Launch a data science career!

Name:

Email address:

Join the Newsletter

About

New? Start here!

Data School Courses

Join my 60,000+ YouTube subscribers

Privacy Policy

November 19, 2014 · MACHINE LEARNING TUTORIAL

ROC curves and Area Under the Curve explained (video)

While competing in a **Kaggle** competition this summer, I came across a simple **visualization** (created by a fellow competitor) that helped me to gain a better intuitive understanding of ROC curves and Area Under the Curve (AUC). I created a video explaining this visualization to serve as a learning aid for my data science students, and decided to share it publicly to help others understand this complex topic.

An ROC curve is the most commonly used way to **visualize the performance of a binary classifier**, and AUC is (arguably) the best way to **summarize its performance in a single number**. As such, gaining a deep understanding of ROC curves and AUC is beneficial for data scientists, machine learning practitioners, and medical researchers (among others).

The **14-minute video** is embedded below, followed by the complete transcript (including graphics). **If you want to skip to a particular section in the video**, simply click one of the time codes listed in the transcript (such as **0:52**).

I welcome your feedback and questions in the comments section!

P.S. Want more content like this in your inbox? **Subscribe to the Data School newsletter**.

Kevin Markham

@justmarkham

Think you understand ROC curves & AUC? Are you sure? In-depth video: youtu.be/OAl6eAyP-yo Transcript & screenshots: dataschool.io/roc-curves-and…

7:59 PM - Nov 20, 2014

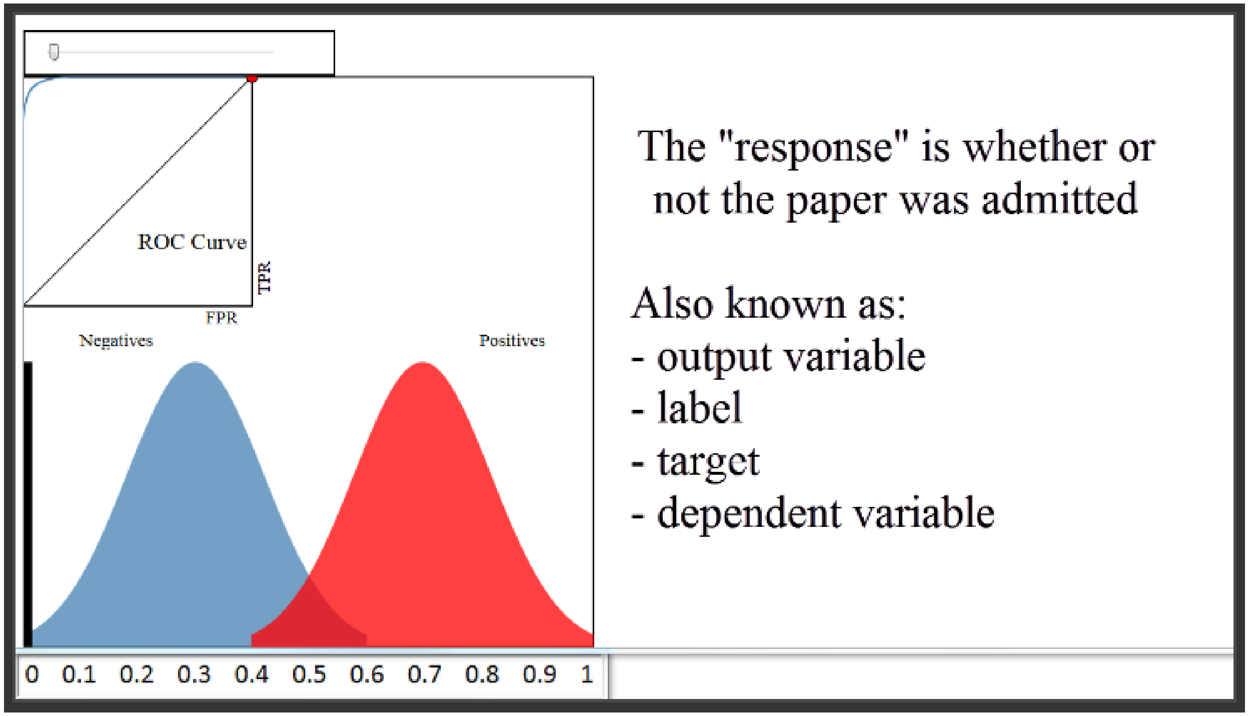
34 43 people are talking about this

Video Transcript

(**0:00**) This video should help you to gain an intuitive understanding of ROC curves and Area Under the Curve, also known as AUC.

An ROC curve is a commonly used way to **visualize the performance of a binary classifier**, meaning a classifier with two possible output classes.

For example, let's pretend you built a classifier to predict whether a research paper will be admitted to a journal, based on a variety of factors. The features might be the length of the paper, the number of authors, the number of papers those authors have previously submitted to the journal, et cetera. The response (or "output variable") would be whether or not the paper was admitted.

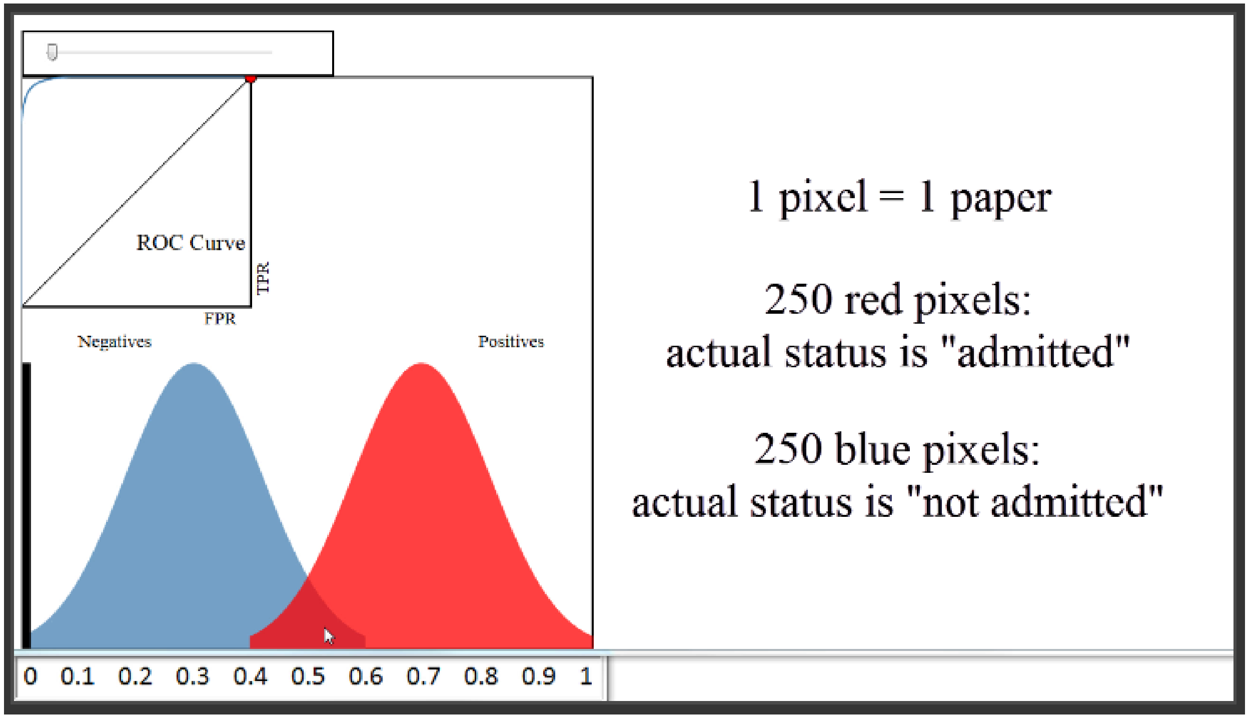


The "response" is whether or not the paper was admitted

Also known as:
- output variable
- label
- target
- dependent variable

(**0:52**) Let's first take a look at the bottom portion of this diagram, and ignore the everything except the blue and red distributions. We'll pretend that **every blue and red pixel represents a paper** for which you want to predict the admission status.

This is your validation (or "hold-out") set, so you know the true admission status of each paper. The 250 red pixels are the papers that were actually admitted, and the 250 blue pixels are the papers that were not admitted.

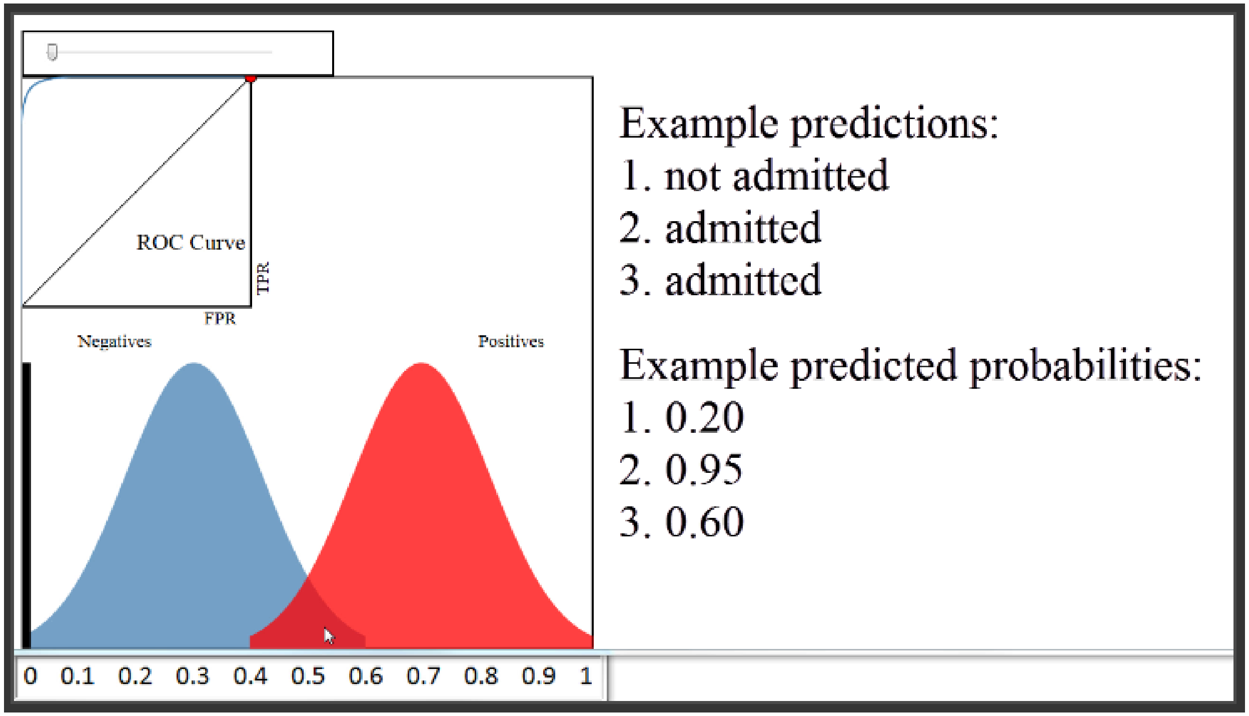


1 pixel = 1 paper

250 red pixels:
actual status is "admitted"

250 blue pixels:
actual status is "not admitted"

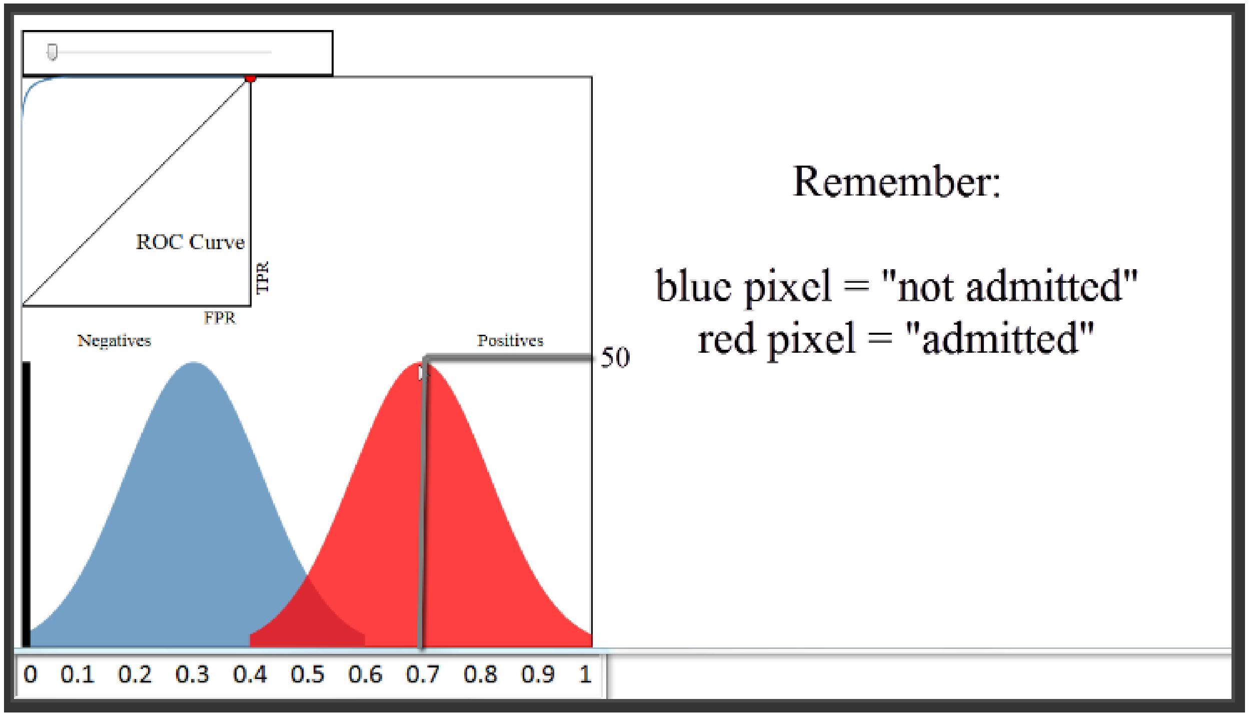
(**1:32**) Since this is your validation set, you want to judge how well your model is doing by comparing your model's predictions to the true admission statuses of those 500 papers. We'll assume that you used a classification method such as logistic regression that can not only make a **prediction** for each paper, but can also output a **predicted probability** of admission for each paper. These blue and red distributions are one way to visualize how those predicted probabilities compare to the true statuses.



Example predictions:
1. not admitted
2. admitted
3. admitted

Example predicted probabilities:
1. 0.20
2. 0.95
3. 0.60

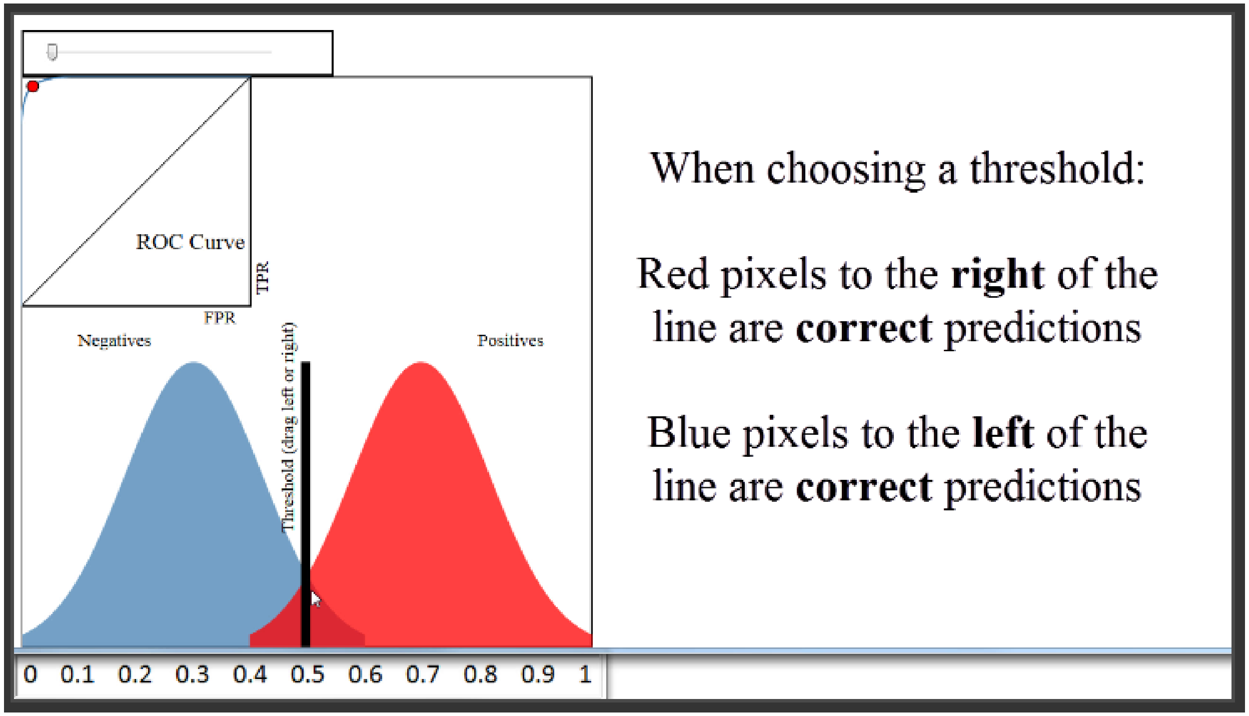
(**2:08**) Let's examine this plot in detail. The x-axis represents your **predicted probabilities**, and the y-axis represents a **count of observations**, kind of like a histogram. Let's estimate that the height at 0.1 is 10 pixels. This plot tells you that there were 10 papers for which you predicted an admission probability of 0.1, and the true status for all 10 papers was negative (meaning not admitted). There were about 50 papers for which you predicted an admittance probability of 0.3, and none of those 50 were admitted. There were about 20 papers for which you predicted a probability of 0.5, and half of those were admitted and the other half were not. There were 50 papers for which you predicted a probability of 0.7, and all of those were admitted. And so on.



Remember:

blue pixel = "not admitted"
red pixel = "admitted"

(**3:16**) Based on this plot, you might say that your classifier is doing quite well, since it did a good job of **separating the classes**. To actually make your class predictions, you might set your **"threshold"** at 0.5, and classify everything above 0.5 as admitted and everything below 0.5 as not admitted, which is what most classification methods will do by default. With that threshold, your **accuracy rate** would be above 90%, which is probably very good.

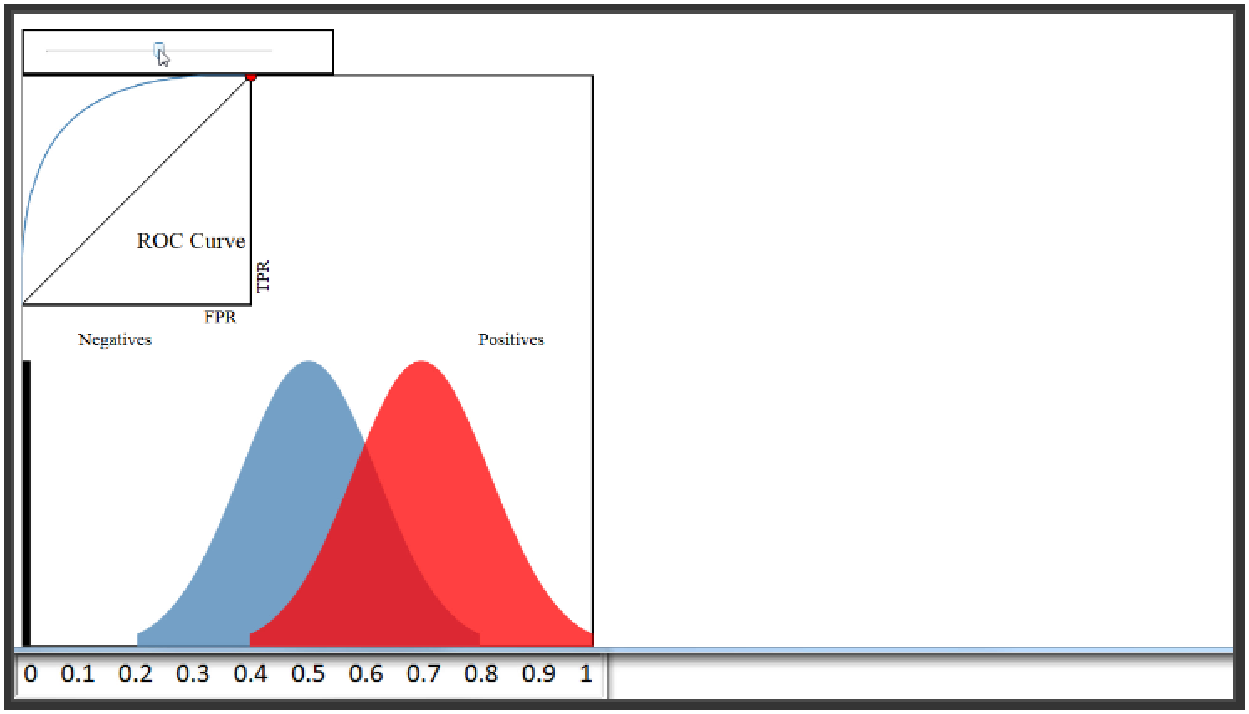


When choosing a threshold:

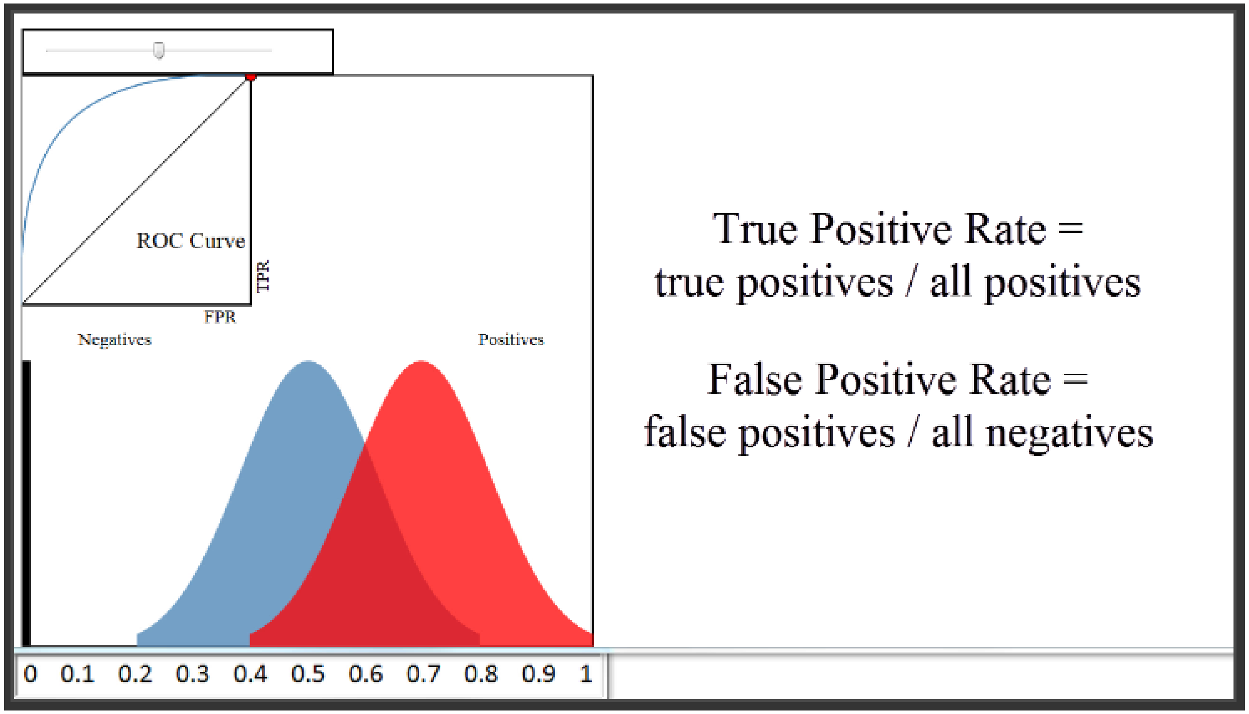
Red pixels to the **right** of the line are **correct** predictions

Blue pixels to the **left** of the line are **correct** predictions

(**3:58**) Now let's pretend that your classifier didn't do nearly as well and move the blue distribution. You can see that there is a lot more overlap here, and regardless of where you set your threshold, your classification accuracy will be much **lower** than before.

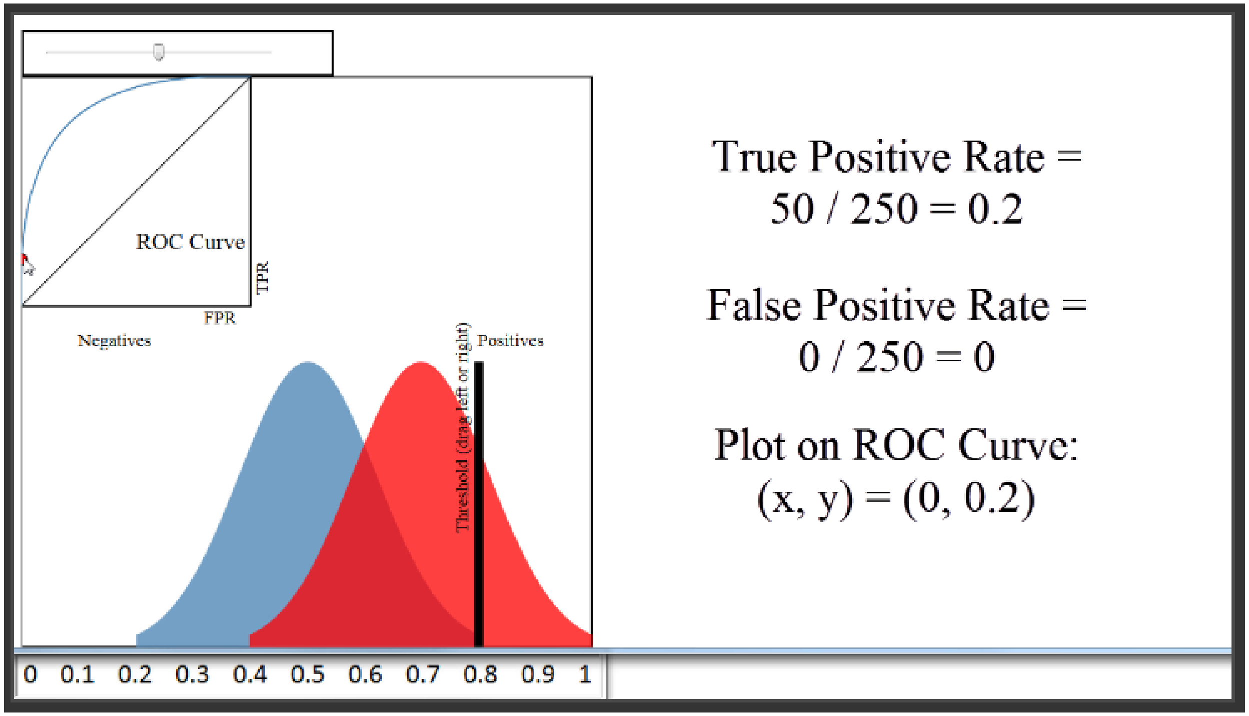


(**4:19**) Now let's talk about the ROC curve that you see here in the upper left. So, what is an ROC curve? It is a plot of the **True Positive Rate (on the y-axis)** versus the **False Positive Rate (on the x-axis)** for every possible classification threshold. As a **reminder**, the True Positive Rate answers the question, "When the actual classification is positive (meaning admitted), how often does the classfier predict positive?" The False Positive Rate answers the question, "When the actual classification is negative (meaning not admitted), how often does the classifier incorrectly predict positive?" Both the True Positive Rate and the False Positive Rate **range from 0 to 1**.



(**5:15**) To see how the ROC curve is actually generated, let's set some example thresholds for classifying a paper as admitted.

A threshold of 0.8 would classify 50 papers as admitted, and 450 papers as not admitted. The True Positive Rate would be the **red pixels to the right of the line divided by all red pixels**, or 50 divided by 250, which is 0.2. The False Positive Rate would be the **blue pixels to the right of the line divided by all blue pixels**, or 0 divided by 250, which is 0. Thus, we would plot a point at 0 on the x-axis, and 0.2 on the y-axis, which is right here.

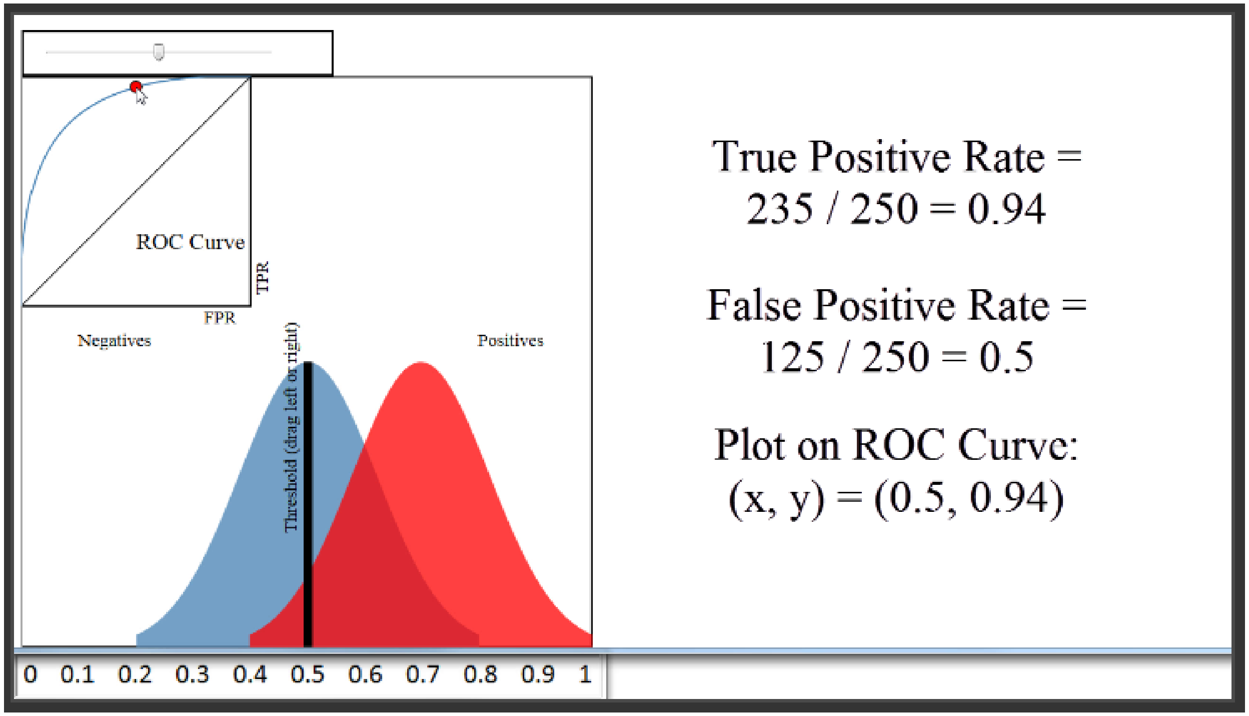


True Positive Rate =
50 / 250 = 0.2

False Positive Rate =
0 / 250 = 0

Plot on ROC Curve:
(x, y) = (0, 0.2)

(**6:16**) Let's set a different threshold of 0.5. That would classify 360 papers as admitted, and 140 papers as not admitted. The True Positive Rate would be 235 divided by 250, or 0.94. The False Positive Rate would be 125 divided by 250, or 0.5. Thus, we would plot a point at 0.5 on the x-axis, and 0.94 on the y-axis, which is right here.



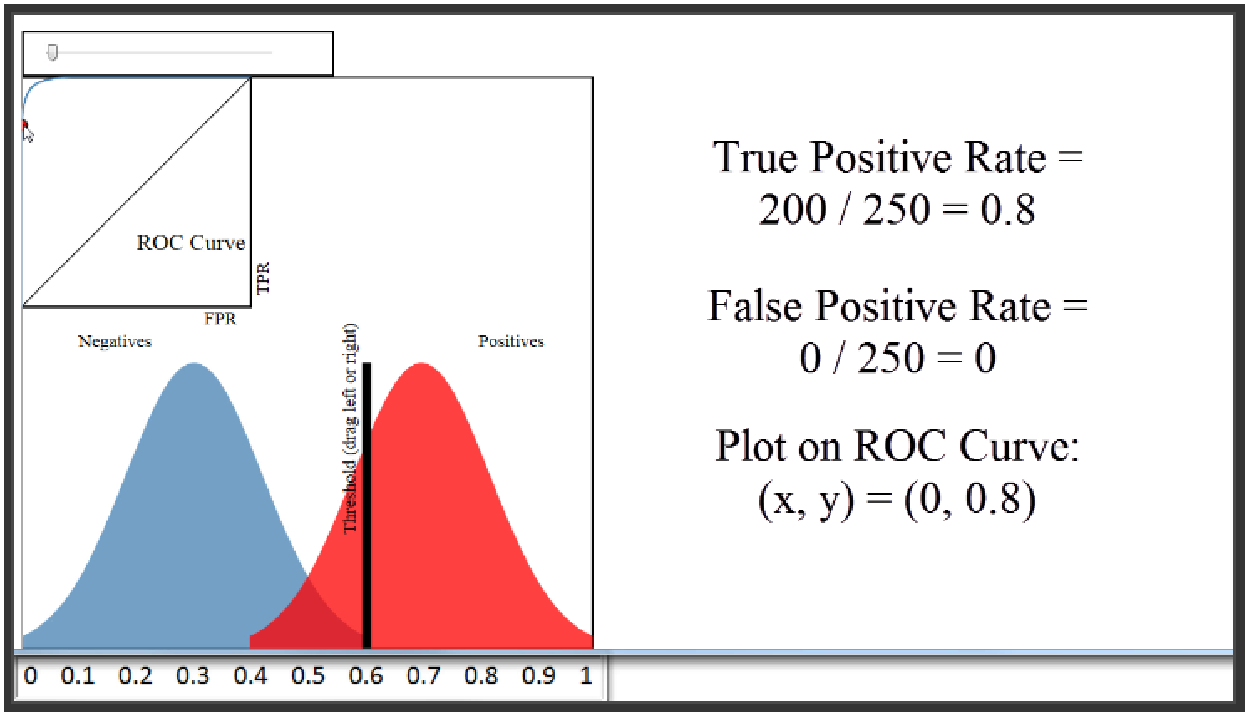
True Positive Rate = 235 / 250 = 0.94

False Positive Rate = 125 / 250 = 0.5

Plot on ROC Curve: (x, y) = (0.5, 0.94)

(**7:05**) We've plotted two points, but to generate the entire ROC curve, all we have to do is to plot the True Positive Rate versus the False Positive Rate for all possible classification thresholds which range from 0 to 1. That is a huge benefit of using an ROC curve to evaluate a classifier instead of a simpler metric such as misclassification rate, in that **an ROC curve visualizes all possible classification thresholds, whereas misclassification rate only represents your error rate for a single threshold**. Note that you can't actually see the thresholds used to generate the ROC curve anywhere on the curve itself.

Now, let's move the blue distribution back to where it was before. Because the classifier is doing a very good job of separating the blues and the reds, I can set a threshold of 0.6, have a True Positive Rate of 0.8, and still have a False Positive Rate of 0.

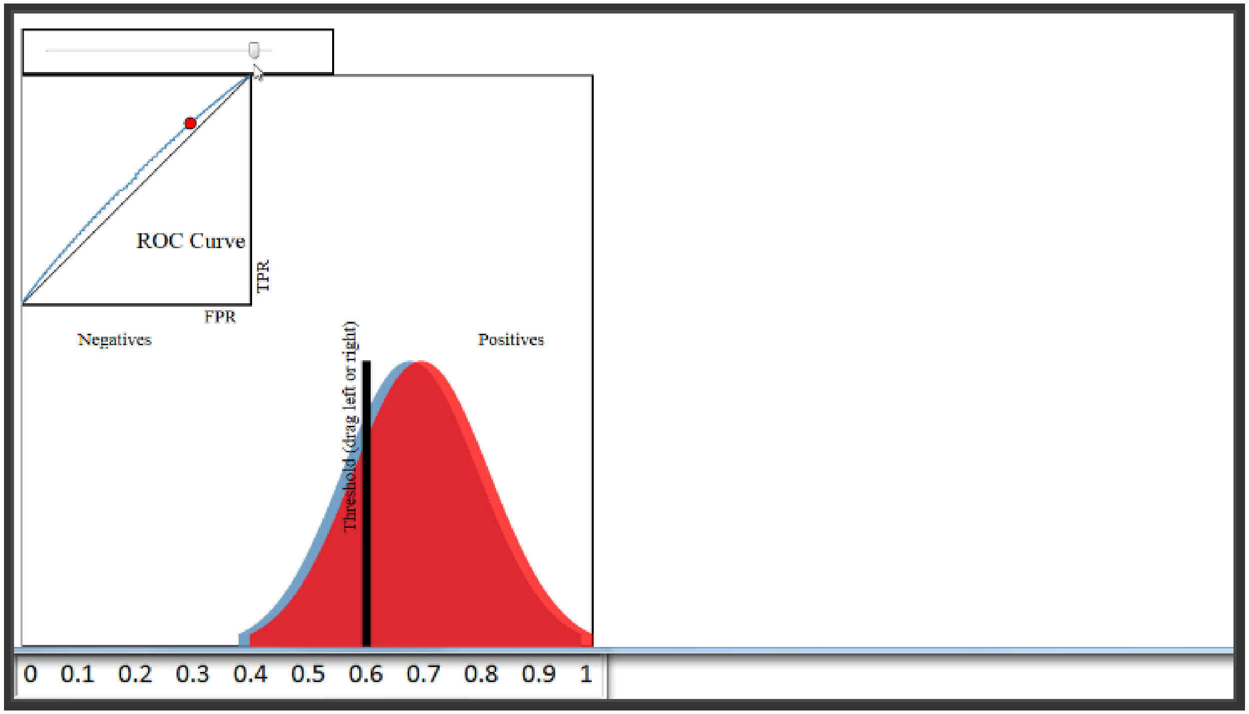


True Positive Rate =
200 / 250 = 0.8

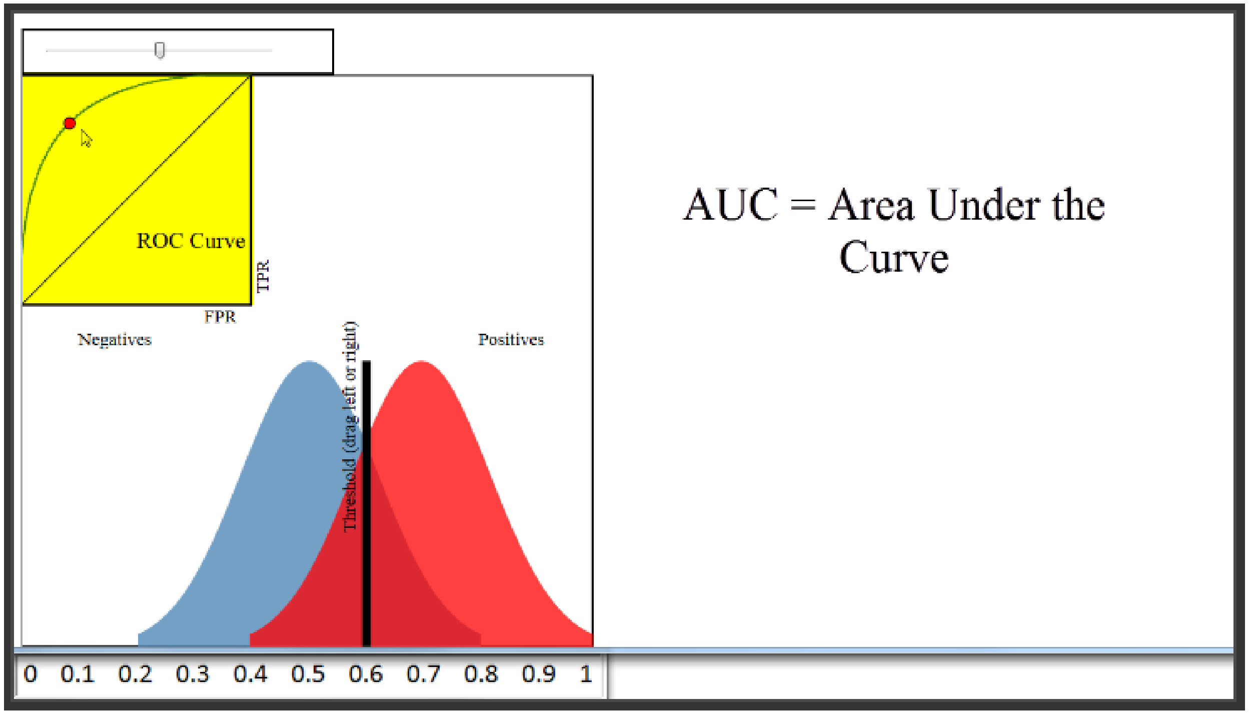
False Positive Rate =
0 / 250 = 0

Plot on ROC Curve:
(x, y) = (0, 0.8)

(**8:24**) Therefore, a classifier that does a very **good job separating the classes** will have an ROC curve that hugs the upper left corner of the plot. Conversely, a classifier that does a very **poor job separating the classes** will have an ROC curve that is close to this black diagonal line. That line essentially represents a classifier that does no better than random guessing.

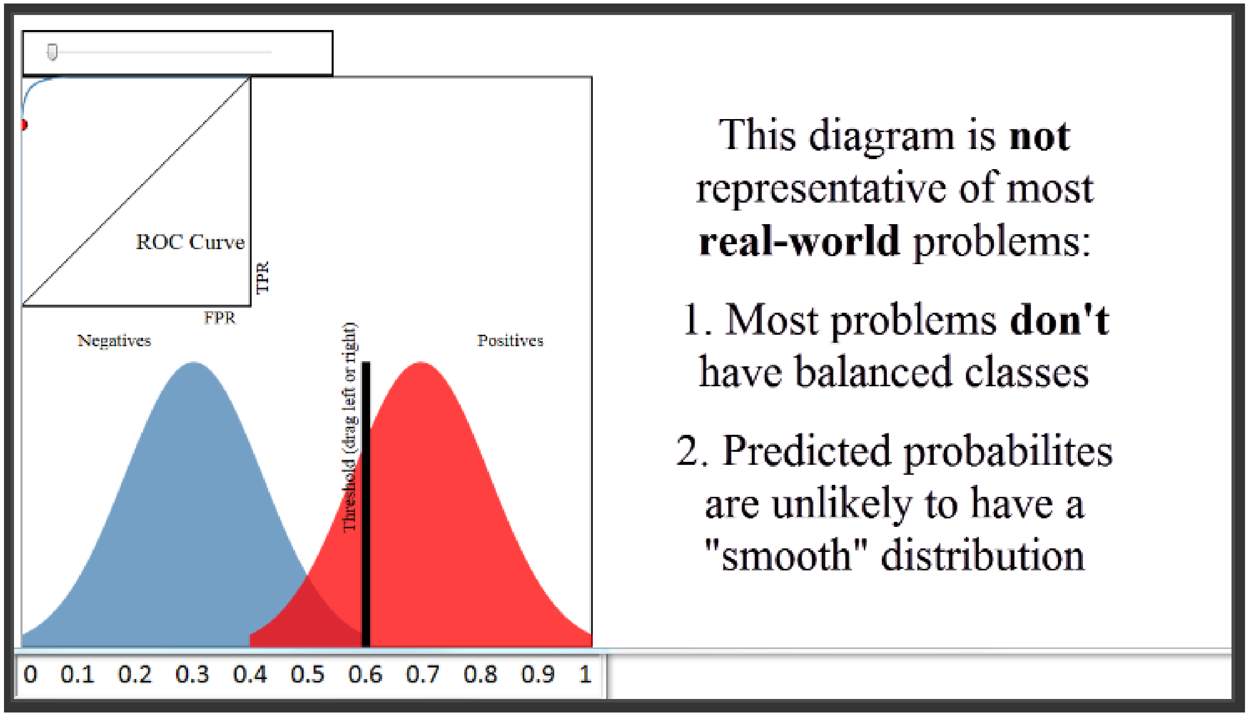


(**8:55**) Naturally, you might want to use the ROC curve to **quantify the performance of a classifier**, and give a higher score for this classifier than this classifier. That is the purpose of AUC, which stands for **Area Under the Curve**. AUC is literally just the percentage of this box that is under this curve. This classifier has an AUC of around 0.8, a very poor classifier has an AUC of around 0.5, and this classifier has an AUC of close to 1.



(**9:45**) There are two things I want to mention about this diagram. First, this diagram shows a case where your **classes are perfectly balanced**, which is why the size of the blue and the red distributions are identical. In most real-world problems, this is not the case. For example, if only 10% of papers were admitted, the blue distribution would be nine times larger than the red distribution. However, that doesn't change how the ROC curve is generated.

A second note about this diagram is that it shows a case where your **predicted probabilities have a very smooth shape**, similar to a normal distribution. That was just for demonstration purposes. The probabilities output by your classifier will not necessarily follow any particular shape.

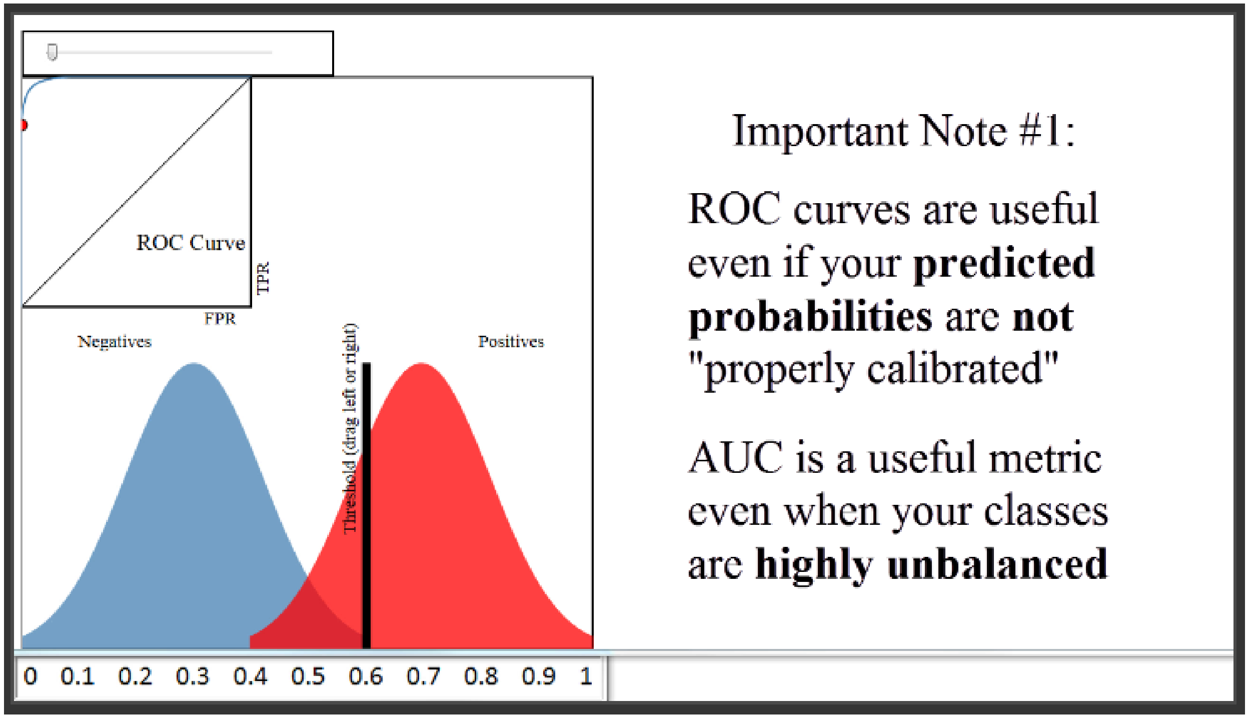


This diagram is **not** representative of most **real-world** problems:

1. Most problems **don't** have balanced classes

2. Predicted probabilites are unlikely to have a "smooth" distribution

(**10:40**) To close, I want to add three other important notes. The first note is that the ROC curve and AUC are **insensitive to whether your predicted probabilities are properly calibrated** to actually represent probabilities of class membership. In other words, the ROC curve and the AUC would be identical even if your predicted probabilities ranged from 0.9 to 1 instead of 0 to 1, as long as the ordering of observations by predicted probability remained the same. All the AUC metric cares about is how well your classifier separated the two classes, and thus **it is said to only be sensitive to rank ordering**. You can think of AUC as representing the probability that a classifier will rank a randomly chosen positive observation higher than a randomly chosen negative observation, and thus it is a **useful metric even for datasets with highly unbalanced classes**.

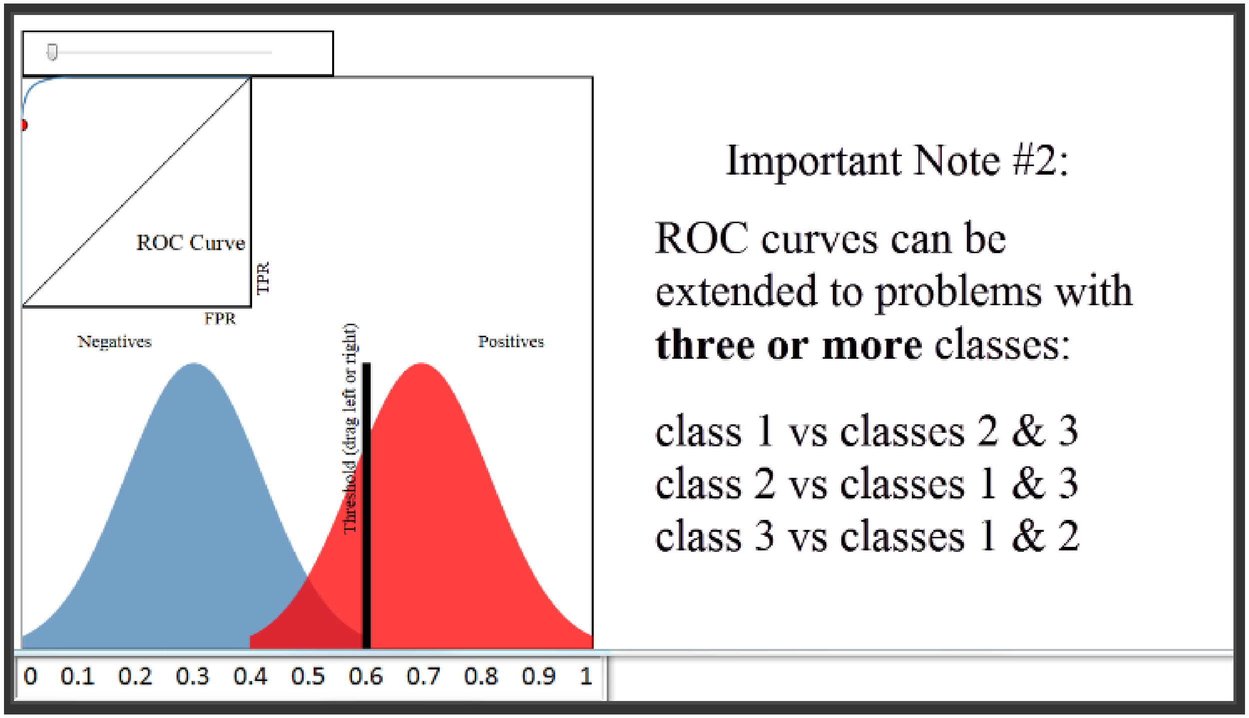


Important Note #1:

ROC curves are useful even if your **predicted probabilities** are **not** "properly calibrated"

AUC is a useful metric even when your classes are **highly unbalanced**

(**11:52**) The second note is that ROC curves can be extended to **classification problems with three or more classes** using what is called a "one versus all" approach. That means if you have three classes, you would create three ROC curves. In the first curve, you would choose the first class as the positive class, and group the other two classes together as the negative class. In the second curve, you would choose the second class as the positive class, and group the other two classes together as the negative class. And so on.

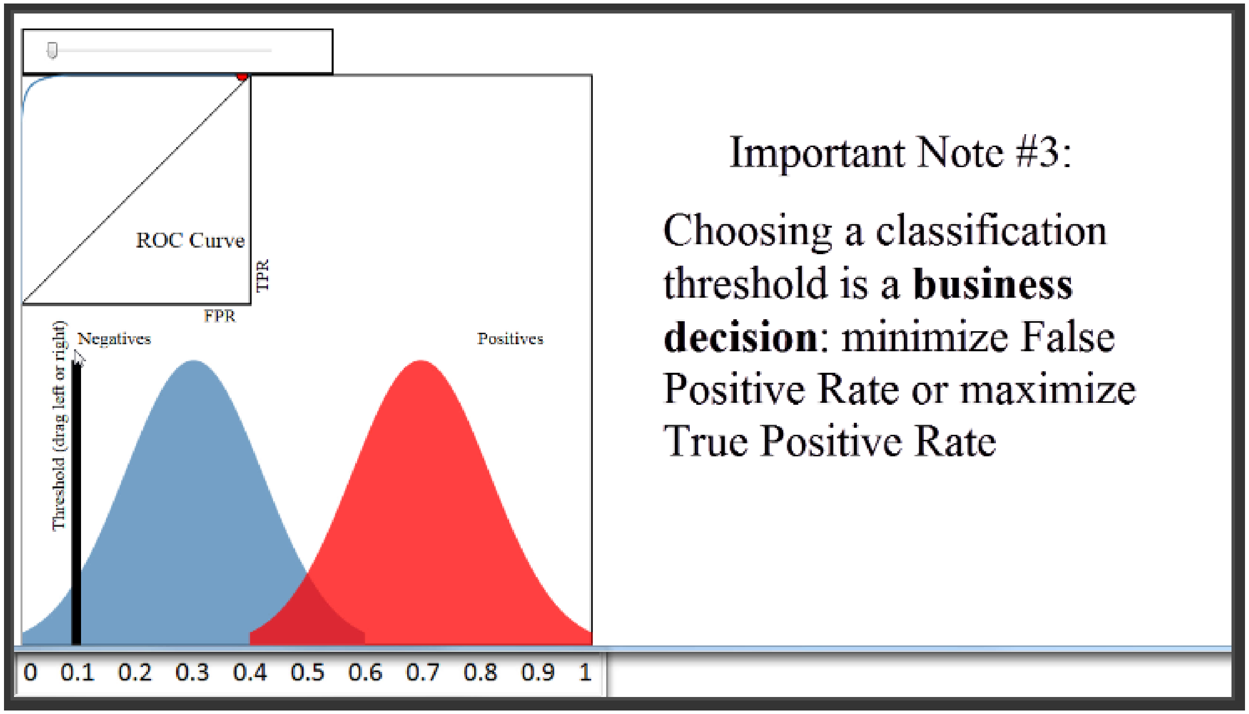


Important Note #2:

ROC curves can be extended to problems with **three or more** classes:

class 1 vs classes 2 & 3
class 2 vs classes 1 & 3
class 3 vs classes 1 & 2

(**12:30**) Finally, you might be wondering how you should set your classification threshold, once you are ready to use it to predict out-of-sample data. That's actually more of a **business decision**, in that you have to decide whether you would rather **minimize your False Positive Rate or maximize your True Positive Rate**. In our journal example, it's not obvious what you should do. But let's say your classifier was being used to predict whether a given credit card transaction might be fraudulent and thus should be reviewed by the credit card holder. The business decision might be to set the threshold very low. That will result in a lot of false positives, but that might be considered acceptable because it would maximize the true positive rate and thus minimize the number of cases in which a real instance of fraud was not flagged for review.



Important Note #3:

Choosing a classification threshold is a **business decision**: minimize False Positive Rate or maximize True Positive Rate

(**13:34**) In the end, you will always have to choose a classification threshold, but the ROC curve will help you to visually understand the impact of that choice.

Thanks very much to Navan for creating this excellent **vizualization**. Below this video, I've linked to it as well as a very readable **paper** that provides a much more in-depth treatment of ROC curves. I also welcome your questions in the comments.

EMAIL FACEBOOK TWITTER LINKEDIN TUMBLR REDDIT GOOGLE+ POCKET

Data School Comment Policy

All comments are moderated, and will usually be approved by Kevin within a few hours. Thanks for your patience!

76 Comments **Data School** 1 **Login**

♡ **Recommend** 25 **Share** **Sort by Best**

Join the discussion…

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Mamatha Cherukuri • 3 years ago

Thank you..... For the great presentation.

22 ^ | ˅ • Reply • Share ›

Kevin Markham Mod ➦ Mamatha Cherukuri • 3 years ago

You're welcome!

2 ^ | ˅ • Reply • Share ›

Dora Jambor • 2 months ago

Thanks a lot for writing this super clear, intuitive and nice explanation! It's the best explanations I've seen so far!

1 ^ | ˅ • Reply • Share ›

Kevin Markham Mod ➦ Dora Jambor • 2 months ago

You're very welcome! Glad it was helpful to you!

^ | ˅ • Reply • Share ›

khubeb • 3 years ago

how do we calculate the Area under Curve (AUC) ?

1 ^ | ˅ • Reply • Share ›

Kevin Markham Mod ➦ khubeb • 3 years ago

AUC is literally the area under the ROC curve, meaning the percentage of the ROC "box" that is below the ROC curve. Many programming languages can calculate it for you, but if you want to calculate it manually, this is a useful explanation for how to do it: http://stats.stackexchange....

1 ⌃ | ⌄ • Reply • Share ›

Francesca Marazza • 2 months ago

Hi! Thank for your very exaustive explanation of ROC metric.
I have a question about false negative: if I have a lot of 0 and rare 1 (highly unbalanced classes), in my classification problem I have to worry about minimizing false negative rather than false positive rate. I think that ROC_AUC is not the best metrics in this case because this does not take into account false negative.
So, which metric can I use to evaluate better my classificator?
Thanks! Francesca

⌃ | ⌄ • Reply • Share ›

Kevin Markham Mod ➦ Francesca Marazza • a month ago

It sounds like you want to optimize for sensitivity, also known as recall.

⌃ | ⌄ • Reply • Share ›

charan the computer guy • 4 months ago

Hi,

I am trying to generate ROC curves for a dataset which has "Approved " and "Rejected" categories in the target column.
I have created KNN classifier. But coudln't generate proper ROC curves even though I had achieved 78% accuracy.

Please let me know how to generate ROC curves for binary classification using KNN.

Thanks
Sai Charan

⌃ | ⌄ • Reply • Share ›

Kevin Markham Mod ➦ charan the computer guy • 4 months ago

Kevin Markham Mod → charan the computer guy • 4 months ago

Hi Sai. How you would go about generating an ROC curve depends on your programming language. For example, this notebook contains my code for generating an ROC curve in Python with scikit-learn: https://github.com/justmark...

Hope that helps!

^ | ⌄ • Reply • Share ›

Bhekumuzi Mabheka • 5 months ago

Hi Kervin,

I have been working on classification models using Python and I've used your Python Notebooks from GitHub for learning.

There's a portion where you said Stratified Sampling its important in classification problem. So I would to know why it's important?

Kind Regards,

Michael(Junior Data Scientist).

^ | ⌄ • Reply • Share ›

Kevin Markham Mod → Bhekumuzi Mabheka • 5 months ago

When performing model evaluation via cross-validation, stratified sampling means that each response class will be represented with the same proportion in each of the folds. This is important because it most accurately simulates the real world, and thus is the best way to accurately estimate how your model will perform on real-world data. Hope that helps!

4 ^ | ⌄ • Reply • Share ›

Bhekumuzi Mabheka → Kevin Markham • 5 months ago

Thanks Kevin

4 ^ | ⌄ • Reply • Share ›

Pedro Torres • 9 months ago

Hello There, I teach classes on Biostatistics and I was wondering if you could share your app (or sheet), with the sliders. Thanks in advance.

^ | ˅ • Reply • Share ›

Kevin Markham Mod ➦ Pedro Torres • 8 months ago

I didn't create the visualization with the sliders, but here's the link to it: http://www.navan.name/roc/

Enjoy!

^ | ˅ • Reply • Share ›

Bharat Koti • a year ago

Could you please explain how are the values 50 and 235 calculated?

^ | ˅ • Reply • Share ›

Kevin Markham Mod ➦ Bharat Koti • a year ago

Assuming there were 250 pixels under the red curve, 50 is my estimate for the number of red pixels to the right of the line when the threshold is set at 0.8, and 235 is my estimate for the number of red pixels to the right of the line when the threshold is set at 0.5. In other words, these numbers are all just estimates.

^ | ˅ • Reply • Share ›

Nikita Mehta • a year ago

This was incredibly helpful! Thank you very much for the video presentation and reference paper! Can you explain Matthew's correlation coefficient and how it differs from AUC?

^ | ˅ • Reply • Share ›

Kevin Markham Mod ➦ Nikita Mehta • a year ago

Unfortunately, I don't have enough experience with MCC to give you a good answer to this question. I'm sorry!

^ | ˅ • Reply • Share ›

Fatemeh • a year ago

Can we compute AUC for discrete classifiers?

∧ | ∨ • Reply • Share ›

Kevin Markham Mod ➦ Fatemeh • a year ago

AUC is used to assess the performance of a binary classifier. Does that answer your question? Let me know!

∧ | ∨ • Reply • Share ›

Peter Neglen • a year ago

Thanks for a fantastic presentation. I am a surgeon with limited understanding of statistics. I have evaluated a test for measuring presence or non-presence of a type of obstruction. I have used the routine accuracy assessment by identifying the sensitivity, specificity etc. The test showed both poor sensitivity and specificity for detection of the obstruction. Would creating a ROC and measure the AUC have an additional role in this analysis? Would i expect to get any data that would change my clinical use of this test? Really appreciate your comment.

∧ | ∨ • Reply • Share ›

Kevin Markham Mod ➦ Peter Neglen • a year ago

Glad the video was helpful to you! In your situation, generating the ROC curve would allow you to visualize the tradeoffs you can make (more sensitivity for less specificity, or vice versa) by changing the threshold at which you predict presence of an obstruction. Calculating the AUC would allow you to compare different models using a single number to see which model is better. So, I think there is value in both.

If you are a Python user, I discuss these issues in more depth in video 9 of my machine learning series: http://www.dataschool.io/ma...

Hope that helps!

∧ | ∨ • Reply • Share ›

BolzanoWeierstrass • a year ago

Thank you for a very helpful presentation. I have a question regarding the last part, in choosing a classification threshold. Suppose the business decision is to maximise the True Positive Rate, since the ROC curve doesn't actually indicate what the probability threshold is for different points on the curve, how do we actually choose the threshold which maximises the True Positive Rate? Can the ROC-curve be used for this? Or is this something we have to do manually? Thanks for any help you can give me.

^ | ∨ • Reply • Share ›

Kevin Markham Mod ➦ BolzanoWeierstrass • a year ago

If all you want to do is maximize the True Positive Rate, then the threshold should be set at 0, and your True Positive Rate will be 100%. But I assume what you are actually trying to do is to choose a threshold that balances the True Positive Rate and False Positive Rate appropriately. In that case, you will either have to experiment with thresholds or write some custom code to choose the appropriate threshold, since you can't see the threshold on the ROC curve itself. If you are a Python user, I have an example of this in video 9 of my scikit-learn series: http://www.dataschool.io/ma...

Hope that helps!

^ | ∨ • Reply • Share ›

BolzanoWeierstrass ➦ Kevin Markham • a year ago

Oh thanks, I'll check it out! Thanks for the great materials!

^ | ∨ • Reply • Share ›

BolzanoWeierstrass ➦ BolzanoWeierstrass • a year ago

Nevermind, I suppose we can just plot the probability histogram and look at it...

^ | ∨ • Reply • Share ›

zahra zol • 2 years ago

hie. i watched the video. but i still don't know why my ROCs look so weird and why they aren't actually curves? can i send u the piece of code and data that results in plotting my ROCs?

^ | ∨ • Reply • Share ›

Kevin Markham Mod ➦ zahra zol • 2 years ago

Video 9 of my scikit-learn video series shows how to plot ROC curves in Python - perhaps that would be helpful to you?

Video: https://www.youtube.com/wat...
Related code: https://github.com/justmark...

^ | ˅ • Reply • Share ›

zahra zol ➦ Kevin Markham • 2 years ago

hie. thanks for answering but i know nothing from Python. all of my codes are in matlab. and the shapes are sooooo weired. i have no idea why it is so...

^ | ˅ • Reply • Share ›

Ashtray Kim • 2 years ago

FANTASTIC!!! BEST EXPLANATION I'VE EVER SEEN!!
and 1 question about important note#2 for perfect understanding :)

1. Should I always make 3 ROC curves and WHY?
If i have a datasets with 3 classes and i'm just interested in class 1, then i think i can just make 1 ROC curve (with class 1 positive vs class 2,3 negative). Or it could be done when preprocessing.(class 1 to positive and class 2&3 to negative).
I think i dont understand your good explanation. Can you tell me 3 ROC Curves needed example for understanding?

THANK YOU

^ | ˅ • Reply • Share ›

Kevin Markham Mod ➦ Ashtray Kim • 2 years ago

Thanks for your kind words!

Creating 3 ROC curves using the "one versus all" approach is the usual way for handling a 3-class problem. However, if you only care about 2 classes ("class 1" and "not class 1"), then you are correct that you only need to create 1 ROC curve. In that case, I would convert this to a binary classification problem in the preprocessing phase, as you suggested.

Hope that helps!

^ | ˅ • Reply • Share ›

RatSavage • 2 years ago

Thank you. This was exactly what I was missing. I was having a hard time making the conceptual leap from having a single point to having a curve. This was the explanation I was looking for.

^ | ˅ • Reply • Share ›

Kevin Markham Mod ➦ RatSavage • 2 years ago

You're very welcome! Glad it was helpful to you!

^ | ˅ • Reply • Share ›

m... b... • 2 years ago

Hi, as stated by others this is a nice simple explanation of what seems at first to be a little bit more magic than math when I started trying to learn about ROC curves and Signal Detection.
I'm doing some research that involves language perception and am getting some data where the responses from one group of participants is actually lower than chance on the TPRxFPR scale. I wonder if you have any information or sources about what this says. I have read that you can invert the results on a "negative result" that is still significantly less than .50 (chance) but I don't really understand what the implications are for that. Any help you could provide would be greatly appreciated.

^ | ˅ • Reply • Share ›

Kevin Markham Mod ➦ m... b... • 2 years ago

You are asking what to do if your AUC is less than 0.5, is that correct? In that case, you can indeed reverse all your predictions (make zeros into ones and ones into zeros), and your AUC will automatically be above 0.5. For instance, if your AUC is 0.3, reversing your predictions will change your AUC to 0.7.

As for why this would occur in the first place, the most likely explanation is that you accidentally reversed your labels at some point during the coding process.

^ | ⌄ • Reply • Share ›

m... b... ➦ Kevin Markham • 2 years ago

First, Thanks for your prompt reply, I wasn't really sure if this page was still being monitored.

Second, Well, it's not that it was miss coded - this is an experiment where people say respond and apparently one group responded inline with my hypothesis, and another responded more "opposite" - I don't really know why yet, but was just trying to figure out what a scientific rational would be to saying that I flipped the results to get a "greater than 0.5" curve instead of one that was statistically less.
I just haven't seem much in the ROC information online that even deals with points under the tpr=fpr line, or when or why it's ok to invert the data. I don't think I can invert it perse since it is actually opposite of my hypothesis, but I do want to say that the discrimination task was different than chance but just not inline with my hypothesis... does that make sense? I wasn't sure if really anything in the negative (below 0.5 or tpr=fpr line) was an actual reading and not just still chance, but "worse"...

^ | ⌄ • Reply • Share ›

Kevin Markham Mod ➦ m... b... • 2 years ago

It's not clear to me whether you are using ROC/AUC in the appropriate context. ROC/AUC is used to evaluate the performance of a binary classifier, and not to measure whether the results of an experiment validate a hypothesis.

I'm sorry I can't be of more help, but this is a case where a lot more discussion would be needed to provide appropriate advice! Good luck!

^ | ⌄ • Reply • Share ›

m... b... ➦ Kevin Markham • 2 years ago m... b... › Kevin Markham · 2 years ago

OK, sorry, please let me try one more time to explain... essentially I'm using the signal detection method to determine hits/misses/correct rejections/false alarms, when someone