



8QA01 OGO Beeldanalyse

COLLEGE 2

M.H.A. Janse

IMAG/e group, department of Biomedical Engineering

TU/e EINDHOVEN
UNIVERSITY OF
TECHNOLOGY

1

One-minute paper

Naam docent: Mark Janse

Datum: 09-12-2022

Onderdeel: 8QA01 college 2

Graag invullen/inleveren na einde college;

TU/e

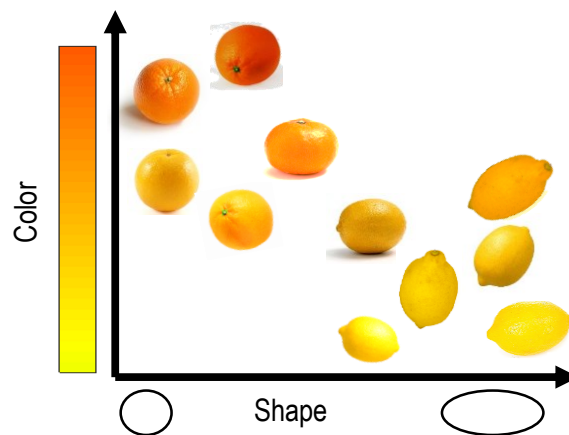
2

Programma voor vandaag

- Machine learning en classificatie
- Evaluatie van het algoritme
- Uitleg over de eindopdracht

TU/e

3



Slide courtesy Fons van der Sommen

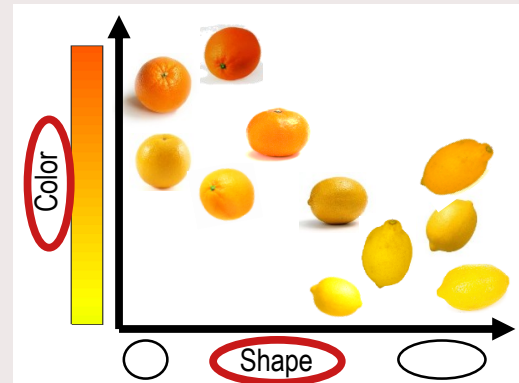
TU/e

4

De tussenopdracht

Tot nu toe hebben we de features gemeten

Hoe komen we nu naar een voorspelling?



5

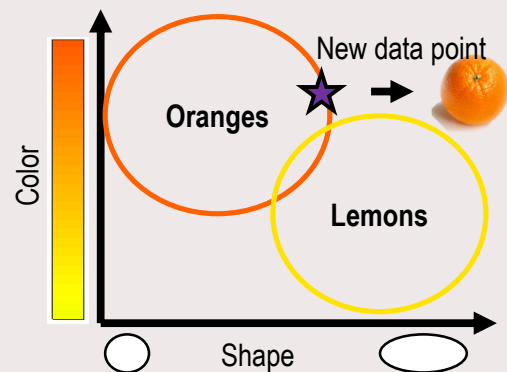
TU/e

5

De tussenopdracht

Tot nu toe hebben we de features gemeten

Hoe komen we nu naar een voorspelling?



6

TU/e

6

Classificatie

In statistics, **classification** is the problem of identifying which of a set of categories (sub-populations) an observation, (or observations) belongs to. Examples are assigning a given email to the "spam" or "non-spam" class, and assigning a diagnosis to a given patient based on observed characteristics of the patient (sex, blood pressure, presence or absence of certain symptoms, etc.).

Bron: wikipedia, statistical classification

7

TU/e

7

Classificatie binnen machine learning

Simpel gezegd: hoe kunnen we een computer leren, nieuwe data in te delen in categorieën, op basis van voorbeeld data.

Voor ons project: hoe kunnen we, op basis van de features, kijken of een vlekje een melanoom, wrat (keratose) of niets bijzonders is, als we de computer voorbeelden geven.

8

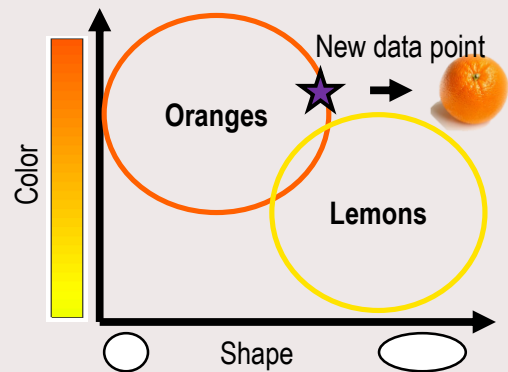
TU/e

8

Vele manieren om dit te doen!

1. Enkelvoudige drempelwaarde
2. Lineair model
3. K-nearest neighbors
4. SVM
5. *Neurale netwerken*

Nog vele duizenden manieren om dit te doen! Alleen de belangrijkste voor dit project hier



9

TU/e

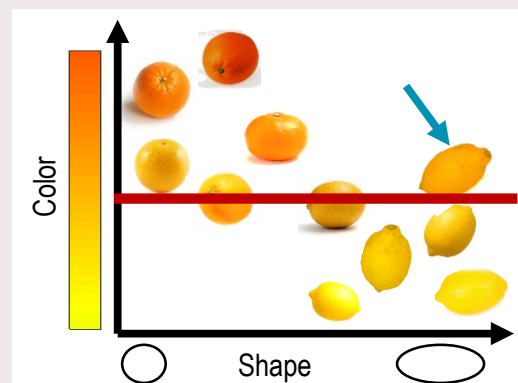
9

1. Enkelvoudige drempelwaarde

Kies één feature, kies een afkappunt. Alles kleiner dan het afkappunt is categorie 1, alles groter is categorie 2

Voordeel: erg simpel

Nadeel: niet erg precies



10

TU/e

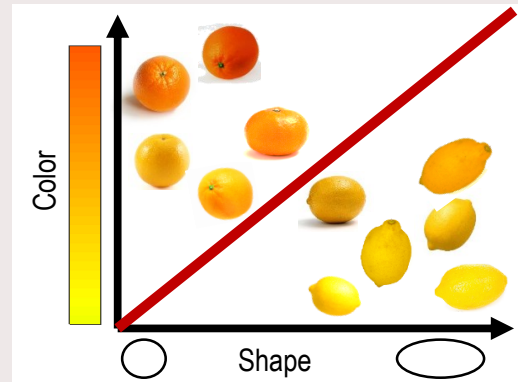
10

2. Lineair model

Trek een lijn tussen een combinatie van features, zodanig dat je de beste scheiding tussen categorieën hebt

Voordeel: nog steeds simpel, maar je neemt meerdere features mee

Nadeel: meeste 'real-world' data is niet zo makkelijk te verdelen dus niet altijd bruikbaar



11

TU/e

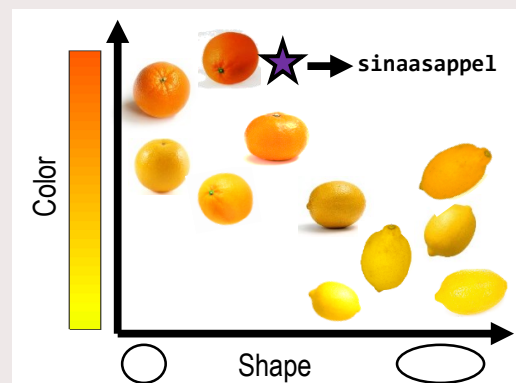
11

3a. Nearest neighbor

Kies uit bestaande data de categorie waar het nieuwe datapunt het dichtste bij zit

Voordeel: nog steeds simpel, veel krachtiger dan een lineair model

Nadeel: erg gevoelig voor ruis in de voorbeeld (training) data



12

TU/e

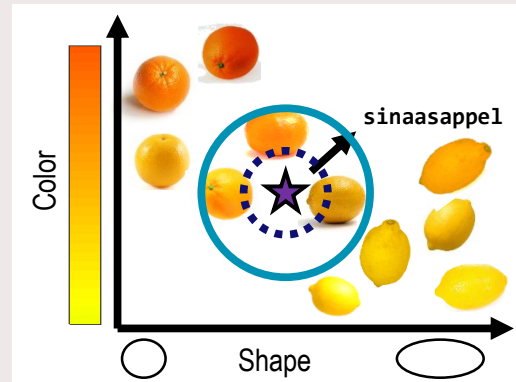
12

3b. k-Nearest neighbor

Kies uit bestaande data de categorie met minimaal k datapunten het dichtste bij.

Voordeel: minder gevoelig voor ruis in de training data dan normale nearest neighbors

Nadeel: goede waarde voor 'k' kiezen erg lastig



13

TU/e

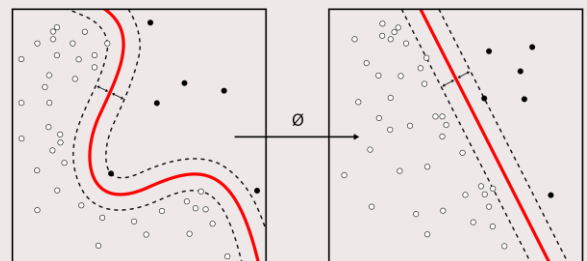
13

4. Support Vector Machine (SVM)

Trekt een lijn de categoriën maximaal uit elkaar haalt

Voordeel: een van de beste algoritmes, kan ingewikkelde distributies wel uit elkaar halen

Nadeel: veel parameters die je kan veranderen; kan soms erg goed fout gaan ('overfitting')



Redelijk geavanceerd, niet verplicht voor dit vak!

14

TU/e

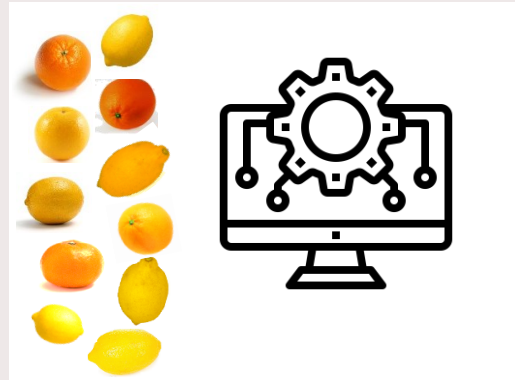
14

5. Neurale netwerken / Artificial Intelligence / Deep Learning

Stop alles in de computer en laat die het lekker uitzoeken.

Voordeel: geen features nodig!

Nadeel: ontzettend veel data nodig;
als het fout gaat, kan het gigantisch
fout gaan (voorbeelden college 1);
Niet te herleiden *waarom*
beslissingen worden genomen



Gaan we in dit vak niet doen

Icon: flaticon.com

15

TU/e

15

Kort samengevat

Methode	Wat het doet
1. Enkelvoudige drempelwaarde	Neem één feature, bepaal een afkappunt tussen categorieën
2. Linear model	Trek een lijn door je feature space
3a. Nearest neighbor	Neem de categorie van het dichtstbijzijnde bekende punt
3b. k-Nearest neighbor	Neem de categorie waarvan k punten het dichtste bij zitten
4. SVM	Trek een - niet perse rechte - lijn die de categorieën maximaal uit elkaar haalt
5. Neurale netwerken	Laat de computer alles zelf uitzoeken



16

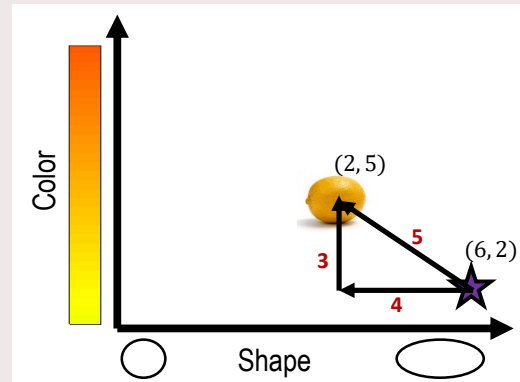
TU/e

16

Nearest neighbor: afstand uitrekenen

Veel manieren om de afstand te berekenen! Meest gebruikte is de *'Euclidische afstand'*.
Uitrekenen met de stelling van Pythagoras

$$\sqrt{(6-2)^2 + (5-2)^2} \\ = \sqrt{16 + 9} = 5$$



17

TU/e

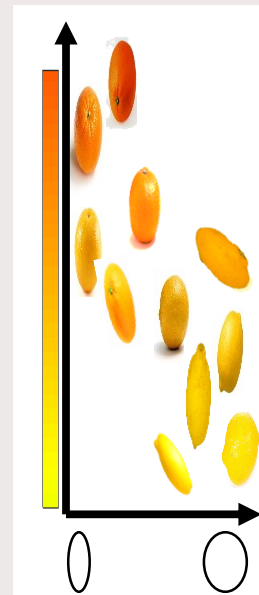
17

Normalisatie van de features

Verschillende features hebben verschillende schalen:

- **Vorm:** 0-1
- **Kleur:** 0-255
- **Gewicht:** 120-250 gram
- **Compactness:** 1.0-infinite

Dit kan de afstands berekening beïnvloeden:
kleur is hierdoor veel belangrijker dan vorm



18

TU/e

18

Normalisatie van de features

Oplossing: *normaliseer* alle features tussen 0 en 1.

- **Vorm:** 0-1
- **Kleur:** 0-255
- **Gewicht:** 120-250 gram
- **Compactness:** 1-infinite

19

TU/e

19

Normalisatie van de features

Oplossing: *normaliseer* alle features tussen 0 en 1.

- **Vorm:** 0-1 → niks 😊
- **Kleur:** 0-255 → $\frac{kleur}{255}$
- **Gewicht:** 120-250 gram → $\frac{gewicht-120}{130}$
- **Compactness:** 1-infinite
 - Kies een maximum aan de hand van de data (wat zien we?)
 - Kap af op een bepaald percentiel als er een flink uitschiet (95^{ste} percentiel)

20

TU/e

20

Normalisatie van de features

Oplossing: *normaliseer* alle features tussen 0 en 1.

- Vorm: 0-
- Kleur: 0-
- Gewicht
- Compact
 - Kies een
 - Kap af op een bepaald percentiel als er een flink uitschiet (95^{ste} percentiel)

**Pas op met de normalisatie-
functies in Python!
Deze doen soms hele andere
dingen dan je denkt.**

21 8QA01 Part 1: Introduction

TU/e

21

Deel 2: Evaluatie

8QA01 COC 2

M.H.A. Janse

22

22

Evaluatie: bepalen hoe goed we zijn

Ons doel: evalueren of onze methode geschikt is voor huidlaesies te classificeren

Q1: Age	...	Q7: Color	New feature
20		1	1.7
25		1	2.5
40		2	1.3
70		3	0.1



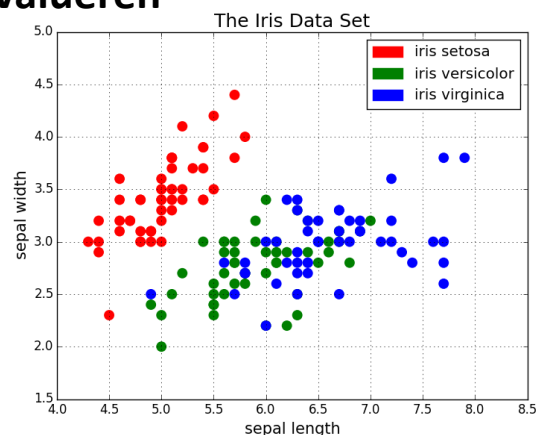
True label	Predicted label
0	0
1	1
0	0
1	0

TU/e

23

Manieren om features te evalueren

- Maak plotjes! Zijn er clusters?
- Kijk op simpele kunstmatige plaatjes: zijn de waarden logisch?
 - Bereken bijv. compactness van een cirkel; aantal kleuren in een zelf gemaakt plaatje
- Vergelijk met de handmatig gemeten features:
 - Is het gecorreleerd?
 - Veel interobserver afwijkingen?



24 8QA01 Part 1: Introduction

TU/e

24

Classificatie evalueren

Om te kijken hoe goed ons algoritme werkt, moeten we die beoordelen.

De meest simpele maat: *accuracy*
= percentage goed voorspeld.

In dit geval: $3 \div 4 \times 100\% = 75\%$

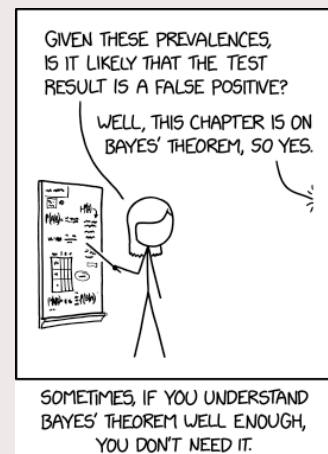
True label	Predicted label
0	0
1	1
0	0
1	0

Accuracy is vaak niet geschikt

In de *real world* zal onze data hoogstens 1% kanker hebben.

Alles voorspellen als gezond is dan 99% nauwkeurig 😊 Maar wel met 100% vals negatieven.

Oplossing: gebruik andere maten, zoals sensitiviteit, specificiteit, AUC enz.



Veel classifiers geven een kans ipv label

Als de classifier een kans voorspelt, kunnen we door te *thresholden* (drempelen) een label er van maken.

De threshold is een *hyperparameter*. Deze hoeft niet perse 0.5 te zijn.

Predicted probability		Predicted label	True label
0.2	$> 0.5 ?$	0	0
0.9		1	1
0.3		0	0
0.4		0	1

27 8QA01 Part 1: Introduction

TU/e

27

Confusion matrices

Een confusion matrix zet de voorspellingen uit tegenover de *ground truth*.

Het maakt het om de *sensitiviteit* en de *specificiteit* uit te rekenen!

Sensitiviteit: hoe groot is de kans dat ik vind wat ik zoek?

Specificiteit: hoe groot is de kans dat als ik het niet vind, het er niet is?

Voorspeld → Waarheid ↓	Kanker	Geen kanker
Kanker	95	5
Geen kanker	30	70

$$\text{Sensitiviteit: } \frac{95}{95 + 30}$$

28 8QA01 Part 1: Introduction

TU/e

28

Voorbeeldberekening

Sensitiviteit: hoe groot is de kans dat ik vind wat ik zoek?

Specificiteit: hoe groot is de kans dat als ik het niet vind, het er niet is?

Tip: het Engelse Wikipedia artikel hierover is érg goed!

globale

Algoritme → Waarheid ↓	Kanker	Geen kanker
Kanker	95	5
Geen kanker	30	70

$$\text{Sensitiviteit: } \frac{95}{95+5} = 95\%$$

$$\text{Specificiteit: } \frac{70}{70+30} = 70\%$$

29 8QA01

TU/e

29

Andere dingen om naar te kijken als het niet werkt

- Kijk zelf naar plaatjes die fout gaan, zijn er patronen?
- Bepaalde soorten plaatjes die altijd fout gaan?
- Wat gebeurt er als je een feature weg laat? Of juist toevoegt?

30 8QA01

TU/e

30

Andere dingen om te overwegen

- Durven mensen het algoritme wel te gebruiken?
- Is het in de praktijk wel bruikbaar zoals wij denken?
- Hoe is de gezondheidswinst in de praktijk?

31 8QA01

TU/e

31

Deel 3: Trainen/testen/validatie

M.H.A. Janse

IMAG/e group, department of Biomedical Engineering

TU/e EINDHOVEN
UNIVERSITY OF
TECHNOLOGY

32


```

47 def knn_classifier(X_train, y_train, X_validation, X_test, k):
48     # Returns the labels for test_data, predicted by the k-NN classifier trained on X_train and y_train
49     # Input:
50     # X_train - num_train x num_features matrix with features for the training data
51     # y_train - num_train x 1 vector with labels for the training data
52     # X_validation - num_test x num_features matrix with features for the validation data
53     # X_test - num_test x num_features matrix with features for the test data
54     # k - Number of neighbors to take into account
55     # Output:
56     # y_pred_validation - num_test x 1 predicted vector with labels for the validation data
57     # y_pred_test - num_test x 1 predicted vector with labels for the test data
58
59     X_test_val = np.vstack((X_validation, X_test))
60
61     # Compute standardized euclidian distance of validation and test points to the other points
62     D = cdist(X_test_val, X_train, metric='seuclidean')
63
64     # Sort distances per row and return array of indices from low to high
65     sort_ix = np.argsort(D, axis=1)
66

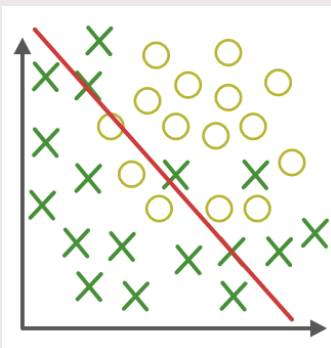
```

TU/e

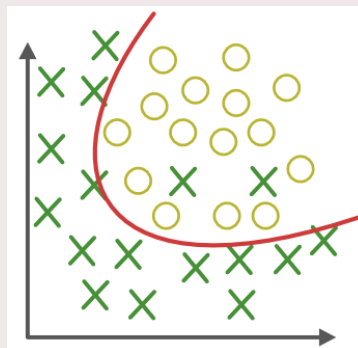
33

Goede en slechte modellen

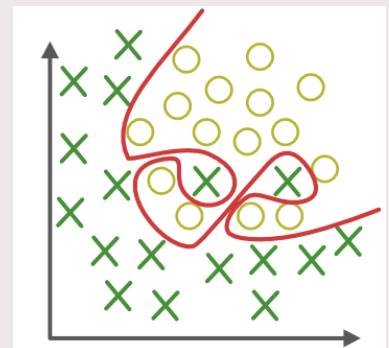
Onderfitten



Goede fit



Overfitten

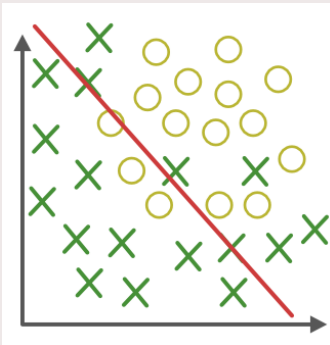


TU/e

34

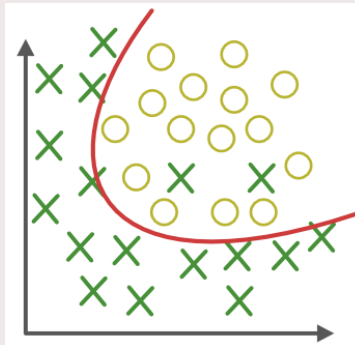
Meten met een testset

Onderfitten



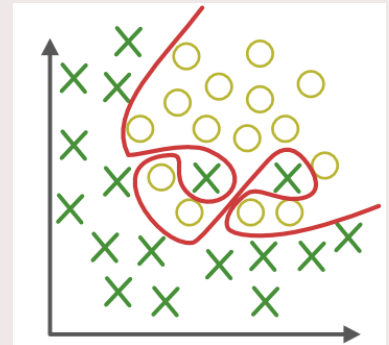
Training accuracy: 70%
Validatie dataset acc.: 70%

Goede fit



Training accuracy: 90%
Validatie dataset acc.: 90%

Overfitten



Training accuracy: 100%
Validatie dataset acc.: 40%

35 8QA01

TU/e

35

Hoe komen we aan zo'n extra set?

- Losse onafhankelijke dataset
- Cross-validatie
 - Verdeel dataset in blokjes, gebruik 1 blokje om te testen en de rest om te trainen

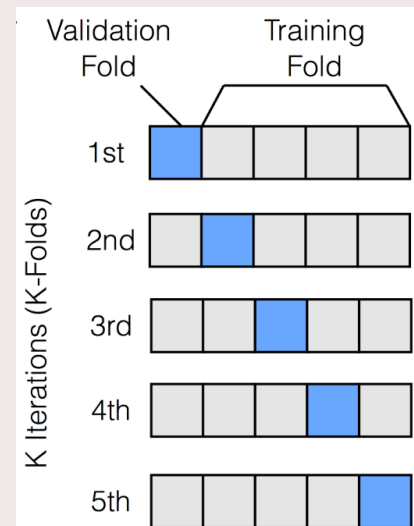
36 8QA01

TU/e

36

Cross-validation

Verdeel de data in k delen, laat steeds een deel er uit als validatie set, en herhaal het experiment k keer.



37 8QA01

TU/e

37

En wat over de test set?

Ons model heeft wat parameters die we kunnen tweak: bij k-NN bijvoorbeeld het getal k .

Als we veel verschillende k proberen, is er een kans dat we een model vinden dat goed werkt op onze validatie set, maar weer niet op een nieuwe, ongeziene set.

Daarom wordt er ook vaak een extra set gebruikt om op het einde op te testen, dit is de test set. Deze wordt maar 1x gebruikt.

In de voorbeeldcode van ons, worden de validatie- en testset als één ding gebruikt. Voor dit project is het niet heel belangrijk dit goed te doen.

38 8QA01

TU/e

38

Deel 4: Eindopdracht

8QA01 COC 2

M.H.A. Janse

39

39

Eindopdracht

Twee opdrachten: verslag (33.3%) en een presentatie (16.7%).
(peer review en tutor beoordeling tellen allebei 25% mee)

Verslag: Wetenschappelijk, 10 A4'tjes excl. titelpagina/bijlagen.

Presentatie: 5 minuten, populair wetenschappelijk

40

Eindverslag

Het bevat de standaardonderdelen:

- **Introductie:** hoe, wat en waarom; Onderzoeksvraag.
- **Methoden:** globale uitleg, gedetailleerde uitleg over features en voorspelling; info over **experimenten:** hoe hebben jullie getest of de methode nauwkeurig is?
- **Resultaten:** hoe nauwkeurig is het?
- **Discussie:** interpreteer de resultaten; Vat samen hoe (goed) het werkt
- **Appendix:** evt andere resultaten. *Geen code of Excel!*

41 8QA01

TU/e

41

Eindverslag tips

• Bronvermeldingen!

- Inhoud over vorm: besteed zo min mogelijk tijd aan de opmaak
- Het publiek is 1^e-jaars BMT studenten die het vak nog moeten doen, leg terminologie dus uit
- Wees consistent met afkortingen en dergelijke
- Neem de hand-out 'Verslag schrijven' door (van de vorige OGO)

Verdere info staat op de Github en de casusinfo.

42 8QA01

TU/e

42

Eindpresentatie

Lengte: 5 minuten presentatie + 4 minuten discussie = 9 min

Vorm: Alles mag! Toneelstuk, pubquiz, filmpje, presentatie

Publiek: Mogen jullie zelf bepalen! Mogen dermatologen zijn, bejaarden, studenten, vakantiegangers. Géén vakgenoten.

Denk goed na over de boodschap die je over wilt brengen.

Maak het memorabel met een beetje humor, maar wel begrijpelijk :)

Vergeet niet de **bronvermeldingen** in de credits! (telt niet mee voor de tijd)

Meer tips? Zie de handouts op canvas, en natuurlijk de github.

43 8QA01

TU/e

43

Belangrijke data

Vrijdag 13 januari om 13:00: verslag inleveren op Canvas en **2x op papier!**

Dinsdag 17 januari om 13:30: Eindpresentaties!

Woensdag 18 januari: Peer review en tutorbeoordeling af

Donderdag 19 januari vanaf 9:40: OGO nabespreking + beoordeling

44 8QA01 Part 1: Introduction

TU/e

44