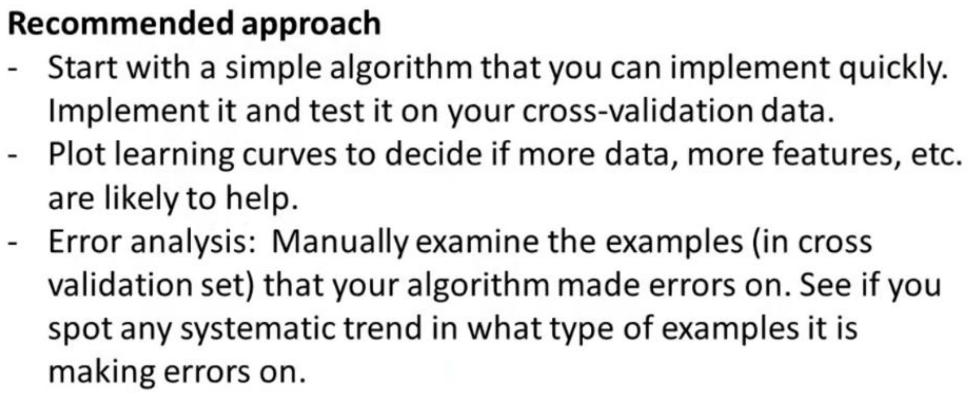
Plan

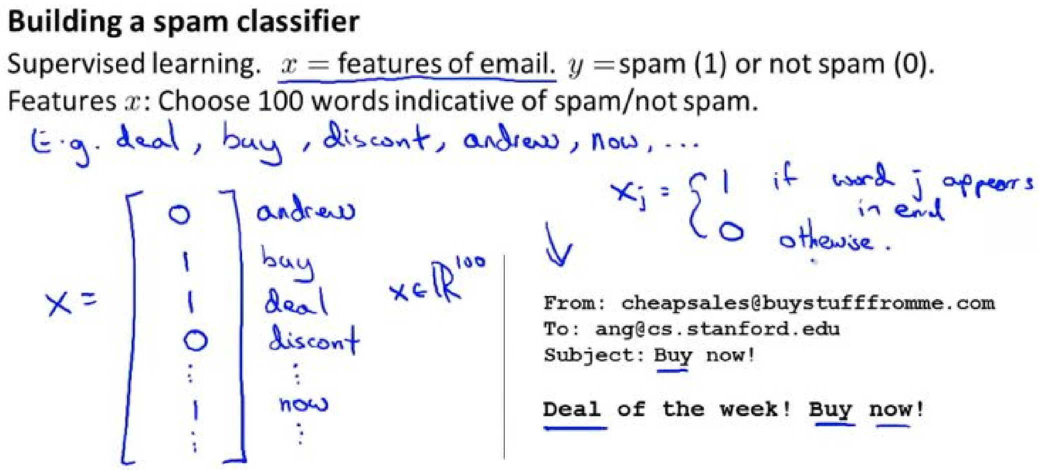
1. Machine learning system design - Practical recommendation on building ML-system
    a. Key points



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.

    b. Numerical algorithm evalution
    c. Error analysis
    d. Spam-classifier example



Building a spam classifier

Supervised learning. $\underline{x = \text{features of email.}}$ $y =$spam (1) or not spam (0).
Features x: Choose 100 words indicative of spam/not spam.

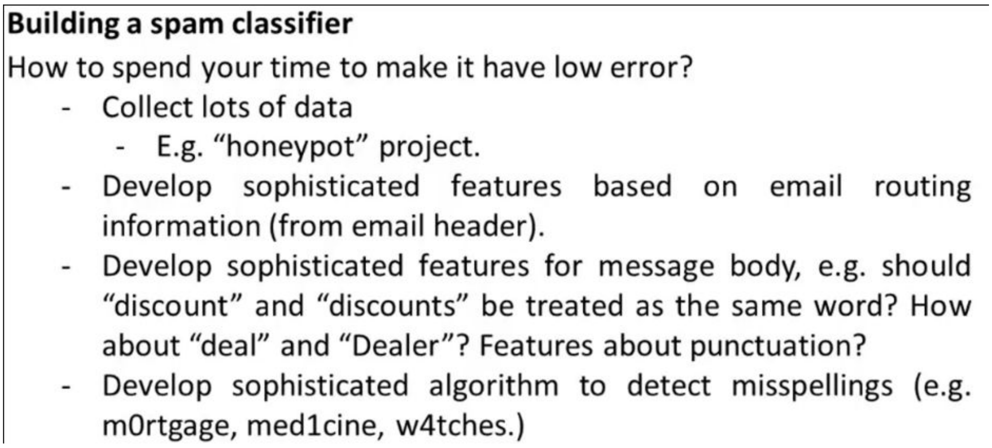
E.g. deal, buy, discont, andrew, now, ...

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

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

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

Deal of the week! Buy now!
```

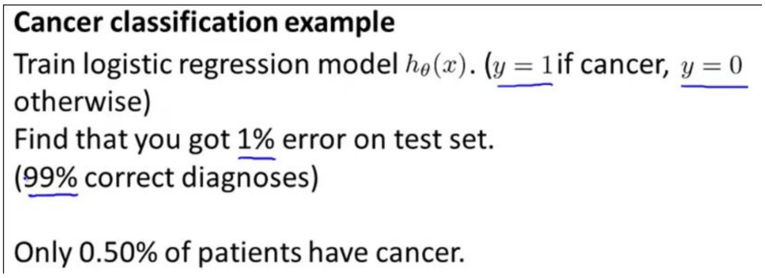


Building a spam classifier

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.)

2. Handling Skewed Data
    a. What is skewed data



Cancer classification example

Train logistic regression model $h_\theta(x)$. ($\underline{y = 1}$if cancer, $\underline{y = 0}$ otherwise)
Find that you got $\underline{1\%}$ error on test set.
($\underline{99\%}$ correct diagnoses)

Only 0.50% of patients have cancer.

return 0;

b. Better metrics

		Actual class	
		1	0
Predicted class	1	True Positive	False Positive
	0	False Negative	True Negative

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

$$\text{Recall} = \frac{\text{True positives}}{\text{\# actual positives}} = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}}$$

c. Trading off precision and recall

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

Predict 1 if $h_\theta(x) \geq 0.5$

Predict 0 if $h_\theta(x) < 0.5$

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.

	Precision(P)	Recall (R)	~~Average~~	F_1 Score
→ Algorithm 1	0.5	0.4	~~0.45~~	0.444 ←
→ Algorithm 2	0.7	0.1	~~0.4~~	0.175 ←
Algorithm 3	0.02	1.0 ← Predict y=1 all the time	~~$\boxed{0.51}$~~	0.0392 ←

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

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

Questions:

1. When and why do we want to use precision and recall instead of accuracy as the main metric of the model?
2. What is skewed data?
3. Suppose we have logistic classifier:
   y = 1 if h(x)>=d
   y = 0 if h(x)<d
   How can we choose the most optimal value for d?

Glossary:

Skewed data - data in which classes distributed extremely unbalanced - *That case is the case of what's called skewed classes.*

Recall - how many real positive cases model classified as positive - *One such evaluation metric is what's called recall.*