

Naive Bayes algorithm

OPGAVE 1A

Het programma, dat bijgeleverd is, gebruikt de Naive Base classifier om te bepalen of de gegeven situatie resulteert in het wél of niet spelen van tennis.

Een screenshot van de werking van deze programma ziet u hiernaast. Een korte uitleg is al gegeven. Toch zal ik kort uitleggen waarom hij deze resultaat geeft.

Eerst zoekt het programma voor soortgelijke trainingsvoorbeelden. Hij doet dit in twee beurten:

- Eerst zoekt het algoritme voor voorbeelden waar de uitkomst **Yes** is;
- Dan zoekt het algoritme voor voorbeelden waar de uitkomst **No** is.

Days	Outlook	Temperature	Humidity	Wind	PlayTennis
Day 1	Sunny	Hot	High	Weak	No
Day 2	Sunny	Hot	High	Strong	No
Day 3	Overcast	Hot	High	Weak	Yes
Day 4	Rain	Mild	High	Weak	Yes
Day 5	Rain	Cool	Normal	Weak	Yes
Day 6	Rain	Cool	Normal	Strong	No
Day 7	Overcast	Cool	Normal	Strong	Yes
Day 8	Sunny	Mild	High	Weak	No
Day 9	Sunny	Cool	Normal	Weak	Yes
Day 10	Rain	Mild	Normal	Weak	Yes
Day 11	Sunny	Mild	Normal	Strong	Yes
Day 12	Overcast	Mild	High	Strong	Yes
Day 13	Overcast	Hot	Normal	Weak	Yes
Day 14	Rain	Mild	High	Strong	No

Computing results for:
Sunny, Cool, High, Strong

Compute for P(y):
 $P(y) = 9 / 14$
 $P(\text{Sunny}|y) = 2 / 9$
 $P(\text{Cool}|y) = 3 / 9$
 $P(\text{High}|y) = 3 / 9$
 $P(\text{Strong}|y) = 3 / 9$
 Result
 $= (9 / 14) * (2 / 9) * (3 / 9) * (3 / 9) * (3 / 9)$
 $= 0,0053$

Compute for P(n):
 $P(n) = 5 / 14$
 $P(\text{Sunny}|n) = 3 / 5$
 $P(\text{Cool}|n) = 1 / 5$
 $P(\text{High}|n) = 4 / 5$
 $P(\text{Strong}|n) = 3 / 5$
 Result
 $= (5 / 14) * (3 / 5) * (1 / 5) * (4 / 5) * (3 / 5)$
 $= 0,0206$

Winning result is:
PlayTennis = No

Per beurt kijkt de algoritme verder naar de features die overeen komen met de gegeven situatie. Voor opgave 1A geldt voor de uitkomst **Yes**:

- 9 van de 14 voorbeelden geven **Yes**;
- 2 van die 9 geven **Sunny**;
- 3 van diezelfde 9 geven **Cool**;
- 3 van diezelfde 9 geven **High**;
- 3 van diezelfde 9 geven **Strong**.

Met behulp van een eenvoudig formule krijgen we het resultaat voor P(y):

$$(9 / 14) * (2 / 9) * (3 / 9) * (3 / 9) * (3 / 9) = 0,0053$$

We herhalen dit voor P(n) om het volgende resultaat te krijgen:

$$(5 / 14) * (3 / 5) * (1 / 5) * (4 / 5) * (3 / 5) = 0,0206$$

Omdat het laatste groter is, kunnen we concluderen dat de situatie de uitkomst **No** geeft.

OPGAVE 1B

Precies hetzelfde wordt uitgevoerd met de tweede situatie, die u hiernaast ziet.

Het resultaat:

$$P(y) = (9 / 14) * (3 / 9) * (2 / 9) * (6 / 9) * (6 / 9) = 0,0212$$

$$P(n) = (5 / 14) * (2 / 5) * (2 / 5) * (1 / 5) * (2 / 5) = 0,0046$$

Voor deze situatie is de uitkomst dus **No**.

Days	Outlook	Temperature	Humidity	Wind	PlayTennis
Day 1	Sunny	Hot	High	Weak	No
Day 2	Sunny	Hot	High	Strong	No
Day 3	Overcast	Hot	High	Weak	Yes
Day 4	Rain	Mild	High	Weak	Yes
Day 5	Rain	Cool	Normal	Weak	Yes
Day 6	Rain	Cool	Normal	Strong	No
Day 7	Overcast	Cool	Normal	Strong	Yes
Day 8	Sunny	Mild	High	Weak	No
Day 9	Sunny	Cool	Normal	Weak	Yes
Day 10	Rain	Mild	Normal	Weak	Yes
Day 11	Sunny	Mild	Normal	Strong	Yes
Day 12	Overcast	Mild	High	Strong	Yes
Day 13	Overcast	Hot	Normal	Weak	Yes
Day 14	Rain	Mild	High	Strong	No

Computing results for:
Rain, Hot, Normal, Weak

Compute for P(y):
 $P(y) = 9 / 14$
 $P(\text{Rain}|y) = 3 / 9$
 $P(\text{Hot}|y) = 2 / 9$
 $P(\text{Normal}|y) = 6 / 9$
 $P(\text{Weak}|y) = 6 / 9$
 Result
 $= (9 / 14) * (3 / 9) * (2 / 9) * (6 / 9) * (6 / 9)$
 $= 0,0212$

Compute for P(n):
 $P(n) = 5 / 14$
 $P(\text{Rain}|n) = 2 / 5$
 $P(\text{Hot}|n) = 2 / 5$
 $P(\text{Normal}|n) = 1 / 5$
 $P(\text{Weak}|n) = 2 / 5$
 Result
 $= (5 / 14) * (2 / 5) * (2 / 5) * (1 / 5) * (2 / 5)$
 $= 0,0046$

Winning result is:
PlayTennis = Yes

OPGAVE 2

Ik gaan twee manieren behandelen om features na te gaan op hun relevantie:

- Leave-one-out-cross-validation (LOOCV)
- Forward Selection (FS)

De features worden op hun relevantie berekend door het verschil tussen het verwachte resultaat en het berekende resultaat te geven. Dit is de feature's afwijking. Bijvoorbeeld: de afwijking van *alle* features is gelijk aan het verschil tussen de som van de gegeven antwoorden uit de trainingsvoorbeelden (dit is 9,0) en de som van de berekende resultaat van elk trainingsvoorbeeld. Hiernaast ziet u het antwoord.

Days	Outlook	Temperature	Humidity	Wind	PlayTennis
Day 1	Sunny	Hot	High	Weak	No
Day 2	Sunny	Hot	High	Strong	No
Day 3	Overcast	Hot	High	Weak	Yes
Day 4	Rain	Mild	High	Weak	Yes
Day 5	Rain	Cool	Normal	Weak	Yes
Day 6	Rain	Cool	Normal	Strong	No
Day 7	Overcast	Cool	Normal	Strong	Yes
Day 8	Sunny	Mild	High	Weak	No
Day 9	Sunny	Cool	Normal	Weak	Yes
Day 10	Rain	Mild	Normal	Weak	Yes
Day 11	Sunny	Mild	Normal	Strong	Yes
Day 12	Overcast	Mild	High	Strong	Yes
Day 13	Overcast	Hot	Normal	Weak	Yes
Day 14	Rain	Mild	High	Strong	No

Beide methoden bepalen aan de hand van de afwijking de beste feature.

- LOOCV gooit de feature eruit dat voor de grootste afwijking zorgt;
- FS selecteert de feature die voor de kleinste afwijking zorgt, en gooit de rest eruit.

Met het programma is in te stellen op hoeveel features de functie toegepast moet worden, en dit geeft voor elk methode een ander effect:

- LOOCV gooit net zoveel features eruit als dat u aangegeven heeft;
- FS selecteert net zoveel features als dat u aangegeven heeft.

Om te controleren dat de methodes hetzelfde resultaat geven (dit zou immers wel moeten, als ze volgens hetzelfde principe werken), kunt u aangeven dat LOOCV drie features moet verwijderen, en FS één feature moet selecteren. Beide zullen ze als resultaat geven dat de feature **Wind** het relevantst is.

Zijn alle features nodig?

Met alle vier features is de afwijking 8,917. Dat ligt al aardig in de buurt van de verwachte 9,0.

Als we kijken naar de afwijkingen bij het weglaten van één van de vier features, neemt deze alleen maar toe. Dit ziet u in het plaatje hiernaast. Ook voor het weglaten van twee of drie features komt er niets zo dichtbij 9,0 als met alle vier features.

Alle vier features geven dus het beste classificatie in dit geval.

Days	Outlook	Temperature	Humidity	Wind	PlayTennis
Day 1	Sunny	Hot	High	Weak	No
Day 2	Sunny	Hot	High	Strong	No
Day 3	Overcast	Hot	High	Weak	Yes
Day 4	Rain	Mild	High	Weak	Yes
Day 5	Rain	Cool	Normal	Weak	Yes
Day 6	Rain	Cool	Normal	Strong	No
Day 7	Overcast	Cool	Normal	Strong	Yes
Day 8	Sunny	Mild	High	Weak	No
Day 9	Sunny	Cool	Normal	Weak	Yes
Day 10	Rain	Mild	Normal	Weak	Yes
Day 11	Sunny	Mild	Normal	Strong	Yes
Day 12	Overcast	Mild	High	Strong	Yes
Day 13	Overcast	Hot	Normal	Weak	Yes
Day 14	Rain	Mild	High	Strong	No

De situaties van opdracht 1 zijn eenvoudig opnieuw te berekenen voor elk combinatie van features, zoals de berekening bij opgave 1 gegeven is. Bijvoorbeeld: bij het weglaten van de feature **Humidity** en **Wind** bij opgave 1A krijgen we dit resultaat:

$$P(y) = (9 / 14) * (2 / 9) * (3 / 9) = 0,0476$$

$$P(n) = (5 / 14) * (3 / 5) * (1 / 5) = 0,0429$$

Hierbij past de uitkomst **No**.

OPGAVE 3

De classificatiefout is inderdaad verschillend bij de K-nearest-neighbor algoritme en de Naive Bayes classifier. Zelf heb ik meer vertrouwen in de manier waarop Bayes werkt.

Over de selectie van features kan ik weinig zeggen, want opgave 2 van taak 3 heb ik niet helemaal goed gedaan. Over een verschil in de afwijkingen is er wel sprake.

Wat dit zegt, is dat de manier waarop de algoritmen werken heel verschillend is. Ik denk dat je een algoritme moet kiezen afhankelijk van het doel dat je wil bereiken.