Learning Theory, Regularization and Model Selection

Carlos Valle

Departamento de Informática
Universidad Técnica Federico Santa María

cvalle@inf.utfsm.cl

November 5, 2015

Overview

1. Bias-variance tradeoff
2. PAC
3. Error bounds
4. VC dimension
5. Cross validation
6. Feature selection
7. Stability and generalization

Bias-variance tradeoff

- A main challenge is to construct a learning algorithm able to extrapolate.
- That is, to estimate future examples based on the observed phenomenon in the training set. (generalization error)
- This key property of an algorithm is known as the generalization ability.
- On the other hand, we find the algorithms that memorize the training samples but have poor predictive performance with unknown examples, this undesirable problem is well-known as overfitting.
- If we compute a different model for each training sample that we have collected; the bias is the set of points that cannot be well predicted by the model, and the variance measures how different the predictions along the training samples are.

Bias-variance tradeoff (2)

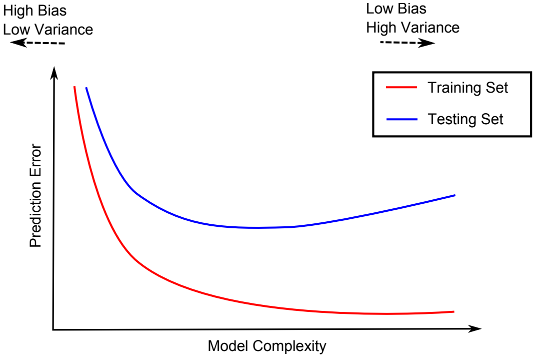
- Mathematically,

$$\begin{aligned}
E[(y-f(\mathbf{x}))^2] &= E[((y-E[f(\mathbf{x})])+(E[f(\mathbf{x})]-f(\mathbf{x})))^2] \\
&= E[(y-E[f(\mathbf{x})])]^2 + E[2(y-E[f(\mathbf{x})])(E[f(\mathbf{x})]-f(\mathbf{x}))] \\
&\quad + E[(E[f(\mathbf{x})]-f(\mathbf{x}))^2] \\
&= E[(y-E[f(\mathbf{x})])]^2 + E[(E[f(\mathbf{x})]-f(\mathbf{x}))^2] \\
\text{MSE} &= \text{bias}^2(f)+\text{var}(f)
\end{aligned}$$

- From the machine learning point of view, this trade-off is strongly related with the complexity of the learner f.
- A learner with low complexity has high bias covering the training points, which can lead to underfitting.

Bias-variance tradeoff (3)

- While, if the complexity of the learner is too high, the prediction tends to be closer (lower bias) to the training data and consequently generates overfitting.



Theoretical Bounds

- Let's consider some theoretical results:

Lemma (Union bound)

Let $A_1, A_2, \ldots, A_k$ be k different events (that may not be independent). Then

$$P\left(A_1 \cup \cdots \cup A_k\right) \leq P(A_1) + \cdots + P(A_k)$$

Markov inequality

Theorem (Markov inequality)

If U is a non-negative random variable on $\mathbb{R}$, then $\forall t > 0$:

$$P(U \geq t) \leq \frac{1}{t} E[U]$$

Proof.

$$\begin{aligned} E[U] &= \int_0^t u f(u) du + \int_t^\infty u f(u) du \\ &\geq \int_t^\infty u f(u) du \\ &\geq \int_t^\infty t f(u) du \\ &= t P(U \geq t) \end{aligned}$$

□

Chebyshev inequality

Corollary (Chebyshev inequality)

If Z is a random variable on $\mathbf{R}$ with mean μ and variance σ^2 then

$$P(|Z-\mu| \geq \sigma t) \leq \frac{1}{t^2}$$

Proof.

$$P(|Z-\mu| \geq \sigma t) = P((Z-\mu)^2 \geq \sigma^2 t^2)$$

Since $(\cdot)^2$ is an increasing monotonic function and $|Z-\mu|^2 = (Z-\mu)^2$. The, applying the Markov inequality we have

$$\begin{aligned} P((Z-\mu)^2 \geq \sigma^2 t^2) &\leq \frac{1}{\sigma^2 t^2} E[(Z-\mu)^2] \\ &= \frac{\sigma^2}{\sigma^2 t^2} = \frac{1}{t^2} \end{aligned}$$

□

Chernoff bounding

Corollary (Chernoff bounding)

Let Z be a random variable on $\mathbb{R}$. Then for all $t > 0$

$$P(z \geq t) \leq \inf_{s>0} e^{-st} M_Z(s)$$

where $M_Z = E[e^{sZ}]$ is the moment-generating function of Z.

Proof.

For any $s > 0$ we can use Markov's inequality to obtain:

$$P(Z \geq t) = P(sZ \geq st) = P(e^{sZ} \geq e^{st}) \leq e^{-st} E[e^{sZ}]$$

Since $s > 0$ was arbitrary the corollary follows. □

Hoeffding inequality

Lemma (Hoeffding inequality)

Let $Z_1, Z_2, \ldots, Z_M$ *be* M *independent and identically distributed (iid) random variables drawn from a Bernoulli*(p) *distribution. Let* $\hat{p} = \frac{1}{M}\sum_{m=1}^{M} Z_m$ *be the mean of these random variables, and let any* $\gamma > 0$ *be fixed. Then*

$$P(|p - \hat{p}| > \gamma) \leq 2e^{-2\gamma^2 M}$$

- It states if we take $\hat{p}$ the average of M Bernoulli(p) random variables to estimate p. Then, the probability of being close to p is large when M is large.

Preliminary definitions

- Given a training set $S_M = \{(\mathbf{x}_m, y_m)\}, m = 1 \ldots M$, where the training examples $(\mathbf{x}_m, y_m)$ are drawn iid from some unknown probability distribution $\mathcal{D}$.
- We have a realizable case where $S_M = \{(\mathbf{x}_m, c_t(\mathbf{x}_m))\}, m = 1 \ldots M$ and $\mathcal{D}$ is defined over $\mathcal{X}$ (since y_m are given by $c_t(\mathbf{x}_m)$)
- And the unrealizable case where $S_M = \{(\mathbf{x}_m, y_m)\}, m = 1 \ldots M$ and $\mathcal{D}$ is defined over $\mathcal{X} \times \mathcal{Y}$.

Generalization error

- We define the generalization error to be

$$\varepsilon(f) = P_{(\mathbf{x},y)\sim\mathcal{D}}(\ell(y_m, f(\mathbf{x}_m)))$$

- In particular, for binary classification problem ℓ is the misclassification function

$$\varepsilon(f) = P_{(\mathbf{x},y)\sim\mathcal{D}}(y_m \neq f(\mathbf{x}_m))$$

- Hence, the probability to misclassify a new example $(\mathbf{x}, y)$ from the distribution $\mathcal{D}$.
- In the realizable case

$$\varepsilon(f) = \int_{\mathbf{x}\in\mathcal{X}} I(c_t(\mathbf{x}_m) \neq f(\mathbf{x}_m))d\mathcal{D}(\mathbf{x}) = \int_{\mathbf{x}\in\mathcal{X}} I(c_t(\mathbf{x}_m)) \neq f(\mathbf{x}_m))\mathcal{D}(\mathbf{x})d\mathbf{x}$$

- Note that $\varepsilon(c_t(\mathbf{x}_m)) = 0$.
- In the unrealizable case

$$\varepsilon(f) = \int_{\mathbf{x}\in\mathcal{X}} I(y_m \neq f(\mathbf{x}_m))dP(\mathbf{x}, y)$$

PAC learnable

- An algorithm $\mathcal{A}$ learns a concept family $\mathcal{C}$ in the formal sense (PAC learnable) if for any $c_t \in \mathcal{C}$ and for every distribution $\mathcal{D}$ on the instance space $\mathcal{X}$, the algorithm $\mathcal{A}$ generates efficiently a concept function $f \in \mathcal{C}$ such that

$$P(\varepsilon(f) \leq \gamma) \leq 1 - \delta.$$

- Note that we would like the accuracy performance to hold under all training examples.
- In the unrealizable case, there maybe no function $f \in \mathcal{C}$ for which $\varepsilon(f) = 0$.
- For this case, we define the best a learning can achieve:

$$\mathsf{Opt}(C) = \min_{f \in \mathcal{C}} \varepsilon(f)$$

- Then, a learner seeks a hypothesis $f \in \mathcal{C}$ such that

$$P(\varepsilon(f) \leq \mathsf{Opt}(C) + \gamma) \geq 1 - \delta$$

PAC learnable (2)

Definition

There is an integer $M_0(\gamma, \delta)$ such that if $M \geq M_0$ then, for any probability distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$. If S^M is a training set of length M from $\mathcal{S}$ then, with probability at least $1 - \delta$ the hypothesis $f = \mathcal{A}(S^M) \in \mathcal{C}$ produced by $\mathcal{A}$ is such that $\varepsilon(f) \leq \text{Opt}(C) + \gamma$ We say that C is learnable or PAC learnable if there is a learning algorithm for $\mathcal{C}$

- Decreasing γ and δ values require a larger sample size.
- M_0 does not depend on the distribution $\mathcal{D}$ (because we are assuming that this distribution does not change).

Training error

- For a learning hypothesis f and a loss function ℓ. We define the training error, or empirical risk, or empirical error to be

$$\hat{\varepsilon}(f) = \frac{1}{M}\sum_{m=1}^{M} \ell(y_m, f(\mathbf{x}_m)).$$

- In particular, for binary classification problem ℓ is the misclassification function $I(y_m \neq f(\mathbf{x}_m))$.
- Here, the empirical error results the fraction of training examples that f misclassifies.
- Consider an hypothesis $f_\theta(x)$ which is parametrized by θ. We would like to find $\hat{\theta}$ which minimizes

$$\hat{\theta} = \text{argmin}_\theta \hat{\varepsilon}(f_\theta).$$

- This is called empirical risk minimization (ERM).

Generalization error

- The resulting hypothesis is $f_{\hat{\theta}}$
- Let $\mathcal{H}$ be the hypothesis space defined as the set of all hypotheses that might possibly be returned by it.
- For example, for linear classification

$$\mathcal{H} = \{f_\beta : f_\beta(\mathbf{x}) = I(\beta^T \mathbf{x} \geq 0), \beta \in \mathbb{R}^{I+1}\}$$

- Now, we can see the ERM process as a minimization over the hypothesis space $\mathcal{H}$:

$$\hat{f} = \text{argmin}_{f \in \mathcal{H}} \hat{\varepsilon}(f)$$

The case of finite $\mathcal{H}$

- Consider a learning problem in which the number of possible hypotheses is finite.
- Let $\mathcal{H} = \{f_1, \ldots, f_K\}$. Hence, we have K functions mapping from $\mathcal{X}$ to $\{0, 1\}$.
- ERM selects $\hat{f}$ with the smallest training error.
- Select one fixed $f_i \in \mathcal{H}$. And set $Z_m = I(f_i(\mathbf{x}_m \neq y_m))$.
- Each Z indicates whether f_i misclassifies an example $(\mathbf{x}_m, y_m)$ from $\mathcal{D}$.
- Thus, the training error can be expressed

$$\hat{\varepsilon}(f_i) = \frac{1}{M} \sum_{m=1}^{M} Z_m.$$

The case of finite $\mathcal{H}$ (2)

- Note that $\hat{\varepsilon}(f_k)$ is exactly the mean of the M random variables Z_m that are drawn iid from a Bernoulli distribution with mean $\varepsilon(f_k)$.
- Applying the Hoeffding inequality we obtain

$$P(|\varepsilon(f_k) - \hat{\varepsilon}(f_k)| > \gamma) \leq 2e^{-2\gamma^2 M}$$

- This shows for a hypothesis f_k that training error will be close to generalization error with high probability
- Using the union bound we can extend this result to all possible hypothesis:

$$\begin{aligned} P(\exists f \in \mathcal{H}, |\varepsilon(f_k) - \hat{\varepsilon}(f_k)| > \gamma) &= P\left(\bigcup_{k=1}^{K} |\varepsilon(f_k) - \hat{\varepsilon}(f_k)| > \gamma\right) \\ &\leq \sum_{k=1}^{K} P\left(|\varepsilon(f_k) - \hat{\varepsilon}(f_k)| > \gamma\right) \\ &\leq \sum_{k=1}^{K} 2e^{-2\gamma^2 M} \\ &= 2Ke^{-2\gamma^2 M} \end{aligned}$$

The case of finite $\mathcal{H}$ (3)

- Using the complement we have

$$\begin{aligned} P(\neg\exists f \in \mathcal{H}, |\varepsilon(f_i) - \hat{\varepsilon}(f_i)| > \gamma) &= P(\forall f \in \mathcal{H} |\varepsilon(f_i) - \hat{\varepsilon}(f_i)| \leq \gamma) \\ &\geq 1 - 2Ke^{-2\gamma^2 M} \end{aligned}$$

- This is called uniform convergence because this is a bound that holds for all $f \in \mathcal{H}$.
- With this bound, we can obtain how large must be the training set to guarantee with probability at least $1 - \delta$:

$$\begin{aligned} 1 - \delta &\leq 1 - 2Ke^{-2\gamma^2 M} \\ \delta &\geq 2Ke^{-2\gamma^2 M} \\ \frac{\delta}{2K} &\geq e^{-2\gamma^2 M} \\ \log\left(\frac{\delta}{2K}\right) &\geq -2\gamma^2 M \\ \frac{1}{2\gamma^2}\log\left(\frac{2K}{\delta}\right) &\leq M \end{aligned}$$

The case of finite $\mathcal{H}$ (4)

- This bound is called sample complexity
- This states how many training example we need to guarantee certain probability.
- Also, this say that the number of examples to train an algorithm increases logarithmic in k.
- It can also be proved that the bound for the PAC model is

$$M \geq \frac{1}{\gamma} \log \frac{|C|}{\delta}$$

- Analogously, we can solve for γ (fixing M and δ and we obtain that for all $f \in \mathcal{H}$

$$|\varepsilon(f_k) - \hat{\varepsilon}(f_k)| \leq \sqrt{\frac{1}{2M} \log\left(\frac{2K}{\delta}\right)}$$

- Assuming that this bound holds for every f_k

The case of finite $\mathcal{H}$ (5)

- Recall that $\hat{f}$ is the hypothesis obtained by the ERM process.
- Let $f^* = \text{argmin}_{f\in\mathcal{H}}\varepsilon(f)$ be the best hypothesis in $\mathcal{H}$.
- Now we have

$$\begin{aligned}\varepsilon(\hat{f}) &\leq \hat{\varepsilon}(\hat{f}) + \gamma \\ &\leq \hat{\varepsilon}(f^*) + \gamma \\ &\leq \varepsilon(f^*) + 2\gamma\end{aligned}$$

- This states that if uniform convergence ocurrs, the generalization error of $\hat{f}$ is at most 2γ worse that the best possible hypothesis that we could select.

The case of finite $\mathcal{H}$ (6)

- Using the last results, we have

Theorem

Let $|\mathcal{H}| = K$ and let any M, δ be fixed. Then with probability at least $1-\delta$, we have that

$$\varepsilon(\hat{f}) \leq \left(\min_{f \in \mathcal{H}} \varepsilon(f) \right) + 2\sqrt{\frac{1}{2M} \log\left(\frac{2K}{\delta}\right)}$$

- This theorem is in concordance with the bias-variance tradeoff:
- If we select the hypothesis from a larger hypothesis space $\mathcal{H}' \supseteq \mathcal{H}$, then we can only decrease the first term of the theorem (bias).
- However, when K increases, the second term ("variance") also increases when we use an algorithm with a more complex hypothesis space.

The case of finite $\mathcal{H}$ (7)

- Using the last results, we have

Corollary

Let $|\mathcal{H}| = K$ *and let any* γ, δ *be fixed. Then for* $\varepsilon(\hat{f}) \leq \min_{f \in \mathcal{H}} \varepsilon(f) + 2\gamma$ *to hold with probability at least* $1 - \delta$, *it suffices that*

$$\begin{aligned} m &\geq \frac{1}{2\gamma^2} \log \frac{2K}{\delta} \\ &= O\left(\frac{1}{\gamma^2} \log \frac{K}{\delta}\right) \end{aligned}$$

The case of infinite $\mathcal{H}$

- As we saw in the previous chapters, the most of the hypothesis spaces include real valued parameters.
- Suppose we have a hypothesis space $\mathcal{H}$ parametrized by real numbers.
- Computationally, in C each real number uses 64 bits to represent a floating point number.
- Thus, $K = 2^{64d}$. From the last corollary we have

$$M \geq O\left(\frac{1}{\gamma^2}\log 2^{64d}\delta\right) = O\left(\frac{d}{\gamma^2}\log 1\delta\right) = O_{\gamma,\delta}(d)$$

- Is at most linear in the number of parameters of the model.

The case of infinite $\mathcal{H}$ (2)

- However we can reparametrize any hypothesis by using more parameters, for example
- The space of the linear classifiers can be written

$$f_\beta = I(\beta_0 + \beta_1 x^{(1)} + \cdots + \beta_I x^{(I)} \geq 0)$$

- Or, it could be rearranged

$$f_{u,v} = I((u_0^2 - v_0^2) + (u_1^2 - v_1^2)x^{(1)} + \cdots + (u_I^2 - v_I^2)x^{(I)} \geq 0)$$

- Therefore, the number of parameter is not reliable.

The case of infinite $\mathcal{H}$ (3)

- Let's define a quantity that measures the capacity of a learning algorithm to correctly classify a set of examples.
- First, we will introduce the following two definitions:

Definition (Dichotomy)

A dichotomy of a set S is a partition of S into two disjoint subsets.

Definition

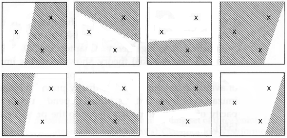
A a set of instances S is shattered by hypothesis space H if and only if for every dichotomy of S there exist some hypothesis $f \in \mathcal{H}$ consistent with this dichotomy.

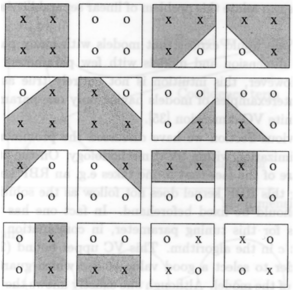
VC dimension

- Now we can define the VC dimension:

Definition (VC dimension)

Given a hypothesis space $\mathcal{H}$ the Vapnik-Chervonenkis dimension (VC($\mathcal{H}$)) to be the size of the largest set that is shattered by $\mathcal{H}$. If $\mathcal{H}$ can shatter arbitrarily large sets, then VC($\mathcal{H}$) $= \infty$.





VC dimension of a hyperplane

- In $\mathbf{R}^2$ we only can shattered 3 points.
- In $\mathbf{R}^I$ we only can shattered $I + 1$ points.
- This is implied from Radon's theorem:

Theorem

Any set $S \subset \mathbf{R}^I$ with $|S| \geq I + 2$ can be partitioned into 2 subsets A and B such that convex$(A) \cap$ convex$(B) \neq \emptyset$

- Radon's theorem immediately implies that half-spaces in I dimensions do not shatter any set of $I + 2$ points.

VC dimension of a hyperplane (2)

- Let's divide the set of $I + 2$ points into sets A and B.
- According to Radon's theorem convex$(A) \cap$ convex$(B) \neq \emptyset$
- Now suppose that some half space separates A from B.
- Then the half space contains A and the complement of the half space contains B.
- Hence, convex$(A) \cap$ convex$(B) \neq \emptyset$ which contradicts Radons Theorem.
- Therefore, no set of $I + 2$ points can be shattered by half planes in I dimensions.

Bounds for VC dimension

Theorem

Let $\mathcal{H}$ be given, and let $d = VC(\mathcal{H})$. Then, with probability at least $1-\delta$, we have that for all $f \in \mathcal{H}$,

$$|\varepsilon(f) - \hat{\varepsilon}(f)| \leq \sqrt{\frac{d}{M}\log\frac{M}{d} + \frac{1}{M}\log\frac{1}{\delta}}$$

- Thus, with probability at least $1-\delta$ we also have that

$$\hat{\varepsilon}(f) \leq \varepsilon(f^*) + \sqrt{\frac{d}{M}\log\frac{M}{d} + \frac{1}{M}\log\frac{1}{\delta}}$$

- This means that if a hypothesis space has finite VC dimension, then uniform convergence ocurrs when M is large.

Bounds for VC dimension (2)

Corollary

For $|\varepsilon(f) - \hat{\varepsilon}(f)| \leq \gamma$ to hold for all $f \in \mathcal{H}$ (and hence $\hat{\varepsilon}(f) \leq \varepsilon(f^*) + 2\gamma$) with probability at least $1 - \delta$ it suffices that $M = O_{\gamma,\delta}(d)$

- In other words, the number of training examples needed to generalize with probability $1 - \delta$ using a hypothesis from $\mathcal{H}$ is linear in the VC dimension of $\mathcal{H}$.
- An hypothesis f from a hypothesis space with higher VC dimension may fall into overfitting.
- Also, if the model assumes a reasonable parametrization, this is roughly linear in the number of parameters of $\mathcal{H}$.

Theorem

Concept class $\mathcal{C}$ with VC($\mathcal{C}$) $= \infty$ is not learnable in the formal case.

Critics to VC dimension

- The first drawback is that the VC dimension must actually be determined (or at least bounded) for the class of interest, and this is often not easy to do. (However, bounds on the VC dimension d have been computed for many natural decision function classes, including parametric classes involving standard arithmetic and boolean operations. See Anthony and Bartlett [1999] for a review of these results.)
- The second (more serious) drawback is that the analysis ignores the structure of the mapping from training samples to hypotheses, and concentrates solely on the range of the learners possible outputs. Ignoring the details of the learning map can omit many of the factors that are crucial for determining the success of the learning algorithm in real situations. [...]
- Smola et al. Advances in Large Margin Classifiers

How to select a model in practice?

- We have seen that some models have additional parameters.
- α for gradient descent, τ for locally weighted regression, C or ν for SVM.
- How to automatically choose these parameters?
- We will assume we have a finite set of models $\mathcal{M} = \{M_1, \ldots, M_d\}$
- In case that the parameter(s) is (are) continuous we can discretize it (them).
- As we stated before, choosing the parameters which minimizes the training error does not guarantee generalization.

Hold-out cross validation

- One option is to do the following:

Algorithm 1 Hold-out cross validation

```
1: Randomly split $S$ into $S_{\text{train}}$ (For instance 75% of the data) and $S_{cv}$ (the remaining 25%). Where $S_{cv}$ is the hold-out cross validation set.
2: Train each model $M_i$ on $S_{\text{train}}$ only to get the hypothesis $f_i$.
3: Select and output the hypothesis $f_i$ with the smallest error $\hat{\varepsilon}_{S_{CV}}(f_i)$ on the hold out cross validation set.
```

- Thus, for each f_i we obtain a better estimation of the generalization error.
- And we can select the hypothesis with the minimum estimated generalization error.
- Usually the size of the hold out cross validation set is set between $1/4$ and $1/3$.
- We are losing data that we are not using for training. And we estimate the generalization error only from one cross validation set.

K-fold cross validation

Algorithm 2 K-fold cross validation

```
Randomly split $S$ into $K$ disjoint subsets of $M/K$ training examples each. Let's call these subsets $S_1, \ldots, S_K$.
for each model $M_i$ do
    for $j = 1, \ldots, K$ do
        $f_{ij} \leftarrow \mathcal{A}_i(S \setminus S_j)$ (train on all data except $S_j$ )
        $\hat{\varepsilon}_{S_j}(f_{ij}) = \frac{1}{|S_j|} \sum_{(\mathbf{x}_m, y_m) \in S_j} \ell(y_m, f_{ij}(\mathbf{x}_m))$ (Compute the error over the validation test)
    end for
    $\hat{\varepsilon}_{M_i} = \frac{1}{k} \sum_{k=1}^{K} \hat{\varepsilon}_{S_k}(f_{ik})$ (Calculate the average of the validation error over the $k$ folds)
end for
Pick the model $M_i$ with the lowest $\hat{\varepsilon}_{M_i}$
$f \leftarrow \mathcal{A}_i(S)$ (train a hypothesis with all training data according to the model $M_i$).
Output: $f$
```

- $\mathcal{A}_i$ is the learning algorithm according to the model M_i

K-fold cross validation (2)

- Usually $K = 5$ or $K = 10$.
- When $K = M$ this method is called leave-one-out cross validation.
- In classification problems, sometimes the proportion of examples of each class are unequal. In this case, we can adapt cross validation in such a way that each fold has the same proportions. Thus, all folds will be equally unbalanced.
- This method is called stratified K-fold cross validation.

Feature selection

- One underlying problem when we try to create a map between the inputs and the outputs is to know if the inputs features are relevant for the model.
- If we use an hyperplane as a decision boundary, the VC dimension is $I + 1$.
- Thus, the more variables we use the more training data we need in order to assure good generalization.
- Moreover, in some supervised problems the number of features I is very large $(I \gg M)$.
- In these cases the VC dimension of out hyperplanes would be $O(I)$
- Thus, we would need $M = O(I)$ to avoid overfitting.

Feature selection (2)

- We can apply feature selection to reduce the number of features.
- Note that we can get 2^I different feature subsets.
- We need an heuristic search procedure to find a good feature subset.

Algorithm 3 Forward Selection

```
Initialize $\mathcal{F} = \emptyset$.
repeat
    for $i = 1, \dots, I$ do
        if $i \notin \mathcal{F}$ then
            $\mathcal{F}_i = \mathcal{F} \cup \{i\}$.
            Use cross-validation to evaluate features $\mathcal{F}_i$.
        end if
    end for
    Set $\mathcal{F} \leftarrow \min_i \hat{\varepsilon}_{CV}(\mathcal{F}_i)$
until Convergence criterion
Output: $\mathcal{F}$
```

Wrapper methods

- Convergence criterion can be either an error threshold or a $|\mathcal{F}|$ threshold.
- This is called a wrapper model because it is a procedure that wraps around the learning algorithm, making different calls to the learning algorithm to evaluate how well it performs with different feature subsets.
- Backward selection: Similar to forward selection, but in this case one starts with $\mathcal{F} = \bigcup_i^I \mathcal{F}_i$. Then, repeatedly deletes features one at a time.
- These are expensive methods $O(I^2)$.

Filter feature selection

- We can use an heuristic to make a cheaper feature selection.
- We can compute a simple score $S(i)$ that measures how informative each feature $X^{(i)}$ is about the targets y.
- Then, we choose the K features with the largest scores $S(i)$.
- It seems natural to choose the features that are most correlated with the targets.
- Mutual information is one of the most popular scores:

$$\mathsf{MI}(\mathbf{x}^{(i)}, Y) = \sum_{y \in \mathcal{Y}} \sum_{\mathbf{x}^{(i)} \in \mathcal{X}} p(\mathbf{x}^{(i)}, y) \log \left(\frac{p(\mathbf{x}^{(i)}, y)}{p(\mathbf{x}^{(i)})p(y)} \right)$$

- Here, $\mathcal{Y}$ is the set of values that Y takes. and $\mathcal{X}$ is the set of values that $\mathbf{x}^{(i)}$ takes.

Filter feature selection

- $p(\mathbf{x}^{(i)}, y)$, $p(\mathbf{x})$, $p(y)$ can all be estimated according using the training set.
- The mutual information can also be expressed as a Kullback-Leibler (KL) divergence:

$$\mathsf{MI}(\mathbf{x}^{(i)}, y) = \mathsf{KL}(p(\mathbf{x}^{(i)}, y)||p(\mathbf{x}^{(i)})p(y))$$

- This measure how different the probability distributions $p(\mathbf{x}^{(i)}, y)$ and $p(\mathbf{x}^{(i)})p(y)$ are.
- If $\mathbf{x}^{(i)}$ and y are independent, then $p(\mathbf{x}^{(i)}, y) = p(\mathbf{x}^{(i)})p(y)$. Thus, the KL-divergence between the two distributions will be zero (non informative).

Filter feature selection (2)

- Conversely, if $\mathbf{x}^{(i)}$ is very informative about y then MI$(\mathbf{x}^{(i)}, y)$ would be large.
- These measures came from Information Theory. For more details about this discipline please refer to "Cover & Thomas. Elements of the Information Theory".

Regularization

- According Jacques Hadamard a problem is well-posed if it meets the following properties:
  1. A solution exists
  2. The solution is unique
  3. The solution's behavior changes continuously with the initial conditions.
- On the contrary we say that the problem is ill-posed.
- The problem to approximate the underlying probability distribution by choosing the hypothesis that minimizes the empirical risk is ill-posed because we can find because it is possible to construct infinite functions from taking the training examples with zero training error.
- Thus, minimizing the empirical risk may lead to bad generalization performance.

Regularization (2)

- The key idea in regularization is to restrict the hipothesis space $\mathcal{H}$ (where $f \in \mathcal{H}$) of the empirical risk functional $\hat{\varepsilon}(f)$ such that $\mathcal{H}$ becomes a compact set.
- This technique was introduced by Tikhonov and Arsenin for solving inverse problems.
- In machine learning this techniques have been applied with great success.
- In statistics, the corresponding estimators are often referred to as shrinkage estimators.
- We will assume that $\hat{\varepsilon}(f)$ is continuous in f.
- Note that binary valued loss functions do not meet this requirement, but we can use margin-based loss functions.

Regularization (3)

Lemma (Operator Inversion)

Let X be a compact set and let the map $f : X \to Y$ be continuous. Then there exists an inverse map $f^{-1} : f(X) \to X$ that is also continuous.

- We have that for a compact $\mathcal{H}$, the inverse map from the minimum of the empirical risk functional $\varepsilon(f) : \mathcal{H} \to \mathbb{R}$ to its minimizer $\hat{f}$ is continuous and the optimization problem well-posed.
- Instead of specifying a hypothesis space $\mathcal{H}$ we add a stabilization (regularization) term $\Omega(f)$ to the original objective function:

$$\varepsilon_{reg}(f) = \hat{\varepsilon}(f) + \Omega(f)$$

- $\alpha > 0$ is the so-called regularization parameter which specifies the tradeoff between minimization of $\varepsilon(f)$ and the smoothness or simplicity which is enforced by small $\Omega(f)$.

Regularization (4)

- Usually one chooses $\Omega(f)$ to be convex, since this ensures that there exists only one global minimum.
- As we saw SVM minimizes

$$\min_{\mathbf{w},b} \frac{\lambda}{2}||\mathbf{w}||^2 + \sum_{m=1}^{M} (1 - y_m f(\mathbf{x}_m))_+ .$$

- While in classification and regression problems, we can penalize large values of β.
- For instance, ridge regression uses

$$J(\beta) = \frac{1}{2} \sum_{m=1}^{M} (f(\mathbf{x}_m) - y_m)^2 + \lambda||\beta||^2,$$

- Whereas logistic regression selects the hypothesis maximizing

$$\text{argmax}_{\beta} \log \sum_{m=1}^{M} P(y_m|x_m;\beta) - \lambda||\beta||^2.$$

Stability and Generalization

- For the most of the real cases, $|\mathcal{H}|$ is infinite.
- In this scenario the VC dimension gives us a thick generalization bound.
- We would like to know how the generalization of an learning algorithm is perturbed by small changes to its inputs.
- This is known as algorithmic stability.
- A stable learning algorithm is one for which the prediction does not change much when the training data is modified slightly.
- We will briefly study some theoretical result from algorithmic stability.

Notations and definitions

- Let $S = \{\mathbf{z}_m\}_{m=1}^{M}$ be the training set, where $\mathbf{z}_m = (\mathbf{x}_m, y_m)$.
- $S^{\setminus m}$ is the training set obtained by removing point $(\mathbf{x}_m, y_m)$ from S.

$$S^{\setminus m} = \{\mathbf{z}_1, \ldots, \mathbf{z}_{m-1}, \mathbf{z}_{m+1}, \ldots, \mathbf{z}_M\}$$

- $S^{\mathbf{u},m}$ is the training set obtained by replacing point $(\mathbf{x}_m, y_m)$ from S into $\mathbf{u} \sim \mathcal{D}$.

$$S^{u,m} = \{\mathbf{z}_1, \ldots, \mathbf{z}_{m-1}, \mathbf{u}, \mathbf{z}_{m+1}, \ldots, \mathbf{z}_m\}$$

- We define $f_S \leftarrow \mathcal{A}(S)$, and we require that $0 \leq \ell(f, \mathbf{z}) \leq L$. For some constant L.
- The generalization error of a learning rate is defined

$$\text{gen}(S) = \varepsilon(f_S) - \hat{\varepsilon}(f_S)$$

Uniform hypothesis stability

Definition (Uniform hypothesis stability)

An algorithm $\mathcal{A}$ has **uniform hypothesis stability** β with respect to the loss function ℓ if the following holds

$$\forall S \in Z^M, \forall m, \forall \mathbf{u}, \mathbf{z} \in Z, |\ell(f_S, \mathbf{z}) - \ell(f_S^{\mathbf{u},m}), \mathbf{z})| \leq \beta$$

- We can view β as a function M. We are interested in the case where $\beta = \lambda/M$.
- Moreover, Bousquet and Elisseeff prove that regularization has uniform hypothesis stability $\mathcal{O}(1/M)$.
- They also obtain bounds on generalization error.

Uniform hypothesis stability (2)

Theorem (Bousquet and Elisseeff)

If $\mathcal{A}$ has uniform hypothesis stability β, then for all $\tau > 0$

$$P(|\textit{gen}(S)| > \tau + \beta) \leq 2\exp\left(\frac{-\tau^2 M}{2(M\beta + L)^2}\right)$$

- This theorem gives good bounds on generalization error when $\beta = \mathcal{O}(1/M)$.
- However, just a few algorithms are uniformly hypothesis stable.
- if $\mathcal{A}$ is a ± 1-algorithm ($\forall x \in \mathcal{X}, f_s \in \{-1, 1\}$). $\mathcal{A}$ would be β-hypothesis-stable for some $\beta < 1$ only if $\mathcal{A}$ is the constant algorithm, this is $\forall \mathbf{x} \in S, f_S(\mathbf{x}) = f$.

Uniform hypothesis stability (3)

- If $|\mathcal{H}|$ is finite. Let $\mathcal{A}$ be a learning algorithm which performs ERM. Let's assume that $\mathcal{A}$ is not constant, then $\mathcal{A}$ is not uniformly β-hypothesis stable for any $\beta = \mathcal{O}(1)$.
- To show this result, we can divide Z^M into regions: $\forall f \in \mathcal{H}, R(f) = \{S \in Z^M | f_s = f\}$. Thus, some training sets S must lie on the boundary between regions, so the stability β is at least

$$\min_{f,f' \in \mathcal{H}} \left\{ \sup_{\mathbf{z} \in Z} \{ |\ell(f, \mathbf{z}) - \ell(f', \mathbf{z})| \} \right\}$$

Weak hypothesis stability

Definition (Weak hypothesis stability)

An algorithm $\mathcal{A}$ has **weak hypothesis stability** (β, δ), if for any $m = 1..M$ with probability at least $1 - \delta$, we have that

$$\forall S \in Z^M, \forall \mathbf{u} \in Z, \max_{\mathbf{z} \in Z} |\ell(f_S, \mathbf{z}) - \ell(f_S^{\mathbf{u},m}), \mathbf{z})| \leq \beta$$

- We are interested in $\beta = \mathcal{O}(1/M)$
- However this definition is still restrictive.
- Kutin and Niyogi proved that ERM over a space of finite VC dimension is not necessarily weakly hipothesis stable.

Weak error stability

Definition (Weak error stability)

An algorithm $\mathcal{A}$ has **weak error stability** (β, δ), if for any $m = 1..M$ with probability at least $1 - \delta$, we have that

$$\forall S \in Z^M, \forall \mathbf{u} \in Z, |\varepsilon(f_S) - \varepsilon(f_S^{\mathbf{u},m})| \leq \beta$$

- It is clear for any S and S', $|\varepsilon(f_S) - \varepsilon(f_S^{\mathbf{u},m})| \leq L$
- This imply that if $\mathcal{A}$ is weakly (β, δ)-error stable then $\varepsilon(f_S)$ is weakly difference bounded by L, β and δ
- Weak error stability implies that $\varepsilon(f_S)$ is concentrated about its mean.
- However, weak error stability is not strong enough to imply good bounds on generalization error.

Cross-validation and overlap stability

Definition (Cross-validation stability)

An algorithm $\mathcal{A}$ has **cross-validation stability** (β, δ), if for any $m = 1..M$ with probability at least $1 - \delta$, we have that

$$\forall S \in Z^M, \forall \mathbf{u} \in Z, |\ell(f_S, \mathbf{u}) - \ell(f_S^{\mathbf{u},m}), \mathbf{u})| \leq \beta$$

Definition (Overlap stability)

An algorithm $\mathcal{A}$ has **overlap stability** (β, δ), if for any $m = 1..M$ with probability at least $1 - \delta$, we have that

$$\forall S \in Z^M, \forall \mathbf{u} \in Z, |\hat{\varepsilon}_{S^m}(f_S) - \hat{\varepsilon}_{S^m}(f_S^{\mathbf{u},m})| \leq \beta$$

- We are saying that the most training set S, S' differing in only one example, f_S and $f_{S'}$ have similar performance on $S^m = S \cap S'$

Training stability and generalization bounds

Definition (Training stability)

An algorithm $\mathcal{A}$ has **training stability** (β, δ), if $\mathcal{A}$ has CV stability $\beta, \delta)$ and $\mathcal{A}$ has overlap stability (β, δ).

- Note that weak hypothesis stability (β, δ) implies training stability (β, δ).

Training stability and generalization bounds

- Additional theoretical bounds are:

Definition

If $\mathcal{A}$ has CV stability (β, δ)

$$E_{S,\mathbf{z}\sim\mathcal{D}^{M+1}}\left[\left|\ell(f_S,\mathbf{z}) - \ell(f_S^{m,\mathbf{z}},\mathbf{z})\right|\right] \leq \beta + \delta L$$

Definition

If $\mathcal{A}$ has overlap stability (β, δ) then

$$\forall S \in Z^M, \forall \mathbf{u} \in Z, \max_{\mathbf{z}\in Z} |\ell(f_S,\mathbf{z}) - \ell(f_S^{\mathbf{u},m}),\mathbf{z})| \leq \frac{L}{M} + \frac{M-1}{M}\beta$$

Any questions?