Data Science

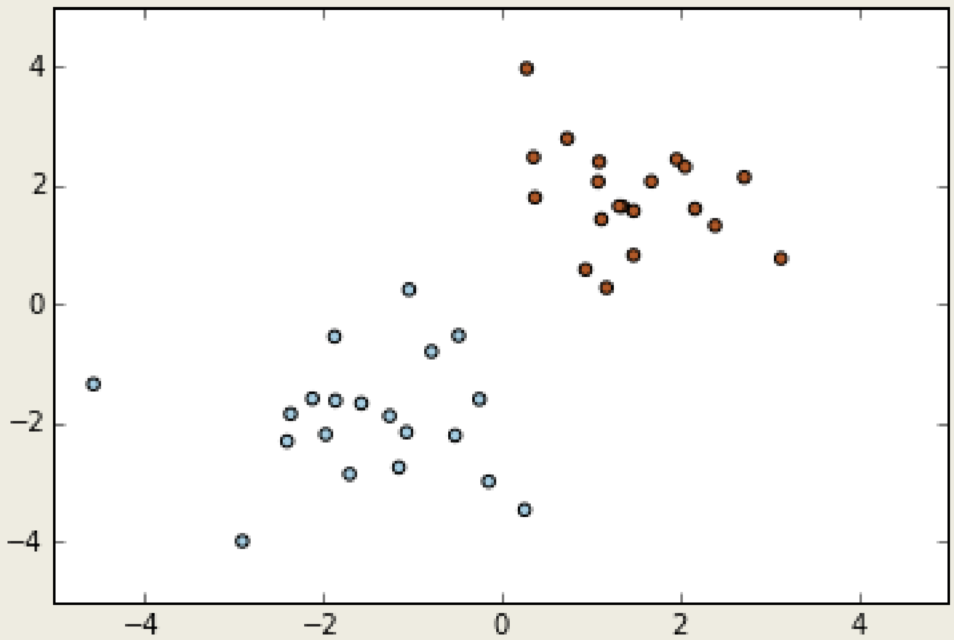
Support Vector Machines

Agenda

- Support Vector Machine
- Non-linear classification, Kernels
- Noisy examples
- SVM Practice in Python

Support Vector Machines

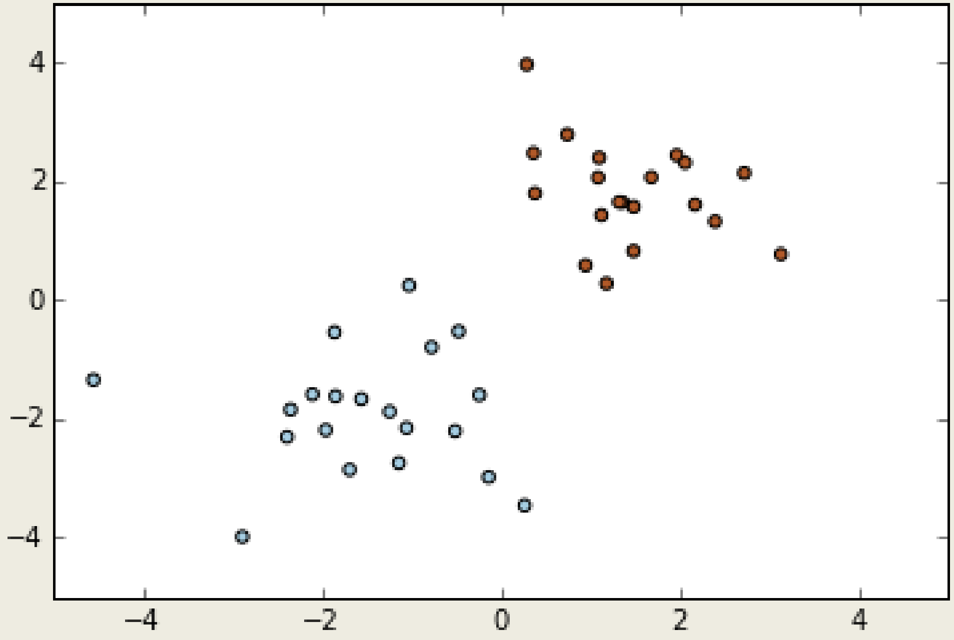
Q: How would you build a classifier for this problem?



Support Vector Machines

Q: How would you build a classifier for this problem?

A: A line can separate the classes (linearly separable)

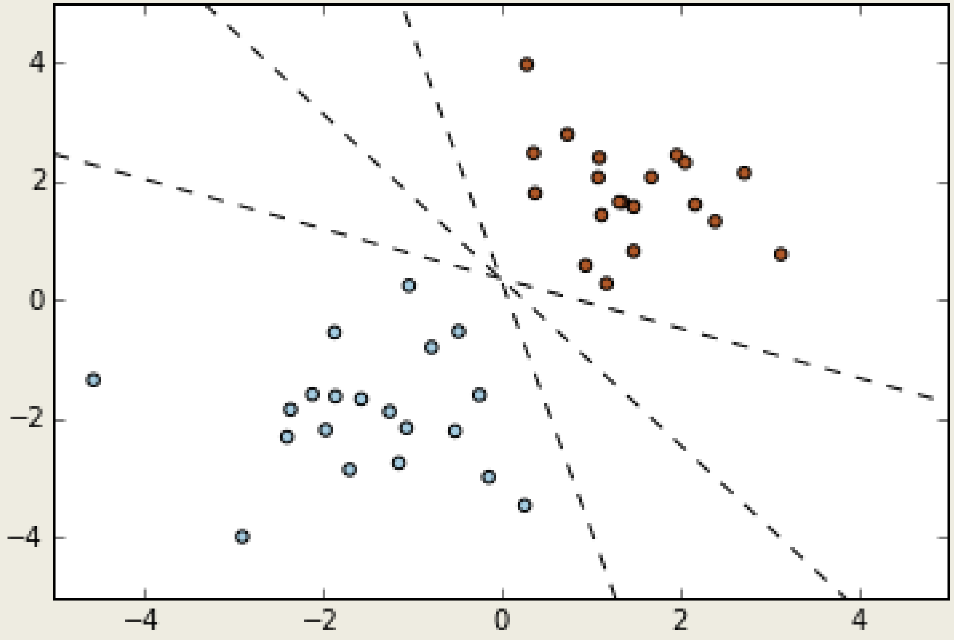


Support Vector Machines

Q: How would you build a classifier for this problem?

A: A line can separate the classes (linearly separable)

Q: Which line is the best?

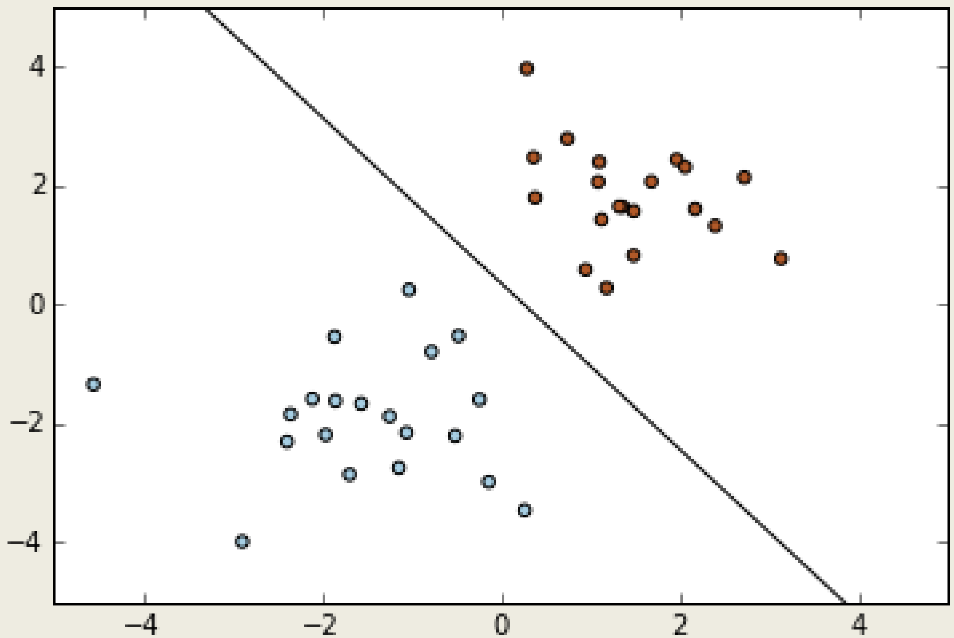


Support Vector Machines

Q: How would you build a classifier for this problem?

A: A line can separate the classes (linearly separable)

Q: Which line is the best? Why?



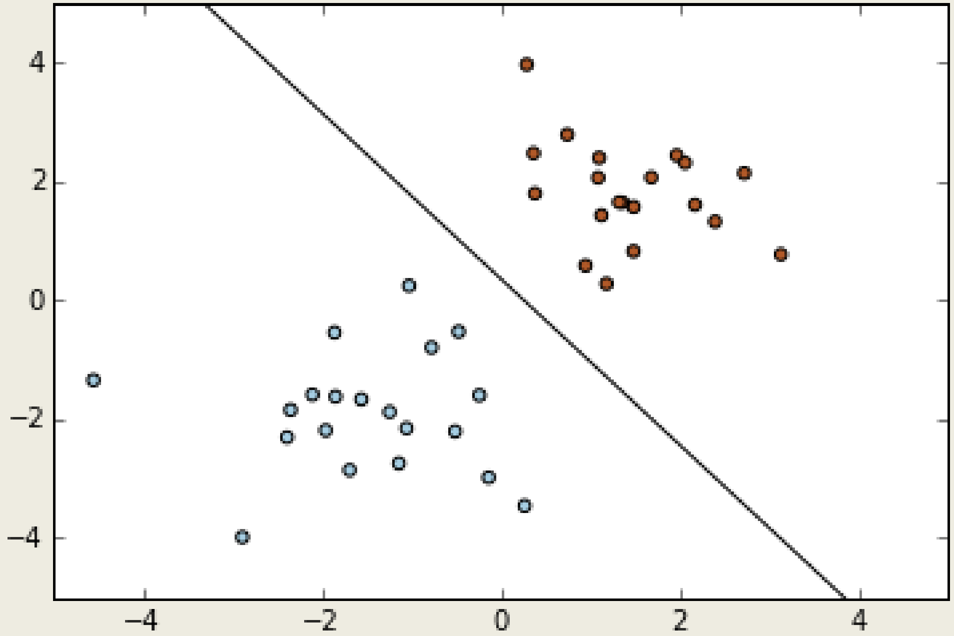
Support Vector Machines

Q: How would you build a classifier for this problem?

A: A line can separate the classes (linearly separable)

Q: Which line is the best? Why?

A: Minimizes the generalization error



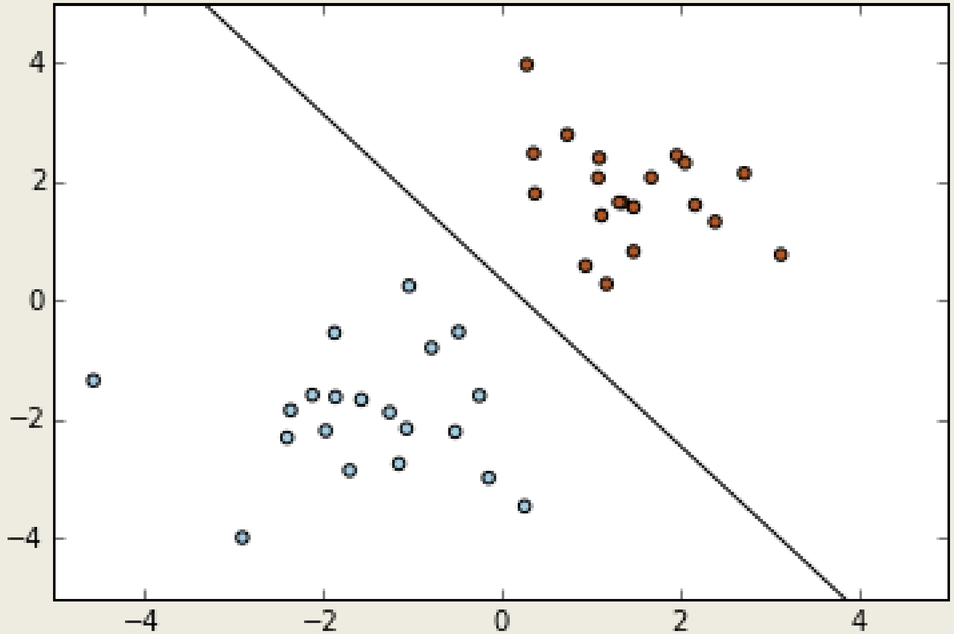
Support Vector Machines

Q: How would you build a classifier for this problem?

A: A line can separate the classes (linearly separable)

Q: Which line is the best? Why?

A: Minimizes the generalization error. Maximizes the margin



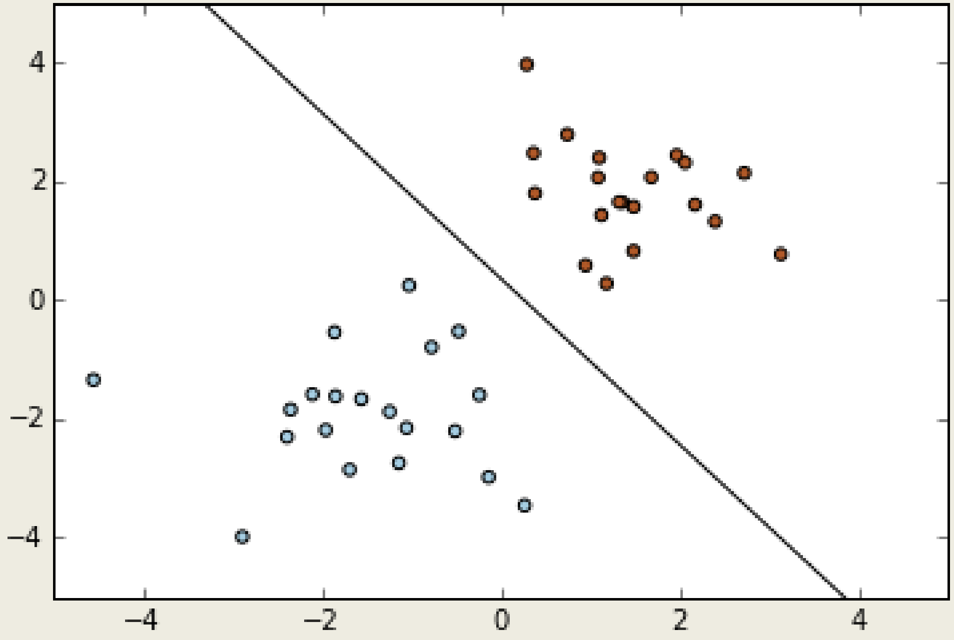
Support Vector Machines

Q: How would you build a classifier for this problem?

A: A line can separate the classes (linearly separable)

Q: Which line is the best? Why?

A: Minimizes the generalization error. Maximizes the margin



Q: What’s the margin?

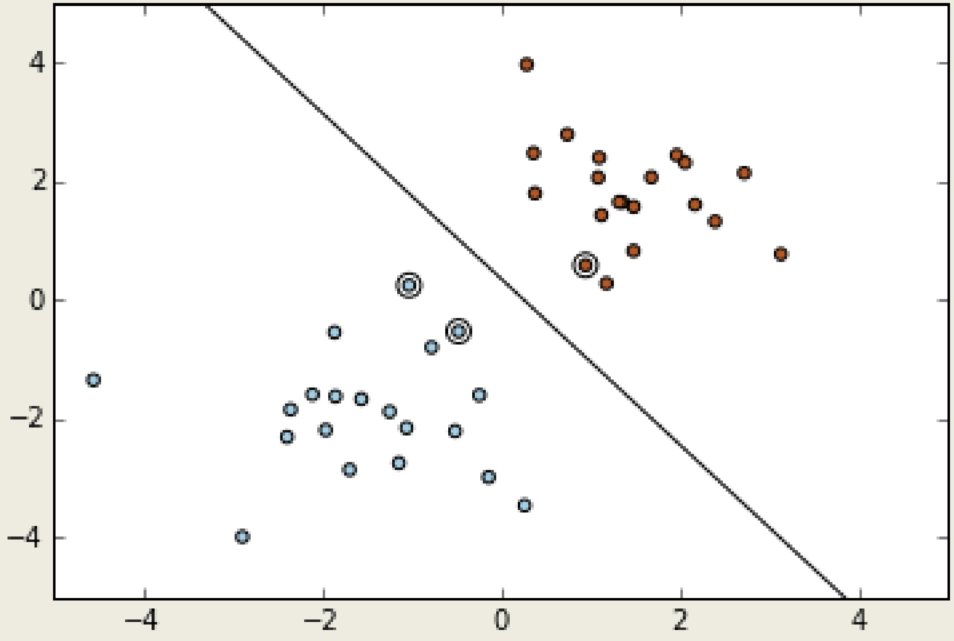
Support Vector Machines

Q: How would you build a classifier for this problem?

A: A line can separate the classes (linearly separable)

Q: Which line is the best? Why?

A: Minimizes the generalization error. Maximizes the margin



Q: What's the margin?
A: Distance from the line to the closest points of each class

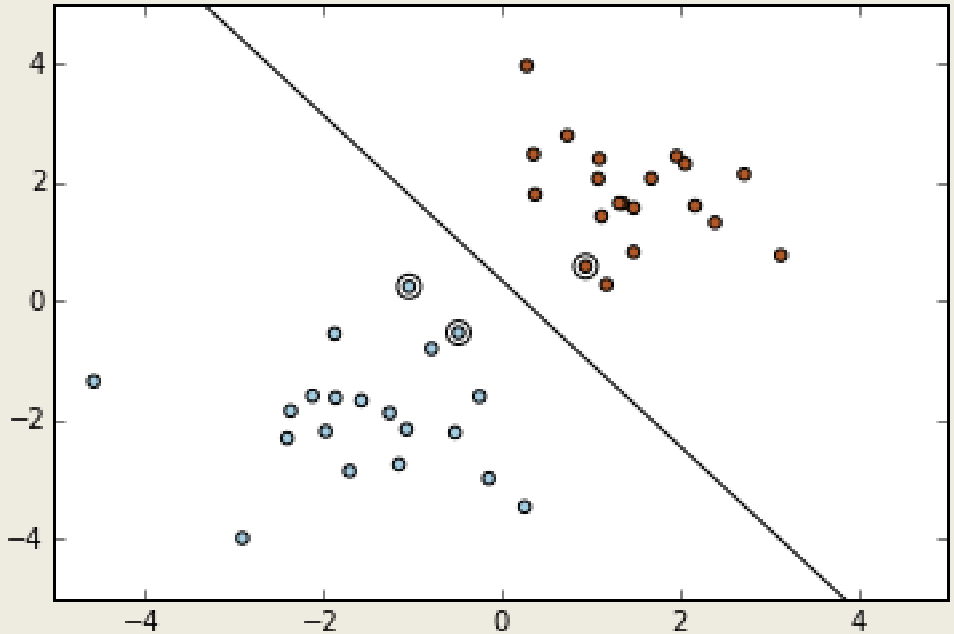
Support Vector Machines

Q: How would you build a classifier for this problem?

A: A line can separate the classes (linearly separable)

Q: Which line is the best? Why?

A: Minimizes the generalization error. Maximizes the margin



Q: What's the margin?
A: Distance from the line to the closest points of each class

These points are called the **Support Vectors**

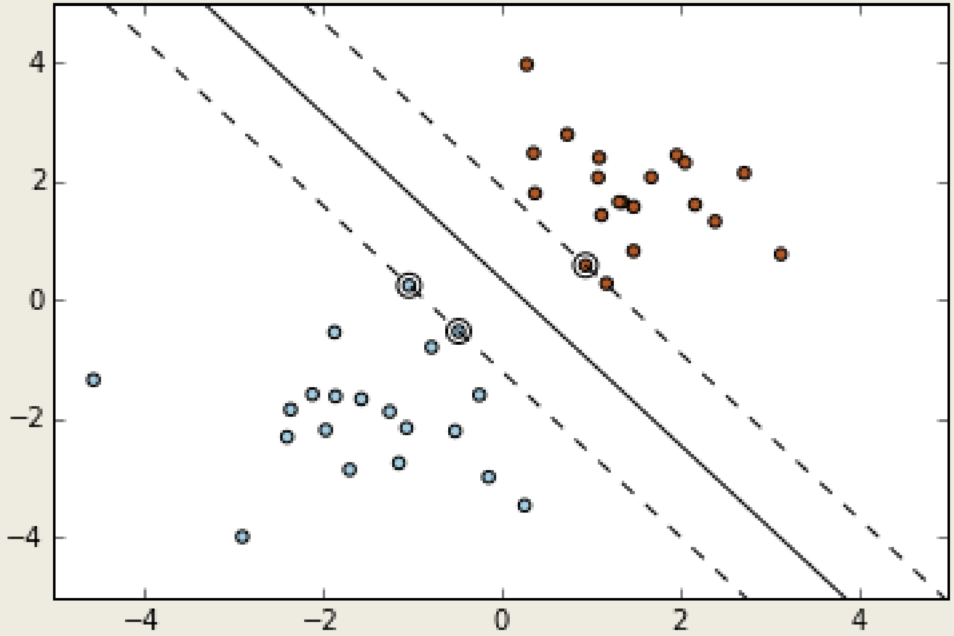
Support Vector Machines

Q: How would you build a classifier for this problem?

A: A line can separate the classes (linearly separable)

Q: Which line is the best? Why?

A: Minimizes the generalization error. Maximizes the margin



Q: What's the margin?
A: Distance from the line to the closest points of each class

These points are called the **Support Vectors**

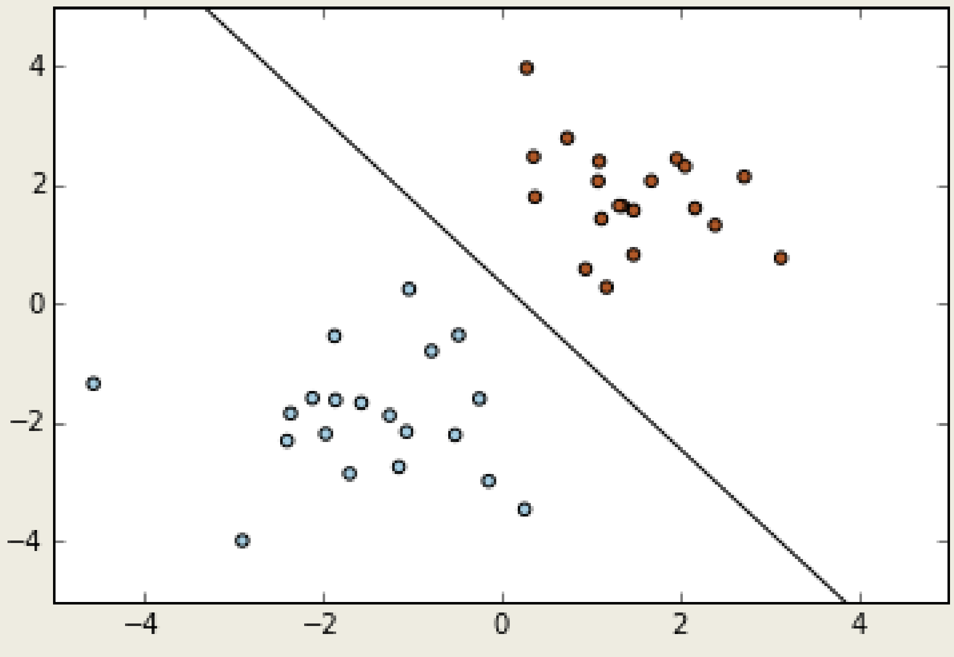
Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

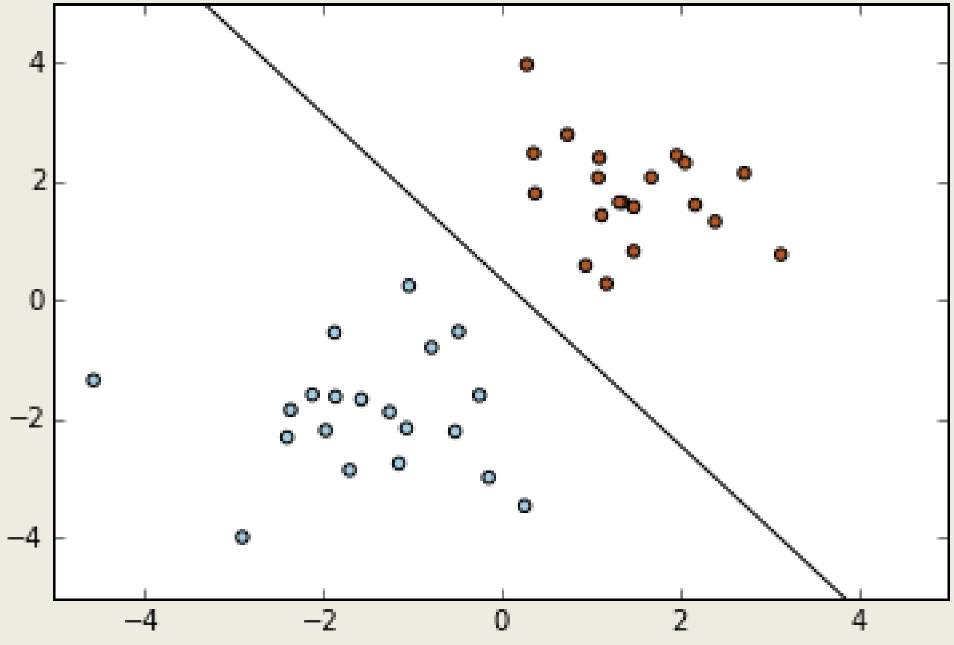
How is this done?



Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

How is this done? (Remember equation of line $f(x) = w^T x + b$)

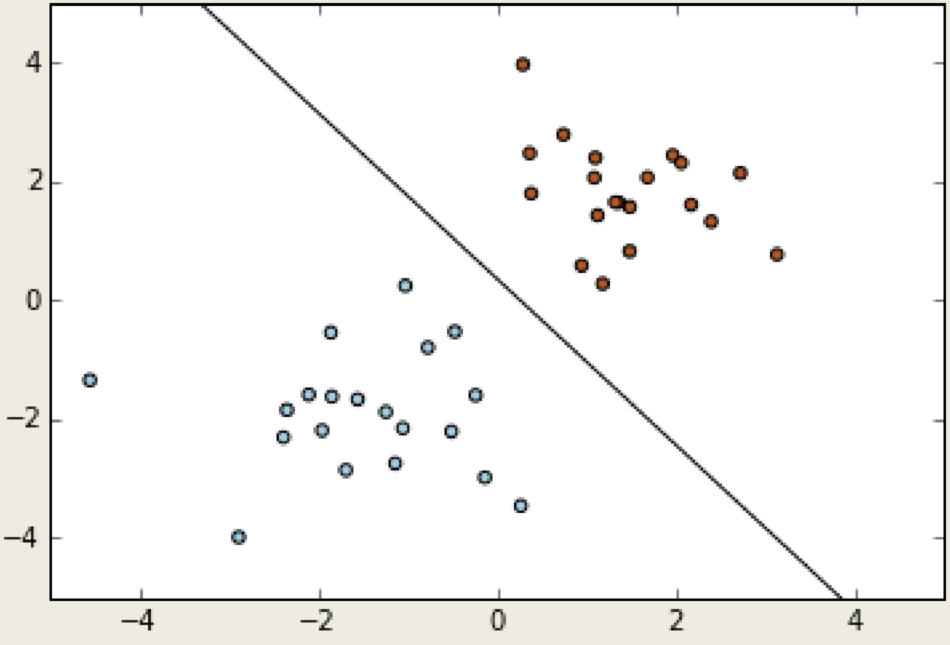


Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

How is this done? (Remember equation of line $f(x) = w^T x+b$)

Set the separating line to:
$w^T x+b = 0$

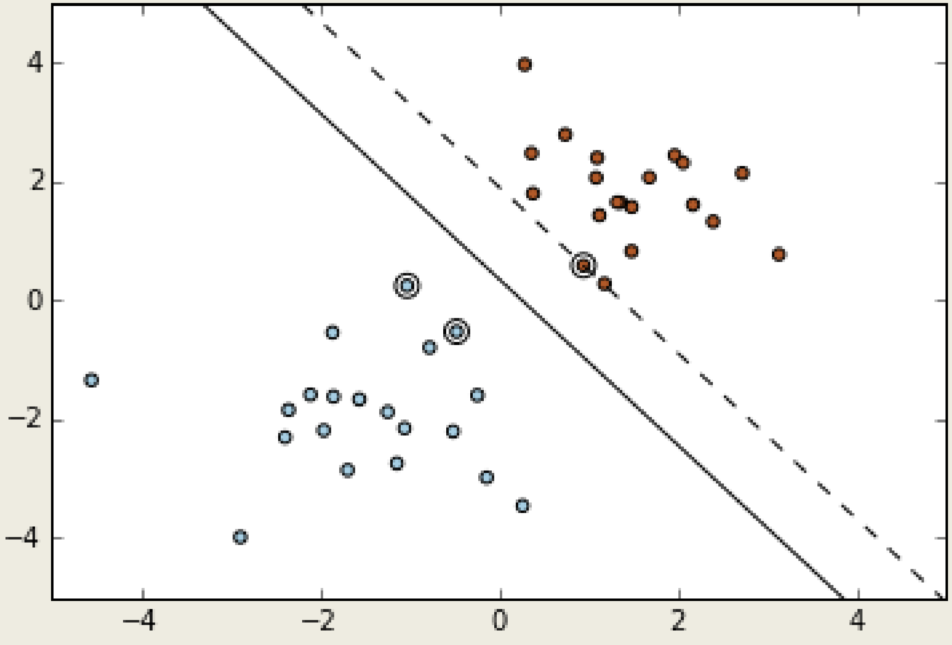


Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

How is this done? (Remember equation of line $f(x) = w^Tx+b$)

Set the separating line to:
$w^Tx+b = 0$



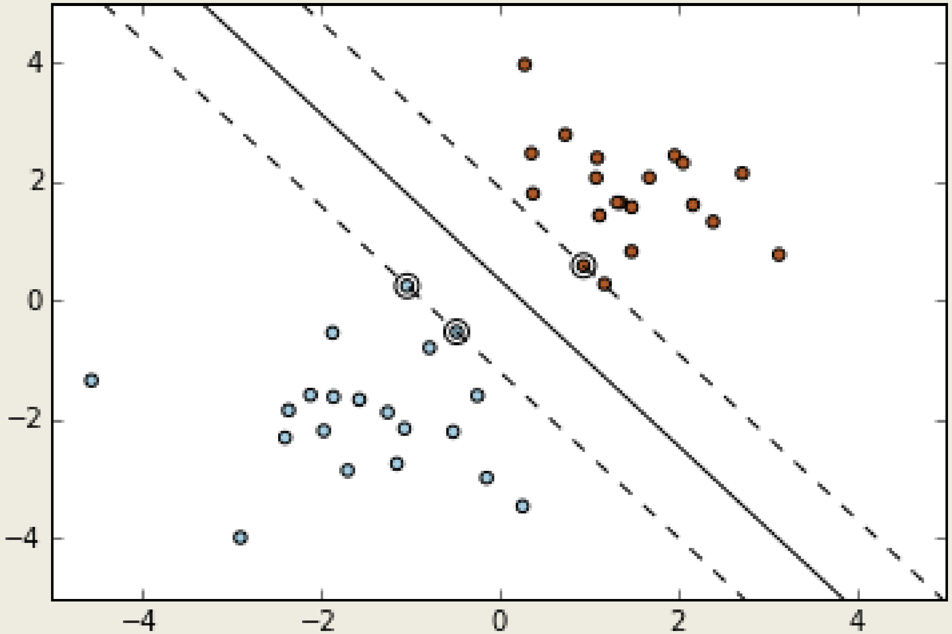
Set class 1 line to :
$w^Tx+b = 1$

Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

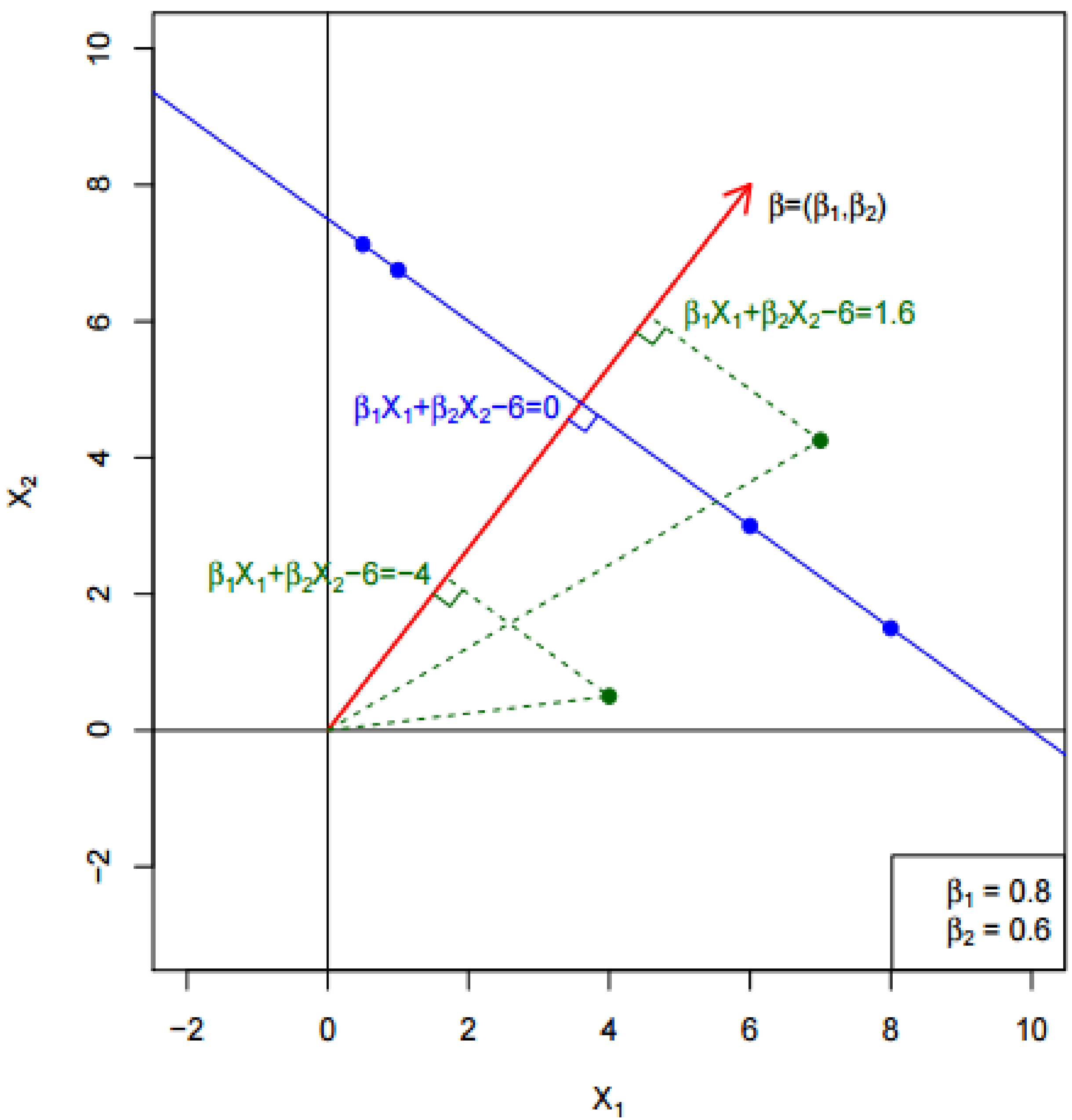
How is this done? (Remember equation of line $f(x) = w^Tx+b$)

Set the separating line to:
$w^Tx+b = 0$



Set class 1 line to :
$w^Tx+b = 1$

Set class -1 line to :
$w^Tx+b = -1$



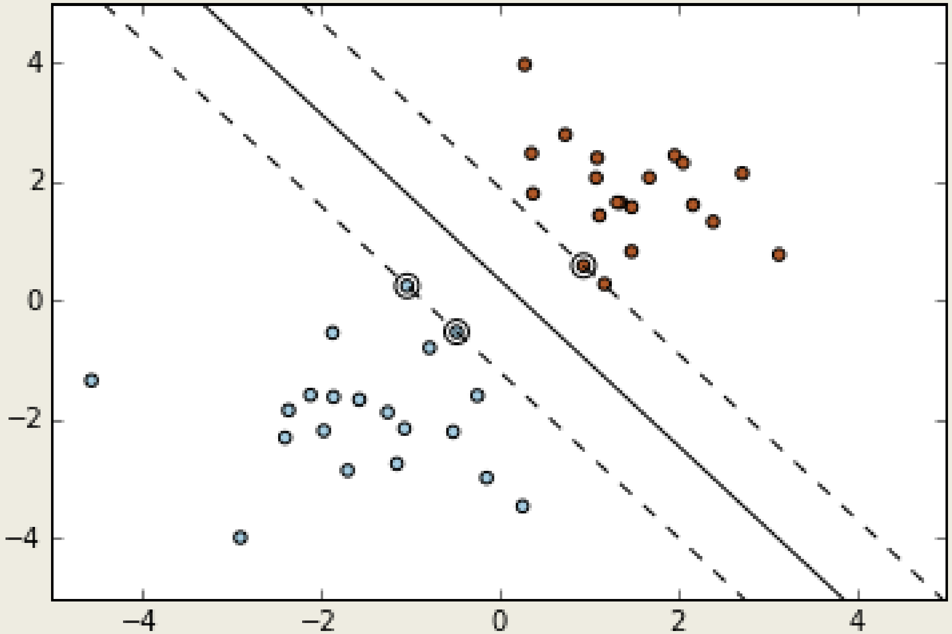
$\beta=(\beta_1,\beta_2)$
$\beta_1X_1+\beta_2X_2-6=1.6$
$\beta_1X_1+\beta_2X_2-6=0$
$\beta_1X_1+\beta_2X_2-6=-4$
$\beta_1 = 0.8$
$\beta_2 = 0.6$
X_1
X_2
−2
0
2
4
6
8
10

Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

How is this done? (Remember equation of line $f(x) = w^T x+b$)

Set the separating line to:
$w^T x+b = 0$



- w is perpendicular to the lines

Set class 1 line to :
$w^T x+b = 1$

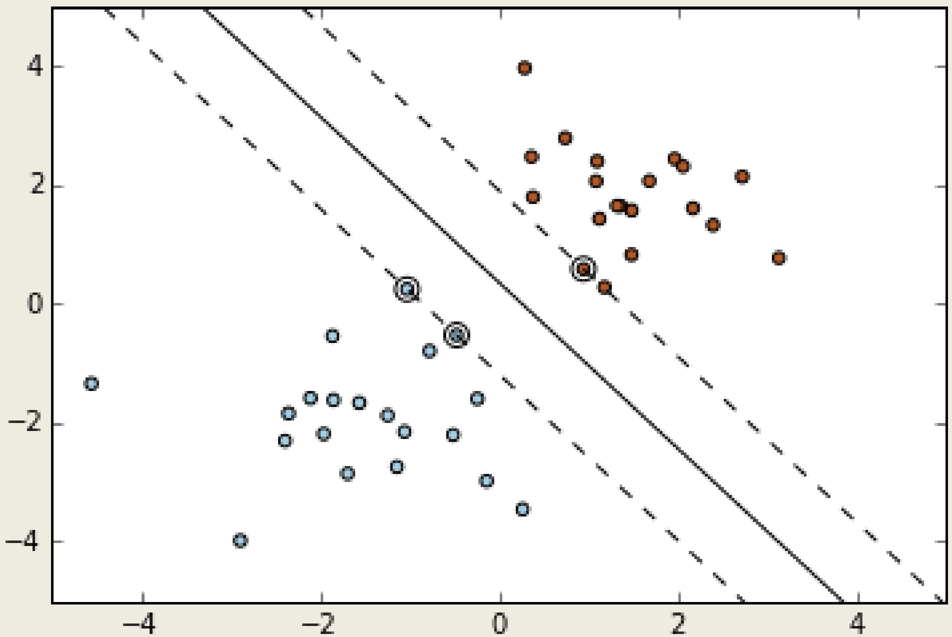
Set class -1 line to :
$w^T x+b = -1$

Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

How is this done? (Remember equation of line $f(x) = w^T x+b$)

Set the separating line to:
$w^T x+b = 0$



Set class -1 line to :
$w^T x+b = -1$

Set class 1 line to :
$w^T x+b = 1$

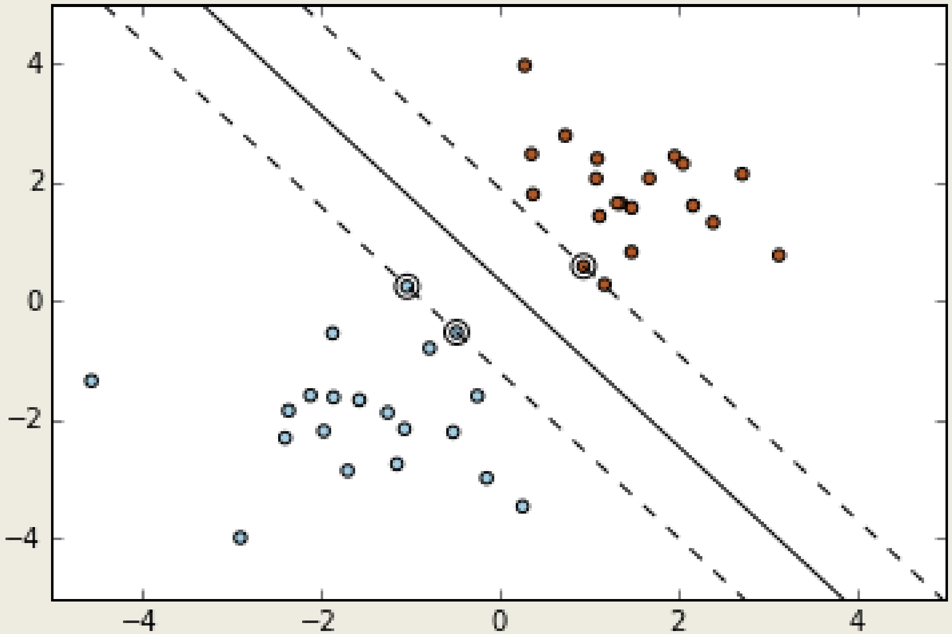
- w is perpendicular to the lines
- Margin $M = 2/||w||$

Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

How is this done? (Remember equation of line $f(x) = w^Tx+b$)

Set the separating line to:
$w^Tx+b = 0$



Set class 1 line to :
$w^Tx+b = 1$

Set class -1 line to :
$w^Tx+b = -1$

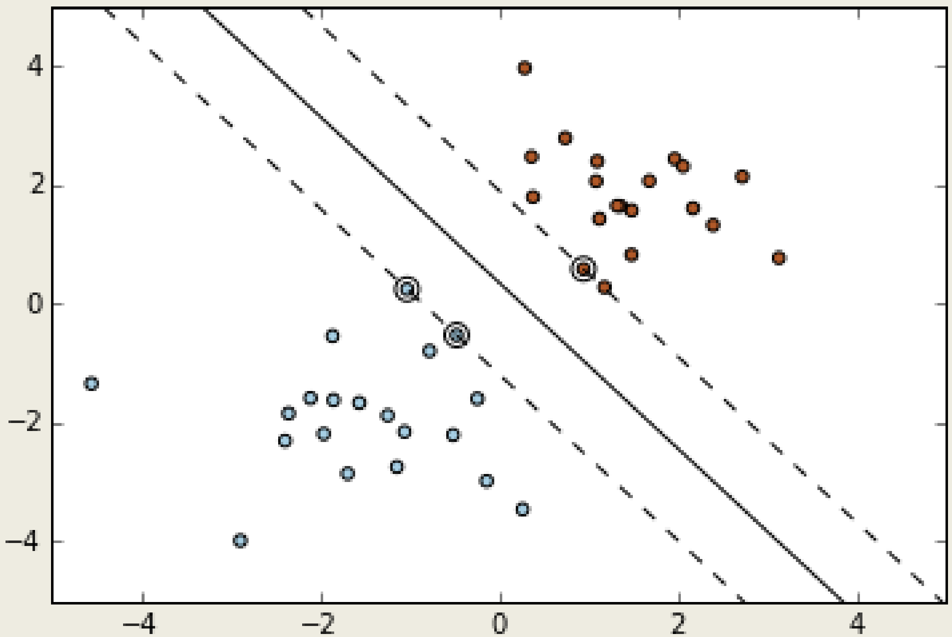
- w is perpendicular to the lines
- Margin $M = 2/||w||$
  - $\text{Max } M = \text{Max } 2/||w||$

Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

How is this done? (Remember equation of line $f(x) = w^Tx+b$)

Set the separating line to:
$w^Tx+b = 0$



Set class -1 line to :
$w^Tx+b = -1$

Set class 1 line to :
$w^Tx+b = 1$

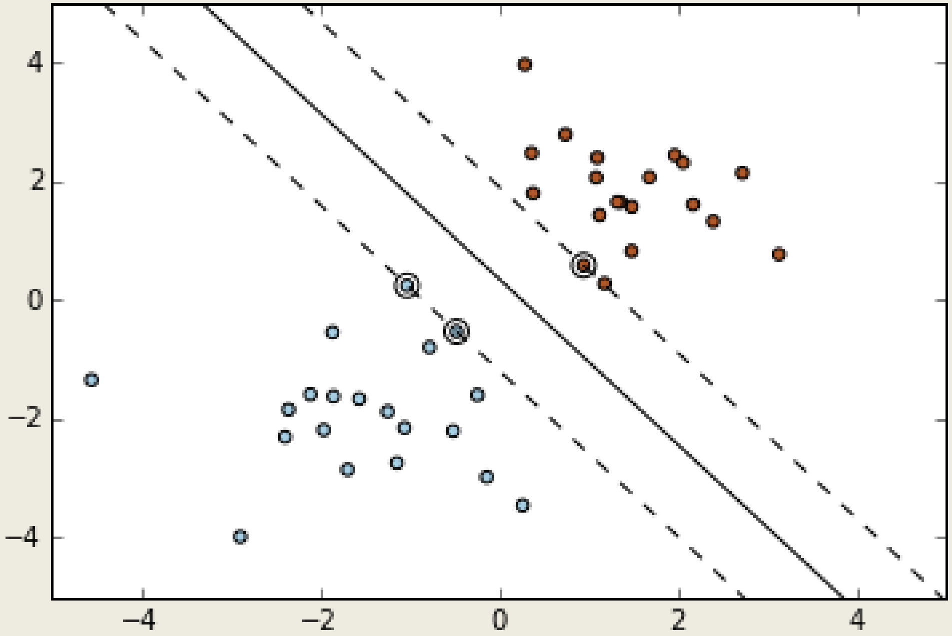
- w is perpendicular to the lines
- Margin $M = 2/||w||$
  - Max M = Max $2/||w||$
- Minimize $||w||^2/2$

Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

How is this done? (Remember equation of line $f(x) = w^T x+b$)

Set the separating line to:
$w^T x+b = 0$



Set class -1 line to :
$w^T x+b = -1$

- w is perpendicular to the lines
- Margin $M = 2/||w||$
    - Max M = Max $2/||w||$
- Minimize $||w||^2/2$
- s.t. $y_i(w^T x_i + b) >= 1$

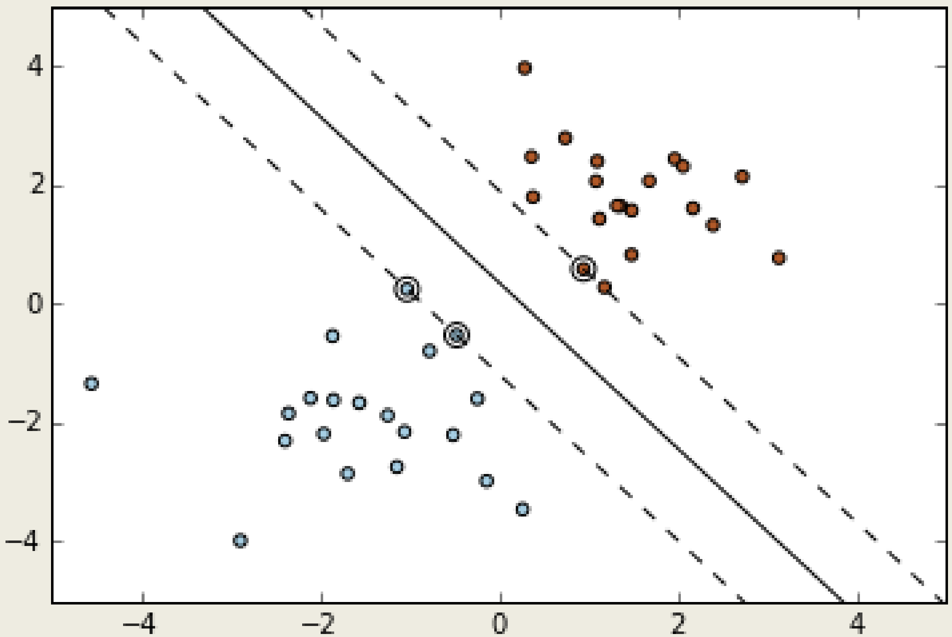
Set class 1 line to :
$w^T x+b = 1$

Support Vector Machines

Support Vector Machine: Binary linear classifier with the maximum margin. Let the classes be {-1,+1}

How is this done? (Remember equation of line $f(x) = w^Tx+b$)

Set the separating line to:
$w^Tx+b = 0$



Set class -1 line to :
$w^Tx+b = -1$

- w is perpendicular to the lines
- Margin $M = 2/||w||$
  - Max M = Max $2/||w||$
- Minimize $||w||^2/2$
- s.t. $y_i(w^Tx_i + b) >= 1$
- **Classify:**$f(x)=\text{sign}(w^Tx+b)$

Set class 1 line to :
$w^Tx+b = 1$

Support Vector Machines

- Objective:
  - Minimize $||w||^2/2$
  - S.T. $y_i(w^T x_i + b) >= 1$
- It turns out that we can reformulate this problem to:
  - Max $\Sigma_i \alpha_i - 1/2\ \Sigma_{ij} \alpha_i \alpha_j y_i y_j x_i^T x_j$
  - s.t. $\alpha_i >= 0$ and $\Sigma_i \alpha_i y_i = 0$
- Then $w = \Sigma_i \alpha_i y_i x_i$
- It turns out that $\alpha_i = 0$ for non support vectors
- Now we can classify by: $f(x) = \text{sign}(\Sigma_i \alpha_i y_i <x_i, x>)$
- Note that this is line Nearest Neighbors where the neighbors have been selected for you

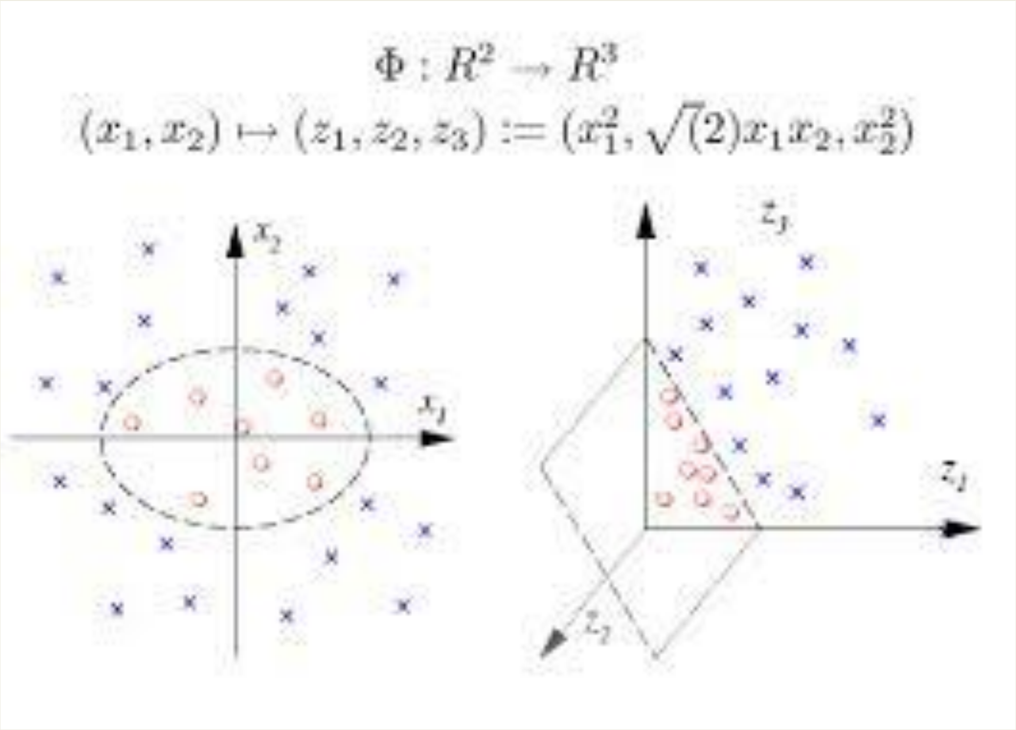
Support Vector Machines

Q: What happens if our data is not linearly separable?

Support Vector Machines

Q: What happens if our data is not linearly separable?

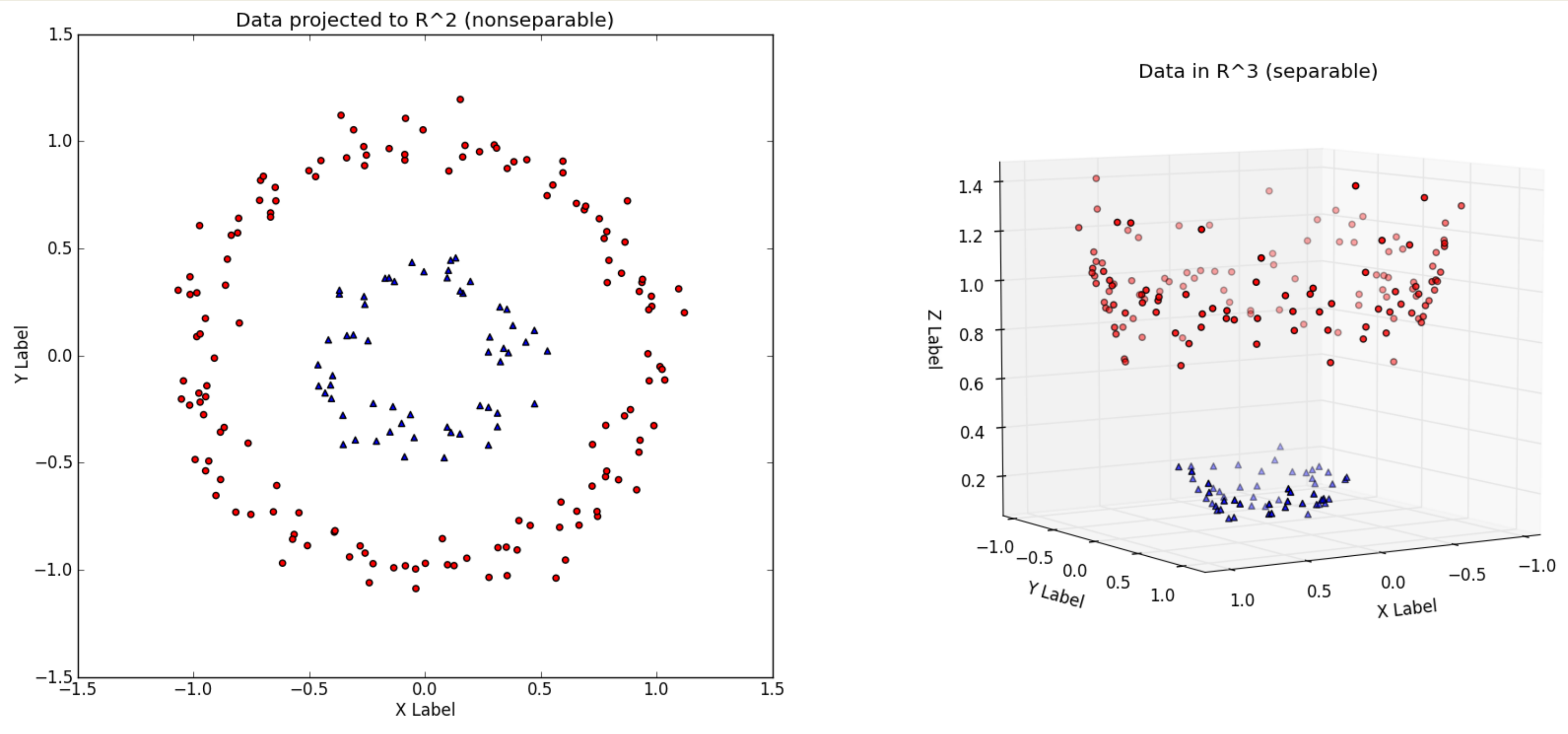
A: Transform the features to a higher dimensional space where the data is linearly separable



Support Vector Machines

Q: What happens if our data is not linearly separable?

A: Transform the features to a higher dimensional space where the data is linearly separable



Support Vector Machines

- These feature transformations can be computationally difficult

Support Vector Machines

- These feature transformations can be computationally difficult
- We are saved by the following:

Support Vector Machines

- These feature transformations can be computationally difficult
- We are saved by the following:
  - We don’t need to know all the points just their inner products

Support Vector Machines

- These feature transformations can be computationally difficult
- We are saved by the following:
  - We don't need to know all the points just their inner products
  - If z_1,z are the transformations of the original x_1,x it can be shown that $<z_1,z> = K(x_1,x)$ where K is something called a kernel function

Support Vector Machines

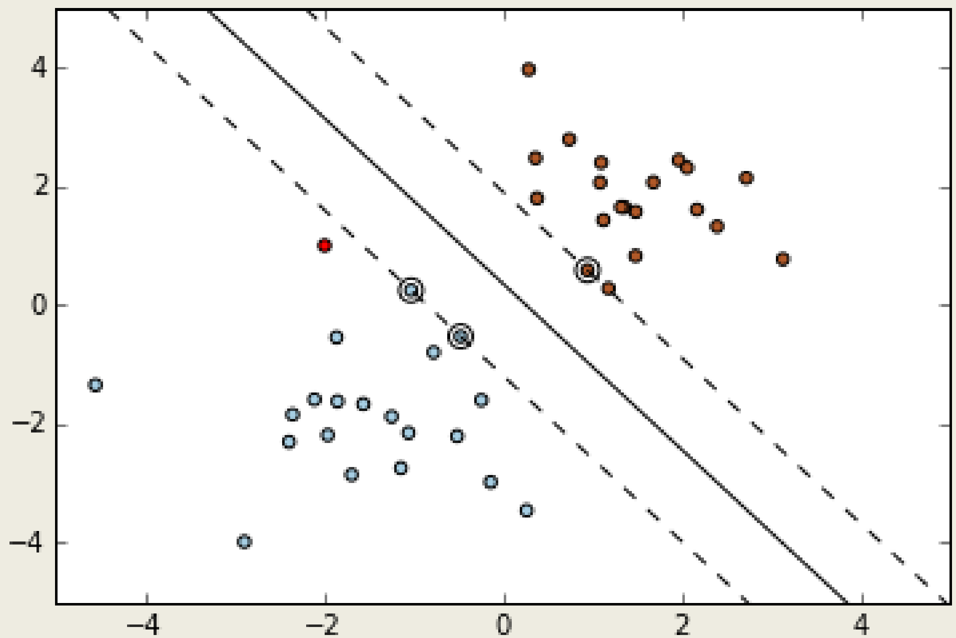
- These feature transformations can be computationally difficult
- We are saved by the following:
  - We don't need to know all the points just their inner products
  - If z_1,z are the transformations of the original x_1,x it can be shown that $<z_1,z> = K(x_1,x)$ where K is something called a kernel function
- Common Kernel functions:
  - Gaussian: $K(x_1,x) = \exp(-||x_1-x||^2/(2\sigma^2))$
  - Linear: $K(x_1,x) = <x_1,x>$
  - Polynomial: $K(x_1,x) = \exp(<x_1,x> + a)^r$

Support Vector Machines

- These feature transformations can be computationally difficult
- We are saved by the following:
  - We don't need to know all the points just their inner products
  - If z_1,z are the transformations of the original x_1,x it can be shown that $\langle z_1, z \rangle = K(x_1,x)$ where K is something called a kernel function
- Common Kernel functions:
  - Gaussian: $K(x_1,x) = \exp(-||x_1-x||^2/(2\sigma^2))$
  - Linear: $K(x_1,x) = \langle x_1,x \rangle$
  - Polynomial: $K(x_1,x) = \exp(\langle x_1,x \rangle + a)^r$
- **Our new classifier is now $f(x) = \mathrm{sign}(\Sigma_i \alpha_i y_i K(x_i,x))$**

Support Vector Machines

Q: What happens if our data is not linearly separable as in this case?

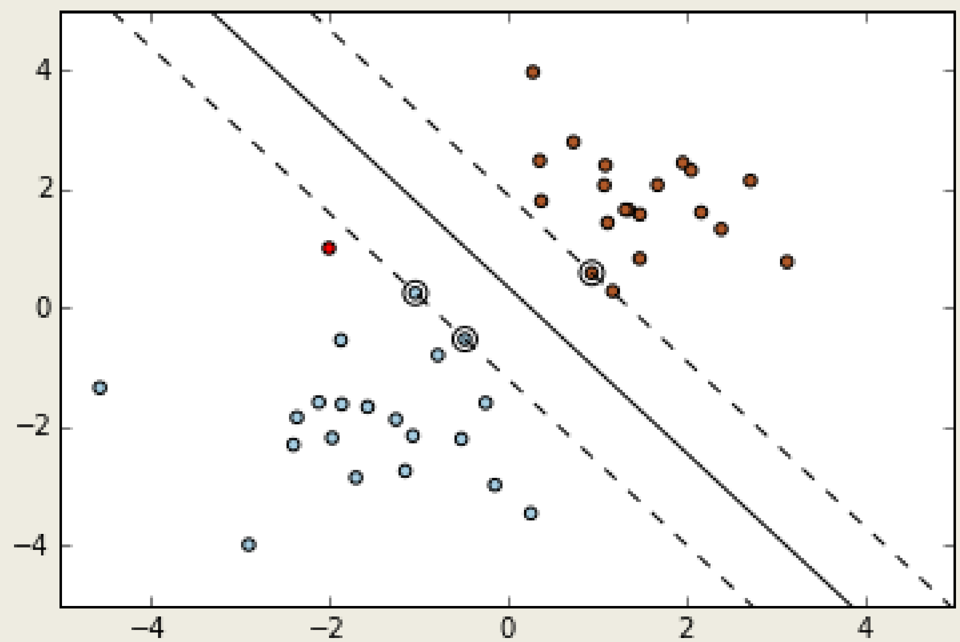


Support Vector Machines

Q: What happens if our data is not linearly separable as in this case?

A: We introduce the idea of a slack variable:

- ζ_i is a slack variable
- Now solve min $||w||^2 + C\Sigma_i \zeta_i$
- Now the constraint is $y_i(w^T x_i + b) >= 1-\zeta_i$ where $\zeta_i >= 0$



Summarize

- We want to find the maximum margin classifier

Summarize

- We want to find the maximum margin classifier
- This is done via min $||w||^2/2$ s.t. $y_i(w^Tx_i + b) >= 1$

Summarize

- We want to find the maximum margin classifier
- This is done via min $||w||^2/2$ s.t. $y_i(w^Tx_i + b) >= 1$
- We can reformulate this as:
  - Max $\Sigma_i\alpha_i - 1/2\ \Sigma_{ij}\alpha_i\alpha_j y_i y_j x_i^T x_j$
  - s.t. $\alpha_i >= 0$ and $\Sigma_i\alpha_i y_i = 0$
  - Where $\alpha_i = 0$ for non support vectors. Example selection

Summarize

- We want to find the maximum margin classifier
- This is done via min $||w||^2/2$ s.t. $y_i(w^T x_i + b) >= 1$
- We can reformulate this as:
  - Max $\Sigma_i \alpha_i - 1/2\ \Sigma_{ij} \alpha_i \alpha_j y_i y_j x_i^T x_j$
  - s.t. $\alpha_i >= 0$ and $\Sigma_i \alpha_i y_i = 0$
  - Where $\alpha_i = 0$ for non support vectors. Example selection
- When the decision boundary is non-linear we use kernels to find the inner product in higher dimensional spaces where the data is linearly separable

Summarize

- We want to find the maximum margin classifier
- This is done via min $||w||^2/2$ s.t. $y_i(w^T x_i + b) >= 1$
- We can reformulate this as:
  - Max $\Sigma_i \alpha_i - 1/2\ \Sigma_{ij} \alpha_i \alpha_j y_i y_j x_i^T x_j$
  - s.t. $\alpha_i >= 0$ and $\Sigma_i \alpha_i y_i = 0$
  - Where $\alpha_i = 0$ for non support vectors. Example selection
- When the decision boundary is non-linear we use kernels to find the inner product in higher dimensional spaces where the data is linearly separable
- When we have noisy data we introduced slack variables, ζ_i, so the new optimization problem becomes:
  - solve min $||w||^2 + C\Sigma_i \zeta_i$
  - S.t. $y_i(w^T x_i + b) >= 1-\zeta_i$ and $\zeta_i >= 0$

Summarize

- We want to find the maximum margin classifier
- This is done via min $||w||^2/2$ s.t. $y_i(w^T x_i + b) >= 1$
- We can reformulate this as:
  - Max $\Sigma_i \alpha_i - 1/2\ \Sigma_{ij} \alpha_i \alpha_j y_i y_j x_i^T x_j$
  - s.t. $\alpha_i >= 0$ and $\Sigma_i \alpha_i y_i = 0$
  - Where $\alpha_i = 0$ for non support vectors. Example selection
- When the decision boundary is non-linear we use kernels to find the inner product in higher dimensional spaces where the data is linearly separable
- When we have noisy data we introduced slack variables, ζ_i, so the new optimization problem becomes:
  - solve min $||w||^2 + C\Sigma_i \zeta_i$
  - S.t. $y_i(w^T x_i + b) >= 1-\zeta_i$ and $\zeta_i >= 0$
- We can classify new points with $\mathbf{f(x) = sign(\Sigma_i \alpha_i y_i K(x_i,x))}$

Questions?