Data Mining 2

Topic 03 : Review of Model Building

Lecture 03 : Classification Models

Dr Kieran Murphy

Department of Computing and Mathematics, WIT.
(kmurphy@wit.ie)

Spring Semester, 2022

Outline

- Confusion matrix
- Precision, Accuracy, Recall and Specificity
- ROC and AUC

Outline

A Non-perfect Test — Type I and Type II Errors

Consider an imperfect test with two outcomes, there are four possible outcomes:

A Non-perfect Test — Type I and Type II Errors

Consider an imperfect test with two outcomes, there are four possible outcomes:

		Predicted		
		Negative	Positive	
Actual	Negative	✔ True Negative (TN)		N
	Positive		✔ True Positive (TP)	P
		$\hat{N}$	$\hat{P}$	T

A Non-perfect Test — Type I and Type II Errors

Consider an imperfect test with two outcomes, there are four possible outcomes:

Confusion Matrix

		Predicted: Negative	Predicted: Positive	
Actual	Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
Actual	Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
		$\hat{N}$	$\hat{P}$	T

A Non-perfect Test — Type I and Type II Errors

Consider an imperfect test with two outcomes, there are four possible outcomes:

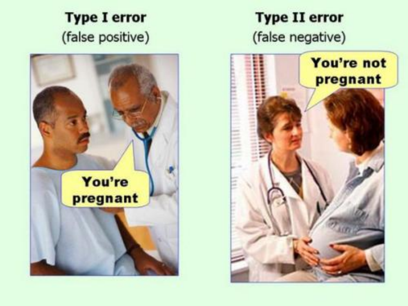
Confusion Matrix

		Predicted: Negative	Predicted: Positive	
Actual	Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
	Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
		$\hat{N}$	$\hat{P}$	T

- If the test is applied to $T = P + N = \hat{P} + \hat{N}$ observations / subjects / instances then we have four independent quantities TP, TN, FP, and FN.
- How do we combines these quantities into a single metric ?
- The fraction of correct results seems like a good idea

$$\text{accuracy} = \frac{TP + TN}{P + N}$$

But what happens, if we are testing for an rare event? Maximising accuracy will result in the test always returning negative.



- Ideally we want the probability of either error to be zero but that may not be possible.
- Depending on the conditions we often modify the test to reduce probability of the type of error we don't want at the expense of increasing the probability of the other — think court case vs medical condition.

A Non-perfect Test — Type I and Type II Errors

Consider an imperfect test with two outcomes, there are four possible outcomes:

Confusion Matrix

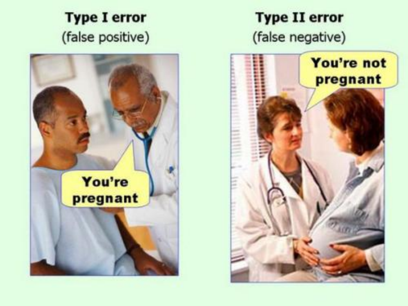
Predicted

Actual	Negative	Positive	
Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
	$\hat{N}$	$\hat{P}$	T

- If the test is applied to $T = P + N = \hat{P} + \hat{N}$ observations / subjects / instances then we have four independent quantities TP, TN, FP, and FN.
- How do we combines these quantities into a single metric ?
- The fraction of correct results seems like a good idea

$$\text{accuracy} = \frac{TP + TN}{P + N}$$

But what happens, if we are testing for an rare event? Maximising accuracy will result in the test always returning negative.



- Ideally we want the probability of either error to be zero but that may not be possible.
- Depending on the conditions we often modify the test to reduce probability of the type of error we don't want at the expense of increasing the probability of the other — think court case vs medical condition.

A Non-perfect Test — Type I and Type II Errors

Consider an imperfect test with two outcomes, there are four possible outcomes:

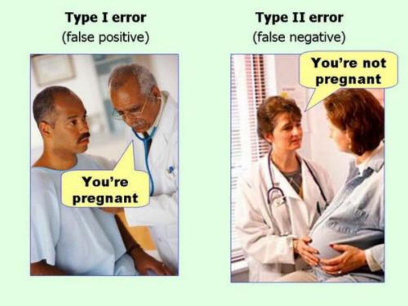
Confusion Matrix

		Predicted: Negative	Predicted: Positive	
Actual	Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
	Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
		$\hat{N}$	$\hat{P}$	T

- If the test is applied to $T = P + N = \hat{P} + \hat{N}$ observations / subjects / instances then we have four independent quantities TP, TN, FP, and FN.
- How do we combines these quantities into a single metric ?
- The fraction of correct results seems like a good idea

$$\text{accuracy} = \frac{TP + TN}{P + N}$$

But what happens, if we are testing for an rare event? Maximising accuracy will result in the test always returning negative.



- Ideally we want the probability of either error to be zero but that may not be possible.
- Depending on the conditions we often modify the test to reduce probability of the type of error we don't want at the expense of increasing the probability of the other — think court case vs medical condition.

A Non-perfect Test — Type I and Type II Errors

Consider an imperfect test with two outcomes, there are four possible outcomes:

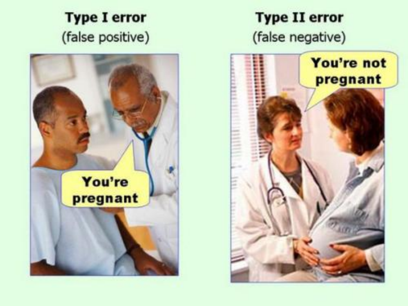
Confusion Matrix

		Predicted: Negative	Predicted: Positive	
Actual	Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
Actual	Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
		$\hat{N}$	$\hat{P}$	T

- If the test is applied to $T = P + N = \hat{P} + \hat{N}$ observations / subjects / instances then we have four independent quantities TP, TN, FP, and FN.
- How do we combines these quantities into a single metric ?
- The fraction of correct results seems like a good idea

$$\text{accuracy} = \frac{TP + TN}{P + N}$$

But what happens, if we are testing for an rare event? Maximising accuracy will result in the test always returning negative.



- Ideally we want the probability of either error to be zero but that may not be possible.
- Depending on the conditions we often modify the test to reduce probability of the type of error we don't want at the expense of increasing the probability of the other — think court case vs medical condition.

Confusion matrix (Contingency table) Metrics I

$$\textbf{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{P} + \text{N}}$$

(How often is the classifier correct?)

- False negative rate (FNR) $= \frac{FN}{\text{P}} = 1 - TPR$
- **Sensitivity** = **Recall** = True positive rate (TPR) $= \frac{TP}{\text{P}} = 1 - FNR$
  (Of positive cases that exist how many did we mark positive?)
- **Specificity** $= \frac{\text{TN}}{\text{N}} = 1 - \text{FPR}$
  (When it's actually no, how often does we predict no?)
  (Of cases that are negative, how many did we mark negative?)
- False positive rate (FPR) = false acceptance $= \frac{FP}{\text{N}} = 1 - \text{Specificity}$
- **Precision** = positive predictive value (PPV) $= \frac{TP}{\hat{\text{P}}} = \frac{TP}{\text{TP} + \text{FP}}$
  (Of cases that we marked positive, how many were correct?)

		Predicted: Negative	Predicted: Positive	
Actual	Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
Actual	Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
		$\hat{N}$	$\hat{P}$	T

Confusion matrix (Contingency table) Metrics I

		Predicted: Negative	Predicted: Positive	
Actual	Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
	Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
		$\hat{N}$	$\hat{P}$	T

$$\textbf{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{P} + \text{N}}$$

(How often is the classifier correct?)

- False negative rate (FNR) $= \dfrac{FN}{\text{P}} = 1 - TPR$
- **Sensitivity** = **Recall** = True positive rate (TPR) $= \dfrac{TP}{\text{P}} = 1 - FNR$
  (Of positive cases that exist how many did we mark positive?)
- **Specificity** $= \dfrac{\text{TN}}{\text{N}} = 1 - \text{FPR}$
  (When it's actually no, how often does we predict no?)
  (Of cases that are negative, how many did we mark negative?)
- False positive rate (FPR) = false acceptance $= \dfrac{FP}{\text{N}} = 1 - \text{Specificity}$
- **Precision** = positive predictive value (PPV) $= \dfrac{TP}{\hat{\text{P}}} = \dfrac{TP}{\text{TP} + \text{FP}}$
  (Of cases that we marked positive, how many were correct?)

Confusion matrix (Contingency table) Metrics I

$$\textbf{Accuracy} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{P} + \mathrm{N}}$$

(How often is the classifier correct?)

Actual \ Predicted	Negative	Positive	
Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
	$\hat{N}$	$\hat{P}$	T

- False negative rate (FNR) $= \frac{FN}{\mathrm{P}} = 1 - TPR$
- **Sensitivity** = **Recall** = True positive rate (TPR) $= \frac{TP}{\mathrm{P}} = 1 - FNR$
  (Of positive cases that exist how many did we mark positive?)
- **Specificity** $= \frac{\mathrm{TN}}{\mathrm{N}} = 1 - \mathrm{FPR}$
  (When it's actually no, how often does we predict no?)
  (Of cases that are negative, how many did we mark negative?)
- False positive rate (FPR) = false acceptance $= \frac{FP}{\mathrm{N}} = 1 - \mathrm{Specificity}$
- **Precision** = positive predictive value (PPV) $= \frac{TP}{\hat{\mathrm{P}}} = \frac{TP}{\mathrm{TP} + \mathrm{FP}}$
  (Of cases that we marked positive, how many were correct?)

Confusion matrix (Contingency table) Metrics I

		Predicted: Negative	Predicted: Positive	
Actual	Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
	Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
		$\hat{N}$	$\hat{P}$	T

Accuracy $= \dfrac{\text{TP} + \text{TN}}{\text{P} + \text{N}}$

(How often is the classifier correct?)

- False negative rate (FNR) $= \dfrac{FN}{\text{P}} = 1 - TPR$
- **Sensitivity** = **Recall** = True positive rate (TPR) $= \dfrac{TP}{\text{P}} = 1 - FNR$

  (Of positive cases that exist how many did we mark positive?)
- **Specificity** $= \dfrac{\text{TN}}{\text{N}} = 1 - \text{FPR}$

  (When it's actually no, how often does we predict no?)

  (Of cases that are negative, how many did we mark negative?)
- False positive rate (FPR) = false acceptance $= \dfrac{FP}{\text{N}} = 1 - \text{Specificity}$
- **Precision** = positive predictive value (PPV) $= \dfrac{TP}{\hat{\text{P}}} = \dfrac{TP}{\text{TP} + \text{FP}}$

  (Of cases that we marked positive, how many were correct?)

Confusion matrix (Contingency table) Metrics I

		Predicted: Negative	Predicted: Positive	
Actual	Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
	Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
		$\hat{N}$	$\hat{P}$	T

Accuracy $= \dfrac{\text{TP} + \text{TN}}{\text{P} + \text{N}}$
(How often is the classifier correct?)

- False negative rate (FNR) $= \dfrac{FN}{\text{P}} = 1 - TPR$
- **Sensitivity** = **Recall** = True positive rate (TPR) $= \dfrac{TP}{\text{P}} = 1 - FNR$
  (Of positive cases that exist how many did we mark positive?)
- **Specificity** $= \dfrac{\text{TN}}{\text{N}} = 1 - \text{FPR}$
  (When it's actually no, how often does we predict no?)
  (Of cases that are negative, how many did we mark negative?)
- False positive rate (FPR) = false acceptance $= \dfrac{FP}{\text{N}} = 1 - \text{Specificity}$
- **Precision** = positive predictive value (PPV) $= \dfrac{TP}{\hat{\text{P}}} = \dfrac{TP}{\text{TP} + \text{FP}}$
  (Of cases that we marked positive, how many were correct?)

Confusion matrix (Contingency table) Metrics I

		Predicted: Negative	Predicted: Positive	
Actual	Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
	Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
		$\hat{N}$	$\hat{P}$	T

Accuracy $= \dfrac{\mathrm{TP} + \mathrm{TN}}{\mathrm{P} + \mathrm{N}}$
(How often is the classifier correct?)

- False negative rate (FNR) $= \dfrac{FN}{\mathrm{P}} = 1 - TPR$
- **Sensitivity** = **Recall** = True positive rate (TPR) $= \dfrac{TP}{\mathrm{P}} = 1 - FNR$
  (Of positive cases that exist how many did we mark positive?)
- **Specificity** $= \dfrac{\mathrm{TN}}{\mathrm{N}} = 1 - \mathrm{FPR}$
  (When it's actually no, how often does we predict no?)
  (Of cases that are negative, how many did we mark negative?)
- False positive rate (FPR) = false acceptance $= \dfrac{FP}{\mathrm{N}} = 1 - \mathrm{Specificity}$
- **Precision** = positive predictive value (PPV) $= \dfrac{TP}{\hat{\mathrm{P}}} = \dfrac{TP}{\mathrm{TP} + \mathrm{FP}}$
  (Of cases that we marked positive, how many were correct?)

Confusion matrix (Contingency table) Metrics I

Accuracy — how well model is trained and perform in general

$$\textbf{Accuracy} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{P}+\mathrm{N}}$$

(How often is the classifier correct?)

		Predicted: Negative	Predicted: Positive	
Actual	Negative	✔ True Negative (TN)	Type I error False Positive (FP)	N
	Positive	Type II error False Negative (FN)	✔ True Positive (TP)	P
		$\hat{\mathrm{N}}$	$\hat{\mathrm{P}}$	T

- False negative rate (FNR) $= \dfrac{FN}{\mathrm{P}} = 1 - TPR$
- **Sensitivity** = **Recall** = True positive rate (TPR) $= \dfrac{TP}{\mathrm{P}} = 1 - FNR$
  (Of positive cases that exist how many did we mark positive?)
- **Specificity** $= \dfrac{\mathrm{TN}}{\mathrm{N}} = 1 - \mathrm{FPR}$
  (When it's actually no, how often does we predict no?)
  (Of cases that are negative, how many did we mark negative?)
- False positive rate (FPR) = false acceptance $= \dfrac{FP}{\mathrm{N}} = 1 - \mathrm{Specificity}$
- **Precision** = positive predictive value (PPV) $= \dfrac{TP}{\hat{\mathrm{P}}} = \dfrac{TP}{\mathrm{TP}+\mathrm{FP}}$
  (Of cases that we marked positive, how many were correct?)

Recall — important when the costs of false negatives are high

Precision — important when the costs of false positives are high

Confusion matrix (Contingency table) Metrics II

F_1 Score

The F-measure or balanced F-score (F_1 score) is the harmonic mean of precision and recall:

$$F_1 = 2\left[\frac{1}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}}\right] = 2\left[\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}\right]$$

A word of Caution . . .

Consider the three binary classifiers A, B and C

	A		B		C	
	T	F	T	F	T	F
T	0.9	0.1	0.8	0	0.78	0
F	0	0	0.1	0.1	0.12	0.1

Confusion matrix (Contingency table) Metrics II

F_1 Score

The F-measure or balanced F-score (F_1 score) is the harmonic mean of precision and recall:

$$F_1 = 2\left[\frac{1}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}}\right] = 2\left[\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}\right]$$

A word of Caution . . .

Consider the three binary classifiers A, B and C

	A		B		C	
	T	F	T	F	T	F
T	0.9	0.1	0.8	0	0.78	0
F	0	0	0.1	0.1	0.12	0.1

Confusion matrix (Contingency table) Metrics II

F_1 Score

The F-measure or balanced F-score (F_1 score) is the harmonic mean of precision and recall:

$$F_1 = 2\left[\frac{1}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}}\right] = 2\left[\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}\right]$$

A word of Caution ...

Consider the three binary classifiers A, B and C

	A		B		C	
	T	F	T	F	T	F
T	0.9	0.1	0.8	0	0.78	0
F	0	0	0.1	0.1	0.12	0.1

Clearly classifier A is useless since it always predicts label T regardless of the input. Also, B is slightly better than C (lower off- diagonal total).

Confusion matrix (Contingency table) Metrics II

F_1 Score

The F-measure or balanced F-score (F_1 score) is the harmonic mean of precision and recall:

$$F_1 = 2\left[\frac{1}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}}\right] = 2\left[\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}\right]$$

A word of Caution ...

Consider the three binary classifiers A, B and C

	A		B		C	
	T	F	T	F	T	F
T	0.9	0.1	0.8	0	0.78	0
F	0	0	0.1	0.1	0.12	0.1

Metric	**A**	**B**	**C**	(best)
Accuracy	0.9	0.9	0.88	**AB**
Precision	0.9	1.0	1.0	**BC**
Recall	1.0	0.888	0.8667	**A**
F-score	0.947	0.941	0.9286	**A**

Clearly classifier A is useless since it always predicts label T regardless of the input. Also, B is slightly better than C (lower off- diagonal total).
Yet look at the performance metrics – B is never the clear winner.

Confusion matrix (Contingency table) Metrics II

F_1 Score

The F-measure or balanced F-score (F_1 score) is the harmonic mean of precision and recall:

$$F_1 = 2\left[\frac{1}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}}\right] = 2\left[\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}\right]$$

A word of Caution . . .

Consider the three binary classifiers A, B and C

	A		B		C	
	T	F	T	F	T	F
T	0.9	0.1	0.8	0	0.78	0
F	0	0	0.1	0.1	0.12	0.1

Metric	**A**	**B**	**C**	(best)
Accuracy	0.9	0.9	0.88	**AB**
Precision	0.9	1.0	1.0	**BC**
Recall	1.0	0.888	0.8667	**A**
F-score	0.947	0.941	0.9286	**A**

Clearly classifier *A* is useless since it always predicts label T regardless of the input. Also, B is slightly better than C (lower off- diagonal total).

Yet look at the performance metrics – B is never the clear winner.

We use some metrics because they are easy to understand, and not because they always give the "correct" result.

Mutual Information is a Better Metric

The mutual information between predicted and actual label (case) is defined

$$I(\hat{y}, y) = \sum_{\hat{y}=\{0,1\}} \sum_{y=\{0,1\}} p(\hat{y}, y) \log \frac{p(\hat{y}, y)}{p(\hat{y})p(y)}$$

where $p(\hat{y}, y)$ is the joint probability distribution function.
This gives the intuitively correct rankings $B > C > A$

Metric	**A**	**B**	**C**
Accuracy	0.9	0.9	0.88
Precision	0.9	1.0	1.0
Recall	1.0	0.888	0.8667
F-score	0.947	0.941	0.9286
Mutual information	**0**	**0.1865**	**0.1735**

Receiver Operating Characteristic (ROC) I

Many classification systems have thresholds and parameters that can vary their performance*. In such cases, it can be useful to look at the TPR and FPR as the threshold/parameters are changed. One can seek to determine the best possible values for the threshold/parameters by finding a particular TPR vs FPR ratio. This is done by plotting an ROC curve.

For example, consider the Titanic survivor dataset, a part of the output from a classifier is given bellow:

Output from classifier: class probabilities

	Survived	Died
15	0.092	0.908
16	0.904	0.096
17	0.646	0.354
18	0.740	0.260
19	0.460	0.540

Sorted probabilities

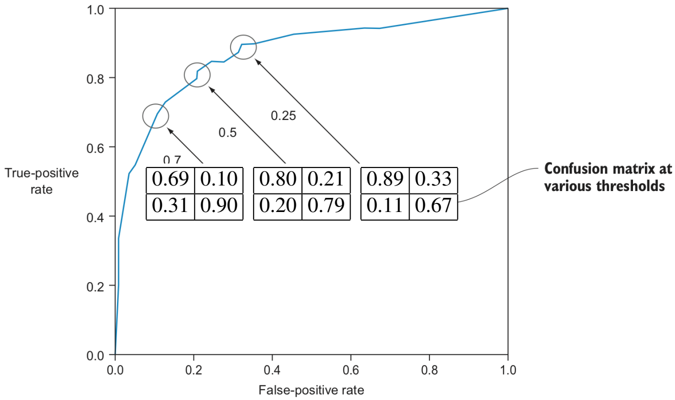
	Survived	Died
308	0.705	0.295
215	0.703	0.297
217	0.700	0.300
54	0.698	0.302
169	0.698	0.302

Threshold: "survived" probabilities > 0.7

After sorting the full table by decreasing survival probability, we can set a threshold and consider all rows above this threshold as survived.

*Think logistic regression – at what probability value do we switch from predicting 0 to predicting 1?

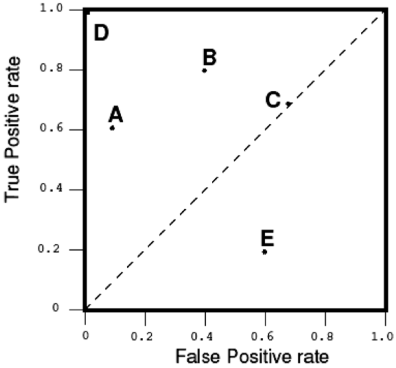
Receiver Operating Characteristic (ROC) II



The ROC curve defined by calculating the confusion matrix and ROC metrics at various threshold points from 0 to 1. By convention, we plot the false-positive rate on the x-axis and the true-positive rate on the y-axis.

ROC Space

- Each point in ROC represents a classifier.
- Lower left point $(0, 0)$ represents the strategy of never issuing a positive classification: such a classier commits no false positive errors but also gains no true positives
- Upper right corner $(1, 1)$ represents the opposite strategy, of unconditionally issuing positive classifications.
- Point $(0, 1)$ represents perfect classification D's performance is perfect as shown.
- Informally, one point in ROC space is better than another if it is to the northwest of the first: TPR is higher, FPR is lower, or both.

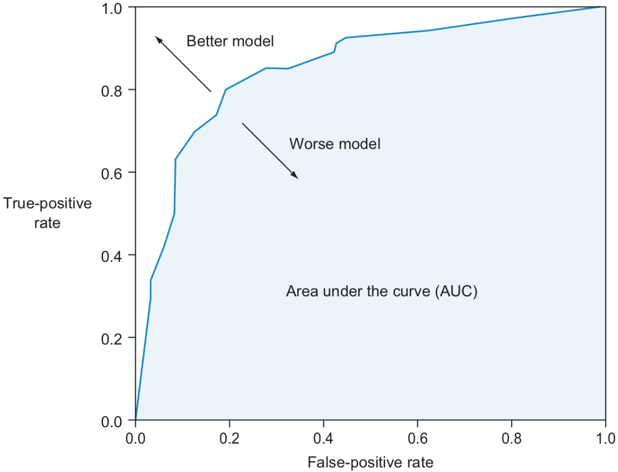


An ROC graph with five classifiers: A,. . . , E.

- The diagonal line $y = x$ represents the strategy of randomly guessing a class.

Area Under an ROC Curve (AUC)

Rather than looking at individual points (based on particular threshold values) we can compare classifiers by using the area under the ROC.



The AUC of a classifier is equivalent to the probability that the classier will rank a randomly chosen positive instance higher than a randomly chosen negative instance.

The bigger AUC the better

Micro Average vs Macro Average Performance

In a multi-class classifier we have more than two classes (iris dataset):

- The ROC is computed for each class using the one-vs-all — for each class, we denote the particular class as the positive class, and everything else as the negative class, and we draw the ROC curves as usual.
- To combine the metrics for individual classes to get an overall system metrics, we can apply either

  Micro-Average Method
  : Sum up the individual true positives, false positives, and false negatives of the system for different classes and then apply totals to get the statistics.

  Macro-average Method
  : Average the precision and recall of the system on different classes.

See `classification_report` from `sklearn.metrics`.