Het gebruik van R m.b.t. **Chapter 17** uit 'The Analysis of Biological Data'
Lineaire regressie

Huiswerk
Het huiswerk voor het R practicum in week 10 is het doornemen van het Uitgewerkte voorbeeld. Het doornemen daarvan is de noodzakelijke voorbereiding om de opgaven te kunnen maken die je tijdens het practicum krijgt.
In het uitgewerkte voorbeeld komen geen Opdrachten voor die je moet maken: het script en de output staan in de tekst. Er komen wel Vragen voor die je moet maken. **Aan het begin van het R practicum in week 10 zullen de assistenten je antwoorden op de vragen willen zien als 'bewijs' dat je het uitgewerkte voorbeeld hebt doorgenomen.** Kun je geen antwoorden laten zien dan loop je het risico een 'nav' te krijgen voor het R onderdeel van deze cursus.

Wat ga je leren?
In dit hoofdstuk leer je eerst hoe je in R een **regressiemodel** (Y'=a+bX) kunt opstellen en welke twee aannames je moet controleren voordat je dat model mag opstellen. Daarna leer je hoe je het model kunt toetsen. Ook daarvoor moet je weer twee aannames controleren, en dat doe je aan de hand van een **residual plot**, waarin de verschillen tussen de gemeten Y-waardes en de door het model voorspelde Y-waardes staan uitgezet tegen de X-waardes (zie boek p. 559, Detecting non-normality and unequal variance).
Bij het toetsen van het model worden intercept (a) en regressiecoëfficiënt (b) getoetst met een t-toets (handmatige uitvoering: zie boek p. 552, The t-test of regression slope) en wordt het model als geheel getoetst wordt met een ANOVA (handmatige uitvoering: zie boek p. 554, The ANOVA approach).
Tot slot leer je hoe je in R betrouwbaarheidsintervallen kunt opstellen van intercept (a) en regressiecoëfficiënt (b).

Uitgewerkt voorbeeld.
Het uitvoeren van een correlatieanalyse wordt voorgedaan aan de hand van het bestand kabeljauw.xlsx. Het script en de output staan in de tekst, dus je kunt het voorbeeld doornemen zonder dat je RStudio hoeft te openen. Toch is het zinnig om op zijn minst het bestand even te hebben bekeken, dus als je je laptop bij de hand hebt, voer dan **één** van de volgende opdrachten uit.

Opdracht Ga naar Blackboard, knop Computerpractica R en download file kabeljauw.xlsx die je vindt onder R practicum – week 10.
Open kabeljauw.xlsx in Excel en bestudeer de opbouw en inhoud van de file.
Lees daarna verder bij **Achtergrond**.

Opdracht Ga naar Blackboard, knop Computerpractica R en download file kabeljauw.xlsx die je vindt onder R practicum – week 10.
Open RStudio. Maak alle vensters leeg.
Importeer het bestand kabeljauw.xlsx en geef het de naam **cod** (de Engelse naam van kabeljauw). Het bestand bevat geen *missing values*.
Open een nieuw script en run onderstaande code.

```
1 cod=as.data.frame(cod)
2
```

Je kunt in dit script verder werken en er de scriptregels aan toevoegen die hierna bij de bespreking van het voorbeeld in de tekst genoemd worden (bv. m.b.v. Copy/Paste). Op die manier kun je het uitgewerkte voorbeeld zelf mee-maken.

Achtergrond.

De kabeljauwpopulatie in de Baltische zee is halverwege de jaren tachtig ingestort. Zelfs na het inperken van de kabeljauwvisserij herstelde de populatie zich niet. Wetenschappers gingen op zoek naar een verklaring voor het uitblijven van herstel.
De onderzoekshypothese was dat de Baltische zee niet (meer) genoeg voedsel bevat voor volwassen kabeljauwen waardoor vrouwtjeskabeljauwen ondervoed raken en minder eitjes kunnen produceren. Dit laatste zou dan komen omdat de hoeveelheid eieren die een kabeljauw kan produceren afhankelijk is van de vetvoorraden rond de voortplantingsorganen. Om het effect van te weinig voedsel (en dus lage vetvoorraden) op de productie van eitjes te onderzoeken, werd in een aantal kabeljauwen zowel de vetvoorraad rond de voortplantingsorganen als het uiteindelijke aantal geproduceerde eitjes bepaald. Van 21 van deze kabeljauwen staan de metingen in bestand kabeljauw.xlsx (dataframe **cod**). Om didactische redenen zijn een aantal metingen aangepast.

In de nu komende paragrafen ga je de lineaire regressievergelijking opstellen (het 'regressiemodel') die de hoeveelheid eitjes voorspelt uit de aanwezige vetvoorrraad. Vervolgens ga je dit model toetsen. Uiteraard ga je steeds na of er aan de aannames is voldaan.

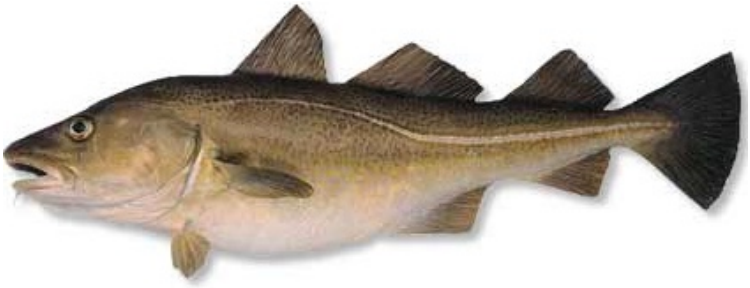
Het bestand.

Cod bevat metingen aan 21 vrouwtjeskabeljauwen uit de Baltische zee (Skjaeraasen et al., 2006, aangepast). Van elk vrouwtjes staat in kolom LipidReserves de hoeveelheid vetreserve rondom de voortplantingsorganen (aangegeven als verbrandingswaarde in kJ) en staat in kolom NumberEggs de hoeveelheid geproduceerde eitjes.

De ruwe data bekijken.

Hiernaast staan de eerste 6 rows van dataframe **cod**.
In totaal bevat **cod** 21 rijen en 2 kolommen.

	LipidReserves	NumberEggs
1	6433.826	2342330
2	7232.024	2481209
3	7372.072	2696324
4	8289.623	3042345
5	8301.900	2910665
6	9056.520	3230766



Kabeljauw (Gadus morhua).

Nagaan of de variabelen het juiste *type* hebben en dit eventueel aanpassen.

Zoals gebruikelijk dien je eerst na te gaan of de *types* die R aan de kolomvectoren heeft toegekend, reëel zijn gezien het soort variabelen dat deze kolomvectoren representeren:

Script:

```
str(cod)
```

Output:

```
> str(cod)
'data.frame':    21 obs. of  2 variables:
 $ LipidReserves: num  6434 7232 7372 8290 8302 ...
 $ NumberEggs   : num  2342330 2481209 2696324 3042345 2910665 ...
```

'Vetvoorraad' is gemeten in kJ, dus het is correct dat $LipidReserves het "numeric" type heeft.
'Aantal eitjes' is in een discrete numerieke variabele en behoort dus het "integer" type te hebben:

Script:

```
cod$NumberEggs=as.integer(cod$NumberEggs)
str(cod)
```

Output:

```
> cod$NumberEggs=as.integer(cod$NumberEggs)
> str(cod)
'data.frame':    21 obs. of  2 variables:
 $ LipidReserves: num  6434 7232 7372 8290 8302 ...
 $ NumberEggs   : int  2342330 2481208 2696324 3042345 2910665 ...
```

De data bekijken.

Zoals je inmiddels weet, is het een goede gewoonte om voordat je de data gaat analyseren, deze eerst grafisch weer te geven om te zien hoe de data verdeeld zijn en om reeds eventuele patronen te ontdekken.

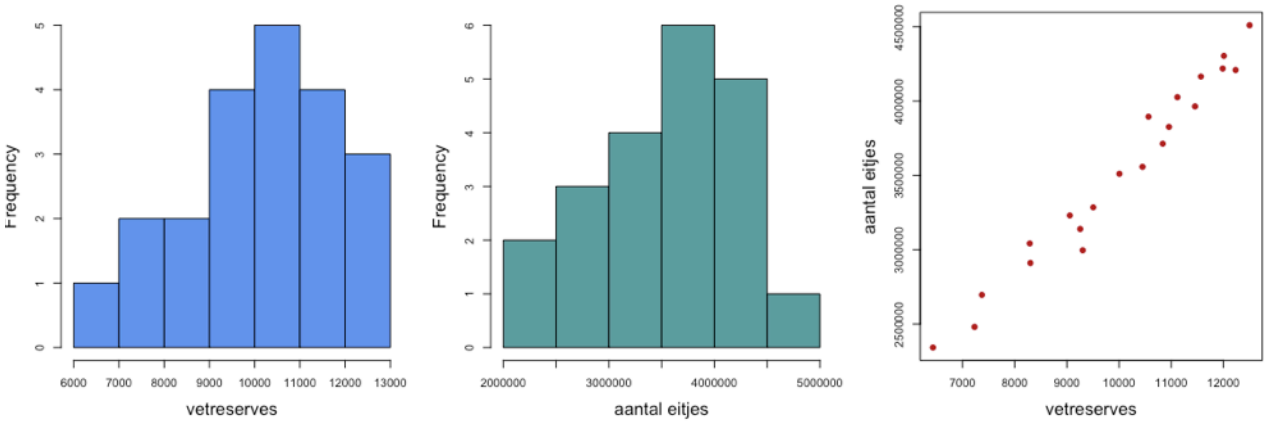
Grafische weergaven.

We maken een histogram van elk van de variabelen apart en een scatterplot van beide variabelen samen[1].

Script (niet alle gebruikte parameters staan vermeld)**:**

```
hist(cod$LipidReserves,col="cornflowerblue",main=NULL,xlab="vetreserves")
hist(cod$NumberEggs,col="cadetblue",main=NULL,xlab="aantal eitjes")
plot(cod$LipidReserves,od$NumberEggs,pch=16,col="firebrick",
                                  xlab="vetreserves",ylab="aantal eitjes")
```

Output (plots-paneel):



Controleren van de aannames.

Als we een lineaire vergelijking willen opstellen om de hoeveelheid eitjes (Y) te voorspellen uit de hoeveelheid vetreserves (X), dus Y' = a+bX, dan moeten we eerst twee aannames controleren alvorens we deze vergelijking mogen opstellen: de *Assumption of random sampling* en de *Assumption of lineair relationship*.

Aanname (a). *Assumption of random sampling.*

"De Y-waarde(s) die bij een bepaalde X-waarde kunnen voorkomen, zijn altijd random trekkingen uit de populatie van mogelijke Y-waardes bij die X-waarde."

---

[1] Zie de files R_histogram.pdf en R_scatterplot.pdf voor uitvoerige informatie over het maken van deze grafische weergaven. Je vindt deze files op Blackboard, knop Computerpractica R, Grafieken maken in R.

Deze aanname kun je niet uit de data afleiden. We zullen ervan uitgaan dat de gemeten kabeljauw een *random sample* is uit de populatie van kabeljauwen in de Baltische Zee, en dat de hoeveelheid eitjes die er bij een kabeljauw is geteld, een random waarde is uit alle mogelijke hoeveelheden eitjes voor de vetvoorraad van die kabeljauw.

Aanname (b). *Assumption of lineair relationship*.

"Variabelen X en Y moeten een lineaire samenhang vertonen."
De reeds gemaakte scatterplot laat een langgerekte puntenwolk zien die geöriënteerd ligt langs een stijgende, rechte, denkbeeldige lijn. Er zijn geen subgroepen of *extremes*. Aan de voorwaarde van lineariteit is dus voldaan.

Opstellen van de lineaire regressievergelijking.

Omdat er aan de twee bovengenoemde aannames is voldaan, mogen we een lineaire regressievergelijking opstellen. Je doet dat in R met functie **`lm()`**. De afkorting 'lm' staat voor 'lineair model'. Een regressievergelijking heet namelijk ook wel (regressie)model.
De syntax van de functie is `lm(y~x)`. Hierin is `y` de *response variable* (in de regressieanalyse ook wel *depending variable* genoemd) en `x` de *explanatory variable* (in de regressieanalyse ook wel *predictor variable* genoemd)[2].

Vraag 17.1 Verklaar de benamingen *depending variable* en *predictor variable*.

Je kunt alleen een regressievergelijking opstellen als beide variabelen numeriek zijn, en dat is hier het geval[3]. Omdat we de output van `lm()` nog vaker nodig hebben, is het slim om deze output toe te kennen aan een variabele (die we, omdat de output een regressiemodel is, hier `model` zullen noemen, maar uteraard is elke naam toegestaan):

Script:

```
model=lm(cod$NumberEggs~cod$LipidReserves)
model
```

Output:

```
> model=lm(cod$NumberEggs~cod$LipidReserves)
> model

Call:
lm(formula = cod$NumberEggs ~ cod$LipidReserves)

Coefficients:
      (Intercept)  cod$LipidReserves
          -6195.8              352.4
```

Vraag 17.2 Welke regressievergelijking volgt er uit deze output?

a) `NumberEggs = -6195.8 + 352.4 * LipidReserves`
b) `NumberEggs = 352.4 - 6195.8 * LipidReserves`
c) `LipidReserves = -6195.8 + 352.4 * NumberEggs`
d) `LipidReserves = 352.4 - 6195.8 * NumberEggs`

---

[2] In andere functies ben je de constructie 'response variable ~ grouping variable' (in R jargon heet dat een formula) al tegengekomen, maar daar was die steeds genoteerd als `x~A` (bv. `aov(x~A)`, `boxplot(x~A)`, `wilcox.test(x~A)`).
Bij de functie `lm()` is de notatie `lm(y~x)` iets logischer omdat de *response variable* op de y-as staat en de *explanatory variable* (die als *grouping variable* fungeert) op de x-as.

[3] 'Vetvoorraad' is een numeriek continue variabele en kolomvector $LipidReserve heeft dienovereenkomstig het "numeric" type; 'Aantal eitjes' is een numeriek discrete variabele en kolomvector $NumberEggs heeft het daarmee corresponderende "integer" type. Zie p.17–3.

Vraag 17.3 Hoeveel eitjes verwacht je dat er gelegd worden door een vrouwtjeskabeljauw met een vetreserve van 8000 kJ? Gebruik je rekenmachine, maak de berekening in Excel, of (nog beter) maak de berekening in R.

Vraag 17.4 Waarom mag je de regressievergelijking niet gebruiken om te voorspellen hoeveel eitjes een vrouwtje met een vetreserve van 5000 kJ zal leggen? Tip: zie de eerder gemaakte grafische weergaven op p.17–3.

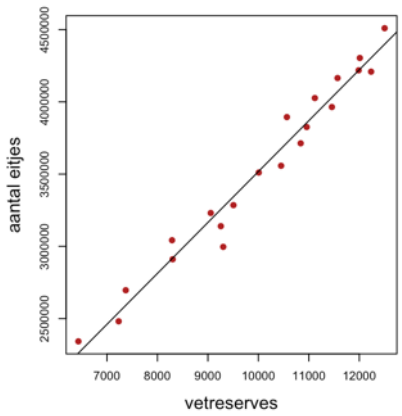
De regressielijn toevoegen aan de scatterplot.

Als je de regressievergelijking hebt opgesteld, kun je de regressielijn in het scatterplot laten weergeven met de functie **`abline()`** (spreek dit als uit: ee bie lain), door simpelweg de variabele die het regressiemodel bevat tussen de functiehaken van deze functie te plaatsen:

Script:

```
abline(model)
```

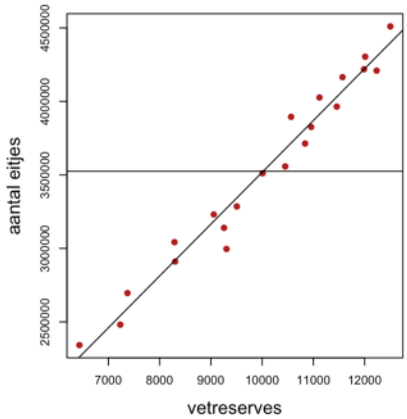
Output (plots-paneel)**:**



De algemene syntax van `abline()` is `abline(a,b)`, waarin `a` het intercept (= snijpunt met de verticale as) voorstelt en `b` de richtingscoëfficiënt. Door het runnen van `abline(a,b)` wordt de lijn y=a+bx getekend in het laatst gegeneerde plot.

In het geval je in een scatterplot specifiek de regressielijn wilt laten tekenen, dan kun je ook `abline(model)` gebruiken, waarbij `model` dan de variabele moet zijn die het regressiemodel bevat.

Vraag 17.5 Veronderstel dat je in het scatterplot ook de lijn $Y=\overline{Y}$ wilt laten tekenen (zie figuur hieronder). Hoe zou je dat dan doen?



Intermezzo
Een lineaire regressievergelijking van de vorm Y'=a+bX is de meeste simpele vorm van een lineaire regressievergelijking en heet dan ook Simple Linear Regression. De vergelijking bevat slechts één predictor variable X.
Je kunt méér predictor variables in je regressievergelijking opnemen (aannemende dat deze in je dataset voorkomen). Voor twee predictor variables breidt de vergelijking uit naar $Y'=a+bX_1+cX_2$, en voor drie predictor variables breidt de vergelijking uit naar $Y'=a+bX_1+cX_2+dX_3$. Etcetera.
In R zou je dat dan programmeren met `lm(y~x1+x2)` resp. `lm(y~x1+x2+x3)`.
Als er sprake is van meer dan één predictor variable spreek je niet meer van Simple Linear Regression maar van Multiple Linear Regression. In deze handleiding beperken we ons tot Simple Linear Regression.

Toetsen van het regressiemodel.

De opgestelde lineaire regressievergelijking geldt voor de sample data (dataframe **cod**). Maar is er op populatieniveau ook sprake van regressie van Y (hoeveelheid eitjes) op X (hoeveelheid vetreserve)? Met andere woorden: kun je ook in werkelijkheid voor een willekeurige vrouwelijke Kabeljauw uit de Baltische zee de hoeveelheid eitjes voorspellen uit de vetreserves rondom de gonaden? Om die vraag te beantwoorden, moeten we het opgestelde regressiemodel toetsen. Wil deze toetsing zinvol zijn, dan moet er opnieuw aan twee aannames voldaan zijn: de *Assumption of normality* en de *Assumption of equality of variance*. Om de redelijkheid van deze twee aannames na te gaan, moeten we eerst een **residual plot** maken:

De residual plot.

In een residual plot staan de **residuals** van Y geplot tegen X. De residuals van Y zijn de verschillen tussen de gemeten Y-waarden en de voorspelde Y-waarden (in het scatterplot dus de verticale afstanden tussen de punten en de regressielijn). Elk verschil tussen een gemeten Y-waarde en diens voorspelde Y-waarde representeert de spreiding van variabele Y voor de betreffende waarde van variabele X (boek, p. 546).

Om een residual plot te maken, moet je eerst voor elke X-waarde de residual berekenen (dus Y–Y'). Dit kan in R met de functie **`residuals()`**. Binnen de functiehaken moet je een regressiemodel opnemen (die ons geval `model` heet).
Omdat de output van `residuals()` nodig is om een residual plot te maken, is het zinnig om ook nu weer de output aan een variabele toe te kennen. Het netst is om daar meteen een kolomvector in dataframe **cod** van te maken:

Script:

```
cod$ResidualEggs=residuals(model)
cod$ResidualEggs
```

Output:

```
> cod$ResidualEggs=residuals(model)
> cod$ResidualEggs

 [1]   81517.8596  -60855.0593  104913.4558  127628.3800   -8377.4978   45827.3400
 [7] -115627.2790 -275740.2591  -58259.2445   -9399.7925 -118442.8122  178233.5669
[13]  -99659.3433  -28637.9503  114661.2904  -66501.7357   94613.6368     770.4562
[19]   77878.6211  -95596.6722  111053.0391
```

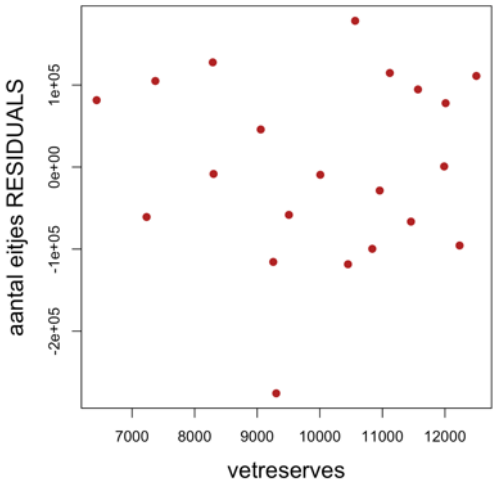
Vraag 17.6 Hebben de punten in het scatterplot die ónder de regressielijn liggen een negatieve of een positieve waarde voor hun residual?

Als we de residuals plotten tegen hun X-waardes, ontstaat een residual plot:

Script (niet alle gebruikte parameters staan vermeld)**:**

```
plot(cod$LipidReserves,cod$ResidualEggs,pch=16,col="firebrick",
                        xlab="vetreserves",ylab="aantal eitjes RESIDUALS")
```

Output (plots-paneel)**:**



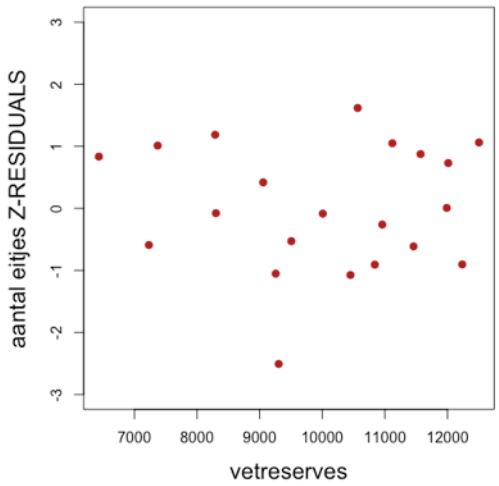
In dit plot staan de residuals uitgezet tegen de X-waardes. Bij het controleren van de aannames van normaliteit en gelijkheid van variantie is het echter makkelijker als de gestandaardiseerde residuals staan uitgezet tegen de X-waardes. Gestandaardiseerde residuals zijn residuals die met een Z-transformatie zijn omgezet in Z-waardes.

Met functie **`rstandard()`** kun je gestandaardiseerde residuals genereren. Binnen de functiehaken moet je een regressiemodel opnemen (in ons geval `model`). Als je de output van `rstandard()` toekent aan een variabele, kun je daar dan vervolgens weer een scatterplot van maken. Het is wederom het netst om een nieuwe kolomvector in dataframe **cod** aan te maken als variabele:

Script (niet alle gebruikte parameters staan vermeld)**:**

```
cod$ZResidualEggs=rstandard(model)
plot(cod$LipidReserves,cod$ZResidualEggs,pch=16,col="firebrick",
                                                    ylim=c(-3,3))
```

Output (plots-paneel)**:**

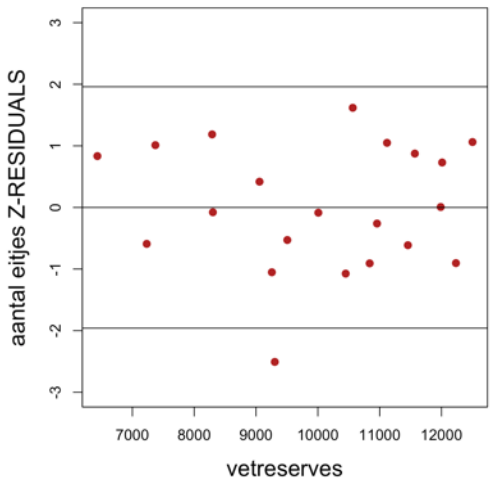


Merk op dat we de Y-as hebben gestretched (waar vind je dat terug in het script?) vooruitlopend op het nu volgende scriptgedeelte, waarin we in het residual plot de lijnen Z=0, Z=–1.96 en Z=+1.96 laten tekenen. Deze lijnen helpen ons straks om makkelijker de aanname van normaliteit te kunnen controleren.

Script:

```
abline(-1.96,0)
abline(0,0)
abline(1.96,0)
```

Output (plots-paneel)**:**



Figuur 17.1
Residual plot van gestandaardiseerde residuals van NumberEggs (Y) uitgezet tegen LipidReserves (X).
In de plot staan ook de lijnen Z=-1.96, Z=0 en Z=+1.96 getekend.

Nu is het residual plot klaar om te kunnen gebruiken bij het controleren van de aannames van normaliteit en gelijkheid van varianties.

Aanname (c). *Assumption of normality.*
"Bij elke waarde van X hoort een normaal verdeelde populatie van mogelijke Y-waardes."
In termen van het residual plot betekent dat dus dat bij elke waarde van vetreserves (X-as) de gestandaardiseerde residual uit een Z-verdeling moet komen.
Hoe kun je zien of dat zo is? Je moet naar twee aspecten kijken:
(1) Rondom de lijn Z=0 moet de puntenwolk de grootste dichtheid vertonen, en hoe verder van de lijn Z=0, des te ijler de puntenwolk moet zijn. Dit is natuurlijk omdat in een Z-verdeling de meeste waarden rondom Z=0 liggen en steeds minder punten in de richting van de staarten.
Hoe is dat hier? Omdat we te maken hebben met een kleine steekproef (n=21), is dit aspect wat moeilijker te beoordelen maar gaat wel in de goede richting (Figuur 17.1, hierboven).
In het boek geeft Figuur 17.5-4 op p. 560 een voorbeeld van een grotere steekproef waarbij het genoemde aspect goed te zien is:

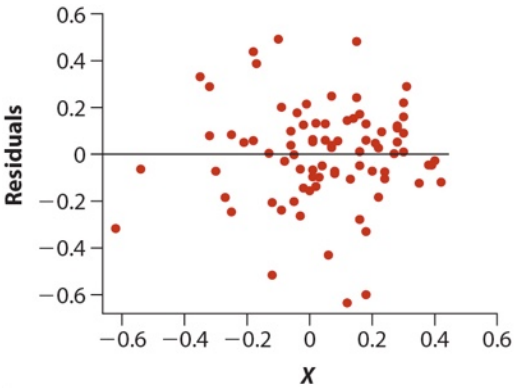


Figure 17.5-4 (left panel) uit het boek (p. 560).
De figuur demonstreert een residual plot waarbij vrij goed aan de voorwaarde van normaliteit is voldaan.

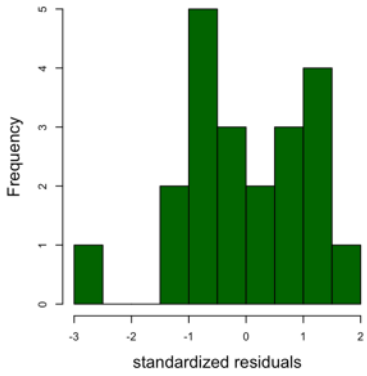
(2) Hoogstens 5% van de punten mag zich onder de lijn Z=–1.96 of boven de lijn Z=+1.96 bevinden. Dat is omdat in een Z-verdeling slechts 5% van de waarden buiten het interval [–1.96,1.96] ligt. Als de residuals normaal verdeeld zijn (en de gestandaardiseerde residuals dus Z-verdeeld zijn) dan moet dit terug te zien zijn in de ligging van de punten: 95% moet dan binnen het gebied liggen dat begrensd wordt door de lijnen Z=–1.96 en Z=1.96.
Hoe is dat hier? In de residual plot (Figuur 17.1, vorige bladzijde) zie je dat slechts 1 van de 21 gestandaardiseerde residuals onder de lijn Z=–1.96 ligt en dus buiten het interval [–1.96,1.96] valt. Omdat 1 van de 21 minder is dan 5%, is dus aan dit aspect van de normaliteit zeker voldaan.

Vraag 17.7 Hiernaast staat het histogram van de gestandaardiseerde residuals en hieronder staat de output van de Shapiro-Wilk test of normality uitgevoerd op de gestandaardiseerde residuals.

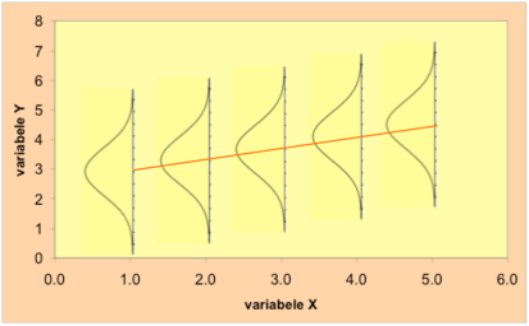
```
        Shapiro-Wilk normality test

data:  cod$ZResidualEggs
W = 0.94689, p-value = 0.2972
```

Waarom is het genereren van zo'n histogram en het runnen van een Shapiro-Wilk test **geen** bruikbare methode om de normaliteit van de residuals te beoordelen?

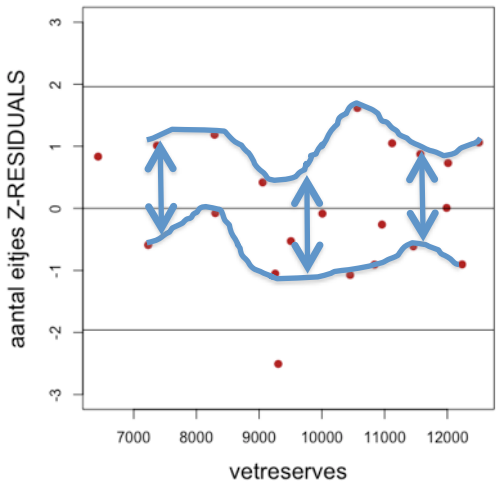


Aanname (d). *Assumption of equality of variance*.
"De normaalverdeelde populaties waaruit de gemeten Y-waardes afkomstig zijn, hebben dezelfde variantie." (Figuur 17.2).



Figuur 17.2
De normaal verdeelde populaties waaruit (per X-waarde) de Y-waardes afkomstig zijn, hebben dezelfde variantie. (Plaatje uit het hoorcollege over Chapter 17).

Of er aan deze aanname voldaan is, kan worden opgemaakt uit het residual plot:
over de hele breedte van het plot moet de spreiding van de puntenwolk in de y-richting ongeveer gelijk zijn. Hieraan wordt redelijk voldaan (afgezien van het ene punt dat ook al afwijkend was bij de beoordeling van de normaliteit):



Figuur 17.3
Als de spreiding van de punten in verticale richting over de gehele breedte van het plot ongeveer gelijk is, duidt dat erop dat de populaties waaruit de Y-waarden afkomstig zijn, dezelfde variantie hebben.

Figure 17.5-4 van het boek toont een residual plot waarin duidelijk **geen** sprake is van gelijke variantie:

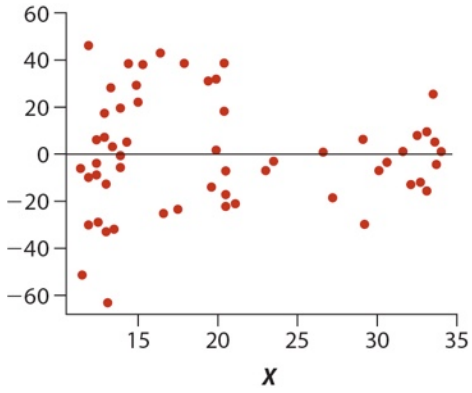


Figure 17.5-4 (right panel) uit het boek (p. 560).
De figuur demonstreert een residual plot waarbij duidelijk NIET aan de voorwaarde van gelijke variantie is voldaan.

De nulhypothese.

Omdat er zowel aan de aanname van normaliteit is voldaan als aan de aanname van gelijke varianties, kunnen we het gevonden regressiemodel gaan toetsen.

De nulhypothese luidt: H0: $\beta = 0$.

Dat komt neer op: bij kabeljauw uit de Baltische zee is de hoeveelheid eitjes niet te voorspellen uit de vetreserve rondom de voortplantingsorganen: voor elke hoeveelheid vetreserve (X) is de beste voorspelling (Y') altijd het gemiddelde aantal eitjes uit eerdere metingen ($\mu_Y = \alpha$.)

De nulhypothese kan worden getoetst door het regressiemodel op te nemen binnen de functie **`summary()`**. Op dezelfde wijze heb je in het hoofdstuk over Chapter 15[4] ook de nulhypothese van een One-way ANOVA model getoetst.

Script:

```
summary(model) #NB.'model' is het regressiemodel zoals verkregen uit lm()
```

Output:

```
> summary(model)

Call:
lm(formula = cod$NumberEggs ~ cod$LipidReserves)

Residuals:
    Min      1Q  Median      3Q     Max
-275740  -66502   -8377   94614  178234

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        -6195.83  145587.01  -0.043    0.966
cod$LipidReserves    352.36      14.32  24.611 7.12e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 113200 on 19 degrees of freedom
Multiple R-squared:  0.9696,     Adjusted R-squared:  0.968
F-statistic: 605.7 on 1 and 19 DF,  p-value: 7.116e-16
```

[4] R2 Ch15a One-way ANOVA.pdf

De output is vrij uitgebreid en geeft allerhande informatie. Dit komt vaker voor bij R output en de kunst is om de informatie waar het om gaat, terug te vinden in die output.

We bekijken eerst de output met betrekking tot de residuals:

```
Residuals:
    Min      1Q  Median      3Q     Max
-275740  -66502   -8377   94614  178234
```

Vraag 17.8 Welke conclusie kun je op grond van deze output trekken?
Kies uit de volgende 3 opties:

a) Hoogstens de helft van de residuals heeft een negatieve waarde.
b) Precies de helft van de residuals heeft een negatieve waarde.
c) Minstens de helft van de residuals heeft een negatieve waarde.

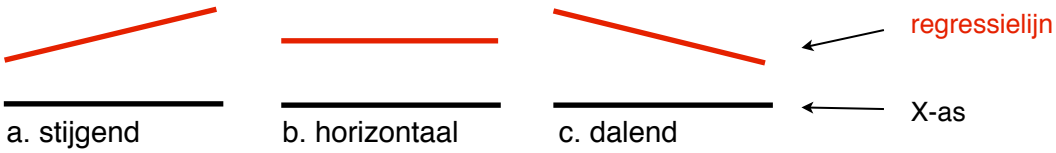
Nu kijken we naar de output met betrekking tot de coëfficiënten (de 'a' en de 'b'):

```
Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        -6195.83  145587.01  -0.043    0.966
cod$LipidReserves    352.36      14.32  24.611 7.12e-16 ***
```

Vraag 17.9a Welke regel geeft de resultaten van het toetsen van H0: $\beta = 0$?

Vraag 17.9b Welke waarde heeft de toetsingsgrootheid bij deze toets?

Vraag 17.9c Wat zegt dat over de hoek van de regressielijn op populatieniveau?
Kies uit de volgende 3 opties.

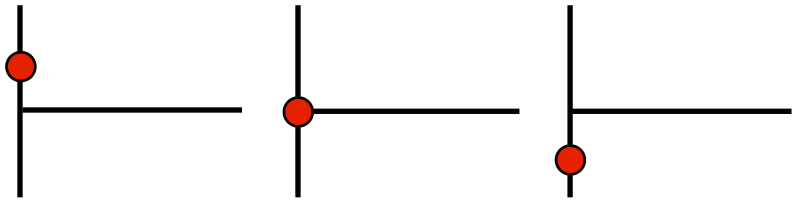


a. stijgend b. horizontaal c. dalend

Vraag 17.10a Welke regel geeft de resultaten van het toetsen van H0: $\alpha = 0$?

Vraag 17.10b Moet de nulhypothese H0: $\alpha = 0$ verworpen worden?

Vraag 17.10c Wat zegt dat over het snijpunt van de populatie-regressielijn met de y-as?
Kies uit de volgende 3 opties.



a. (0,positief) b. (0,0) c. (0,negatief)

Vraag 17.11 Combineer je antwoorden op 17.9c en 17.10c en schets hoe de regressielijn op populatieniveau verloopt.

Het snijpunt van de regressielijn met de Y-as valt buiten het domein van de metingen en heeft dus niet zoveel betekenis in voorspellende zin. Maar wat ook de werkelijke regressiecurve is voor kleine waardes van X (misschien is de regressie daar niet lineair), er zijn slechts 3 mogelijkheden voor een snijpunt met een as:

a) de regressiecurve begint ergens op de Y-as
b) de regressiecurve begint in (0,0)
c) de regressiecurve begint ergens op de X-as

Vraag 17.12 Gezien het feit dat het hier gaat om de samenhang tussen het leggen van eitjes en de hoeveelheid vetreserves, welk van de 3 mogelijkheden lijkt je vanuit biologisch oogpunt dan het meest waarschijnlijk en welke lijkt je het minst waarschijnlijk?

Tot slot bekijken we in de output naar het laatste gedeelte:

```
Residual standard error: 113200 on 19 degrees of freedom
Multiple R-squared:  0.9696,     Adjusted R-squared:  0.968
F-statistic: 605.7 on 1 and 19 DF,  p-value: 7.116e-16
```

Op de eerste regel staat de `Residual standard error`: dit is de $SE_{residuals}$ waarmee R uit de residuals de gestandaardiseerde residuals heeft berekend.
Op de tweede regel staan de `Multiple R-squared` en de `Adjusted R-squared`.
De `Multiple R-squared` is de R^2 die in het boek gedefiniëerd staat als $SS_{regression}/SS_{total}$ (p. 555) en geeft aan hoeveel van de variantie in 'NumberEggs' verklaard wordt door het model.

Vraag 17.13 Hoeveel procent van de variantie in het aantal gelegde eitjes kan worden toegeschreven aan de samenhang met de hoeveelheid vetreserves?

De overige procenten moeten worden toegeschreven aan andere, niet onderzochte, variabelen (bijvoorbeeld leeftijd en/of fysiologische fitheid van de kabeljauw).
De `Adjusted R-squared` is een gecorrigeerde versie van de Multiple R-squared: als je model meer dan één predictor variable bevat, geeft de Adjusted R-squared een betere indruk van de verklaarde variantie dan de Multiple R-squared. In het kabeljauwen-regressiemodel is slechts sprake van één predictor variable (welke?), dus volstaat het om de Multiple R-squared te gebruiken.
Op de laatste regel staat een `F-statistic` met als vrijheidsgraden 1 en 19, en P-waarde `7.116e-16`. De `F-statistic` is de toetsingsgrootheidwaarde die je vindt als je de data toetst met een ANOVA (boek p. 552, The ANOVA approach). De Sum of Squares en de Mean Squares worden echter niet vermeld. Als je die ook wilt zien, moet je een aparte One-way ANOVA uitvoeren op de data:

Script:

```
model2=aov(cod$NumberEggs~cod$LipidReserves)
summary(model2)
```

Output:

```
> model2=aov(cod$NumberEggs~cod$LipidReserves)
> summary(model2)
                  Df    Sum Sq   Mean Sq F value   Pr(>F)
cod$LipidReserves  1 7.756e+12 7.756e+12   605.7 7.12e-16 ***
Residuals         19 2.433e+11 1.281e+10
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Vraag 17.14 Laat zien dat als je de Sum of Squares van `cod$LipidReserves` deelt door de totale Sum of Squares, je dan de waarde van de Multiple R-squared krijgt.

Conclusie.
Bij kabeljauwen uit de Baltische zee kan het aantal eitjes dat een vrouwtje legt middels een lineair regressiemodel voorspeld worden uit de hoeveelheid vetreserves rondom de voorplantingsorganen van het vrouwtje.

Betrouwbaarheidsintervallen voor de coëfficiënten.

Naast het toetsen van de coëfficiënten van het regressiemodel met behulp van de functie `summary()`, kun je ook betrouwbaarheidsintervallen opstellen voor de coëfficiënten met behulp van functie **`confint()`**. Binnen de functiehaken moet het regressiemodel komen.

Script:

```
confint(model)
```

Output:

```
> confint(model)
                          2.5 %      97.5 %
(Intercept)       -310912.9475 298521.2831
cod$LipidReserves     322.3916    382.3238
```

Vraag 17.15 Hoe kun je in deze output zien dat de eerder getoetste H0: $\alpha = 0$ inderdaad niet verworpen kan worden bij een significantieniveau van 0.05? Leg uit.

Default geeft R als output het 95% betrouwbaarheidsinterval. Wil je een ander percentage dan kun je parameter `level=` opnemen met daarachter een fractie die aangeeft welk percentage betrouwbaarheidsinterval je wilt.
Zo bepaal je het 99% betrouwbaarheidsinterval met `confint(model,level=0.99)`.

Het doornemen van dit uitgewerkte voorbeeld en het beantwoorden van de 15 vragen is het huiswerk voor week 10. **Zorg dat je bij aanvang van het practicum de antwoorden op de vragen bij de hand hebt zodat de assistenten kunnen zien dat je ze hebt gemaakt.** Sommige vragen waren wat moeilijker dan andere. Als je op een vraag het antwoord niet weet, **schrijf dan niet niks op**, maar schrijf je gedachtengang of overweging op of de richting waarin je het antwoord mogelijk moet zoeken.

Tijdens het practicum kun je onderstaande Samenvattingen gebruiken als leidraad bij het uitvoeren van een Normaliteitsonderzoek of het transformeren van een variabele. Weet je niet meer hoe iets moet, dan kun je het Uitgewerkte voorbeeld raadplegen of assistentie inroepen.

Samenvatting.

1. Ga na of beide variabelen 'numeric' vectors zijn.

```
str(dataframe)
```

2. Ga na of er aan de volgende 2 aannames is voldaan:

(a) *Assumption of random sampling.*
Hiervoor is geen script, ga na hoe de *sample* is verkregen.

(b) *Assumption of linear relationship.*
Maak een scatterplot en bekijk op het oog of beide variabelen lineair samenhangen.
`plot(x,y)` (zie R_scatterplot.pdf voor details)
Als aan beide aannames is voldaan, kun je door naar punt 3.

3. Stel de lineaire regressievergelijking op, en plot de regressielijn in het scatterplot.

```
model1=lm(y~x)
model1
abline(model1)
```

4. Maak een residual plot waarin de gestandaardiseerde residuals worden uitgezet tegen x, en plot de lijnen Z=–1.96, Z=0 en Z=+1.96.

```
Zres=rstandard(model1)
plot(x,Zres)
abline(-1.96,0)
abline(0,0)
abline(1.96,0)
```

5. Ga na of er aan de volgende 2 aannames is voldaan:

(c) *Assumption of normality.*

Ga dit na in het residual plot:

1) de meeste punten moeten dicht bij de lijn Z=0 liggen; hoe verder van de lijn, hoe lager de puntendichtheid moet zijn;

2) hoogstens 5% van de punten mag verder weg liggen dan de lijn Z=–1.96 of de lijn Z=1.96.

(d) *Assumption of equality of variance.*

Ga dit na in het residual plot:

Over de gehele breedte van het plot moet de spreiding in verticale richting ongeveer gelijk zijn (zie Figuur 17.3, bladzijde 17–9).

6. Toets het regressiemodel.

Toetsen mag alleen als er aan aannames (c) en (d) is voldaan.

`summary(model1)`: geeft de resultaten van t-toetsen voor de coëfficiënten en van een ANOVA voor het regressiemodel als geheel, en de waarde voor R^2

```
model2=aov(y~x)
```

`summary(model2)`: geeft een ANOVA tabel voor het regressiemodel als geheel

7. Genereer betrouwbaarheidsintervallen voor de coëfficiënten.

```
confint(model1)
```