6. 모델 평가, 편향성 및 분산 진단

Hunmin DO
gnsals9262@g.skku.edu
TNT ML Team
2024/09/26

TRAIN AND TEST

Contents

- Evaluating a Model
- Model selection(Training/Cross Validation/Test sets)
- Diagnosing bias and variance
- Learning curves
- Iterative loop of ML development
- Error analysis
- Adding Data
- Transfer learning
- Fairness, bias and ethics.

Evaluating a Model

Debugging a learning algorithm

You've implemented regularized linear regression on housing prices

$$J(\vec{w}, b) = \frac{1}{2m}\sum_{i=1}^{m}\left(f_{\vec{w},b}\left(\vec{x}^{(i)}\right) - y^{(i)}\right)^2 + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2$$

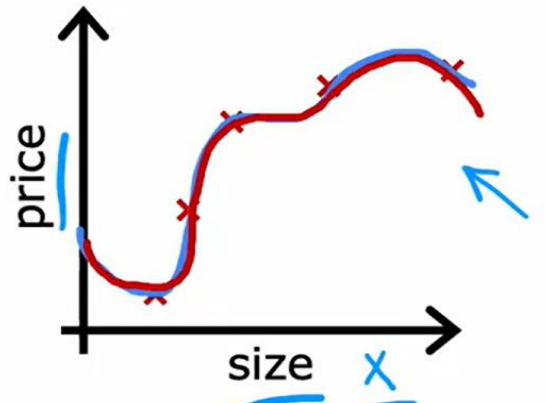
But it makes unacceptably large errors in predictions. What do you try next?

- Get more training examples
- Try smaller sets of features $x, x^2, x^3, x^4, x^5 \ldots$
- Try getting additional features
- Try adding polynomial features $(x_1^2, x_2^2, x_1 x_2, etc)$
- Try decreasing λ
- Try increasing λ

TRAIN AND TEST

Evaluating a Model(regression/classification)

Evaluating your model



Model fits the training data well but will fail to generalize to new examples not in the training set.

$x = x_1$

$$f_{\vec{w},b}(\vec{x}) = w_1 x + w_2 x^2 + \cdots + w_n x^n + b$$

$w_4 x^4$

just x

- x_1 = size in feet^2
- x_2 = no. of bedrooms
- x_3 = no. of floors
- x_4 = age of home in years

$f(\vec{x})$

TRAIN AND TEST

Evaluating a Model(regression/classification)

Evaluating your model

Dataset:

size	price
2104	400
1600	330
2400	369
1416	232
3000	540
1985	300
1534	315
1427	199
1380	212
1494	243

70%

30%

training set → $(x^{(1)}, y^{(1)})$, $(x^{(2)}, y^{(2)})$, ⋮, $(x^{(m_{train})}, y^{(m_{train})})$

m_{train} = no. training examples

=7

test set → $(x_{test}^{(1)}, y_{test}^{(1)})$, ⋮, $(x_{test}^{(m_{test})}, y_{test}^{(m_{test})})$

m_{test} = no. test examples

=3

TRAIN AND TEST

Evaluating a Model(regression/classification)

Train/test procedure for linear regression (with squared error cost)

Fit parameters by minimizing cost function $J(\vec{w}, b)$

$$J(\vec{w}, b) = \left[\frac{1}{2m_{train}} \sum_{i=1}^{m_{train}} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right)^2 + \frac{\lambda}{2m_{train}} \sum_{j=1}^{n} w_j^2 \right]$$

Compute test error:

$$J_{test}(\vec{w}, b) = \frac{1}{2m_{test}} \left[\sum_{i=1}^{m_{test}} \left(f_{\vec{w},b}\left(\vec{x}_{test}^{(i)}\right) - y_{test}^{(i)} \right)^2 \right]$$

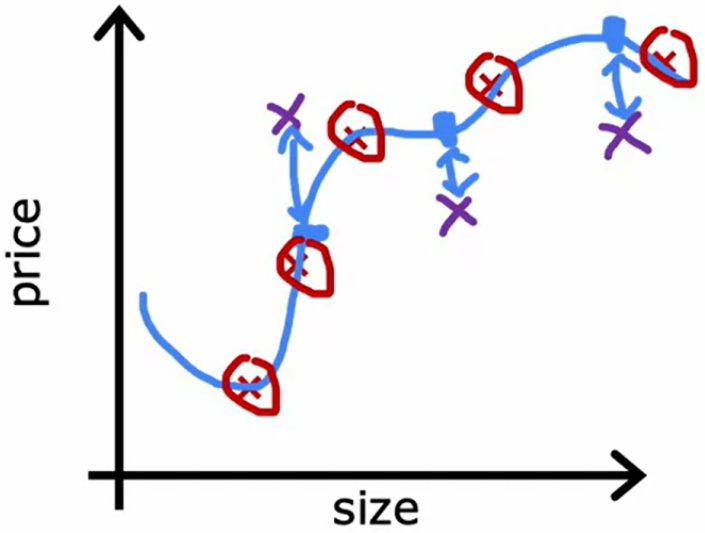
Compute training error:

$$J_{train}(\vec{w}, b) = \frac{1}{2m_{train}} \left[\sum_{i=1}^{m_{train}} \left(f_{\vec{w},b}\left(\vec{x}_{train}^{(i)}\right) - y_{train}^{(i)} \right)^2 \right]$$

TRAIN AND TEST

Evaluating a Model(regression/classification)

Train/test procedure for linear regression (with squared error cost)



X = train

X = test

$J_{train}(\vec{w}, b)$ will be low

$J_{test}(\vec{w}, b)$ will be high

TRAIN AND TEST

Evaluating a Model(regression/classification)

Train/test procedure for classification problem

0/1

Fit parameters by minimizing $J(\vec{w}, b)$ to find $\vec{w}, b$

E.g.,

$$J(\vec{w}, b) = -\frac{1}{m_{train}} \sum_{i=1}^{m_{train}} \left[y^{(i)} \log\left(f_{\vec{w},b}(\vec{x}^{(i)})\right) + (1 - y^{(i)}) \log\left(1 - f_{\vec{w},b}(\vec{x}^{(i)})\right) \right] + \frac{\lambda}{2m_{train}} \sum_{j=1}^{n} w_j^2$$

Compute test error:

$$J_{test}(\vec{w}, b) = -\frac{1}{m_{test}} \sum_{i=1}^{m_{test}} \left[y_{test}^{(i)} \log\left(f_{\vec{w},b}\left(\vec{x}_{test}^{(i)}\right)\right) + \left(1 - y_{test}^{(i)}\right) \log\left(1 - f_{\vec{w},b}\left(\vec{x}_{test}^{(i)}\right)\right) \right]$$

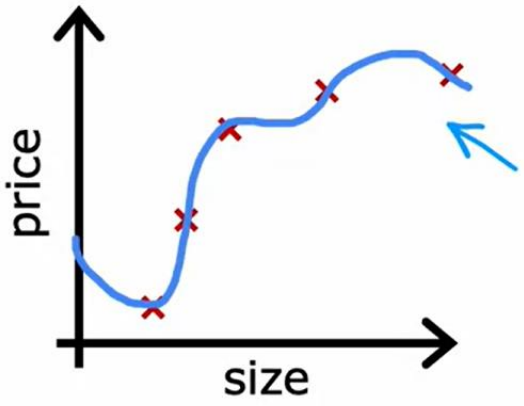
Compute train error:

$$J_{train}(\vec{w}, b) = -\frac{1}{m_{train}} \sum_{i=1}^{m_{train}} \left[y_{train}^{(i)} \log\left(f_{\vec{w},b}\left(\vec{x}_{train}^{(i)}\right)\right) + \left(1 - y_{train}^{(i)}\right) \log\left(1 - f_{\vec{w},b}\left(\vec{x}_{train}^{(i)}\right)\right) \right]$$

TRAIN AND TEST

Model selection(Training/Cross Validation/Test sets)

Model selection (choosing a model)



Once parameters $\vec{w}, b$ are fit to the training set, the training error $J_{train}(\vec{w}, b)$ is likely lower than the actual generalization error.

$J_{test}(\vec{w}, b)$ is better estimate of how well the model will generalize to new data compared to $J_{train}(\vec{w}, b)$.

$x = x_1$

$$f_{\vec{w},b}(\vec{x}) = w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + b$$

TRAIN AND TEST

Model selection(Training/Cross Validation/Test sets)

Model selection (choosing a model)

d=1 1. $f_{\vec{w},b}(\vec{x}) = w_1 x + b$ → $w^{<1>}, b^{<1>} \rightarrow J_{test}(w^{<1>}, b^{<1>})$

d=2 2. $f_{\vec{w},b}(\vec{x}) = w_1 x + w_2 x^2 + b$ → $w^{<2>}, b^{<2>} \rightarrow J_{test}(w^{<2>}, b^{<2>})$

d=3 3. $f_{\vec{w},b}(\vec{x}) = w_1 x + w_2 x^2 + w_3 x^3 + b$ → $w^{<3>}, b^{<3>} \rightarrow J_{test}(w^{<3>}, b^{<3>})$

⋮

d=10 10. $f_{\vec{w},b}(\vec{x}) = w_1 x + w_2 x^2 + \cdots + w_{10} x^{10} + b$ → $J_{test}(w^{<10>}, b^{<10>})$

Choose $w_1 x + \cdots + w_5 x^5 + b$ d=5 $J_{test}(w^{<5>}, b^{<5>})$

How well does the model perform? Report test set error $J_{test}(w^{<5>}, b^{<5>})$?

The problem: $J_{test}(w^{<5>}, b^{<5>})$ is likely to be an optimistic estimate of generalization error (ie. $J_{test}(w^{<5>}, b^{<5>}) <$ generalization error). Because an extra parameter d (degree of polynomial) was chosen using the test set.

w, b are overly optimistic estimate of generalization error on training data.

TRAIN AND TEST

Model selection(Training/Cross Validation/Test sets)

Training/cross validation/test set

size	price
2104	400
1600	330
2400	369
1416	232
3000	540
1985	300
1534	315
1427	199
1380	212
1494	243

training set 60% → $(x^{(1)}, y^{(1)}) \ \vdots \ (x^{(m_{train})}, y^{(m_{train})})$ $m_{train} = 6$

cross validation 20% → $(x_{cv}^{(1)}, y_{cv}^{(1)}) \ \vdots \ (x_{cv}^{(m_{cv})}, y_{cv}^{(m_{cv})})$ $m_{cv} = 2$

test set 20% → $(x_{test}^{(1)}, y_{test}^{(1)}) \ \vdots \ (x_{test}^{(m_{test})}, y_{test}^{(m_{test})})$ $m_{test} = 2$

TRAIN AND TEST

Model selection(Training/Cross Validation/Test sets)

Training/cross validation/test set

Training error: $$J_{train}(\vec{w}, b) = \frac{1}{2m_{train}} \left[\sum_{i=1}^{m_{train}} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right)^2 \right]$$

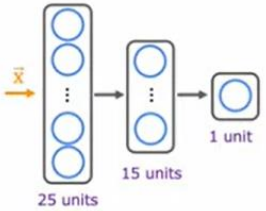
Cross validation error: $$J_{cv}(\vec{w}, b) = \frac{1}{2m_{cv}} \left[\sum_{i=1}^{m_{cv}} \left(f_{\vec{w},b}\left(\vec{x}_{cv}^{(i)}\right) - y_{cv}^{(i)} \right)^2 \right]$$ (validation error, dev error)

Test error: $$J_{test}(\vec{w}, b) = \frac{1}{2m_{test}} \left[\sum_{i=1}^{m_{test}} \left(f_{\vec{w},b}\left(\vec{x}_{test}^{(i)}\right) - y_{test}^{(i)} \right)^2 \right]$$

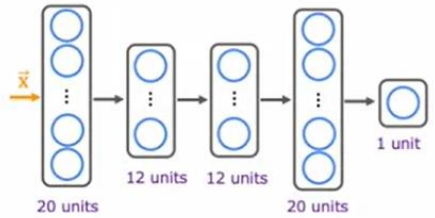
TRAIN AND TEST

Model selection(Training/Cross Validation/Test sets)

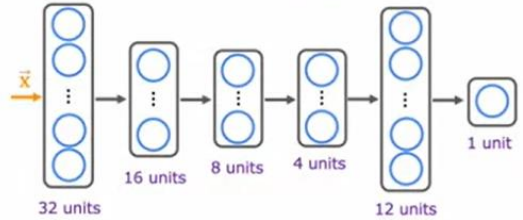
Model selection – choosing a neural network architecture

1. 

$w^{[1]}, b^{[1]}$ $J_{cv}(w^{[1]}, b^{[1]})$

→ 2. 

$w^{[2]}, b^{[2]}$ $J_{cv}(w^{[2]}, b^{[2]})$

3. 

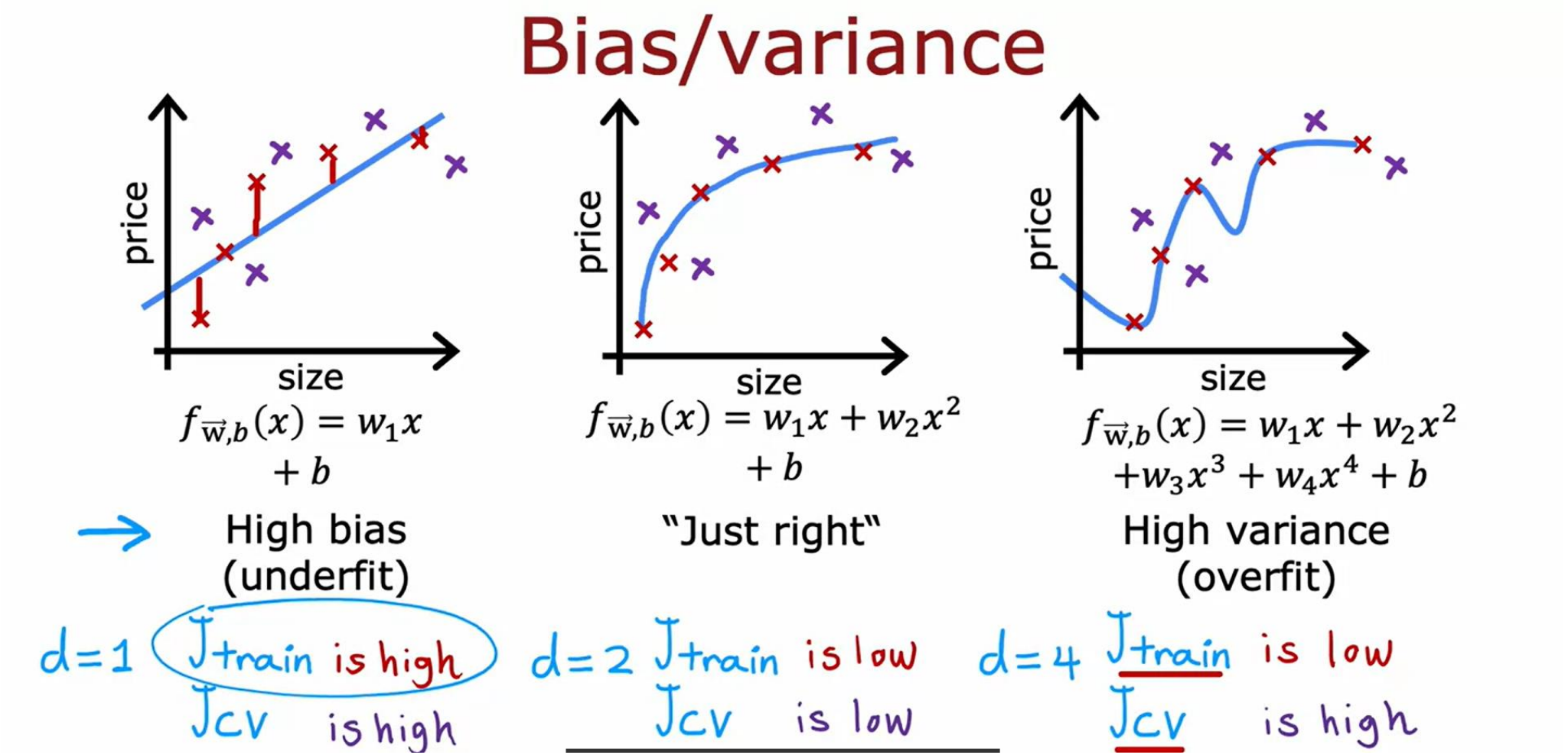
$w^{[3]}, b^{[3]}$ $J_{cv}(w^{[3]}, b^{[3]})$

Pick $\boldsymbol{w}^{<2>}, \boldsymbol{b}^{<2>}$

Estimate generalization error using the test set: $J_{test}(\boldsymbol{w}^{<2>}, \boldsymbol{b}^{(<2>)})$

TRAIN AND TEST

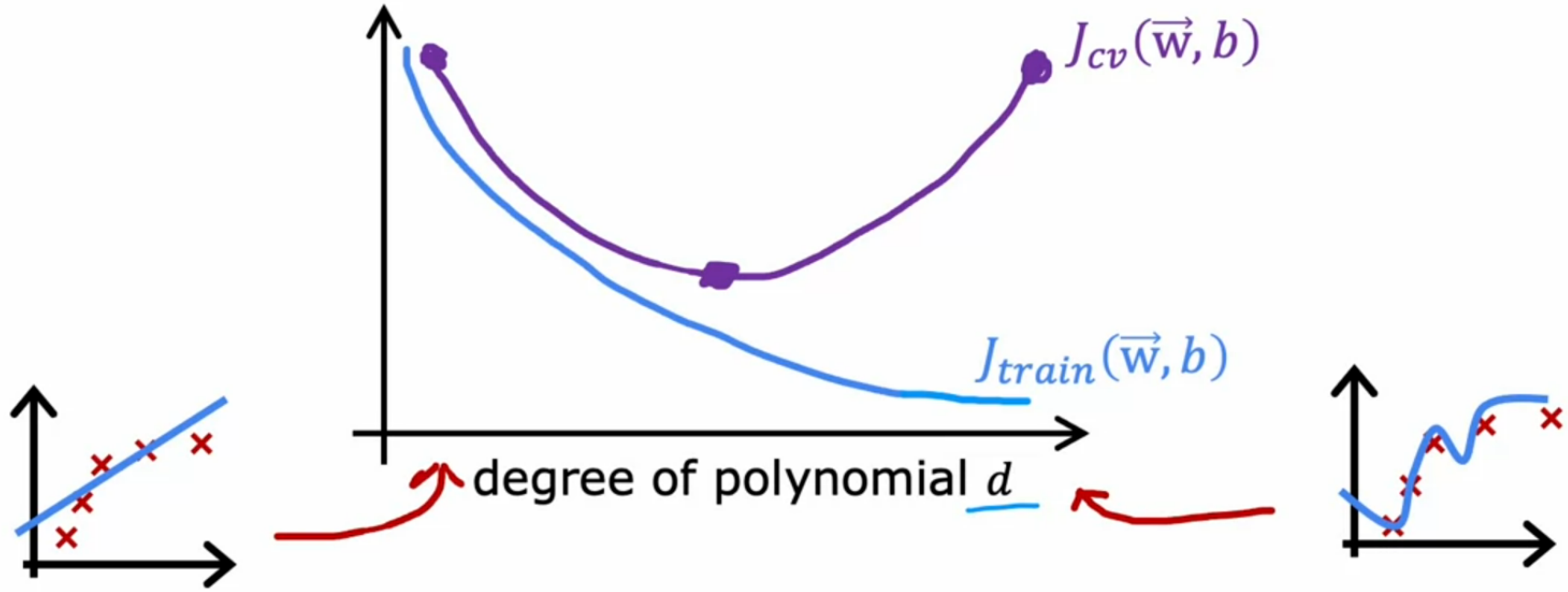
Diagnosing bias and variance



TRAIN AND TEST

Diagnosing bias and variance

Understanding bias and variance

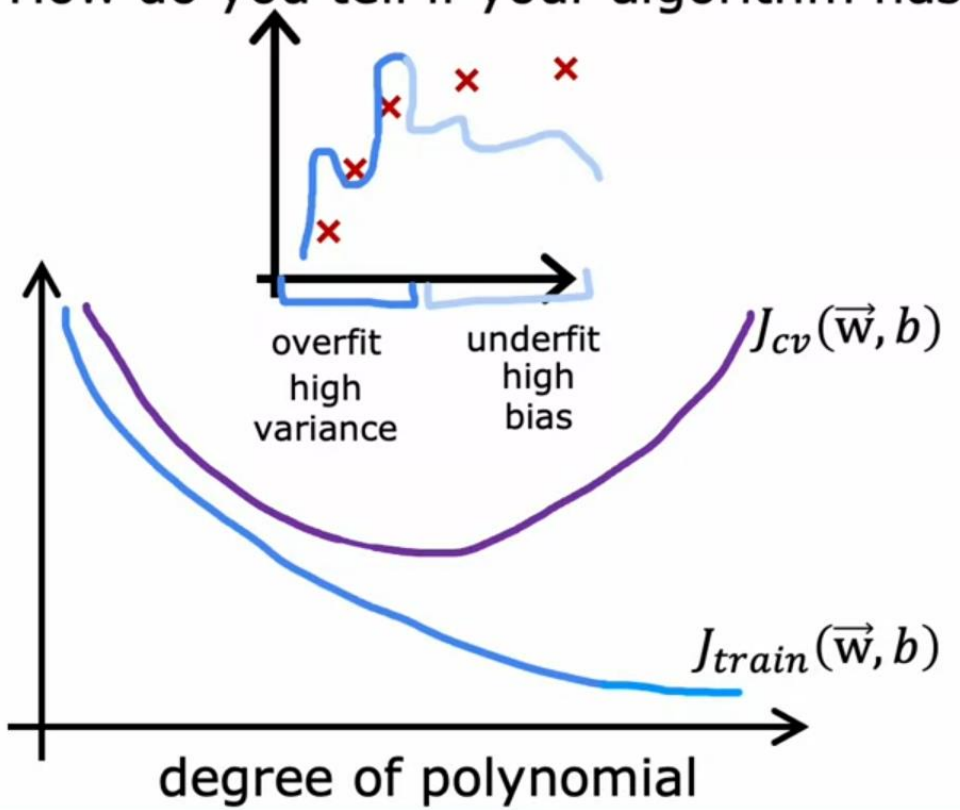


TRAIN AND TEST

Diagnosing bias and variance

Diagnosing bias and variance

How do you tell if your algorithm has a bias or variance problem?



High bias (underfit)

J_{train} will be high

($J_{train} \approx J_{cv}$)

High variance (overfit)

$J_{cv} \gg J_{train}$

(J_{train} may be low)

High bias and high variance

J_{train} will be high

and $J_{cv} \gg J_{train}$

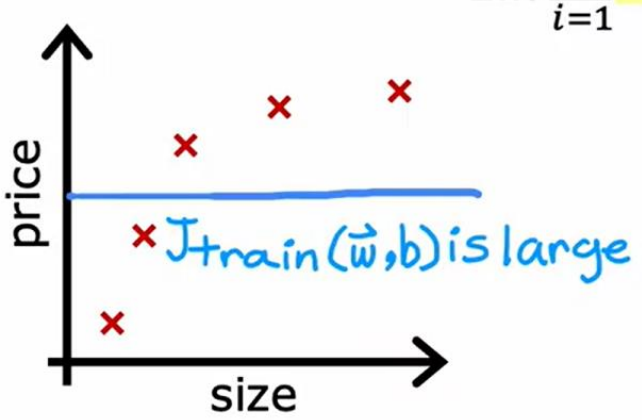
TRAIN AND TEST

Diagnosing bias and variance

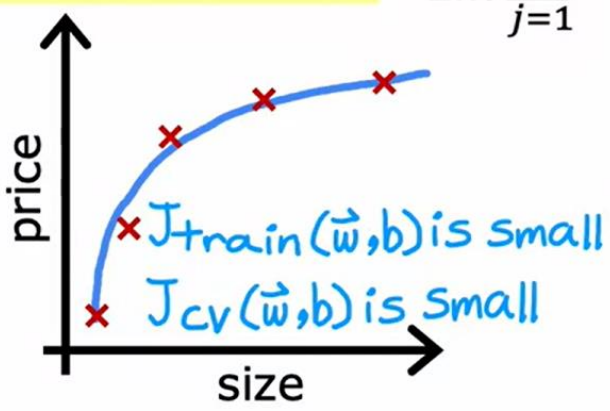
Linear regression with regularization

Model: $f_{\vec{w},b}(x) = w_1x + w_2x^2 + w_3x^3 + w_4x^4 + b$

$$J(\vec{w}, b) = \frac{1}{2m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)^2 + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2$$

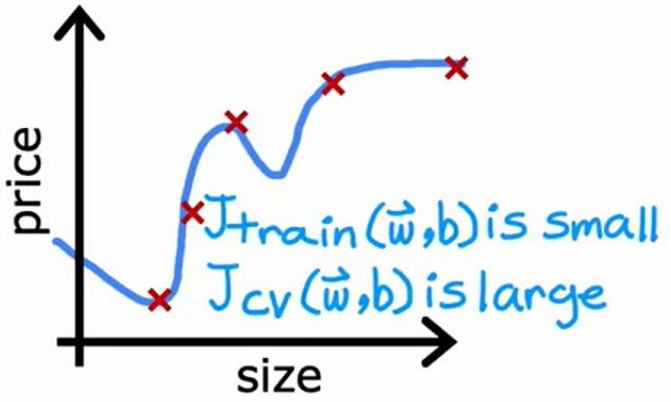


Large λ
High bias (underfit)
$\lambda = 10{,}000$ $w_1 \approx 0, w_2 \approx 0$
$f_{\vec{w},b}(\vec{x}) \approx b$



Intermediate λ

λ



Small λ
High variance (overfit)
$\lambda = 0$

TRAIN AND TEST

Diagnosing bias and variance

Choosing the regularization parameter λ

Model: $f_{\vec{w},b}(x) = w_1x + w_2x^2 + w_3x^3 + w_4x^4 + b$

1. Try $\lambda = 0$ → $\min_{\vec{w},b} J(\vec{w},b)$ → $w^{<1>},b^{<1>}$ → $J_{cv}(w^{<1>},b^{<1>})$
2. Try $\lambda = 0.01$ → $w^{<2>},b^{<2>}$ → $J_{cv}(w^{<2>},b^{<2>})$
3. Try $\lambda = 0.02$ → $J_{cv}(w^{<3>},b^{<3>})$
4. Try $\lambda = 0.04$
5. Try $\lambda = 0.08$

$J_{cv}(w^{<5>},b^{<5>})$

⋮

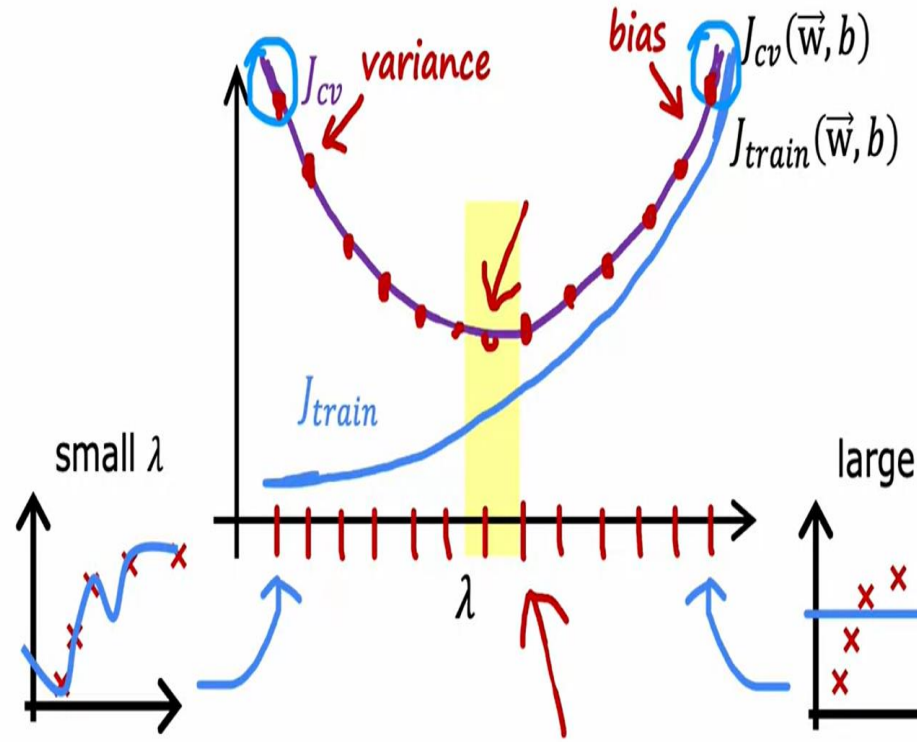
12. Try $\lambda \approx 10$ → $w^{<12>},b^{<12>}$ → $J_{cv}(w^{<12>},b^{<12>})$

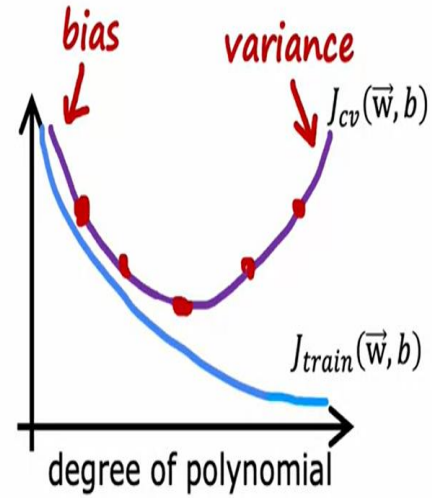
Pick $w^{<5>},b^{<5>}$

Report test error: $J_{test}(w^{<5>},b^{<5>})$

Bias and variance as a function of regularization parameter λ

$$J(\vec{w},b) = \frac{1}{2m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)^2 + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2$$





TRAIN AND TEST

Diagnosing bias and variance

Speech recognition example

Human level performance : 10.6%

Training error J_{train} : 10.8%

Cross validation error J_{cv} : 14.8%

0.2%

4.0%

Diagnosing bias and variance

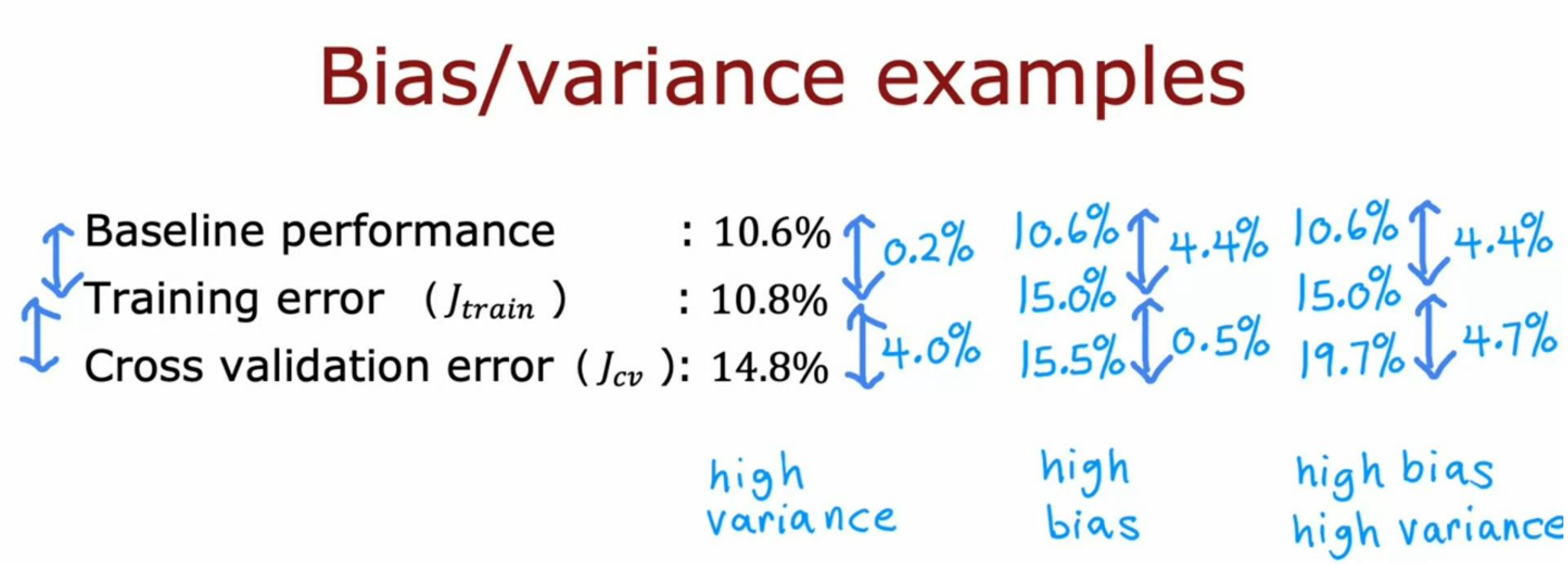
Establishing a baseline level of performance

What is the level of error you can reasonably hope to get to?

- Human level performance
- Competing algorithms performance
- Guess based on experience

TRAIN AND TEST

Diagnosing bias and variance



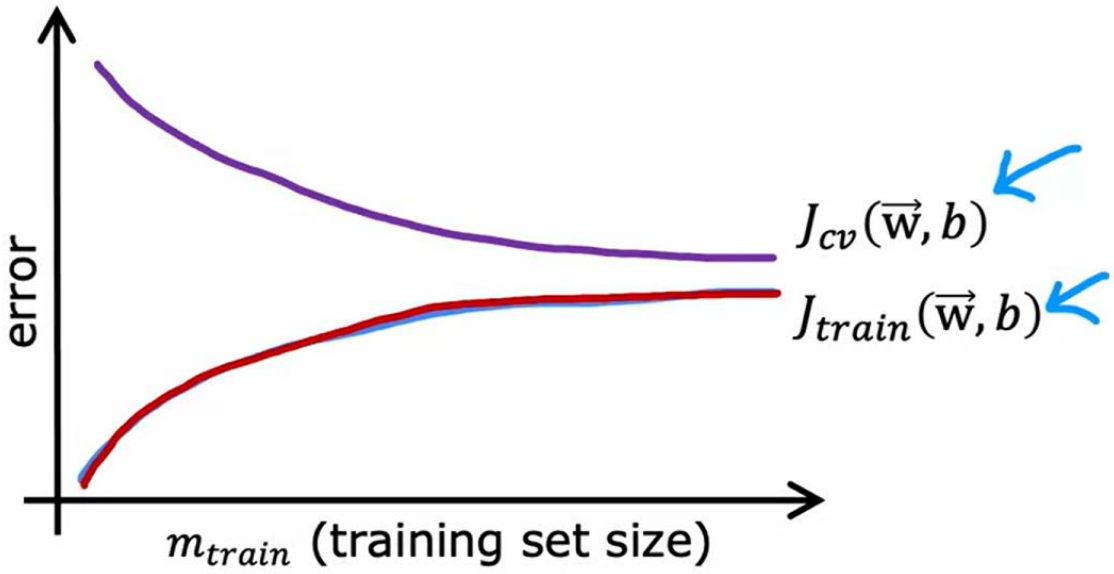
TRAIN AND TEST

Learning curves

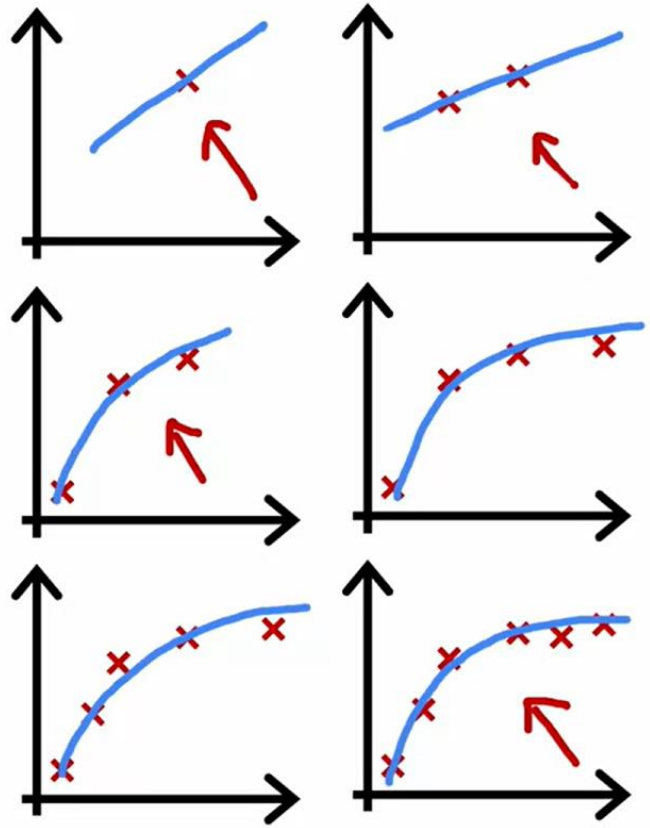
Learning curves

J_{train} = training error

J_{cv} = cross validation error



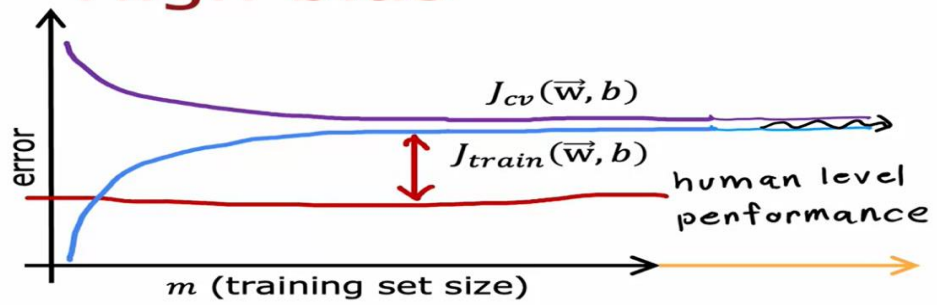
$f_{\vec{w},b}(x) = w_1 x + w_2 x^2 + b$



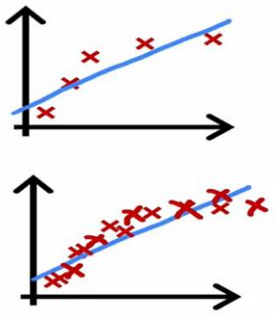
TRAIN AND TEST

Learning curves

High bias



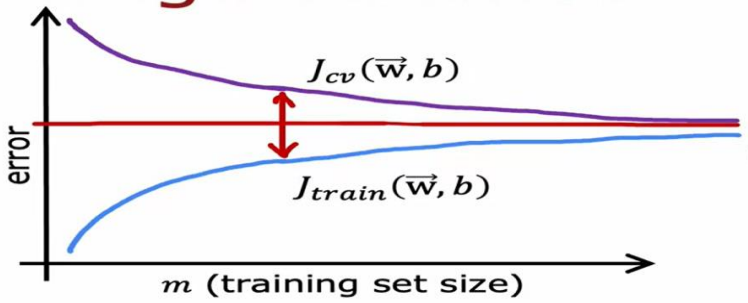
$f_{\vec{w},b}(x) = w_1 x + b$

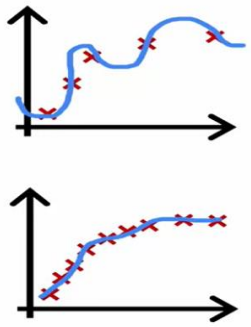


if a learning algorithm suffers from high bias, getting more training data will not (by itself) help much.

High variance

$f_{\vec{w},b}(x) = w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + b$
(with small λ)





If a learning algorithm suffers from high variance, getting more training data is likely to help.

TRAIN AND TEST

Learning curves

Debugging a learning algorithm

You've implemented regularized linear regression on housing prices

$$J(\vec{w}, b) = \frac{1}{2m}\sum_{i=1}^{m}\left(f_{\vec{w},b}\left(\vec{x}^{(i)}\right) - y^{(i)}\right)^2 + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2$$

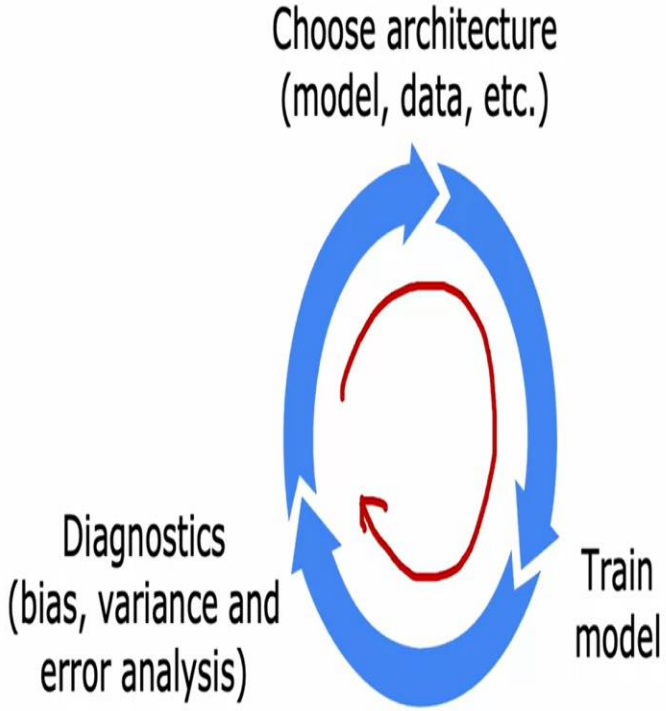
But it makes unacceptably large errors in predictions. What do you try next?

→ Get more training examples	fixes high variance
→ Try smaller sets of features $x, x^2, x^3, x^4, x^5 \ldots$	fixes high variance
→ Try getting additional features	fixes high bias
→ Try adding polynomial features $(x_1^2, x_2^2, x_1x_2, etc)$	fixes high bias
→ Try decreasing λ	fixes high bias
→ Try increasing λ	fixes high variance

TRAIN AND TEST

Iterative loop of ML development

Iterative loop of ML development



Spam classification example

From: cheapsales@buystufffromme.com
To: Andrew Ng
Subject: Buy now!

Deal of the week! Buy now!
Rolex w4tchs - $100
Med1cine (any kind) - £50
Also low cost M0rgages
available.

From: Alfred Ng
To: Andrew Ng
Subject: Christmas dates?

Hey Andrew,
Was talking to Mom about plans
for Xmas. When do you get off
work. Meet Dec 22?
Alf

TRAIN AND TEST

Iterative loop of LM development

Building a spam classifier

Supervised learning: $\vec{x}$ = features of email

y = spam (1) or not spam (0)

Features: list the top 10,000 words to compute $x_1, x_2, \cdots, x_{10,000}$

$$\vec{x} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ \vdots \end{bmatrix} \begin{matrix} a \\ andrew \\ buy \\ deal \\ discount \\ \vdots \end{matrix}$$

From: cheapsales@buystufffromme.com
To: Andrew Ng
Subject: Buy now!

Deal of the week! Buy now!
Rolex w4tchs - $100
Med1cine (any kind) - £50
Also low cost M0rgages
available.

Building a spam classifier

How to try to reduce your spam classifier's error?

- Collect more data. E.g., "Honeypot" project.
- Develop sophisticated features based on email routing (from email header).
- Define sophisticated features from email body. E.g., should "discounting" and "discount" be treated as the same word.
- Design algorithms to detect misspellings. E.g., w4tches, med1cine, m0rtgage.

TRAIN AND TEST

Error analysis

Error analysis

m_{cv} = 500 examples in cross validation set.

Algorithm misclassifies 100 of them.

Manually examine 100 examples and categorize them based on common traits.

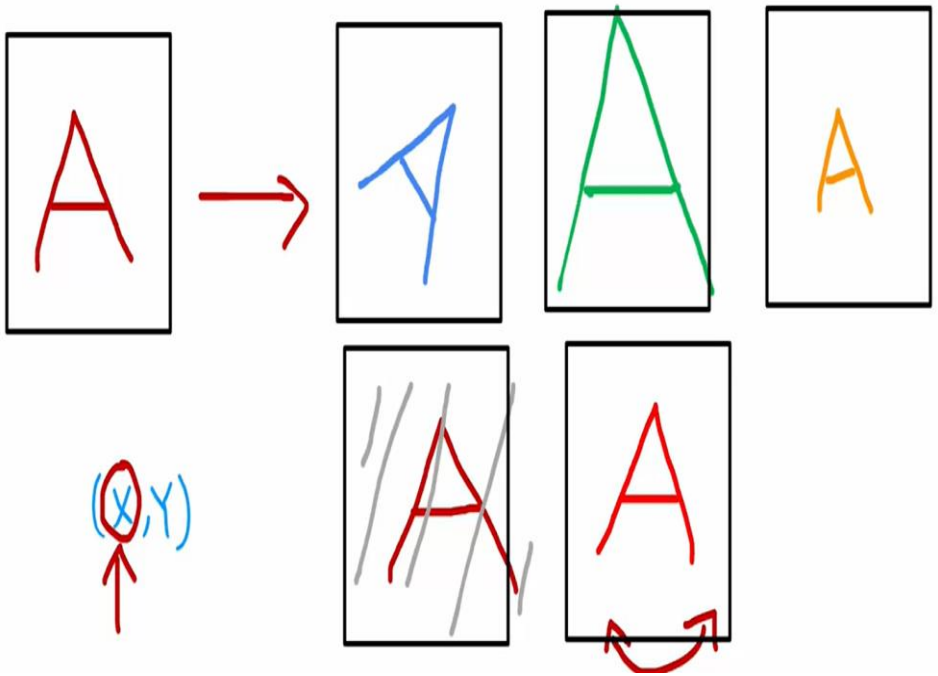
- Pharma: 21
- Deliberate misspellings (w4tches, med1cine): 3
- Unusual email routing: 7
- Steal passwords (phishing): 18
- Spam message in embedded image: 5

TRAIN AND TEST

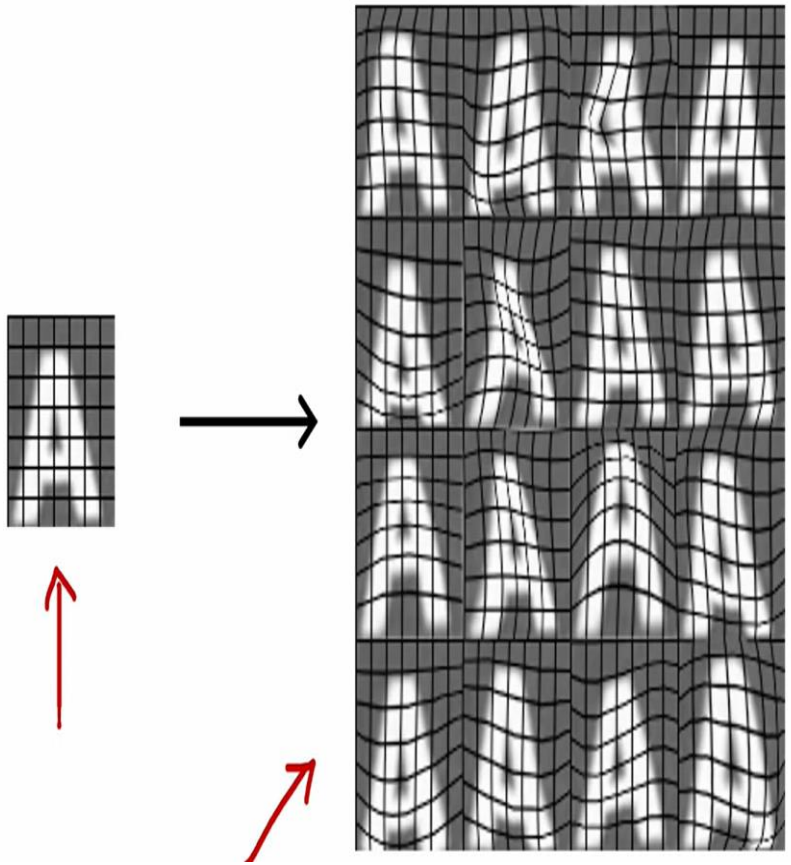
Adding Data

Data augmentation

Augmentation: modifying an existing training example to create a new training example.



Data augmentation by introducing distortions



TRAIN AND TEST

Adding Data

TRAIN AND TEST

Adding Data

Artificial data synthesis for photo OCR

Real data

TRAIN AND TEST

Adding Data

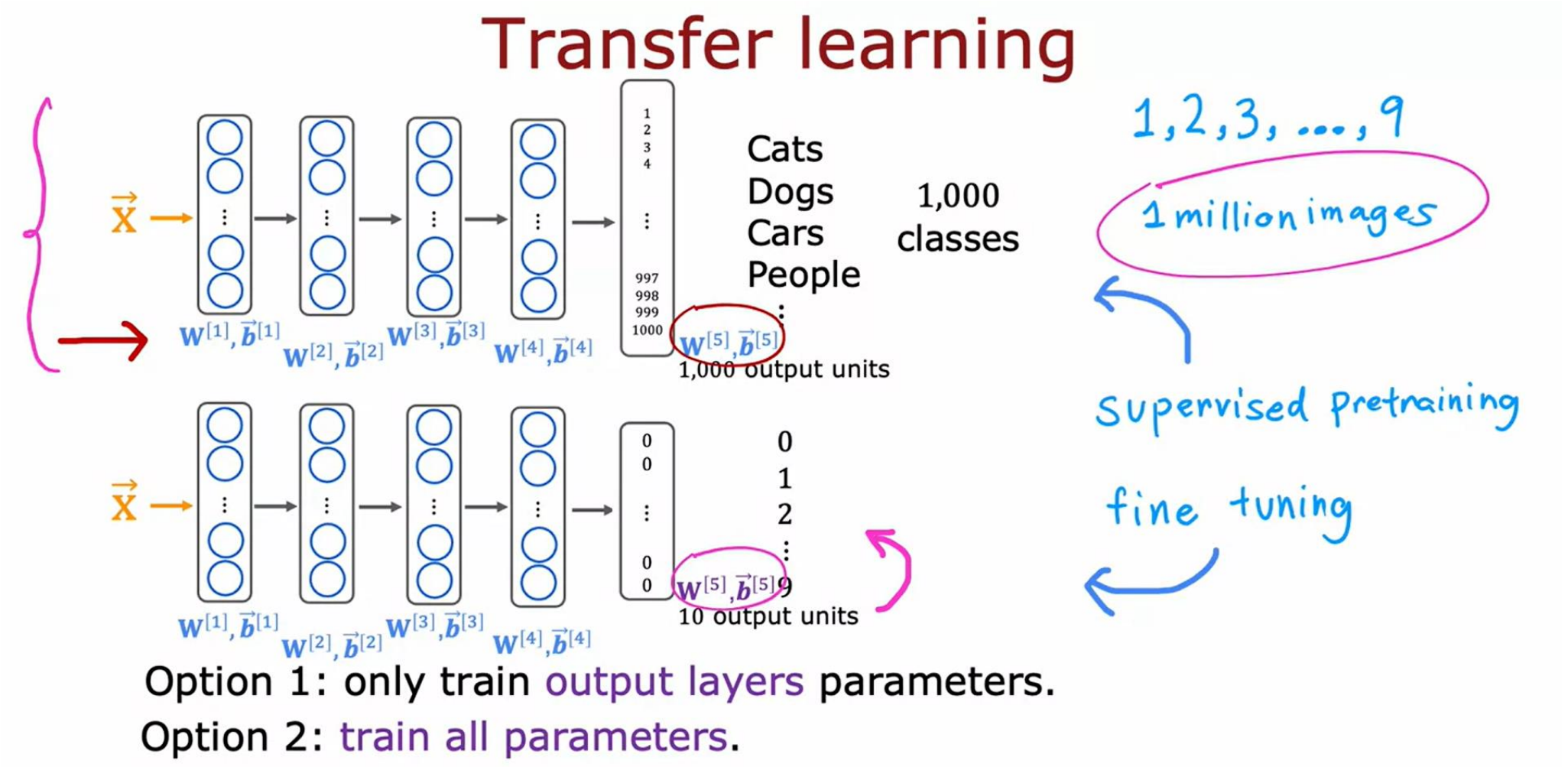
Artificial data synthesis for photo OCR

Real data

Synthetic data

TRAIN AND TEST

Transfer learning



Fairness, bias and ethics.

Bias

Hiring tool that discriminates against women.

Facial recognition system matching dark skinned individuals to criminal mugshots.

Biased bank loan approvals.

Toxic effect of reinforcing negative stereotypes.

Adverse use cases

Deepfakes

Spreading toxic/incendiary speech through optimizing for engagement.

Generating fake content for commercial or political purposes.

Using ML to build harmful products, commit fraud etc.

Spam vs anti-spam : fraud vs anti-fraud.

TRAIN AND TEST

TRAIN AND TEST