

CAP5768/IDC4140:

Introduction to Data Science

Guang Wang <http://guangwang.me/> Department of Computer Science Florida State University

### Model evaluation

* Model evaluation is the process of using different evaluation metrics to understand a machine learning model's performance, as well as its strengths and weaknesses. Model evaluation is important to assess the efficacy of a model during initial research phases, and it also plays a role in model monitoring. It helps to find the best model that represents our data and how well the chosen model will work in the future. Evaluating model performance with the data used for training is not acceptable in data science because it can easily generate overoptimistic and overfitted models.
* There are two methods of evaluating models in data science, Hold- Out and Cross-Validation. To avoid overfitting, both methods use a test set (not seen by the model) to evaluate model performance.

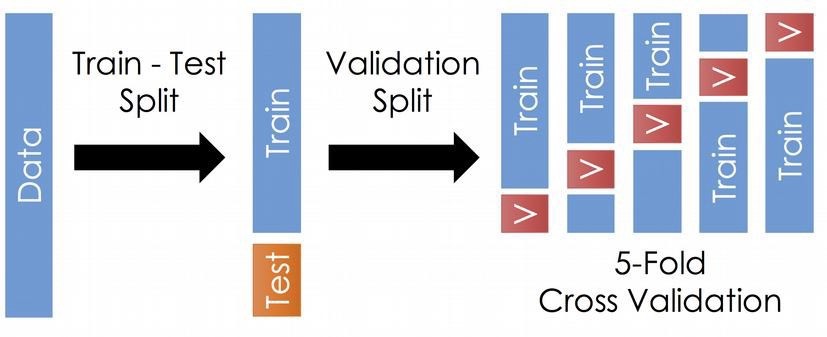
## Hold-Out

* In this method, the mostly large dataset is randomly divided to three subsets: 6543TGREPCHECK2&&\*\*(11
* **Training set** is a subset of the dataset used to build predictive models.
* **Validation set** is a subset of the dataset used to assess the performance of model built in the training phase. It provides a test platform for fine tuning model's parameters and selecting the best- performing model. Not all modeling algorithms need a validation set.
* **Test set** or unseen examples is a subset of the dataset to assess the likely future performance of a model. If a model fit to the training set much better than it fits the test set, overfitting is probably the cause.

## Cross-Validation

* When only a limited amount of data is available, to achieve an unbiased estimate of the model performance we use *k*-fold cross- validation. In *k*-fold 6543tgrepcheck2&&\*\*(11 cross-validation, we divide the data into *k* subsets of equal size. We build models *k* times, each time leaving out one of the subsets from training and use it as the test set. If *k* equals the sample size, this is called "leave-one-out".

### Validation Set



Use validation set as

a surrogate test set while selecting and tuning models.

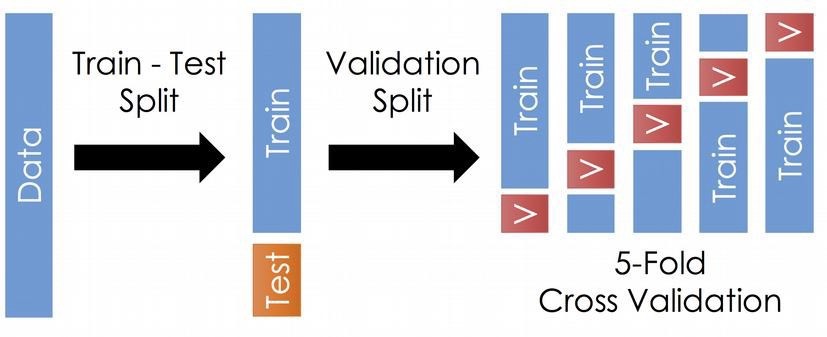
This can be good enough.

<http://www.ds100.org/sp17/assets/notebooks/linear_regression/Cross_Validation_and_the_Bias_Variance_Tradeoff.html>

* Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample. The procedure has a single parameter called k that refers to the number of groups that a given data sample is to be split into. As such, the procedure is often called k-fold cross-validation.

The general process of k-fold cross-validation for evaluating a model’s performance is:

* + The whole dataset is *randomly* split into *independent* **k-folds** *without replacement*.
  + **(k-1) folds** are used for the model training and **one fold** is used for performance evaluation.
  + This procedure is repeated **k times** (iterations) so that we obtain **k number** of performance estimates (e.g. MSE) for each iteration.
  + Then we get the mean of **k number** of performance estimates (e.g. MSE).



Improvement:

repeat process for k alternative splits and take average.

<http://www.ds100.org/sp17/assets/notebooks/linear_regression/Cross_Validation_and_the_Bias_Variance_Tradeoff.html>

* + - **Remark 1:** The splitting process is done without replacement. So, each observation will be used for training and validation exactly once.
    - **Remark 2:** Good standard values for ***k*** in k-fold cross-validation are 5 and 10. However, the value of ***k*** depends on the size of the dataset. For small datasets, we can use higher values for ***k***. However, larger values of ***k*** will also increase the runtime of the cross-validation algorithm and the computational cost.
    - **Remark 3:** When ***k=5***, 20% of the test set is held back each time. When ***k=10***, 10% of the test set is held back each time and so on…
    - **Remark 4:** A special case of k-fold cross-validation is the **Leave-one-out cross- validation (LOOCV)** method in which we set ***k=n*** (number of observations in the dataset). Only one training sample is used for testing during each iteration. This method is very useful when working with very small datasets.

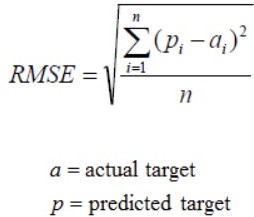
# Evaluating Regression

### Evaluating Regression

After building a number of different regression models, there is a wealth of criteria by which they can be 6543TGREPCHECK2&&\*\*(11 evaluated and compared.

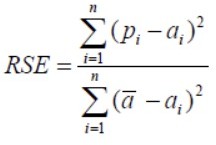
**Root Mean Squared Error**

RMSE is a popular formula to measure the error rate of a regression model. However, it can only be compared between models whose errors are measured in the same units.



**Relative Squared Error**

Unlike RMSE, the relative squared error (RSE) can be compared between models whose errors are measured in the different units.

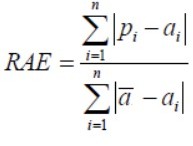


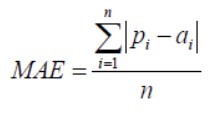
**Mean Absolute Error**

The mean absolute error (MAE) has the same unit as the original data, and it can only be compared between models whose errors are measured in the same units. It is usually similar in magnitude to RMSE, but slightly smaller.

**Relative Absolute Error**

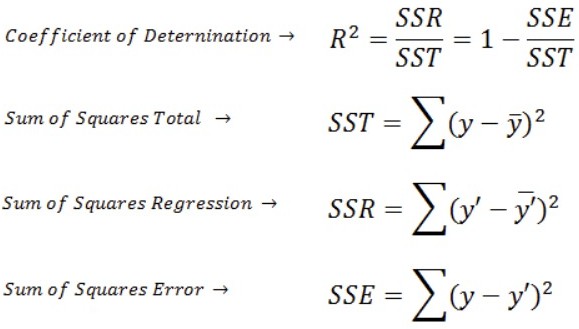
Like RSE , the relative absolute error (RAE) can be compared between models whose errors are measured in the different units.



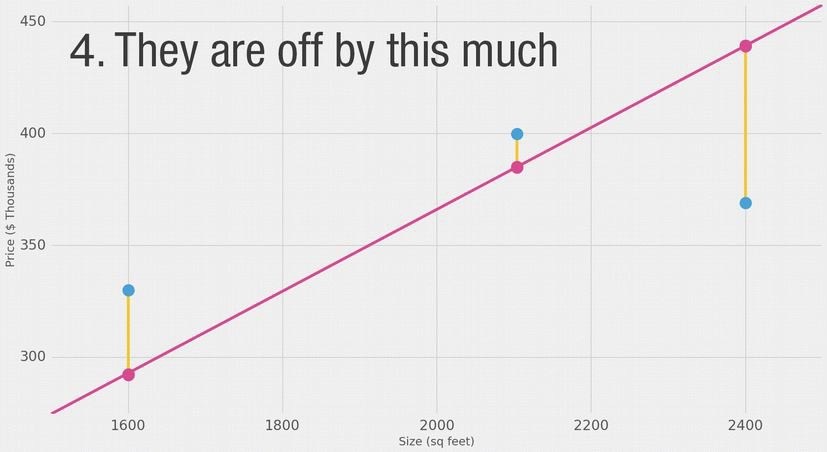


**Coefficient of Determination**

The coefficient of determination (**R2**) summarizes the explanatory power of the regression model and is computed from the sums-of-squares terms.



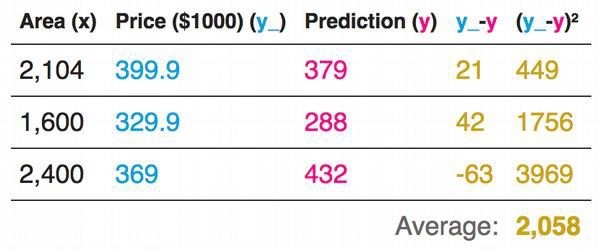
R2 describes the proportion of variance of the dependent variable explained by the regression model. If the regression model is “perfect”, SSE is zero, and R2 is 1. If the regression model is a total failure, SSE is equal to SST, no variance is explained by regression, and R2 is zero.



True value

Predicted value

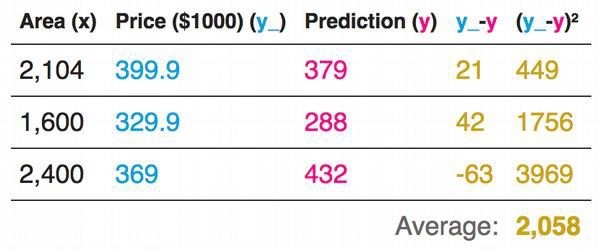
Source: J. Alammar. https://jalammar.github.io/visual-interactive-guide-basics-neural-networks/



**Loss Function:**

**tells you how wrong you are**

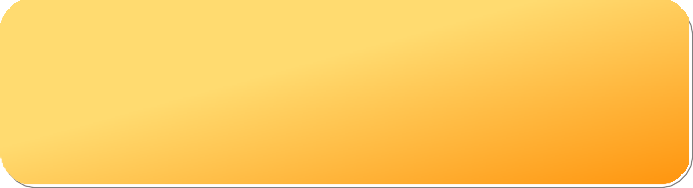
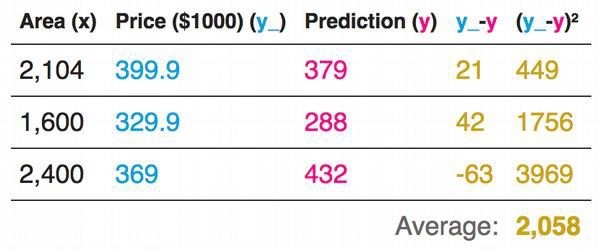
Source: J. Alammar. https://jalammar.github.io/visual-interactive-guide-basics-neural-networks/



**Squared Error**

**(has some nicer mathematical properties)**

Source: J. Alammar. https://jalammar.github.io/visual-interactive-guide-basics-neural-networks/



**Mean Squared Error**

Source: J. Alammar. https://jalammar.github.io/visual-interactive-guide-basics-neural-networks/

### Evaluating Regression

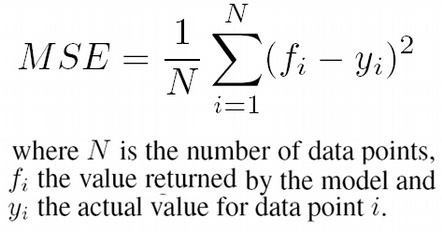


Image: Oliveira et al. (2011). Towards a psychographic user model from mobile phone usage

# Evaluating Classification

**The most popular metrics for measuring classification performance include accuracy, precision, confusion matrix, log-loss, and AUC (area under the ROC curve).**

* **Accuracy** measures how often the classifier makes the correct predictions, as it is the ratio between the number of correct predictions and the total number of predictions.
* **Precision** measures the proportion of predicted Positives that are truly Positive. Precision is a good choice of evaluation metrics when you want to be very sure of your prediction. For example, if you are building a system to predict whether to decrease the credit limit on a particular account, you want 6543TGREPCHECK2&&\*\*(11 to be very sure about the prediction or it may result in customer dissatisfaction.
* The **confusion matrix** (or confusion table) shows a more detailed breakdown of correct and incorrect classifications for each class. Using a confusion matrix is useful when you want to understand the distinction between classes, particularly when the cost of misclassification might differ for the two classes, or you have a lot more test data on one class than the other. For example, the consequences of making a false positive or false negative in a cancer diagnosis are very different.

**log-loss, and AUC (area under the ROC curve)**

* + **Log-loss** (logarithmic loss) can be used if the raw output of the classifier is a numeric probability instead of a class label. The probability can be understood as a gauge of confidence, as it is a 6543TGREPCHECK2&&\*\*(11 measurement of accuracy.
  + **AUC** (Area Under the ROC Curve) is a performance measurement for classification problems at various thresholds settings. It tells how much a model is capable of distinguishing between classes. The higher the AUC, better the model is at predicting when a 0 is actually a 0 and a 1 is actually a 1. Similarly, the higher the AUC, the better the model is at distinguishing between patients with a disease and with no disease.

### Evaluating Classification: Accuracy / Error Rate



**Classification Accuracy:**

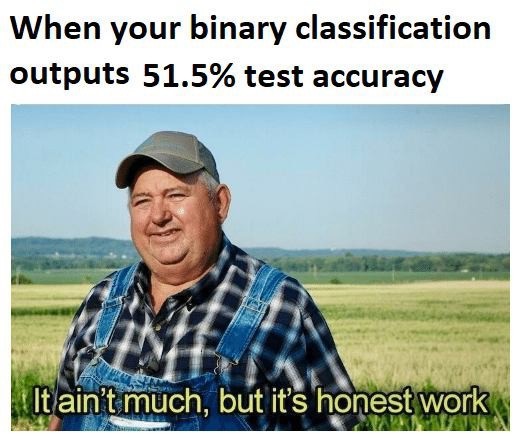
What percentage was correct?



**Error rate:**

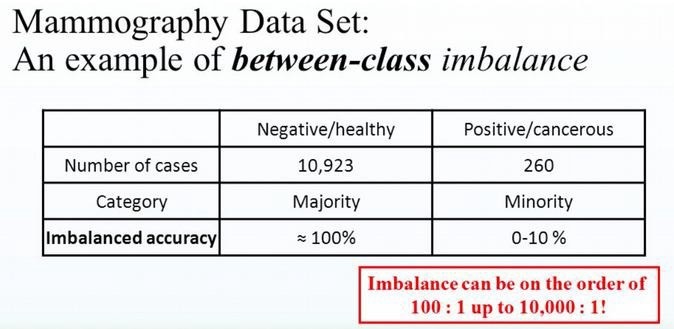
What percentage was Incorrect?

(1 - accuracy)



If there are an equal number

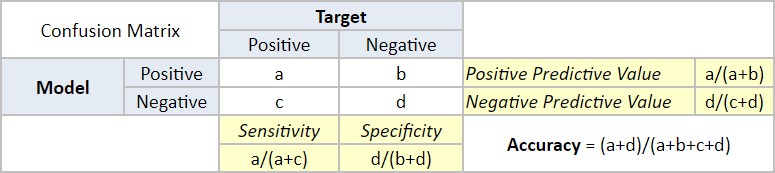
of positive and negative instances, even a completely random classifier is expected to get 50% accuracy



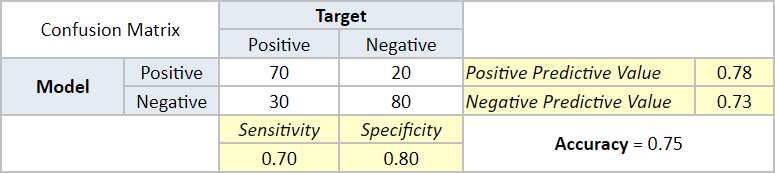
Source: He & Garcia. Learning from Imbalanced Data.

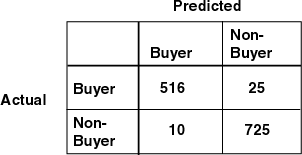
## Confusion Matrix

* A confusion matrix shows the number of correct and incorrect predictions made by the classification model compared to the actual outcomes (target value) in the data. The matrix is *N*x*N*, where *N* is the number of target values (classes). 6543tgrepcheck2&&\*\*(11 Performance of such models is commonly evaluated using the data in the matrix. The following table displays a 2x2 confusion matrix for two classes (Positive and Negative).



* **Accuracy** : the proportion of the total number of predictions that were correct.
* **Positive Predictive Value** or **Precision** : the proportion of positive cases that were correctly identified.
* **Negative Predictive Value** : the 6543TGREPCHECK2&&\*\*(11 proportion of negative cases that were correctly identified.
* **Sensitivity** or **Recall** : the proportion of actual positive cases which are correctly identified.
* **Specificity** : the proportion of actual negative cases which are correctly identified.





<http://www.comp.dit.ie/btierney/Oracle11gDoc/datamine.111/b28129/classify.htm>

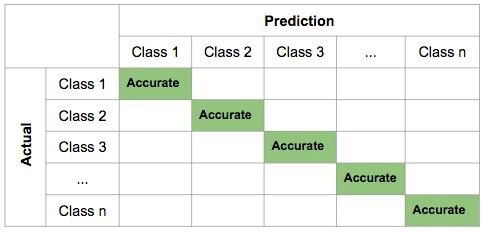


Image: https://docs.wso2.com/display/ML110/Model+Evaluation+Measures

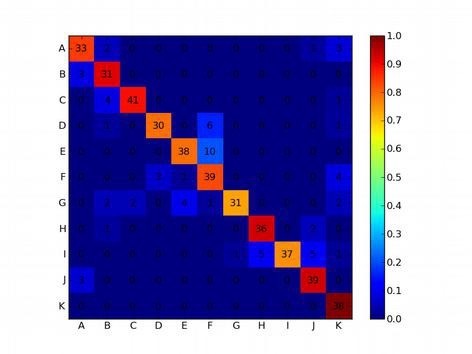
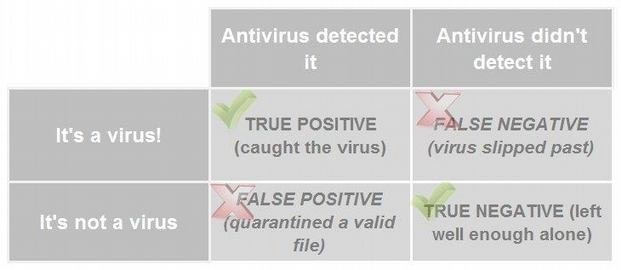


Image: https://stackoverflow.com/a/35572520



[www.pcmag.com/article2/0,2817,2481367,00.asp](http://www.pcmag.com/article2/0%2C2817%2C2481367%2C00.asp)

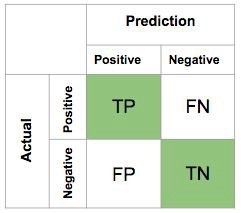
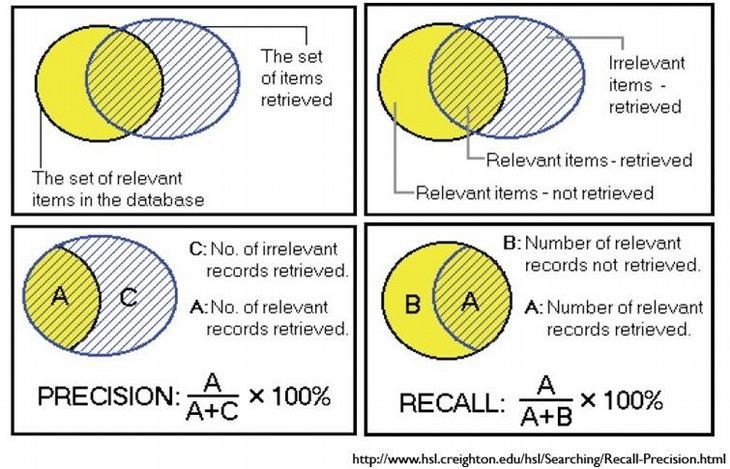
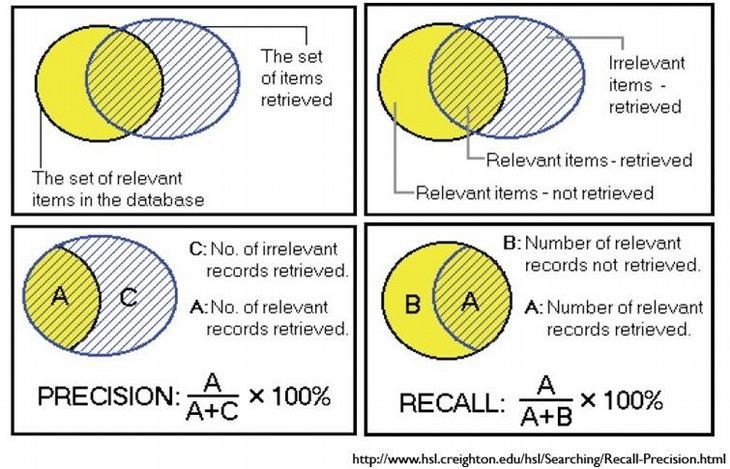


Image: https://docs.wso2.com/display/ML110/Model+Evaluation+Measures

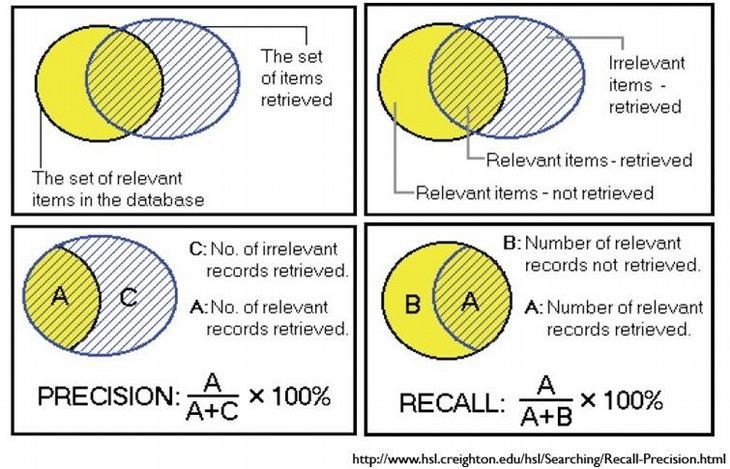
### Evaluating Classification: Precision and Recall

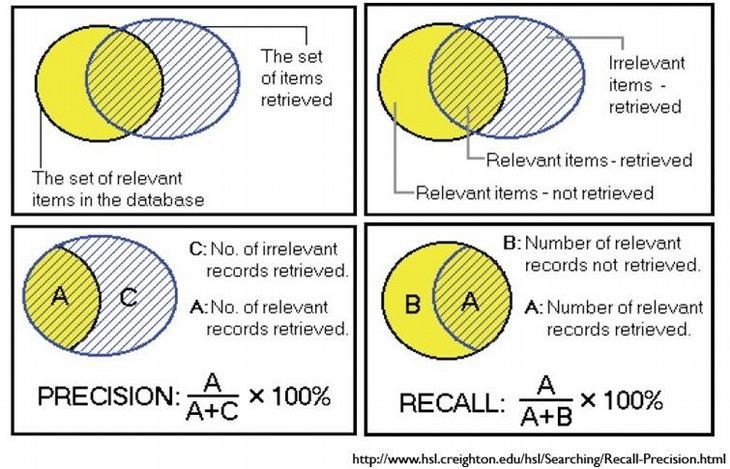
Image:https://commons.wikimedia.org/wiki/File:Precisionrecall.svg

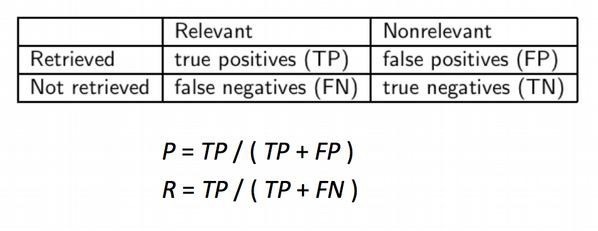


**Breaking it down: intersection**

**vs. remaining parts**





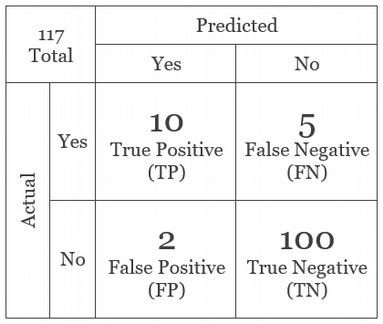


Source: Paula Velardi





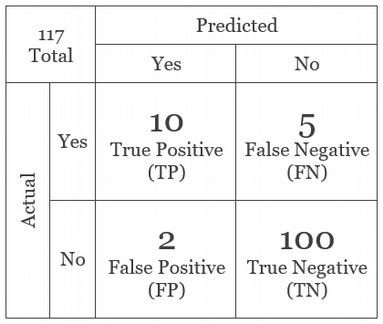
**What is the accuracy?**

<http://technopreneurlife.com/primer-on-binary-classification/>





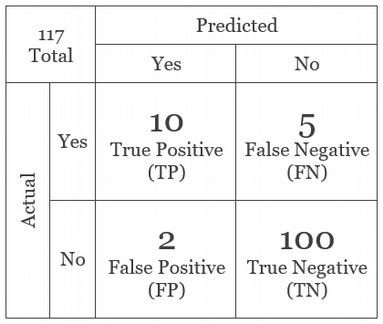
**What is the precision?**

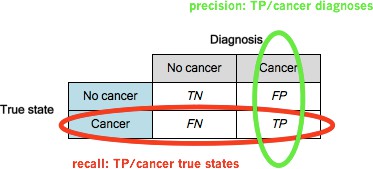
<http://technopreneurlife.com/primer-on-binary-classification/>





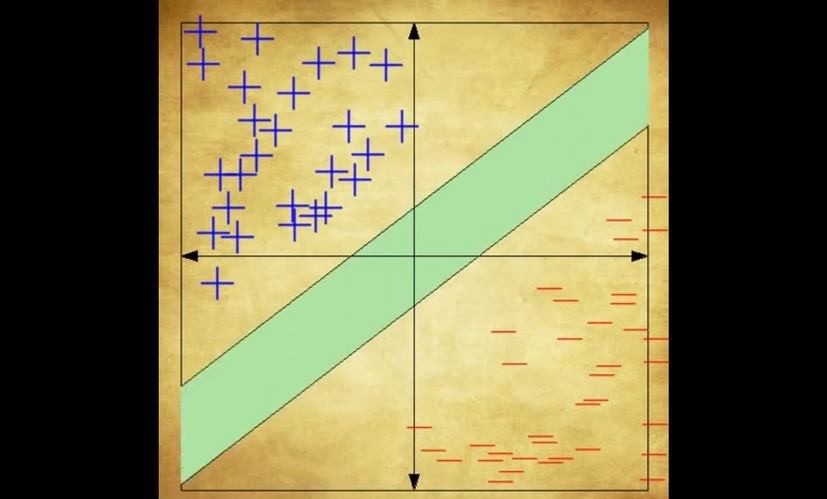
**What is the recall?**

<http://technopreneurlife.com/primer-on-binary-classification/>

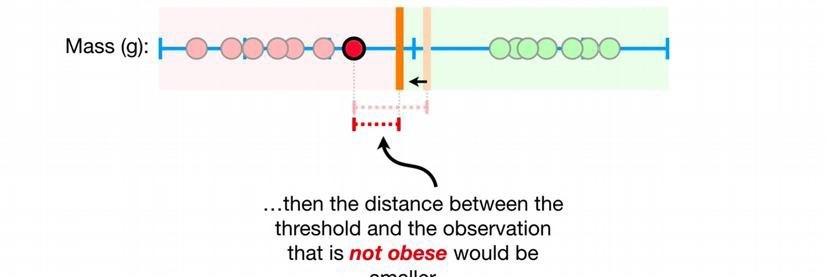


**Why is recall not enough?**

[www.quora.com/What-is-the-best-way-to-understand-the-terms-precision-and-recall/answer/Joel-Chan](http://www.quora.com/What-is-the-best-way-to-understand-the-terms-precision-and-recall/answer/Joel-Chan)



https://youtu.be/5zRmhOUjjGY



Lowering the threshold means more items will be classified as positive.

Score

Source: Adapted from StatQuest with Josh Starmer. https://youtu.be/efR1C6CvhmE

### Precision and Recall:

Trade-Off

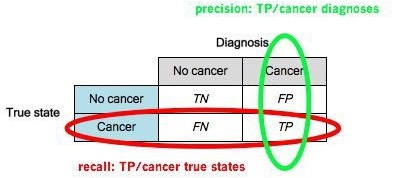
**We can easily get perfect recall: Total Recall, i.e.**



**just classify all as positive**

Total Recall (1990 movie)

### Evaluating Classification: Precision and Recall



**Why is precision not enough?**

[www.quora.com/What-is-the-best-way-to-understand-the-terms-precision-and-recall/answer/Joel-Chan](http://www.quora.com/What-is-the-best-way-to-understand-the-terms-precision-and-recall/answer/Joel-Chan)

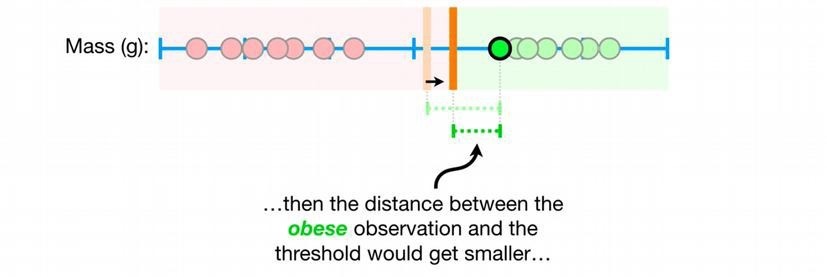
### Precision and Recall: How to Change it?

Score

If we keep increasing the threshold, we will eventually only accept those items that have extremely high goodness scores

**We can often get great precision by only accepting those we are**

**really really sure**



**about**

**So then why don't we just always do this?**

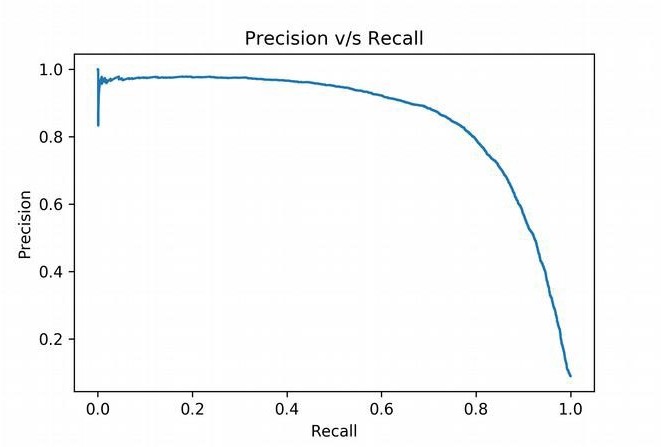
There will be many items that we miss (false negatives) We will lose a lot of recall.

Out of those, often a large percentage is correct.

Source: Adapted from StatQuest with Josh Starmer. https://youtu.be/efR1C6CvhmE

### Precision and Recall:

Trade-Off



**We can often get great precision by only accepting those we are**

**really really sure**

**about**

[www.sanyamkapoor.com/machine-learning/confusion-matrix-visualization/](http://www.sanyamkapoor.com/machine-learning/confusion-matrix-visualization/)

### Evaluating Classification: F1 score

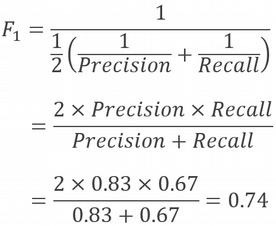
**What if we want**

**to get a single number to evaluate both precision and recall?**

<http://technopreneurlife.com/primer-on-binary-classification/>

### Evaluating Classification: F1

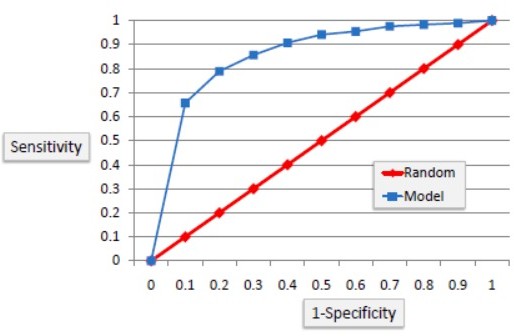
score



**Harmonic mean of precision and recall**

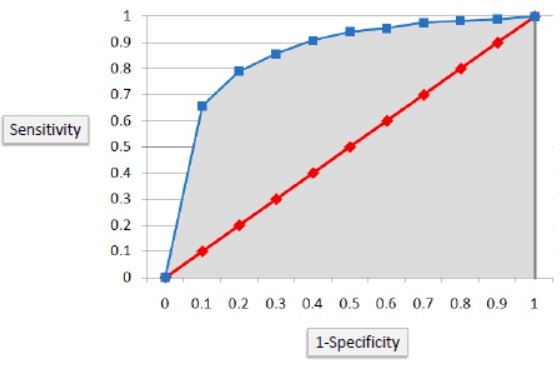
<http://technopreneurlife.com/primer-on-binary-classification/>

## ROC Chart

* The ROC chart shows false positive rate (1-specificity) on X-axis, the probability of target=1 when its true value is 0, against true positive rate (sensitivity) on Y-axis, the probability of target=1 when its true value is 1. Ideally, the curve will climb quickly toward the top-left meaning the model correctly predicted the cases. The diagonal red line is for a random model 6543tgrepcheck2&&\*\*(11

https://[www.saedsayad.com/model\_evaluation.htm](http://www.saedsayad.com/model_evaluation.htm)

## Area Under the Curve (AUC)

Area under ROC curve is often used as a measure of quality of the classification models. A random classifier has an area under the curve of 0.5, while AUC for a perfect classifier is equal to 1. In practice, most of the classification models have an AUC between 0.5 and 1.

An area under the ROC curve of 0.8, for example, means that a randomly selected case from the group with the target equals 1 has a score larger than that for a randomly chosen case from the group with the target equals 0 in 80% of the time. When a classifier cannot distinguish between the two groups, the area will be equal to 0.5 (the ROC curve will coincide with the diagonal). When there is a perfect separation of the two groups, i.e., no overlapping of the distributions, the area under the ROC curve reaches to 1 (the ROC curve will reach the upper left corner of the plot).

# Further Analysis

### Learning Behavior

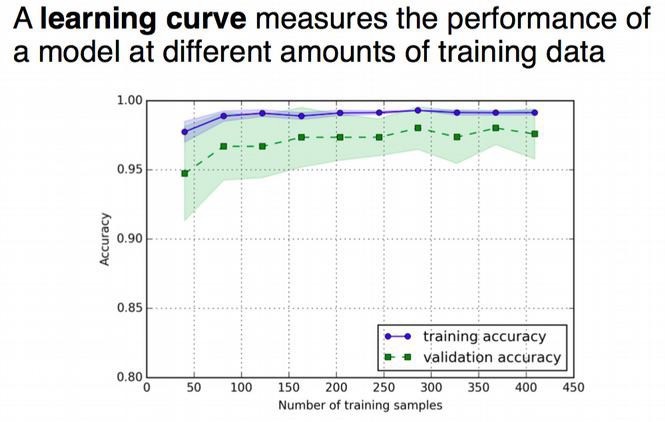


Image: Michael Paul, University of Colorado Boulder