CS 360: Machine Learning

Prof. Sara Mathieson

Fall 2020

Admin

- Office hours next week:
  - Monday 9:45-11am
  - **Monday** 4:30-6pm

- Lab 5 due **October 20**

Outline for October 16

- Maximum Likelihood Estimation (MLE) in other fields
- Recap handouts and logistic regression so far
- Regularization
- Multi-class logistic regression

Outline for October 16

- Maximum Likelihood Estimation (MLE) in other fields
- Recap handouts and logistic regression so far
- Regularization
- Multi-class logistic regression

MLE in other fields

1. Chemistry

Estimating the number of pure chemical components in a mixture by maximum likelihood

E. Levina[1*], A.S. Wagaman[1], A.F. Callender[2], G.S. Mandair[2] and M.D. Morris[2]

[1]Department of Statistics, The University of Michigan, Ann Arbor, MI 48109, USA
[2]Department of Chemistry, The University of Michigan, Ann Arbor, MI 48109, USA

Received 31 August 2006; Revised 4 December 2006; Accepted 6 December 2006

2. Physics

Maximum Likelihood Blood Velocity Estimator Incorporating Properties of Flow Physics

Malene Schlaikjer and Jørgen Arendt Jensen, *Senior Member, IEEE*

3. Biology

Maximum-Likelihood Estimation of Admixture Proportions From Genetic Data

Jinliang Wang

GENETICS *June 1, 2003 vol. 164 no. 2 747-765*

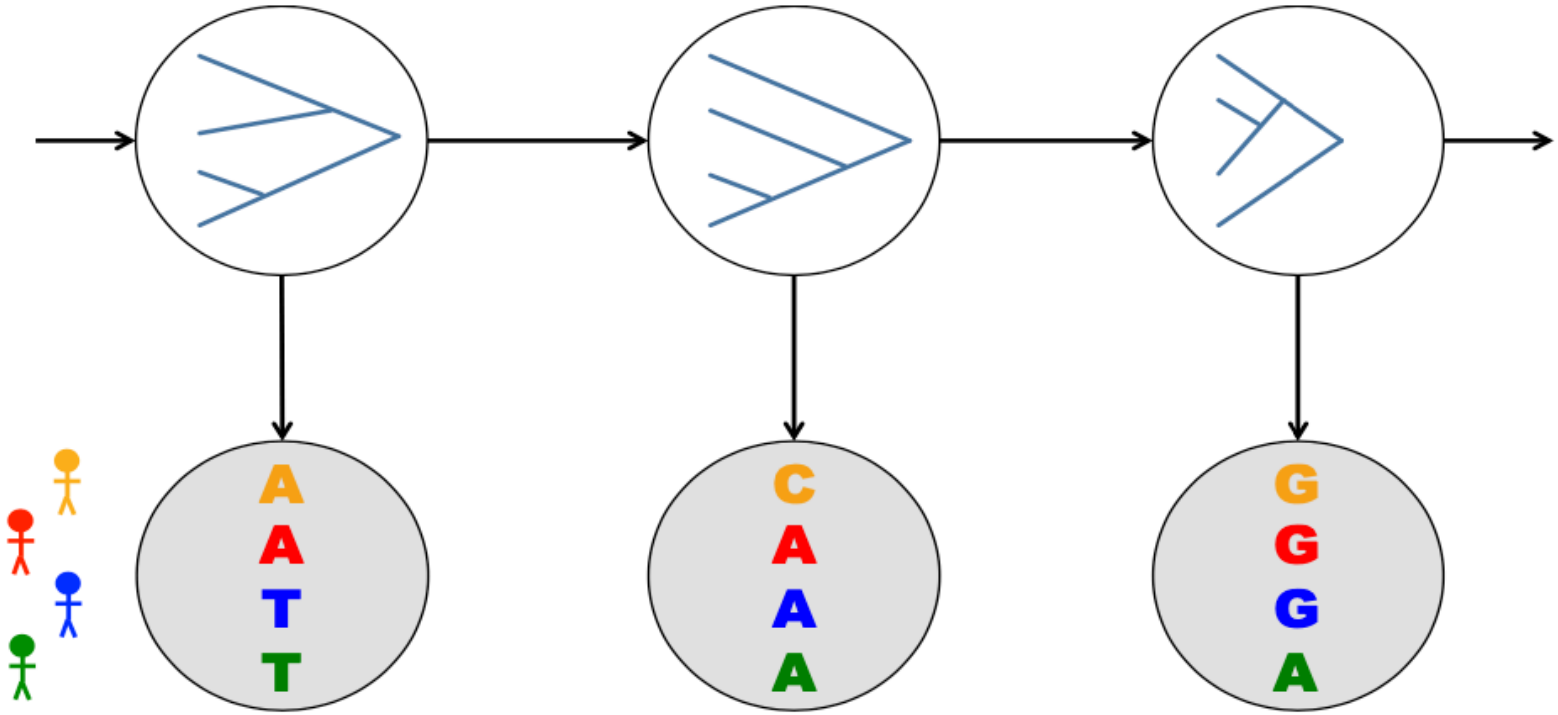
4. Economics

Maximum likelihood estimation of mixed C-vines with application to exchange rates

Claudia Czado[1], Ulf Schepsmeier[1] and Aleksey Min[1]
[1]Zentrum Mathematik, Technische Universität München, Boltzmannstr. 3, 85747 Garching bei München, Germany

MLE in my own research

- Known: DNA data from multiple individuals
- Unknown: tree of relationships between the individuals



Outline for October 16

- Maximum Likelihood Estimation (MLE) in other fields
- Recap handouts and logistic regression so far
- Regularization
- Multi-class logistic regression

Logistic Regression Warmup

1. The output of logistic regression is a model that creates:

   (a) a linear decision boundary

   (b) a logistic decision boundary

   (c) no decision boundary

2. We use logistic regression for:

   (a) classification

   (b) regression

   (c) both

3. Our hypothesis in logistic regression is:

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = p(y = 1|\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w}\cdot\boldsymbol{x}}}$$

   If $\boldsymbol{w}$ is the zero vector (as it would be when starting SGD), what is the probability $y = 1$?

4. How did we define the cost function for logistic regression? (Bonus: write down the cost function)

   (a) likelihood

   (b) log likelihood

   (c) negative log likelihood

Logistic Regression Warmup

1. The output of logistic regression is a model that creates:

   (a) a linear decision boundary

   (b) a logistic decision boundary

   (c) no decision boundary

Logistic Regression Warmup

1. The output of logistic regression is a model that creates:

   (a) a linear decision boundary

   (b) a logistic decision boundary

   (c) no decision boundary

2. We use logistic regression for:

   (a) classification

   (b) regression

   (c) both

Logistic Regression Warmup

1. The output of logistic regression is a model that creates:

   (a) a linear decision boundary

   (b) a logistic decision boundary

   (c) no decision boundary

2. We use logistic regression for:

   (a) classification

   (b) regression

   (c) both

3. Our hypothesis in logistic regression is:

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = p(y = 1|\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w}\cdot\boldsymbol{x}}}$$

   If $\boldsymbol{w}$ is the zero vector (as it would be when starting SGD), what is the probability $y = 1$?

Logistic Regression Warmup

1. The output of logistic regression is a model that creates:

   (a) a linear decision boundary

   (b) a logistic decision boundary

   (c) no decision boundary

2. We use logistic regression for:

   (a) classification

   (b) regression

   (c) both

3. Our hypothesis in logistic regression is:

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = p(y = 1|\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w}\cdot\boldsymbol{x}}}$$

   If $\boldsymbol{w}$ is the zero vector (as it would be when starting SGD), what is the probability $y = 1$?

   ½

4. How did we define the cost function for logistic regression? (Bonus: write down the cost function)

   (a) likelihood

   (b) log likelihood

   (c) negative log likelihood

Logistic Regression Warmup

1. The output of logistic regression is a model that creates:

   (a) a linear decision boundary

   (b) a logistic decision boundary

   (c) no decision boundary

2. We use logistic regression for:

   (a) classification

   (b) regression

   (c) both

3. Our hypothesis in logistic regression is:

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = p(y = 1|\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w}\cdot\boldsymbol{x}}}$$

   If $\boldsymbol{w}$ is the zero vector (as it would be when starting SGD), what is the probability $y = 1$?

   ½

4. How did we define the cost function for logistic regression? (Bonus: write down the cost function)

   (a) likelihood

   (b) log likelihood

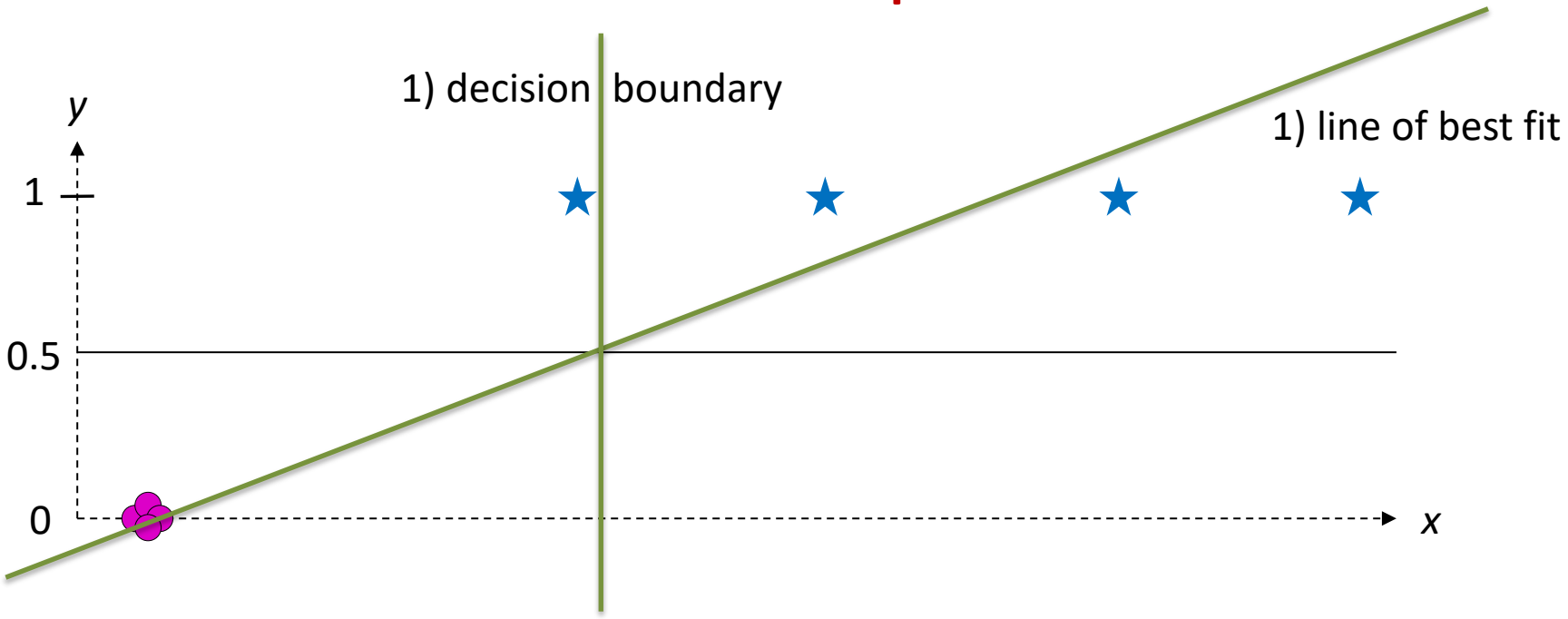
   (c) negative log likelihood

Extra Example

1) What line of best fit would be produced by linear regression? (roughly)

2) What linear decision boundary would be produced by logistic regression? (roughly)

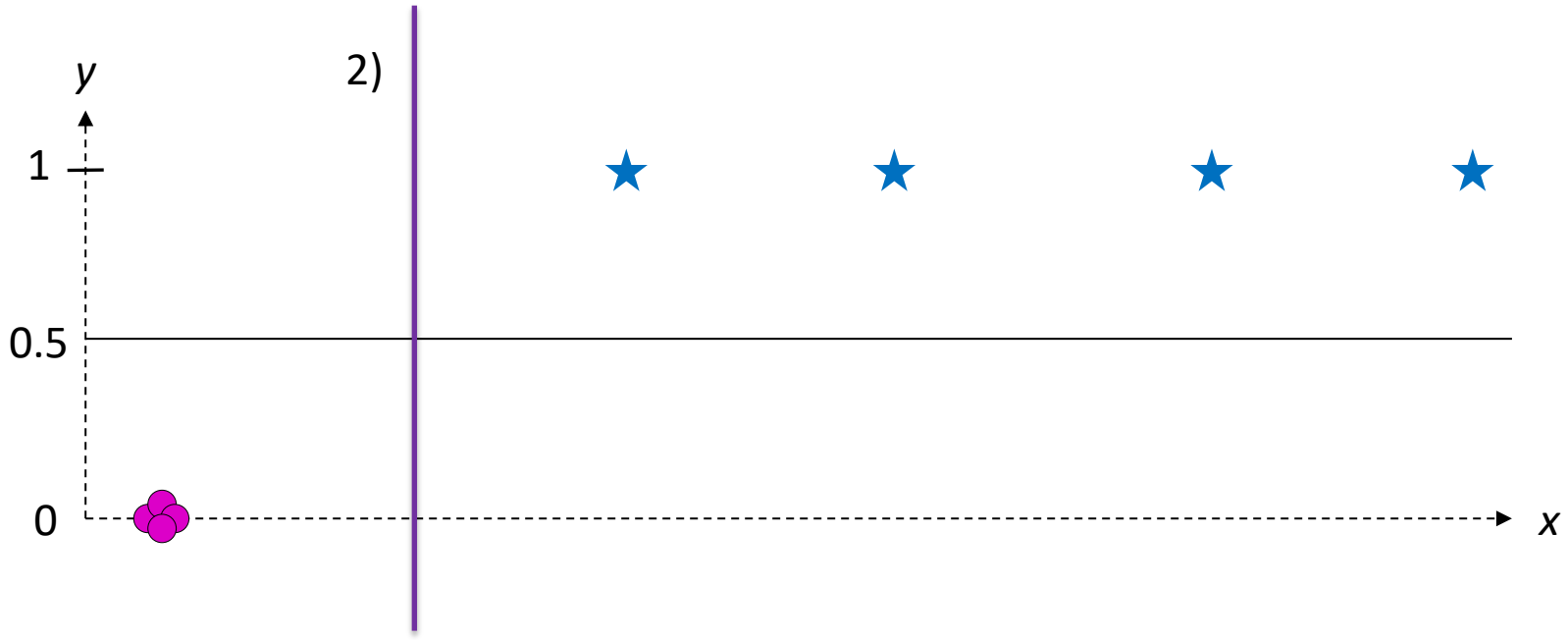
Extra Example



1) What line of best fit would be produced by linear regression? (roughly)

2) What linear decision boundary would be produced by logistic regression? (roughly)

Extra Example



1) What line of best fit would be produced by linear regression? (roughly)

2) What linear decision boundary would be produced by logistic regression? (roughly)

Stochastic Gradient Descent for Logistic Regression (binary classification)

set w = 0 vector

while cost J(w) still changing:

    shuffle data points

    for i = 1…n:

        w <- w – alpha(derivative of J(w) wrt x_i)

    store J(w)

3 important pieces to SGD

- Hypothesis function (prediction)

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = p(y = 1|\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w}\cdot\boldsymbol{x}}}$$

3 important pieces to SGD

- Hypothesis function (prediction)

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = p(y = 1|\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w}\cdot\boldsymbol{x}}}$$

- Cost function (want to minimize)

$$J(\boldsymbol{w}) = -\sum_{i=1}^{n} y_i \log h_{\boldsymbol{w}}(\boldsymbol{x_i}) + (1 - y_i) \log(1 - h_{\boldsymbol{w}}(\boldsymbol{x_i}))$$

3 important pieces to SGD

- Hypothesis function (prediction)

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = p(y = 1|\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w}\cdot\boldsymbol{x}}}$$

- Cost function (want to minimize)

$$J(\boldsymbol{w}) = -\sum_{i=1}^{n} y_i \log h_{\boldsymbol{w}}(\boldsymbol{x_i}) + (1 - y_i) \log(1 - h_{\boldsymbol{w}}(\boldsymbol{x_i}))$$

- Gradient of cost wrt single data point x_i

$$\nabla J_{\boldsymbol{x}_i}(\boldsymbol{w}) = (h_{\boldsymbol{w}}(\boldsymbol{x_i}) - y_i)\boldsymbol{x_i}$$

3 important pieces to SGD

- Hypothesis function (prediction)

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = p(y = 1|\boldsymbol{x}) = \frac{1}{1 + e^{-\vec{\boldsymbol{w}} \cdot \vec{\boldsymbol{x}}}}$$

- Cost function (want to minimize)

$$J(\boldsymbol{w}) = -\sum_{i=1}^{n} y_i \log h_{\boldsymbol{w}}(\boldsymbol{x_i}) + (1 - y_i) \log(1 - h_{\boldsymbol{w}}(\boldsymbol{x_i}))$$

- Gradient of cost wrt single data point x_i

$$\nabla J_{\boldsymbol{x}_i}(\boldsymbol{w}) = (h_{\boldsymbol{w}}(\boldsymbol{x_i}) - y_i)\boldsymbol{x_i}$$

don't threshold

0.7 0.9 1

p=1

Handout 9

1. Say I train a binary logistic regression model (i.e. outcomes $\in \{0, 1\}$) and end up with $\hat{\boldsymbol{w}} = [\hat{w}_0, \hat{w}_1]^T = [-4, -5]^T$. What is the decision boundary? Sketch a graph of this logistic model and label the decision boundary. How would you classify a new point $x_{\text{test}} = -2$?

$\vec{w} \cdot \vec{x} > 0 \Rightarrow$ predict 1

$-4 - 5x > 0$

$-5x > 4$

$x < -\frac{4}{5}$ $\Rightarrow$ predict 1

$\hat{y}_{test} = 1$

$\frac{1}{1+e^{-\vec{w}\cdot\vec{x}}} > \frac{1}{2}$

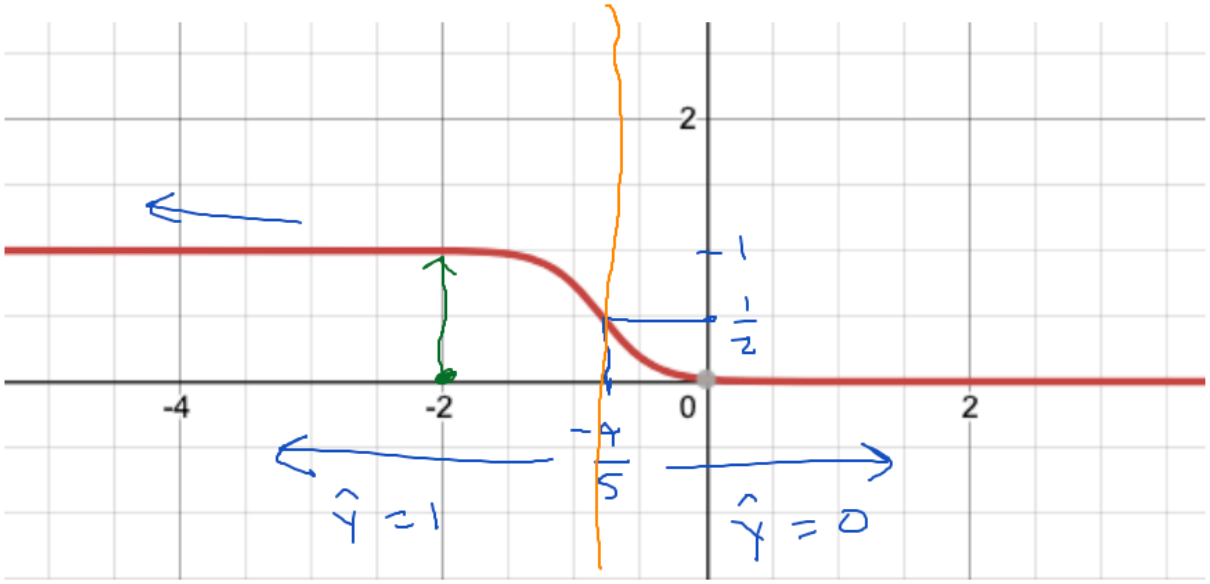
$\vec{w} \cdot \vec{x} > 0$

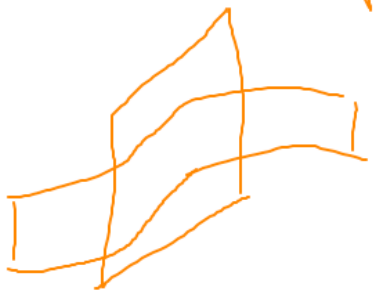
p=2

$\Rightarrow \begin{bmatrix} 5 \\ -3 \end{bmatrix}$

$w_0 + w_1 x_1 + w_2 x_2 > 0$

plane



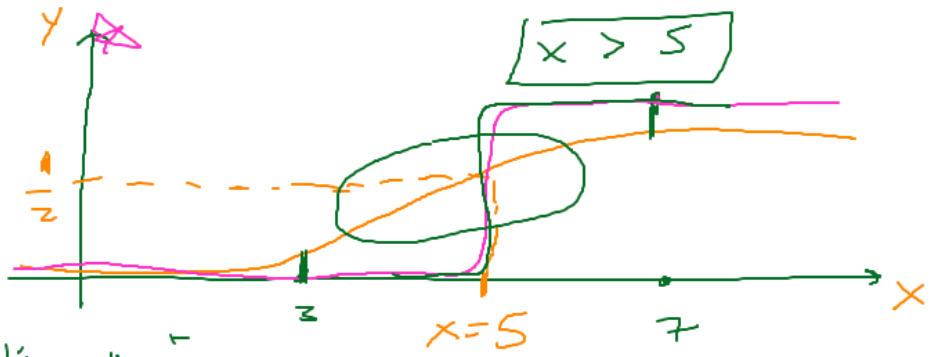


$$\frac{1}{\left(1+e^{-(-4-5x)}\right)}$$

Lab 5 intro

Say we have $p = 1$ and two training examples: $(x_1, y_1) = (3, 0)$ and $(x_2, y_2) = (7, 1)$, and we would like to fit a logistic model to this dataset.

1. Draw these two examples on a coordinate system and sketch a logistic function that would fit them (roughly). What is the optimal decision boundary? Does this help us uniquely determine $\hat{w}_0$ and $\hat{w}_1$?



$-10 + 2x > 0$

$x > 5$

$-5 + x > 0$

$w_0 = -5$

$w_1 = 1$

$\Rightarrow h_{\vec{w}}(\vec{x}) = \frac{1}{1 + e^{-(-5+x)}}$

better

$w_0 = -10$

$w_1 = 2$

$w_0 = -100$

$w_1 = 20$

lin reg: $(w \cdot x - y)^2$

$J(\vec{w})$

$\vec{w}$

as magnitude of weight ↑

$e^{\square} \to 0$

$\to \infty$

log reg: $\left(\frac{1}{1+e^{-wx}}\right)^y$

$J(\vec{w})$

family

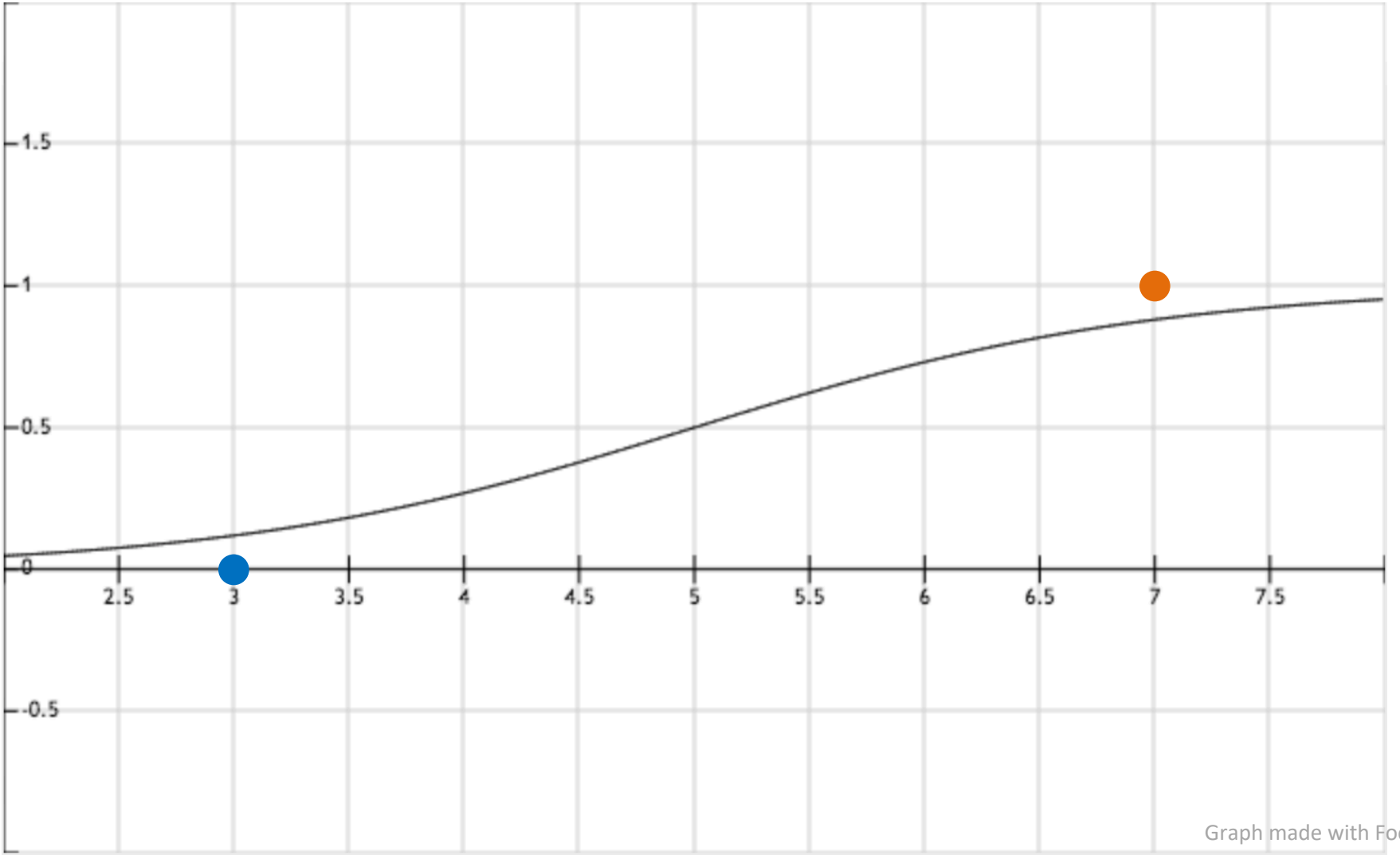
$w_0 = -5a$

$w_1 = a$

$\Rightarrow$ overly confident

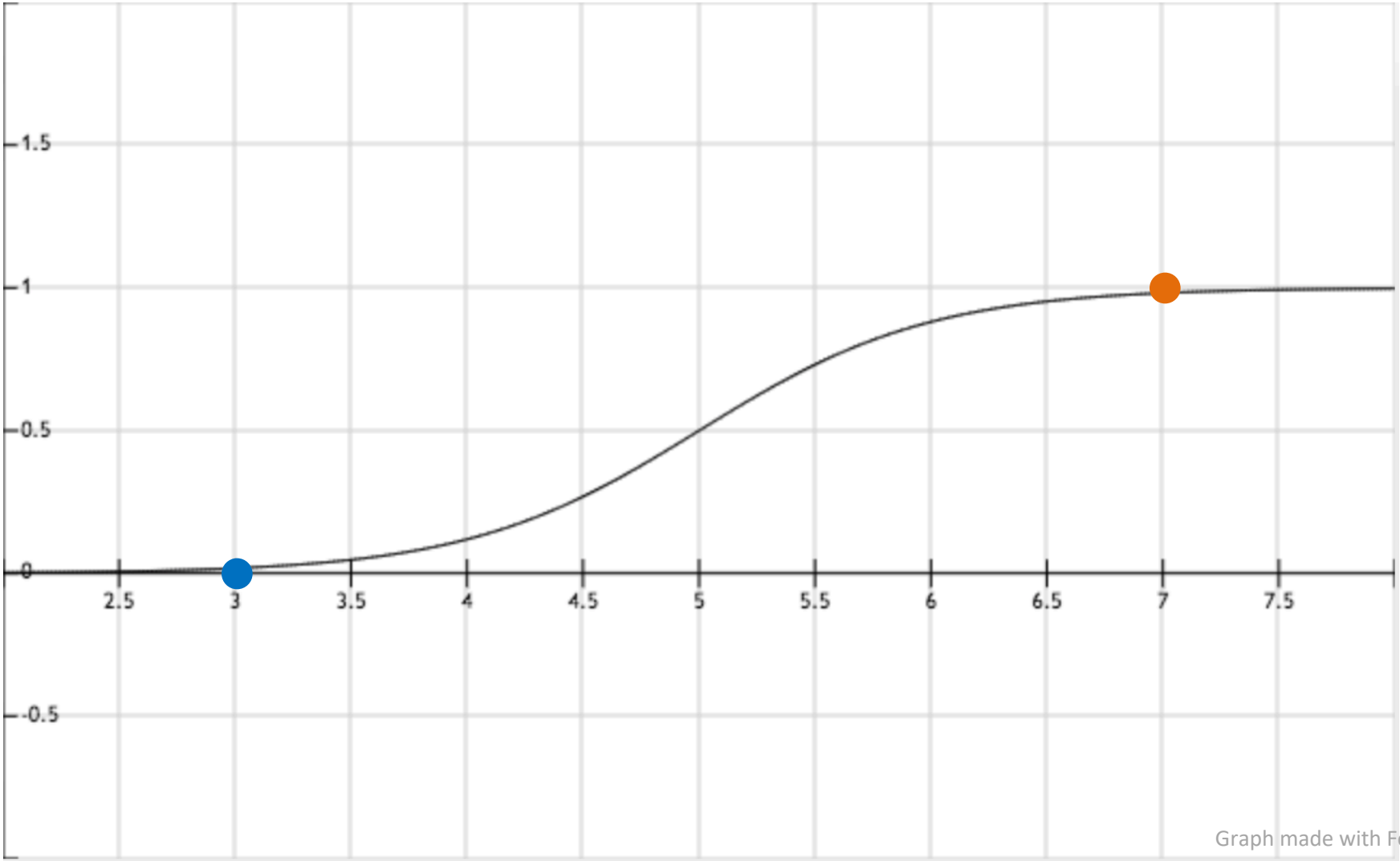
$\vec{w}$

w_0=-5, w_1=1 $h_w(x) = 1/(1+e^{(5-x)})$

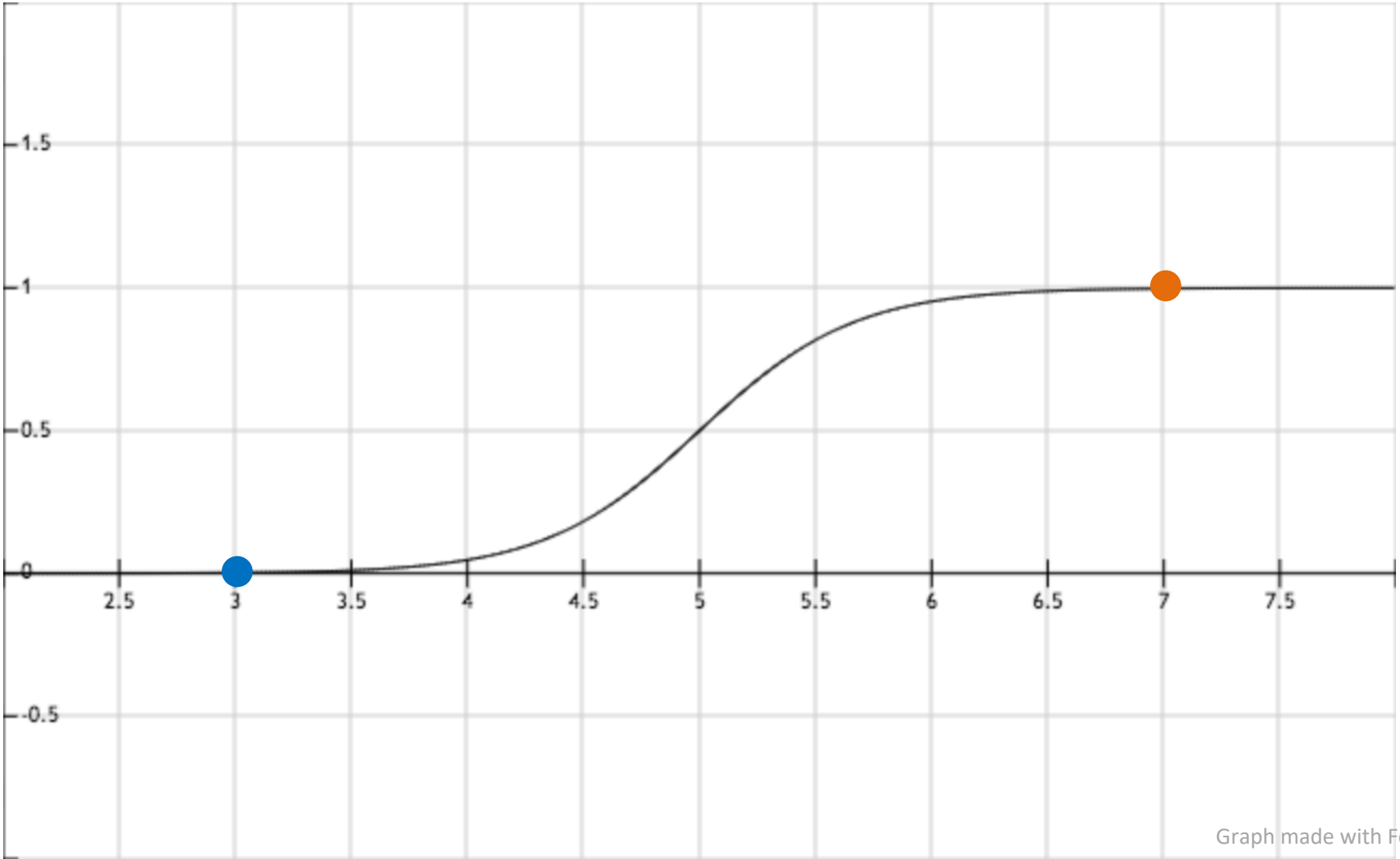


Graph made with FooPlot

w_0=-10, w_1=2 $h_w(x) = 1/(1+e^{(10-2x)})$

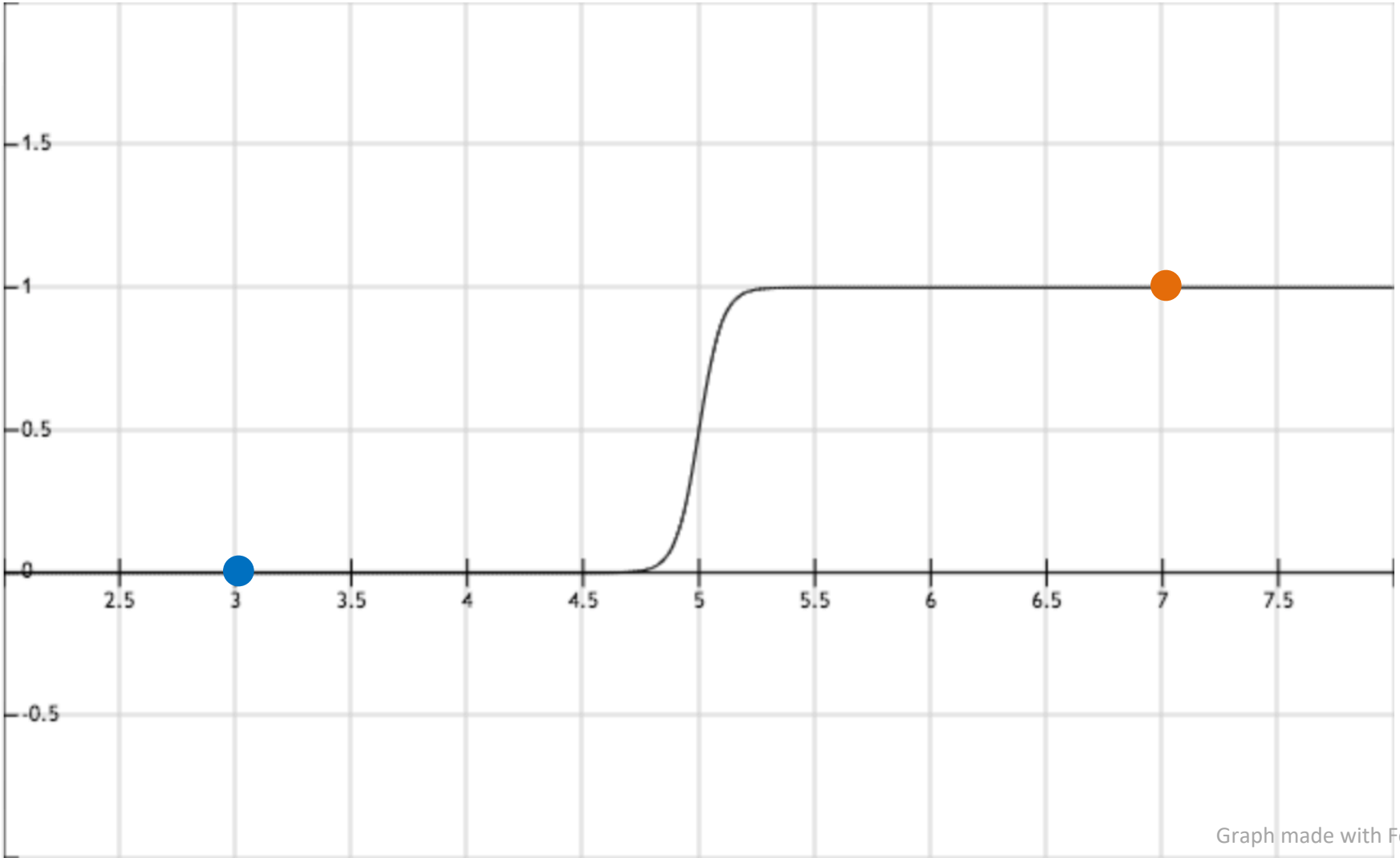


w_0=-15, w_1=3 $h_w(x) = 1/(1+e^{(15-3x)})$



Graph made with FooPlot

w_0=-100, w_1=20 $h_w(x)$ = 1/(1+e^(100-20x))



Lab 5 intro

2. Say in our SGD method, we choose to analyze (x_2, y_2) first. Before starting SGD, we set $w_0 = 0$ and $w_1 = 0$. After analyzing (x_2, y_2), what are w_0 and w_1? Choose $\alpha = 0.1$. Plot the decision boundary on your graph above.

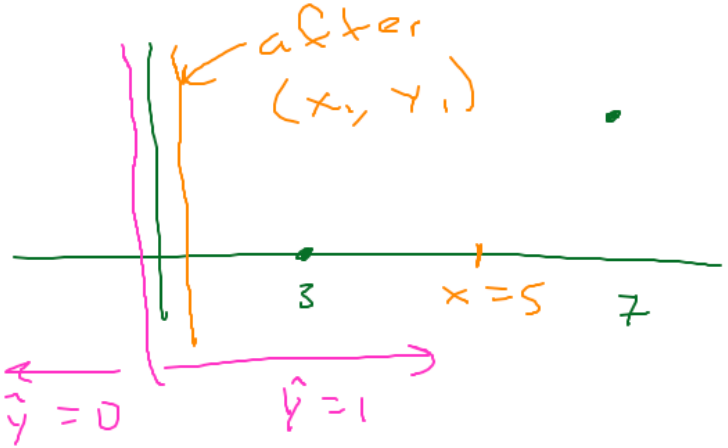
$$\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} \leftarrow \begin{bmatrix} 0 \\ 0 \end{bmatrix} - 0.1 \left(\frac{1}{1+e^{0}} - 1 \right) \begin{bmatrix} 1 \\ 7 \end{bmatrix}$$

$$\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \begin{bmatrix} 0.05 \\ 0.35 \end{bmatrix}$$

$$0.05 + 0.35x > 0$$

$$x > -\frac{1}{7}$$

$(x_1, y_1) = (3, 0)$



3.

$$\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} \leftarrow \begin{bmatrix} 0.05 \\ 0.35 \end{bmatrix} - 0.1 \left[\frac{1}{1+e^{-(0.05+0.35(3))}} - 0 \right] \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$

$$\approx \begin{bmatrix} -0.025 \\ 0.125 \end{bmatrix}$$

$$x > \frac{1}{6}$$

$$\vec{w} \quad [0.05 \;\; 0.35] \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$

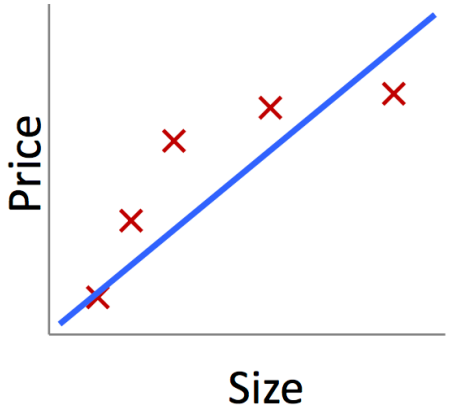
Outline for October 16

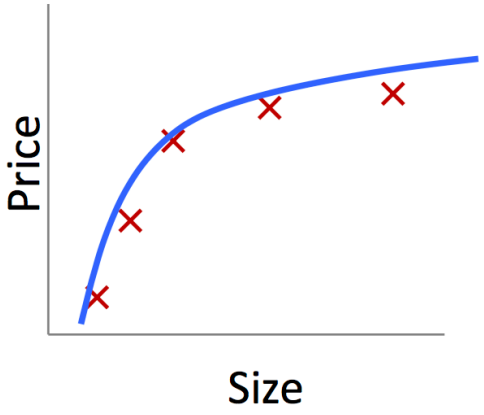
- Maximum Likelihood Estimation (MLE) in other fields
- Recap handouts and logistic regression so far
- Regularization
- Multi-class logistic regression

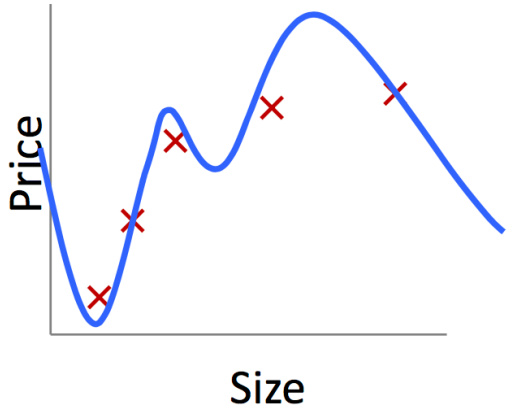
Adapted from slide by Jessica Wu
[example by Andrew Ng]

Generalization error

- Example: price vs. size (i.e. of a house or car)



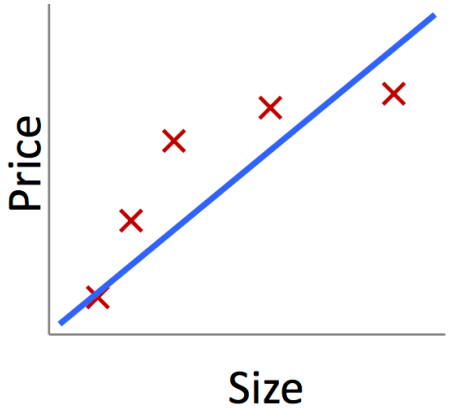




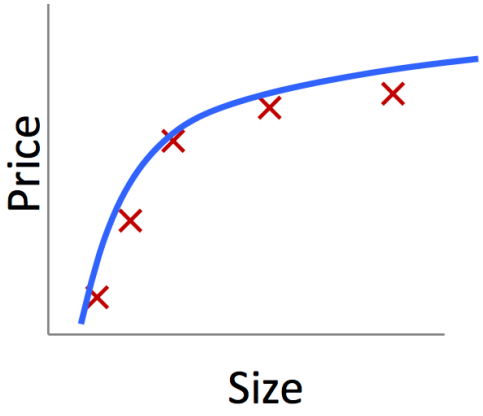
Adapted from slide by Jessica Wu
[example by Andrew Ng]

Generalization error

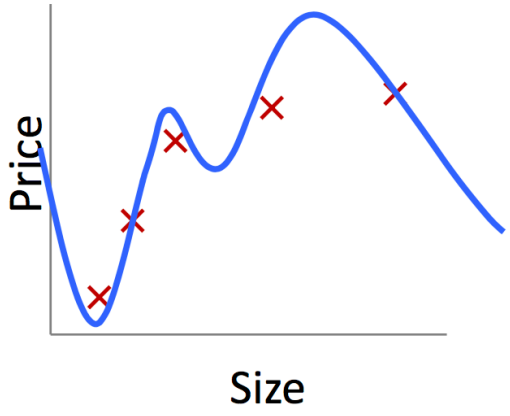
- Example: price vs. size (i.e. of a house or car)



underfitting
(high bias)



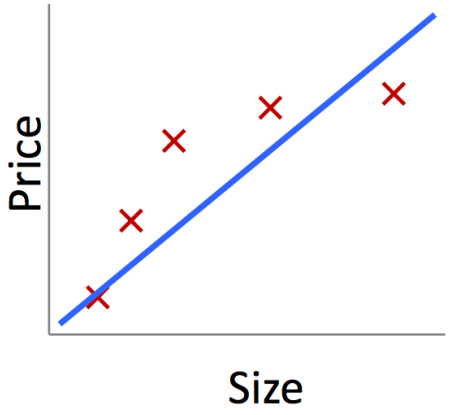
correct fit

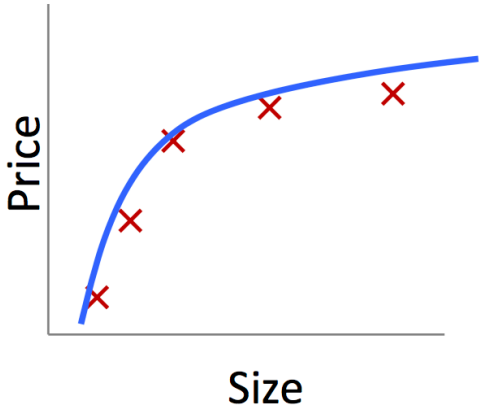


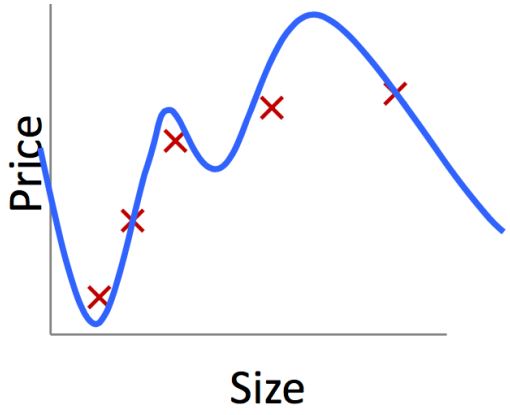
overfitting
(high variance)

Generalization error

- Example: price vs. size (i.e. of a house or car)







underfitting
(high bias)

correct fit

overfitting
(high variance)

<u>Structural error:</u>
Hypothesis space cannot model true relationship

-More data doesn't help
-Need a more flexible model

balance ⟺

<u>Estimation (approximation) error:</u>
Hypothesis space *can* model true relationship, BUT hard to identify correct model due to large hypothesis space, small *n*, or noise
-Reduce hypothesis space
-Add more data

Regularization

What if ...

- we have a limited # of training examples ($n<p$), or
- we want to automatically control the complexity of the learned hypothesis?

Regularization

What if ...

- we have a limited # of training examples ($n<p$), or
- we want to automatically control the complexity of the learned hypothesis?

Idea: penalize large values of w_j

Why prefer small weights?

- if large weights, small change in feature can result in large change in prediction
- prevent giving too much weight to any one feature
- might prefer zero weight for useless features

Common Regularizers

$$||\vec{w}||_0 = \sum_{j:w_j \neq 0} 1$$

L_0 norm

- Number of non-zero entries
- Minimizing L_0 norm is NP hard

Common Regularizers

$$||\vec{w}||_0 = \sum_{j:w_j \neq 0} 1$$

L_0 norm

- Number of non-zero entries
- Minimizing L_0 norm is NP hard

$$||\vec{w}||_1 = \sum_{j=1}^{p} |w_j|$$

L_1 norm

- Sum of magnitude of weights
- Not differentiable

Common Regularizers

$$||\vec{w}||_0 = \sum_{j:w_j \neq 0} 1$$

L_0 norm

- Number of non-zero entries
- Minimizing L_0 norm is NP hard

$$||\vec{w}||_1 = \sum_{j=1}^{p} |w_j|$$

L_1 norm

- Sum of magnitude of weights
- Not differentiable

$$||\vec{w}||_2 = \sqrt{\sum_{j=1}^{p} w_j^2}$$

L_2 norm

- Sum of squared weights
- Differentiable

Slide adapted from Jessica Wu

Adding Regularization to Logistic Regression

$$\min_{\vec{w}} J^R(\vec{w}) = -\left[\sum_{i=1}^{n} \underbrace{y_i \log h(\vec{x}_i)}_{y_i = 1} + \underbrace{(1 - y_i) \log(1 - h(\vec{x}_i))}_{y_i = 0}\right] + \frac{\lambda}{2} \sum_{j=1}^{p} w_j^2$$

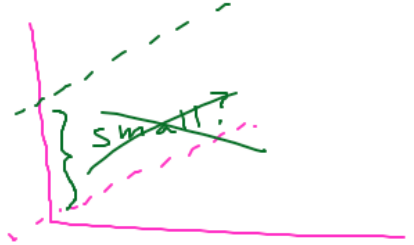
λ = regularization param (hyper)

abs. value

=> small + positive } $[0, 1)$

and

don't regularize bias (w_0)



SGD

$$\vec{w} \leftarrow \vec{w} - \alpha \left[\underbrace{(h(\vec{x}_i) - y_i)}_{p+1} \vec{x}_i + \lambda \vec{w}^* \right] \quad \vec{w}^* = \begin{bmatrix} 0 \\ w_1 \\ \vdots \\ w_p \end{bmatrix} \; p+1$$

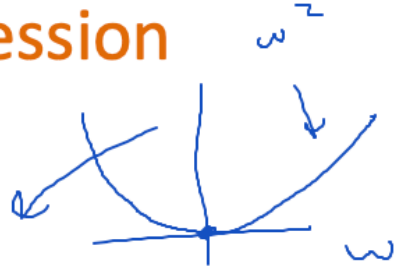
$$\vec{w} \leftarrow \underbrace{(1 - \alpha\lambda)}_{\text{pulls } \vec{w} \text{ toward } 0} \vec{w} - \alpha \nabla J^R_{\vec{x}_i}(\vec{w}) \quad \nabla J^R_{\vec{x}_i}(\vec{w}) = \left[(h(x_i) - y_i) x_i\right]$$

$[0, 1)$

- $\lambda \leftarrow$ regularizer
- $\alpha \leftarrow$ learning rate
- $\varepsilon \leftarrow$ stopping criteria

Adding Regularization to Logistic Regression

$$\min_{w} J(w) = \text{[rectangle]} + \frac{\lambda}{2} w^2$$



min with $w = 0$

$$J'(w) = \lambda w = 0$$

$$\Rightarrow \boxed{w = 0}$$

Outline for October 16

- Maximum Likelihood Estimation (MLE) in other fields
- Recap handouts and logistic regression so far
- Regularization
- Multi-class logistic regression

In "Sunday" video!