Werkblad week 3

d.r.m.langers@pl.hanze.nl

7 oktober 2021

1 De confusion matrix

De prestaties van een classificatie-algoritme worden vaak samengevat door middel van een *confusion matrix*. Deze bestaat uit een kruistabel die weergeeft hoe de instances van elke klasse geclassificeerd worden. Wanneer we ons beperken tot twee klassen ontstaat een 2×2 matrix. Meestal kunnen we één van de klasse-labels zien als "positief" en de andere als "negatief". Bijvoorbeeld, patiënten testen positief op de aanwezigheid van een ziekte, of watermonsters testen positief op de aanwezigheid van toxische stoffen. Als niet duidelijk is welke klassen als positief of als negatief gezien moeten worden dient dit expliciet gespecificeerd te worden (bijvoorbeeld wanneer de klasse een geslacht betreft).

Zetten we alle mogelijke combinaties tegen elkaar uit, dan zijn er vier categorieën instances:

- *True positives* (TP): instances die positief geclassificeerd worden, en ook daadwerkelijk positief zijn;
- *True negatives* (TN): instances die negatief geclassificeerd worden, en ook daadwerkelijk negatief zijn;
- *False positives* (FP): instances die weliswaar positief geclassificeerd worden, maar eigenlijk negatief zijn;
- *False negatives* (FN): instances die weliswaar negatief geclassificeerd worden, maar eigenlijk positief zijn.

False positives worden overigens ook wel *type-I errors* genoemd; false negatives heten *type-II errors*.

De confusion matrix ziet er nu als volgt uit.

		Toegekende klasse:	
		Positief	Negatief
Ware klasse:	Positief	TP	FN
	Negatief	FP	TN

De accuracy van een algoritme, dat wil zeggen nauwkeurigheid, wordt gekwantificeerd als de fractie juist geclassificeerde instances binnen alle instances.

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP} + \mathrm{TN}}$$

Complementair daaraan is de error rate, dat wil zeggen foutfrequentie, gelijk aan de fractie onjuist geclassificeerde instances binnen alle instances.

$$\mathrm{ERR} = \frac{\mathrm{FN} + \mathrm{FP}}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP} + \mathrm{TN}}$$

Vergelijkbare prestatiematen kunnen ook toegepast worden op de individuele rijen of kolommen van de confusion matrix.

- Kijken we slechts naar de bovenste rij met ware positieven, dan verkrijgen we als nauwkeurigheidsmaat de *True Positive Rate*, $\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP+FN}}$, en als foutenmaat de *False Negative Rate*, $\mathrm{FNR} = \frac{\mathrm{FN}}{\mathrm{TP+FN}}$;
- Kijken we slechts naar de onderste rij met ware negatieven, dan verkrijgen we als nauwkeurigheidsmaat de *True Negative Rate*, $\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{FP+TN}}$, en als foutenmaat de *False Positive Rate*, $\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP+TN}}$;
- Kijken we slechts naar de linker kolom met voorspelde positieven, dan verkrijgen we als nauwkeurigheidsmaat de *Positive Predictive Value*, $\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP+FP}}$, en als foutenmaat de *False Discovery Rate*, $\mathrm{FDR} = \frac{\mathrm{FP}}{\mathrm{TP+FP}}$;
- Kijken we slechts naar de rechter kolom met voorspelde negatieven, dan verkrijgen we als nauwkeurigheidsmaat de *Negative Predictive Value*, $\mathrm{NPV} = \frac{\mathrm{TN}}{\mathrm{FN+TN}}$, en als foutenmaat de *False Omission Rate*, $\mathrm{FOR} = \frac{\mathrm{FN}}{\mathrm{FN+TN}}$.

In alle gevallen tellen de overeenkomstige nauwkeurigheidsmaat en foutenmaat op tot 100%, dus deze geven in essentie hetzelfde soort informatie. De TPR en TNR worden respectievelijk ook wel *sensitivity* en *specificity* genoemd, en de TPR en PPV heten respectievelijk ook wel *recall* en *precision*.

Opgave 1. Wanneer met Weka de weather-numeric dataset wordt geclassificeerd door de J48-Tree classifier met leave-one-out kruis-validatie, dan wordt de volgende confusion matrix gerapporteerd:

```
a b   <-- classified as
7 2 | a = yes
3 2 | b = no
```

Bereken de behaalde ACC, ERR, FPR, TPR, FNR, TNR, PPV, NPV, FDR en FOR.

Opgave 2. Je hebt een zelftest gedaan op de aanwezigheid van een virus en je test positief. Nu wil je weten hoe groot de kans is dat je desondanks toch niet werkelijk besmet bent. Welke prestatiemaat zegt hier iets over?

Opgave 3. Welke prestatiemaat zou je kunnen schrijven als de onvoorwaardelijke kans P(voorspelde klasse=ware klasse)? En welke prestatiemaat zou je kunnen schrijven als de voorwaardelijke kans P(voorspelde klasse=negatief | ware klasse=positief)? Tenslotte, hoe zou je de Precision kunnen schrijven als een voorwaardelijke kans?

Wanneer er drie onafhankelijke prestatiematen bekend zijn, kunnen alle andere worden berekend. Het eenvoudigst kan dit worden bereikt door één aantal in de confusion matrix arbitrair te kiezen, en dan de drie overige aantallen één voor één aan te vullen op grond van de drie gegevens.

Stel bijvoorbeeld dat bekend is dat de recall gelijk is aan 90%, de precision gelijk is aan 75%, en de FPR gelijk is aan 20%. Bepalen we nu alle nauwkeurigheids- en foutenmaten.

De recall is gelijk aan de TPR, en wil deze 90% zijn dan moeten dus 90% van alle ware positieven ook gelabeld worden als positieven. Stellen we het aantal positieven op honderd dan hebben we 90 TPs en 10 FNs. Vervolgens kunnen we gebruik maken van het feit dat de precision, oftewel de PPV, 75% is. Dat wil zeggen dat 75% van de instances die positief gelabeld worden daadwerkelijk positief zijn. Dit levert op $\mathrm{TP} = 0.75 \cdot (\mathrm{TP} + \mathrm{FP})$ wat vereenvoudigt tot $\mathrm{TP} = 3 \cdot \mathrm{FP}$. Omdat we 90 TPs hadden kunnen we hier eenvoudig uit afleiden dat er 30 FPs moeten zijn. Om een FPR van 20% te bereiken moeten deze 30 FPs aangevuld worden met 120 TNs, en daarmee is de confusion matrix compleet.

		Toegekende klasse:	
		Positief	Negatief
Ware klasse:	Positief	90	10
	Negatief	30	120

Hieruit kunnen alle uitkomsten worden berekend: ACC $= \frac{210}{250} = 0.84$; ERR $= \frac{40}{250} = 0.16$; FPR $= \frac{30}{150} = 0.20$; TPR $= \frac{90}{100} = 0.90$; FNR $= \frac{10}{100} = 0.10$; TNR $= \frac{120}{150} = 0.80$; PPV $= \frac{90}{120} = 0.75$; NPV $= \frac{120}{130} = 0.923$; FDR $= \frac{30}{120} = 0.25$; FOR $= \frac{10}{130} = 0.077$. Merk op dat het mogelijk is dat er gedurende deze procedure niet-gehele aantallen in de confusion matrix verschijnen. Dit is geen probleem; de aantallen kunnen desgewenst worden herschaald om toch op gehele getallen uit te komen, maar voor de uitkomsten maakt dit niet uit.

Opgave 4. Een classificatieprobleem levert als uitkomst TPR = 0.60, TNR = 0.80, en FDR = 0.20. Leid uit deze gegevens de behaalde nauwkeurigheid ACC af.

Opgave 5. Een algoritme behaalt een False Negative Rate van 5%, een False Discovery Rate van 10%, en een error rate van $12\frac{1}{2}$%. Bereken hiermee de False Positive Rate.

Voor datasets met meer dan twee klassen kunnen soortgelijke berekeningen worden uitgevoerd, hoewel dan wel moet worden gespecificeerd welke klasse als positief wordt behandeld. Bijvoorbeeld, in de iris dataset zijn er drie klassen: *Iris setosa*, *Iris versicolor*, en *Iris virginica*. Het is nu mogelijk om bijvoorbeeld de TPR voor *Iris setosa* te bepalen, door te berekenen welke fractie van alle *Iris setosa* bloemen ook daadwerkelijk als *Iris setosa* gelabeld wordt; soortgelijk is de PPV voor *Iris versicolor* gelijk aan de fractie van alle als *Iris versicolor* gelabelde bloemen die ook daadwerkelijk *Iris versicolor* zijn. De genoemde soort telt telkens als de positieve en de andere twee fungeren samen als negatieven.

Opgave 6. Wanneer met Weka de iris dataset wordt geclassificeerd door de One-R classifier met 10-fold cross-validation, dan wordt de volgende confusion matrix gerapporteerd:

```
 a  b  c   <-- classified as
50  0  0 |  a = Iris-setosa
 0 45  5 |  b = Iris-versicolor
 0  7 43 |  c = Iris-virginica
```

Bereken de behaalde Precision en Recall voor de klassen *Iris setosa*, *Iris versicolor*, en *Iris virginica*.

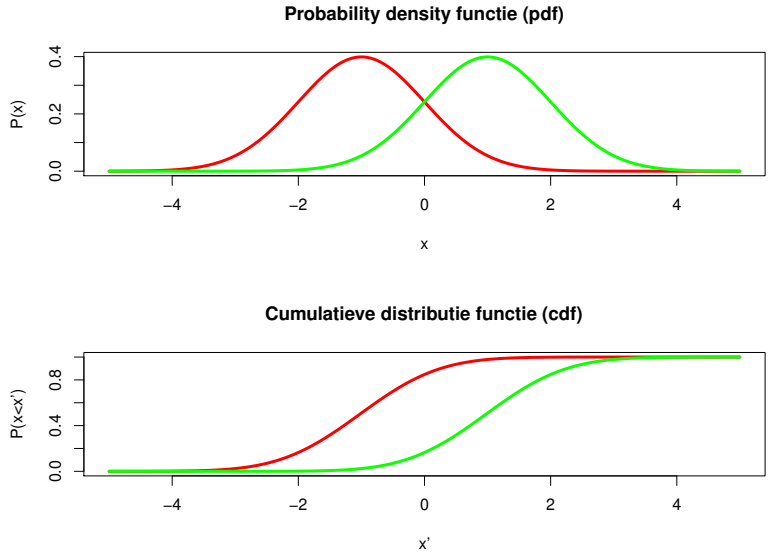
2 De ROC-curve

De genoemde uitkomstmaten geven in hun eentje nog niet per se nuttige informatie omtrent hoe goed een algoritme presteert. Het is bijvoorbeeld eenvoudig om een perfecte TPR te behalen, namelijk door alle instances als positieven te classificeren: dan worden alle instances die daadwerkelijk positief zijn inderdaad ook juist geclassificeerd. Dit gaat ten koste van de TNR, want de instances die eigenlijk negatief zijn worden dan allemaal onjuist geclassificeerd. Omgekeerd kan ook een perfecte TNR worden bereikt door alle instances negatief te labelen, maar dit gaat ten koste van de TPR. Op deze manier kan de ene nauwkeurigheid tegen de andere worden uitgewisseld.

Voor de PPV en NPV geldt iets soortgelijks: als een algoritme alleen de instances waarvoor het absoluut zeker is dat ze positief zijn labelt als positief, dan zal het een goede PPV kunnen behalen (maar een slechte NPV); als een algoritme daarentegen alleen de instances waarvoor het absoluut zeker is dat ze negatief zijn labelt als negatief, dan zal het een goede NPV kunnen behalen (ten koste van de PPV).

Om beter inzicht te krijgen in de prestaties van een algoritme is het gebruikelijk om de ene prestatiemaat uit te zetten tegen de andere. De meest gebruikte weergave is daarbij de *Receiver Operating Characteristic* curve, de *ROC-curve*, waarbij de TPR langs de y-as wordt uitgezet als functie van de FPR langs de x-as.

Laten we als voorbeeld eens een tweetal klassen beschouwen waarvoor één numeriek attribuut x gemeten is dat voor beide klassen normaal verdeeld is. Denk bijvoorbeeld aan de gemeten bloeddruk bij patiënten met hypertensie (de positieve klasse) en gezonde vrijwilligers (de negatieve klasse). Stel voor het gemak dat de negatieve klasse (in rood) een gemiddelde x-waarde heeft van $\mu = -1$ met standaardafwijking $\sigma = 1$, en dat de positieve klasse (in groen) een gemiddelde $\mu = +1$ en standaardafwijking $\sigma = 1$ heeft. We vinden dan de *probability density function* (pdf) verdelingen in de bovenste figuur: de rode functie piekt rond $x = -1$ en de groene rond $x = +1$, maar verder hebben beide functies dezelfde vorm.

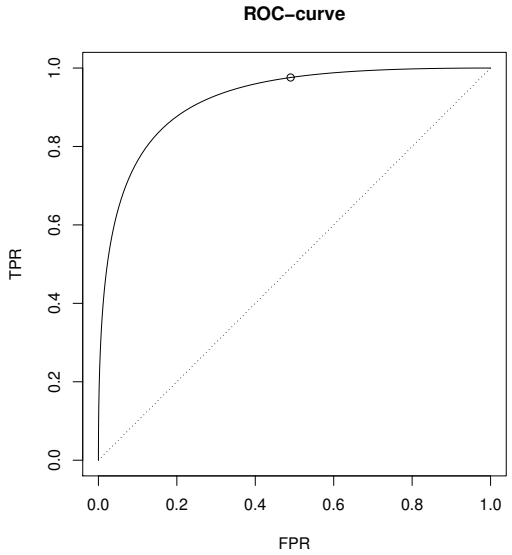


In de onderste figuur staat de bijbehorende *cumulative distribution function* (cdf) uit, die weergeeft welke fractie van instances een waarde hebben kleiner dan een zekere grenswaarde x'. Bijvoorbeeld, ongeveer 84% van de negatieve (rode) instances heeft een waarde kleiner dan nul, terwijl slechts ongeveer 16% van de positieve (groene) instances een waarde heeft kleiner dan nul.

Merk op dat de pdf de afgeleide is van de cdf, en de cdf de primitieve van de pdf; met andere woorden, ze zijn aan elkaar gerelateerd door differentiëren en integreren.

Stel we gebruiken een classificatie-algoritme dat alle instances met x kleiner dan een zekere grenswaarde x' labelt als negatief (rood), en alle instances met x groter dan die grenswaarde x' als positief (groen). Ga voor jezelf na dat de rode cdf-curve hierboven dan de TNR (oftewel $1 - \text{FPR}$) weergeeft: hoe hoger de grenswaarde is, hoe meer instances als negatief worden bestempeld, en hoe beter de prestaties van het algoritme dus zijn voor de negatieve klasse. De groene curve geeft de FNR (oftewel $1 - \text{TPR}$) weer: hoe hoger de grenswaarde, hoe minder instances positief gelabeld worden, en hoe slechter het algoritme de ware positieve instances dus zal herkennen. Als we bijvoorbeeld een grenswaarde $x' = -1$ kiezen, dan labelen we de helft van de negatieve instances juist en de andere helft onjuist ($\text{FPR} = 0.500$), maar dan labelen we wel de overgrote meerderheid van de positieve instances juist ($\text{TPR} = 0.977$).

Plotten we de TPR nu als functie van de FPR voor verschillende grenswaarden x', dan verkrijgen we de ROC-curve die hieronder wordt getoond. Het zojuist gevonden punt met $\text{FPR} = 0.500$ en $\text{TPR} = 0.977$ is gemarkeerd.



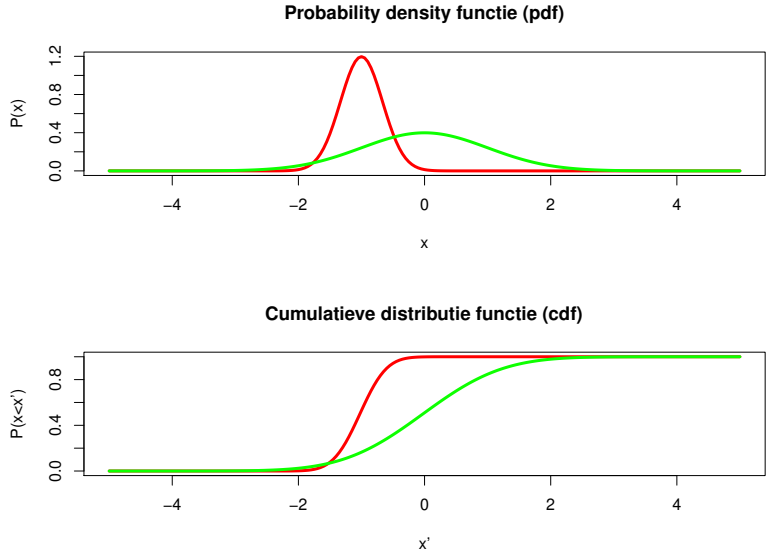
De punten in de linkerbovenhoek combineren een lage FPR met een hoge TPR; dit zijn beide wenselijke kenmerken. De punten nabij de oorsprong linksonder combineren een lage FPR met een lage TPR, oftewel er worden überhaupt nauwelijks instances positief gelabeld; de punten rechtsboven combineren FPR = 1 met TPR = 1, oftewel alle instances worden positief gelabeld.

De punten langs de gestippelde diagonaal kunnen bereikt worden door willekeurig een klasselabel toe te kennen. Stel dat bijvoorbeeld een munt wordt opgeworpen om het klasselabel te bepalen, dan verwachten we FPR = $\frac{1}{2}$ en TPR = $\frac{1}{2}$; de kansen op een positieve voorspelling zijn voor beide klassen gelijk, zodat FPR = TPR (wat overeenkomt met de diagonaal $y = x$ in de ROC-curve). Curves boven de diagonaal duiden op een algoritme dat het systematisch beter doet dan een dergelijk willekeurig algoritme, en curves onder de diagonaal duiden op een algoritme dat het systematisch slechter doet. De punten in de rechter benedenhoek duiden op een bijzonder slecht algoritme dat de positieve instances classificeert als negatief, en andersom. Echter, een dergelijk algoritme kan worden omgezet in een nuttig algoritme door de geproduceerde labels om te keren.

Een maat die gevoelig is voor de kwaliteit van het algoritme zonder daarbij af te hangen van de gekozen grenswaarde wordt gevormd door de oppervlakte onder de bovenstaande ROC-curve (Area Under the Curve, AUC). Voor de bovenstaande curve is die gelijk aan AUC = 0.921; een ideaal algoritme benadert AUC = 1. Als vuistregel worden oppervlakten boven de 0.90 als uitstekend beschouwd en boven de 0.70 als voldoende. Dit hangt overigens wel van de toepassing af: voor de diagnose van een ziekte op grond van bloedwaarden is de verwachte AUC vermoedelijk minder hoog dan voor het onderscheiden van katten en honden op basis van een digitale foto.

Opgave 7. Stel dat de distributie van de negatieve klasse (in rood) gekarakteriseerd wordt door $\mu = -1$ en $\sigma = \frac{1}{3}$, en dat de positieve klasse (in groen) gekarakteriseerd wordt door $\mu = 0$ en $\sigma = 1$, overeenkomstig de cdf en pdf hieronder. Stel dat de grenswaarde

gelijk gekozen wordt aan $x' = 0$; dat wil zeggen, alle $x < 0$ wordt geclassificeerd als negatief, alle $x > 0$ wordt geclassificeerd als positief. Schat de TPR en de FPR.



Opgave 8. Schets voor de bovenstaande distributies van positieven en negatieven de bijbehorende ROC-curve en schat op het oog de AUC. Wat zou een geschikte grenswaarde x' zijn, denk je?

Om de prestaties van een classificatie-algoritme te evalueren kan de ROC-curve prima gebruikt worden. Het is hiervoor gebruikelijk om de probabilistische uitkomsten van het algoritme in ogenschouw te nemen. Beschouw bijvoorbeeld de weather_numeric dataset waarop het Naive Bayes algoritme is toegepast om de kansen op de klasse-labels Yes en No te berekenen. De 14 instances zijn hieronder gesorteerd op grond van hun probabilistische classificatie (zie de laatste twee kolommen).

Attributen				Klasse	Classificatie	
Outlook	Temp.	Humid.	Windy	Play	P(Yes)	P(No)
sunny	85	85	FALSE	No	0.157	0.843
sunny	80	90	TRUE	No	0.175	0.825
sunny	72	95	FALSE	No	0.392	0.608
rainy	71	91	TRUE	No	0.441	0.559
rainy	70	96	FALSE	Yes	0.539	0.461
sunny	75	70	TRUE	Yes	0.694	0.306
overcast	72	90	TRUE	Yes	0.744	0.256
overcast	83	86	FALSE	Yes	0.750	0.250
rainy	65	70	TRUE	No	0.761	0.239
rainy	75	80	FALSE	Yes	0.787	0.213
rainy	68	80	FALSE	Yes	0.796	0.204
sunny	69	70	FALSE	Yes	0.848	0.152
overcast	81	75	FALSE	Yes	0.908	0.092
overcast	64	65	TRUE	Yes	0.944	0.056

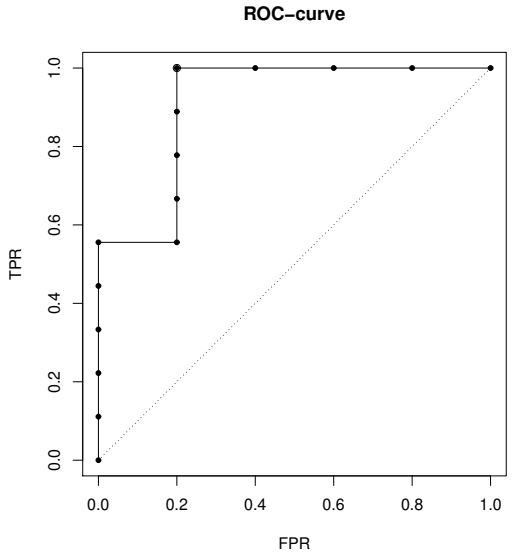
Drempelen we de toegekende kansen standaard op $P(\text{Yes}) = \frac{1}{2}$, dan worden de instances gescheiden langs de horizontale lijn in de tabel. Alle instances boven de lijn krijgen het klasselabel No toegekend door Naive Bayes, en alle instances onder de lijn het klasselabel Yes. Alle 9 instances met de ware klasse Yes worden derhalve juist geclassificeerd, en van de 5 instances met de ware klasse No worden er 4 juist geclassificeerd. We komen hiermee tot de onderstaande confusion matrix.

		Toegekende klasse:	
		Yes	No
Ware klasse:	Yes	9	0
	No	1	4

Relevant voor het construeren van de ROC-curve zijn de uitkomsten TPR $= \frac{9}{9} = 1.00$ en FPR $= \frac{1}{5} = 0.20$.

Opgave 9. Stel dat we in het voorbeeld hierboven de drempel leggen op $P(\text{Yes}) = \frac{1}{4}$. Construeer zelf de confusion matrix die dan bereikt wordt en bepaal de TPR en FPR.

Door de drempel te variëren kunnen fout-positieven en fout-negatieven tegen elkaar worden uitgewisseld, zoals dat geschiedt bij cost-sensitive classification. Laten we de drempel oplopen van $P(\text{Yes}) = 0$ (waarbij alle instances als Yes gelabeld worden) tot $P(\text{Yes}) = 1$ (waarbij alle instances als No gelabeld worden), dan nemen de TPR en FPR beide af van 1.00 naar 0.00. De ROC-curve die we hierbij vinden ziet eruit zoals hieronder. De hierboven uitgewerkte drempeling op $P(\text{Yes}) = \frac{1}{2}$ komt overeen met het vet gemarkeerde datapunt.



Opgave 10. Waar in deze ROC-curve vind je het punt dat overeenkomt met de drempeling op $P(\text{Yes}) = \frac{1}{4}$ uit de vorige opgave?

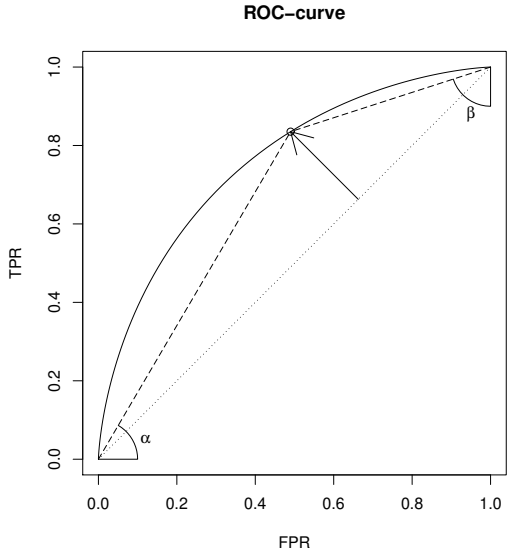
Opgave 11. Stel dat de cost van een fout-positieve in de weather_numeric dataset $3\frac{1}{2}$ maal zo hoog zou worden gekozen als die van een fout-negatieve. Ga na dat bij cost-sensitive classification dan de drempel nabij $P(\text{Yes}) = 0.778$ zou worden gelegd in plaats van op $P(\text{Yes}) = 0.500$. Op welk punt in de eerder getekende trapvormige ROC-curve komen we dan uit?

De kwaliteit van het classificatiemodel kan worden gekwantificeerd door naar de oppervlakte onder de ROC-curve te kijken. Het is kwestie van "hokjes tellen" om te ontdekken dat voor het model hierboven AUC $= \frac{41}{45} = 0.911$, hetgeen een prima classificatie aanduidt.

Opgave 12. Ga na wat er in het gegeven voorbeeld van de weather_numeric dataset met de ROC-curve zou gebeuren als de toegekende klassen exact zouden worden omgekeerd, dat wil zeggen de instances boven de lijn in de tabel krijgen het klasselabel Yes en die eronder het klasselabel No. Ga eveneens na wat er zou gebeuren als de klasse No als de positieve klasse zou worden behandeld, en de klasse Yes als de negatieve. Hoe verandert in beide gevallen de AUC?

Gegeven een ROC-curve, kan tenslotte de vraag gesteld worden welke drempeling het beste gehanteerd kan worden. Zoals gezegd zijn punten in de linker bovenhoek het meest wenselijk. De nauwkeurigheidsmaten TPR en TNR staan uitgezet in de ROC-grafiek en kunnen dus direct afgelezen worden: hoe hoger in de grafiek langs de y-as, hoe hoger de TPR (en hoe lager daarmee de FNR); hoe verder naar links in de grafiek langs de x-as, hoe hoger de TNR (en hoe lager daarmee de FPR). De PPV en NPV kunnen niet direct uit de grafiek worden afgelezen, maar desalniettemin is eenvoudig te zien welke punten

op de curve deze maten optimaliseren: de PPV is namelijk hoger (en de FDR daarmee lager) naarmate de helling van de verbindingslijn met de oorsprong (de hoek α in de onderstaande grafiek) groter is; de NPV is hoger (en de FOR daarmee lager) naarmate de soortgelijke hoek β groter is. De accuracy ACC tenslotte is grofweg het hoogst (en de error rate ERR daarmee grofweg het laagst) voor het punt dat het verst van de diagonaal af ligt, hieronder aangegeven met de pijl (deze interpretatie hangt eigenlijk af van de verhouding tussen het aantal positieve en negatieve instances en klopt alleen exact als er evenveel positieve als negatieve instances zijn, maar als vuistregel kan deze benadering wel worden gehanteerd). Kortom: het optimale punt langs de curve ligt ver naar boven en ver naar links, zorgt voor grote hoeken α en β, en ligt zo ver mogelijk schuin boven de diagonaal.



Opgave 13. Bereken voor de drie genoemde drempels uit de vorige opgaven ($P(\text{Yes}) = \frac{1}{4}$, $P(\text{Yes}) = \frac{1}{2}$, en $P(\text{Yes}) = 0.778$) de uitkomsten voor de verschillende nauwkeurigheidsmaten (ACC, TPR, TNR, PPV, NPV). Welke drempel bereikt voor elke maat de beste resultaten, en hoe had je dat direct uit de ROC-curve kunnen aflezen?