

---

# Model Evaluation Measures

---

---

# Metrics for Performance Evaluation

- Focus on the **predictive capability** of a model
  - **Not** how fast it takes to classify or build models, scalability, etc.
- Confusion Matrix:

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

**a: TP (true positive)**

**b: FN (false negative)**

**c: FP (false positive)**

**d: TN (true negative)**

# Metrics for Performance Evaluation...

ACTUAL CLASS	PREDICTED CLASS	
	Class=Yes	Class=No
Class=Yes	a (TP)	b (FN)
	c (FP)	d (TN)

- Most widely-used metric:

$$\text{Accuracy} = \frac{a + d}{a + b + c + d} = \frac{TP + TN}{TP + TN + FP + FN}$$

# Self test

ACTUAL CLASS	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	5	13
	Class=No	12	2005

- What is the accuracy of this classifier, given the confusion matrix?

$$= (5+2005) / (12+13+5+2005)$$

$$= (2010) / (2035)$$

$$= 98.77\%$$

# Limitation of Accuracy

---

- Ignores the **cost** of misclassifications
- Consider an **imbalanced** 2-class problem
  - Number of Class 0 examples = 9990
  - Number of Class 1 examples = 10
- If model predicts everything to be class 0, accuracy is  $9990/10000 = 99.9\%$ 
  - Accuracy is misleading because model does not detect any class 1 example

# Cost Matrix

---

	PREDICTED CLASS		
	$C(i j)$	Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	$C(\text{Yes} \text{Yes})$	$C(\text{No} \text{Yes})$
	Class=No	$C(\text{Yes} \text{No})$	$C(\text{No} \text{No})$

Define a cost function based on your expertise with problem:

$C(i | j)$ : Cost of misclassifying class  $j$  example as class  $i$

# Cost Matrix Examples

Lower cost  
means “better”

Cost Matrix	PREDICTED CLASS		
	$C(i j)$	+	-
	+	-1	100
	-	1	0

*i.e.*, medical  
diagnosis costs?

# Self test: Cost Matrix Examples

Lower cost  
means “better”

Cost Matrix	PREDICTED CLASS		
ACTUAL CLASS	C(i j)	+	-
	+	-1	100
	-	1	0

i.e., medical  
diagnosis costs?

What are the accuracy and cost of these two confusion matrices?  
which classifier is “better” as derived by accuracy and by cost?

Model M <sub>1</sub>	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	150	40
	-	60	250

Accuracy = 80%

Cost = 3910

Model M <sub>2</sub>	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	250	45
	-	5	200

Accuracy = 90%

Cost = 4255



# Cost-Sensitive Measures

ACTUAL CLASS	PREDICTED CLASS		
		Class= Yes	Class= No
	Class= Yes	a	b
	Class= No	c	d

$$\text{Precision (p)} = \frac{a}{a + c} \quad \begin{array}{l} \text{Higher Precision ==} \\ \text{Lower false positives} \end{array}$$

$$\text{Recall (r)} = \frac{a}{a + b} \quad \begin{array}{l} \text{Higher Recall ==} \\ \text{Lower false negatives} \end{array}$$

$$\text{F - measure (F)} = \frac{2rp}{r + p} = \frac{2a}{2a + b + c}$$

- Precision is biased towards
  - $C(p=\text{Yes} \mid a=\text{Yes})$  &  $C(p=\text{Yes} \mid a=\text{No})$
- Recall is biased towards
  - $C(p=\text{Yes} \mid a=\text{Yes})$  &  $C(p=\text{No} \mid a=\text{Yes})$
- F-measure is biased towards all except  $C(p=\text{No} \mid a=\text{No})$

Higher F1 ==  
Lower FN & FP

$$\text{Weighted Accuracy} = \frac{w_1 a + w_4 d}{w_1 a + w_2 b + w_3 c + w_4 d}$$

# Self test: precision versus recall

Does precision or recall make more sense for the following:

Screening for cancer patients

**recall:** fewer false negatives!!

Retrieving similar documents from an online database

**precision:** if we say its similar, it better be

ACTUAL CLASS	PREDICTED CLASS		
		Class= Yes	Class= No
	Class= Yes	a	b
	Class= No	c	d

$$\text{Precision (p)} = \frac{a}{a + c}$$

$$\text{Recall (r)} = \frac{a}{a + b}$$

# Model Evaluation

---

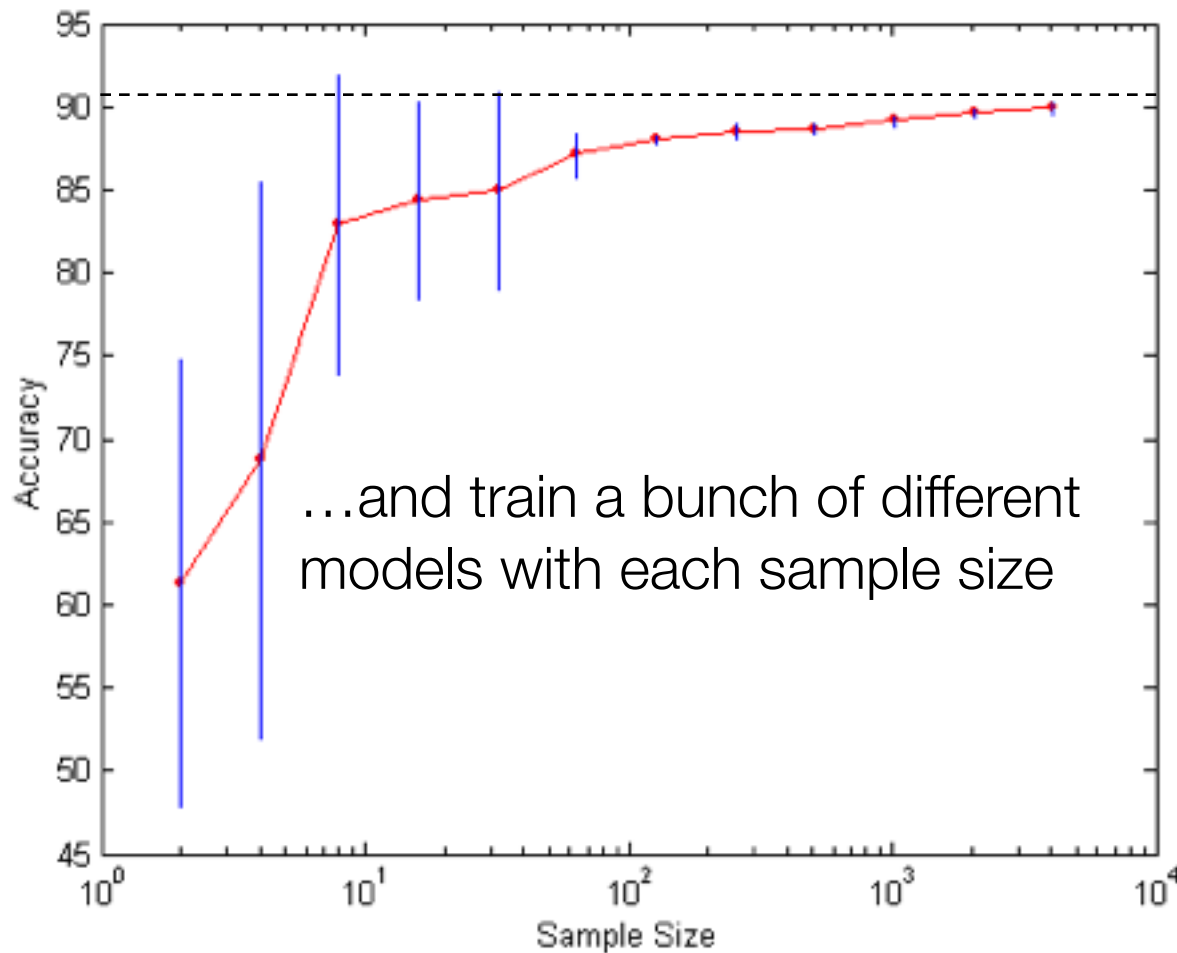
- How reliable are the estimates of performance?

# Methods for Performance Evaluation

---

- How to obtain a reliable estimate of performance?
- Performance of a model may depend on other factors besides the learning algorithm:
  - Class distribution
  - Cost of misclassification
  - Size of training and test sets

# Learning Curve: Number of Samples



Learning curve shows how accuracy changes with varying sample size

Effect of small sample size:

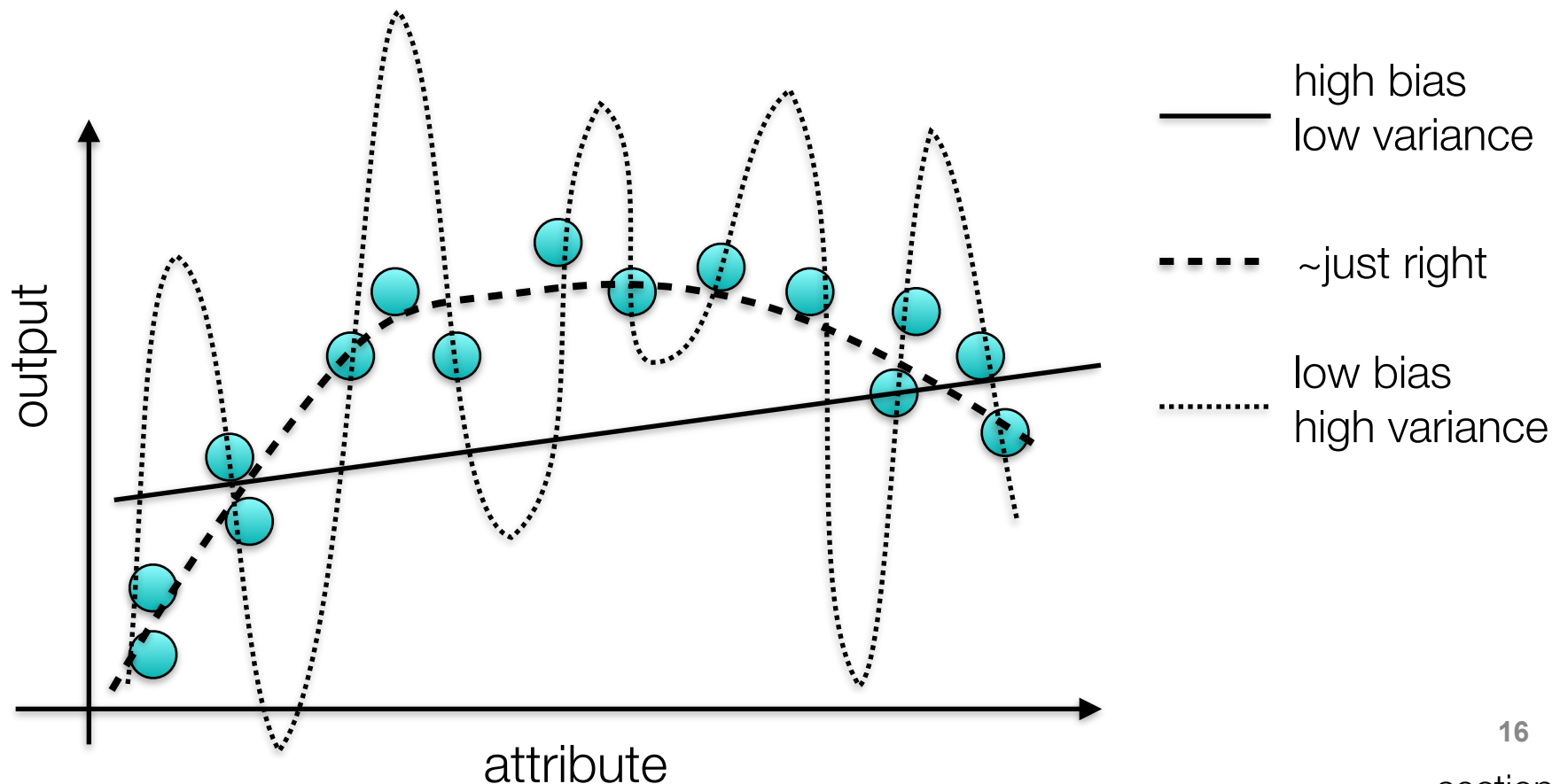
- Bias in the estimate
- Variance of estimate

You cannot estimate this curve without **collecting the data**. Some **bounds exist**, but they are **too loose** to be **useful!!!!**

randomly get this number of samples

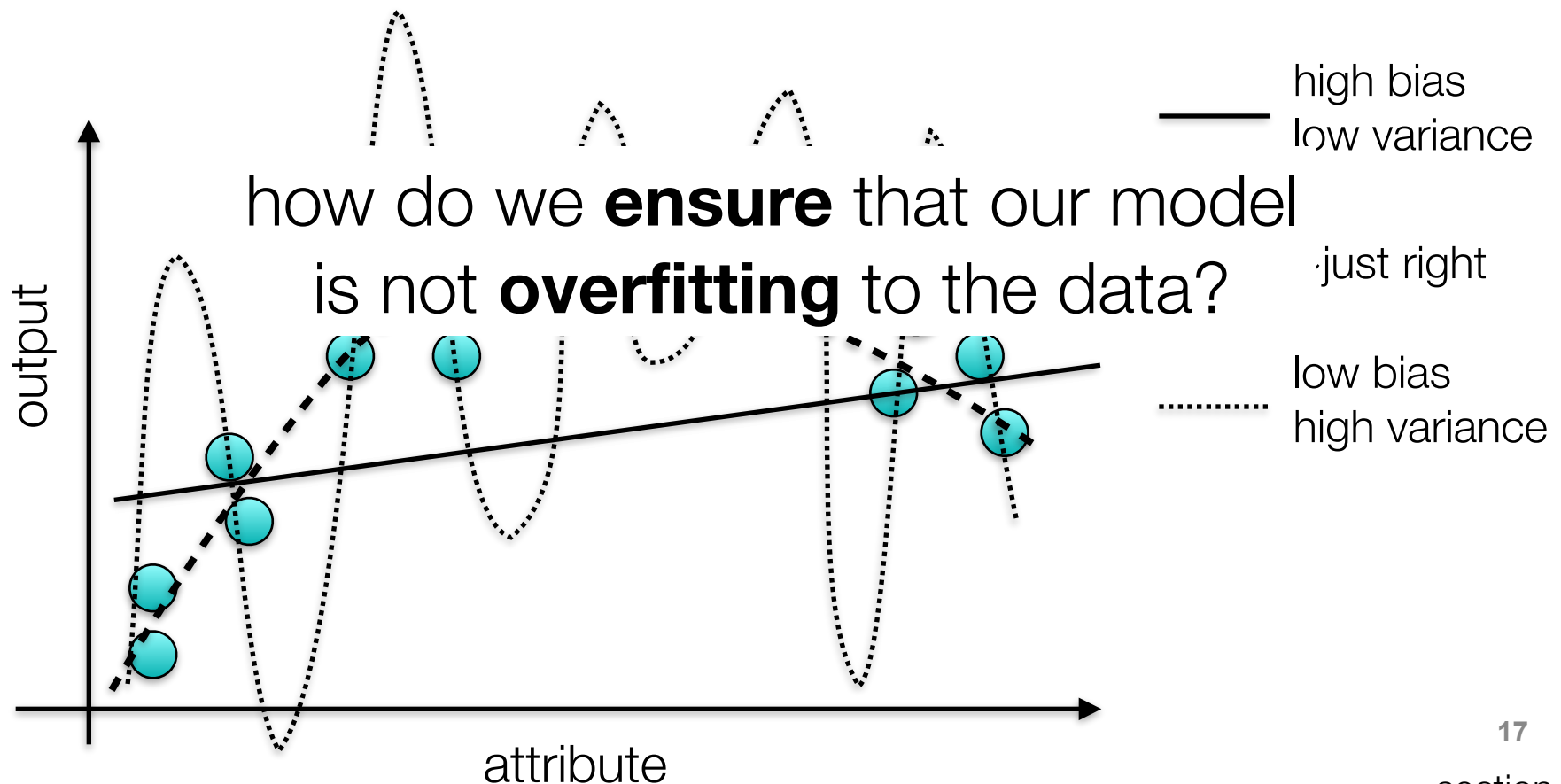
# Bias Variance Tradeoff

- **Complex** models can really fit the training data, giving **lower bias**
- **Simpler** models have trouble fitting data, resulting in **higher bias**
- But complex models can have **high variance** in their decision!



# Bias Variance Tradeoff

- **Complex** models can really fit the training data, giving **lower bias**
- **Simpler** models have trouble fitting data, resulting in **higher bias**
- But complex models can have **high variance** in their decision!



# Methods of Estimating Generalization

---

**solution:** use testing set, and *never, never, never* let the model see it

- Holdout
  - Reserve  $2/3$  for training and  $1/3$  for testing
- Random subsampling
  - Repeated holdout, with replacement
- Cross validation
  - Partition data into  $k$  disjoint subsets
  - $k$ -fold: train on  $k-1$  partitions, test on the remaining one
  - Leave-one-out:  $k=n$
- Stratified Cross Validation
  - Select samples, keeping overall class distribution same for each fold



# Self test

---

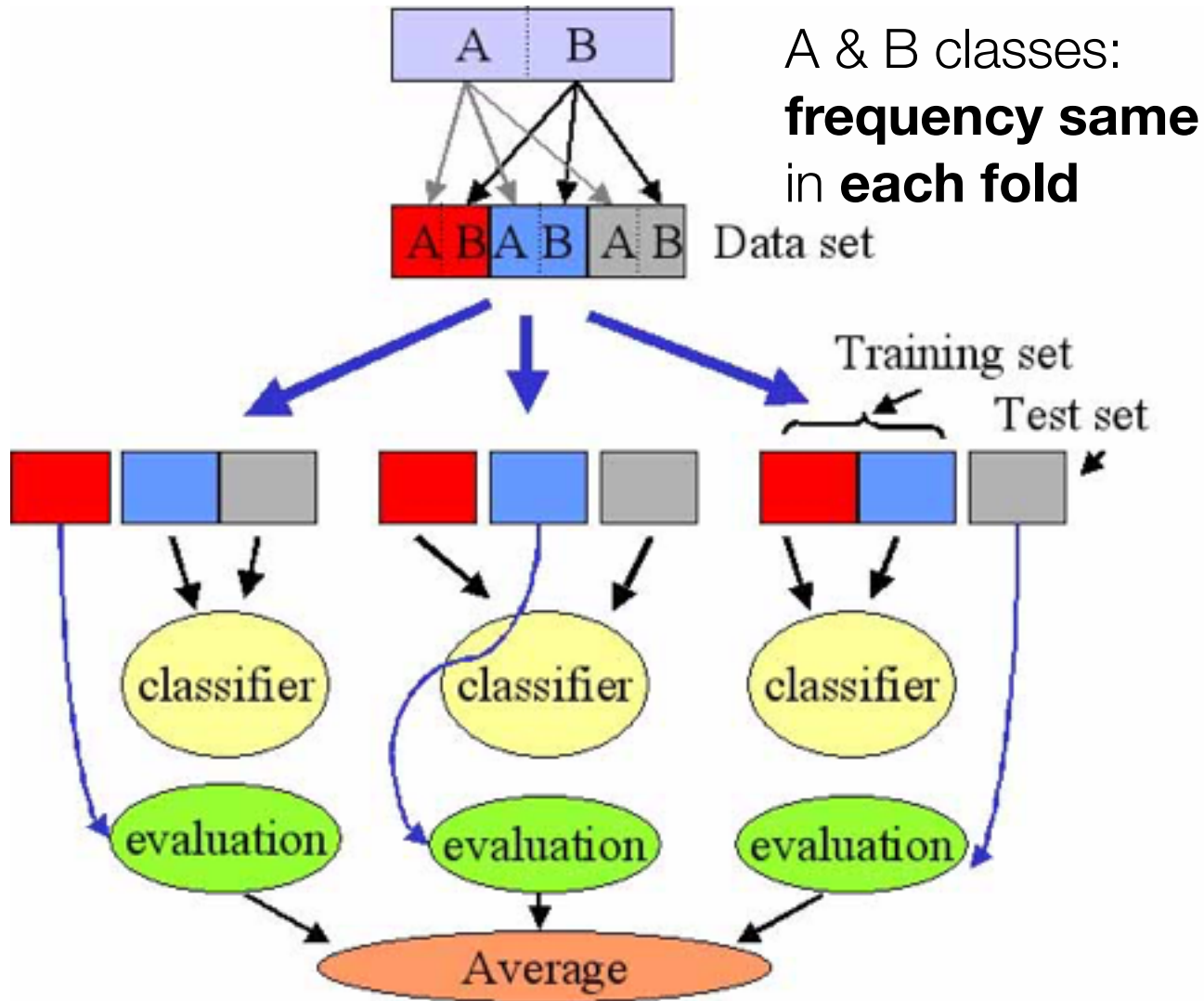
John has a client that is building a predictor model of leak detection for different homes.

Each city where the predictor is installed will have some training examples of leaks they have seen in the past, but John expects that he will need to create entirely different prediction models for each new city where it is installed.

The client gave John a **big** dataset of one “pilot” city. How much data should John use for training and for testing from this atypical dataset?

**Answer:** About as much as John expects to have from each new city installation, then used repeated holdout to see how well it does with different small samples of data.

# Stratified Cross Validation



# Self test

---

From our previous question:

Should John test out his model using 10 fold cross validation?

Answer: No! No! No! No! No! No! No! No! No! No!

That's a lot of training data for a problem where John does not expect to have much training data for each model



# Self test

---

Your boss comes to you and says that he/she created a better regression model than you yesterday in Excel. And has the graph and  $R^2$  value to prove it.

How do you respond?

**You:** Are you testing on the same data you trained with?

**Boss:** No! I'm not dumb

**You:** how many times did you change the algorithm parameters and look at the test set results before making this graph?

**Boss:** ...

**You:** I'm gonna need you to come in on Saturday...

---

# Evaluating Binary Classification: ROC

---

---

# ROC (Receiver Operating Characteristic)

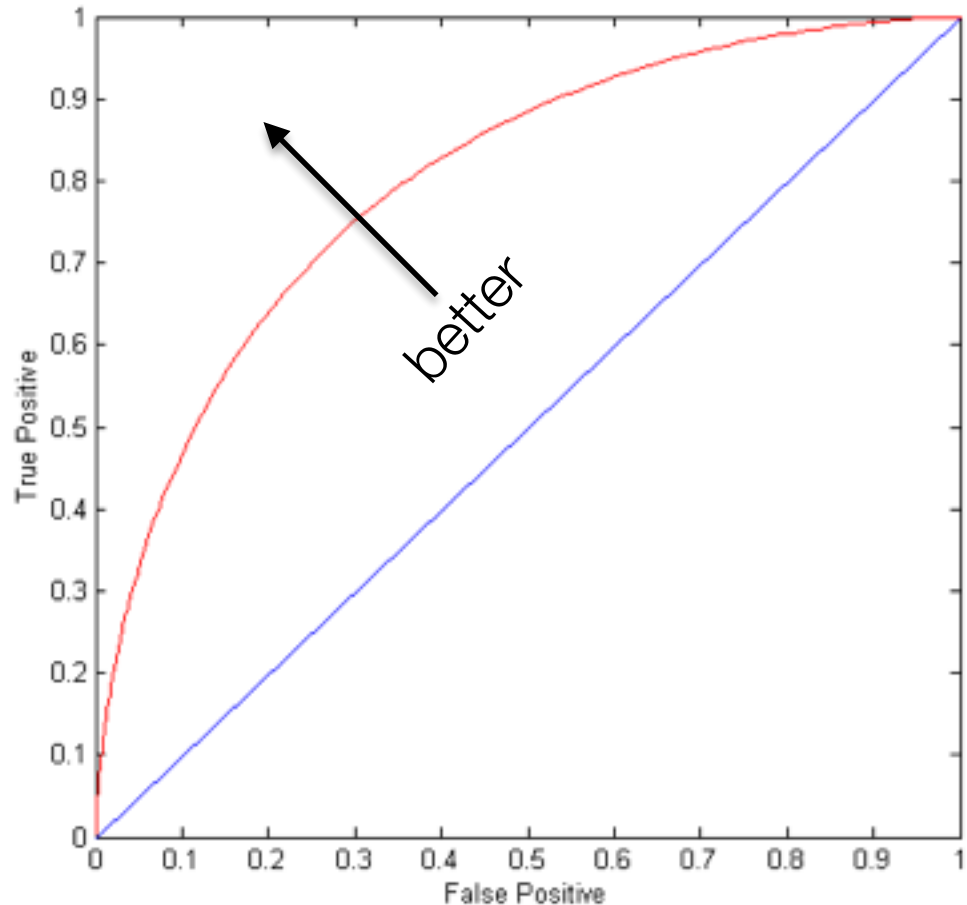
---

- Developed in 1950s for signal detection theory to analyze noisy signals
  - Characterize the trade-off between positive hits and false alarms
- ROC curve plots TP (on the y-axis) against FP (on the x-axis)
- Performance of each classifier represented as a point on the ROC curve
  - changing the threshold of algorithm, sample distribution or cost matrix changes the location of the point

# ROC Curve

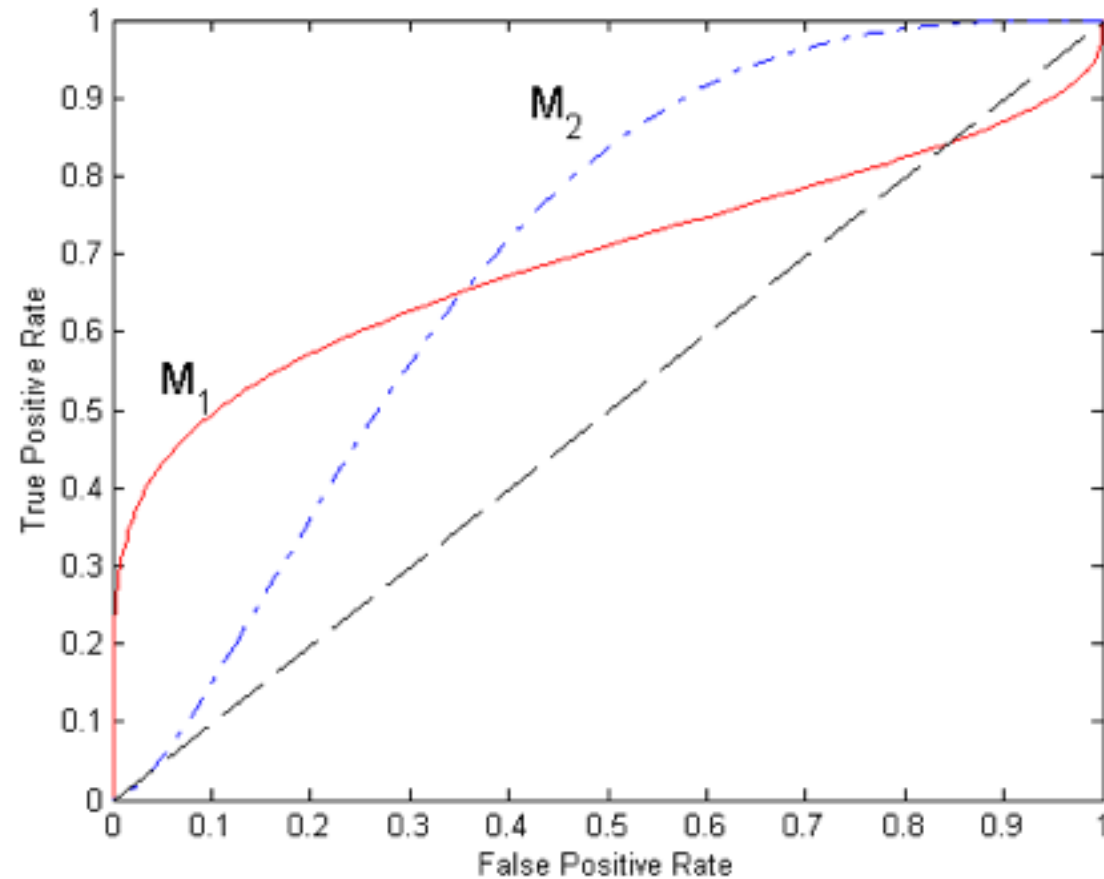
(TP,FP):

- (0,0): declare everything to be negative class
- (1,1): declare everything to be positive class
- (1,0): ideal
- Diagonal line:
  - Random guessing for equal number of classes
  - Below diagonal line:
    - ◆ prediction is opposite of the true class





# Using ROC for Model Comparison



- No model consistently outperforms the other
- $M_1$  is better for small FPR
- $M_2$  is better for large FPR
- Area Under the ROC curve
- Ideal: Area = 1.0

# How to Construct an ROC curve

classifier score

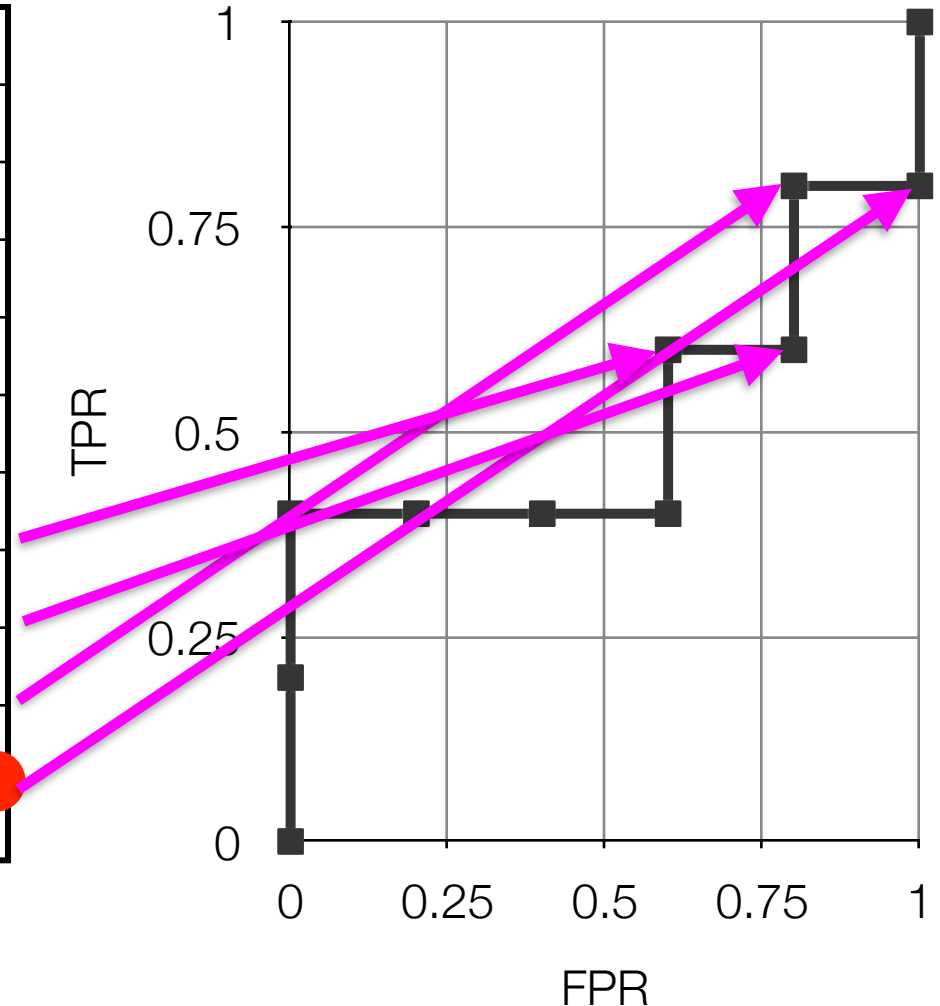
Instance #	$P(+ A)$	True Class
6	0.95	+
2	0.93	+
5	0.87	-
4	0.85	-
9	0.85	-
1	0.85	+
10	0.76	-
8	0.53	+
3	0.43	-
7	0.25	+

- Use classifier that produces probability score for each test instance  $P(+|A)$
- Sort the instances according to  $P(+|A)$  in decreasing order
- Apply threshold,  $T$ , at each unique value of  $P(+|A)$
- $P(+|A) < T$ , is negative class, else it is a positive class
- Count the number of TP, FP, TN, FN at each threshold
- TP rate,  $TPR = TP/Positives$
- FP rate,  $FPR = FP/Negatives$ <sup>28</sup>

# How to Construct an ROC curve

classifier score

Instance #	$P(+ A)$	True Class
6	0.95	+
2	0.93	+
5	0.87	-
4	0.85	-
9	0.85	-
1	0.85	+
10	0.76	-
8	0.53	+
3	0.43	-
7	0.25	+



- TP rate,  $TPR = TP / \text{Positives}$
- FP rate,  $FPR = FP / \text{Negatives}$

# Test of Significance

---

- Given two models:
  - Model  $M_1$ : accuracy = 85%, tested on 30 instances
  - Model  $M_2$ : accuracy = 75%, tested on 5000 instances
- Can we say  $M_1$  is better than  $M_2$ ?
  - How much confidence can we place on accuracy of  $M_1$  and  $M_2$ ?
  - Can the difference in performance measure be explained as a result of random fluctuations in the test set?

# Comparing Performance of 2 Models

- Given two models,  $M_1$  and  $M_2$ , which is better?
  - $M_1$  is tested on  $D_1$  (size= $n_1$ ), found error rate =  $e_1$
  - $M_2$  is tested on  $D_2$  (size= $n_2$ ), found error rate =  $e_2$
  - Assume  $D_1$  and  $D_2$  are independent
  - If  $n_1$  and  $n_2$  are sufficiently large, then

$$e_1 \sim N(\mu_1, \sigma_1)$$


$$e_2 \sim N(\mu_2, \sigma_2)$$

- Approximate: 
$$\hat{\sigma}_i^2 = \frac{e_i(1 - e_i)}{n_i}$$

comes from **binomial distribution**,  
which is approximated well by **normal distribution**

# Comparing Performance of 2 Models

- To test if performance difference is statistically significant:  $d = e_1 - e_2$  ← **estimate of the mean difference**
  - $d \sim N(d_t, \sigma_t)$  where  $d_t$  is the true difference
  - Since  $D_1$  and  $D_2$  are independent, their variance adds up:

$$\begin{aligned}\sigma_t^2 &= \sigma_1^2 + \sigma_2^2 \cong \hat{\sigma}_1^2 + \hat{\sigma}_2^2 \\ &= \frac{e1(1-e1)}{n1} + \frac{e2(1-e2)}{n2}\end{aligned}$$


**estimate of the variance in subtracted error rates**

- At  $(1-\alpha)$  confidence level,  $d_t = d \pm Z_{\alpha/2} \hat{\sigma}_t$

**does this interval include zero?**

# An Illustrative Example

$$d_t = d \pm Z_{\alpha/2} \hat{\sigma}_t$$

- Given:  $M_1: n_1 = 30, e_1 = 0.15$   
 $M_2: n_2 = 5000, e_2 = 0.25$
- $d = |e_2 - e_1| = 0.1$  (2-sided test)

$$\hat{\sigma}_d^2 = \frac{0.15(1-0.15)}{30} + \frac{0.25(1-0.25)}{5000} = 0.0043$$

- At 95% confidence level,  $Z_{\alpha/2} = 1.96$

$$d_t = 0.100 \pm 1.96 \times \sqrt{0.0043} = 0.100 \pm 0.128$$

- $\Rightarrow$  Interval contains 0  $\Rightarrow$  difference may not be statistically significant

# Another illustrative example

---



**“THERE ARE LIES, DAMNED LIES AND  
STATISTICS.”**

**MARK TWAIN**

© Lifehack Quotes

$M_1: n_1 = 30, e_1 = 0.15$

$M_2: n_2 = 5000, e_2 = 0.25$

Use common sense and choose the classifier with 5000 training examples



# Comparing Performance of 2 Algorithms

- Each learning algorithm may produce  $k$  models:
  - $L_1$  may produce  $M_{11}, M_{12}, \dots, M_{1k}$
  - $L_2$  may produce  $M_{21}, M_{22}, \dots, M_{2k}$
- If models are generated on the same test sets  $D_1, D_2, \dots, D_k$  (e.g., via cross-validation)
  - For each set: compute  $d_j = e_{1j} - e_{2j}$ , the  $j^{\text{th}}$  difference
  - $d_j$  has mean  $d$  and variance  $\sigma_t$

$$\sigma_t^2 = \frac{1}{k-1} \sum_j^k (\bar{d} - d_j)^2$$
$$d_t = \bar{d} \pm \frac{1}{\sqrt{k}} t_{1-\alpha, k-1} \sigma_t$$

**now we can bound to  
← get a better idea about  
how the accuracy varies**

$$t(95\%, k=10) = 2.26$$

# An illustrative Example

	$\mu_1$	$\mu_2$	DIFF	
FOLD 1	$e_{11} = .1$	$e_{21} = .5$	$-.4$	$d_1$
FOLD 2	$e_{12} = .1$	$e_{22} = .01$	$.09$	$d_2$
FOLD 3	$e_{13} = .2$	$e_{23} = .01$	$.19$	$d_3$
FOLD 4	$e_{14} = .5$	$e_{24} = .1$	$.05$	$d_4$

$\bar{d} = -0.0175$   
 $\sigma^2 = \frac{1}{k-1} \sum (d_j - \bar{d})^2$   
 $\approx 0.0514$

$$\sigma_t^2 = \frac{1}{k-1} \sum_j^k (\bar{d} - d_j)^2$$

$$d_t = \bar{d} \pm \frac{1}{\sqrt{k}} t_{1-\alpha, k-1} \sigma_t$$

$$CI = \bar{d} \pm \frac{1}{\sqrt{k}} t_{\alpha, k-1} \sigma$$

95%,  $k-1=3 \Rightarrow 3.182$

$$= -0.0175 \pm \frac{1}{\sqrt{4}} 3.182 (.23)$$

$$= -0.0175 \pm 0.36$$

INCLUDES ZERO,  
MODELS MIGHT BE THE SAME

# Comparing Performance of 2 Algorithms

- What about the fact that each model is computed on the same dataset?
- Even if accuracies are similar, what if the errors are on different instances in the data?
- Always a good idea to check the confusions between the classifiers:

Count	Model M2		
		incorrect	correct
Model M1	incorrect	a	b
	correct	c	d

Prob.	C2		
C1		incorrect (0)	correct (1)
	incorrect (0)	0.04	0.16
	correct (1)	0.16	0.64

Prob.	C2		
C1		incorrect (0)	correct (1)
	incorrect (0)	0.18	0.02
	correct (1)	0.02	0.78

# Model Comparison: summary

- If models taken on different datasets
  - Use error rates and number of instances

$$\sigma_t^2 = \sigma_1^2 + \sigma_2^2 \cong \hat{\sigma}_1^2 + \hat{\sigma}_2^2$$

$$= \frac{e1(1-e1)}{n1} + \frac{e2(1-e2)}{n2}$$

$$d_t = d \pm Z_{\alpha/2} \hat{\sigma}_t$$

Does interval include zero?

- If using ten fold cross validation:
  - Use variance and mean of error per fold

$$\sigma_t^2 = \frac{1}{k-1} \sum_j^k (\bar{d} - d_j)^2$$

$$d_t = \bar{d} \pm \frac{1}{\sqrt{k}} t_{1-\alpha, k-1} \sigma_t$$

check that M1 is different from M2,  
Based on comparison matrix!

Does interval include zero?

# Before moving on...

---

- Is there any reason to prefer one learning method over another, if we are only interested in how the algorithm generalizes?
- Should any one algorithm be superior to another in all situations?
- No, there is no free lunch...
- So we need to talk about many classification algorithms!!