CS145 Discussion: Week 3
Decision Tree & SVM

Junheng Hao

Friday, 10/23/2020

Roadmap

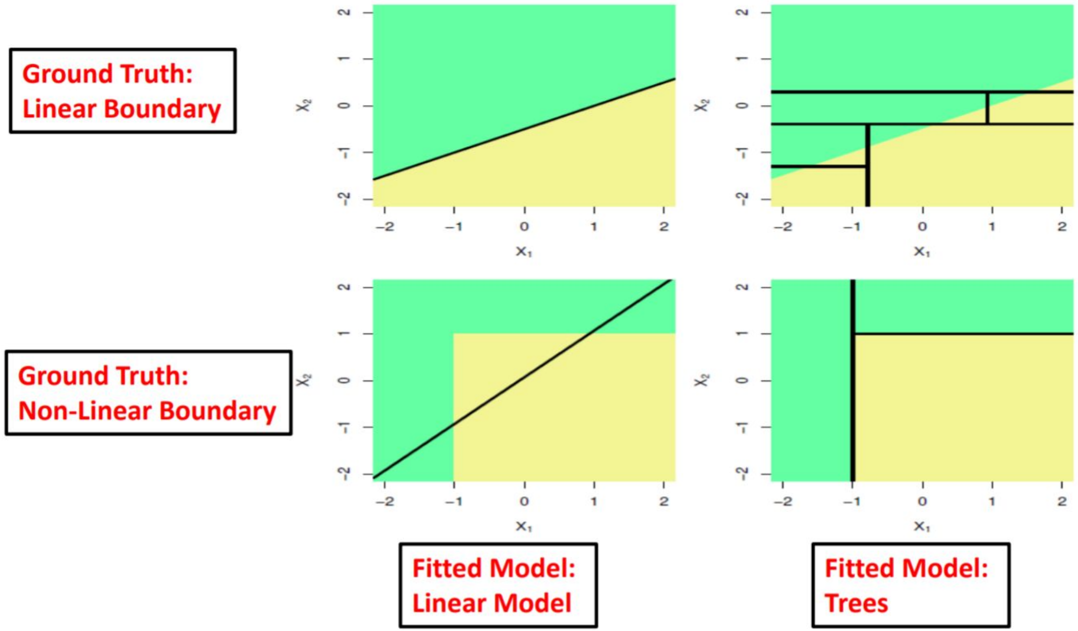
- Announcement
- Decision Tree
- SVM (Part I)

Announcements

- Homework 1 due on **Oct 30 (Friday) 11:59 PT**
  - Submit through GradeScope of 1 PDF (2 python file and 1 jupyter notebook into 1 PDF file)
  - Assign pages to the questions on GradeScope
- Group formation
  - Please email the TA whose session you're enrolled in for help if you cannot find a group with 4-5 members.
  - You may also find 1 or 2 additional team members if your group has someone who has dropped the class (before the end of Week 3)

Decision Boundary

- Comparison: Logistic Regression vs Decision Tree



Decision Boundary: Exercise

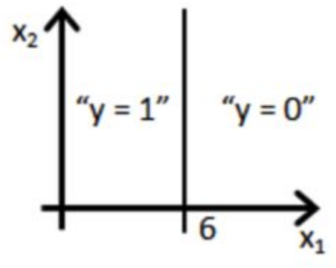
One more question on logistics regression:

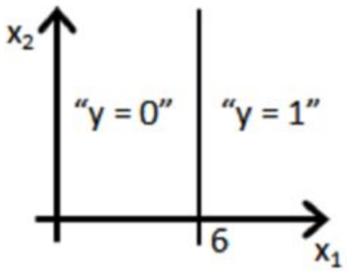
Suppose you train a logistic classifier $h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$. Suppose $\theta_0 = 6, \theta_1 = 0, \theta_2 = -1$. Which of the following figures represents the decision boundary found by your classifier?

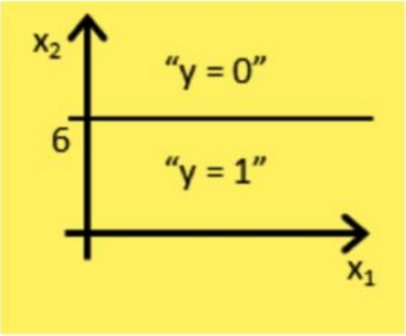
$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}},$$

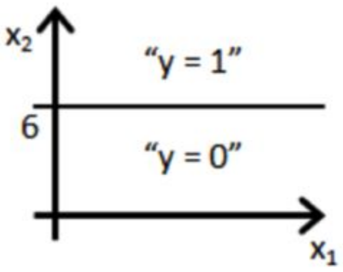
Decision Boundary: Exercise

Suppose you train a logistic classifier $h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$. Suppose $\theta_0 = 6, \theta_1 = 0, \theta_2 = -1$. Which of the following figures represents the decision boundary found by your classifier?





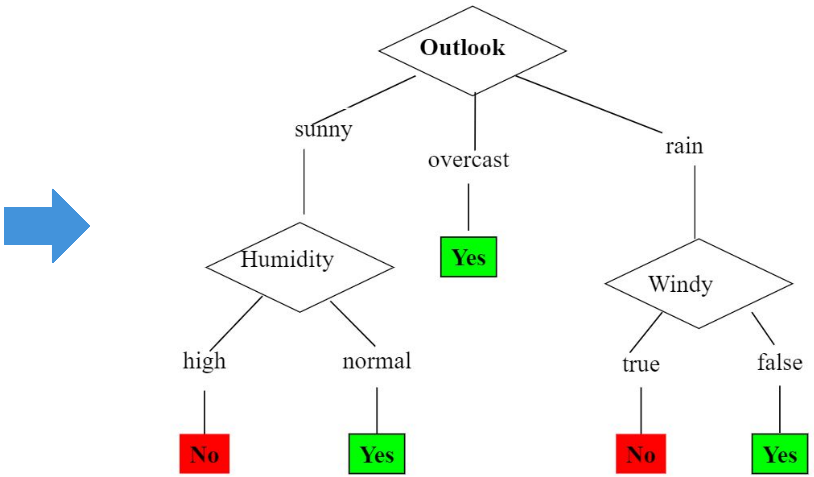




Decision Tree

- Decision Tree Classification: From data to model

Outlook	Temperature	Humidity	Windy	Play?
sunny	hot	high	false	No
sunny	hot	high	true	No
overcast	hot	high	false	Yes
rain	mild	high	false	Yes
rain	cool	normal	false	Yes
rain	cool	normal	true	No
overcast	cool	normal	true	Yes
sunny	mild	high	false	No
sunny	cool	normal	false	Yes
rain	mild	normal	false	Yes
sunny	mild	normal	true	Yes
overcast	mild	high	true	Yes
overcast	hot	normal	false	Yes
rain	mild	high	true	No



Decision Tree: Takeaway

- Choosing the Splitting Attribute
- At each node, available attributes are evaluated on the basis of separating the classes of the training examples.
- A goodness function (information measurement) is used for this purpose:
  - **Information Gain**
  - **Gain Ratio**
  - Gini Index*

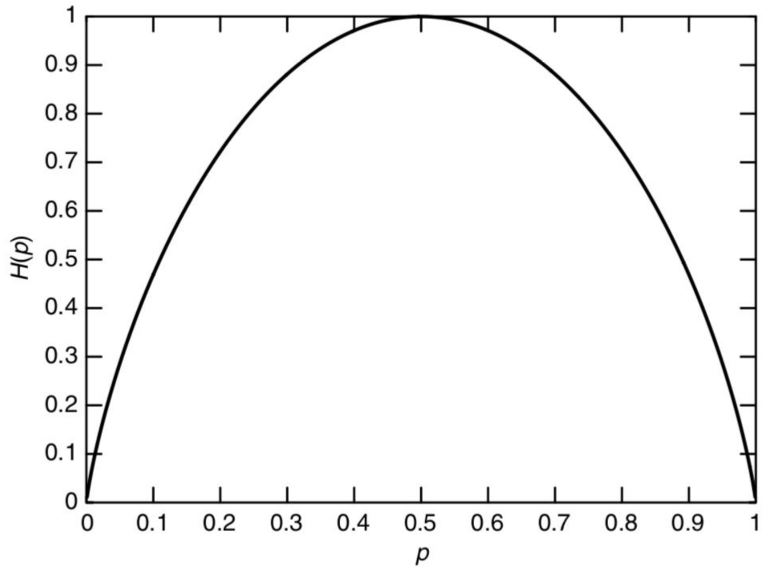
Decision Tree: Attribute Selection

- Which is the best attribute?
  - The one which will result in the smallest tree
  - Heuristic: choose the attribute that produces the “purest” nodes
- Popular *impurity criterion*: *information gain*
  - Information gain increases with the average purity of the subsets that an attribute produces
- Strategy: choose attribute that results in greatest information gain

Decision Tree: Entropy of Random Variable

$$X = \begin{cases} 1 & \text{with probability} \quad p \\ 0 & \text{with probability} \quad 1-p \end{cases}$$

$$H(X) = -p\log p - (1-p)\log(1-p) \stackrel{\text{def}}{=} H(p)$$



Decision Tree: Attribute Selection

- Information in a split with ***x*** items of one class, ***y*** items of the second class

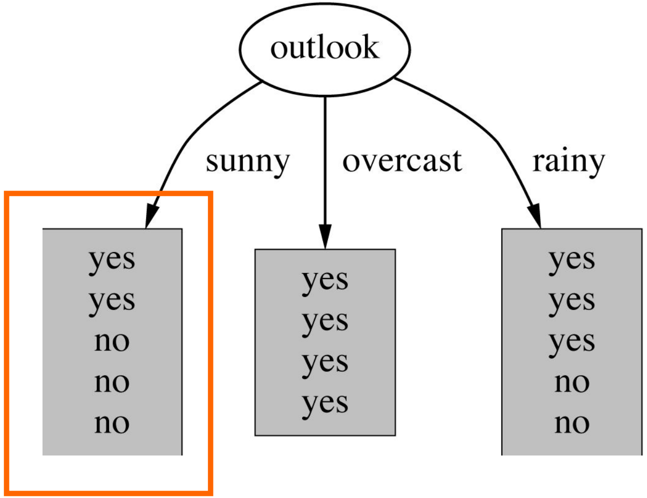
$$\mathrm{info}([\mathrm{x},\mathrm{y}]) = \mathrm{entropy}(\frac{\mathrm{x}}{\mathrm{x}+\mathrm{y}}, \frac{\mathrm{y}}{\mathrm{x}+\mathrm{y}})$$

$$= -\frac{x}{x+y}\log(\frac{x}{x+y}) - \frac{y}{x+y}\log(\frac{y}{x+y})$$

Attribute: "Outlook" = "Sunny"

- "Outlook" = "Sunny": 2 and 3 split

$$\text{info}([2,3]) = \text{entropy}(2/5,3/5) = -\frac{2}{5}\log(\frac{2}{5}) - \frac{3}{5}\log(\frac{3}{5}) = 0.971 \text{ bits}$$

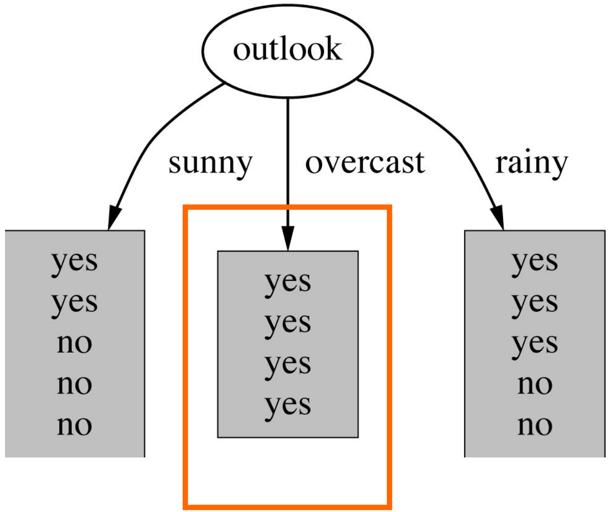


Attribute: "Outlook" = "Overcast"

- "Outlook" = "Overcast": 4/0 split

$$\text{info}([4,0]) = \text{entropy}(1,0) = -1\log(1) \boxed{-0\log(0)} = 0 \text{ bits}$$

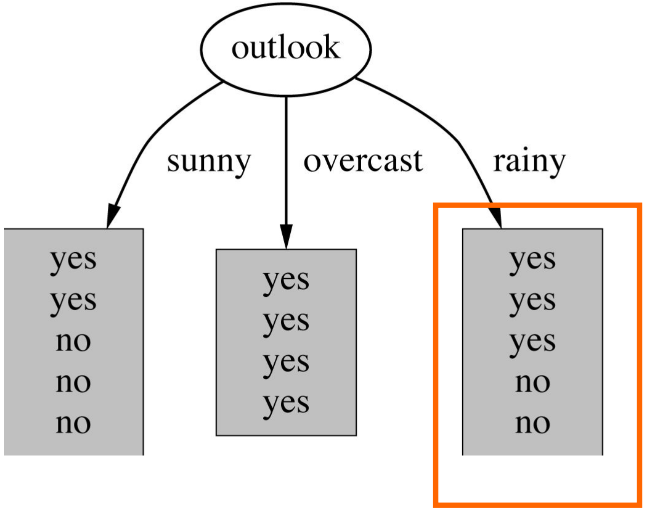
Note: log(0) is not defined, but we evaluate 0*log(0) as zero.



Attribute: "Outlook" = "Rainy"

- "Outlook" = "Rainy":

$$\text{info}([3,2]) = \text{entropy}(3/5, 2/5) = -\frac{3}{5}\log(\frac{3}{5}) - \frac{2}{5}\log(\frac{2}{5}) = 0.971\ \text{bits}$$



Expected Information of Attribute "Outlook"

Expected information for attribute:

$$\text{info}([3,2],[4,0],[3,2]) = (5/14)\times 0.971 + (4/14)\times 0 + (5/14)\times 0.971$$

$$= 0.693 \text{ bits}$$

Compute Information Gain

Information gain:

(information before split) – (information after split)

$$\text{gain}(\text{"Outlook"}) = \text{info}([9,5]) - \text{info}([2,3],[4,0],[3,2]) = 0.940 - 0.693$$
$$= 0.247 \text{ bits}$$

Information gain for attributes from all weather data:

$$\text{gain}(\text{"Outlook"}) = 0.247 \text{ bits}$$

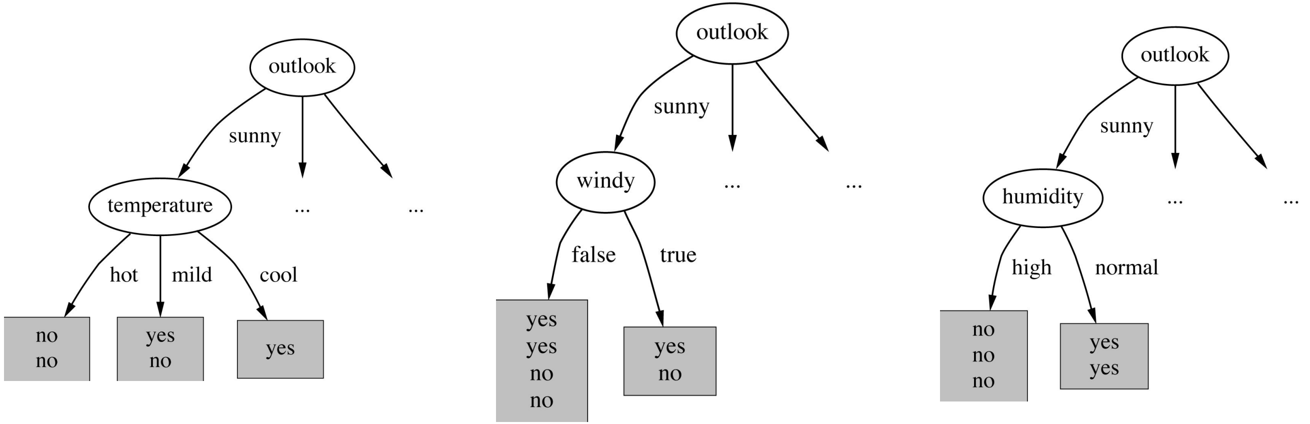
$$\text{gain}(\text{"Temperature"}) = 0.029 \text{ bits}$$

$$\text{gain}(\text{"Humidity"}) = 0.152 \text{ bits}$$

$$\text{gain}(\text{"Windy"}) = 0.048 \text{ bits}$$

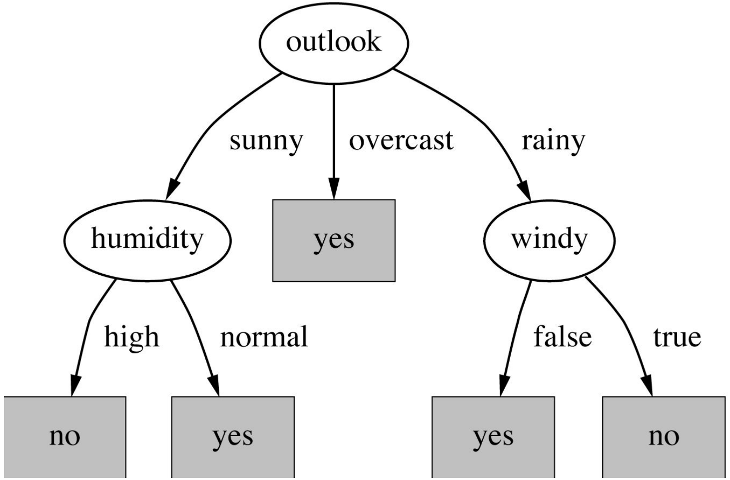
Decision Tree: Example for Practice

Continue to Split



gain("Temperature") = 0.571 bits   gain("Windy") = 0.020 bits   gain("Humidity") = 0.971 bits

Final Tree



- Note: Not all leaves need to be pure. Sometimes identical instances have different classes. → Splitting can stop when data can't be split any further

Final Tree

- SplitInfo and Gain Ratio

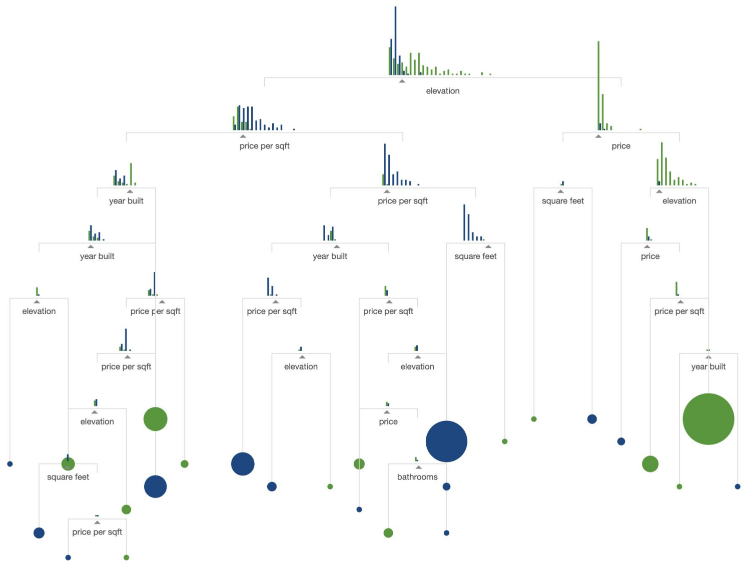
$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

GainRatio(A) = **Gain(A)** / **SplitInfo(A)**

- Why Gain Ratio?
  - Information gain: biased towards attributes with a large number of values
- Practice: What is the gain ratio for attribute “Outlook” in the previous example?

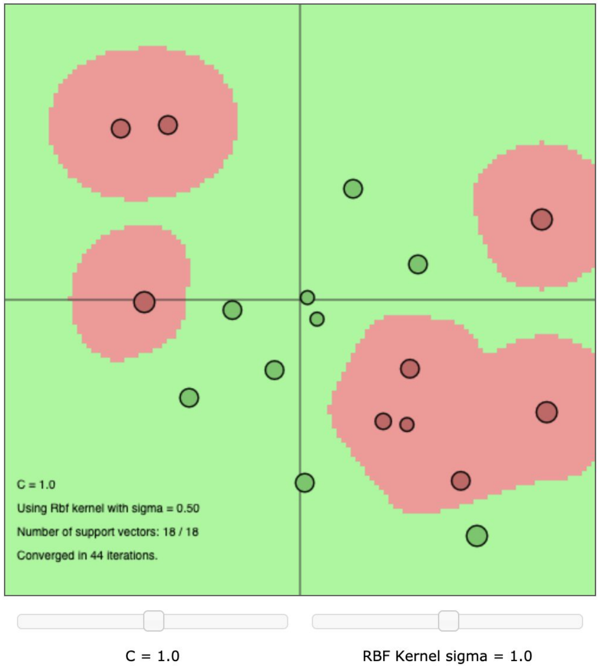
Decision Tree: Visual Tutorials

- Demo links
  - http://www.r2d3.us/visual-intro-to-machine-learning-part-1/
  - http://explained.ai/decision-tree-viz/
- Does decision tree also have the bias-variance trade-off?
  - A visual demo: http://www.r2d3.us/visual-intro-to-machine-learning-part-2/



SVM: Visual Tutorials

- Links: https://cs.stanford.edu/people/karpathy/svmjs/demo/



SVM: Takeaway

- Hyperplane separating the data points

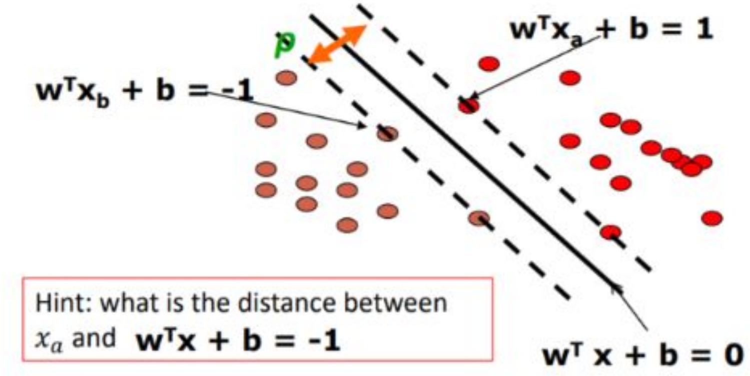
$$\boldsymbol{w}^T\mathbf{x} + \mathrm{b} = 0$$

- Maximize margin

$$\rho = \frac{2}{||w||}$$

- Solution by solving its dual problem

$$\boldsymbol{w} = \sum \alpha_i y_i \mathbf{x}_i \qquad b = \sum\nolimits_{k:\alpha_k \neq 0} (y_k - \boldsymbol{w}^T\mathbf{x}_k)/N_k$$



SVM: Start with the margin

Margin Lines

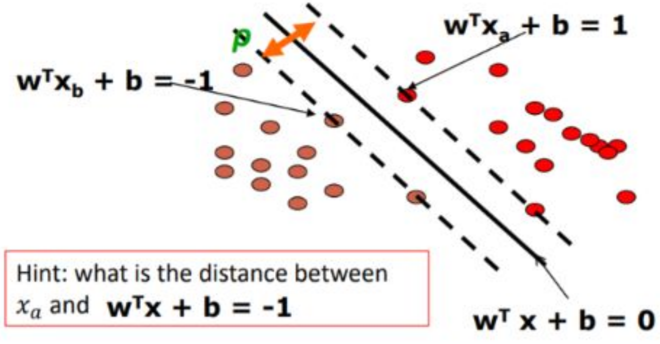
$$\boldsymbol{w}^T\mathbf{x}_a + \mathrm{b} = 1 \qquad \boldsymbol{w}^T\mathbf{x}_b + \mathrm{b} = -1$$

Distance between parallel lines of $ax_1+bx_2=c_1/c_2$)

$$d = \frac{|c_2 - c_1|}{\sqrt{a^2+b^2}}$$

Margin

$$\rho = \frac{|(b+1)-(b-1)|}{||w||} = \frac{2}{||w||}$$



SVM: Steps to understand SVM

1. Formulation of the Linear SVM problem: maximizing margin
2. Formulation of Quadratic Programming (optimization with linear constraints) → Primal problem
3. Solving linear SVM problem with “great” math*
    a. (Generalized) Lagrange function, lagrange multiplier
    b. Identify primal and dual problem (duality) → KKT conditions
    c. Solution to *w* and b regarding alpha
4. Support Vectors, SVM Classifier Inference
5. Non-linear SVM, Kernel tricks

SVM: Math*

- Slides: http://people.csail.mit.edu/dsontag/courses/ml13/slides/lecture6.pdf
- Notes: https://see.stanford.edu/materials/aimlcs229/cs229-notes3.pdf

**To show in hand notes*

Linear SVM: Example for Practice

Given labeled data and alpha values

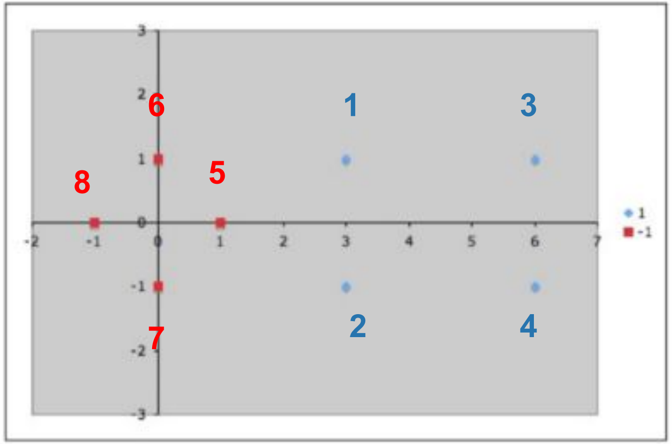
- Positively labeled data points (1 to 4)

$$\left\{\begin{pmatrix}3\\1\end{pmatrix},\begin{pmatrix}3\\-1\end{pmatrix},\begin{pmatrix}6\\1\end{pmatrix},\begin{pmatrix}6\\-1\end{pmatrix}\right\}$$

- Negatively labeled data points (5 to 8)

$$\left\{\begin{pmatrix}1\\0\end{pmatrix},\begin{pmatrix}0\\1\end{pmatrix},\begin{pmatrix}0\\-1\end{pmatrix},\begin{pmatrix}-1\\0\end{pmatrix}\right\}$$

- Alpha values
  - $\alpha_1 = \boxed{0.25}$
  - $\alpha_2 = \boxed{0.25}$
  - $\alpha_5 = \boxed{0.5}$
  - Others = 0

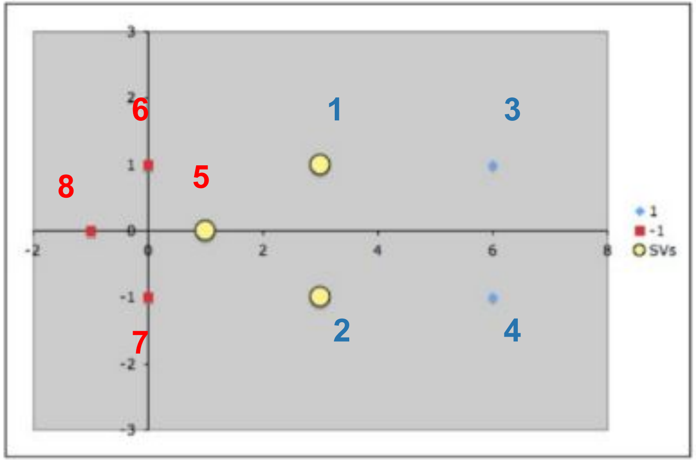


Linear SVM: Example for Practice

Questions

- Which points are support vectors?
- Calculate normal vector of hyperplane: $\boldsymbol{w}$
- Calculate the bias term
- What is the decision boundary?
- Predict class of new point (4, 1)

$$\mathbf{w} = \sum \alpha_i y_i \mathbf{x}_i \qquad b = \sum_{k:\alpha_k \neq 0} (y_k - \mathbf{w}^T \mathbf{x}_k)/N_k$$



Predictions for new data

$$y \leftarrow \text{sign}(\vec{w} \cdot \vec{x} + b)$$

Using dual solution

$$y \leftarrow \text{sign}\left[\sum_i \alpha_i y_i \underbrace{(\vec{x}_i \cdot \vec{x})} + b\right]$$

dot product of feature vectors of new example with support vectors

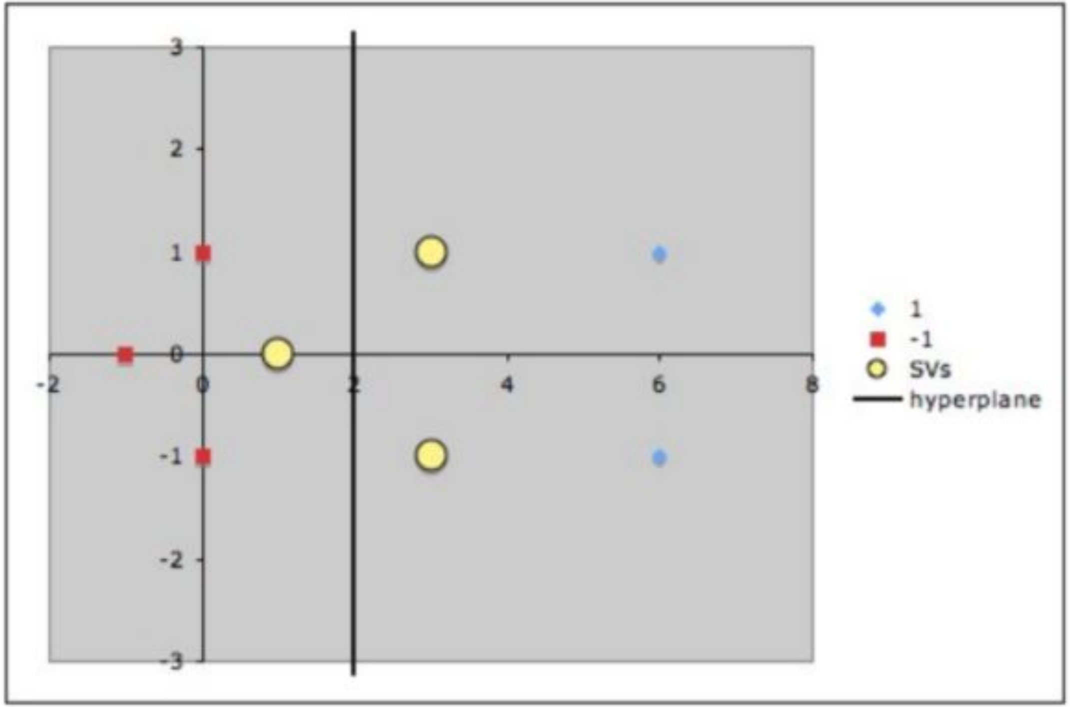
$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$

$$b = y_k - \mathbf{w}.\mathbf{x}_k$$

for any k where $C > \alpha_k > 0$

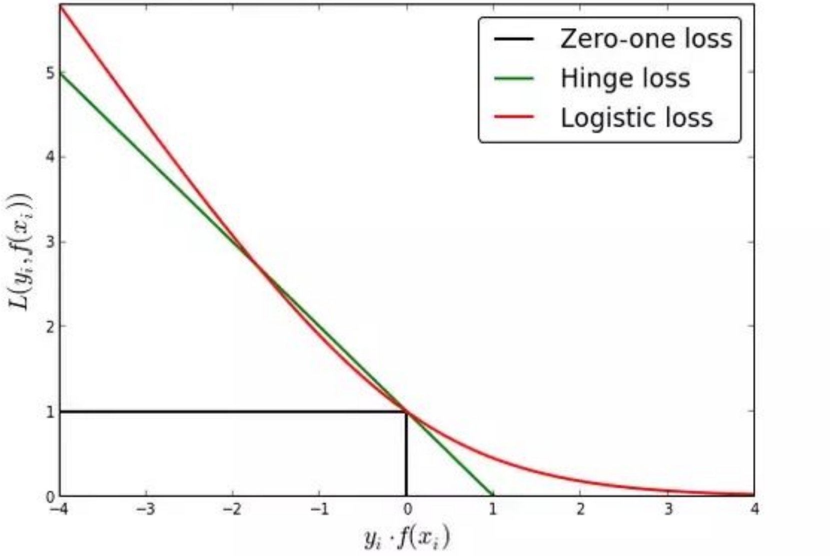
Linear SVM: Example for Practice

Plot



Linear SVM vs Logistic Regression

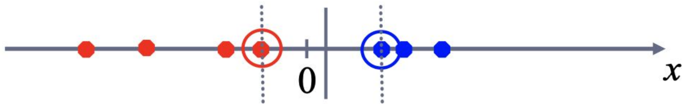
- Decision boundaries?
- Loss functions?



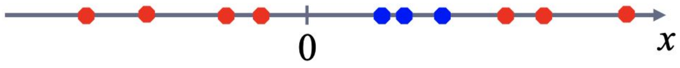
Reading: http://www.cs.toronto.edu/~kswersky/wp-content/uploads/svm_vs_lr.pdf

Non-linear SVM

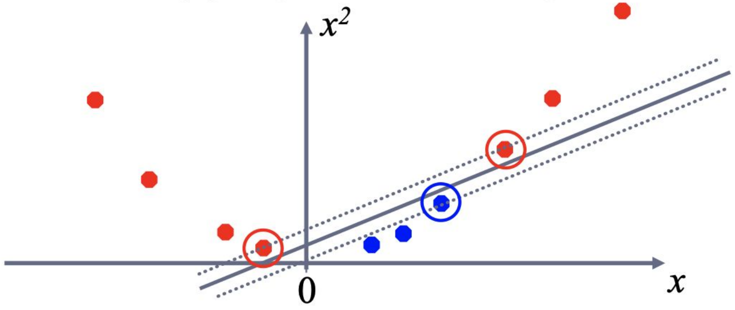
- Datasets that are linearly separable (with some noise) work out great:



- But what are we going to do if the dataset is just too hard?

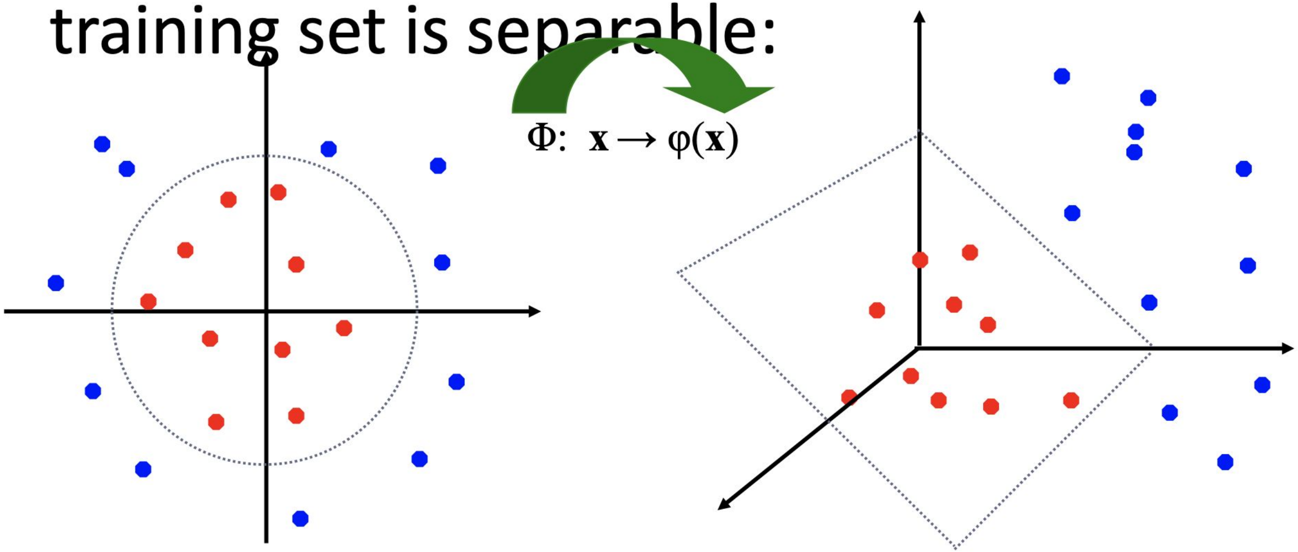


- How about … mapping data to a higher-dimensional space:



Non-linear SVM

- General idea: the original feature space can always be mapped to some higher-dimensional feature space where the training set is separable:



SVM: Kernel

$$\text{maximize}_\alpha \quad \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i\alpha_j y_i y_j \boxed{\mathbf{x}_i\mathbf{x}_j}$$

$$\sum_i \alpha_i y_i = 0$$
$$C \geq \alpha_i \geq 0$$

$$\text{maximize}_\alpha \quad \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i\alpha_j y_i y_j \boxed{K(\mathbf{x}_i, \mathbf{x}_j)}$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$$
$$\sum_i \alpha_i y_i = 0$$
$$C \geq \alpha_i \geq 0$$

SVM: Kernel

- The linear SVM relies on an inner product between data vectors,

$$K(\mathbf{x_i}, \mathbf{x_j}) = \mathbf{x_i^T x_j}$$

- If every data point is mapped into high-dimensional space via transformation, the inner product becomes,

$$K(\mathbf{x_i}, \mathbf{x_j}) = \phi^T(\mathbf{x_i}) \cdot \phi(\mathbf{x_j})$$

- Do we need to compute ***ϕ(x)*** explicitly for each data sample? → **Directly compute kernel function *K(xi, xj)***

SVM: Kernel Function Example

$$
\begin{aligned}
k(\mathbf{x}, \mathbf{z}) &= (\mathbf{x}^T\mathbf{z} + c)^2 = \left(\sum_{j=1}^{n} x^{(j)} z^{(j)} + c\right)\left(\sum_{\ell=1}^{n} x^{(\ell)} z^{(\ell)} + c\right) \\
&= \sum_{j=1}^{n}\sum_{\ell=1}^{n} x^{(j)} x^{(\ell)} z^{(j)} z^{(\ell)} + 2c\sum_{j=1}^{n} x^{(j)} z^{(j)} + c^2 \\
&= \sum_{j,\ell=1}^{n} (x^{(j)} x^{(\ell)})(z^{(j)} z^{(\ell)}) + \sum_{j=1}^{n} (\sqrt{2c}x^{(j)})(\sqrt{2c}z^{(j)}) + c^2,
\end{aligned}
$$

Feature mapping given by:

$$\mathbf{\Phi}(\mathbf{x}) = [x^{(1)2}, x^{(1)}x^{(2)}, ..., x^{(3)2}, \sqrt{2c}x^{(1)}, \sqrt{2c}x^{(2)}, \sqrt{2c}x^{(3)}, c]$$

SVM: Kernel Function Example

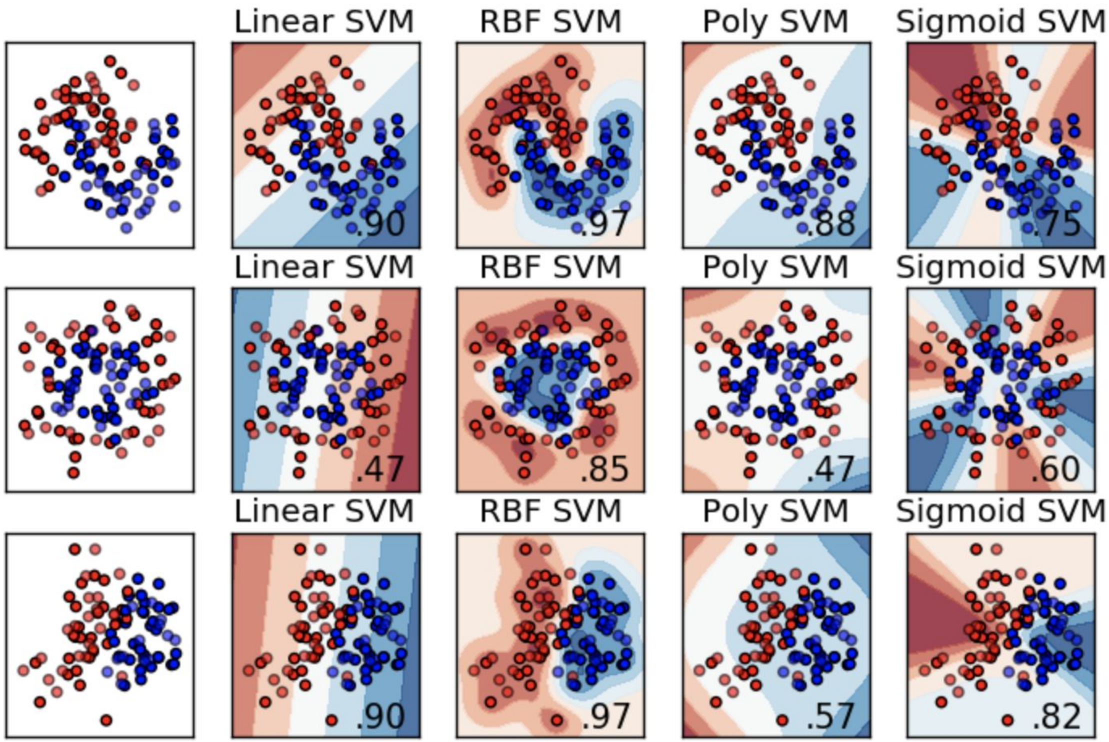
$$\textbf{Polynomial kernel of degree } h: \quad K(X_i, X_j) = (X_i \cdot X_j + 1)^h$$

$$\textbf{Gaussian radial basis function kernel}: \quad K(X_i, X_j) = e^{-\|X_i - X_j\|^2 / 2\sigma^2}$$

$$\textbf{Sigmoid kernel}: \quad K(X_i, X_j) = \tanh(\kappa X_i \cdot X_j - \delta)$$

- Given the same data samples, what is the difference between linear kernel and non-linear kernel? Is the decision boundary linear (in original feature space)?

SVM: Demo of different kernels



Non-linear SVM: Example Practice

- Positively labeled data points (1 to 4)

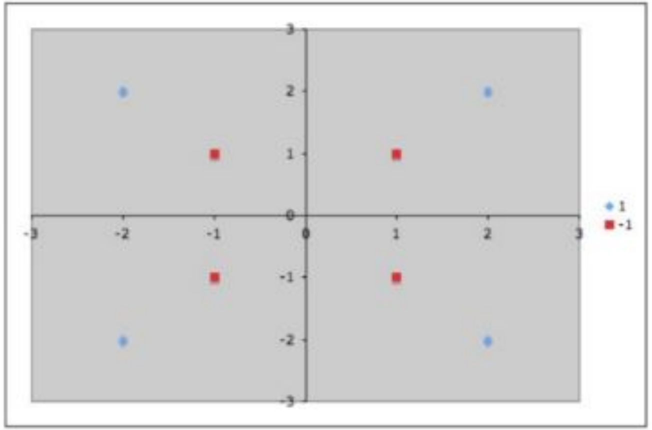
$$\left\{\begin{pmatrix}2\\2\end{pmatrix},\begin{pmatrix}2\\-2\end{pmatrix},\begin{pmatrix}-2\\-2\end{pmatrix},\begin{pmatrix}-2\\2\end{pmatrix}\right\}$$

- Negatively labeled data points (5 to 8)

$$\left\{\begin{pmatrix}1\\1\end{pmatrix},\begin{pmatrix}1\\-1\end{pmatrix},\begin{pmatrix}-1\\-1\end{pmatrix},\begin{pmatrix}-1\\1\end{pmatrix}\right\}$$

- Non-linear mapping

$$\Phi_1\begin{pmatrix}x_1\\x_2\end{pmatrix}=\begin{cases}\begin{pmatrix}4-x_2\\4-x_1\end{pmatrix} & \text{if } \sqrt{x_1^2+x_2^2}>2\\ \begin{pmatrix}x_1\\x_2\end{pmatrix} & \text{otherwise}\end{cases}$$



Non-linear SVM Example

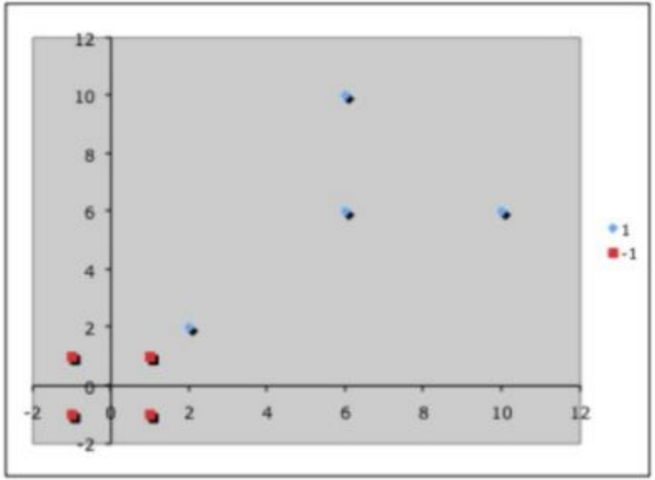
- New positively labeled data points (1 to 4)

$$\left\{\begin{pmatrix} 2 \\ 2 \end{pmatrix}, \begin{pmatrix} 6 \\ 2 \end{pmatrix}, \begin{pmatrix} 6 \\ 6 \end{pmatrix}, \begin{pmatrix} 2 \\ 6 \end{pmatrix}\right\}$$

- New negatively labeled data points (5 to 8)

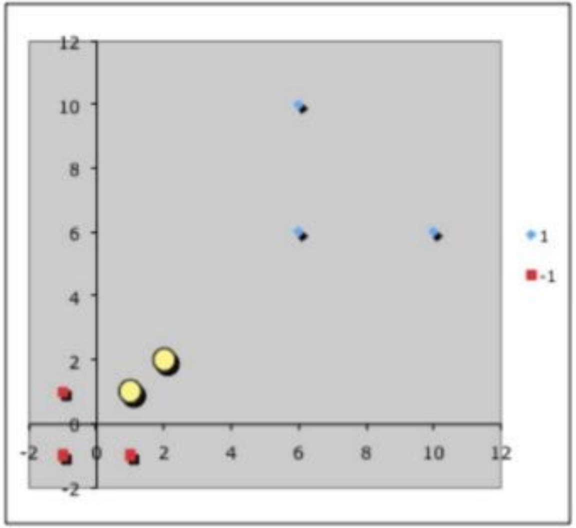
$$\left\{\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}\right\}$$

- Alpha values
  - $\alpha_1 = \boxed{1.0}$
  - $\alpha_5 = \boxed{1.0}$
  - Others = 0



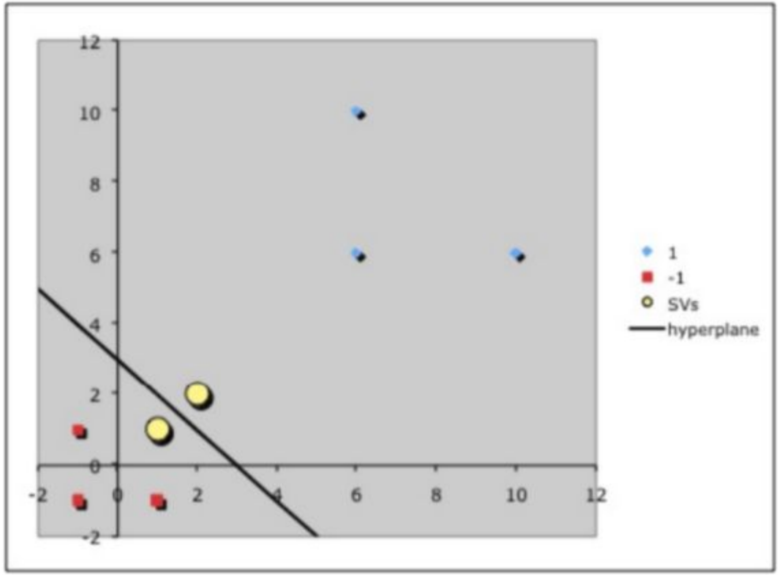
Non-linear SVM Example

- Which points are support vectors?
- Calculate normal vector of hyperplane: $\boldsymbol{w}$
- Calculate the bias term
- What is the decision boundary?
- Predict class of new point (4, 5)

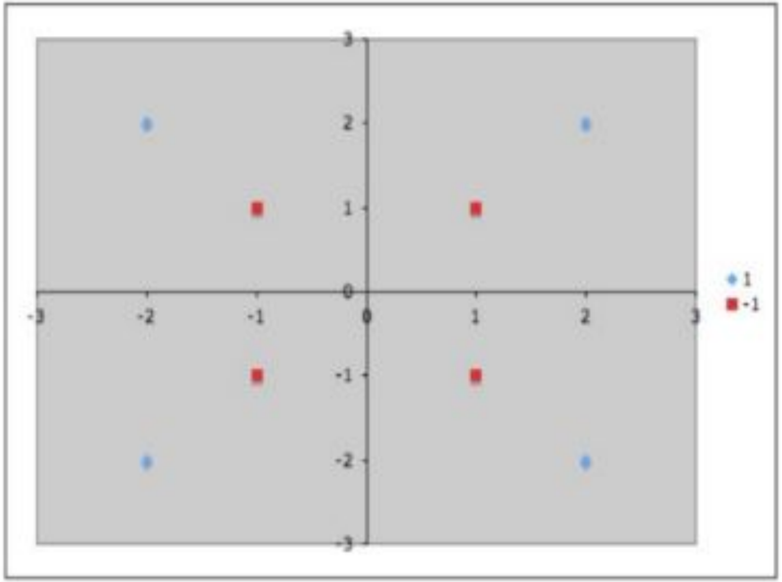


UCLA

Plot



12
10
8
6
4
2
0
-2
-2
0
2
4
6
8
10
12
1
-1
SVs
hyperplane



3
2
1
0
-1
-2
-3
-3
-2
-1
0
1
2
3
1
-1

UCLA Samueli Computer Science

Thank you!

Q & A