

Lecture Notes voor Exploratieve en Descriptieve Data Analyse

B. Depaire

2019-06-01

Contents

Contents	1
Voorwoord	3
Hoe deze lecture notes te gebruiken	3
Over de auteur	3
1 Inleiding tot exploratieve data analyse in de bedrijfswereld	5
1.1 Netflix	5
1.2 Waar wordt data voor gebruikt in de bedrijfswereld?	6
1.3 Hoeveel data is er beschikbaar?	7
1.4 Waar komt data vandaan?	7
1.5 Waarover verzamelen bedrijven data	9
1.6 Van data tot ‘actionable insights’	9
1.7 Data Scientists	9
1.8 Verschillende soorten van data analyse	10
1.9 De kunst van data analyse	12
1.10 De kracht van descriptieve en exploratieve data analyse	12
1.11 Referenties	12
2 Datatypes en datavisualisatie	15
2.1 Data	15
2.2 Dataset	15
2.3 Klassieke datatypologie	16
2.4 De klassieke datatypologie is misleidend	17
2.5 Alternatieve datatypologie	17
2.6 Datavisualisatie	18
2.7 Datavisualisatie van 1 variabele (univariaat)	18
2.8 Datavisualisatie van 2 variabelen	23
2.9 Datavisualisatie met meer dan 2 variabelen	27
2.10 Referenties	28
3 Beschrijvende statistieken	29
3.1 Beschrijvende statistieken versus exploratieve plots	29
3.2 Notatie	29
3.3 Data	30
3.4 Univariate statistieken	30
3.5 Bivariate statistieken	33
Referenties	37
4 Exploratief Data Analyse Proces	39

5	Datavoorbereiding	41
5.1	Beginnen bij het begin	41
5.2	Data inlezen	42
5.3	Dataproblemen identificeren en corrigeren	50
5.4	Data opwaarderen	57
	Referenties	62
6	Exploratieve analyse van tijdgerelateerde data	63
6.1	Inleiding	63
6.2	Tijdstippen	63
6.3	Periode-data	67
6.4	Analyseren van tijdgerelateerde data	69
6.5	Referenties	73
7	Tidy Data	75
7.1	Inleiding	75
7.2	Case: NYC Vluchten 2013	75
7.3	Data in een lang formaat plaatsen (voor visuele analyses)	77
7.4	Data in een breed formaat plaatsen (voor overzichtelijke tabellen)	80
7.5	Referenties	81

Voorwoord

Dit boek bevat de lecture notes voor de cursus “Exploratieve en Descriptieve Data Analyse” (1ste Ba Handel ingenieur/Handel ingenieur in de Beleidsinformatica) aan de Universiteit Hasselt. Het idee van dit document is een begeleidende tekst aan te reiken ter ondersteuning van de slide-decks die gebruikt worden tijdens de hoorcolleges. Deze tekst is “bullet-point” gewijs opgebouwd en helpt het verhaal dat tijdens het hoorcollege wordt verteld terug op te roepen. Daarnaast zal er per hoofdstuk ook een referentielijst aangereikt worden met werken die de diverse topics in detail uitleggen.

Hoe deze lecture notes te gebruiken

- (optioneel) Neem een ‘geprinte’ versie van de slidedecks mee naar het hoorcollege. Gebruik dit om belangrijke aspecten tijdens het hoorcollege te markeren en korte nota’s toe te voegen. Ga zeker niet de volledige uitleg van het hoorcollege noteren. Dit is vaak niet mogelijk en indien je er toch in slaagt zal je tijdens het hoorcollege niet in staat zijn geweest om een eerste keer te reflecteren over de leerstof.
- Bestudeer na de les de lecture notes samen met de notities die je eventueel genomen hebt. Controleer of je alles begrijpt en waar nodig noteer je aanvullingen. Probeer een overzicht te verkrijgen van de diverse concepten die je tijdens het hoorcollege bestudeerd hebt en tracht na te gaan hoe je deze inzichten kunt gebruiken voor exploratieve en descriptieve data analyse.
- (optioneel) Lees de bronnen in de referentielijst. Indien er elementen niet duidelijk zijn in je eigen notities of de lecture notes, dan ga je best gericht op zoek naar de antwoorden op je vragen in de referentiewerken.

Over de auteur

Benoît Depaire is hoofddocent Beleidsinformatica aan de Universiteit Hasselt.

Chapter 1

Inleiding tot exploratieve data analyse in de bedrijfswereld

1.1 Netflix

- Netflix Prize (2006)
 - Wereldwijde open competitie voor de constructie van een nieuw algoritme dat moest voorspellen hoe goed een klant een film zou beoordelen op basis van zijn of haar filmvoorkeuren.
 - Winnaar was het team dat als eerste een verbetering van 10% kon realiseren ten opzichte van het algoritme van Netflix zelf.
 - Eerste prijs was 1 miljoen USD.
 - Hiervoor stelde Netflix een dataset ter beschikking met 100 miljoen filmbeoordelingen van 500 000 klanten met betrekking tot 18 000 films.
- Het kunnen voorspellen hoe hun klanten gaan reageren op specifieke films/series laat Netflix toe hun aanbod aan films en series te optimaliseren om het huidige klantenbestand te behouden en nieuwe klanten aan te trekken.
- De hoeveelheid data die door Netflix wordt verzameld is enorm.
 - In 2016 had Netflix 93.8 miljoen leden.
 - Netflix weet wanneer je pauzeert.
 - Netflix weet op welke dagen en welke uren je kijkt.
 - Netflix weet wat je kijkt.
 - Netflix weet van waar je kijkt.
 - Netflix weet op welk soort toestellen je kijkt.
 - Netflix weet wanneer je definitief stopt met het bekijken van een serie.
 - Netflix weet hoe snel je verschillende afleveringen van een serie achter elkaar kijkt.
 - Netflix weet welke titels je zoekt.
- Netflix komt op deze manier zeer veel te weten over het kijkgedrag van zijn klanten en kan op basis van deze inzichten betere beslissingen nemen. Bijvoorbeeld:
 - Netflix ontdekt uit haar data dat 40% van haar klanten een serie zijn begonnen te kijken die door het oorspronkelijke productiehuis is stopgezet.
 - Stel dat Netflix uit de data ook ontdekt dat 85% van deze klanten de serie volledig uitkijken zonder dat het tempo waartegen men afleveringen kijkt significant afneemt.
 - Op basis van deze inzichten kan Netflix eventueel beslissen om de rechten van de serie te kopen (die goedkoop zullen zijn aangezien de serie was stopgezet) en zelf een nieuw seizoen voor de serie te maken.
- House of Cards
 - Netflix deed het beste bod voor de serie House of Cards waardoor het won van kanalen zoals HBO.

- Ze kochten initieel 2 seizoenen van de serie waar een prijskaartje aan vast hing van meer dan 100 miljoen dollar.
- Deze beslissing was voor een groot stuk gebaseerd op data:
 - * Netflix leerde uit haar data dat haar klanten geïnteresseerd waren in producties van regisseur David Fincher.
 - * Netflix leerde uit haar data dat haar klanten geïnteresseerd waren in de oorspronkelijke Britse versie van House of Cards.
 - * Netflix leerde uit haar data dat haar klanten geïnteresseerd waren in producties met Kevin Spacey.
- Maar ook na de beslissing om deze serie te maken, bleef Netflix haar data gebruiken om slimme beslissingen te nemen.
 - * Er werden verschillende trailers gemaakt en afhankelijk van je voorkeuren kreeg je een trailer op maat te zien.
 - * Klanten die vooral graag Kevin Spacey zagen, kregen een trailer waar vooral Kevin Spacey in voorkwam.
 - * Klanten die vooral geïnteresseerd waren in films van David Fincher, kregen een trailer te zien die de typische “look&feel” had van David Fincher.
 - * Klanten die ook de Britse versie hadden gezien, kregen een trailer te zien die vooral op het verhaal focuste.

1.2 Waar wordt data voor gebruikt in de bedrijfswereld?

- Er zijn verschillende redenen waarom bedrijven data bijhouden. Deze kunnen we onderverdelen in volgende categorieën: Geschiedenis bijhouden, beslissingen nemen en voorspellingen maken.

Geschiedenis bijhouden

- Je registreert feiten zodat je achteraf met zekerheid kunt weten wat de realiteit in het verleden was.
- Dit is belangrijk als je wilt evalueren of een bedrijf goed beheerd wordt. Hiervoor heb je inzicht in het verleden nodig.
- De gegevens die worden bijgehouden in een boekhouding en jaarrekeningen zijn hier een typisch voorbeeld van.

Dagelijkse werking

- Omdat een bedrijf zijn dagelijkse werking kan uitvoeren, is het essentieel een up to date zicht te hebben van de werkelijkheid. Als een klant belt met een klacht over een levering, dan moet je als onderneming kunnen achterhalen wat de klant precies besteld heeft, of dit reeds geleverd is, of de klant al betaald heeft, enzovoort. Zonder deze informatie kan een onderneming haar dagelijkse werking niet garanderen.
- Om de dagelijkse werking te verzekeren, hebben bedrijven altijd al data bijgehouden. Denk maar aan informatie over aankoop- en verkooporders, de financiële gegevens in de boekhouding, de afschriften van een bank, de productieplanning, enzovoort.

Beslissingen nemen

- Een bedrijf neemt dagelijks talrijke beslissingen op verschillende niveaus
 - Operationeel.
 - * Vb: Moet ik een nieuwe bestelling plaatsen voor grondstof X of hebben we nog genoeg voorraad?
 - * Dit zijn typisch zeer frequente beslissingen die nodig zijn om de dagelijkse werking te garanderen.
 - * Deze beslissingen worden genomen door mensen op de werkvlloer of door het (lager) management.

- Tactisch/Management.
 - * Vb: Sluit ik best een exclusief contract af met 1 leverancier voor grondstof X voor een vaste periode en tegen een vaste verkoopsprijs of koop ik wanneer nodig tegen de marktprijs?
 - * Deze beslissingen worden minder frequent genomen dan operationele beslissingen en zijn typisch nodig om de werking van de onderneming op middellange termijn te optimaliseren.
 - * Deze beslissingen worden genomen door het management van een onderneming en hebben een aanzienlijke impact.
- Strategisch.
 - * Vb: Zullen we grondstof X aankopen op de markt of beslissen we deze grondstof zelf te produceren?
 - * Deze beslissingen hebben een zeer grote impact op de onderneming en worden niet frequent genomen. Ze vergen typisch ook lange voorbereidingstijd en bepalen de richting en toekomst van de onderneming op lange termijn.
 - * Deze beslissingen worden genomen door het topmanagement van een onderneming.
- Data kan bedrijven helpen bij het nemen van beslissingen.
 - Dit betekent echter niet dat beslissingen enkel en alleen op data gebaseerd zijn.
 - Vaak wordt data gecombineerd met ervaring en expertise om een beslissing te nemen.
- Bij het nemen van beslissingen op basis van data, kunnen we zowel patronen in historische data gebruiken alsook voorspellingen op basis van data.

1.3 Hoeveel data is er beschikbaar?

- De hoeveelheid data die de laatste decennia gegenereerd en opgeslagen wordt is enorm toegenomen.
- Deze groei is exponentieel (de groei gaat steeds sneller). Meer specifiek verdubbelt de hoeveelheid data in het digitaal universum iedere 2 jaar.
- Volgens een studie van IDC, bestond het digitaal universum in 2013 uit 4.4 Zetabytes data
 - 1 Zetabyte = 1024 Exabytes
 - 1 Exabyte = 1024 Petabytes
 - 1 Petabyte = 1024 Terabytes
 - 1 Terabyte = 1024 Gigabytes
- Volgens dezelfde studie zal het digitaal universum in 2020 uit 40 Zetabytes bestaan
- Echter, slechts 22% van deze data (in 2013) is geschikt voor analyse.
 - Er wordt geschat dat dit zal stijgen tot 35% in 2020.
- Slechts 5% van de geschikte data voor analyse wordt feitelijk geanalyseerd (2013).

1.4 Waar komt data vandaan?

- Data over een maatschappij
 - Het verzamelen van data is iets dat teruggaat tot in de oudheid.
 - Denk hierbij aan de volkstellingen die reeds plaatsvonden ten tijden van de Romeinen.
 - Een volkstelling gaat alle inwoners van een bevolking registreren, samen met diverse kenmerken zoals burgerlijke status, leeftijd, geslacht, enzovoort.
 - Volkstellingen waren en zijn nog steeds belangrijk voor een overheid om de impact van haar openbaar beleid te kunnen inschatten.
- Scientific Management
 - Frederick Taylor
 - Eind 19de eeuw
 - Benaderde het organiseren van werk op een wetenschappelijke manier.

- Ging data verzamelen om vervolgens te analyseren hoe men werk efficiënter kon organiseren.
- Een van de eerste vormen van dataverzameling en -analyse om bedrijfswaarde (productiviteit) te creëren.
- Beperkt in hoeveelheid data omdat registratie en analyse nog manueel gebeurde.

- Het ontstaan van het digitale tijdperk

- Met de uitvinding van de computer tijdens en na de tweede wereldoorlog, is de mensheid het digitale tijdperk ingegaan.
- De computer zorgt ervoor dat we data in een digitale vorm (als een reeks van één en nullen) opslaan. Dit biedt het voordeel dat exacte kopieën van de data gemaakt kunnen worden met één muisklik.

- Digitalisatie van de werkvloer

- Computers op de werkvloer dateert terug tot midden vorige eeuw, maar de grote doorbraak komt er met de opkomst van de personal computer
 - * 1977: Apple Home Computer II
 - * 1981: IBM Personal Computer
 - * Eind jaren 80, begin jaren 90 was de PC wijdverspreid op de werkvloer.
 - * Dit liet toe meer data te registreren, maar deze was nog moeilijk te delen met andere computers.
- Opkomst Internet/WWW in de bedrijfswereld
 - * 1990: De technologie voor WWW werd publiek gedeeld door Tim Berners-Lee.
 - * Dankzij WWW en internettechnologie werd het steeds eenvoudiger om digitaal werk te delen.
- Opkomst van e-commerce
 - * 1995: Begin van dot-com bubble/hype.
 - * Opkomst van digitale ondernemingen (vb. Amazon, Netflix, Google, ...).
 - * Digitale handel maakt het eenvoudiger om gegevens hierover te registreren.

- Digitalisatie van mensen

- Opkomst Web 2.0 (begin 2000)
 - * Inhoud van het web wordt nu gecreëerd door de bezoekers/gebruikers/klanten.
 - * Websites worden dynamisch (passen zich aan de context en bezoeker aan).
- Opkomst sociale media
 - * Gebruikers gaan spontaan hun leven digitaliseren.
 - * Hiervoor worden diverse media gebruikt (foto, video, tekst, ...).
 - * Facebook, Twitter, Instagram, Persoonlijke blogs,
 - * Nog nooit heeft zo'n groot deel van de wereldbevolking informatie gecreëerd en gedeeld met de rest van de wereld.

- Digitalisatie van dingen

- Opkomst goedkope sensoren
- Steeds meer “dingen” (machines, auto’s, huishoudtoestellen, huizen, steden, ...) worden ‘intelligent’
- Internet of Things (IoT): Al deze intelligente dingen worden via het Internet met elkaar verbonden.
- De hoeveelheid data die hiermee gegenereerd zal worden is ongezien.
- Volgens IDC studie waren in 2013 reeds 7% van de “verbindbare dingen” geconnecteerd aan het Internet of Things.
- In dezelfde studie voorspellen ze dat dit zal stijgen tot 15% in 2020.
- In 2013 werd 2% van alle data in het digitaal universum geproduceerd door het IoT.
- Verwacht wordt dat dit zal stijgen tot 10% in 2020.

1.5 Waarover verzamelen bedrijven data

- Het ultieme doel van een onderneming is gegevens te verzamelen die hen toelaten om het gedrag van hun omgeving beter te begrijpen, alsook de werking van hun eigen onderneming.
- Onder omgeving verstaan we:
 - Klanten
 - Concurrenten
 - Leveranciers
 - Alternatieve markten
 - Overheden
- Onder werking van eigen onderneming vertaan we o.a.:
 - Werknemers
 - Processen
 - Producten
 - Diensten

1.6 Van data tot ‘actionable insights’

- Management by data
 - Nieuwe discipline van management waarbij men inzichten uit data gebruikt om beslissingen te nemen.
 - Om beslissingen te kunnen nemen uit data, moet men deze eerst transformeren naar ‘actionable insights’
- Data
 - Data verwijst typisch naar de gegevens die geregistreerd en opgeslagen worden.
 - Data beschrijft een heel klein aspect van een realiteit (bijvoorbeeld op welk exact tijdstip ben ik aflevering 2 van “House of Cards” beginnen te kijken).
 - Data op zich heeft echter heel weinig waarde.
- Informatie
 - Als we echter data gaan analyseren, dan kunnen we dit transformeren tot informatie.
 - Informatie beschrijft een realiteit en gaat typisch op zoek naar patronen in de data en afwijkingen op deze patronen.
 - Bijvoorbeeld: Ik kijk typisch House of Cards gedurende de week om 20u00 ’s avonds, maar stop meestal met kijken om 20u30, waardoor ik in de week zelden een aflevering in 1 keer uitkijk.
 - Informatie is beschrijvend en zegt ons WAT de realiteit is.
- Actionable Insights
 - Actionable Insights is informatie die ons niet enkel zegt WAT de realiteit is, maar ons ook het inzicht verschafft HOE we moeten handelen.
 - Niet alle informatie is actionable.
 - Op basis van actionable insights en in combinatie met onze eigen ervaringen en kennis die we reeds bezitten, komen we soms tot inzichten die beschrijven HOE we moeten handelen.

1.7 Data Scientists

- Nieuwe jobomschrijving.
- Verantwoordelijk om data te transformeren naar ‘actionable insights’ en hier iets mee te doen om bedrijfswaarde te creëren.
- Omschreven als meest ‘sexy job’ van de 21ste eeuw door HBR

- Opvolgers van de Wall Street ‘Quants’ uit de jaren 80 en 90.
- Vaardigheden
 - Bedrijfskunde
 - * Productontwikkeling
 - * Management
 - Machine Learning / Big Data
 - * Ongestructureerde data
 - * Gestruktureerde data
 - * Machine Learning
 - * Big Data
 - Wiskunde en Operationeel Onderzoek
 - * Optimalisatie
 - * Wiskunde
 - * Simulatie
 - Programmeren
 - Statistiek
 - * Visualisatie
 - * Tijdreeksanalyse
 - * Wetenschappelijk onderzoek
 - * Data Manipulatie
- 4 profielen van data scientists
 - Data Businessperson
 - * Focust voornamelijk hoe data omzet kan genereren.
 - * Vaak in een leidinggevende rol.
 - * Werken zelf ook met data en beschikken over de nodige technische vaardigheden.
 - Data Creatives
 - * Zijn in staat een volledige data analyse zelfstandig uit te voeren.
 - * Hebben een hele brede bagage aan technische vaardigheden.
 - * Beschikken in zekere mate over bedrijfskundige vaardigheden.
 - * Gaan vaak innovatief om met data.
 - Data Developer
 - * Is voornamelijk gefocust op de technische uitdagingen met betrekking tot het beheer van data.
 - * Sterke programmeervaardigheden. Zijn in staat productie-code te schrijven.
 - * Zijn sterk in het gebruik van machine learning technieken.
 - Data Researcher
 - * Vaak mensen met een wetenschappelijke achtergrond (doctoraat).
 - * Sterk in statistische vaardigheden en wetenschappelijk onderzoek.

1.8 Verschillende soorten van data analyse

- Er zijn verschillende manieren om data analyse taken te classificeren.
- De classificatie die we hier hanteren is gebaseerd op het doel van de data analyse.

Descriptieve data analyse

- Deze analyse focust zich op het beschrijven van de data.
- Deze analyse gaat over het samenvatten van de grote hoeveelheid data in enkele statistische cijfers en grafieken.

- Deze analyse wordt gebruikt als je een grote hoeveelheid data krijgt en je snel inzicht wilt krijgen in de data.
- Voorbeelden:
 - Je hebt een dataset met alle studieresultaten van de studenten van 1ste bachelor HI/BI en je wilt weten wat de gemiddelde score is per vak.
 - Je hebt de verkoopscijfers van het afgelopen jaar en je wil weten welke drie producten het beste verkochten (zowel in aantal als in omzet).
- Descriptieve data analyse zegt alleen iets over de realiteit die door de data is beschreven. Je kan **geen** conclusies trekken die verder reiken dan de geobserveerde data.
- Je kan een descriptieve data analyse vergelijken met het werk van een detective die als taak heeft een beschrijving te maken van de misdaadscene.

Exploratieve data analyse

- Exploratieve analyse focust op het verkennen van de data en het zoeken naar interessante patronen en afwijkingen van deze patronen.
- Net als bij descriptieve data analyse zal exploratieve analyse de beschikbare data beschrijven en zeggen de resultaten **niets** over ongeobserveerde feiten.
- In tegenstelling tot bij descriptieve data analyse, gaat exploratieve data analyse verder dan het louter beschrijven van de data en tracht men interessante patronen te ontdekken in de data.
- Voorbeelden:
 - Zijn er specifieke kenmerken van studenten die sterk gerelateerd zijn aan hun studieresultaten.
 - Zijn er opmerkelijke verschillen tussen vakken wat betreft de punten die behaald worden. Zo ja, wat zijn dan deze verschillen.
 - Zijn er producten in ons gamma die gevoelig zijn voor seizoenseffecten?
- Je kan een exploratieve data analyse vergelijken met het werk van een detective die als taak heeft verbanden te ontdekken tussen verschillende bewijsstukken om zo inzicht te verschaffen wat er gebeurd is tijdens de misdaad.

Confirmatorische data analyse

- Confirmatorische analyse focust op het bevestigen of weerleggen van vermoedens die men heeft met behulp van de beschikbare data.
- In tegenstelling tot descriptieve en exploratieve data analyse zal men bij confirmatorische data analyse wel conclusies trekken die verder gaan dan de geobserveerde data.
- Omdat confirmatorische data analyses ook uitspraken doen over ongeobserveerde data, is er altijd een mate van onzekerheid over de correctheid van de resultaten.
- Voorbeelden:
 - Halen studenten met 8u Wiskunde achtergrond betere resultaten dan studenten met 6u Wiskunde achtergrond? In welke mate zijn we zeker dat dit voor alle studenten geldt en niet enkel voor de studenten waarover we data hebben?
 - Verkoopt product X beter bij mannen dan bij vrouwen? In welke mate zijn we zeker dat dit verschil niet een toevalligheid in de data is?
- Je kan een confirmatorische data analyse vergelijken met het werk van een rechter die op basis van het aangeboden bewijsmateriaal moet beslissen of er genoeg bewijs is om iemand te veroordelen van de misdaad.

Predictieve data analyse

- Het doel van predictieve analyse is om op basis van de beschikbare data voorspellingen te doen over de toekomst of over nieuwe/alternatieve situaties.

- Net als bij confirmatorische data analyse zal predictieve data analyse uitspraken doen die ook van toepassing zijn voor ongeobserveerde feiten/situaties.
- Bijgevolg is er net als bij confirmatorische data analyse dus een zekere onzekerheid over de conclusies die men trekt.
- Voorbeelden:
 - Zal een studente die met meer dan 80% haar diploma van het middelbaar onderwijs behaalt slagen in eerste zit voor het vak Exploratieve en Descriptieve Data Analyse?
 - Zullen de verkoopcijfers van product Y het komende jaar verder stijgen en met hoeveel procent?
- Je kan een predictieve data analyse vergelijken met het werk van een detective die op basis van het bewijsmateriaal op een misdaadscène moet voorspellen waar en wanneer de dader opnieuw zal toeslaan.

1.9 De kunst van data analyse

- Data analyse is een kunst. Net als bij iedere kunst, kunnen we hierbij drie componenten onderscheiden: kennis en vaardigheden, ervaring en creativiteit.
- Kennis en vaardigheden
 - Als data analist moet je de juiste hulpmiddelen kunnen identificeren voor het voorgelegde probleem.
 - Deze diverse hulpmiddelen moet je zo goed mogelijk beheersen.
 - Bij (exploratieve) data analyse gaat het hierbij zowel over analysetechnieken als over datavaardigheden.
 - Dit aspect kun je leren en laat je reeds toe om correcte analyses uit te voeren.
- Ervaring
 - Hoe meer data je analyseert, hoe beter je er in wordt.
 - Ook laat ervaring toe om sneller vaste patronen in je werk te herkennen en efficiënter te worden in wat je doet.
 - Ervaring is ook essentieel om complexere uitdagingen beheersbaar te maken.
 - Dit deel kunnen we je niet ‘leren’, maar heb je wel volledig in de hand.
- Creativiteit
 - Een kunstenaar die over kennis, vaardigheden en ervaring beschikt, maar creativiteit ontbreekt, kan perfecte replica's maken van een kustwerk, maar kan zelf geen nieuwe kunst creëren.
 - Creativiteit is in staat zijn op een nieuwe en onverwachte manier naar data te kijken en deze te visualiseren.
 - Het is niet zeker dat dit aspect aan te leren is. Maar dit hoeft niet te verhinderen dat je een goede data scientist wordt, zolang je maar voldoende aandacht besteedt aan de andere twee componenten.

1.10 De kracht van descriptieve en exploratieve data analyse

<https://www.youtube.com/watch?v=RUwS1uAdUcI>

1.11 Referenties

1. [Netflix Prize](#)
2. [How Netflix Uses Analytics](#)
3. [The Digital Universe of Opportunities - website](#)
4. [The Digital Universe of Opportunities - videoclip](#)
5. [How the Computer Changed the Office Forever](#)
6. [History of Computers in the Workplace](#)
7. [From Data to Understanding](#)

8. Data Scientist, the Sexiest Job of the 21st Centure
9. Analyzing the Analyzers

Chapter 2

Datatypes en datavisualisatie

2.1 Data

- Data is het resultaat van een meting van een attribuut van een specifiek object met een specifiek meetinstrument.
 - Het object verwijst naar wat je gaat meten.
 - * vb.: Student “Karel Jespers”.
 - Een object hoort meestal tot een verzameling van objecten. Deze verzameling wordt ook wel de populatie genoemd.
 - * vb.: Populatie “Studenten 1ste Ba HI/BI”.
 - Een specifiek object uit de populatie wordt ook wel element genoemd.
 - * vb.: “Karel Jespers” is een element uit de populatie “Student 1ste Ba HI/BI”.
 - Je meet altijd een specifiek aspect van het object. Omdat de meetwaarde van dit aspect kan variëren tussen verschillende objecten (elementen) in je verzameling (populatie), worden zulke aspecten ook variabelen genoemd.
 - * vb.: Lengte is een specifiek aspect (variabele) van de student “Karel Jespers” (element).
 - De meting gebeurt met behulp van een meetinstrument. Het is belangrijk te beseffen dat een meetinstrument altijd een zekere nauwkeurigheid heeft (tot hoeveel cijfers na de komma exact kan je meten?) en mogelijk ook onderhevig kan zijn aan willekeurige en/of systematische meetfouten.
 - * vb.: Student “Karel Jespers” wordt gemeten met een meetlat bevestigd tegen de muur. De meetlat heeft een nauwkeurigheid van 1cm, dus we kunnen zijn lengte niet uitdrukken in millimeters. Verder is de meetlat 2cm te laag opgehangen. Bijgevolg is er een systematische meetfout van 2cm. Tenslotte wordt de meting geregistreerd door een arts die vluchtig kijkt waar de student uitkomt op de meetlat. Het is dus niet onmogelijk dat de werkelijke lengte (willekeurig) afwijkt van de geregistreerde lengte.
 - * Tenzij anders vermeld wordt, gaan we in dit hoofdstuk uit van meetinstrumenten met oneindige nauwkeurigheid en zonder meetfouten.
 - De uitkomst van een meting voor een specifiek element wordt de waarde genoemd.
 - * vb.: 1m80 is de waarde van de variabele “lengte” voor element “student Karel Jespers”

2.2 Dataset

- Een dataset is een verzameling van data waarbij
 - Iedere rij één element uit de populatie voorstelt.
 - Iedere kolom een variabele is die gemeten wordt.
 - De verschillende rijen verschillende elementen uit dezelfde populatie voorstellen.
 - De waarde in een cel de meting is van de betreffende variabele voor het betreffend element.

Table 2.1: Uitgaande vluchten NYC 2013

luchthaven	maatschappij	datum	vertrek_vertraging	aankomst_vertraging	afstand	vliegtijd
EWR	United Air Lines Inc.	2013-01-01 05:15:00	2	11	1400	227
LGA	United Air Lines Inc.	2013-01-01 05:29:00	4	20	1416	227
JFK	American Airlines Inc.	2013-01-01 05:40:00	2	33	1089	160
LGA	Delta Air Lines Inc.	2013-01-01 06:00:00	-6	-25	762	116
EWR	United Air Lines Inc.	2013-01-01 05:58:00	-4	12	719	150
EWR	JetBlue Airways	2013-01-01 06:00:00	-5	19	1065	158
LGA	ExpressJet Airlines Inc.	2013-01-01 06:00:00	-3	-14	229	53
JFK	JetBlue Airways	2013-01-01 06:00:00	-3	-8	944	140
LGA	American Airlines Inc.	2013-01-01 06:00:00	-2	8	733	138
JFK	JetBlue Airways	2013-01-01 06:00:00	-2	-2	1028	149

2.3 Klassieke datatypologie

- Klassieke onderverdeling van data
 - Nominaal, Ordinaal, Interval en Ratio
 - Gebaseerd op de publicatie “On the Theory of Scales of Measurement” (1946)
 - * Beschrijft een hiërarchie van ‘datatypes’
 - Alles wat ordinaal is, is ook nominaal, maar niet omgekeerd.
 - Alles wat interval is, is ook ordinaal, maar niet omgekeerd.
 - Alles wat ratio is, is ook interval, maar niet omgekeerd.
 - * Identificeert geschikte statistische testen voor ieder type.
- Ieder datatype voldoet aan één of meerdere van de volgende eigenschappen:
 - Identiteit: Iedere waarde heeft een unieke betekenis.
 - Grootorde: Er is een natuurlijke volgorde tussen de waarden.
 - Gelijke intervals: Eenheidsverschillen zijn overal even groot. Dus het verschil tussen 1 en 2 is even groot als het verschil tussen 19 en 20.
 - Absoluut nulpunt: De waarde 0 betekent dat er ook feitelijk niets aanwezig is van de variabele en is niet een arbitrair gekozen nulpunt.

Nominaal

- Voorbeelden:
 - Geslacht: Man, Vrouw.
 - Ondernemingsvorm: vzw, bvba, nv.
- Voldoet enkel aan de eigenschap ‘identiteit’.
- Dit betekent dat we enkel concluderen of twee waardes gelijk zijn of niet. Er bestaat geen natuurlijke volgorde tussen de verschillende waardes.

Ordinaal

- Voorbeeld:
 - Opleidingsniveau: Lager onderwijs, Middelbaar onderwijs, Hoger onderwijs.
 - Klantentevredenheid: Ontevreden, Matig tevreden, Tevreden, Zeer tevreden.
- Voldoet aan de eigenschappen ‘identiteit’ en ‘grootorde’.
- Dit betekent dat we niet alleen kunnen concluderen of twee waardes gelijk zijn of niet. Het is ook mogelijk te bepalen welke waarde ‘groter’ is.
- We kunnen echter niet zeggen hoeveel groter één waarde is dan de andere.

Interval

- Voorbeeld:
 - Temperatuur (Celsius).
- Voldoet aan de eigenschappen ‘identiteit’, ‘grootorde’ en ‘gelijke intervals’.
- We kunnen nu twee waardes vergelijken, bepalen welke groter is alsook de verschillen tussen waardes met elkaar vergelijken.
 - We kunnen dus stellen dat het verschil tussen 8 en 9 graden Celsius daadwerkelijk minder groot is dan het verschil tussen 12 en 20 graden Celsius.

Ratio

- Voorbeeld:
 - Gewicht
- Voldoet aan alle 4 de eigenschappen.
- We kunnen verschillende gewichten met elkaar vergelijken, we kunnen bepalen wat zwaarder is en we kunnen gewichtsverschillen onderling vergelijken. Hierbij komt nu ook nog dat we kunnen zeggen hoeveel keer iets zwaarder is dan iets anders.
- Dit is een gevolg van het feit dat de waarde 0 nu feitelijk betekent dat iets geen gewicht heeft.

2.4 De klassieke datatypologie is misleidend

- Voorbeeld:
 - Op een feestje wordt bij het binnengaan oplopende nummers toegewezen aan iedere gast, beginnend bij 1.
 - Tijdens het feestje wordt er een tombola georganiseerd en wie nummer 126 heeft, heeft gewonnen.
 - 1 gast vergelijkt dit nummer met haar kaartje en ziet dat ze gewonnen heeft. Zij beschouwde de waarde op haar ticket dus als een nominale variabele want het enige wat ze vergelijkt is of de waarde op haar ticket verschillend is van de winnende waarde.
 - Een andere gast kijkt naar zijn kaartje en ziet dat hij nummer 56 heeft. Hij concludeert dat hij te vroeg is binnengekomen en beschouwt de waarde op zijn kaartje dus als ordinale.
 - Nog een andere gast heeft een kaartje met nummer 70 en beschikt over bijkomende data omtrent het ritme waarmee gasten zijn binnengekomen. Deze gast kan dus schatten hoeveel later hij had moeten binnenkomen om te winnen en interpreteert zijn nummer dus als een interval variabele.
- Dit voorbeeld illustreert dat het datatype niet een vaststaand kenmerk is van de data, maar afhankelijk is van de vraag die je tracht te beantwoorden en de extra informatie waarover je beschikt.

2.5 Alternatieve datatypologie

- Alternatieve taxonomie van data
 - Graden: vb. academische graad: “op voldoende wijze”, “onderscheiding”, “grote onderscheiding”, ... (geordende labels)
 - Rangordes: vb. plaats in voetbalklassemement: 1, 2, 3, ..., 16 (gehele getallen die beginnen bij 1)
 - Fracties: vb. percentage opgenomen verlof: van 0% tot 100% (ligt tussen 0 en 1, als percentage uit te drukken).
 - Aantallen: vb aantal kinderen: 0, 1, 2, ... (niet-negatieve gehele waarden).
 - Hoeveelheden: vb. inkomen (niet-negatieve reële waarden).
 - Saldo: vb. winst (negatieve en positieve reële waarden).
- Voor deze cursus volstaat het meestal een onderscheid te maken tussen categorische en continue variabelen.

- Categorisch: Nominaal + Ordinaal.
- Continu: Interval + Ratio.

2.6 Datavisualisatie

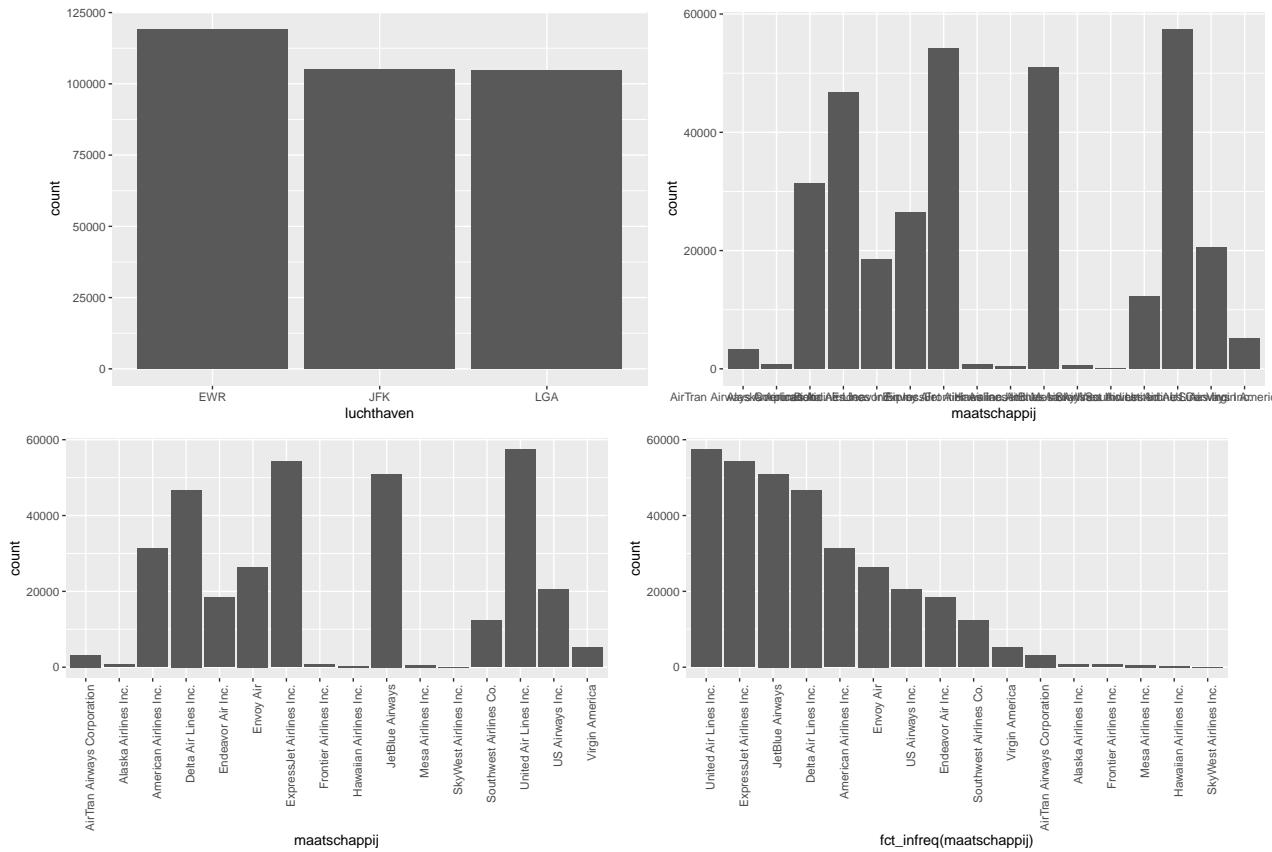
- Vaak de eerste stap om zicht te krijgen op de data.
- Relatief eenvoudig om patronen te zien, maar minder geschikt om exacte waarden te zien.
- We moeten hierbij onderscheid maken tussen exploratieve visualisaties en informatieve visualisaties om een boodschap over te brengen.
 - Exploratieve visualisaties dienen om snel inzicht te krijgen in patronen in de data. Men besteedt hierbij veel minder aandacht aan de opmaak van de visualisatie. Vaak is deze visualisatie tijdelijk en niet bedoeld voor communicatie naar derden.
 - Informatieve visualisaties dienen om een boodschap over te brengen aan derden. Hier dient men heel veel aandacht te besteden aan de opmaak zodat de boodschap duidelijk en helder gecommuniceerd wordt.
- We kunnen bij exploratieve visualisaties een onderscheid maken tussen univariate, bivariate en multivariate visualisaties.

2.7 Datavisualisatie van 1 variabele (univariaat)

- Als we slechts 1 variabele bestuderen, dan zijn we voornamelijk geïnteresseerd in de spreiding van de data. Dit wordt de verdeling van de data genoemd.
- Welke vragen kunnen we beantwoorden met dit soort visualisaties?
 - Wat is de meest voorkomende waarde van de data? Dit wordt ook de modus genoemd.
 - Bezit de data 1 modus, i.e. 1 waarde die duidelijk dominant is, of meerdere modi?
 - * Indien er slechts 1 afgetekende modus is, dan wordt de verdeling unimodaal genoemd.
 - * Indien er meerdere modi zijn (dominante waarden), dan wordt de verdeling multimodaal genoemd.
 - * Een multimodale verdeling kan er op wijzen dat de objecten in je data niet allemaal van hetzelfde type zijn en dat je in feiten twee populaties in je data aanwezig hebt.
 - Is de data geconcentreerd rond de modus of eerder breed verspreid. Met andere woorden, wat is de spreiding? Dit geeft inzicht in de variabiliteit van de data.
 - Is de data gelijkmataig verdeeld aan weerszijden van de modus of ziet we duidelijk meer data aan één zijde van de verdeling? Indien er meer data aan één zijde van de verdeling ligt (ten opzichte van de modus) dan zegt men dat de verdeling asymetrisch verdeeld is.
 - Zijn er waarden die opmerkelijk ver van de modus verwijderd zijn en geïsoleerd zijn van andere observaties? Dit worden extreme waarden of outliers genoemd. Deze verdienen meestal extra aandacht.

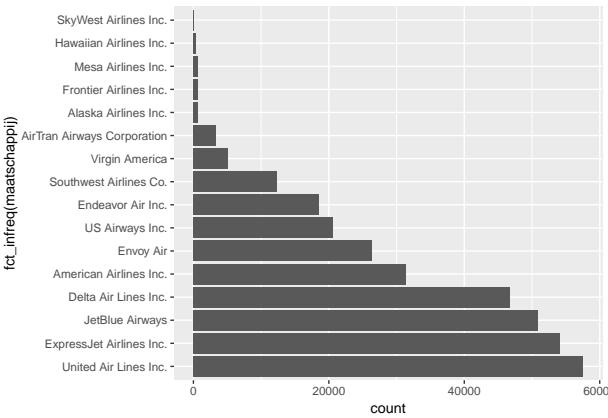
2.7.1 Categorische variabele

- Verticale barplot
 - Op de X-as staan de verschillende waarden van de categorische variabele.
 - Bij iedere waarde tekenen we een verticale balk die aangeeft hoe vaak die waarde in de dataset voorkomt.
 - Minder geschikt indien er veel waarden zijn. Dan wordt de X-as snel onleesbaar.
 - Je kan natuurlijk de labels roteren. Maar dit kan nog steeds onhandig zijn om te lezen.
 - In geval van een nominale variabele zijn er twee mogelijkheden om de waarden te rangschikken:
 - * Alfabetisch. Dit is handig om snel waarden terug te vinden.
 - * Volgens frequentie. Dit is handig om snel te zien welke waarden vaak/weinig voorkomen en geeft ook een beter beeld van de verdeling van de waarden.



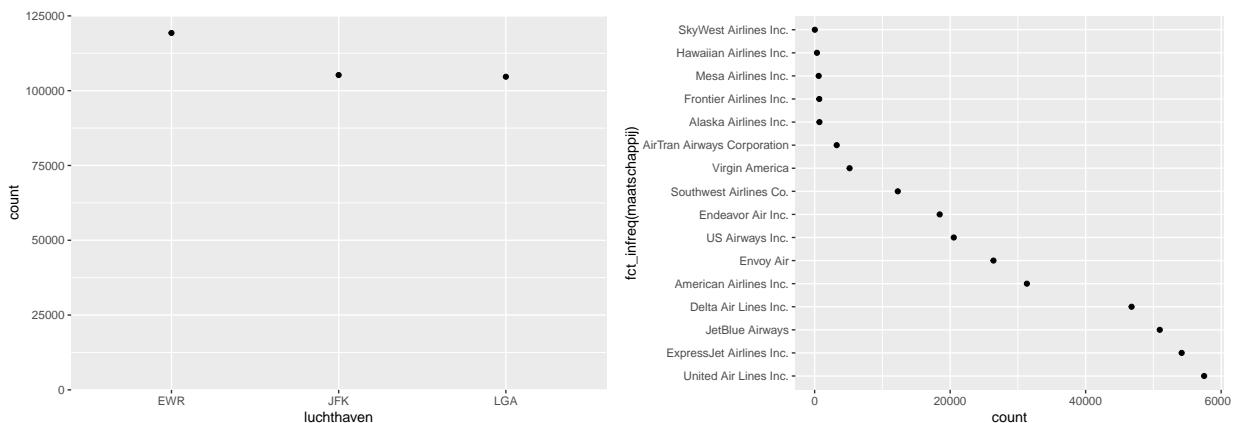
- Horizontale barplot

- Zelfde principe als verticale barplots, maar dan met horizontale balken.



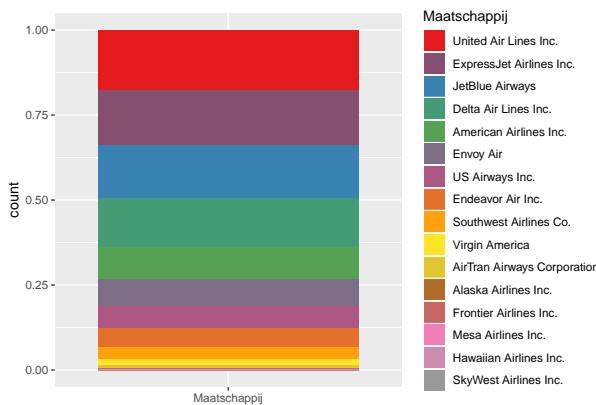
- Verticale/Horizontale dotplot

- In plaats van balken te gebruiken om de frequentie van een waarde aan te geven, kan je dit ook met punten doen.
- Een dotplot laat duidelijker zien waar de sprongen in de verdeling zit. Daarom is de dotplot vooral relevant als je de waarden ordent volgens frequentie.
- Net als de barplot kan je zowel een verticale als horizontale dotplot maken.



- Stacked barplot

- We maken nu slechts 1 kolom. Iedere waarde is een andere kleur en neemt een deel van de balk in beslag. De volledige balk stelt 100% van de data voor.
- Kan nuttig zijn om data cumulatief te bestuderen.
- Hiermee kunnen we vragen beantwoorden zoals: “Welke waarden moeten we nemen om met zo weinig mogelijk waarden x% van de objecten te hebben?”



- Waarom geen pie charts?

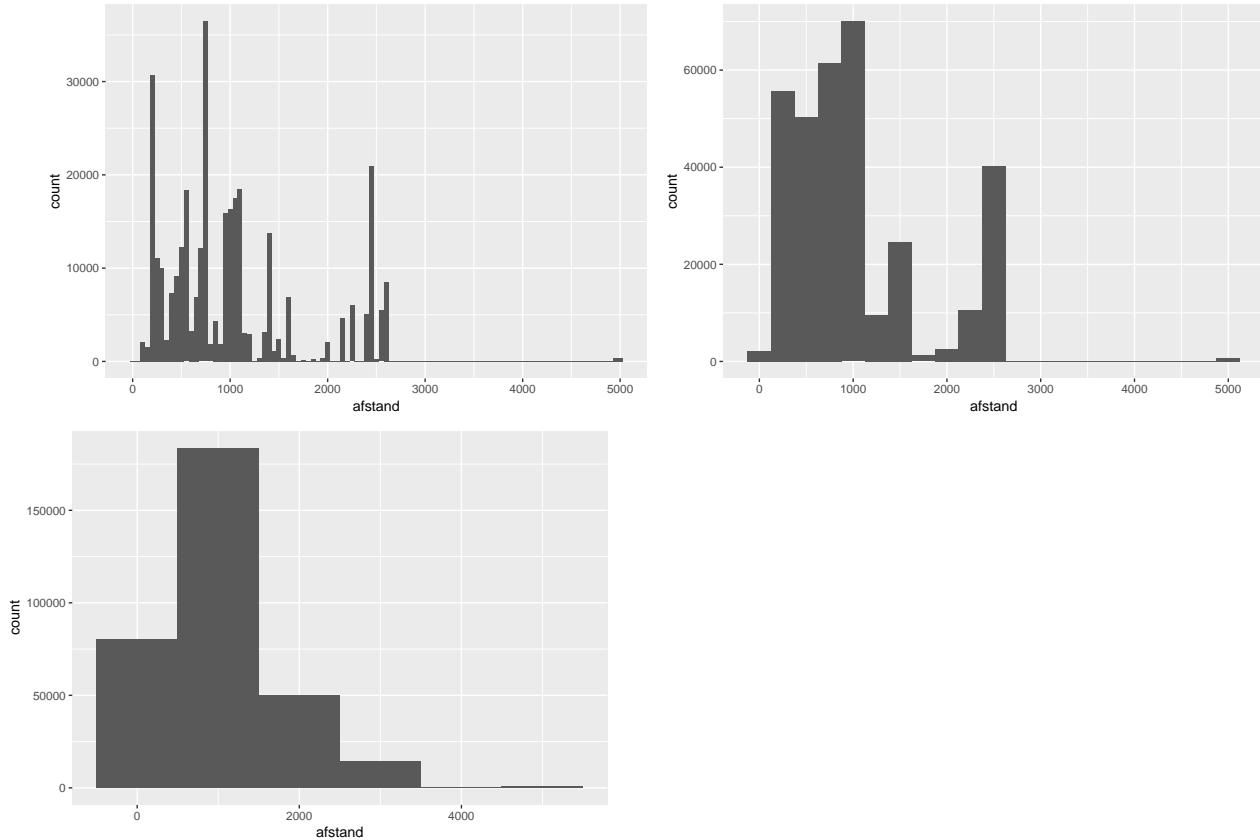
- Moeilijk te interpreteren.
- Verschillen tussen waarden zijn enkel duidelijk bij grote verschillen, terwijl barplots en dotplots deze ook bij kleine verschillen kunnen tonen.
- Voor cumulatieve analyses van de data zijn stacked barplots beter omdat het hier eenvoudiger is om af te leiden waar x% zicht bevindt.

2.7.2 Continue variabele

- Histogram

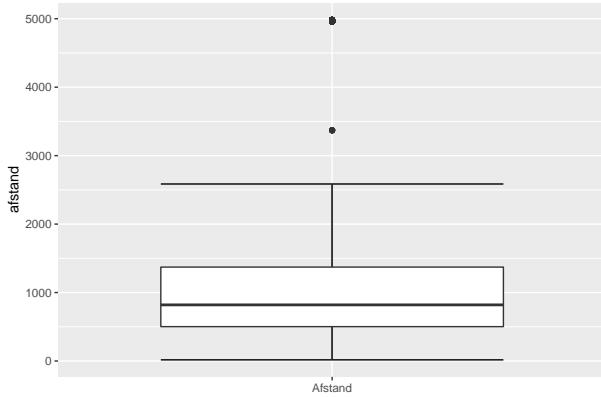
- Analoog met barplot, alleen gaan we hier eerst onze “categorieën” definiëren.
- Dit wordt ‘binning’ genoemd en wordt bepaald door een bin-breedte te kiezen.
 - * Je kan de binbreedte rechtstreeks kiezen of bepalen door vast te leggen hoeveel categorieën/bins je wenst.
- Voor de visualisatie, worden alle waarden gegroepeerd per ‘bin’.
- De binbreedte kan een enorme impact hebben op het uitzicht van de verdeling.
 - * Hoe breder de bins, hoe minder modi je kan detecteren.

- * Hoe smaller de bins, hoe meer modi je gaat zien, hoewel dit niet altijd even betekenisvol is.
- * Hoe smaller de bins, hoe minder data er in iedere bin gaan zitten en dan kunnen patronen wel in jouw dataset bestaan maar louter ten gevolge van toeval.

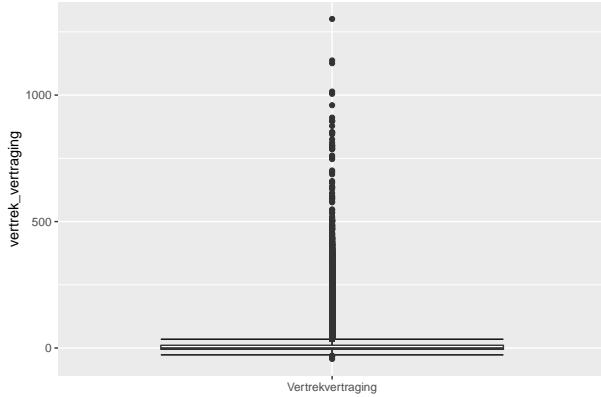


- Boxplot

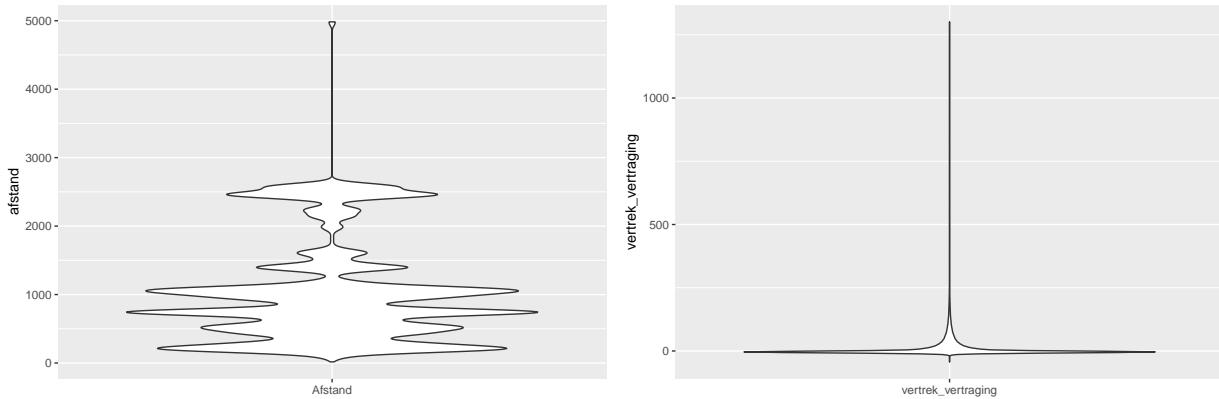
- De lijn in het midden duidt de mediaan aan. Dit betekent dat 50% van je data onder deze lijn ligt, terwijl 50% er boven ligt.
- De box in het midden duidt de middelste 50% van je data aan. Dit wordt ook de interkwartiel-box genoemd. Dit betekent dat 25% van je data onder deze box zit en nog eens 25% boven deze box ligt. Hoe groter de box, des te meer de data gespreid is.
- Indien de box aan één zijde van de mediaanlijn groter is dan aan de andere zijde, dan wijst dit er op dat de data meer gespreid is aan die kant.
- De “whiskers” geven de laatste datapunten aan die als “normaal” beschouwd worden. Datapunten buiten deze grenzen beschouwt een boxplot als outliers of extreme waarden.
 - * De grens waar data van normaal naar extreem overgaat wordt door de boxplot bepaald door anderhalf keer de grootte van de interkwartiel-box op te tellen (en af te trekken) van de bovenste (onderste) grens van de interkwartiel-box. Punten die hier buiten liggen zijn outliers en worden als aparte punten aangeduid. De whiskers zelf duiden de laatste datapunten aan binnen deze grenzen.



- Het is niet abnormaal dat er outliers in je data aanwezig zijn.
- Bij normaal verdeelde data zal je gemiddeld 7 outliers per 1000 datapunten mogen verwachten.
 - Een normale verdeling is een bepaalde manier waarop data waarden verdeeld kunnen zijn die in de realiteit vaak voorkomt.
- Indien je echter veel meer outliers ziet op je boxplot visualisatie, dan is de kans reëel dat er meer aan de hand is:
 - Er zijn bijvoorbeeld systematische meetfouten
 - De objecten in je data zijn in feite op bepaalde aspecten significant verschillend waardoor je ze apart zou moeten bestuderen.



- Violinplot
 - Een violinplot kan je beschouwen als een combinatie van een histogram en een boxplot.
 - Net als bij een boxplot wordt op verticale wijze getoond hoe de data verspreid is.
 - Net als bij een histogram kan je goed zien waar het volume (de massa) van de data zich bevindt.
 - Net als bij een histogram kan je detecteren hoeveel modi de data bezit.
 - In tegenstelling tot de boxplot, kan je bij een violinplot wel niet duidelijk zien waar bijvoorbeeld het 'midden' van je data is.



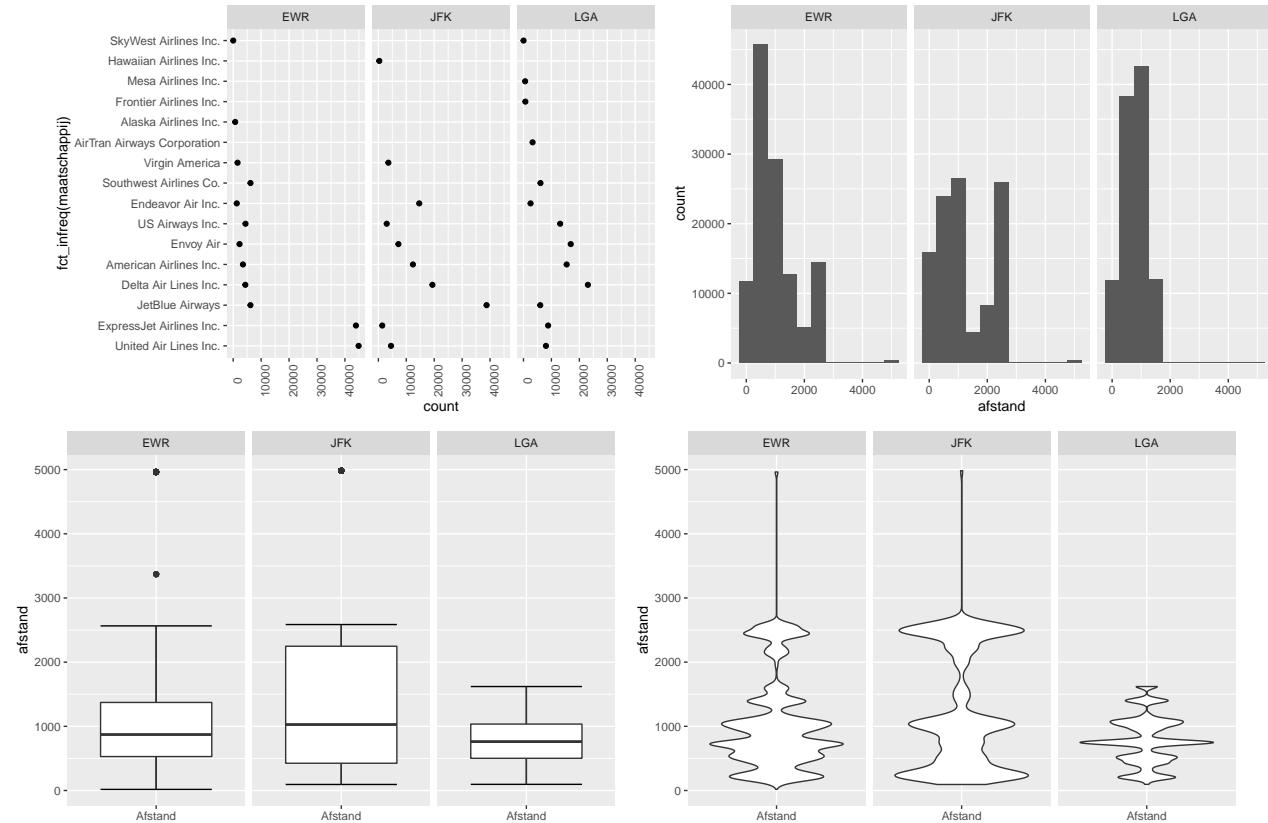
2.8 Datavisualisatie van 2 variabelen

- Van zodra er twee variabelen zijn, gaan we op zoek naar patronen in relaties tussen twee variabelen.
- Het is belangrijk en essentieel te beseffen dat mensen een automatische reflex hebben om te denken in termen van oorzaak-gevolg als we kijken naar relaties tussen twee variabelen.
 - Het is echter niet omdat er een duidelijke relatie bestaat tussen twee variabelen (correlatie), dat hier sprake is van een oorzaak-gevolg verband (causaliteit).
 - * Bijvoorbeeld: Indien in de zomer de verkoop van paraplu's sterk stijgt, dan zal de graanopbrengst in het najaar dalen. Dit betekent niet dat de verkoop van paraplu's een impact heeft op de graanopbrengst. Wat hier waarschijnlijk gebeurt, is dat door hevige regenval in de zomermaanden, de verkoop van paraplu's is toegenomen en de graanoogst tegenvalt.
 - * Soms is het intuïtief zeer onwaarschijnlijk dat de waargenomen correlatie causaliteit impliceert. Kijk hiervoor maar eens naar de voorbeelden op <http://www.tylervigen.com/spurious-correlations>
 - * Wanneer het echter plausibel is dat de waargenomen correlatie causaliteit voorstelt, is het belangrijk dat we tegen onze natuurlijke reflex in gaan en niet in termen van oorzaak-gevolg denken.
 - * Het aantonen van causaliteit is nooit mogelijk met descriptieve en exploratieve data analyse!
- Toch helpt het bij het maken van een datavisualisatie met 2 variabelen te denken in termen van oorzaak-gevolg, ook al weten we dat we dit nooit mogen concluderen!
 - De variabele die we het label “oorzaak” geven, zullen we voortaan “onafhankelijke variabele” noemen.
 - De variabele die we het label “gevolg” geven, zullen we voortaan “afhankelijke variabele” noemen.
- Waar we eigenlijk in geïnteresseerd zijn bij een visualisatie van 2 variabelen is de impact van de onafhankelijke variabele op de afhankelijke variabele weer te geven.
- Alle vragen die we kunnen stellen bij de visualisatie van 1 variabele, kunnen we nog steeds stellen, met telkens de bijkomende vraag of het waargenomen patroon verandert als de onafhankelijke variabele van waarde verandert.

2.8.1 Situatie 1: De onafhankelijke variabele is categorisch

- Oplossing 1: Meerdere univariate plots per waarde van de onafhankelijke variabele
 - Bij een categorische onafhankelijke variabele kan je altijd aparte univariante visualisaties maken van de afhankelijke variabele, per mogelijke waarde van de onafhankelijke variabele.
 - Om dit soort plots zo betekenisvol mogelijk te maken, moet je er voor zorgen dat het eenvoudig is patronen te vergelijken tussen verschillende waarden van de onafhankelijke variabele. Daarom respecteer je best volgende tips:

- * Plaats de verschillende plots in een intuïtief logische volgorde (natuurlijke volgorde bij ordinale onafhankelijke variabele, alfabetisch bij nominale variabele, ...)
- * Gebruik dezelfde schalen op de assen in iedere plot



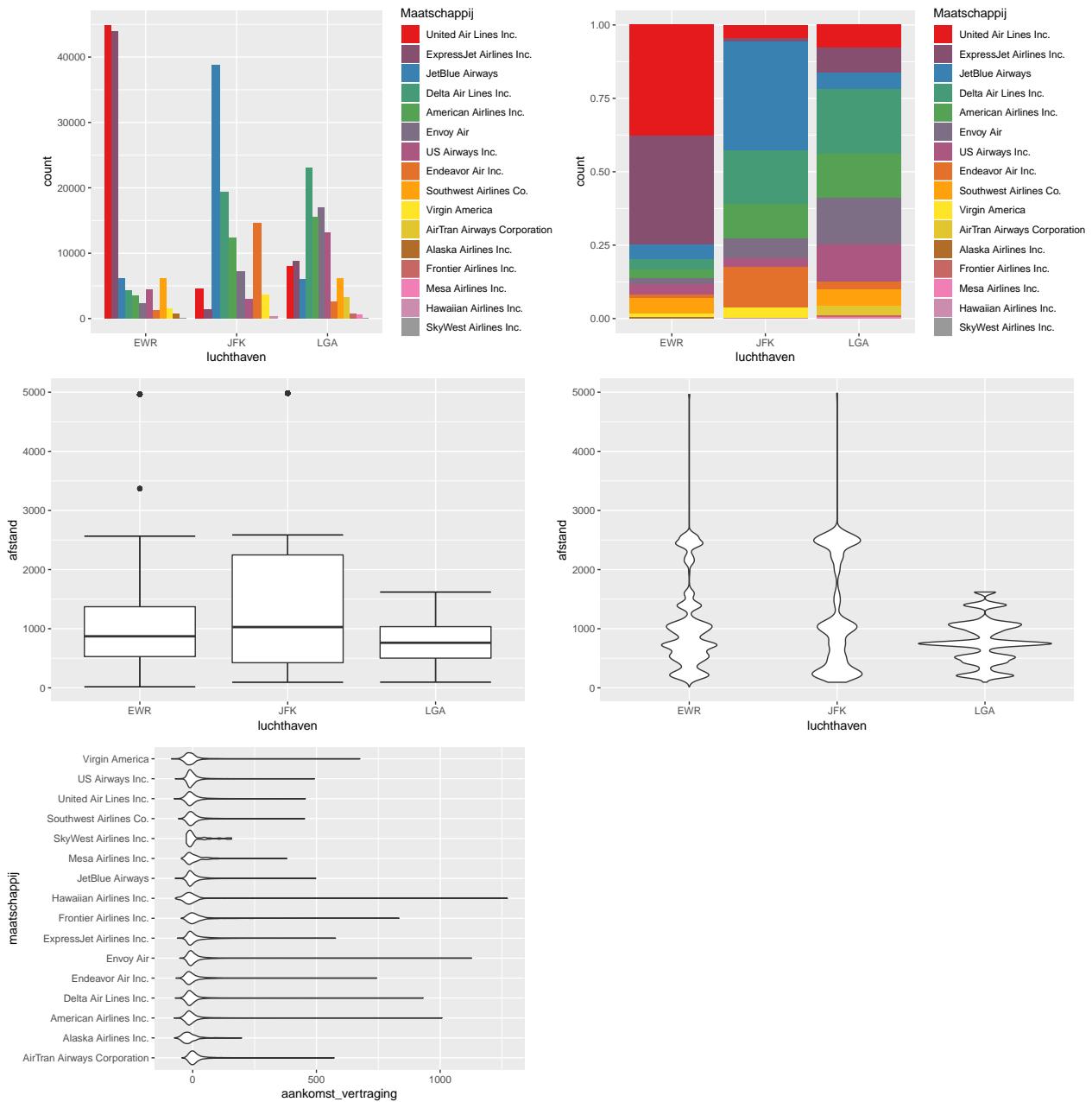
- Oplossing 2: 1 bivariate plot

- In dit geval plaats je de onafhankelijke variabele altijd op de X-as!
- Indien de afhankelijke variabele een categorische variabele is:

- * Kan je meerdere barplots op 1 grafiek visualiseren, met telkens de bars gegroepeerd per waarde van de onafhankelijke variabele.
- * Kan je meerdere stacked barplots op 1 grafiek plaatsen, met telkens een volledige stack per waarde van de onafhankelijke variabele.

- Indien de afhankelijke variabele een continue variabele is:

- * Kan je meerdere boxplots op 1 grafiek visualiseren, met telkens 1 boxplot per waarde van de onafhankelijke variabele.
- * Kan je meerdere violinplots op 1 grafiek tonen, met telkens 1 violinplot per waarde van de onafhankelijke variabele.

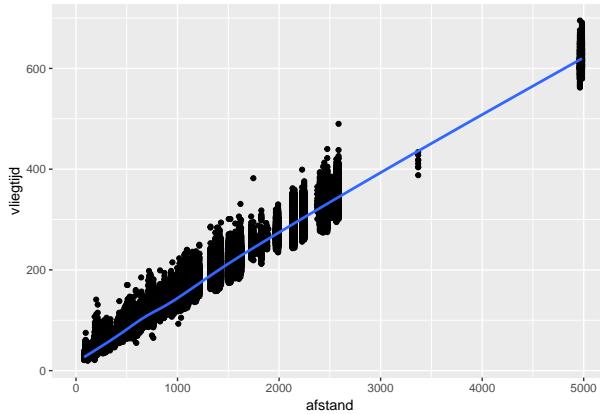


2.8.2 Situatie 2: De onafhankelijke variabele is continu

- In dit geval kan je geen aparte plot per mogelijke waarde van de onafhankelijke variabele maken omdat er mogelijk oneindig veel waarden zijn.
- Indien de afhankelijke variabele continu is, dan kan je een scatterplot maken.
 - Iedere observatie is een punt in je grafiek, waarbij de x-waarde op de grafiek overeenkomt met de waarde van de onafhankelijke variabele en de y-waarde op de grafiek overeenkomt met de waarde van de afhankelijke variabele.
 - * Om patronen beter te herkennen kan je een “trend-lijn” toevoegen.
- Indien de afhankelijke variabele categorisch is, dan kan je niet rechtstreeks een betekenisvolle plot maken omdat er waarschijnlijk te weinig datapunten zijn voor iedere mogelijke waarde van de onafhankelijke variabele.

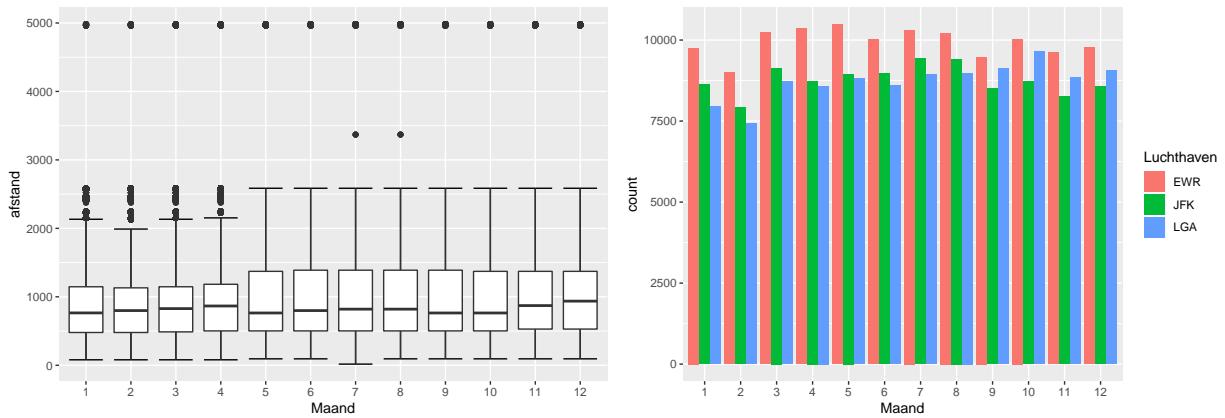
- Wat je dan best kan doen, is de onafhankelijke continue variabele categorisch maken door bins te definiëren. En dan ben je terug in de situatie waarbij de onafhankelijke variabele categorisch is.

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

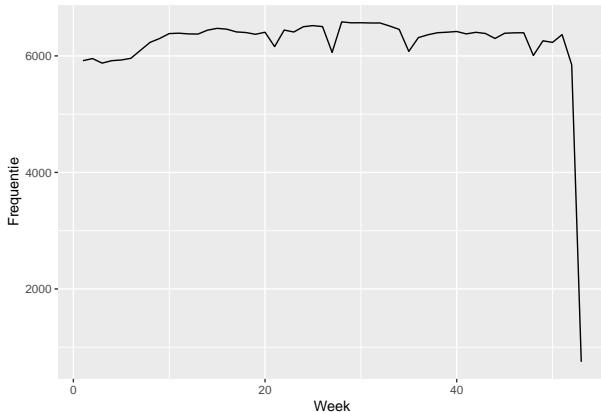


2.8.3 Situatie 3: De onafhankelijke variabele stelt de tijd voor

- Dit is een specifieke situatie waarbij je de onafhankelijke variabele zowel als een continue en als een categorische variabele kunt beschouwen.
- Hoe nauwkeuriger de tijdmeting, des te groter het continue karakter van de data.
- Op zich kan je bovenstaande visualisaties dus ook maken met een onafhankelijke variabele die de tijd voorstelt. Hierbij stelt de X-as nu de tijd voor.



- Er is echter een specifieke situatie waarbij een betere visualisatie mogelijk is, namelijk wanneer er op ieder mogelijk tijdstip slechts 1 observatie is.
 - Dit doet zich voor als men bijvoorbeeld op ieder uur van de dag de temperatuur opneemt in Ukkel.
 - In zulke gevallen kan men best een lijnplot gebruiken.

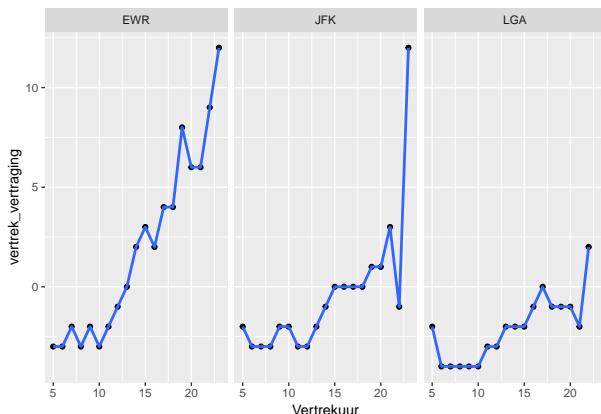


2.9 Datavisualisatie met meer dan 2 variabelen

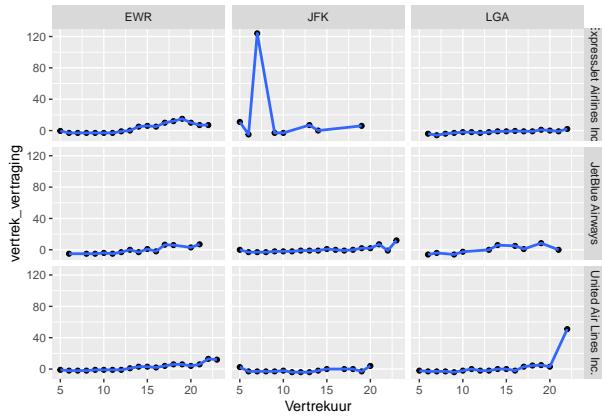
- Datavisualisatie van patronen tussen meer dan 2 variabelen worden snel te complex om te interpreteren.
- Het basisprincipe is wel eenvoudig.
 - Je hebt typisch 1 afhankelijke variabele (Y) en een aantal onafhankelijke variabelen (A, B, ...).
 - De bedoeling is het effect van de onafhankelijke variabelen op de afhankelijke variabele te visualiseren.
 - Hierbij definiëren we een orde tussen de onafhankelijke variabelen. We duiden de eerste-orde onafhankelijke variabele aan als A, de tweede-orde onafhankelijke variabele als B, enzovoort.
 - De bedoeling is om in eerste instantie het patroon weer te geven tussen A en Y en vervolgens de invloed van B op dit patroon.
 - Dit betekent dat je in eerste instantie een gewone bivariate plot tussen A en Y construeert en deze dan aanpast om de impact van B te visualiseren.

2.9.1 Onafhankelijke variabele B (2de orde) is categorisch.

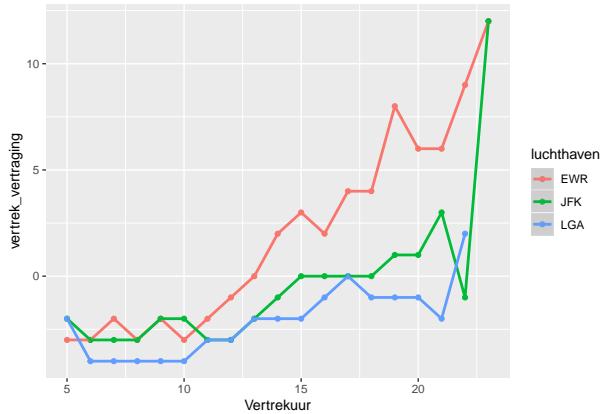
- Mogelijkheid 1 is per mogelijke waarde van variabele B een facet aan te maken.



- Je kan dit principe ook toepassen als je nog een derde-orde onafhankelijke categorische variabele hebt.



- Een tweede mogelijkheid is een ander aspect van de visualisatie aan de tweede onafhankelijke variabele B te koppelen. Bijvoorbeeld door een andere kleur te gebruiken voor verschillende waarden van variabele B.



2.10 Referenties

1. Scales of Measurement
2. Nominal, Ordinal, Interval, and Ratio Typologies are Misleading

Chapter 3

Beschrijvende statistieken

3.1 Beschrijvende statistieken versus exploratieve plots

- Plots zijn vooral sterk om patronen in de data te visualiseren.
 - Plots zijn minder geschikt om de ‘sterkte’ of ‘grootte’ van een patroon uit te drukken.
 - Beschrijvende statistieken laten dit wel toe aangezien aspecten van de patronen in een exploratieve plot in exacte getallen worden gegoten.
 - Er kunnen hoofdzakelijk 3 soorten beschrijvende statistieken worden onderscheiden:
 - Centrummaten
 - Spreidingsmaten
 - Associatiematen
-
- Centrummaten en spreidingsmaten zijn univariate statistieken en hebben als doel de verdeling van 1 variabele data samen te vatten in 2 cijfers.
 - Associatiematen zijn typisch bivariate statistieken en hebben als doel de samenhang tussen twee variabelen samen te vatten.

3.2 Notatie

- n : aantal observaties.
- X, Y : variabelen.
- x_i, y_i : de waarden voor variabelen X en Y voor observatie i .
- $x_{(i)}$: de i -de waarde voor X na rangschikking van klein naar groot.

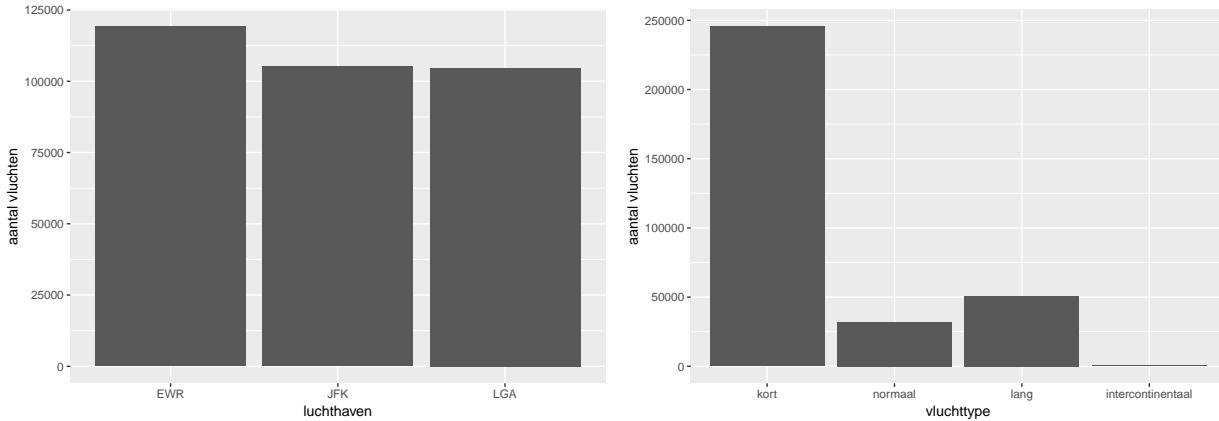
Table 3.1: Uitgaande vluchten NYC 2013

luchthaven	maatschappij	datum	vertrek_vertraging	aankomst_vertraging	afstand	vlieg
EWR	United Air Lines Inc.	2013-01-01 05:15:00	2	11	1400	
LGA	United Air Lines Inc.	2013-01-01 05:29:00	4	20	1416	
JFK	American Airlines Inc.	2013-01-01 05:40:00	2	33	1089	
LGA	Delta Air Lines Inc.	2013-01-01 06:00:00	-6	-25	762	
EWR	United Air Lines Inc.	2013-01-01 05:58:00	-4	12	719	
EWR	JetBlue Airways	2013-01-01 06:00:00	-5	19	1065	
LGA	ExpressJet Airlines Inc.	2013-01-01 06:00:00	-3	-14	229	
JFK	JetBlue Airways	2013-01-01 06:00:00	-3	-8	944	
LGA	American Airlines Inc.	2013-01-01 06:00:00	-2	8	733	
JFK	JetBlue Airways	2013-01-01 06:00:00	-2	-2	1028	

3.3 Data

3.4 Univariate statistieken

3.4.1 Categorische variabele



Frequentietabel

- De absolute frequentie f geeft aan hoe vaak een waarde voorkomt.
- De relatieve frequentie f/n geeft aan welk aandeel deze frequentie heeft in het totaal aantal elementen n .
- De cumulatieve frequentie $F_n(x)$ van een bepaalde waarde x geeft aan hoeveel observaties kleiner zijn dan of gelijk zijn aan x .
- De cumulatieve relatieve frequentie $F_n(x)/n$ van een bepaalde waarde x geeft aan hoeveel percent van de observaties kleiner zijn dan of gelijk zijn aan x .
- Een frequentietabel laat voor alle mogelijke waarden van een categorische variabele de absolute en relatieve frequentie zien (zowel normaal als cumulatief).
- Een frequentietabel laat zien waar een bepaalde waarde zich precies in de verdeling bevindt en hoe uitzonderlijk het is een specifieke waarde in de data te zien (of een waarde groter/kleiner dan).

Centrummaten

- Modus
 - Meest voorkomende waarde.

Table 3.2: Aantal vluchten per luchthaven

luchthaven	freq	rel_freq	cum_freq	cum_rel_freq
EWR	119282	0.36	119282	0.36
JFK	105230	0.32	224512	0.68
LGA	104662	0.32	329174	1.00

Table 3.3: Aantal vluchten per vluchtype

vluchtype	freq	rel_freq	cum_freq	cum_rel_freq
kort	245666	0.75	245666	0.75
normaal	31813	0.10	277479	0.85
lang	50980	0.15	328459	1.00
intercontinentaal	715	0.00	329174	1.00

Table 3.4: Centrummaten voor vluchtype

variabele	mediaan
vluchtype	kort

- Enige centrummaat voor nominale variabele.
- Ook bruikbaar voor ordinale variabele.
- Een variabele kan meerdere modi hebben.
- De modus is robuust tegen uitschieters.
- De modus kan je aflezen als de eerste rij in een frequentietabel als je deze ordent van de meest voorkomende tot de minst voorkomende waarde.

- Mediaan

- De middelste waarde na rangschikken van de gegevens.
- Voor ordinale variabelen definiëren we de mediaan aan de hand van de relatieve cumulatieve frequentie. De mediaan is de kleinste waarde waar 50% van de observaties kleiner dan of gelijk aan is.
- De mediaan is robuust tegen uitschieters.

Spreidingsmaten

- Kwantilen.

- Kwantilen (of percentielen) zijn gebaseerd op de cumulatieve relatieve frequentie.
- Het p% kwantiel is de kleinste waarde waar p% van de observaties kleiner dan of gelijk aan is.
- Het 50% kwantiel komt overeen met de mediaan.
- Veel voorkomende kwantilen om de spreiding van de data weer te geven zijn het 25% en 75% kwantiel.

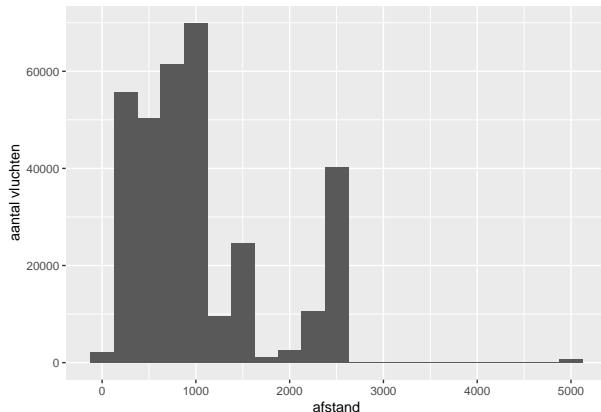
Table 3.5: Kwantielenvoor vluchtype

variabele	Q25	Q50	Q75
vluchtype	kort	kort	normaal

Table 3.6: Afstand (centrummaten)

variabele	gemiddelde	mediaan
afstand	1026.98	820

3.4.2 Continue variabele



Centrummaten

- Modus
 - Vaak minder bruikbaar bij een continue variabelen omdat iedere waarde zeer weinig voorkomt. Bijgevolg zijn er vaak zeer veel modi met telkens slechts enkele observaties.
- Mediaan
 - De middelste waarde na rangschikking van de gegevens.
 - In geval van een oneven aantal observaties, komt dit overeen met $x_{\frac{(n+1)}{2}}$.
 - In geval van een even aantal observaties zijn er twee ‘middelste’ observaties en is de mediaan gelijk aan $\frac{1}{2}(x_{\frac{n}{2}} + x_{\frac{n}{2}+1})$.
 - De mediaan is robuust tegen uitschieters.
- (Rekenkundig) Gemiddelde
 - $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$
 - Het gemiddelde is gevoelig voor uitschieters.
 - Dit is de centrummaat die mensen intuïtief selecteren indien mogelijk.

Spreidingsmaten

- Kwantielenvoor vluchtype
- Bereik
 - Dit is het verschil tussen de grootste en kleinste waarde.
 - Zeer gevoelig voor uitschieters.

Table 3.7: Afstand (spreidingsmaten)

variabele	minimum	Q25	Q50	Q75	maximum	bereik	IQR	var	sd
afstand	17	502	820	1372	4983	4966	870	542630.2	736.6344

- Is slechts gebaseerd op 2 observaties en bevat dus weinig informatie. Hiermee bedoelen we dat de spreiding van 2 variabelen sterk kan verschillen terwijl ze toch hetzelfde bereik hebben.
- Interkwartielafstand (IQR)
 - Dit is het verschil tussen Q75 en Q25.
 - Zelfde principe als het bereik, maar minder gevoelig voor uitschieters.
 - IQR is ook slechts gebaseerd op 2 observaties.
- Gemiddelde absolute afwijking (average absolute deviation)
 - Dit is de gemiddelde afwijking ten opzichte van het gemiddelde over alle observaties.
 - $\frac{1}{n} \sum_{i=1}^n |x_i - \bar{x}|$.
- Variantie
 - $s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$.
 - Vergelijkbaar met gemiddelde absolute afwijking, maar nu wordt het kwadraat gebruikt om te voorkomen dat de verschillen ten opzichte van het gemiddelde elkaar opheffen.
 - Vanuit analytisch standpunt is deze spreidingsmaat interessanter (geen absolute waardes, waardoor afgeleiden bijvoorbeeld eenvoudiger worden om te berekenen).
 - Wel gevoelig voor uitschieters en door het kwadraat wordt het effect van deze uitschieters ook nog eens vergroot.
 - De wortel van de variantie wordt de standaardafwijking genoemd. De standaardafwijking heeft het voordeel dat het indezelfde eenheid uitgedrukt wordt als de oorspronkelijke data.
- Median Absolute Deviation (MAD)
 - Dit is de middelste afwijking ten opzichte van de mediaan over alle observaties.
 - $MAD = \text{median}(|X_i - \text{median}(X)|)$.
 - Deze maatstaf is robuster tegen outliers.

3.5 Bivariate statistieken

3.5.1 Continu versus Continu

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

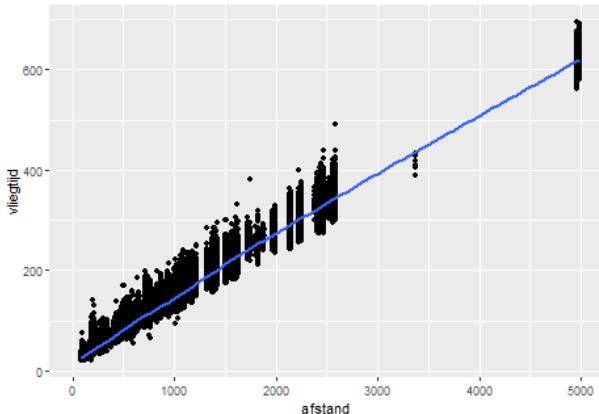


Table 3.8: Correlatie tussen afstand en vliegtijd

variabelenpaar	pearson	spearman
afstand-vliegtijd	0.99	0.98

Correlatie

- Covariantie
 - $cov(x, y) = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})$.
 - Bij een positieve associatie tussen twee variabelen zal de covariantie positief zijn.
 - Bij een negatieve associatie tussen twee variabelen zal de covariantie negatief zijn.
 - De covariantie is echter afhankelijk van de maateenheid van de variabelen, waardoor ze weinig bruikbaar is om de sterke van de associatie weer te geven.
- Pearson correlatiecoëfficiënt
 - Herschaalt de covariantie naar de schaal $[-1, 1]$
 - Laat toe om de sterke van een associatie te evalueren.
 - $r(x, y) = \frac{cov(x, y)}{s_x s_y}$
 - $r(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}$
 - Meet **lineaire** associatie tussen 2 variabelen.
 - Twee variabelen kunnen positief geassocieerd zijn, maar in een niet-lineaire wijze, waardoor de correlatiecoëfficiënt naar nul gaat.
 - Meest gebruikelijke correlatiecoëfficiënt voor continue variabelen.
 - Daarom best altijd samen met een puntenwolk bekijken.
- Spearman's rangcorrelatiecoëfficiënt.
 - Zelfde principe als Pearson's, maar dan gebaseerd op de rangorde van de waarden in plaats van de waarden zelf.
 - r_i : rangorde van waarde x_i . Bijvoorbeeld $r_i = 4$ betekent dat de waarde x_i de vierde kleinste waarde is.
 - s_i : rangorde van waarde y_i .
 - $\rho(x, y) = \frac{\sum_{i=1}^n (r_i - \bar{r})(s_i - \bar{s})}{\sqrt{\sum_{i=1}^n (r_i - \bar{r})^2} \sqrt{\sum_{i=1}^n (s_i - \bar{s})^2}}$
 - Meet associatie tussen 2 variabelen, dus niet specifiek lineaire associatie.
- Kendall's correlatiecoëfficiënt
 - Ook wel Kendall's tau genoemd.
 - De methode is gebaseerd door alle mogelijke observatieparen (x_i, y_i) en (x_j, y_j) te bestuderen.
 - Net als Spearman's aanpak gebaseerd op rangorde (r_i, s_i) en niet de feitelijke waarden.
 - Indien $r_i > r_j$ en $s_i > s_j$ (of $r_i < r_j$ en $s_i < s_j$) dan zijn observaties i en j concordant.
 - Indien $r_i > r_j$ en $s_i < s_j$ (of $r_i < r_j$ en $s_i > s_j$) dan zijn observaties i en j discordant.
 - Notatie: C en D zijn respectievelijk het aantal concordante en discordante paren.
 - $\tau = \frac{C-D}{\frac{1}{2}n(n-1)}$
 - Net als Spearman's correlatiecoëfficiënt, focust Kendall's tau op de associatie (positief of negatief) en niet specifiek op lineaire associatie.
 - Het nadeel van Kendall's tau is dat je alle observatieparen moet bestuderen en het aantal kan snel exploderen bij veel observaties. Immers het aantal paren is $\frac{n!}{2!(n-2)!}$. Hierdoor kan je Kendall in de praktijk niet gebruiken als je veel observaties hebt.

Table 3.9: Fictieve dataset

x	y
1	0.0
2	4.0
3	5.0
4	5.5
5	7.0
6	15.0
7	15.6
8	16.0
9	50.0
10	1000.0

Table 3.10: Correlatiecoëfficiënten fictieve dataset

variabelenpaar	pearson	spearman	kendall
x-y	0.55	1	1

Vergelijking correlatiecoëfficiënten

- Rangcorrelatiecoëfficiënten meten associatie, terwijl Pearson correlatiecoëfficiënt **lineaire** associatie meet!

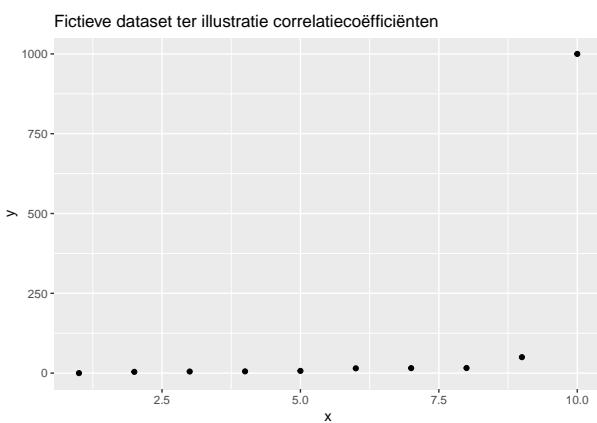


Table 3.11: Afstand-Luchthaven (centrummaten)

	luchthaven	gemiddelde	mediaan
EWR		1049.58	872
JFK		1247.16	1028
LGA		779.84	762

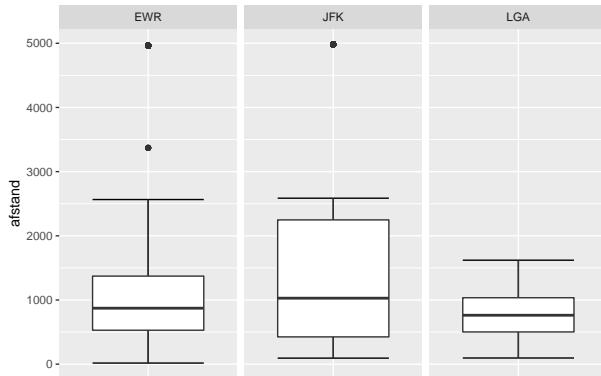
Table 3.12: Afstand-Luchthaven (spreidingsmaten)

luchthaven	var	min	Q25	Q50	Q75	max	bereik	IQR	sd
EWR	536177.0	17	529	872	1372	4963	4946	843	732.2411
JFK	842460.4	94	425	1028	2248	4983	4889	1823	917.8564
LGA	138132.3	96	502	762	1035	1620	1524	533	371.6615

Table 3.13: Correlatie tussen vluchtype en vliegtijd

variabelenpaar	spearman
vluchtype-vliegtijd	0.76

3.5.2 Categorisch versus Continu



Univariate statistieken per categoriewaarde

- Je toont de relevante centrum- en spreidingsmaten voor de afhankelijke categorische variabele per waarde van de onafhankelijke categorische variabele.

Correlatie

- Enkel toepasbaar als de categorische variabele ordinaal is.
- Pearson's correlatiecoëfficiënt kan je NIET toepassen.
- Spearman rangcorrelatiecoëfficiënt (ρ).
- Kendall's rangcorrelatiecoëfficiënt (τ) kan theoretisch wel toegepast worden, maar is in de praktijk vaak niet haalbaar.

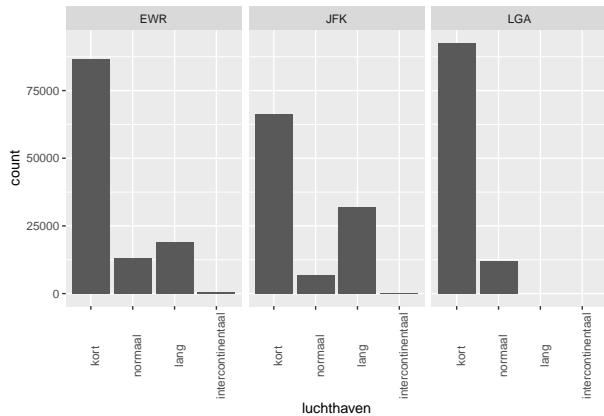
Table 3.14: Centrummaten voor vluchtype-luchthaven

luchthaven	variabele	mediaan
EWR	vluchtype	kort
JFK	vluchtype	kort
LGA	vluchtype	kort

Table 3.15: Kwantilen voor vluchtype-luchthaven

luchthaven	variabele	Q25	Q50	Q75
EWR	vluchtype	kort	kort	normaal
JFK	vluchtype	kort	kort	lang
LGA	vluchtype	kort	kort	kort

3.5.3 Categorisch versus Categorisch



Univariate statistieken per categoriewaarde

- Je toont de relevante centrum- en spreidingsmaten voor de afhankelijke continue variabele per waarde van de onafhankelijke categorische variabele.

Referenties

1. Tekst Beleidsstatistiek: Hoofdstukken 1 en 2 en secties 4.2 en 4.3 (Blackboard)
2. Spearman's rangcorrelatiecoëfficiënt
3. Kendall's rangcorrelatiecoëfficiënt
4. Spearman versus Kendall's correlatiecoëfficiënt

Chapter 4

Exploratief Data Analyse Proces

Voor dit topic zijn er geen aparte lecture nodes, maar wordt verwezen naar de eerste 4 hoofdstukken in “The Art of Data Science” van Roger Peng en Elizabeth Matsui (tot en met hoofdstuk ‘Exploratory Data Analysis’). Dit boek kan men gratis bekomen via volgende link: <https://leanpub.com/artofdatascience> .

Let op: Boven aan de pagina wordt het boek + video lectures aangeboden voor een minimum prijs van 20 dollar. Maar als je naar onder scrollt vind je ook een link naar enkel het boek waarvoor de minimumprijs 0 dollar is. Als je hier op klikt kan je vervolgens zelf je prijs bepalen (en dus ook op 0 dollar zetten) om vervolgens het boek gratis aan te schaffen.

Chapter 5

Datavoorbereiding

5.1 Beginnen bij het begin

- Alvorens we aan een exploratieve data analyse kunnen beginnen, moeten we eerst onze data voorbereiden.
- Er kunnen drie grote fases geïdentificeerd worden tijdens de datavoorbereiding.
 - Correct inlezen van de data.
 - Identificeren van problemen in de data en deze corrigeren van de data.
 - Opwaarderen van de data.
- Data kan in diverse formaten aangeleverd worden en de eerste stap is ervoor zorgen dat de data ingeladen is in R. Hierbij zijn er twee specifieke elementen om aandacht aan te schenken:
 - De data analyst moet ervoor zorgen dat de data correct ingeladen wordt en dat de inhoud na het inladen overeenkomt met de inhoud toen deze data de laatste keer werd opgeslagen.
 - De data analyst moet ervoor zorgen dat de verschillende variabelen het juiste data type hebben.
- De volgende fase is het opkuisen van de data. Dit betekent dat men fouten gaat identificeren en deze ‘oplost’ alvorens verder te gaan.
 - Er zijn verschillende soorten fouten die in de data kunnen sluipen. Enkele mogelijke fouten zijn:
 - * Sommige waarden ontbreken (geen waarde voor bepaalde variabele bij bepaalde observaties).
 - * Sommige waarden zijn fout. Bijvoorbeeld: voor een deel observaties is de afstand in km opgeslagen ipv mijl of is er een typfout in de waarde van een categorische variabele.
 - * Sommige observaties staan meerdere keren in de dataset.
 - Het opkuisen van data gebeurt in principe in 2 stappen:
 - * Eerst moeten we de data bestuderen en fouten identificeren.
 - * Vervolgens moeten we de fouten in de data ‘corrigeren’ (indien mogelijk).
- Het opwaarderen van de data betreft een reeks transformaties met als doel de data bruikbaarder te maken voor exploratieve data analyse. Er zijn verschillende manieren om dit te bereiken:
 - Bestaande variabelen transformeren naar nieuwe variabelen die geschikter zijn om patronen in de data bloot te leggen. Bijvoorbeeld het transformeren van een continue variabele naar een categorische variabele of het creëren van een nieuwe variabele ‘gemiddelde snelheid’ op basis van de variabelen ‘reistijd’ en ‘totaal afgelegde afstand’.
 - Het opsplitsen van de dataset in meerdere datasets die apart bestudeerd worden. Dit is vooral zinvol indien de dataset verschillende soorten observaties bevat of een deel observaties met uitzonderlijke waarden.

5.2 Data inlezen

5.2.1 Uitdagingen bij het correct inlezen van data

- Data kan in verschillende formaten aangeleverd worden. Afhankelijk van het formaat, zal je andere functies moeten gebruiken om de data correct in te lezen
- Maar zelfs als je de juiste functie gebruikt, kan het inlezen fout gaan omdat een computer data altijd als een reeks van 1 en 0'en opslaat en er daarom een soort vertaalsleutel nodig is van 1 en 0'en naar leesbare tekst. Deze vertaalslag wordt gerealiseerd door encodingschema's en je moet er voor zorgen dat bij het inlezen van data je het juiste encodingschema hanteert.
- Eenmaal de data is ingeladen, moet je er voor zorgen dat R de datatypes juist identificeert. De meeste dataformaten houden geen informatie bij van welk datatype een specifieke variabele is en dus moet R dit 'raden'. Omdat dit wel eens fout kan gaan, moet je als analist dit controleren en corrigeren waar nodig.

5.2.2 Dataformaten

5.2.2.1 Flat-file databestanden

- Lees de bron over [delimited en fixed-width bestanden](#).
- Flat-file databestanden bevatten data die in een tabelvorm passen:
 - Iedere rij is een observatie.
 - Iedere kolom stelt een variabele voor.
 - Alle items in een kolom zijn van dezelfde soort.
 - Cellen van de tabel bevatten enkelvoudige gegevens (dus niet een kolom hobby's met hierin meerdere hobby's in 1 cel).
 - De volgorde van de kolommen is niet van belang.
 - De volgorde van de rijen is niet belangrijk.
- Volgende data kan dus in een flat-file bestand opgeslagen worden:

naam	voornaam
Nelissen	Rob
Franssen	Ann

- De twee meest gebruikte formaten voor flat-file databestanden zijn delimited en fixed-width bestanden.
- Een delimited bestand (vaak ook wel csv-bestand of comma-separated values bestand genoemd):
 - gebruikt voor iedere rij een nieuwe regel,
 - splitst de kolommen op met behulp van een specifiek splitsingsteken (vaak de komma of de puntkomma),
 - kan het begin en het einde van een karakterstring met behulp van een specifiek quote-teken (vaak ' of ") aanduiden.
- Bovenstaande tabel kan als een delimited databestand opgeslagen worden en ziet er dan als volgt uit:

```
naam;voornaam
Nelissen;Rob
Franssen;Ann
```

- Een fixed-width bestand gebruikt eveneens een aparte regel per rij, maar gebruikt een vast aantal karakters per kolom en heeft dus geen splitsingsteken, noch quote-teken nodig.
- Bovenstaande tabel kan als een fixed-width databestand opgeslagen worden en ziet er dan als volgt uit:

```

naam      voornaam
Nelissen Rob
Franssen Ann

```

- Het nadeel van een delimited bestand is dat je het splitsings- en quote-teken niet kunt gebruiken in je data.
- Het voordeel van een delimited bestand is dat een veld niet meer ruimte in beslag neemt dan nodig.
- Bestudeer hoofdstuk [Data Import](#) van het boek ‘R for Data Science’ om te weten hoe je in R data uit flat-file bestanden kunt inlezen.

5.2.2.2 Hiërarchische databestanden

- Het nadeel van flat-file data bestanden is dat de data in een tabelvorm moet passen.
- Indien data een complexere (vaak hiërarchische) structuur heeft, dan is dit niet evident om correct in een tabelvorm te gieten.
 - Je hebt bijvoorbeeld data over de studenten en van iedere student heb je naamgegevens en de resultaten van de verschillende afgelegde vakken.
 - Het aantal afgelegde vakken verschilt echter van student tot student.
 - Ook per student kan het aantal opgenomen kansen per vak verschillen van vak tot vak.
 - Het aantal scores per student kan hierdoor sterk variëren.
- Voor hiërarchische databestanden wordt daarom vaak gebruikt gemaakt van XML-bestanden of JSON-bestanden.
- XML- en JSON-bestanden zijn ook zeer populair om gegevens via het web uit te wisselen.

5.2.2.2.1 XML-bestanden

- Bestudeer de bron over [XML](#) (tot en met XML attributes) om te begrijpen hoe een XML-bestand is opgebouwd.
- Een XML-bestand bestaat uit XML-elementen.
 - De naam van het XML-element wordt bepaald door het openings- en sluitingslabel.
 - Het openingslabel volgt het formaat `<element-naam>`.
 - Het sluitingslabel heeft dezelfde naam en volgt het formaat `</element-naam>`.
 - Tussen het openings- en sluitingslabel plaatsen we de inhoud van het XML-element.
- Voorbeeld: `<student>Rob Nelissen</student>`.
 - Hier wordt het XML-element student gedefinieerd.
 - De inhoud van dit XML-element is Rob Nelissen.
- De inhoud van een XML-element kan ook bestaan uit andere XML-elementen. Op deze manier kan je volledige tabellen in XML opslaan. Onderstaande voorbeeld is de vertaling van voorgaande data in tabelvorm naar XML.
- Voorbeeld:

```

<studenten>
  <student>
    <naam>Nelissen</naam>
    <voornaam>Rob</voornaam>
  </student>
  <student>
    <naam>Franssen</naam>
    <voornaam>Ann</voornaam>

```

```
</student>
</studenten>
```

- Zoals blijkt uit de vergelijking tussen de XML-representatie en voorgaande flat-file representaties, bevat een XML-bestand redelijk veel overhead om tabelvorm-data op te slaan.
- De kracht van XML ten opzichte van de flat-file bestanden is echter dat je veel complexere datastructuren kunt opslaan. Onderstaand voorbeeld opslaan in een tabelvorm (en dus flat-files) is allesbehalve evident.
- Voorbeeld:

```
<studenten>
  <student>
    <naam>Nelissen</naam>
    <voornaam>Rob</voornaam>
    <vakken>
      <vak>
        <naam>Exploratieve en Descriptieve Data Analyse</naam>
        <academiejaar>20162017</academiejaar>
        <score_kans1>8</score_kans1>
        <score_kans2>12</score_kans2>
      </vak>
      <vak>
        <naam>Macro-economie</naam>
        <academiejaar>20152016</academiejaar>
        <score_kans1>8</score_kans1>
        <score_kans2>7</score_kans2>
      </vak>
      <vak>
        <naam>Macro-economie</naam>
        <academiejaar>20162017</academiejaar>
        <score_kans1>14</score_kans1>
      </vak>
      ...
    </vakken>
  </student>
  <student>
    <naam>Franssen</naam>
    <voornaam>Ann</voornaam>
    <vakken>
      <vak>
        <naam>Exploratieve en Descriptieve Data Analyse</naam>
        <academiejaar>20162017</academiejaar>
        <score_kans1>15</score_kans1>
      </vak>
      <vak>
        <naam>Macro-economie</naam>
        <academiejaar>20162017</academiejaar>
        <score_kans1>16</score_kans1>
      </vak>
      ...
    </vakken>
  </student>
```

```
...
</studenten>
```

5.2.2.2.2 JSON-bestanden

- Bestudeer de [JSON Tutorial](#) om te begrijpen hoe een JSON-bestand is opgebouwd.
- JSON is een ander formaat dat steeds populairder wordt om hiërarchische data op te slaan en uit te wisselen.
 - In vergelijking met XML is JSON korter en eenvoudiger te lezen.
- Een JSON-bestand bestaat voornamelijk uit JSON-objecten en JSON-lijsten.
- JSON-objecten komen typisch overeen met een observatie (rij) in een dataset.
 - Een JSON-object wordt omsloten door accolades.
 - De inhoud van een JSON-object bestaat uit key-value paren.
 - * De key-value paren zijn van elkaar gescheiden door middel van een komma.
 - * De key is een string omsloten door dubbele aanhalingstekens en geeft aan wat de value voorstelt.
 - * Key en value zijn van elkaar gescheiden door middel van een dubbelpunt.
- Voorbeelden van een JSON-object dat de student Rob Nelissen voorstelt:
 - `{"naam": "Nelissen", "voornaam": "Rob"}`
- Een JSON-lijst (array genoemd) bestaat uit een lijst van waarden gescheiden door een komma en omgeven door rechte haakjes.
 - De waarden van de verschillende elementen in een JSON-lijst moeten van hetzelfde type zijn.
 - Toegelaten waarden zijn o.a. strings, getallen, objecten, andere arrays (lijsten).
- Indien je data in tabelvorm wenst voor te stellen, zal je iedere rij als een JSON-object voorstellen en de volledige tabel als een lijst van deze JSON-objecten.
- Onderstaand voorbeeld is de vertaling van voorgaande studentendata in tabelvorm naar JSON.

```
[{"naam": "Nelissen",
 "voornaam": "Rob"
},
 {"naam": "Franssen",
 "voornaam": "Ann"
}]
```

- Net als bij XML kan je met JSON complexe datastructuren voorstellen.
- Voorbeeld:

```
[{"naam": "Nelissen",
 "voornaam": "Rob",
 "vakken": [
   {"naam": "Exploratieve en Descriptieve Data Analyse",
    "academiejaar": "20162017",
    "score_kans1": 8,
    "score_kans2": 12
  },
  ...]
```

```

    {"naam": "Macro-economie",
     "academiejaar": "20152016",
     "score_kans1": 8,
     "score_kans2": 7
   },
   {"naam": "Macro-economie",
     "academiejaar": "20162017",
     "score_kans1": 14
   }
 ]
},
{"naam": "Franssen",
 "voornaam": "Ann",
 "vakken": [
   {"naam": "Exploratieve en Descriptieve Data Analyse",
     "academiejaar": "20162017",
     "score_kans1": 15
   },
   {"naam": "Macro-economie",
     "academiejaar": "20162017",
     "score_kans1": 16
   }
 ]
}
]

```

5.2.2.3 Applicatie-specifieke dataformaten

- Naast deze standaard dataformaten, waarbij data als tekstbestanden worden opgeslagen, bestaan er verschillende applicatie-specifieke dataformaten.
- Het voordeel van applicatie-specifieke dataformaten is dat deze extra informatie over de data kunnen opslaan die specifiek is voor de applicatie.
 - Zo zal een Excel databestand ook informatie bevatten over de formules en opmaak van de data (vet, cursief, ...).
- Het nadeel van deze applicatie-specifieke dataformaten is dat ze niet altijd leesbaar zijn door andere applicaties.

5.2.2.4 R-packages voor het inladen van diverse dataformaten

- Standaard R heeft een aantal functies om delimited bestanden in te lezen, namelijk `read.csv()` en `read.csv2()`. Het verschil tussen beide functies heeft betrekking op de standaardwaarden voor specifieke parameters, zoals welk teken als delimiter gebruikt wordt (`read.csv` veronderstelt het komma-teken als delimiter, terwijl `read.csv2` er van uitgaat dat de puntkomma gebruikt wordt om waarden van elkaar te scheiden).
- Er is ook het R pacakge ‘`readr`’ dat ook twee soortgelijke functies aanbiedt - `read_csv()` en `read_csv2()` - die performanter zijn dan de standaardfuncties.
- Om xml-bestanden in te lezen, wordt typisch het R package ‘`xml`’ gebruikt. Voor JSON-bestanden kan gebruik gemaakt worden van de packages ‘`rjson`’ of ‘`jsonlite`’.
- Voor een aantal applicatie-specifieke formaten zijn ondertussen ook al R-packages ontwikkeld. Zo is er bijvoorbeeld het R-package ‘`readxl`’ dat het inladen van Excel bestanden relatief eenvoudig maakt.

5.2.3 Data-encodering

5.2.3.1 Binair, decimaal en hexadecimaal rekenstelsel

- Een computer kan slechts 2 waarden opslaan, typisch voorgesteld als 0 en 1.
- Iedere opslaglocatie op een computer kan dus slechts 2 verschillende waarden opslaan en wordt een bit genoemd.
 - De afkorting van bit is de kleine letter ‘b’.
- Een rekenstelsel waarbij iedere locatie slechts 2 waarden kan voorstellen noemen we een binair rekenstelsel.
 - Indien we 2 opslaglocaties (2 bits) gebruiken, kunnen we 4 verschillende waarden opslaan: 00, 01, 10 en 11.
 - Indien we 3 bits gebruiken zijn er 8 mogelijke waarden, bij 4 bits zijn er 16 mogelijke waarden.
 - Het aantal waarden dat men met n bits kan opslaan is gelijk aan 2^n .
- Merk op dat in het rekenstelsel dat door mensen gebruikt wordt iedere opslaglocatie 10 verschillende waarden gebruikt kunnen worden (0, 1, 2, 3, 4, 5, 6, 7, 8, 9).
 - Dit noemen we het decimaal rekenstelsel.
 - Met 2 opslaglocaties kunnen we in het decimaal rekenstelsel 100 ($= 10^2$) waarden opslaan: van 00 tot 99.
- Een ander rekenstelsel dat vaak gebruikt wordt binnen computerwetenschappen is het hexadecimaal rekenstelsel.
 - In dit rekenstelsel kan iedere opslaglocatie 16 waardes opslaan, nl. 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E en F.
 - Het hexadecimaal rekenstelsel is interessant omdat 1 locatie overeenstemt met 4 bit (4 locaties in het binair rekenstelsel).
- Om in computerprogramma’s aan te geven dat iets voorgesteld wordt in het hexadecimaal stelsel, laten we het voorafgaan door een 0x. Om aan te geven dat iets voorgesteld wordt in het binair stelsel, gebruiken we prefix 0b. Zonder prefix verwijzen we typisch naar het decimaal rekenstelsel.

decimaal	hexadecimaal	binair
0	0x0	0b0000
1	0x1	0b0001
2	0x2	0b0010
3	0x3	0b0011
4	0x4	0b0100
5	0x5	0b0101
6	0x6	0b0110
7	0x7	0b0111
8	0x8	0b1000
9	0x9	0b1001
10	0xA	0b1010
11	0xB	0b1011
12	0xC	0b1100
13	0xD	0b1101
14	0xE	0b1110
15	0xF	0b1111

Tabel: conversietabel decimaal, hexadecimaal en binair.

- 8 bits worden ook een byte genoemd wat afgekort wordt met de hoofdletter B. 1B bestaat dus uit 8b en kan dus 256 ($= 2^8$) waarden opslaan.

5.2.3.2 Tekst opslaan

- Bestudeer de bron over [Encoding](#) tot sectie ‘Encodings en PHP’.
- Lees de bron over [de geschiedenis van ASCII](#) (enkel sectie ‘A Historical Perspective’).
- Aangezien computers enkel bits kunnen opslaan, hebben we een conversieschema nodig om tekst op te slaan. Ieder letterteken zal moeten omgezet worden naar een string van bits. Dit conversieschema wordt een ‘encoding scheme’ of encodingsschema genoemd.
- Ieder encodingsschema voorziet de vertaling van een specifieke set van karakters naar bijhorende bitstrings. Deze sets van karakters noemen we ‘character sets’ of karaktersets.
- Er bestaan zeer veel encodingsschema’s.
 - ASCII is een van de oudste encodingsschema’s en is voornamelijk bruikbaar voor Engelstalige tekst.
 - * De ASCII karakterset bestaat uit 128 lettertekens en bevat o.a. de cijfers 0 tot 9, de letters a-z en A-Z.
 - * ASCII gebruikt 1 byte per letterteken en kan dus in principe 256 verschillende lettertekens opslaan.
 - * Aangezien ASCII slechts een karakterset van 128 tekens heeft, gebruikt het dus slechts 7 bit van de beschikbare byte ($2^7 = 128$).
 - * Omdat de ASCII tekenset gemaakt was voor de Engelse taal ontbreken er verschillende tekens voor andere talen.
 - De ANSI-standaard nam de ASCII tekenset over, maar voegt hier vervolgens 128 tekens aan toe door de volledige byte te gebruiken.
 - * Met welke tekens de karakterset wordt uitgebreid ligt echter niet vast binnen de ANSI-standaard, maar is afhankelijk van de gekozen codepage (of karakterset).
 - * Er bestaan zeer veel ANSI codepages (die eigenlijk Windows codepages genoemd moeten worden). Voor de eerste 128 tekens maakt de specifieke codepage niet uit, maar voor de laatste 128 code pages is dit wel belangrijk.
 - Voor talen waar 1 byte per letterteken onvoldoende is, werden dan weer nieuwe encodingsschema’s gebruikt die 2 bytes gebruiken en zo 65536 lettertekens kunnen voorstellen.
 - Unicode is een poging om tot 1 karakterset te komen voor alle tekens die gebruikt worden in tekst.
 - * Unicode is een standaard en definieert zelf geen encoding. Ze vertaalt dus zelf geen lettertekens naar bitstrings.
 - * Unicode legt wel codepoints vast, wat een mapping is tussen lettertekens en een hexadecimaal getal. Zo is de letter ‘A’ gekoppeld aan het codepoint 0x0041.
 - * Het Unicode systeem bevat in totaal meer dan 1 miljoen codepoints en omvat niet enkel cijfers en letters, maar ook emoji’s. Zo heeft de ‘lachend gezicht’-emoji codepoint 0x1F642.
 - * De feitelijke omzetting van de codepoints naar een bitstring gebeurt door een specifieke encodering, waarbij UTF-8 de meest voorkomende is.
- Het gevolg is dat we een grote waaier aan encodingsschema’s hebben.
 - Als je dus een tekst opslaat volgens het ene encodingsschema en vervolgens terug inleest volgens een ander encodingsschema, dan kan het zijn dat delen van de tekst geen steek meer houden.
 - Als je bijvoorbeeld de letter ‘í’ opslaat volgens de Windows-1252 codepage dan zal dit binair als 11101111 opgeslagen worden (0xEF). Als je deze reeks van 8 bits echter later weer inleest volgens de Windows-1257 (Windows-Baltic), dan zal de binaire reeks 11101111 geïnterpreteerd worden als ‘í’.
- Het R-package ‘readr’ gaat er van uit dat tekst geëncodeerd is in UTF-8.

5.2.4 Datatypes controlleren en corrigeren

- We zullen de datavoorbereidingsfase illustreren aan de hand van de vluchtgegevens van de drie luchthavens in New York City.
- Voor we kunnen beginnen is het altijd verstandig een snel overzicht te maken van de dataset, zodat we weten welke variabelen we voor handen hebben alsook hun datatypes.
- Met de glimpse functie krijg je snel een overzicht van de verschillende variabelen en van welk type ze zijn.

```
## Observations: 329,174
## Variables: 7
## $ luchthaven      <fct> EWR, LGA, JFK, LGA, EWR, LGA, JFK, LG...
## $ maatschappij    <fct> United Air Lines Inc., United Air Lines In...
## $ vertrek_vertraging <dbl> 2, 4, 2, -6, -4, -5, -3, -3, -2, -2, -2, -...
## $ aankomst_vertraging <dbl> 11, 20, 33, -25, 12, 19, -14, -8, 8, -2, -...
## $ afstand          <chr> "1400", "1416", "1089", "762", "719", "106...
## $ vliegtijd        <dbl> 227, 227, 160, 116, 150, 158, 53, 140, 138...
## $ vluchttype       <ord> normaal, normaal, kort, kort, kort, ...
```

- Deze output bevat al een opmerkelijk resultaat. Zo zien we dat de variabele ‘afstand’ als tekst is opgeslagen in plaats van als een continue (numerieke) variabele.
- Soms gebeurt het dat R niet het juiste variabeletype herkent. Zo kan het zijn dat een categorische variabele als numerieke variabele wordt beschouwd omdat de categorieën gehele getallen zijn (vb. aantal cylinders: 4, 6 of 8).
- In onze dataset hebben we opgemerkt dat de variabele ‘afstand’ niet als numerieke variabele wordt geïnterpreteerd, maar als een ‘tekst’-variabele. Dit kan verschillende oorzaken hebben.
 - Zo kan het zijn dat er voor 1 van de observaties een waarde geregistreerd is met een niet-numeriek teken (vb. 1OO ipv 100). In dat geval zal je eerst deze waarden moeten corrigeren naar de juiste waarde.
 - Een andere vaak voorkomende oorzaak is dat R een punt als decimaalteken verwacht, terwijl dat een komma is in de dataset. Dit valt vaak op te lossen door de data met andere opties in te lezen in R.
- Indien de fouten zijn gecorrigeerd, dan moeten we nog altijd de data omzetten van een ‘tekst’-variabele naar een numerieke variabele (in ons geval). Dit doen we door middel van de ‘mutate’-functie en de ‘as.numeric’-functie.
- Merk op dat als er toch nog een waarde aanwezig is die niet kan omgezet worden naar het nieuwe variabeletype, R een waarschuwing zal geven en de waarde zal vervangen door ‘NA’. In dat geval ga je eerst moeten zoeken naar de oorzaak van de waarschuwing, deze aanpakken en dan de variabele transformeren naar het nieuwe type.
- De belangrijkste datatypes in R en de bijhorende transformatiefuncties zijn:
 - numeric (decimale getallen) - as.numeric()
 - integer (gehele getallen) - as.integer()
 - character (tekst) - as.character()
 - factor (nominale variabele) - as.factor()
 - ordered factor (ordinale variabele) - as.ordered()

Table 5.3: Maatschappijen geordend volgens stijgende frequentie.

maatschappij	n
XpressJet Airlines Inc.	1
Envoi Air	19
SkyWest Airlines Inc.	32
Hawaiian Airlines Inc.	342
Mesa Airlines Inc.	601
Frontier Airlines Inc.	685
Alaska Airlines Inc.	714
AirTran Airways Corporation	3260
Virgin America	5162
Southwest Airlines Co.	12275
Endeavor Air Inc.	18460
US Airways Inc.	20536
Envoy Air	26378
American Airlines Inc.	31327
Delta Air Lines Inc.	46779
JetBlue Airways	50940
ExpressJet Airlines Inc.	54172
United Air Lines Inc.	57491

5.3 Dataproblemen identificeren en corrigeren

5.3.1 Overzicht

- We onderscheiden drie soorten problemen die kunnen opduiken met data en die best op voorhand gecorrigeerd worden:
 - Foutieve waarden.
 - Ontbrekende waarden.
 - Inconsistente waarden.

5.3.2 Foutieve waarden

5.3.2.1 Categorische variabele

- Coderingsfouten bij categorische variabelen uiten zich typisch in redundante categorielabels. Dit zijn labels met een typfout die door R als een aparte categorie worden beschouwd, maar dit niet zijn.
- Om dit soort coderingsfouten te detecteren, moet je de verschillende labels van een categorische variabele bestuderen.
 - Omdat deze foute categorielabels meestal uitzonderlijk zijn, kan je best de verschillende categorielabels bekijken volgens stijgende frequentie.
 - Een andere aanpak is de categorielabels alfabetisch te ordenen.
- Eenmaal men deze coderingsfouten gedetecteerd heeft, kan men ze manueel corrigeren door gebruik te maken van de functies mutate (dplyr) en fct_recode (forcats).

Case: Vluchtdata NYC

- Als we de categorische variabele maatschappij analyseren op foutieve labels dan zien we dat 1 vlucht foutief het label ‘XpressJet Airlines Inc.’ heeft gekregen in plaats van ‘ExpressJet Airlines Inc.’.

Table 5.4: Maatschappij alfabetisch geordend.

maatschappij	n
AirTran Airways Corporation	3260
Alaska Airlines Inc.	714
American Airlines Inc.	31327
Delta Air Lines Inc.	46779
Endeavor Air Inc.	18460
Envoi Air	19
Envoy Air	26378
ExpressJet Airlines Inc.	54172
Frontier Airlines Inc.	685
Hawaiian Airlines Inc.	342
JetBlue Airways	50940
Mesa Airlines Inc.	601
SkyWest Airlines Inc.	32
Southwest Airlines Co.	12275
United Air Lines Inc.	57491
US Airways Inc.	20536
Virgin America	5162
XpressJet Airlines Inc.	1

- Indien we de labels van de categorische variabele maatschappij alfabetisch ordenen dan zien we ook dat er een aantal vluchten foutief gecodeerd zijn als ‘Envoi Air’ in plaats van ‘Envoy Air’.
- We kunnen deze foutieve labels corrigeren met behulp van de functie `fct_recode` uit het `forcats` package.

5.3.2.2 Ordinale variabelen

- Bij ordinale variabelen kunnen dezelfde coderingsfouten voorkomen als bij categorische variabelen. Deze worden op dezelfde manier gedetecteerd en gecorrigeerd.
- Er is echter nog een bijkomende coderingsfout voor ordinale variabelen, namelijk wanneer de voorgedefinieerde volgorde tussen de labels fout is.
- Om dit te detecteren, moet je de verschillende labels ('levels') opvragen met behulp van de `unique`-functie.
- Indien we de ordinale variabele vluchtype analyseren dan zien we dat de voorgedefinieerde volgorde van de labels foutief is.

```
## [1] normaal      kort       lang       intercontinentaal
## Levels: lang < kort < normaal < intercontinentaal
```

- We kunnen de volgorde tussen de labels van een ordinale variabele corrigeren met behulp van de functies `mutate` (`dplyr`) en `fct_relevel` (`forcats`).

```
## [1] normaal      kort       lang       intercontinentaal
## Levels: kort < normaal < lang < intercontinentaal
```

5.3.2.3 Continue variabelen

- Foutieve waarden bij een continue variabelen detecteren is een stuk moeilijker omdat een foutieve waarde nog steeds een geldige waarde kan zijn (nog steeds een getal).

Table 5.5: Vluchten met kortste vliegtijd.

luchthaven	maatschappij	afstand	vliegtijd
JFK	Endeavor Air Inc.	94	0.5833333
EWR	JetBlue Airways	1065	2.7666667
JFK	Delta Air Lines Inc.	2248	5.1500000
EWR	ExpressJet Airlines Inc.	116	20.0000000
EWR	ExpressJet Airlines Inc.	116	20.0000000
EWR	ExpressJet Airlines Inc.	116	21.0000000
EWR	ExpressJet Airlines Inc.	80	21.0000000
EWR	ExpressJet Airlines Inc.	116	21.0000000
EWR	ExpressJet Airlines Inc.	80	21.0000000
LGA	US Airways Inc.	184	21.0000000

- Ook de aanpak om naar weinig voorkomende waarden te kijken, zoals bij categorische variabelen, werkt niet goed omdat bij een continue variabele vaak veel waarden zijn zeer weinig voorkomen.
- De meest voor de hand liggende aanpak is de waarden te bestuderen die opmerkelijk hoog of laag zijn in vergelijking met de andere waarden van de variabele.
- Het is belangrijk te beseffen dat niet iedere extreme waarde per definitie een foutieve waarde is. Uitzonderlijk hoge of lage waarden zijn natuurlijk altijd mogelijk.
- Daarom moet men altijd voorzichtig te werk gaan bij het bepalen of iets een foutieve waarde is (meetfout, ingavefout) of een uitzonderlijke doch correcte waarde. Domeinkennis kan hierbij helpen.
- Indien je door te kijken naar de uiterste waarden mogelijke problemen hebt gedetecteerd, moet je deze observaties van nabij bestuderen om te achterhalen of het meetfouten kunnen zijn of niet. Ga hiervoor steeds naar de volledige observatie kijken en niet enkel naar de waarde voor de continue variabele.
- Een andere manier om te detecteren of een continue variabele uitzonderlijke waarden bevat, is door middel van een boxplot. Uitzonderlijk grote/kleine waarden vallen buiten de ‘whiskers’ en worden door punten aangeduid in een boxplot. Let wel op, de filosofie achter uitzonderlijke waarden is gebaseerd op een normale verdeling van de data. Indien de data werkelijk normaal verdeeld is, dan is de kans op een uitzonderlijke waarde slechts 0.7%. Dit betekent echter dat men best op voorhand het histogram bekijkt om te controleren of de data enigszins normaal verdeeld is, alvorens de boxplot te hanteren.
- Bij foutieve waarden van een continue variabele is het vaak niet mogelijk om de correcte waarde af te leiden (zoals bij een categorische variabele). Daarom is de enige juiste correctie deze foutieve waarden te vervangen met “missing values”.
- We zullen de variabele vliegtijd analyseren op foutieve waarden.
- We zullen eerst de kleinste waarden bestuderen. Hiervoor selecteren we de 10 vluchten met de kortste vliegtijd en rangschikken deze volgens stijgende vliegtijd. We kijken hierbij niet alleen naar de vliegtijd, maar ook naar de luchthaven, de maatschappij en de afgelegde afstand.
- Deze analyse doet vermoeden dat de eerste drie vluchten waarschijnlijk meetfouten zijn. Het betreffen hier drie vluchten van minder dan 6 minuten wat zeer onwaarschijnlijk is, zeker wanneer we zien dat de tweede en derde vlucht lange vluchten zijn.
- De overige vluchten zijn vluchten van 20 minuten of meer, maar aangezien het hier om korte vluchten gaan is dit mogelijk correct. We zullen enkel de eerste drie observaties (met een vliegtijd kleiner dan 6) als foutief beschouwen.
- We bestuderen vervolgens de vluchten met de grootste vliegtijd.
- Deze resultaten doen vermoeden dat het hier NIET om meetfouten gaat. Het gaat hier immers om zeer verre vluchten en de maatschappijnaam doet vermoeden dat het hoofdzakelijk vluchten naar Hawaï zijn. We hebben daarom via Google opgezocht hoe lang een vlucht van New York naar Hawaï duurt en dit komt overeen met de vliegtijden van 11 tot 12u in deze dataset. Daarom besluiten we dat deze waarden geen foutieve waarden zijn.

Table 5.6: Vluchten met langste vliegtijd.

luchthaven	maatschappij	afstand	vliegtijd
EWR	United Air Lines Inc.	4963	695
JFK	Hawaiian Airlines Inc.	4983	691
JFK	Hawaiian Airlines Inc.	4983	686
JFK	Hawaiian Airlines Inc.	4983	686
JFK	Hawaiian Airlines Inc.	4983	683
JFK	Hawaiian Airlines Inc.	4983	679
EWR	United Air Lines Inc.	4963	676
JFK	Hawaiian Airlines Inc.	4983	676
JFK	Hawaiian Airlines Inc.	4983	675
EWR	United Air Lines Inc.	4963	671

Table 5.7: Vluchten met kortste vliegtijd.

luchthaven	maatschappij	afstand	vliegtijd
EWR	ExpressJet Airlines Inc.	116	20
EWR	ExpressJet Airlines Inc.	116	20
EWR	ExpressJet Airlines Inc.	116	21
EWR	ExpressJet Airlines Inc.	80	21
EWR	ExpressJet Airlines Inc.	116	21
EWR	ExpressJet Airlines Inc.	80	21
LGA	US Airways Inc.	184	21
JFK	Endeavor Air Inc.	94	21
EWR	ExpressJet Airlines Inc.	116	21
EWR	ExpressJet Airlines Inc.	116	21

- Vervolgens zullen we voor de drie vluchten met een vliegtijd van minder dan 6 minuten de waarde van de vliegtijd vervangen door een ontbrekende waarde. In R wordt dit aangegeven door de waarde NA dat voor ‘not available’ staat.

5.3.3 Ontbrekende waarden

- Soms gebeurt het dat voor bepaalde observaties waarden ontbreken voor een specifieke variabele. In zulke gevallen spreken we van ontbrekende waarden of missing values.
- Het detecteren van ontbrekende waarden is relatief eenvoudig, omdat deze normaal als NA gecodeerd zijn in een dataset (NA = ‘not available’).
- We kunnen onderscheid maken tussen drie soorten van ontbrekende waarden.
 - Missing completely at random (MCAR): Indien het ontbreken van waarden voor een specifieke variabele volledig willekeurig is, dan spreekt men over MCAR.
 - Missing at random (MAR): Indien het ontbreken van waarden voor variabele X_1 niet willekeurig is, maar afhankelijk van de waarden van andere variabelen X_2, X_3, \dots , dan spreekt men over MAR.
 - Not missing at random (NMAR): Indien het ontbreken van waarden voor variabele X_1 niet willekeurig is, maar afhankelijk is van de waarde van X_1 of van de waarden van ongeobserveerde variabelen, dan spreekt men over NMAR.
- Om te bepalen of data al dan niet MCAR is, moet men achterhalen of het ontbreken van waarden voor variabele X_1 gecorreleerd is met de waarden van een andere variabele X_2 . Een mogelijkheid is om de dataset in twee te splitsen, i.e. alle observaties met een waarde voor X_1 en alle observaties met een missing value voor X_1 . Vervolgens kijken we naar de verdeling van variabele X_2 . Indien

deze hetzelfde is voor beide datasets, dan suggereert dit dat er geen relatie bestaat tussen de waarde van X_2 en het al dan niet ontbreken van de waarde voor X_1 . Indien deze verdeling van X_2 sterkt verschilt tussen beide datasets, dan is er mogelijk wel een relatie en dan is de data niet MCAR.

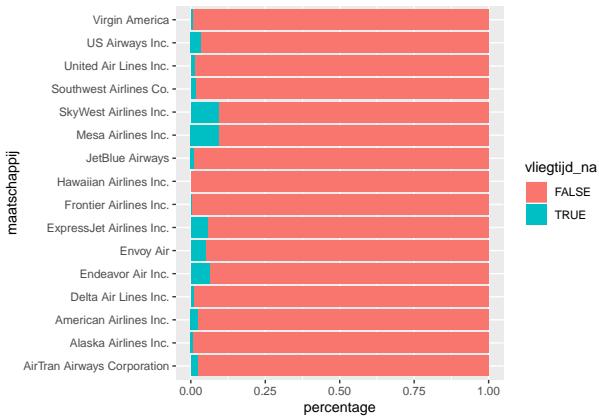
- Het soort ontbrekende waarde heeft belangrijke implicaties hoe je correct met ontbrekende waarden kan omgaan in het kader van confirmatorische data analyse. Zo zal het ‘weglaten’ van observaties met ontbrekende waarden enkel in het geval van MCAR geen vertrekking geven in de resultaten van een confirmatorische data analyse.
- In het kader van een descriptieve of exploratieve data analyse, zijn de implicaties eerder beperkt, omdat men toch enkel uitspraken wenst te doen voor de beschikbare data.
- Wel kan het identificeren van het type ontbrekende waarden op zich interessante inzichten geven. Zo kan het het patroon dat het jaarsalaris voornamelijk ontbreekt bij mensen die hogere studies gevolgd hebben op zich ook interessant zijn voor verdere interpretatie.
- In descriptieve en exploratieve analyses zijn er 3 manieren om met ontbrekende waarden om te gaan:
 - We verwijderen de variabele waarvoor we missing values hebben.
 - We verwijderen de observaties met missing values.
 - We beschouwen de missing values als een aparte waarde.
- Het verwijderen van de variabele zelf is een drastische maatregel. Dit betekent immers dat we de variabele volledig buiten beschouwing laten in onze analyse. Dit is vaak het laatste redmiddel en wordt enkel toegepast als er een te hoog percentage ontbrekende waarden is.
- Bij het verwijderen van observaties moet men met de nodige aandacht te werk gaan. Indien de ontbrekende waarden MAR zijn (en niet MCAR), dan gaan men mogelijk waardevolle patronen tussen andere variabelen ook verwijderen. Een mogelijke manier om dit te omzeilen is de observaties met ontbrekende waarden te negeren bij analyses van de variabelen waarvoor de waarden ontbreken. Dit zorgt ervoor dat deze observaties wel nog beschikbaar zijn voor de analyse van andere variabelen.
- Indien de data suggereert dat de ontbrekende waarden MCAR zijn, dan kan men overwegen deze observaties te verwijderen. Indien dit niet het geval is, dan is het beter deze NA-waarden als een aparte categorie te beschouwen.
 - Omdat R de waarde ‘NA’ anders behandelt dan reguliere waarden, is het vaak aangeraden om deze waarde te transformeren (indien je de ontbrekende waarden als een aparte categorie wenst te beschouwen).
 - In geval van een categorische variabele, kan je de ‘NA’ waarde transformeren naar een aparte categorie (vb ‘waarde ontbreekt’).
 - In geval van een continue variabele, is het aangeraden een nieuwe categorische variabele aan te maken die aangeeft of er wel of niet een waarde aanwezig was voor de continue variabele.
- De eerste stap is na te gaan welke variabelen ontbrekende waarden hebben en hoe vaak deze variabelen ontbrekende waarden hebben. Dit functie summary() is hiervoor zeker nuttig.

```
## luchthaven          maatschappij    vertrek_vertraging
## EWR:119282  United Air Lines Inc. :57491  Min.   : -43.00
## JFK:105230  ExpressJet Airlines Inc.:54173  1st Qu.: -5.00
## LGA:104662  JetBlue Airways       :50940  Median  : -2.00
##                  Delta Air Lines Inc. :46779  Mean    : 12.71
##                  American Airlines Inc. :31327  3rd Qu.: 11.00
##                  Envoy Air           :26397  Max.   :1301.00
##                  (Other)            :62067  NA's    :8214
## aankomst_vertraging afstand      vliegtijd
## Min.   :-86.000   Min.   : 17   Min.   : 20.0
## 1st Qu.:-17.000   1st Qu.: 502  1st Qu.: 81.0
## Median : -5.000   Median : 820  Median :127.0
## Mean   :  6.987   Mean   :1027  Mean   :149.6
## 3rd Qu.: 14.000   3rd Qu.:1372  3rd Qu.:184.0
```

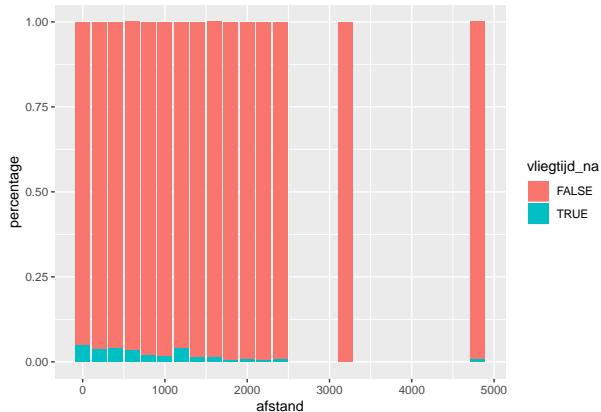
```

##  Max.    :1272.000    Max.    :4983    Max.    :695.0
##  NA's     :9365          NA's     :9368
##          vluchtype
##  kort           :245666
##  normaal        : 31813
##  lang            : 50980
##  intercontinentaal:   715
##
## 
## 
## 
```

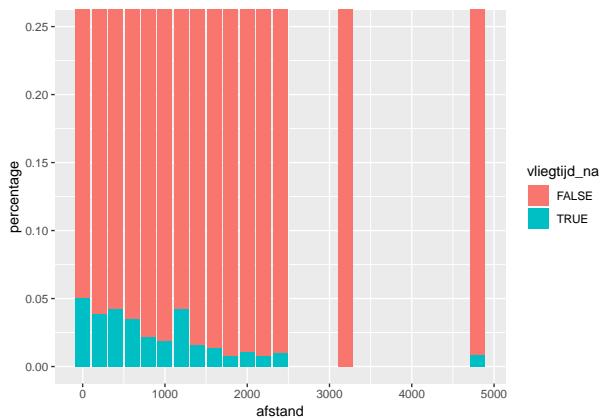
- Uit deze analyse blijkt dat de variabelen vertrek_vertraging, aankomst_vertraging en vliegtijd last hebben van ontbrekende waarden.
- Om vervolgens te analyseren of deze ontbrekende waarden MCAR zijn of niet, zullen we voor ieder van de drie continue variabelen een nieuwe categorische variabele maken die aangeeft of de waarde ontbreekt of niet.
- Nu kunnen we met ggplot achterhalen of de andere variabelen zich ‘anders’ gedragen als er voor één van deze drie variabelen een waarde ontbreekt.
- We illustreren voor ‘afstand’ (continu) en voor ‘maatschappij’ (categorisch).
- Indien we dit bestuderen voor ‘maatschappij’ gaan we voor iedere maatschappij laten zien welk percentage cases een ontbrekende waarde voor vliegtijd heeft. Indien er geen verband is, dan zouden we geen grote verschillen mogen zien tussen de maatschappijen.



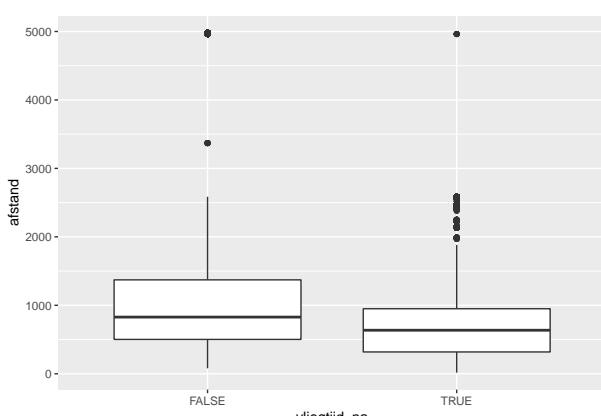
- Uit deze resultaten blijkt dat voor sommige maatschappijen een aanzienlijk hoger percentage ontbrekende waarden bij vliegtijd voorkomt (SkyWest, Mesa en ook ExpressJet, Envoy en Endeavor).
- We kunnen een soortgelijke analyse ook uitvoeren voor de variabele afstand. Hiervoor zullen we eerst de variabele afstand omvormen tot een categorische variabele. We doen dit door de gehele deling uit te voeren (hiervoor gebruik je de %/% operator).



- Deze resultaten lijken te suggereren dat naarmate de vlucht langer wordt, het percentage ontbrekende waarden bij vliegtijd afneemt (met een uitzonderlijke piek bij vluchten rond 1200 mijl).
- Omdat het percentage ontbrekende waarden eerder klein is, is het moeilijk om het patroon duidelijk te zien. We kunnen ook dezelfde plot maken, maar de y-as laten stoppen bij een waarde van 0.25. Op deze manier wordt het patroon duidelijker.



- Tenslotte kunnen we ook nog op een andere manier het verband tussen de afstand en het voorkomen van ontbrekende waarden bij vliegtijd bestuderen, nl. via 2 boxplots.



- Hier zien we dat vluchten waarvoor de vliegtijd ontbreekt vaak kortere vluchten zijn dan waarvoor we de vliegtijd wel hebben. Dit komt overeen met de vorige bevinding.

- Op basis van deze resultaten kunnen we dus stellen dat het ontbreken van de vliegtijd niet willekeurig is, maar vaker voorkomt bij bepaalde maatschappijen en eerder bij kortere dan bij langere vluchten.
- We zouden nog verder kunnen onderzoeken of deze maatschappijen eerder langere of kortere vluchten organiseren.
- Soortgelijke analyses kunnen we ook uitvoeren voor de variabelen vertrek_vertraging en aankomst_vertraging.

5.3.4 Inconsistente waarden

- Data is inconsistent als het niet voldoet aan een aantal regels/beperkingen die horen te gelden op basis van domeinkennis.
- De vorm van inconsistenties waar we ons op focussen, betreft in-record inconsistencies. Dit zijn tegenstrijdigheden die aanwezig zijn binnen één enkele observatie. Enkele voorbeelden zijn:
 - De gemiddelde snelheid van een vlucht ligt hoger dan de maximale theoretische snelheid van het vliegtuig.
 - Het aankomsttijdstip van een vlucht vindt plaats voor het vertrektijdstip.
 - De aankomsttijdstip komt niet overeen met het vertrektijdstip + vertrekvertraging + vluchtduur.
- Het identificeren van inconsistenties kan door middel van diverse dplyr-functies, waarbij je voor iedere observatie test of deze voldoen aan de opgelegde beperkingsregel.
- Daarnaast is er ook het editrules package dat nuttige functies aanbiedt om op een gestructureerdere manier consistentie te evalueren.

5.4 Data opwaarderen

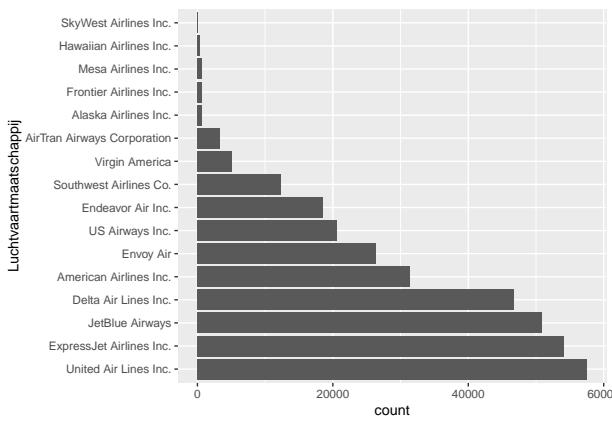
- Van zodra de data geen foutieve en/of ontbrekende waarde meer bevat, kunnen we een aantal technieken toepassen om de data bruikbaarder te maken voor exploratieve analyses. We onderscheiden hierbij 2 technieken:
 - Transformatie van bestaande variabelen.
 - Selectie van observaties.

5.4.0.1 Categorische variabelen

- Soms is het beter om de categorieën van een categorische variabelen te wijzigen door sommige categorieën samen te nemen. Er zijn verschillende situaties waarbij dit het overwegen waard is, zoals:
 - De labels van een categorische variabele is op een te gedetailleerd niveau gedefinieerd, met als gevolg dat de exploratieve analyse al snel complex wordt door de vele categorieën. In zulke gevallen kan het zinvol zijn om het aantal categorieën te verminderen door categorieën die inhoudelijk bij elkaar horen samen te nemen.
 - Een categorische variabele bestaat uit een beperkt aantal categorieën met veel observaties en een groot aantal categorieën met zeer weinig observaties. In zulke gevallen kan het zinvol zijn om de categorieën met weinig observaties samen te nemen in 1 categorie “Overige”.
- Om te bepalen welke categorieën men kan samenvoegen, kan een frequentietabel of barplot gemaakt worden.
- Het herdefiniëren van de labels gebeurt vervolgens met de functie fct_recode (forcats). Hierbij heeft men steeds de keuze om de oorspronkelijke variabele te vervangen of een nieuwe variabele aan te maken.
- Laten we eens aan de hand van een barplot naar de variabele ‘luchtvaartmaatschappij’ kijken. We zien hierbij dat er relatief veel luchtvaartmaatschappijen (categorieën) in onze data zijn en dat er een aantal verwaarloosbaar weinig vluchten bevatten.

Table 5.8: Luchtvaartmaatschappijen geordend volgens stijgend aantal vluchten.

maatschappij	n
SkyWest Airlines Inc.	32
Hawaiian Airlines Inc.	342
Mesa Airlines Inc.	601
Frontier Airlines Inc.	685
Alaska Airlines Inc.	714
AirTran Airways Corporation	3260
Virgin America	5162
Southwest Airlines Co.	12275
Endeavor Air Inc.	18460
US Airways Inc.	20536
Envoy Air	26397
American Airlines Inc.	31327
Delta Air Lines Inc.	46779
JetBlue Airways	50940
ExpressJet Airlines Inc.	54173
United Air Lines Inc.	57491



- We kunnen de exacte aantallen achterhalen met behulp van een frequentietabel.
- Op basis van deze analyse beslissen we om de luchtvaartmaatschappijen met minder dan 10000 vluchten samen te voegen in een nieuwe categorie met het label “Overige”. We opteren ervoor de oorspronkelijke variabelen te vervangen.
- De nieuwe frequentietabel toont het resultaat.

5.4.0.2 Continue variabelen

- Bij continue variabelen zijn er verschillende transformaties die regelmatig uitgevoerd worden:
 - De transformatie van een continue variabele naar een categorische variabele.
 - Het herschalen van de continue variabele.
 - De creatie van een nieuwe variabele op basis van bestaande continue variabelen.
- Ook hier hebben we weer steeds de mogelijkheid om de bestaande variabele te vervangen of een nieuwe variabele aan te maken.

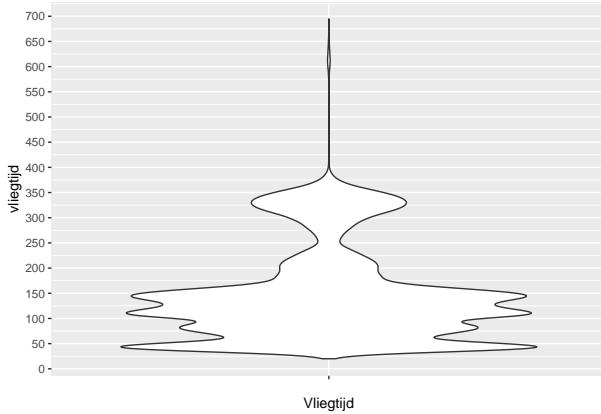
Table 5.9: Luchtvaartmaatschappijen geordend volgens stijgend aantal vluchten.

maatschappij	n
Overige	10796
Southwest Airlines Co.	12275
Endeavor Air Inc.	18460
US Airways Inc.	20536
Envoy Air	26397
American Airlines Inc.	31327
Delta Air Lines Inc.	46779
JetBlue Airways	50940
ExpressJet Airlines Inc.	54173
United Air Lines Inc.	57491

Table 5.10: Frequentietabel vliegtijd (factor)

vliegtijd_fct	n
kort	53220
normaal	210758
lang	55828
NA	9368

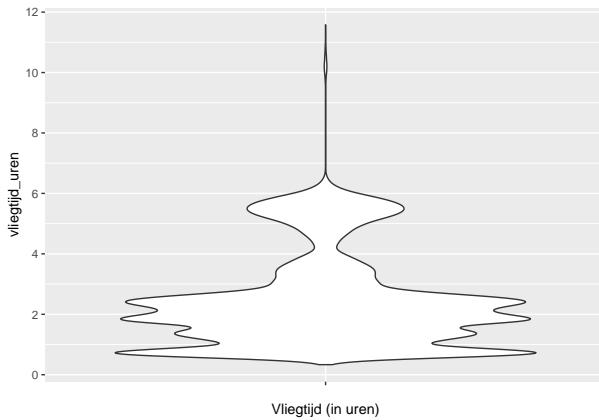
- Laten we de variabele vliegtijd eens onder de loep nemen. We beginnen met een visuele analyse aan de hand van een violinplot.



- Op basis van deze plot beslissen we een nieuwe categorische variabele ‘vliegtijd_fct’ aan te maken, waarbij ‘kort’ overeenkomt met een vlucht die minder dan een uur duurt, ‘normaal’ overeenkomt met een vlucht tussen 1 en 4 uur (60-240) en ‘lang’ overeenkomt met een vlucht van meer dan 4 uur. Hiervoor maken we gebruik van de functie cut.
- Aan de hand van een frequentietabel kunnen we nu het resultaat bekijken.
- Vervolgens beslissen we een nieuwe variabele te maken die de vliegtijd uitdrukt in uren in plaats van minuten.

```
df %>%
  mutate(vliegtijd_uren = vliegtijd/60) -> df
```

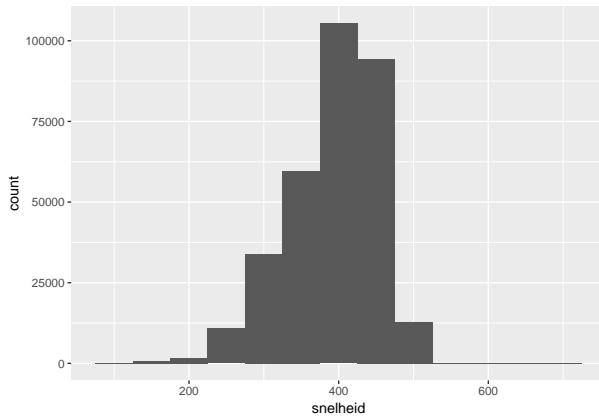
- We kunnen het resultaat bekijken met een violinplot.



- Tenslotte maken we een nieuwe variabele die de gemiddelde snelheid van het vliegtuig uitdrukt door de afstand te delen door de vliegtijd.

```
df %>%
  mutate(snelheid = afstand / vliegtijd_uren) -> df
```

- Laten we het resultaat aan de hand van een histogram bekijken.



5.4.1 Sampling

- Soms is een dataset zo groot, dat analyses veel tijd in beslag nemen. In zulke gevallen kan het nuttig zijn om een random sample te nemen van de oorspronkelijke data om een eerste exploratieve analyse op uit te voeren.
- Zolang de sample willekeurig getrokken wordt en de nieuwe dataset niet te klein wordt, is de kans dat je patronen ontdekt in de sample die niet voorkomen in de volledige dataset eerder klein.
- Na een eerste exploratieve data analyse op de beperkte sample, kan men vervolgens gerichter de volledige dataset analyseren.
- Laten we een sample van 10000 vluchten nemen uit de oorspronkelijke dataset.

```
df.10000 <- df %>% sample_n(10000)
```

- We kunnen nu een eerste blik op deze sample werpen met behulp van de summary functie.

```

## luchthaven          maatschappij  vertrek_vertraging
## EWR:3672  United Air Lines Inc. :1770  Min.   :-22.00
## JFK:3162  ExpressJet Airlines Inc.:1614  1st Qu.: -5.00
## LGA:3166  JetBlue Airways       :1574  Median  : -1.00
##          Delta Air Lines Inc.  :1414  Mean    : 12.23
##          American Airlines Inc. : 930  3rd Qu.: 11.00
##          Envoy Air            : 810  Max.    :436.00
##          (Other)              :1888  NA's    :239
## aankomst_vertraging afstand      vliegtijd
## Min.   :-71.000    Min.   : 80   Min.   : 21.0
## 1st Qu.:-17.000    1st Qu.: 502  1st Qu.: 80.0
## Median : -5.000    Median : 799  Median :126.0
## Mean   :  6.638    Mean   :1023  Mean   :149.1
## 3rd Qu.: 14.000    3rd Qu.:1372  3rd Qu.:184.0
## Max.   :399.000    Max.   :4983  Max.   :671.0
## NA's   :271        NA's   :271  NA's   :271
##          vluchtype     vertrek_vertraging_na aankomst_vertraging_na
## kort           :7481  Mode :logical      Mode :logical
## normaal        : 984  FALSE:9761      FALSE:9729
## lang           :1512  TRUE :239       TRUE :271
## intercontinentaal: 23
##
##
##
## vliegtijd_na  vliegtijd_fct  vliegtijd_uren      snelheid
## Mode :logical  kort   :1680  Min.   : 0.350  Min.   :130.9
## FALSE:9729    normaal:6362  1st Qu.: 1.333  1st Qu.:355.2
## TRUE :271     lang   :1687  Median : 2.100  Median :401.7
##          NA's   : 271   Mean   : 2.485  Mean   :391.2
##          NA's   : 271   3rd Qu.: 3.067  3rd Qu.:435.2
##          NA's   : 271   Max.   :11.183  Max.   :524.7
##          NA's   : 271   NA's   :271   NA's   :271

```

- Als we dit vergelijken met de volledige dataset, dan zien we relatief weinig verschillen wat betreft de centrummaten en de robuste spreidingsmaten.
- Merk op dat minima's en maxima's wel sterk kunnen verschillen. Dit is omdat dit geen robuste maatstaven zijn.

```

## luchthaven          maatschappij  vertrek_vertraging
## EWR:119282  United Air Lines Inc. :57491  Min.   :-43.00
## JFK:105230  ExpressJet Airlines Inc.:54173  1st Qu.: -5.00
## LGA:104662  JetBlue Airways       :50940  Median  : -2.00
##          Delta Air Lines Inc.  :46779  Mean    : 12.71
##          American Airlines Inc. :31327  3rd Qu.: 11.00
##          Envoy Air            :26397  Max.   :1301.00
##          (Other)              :62067  NA's   :8214
## aankomst_vertraging afstand      vliegtijd
## Min.   :-86.000    Min.   : 17   Min.   : 20.0
## 1st Qu.:-17.000    1st Qu.: 502  1st Qu.: 81.0
## Median : -5.000    Median : 820  Median :127.0
## Mean   :  6.987    Mean   :1027  Mean   :149.6
## 3rd Qu.: 14.000    3rd Qu.:1372  3rd Qu.:184.0
## Max.   :1272.000   Max.   :4983  Max.   :695.0
## NA's   :9365        NA's   :9368  NA's   :9368
##          vluchtype     vertrek_vertraging_na aankomst_vertraging_na

```

```

##   kort          :245666  Mode :logical           Mode :logical
##   normaal        : 31813  FALSE:320960        FALSE:319809
##   lang           : 50980  TRUE :8214            TRUE :9365
##   intercontinentaal:    715

##
##
##
##   vliegtijd_na    vliegtijd_fct    vliegtijd_uren      snelheid
##   Mode :logical   kort     : 53220  Min.   : 0.333  Min.   : 76.8
##   FALSE:319806    normaal:210758  1st Qu.: 1.350  1st Qu.:356.3
##   TRUE :9368      lang    : 55828  Median : 2.117  Median :402.6
##                      NA's    : 9368   Mean   : 2.493  Mean   :392.1
##                               3rd Qu.: 3.067  3rd Qu.:436.2
##                               Max.   :11.583  Max.   :703.4
##                               NA's   :9368   NA's   :9368

```

Referenties

1. [Encoding](#)
2. [De geschiedenis van ASCII](#)
3. [Windows code pages](#)
4. [Data importeren in R](#)
5. [Delimited en fixed-width bestanden](#)
6. [XML](#)
7. [JSON Tutorial](#)
8. [Unique-functie](#)
9. [Factors](#)
10. [Fct_relevel](#)
11. [From continuous to categorical](#)

Chapter 6

Exploratieve analyse van tijdgerelateerde data

6.1 Inleiding

6.1.1 Tijdstippen versus periodes

- We kunnen tijdgerelateerde data in twee categorieën onderverdelen: tijdstippen en periodes.
- Tijdstip.
 - Verwijst naar een specifiek moment in de tijd.
 - 3 varianten:
 - * datum (“01-01-2017”) verwijst naar een specifieke dag.
 - * datum-tijdstip (“01-01-2017 13:54”) verwijst naar een specifiek moment op een specifieke dag.
 - * tijdstip (“13:54”) verwijst naar een specifiek moment op een ongedefinieerde dag.
- Periode.
 - Verwijst naar een periode en wordt typisch uitgedrukt aan de hand van de duur van de periode.
 - * Bijvoorbeeld: Een periode van “3605 seconden” of een periode van “2 maanden en 1 dag”.
 - Soms wordt een periode specifiek gedefinieerd aan de hand van twee specifieke tijdstippen die het begin en het einde van de periode aangeven.
 - * Bijvoorbeeld: De periode van 01-01-2017 tot 03-01-2017.
- Bestudeer hoofdstuk 16 van het boek ‘R for Data Science’ van Grolemund en Wickham !

6.2 Tijdstippen

6.2.1 Creatie van tijdstippen

- Omdat R correct met tijdstippen omgaat, is het belangrijk dat tijdstip-variabelen ook correct als tijdstippen herkend worden.
- Er zijn verschillende manieren om tijdstippen te creëren in R.
 - Op basis van het huidige tijdstip.
 - * Dit is mogelijk met de functies today() en now() om respectievelijk een tijdstip van de huidige dag (datum) of het huidige tijdstip (datum_tijd) aan te maken.
 - Op basis van karakterstring. Indien men reeds tijdstippen heeft in de dataset, maar deze zijn gecodeerd als karakterstrings, dan voorziet de lubridate package een aantal handige functies hiervoor:
 - * ymd(), ydm(), mdy(), myd(), dmy(), dym() voor datum-tijdstippen.
 - * ymd_hms(), ymd_hm(), ymd_h(), dmy_hms(), dmy_hm(), dmy_h() voor datumtijd-tijdstippen.
 - * Al deze functies omschrijven de structuur van de karakterstring, waarbij y voor jaar staat, de eerste m voor maand, d voor dag, h voor uur, de tweede m voor minuten en s voor seconden.

- Om de karakterstring “2017-21-02 5:15” correct om te zetten naar een tijdstip, moet je dus de functie `ydm_hm()` gebruiken.
- Op basis van verschillende variabelen die ieder een verschillende component (jaar, maand, dag, uur, minuten, seconden) bevatten.
 - * Hiervoor kan je de functies ‘`make_date()`’ of ‘`make_datetime()`’ gebruiken, afhankelijk of je een datum- of een datumtijd-tijdstip wenst aan te maken.

6.2.2 Extractie van tijdstipinformatie

- Eénmaal variabelen in R als tijdstippen gecodeerd zijn, is het eenvoudig om de verschillende componenten hieruit te extraheren.
- De componenten die je onmiddellijk op het oog kunt herkennen in de oorspronkelijke karakterstring zijn te extraheren met de volgende functies:
 - `year()`.
 - `month()`.
 - `mday()`.
 - `hour()`.
 - `minute()`.
 - `second()`.
- Daarnaast zijn er ook andere componenten die je uit een tijdstip kunt extraheren, dewelke niet rechtstreeks af te lezen zijn uit de oorspronkelijke karakterstring.
 - `week()`: De week van het jaar. “5 feb 2017” is bijvoorbeeld de 6de week van het jaar.
 - `yday()`: De dag in het jaar. “5 feb 2017” is bijvoorbeeld de 36ste dag van het jaar.
 - `wday()`: De dag van de week. “5 feb 2017” is bijvoorbeeld de eerste dag van de week. Let wel op dat we hierbij de conventie hanteren dat een week op zondag start.
 - `wday(..., label=TRUE)`: De naam van de dag van de week. “5 feb 2017” is bijvoorbeeld zondag.

6.2.3 Afronden van tijdstippen

- Ieder tijdstip heeft een zekere nauwkeurigheid. Sommige tijdstippen zijn tot op de seconde gedefinieerd terwijl andere slechts een nauwkeurigheid hebben van weken of maanden.
- Soms kan het voor visualisaties of analyses zinvol zijn om tijdstippen minder nauwkeurig te maken en deze af te ronden. Hiervoor zijn er drie mogelijke functies, afhankelijk van het soort afronding dat men wenst.
 - `floor_date()`: afronden naar onder toe.
 - `round_date()`: normale afrondingsregels.
 - `ceiling_date()`: afronden naar boven toe.
- Deze drie functies hebben 1 belangrijke parameter (`unit`) waarmee je het afrondingsniveau kunt bepalen.

6.2.4 Case: NYC Vluchten 2013

- We zullen de concepten omtrent tijdgerelateerde data illustreren aan de hand van een dataset over de vluchten vanuit NYC in 2013.
- Hieronder vind je een samenvatting van de verschillende variabelen die aanwezig zijn in de dataset.

```
## Observations: 319,809
## Variables: 12
## $ vertrekvluchthaven <chr> "EWR", "LGA", "JFK", "LGA", "EWR", "EWR", ...
## $ aankomstvluchthaven <chr> "George Bush Intercontinental", "George Bu...
## $ maatschappij <chr> "United Air Lines Inc.", "United Air Lines...
## $ jaar_vertrek <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, ...
## $ maand_vertrek <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
```

```
## $ dag_vertrek      <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ uur_vertrek     <dbl> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, ...
## $ minuut_vertrek   <dbl> 17, 33, 42, 54, 54, 55, 57, 57, 58, 58, ...
## $ tijdstip_aankomst <chr> "2013-01-01 08:30:00", "2013-01-01 08:50:0...
## $ vertrek_vertraging <dbl> 2, 4, 2, -6, -4, -5, -3, -3, -2, -2, -2, -...
## $ aankomst_vertraging <dbl> 11, 20, 33, -25, 12, 19, -14, -8, 8, -2, -...
## $ afstand           <dbl> 1400, 1416, 1089, 762, 719, 1065, 229, 944...
```

- Allereerst willen we de variabele tijdstip_aankomst omzetten van een karakterstring naar een datumtijd tijdstip.

```
df %>%
  mutate(tijdstip_aankomst = ymd_hms(tijdstip_aankomst)) -> df
glimpse(df)
```

```
## Observations: 319,809
## Variables: 12
## $ vertrekvluchthaven <chr> "EWR", "LGA", "JFK", "LGA", "EWR", "EWR", ...
## $ aankomstvluchthaven <chr> "George Bush Intercontinental", "George Bu...
## $ maatschappij        <chr> "United Air Lines Inc.", "United Air Lines...
## $ jaar_vertrek       <int> 2013, 2013, 2013, 2013, 2013, 2013, ...
## $ maand_vertrek      <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ dag_vertrek        <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ uur_vertrek        <dbl> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, ...
## $ minuut_vertrek    <dbl> 17, 33, 42, 54, 54, 55, 57, 57, 58, 58, ...
## $ tijdstip_aankomst <dttm> 2013-01-01 08:30:00, 2013-01-01 08:50:00, ...
## $ vertrek_vertraging <dbl> 2, 4, 2, -6, -4, -5, -3, -3, -2, -2, -2, -...
## $ aankomst_vertraging <dbl> 11, 20, 33, -25, 12, 19, -14, -8, 8, -2, -...
## $ afstand            <dbl> 1400, 1416, 1089, 762, 719, 1065, 229, 944...
```

- Verder willen we ook een nieuwe variabele tijdstip_vertrek aanmaken op basis van de variabelen jaar_vertrek, maand_vertrek, dag_vertrek, uur_vertrek en minuut_vertrek.

```
df %>%
  mutate(tijdstip_vertrek = make_datetime(jaar_vertrek,
                                           maand_vertrek,
                                           dag_vertrek,
                                           uur_vertrek,
                                           minuut_vertrek)) %>%
  select(-jaar_vertrek,
         -maand_vertrek,
         -dag_vertrek,
         -uur_vertrek,
         -minuut_vertrek) -> df
glimpse(df)
```

```
## Observations: 319,809
## Variables: 8
## $ vertrekvluchthaven <chr> "EWR", "LGA", "JFK", "LGA", "EWR", "EWR", ...
## $ aankomstvluchthaven <chr> "George Bush Intercontinental", "George Bu...
## $ maatschappij        <chr> "United Air Lines Inc.", "United Air Lines...
## $ tijdstip_aankomst <dttm> 2013-01-01 08:30:00, 2013-01-01 08:50:00, ...
## $ vertrek_vertraging <dbl> 2, 4, 2, -6, -4, -5, -3, -3, -2, -2, -2, -...
## $ aankomst_vertraging <dbl> 11, 20, 33, -25, 12, 19, -14, -8, 8, -2, -...
```

```
## $ afstand          <dbl> 1400, 1416, 1089, 762, 719, 1065, 229, 944...
## $ tijdstip_vertrek <dttm> 2013-01-01 05:17:00, 2013-01-01 05:33:00,...
```

- Vervolgens willen we graag enkele nieuwe variabelen aanmaken die de volgende informatie bevatten: de weekdag van vertrek (maandag, dinsdag, ...), de week van vertrek, de maand van vertrek en de maanddag (1, 2, ..., 31) van vertrek.

```
df %>%
  mutate(weekdag_vertrek = wday(tijdstip_vertrek, label = T),
        week_vertrek = week(tijdstip_vertrek),
        maand_vertrek = month(tijdstip_vertrek),
        maanddag_vertrek = mday(tijdstip_vertrek)) -> df
glimpse(df)
```

```
## Observations: 319,809
## Variables: 12
## $ vertrekvluchthaven <chr> "EWR", "LGA", "JFK", "LGA", "EWR", "EWR", ...
## $ aankomstvluchthaven <chr> "George Bush Intercontinental", "George Bu...
## $ maatschappij         <chr> "United Air Lines Inc.", "United Air Lines...
## $ tijdstip_aankomst    <dttm> 2013-01-01 08:30:00, 2013-01-01 08:50:00,...
## $ vertrek_vertraging   <dbl> 2, 4, 2, -6, -4, -5, -3, -3, -2, -2, -2, -...
## $ aankomst_vertraging  <dbl> 11, 20, 33, -25, 12, 19, -14, -8, 8, -2, -...
## $ afstand              <dbl> 1400, 1416, 1089, 762, 719, 1065, 229, 944...
## $ tijdstip_vertrek     <dttm> 2013-01-01 05:17:00, 2013-01-01 05:33:00,...
## $ weekdag_vertrek      <ord> di, di, di, di, di, di, di, di, di...
## $ week_vertrek         <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maand_vertrek        <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maanddag_vertrek     <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
```

- Tenslotte zullen we een nieuwe variabele maken dewelke het vertrekmoment afrondt tot op de dag nauwkeurig (dus zonder het specifieke uur).

```
df %>%
  mutate(dag_vertrek = floor_date(tijdstip_vertrek, "day")) -> df
glimpse(df)
```

```
## Observations: 319,809
## Variables: 13
## $ vertrekvluchthaven <chr> "EWR", "LGA", "JFK", "LGA", "EWR", "EWR", ...
## $ aankomstvluchthaven <chr> "George Bush Intercontinental", "George Bu...
## $ maatschappij         <chr> "United Air Lines Inc.", "United Air Lines...
## $ tijdstip_aankomst    <dttm> 2013-01-01 08:30:00, 2013-01-01 08:50:00,...
## $ vertrek_vertraging   <dbl> 2, 4, 2, -6, -4, -5, -3, -3, -2, -2, -2, -...
## $ aankomst_vertraging  <dbl> 11, 20, 33, -25, 12, 19, -14, -8, 8, -2, -...
## $ afstand              <dbl> 1400, 1416, 1089, 762, 719, 1065, 229, 944...
## $ tijdstip_vertrek     <dttm> 2013-01-01 05:17:00, 2013-01-01 05:33:00,...
## $ weekdag_vertrek      <ord> di, di, di, di, di, di, di, di, di...
## $ week_vertrek         <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maand_vertrek        <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maanddag_vertrek     <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ dag_vertrek          <dttm> 2013-01-01, 2013-01-01, 2013-01-01, 2013-...
```

6.3 Periode-data

- We kunnen 3 soorten van periodes onderscheiden, waarbij het eerste type (interval) naar een specifieke periode tussen 2 tijdstippen verwijst en de 2 andere types (duration en period) naar een periode van een specifieke duur verwijzen maar telkens onafhankelijk van het specifieke tijdstip.
- Om de verschillen duidelijk te illustreren werken we met twee specifieke tijdstippen “1 jan 2016” en “1 jan 2017”.

```
t1 <- ymd(160101)
t1
## [1] "2016-01-01"

t2 <- ymd(170101)
t2
## [1] "2017-01-01"
```

6.3.1 Interval

- Een interval is een periode die bepaald wordt door twee specifieke tijdstippen.
- Een interval creëer je met behulp van de speciale operator `%--%`.
- Intervals worden weinig gebruikt om rechtstreeks te analyseren, maar kunnen als tussenstap gebruikt worden om de duurtijd van specifieke periodes te bepalen.

```
interval_t2t1 <- t1 %--% t2
interval_t2t1
## [1] 2016-01-01 UTC--2017-01-01 UTC
```

6.3.2 Duration

- Duration is de duur van een periode uitgedrukt als het exact aantal seonden die feitelijk verstrekken zijn tussen twee tijdstippen.
- Tussen ‘26 maart 2017 02:00:00’ en ‘26 maart 2017 03:00:01’ is slechts 1 seconde feitelijk verstrekken omdat we van 2u naar 3u zijn overgeschakeld op het zomeruur.
- Durations gebruik je voornamelijk als je de werkelijke tijd tussen twee tijdstippen wenst te berekenen of wanneer je een aantal seonden wenst toe te voegen bij of af te trekken van een specifiek tijdstip.
- Om een duration van een specifieke duur te creëren gebruik je volgende functies:
 - `dseconds()`.
 - `dminutes()`.
 - `dhours()`.
 - `ddays()`.
 - `dweeks()`.
 - `dyears()`.
- Om de duration van een interval te bepalen, gebruik je de functie:
 - `as.duration()`.

```
t1 + dyears(1)
## [1] "2016-12-31"
```

```
interval_t2t1 / dyears(1)

## [1] 1.00274

as.duration(interval_t2t1)

## [1] "31622400s (~1 years)"
```

6.3.3 Period

- De tijd die verstrekken ‘lijkt’ te zijn (op een klok) tussen twee tijdstippen.
- Dus tussen ‘26 maart 2017 02:00:00’ en ‘26 maart 2017 03:00:01’ zit een period van 1 uur en 1 seconde.
- Periods gebruik je voornamelijk als je periodes wilt toevoegen aan tijdstippen zonder rekening te moeten houden met onverwachte sprongen in de tijd (zomertijd/wintertijd, schrikkeljaren, ...).
 - Dus als je bij ieder tijdstip 1 dag (24u) wenst toe te voegen, kan je beter een period gebruiken dan een duration, omdat je anders rekening moet houden met de dag waarop we van zomer- naar winteruur gaan en omgekeerd.
- Belangrijke functies die periods aanmaken zijn:
 - seconds().
 - minutes().
 - hours().
 - days().
 - months().
 - weeks().
 - years().

```
t1 + years(1)

## [1] "2017-01-01"

interval_t2t1 / years(1)

## [1] 1

as.period(interval_t2t1)

## [1] "1y 0m 0d 0H 0M 0S"
```

6.3.4 Case: NYC vluchten 2013

- Momenteel bevat onze dataset enkel het geplande vertrek- en aankomstmoment. We gaan nu aan de hand van de informatie over de vertrek- en aankomstvertraging de werkelijke vertrek- en aankomstmomenten bepalen.

```
df %>%
  mutate(vertrek_werkelijk = tijdstip_vertrek + dminutes(vertrek_vertraging),
         aankomst_werkelijk = tijdstip_aankomst + dminutes(aankomst_vertraging)) %>%
  rename(aankomst_gepland = tijdstip_aankomst,
         vertrek_gepland = tijdstip_vertrek) %>%
  select(vertrekluchthaven, aankomstluchthaven, maatschappij, vertrek_gepland,
```

```

vertrek_werkelijk, vertrek_vertraging, aankomst_gepland,
aankomst_werkelijk, aankomst_vertraging, afstand, weekdag_vertrek,
week_vertrek, maand_vertrek, dag_vertrek, maanddag_vertrek) -> df
glimpse(df)

## Observations: 319,809
## Variables: 15
## $ vertrekvluchthaven <chr> "EWR", "LGA", "JFK", "LGA", "EWR", ...
## $ aankomstvluchthaven <chr> "George Bush Intercontinental", "George Bu...
## $ maatschappij <chr> "United Air Lines Inc.", "United Air Lines...
## $ vertrek_gepland <dttm> 2013-01-01 05:17:00, 2013-01-01 05:33:00, ...
## $ vertrek_werkelijk <dttm> 2013-01-01 05:19:00, 2013-01-01 05:37:00, ...
## $ vertrek_vertraging <dbl> 2, 4, 2, -6, -4, -5, -3, -3, -2, -2, -2, -...
## $ aankomst_gepland <dttm> 2013-01-01 08:30:00, 2013-01-01 08:50:00, ...
## $ aankomst_werkelijk <dttm> 2013-01-01 08:41:00, 2013-01-01 09:10:00, ...
## $ aankomst_vertraging <dbl> 11, 20, 33, -25, 12, 19, -14, -8, 8, -2, -...
## $ afstand <dbl> 1400, 1416, 1089, 762, 719, 1065, 229, 944...
## $ weekdag_vertrek <ord> di, di, di, di, di, di, di, di, di...
## $ week_vertrek <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maand_vertrek <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ dag_vertrek <dttm> 2013-01-01, 2013-01-01, 2013-01-01, 2013-...
## $ maanddag_vertrek <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...

```

6.4 Analyseren van tijdgerelateerde data

6.4.1 Case: NYC Vluchten 2013

- Een eerste stap om inzicht te krijgen in de tijdgerelateerde data is met behulp van de summary() functie. Het is vooral nuttig om naar de minima en maxima te kijken. Dit geeft vaak aan of de tijdsperiode waarvoor de data verzameld is overeenkomt met de verwachte periode. In onderstaand geval blijkt dit in orde te zijn.

```

summary(df)

## vertrekvluchthaven aankomstvluchthaven maatschappij
## Length:319809      Length:319809      Length:319809
## Class :character   Class :character   Class :character
## Mode  :character   Mode  :character   Mode  :character
##
## 
## 
## 
## vertrek_gepland           vertrek_werkelijk
## Min.   :2013-01-01 05:17:00  Min.   :2013-01-01 05:19:00
## 1st Qu.:2013-04-05 09:07:00  1st Qu.:2013-04-05 09:10:00
## Median :2013-07-04 13:43:00  Median :2013-07-04 13:47:00
## Mean   :2013-07-03 21:27:29  Mean   :2013-07-03 21:40:07
## 3rd Qu.:2013-10-01 20:37:00  3rd Qu.:2013-10-01 20:39:00
## Max.   :2013-12-31 23:32:00  Max.   :2014-01-01 00:19:00
##
## vertrek_vertraging aankomst_gepland
## Min.   : -43.00    Min.   :2013-01-01 07:02:00
## 1st Qu.: -5.00     1st Qu.:2013-04-05 11:22:00

```

```

## Median : -2.00    Median :2013-07-04 15:42:00
## Mean   : 12.62    Mean   :2013-07-03 23:42:08
## 3rd Qu.: 11.00    3rd Qu.:2013-10-01 22:27:00
## Max.   :1301.00    Max.   :2014-01-01 01:10:00
##
## aankomst_werkelijk      aankomst_vertraging      afstand
## Min.   :2013-01-01 06:55:00  Min.   :-86.000    Min.   : 80
## 1st Qu.:2013-04-05 11:21:00  1st Qu.:-17.000    1st Qu.: 502
## Median :2013-07-04 15:36:00  Median :-5.000    Median : 828
## Mean   :2013-07-03 23:49:08  Mean   : 6.987    Mean   :1035
## 3rd Qu.:2013-10-01 22:12:00  3rd Qu.: 14.000    3rd Qu.:1372
## Max.   :2014-01-01 01:53:00  Max.   :1272.000   Max.   :4983
##
## weekdag_vertrek  week_vertrek  maand_vertrek
## zo:44396        Min.   : 1.00  Min.   : 1.000
## ma:48246        1st Qu.:14.00  1st Qu.: 4.000
## di:48084        Median :27.00  Median : 7.000
## wo:47597        Mean   :26.77  Mean   : 6.569
## do:47378        3rd Qu.:40.00  3rd Qu.:10.000
## vr:47455        Max.   :53.00  Max.   :12.000
## za:36653
##
## dag_vertrek      maanddag_vertrek
## Min.   :2013-01-01 00:00:00  Min.   : 1.00
## 1st Qu.:2013-04-05 00:00:00  1st Qu.: 8.00
## Median :2013-07-04 00:00:00  Median :16.00
## Mean   :2013-07-03 07:43:47  Mean   :15.74
## 3rd Qu.:2013-10-01 00:00:00  3rd Qu.:23.00
## Max.   :2013-12-31 00:00:00  Max.   :31.00
##

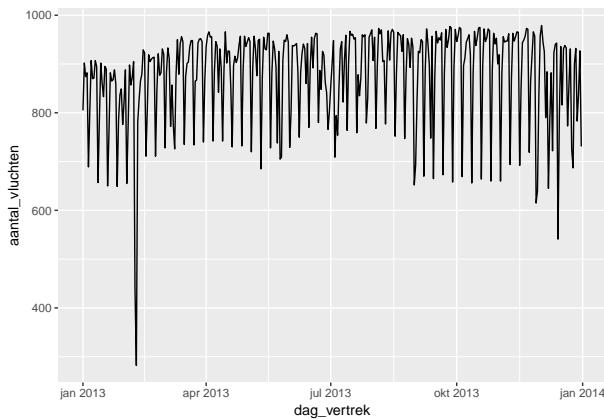
```

6.4.1.1 Analyse visuele tijdreekspatronen

- Eén van de meest voorkomende exploratieve visuele analysetechnieken voor tijdgerelateerde data is het zoeken naar patronen hoe een variabele doorheen de tijd verandert.
- De eerste stap is hierbij telkens de tijdreekspatronen te visualiseren. Om dit te doen kan je volgend stappenplan toepassen.
 - Bepaal over welke tijdsdimensie je patronen wenst te bestuderen. Dit is je **X**-variabele. De **X**-variabele bepaalt de granulariteit van je visualisatie. Wens je op niveau van dagen te visualiseren, dan is je tijdsdimensie ‘dag’, en dan ga je gedetailleerder naar de patronen kijken, dan wanneer je op niveau van bijvoorbeeld ‘maand’ naar de data kijkt.
 - Bepaal welke variabele je doorheen de tijd wenst te bestuderen. Dit is je **Y**-variabele.
 - Je gaat voor iedere **X** waarde 1 **Y** waarde moeten hebben. Vaak betekent dit dat je deze **Y**-variabele nog moet aanmaken. Mogelijke **Y** variabelen zijn het aantal observaties per tijdsseenheid of de centrummaat (bv. mediaan) van een specifieke variabele.
 - Je R-code vertrekt steeds van de oorspronkelijke dataset, groepeert vervolgens op de tijdsdimensie, berekent de gewenste samenvattende statistiek (`summarise()`) en visualiseert vervolgens via `ggplot() + geom_line()`.
- We willen bijvoorbeeld de evolutie zien van het aantal vluchten per dag. De tijdsdimensie is dus `dag_vertrek` en de **Y**-variabele wordt gemaakt door het aantal rijen per dag te tellen.
- De analyse van onderstaande grafiek toont een aantal opvallende zaken:
 - Er is een zware en niet-wederkerende daling tussen januari en april. Hier moet iets uitzonderlijks gebeurd zijn.
 - We zien een terugkerend patroon, waarbij om de aantal dagen een daling is in het aantal vluchten.

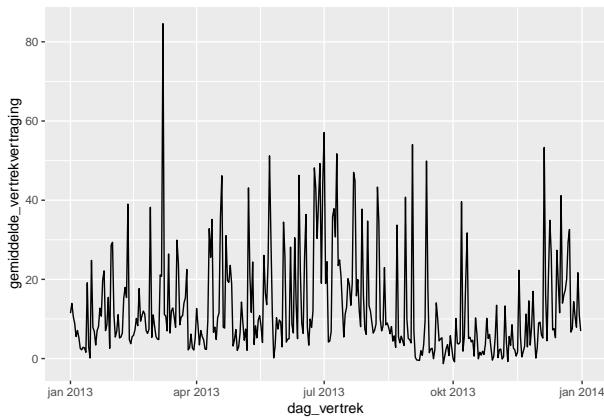
- De schommelingen en met name de daling op het einde van ieder terugkerend patroon wordt groter op het einde van het jaar.

```
df %>%
  group_by(dag_vertrek) %>%
  summarise(aantal_vluchten = n()) %>%
  ggplot(aes(x=dag_vertrek, y=aaltonal_vluchten)) +
  geom_line()
```



- We kunnen een soortgelijke analyse doen voor de gemiddelde vertrekvertraging.

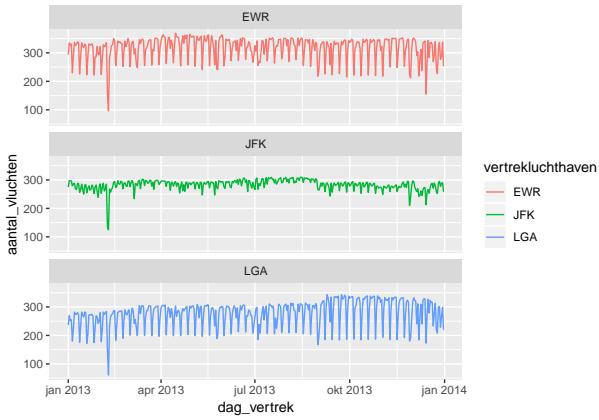
```
df %>%
  group_by(dag_vertrek) %>%
  summarise(gemiddelde_vertrekvertraging = mean(vertrek_vertraging)) %>%
  ggplot(aes(x=dag_vertrek, y=gemiddelde_vertrekvertraging)) +
  geom_line()
```



- Een volgende stap is vaak om de tijdreeks patronen apart te visualiseren voor de verschillende waarden van een categorische variabele.
- Dit kan op een eenvoudige wijze door in onze R-code deze categorische variabele op te nemen in het group_by() gedeelte en vervolgens aparte plots te creëren met behulp van facet_wrap().
- Laten we de evolutie van het aantal vluchten per dag bijvoorbeeld uitsplitsen per luchthaven.
- Uit onderstaande analyse blijkt dan dat het aantal vluchten vanuit JFK veel minder sterk schommelt dan EWR en LGA. Wel valt op dat alle drie de luchthavens een sterke uitzonderlijke daling kenden in de eerste helft van het jaar.

```
df %>%
```

```
group_by(dag_vertrek, vertrekvluchthaven) %>%
summarise(aantal_vluchten = n()) %>%
ggplot(aes(x = dag_vertrek, y = aantal_vluchten, colour=vertrekvluchthaven)) +
geom_line() + facet_wrap(~vertrekvluchthaven, ncol = 1)
```



6.4.1.2 Identificeren van opmerkelijke gebeurtenissen in een tijdreeks

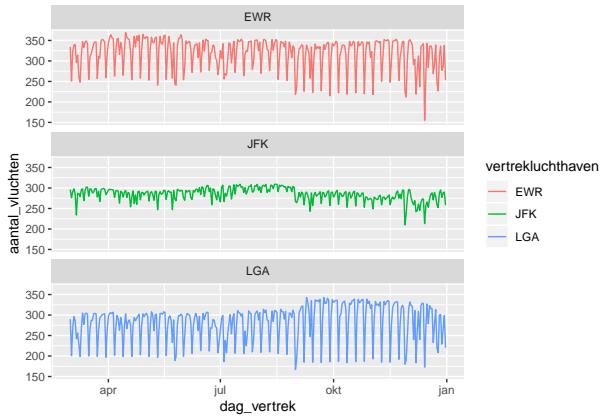
- In de evolutie van het aantal vluchten valt op dat er een uitzonderlijke daling plaatsvond in de periode tussen januari en april.
- In zulke gevallen is het best te achterhalen wat hier precies de oorzaak is.
- De eerste stap is dan ook het exacte tijdstip te identificeren.
- We kunnen dit doen door de data te filteren op die dagen dat er zeer weinig vluchten zijn.

```
df %>%
```

```
group_by(dag_vertrek, vertrekvluchthaven) %>%
summarise(aantal_vluchten = n()) %>%
filter(aantal_vluchten < 160)
```

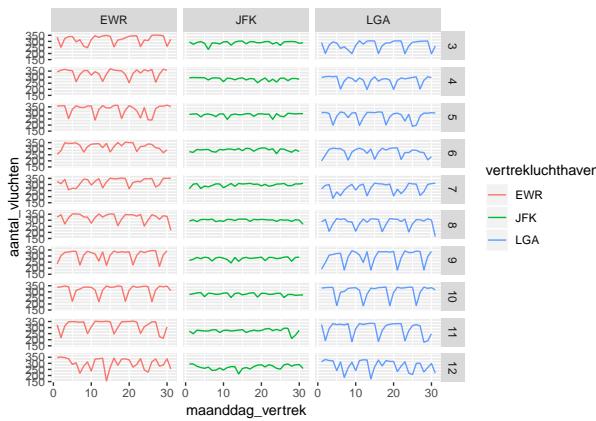
```
## # A tibble: 7 x 3
## # Groups:   dag_vertrek [3]
##   dag_vertrek     vertrekvluchthaven aantal_vluchten
##   <dttm>           <chr>                <int>
## 1 2013-02-08 00:00:00 EWR                  159
## 2 2013-02-08 00:00:00 JFK                  133
## 3 2013-02-08 00:00:00 LGA                  148
## 4 2013-02-09 00:00:00 EWR                  96
## 5 2013-02-09 00:00:00 JFK                 125
## 6 2013-02-09 00:00:00 LGA                  61
## 7 2013-12-14 00:00:00 EWR                  155
```

- Uit deze analyse blijkt dat de daling plaatsvond op 8 en 9 februari 2013. Na enig opzoekwerk blijkt dat New York toen geteisterd werd door een hevige sneeuwstorm waardoor zeer veel vluchten geannuleerd moesten worden.
- Omdat dit moment niet representatief is voor een normaal jaar, beslissen we om enkel met de tijdgere-lateerde data van maart tot en met december verder te gaan.
- We kunnen de tijdreeks van de nieuwe periode opnieuw visualiseren.



- We kunnen verder inzoomen in de data door naar de tijdreeks patronen te kijken per maand en per luchthaven.

```
df_mardec %>%
  group_by(maanddag_vertrek, maand_vertrek, vertrekvluchthaven) %>%
  summarise(aantal_vluchten = n()) %>%
  ggplot(aes(x=maanddag_vertrek, y=aantal_vluchten, colour=vertrekvluchthaven)) +
  geom_line() + facet_grid(maand_vertrek ~ vertrekvluchthaven)
```



6.5 Referenties

1. ‘R for Data Science’ van Grolemund en Wickham

Chapter 7

Tidy Data

7.1 Inleiding

- In werkelijkheid komt data niet altijd in het geschikte formaat om de gewenste analyses op uit te voeren.
 - Vaak is data verspreid over meerdere datasets en moeten we hier 1 dataframe van maken voor onze analyses.
 - Soms stelt een rij niet de observatie voor die willen bestuderen (bv: één rij stelt de gegevens van één auto betrokken in een ongeval voor, terwijl we willen dat iedere rij een ongeval voorstelt met de gegevens van alle betrokken voertuigen).
- Het manipuleren van de data opdat het in het juiste formaat staat, wordt ook wel de creatie van ‘tidy data’ genoemd.
- Bestudeer secties 12.1 tot en met 12.4 en 12.6 in ‘R for Data Science’ van Grolemund en Wickham!
- Bestudeer hoofdstuk 13 in ‘R for Data Science’ van Grolemund en Wickham!

7.2 Case: NYC Vluchten 2013

7.2.1 Datasets samenvoegen

- We vertrekken van een dataset met vluchten opgestegen vanuit NYC in 2013. Hieronder een overzicht van de variabelen in de dataset.

```
## Observations: 319,809
## Variables: 13
## $ id                  <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, ...
## $ vertrekvluchthaven   <chr> "EWR", "LGA", "JFK", "LGA", "EWR", "EWR", ...
## $ aankomstvluchthaven  <chr> "George Bush Intercontinental", "George Bu...
## $ maatschappij          <chr> "United Air Lines Inc.", "United Air Lines...
## $ tijdstip_aankomst    <dttm> 2013-01-01 08:30:00, 2013-01-01 08:50:00, ...
## $ vertrek_vertraging   <dbl> 2, 4, 2, -6, -4, -5, -3, -3, -2, -2, -2, -...
## $ aankomst_vertraging  <dbl> 11, 20, 33, -25, 12, 19, -14, -8, 8, -2, -...
## $ afstand               <dbl> 1400, 1416, 1089, 762, 719, 1065, 229, 944...
## $ tijdstip_vertrek     <dttm> 2013-01-01 05:17:00, 2013-01-01 05:33:00, ...
## $ weekdag_vertrek      <ord> di, di, di, di, di, di, di, di, di...
## $ week_vertrek          <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maand_vertrek         <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maanddag_vertrek     <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
```

- We beschikken nu ook over een tweede dataset met de gegevens van de luchthavens. Hieronder een overzicht van de variabelen in deze dataset.

```
## Observations: 1,458
## Variables: 8
## $ faa    <chr> "04G", "06A", "06C", "06N", "09J", "0A9", "0G6", "0G7", ...
## $ name   <chr> "Lansdowne Airport", "Moton Field Municipal Airport", "S...
## $ lat    <dbl> 41.13047, 32.46057, 41.98934, 41.43191, 31.07447, 36.371...
## $ lon    <dbl> -80.61958, -85.68003, -88.10124, -74.39156, -81.42778, -...
## $ alt    <int> 1044, 264, 801, 523, 11, 1593, 730, 492, 1000, 108, 409, ...
## $ tz     <dbl> -5, -6, -6, -5, -5, -5, -5, -5, -8, -5, -6, -5, -5, ...
## $ dst    <chr> "A", "A", "A", "A", "A", "A", "U", "A", "A", "...
## $ tzone <chr> "America/New_York", "America/Chicago", "America/Chicago"...
```

- Als we deze datasets vergelijken zien we een mogelijke relatie tussen beiden.
 - In de oorspronkelijke dataset stelt iedere rij een vlucht voor en wordt de vertrekvluchthaven voorgesteld door een 3-letterige code.
 - In de airports-dataset stelt iedere rij een luchthaven voor en vinden we een 3-letterige code terug in de kolom ‘faa’.
- We willen nu graag deze twee datasets aan elkaar koppelen door de gegevens van de vertrekvluchthavens uit de airports-dataset te halen en toe te voegen aan iedere vlucht.
- Alvorens we dit kunnen doen, moeten we eerst controleren of de faa-code in de airports-dataset uniek is.
 - Dit is een essentiële vereiste om de gegevens van de airports-dataset te kunnen toevoegen aan de oorspronkelijke dataset.
 - Indien er bijvoorbeeld 2 luchthavens in de airports-dataset zouden zitten met faa-code ‘EWR’, dan zou R niet kunnen achterhalen van welke luchthaven de gegevens moeten worden toegevoegd aan de vluchten met als vertrekvluchthaven ‘EWR’.
 - In zulke gevallen gaat R de vlucht dupliveren en iedere kopie (van de vlucht) koppelen aan een andere luchthaven uit de airports-dataset met faa-code EWR.

```
## # A tibble: 0 x 2
## # ... with 2 variables: faa <chr>, n <int>

• Uit bovenstaande analyse blijkt dat er geen twee rijen zijn in de airports-dataset met dezelfde faa-code.
• We kunnen nu de gegevens van de airports-dataset toevoegen aan het oorspronkelijk dataframe. We doen dit met behulp van een left_join() en geven aan via welke variabelen de link gelegd moet worden.
```

```
## Observations: 319,809
## Variables: 20
## $ id                  <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, ...
## $ vertrekvluchthaven <chr> "EWR", "LGA", "JFK", "LGA", "EWR", "EWR", ...
## $ aankomstvluchthaven <chr> "George Bush Intercontinental", "George Bu...
## $ maatschappij        <chr> "United Air Lines Inc.", "United Air Lines...
## $ tijdstip_aankomst   <dttm> 2013-01-01 08:30:00, 2013-01-01 08:50:00, ...
## $ vertrek_vertraging  <dbl> 2, 4, 2, -6, -4, -5, -3, -3, -2, -2, -2, -...
## $ aankomst_vertraging <dbl> 11, 20, 33, -25, 12, 19, -14, -8, 8, -2, -...
## $ afstand              <dbl> 1400, 1416, 1089, 762, 719, 1065, 229, 944...
## $ tijdstip_vertrek    <dttm> 2013-01-01 05:17:00, 2013-01-01 05:33:00, ...
## $ weekdag_vertrek    <ord> di, di, di, di, di, di, di, di, di...
## $ week_vertrek        <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maand_vertrek       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maanddag_vertrek   <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ name                <chr> "Newark Liberty Intl", "La Guardia", "John...
## $ lat                 <dbl> 40.69250, 40.77725, 40.63975, 40.77725, 40...
## $ lon                 <dbl> -74.16867, -73.87261, -73.77893, -73.87261...
```

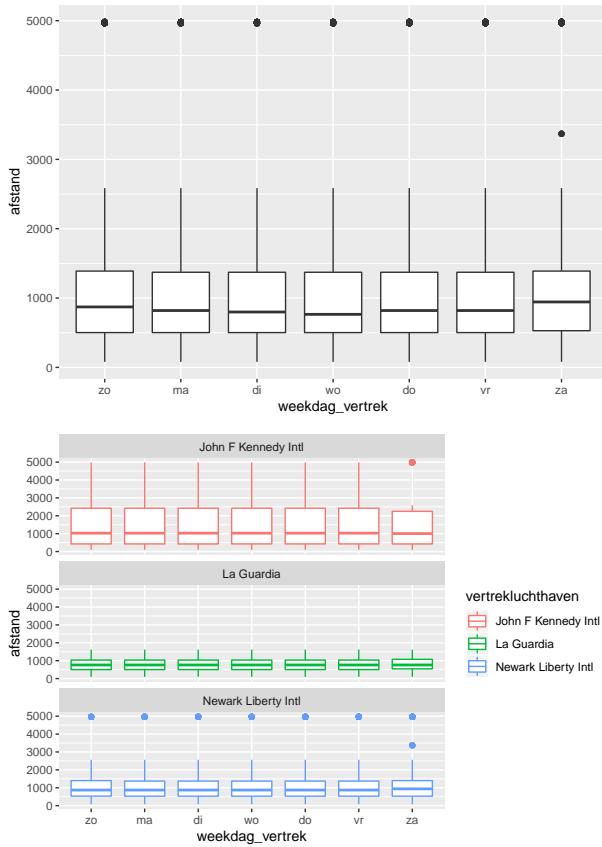
```
## $ alt <int> 18, 22, 13, 22, 18, 18, 22, 13, 22, 13, 13...
## $ tz <dbl> -5, -5, -5, -5, -5, -5, -5, -5, -5, -5...
## $ dst <chr> "A", "A", "A", "A", "A", "A", "A", "A"...
## $ tzone <chr> "America/New_York", "America/New_York", "A..."
```

- Bovenstaande output laat zien dat 7 kolommen zijn toegevoegd aan de oorspronkelijke dataset.
- Merk op dat de faa-kolom van het airports-dataframe niet is toegevoegd. Dit is niet nodig aangezien we in de join-functie hadden aangegeven dat deze kolom overeenkwam met de kolom vertrekvluchthaven uit de oorspronkelijke dataset.
- Controleer ook altijd of het aantal observaties niet gewijzigd is, daar dit vaak wijst op een fout in de join. In dit geval is het aantal observaties niet veranderd.
- In een volgende stap verwijderen we een aantal kolommen die we verder niet nodig gaan hebben en veranderen we de kolom ‘name’ in ‘vertrekvluchthaven’. Zoals je in het resultaat kan zien bevat onze nieuwe dataset nu de volledige naam van de vertrekvluchthaven en niet enkel de faa-code.

```
## Observations: 319,809
## Variables: 13
## $ id <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, ...
## $ aankomstvluchthaven <chr> "George Bush Intercontinental", "George Bu...
## $ maatschappij <chr> "United Air Lines Inc.", "United Air Lines...
## $ tijdstip_aankomst <dttm> 2013-01-01 08:30:00, 2013-01-01 08:50:00, ...
## $ vertrek_vertraging <dbl> 2, 4, 2, -6, -4, -5, -3, -3, -2, -2, -2, -...
## $ aankomst_vertraging <dbl> 11, 20, 33, -25, 12, 19, -14, -8, 8, -2, -...
## $ afstand <dbl> 1400, 1416, 1089, 762, 719, 1065, 229, 944...
## $ tijdstip_vertrek <dttm> 2013-01-01 05:17:00, 2013-01-01 05:33:00, ...
## $ weekdag_vertrek <ord> di, di, di, di, di, di, di, di, di, di...
## $ week_vertrek <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maand_vertrek <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ maanddag_vertrek <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ vertrekvluchthaven <chr> "Newark Liberty Intl", "La Guardia", "John..."
```

7.3 Data in een lang formaat plaatsen (voor visuele analyses)

- Bij een bivariate visualisatie heb je steeds het basisprincipe dat je de relatie tussen twee variabelen wenst weer te geven.
- Bij een multivariate visualisatie ga je vaak weergeven hoe deze relatie verandert in functie van een derde variabele.
- Deze derde variabele is vaak categorisch en de verschillende categorieën stellen hierbij groeperingen van de observaties voor waarvoor je de relatie tussen X en Y wenst weer te geven.
 - Je wil bijvoorbeeld initieel de relatie tussen weekdag en afstand van de vluchten weergeven. Hiervoor kan je een bivariate plot maken waarbij X categorisch is en Y continu. Een mogelijkheid hiervoor is een boxplot.
 - In een volgende stap kan je de relatie tussen afstand en weekdag opsplitsen per luchthaven. Je wil dus weten hoe deze relatie verschilt tussen diverse luchthavens. Hiervoor gebruik je de categorische variabele ‘vertrekvluchthaven’ en kan je bijvoorbeeld de kleur van de boxplot koppelen aan de vertrekvluchthaven of aparte ‘facets’ maken voor iedere luchthaven.
 - Hieronder zie je de bijhorende plots.



- Stel nu dat je het effect wenst te weten van de weekdag van vertrek op de vertraging van een vlucht, maar je wil hierbij onderscheid maken tussen vertrek- en aankomstvertraging.
- Volgens bovenstaande aanpak zou je dan een Y-variabele moeten hebben die de vertraging meet en een Z-variabele die het type van vertraging aangeeft (aankomst of vertrek).
- Onze dataset is echter anders opgebouwd. In de beschikbare data is de vertraging van een vlucht opgeslagen met behulp van twee aparte variabelen, namelijk vertrek- en aankomstvertraging. Dit blijkt uit onderstaande tabel.

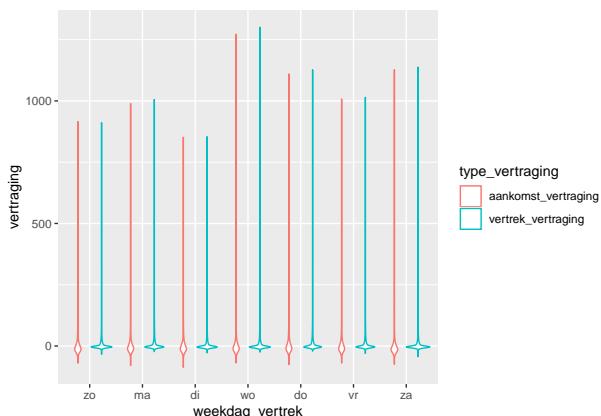
```
## # A tibble: 319,809 x 5
##       id vertrekluchthav~ vertrek_vertrag~ aankomst_vertra~ weekdag_vertrek
##   <int> <chr>           <dbl>           <dbl> <ord>
## 1     1 Newark Liberty ~      2            11 di
## 2     2 La Guardia          4            20 di
## 3     3 John F Kennedy ~    2            33 di
## 4     4 La Guardia         -6           -25 di
## 5     5 Newark Liberty ~   -4            12 di
## 6     6 Newark Liberty ~   -5            19 di
## 7     7 La Guardia         -3           -14 di
## 8     8 John F Kennedy ~   -3            -8 di
## 9     9 La Guardia         -2             8 di
## 10   10 John F Kennedy ~  -2            -2 di
## # ... with 319,799 more rows
```

- We moeten de data dus omzetten zodat het type vertraging niet gecodeerd wordt als aparte variabelen, maar door middel van 1 categorische variabele.

- Hiervoor kunnen we de gather() functie hanteren. Deze functie zal een set van variabelen (in dit geval ‘vertrek_vertraging’ en ‘aankomst_vertraging’) transformeren naar 2 variabelen, namelijk een key-variabele en een value-variabele.
 - De key-variabele is een categorische variabele en de categorieën komen overeen met de variablenamen in onze set van variabelen die we wensen te transformeren. In ons geval zijn dit dus de categorieën ‘vertrek_vertraging’ en ‘aankomst_vertraging’.
 - De value-variabele bevat de bijhorende waarde uit de oorspronkelijke dataset.
- De gather() functie bestaat uit 3 delen.
 - Eerst vermeld je alle variabelen die je wenst te vervangen.
 - Vervolgens geef je de naam van de nieuwe key-variabele.
 - Tenslotte geef je de naam van de nieuwe value-variabele.

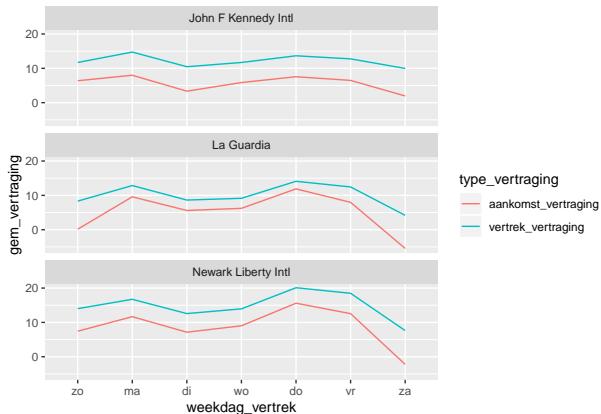
```
## # A tibble: 639,618 x 5
##   id vertrekhaven weekdag_vertrek type_vertraging vertraging
##   <int> <chr>        <ord>       <chr>           <dbl>
## 1 1 Newark Liberty Intl di    vertrek_vertraging 2
## 2 1 Newark Liberty Intl di    aankomst_vertraging 11
## 3 2 La Guardia          di    vertrek_vertraging 4
## 4 2 La Guardia          di    aankomst_vertraging 20
## 5 3 John F Kennedy Intl di    vertrek_vertraging 2
## 6 3 John F Kennedy Intl di    aankomst_vertraging 33
## 7 4 La Guardia          di    vertrek_vertraging -6
## 8 4 La Guardia          di    aankomst_vertraging -25
## 9 5 Newark Liberty Intl di    vertrek_vertraging -4
## 10 5 Newark Liberty Intl di   aankomst_vertraging 12
## # ... with 639,608 more rows
```

- Merk op dat het aantal rijen nu verdubbeld is. Dit komt omdat je nu voor zowel vertrek- als aankomstvertraging een aparte rij hebt gecreëerd.
 - Hierdoor krijg je een andere definitie van de observatie die in een rij staat. In de oorspronkelijke dataset was iedere rij (observatie) een vlucht vanuit NYC in 2013. In de nieuwe dataset stelt iedere rij het vertrek of de aankomst van een vlucht vanuit NYC in 2013 voor!
- Indien je dus de gather() functie hanteert gaat het aantal rijen toenemen. Het aantal kolommen zal afnemen indien de variabelenset, die je wenst te transformeren, uit meer dan 2 variabelen bestaat.
- Hierdoor krijg je een dataset die minder breed is en vooral langer. Daarom wordt dit het lange formaat genoemd.
- Data in een lang formaat zijn voornamelijk nuttig om visualisaties te realiseren met ggplot.
- Met dit lange formaat kunnen we de relatie tussen weekdag van vertrek en de vertraging, uitgesplitst volgens vertrek- of aankomstvertraging, visualiseren.



- Indien we de relatie tussen de weekdag en de gemiddelde vertraging, uitgesplitst volgens vertragingstype, wensen te visualiseren, moeten we eerst de gemiddelde vertraging berekenen.

```
## # A tibble: 42 x 4
## # Groups:   vertrek_luchthaven, type_vertraging [?]
##   vertrek_luchthaven type_vertraging weekdag_vertrek gem_vertraging
##   <chr>              <chr>          <ord>            <dbl>
## 1 John F Kennedy Intl aankomst_vertraging zo      6.39
## 2 John F Kennedy Intl aankomst_vertraging ma     7.99
## 3 John F Kennedy Intl aankomst_vertraging di     3.34
## 4 John F Kennedy Intl aankomst_vertraging wo     5.86
## 5 John F Kennedy Intl aankomst_vertraging do     7.56
## 6 John F Kennedy Intl aankomst_vertraging vr     6.49
## 7 John F Kennedy Intl aankomst_vertraging za     1.96
## 8 John F Kennedy Intl vertrek_vertraging zo    11.7
## 9 John F Kennedy Intl vertrek_vertraging ma    14.7
## 10 John F Kennedy Intl vertrek_vertraging di    10.5
## # ... with 32 more rows
```



7.4 Data in een breed formaat plaatsen (voor overzichtelijke tabellen)

- Voor de laatste visualisatie hebben we een dataset gecreëerd met gemiddelde vertragingen per vertrek-luchthaven, weekdag van vertrek en type vertraging.
- Om snel verbanden te zoeken en te evalueren is dit formaat niet erg handig. Voor zulke situaties kan je best voor een breed formaat opteren.
 - Hierbij moet je 2 variabelen selecteren: de key-variabele en de value-variabele.
 - De key-variabele is altijd een categorische variabele en de value-variabele kan zowel categorisch als continu zijn.
 - Voor ieder level van de categorische key-variabele zal er een aparte kolom aangemaakt worden.
- Je kan een dataset van lang naar breed formaat omzetten met behulp van de spread() functie.

```
df_long_summary %>%
  spread(key=weekdag_vertrek, value=gem_vertraging) %>%
  arrange(vertrek_luchthaven, type_vertraging)
```

7.5 Referenties

1. ‘R for Data Science’ van Grolemund en Wickham

Table 7.1: Gemiddelde vertraging (lang formaat)

weekdag_vertrek	vertrekluchthaven	type_vertraging	gem_vertraging
zo	John F Kennedy Intl	aankomst_vertraging	6.39
zo	John F Kennedy Intl	vertrek_vertraging	11.70
zo	La Guardia	aankomst_vertraging	0.15
zo	La Guardia	vertrek_vertraging	8.33
zo	Newark Liberty Intl	aankomst_vertraging	7.44
zo	Newark Liberty Intl	vertrek_vertraging	14.01
ma	John F Kennedy Intl	aankomst_vertraging	7.99
ma	John F Kennedy Intl	vertrek_vertraging	14.74
ma	La Guardia	aankomst_vertraging	9.58
ma	La Guardia	vertrek_vertraging	12.86
ma	Newark Liberty Intl	aankomst_vertraging	11.67
ma	Newark Liberty Intl	vertrek_vertraging	16.73
di	John F Kennedy Intl	aankomst_vertraging	3.34
di	John F Kennedy Intl	vertrek_vertraging	10.47
di	La Guardia	aankomst_vertraging	5.60
di	La Guardia	vertrek_vertraging	8.63
di	Newark Liberty Intl	aankomst_vertraging	7.15
di	Newark Liberty Intl	vertrek_vertraging	12.57
wo	John F Kennedy Intl	aankomst_vertraging	5.86
wo	John F Kennedy Intl	vertrek_vertraging	11.71
wo	La Guardia	aankomst_vertraging	6.23
wo	La Guardia	vertrek_vertraging	9.15
wo	Newark Liberty Intl	aankomst_vertraging	9.02
wo	Newark Liberty Intl	vertrek_vertraging	13.95
do	John F Kennedy Intl	aankomst_vertraging	7.56
do	John F Kennedy Intl	vertrek_vertraging	13.65
do	La Guardia	aankomst_vertraging	11.89
do	La Guardia	vertrek_vertraging	14.10
do	Newark Liberty Intl	aankomst_vertraging	15.60
do	Newark Liberty Intl	vertrek_vertraging	20.10
vr	John F Kennedy Intl	aankomst_vertraging	6.49
vr	John F Kennedy Intl	vertrek_vertraging	12.76
vr	La Guardia	aankomst_vertraging	7.97
vr	La Guardia	vertrek_vertraging	12.45
vr	Newark Liberty Intl	aankomst_vertraging	12.55
vr	Newark Liberty Intl	vertrek_vertraging	18.49
za	John F Kennedy Intl	aankomst_vertraging	1.96
za	John F Kennedy Intl	vertrek_vertraging	9.97
za	La Guardia	aankomst_vertraging	-5.44
za	La Guardia	vertrek_vertraging	4.19
za	Newark Liberty Intl	aankomst_vertraging	-2.22
za	Newark Liberty Intl	vertrek_vertraging	7.63

Table 7.2: Gemiddelde vertraging (breed formaat).

vertrek luchthaven	type_vertraging	zo	ma	di	wo	do	vr	za
John F Kennedy Intl	aankomst_vertraging	6.39	7.99	3.34	5.86	7.56	6.49	1.96
John F Kennedy Intl	vertrek_vertraging	11.70	14.74	10.47	11.71	13.65	12.76	9.97
La Guardia	aankomst_vertraging	0.15	9.58	5.60	6.23	11.89	7.97	-5.44
La Guardia	vertrek_vertraging	8.33	12.86	8.63	9.15	14.10	12.45	4.19
Newark Liberty Intl	aankomst_vertraging	7.44	11.67	7.15	9.02	15.60	12.55	-2.22
Newark Liberty Intl	vertrek_vertraging	14.01	16.73	12.57	13.95	20.10	18.49	7.63