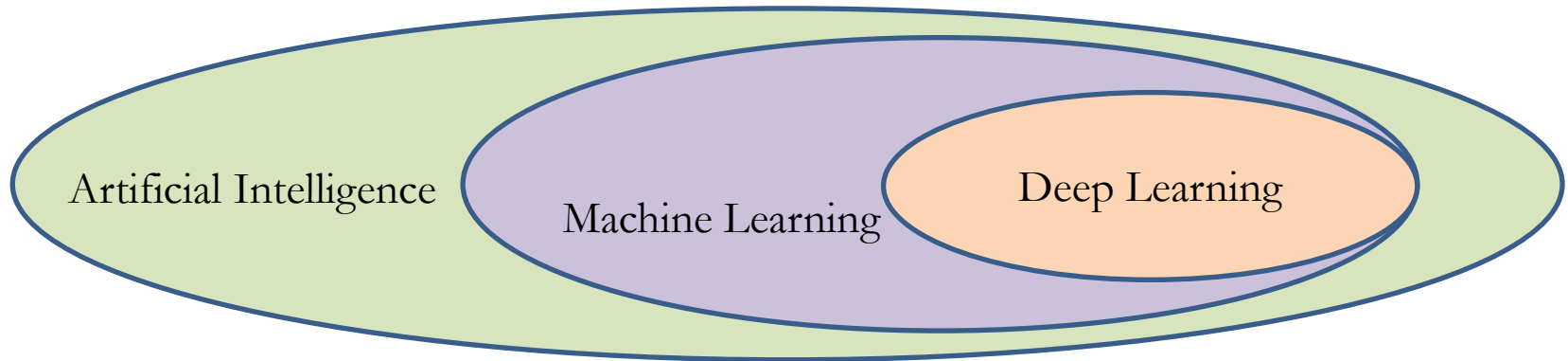


CSCE 5218 & 4930

Deep Learning

Machine Learning Overview

Machine Learning Overview



- Deep learning is a specific group of algorithms falling in the broader realm of machine learning
- All ML/DL algorithms roughly match schema:
 - Learn a mapping from input to output $f: x \rightarrow y$
 - x : image, text, etc.
 - y : {cat, notcat}, {1, 1.5, 2, ...}, etc.
 - f : this is where the magic happens

Machine Learning Overview


$$y' = f(\mathbf{x})$$

A diagram illustrating the machine learning equation $y' = f(\mathbf{x})$. Three red arrows point from labels below to variables in the equation: one from 'output' to y' , one from 'prediction function' to f , and one from 'input' to \mathbf{x} .

- **Training:** given a *training set* of labeled examples $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$, estimate the prediction function f by minimizing the prediction error on the training set
- **Testing:** apply f to a never-before-seen *test example* \mathbf{x} and output the predicted value $y' = f(\mathbf{x})$

Machine Learning Overview

- Example:
 - Predict whether an email is spam or not:

Sebring, Tracy 
To: Batra, Dhruv
ECE 4424 proposal

CUSP has approved ECE 4424 with the following changes: Can you copy of the proposal with these items addressed? (see below)
Thanks!!!
Tracy

VS

nadia bamba
To: undisclosed recipients: ;
Reply-To: nadia bamba
From Miss Nadia BamBa,

January 19, 2015 5:57 AM
[Hide Details](#)

From Miss Nadia BamBa,

Greeting, Permit me to inform you of my desire of going into business relationship with you. I am Nadia BamBa the only Daughter of late Mr and Mrs James BamBa, My father was a director of cocoa merchant in Abidjan, the economic capital of Ivory Coast before he was poisoned to death by his business associates on one of their outing to discus on a business deal. When my mother died on the 21st October 2002, my father took me very special because i am motherless.

Before the death of my father in a private hospital here in Abidjan, He secretly called me on his bedside and told me that he had a sum of \$6, 8000.000(SIX Million EIGHT HUNDRED THOUSAND), Dollars) left in a suspense account in a Bank here in Abidjan, that he used my name as his first Daughter for the next of kin in deposit of the fund.

He also explained to me that it was because of this wealth and some huge amount of money That his business associates supposed to balance him from the deal they had that he was poisoned by his business associates, that I should seek for a God fearing foreign partner in a country of my choice where I will transfer this money and use it for investment purposes, (such as real estate Or Hotel management).please i am honourably seeking your assistance in the following ways.

- 1) To provide a Bank account where this money would be transferred to.
- 2) To serve as the guardian of this Money since I am a girl of 19 years old.
- 3)Your private phone number's and your family background' s that we can know each order more.

Machine Learning Overview

- Example:
 - Predict whether an email is spam or not.
 - \mathbf{x} = words in the email, multi-hot representation of size $|V| \times 1$, where V is the full vocabulary and $x(j) = 1$ iff word j is mentioned
 - $y = 1$ (if spam) or 0 (if not spam)
 - $y' = f(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$
 - \mathbf{w} is a vector of the same size as \mathbf{x}
 - One weight per dimension of \mathbf{x} (i.e. one weight per word)
 - Weight can be positive, zero, negative...

Simple strategy: Let's count!

This is X

$$\begin{pmatrix} \text{free} & 100 \\ \text{money} & 2 \\ \vdots & \vdots \\ \text{account} & 2 \\ \vdots & \vdots \end{pmatrix}$$

This is Y



= 1 or 0?

nadia bamba

To: undisclosed recipients ;


Reply-To: nadia bamba

From Miss Nadia BamBa,

From Miss Nadia BamBa,

Greeting, Permit me to inform you of my desire of going i
Nadia BamBa the only Daughter of late Mr and Mrs Jame
cocoa merchant in Abidjan, the economic capital of Ivory
his business associates on one of their outing to discus c
on the 21st October 2002, my father took me very speci

Before the death of my father in a private hospital here in
bedside and told me that he had a sum of \$6, 8000.000(S
Dollars) left in a suspense account in a Bank here in Abic
Daughter for the next of kin in deposit of the fund.

Sebring, Tracy 

To: Batra, Dhruv

ECE 4424 proposal

CUSP has approved ECE 4424 with the following changes: Can
copy of the proposal with these items addressed? (see below)

Thanks!!!

Tracy

$$\begin{pmatrix} \text{free} & 1 \\ \text{money} & 1 \\ \vdots & \vdots \\ \text{account} & 2 \\ \vdots & \vdots \end{pmatrix}$$

Weigh counts and sum to get prediction

nadia bamba

To: undisclosed recipients ;


Reply-To: nadia bamba

From Miss Nadia BamBa,

From Miss Nadia BamBa,

Greeting, Permit me to inform you of Nadia BamBa the only Daughter of k cocoa merchant in Abidjan, the econ his business associates on one of th on the 21st October 2002, my father

Before the death of my father in a p bedside and told me that he had a su Dollars) left in a suspense account in Daughter for the next of kin in depos


$$\begin{pmatrix} 100 \times 0.2 \\ 2 \times 0.3 \\ \vdots \\ 2 \times 0.3 \\ \vdots \end{pmatrix}$$

$$\begin{pmatrix} \text{free} & 100 \\ \text{money} & 2 \\ \vdots & \vdots \\ \text{account} & 2 \\ \vdots & \vdots \end{pmatrix}$$

This is a *linear classifier*

Machine Learning Overview

- Example:
 - Apply a prediction function to an image to get the desired label output:

$$f(\text{apple image}) = \text{"apple"}$$

$$f(\text{tomato image}) = \text{"tomato"}$$

$$f(\text{cow image}) = \text{"cow"}$$

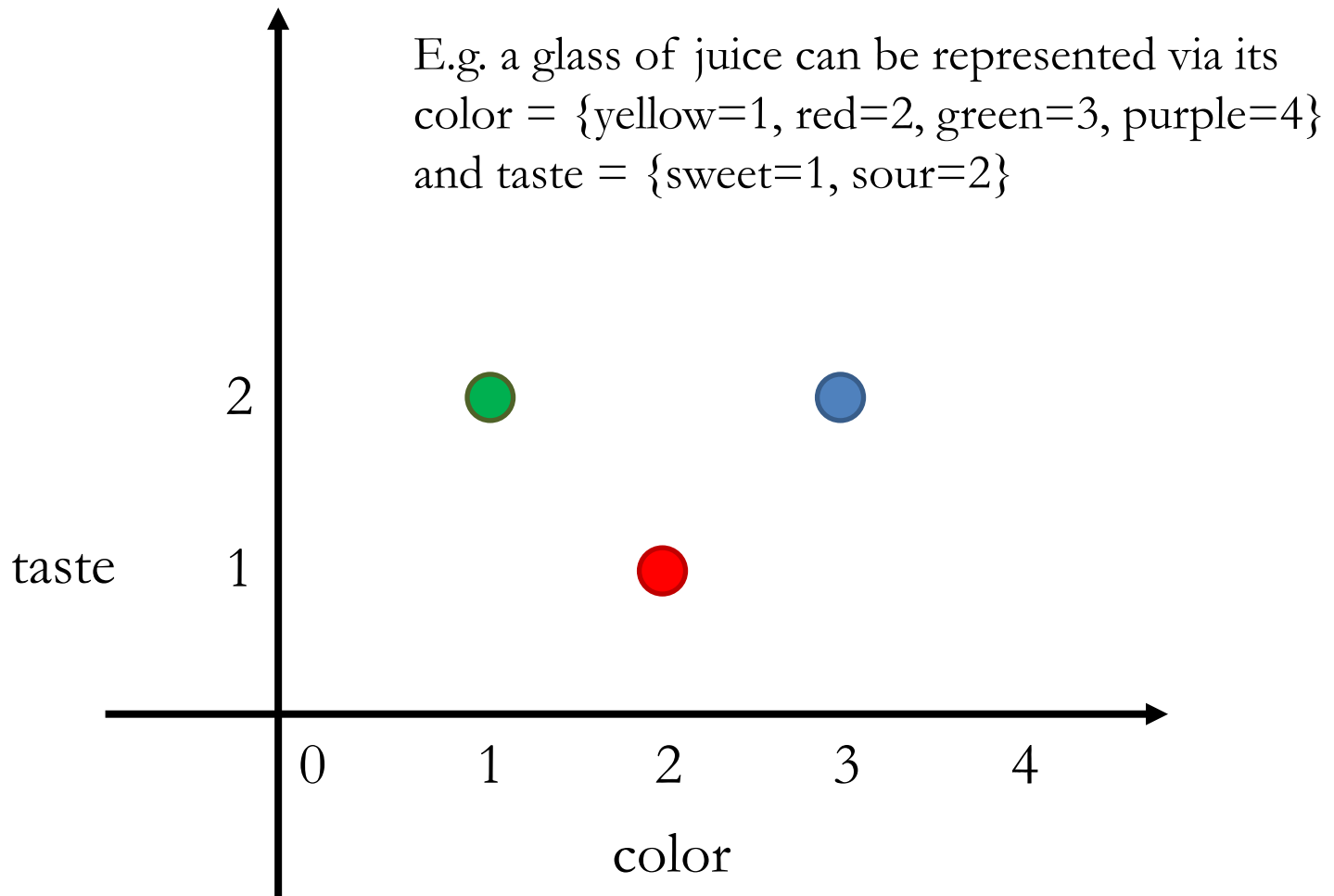
Machine Learning Overview

- Example:
 - \mathbf{x} = pixels of the image (concatenated to form a vector)
 - y = integer (1 = apple, 2 = tomato, etc.)
 - $y' = f(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$
 - \mathbf{w} is a vector of the same size as \mathbf{x}
 - One weight per each dimension of \mathbf{x} (i.e. one weight per pixel)

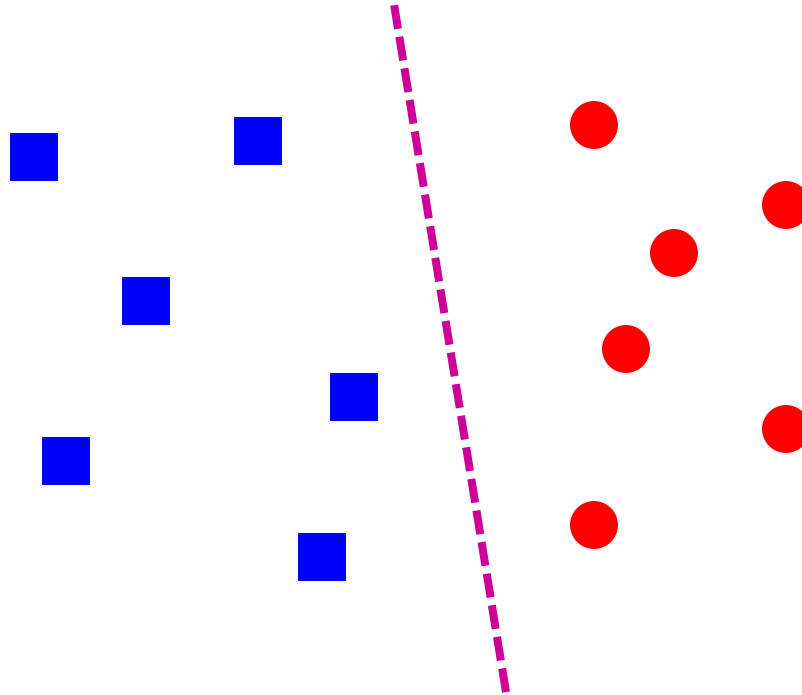
Feature representation (\mathbf{x})

- A vector representing measurable characteristics of a data sample we have
- E.g. a glass of juice can be represented via its color = {yellow=1, red=2, green=3, purple=4} and taste = {sweet=1, sour=2}
- For a given glass i , this can be represented as a vector: $\mathbf{x}_i = [3 \ 2]$ represents sour green juice
- For D features, this defines a D -dimensional space where we can measure similarity between samples

Feature representation (\mathbf{x})



Linear classifier

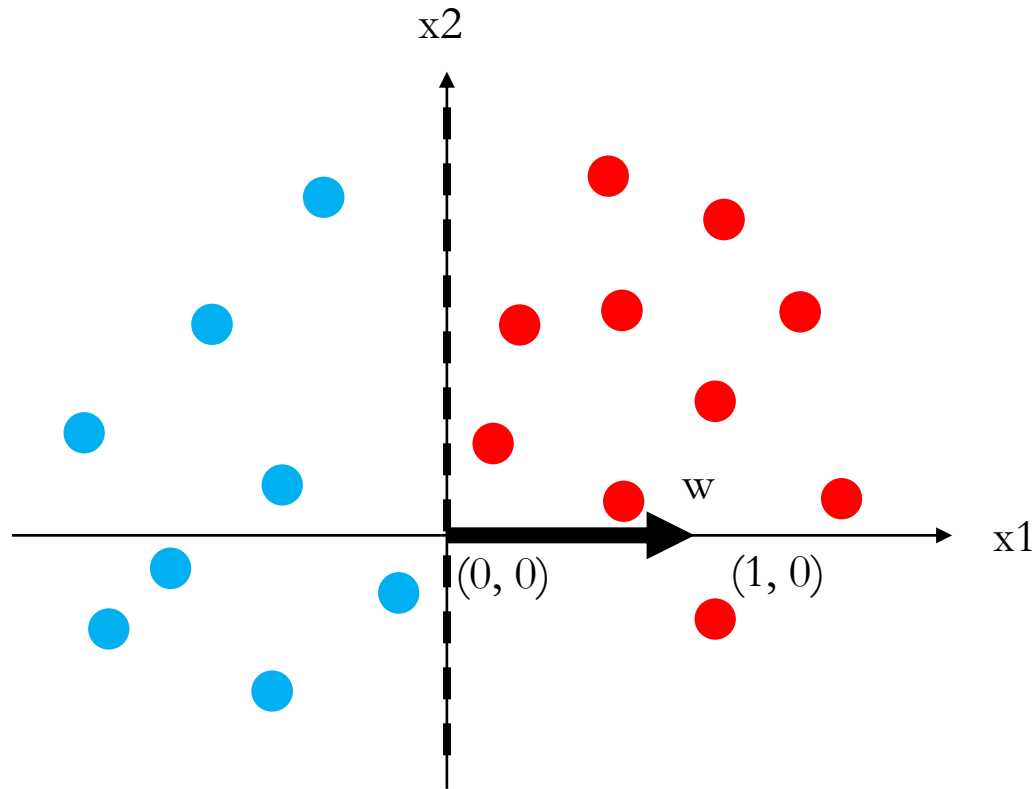


- Find a *linear function* to separate the classes

$$f(\mathbf{x}) = \text{sgn}(w_1x_1 + w_2x_2 + \dots + w_Dx_D) = \text{sgn}(\mathbf{w}^T\mathbf{x})$$

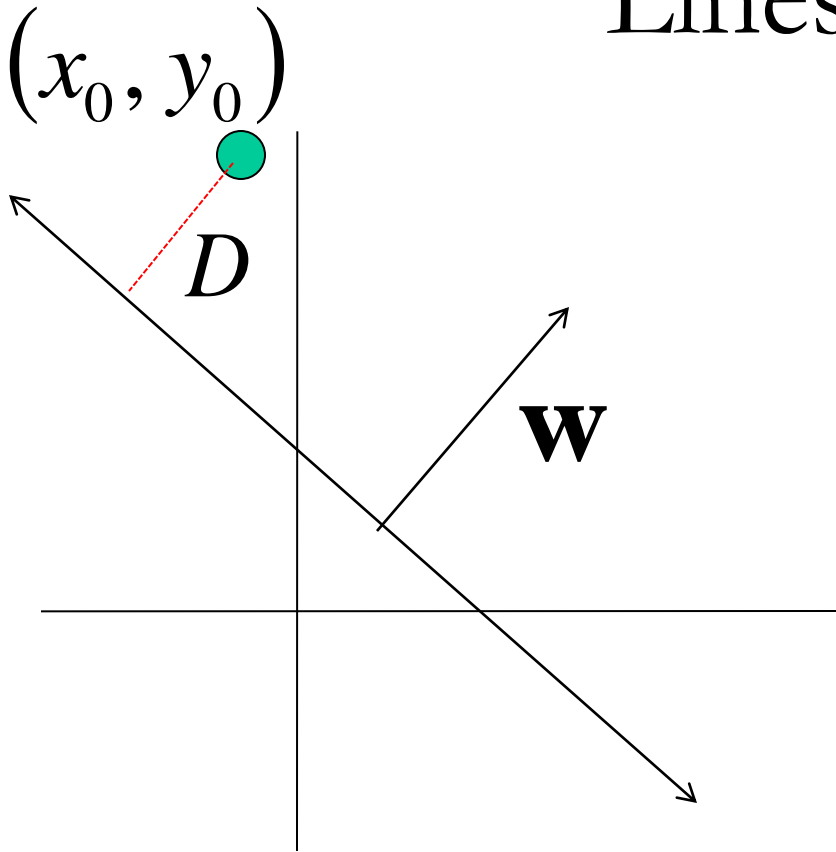
Linear classifier

- Decision = $\text{sign}(\mathbf{w}^T \mathbf{x}) = \text{sign}(w_1 * x_1 + w_2 * x_2)$



- What could the weights be?

Lines in \mathbb{R}^2



Let $\mathbf{w} = \begin{bmatrix} a \\ c \end{bmatrix}$ $\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}$

$$ax + cy + b = 0$$

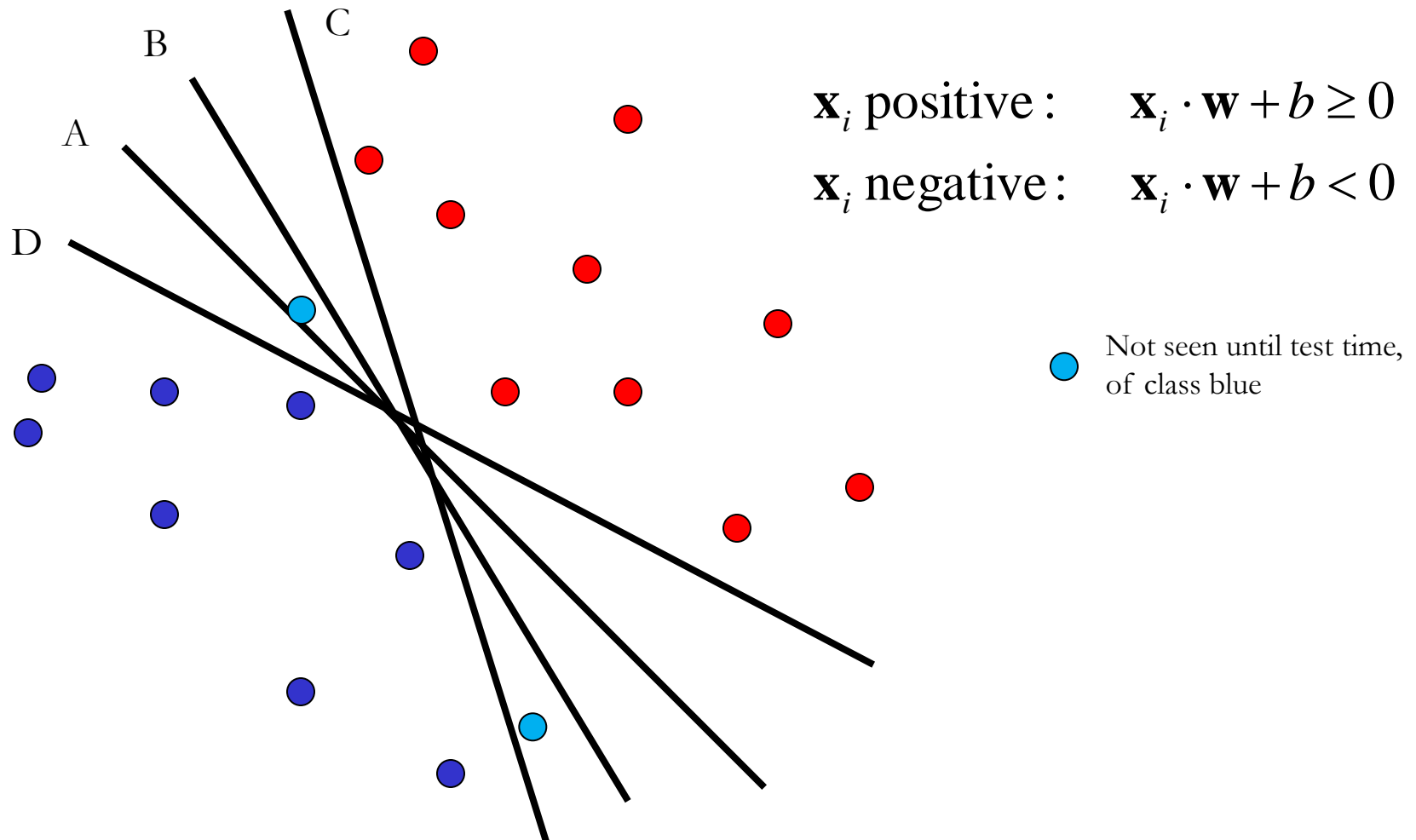


$$\mathbf{w} \cdot \mathbf{x} + b = 0$$

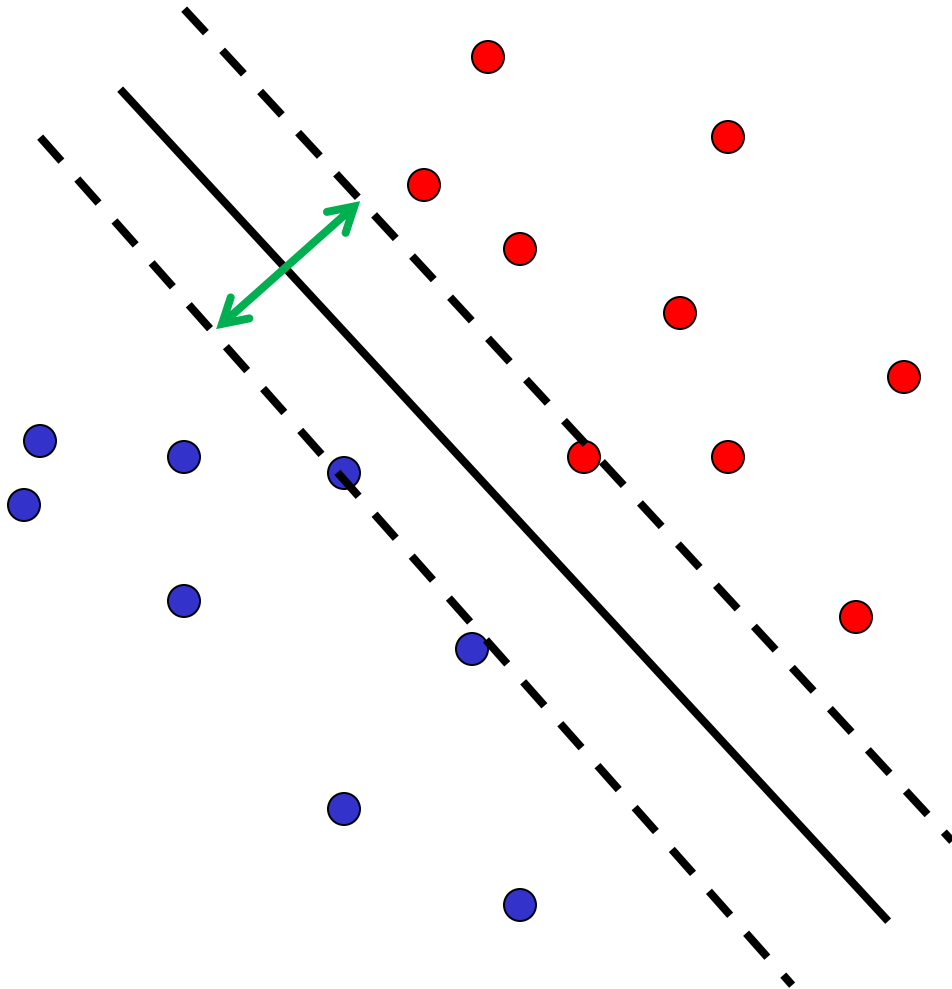
$$D = \frac{|ax_0 + cy_0 + b|}{\sqrt{a^2 + c^2}} = \frac{|\mathbf{w}^T \mathbf{x} + b|}{\|\mathbf{w}\|} \left. \vphantom{\frac{|\mathbf{w}^T \mathbf{x} + b|}{\|\mathbf{w}\|}} \right\} \begin{array}{l} \text{distance from} \\ \text{point to line} \end{array}$$

Linear classifiers

- Find linear function to separate positive and negative examples



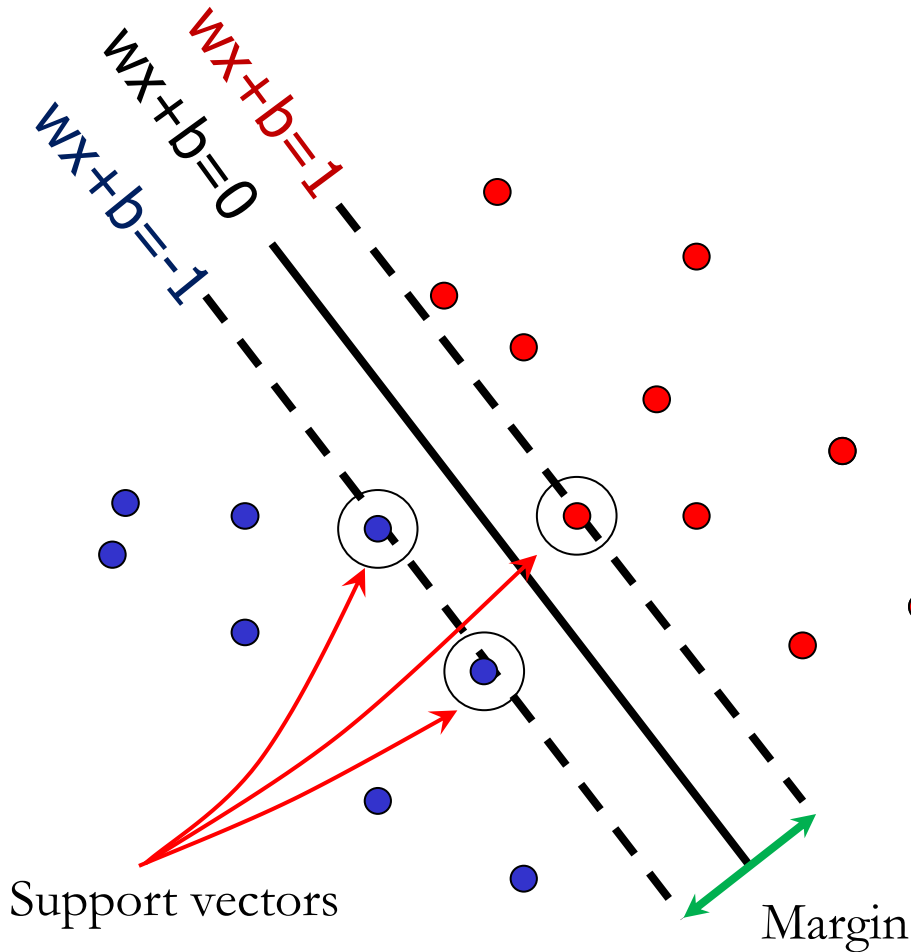
Support vector machines



- Discriminative classifier based on *optimal separating line (for 2d case)*
- Maximize the *margin* between the positive and negative training examples

Support vector machines

- Want line that maximizes the margin.



$$\mathbf{x}_i \text{ positive } (y_i = 1): \quad \mathbf{x}_i \cdot \mathbf{w} + b \geq 1$$

$$\mathbf{x}_i \text{ negative } (y_i = -1): \quad \mathbf{x}_i \cdot \mathbf{w} + b \leq -1$$

$$\text{For support vectors,} \quad \mathbf{x}_i \cdot \mathbf{w} + b = \pm 1$$

$$\text{Distance between point and line:} \quad \frac{|\mathbf{x}_i \cdot \mathbf{w} + b|}{\|\mathbf{w}\|}$$

For support vectors:

$$\frac{\mathbf{w}^T \mathbf{x} + b}{\|\mathbf{w}\|} = \frac{\pm 1}{\|\mathbf{w}\|} \quad M = \left| \frac{1}{\|\mathbf{w}\|} - \frac{-1}{\|\mathbf{w}\|} \right| = \frac{2}{\|\mathbf{w}\|}$$

Finding the maximum margin line

1. Maximize margin $2/\|\mathbf{w}\|$
2. Correctly classify all training data points:

$$\mathbf{x}_i \text{ positive } (y_i = 1): \quad \mathbf{x}_i \cdot \mathbf{w} + b \geq 1$$

$$\mathbf{x}_i \text{ negative } (y_i = -1): \quad \mathbf{x}_i \cdot \mathbf{w} + b \leq -1$$

Quadratic optimization problem:

$$\text{Minimize} \quad \frac{1}{2} \mathbf{w}^T \mathbf{w}$$

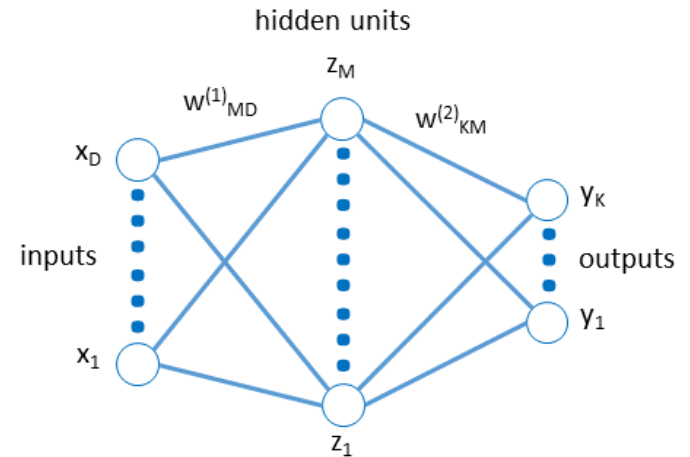
$$\text{Subject to } y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1$$

One constraint for each training point.

Note sign trick.

Deep Learning in a Nutshell

- Input \rightarrow network \rightarrow outputs
- Input X is raw (e.g. raw image, one-hot representation of text)
- Network extracts features: abstraction of input (e.g. not pixels, but edges)
- Output is the labels Y
- All parameters of the network *learned* (during *training*) by checking how well predicted/true Y agree, using labels in the training set



Elements of Machine Learning

- Every machine learning algorithm has:
 - Data representation (x, y)
 - Problem representation (network)
 - Evaluation / objective function
 - Optimization (solve for parameters of network)

Data representation

- Let's brainstorm what our “X” should be for various “Y” prediction tasks...
- Weather prediction?
- Movie ratings predictions?

Problem representation

- Instances
- Decision trees
- Sets of rules / Logic programs
- Support vector machines
- Graphical models (Bayes/Markov nets)
- **Neural networks**
- Model ensembles
- Etc.

Evaluation / objective function

- Accuracy
- Precision and recall
- Squared error
- Likelihood
- Posterior probability
- Cost / Utility
- Margin
- Entropy
- K-L divergence
- Etc.

Loss functions

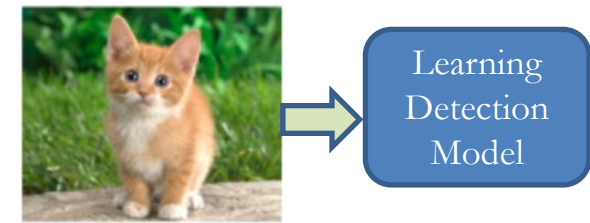
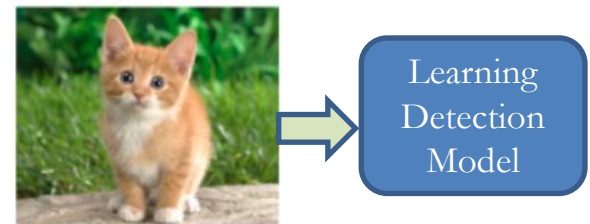
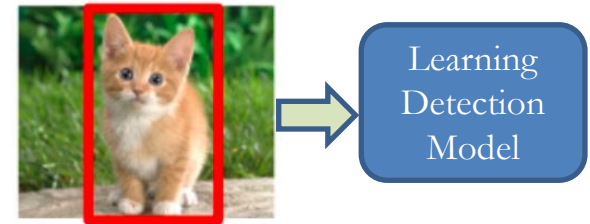
- Measure error
- Can be defined for discrete or continuous outputs
- E.g. if task is classification – could use cross-entropy loss $-\sum_{c=1}^M y_{o,c} \log(p_{o,c})$
- If task is regression – use L2 loss i.e. $||y-y'||$

Optimization

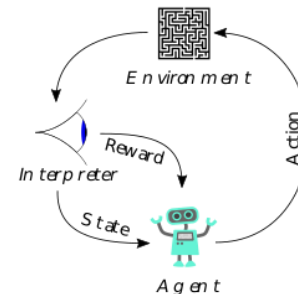
- Optimization means we need to solve for the parameters \mathbf{w} of the model
- For a (non-linear) neural network, there is no closed-form solution to solve for \mathbf{w} ; cannot set up linear system with \mathbf{w} as the unknowns
- Thus, optimization solutions look like this:
 1. Initialize \mathbf{w} (e.g. randomly)
 2. Check error (ground-truth vs predicted labels on training set) under current model
 3. Use gradient (derivative) of error wrt \mathbf{w} to update \mathbf{w}
 4. Repeat from 2 until convergence

Types of Learning

- Supervised learning
 - Training data includes desired outputs
- Unsupervised learning
 - Training data does not include desired outputs
- Weakly supervised learning
 - Training data includes a few desired outputs, or contains labels that only approximate the labels desired at test time (noisy)
- Reinforcement learning
 - Rewards from sequence of actions



CAT



Validation strategies

- Ultimately, for our application, what do we want?
 - High accuracy on training data?
 - No, high accuracy on *unseen/new/test data*!
 - Why is this tricky?
- Training data
 - Features (x) and labels (y) used to learn mapping f
- Test data
 - Features used to make a prediction
 - Labels only used to see how well we've learned f !!!
- Validation data
 - Held-out set of the *training data*
 - Can use both features and labels to tune model *hyperparameters*
 - *Hyperparameters* are “knobs” of the algorithm tuned by the designer: number of iterations for learning, learning rate, etc.
 - We train multiple model (one per hyperparameter setting) and choose the best one, on the validation set

Validation strategies

Idea #1: Split data into **train**, **val**, and **test**; choose hyperparameters on val and evaluate on test

Better!

train	validation	test
-------	------------	------

Idea #2: Cross-Validation: Split data into **folds**, try each fold as validation and average the results

fold 1	fold 2	fold 3	fold 4	fold 5	test
fold 1	fold 2	fold 3	fold 4	fold 5	test
fold 1	fold 2	fold 3	fold 4	fold 5	test

Useful for small datasets, but not used too frequently in deep learning

Generalization



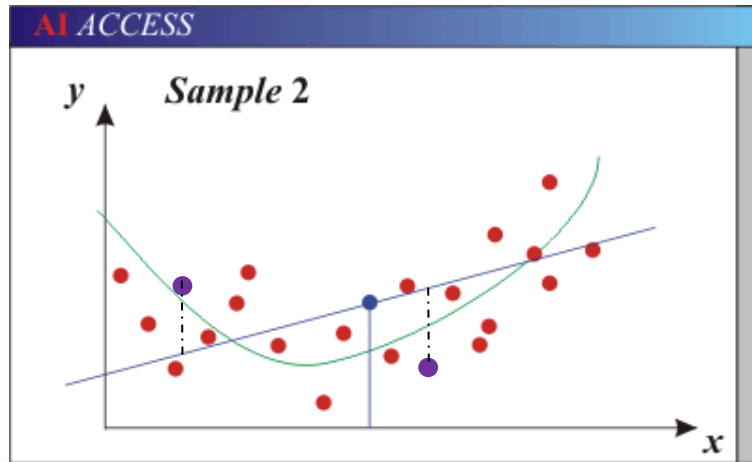
Training set (labels known)



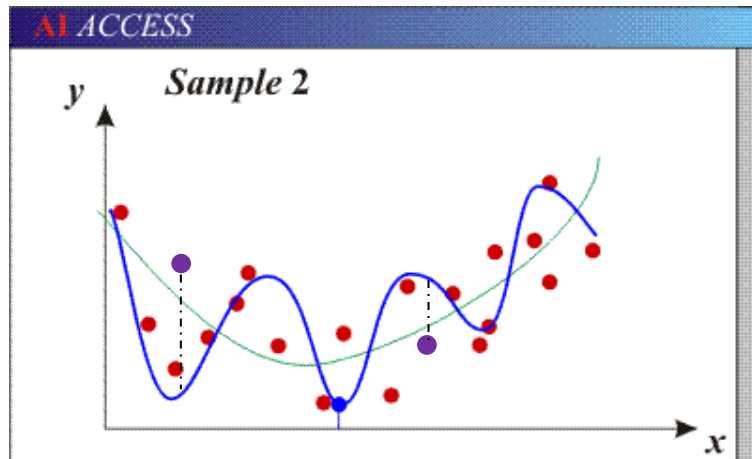
Test set (labels unknown)

- How well does a learned model generalize from the data it was trained on to a new test set?

Generalization



- Underfitting: Models with too few parameters are inaccurate because of a large bias (not enough flexibility).



- Overfitting: Models with too many parameters are inaccurate because of a large variance (too much sensitivity to the sample).

Purple dots = possible test points

Red dots = training data (all that we see before we ship off our model!)

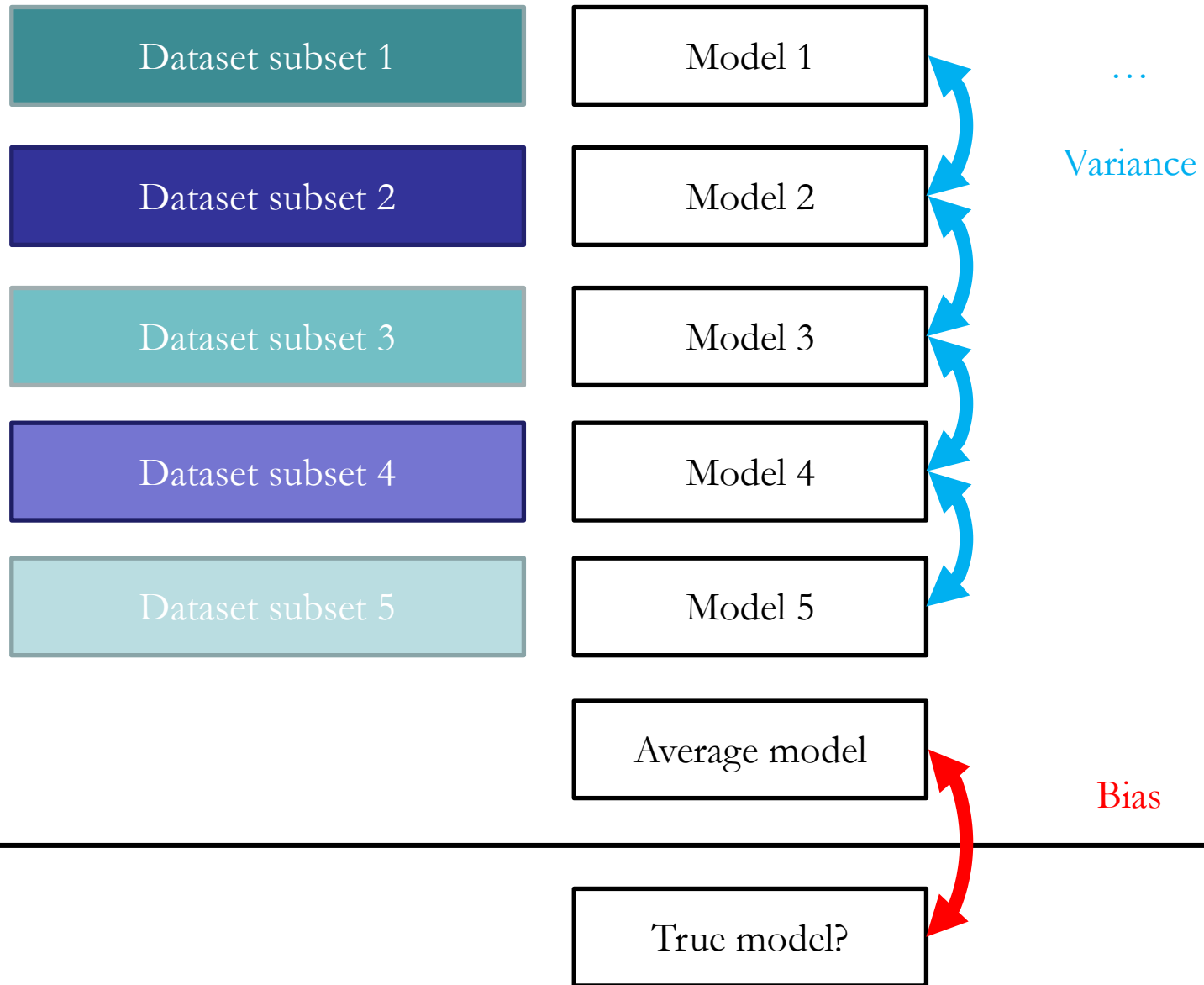
Green curve = true underlying model

Blue curve = our predicted model/fit

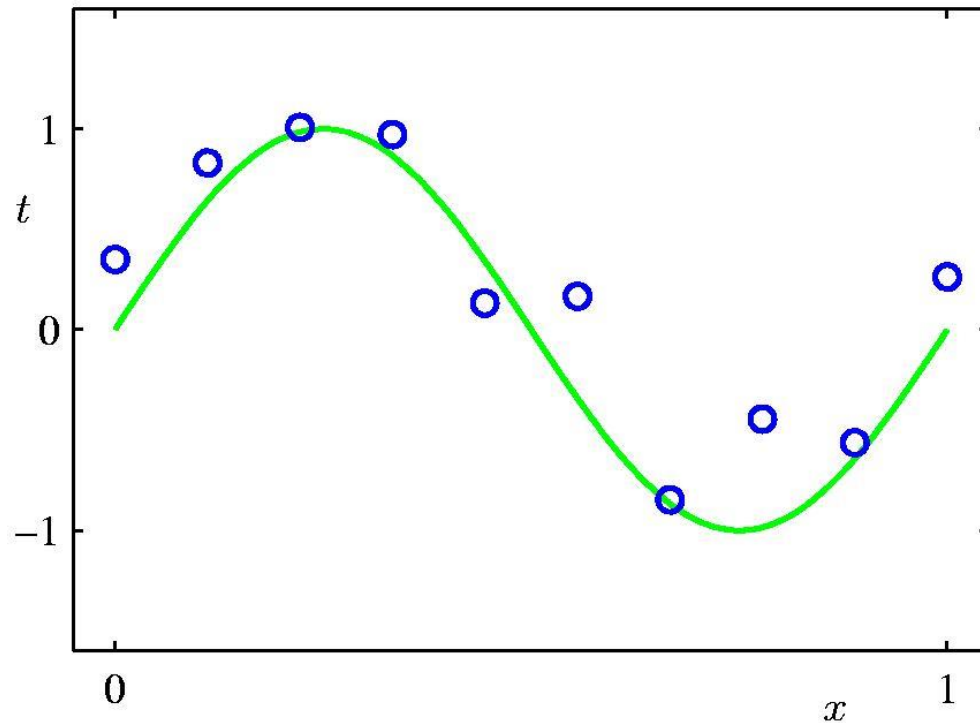
Generalization

- Components of generalization error
 - **Noise** in our observations: unavoidable
 - **Bias**: due to inaccurate assumptions/simplifications by model
 - **Variance**: models estimated from different training sets differ greatly from each other
- **Underfitting**: model is too “simple” to represent all the relevant class characteristics
 - High bias and low variance
 - High training error and high test error
- **Overfitting**: model is too “complex” and fits irrelevant characteristics (noise) in the data
 - Low bias and high variance
 - Low training error and high test error

Generalization

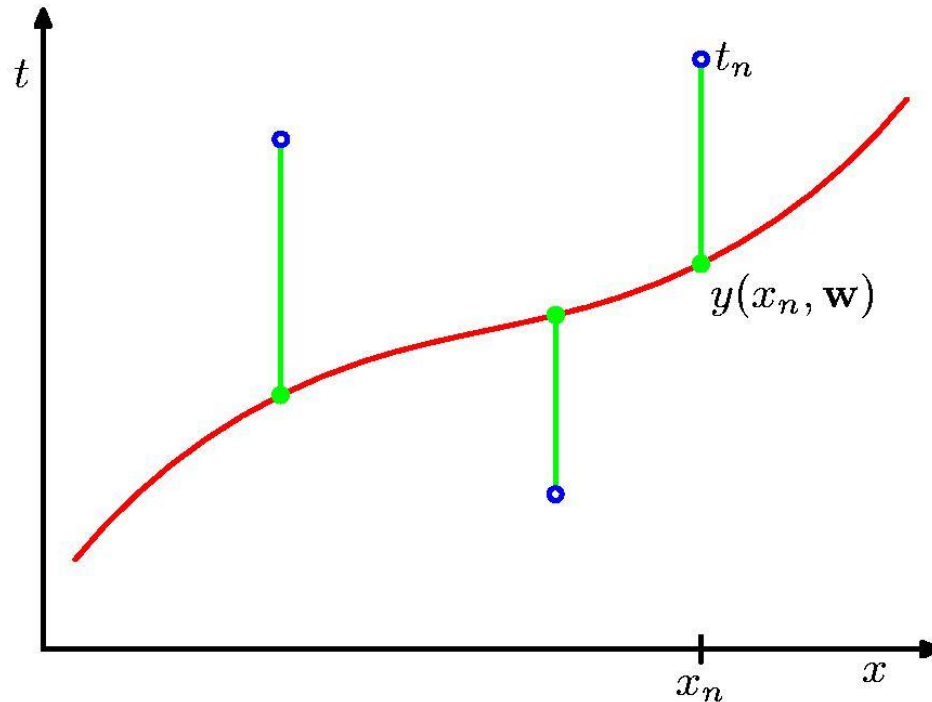


Polynomial Curve Fitting



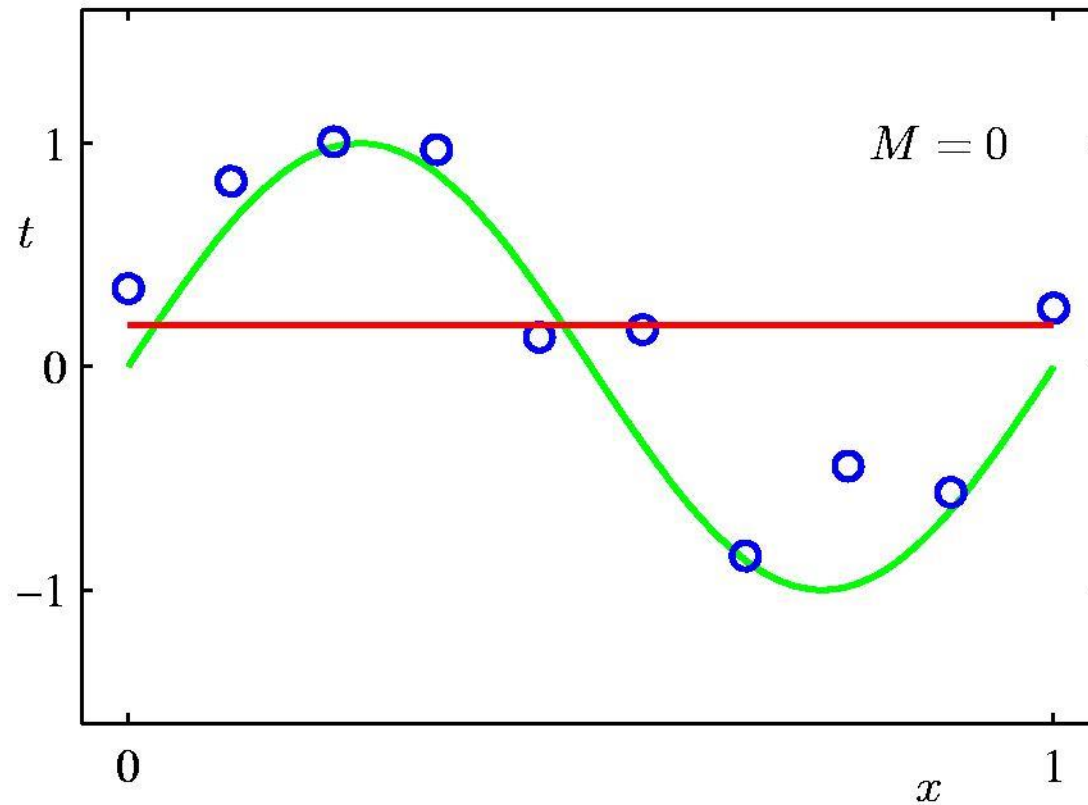
$$y(x, \mathbf{w}) = w_0 + w_1x + w_2x^2 + \dots + w_Mx^M = \sum_{j=0}^M w_jx^j$$

Sum-of-Squares Error Function

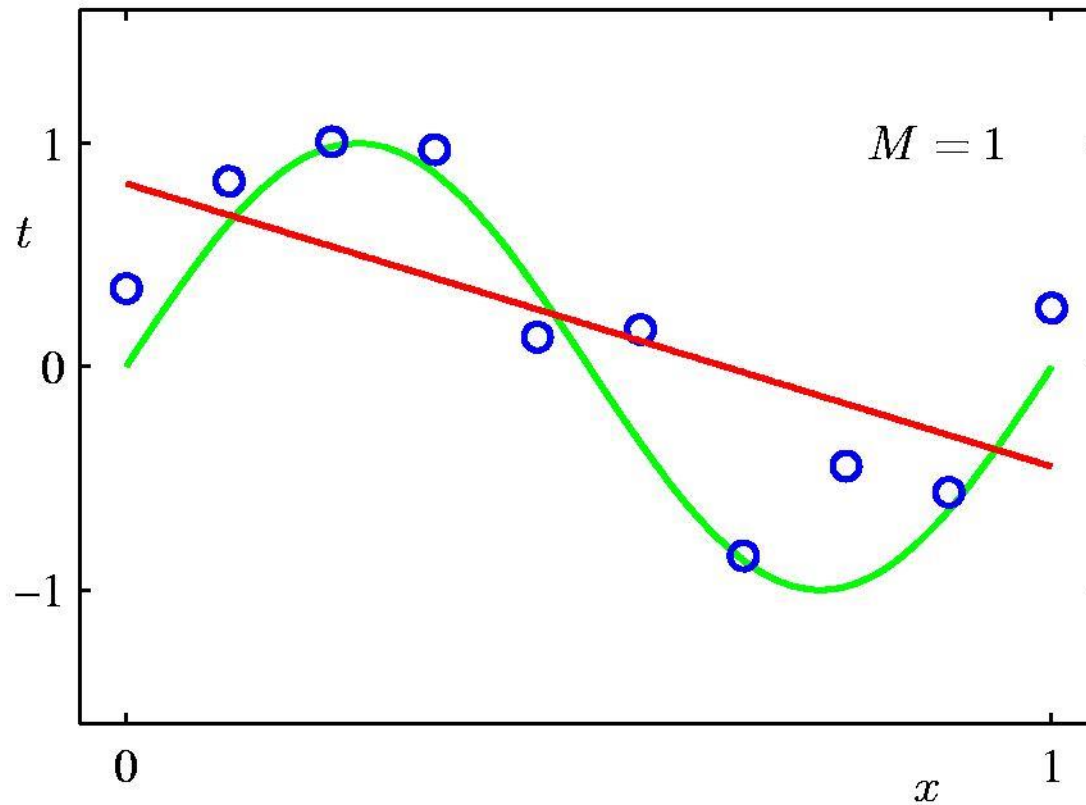


$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2$$

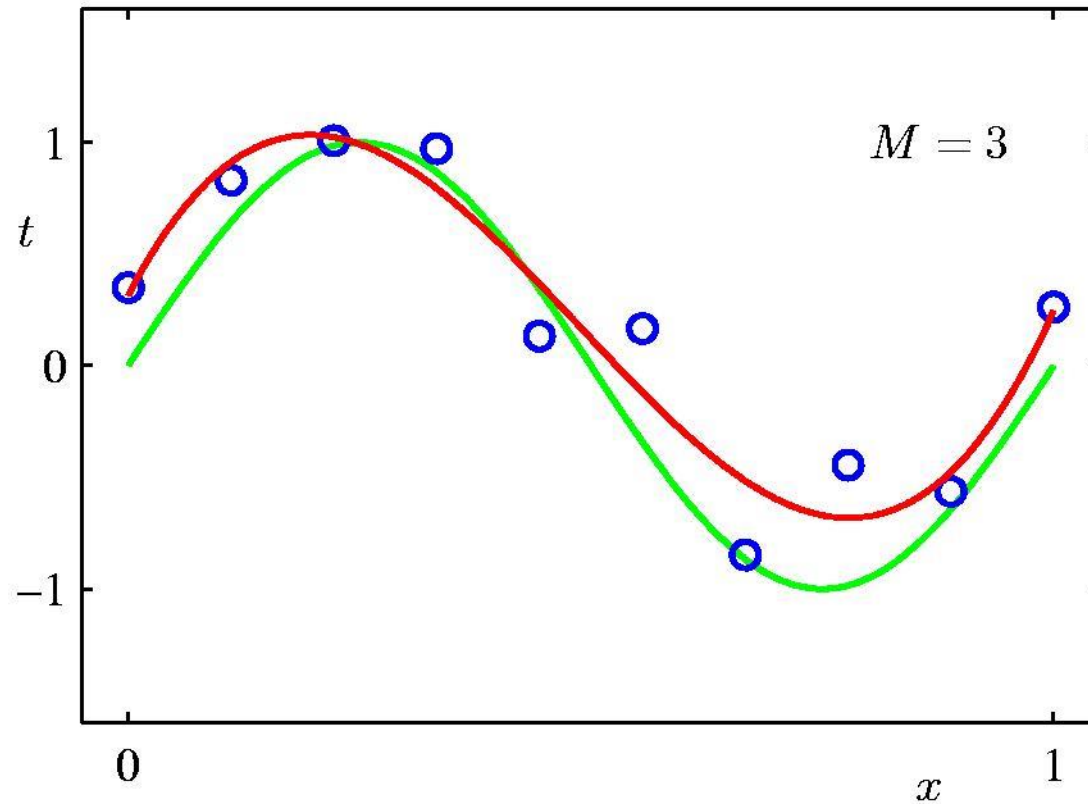
0th Order Polynomial



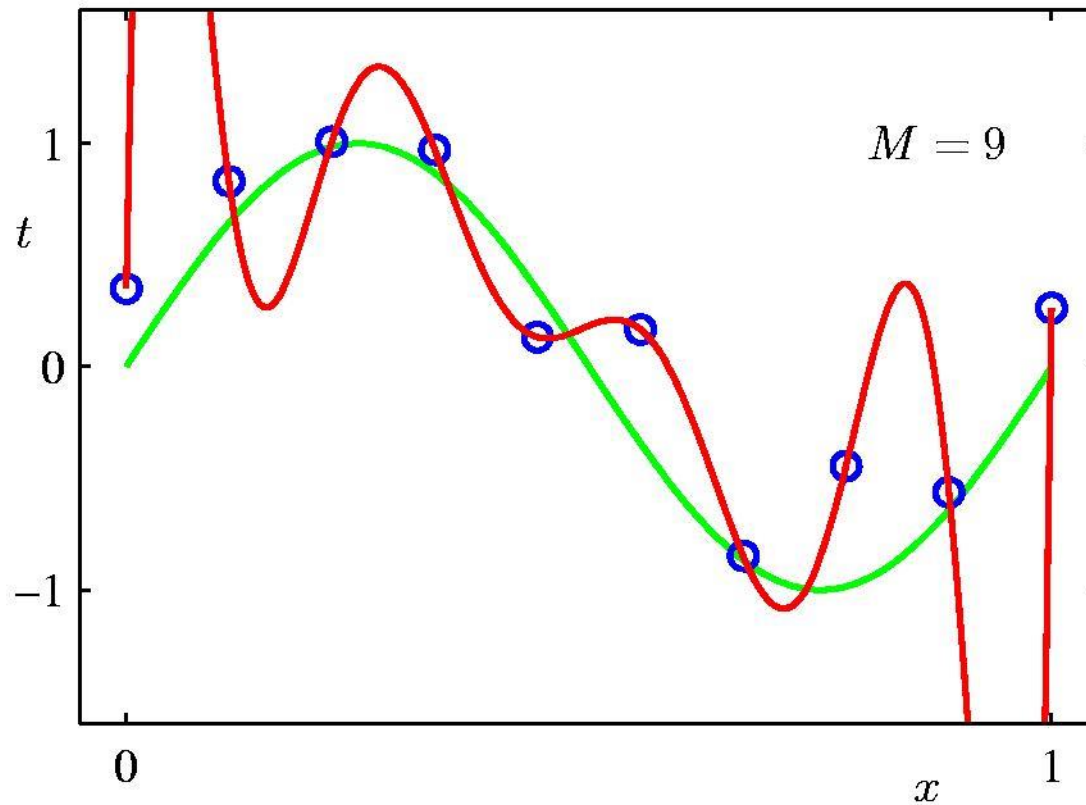
1st Order Polynomial



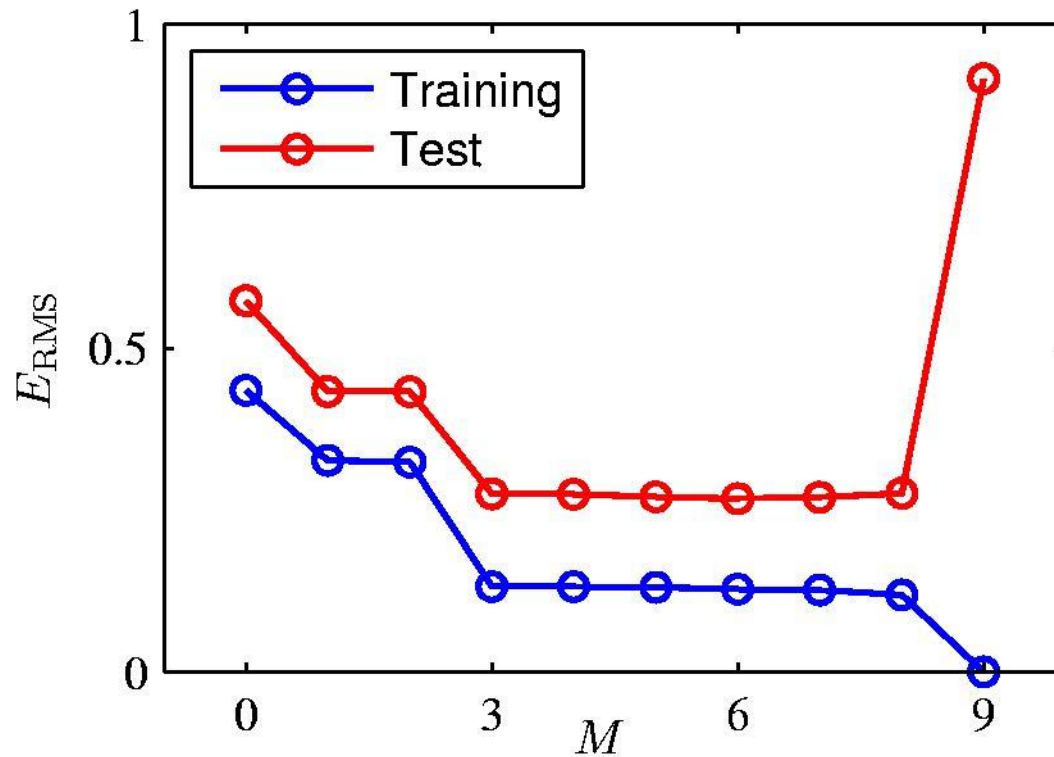
3rd Order Polynomial



9th Order Polynomial



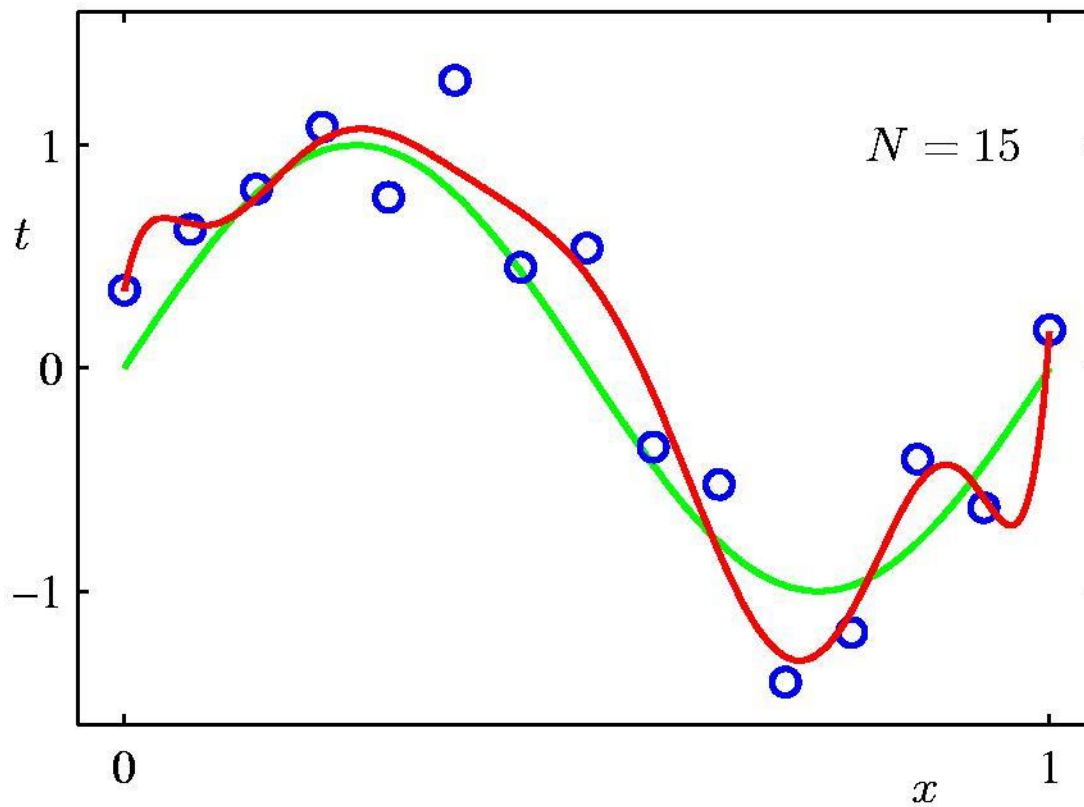
Over-fitting



Root-Mean-Square (RMS) Error: $E_{\text{RMS}} = \sqrt{2E(\mathbf{w}^*)/N}$

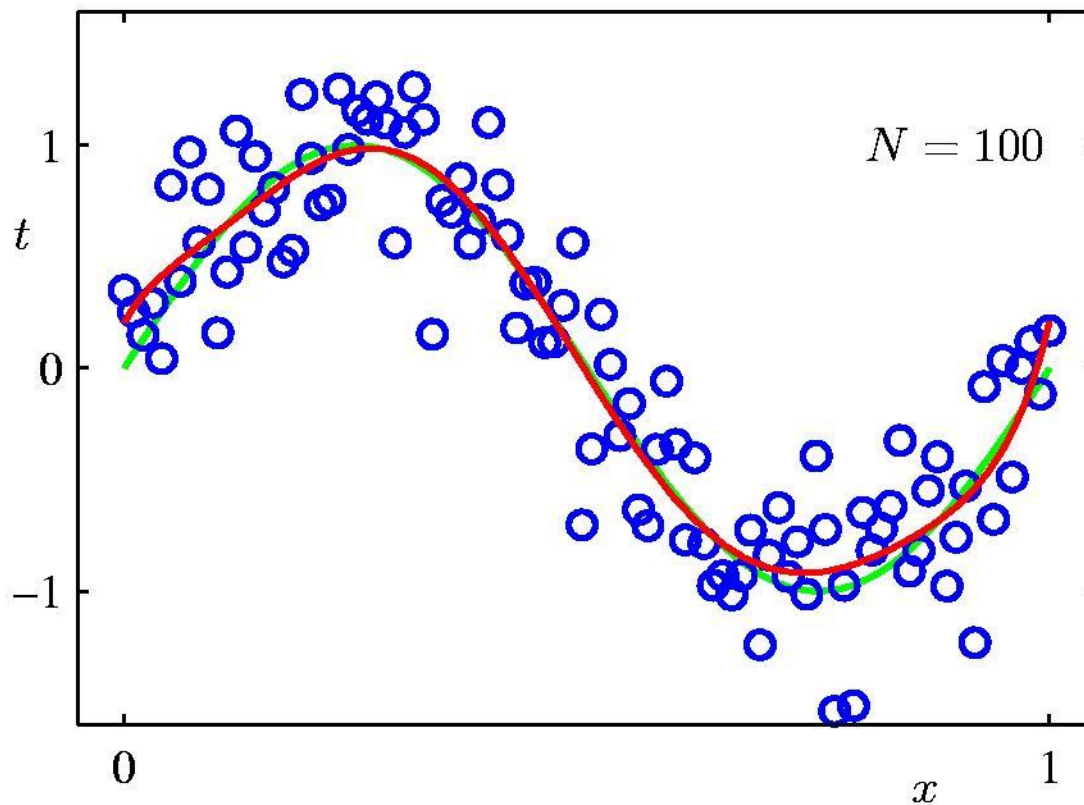
Data Set Size: $N = 15$

9th Order Polynomial



Data Set Size: $N = 100$

9th Order Polynomial



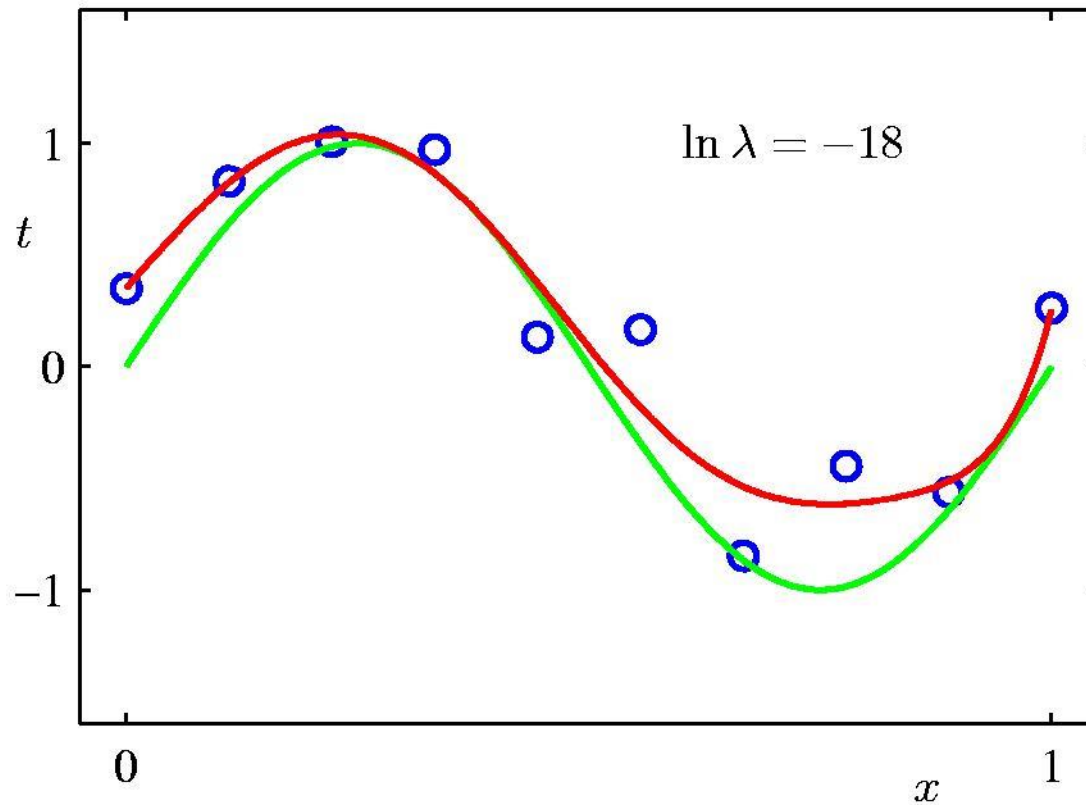
Regularization

Penalize large coefficient values

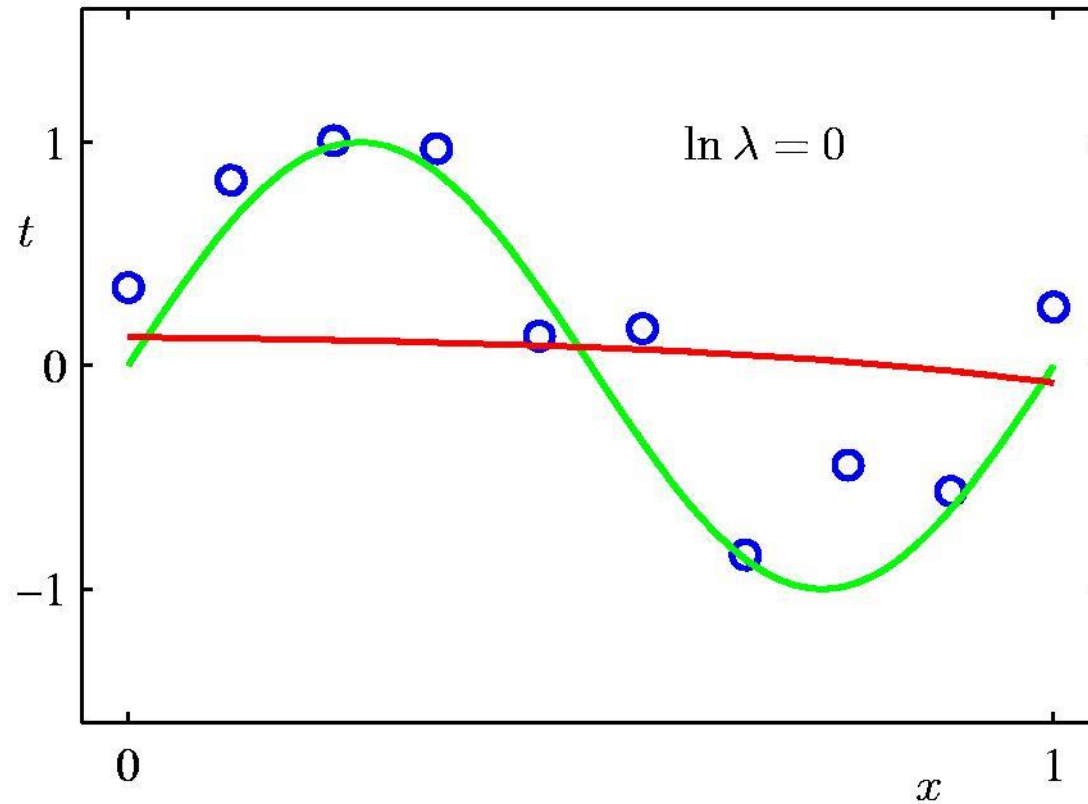
$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

(Remember: We want to minimize this expression.)

Regularization: $\ln \lambda = -18$



Regularization: $\ln \lambda = 0$



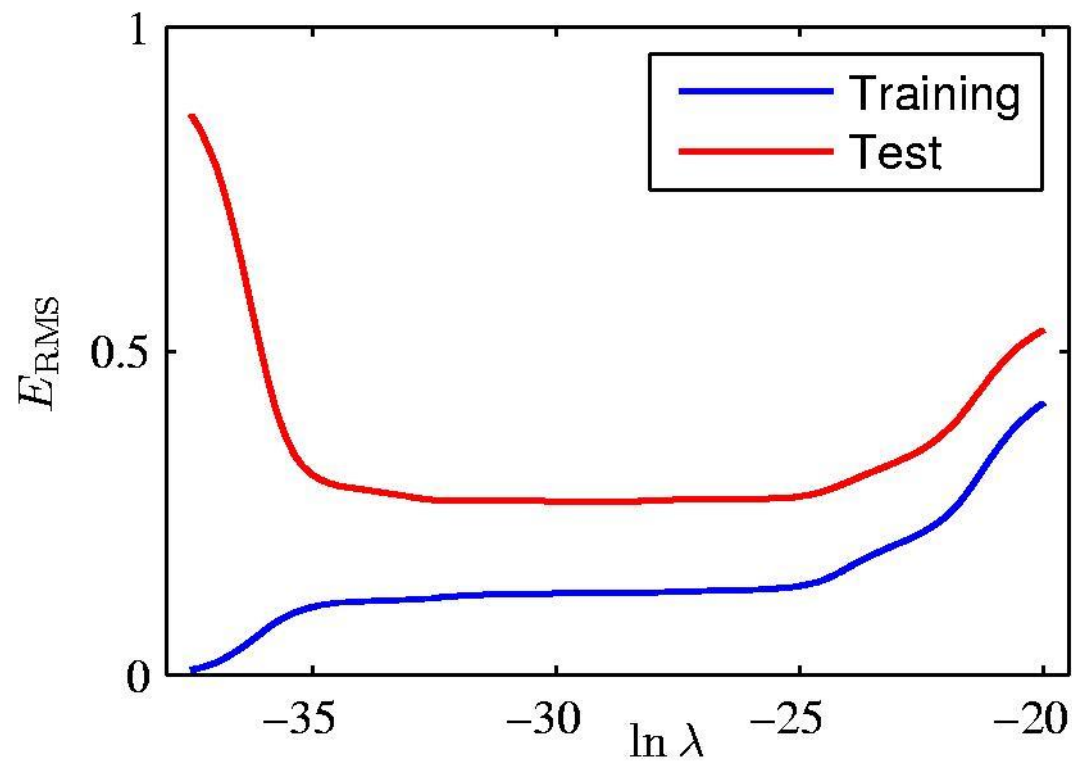
Polynomial Coefficients

	$M = 0$	$M = 1$	$M = 3$	$M = 9$
w_0^*	0.19	0.82	0.31	0.35
w_1^*		-1.27	7.99	232.37
w_2^*			-25.43	-5321.83
w_3^*			17.37	48568.31
w_4^*				-231639.30
w_5^*				640042.26
w_6^*				-1061800.52
w_7^*				1042400.18
w_8^*				-557682.99
w_9^*				125201.43

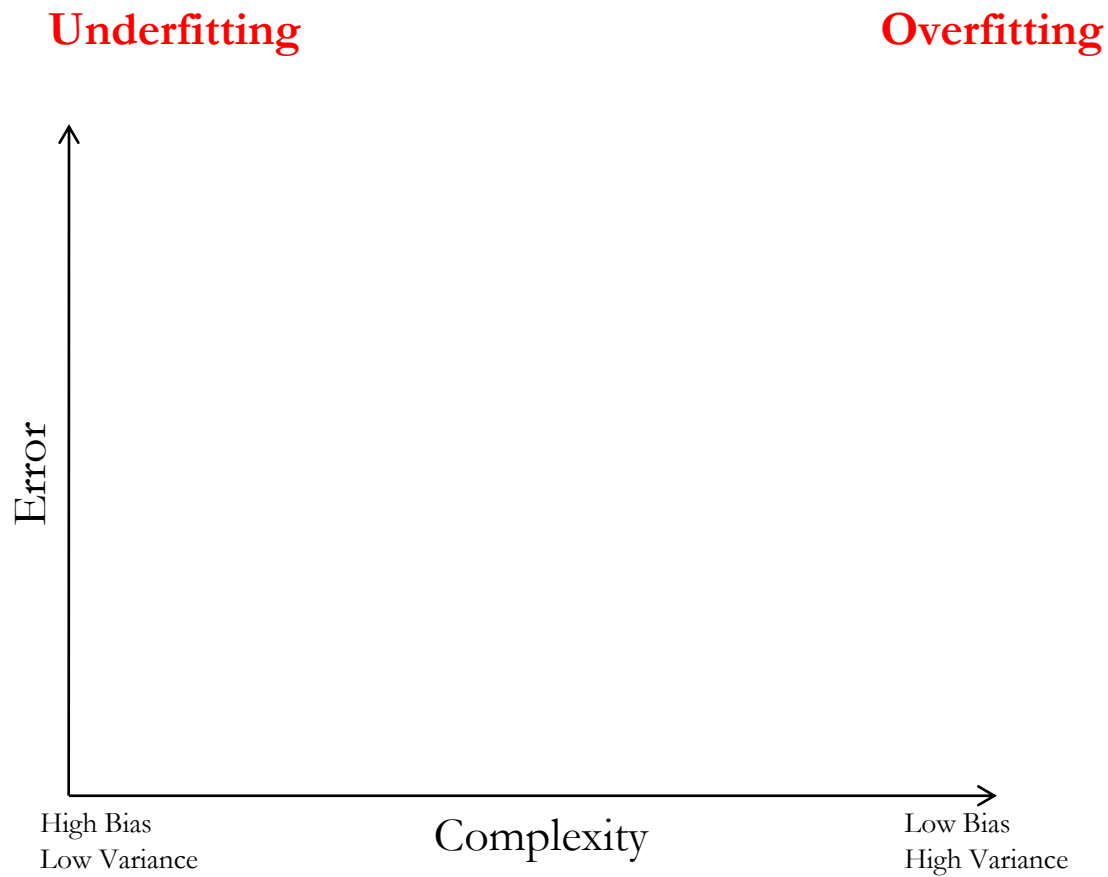
Polynomial Coefficients

	No regularization		Huge regularization
	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
w_0^*	0.35	0.35	0.13
w_1^*	232.37	4.74	-0.05
w_2^*	-5321.83	-0.77	-0.06
w_3^*	48568.31	-31.97	-0.05
w_4^*	-231639.30	-3.89	-0.03
w_5^*	640042.26	55.28	-0.02
w_6^*	-1061800.52	41.32	-0.01
w_7^*	1042400.18	-45.95	-0.00
w_8^*	-557682.99	-91.53	0.00
w_9^*	125201.43	72.68	0.01

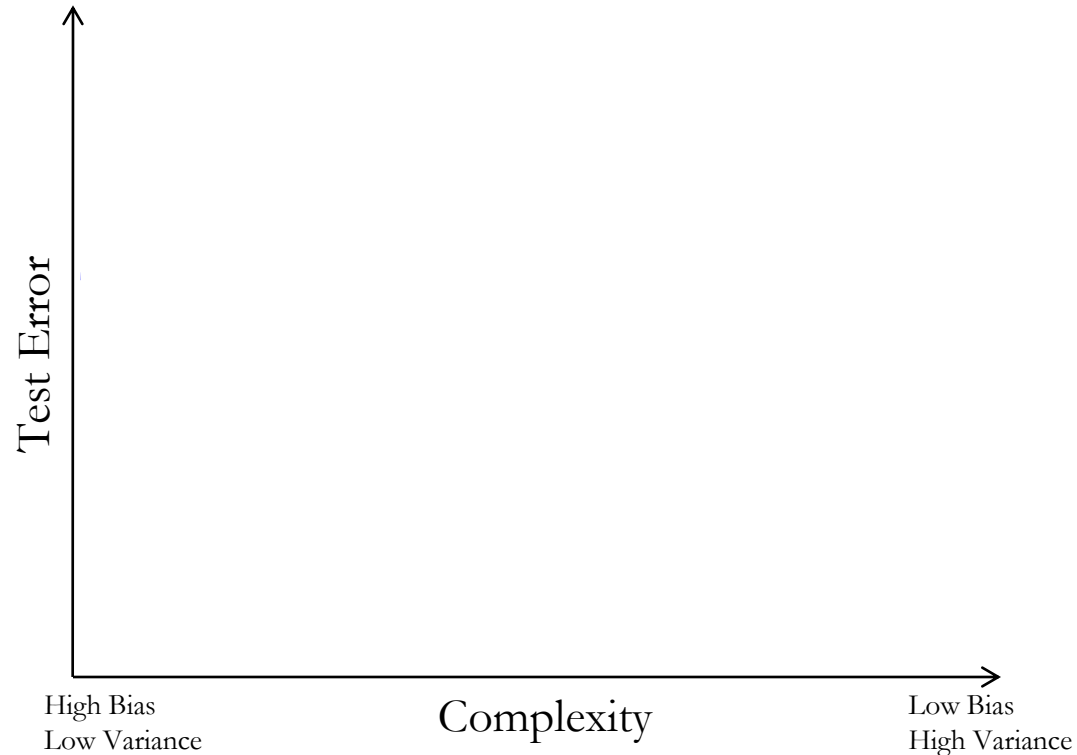
Regularization: E_{RMS} vs. $\ln \lambda$



Training vs test error

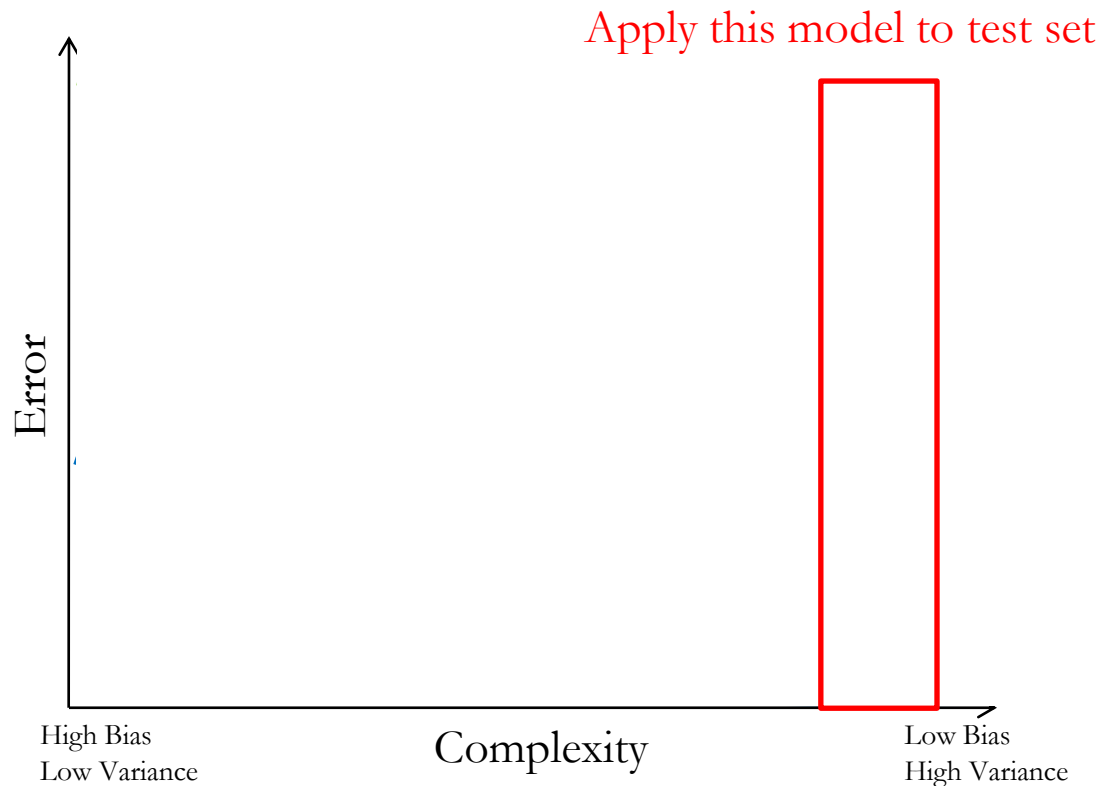


The effect of training set size



Choosing the trade-off between bias and variance

- Need validation set (separate from the test set)



Summary of generalization

- Try simple classifiers first
- Better to have smart features and simple classifiers than simple features and smart classifiers
- Use increasingly powerful classifiers with more training data
- As an additional technique for reducing variance, try regularizing the parameters