGP) kernel

See also

Kernel methods for vector output

Kernel density estimation

Representer theorem

Similarity learning

Cover's theorem

References

Further reading

Shawe-Taylor, J. ; Cristianini, N. (2004). Kernel Methods for Pattern Analysis . Cambridge University Press. ISBN 9780511809682 .

Liu, W.; Principe, J.; Haykin, S. (2010). Kernel Adaptive Filtering: A Comprehensive Introduction . Wiley. ISBN 9781118211212 .

Schölkopf, B. ; Smola, A. J.; Bach, F. (2018). Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond . MIT Press. ISBN 978-0-262-53657-8 .

External links

Kernel-Machines Org —community website

onlineprediction.net Kernel Methods Article