CS 461: Machine Learning
Lecture 3

Dr. Kiri Wagstaff
kiri.wagstaff@calstatela.edu

Questions?

- Homework 2
- Project Proposal
- Weka
- Other questions from Lecture 2

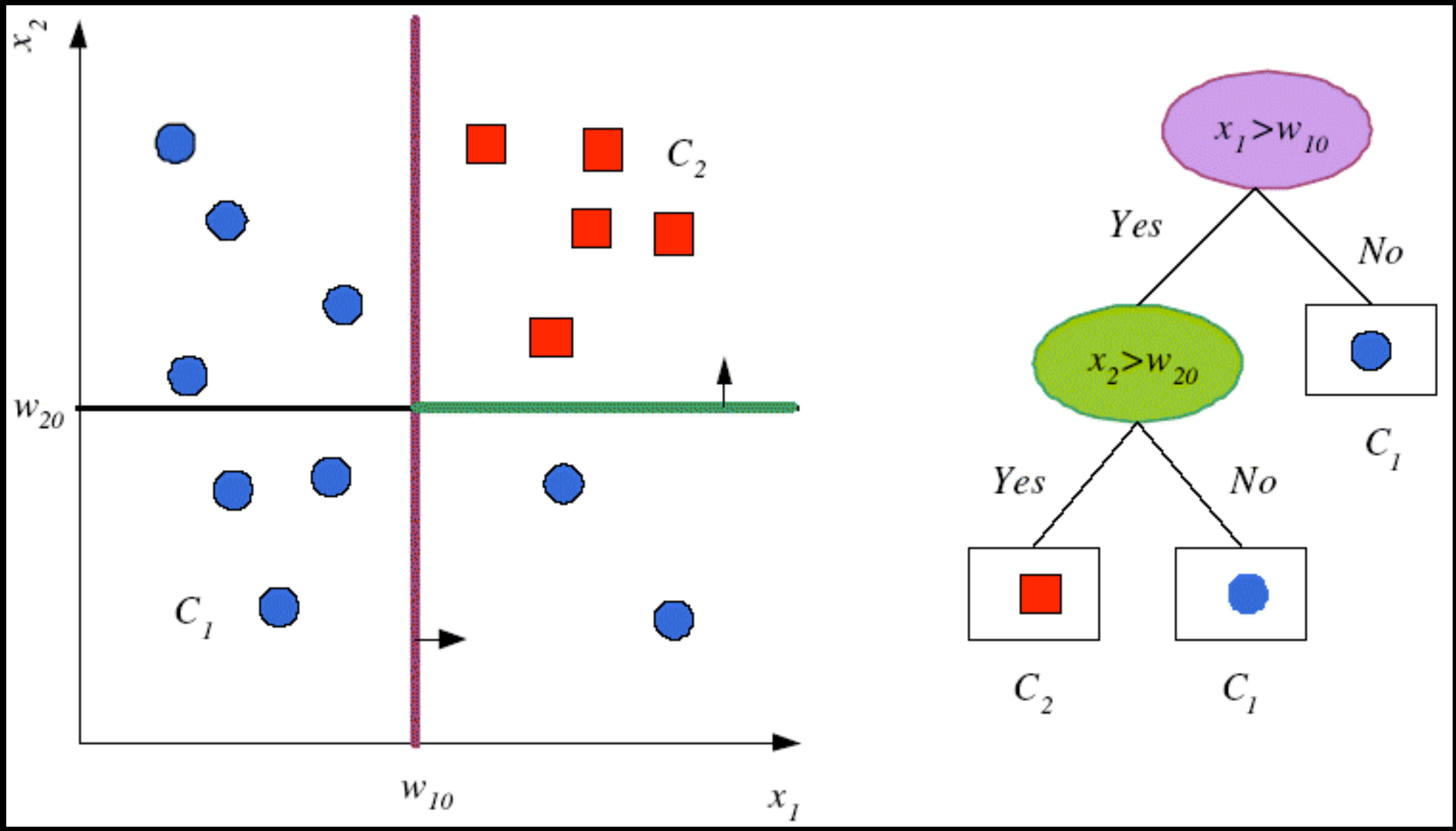
Review from Lecture 2

- Representation, feature types (continuous, discrete, ordinal)
- Model selection, bias, variance, Occam's razor
- Noise: errors in label, features, or unobserved
- Decision trees: nodes, leaves, greedy, hierarchical, recursive, non-parametric
- Impurity: misclassification error, entropy
- Turning trees into rules
- Evaluation: confusion matrix, cross-validation

Plan for Today

- Decision trees
  - Regression trees, pruning
- Evaluation
  - One classifier: errors, confidence intervals, significance
  - Comparing two classifiers
- Support Vector Machines
  - Classification
    - Linear discriminants, maximum margin
    - Learning (optimization)
    - Non-separable classes
  - Regression

Remember Decision Trees?



Algorithm: Build a Decision Tree

GenerateTree($\mathcal{X}$)
    If NodeEntropy($\mathcal{X}$)$< \theta_I$ /* eq. 9.3
        Create leaf labelled by majority class in $\mathcal{X}$
        Return
    $i \leftarrow$ SplitAttribute($\mathcal{X}$)
    For each branch of $\boldsymbol{x}_i$
        Find $\mathcal{X}_i$ falling in branch
        GenerateTree($\mathcal{X}_i$)
SplitAttribute($\mathcal{X}$)
    MinEnt$\leftarrow$ MAX
    For all attributes $i = 1, \ldots, d$
        If $\boldsymbol{x}_i$ is discrete with n values
            Split $\mathcal{X}$ into $\mathcal{X}_1, \ldots, \mathcal{X}_n$ by $\boldsymbol{x}_i$
            e $\leftarrow$ SplitEntropy($\mathcal{X}_1, \ldots, \mathcal{X}_n$) /* eq. 9.8 */
            If e<MinEnt MinEnt $\leftarrow$ e; bestf $\leftarrow$ i
        Else /* $\boldsymbol{x}_i$ is numeric */
            For all possible splits
                Split $\mathcal{X}$ into $\mathcal{X}_1, \mathcal{X}_2$ on $\boldsymbol{x}_i$
                e$\leftarrow$SplitEntropy($\mathcal{X}_1, \mathcal{X}_2$)
                If e<MinEnt MinEnt $\leftarrow$ e; bestf $\leftarrow$ i
    Return bestf

 [Alpaydin 2004 © The MIT Press]

Building a Regression Tree

- Same algorithm… different criterion
- Instead of impurity, use Mean Squared Error (in local region)
  - Predict mean output for node
  - Compute training error
  - (Same as computing the variance for the node)
- Keep splitting until node error is acceptable; then it becomes a leaf
  - Acceptable: error < threshold

Bias and Variance

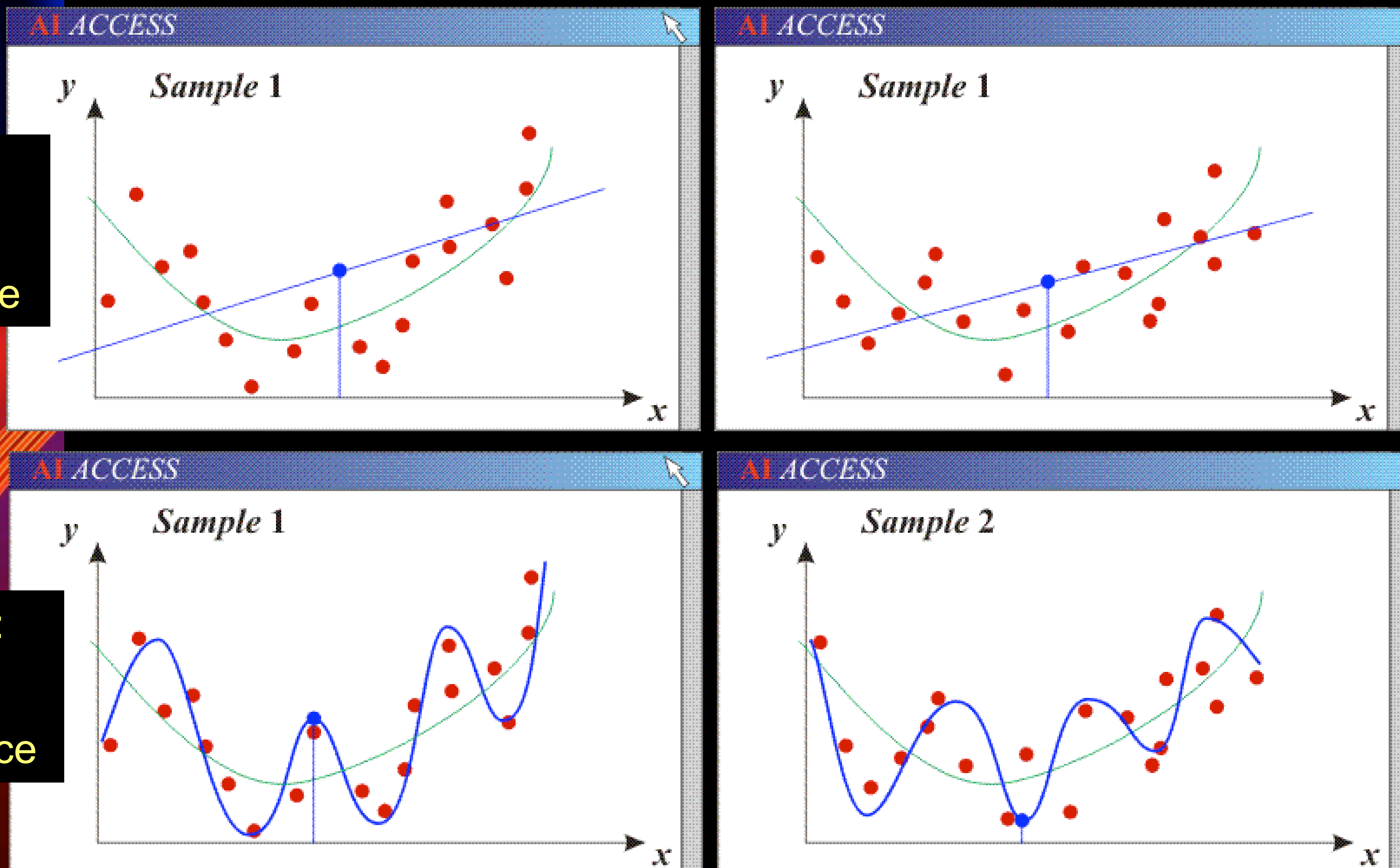
Data Set 1 Data Set 2

Linear:

High bias,
low variance

Polynomial:

Low bias,
high variance



[http://www.aiaccess.net/e_gm.htm]

Evaluating a Single Algorithm

Chapter 14

Measuring Error

	Predicted class	
True Class	Yes	No
Yes	TP: True Positive	FN: False Negative
No	FP: False Positive	TN: True Negative

Iris

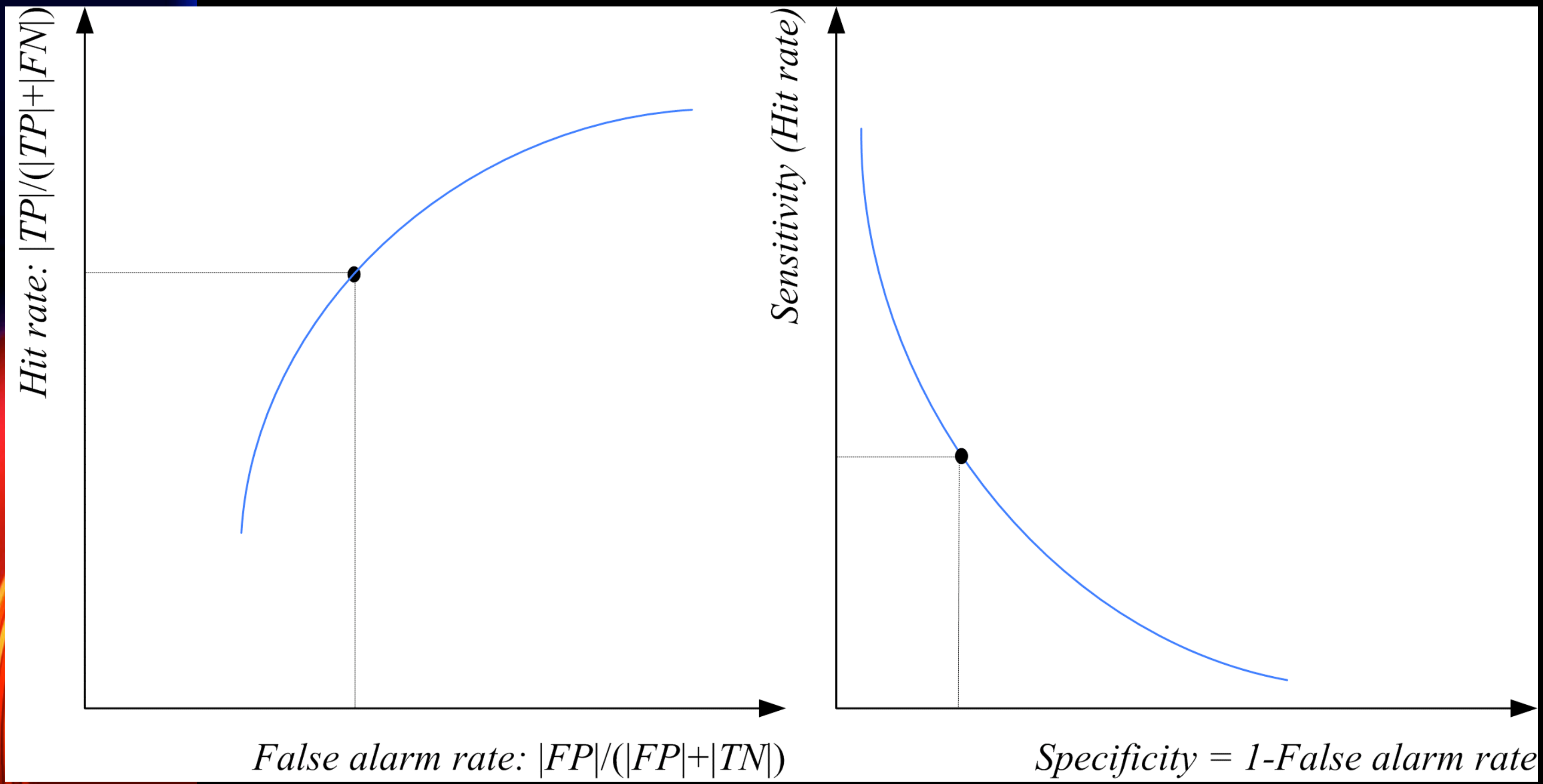
	Setosa	Versicolor	Virginica
Setosa	10	0	0
Versicolor	0	10	0
Virginica	0	1	9

Haberman

	Survived	Died
Survived	9	3
Died	4	4

 [Alpaydin 2004 © The MIT Press]

ROC Curves

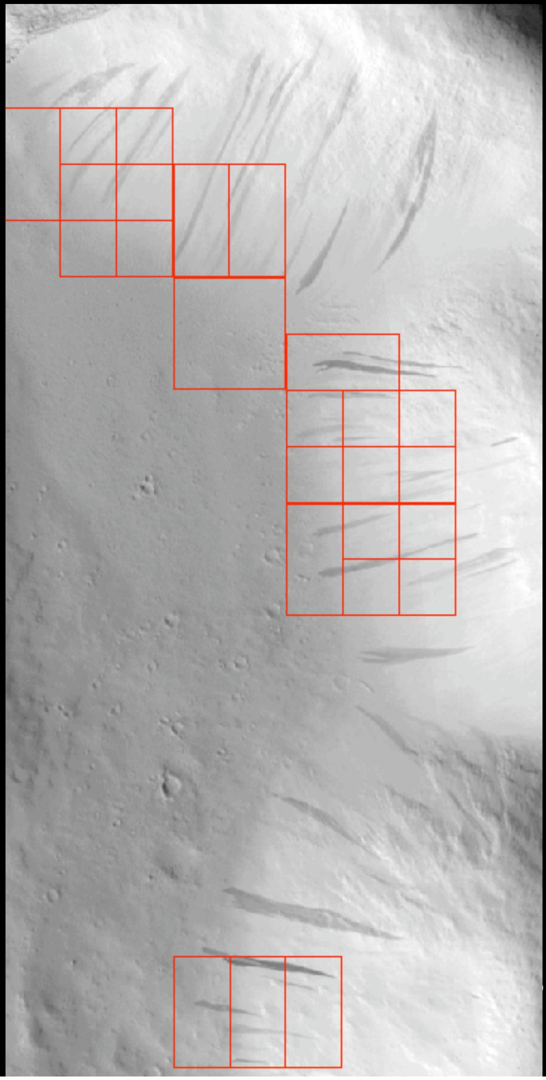


 [Alpaydin 2004 © The MIT Press]

Example: Finding Dark Slope Streaks on Mars

Marte Vallis,
HiRISE on MRO

Output of statistical
landmark detector: top 10%



Results

TP: 13

FP: 1

FN: 16

Recall = 13/29 = 45%

Precision = 13/14 = 93%

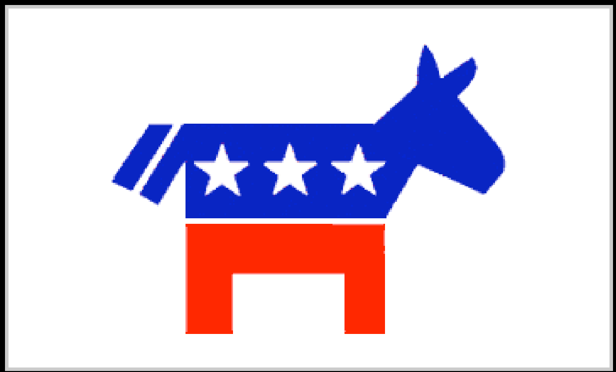
Evaluation Methodology

- Metrics: What will you measure?
  - Accuracy / error rate
  - TP/FP, recall, precision…
- What train and test sets?
  - Cross-validation
  - LOOCV
- What baselines (or competing methods)?
- Are the results significant?

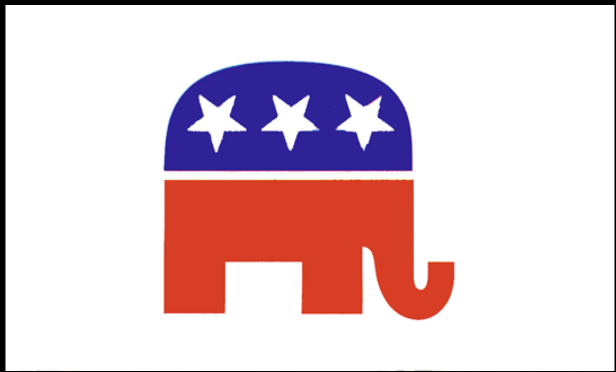
Baselines

- Simple rule
- "Straw man"
- If you can't beat this… don't bother!

- Imagine:



vs.

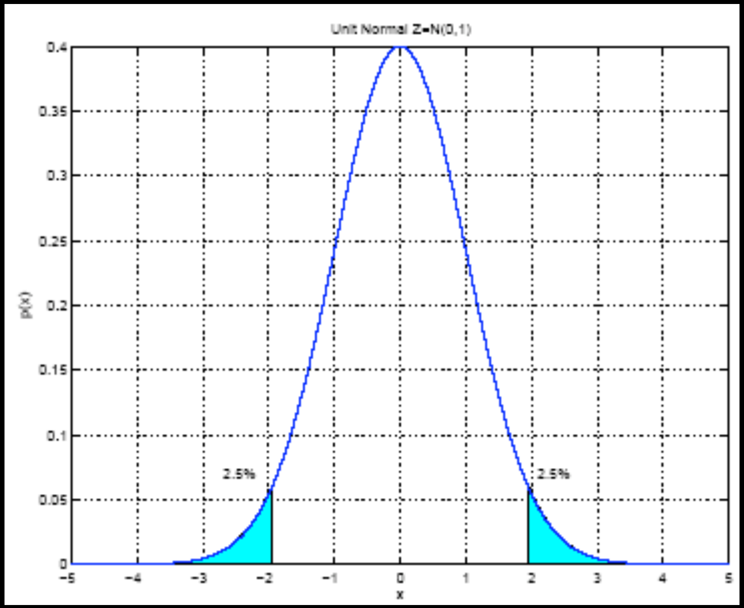


Statistics

- Confidence intervals
- Significant comparisons
- Hypothesis testing

Confidence Intervals

- Normal distribution
  - applet
- t-distribution
  - applet



[Alpaydin 2004 © The MIT Press]

- Confidence interval (CI):
  - Two-sided test:
    - With x% confidence, value is between v1 and v2

CI with Known Variance

- Known variance (use normal dist): CI applet

$$\sqrt{N}\frac{(m-\mu)}{\sigma} \sim Z$$

$$P\left\{-1.96 < \sqrt{N}\frac{(m-\mu)}{\sigma} < 1.96\right\} = 0.95$$

$$P\left\{m - 1.96\frac{\sigma}{\sqrt{N}} < \mu < m + 1.96\frac{\sigma}{\sqrt{N}}\right\} = 0.95$$

$$P\left\{m - z_{\alpha/2}\frac{\sigma}{\sqrt{N}} < \mu < m + z_{\alpha/2}\frac{\sigma}{\sqrt{N}}\right\} = 1-\alpha$$

 [Alpaydin 2004 © The MIT Press]

CI with Unknown Variance

- Unknown variance (use t-dist): CI applet

$$S^2 = \sum_t \left(x^t - m\right)^2 / (N-1) \qquad \frac{\sqrt{N}(m-\mu)}{S} \sim t_{N-1}$$

$$P\left\{m - t_{\alpha/2,N-1}\frac{S}{\sqrt{N}} < \mu < m + t_{\alpha/2,N-1}\frac{S}{\sqrt{N}}\right\} = 1-\alpha$$

 [Alpaydin 2004 © The MIT Press]

Significance (Hypothesis Testing)

- Null hypothesis
  - E.g.: "Average class age is 21 years"
  - "Decision tree has accuracy 93%"
- Accept it with significance α if:
  - Value is in the 100(1- α) confidence interval

$$\frac{\sqrt{N}\left(m-\mu_0\right)}{\sigma} \in \left(-z_{\alpha/2}, z_{\alpha/2}\right)$$

Significance with Cross-Validation: t-test

- K folds = K train/test pairs
  - m = mean error rate
  - S = std dev of error rate
  - p0 = hypothesized error rate
- Accept with significance α if:

$$\frac{\sqrt{K}\left(m - p_0\right)}{S} \sim t_{K-1}$$

  - is less than $t_{\alpha,K-1}$

Comparing Two Algorithms

Chapter 14

Machine Learning Showdown!

- McNemar's Test

e_{00}: Number of examples misclassified by both	e_{01}: Number of examples misclassified by 1 but not 2
e_{10}: Number of examples misclassified by 2 but not 1	e_{11}: Number of examples correctly classified by both

- Under H_0, we expect $e_{01} = e_{10} = (e_{01} + e_{10})/2$

$$\frac{\left(|e_{01} - e_{10}| - 1\right)^2}{e_{01} + e_{10}} \sim \mathcal{X}_1^2$$

Accept if $< \mathcal{X}^2_{\alpha,1}$

[Alpaydin 2004 © The MIT Press]

K-fold CV Paired t-Test

- Use *K*-fold CV to get *K* training/validation folds
- p_i^1, p_i^2: Errors of classifiers 1 and 2 on fold *i*
- $p_i = p_i^1 - p_i^2$: Paired difference on fold *i*
- The null hypothesis is whether p_i has mean 0

$$H_0 : \mu = 0 \text{ vs. } H_0 : \mu \neq 0$$

$$m = \frac{\sum_{i=1}^{K} p_i}{K} \qquad s^2 = \frac{\sum_{i=1}^{K} (p_i - m)^2}{K-1}$$

$$\frac{\sqrt{K}(m-0)}{s} = \frac{\sqrt{K} \cdot m}{s} \sim t_{K-1} \text{ Accept if in } \left(-t_{\alpha/2,K-1}, t_{\alpha/2,K-1}\right)$$

Note: this tests whether they are the same!

[Alpaydin 2004 © The MIT Press]

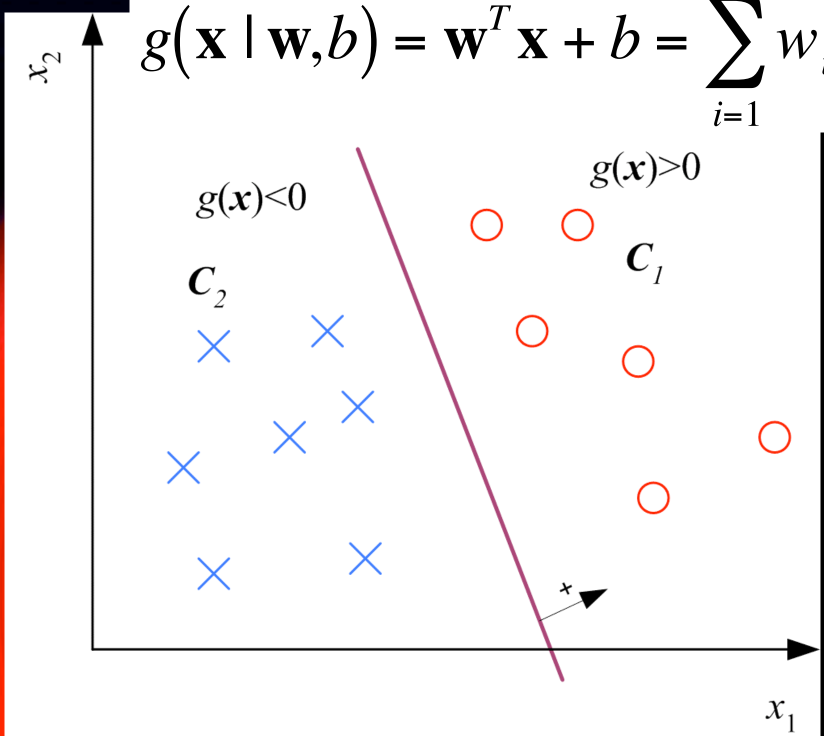
Support Vector Machines

Chapter 10

Linear Discrimination

- Model class boundaries (not data distribution)
- Learning: maximize accuracy on labeled data
- Inductive bias: form of discriminant used

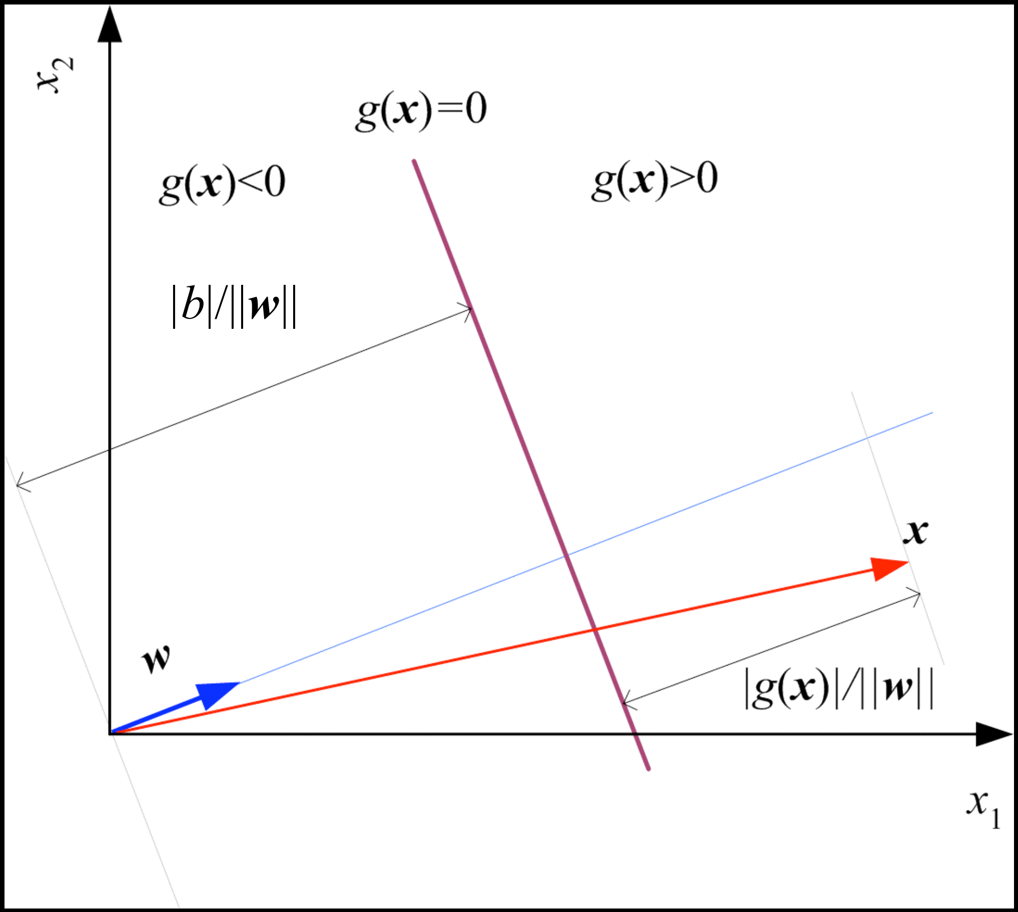
$$g(\mathbf{x} \mid \mathbf{w}, b) = \mathbf{w}^T \mathbf{x} + b = \sum_{i=1}^{d} w_i x_i + b$$



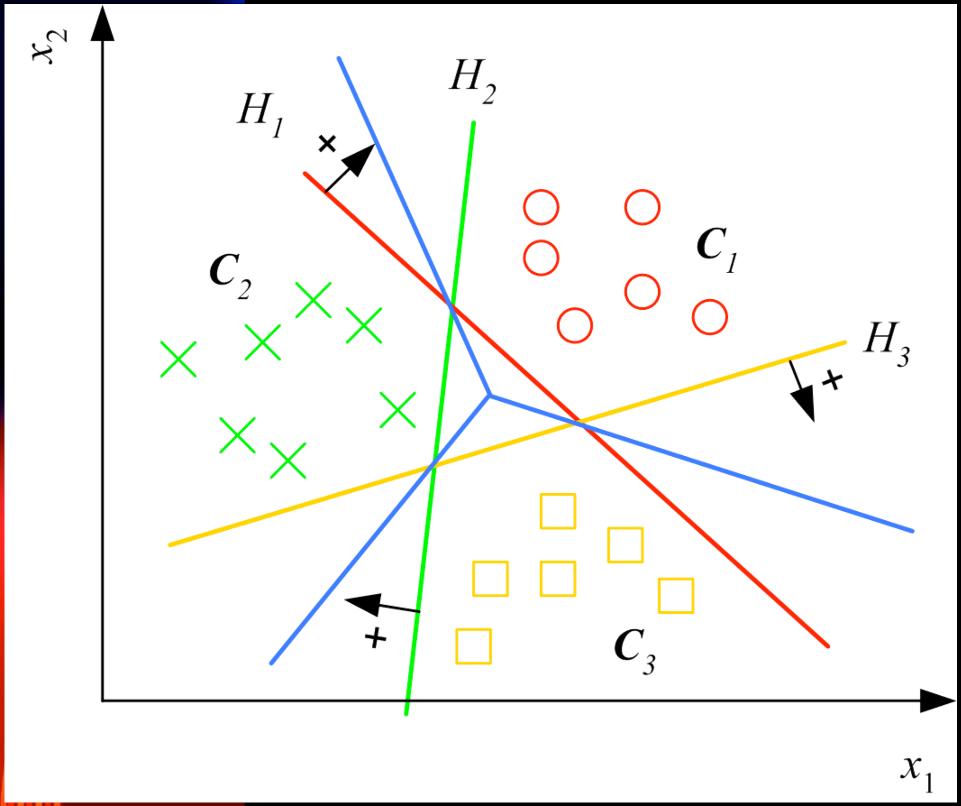
$$\text{choose} \begin{cases} C_1 & \text{if } g(\boldsymbol{x}) > 0 \\ C_2 & \text{otherwise} \end{cases}$$

Linear Discriminant Geometry

$$g(\mathbf{x} \mid \mathbf{w}, b) = \mathbf{w}^T \mathbf{x} + b = \sum_{i=1}^{d} w_i x_i + b$$



Multiple Classes



$$g_i(\mathbf{x} \mid \mathbf{w}_i, b_i) = \mathbf{w}_i^T \mathbf{x} + b_i$$

Choose C_i if

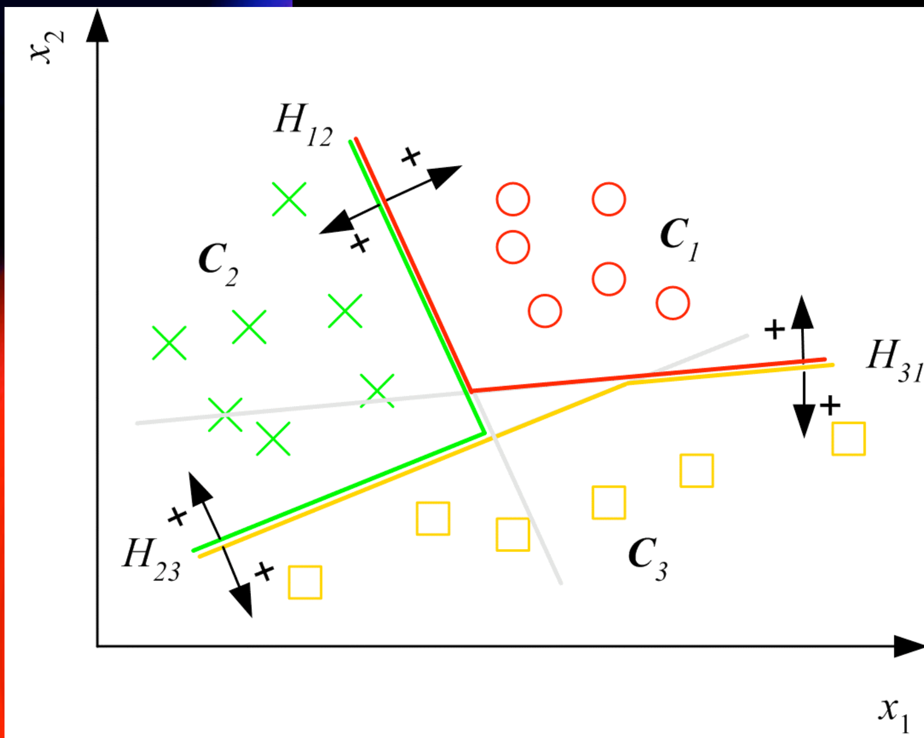
$$g_i(\boldsymbol{x}) = \max_{j=1}^{K} g_j(\boldsymbol{x})$$

Classes are
linearly separable

[Alpaydin 2004 © The MIT Press]

Multiple Classes, not linearly separable

- … but pairwise linearly separable
- Use a one-vs.-one (pairwise) approach



$$g_{ij}\left(\mathbf{x} \mid \mathbf{w}_{ij}, b_{ij}\right) = \mathbf{w}_{ij}^T \mathbf{x} + b_{ij}$$

$$g_{ij}(\boldsymbol{x}) = \begin{cases} > 0 & \text{if } \boldsymbol{x} \in C_i \\ \leq 0 & \text{if } \boldsymbol{x} \in C_j \\ \text{don't care} & \text{otherwise} \end{cases}$$

$$\text{choose } C_i \text{ if} \quad \forall j \neq i, g_{ij}(\boldsymbol{x}) > 0$$

How to find best w, b?

- E($\boldsymbol{w}$|$\mathcal{X}$) is error with parameters $\boldsymbol{w}$ on sample $\mathcal{X}$

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} E(\boldsymbol{w} \mid \mathcal{X})$$

- Gradient

$$\nabla_w E = \left[\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \ldots, \frac{\partial E}{\partial w_d} \right]^T$$

- Gradient-descent:

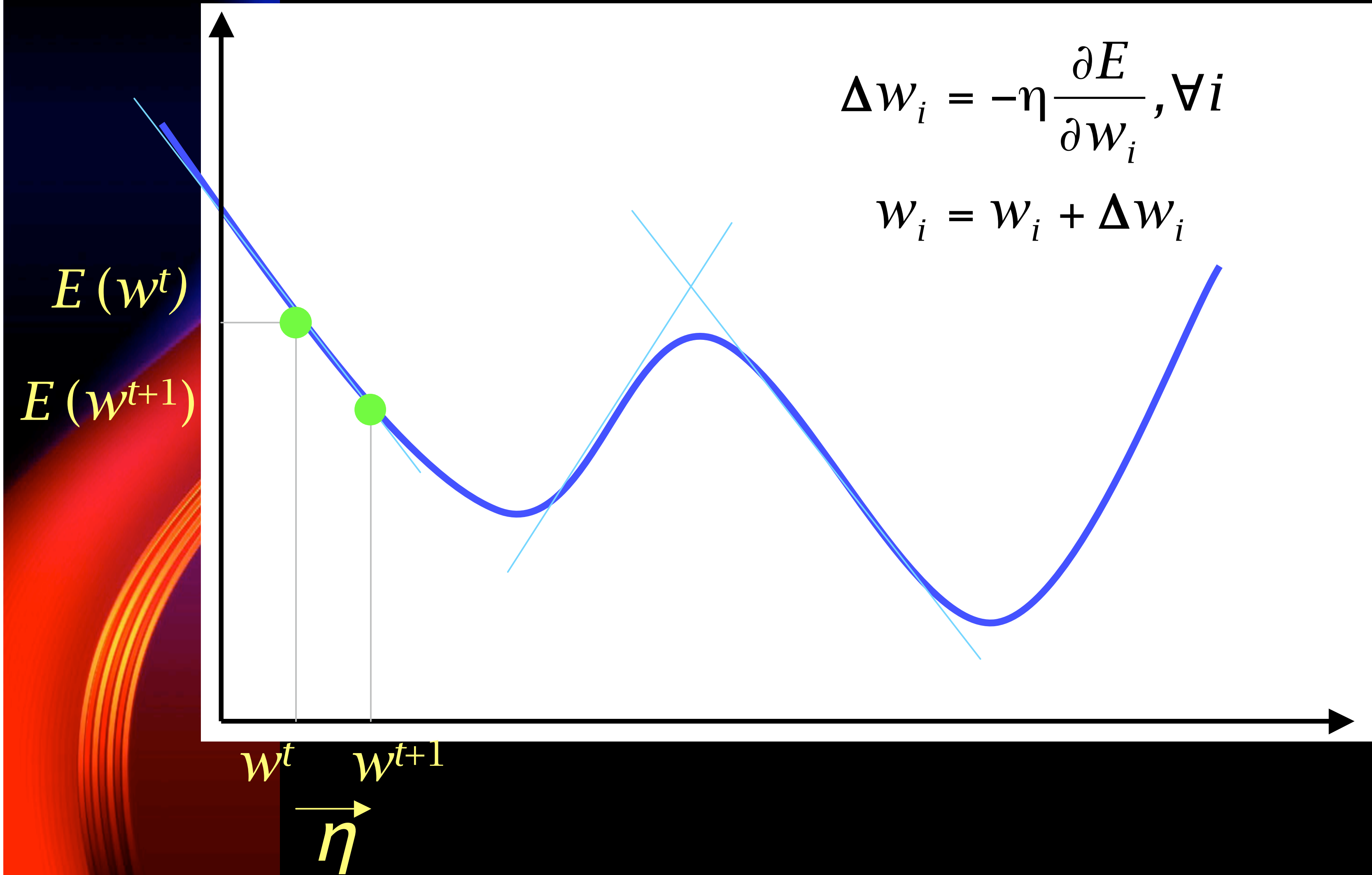
  Starts from random $\boldsymbol{w}$ and updates $\boldsymbol{w}$ iteratively in the negative direction of gradient

[Alpaydin 2004 © The MIT Press]

Gradient Descent

$$\Delta w_i = -\eta \frac{\partial E}{\partial w_i}, \forall i$$

$$w_i = w_i + \Delta w_i$$



Support Vector Machines

- Maximum-margin linear classifiers
  - [Andrew Moore's slides]
- How to find best w, b?
  - Quadratic programming:

$$\min \frac{1}{2}\|\mathbf{w}\|^2 \text{ subject to } y^t\left(\mathbf{w}^T\mathbf{x}^t + b\right) \geq +1, \forall t$$

Optimization (primal formulation)

$$\min \frac{1}{2}\|\mathbf{w}\|^2 \text{ subject to } y^t\left(\mathbf{w}^T\mathbf{x}^t + b\right) \geq +1, \forall t$$

Must get training data right!

$$L_p = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{t=1}^{N} \alpha^t \left[y^t \left(\mathbf{w}^T\mathbf{x}^t + b\right) - 1 \right]$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{t=1}^{N} \alpha^t y^t \left(\mathbf{w}^T\mathbf{x}^t + b\right) + \sum_{t=1}^{N} \alpha^t$$

N + d +1 parameters

Optimization (dual formulation)

We know:

$$\frac{\partial L_p}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{t=1}^{N} \alpha^t y^t \mathbf{x}^t$$

$$\frac{\partial L_p}{\partial b} = 0 \Rightarrow \sum_{t=1}^{N} \alpha^t y^t = 0$$

So re-write:

$$L_p = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{t=1}^{N} \alpha^t y^t \left(\mathbf{w}^T \mathbf{x}^t + b\right) + \sum_{t=1}^{N} \alpha^t$$

$$L_d = \frac{1}{2}\left(\mathbf{w}^T \mathbf{w}\right) - \mathbf{w}^T \sum_t \alpha^t y^t \mathbf{x}^t - b \sum_t \alpha^t y^t + \sum_t \alpha^t$$

$$= -\frac{1}{2}\left(\mathbf{w}^T \mathbf{w}\right) + \sum_t \alpha^t$$

$$= -\frac{1}{2}\sum_t \sum_s \alpha^t \alpha^s y^t y^s \left(\mathbf{x}^t\right)^T \mathbf{x}^s + \sum_t \alpha^t$$

$$\text{subject to } \sum_t \alpha^t y^t = 0 \text{ and } \alpha^t \geq 0, \forall t$$

$\alpha^t > 0$ are the SVs

Optimization in action:
SVM applet

N parameters. Where did **w** and b go?

What if Data isn't Linearly Separable?

1. Add "slack" variables to permit some errors
   - [Andrew Moore's slides]
2. Embed data in higher-dimensional space
   - Explicit: Basis functions (new features)
   - Implicit: Kernel functions (new dot product)
   - Still need to find a linear hyperplane

SVM in Weka

- SMO: Sequential Minimal Optimization
  - Faster than QP-based versions
  - Try linear, RBF kernels

Summary: Key Points for Today

- Decision trees
  - Regression trees, pruning
- Evaluation
  - One classifier: errors, confidence intervals, significance
  - Comparing two classifiers
- Support Vector Machines
  - Classification
    - Linear discriminants, maximum margin
    - Learning (optimization)
    - Non-separable classes

Next Time

- Neural Networks
  (read Ch. 11.1-11.8)

- Questions to answer from the reading
  - Posted on the website (calendar)