KU LEUVEN kulak

Bachelor in de wiskunde,
bio-ingenieurswetenschappen,
fysica (keuze),
informatica (keuze)

Statistische modellen en data-analyse

Academiejaar 2018 – 2019

Oefeningen

Inhoudsopgave

1 Testprincipe

Inleiding. De oefeningen SMDA zullen erg praktijkgericht zijn en voortdurend R gebruiken voor de berekeningen. Het is dus noodzakelijk om de handleiding *Statistiek in R* grondig te herhalen, in het bijzonder de technieken rond hypothesetesten in sectie 4. Wat niet aan bod kwam in deze handleiding of in de basiscursus Statistiek, komt hieronder expliciet aan bod.

Random getallen. Dichtheidsfuncties, verdelingsfuncties en kwantielfuncties berekenen in R kan voor de meest courante verdelingen met commando's van de vorm `d<dist>()`, `p<dist>()` en `q<dist>()`, bijvoorbeeld `pnorm(x,mu,sigma)` of `qbinom(q,n,p)`. Bij elke verdeling hoort in R een (pseudo) random generator `r<dist>()` die een lijst van `n` random getallen uit de gegeven verdeling trekt. Deze random generatoren kunnen we onder meer gebruiken bij randomisatie en voor het bestuderen van verdelingen. Voor de normale en binomiale verdeling werkt deze random generator als volgt:

```
> rnorm(10, 5, 1)
 [1] 5.639560 4.697590 4.459021 5.374341 5.971379
 [6] 5.237486 4.348090 4.293595 4.425965 4.986101
> rbinom(10, 10, 0.2)
 [1] 1 0 0 2 2 1 2 1 1 4
```

Grafieken en programmeerstructuren in R. Er bestaan twee soorten grafiek-commando's in R. De high-level-commando's zoals `plot()` of `hist()` maken een nieuw grafiekvenster. Low-level-commando's zoals `points()` of `lines()` voegen elementen toe aan een bestaande grafiek. Volgende code gebruikt `for()`-lussen om een aantal normale dichtheidsfuncties naast elkaar te zetten. Meer informatie over programmeerstructuren en grafieken is te vinden in secties 1.4 en 3.7 van de handleiding *Statistiek in R*.

```
> x = seq(-7, 7, length=100)
> plot(c(-7,7), c(0,0.4), type='n')
> for (k in -2:2) {lines(x,dnorm(x,k,1),col=k+3)}
> for (k in 1:4) {lines(x,dnorm(x,0,k),col=k)}
```

Hypothesetesten in R. Het commando voor het uitvoeren van een hypothesetest in R is doorgaans van de vorm `<stat>.test()`. De output is typisch een samenvatting met de belangrijkste statistieken, maar het resultaat is een object dat via de attributen verder kan worden bevraagd.

```
> tt = t.test(x,mu=1); tt
        One Sample t-test
t = -2.4375, df = 99, p-value = 0.01657
        [...]
> attributes(tt)
$names
[1] "statistic"   "parameter"   "p.value"     "conf.int"    "estimate"
[6] "null.value"  "alternative" "method"      "data.name"
> tt$statistic
-2.437458
> tt$p.value
[1] 0.01657496
```

Kwaliteit van hypothesetesten. Een statistische test vertrekt steeds van het uitgangspunt dat de nulhypothese waar is, bijvoorbeeld $H_0 : \mu = \mu_0$. Op basis daarvan wordt dan berekend hoe waarschijnlijk het is een bepaalde steekproefwaarde $\overline{x}_n$ te vinden voor een gepaste statistiek $\overline{X}_n$. De nulhypothese wordt verworpen als de steekproefstatistiek uitzonderlijk afwijkt van de hypothese, wat dus niet noodzakelijk betekent dat de nulhypothese onwaar is, hoogstens dat deze onwaarschijnlijk is. De kans om op deze manier een ware nulhypothese te verwerpen is de type I-fout.

- Bereken 100 random getallen uit de standaardnormale verdeling.
- Gebruik een significantieniveau van 5% en een gepaste test om na te gaan of het steekproefgemiddelde significant verschilt van nul.
- Herhaal dit procedé 1000 keer en tel hoe vaak de (ware!) nulhypothese wordt verworpen.

Omgekeerd is een nulhypothese die wordt aanvaard niet noodzakelijk waar. In werkelijkheid kan een van de nulhypothese afwijkend populatiegemiddelde μ_1 toch aanleiding geven tot een steekproefgemiddelde in het aanvaardingsgebied. Dit geeft aanleiding tot een type II-fout. De kans dat die voorkomt, hangt af van de grootte van het verschil tussen het hypothetische gemiddelde μ_0 en het populatiegemiddelde μ_1: een zeer klein verschil zal vaak toch aanleiding geven tot een aanvaardbare steekproefwaarde, terwijl een groot verschil tussen beide meestal wel zal leiden tot een significant verschil.

- Bereken 100 random getallen uit de $N(\mu_1, 1)$-normale verdeling voor een zelf gekozen waarde $\mu_1 \neq 0$ en test of het steekproefgemiddelde significant verschilt van nul.
- Tel hoe vaak de (valse!) nulhypothese wordt aanvaard als deze test 1000 keer wordt herhaald voor verschillende steekproeven.
- Herhaal dit procedé voor enkele verschillende (interessante!) waarden μ_i en tel telkens het aantal type II-fouten n_i. Maak een grafiek van de punten (μ_i, n_i) en bestudeer het verloop.

Steekproefgrootte en Centrale Limietstelling. Een t-test is slechts betrouwbaar onder de voorwaarde dat het steekproefgemiddelde normaal verdeeeld is. Als hieraan niet voldaan is zal het gebruikte significantieniveau niet overeenkomen met de type I-fout, de test is niet betrouwbaar.

- Bereken $n = 10$ random getallen uit een binaire verdeling met kans op succes $p = 10\%$ en gebruik een t-test voor één gemiddelde om na te gaan of het steekproefgemiddelde significant verschilt van $p = 0.1$.
- Tel hoe vaak de (ware!) nulhypothese wordt verworpen als deze test 1000 keer wordt herhaald voor verschillende steekproeven.

Gepoolde of afzonderlijke variantieschattingen. In het geval twee gemiddelden worden vergeleken uit ongepaarde groepen waarnemingen, is een belangrijke vraag of er een gepoolde variantieschatting kan worden gebruikt, dan wel of er afzonderlijke variantieschattingen nodig zijn.

- Voer beide t-tests uit in het geval van twee random steekproeven van telkens 50 elementen uit een $N(0, 1)$- en een $N(0, 10^2)$-verdeling. Herhaal dit procedé 1000 keer en schat in beide gevallen de type I-fout.
- Herhaal hetzelfde, maar nu in het geval dat de ene steekproef 10 keer groter is dan de andere. Vergelijk opnieuw de type I-fouten.

Deel I
Multivariate statistiek

2 Cluster analyse

Extra libraries gebruiken. Extra bibliotheken dienen met `install.packages()` te worden geïnstalleerd voor het eerste gebruik. Kies bij de allereerste installatie-opdracht voor de Belgische server van de Associatie KU Leuven. De pakketten worden automatisch gedownload en kunnen daarna worden gebruikt. Om een pakket na installatie effectief te gebruiken, is er het commando `library()`. Een pakket dient in elke `R`-sessie opnieuw te worden ingeladen. Eenmalig een commando uit een pakket gebruiken (zonder het dus helemaal in te lezen) of gelijknamige commando's uit verschillende pakketten door elkaar gebruiken kan met de constructie `pakket::commando`.

```
> install.packages("cluster") # Eenmalig
> library(cluster)            # Elke sessie
```

Partitionerende algoritmes. De bedoeling van clustering is om binnen een dataset te zoeken naar groepen van gelijkaardige meetpunten. Er zijn verschillende technieken en voorstellingswijzen die elk bijkomend afhangen van de manier waarop wordt gemeten in hoeverre twee meetpunten van elkaar verschillen (*dissimilarity*). Hieronder een overzicht van de belangrijkste partitionerende methoden en voorstellingswijzen uit het pakket `cluster`. De eigenlijke clustering zit steeds vervat in een attribuut `$cluster` of `$clustering`.

```
> X.km = kmeans(X,k)            # K-means clustering
> X.km$cluster                  # Vector met groepsindeling
> X.pam = pam(X,k)              # Partitioning Around Medoids
> silhouette(X.pam)             # Silhouette plot
> clusplot(X.pam)               # Cluster Plot
> X.fa = fanny(X,k)             # Fuzzy Analysis
> X.fa$membership               # Matrix met memberships
```

Dataset iris. De dataset `iris` (uit het standaard ingeladen pakket `datasets`) bevat afmetingen van de bloemblaadjes van drie iris-variëteiten. Cluster analyse laat toe om deze bloemen in groepen in te delen op basis van deze afmetingen om te kijken of er zich groepen aftekenen, zonder voorkennis over de werkelijke variëteit.

In dit geval is de variëteit gekend en kan worden nagegaan in welke mate een clustering op basis van de afmetingen van de bloemblaadjes overeenkomt met de werkelijke variëteiten. Een essentieel andere techniek is om te proberen de variëteit uit de afmetingen af te leiden, dat is classificatie en komt in een later hoofdstuk aan bod.

- Maak een boxplot van de fysieke afmetingen van de bloemblaadjes per soort en ga na dat die inderdaad verschillen per soort. Uit voorkennis over de concrete groepen lijkt het dus inderdaad mogelijk om soorten te herkennen.
- Ga na in welke mate K-means clustering de variëteiten herkent op basis van de vier afmetingen. Gebruik als tweede argument een 3×4-matrix met startwaarden. Kies daarvoor de kwartielen van elke veranderlijke. Is het nodig de gegevens te herschalen? Vergelijk het resultaat met en zonder herschaling.
- Is het tweede argument enkel een getal, dan bepaalt dit het aantal clusters en gebruikt het commando `kmeans()` verder random startwaarden. Dit kan in principe tot een andere indeling leiden. Bereken enkele dergelijke clusteringen en vergelijk met deze waarbij de kwartielen als startwaarden golden.
- Vergelijk de resultaten van K-means clustering met die van het PAM-algoritme.
- Bereken de silhouette-waarden $s(i) \in [-1, 1]$ van elk object i met `silhouette()` en onderzoek deze met `summary()` en `plot()`. Goed geclassificeerde waarden hebben een silhouette rond 1, ligt de waarde $s(i)$ dicht bij nul, dan is het object moeilijk in te delen. Negatieve waarde duiden op een slechte indeling. Ga na of verkeerd ingedeelde bloemen inderdaad een kleine silhouette-breedte hebben en of het dichtstbij zijnde cluster in die gevallen met de correcte soort overeenstemt.

- De gemiddelde silhouette-waarde $\overline{s(i)}$ of silhouette-breedte is een maat voor de kwaliteit van de indeling in clusters die afhangt van het aantal groepen. Voor een bepaald aantal clusters is de silhouette-breedte maximaal, de *silhouette-coëfficiënt*. Voor hoeveel groepen detecteert het PAM-algoritme de sterkste structuur?
- Teken ook de cluster-plot met `clusplot()` die de gevonden clustering voorstelt op een grafiek met de eerste twee principaalcomponenten.
- Fuzzy clustering berekent een gewicht u_{ij} met $\sum_{j=1}^{K} u_{ij} = 1$ dat aangeeft hoe sterk object i bij cluster j hoort. Verifieer dat deze gewichten bij verkeerd ingedeelde bloemen inderdaad niet eenduidig zijn. Bereken uit de `membership`-matrix de Dunn-coëfficiënt $\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k} u_{ij}^2$.

Hiërarchische algoritmes. Waar partitionerende methoden starten van een vast aantal clusters, zullen hiërarchische algoritmes focussen op de samenhang tussen indelingen met een oplopend aantal clusters: van één grote cluster tot n clusters die telkens slechts één object bevatten.

```
> X.an = agnes(X, method=...)   # Agglomerate Nesting
> X.da = diana(X)               # Divisive Analysis
> bannerplot(X.an)              # Banner Plot
> pltree(X.an)                  # Dendrogram
> cutree(X.an)                  # Cut tree in clusters
```

- Hoeveel groepen en welke indeling suggereren de hiërarchische methodes in bovenstaande dataset?
- Vergelijk de verschillende dendrogrammen en banner plots bij onderstaande dissimilariteitsmatrix (optie `diss=TRUE`) naargelang agglomerate clustering toegepast wordt met de *group average*-, *single linkage*- of *complete linkage*-methode (optie `method`).

	a	b	c	d	e
a	0				
b	2	0			
c	6	3	0		
d	8	7	5	0	
e	9	6	5	4	0

- Tracht groepen te onderscheiden in de `MASS::crabs`-dataset, maak grafische voorstellingen en interpreteer deze. Wat is het verschil tussen de verschillende groepen? Hebben deze iets te maken met het geslacht `sex` en de soort `sp` van de krab.
- Zijn er duidelijke groepen aanwezig in de dataset `car::Prestige`? Hoe onderscheiden deze groepen zich? Is het mogelijk om het type job (`wc`, `bc`, `prof`) te herkennen in deze groepen?

3 Multivariate verdeling en schatters

Matrixrekenen in R. Om vertrouwd te geraken met het matrixformalisme uit de cursus kan het nuttig zijn om de formules door te rekenen in R. Daarvoor is het nodig om matrices te construeren, transponeren, inverteren en vermenigvuldigen, waarvoor hieronder voorbeeldcode te vinden is. Belangrijke opmerkingen zijn dat R niet steeds een duidelijk onderscheid maakt tussen rij- of kolomvectoren, dat een 1×1-matrix niet steeds als scalair kan worden gebruikt en dat R andere methoden gebruikt voor de klassen `matrix` en `data.frame`

```
> x = c(1,2,3)
> y = c(4,5,6)
> A = cbind(x,y);A
     x y
[1,] 1 4
[2,] 2 5
[3,] 3 6
> t(A)
  [,1] [,2] [,3]
x    1    2    3
y    4    5    6
> B = t(A) %*% A; B
   x  y
x 14 32
y 32 77
> eigen(B)
$values
[1] 90.4026725  0.5973275
$vectors
           [,1]       [,2]
[1,] 0.3863177 -0.9223658
[2,] 0.9223658  0.3863177
> C = solve(B); C
            x          y
x  1.4259259 -0.5925926
y -0.5925926  0.2592593
> B %*% C
  x y
x 1 0
y 0 1
> diag(3)
     [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
> xx = x %*% x; xx
[1,]   14
> xx*A
Error in xx * A : non-conformable arrays
> as.numeric(xx)*A
x  y
[1,] 14 56
[2,] 28 70
[3,] 42 84
> D = data.frame(x,y)
> t(D)%*%D
Error in t(D) %*% D : requires numeric/complex matrix/vector arguments
> t(as.matrix(D))%*%as.matrix(D)
x  y
x 14 32
```

```
y 32 77
> mean(A)
[1] 3.5
> colMeans(A)
x y
2 5
> colMeans(D)
x y
2 5
```

Beschrijvende multivariate statistiek. Bereken met onderstaande code een gegevensmatrix $X \in \mathbb{R}^{100\times 3}$ en bereken volgende grootheden enkel door gebruik te maken van matrixrekenen. Vergelijk met het ingebouwde commando.

- Het multivariate steekproefgemiddelde $\overline{\boldsymbol{x}} = \frac{1}{n}(\mathbf{1}_n \cdot X)$ waarin $\mathbf{1}_n = (\underbrace{1, \ldots, 1}_{n})$;
- De SSCP matrix $W = (X - \mathbf{1}_n^\tau \cdot \overline{\boldsymbol{x}})^\tau \cdot (X - \mathbf{1}_n^\tau \cdot \overline{\boldsymbol{x}})$;
- De empirische covariantiematrix $S = \frac{1}{n-1}W$;
- De empirische correlatiematrix $R = D^{-1/2}SD^{-1/2}$ met $D = \mathrm{diag}(s_{11}, \ldots, s_{pp})$.

Interpreteer de bekomen correlaties en lees het effect af van de matrix van scatterplots bekomen met het commando `pairs(X)`.

```
> x = rnorm(100)
> y = rnorm(100)
> z = x-2*y
> X = cbind(x,y,z)
```

De Mahalanobisafstand. Om te bepalen hoe ver een punt van het gemiddelde ligt, worden drie mogelijke maten geïntroduceerd:

- De Euclidische afstand $d_E(\boldsymbol{x}, \overline{\boldsymbol{x}}) = \sqrt{(\boldsymbol{x} - \overline{\boldsymbol{x}})^\tau(\boldsymbol{x} - \overline{\boldsymbol{x}})}$ van het meetpunt $\boldsymbol{x}$ tot het gemiddelde $\overline{\boldsymbol{x}}$;
- De Euclidische afstand $d_E(\boldsymbol{x}_s, \mathbf{0}) = \sqrt{(\boldsymbol{x} - \overline{\boldsymbol{x}})^\tau D^{-1}(\boldsymbol{x} - \overline{\boldsymbol{x}})}$ van het gestandaardiseerde punt tot de oorsprong;
- De Mahalanobisafstand $d_S(\boldsymbol{x}, \overline{\boldsymbol{x}}) = \sqrt{(\boldsymbol{x} - \overline{\boldsymbol{x}})^\tau S^{-1}(\boldsymbol{x} - \overline{\boldsymbol{x}})}$ van het meetpunt $\boldsymbol{x}$ tot het gemiddelde $\overline{\boldsymbol{x}}$.

Enkel de laatste afstandsfunctie beschrijft in hoeverre een punt zich van het centrum van de (multivariaat normale) puntenwolk distantieert.

- Neem aan dat in een bepaalde steekproef de gemiddelde lengte 1.85 m is, met standaarddeviatie 0.1 m. De gemiddelde massa is 85 kg met standaarddeviatie 10 kg. De correlatie tussen lengte en massa is 0.9. Bereken dan de afstand van vier datapunten **a**, **b**, **c** en **d** tot het gemiddelde $\overline{\mathbf{x}} = (1.85, 85)$. Reken de Mahalanobisafstand na met `mahalanobis(x,m,S)`, waarin de datamatrix `x` precies p kolommen telt, het centrum `m` een vector van lengte p is en `S` de $p \times p$-covariantiematrix.

	Massa	Lengte
$\overline{\mathbf{x}}$	75	1.75
a	75	1.85
b	85	1.75
c	85	1.85
d	65	1.85

Meetkundige interpretatie. Punten op gelijke Mahalanobisafstand van het gemiddelde liggen op een ellips of (hyper)ellipsoïde naargelang het aantal veranderlijken p. Enkel in het tweedimensionale geval is dit makkelijk te visualiseren.

```
> library(ellipse)
> x = rnorm(100); y = rnorm(100)
> A = cbind(x,y)
> plot(x,y,main="Twee ongecorreleerde veranderlijken")
> points(mean(x),mean(y),pch=8,col='red')
> lines(ellipse(cov(A),centre=colMeans(A),level=.9),col='red')
```

- In het voorbeeld zijn de twee veranderlijken onafhankelijk. Teken een punt dat in x- én y-richting één standaardafwijking groter is dan gemiddeld. Wijkt dit punt uitzonderlijk ver af van het gemiddelde? Maak nu de scatterplot $(x, y - 2x)$ en duid opnieuw het punt aan dat één standaardafwijking groter is dan gemiddeld in x- én y-richting. Bekijk de afwijking ten opzichte van het gemiddelde.
- De betekenis van `level` is enkel van toepassing als de punten zich volgens een multivariaat normale verdeling gedragen: uit het vervolg blijkt dat de kwadratische Mahalanobisafstand van een p-variaat normaal verdeelde steekproef χ^2_p-verdeeld is. Teken nu de ellips door het hierboven getekende punt, dat we P noemen. Dit kan je doen door de kans p_χ te berekenen dat een χ^2_2-verdeelde variabele een waarde aanneemt kleiner dan $d^2_S(P, \bar{x})$ (gebruik hiervoor het commando `pchisq()`). De ellips door het punt P bekom je door `level`$= p_\chi$ te stellen in het commando `ellipse()`.

Multivariate normaliteit nagaan. Hoewel veel technieken vereisen dat de steekproef $X \in \mathbb{R}^{n\times p}$ is getrokken uit een multivariaat normale verdeling $N_p(\boldsymbol{\mu}, \Sigma)$, is het niet gemakkelijk om na te gaan of aan deze voorwaarde is voldaan. Volgende criteria laten (hoogstens) toe om na te gaan of de veranderlijken niet al te zeer afwijken van multivariate normaliteit:

1. Univariate marginalen. Voer (univariate) testen voor normaliteit uit voor elke component.
2. Bivariate marginalen. Plot de componenten twee aan twee en ga na of de data elliptische omtreklijnen hebben.
3. Radiale marginalen. Ga met een kwantielplot (`qqplot()` en `ppoints()`) na of de kwadratische Mahalanobisafstanden de χ^2_p-verdeling volgen.

- Ga na of de lengte `CL` en breedte `CW` van het schild van een krab op basis van de data `MASS::crabs` uit de bibliotheek `MASS` een bivariaat normale verdeling volgen.
- Herhaal dezelfde analyse om na te gaan of de vijf continue veranderlijken N_5-verdeeld zijn.

Hotelling test voor één groep. In essentie is de Hotelling T^2-test de multivariate versie van de t-test voor gemiddelden. Gegeven een onafhankelijke steekproef uit een $N_p(\boldsymbol{\mu}, \Sigma)$-verdeling, is het mogelijk om te testen of het steekproefgemiddelde $\overline{X}$ significant verschilt van een hypothetische waarde $\boldsymbol{\mu}_0$ met behulp van de statistiek

$$\frac{n(n-p)}{p(n-1)}(\overline{X} - \boldsymbol{\mu}_0)^t S^{-1}(\overline{X} - \boldsymbol{\mu}_0) \sim F_{p,n-p}.$$

- Ga met twee univariate testen de hypothesen na of de gemiddelde lengte van een krabbenschild significant verschilt van 32.5 mm en of de breedte van zo een schild gemiddeld gelijk is aan 36 mm.
- Voer met behulp van het commando `T2.test(X,mu=c(32.5,36))` uit de `rrcov`-library de Hotelling test voor de gezamenlijke hypothese uit.
- Bereken de univariate 95% betrouwbaarheidsintervallen en duid deze samen met steekproefgemiddelde en hypothetisch gemiddelde aan op een puntenplot. Teken de bivariate betrouwbaarheidsellips en interpreteer de gevonden resultaten.
- Hoe zal het verschil tussen 2 t-testen enerzijds en een T^2-test anderzijds veranderen naarmate de betrokken veranderlijken meer of minder gecorreleerd zijn?
- Reken de Hotelling test na met behulp van bovenstaande formule.

Hotelling test voor twee groepen. Analoog kan de Hotelling-test gebruikt worden om simultaan de gemiddelden uit twee groepen te vergelijken, de syntax is dan `T2.test(X,Y)` met in elk argument de data uit de desbetreffende groep.

- Gebruik univariate tests om te bepalen of de lengte en breedte van krabbenschilden bij mannelijke en vrouwelijke krabben verschillen.
- Duid beide groepen meetwaarden aan op een grafiek en teken passende ellipsen om na te gaan of een bivariaat normale verdeling aanvaardbaar is.
- Voer de Hotelling test uit voor de hypothese $H_0 : (\mu_{\text{lengte}}, \mu_{\text{breedte}})_{\male} = (\mu_{\text{lengte}}, \mu_{\text{breedte}})_{\female}$ en geef een interpretatie.

4 Principaalcomponentenanalyse

Constructie. De bedoeling van principaalcomponentenanalyse is om een groot aantal variabelen te beschrijven aan de hand van slechts enkele veranderlijken, typisch onderliggende effecten, door lineaire combinaties van de originele veranderlijken te nemen die de beschreven variantie maximaliseren.

Neem een gegevensmatrix $X = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^\tau = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_p) \in \mathbb{R}^{n\times p}$ met observaties $\boldsymbol{x}_i \in \mathbb{R}^p$ en *gestandaardiseerde* veranderlijken $\boldsymbol{v}_j \in \mathbb{R}^n$. Dan kan de correlatiematrix $S = \mathrm{cor}(X) = \frac{1}{n-1}X^\tau X$ worden gediagonaliseerd door een orthogonale transformatie $P = (\boldsymbol{p}_1, \ldots, \boldsymbol{p}_p)$,

$$P^\tau S P = \mathrm{diag}(\lambda_1, \ldots, \lambda_p),$$

waarbij $\boldsymbol{p}_i$ en λ_i de eigenvectoren respectievelijk eigenwaarden zijn. Deze transformatie resulteert in een nieuw coördinatensysteem van zogenaamde principaalcomponenten $Y = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_p)$ die niet gecorreleerd zijn,

$$Y = XP \text{ met } \mathrm{cor}(Y) = \frac{1}{n-1}Y^\tau Y = \frac{1}{n-1}P^\tau X^\tau X P = \mathrm{diag}(\lambda_1, \ldots, \lambda_p).$$

De i-de principaalcomponent $\boldsymbol{y}_i$ = wordt dus bepaald als lineaire combinatie uit de oorspronkelijke veranderlijken met behulp van de i-e eigenvector $\boldsymbol{p}_i$.

Totale variantie. Het spoor van de variantie-covariantiematrix S, is een maat voor de totale ruimtelijke variabiliteit en is ook gelijk aan de som van eigenwaarden van S,

$$\begin{aligned} \mathrm{Tr}\, S &= \frac{1}{n-1}\sum_{j=1}^{p}\sum_{i=1}^{n} x_{ij}^2 = \sum_{j=1}^{p} \mathrm{Var}\, \boldsymbol{v}_i \\ &= \sum_{j=1}^{p} \lambda_j = \sum_{j=1}^{p} \mathrm{Var}\, \boldsymbol{y}_i. \end{aligned}$$

Dit toont dat een plausibele maat voor het aandeel van de i-de principaalcomponent $\boldsymbol{y}_i$ in de totale variantie gelijk is aan

$$\frac{\lambda_i}{\sum_{j=1}^{p}\lambda_j}.$$

Interpretatie. Onder de conventie van aflopend gesorteerde eigenwaarden, $\lambda_1 \geq \cdots \geq \lambda_p$, bepalen de eerste $r < p$ eigenwaarden dus het belangrijkste deel van de variabiliteit en is het mogelijk om een meerdimensionaal probleem te reduceren tot twee of drie dimensies. De componenten van de eigenvectoren $\boldsymbol{p}_i$ geven het aandeel van de oorspronkelijke veranderlijken in de weerhouden principaalcomponenten aan en laten idealiter toe om een betekenis te hechten aan deze nieuwe veranderlijken. Als de oorspronkelijke gegevens multivariaat normaal verdeeld zijn, bepalen de eigenvectoren de assen van de ellipsen bepaald door gelijke Mahalanobisafstand.

Dataset crabs. Herneem de dataset `crabs` uit de bibliotheek `MASS` die gegevens bevat over 200 krabben: telkens 5 morfologische gegevens van 50 mannetjes en vrouwtjes van twee varianten, blauwe en oranje. Principaalcomponenten zullen toelaten om op basis van deze fysieke kenmerken onderscheid te maken tussen de verschillende soorten.

- Bereken met `eigen()` de eigenwaarden $\lambda_1 \geq \ldots \geq \lambda_5$ van de correlatiematrix van de *gestandaardiseerde gegevens* (gebruik `scale()`) en teken de *scree plot* (i, λ_i). Hoeveel van de variatie wordt er bepaald door de belangrijkste principaalcomponenten? Bekijk de eigenvectoren en tracht deze te interpreteren. Vergelijk deze resultaten met de output van `prcomp()`, een object dat je bevraagt met `attributes()`, `plot()` en `summary()`.
- Bereken door matrixvermenigvuldiging de getransformeerde gegevensmatrix Y en vergelijk de gevonden matrix met `predict()`. Geef punten op de grafieken (y_1, y_2), (y_1, y_3) en (y_2, y_3) elk een kleur (`col`) en symbool (`pch`) naargelang het soort en het geslacht van de krab. Hoe kan je geslacht en soort aflezen van de morfologische kenmerken van de krab?
- Maak met `biplot()` een grafiek die informatie over de observaties $\boldsymbol{x}_i$ en de variabelen $\boldsymbol{v}_j$ bundelt. Tracht deze voorstelling te interpreteren.

Dataset UScereal. Deze gegevens uit het pakket `MASS` bevatten voedingsinformatie over 65 soorten ontbijtgranen.

- Zet potassium en sodium (uitgedrukt in milligram per cup) om naar gram per cup. Voer PCA uit op alle veranderlijken die massa(dichtheid) voorstellen. Doe dit zonder en met herschalen en vergelijk de resultaten. Welke van beide technieken is aangewezen?
- Welke granen onderscheiden zich het meest van de andere en waarom?
- Welke granen liggen op de bovenste plank?

5 Classificatiemethodes

Clustering versus classificatie. De clusteringmethodes uit het vorige hoofdstuk gebruiken de structuur van data om na te gaan of en hoeveel groepen er zich in een gegevensverzameling aftekenen. Er wordt daarbij geen a priori verdeling van de data in groepen gebruikt. De voorbeelden in de vorige sectie illustreren dat een dergelijke clustering in zekere mate kan overeenstemmen met een bestaande categorische veranderlijke, maar dat is zeker niet altijd zo en dat is ook niet de opzet. Bij classificatiemethodes is het wel de bedoeling om een bestaande indeling te beschrijven en om te voorspellen tot welke groep een bepaald object behoort: gegeven verklarende veranderlijken $X_1, \ldots, X_p$ is het de bedoeling een categorische veranderlijke Y te verklaren. Hoewel de berekeningsmethode en de voorwaarden drastisch verschillen, is het doel van classificatiemethodes dus quasi identiek met dat van (logistische) regressie, zij het voor een categorische in plaats van continue (of binaire) respons.

Van a priori naar a posteriori-kans. Gegeven een p-variate veranderlijke $\boldsymbol{X}$ die behoort tot één populatie π_i met a priori-kans $p_i = P(\boldsymbol{X} \in \pi_i)$, is het de bedoeling om regio's R_i af te bakenen zo dat de indeling $\boldsymbol{x} \in R_i$ zo goed mogelijk overeenstemt met de populaties π_i $(i = 1, \ldots, g)$. Is de dichtheid f_i van de veranderlijke $\boldsymbol{X}$ voor elke populatie π_i bekend, dan kan de a posteriori-kans dat een observatie $\boldsymbol{x}$ tot de populatie π behoort, berekend worden als

$$P(\boldsymbol{X} \in \pi_i \mid \boldsymbol{X} = \boldsymbol{x_0}) = \frac{p_i f_i(\boldsymbol{x_0})}{\sum_{j=1}^{g} p_j f_j(\boldsymbol{x_0})}.$$

Deze methode werkt dus enkel als de dichtheden f_i bekend zijn, bijvoorbeeld in het geval van multivariaat normaal verdeelde veranderlijke $\boldsymbol{X} \sim N_p(\mu_i, \Sigma_i)$,

$$f_i(\boldsymbol{x_0}) = \frac{1}{(2\pi)^{p/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x_0} - \boldsymbol{\mu}_i)^\tau \Sigma_i^{-1}(\boldsymbol{x_0} - \boldsymbol{\mu}_i)\right).$$

Lineaire discriminantmethode. Als er kan worden uitgegaan van homogene covariantiematrices kunnen alle dichtheden worden berekend met dezelfde gepoolde schatter

$$\hat{\Sigma}_p = \sum_{i=1}^{g} \frac{n_i - 1}{N - g} \hat{\Sigma}_i.$$

Wordt een observatie $\boldsymbol{x_0}$ ingedeeld in groep i waarvoor $P(\boldsymbol{X} \in \pi_i \mid \boldsymbol{x} = \boldsymbol{x}_0)$ maximaal is, dan resulteert dit in lineaire voorwaarden en spreekt men van de lineaire discriminantmethode.

Kwadratische discriminantmethode. Wordt in elke groep i een afzonderlijke schatter $\hat{\Sigma}_i$ voor de covariantiematrix gebruikt, dan resulteert de methode in kwadratische voorwaarden en spreekt men van de kwadratische discriminantmethode.

Nearest neighbours. In het geval de verdeling niet normaal is, levert de *k-nearest neighbour method* een alternatief. Deze gebruikt geen informatie over verdeling en gebruikt de Euclidische in plaats van Mahalanobisafstand. Dat maakt de methode in het algemeen minder krachtig maar wel breder toepasbaar.

Kwaliteit van de classificatie. Om te bestuderen hoe goed een classificatieregel op basis van een steekproef is, wordt de kans op misclassificatie berekend. De meest naïeve maat hiervoor is de *apparent error rate* (APER), de verhouding van verkeerd geklasseerde observaties uit de steekproef op het totaal aantal observaties. Dit levert typisch onderschattingen aangezien de classificatie berekend wordt op basis van de steekproef zelf. Alternatief kan voor elke observatie in de steekproef de classificatie berekend worden terwijl dat ene element buiten beschouwing wordt gelaten, deze zogenaamde *leave-one-out*-procedure levert een onvertekende schatter voor de foutkans: de *actual error rate* (AER).

Discriminantanalyse in R. De commando's `lda()` en `qda()` uit de bibliotheek `MASS` berekenen de coëfficiënten voor de lineaire en kwadratische discriminantenmethode, met de optie `CV=TRUE` volgens *leave-one-out cross-validation*. Met de methode `predict()` kunnen vervolgens de a posteriori-kansen (`$posterior`) en classificatie (`$class`) worden bekomen. Het commando `partimat()` uit de bibliotheek `klaR` geeft een grafische voorstelling voor alle bivariate modellen met telkens twee verklarende veranderlijken. De k-nearest neighbour-methode is geïmplementeerd in het commando `knn()` uit de `class`-bibliotheek (standaard met argument voor training en validation set, buiten bestek van de cursus). Om formules na te rekenen is er de multivariaat normale dichtheid `dmnorm()` uit het pakket `mnormt`.

```
> library(MASS)
> lda(X,y)
> lda(y~x1+...+xp)
> lda(y~x1+...+xp, CV=TRUE)
> predict(lda(X,y))$class
> predict(lda(X,y))$posterior
> library(klaR)
> partimat(y~x1+x2, method='lda', imageplot=FALSE)
> partimat(y~x1+...+xp, method='lda', imageplot=FALSE, plot.matrix=TRUE)
> library(class)
> knn(train,test,cv,k)
> knn.cv(train,cv,k)
> library(mnormt)
> dmnorm(X,mu,Sigma)
```

- Gebruik de discriminantmethode om het geslacht van krabben in de dataset `crabs` te voorspellen: Zijn de veronderstellingen voldaan? Vergelijk grafisch de lineaire en kwadratische methode. Bereken APER en AER.
- Wat is de kans dat een krab van het vrouwelijke geslacht is, te weten dat *frontal lobe* en de *rear width* beide 10 mm meten?
- Neem aan dat het geslacht van krabben gedetermineerd moet worden voor een (fictief) kweekprogramma waarbij één mannetje telkens meerdere vrouwtjes dient te bezwangeren. De kost $c(\boldsymbol{x} \in R_{\male} \mid \pi_{\female})$ voor misclassificatie van een mannetje als vrouwtje wordt tien keer hoger ingeschat dan de misclassificatiekost $c(\boldsymbol{x} \in R_{\female} \mid \pi_{\male})$. Gebruik onderstaande formule om te bepalen hoe de krabben dan best worden ingedeeld,

$$(\overline{\boldsymbol{x}}_i - \overline{\boldsymbol{x}}_j)^\tau \hat{\Sigma}_p^{-1} \boldsymbol{x}_0 - \frac{1}{2}(\overline{\boldsymbol{x}}_i - \overline{\boldsymbol{x}}_j)^\tau \hat{\Sigma}_p^{-1}(\overline{\boldsymbol{x}}_i + \overline{\boldsymbol{x}}_j) \geq \log\left(\frac{c(\boldsymbol{x} \in R_i \mid \pi_j)\hat{p}_j}{c(\boldsymbol{x} \in R_j \mid \pi_i)\hat{p}_i}\right)$$

  Vergelijk met de resultaten in geval van gelijke kost (via de formule en door ingebouwde commando's).
- Voer discriminantanalyse en de k-nearest neighbour methode uit om in de dataset `Prestige` het type van een beroep (`wc`, `bc`, `prof`) te bepalen. Vergelijk de resultaten en vergelijk telkens ook met de getransformeerde data `log10(income)` en `logit(women)` (zie sectie 3).

Deel II
Regressie

6 Matrixformalisme

De regressievergelijking. In matrixvorm kunnen de vergelijkingen

$$y_i = \beta_0 + \beta_1 \cdot x_{i1} + \beta_2 \cdot x_{i2} + \ldots + \beta_{p-1} \cdot x_{i,p-1} + \varepsilon_i$$

met $i = 1, \ldots, n$ geschreven worden als

$$\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

waarin

- $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)^t$ de responsveranderlijke,
- $X = \begin{pmatrix} 1 & x_{11} & \ldots & x_{1,p-1} \\ 1 & x_{21} & \ldots & x_{2,p-1} \\ \vdots & \vdots & & \vdots \\ 1 & x_{n1} & \ldots & x_{n,p-1} \end{pmatrix}$ met x_{ij} de i-e observatie van de j-e veranderlijke,
- $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_{p-1})^t$ de coëfficiëntenvector,
- $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n)^t$ de residuen.

Kleinste kwadratenschatters. De kleinste kwadratenmethode minimaliseert $\boldsymbol{\varepsilon}^t\boldsymbol{\varepsilon} = \sum_{i=1}^n \varepsilon_i^2$ in functie van $\boldsymbol{\beta}$,

$$\begin{aligned} 0 &= \nabla\left((\boldsymbol{y} - X\boldsymbol{\beta})^t(\boldsymbol{y} - X\boldsymbol{\beta})\right) \\ &= -2X^t\boldsymbol{y} + 2X^tX\boldsymbol{\beta}. \end{aligned}$$

Dit leidt tot de voorwaarde

$$X^tX\boldsymbol{\beta} = X^t\boldsymbol{y}.$$

Hieruit kunnen met behulp van de zogenaamde *hat matrix* $H = X(X^tX)^{-1}X^t$ kleinste kwadratenschattingen worden berekend voor de coëfficiënten $\hat{\boldsymbol{\beta}}_{LS}$, respons $\hat{\boldsymbol{y}}$ en residuen $\boldsymbol{e}$.

$$\begin{aligned} \hat{\boldsymbol{\beta}} &= (X^tX)^{-1}X^t\boldsymbol{y}, \\ \hat{\boldsymbol{y}} &= X(X^tX)^{-1}X^t\boldsymbol{y} = H\boldsymbol{y}, \\ \boldsymbol{e} &= (I_n - H)\boldsymbol{y}, \end{aligned}$$

Schattingen voor de variantie en de variantie-covariantiematrices worden bekomen met behulp van de rekenregel $\Sigma(A\boldsymbol{X}) = A\Sigma(\boldsymbol{X})A^t$,

$$\begin{aligned} S^2 &= \frac{\boldsymbol{e}^t\boldsymbol{e}}{n-p}, \\ \hat{\Sigma}(\hat{\boldsymbol{\beta}}) &= S^2(X^tX)^{-1}, \\ \hat{\Sigma}(\hat{\boldsymbol{y}}) &= S^2H, \\ \hat{\Sigma}(\boldsymbol{e}) &= S^2(I_n - H). \end{aligned}$$

Equivariantie-eigenschappen. De kleinste kwadratenmethode levert schatters voor de regressiecoëfficiënten met interessante eigenschappen, ze zijn regressie-, schaal- en affien equivariant:

$$\begin{aligned} \hat{\boldsymbol{\beta}}(\boldsymbol{x}_i, y_i + \boldsymbol{x}_i^\tau\boldsymbol{v}) &= \hat{\boldsymbol{\beta}}(\boldsymbol{x}_i, y_i) + \boldsymbol{v}, \\ \hat{\boldsymbol{\beta}}(\boldsymbol{x}_i, cy_i) &= c\hat{\boldsymbol{\beta}}(\boldsymbol{x}_i, y_i), \\ \hat{\boldsymbol{\beta}}(A\boldsymbol{x}_i, y_i) &= (A^\tau)^{-1}\hat{\boldsymbol{\beta}}(\boldsymbol{x}_i, y_i). \end{aligned}$$

Het gestandaardiseerde regressiemodel. Voor gestandaardiseerde waarden $y'_i = \frac{y_i - \overline{y}}{s_y}$ en $x'_{ij} = \frac{x_{ij} - \overline{x}_j}{s_j}$ valt de intercept weg uit het model en wordt de regressievergelijking

$$y'_i = \beta'_1 \cdot x'_{i1} + \beta'_2 \cdot x'_{i2} + \ldots + \beta'_{p-1} \cdot x'_{i,p-1} + \varepsilon'_i$$

met $\beta'_j = \frac{s_j}{s_Y}\beta_j$ en $\varepsilon' = \frac{\varepsilon_i}{s_y}$.

Dataset Prestige. Gebruik de dataset `Prestige` uit het pakket `car` om bovenstaande formules als volgt na te rekenen teneinde `prestige` te verklaren in functie van `education`, `income` en `women`.

- Schrijf in bovenstaande formules bij elke matrix de dimensies. Reken de kleinste kwadratenmethode in detail na (rekenregels of componentsgewijs). Denk na over de betekenis van elke grootheid.
- Construeer X en H. Verifieer numeriek dat H symmetrisch ($H^t = H$) en idempotent ($HH = H$) is en bereken de eigenwaarden.
- Gebruik gepaste matrixbewerkingen om $\hat{\boldsymbol{\beta}}$, $\hat{\boldsymbol{y}}$, $\boldsymbol{e}$, S^2 te berekenen. Zorg dat S^2 als getal en niet als matrix wordt opgeslagen.
- Bereken de variantie-covariantiematrices bij voorgaande resultaten.
- Gebruik regressie-equivariantie om aan te tonen dat regressie van de residuen in functie van de onafhankelijke veranderlijken resulteert in de nulvector en reken dit na in `R`.
- Verifieer schaal-equivariantie door `women` $\in [0, 100]$ te vervangen door `women`$/100 \in [0, 1]$ en het resultaat te vergelijken.
- Stel de gestandaardiseerde regressievergelijking op en vergelijk coëfficiënten β'_j met de originele β_j.
- Een toepassing van affiene equivariantie die later nog aan bod komt is regressie met principaalcomponenten. Verklaar `prestige` door de principaalcomponenten $Y = XP$ en bereken daaruit de eerder gevonden coëfficiëntenschatters β'_j met behulp van de rotatiematrix P.

7 Inferentie en variantie-analyse

Testen voor individuele parameters. Gelden de Gauss-Markov-voorwaarden, $\boldsymbol{\varepsilon} = N_n(0, \sigma^2 I_n)$, dan kan per parameter een t-test worden uitgevoerd om te testen of één specifieke regressiecoëfficiënt $\hat{\beta}_j$ significant afwijkt van een hypothetische waarde β_j dankzij de verdeling

$$\frac{\hat{\beta}_j - \beta_j}{s(\hat{\beta}_j)} \sim t_{n-p}.$$

Simultane uitspraken over meerdere parameters. Opdat de type I-fout bij gezamenlijke uitspraken over q verschillende parameters maximaal α zou blijven, kan voor de afzonderlijke testen significantieniveau α/q gebruikt worden: de Bonferroni-correctie. In plaats daarvan past het commando `p.adjust()` de p-waarde aan zodat verwerpingsgrens en betrouwbaarheid behouden blijven.

Deze methode is suboptimaal omdat er geen rekening wordt gehouden met de correlatie tussen alle parameters. Kwadratensommen laten toe om het model op een andere manier te bekijken.

Sommen van kwadraten. Uit de identiteit $(y_i - \overline{y}) = (y_i - \hat{y}_i) + (\hat{y}_i - \overline{y})$, kan een belangrijk verband tussen sommen van kwadraten worden afgeleid, namelijk

$$\begin{array}{ccccc} \sum_{i=1}^n (y_i - \overline{y})^2 & = & \sum_{i=1}^n (\hat{y}_i - \overline{y})^2 & + & \sum_{i=1}^n (y_i - \hat{y})^2 \\ \text{SST} & = & \text{SSR} & + & \text{SSE}, \end{array}$$

met daarin de volgende termen:

SST $= \sum_{i=1}^n (y_i - \overline{y})^2$, *total sum of squares*, enkel bepaald door de respons, maat voor de totale variabiliteit die moet worden verklaard door het regressiemodel;

SSR $= \sum_{i=1}^n (\hat{y}_i - \overline{y})^2$, *regression sum of squares*, het deel van de totale variabiliteit dat wordt verklaard door het regressiemodel, hangt af van de gekozen (transformaties van) onafhankelijke veranderlijken;

SSE $= \sum_{i=1}^n (y_i - \hat{y}_i)^2$, *error sum of squares*, het onverklaarde deel van de variabiliteit, hoe kleiner deze waarde, hoe dichter de respons bij het model aansluit.

Determinatiecoëfficiënt. Intuïtieve maten voor de kwaliteit van een model zijn dan de (aangepaste) determinatiecoëfficiënten,

$$\begin{aligned} R^2 &= 1 - \frac{SSE}{SST}, \\ R^2_{\text{adj}} &= 1 - \frac{SSE/(n-p)}{SST/(n-1)}. \end{aligned}$$

Extra sommen van kwadraten. Om modellen te vergelijken met een verschillend aantal veranderlijken maar zelfde totale som van kwadraten, wordt de extra som van kwadraten berekend

$$\text{SSR}(X_2|X_1) = \text{SSR}(X_1, X_2) - \text{SSR}(X_1).$$

Zodoende kan een meervoudig model stapsgewijs worden opgebouwd door telkens een term toe te voegen met een extra som van kwadraten als gevolg,

$$\text{SST} = \text{SSR}(X_1) + \text{SSR}(X_2|X_1) + \text{SSR}(X_3|X_1, X_2) + \ldots + \text{SSR}(X_p|X_1, \ldots, X_{p-1}) + \text{SSE}(X_1, \ldots, X_p).$$

Is `lm1` een meervoudig regressiemodel in `R`, dan worden deze termen samengevat in een ANOVA-tabel met het commando `anova(lm1)`.

Partiële F-test. Als de Gauss-Markov-voorwaarden geldig zijn, kan worden getest of de extra som van kwadraten $\mathrm{SSR}(X_{p-q+1}, \ldots, X_p | X_1, \ldots, X_{p-q}) = \mathrm{SSE}_{p-q} - \mathrm{SSE}_p$ door simultaan toevoegen van q veranderlijken significant verschilt van nul dankzij de statistiek

$$F = \frac{(\mathrm{SSE}_{p-q} - \mathrm{SSE}_p)/q}{\mathrm{SSE}_p/(n-p)} \sim F_{q,n-p}.$$

Dit levert een manier om simultane uitspraken te doen die wel rekening houdt met onderlinge correlaties tussen de parameters.

Zijn `lm1` en `lm2` twee lineaire modellen in `R`, dan geeft `anova(lm1,lm2)` een vergelijking tussen beide modellen.

Globale F-test. Analoog kan worden getest of alle veranderlijken samen een significant deel van de variabiliteit verklaren en dus of de verklaarde som van kwadraten SSR significant verschilt van nul,

$$F = \frac{\mathrm{SSR}_p/(p-1)}{\mathrm{SSE}_p/(n-p)} \sim F_{p-1,n-p}.$$

Dit is een speciaal geval van de partiële test hierboven want dit komt overeen met het vergelijken van het model met alle p veranderlijken met een model zonder verklarende veranderlijken maar enkel een intercept, aangezien voor dit laatste model de identiteiten $\mathrm{SSR}_1 = 0$ en $\mathrm{SSE}_1 = \mathrm{SST}$ gelden.

General linear hypothesis. De meest algemene lineaire hypothesen die over de coëfficiënten kunnen worden geformuleerd, zijn van de vorm

$$C\boldsymbol{\beta} = \boldsymbol{\beta}_0$$

met C een matrix van rang q. Hiermee worden dus eigenlijk q lineaire restricties op de p parameters $\beta_0, \ldots, \beta_{p-1}$ gelegd. Zodoende kan de extra som van kwadraten berekend worden van een model met $p - q$ lineair onafhankelijke parameters ten opzichte van het volledige model met p parameters en kan hierop een partiële F-test worden uitgevoerd.

In `R` kan zo een algemene lineaire hypothese uitgevoerd worden met het commando `glh.test(lm, C, beta0)` uit het `gmodels`-pakket.

Commando's in R. Het commando `lm()` berekent een regressiemodel, `summary()` vat de belangrijkste statistieken samen en `anova()` toont de ANOVA-tabel.

```
? lm
model.lm = lm(y~X1+X2)
model.lm$coefficients            # \hat{\vec{\beta}}
model.lm$residuals -> e          # \vec{e}
model.lm$rank                    # p
model.lm$fitted.values -> yhat   # \hat{\vec{y}}
model.lm$df.residual             # n-p
model.lm$model                   # gebruikte variabelen, enkel cases zonder NA

? summary.lm
summary(model.lm)$coefficients   # \hat{\vec{\beta}}, s, t en p
summary(model.lm)$sigma          # S
summary(model.lm)$r.squared      # R^2
summary(model.lm)$adj.r.squared  # R^2_adj
summary(model.lm)$fstatistic     # F

? anova
model.aov = anova(model.lm)      # ANOVA-tabel opvragen
attributes(model.aov)
anova(model.lm1,model.lm2)       # twee modellen vergelijken

library(gmodels)
glh.test(lm, C, beta0)
```

Prestige-dataset. Maak volgende modellen met $Y = \texttt{prestige}$, $X_1 = \texttt{education}$, $X_2 = \log_{10}(\texttt{income})$ en $X_3 = \texttt{women}$, waarbij het verschil tussen de laatste twee modellen enkel de volgorde van de termen is,

$$\begin{aligned}
\texttt{prestige.1:} \quad Y &= \beta_0 + \beta_1 \cdot X_1, \\
\texttt{prestige.2:} \quad Y &= \beta_0 + \beta_2 \cdot X_2, \\
\texttt{prestige.12:} \quad Y &= \beta_0 + \beta_1 \cdot X_1 + \beta_2 \cdot X_2, \\
\texttt{prestige.123:} \quad Y &= \beta_0 + \beta_1 \cdot X_1 + \beta_2 \cdot X_2 + \beta_3 \cdot X_3, \\
\texttt{prestige.321:} \quad Y &= \beta_0 + \beta_3 \cdot X_3 + \beta_2 \cdot X_2 + \beta_1 \cdot X_1.
\end{aligned}$$

Beantwoord volgende vragen, uitgaand van de (twijfelachtige) assumptie dat de modelveronderstellingen geldig zijn (onderzoek daarvan komt later aan bod):

- Reken de t-statistiek en p-waarde bij de coëfficiënt van X_2 na in model `prestige.123`.
- Test de hypothesen dat $H_0 : \beta_2 = 25$ en $H_1 : \beta_2 \neq 25$ in het model `prestige.2` en in het model `prestige.123`.
- Gebruik t-testen om de hypothese $H_0 : (\beta_1, \beta_3) = (4.5, 0.1)$ te testen op het 5%-significantieniveau.
- Bereken de (aangepaste) determinatiecoëfficiënten op basis van de gegevens uit de ANOVA-tabel van model `prestige.123`.
- Vergelijk de ANOVA-tabellen van bovenstaande modellen, verklaar en benoem elke kwadratensom (SST, SSR(X_1), SSE(X_1), SSR($X_2|X_1$), ...).
- Bereken de extra som van kwadraten SSR($X_2|X_1, X_3$) en test of deze significant is.
- Test of de extra som van kwadraten door het simultaan toevoegen van de termen X_3, X_3^2 en X_3^3 aan een model met enkel X_1 en X_2 significant verschilt van nul.
- Test opnieuw de hypothese $H_0 : (\beta_1, \beta_3) = (4.5, 0.1)$ in model `prestige.123` maar gebruik nu één F-test.
- Test de hypothese $H_0 : \beta_2 = 10 \cdot \beta_1$ versus $H_1 : \beta_2 \neq 10 \cdot \beta_1$ in het model `prestige.123`.

8 Categorische voorspellers

Terwijl partiële F-testen in het vorige deel enkel oplossing lijken te bieden in zeer specifieke situaties, wordt ANOVA dadelijk onontbeerlijk in het geval van categorische voorspellers. Telt zo een veranderlijke G immers k verschillende groepen $\{g_0, \ldots, g_{k-1}\}$, dan geeft dit aanleiding tot $k-1$ dummy veranderlijken

$$X_i = \begin{cases} 1 \text{ als } G = g_i \\ 0 \text{ anders} \end{cases} \quad \text{met } i = 1, \ldots, k-1.$$

De vraag of de veranderlijke G een significante bijdrage levert, resulteert dan in het simultaan testen of $k-1$ coëfficiënten gelijk zijn aan nul.

Geldigheid van uitspraken gebaseerd op regressie of ANOVA komt later aan bod, hieronder wordt gefocust op de interpretatie van verschillende soorten modellen.

One-way ANOVA. In het geval van een model met slechts één, categorische, veranderlijke spreekt men over one-way ANOVA. In plaats van met `lm()` wordt in deze situatie doorgaans met `aov()` gewerkt, dat specifieke methoden kent: `model.tables()` voor het berekenen van groepsgemiddelden en -effecten, de Levene-test (`leveneTest()`) voor homogeniteit van varianties en de *Tukey Honest Significant Differences*-testen (`TukeyHSD()`) voor paarsgewijs testen van gemiddelden.

- Bekijk de output van `summary()` en `lm()` voor het model `aov(prestige~type)` en interpreteer de cijfers.
- Ga na tussen welke groepen het gemiddelde verschilt.
- Bereken een 95%-betrouwbaarheidsinterval voor de gemiddelde prestige van een arbeider (`type==bc`) op de klassieke manier (`t.test()`) en via het lineaire model met `predict()`. Verklaar het verschil.

ANCOVA. In het geval van een model dat continue en categorische veranderlijken mengt, wordt soms gesproken over *Analyse van Covariantie*. In het meest algemene model komt de categorische veranderlijke voor als hoofd- én interactie-effect. Als dit laatste niet significant is, wordt het uit het model verwijderd. Hoofdeffecten worden in het algemeen niet verwijderd zolang zo een effect nog als factor in een interactieterm voorkomt. Hoewel de coëfficiënten in zo een model een duidelijke interpretatie hebben, zijn de bijhorende t-testen doorgaans ontoereikend om de statistische significantie van de relevante vragen na te gaan, gebruik daarvoor de partiële F-testen in de output van `anova()`.

- Ga na of de manier waarop `Education`, $\log_{10}$(`income`) en `women` de prestige van een beroepsgroep bepalen, afhangt van het type beroep. Voeg daartoe alle interactietermen met `type` toe aan het model `prestige` $= \beta_0 + \beta_1 \cdot \log_{10}$(`income`) $+ \beta_2 \cdot$ `education` $+ \beta_3 \cdot$ `women` om te bestuderen of er verschillen zijn tussen de verschillende beroepsgroepen. Bestudeer de output van `anova()` en vereenvoudig het model tot alle interactie-effecten significant zijn.
- Voer een partiële F-test uit om na te gaan of het bekomen ANCOVA-model significant meer verklaart dan het regressiemodel zonder de veranderlijke `type`. Hou er rekening mee dat de totale som van kwadraten hetzelfde moet zijn, wat bemoeilijkt wordt door ontbrekende waarden. Gebruik `na.omit()` om corresponderende observaties weg te laten.
- Schrijf de bekomen vergelijkingen uit en formuleer de belangrijkste conclusies die op basis van `summary()` te trekken zijn.
- Gebruik een algemene lineaire hypothesetest om na te gaan of het inkomenseffect verschilt tussen bedienden en *professionals*.

Lack-of-fit-test. Als een in wezen numerieke veranderlijke slechts enkele waarden kent, is het niet steeds duidelijk of deze veranderlijke beter als numeriek dan wel als categorisch wordt beschouwd. In dat geval kan mits de omzetting `as.factor()` een one-way-anova model worden gemaakt en vergeleken met het regressiemodel. Het verschil in som van kwadraten `anova()` tussen de voorspellingen van het regressiemodel `lm()` en de afzonderlijke schattingen voor het gemiddelde `aov()` kan worden vergeleken met een partiële F-test, de zogenaamde lack-of-fit-test.

- Gebruik de gegevens `case0802` uit het `Sleuth3`-pakket om het regressiemodel `Time` $= \beta_0 + \beta_1 \cdot$ `Voltage` te maken: de tijd waarna de isolerende eigenschappen van een vloeistof verdwijnen onder zeven bepaalde voltages.
- Beschouw de veranderlijke `Voltage` nu als een categorische veranderlijke en benader het probleem met one-way ANOVA.
- Stel beide modellen grafisch voor en vergelijk ze via de partiële F-test. Welk van beide is te verkiezen? Bestudeer in beide gevallen de modelveronderstellingen.
- Ga na dat $\log_{10}($`Time`$) = \beta_0 + \beta_1 \cdot \log_{10}($`Voltage`$)$ veel betere modellen levert. Voer opnieuw de lack-of-fit-test uit en selecteer het beste model.

Two-way ANOVA. Two-way ANOVA combineert principes van ANOVA en multipele regressie in het specifieke geval van twee categorische predictoren A en B met een continue respons Y. Neem aan dat de eerste categorische veranderlijke m mogelijke uitkomsten kent, de tweede n en dat er per combinatie van uitkomsten een of meerdere metingen voor de respons zijn. Dan kunnen de resultaten worden voorgesteld in een $m \times n$-tabel met in elke cel het gemiddelde van de metingen bij die combinatie van voorspellers. Er zijn twee modellen mogelijk:

Multiplicatief model. Het gemiddelde $\mathrm{E}(Y \mid A = a \wedge B = b)$ wordt in elk van $m \cdot n$ groepen afzonderlijk geschat, dit wordt ook wel het volledige model genoemd.

Additief model. Voor elke uitkomst $A = a$ of $B = b$ wordt afzonderlijk een effect $\mathrm{E}(Y \mid A = a) - \mathrm{E}(Y)$ of $\mathrm{E}(Y \mid B = b) - \mathrm{E}(Y)$ geschat ten opzichte van de schatting voor het globale gemiddelde $\mathrm{E}(Y)$. De $m + n$ schattingen voor $E(Y|A = a \wedge B = b)$ worden dan bekomen als som van het globale gemiddelde en elk van beide effecten.

De dataset `ex1320` uit het pakket `Sleuth3` bevat scores voor een wiskunde-test uit 1989 bij laatstejaarsstudenten met weinig (`a`), gemiddeld (`b`) of veel (`c`) voorkennis (`Background`). Onderzoek of de score voor deze test, rekening houdend met de voorkennis, verschilt per geslacht en of dat eventuele verschil varieert naargelang de voorkennis.

- Maak het additieve en multiplicatieve model en interpreteer de resultaten.
- Wat leer je meer uit deze modellen dan uit een vergelijkende t-test voor de gemidelde scores voor mannen en vrouwen? Hoe kan je onderzoeken of de voorkennis van mannen en vrouwen verschilt? Formuleer globale conclusies.

9 Selectie van veranderlijken

Stapsgewijze regressie. De methode van achterwaartse regressie die werd uitgelegd bij het vak Statistiek is erg naïef: t-testen meten in wezen niet wat de bijdrage is van een veranderlijke in een model en ze zijn bovendien helemaal niet geschikt voor het beoordelen van categorische veranderlijken. De determinatiecoëfficiënt R^2 en zelfs R^2_{adj} geven de voorkeur aan modellen met te veel veranderlijken (overfitten). Ook de ANOVA-tabel biedt geen oplossing omdat modellen met eenzelfde aantal parameters daarin niet vergeleken worden.

Er moeten dus andere maten worden gebruikt, zoals C_p, AIC en BIC, die kijken naar een geschaalde som van kwadraten van residuen, verhoogd met een term die oploopt met het aantal veranderlijken. Vertrekkend van een welbepaald model, berekent stapsgewijze regressie één van deze criteria voor alle modellen die één term verschillen van dit model, om zo op zoek te gaan naar een model met een extremale waarde voor het betreffende criterium. Dit is geïmplementeerd in `stepAIC()` uit de `MASS`-bibliotheek, dat standaard het AIC-criterium gebruikt en als uitvoer het resulterende `lm`-model geeft. Alternatief kan het BIC criterium worden gebruikt door de optie `k=log(n)` toe te voegen en met de optie `list(lower=..., upper=...)` kunnen een minimaal en maximaal model worden meegegeven.

- Gebruik stapsgewijze regressie om te bepalen welke veranderlijken zijn aangewezen om de prestige van een beroepsgroep te verklaren (dataset `Prestige` in bibliotheek `car`): start van het interactiemodel met `type`.
- Zoek het model met de beste AIC-waarde waarin de interactieterm tussen `type` en `education` voorkomt en vergelijk met het voorgaande.
- Doe hetzelfde voor het aantal calorieën in een portie ontbijtgranen (dataset `UScereal`, bibliotheek `MASS`). Sluit de veranderlijke `mfr` uit.

Alle submodellen vergelijken. Stapsgewijze regressie vergelijkt steeds een beperkt aantal modellen, terwijl het met voldoende rekenkracht eigenlijk geen probleem is om alle mogelijk modellen naast elkaar te zetten. Het commando `regsubsets()` uit de bibliotheek `leaps` berekent voor een oplopend aantal coëfficiënten telkens het model met de beste AIC-waarde. De uitvoer van `summary()` toont welke veranderlijken aanwezig zijn in het beste model met oplopend aantal veranderlijken, de feitelijke criteria bij elk model zijn te vinden als attributen.

- Bereken de BIC-waarde voor de beste submodellen van `prestige~(education+income+women)*type` met een oplopend aantal coëfficiënten: met de optie `nvmax=n` kunnen modellen met meer termen dan standaard worden gemaakt. Teken de (p, BIC)-grafiek en ga na wanneer de BIC-waarde minimaal is. Hoeveel termen schrijft deze methode voor?
- Maak voor een oplopend aantal termen telkens de beste 5 (met de optie `nbest=5`) voor het verklaren van het aantal calorieën in de dataset `UScereal`. Maak opnieuw de (p, BIC)-grafiek en lees af welk model het beste is.

10 Problemen en remedies

1. Afwijking van lineariteit
   - Probleem: de vorm van het model is fout, data voldoen niet aan het vooropgestelde verband.
   - Gevolg: het hele model en dus ook alle schattingen en voorspellingen zijn fout.
   - Opsporen: residuen moeten gemiddeld constant nul zijn in residuplots (13.7).
   - Oplossing: transformatie aflezen van de puntenplots (16.1-16.3)?
2. Heteroscedasticiteit
   - Probleem: de residuen hebben geen constante variantie in functie van de respons.
   - Gevolg: schattingen onvertekend, testen en betrouwbaarheidsintervallen onbetrouwbaar.
   - Opsporen: residuen moeten constante spreiding hebben in residuplots (13.7).
   - Oplossing: transformaties of gewogen regressie (16.4)?
3. Afwijking van normaliteit
   - Probleem: de residuen zijn niet normaal verdeeld.
   - Gevolg: betrouwbaarheidsintervallen robuust in grotere datasets, voorspellingsintervallen niet.
   - Opsporen: normale probabiliteitsplot van de residuen, normaliteitstests (13.7).
   - Oplossing: transformaties (16.1-16.3)?
4. Clustering
   - Probleem: gegevens behoren tot verschillende groepen.
   - Gevolg: onafhankelijkheid van metingen/residuen is geschaad als deze afhangen van de groep.
   - Opsporen: aanwezigheid van groepen vaststellen in residuplots of experimental design.
   - Oplossing: gepaste categorische predictoren zoeken (9, 10, 14)?
5. Multicollineariteit
   - Probleem: regressoren zijn onderling afhankelijk.
   - Gevolg: Modelveronderstellingen voldaan maar berekeningen onstabiel: grote variantie.
   - Opsporen: correlatie-analyse (twee aan twee, 18.1) en variantie-inflatie-factor (18.2).
   - Oplossing: principaalcomponenten, ridge regression (18.3)?
6. Outliers
   - Probleem: één of meerdere waarden horen niet thuis in de dataset.
   - Gevolg: schattingen ten onrechte beïnvloeden en/of variantie opdrijven.
   - Opsporen: residuen, Mahalanobis- en Cook's afstand,... (19).
   - Oplossing: betreffende meetwaarden onderzoeken, rapporteren en in extremis uitsluiten?
7. Autocorrelatie
   - Probleem: opeenvolgende metingen kunnen elkaar beïnvloeden.
   - Gevolg: onafhankelijkheid van metingen geschaad: kansuitspraken zijn onbetrouwbaar.
   - Opsporen: residuen versus tijd/case-nummer plotten.
   - Oplossing: tijdreeksanalyse (buiten bestek van de cursus)?

Gestandaardiseerde residuen. Normaliteit van de residuen wordt best nagegaan op basis van de gestandaardiseerde residuen,

$$e_i^{(s)} = \frac{e_i}{s\sqrt{1-h_{ii}}},$$

die kunnen worden berekend met het commando `rstandard()`. Deze gestandaardiseerde residuen geven eveneens een eerste indicatie voor eventuele uitschieters.

```
> e = model$residuals      # residuen
> s = summary(model)$sigma # standaarddeviatie van de residuen
> es = rstandard(model)    # gestandaardiseerde residuen
```

Diagnostische grafieken. Het commando `plot()` toegepast op een regressiemodel (type `lm`) genereert vier grafieken:

1. Residuen in functie van schattingen samen met een gladde fit-curve door de *gemiddelde* residuen: zijn de residuen overal gemiddeld nul, $E(\varepsilon_i) = 0$, dan valt deze curve samen met de horizontale as;
2. Normale kwantielplot van de gestandaardiseerde residuen: zijn de residuen normaal verdeeld, dan volgen de punten de regressielijn;
3. Wortel van de absolute waarde van de gestandaardiseerde residuen in functie van de schattingen, opnieuw met een gladde fit-curve: is de variantie constant, $\text{Var}(\varepsilon_i) = \sigma^2$, dan is dit een horizontale rechte;
4. De vierde grafiek toont leverage en Cook's afstand van elk punt in het model en is nuttig om invloedrijke punten te vinden (hoofdstuk robuste regressie).

Het commando `par()` laat toe om deze vier grafieken tegelijk zichtbaar te maken. Bijkomend kan de invloed van individuele regressoren worden nagegaan met de partiële residuplots $(x_i, e_i + \hat{\beta}_i \cdot x_i)$ in de output van `termplot()`.

```
> par(mfrow=c(2,2))  # grafieken in twee rijen en twee kolommen zetten
> plot(model.lm)
> par(mfrow=c(1,1))  # grafiek opnieuw schermvullend
> termplot(model.lm, partial.resid = TRUE)
```

Transformaties. Dezelfde mogelijkheden als eerder (sectie **??**) blijven geldig voor het transformeren van veranderlijken.

Normaliteit van de respons (of regressoren) is weliswaar geen voorwaarde voor regressie, daarom wordt in deze context de machtstransformatie gezocht die de residuen normaliseert. Het commando `powerTransform()` uit het pakket `car` werkt daarom ook op objecten van het type `lm`.

Transformeren van proporties kan op verschillende manieren: de arcsin($\sqrt{\ }$)-transformatie stabiliseert de variantie in regressiemodellen (pagina 108) terwijl de logit-transformatie resulteert in een interpreteerbare maat (zie hoofdstuk logistische regressie).

- Ga na welke transformaties aangewezen zijn in het model `prestige` $\sim \log_{10}$(`income`)$+$`education`$+$ `women`. Bestudeer het effect op de diagnostische plots.
- Doe hetzelfde voor `Time` $\sim$ `Voltage`.
- Maak een model dat de proportie vrouwen in een beroepsgroep verklaart in functie van de studieduur. Bestudeer de modelveronderstellingen en vergelijk verschillende transformaties van de respons.

Gewogen regressie. Als een model $\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ aanleiding geeft tot een trend $\sigma_i^2 = \sigma^2/w_i$ in de variantie van de residuen, kan heteroscedasticiteit worden verdreven door het wegen van de steekproefelementen,

$$\sqrt{w_i} \cdot y_i = \sqrt{w_i} \cdot \beta_0 + \sqrt{w_i} \cdot \beta_1 \cdot x_{i1} + \ldots + \sqrt{w_i} \cdot \beta_{p-1} \cdot x_{i,p-1} + \sqrt{w_i} \cdot \varepsilon_i.$$

Voor dit nieuwe model $\boldsymbol{y}^{(W)} = X^{(W)}\boldsymbol{\beta} + \boldsymbol{\varepsilon}^{(W)}$ is de variantie van de residuen dan homogeen σ^2. Let er op dat een model met gewichten aanleiding geeft tot gewogen residuen

$$e_i^{(w)} = y_i^{(w)} - \hat{y}_i^{(w)} = \sqrt{w_i} \cdot (y_i - \hat{y}_i) = \sqrt{w_i} \cdot e_i.$$

Een model `lm(y~x,weights=w)` in `R` geeft als residuen de grootheden e_i terug, die dus nog moeten worden gewogen om na te gaan of de heteroscedasticiteit inderdaad is verdwenen.

- De dataset `strongx` uit het pakket `faraway` bevat gegevens van een experimentele studie naar de interactie tussen bepaalde elementaire deeltjes. De veranderlijke `crossx` zou lineair afhangen van de veranderlijke `energy` en werd bij verschillende impulsen herhaaldelijk gemeten, zodat bij elke waarde de standaarddeviatie (`sd`) gekend is. Maak het gewone en het gewogen regressiemodel. Vergelijk de modelveronderstellingen.
- Stel beide modellen grafisch voor op de puntenplot. Gebruik de optie `cex` zo dat de grootte van elk meetpunt evenredig is met het respectievelijke gewicht.

In de praktijk is er vaak wel heteroscedasticiteit maar zijn standaardfouten niet gekend. Het is dan soms mogelijk de variantie te schatten en te verdrijven door het maken van drie opeenvolgende regressiemodellen:

1. Voer gewone lineaire regressie `lm(y~x)` uit.
2. Om de gewichten w_i te schatten, wordt een tweede regressie `lm(abs(e)~yhat)` uitgevoerd van de absolute waarde van de residuen $|e_i|$ op de schattingen voor de respons $\hat{y}_i$.
3. De voorspellingen $|\hat{e}_i|$ uit dit tweede model worden dan gebruikt als schattingen voor de variantie $\hat{\sigma}_i^2 = s_i^2$ en geven aanleiding tot gewichten $w_i = 1/s_i^2 = 1/\hat{e}_i^2$ voor een derde regressiemodel met het commando `lm(y~x,weights=w)`.

Werk voor de aanschouwelijkheid met het eenvoudige regressiemodel `dist` $= \beta_0 + \beta_1 \cdot$ `speed` dat de remafstand in functie van de snelheid beschrijft. De gegevens zijn in te laden met `attach(cars)`.

- Maak het model en controleer de modelveronderstellingen. Er is een lichte vorm van heteroscedasticiteit zichtbaar op de residuplot.
- Bereken het model `e.lm = lm(abs(e)~yhat)`, ga na of het significant is en teken de regressierechte op de puntenplot $(\hat{y}_i, |e_i|)$.
- Gebruik vervolgens gewogen regressie, stel beide modellen samen grafisch voor en teken telkens 95%-predictie-banden. Visualiseer opnieuw de gewichten in de puntgrootte van de observaties. Ga opnieuw de modelveronderstellingen na, in het bijzonder of de gewogen residuen $e_i^{(w)}$ nog steeds heteroscedasticiteit suggereren.
- Verifieer tot slot dat de heteroscedasticiteit is verdwenen door de gewogen residuen opnieuw te regresseren in functie van de schattingen voor de respons, `e2.lm = lm(abs(e2)~yhat2)`.

11 Ridge & robust regression

Multicollineariteit constateren. Behalve over- of onderfitten is een groot probleem voor het selecteren van geschikte regressoren de correlatie tussen de veranderlijken onderling. Zijn deze immers sterk gecorreleerd, dan is het gevonden model sterk gegevensafhankelijk, zijn belangrijke termen mogelijk niet significant en is het moeilijk het model te interpreteren, omdat het gevonden effect misschien een gemeenschappelijke, onderliggende oorzaak heeft. Een eerste diagnostiek voor het detecteren van multicollineariteit is expliciet de correlatie tussen de regressoren berekenen, maar dit meet natuurlijk enkel of veranderlijke twee aan twee afhankelijk zijn.

Een methode om te ontdekken of een veranderlijke lineair afhankelijk is van verschillende andere wordt gegeven door de *variance inflation factors* (VIF),

$$\mathrm{VIF}_j = \frac{1}{1 - R_j^2} = (R_{XX}^{-1})_{jj}.$$

Hierin is R_j^2 de determinatiecoëfficiënt van het model dat de veranderlijke X_j verklaart in functie van de andere regressoren en R_X de correlatiematrix van X. Idealiter is deze statistiek voor elke regressor gelijk aan één, maar is het resultaat van deze veranderlijke voor een veranderlijken groter dan tien, of is de gemiddelde VIF in een model substantieel groter dan één, dan is er sprake van multicollineariteit.

Remedies tegen multicollineariteit. Als de indicatoren er op wijzen dat multicollineariteit een probleem is, is de meest voor de hand liggende oplossing om één of meerdere veranderlijken uit het model weg te laten. Als dit de determinatiecoëfficiënt niet al te zeer beïnvloedt en er hierdoor geen interessante termen verloren gaan, is dat de meest eenvoudige en effectieve methode.

- Bereken deze statistieken VIF_j voor de veranderlijken uit het model

  `lm(prestige~education+log10(income)+logit(women))`

  met het commando `vif()`.

- De dataset `case0902` uit het pakket `Sleuth3` bevat gegevens over kenmerken van verschillende diersoorten waaronder de grootte van de worp. Ga na dat er multicollineariteit is in het model

  `lm(Litter~Gestation+log10(Body)+log10(Brain))`,

  los deze op en bestudeer het effect van `Body` op `Litter`.

Veranderlijken centreren. In het geval van polynomiale of interactietermen zal er automatisch een hoge correlatie in het model optreden. Die kan vaak worden verdreven door de veranderlijken te centreren. Gebruik voor het vervolg onderstaande random gegevens.

```
> x1 = rnorm(20,5)
> x2 = rnorm(20,5)
> x3 = rnorm(20,mean=x1,sd=.01)
> y1 = rnorm(20,mean=3+x1+x1**2,5)
> y2 = rnorm(20,mean=3+x1+x3)
```

- Ga na dat de VIF-waarden van het model $y_1 = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_1^2$ inderdaad wijzen op multicollineariteit, maar dat deze verdwijnt in het model $y_1 = \beta_0 + \beta_1 \cdot (x_1 - \overline{x_1}) + \beta_2 \cdot (x_1 - \overline{x_1})^2$.

- Ga na wat er gebeurt met de VIF-waarden als de veranderlijken in het model $y_1 = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \beta_3 \cdot x_1 \cdot x_2$ worden gecentreerd. Doe hetzelfde met het model $y_1 = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_3 + \beta_3 \cdot x_1 \cdot x_3$.

Ridge regression. Als geen van bovenstaande oplossingen werkt, vormt ridge regression een mogelijke oplossing. Multicollineariteit leidt tot hoge varianties op de schatters en kan er toe leiden dat deze een verkeerde grootte-orde of zelfs een verkeerd teken hebben, wat interpretatie van die termen onmogelijk maakt. Ridge regression start vanaf het gestandaardiseerde model en introduceert een bias op de regressiecoëfficiënten die de variantie van de coëfficiëntenschatters doet afnemen. Door de bias zijn betrouwbaarheids- en predictie-intervallen niet meer betekenisvol, maar nemen de schatters vaak wel het juiste teken en grootte-orde aan, waardoor ze kunnen worden geïnterpreteerd.

- Bereken de gewone kleinste kwadratenschatters voor $y_2 = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_3$ met bovenstaande gegevens, vergelijk de parameterschattingen met de verwachtingen, bekijk de varianties en ga na dat er inderdaad sprake is van multicollineariteit.
- Gebruik het commando `lm.ridge()` uit de `MASS`-bibliotheek om ridge regression toe te passen. Ga na dat het resultaat identiek is met voorgaande indien het commando met optie `lambda=0` wordt uitgevoerd.
- Bereken 10 verschillende modellen door toevoeging van de parameter `lambda=seq(0,1,length=10)` en interpreteer de output. De methode `plot()` toegepast op het `lm.ridge`-object toont automatisch de ridge trace: lees een geschikte parameter af en interpreteer het model.
- Bestudeer het model voor `litter` met ridge regression.

Outlier-onderzoek. Een methode is robuust als ze niet overmatig beïnvloed wordt door een of enkele uitschietende waarden. De gewone kleinste kwadratenmethode is niet robuust, omdat punten met een groot residu of grote leverage de coëfficiënten overmatig kunnen beïnvloeden.

Een eerste probleem is een robuuste manier om uitschietende waarden te herkennen, omdat het berekenen van de regressievergelijking en dus de residuen zelf niet robuust is. Daarom wordt bij onderzoek naar uitschieters eerder gekeken naar de *studentized residuals*: het (gestandaardiseerde) residu van een punt in het model dat gemaakt is zonder dat specifieke punt.

Niet enkel punten met een groot residu maar ook meetwaarden die zich van de rest van de puntenwolk distantiëren, kunnen een model bovenmatig beïnvloeden. Om deze punten te identificeren wordt de robuuste Mahalanobisafstand berekend, op basis van robuuste MCD-schattingen van gemiddelde en covariantiematrix.

De robuuste Mahalanobisafstand geeft een maat van hoe ver een punt zich in het argumenthypervlak van het centrum is verwijderd, de studentized residuals meten hoe ver de punten vertikaal van het regressiehypervlak liggen. Samen leveren deze dus een tweedimensionale diagnostische plot van de puntenwolk waarop uitschieters makkelijk te herkennen zijn.

Een robuuste manier om parameterschattingen te maken is de *Least Trimmed Squares*-methode (LTS) die de grootste residuen buiten beschouwing laat bij het minimaliseren van de kwadratensom. Deze methode is geïmplementeerd in het commando `lstReg()` uit het pakket `robustbase`, dat voor de rest zeer analoog werkt als `lm()`.

```
> library(robustbase)
> rstudent(model)                # studentized residuals
> Mr = covMcd(X)$center          # minimum covariance determinant
> Cr = covMcd(X)$cov             # minimum covariance determinant
> mahalanobis(X, Mr, Cr)         # robuuste kwadratische Mahalanobisafstand
> ltsReg(y~.,data=...)           # least trimmed squares
```

- Onderzoek bovenstaande regressiemodellen voor `prestige` en `litter` op de aanwezigheid van outliers door het maken van de diagnostische plot en vergelijk het OLS- met het LTS-model.

12 Logistische regressie

General linear models. De kleinste kwadratenmethode zoals die tot nu toe werd gebruikt, resulteert in de maximum likelihoodschatters voor de coëfficiënten van een lineair regressiemodel in de veronderstelling dat de residuen normaal verdeeld zijn. Aan deze voorwaarde kan in veel gevallen worden voldaan door transformatie van regressoren of respons, tenminste voor zover deze laatste een continue veranderlijke is. Is de respons echter een binaire veranderlijke, dan is deze berekening niet houdbaar, verliest de respons alle betekenis buiten het interval $[0, 1]$, heeft de variantie van de residuen geen constante variantie en zo voort. Daarom vereist regressie in dit geval een andere aanpak.

Binaire respons. Is de respons Bernouilli verdeeld, dan zijn de enige mogelijke waarden voor y_i de 0 en de 1. De gemiddelde respons bij gegeven $\boldsymbol{x}_i$ is dan een proportie $\pi_i = P(Y = 1|\boldsymbol{X} = \boldsymbol{x}_i)$, de kans op succes binnen de groep $\boldsymbol{X} = \boldsymbol{x}_i$. Om er voor te zorgen dat een regressiemodel resulteert in een proportie $\pi_i \in [0, 1]$, wordt de logit-transformatie $\text{logit}(\pi_i)$ gebruikt. Dit is de zogenaamde *log-odd*, het logaritme van de verhouding van het aantal successen op het aantal falingen. Het regressiemodel wordt dus van de vorm

$$\text{logit}(\pi_i) = \log\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + \sum_{j=1}^{p-1} \beta_i x_{i,j} = \boldsymbol{x}_i^t \boldsymbol{\beta}.$$

De betekenis van de parameters β_j hierin is dat bij toename van de j-e veranderlijke de *odds-ratio* gelijk is aan $\exp(\beta_j)$, of dat dus de odds veranderen met deze factor. De proportie π_i kan berekend worden als

$$\pi_i = \frac{1}{1 + \exp(-\boldsymbol{x}_i^t \boldsymbol{\beta})}.$$

Een logistisch model in `R` wordt berekend met het commando `glm(y~x,family=binomial)` (*general linear model*). Los van deze transformatie en de andere interpretatie van de coëfficiënten, wordt ook een andere kansverdeling gebruikt voor het schatten van de parameters. Hierdoor is de klassieke t-test voor coëfficiënten niet meer geldig maar wordt de zogenaamde Wald-test gebruikt, gebaseerd op een normaal verdeelde teststatistiek,

$$\frac{\hat{\beta}_j - \beta_j}{s(\beta_j)} \approx N(0, 1).$$

Omdat de kwadraten van de residuen hun betekenis verliezen, zijn ook de F-tests niet meer geldig en is er als corresponderende maat de deviantie waarop een zogenaamde likelihood ratio test kan worden uitgevoerd. Het `anova()`-commando schakelt bij general linear models automatisch over op de devianties maar de likelihood ratio test moet manueel uitgevoerd worden,

$$\text{DEV}(H_1) - \text{DEV}(H_0) \sim \chi^2_q,$$

waarbij q het verschil in aantal vrijheidsgraden tussen beide modellen is. Om na te gaan of de vorm van een logistisch model met $n - p$ vrijheidsgraden voldoet (*goodness of fit*), kan de modeldeviantie $\text{DEV}(H_1) =$ `residual deviance` vergeleken worden met die van het verzadigde model $\text{DEV}(H_0) = 0$. Om na te gaan of de invloed van $p-1$ regressoren significant is, kan de modeldeviantie $\text{DEV}(H_0) =$ `Residual Deviance` vergeleken worden met een model dat enkel een intercept bevat, $\text{DEV}(H_1) =$ `Null Deviance`.

De variantie-covariantiematrix van de coëfficiënten $\hat{\Sigma}(\hat{\boldsymbol{\beta}})$ kan aan het `summary.glm`-object worden onttrokken als het `$cov.unscaled`-attribuut. Hiermee kan een betrouwbaarheidsinterval $\hat{\eta}_0 \pm z_{\alpha/2} s(\hat{\eta}_0)$ voor de respons worden geconstrueerd, met namelijk

$$s^2(\hat{\eta}_0) = \boldsymbol{x}_0^t \hat{\Sigma}(\hat{\boldsymbol{\beta}}) \boldsymbol{x}_0.$$

Space Shuttle. Na de ramp met de Challenger wordt de koude lanceertemperatuur als mogelijke schuldige naar voor geschoven. Bij eerdere lanceringen van Space Shuttles waren ook al problemen met de brandstofafsluiting vastgesteld. De gegevens `ex2011` uit het pakket `Sleuth3` bevatten de lanceertemperatuur (Fahrenheit) en het eventueel falen van de afsluitringen.

- Verklaart de temperatuur inderdaad eventuele defecten? Voer de goodness-of-fit test uit. Test of bijdrage van de temperatuur significant is: vergelijk de Wald-test en de χ^2-test voor de devianties.

- Interpreteer de coëfficiënten en stel het model grafisch voor samen met betrouwbaarheidsbanden. Bij welke temperatuur is de voorspelde kans op falen gelijk aan 50%?
- Wat is de kans op een defect bij 31 graden Fahrenheit, de lanceertemperatuur bij de ontploffing van de Challenger? Waarom dient deze uitkomst met de nodige omzichtigheid te worden behandeld?

Donner Party. De gegevens in het volgende voorbeeld (`case2001`) gaan over een reisgezelschap van 45 volwassenen dat in 1846 een lange en gevaarlijke doorreis door Amerika maakte: slechts 20 reizigers overleefden de ontberingen. Er wordt onderzocht of de overlevingskans afhankelijk is van leeftijd en geslacht.

- Maak een logistisch model dat de overlevingskans beschrijft in functie van leeftijd en geslacht. Schrijf het model in de vorm $\pi_{\male} = \ldots$ en $\pi_{\female} = \ldots$ en teken de grafieken.
- Formuleer correcte kansuitspraken bij elke gevonden coëfficiënt. Geef een 95%-betrouwbaarheidsinterval voor de man/vrouw odds-ratio voor overleven.
- Voor welke leeftijden hebben mannen resp. vrouwen minstens 50% overlevingskans?
- Ga na of de invloed van de leeftijd bij mannen en vrouwen verschilt. Interpreteer hoe dan ook de extra coëfficiënt, schrijf opnieuw het model uit en vergelijk grafisch met het eerste model.

Binomiale respons. In het geval de respons niet enkel succes of faling is, maar het aantal successen Y_i gegeven het aantal experimenten m_i, kan op vergelijkbare manier logistische regressie worden toegepast op de schattingen $\overline{\pi}_i = Y_i/m_i$ voor de kans op succes. Omdat een schatting op basis van een groter aantal pogingen nauwkeuriger is, wordt in het geval van een binomiale respons elke case gewogen. De syntax `family=binomial` verraadt dat het geval van de binaire respons hierboven eigenlijk een triviaal geval is voor dat van binomiale respons, met telkens 1 succes of 1 faling en gewicht 1 voor elke case.

```
> Krunnit = case2101; attach(Krunnit)
> summary(Krunnit.glm)
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.19620    0.11845 -10.099  < 2e-16 ***
log(Area)   -0.29710    0.05485  -5.416 6.08e-08 ***
    Null deviance: 45.338  on 17  degrees of freedom
Residual deviance: 12.062  on 16  degrees of freedom
AIC: 75.394
```

Poissonrespons. In het geval van een Poisson respons is er geen bovenlimiet op het aantal mogelijke successen, is er bijgevolg geen sprake van proporties en vervalt de nood voor een `logit`-transformatie. De uitkomsten bij een Poissonverdeling is voor kleinere aantallen wel erg scheef, waardoor een logaritmische transformatie aangewezen is. Verder is het enige verschil met voorgaande gevallen dat de Poissondichtheid gebruikt wordt in de likelihoodberekeningen: Wald test, devianties en likelihood ratio test blijven gelden.

```
> elephant = case2201; attach(elephant)
> elephant.glm = glm(Matings~Age,family=poisson)
> summary(elephant.glm)
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.58201    0.54462  -2.905  0.00368 **
Age          0.06869    0.01375   4.997 5.81e-07 ***
    Null deviance: 75.372  on 40  degrees of freedom
Residual deviance: 51.012  on 39  degrees of freedom
AIC: 156.46
```