
Vogelgedrag aan de hand van GPS data

LEREN & BESLISSEN

Auteurs:

Areg SHAHBAZIAN

Marysia WINKELS

Jouke VAN DER MAAS

Koen KEUNE

Opdrachtgever:

Emiel VAN LOON

Begeleider:

Frank SMIT



UNIVERSITY OF AMSTERDAM

Inhoudsopgave

1	Introductie	2
2	Probleemspecificatie	2
2.1	Voorgaand werk	2
2.2	Probleem	3
3	Data	5
3.1	Beschikbare data	5
3.2	Analyse	6
4	Plan van aanpak	7
4.1	Software	7
4.2	Leren	7
5	Pre-processing van de data	9
5.1	Bruikbaarheid van de data	9
5.2	Pre-processing	9
6	Features	10
7	Resultaten	12
8	Conclusie	12

1 Introductie

Het analyseren van dierlijk gedrag, de tijdsbesteding en interactie met de leefomgeving, vormen de basis van de ecologie. De visuele observaties van dieren in het wild wordt beperkt door de toegankelijkheid van het leefgebied van de desbetreffende diersoort en uiteraard het tijdstip van de dag. De technologische ontwikkelingen die het mogelijk maken dieren van een niet hinderend zendertje te voorzien dat allerlei metingen kan verrichten zonder dat er een onderzoeker ter plaatse hoeft te zijn, biedt dus een enorme uitkomst voor ecologen. Op deze manier kan met een grotere regelmaat data verzameld worden met een grotere precisie. Echter, zonder de mogelijkheid deze gegevens juist te kunnen interpreteren is de ruwe data niet nuttig om als basis te dienen voor het verdere onderzoek van de ecoloog. Zo is het, bij wijze van voorbeeld, alleen maar mogelijk om de tijdsbesteding van een bepaalde diersoort voor en na een olieramp te vergelijken als de data en de gedragingen van de gezuiderde dieren juist worden geïnterpreteerd.

Machine learning algoritmes stellen ons in staat een model op te stellen dat op basis van de geleverde data dat, zonder verdere tussenkomst van mensen met uitzondering van een aantal aanpassingen op de data, het gedrag met een zekere mate van juistheid classificeert. Zo'n model bespaart onderzoekers een hoop veldwerk en zal dus zeker van nut kunnen zijn. In dit specifieke geval, ons gepresenteerd door opdrachtgever en ecoloog Emiel van Loon, gaat het classificeren van het gedrag van de scholekster.

2 Probleemspecificatie

2.1 Voorgaand werk

De scholeksters waarvan de data verzameld is leven op Schiermonikoog, een eiland in de Nederlandse Waddenzee (53.26° N, 06.10° W). Het waddengebied wordt de vogel gebruikt om te foerageren (het regelmatig terugkerend zoeken en vinden van voedsel door dieren) en hun nesten zijn gevestigd op de zoutmoerassen, het deel van het land dat regelmatig bloot komt te staan aan het zoute zeewater. In het onderzoek beschreven in *From Sensor Data to Animal Behaviour: an Oystercatcher Example*¹ door Shamoun-Baranes et al. (2012)² zijn drie van deze scholeksters van een accelerometer, een door zonne-energie aangedreven sensor in staat om posture, body movement en de

¹Ook naar gerefereerd als S2B

²<http://dx.doi.org/10.1371/journal.pone.0037997>

snelheid van de vogels te meten, en GPS tracking device die de geografische positie, tijd en huidige snelheid meet voorzien. Gebaseerd op de accelerometer gegevens en de observaties van het vogelgedrag door experts in het gebied zelf die aan de accelerometerdata gekoppeld kon worden, hebben zij drie verschillende modellen geprobeerd construeren, oplopend in ingewikkeldheid en detail.

Het eerste model was bedoeld om drie verschillende basis gedragingen van de scholekster te kunnen classificeren; *Fly*, *Terrestrial Locomotion* en *No Locomotion*, zoals te zien is in de eerste kolom van tabel 1. Een model gebaseerd op enkel de snelheid (S3) leek niet voldoende. Gebaseerd op de accelerometerdata en vijftien extra predictors leidde dit tot een model (SA3) met een 0.14 cross-validation error.

Een uitbreiding hierop was een poging om de acht belangrijkste gedragsklassen te classificeren. Onder *Fly* valt in dit model van acht gedragsklassen alleen *fly* zelf, onder *Terrestrial Locomotion* valt *forage* en onder *No Locomotion* valt *body care*, *stand* en *sit*. Ook *aggression*, *handle* en *walk* behoorde bij de acht belangrijkste gedragsklassen, maar deze werden door het model niet herkend. Alle instanties waarvoor *walk* was geobserveerd, viel bijvoorbeeld onder *forage* (of andersom). Dit is niet onverklaarbaar, want foerageren bevat ook het proces van het voedsel zoeken, wat de scholekster doet door het waddengebied te bewandelen. Zoals te zien is in tabel 2 is een overeenkomst tussen *aggression* en *handle* dat de verschillende sub-gedragingen (kolom 3) van het gedrag onder verschillende klassen in het 3-klassen model vallen. Dit kan eraan hebben bijgedragen dat het door het model niet goed werd herkend. Het model gevonden door Shamoun-Baranes et al. (2012) dat deze vijf belangrijkste gedragingen classificeerde (SA8) had een cross-validation fout van 0.28

Het derde model, dat alle zestien verschillende sub-behaviours had moeten kunnen modelleren, was niet haalbaar.

2.2 Probleem

Een nadeel aan de op zonne-energie aangedreven accelerometer sensor is dat de frequentie van de metingen van de gegevens afhangt van het al dan niet schijnen van de zon. De GPS zender wordt ook aangedreven door zonne-energie, maar het meten van de geografische positie, altitude boven het gemiddelde zee niveau, tijd en huidige snelheid dat de GPS zender doet kost aanzienlijk minder energie dan de metingen van de accelerometer kosten. Het

3-class model behaviour	behaviour	sub-behaviour
Fly	Fly	Normal flight
Terrestrial Locomotion	Forage	By sight By touch
No Locomotion	Body care Stand Sit	Preen Wash

Tabel 1: Tabel met classificeerbare gedragingen van de scholekster door de modellen van Shamoun-Barenes et al. (2012)

behaviour	3-class model behaviour	sub-behaviour
Agression	No Locomotion Terrestrial Locomotion	Bobbing Stand Solitary piping Chasing Piping ceremony Walk Solitary piping
Handle	Terrestrial Locomotion No Locomotion	Walking with prey Handling at surface Handling in situ
Walk		

Tabel 2: Gedragingen door de scholekster die niet door de modellen uit S2B herkent werden

zou om deze reden dus wenselijk zijn op basis van de GPS data alleen een model te kunnen maken dat het vogelgedrag kan classificeren.

Gezien het feit dat uit de GPS gegevens de huidige snelheid kan worden gehaald, ligt het voor de hand dat *Fly* gemakkelijk van andere klassen onderscheiden kan worden - de huidige snelheid van de scholekster zal bij vliegen aanzienlijk hoger liggen dan als er sprake is van *No Locomotion* of *Terrestrial Locomotion*. Ook *No Locomotion* zal hoogstwaarschijnlijk zonder al te veel moeite geclassificeerd kunnen worden — rekeninghoudend met de ruis in de GPS metingen zal alles wat onder een bepaalde grens van totale verplaatsing ligt als 'geen beweging' worden gezien. Dit zal uiteraard geverifieerd moeten worden.

Een interessanter probleem is het geval van de acht gedragsklassen. Is het mogelijk om aan de hand van de GPS-data het vogelgedrag in dezelfde acht klassen te kunnen onderverdelen als werd gedaan door Shamoun-Baranes et al (2012) aan de hand van de accelerometerdata?

3 Data

3.1 Beschikbare data

Voor dit vraagstuk hadden we verschillende datasets tot onze beschikking. Twee van deze datasets bevatten de ruwe data die is verzameld met de GPS-sensoren en de accelerometers. De **gps** dataset bestaat uit een GPS-positie in de vorm van een x- en een y-coördinaat, de snelheid op het moment van de meting en de tijd, in de vorm “*jjjj mm dd*”. Ook is aangegeven bij welke van de drie geobserveerde vogels een datapunt hoort.

De **accelerometer** dataset is een verzameling van versnellings-waardes in de x-, y- en z-richtingen. Na elke GPS observatie zijn deze waardes gedurende drie seconden gemeten en opgeslagen. Per GPS-observatiemoment komt dit neer op ongeveer 60 acceleratie-waardes. Om betekenis te geven aan deze ongeveer zestig waardes zijn features berekend die per observatie gelden. Deze zijn verzameld in de dataset **predictors**, opgesteld door Shamoun-Baranes et al (2012) om hun model te bouwen en bevat onder andere informatie over de gemiddelde hoek van het lichaam, de maximale dynamische acceleratie, de gemiddelde dynamische acceleratie en dergelijke.

De dataset **habitat** bevat informatie over de dagdeel in twee (*day*, *night*) of

drie (*day*, *twilight*, *night*) klassen en het geografische gebied waar een observatie is gedaan in vier klassen (roost, territory, mudflats, mudflats territory) per unieke gps fix.

In de **behaviour** dataset staat een deel van de GPS observatiepunten (ongeveer 4%) met de geobserveerde gedrag in de zestien verschillende subklassen, de bijbehorende gedragingen in acht klassen en de simpele drie klassen verdeling (zie tabel 1 en 2).

Ten slotte zijn we voorzien van een dataset **model_output** waarin de classificatie-resultaten staan van het classificatie-modellen uit *From Sensor Data to Animal Behaviour*, namelijk het SA3 en SA8 model (zie sectie 2.1), gekoppeld aan elke unieke identifier per gps-fix.

3.2 Analyse

Voor de GPS posities zijn er 16434 observaties gedaan bij drie verschillende vogels (Bird-ID 166, 167 en 169). Bij elke individuele scholekster horen respectievelijk 6187, 4786 en 5459 observaties.

De GPS metingen zijn verricht in een aantal periodes, allen zich bevindend tussen 30 – 06 – 2009 en 24 – 03 – 2010. In deze periodes zijn, niet onafgebroken, metingen verricht met de GPS-sensoren op de drie verschillende vogels. Dit is gedaan op tijdstippen verspreid over de gehele dag. De tijd tussen twee willekeurige metingen varieert van 15 seconden tot 165 dagen.

Het gemiddelde van de snelheden van alle drie de vogels is 0.327 m/s . De standaardafwijking hiervan is 1.48 m/s . De grote waarde voor de standaardafwijking komt door dat er een groot verschil is tussen loop- en vliegsnelheden van de vogels. Ook voeren de vogels geregeld acties uit, zoals *sit* of *body care*, waarbij ze voornamelijk stil staan en deze periodes van stilstand worden ook meegenomen in de berekening van de gemiddelde snelheid.

Doordat de GPS-sensoren last hebben van ruis, zijn de verschillende posities die samen een traject vormen niet altijd even precies. Zo is, wanneer er een lijn tussen opeenvolgende punten wordt getrokken, de vorm van het traject waarbij de door experts de observaties 'zit' is gemaakt, een kriskras patroon met een klein oppervlak. Ook een traject waarbij de vogel vliegt zal zelden enkel een rechte lijn vormen, zelfs als dit de vorm is van het originele door de vogel afgelegde traject, maar voornamelijk een zigzag patroon.

4 Plan van aanpak

4.1 Software

Weka

Weka³ wordt gebruikt om de leeralgoritmes uit te voeren. Weka is een collectie van machine learning algorithms voor data mining tasks, die direct uitgevoerd kunnen worden op de dataset. Ook zitten er in de Weka-toolkit zelf pre-processing tools, bijvoorbeeld de mogelijkheid om missende informatie te vervangen door het gemiddelde uit de kolom, die handig zouden kunnen zijn en worden de resultaten van het leeralgoritme op de dataset helder weergegeven.

Python

De high-level programmeertaal Python wordt gebruikt om Weka aan te kunnen sturen vanuit de terminal en het converteren van de `.csv` data naar `.arff` bestanden, zodat deze met Weka gebruikt kunnen worden. Python is ook nuttig geweest bij het analyseren van de resultaten van de machine learning algoritmes, voornamelijk om alle clusters en de correlaties hiertussen te vinden.

Matlab

Matlab, een technische softwareomgeving uiterst geschikt voor het uitvoeren van wiskundige toepassingen zoals het schrijven en implementeren van algoritmen, is in eerste instantie gebruikt om de data te prepareren, d.w.z. leesbaar te maken voor Weka, zoals te lezen valt in sectie 5. Ook maken door ons geschreven functies in Matlab het mogelijk om de resultaten van de clusteringsalgoritmen visueel zichtbaarder te maken, zoals omschreven in sectie 6, waardoor het onder andere mogelijk werd te zien welke *features* veel en weinig invloed hadden op het clusteren.

4.2 Leren

Zoals vermeld in sectie 3 is een klein deel van de dataset voorzien van een kolom dat het geobserveerde gedrag noteert. In total zijn op deze manier 702 GPS- en accelerometerdatapunten voorzien van een classificatie⁴. Dit maakt het mogelijk om een supervised learning algoritme toe te passen, aangezien supervised learning uit gaat van gelabelde data. Hoe meer gelabelde instanties er zijn voor classifier om het model op te baseren, hoe nauwkeuriger

³<http://www.cs.waikato.ac.nz/ml/weka/>

⁴Zie `behaviour.csv` file

het uiteindelijke model zal zijn. Dit is wat gedaan en omschreven is door Shamoun-Baranes et al. om het S3, SA3 en SA8 model te bouwen.

Echter, voor GPS posities zijn in totaal 16434 datapunten geregistreerd⁵, wat de dataset van 702 punten waarvoor ook werkelijk een gedrag genoteerd is relatief klein maakt, namelijk maar ongeveer 4% van alle GPS posities. De twee modellen, SA3 en SA8, uit S2B zijn gebruikt om aan de rest van de GPS datapunten ook van een label te voorzien - de model_output dataset. Het is een mogelijkheid om deze allemaal voor waar aan te nemen en supervised learning op de gehele dataset toe te passen, maar hier zullen dus ook foute classificaties tussen zitten. Een andere methode om de complete dataset - zonder klasse - te gebruiken is door unsupervised learning toe te passen, door bijvoorbeeld te clusteren. Een clusteralgoritme zoekt naar patronen in de data om de output te bepalen. De output in het geval van een clusteralgoritme zijn verschillende groepen van datapunten die relatief dicht bij elkaar liggen.

Vanwege het feit dat we beschikking hebben tot door mensen met kennis van vogelgedrag gelabelde data, leek het in eerste instantie verstandig om supervised leeralgoritmes op de data toe te passen. Dat is ook de keuze die de schrijvers van *From Sensor Data to Animal Behaviour: An Oystercatcher Example* hebben gemaakt. Na de teleurstellende resultaten hiervan is toch over gestapt op een vorm van unsupervised learning, zoals omschreven in sectie 7, waarbij geclusterd wordt gebaseerd op de gehele dataset en de correctheid van de clustering onder andere wordt onderzocht door te kijken of de gelabelde data allemaal in dezelfde cluster is belandt. Op basis hiervan kunnen de clusters gelabeld worden en het voordeel hiervan is ook dat er misschien patronen in de data worden ontdekt die gelinkt kunnen worden aan gedrag wat nog niet in de labels voorkomt — een vorm van semi-supervised learning.

Het clusteren is gedaan aan de hand van het EM-algoritme. Dit algoritme berekent iteratief de gemiddelde waarde van de clusters per feature en elke cluster wordt gevuld met de waardes die het meest waarschijnlijk bij die gemiddelde waardes horen. Het voordeel van dit algoritme is dat het snel kan clusteren en als er verschillende clusters in de data duidelijk aanwezig zijn daar succesvol in is. Er werd op voorhand aangegeven dat er naar acht clusters moest worden gezocht, dat is één per gedrag.

⁵Zie `gps.csv` file

5 Pre-processing van de data

5.1 Bruikbaarheid van de data

Gezien de hoge kosten die de acceleratie-metingen me zich meebrengen en het doel van de opdracht ligt het voor de hand dat alleen de GPS-data gebruikt wordt. Per individuele vogel blijven volgende data-attributen over:

Attribuut	Datum/tijd	GPS y	GPS x	Snelheid
Vorm	“jjjj-mm-dd uu-mm-ss”	y-coördinaat	x-coördinaat	in m/s

Om de data voor de **Weka**-toolkit bruikbaar te kunnen laden maken was het noodzakelijk dat de bestanden omgeschreven werden naar een **.arff**-formaat. Dit is deels handmatig en deels door middel van een **Python**-script gedaan. Ook hebben we om te testen **Matlab**-functies gebruikt om verschillende kolommen aan elkaar vast te plakken.

5.2 Pre-processing

Matlabfuncties zijn een belangrijk deel van het pre-processing van de data geweest, o.a. voor het inlezen van de **.csv** tabellen in Matlab om de data aan te kunnen passen, het verwijderen van de header (de namen van de kolommen), het omzetten van *string*-waardes van de classificaties naar discrete numerieke waardes, het splitsen van de data in trajecten en het vervangen van de *string* “NA”⁶ met de “?” (de gebruikelijke notatie van **Weka** om missende data aan te geven en een vereiste als de missende data mee genomen dient te worden in het leerproces).

Het splitsen van de data in trajecten vereisen twee stappen. In eerste instantie worden alle datapunten die onderling minder dan 60 seconden van elkaar af liggen samengenomen als traject. Vervolgens worden deze trajecten opgesplitst in kleinere trajecten van 60 seconden elk. Als dit opsplitsen niet precies uitkomt, worden de datapunten die over zijn verwaarloosd.

De eerste stap zorgt ervoor dat metingen die bij elkaar horen (een periode dat de GPS tracker in staat was verschillende metingen achter elkaar te verrichten) bij elkaar in een traject genomen worden, de tweede stap is noodzakelijk omdat de periode dat de GPS-tracker genoeg energie had om de metingen achtereenvolgens te verrichten extreem kan verschillen in tijd, waardoor de trajecten niet van gelijke lengte zouden zijn. Daarbij komt ook dat een vogel

⁶Not Available - missende data

in een traject, als het een langere periode van tijd in beslag neemt, verscheidenen gedragingen kan vertonen. Een tijdsperiode van zestig seconde is verkozen, omdat in deze periode genoeg metingen zitten om features eruit te halen en er genoeg trajecten over blijven om te clusteren. Een andere reden is dat de scholekster waarschijnlijk weinig van gedrag wisselt in deze periode.

Deze trajecten vervangen de individuele punten in de data, waarvan in totaal 113 gelabeld waren. **Python** werd vervolgens gebruikt om de classificatie algoritmes in **Weka** geautomatiseerd aan te roepen.

6 Features

Alleen de data gebruiken die is aangeleverd is niet genoeg om een goed model te maken om de gedragingen van de scholeksters te kunnen classificeren. Alhoewel het gedrag van de vogel afhankelijk kan zijn van waar hij zich precies bevindt, is de exacte positie in x- en y-coördinaten en huidige snelheid hoogstwaarschijnlijk een slechte indicatie van het exacte gedrag op dat moment. Om het leeralgoritme goed in staat te stellen om een passend model te vinden is het van belang om goede *features* toe te voegen. De features zijn allemaal uit de GPS-data gehaald en er is niet gekeken naar de snelheid van de vogel, omdat die voor kleine verschillen te inaccuraat is (Shamoun-Baranes et al. (2012)) en omdat het weinig tot geen invloed leek te hebben op het clusteringproces. De onderstaande features gaven waren wel van invloed op de cluster/klasse correlatie die een meting gaf van de effectiviteit van het clusteralgoritme (zie sectie 7).

Absolute Distance

Absolute Distance, ofwel absolute afstand, is het absolute verschil in GPS-positie tussen het eerste en het laatste punt in een traject. Dit zou een goede indicatie kunnen zijn voor het gedrag van de vogel, omdat het duidelijk maakt hoeveel de vogel verplaatst is. Zo is het bijvoorbeeld mogelijk dat het leeralgoritme het gedrag van de vogel wat bij dat traject hoort eerder als 'sit', 'stand' of 'body care' (allemaal vallend onder 'No Locomotion') zal classificeren als de absolute verschil in afstand klein is, en 'fly' of 'forage' als het absolute verschil tussen het begin- en eindpunt groot is.

Total Distance

De absolute afstand tussen het begin- en eindpunt kan klein zijn, terwijl in werkelijkheid de vogel veel kan hebben bewogen. Dit is bijvoorbeeld het geval als er tijdens een traject een richting op gelopen wordt en de vogel zich

vervolgens omkeert. Daarom wordt ook gekeken naar de totale afgelegde afstand, dus de som van de absolute verschillen tussen elk GPS-datapunt in een traject.

Tortuosity

Tortuosity is de eigenschap dat een lijn bochtig is, of veel bochten heeft. Het geeft aan dat er een verandering van richting heeft plaats gevonden tijdens het afleggen van de route en wordt berekend door de verhouding tussen Total Distance en Absolute Distance te nemen. Door de verhouding te nemen komt er een hoge tortuosity als de vogel veel heeft bewogen, maar over een traject gezien juist weinig. Dit geeft in feite aan in hoeverre de vogel in een traject doelloos rond heeft gelopen.

Area

Tevens is er gekeken naar de oppervlakte van het gebied dat de vogel heeft bewandeld. Deze oppervlakte wordt berekend door de buitenste punten in de GPS-data als hoeken voor een polynomial te gebruiken, zoals te zien is in figuur 1, en vervolgens de ingebouwde Matlab functie `polyarea` te gebruiken om de oppervlakte hiervan te berekenen.

Time

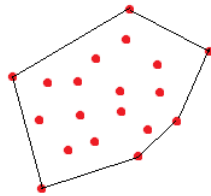
De **Weka**-tookit vertaalt de datum, in de vorm zoals hierboven gegeven, als het aantal verstreken milliseconden sinds 01-01-1970. Dit is naar onze mening niet de beste manier om de tijd als factor te gebruiken voor classificatie.

Bij de meeste diersoorten is er een sterk verband tussen de tijd van het jaar en tijdstip op de dag, en het gedragspatroon. Dit verband tussen tijd en gedrag is veel sterker dan het verband tussen de verstreken tijd in miliseconden en gedrag; dit laatste zou een goede indicatie van tijd zijn als er verschil is tussen het gedrag in verschillende levensperiodes van een enkele vogel, maar dit is niet zozeer het geval. Ook geldt dat een individuele vogel maar een beperkte tijd leeft in de absolute tijd en de grenzen die een dergelijke tijdwaarde voor een specifiek individu kan aannemen zijn per definitie gelegd op het begin en het einde van het leven van het individu. Omdat elk individu leeft in een unieke periode is de kans groot dat er uit de tijd-data zoals hierboven gegeven niet gegeneraliseerd kan worden. Dit is de reden dat we de datum/tijd kolom gesplitst hebben in twee kolommen die staan voor de x 'de dag in het jaar en de y 'de minuut in de dag.

Bekerkingen features

Er werd gezocht naar zoveel mogelijk discriminerende features, alleen het grootste deel van de trajecten vallen in een klein interval van waardes van deze features, zoals te zien is in figuur 2.

Een andere beperkende factor op de werking van de features is dat de features in zekere zin van elkaar afhankelijk zijn; alle features zijn immers afgeleid van de positie. Als het één het geval is, zal het andere in bepaalde mate ook het geval zijn. Een grote absolute afstand tussen begin- en eindpunt kan enkel het gevolg zijn van een grote totaal afgelegde afstand. En zo zal, in het geval dat er sprake is van een groot oppervlak van het traject, er ook sprake zijn van een grote afgelegde afstand (total distance). Dit probleem is minder groot als het aantal datapunten binnen een traject hoog ligt, want hoe meer datapunten er genomen worden hoe discriminerender de features zijn, maar dit resulteert in minder trajecten.



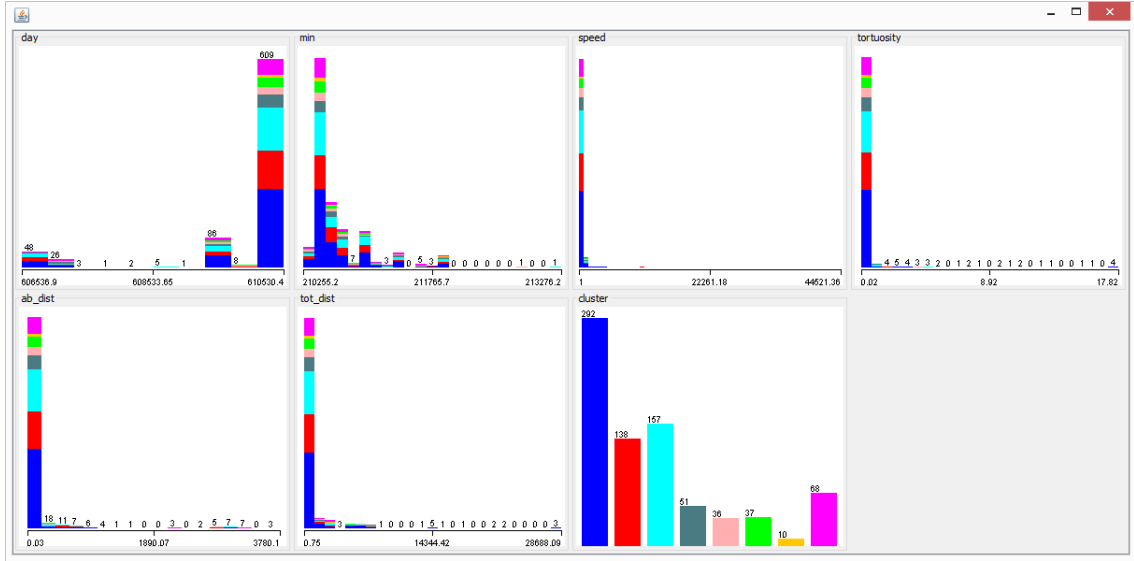
Figuur 1: Een polynomial om de buitenste punten

7 Resultaten

Om te testen hoe goed het door het clusteralgoritme (EM) gevonden model werkt is telkens dat een model werd gebouwd op de gehele dataset van trajecten geclusterd, op de gelabelde trajecten na. Vervolgens is van de gelabelde trajecten bepaald bij welk van de gevonden clusters deze horen en de verdeling van deze datapunten over de clusters is vergeleken met hun verdeling over de klassen. In tabel 3 is te zien hoe de klasse/cluster verdeling is met de gebruikte features (zie sectie 6) en trajecten.

8 Conclusie

Uit tabel 3 kan worden opgemaakt dat de gebruikte features en trajecten geen goede cluster/klasse-verdeling opleveren. In een ideaal geval zullen alle gelabelde data maar in één cluster voorkomen en een cluster enkel uit trajecten met dezelfde klasse bestaan. Dit is niet het geval. Veruit de meeste gelabelde data werd in cluster **a** ($\frac{33}{113} = 29.2\%$) of cluster **c** ($\frac{64}{113} = 56.64\%$) gestopt. Qua grootte komen deze clusters wel *in de buurt* van de grootte



Figuur 2: Per feature staat hoe vaak een traject voorkomt (y-as) en in welk interval het valt (x-as). De features zijn in volgorde: day, minute, speed, tortuosity, absolute distance, total distance en in het laatste vak zit de clusterverdeling. Op een paar uitschieters na is te zien dat over het algemeen het grootste deel van de trajecten in een klein interval zit.

van de klassen. Een groot gedeelte van de data heeft het label 'forage' ($\frac{51}{113}$) en een aanzienlijk deel 'sit' ($\frac{21}{113}$), maar het is niet zo dat alle instanties met label 'forage' exclusief bij of cluster **a** of **c** hoort — het is hierover redelijk gelijkverdeeld. Hetzelfde geldt voor de instanties met het label 'sit'. De cluster/klasse-correlatie is laag en het model werkt dus niet op de gelabelde data.

Het feit dat het model slecht werkt betekent dat het niet bruikbaar is. Op dit moment kan worden vastgesteld dat clusteren met het EM algoritme niet een goede methode is met de gegeven GPS-dataset, de trajecten gesplitst op de manier zoals beschreven in sectie 5.2 en de features uit sectie 6, om acht verschillende gedragssklassen van de scholekster te kunnen classificeren.

De voornaamste reden hiervoor is dat de features op deze trajecten erg weinig variatie hebben, wat het lastig maakt voor de clusteralgoritmes om clusters te vinden; het overgrote merendeel van de data ligt (te dicht) bij elkaar. Een ander clusteralgoritme gebruiken zou dus in dit geval dus geen oplossing bieden. Meer features, die gebruik maken van de ratio tussen twee al bekende features (zoals bijvoorbeeld area:afgelegde afstand), zal ook van weinig in-

Cluster	Aggression	Body Care	Fly	Forage	Handle	Sit	Stand	Walk
a	0	2	0	17	3	4	5	2
b	0	1	2	3	0	0	0	0
c	4	8	0	26	2	15	5	4
d	0	0	0	0	0	1	0	0
e	0	0	0	1	0	0	0	0
f	0	4	0	4	0	1	0	0
g	0	0	0	0	0	0	0	0
h	0	0	0	0	0	0	0	0

Tabel 3: De klasse-naar-clustercorrelatie van de trajecten.

vloed zijn.

Een andere reden is de manier waarop trajecten worden opgebouwd. Een traject dat in de eerste stap (zie sectie 5.2) een tijdsperiode van vijftien minuten overspande wordt in stap twee opgesplitst in vijftien trajecten van ieder één minuut. Dit zorgt vervolgens voor een scheve verdeling, want alhoewel zit gedrag en vlieg gedrag misschien even vaak voor kan komen, zal het vlieg gedrag een langere periode en dus meer mini-trajecten beslaan. Ook speelt het probleem dat in het artikel van Shamoun-Baranes (2012) al voorkomt, dat maar vijf van de acht gedragingen correct geclassificeerd konden worden omdat sommige gedragingen simpelweg te veel op elkaar leken, ook in onze clusterpogingen mee als hinderende factor. Om cluster-algoritmes bruikbare resultaten te laten geven, zullen deze punten verbeterd moeten worden.