CS 260 – Foundations of Data Science

Lecture 6 – Multiple Linear Regression, Gradient Descent

09/10/2026

Adam Poliak

Announcements

HW3 due Tuesday night

Office Hours:

Tuesdays: 4pm-4:30pm

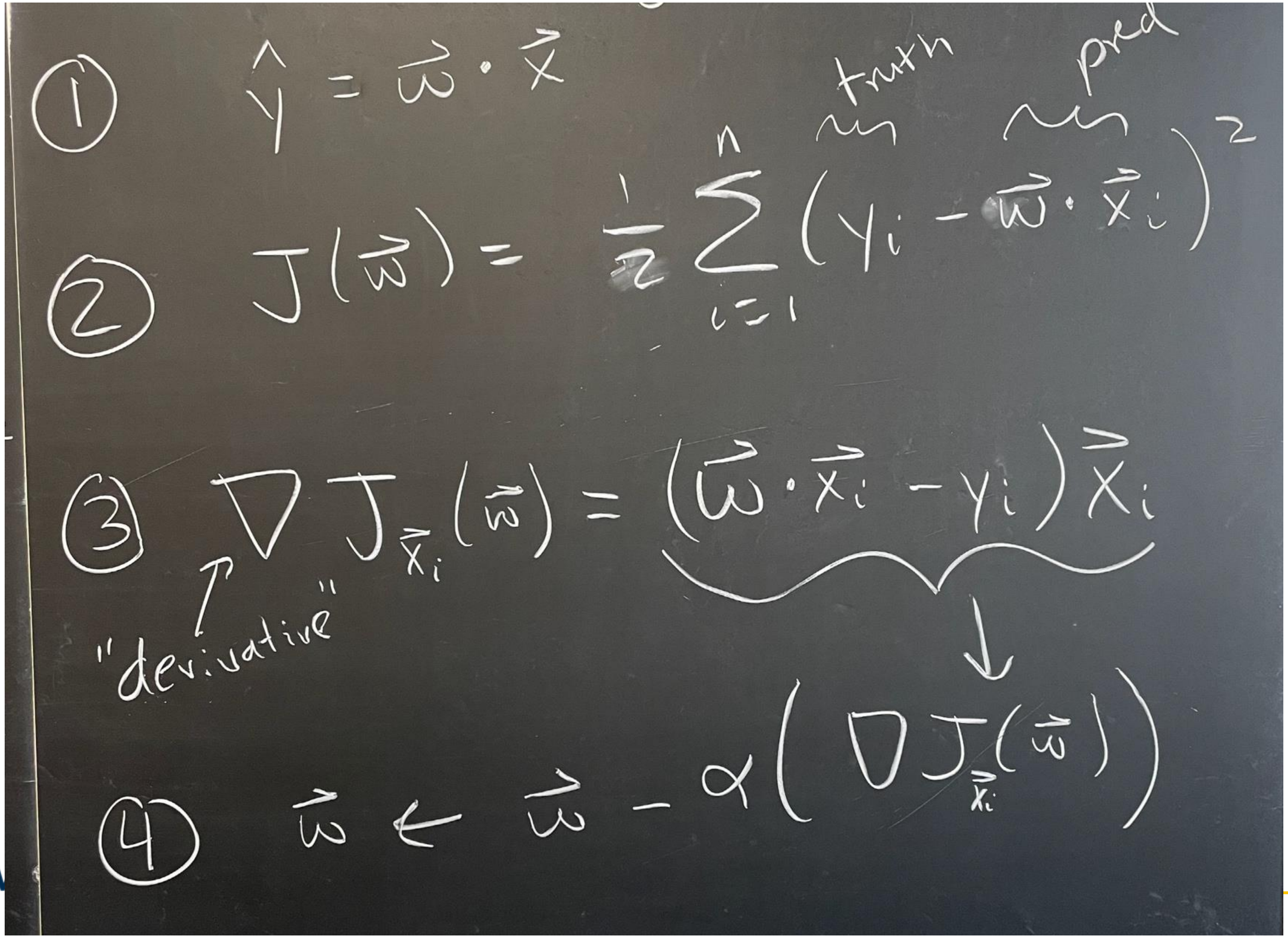
Thursdays 2-3:30pm, Park 200D

Mini-quiz for linear regression

For each of the following terms/descriptions, write out the corresponding equation:

1) Linear regression **model**
2) Linear regression **cost function**
3) **Gradient** of cost function wrt one datapoint
4) **Gradient descent** weight vector update

Mini-quiz for linear regression



Stochastic Gradient Descent for Linear Regression

Key Idea: take the derivative of **one datapoint** at a time and use that to update w

```
set w = 0 vector
while cost J(w) still changing (or max iter reached):
        shuffle data points
        for i = 1...n:
                w <- w – alpha(derivative of J(w) wrt xi)
        store J(w)
```

Outline

Introduction to classification

Evaluation Metrics

Binary classification examples

Transactions that indicate credit card fraud

Accounts that are bots

Detecting which scans show tumors

Prenatal test for Down's Syndrome

Finding genes under natural selection

Regions of the environment that contains the object the robot is searching for

In all these examples, we are trying to find unusual items ("needle in a haystack") -- we call these *positives*

Introduction to Classification

Decision tree

How to classify an example

Handout 6

Page 1

Outline

Introduction to classification

Evaluation Metrics

- Confusion matrices
- Precision and recall
- ROC curves

Goals of Evaluation

Think about what metrics are important for the problem at hand

Compare different methods or models on the same problem

Common set of tools that other researchers/users can understand

Training and Testing
(high-level idea)

Separate data into “**train**” and “**test**”

n = num training examples

m = num testing examples

Fit (create) the model using **training data**

e.g., sea_ice_1979-2012.csv

Evaluate the model using **testing data**

e.g., sea_ice_2013-2020.csv

Classify a tweet as viral or not

Accuracy

Model A performs 60% accuracy, would you say this is good, decent, or awful?

Model B performs 80% accuracy, would you say this is good, decent, or awful

Model C performs 98% accuracy, would you say this is good decent or awful?

Evaluation: Accuracy

- Imagine we saw 1 million tweets
  - 100 of them were viral
  - 999,900 were not
- We could build a dumb classifier that just labels every tweet "not viral"
  - It would get 99.99% accuracy!!! Wow!!!!
  - But useless! Cant find the viral tweets!
- When should we not we use **accuracy** as our metric?
  - When data isn't balanced across labels/classes

The 2-by-2 confusion matrix

true positive	**false positive**
false negative	**true negative**

The 2-by-2 confusion matrix

y

gold standard labels

$\hat{y}$ *system output labels*		gold positive	gold negative
	system positive	**true positive**	**false positive**
	system negative	**false negative**	**true negative**

The 2-by-2 confusion matrix

gold standard labels

system output labels		gold positive	gold negative	
	system positive	**true positive**	**false positive**	**precision** $= \dfrac{\text{tp}}{\text{tp+fp}}$
	system negative	**false negative**	**true negative**	
		recall $= \dfrac{\text{tp}}{\text{tp+fn}}$		**accuracy** $= \dfrac{\text{tp+tn}}{\text{tp+fp+tn+fn}}$

Evaluation: Precision

% of items the system detected (i.e., items the system labeled as positive) that are in fact positive (according to the human gold labels)

$$\textbf{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$

Evaluation: Recall

% of items actually present in the input that were correctly identified by the system.

$$\textbf{Recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

Why Precision and recall

Our dumb viral-classifier

label no tweets as ”viral"

Accuracy=99.99%

but

Recall = 0

(it doesn't get any of the 100 viral tweets)

Precision and recall, unlike accuracy, emphasize true positives:

finding the things that we are supposed to be looking for.

A combined measure: F

F measure: a single number that combines P and R:

$$F_\beta = \frac{(\beta^2+1)PR}{\beta^2 P + R}$$

We almost always use balanced F_1 (i.e., $\beta = 1$)

$$\mathrm{F}_1 = \frac{2PR}{P+R}$$

Confusion Matrix for 3-class classification

gold labels

system output	urgent	normal	spam	
urgent	8	10	1	**precision**$_u$=
normal	5	60	50	**precision**$_n$=
spam	3	30	200	**precision**$_s$=
	recall$_u$ =	**recall**$_n$ =	**recall**$_s$ =	

Confusion Matrix for 3-class classification

		gold labels			
		urgent	normal	spam	
system output	urgent	8	10	1	**precision**$_u = \frac{8}{8+10+1}$
	normal	5	60	50	**precision**$_n = \frac{60}{5+60+50}$
	spam	3	30	200	**precision**$_s = \frac{200}{3+30+200}$
		recall$_u = \frac{8}{8+5+3}$	**recall**$_n = \frac{60}{10+60+30}$	**recall**$_s = \frac{200}{1+50+200}$	

How to combine Precision/Recall from 3 classes to get one metric

Macroaveraging:

compute the performance for each class, and then average over classes

Microaveraging:

collect decisions for all classes into one confusion matrix

compute precision and recall from that table.

Macroaveraging and Microaveraging

Class 1: Urgent

	true urgent	true not
system urgent	8	11
system not	8	340

precision =

Class 2: Normal

	true normal	true not
system normal	60	55
system not	40	212

precision =

Class 3: Spam

	true spam	true not
system spam	200	33
system not	51	83

precision =

microaverage precision =

macroaverage precision =

Macroaveraging and Microaveraging

Class 1: Urgent

	true urgent	true not
system urgent	8	11
system not	8	340

$$\text{precision} = \frac{8}{8+11} = .42$$

Class 2: Normal

	true normal	true not
system normal	60	55
system not	40	212

$$\text{precision} = \frac{60}{60+55} = .52$$

Class 3: Spam

	true spam	true not
system spam	200	33
system not	51	83

$$\text{precision} = \frac{200}{200+33} = .86$$

microaverage precision =

macroaverage precision =

Macroaveraging and Microaveraging

Class 1: Urgent

	true urgent	true not
system urgent	8	11
system not	8	340

$$\text{precision} = \frac{8}{8+11} = .42$$

Class 2: Normal

	true normal	true not
system normal	60	55
system not	40	212

$$\text{precision} = \frac{60}{60+55} = .52$$

Class 3: Spam

	true spam	true not
system spam	200	33
system not	51	83

$$\text{precision} = \frac{200}{200+33} = .86$$

Pooled

	true yes	true no
system yes		
system no		

microaverage precision =

macroaverage precision =

Macroaveraging and Microaveraging

Class 1: Urgent

	true urgent	true not
system urgent	8	11
system not	8	340

$$\text{precision} = \frac{8}{8+11} = .42$$

Class 2: Normal

	true normal	true not
system normal	60	55
system not	40	212

$$\text{precision} = \frac{60}{60+55} = .52$$

Class 3: Spam

	true spam	true not
system spam	200	33
system not	51	83

$$\text{precision} = \frac{200}{200+33} = .86$$

Pooled

	true yes	true no
system yes	268	99
system no	99	635

microaverage precision = $\boxed{}$

macroaverage precision = $\boxed{}$

Macroaveraging and Microaveraging

Class 1: Urgent

	true urgent	true not
system urgent	8	11
system not	8	340

$$\text{precision} = \frac{8}{8+11} = .42$$

Class 2: Normal

	true normal	true not
system normal	60	55
system not	40	212

$$\text{precision} = \frac{60}{60+55} = .52$$

Class 3: Spam

	true spam	true not
system spam	200	33
system not	51	83

$$\text{precision} = \frac{200}{200+33} = .86$$

Pooled

	true yes	true no
system yes	268	99
system no	99	635

$$\text{microaverage precision} = \frac{268}{268+99} = \mathbf{.73}$$

$$\text{macroaverage precision} = \frac{.42+.52+.86}{3} = \mathbf{.60}$$

Precision and Recall

Precision: of all the “flagged” examples, which ones are actually relevant (i.e., positive)?

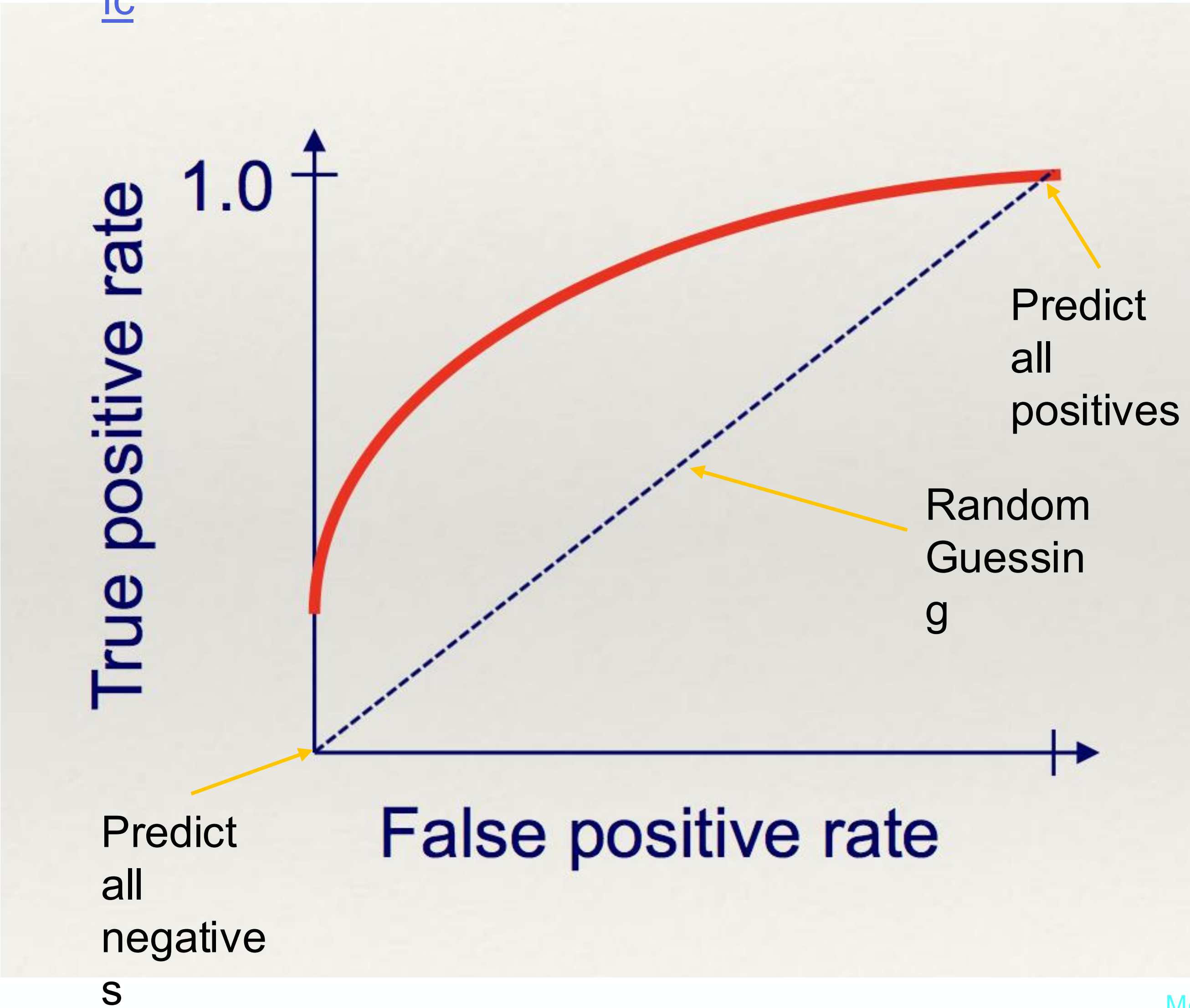
(Purity)

Recall: of all the relevant results, which ones did I actually return?

(Completeness)

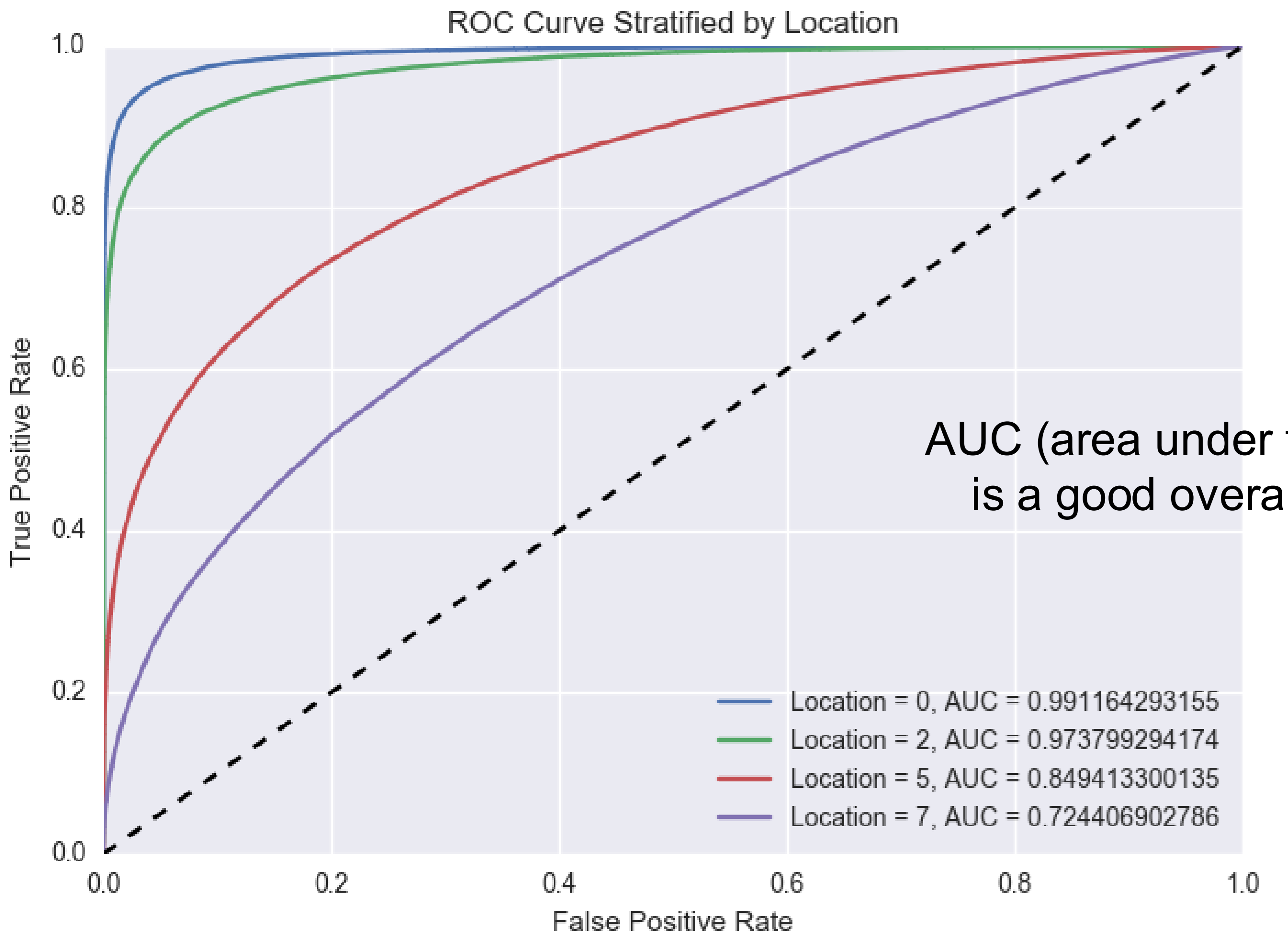
ROC curve (Receiver Operating Characteristic)

More history here! https://en.wikipedia.org/wiki/Receiver_operating_characteristic



Modified from Ameet Soni

ROC curve example: comparing methods



AUC (area under the curve)
is a good overall metric

Example of a ROC curve
Chan, Perrone, Spence, Jenkins, Mathieson, Song

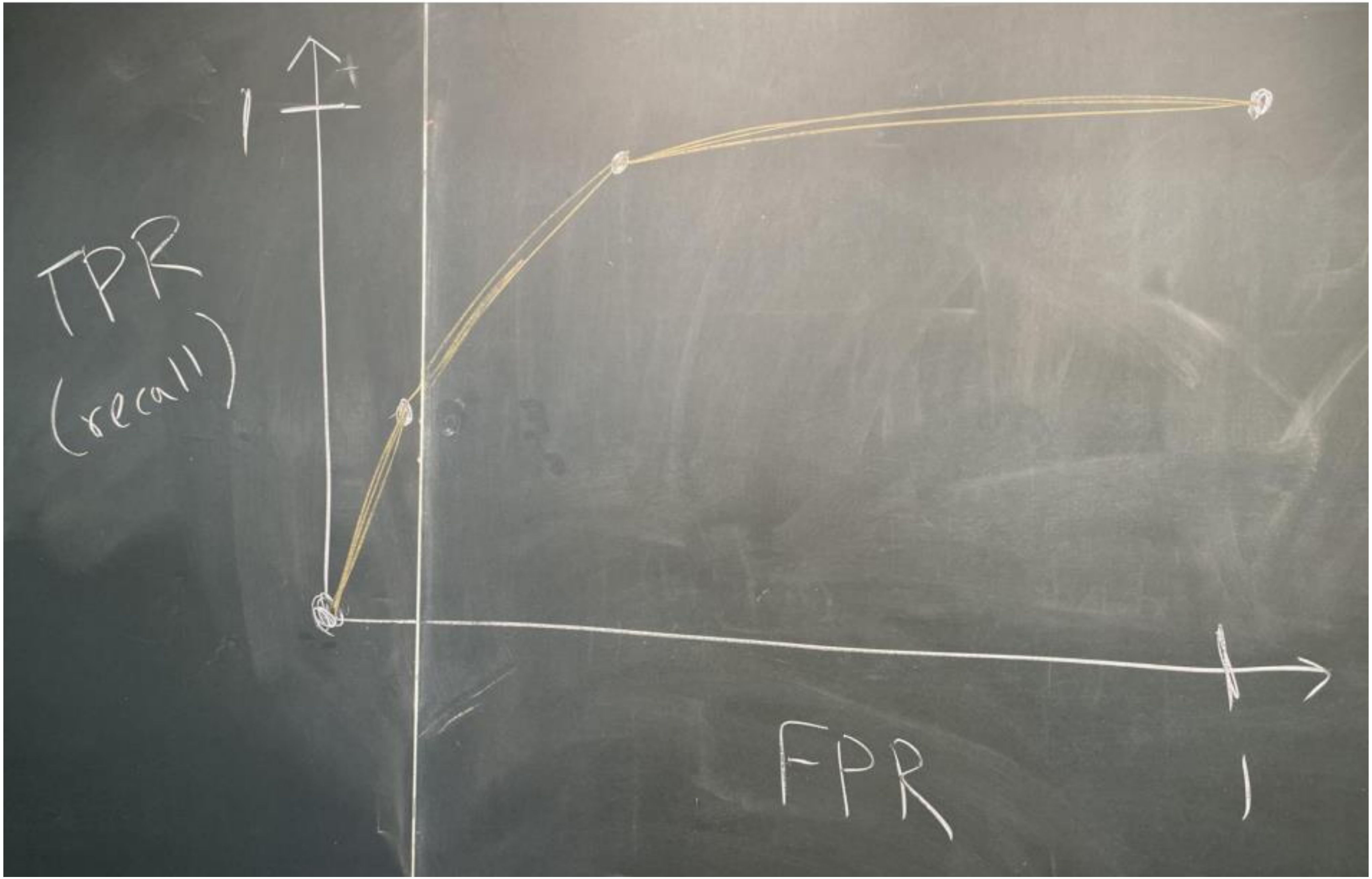
How to get an ROC curve for probabilistic methods?

Typically use 0.5 as a threshold for binary classification

Vary the threshold! (i.e. choose 0, 0.1, 0.2, 0.3, …)

$p(y = 1 \mid x) \geq 0.2$ => classify as positive
$p(y = 1 \mid x) < 0.2$ => classify as positive

Handout 6


1
TPR
(recall)
FPR
1