Real World ML

Lyle Ungar

Evaluation metrics
The final project
Real-world ML projects
Future classes: visualization and causality

Evaluating ML

Lyle Ungar

Probability vs loss
Confusion matrix: TP/TN/FP/FN
Precision, Recall, Sensitivity, Specificity
ROC curves

What is Netflix trying to do?

?

Top

Start the presentation to see live content. For screen share software, share the entire screen. Get help at **pollev.com/app**

Loss functions come from decision making

- **We often optimize a loss function which is a surrogate for our true loss function**
- **Don't confuse *probability* or *score* with loss**
  - One can optimize a model for probability and then use the probability in a decision rule
  - Or just directly optimize the loss resulting from a decision rule

Regression loss function

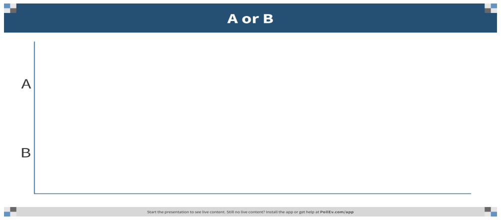
- **For a linear regression predicting dollar amounts (e.g. income, housing prices)**
  - What is the loss function being optimized for
  - What is the residual plot likely to look like?

- **Does this meet the assumptions of the linear regression model?**
  - If not, how could you fix it?

Precision, Recall, Sensitivity, Specificity and ROC curves

Have you seen ROC curves?

A) Yes
B) No



Ways to be right or wrong

Claim\Is	True Yes	True No
Classify Yes	True Positive	False Positive
Classify No	False Negative	True Negative

Accuracy = (TP + TN)/(TP+FP+FN+TN)

Measuring Performance

- **Accuracy (symmetric)**
  - % correctly classified
- **Asymmetric measures**
  - **Precision**
    - P(yes | predicted as yes)
  - **Recall (or Sensitivity)**
    - P(predicted as yes | yes)
  - **Specificity**
    - P(predicted as no)| no)

Precision/Recall Sensitivity/Specificity

Claim\Is	True Yes	True No
Classify Yes	True Positive	False Positive
Classify No	False Negative	True Negative

- **Precision**
  - P(yes | predicted as yes) = TP/(TP+FP)
- **Recall (or Sensitivity)**
  - P(predicted as yes | yes) = TP/(TP+FN)
- **Specificity**
  - P (predicted as no)| no) = TN/(TN+FP)

Precision/Recall Example

Claim\Is	True Good	True Not Good	
Classify "Good"	70	50	
Classify "Not good"	30	350	
			500

- **Precision**
  - P(good | predicted as good) = 70/(70+50)
- **Recall (or Sensitivity) = True Positive Rate (TPR)**
  - P(predicted as good | good) = 70/(70+30)
- **Specificity = 1 – (False Positive Rate)**
  - P (predicted as bad| bad) = 350/(350+50)

F1 combines Precision and Recall

Claim\Is	True Yes	True No	
Classify Yes	True Positive	False Positive	
Classify No	False Negative	True Negative	

- **Precision**
  - TP/(TP+FP)
- **Recall**
  - TP/(TP+FN)

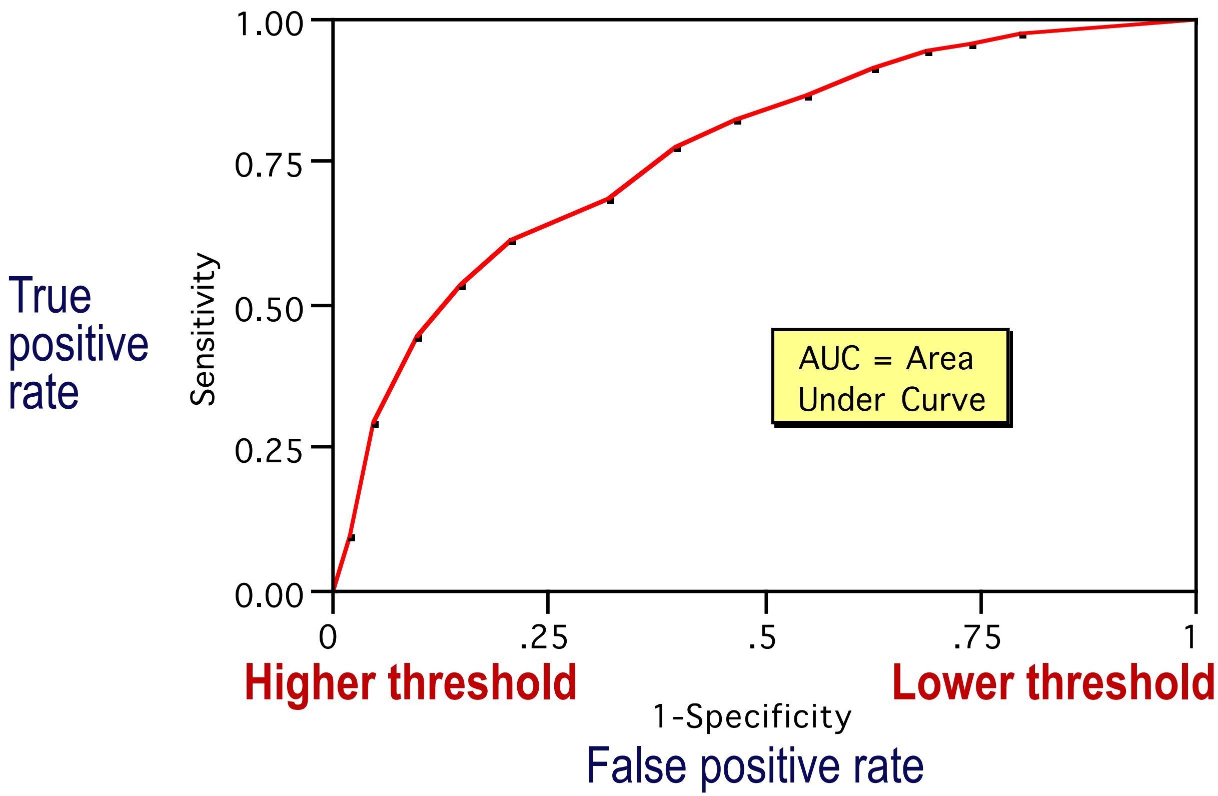
- **F1**
  - 2 precision * recall/(precision + recall)

ROC (Receiver Operating Characteristic) Curve

- Sort all examples from highest probability (or score) of being 'yes', p(y='yes'|**x**), to lowest
- Sweep the threshold for predicting an example to be labeled 'yes' from 1 down to 0
  - This varies *specificity* from 1 to 0.
- At each threshold compute the *sensitivity*
  - i.e., the fraction of the true positives you found
- Plot the curve

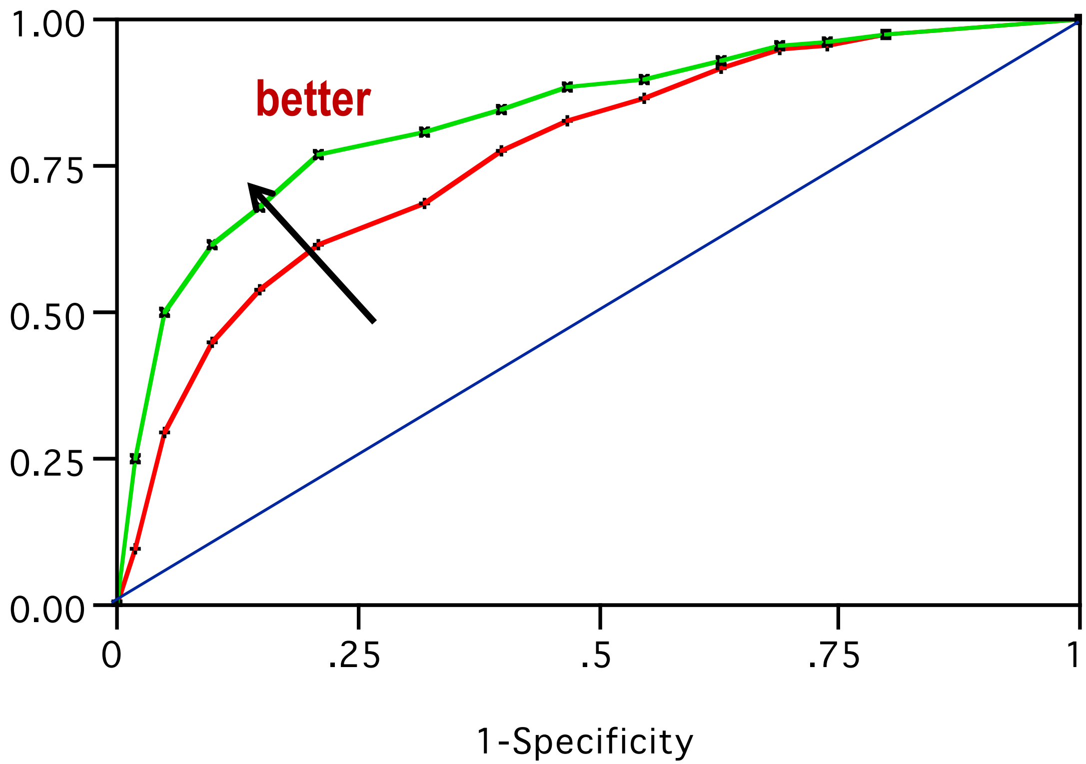
https://en.wikipedia.org/wiki/Receiver_operating_characteristic

ROC Chart Varies Threshold



p(y|x) > threshold **to be in class**

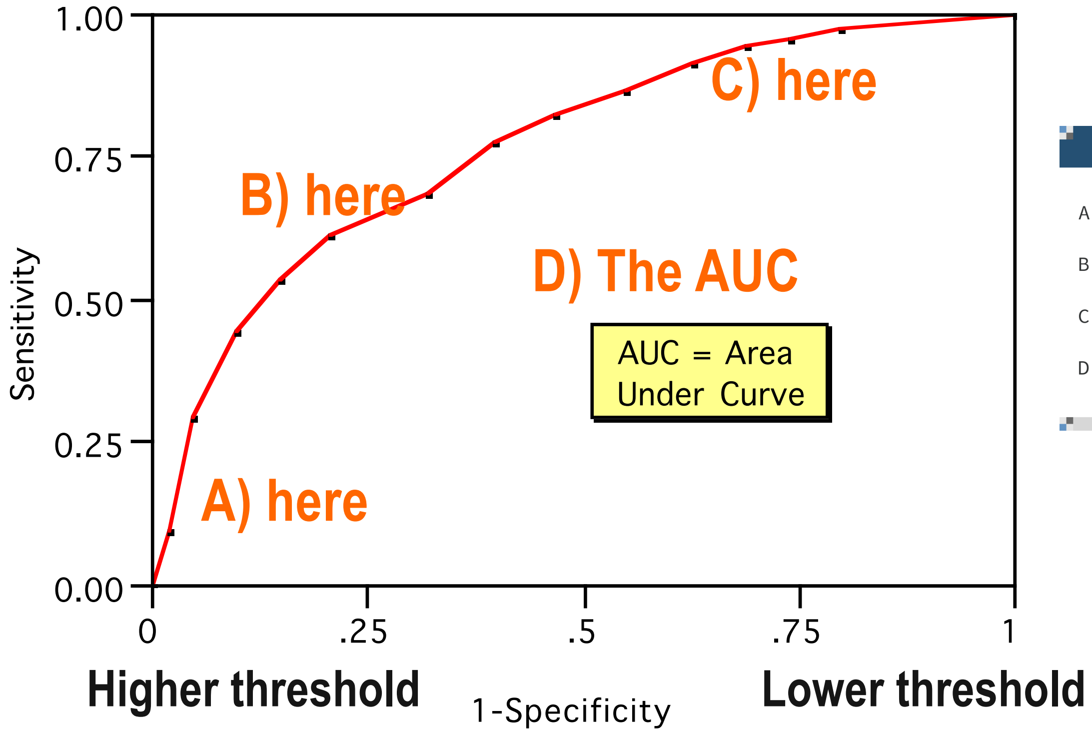
ROC charts support comparison

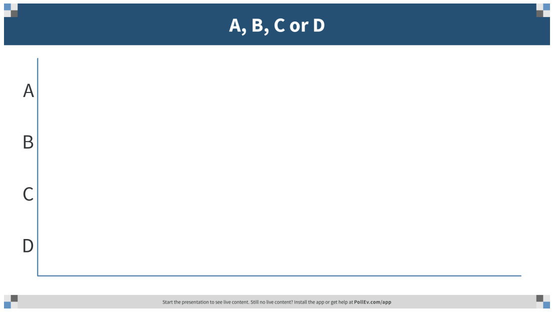


AUC = 0.5 is random guessing
AUC = 1.0 is perfection

AUC = Area Under the Curve

Where does google care about?



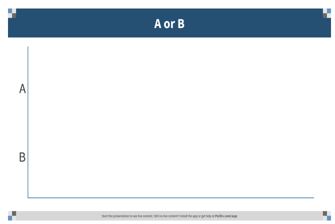


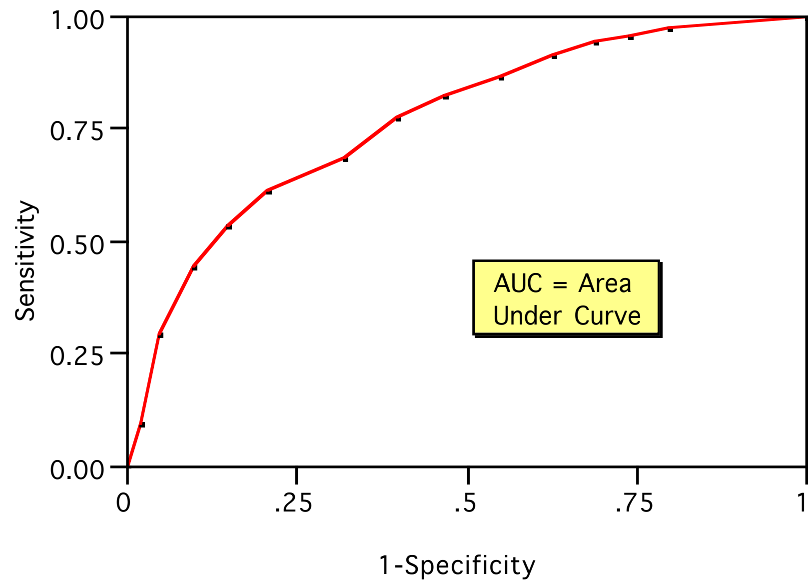
***p(y|x)* > *threshold* to be in class**

- **Which method is most likely to be better for generating an ROC curve?**

  A) Logistic regression

  B) SVM





The Truth

Test Score:	Has the disease	Does not have the disease	
Positive	True Positives (TP) a	b False Positives (FP)	**PPV =** $\frac{TP}{TP + FP}$
Negative	c False Negatives (FN)	d True Negatives (TN)	**NPV =** $\frac{TN}{TN + FN}$

Sensitivity

$$\frac{TP}{TP + FN}$$

Specificity

$$\frac{TN}{TN + FP}$$

Or,

$$\frac{a}{a + c} \qquad \frac{d}{d + b}$$

Confusion Matrix

- A confusion matrix shows the counts of the actual versus predicted class values.
- Example (overall accuracy rate of 73.9%)

		Actual Class		
		Class A	Class B	Class C
Predicted Class	Class A	20	5	2
	Class B	6	20	4
	Class C	4	2	25

For the confusion matrix

		Actual	
		purchase	no purchase
Predicted	purchase	10	60
	no purchase	20	200

- What is its precision?
- What is its recall?
- How do you

  a) increase precision (but decrease recall)

  b) increase both precision and recall

a) 10/20
b) 10/(10+20)
c) 10/60
d) 10/(10+60
e) other

Optimizing for true utility

- **Could one directly learn a model to optimize**
  - An asymmetric loss function?
  - AUC?

You should know

- **Probability vs. loss**
  - Often use model to estimate score; then threshold for decision
- **Loss function vs. utility function**
- **Confusion matrix:**
  - TP/TN/FP/FN or TPR/TNR/FPR/FNR
- **Precision, Recall, Sensitivity, Specificity, F1**
- **ROC curves**
  - AUC

Final project

- Pick a group of 2-4 students – and a team name
- Pick a problem and data set
- Look up related problems
- Formulate as an ML problem
- Run 3-5 methods, plus a baseline
  - Optimize hyperparameters
  - Table showing results
- What can you do that is clever?

Final project deliverables

- **11/16 Project proposal**
  - Give us enough information to give you feedback
- **11/27 Project checkpoint**
  - Show that you are making progress
- **12/9 Project report, code and notebook**

Real World ML

- Loss functions
- Model form
  - Feature engineering
- Visualization
- Causality: “what if?”

What you should know

- **Think about the true loss function**
  - Distinguish modeling from decision making
- **Think about the features**
  - What to you have? What can to get?
  - How should they be regularized (blocks)
- **Think about what ML methods fit best**
  - Compare several