Title: Stability (learning theory)

URL: https://en.wikipedia.org/wiki/Stability_(learning_theory)

PageID: 33886025

Categories: Category:Learning, Category:Machine learning

Stability , also known as algorithmic stability , is a notion in computational learning theory of how a machine learning algorithm output is changed with small perturbations to its inputs. A stable learning algorithm is one for which the prediction does not change much when the training data is modified slightly. For instance, consider a machine learning algorithm that is being trained to recognize handwritten letters of the alphabet, using 1000 examples of handwritten letters and their labels ("A" to "Z") as a training set. One way to modify this training set is to leave out an example, so that only 999 examples of handwritten letters and their labels are available. A stable learning algorithm would produce a similar classifier with both the 1000-element and 999-element training sets.

Stability can be studied for many types of learning problems, from language learning to inverse problems in physics and engineering, as it is a property of the learning process rather than the type of information being learned. The study of stability gained importance in computational learning theory in the 2000s when it was shown to have a connection with generalization . [1] It was shown that for large classes of learning algorithms, notably empirical risk minimization algorithms, certain types of stability ensure good generalization.

History

A central goal in designing a machine learning system is to guarantee that the learning algorithm will generalize , or perform accurately on new examples after being trained on a finite number of them. In the 1990s, milestones were reached in obtaining generalization bounds for supervised learning algorithms . The technique historically used to prove generalization was to show that an algorithm was consistent , using the uniform convergence properties of empirical quantities to their means. This technique was used to obtain generalization bounds for the large class of empirical risk minimization (ERM) algorithms. An ERM algorithm is one that selects a solution from a hypothesis space H {\displaystyle H} in such a way to minimize the empirical error on a training set S {\displaystyle S} .

A general result, proved by Vladimir Vapnik for an ERM binary classification algorithms, is that for any target function and input distribution, any hypothesis space H {\displaystyle H} with VC-dimension d {\displaystyle d} , and n {\displaystyle n} training examples, the algorithm is consistent and will produce a training error that is at most O (d n) {\displaystyle O\left({\sqrt {\frac {d}{n}}}\right)} (plus logarithmic factors) from the true error. The result was later extended to almost-ERM algorithms with function classes that do not have unique minimizers.

Vapnik's work, using what became known as VC theory , established a relationship between generalization of a learning algorithm and properties of the hypothesis space H {\displaystyle H} of functions being learned. However, these results could not be applied to algorithms with hypothesis spaces of unbounded VC-dimension. Put another way, these results could not be applied when the information being learned had a complexity that was too large to measure. Some of the simplest machine learning algorithms—for instance, for regression—have hypothesis spaces with unbounded VC-dimension. Another example is language learning algorithms that can produce sentences of arbitrary length.

Stability analysis was developed in the 2000s for computational learning theory and is an alternative method for obtaining generalization bounds. The stability of an algorithm is a property of the learning process, rather than a direct property of the hypothesis space H {\displaystyle H} , and it can be assessed in algorithms that have hypothesis spaces with unbounded or undefined VC-dimension such as nearest neighbor. A stable learning algorithm is one for which the learned

function does not change much when the training set is slightly modified, for instance by leaving out an example. A measure of Leave one out error is used in a Cross Validation Leave One Out (CVloo) algorithm to evaluate a learning algorithm's stability with respect to the loss function. As such, stability analysis is the application of sensitivity analysis to machine learning.

Summary of classic results

Early 1900s - Stability in learning theory was earliest described in terms of continuity of the learning map L {\displaystyle L} , traced to Andrey Nikolayevich Tikhonov [citation needed] .

1979 - Devroye and Wagner observed that the leave-one-out behavior of an algorithm is related to its sensitivity to small changes in the sample. [2]

1999 - Kearns and Ron discovered a connection between finite VC-dimension and stability. [3]

2002 - In a landmark paper, Bousquet and Elisseeff proposed the notion of uniform hypothesis stability of a learning algorithm and showed that it implies low generalization error. Uniform hypothesis stability, however, is a strong condition that does not apply to large classes of algorithms, including ERM algorithms with a hypothesis space of only two functions. [4]

2002 - Kutin and Niyogi extended Bousquet and Elisseeff's results by providing generalization bounds for several weaker forms of stability which they called almost-everywhere stability . Furthermore, they took an initial step in establishing the relationship between stability and consistency in ERM algorithms in the Probably Approximately Correct (PAC) setting. [5]

2004 - Poggio et al. proved a general relationship between stability and ERM consistency. They proposed a statistical form of leave-one-out-stability which they called CVEEEloo stability , and showed that it is a) sufficient for generalization in bounded loss classes, and b) necessary and sufficient for consistency (and thus generalization) of ERM algorithms for certain loss functions such as the square loss, the absolute value and the binary classification loss. [6]

2010 - Shalev Shwartz et al. noticed problems with the original results of Vapnik due to the complex relations between hypothesis space and loss class. They discuss stability notions that capture different loss classes and different types of learning, supervised and unsupervised. [7]

2016 - Moritz Hardt et al. proved stability of gradient descent given certain assumption on the hypothesis and number of times each instance is used to update the model. [8]

Preliminary definitions

We define several terms related to learning algorithms training sets, so that we can then define stability in multiple ways and present theorems from the field.

A machine learning algorithm, also known as a learning map L {\displaystyle L} , maps a training data set, which is a set of labeled examples (x , y) {\displaystyle (x,y)} , onto a function f {\displaystyle f} from X {\displaystyle X} to Y {\displaystyle Y} , where X {\displaystyle X} and Y {\displaystyle Y} are in the same space of the training examples. The functions f {\displaystyle f} are selected from a hypothesis space of functions called H {\displaystyle H} .

The training set from which an algorithm learns is defined as

S = { z 1 = (x 1 , y 1) , . . . , z m = (x m , y m) } {\displaystyle S=\{z_{1}=(x_{1},\ y_{1})\ ,..,\ z_{m}=(x_{m},\ y_{m})\}}

and is of size m {\displaystyle m} in Z = X × Y {\displaystyle Z=X\times Y}

drawn i.i.d. from an unknown distribution D.

Thus, the learning map L {\displaystyle L} is defined as a mapping from Z m {\displaystyle Z_{m}} into H {\displaystyle H} , mapping a training set S {\displaystyle S} onto a function f S {\displaystyle f_{S}} from X {\displaystyle X} to Y {\displaystyle Y} . Here, we consider only deterministic algorithms where L {\displaystyle L} is symmetric with respect to S {\displaystyle S} , i.e. it does not depend on the order of the elements in the training set. Furthermore, we assume that all functions are measurable and all sets are countable.

The loss V {\displaystyle V} of a hypothesis f {\displaystyle f} with respect to an example z = (x , y) {\displaystyle z=(x,y)} is then defined as V (f , z) = V (f (x) , y) {\displaystyle V(f,z)=V(f(x),y)} .

The empirical error of f {\displaystyle f} is I S [f] = 1 n ∑ V (f , z i) {\displaystyle I_{S}[f]={\frac {1}{n}}\sum V(f,z_{i})} .

The true error of f {\displaystyle f} is I [f] = E z V (f , z) {\displaystyle I[f]=\mathbb {E} _{z}V(f,z)}

Given a training set S of size m, we will build, for all i = 1....,m, modified training sets as follows:

By removing the i-th element

S | i = { z 1 , . . . , z i – 1 , z i + 1 , . . . , z m } {\displaystyle S^{|i}=\{z_{1},...,\ z_{i-1},\ z_{i+1},...,\ z_{m}\}}

By replacing the i-th element

S i = { z 1 , . . . , z i – 1 , z i ′ , z i + 1 , . . . , z m } {\displaystyle S^{i}=\{z_{1},...,\ z_{i-1},\ z_{i}',\ z_{i+1},...,\ z_{m}\}}

Definitions of stability

Hypothesis Stability

An algorithm L {\displaystyle L} has hypothesis stability β with respect to the loss function V if the following holds:

∀ i ∈ { 1 , . . . , m } , E S , z [| V (f S , z) – V (f S | i , z) |] ≤ β . {\displaystyle \forall i\in \{1,...,m\},\mathbb {E} _{S,z}[|V(f_{S},z)-V(f_{S^{|i}},z)|]\leq \beta .}

Point-wise Hypothesis Stability

An algorithm L {\displaystyle L} has point-wise hypothesis stability β with respect to the loss function V if the following holds:

∀ i ∈ { 1 , . . . , m } , E S [| V (f S , z i) – V (f S | i , z i) |] ≤ β . {\displaystyle \forall i\in \{1,...,m\},\mathbb {E} _{S}[|V(f_{S},z_{i})-V(f_{S^{|i}},z_{i})|]\leq \beta .}

Error Stability

An algorithm L {\displaystyle L} has error stability β with respect to the loss function V if the following holds:

∀ S ∈ Z m , ∀ i ∈ { 1 , . . . , m } , | E z [V (f S , z)] – E z [V (f S | i , z)] | ≤ β {\displaystyle \forall S\in Z^{m},\forall i\in \{1,...,m\},|\mathbb {E} _{z}[V(f_{S},z)]-\mathbb {E} _{z}[V(f_{S^{|i}},z)]|\leq \beta }

Uniform Stability

An algorithm L {\displaystyle L} has uniform stability β with respect to the loss function V if the following holds:

∀ S ∈ Z m , ∀ i ∈ { 1 , . . . , m } , sup z ∈ Z | V (f S , z) – V (f S | i , z) | ≤ β {\displaystyle \forall S\in Z^{m},\forall i\in \{1,...,m\},\sup _{z\in Z}|V(f_{S},z)-V(f_{S^{|i}},z)|\leq \beta }

A probabilistic version of uniform stability β is:

∀ S ∈ Z m , ∀ i ∈ { 1 , . . . , m } , P S { sup z ∈ Z | V (f S , z) – V (f S | i , z) | ≤ β } ≥ 1 – δ {\displaystyle \forall S\in Z^{m},\forall i\in \{1,...,m\},\mathbb {P} _{S}\{\sup _{z\in Z}|V(f_{S},z)-V(f_{S^{|i}},z)|\leq \beta \}\geq 1-\delta }

An algorithm is said to be stable , when the value of β {\displaystyle \beta } decreases as O (1 m) {\displaystyle O({\frac {1}{m}})} .

Leave-one-out cross-validation (CVloo) Stability

An algorithm L {\displaystyle L} has CVloo stability β with respect to the loss function V if the following holds:

$\forall i \in \{1,\ldots,m\}, P_S\{|V(f_S,z_i)-V(f_{S^{|i|}},z_i)|\leq \beta_{CV}\}\geq 1-\delta_{CV}$

The definition of (CVloo) Stability is equivalent to Pointwise-hypothesis stability seen earlier.

Expected-leave-one-out error ($Eloo_{err}$) Stability

An algorithm L has $Eloo_{err}$ stability if for each n there exists a β_{EL}^{m} and a δ_{EL}^{m} such that:

$\forall i \in \{1,\ldots,m\}, P_S\{|I[f_S]-\frac{1}{m}\sum_{i=1}^{m}V(f_{S^{|i|}},z_i)|\leq \beta_{EL}^{m}\}\geq 1-\delta_{EL}^{m}$, with β_{EL}^{m} and δ_{EL}^{m} going to zero for $m \rightarrow \infty$

Classic theorems

From Bousquet and Elisseeff (02) :

For symmetric learning algorithms with bounded loss, if the algorithm has Uniform Stability with the probabilistic definition above, then the algorithm generalizes.

Uniform Stability is a strong condition which is not met by all algorithms but is, surprisingly, met by the large and important class of Regularization algorithms.

The generalization bound is given in the article.

From Mukherjee et al. (06) :

For symmetric learning algorithms with bounded loss, if the algorithm has both Leave-one-out cross-validation (CVloo) Stability and Expected-leave-one-out error ($Eloo_{err}$) Stability as defined above, then the algorithm generalizes.

Neither condition alone is sufficient for generalization. However, both together ensure generalization (while the converse is not true).

For ERM algorithms specifically (say for the square loss), Leave-one-out cross-validation (CVloo) Stability is both necessary and sufficient for consistency and generalization.

This is an important result for the foundations of learning theory, because it shows that two previously unrelated properties of an algorithm, stability and consistency, are equivalent for ERM (and certain loss functions).

The generalization bound is given in the article.

Algorithms that are stable

This is a list of algorithms that have been shown to be stable, and the article where the associated generalization bounds are provided.

Linear regression [9]

k-NN classifier with a {0-1} loss function. [2]

Support Vector Machine (SVM) classification with a bounded kernel and where the regularizer is a norm in a Reproducing Kernel Hilbert Space. A large regularization constant C leads to good stability. [4]

Soft margin SVM classification. [4]

Regularized Least Squares regression. [4]

The minimum relative entropy algorithm for classification. [4]

A version of bagging regularizers with the number k of regressors increasing with n. [10]

Multi-class SVM classification. [10]

All learning algorithms with Tikhonov regularization satisfies Uniform Stability criteria and are, thus, generalizable. [11]

References

Further reading

S.Kutin and P.Niyogi.Almost-everywhere algorithmic stability and generalization error. In Proc. of UAI 18, 2002

S. Rakhlin, S. Mukherjee, and T. Poggio. Stability results in learning theory. Analysis and Applications, 3(4):397–419, 2005

V.N. Vapnik. The Nature of Statistical Learning Theory. Springer, 1995

Vapnik, V., Statistical Learning Theory. Wiley, New York, 1998

Poggio, T., Rifkin, R., Mukherjee, S. and Niyogi, P., "Learning Theory: general conditions for predictivity", Nature, Vol. 428, 419-422, 2004

Andre Elisseeff, Theodoros Evgeniou, Massimiliano Pontil, Stability of Randomized Learning Algorithms, Journal of Machine Learning Research 6, 55–79, 2010

Elisseeff, A. Pontil, M., Leave-one-out Error and Stability of Learning Algorithms with Applications, NATO SCIENCE SERIES SUB SERIES III COMPUTER AND SYSTEMS SCIENCES, 2003, VOL 190, pages 111-130

Shalev Shwartz, S., Shamir, O., Srebro, N., Sridharan, K., Learnability, Stability and Uniform Convergence, Journal of Machine Learning Research, 11(Oct):2635-2670, 2010