CS 360: Machine Learning

Prof. Sara Mathieson

Fall 2019

HAVERFORD
COLLEGE

Outline for October 31

- Reading Quiz
- Recap Perceptron Algorithm
- Introduction to Support Vector Machines
- Lab check in TODAY! (Parts 1&2 complete)

Outline for October 31

- Reading Quiz
- Recap Perceptron Algorithm
- Introduction to Support Vector Machines

Reading Quiz

1. What is the goal of the perceptron algorithm? Circle all that apply:

   (a) predict a continuous outcome

   (b) quantify how important each feature is for predicting the outcome

   (c) create a linear decision boundary between positives and negatives

   (d) obtain the probability of a positive label for each test example

Reading Quiz

1. What is the goal of the perceptron algorithm? Circle all that apply:
    (a) predict a continuous outcome
    [✓ (b) quantify how important each feature is for predicting the outcome]
    [✓ (c) create a linear decision boundary between positives and negatives]
    (d) obtain the probability of a positive label for each test example

2. *True or False*: The perceptron algorithm was inspired by how neurons are activated in our brains.

Reading Quiz

1. What is the goal of the perceptron algorithm? Circle all that apply:

   (a) predict a continuous outcome

   [✓ (b) quantify how important each feature is for predicting the outcome]

   [✓ (c) create a linear decision boundary between positives and negatives]

   (d) obtain the probability of a positive label for each test example

2. *True or False*: The perceptron algorithm was inspired by how neurons are activated in our brains.

   True

3. Say at some point in the perceptron algorithm I have $\vec{w} = [3, -1, 2]^T$ and $\vec{x} = [1, 2, -2]^T$. What label would we predict for $\vec{x}$?

Reading Quiz

1. What is the goal of the perceptron algorithm? Circle all that apply:
    - (a) predict a continuous outcome
    - [✓ (b) quantify how important each feature is for predicting the outcome]
    - [✓ (c) create a linear decision boundary between positives and negatives]
    - (d) obtain the probability of a positive label for each test example

2. *True or False*: The perceptron algorithm was inspired by how neurons are activated in our brains.

True

3. Say at some point in the perceptron algorithm I have $\vec{w} = [3, -1, 2]^T$ and $\vec{x} = [1, 2, -2]^T$. What label would we predict for $\vec{x}$?

Dot product = -3 => predict label -1

4. In the example above, say the true label is -1. How would the weights be updated when using this point?

Reading Quiz

1. What is the goal of the perceptron algorithm? Circle all that apply:

   (a) predict a continuous outcome

   [✓ (b) quantify how important each feature is for predicting the outcome]

   [✓ (c) create a linear decision boundary between positives and negatives]

   (d) obtain the probability of a positive label for each test example

2. *True or False*: The perceptron algorithm was inspired by how neurons are activated in our brains.

   True

3. Say at some point in the perceptron algorithm I have $\vec{w} = [3, -1, 2]^T$ and $\vec{x} = [1, 2, -2]^T$. What label would we predict for $\vec{x}$?

   Dot product = -3 => predict label -1

4. In the example above, say the true label is -1. How would the weights be updated when using this point?

   No weight update!

Outline for October 31

- Reading Quiz

- Recap Perceptron Algorithm

- Introduction to Support Vector Machines

Perceptron as a neural network

Biological model of a neuron

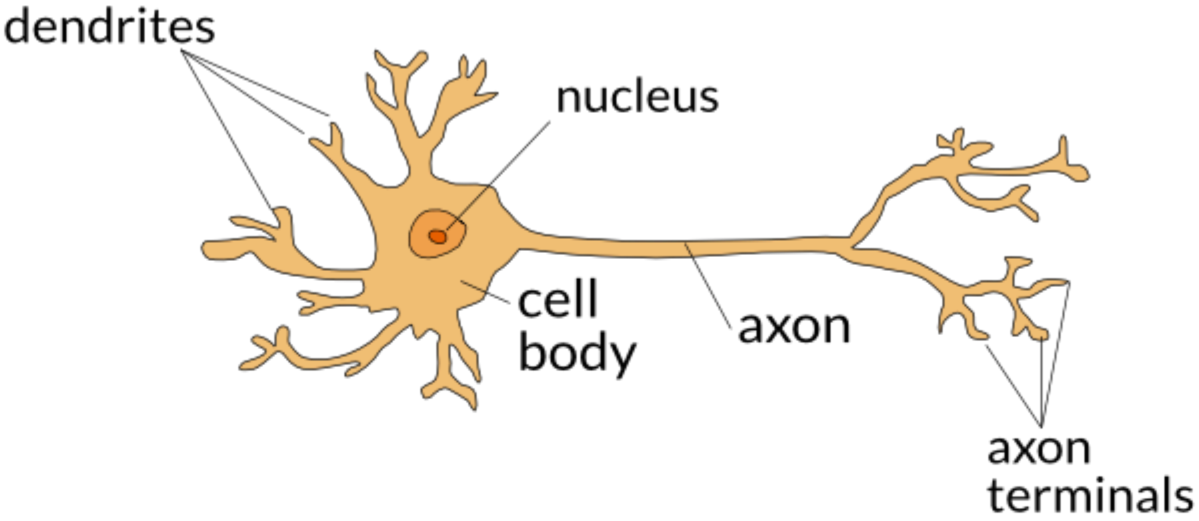


Image: modified from "Applied Go"

Perceptron as a neural network

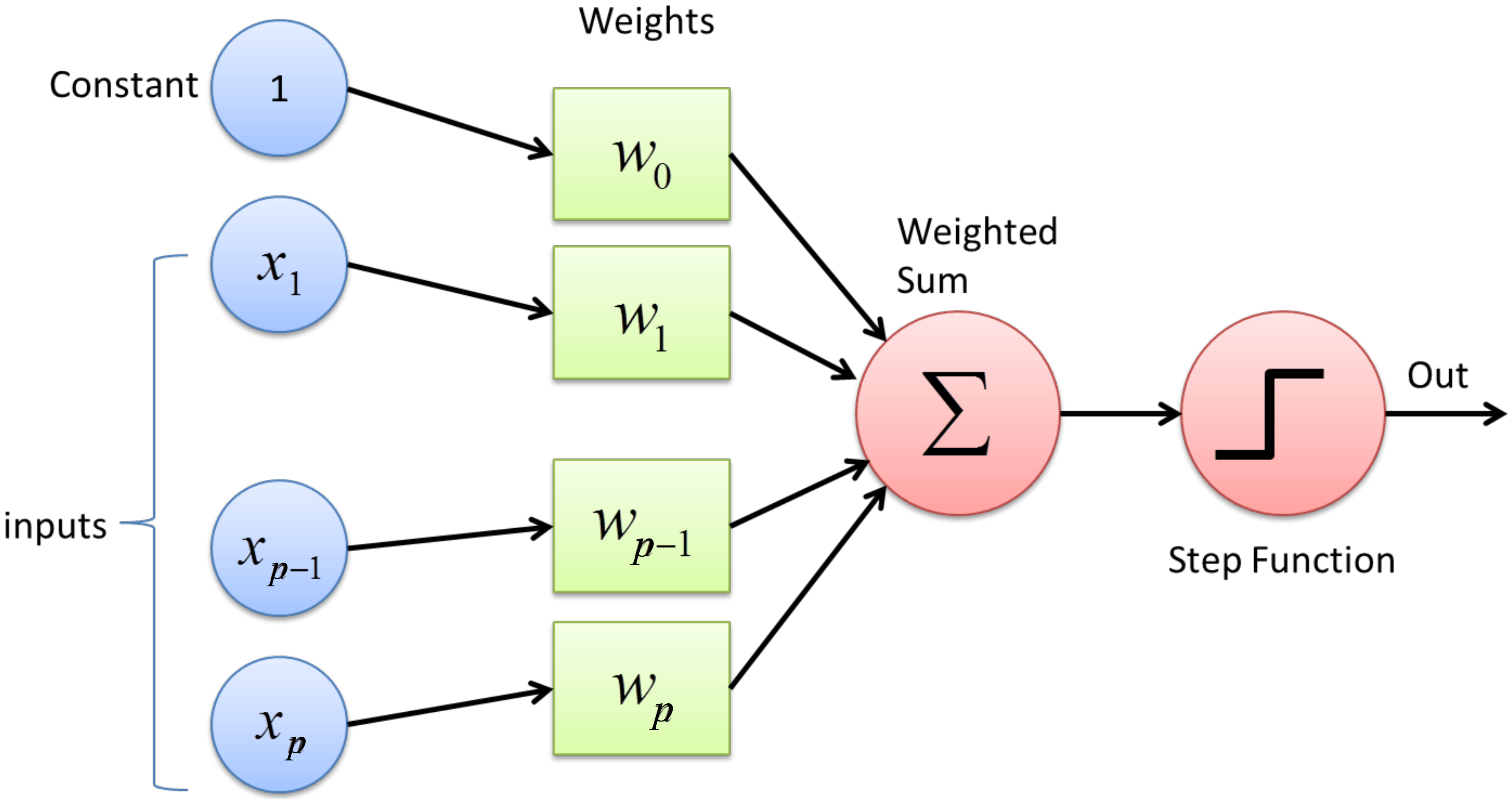


Image: modified from “Towards Data Science”

History of the Perceptron

- Invented in 1957 by Frank Rosenblatt
- Initially thought to be the "solution to AI"

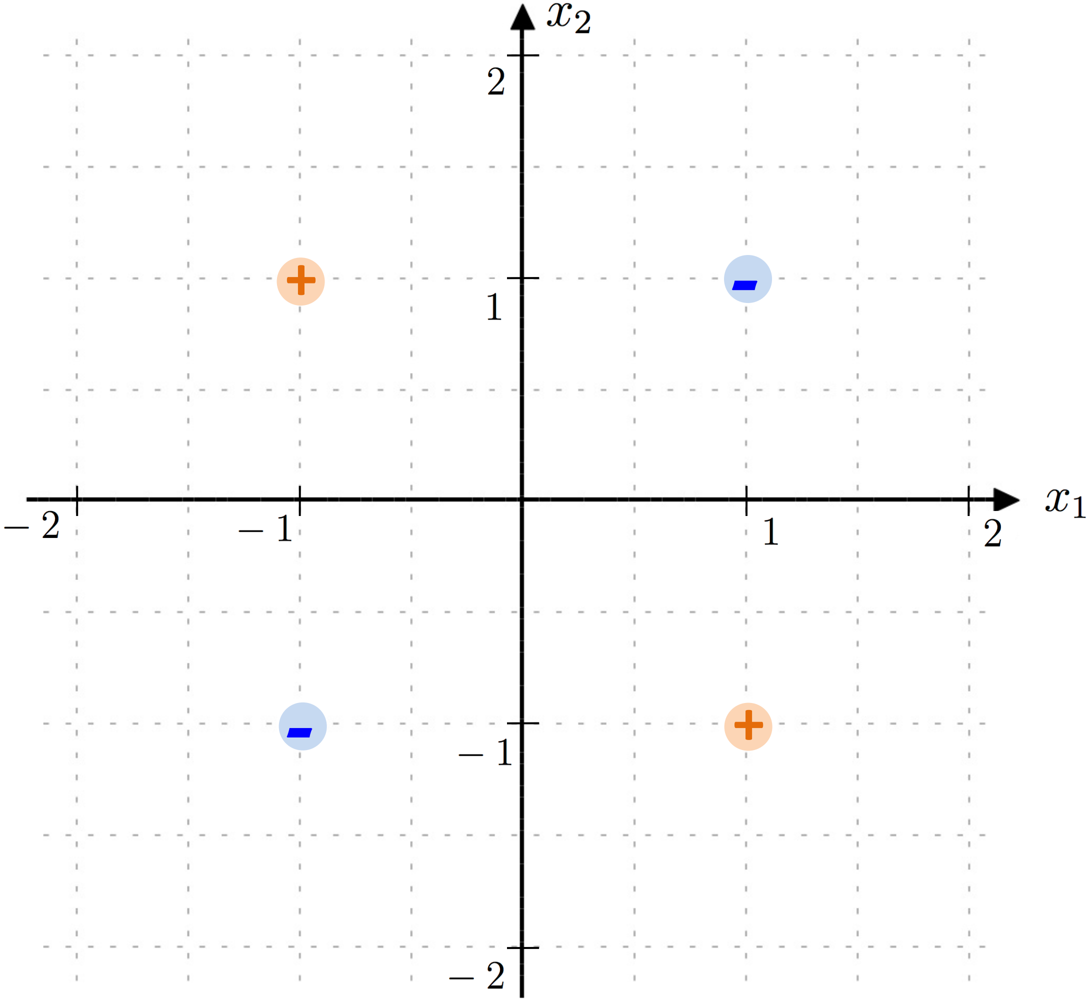
  NYT said the perceptron was "*the embryo of an electronic computer that [the Navy] expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence*"

- Famous book "Perceptrons" by Marvin Minsky and Seymour Papert (1969)
- Confusion about the text contributed to first "AI winter"

Perceptron cannot learn XOR

($x_1 = 1$ or $x_2 = 1$, but not both)

Why?

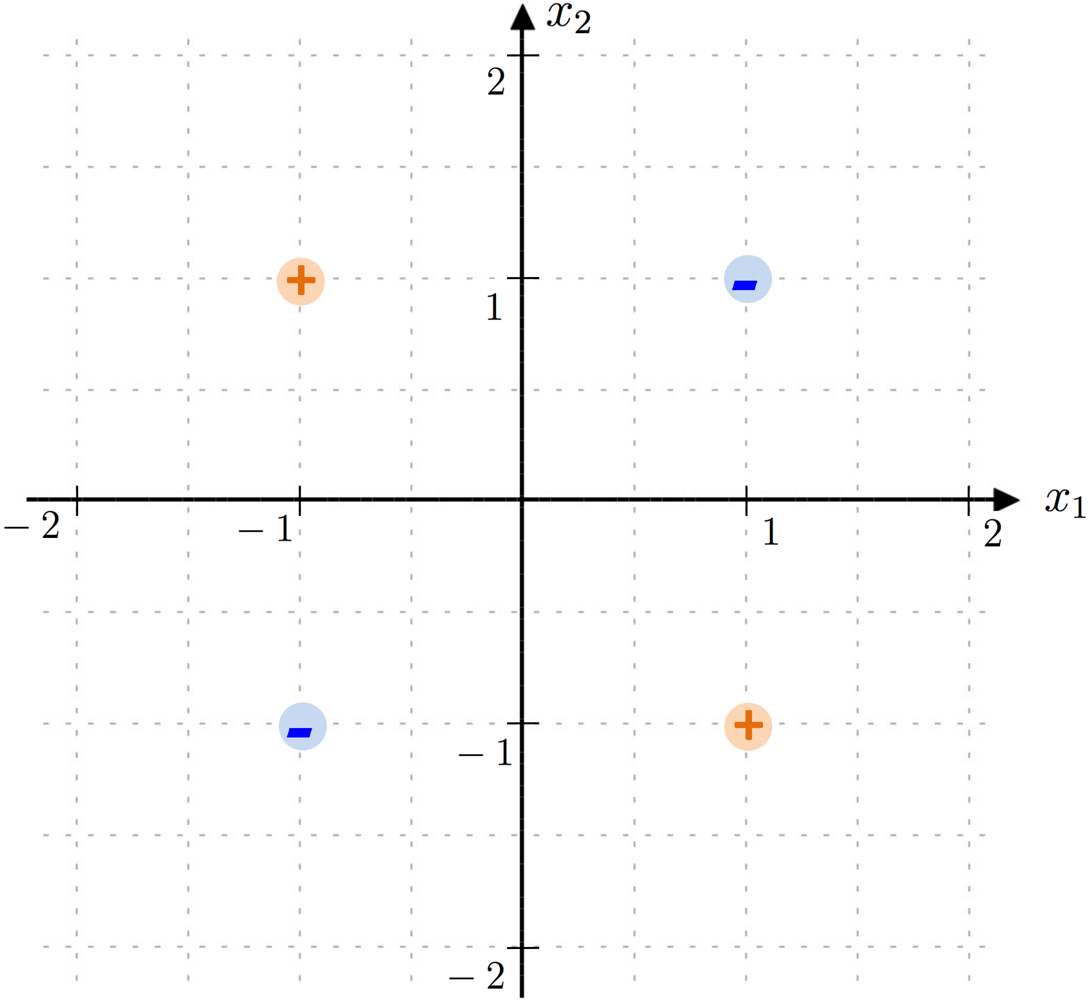


Perceptron cannot learn XOR

($x_1 = 1$ or $x_2 = 1$, but not both)

Why?

Not linearly separable!



Convergence Guarantee

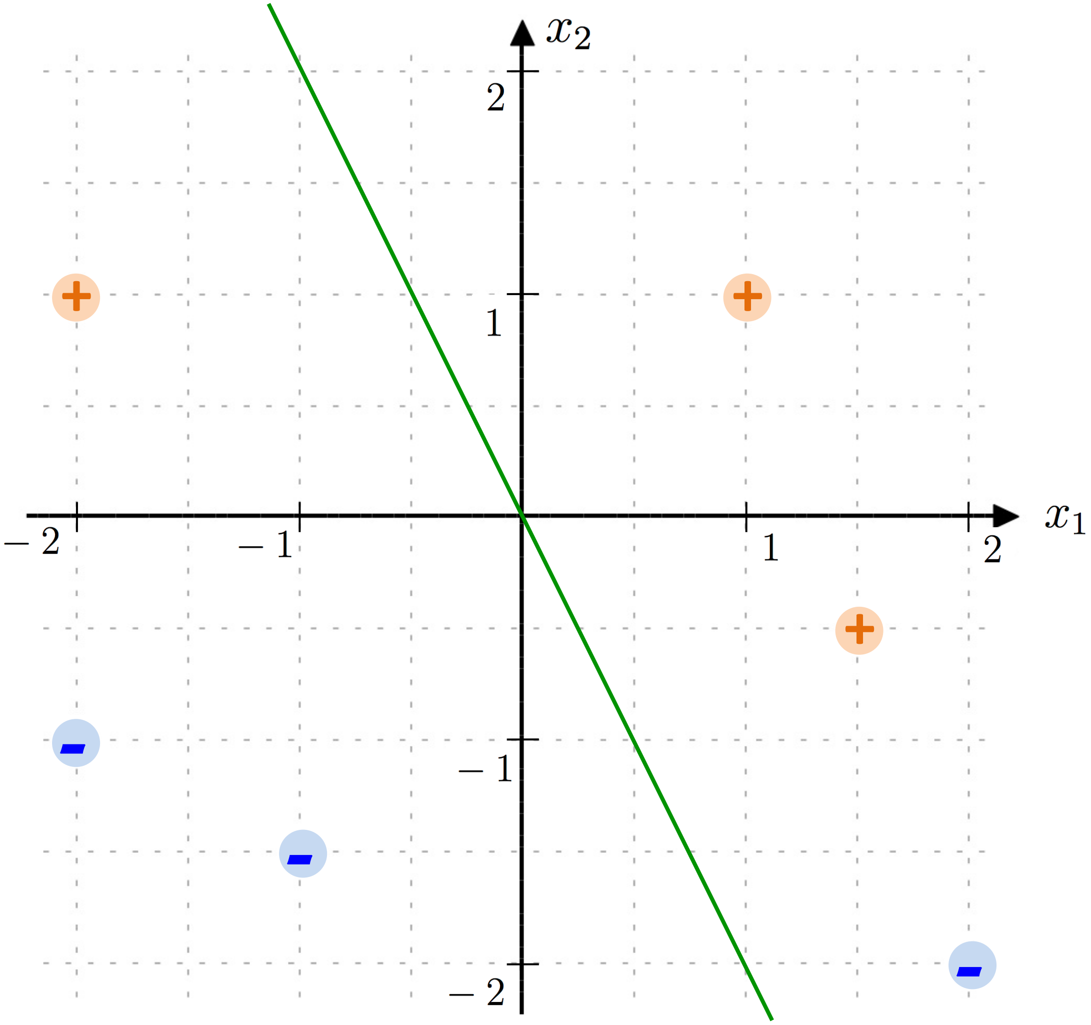
- Perceptron is guaranteed to converge to a solution if a separating hyperplane exists
- Not guaranteed to converge to a “good” solution
- No guarantees about behavior if a separating hyperplane does not exist!

Handout 15 example

Initial values:

$$\alpha = 0.2$$

$$\vec{w} = \begin{bmatrix} 0 \\ 1 \\ 0.5 \end{bmatrix}$$



Example modified from Achim J. Lilienthal & Thorsteinn Rögnvaldsson

Handout 15 example

$$\alpha = 0.2$$

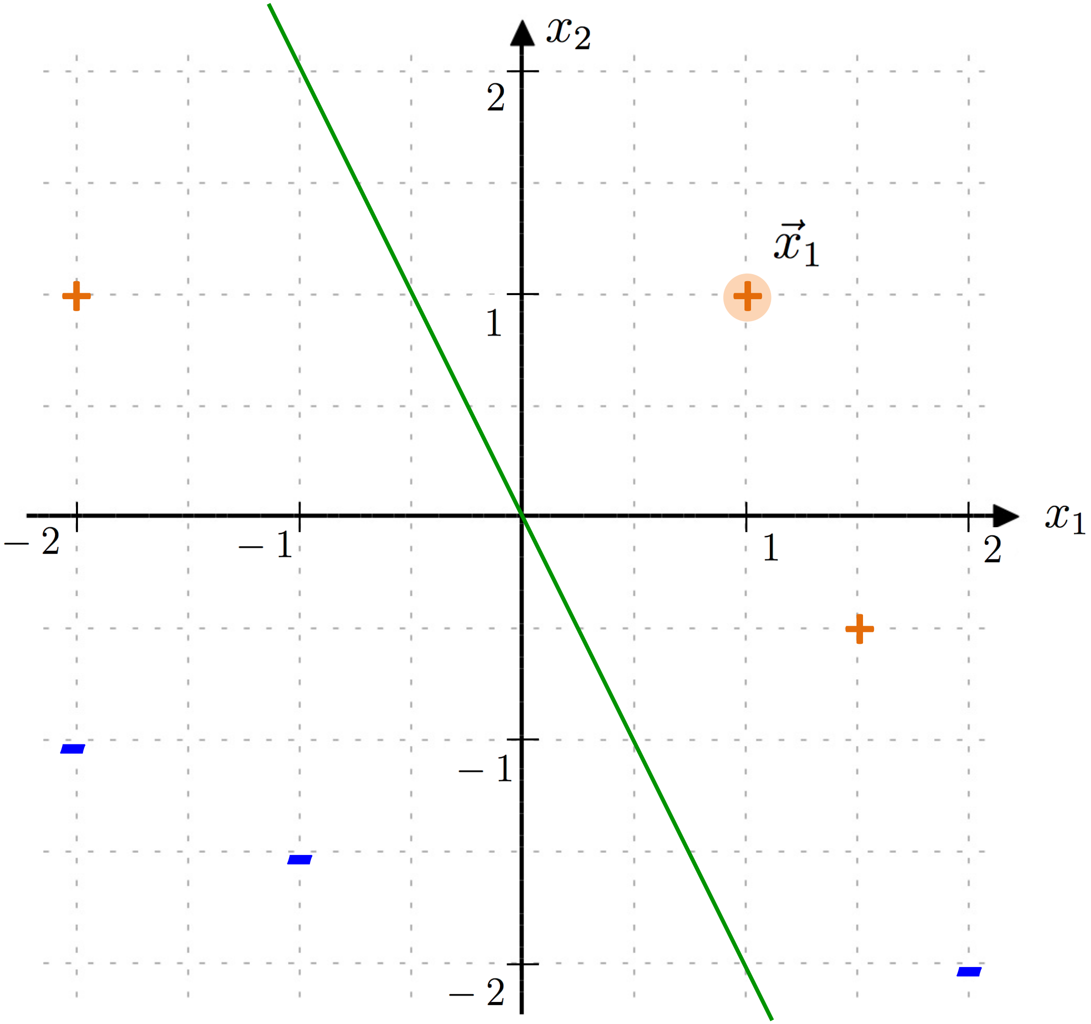
$$\vec{w} = \begin{bmatrix} 0 \\ 1 \\ 0.5 \end{bmatrix}$$

Round 1:

$$\vec{x}_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\vec{w} \cdot \vec{x}_1 > 0$$

Correct classification, no action



Example modified from Achim J. Lilienthal & Thorsteinn Rögnvaldsson

Handout 15 example

$$\alpha = 0.2$$

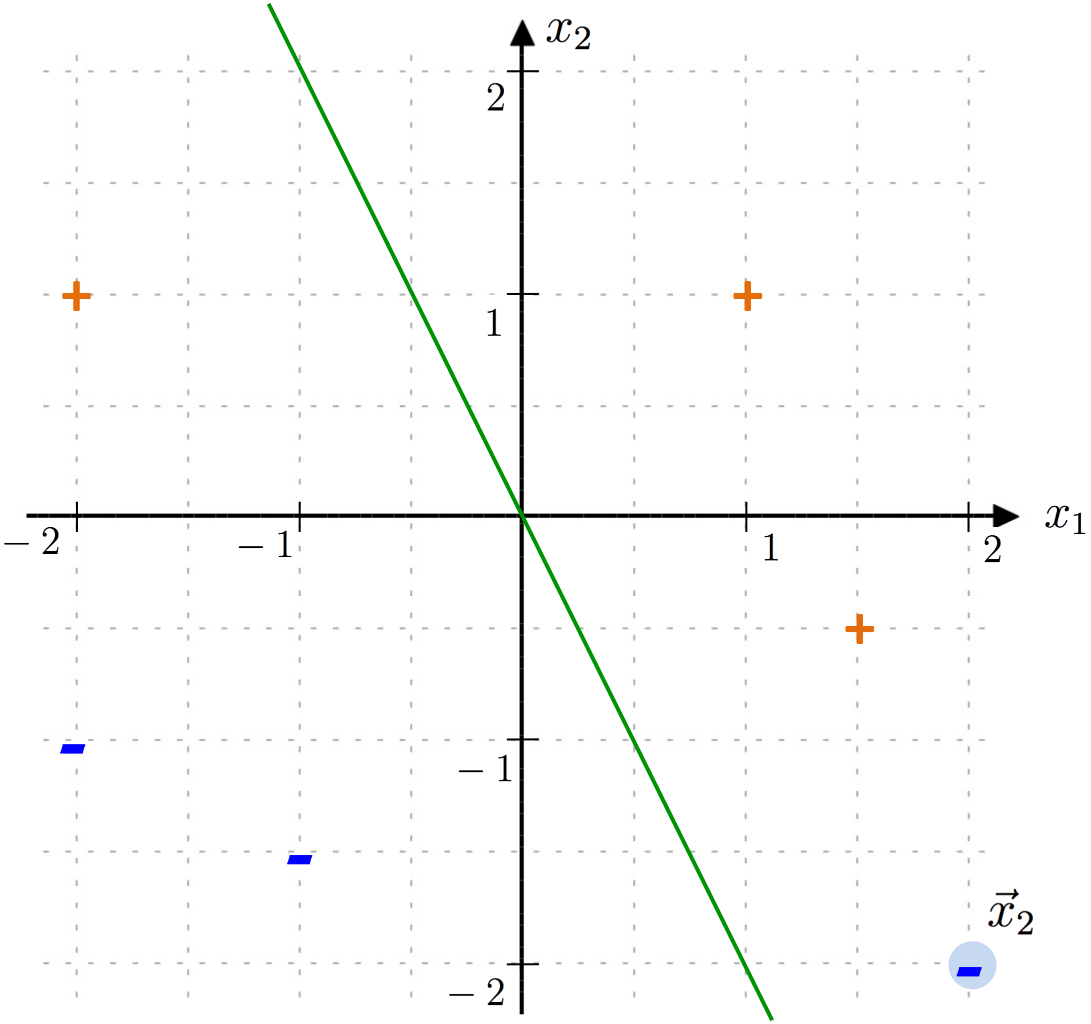
$$\vec{w} = \begin{bmatrix} 0 \\ 1 \\ 0.5 \end{bmatrix}$$

Round 2:

$$\vec{x}_2 = \begin{bmatrix} 1 \\ 2 \\ -2 \end{bmatrix}$$

$$\vec{w} \cdot \vec{x}_2 > 0$$

Incorrect classification



Example modified from Achim J. Lilienthal & Thorsteinn Rögnvaldsson

Handout 15 example

$$\alpha = 0.2$$

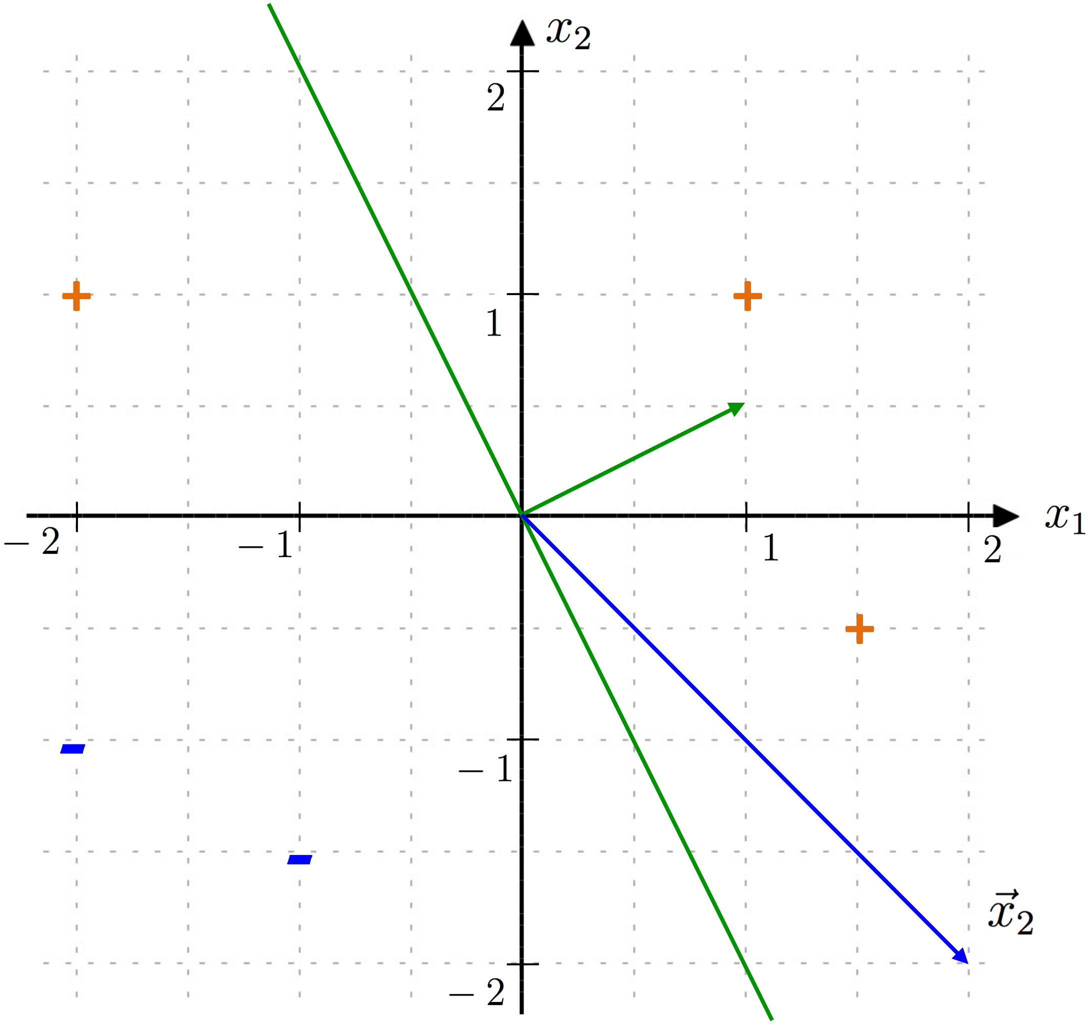
$$\vec{w} = \begin{bmatrix} 0 \\ 1 \\ 0.5 \end{bmatrix}$$

Round 2:

$$\vec{x}_2 = \begin{bmatrix} 1 \\ 2 \\ -2 \end{bmatrix}$$

$$\vec{w} \cdot \vec{x}_2 > 0$$

Incorrect classification



Example modified from Achim J. Lilienthal & Thorsteinn Rögnvaldsson

Handout 15 example

$$\alpha = 0.2$$

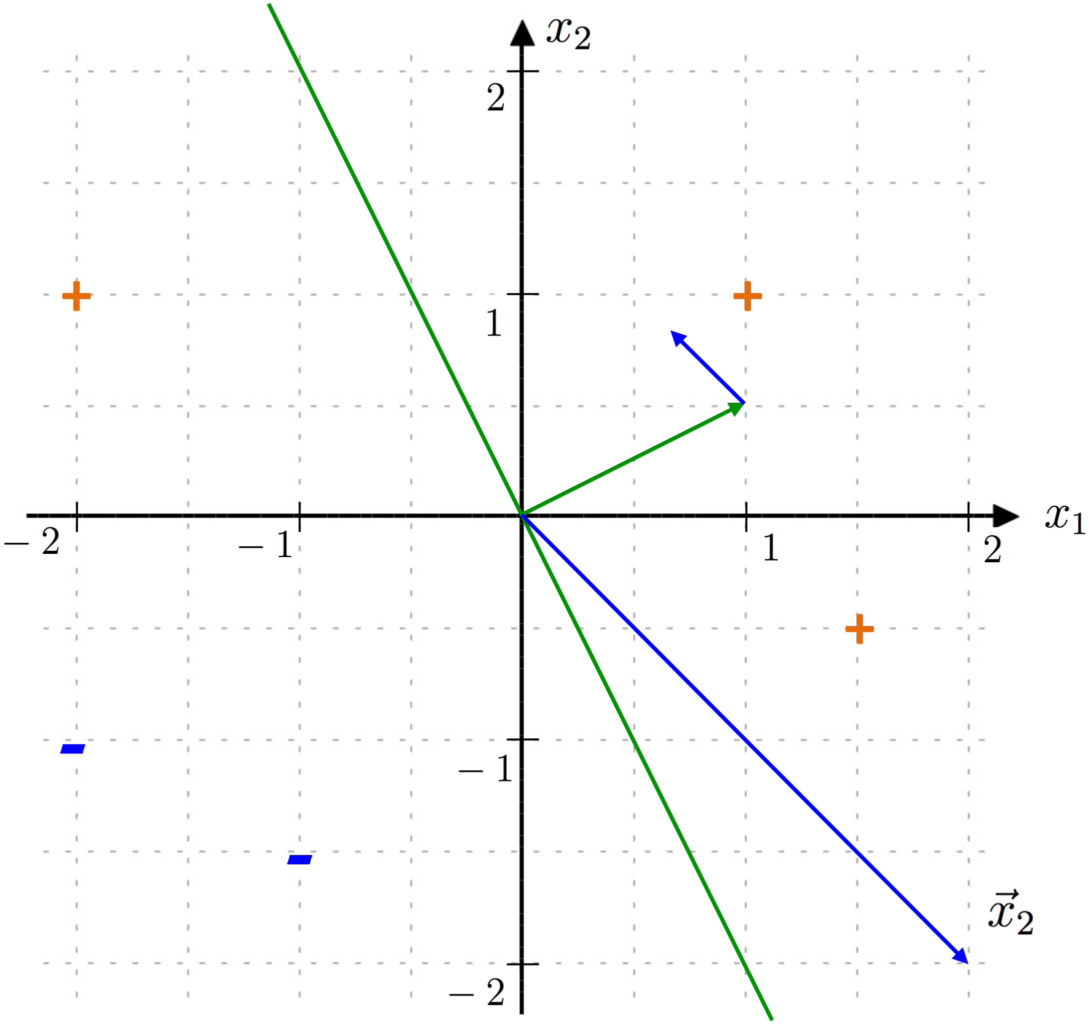
$$\vec{w} = \begin{bmatrix} 0 \\ 1 \\ 0.5 \end{bmatrix}$$

Round 2:

$$\vec{x}_2 = \begin{bmatrix} 1 \\ 2 \\ -2 \end{bmatrix}$$

$$\vec{w} \cdot \vec{x}_2 > 0$$

Incorrect classification
"Push" ***w*** away from negative point



Example modified from Achim J. Lilienthal & Thorsteinn Rögnvaldsson

Handout 15 example

$$\alpha = 0.2$$

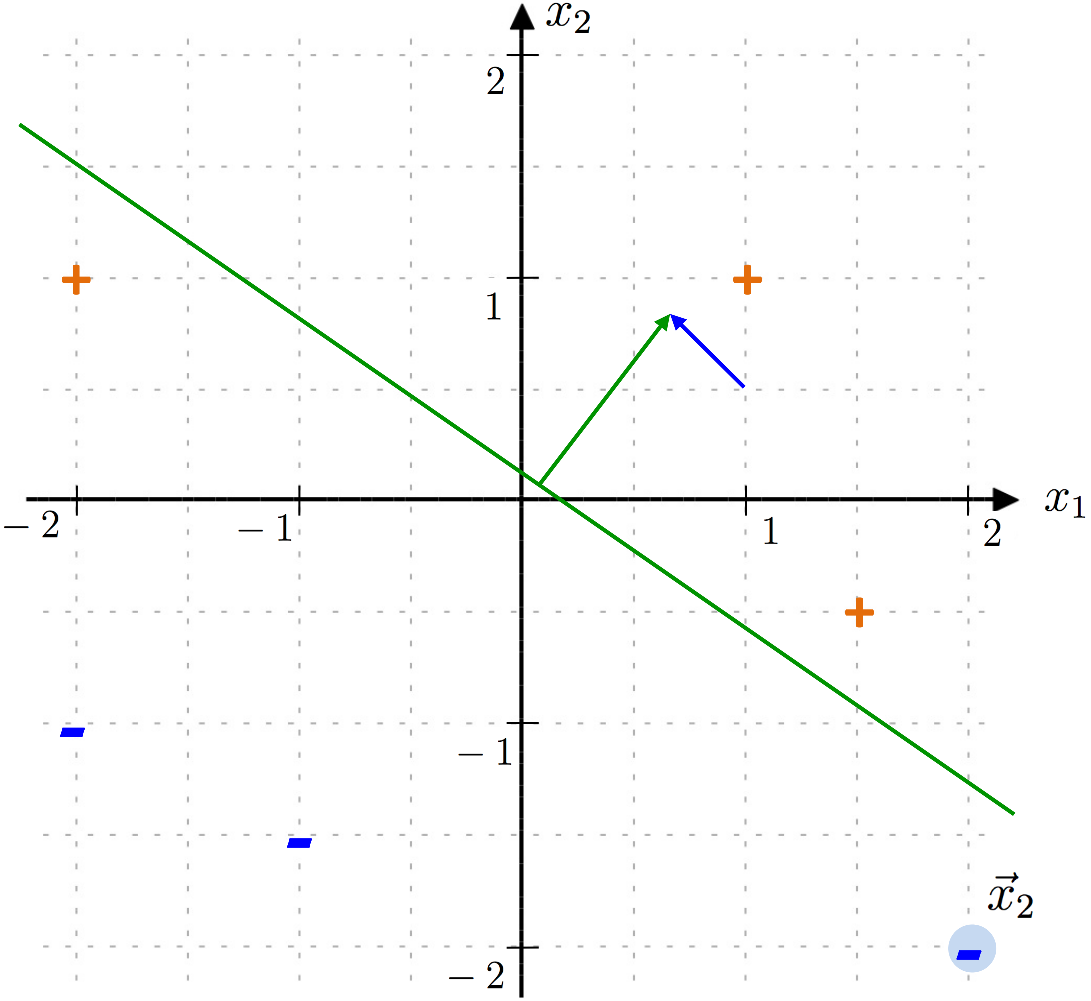
$$\vec{w} = \begin{bmatrix} 0 \\ 1 \\ 0.5 \end{bmatrix}$$

Round 2:

$$\vec{x}_2 = \begin{bmatrix} 1 \\ 2 \\ -2 \end{bmatrix}$$

$$\vec{w} \cdot \vec{x}_2 > 0$$

Incorrect classification
"Push" ***w*** away from negative point



Example modified from Achim J. Lilienthal & Thorsteinn Rögnvaldsson

Handout 15 example

Final solution (so you can check your work):

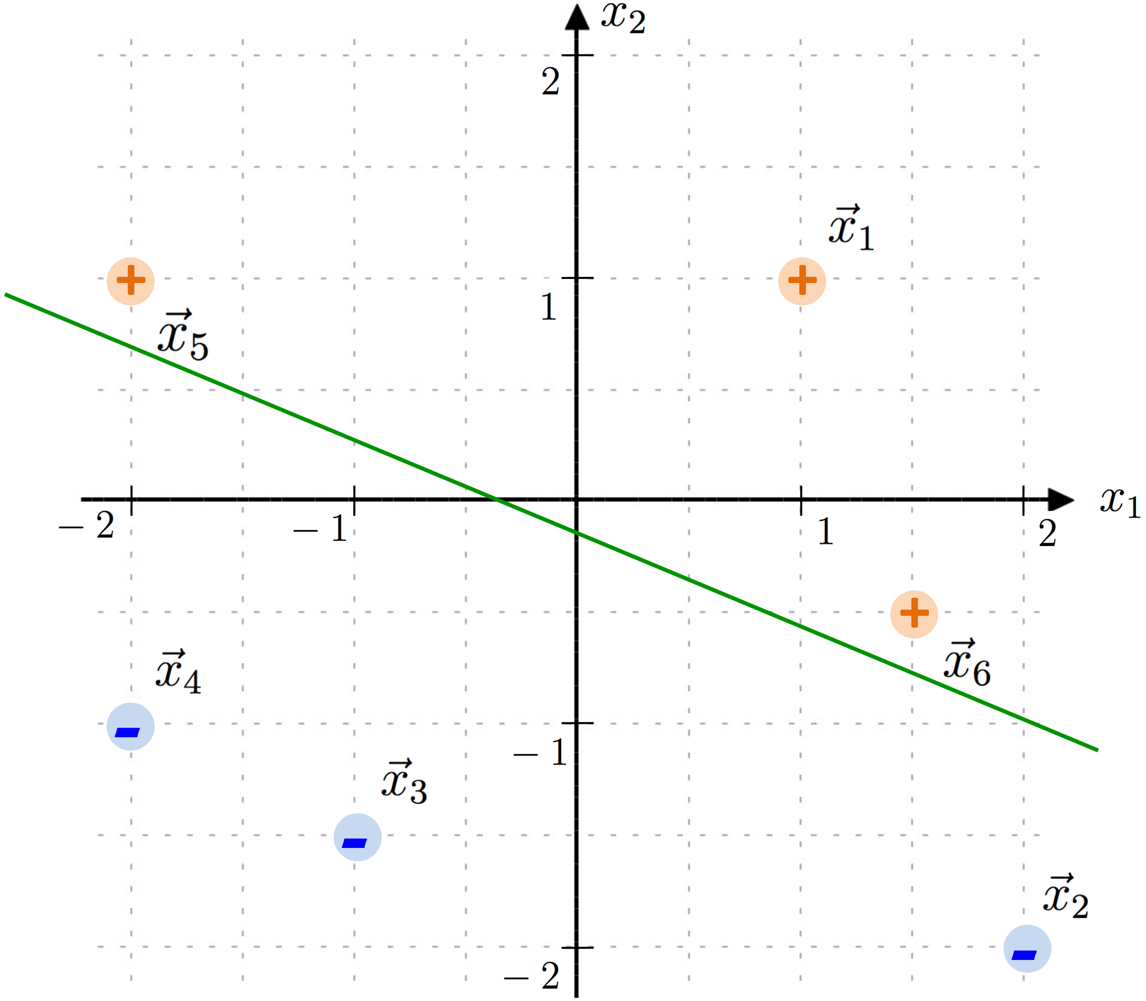
$$\vec{w}^* = \begin{bmatrix} 0.2 \\ 0.5 \\ 1 \end{bmatrix}$$

Final hyperplane:

$$0.2 + 0.5x_1 + x_2 = 0$$

$$\Rightarrow$$

$$x_2 = -0.2 - 0.5x_1$$



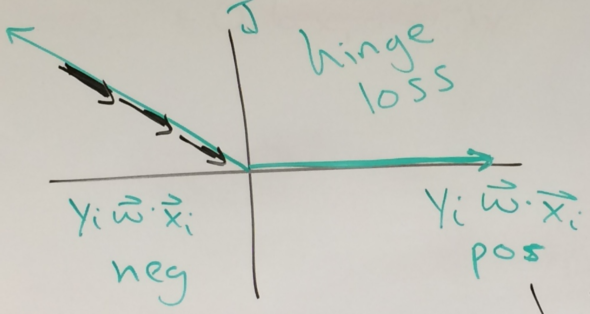
Example modified from Achim J. Lilienthal & Thorsteinn Rögnvaldsson

Perceptron Updates

in terms of cost.

$$J(\vec{w}) = \sum_{i=1}^{n} \max(0, \underbrace{-y_i \vec{w}\cdot\vec{x}_i})$$

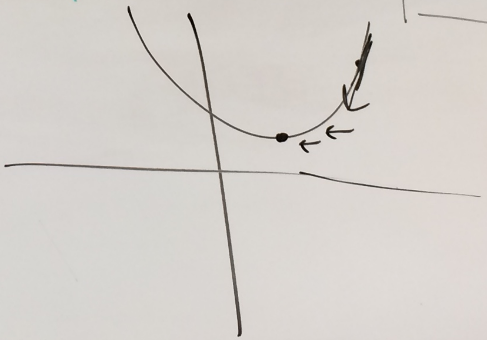
if same sign then max is 0. (correct)



$$\boxed{\nabla J(\vec{w}) = -y_i \vec{x}_i}$$

updates. SGD

$$\boxed{\vec{w} \leftarrow \vec{w} + \alpha y_i \vec{x}_i}$$



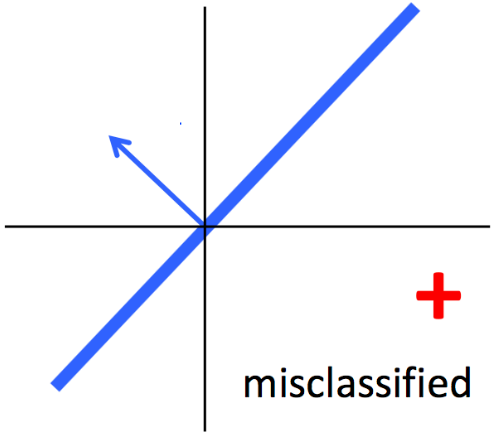
Informal discussion with a partner

1) What is the relationship between the weight vector ***w*** and the hyperplane?

2) Why is the perceptron cost function intuitive?

$$J(\vec{w}) = \sum_{i=1}^{n} \max\left(0, -y_i(\vec{w}^T \vec{x}_i)\right)$$

3) In the example to the right, how will the slope of the hyperplane change?



4) What are the weaknesses of the perceptron? Create a binary classifier "wishlist".

Informal discussion with a partner

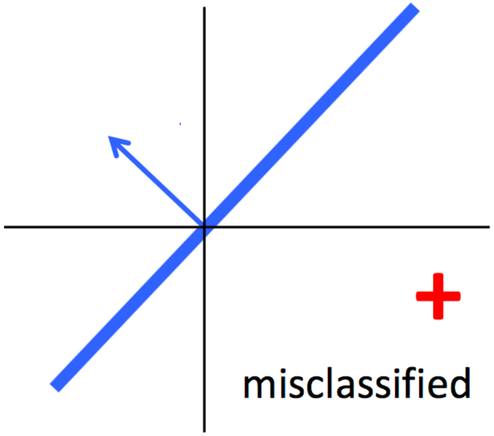
1) What is the relationship between the weight vector ***w*** and the hyperplane?

They are perpendicular

2) Why is the perceptron cost function intuitive?

$$J(\vec{w}) = \sum_{i=1}^{n} \max\left(0, -y_i(\vec{w}^T \vec{x}_i)\right)$$

3) In the example to the right, how will the slope of the hyperplane change?



4) What are the weaknesses of the perceptron? Create a binary classifier "wishlist".

Informal discussion with a partner

1) What is the relationship between the weight vector ***w*** and the hyperplane?

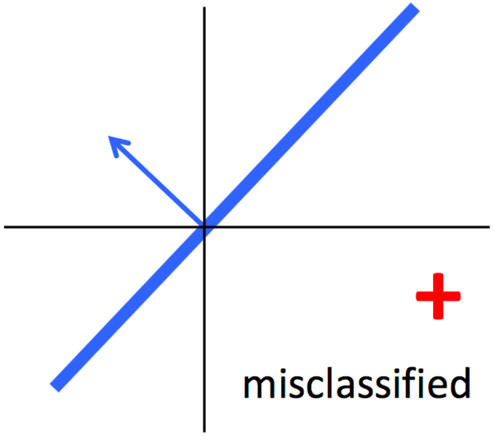
They are perpendicular

2) Why is the perceptron cost function intuitive?

$$J(\vec{w}) = \sum_{i=1}^{n} \max\left(0, -y_i(\vec{w}^T \vec{x}_i)\right)$$

Cost function is 0 when classification is correct, and positive when incorrect

3) In the example to the right, how will the slope of the hyperplane change?



4) What are the weaknesses of the perceptron? Create a binary classifier "wishlist".

Perceptron algorithm and intuition

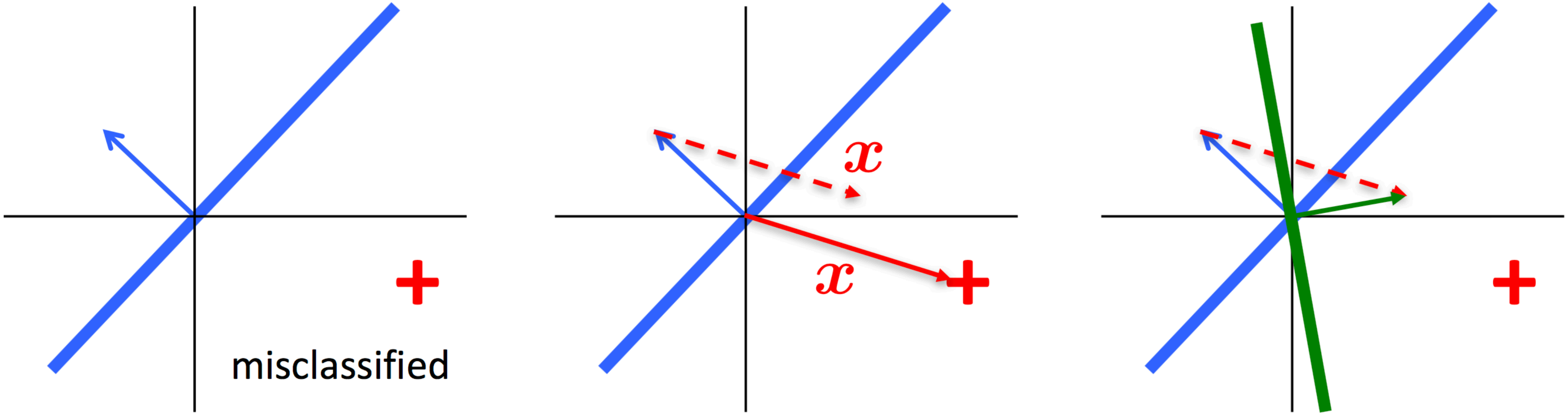
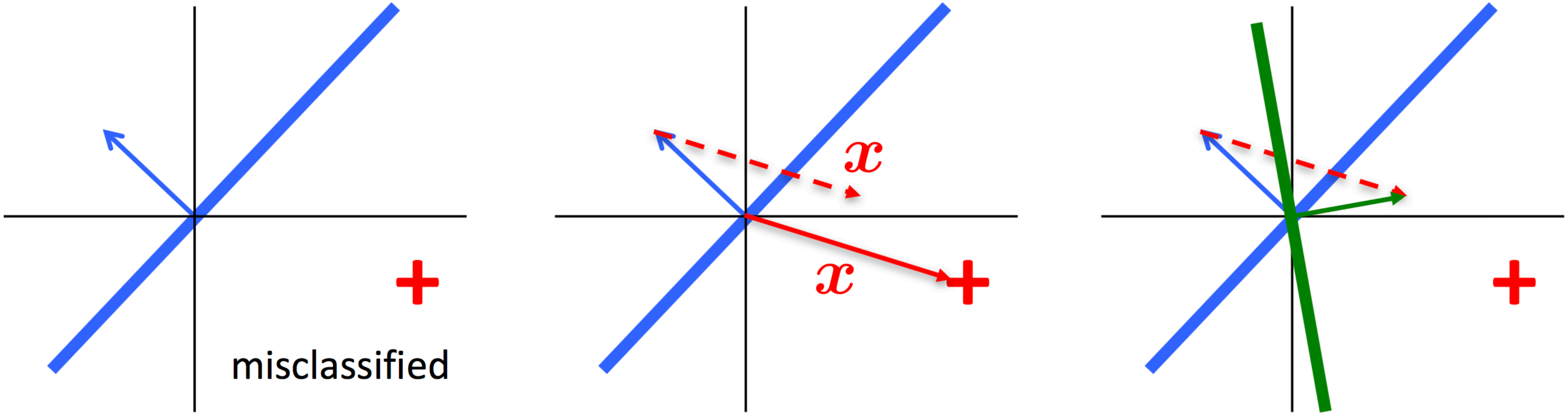


Image and Algorithm: modified from Eric Eaton

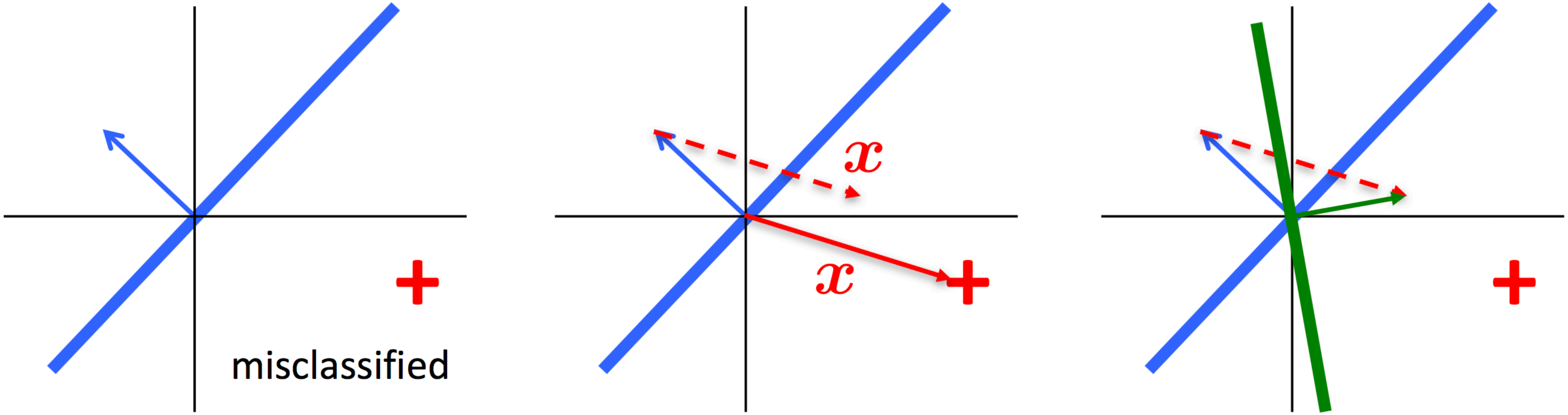
Perceptron algorithm and intuition



```
Let $\vec{w} = [0, 0, \cdots, 0]^T$
Repeat until convergence:
    Receive training example $(\vec{x}_i, y_i)$
    If $y_i(\vec{w}^T \vec{x}_i) \leq 0$    (incorrectly classified)
        $\vec{w} \leftarrow \vec{w} + \alpha y_i \vec{x}_i$
```

Image and Algorithm: modified from Eric Eaton

Perceptron algorithm and intuition



```
Let $\vec{w} = [0, 0, \cdots, 0]^T$
Repeat until convergence:
    Receive training example $(\vec{x}_i, y_i)$
    If $y_i(\vec{w}^T \vec{x}_i) \leq 0$ (incorrectly classified)
        $\vec{w} \leftarrow \vec{w} + \alpha y_i \vec{x}_i$
```

Convergence:

- All data points correctly classified
- Fixed number of iterations passed

Often: alpha = 1 (only changes magnitude of weight vector)

Image and Algorithm: modified from Eric Eaton

Binary classifier wishlist

- If data is linearly separable, want a “good” hyperplane (idea: far from points close to the boundary)

- If data is not linearly separable, want something reasonable (not just give up or fail to converge)

- Might not want to constrain ourselves to linear separators

Outline for October 31

- Reading Quiz
- Recap Perceptron Algorithm
- Introduction to Support Vector Machines

Support Vector Machines (SVMs)

- Will give us everything on our wishlist!
- Often considered the best “off the shelf” binary classifier
- Widely used in many fields

Brief history

- 1963: Initial idea by Vladimir Vapnik and Alexey Chervonenkis
- 1992: nonlinear SVMs by Bernhard Boser, Isabelle Guyon and Vladimir Vapnik
- 1993: “soft-margin” by Corinna Cortes and Vladimir Vapnik

Idea: “best” hyperplane has a large margin

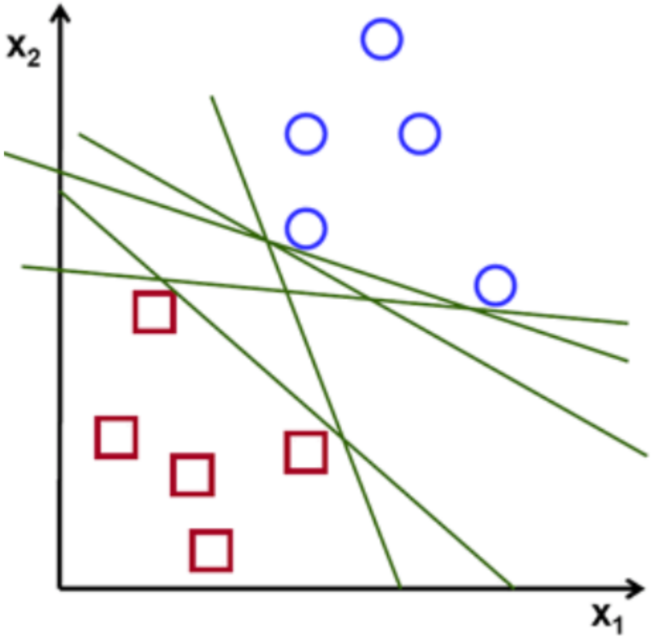


Image: “Towards Data Science”

Idea: “best” hyperplane has a large margin

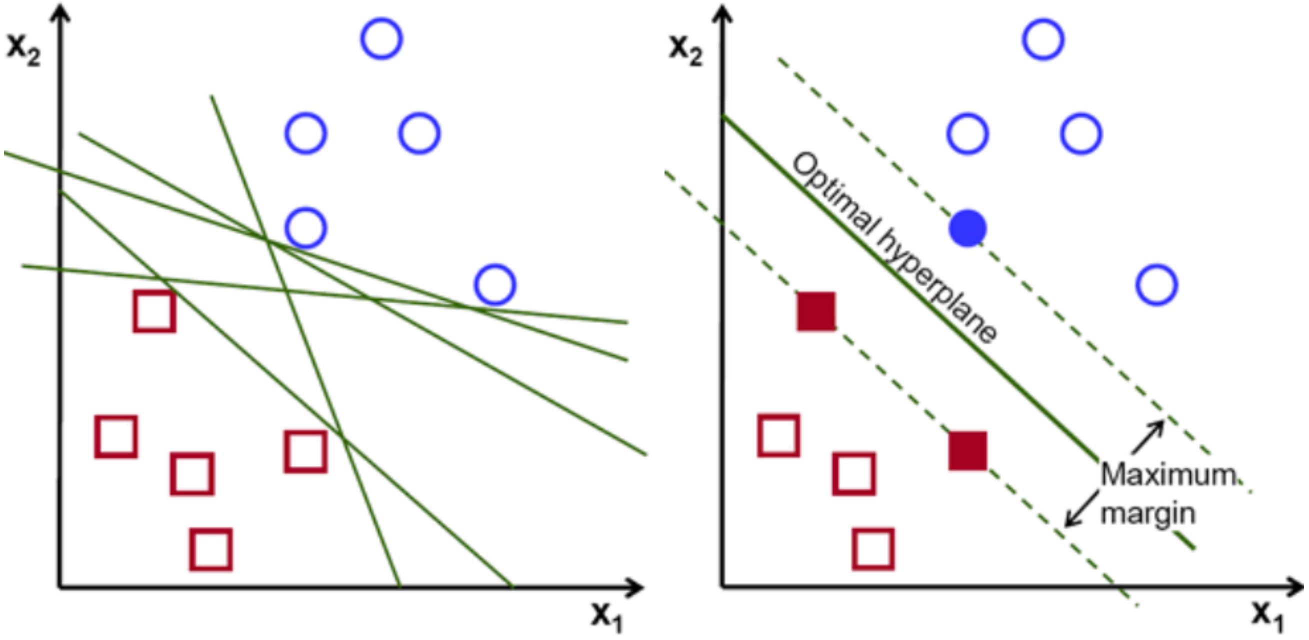
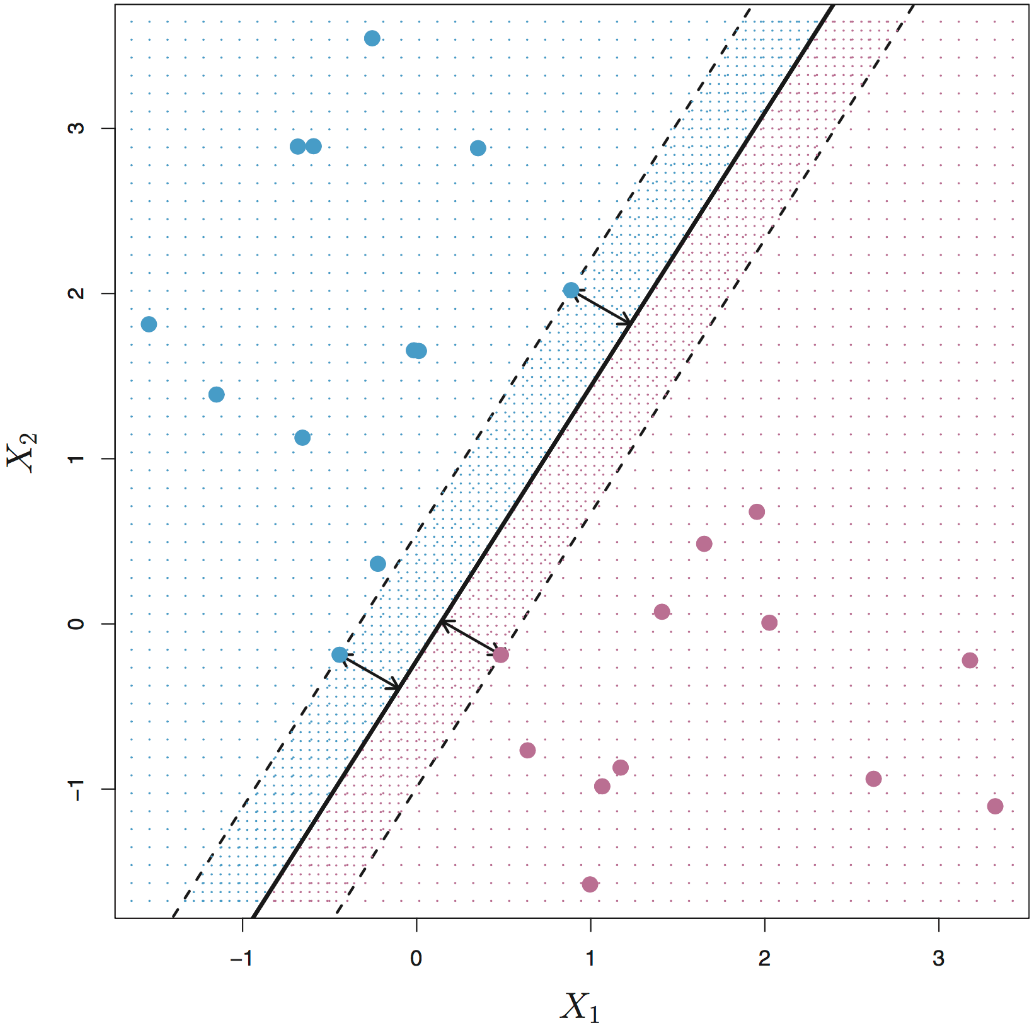


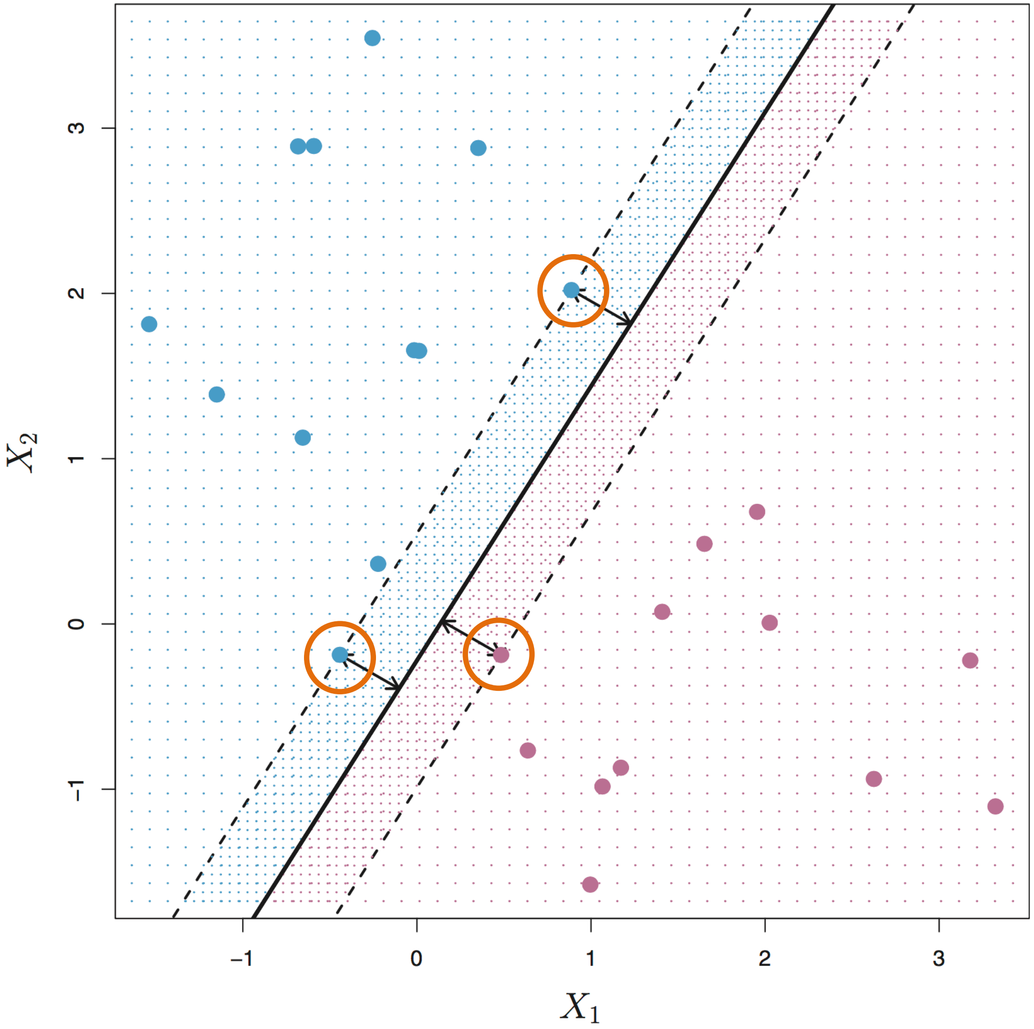
Image: “Towards Data Science”

Datapoints that lie on the margin are called "support vectors"



Modified from ISL: Figure 9.3

Datapoints that lie on the margin are called "support vectors"



Support vectors

Modified from ISL: Figure 9.3

Support Vector Machines

let $h(\vec{x}) = \text{sign}(\vec{w}\cdot\vec{x}+b)$ same as perceptron

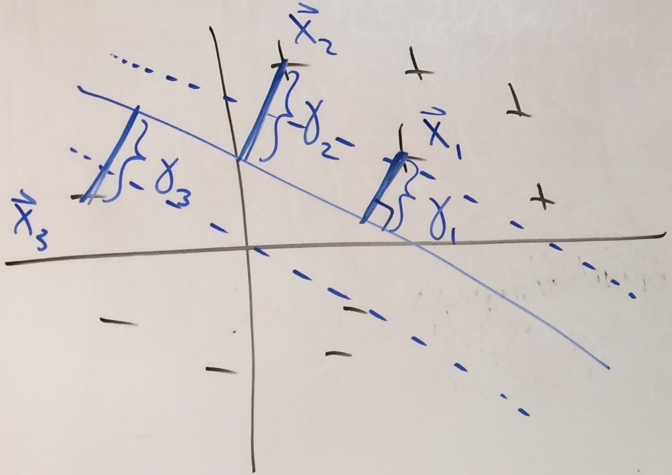
functional margin $\boxed{\hat{\gamma}_i = y_i(\vec{w}\cdot\vec{x}_i + b)}$

if correct: $\hat{\gamma}_i > 0$ ☆

if incorrect: $\hat{\gamma}_i < 0$ ✗

bad: increase magnitude of $\vec{w}$ & and b to increase $\hat{\gamma}_i$

good: arbitrary constraint on $\vec{w}$ & b.



Idea: want to maximize $\boxed{\hat{\gamma} = \min_{i=1\cdots n} \hat{\gamma}_i}$

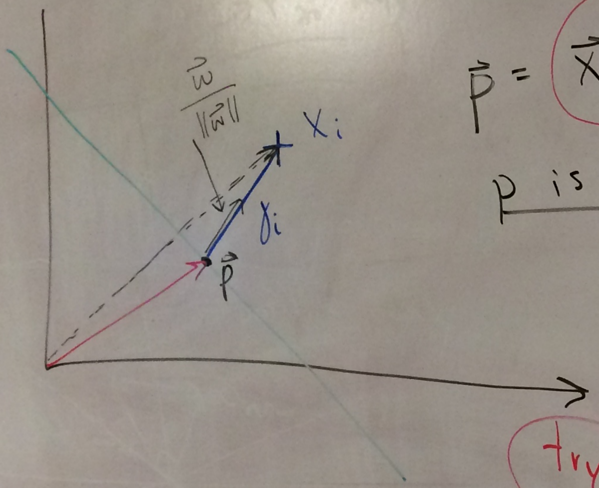
overall functional margin

Geometric margin → $\gamma_i = ?$

distance between pt & hyperplane

$$\boxed{\vec{p} + \gamma_i y_i \frac{\vec{w}}{\|\vec{w}\|} = \vec{x}_i}$$

unit vector



$$\vec{p} = \vec{x}_i - \gamma_i y_i \frac{\vec{w}}{\|\vec{w}\|}$$

p is on the hyperplane

$$0 = \vec{w} \cdot \vec{p} + b$$

plug in $\vec{p}$ &
solve for γ_i

exercise!

$$\boxed{\gamma_i = y_i \left(\frac{\vec{w}}{\|\vec{w}\|} \cdot \vec{x}_i + \frac{b}{\|\vec{w}\|} \right)}$$

try to maximize!!

Functional and Geometric Margins

SVM classifier:

(same as perceptron)

$$h(\vec{x}) = \operatorname{sign}\big(\vec{w} \cdot \vec{x} + b\big)$$

Functional and Geometric Margins

SVM classifier:

(same as perceptron)

$$h(\vec{x}) = \text{sign}\big(\vec{w} \cdot \vec{x} + b\big)$$

Functional Margin:

$$\hat{\gamma}_i = y_i(\vec{w} \cdot \vec{x}_i + b)$$

Functional and Geometric Margins

SVM classifier:

(same as perceptron)

$$h(\vec{x}) = \mathrm{sign}\left(\vec{w} \cdot \vec{x} + b\right)$$

Functional Margin:

$$\hat{\gamma}_i = y_i(\vec{w} \cdot \vec{x}_i + b)$$

Geometric Margin:

(distance between example and hyperplane)

$$\gamma_i = y_i \left(\frac{\vec{w}}{||\vec{w}||} \cdot \vec{x}_i + \frac{b}{||\vec{w}||} \right)$$

Functional and Geometric Margins

SVM classifier:
(same as perceptron)

$$h(\vec{x}) = \operatorname{sign}\left(\vec{w} \cdot \vec{x} + b\right)$$

Functional Margin:

$$\hat{\gamma}_i = y_i(\vec{w} \cdot \vec{x}_i + b)$$

Geometric Margin:
(distance between
example and hyperplane)

$$\gamma_i = y_i \left(\frac{\vec{w}}{||\vec{w}||} \cdot \vec{x}_i + \frac{b}{||\vec{w}||} \right)$$

Note:

$$\boxed{\gamma_i = \frac{\hat{\gamma}_i}{||\vec{w}||}}$$

Optimization Problem: try 1

Goal: maximize the minimum distance between example and hyperplane

$$\gamma = \min_{i=1,\cdots,n} \gamma_i$$

Optimization Problem: try 1

Goal: maximize the minimum distance between example and hyperplane

$$\gamma = \min_{i=1,\cdots,n} \gamma_i$$

Formulation: optimize a function with respect to a constraint

$$\begin{aligned} \max_{\gamma,\vec{w},b} \quad & \gamma \\ \text{s.t.} \quad & y_i(\vec{w} \cdot \vec{x}_i + b) \geq \gamma, \quad i = 1, \cdots, n \\ \text{and} \quad & \|\vec{w}\| = 1 \end{aligned}$$

(force functional and geometric margin to be equal)

Optimization Problem: try 2

Idea: substitute functional margin
divided by magnitude of weight vector

$$\max_{\hat{\gamma},\vec{w},b} \quad \frac{\hat{\gamma}}{\|\vec{w}\|}$$

$$\text{s.t.} \quad y_i(\vec{w}\cdot\vec{x}_i + b) \geq \hat{\gamma}, \quad i = 1,\cdots,n$$

(gets rid of non-convex constraint)

Optimization Problem: try 3

Idea: put arbitrary constraint on functional margin

$$\boxed{\hat{\gamma} = 1}$$

$$\begin{aligned} \min_{\vec{w},b} \quad & \frac{1}{2}\|\vec{w}\|^2 \\ \text{s.t.} \quad & y_i(\vec{w}\cdot\vec{x}_i + b) \geq 1, \quad i = 1,\cdots,n \end{aligned}$$

Optimization Problem: try 3

Idea: put arbitrary constraint on functional margin

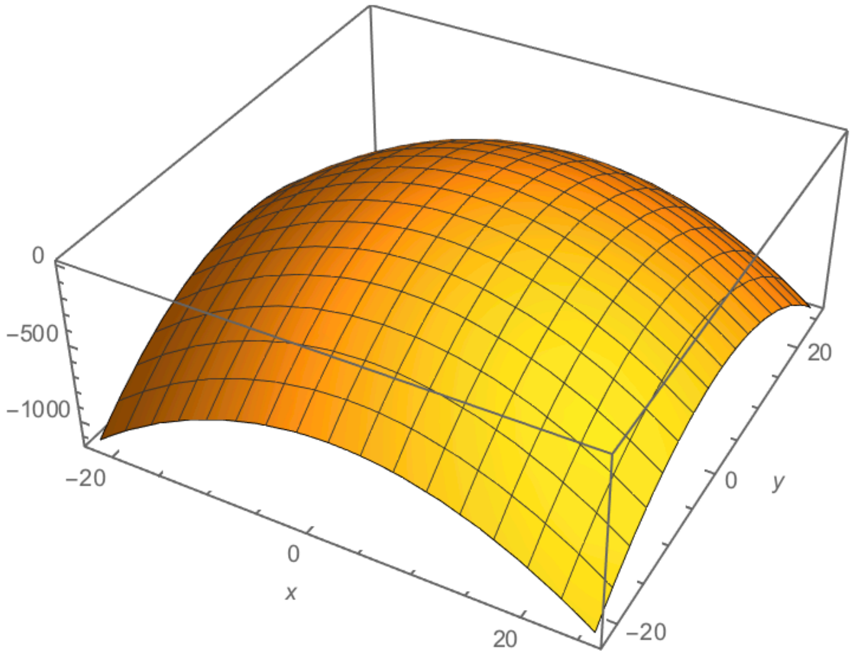
$$\hat{\gamma} = 1$$

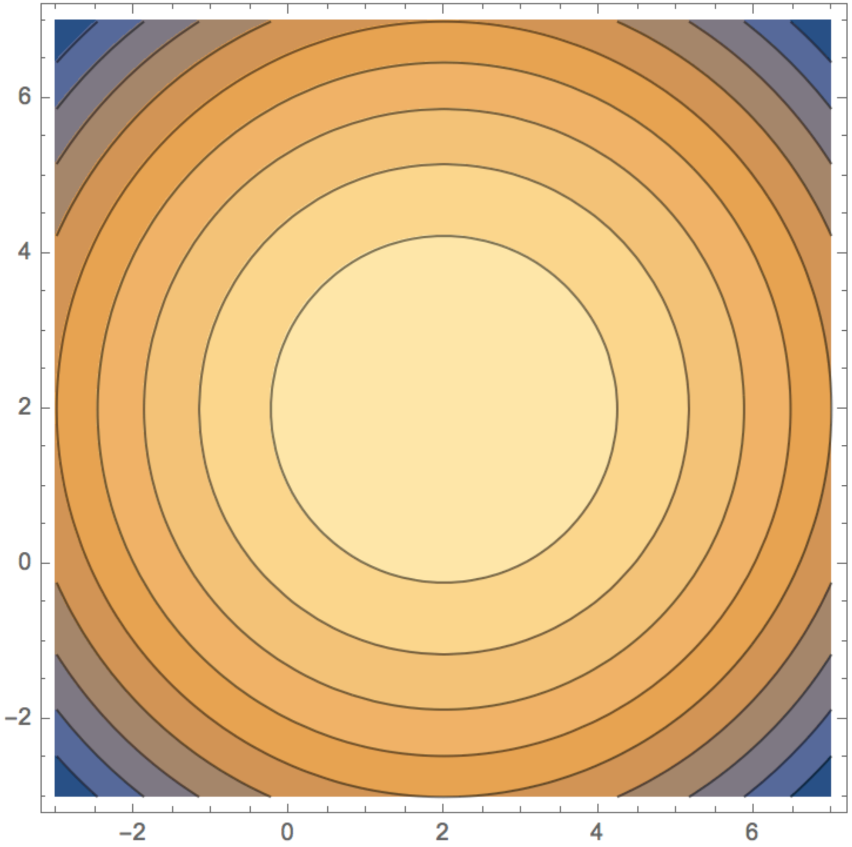
$$\begin{aligned} \min_{\vec{w},b} \quad & \frac{1}{2}\|\vec{w}\|^2 \\ \text{s.t.} \quad & y_i(\vec{w}\cdot\vec{x}_i + b) \geq 1, \quad i = 1, \cdots, n \end{aligned}$$

$$\begin{aligned} \min_{\vec{w},b} \quad & \frac{1}{2}\|\vec{w}\|^2 \\ \text{s.t.} \quad & -y_i(\vec{w}\cdot\vec{x}_i + b) + 1 \leq 0, \quad i = 1, \cdots, n \end{aligned}$$

Lagrange multipliers example 1

$$f(x,y) = 5 - (x-2)^2 - (y-2)^2$$





Contour plot of *f*(*x*,*y*)

$$\text{maximize}_{x,y} \quad f(x,y)$$

$$s.t. \quad g(x,y) = 0$$

$$g(x,y) = -5 + x + y$$

Detour to Lagrange Multipliers

Goal: * maximize function subject to constraint

$$\underset{x,y}{\text{maximize}} \quad f(x,y)$$

$$\text{s.t.} \quad g(x,y) = 0$$

no constraint $\Rightarrow x^* = 2$, $y^* = 2$

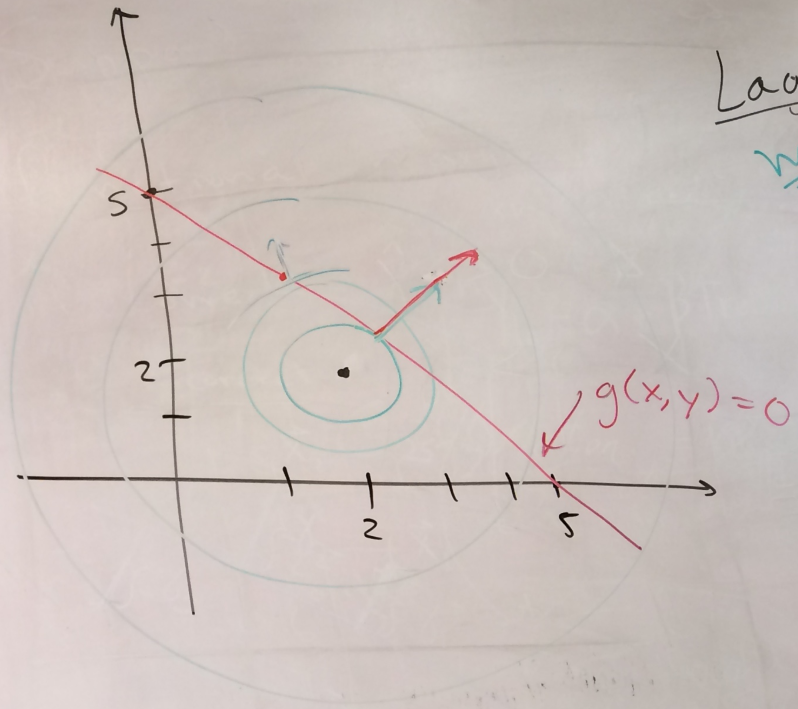
example: $\overset{\text{max}}{f(x,y)} = 5 - (x-2)^2 - (y-2)^2$

$$\text{s.t.} \quad g(x,y) = -5 + x + y$$

$$\boxed{g(x,y) = 0}$$

$$-5 + x + y = 0$$

$$\Rightarrow \boxed{y = -x + 5}$$



Lagrangian

could also add.

max $h(x,y,\lambda) = f(x,y) - \lambda \cdot g(x,y)$

Lagrange multiplier

Lagrange multiplier

$\nabla h(x,y,\lambda) = 0 \}$ want!

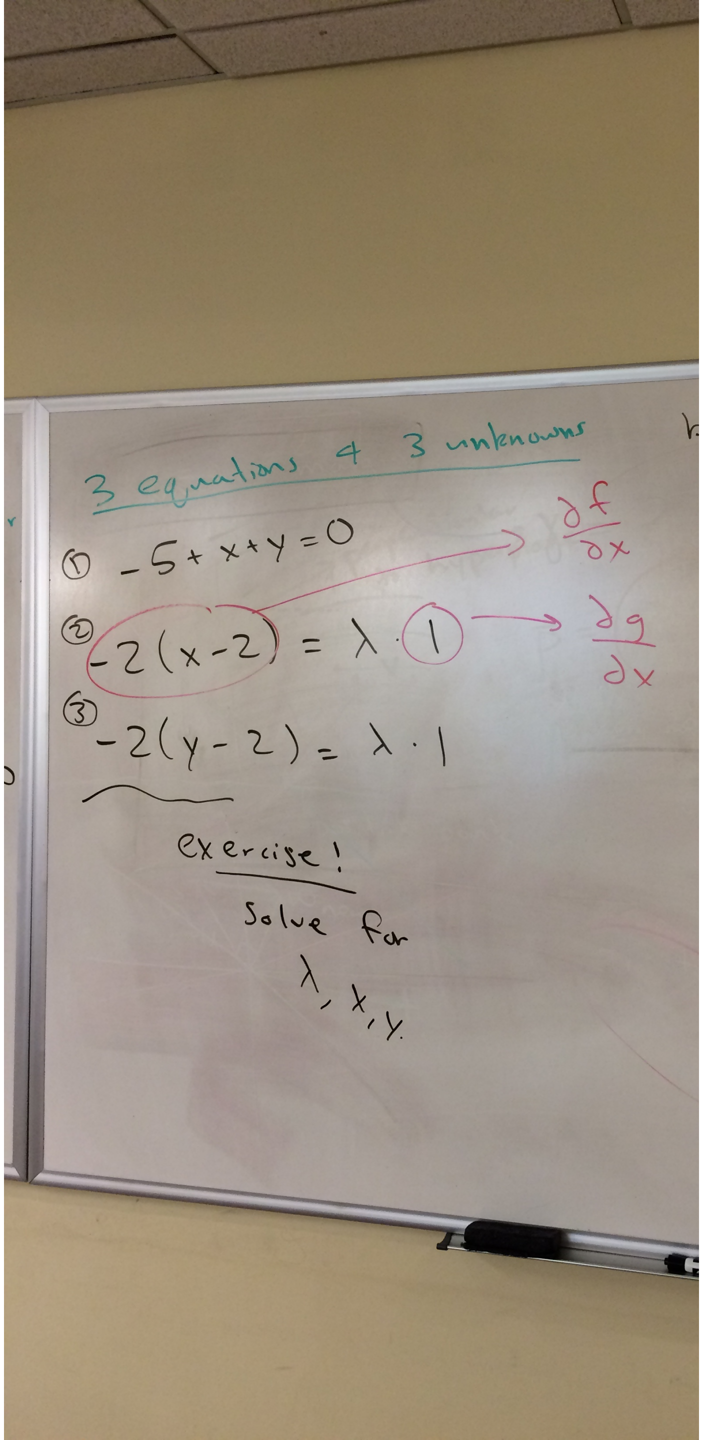
$\nabla_{x,y} h(x,y,\lambda) = \nabla f(x,y) - \lambda \nabla g(x,y) = 0$

$\frac{\partial h}{\partial \lambda} = \boxed{g(x,y) = 0}$ 1 equation

constraint $\Rightarrow \boxed{\nabla f(x,y) = \lambda \nabla g(x,y)}$

2 equations

①
②
③



3 equations & 3 unknowns

① $-5+x+y=0$

② $\underbrace{-2(x-2)}_{\frac{\partial f}{\partial x}} = \lambda \cdot \underbrace{1}_{\frac{\partial g}{\partial x}}$

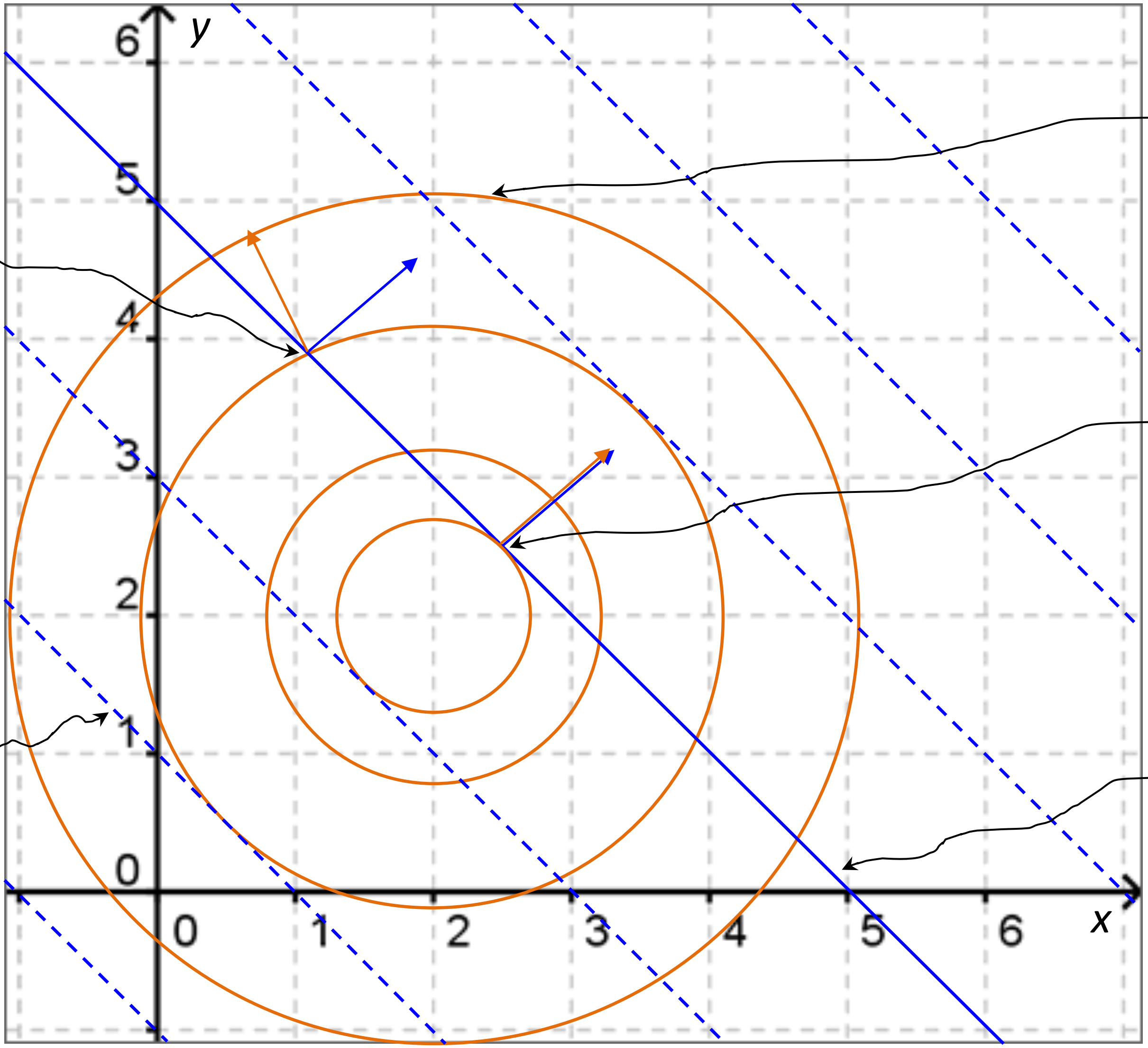
③ $-2(y-2) = \lambda \cdot 1$

exercise!

Solve for

$\lambda, x, y.$

Lagrange multipliers example 1



Normals (and derivatives) **not** parallel

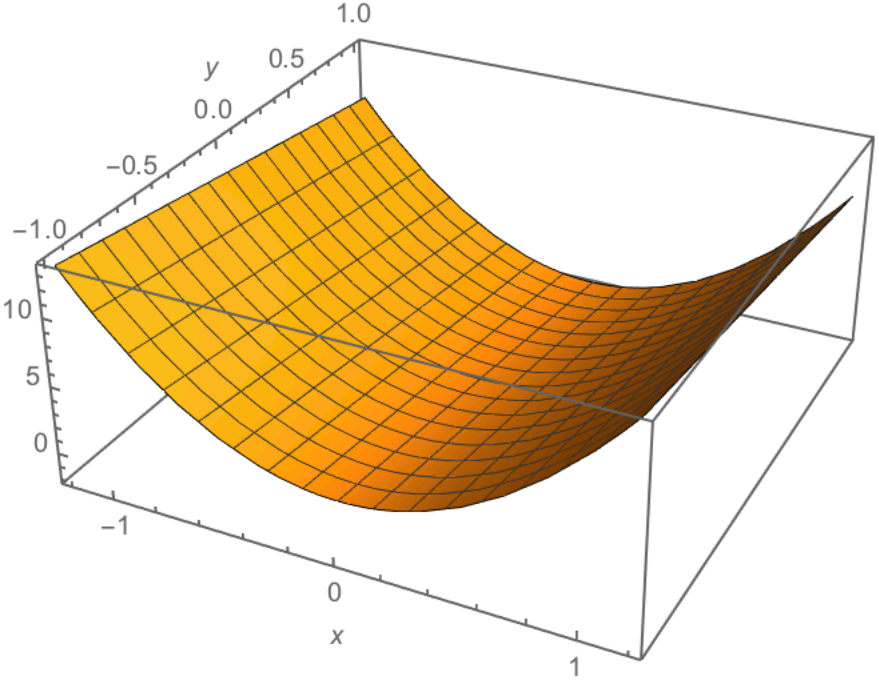
level curves of $g(x,y)$

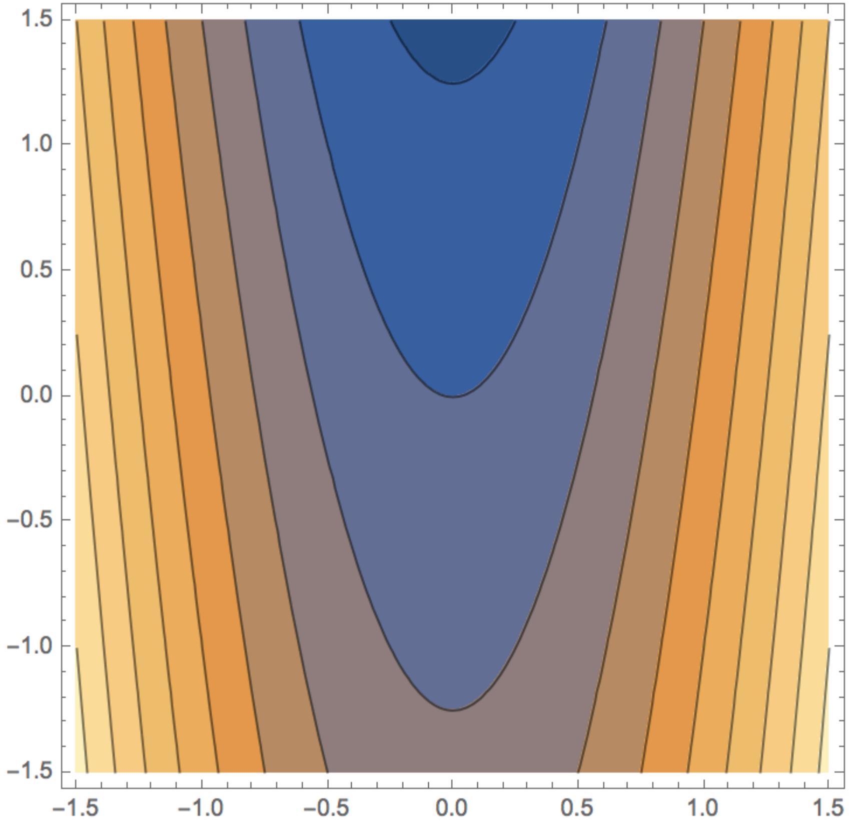
level curves of $f(x,y)$

<u>SOLUTION</u>:
Normals (and derivatives) parallel
$x = 2.5$
$y = 2.5$

$g(x,y) = 0$

Lagrange multipliers example 2



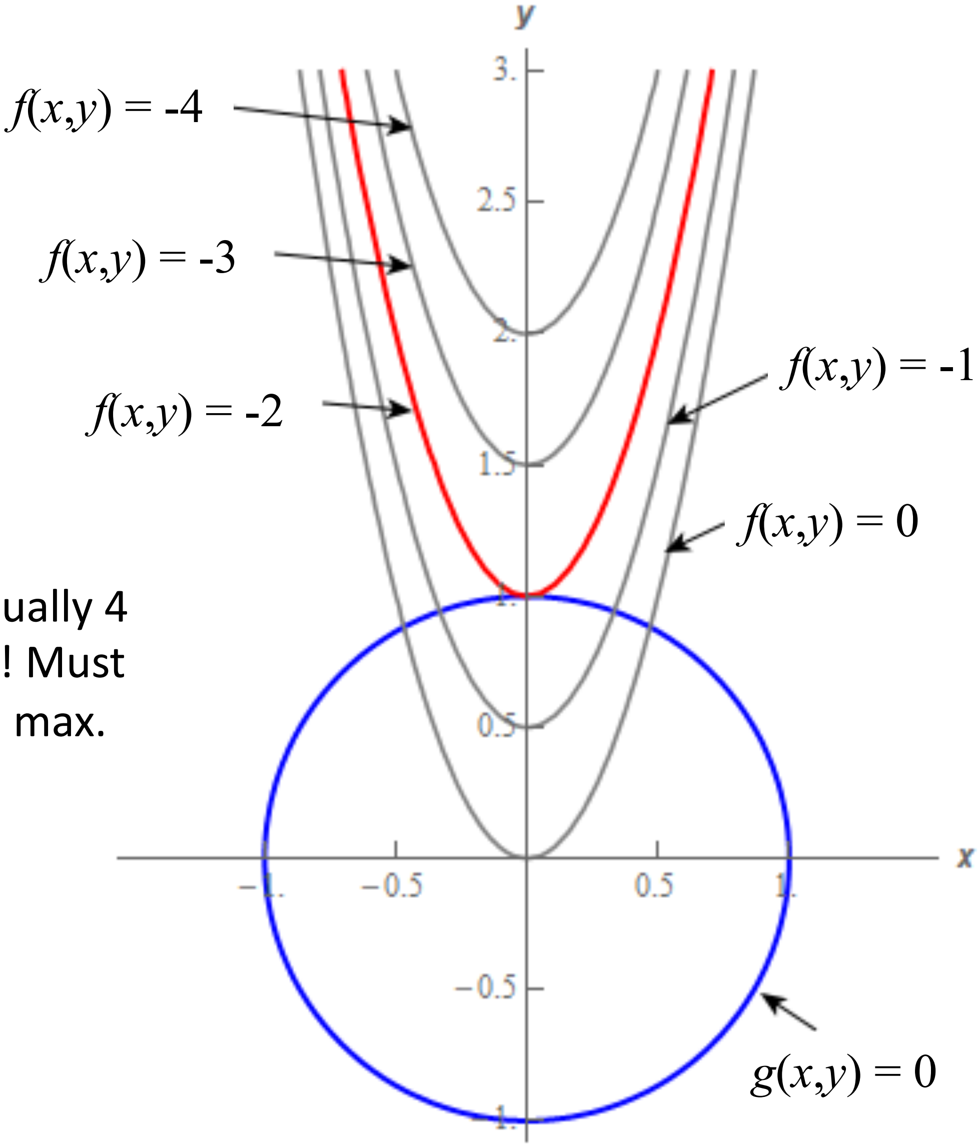


Contour plot of f(x,y)

$$\text{maximize}_{x,y} \quad f(x,y)$$

$$s.t. \quad g(x,y) = 0$$

Lagrange multipliers example 2



Note: there are actually 4 potential solutions! Must plug in to *f* to find max.