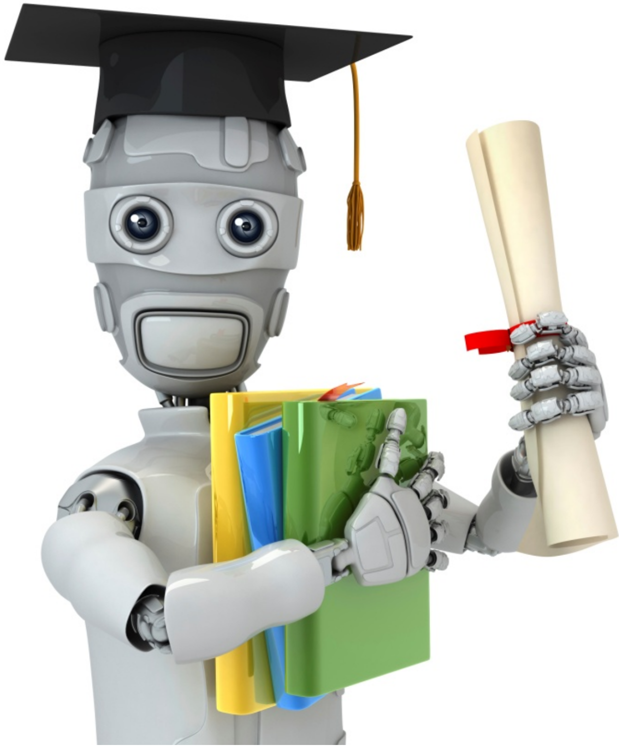


Machine Learning

Machine learning system design

Prioritizing what to work on: Spam classification example

Building a spam classifier

From: cheapsales@buystufffromme.com
To: ang@cs.stanford.edu
Subject: Buy now!

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

Spam (1)

From: Alfred Ng
To: ang@cs.stanford.edu
Subject: Christmas dates?

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

Non-spam (0)

Building a spam classifier

Supervised learning. $x =$ features of email. $y =$ spam (1) or not spam (0).

Features x: Choose 100 words indicative of spam/not spam.

E.g. deal, buy, discont, andrew, now, ...

$$x = \begin{array}{l} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 1 \\ \vdots \end{bmatrix} \end{array} \begin{array}{l} \text{andrew} \\ \text{buy} \\ \text{deal} \\ \text{discont} \\ \vdots \\ \text{now} \\ \vdots \end{array} \qquad x \in \mathbb{R}^{100}$$

$$x_j = \begin{cases} 1 & \text{if word } j \text{ appears in email} \\ 0 & \text{otherwise.} \end{cases}$$

```
From: cheapsales@buystufffromme.com
To: ang@cs.stanford.edu
Subject: Buy now!

Deal of the week! Buy now!
```

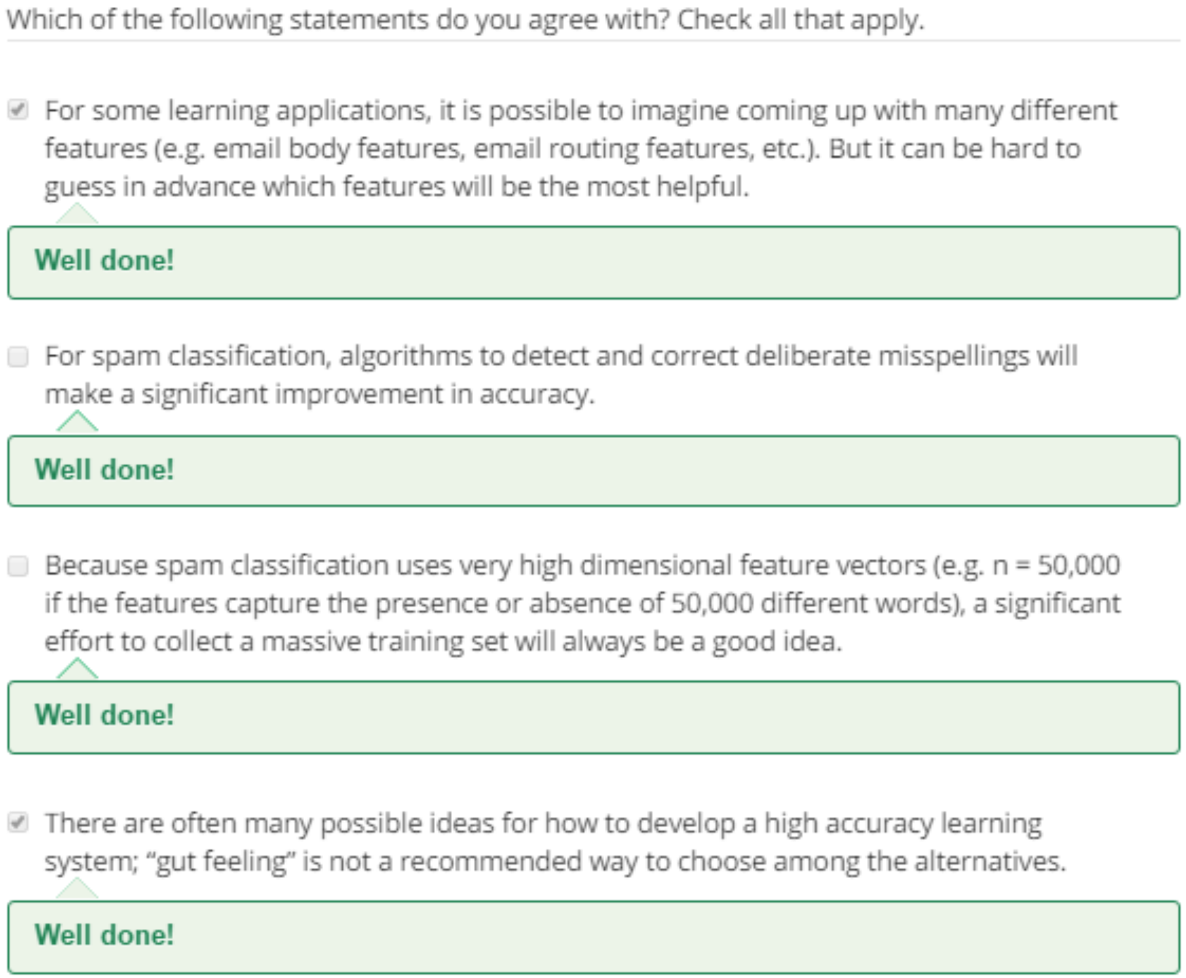
Note: In practice, take most frequently occurring n words (10,000 to 50,000) in training set, rather than manually pick 100 words.

Andrew Ng

Building a spam classifier

How to spend your time to make it have low error?

- Collect lots of data
  - E.g. “honeypot” project.
- Develop sophisticated features based on email routing information (from email header).
- Develop sophisticated features for message body, e.g. should “discount” and “discounts” be treated as the same word? How about “deal” and “Dealer”? Features about punctuation?
- Develop sophisticated algorithm to detect misspellings (e.g. m0rtgage, med1cine, w4tches.)



Which of the following statements do you agree with? Check all that apply.

☑ For some learning applications, it is possible to imagine coming up with many different features (e.g. email body features, email routing features, etc.). But it can be hard to guess in advance which features will be the most helpful.

Well done!

□ For spam classification, algorithms to detect and correct deliberate misspellings will make a significant improvement in accuracy.

Well done!

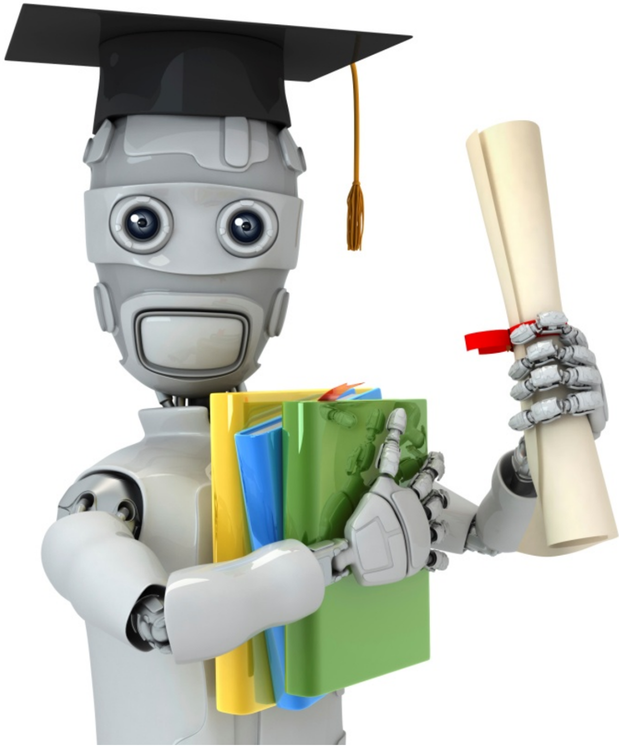
□ Because spam classification uses very high dimensional feature vectors (e.g. n = 50,000 if the features capture the presence or absence of 50,000 different words), a significant effort to collect a massive training set will always be a good idea.

Well done!

☑ There are often many possible ideas for how to develop a high accuracy learning system; "gut feeling" is not a recommended way to choose among the alternatives.

Well done!

Andrew Ng



Machine Learning

Machine learning system design

Error analysis

Recommended approach

- Start with a simple algorithm that you can implement quickly. Implement it and test it on your cross-validation data.
- Plot learning curves to decide if more data, more features, etc. are likely to help.
- Error analysis: Manually examine the examples (in cross validation set) that your algorithm made errors on. See if you spot any systematic trend in what type of examples it is making errors on.

Error Analysis

$m_{CV} =$ 500 examples in cross validation set
Algorithm misclassifies 100 emails.
Manually examine the 100 errors, and categorize them based on:

→ (i) What type of email it is pharma, replica, steal passwords, ...

→ (ii) What cues (features) you think would have helped the algorithm classify them correctly.

Pharma: 12
Replica/fake: 4
→ Steal passwords: 53
Other: 31

→ Deliberate misspellings: 5
(m0rgage, med1cine, etc.)
→ Unusual email routing: 16
→ Unusual (spamming) punctuation: 32
⋮

Andrew Ng

The importance of numerical evaluation

Should discount/discounts/discounted/discounting be treated as the same word?

Can use “stemming” software (E.g. “Porter stemmer”)

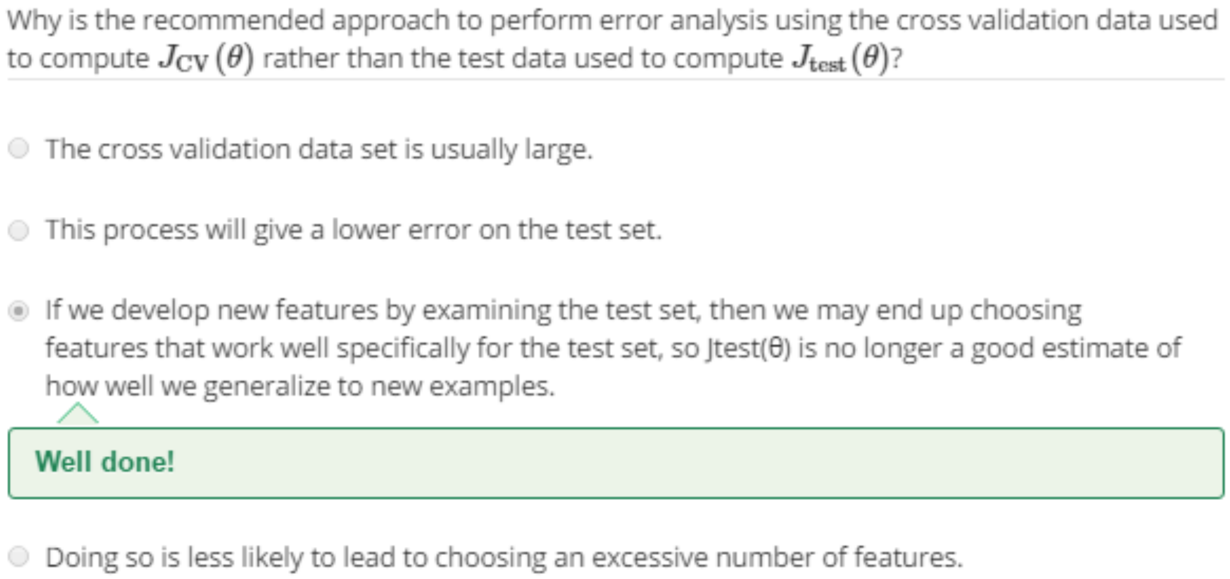
universe/university.

Error analysis may not be helpful for deciding if this is likely to improve performance. Only solution is to try it and see if it works.

Need numerical evaluation (e.g., cross validation error) of algorithm’s performance with and without stemming.

Without stemming: 5% error With stemming: 3% error

Distinguish upper vs. lower case (Mom/mom): 3.2%



Why is the recommended approach to perform error analysis using the cross validation data used to compute $J_{\text{CV}}(\theta)$ rather than the test data used to compute $J_{\text{test}}(\theta)$?

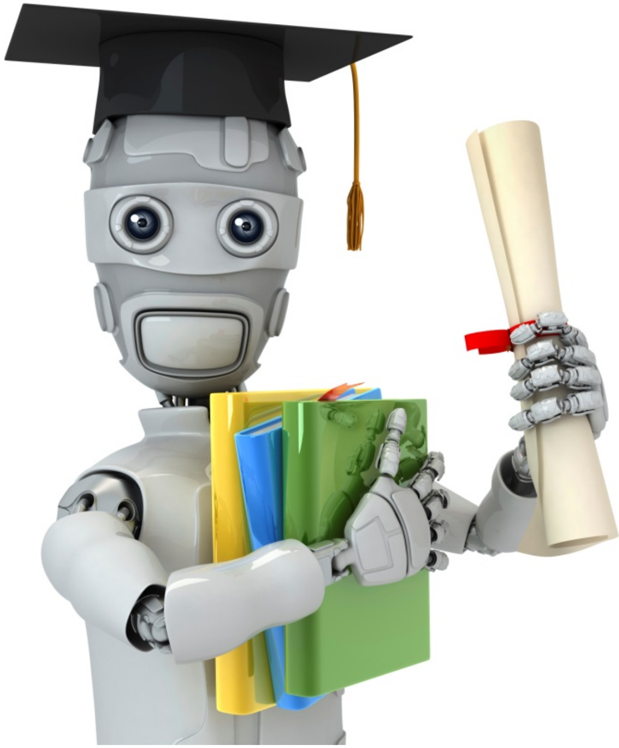
○ The cross validation data set is usually large.

○ This process will give a lower error on the test set.

● If we develop new features by examining the test set, then we may end up choosing features that work well specifically for the test set, so Jtest(θ) is no longer a good estimate of how well we generalize to new examples.

Well done!

○ Doing so is less likely to lead to choosing an excessive number of features.



Machine Learning

Machine learning system design

Error metrics for skewed classes

Cancer classification example

Train logistic regression model $h_\theta(x)$. ($\underline{y = 1}$ if cancer, $\underline{y = 0}$ otherwise)

Find that you got 1% error on test set.
(99% correct diagnoses)

Only 0.50% of patients have cancer.

→ skewed classes.

```
function y = predictCancer(x)
  → y = 0; %ignore x!
return
```

→ 0.5% error

→ 99.2% accury (0.8% error)

→ 99.5% accuracy (0.5% error)

Andrew Ng

Precision/Recall

$y = 1$ in presence of rare class that we want to detect

Actual class

		1	0
Predicted class	1	True positive	False positive
	0	False negative	True negative

$y = 0$

$\text{recall} = 0$

Precision

(Of all patients where we predicted $y = 1$, what fraction actually has cancer?)

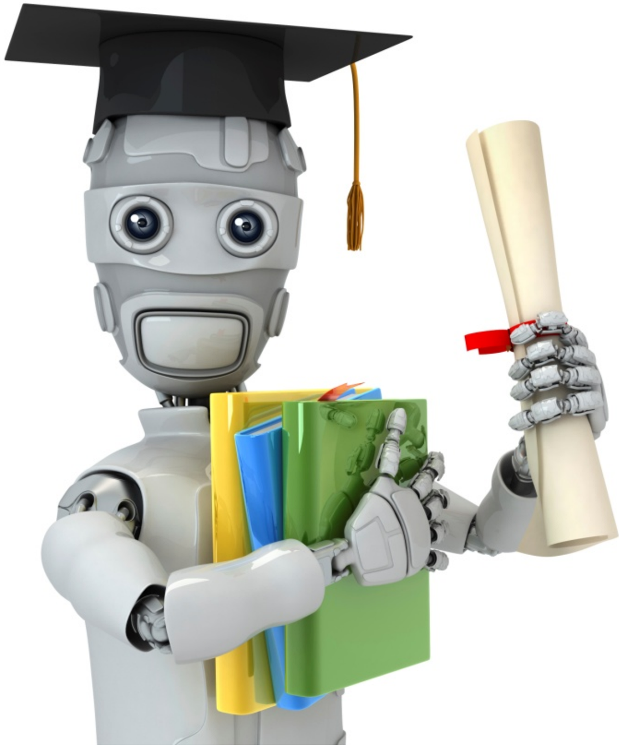
$$\frac{\text{True positives}}{\text{\# predicted positive}} = \frac{\text{True positive}}{\text{True pos} + \text{False pos}}$$

Recall

(Of all patients that actually have cancer, what fraction did we correctly detect as having cancer?)

$$\frac{\text{True positives}}{\text{\# actual positives}} = \frac{\text{True positives}}{\text{True pos} + \text{False neg}}$$

Andrew Ng



Machine Learning

Machine learning system design

Trading off precision and recall

Trading off precision and recall

→ precision = $\dfrac{\text{true positives}}{\text{no. of predicted positive}}$

→ recall = $\dfrac{\text{true positives}}{\text{no. of actual positive}}$

→ Logistic regression: $0 \leq h_\theta(x) \leq 1$

Predict 1 if $h_\theta(x) \geq$ ~~0.5~~ ~~0.7~~ ~~0.9~~ 0.3 ←

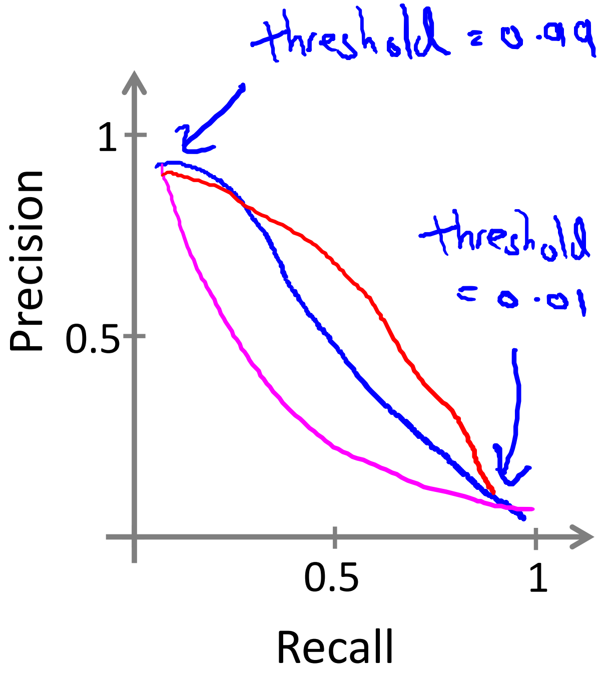
Predict 0 if $h_\theta(x) <$ ~~0.5~~ ~~0.7~~ ~~0.9~~ 0.3

→ Suppose we want to predict $y = 1$ (cancer) only if very confident.

→ Higher precision, lower recall.

→ Suppose we want to avoid missing too many cases of cancer (avoid false negatives).

→ Higher recall, lower precision.



More generally: Predict 1 if $\boxed{h_\theta(x) \geq \text{threshold.}}$ ←

Andrew Ng

F_1 Score (F score)

How to compare precision/recall numbers?

	Precision(P)	Recall (R)	
→ Algorithm 1	0.5	0.4	⇐
→ Algorithm 2	0.7	0.1	⇐
Algorithm 3	0.02	1.0	⇐

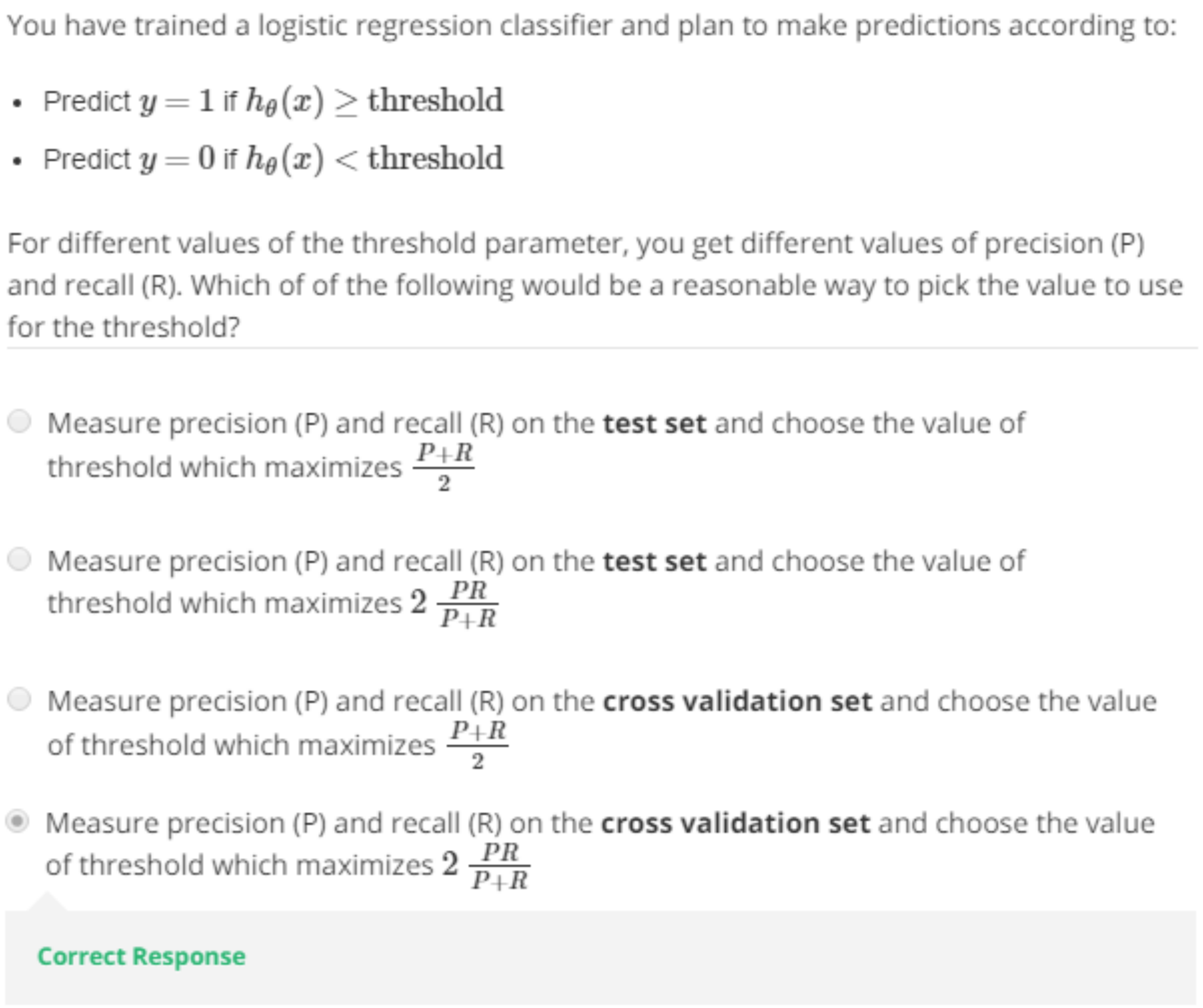
Predict y=1 all the time

Average: $\cancel{\frac{P+R}{2}}$

$$\boxed{F_1 \text{ Score: } 2\frac{PR}{P+R}}$$

P=0 or R=0 ⇒ F-score = 0.

P=1 and R=1 ⇒ F-score = 1



You have trained a logistic regression classifier and plan to make predictions according to:

- Predict $y = 1$ if $h_\theta(x) \geq \text{threshold}$
- Predict $y = 0$ if $h_\theta(x) < \text{threshold}$

For different values of the threshold parameter, you get different values of precision (P) and recall (R). Which of of the following would be a reasonable way to pick the value to use for the threshold?

○ Measure precision (P) and recall (R) on the **test set** and choose the value of threshold which maximizes $\frac{P+R}{2}$

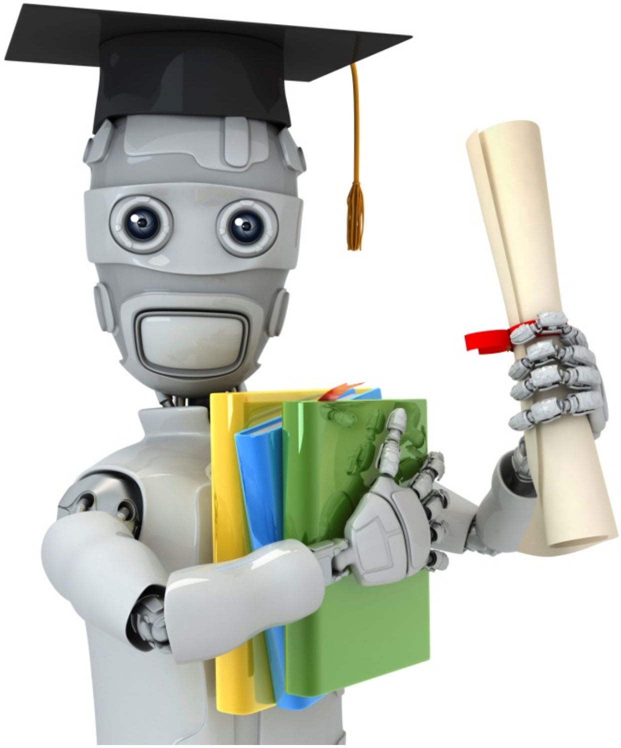
○ Measure precision (P) and recall (R) on the **test set** and choose the value of threshold which maximizes $2\frac{PR}{P+R}$

○ Measure precision (P) and recall (R) on the **cross validation set** and choose the value of threshold which maximizes $\frac{P+R}{2}$

● Measure precision (P) and recall (R) on the **cross validation set** and choose the value of threshold which maximizes $2\frac{PR}{P+R}$

Correct Response

Andrew Ng



Machine Learning

Machine learning system design

Data for machine learning

Designing a high accuracy learning system

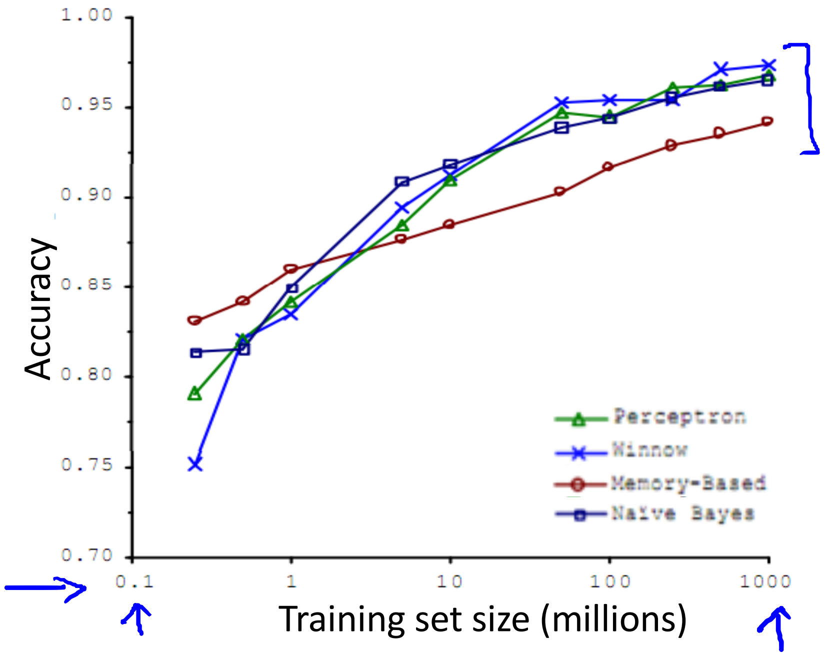
E.g. Classify between confusable words.

{to, two, too}, {then, than}

For breakfast I ate two ______ eggs.

Algorithms

- Perceptron (Logistic regression)
- Winnow
- Memory-based
- Naïve Bayes



"It's not who has the best algorithm that wins.

It's who has the most data."

[Banko and Brill, 2001]

Large data rationale

Assume feature $x \in \mathbb{R}^{n+1}$ has sufficient information to predict y accurately.

~~to~~, ~~too~~

Example: For breakfast I ate two eggs.

Counterexample: Predict housing price from only size (feet^2) and no other features.

Useful test: Given the input x, can a human expert confidently predict y?

Large data rationale

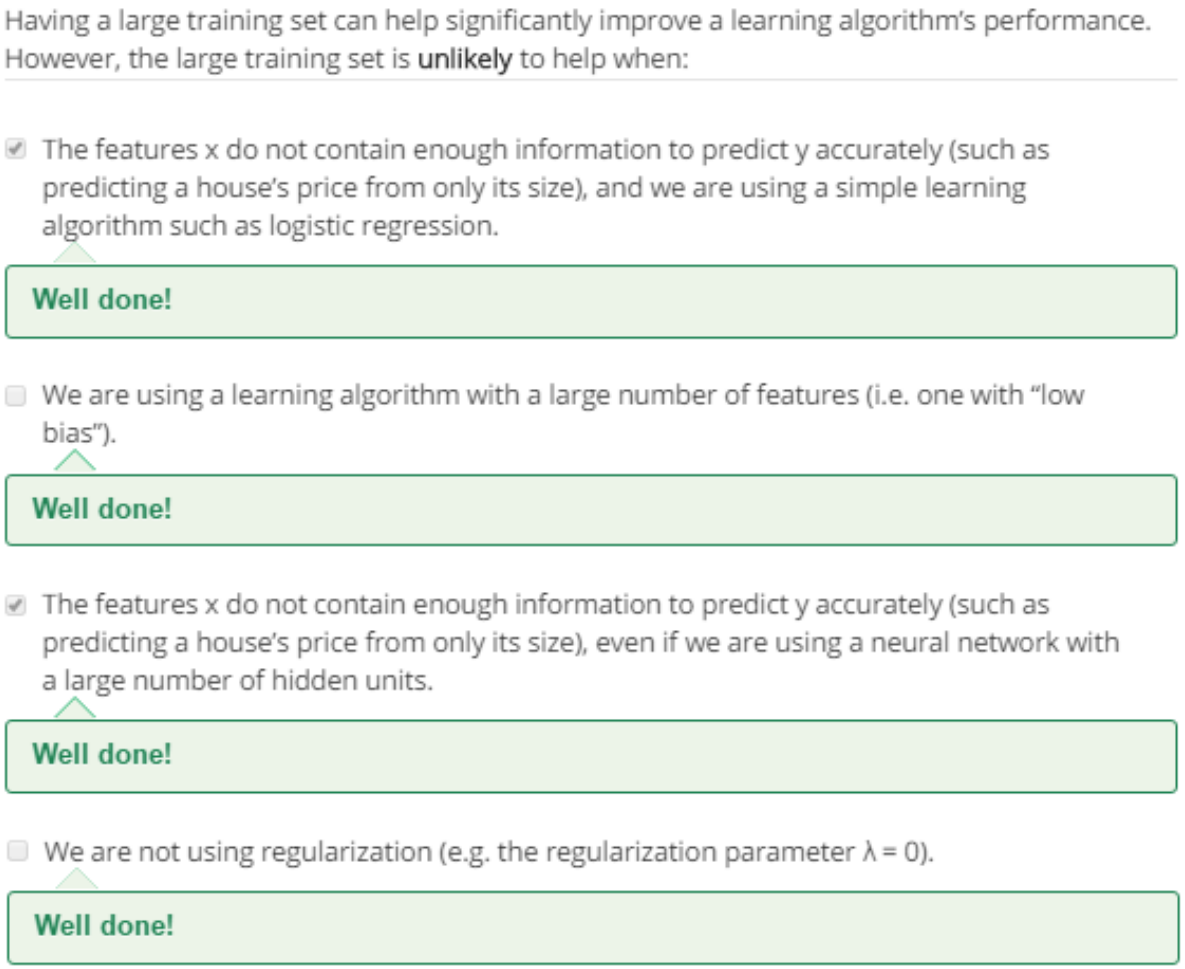
→ Use a learning algorithm with many parameters (e.g. logistic regression/linear regression with many features; neural network with many hidden units). low bias algorithms. ←

→ $J_{train}(\theta)$ will be small.

Use a very large training set (unlikely to overfit) low variance ↙

→ $J_{train}(\theta) \approx J_{test}(\theta)$

→ $J_{test}(\theta)$ will be small



Having a large training set can help significantly improve a learning algorithm's performance. However, the large training set is **unlikely** to help when:

☑ The features x do not contain enough information to predict y accurately (such as predicting a house's price from only its size), and we are using a simple learning algorithm such as logistic regression.

Well done!

□ We are using a learning algorithm with a large number of features (i.e. one with "low bias").

Well done!

☑ The features x do not contain enough information to predict y accurately (such as predicting a house's price from only its size), even if we are using a neural network with a large number of hidden units.

Well done!

□ We are not using regularization (e.g. the regularization parameter λ = 0).

Well done!