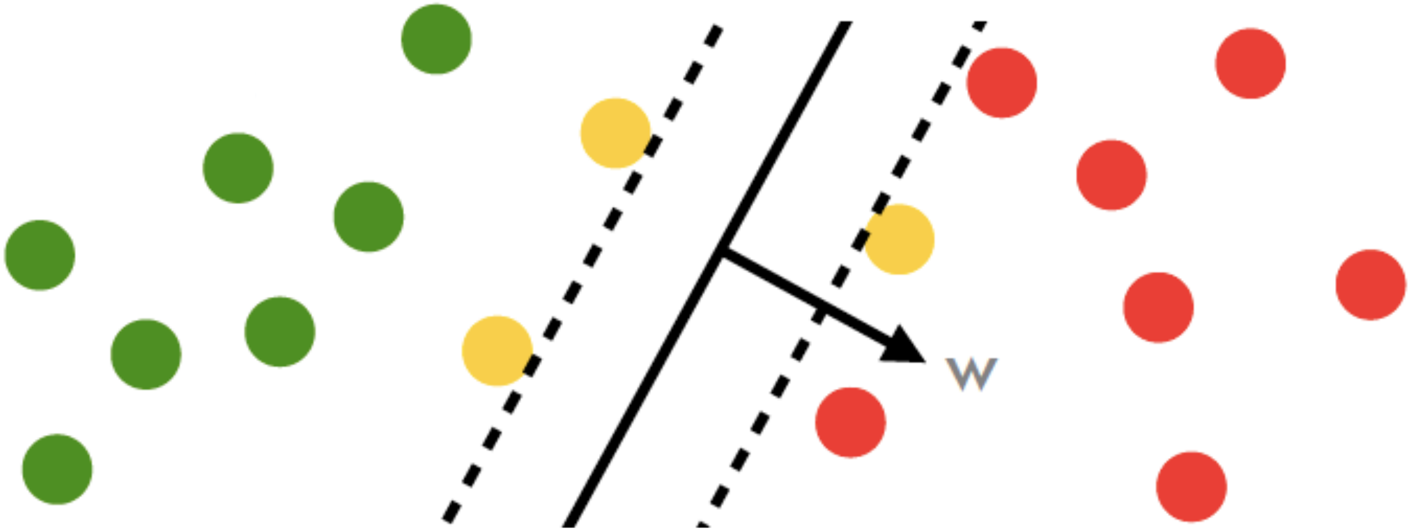
Artificial Intelligence and Machine Learning

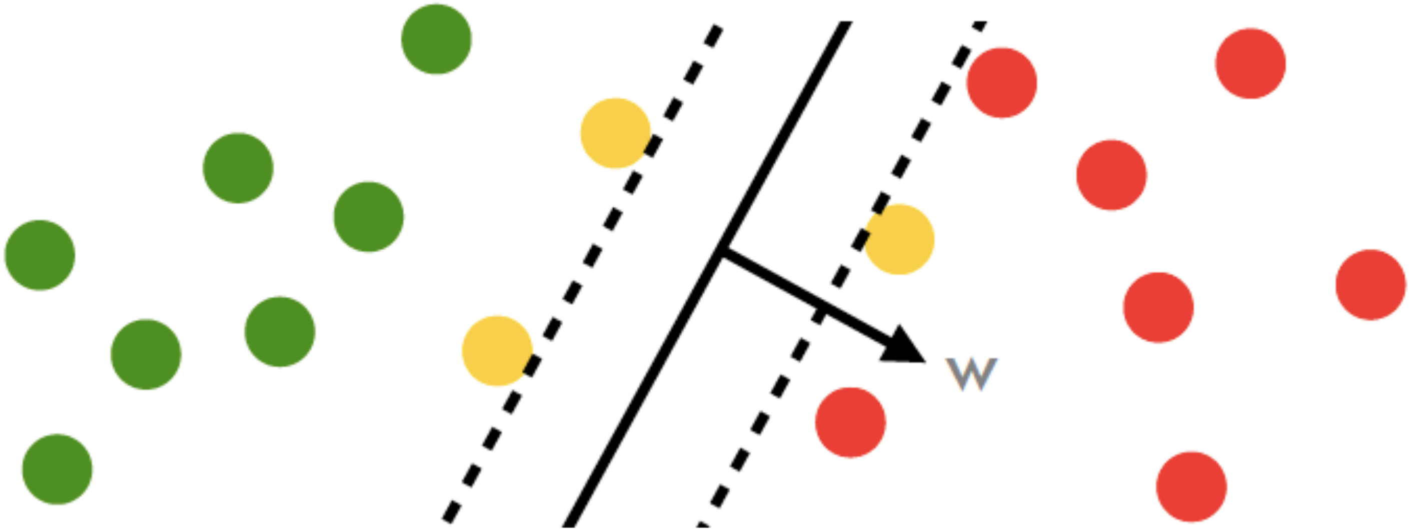
Barbara Caputo

Support Vector Machines



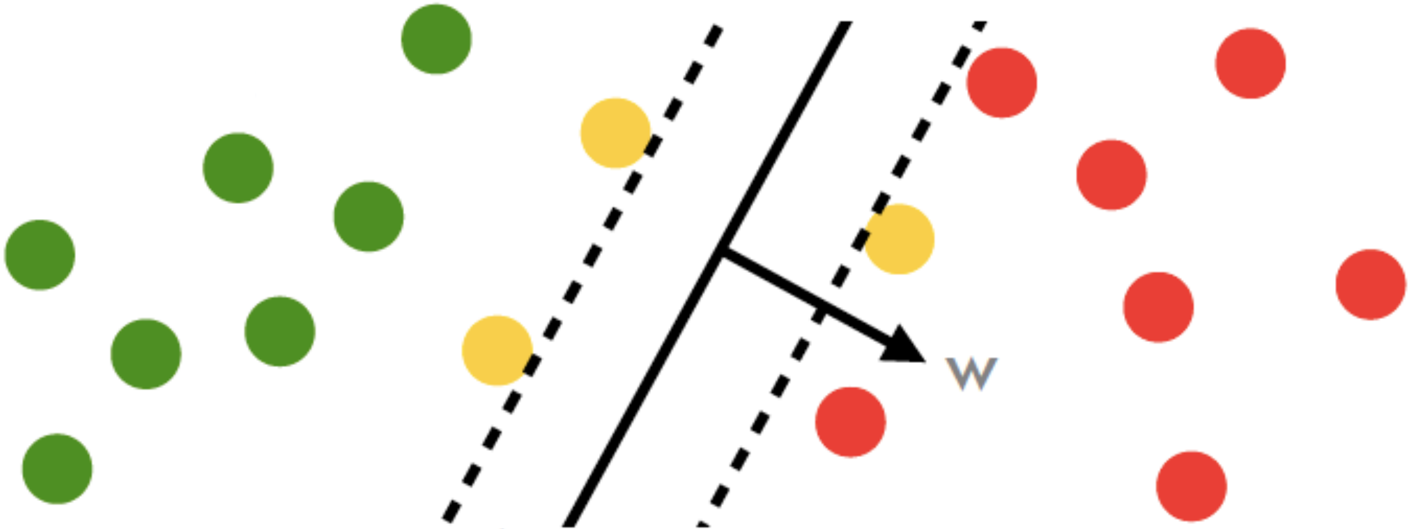
Support Vector Machines

$$\underset{w.b}{\text{minimize}} \frac{1}{2} \|w\|^2 \text{ subject to } y_i \left[\langle x_i, w\rangle + b\right] \geq 1$$



Support Vector Machines

$$\underset{w.b}{\text{minimize}}\ \frac{1}{2}\|w\|^2 \ \text{subject to}\ y_i\left[\langle x_i, w\rangle + b\right] \geq 1$$



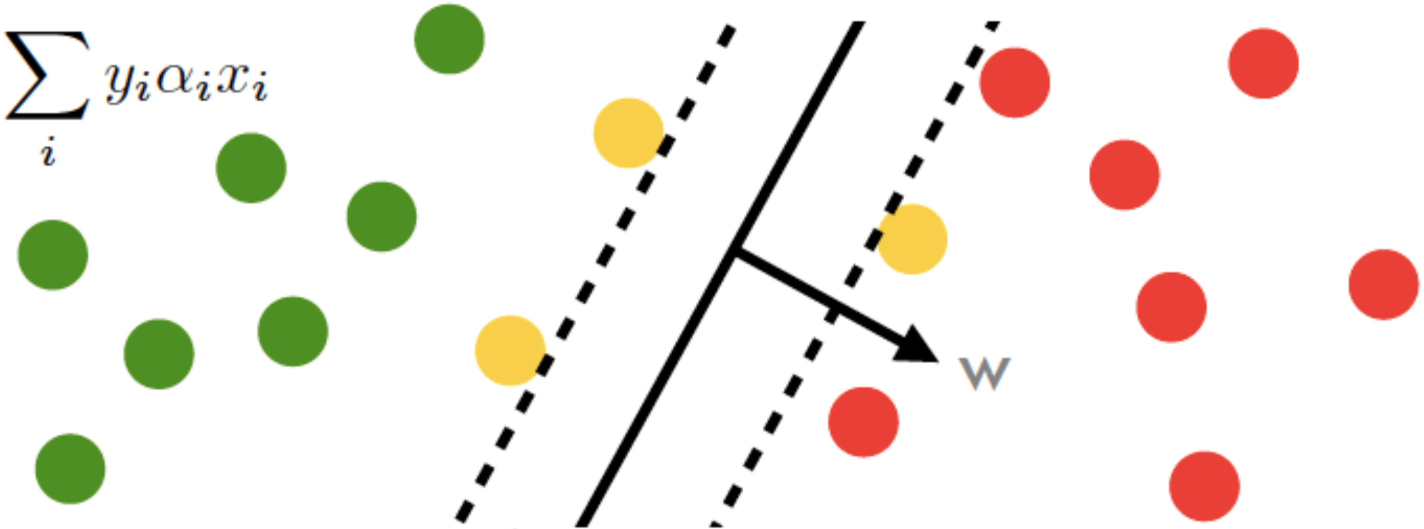
$$\underset{\alpha}{\text{maximize}} - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \langle x_i, x_j\rangle + \sum_i \alpha_i$$

$$\text{subject to}\ \sum \alpha_i y_i = 0 \ \text{and}\ \alpha_i \geq 0$$

Support Vector Machines

$$\underset{w,b}{\text{minimize}} \frac{1}{2} \|w\|^2 \text{ subject to } y_i \left[\langle x_i, w \rangle + b\right] \geq 1$$

$$w = \sum_i y_i \alpha_i x_i$$



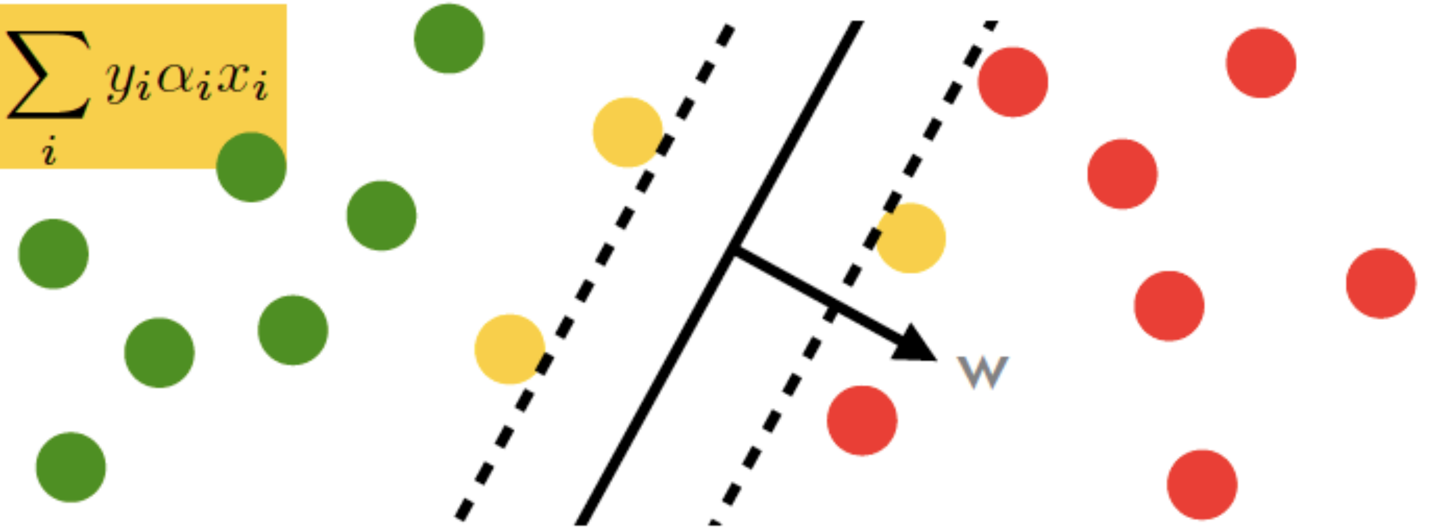
$$\underset{\alpha}{\text{maximize}} - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle + \sum_i \alpha_i$$

$$\text{subject to } \sum \alpha_i y_i = 0 \text{ and } \alpha_i \geq 0$$

Support Vectors

$$\underset{w,b}{\text{minimize}}\ \frac{1}{2}\|w\|^2 \text{ subject to } y_i\left[\langle x_i, w\rangle + b\right] \geq 1$$

$$w = \sum_i y_i \alpha_i x_i$$



Karush Kuhn Tucker

Optimality condition

$$\alpha_i\left[y_i\left[\langle w, x_i\rangle + b\right] - 1\right] = 0$$

$$\alpha_i = 0$$

$$\alpha_i > 0 \Longrightarrow y_i\left[\langle w, x_i\rangle + b\right] = 1$$

Soft Margin Classifiers

Adding slack variables

- Hard margin problem

Adding slack variables

- Hard margin problem

$$\underset{w,b}{\text{minimize}}\ \frac{1}{2}\|w\|^2 \text{ subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1$$

Adding slack variables

- Hard margin problem

$$\underset{w,b}{\text{minimize}}\ \frac{1}{2}\|w\|^2 \ \text{ subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1$$

- With slack variables

Adding slack variables

- Hard margin problem

$$\underset{w,b}{\text{minimize}}\,\frac{1}{2}\|w\|^2 \text{ subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1$$

- With slack variables

$$\underset{w,b}{\text{minimize}}\;\frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

Adding slack variables

- Hard margin problem

$$\underset{w,b}{\text{minimize}}\ \frac{1}{2}\|w\|^2 \text{ subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1$$

- With slack variables

$$\underset{w,b}{\text{minimize}}\ \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

$$\text{subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

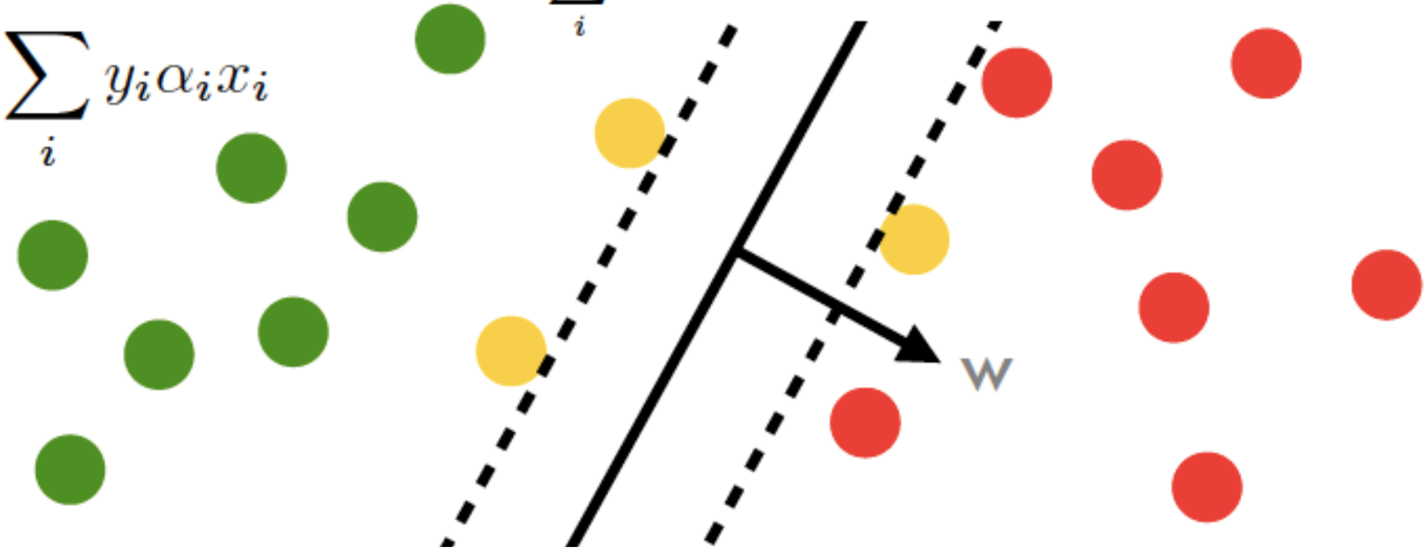
Problem is always feasible.

Karush Kuhn Tucker Conditions

$$\underset{\alpha}{\text{maximize}} - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle + \sum_i \alpha_i$$

$$\text{subject to } \sum_i \alpha_i y_i = 0 \text{ and } \alpha_i \in [0, C]$$

$$w = \sum_i y_i \alpha_i x_i$$

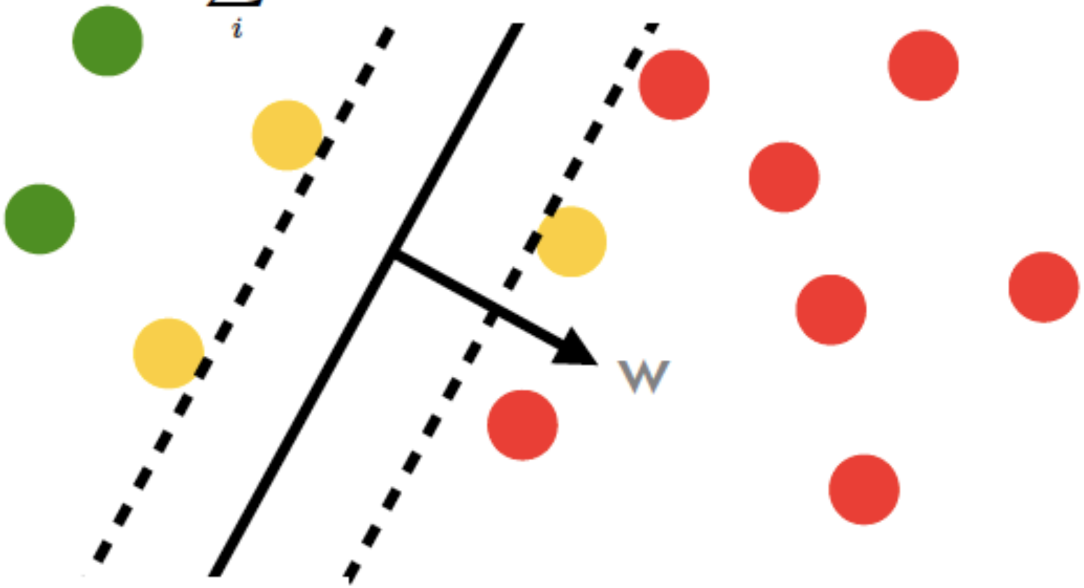


Karush Kuhn Tucker Conditions

$$\underset{\alpha}{\text{maximize}} -\frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \langle x_i, x_j\rangle + \sum_i \alpha_i$$

$$\text{subject to } \sum_i \alpha_i y_i = 0 \text{ and } \alpha_i \in [0, C]$$

$$w = \sum_i y_i \alpha_i x_i$$



$$\alpha_i \left[y_i \left[\langle w, x_i\rangle + b\right] + \xi_i - 1\right] = 0$$

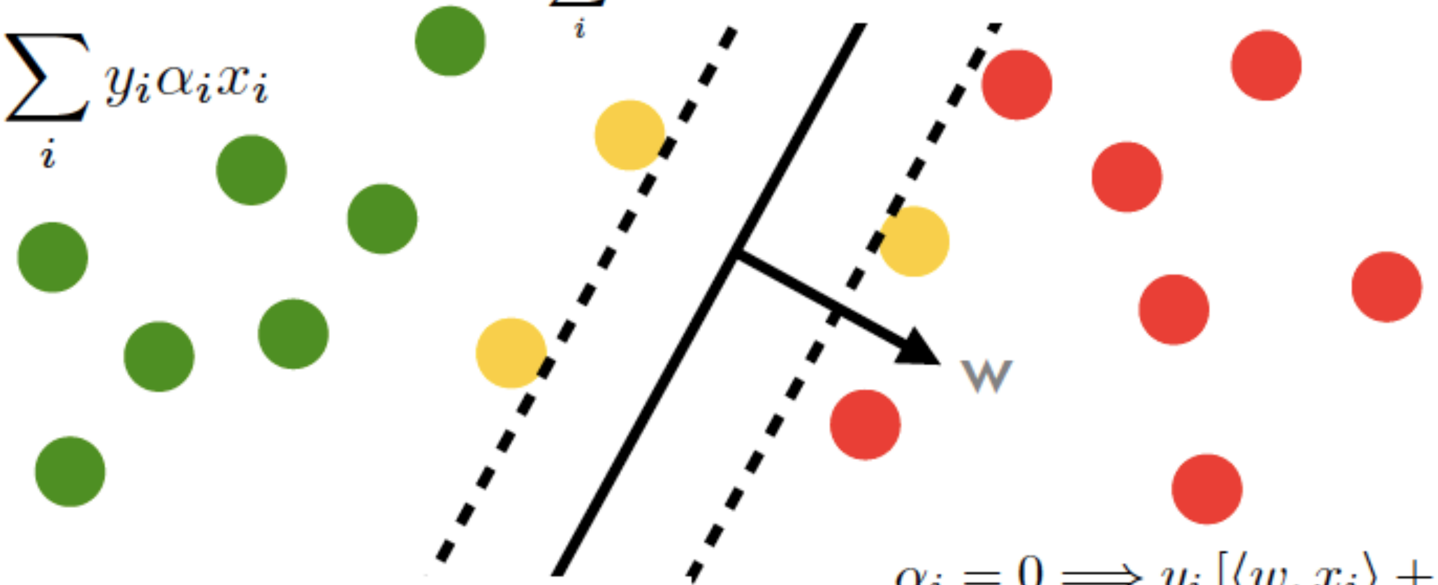
$$\eta_i \xi_i = 0$$

Karush Kuhn Tucker Conditions

$$\underset{\alpha}{\text{maximize}} -\frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \langle x_i, x_j\rangle + \sum_i \alpha_i$$

$$\text{subject to } \sum_i \alpha_i y_i = 0 \text{ and } \alpha_i \in [0, C]$$

$$w = \sum_i y_i \alpha_i x_i$$



$$\alpha_i \left[y_i \left[\langle w, x_i\rangle + b\right] + \xi_i - 1\right] = 0$$

$$\eta_i \xi_i = 0$$

$$\alpha_i = 0 \Longrightarrow y_i \left[\langle w, x_i\rangle + b\right] \geq 1$$

$$0 < \alpha_i < C \Longrightarrow y_i \left[\langle w, x_i\rangle + b\right] = 1$$

$$\alpha_i = C \Longrightarrow y_i \left[\langle w, x_i\rangle + b\right] \leq 1$$

Nonlinear Separation

The Kernel Trick

The Kernel Trick

- Linear soft margin problem

$$\underset{w,b}{\text{minimize}}\ \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

The Kernel Trick

- Linear soft margin problem

$$\underset{w,b}{\text{minimize}}\ \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

$$\text{subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

- Dual problem

The Kernel Trick

- Linear soft margin problem

$$\underset{w,b}{\text{minimize}} \; \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

$$\text{subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

- Dual problem

$$\underset{\alpha}{\text{maximize}} - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \langle x_i, x_j\rangle + \sum_i \alpha_i$$

The Kernel Trick

- **Linear soft margin problem**

$$\underset{w,b}{\text{minimize}} \ \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

$$\text{subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

- **Dual problem**

$$\underset{\alpha}{\text{maximize}} - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \langle x_i, x_j\rangle + \sum_i \alpha_i$$

$$\text{subject to } \sum_i \alpha_i y_i = 0 \text{ and } \alpha_i \in [0, C]$$

- **Support vector expansion**

$$f(x) = \sum_i \alpha_i y_i \langle x_i, x\rangle + b$$

The Kernel Trick

- Linear soft margin problem

$$\underset{w,b}{\text{minimize}} \ \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

$$\text{subject to } y_i\left[\langle w, \phi(x_i)\rangle + b\right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

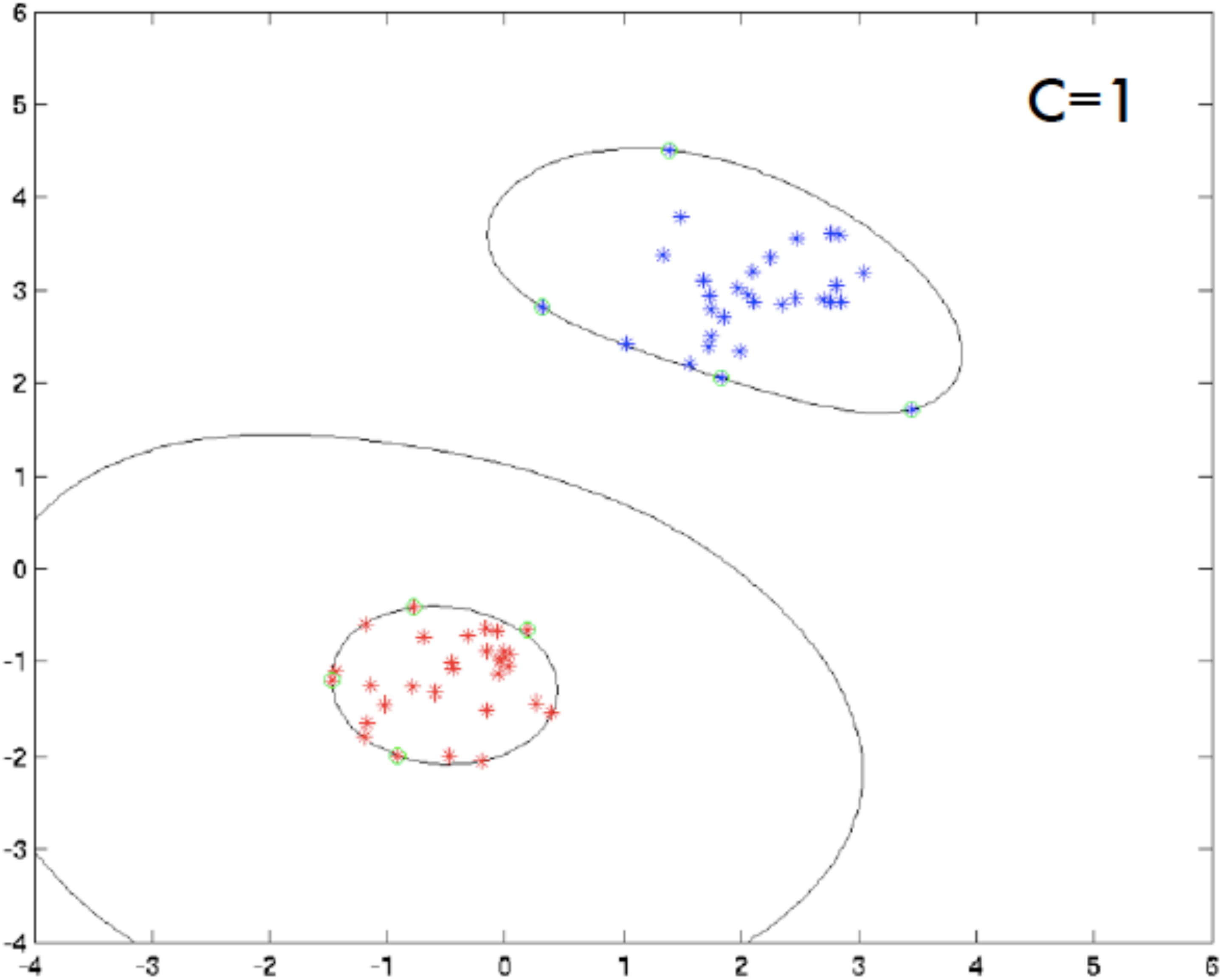
- Dual problem

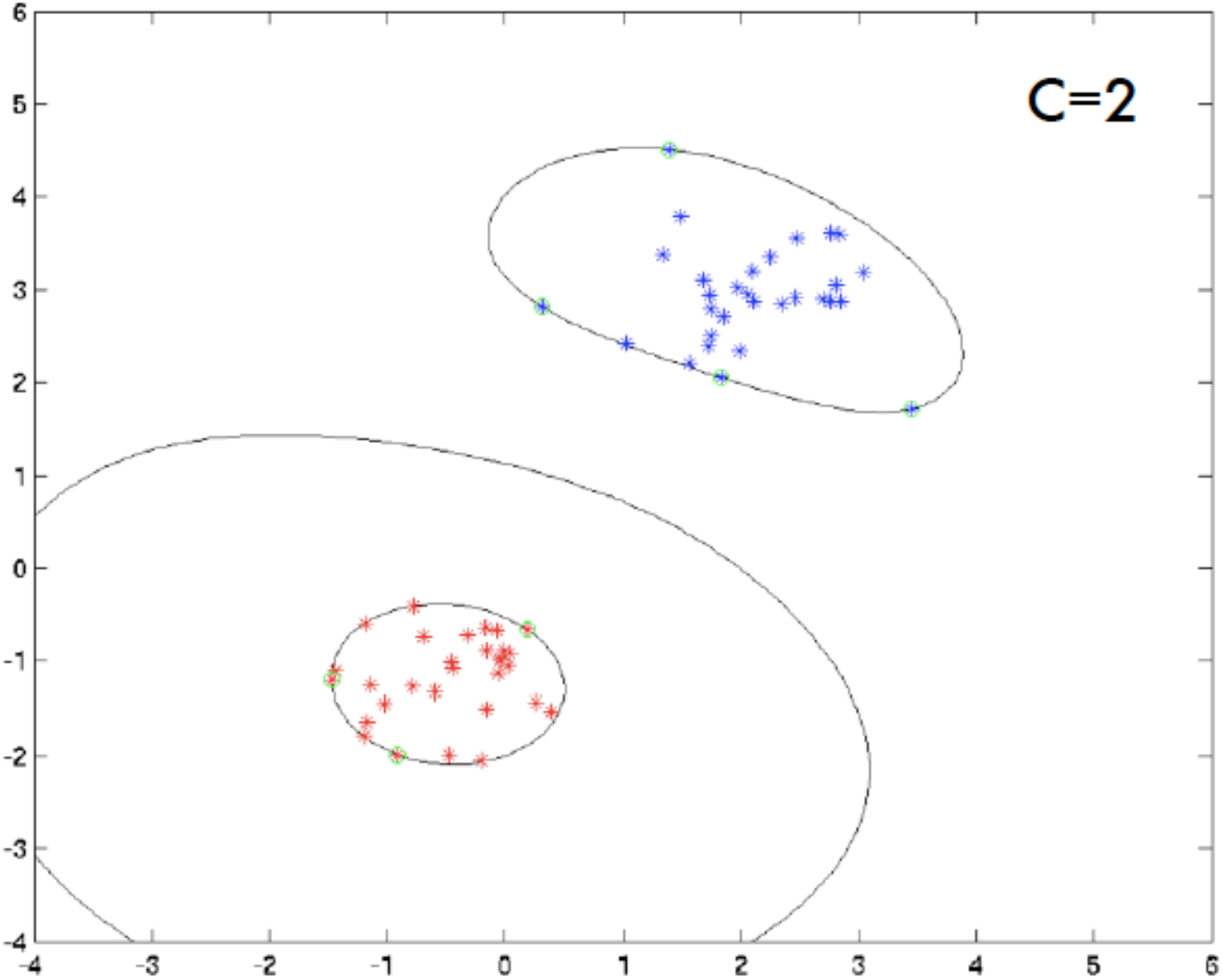
$$\underset{\alpha}{\text{maximize}} - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j k(x_i, x_j) + \sum_i \alpha_i$$

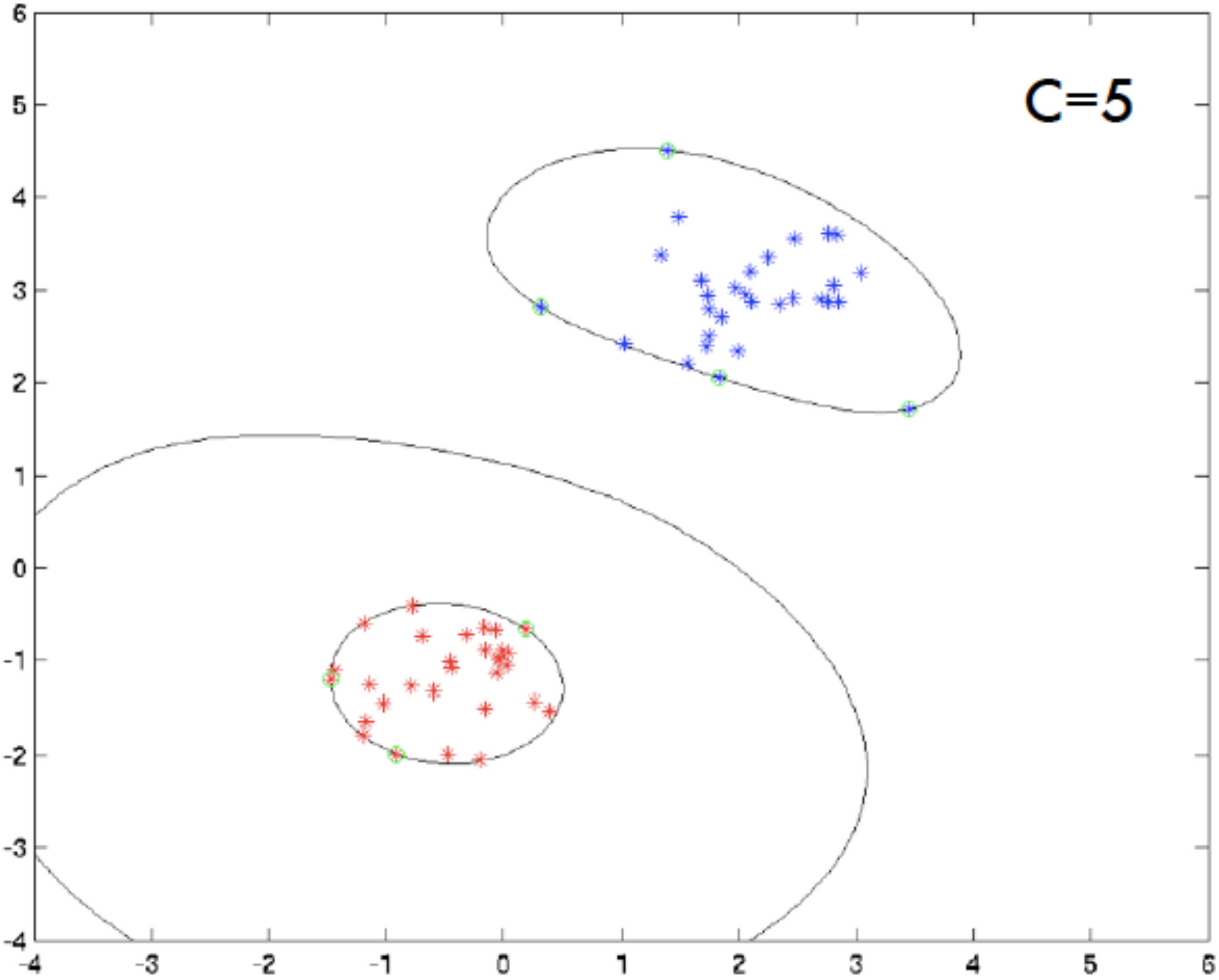
$$\text{subject to } \sum_i \alpha_i y_i = 0 \text{ and } \alpha_i \in [0, C]$$

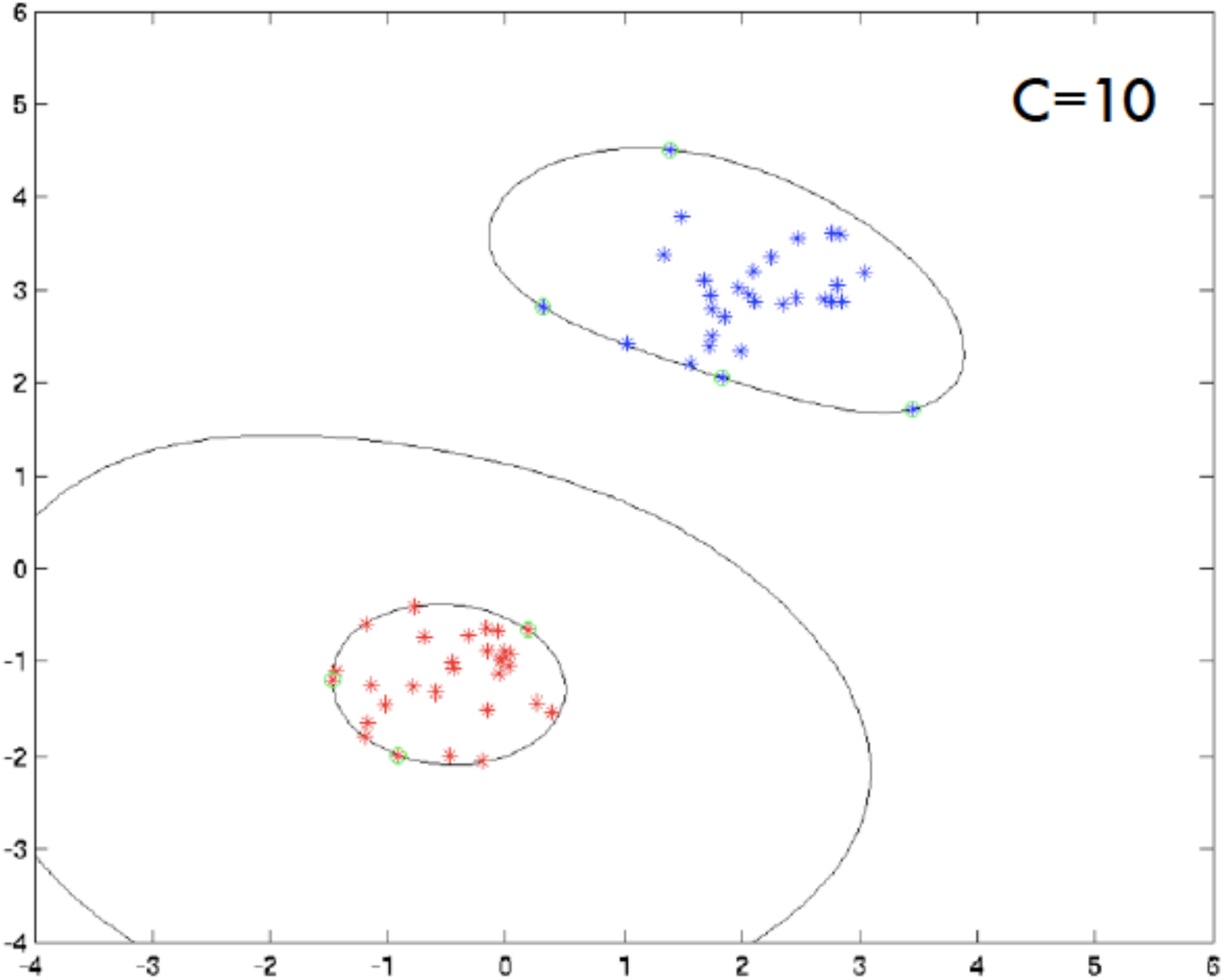
- Support vector expansion

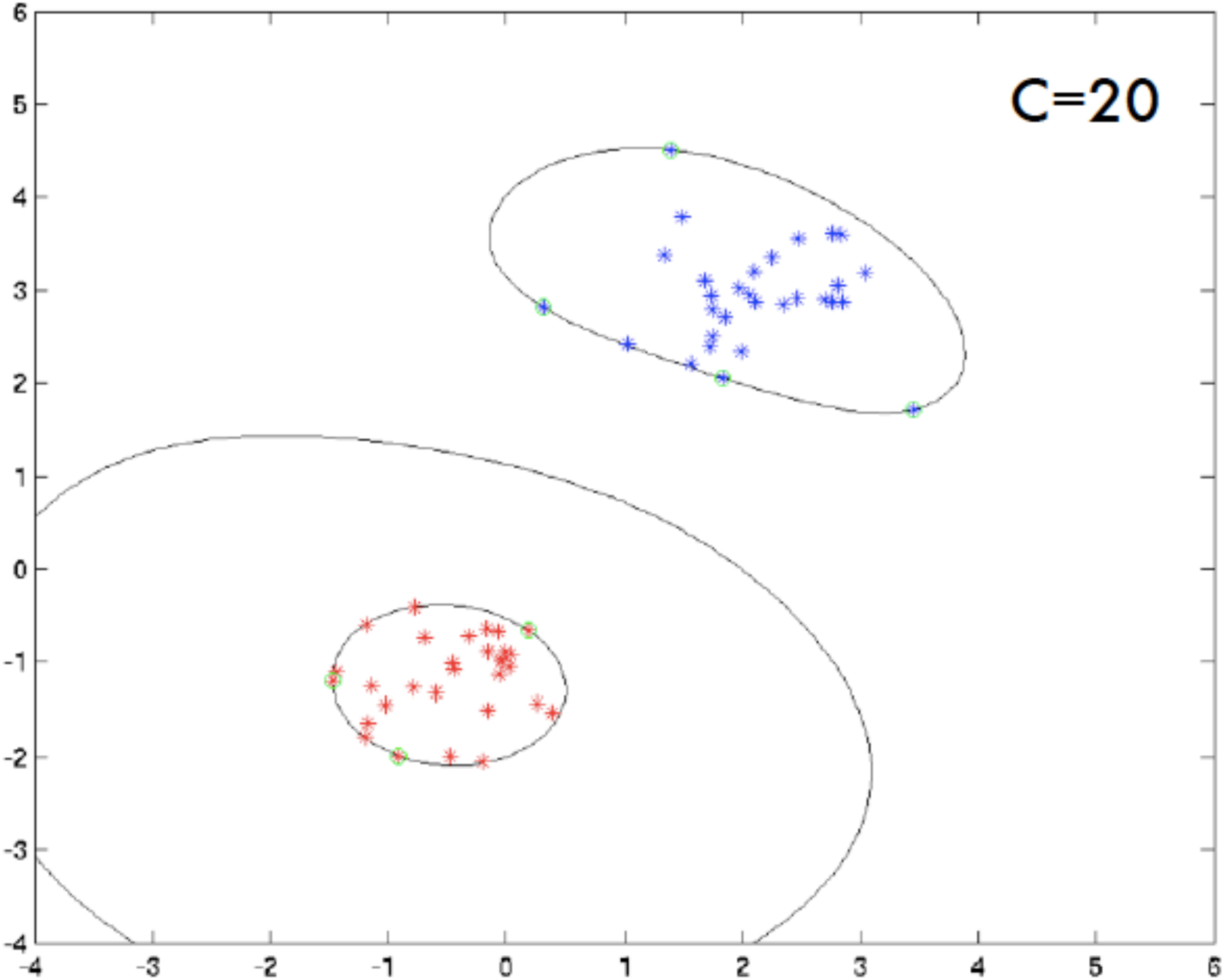
$$f(x) = \sum_i \alpha_i y_i k(x_i, x) + b$$

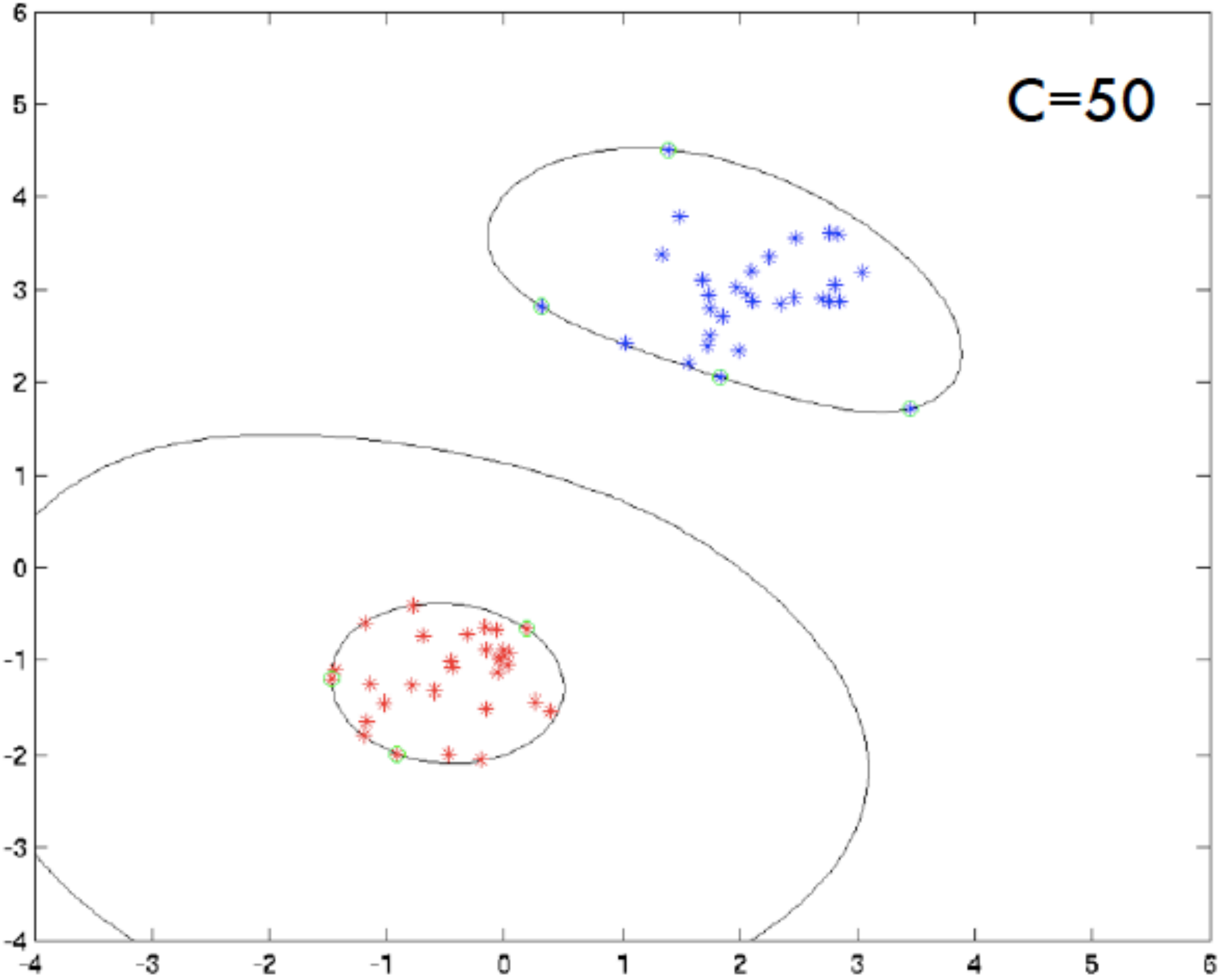

C=1
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

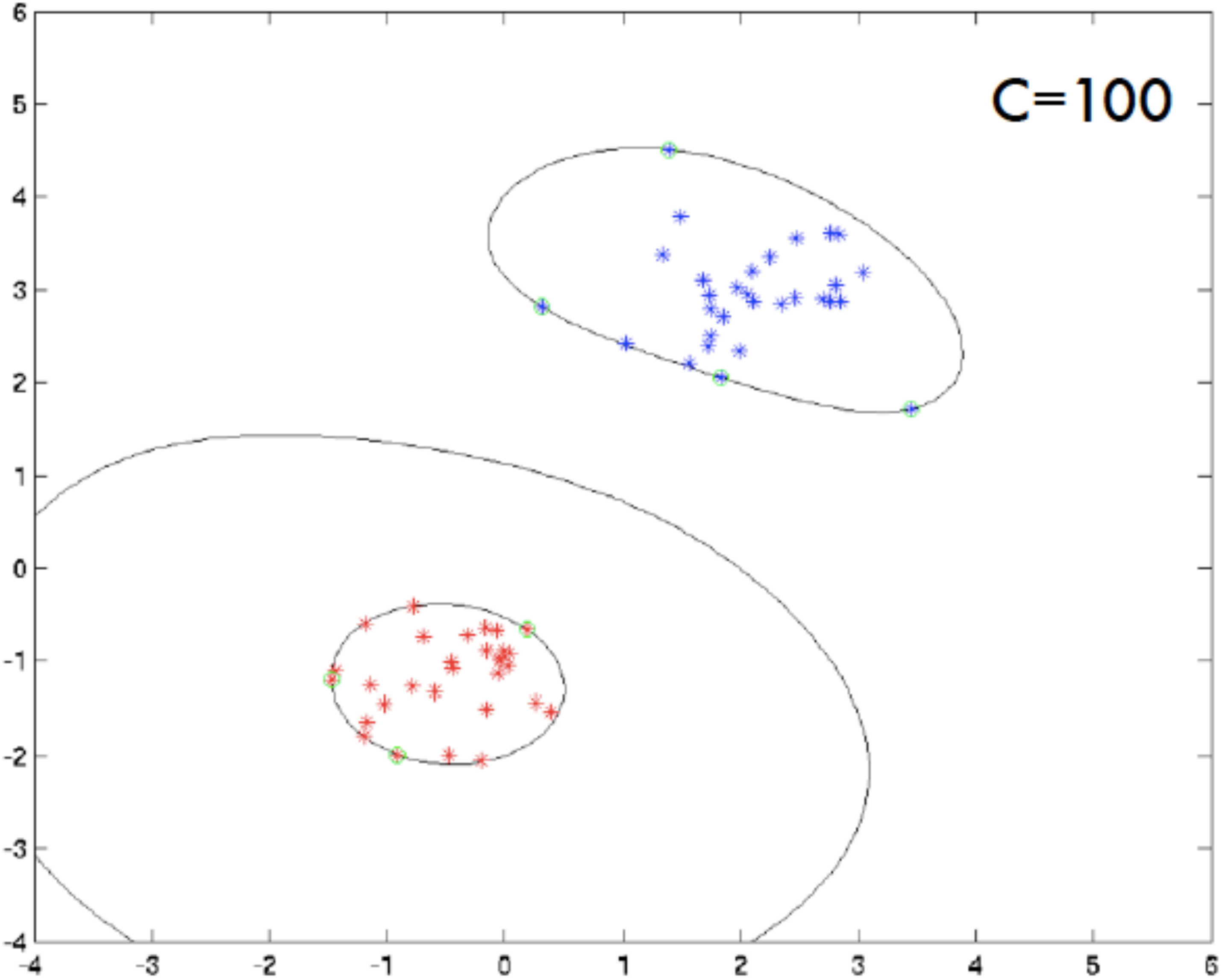

C=2
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

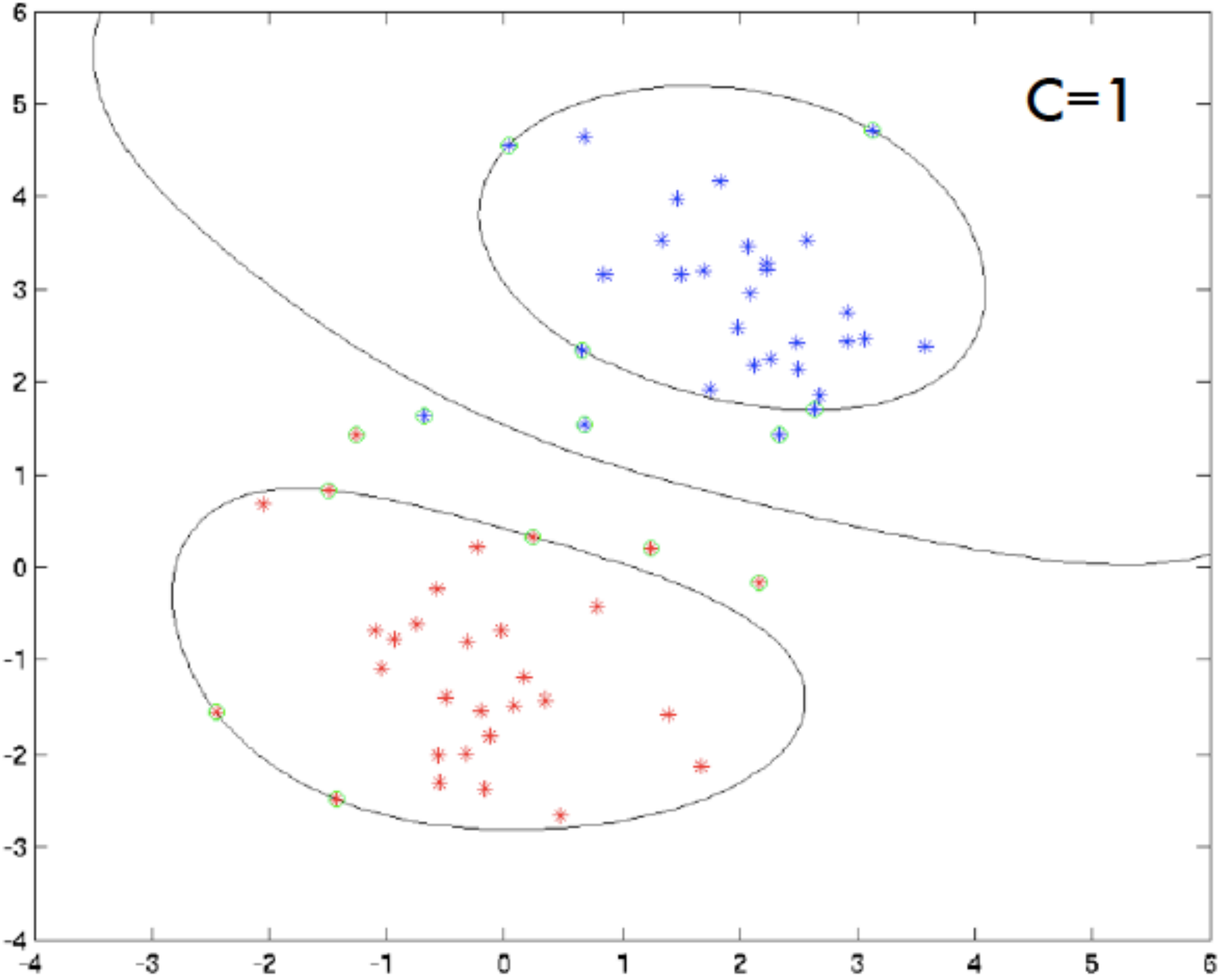

C=5
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

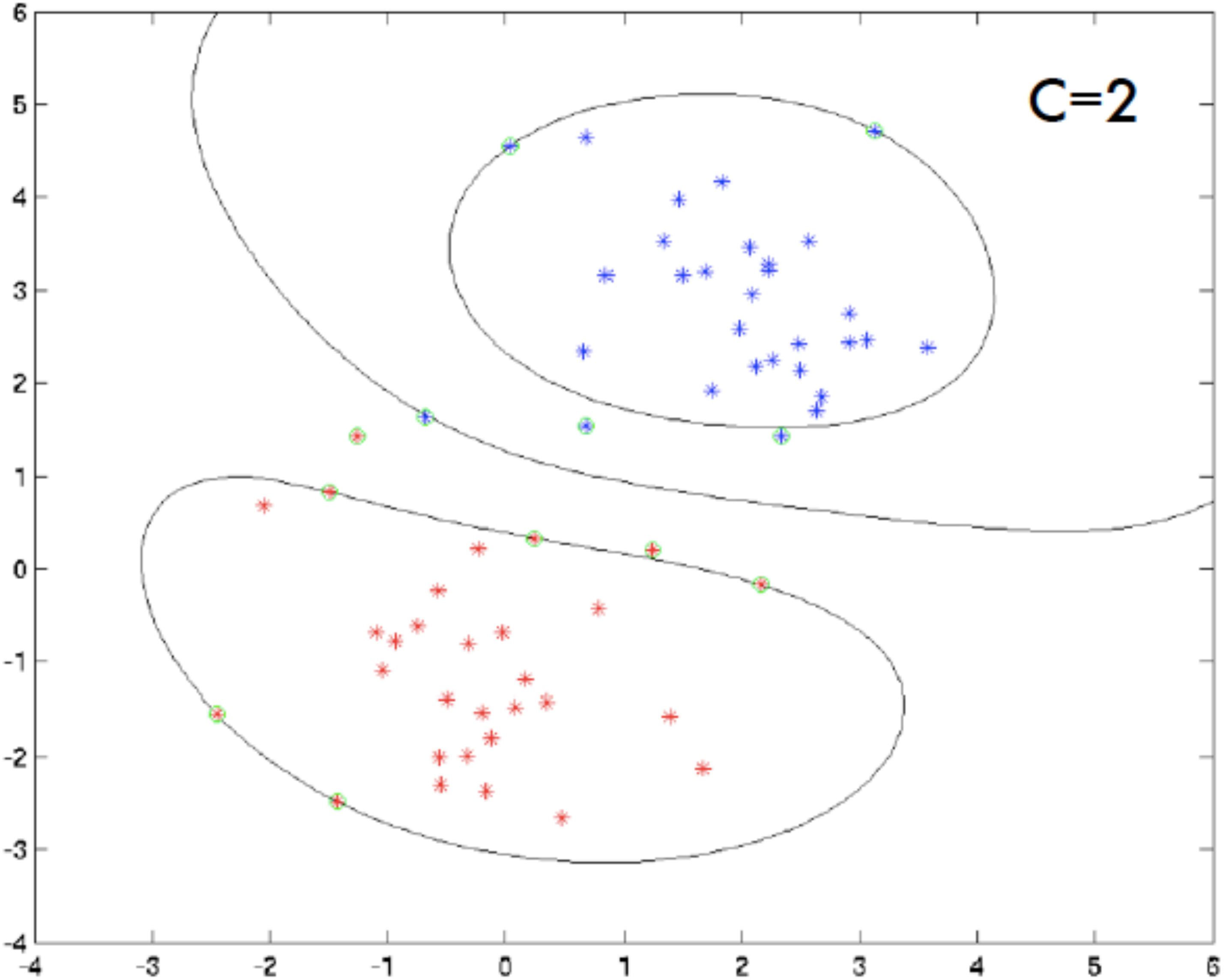

C=10
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6


C=20
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

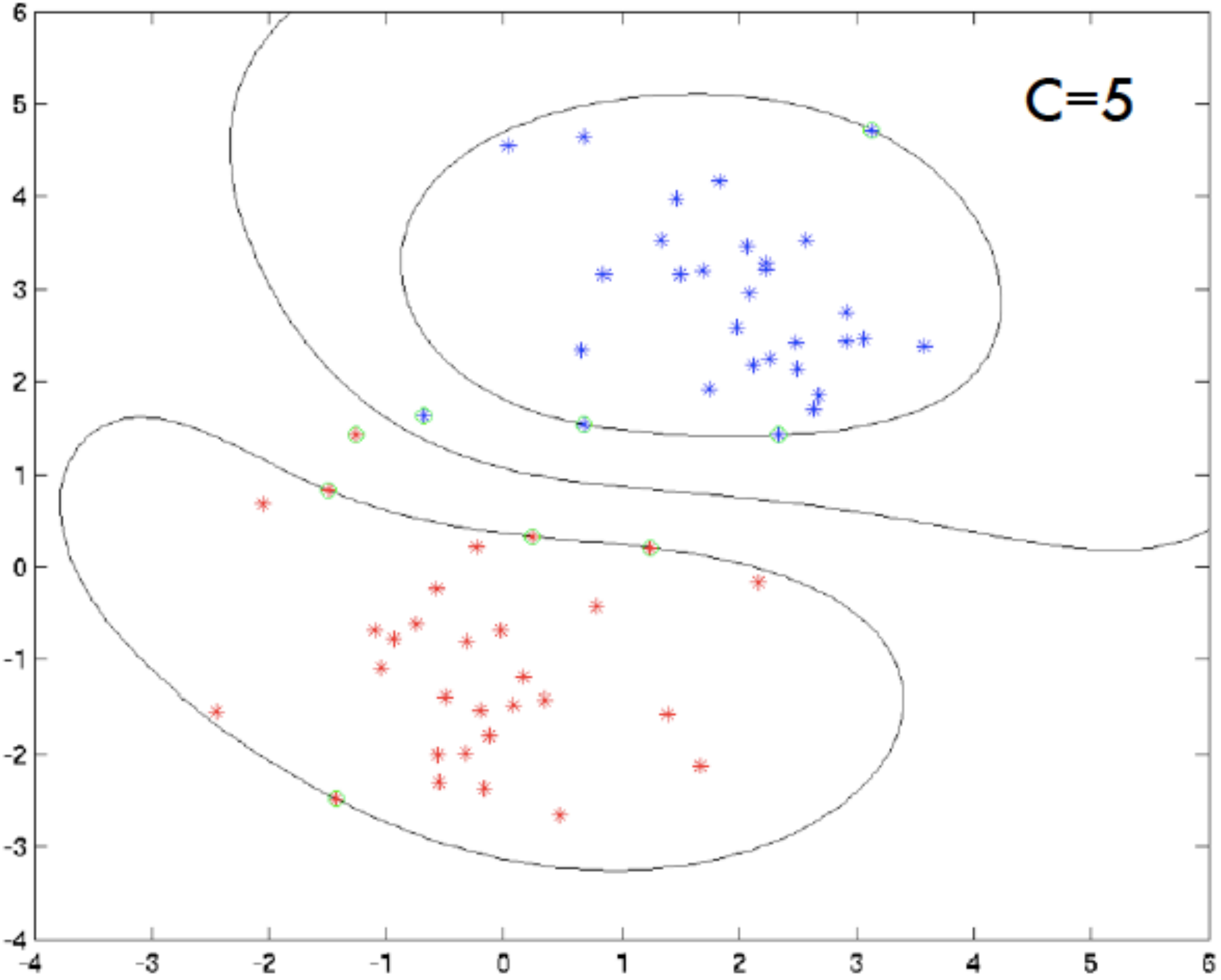

C=50
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

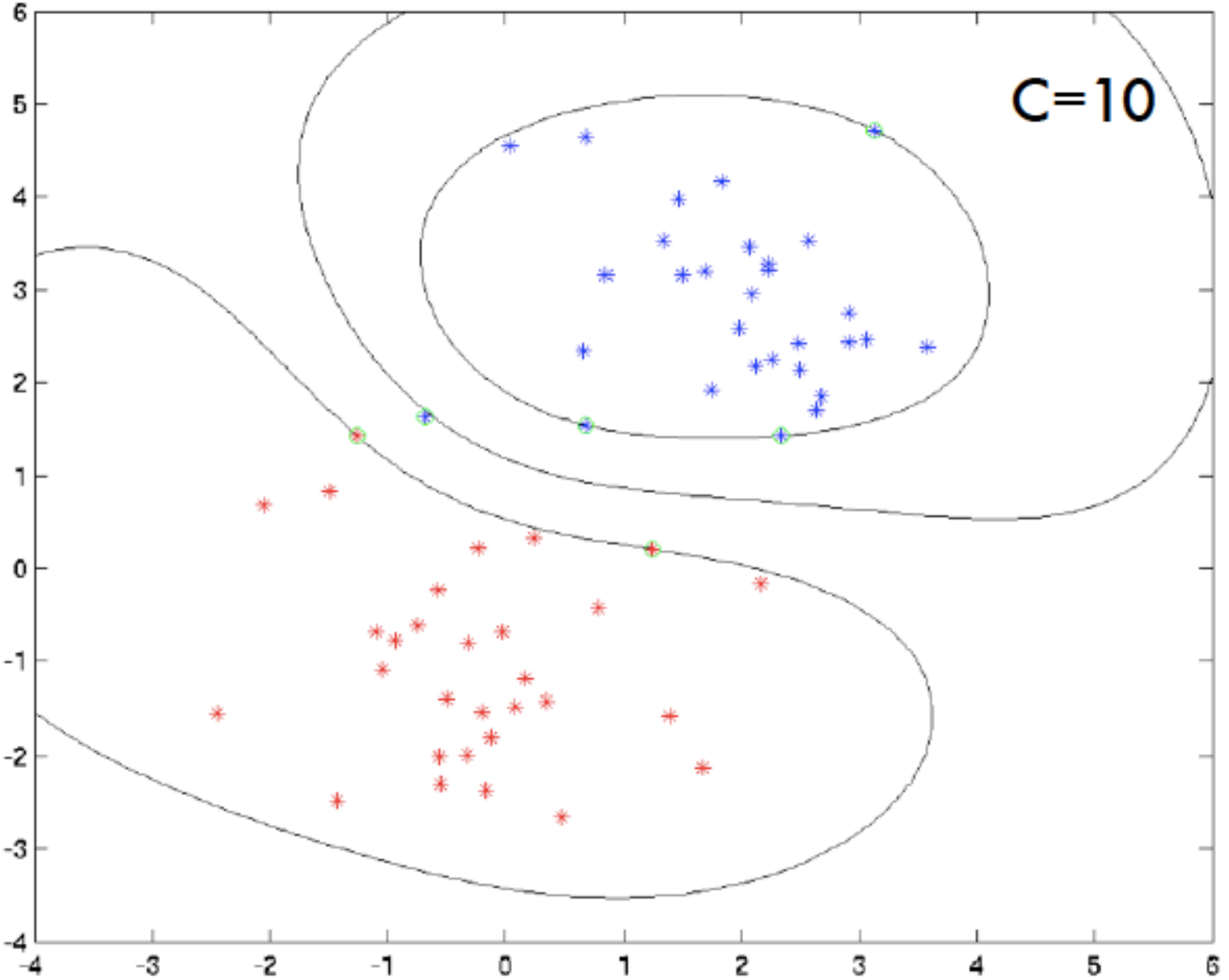

C=100
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

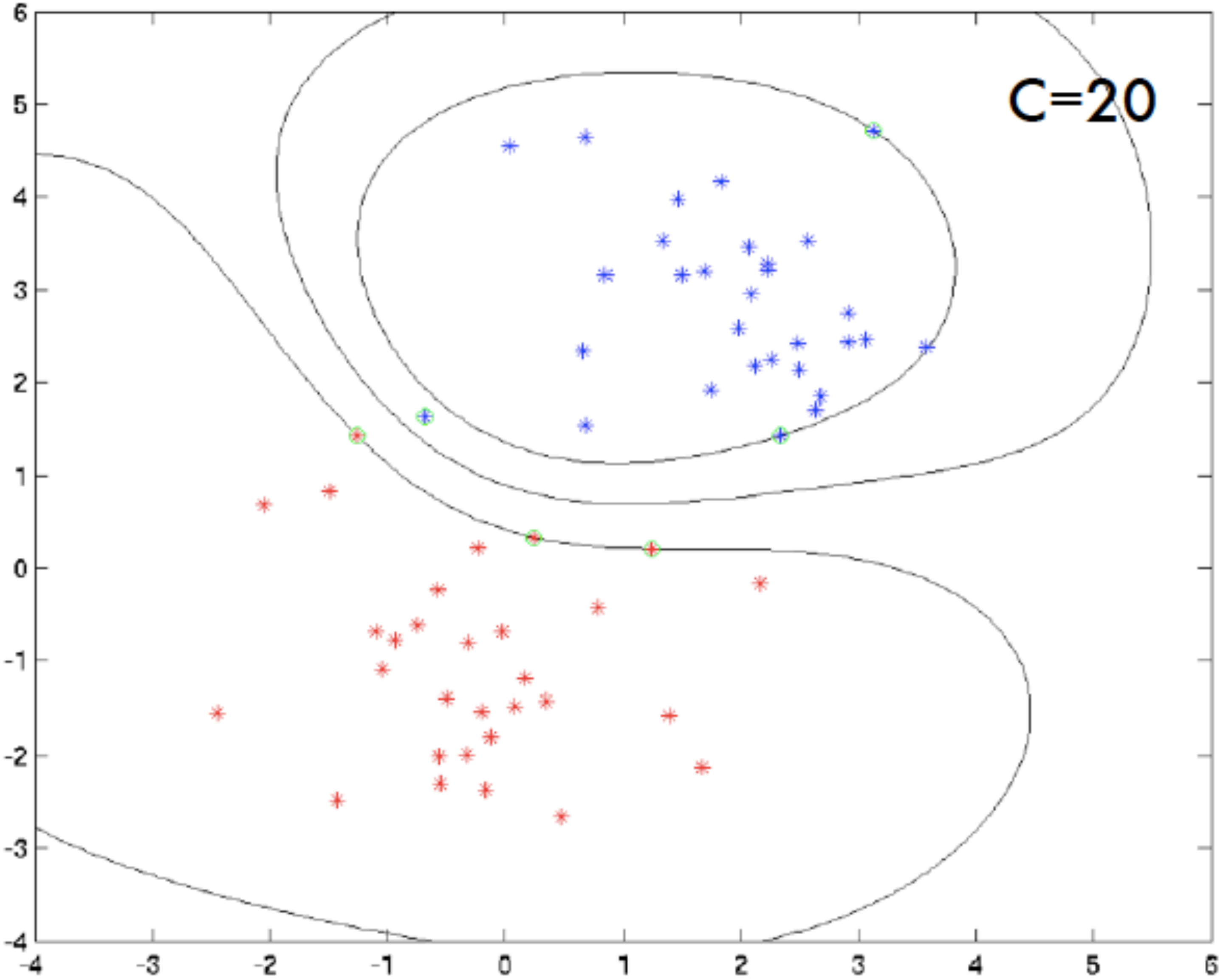

C=1
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

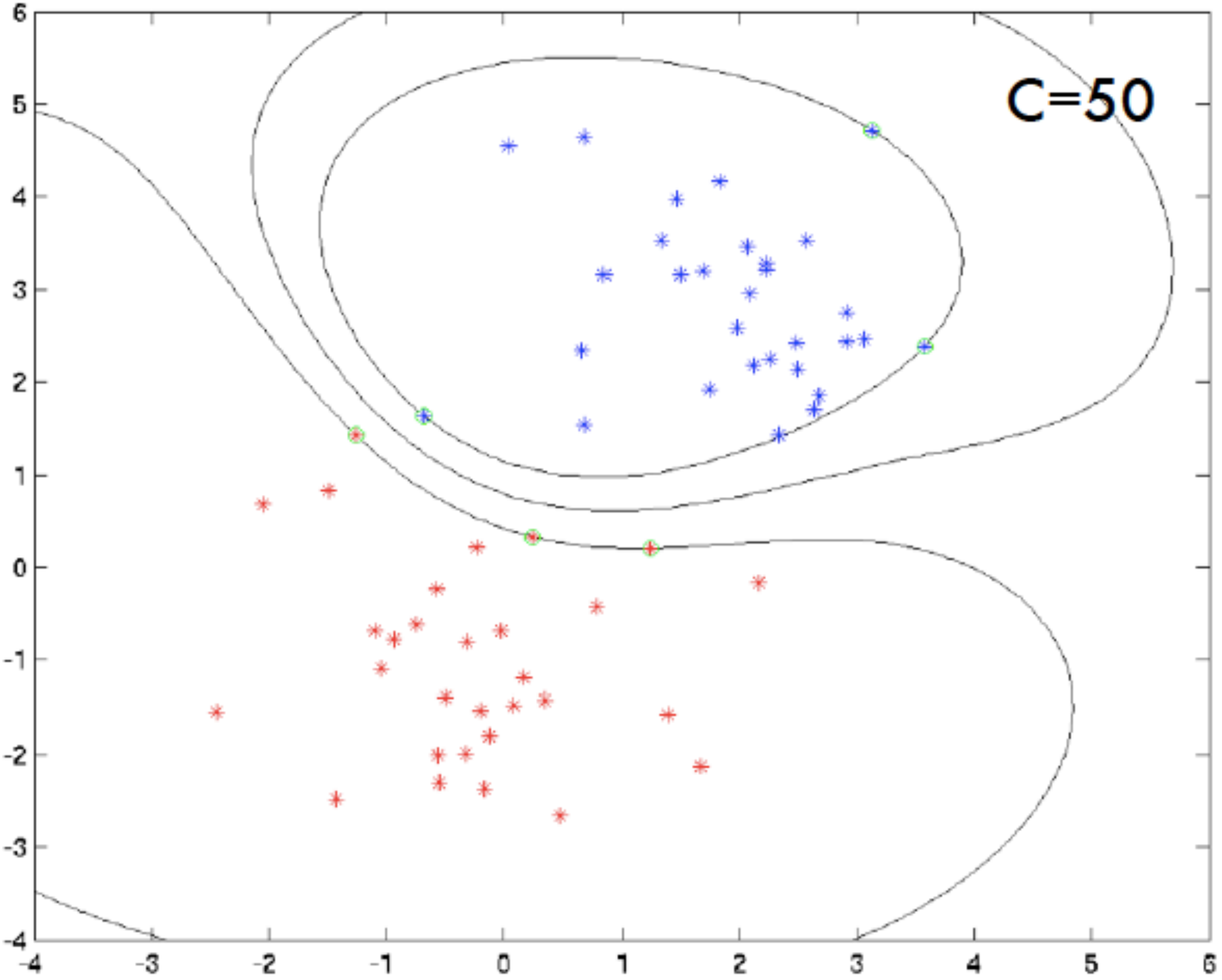


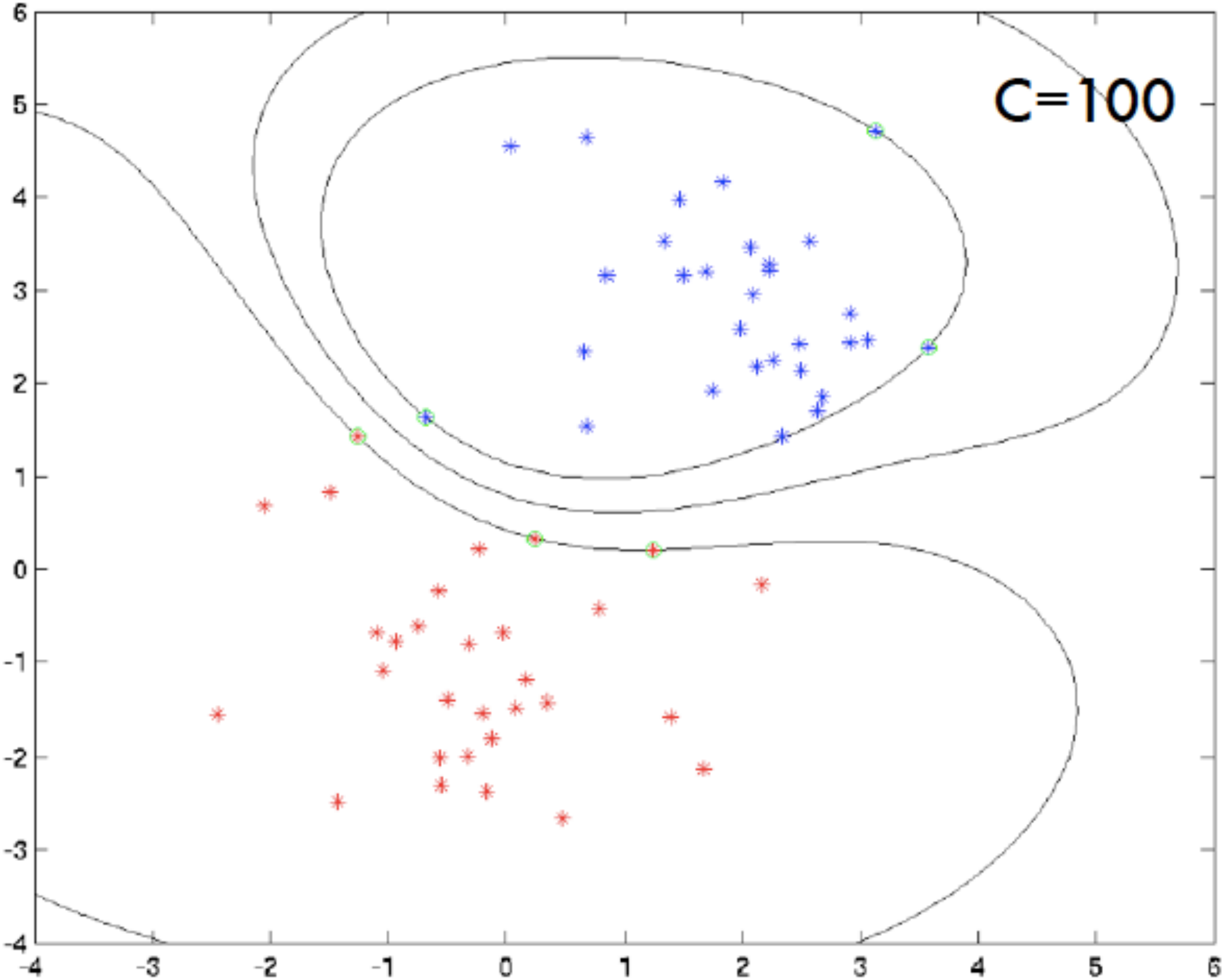
C=2
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

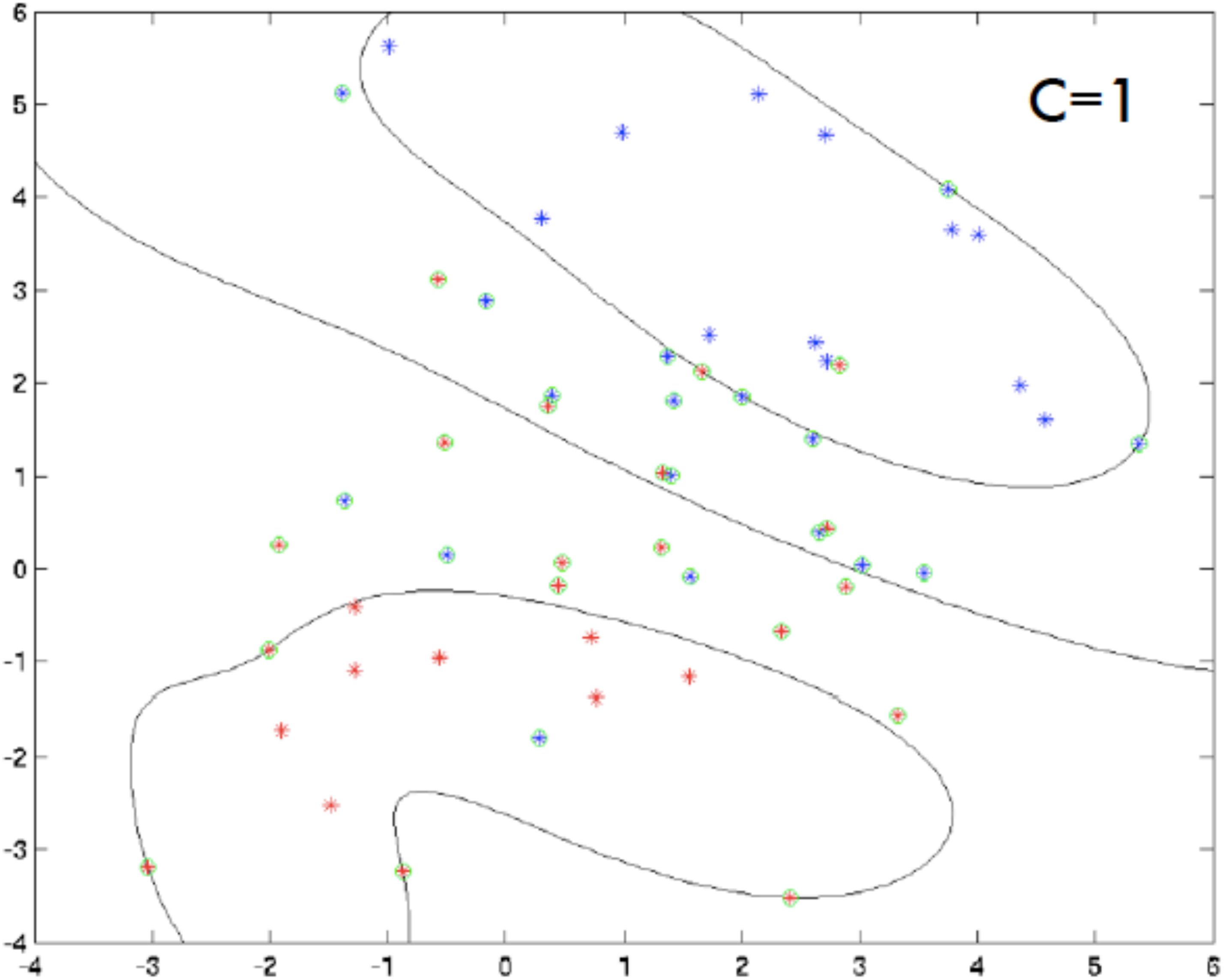

C=5
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

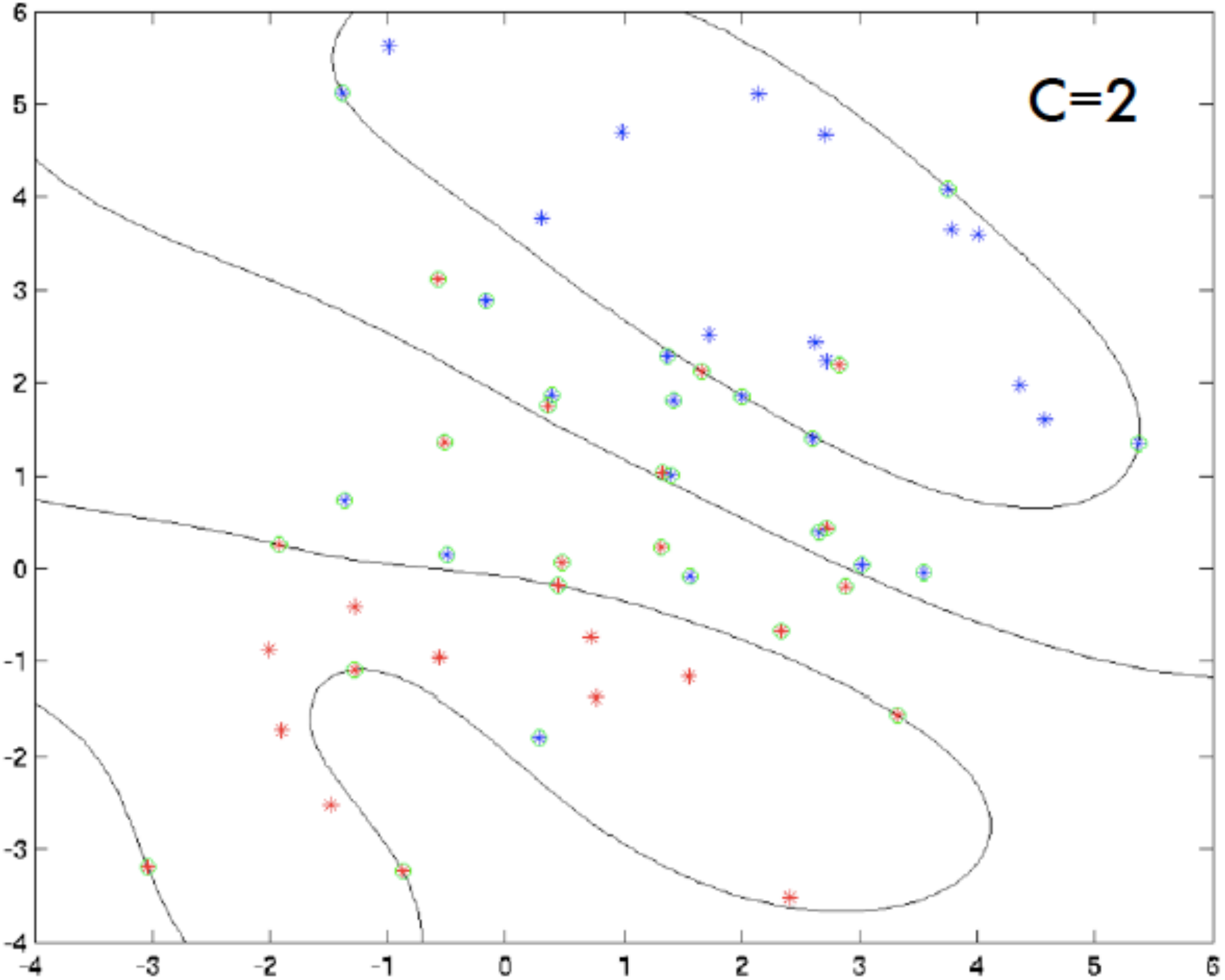

C=10
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

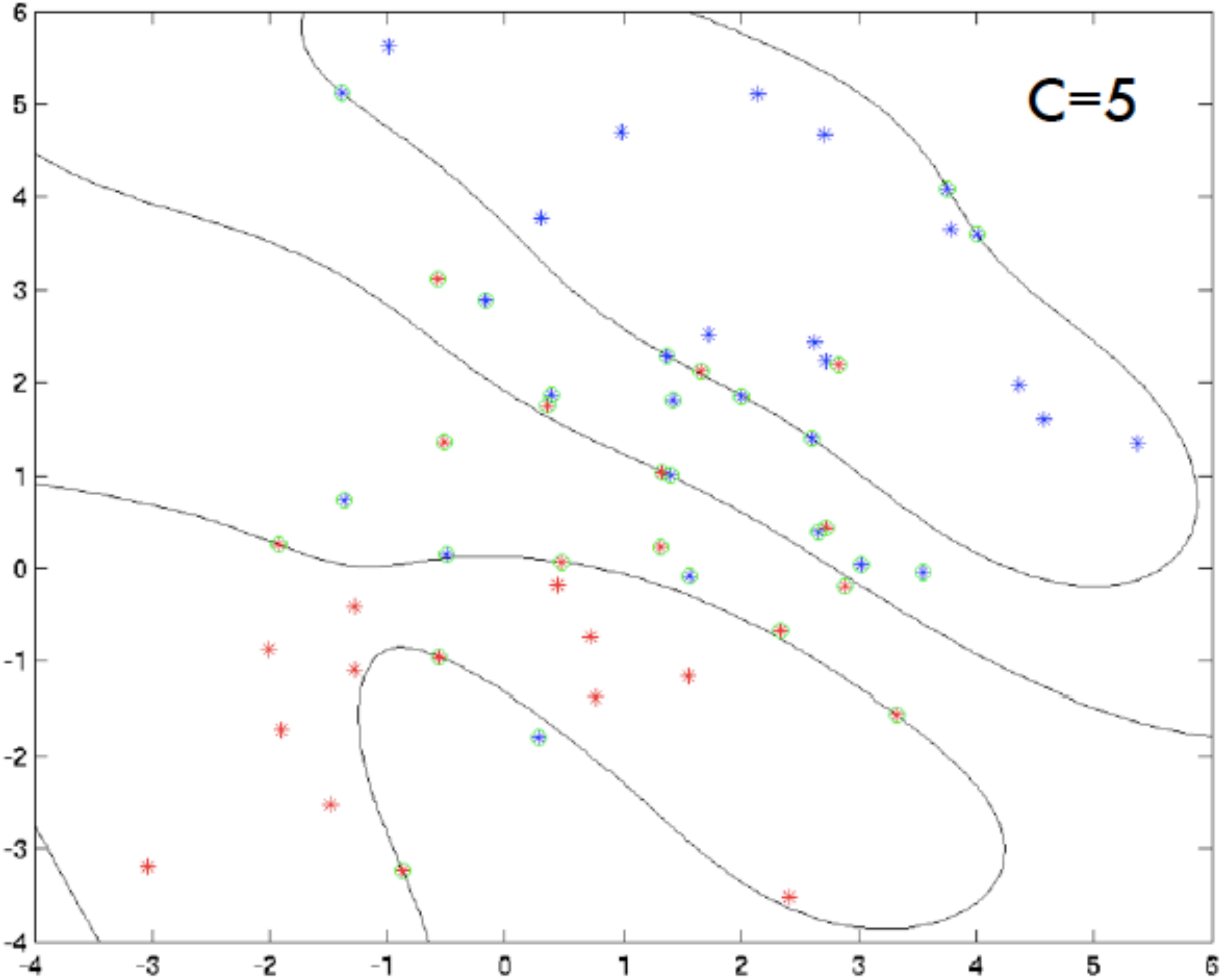

C=20
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

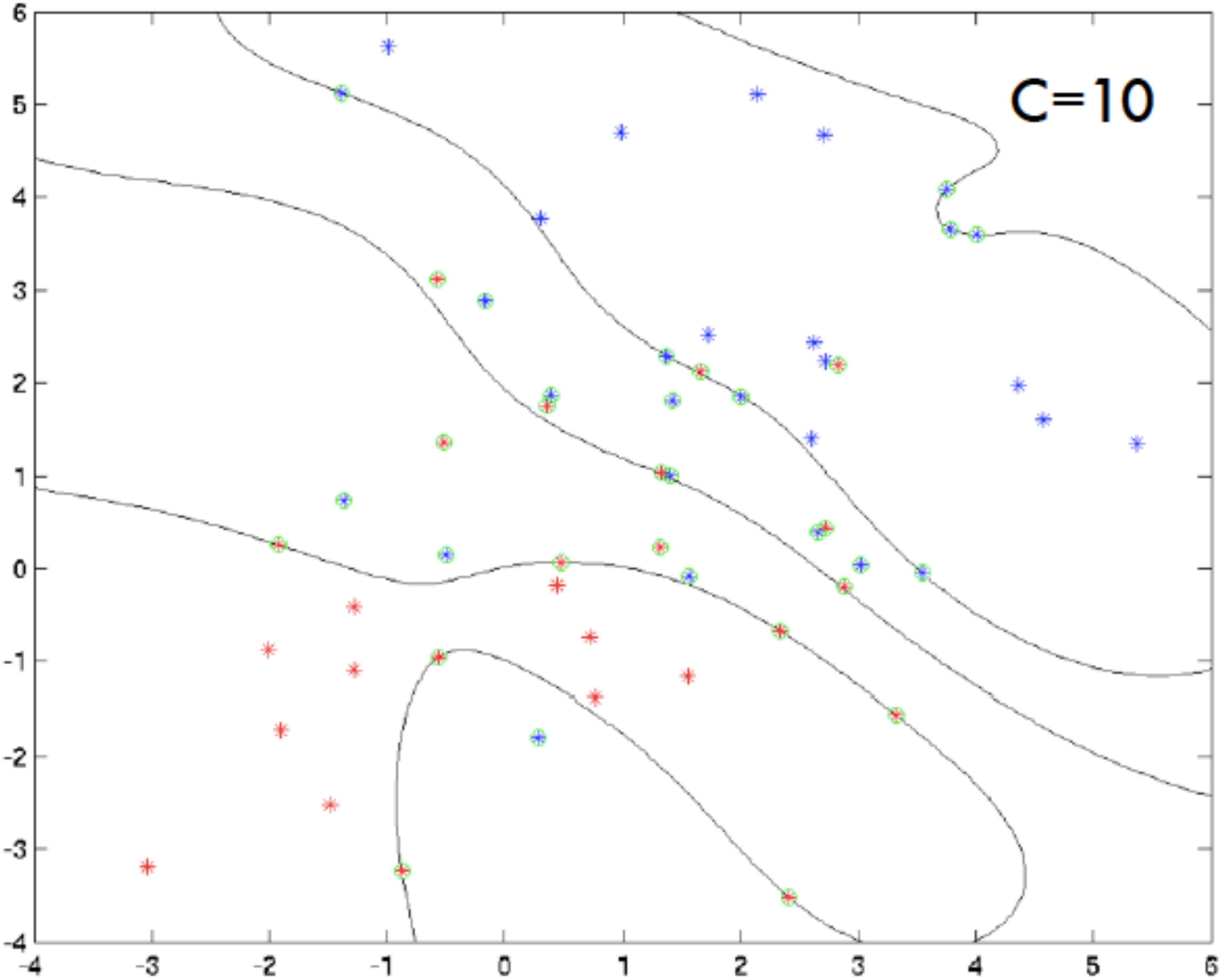

C=50
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

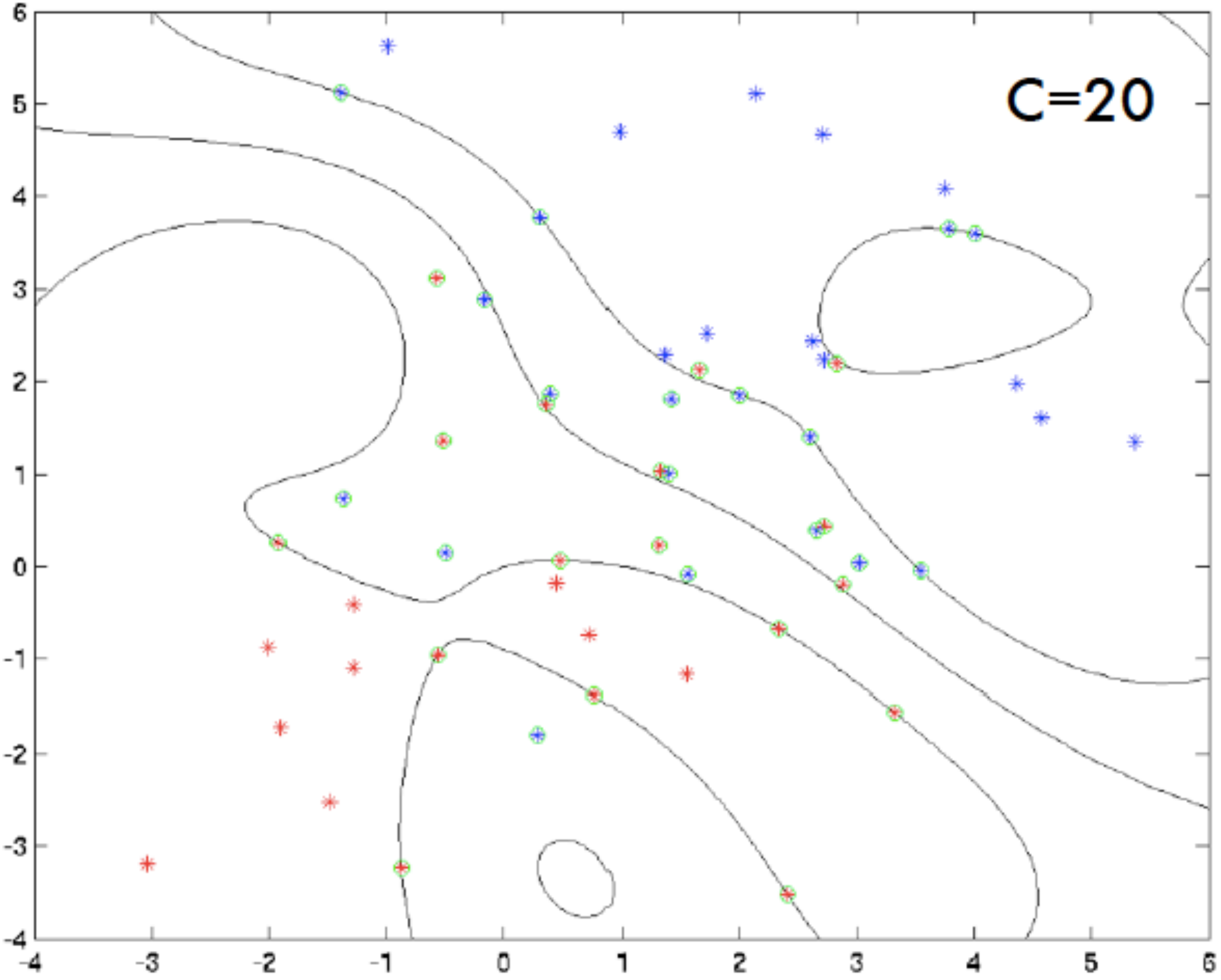

C=100
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

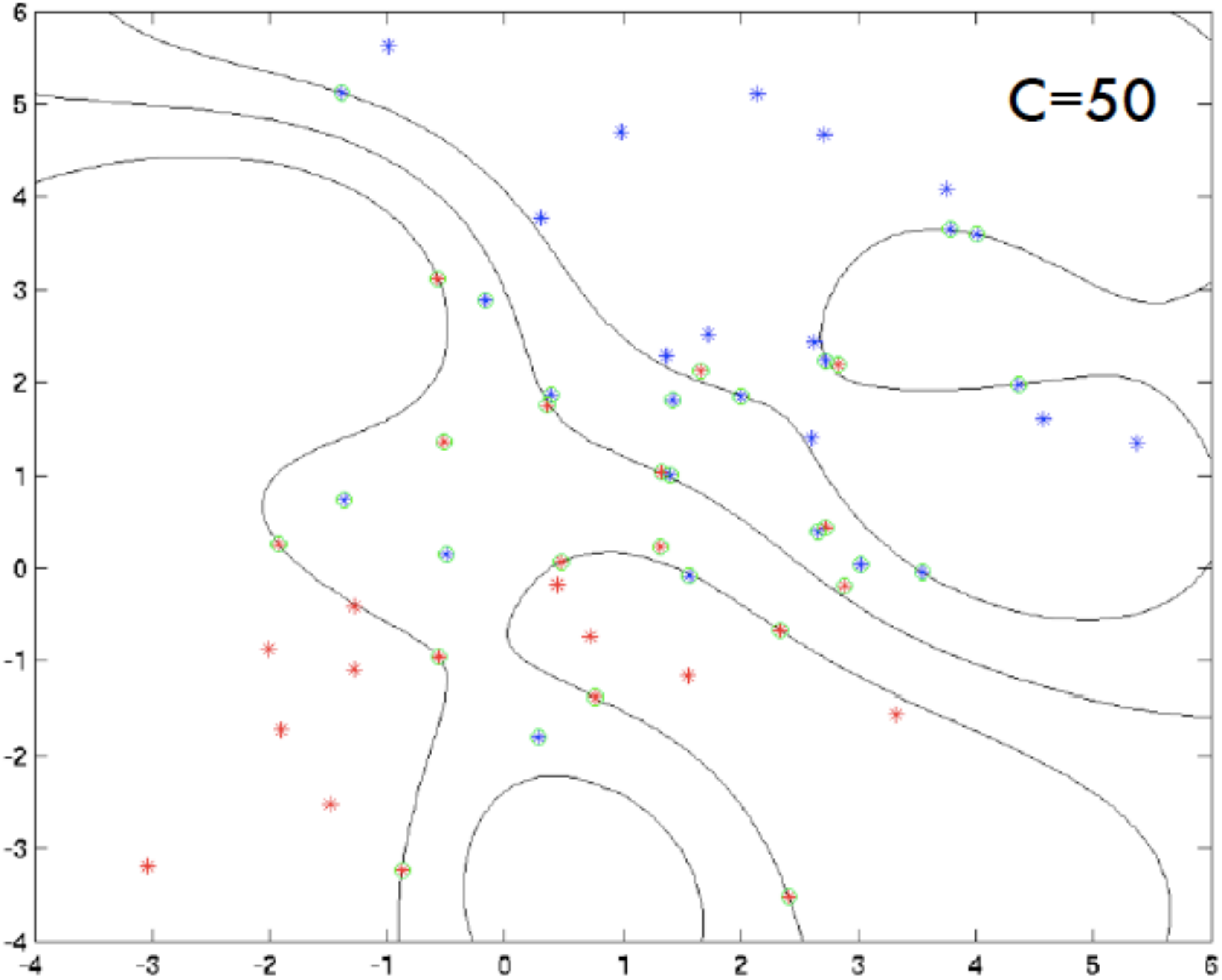

C=1
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

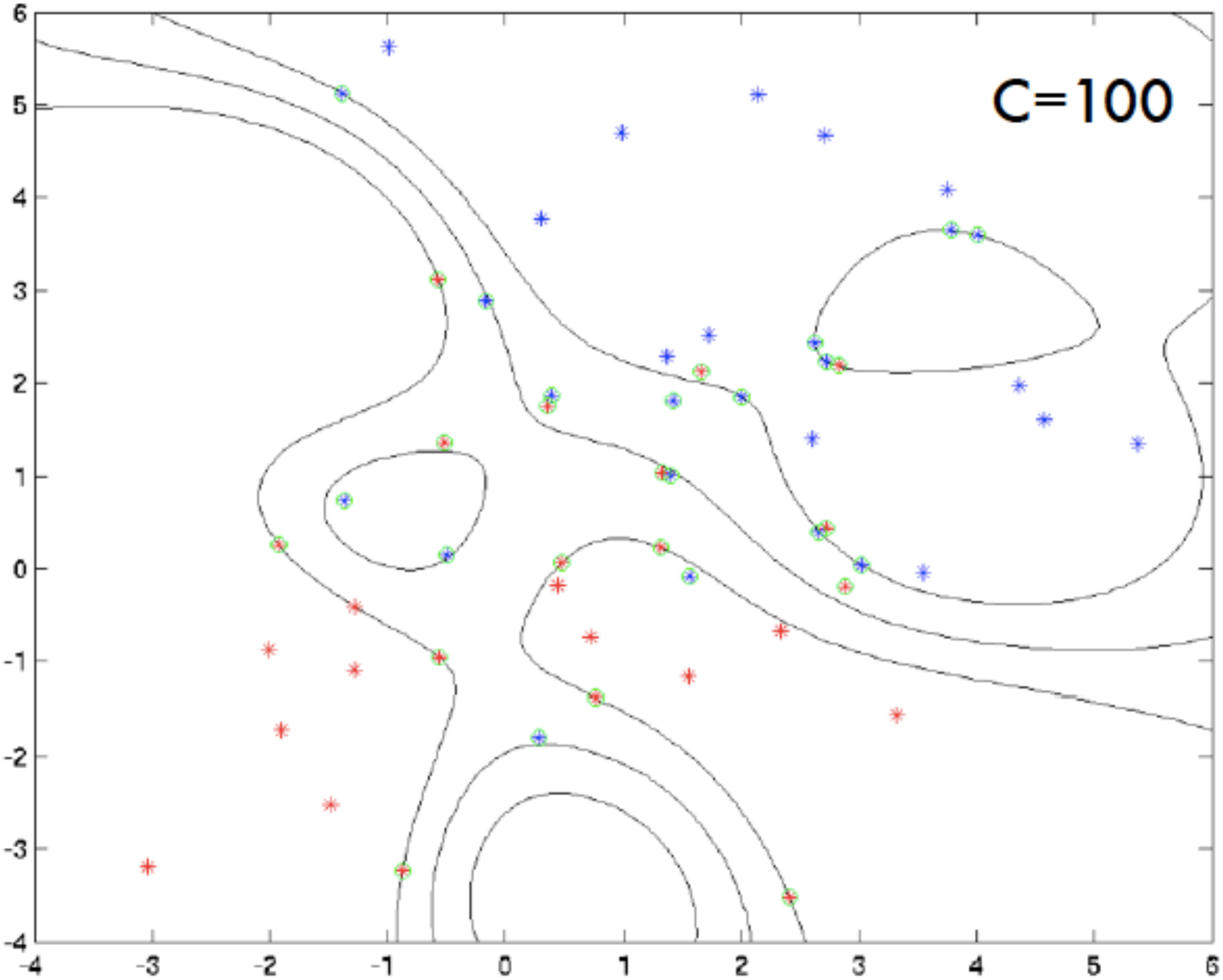

C=2
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6


C=5
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6


C=10
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

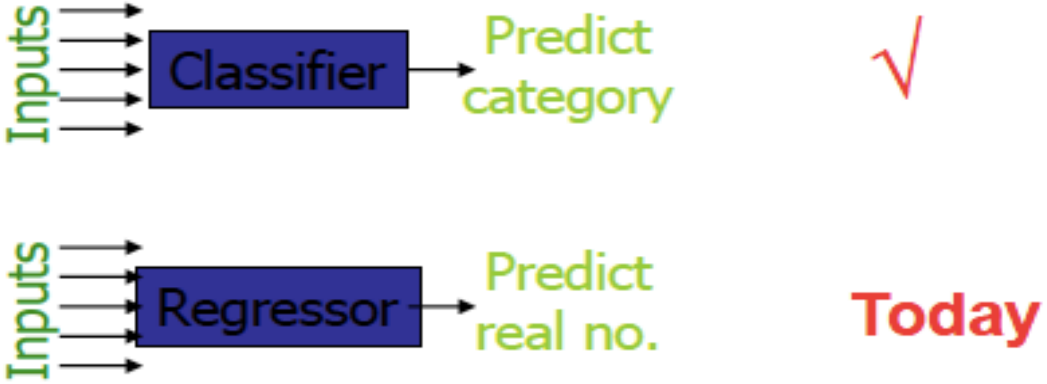

C=20
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6


C=50
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6


C=100
6
5
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6

Regression

Where we are



Inputs

Classifier

Predict category

√

Inputs

Regressor

Predict real no.

Today

Choosing a restaurant

- In everyday life we need to make decisions by taking into account lots of factors
- The question is what weight we put on each of these factors (how important are they with respect to the others).

Choosing a restaurant

- In everyday life we need to make decisions by taking into account lots of factors
- The question is what weight we put on each of these factors (how important are they with respect to the others).
- Assume we would like to build a recommender system for ***ranking*** potential restaurants based on an individuals' preferences

Reviews (out of 5 stars)	$	Distance	Cuisine (out of 10)
4	30	21	7
2	15	12	8
5	27	53	9
3	20	5	6

Choosing a restaurant

- In everyday life we need to make decisions by taking into account lots of factors
- The question is what weight we put on each of these factors (how important are they with respect to the others).
- Assume we would like to build a recommender system for ***ranking*** potential restaurants based on an individuals' preferences
- If we have many observations we may be able to recover the weights

Reviews (out of 5 stars)	$	Distance	Cuisine (out of 10)
4	30	21	7
2	15	12	8
5	27	53	9
3	20	5	6

?

Linear regression

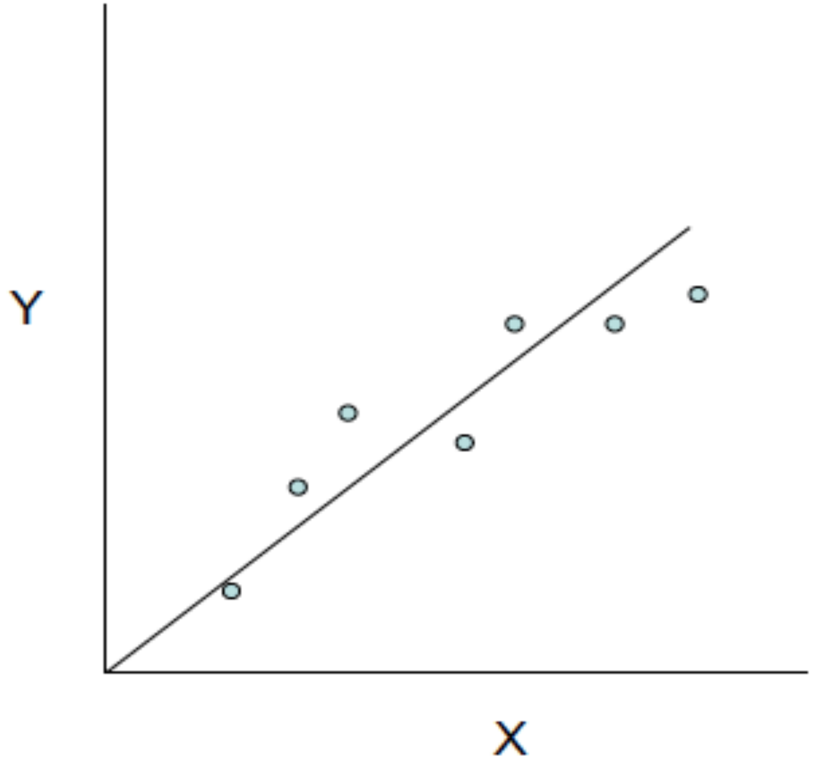
Linear regression

- Given an input x we would like to compute an output y
- For example:

  - Predict height from age

  - Predict Google's price from Yahoo's price

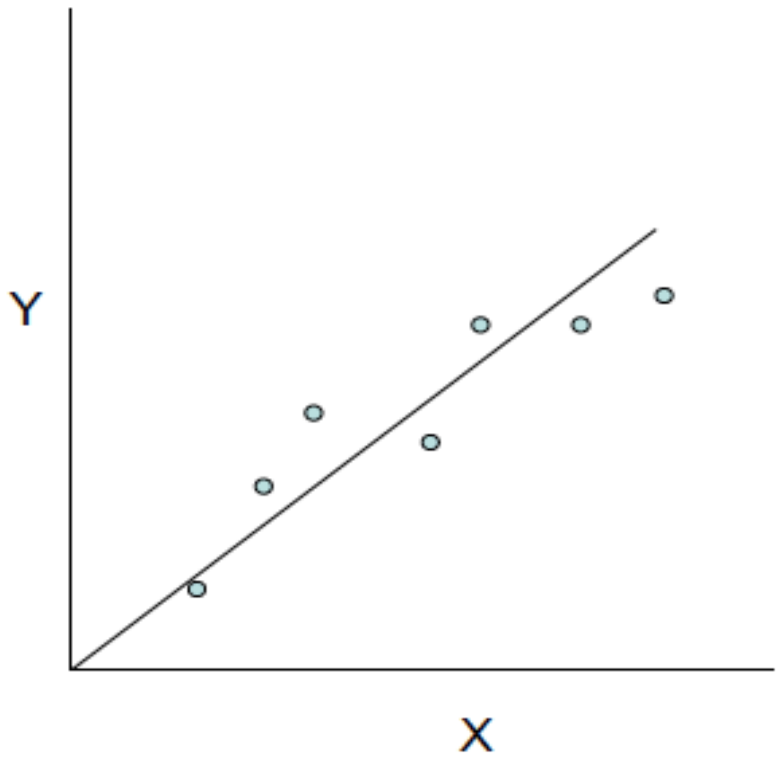
  - Predict distance from wall using sensor readings



Note that now Y can be **continuous**

Linear regression

- Given an input x we would like to compute an output y
- In linear regression we assume that y and x are related with the following equation:



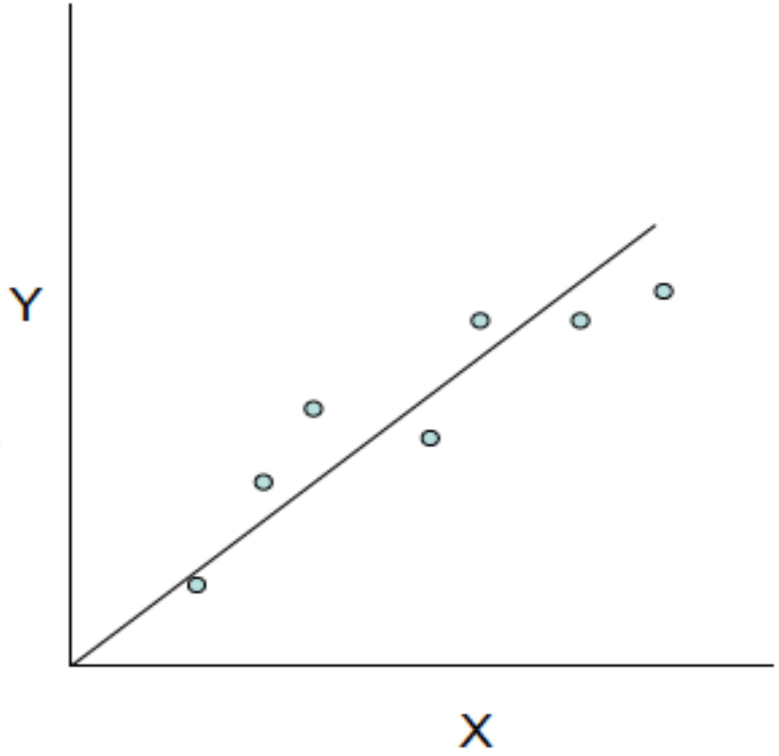
Linear regression

- Given an input x we would like to compute an output y
- In linear regression we assume that y and x are related with the following equation:

What we are trying to predict

Observed values

$$y = wx+\varepsilon$$



Linear regression

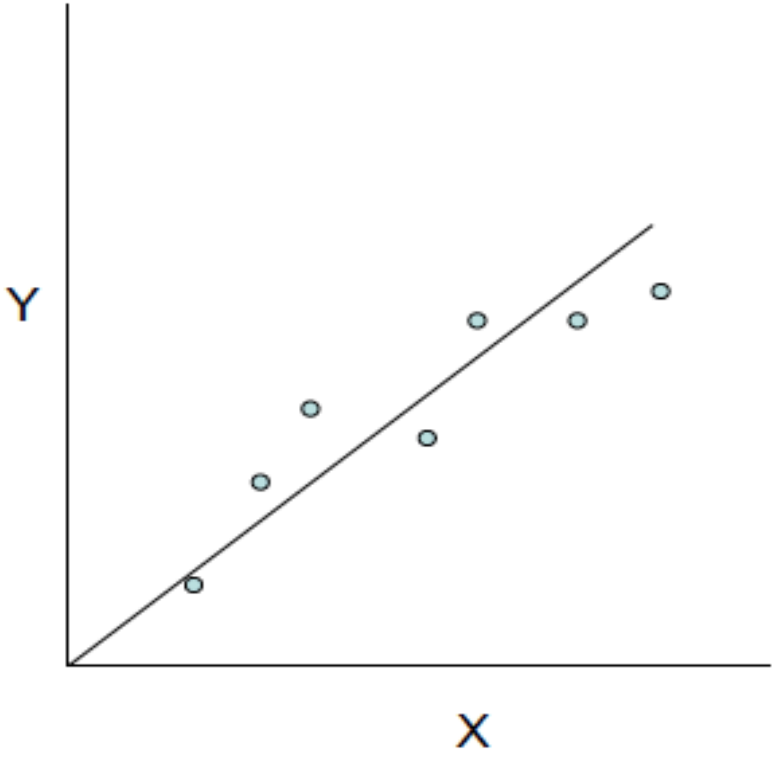
- Given an input x we would like to compute an output y
- In linear regression we assume that y and x are related with the following equation:

What we are trying to predict

Observed values

$$y = wx+\varepsilon$$

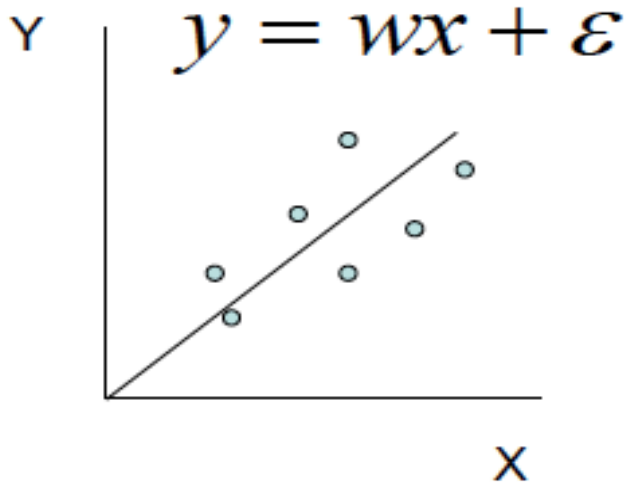
where w is a parameter and ε represents measurement or other noise



Linear regression

Linear regression

- Our goal is to estimate *w* from a training data of $<x_i,y_i>$ pairs

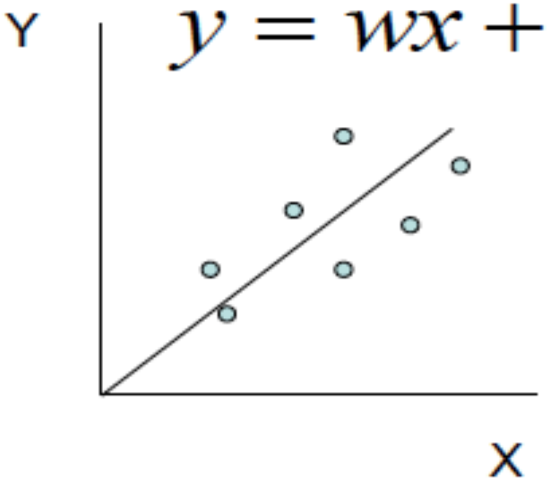


Linear regression

- Our goal is to estimate *w* from a training data of <x_i,y_i> pairs
- One way to find such relationship is to minimize the a least squares error:

$$\arg\min_w \sum_i (y_i - wx_i)^2$$

$$y = wx + \varepsilon$$



- Several other approaches can be used as well
- So why least squares?

  - minimizes squared distance between measurements and predicted line
  - has a nice probabilistic interpretation
  - easy to compute

If the noise is Gaussian with mean 0 then least squares is also the maximum likelihood estimate of w

5 minutes break

Solving linear regression using least squares minimization

- We just take the derivative w.r.t. to w and set to 0:

Solving linear regression using least squares minimization

- We just take the derivative w.r.t. to w and set to 0:

$$\frac{\partial}{\partial w}\sum_i (y_i - wx_i)^2 = 2\sum_i -x_i(y_i - wx_i) \Rightarrow$$

Solving linear regression using least squares minimization

- We just take the derivative w.r.t. to w and set to 0:

$$\frac{\partial}{\partial w}\sum_i (y_i - wx_i)^2 = 2\sum_i -x_i(y_i - wx_i) \Rightarrow$$

$$2\sum_i x_i(y_i - wx_i) = 0 \Rightarrow$$

Solving linear regression using least squares minimization

- We just take the derivative w.r.t. to w and set to 0:

$$\frac{\partial}{\partial w}\sum_i (y_i - wx_i)^2 = 2\sum_i -x_i(y_i - wx_i) \Rightarrow$$

$$2\sum_i x_i(y_i - wx_i) = 0 \Rightarrow$$

$$\sum_i x_i y_i = \sum_i w x_i^2 \Rightarrow$$

Solving linear regression using least squares minimization

- We just take the derivative w.r.t. to w and set to 0:

$$\frac{\partial}{\partial w}\sum_i (y_i - wx_i)^2 = 2\sum_i -x_i(y_i - wx_i) \Rightarrow$$

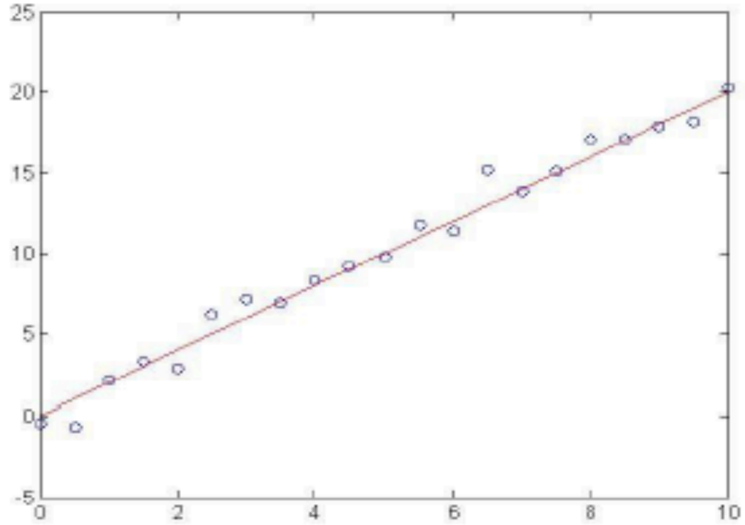
$$2\sum_i x_i(y_i - wx_i) = 0 \Rightarrow$$

$$\sum_i x_i y_i = \sum_i wx_i^2 \Rightarrow$$

$$w = \frac{\sum_i x_i y_i}{\sum_i x_i^2}$$

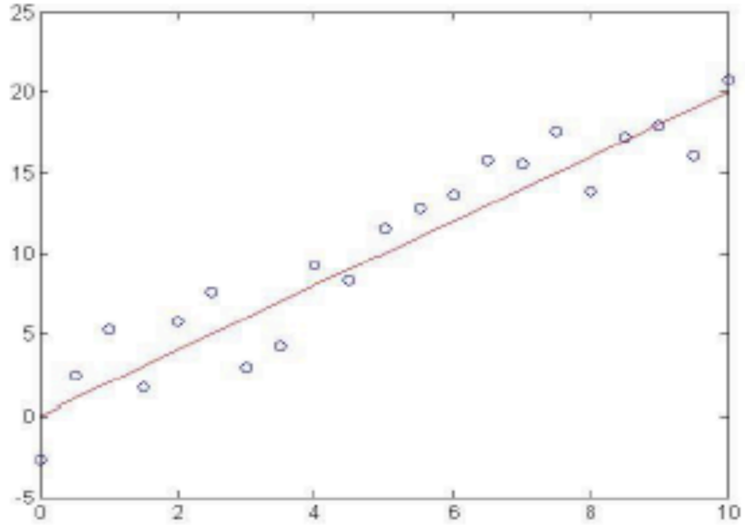
Regression example

- Generated: w=2
- Recovered: w=2.03
- Noise: std=1



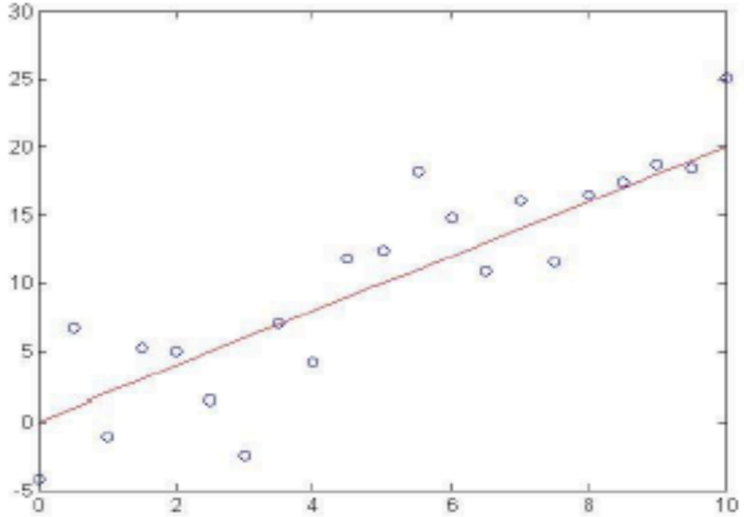
Regression example

- Generated: w=2
- Recovered: w=2.05
- Noise: std=2



Regression example

- Generated: w=2
- Recovered: w=2.08
- Noise: std=4

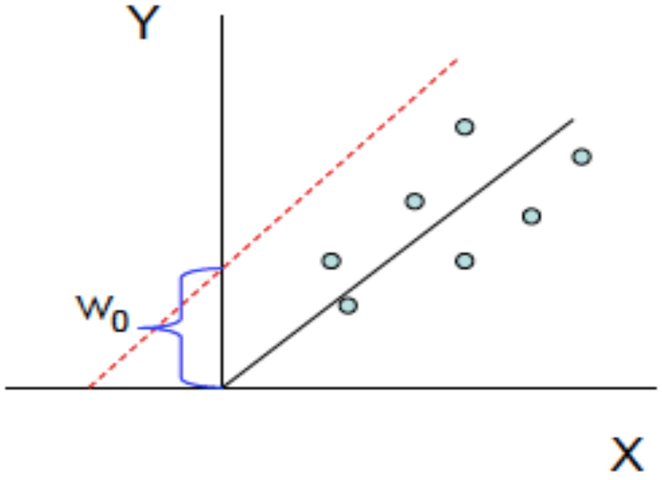


Bias term

Bias term

- So far we assumed that the line passes through the origin
- What if the line does not?
- No problem, simply change the model to

$$y = w_0 + w_1 x + \varepsilon$$



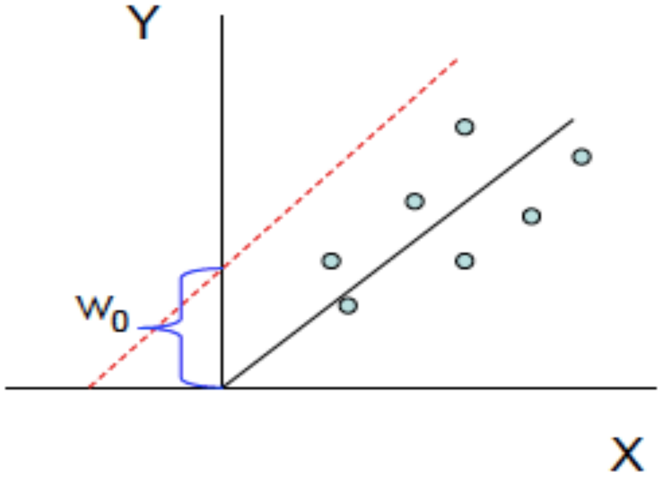
Bias term

- So far we assumed that the line passes through the origin
- What if the line does not?
- No problem, simply change the model to

$$y = w_0 + w_1x+\varepsilon$$

- Can use least squares to determine w_0 , w_1

$$w_0 = \frac{\sum_i y_i - w_1 x_i}{n}$$

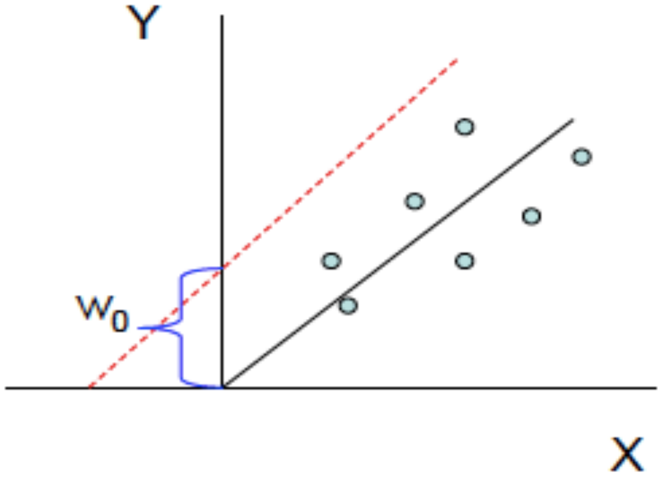


Bias term

- So far we assumed that the line passes through the origin
- What if the line does not?
- No problem, simply change the model to

$$y = w_0 + w_1 x + \varepsilon$$

- Can use least squares to determine w_0 , w_1



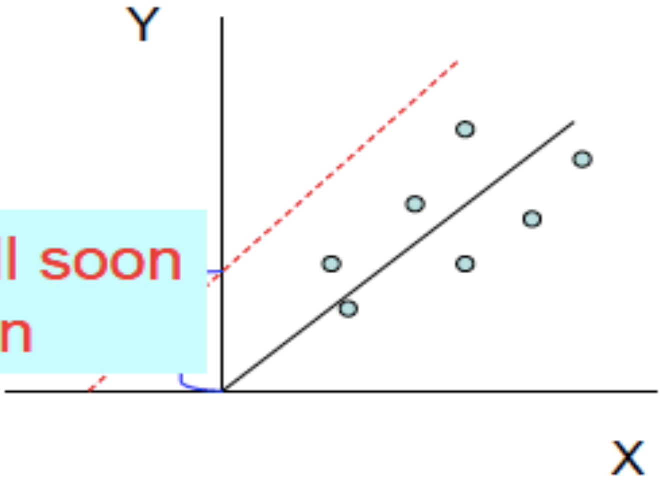
$$w_0 = \frac{\sum_i y_i - w_1 x_i}{n} \qquad w_1 = \frac{\sum_i x_i (y_i - w_0)}{\sum_i x_i^2}$$

Bias term

- So far we assumed that the line passes through the origin
- What if the line does not?
- No problem, simply change the model to

y = w

Just a second, we will soon give a simpler solution



- Can use least squares to determine w_0 , w_1

$$w_0 = \frac{\sum_i y_i - w_1 x_i}{n} \qquad w_1 = \frac{\sum_i x_i (y_i - w_0)}{\sum_i x_i^2}$$

Multivariate regression

- What if we have several inputs?

  - Stock prices for Yahoo, Microsoft and Ebay for the Google prediction task

Multivariate regression

- What if we have several inputs?

  - Stock prices for Yahoo, Microsoft and Ebay for the Google prediction task

- This becomes a multivariate linear regression problem

Multivariate regression

- What if we have several inputs?

  - Stock prices for Yahoo, Microsoft and Ebay for the Google prediction task

- This becomes a multivariate linear regression problem

- Again, its easy to model:

$$y = w_0 + w_1x_1 + \ldots + w_kx_k + \varepsilon$$

Google's stock price

Microsoft's stock price

Yahoo's stock price

Multivariate regression

- What if we have several inputs?

  - Stock prices for Yahoo, Microsoft and Ebay for the Goo

- This be n problem

- Again, its easy to model:

$$y = w_0 + w_1x_1 + \ldots + w_kx_k + \varepsilon$$

Not all functions can be approximated using the input values directly

$$y=10+3x_1^2-2x_2^2+\varepsilon$$

In some cases we would like to use polynomial or other terms based on the input data, are these still linear regression problems?

$$y=10+3x_1^2-2x_2^2+\varepsilon$$

In some cases we would like to use polynomial or other terms based on the input data, are these still linear regression problems?

Yes. As long as the coefficients are linear the equation is still a linear regression problem!

Non-Linear basis function

- So far we only used the observed values
- However, linear regression can be applied in the same way to functions of these values

Non-Linear basis function

- So far we only used the observed values
- However, linear regression can be applied in the same way to functions of these values
- As long as these functions can be directly computed from the observed values the parameters are still linear in the data and the problem remains a linear regression problem

Non-Linear basis function

- So far we only used the observed values
- However, linear regression can be applied in the same way to functions of these values
- As long as these functions can be directly computed from the observed values the parameters are still linear in the data and the problem remains a linear regression problem

$$y = w_0 + w_1 x_1^2 + \ldots + w_k x_k^2 + \varepsilon$$

Non-Linear basis function

Non-Linear basis function

- What type of functions can we use?
- A few common examples:

- Polynomial: $\phi_j(x) = x^j$ for j=0 … n

Non-Linear basis function

- What type of functions can we use?
- A few common examples:

- Polynomial: $\phi_j(x) = x^j$ for j=0 … n

- Gaussian: $\phi_j(x) = \frac{(x - \mu_j)}{2\sigma_j^2}$

Non-Linear basis function

- What type of functions can we use?
- A few common examples:

- Polynomial: $\phi_j(x) = x^j$ for j=0 … n

- Gaussian: $\phi_j(x) = \frac{(x-\mu_j)}{2\sigma_j^2}$

- Sigmoid: $\phi_j(x) = \frac{1}{1+\exp(-s_j x)}$

Any function of the input values can be used. The solution for the parameters of the regression remains the same.

General linear regression problem

General linear regression problem

- Using our new notations for the basis function linear regression can be written as

$$y = \sum_{j=0}^{n} w_j \phi_j(x)$$

General linear regression problem

- Using our new notations for the basis function linear regression can be written as

$$y = \sum_{j=0}^{n} w_j \phi_j(x)$$

- Where $\phi_j(x)$ can be either x_j for multivariate regression or one of the non linear basis we defined

General linear regression problem

- Using our new notations for the basis function linear regression can be written as

$$y = \sum_{j=0}^{n} w_j \phi_j(x)$$

- Where $\phi_j(x)$ can be either x_j for multivariate regression or one of the non linear basis we defined
- Once again we can use 'least squares' to find the optimal solution.

LMS for the general linear regression problem

LMS for the general linear regression problem

$$y = \sum_{j=0}^{n} w_j \phi_j(x)$$

Our goal is to minimize the following loss function:

$$J(\mathrm{w}) = \sum_{i} (y^i - \sum_{j} w_j \phi_j(x^i))^2$$

LMS for the general linear regression problem

$$y=\sum_{j=0}^{n} w_j \phi_j(x)$$

Our goal is to minimize the following loss function:

$$J(\mathrm{w})=\sum_i (y^i-\sum_j w_j \phi_j(x^i))^2$$

w – vector of dimension k+1

ϕ(x^i) – vector of dimension k+1

y^i – a scaler

Moving to vector notations we get:

$$J(\mathrm{w})=\sum_i (y^i-\mathrm{w}^{\mathrm{T}}\phi(x^i))^2$$

LMS for the general linear regression problem

$$y = \sum_{j=0}^{n} w_j \phi_j(x)$$

Our goal is to minimize the following loss function:

$$J(\mathrm{w}) = \sum_i \left(y^i - \sum_j w_j \phi_j(x^i)\right)^2$$

w – vector of dimension k+1

ϕ(x^i) – vector of dimension k+1

y^i – a scaler

Moving to vector notations we get:

$$J(\mathrm{w}) = \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i))^2$$

We take the derivative w.r.t **w**

$$\frac{\partial}{\partial w} \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i))^2 = 2 \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}}$$

LMS for the general linear regression problem

$$y = \sum_{j=0}^{n} w_j \phi_j(x)$$

Our goal is to minimize the following loss function:

$$J(\mathrm{w}) = \sum_i (y^i - \sum_j w_j \phi_j(x^i))^2$$

w – vector of dimension k+1

ϕ(x^i) – vector of dimension k+1

y^i – a scaler

Moving to vector notations we get:

$$J(\mathrm{w}) = \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i))^2$$

We take the derivative w.r.t **w**

$$\frac{\partial}{\partial w} \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i))^2 = 2 \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}}$$

Equating to 0 we get $2 \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}} = 0 \Rightarrow$

LMS for the general linear regression problem

$$y = \sum_{j=0}^{n} w_j \phi_j(x)$$

Our goal is to minimize the following loss function:

$$J(\mathrm{w}) = \sum_i (y^i - \sum_j w_j \phi_j(x^i))^2$$

w – vector of dimension k+1

φ(x^i) – vector of dimension k+1

y^i – a scaler

Moving to vector notations we get:

$$J(\mathrm{w}) = \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i))^2$$

We take the derivative w.r.t **w**

$$\frac{\partial}{\partial w} \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i))^2 = 2 \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}}$$

Equating to 0 we get $2 \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}} = 0 \Rightarrow$

$$\sum_i y^i \phi(x^i)^{\mathrm{T}} = \mathrm{w}^{\mathrm{T}} \left[\sum_i \phi(x^i) \phi(x^i)^{\mathrm{T}} \right]$$

LMS for general linear regression problem

LMS for general linear regression problem

$$J(\mathrm{w})=\sum_i (y^i-\mathrm{w}^{\mathrm{T}}\phi(x^i))^2$$

We take the derivative w.r.t **w**

$$\frac{\partial}{\partial w}\sum_i (y^i-\mathrm{w}^{\mathrm{T}}\phi(x^i))^2=2\sum_i (y^i-\mathrm{w}^{\mathrm{T}}\phi(x^i))\phi(x^i)^{\mathrm{T}}$$

LMS for general linear regression problem

$$J(\mathbf{w}) = \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i))^2$$

We take the derivative w.r.t **w**

$$\frac{\partial}{\partial w} \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i))^2 = 2 \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}}$$

Equating to 0 we get $2 \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}} = \mathbf{0} \Rightarrow$

LMS for general linear regression problem

$$J(\mathbf{w}) = \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i))^2$$

We take the derivative w.r.t **w**

$$\frac{\partial}{\partial w} \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i))^2 = 2 \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}}$$

Equating to 0 we get

$$2 \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}} = \mathbf{0} \Rightarrow$$

$$\sum_i y^i \phi(x^i)^{\mathrm{T}} = \mathbf{w}^{\mathrm{T}} \left[\sum_i \phi(x^i) \phi(x^i)^{\mathrm{T}} \right]$$

LMS for general linear regression problem

$$J(\mathbf{w}) = \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i))^2$$

We take the derivative w.r.t **w**

$$\frac{\partial}{\partial w} \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i))^2 = 2 \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}}$$

Equating to 0 we get

$$2 \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}} = 0 \Rightarrow$$

$$\sum_i y^i \phi(x^i)^{\mathrm{T}} = \mathbf{w}^{\mathrm{T}} \left[\sum_i \phi(x^i) \phi(x^i)^{\mathrm{T}} \right]$$

Define:

$$\Phi = \begin{pmatrix} \phi_0(x^1) & \phi_1(x^1) & \cdots & \phi_m(x^1) \\ \phi_0(x^2) & \phi_1(x^2) & \cdots & \phi_m(x^2) \\ \vdots & \vdots & \cdots & \vdots \\ \phi_0(x^n) & \phi_1(x^n) & \cdots & \phi_m(x^n) \end{pmatrix}$$

LMS for general linear regression problem

$$J(\mathbf{w}) = \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i))^2$$

We take the derivative w.r.t **w**

$$\frac{\partial}{\partial w} \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i))^2 = 2 \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}}$$

Equating to 0 we get

$$2 \sum_i (y^i - \mathbf{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}} = 0 \Rightarrow$$

$$\sum_i y^i \phi(x^i)^{\mathrm{T}} = \mathbf{w}^{\mathrm{T}} \left[\sum_i \phi(x^i) \phi(x^i)^{\mathrm{T}} \right]$$

Define:

$$\Phi = \begin{pmatrix} \phi_0(x^1) & \phi_1(x^1) & \cdots & \phi_m(x^1) \\ \phi_0(x^2) & \phi_1(x^2) & \cdots & \phi_m(x^2) \\ \vdots & \vdots & \cdots & \vdots \\ \phi_0(x^n) & \phi_1(x^n) & \cdots & \phi_m(x^n) \end{pmatrix}$$

Then deriving w
we get:

$$\mathbf{w} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{y}$$

LMS for general linear regression problem

LMS for general linear regression problem

$$J(\mathbf{w}) = \sum_i (y^i - \mathbf{w}^T \phi(x^i))^2$$

Deriving w we get: $\mathbf{w} = (\Phi^T\Phi)^{-1}\Phi^T\mathbf{y}$

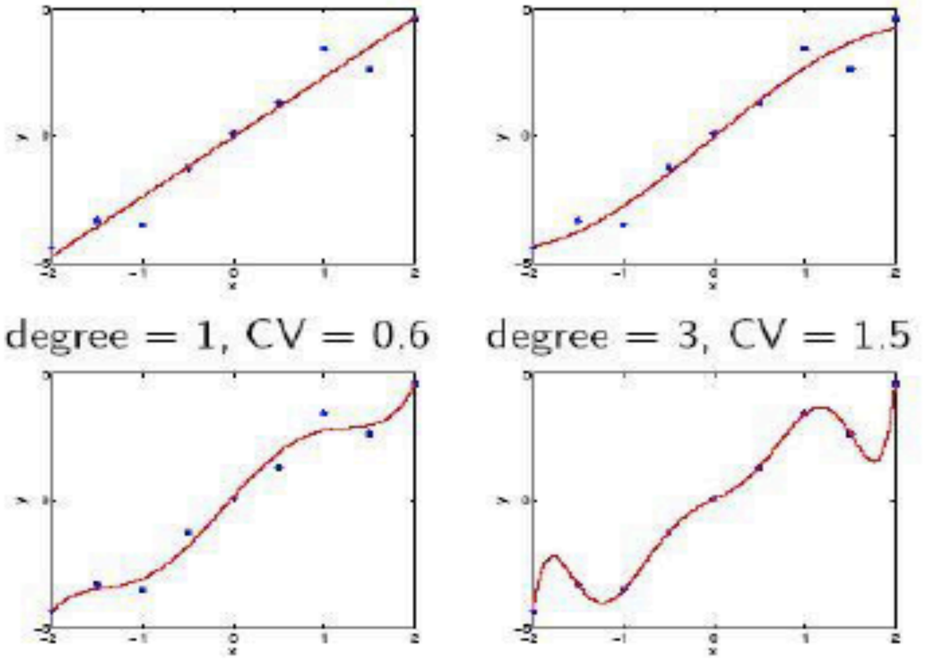
k+1 entries vector

n by k+1 matrix

n entries vector

This solution is also known as ‘psuedo inverse’

Example: Polynomial regression



degree = 1, CV = 0.6   degree = 3, CV = 1.5

degree = 5, CV = 6.0   degree = 7, CV = 15.6

A probabilistic interpretation

Our least squares minimization solution can also be motivated by a probabilistic in interpretation of the regression problem: $y = \mathbf{w}^{\mathrm{T}} \phi(x) + \varepsilon$

A probabilistic interpretation

Our least squares minimization solution can also be motivated by a probabilistic in interpretation of the regression problem: $y = \mathbf{w}^{\mathrm{T}}\phi(x) + \varepsilon$

The MLE for w in this model is the same as the solution we derived for least squares criteria:

$$\mathbf{w} = (\Phi^T\Phi)^{-1}\Phi^T\mathbf{y}$$

Other types of linear regression

- Linear regression is a useful model for many problems
- However, the parameters we learn for this model are **global**; they are the same regardless of the value of the input x
- Extension to linear regression adjust their parameters based on the region of the input we are dealing with

That’s all!