

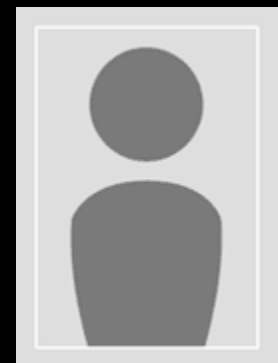
Data Science 2

Evaluatiemetrieken

Quote van de week

“Every line is the perfect length if you don't measure it.”

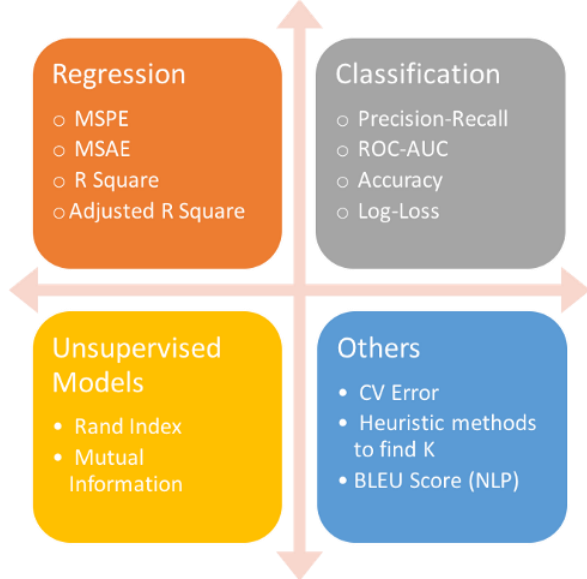
Marty Rubin



Agenda



1. Wat zijn evaluatiemetrieken?
2. Classificatie-metrieken
3. Evaluatiemetrieken in R
4. In de IT praktijk
5. In de Media
6. Antwoorden vragenlijst
7. Opdrachten



Wat zijn evaluatiemetrieken?

Evaluatiemetriecken

Machine Learning Algorithmen

Supervised Machine Learning Algorithmen

Regressie

pogingen om de sterkte en het karakter van de relatie tussen een afhankelijke variabele en een of een reeks onafhankelijke variabelen te bepalen.

Classificatie

het probleem van het identificeren tot welke van een reeks categorieën (subpopulaties) een waarneming (of waarnemingen) behoort.

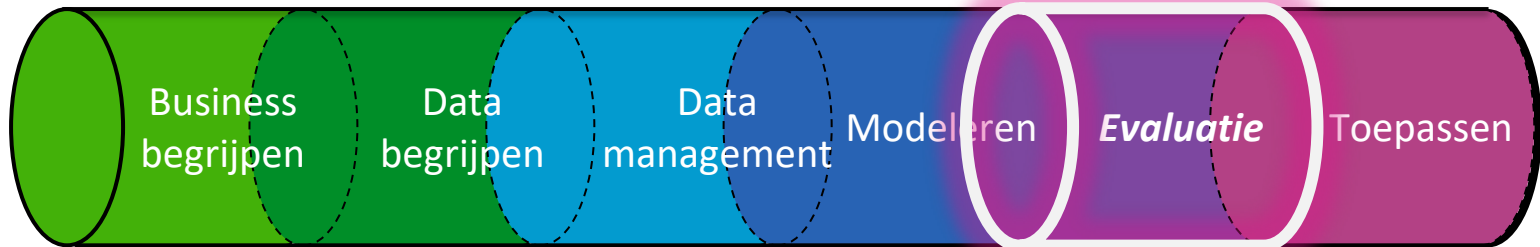
Unsupervised Machine Learning Algorithmen

Clustering

het groeperen van objecten, zodat objecten in hetzelfde cluster meer op elkaar lijken dan op objecten in een ander cluster.

Associatie

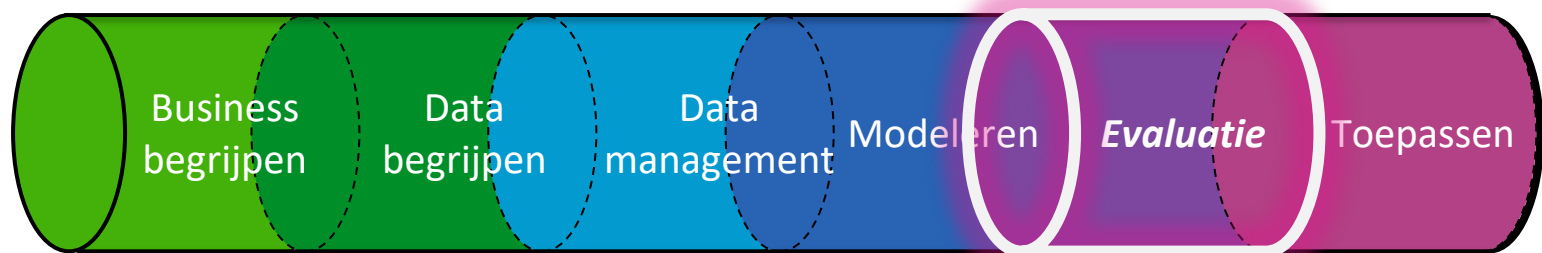
vindt belangrijke relaties tussen variabelen of functies in een gegevensset.



Evaluatiemetrieken

Maatstaf (**metriek**) om kwaliteit van het model te berekenen

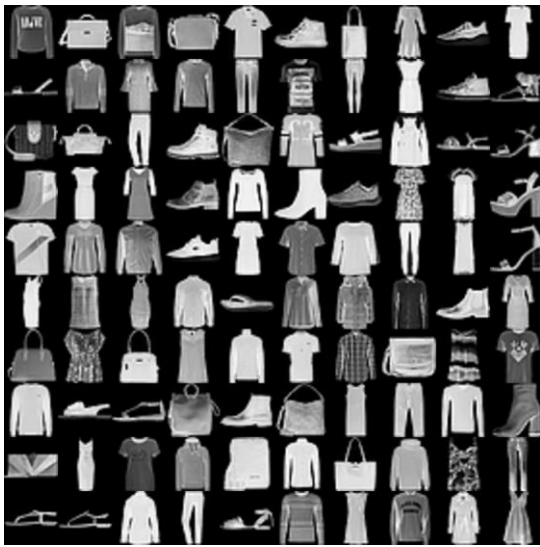
In Data Science behandelde modelleringstechniek	Evaluatietechniek	
Lineaire regressie	R^2 , MAE, RMSE, MAPE	Regressie
Forecasting, ANN	MAE, RMSE, MAPE	
Clustering	interclusterafstand, intraclusterafstand	Clustering
Association rules	support, confidence, lift	Associatie
Metaheuristieken	objective function	Andere
Naive Bayes, beslissingsbomen, discriminant analyse, ANN	accuracy, precision, recall, TPR, FPR, ROC	Classificatie



Evaluatiemetrieken - Regressie

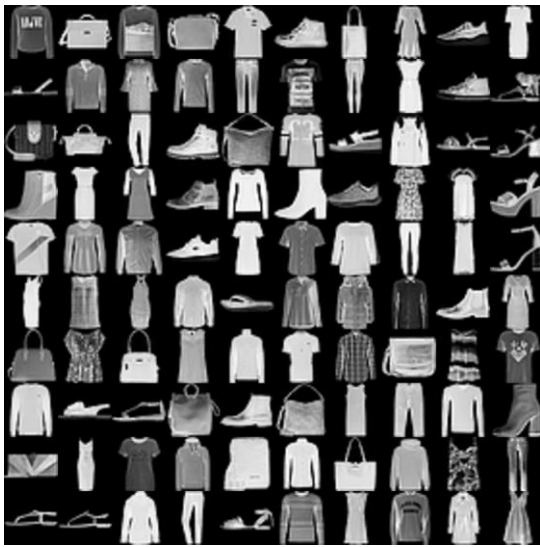
De regressie-evaluatiemetrieken uit Data Science 1 nog even op een rijtje:

- $R^2 = \left(\frac{1}{n} \sum_{i=1}^n Z_{x_i} Z_{y_i} \right)^2 = r_{xy}^2$
- Mean Absolute Error, $MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$
- Root Mean Square Error, $RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$
- Mean Absolute Percentage Error, $MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|$



Classificatie-metrieken

- ↪ Binaire vs.
Multi-class classificatie
- ↪ Metrieken voor binaire en
multiclass classificatie
- ↪ Metrieken enkel voor binaire
classificatie



Classificatie-metrieken

- ↪ **Binaire vs.**
- Multi-class classificatie**
- ↪ Metrieken voor binaire en multiclass classificatie
- ↪ Metrieken enkel voor binaire classificatie

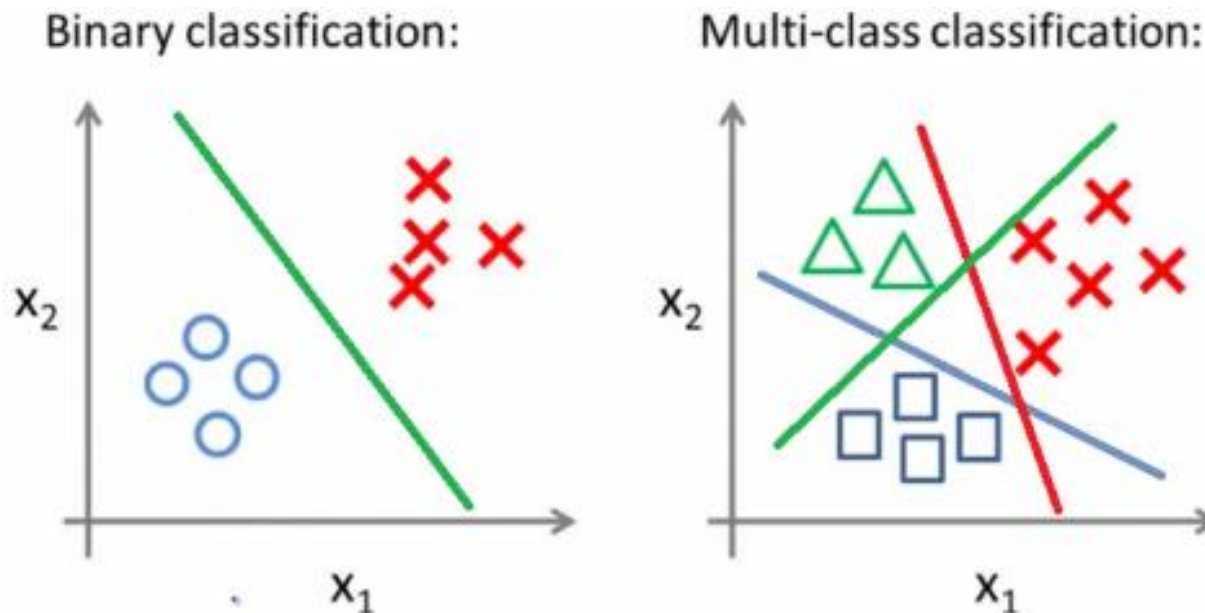
Binaire vs. Multi-class classificatie

Binaire classificatie - er zijn slechts twee targets/klassen

Vbn: M/V, goed/slecht, fraude/geen fraude,...

Multi-Class classificatie - er meer dan twee targets/klassen

vb: Onvoldoende/Voldoende/Onderscheiding/Grote onderscheiding



Classifier = Model dat de klassen voorspelt. (vb. Decision tree)

Evaluatiemetrieken - Classificatie

Een overzicht van de belangrijkste metrieken voor classificatie:

a. Accuracy (ACC)
 b. Precision & Recall
 c. F-measure

}
 }
 }

binaire en multi-class classificatie
(op basis van een Confusion Matrix)

d. TP Rate & FP Rate
 (Confusion Matrix)
 e. ROC – curve
 (Receiver Operating Characteristic)

}
 }
 }

Enkel binaire classificatie

<u>True negative</u> Predicted negative Actual negative	<u>False positive</u> Predicted positive Actual negative
<u>False negative</u> Predicted negative Actual positive	<u>True positive</u> Predicted positive Actual positive

Classificatie-metrieken

- ↪ Binaire vs.
Multi-class classificatie
- ↪ **Metrieken voor binaire en multiclass classificatie**
- ↪ Metrieken enkel voor binaire classificatie

Binaire en multi-class - Confusion Matrix

Een kruistabel waarin de aantallen van **juiste en onjuiste voorspellingen worden opgelijst**

		Voorspelde klasse		
		kat	hond	konijn
Werkelijke klasse	kat	60	3	1
	hond	6	40	1
	konijn	2	10	20

Binaire en multi-class

Gebruikte terminologie:

1. **echt positiefs (TP)**: aantal instanties die foutloos als **klasse** geclassificeerd worden
2. **echt negatives (TN)**: aantal instanties die foutloos als **andere klasse** geclassificeerd werden
3. **vals positiefs (FP)**: aantal instanties die verkeerdelijk als **klasse** worden geclassificeerd.
4. **vals negatives (FN)**: aantal instanties die verkeerdelijk als **andere klasse** worden geclassificeerd.

		Voorspelde klasse	
		positief	negatief
Werkelijke klasse	positief	TP	FN
	negatief	FP	TN

Binaire en multi-class - Confusion Matrix

Classificatie

Confusionmatrix voor klassen KAT HOND en KONIJN.
TP, TN, FN, FP in functie van KAT

			Voorspelde klasse		
			KAT	NIET KAT	
				HOND	KONIJN
Werkelijke klasse	KAT	KAT	TP	FN	FN
	NIET KAT	HOND	FP	TN	TN
		KONIJN	FP	TN	TN

Binaire en multi-class - Confusion Matrix

Vereenvoudigde confusionmatrix in functie van de klasse KAT
-> Binaire confusionmatrix

		Voorspelde klasse	
		KAT	NIET KAT
Werkelijke klasse	KAT	TP	FN
	NIET KAT	FP	PTN

Hoe groter de aantallen op de hoofddiagonaal, relatief ten opzichte van de andere waarden, hoe beter de classifieer.

Binaire en multi-class– a. Accuracy

Accuracy

Een van de meest gebruikte maatstaven voor classificatie, maar niet altijd de meest informatieve en nuttige (zie verder).

Hoeveel keer heeft de classifier de correcte voorspelling gedaan t.o.v. het totaal aantal?

$$\text{accuracy} = \frac{\text{Aantal juiste voorspellingen}}{\text{Aantal voorspellingen}}$$

Dit is de meest intuïtieve manier om de kwaliteit van een classifier te meten.

Confusion matrix: Aantallen op hoofddiagonaal delen door totaal.

$$\text{accuracy} = \frac{TP + TN}{TP + FP + TN + FN}$$

Binaire en multi-class - Confusion Matrix

Voorbeeld – ternaire classificatie (klassen kat, hond & konijn)

		Voorspelde klasse			
		kat	hond	konijn	Totaal werkelijk
Werkelijke klasse	kat	60	3	1	64
	hond	6	40	1	47
	konijn	2	10	20	32
Totaal voorspeld		68	53	22	143

$$accuracy = \frac{60 + 40 + 20}{143} = 0,84$$

Binaire en multi-class – b. Precision & Recall

Precision (P)

Hoe goed is een classifier in het selecteren van de 'echt positiefs' (TP)? Hoeveel van de door de classifier gemarkeerde positieven zijn ook effectief positief?

$$P = \textit{precision} = \frac{TP}{TP + FP}$$

Te berekenen per klasse

Op zichzelf geeft precision onvoldoende informatie over de kwaliteit van de classifier.

Waarom?

Een 'voorzichtige' classifier die bv. slechts één instantie correct als positief classificeert en alle andere als negatief, heeft een precision van 100%. Toch is dit geen goede classifier, want zijn recall (zie verder) zal laag zijn.

Opmerking: **Positive Predictive Value (PPV)** is een andere naam voor precision

Binaire en multi-class - b. Precision & Recall

Voorbeeld – ternaire classificatie (klassen kat, hond & konijn)

$Precision(A) = \frac{echt\ A}{Totaal\ voorspeld\ A}$

$Recall(A) = \frac{echt\ A}{Totaal\ echt\ A}$

		Voorspelde klasse				
		kat	hond	konijn	Totaal werkelijk	Class Recall
Werkelijke klasse	kat	60	3	1	64	?
	hond	6	40	1	47	?
	konijn	2	10	20	32	?
	Totaal voorspeld	68	53	22	143	
	Class Precision	0,88	?	?		

$Precision(kat) = \frac{60}{68}$

Binaire en multi-class – b. Precision & Recall

Recall (R)

Hoeveel positieve instanties van alle positieve instanties in de populatie kan de classifier aanduiden?

$$R = \textit{recall} = \frac{TP}{TP + FN}$$

Te berekenen per klasse

Op zichzelf geeft recall –net als precision- ook weer onvoldoende informatie over de kwaliteit van de classifier.

Waarom?

Een 'onvoorzichtige' classifier die domweg alle instanties als positief classificeert, zal de hoogst mogelijke recall hebben. Toch is dit geen goede classifier, want zijn precision (zie hiervoor) zal laag zijn.

Opmerking: **True Positive Rate (TPR)**, **sensitivity** zijn andere namen voor recall

Binaire en multi-class - Confusion Matrix

Voorbeeld – ternaire classificatie (klassen kat, hond & konijn)

$$\text{Precision}(A) = \frac{\text{echt } A}{\text{Totaal voorspeld } A}$$

$$\text{Recall}(A) = \frac{\text{echt } A}{\text{Totaal echt } A}$$

		Voorspelde klasse				
		kat	hond	konijn	Totaal werkelijk	Class Recall
Werkelijke klasse	kat	60	3	1	64	0,94
	hond	6	40	1	47	?
	konijn	2	10	20	32	?
Totaal voorspeld		68	53	22	143	
Class Precision		0,88	?	?		

$\text{Recall (kat)} = \frac{\text{echt kat}}{\text{Totaal echt kat}} = \frac{60}{64} = 0,94$

Binaire en multi-class - Confusion Matrix

Voorbeeld – ternaire classificatie (klassen kat, hond & konijn)

$$\text{Precision}(A) = \frac{\text{echt } A}{\text{Totaal voorspeld } A}$$

$$\text{Recall}(A) = \frac{\text{echt } A}{\text{Totaal echt } A}$$

		Voorspelde klasse				
		kat	hond	konijn	Totaal werkelijk	Class Recall
Werkelijke klasse	kat	60	3	1	64	0,94
	hond	6	40	1	47	0,85
	konijn	2	10	20	32	0,63
	Totaal voorspeld	68	53	22	143	
	Class Precision	0,88	0,75	0,91		

Als ik voorspel dat het dier een konijn is dan is dat meestal ook zo...

... maar ik voorspel regelmatig een (werkelijk) konijn als een ander dier

Binaire en multi-class – c. F-measure

F-measure

Het is duidelijk dat precision en recall op gespannen voet staan met elkaar. Hoe groter de precision, hoe kleiner de recall en vice versa.

De F-measure neemt het **gewogen harmonisch gemiddelde** van beide maatstaven en geeft ons één goede maatstaf.

$$F_{\alpha} = \frac{2}{\alpha \frac{1}{P} + (1 - \alpha) \frac{1}{R}}$$

$$F_{\beta} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

waarbij $\beta = \frac{1-\alpha}{\alpha}$

- $\beta = 1$ ($\Rightarrow \alpha = 0.5$): dit zorgt voor de F_1 -measure die een gelijk gewicht toekent aan de precision en recall
- $\beta > 1$: meer belang hechten aan recall
- $\beta < 1$: meer belang hechten aan precision

Binaire en multi-class – c. F-measure

$$F_{\beta} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

$$F_1 \text{ voor kat} = \frac{(1^2 + 1) \cdot 0,88 \cdot 0,94}{(1^2 \cdot 0,88) + 0,94} = \frac{1,6544}{1,82} = 0,901$$

*Precision(A) =
echt A / Totaal voorspeld A*

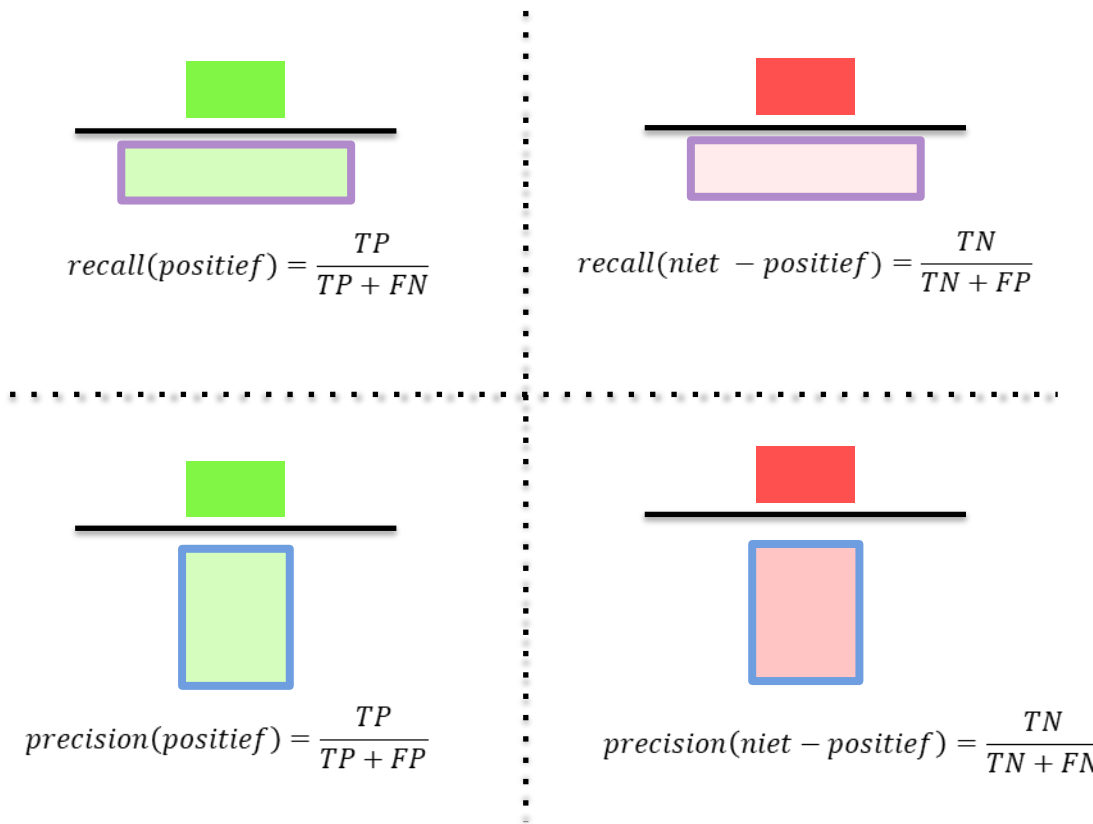
*Recall(A) =
echt A / Totaal echt A*

		Voorspelde klasse				
		kat	hond	konijn	Totaal werkelijk	Class Recall
Werkelijke klasse	kat	60	3	1	64	0,94
	hond	6	40	1	47	0,85
	konijn	2	10	20	32	0,63
	Totaal voorspeld	68	53	22	143	
		Class Precision	0,88	0,75	0,91	

Binaire en multi-class

Binaire classificatie is speciaal geval multi-class classificatie

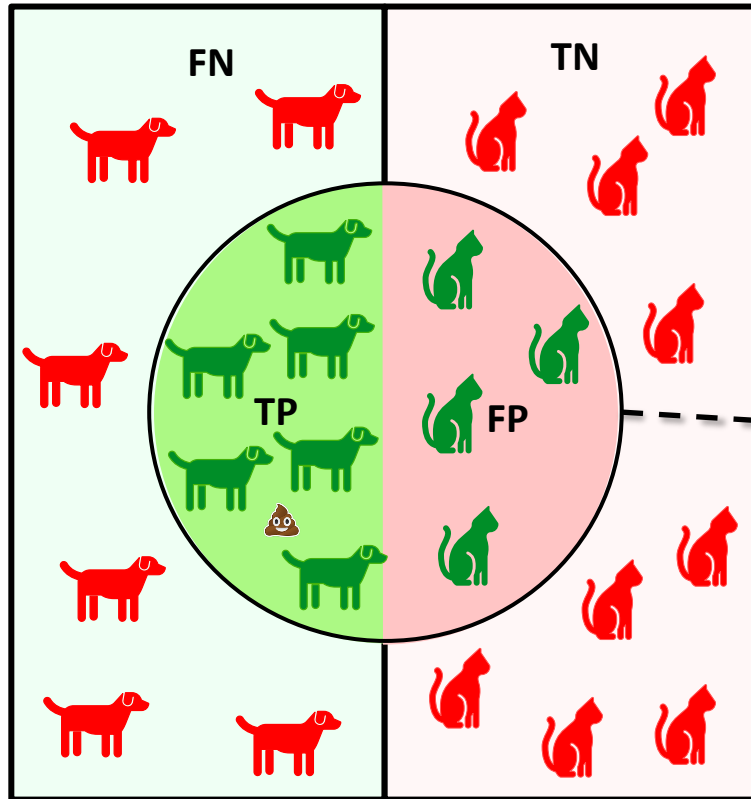
Je kan dus bij binaire classificatie eigenlijk ook twee precisions, recalls & f-measures berekenen.



		Voorspelde klasse	
		positief	negatief
Werkelijke klasse	positief	TP	FN
	negatief	FP	TN

Binaire en multi-class

Voorspellen honden: Precision & Recall



$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

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

Alle gevallen binnen de cirkel worden door de classifier als positief (hond) voorspeld.
Sommige van deze zijn correct (TP) maar anderen zijn verkeerd (FP)

Alle gevallen buiten de cirkel worden voorspeld als negatief (geen hond).
Sommigen van hen zijn correct (TN) maar anderen zijn verkeerd (FN)

relevante gevallen

Binaire en multi-class - Opmerking

Slechte classifier
Voorspelt enkel B
→ inherent grote skew in dataset
A = 110
B = 990

		Voorspelde klasse	
		A	B
Werkelijke klasse	A	0	110
	B	0	990

	accuracy	precision	recall
A	😊	😞	😞
B	😊	😊	😊



Classificatie-metrieken

- ↪ Binaire vs. Multi-class classificatie
- ↪ Metrieken voor binaire en multiclass classificatie
- ↪ **Metrieken enkel voor binaire classificatie**

Enkel binaire – d. TP Rate & FP Rate

-- $TPR = \frac{TP}{TP+FN}$ (True Positive Rate: echt positieven gedeeld door alle positieven)

$FPR = \frac{FP}{FP+TN}$ (False Positive Rate: valse positieven gedeeld door alle negatieven)

-- Dit is gewoon de Recall

		Voorspelde klasse	
		positief	negatief
Werkelijke klasse	positief	TP	FN
	negatief	FP	TN

Enkel binaire – d. TP Rate & FP Rate

Modellen die binair classificeren geven vaak een kans of score in plaats van een exacte klassen.
Er wordt een **threshold** vastgelegd waarboven een positive wordt toegekend.
Door de threshold aan te passen veranderen ook de uitkomsten van de metrieken.

Vb: Google [reCAPTCHA](#) gebruikt browserinfo om in te schatten of je een echte persoon bent en geen robot.
Dat model geeft een score terug. Boven een bepaalde score (vb. 60%) wordt je als “echt” beschouwd. Die 60% is de threshold.

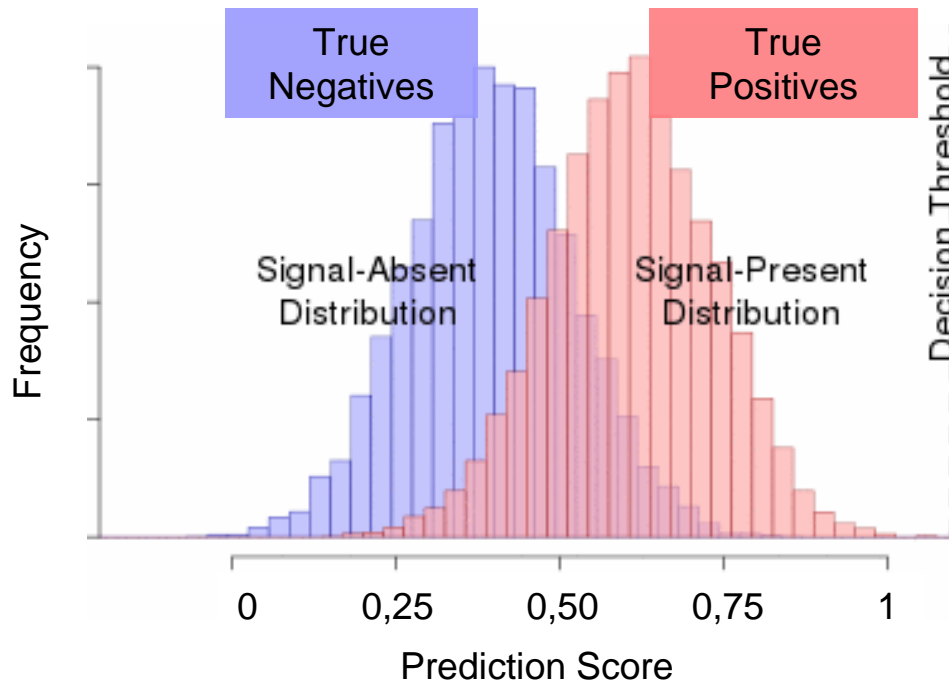
model %>% predict(x_train)		model %>% predict_classes(x_train)	#	echt class				
[1,] 0.0005334256	$t = 0.5$ →	[1,] 0	<table><tr><td>4</td><td>0</td></tr><tr><td>1</td><td>4</td></tr></table> <div>$TPR = \frac{4}{4 + 0} = 1$</div> <div>$FPR = \frac{1}{4 + 1} = \frac{1}{5}$</div>	4	0	1	4	[1,] 0
4		0						
1		4						
[2,] 0.9745124578		[2,] 1		[2,] 1				
[3,] 0.0013991351		[3,] 0		[3,] 0				
[4,] 0.9966866374		[4,] 1		[4,] 1				
[5,] 1.0000000000		[5,] 1		[5,] 1				
[6,] 0.1320384443		[6,] 0		[6,] 0				
[7,] 0.7503725886		[7,] 1		[7,] 1				
[8,] 0.5634282827	[8,] 1	[8,] 0						
[9,] 0.0058543077	[9,] 0	[9,] 0						

model %>% predict(x_train)		model %>% predict_classes(x_train)	#	echt class				
[1,] 0.0005334256	$t = 0.6$ →	[1,] 0	<table><tr><td>4</td><td>0</td></tr><tr><td>0</td><td>5</td></tr></table>	4	0	0	5	[1,] 0
4		0						
0		5						
[2,] 0.9745124578		[2,] 1	[2,] 1					
[3,] 0.0013991351		[3,] 0	[3,] 0					
[4,] 0.9966866374		[4,] 1	[4,] 1					
[5,] 1.0000000000		[5,] 1	[5,] 1					
[6,] 0.1320384443		[6,] 0	[6,] 0					
[7,] 0.7503725886		[7,] 1	[7,] 1					
[8,] 0.5634282827	[8,] 0	[8,] 0						
[9,] 0.0058543077	[9,] 0	[9,] 0						

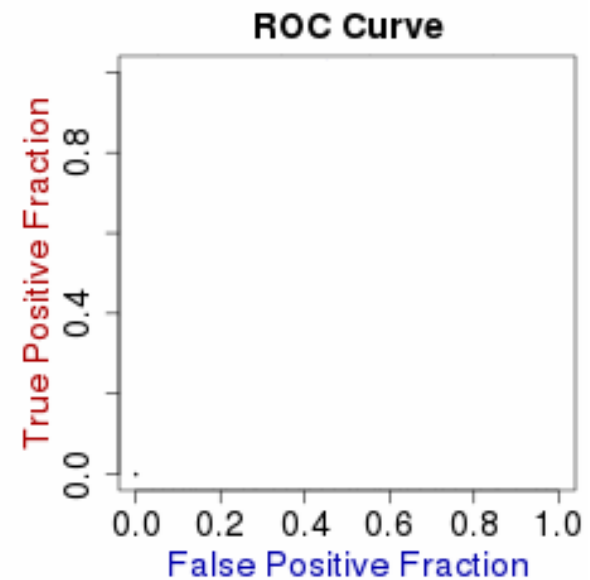


Threshold t bepaalt de waarden van de evaluatiemetrieken.

Enkel binaire – e. Receiver Operator Characteristic Curve



Een ROC-curve is een twee dimensionele curve die FPR en TPR toont voor variërende threshold-waarden.



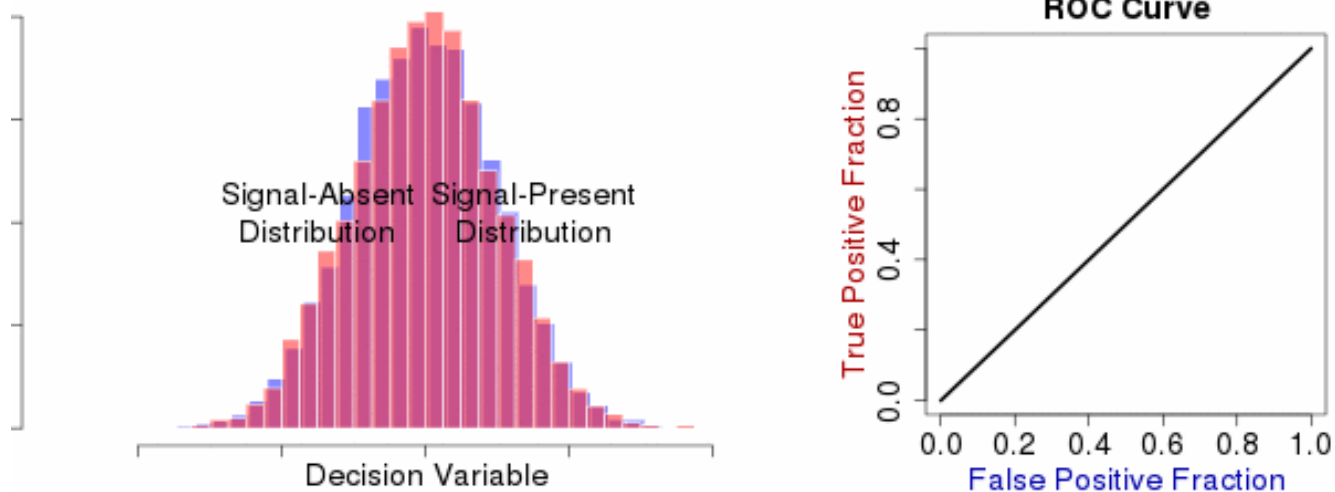
Enkel binaire – e. Receiver Operator Characteristic Curve

Hoe beter de binaire classifieer de twee klassen kan scheiden hoe groter de oppervlakte onder de ROC curve.

AUC = Area Under Curve = maatstaf voor kwaliteit classifieer.

AUC = 1 → perfecte classifieer (zeer goed)

AUC = 0.5 → random classifieer (zeer slecht)



Enkel binaire – e. Receiver Operator Characteristic Curve

Twee aspecten bepalen hoe goed je classificatie is:

1. intrinsieke kwaliteit van de classifier

- hoe goed worden positieve instanties afgebeeld op waarden dichtbij 1
- hoe goed worden negatieve instanties afgebeeld op waarden dichtbij 0

[1,] 0.4643670
[2,] 0.4683865
[3,] 0.4680982
[4,] 0.4743361
[5,] 0.4898452
[6,] 0.4569134
[7,] 0.4657828
[8,] 0.4676911
[9,] 0.4646666

slechte classifier
weinig onderscheidend

[1,] 0.0001589166
[2,] 0.9594721198
[3,] 0.0007447252
[4,] 0.8748258352
[5,] 0.9998613596
[6,] 0.0043711206
[7,] 0.6282945275
[8,] 0.6142016053
[9,] 0.0106160529

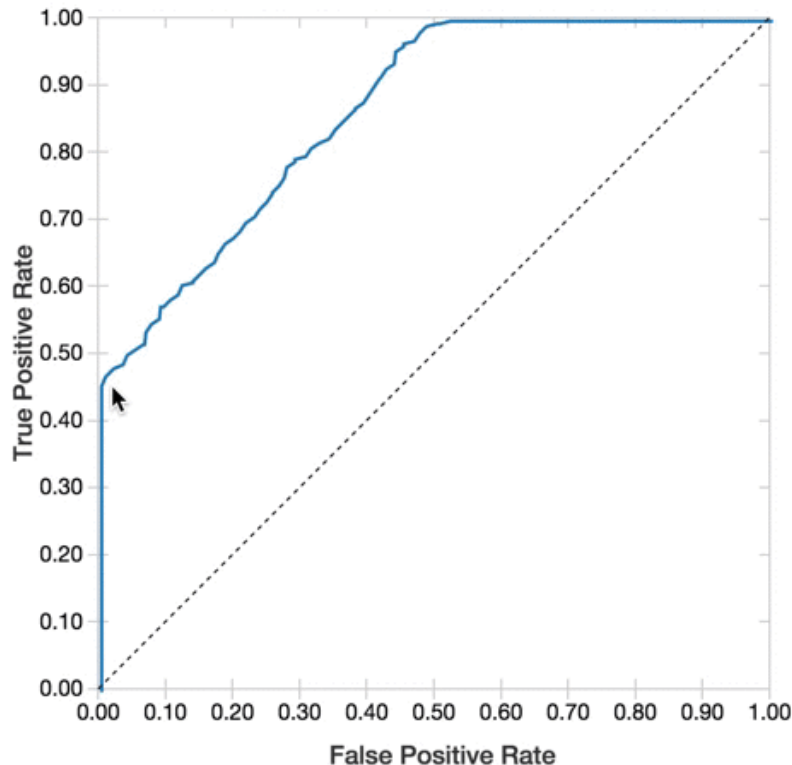
goede classifier
goed onderscheidend

Enkel binaire – e. Receiver Operator Characteristic Curve

Twee aspecten bepalen hoe goed je classificatie is:

2. keuze van de **threshold-waarde** bepaalt TPR-FPR trade-off

- $\text{TPR} \uparrow$ (goed) \rightarrow $\text{FPR} \uparrow$ (slecht)
- $\text{FPR} \downarrow$ (goed) \rightarrow $\text{TPR} \downarrow$ (slecht)



TPR-FPR trade-off in de praktijk

Welke threshold kies je als...

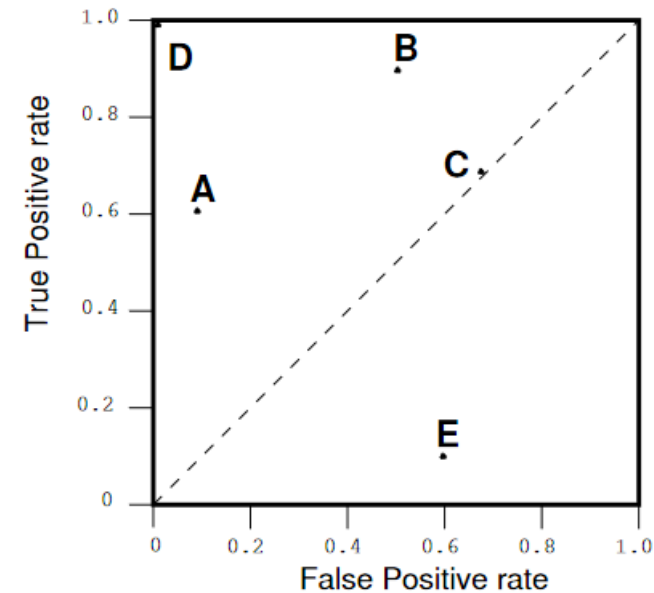
... het de screening voor borstkanker in de Mamobiel betreft ?

... KdG studenten waarvan de kans groot is dat ze het moeilijk zullen hebben met de opleiding proactief willen begeleiden, met de wetenschap dat de middelen voor begeleiding beperkt zijn?

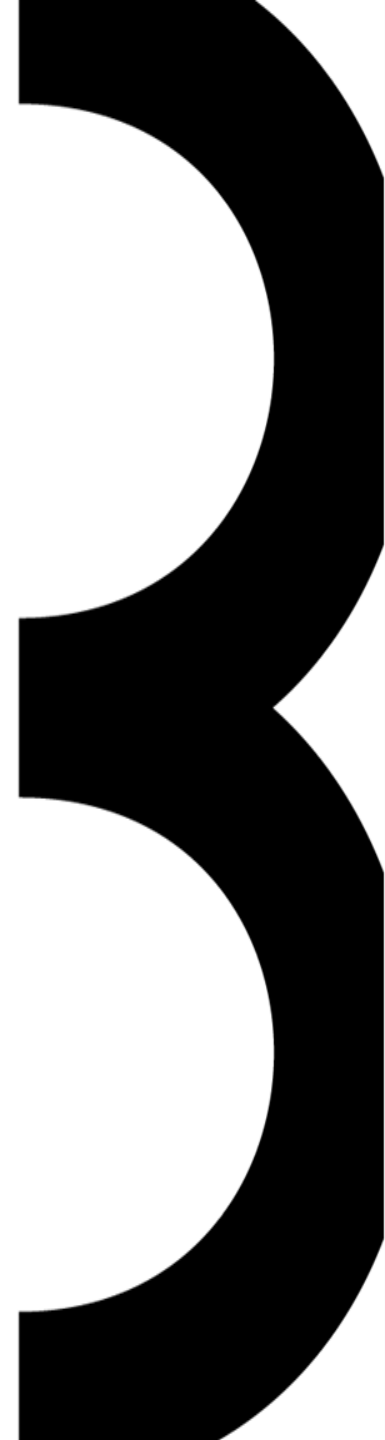
Enkel binaire – e. Receiver Operator Characteristic Curve

Enkele binaire classifiers in ROC – ruimte:

- **D** perfecte classifier, $TPR=1$, en $FPR=0$
- **C** een willekeurige classifier, $TPR=FPR$, maakt geen onderscheid
- **E** de omgekeerde classifier, voorspelt meer fout dan goed
- **A vs B**
 - *A is een “voorzichtigere” classifier dan B*
 - *A voorspelt minder ten onrechte positief*
 - *A voorspelt ook minder true positives als positief*
- **(0,0)** de “lafaard” classifier, niets fout, maar ook niets correct
- **(1,1)** de “blinde” classifier, voorspelt alleen de positieve klasse



Evaluatiemetrieken in Python



Evaluatie metriecken in Python

Een overzicht van de belangrijkste evaluatie metriecken:

- Confusion Matrix
- Accuracy
- Precision and Recall
- F-measure

SciKit Learn en Pandas bevatten deze functies, maar werken enkel op 'ruwe' data.

Indien je enkel over de confusion matrix beschikt,
→ implementeer zelf de nodige functies in Python die de evaluatie metriecken berekenen vertrekkende van een confusion matrix.

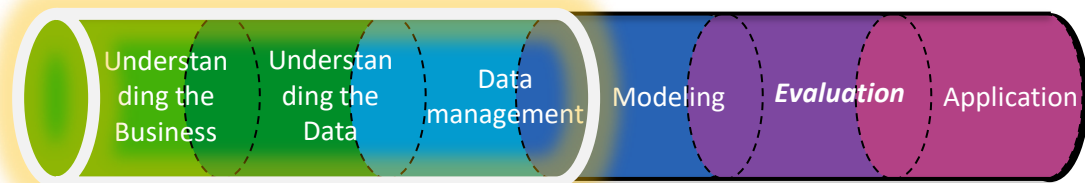
- ROC - curve

sklearn.metrics + zelfgemaakte plot functie

Accuracy, Precision, Recall en F-Measure in Python

Stap 1: Zorg ervoor dat de gegevens beschikbaar zijn:

```
>>> # download the file iris.csv from canvas.  
>>> iris = pd.read_csv('iris.csv', delimiter=',',  
                        decimal='.')  
>>> iris.info()
```



Stap 2: Maak een model om classificaties te maken

Lineaire discriminante analyse wordt/werd in een andere DS2-les behandeld.
Het enige dat je nu moet weten/herinneren, is dat je de klasse van "Species"
kunt voorspellen op basis van de kolommen "Sepal" en "Petal".

```
>>> from sklearn.discriminant_analysis import LinearDiscriminantAnalysis  
>>> X = iris[['sepal length', 'sepal width', 'petal length', 'petal width']]  
>>> y = iris['target']  
>>> model = LinearDiscriminantAnalysis()  
>>> model.fit(X, y)
```



Remark: see https://en.wikipedia.org/wiki/Iris_flower_data_set

Accuracy, Precision, Recall en F-Measure in Python

Stap 3: Maak de confusion matrix

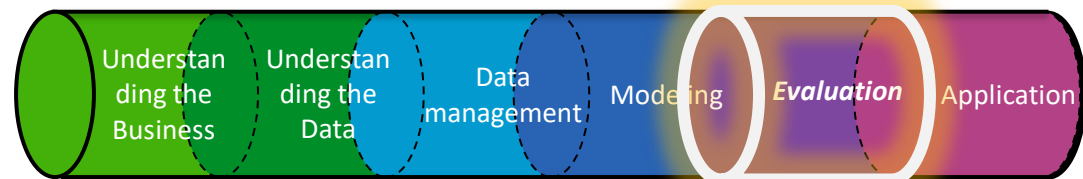
Nadat je een model hebt gemaakt, kan je de klassen voorspellen
met de predict-functie

```
>>> real = iris.target  
>>> listoflabels = sorted(real.unique().tolist())  
>>> predicted = model.predict(X)
```

Maak een confusion array met werkelijke waarden in de rijen en
de voorspelde waarden in de kolommen.

```
>>> from sklearn.metrics import confusion_matrix  
>>> conf_matrix = confusion_matrix(real,predicted,labels = listoflabels)  
>>> conf_matrix
```

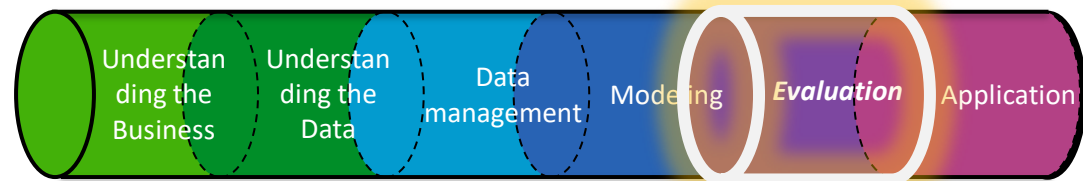
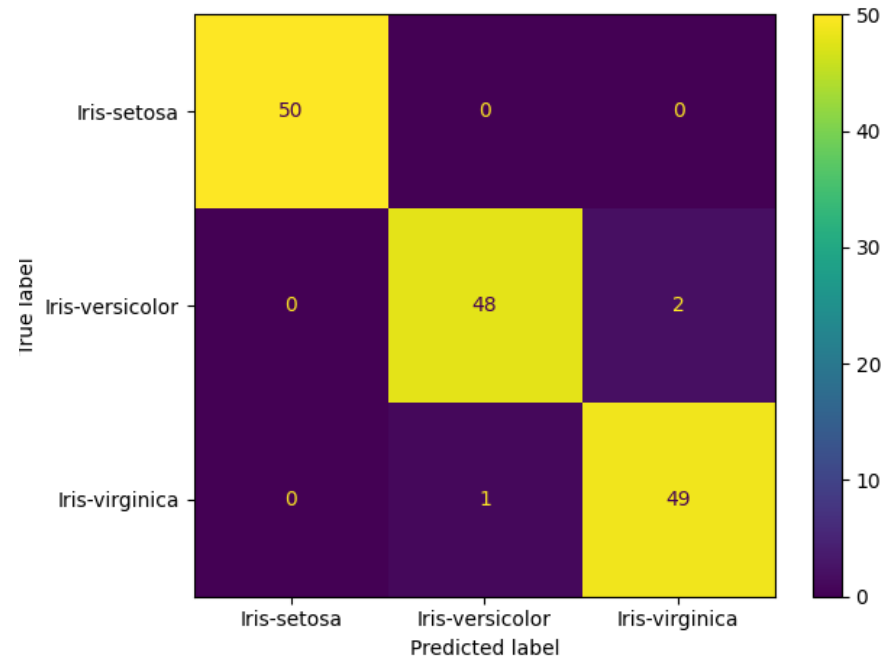
```
array([[50,  0,  0],  
       [ 0, 48,  2],  
       [ 0,  1, 49]], dtype=int64)
```



Accuracy, Precision, Recall en F-Measure in Python

visualisatie van de confusion matrix

```
>>> from sklearn.metrics import ConfusionMatrixDisplay
>>> plt.figure()
>>> cmd = ConfusionMatrixDisplay(conf_matrix, display_labels=listoflabels)
>>> cmd.plot()
>>> plt.show()
```



Accuracy, Precision, Recall en F-Measure in Python

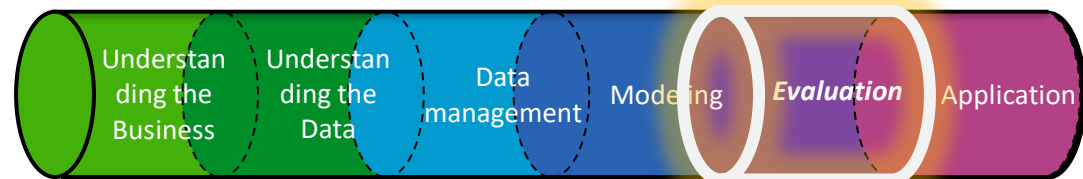
Stap 4: Bereken de verschillende metrieken

```
>>> from sklearn.metrics import accuracy_score,  
                                         precision_recall_fscore_support  
  
>>> print(real.unique())  
>>> accuracy_score (y_true=real,y_pred=predicted)  
Out[   ]: 0.98  
  
>>> precision_recall_fscore_support (y_true=real,y_pred=predicted,labels =  
real.unique(), beta=1.0)
```

```
['Iris-setosa' 'Iris-versicolor' 'Iris-virginica']
```

```
Out[   ]:
```

```
(array([1.          , 0.97959184, 0.96078431]), ← precision  
array([1.   , 0.96, 0.98]), ← recall  
array([1.          , 0.96969697, 0.97029703]), ← F1  
array([50, 50, 50], dtype=int64))
```



ROC in Python: met Scikit Learn

```
>>> from sklearn.metrics import roc_curve, roc_auc_score
```

De `roc_curve` functie heeft als input twee arrays:

1. Lijst van de werkelijke waarden (0 or 1) of labels (vb. 'M' of 'F', 'goed' of 'slecht')
2. lijst van kansen of scores van de classifier (waarden tussen -1 en 1 ofwel tussen 0 en 1)

en geeft de coördinaten van de ROC curve terug met behulp van drie arrays:

1. fpr - array: bevat **x coördinaten**
2. tpr - array: bevat **y coördinaten**
3. thresholds: bevat de **thresholds** voor de (fpr, tpr) – coördinaten

De `roc_auc_score` functie heeft als input twee arrays:

1. Lijst van de werkelijke waarden (0 or 1) of labels (vb. 'M' of 'F', 'goed' of 'slecht')
2. lijst van kansen of scores van de classifier (waarden tussen -1 en 1 ofwel tussen 0 en 1)

en geeft de **area under the ROC curve** (aka AUC) terug

ROC in Python – zelfgemaakte plot functie

```
def plot_roc(y_true, y_score, title='ROC Curve', **kwargs):
    if 'pos_label' in kwargs:
        fpr, tpr, thresholds = roc_curve(y_true=y_true, y_score=y_score, pos_label=kwargs.get('pos_label'))
        auc = roc_auc_score(y_true, y_score)
    else:
        fpr, tpr, thresholds = roc_curve(y_true=y_true, y_score=y_score)
        auc = roc_auc_score(y_true, y_score)

    # bereken de optimale cut-off met de Youden index methode
    optimal_idx = np.argmax(tpr - fpr)
    optimal_threshold = thresholds[optimal_idx]

    figsize = kwargs.get('figsize', (7, 7))
    fig, ax = plt.subplots(1, 1, figsize=figsize)
    ax.grid(linestyle='--')

    # plot de ROC curve
    ax.plot(fpr, tpr, color='darkorange', label='AUC: {}'.format(auc))
    ax.set_title(title)
    ax.set_xlabel('False Positive Rate (FPR)')
    ax.set_ylabel('True Positive Rate (TPR)')
    ax.fill_between(fpr, tpr, alpha=0.3, color='darkorange', edgecolor='black')

    # plot de classifier
    ax.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')

    # plot de optimale cut-off
    ax.scatter(fpr[optimal_idx], tpr[optimal_idx],
               label='optimal cutoff {:.2f} op ({:.2f},{:.2f})'.format(optimal_threshold, fpr[optimal_idx],
                                                                    tpr[optimal_idx]), color='red')
    ax.plot([fpr[optimal_idx], fpr[optimal_idx]], [0, tpr[optimal_idx]], linestyle='--', color='red')
    ax.plot([0, fpr[optimal_idx]], [tpr[optimal_idx], tpr[optimal_idx]], linestyle='--', color='red')

    ax.legend(loc='lower right')
    plt.show()
```

ROC in Python – Voorbeeld

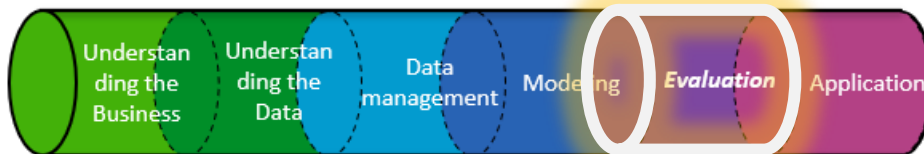
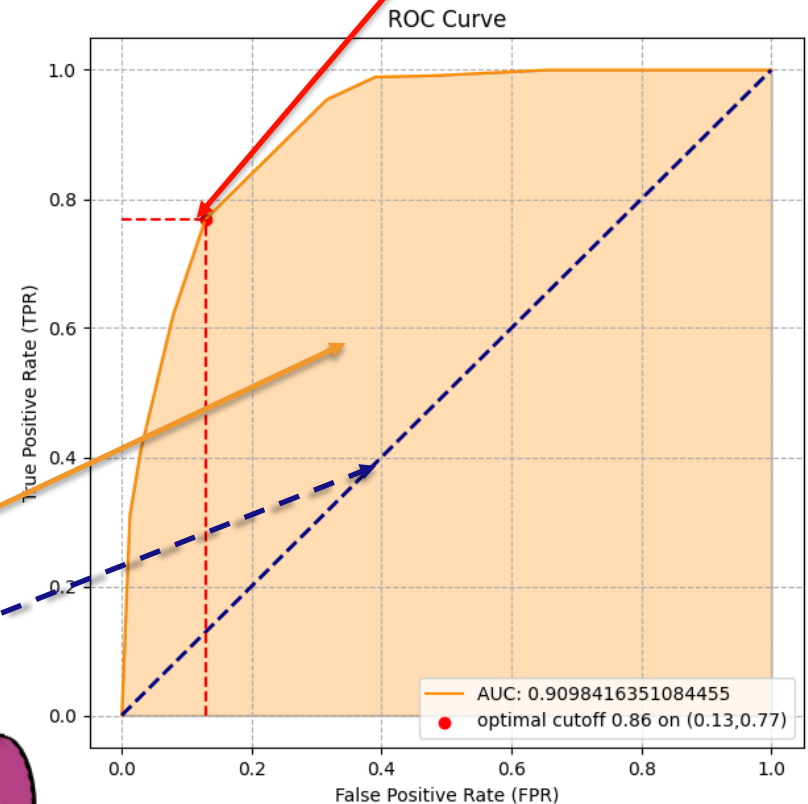
```
# Stap 1: lees de data
>>> biopsy = pd.read_csv('biopsy.csv', delimiter=',')
# Stap 2: Maak een model - LDA model fit van 1 input
# variabele (V1)
>>> X = biopsy[['V1']]
>>> y = biopsy[['class']]
>>> model = LinearDiscriminantAnalysis()
>>> model.fit(X,y)
# Stap 3: confusion matrix
>>> predicted = model.predict(X)
>>> conf_matrix = confusion_matrix(y_true=y,
                                   y_pred=predicted, labels = y.unique())
# Stap 4: bereken de metrieken
>>> accuracy_score(y_true=y,y_pred=predicted)
>>> precision_recall_fscore_support(y_true=y,
                                   y_pred=predicted, beta=1.0)
# Stap 5: Plot de ROC curve
>>> y_true = y
>>> y_score = model.predict_proba(X)[:,:0]
>>> plot_roc(y_true, y_score, pos_label='benign')
```

Lda is een binaire classifier
(zie later)

Beste threshold-waarde
threshold = 0.86
TPR = 0.77
FPR = 0.13

Area Under the Curve (AUC) is 0.910

ROC van een "willekeurige" binaire classifier



ROC in Python – Voorbeeld

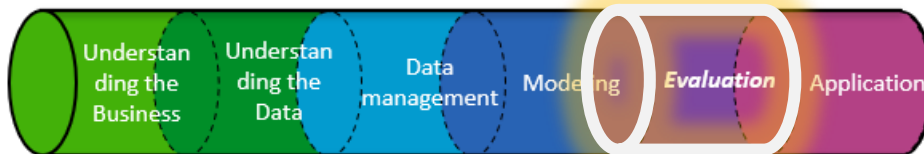
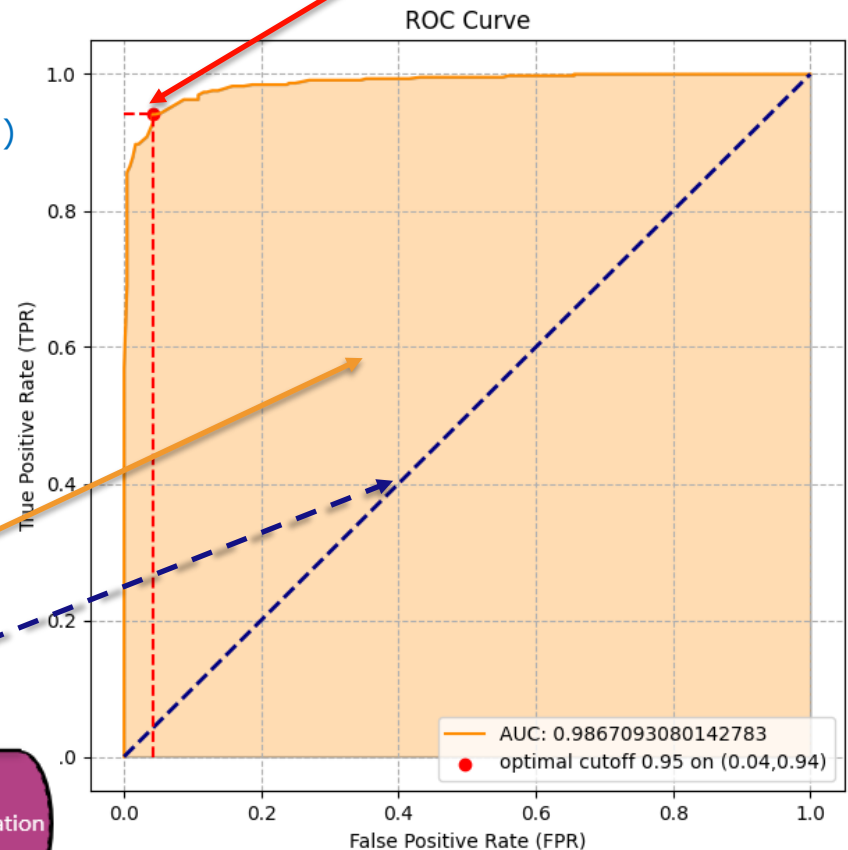
```
# Stap 1: lees de data
>>> biopsy = pd.read_csv('biopsy.csv', delimiter=',')
# Stap 2: Maak een model - LDA model fit van 2 input
variabelen (V1 & V2)
>>> X = biopsy[['V1', 'V2']]
>>> y = biopsy[['class']]
>>> model = LinearDiscriminantAnalysis()
>>> model.fit(X,y)
# Stap 3: confusion matrix
>>> predicted = model.predict(X)
>>> conf_matrix = confusion_matrix(y_true=y,
                                   y_pred=predicted, labels = y.unique())
# Stap 4: bereken de metrieken
>>> accuracy_score(y_true=y, y_pred=predicted)
>>> precision_recall_fscore_support(y_true=y,
                                   y_pred=predicted, beta=1.0)
# Stap 5: Plot de ROC curve
>>> y_true = y
>>> y_score = model.predict_proba(X)[:,:0]
>>> plot_roc(y_true, y_score, pos_label='benign')
```

Lda is een binaire classifier
(zie later)

Beste threshold-waarde
threshold = 0.95
TPR = 0.94
FPR = 0.04

Area Under the Curve (AUC) is 0.987

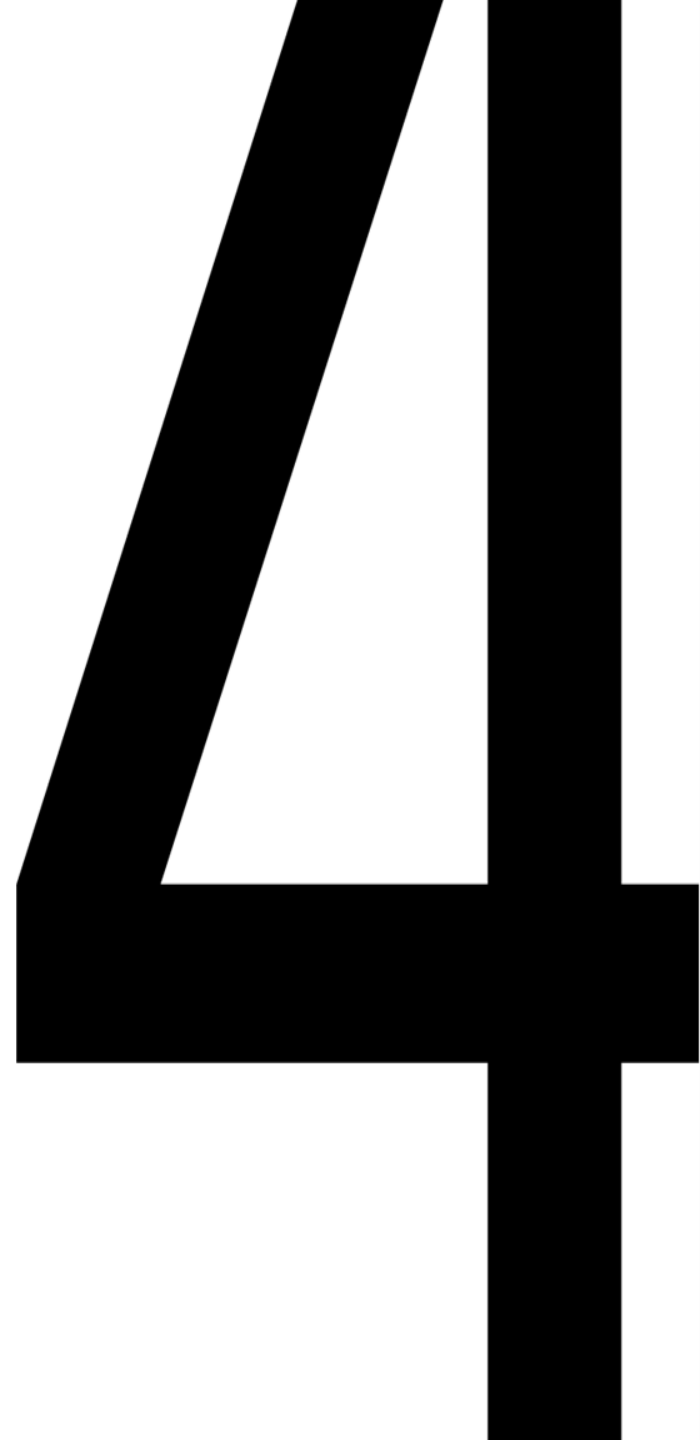
ROC voor een "willekeurige" binaire classifier





**BUT DAD, THAT IS THE MOST SEARCHED
KEYWORD ON SEARCH ENGINES...**

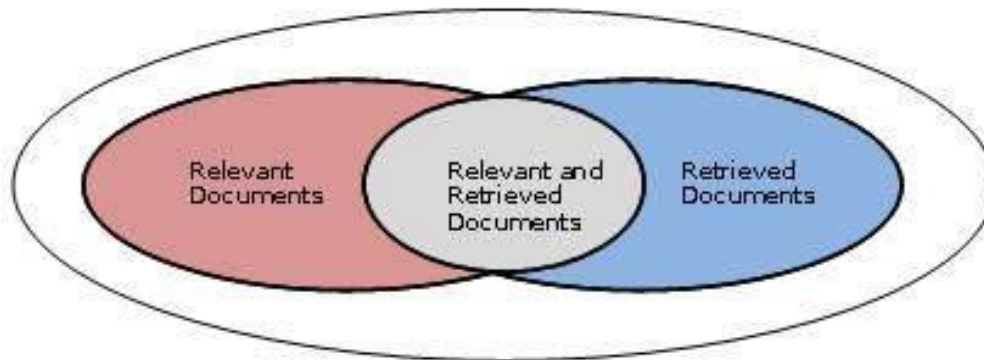
In de IT praktijk



Information retrieval

Het opzoeken van informatie in een informatie systeem wordt aangeduidt met de term "**Information retrieval**". Het betreft o.a. het zoeken naar documenten, naar specifieke informatie in documenten, naar metadata dat informatie beschrijft, ...

Evaluatiemetrieken worden hier gebruikt om in te schatten in welke mate het resultaat van de opzoeking overeenkomt met de zoekintentie van de gebruiker.



Hierbij wordt een onderscheid gemaakt tussen 'online metrieken' en 'offline metrieken'.

Information retrieval

'Online metriecken' :

Deze meten de user interactie met het informatie systeem.

- Session abandonment rate
- Click-through rate (CTR)
- Session success rate
- Zero result rate (ZRR)

'Offline metriecken':

Deze meten de relevantie. Komt het zoekresultaat overeen met de informatie verwachtingen van de gebruiker?

- Precision
- Recall
- Fall-out
- F-measure
- Average precision
- Precision at K
- R-precision
- Mean average precision
- Discounted cumulative gain
- ...

Managed Security Services (MSS)

Managed security services (MSS) is a systematic approach to managing an organization's security needs. The services may be conducted in house or outsourced to a service provider that oversees other companies' network and information system security. Functions of a managed security service include round-the-clock monitoring and management of **intrusion detection** systems and **firewalls**, overseeing **patch management** and upgrades, performing security assessments and **security audits**, and responding to emergencies.

Bron: <https://searchmidmarketsecurity.techtarget.com/definition/managed-security-services>

Voorbeelden:

Semantic:

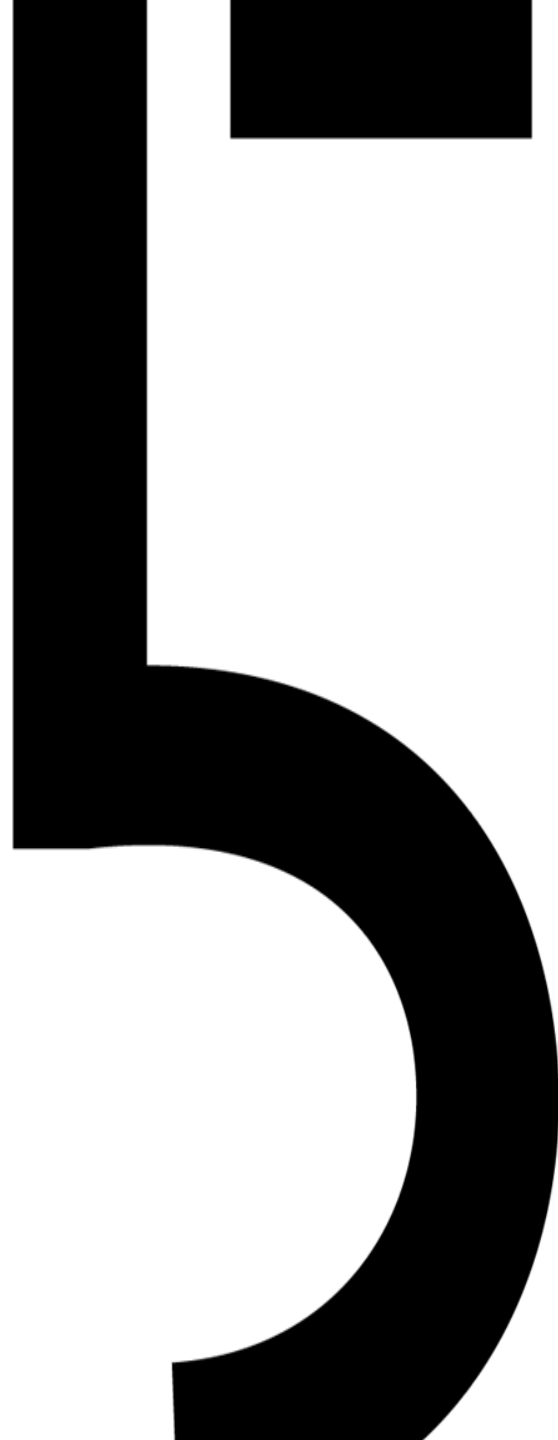
<https://www.birger.technology/sites/default/files/articles/pdf/datasheet-symantec-managed-security-services.pdf>

Verizon:

https://enterprise.verizon.com/service_guide/reg/cp_mssccloud_plus_mngd_security_svcs_cloud.pdf

Merk op: er is steeds sprake van false positives, maar false negatives worden niet vermeld.

In de media



Fatale crash Uber-auto ontstond doordat software overstekende voetganger niet zag

NS | 06 november 2019 | 19u50 | Bron: AD.nl - Tweakers



© AP - Auto's van Uber worden onderzocht na de fatale crash van een vrouw die oversteekte.

AUTO De zelfrijdende Volvo XC90-auto van Uber die in maart vorig jaar in de VS betrokken was bij een fataal ongeluk met een voetganger, had software aan boord die onvoldoende in staat was om rekening te houden met voetgangers die op onbedoelde plekken de weg oversteken.

De Amerikaanse National Transportation Safety Board (NTSB) meldt in een vrijgegeven document dat de Uber-auto niet de software aan boord had om te reageren op voetgangers die aan 'jaywalking' doen, oftewel voetgangers die op 'onwettige wijze' een weg oversteken op een plek waar bijvoorbeeld geen zebrapad of oversteekplaats is.

Binnen twee weken publiceert de NTSB zijn volledige rapport over de crash van 18 maart 2018, waarbij in het Amerikaanse Tempe (Arizona) een 49-jarige vrouw om het leven kwam toen ze in het donker de weg overstak op een plek waar geen overgang was. Ze had een fiets aan de hand toen ze weg overstak.

'Overige'

Dat laatste was bleek ook een noodlottig element. De Volvo XC90 reed 70 km/u toen de vrouw werd aangereden. Toen de auto haar detecteerde, op 5,6 seconden voor de impact, werd de vrouw als een voertuig aangeduid. Vervolgens veranderde het brein van de auto dat in 'overige' en weer terug naar voertuig; dat patroon herhaalde zich nog twee keer. Pas op 1,2 seconden voor de crash herkende het systeem de situatie correct.

Met die resterende 1,2 seconden speling had het systeem de remmen kunnen activeren, maar technici van Uber bleken een ingebouwd, automatisch remsysteem van Volvo te hebben uitgeschakeld. De automaker gaf eerder aan dat dat systeem het ongeluk wellicht zelfs had kunnen voorkomen. Uber-technici hadden dit uitgeschakeld, omdat het onveilig zou zijn als de auto twee actieve softwaresystemen zou hebben die parallel aan elkaar opereren.

Vertraging

Het NTSB-rapport bevestigt verder dat er een vertraging van een seconde in de software was ingebouwd tussen het detecteren van een situatie waarin een crash waarschijnlijk is en het reageren daarop. Uber had dit gedaan, omdat het bedrijf bang was voor teveel 'false positives'. Tijdens de 1,2 seconden voor de impact leidde dat ertoe dat de auto niet de remmen activeerde, omdat het de aard van het gevaar eerst verifieerde. De aanwezige bestuurder van de Volvo keek op dat moment niet op de weg. Met 0,2 seconden te gaan gaf de auto een geluidswaarschuwing, waarna de bestuurder het stuur ter hand nam, maar ze remde pas een seconde nadat de botsing al had plaatsgevonden.

In het onderzoeksrapport staat dat Uber inmiddels veranderingen heeft doorgevoerd aan zijn veiligheidsprogramma. Verder heeft het bedrijf de training van veiligheidschauffeurs aangescherpt en rijden er nu steeds twee personen in de zelfrijdende auto's.

Lees ook



Chauffeur keek The Voice tijdens fatale aanrijding met zelfrijdende Uber-auto

IT-securityoorlog

"Antivirusbedrijf Kaspersky saboteert concurrenten": CEO ontkent

Het Russische antivirusbedrijf Kaspersky zou geprobeerd hebben concurrenten schade te berokkenen door hun antivirussoftware te misleiden. Daardoor werden goedaardige bestanden als kwaadaardig beschouwd, zogenoemde 'foute positieven' (false positives). Dat meldt

het persagentschap Reuters op zijn publieke website, op basis van twee anonieme ex-werknemers van Kaspersky. Het bedrijf zelf ontkent ten stelligste.

14 augustus 2015

Bron: <https://www.demorgen.be/tech-wetenschap/antivirusbedrijf-kaspersky-saboteert-concurrenten-ceo-ontkent~b521f5c5/>

Volgens de ex-werknemers was de geheime campagne, die een decennium geleden zou zijn begonnen, gericht tegen Microsoft Corp, AVG Technologies, Avast Software en andere rivalen. Sommige van hen zouden misleid zijn en verwijderden belangrijke bestanden op de computers van hun klanten, of maakten die bestanden ontoegankelijk, terwijl die niet schadelijk waren.

Enkele van die aanvallen zouden bevolen zijn door de co-stichter van Kaspersky Lab, Eugene Kaspersky, gedeeltelijk als vergelding op kleinere concurrenten. Volgens hem aapten die bedrijven Kaspersky-software na. "Eugene beschouwde dit als stelen", zei een van de ex-werknemers aan Reuters. Beide bronnen vroegen anonimiteit. Volgens hen weerspiegelde de keuze van de te saboteren concurrenten ook de wens om marktaandeel op te bouwen.

Het bedrijf zelf ontkent de beschuldigingen in een verklaring aan het persagentschap. "Zulke daden zijn onethisch, oneerlijk en de legaliteit ervan is op zijn minst twijfelachtig", klinkt het.

Eugene Kaspersky zelf reageerde heftig op Twitter. "Gewoonlijk lees ik Reuters niet", klonk het. "Maar wanneer ik dat doe, zie ik foute positieven." Hij doet het verhaal af als complete onzin.

Kaspersky is een van de grootste antivirusbedrijven en stelt meer dan 3.000 specialisten te werk. Het bedrijf stelt 400 miljoen gebruikers en 270.000 bedrijfsklanten te hebben.

In de medische literatuur wordt volgende terminologie gebruikt:

- **Sensitiviteit** = recall = TPR
- **Specificiteit** = 1-FPR

‘Hartscreening niet aanbevolen bij alle jongeren’

04/02/2013 om 16:12 door loa | Bron: Belga

De Hoge Gezondheidsraad vindt het niet aangewezen om alle jongeren een harttest te laten ondergaan. Dat zegt hij in een advies aan de federale regering. De raad ziet wel enig nut in een screening van jonge sporters.

Aanleiding van het advies is het aanhoudende pleidooi van cardioloog Pedro Brugada om alle twaalfjarigen verplicht een harttest te laten ondergaan om zo een plotse dood te voorkomen.

De Hoge Gezondheidsraad vindt een dergelijke screening niet nodig. Volgens de leden beschikt men momenteel niet over een screeningsmethode met voldoende sensitiviteit en specificiteit om de verscheidenheid aan zeldzame hartafwijkingen op te sporen.

De raad is ook bang dat te veel jongeren onterecht doorverwezen zouden worden. ‘Naast de nadelige psychologische effecten die hiermee kunnen gepaard gaan, is te verwachten dat de kosten-baten verhouding van een dergelijk programma ongunstig is, zeker indien herhaaldelijk uitgevoerd’, klinkt het.

Jonge sporters

Een systematische screening van sportende jongeren (tussen 15 en 35 jaar oud) vindt de Hoge Gezondheidsraad wel gunstig. ‘Het risico op een plotse dood wordt immers veel groter wanneer iemand met een hartafwijking sport’, klinkt de redenering.

Al volgt meteen de nuance: de raad oordeelt dat ‘de wetenschappelijke evidentie’ voor een dergelijke screening momenteel ‘onvoldoende eenduidig’ is, maar dat er wel een maatschappelijk en/of professioneel draagvlak aanwezig is om een systematische screening in overweging te nemen.

Het zou daarbij aangewezen zijn om niet alleen naar hartafwijkingen te zoeken maar ook andere aspecten te bekijken die van belang zijn voor een kandidaat-sporter. ‘Dit moet worden uitgevoerd door een arts met bijzondere bekwaamheid in sportkeuring en onder bepaalde voorwaarden’, aldus de leden.

‘Geen garantie’

De raad vraagt daarbij om screening voldoende te duiden bij de bevolking. ‘Ook al is screening een hulpmiddel om zoveel mogelijk mensen op een zo veilig en gezond mogelijke manier aan sport te laten doen, biedt hij geen garantie dat alle sportgerelateerde aandoeningen of overlijdens zullen voorkomen worden’, klinkt het. De raad pleit voor een sensibiliseringscampagne om de screening zo veel mogelijk bekend te maken en aan te raden. ‘Sportfederaties hebben daarbij een grote verantwoordelijkheid.’

Tot slot zegt de Hoge Gezondheidsraad dat er nood is aan meer wetenschappelijk onderzoek naar de oorzaak en preventie van plotse dood bij jongeren. De gegevens van de screenings zijn daarbij essentieel.

Slechts de helft van de rusthuisbewoners met ziektesymptomen blijkt besmet met covid-19. Hebben we de overlijdens als gevolg van corona in die woonzorgcentra overschat?

Aantal gerapporteerde patiënten in België

	Op 15 april	In totaal
Bevestigde covid-19 gevallen	2.454	33.573
Sterfgevallen	283	4.440
• in ziekenhuizen	103	2.252
• in woonzorgcentra	179	2.029
• Bevestigde gevallen	5	63
• Mogelijke gevallen	174	1966

Resultaten van de testen op covid-19 in woonzorgcentra

Bij het personeel

	Aantal testen	Aantal positieve testen	% positieve testen
België	Totaal 5.669	790	14
met symptomen	541	206	38
zonder symptomen	5.128	584	11
Vlaanderen	Totaal 2.129	346	16
met symptomen	224	92	41
zonder symptomen	1.905	254	13

Bij de bewoners

	Aantal testen	Aantal positieve testen	% positieve testen
België	Totaal 5.202	1.047	20
met symptomen	657	313	48
zonder symptomen	4.545	734	16
Vlaanderen	Totaal 4.932	1.000	20
met symptomen	588	289	49
zonder symptomen	4.344	711	16

DS Infografiek | Bron: Sciensano

VAN ONZE REDACTRICE VEERLE BEEL

BRUSSEL | Eergisteren overleden er 283 landgenoten door covid-19. Van hen stierven er 103 in het ziekenhuis en 179 in een woonzorgcentrum. Weliswaar konden maar vijf overlijdens in woonzorgcentra bevestigd worden door een positieve covid-19-test. 174 andere overlijdens konden enkel vermoedelijk in verband worden gebracht met de epidemie.

Alles bijeen zijn sinds het begin van de coronacrisis al 2.029 overlijdens in woonzorgcentra in verband gebracht met de epidemie. Daarbij waren er maar 63 bevestigd met een positieve covid-19-test.

tigd met een positieve covid-19-test. Verrassend, maar toch ook weer niet, aangezien de woonzorgcentra tot begin april nauwelijks over tests konden beschikken. Ook daarna kwam het testen maar mondjesmaat op gang. De richtlijn van het agentschap Zorg en Gezondheid luidde dat indien een of enkele bewoners positief hadden getest, ook alle andere bewoners met gelijkaardige symptomen als covid-19-positief mochten worden beschouwd.

Niet volledig betrouwbaar

Bij die strategie worden nu vragen gesteld. Want de voorbije

dagen is op grote schaal getest in 85 woonzorgcentra en daaruit blijkt dat slechts de helft van de bewoners met covid-achtige symptomen ook daadwerkelijk positief test. De andere helft is dus, ondanks bepaalde ziektesymptomen, niet besmet. Heel wat van die mensen zijn wellicht in strenge quarantaine geplaatst, terwijl het niet nodig was.

Om het helemaal ingewikkeld te maken, dat laatste is toch niet helemaal zeker. Professor Jan Delepeleire (KU Leuven) waarschuwt dat de covid-19-test niet honderd procent betrouwbaar is. 'De test geeft tot 30 procent vals-negatieve resultaten. Er is zeker iets aan de hand in de woonzorgcentra, maar de ware omvang van deze epidemie zullen we pas kennen als we weten hoe groot de oversterfte is geweest.'

Andere landen nemen de overlijdens in hun ouderenzorginstellingen niet mee in hun officiële cijfers van sterfte door het coronavirus. Zij scoren daardoor verhoudingsgewijs veel lager dan België. Binnen de regering leeft daar ongerustheid over. In het buitenland vraagt men zich af wat er in ons land aan de hand is. Het kan problemen geven als we straks de landsgrenzen willen heropenen. De kans bestaat dat onze buurlanden op de rem gaan staan.

Meest volledige registratie

Viroloog Marc Van Ranst chargeerde eerder deze week toen hij in een tv-programma stelde dat zowat elk overlijden in een woonzorgcentrum nu als een covid-19-gerelateerd overlijden wordt doorgegeven. 'Ook wanneer iemand overlijdt aan een hartstilstand.'

Ook de Vlaamse minister van Welzijn, Wouter Beke (CD&V), vermoedt nu dat de sterftecijfers omtrent covid-19 in de woonzorgcentra overtrokken zijn. Hij pareerde daarmee de kritiek van de oppositie dat er in de woonzorgcentra te laat is gereageerd op de coronacrisis. 'Er is maar één duidelijke indicator en dat is te kijken naar de oversterfte.'

Huisarts Gijs Van Pottelberg, coördinerend arts in een woonzorgcentrum en ook verbonden aan de KU Leuven, denkt dat er zowel een overschatting als een onderschatting plaatsvindt. 'Er overlijden elk jaar een aantal bewoners aan een longontsteking. De vraag is: hoeveel meer zijn het

Hoeveel covid-19-doden telt België echt?



Slechts 63 van de 2.029 covid-19-overlijdens in woonzorgcentra werden

bevestigd met een positieve test. © pti

De cijfers kunnen problemen geven als we straks de grenzen willen heropenen. De kans bestaat dat buurlanden op de rem gaan staan

'In centra waar de besmetting al is opgedoken, zal de arts sneller covid-19 noteren op het overlijdensformulier. In centra zonder besmetting zal men dat niet doen. Het kan in beide gevallen fout zijn'

GIJS VAN POTTELBERG
Coördinerend arts
in een woonzorgcentrum

er dit jaar? In woonzorgcentra waar de besmetting al is opgedoken, zal de arts sneller covid-19 mee noteren op het overlijdensformulier. In centra zonder vastgestelde besmetting zal men dat niet doen. Het kan in beide gevallen fout zijn.'

Van Pottelberg zegt ook dat er wel degelijk nog bewoners aan andere oorzaken overlijden. Viroloog Steven Van Gucht, van Sciensano, blijft erbij dat het belangrijk is om zicht te behouden op wat er in de woonzorgcentra aan de hand is: 'We zijn het land met de meest volledige registratie. We doen dat niet om onszelf te vergelijken met andere landen.

We doen het om goed in kaart te brengen wat er aan de hand is. Zo dat we ook oplossingen kunnen zoeken.'

Niet ziek, toch besmet

Wat de tests bij 85 woonzorgcentra met een uitbraak ook hebben aangetoond, is dat er evenveel bewoners en medewerkers besmet zijn die geen symptomen vertonen. Zij maken het virus door zonder er iets van te merken. Het betreft in België 16 procent van de bewoners en 11 procent van de medewerkers.

Alles bijeen is in die 85 centra 14 procent van de medewerkers besmet, en 20 procent van de be-

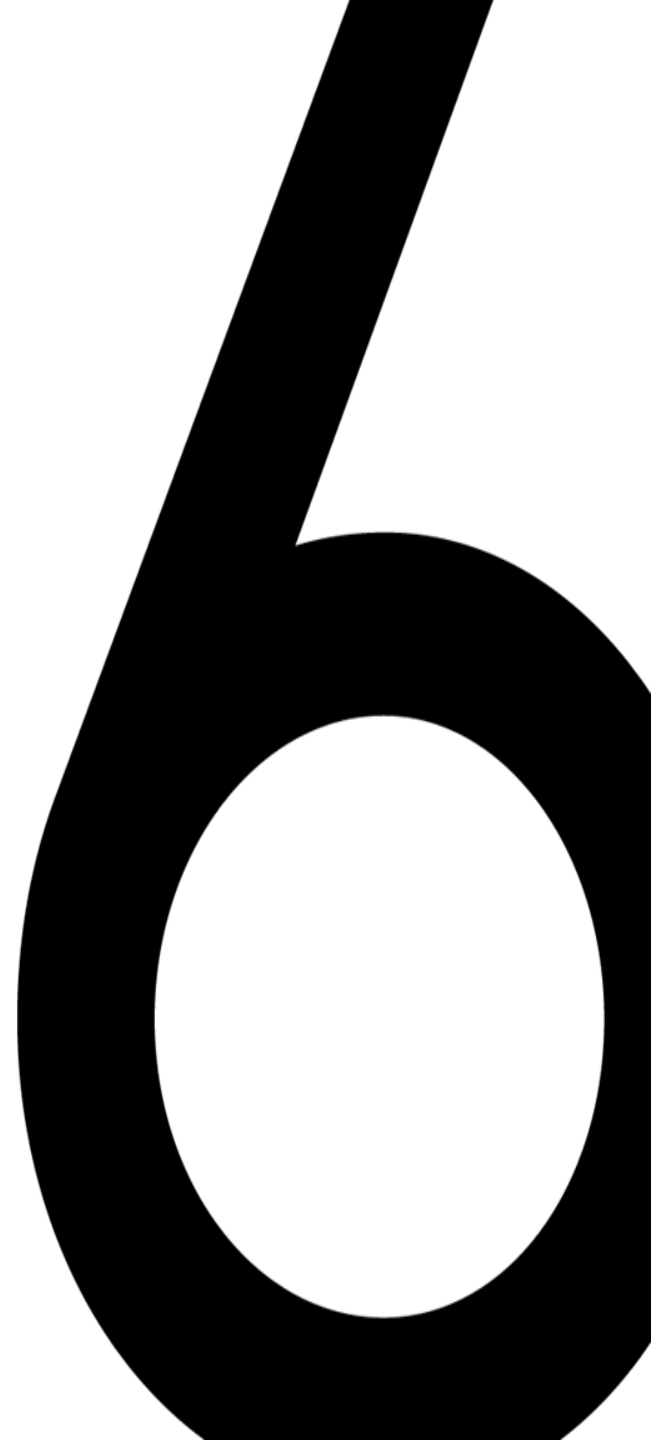
woners (zie tabel). 'Dat er ook mensen positief hebben getest die zich niet ziek voelen, sterkt ons in de overtuiging dat verder testen nodig is', zegt Karine Moykens, de voorzitter van de taskforce zorg. 'Een correct gebruik van het beschermingsmateriaal en goede handhygiëne blijven ook uitermate belangrijk.'

De 120.000 tests die minister Philippe De Backer (Open VLD) bijkomend aan Vlaanderen ter beschikking stelt, worden behalve in de ouderenzorg, ook ingezet in de revalidatiesector, de psychiatrische verzorgingshuizen en voorzieningen met kwetsbare personen met een handicap.

De Standaard-
16 april 2020



Vragenlijst



Vragenlijst



- Download het bestand *vragenlijst 21-22.xlsx* van Canvas
- Exporteer het excel-bestand als een csv bestand
- Plaats '*vragenlijst 21-22.csv*' in je Python workspace
- Lees de data in en plaats het in het dataframe ***studenq***

```
>>> import pandas as pd
>>> studenq = pd.read_csv('vragenlijst 21-22.csv', delimiter=';',
decimal='.')
```

Vragenlijst



Vraag 1.: We willen gaan voorspellen of studenten heel gemotiveerd zijn om een diploma Toegepaste Informatica te behalen.

A. De onafhankelijke gegevens (de voorspellers) zijn "het aantal broers en/of zussen", "aantal samenzweringen" en "het aantal uren wiskunde". Plaats de onafhankelijke variabelen in een nieuw dataframe met naam **df** en zorg ervoor dat de waarden van de voorspellers numeriek zijn en dat eventueel onrealistische gegevens worden omgezet in nan-waarden.

Vragenlijst



B. In de vragenlijst wil je het "belang van informatica" gaan voorspellen. Dit is een ordinaal gegeven dat je zal moeten reduceren tot twee mogelijke waarden: alles gelijk of meer dan "groot" is "groot", alles kleiner is "matig". Voeg het resultaat als een kolom toe aan het dataframe ***df***

Vragenlijst



C. Gebruik lineaire discriminant analyse als model om de afhankelijke variabele "belang van Informatica" te voorspellen met behulp van de onafhankelijke variabelen (voorspellers). Maak de voorspellingen voor de beschikbare gegevens.



D. Bestudeer de resultaten van de evaluatie metrieken:

- Maak de confusion matrix
- Bereken de accuracy, precision, recall en F1
- Maak de ROC-curve

