

//Deze code is geschreven voor Stata, versie 14.

//De analyses kunnen uitgevoerd worden door de code in een do-file van Stata in te voegen en uit te voeren

//Voor de verschillende simulaties is alleen het bezit van Stata nodig. Voor de overige analyses zijn de afnamedata van de DTT pilot ook vereist

//De verschillende analyses zijn gebaseerd op ofwel gesimuleerde data ofwel data van de DTT pilot afnames. Bij elke aparte analyse is omschreven welke data het betreffen

//Aanvullende uitleg staat doorlopend in dit document aangegeven, vooraf gegaand door '//'

//De indicaties '//' markeren ook de figuren uit het rapport waarbij de daaropvolgende code hoort

//Bij vragen kan contact opgenomen worden met Ron Diris via r.diris@maastrichtuniversity.nl

//FIGUUR 2: SIMULATIE

clear

set seed 999

set matsize 1500

set obs 1800

drawnorm d f g h

gen theta=d

gen id_new=_n

forvalues i=1/200 {

gen uniform`i'=uniform()

gen A`i`_`=0.196+((1.88-0.196)*uniform`i') if id_new==`i'

egen A`i`_`=max(A`i`'_)

drop A`i`'_

gen B`i`_`=(uniform`i`'-0.5)*3 if id_new==`i'

egen B`i`_`=max(B`i`'_)

gen p`i`_`=1/(1+2.71828^(-A`i`_`*(theta-B`i`'_)))

}

```
drop A* B*
```

```
forvalues i=1/200{  
gen q`i'=p`i'>uniform`i'  
}
```

```
drop p* uniform*
```

```
order *, alpha
```

```
order *, seq
```

```
foreach i in 3 5 7 10 15 20 30 40 50 75 100 200{
```

```
quietly irt 2pl q1-q`i', intpoints(7)  
predict beta_latent`i', latent se(se_latent`i')  
egen meanse`i'=mean(se_latent`i')  
gen segi`i'=1.96*meanse`i'  
}
```

```
gen count1=_n
```

```
gen count2=.
```

```
replace count2=3 if count1<=150
```

```
replace count2=5 if count1>150 & count1<=300
```

```
replace count2=7 if count1>300 & count1<=450
```

```
replace count2=10 if count1>450 & count1<=600
```

```
replace count2=15 if count1>600 & count1<=750
```

```
replace count2=20 if count1>750 & count1<=900
```

```
replace count2=30 if count1>900 & count1<=1050
```

```
replace count2=40 if count1>1050 & count1<=1200
```

```
replace count2=50 if count1>1200 & count1<=1350
```

```
replace count2=75 if count1>1350 & count1<=1500
```

```
replace count2=100 if count1>1500 & count1<=1650
replace count2=200 if count1>1650 & count1<=1800
```

```
gen LB=.
gen UB=.
gen beta=.
gen se=.
```

```
foreach i in 3 5 7 10 15 20 30 40 50 75 100 200{
replace UB=seci`i' if count2==`i'
replace LB=-1*seci`i' if count2==`i'
replace beta=beta_latent`i' if count2==`i'
replace se=se_latent`i' if count2==`i'
replace se=se*(1/.876568)
}
```

```
egen beta_std=std(beta)
```

```
gen beta_norm=normal(beta_std)
gen LB_norm=normal(LB)
gen UB_norm=normal(UB)
```

```
replace beta_norm=beta_norm*100
replace LB_norm=LB_norm*100
replace UB_norm=UB_norm*100
```

```
twoway (line LB_norm count2, color(black))(line UB_norm count2, color(black) legend(off)
xtitle("Aantal vragen") ytitle(Geschatte ranking) graphregion(color(white)))
```

```
//FIGUUR 3: simulatie
```

```
clear
```

```
set obs 10000
```

```
global onzekerheid=0
```

```
global omdraaien=0
```

```
drawnorm skill
```

```
if $onzekerheid {
```

```
  gen group=0
```

```
  replace group=1 if _n>=8000
```

```
  replace group=2 if _n>=9000
```

```
  replace skill=0 if group==1
```

```
  replace skill=2 if group==2
```

```
}
```

```
gen sum=0
```

```
forvalues i=1/10 {
```

```
  drawnorm e
```

```
  gen v`i'=skill+e>-1.1+`i'/5
```

```
  replace sum=sum+v`i'
```

```
  drop e
```

```
}
```

```
irt 2pl v*
```

```
predict s1, latent se(ss1)
```

```
forvalues i=11/20 {  
drawnorm e  
gen v`i'=skill+e>-3.1+(`i'-10)/5  
drop e  
}
```

```
forvalues i=21/30 {  
drawnorm e  
gen v`i'=skill+e>0.9+(`i'-20)/5  
drop e  
}
```

```
irt 2pl v*  
predict s2, latent se(ss2)
```

```
irt 2pl v1-v10 v11 v14 v15 v18 v20 v21 v24 v25 v28 v30  
predict s4, latent se(ss4)
```

```
if $omdraaien==0 {  
forvalues i=11/20 {  
replace v`i'=. if s1>=0  
}  
}
```

```
forvalues i=21/30 {  
replace v`i'=. if s1<0  
}  
}
```

```
if $omdraaien==1 {
```

```
forvalues i=11/20 {  
  replace v`i'=. if s1<0  
}
```

```
forvalues i=21/30 {  
  replace v`i'=. if s1>=0  
}  
}
```

```
irt 2pl v*
```

```
predict s3, latent se(ss3)  
gen g=trunc(skill*10)  
replace g=. if skill<-3  
replace g=. if skill>3
```

```
bys g: egen m1=mean(s1)  
bys g: egen m2=mean(s2)  
bys g: egen m3=mean(s3)  
bys g: egen m4=mean(s4)  
bys g: egen ms1=mean(ss1)  
bys g: egen ms2=mean(ss2)  
bys g: egen ms3=mean(ss3)  
bys g: egen ms4=mean(ss4)  
bys g: gen tag=_n==1
```

```
scatter ms3 ms1 ms2 ms4 g if tag, legend(col(4) order(1 "Adaptief (20)" 2 "Niet-adaptief (10)" 3  
"Niet-adaptief (30)" 4 "Niet-adaptief (20)")) graphregion(color(white)) xtitle(Niveau leerling)
```

```
////////////////////////////////////
```

```
//FIGUUR 4: Simulatie
clear

set obs 10000

global onzekerheid=0
global omdraaien=0

drawnorm skill

if $onzekerheid {
gen group=0
replace group=1 if _n>=8000
replace group=2 if _n>=9000

replace skill=0 if group==1
replace skill=2 if group==2
}

gen sum=0

forvalues i=1/10 {
drawnorm e

gen v`i'=skill+2*e>-1.1+`i'/5
replace sum=sum+v`i'
drop e
}

irt 2pl v*
```

```
predict s1, latent se(ss1)
```

```
forvalues i=11/20 {
```

```
drawnorm e
```

```
gen v`i'=skill+2*e>-3.1+(`i'-10)/5
```

```
drop e
```

```
}
```

```
forvalues i=21/30 {
```

```
drawnorm e
```

```
gen v`i'=skill+2*e>0.9+(`i'-20)/5
```

```
drop e
```

```
}
```

```
irt 2pl v*
```

```
predict s2, latent se(ss2)
```

```
irt 2pl v1-v10 v11 v14 v15 v18 v20 v21 v24 v25 v28 v30
```

```
predict s4, latent se(ss4)
```

```
if $omdraaien==0 {
```

```
forvalues i=11/20 {
```

```
replace v`i'=. if s1>=0
```

```
}
```

```
forvalues i=21/30 {
```

```
replace v`i'=. if s1<0
```

```
}
```



```

}
if $omdraaien==1 {
forvalues i=11/20 {
replace v`i'=. if s1<0
}

forvalues i=21/30 {
replace v`i'=. if s1>=0
}
}

irt 2pl v*

predict s3, latent se(ss3)

gen g=trunc(skill*10)
replace g=. if skill<-3
replace g=. if skill>3

bys g: egen m1=mean(s1)
bys g: egen m2=mean(s2)
bys g: egen m3=mean(s3)
bys g: egen m4=mean(s4)
bys g: egen ms1=mean(ss1)
bys g: egen ms2=mean(ss2)
bys g: egen ms3=mean(ss3)
bys g: egen ms4=mean(ss4)
bys g: gen tag=_n==1

scatter ms3 ms1 ms2 ms4 g if tag, legend(col(4) order(1 "Adaptief (20)" 2 "Niet-adaptief (10)" 3
"Niet-adaptief (30)" 4 "Niet-adaptief (20)")) graphregion(color(white)) xtitle(Niveau leerling)

```

```
//FIGUUR 5: Simulatie
```

```
clear
```

```
set seed 113
```

```
set matsize 600
```

```
set obs 4000
```

```
drawnorm d
```

```
gen theta=d
```

```
forvalues i=1/9 {
```

```
gen uniform`i'=uniform()
```

```
gen A`i'=2.5-0.1*uniform`i'
```

```
gen B`i'=-1.5+((3/8)*(`i'-1))
```

```
gen Qp`i'=1/(1+2.71828^(-A`i'*(theta-B`i')))
```

```
}
```

```
drop A* B*
```

```
drop uniform*
```

```
forvalues i=1/9{
```

```
gen uniform`i'=uniform()
```

```
//gen THA`i'=0.45+(((`i'-8)/34)*0.1)
```

```
gen QA`i'=Qp`i'>(.5+(1.5*(uniform`i'-0.5)))
```

```
}
```

```
drop uniform* Qp* d
```

```
forvalues a =1/5 {
```

```

forvalues b =2/6 {
forvalues c =3/7 {
forvalues d =4/8 {
forvalues e =5/9 {

gen q1`a'=QA`a'
gen q2`b'=QA`b'
gen q3`c'=QA`c'
gen q4`d'=QA`d'
gen q5`e'=QA`e'

gen scorepath`a``b``c``d``e'= string(`a') + "," + string(`b') + "," + string(`c') + "," + string(`d') + "," +
string(`e')

drop q1`a' q2`b' q3`c' q4`d' q5`e'

if
`a'==`b'|`a'==`c'|`a'==`d'|`a'==`e'|`b'==`c'|`b'==`d'|`b'==`e'|`c'==`d'|`c'==`e'|`d'==`e'|`a`>`b'|`b`>`c'|`c
`>`d'|`d`>`e' {

drop scorepath`a``b``c``d``e'

}

}

}

}

}

}

}

egen nrcorrect=rsum(QA1-QA9)

gen voldoende=nrcorrect>=5

egen meanvoldoende=mean(voldoende)

gen p0=.5

```

```

forvalues i=1/9{
local j=`i'-1
bys voldoende: egen Vmeanprob_Q`i`_`=mean(QA`i`) if voldoende==1
egen Vmeanprob_Q`i`_`=max(Vmeanprob_Q`i`_)
bys voldoende: egen Omeanprob_Q`i`_`=mean(QA`i`) if voldoende==0
egen Omeanprob_Q`i`_`=max(Omeanprob_Q`i`_)
drop Omeanprob_Q`i`_ Vmeanprob_Q`i`_
}

```

```

foreach var in scorepath12345 scorepath12346 scorepath12347 scorepath12348 scorepath12349
scorepath12356 scorepath12357 scorepath12358 scorepath12359 scorepath12367 scorepath12368
scorepath12369 scorepath12378 scorepath12379 scorepath12389 scorepath12456 scorepath12457
scorepath12458 scorepath12459 scorepath12467 scorepath12468 scorepath12469 scorepath12478
scorepath12479 scorepath12489 scorepath12567 scorepath12568 scorepath12569 scorepath12578
scorepath12579 scorepath12589 scorepath12678 scorepath12679 scorepath12689 scorepath12789
scorepath13456 scorepath13457 scorepath13458 scorepath13459 scorepath13467 scorepath13468
scorepath13469 scorepath13478 scorepath13479 scorepath13489 scorepath13567 scorepath13568
scorepath13569 scorepath13578 scorepath13579 scorepath13589 scorepath13678 scorepath13679
scorepath13689 scorepath13789 scorepath14567 scorepath14568 scorepath14569 scorepath14578
scorepath14579 scorepath14589 scorepath14678 scorepath14679 scorepath14689 scorepath14789
scorepath15678 scorepath15679 scorepath15689 scorepath15789 scorepath16789 scorepath23456
scorepath23457 scorepath23458 scorepath23459 scorepath23467 scorepath23468 scorepath23469
scorepath23478 scorepath23479 scorepath23489 scorepath23567 scorepath23568 scorepath23569
scorepath23578 scorepath23579 scorepath23589 scorepath23678 scorepath23679 scorepath23689
scorepath23789 scorepath24567 scorepath24568 scorepath24569 scorepath24578 scorepath24579
scorepath24589 scorepath24678 scorepath24679 scorepath24689 scorepath24789 scorepath25678
scorepath25679 scorepath25689 scorepath25789 scorepath26789 scorepath34567 scorepath34568
scorepath34569 scorepath34578 scorepath34579 scorepath34589 scorepath34678 scorepath34679
scorepath34689 scorepath34789 scorepath35678 scorepath35679 scorepath35689 scorepath35789
scorepath36789 scorepath45678 scorepath45679 scorepath45689 scorepath45789 scorepath46789
scorepath56789 {
local a =substr(`var',1,1)
local b =substr(`var',3,1)
local c =substr(`var',5,1)
local d =substr(`var',7,1)
local e =substr(`var',9,1)
gen p1_`var'=(Vmeanprob_Q`a`*p0)/((Vmeanprob_Q`a`*p0)+((Omeanprob_Q`a`)*(1-p0))) if QA`a`==1

```

```
replace p1_`var'=((1-Vmeanprob_Q`a')*p0)/(((1-Vmeanprob_Q`a')*p0)+(((1-Omeanprob_Q`a'))*(1-
p0))) if QA`a'==0
```

```
gen p2_`var'=(Vmeanprob_Q`b'*p1_`var')/((Vmeanprob_Q`b'*p1_`var')+((Omeanprob_Q`b')*(1-
p1_`var'))) if QA`b'==1
```

```
replace p2_`var'=((1-Vmeanprob_Q`b')*p1_`var')/(((1-Vmeanprob_Q`b')*p1_`var')+(((1-
Omeanprob_Q`b'))*(1-p1_`var'))) if QA`b'==0
```

```
gen p3_`var'=(Vmeanprob_Q`c'*p2_`var')/((Vmeanprob_Q`c'*p2_`var')+((Omeanprob_Q`c')*(1-
p2_`var'))) if QA`c'==1
```

```
replace p3_`var'=((1-Vmeanprob_Q`c')*p2_`var')/(((1-Vmeanprob_Q`c')*p2_`var')+(((1-
Omeanprob_Q`c'))*(1-p2_`var'))) if QA`c'==0
```

```
gen p4_`var'=(Vmeanprob_Q`d'*p3_`var')/((Vmeanprob_Q`d'*p3_`var')+((Omeanprob_Q`d')*(1-
p3_`var'))) if QA`d'==1
```

```
replace p4_`var'=((1-Vmeanprob_Q`d')*p3_`var')/(((1-Vmeanprob_Q`d')*p3_`var')+(((1-
Omeanprob_Q`d'))*(1-p3_`var'))) if QA`d'==0
```

```
gen p5_`var'=(Vmeanprob_Q`e'*p4_`var')/((Vmeanprob_Q`e'*p4_`var')+((Omeanprob_Q`e')*(1-
p4_`var'))) if QA`e'==1
```

```
replace p5_`var'=((1-Vmeanprob_Q`e')*p4_`var')/(((1-Vmeanprob_Q`e')*p4_`var')+(((1-
Omeanprob_Q`e'))*(1-p4_`var'))) if QA`e'==0
```

```
gen done_na2_`var'=p2_`var'>0.8
```

```
gen done_na3_`var'=p3_`var'>0.8
```

```
gen done_na4_`var'=p4_`var'>0.8
```

```
replace done_na3_`var'=0 if done_na2_`var'==1
```

```
replace done_na4_`var'=0 if done_na2_`var'==1
```

```
replace done_na4_`var'=0 if done_na3_`var'==1
```

```
drop p1_`var' p2_`var' p3_`var' p4_`var'
```

```
gen Op5_`var'=1-p5_`var'
```

```
egen maxp5_`var'=rmax(p5_`var' Op5_`var')
```

```
gen certainty_`var'=maxp5_`var'>0.8
```

```
egen meancertainty_`var`=mean(certainty_`var`)
drop certainty_`var`
gen testtime_`var`=5*75+100*(uniform()-0.5)
replace testtime_`var`=testtime_`var`-75 if done_na4_`var`==1
replace testtime_`var`=testtime_`var`-150 if done_na3_`var`==1
replace testtime_`var`=testtime_`var`-225 if done_na2_`var`==1
egen meantesttime_`var`=mean(testtime_`var`)
}
```

```
order *, alpha
```

```
egen maxmeancertainty=rmax(meancertainty_scorepath12345 meancertainty_scorepath12346
meancertainty_scorepath12347 meancertainty_scorepath12348 meancertainty_scorepath12349
meancertainty_scorepath12356 meancertainty_scorepath12357 meancertainty_scorepath12358
meancertainty_scorepath12359 meancertainty_scorepath12367 meancertainty_scorepath12368
meancertainty_scorepath12369 meancertainty_scorepath12378 meancertainty_scorepath12379
meancertainty_scorepath12389 meancertainty_scorepath12456 meancertainty_scorepath12457
meancertainty_scorepath12458 meancertainty_scorepath12459 meancertainty_scorepath12467
meancertainty_scorepath12468 meancertainty_scorepath12469 meancertainty_scorepath12478
meancertainty_scorepath12479 meancertainty_scorepath12489 meancertainty_scorepath12567
meancertainty_scorepath12568 meancertainty_scorepath12569 meancertainty_scorepath12578
meancertainty_scorepath12579 meancertainty_scorepath12589 meancertainty_scorepath12678
meancertainty_scorepath12679 meancertainty_scorepath12689 meancertainty_scorepath12789
meancertainty_scorepath13456 meancertainty_scorepath13457 meancertainty_scorepath13458
meancertainty_scorepath13459 meancertainty_scorepath13467 meancertainty_scorepath13468
meancertainty_scorepath13469 meancertainty_scorepath13478 meancertainty_scorepath13479
meancertainty_scorepath13489 meancertainty_scorepath13567 meancertainty_scorepath13568
meancertainty_scorepath13569 meancertainty_scorepath13578 meancertainty_scorepath13579
meancertainty_scorepath13589 meancertainty_scorepath13678 meancertainty_scorepath13679
meancertainty_scorepath13689 meancertainty_scorepath13789 meancertainty_scorepath14567
meancertainty_scorepath14568 meancertainty_scorepath14569 meancertainty_scorepath14578
meancertainty_scorepath14579 meancertainty_scorepath14589 meancertainty_scorepath14678
meancertainty_scorepath14679 meancertainty_scorepath14689 meancertainty_scorepath14789
meancertainty_scorepath15678 meancertainty_scorepath15679 meancertainty_scorepath15689
meancertainty_scorepath15789 meancertainty_scorepath16789 meancertainty_scorepath23456
meancertainty_scorepath23457 meancertainty_scorepath23458 meancertainty_scorepath23459
meancertainty_scorepath23467 meancertainty_scorepath23468 meancertainty_scorepath23469
meancertainty_scorepath23478 meancertainty_scorepath23479 meancertainty_scorepath23489
meancertainty_scorepath23567 meancertainty_scorepath23568 meancertainty_scorepath23569
meancertainty_scorepath23578 meancertainty_scorepath23579 meancertainty_scorepath23589
meancertainty_scorepath23678 meancertainty_scorepath23679 meancertainty_scorepath23689
meancertainty_scorepath23789 meancertainty_scorepath24567 meancertainty_scorepath24568)
```


meancertainty_scorepath34689 meancertainty_scorepath34789 meancertainty_scorepath35678
meancertainty_scorepath35679 meancertainty_scorepath35689 meancertainty_scorepath35789
meancertainty_scorepath36789 meancertainty_scorepath45678 meancertainty_scorepath45679
meancertainty_scorepath45689 meancertainty_scorepath45789 meancertainty_scorepath46789
meancertainty_scorepath56789 {

if `var`==maxmeancertainty {

sum `var`

}

}

foreach var in meantesttime_scorepath12345 meantesttime_scorepath12346
meantesttime_scorepath12347 meantesttime_scorepath12348 meantesttime_scorepath12349
meantesttime_scorepath12356 meantesttime_scorepath12357 meantesttime_scorepath12358
meantesttime_scorepath12359 meantesttime_scorepath12367 meantesttime_scorepath12368
meantesttime_scorepath12369 meantesttime_scorepath12378 meantesttime_scorepath12379
meantesttime_scorepath12389 meantesttime_scorepath12456 meantesttime_scorepath12457
meantesttime_scorepath12458 meantesttime_scorepath12459 meantesttime_scorepath12467
meantesttime_scorepath12468 meantesttime_scorepath12469 meantesttime_scorepath12478
meantesttime_scorepath12479 meantesttime_scorepath12489 meantesttime_scorepath12567
meantesttime_scorepath12568 meantesttime_scorepath12569 meantesttime_scorepath12578
meantesttime_scorepath12579 meantesttime_scorepath12589 meantesttime_scorepath12678
meantesttime_scorepath12679 meantesttime_scorepath12689 meantesttime_scorepath12789
meantesttime_scorepath13456 meantesttime_scorepath13457 meantesttime_scorepath13458
meantesttime_scorepath13459 meantesttime_scorepath13467 meantesttime_scorepath13468
meantesttime_scorepath13469 meantesttime_scorepath13478 meantesttime_scorepath13479
meantesttime_scorepath13489 meantesttime_scorepath13567 meantesttime_scorepath13568
meantesttime_scorepath13569 meantesttime_scorepath13578 meantesttime_scorepath13579
meantesttime_scorepath13589 meantesttime_scorepath13678 meantesttime_scorepath13679
meantesttime_scorepath13689 meantesttime_scorepath13789 meantesttime_scorepath14567
meantesttime_scorepath14568 meantesttime_scorepath14569 meantesttime_scorepath14578
meantesttime_scorepath14579 meantesttime_scorepath14589 meantesttime_scorepath14678
meantesttime_scorepath14679 meantesttime_scorepath14689 meantesttime_scorepath14789
meantesttime_scorepath15678 meantesttime_scorepath15679 meantesttime_scorepath15689
meantesttime_scorepath15789 meantesttime_scorepath16789 meantesttime_scorepath23456
meantesttime_scorepath23457 meantesttime_scorepath23458 meantesttime_scorepath23459
meantesttime_scorepath23467 meantesttime_scorepath23468 meantesttime_scorepath23469
meantesttime_scorepath23478 meantesttime_scorepath23479 meantesttime_scorepath23489
meantesttime_scorepath23567 meantesttime_scorepath23568 meantesttime_scorepath23569
meantesttime_scorepath23578 meantesttime_scorepath23579 meantesttime_scorepath23589
meantesttime_scorepath23678 meantesttime_scorepath23679 meantesttime_scorepath23689
meantesttime_scorepath23789 meantesttime_scorepath24567 meantesttime_scorepath24568
meantesttime_scorepath24569 meantesttime_scorepath24578 meantesttime_scorepath24579
meantesttime_scorepath24589 meantesttime_scorepath24678 meantesttime_scorepath24679
meantesttime_scorepath24689 meantesttime_scorepath24789 meantesttime_scorepath25678
meantesttime_scorepath25679 meantesttime_scorepath25689 meantesttime_scorepath25789

```
meantesttime_scorepath26789 meantesttime_scorepath34567 meantesttime_scorepath34568
meantesttime_scorepath34569 meantesttime_scorepath34578 meantesttime_scorepath34579
meantesttime_scorepath34589 meantesttime_scorepath34678 meantesttime_scorepath34679
meantesttime_scorepath34689 meantesttime_scorepath34789 meantesttime_scorepath35678
meantesttime_scorepath35679 meantesttime_scorepath35689 meantesttime_scorepath35789
meantesttime_scorepath36789 meantesttime_scorepath45678 meantesttime_scorepath45679
meantesttime_scorepath45689 meantesttime_scorepath45789 meantesttime_scorepath46789
meantesttime_scorepath56789 {
```

```
if `var'==minmeantesttime {
```

```
sum `var'
```

```
}
```

```
}
```

```
di 75*5
```

```
keep if _n==1
```

```
gen id_new=1
```

```
reshape long scorepath meantesttime_scorepath meancertainty_scorepath, i(id_new) j(Q)
```

```
scatter meantesttime_scorepath meancertainty_scorepath
```

```
//FIGUUR 6 : DTT DATA
```

```
//DEZE ANALYSE IS GEBASEERD OP DE AFNAMEDATA VAN DE DTT PILOT
```

```
//WE WEEKEN HIER VANAF HET ANALYSEBESTAND VAN DE DTT AFNAME (VOOR WISKUNDE IN DE
2016 AFNAME)
```

```
sort id datestamp
```

```
bys id : gen item_chron=_n
```

```
egen id_new = group(id)
```

```
bys id: egen nrq = max(item_chron)
```

```
bys id: egen max_itemchron=max(item_chron)
```

```
egen oplniv_num = group(oplniv)
```

```
foreach var of varlist wp1-e3p3{
```

```
  bys id item_chron: gen `var'_last_=`var' if item_chron==max_itemchron
```

```
  bys id: egen `var'_last=max(`var'_last_)
```

```
  drop `var'_last_
```

```
}
```

```
gen wis_cat=1 if wp1_last>wp2_last & wp1_last>wp3_last
```

```
replace wis_cat=2 if wp2_last>wp1_last & wp2_last>wp3_last
```

```
replace wis_cat=3 if wp3_last>wp1_last & wp3_last>wp2_last
```

```
local X "d0 d1 d2 d3 e0 e1 e2 e3"
```

```
foreach x of local X {
```

```
  gen wis`x'_cat=1 if `x'p1_last>`x'p2_last & `x'p1_last>`x'p3_last
```

```
  replace wis`x'_cat=2 if `x'p2_last>`x'p1_last & `x'p2_last>`x'p3_last
```

```
  replace wis`x'_cat=3 if `x'p3_last>`x'p1_last & `x'p3_last>`x'p2_last
```

```
}
```

```
gen wis_catU=wis_cat
```

```
replace wis_catU=1.5 if wis_catU==1 & wp1_last<0.8
```

```
replace wis_catU=2.5 if wis_catU==2 & wp2_last<0.8
```

```
replace wis_catU=1.5 if wis_catU==2 & wp2_last<0.8 & wp1_last>wp3_last
```

```
replace wis_catU=2.5 if wis_catU==2 & wp2_last<0.8 & wp1_last<wp3_last
```

```
local X "d0 d1 d2 d3 e0 e1 e2 e3"
```

```
foreach x of local X {
```

```

gen wis`x'_catU=wis`x'_cat
replace wis`x'_catU=1.5 if wis`x'_catU==1 & `x'p1_last<0.8
replace wis`x'_catU=2.5 if wis`x'_catU==3 & `x'p3_last<0.8
replace wis`x'_catU=1.5 if wis`x'_catU==2 & `x'p2_last<0.8 & `x'p1_last>`x'p3_last
replace wis`x'_catU=2.5 if wis`x'_catU==2 & `x'p2_last<0.8 & `x'p1_last<`x'p3_last
}

```

//FIGUUR 6:

```

histogram wis_catU, discrete fraction xtitle("Wiskunde (overkoepelend)") ytitle("Aandeel per
categorie") graphregion(color(white)) title("(a)")

histogram wisd0_catU, discrete fraction xtitle("Wiskunde-D") ytitle("Aandeel per categorie")
graphregion(color(white)) title("(b)")

histogram wisd2_catU, discrete fraction xtitle("Wiskunde-D2") ytitle("Aandeel per categorie")
graphregion(color(white)) title("(c)")

```

//FIGUUR 8: DTT data

//Vanaf de code die eerder gebruikt is voor Figuur 6 gaan we eerst verder:

```

gen item_short = substr(item, -6, 3)
destring item_short, replace

//subdomeinen voor de vragen (D1,D2,D3,E1,E2,E3) zitten in de item-code
gen item_domain = substr(item,-2,2)
gen item_level = substr(item,-10,2)

//onderscheiden van vragen voor vmbo vs. havo/vwo
replace item_short = item_short+1000 if item_level=="vm"
replace item_short = item_short+2000 if item_level=="hv"

drop wp1-e3p3 item datestamp attempts wp1_last- e3p3_last

```

```

sort item_short, stable
egen item_ordered = group(item_short)

drop item_ordered

drop score antwoord*
drop scoresub7-scoresub20

//eerste reshape, om subvragen als aparte rij te krijgen
gen n=_n
reshape long scoresub, i(n) j(subvraag)
drop if scoresub==.
replace item_s=item_s*10+subvraag

drop n subvraag

rename scoresub score

//tweede reshape, om alle vragen in kolommen te krijgen
reshape wide duration score item_chron item_domain, i(id) j(item_short)

order *, alpha
order *, seq

order id id2 id_new afnamegroepid brin brinvn datum gebdat geslacht kenmerk1 locatiennaam oplniv
oplniv_num toets vakcode

foreach var of varlist item_domain10031-item_domain23791 {
gsort -`var'
replace `var'=`var'[_n-1] if `var'==""
}

```

```
local X 10031 10051 10061 10071 10081 10131 10132 10133 10151 10152 10161 10171 10191
10192 10201 10221 10222 10241 10271 10272 10281 10282 10283 10284 10331 10351 10401 10511
10512 10513 10514 10551 10561 10581 10591 10601 10621 10622 10631 10651 10661 10662 10681
10721 10722 10741 10791 10831 10832 10833 10841 10842 10843 10861 10881 10911 10921 10931
10932 10941 10951 10961 10971 11101 11111 11121 11141 11151 11152 11161 11271 11281 11291
11311 11351 11361 11362 11363 11364 11391 11392 11401 11421 11461 11511 11541 11591 11631
11661 11671 11691 11692 11693 11751 11851 11861 11901 11921 11931 11951 11961 11962 11991
11992 11993 12081 12091 12111 12112 12161 12162 12171 12172 12173 12174 12181 12191 12211
12212 12221 12231 12232 12233 12234 12261 12262 12263 12264 12321 12331 12361 12401 12402
12403 12404 12421 12471 12501 12511 12521 12522 12531 12581 12621 12651 12661 12662 12671
12701 12751 12821 12881 12951 12981 13041 13042 13101 13131 13141 13142 13143 13221 13231
13232 13233 13234 13235 13291 13341 13431 13441 13481 13491 13492 13581 13611 13641 13651
13652 13653 13654 13661 13671 13761 20071 20091 20092 20101 20121 20311 20521 20541 20611
20621 20622 20641 20642 20643 20651 20671 20681 20691 20741 20742 20791 20811 20812 20813
20814 20821 20961 20991 21021 21061 21091 21101 21141 21151 21161 21221 21231 21232 21233
21234 21235 21251 21261 21311 21331 21401 21441 21521 21571 21572 21573 21574 21601 21611
21651 21681 21781 21782 21801 21811 21812 21813 21831 21881 21891 21892 21951 21961 22051
22521 22561 22611 22612 22613 22614 22621 22651 22652 22653 22654 22655 22656 22731 22741
22742 22743 22744 22771 22801 22821 22831 22861 22981 22982 22983 22984 23011 23021 23071
23091 23191 23192 23193 23194 23195 23196 23201 23241 23261 23281 23282 23291 23321 23322
23371 23431 23491 23501 23531 23561 23562 23581 23582 23601 23602 23603 23621 23651 23721
23791
```

```
foreach name in "D1" "D2" "D3" "E1" "E2" "E3" {
```

```
  foreach x of local X {
```

```
    quietly sum score`x' if item_domain`x'=="`name'"
```

```
    if r(N)!=0 gen `name'score`x' =score`x'
```

```
  }
```

```
}
```

```
drop score*
```

```
drop dura* item_chron* item_domain*
```

```
//sommige irt's zullen in Stata niet werken. In dat geval wordt mplus gebruikt.
```

```
irt grm D1score10051- D1score23261
```

```
predict latD1, latent se(seD1)
```

```
irt grm D2score10131- D2score23291
predict lat_D2_adapt, latent se(se_D2_adapt)
```

```
irt grm D3score10031- D3score23721
predict latD3, latent se(seD3)
```

```
irt grm E1score10061- E1score23282
predict lat_E1_adapt, latent se(se_E1_adapt)
```

```
irt grm E2score10161- E2score23491
predict lat_E2_adapt, latent se(se_E2_adapt)
```

```
irt grm E3score10081- E3score23621
predict lat_E3_adapt, latent se(se_E3_adapt)
```

```
gen sample="16A"
```

```
foreach var of varlist latD1 latD3 lat_all lat_Dall lat_Eall latD2 latE1 latE2 latE3 {
sum `var'
replace `var'=(`var'-r(mean))/r(sd)
}
```

```
//Nu volgt de analyse voor de figuren 8 tot en met 11
```

```
keep if oplniv_num==3
```

```
bys afn: gen nK=_N
```

```
bys brin: gen nS=_N
```

```
gen sesqD2=seD2^2
```

```
bys afn: egen meanKD2= mean(latD2)
bys afn: egen sumKD2se=sum(sesqD2)
gen seD2_klas=sqrt((sumKD2se)/(nK^2))
```

```
bys brin: egen meanSD2= mean(latD2)
bys brin: egen sumSD2se=sum(sesqD2)
gen seD2_school=sqrt((sumSD2se)/(nS^2))
```

```
gen latD2_ubI=latD2+2*seD2
gen latD2_lbI=latD2-2*seD2
```

```
gen latD2_ubK=meanKD2+2*seD2_klas
gen latD2_lbK=meanKD2-2*seD2_klas
```

```
gen latD2_ubS=meanSD2+2*seD2_school
gen latD2_lbS=meanSD2-2*seD2_school
bys brin: egen meanD2cat1 = mean(wisD2_cat==1)
bys brin: egen meanD2cat3 = mean(wisD2_cat==3)
bys brin: egen meanD2cat = mean(wisD2_cat)
```

```
gen x=.
gen y=.
```

```
drawnorm a
replace a=a*2 if a>-2 & a<2
gen b=a
bys a: egen meanb=mean(b)
sort a
```

```
//FIGUUR 8
```



```

twoway (line a meanb, lcolor(black) lpattern(dash))(scatter latD2_ubi latD2_lbi latD2, xline(-.3)
xline(.4) yline(-.3) yline(.4)), xtitle("D2 (continue)") ytitle("D2 (continue)") graphregion(color(white))
ylabel(-4(2)4) xlabel(-4(2)4)

```

```
//FIGUUR 11
```

```

twoway (line a meanb, lcolor(black) lpattern(dash))(scatter latD2_ubi latD2_lbi latD2, xline(-.3)
xline(.4) yline(-.3) yline(.4))(scatter latD2_ubS latD2_lbS meanSD2 if nS>10), xtitle("Latent D2")
ytitle("Latent D2") graphregion(color(white)) ylabel(-4(2)4) xlabel(-4(2)4) legend(label(2 "Upper
bound D2 (I)") label(3 "Lower bound D2 (I)") label(1 "D2") label(4 "Upper bound D2 (S)") label(5
"Lower bound D2 (S)") cols(3) size(small))

```

```
gen sesq_Dall=se_Dall^2
```

```
bys afn: egen meanK_Dall= mean(lat_Dall)
```

```
bys afn: egen sumK_Dallse=sum(sesq_Dall)
```

```
gen se_Dall_klas=sqrt((sumK_Dallse)/(nK^2))
```

```
bys brin: egen meanS_Dall= mean(lat_Dall)
```

```
bys brin: egen sumS_Dallse=sum(sesq_Dall)
```

```
gen se_Dall_school=sqrt((sumS_Dallse)/(nS^2))
```

```
gen lat_Dall_ubi=lat_Dall+2*se_Dall
```

```
gen lat_Dall_lbi=lat_Dall-2*se_Dall
```

```
gen lat_Dall_ubK=meanK_Dall+2*se_Dall_klas
```

```
gen lat_Dall_lbK=meanK_Dall-2*se_Dall_klas
```

```
gen lat_Dall_ubS=meanS_Dall+2*se_Dall_school
```

```
gen lat_Dall_lbS=meanS_Dall-2*se_Dall_school
```

```
sort a
```

```
//FIGUUR 9
```

```

twoway (line a meanb, lcolor(black) lpattern(dash))(scatter lat_Dall_ubi lat_Dall_lbi lat_Dall if
max_itemchron>25, xline(-.7) xline(.85) yline(-.7) yline(.85)), xtitle("Latent D") ytitle("Latent D")

```

```
graphregion(color(white)) ylabel(-4(2)4) xlabel(-4(2)4) legend(label(2 "Upper bound") label (3 "Lower bound") label (1 "Math D") cols(3))
```

```
gen sesq_all=se_all^2
```

```
bys afn: egen meanK_all= mean(lat_all)
```

```
bys afn: egen sumK_allse=sum(sesq_all)
```

```
gen se_all_klas=sqrt((sumK_allse)/(nK^2))
```

```
bys brin: egen meanS_all= mean(lat_all)
```

```
bys brin: egen sumS_allse=sum(sesq_all)
```

```
gen se_all_school=sqrt((sumS_allse)/(nS^2))
```

```
gen lat_all_ubl=lat_all+2*se_all
```

```
gen lat_all_lbl=lat_all-2*se_all
```

```
gen lat_all_ubK=meanK_all+2*se_all_klas
```

```
gen lat_all_lbK=meanK_all-2*se_all_klas
```

```
gen lat_all_ubS=meanS_all+2*se_all_school
```

```
gen lat_all_lbS=meanS_all-2*se_all_school
```

```
sort a
```

```
//FIGUUR 10
```

```
twoway (line a meanb, lcolor(black) lpattern(dash))(scatter lat_all_ubl lat_all_lbl lat_all if  
max_itemchron>25, xline(-.7) xline(.85) yline(-.7) yline(.85)), xtitle("Latent math") ytitle("Latent  
math") graphregion(color(white)) ylabel(-4(2)4) xlabel(-4(2)4) legend(label(2 "Upper bound") label (3  
"Lower bound") label (1 "Math") cols(3))
```

```
gen sesqD1=seD1^2
```

```
bys afn: egen meanKD1= mean(latD1)
```

```
bys afn: egen sumKD1se=sum(sesqD1)
gen seD1_klas=sqrt((sumKD1se)/(nK^2))
```

```
bys brin: egen meanSD1= mean(latD1)
bys brin: egen sumSD1se=sum(sesqD1)
gen seD1_school=sqrt((sumSD1se)/(nS^2))
```

```
gen latD1_ubl=latD1+2*seD1
gen latD1_lbl=latD1-2*seD1
```

```
gen latD1_ubK=meanKD1+2*seD1_klas
gen latD1_lbK=meanKD1-2*seD1_klas
```

```
gen latD1_ubS=meanSD1+2*seD1_school
gen latD1_lbS=meanSD1-2*seD1_school
```

```
bys brin: egen meanDcatM = mean(wisD_cat)
bys brin: egen meanDcat1 = mean(wisD1_cat)
```

```
twoway (scatter meanDcatM meanS_Dall if nS>10)
```

```
//FIGUUR 13
```

```
twoway (scatter meanDcat1 meanSD1 if nS>10), graphregion(color(white)) xtitle("Gemiddelde  
continue") ytitle("Gemiddelde categoriaal")
```

```
//Voor Figuur 14 is gebruik gemaakt van wiskunde uit de 2017 adaptieve afname
```

```
//We specificeren eerst de code die gebruikt is om voor wiskunde 2017 de data om te vormen voor  
de analyses
```

```
sort id datestamp
```

```
bys id: gen item_chron=_n
```

```

bys id: egen nrq = max(item_chron)
bys id: egen max_itemchron=max(item_chron)
egen oplniv_num = group(oplniv)

foreach var of varlist wp1-f3p3{
bys id item_chron: gen `var'_last_=`var' if item_chron==max_itemchron
bys id: egen `var'_last=max(`var'_last_)
drop `var'_last_
}

gen wis_cat=1 if wp1_last>wp2_last & wp1_last>wp3_last
replace wis_cat=2 if wp2_last>wp1_last & wp2_last>wp3_last
replace wis_cat=3 if wp3_last>wp1_last & wp3_last>wp2_last

local X "b0 b1 b2 b3 c0 c1 c2 c3 d0 d1 d2 d3 e0 e1 e2 e3 f0 f1 f2 f3"

foreach x of local X {

gen wis`x'_cat=1 if `x'p1_last>`x'p2_last & `x'p1_last>`x'p3_last
replace wis`x'_cat=2 if `x'p2_last>`x'p1_last & `x'p2_last>`x'p3_last
replace wis`x'_cat=3 if `x'p3_last>`x'p1_last & `x'p3_last>`x'p2_last
}

gen wis_catU=wis_cat

replace wis_catU=1.5 if wis_catU==1 & wp1_last<0.9
replace wis_catU=2.5 if wis_catU==2 & wp2_last<0.9
replace wis_catU=1.5 if wis_catU==2 & wp2_last<0.9 & wp1_last>wp3_last
replace wis_catU=2.5 if wis_catU==2 & wp2_last<0.9 & wp1_last<wp3_last

local X "b0 b1 b2 b3 c0 c1 c2 c3 d0 d1 d2 d3 e0 e1 e2 e3 f0 f1 f2 f3"

```

```

foreach x of local X {
gen wis`x'_catU=wis`x'_cat
replace wis`x'_catU=1.5 if wis`x'_catU==1 & `x'p1_last<0.9
replace wis`x'_catU=2.5 if wis`x'_catU==3 & `x'p3_last<0.9
replace wis`x'_catU=1.5 if wis`x'_catU==2 & `x'p2_last<0.9 & `x'p1_last>`x'p3_last
replace wis`x'_catU=2.5 if wis`x'_catU==2 & `x'p2_last<0.9 & `x'p1_last<`x'p3_last
}

```

```

gen item_short = substr(item, -7, 4)
destring item_short, replace

```

```

gen item_domain = substr(item,-2,2)
gen item_level = substr(item,-10,2)

```

```

replace item_short = item_short+10000 if item_level=="vm"
replace item_short = item_short+20000 if item_level=="hv"

```

```

drop wp1_last- f3p3_last wisb0_catU- wisf3_catU item_sD2_1- item_sD2_6 maxplastb0-
pmax_item68 sdscoresub1 sample count dup D2 D2_inv item_chronD2 D1 D1_inv item_chronD1 D3
D3_inv item_chronD3 antwoord* attempts wp1- f3p3

```

```

drop n
sort id datestamp
gen n=_n
reshape long scoresub, i(n) j(subvraag)
drop if scoresub==.
replace item_s=item_s*10+subvraag

```

```

drop n subvraag
drop score

```

```
rename scoresub score
```

```
drop item datestamp maxduration
```

```
reshape wide duration score item_chron item_domain, i(id) j(item_short)
```

```
order *, alpha
```

```
order *, seq
```

```
gen id_new=_n
```

```
order id id2 id_new afnamegroepid brin datum geslacht locatiennaam oplniv oplniv_num toets  
vakcode
```

```
foreach var of varlist item_domain100031-item_domain220321 {
```

```
gsort -`var'
```

```
replace `var'=`var'[_n-1] if `var'==""
```

```
}
```

```
local X 100031 100051 100061 100071 100081 100101 100111 100121 100131 100132 100133  
100151 100152 100161 100171 100191 100192 100201 100221 100222 100241 100251 100271  
100272 100281 100282 100283 100284 100301 100321 100331 100351 100371 100401 100411  
100421 100422 100451 100461 100481 100491 100501 100511 100512 100513 100514 100541  
100551 100561 100581 100591 100601 100621 100622 100631 100651 100661 100662 100681  
100711 100721 100722 100731 100741 100821 100831 100832 100833 100841 100842 100843  
100861 100871 100881 100891 100892 100893 100911 100921 100931 100932 100941 100951  
100961 100971 100981 100982 100983 100984 101021 101031 101041 101051 101061 101081  
101091 101101 101121 101131 101141 101151 101152 101161 101181 101191 101192 101193  
101194 101221 101251 101261 101271 101281 101291 101311 101351 101361 101362 101363  
101364 101381 101391 101392 101401 101421 101431 101451 101461 101511 101531 101561  
101571 101581 101591 101611 101621 101622 101623 101631 101651 101661 101671 101681  
101691 101692 101693 101701 101711 101751 101781 101851 101861 101881 101901 101921  
101931 101951 101961 101962 101991 101992 101993 102001 102021 102041 102051 102061  
102081 102091 102101 102111 102112 102121 102131 102141 102151 102161 102162 102171  
102172 102173 102174 102181 102191 102211 102212 102221 102231 102232 102233 102234  
102251 102261 102262 102263 102264 102271 102291 102321 102331 102341 102351 102361  
102371 102391 102401 102402 102403 102404 102421 102431 102432 102433 102441 102461  
102471 102501 102511 102521 102522 102531 102541 102551 102571 102572 102573 102581
```

102621 102631 102651 102671 102681 102701 102731 102741 102751 102781 102782 102783
102784 102821 102881 102891 102951 103001 103011 103041 103042 103051 103052 103053
103081 103101 103131 103141 103142 103143 103181 103231 103232 103233 103234 103235
103251 103261 103291 103331 103341 103351 103411 103431 103441 103481 103491 103492
103521 103581 103641 103651 103652 103653 103654 103661 103671 103761 103771 103781
103801 103811 103861 103901 103902 103903 103911 103912 103913 103921 103941 103942
103943 103944 103951 103971 103991 104011 104031 104051 104061 104081 104121 104141
104151 104181 104182 104183 104184 104201 104202 104203 104204 104291 104361 104381
104382 104401 104421 104471 104491 104511 104521 104531 104541 104542 104543 104561
104562 104571 104572 104573 104581 104591 104611 104631 104641 104651 104671 104721
104731 104771 104811 104841 104851 104861 104881 104911 104941 104942 104943 104944
104951 104961 105021 105091 105092 105101 105121 105141 105191 105201 105221 105251
105261 105271 105291 105331 105341 105381 105391 105431 105471 105481 105491 105511
105531 105532 105533 105541 105551 105561 105562 105581 105582 105583 105584 105585
105591 105592 105593 105601 105611 105612 105621 105641 105661 105662 105671 105681
105691 105701 105702 105703 105711 105712 105713 105721 105722 105723 105731 105732
105733 105741 105742 105743 105744 105751 105761 105781 105791 105801 105811 105821
105831 105841 105842 105851 105861 105881 105891 105892 105893 105894 105901 105902
105921 105931 105932 105961 105971 105981 106001 106021 106031 106032 106041 106051
106071 106091 106101 106102 106103 106104 106111 106121 106141 106142 106143 106151
106161 106171 106191 106211 106281 106291 106331 106341 106351 106361 106371 106391
106401 106411 106412 106413 106414 106415 106421 106431 106501 106551 106591 106592
106593 106641 106691 106711 106721 106722 106741 106761 106771 106772 106773 106774
106861 106941 107041 107042 107071 107111 107141 107142 107221 107222 107223 120081
120121 120241 120251 120261 200011 200031 200071 200091 200092 200101 200121 200271
200291 200292 200293 200294 200311 200341 200342 200343 200351 200352 200353 200371
200521 200541 200561 200562 200591 200592 200611 200621 200622 200641 200642 200643
200671 200681 200691 200741 200742 200791 200811 200812 200813 200814 200821 200831
200861 200891 200961 200971 200991 201001 201002 201003 201021 201031 201032 201033
201034 201061 201071 201091 201101 201111 201112 201113 201121 201122 201123 201124
201141 201151 201161 201201 201202 201203 201221 201231 201232 201233 201234 201235
201241 201251 201261 201281 201331 201361 201362 201371 201401 201441 201451 201452
201453 201461 201462 201463 201471 201472 201473 201521 201541 201561 201601 201611
201621 201651 201671 201681 201691 201711 201761 201781 201782 201811 201812 201813
201831 201861 201881 201891 201892 202041 202051 202511 202561 202571 202572 202573
202591 202621 202651 202652 202653 202654 202655 202656 202681 202682 202683 202684
202685 202691 202741 202742 202743 202744 202771 202801 202821 202831 202861 202891
202981 202982 202983 202984 203011 203021 203071 203091 203111 203112 203121 203122
203123 203124 203191 203192 203193 203194 203195 203196 203201 203211 203212 203213
203214 203221 203231 203261 203281 203282 203291 203321 203322 203371 203391 203411
203431 203491 203501 203511 203531 203561 203562 203581 203582 203601 203602 203603
203621 203651 203761 203791 203821 203841 203842 203871 203872 203891 203951 203952
203953 203954 203961 204071 204081 204101 204141 204191 204291 204391 204431 204471
204491 204501 204541 204591 204592 204593 204641 204701 204841 204921 204951 204961
204971 204981 205041 205131 205191 205192 205193 205194 205271 205291 205321 205322
205323 205324 205431 205471 205481 205511 205581 205621 205671 205691 205761 205771
205781 205791 205801 205811 205831 205841 205901 205911 205961 205971 205991 206021

206031 206121 206122 206201 206211 206221 206222 206223 206241 206281 206311 206321
206361 206481 206721 206941 207271 220221 220231 220301 220311 220321

```
foreach name in "B1" "B2" "B3" "C1" "C2" "C3" "D1" "D2" "D3" "E1" "E2" "E3" "F1" "F2" "F3"{
```

```
  foreach x of local X {
```

```
    quietly sum score`x' if item_domain`x'=="`name'"
```

```
    if r(N)!=0 gen `name'score`x' =score`x'
```

```
  }
```

```
}
```

```
drop score*
```

```
drop dura* item_chron* item_domain*
```

```
irt grm B1score100111- B1score106941
```

```
predict latB1_vmbo, latent se(seB1_vmbo)
```

```
irt grm B1score201541- B1score206321
```

```
predict latB1_hv, latent se(seB1_hv)
```

```
irt grm B2score100981- B2score206211
```

```
predict latB2, latent se(seB2)
```

```
irt grm B3score100251- B3score206721
```

```
predict latB3, latent se(seB3)
```

```
irt grm C1score100101- C1score205831
```

```
predict latC1, latent se(seC1)
```

```
irt grm C2score101191- C2score206941
```

```
predict latC2, latent se(seC2)
```


irt grm C3score100301- C3score207271
predict latC3, latent se(seC3)

irt grm D1score100051- D1score220321
predict latD1, latent se(seD1)

irt grm D2score100131- D2score204641
predict latD2, latent se(seD2)

irt grm D3score100031- D3score220231
predict latD3, latent se(seD3)

irt grm E1score100061- E1score204701
predict latE1, latent se(seE1)

irt grm E2score100161- E2score203872
predict latE2, latent se(seE2)

irt grm E3score100081- E3score205041
predict latE3, latent se(seE3)

irt grm F1score200011- F1score206361
predict latF1, latent se(seF1)

irt grm F2score200341- F2score204491
predict latF2, latent se(seF2)

irt grm F3score201361- F3score205961
predict latF3, latent se(seF3)

```
gen sample="17A"
```

```
drop if oplniv=="
```

```
gen lat_all=.
```

```
replace lat_all=latwisvmA_all if leerweg=="VMBO-BB"|leerweg=="VMBO-KB"
```

```
replace lat_all=latwisvmB_all if leerweg=="VMBO-GL"|leerweg=="VMBO-TL"
```

```
replace lat_all=latwishv_all if oplniv!="VO-VMBO"
```

```
gen se_all=.
```

```
replace se_all=sewisvmA_all if leerweg=="VMBO-BB"|leerweg=="VMBO-KB"
```

```
replace se_all=sewisvmB_all if leerweg=="VMBO-GL"|leerweg=="VMBO-TL"
```

```
replace se_all=sewishv_all if oplniv!="VO-VMBO"
```

```
drop latwisvmA_all sewisvmA_all latwisvmB_all sewisvmB_all latwishv_all sewishv_all
```

```
rename latD1 lat_D1
```

```
rename latF1 lat_F1
```

```
rename latF3 lat_F3
```

```
rename seD1 se_D1
```

```
rename seF1 se_F1
```

```
rename seF3 se_F3
```

```
order *, alpha
```

```
order *, seq
```

```
order id id2 id_new afnamegroepid brin datum geslacht locatiennaam oplniv oplniv_num toets  
vakcode leerweg max_itemchron nrq sample
```

```
gen track=.
```

```
replace track=1 if leerweg=="VMBO-BB"
```

```
replace track=2 if leerweg=="VMBO-KB"
```

```
replace track=3 if leerweg=="VMBO-GL"|leerweg=="VMBO-TL"
```

```
replace track=4 if oplniv=="VO-HAVO"
```

```
replace track=5 if oplniv=="VO-VWO"
```

```
bys brin track: gen ssize=_N
```

```
bys brin track: gen stag=_n==1
```

```
keep if oplniv_num==3
```

```
dropmiss, force
```

```
drop B1score201541-F3score205781
```

```
bys brin: egen meanwisF3S_cont=mean(lat_F3)
```

```
bys brin: egen meanwisF3S_cat=mean(wisf3_cat)
```

```
bys brin: egen meanwisF3S_cat1=mean(wisf3_cat==1)
```

```
bys brin: egen meanwisF3S_cat3=mean(wisf3_cat==3)
```

```
egen schoolid=group(brin)
```

```
bys brin: gen nS=_N
```

```
//FIGUUR 14
```

```
twoway (scatter meanwisF3S_cat1 meanwisF3S_cont, mlabel(schoolid)) (scatter meanwisF3S_cat3  
meanwisF3S_cont, mlabel(schoolid)) if nS>10, graphregion(color(white)) xtitle("Gemiddelde  
continue") ytitle("Aandeel laag/hoog") legend(order(1 "Aandeel onder" 2 "Aandeel boven"))
```

```
//VOOR DE FIGUREN 15 T/M 18 IS OOK DE 2017 AFNAME GEBRUIKT, MAAR DAN VOOR NEDERLANDS
```

```
//WE SPECIFICEREN EERST DE CODE DIE NODIG IS OM DE DATA OM TE ZETTEN VOOR DIRECT  
GEBRUIK
```

```
sort id datestamp
```

```
bys id: gen item_chron=_n
```

```
drop if np1==. & item_chron!=1
```

```

bys id: egen nrq = max(item_chron)
bys id: egen max_itemchron=max(item_chron)
egen oplniv_num = group(oplniv)

foreach var of varlist np1- n43p3{
bys id item_chron: gen `var'_last_=`var' if item_chron==max_itemchron
bys id: egen `var'_last=max(`var'_last_)
drop `var'_last_
}

gen ned_cat=1 if np1_last>np2_last & np1_last>np3_last
replace ned_cat=2 if np2_last>np1_last & np2_last>np3_last
replace ned_cat=3 if np3_last>np1_last & np3_last>np2_last

local X "n10 n11 n12 n13 n20 n21 n22 n23 n30 n31 n32 n33 n40 n41 n42 n43"

foreach x of local X {

gen ned`x'_cat=1 if `x'p1_last>`x'p2_last & `x'p1_last>`x'p3_last
replace ned`x'_cat=2 if `x'p2_last>`x'p1_last & `x'p2_last>`x'p3_last
replace ned`x'_cat=3 if `x'p3_last>`x'p1_last & `x'p3_last>`x'p2_last
}

gen ned_catU=ned_cat

replace ned_catU=1.5 if ned_catU==1 & np1_last<0.9
replace ned_catU=2.5 if ned_catU==2 & np2_last<0.9
replace ned_catU=1.5 if ned_catU==2 & np2_last<0.9 & np1_last>np3_last
replace ned_catU=2.5 if ned_catU==2 & np2_last<0.9 & np1_last<np3_last

```

```
local X "n10 n11 n12 n13 n20 n21 n22 n23 n30 n31 n32 n33 n40 n41 n42 n43"
```

```
foreach x of local X {
```

```
gen ned`x'_catU=ned`x'_cat
```

```
replace ned`x'_catU=1.5 if ned`x'_catU==1 & `x'p1_last<0.9
```

```
replace ned`x'_catU=2.5 if ned`x'_catU==3 & `x'p3_last<0.9
```

```
replace ned`x'_catU=1.5 if ned`x'_catU==2 & `x'p2_last<0.9 & `x'p1_last>`x'p3_last
```

```
replace ned`x'_catU=2.5 if ned`x'_catU==2 & `x'p2_last<0.9 & `x'p1_last<`x'p3_last
```

```
}
```

```
replace item="ITM-dtt-ne-hv-0218-SC" if item=="ITM-dtt-ne-hv-0218-SC_NIEUW"
```

```
replace item="ITM-dtt-ne-hv-0219-SC" if item=="ITM-dtt-ne-hv-0219-SC_NIEUW"
```

```
gen subitem = substr(item, -2, 1)=="-"
```

```
gen subitem2 = substr(item, -2, 2)
```

```
gen item_short = substr(item, -7, 4)
```

```
replace item_short = substr(item, -9, 4) if subitem==1
```

```
destring item_short, replace
```

```
replace item_s=452 if item_s==8064
```

```
replace item_s=453 if item_s==8083
```

```
replace item_s=454 if item_s==8098
```

```
replace item_s=455 if item_s==8112
```

```
replace item_s=456 if item_s==8142
```

```
replace item_s=457 if item_s==8162
```

```
replace item_s=458 if item_s==8180
```

```
replace item_s=459 if item_s==8223
```

```
replace item_s=4560 if item_s==9093
```

```
replace item_s=234 if item_s==3158
```

```
replace item_s=235 if item_s==8017
replace item_s=236 if item_s==8054
replace item_s=237 if item_s==8092
replace item_s=238 if item_s==8146
replace item_s=239 if item_s==8150
replace item_s=240 if item_s==8178
```

```
replace item_short=item_short*10
replace item_short=item_short+1 if subitem2=="-a"
replace item_short=item_short+2 if subitem2=="-b"
replace item_short=item_short+3 if subitem2=="-c"
replace item_short=item_short+4 if subitem2=="-d"
```

```
gen item_level = substr(item,-10,2)
replace item_level=substr(item,-12,2) if item_level=="-0"
```

```
replace item_short = item_short+10000 if item_level=="vm"
replace item_short = item_short+20000 if item_level=="hv"
```

```
gen item_domain=""
```

```
xtset id item_chron
```

```
foreach name in "n11p1" "n12p1" "n13p1" "n21p1" "n22p1" "n23p1" "n31p1" "n32p1" "n33p1"
"n41p1" "n42p1" "n43p1"{
replace item_domain ="`name'" if D.`name'!=0 & D.`name'!=.
}

```

```
foreach name in "n11p1" "n12p1" "n13p1" "n21p1" "n22p1" "n23p1" "n31p1" "n32p1" "n33p1"
"n41p1" "n42p1" "n43p1"{
replace item_domain ="`name'" if `name'!=. & D.`name'!=.
}

```

```
egen item_domain_num=group(item_domain)
bys item: egen maxitem_domain_num=max(item_domain_num)
replace item_domain_num=maxitem_domain_num if item_domain_num==.
```

```
replace item_domain="n11p1" if item_domain_num==1
replace item_domain="n12p1" if item_domain_num==2
replace item_domain="n13p1" if item_domain_num==3
replace item_domain="n21p1" if item_domain_num==4
replace item_domain="n22p1" if item_domain_num==5
replace item_domain="n23p1" if item_domain_num==6
replace item_domain="n31p1" if item_domain_num==7
replace item_domain="n32p1" if item_domain_num==8
replace item_domain="n33p1" if item_domain_num==9
replace item_domain="n41p1" if item_domain_num==10
replace item_domain="n42p1" if item_domain_num==12
replace item_domain="n43p1" if item_domain_num==12
```

```
drop if item_domain=="
```

```
drop nrq item_chron max_itemchron
sort id datestamp
bys id: gen item_chron=_n
bys id: egen nrq = max(item_chron)
bys id: egen max_itemchron=max(item_chron)
```

```
drop np1-n43p3 item datestamp attempts np1_last-n43p3_last item_domain_num
maxitem_domain_num
```

```
sort item_short, stable
```

```
drop score antwoord* ned_catU nedn10_catU- nedn43_catU datum toets locatiennaam maxduration
dup gebdat subitem subitem2 max_itemchron item_level kenmerk1 sumscoresub
```

```
//eerste reshape, om subvragen als aparte rij te krijgen
```

```
gen n=_n
```

```
reshape long scoresub, i(n) j(subvraag)
```

```
drop if scoresub==.
```

```
replace item_s=item_s*100+subvraag
```

```
drop n subvraag
```

```
rename scoresub score
```

```
drop duration
```

```
reshape wide score item_chron item_domain, i(id) j(item_short)
```

```
order *, alpha
```

```
gen id_new=_n
```

```
order id id2 id_new afnamegroepid brin geslacht oplniv oplniv_num
```

```
foreach var of varlist item_domain1002001- item_domain2239003 {
```

```
gsort -`var'
```

```
replace `var'=`var'[_n-1] if `var'==""
```

```
}
```

```
foreach var of varlist item_domain1002001- item_domain2239003 {
```

```
replace `var'=substr(`var', 1, 3)
```

```
}
```

```
local X 1002001 1002002 1002003 1002004 1003001 1005001 1005002 1005003 1005004 1006001
1006002 1006003 1007001 1008001 1008002 1008003 1008004 1008005 1008006 1011001
```


1011002 1011003 1012001 1014001 1014002 1014003 1014004 1014005 1014006 1015001
1015002 1015003 1015004 1015005 1015006 1015007 1015008 1018001 1018002 1018003
1018004 1018005 1018006 1019001 1020001 1020002 1020003 1020004 1020005 1020006
1020007 1020008 1022001 1022002 1022003 1022004 1024001 1024002 1024003 1024004
1024005 1024006 1036001 1036002 1036003 1036004 1036005 1038001 1038002 1038003
1038004 1038005 1038006 1038007 1038008 1038009 1045001 1045002 1045003 1045004
1045005 1045006 1045007 1048001 1048002 1048003 1048004 1048005 1061001 1061002
1061003 1061004 1061005 1062001 1062002 1062003 1062004 1062005 1063001 1063002
1063003 1065001 1065002 1065003 1065004 1067001 1067002 1067003 1067004 1067005
1068001 1068002 1068003 1068004 1069001 1069002 1069003 1069004 1081001 1081002
1081003 1081004 1082001 1082002 1082003 1082004 1082005 1082006 1082007 1082008
1082009 1082010 1082011 1085001 1096001 1096002 1096003 1096004 1096005 1096006
1096007 1096008 1096009 1096010 1097001 1097002 1097003 1097004 1097005 1097006
1097007 1097008 1097009 1097010 1100001 1100002 1100003 1100004 1100005 1100006
1100007 1107001 1107002 1107003 1109001 1109002 1109003 1109004 1109005 1109006
1110001 1110002 1110003 1110004 1110005 1110006 1110007 1110008 1110009 1111001
1111002 1111003 1111004 1115001 1115002 1116001 1116002 1116003 1116004 1116005
1119001 1125001 1130001 1133001 1133002 1134001 1134002 1134003 1134004 1134005
1134006 1134007 1134008 1134009 1134010 1135001 1136001 1136002 1136003 1139001
1139002 1139003 1139004 1140001 1140002 1140003 1141001 1161001 1161002 1161003
1169001 1169002 1169003 1169004 1169005 1170001 1170002 1170003 1171001 1171002
1171003 1171004 1171005 1172001 1172002 1172003 1172004 1172005 1172006 1172007
1173001 1174001 1174002 1174003 1174004 1174005 1177001 1177002 1177003 1177004
1178001 1178002 1178003 1178004 1178005 1179001 1181001 1182001 1182002 1182003
1182004 1182005 1182006 1183001 1183002 1183003 1183004 1183005 1183006 1184001
1184002 1184003 1184004 1184005 1184006 1184007 1185001 1185002 1185003 1185004
1185005 1185006 1185007 1186001 1186002 1187001 1187002 1187003 1187004 1188001
1188002 1189001 1189002 1190001 1190002 1190003 1190004 1191001 1192001 1193001
1193002 1193003 1194001 1194002 1194003 1195001 1195002 1195003 1195004 1195005
1197001 1198001 1199001 1199002 1199003 1200001 1200002 1200003 1200004 1201001
1202001 1202002 1202003 1202004 1202005 1202006 1202007 1202008 1202009 1202010
1203001 1203002 1203003 1204001 1205001 1205002 1205003 1205004 1205005 1205006
1206001 1206002 1206003 1206004 1207001 1207002 1207003 1207004 1208001 1208002
1208003 1209001 1209002 1209003 1209004 1210001 1210002 1210003 1210004 1211001
1212001 1213001 1213002 1213003 1213004 1213005 1214001 1214002 1214003 1214004
1214005 1215001 1216001 1217001 1219001 1221001 1222001 1222002 1222003 1224001
1224002 1224003 1225001 1225002 1225003 1226001 1230001 1230002 1230003 1230004
1231001 1231002 1231003 1231004 1232001 1233001 1233002 1235001 1236001 1236002
1240001 1240002 1240003 1240004 1240005 1240006 1240007 1241001 1242001 1242002
1242003 1242004 1244001 1244002 1245001 1245002 1245003 1245004 1245005 1245006
1246001 1246002 1246003 1246004 1247001 1247002 1247003 1247004 1247005 1248001
1249001 1249002 1249003 1249004 1249005 1252001 1252002 1252003 1252004 1252005
1254001 1254002 1254003 1254004 1254005 1255001 1255002 1255003 1255004 1255005
1255006 1255007 1256001 1256002 1256003 1257001 1258001 1258002 1258003 1258004
1259001 1260001 1262001 1262002 1262003 1262004 1262005 1263001 1263002 1263003
1263004 1263005 1264001 1264002 1264003 1264004 1264005 1265001 1265002 1265003
1265004 1265005 1266001 1266002 1266003 1267001 1268001 1268002 1268003 1268004
1268005 1269001 1269002 1270001 1270002 1270003 1270004 1271001 1271002 1271003

1271004 1272001 1272002 1272003 1272004 1272005 1273001 1273002 1273003 1273004
1273005 1274001 1274002 1274003 1274004 1274005 1275001 1275002 1276001 1276002
1276003 1276004 1278001 1279001 1282001 1282002 1282003 1282004 1284001 1284002
1284003 1285001 1285002 1285003 1286001 1286002 1287001 1288001 1289001 1289002
1289003 1290001 1290002 1291001 1291002 1292001 1292002 1293001 1294001 1295001
1295002 1296001 1297001 1297002 1297003 1297004 1297005 1298001 1299001 1299002
1299003 1300001 1310001 1310002 1310003 1310004 1310005 1310006 1311001 1311002
1311003 1311004 1312001 1313001 1313002 1313003 1313004 1313005 1314001 1314002
1314003 1317001 1317002 1317003 1317004 1319001 1319002 1319003 1320001 1320002
1320003 1320004 1323001 1323002 1323003 1323004 1323005 1324001 1324002 1324003
1324004 1324005 1325001 1326001 1326002 1326003 1326004 1326005 1329001 1329002
1329003 1330001 1330002 1331001 1331002 1331003 1332001 1332002 1332003 1333001
1334001 1334002 1334003 1334004 1334005 1334006 1334007 1334008 1337001 1337002
1337003 1337004 1337005 1338001 1338002 1338003 1338004 1339001 1339002 1340001
1340002 1341001 1341002 1342001 1342002 1344001 1344002 1345001 1345002 1345003
1347001 1347002 1350001 1353001 1353002 1353003 1353004 1353005 1354001 1355001
1355002 1357001 1360001 1360002 1360003 1360004 1361001 1363001 1363002 1363003
1363004 1364001 1364002 1366001 1366002 1366003 1368001 1368002 1369001 1370001
1370002 1370003 1372001 1372002 1372003 1372004 1373001 1373002 1373003 1373004
1375001 1375002 1375003 1375004 1377001 1377002 1377003 1377004 1377005 1378001
1379001 1379002 1379003 1379004 1379005 1379006 1380001 1380002 1380003 1380004
1380005 1381001 1383001 1383002 1383003 1383004 1384001 1384002 1384003 1387001
1388001 1389001 1390001 1390002 1390003 1390004 1390005 1390006 1390007 1390008
1391001 1391002 1391003 1392001 1392002 1392003 1393001 1393002 1393003 1393004
1393005 1394001 1395001 1396001 1396002 1396003 1396004 1396005 1396006 1397001
1397002 1397003 1397004 1397005 1398001 1398002 1398003 1398004 1398005 1400001
1400002 1400003 1402001 1404001 1408001 1409001 1411001 1415001 1417001 1417002
1419001 1419002 1419003 1420001 1420002 1421001 1422001 1422002 1422003 1422004
1422005 1422006 1422007 1422008 1423001 1423002 1423003 1423004 1424001 1424002
1424003 1425001 1425002 1425003 1425004 1425005 1426001 1426002 1426003 1426004
1426005 1426006 1427001 1427002 1427003 1427004 1427005 1428001 1428002 1428003
1428004 1428005 1428006 1428007 1429001 1429002 1429003 1429004 1429005 1429006
1430001 1430002 1430003 1430004 1430005 1430006 1431001 1431002 1431003 1431004
1431005 1432001 1432002 1432003 1432004 1432005 1433001 1433002 1433003 1433004
1435001 1435002 1435003 1435004 1436001 1436002 1436003 1437001 1437002 1437003
1437004 1437005 1437006 1437007 1437008 1439001 1439002 1439003 1439004 1439005
1441001 1441002 1442001 1442002 1442003 1442004 1442005 1442006 1442007 1442008
1442009 1443001 1443002 1443003 1443004 1443005 1443006 1443007 1443008 1444001
1444002 1445001 1445002 1445003 1445004 1446001 1446002 1446003 1446004 1447001
1447002 1447003 1447004 1447005 1447006 1448001 1448002 1448003 1448004 1449001
1449002 1449003 1449004 1450001 1450002 1450003 1450004 1450005 1451001 1451002
1451003 1451004 1451005 1453001 1453002 1453003 1453004 1453005 1453006 1453007
1454001 1454002 1454003 1454004 1454005 1454006 1454007 1455001 1455002 1455003
1455004 1456001 1456002 1456003 1458001 1459001 1459002 1459003 1459004 1459005
1459006 1459007 1459008 2012001 2012002 2012003 2012004 2012005 2012006 2012007
2012008 2012009 2012010 2014001 2014002 2014003 2015001 2016001 2018001 2019001
2020001 2021001 2021002 2021003 2022001 2022002 2022003 2022004 2022005 2024001
2024002 2024003 2025001 2025002 2026001 2026002 2026003 2026004 2026005 2026006

2027001 2027002 2027003 2027004 2028001 2028002 2028003 2029001 2030001 2031001
2031002 2031003 2031004 2031005 2031006 2032001 2032002 2032003 2032004 2032005
2032006 2033001 2034001 2034002 2034003 2034004 2035001 2036001 2040101 2040201
2040301 2041001 2043101 2043201 2043301 2043401 2044001 2045001 2047001 2047002
2047003 2047004 2047005 2047006 2047007 2047008 2047009 2048001 2048002 2048003
2048004 2048005 2048006 2049001 2049002 2053001 2053002 2053003 2056001 2056002
2056003 2056004 2056005 2056006 2056007 2056008 2056009 2056010 2058001 2058002
2058003 2058004 2061001 2061002 2061003 2061004 2063001 2063002 2063003 2064001
2064002 2064003 2065001 2065002 2065003 2065004 2066001 2066002 2067001 2068001
2069001 2071001 2072001 2072002 2072003 2072004 2072005 2073001 2074001 2078001
2078002 2078003 2078004 2078005 2078006 2078007 2079001 2079002 2079003 2079004
2079005 2079006 2079007 2079008 2080001 2080002 2080003 2081001 2081002 2081003
2081004 2081005 2082201 2082301 2087001 2087002 2087003 2087004 2087005 2087006
2087007 2087008 2087009 2087010 2087011 2087012 2088101 2088102 2088201 2089101
2089102 2089103 2089104 2089105 2089106 2089107 2089201 2089202 2089203 2089204
2089205 2089206 2090001 2090002 2090003 2090004 2090005 2091001 2091002 2091003
2094001 2094002 2094003 2094004 2095001 2095002 2095003 2095004 2098001 2099001
2103001 2103002 2103003 2103004 2105001 2105002 2105003 2105004 2105005 2105006
2105007 2105008 2105009 2107101 2107201 2109001 2109002 2109003 2109004 2109005
2109006 2109007 2109008 2109009 2109010 2112001 2112002 2112003 2112004 2112005
2113101 2113201 2114101 2114201 2115001 2115002 2115003 2115004 2115005 2116001
2116002 2116003 2116004 2117001 2117002 2117003 2117004 2117005 2118001 2118002
2118003 2118004 2119001 2119002 2119003 2119004 2119005 2119006 2120001 2120002
2120003 2120004 2120005 2120006 2120007 2120008 2120009 2120010 2120011 2121001
2121002 2121003 2121004 2121005 2121006 2122001 2122002 2122003 2122004 2122005
2124001 2124002 2124003 2124004 2124005 2124006 2124007 2124008 2125101 2125201
2127001 2127002 2127003 2127004 2127005 2129001 2129002 2129003 2130001 2130002
2130003 2130004 2130005 2131001 2131002 2131003 2131004 2132001 2133001 2134001
2135101 2135201 2137001 2138001 2138002 2138003 2141001 2141002 2144101 2144201
2148001 2148002 2148003 2154001 2154002 2155001 2155002 2156001 2156002 2156003
2156004 2156005 2156006 2156007 2156008 2157001 2157002 2157003 2157004 2157005
2157006 2157007 2159001 2159002 2159003 2159004 2159005 2159006 2160001 2160002
2160003 2160004 2160005 2160006 2160007 2162001 2162002 2162003 2162004 2163001
2163002 2163003 2165001 2166201 2167001 2168001 2170001 2170002 2170003 2171001
2171002 2171003 2171004 2172001 2172002 2172003 2172004 2173001 2173002 2173003
2173004 2174001 2174002 2174003 2174004 2174005 2174006 2175001 2175002 2175003
2175004 2175005 2176001 2176002 2176003 2176004 2177001 2177002 2177003 2177004
2177005 2179001 2179002 2179003 2180001 2180002 2180003 2181001 2181002 2182001
2182002 2183001 2184001 2184002 2184003 2184004 2184005 2184006 2184007 2186001
2186002 2186003 2186004 2186005 2186006 2187001 2187002 2187003 2188001 2189001
2189002 2189003 2189004 2193001 2193002 2193003 2193004 2193005 2193006 2194001
2195001 2195002 2195003 2195004 2195005 2196001 2196002 2196003 2196004 2196005
2196006 2196007 2196008 2198001 2198002 2198003 2200001 2200002 2200003 2200004
2200005 2200006 2200007 2200008 2200009 2201001 2201002 2202001 2202002 2202003
2202004 2202005 2203101 2203102 2203103 2203104 2203105 2203106 2204001 2205101
2205201 2205301 2212001 2212002 2212003 2212004 2212005 2212006 2212007 2215101
2215201 2216001 2220001 2220002 2220003 2220004 2220005 2221001 2221002 2221003
2222001 2222002 2222003 2222004 2224001 2225001 2227001 2227002 2228001 2229001

2230001 2230002 2230003 2231001 2233001 2235001 2236001 2236002 2236003 2236004
2236005 2236006 2236007 2236008 2236009 2236010 2237001 2237002 2237003 2239001
2239002 2239003

```
foreach name in "n11" "n12" "n13" "n21" "n22" "n23" "n31" "n32" "n33" "n41" "n42" "n43"{
```

```
  foreach x of local X {
```

```
    quietly sum score`x' if item_domain`x'=="`name'"
```

```
    if r(N)!=0 gen `name'score`x' =score`x'
```

```
  }
```

```
}
```

```
drop score*
```

```
drop item_chron* item_domain*
```

```
drop n42score1455004 n42score1139002 n42score1139003 n42score1139004 n42score1081003  
n42score1081004
```

```
irt grm n11score1015001-n11score2236010
```

```
predict latn11, latent se(sen11)
```

```
irt grm n12score1002001-n12score2235001
```

```
predict latn12, latent se(sen12)
```

```
irt grm n13score1008001-n13score2233001
```

```
predict latn13, latent se(sen13)
```

```
irt grm n21score1067001-n21score2215201
```

```
predict latn21, latent se(sen21)
```

```
irt grm n22score1003001-n22score2216001
```

```
predict latn22, latent se(sen22)
```

```
irt grm n23score1022001-n23score2201002  
predict latn23, latent se(sen23)
```

```
irt grm n31score1063001-n31score2239003  
predict latn31, latent se(sen31)
```

```
irt grm n32score1018001-n32score2237003  
predict latn32, latent se(sen32)
```

```
irt grm n33score1036001-n33score2231001  
predict latn33, latent se(sen33)
```

```
irt grm n41score1014001-n41score2196008  
predict latn41, latent se(sen41)
```

```
irt grm n42score1038001-n42score2202005  
predict latn42, latent se(sen42)
```

```
irt grm n43score1005001-n43score2236010  
predict latn43, latent se(sen43)
```

```
gen sample="17A"
```

```
gen latall=.
```

```
replace latall=latnedvmB_all if leerweg=="VMBO-BB"
```

```
replace latall=latnedvmK_all if leerweg=="VMBO-KB"
```

```
replace latall=latnedvmG_all if leerweg=="VMBO-GL"|leerweg=="VMBO-TL"
```

```
replace latall=latnedH_all if oplniv=="VO-HAVO"
```

```
replace latall=latnedvwo_all if oplniv=="VO-VWO"
```

```
gen seall=.
```

```
replace seall=senedvmB_all if leerweg=="VMBO-BB"
```

```
replace seall=senedvmK_all if leerweg=="VMBO-KB"
replace seall=senedvmG_all if leerweg=="VMBO-GL"|leerweg=="VMBO-TL"
replace seall=senedH_all if oplniv=="VO-HAVO"
replace seall=senedvwo_all if oplniv=="VO-VWO"

drop latnedvmB_all senedvmB_all latnedvmK_all senedvmK_all latnedvmG_all senedvmG_all
latnedH_all senedH_all latnedvwo_all senedvwo_all

drop if oplniv=="

rename latn11 latned11
rename latn12 latned12
rename latn21 latned21
rename latn31 latned31
rename sen11 sened11
rename sen12 sened12
rename sen21 sened21
rename sen31 sened31

order *, alpha
order *, seq
order id id2 id_new afnamegroepid brin geslacht oplniv oplniv_num vakcode leerweg nrq sample
```

```
//HIERNA VOLGT DE ANALYSE VOOR DE SCHOOLRAPPORTAGES
```

```
gen track=.
replace track=1 if leerweg=="VMBO-BB"
replace track=2 if leerweg=="VMBO-KB"
replace track=3 if leerweg=="VMBO-GL"|leerweg=="VMBO-TL"
replace track=4 if oplniv=="VO-HAVO"
replace track=5 if oplniv=="VO-VWO"
```

bys brin track: gen ssize=_N

bys brin track: gen stag=_n==1

keep if oplniv_num==3|oplniv_num==1

dropmiss, force

rename latall latned5

rename seall sened5

rename latned13 latned16

rename latned12 latned14

rename latned11 latned12

rename latned23 latned26

rename latned22 latned24

rename latned21 latned22

rename latned33 latned36

rename latned32 latned34

rename latned31 latned32

rename latned43 latned46

rename latned42 latned44

rename latned41 latned42

rename sened13 sened16

rename sened12 sened14

rename sened11 sened12

rename sened23 sened26

rename sened22 sened24

```
rename sened21 sened22
```

```
rename sened33 sened36
```

```
rename sened32 sened34
```

```
rename sened31 sened32
```

```
rename sened43 sened46
```

```
rename sened42 sened44
```

```
rename sened41 sened42
```

```
local X ned10 ned20 ned30 ned40 ned12 ned14 ned16 ned22 ned24 ned26 ned32 ned34 ned36  
ned42 ned44 ned46 ned5
```

```
foreach var of local X {  
egen std_lat`var`=std(lat`var')  
drop lat`var`  
rename std_lat`var` lat`var`  
}
```

```
egen latned40_RP=rmean(latned42 latned44 latned46)
```

```
egen latned30_RP=rmean(latned32 latned34 latned36)
```

```
egen latned20_RP=rmean(latned22 latned24 latned26)
```

```
egen latned10_RP=rmean(latned12 latned14 latned16)
```

```
egen latned5_RP=rmean(latned40_RP latned30_RP latned20_RP latned10_RP)
```

```
drop latned5 latned10 latned20 latned30 latned40
```

```
rename latned40_RP latned40
```

```
rename latned30_RP latned30
```

```
rename latned20_RP latned20
```

```
rename latned10_RP latned10
```



```
rename latned5_RP latned5
```

```
////////////////////////////////////
```

```
drop n11score2119001- n43score2228001
```

```
//drop n11score1015001- n43score2236010
```

```
local X ned10 ned20 ned30 ned40 ned12 ned14 ned16 ned22 ned24 ned26 ned32 ned34 ned36  
ned42 ned44 ned46 ned5
```

```
foreach var of local X {
```

```
  bys brin track: egen Smean_`var'=mean(lat`var')
```

```
  bys track: egen mean_`var'T=mean(lat`var')
```

```
  gen rel_score`var'=Smean_`var'-mean_`var'T
```

```
  gen sesq_`var'=se`var'^2
```

```
  bys brin track: egen Ssesum`var'=sum(sesq_`var')
```

```
  gen schoolse_`var'=sqrt((Ssesum`var')/(ssize^2))
```

```
  xtile rel_scoreP`var'=rel_score`var', nq(100)
```

```
  gen lrel_lat`var'=lat`var'-mean_`var'T
```

```
  xtile lrel_latP`var'=lrel_lat`var', nq(100)
```

```
  bys track: egen mean_`var'_havo_=mean(lat`var') if track==4
```

```
  egen mean_`var'_havo=max(mean_`var'_havo_)
```

```
  drop mean_`var'_havo_
```

```
  bys track: egen mean_`var'_vwo_=mean(lat`var') if track==5
```

```
  egen mean_`var'_vwo=max(mean_`var'_vwo_)
```

```
  drop mean_`var'_vwo_
```

```
}
```

```

egen Platned40_RP=rmean(rel_scorePned42 rel_scorePned44 rel_scorePned46)
egen Platned30_RP=rmean(rel_scorePned32 rel_scorePned34 rel_scorePned36)
egen Platned20_RP=rmean(rel_scorePned22 rel_scorePned24 rel_scorePned26)
egen Platned10_RP=rmean(rel_scorePned12 rel_scorePned14 rel_scorePned16)
egen Platned5_RP=rmean(Platned10_RP Platned20_RP Platned30_RP Platned40_RP)

drop rel_scorePned40 rel_scorePned30 rel_scorePned20 rel_scorePned10 rel_scorePned5

rename Platned40_RP rel_scorePned40
rename Platned30_RP rel_scorePned30
rename Platned20_RP rel_scorePned20
rename Platned10_RP rel_scorePned10
rename Platned5_RP rel_scorePned5

local X ned10 ned20 ned30 ned40 ned12 ned14 ned16 ned22 ned24 ned26 ned32 ned34 ned36
ned42 ned44 ned46 ned5

foreach var of local X {
gen Srel_score`var'=Smean_`var'-Smean_ned5
gen relscore_ul_`var'=rel_score`var'+1.96*schoolse_`var'
gen relscore_ll_`var'=rel_score`var'-1.96*schoolse_`var'
gen Srelscore_ul_`var'=Srel_score`var'+1.96*schoolse_`var'
gen Srelscore_ll_`var'=Srel_score`var'-1.96*schoolse_`var'

gen Srel_scoreP`var'=rel_scoreP`var'-rel_scorePned5

gen Srel_scoreOT`var'=Smean_`var'-mean_`var'_havo if track==5
replace Srel_scoreOT`var'=Smean_`var'-mean_`var'_vwo if track==4
}

```

```
order *, alpha
```

```
order *, seq
```

```
order id id2 id_new afnamegroepid brin brinvn geslacht oplniv oplniv_num vakcode nrq sample
```

```
egen brin_id=group(brin brinvn track)
```

```
local X ned10 ned20 ned30 ned40 ned12 ned14 ned16 ned22 ned24 ned26 ned32 ned34 ned36  
ned42 ned44 ned46 ned5
```

```
foreach var of local X {
```

```
  bys brin_id: egen relscoreP25_`var'=pctile(Irel_lat`var'), p(24)
```

```
  bys brin_id: egen relscoreP50_`var'=pctile(Irel_lat`var'), p(49)
```

```
  bys brin_id: egen relscoreP75_`var'=pctile(Irel_lat`var'), p(76)
```

```
  bys track: egen relscoreTP25_`var'=pctile(Irel_lat`var'), p(24)
```

```
  bys track: egen relscoreTP50_`var'=pctile(Irel_lat`var'), p(49)
```

```
  bys track: egen relscoreTP75_`var'=pctile(Irel_lat`var'), p(76)
```

```
  bys brin_id: egen relscoreMean_`var'=mean(lat`var')
```

```
  bys brin_id: egen relscoreMeanP_`var'=mean(Irel_latP`var')
```

```
  gen PrelscoreP25_`var'_=Irel_latP`var' if relscoreP25_`var'==Irel_lat`var'
```

```
  bys brin_id: egen PrelscoreP25_`var'=mean(PrelscoreP25_`var'_)
```

```
  gen PrelscoreP50_`var'_=Irel_latP`var' if relscoreP50_`var'==Irel_lat`var'
```

```
  bys brin_id: egen PrelscoreP50_`var'=mean(PrelscoreP50_`var'_)
```

```
  gen PrelscoreP75_`var'_=Irel_latP`var' if relscoreP75_`var'==Irel_lat`var'
```

```
  bys brin_id: egen PrelscoreP75_`var'=mean(PrelscoreP75_`var'_)
```

```
}
```

```
egen TrelscoreMeanP_ned40=rmean(relscoreMeanP_ned42 relscoreMeanP_ned44  
relscoreMeanP_ned46)
```

```

egen TrelscoreMeanP_ned30=rmean(relscoreMeanP_ned32 relscoreMeanP_ned34
relscoreMeanP_ned36)

egen TrelscoreMeanP_ned20=rmean(relscoreMeanP_ned22 relscoreMeanP_ned24
relscoreMeanP_ned26)

egen TrelscoreMeanP_ned10=rmean(relscoreMeanP_ned12 relscoreMeanP_ned14
relscoreMeanP_ned16)

egen TrelscoreMeanP_ned5=rmean(TrelscoreMeanP_ned10 TrelscoreMeanP_ned20
TrelscoreMeanP_ned30 TrelscoreMeanP_ned40)

drop relscoreMeanP_ned40 relscoreMeanP_ned30 relscoreMeanP_ned20 relscoreMeanP_ned10
relscoreMeanP_ned5

rename TrelscoreMeanP_ned40 relscoreMeanP_ned40
rename TrelscoreMeanP_ned30 relscoreMeanP_ned30
rename TrelscoreMeanP_ned20 relscoreMeanP_ned20
rename TrelscoreMeanP_ned10 relscoreMeanP_ned10
rename TrelscoreMeanP_ned5 relscoreMeanP_ned5

local X ned10 ned20 ned30 ned40 ned12 ned14 ned16 ned22 ned24 ned26 ned32 ned34 ned36
ned42 ned44 ned46 ned5

foreach var of local X {
gen Prelscore_ul_`var'=relscoreMeanP_`var'+1.96*Pschoolse_`var'
gen Prelscore_ll_`var'=relscoreMeanP_`var'-1.96*Pschoolse_`var'

}

local X ned10 ned20 ned30 ned40 ned12 ned14 ned16 ned22 ned24 ned26 ned32 ned34 ned36
ned42 ned44 ned46 ned5

foreach var of local X {
gen meanH_`var'_=relscoreMeanP_`var' if id==99
egen meanH_`var'=max(meanH_`var'_)
gen relscoreH_`var'=relscoreMeanP_`var'-meanH_`var'

```

```
drop meanH_`var'_  
}
```

```
gen test1=.
```

```
gen test2=.
```

```
gen test3=.
```

```
gen test4=.
```

```
drop stag
```

```
bys brin_id: gen stag=_n==1
```

```
keep if stag==1
```

```
reshape long rel_scorened schoolse_ned sened sesq_ned Srel_scorened relscore_ll_ned  
relscore_ul_ned Srelscore_ll_ned Srelscore_ul_ned rel_scorePned Srel_scorePned Smean_ned  
relscoreP25_ned relscoreP50_ned relscoreP75_ned relscoreTP25_ned relscoreTP50_ned  
relscoreTP75_ned PrelscoreP25_ned PrelscoreP50_ned PrelscoreP75_ned relscoreMean_ned  
relscoreMeanP_ned Prelscore_ul_ned Prelscore_ll_ned, i (brin_id) j(D)
```

```
drop mean*
```

```
label define domains 5 "TOTAAL" 10 "AFSTEMMEN TEKST" 20 "STRUCTUUR" 30  
"ZINNEN/WOORDEN" 40 "SPELLING" 12 "Inschatting" 14 "Toonzetting" 16 "Schrijfdoel" 22  
"Tekstelementen" 24 "Indeling" 26 "Samenhang" 32 "Zinsbouw" 34 "Schrijfstijl" 36 "Vocabulaire" 42  
"Spelling ww" 44 "Spelling niet-ww" 46 "Interpunctie"
```

```
label values D domains
```

```
//FIGUUR 15
```

```
sum relscoreMeanP_ned if brin=="30GC" & D==5
```

```
local mean=r(mean)
```

```
replace test1=50
```

```
replace test2=r(mean)
```

```
twoway (scatter D relscoreMeanP_ned, xaxis(1)) (rcap Prelscore_ul_ned Prelscore_ll_ned D,  
horizontal xaxis(2) color(navy))(line D test1, color(black))(line D test2, color(red)) if brin=="30GC", ///
```

```
xla(25(25)75, axis(1)) xla(25 " " 50 "L" `mean' "S" 75 " ", axis(2) tlength(zero) labsize(vsmall))  
xtitle("Score", axis(1)) xtitle("", axis(2)) xline(50, lcolor(black)) xline(`mean', lcolor(red)) ///
```

```
ylabel(5 10 12 14 16 20 22 24 26 30 32 34 36 40 42 44 46, valuelabel angle(0) labsize(vsmall))
ytile("") xtitle("Score") graphregion(color(white)) legend(off) title("Anonieme school: Nederlands
(met betrouwbaarheid)", size(small) color(black))
```

```
//FIGUUR 16
```

```
sum relscoreMeanP_ned if brin=="30GC" & D==5 & track==5
```

```
local mean=r(mean)
```

```
replace test1=50
```

```
replace test2=r(mean)
```

```
replace test3=25
```

```
replace test4=75
```

```
twoway (scatter D relscoreMeanP_ned, xaxis(1)) (rcap PrelscoreP75_ned PrelscoreP25_ned D,
horizontal xaxis(2) color(forest_green))(line D test1, color(black))(line D test2, color(red))(line D test3,
color(black))(line D test4, color(black)) if brin=="30GC" & track==5, ///
```

```
xla(0(25)100, axis(1)) xla(0 " " 25 " " 50 "L" `mean' "S" 75 " " 100 " ", axis(2) tlength(zero)
labsize(vsmall)) xtitle("Score", axis(1)) xtitle("", axis(2)) xline(50, lcolor(black)) xline(`mean',
lcolor(red)) xline(25, lcolor(black)) xline(75, lcolor(black)) ///
```

```
ylabel(5 10 12 14 16 20 22 24 26 30 32 34 36 40 42 44 46, valuelabel angle(0) labsize(vsmall))
ytile("") graphregion(color(white)) legend(off) title("Anonieme school: Nederlands (met spreiding)",
size(small) color(black))
```

```
//FIGUUR 17
```

```
sum relscoreMeanP_ned if brin=="30GC" & D==5
```

```
replace test2=r(mean)
```

```
gen SSrel_scorePned= relscoreMeanP_ned-test2
```

```
sum SSrel_scorePned if brin=="30GC" & D==5
```

```
local mean=r(mean)
```

```
replace test1=0
```

```
replace test2=r(mean)
```

```
replace test3=25
```

```
replace test4=75
```

```

twoway (scatter D SSrel_scorePned, xaxis(1)) (rcap SSrel_scorePned SSrel_scorePned D, horizontal
axis(2) color(navy))(line D test1, color(red)) if brin=="30GC", ///

xla(-25(25)25, axis(1)) xla(-25 " " 0 "S" 25 " ", axis(2) tlength(zero) labsize(vsmall)) xttitle("Score",
axis(1)) xttitle("", axis(2)) xline(0, lcolor(red)) ///

ylabel(5 10 12 14 16 20 22 24 26 30 32 34 36 40 42 44 46, valuelabel angle(0) labsize(vsmall))
yttitle("") graphregion(color(white)) legend(off) title("Anonieme school: Nederlands (relatief aan
totaal)", size(small) color(black))

```

//FIGUUR 18

```

sum relscoreH_ned if brin=="16RB" & D==5

replace test1=0

replace test2=r(mean)

sum relscoreH_ned if D==5

replace test3=r(mean)

twoway (scatter D relscoreH_ned, xaxis(1)) (rcap relscoreH_ned relscoreH_ned D, horizontal xaxis(2)
color(navy))(line D test1, color(black))(line D test2, color(red))(line D test3, color(black)) if
brin=="16RB", ///

xla(-15(15)30, axis(1)) xla(-15 " " 0 "HAVO" 14.72 "S" 20.701 "VWO" 15 " " 30 " ", axis(2) tlength(zero)
labsize(vsmall)) xttitle("Score", axis(1)) xttitle("", axis(2)) xline(0, lcolor(black))xline(14.72347,
lcolor(red))xline(20.70158, lcolor(black)) ///

ylabel(5 10 12 14 16 20 22 24 26 30 32 34 36 40 42 44 46, valuelabel angle(0) labsize(vsmall))
yttitle("") graphregion(color(white)) legend(off) title("", size(small) color(black))

```

//FIGUUR 19: SIMULATIE

```

clear

set seed 1111

set obs 1856

drawnorm d f g h

gen theta=d

gen id_new=_n

```

```

forvalues i=1/40 {
gen uniform`i'=uniform()
gen A`i`_ =0.196+((1.88-0.196)*uniform`i') if id_new==`i'
egen A`i`=max(A`i`_)
drop A`i`_
gen B`i'=-1.5
replace B`i'=1.5 if `i'>20
gen p`i'=1/(1+2.71828^(-A`i'*(theta-B`i')))
}

forvalues i=1/40 {
gen q`i'=p`i'>uniform`i'
}

sort id_new

outfile id_new q1 q2 q3 q4 q5 q6 q7 q8 q9 q10 q11 q12 q13 q14 q15 q16 q17 q18 q19 q20 q21 q22
q23 q24 q25 q26 q27 q28 q29 q30 q31 q32 q33 q34 q35 q36 q37 q38 q39 q40 using "irt_simulD.txt",
replace nolabel

// RUN MPLUS

clear

infile q1 q2 q3 q4 q5 q6 q7 q8 q9 q10 q11 q12 q13 q14 q15 q16 q17 q18 q19 q20 q21 q22 q23 q24
q25 q26 q27 q28 q29 q30 q31 q32 q33 q34 q35 q36 q37 q38 q39 q40 id_new latDsim seDsim using
"irt_simulD_uit.txt"

keep id_new latDsim seDsim

gen latDsim_UB=latDsim+1.96*seDsim
gen latDsim_LB=latDsim-1.96*seDsim

drawnorm x

//replace x=x*2 if a>-2 & a<2

gen y=x

```



```
bys x: egen meany=mean(y)

sort x

twoway (line x meany, lcolor(black) lpattern(dash))(scatter latDsim_UB latDsim_LB latDsim),
legend(col(3))
```

```
////////////////////////////////////
```

```
//FIGUUR 20: SIMULATIE
```

```
clear
```

```
set seed 1111
```

```
set obs 1856
```

```
drawnorm d f g h
```

```
gen theta=d
```

```
gen id_new=_n
```

```
forvalues i=1/14 {
```

```
gen uniform`i'=uniform()
```

```
gen A`i`_`=0.103+((4.96-0.103)*uniform`i') if id_new==`i'
```

```
egen A`i`_`=max(A`i`_`_)
```

```
drop A`i`_`_
```

```
gen B`i`_`=-1.5
```

```
replace B`i`_`=1.5 if `i`_`>7
```

```
gen p`i`_`=1/(1+2.71828^(-A`i`_`*(theta-B`i`_`)))
```

```
}
```

```
forvalues i=1/14 {
```

```
gen q`i`_`=p`i`_`>uniform`i`_`
```

```
}
```

```
sort id_new
```

```
outfile id_new q1 q2 q3 q4 q5 q6 q7 q8 q9 q10 q11 q12 q13 q14 using "irt_simulD3.txt", replace  
nolabel
```

```
// RUN MPLUS
```

```
clear
```

```
infile q1 q2 q3 q4 q5 q6 q7 q8 q9 q10 q11 q12 q13 q14 id_new latD2sim seD2sim using  
"irt_simulD3_uit.txt"
```

```
keep id_new latD2sim seD2sim
```

```
gen latD2sim_UB=latD2sim+1.96*seD2sim
```

```
gen latD2sim_LB=latD2sim-1.96*seD2sim
```

```
drawnorm x
```

```
gen y=x
```

```
bys x: egen meany=mean(y)
```

```
sort x
```

```
twoway (line x meany, lcolor(black) lpattern(dash))(scatter latD2sim_UB latD2sim_LB latD2sim),  
legend(col(3)) graphregion(color(white)) xtitle("D2") ytitle("D2")
```

```
//FIGUUR 23
```

```
gen lowsessshare_=0.65+(uniform()/2)-(relscoreMeanP_ned/100) if D==5
```

```
bys brin_id: egen lowsessshare=max(lowsessshare_)
```

```
gen uniD_=uniform() if brin=="30GC"
```

```
bys D: egen uniD=max(uniD_)
```

```
gen relscoreMeanP_ned_corr=relscoreMeanP_ned+lowsessshare*uniD*10
```

```
sum relscoreMeanP_ned if brin=="16RB" & D==5
```

```
local mean=r(mean)
```

```
replace test1=50
```

```
replace test2=r(mean)
```

```
sum relscoreMeanP_ned_corr if brin=="16RB" & D==5
```

```

replace test3=r(mean)

twoway (scatter D relscoreMeanP_ned, xaxis(1) color(navy)) (scatter D relscoreMeanP_ned_corr,
xaxis(2) color(maroon) msymbol(diamond))(line D test1, color(black))(line D test2, color(red))(line D
test3, color(red)) if brin=="16RB", ///

xla(25(25)75, axis(1)) xla(25 " " 50 "L" `mean' "S1" 46.85079 "S2" 75 " ", axis(2) tlength(zero)
labsize(vsmall)) xtitle("Score", axis(1)) xtitle("", axis(2)) xline(50, lcolor(black)) xline(42.38, lcolor(red))
xline(46.85079, lcolor(red)) ///

ylabel(5 10 12 14 16 20 22 24 26 30 32 34 36 40 42 44 46, valuelabel angle(0) labsize(vsmall))
yttitle("") legend (order(1 "Score" 5 "Gecorrigeerde score")) graphregion(color(white)) title("Correctie
voor SES (VWO nederlands)", size(small) color(black))

```

```
//VOOR DE FIGUREN 24 EN 25 IS DE PRE-TEST 2015 GEBRUIKT
```

```
//WE SPECIFICEREN ALLEREERST DE CODE OM HET BESTAND OM TE ZETTEN
```

```
bys id: egen nrq = max(item_chron)
```

```
bys id item_short: gen countX=_n
```

```
drop if countX==2
```

```
gen item_level = substr(item,-10,2)
```

```
replace item_level=substr(item,-12,2) if item_level=="-0"
```

```
replace item_short = item_short+10000 if item_level=="vm"
```

```
replace item_short = item_short+20000 if item_level=="hv"
```

```
gen item_domain=""
```

```
gen sample=1
```

```
drop item datestamp attempts
```

```
sort item_short, stable
```

```
egen item_ordered = group(item_short)
```

```
drop item_ordered
```

```
drop score
```

```
drop brinvn exmkd_id gebdat
```

```
egen scoreB = rsum(scoresub1 scoresub2 scoresub3 scoresub4 scoresub5 scoresub6 scoresub7  
scoresub8 scoresub9 scoresub10 scoresub11 scoresub12 scoresub13)
```

```
drop scoresub*
```

```
drop oplniv_num_ leerweg_num_ subitem*
```

```
gen duration_std=.
```

```
drop if duration>900
```

```
bys id: gen countq=_N
```

```
drop if countq<=2
```

```
joinby id using "dttned_reshaped_withirtF_.dta", unmatched(none)
```

```
local X 10020 10030 10050 10060 10070 10080 10110 10120 10140 10150 10180 10190 10200  
10220 10240 10360 10380 10450 10480 10610 10620 10630 10650 10670 10680 10690 10810 10820  
10850 10960 10970 11000 11070 11090 11100 11110 11150 11160 11190 11330 11340 11350 11360  
11390 11400 11410 11610 11690 11700 11710 11720 11730 11740 11750 11770 11780 11790 11810  
11820 11830 11840 11850 11860 11870 11880 11890 11900 11910 11920 11930 11940 11950 11970  
11980 11990 12000 12010 12020 12030 12040 12050 12060 12070 12080 12090 12100 12110 12120  
12130 12140 12150 12160 12170 12190 12210 12220 20060 20120 20140 20150 20160 20180 20190  
20200 20210 20220 20230 20240 20250 20260 20270 20280 20290 20300 20310 20320 20330 20340  
20350 20360 20380 20401 20402 20403 20410 20431 20432 20433 20434 20440 20450 20470 20480  
20490 20530 20550 20560 20570 20580 20590 20610 20630 20640 20650 20660 20670 20680 20690  
20710 20720 20730 20780 20790 20800 20810 20822 20823 20870 21050 21071 21072 21080 21090  
21131 21132 21141 21142 21160 21190 21200 21210 21251 21252 21300 21320 21410 21441 21442  
21480 21560 21570 21670 21730 21790 21830 21840 21960 22031
```

```
foreach x of local X {
```

```
egen meandurationitemid`x`=mean(duration) if item_short==`x'
```

```
replace duration_std = duration-meandurationitemid`x' if item_short==`x'
```

```
drop meandurationitemid`x`  
}
```

```
bys item_short: egen maxscore=max(scoreB)  
gen vraaggoed=0  
replace vraaggoed=1 if scoreB==maxscore
```

```
gen deelsgoed=0  
replace deelsgoed=1 if scoreB>0
```

```
bys item_chron: egen meanduration_std = mean(duration_std)  
//FIGUUR 24  
scatter meanduration_std item_chron
```

```
bys item_chron: egen meanduration = mean(duration)  
bys item_chron vraaggoed: egen meanduration_stdC = mean(duration_std)  
gen track=.  
replace track=1 if oplniv_num==2 & leerweg_num==1  
replace track=2 if oplniv_num==2 & leerweg_num==2  
replace track=3 if oplniv_num==2 & leerweg_num==3  
replace track=4 if oplniv_num==1  
replace track=5 if oplniv_num==3
```

```
bys item_chron track: egen meandurationT = mean(duration)  
bys item_chron track: egen meanduration_stdT = mean(duration_std)
```

```
bys item_chron geslacht: egen meanduration_stdG = mean(duration_std)  
bys item_chron geslacht track: egen meanduration_stdTG = mean(duration_std)
```

```
gen lat_all=latned_vmbo  
replace lat_all=latned_hv if lat_all==.
```

```
egen meanlatall=mean(lat_all)
bys track: egen meanlatallT=mean(lat_all)
gen tophalf=lat_all>meanlatall
gen tophalfT=lat_all>meanlatallT
bys item_chron tophalf: egen meanduration_std50 = mean(duration_std)
bys item_chron tophalfT track: egen meanduration_std50T = mean(duration_std)
```

```
//FIGUUR 25
```

```
twoway (scatter meanduration_std50 item_chron if tophalf==0)(scatter meanduration_std50
item_chron if tophalf==1), graphregion(color(white)) xtitle("Item (chron)") ytitle("Duur per vraag")
legend(label(1 "Onderste helft") label (2 "Bovenste helft"))
```