Training/Testing and Regularization

Il-Chul Moon
Dept. of Industrial and Systems Engineering
KAIST
icmoon@kaist.ac.kr

Weekly Objectives

- Understand the concept of bias and variance
  - Know the concept of over-fitting and under-fitting
  - Able to segment two sources, bias and variance, of error
- Understand the bias and variance trade-off
  - Understand the concept of Occam's razor
  - Able to perform cross-validation
  - Know various performance metrics for supervised machine learning
- Understand the concept of regularization
  - Know how to apply regularization to
    - Linear regression
    - Logistic regression
    - Support vector machine

CONCEPT OF BIAS AND VARIANCE

Up To This Point…

- Now, you are supposed to have some knowledge in classifications
  - Naïve Bayes
  - Logistic Regression
  - Support Vector Machine
- SVM is still a commonly used machine learning algorithm for classifications

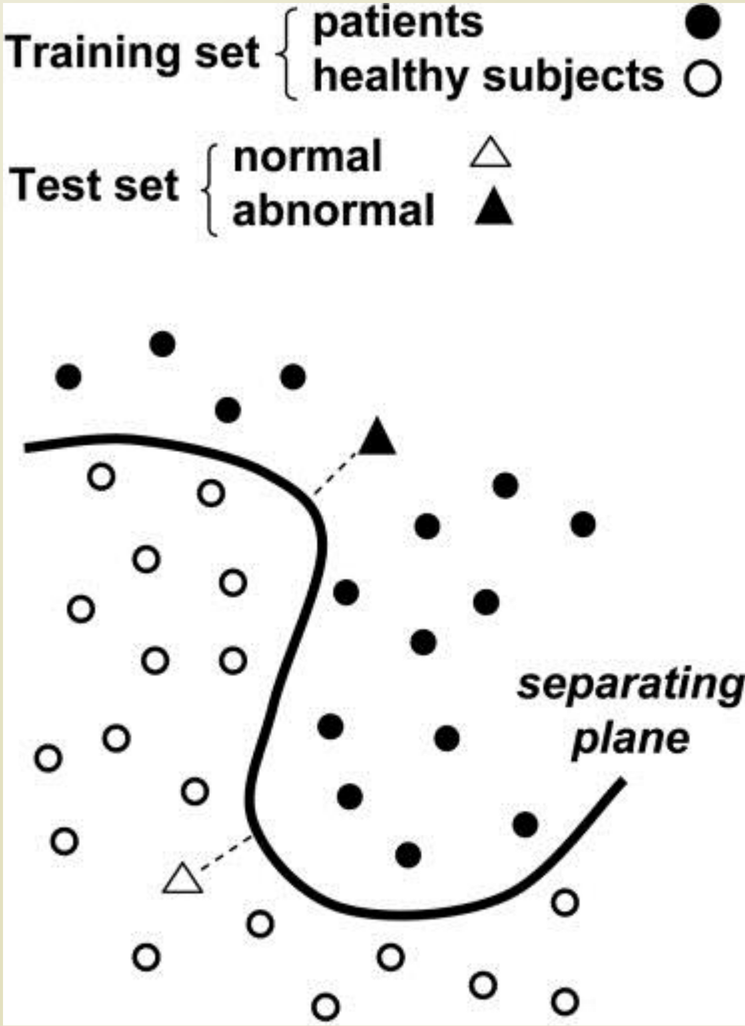
- Functioning is ***kind of*** done
- Efficiency and accuracy now becomes a problem

Better Machine Learning Approach?

- Accurate prediction result
  - Ex) with this NB classifier, I can filter spams with 95% accuracy!
- Is this a right claim?
  - The validity of accuracy
    - No clear definition
    - Why not use other performance metrics? Such as Precision/Recall, F-Measure
  - The validity of dataset
    - Spams??
    - How many spams?
    - Where did you gathered?
    - Big variance in the spams?
    - Is the spam mail evolving?
      - From Nigerian prince scheme to something else?

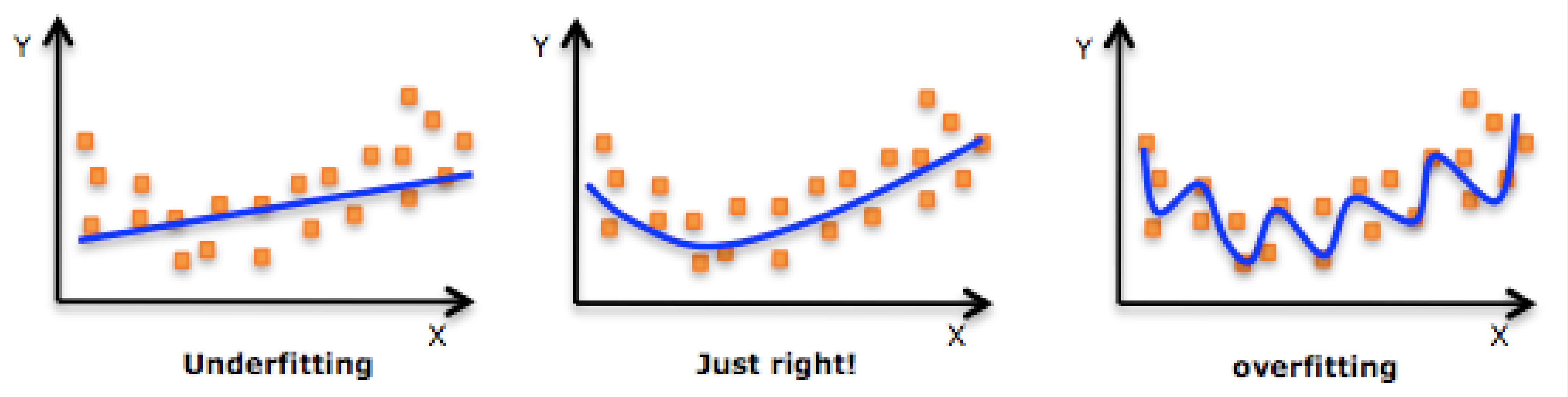
Training and Testing

- Training
  - Parameter inference procedure
  - Prior knowledge, past experience
  - There is no guarantee that this will work in the future
    - ML's Achilles gun is the stable/static distribution of learning targets.
  - Why ML does not work in the future?
    - The domain changes, or the current domain does not show enough variance
    - The ML algorithms inherently have problems
- Testing
  - Testing the learned ML algorithms/the inferred parameters
  - New dataset that is unrelated to the training process
  - Imitating the future instances
    - By setting aside a subset of observations



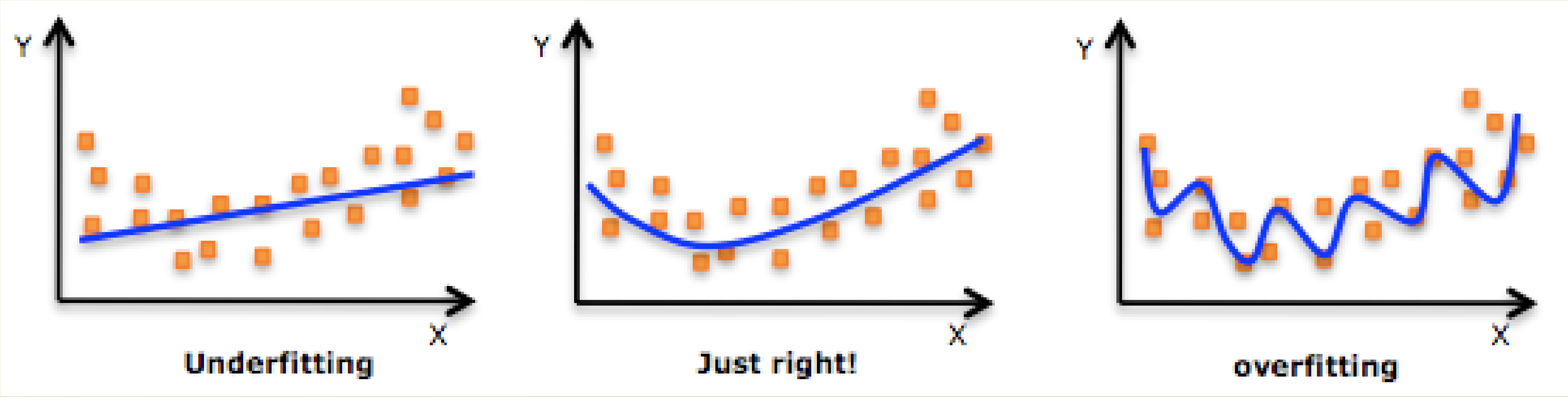
Over-Fitting and Under-Fitting

- Imaging this scenario
  - You are given N points to train a ML algorithm
  - You are going to learn a simple polynomial regression function
    - Y=F(x)
    - The degree of F is undetermined. Can be linear or non-linear
- Considering the three Fs in the below, which looks better?



Tuning Model Complexity

- One degree, two degree, and N degree trained functions
  - As the degree increases, the model becomes complex
  - Is complex model better?
- Then, where do we stop in developing a complex model?
  - Is there any measure to calculate the complexity and the generality?
- There is a trade-off between the complexity of a model and the generality of a dataset.



Sources of Error in ML

- Source of error is in two-folds
  - Approximation and generalization
- $E_{out} \leq E_{in} + \Omega$
  - E_{out} is the estimation error, considering a regression case, of a trained ML algorithm
  - E_{in} is the error from approximation by the learning algorithms
  - Ω is the error caused by the variance of the observations
- Here, we define a few more symbols
  - f: the target function to learn
  - g: the learning function of ML
  - $g^{(D)}$: the learned function by using a dataset, D, or an instance of hypothesis
  - D: an available dataset drawn from the real world
  - $\bar{g}$: the average hypothesis of a given infinite number of Ds
    - Formally, $\bar{g}(x) = E_D[g^{(D)}(x)]$

Bias and Variance

- $E_{out} \leq E_{in} + \Omega$
- Error of a single instance of a dataset D
  - $E_{out}(g^{(D)}(x)) = E_X[\left(g^{(D)}(x) - f(x)\right)^2]$
- Then, the expected error of the infinite number of datasets, D
  - $E_D[E_{out}\left(g^{(D)}(x)\right)] = E_D[E_X[\left(g^{(D)}(x) - f(x)\right)^2]] = E_X[E_D[\left(g^{(D)}(x) - f(x)\right)^2]]$
- Let's simplify the inside term, $E_D[\left(g^{(D)}(x) - f(x)\right)^2]$
  - $E_D\left[\left(g^{(D)}(x) - f(x)\right)^2\right] = E_D\left[\left(g^{(D)}(x) - \bar{g}(x) + \bar{g}(x) - f(x)\right)^2\right]$
  - $= E_D\left[(g^{(D)}(x) - \bar{g}(x))^2 + \left(\bar{g}(x) - f(x)\right)^2 + 2(g^{(D)}(x) - \bar{g}(x))(\bar{g}(x) - f(x))\right]$
  - $= E_D[(g^{(D)}(x) - \bar{g}(x))^2] + \left(\bar{g}(x) - f(x)\right)^2 + E_D[2(g^{(D)}(x) - \bar{g}(x))(\bar{g}(x) - f(x))]$
- $E_D[2(g^{(D)}(x) - \bar{g}(x))(\bar{g}(x) - f(x))] = 0$
  - Because of the definition of $\bar{g}(x)$
- Then, eventually the error becomes ….
  - $E_D[E_{out}\left(g^{(D)}(x)\right)] = E_X[E_D\left[\left(g^{(D)}(x) - \bar{g}(x)\right)^2\right] + \left(\bar{g}(x) - f(x)\right)^2]$

Bias and Variance Dilemma

- $E_D[E_{out}\left(g^{(D)}(x)\right)]=E_X[E_D\left[\left(g^{(D)}(x)-\bar{g}(x)\right)^2\right]+\left(\bar{g}(x)-f(x)\right)^2]$
- Let’s define
  - Variance(x)=$E_D\left[\left(g^{(D)}(x)-\bar{g}(x)\right)^2\right]$
  - Bias^2(X)=$\left(\bar{g}(x)-f(x)\right)^2$
- Semantically, what do they mean?
  - Variance is an inability to train a model to the average hypothesis because of the dataset limitation
  - Bias is an inability to train an average hypothesis to match the real world
- How to reduce the bias and the variance?
  - Reducing the variance
    - Collecting more data
  - Reducing the bias
    - More complex model
- However, if we reduce the bias, we increase the variance, and vice versa
  - Bias and Variance Dilemma
  - We will see why this is in the next slide by empirical evaluations….