KYOTO UNIVERSITY

Statistical Machine Learning Theory

(Introduction to) **Statistical Learning Theory**

Hisashi Kashima
kashima@i.Kyoto-u.ac.jp

DEPARTMENT OF INTELLIGENCE SCIENCE AND TECHNOLOGY

Statistical learning theory: Theoretical guarantee for learning from limited data

- What is the test performance of a classifier with a particular training performance?
- How far is a classifier from the best performance model?
- How many training instances are needed to ensure a certain accuracy of the estimate?

REFERENCE:
Bousquet, Boucheron, and Lugosi.
"Introduction to statistical learning theory."
Advanced lectures on machine learning. pp. 169-207, 2004.

Error Bounds

True risk and empirical risk: We are interested in true risk but can access only to empirical risk

- Training dataset $\{(x^{(1)}, y^{(1)}), \dots, (x^{(N)}, y^{(N)})\}$ is sampled from probability distribution P in an i.i.d manner
  - $y^{(i)} \in \{+1, -1\}$: Binary classification
  - We want to estimate $f\colon \mathcal{X} \to \{+1, -1\}$

Indicator function
(0 − 1 loss)

- (True) risk: $R(f) = \Pr(f(x) \neq y) = E_{(x,y)\sim P}\left[1_{f(x)\neq y}\right]$
  - We cannot directly evaluate this since we do not know P
- Empirical risk: $R_N(f) = \frac{1}{N}\sum_{i=1}^{N} 1_{f(x^{(i)})\neq y^{(i)}}$
  - Usually we estimate a classifier that minimizes this

Our goal: How good is the classifier learned by empirical risk minimization?

- Ultimate goal: find the best f in function class $\mathcal{F}$
  - Best function: $f^* = \mathrm{argmin}_{f \in \mathcal{F}} R(f)$ True risk
- Instead, empirical risk minimization: $f_N = \mathrm{argmin}_{f \in \mathcal{F}} R_N(f)$
  - With regularization: $f_N = \mathrm{argmin}_{f \in \mathcal{F}} R_N(f) + \lambda \|f\|^2$
- Our targets: We want to know how good f_N is

1. $R(f_N) - R_N(f_N) \leq B(N, \mathcal{F})$: Estimate of the true risk of a trained classifier from its empirical risk
2. $R(f_N) - R(f^*) \leq B(N, \mathcal{F})$: Estimate how far the true risk of a trained classifier is from the best one

Error bound:
We want to give an error bound for a *finite* dataset

- Let us consider to find a bound $R(f_N) - R_N(f_N) \leq B(N, \mathcal{F})$
  - We want a bound depending on N

- $R(f) - R_N(f) = E_{(x,y)\sim P}[1_{f(x)\neq y}] - \frac{1}{N}\sum_{i=1}^{N} 1_{f(x^{(i)})\neq y^{(i)}}$
  - By the law of large numbers, this will converge to 0
    - Empirical risk is a good estimate of the true risk
  - But we want to know $B(N, \mathcal{F})$ depending on a finite N

The bound is a function of N

Hoeffding's inequality: A tool to analyze difference of expectation and empirical mean for small sample

- Hoeffding's inequality: Let $Z^{(1)}, \dots, Z^{(N)}$ be N i.i.d. random variables with $Z^{(i)} \in [a, b]$. Then, for any $\epsilon > 0$,

$$\Pr\left[\ \left| E[Z] - \frac{1}{N}\sum_{i=1}^{N} Z^{(i)} \right| > \epsilon\ \right] \leq 2\exp\left(-\frac{2N\epsilon^2}{(b-a)^2}\right)$$

  - Gives the bound of probability of difference between expected value and empirical estimate exceeding ϵ
  - As N gets larger, the upper bound will get smaller and converge to zero
  - As ϵ gets smaller, the upper bound will get larger

Applying Hoeffding's inequality: Bound of true risk for a fixed classifier

- Now we apply the Hoeffding's inequality to our case:

$$\Pr\left[\ \left|E[Z] - \frac{1}{N}\sum_{i=1}^{N} Z^{(i)}\right| > \epsilon\ \right] \leq 2\exp\left(-\frac{2N\epsilon^2}{(b-a)^2}\right)$$

- For a classifier $f \in \mathcal{F}$, setting $Z = 1_{f(x)\neq y}$ gives

$$\Pr[\ |R(f) - R_N(f)| > \epsilon\ ] \leq 2\exp(-2N\epsilon^2) \equiv \delta$$

  - $(b-a)^2 \leq 1$

- With probability at least $1-\delta$,

$$R(f) - R_N(f) \leq \sqrt{\frac{\log\frac{2}{\delta}}{2N}}$$

A bad news: Simple application of Hoeffding's inequality does not give the error bound

- For *a fixed* classifier f, its true risk is estimated by Hoeffding's inequality
  - With a fixed f, we can draw a sample with the bounded error with high probability
- But, this is not the estimate of the true risk of the algorithm
  - For a fixed sample, there can be many classifiers in the pool that violate the error bound
  - We do not know which classifier the algorithm will be chosen before seeing the data
  - So, we want a bound which holds for *all* classifier $f \in \mathcal{F}$

Error bound:
Depends on the log number of possible classifiers

- Theorem: With probability at least $1-\delta, \forall f \in \mathcal{F}$

$$R(f) - R_N(f) \le \sqrt{\frac{\log|\mathcal{F}| + \log\frac{2}{\delta}}{2N}}$$

- This also implies: for $f_N = \mathrm{argmin}_{f\in\mathcal{F}} R_N(f)$,

$$R(f_N) - R_N(f_N) \le \sqrt{\frac{\log|\mathcal{F}| + \log\frac{2}{\delta}}{2N}}$$

- The bound depends on the log number of functions in $\mathcal{F}$
  - $|\mathcal{F}|$: The size of the hypothesis space

Slow increase

Error bound:
Proof using the union bound

- We apply the Hoeffding's inequality to all classifiers in $\mathcal{F}$ simultaneously
- Union bound:
  - For two events A_1, A_2, $\Pr[A_1 \cup A_2] \leq \Pr[A_1] + \Pr[A_2]$
  - For K events, $\Pr[A_1 \cup \cdots \cup A_K] \leq \sum_{i=1}^{K} \Pr[A_K]$
- Hoeffding + union bound gives:
  - $\Pr[\exists f \in \mathcal{F}: |R(f) - R_N(f)| > \epsilon] \leq 2|\mathcal{F}| \exp(-2N\epsilon^2)$
  - Equate the right hand side to δ to obtain the upper bound

Sample complexity: Number of examples required to ensure a certain accuracy

- Theorem: With probability at least $1-\delta$, $\forall f \in \mathcal{F}$

$$R(f) - R_N(f) \leq \sqrt{\frac{\log|\mathcal{F}| + \log\frac{2}{\delta}}{2N}}$$

- This theorem means, in other words, for any $\epsilon > 0$ if we take $N \geq \frac{\log|\mathcal{F}|+\log\frac{2}{\delta}}{2\epsilon^2}$ examples, with probability at least $1-\delta$, we have

$$R(f) - R_N(f) \leq \epsilon$$

Error bound against the optimal classifier: Similar bound holds

- We are also interested in how far the true risk of a trained classifier from the best one in $\mathcal{F}$
  - $R(f_N) - R(f^*) \leq B(N, \mathcal{F})$
- Similar analysis gives a bound depending on $\log|\mathcal{F}|$
- Theorem: With probability at least $1 - \delta$,

$$R(f_N) - R(f^*) \leq 2\sqrt{\frac{\log|\mathcal{F}| + \log\frac{2}{\delta}}{2N}}$$

Summary: How good is the classifier learned by empirical risk minimization?

- Empirical risk minimization: $f_N = \operatorname{argmin}_{f \in \mathcal{F}} R_N(f)$
- Unknown best function: $f^* = \operatorname{argmin}_{f \in \mathcal{F}} R(f)$ R: true risk
- We can know how good f_N is in two ways:

1. $R(f_N) \leq R_N(f_N) + \sqrt{\frac{\log|\mathcal{F}| + \log\frac{2}{\delta}}{2N}}$: Estimate of the true risk of a trained classifier from its empirical risk

2. $R(f_N) - R(f^*) \leq 2\sqrt{\frac{\log|\mathcal{F}| + \log\frac{2}{\delta}}{2N}}$: How far is the true risk of a trained classifier from the best one?

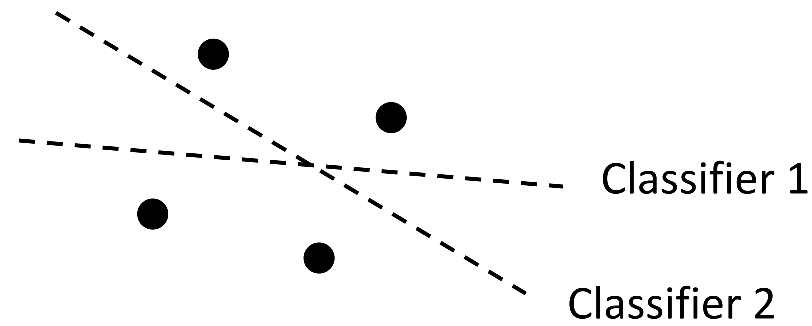
Infinite Case

Infinite case: Previous results assume finite number of classifiers

- We assumed the number of classifiers is finite
  - The bound depends on the number of classifiers in the class $\mathcal{F}: R(f_N) - R_N(f_N) \leq \sqrt{\frac{\log|\mathcal{F}| + \log\frac{2}{\delta}}{2N}}$
    - $\log|\mathcal{F}|$ is considered as the complexity of class $\mathcal{F}$
  - So far we measure the complexity of the model using the number of possible classifiers (= size of hypothesis space)
- What if it is infinite? (E.g. linear classifiers $f = \mathbf{w}^\top \mathbf{x}$)
  - The upper bound goes to infinity ☹
- Do we have another complexity measure for the infinite case?

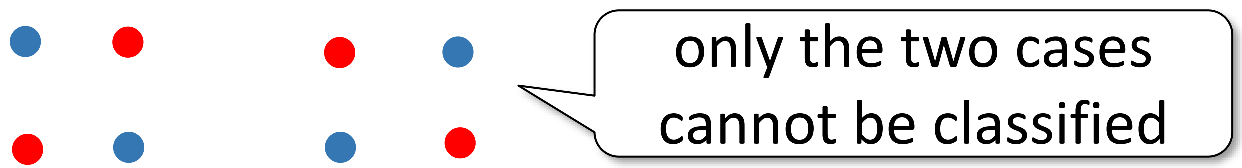
Growth function: Infinite number of functions can be grouped into finite number of function groups

- Use "growth function" as a complexity measure of infinite numbers of classifiers
- Idea: group the infinite number of classifiers into a finite number of equivalent sets
  - Two classifiers make same predictions for the 4 data points
  - They can be considered equivalent for the purpose of classifying the 4 data points



Growth function:
Error bound using growth function

- Growth function $\mathcal{S}_{\mathcal{F}}(N)$: The maximum number of ways into which N points can be classified by the function class $\mathcal{F}$
  - Apparently, $\mathcal{S}_{\mathcal{F}}(N) \leq 2^N$
  - For two-dimensional linear classifiers, $\mathcal{S}_{\mathcal{F}}(4) = 14 \leq 2^4$



- Theorem: With probability at least $1 - \delta$, $\forall f \in \mathcal{F}$

$$R(f) - R_N(f) \leq 2\sqrt{2\frac{\log \mathcal{S}_{\mathcal{F}}(2N) + \log \frac{2}{\delta}}{N}}$$

VC dimension: Intrinsic dimension of function class

- When $\mathcal{S}_{\mathcal{F}}(N) = 2^N$, any classification of N points is possible (we say that $\mathcal{F}$ shatters the set)

- VC dimension h of class $\mathcal{F}$:
  The largest N such that $\mathcal{S}_{\mathcal{F}}(N) = 2^N$

- For two-dimensional linear classifiers, $h = 3$

- Generally, for d-dimensional linear classifiers, $h = d + 1$

- Theorem: With probability at least $1 - \delta$, $\forall f \in \mathcal{F}$

$$R(f) - R_N(f) \leq 2\sqrt{2\frac{\boldsymbol{h}\log\frac{2eN}{\boldsymbol{h}} + \log\frac{2}{\delta}}{N}}$$

Statistical learning theory:
Theoretical guarantee for learning from limited data

- What is the test performance of a classifier with a particular training performance?
- How far is a classifier from the best performance model?
- How many training instances are needed to ensure a certain accuracy of the estimate?

REFERENCE:
Bousquet, Boucheron, and Lugosi.
"Introduction to statistical learning theory."
Advanced lectures on machine learning. pp. 169-207, 2004.