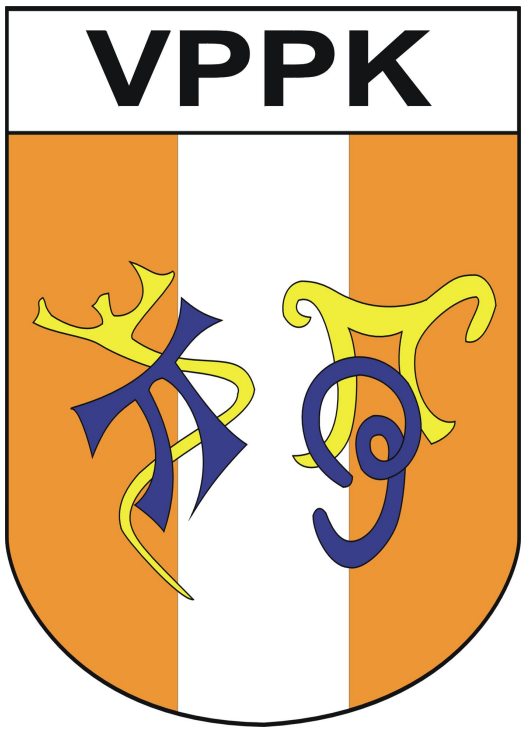
Statistiek II

Sessie 6

Feedback

Deel 6



VPPK

Universiteit Gent

2017-2018

Feedback Oefensessie 6

We hebben besloten de bekomen grafieken in `R` niet in het document in te voegen, dit omdat het document met de code al vrij lang is, en de grafieken nog veel meer plaats zouden in beginnen nemen dat het onoverzichtelijk wordt, en ook niet meer echt milieuvriendelijk als er mensen dit willen printen. We gaan ervan uit dat jullie met het volgen van de code zelf de grafieken kunnen maken (aangezien ook de code van de grafieken in deze bundel staan).

1 Statismex, gewicht en bloeddruk

1. Afhankelijke variabele: Bloeddruk (`bloeddruk$BD`)
   Onafhankelijke variabelen: Dosis (`bloeddruk$dosis`) en gewicht (`bloeddruk$gewicht`)
   $H_0 : \beta_{\text{dosis}} = 0$ vs. $H_a : \beta_{\text{dosis}} \neq 0$
   We gaan een lineaire regressie uitvoeren en dan kijken naar de t-waarde en bijhorende p-waarde van dosis.

2. Alle variabelen zijn van minstens intervalniveau, dus deze voorwaarde is voldaan

   ```
   > load("bloeddruk.RData")
   > lm_bloeddruk <- lm(BD ~ dosis + gewicht, data=bloeddruk)
   > plot(lm_bloeddruk)
   ```

   De data is normaalverdeeld (de plot is hier niet duidelijk, maar als we gaan kijken naar de output via het commando `summary()` dan zien we bij residuals dat de mediaan zeer dicht bij 0 ligt, het eerste en derde kwartiel ongeveer even ver van 0 liggen, alsook de minimum en maximum), de residuen geven gemiddeld 0 en de homoscedasticiteit is in orde.

3. Dit is niet gevraagd, maar we geven het er liever wel bij. We aanvaarden de nulhypothese, want als we gaan kijken naar de output zien we dat de t-waarde bij dosis 0.195 is met bijhorende p-waarde 0.846 en dus groter dan 0.05.

   ```
   > summary(lmbloeddruk)

   Call:
   lm(formula = BD ~ dosis + gewicht, data = bloeddruk)

   Residuals:
       Min      1Q  Median      3Q     Max
   -3.2939 -0.7027  0.0525  0.7261  3.4197

   Coefficients:
   Estimate Std. Error t value Pr(>|t|)
   (Intercept) 10.06915    1.05041   9.586 1.05e-15 ***
   dosis        0.02057    0.10566   0.195    0.846
   gewicht      0.02041    0.01600   1.276    0.205
   ---
   Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

   Residual standard error: 1.356 on 97 degrees of freedom
   Multiple R-squared:  0.02226,        Adjusted R-squared:  0.0021
   F-statistic: 1.104 on 2 and 97 DF,  p-value: 0.3356
   ```

2 Uitslagen en didactisch softwarepakket

1. Afhankelijke variabel: uitslagen
   Onafhankelijke variabele/predictor: Softwarepakket → Nominaal meetniveau (dus maken we gebruik van ANOVA)
   $H_0 : \beta_{\text{HV1}} = \beta_{\text{HV2}} = 0$ vs. H_a : minstens 1 β verschilt van 0

2. `R` gebruikt automatisch dummy-codering (voor een ander coderingschema dient dit specifiek aangepast te worden). Het eerste niveau is het referentieniveau, want deze komt niet voor in de output. We krijgen een p-waarde van 0.02856 dus verwerpen we de nulhypothese.

```
> load("uitslagen.RData")
> lm_uitslag <- lm(formula = uitslagen$uitslag ~ uitslagen$soft)
> summary(lm_uitslag)

Call:
lm(formula = uitslagen$uitslag ~ uitslagen$soft)

Residuals:
     Min       1Q   Median       3Q      Max
-12.6865  -3.8038  -0.5868   4.1334  11.9115

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         54.3927     0.8024  67.786  < 2e-16 ***
uitslagen$softB     -1.6278     1.0894  -1.494  0.13737
uitslagen$softgeen  -2.9691     1.0995  -2.700  0.00778 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 5.262 on 140 degrees of freedom
Multiple R-squared:  0.04953,        Adjusted R-squared:  0.03595
F-statistic: 3.647 on 2 and 140 DF,  p-value: 0.02856
```

3.
```
> aggregate(uitslagen$uitslag, by = list(uitslagen$soft), FUN = mean)
  Group.1        x
1       A 54.39274  #intercept
2       B 52.76497  #intercept + beta1
3    geen 51.42360  #intercept + beta2
```

4. Modelassumpties: alle assumpties blijken stand te houden

```
> plot(lm_uitslag)
```

3 Uitslagen en geslacht

```
> man <- uitslagen$uitslag[uitslagen$geslacht=="m"]
> vrouw <- uitslagen$uitslag[uitslagen$geslacht=="v"]
>
> t.test(man, vrouw)

        Welch Two Sample t-test
```

```
data:  man and vrouw
t = 3.3174, df = 140.04, p-value = 0.001158
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 1.161657 4.588663
sample estimates:
mean of x mean of y
 54.16202  51.28686

> lm_geslacht <- lm(formula = uitslag ~ geslacht, data=uitslagen)
> summary(lm_geslacht)

Call:
lm(formula = uitslag ~ geslacht, data = uitslagen)

Residuals:
     Min       1Q   Median       3Q      Max
-11.5103  -3.8956  -0.3256   4.3251  15.0174

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  54.1620     0.5981  90.552  < 2e-16 ***
geslachtv    -2.8752     0.8674  -3.315  0.00117 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 5.18 on 141 degrees of freedom
Multiple R-squared:  0.07229,        Adjusted R-squared:  0.06571
F-statistic: 10.99 on 1 and 141 DF,  p-value: 0.001166
```

We kunnen zowel een ANOVA met 1 hulpveranderlijke als een t-test gebruiken. We zien dat beide p-waarden de nulhypothese verwerpen en dat er dus een invloed is van geslacht op de uitslagen. Er zit een klein verschil op beide p-waarden. Dit komt omdat in de Welch t-test van verschillende varianties tussen de groepen wordt uitgegaan, in de ANOVA daarentegen hebben we de assumptie van homoscedasticiteit. Wanneer je in het commando van `t.test()` het argument `var = TRUE` zou toevoegen krijg je exact dezelfde p-waarden.

4 Uitslagen, didactisch softwarepakket en geslacht

1. De afhankelijke variabele, uitslag, is van ratio meetniveau
   Beide andere variabelen zijn van nominaal meetniveau

2. $H_0 : \beta_{\text{HV Software1}} = \beta_{\text{HV Software2}} = 0$ vs. H_a : minstens 1 van de β's verschilt van 0
   Voor deze nulhypothese gaan we het model met enkel geslacht vergelijken met het model waar ook het softwarepakket in is opgenomen:

   ```
   > lm_softgeslacht <- lm(formula = uitslag ~ soft + geslacht, data = uitslagen)
   > anova(lm_geslacht, lm_softgeslacht)
   Analysis of Variance Table

   Model 1: uitslag ~ geslacht
   ```

```
Model 2: uitslag ~ soft + geslacht
  Res.Df    RSS Df Sum of Sq      F   Pr(>F)
1    141 3783.3
2    139 3523.9  2    259.44 5.1169 0.007174 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

We observeren een p-waarde van 0.007174 dewelke kleiner is dan 0.05 en dus verwerpen we de nulhypothese. Dit wil zeggen dat, rekening houdend met geslacht, het softwarepakket een invloed heeft op de uitslagen.

3.
```
> summary(lmsoftgeslacht)

Call:
lm(formula = uitslag ~ soft + geslacht, data = uitslagen)

Residuals:
     Min       1Q   Median       3Q      Max
-11.2893  -3.5283  -0.2368   3.6996  13.3087

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  56.1577     0.9021  62.254  < 2e-16 ***
softB        -1.9666     1.0464  -1.879  0.06228 .
softgeen     -3.3788     1.0578  -3.194  0.00174 **
geslachtv    -3.1622     0.8483  -3.728  0.00028 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 5.035 on 139 degrees of freedom
Multiple R-squared:  0.1359,        Adjusted R-squared:  0.1173
F-statistic: 7.288 on 3 and 139 DF,  p-value: 0.0001418
```

- Mannen in groep A: intercept $= 56.1577$
- Mannen in groep B: intercept $+ \beta_{\text{softB}} = 54.1911$
- Vrouwen in groep B: intercept $+ \beta_{\text{softB}} + \beta_{\text{geslachtv}} = 51.0289$

5 Massachusetts Test Score Data Set

1. We kunnen niet veel doen met de administratieve gegevens van het district (code en district) - dit omdat het een nominale variabele is (het heeft ook ontzettend veel niveaus, als we deze omzetten naar hulpveranderlijken krijgen we problemen omdat we te weinig vrijheidsgraden hebben). Ook heeft het geen zin om de scores in de 8ste graad op te nemen, omdat we zoeken naar predictoren voor scores in de 4de graad - chronologisch gezien kan dit geen predictor zijn.

2.
```
> lmMCAS1 <- lm(formula = totsc4 ~ regday + specneed + bilingua + totday + spc
                + speced + tchratio + logpercap + avgsalary, data = MCAS)
> summary(lmMCAS1)

Call:
lm(formula = totsc4 ~ regday + specneed + bilingua + totday +
spc + speced + tchratio + logpercap + avgsalary, data = MCAS)
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-23.639  -5.618   1.075   5.184  29.095

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.315e+02  1.563e+01  40.399  < 2e-16 ***
regday       5.382e-03  4.268e-03   1.261 0.209294
specneed    -1.342e-04  7.352e-04  -0.183 0.855380
bilingua    -6.504e-06  3.219e-05  -0.202 0.840176
totday      -9.909e-03  4.177e-03  -2.372 0.018995 *
spc         -1.543e-01  2.950e-01  -0.523 0.601690
speced      -3.619e-01  2.749e-01  -1.317 0.190073
tchratio    -1.581e+00  4.216e-01  -3.749 0.000256 ***
logpercap    1.172e+02  9.565e+00  12.257  < 2e-16 ***
avgsalary   -1.431e-01  3.334e-01  -0.429 0.668419
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 9.401 on 144 degrees of freedom
Multiple R-squared:  0.6786,        Adjusted R-squared:  0.6585
F-statistic: 33.78 on 9 and 144 DF,  p-value: < 2.2e-16
```

We zien dat de set van predictoren goede predictoren bevat, want we hebben een p-waarde bij de F-toets kleiner dan 0.05. Maar het is nog geen goede set, want er zitten non-significante predictoren tussen. Voordat we een besluit gaan nemen op basis van de significantie gaan we eerst zien naar de collineariteit.

3.
```
> library(car)
> vif(lmMCAS1)
   regday  specneed  bilingua    totday       spc
16.501800  2.719184  1.044507 20.458546  1.057020
  speced  tchratio logpercap avgsalary
1.403713  1.463127  1.853955  1.967507
```

We merken een hoge collineariteit tussen **regday** en **totday**. We kunnen best **totday** weglaten, deze heeft de hoogste VIF, alsook is deze informatie vervat in **regday**, **bilingua** en **specneed**.

```
> lmMCAS2 <- lm(formula = totsc4 ~ regday + specneed + bilingua + spc
                + speced + tchratio + logpercap + avgsalary, data = MCAS)
> vif(lmMCAS2)
regday  specneed  bilingua       spc    speced
2.341506  1.462944  1.043719  1.054226  1.100674
tchratio logpercap avgsalary
1.401062  1.818519  1.899639
```

Nu krijgen we geen immens grote afwijkingen meer, en zitten ze allemaal onder de waarde van 3.

4. *We gaan nu nog eenmaal de volledige output geven, maar om ruimte en papier te besparen gaan we voor de rest niet meer de volledige output geven. We veronderstellen dat met de code de output kan gerepliceerd worden.*

```
> summary(lmMCAS2)
```

```
Call:
lm(formula = totsc4 ~ regday + specneed + bilingua + spc + speced +
tchratio + logpercap + avgsalary, data = MCAS)

Residuals:
     Min       1Q   Median       3Q      Max
-22.2462  -6.8130   0.3754   5.8650  29.0949

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.418e+02  1.524e+01  42.102  < 2e-16 ***
regday      -3.997e-03  1.633e-03  -2.447  0.01559 *
specneed    -1.320e-03  5.478e-04  -2.409  0.01724 *
bilingua    -4.406e-06  3.269e-05  -0.135  0.89297
spc         -1.903e-01  2.993e-01  -0.636  0.52586
speced      -6.648e-01  2.472e-01  -2.689  0.00801 **
tchratio    -1.787e+00  4.190e-01  -4.263 3.61e-05 ***
logpercap    1.204e+02  9.624e+00  12.509  < 2e-16 ***
avgsalary   -2.900e-01  3.328e-01  -0.871  0.38497
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 9.55 on 145 degrees of freedom
Multiple R-squared:  0.666,        Adjusted R-squared:  0.6476
F-statistic: 36.14 on 8 and 145 DF,  p-value: < 2.2e-16
```

We merken dat de grootste p-waarde voorkomt bij bilingua, en dat deze niet significant is (let, we stellen nu $\alpha = 0.02$), dus we gaan deze laten vallen.

```
> lmMCAS3 <- lm(formula = totsc4 ~ regday + specneed + spc + speced
               + tchratio + logpercap + avgsalary, data = MCAS)
> summary(lmMCAS3)
```

We zien dat nu `spc` de hoogste p-waarde heeft. Dus we laten deze vallen.

```
> lmMCAS4 <- lm(formula = totsc4 ~ regday + specneed + speced
                   + tchratio + logpercap + avgsalary, data = MCAS)
> summary(lmMCAS4)
```

Nu laten we `avgsalary` weg.

```
> lmMCAS5 <- lm(formula = totsc4 ~ regday + specneed + speced
                   + tchratio + logpercap, data = MCAS)
> summary(lmMCAS5)
```

We zien dat nu alle p-waarden kleiner zijn dan $\alpha = 0.02$. Dan gaan we nu naar de verklaarde variantie kijken. In deze cursus gebruiken we bij meervoudige lineaire regressie de `Adjusted R-squared`. De reden waarom er vaak voor de Adjusted gekozen wordt is dat deze inherent een strafmaat heeft. De determinatiecoëfficiënt R^2 zal altijd stijgen wanneer er predictoren worden toegevoegd, want de SS_{Mod} zal stijgen (en de SS_{Res} dus dalen). Kijken we naar de formule op o.a. pp. 152 in de cursus kunnen we daaruit infereren dat dus R^2 zal stijgen. Als we gaan kijken naar de formule voor $\bar{R}^2$ op o.a. pp. 154 zien we dat er nu rekening wordt gehouden met het aantal observaties en het aantal predictoren in het model. Als we meer predictoren gaan opnemen is er nu ook de kans dat onze maat gaat dalen. Dit wordt ook gebruikt bij andere vormen van modelselectie. Wanneer een predictor wordt toegevoegd en deze zorgt

voor een daling in onze $\bar{R}^2$, dan is het een predictor die niet veel bijdraagt en beter niet kan opgenomen worden. Indien onze $\bar{R}^2$ daarentegen stijgt is de predictor wel belangrijk, want deze zorgt voor nog meer verklaarde variantie bovenop de rest, zelfs met de strafmaat.
Dus we zeggen hier dat onze verklaarde variantie 65.19% is.

5. Als we een negatieve coëfficiënt hebben wil dit zeggen dat wanneer we alle andere variabelen constant houden, en we stijgen op de variabele waarvan we de coëfficiënt bestuderen, dan daalt de geschatte waarde van onze afhankelijke variabele. In deze oefening: als we alle andere predictoren constant houden, en we stijgen op `regday` dan zullen we een lagere score in de 4de graad verwachten. Dus als we meer spenderen verwachten we een lagere score.
Als we een positieve coëfficiënt hebben daarentegen zal de afhankelijke variabele stijgen als we stijgen op de variabele die we onderzoeken, gegeven dat al de andere variabelen constant blijven.

6. Zoals gezegd is er geen uniek antwoord voor deze oefening. Daarom gaan we het voorbeeld gebruiken van de oefeningenbundel om duidelijk te maken wat er precies gebeurt. We hebben onze geobserveerde verklaarde variantie, dewelke afgerond $65\% = 0.65$ is. We verwachten dat IQ bovenop de andere predictoren ook een invloed heeft, dus moeten we bepalen hoeveel extra verklaarde variantie we willen kunnen detecteren. De oefeningenbundel suggereert een extra verklaarde variantie van 5%, dus $0.65 + 0.05 = 0.70$. Met deze gegevens kunnen we al onze f^2 berekenen met de formule op pp. 206:

```
> f2 <- (0.70 - 0.65)/(1-0.70)
> f2
[1] 0.1666667
```

Vervolgens dienen we u te berekenen. u staat voor het verschil van aantal predictoren tussen de twee modellen. In deze oefening willen we enkel IQ toevoegen, dus is er 1 predictor meer, dus $u = 1$:

```
> library(pwr)
> pwr.f2.test(power=0.95, u=1, f2=f2, sig.level=0.05)

     Multiple regression power calculation

              u = 1
              v = 77.94169
             f2 = 0.1666667
      sig.level = 0.05
          power = 0.95
```

Uit deze output kunnen we nu v aflezen, met name $v = 78$ (we ronden dit altijd naar boven af). We weten dat $v = n - p - 1$, dus hervormen we de formule naar $n = v + p + 1$. Het aantal predictoren, p, is het aantal in het huidige model plus IQ, dus 6. Dus: $n = 78 + 5 + 1 = 85$. Onze conclusie is, voor een power van 0.95 en voor het verschil van 0.05 verklaarde variantie, hebben we 85 observaties nodig.

6 California Test Score Data Test

1. We merken dat alle punten inderdaad op een vlak liggen. Dit is logisch want `testscr` is een lineaire functie van `mathscr` en `readscr`. Het beste dat we dan kunnen doen is enkel `testscr` mee opnemen in het model, want deze zullen perfect gecorreleerd zijn (en vormen samen de afhankelijke variabele), en geven dus geen informatie over de predictie van de afhankelijke variabele.

2. We nemen alle predictoren op behalve deze met administratieve informatie over het district en de twee net vernoemde predictoren:

```
> pairs(Caschool[,c(2,3,4,5,6,7,8,9,10,11)],lower.panel = NULL)
```

We zien al wel een sterk lineair verband tussen de afhankelijke variabele en `logavginc`, en de afhankelijke variabele en `elpct`. Dus we zullen deze waarschijnlijk in ons finaal model hebben.

3. We zien niet echt non-lineaire verbanden tussen de predictoren en de afhankelijke variabele

4. We zien een sterke correlatie tussen `teachers` en `ernltot`, `computer` en `teachers`, en `ernltot` en `computer`. Hier schuilt dus zeker gevaar voor collineariteit.

5.
```
> lmca1 <- lm(formula = testscr ~ grspan + enrltot + teachers
             + computer + compstu + expnstu + str + logavginc
             + elpct, data = Caschool)
> vif(lmca1)
   grspan    enrltot   teachers   computer    compstu    expnstu        str
 1.090469 242.910541 273.518825  11.177362   1.485509   1.779020   2.244949
logavginc      elpct
 1.430550   1.495337
```

We merken een zeer hoge VIF bij 2 predictoren. Aangezien `teachers` al vervat zit in `str` is het meest logische om `teachers` te laten vallen:

6.
```
> lmca2 <- lm(formula = testscr ~ grspan + enrltot
                 + computer + compstu + expnstu + str + logavginc
                 + elpct, data = Caschool)
> vif(lmca2)
  grspan   enrltot  computer   compstu   expnstu       str logavginc     elpct
1.088007  9.751880  9.164298  1.423178  1.764857  1.818356  1.422211  1.491291
```

We merken nu een grote VIF bij `enrltot` en `computer`. Aangezien `computer` al vervat zit in `compstu` verkiezen we `computer` te laten vallen:

7.
```
> lmca3 <- lm(formula = testscr ~ grspan + enrltot
             + compstu + expnstu + str + logavginc
             + elpct, data = Caschool)
> vif(lmca3)
  grspan   enrltot   compstu   expnstu       str logavginc     elpct
1.077128  1.339696  1.188017  1.759503  1.816526  1.398746  1.489539
```

Met deze predictoren is de collineariteit volledig in orde.

8. *Aangezien ook hier de output van alles het onoverzichtelijk, overdreven lang, en papierverspillend zou maken nemen we deze niet mee op. Met de code is dit eenvoudig zelf te doen.*

```
> lmcavw1 <- lm(formula = testscr ~ grspan, data = Caschool)
> lmcavw2 <- lm(formula = testscr ~ enrltot, data = Caschool)
> lmcavw3 <- lm(formula = testscr ~ compstu, data = Caschool)
> lmcavw4 <- lm(formula = testscr ~ expnstu, data = Caschool)
> lmcavw5 <- lm(formula = testscr ~ str, data = Caschool)
> lmcavw6 <- lm(formula = testscr ~ logavginc, data = Caschool)
> lmcavw7 <- lm(formula = testscr ~ elpct, data = Caschool)
>
> summary(lmcavw1)
> summary(lmcavw2)
> summary(lmcavw3)
> summary(lmcavw4)
```

```
> summary(lmcavw5)
> summary(lmcavw6)
> summary(lmcavw7)
```

We observeren de grootste F-waarde bij het zesde model, dus `logavginc` nemen we als eerste op in het model.

9.
```
> lmcavw1 <- lm(formula = testscr ~ logavginc + grspan, data = Caschool)
> lmcavw2 <- lm(formula = testscr ~ logavginc + enrltot, data = Caschool)
> lmcavw3 <- lm(formula = testscr ~ logavginc + compstu, data = Caschool)
> lmcavw4 <- lm(formula = testscr ~ logavginc + expnstu, data = Caschool)
> lmcavw5 <- lm(formula = testscr ~ logavginc + str, data = Caschool)
> lmcavw6 <- lm(formula = testscr ~ logavginc + elpct, data = Caschool)
>
> summary(lmcavw1)
> summary(lmcavw2)
> summary(lmcavw3)
> summary(lmcavw4)
> summary(lmcavw5)
> summary(lmcavw6)
```

Hier zien we de grootste F-waarde bij het zesde model, dus `elpct` is de tweede predictor in het model.

10. Gelijkaardig als de vorige 2 punten gaan we stuk voor stuk nakijken. De volgende die we erin voegen is `compstu`, en als laatste `grspan`, want hierna zien we dat er geen significante predictoren meer bijkomen. Dus het uiteindelijke model bevat: `logagvinc`, `elpct`, `compstu` en `grspan`. We hebben een totale verklaarde variantie van 71.81%.

11. De interpretatie van het teken van de coëfficiënten is exact hetzelfde als in de vorige oefening, dus gaan we er niet meer op in.

12.
```
> plot(lmacw_finaal)
```

Alle assumpties houden stand (hoewel er een kleine afwijking is bij de residuen die rond nul verdeeld zijn - maar dit is geen al te grote afwijking)

13. De verklaarde variantie van beide oefeningen zijn ongeveer gelijk. Ook zien we een overeenkomst in de predictoren, beide gebruiken het inkomen en de kosten per student.

7 Uitslagen één jaar later

$H_0 : \beta_1 = 0$ versus $H_a : \beta_1 \neq 0$.

```
> lm_uitslag <- lm(formula = uitslag ~ uitslag2, data = uitslagen)
> plot(lm_uitslag)
> summary(lm_uitslag)

Call:
lm(formula = uitslag ~ uitslag2, data = uitslagen)

Residuals:
     Min       1Q   Median       3Q      Max
-10.8524  -4.2991  -0.6522   4.5011  12.2142
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 54.25441    1.32313  41.005   <2e-16 ***
uitslag2    -0.02740    0.02337  -1.172    0.243
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 5.352 on 141 degrees of freedom
Multiple R-squared:  0.009652,        Adjusted R-squared:  0.002628
F-statistic: 1.374 on 1 and 141 DF,  p-value: 0.2431
```

De homoscedasticiteit is voor discussie vatbaar. Indien we het de voordeel van de twijfel geven zien we dat we de nulhypothese niet kunnen verwerpen. Dus dat we de scores van een jaar vroeger niet kunnen gebruiken als predictor.

8 Opleiding en politieke opinie

1. Onze nulhypothese stelt dat de verdeling van politieke opinie dezelfde is voor de drie populaties: $H_0 : \pi_{.1} = \pi_{.2} = \pi_{.3} = \pi_{.4}$. De alternatieve stelt dat er ten minste 1 π anders is, dus dat de verdeling verschillend is voor de populaties

2.
```
> prop.table(table(politiek), margin = 1)
      opleiding
opinie       ped       psy        sw
     A 0.3404255 0.4468085 0.2127660
     B 0.3068182 0.5454545 0.1477273
     C 0.2187500 0.5859375 0.1953125
     D 0.1891892 0.3783784 0.4324324
```

3. Aangezien we gaan zien naar de verdeling over verschillende populaties gebruiken we een χ^2-toets:

```
> chisq.test(table(politiek))

        Pearson's Chi-squared test

data:  table(politiek)
X-squared = 16.785, df = 6, p-value = 0.01011
```

   We bekomen $\chi^2_6 = 16.785; p = 0.01011$, en verwerpen dus de nulhypothese. De verdeling van politieke opinie is verschillend voor de drie populaties.

9 Etnische afkomst en vakbond

Ook hier is geen uniek antwoord omdat je zelf de verdeling mag kiezen, zolang je jezelf maar kan houden aan de opgelegde beperkingen (het verschil van 0.15).

1. Wij hebben gekozen voor volgende verdelingen (als H_a):

```
> prop <-matrix(data = c(0.35, 0.30, 0.20,0.15,
+                        0.20,0.30,0.25,0.25,
```

```
+                         0.20,0.30,0.25,0.25,
+                         0.20,0.30,0.25,0.25), nrow=4, byrow=TRUE)
>
> rownames(prop) <- c("EU", "N-A","M-O", "Andere")
> colnames(prop) <- c("ACV", "ABVV","ACLVB", "Andere")
>
> prop
ACV ABVV ACLVB Andere
EU     0.35  0.3  0.20   0.15
N-A    0.20  0.3  0.25   0.25
M-O    0.20  0.3  0.25   0.25
Andere 0.20  0.3  0.25   0.25
```

2.
```
> prop.k <- prop/dim(prop)[2]
> prop.k
           ACV  ABVV  ACLVB Andere
EU     0.0875 0.075 0.0500 0.0375
N-A    0.0500 0.075 0.0625 0.0625
M-O    0.0500 0.075 0.0625 0.0625
Andere 0.0500 0.075 0.0625 0.0625
> w <- ES.w2(prop.k)
> pwr.chisq.test(w=w, power=0.95, df=(4-1)*(4-1), sig.level=0.05)

     Chi squared power calculation

              w = 0.1675416
              N = 840.3737
             df = 9
      sig.level = 0.05
          power = 0.95

NOTE: N is the number of observations
```

We hebben in ons voorbeeld dus 841 observaties nodig.

Extra informatie
Antwoorden op vragen

F-toets: eenzijdig of tweezijdig?

Blijkbaar is er verwarring ontstaan over de F-toets. We hebben de bundels nog eens nagelezen en hebben gemerkt dat het iets te ambigu uitgelegd is. We zullen eerst de verdeling bestuderen:

F-verdeling

Onze F-verdeling gaat van 0 tot $+\infty$. Deze is nauw verwant met de t-verdeling, maar we weten dat de t-verdeling van $-\infty$ tot $+\infty$ gaat. Dus hoe zijn deze verwant. Hiervoor gaan we naar de meest eenvoudige casus:

- Stel, we hebben een t-verdeling met 50 vrijheidsgraden. Als we dan gaan kijken naar de grenzen waarbinnen 95% ligt krijgen we $[-2.00859; 2.00859]$. We hebben een ondergrens en een bovengrens.
- We willen nu een F-verdeling bestuderen. Stel dat we 1 en 50 vrijheidsgraden hebben. We gaan nu zoeken naar de grenzen waarbinnen 95% ligt. Maar we hebben nu al een ondergrens, namelijk 0. Dus we gaan enkel zoeken naar een bovengrens. Deze bovengrens is 4.034.
- Wat is nu het verband? Als we de onder- of bovengrens gaan nemen van de t-verdeling, en deze kwadrateren dan krijgen we $2.00859^2 = 4.034$, wat hetzelfde is als de bovengrens in de F-verdeling.
  Hoe het zit met een F-verdeling met als df_1 groter dan 1 gaan we hier niet uitleggen omdat dit een zeer wiskundig en complex gegeven is.

Dus we weten nu dat de F-verdeling een eenzijdige verdeling is die overeenkomt met een tweezijdige t-verdeling.

F-toets

Als we nu gaan zien naar de toetsen die we doen dan zien we hoe bovenstaande gegevens kunnen gebruikt worden. Als we 1 predictor hebben dan komt onze p-waarde van de t-toets overeen met die van de F-toets, en zien we ook dat de gekwadrateerde t-waarde hetzelfde is als de F-waarde. Dus we hebben het al kunnen integreren. We weten dat onze nulhypothese stelt dat onze $\beta_1 = 0$, en we hebben dus een tweezijdige toets - dit komt overeen met de logica met boven- en ondergrens van de t-verdeling. Omdat we dezelfde p-waarde krijgen bij de F-toets kunnen we al infereren dat de test ook tweezijdig is. Een F-toets houdt geen rekening met het teken, dus daar maakt het niet uit of β_1 groter of kleiner is dan 0, dat kunnen we niet detecteren met een F-toets, we detecteren enkel de magnitude van de afwijking.

Conclusie

Bij de F-toets gaan we dus zeggen dat het een **tweezijdige toets** is, en dit gaan we berekenen via een **eenzijdige verdeling**. Dus de bekomen p-waarde bij een F-toets moeten we niet meer vermenigvuldigen met 2, want het is al een tweezijdige p-waarde (eigenschap van de **tweezijdige toets**).
Noot: dezelfde redenering geldt bij de χ^2. De verdeling is eenzijdig, de toets is tweezijdig. Dus ook hier vermenigvuldigen we de p-waarde niet meer met 2, want het is al tweezijdig. De inferentie van de χ^2-verdeling laten we hier volledig achterwege omdat ook dit een zeer complex gegeven is, en niets te doen heeft met de cursus.

ANOVA: wat is deze toets, en wanneer gebruiken we deze?

Wanneer we spreken over een ANOVA, dan is dit eigenlijk een andere benaming voor een lineaire regressie met nominale predictoren. ANOVA staat voor "Analysis of Variance". Dit omdat we de variantie tussen en binnen

groepen gaan gebruiken in onze test om te zien of er significante verschillen zijn tussen de groepen. Dus ANOVA is slechts een andere benaming. Omdat ANOVA een veel gebruikte benaming is, maar niet echt gebruikt wordt in de cursus, komt dit verwarrend over, waarvoor onze excuses.

Veel succes met het studeren!

Vragen voor een laatste bundel mogen gesteld worden tot
de deadline gecommuniceerd door het VPPK