# Classifier Evaluation and Fitting (Reading: 17.6 - 17.7)

**Learning goals:**

- Understand the drawbacks of accuracy as an evaluation metric for classification.
- Introduce the ROC curve and AUC metric.
- Derive the SGD update rule for logistic regression.

---

**UC Berkeley Data 100 Summer 2019**
**Sam Lau**

(Slides adapted from John DeNero)

CC BY NC SA

# Announcements

- HW6 due **today**
- Project 2 out today
  - Due next Tuesday, Aug 5.
- Small group tutoring: tinyurl.com/d100-tutor-week6
  - Suggest signing up if you're having trouble understanding the concepts in class.
  - If you get everything and want a challenge, come see us during OH.

CC BY NC SA

# Classifiers and Decisions

BY NC SA

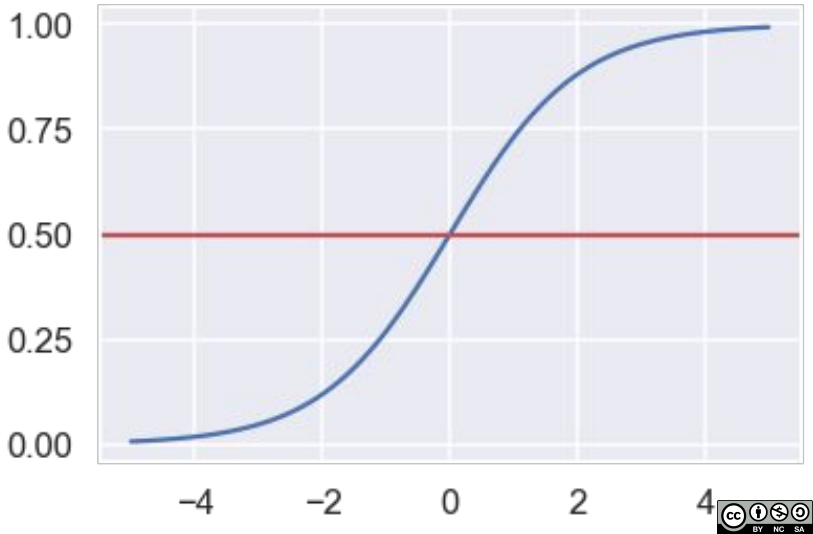
# Decision Rules

- A classifier is a function outputs a prediction of y: 0 or 1.
- Logistic regression finds a function that estimates P(Y=1|X).
- Need a **decision rule** (aka **classification rule**) to convert from probability to class. Most commonly:

$$f(x) = \begin{cases} 1 & \text{if } P(Y=1|X=x) \geq \frac{1}{2} \\ 0 & \text{if } P(Y=1|X=x) < \frac{1}{2} \end{cases}$$

Which is equivalent to:

$$f(x) = \begin{cases} 1 & \text{if } \boldsymbol{\theta} \cdot \boldsymbol{x} \geq 0, \\ 0 & \text{if } \boldsymbol{\theta} \cdot \boldsymbol{x} < 0 \end{cases}$$



# Decision Cutoffs

- We don’t have to choose 1/2 as our decision rule cutoff.
- Can change to avoid certain types of errors.
  - E.g. Set higher if we want to avoid false positives
- Appropriate decision rules depend on domain!
  - E.g. If accurate predictions very important, report “can’t decide” if predicted probability too close to 0.5.

CC BY NC SA

# Evaluating Classifiers

BY NC SA

# Evaluating Classifiers

| Prediction \ Truth | 1 | 0 |
|---|---|---|
| 1 | True positive (TP) | False positive (FP) |
| 0 | False negative (FN) | True negative (TN) |

How I remember these:
**Accuracy:** Proportion correct
**Error:** 1 - Accuracy
**Precision:** Of predicted 1s, what prop was actually 1? Aka: how precise were my predictions?
**Recall:** Of actual 1s, what prop was predicted 1? Aka: how many examples did my classifier recall?

**Accuracy**: (TP + TN) / n

**Error rate**: (FP + FN) / n

The most obvious/common evaluation metric

**Precision**: TP / (TP + FP)

**Recall**: TP / (TP + FN)

Used when detecting rare outcomes

CC BY NC SA

# The Problem with Accuracy

Suppose we have 100 emails and only 5 of them are spam.

What is accuracy of classifier that only predicts “not spam”? (Assume that “spam” is labeled 1.)

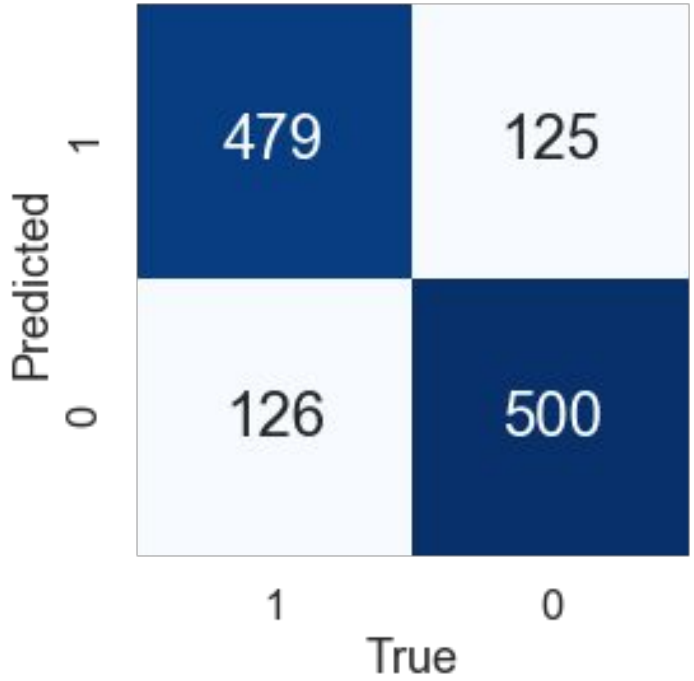
Because of this, we don’t rely on accuracy alone to evaluate models.

| | | Truth: 1 | Truth: 0 |
|---|---|---|---|
| Prediction | 1 | True positive (TP) | False positive (FP) |
| Prediction | 0 | False negative (FN) | True negative (TN) |

**Accuracy:** Proportion correct
**Error:** 1 - Accuracy
**Precision:** Of predicted 1s, what prop was actually 1? Aka: how precise were my predictions?
**Recall:** Of actual 1s, what prop was predicted 1? Aka: how many examples did my classifier recall?

# Confusion Matrices

**Confusion matrix** shows predicted vs. true classes:



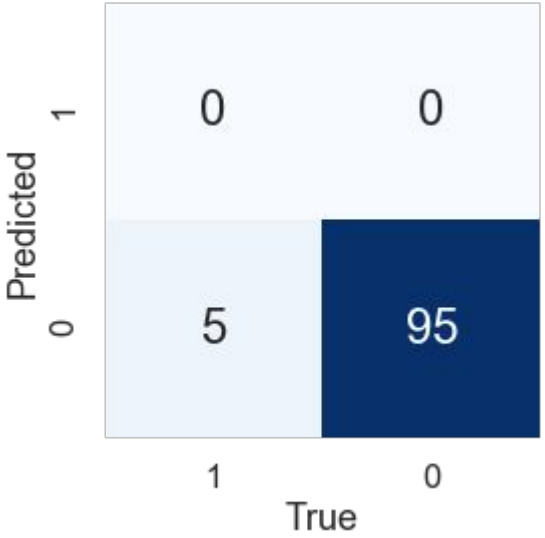

| | Truth: 1 | Truth: 0 |
|---|---|---|
| Prediction: 1 | True positive (TP) | False positive (FP) |
| Prediction: 0 | False negative (FN) | True negative (TN) |

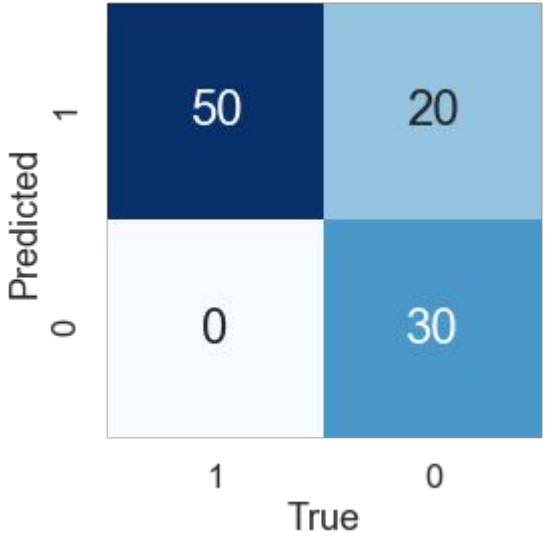
Be careful; packages give confusion matrix in different formats (e.g. sklearn gives 0 first, then 1).

**(Demo)**

# Practice with Metrics

Find accuracy, precision, recall for the following classifiers.



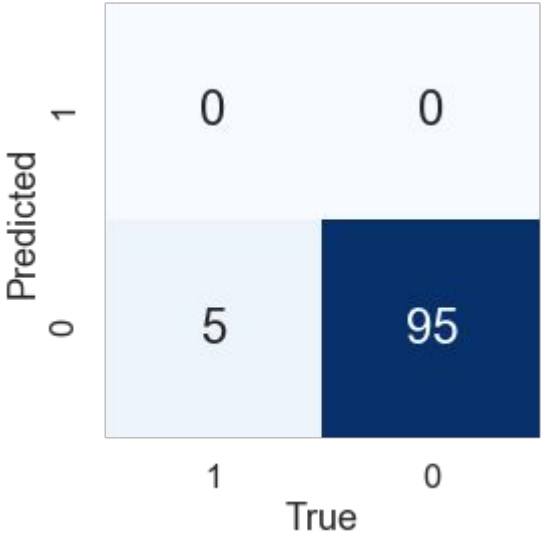

**Accuracy:** Proportion correct

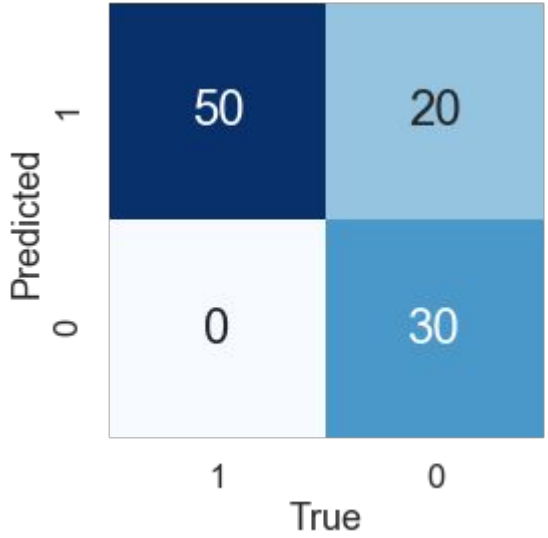
**Precision:** Of predicted 1s, what prop was actually 1? Aka: how precise were my predictions?

**Recall:** Of actual 1s, what prop was predicted 1? Aka: how many examples did my classifier recall?

# Practice with Metrics

Find accuracy, precision, recall for the following classifiers.




| | | |
|---|---|---|
| **Accuracy:** | 0.95 | 0.8 |
| **Precision:** | 0 (or undefined) | 50/70 = 0.71 |
| **Recall:** | 0 | 50/50 = 1.00 |

**Accuracy:** Proportion correct

**Precision:** Of predicted 1s, what prop was actually 1? Aka: how precise were my predictions?

**Recall:** Of actual 1s, what prop was predicted 1? Aka: how many examples did my classifier recall?

# FPR and TPR

False Positive Rate (FPR):

- FP / (FP + TN)
- “What prop of ham emails did I label spam?”

True Positive Rate (TPR):

- TP / (TP + FN)
- “What prop of spam emails did I label spam?”
- Note that TPR = recall.

| | | Truth: 1 | Truth: 0 |
|---|---|---|---|
| Prediction | 1 | True positive (TP) | False positive (FP) |
| | 0 | False negative (FN) | True negative (TN) |

**Yes, there are a lot of metrics! Sadly, this is the state of the world; in fact, there are even more metrics that we won’t cover.**

CC BY NC SA

# Decision Cutoffs

Changing decision rule cutoffs changes FPR and TPR.

If decision rule cutoff is set to 0, then all points labeled 1.

If decision rule cutoff is set to 1, all points labeled 0.

What's FPR and TPR for each of these cases?

| | | Truth | |
|---|---|---|---|
| | | **1** | **0** |
| **Prediction** | **1** | True positive (TP) | False positive (FP) |
| | **0** | False negative (FN) | True negative (TN) |

**FPR:** FP / (FP + TN), or "what prop of ham emails did I label spam?"

**TPR:** TP / (TP + FN), or "what prop of spam emails did I label spam?"

CC BY NC SA

# ROC Curves

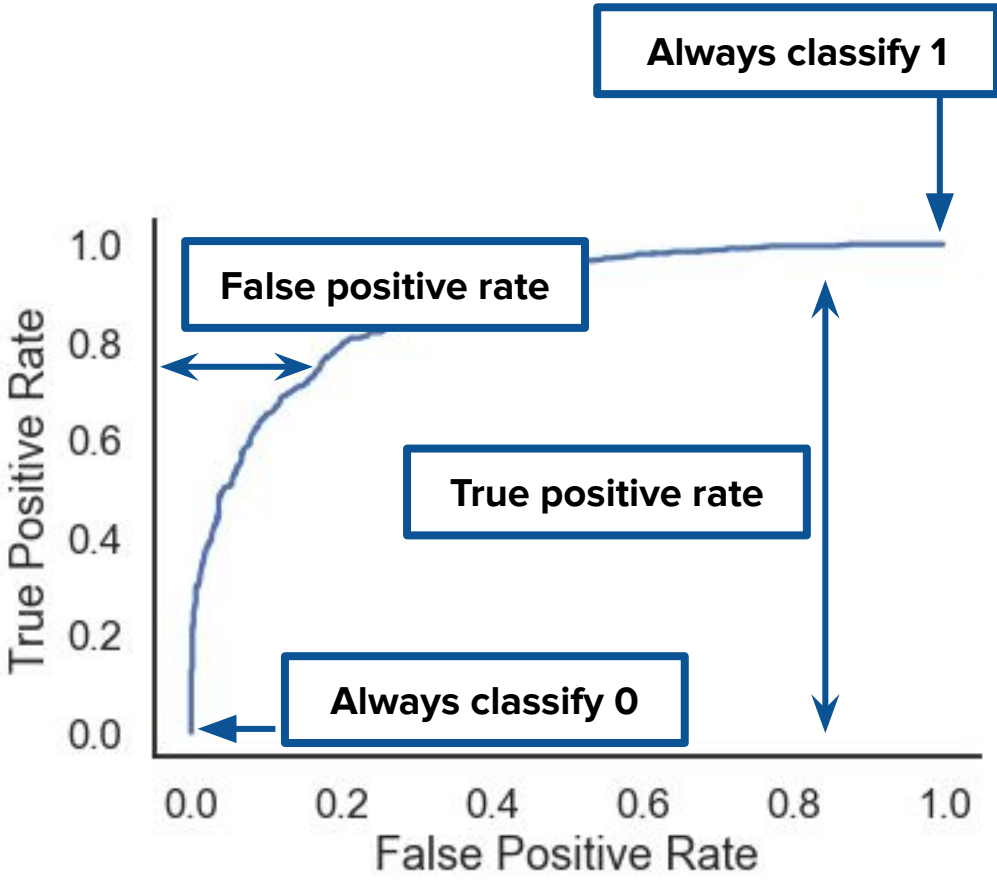
Decreasing the decision cutoff increases TPR but also increases FPR!

An ROC curve shows this tradeoff.

X-axis: False positive rate

Y-axis: True positive rate

**(Demo)**



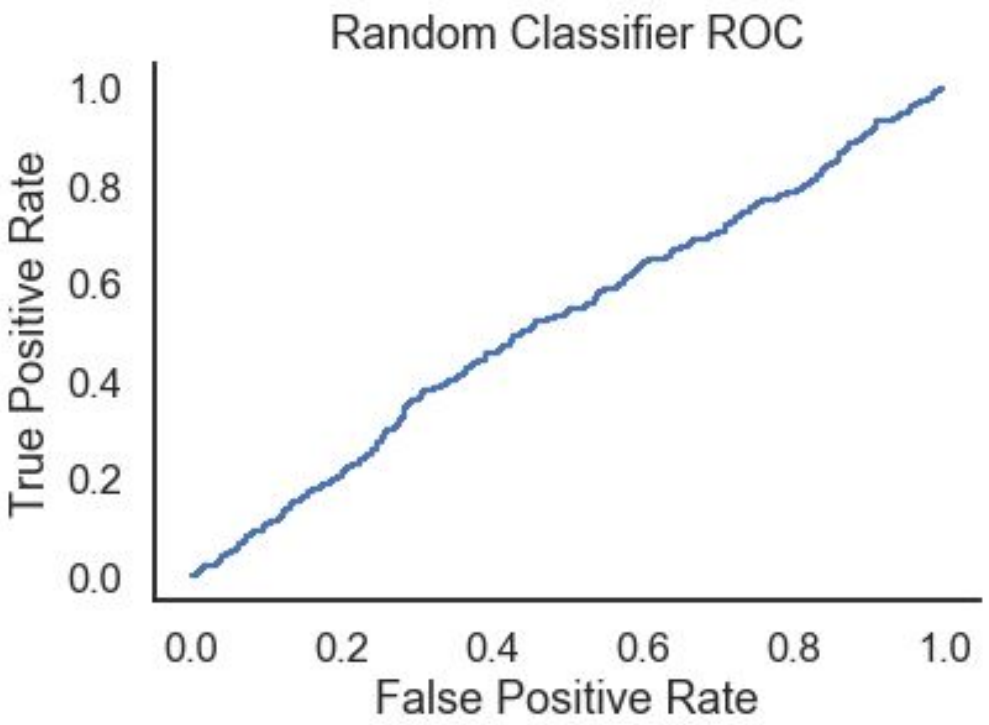

**In real life, use test set to pick point on plot which gets acceptable balance between FPR and FNR.**

# ROC Curves

Random classifier will have a diagonal line through plot. Why?

Area Under Curve (AUC) metric combines both FPR and FNR.

Random model has AUC = 0.5 so your model had better do better!




CC BY NC SA

# Break!
# Fill out Attendance:
# http://bit.ly/at-d100

BY NC SA

# Gradient Descent for Logistic Regression

BY NC SA

# Stochastic Gradient Descent Rule

Let's derive the SGD rule for logistic regression.

First, compute the gradient of the sigmoid function:

$$\sigma(t) = \frac{1}{1 + \exp(-t)}$$

$$\nabla_t \sigma(t) = ?$$

$$\nabla_t \sigma(t) = \frac{\exp(-t)}{(1 + \exp(-t))^2}$$

$$\nabla_t \sigma(t) = \sigma(t)(1 - \sigma(t))$$

**Neat simplification!**

I'm skipping most intermediate steps today, but you should know how to derive these results.

CC BY NC SA

# Stochastic Gradient Descent Rule

$$\nabla_t \sigma(t) = \sigma(t)(1 - \sigma(t))$$

Let $\sigma_i = f_{\boldsymbol{\theta}}(\boldsymbol{X_i}) = \sigma(\boldsymbol{X_i} \cdot \boldsymbol{\theta})$.

$$\nabla_{\boldsymbol{\theta}} \sigma_i = \sigma_i(1 - \sigma_i)(\boldsymbol{X_i}) \qquad \text{Since } \nabla_{\boldsymbol{\theta}}(\boldsymbol{X_i} \cdot \boldsymbol{\theta}) = \boldsymbol{X_i}$$

$$\nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta}, \boldsymbol{X_i}, y_i) = -y_i \log \sigma_i - (1 - y_i) \log(1 - \sigma_i)$$

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}} \ell &= -\frac{y_i}{\sigma_i} \nabla_{\boldsymbol{\theta}} \sigma_i - \frac{1 - y_i}{1 - \sigma_i}(-1) \nabla_{\boldsymbol{\theta}} \sigma_i \\
&= -(y_i - \sigma_i) \boldsymbol{X_i} \\
&= -(y_i - \sigma(\boldsymbol{X_i} \cdot \boldsymbol{\theta})) \boldsymbol{X_i}
\end{aligned}$$

CC BY NC SA

# Stochastic Gradient Descent Rule

$$\nabla_{\boldsymbol{\theta}} \ell = -(y_i - \sigma(\boldsymbol{X_i} \cdot \boldsymbol{\theta}))\boldsymbol{X_i}$$

**Scalar (p x 1) = (p x 1), as desired**

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \alpha \cdot \nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta}^{(t)}, \boldsymbol{X_i}, y)$$

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \alpha \cdot \left[-(y_i - \sigma(\boldsymbol{X_i} \cdot \boldsymbol{\theta}^{(t)}))\boldsymbol{X_i}\right]$$

CC BY NC SA

# Numerical Optimization for Logistic Regression

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \alpha \cdot \left[ -(y_i - \sigma(\boldsymbol{X_i} \cdot \boldsymbol{\theta}^{(t)}))\boldsymbol{X_i} \right]$$

- Use (S)GD since no closed-form solution for log reg.
- A few tricks for helping GD converge:
  - Start with high learning rate, then decrease it as iteration count increases (called **learning rate decay**).
  - Clip gradients if they get too far from 0 (called **gradient clipping**).

**(Demo)**

Lots more tricks for GD. Many of them are ad-hoc! GD theory is an active research area.

CC BY NC SA

# Summary

- Many metrics to evaluate classifiers!
  - Don’t memorize, but know their names so you can look them up.
- ROC curves (and AUC metric) particularly useful because it lets us decide tradeoff between FPR and TPR.
  - Learn how to draw and interpret an ROC curve.
- Practice the derivation for the gradient descent update rule for logistic regression until it becomes second-nature.

CC BY NC SA