Topic 4
Linear Classification & Logistic Regression

- http://cs229.stanford.edu/notes2020fall/notes2020fall/cs229-notes1.pdf
- https://eight2late.wordpress.com/2017/07/11/a-gentle-introduction-to-logistic-regression-and-lasso-regularisation-using-r/

PROF. LINDA SELLIE

Some slides/approaches used are from Prof. Rangan
Many approaches used are from CMU 18-661

Learning objectives

- Know how to use a hyperplane for binary classification
- Use the sigmoid function to scale a number in the range $[-\infty, \infty]$ into $[0,1]$
- Apply the principle of maximum likelihood estimation (MLE) to learn the parameters of a probabilistic model
- Derive the conditional log-likelihood
- How to apply gradient ascent to find the parameters of the the conditional log-likelihood
- Evaluate performance with different measures
- Create more complex models by feature transformation
- Understand how to add L1 and L2 regularization to the objective function
- Know how to interpret the output of soft-max

Logistic Regression

Data: $(\mathbf{x}^{(i)}, y^{(i)}), i = 1,2,\ldots,N$ where $\mathbf{x} \in \mathbb{R}^d$ and $y \in \{0,1\}$

model: Logistic function applied to $\mathbf{w}^T\mathbf{x}$

$$p(y = 1 \mid \mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

We just developed an intuition on why this makes sense

Next we will show why this is true

And find an optimizer to find the "best" w

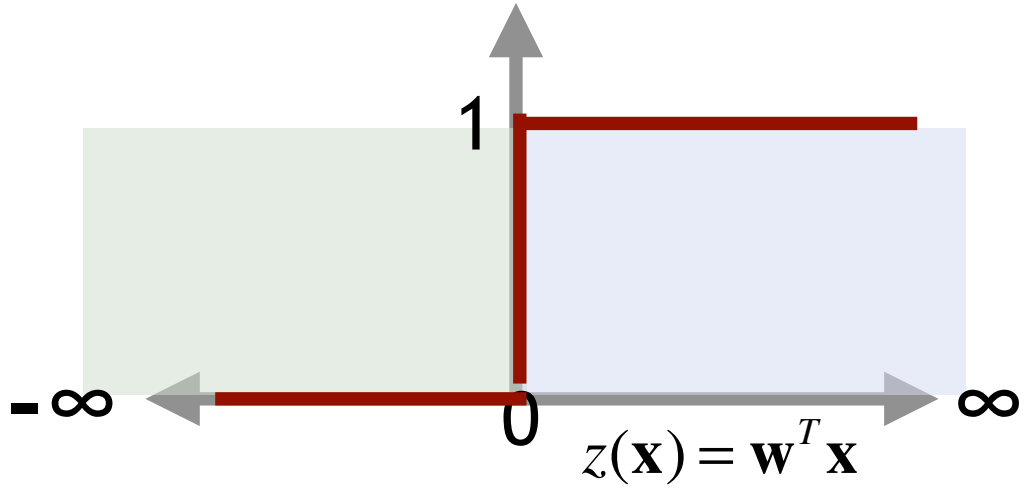
Learning: find parameters that maximizes the **objective function:**

$$\mathbf{w}* = \arg\max_{\mathbf{w}} \left(\sum_{i=1}^{N} y^{(i)} \ln(\sigma(\mathbf{w}^T\mathbf{x}^{(i)})) + (1 - y^{(i)}) \ln\left(1 - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right) \right)$$

where $\sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$

Prediction: $\hat{y} = \arg\max_{y \in \{0,1\}} p(y \mid \mathbf{x}; \mathbf{w})$ or $\hat{y} = p(y \mid \mathbf{x}; \mathbf{w})$

Intuition: Logistic Regression



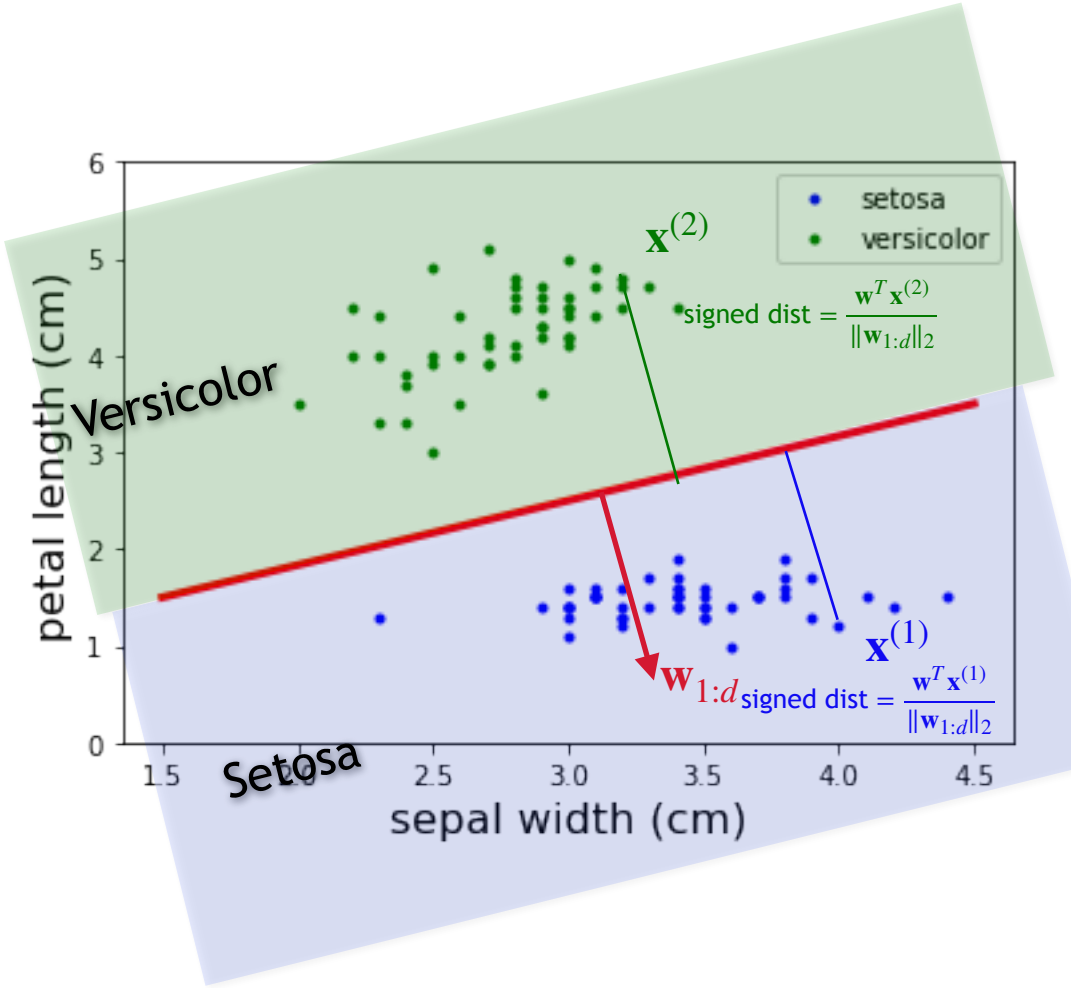
Half-spaces:

$$\mathcal{H}^- = \{\mathbf{x} : \mathbf{w}^T\mathbf{x} < 0\}$$

$$\mathcal{H}^+ = \{\mathbf{x} : \mathbf{w}^T\mathbf{x} > 0\}$$

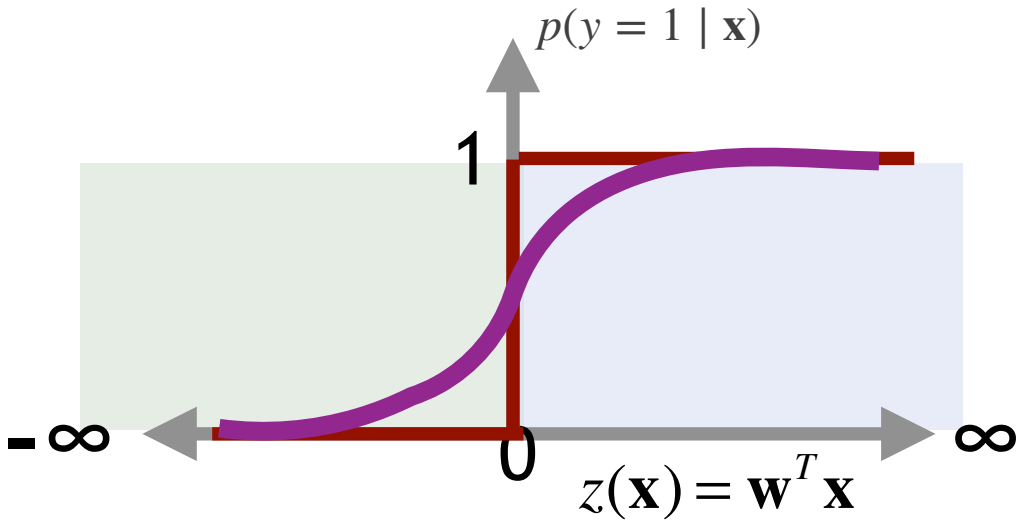
Hyperplane:

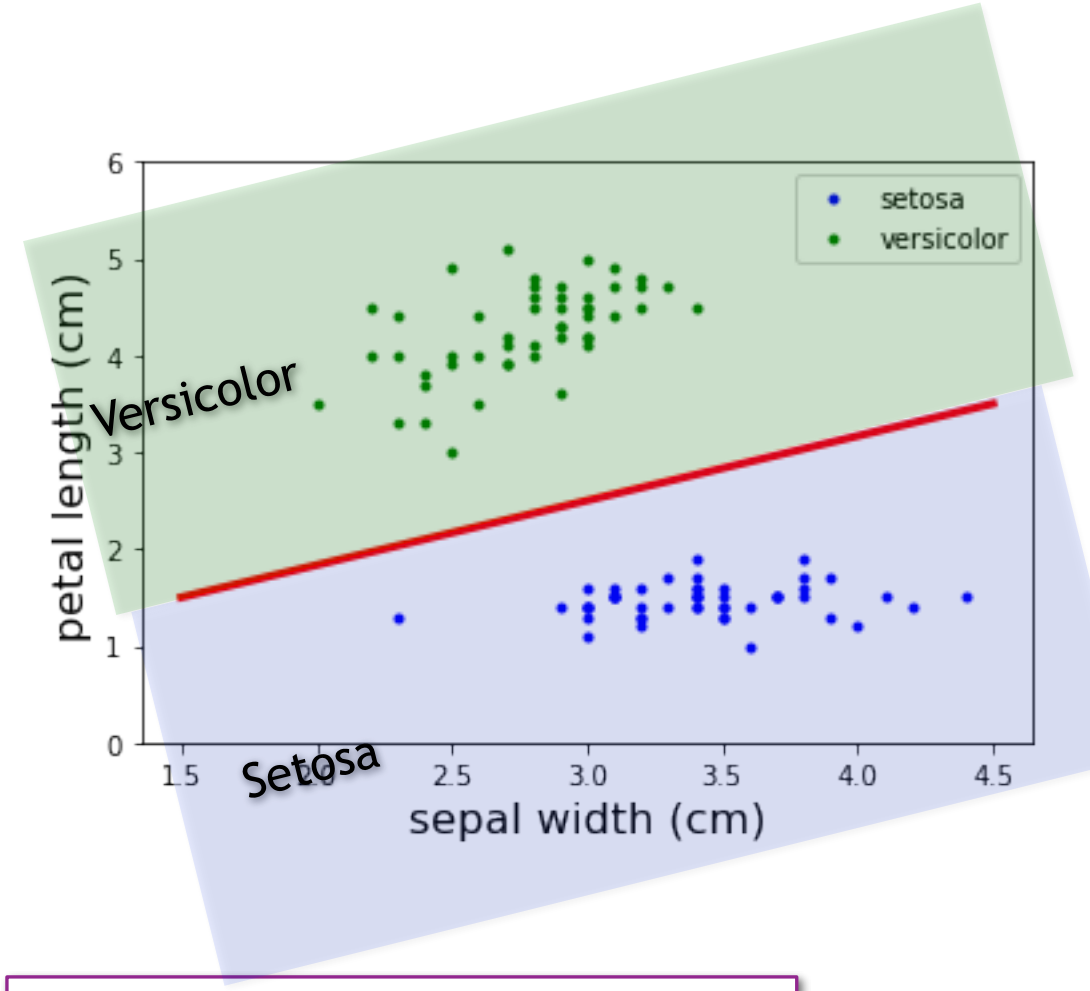
$$\mathcal{H} = \{\mathbf{x} : \mathbf{w}^T\mathbf{x} = 0\}$$



Intuition: Logistic Regression

$$\sigma(z(\mathbf{x})) = \frac{1}{1+e^{-z(\mathbf{x})}} = \frac{1}{1+e^{-(\mathbf{w}^T\mathbf{x})}}$$ returns the probability $\mathbf{x}$ has label 1.





Note: We still have to find $\mathbf{w}$

Outline

- Motivating example: How can we classify? — How can we use a hyperplane for a classification problem?
- Estimating probabilities — Can we predict not only which class an example belongs to - but a confidence score of that classification
- Maximum likelihood — How can we find the most likely hyperplane? Could we write a function to describe how likely a hyperplane was to have generated the dataset?
  - Iterative approach - gradient ascent — Maximizing the function
- Thinking about different types of error — Some errors are more costly than other errors. Can we modify our predictions to decrease one type of error (and perhaps increase another type of error?
- Transformation of the features — Extending our algorithm to nonlinear decision boundaries
- Multiple classes — What if we have more than two classes?

Outline

- Motivating example: How can we classify?
  - How can we use a hyperplane for a classification problem?
- Which model
  - Can we predict not only which class an example belongs to - but a confidence score of that classification
- Finding an objective function
  - How can we find the most likely hyperplane? Could we write a function to describe how likely a hyperplane was to have generated the dataset?
- Optimizer
  - Ite - gradient ascent
    - Maximizing the function
- Thinking about different types of error
  - Some errors are more costly than other errors. Can we modify our predictions to decrease one type of error (and perhaps increase another type of error?
- Transformation of the features
  - Extending our algorithm to nonlinear decision boundaries
- Multiple classes
  - What if we have more than two classes?

How can we find the “best” hyperplane, $\mathbf{w}$?

Optimize $\mathbf{w}$

We first need to decide what makes one hyperplane better than another (i.e. an objective function)

MLE!

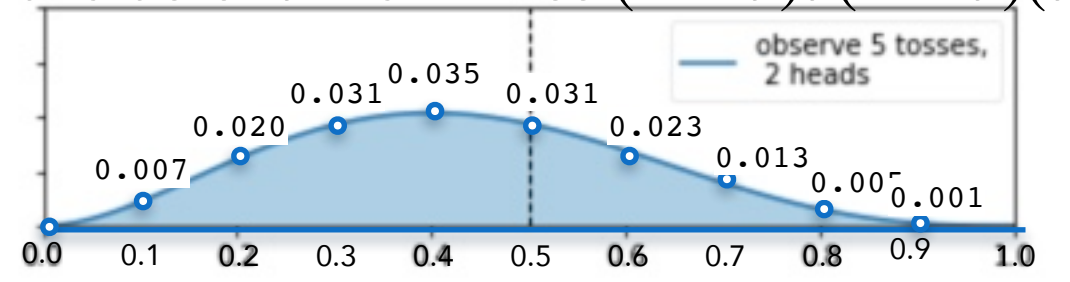
We will first find the MLE for a simpler problem

Afterwards, we will find the MLE for the problem we want to solve

Data → Estimation

- T, H, T, H, T,
- If we want to predict θ (the probability of heads), how can we estimate (learn) θ
- One measure of goodness is the θ that most *likely* generated the data

What value of θ maximizes $(1-\theta)\theta(1-\theta)(\theta)(1-\theta)$?



$$L(\theta) = \theta^{N_H}(1-\theta)^{N_T}$$

0	$0^2(1-0)^3$
0.1	$0.1^2(1-0.1)^3$
0.2	$0.2^2(1-0.2)^3$
0.3	$0.3^2(1-0.3)^3$
0.4	$0.4^2(1-0.4)^3$
0.5	$0.5^2(1-0.5)^3$
0.6	$0.6^2(1-0.6)^3$
0.7	$0.7^2(1-0.7)^3$
0.8	$0.8^2(1-0.8)^3$
0.9	$0.9^2(1-0.9)^3$
1	$1^2(1-1)^3$

Data → Estimation

- T, H, T, H, T,
- If we want to predict θ (the probability of heads), I estimate (learn) θ
- One measure of goodness is the θ that most *likely* ge the data

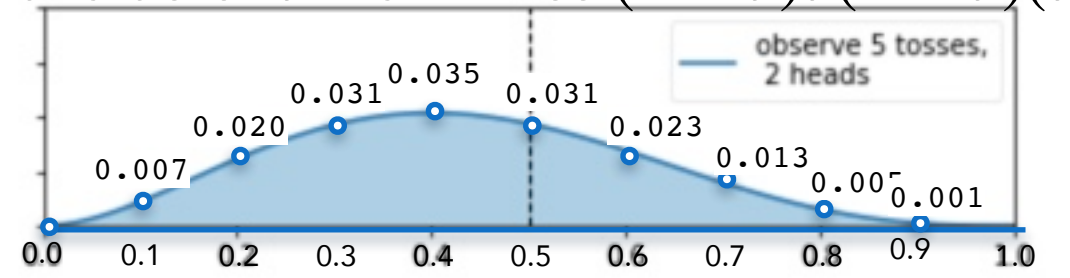
How does the likelihood of seeing the data change as we change θ?

Which θ makes observing the data $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}$ most likely?

Likelihood is always relative to some model - In this example the model is Bernoulli

Coin images very slightly modified from Classical Numismatic Group, Inc. http://www.cngcoins.com [CC BY-SA (http://creativecommons.org/licenses/by-sa/3.0/)]

What value of θ maximizes $(1-\theta)\theta(1-\theta)(\theta)(1-\theta)$?



$$L(\theta) = \theta^{N_H}(1-\theta)^{N_T}$$

0.1	$0.1^2(1-0.1)^3$
0.2	$0.2^2(1-0.2)^3$
1	$1^2(1-1)^3$

Typically we view the distribution θ as fixed, and the examples as parameters. We are turning this idea "on its head". Here the examples are fixed and we are considering different choices for the parameter values

Maximum likelihood estimation (MLE)

https://upload.wikimedia.org/wikipedia/commons/3/3b/

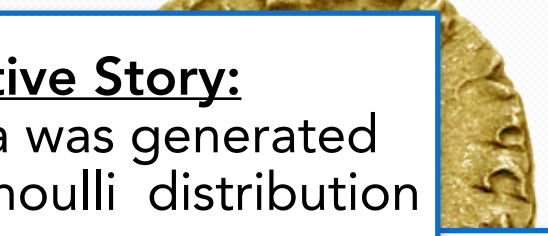
- Flip a (unfair?) coin 100 times and if N_H=55 and N_T = 45
- What is p(H)?
- **Likelihood function** $L(\theta)$ is the probability of the observed data as a function of θ.
  For this example: $L(\theta) = p(D \mid \theta) = \theta^{N_H}(1-\theta)^{N_T}$
- **Log-likelihood function** $\ell(\theta) = \log L(\theta)$
- **Maximum likelihood criterion** the most likely parameter is the one that maximizes $\ell(\theta)$
- How to maximize $\ell(\theta)$

Extremely small value

If θ was 0.5, then
$L(0.5) = 0.5^{100} \approx 7.9 \times 10^{-31}$

If θ was 0.5, then
$\ell(0.5) = \log 0.5^{100} = 100 \log 0.5 = -69.31$

Maximum likelihood estimation (MLE)



Create Generative Story:
Assume the data was generated i.i.d. from a Bernoulli distribution

Coin flips are conditionally independently p(Heads)=θ and identically distributed (i.i.d.)

- Flip a (unfair?) coin 100 times and if N_H=55 a
- What is p(H)?
- **Likelihood function** $L(\theta)$ is the probability o observed data as a function of θ.
  For this example: $L(\theta) = p(D \mid \theta) = \theta^{N_H}(1-\theta)^{N_T}$

L is a function of the model parameters, not the data

- **Log-like** $= \log L(\theta)$
- **Maximum likelihood criterion** the most likely parameter is the one that maximizes $\ell(\theta)$

In computer science log is always base 2…. In Machine learning log is always base e

Maximizing $\ell(\theta)$ is the same as maximizing $L(\theta)$. Why?

- How to maximize $\ell(\theta)$?

What if we had 100 coin tosses, 40 heads and 60 tails

Which is the right likelihood function?

θ will be your estimated probability of flipping a coin and getting heads.

A) $L(\theta) = (0.4)^{40}(1 - 0.4)^{60}$

B) $L(\theta) = (\theta)^{40}(1 - \theta)^{60}$

C) $L(\theta) = (0.4)^{\theta}(1 - 0.4)^{1-\theta}$

D) $L(\theta) = (0.8)^{60}(1 - 0.8)^{100}$

What if we had 100 coin tosses, 40 heads and 60 tails

Which is the right likelihood function?

θ will be your estimated probability of flipping a coin and getting heads.

A) $L(\theta) = (0.4)^{40}(1 - 0.4)^{60}$

B) $L(\theta) = (\theta)^{40}(1 - \theta)^{60}$ 

C) $L(\theta) = (0.4)^{\theta}(1 - 0.4)^{1-\theta}$

D) $L(\theta) = (0.8)^{60}(1 - 0.8)^{100}$

We will extend this idea to Conditional likelihood.

What is the probability seeing y conditioned on **x**

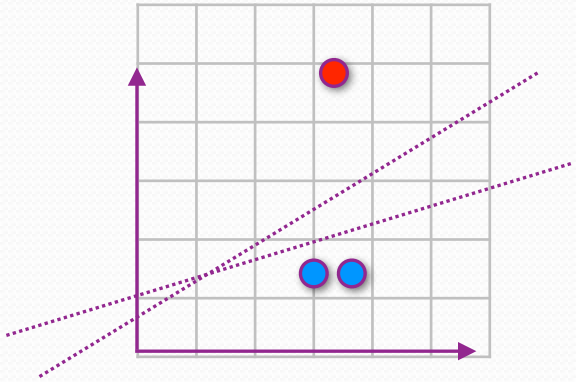
Given $D = \{(\mathbf{x}^{(1)}, y^{(1)}), \ldots, (\mathbf{x}^{(N)}, y^{(N)})\}$,

How can we find the best $\mathbf{w}$?

How can we use MLE for our problem?

Pair share

Likelihood of seeing the data



- Our model: $p(y^{(i)} \mid \mathbf{x}^{(i)}; \mathbf{w}) = \begin{cases} \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) & \text{for } y^{(i)} = 1 \\ 1 - \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) & \text{for } y^{(i)} = 0 \end{cases}$

- Given the following data:

$\mathbf{x}^{(1)} = [1,\ 3.2\ \ 4.7]$ $\quad y^{(1)} = 0$

$\mathbf{x}^{(2)} = [1,\ 3.5\ \ 1.4\ ]$ $\quad y^{(2)} = 1$

$\mathbf{x}^{(3)} = [1,\ 3.\ \ 1.4\ ]$ $\quad y^{(3)} = 1$

- How likely were we to see the data if the line was:

$$\mathbf{w} = \begin{bmatrix} 1/2 \\ 2/3 \\ -1 \end{bmatrix} \qquad \left(1 - \frac{1}{1+e^{-(1/2+(2/3)3.2-4.7)}}\right)\left(\frac{1}{1+e^{-(1/2+(2/3)3.5-1.4)}}\right)\left(\frac{1}{1+e^{-(1/2+(2/3)3-1.4)}}\right) = 0.54$$

1-0.11 $\qquad$ 0.81 $\qquad$ 0.75

$$\mathbf{w} = \begin{bmatrix} 1 \\ 1/3 \\ -1 \end{bmatrix} \qquad \left(1 - \frac{1}{1+e^{-(1+(1/3)3.2-4.7)}}\right)\left(\frac{1}{1+e^{-(1+(1/3)3.5-1.4)}}\right)\left(\frac{1}{1+e^{-(1+(1/3)3-1.4)}}\right) = 0.41$$

Pair share: Write the conditional likelihood function for these three examples

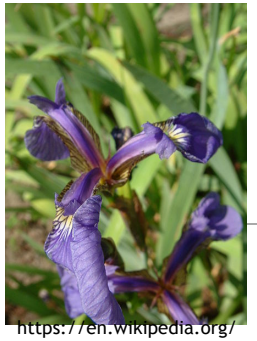
Our model:

$$p(y^{(i)} \mid \mathbf{x}^{(i)}; \mathbf{w}) = \begin{cases} \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) & \text{for } y^{(i)} = 1 \\ 1 - \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) & \text{for } y^{(i)} = 0 \end{cases}$$

$$p(y = 1 \mid \mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T \mathbf{x}) = \frac{1}{1 + e^{-(\mathbf{w}^T \mathbf{x})}}$$

$$p(y = 0 \mid \mathbf{x}; \mathbf{w}) = 1 - \sigma(\mathbf{w}^T \mathbf{x}) = 1 - \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x}}}$$

$$D = \{(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), (\mathbf{x}^{(3)}, y^{(3)})\}$$



https://en.wikipedia.org/wiki/Iris_flower_data_set#/media/

setosa

$\mathbf{x}^{(2)} = [1, \ 3.5 \ \ 1.4 \ ] \ y^{(2)} = 1$

$\mathbf{x}^{(3)} = [1, \ 3. \ \ \ 1.4 \ ] \ y^{(3)} = 1$

https://commons.wikimedia.org/

versicolor

$\mathbf{x}^{(1)} = [1, \ 3.2 \ \ 4.7] \ \ y^{(1)} = 0$

How could we write the conditional likelihood function

Class exercise

versicolor

$\mathbf{x}^{(1)}$ = [1 3.2 4.7] $y^{(1)} = 0$

setosa

$\mathbf{x}^{(2)}$ = [1 3.5 1.4] $y^{(2)} = 1$

$\mathbf{x}^{(3)}$ = [1 3. 1.4] $y^{(3)} = 1$

$$p(y = 1 \mid \mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T\mathbf{x}}}$$

$$p(y = 0 \mid \mathbf{x}; \mathbf{w}) = 1 - \sigma(\mathbf{w}^T\mathbf{x}) = 1 - \frac{1}{1 + e^{-\mathbf{w}^T\mathbf{x}}}$$

$$L(\mathbf{w}) = \left(1 - \frac{1}{1 + e^{-(\mathbf{w}^T\mathbf{x}^{(1)})}}\right)\left(\frac{1}{1 + e^{-(\mathbf{w}^T\mathbf{x}^{(2)})}}\right)\left(\frac{1}{1 + e^{-(\mathbf{w}^T\mathbf{x}^{(3)})}}\right)$$

$$L(\mathbf{w}) = \left(1 - \frac{1}{1 + e^{-(w_0 + w_1 3.2 + w_2 4.7)}}\right)\left(\frac{1}{1 + e^{-(w_0 + w_1 3.5 + w_2 1.4)}}\right)\left(\frac{1}{1 + e^{-(w_0 + w_1 3 + w_2 1.4)}}\right)$$

$$L(\mathbf{w}) = \left(1 - p(y = 1 \mid \mathbf{x}^{(1)}; \mathbf{w})\right)\ p(y = 1 \mid \mathbf{x}^{(2)}; \mathbf{w}) p(y = 1 \mid \mathbf{x}^{(3)}; \mathbf{w}) = \prod_{i=1}^{N} p(y^{(i)} \ \begin{matrix}\text{correctly} \\ \text{predicted}\end{matrix} \mid \mathbf{x}^{(i)}; \mathbf{w})$$

$$L(\mathbf{w}) = \prod_{i:y^{(i)}=1} p(y^{(i)} = 1 \mid \mathbf{x}^{(i)}; \mathbf{w}) \prod_{i:y^{(i)}=0} \left(1 - p(y^{(i)} = 1 \mid \mathbf{x}^{(i)}; \mathbf{w})\right)$$

The conditional likelihood function

Conditional likelihood function (conditioned on **x**). Larger value means more likely

$$L(\mathbf{w}) = \prod_{i:y^{(i)}=1} p(y^{(i)}=1 \mid \mathbf{x}^{(i)};\mathbf{w}) \prod_{i:y^{(i)}=0} \left(1 - p(y^{(i)}=1 \mid \mathbf{x}^{(i)};\mathbf{w})\right)$$

Here we assume all the examples are independent

$$\prod_{i:y^{(i)}=1} p(y^{(i)}=1 \mid \mathbf{x}^{(i)};\mathbf{w})^{y^{(i)}} \left(1 - p(y^{(i)}=1 \mid \mathbf{x}^{(i)};\mathbf{w})\right)^{1-y^{(i)}} \prod_{i:y^{(i)}=0} \left(1 - p(y^{(i)}=1 \mid \mathbf{x}^{(i)};\mathbf{w})\right)^{1-y^{(i)}} p(y^{(i)}=1 \mid \mathbf{x}^{(i)};\mathbf{w})^{y^{(i)}}$$

$$L(\mathbf{w}) = \prod_{i=1}^{N} p(y^{(i)}=1 \mid \mathbf{x}^{(i)};\mathbf{w})^{y^{(i)}} (1 - p(y^{(i)}=1 \mid \mathbf{x}^{(i)};\mathbf{w}))^{1-y^{(i)}}$$

$$= \prod_{i=1}^{N} \sigma(\mathbf{w}^T\mathbf{x}^{(i)})^{y^{(i)}} \left(1 - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)^{1-y^{(i)}}$$

Define: $p(y^{(i)}=1 \mid \mathbf{x}^{(i)};\mathbf{w}) = \sigma(\mathbf{w}^T\mathbf{x}^{(i)}) = \dfrac{1}{1+e^{-\mathbf{w}^T\mathbf{x}^{(i)}}}$

Pair share: how do we find the $\mathbf{w}$ that maximizes this function

Maximize $L(\mathbf{w}) = \prod_{i=1}^{N} \sigma(\mathbf{w}^T \mathbf{x}^{(i)})^{y^{(i)}} \left(1 - \sigma(\mathbf{w}^T \mathbf{x}^{(i)})\right)^{1-y^{(i)}}$

The log-likelihood function

Define: $p(y^{(i)} = 1 \mid \mathbf{x}^{(i)}; \mathbf{w}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T\mathbf{x}}}$

- We wanted to maximize $L(\mathbf{w}) = \prod_{i=1}^{N} \sigma(\mathbf{w}^T\mathbf{x}^{(i)})^{y^{(i)}} \left(1 - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)^{1-y^{(i)}}$

- This is the same as maximizing $\ell(\mathbf{w}) = \ln(L(\mathbf{w}))$

$$= \ln\left[\prod_{i=1}^{N} \sigma(\mathbf{w}^T\mathbf{x}^{(i)})^{y^{(i)}} \left(1 - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)^{1-y^{(i)}}\right]$$

$$= \sum_{i=1}^{N} \ln\left[\sigma(\mathbf{w}^T\mathbf{x}^{(i)})^{y^{(i)}} \left(1 - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)^{1-y^{(i)}}\right]$$

$\log a^c b^d = c \log a + d \log b$

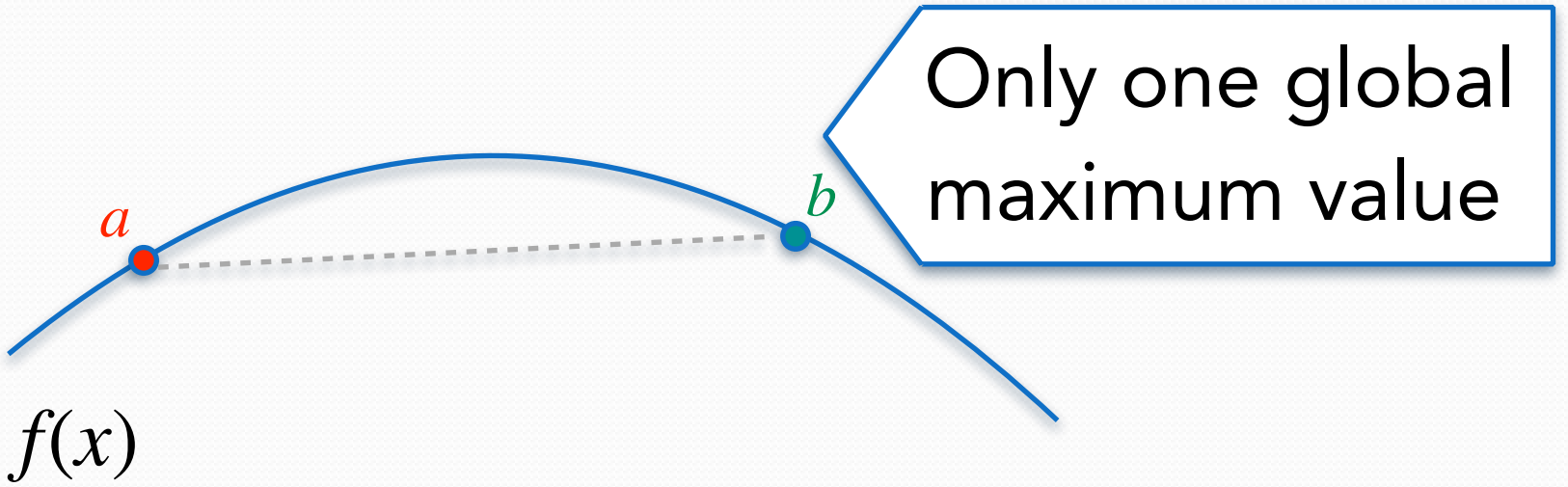
a^c b^d

$$= \sum_{i=1}^{N} \left[y^{(i)} \ln \sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + (1 - y^{(i)}) \ln\left(1 - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\right]$$

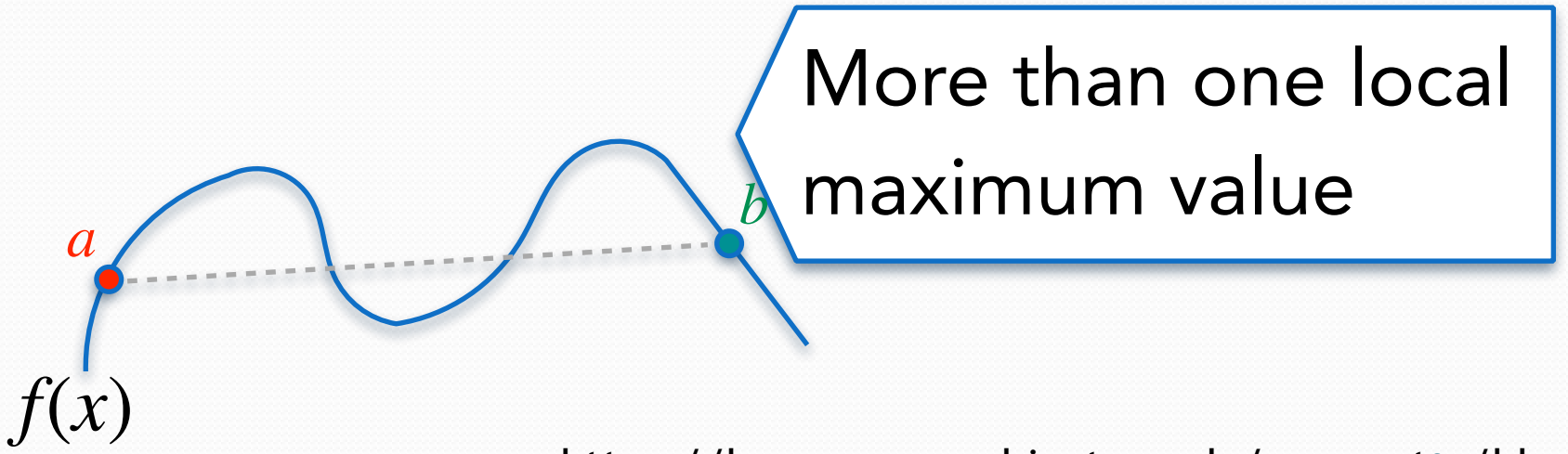
$c \log a$ $d \log b$

- How do we maximize the conditional likelihood?

Concave function



Not a concave function



https://homes.cs.washington.edu/~marcotcr/blog/concavity/

Finding $\mathbf{w}$

$$\ell(\mathbf{w}) = \sum_{i=1}^{N} \left[y^{(i)} \ln \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) + (1 - y^{(i)}) \ln \left(1 - \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) \right) \right]$$

- There is no closed form solution (i.e. we don't have a global optimization technique)
- We can move toward the optimal value using gradient *ascent*
- To find which way to move, we take the gradient of $\ell(\mathbf{w})$.

Logistic Regression

Data: $(\mathbf{x}^{(i)}, y^{(i)}), i = 1,2,\ldots,N$ where $\mathbf{x} \in \mathbb{R}^d$ and $y \in \{0,1\}$

model: Logistic function applied to $\mathbf{w}^T\mathbf{x}$

$$p(y = 1 \mid \mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

Learning: find parameters that maximizes the **objective function:**

$$\mathbf{w}* = \arg\max_{\mathbf{w}} \left(\frac{1}{N} \sum_{i=1}^{N} y^{(i)} \ln(\sigma(\mathbf{w}^T\mathbf{x}^{(i)})) + (1 - y^{(i)}) \ln\left(1 - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right) \right)$$

where $\sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$

Maximum Likelihood estimator (MLE) **w***

Next we will show how to find the optimal **w***

Prediction: either $\hat{y} = p(y \mid \mathbf{x}; \mathbf{w})$ or $\hat{y} = \arg\max_{y \in \{0,1\}} p(y \mid \mathbf{x}; \mathbf{w})$

Equivalent objective function choices

$$\ell(\mathbf{w}) = \sum_{i=1}^{N} \left[y^{(i)} \ln \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) + (1 - y^{(i)}) \ln \left(1 - \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) \right) \right]$$

$$\frac{1}{N}\ell(\mathbf{w}) = \frac{1}{N} \sum_{i=1}^{N} \left[y^{(i)} \ln \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) + (1 - y^{(i)}) \ln \left(1 - \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) \right) \right]$$

Finds same $\mathbf{w}$

$$-\ell(\mathbf{w}) = \sum_{i=1}^{N} - \left[y^{(i)} \ln \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) + (1 - y^{(i)}) \ln \left(1 - \sigma(\mathbf{w}^T \mathbf{x}^{(i)}) \right) \right]$$

Negative log likelihood is the Cross entropy error (minimize function)

Outline

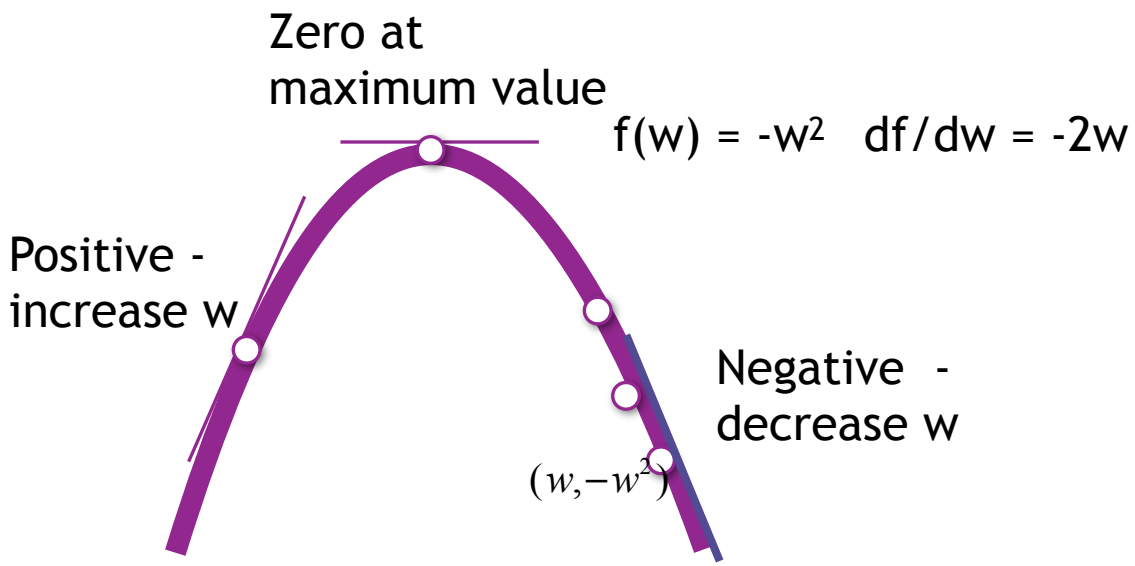
- Motivating example: How can we classify? — How can we use a hyperplane for a classification problem?
- Estimating probabilities — Can we predict not only which class an example belongs to - but a confidence score of that classification
- Maximum likelihood — How can we find the most likely hyperplane? Could we write a function to describe how likely a hyperplane was to have generated the dataset?
  - Iterative approach - gradient ascent — Maximizing the function
- Thinking about different types of error — Some errors are more costly than other errors. Can we modify our predictions to decrease one type of error (and perhaps increase another type of error?
- Transformation of the features — Extending our algorithm to nonlinear decision boundaries
- Multiple classes — What if we have more than two classes?

Maximize a function by repeatedly moving toward the maximum

1. For i = 1 to num_iters:
if f'(w) > 0 then f is increasing,
move w a little to the ___

if f'(w) < 0 then f is decreasing,
move w a little to the ___

- right, right
- left, left
- right, left
- left, right
- none of these options



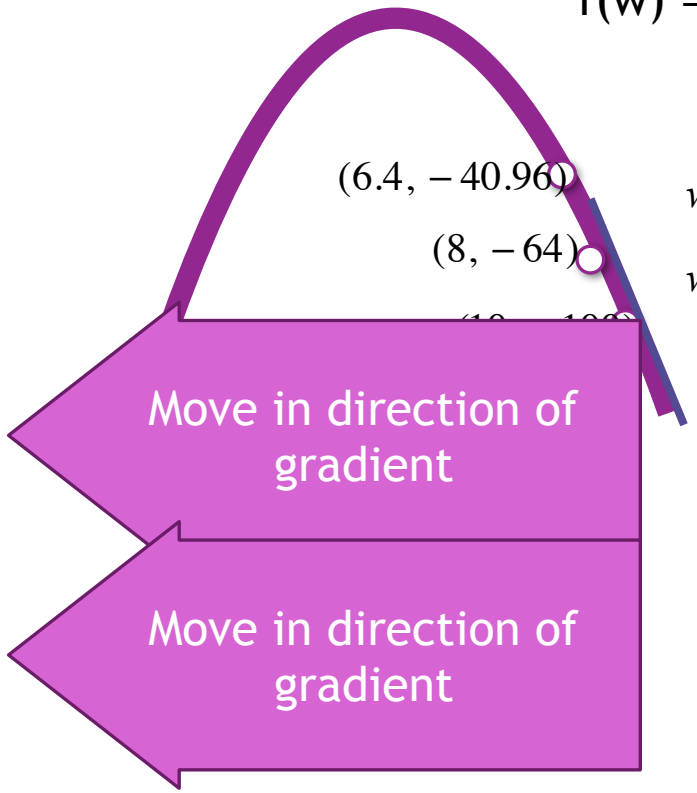
Maximize a function by repeatedly moving toward the maximum

Algorithm:
For i = 1 to num_iters:

if $f'(w) > 0$ then f is increasing, move w a little to the right

if $f'(w) < 0$ then f is decreasing, move w a little to the left

f(w) = -w² df/dw = -2w



$$w_{new} = w + \alpha \cdot (-2 \cdot w) \quad \alpha = 0.1$$

$$w_{new} = 10 + \alpha \cdot (-2 \cdot 10) = 10 - (0.1) \cdot 2 \cdot 10 = 8$$

$$w_{new} = 8 - \alpha \cdot 2 \cdot 8 \quad = 8 - (0.1) \cdot 2 \cdot 8 = 6.4$$

The Gradient

- We wanted to maximize $\frac{1}{N}\ell(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N}\left[y^{(i)}\ln\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + (1-y^{(i)})\ln\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\right]$

- We would like to find where the derivative is zero (ie the most likely values for **w**.)
  We do this by taking steps towards the maximum value

$$w_j = w_j + \frac{\alpha}{N}\frac{\partial}{\partial w_j}\ell(\mathbf{w})$$

for k = 1 to num_iter

$$temp0 = w_0 + \frac{\alpha}{N}\frac{\partial\,\ell(\mathbf{w})}{\partial w_0}$$

$$temp1 = w_1 + \frac{\alpha}{N}\frac{\partial\,\ell(\mathbf{w})}{\partial w_1}$$

$$temp2 = w_2 + \frac{\alpha}{N}\frac{\partial\,\ell(\mathbf{w})}{\partial w_2}$$

$$w_0^{new} = temp0$$

$$w_1^{new} = temp1$$

$$w_2^{new} = temp2$$

```
[1  3.2  4.7 ]
[1  3.2  4.5 ]
[1  3.1  4.9 ]
[1  2.3  4.  ]
[1  2.8  4.6 ]
[1  2.8  4.5 ]
[1  3.3  4.7 ]
[1  2.4  3.3 ]
[1  2.9  4.6 ]
[1  2.7  3.9 ]
```

Label 0 examples

```
[1  3.5  1.4 ]
[1  3.   1.4 ]
[1  3.2  1.3 ]
[1  3.1  1.5 ]
[1  3.6  1.4 ]
[1  3.9  1.7 ]
[1  3.4  1.4 ]
[1  3.4  1.5 ]
[1  2.9  1.4 ]
[1  3.1  1.5 ]
```

Label 1 examples

Its just math…

$$h(\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T\mathbf{x})}}$$

Our update rule:

- We wanted to maximize $\frac{1}{N}\ell(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N}\left[y^{(i)}\ln\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + (1-y^{(i)})\ln\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\right]$
- We would like to find where the gradient is zero (ie the most likely values for **w**.) We do this by taking steps towards the maximum value $w_j = w_j + \frac{\alpha}{N}\frac{\partial}{\partial w_j}\ell(\mathbf{w})$ for each j

The derivative of the sigmoid function:

$$\frac{d}{dz}\left(\frac{1}{1+e^{-z}}\right) = \frac{1}{\left(1+e^{-z}\right)^2}e^{-z}$$

$$= \frac{1}{\left(1+e^{-z}\right)^2}(e^{-z}+1-1) \quad = \frac{1+e^{-z}}{\left(1+e^{-z}\right)^2} - \frac{1}{\left(1+e^{-z}\right)^2}$$

$$= \frac{1}{\left(1+e^{-z}\right)} - \frac{1}{\left(1+e^{-z}\right)^2} \quad = \frac{1}{\left(1+e^{-z}\right)}\left(1 - \frac{1}{\left(1+e^{-z}\right)}\right)$$

You will not be tested on this material with the pink background

Thus:

$$\sigma(z) = \left(\frac{1}{1+e^{-z}}\right)$$

$$\frac{d\sigma(z)}{dz} = \sigma(z)(1-\sigma(z))$$

Useful to know before we derive $\frac{\partial}{\partial w_j}\ell(\mathbf{w})$

$$\frac{\partial\, \mathbf{w}^T\mathbf{x}}{\partial w_j} = \frac{\partial\,(w_0x_0 + w_1x_1 + \cdots w_jx_j + \cdots w_dx_d)}{\partial w_j} = x_j$$

$$\frac{\partial\sigma(\mathbf{w}^T\mathbf{x})}{\partial w_j} = \sigma(\mathbf{w}^T\mathbf{x})(1-\sigma(\mathbf{w}^T\mathbf{x}))\frac{\partial \mathbf{w}^T\mathbf{x}}{w_j} = \sigma(\mathbf{w}^T\mathbf{x})\left(1-\sigma(\mathbf{w}^T\mathbf{x})\right)x_j$$

$$\boxed{\frac{\partial\sigma(\mathbf{w}^T\mathbf{x})}{\partial w_j} = \sigma(\mathbf{w}^T\mathbf{x})\left(1-\sigma(\mathbf{w}^T\mathbf{x})\right)x_j}$$

The Gradient

$$\sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T\mathbf{x})}} \qquad \frac{\partial \sigma(\mathbf{w}^T\mathbf{x})}{\partial w_j} = \sigma(\mathbf{w}^T\mathbf{x})(1-\sigma(\mathbf{w}^T\mathbf{x}))\frac{\partial \mathbf{w}^T\mathbf{x}}{w_j} \qquad \frac{\partial \mathbf{w}^T\mathbf{x}}{\partial w_j} = x_j$$

$$\frac{\partial}{\partial w_j}\ell(\mathbf{w}) = \frac{\partial}{w_j}\sum_{i=1}^{N}\left[y^{(i)}\ln\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + (1-y^{(i)})\ln\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\right] = \sum_{i=1}^{N}\left[y^{(i)}\frac{\partial}{w_j}\ln\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + (1-y^{(i)})\frac{\partial}{w_j}\ln\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\right]$$

$$= \sum_{i=1}^{N}\left[y^{(i)}\frac{1}{\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}\frac{\partial\,\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}{w_j} - (1-y^{(i)})\frac{1}{1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}\frac{\partial\,\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}{w_j}\right] = \sum_{i=1}^{N}\left[y^{(i)}\frac{1}{\sigma(\mathbf{w}^T\mathbf{x}^{(i)})} - (1-y^{(i)})\frac{1}{1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}\right]\frac{\partial\,\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}{w_j}$$

$$= \sum_{i=1}^{N}\left[y^{(i)}\frac{1}{\sigma(\mathbf{w}^T\mathbf{x}^{(i)})} - (1-y^{(i)})\frac{1}{1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}\right]\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\frac{\partial\,\mathbf{w}^T\mathbf{x}^{(i)}}{w_j} = \sum_{i=1}^{N}\left(y^{(i)}\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right) - (1-y^{(i)})\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)x_j^{(i)}$$

$$= \sum_{i=1}^{N}\left(y^{(i)}\cdot 1 - y^{(i)}\cdot\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) - 1\cdot\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + y^{(i)}\cdot\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)x_j^{(i)}$$

$$= \sum_{i=1}^{N}\left(y^{(i)} - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)x_j^{(i)}$$

Error of i^{th} training example

You will not be tested on this material with the purple background

The Gradient

$$\sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T\mathbf{x})}} \qquad \frac{\partial \sigma(\mathbf{w}^T\mathbf{x})}{\partial w_j} = \sigma(\mathbf{w}^T\mathbf{x})(1-\sigma(\mathbf{w}^T\mathbf{x}))\frac{\partial \mathbf{w}^T\mathbf{x}}{w_j} \qquad \frac{\partial \mathbf{w}^T\mathbf{x}}{\partial w_j} = x_j$$

$$\frac{\partial}{\partial w_j}\ell(\mathbf{w}) = \frac{\partial}{w_j}\sum_{i=1}^{N}\left[y^{(i)}\ln\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + (1-y^{(i)})\ln\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\right] = \sum_{i=1}^{N}\left[y^{(i)}\frac{\partial}{w_j}\ln\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + (1-y^{(i)})\frac{\partial}{w_j}\ln\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\right]$$

$$= \sum_{i=1}^{N}\left[y^{(i)}\frac{1}{\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}\frac{\partial\,\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}{w_j} - (1-y^{(i)})\frac{1}{1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}\frac{\partial\,\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}{w_j}\right] = \sum_{i=1}^{N}\left[y^{(i)}\frac{1}{\sigma(\mathbf{w}^T\mathbf{x}^{(i)})} - (1-y^{(i)})\frac{1}{1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}\right]\frac{\partial\,\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}{w_j}$$

$$= \sum_{i=1}^{N}\left[y^{(i)}\frac{1}{\sigma(\mathbf{w}^T\mathbf{x}^{(i)})} - (1-y^{(i)})\frac{1}{1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})}\right]\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\frac{\partial\,\mathbf{w}^T\mathbf{x}^{(i)}}{w_j} = \sum_{i=1}^{N}\left(y^{(i)}\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right) - (1-y^{(i)})\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)x_j^{(i)}$$

$$= \sum_{i=1}^{N}\left(y^{(i)}\cdot 1 - y^{(i)}\cdot\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) - 1\cdot\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + y^{(i)}\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)x_j^{(i)}$$

$$= \sum_{i=1}^{N}\left(y^{(i)} - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)x_j^{(i)}$$

Error of i^{th} training example

You will not be tested on this material with the purple background

$$\frac{\partial}{\partial w_j}\ell(\mathbf{w}) = \sum_{i=1}^{N}\left(y^{(i)} - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)x_j^{(i)}$$

Interpretation:

- We wanted to maximize $\frac{1}{N}\ell(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N}\left[y^{(i)} \ln \sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + (1-y^{(i)})\ln\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\right]$

- We would like to find where the partial derivative is zero (ie the most likely values for **w**). We do this by taking steps towards the maximum value

$$w_j = w_j + \frac{\alpha}{N}\frac{\partial}{\partial w_j}\ell(\mathbf{w}) = w_j + \frac{\alpha}{N}\sum_{i=1}^{N}\left(y^{(i)} - \sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)x_j^{(i)}$$

- How does an example affect the update?
  - if y=1 and $\sigma(\mathbf{w}^T\mathbf{x})\approx 1$, then $(y-\sigma(\mathbf{w}^T\mathbf{x}))\approx 0$, almost no change!
  - if y=1 and $\sigma(\mathbf{w}^T\mathbf{x})\approx 0$, then $(y-\sigma(\mathbf{w}^T\mathbf{x}))\approx 1$, approx α/N times the j^{th} feature
  - if y=0 and $\sigma(\mathbf{w}^T\mathbf{x})\approx 0$, then $(y-\sigma(\mathbf{w}^T\mathbf{x}))\approx 0$, almost no change!
  - if y=0 and $\sigma(\mathbf{w}^T\mathbf{x})\approx 1$, then $(y-\sigma(\mathbf{w}^T\mathbf{x}))\approx -1$, approx $-\alpha/N$ times the j^{th} feature

Gradient Ascent Algorithm

$$\frac{1}{N}\ell(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N}\left[y^{(i)}\ln\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) + (1-y^{(i)})\ln\left(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)})\right)\right]$$

for k = 1 to num_iter

$$temp0 = w_0 + \alpha\frac{\partial\ell(\mathbf{w})/N}{\partial w_0} = w_0 + \frac{\alpha}{N}\sum_{i=1}^{N}(y^{(i)} - \sigma(\mathbf{w}^T\mathbf{x}^{(i)}))x_0^{(i)}$$

$$temp1 = w_1 + \alpha\frac{\partial\ell(\mathbf{w})/N}{\partial w_1} = w_1 + \frac{\alpha}{N}\sum_{i=1}^{N}(y^{(i)} - \sigma(\mathbf{w}^T\mathbf{x}^{(i)}))x_1^{(i)}$$

$$\vdots$$

$$tempd = w_d + \alpha\frac{\partial\ell(\mathbf{w})/N}{\partial w_d} = w_d + \frac{\alpha}{N}\sum_{i=1}^{N}(y^{(i)} - \sigma(\mathbf{w}^T\mathbf{x}^{(i)}))x_d^{(i)}$$

$x^{(i)}_0=1$

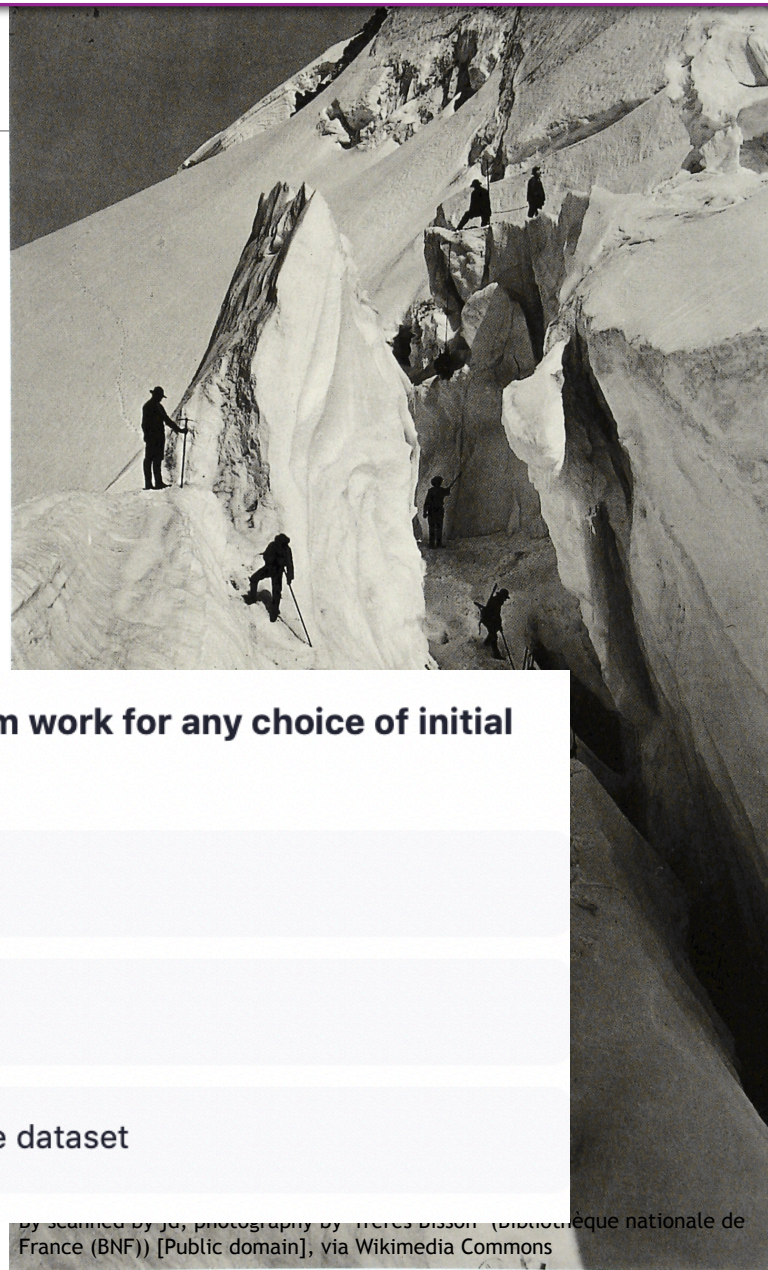
$$w_0 = temp0$$

$$w_1 = temp1$$

$$\vdots$$

$$w_d = tempd$$

Simultaneous update



1. Does this algorithm work for any choice of initial values for w?

- Yes
- No
- It depends on the dataset