



Linear classifiers:



Logistic regression

STAT/CSE 416: Machine Learning
Emily Fox
University of Washington
April 19, 2018

©2018 Emily Fox

How confident is your prediction?

"The sushi & everything
else were awesome!"

Definite **+1**

$$P(y=+1|\mathbf{x}=\text{"The sushi & everything else were awesome!"}) = 0.99$$

"The sushi was good,
the service was OK"

Not sure

$$P(y=+1|\mathbf{x}=\text{"The sushi was good, the service was OK"}) = 0.55$$

Many classifiers provide a degree of certainty:

Output label

$P(y|\mathbf{x})$

Input sentence

Extremely useful in practice

8

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Sentence from review

Input: \mathbf{x}

Predict most likely class
 $\hat{P}(\mathbf{y}|\mathbf{x})$ = estimate of class probabilities

If $\hat{P}(\mathbf{y}=+1|\mathbf{x}) > 0.5$:
 $\hat{y} = +1$

Else:
 $\hat{y} = -1$

Estimating $\hat{P}(\mathbf{y}|\mathbf{x})$ improves **interpretability**:
 – Predict $\hat{y} = +1$ **and** tell me how sure you are

9

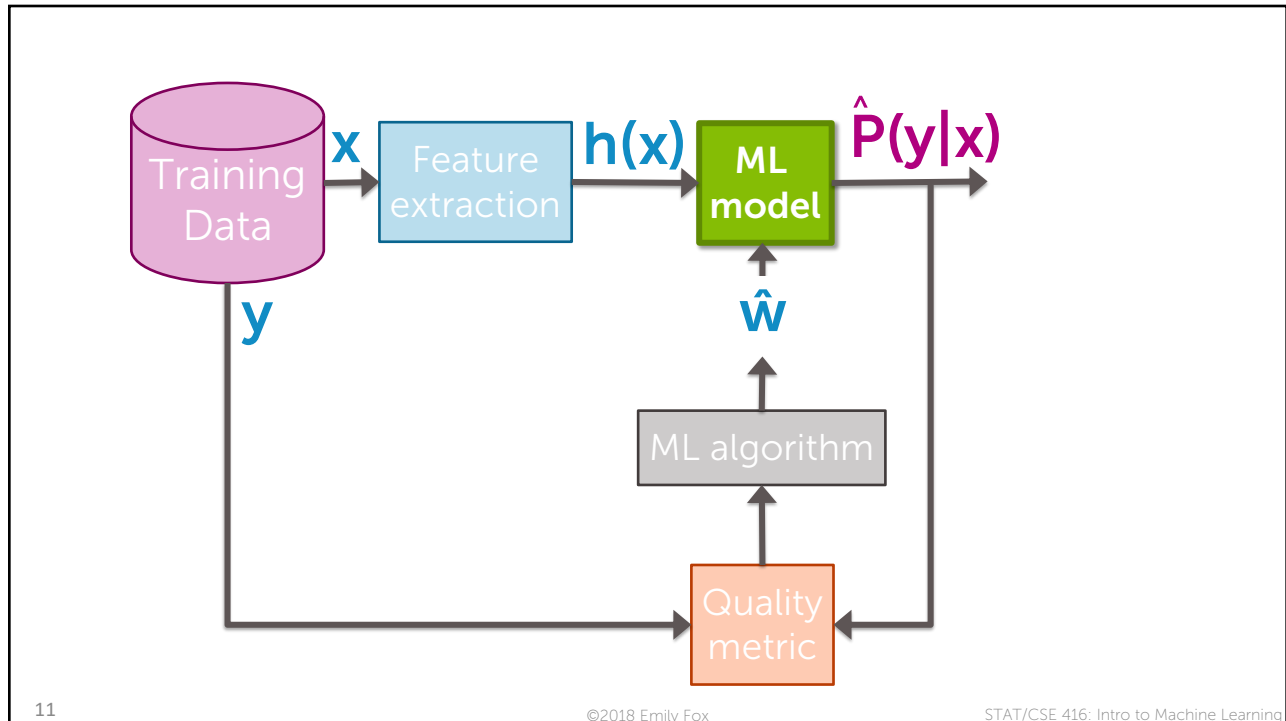
©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Predicting class probabilities with
 logistic regression

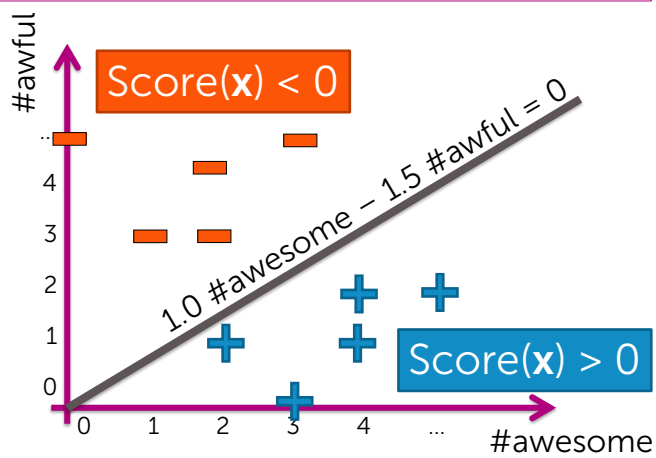
©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning



Thus far, we focused on decision boundaries

$$\begin{aligned} \text{Score}(\mathbf{x}_i) &= w_0 h_0(\mathbf{x}_i) + w_1 h_1(\mathbf{x}_i) + \dots + w_D h_D(\mathbf{x}_i) \\ &= \mathbf{w}^T \mathbf{h}(\mathbf{x}_i) \end{aligned}$$

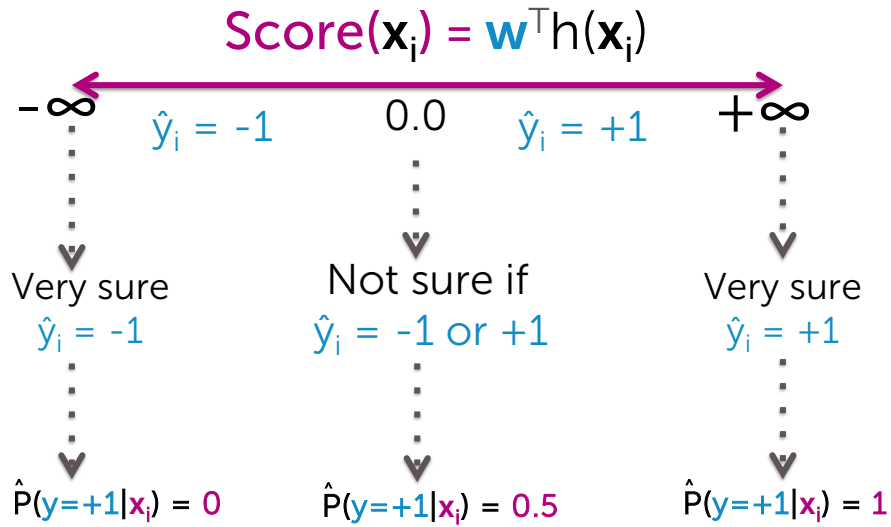


12

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Interpreting $\text{Score}(\mathbf{x}_i)$

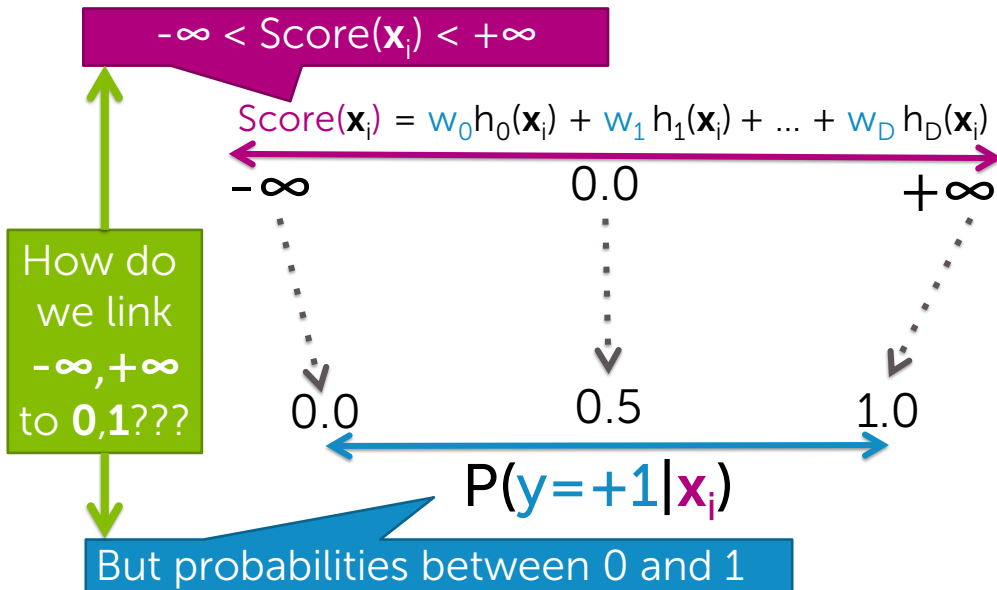


13

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Why not just use regression to build classifier?



14

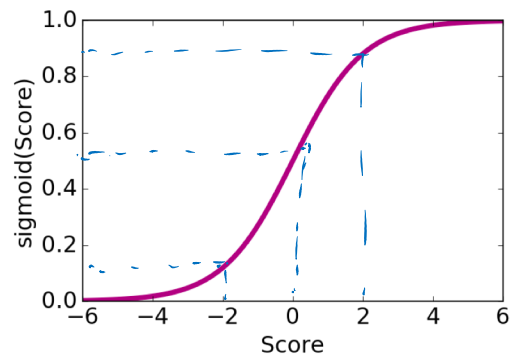
©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Logistic function (sigmoid, logit)

$$\text{sigmoid}(\text{Score}) = \frac{1}{1 + e^{-\text{Score}}}$$

Score	$-\infty$	-2	0.0	+2	$+\infty$
sigmoid(Score)	$\frac{1}{1+e^{\infty}}$ = 0	0.12	$\frac{1}{1+e^0}$ = 0.5	0.88	$\frac{1}{1+e^{-\infty}}$ = 1

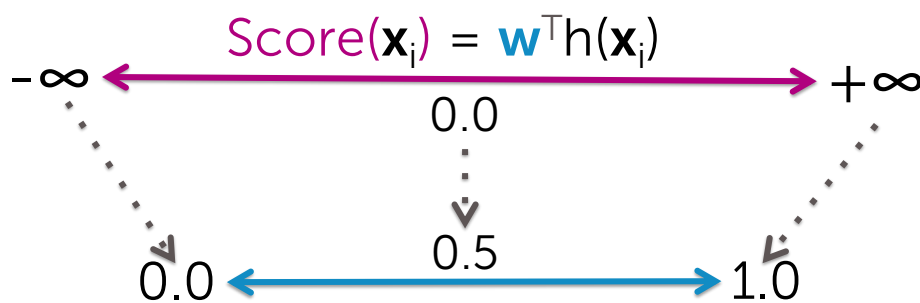


15

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Logistic regression model



$$P(y=+1|x_i, \mathbf{w}) = \text{sigmoid}(\text{Score}(x_i))$$

$$= \frac{1}{1 + e^{-\text{score}(x)}}$$

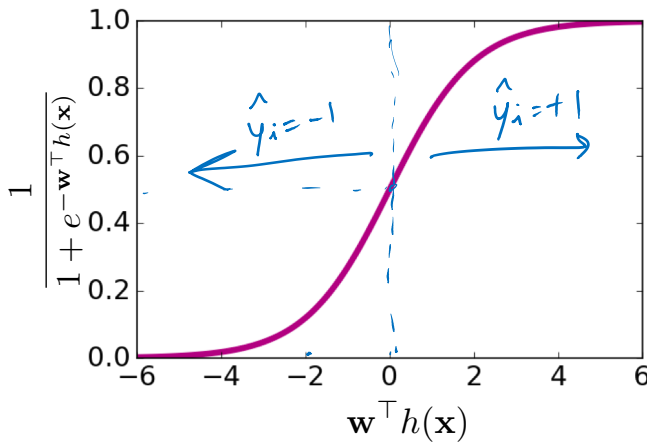
16

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Understanding the logistic regression model

$$P(y=+1|x_i, w) = \text{sigmoid}(\text{Score}(x_i)) = \frac{1}{1 + e^{-w^T h(x)}}$$



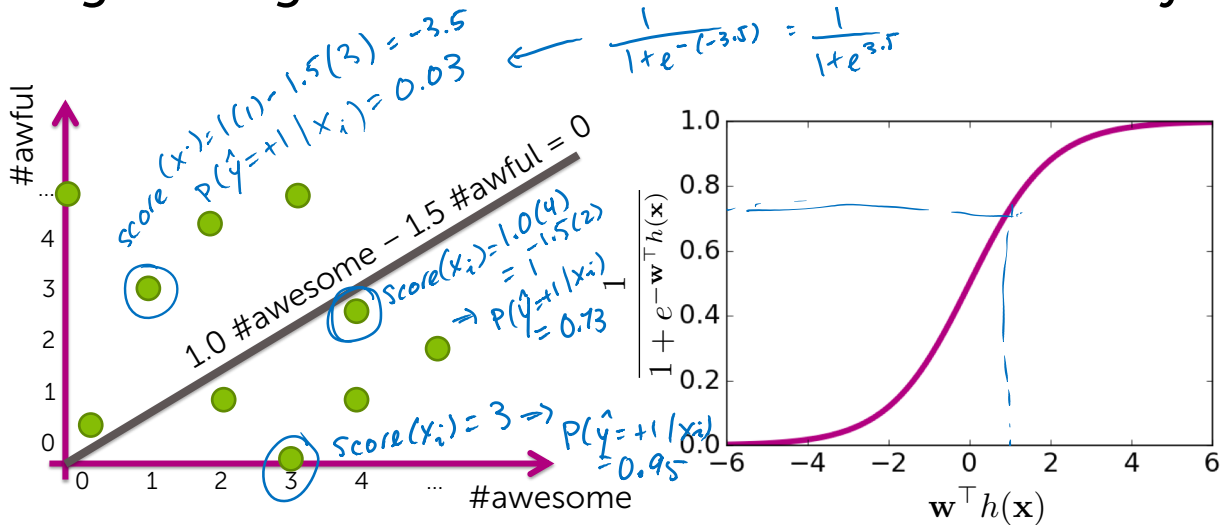
Score(x _i)	P(y=+1 x _i , w)
0	0.5
-2	0.12 < 0.5 ⇒ ŷ _i = -1
2	0.88 > 0.5 ⇒ ŷ _i = +1
4	0.98 > 0.5 ⇒ ŷ _i = +1

17

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Logistic regression → Linear decision boundary

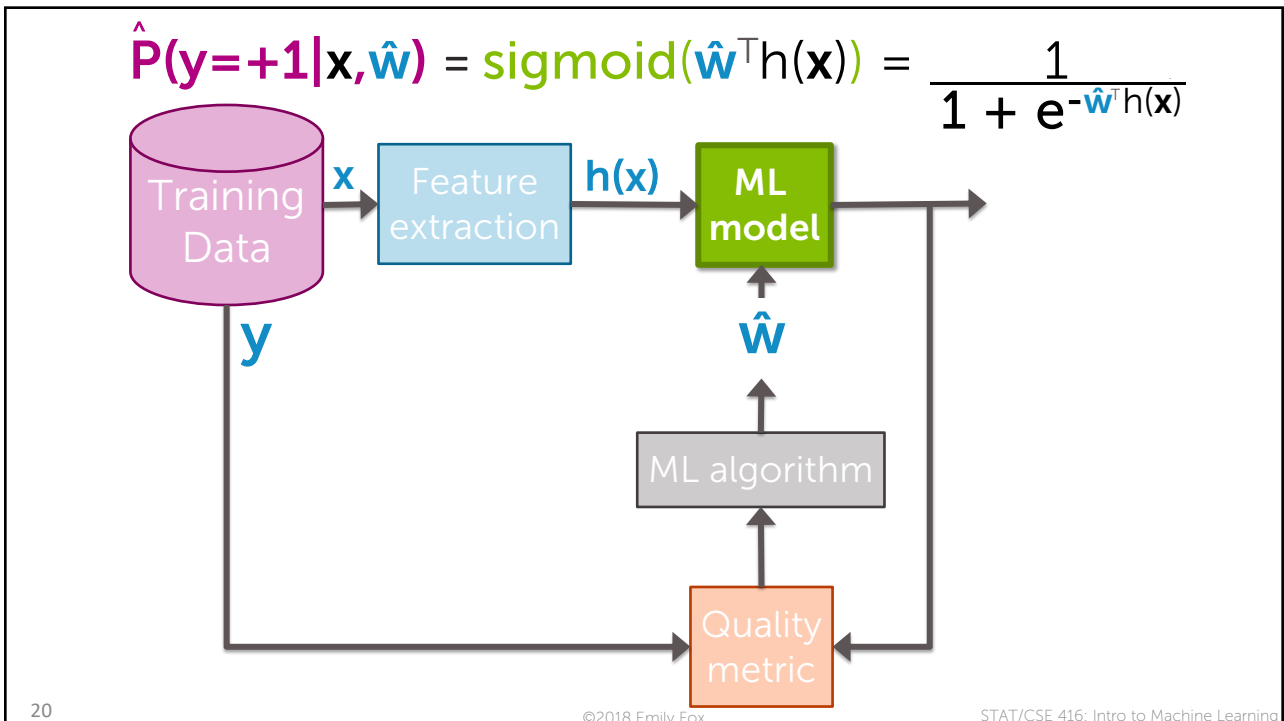
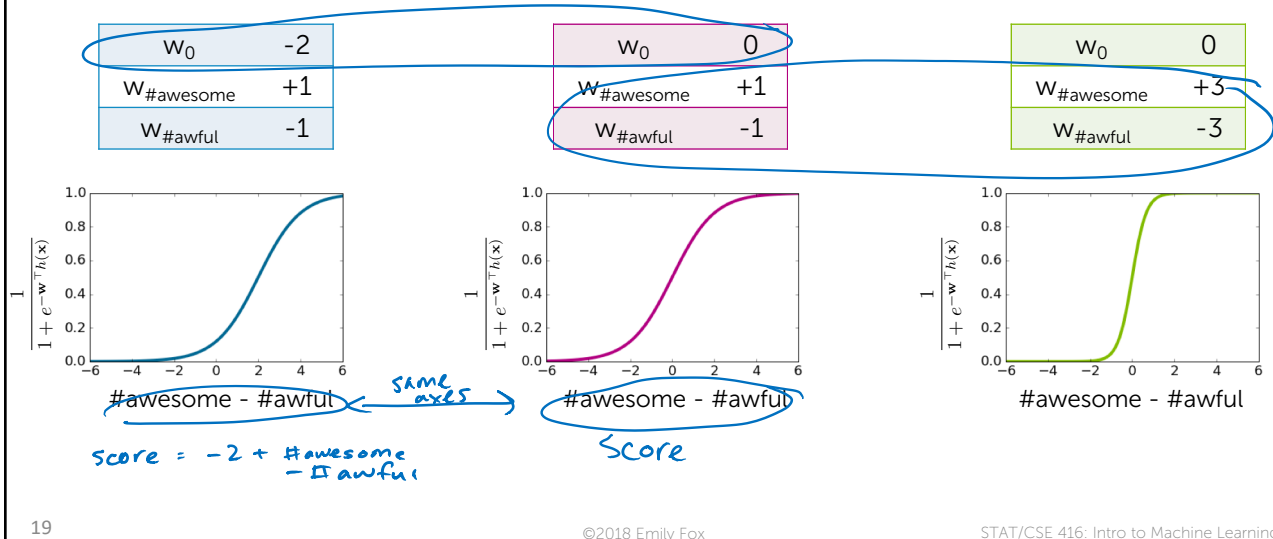


18

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

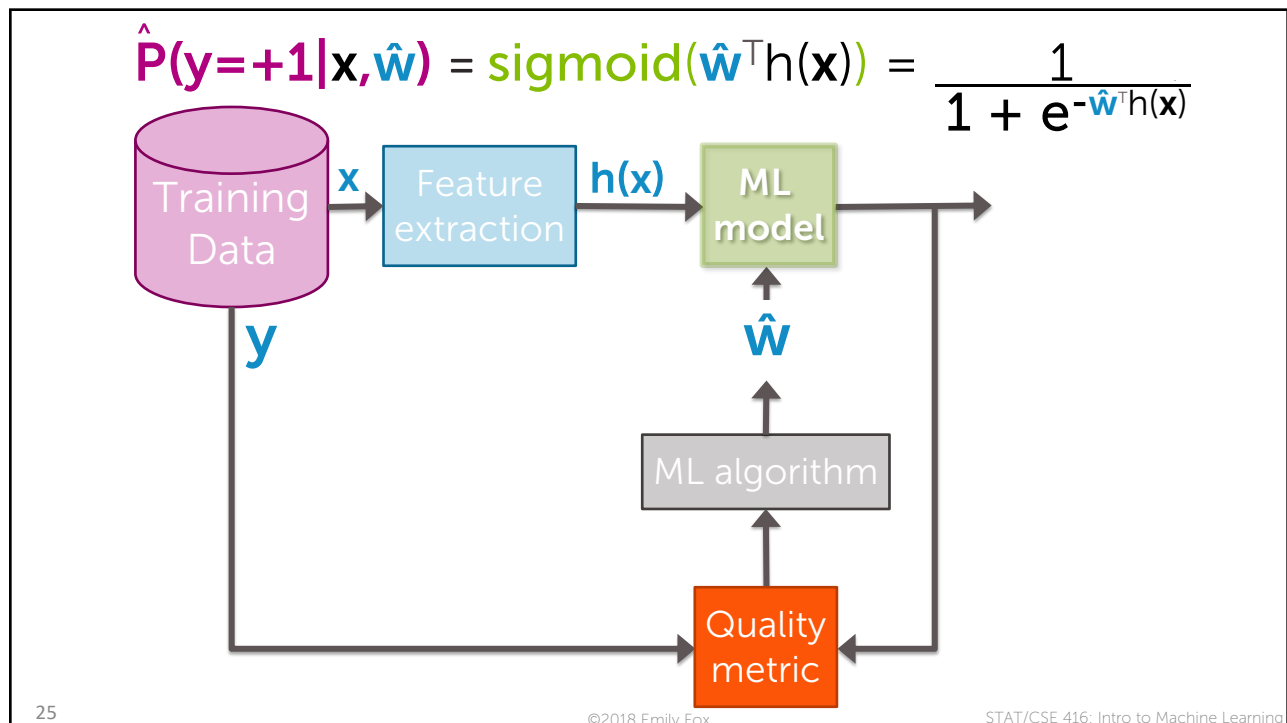
Effect of coefficients on logistic regression model



Quality metric for logistic regression: Maximum likelihood estimation

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning



Learning a (linear) classifier

Will use training data to learn a weight for each word

Word	Coefficient	Value
	\hat{w}_0	-2.0
good	\hat{w}_1	1.0
great	\hat{w}_2	1.5
awesome	\hat{w}_3	2.7
bad	\hat{w}_4	-1.0
terrible	\hat{w}_5	-2.1
awful	\hat{w}_6	-3.3
restaurant, the, we, ...	$\hat{w}_7, \hat{w}_8, \hat{w}_9, \dots$	0.0
...		...

26

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Learning problem

Training data:

N observations (\mathbf{x}_i, y_i)

$x[1] = \text{\#awesome}$	$x[2] = \text{\#awful}$	$y = \text{sentiment}$
2	1	+1
0	2	-1
3	3	-1
4	1	+1
1	1	+1
2	4	-1
0	3	-1
0	1	-1
2	1	+1

Optimize
quality metric
on training
data

 \hat{w}

27

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Finding best coefficients

x[1] = #awesome	x[2] = #awful	y = sentiment
2	1	+1
0	2	-1
3	3	-1
4	1	+1
1	1	+1
2	4	-1
0	3	-1
0	1	-1
2	1	+1

28

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Finding best coefficients

x[1] = #awesome	x[2] = #awful	y = sentiment
0	2	-1
3	3	-1
2	4	-1
0	3	-1
0	1	-1
2	4	-1
0	3	-1
0	1	-1

x[1] = #awesome	x[2] = #awful	y = sentiment
2	1	+1
4	1	+1
1	1	+1
2	1	+1
1	1	+1
2	1	+1

29

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Finding best coefficients

x[1] = #awesome	x[2] = #awful	y = sentiment
0	2	-1
3	3	-1
2	4	-1
0	3	-1
0	1	-1

$$P(y=+1|\mathbf{x}_i, \mathbf{w}) = 0.0$$

x[1] = #awesome	x[2] = #awful	y = sentiment
2	1	+1
4	1	+1
1	1	+1
2	1	+1

$$P(y=+1|\mathbf{x}_i, \mathbf{w}) = 1.0$$

Want $\hat{\mathbf{w}}$ that makes

30

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Quality metric = Likelihood function

Negative data points

Positive data points

$$P(y=+1|\mathbf{x}_i, \mathbf{w}) = 0.0$$

$$P(y=+1|\mathbf{x}_i, \mathbf{w}) = 1.0$$

No $\hat{\mathbf{w}}$ achieves perfect predictions (usually)

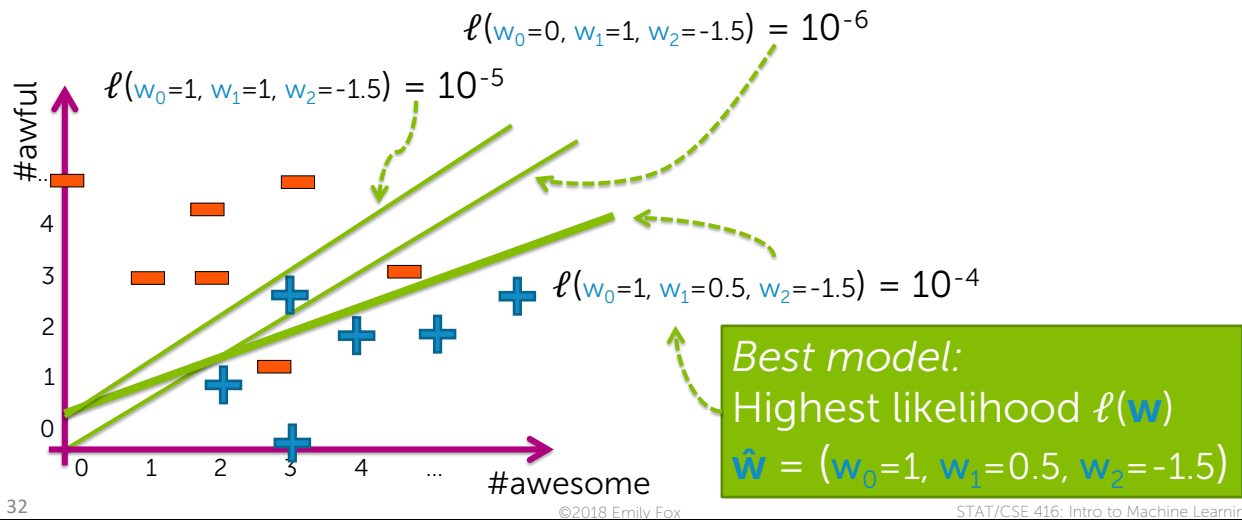
Likelihood $\ell(\mathbf{w})$: Measures quality of fit for model with coefficients \mathbf{w}

31

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Find "best" classifier =
 Maximize quality metric over all possible w_0, w_1, w_2
 Likelihood $\ell(\mathbf{w})$



32

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Data likelihood

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Fitting to individual data points

x[1] = #awesome	x[2] = #awful	y = sentiment
2	1	+1

If model good, should predict:
 $\hat{y}_i = +1$

Pick w to maximize:
 $P(\hat{y}_i = +1 | x_i, w) = P(\hat{y}_i = +1 | x[1]=2, x[2]=1, w)$

x[1] = #awesome	x[2] = #awful	y = sentiment
0	2	-1

If model good, should predict:
 $\hat{y}_i = -1$

Pick w to maximize:
 $P(\hat{y}_i = -1 | x_i, w)$

34

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Learn logistic regression model with maximum likelihood estimation (MLE)

Data point	x[1]	x[2]	y	Choose w to maximize
x_1, y_1	2	1	+1	$P(y=+1 x[1]=2, x[2]=1, w)$
x_2, y_2	0	2	-1	$P(y=-1 x[1]=0, x[2]=2, w)$
x_3, y_3	3	3	-1	$P(y=-1 x[1]=3, x[2]=3, w)$
x_4, y_4	4	1	+1	$P(y=+1 x[1]=4, x[2]=1, w)$

$$P(y=+1 | x, w) = \frac{1}{1 + e^{-w^T h(x)}}$$

$$P(y=-1 | x, w) = \frac{e^{-w^T h(x)}}{1 + e^{-w^T h(x)}}$$

$$P(y | x, w) = \begin{cases} \frac{1}{1 + e^{-w^T h(x)}} & \text{if } y=+1 \\ \frac{e^{-w^T h(x)}}{1 + e^{-w^T h(x)}} & \text{if } y=-1 \end{cases}$$

$$\ell(w) = P(y_1 | x_1, w) P(y_2 | x_2, w) P(y_3 | x_3, w) P(y_4 | x_4, w)$$

$$\ell(w) = \prod_{i=1}^N P(y_i | x_i, w)$$

total # obs N

pick w to make this as large as possible

35

©2018 Emily Fox

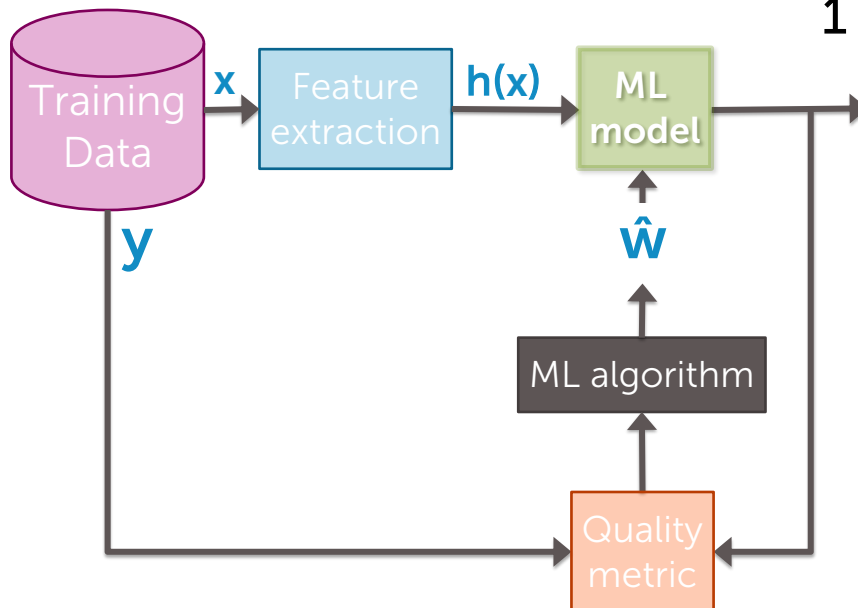
STAT/CSE 416: Intro to Machine Learning

Fitting logistic regression models

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

$$\hat{P}(y=+1|\mathbf{x}, \hat{\mathbf{w}}) = \text{sigmoid}(\hat{\mathbf{w}}^T \mathbf{h}(\mathbf{x})) = \frac{1}{1 + e^{-\hat{\mathbf{w}}^T \mathbf{h}(\mathbf{x})}}$$



37

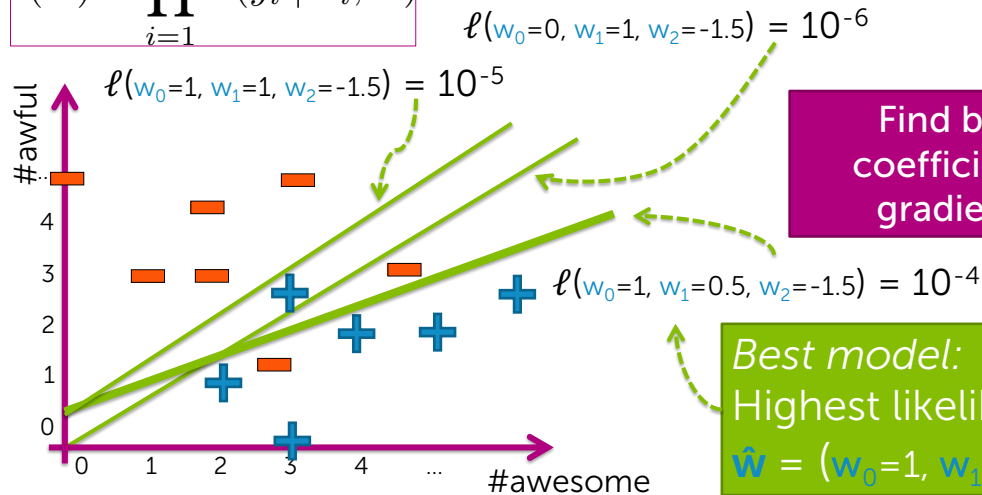
©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Find "best" classifier

Maximize likelihood over all possible w_0, w_1, w_2

$$\ell(\mathbf{w}) = \prod_{i=1}^N P(y_i | \mathbf{x}_i, \mathbf{w})$$

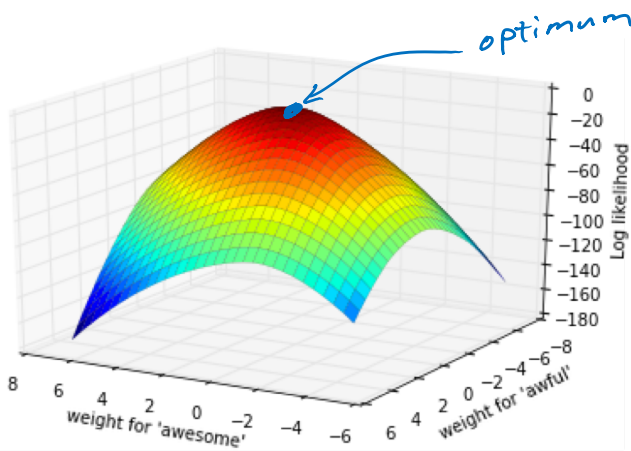


38

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Maximizing likelihood



No closed-form solution \rightarrow use gradient ascent

Maximize function over all possible w_0, w_1, w_2

$$\max_{w_0, w_1, w_2} \prod_{i=1}^N P(y_i | \mathbf{x}_i, \mathbf{w})$$

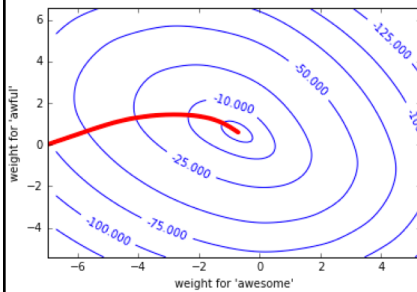
$\ell(w_0, w_1, w_2)$ is a function of 3 variables

39

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Gradient ascent for logistic regression



init $\mathbf{w}^{(1)} = 0$ (or randomly, or smartly), $t = 1$

while $\|\nabla \ell(\mathbf{w}^{(t)})\| > \epsilon$ \leftarrow *small threshold*
 Difference between truth and prediction

for $j = 0, \dots, D$

$$\text{partial}[j] = \sum_{i=1}^N h_j(\mathbf{x}_i) \left(\mathbb{1}[y_i = +1] - P(y = +1 | \mathbf{x}_i, \mathbf{w}^{(t)}) \right)$$

$$\mathbf{w}_j^{(t+1)} \leftarrow \mathbf{w}_j^{(t)} + \eta \text{partial}[j]$$

$t \leftarrow t + 1$

$$\mathbb{1}(y_i = +1) = \begin{cases} 1 & \text{if } y_i = +1 \\ 0 & \text{otherwise} \end{cases}$$

40

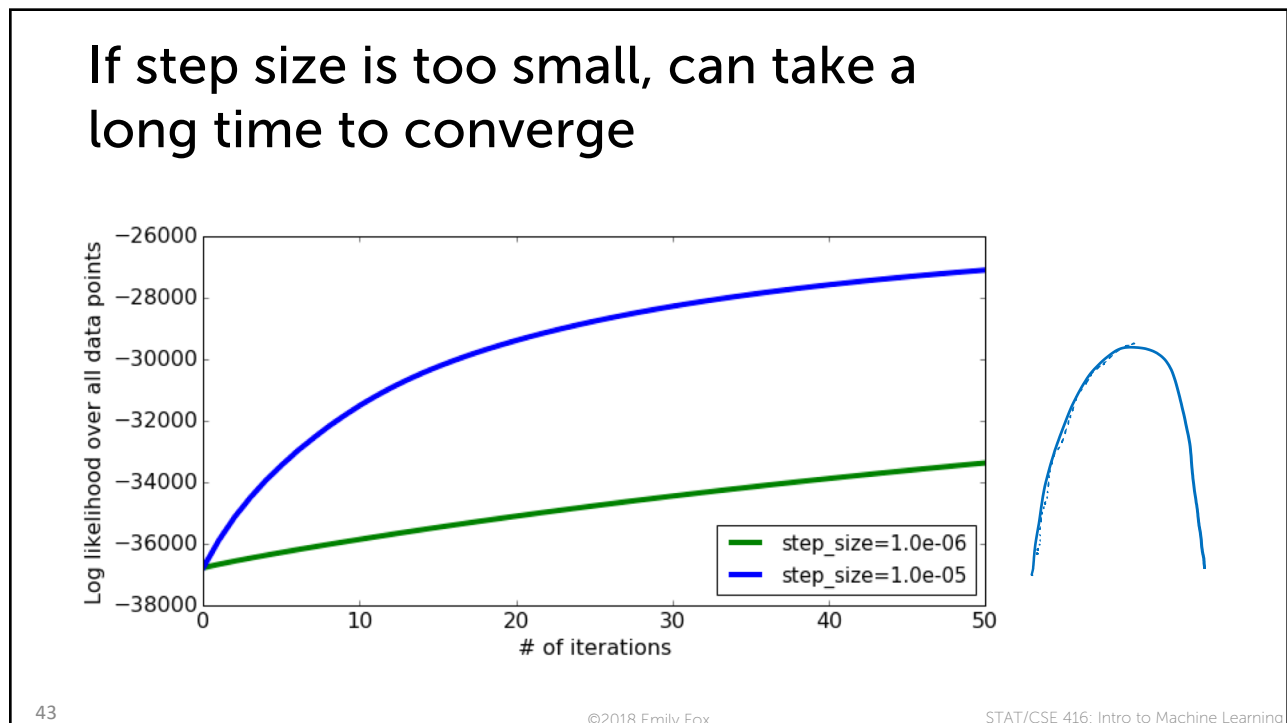
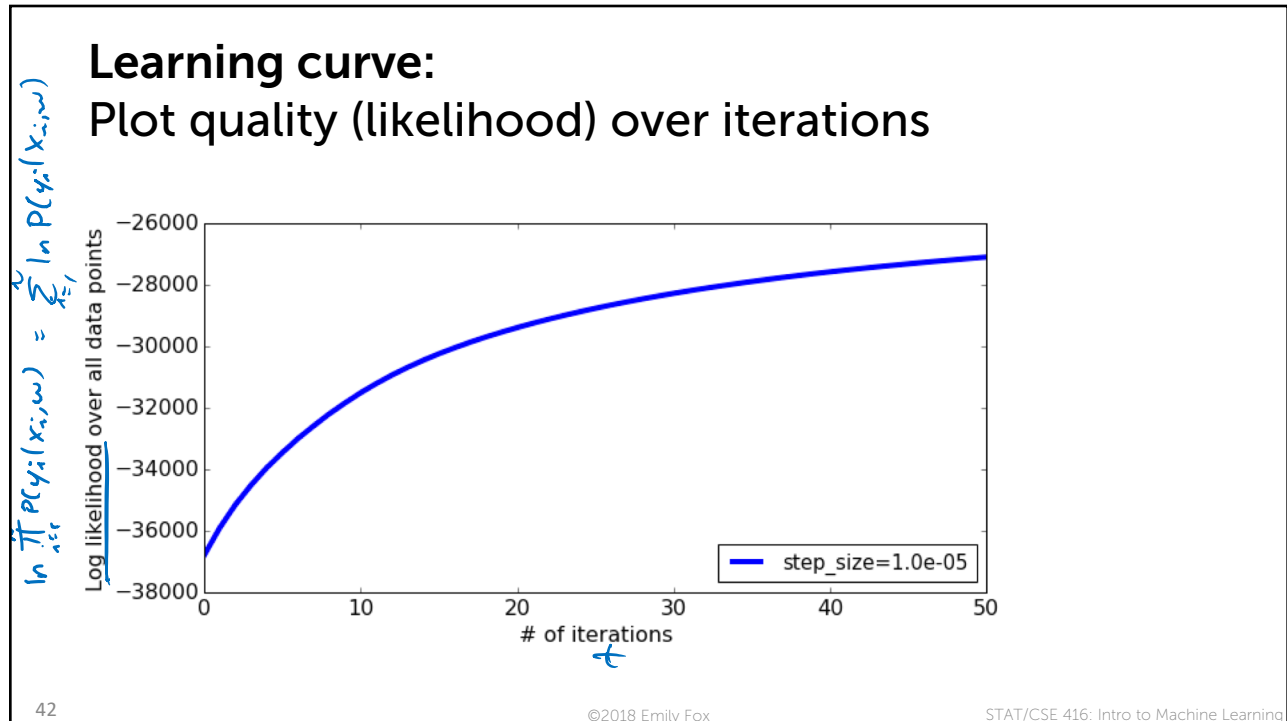
©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

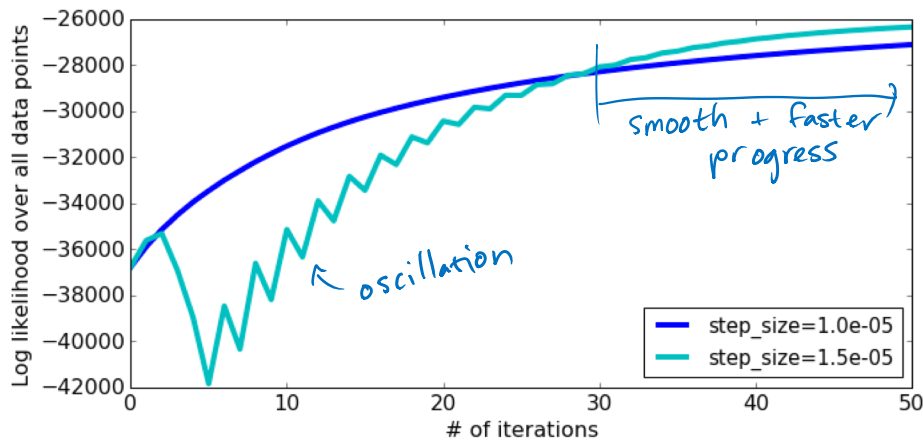
Choosing the step size η

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning



Compare converge with different step sizes

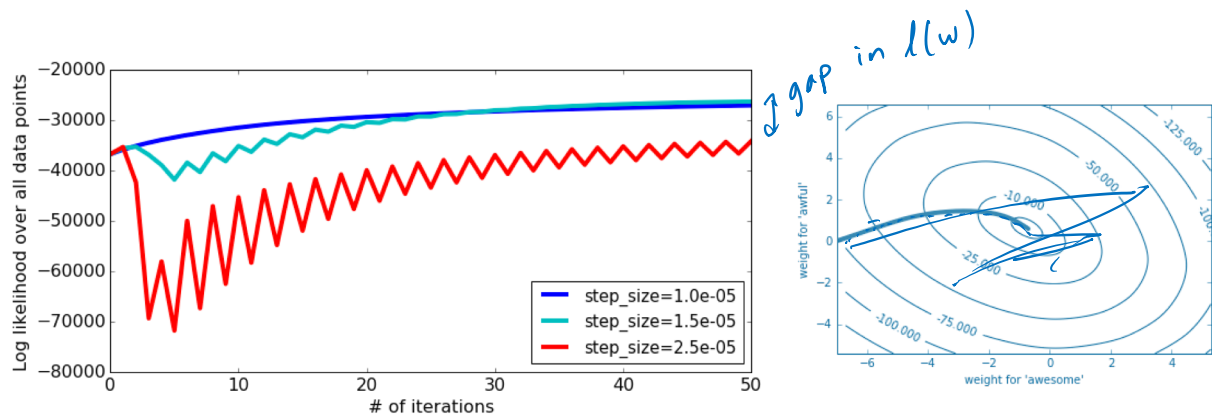


44

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Careful with step sizes that are too large

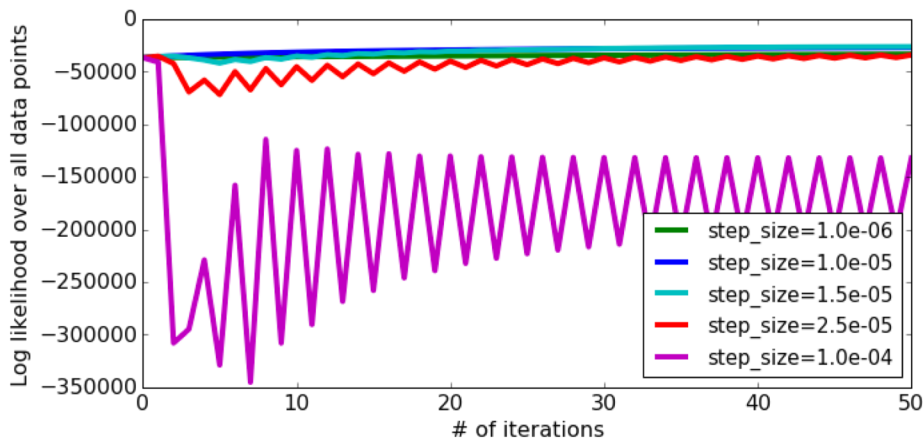


45

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Very large step sizes can even cause divergence or wild oscillations



46

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Simple rule of thumb for picking step size η

- Unfortunately, picking step size requires a lot of trial & error ☹️
- Try a several values, exponentially spaced
 - **Goal:** plot learning curves to
 - find one η that is too small (smooth but moving too slowly)
 - find one η that is too large (oscillation or divergence)
- Try values in between to find “best” η
- *Advanced tip:* can also try step size that decreases with iterations, e.g.,

$$\eta_t = \frac{\eta_0}{t}$$



47

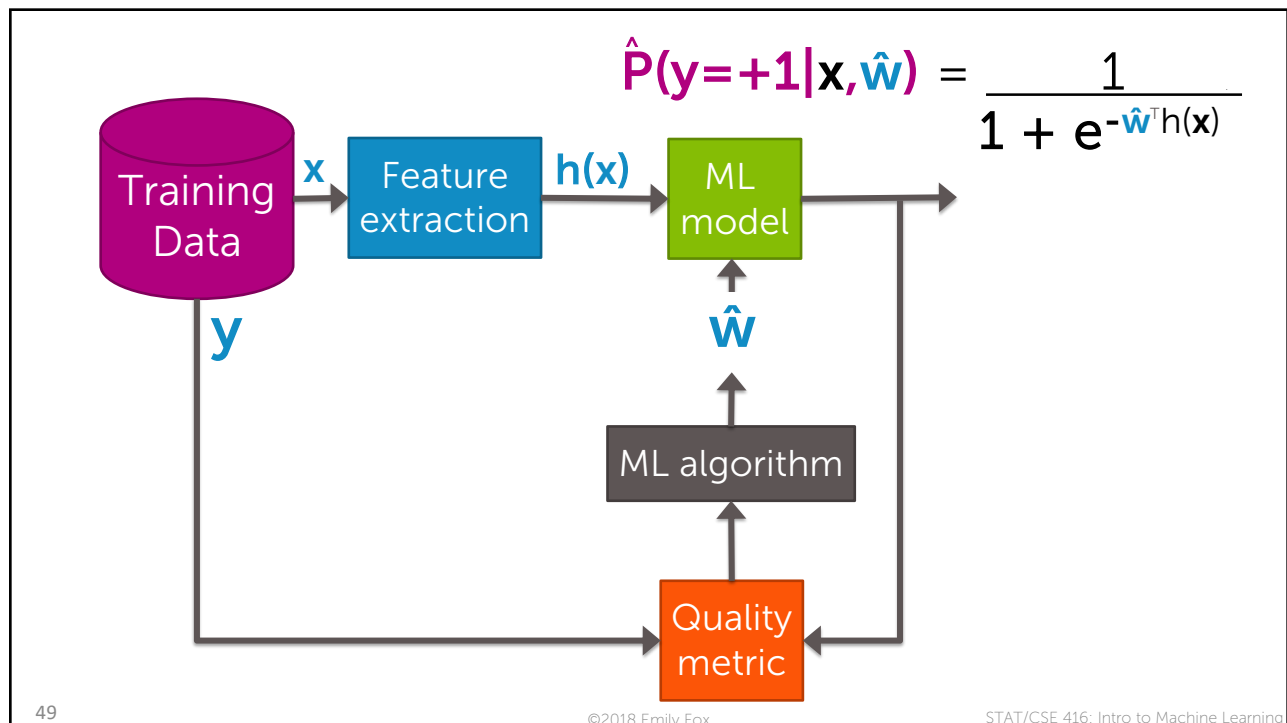
©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Summary of logistic regression classifier

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning



49

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

What you can do now...

- Use class probability to express degree of confidence in prediction
- Define a logistic regression model
- Interpret logistic regression outputs as class probabilities
- Describe impact of coefficient values on logistic regression output
- Measure quality of a classifier using the likelihood function
- Optimize resulting objective using gradient descent

50

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning



Linear classifiers:

Handling overfitting, categorical inputs, & multiple classes

STAT/CSE 416: Machine Learning
Emily Fox
University of Washington
April 19, 2018

©2018 Emily Fox

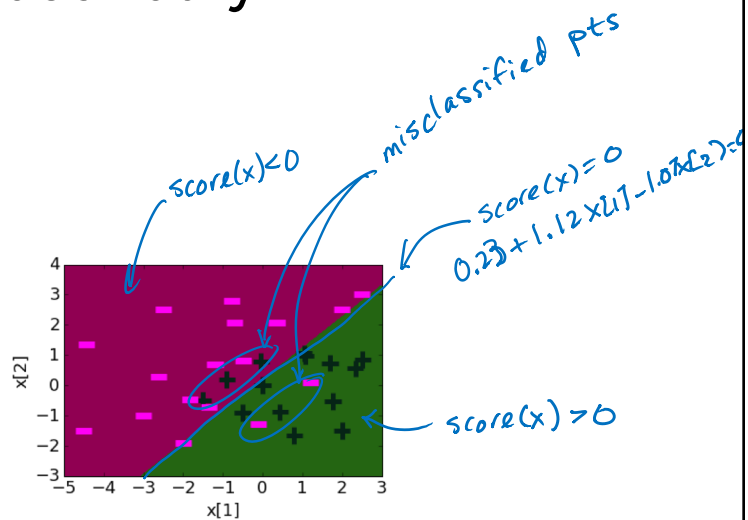
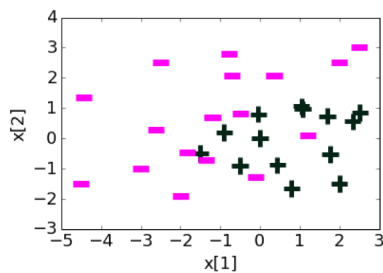
Overfitting in classification

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Learned decision boundary

Feature	Value	Coefficient learned
$h_0(\mathbf{x})$	1	0.23
$h_1(\mathbf{x})$	$x[1]$	1.12
$h_2(\mathbf{x})$	$x[2]$	-1.07



53

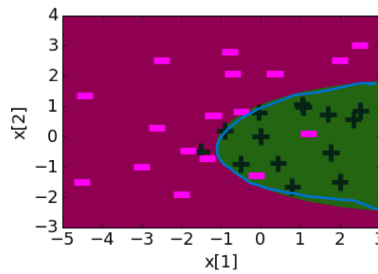
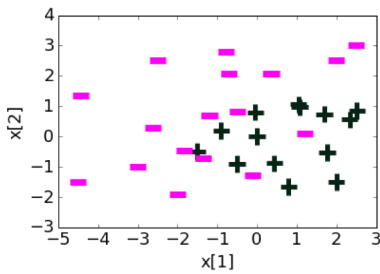
©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Quadratic features (in 2d)

Note: we are not including cross terms for simplicity

Feature	Value	Coefficient learned
$h_0(\mathbf{x})$	1	1.68
$h_1(\mathbf{x})$	$x[1]$	1.39
$h_2(\mathbf{x})$	$x[2]$	-0.59
$h_3(\mathbf{x})$	$(x[1])^2$	-0.17
$h_4(\mathbf{x})$	$(x[2])^2$	-0.96



$$1.68 + 1.39x[1] - 0.59x[2] - 0.17(x[1])^2 - 0.96(x[2])^2 = 0$$

54

©2018 Emily Fox

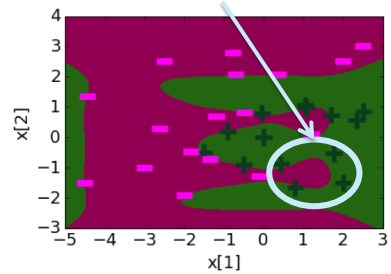
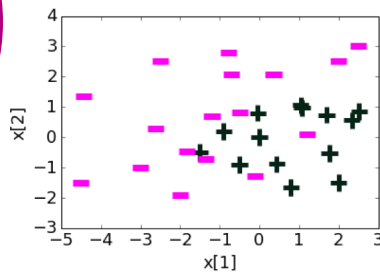
STAT/CSE 416: Intro to Machine Learning

Degree 6 features (in 2d)

Note: we are not including cross terms for simplicity

Feature	Value	Coefficient learned
$h_0(\mathbf{x})$	1	21.6
$h_1(\mathbf{x})$	$x[1]$	5.3
$h_2(\mathbf{x})$	$x[2]$	-42.7
$h_3(\mathbf{x})$	$(x[1])^2$	-15.9
$h_4(\mathbf{x})$	$(x[2])^2$	-48.6
$h_5(\mathbf{x})$	$(x[1])^3$	-11.0
$h_6(\mathbf{x})$	$(x[2])^3$	67.0
$h_7(\mathbf{x})$	$(x[1])^4$	1.5
$h_8(\mathbf{x})$	$(x[2])^4$	48.0
$h_9(\mathbf{x})$	$(x[1])^5$	4.4
$h_{10}(\mathbf{x})$	$(x[2])^5$	-14.2
$h_{11}(\mathbf{x})$	$(x[1])^6$	0.8
$h_{12}(\mathbf{x})$	$(x[2])^6$	-8.6

Coefficient values getting large



Score(x) < 0



55

©2018 Emily Fox

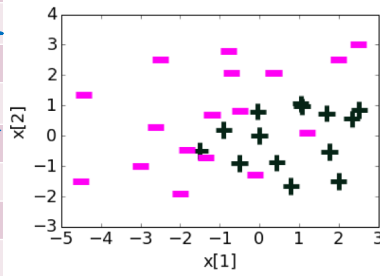
STAT/CSE 416: Intro to Machine Learning

Degree 20 features (in 2d)

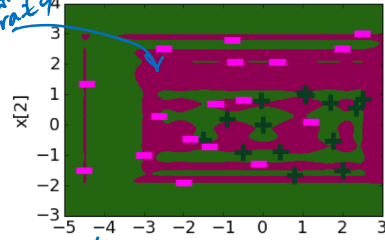
Note: we are not including cross terms for simplicity

Feature	Value	Coefficient learned
$h_0(\mathbf{x})$	1	8.7
$h_1(\mathbf{x})$	$x[1]$	5.1
$h_2(\mathbf{x})$	$x[2]$	78.7
...
$h_{11}(\mathbf{x})$	$(x[1])^6$	-7.5
$h_{12}(\mathbf{x})$	$(x[2])^6$	3803
$h_{13}(\mathbf{x})$	$(x[1])^7$	21.1
$h_{14}(\mathbf{x})$	$(x[2])^7$	-2406
...
$h_{37}(\mathbf{x})$	$(x[1])^{19}$	$-2 \cdot 10^{-6}$
$h_{38}(\mathbf{x})$	$(x[2])^{19}$	-0.15
$h_{39}(\mathbf{x})$	$(x[1])^{20}$	$-2 \cdot 10^{-8}$
$h_{40}(\mathbf{x})$	$(x[2])^{20}$	0.03

Often, overfitting associated with very large estimated coefficients \hat{w}



really crazy

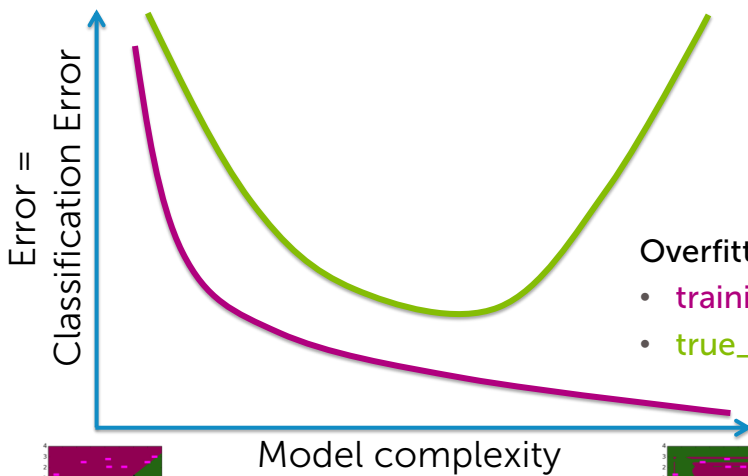


extremely complex decision boundary

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Overfitting in classification



Overfitting if there exists w^* :

- $\text{training_error}(w^*) > \text{training_error}(\hat{w})$
- $\text{true_error}(w^*) < \text{true_error}(\hat{w})$

57

©2018 Emily Fox

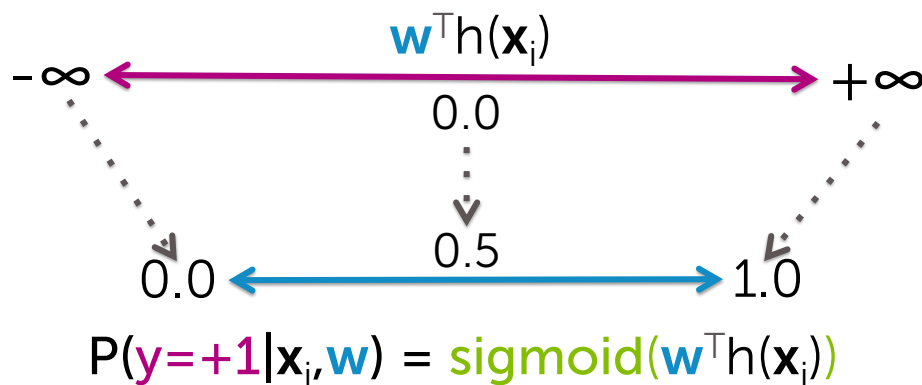
STAT/CSE 416: Intro to Machine Learning

Overfitting in classifiers →
Overconfident predictions

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Logistic regression model



59

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

The subtle (negative) consequence of overfitting in logistic regression

Overfitting → Large coefficient values

$\hat{w}^T \mathbf{x}_i$ is very positive (or very negative) →
sigmoid($\hat{w}^T \mathbf{x}_i$) goes to 1 (or to 0)

Model becomes extremely overconfident of predictions

60

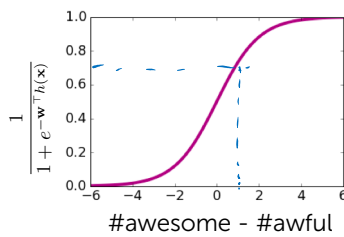
©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

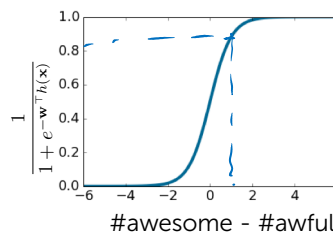
Effect of coefficients on logistic regression model

Input \mathbf{x} : #awesome=2, #awful=1

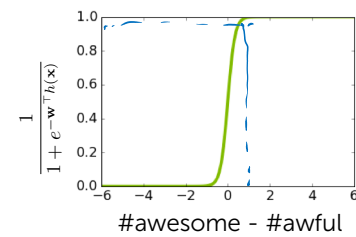
w_0	0
$w_{\#awesome}$	+1
$w_{\#awful}$	-1



w_0	0
$w_{\#awesome}$	+2
$w_{\#awful}$	-2



w_0	0
$w_{\#awesome}$	+6
$w_{\#awful}$	-6



61

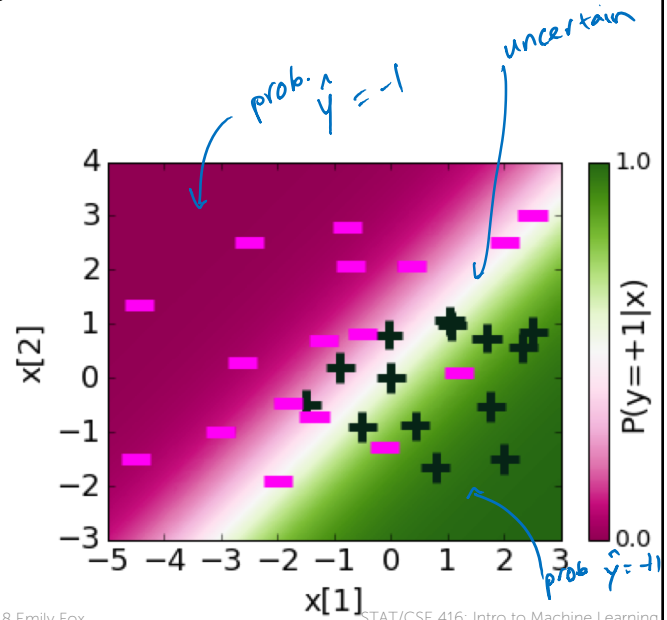
©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Learned probabilities

Feature	Value	Coefficient learned
$h_0(\mathbf{x})$	1	0.23
$h_1(\mathbf{x})$	$\mathbf{x}[1]$	1.12
$h_2(\mathbf{x})$	$\mathbf{x}[2]$	-1.07

$$P(y = +1 | \mathbf{x}, \mathbf{w}) = \frac{1}{1 + e^{-\mathbf{w}^\top \mathbf{h}(\mathbf{x})}}$$



62

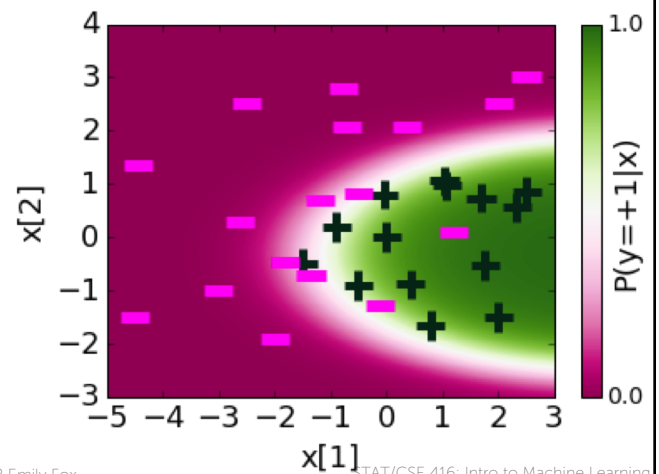
©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Quadratic features: Learned probabilities

Feature	Value	Coefficient learned
$h_0(\mathbf{x})$	1	1.68
$h_1(\mathbf{x})$	$\mathbf{x}[1]$	1.39
$h_2(\mathbf{x})$	$\mathbf{x}[2]$	-0.58
$h_3(\mathbf{x})$	$(\mathbf{x}[1])^2$	-0.17
$h_4(\mathbf{x})$	$(\mathbf{x}[2])^2$	-0.96

$$P(y = +1 | \mathbf{x}, \mathbf{w}) = \frac{1}{1 + e^{-\mathbf{w}^\top \mathbf{h}(\mathbf{x})}}$$



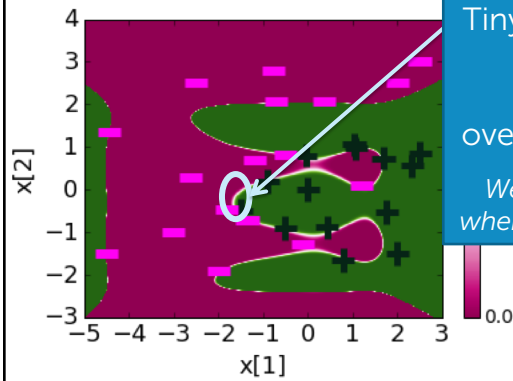
63

©2018 Emily Fox

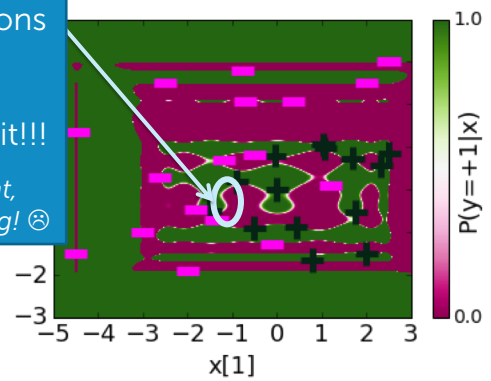
STAT/CSE 416: Intro to Machine Learning

Overfitting → Overconfident predictions

Degree 6: Learned probabilities



Degree 20: Learned probabilities



Tiny uncertainty regions



Overfitting &
overconfident about it!!!

*We are sure we are right,
when we are surely wrong! ☹*

64

©2018 Emily Fox

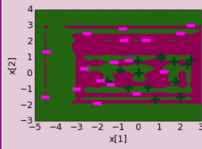
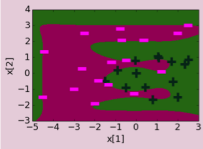
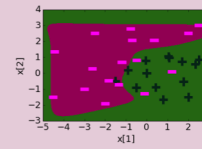
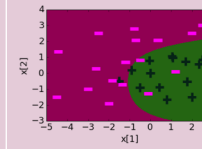
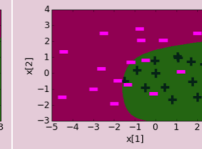
STAT/CSE 416: Intro to Machine Learning

Penalizing large coefficients
to mitigate overfitting

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Degree 20 features, effect of regularization penalty λ

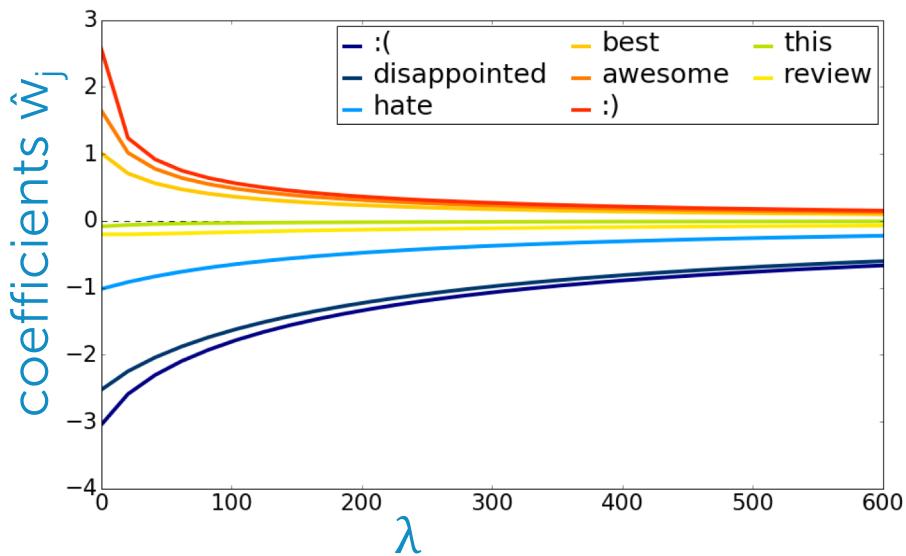
Regularization λ	$\lambda = 0$	$\lambda = 0.00001$	$\lambda = 0.001$	$\lambda = 1$	$\lambda = 10$
Range of coefficients	-3170 to 3803	-8.04 to 12.14	-0.70 to 1.25	-0.13 to 0.57	-0.05 to 0.22
Decision boundary					

68

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Coefficient path – L_2 penalty

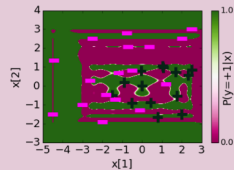
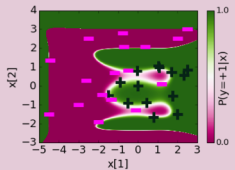
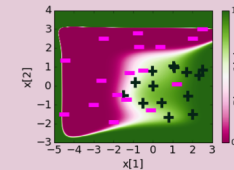
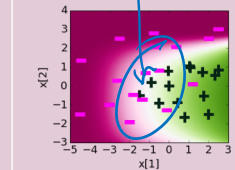


69

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Degree 20 features:
regularization reduces "overconfidence"

Regularization	$\lambda = 0$	$\lambda = 0.00001$	$\lambda = 0.001$	$\lambda = 1$
Range of coefficients	-3170 to 3803	-8.04 to 12.14	-0.70 to 1.25	-0.13 to 0.57
Learned probabilities				

reasonable uncertainty

70

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Sparse logistic regression with L_1 penalty

Select $\hat{\mathbf{w}}$ to maximize:

$$\ell(\mathbf{w}) - \lambda \|\mathbf{w}\|_1$$

λ tuning parameter = balance of fit and magnitude

L_1 regularized
logistic regression

Pick λ using:

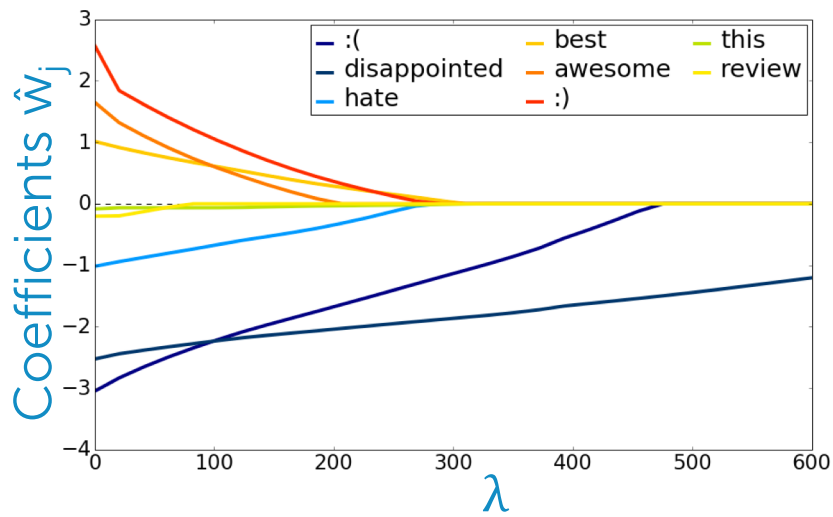
- Validation set (for large datasets)
- Cross-validation (for smaller datasets)
(as in ridge/lasso regression)

71

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning

Coefficient path – L_1 penalty



72

©2018 Emily Fox

STAT/CSE 416: Intro to Machine Learning