UCI Paul Merage School of Business

Leadership for a Digitally Driven World™

MFIN 290: Financial Econometrics

Lecture 7-2

Possible Outcomes

	Actual Default	Actual No Default
Predict Default	.005	.015
Predict No Default	.05	0.93

Possible Outcomes

	Actual Default	Actual No Default
Predict Default	True Positive	False Positive
Predict No Default	False Negative	True Negative

Accuracy: How often does the model get it right, either by correctly classifying defaults or non-defaults?

$= (TP+TN)/Total$

Possible Outcomes

	Actual Default	Actual No Default
Predict Default	.005	.015
Predict No Default	.05	0.93

Accuracy = 0.935

Possible Outcomes

	Actual Default	Actual No Default
Predict Default	True Positive	False Positive
Predict No Default	False Negative	True Negative

We call the True Positive Rate the share of true total defaulters actually classified as such by the model:

TPR = TP/(TP+FN)

This is also called the "Sensitivity" or the "Recall" of the model

Possible Outcomes

	Actual Default	Actual No Default
Predict Default	.005	.015
Predict No Default	.05	0.93

TPR = 0.005/0.055

Possible Outcomes

	Actual Default	Actual No Default
Predict Default	True Positive	False Positive
Predict No Default	False Negative	True Negative

We call the Precision the share of total defaulters predicted by the model that are actually defaulters:

Precision = TP/(TP+FP)

Possible Outcomes

	Actual Default	Actual No Default
Predict Default	.005	.015
Predict No Default	.05	0.93

Precision = 0.005/0.02

Possible Outcomes

	Actual Default	Actual No Default
Predict Default	True Positive	False Positive
Predict No Default	False Negative	True Negative

We call the Specificity the share of total true non defaulters actually classified as such by the model:

Specificity = TN/(TN+FP)

Relevant for current conditions

Possible Outcomes

	Actual Default	Actual No Default
Predict Default	.005	.015
Predict No Default	.05	0.93

Specificity = 0.93/0.945

Possible Outcomes

	Actual Default	Actual No Default
Predict Default	True Positive	False Positive
Predict No Default	False Negative	True Negative

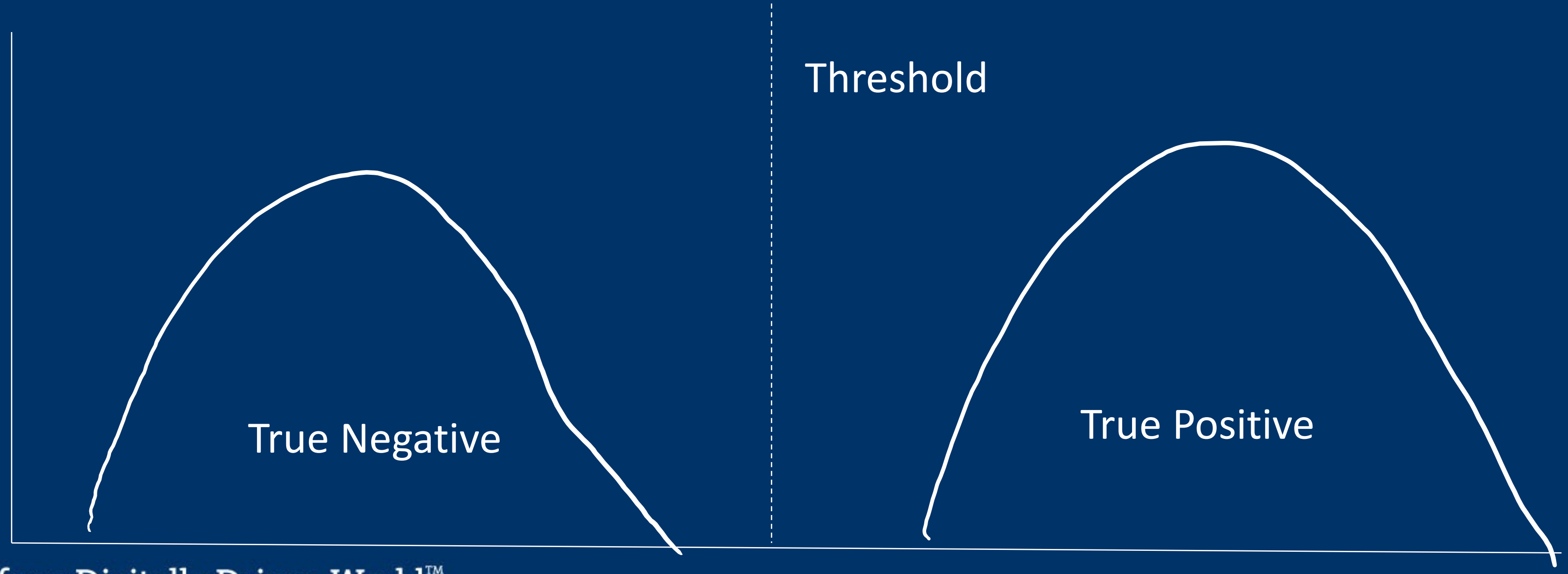
We call the False Positive Rate the share of total non defaulters incorrectly actually classified as such by the model:

FPR = FP/(TN+FP) = 1-Specificity

Possible Outcomes: Example

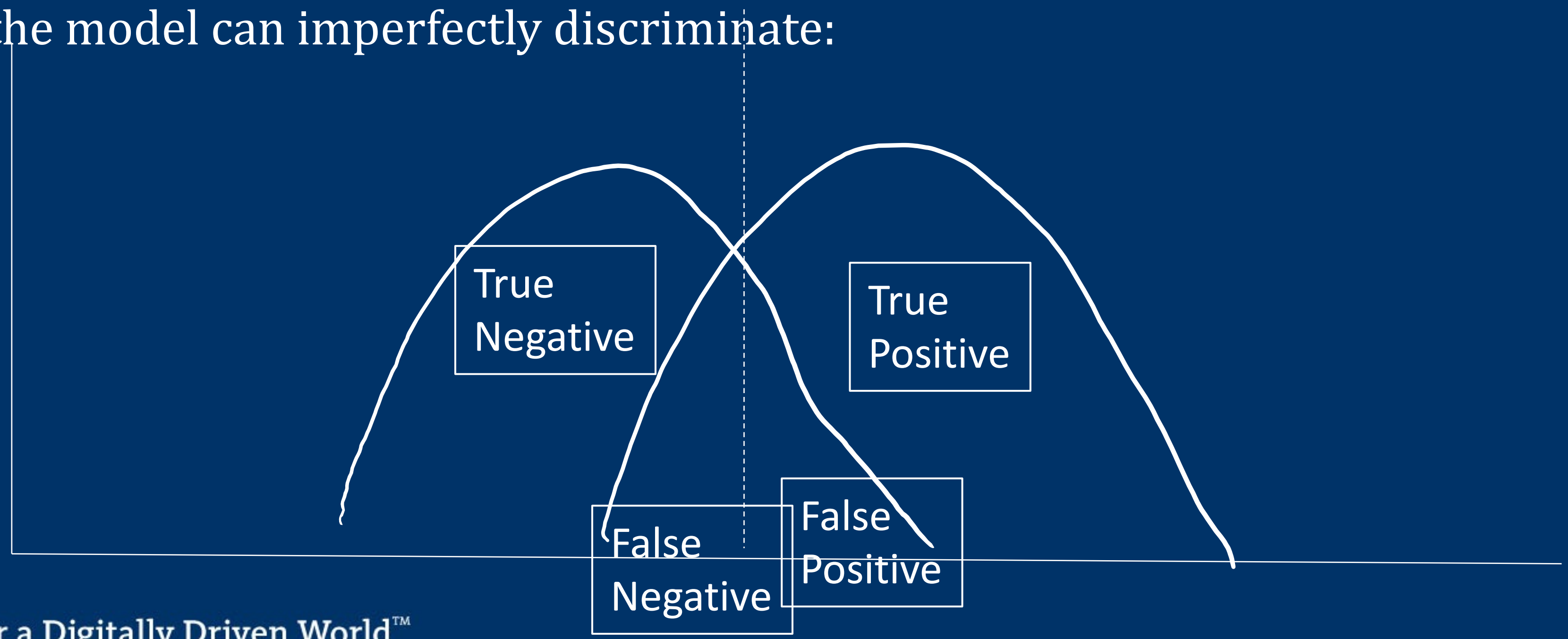
- Think about the separating power of our model between True Positives and True Negatives

- If the model can perfectly discriminate:



Possible Outcomes: Example

- Think about the separating power of our model between True Positives and True Negatives
- If the model can imperfectly discriminate:



Possible Outcomes: Example

- Think about the separating power of our model between True Positives and True Negatives

- As we move this threshold, we are trading off the TPR and FPR. In the limit, the model can't differentiate at all and these graphs will overlap.

Relationships

- So, when we increase sensitivity (=TP/(TP+FN)), we *necessarily* decrease specificity (= TN/(TN+FP))

- Decreasing the classification threshold will increase the number of values classified as a default by the model, which will increase the sensitivity (assuming some of those actually defaulted!) and vice versa.

- Since FPR = 1-Specificity, increasing the true positive rate increases the false positive rate. There is no free lunch.

Possible Outcomes: Example

	Actual Default	Actual No Default
Predict Default	.005	.015
Predict No Default	.05	0.93

Accuracy $= 0.935$

(94.5% don't default… we would do better just saying no one defaulted!)

TPR or Recall =

$TP/(TP+FN) = 0.005/(0.055) = 0.09$

Possible Outcomes: Example

	Actual Default	Actual No Default
Predict Default	.005	.015
Predict No Default	.05	0.93

Precision =

$$TP/(TP+FP) = 0.005/(0.02) = 0.25$$

False Positive Rate =

$$FP/(TN+FP) = 0.015/(0.93+0.015) = 0.0158$$

Receiver Operating Characteristic

- The Receiver Operating Characteristic, or ROC curve, is created by plotting the TPR (y axis) against the FPR (x axis). We know the relationship will be (weakly) increasing.

- A perfect model will have all true positives and zero false positives everywhere and the curve will jump to one immediately and stay flat

- If the model is equal to a coin flip, there should be a 50/50 chance of getting a TPR or FPR, so this should increase along the 45 degree line

Receiver Operating Characteristic

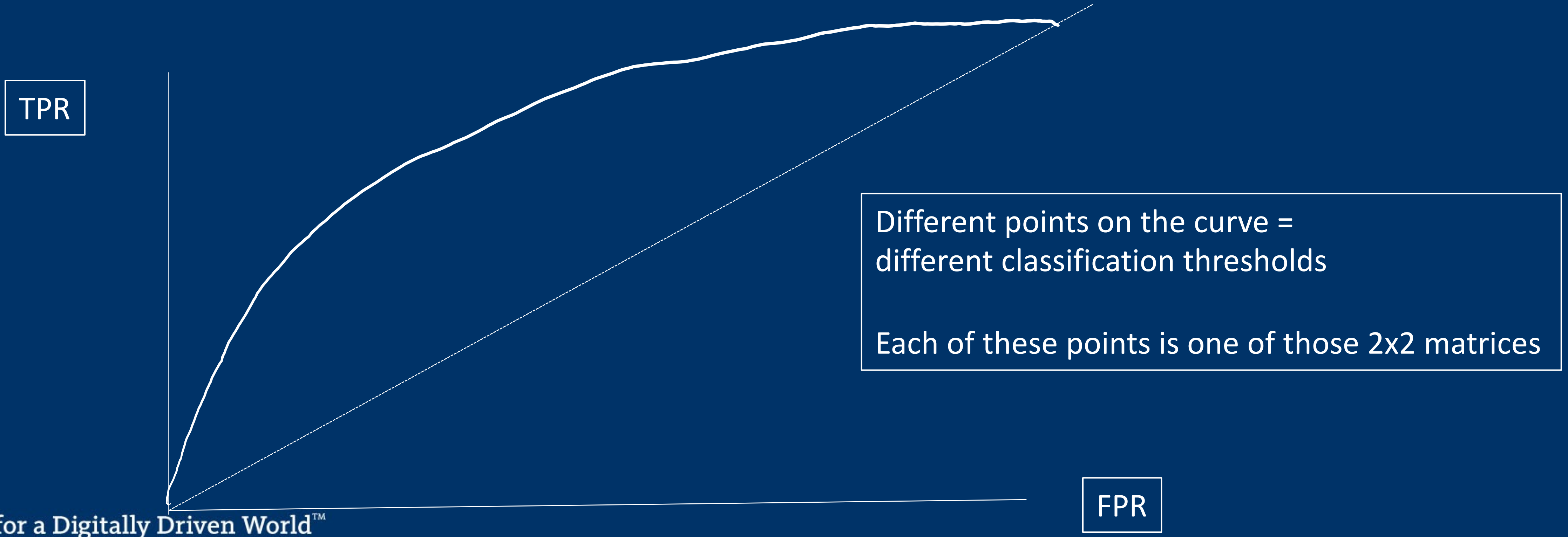
- Often summarized by the Area Under the Curve (AUC): should be between 0.5 and 1 (if it isn't, flip your classifier!)

- Interpretation of AUC is surprisingly straightforward: If you take all possible pairs of defaulters and non-defaulters in the data, the AUC is the percentage of the time where the defaulter has a higher predicted model default rate! Has nothing to do with if the numbers are "right" – will return to this.

- What is *good* for an AUC depends on your data: no two classification problems are equally easy/difficult to get right.

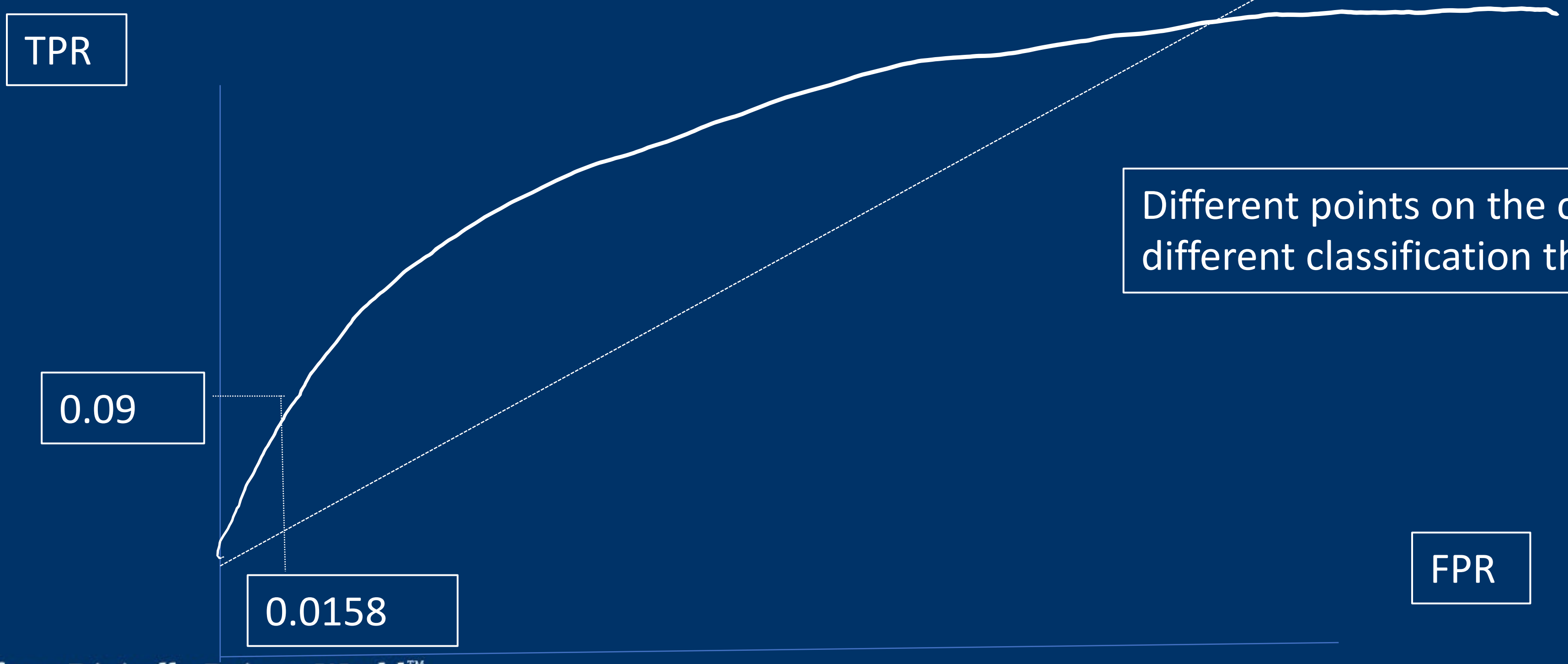
Receiver Operating Characteristic

- The Receiver Operating Characteristic, or ROC curve, is created by plotting the True Positive Pate (y axis) against the False Positive Rate (x axis). We know the relationship will be (weakly) increasing.



Receiver Operating Characteristic

- The Receiver Operating Characteristic, or ROC curve, is created by plotting the True Positive Pate (y axis) against the False Positive Rate (x axis). We know the relationship will be (weakly) increasing.



Different points on the curve = different classification thresholds

Receiver Operating Characteristic

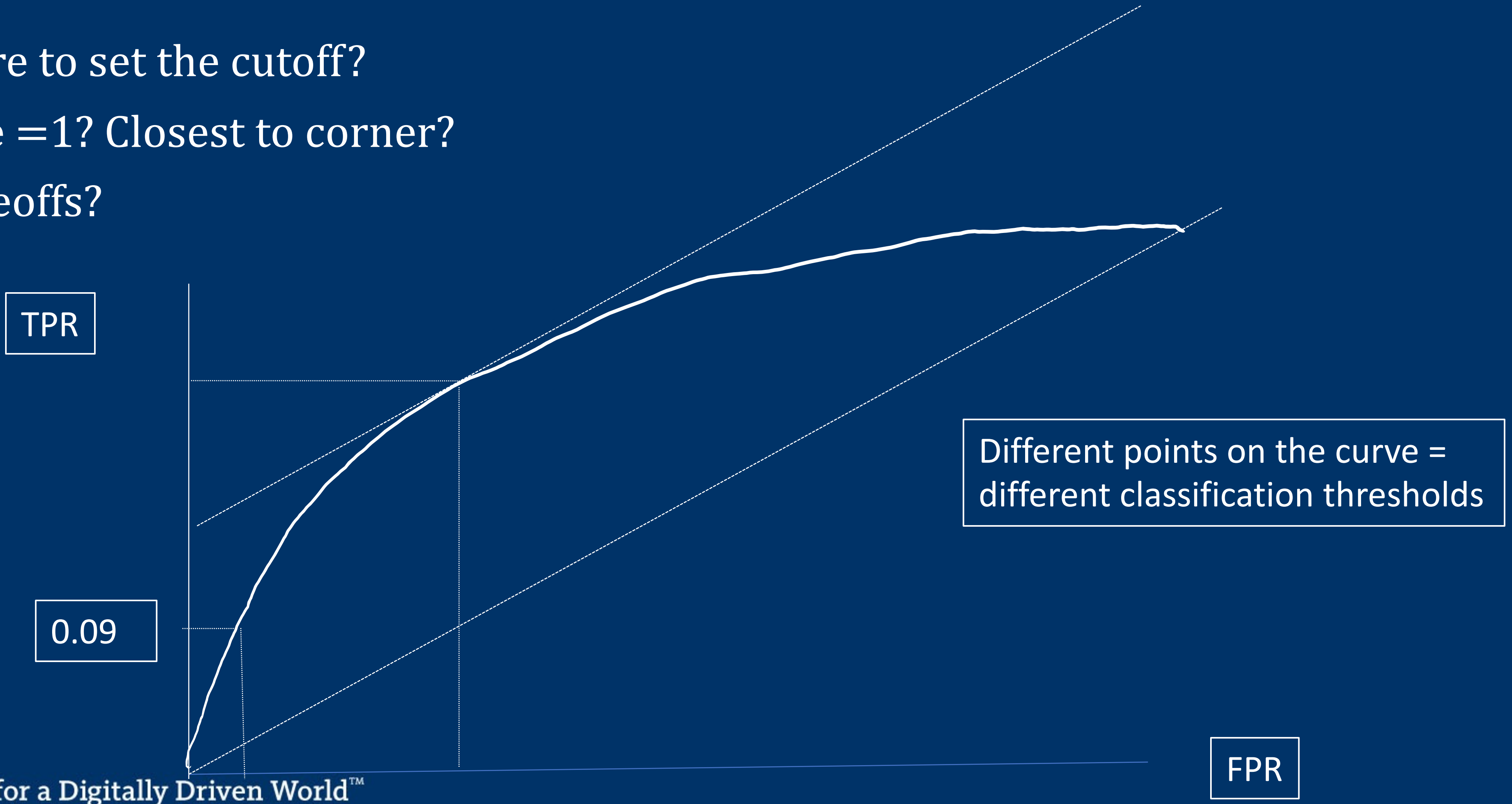
- The Receiver Operating Characteristic, or ROC curve, is created by plotting the TPR (y axis) against the FPR (x axis). We know the relationship will be (weakly) increasing.

- Note:
- This odd name comes from signal detection theory developed during WWII to help analyze radar.
- How well radar receiver operators performed this task was known as their Receiver Operating Characteristic.

Receiver Operating Characteristic

The Paul Merage School of Business

- Where to set the cutoff?
- Slope =1? Closest to corner?
- Tradeoffs?



Leadership for a Digitally Driven World™

Example: Default Data

- We've secured some proprietary default data for the problem set. I ask you to estimate a default model from the excel data.

- 1) Bring in the data
- 2) Generate financial ratios you care about (don't forget about equity!)
- 3) Estimate the relationship between def_flag and the explanatory variables
- 4) Test with ROC curves, PR information, goodness of fit tests (may have to do this manually depending on your software)
- Don't forget to just look at the data.. What do defaults look like over time, etc.

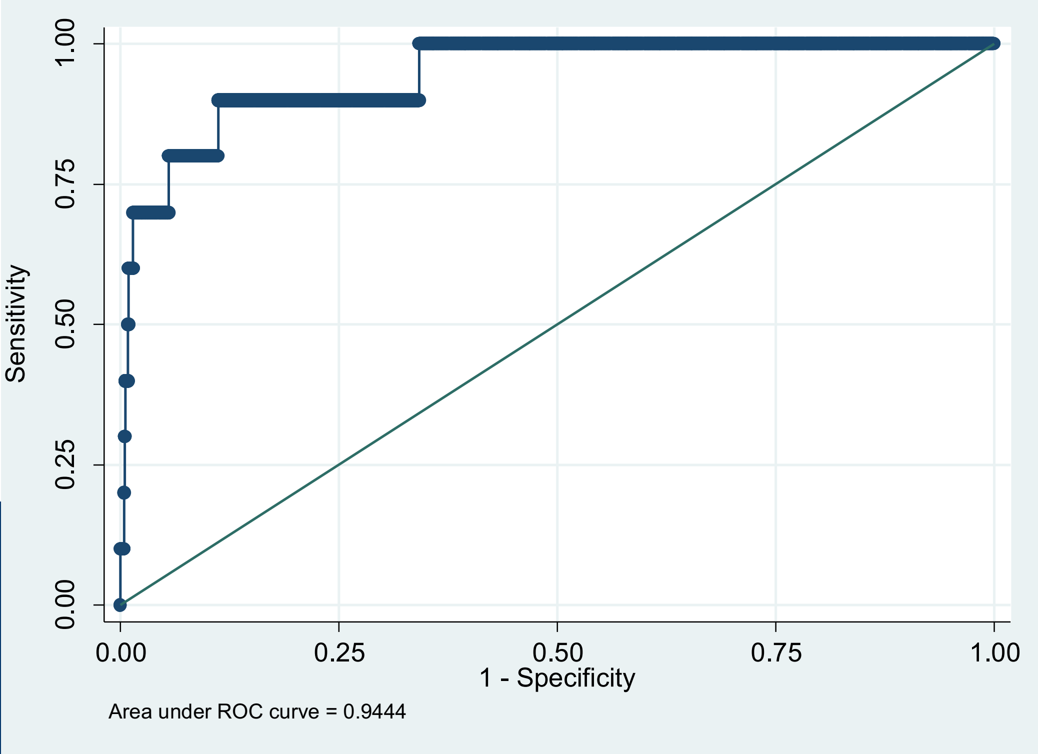
Default Data

```
*insheet using defaultdata.csv, comma names

gen equity = totassets-totliabs

gen roe = netincomeqtr/equity
gen al = totassets/totliabs
gen profit = netincomeqtr/salesqtr

logit def_flag profit al roe
lroc
```



What might be some other good ratios/data to acquire?

Ordinal v. Cardinal Tests

- Interpretation of AUC is surprisingly straightforward: If you take all possible pairs of defaulters and non-defaulters in the data, the AUC is the percentage of the time where the defaulter has a higher predicted model default rate!

- The interpretation of AUC highlights that this statistic gets at whether or not the defaulter has a higher default rate, not whether the predicted default rates are correct.

Ordinal v. Cardinal Tests

- The interpretation of AUC highlights that this statistic gets at whether or not the defaulter has a higher default rate, not whether the predicted default rates are correct.

- We know that we have consistency as a property of the MLE for things like logits, but this suggests that additional tests may be required to ensure that the model is producing results that are *cardinally* appropriate…

- That the loans who you think should default 5% of the time actually default 5% of the time, and not just more often than the 1% loans).

Hosmer – Lemeshow Test

- This intuition motivates the Hosmer Lemeshow test:
- 1) Separate data into k equally sized groups based on the predicted probabilities
- 2) Calculate the average actual rate in each group
- 3) Compare the results

The Paul Merage School of Business

Hosmer – Lemeshow Test

- 3) Compare the results using fit statistics (Pearson goodness of fit)

- $\sum_{j=1}^{K} \frac{(y_{1j} - \hat{y}_{1j})^2}{\hat{y}_{1j}} + \frac{(y_{0j} - \hat{y}_{0j})^2}{\hat{y}_{0j}}$
- Where
- y_{ij} = the actual number of observations in group j with classification i
- $\hat{y}_{ij}$ = the prediction number of observations in group j with classification i

Leadership for a Digitally Driven World™

Hosmer – Lemeshow Test

- 3) Compare the results using a fit statistics (Pearson goodness of fit)

- $\sum_{j=1}^{K} \frac{(y_{1j}-\hat{y}_{1j})^2}{\hat{y}_{1j}} + \frac{(y_{0j}-\hat{y}_{0j})^2}{\hat{y}_{0j}}$
- Where
- y_{ij} = the actual number of observations in group j with classification i
- $\hat{y}_{ij}$ = the prediction number of observations in group j with classification i

- If the model fits well ($y_{ij} \cong \hat{y}_{ij}$) across groups, this will be approximately χ^2 with k-2 df.

Default Data

```
. estat gof, table group(10)

Logistic model for def_flag, goodness-of-fit test

  (Table collapsed on quantiles of estimated probabilities)
```

Group	Prob	Obs_1	Exp_1	Obs_0	Exp_0	Total
1	0.0000	0	0.0	616	616.0	616
2	0.0000	0	0.0	616	616.0	616
3	0.0000	0	0.0	616	616.0	616
4	0.0001	0	0.0	616	616.0	616
5	0.0001	0	0.1	616	615.9	616
6	0.0003	0	0.1	616	615.9	616
7	0.0004	1	0.2	615	615.8	616
8	0.0010	0	0.4	616	615.6	616
9	0.0023	1	1.1	615	614.9	616
10	0.2996	8	8.1	608	607.9	616

```
      number of observations =      6160
            number of groups =        10
     Hosmer-Lemeshow chi2(8) =         3.76
                 Prob > chi2 =         0.8783
```

To run an HL test, type "estat gof" in Stata after you run a logit

Automatically splits into 10 groups, but you can tell it how many. Tells you how many are observed and expected in each bin.

The null here is that we are cardinally correct across all categories (right?), so we DON'T want to reject it!

Example

- Mortgage Delinquency Data, Las Vegas, 2008

- Bring in data, explore relationships

1. Estimate logit model
2. generate classification tables (“estat classification”, or manually)
3. plot ROC curve (“lroc”) and compare sensitivity/specificity (“lsens”)
4. HL test (“estat gof”)

5. What’s not in this data that should be??

Other Categorical models

- There are many other options when we have data other than simple binary dependent variables. Will provide a very brief overview of a few classic examples here that may be of interest depending on your research.

- This is not a complete introduction, and is here to mention that options exist for these types of data as well.

- Hill and Lim Chapter 16 has more information.

Ordered Multiple Choice Models

- What if we have more than one outcome?

- Suppose we are modeling credit ratings instead of defaults? Then we have an ordered relationship we can exploit when identifying the relationships

Ordered Multiple Choice Models

- This is used to construct a likelihood function.
- The changes in probabilities can be complex, as how large the errors need to be to go into each bucket change as the $x\beta$ (the first threshold) change.
- Section 16.5 for Probit models

Unordered Multiple Choice Models

- What if we are modeling a categorical outcome that isn't ordered (which stock to buy given market signals)? This would be an unordered, or multinomial choice model.

- Multinomial Logit: probability is relative to all other classes
- $\Pr\left(Y_j = j\right) = \frac{\exp(x'\beta_j)}{\sum_i \exp(x'\beta_i)}$

- Estimated through MLE. Section 16.3

Independence from Irrelevant Alternatives

- This specification implies that the odds of two outcomes
- $\frac{P_{ij}}{P_{im}}$ is independent of the other probabilities and not a function of P_{ij}

- This is called the independence from irrelevant alternatives assumption.

- A rather strong assumption that can be relaxed –and we can test it by excluding the alternatives from the outcomes and re-estimating the model (we should get the same P_{ij} and P_{im}).

Nested Models

- Sequential Decisions can relax some of our distributional assumptions.

- Perhaps a parent choice is made (whether or not to take a trip), then a second choice (where to go), and then a third (what kind of transportation to take).

- The probability of taking a plane conditional on where I am going is very different than the unconditional probability! Can take this into account with Nested models.

Count Models: Poisson

- Models where dependent variables represent counts have a distribution over {0,1,2,3,…}
- There are several distributions used here, though the Poisson is the most frequent.
- Often, the probabilities implied by the Poisson are too low for the zero case… (number of hardware failures, number of hospital admissions, number of fraud cases)

Two Step: Zero Inflated and Hurdle Models

- Can be addressed through a two-step procedure or a zero inflated model, where there is some additional probability of a zero, and conditional on seeing a number greater than zero, the distribution is Poisson.