# 7.1 CART : Classification and Regression Trees

Previous lessons => how to build, fit, evaluate, and use regression models.**Agenda:**

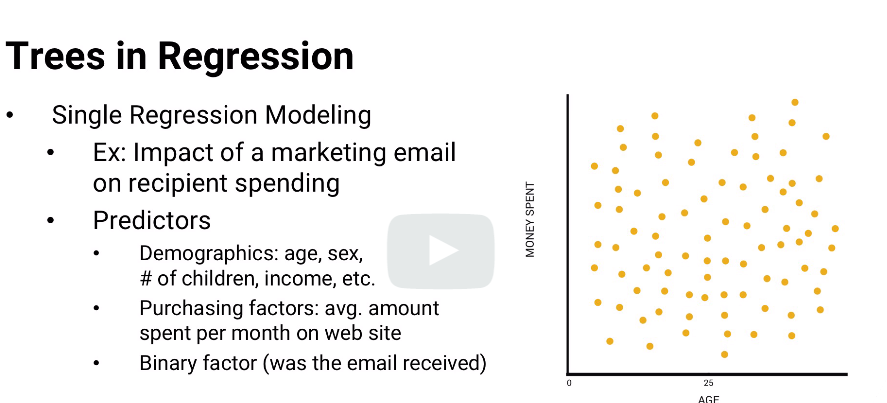
- how to use trees to divide the data set,-specify different models for each subset of the data.**Intro:**

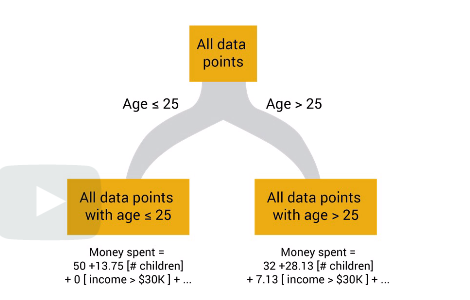
\* tree-based methods used in classification and regression

\* **decision tree :** Trees can also be used for decision making.

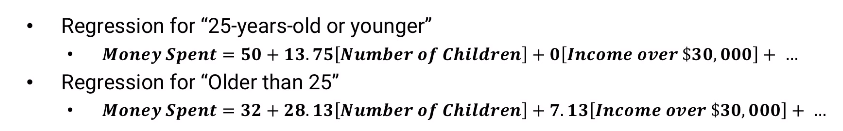
## Approach

**Conventional:**

- fit a single regression model using all of our training data. **example**: estimate the impact that an online retailer sending a marketing email hason the amount of money a recipient spends on their website. 

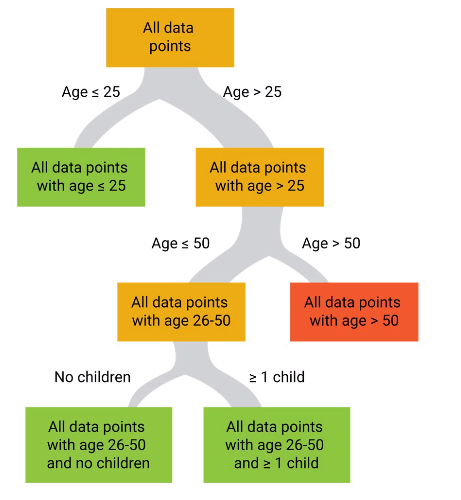
**Why decision trees?** what if some of those factors behave differently in different combinations?For example, maybe the coefficient for receiving a marketing email should really differ by age. 

For each branch, create a separate regression model using only the data points that fit the branch.**Note**

* choose different coefficients for the two sets of data points.
* some factors are significant in one branch but not in the other branch. 

**Extension:**

****

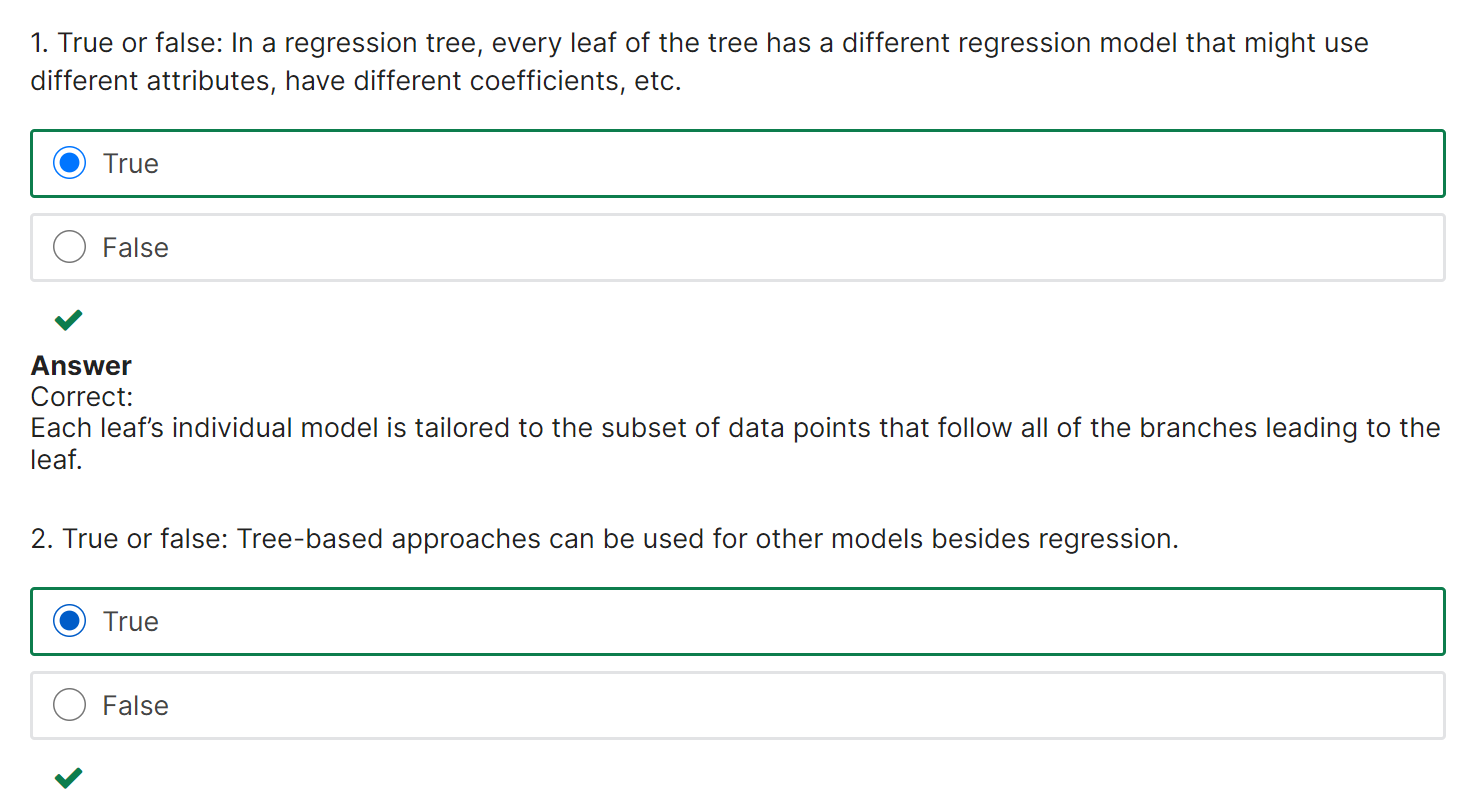
\*four branch endings referred to as leaves.\*each run a separate regression to find each leaf's individual set of coefficients.Use **descriptively**, we can use each leaf's specific coefficients to explain people's behavior within just that leaf,and **predictively**. build more targeted predictions.**figure out what we need to do better.** ****

three green leaves => good R square value

red leaf => low R square (meaning this model Is not good and need changes)

Classificaition

classification tree models each leaf could be a decision whether or not to send a marketing email,



# 7.2 Branching

1. How do we choose what branches to put in the tree?
2. When do we stop branching?

## How to branch?

There are really two questions involved in branching.

* Which factor or factors should be part of the branching decision
* how should they be split?

In theory, we could branch on any combination of factors.

**Example**: predict how much computation time a processor will require to optimize the assignment of courses to classrooms on campus,

* Branch 1: number of courses \* number of available classrooms is at least 1,000.
* Branch 2:number of courses \* number of available classrooms <1000

\* No good algorithm for determining good combinations to use.

\* Branch on one factor at a time.

**Approach for branch: (One of the method)**

1) start with half of the data

2) Build a regression model

3) Whenever there's a leaf we can branch from, calculate the variance of the response among all data points in the leaf.

4) Split on each factor to determine how much lower the total variance of the two branches would be compared to the least variance

5) choose the factor with the lowest total variance.

**Decision**: decrease in variance greater than some threshold delta + enough data points(atleast 5%)in each branch, we make the split.

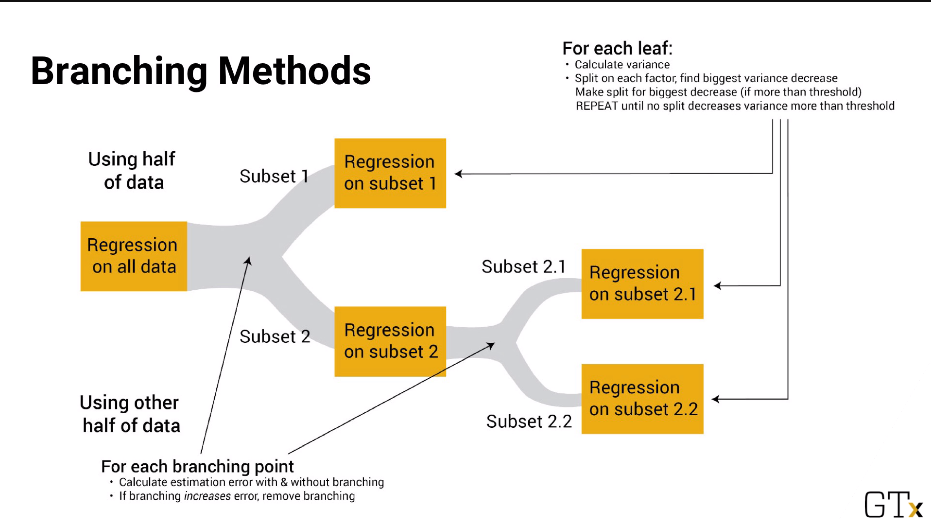
If not, stop the split

## PRUNE THE TREE

When done branching, use other half of the data to prune the tree (reverse process)

For every pair of leaves created by the same branch, use 2nd half of the data and find estimation error is actually improved by the branching.

If the branching does improve error, the branches stay. if not remove



**General Steps**

1) Using a metric that's related to the model's quality,

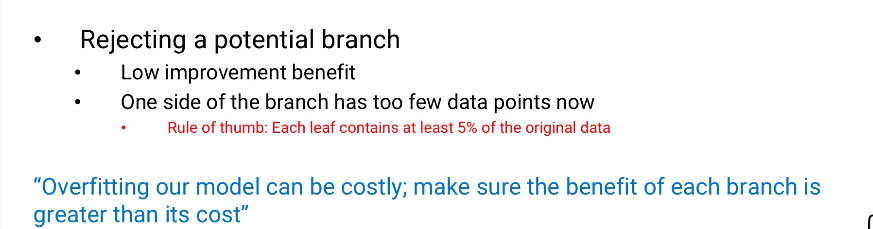
2) first find the best factor to branch on at a leaf ; best value of the factor to branch at.

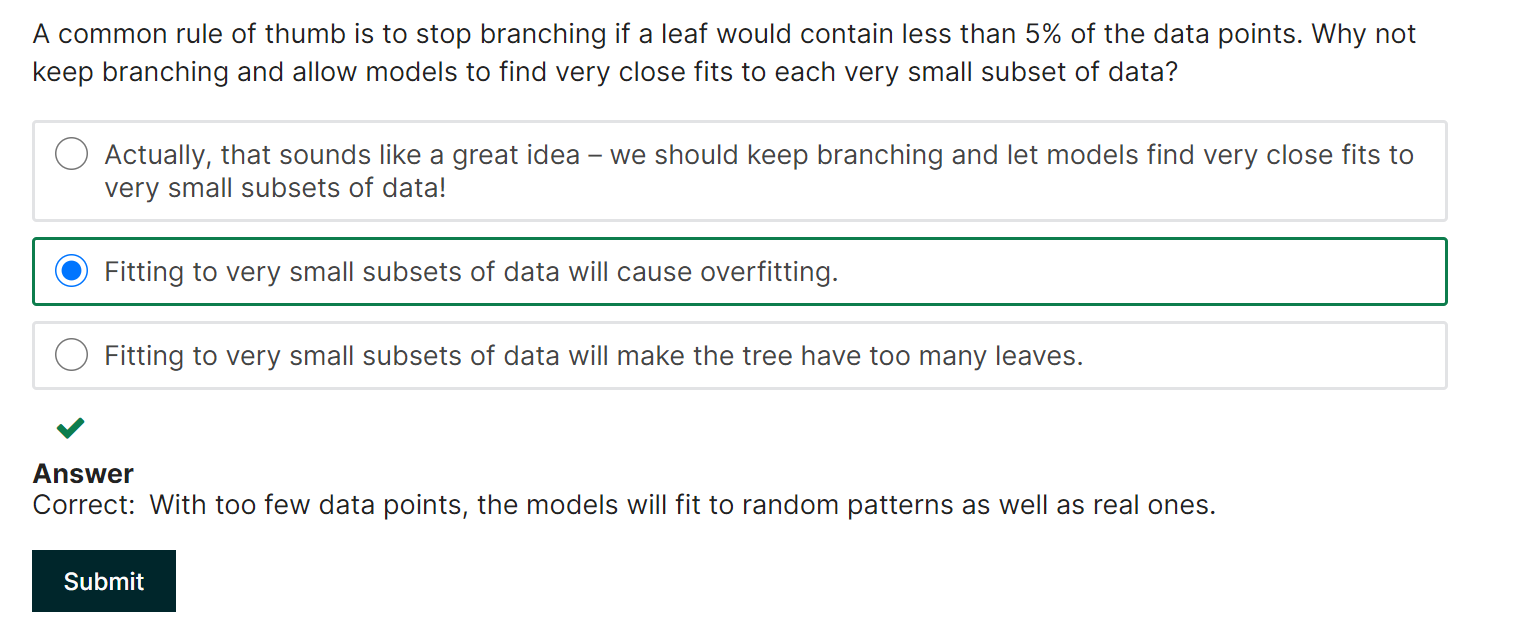
3) Check it really improves the model or else prune the branch back.

4) When branching, we need to have some sort of threshold for accepting a branch.

**Less Braches:** to avoid overfit the model.

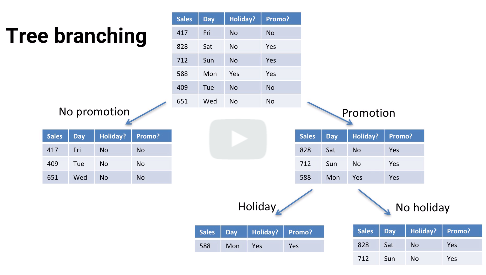
**Worst case:** we could just branch and branch and branch until every leaf has just one data point in it.





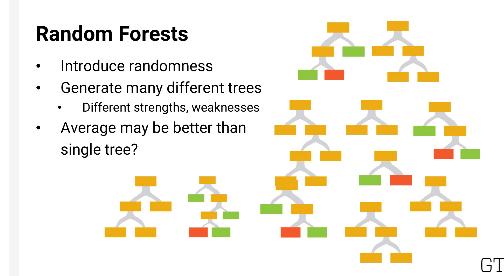
# 7.3 Random Forests

**Tree-based Approach:**



**Random Forest:**

* introduce randomness to allow us to generate different trees that might have different strengths and weaknesses.
* average of all these trees is better than a single tree with a specific strengths and weaknesses.

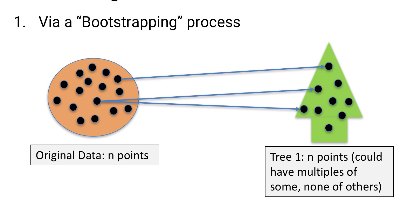


**How to introduce randomness?**

## PICK DATA POINT

1) Give each tree a slightly different set of data (randomly pick end data points for each of our trees)

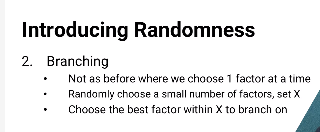
2) for each tree we might pick some of our points for that tree more than once,and never pick other points for it.



## BRANCHING

1) randomly pick a smaller number of factors and choose the factor in that subset(just not the one of our end factors that looks best to branch on)

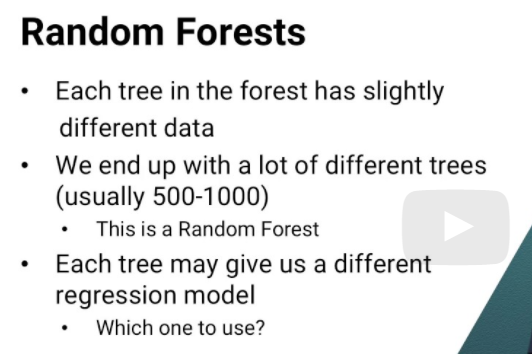
2) A common number of factors to use is 1+logN



## PRUNE: NOT required

Because we're randomly choosing factors at each step, and each tree has slightly different data

If we do the same procedure over and over and over again, we'll end up with lots of different trees.

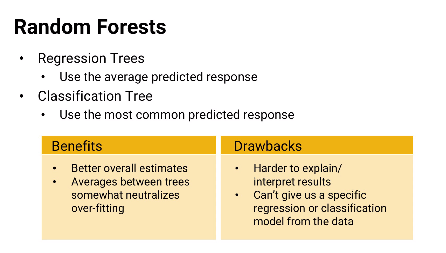


We've got a lot of models.How do we know which one to use?

**Answer** is we don't use a single one.

Regression Tree: use AVERAGE predicted response over all of the trees in our forest.

Classification Tree: use MODE, the most common predicted response over all the trees in our forest.



**Advantages**

* better estimates overall (average overall trees tends to flatten out those overreactions to random effects)

**Disadv:**

1) much harder to explain the output of a Random Forest model.

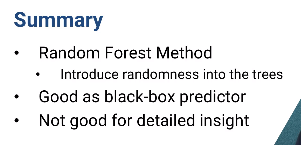
-calculate a relative importance of each variable, But, **doesn't** explain how the variables interact, or how a certain sequence of branches is helpful

MORE GENERAL AND NOT SPECIFIC INSIGHT with forest

(single tree can give us a specific regression or classification model for a set of data points,

the Random Forest can't)

Called **“DEFAULT model”** - good black box sort of predictive model quickly



# 7.4 Logistic Regression (classigication & probability)

**Assumption until so far** response (explain or predict) has a wide range.

Examples of Linear regression (response are continuous)

1) predict a child's height at adulthood,

2) number of sales of a popular product each week,

3) number of people who will see a particular tweet and it's retweets.

4) estimate the value of a home run in baseball,

5) impact of a kidney transplant on a potential recipient's longevity.

**Why Logistic?** estimate a probability

**Example:**

1) what's the probability that a loan recipient will repay the entire loan on time?

2) probability that one college basketball team will beat another

3) probability that a particular liver donor has an infectious form of encephalitis that could spread to a transplant recipient.

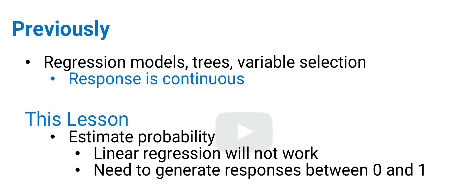
* Linear regression can generate responses outside the range of probabilities between zero and one, even though the data we observe will generally have responses that are either zero or one.

**Example: (0 or 1 responses)**

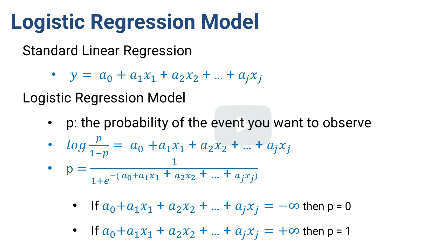
1) past loan recipients have either paid back their loan on time, or haven't.

College basketball teams have either won or lost,

and transplants recipients have either been infected, or haven't.

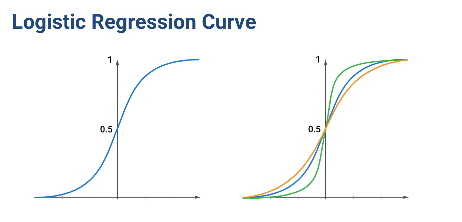


**Logistic regression model(responses as probablioty)**



* logistic model just takes that linear function and puts it into an exponential.
* Range : from negative infinity to positive infinity.
* If it's negative infinity, then the response will be zero,
* if it's positive infinity,then the response will be one.

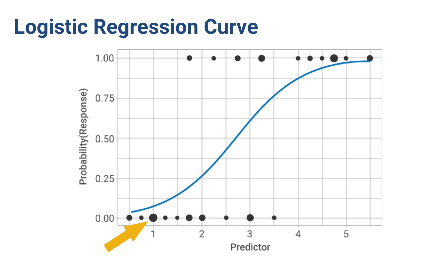
The values of the coefficients can change how steep the middle part of the curve is, and where the steep part is.



In below graph, all of the data points are at the top or bottom of the graph (0 or 1\_

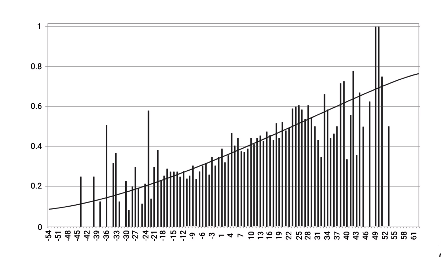
-hard to read

**-Suggestion :**add size of each dot would show how many observations there are.



**Alternate graph for logistic regression:**

For data where there are both zero and one responses for almost every predictor value, each bar shows the fraction of responses that are one for each predictor value.



\*We still need to consider how many observations there are for each predictor value

## Linear vs. logistic

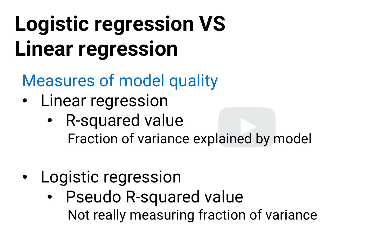
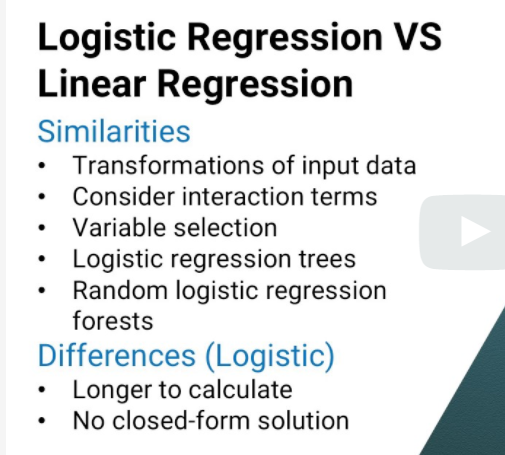
**Similarities**

1. use transformations of input data and consider interaction terms.
2. We can use variable selection methods for logistic regression, and we can build logistic regression trees and random logistic regression forests,

**Difference:**

1. Logistic regressions can take longer to calculate,because they don't have a closed-form solution (but they're still pretty fast to compute)
2. quality of a model : Linear regression has an R-squared value (estimates the fraction of variance and the response that is explained by the model)

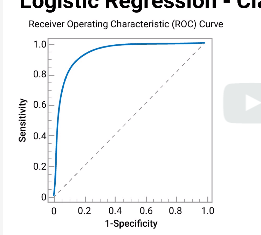
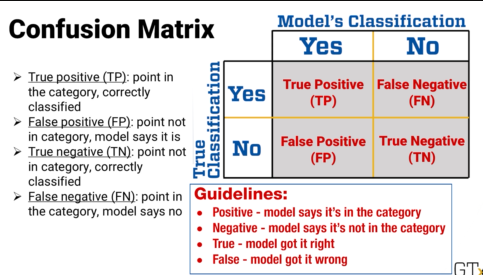
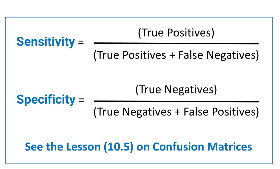
For logistic regression, they have **“pseudo R-squared”**



RLogistic regression used as classification:

* find coefficients as if we're trying to estimate a probability,
* use threshold for yes or no decision
* example, if the model estimates a probability of 0.7 or higher,we'll give this applicant a loan,otherwise we won't give the loan.

A common way of looking at this is to use a **receiver operating characteristic curve, ROC.**

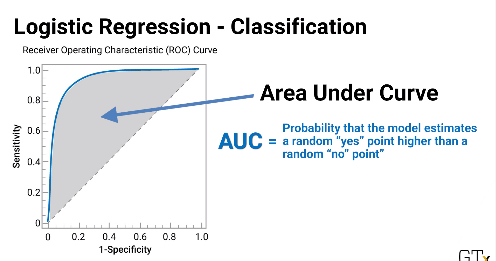
x-axis: (1-specificity)

y-axis: sensitivity, (model’ sensitivity for each threshold)

The area under that curve creatively called, **AUC, shows the probability**

**Example:** if we choose one random person, call him Joe, who repaid the loan, and one random person, Moe, who didn't, then the AUC would be probability

that the model gives Joe's data point a higher response value than Moe's.

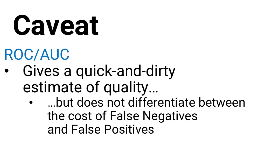


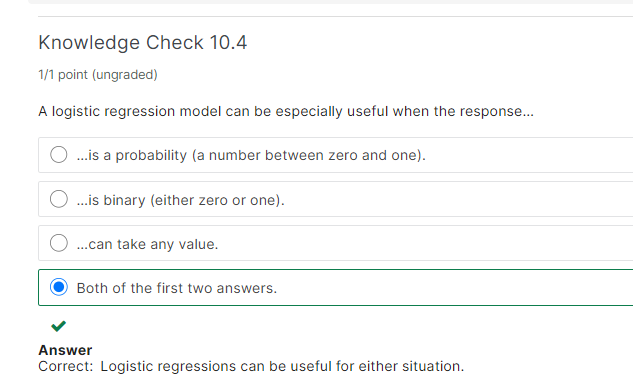
For reference, an AUC, also called **concordance index,** of 0.5, means that we're just really guessing.

Disavantage:

It doesn't differentiate between the cost of false negative and false positive results, among other things, but it can give you a quick and dirty estimate of model quality.

Suggestion: Use confusion matrix to get high value quality.





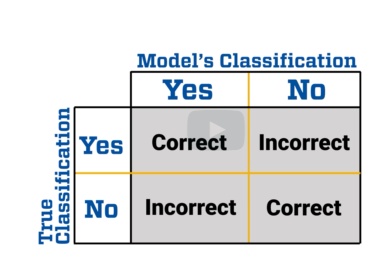
# 7.5 Confusion matrix

**Agenda:** how to measure how well a classification-type model works.

2 classification

1) pure classification model like a support vector machine or k-nearest-neighbor approach that categorizes data points directly

2\_ probability driven approach(logistic regresesion)

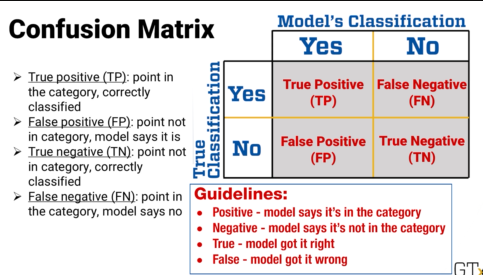


**confusion matrix answers below Questions:**

1. Of all the things in the category, how many does your approach correctly classify into that category and how many are not? (sensitivity)
2. of all the things that are not in the category,how many does your approach correctly say are not in the category? (specificity)
3. how many does it incorrectly classify as part of the category?

**Cell names:**

* true positive or TP: If the model correctly indicates that a data point is in the category
* false positive or FP.: If the model makes a mistake, it takes a data point that's not in the category and says it is
* true negative or TN : If the model correctly indicates that a data point is not in the category.
* false negative or FN: if the model takes a data point that is in the category and says it's not Here's a quick guideline to remembering these names.

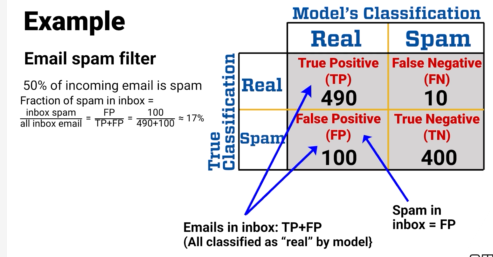


**example.**

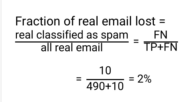
SVM to filter out spam from our email.

After testing it on 1000 emails, 500 of which are vaild and 500 of which are spam, here are the results.

**Question:** user wants to know what fraction of email in their inbox they should expect to be spam, if this **model is used as a filter,** how could they do it?



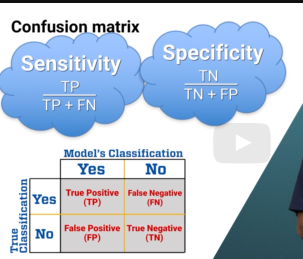
**Question#2:** user want to know what fraction of real email would get lost in the filter.



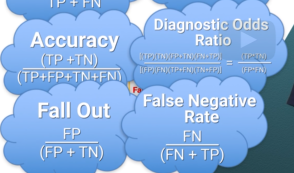
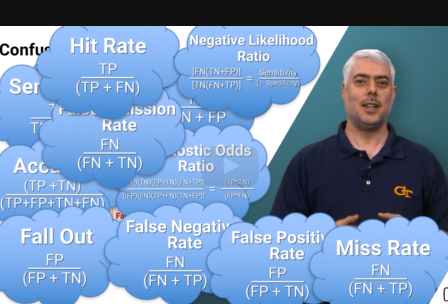
**common measures**

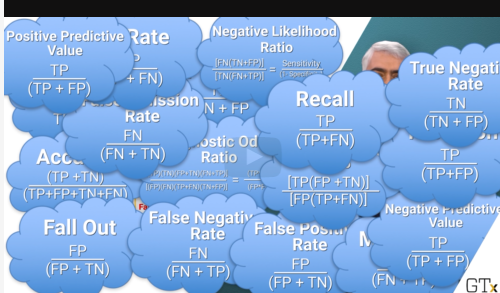
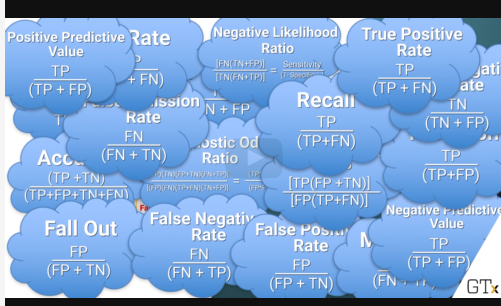
Sensitivity : fraction of category members that are correctly identified.

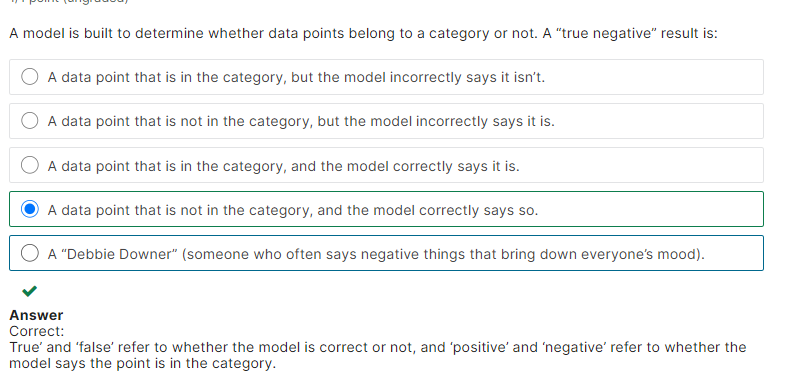
Specificity : fraction of non-category members that are correctly identified.



**Others:**

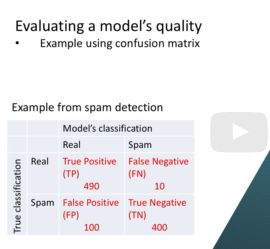
**** ****

**** ****

****

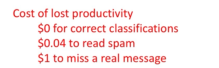
# 7.6 Situationally driven comparison

Estimate model’ quality using confusion matrix



Cost of lost productivity:

Waste time reading spam: $4 cents per mail

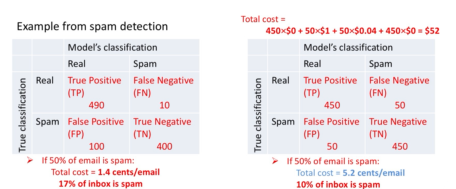
Scenario#2: only 40% of email is spam is 60% is true email

 - we scale the real emails from 50% to 60% and spam emails from 50% to 40%

Looking at % of false positives to all positive(100/(490+100))=17%

It may sound a lot and so we develop a strict model to reduce the number of false positives(to capture the spam email)

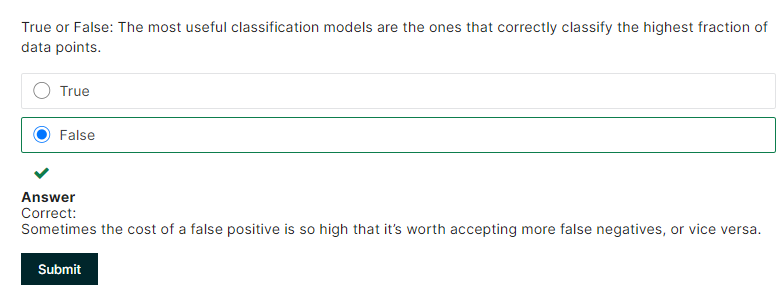
This new model captures spam, but more real emails are accidentally moved to spam



New model reduces the spam to 10% (i.e 1 out of 10 emails are spams). This seems nicer to reduce time to read spam messages

But the cost is 5.2 cents which is much more than first model

Also, the cost of missing an important email is more than reading a spam email.so that model#1 outweights model#2



# 7.7 Advanced Topics in regression

Advanced methods for regression

Parametrics method:

-we choose a form of predictor, like in SLR a(0)+suum of x(ij)

There are other non parametric method as well (no specific form is forced on the predictor)

Example: k-nearest neighbor and spline regression

## 7.7.1 poisson regression

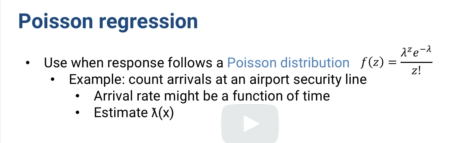
- used when response follows poisson distribution

-instead of single parameter “lamda” to estimate, we use parameter as function of other attributes

Example: base rate of arrival in an airport is function of day of week, holiday, time of day

-we use poisson expression to undersand the function lamda(x).

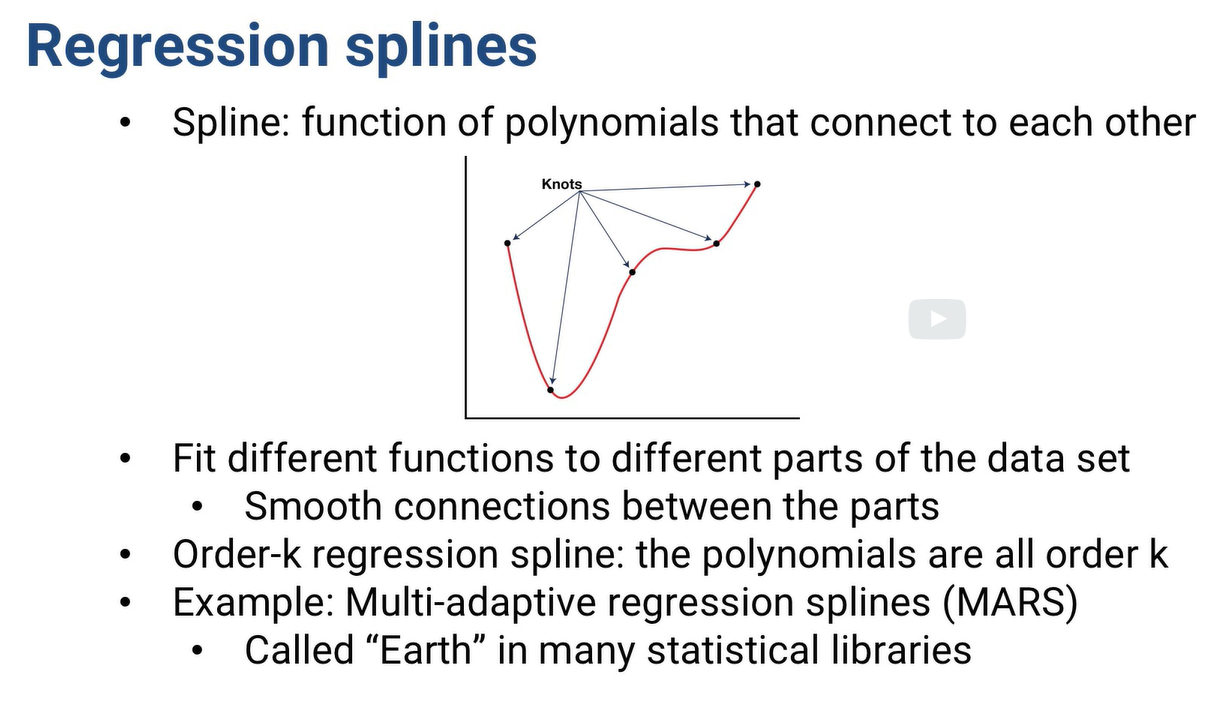
Then make prediction about expected arrival rate



## 7.7.2 regression splines

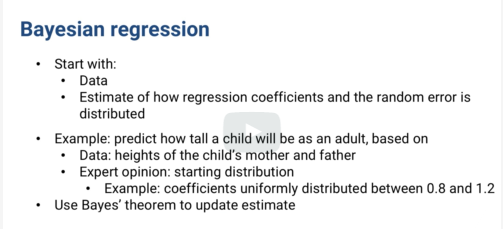
Spline is a function made of polynomials connected to each other

Knots: places where polynomials connect and the polynomial between knots should be smooth



## 7.7.3 Bayesian regression

-we just don’t with data, we have estimates on regression co-efficients and their random error distribution



**Advantages:**

* useful when we don’t have much data
* combine expert opinion+less data to generate model

(or)

Choose broad prior distributionand use it as “seed” data to adjust significantly.

## 7.7.4 K nearest neighbor regression

-simplest

-we don’t guess what function of predictor will be good predictor of response

-we just plot all the data

-predict response by average responses of “k” closet data points

