Lecture 4: Error measures and Noisy data

Presentation · March 2019

1 author:

Alaa Tharwat
Fachhochschule Bielefeld

Lecture 4: Error measures and Noisy data

Alaa Tharwat

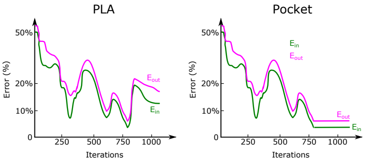
Lecture 4: Error measures and Noisy data

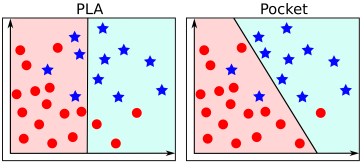
- Review of Lecture 3
- Multiple hypotheses: New terms
- Error measures
- Noisy Targets
- Learning theory

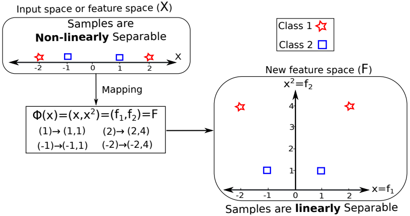
Lecture 4: Error measures and Noisy data

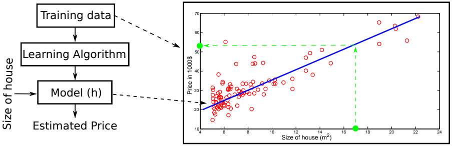
- Review of Lecture 3
- Multiple hypotheses: New terms
- Error measures
- Noisy Targets
- Learning theory

- Any sample in the dataset is represented by a set of features
- PLA vs. Pocket algorithm
- Classification ($h(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x})$) vs. regression ($h(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$)
- Linear Regression, $E_{in} = \frac{1}{N}\sum_i^N (h(\mathbf{x}_i) - y_i)^2$
- $\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \Rightarrow \mathbf{w} = \mathbf{X}\dagger\mathbf{y}$









Lecture 4: Error measures and Noisy data

- Review of Lecture 3
- Multiple hypotheses: New terms
- Error measures
- Noisy Targets
- Learning theory

$$P[|E_{in}(g) - E_{out}(g)| > \epsilon] \leq 2Me^{-2\epsilon^2 N}$$

It is similar to

$$P[|E_{in}(g) - E_{out}(g)| \leq \epsilon] \geq 1 - 2Me^{-2\epsilon^2 N}$$

- Let $\delta = 2Me^{-2\epsilon^2 N}$, this means that with a probability $1 - \delta$, the difference between E_{in} and E_{out} is at most ϵ ($\Rightarrow P[|E_{in}(g) - E_{out}(g)| \leq \epsilon] \geq 1 - \delta$)
- $E_{out} \leq E_{in} + \epsilon$
- $\epsilon = \sqrt{\frac{ln\frac{2M}{\delta}}{2N}}$
- $N = \frac{ln\frac{2M}{\delta}}{2\epsilon^2}$

$$E_{out} \leq E_{in} + \epsilon$$

$$E_{out} \leq E_{in} + \sqrt{\frac{ln\frac{2M}{\delta}}{2N}}$$

- ϵ parameter represents the maximum difference between E_{in} and E_{out}. It is inversely proportional with N and hence more training data are required for decreasing the difference between E_{in} and E_{out}; this is the reason why the accuracy is **expensive**.
  - For example, to increase the accuracy (reduce ϵ) ten times this means $\epsilon' = \epsilon/10$ and hence we need $100N$ samples

$$E_{out} \leq E_{in} + \epsilon$$

$$E_{out} \leq E_{in} + \sqrt{\frac{ln\frac{2M}{\delta}}{2N}}$$

- δ parameter is called **confidence parameter** and it defines the probability of failure
  - For example, given $\delta = 0.05$ and then the confidence is $1 - \delta = 0.95$

$$E_{out} \leq E_{in} + \epsilon$$

$$E_{out} \leq E_{in} + \sqrt{\frac{ln\frac{2M}{\delta}}{2N}}$$

- Increasing the size of hypothesis space (i.e. $M \rightarrow \infty$) increases the difference between E_{in} and E_{out}, even if the training error (E_{in}) is small

$$E_{out} \leq E_{in} + \epsilon$$

$$E_{out} \leq E_{in} + \sqrt{\frac{ln\frac{2M}{\delta}}{2N}}$$

- Increasing the number of samples decreases the gap between E_{in} and E_{out} (see the example in next slide)
- $N \gg lnM \Rightarrow E_{out}(g) \approx E_{in}(g)$, does not depend on X, $P(x)$, f, or how g is found
- Given a data which has 1000 samples and these samples are classified into two classes. First, we trained a classifier with all data and then used the trained model for testing the same data
- The whole dataset (1000 samples) represent the input space and hence the prediction results using all data represents E_{out}
- To test the influence of the number of samples, we calculate the prediction results of the same model but using 50, 250, 500, and 750 samples

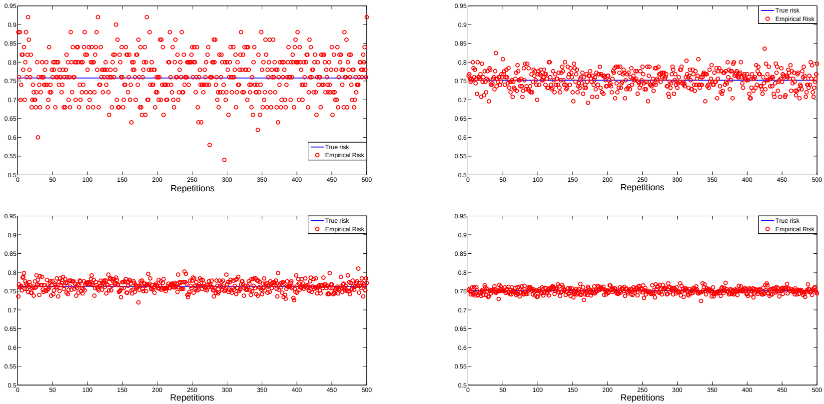
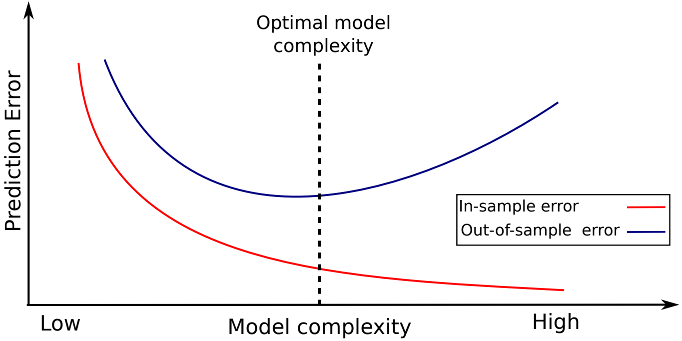


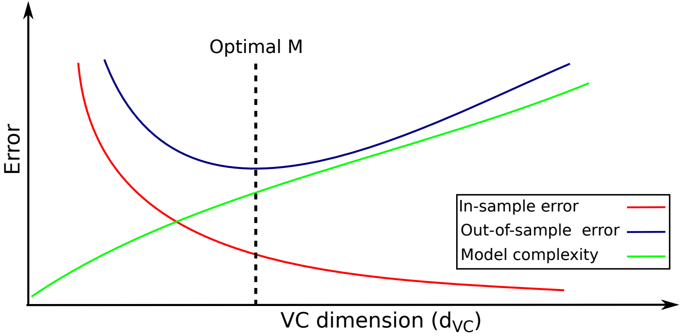
Figure: Visualization of the difference between the empirical risk and the risk using different numbers of training samples; (a) 50 samples, (b) 250 samples, (c) 500 samples and (d) 750 samples.

- In the learning theory, we need $g \approx f \Rightarrow E_{out}(g) \approx 0$. This can be achieved if
  1. Make sure that $E_{out}(g)$ is close to $E_{in}(g)$, $E_{out} \approx E_{in}$ (good generalization)
  2. Minimize $E_{in}(g)$, $E_{in}(g) \approx 0$



Any learning model generates infinite (M) hypotheses, we can say increasing M means increasing the model complexity

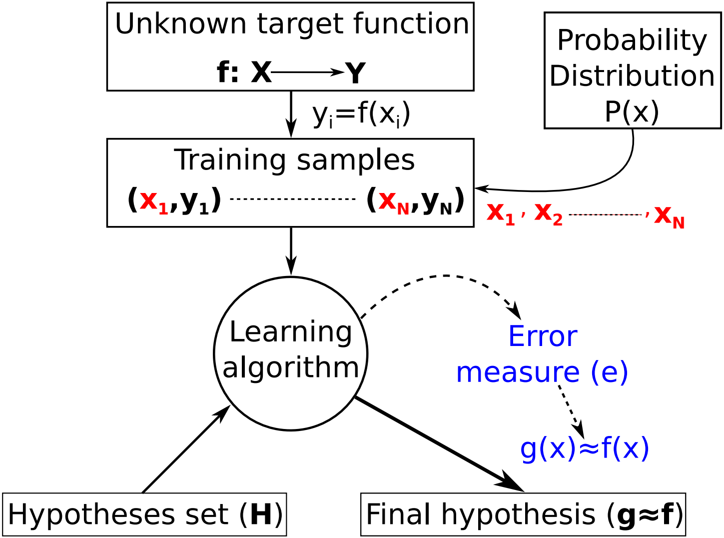
- Model complexity $\uparrow \Rightarrow E_{in} \downarrow$
- Model complexity $\uparrow \Rightarrow E_{out} - E_{in} \uparrow$



Lecture 4: Error measures and Noisy data

- Review of Lecture 3
- Multiple hypotheses: New terms
- Error measures
- Noisy Targets
- Learning theory

- Error measure or loss function is used for evaluating a hypothesis ($E(h, f)$)
- For example, *Squared error* $e(h(\mathbf{x}), f(\mathbf{x})) = (h(\mathbf{x}) - f(\mathbf{x}))^2$
- For example, *Binary error* $e(h(\mathbf{x}), f(\mathbf{x})) = h(\mathbf{x}) \neq f(\mathbf{x}))$
- Overall error (in-sample error) is $E_{in} = \frac{1}{N} \sum_{i=1}^{N} e(h(\mathbf{x}_i), f(\mathbf{x}_i))$
- Out-of-sample error is $E_{out} = E_x[e(h(\mathbf{x}_i), f(\mathbf{x}_i))]$
- E_{in} and E_{out} use the same cost function

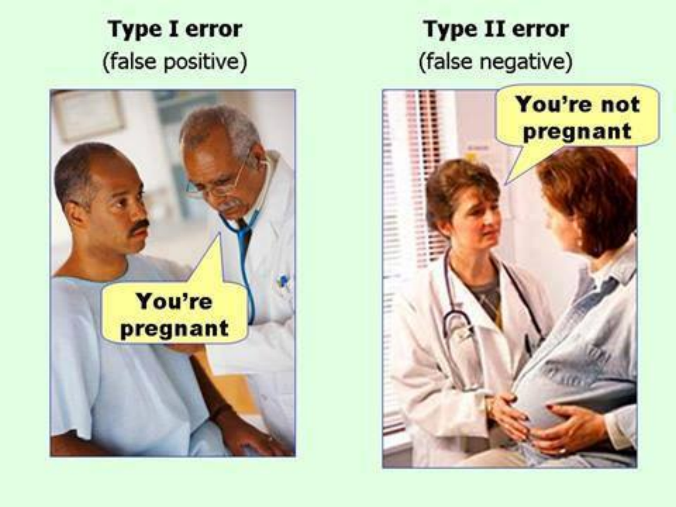

Unknown target function
f: X → Y
y_i=f(x_i)
Probability Distribution P(x)
Training samples
(x_1,y_1) .... (x_N,y_N)
x_1, x_2 ...., x_N
Learning algorithm
Error measure (e)
g(x)≈f(x)
Hypotheses set (H)
Final hypothesis (g≈f)

Types of errors

- From the confusion matrix below, the outputs are: True positive, True negative, false positive, and false negative
- The errors are not the same. For example, in a fingerprint verification
  - In a supermarket application, to get a discount, false rejection is costly than false acceptance
    - In false rejection $\rightarrow$ the customer gets annoyed, while, false acceptance $\rightarrow$ more customers get the discount
  - In security, false acceptance is costly than false rejection (why?)
    - False acceptance in security is a disaster because unauthorized person can access the system, but, with false rejection, the employee need to do more trails

True Class

		Positive (P)	Negative (N)
Predicted Class	True (T)	True Positive (TP)	False Positive (FP)
	False (F)	False Negative (FN)	True Negative (TN)
		P=TP+FN	N=FP+TN


Type I error
(false positive)
You're pregnant
Type II error
(false negative)
You're not pregnant

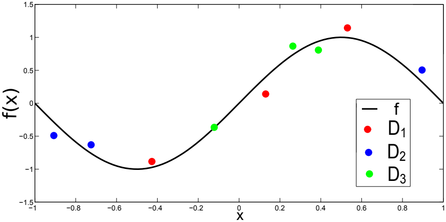
Lecture 4: Error measures and Noisy data

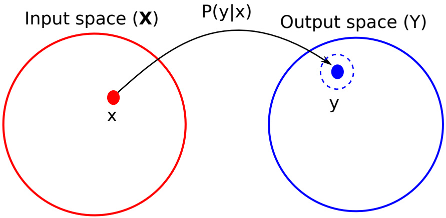
- Review of Lecture 3
- Multiple hypotheses: New terms
- Error measures
- Noisy Targets
- Learning theory

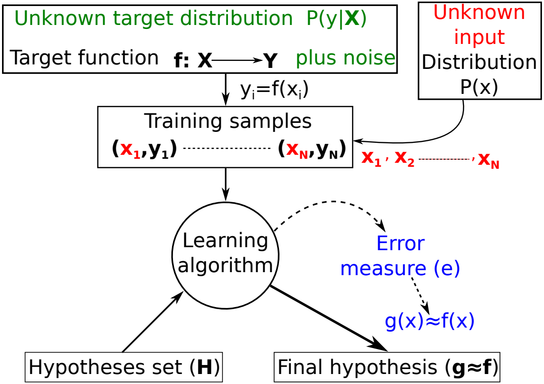
- The target function[1] is not always a function such case credit-approval, weather prediction, or face recognition problems
- For example, two identical customers may have two different behaviors (approved/denied)
- So, instead of using $y = f(\mathbf{x})$, we use target distribution $P(y|\mathbf{x})$
  - $\rightarrow (\mathbf{x}, y)$ is generated now using the joint distribution $P(\mathbf{x})P(y|\mathbf{x})$
- Noisy target $\equiv$ deterministic target $f(\mathbf{x}) = E(y|\mathbf{x})$ plus noise $y - f(\mathbf{x})$
- Hence, deterministic noise is a special case of noisy target, when the noise is zero

[1] Mathematically, a function returns a unique value for every point in the domain.

- $P(\mathbf{x})$ is the input distribution ($\mathbf{x}_i$ is drawn from $\mathbf{X}$ with a probability $P(\mathbf{x}_i)$)
- $P(y|\mathbf{x})$ is the target distribution, and this is what we trying to learn. y_i is observed with probability $P(y_i|\mathbf{x}_i)$ (i.e. $y \sim P(y|\mathbf{x})$); hence, y_i in some papers is called *desired output*
- $P(\mathbf{x}, y) = P(\mathbf{x})P(y|\mathbf{x})$ is the mix of the two concepts
- The observed response is probabilistic $\rightarrow$ the same input can generate different outputs






Unknown target distribution P(y|X)
Target function f: X → Y plus noise
Unknown input Distribution P(x)
y_i=f(x_i)
Training samples
(x_1,y_1) (x_N,y_N)
x_1 , x_2 , x_N
Learning algorithm
Error measure (e)
g(x)≈f(x)
Hypotheses set (H)
Final hypothesis (g≈f)