

# COMMUNICATING RESULTS

*Adam Jones, PhD*

*Data Scientist @ Critical Juncture*

---

## COMMUNICATING RESULTS

---

# LEARNING OBJECTIVES

- Explain the trade-offs between the precision and recall of a model while articulating the cost of false positives vs. false negatives
- Describe the difference between visualization for presentations vs. exploratory data analysis
- Identify the components of a concise, convincing report and how they relate to specific audiences/stakeholders

---

**COURSE**

---

**PRE-WORK**

---

# PRE-WORK REVIEW

---

- Understand results from a confusion matrix and measure true positive rate and false positive rate
- Create and interpret results from a binary classification problem
- Know what a decision line is in logistic regression

---

**OPENING**

---

# COMMUNICATING RESULTS

---

## WE BUILT A MODEL! NOW WHAT?

---

- We've built our model, but there is still a **gap** between your notebook with plots/figures and a slideshow needed to present your results
- Classes so far have focused on two core concepts:
  - developing consistent practices
  - interpreting metrics to evaluate and improve model performance
- But what does that mean to your audience?

---

## WE BUILT A MODEL! NOW WHAT?

---

- Imagine how a non-technical audience might respond to the following statements:
  - The predictive model has an accuracy of 80%
  - Logistic regression was optimized with L2 regularization
  - Gender was more important than age in the predictive model because it has a larger coefficient
  - Here's the AUC chart that shows how well the model did

---

## WE BUILT A MODEL! NOW WHAT?

---

- Who is your audience? Are they technical? What are their concerns?
- **Remember:** in a business setting, you may be *the only person* who can interpret what you've built
- Some people may be familiar with basic visualization,
  - but you will likely have to do a lot of “hand holding”
- You need to be able to efficiently explain your results in a way that makes sense to **all** stakeholders (technical or not)



---

## **WE BUILT A MODEL! NOW WHAT?**

---

- Today, we'll focus on communicating results for “simpler” problems, but this applies to any type of model you may work with
- First, let's review classification metrics, review our knowledge, and talk about how we might communicate what we know

---

**REVIEW**

---

# BACK TO THE CONFUSION MATRIX

---

## BACK TO THE CONFUSION MATRIX

---

- Confusion matrices allow for the interpretation of correct and incorrect predictions for *each class label*
- It is the first step for the majority of classification metrics and goes deeper than just accuracy

# BACK TO THE CONFUSION MATRIX

- Let's recall our confusion matrix

		<u>True class</u>			
		<b>p</b>	<b>n</b>		
<u>Hypothesized class</u>	<b>Y</b>	True Positives	False Positives	$\text{fp rate} = \frac{FP}{N}$	$\text{tp rate} = \frac{TP}{P}$
	<b>N</b>	False Negatives	True Negatives	$\text{precision} = \frac{TP}{TP+FP}$	$\text{recall} = \frac{TP}{P}$
Column totals:		<b>P</b>	<b>N</b>	$\text{accuracy} = \frac{TP+TN}{P+N}$	
				$\text{F-measure} = \frac{2}{1/\text{precision}+1/\text{recall}}$	

---

# ACTIVITY: KNOWLEDGE CHECK

---

## ANSWER THE FOLLOWING QUESTIONS



### EXERCISE

1. How do we calculate the following?
  - a. Accuracy
  - b. True positive rate
  - c. False positive rate

## DELIVERABLE

Answers to the above questions

## **INTRODUCTION**

---

# **PRECISION AND RECALL**

---

# PRECISION AND RECALL

---

- Our previous metrics were primarily designed for less biased data problems:
  - we could be interested in **both outcomes**, so it was important to generalize our approach
- For example, we may be interested if a person will vote for a Republican or Democrat
  - This is a binary problem, but we're interested in both outcomes

---

# PRECISION AND RECALL

---

- Precision and recall, metrics built from the confusion matrix,
  - focus on *information retrieval*,
  - particularly when one class is more “interesting” than the other
- For example, we may want to predict if a person will be a customer
  - We care much more about people who *will* be a customer of ours than people who *won't*



---

# PRECISION AND RECALL

---

- *Precision* aims to product a high amount of relevancy instead of irrelevancy
  - “Out of all of our *positive predictions* (both true positive and false positive), how many were correct?”
- *Recall* aims to see how well a model returns specific data
  - (literally, checking whether the model can *recall* what a class label looked like)
  - “Out of all of our *positive class labels*, how many were correct?”

---

# ACTIVITY: KNOWLEDGE CHECK

---

## ANSWER THE FOLLOWING QUESTIONS



### EXERCISE

1. If the goal of the “recall” metric is to identify specific values of a class correctly, what other metric performs a similar calculation?

## DELIVERABLE

Answers to the above question

# THE MATH FOR RECALL

- Recall is the count of predicted *true positives* over the total count of that class label
- This is the same as True Positive Rate or *sensitivity*

		<u>True class</u>			
		<b>p</b>	<b>n</b>		
<u>Hypothesized class</u>	<b>Y</b>	True Positives	False Positives	fp rate = $\frac{FP}{N}$	tp rate = $\frac{TP}{P}$
	<b>N</b>	False Negatives	True Negatives	precision = $\frac{TP}{TP+FP}$	recall = $\frac{TP}{P}$
Column totals:		<b>P</b>	<b>N</b>	accuracy = $\frac{TP+TN}{P+N}$	
				F-measure = $\frac{2}{1/\text{precision} + 1/\text{recall}}$	

---

## THE MATH FOR RECALL

---

- Imagine predicting the color of a marble as either red or green
  - There are 10 of each
- If the model identifies 8 identifies 8 of the green marbles as green, the recall is  $8 / 10 = 0.80$
- However, this says nothing of the number of *red* marbles that are also identified as green

# THE MATH FOR PRECISION

- Precision, or positive predictive value, is calculated as the count of predicted true positives over the count of all values predicted to be positive

		<u>True class</u>			
		<b>P</b>	<b>N</b>		
<u>Hypothesized class</u>	<b>Y</b>	True Positives	False Positives	$fp\ rate = \frac{FP}{N}$	$tp\ rate = \frac{TP}{P}$
	<b>N</b>	False Negatives	True Negatives	$precision = \frac{TP}{TP+FP}$	$recall = \frac{TP}{P}$
Column totals:		<b>P</b>	<b>N</b>	$accuracy = \frac{TP+TN}{P+N}$	
				$F\text{-measure} = \frac{2}{1/precision + 1/recall}$	

---

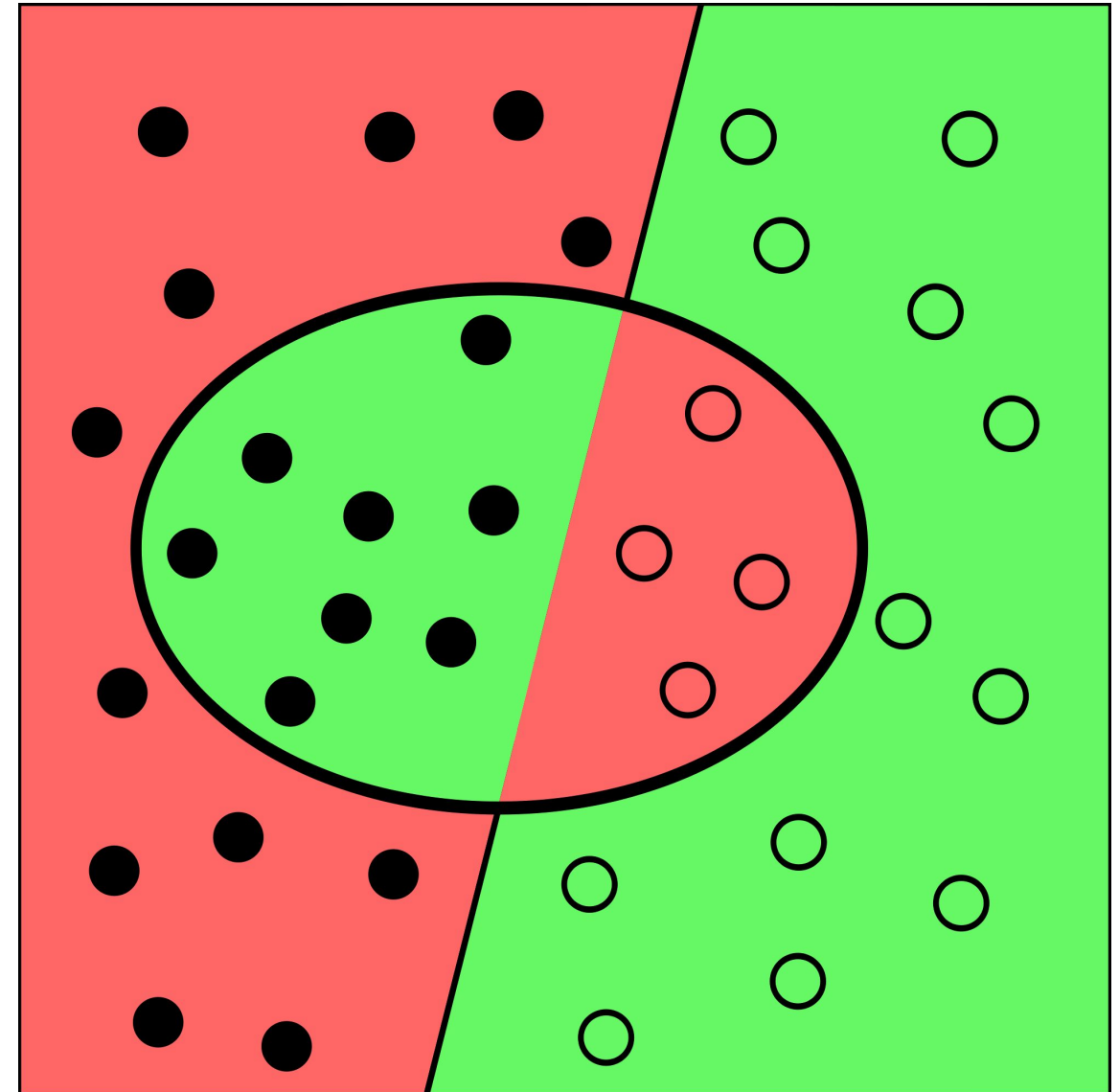
## THE MATH FOR PRECISION

---

- Let's use our marble example again
- If a model predicts 8 of the green marbles as green, then precision would be 1.00, because all marbles predicted as green were in fact green
- Let's assume all red marbles were predicted correctly, and 2 green were predicted as red
- The precision of red marbles would be  $10 / (10 + 2) = 0.833$

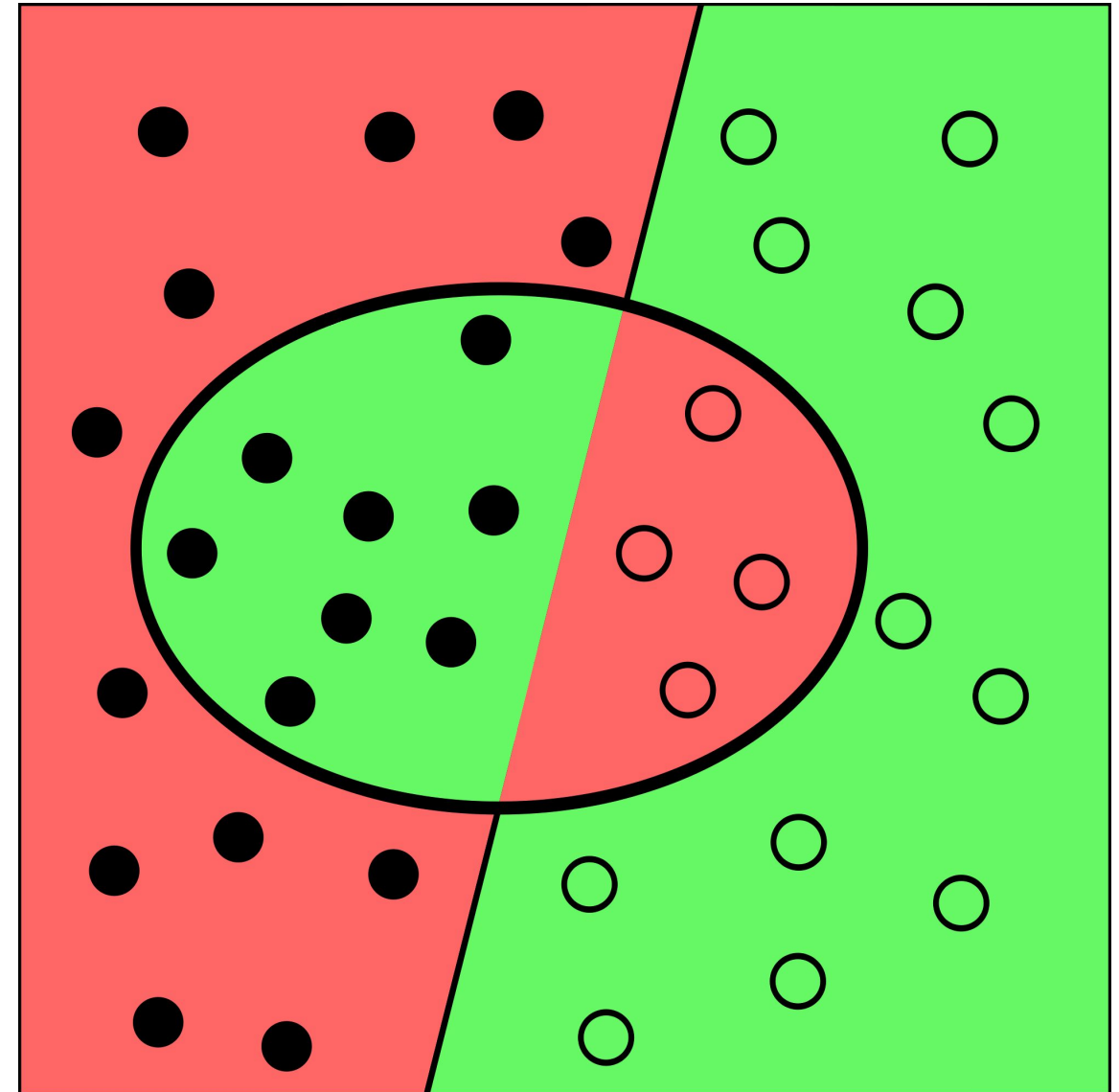
## ANOTHER EXAMPLE

- ▶ Imagine another marble problem
  - green = positive class (1)
  - red = 0
- ▶ Shaded circles = correct predictions (e.g. green was predicted as green)
- ▶ Unshaded circles = incorrect predictions (e.g. green was predicted red)



## ANOTHER EXAMPLE

- Background = color predicted
- E.g. a shaded circle on green = a green marble that was predicted as green
- E.g. an unshaded circle on red = a red marble that was predicted as green





---

## ANOTHER EXAMPLE

---

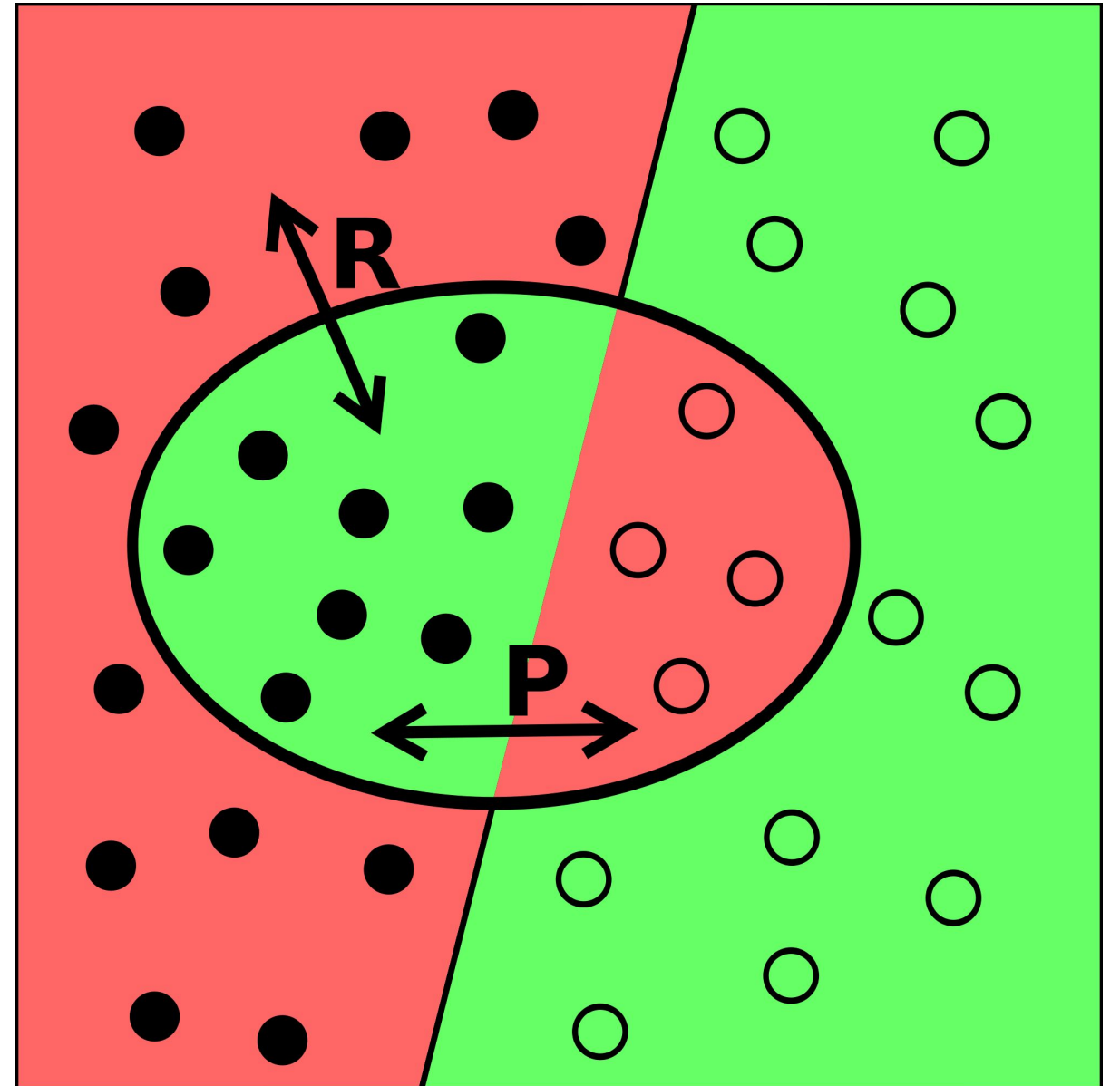
- For this example, we would have the following confusion matrix

		True Class	
		Green	Red
Predicted Class	Green	8	4
	Red	12	12

- We could calculate precision for green marbles as  $8 / (8 + 4) = 0.6666$
- We could calculate recall for green marbles as  $8 / (8 + 12) = 0.4000$

## ANOTHER EXAMPLE

- ▶ Notice we don't talk about the red marbles predicted as green
- ▶ We've chosen to focus on our model's accuracy as it relates to predicting green marbles



# ACTIVITY: KNOWLEDGE CHECK

## ANSWER THE FOLLOWING QUESTIONS

1. What would the precision and recall be for the following confusion matrix (with “green” being “true”)?

	is_green	is_not_green
<i>predicted_green</i>	13	8
<i>predicted_not_green</i>	7	12

## DELIVERABLE

Answers to the above question



---

# THE DIFFERENCE BETWEEN PRECISION AND RECALL

---

- The key difference between the two is the attribution and value of error
- Should our model be more pick in avoiding false positives (precision)?
- Or should it be more pick in avoiding false negatives (recall)?
- The answer should be determined by the problem you're trying to solve

**DEMO**

---

# UNDERSTANDING TRADEOFF

---

# UNDERSTANDING TRADEOFF

---

- Let's consider the following data problem:
  - we are given a data set in order to predict or identify traits for *typically late flights*
- Optimizing toward **recall**, we could assume that every flight will be delayed
- The trade-off, a lower precision, is that this could create even further delays, missed flights, etc.

---

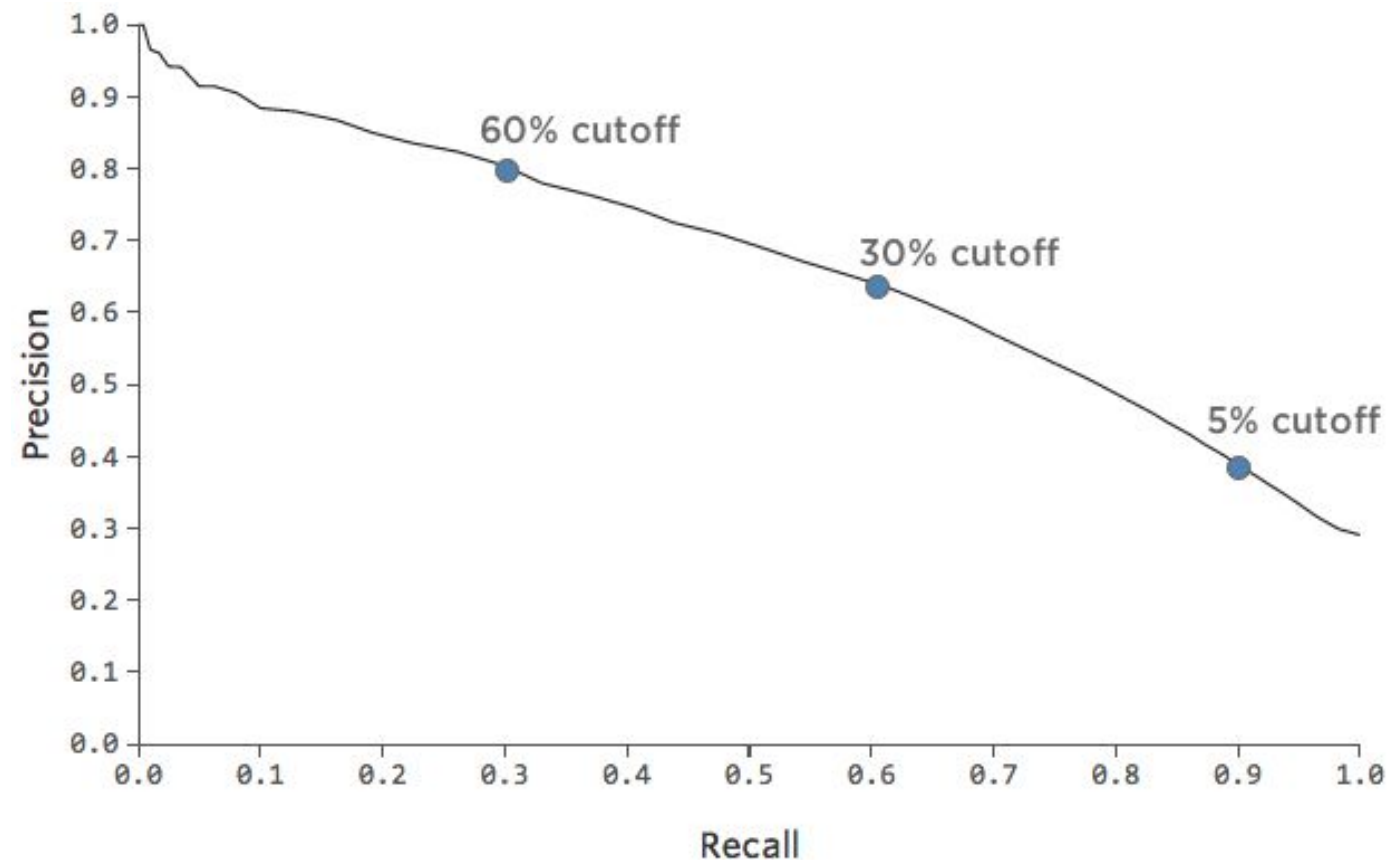
## UNDERSTANDING TRADEOFF

---

- Optimizing toward **precision**, we would specifically look to identify flights that will be late
- The trade-off here would be lower recall
  - We might miss flights that would be delayed, causing a strain on the system

# UNDERSTANDING TRADEOFF

- Below is a sample plot that shows how precision and recall are related for a model used to predict late flights





---

## UNDERSTANDING TRADEOFF

---

- This plot is based on choosing decision line **thresholds**,
  - much like the AUC figure from the previous class
- In terms of modeling delays,
  - this would be like moving the decision line for lateness from a probability of 0.01 up to 0.99,
  - and then calculating the precision and recall each time

---

# UNDERSTANDING TRADEOFF

---

- Takeaways:

- At a lower recall, there is typically greater precision in the model ...and vice-versa
- Establish that your model outperforms some benchmark
- Whether we're optimizing for recall or precision,
  - plotting helps us decide on our threshold

---

## **GUIDED PRACTICE**

---

# **COST BENEFIT ANALYSIS**

---

# ACTIVITY: COST BENEFIT ANALYSIS

---



## EXERCISE

### DIRECTIONS (15 minutes)

One tool that complements the confusion matrix is ***cost-benefit analysis***

- attaching a *value* to correctly and incorrectly predicted data

Like the Precision-Recall tradeoff, there's a balancing point to the *probabilities* of a given position in the confusion matrix,

- and the *cost* or *benefit* to that position

This approach allows you to not only add a *weighting system* to your confusion matrix, but also to speak the language of your business stakeholders

- i.e. communicate your values in *dollars*!

# ACTIVITY: COST BENEFIT ANALYSIS



## EXERCISE

### DIRECTIONS

Consider the following:

You've built a model that reduces user churn--the number of users who decide to stop paying for a product--through a marketing campaign.

Your model generates a confusion matrix with the following probabilities (these probabilities are calculated as the value in that position over the sum of the sample):

TP: 0.2	FP: 0.2
-----	
FN: 0.1	TN: 0.5

# ACTIVITY: COST BENEFIT ANALYSIS



## EXERCISE

### DIRECTIONS (15 minutes)

In this case:

- The *benefit* of a true positive is the retention of a user (\$10 for the month)
- The *cost* of a false positive is the spend of the campaign per user (\$0.05)
- The *cost* of a false negative (someone who could have retained if sent the campaign) is, effectively, 0 (we didn't send it... but we certainly didn't benefit!)
- The *benefit* of a true negative is 0: No spend on users who would have never retained

To calculate Cost-Benefit, we'll use this following function:

$$(P(TP) * B(TP)) + (P(TN) * B(TN)) + (P(FP) * C(FP)) + (C(FN) * C(FN))$$

which for our marketing problem, comes out to this:

$$(.2 * 10) + (.5 * 0) - (.2 * .05) - (.1 * 0)$$

or \$1.99 per user targeted

---

# ACTIVITY: COST BENEFIT ANALYSIS

---



## EXERCISE

### FOLLOW UP QUESTIONS

Think about **precision**, **recall**, and **cost benefit analysis** to answer the following questions:

1. How would you rephrase the business problem if your model was optimizing toward *precision*?
  - i.e. How might the model behave differently, and what effect would it have?
2. How would you rephrase the business problem if your model was optimizing toward *recall*?
3. What would the most ideal model look like in this case?

### DELIVERABLE

Answers to the above questions

---

## **INTRODUCTION**

---

# **SHOWING WORK**



---

## SHOWING WORK

---

- We've spent a lot of time exploring our data and building a reasonable model that performs well
- This can be lost on our audience if our visuals are:
  - Statistically heavy
    - Most people don't understand histograms
  - Overly complicated
    - Scatter matrices produce too much information (internal use only!)
  - Poorly labeled
    - Labels aren't required, so you may not have added them

---

# SHOWING WORK

---

- In order to convey important information to our audience, make sure our charts are:
  - Simplified
  - Easily interpretable
  - Clearly labeled

---

# SIMPLIFIED

---

- At most, include figures that either:
  - explain a variable on its own
  - or explain that variable's relationship with a target
- If your model used a data transformation (like natural log),
  - try to visualize the original data (where practical)
- Try to remove any unnecessary complexity
  - Think hard about what to keep, and how to simplify

---

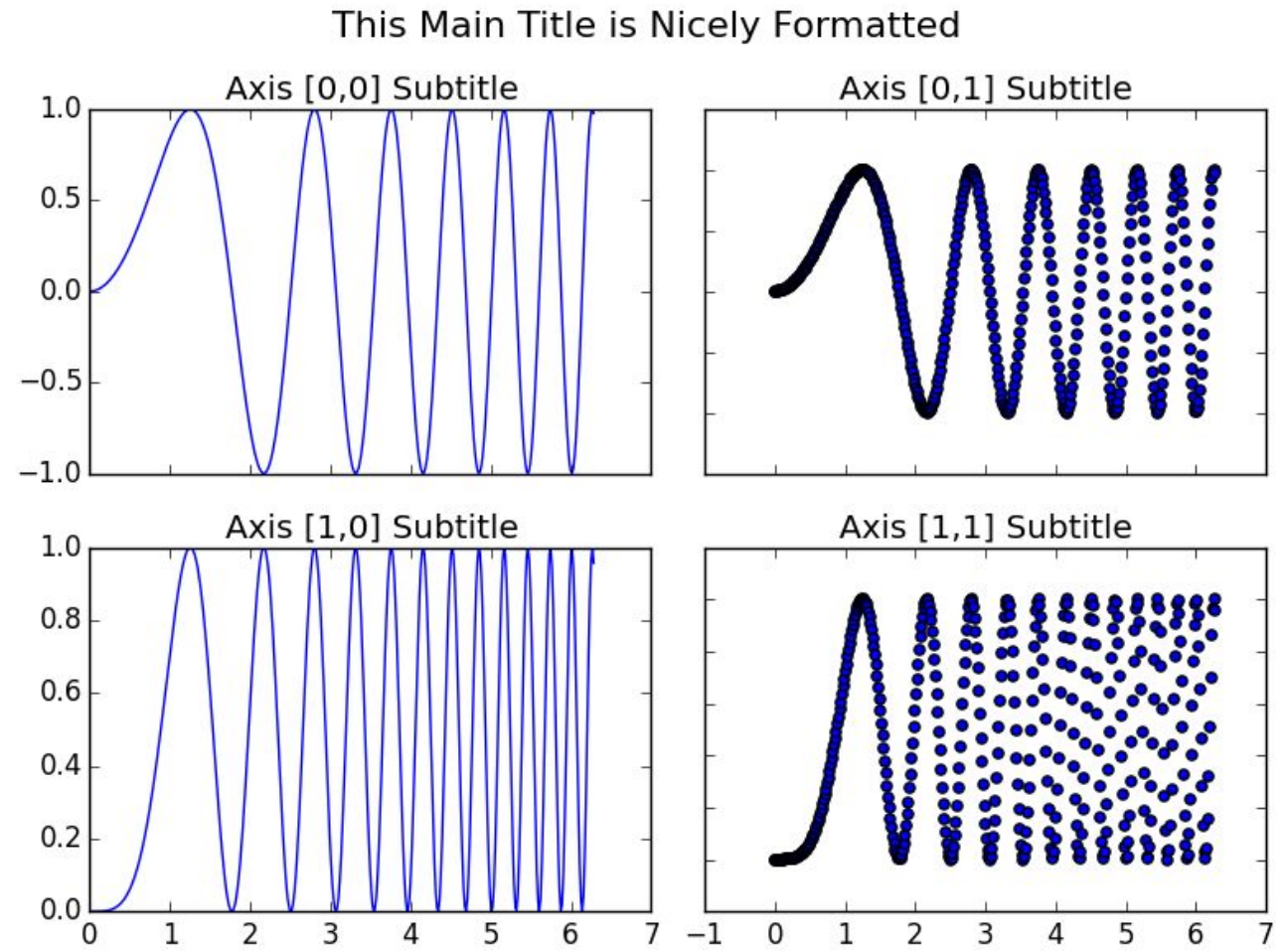
## **EASILY INTERPRETABLE**

---

- Any stakeholder looking at a figure should be seeing the exact same thing you're seeing
- A good test for this is to share the visual with others less familiar with the data and see if they come to the same conclusion
- How long did it take them?

# CLEARLY LABELED

- Take the time to:
  - clearly label your axis
  - title your plot
  - double check your scales
    - especially if the figures should be comparable
- If you're showing two graphs side by side
  - they should follow the same Y axis (where feasible)



---

## QUESTION TO ASK

---

- When building visuals for another audience, ask yourself these questions:
  - **Who:** Who is my target audience for the visual?
  - **What:** What do they already know about this project? What do they need to know?
  - **How:** How does my project affect this audience? How might they interpret (or misinterpret) the data?

**DEMO**

---

# VISUALIZING MODELS OVER VARIABLES

---

## **VISUALIZING MODELS OVER VARIABLES**

---

- One effective way to explain your model over particular variables is to plot the predicted values against the most explanatory variables
- For example, in logistic regression, plotting the probability of a class against a variable can help explain the range of effect of the model



---

# **VISUALIZING MODELS OVER VARIABLES**

---

- Let's build our first model and plot
  - We'll use the flight delay data for all following examples
- Found in the starter code from the class repo

---

# VISUALIZING MODELS OVER VARIABLES

---

```
# read in the file and generate a quick model (assume we've done the data exploration already)
```

```
import pandas as pd
import sklearn.linear_model as lm
import matplotlib.pyplot as plt
```

```
df = pd.read_csv('./datasets/flight_delays.csv')
```

```
df = df.join(pd.get_dummies(df['DAY_OF_WEEK'], prefix='dow'))
```

```
df = df[df.DEP_DEL15.notnull()].copy()
```

---

# VISUALIZING MODELS OVER VARIABLES

---

```
# Build a model
model = lm.LogisticRegression()
features = ['dow_1', 'dow_2', 'dow_3', 'dow_4', 'dow_5', 'dow_6']
model.fit(df[features + ['CRS_DEP_TIME']], df['DEP_DEL15'])

df['probability'] = model.predict_proba(df[features + ['CRS_DEP_TIME']]).T[1]
```

---

# VISUALIZING MODELS OVER VARIABLES

---

```
# Create a plot
ax = plt.subplot(111)
colors = ['blue', 'green', 'red', 'purple', 'orange', 'brown']
for e, c in enumerate(colors):
    df[df[features[e]] == 1].plot(x='CRS_DEP_TIME', y='probability',
kind='scatter', color=c, ax=ax)

ax.set(title='Probability of Delay\n Based on Day of Week and Time of Day')
```

# VISUALIZING MODELS OVER VARIABLES

- This visual highlights the range of effects on delays from both day of the week and time of day
- Some days appear more likely to have delays than others
- Delay likelihood increases as the day goes on



---

# ACTIVITY: TRY IT OUT

---



## EXERCISE

### DIRECTIONS

1. Adjust the model to make delay predictions using **airlines instead of day of week**, and time, then plot the effect on CRS\_DEP\_TIME=1
2. Try plotting the inverse:
  - pick either model and plot the effect on CRS\_DEP\_TIME=0

### DELIVERABLE

The new plots

**DEMO**

---

# **VISUALIZING PERFORMANCE AGAINST BASELINE**

---

## **VISUALIZING PERFORMANCE AGAINST BASELINE**

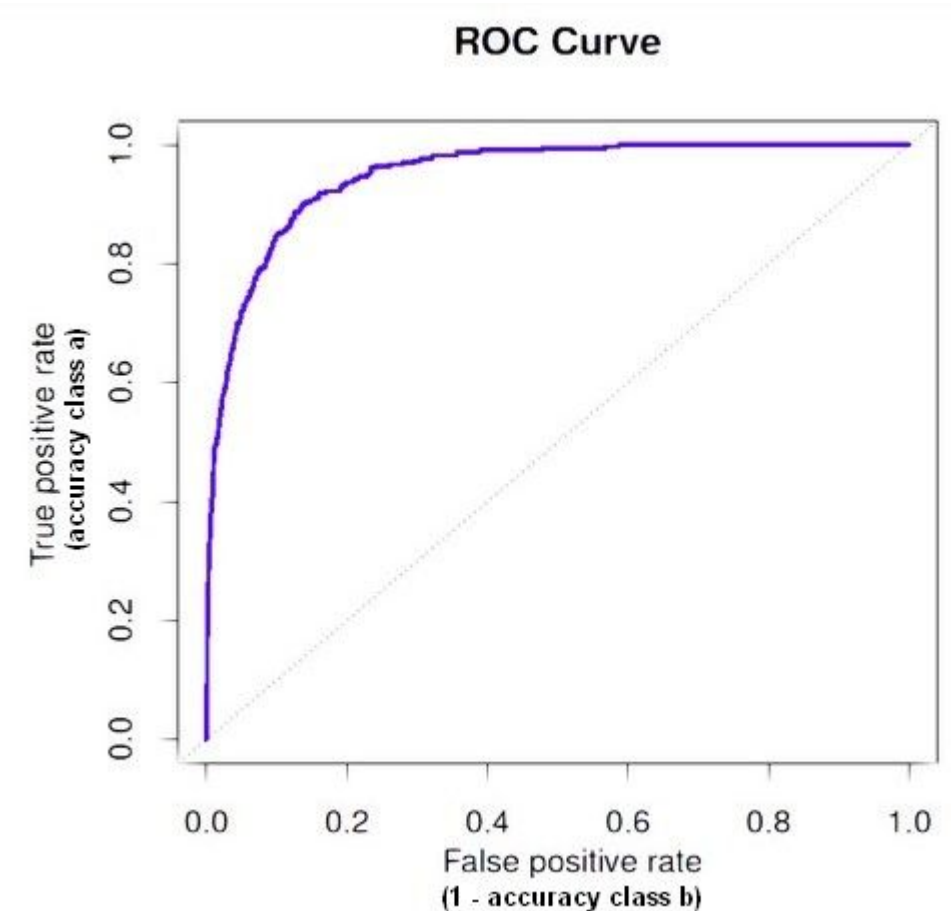
---

- Another approach of visualization is the effect of your model against a baseline
  - or - even better - against previous models
- Plots like this will also be useful when talking to your peers
  - other data scientists or analysts who are familiar with your project and interested in the progress you've made



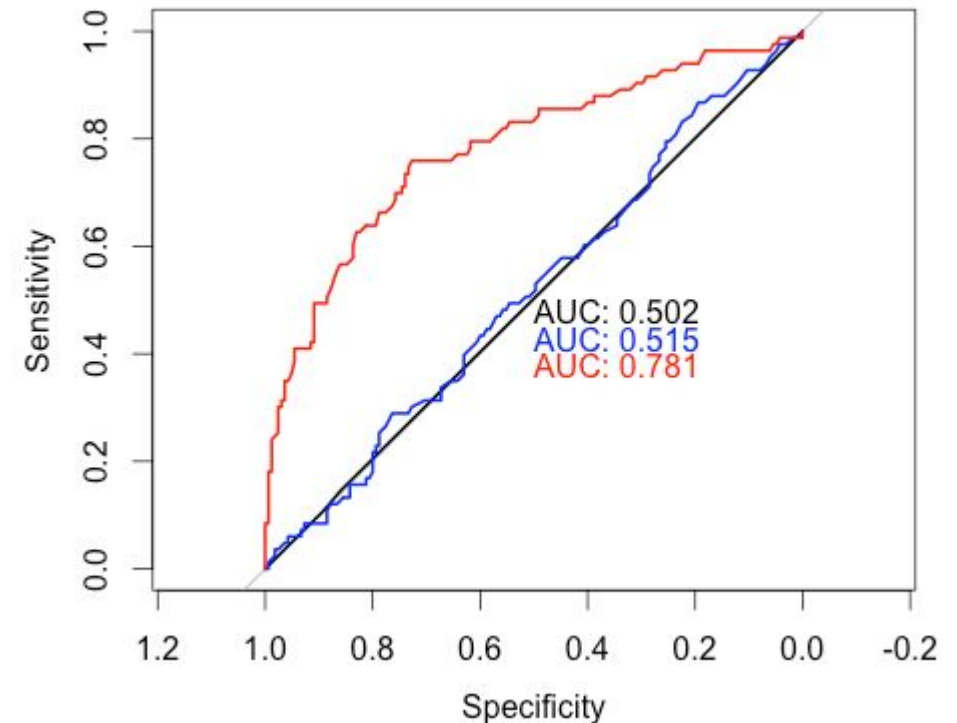
# VISUALIZING PERFORMANCE AGAINST BASELINE

- For classification, we've practiced plotting AUC and precision-recall plots
- Consider the premise of each:
  - AUC plots represent “accuracy” as having the largest area under the curve
    - Good models = high and to the left
  - For precision-recall plots, it will depend on the *cost* requirements
    - Either a model will have good recall at the cost of precision or vice versa



# VISUALIZING PERFORMANCE AGAINST BASELINE

- ▶ When comparing multiple models:
  - ▶ For AUC plots, you'll be interested in which model has the *largest* area under the curve
- For precision-recall plots, based on the cost requirement, you are looking at which model has:
  - the best precision given the same recall
  - or the best recall given the same precision



---

# VISUALIZING PERFORMANCE AGAINST BASELINE

---

- Example found in the starter code...
- We've plotted 2 models for AUC:
  - a dummy model and a logistic regression model

```
model0 = dummy.DummyClassifier()  
model0.fit(df[features[1:]], df.DEP_DEL15)  
df['probability_0'] = model0.predict_proba(df[features[1:]]).T[1]
```

```
model = lm.LogisticRegression()  
model.fit(df[features[1:]], df.DEP_DEL15)  
df['probability_1'] = model.predict_proba(df[features[1:]]).T[1]
```

---

# VISUALIZING PERFORMANCE AGAINST BASELINE

---

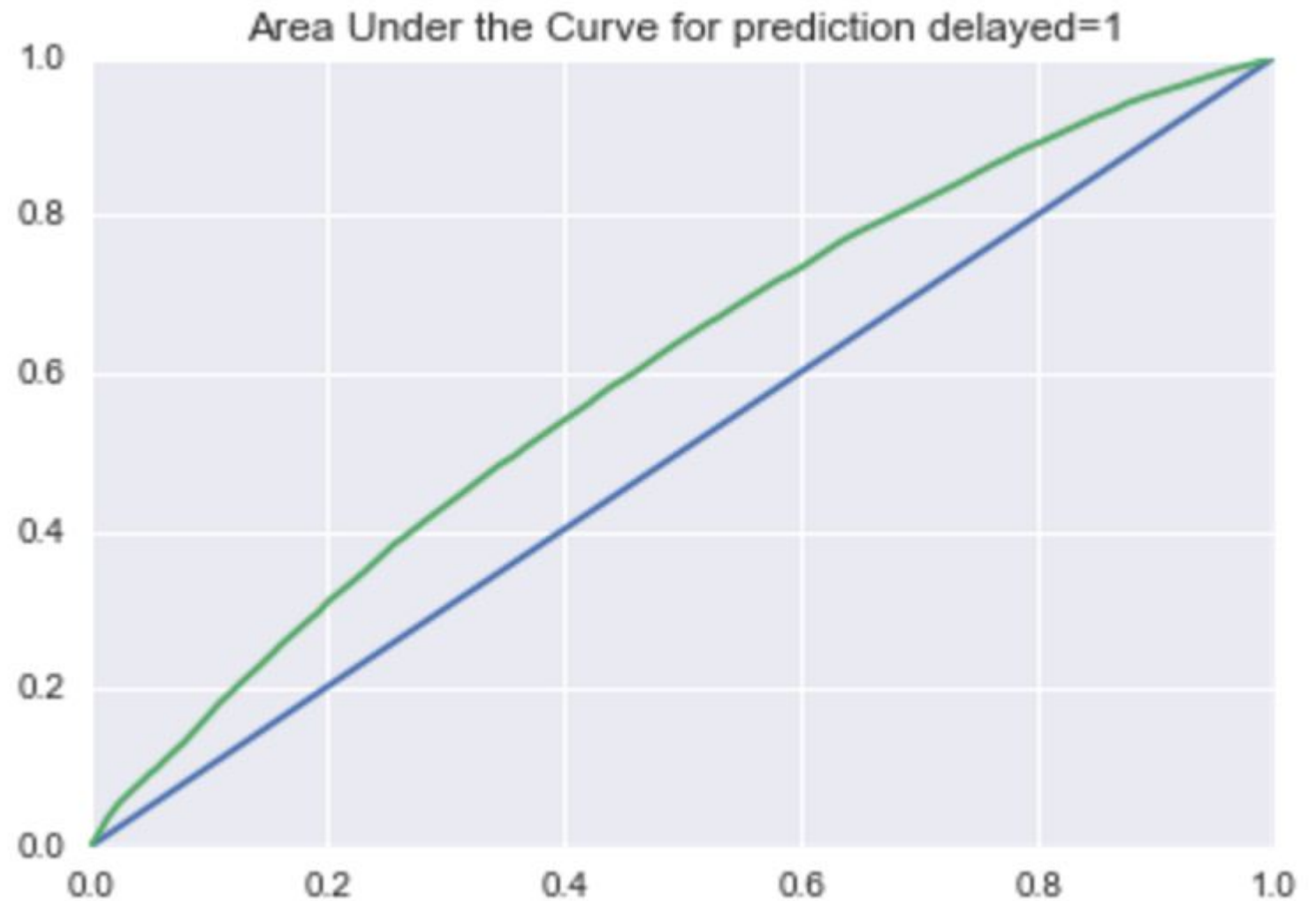
```
ax = plt.subplot(111)
vals = metrics.roc_curve(df.DEP_DEL15, df.probability_0)
ax.plot(vals[0], vals[1])
vals = metrics.roc_curve(df.DEP_DEL15, df.probability_1)
ax.plot(vals[0], vals[1])
```

```
ax.set(title='Area Under the Curve',
xlabel='FRP', ylabel='TRP', xlim=(0, 1), ylim=(0, 1))
```

# VISUALIZING PERFORMANCE AGAINST BASELINE

▸ This plot showcases:

1. The model using data outperforms a baseline dummy model
2. By adding other features, there's some give and take with probability as the model gets more complicated



---

# ACTIVITY: TRY IT OUT

---



## EXERCISE

### DIRECTIONS

1. In a similar approach, use the sklearn `precision_recall_curve` function to enable you to plot the precision-recall curve of the two models from above
  - Keep in mind precision is on the y-axis
2. Explain what is occurring when the recall is below 0.2
3. Based on this performance, is there a clear winner at different thresholds?

### DELIVERABLE

The new plots and associated answers

---

**INDEPENDENT PRACTICE**

---

**PROJECT PRACTICE**

---

# ACTIVITY: PROJECT PRACTICE

---



## EXERCISE

### DIRECTIONS (45 minutes)

Using models built from the flight data problem earlier in class, work through the same problems (your data and models should already be accessible)

#### Your goals:

1. Consider what is a proper "categorical" variable, and keep *only* what is significant
  - You'll have 20+ variables
  - Aim to have at least **3 visuals** that clearly **explain the relationship** of variables you've used against the predictive survival value
2. Generate the AUC or precision-recall curve,
  - and have a statement that defines, compared to a baseline, how your model performs and any caveats

For example:

"My model on average performs at x rate, but the features under-perform and explain less of the data at these thresholds."

### DELIVERABLE

New models and performance statement



---

**CONCLUSION**

---

# TOPIC REVIEW

---

## REVIEW AND NEXT STEPS

---

- What do precision and recall mean? How are they similar and different to True Positive Rate and False Positive Rate?
- How does cost benefit analysis play a role in building models?
- What are at least two very important details to consider when creating visuals for a project's stakeholders?
- Why would an AUC plot work well for a data science audience but not for a business audience? What would be a more effective visualization for that group?

**COURSE**

---

**BEFORE NEXT CLASS**

---

## **BEFORE NEXT CLASS**

---

# **UPCOMING**

- Final Project Proposal & Unit 4 Project Due: Thurs (4/26)

---

**LESSON**

---

**Q & A**

## **LESSON**

---

# **EXIT TICKET**

**DON'T FORGET TO FILL OUT YOUR EXIT TICKET**