CS420, Machine Learning, Lecture 10

Learning Theory and Model Selection

Weinan Zhang

Shanghai Jiao Tong University

http://wnzhang.net

http://wnzhang.net/teaching/cs420/index.html

Content

- Learning Theory
    - Bias-Variance Decomposition
    - Finite Hypothesis Space ERM Bound
    - Infinite Hypothesis Space ERM Bound
    - VC Dimension

- Model Selection
    - Cross Validation
    - Feature Selection
    - Occam’s Razor for Bayesian Model Selection

Learning Theory

- Theorems that characterize classes of learning problems or specific algorithms in terms of computational complexity or sample complexity
  - i.e. the number of training examples necessary or sufficient to learn hypotheses of a given accuracy

$$\underset{\substack{\uparrow\\ \text{Error}}}{\epsilon(d, N, \delta)} = \sqrt{\frac{1}{\underset{\substack{\uparrow\\ \text{\#. Training}\\ \text{samples}}}{2N}}\left(\log \underset{\substack{\uparrow\\ \text{Hypothesis}\\ \text{space}}}{d} + \log \frac{1}{\underset{\substack{\uparrow\\ \text{Probability}\\ \text{of}\\ \text{correctness}}}{\delta}}\right)}$$

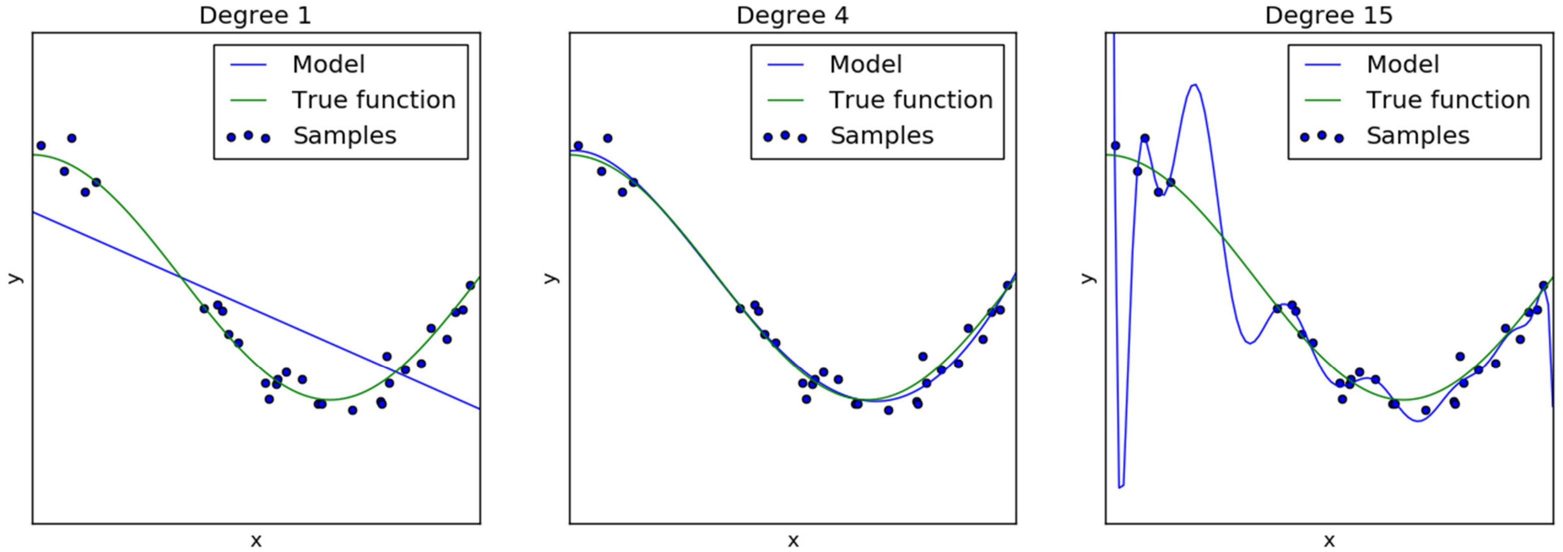
Learning Theory

- Complexity of a learning problem depends on:
  - Size or expressiveness of the hypothesis space
  - Accuracy to which target concept must be approximated
  - Probability with which the learner must produce a successful hypothesis
  - Manner in which training examples are presented, e.g. randomly or by query to an oracle

$$\underset{\substack{\uparrow\\ \text{Error}}}{\epsilon(d, N, \delta)} = \sqrt{\frac{1}{\underset{\substack{\uparrow\\ \text{\#. Training samples}}}{2N}}\left(\log \underset{\substack{\uparrow\\ \text{Hypothesis space}}}{d} + \log \frac{1}{\underset{\substack{\uparrow\\ \text{Probability of correctness}}}{\delta}}\right)}$$

Model Selection

- Which model is the best?



Linear model: underfitting

4th-order model: well fitting

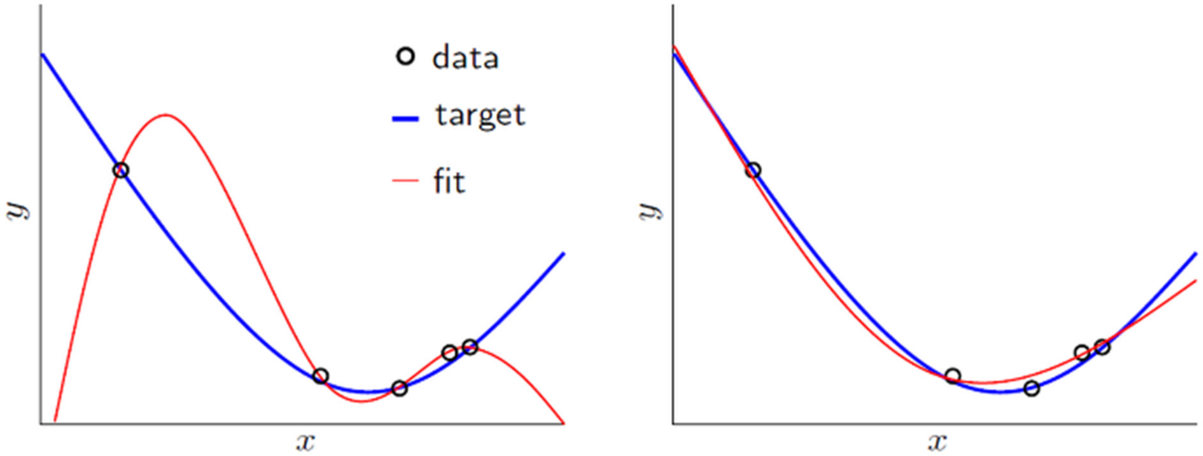
15th-order model: overfitting

- Underfitting occurs when a statistical model or machine learning algorithm cannot capture the underlying trend of the data.
- Overfitting occurs when a statistical model describes random error or noise instead of the underlying relationship

Regularization

- Add a penalty term of the parameters to prevent the model from overfitting the data

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y_i, f_\theta(x_i)) + \lambda \Omega(\theta)$$



(a) without regularization

(b) with regularization

Content

- Learning Theory
    - Bias-Variance Decomposition
    - Finite Hypothesis Space ERM Bound
    - Infinite Hypothesis Space ERM Bound
    - VC Dimension

- Model Selection
    - Cross Validation
    - Feature Selection
    - Occam’s Razor for Bayesian Model Selection

Bias Variance Decomposition

Bias-Variance Decomposition

- Bias-Variance Decomposition
  - Assume $Y = f(X) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$
  - Then the expected prediction error at an input point x_0

$$\begin{aligned}
\mathrm{Err}(x_0) &= \mathbb{E}[(Y - \hat{f}(X))^2 | X = x_0] \\
&= \mathbb{E}[(\epsilon + f(x_0) - \hat{f}(x_0))^2] \\
&= \mathbb{E}[\epsilon^2] + \underbrace{\mathbb{E}[2\epsilon(f(x_0) - \hat{f}(x_0))]}_{=0} + \mathbb{E}[(f(x_0) - \hat{f}(x_0))^2] \\
&= \sigma_\epsilon^2 + \mathbb{E}[(f(x_0) - \mathbb{E}[\hat{f}(x_0)] + \mathbb{E}[\hat{f}(x_0)] - \hat{f}(x_0))^2] \\
&= \sigma_\epsilon^2 + \mathbb{E}[(f(x_0) - \mathbb{E}[\hat{f}(x_0)])^2] + \mathbb{E}[(\mathbb{E}[\hat{f}(x_0)] - \hat{f}(x_0))^2] \\
&\qquad - 2\mathbb{E}[(f(x_0) - \mathbb{E}[\hat{f}(x_0)])(\mathbb{E}[\hat{f}(x_0)] - \hat{f}(x_0))] \\
&= \sigma_\epsilon^2 + \mathbb{E}[(f(x_0) - \mathbb{E}[\hat{f}(x_0)])^2] + \mathbb{E}[(\mathbb{E}[\hat{f}(x_0)] - \hat{f}(x_0))^2] \\
&\qquad - 2\underbrace{(f(x_0)\mathbb{E}[\hat{f}(x_0)] - f(x_0)\mathbb{E}[\hat{f}(x_0)] - \mathbb{E}[\hat{f}(x_0)]^2 + \mathbb{E}[\hat{f}(x_0)]^2)}_{=0} \\
&= \sigma_\epsilon^2 + (\mathbb{E}[\hat{f}(x_0)] - f(x_0))^2 + \mathbb{E}[(\hat{f}(x_0) - \mathbb{E}[\hat{f}(x_0)])^2] \\
&= \sigma_\epsilon^2 + \mathrm{Bias}^2(\hat{f}(x_0)) + \mathrm{Var}(\hat{f}(x_0))
\end{aligned}$$

Bias-Variance Decomposition

- Bias-Variance Decomposition
  - Assume $Y = f(X) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$
  - Then the expected prediction error at an input point *x*$_0$

$$\begin{aligned}\mathrm{Err}(x_0) &= \sigma_\epsilon^2 + (\mathbb{E}[\hat{f}(x_0)] - f(x_0))^2 + \mathbb{E}[(\hat{f}(x_0) - \mathbb{E}[\hat{f}(x_0)])^2] \\ &= \sigma_\epsilon^2 + \mathrm{Bias}^2(\hat{f}(x_0)) + \mathrm{Var}(\hat{f}(x_0))\end{aligned}$$

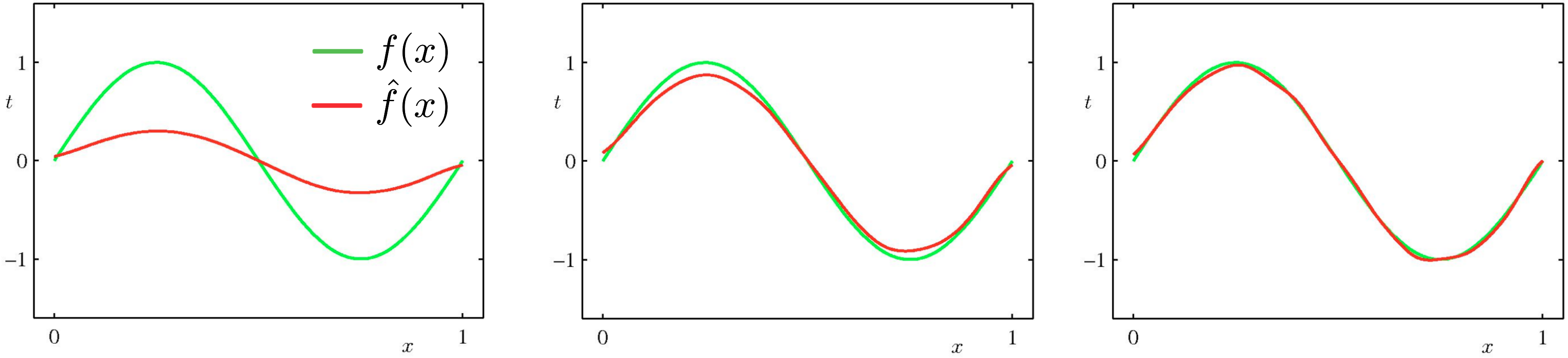
Observation noise (Irreducible error)

How far away the expected prediction is from the truth

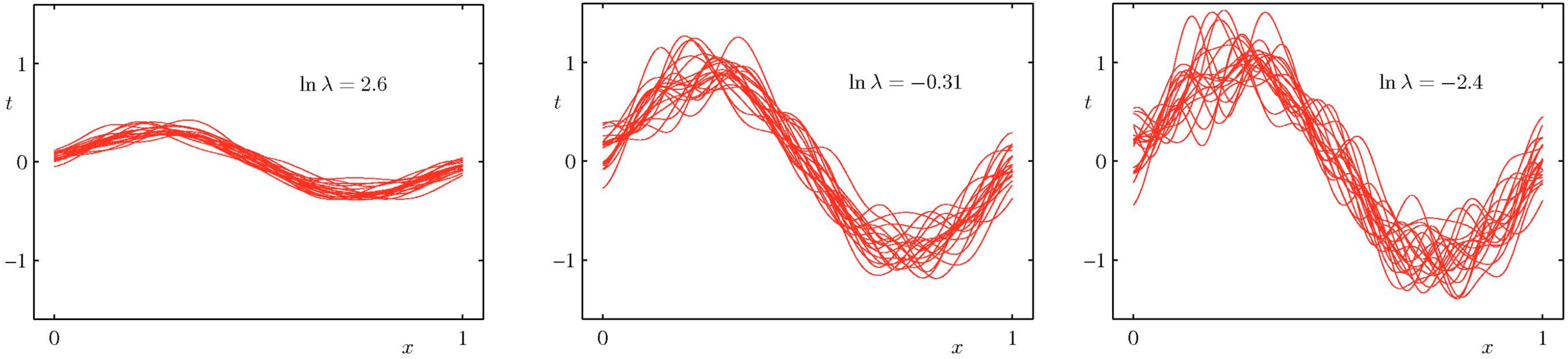
How uncertain the prediction is (given different training settings e.g. data and initialization)

Illustration of Bias-Variance

High ← Bias → Low



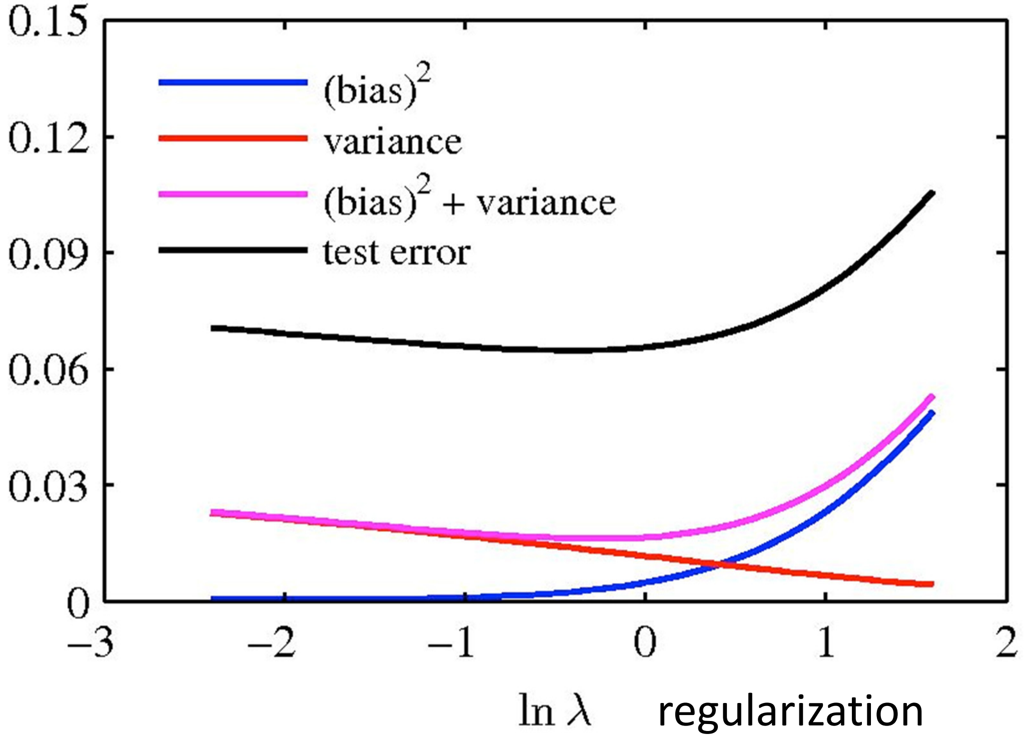
High ← Regularization → Low



Low ← Variance → High

Figures provided by Max Welling

Illustration of Bias-Variance



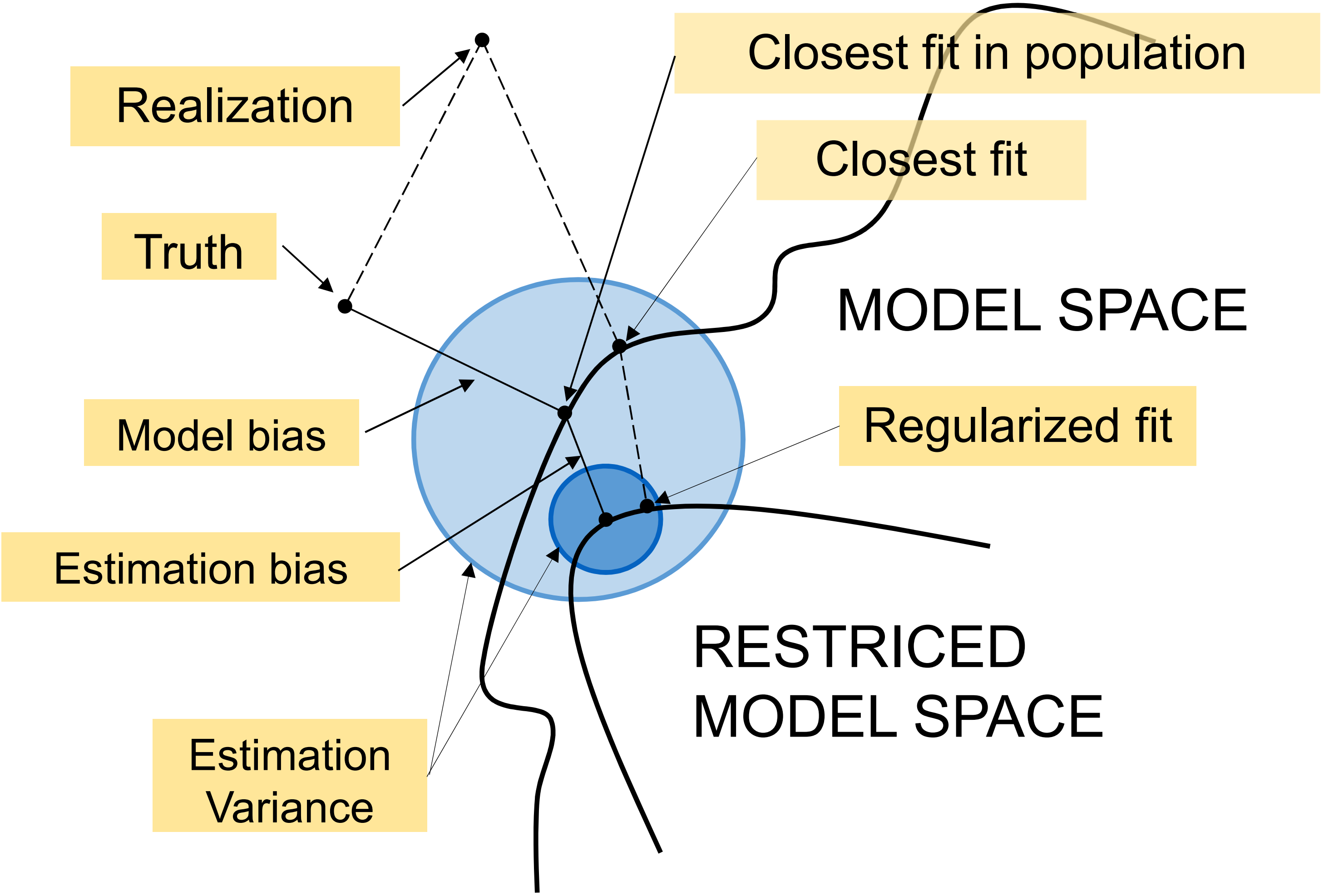
regularization

- Training error measures bias, but ignores variance.
- Testing error / cross-validation error measures both bias and variance.

Figures provided by Max Welling

Bias-Variance Decomposition

- Schematic of the behavior of bias and variance



Slide credit Liqing Zhang

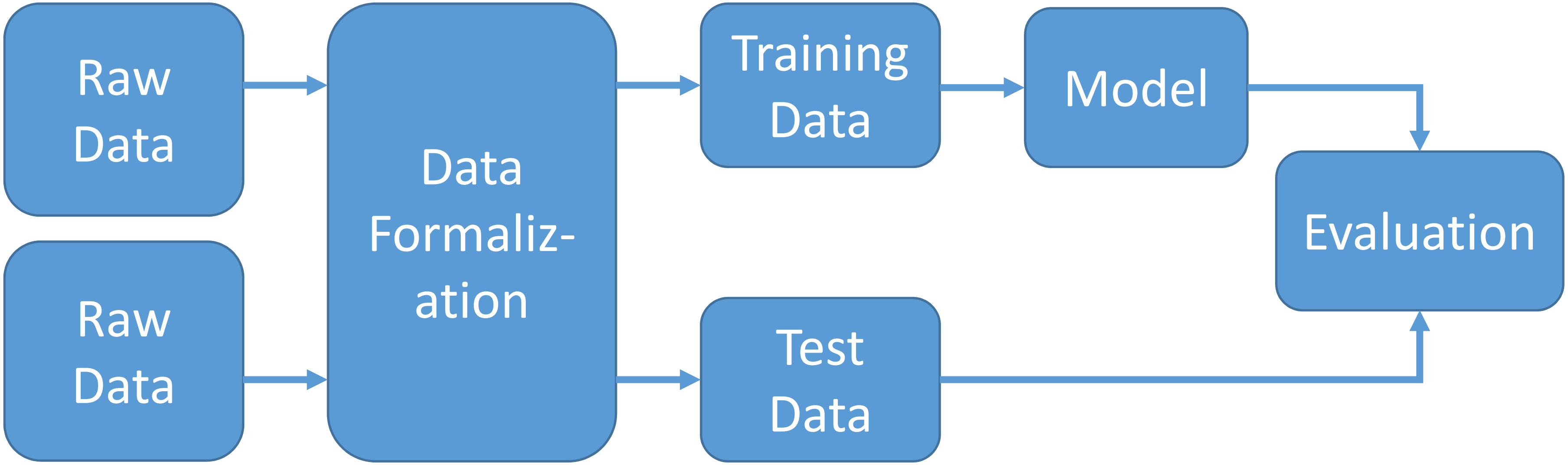
Hypothesis Space ERM Bound

Empirical Risk Minimization

Finite Hypothesis Space

Infinite Hypothesis Space

Machine Learning Process



- After selecting ‘good’ hyperparameters, we train the model over the whole training data and the model can be used on test data.

Generalization Ability

- Generalization Ability is the model prediction capacity on unobserved data
  - Can be evaluated by Generalization Error, defined by

$$R(f) = \mathbb{E}[\mathcal{L}(Y, f(X))] = \int_{X \times Y} \mathcal{L}(y, f(x))p(x, y)dxdy$$

  - where $p(x, y)$ is the underlying (probably unknown) joint data distribution

- Empirical estimation of GA on a training dataset is

$$\hat{R}(f) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(y_i, f(x_i))$$

A Simple Case Study on Generalization Error

- Finite hypothesis set $\mathcal{F} = \{f_1, f_2, \ldots, f_d\}$
- Theorem of generalization error bound:

For any function $f \in \mathcal{F}$, with probability no less than $1 - \delta$, it satisfies

$$R(f) \leq \hat{R}(f) + \epsilon(d, N, \delta)$$

where

$$\epsilon(d, N, \delta) = \sqrt{\frac{1}{2N}\Big(\log d + \log \frac{1}{\delta}\Big)}$$

- *N*: number of training instances
- *d:* number of functions in the hypothesis set

Section 1.7 in Dr. Hang Li's text book.

Lemma: Hoeffding Inequality

Let $X_1, X_2, \ldots, X_N$ be bounded independent random variables $X_i \in [a, b]$, the average variable *Z* is

$$Z = \frac{1}{N}\sum_{i=1}^{N} X_i$$

Then the following inequalities satisfy:

$$P(Z - \mathbb{E}[Z] \geq t) \leq \exp\left(\frac{-2Nt^2}{(b-a)^2}\right)$$

$$P(\mathbb{E}[Z] - Z \geq t) \leq \exp\left(\frac{-2Nt^2}{(b-a)^2}\right)$$

http://cs229.stanford.edu/extra-notes/hoeffding.pdf

Proof of Generalized Error Bound

- For binary classification, the error rate $0 \leq R(f) \leq 1$
- Based on Hoeffding Inequality, for $\epsilon > 0$, we have

$$P(R(f) - \hat{R}(f) \geq \epsilon) \leq \exp(-2N\epsilon^2)$$

- As $\mathcal{F} = \{f_1, f_2, \ldots, f_d\}$ is a finite set, it satisfies

$$\begin{aligned} P(\exists f \in \mathcal{F} : R(f) - \hat{R}(f) \geq \epsilon) &= P(\bigcup_{f \in \mathcal{F}} \{R(f) - \hat{R}(f) \geq \epsilon\}) \\ &\leq \sum_{f \in \mathcal{F}} P(R(f) - \hat{R}(f) \geq \epsilon) \\ &\leq d \exp(-2N\epsilon^2) \end{aligned}$$

Proof of Generalized Error Bound

- Equivalence statements

$$P(\exists f \in \mathcal{F} : R(f) - \hat{R}(f) \geq \epsilon) \leq d \exp(-2N\epsilon^2)$$

$$\Updownarrow$$

$$P(\forall f \in \mathcal{F} : R(f) - \hat{R}(f) < \epsilon) \geq 1 - d \exp(-2N\epsilon^2)$$

- Then setting

$$\delta = d \exp(-2N\epsilon^2) \quad \Leftrightarrow \quad \epsilon = \sqrt{\frac{1}{2N} \log \frac{d}{\delta}}$$

The generalized error is bounded with the probability

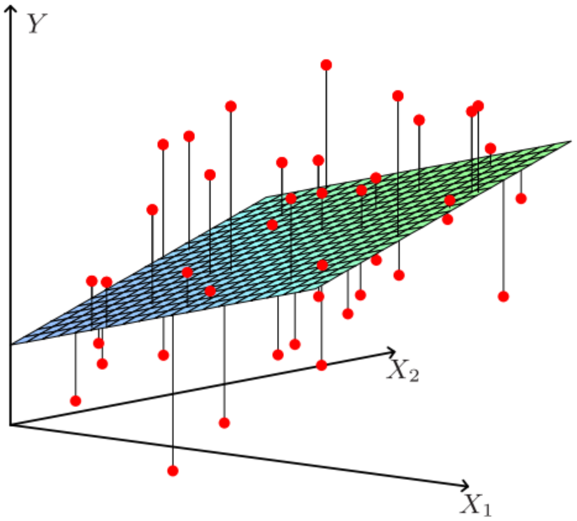
$$P(R(f) < \hat{R}(f) + \epsilon) \geq 1 - \delta$$

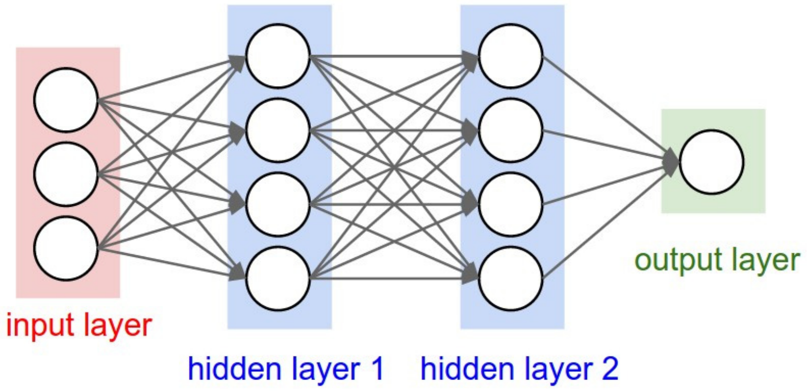
□

For Infinite Hypothesis Space

- Many hypothesis classes, including any parameterized by real numbers actually contain an infinite number of functions
  - E.g., linear models, neural networks

$$f(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 \qquad f(x) = \sigma(W_3(W_2 \tanh(W_1 x + b_1) + b_2) + b_3)$$





Quantizing Real Numbers

- Suppose we have an *H* hypothesis that is parameterized by *m* real numbers
- In a computer, each real number is represented using 64 bits (double floating)
- Thus the hypothesis class actually consists of at most $d = 2^{64m}$ difference hypotheses

$$\epsilon(d, N, \delta) = \sqrt{\frac{1}{2N}\Big(\log d + \log\frac{1}{\delta}\Big)}$$

$$\Rightarrow \epsilon(d, N, \delta) = \sqrt{\frac{1}{2N}\Big(64m + \log\frac{1}{\delta}\Big)}$$

$$\Rightarrow N = \frac{1}{2\epsilon^2}\Big(64m + \log\frac{1}{\delta}\Big) = O_{\epsilon,\delta}(m)$$

Sample Complexity

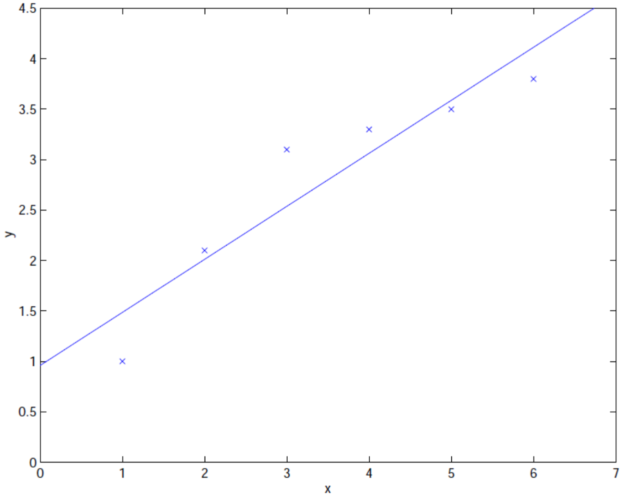
- For a model parameterized by *m* real numbers, in order to acquire the generalization error no higher than ϵ with at least $1 - \delta$ probability, we need *N* training samples as

$$N \geq \frac{1}{2\epsilon^2}\left(64m + \log\frac{1}{\delta}\right) = O_{\epsilon,\delta}(m)$$

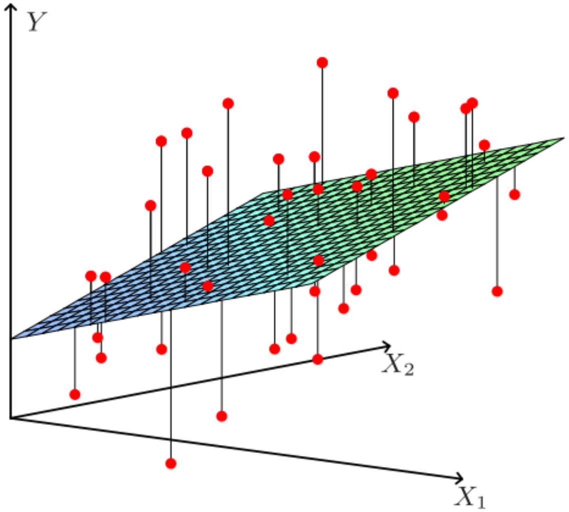
- which is linear w.r.t. the parameter number

Examples of Sample Complexity

- For fitting linear regression on *k*-dimensional data



$$f(x) = \theta_0 + \theta_1 x$$



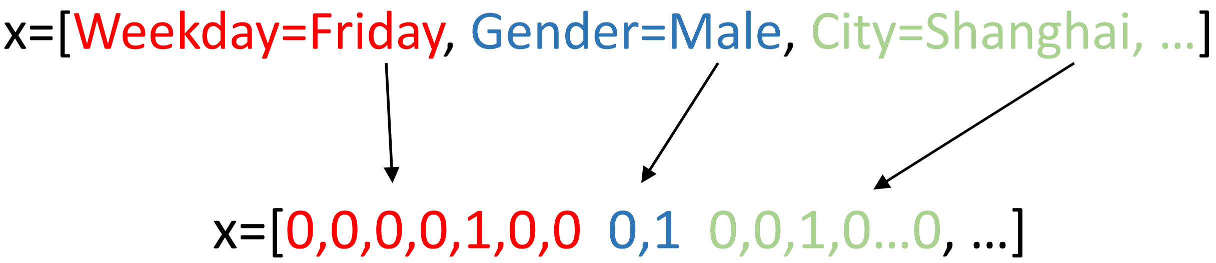
$$f(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

For 1-dimension data linear regression, we normally need around 10 points to fit a straight line with some confidence

For 2-dimension data linear regression, we normally need around 20 points to fit a hyperplane with some confidence

Examples of Sample Complexity

- For fitting linear regression on *k*-dimensional data
- A standard feature engineering paradigm



x=[Weekday=Friday, Gender=Male, City=Shanghai, …]

x=[0,0,0,0,1,0,0 0,1 0,0,1,0…0, …]

1 5:1 9:1 12:1 45:1 154:1 509:1 4089:1 45314:1 988576:1
0 2:1 7:1 18:1 34:1 176:1 510:1 3879:1 71310:1 818034:1

…

$$f(x) = \theta_0 + \sum_{i=1}^{10^6} \theta_i x_i$$

For 1-million dimensional data linear regression, we normally need around 10 million points to fit a straight line with some confidence

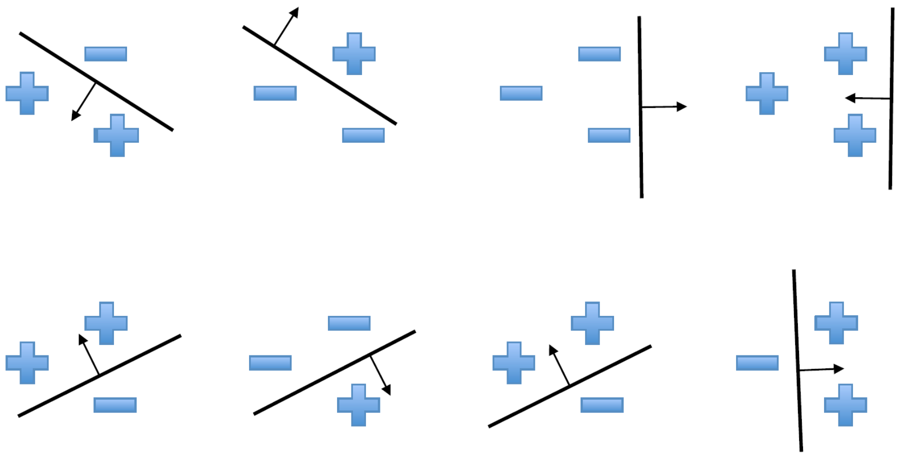
VC Dimensions

Shattering

- Definition
  - A model class can shatter a set of points

$$x^{(1)}, x^{(2)}, \ldots, x^{(n)}$$

if for every possible labeling over those points, there exists a model in that class that obtains zero training error.



For example, linear model class shatters above three-point set

VC Dimension

Vladimir Vapnik

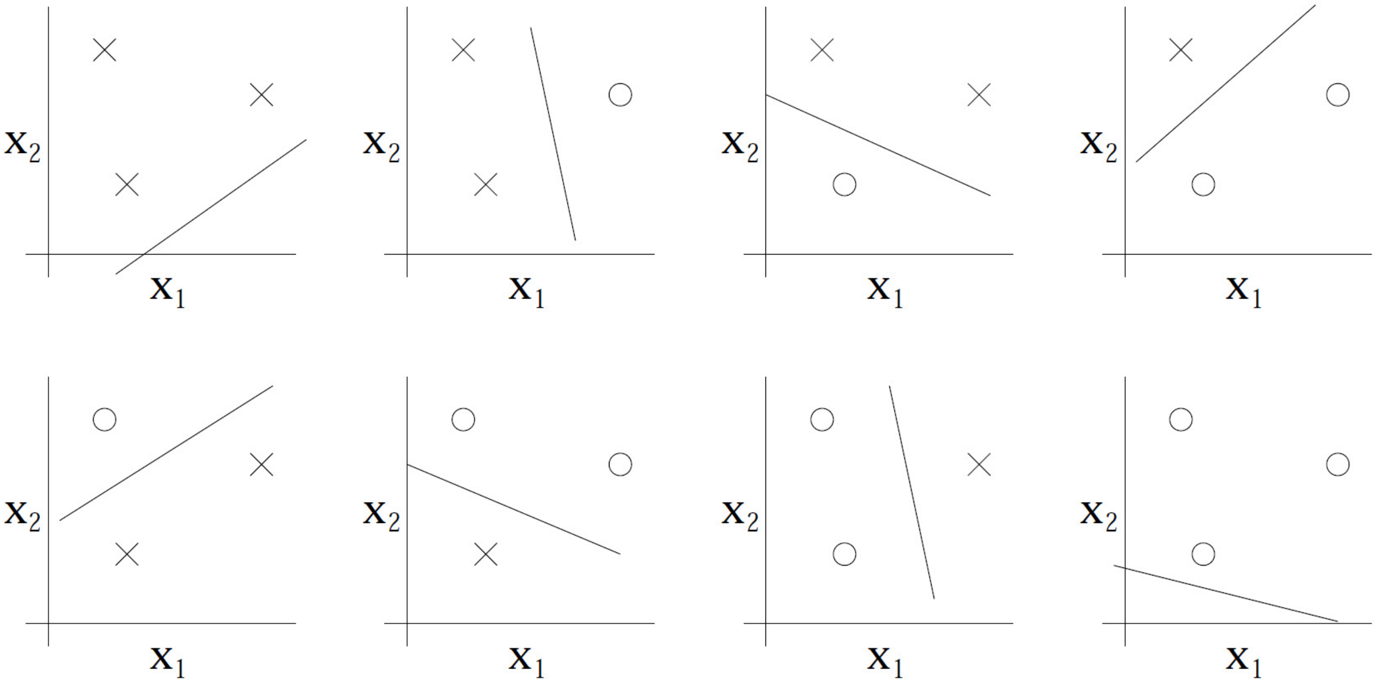
Alexey Chervonenkis

- The larger the subset of *X* that can be shattered, the more expressive the hypothesis space is, i.e. the less biased.
- The Vapnik-Chervonenkis dimension, VC(*H*), of hypothesis space *H* defined over instance space *X* is the size of the largest finite subset of *X* shattered by *H*. If arbitrarily large finite subsets of *X* can be shattered then VC(*H*) = ∞
- If there exists at least one subset of *X* of size *d* that can be shattered then VC(*H*) ≥ *d*. If no subset of size *d* can be shattered, then VC(*H*) < *d.*
- Since $|H| \geq 2^m$, to shatter *m* instances, $\text{VC}(H) \leq \log_2|H|$

Ray Mooney

VC Dimension Example

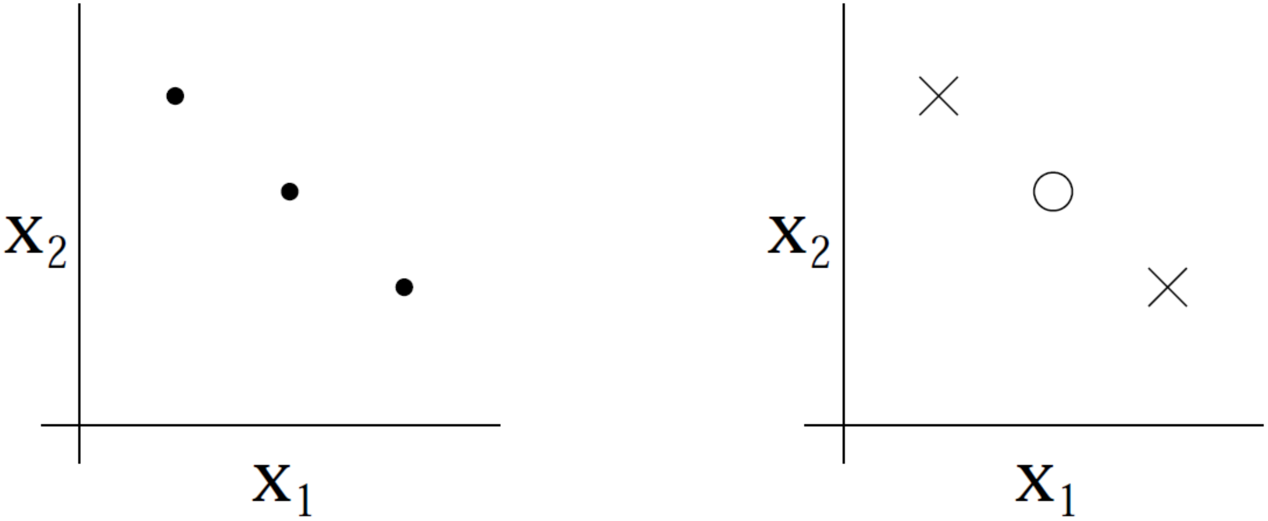
- Consider linear models in the real-plane. Some 3 instances can be shattered.



All 8 possible labeling can be separated.

VC Dimension Example

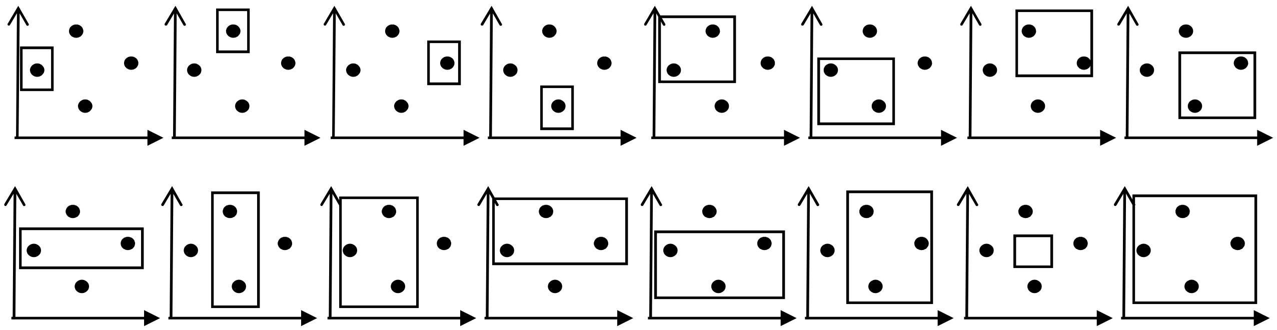
- Consider linear models in the real-plane. Some 3 instances lying in a straight line can NOT be shattered.



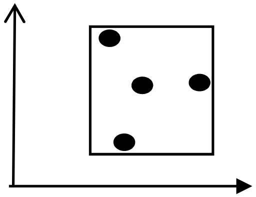
- As we can find a 3-instance set to shatter by the linear model, the VC dimension of linear models is at least 3

VC Dimension Example

- Consider axis-parallel rectangles in the real-plane, i.e. conjunctions of intervals on two real-valued features. Some 4 instances can be shattered.

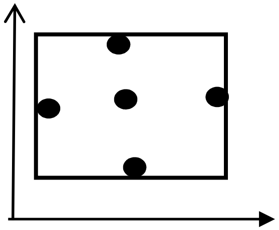


Some 4 instances cannot be shattered:



VC Dimension Example (cont)

- No five instances can be shattered since there can be at most 4 distinct extreme points (min and max on each of the 2 dimensions) and these 4 cannot be included without including any possible 5th point.



- Therefore VC(H) = 4
- Generalizes to axis-parallel hyper-rectangles (conjunctions of intervals in n dimensions): VC(H)=2n.

Ray Mooney

Upper Bound on Sample Complexity with VC

- Using VC dimension as a measure of expressiveness, the following number of examples have been shown to be sufficient for PAC Learning (Blumer *et al.*, 1989).

$$N = \frac{1}{\epsilon}\left(4\log_2\left(\frac{2}{\delta}\right) + 8\mathrm{VC}(H)\log_2\left(\frac{13}{\epsilon}\right)\right)$$

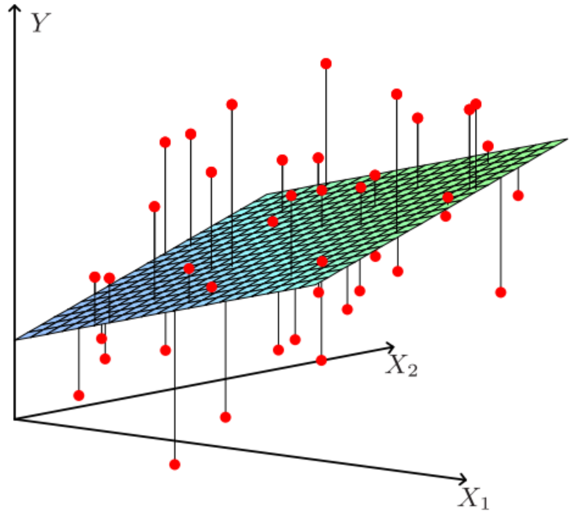
- Compared to the previous result using log|*H*|, this bound has some extra constants and an extra $\log_2(1/\varepsilon)$ factor. Since VC(*H*) ≤ $\log_2$|*H*|, this can provide a tighter upper bound on the number of examples needed for PAC learning.

$$N = \frac{1}{2\epsilon^2}\left(\log|H| + \log\frac{1}{\delta}\right)$$

Ray Mooney

Some Examples of VC Dimension

- The VC dimension of a hyperplane in *d* dimension is *d*+1
  - It is a coincidence that the VC dimension of a hyperplane is almost identical to the number of parameters needed to define a hyperplane

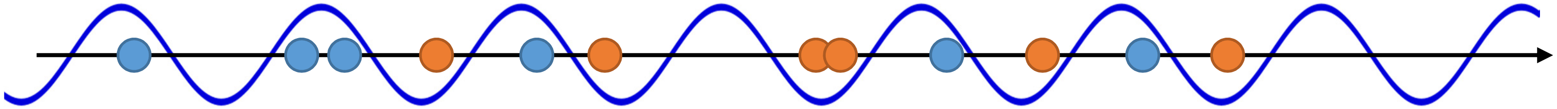


$$f(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

Some Examples of VC Dimension

- A sine wave has infinite VC dimension but only 2 parameters
  - By choosing the phase & period carefully we can shatter any random set of 1D data points

$$h(x) = \sin(ax + b)$$

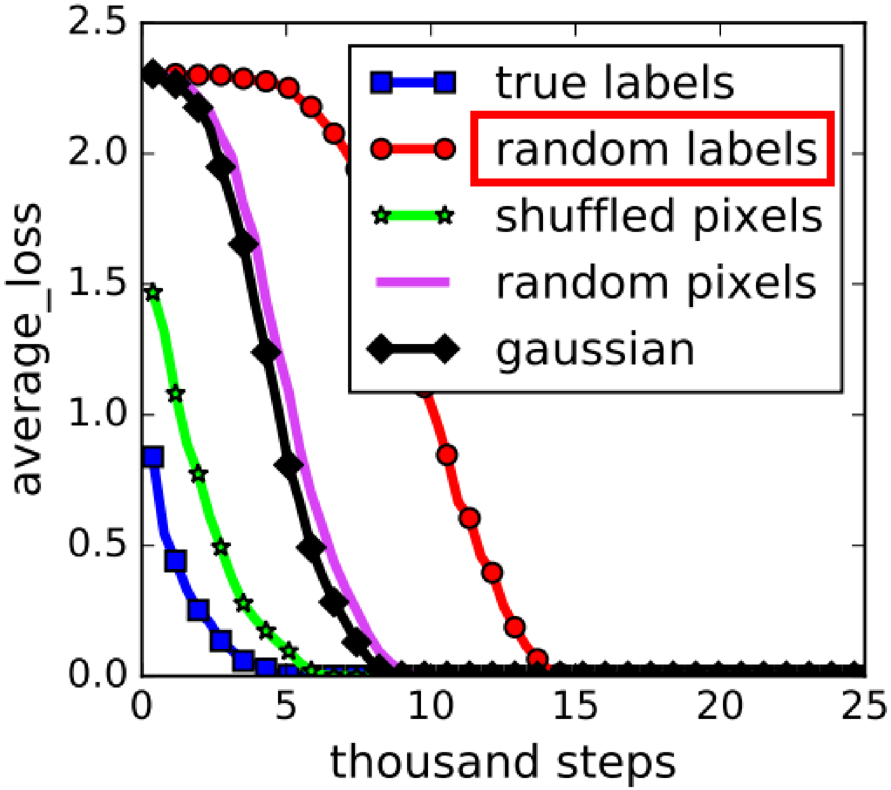


Some Examples of VC Dimension

- Neural networks with some types of activation functions also have infinite VC dimension

- Dataset: CIFAR-10
  - 50,000 training images
  - Net: Inception model

- MLP also converged to zero training loss



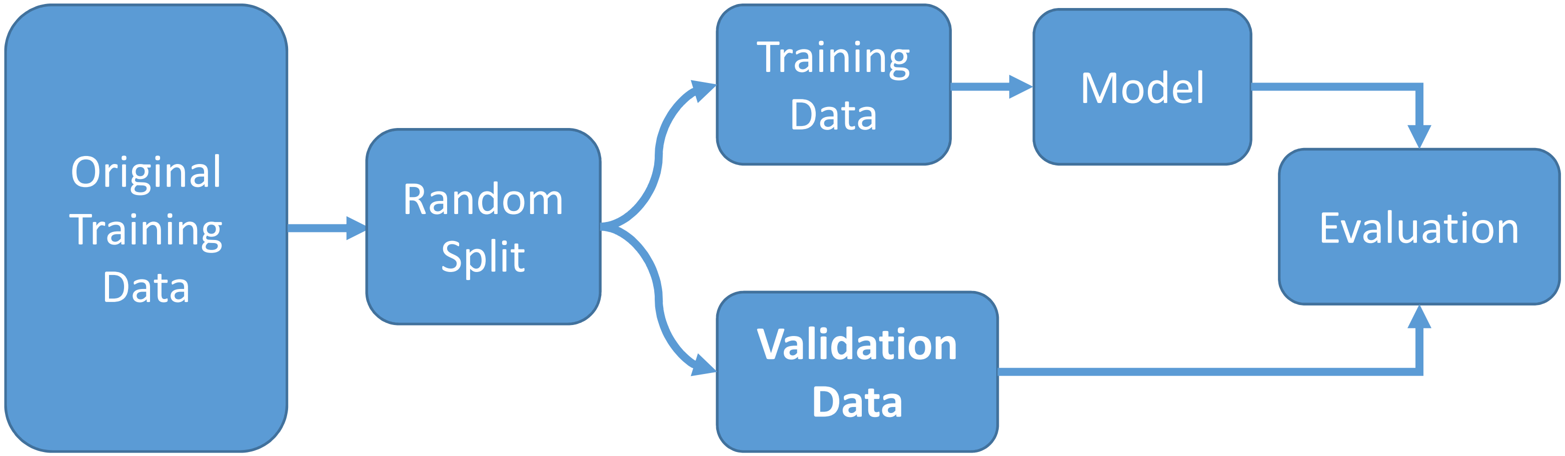
Zhang, Chiyuan, et al. "Understanding deep learning requires rethinking generalization." *arXiv preprint arXiv:1611.03530* (2016).

Content

- Learning Theory
  - Bias-Variance Decomposition
  - Finite Hypothesis Space ERM Bound
  - Infinite Hypothesis Space ERM Bound
  - VC Dimension

- Model Selection
  - Cross Validation
  - Feature Selection
  - Occam’s Razor for Bayesian Model Selection

Cross Validation for Model Selection



Original Training Data

Random Split

Training Data

Model

Validation Data

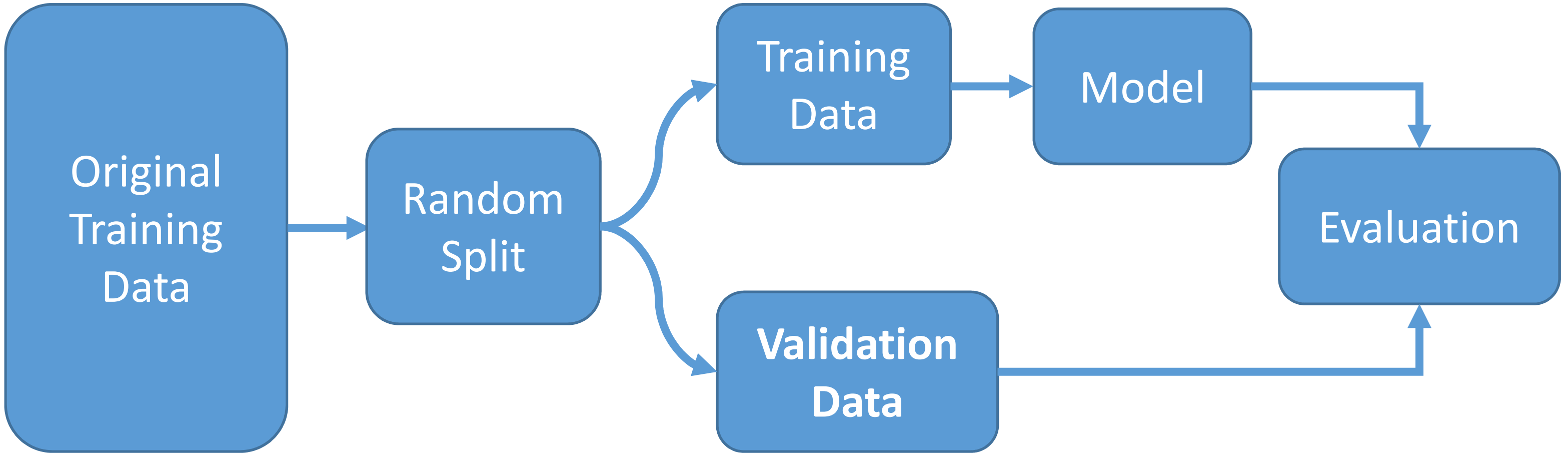
Evaluation

- For example, 5-fold cross validation
  - Split the dataset into 5 folds

1	2	3	4	5

  - Cross validation 1: train the model on 1,2,3,4, and validate on 5
  - Cross validation 2: train the model on 2,3,4,5, and validate on 1
  - …

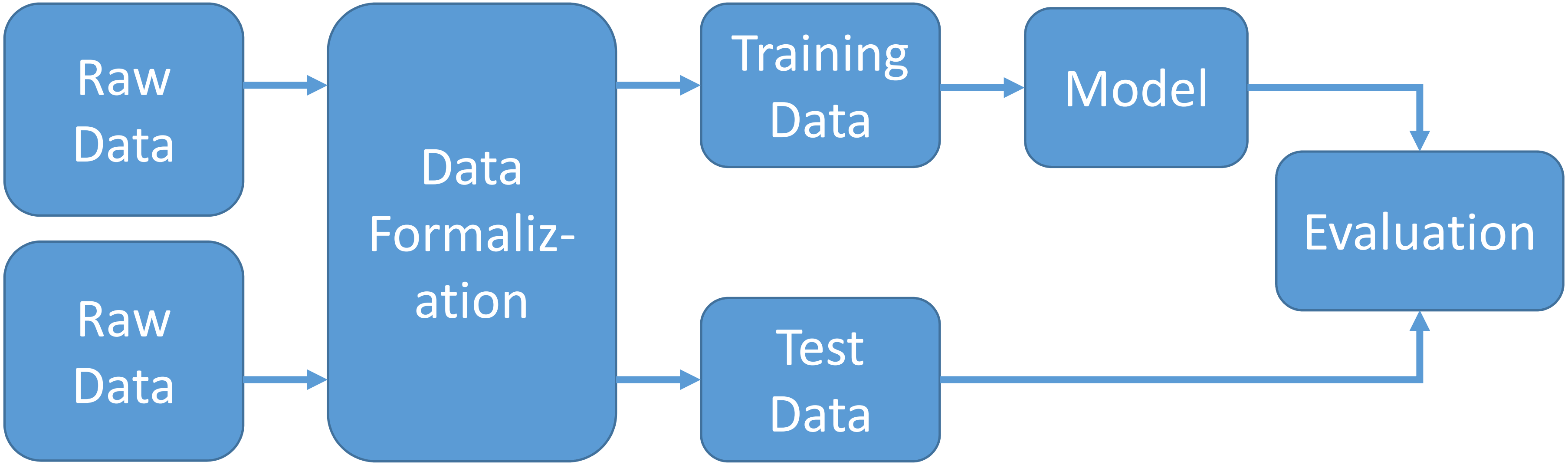
Cross Validation for Model Selection



K-fold Cross Validation

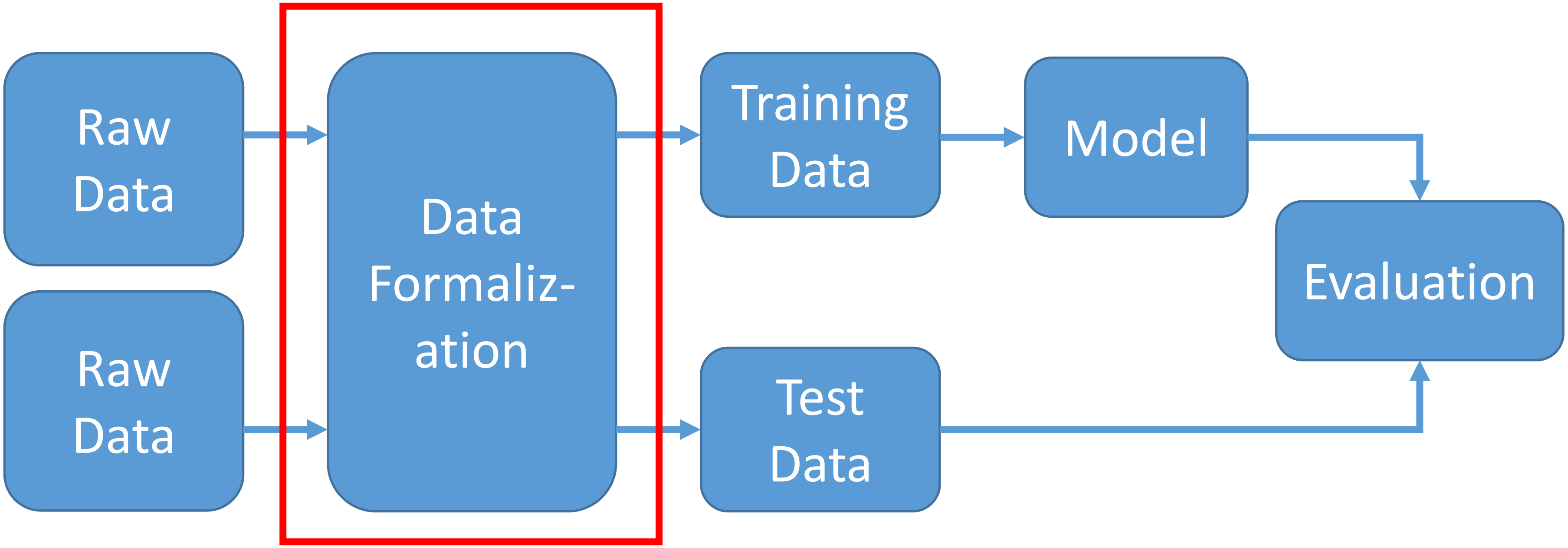
1. Set hyperparameters
2. For *K* times repeat:
   - Randomly split the original training data into training and validation datasets
   - Train the model on training data and evaluate it on validation data, leading to an evaluation score
3. Average the *K* evaluation scores as the model performance

Machine Learning Process



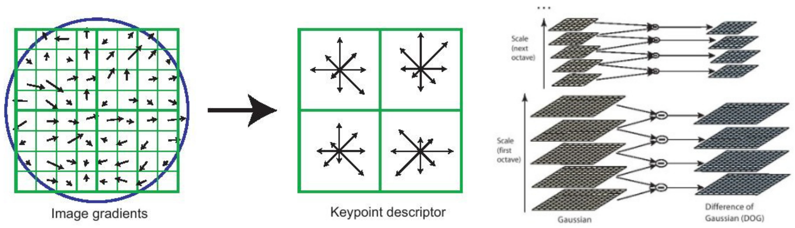
- After selecting ‘good’ hyperparameters, we train the model over the whole training data and the model can be used on test data.

Data Representation

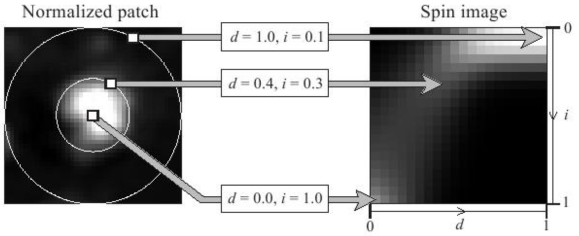


- The data is formalized into feature representation
  - How to select ‘good’ features to improve model performance? i.e. generalization ability

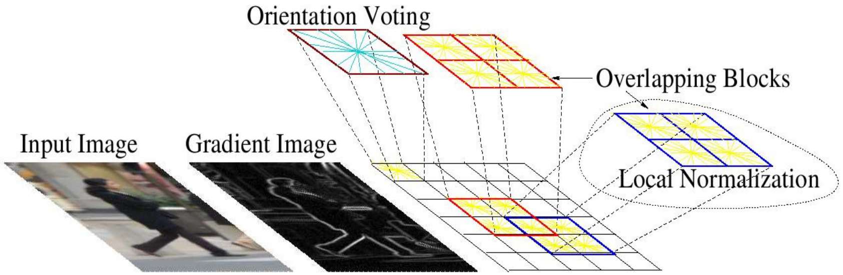
Features in Computer Vision



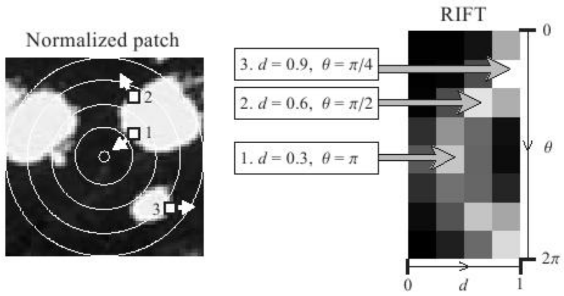
SIFT



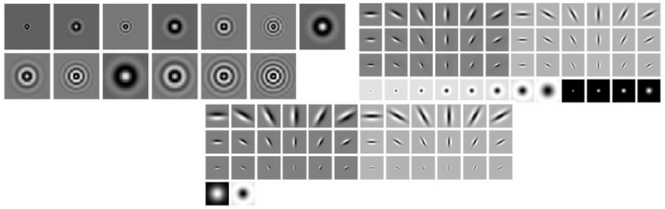
Spin image



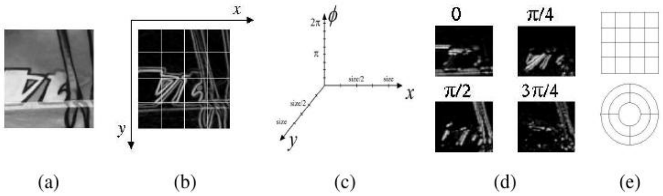
HoG


Normalized patch
RIFT
2
1
3
3. $d = 0.9,\ \theta = \pi/4$
2. $d = 0.6,\ \theta = \pi/2$
1. $d = 0.3,\ \theta = \pi$
0
θ
2π
0
d
1

RIFT



Textons



GLOH

Features in Text Classification

- Input text

SJTU is a public research university in Shanghai, China, established in 1896. Now it is one of C9 universities in China.

- Bag-of-words representation

SJTU:1, is:2, a:1, public:1, research:1, university:2, in:3, Shanghai:1, China:2, establish:1, 1896:1, now:1, it:1, one:1, of:1

- The size of vocabulary would be over 100k

Feature Selection

- Various feature representations make each data instance formalized into a high-dimensional vector
    - which needs a large number of training instances for a reliable model, i.e. the generalization error is small

$$N \geq \frac{1}{2\epsilon^2}\Big(64m + \log\frac{1}{\delta}\Big) = O_{\epsilon,\delta}(m)$$

- We have already known GE is decomposed as

$$\mathrm{Err}(x_0) = \sigma_\epsilon^2 + \mathrm{Bias}^2(\hat{f}(x_0)) + \mathrm{Var}(\hat{f}(x_0))$$

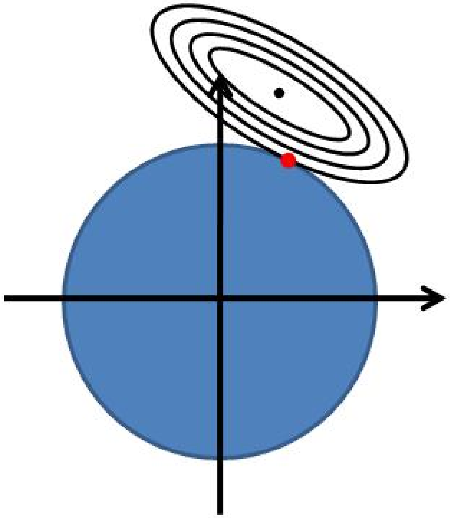
- Small number of features may increase the model bias
- Large number of features may increase the variance
- Feature selection: a trade-off between bias and variance

L1 Regularization for Feature Selection

- L2-Norm (Ridge)

$$\Omega(\theta) = \|\theta\|_2^2 = \sum_{m=1}^{M} \theta_m^2$$

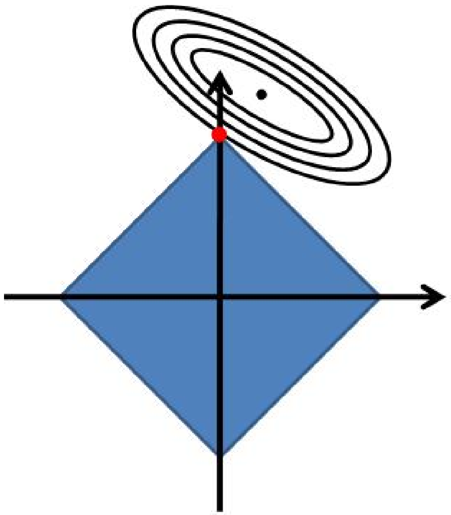
$$\min_\theta \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y_i, f_\theta(x_i)) + \lambda \|\theta\|_2^2$$



- L1-Norm (LASSO)

$$\Omega(\theta) = \|\theta\|_1 = \sum_{m=1}^{M} |\theta_m|$$

$$\min_\theta \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y_i, f_\theta(x_i)) + \lambda \|\theta\|_1$$



Feature Selection Methods

- Unsupervised

	Linear	Non-linear
Selection	Correlation between inputs	Mutual information between inputs
Projection	Principal component analysis	Sammon’s mapping, Self-organizing maps

- Supervised

	Linear	Non-linear
Selection	Correlation between inputs and target	Mutual information between inputs and target, greedy selection, genetic algorithms
Projection	Linear discriminant analysis, partial least squares	Multilayer perceptrons, auto-encoders, projection pursuit

Feature Selection Methods Study

- Yang, Yiming, and Jan O. Pedersen. "A comparative study on feature selection in text categorization." *ICML*. Vol. 97. 1997.

- Studied task: text classification
  - Features: bag of words, each dimension represents a term
  - Instances: a document of words (terms)
  - Target: one of *m* classes of the document

Feature Selection Methods

- Document frequency (DF)
  - i.e., the number of documents in which a feature occurs
  - Select the high DF features
    - Assumption: low frequency features are either non-informative or not influential for global performance
- Information Gain (IG)
  - IG measures the information obtained for target prediction by knowing the feature

$$G(t) = -\sum_{i=1}^{m} P(c_i) \log P(c_i) + P(t) \sum_{i=1}^{m} P(c_i|t) \log P(c_i|t) + P(\bar{t}) \sum_{i=1}^{m} P(c_i|\bar{t}) \log P(c_i|\bar{t})$$

Yang, Yiming, and Jan O. Pedersen. "A comparative study on feature selection in text categorization." *Icml*. Vol. 97. 1997.

Feature Selection Methods

- Mutual Information (MI)
  - MI of two random variables is a measure of the mutual dependence between the two variables

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} \log \frac{p(x,y)}{p(x)p(y)}$$

  - For MI between a feature *t* and the target *c* (as two random variables)

$$I(t,c) = \log \frac{P(t,c)}{P(t)P(c)} \simeq \log \frac{A \times N}{(A+C) \times (A+B)}$$

  - *A*: #. documents *t* and *c* co-occur
  - *B*: #. documents *t* occurs without *c*
  - *C*: #. documents *c* occurs without *t*
  - *N*: #. documents in total

Yang, Yiming, and Jan O. Pedersen. "A comparative study on feature selection in text categorization." *Icml*. Vol. 97. 1997.

Feature Selection Methods

- Mutual Information (MI)
  - MI of two random variables is a measure of the mutual dependence between the two variables

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} \log \frac{p(x,y)}{p(x)p(y)}$$

  - For MI between a feature *t* and the target *c* (as two random variables)

$$I(t,c) = \log \frac{P(t,c)}{P(t)P(c)} \simeq \log \frac{A \times N}{(A+C) \times (A+B)}$$

  - Two ways of measuring the goodness of a feature

$$I_{\text{avg}}(t) = \sum_{i=1}^{m} P(c_i) I(t,c_i) \qquad I_{\max}(t) = \max_{i=1}^{m} \{I(t,c_i)\}$$

Yang, Yiming, and Jan O. Pedersen. "A comparative study on feature selection in text categorization." *Icml*. Vol. 97. 1997.

Feature Selection Methods

- χ^2 Statistic (CHI)
  - Measures the lack of independence between *t* and *c*

$$\chi^2(t,c) = \frac{N \times (AD - CB)^2}{(A+C) \times (B+D) \times (A+B) \times (C+D)}$$

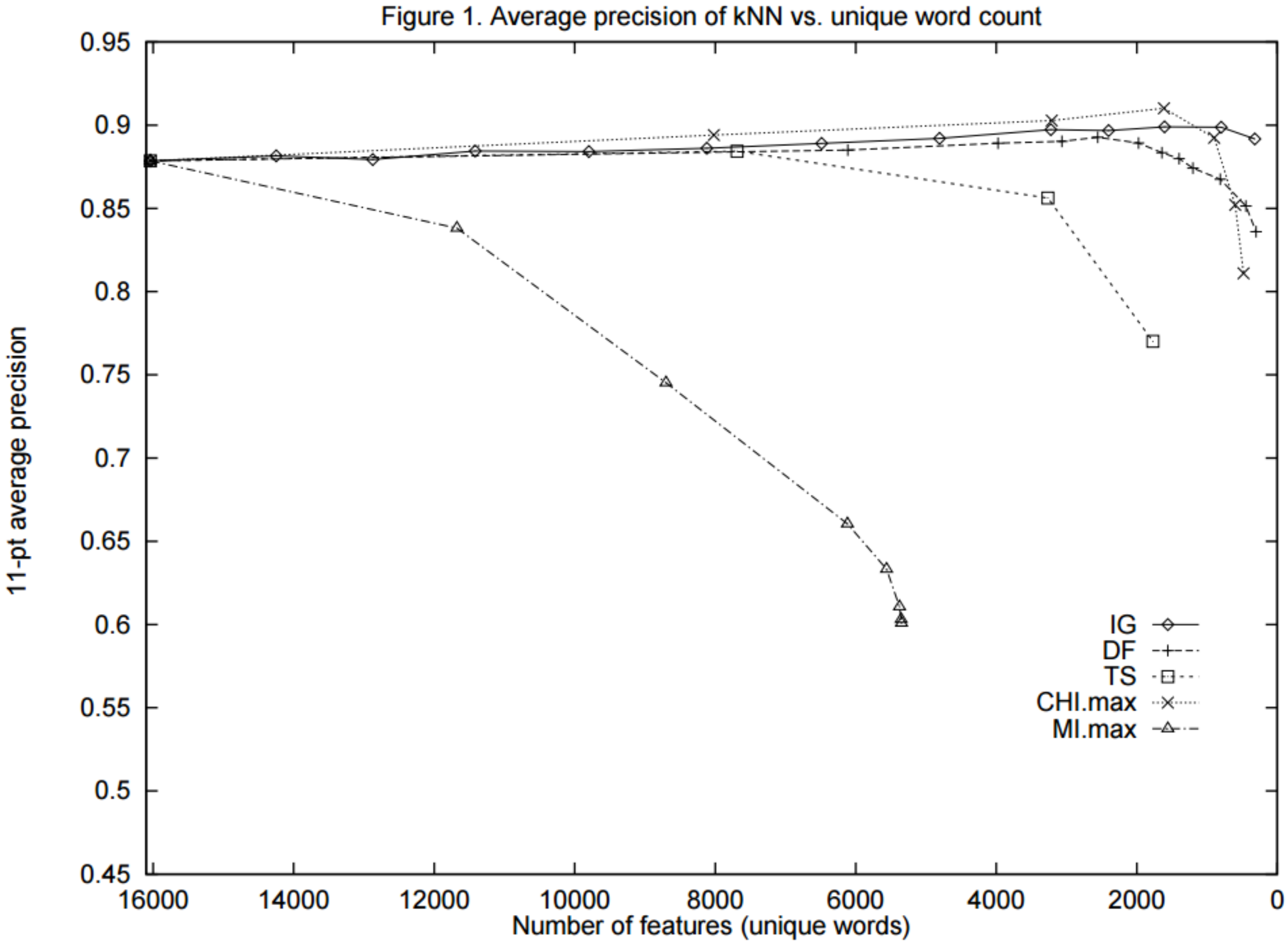
- *A*: #. documents *t* and *c* co-occur
- *B*: #. documents *t* occurs without *c*
- *C*: #. documents *c* occurs without *t*
- *D*: #. documents neither *c* not *t* occurs
- *N*: #. documents in total

  - Two ways of measuring the goodness of a feature

$$I_{\text{avg}}(t) = \sum_{i=1}^{m} P(c_i)\chi^2(t,c_i) \qquad I_{\max}(t) = \max_{i=1}^{m}\{\chi^2(t,c_i)\}$$

Yang, Yiming, and Jan O. Pedersen. "A comparative study on feature selection in text categorization." *Icml*. Vol. 97. 1997.

Empirical Performance

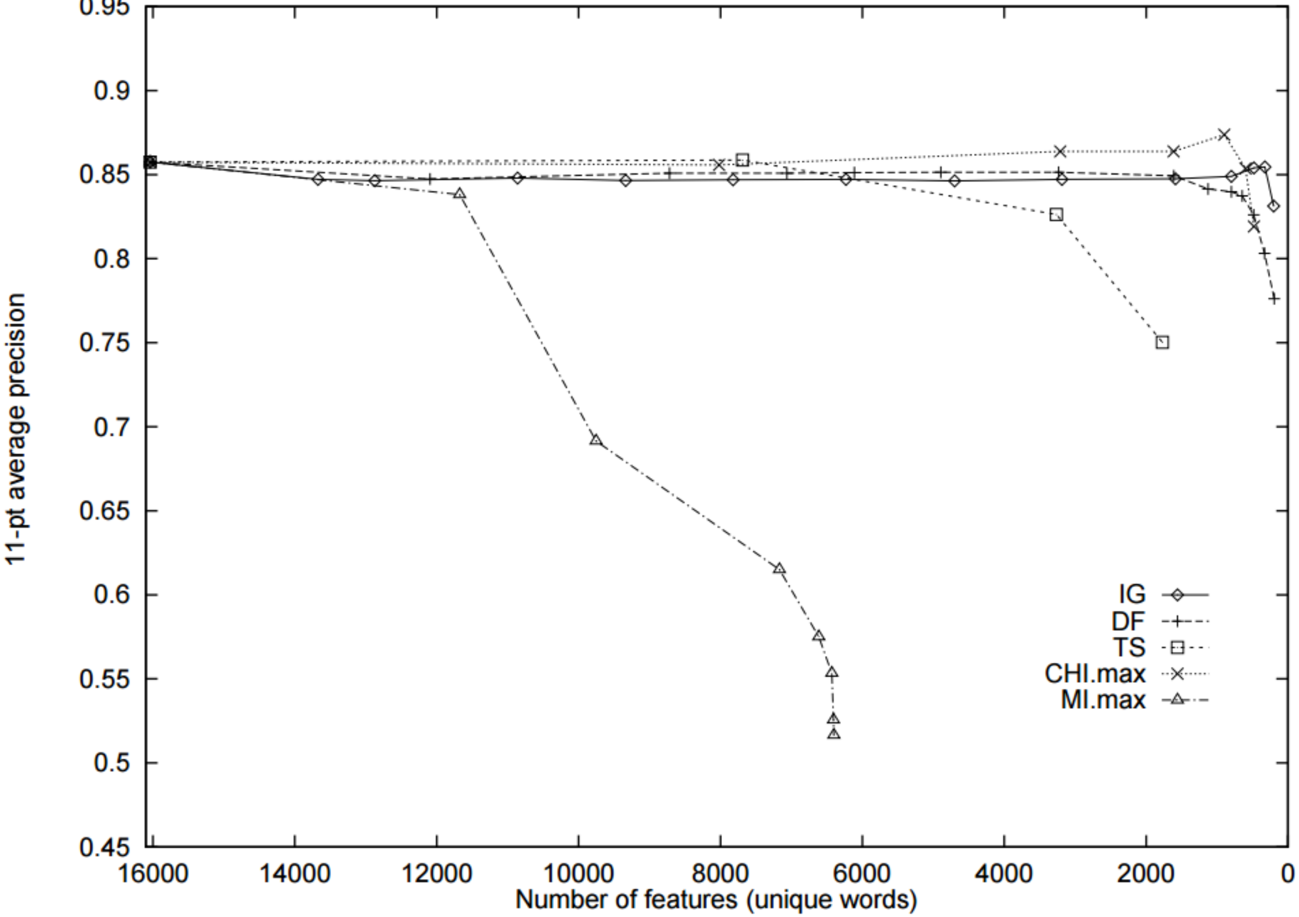


kNN on Reuters dataset: 9610 training document, 3662 test documents

Yang, Yiming, and Jan O. Pedersen. "A comparative study on feature selection in text categorization." *Icml*. Vol. 97. 1997.

Empirical Performance

Figure 2. Average precision of LLSF vs. unique word count



Linear model on Reuters dataset: 9610 training document, 3662 test documents

Yang, Yiming, and Jan O. Pedersen. "A comparative study on feature selection in text categorization." *Icml*. Vol. 97. 1997.

“Occam’s Razor” Result (Blumer *et al*., 1987)

- Assume that a concept can be represented using at most *n* bits in some representation language.
- Given a training set, assume the learner returns the consistent hypothesis representable with the least number of bits in this language.
- Therefore the effective hypothesis space is all concepts representable with at most *n* bits.
- Since *n* bits can code for at most 2^n hypotheses, $|H|=2^n$, so sample complexity if bounded by:

$$\left(\log\frac{1}{\delta}+\log 2^n\right)/\epsilon=\left(\log\frac{1}{\delta}+n\log 2\right)/\epsilon$$

Ray Mooney

Principle of Occam's razor

Among competing hypotheses, the one with the fewest assumptions should be selected.

- Recall the function set $\{f_\theta(\cdot)\}$ is called hypothesis space

$$\min_{\theta} \underbrace{\boxed{\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(y_i, f_\theta(x_i))}}_{\text{Original loss}} + \lambda \underbrace{\boxed{\Omega(\theta)}}_{\text{Penalty on assumptions}}$$

Ray Mooney

Model Selection

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y_i, f_\theta(x_i)) + \lambda ||\theta||_2^2$$

- An ML solution has model parameters θ and optimization hyperparameters λ
- Hyperparameters
  - Define higher level concepts about the model such as complexity, or capacity to learn.
  - **Cannot be learned directly from the data** in the standard model training process and need to be predefined.
  - Can be decided by setting different values, training different models, and choosing the values that test better
- Model selection (or hyperparameter optimization) cares how to select the optimal hyperparameters.

Bayesian Occam's Razor

- For a model *H* and the observed data *D*, the posterior of the parameter is

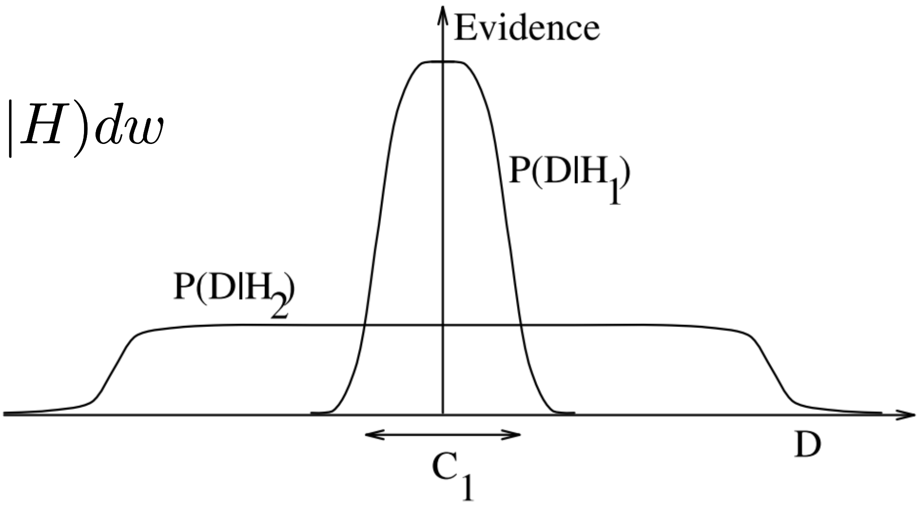
$$p(w|D,H) = \frac{p(D|w,H)p(w|H)}{p(D|H)}$$

- Bayes' rule also provides a posterior over models

$$p(H|D) \propto p(D|H)p(H)$$

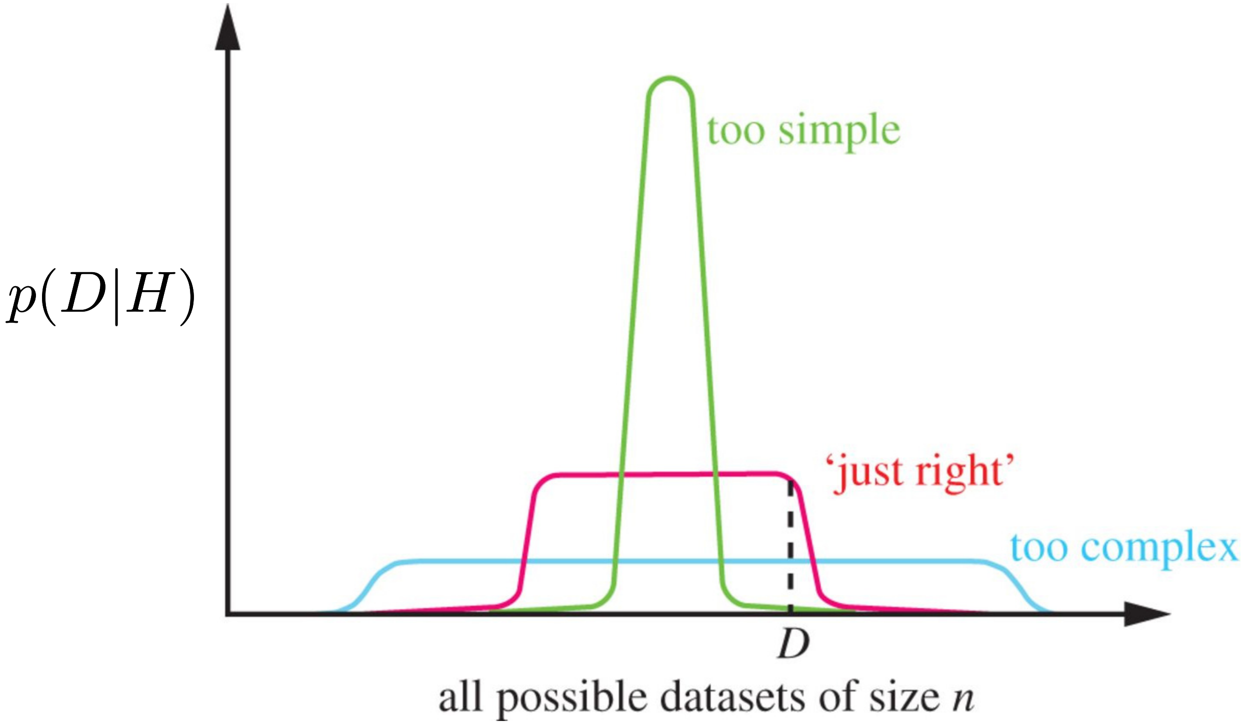
$$p(D|H) = \int_w p(D|w,H)p(w|H)dw$$

- H_1 is a simple model focusing on data in region C_1
- H_2 is a complex model which can model data in a wider region



http://mlg.eng.cam.ac.uk/zoubin/papers/05occam/occam.pdf

Bayesian Occam's Razor



- A complex model spreads its mass over many more possible datasets
- A simple model concentrates its mass on a smaller fraction of possible data
- The normalization $\int_D p(D|H) dD = 1$ is what results in an automatic Occam razor

http://rsta.royalsocietypublishing.org/content/371/1984/20110553