



DIMENSIONLESS
TECHNOLOGY

MODELING THE EXPERT

An Introduction to Logistic Regression

INTRODUCTION



- D2Hawkeye, a medical data mining company.
- The company receives claims data.
- These are data that are generated when an insured patient goes to a medical provider to receive a diagnosis or to have a procedure, for example an x-ray, or to obtain drugs.
- The medical providers need to get compensated, so the claims data provide the means for them to be paid.
- An important question is whether we can assess the quality of health care given this claims data.
- Why assessing the quality of healthcare is an important objective.

Ask the Experts!



- Critical decisions are often made by people with expert knowledge
- Healthcare Quality Assessment
 - Good quality care educates patients and controls costs
 - Need to assess quality for proper medical interventions
 - No single set of guidelines for defining quality of healthcare
 - Health professionals are experts in quality of care assessment

Experts are Human

- Experts are limited by memory and time
- Healthcare Quality Assessment
 - Expert physicians can evaluate quality by examining a patient's records
 - This process is time consuming and inefficient
 - Physicians cannot assess quality for millions of patients

Replicating the Experts

- Can we develop analytical tools that replicate expert assessment on a large scale?
- Learn from expert human judgment
 - Develop a model, interpret results, and adjust the model
- Make predictions/evaluations on a large scale
- Healthcare Quality Assessment
 - Let's identify poor healthcare quality using analytics

Building the Dataset

Claims Data

Medical Claims

Diagnosis, Procedures,
Doctor/Hospital, Cost

Pharmacy Claims

Drug, Quantity, Doctor,
Medication Cost

- Electronically available
- Standardized
- Not 100% accurate
- Under-reporting is common
- Claims for hospital visits can be vague

Creating the Dataset- Claims Sample

Claims Sample

- Large health insurance claims database
- Randomly selected 131 diabetes patients
- Ages range from 35 to 55
- Costs \$10,000 – \$20,000
- September 1, 2003 – August 31, 2005

Creating the Dataset- Expert Review

Claims Sample

Expert Review

- Expert physician reviewed claims and wrote descriptive notes:
 - “Ongoing use of narcotics”
 - “Only on Avandia, not a good first choice drug”
 - “Had regular visits, mammogram, and immunizations”
 - “Was given home testing supplies”

Creating the Dataset- Expert Assessment

Claims Sample

- Rated quality on a two-point scale (poor/good)

Expert Review

“I’d say care was poor – poorly treated diabetes”

Expert Assessment

“No eye care, but overall I’d say high quality”

Creating the Dataset- Variable Extraction

Claims Sample

Expert Review

Expert Assessment

Variable Extraction

- Dependent Variable
 - Quality of care
- Independent Variables
 - ongoing use of narcotics
 - only on Avandia, not a good first choice drug
 - Had regular visits, mammogram, and immunizations
 - Was given home testing supplies

Creating the Dataset- Variable Extraction

Claims Sample

Expert Review

Expert Assessment

Variable Extraction

- Dependent Variable
 - Quality of care
- Independent Variables
 - Diabetes treatment
 - Patient demographics
 - Healthcare utilization
 - Providers
 - Claims
 - Prescriptions

Predicting Quality of Care

- The dependent variable is modeled as a binary variable
 - 1 if low-quality care, 0 if high-quality care
- This is a *categorical variable*
 - A small number of possible outcomes
- Linear regression would predict a continuous outcome
- How can we extend the idea of linear regression to situations where the outcome variable is categorical?
 - Only want to predict 1 or 0
 - Could round outcome to 0 or 1
 - But we can do better with logistic regression

Quick Question

- Which of the following dependent variables are categorical? (Select all that apply.)

☐ Deciding whether to buy, sell, or hold a stock

☐ The weekly revenue of a company

☐ The winner of an election with two candidates

☐ The day of the week with the highest revenue

☐ The number of daily car thefts in New York City

☐ Whether or not revenue will exceed \$50,000

Quick Question

- Which of the following dependent variables are binary? (Select all that apply.)

☐ Deciding whether to buy, sell, or hold a stock

☐ The weekly revenue of a company

☐ The winner of an election with two candidates

☐ The day of the week with the highest revenue

☐ The number of car thefts in New York City

☐ Whether or not revenue will exceed \$50,000

Logistic Regression

Logistic Regression

- Predicts the probability of poor care
 - Denote dependent variable “PoorCare” by y
 - $P(y = 1)$
- Then $P(y = 0) = 1 - P(y = 1)$
- Independent variables x_1, x_2, \dots, x_k
- Uses the Logistic Response Function

Poor Care = 1
Good Care = 0

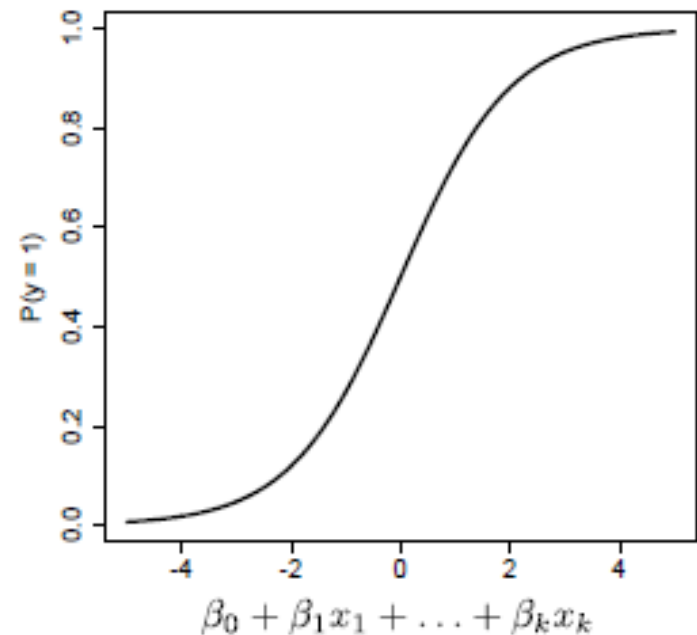
$$P(y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k)}}$$

- Nonlinear transformation of linear regression equation to produce number between 0 and 1

Understanding the Logistic Function

$$P(y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k)}}$$

- Positive values are predictive of class 1
- Negative values are predictive of class 0



Understanding the Logistic Function

$$P(y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k)}}$$

- The coefficients are selected to
 - Predict a high probability for the poor care cases
 - Predict a low probability for the good care cases

Understanding the Logistic Function

$$P(y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k)}}$$

- We can instead talk about Odds (like in gambling)

$$\text{Odds} = \frac{P(y = 1)}{P(y = 0)}$$

- Odds > 1 if $y = 1$ is more likely
- Odds < 1 if $y = 0$ is more likely

The Logit

- It turns out that

$$\text{Odds} = e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k}$$

$$\log(\text{Odds}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k$$

- This is called the “Logit” and looks like linear regression
- The bigger the Logit is, the bigger $P(y = 1)$




Quick Question

Suppose the coefficients of a logistic regression model with two independent variables are as follows:

$$\beta_0 = -1.5, \quad \beta_1 = 3, \quad \beta_2 = -0.5$$

And we have an observation with the following values for the independent variables:

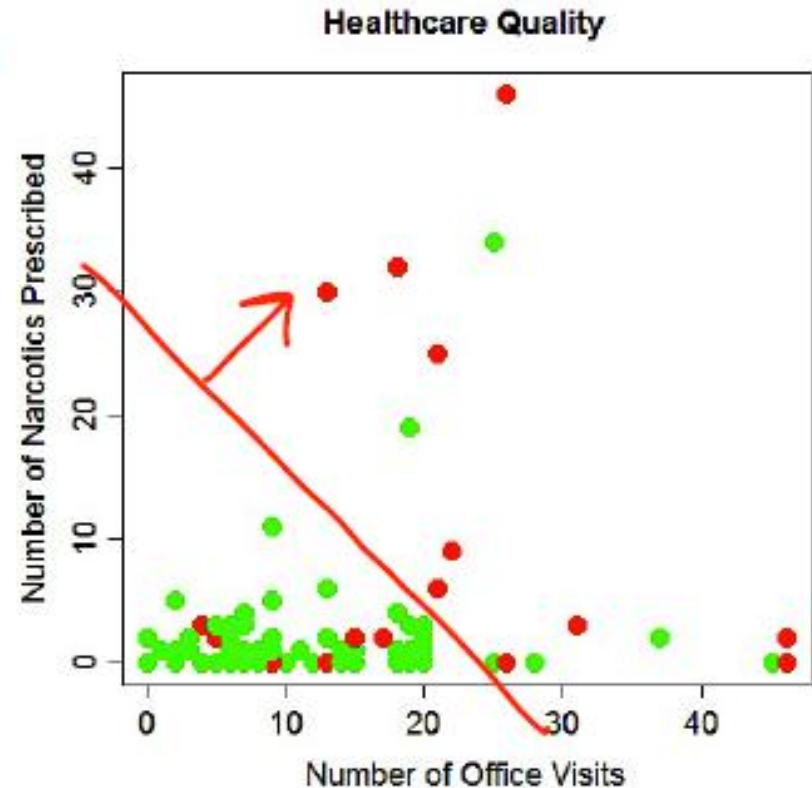
$$x_1 = 1, \quad x_2 = 5$$

- What is the value of the Logit for this observation? Recall that the Logit is $\log(\text{Odds})$. 
- What is the value of the Odds for this observation? 
- What is the value of $P(y = 1)$ for this observation? 

Logistic Regression in R

Model for HealthCare Quality

- Plot of the independent variables
 - Number of Office Visits
 - Number of Narcotics Prescribed
- Red are poor care
- Green are good care



Building the Model

- Read the data file "quality.csv" from <https://storage.googleapis.com/dimensionless/Analytics/quality.csv>
- Save it in data frame "quality"
- Look at the structure of the data frame
- We have 131 observations, one for each of the patients in our data set, and 14 different variables.

Variables



The variables in the dataset `quality.csv` are as follows:

- **MemberID** numbers the patients from 1 to 131, and is just an identifying number.
- **InpatientDays** is the number of inpatient visits, or number of days the person spent in the hospital.
- **ERVisits** is the number of times the patient visited the emergency room.
- **OfficeVisits** is the number of times the patient visited any doctor's office.
- **Narcotics** is the number of prescriptions the patient had for narcotics.
- **DaysSinceLastERVisit** is the number of days between the patient's last emergency room visit and the end of the study period (set to the length of the study period if they never visited the ER).

Variables



- **Pain** is the number of visits for which the patient complained about pain.
- **TotalVisits** is the total number of times the patient visited any healthcare provider.
- **ProviderCount** is the number of providers that served the patient.
- **MedicalClaims** is the number of days on which the patient had a medical claim.
- **ClaimLines** is the total number of medical claims.
- **StartedOnCombination** is whether or not the patient was started on a combination of drugs to treat their diabetes (TRUE or FALSE).
- **AcuteDrugGapSmall** is the fraction of acute drugs that were refilled quickly after the prescription ran out.
- **PoorCare** is the outcome or dependent variable, and is equal to 1 if the patient had poor care, and equal to 0 if the patient had good care.

Model



- The 12 variables from `'InpatientDays'` to `'AcuteDrugGapSmall'` are the independent variables.
- The final variable `'PoorCare'` is our outcome or dependent variable and is equal to 1 if the patient had poor care and equal to 0 if the patient had good care.
- We'll be using the `'number of office visits'` and the `'number of prescriptions for narcotics'` that the patient had for our model.
- After the lecture, try building models with different subsets of independent variables to see what the best model is that you can find.

Baseline Model



- Find out how many patients received poor care and how many patients received good care by using the table function.
- Build a simple baseline model.
 - In a classification problem, a standard baseline method is to just predict the most frequent outcome for all observations.
 - Since good care is more common than poor care, we would predict that all patients are receiving good care.
 - We would get 98 out of the 131 observations correct, or have an accuracy of about 75%.
- We need to build a logistic regression model, which can beat our baseline model.


Logistic Model



- We don't have train and test data separately.
- We need to break down the data set into training and testing data.
 - Install and load the package “caTools” in R.
 - Use the function `sample.split()`
 - `Sample.split()` randomly splits the data into 2. In the desired Split Ratio and returns a logical vector.
 - We will then subset our dataframe “quality” into `qualityTrain(70%)` and `qualityTest(30%)`.
- Now we are ready to build out logistic regression model using `OfficeVisits` and `Narcotics`
 - We will call our model ‘QualityLog’.

Logistic Model

- We will use `glm()` function (generalized linear model)
- `Glm()` takes `formula(y~x)`, `data`, and `family` as arguments.
- ```
QualityLog<-
glm(PoorCare~OfficeVisits+Narcotics,data=qualityTrain,family
= binomial)
```



# Model Summary



- Look at the summary of the model
- You can see the coeff. of Narcotics and OfficeVisits and you will find that both are positive. Which means that higher values in these two variables are indicative of poor care as we suspected from looking at the data.
- Both the coeff. are significant.
- It also shows AIC value. AIC is a measure of the quality of the model and is like Adjusted R-squared in that it accounts for the number of variables used compared to the number of observations.
- Unfortunately, it can only be compared between models on the same data set.
- It provides a means for model selection. The preferred model is the one with the minimum AIC.



# Making Predictions

- We will call it `'predictTrain'`
- Make predictions using the function `'predict'` on model `'QualityLog'`
- Look at the summary of `"predictTrain"`
- Minimum value is 0.07, Maximum Value = 0.98
- Check whether we are getting high probabilities for PoorCare. (hint:- Use `tapply`)

# Quick Question



- Create a logistic regression model to predict "PoorCare" using the independent variables "StartedOnCombination" and "ProviderCount". Use the training set qualityTrain.
- What is the coefficient for "StartedOnCombination"?
- StartedOnCombination is a binary variable, which equals 1 if the patient is started on a combination of drugs to treat their diabetes, and equals 0 if the patient is not started on a combination of drugs. All else being equal, does this model imply that starting a patient on a combination of drugs is indicative of poor care, or good care?

# Thresholding

# Threshold Value

- The outcome of a logistic regression model is a probability
- Often, we want to make a binary prediction
  - Did this patient receive poor care or good care?
- We can do this using a *threshold value*  $t$
- If  $P(\text{PoorCare} = 1) \geq t$ , predict poor quality
- If  $P(\text{PoorCare} = 1) < t$ , predict good quality
- What value should we pick for  $t$ ?

# Threshold Value



- Often selected based on which errors are “better”
- If  $t$  is **large**, predict poor care rarely (when  $P(y=1)$  is large)
  - More errors where we say good care, but it is actually poor care
  - Detects patients who are receiving the worst care
- If  $t$  is **small**, predict good care rarely (when  $P(y=1)$  is small)
  - More errors where we say poor care, but it is actually good care
  - Detects all patients who might be receiving poor care
- With no preference between the errors, select  $t = 0.5$ 
  - Predicts the more likely outcome

# Selecting a Threshold Value

Compare actual outcomes to predicted outcomes using a *confusion matrix* (classification matrix)

|            | Predicted = 0        | Predicted = 1        |
|------------|----------------------|----------------------|
| Actual = 0 | True Negatives (TN)  | False Positives (FP) |
| Actual = 1 | False Negatives (FN) | True Positives (TP)  |

TN:- Actually good care and for which we predict good care.

TP:- Actually Poor care and for which we predict poor care.

FP :- Predict poor care, but they're actually good care.

FN:- Predict good care, but they're actually poor care.

# Outcome Measures

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

(Measures the percentage of actual poor care cases that we classify correctly.)

(Often called the true positive rate.)

$$\text{Specificity} = \frac{TN}{TN + FP}$$

(Measures the percentage of actual good care cases that we classify correctly.)

(Often called the true negative rate.)

A model with a higher threshold will have a lower sensitivity and a higher specificity.

A model with a lower threshold will have a higher sensitivity and a lower specificity.

# Confusion Matrix in R

- Make some classification tables using different threshold values and the table function.
- Rows will be the true outcome.
- Columns will be the prediction.
- Case I:- `Threshold(t) = 0.5`

- `table(qualityTrain$PoorCare, predictTrain >= 0.5)`

FALSE TRUE

0 72 2

1 18 7

- Compute the Sensitivity and Specificity.
  - $\text{Sensitivity} = 7 / (18 + 7) = 7 / 25 = 0.28$
  - $\text{Specificity} = 72 / (72 + 2) = 72 / 74 = 0.97$



# Confusion Matrix in R

• Case II:- Let's increase the threshold value

- $\text{Threshold}(t) = 0.7$
- `table(qualityTrain$PoorCare, predictTrain >= 0.7)`

|   | FALSE | TRUE |
|---|-------|------|
| 0 | 73    | 1    |
| 1 | 19    | 6    |

- Compute the Sensitivity and Specificity.
  - $\text{Sensitivity} = 6/(6+19) = 6/25 = 0.24$
  - $\text{Specificity} = 73/(73+1) = 73/74 = 0.99$

• By increasing the threshold, the sensitivity decreased while the specificity increased.

# Confusion Matrix in R

• Case III:- Let's decrease the threshold value

- $\text{Threshold}(t) = 0.2$
- `table(qualityTrain$PoorCare, predictTrain >= 0.2)`

|   | FALSE | TRUE |
|---|-------|------|
| 0 | 48    | 26   |
| 1 | 7     | 18   |

- Compute the Sensitivity and Specificity.
  - $\text{Sensitivity} = 18 / (18 + 7) = 18 / 25 = 0.72$
  - $\text{Specificity} = 48 / (48 + 26) = 48 / 74 = 0.65$

• So with the lower threshold, our sensitivity went up.

# Quick Question

This question asks about the following two confusion matrices:

Confusion Matrix #1:

|            | Predicted = 0 | Predicted = 1 |
|------------|---------------|---------------|
| Actual = 0 | 15            | 10            |
| Actual = 1 | 5             | 20            |

Confusion Matrix #2:

|            | Predicted = 0 | Predicted = 1 |
|------------|---------------|---------------|
| Actual = 0 | 20            | 5             |
| Actual = 1 | 10            | 15            |

- What is the sensitivity of Confusion Matrix #1?
- What is the specificity of Confusion Matrix #1?

# Quick Question

Confusion Matrix #1:

|            | Predicted = 0 | Predicted = 1 |
|------------|---------------|---------------|
| Actual = 0 | 15            | 10            |
| Actual = 1 | 5             | 20            |

Confusion Matrix #2:

|            | Predicted = 0 | Predicted = 1 |
|------------|---------------|---------------|
| Actual = 0 | 20            | 5             |
| Actual = 1 | 10            | 15            |

•To go from Confusion Matrix #1 to Confusion Matrix #2 ,

☐ We increased the threshold value.

☐ We decreased the threshold value.

# ROC Curves

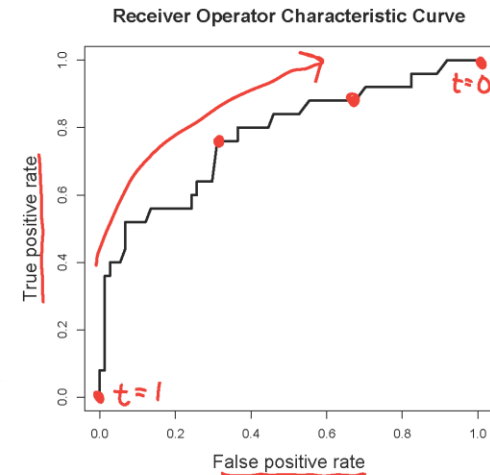
---

Deciding the Threshold

---

# Receiver Operator Characteristic (ROC) Curve

- The line shows how the two outcome measures vary with different threshold values.
- The ROC curve always starts at the point (0, 0). This corresponds to a threshold value of 1.
- If threshold is 1, you will not catch any poor care cases, or have a sensitivity of 0. But you will correctly label all the good care cases, meaning you have a false positive rate of 0.

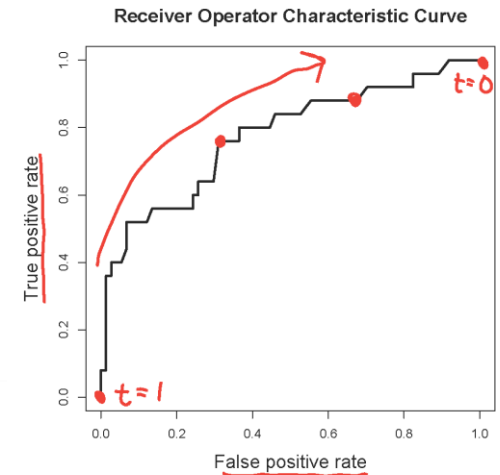


- True positive rate (sensitivity) on y-axis
  - Proportion of poor care caught
- False positive rate (1-specificity) on x-axis
  - Proportion of good care labeled as poor care

# Receiver Operator

## Characteristic (ROC) Curve

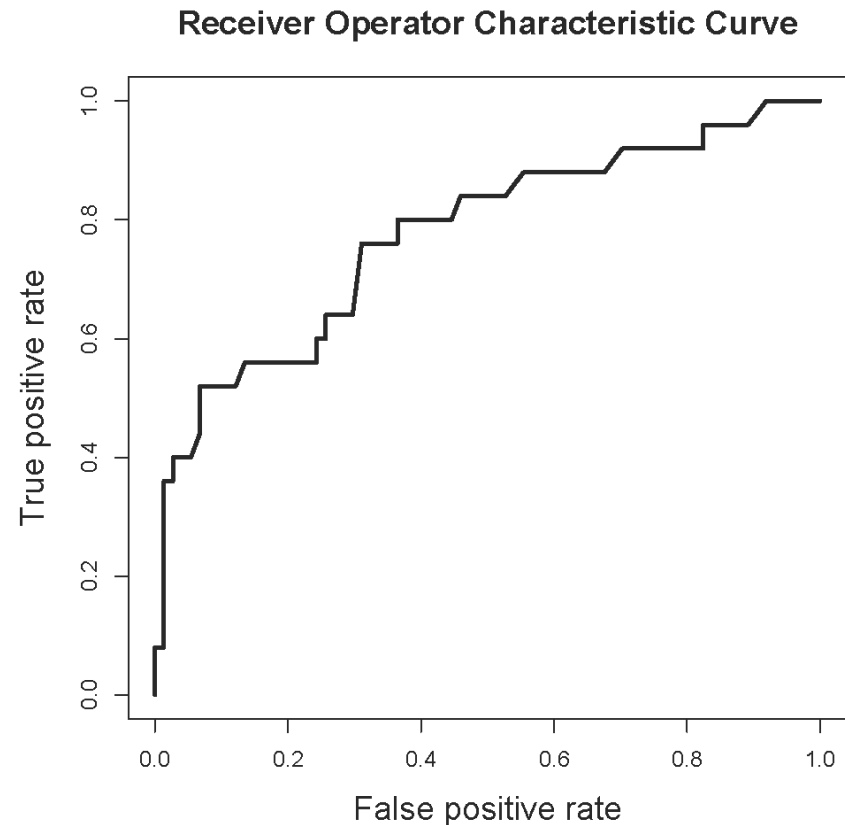
- The ROC curve always ends at the point (1,1) which corresponds to a threshold value of 0.
- If threshold of 0, you'll catch all of the poor care cases, or have a sensitivity of 1, but you'll label all of the good care cases as poor care cases too, meaning you have a false positive rate of 1.
- The threshold decreases as you move from (0,0) to (1,1).
- At the point (0, 0.4), you're correctly labelling about 40% of the poor care cases with a very small false positive rate.
- On the other hand, at the point (0.6, 0.9), you're correctly labeling about 90% of the poor care cases, but have a false positive rate of 60%.
- In the middle, around (0.3, 0.8), you're correctly labeling about 80% of the poor care cases, with a 30% false positive rate.



- True positive rate (sensitivity) on y-axis
  - Proportion of poor care caught
- False positive rate (1-specificity) on x-axis
  - Proportion of good care labeled as poor care

# Selecting a Threshold using ROC

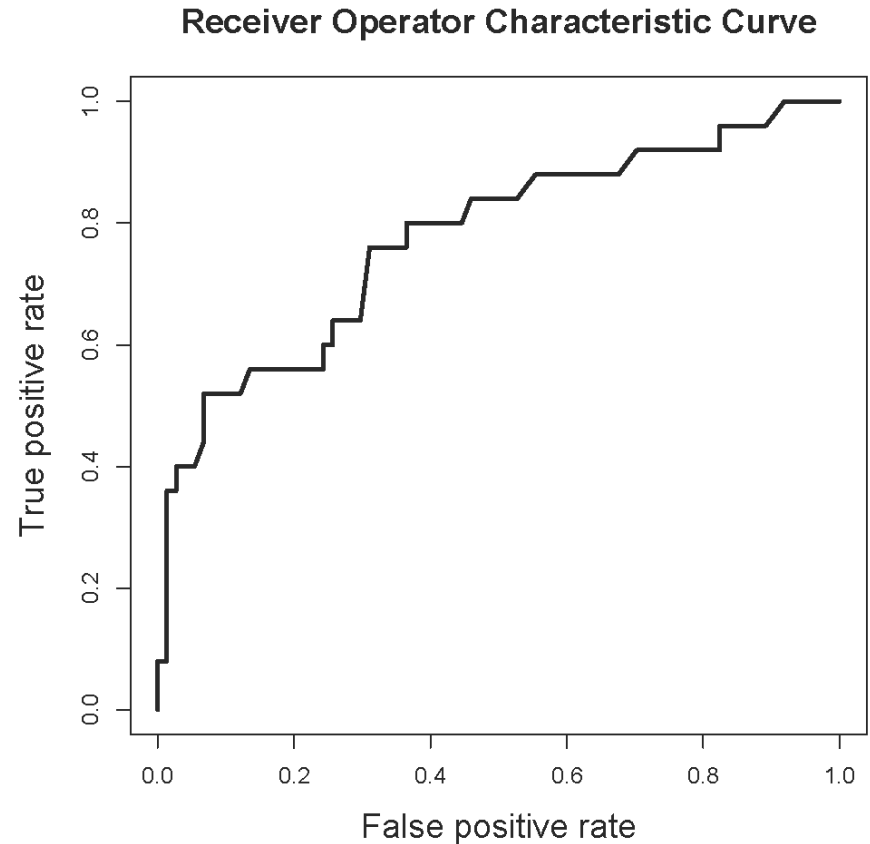
- Captures all thresholds simultaneously
- **High threshold**
  - High specificity
  - Low sensitivity
- **Low Threshold**
  - Low specificity
  - High sensitivity





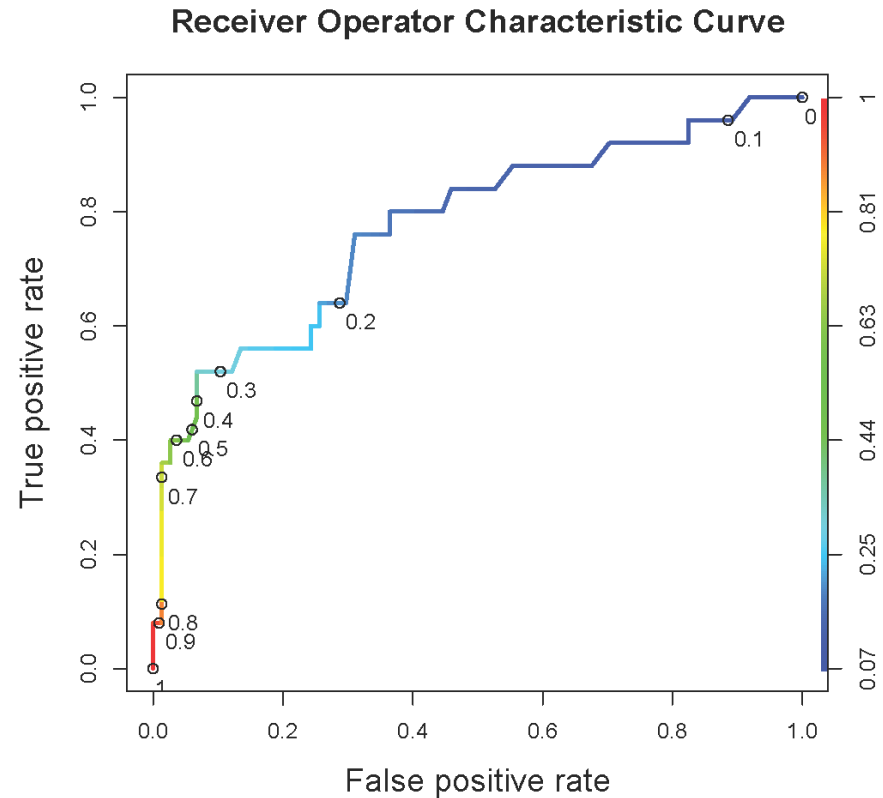
# Which threshold should we choose

- Choose **best threshold** for **best trade off**
  - cost of failing to detect positives
  - costs of raising false alarms



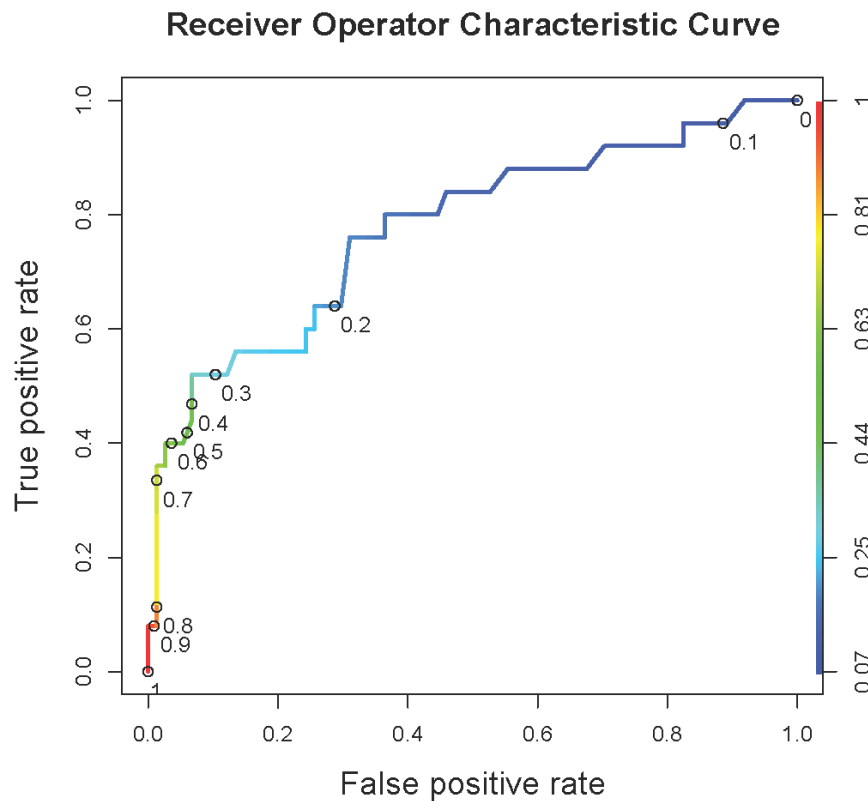
# Which threshold should we choose

- If you're more concerned with having a high specificity or low false positive rate, pick the threshold that maximizes the true positive rate while keeping the false positive rate really low.
- A threshold around (0.1, 0.5) on the ROC curve is a good choice.

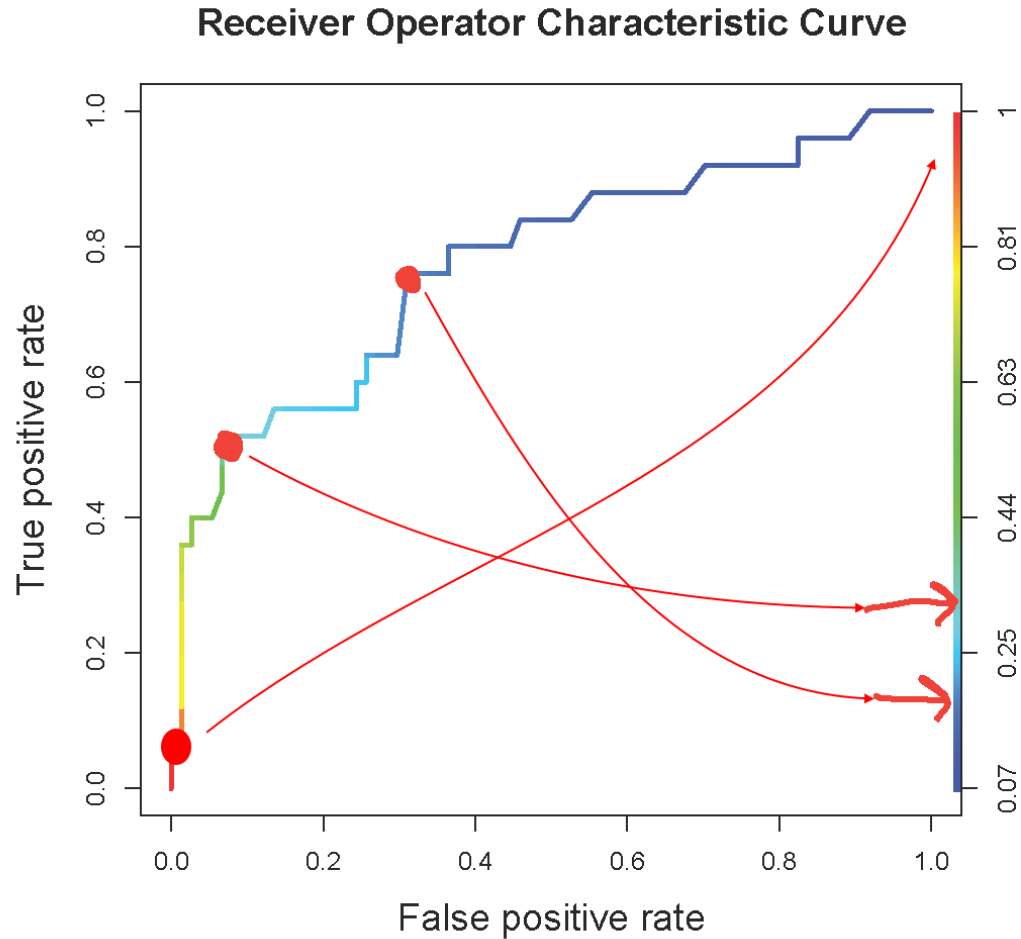


# Which threshold should we choose

- If you're more concerned with having a high sensitivity or high true positive rate, pick a threshold that minimizes the false positive rate but has a very high true positive rate.
- A threshold around (0.3, 0.8) looks like a good choice in this case.



# Legend of the ROC Curve



# ROC in R

# Prediction()

- To generate ROC curves in R, we need to install a new package "ROCR"
- Look at the various functions in ROCR by using `Library(help=ROCR)`
- Earlier we had made predictions on our training set and called them `predictTrain`.
- We'll use these predictions to create our ROC curve.
- We'll use the `prediction()` of the ROCR package.
- It takes 2 argument
  - The predictions we made (`predictTrain`)
  - The true outcomes of our dependent variable (`qualityTrain$PoorCare`)
- We will save the output in `ROCRpred`.
- `Prediction()` calculates the values of confusion matrix for various values of threshold.

# Performance()

- Now, we will use the performance function.
- It defines what we'd like to plot on the x and y-axes of our ROC curve.
- We'll call the output "ROCperf".
- Performance() takes as arguments the output of the prediction function, and then what we want on the x and y-axes.
- In this case, it's true positive rate, or "tpr", and false positive rate, or "fpr".
  - `ROCperf<-performance(ROCpred,"tpr","fpr")`

# ROC plot

- Now, we will plot the output of the performance function, `ROCperf`.

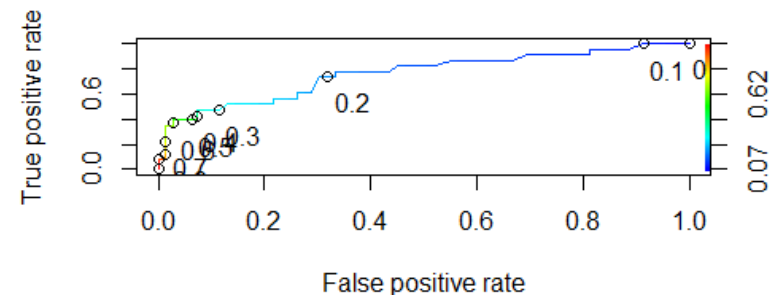
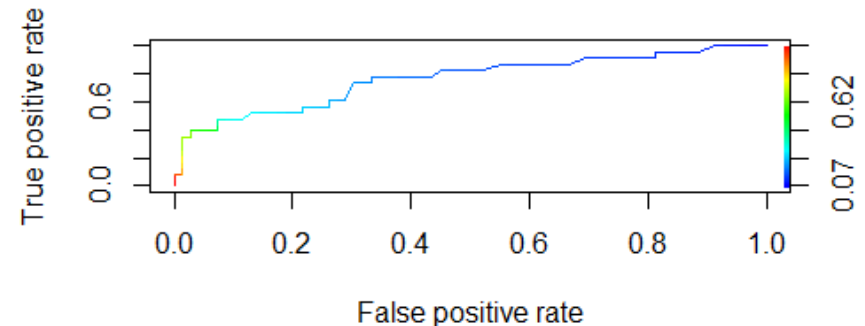
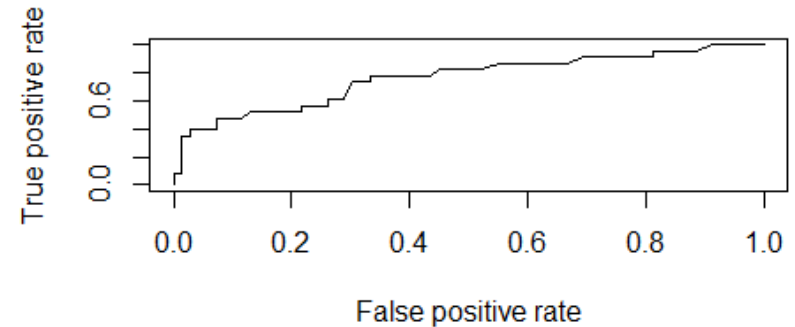
```
plot(ROCperf)
```

- Add colors to the plot
- ```
plot(ROCperf,colorize=TRUE)
```

- Add threshold labels to the plot.

```
plot(ROCperf,colorize=TRUE,print.cutoffs.at=seq(0,1,0.1),text.adj=c(-0.2,1.7))
```

- Using this curve, we can determine which threshold value we want to use depending on our preferences as a decision-maker.



Quick Question

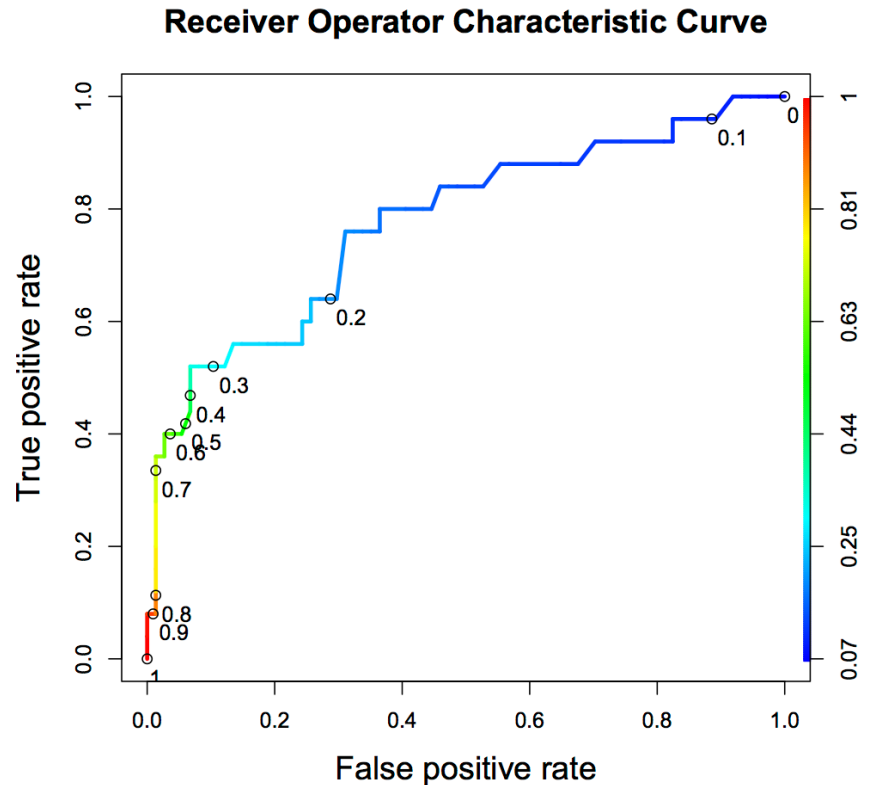
Which threshold would you pick if you wanted to correctly identify a small group of patients who are receiving the worst care with high confidence?

☐ $t = 0.2$

☐ $t = 0.3$

☐ $t = 0.7$

☐ $t = 0.8$



Quick Question

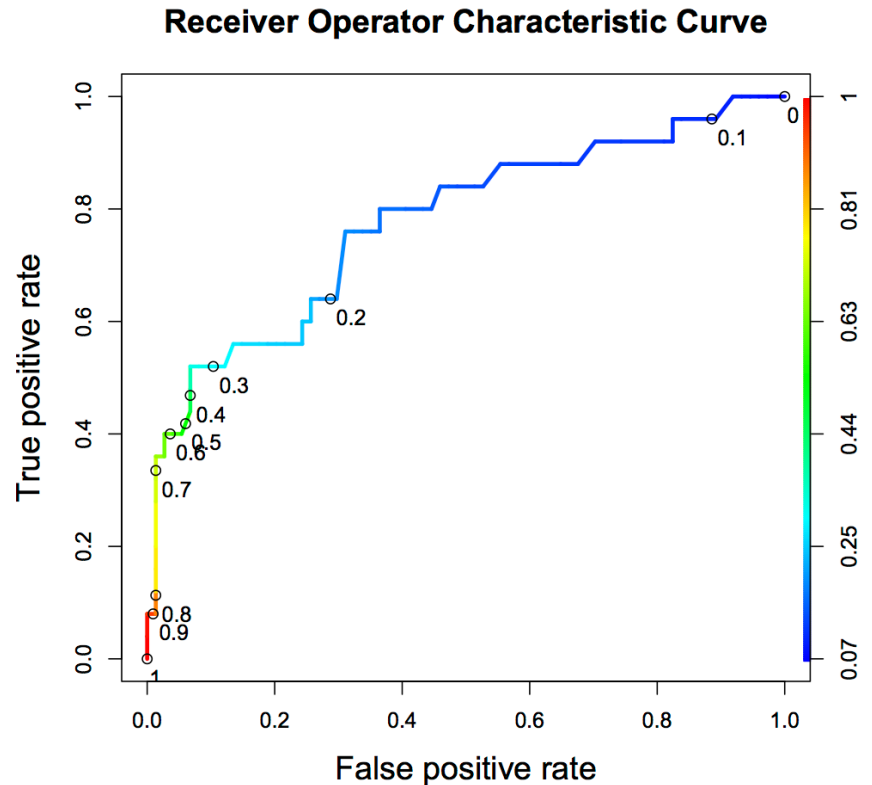
Which threshold would you pick if you wanted to correctly identify half of the patients receiving poor care, while making as few errors as possible?

☐ $t = 0.2$

☐ $t = 0.3$

☐ $t = 0.7$

☐ $t = 0.8$



Interpreting the Model



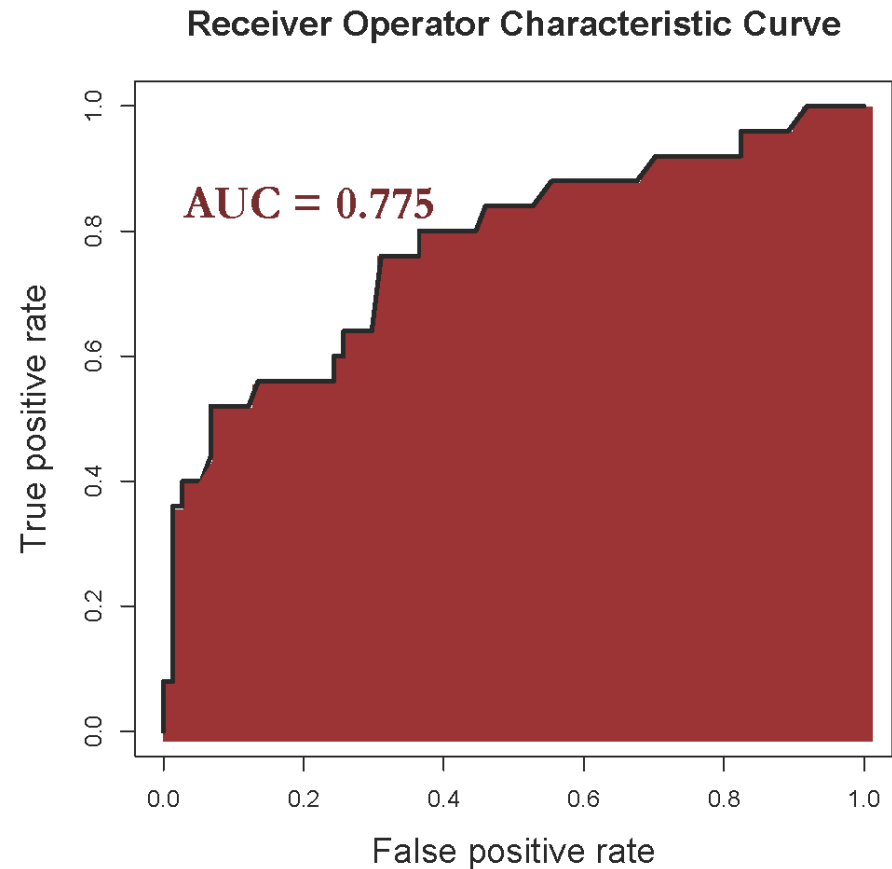
Interpreting the Model



- Multicollinearity could be a problem
 - Do the coefficients make sense?
 - Check correlations
- Measures of accuracy

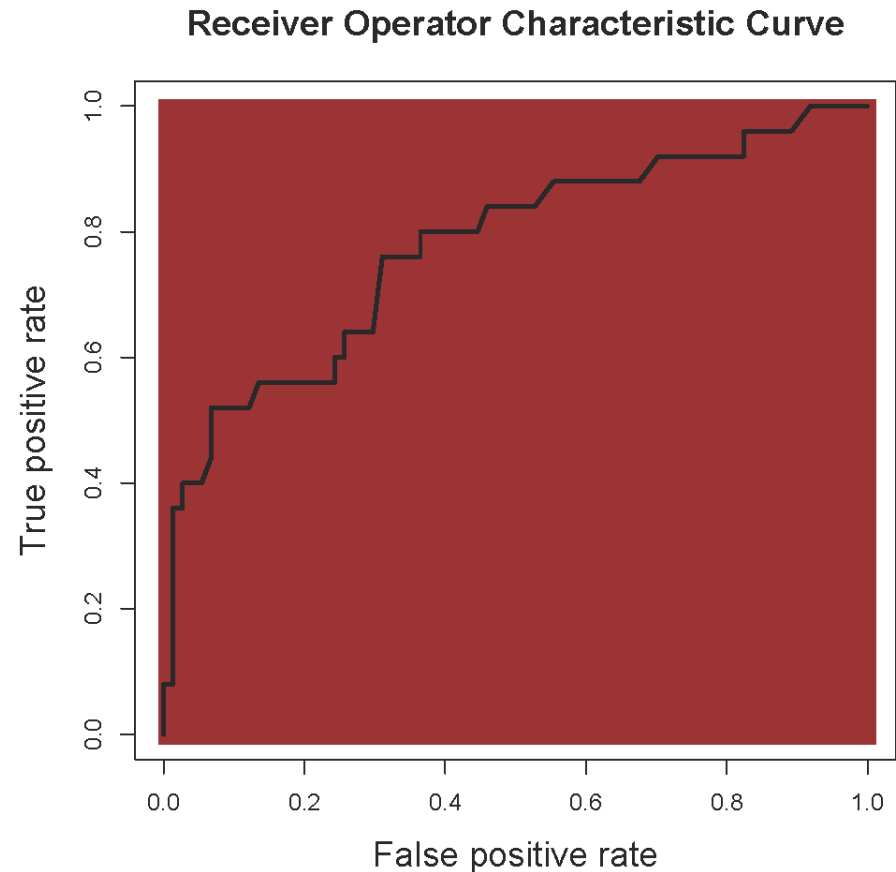
Area under the curve (AUC)

- Just take the area under the curve
- Interpretation
 - Given a random positive and negative, proportion of the time you guess which is which correctly
- Less affected by sample balance than accuracy



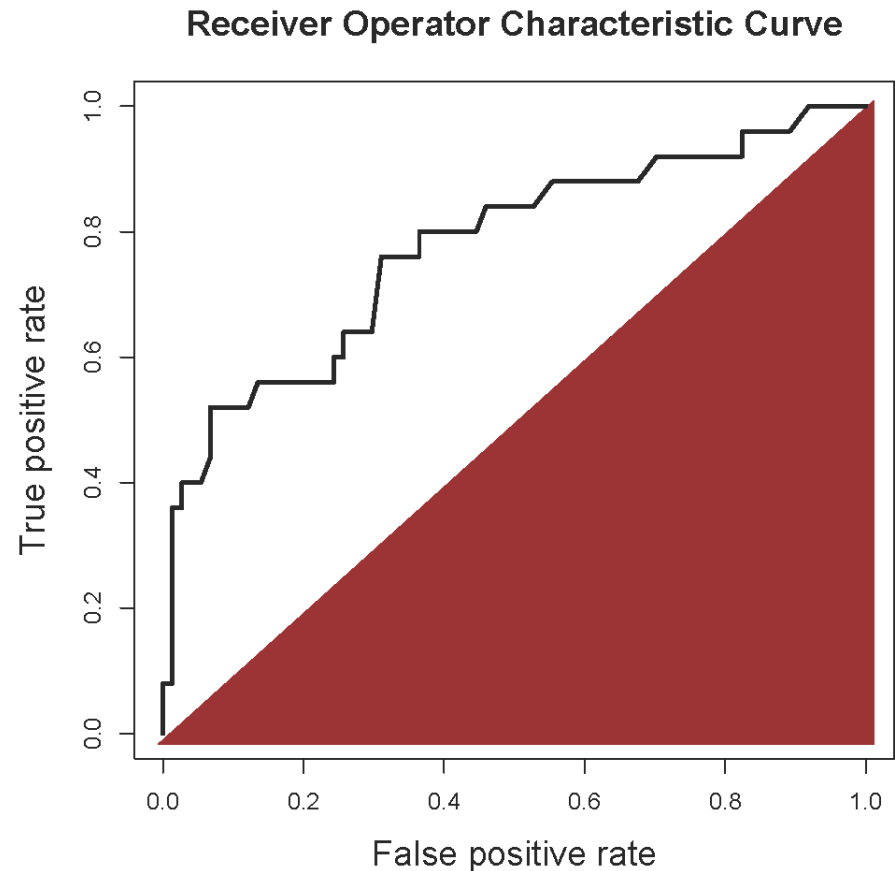
Area under the curve (AUC)

- What is a good AUC?
 - Maximum of 1
(perfect prediction)



Area under the curve (AUC)

- What is a good AUC?
 - Maximum of 1
(perfect prediction)
 - Minimum of 0.5
(just guessing)



Compute Outcome Measures

Confusion Matrix:

	Predicted Class = 0	Predicted Class = 1
Actual Class = 0	True Negatives (TN)	False Positives (FP)
Actual Class = 1	False Negatives (FN)	True Positives (TP)

N = number of observations

Overall accuracy = $(TN + TP)/N$

Overall error rate = $(FP + FN)/N$

Sensitivity = $TP/(TP + FN)$

False Negative Error Rate = $FN/(TP + FN)$

Specificity = $TN/(TN + FP)$

False Positive Error Rate = $FP/(TN + FP)$

Making Predictions

Making Predictions

- Just like in linear regression, we want to make predictions on a test set to compute out-of-sample metrics

```
> predictTest = predict(QualityLog,  
  type="response", newdata=qualityTest)
```
- This makes predictions for probabilities
- If we use a threshold value of 0.3, we get the following confusion matrix

	Predicted Good Care	Predicted Poor Care
Actually Good Care	22	7
Actually Poor Care	2	8

Outcome Measures

- Overall Accuracy = $30/39 = 0.77$
- Overall Error Rate = $9/39 = 0.23$
- Sensitivity = $8/10 = 0.8$
- False Negative Rate = 0.2
- Specificity = $22/29 = 0.76$
- False Positive Rate = 0.24
- Baseline Model Accuracy = $29/39 = 0.75$

Quick Question

- Compute the test set predictions in R by running the command:
 - `predictTest = predict(QualityLog, type="response", newdata=qualityTest)`
- You can compute the test set AUC by running the following two commands in R:
 - `R0CRpredTest = prediction(predictTest, qualityTest$PoorCare)`
 - `auc = as.numeric(performance(R0CRpredTest, "auc")@y.values)`
- What is the AUC of this model on the test set?
 - PS:- The AUC of a model has the following nice interpretation: given a random patient from the dataset who actually received poor care, and a random patient from the dataset who actually received good care, the AUC is the percentage of time that our model will classify which is which correctly.

Conclusions

Conclusions



- An expert-trained model can accurately identify diabetics receiving low-quality care
 - Out-of-sample accuracy of 78%
 - Identifies most patients receiving poor care
- In practice, the probabilities returned by the logistic regression model can be used to prioritize patients for intervention
- Electronic medical records could be used in the future

The Competitive Edge of Models



- While humans can accurately analyze small amounts of information, models allow larger scalability
- Models do not replace expert judgment
 - Experts can improve and refine the model
- Models can integrate assessments of many experts into one final unbiased and unemotional prediction