Learning Theory: stable hypotheses are predictive

Work with Sayan Mukherjee, Ryan Rifkin and Partha Niyogi

Plan

1. **Learning: well-posedness and predictivity**
2. **The supervised learning problem and generalization**
3. **ERM and conditions for generalization (and consistency)**
4. **Motivations for stability: inverse problems and beyond ERM**
5. **Stability definitions**
6. **Theorem a: stability implies generalization**
7. **Theorem b: ERM stability is necessary and sufficient for consistency**
8. Stability of non-ERM algorithms
9. Open problems: hypothesis stability and expected error stability
10. On-line algorithms: stability and generalization?

1. Learning: well-posedness and predictivity

Two key, separate motivations in recent work in the area of learning:

- in "classical" learning theory: learning must be predictive, that is it must *generalize*. For ERM generalization implies consistency. Conditions for consistency of ERM.

- for several algorithms: learning is ill-posed and algorithms must restore well-posedness, especially stability.

In other words...there are two key issues in solving the learning problem:

1. *predictivity* (which translates into **generalization**)

2. **stability** (eg *well-posedness*) of the solution

A priori no connection between *generalization and stability*. In fact there is and we show that for ERM they are equivalent.

Learning, a direction for future research: beyond classical theory

The classical learning theory due to Vapnik et al consists of necessary and sufficient conditions for *learnability* ie *generalization* in the case of about ERM. It would be desirable to have more general conditions that guarantee generalization for arbitrary algorithms and subsume the classical theory in the case of ERM.

Our results show that some specific notions of *stability* may provide a more general theory than the classical condtions on $\mathcal{H}$ and subsume them for ERM.

Preliminary: convergence in probability

Let $\{X_n\}$ be a sequence of bounded random variables. We say that

$$\lim_{n\to\infty} X_n = X \textit{ in probability}$$

if

$$\forall \varepsilon > 0 \lim_{n\to\infty} \mathbb{P}\{\|X_n - X\| \geq \varepsilon\} = 0.$$

or

if for each n there exists a ε_n and a δ_n such that

$$\mathbb{P}\{\|X_n - X\| \geq \varepsilon_n\} \leq \delta_n,$$

with ε_n and δ_n going to zero for $n \to \infty$.

2. The supervised learning problem and generalization

- The learning problem
- Classification and regression
- Loss functions
- Empirical error, generalization error, generalization

The learning problem

There is an unknown **probability distribution** on the product space $Z = XxY$, written $\mu(z) = \mu(x, y)$. We assume that X is a compact domain in Euclidean space and Y a closed subset of $\mathbb{R}^k$.

The **training set** $S = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\} = z_1, ...z_n$ consists of n samples drawn i.i.d. from μ.

$\mathcal{H}$ is the **hypothesis space**, a space of functions $f : X \to Y$.

A **learning algorithm** is a map $L : Z^n \to \mathcal{H}$ that looks at S and selects from $\mathcal{H}$ a function $f_S : \mathbf{x} \to y$ such that $f_S(\mathbf{x}) \approx y$ *in a predictive way*.

Classification and regression

If y is a real-valued random variable, we have **regression**.

If y takes values from a finite set, we have **pattern classification**. In two-class pattern classification problems, we assign one class a y value of 1, and the other class a y value of -1.

Loss Functions

In order to measure goodness of our function, we need a **loss function** V. We let $V(f(x), y) = V(f, z)$ denote the price we pay when we see x and guess that the associated y value is $f(x)$ when it is actually y. We require that *for any $f \in \mathcal{H}$ and $z \in Z$ V is bounded,* $0 \leq V(f, z) \leq M$. We can think of the set $\mathcal{L}$ of functions $\ell(z) = V(f, z)$ with $\ell : Z \to \mathbb{R}$, induced by $\mathcal{H}$ and V.

The most common loss function is square loss or L2 loss:

$$V(f(x), y) = (f(x) - y)^2$$

Empirical error, generalization error, generalization

Given a function f, a loss function V, and a probability distribution μ over Z, the **expected or true error** of f is:

$$I[f] = \mathbb{E}_z V[f, z] = \int_Z V(f, z) d\mu(z)$$

which is the **expected loss** on a new example drawn at random from μ.

We would like to make $I[f]$ small, but in general we do not know μ.

Given a function f, a loss function V, and a training set S consisting of n data points, the **empirical error** of f is:

$$I_S[f] = \frac{1}{n} \sum V(f, z_i)$$

Empirical error, generalization error, generalization

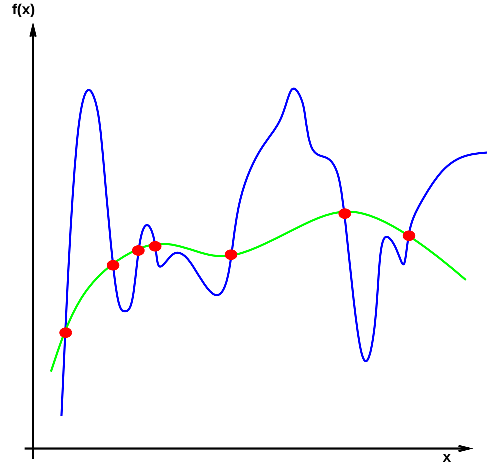
A very natural requirement for f_S is distribution independent **generalization**

$$\forall \mu, \lim_{n \to \infty} |I_S[f_S] - I[f_S]| = 0 \textit{ in probability}$$

A desirable additional requirement is **universal consistency**

$$\forall \varepsilon > 0 \lim_{n \to \infty} \sup_{\mu} \mathbb{P}_S \left\{ I[f_S] > \inf_{f \in \mathcal{H}} I[f] + \varepsilon \right\} = 0.$$

Generalization



In the figure the data was generated from the "true" green function. The blue function fits the data set and therefore has zero empirical error ($I_S[f_{blue}] = 0$). Yet it is clear that on future data, this function f_{blue} will perform poorly as it is far from the true function on most of the domain. Therefore $I[f_{blue}]$ is large. Generalization refers to whether the empirical performance on the training set ($I_S[f]$) will generalize to test performance on future examples ($I[f]$). If an algorithm is guaranteed to generalize, an absolute measure of its future predictivity can be determined from its empirical performance.

3. ERM and conditions for generalization (and consistency)

Given a training set S and a function space $\mathcal{H}$, empirical risk minimization (Vapnik) is the algorithm that looks at S and selects f_S as

$$f_S = \arg\min_{f \in \mathcal{H}} I_S(f)$$

This problem does not in general show generalization and is also **ill-posed**, depending on the choice of $\mathcal{H}$.

If the minimum does not exist we can work with the infimum.

Notice: *For ERM generalization and consistency are equivalent*

Classical conditions for consistency of ERM

Uniform Glivenko-Cantelli Classes

$\mathcal{G}$ is a (weak) uniform Glivenko-Cantelli (uGC) class of functions

if

$$\forall \varepsilon > 0 \quad \lim_{n \to \infty} \sup_{\mu} \mathbb{P} \left\{ \sup_{g \in \mathcal{G}} \left| \frac{1}{n} \sum_{i}^{n} g(z_i) - \mathbb{E}_{\mu} g \right| > \varepsilon \right\} = 0.$$

Theorem [Vapnik and Červonenkis (71), Alon et al (97), Dudley, Giné, and Zinn (91)]

A necessary and sufficient condition for consistency of ERM is that the class of loss functions $\ell(z) = V(f, z)$ (defined for for a fixed V and $f \in \mathcal{H}$) is uGC.

...mapping notation and results in CuckerSmale...

$$\epsilon(f) \longleftrightarrow I(f)$$

$$\epsilon_z(f) \longleftrightarrow I_S(f)$$

Thus

$$L_z \longleftrightarrow I(f) - I_S(f)$$

For ERM

$$f_z \longleftrightarrow f_S$$

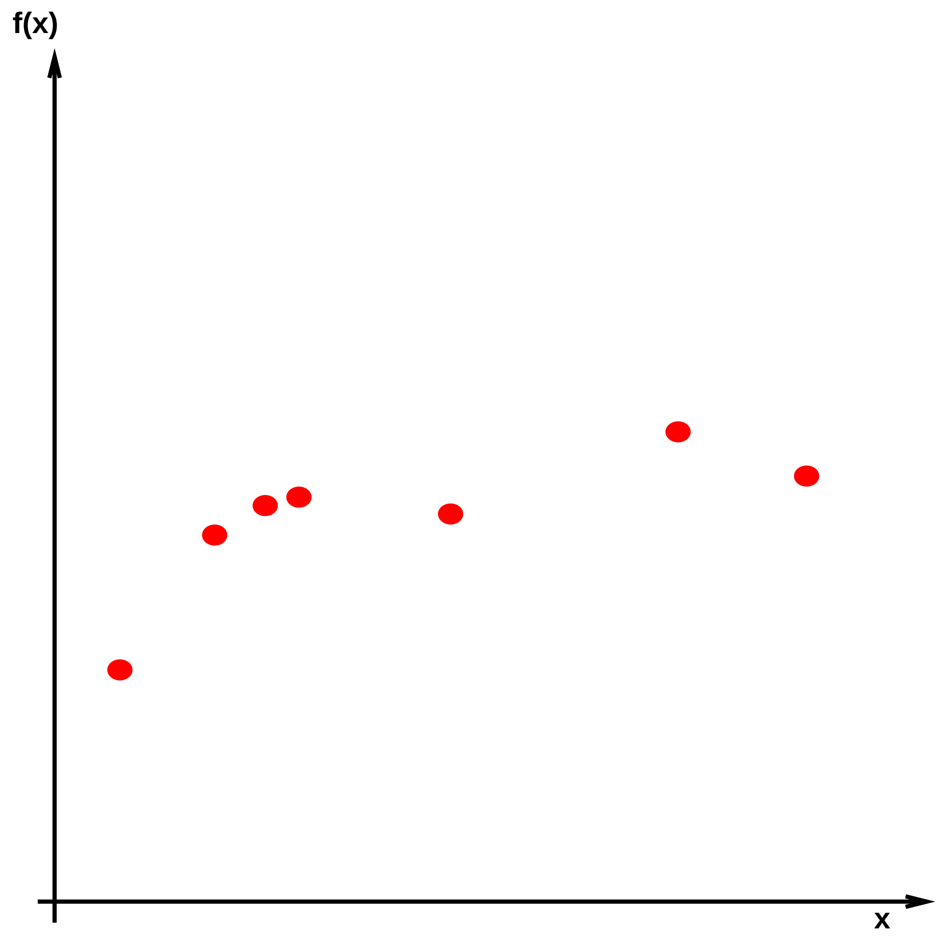
Theorem B (for $\mathcal{H}$ compact) $\longleftrightarrow$ *generalization*, see Theorem a (for general algorithms and general $\mathcal{H}$)

Theorem C (eg $\epsilon_{\mathcal{H}}(f_z) \rightarrow 0$) $\longleftrightarrow$ Theorem b (consistency of ERM) where $\epsilon_{\mathcal{H}}(f) = \epsilon(f) - \epsilon(f_{\mathcal{H}})$,

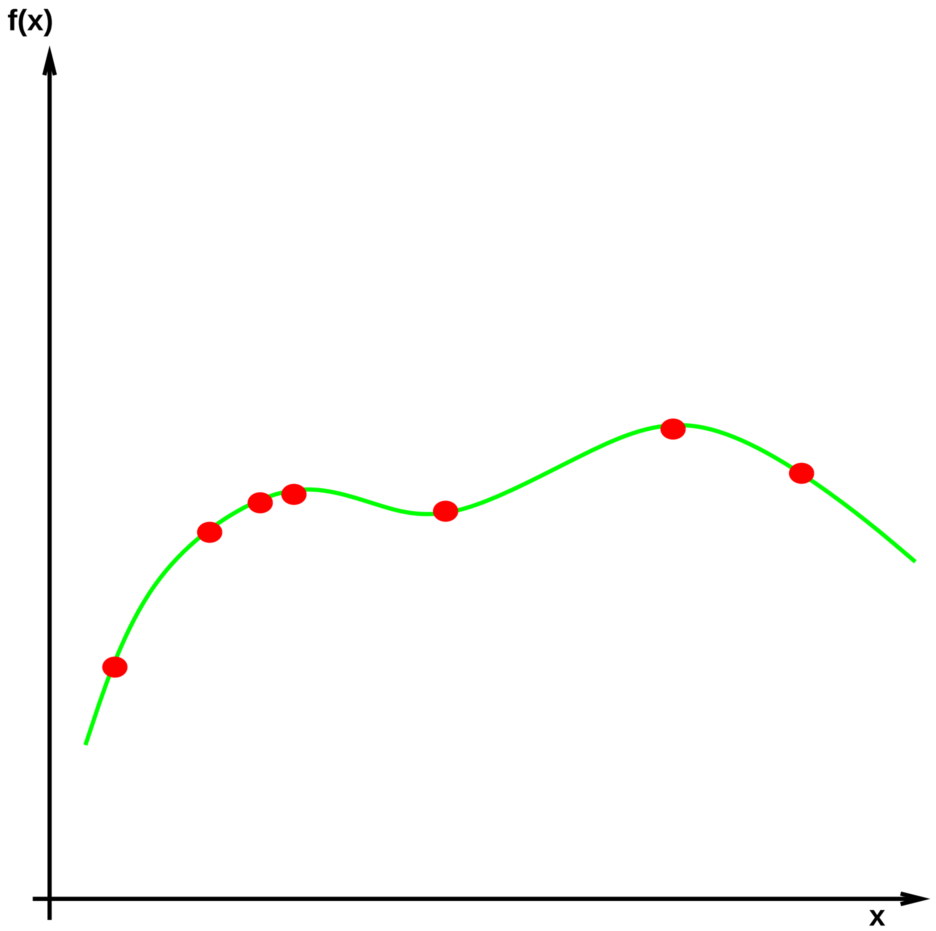
Plan

1. Learning: well-posedness and predictivity
2. The supervised learning problem and generalization
3. ERM and conditions for generalization (and consistency)
4. **Motivations for stability: inverse problems and beyond ERM**
5. Stability definitions
6. Theorem a: stability implies generalization
7. Theorem b: ERM stability is necessary and sufficient for consistency
8. Stability of non-ERM algorithms
9. Open problems: hypothesis stability and expected error stability
10. On-line algorithms: stability and generalization?

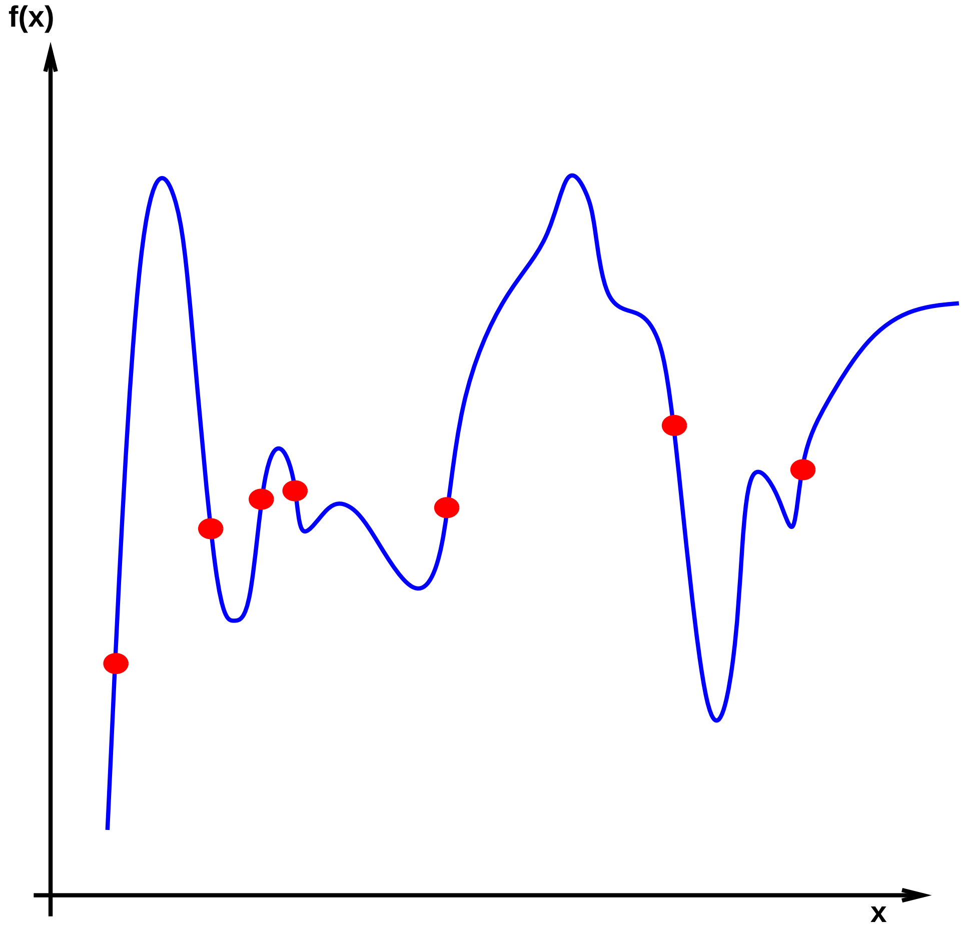
Given a certain number of samples...



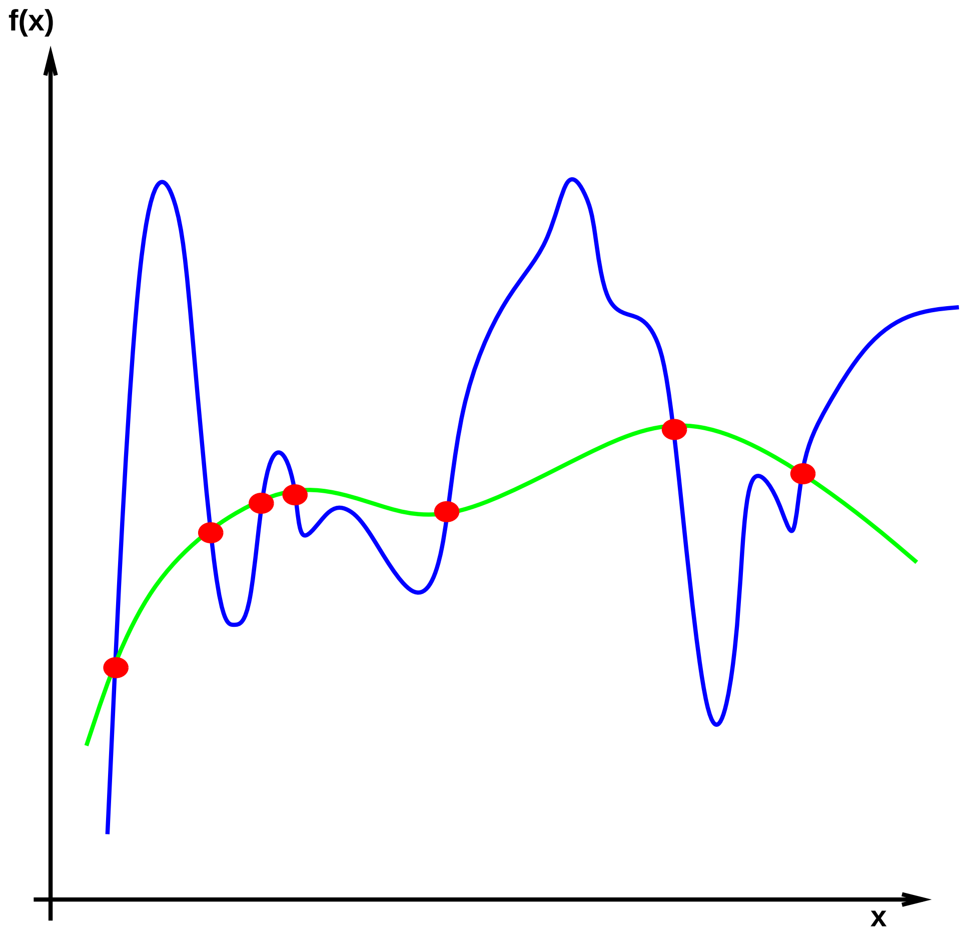
here is one (say, the true) solution...



... but here is another (and very different) one!



Both have zero empirical error: which one should we pick? Issue: stability (and uniqueness)



Well-posed and Ill-posed problems

Hadamard introduced the definition of ill-posedness. Ill-posed problems are typically inverse problems.

As an example, assume g is a function in Y and u is a function in X, with Y and X Hilbert spaces. Then given the linear, continuous operator L, consider the equation

$$g = Lu.$$

The direct problem is is to compute g given u; the inverse problem is to compute u given the data g. In the learning case L is somewhat similar to a "sampling" operation.

The inverse problem of finding u is well-posed when

- the solution exists,
- is unique and
- is *stable*, that is depends continuously on the initial data g.

Ill-posed problems fail to satisfy one or more of these criteria. Often the term ill-posed applies to problems that are **not stable**, which in a sense is the key condition.

Stability of learning

For the learning problem it is clear, but often neglected, that ERM is in general *ill-posed* for any given S. ERM defines a map L ("inverting" the "sampling" operation) which maps the discrete data S into a function f, that is

$$LS = f_S.$$

Consider the following simple, "classical" example.

Assume that the x part of the n examples $(x_1, ..., x_n)$ is fixed.

Then L as an operator on $(y_1, ..., y_n)$ can be defined in terms of a set of evaluation functionals F_i on $\mathcal{H}$, that is $y_i = F_i(u)$.

If $\mathcal{H}$ is a Hilbert space and in it the evaluation functionals F_i are *linear and bounded*, then $\mathcal{H}$ is a RKHS and the F_i can be written as $F_i(u) = (u, K_{x_i})_K$ where K is the kernel associated with the RKHS and we use the inner product in the RKHS.

For simplicity we assume that K is positive definite and sufficiently smooth (see Cucker, Smale).

Stability of ERM (example cont.)

The ERM case corresponds to

$$\min_{f \in \mathcal{B}_R} \frac{1}{n} \sum_{i=1}^{n} (f(\mathbf{x}_i) - y_i)^2.$$

Well-posedness can be ensured by Ivanov regularization that is by enforcing the solution f – which is has the form $f(\mathbf{x}) = \sum_{1=1}^{n} c_i K(\mathbf{x}_i, \mathbf{x})$ since it belongs to the RKHS – to be in the ball $\mathcal{B}_R$ of radius R in $\mathcal{H}$ (eg $\|f\|_K^2 \leq R$), because $\mathcal{H} = \overline{I_K(B_R)}$ – where $I_K : \mathcal{H}_K \to C(X)$ is the inclusion and $C(X)$ is the space of continuous functions with the sup norm – is compact.

In this case the minimizer of the generalization error $I[f]$ is well-posed.

Minimization of the empirical risk is also well-posed: it provides a set of linear equations to compute the coefficients $\mathbf{c}$ of the solution f as

$$K\mathbf{c} = \mathbf{y} \tag{1}$$

where $\mathbf{y} = (y_1, ..., y_n)$ and $(K)_{i,j} = K(\mathbf{x}_i, \mathbf{x}_j)$.

Stability of ERM (example cont.)

In this example, stability of the empirical risk minimizer provided by equation (1) can be characterized using the classical notion of *condition number* of the problem. The change in the solution f due to a perturbation in the data $\mathbf{y}$ can be bounded as

$$\frac{\|\Delta f\|}{\|f\|} \leq \|K\|\|(K)^{-1}\|\frac{\|\Delta \mathbf{y}\|}{\|\mathbf{y}\|}, \tag{2}$$

where $\|K\|(K)^{-1}\|$ is the condition number.

Stability of ERM (example cont.)

Tikhonov regularization – which unike Ivanov regularization is not ERM – replaces the previous equation with

$$\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (f(\mathbf{x}_i) - y_i)^2 + \gamma \|f\|_K^2 \tag{3}$$

which gives the following set of equations for $\mathbf{c}$ (with $\gamma \geq 0$)

$$(K + n\gamma I)\mathbf{c} = \mathbf{y} \tag{4}$$

which reduces for $\gamma = 0$ to equations (1). In this case, stability depends on the condition number $\|K + n\gamma I\| \|(K + n\gamma I)^{-1}\|$ which is now controlled by $n\gamma$. A large value of $n\gamma$ gives condition numbers close to 1.

In general, however, the operator L induced by ERM cannot be expected to be lineara and thus the definition of stability has to be extended beyond condition numbers...

Motivations for stability: inverse problems and beyond ERM

In summary there are two motivations for looking at stability of learning algorithms:

- can we generalize the concept of condition number to measure stability of L? Is stability related to generalization?

- through stability can one have a more general theory that provides *generalization* for general algorithms and *subsumes the classical theory* in the case of ERM?

5. Stability definitions

$S = z_1, ..., z_n$

$S^i = z_1, ..., z_{i-1}, z_{i+1}, ...z_n$

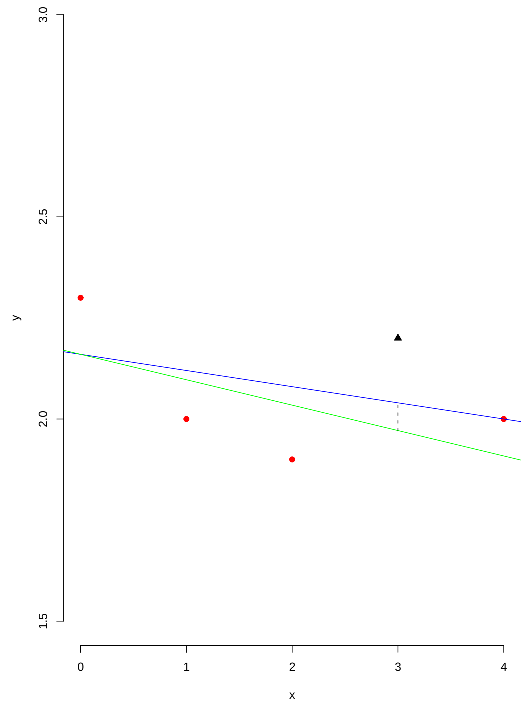
The learning map L has *distribution-independent,* CV_{loo} *stability* **if**

for each n *there exists a* $\beta_{CV}^{(n)}$ *and a* $\delta_{CV}^{(n)}$ *such that*

$$\forall \mu \quad \mathbb{P}_S \left\{ \left| V(f_{S^i}, z_i) - V(f_S, z_i) \right| \leq \beta_{CV}^{(n)} \right\} \geq 1 - \delta_{CV}^{(n)},$$

with $\beta_{CV}^{(n)}$ *and* $\delta_{CV}^{(n)}$ *going to zero for* $n \to \infty$*.*

CV_{loo} stability



The blue line was obtained by a linear regression (eg ERM with square loss on a hypothesis space of linear functions) on all five training points ($n = 5$). The green line was obtained in the same way by "leaving out" the black triangle from the training set. In this case, CV_{loo} stability requires that when a single point is removed from a data set, the change in error at the removed point (here indicated by the black dashed line) is small and decreases to zero in probability for n increasing to infinity.

Stability definitions (cont.)

Bousquet and Elisseeff's *uniform stability*:

the map L induced by a learning algorithm is *uniformly stable* **if**

$\lim_{n\to\infty} \beta^{(n)} = 0$ *with* $\beta^{(n)}$ *satisfying*

$$\forall S \in Z^n, \quad \forall i \in \{1, ..., n\} \qquad \sup_{z \in Z} |V(f_S, z) - V(f_{S^i}, z)| \leq \beta^{(n)}.$$

and $\beta^{(n)} = O(\frac{1}{n})$.

- Uniform stability implies good generalization.
- Tikhonov regularization algorithms are uniformly stable.
- Most algorithms are not uniformly stable: ERM, even with a hypothesis space $\mathcal{H}$ containing just two functions, is not guaranteed to be uniformly stable.
- Uniform stability implies CV$_{loo}$ stability.

Stability definitions (cont.)

- The learning map L has *distribution-independent,* E_{loo} *stability* **if**

  for each n *there exists a* $\beta_{Er}^{(n)}$ *and a* $\delta_{Er}^{(n)}$ *such that for all* $i = 1...n$

  $$\forall \mu \quad \mathbb{P}_S \left\{ |I[f_{S^i}] - I[f_S]| \leq \beta_{Er}^{(n)} \right\} \geq 1 - \delta_{Er}^{(n)},$$

  with $\beta_{Er}^{(n)}$ *and* $\delta_{Er}^{(n)}$ *going to zero for* $n \to \infty$*.*

- The learning map L has *distribution-independent,* EE_{loo} *stability* **if**

  for each n *there exists a* $\beta_{EE}^{(n)}$ *and a* $\delta_{EE}^{(n)}$ *such that for all* $i = 1...n$

  $$\forall \mu \quad \mathbb{P}_S \left\{ |I_{S^i}[f_{S^i}] - I_S[f_S]| \leq \beta_{EE}^{(n)} \right\} \geq 1 - \delta_{EE}^{(n)},$$

  with $\beta_{EE}^{(n)}$ *and* $\delta_{EE}^{(n)}$ *going to zero for* $n \to \infty$*.*

- *The learning map* L *is CVEEE*$_{loo}$ *stable* **if** *it has CV*$_{loo}$*, E*$_{loo}$ *and EE*$_{loo}$ *stability.*

Preview

Two theorems:

- (a) says that CVEEE_{loo} stability is sufficient to guarantee *generalization* of any algorithm

- (b) says that CVEEE_{loo} (and CV_{loo}) stability *subsumes* the "classical" conditions for generalization and consistency of ERM

Plan

1. Learning: well-posedness and predictivity
2. The supervised learning problem and generalization
3. ERM and conditions for generalization (and consistency)
4. Motivations for stability: inverse problems and beyond ERM
5. Stability definitions
6. **Theorem a: stability implies generalization**
7. Theorem b: ERM stability is necessary and sufficient for consistency
8. Stability of non-ERM algorithms
9. Open problems: hypothesis stability and expected error stability
10. On-line algorithms: stability and generalization?

6. Theorem a: stability implies generalization

Theorem (a)

If a learning map is CVEEE$_{loo}$ stable then with probability $1-\delta_{gen}$

$$|I[f_S] - I_S[f_S]| \leq \beta_{gen},$$

where

$$\delta_{gen} = \beta_{gen} = (2M\beta_{CV}+2M^2\delta_{CV}+3M\beta_{Er}+3M^2\delta_{Er}+5M\beta_{EE}+5M^2\delta_{EE})^{1/4}.$$

Thus CVEEE$_{loo}$ stability is strong enough to imply generalization of general algorithms. The question then is whether it is general enough to subsume the "classical" theory, that is the fundamental conditions for consistency of ERM.

7. Theorem b: ERM stability is necessary and sufficient for consistency

Theorem (b)

For "good" loss functions the following statements are equivalent for almost ERM:

1. *the learning algorithm is distribution independent* $CVEEE_{loo}$ *stable.*

2. *almost ERM is universally consistent*

3. $\mathcal{H}$ *is uGC.*

Theorem b, proof sketch: ERM stability is necessary and sufficient for consistency

First, ERM is E_{loo} and EE_{loo} stable, as it can be seen rather directly from its definition.

For CV_{loo} stability, here is a sketch of the proof in the special case of exact minimization of I_S and of I.

1. The first fact used in the proof is that CV_{loo} stability is equivalent to

$$\lim_{n\to\infty} \mathbb{E}_S[|V(f_{S^i}, z_i) - V(f_S, z_i)|] = 0.$$

The equivalence holds since the definition of CV_{loo} stability implies the condition on the expectation, since V is bounded; the opposite direction is obtained using Markov's inequality.

Theorem b: ERM stability is necessary and sufficient for consistency (cont.)

2. The following *positivity* property of exact ERM is the second and key fact used in proving the theorem:

$$\forall i \in \{1, ..., n\} \quad V(f_{S^i}, z_i) - V(f_S, z_i) \geq 0.$$

By the definition of empirical minimization we have

$$\begin{aligned} I_S[f_{S^i}] - I_S[f_S] &\geq 0 \\ I_{S^i}[f_{S^i}] - I_{S^i}[f_S] &\leq 0. \end{aligned}$$

Note that the first inequality can be rewritten as

$$\left[\frac{1}{n}\sum_{z_j \in S^i} V(f_{S^i}, z_j) - \frac{1}{n}\sum_{z_j \in S^i} V(f_S, z_j)\right] + \frac{1}{n}V(f_{S^i}, z_i) - \frac{1}{n}V(f_S, z_i) \geq 0.$$

The term in the bracket is non-positive (because of the second inequality) and thus the positivity property follows.

Theorem b: ERM stability is necessary and sufficient for consistency (cont.)

3. The third fact used in the proof is that – *for ERM* – distribution independent convergence of the expectation of empirical error to the expectation of the expected error of the empirical minimizer is equivalent to (universal) consistency.

The first two properties imply the following equivalences:

$$\begin{aligned}
(\beta,\delta)\ \text{CV}_{loo}\ \text{stability} \quad &\Leftrightarrow \quad \lim_{n\to\infty} \mathbb{E}_S[|V(f_{S^i},z_i) - V(f_S,z_i)|] = 0, \\
&\Leftrightarrow \quad \lim_{n\to\infty} \mathbb{E}_S[V(f_{S^i},z_i) - V(f_S,z_i)] = 0, \\
&\Leftrightarrow \quad \lim_{n\to\infty} \mathbb{E}_S I[f_{S^i}] - \mathbb{E}_S I_S[f_S] = 0, \\
&\Leftrightarrow \quad \lim_{n\to\infty} \mathbb{E}_S I[f_S] = \lim_{n\to\infty} \mathbb{E}_S I_S[f_S].
\end{aligned}$$

Notice that a weaker form of stability (eg CV_{loo} stability without the absolute value) is necessary and sufficient for consistency of ERM.

The third property implies that CV_{loo} stability is necessary and sufficient for the distribution independent convergence $I[f_S] \to I[f^*]$ *in probability* (where f^* is the best function in $\mathcal{H}$), that is for (universal) consistency. It is well known that the uGC property of $\mathcal{H}$ is necessary and sufficient for universal consistency of ERM.

8. Stability of non-ERM algorithms

Potential projects here...

- Regularization and SVMs are CVEEE_{loo} stable

- Bagging (with number of regressors increasing with n)is CVEEE_{loo} stable

- kNN (with k increasing with n)is CVEEE_{loo} stable

- Adaboost??

Plan

1. Learning: well-posedness and predictivity
2. The supervised learning problem and generalization
3. ERM and conditions for generalization (and consistency)
4. Motivations for stability: inverse problems and beyond ERM
5. Stability definitions
6. Theorem a: stability implies generalization
7. Theorem b: ERM stability is necessary and sufficient for consistency
8. **Stability of non-ERM algorithms**
9. Open problems: hypothesis stability and expected error stability
10. On-line algorithms: stability and generalization?

9. Open problems: other sufficient conditions.

CVEEE$_{loo}$ stability answers all the requirements we need: each one is sufficient for generalization in the general setting and subsumes the classical theory for ERM, since it is equivalent to consistency of ERM. It is quite possible, however, that CVEEE$_{loo}$ stability may be equivalent to other, even "simpler" conditions. In particular, we know that other conditions are sufficient for generalizations:

The learning map L is *Eloo$_{err}$ stable* in a distribution-independent way, **if** *for each n there exists a $\beta_{EL}^{(n)}$ and a $\delta_{EL}^{(n)}$ such that*

$$\forall \mu \quad \mathbb{P}_S \left\{ \left| I[f_S] - \frac{1}{n} \sum_{i=1}^{n} V(f_{S^i}, z_i) \right| \leq \beta_{EL} \right\} \geq 1 - \delta_{EL}^{(n)},$$

with $\beta_{EL}^{(n)}$ and $\delta_{EL}^{(n)}$ going to zero for $n \to \infty$.

Theorem: CV$_{loo}$ and Eloo$_{err}$ stability together imply generalization.

Open problems: expected error stability and hypothesis stability.

We conjecture that

- CV_{loo} and EE_{loo} stability are sufficient for generalization for general algorithms (without E_{loo} stability);

- alternatively, it may be possible to combine CV_{loo} stability with a "strong" condition such as hypothesis stability. We know that hypothesis stability together with CV_{loo} stability implies generalization ; we do not know whether or not ERM on a uGC class implies hypothesis stability, though we conjecture that it does.

The learning map L has distribution-independent, leave-one-out *hypothesis stability* **if** *for each n there exists a $\beta_H^{(n)}$*

$$\forall \mu \ \mathbb{E}_S \mathbb{E}_z[|V(f_S, z) - V(f_{S^i}, z)|] \leq \beta_H^{(n)},$$

with $\beta_H^{(n)}$ going to zero for $n \to \infty$.

Notice that Eloo_{err} property is implied – in the general setting – by *hypothesis stability*.

Plan

1. Learning: well-posedness and predictivity
2. The supervised learning problem and generalization
3. ERM and conditions for generalization (and consistency)
4. Motivations for stability: inverse problems and beyond ERM
5. Stability definitions
6. Theorem a: stability implies generalization
7. Theorem b: ERM stability is necessary and sufficient for consistency
8. Stability of non-ERM algorithms
9. Open problems: hypothesis stability and expected error stability
10. **On-line algorithms: stability and generalization?**

10. Open Problems around Stability

- Is it possible to simplify the definition of CVEEE_{loo} stability? In particular, is EE_{loo} stability needed, or is CV_{loo} and E_{loo} enough for generalization?

- Does ERM on a uGC class implies hypothesis stability?

- Relation between stability conditions and bounds based on Radamacher averages

- Online learning algorithms and stability (stochastic gradient descent under some specific assumptions is consistent)

- Conditions about predictivity of online algorithms: implications about synaptic plasticity rules