Performance Measures For Logistic Regression Models

The Confusion Matrix

		Value Predicted By Classifier: 1	Value Predicted By Classifier: 0
Actual Value	1	TP	FN
	0	FP	TN

TN = True Negative

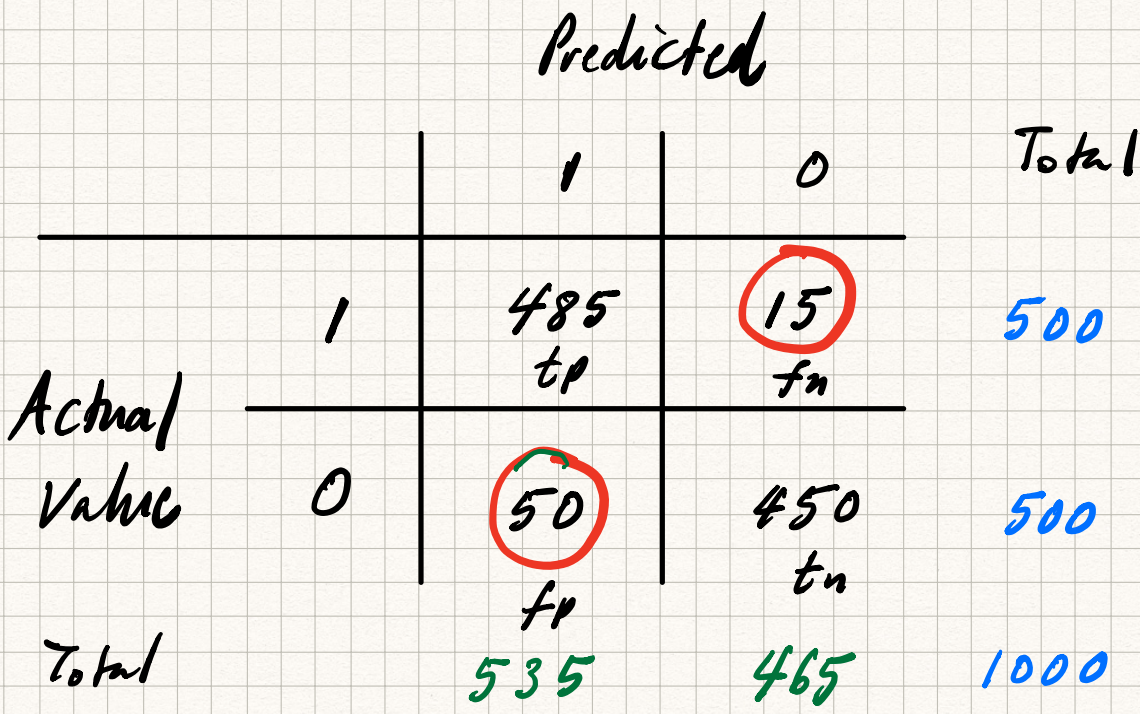
FP = False Positive

FN = False Negative

TP = True Positive

Maximize

Minimize



Goals:

Increase tp → Decrease fn

&

Increase tn → Decrease fp

Make the classifier better at detecting 1s

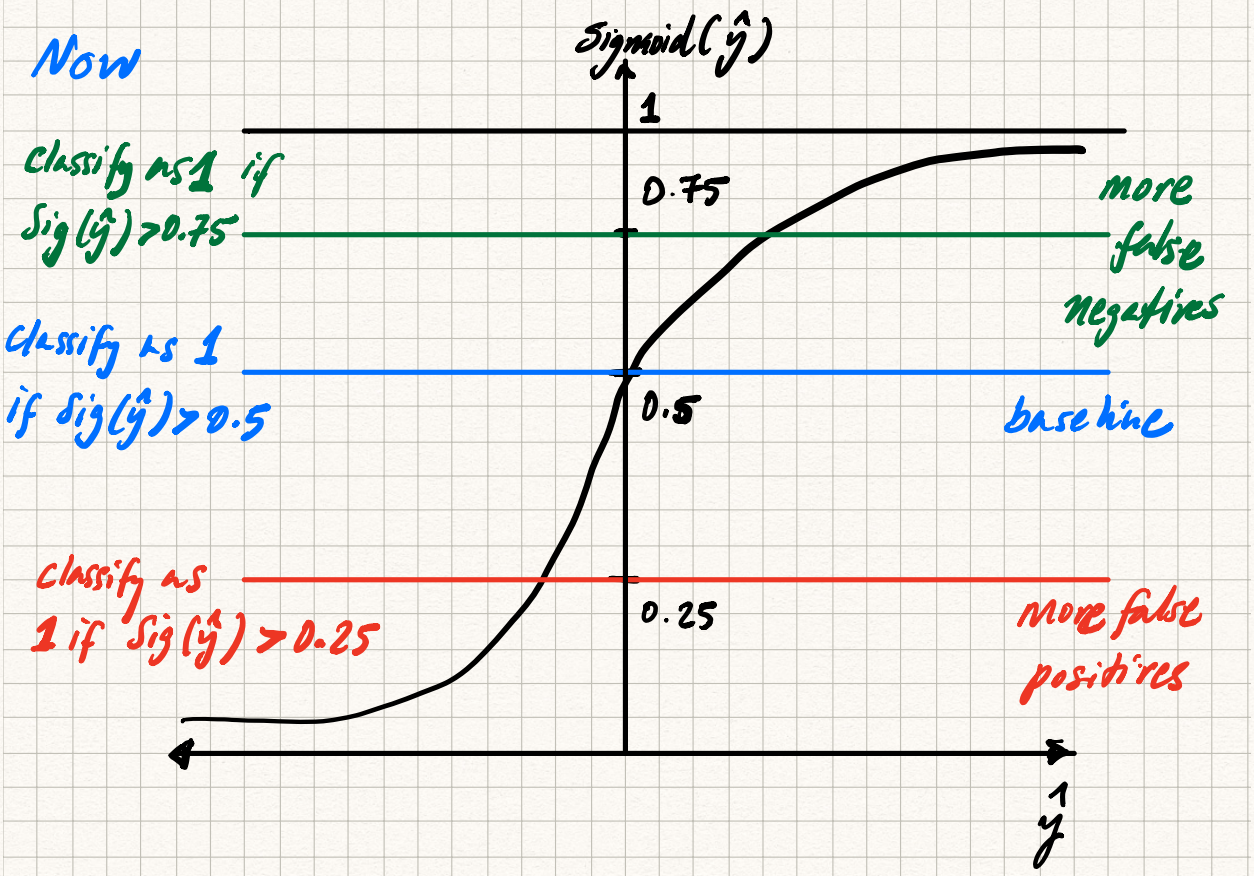
&

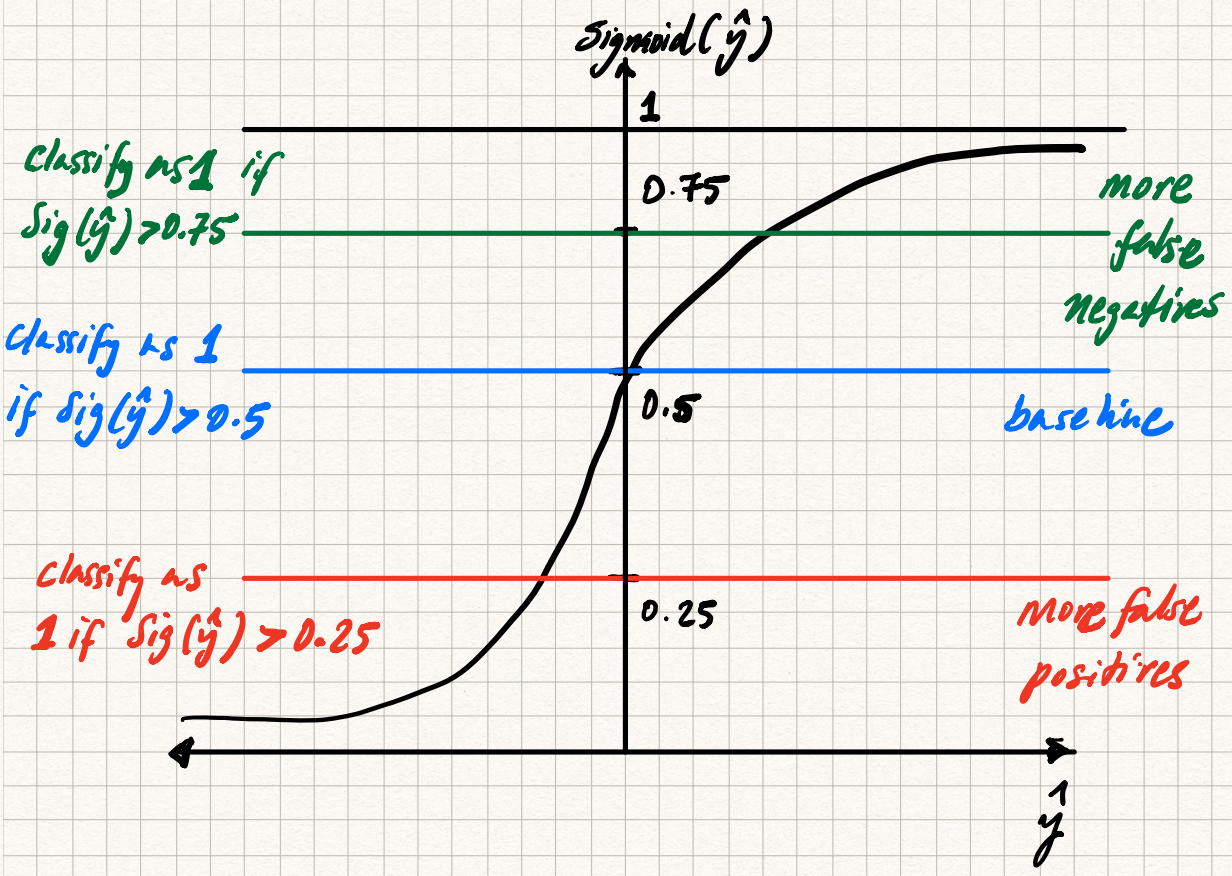
Make the classifier better at detecting 0s

The ways to do this:

Later
- get more data
- Create "better" features
- tune the models' hyperparameters

Now
- change the sigmoid threshold





Actual \ Predicted	1	0	
1	$tp\downarrow$	$fn\uparrow$	$tp+fn$
0	$fp\uparrow$	$tn\downarrow$	$fp+tn$
	$tp+fp$	$fn+tn$	

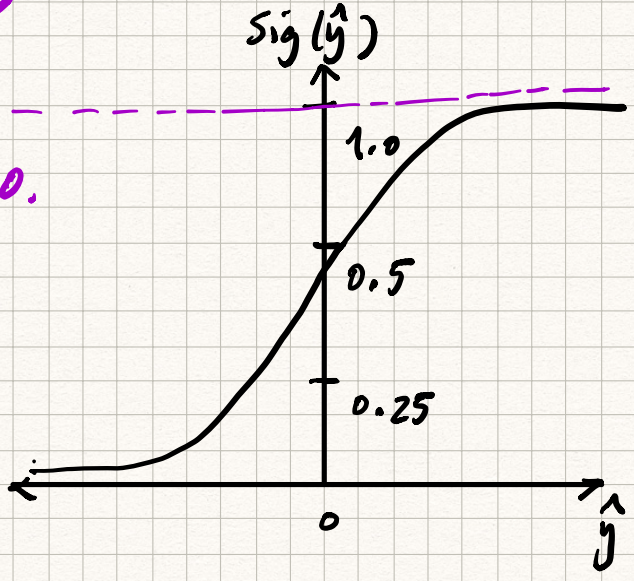
Think of moving the threshold down starting at $sig(\hat{y}) = 1$.

Precision

tolerance for false negatives at the cost of fewer true positives

One extreme

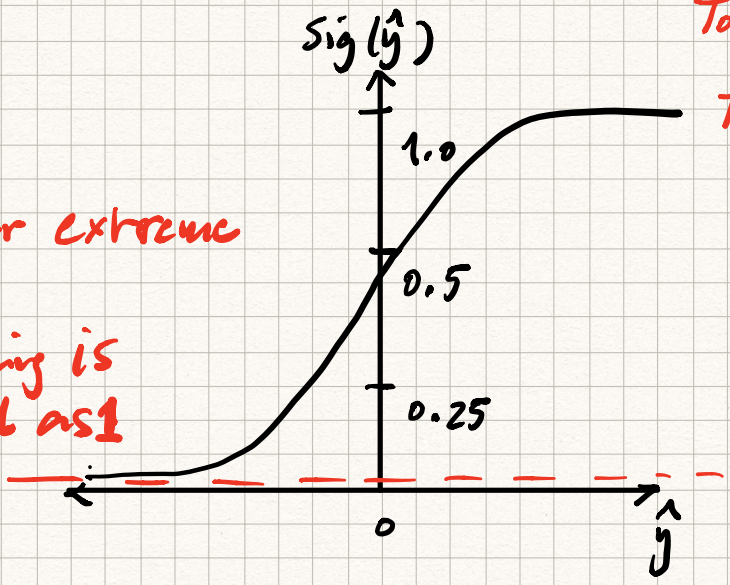
everything is classified 0.



Actual \ Predicted	1	0	
1	~~tp~~ 0	fn	tp + fn
0	~~fp~~ 0	tn	fp + tn
	tp + fp	fn + tn	

Recall

Tolerance for false positives at the cost of fewer true negatives



The other extreme

everything is classified as 1

Actual \ Predicted	1	0	
1	tp	fn ~~0~~	tp + fn
0	fp	tn ~~0~~	fp + tn
	tp + fp	fn + tn	

Measuring Classifier Performance

$$\text{Precision} = \frac{tp}{tp + fp}$$

when the classifier says it's positive, how often is it correct?

$$\text{Recall} = \frac{tp}{tp + fn}$$

of all the positives in the data set, what proportion of positives does the classifier pick up?

Measuring classifier performance (Contd.)

F1 is a measure combining precision & recall

$$F1 = 2 \cdot \frac{Precision * Recall}{Precision + Recall}$$

And finally,

$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn}$$

Accuracy can be tricky, especially when the dataset's target feature is imbalanced.

When the incidence rate is very low or very high, accuracy is not a good measure of classifier performance.

Ready for a surefire scheme?

...

Idea Make a TSA detector that detects if a person is not allowed to board a plane. Detector is >99% accurate!

Going to sell it to LAX and make a lot of money!

Some facts about LAX:

- 1,000 of every 1 million passengers who fly out of LAX should not be allowed to board (low incidence rate).
- 10 million passengers fly out of LAX every year.

Here's how I'm going to build my detector. I'm randomly going to pick 1000 passengers as those that are not allowed to fly.

		Predict 0	Predict 1	
Actual	0	tn 9,989,001	fp 999	9,990,000
	1	fn 9,999	tp 1	10,000
			1,000	

← Based on the facts.

← randomly chosen

$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

$$= \frac{1 + 9,989,001}{10,000,000}$$

$$= 99.89\%$$

But notice that precision and recall will be low and point out the silliness of this scheme.

$$\text{Precision} = \frac{tp}{tp+fp} = \frac{1}{1+999} = 0.001$$

$$= \boxed{0.1\%}$$

$$\text{Recall} = \frac{tp}{tp+fn} = \frac{1}{1+9,999} = 0.0001$$

$$= \boxed{0.01\%}$$

SUMMARY

- The performance of a model depends on a number of factors
  - the amount of data
  - the features of the dataset
  - the model's hyperparameters
  - the decision threshold (where to set the Sigmoid value to classify into one group or the other)

- Try a number models (with default hyperparameters) on a given dataset.

- Measure the performance of logistic regression models using Precision, Recall, and/or F Scores. (The higher the better.)
- Get insight into where a model is doing badly by using the confusion matrix.
- Measure and compare model performance using the <u>training</u> dataset.

- Make predictions on the test dataset.