

Einföldun

Sumarverkefni

Dæmi um skýrslu

1 Inngangur

Aðeins til að byrja, þetta er tilraun hjá mér í að ná að orða hluti aðeins betur og koma gögnunum fram á skiljanlegan hátt. Það vantar líklega enn aðeins upp á það. En það er markmið skýrslunar hér.

Nú, rannsóknarspurninginn sem verið er að vinna með hér er:

Er hægt að mæla framfarir en ekki bara utanaðbókarlærdóm? Bera saman fyrstu svarmöguleika og þá sem hafa komið áður.

2 Gögninn

Til að byrja með, er gott að skoða gögninn. Ég hef fjarlægt sumar breytur, þar sem þær voru ruglingslegar og óþarfi.

```
hashAnswer <- read.csv('Data/hashAnswer4.csv') %>% subset(select = -c(timeStart, fsvfat, fsvfatu, X))
hashAnswer$hsta <- hashAnswer$hsta %>% as.character()
hashAnswer$lectureId <- hashAnswer$lectureId %>% as.factor()
hashAnswer$studentId <- hashAnswer$studentId %>% as.factor()
hashAnswer$nicc <- hashAnswer$nicc %>% as.factor()
hashAnswer %>% glimpse()
```

```
## Rows: 115,197
## Columns: 10
## $ lectureId <fct> 3082, 3082, 3082, 3082, 3082, 3082, 3082, 3082, 3082, 30...
## $ studentId <fct> 1, 1, 1, 1, 1, 18788, 18788, 18788, 18788, 18788, 18788,...
## $ questionId <int> 45591, 44899, 45647, 45647, 45647, 44920, 45021, 45480, ...
## $ correct <int> 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1,...
## $ hash <chr> "d3a978206c877eee8627a29c7a94238d8339d66fc4be460d9e6b6cb..."
## $ fsfat <int> 0, 1, 2, 2, 2, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 9, 9, 10, 1...
## $ hsta <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "...
## $ timeDif <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 254, NA, NA,...
## $ nicc <fct> 3, 3, 3, 3, 3, 3, 7, 3, 3, 4, 5, 6, 6, 5, 3, 3, 3, 3, 7,...
## $ gpow <dbl> 8.2506619, 8.2506619, 8.2506619, 8.2506619, 8.2506619, 0...
```

Gagnasettið, eins og það er núna hefur 10 breytur. Breyturnar eru:

- lectureId: Þetta er Id breyta fyrir hvaða fyrirlestur við erum í.
- studentId: Þetta er Id breyta fyrir hvaða nemenda við erum með.
- questionId: Þetta er Id breyta fyrir hvaða spurningu við erum með.
- correct: Þetta er binary breyta sem segir til hvort það var svarað rétt eða rangt.
- hash: Þetta er hash breyta sem segir til hvaða rétta svar við erum með hérna
- fsfat: “fjöldi spurninga fram að þessu” þetta segir til um hve margar spurningar nemandinn hefur svarað hingað til, innan við þennan fyrirlestur. Talning byrjar á 0.
- hsta: “hef séð þetta áður” er breyta sem segir til um hvort að nemandinn hefur séð rétta svarið áður eða ekki
- timeDif: Þetta er breyta sem segir okkur hve langt það var séðan nemandinn sá þetta svar áður.

- `nicc`: þetta er “number of incorrect choices”, líke þekkt sem `NumQ`. Segir til hve margir rangir svarmöguleikar það eru.
- `gpow`: `gpow` er breyta sem er sett á nemenda þegar hann byrjar á fyrirlestrinum sem segir til hve hratt hann fær erfiðari spurningar.

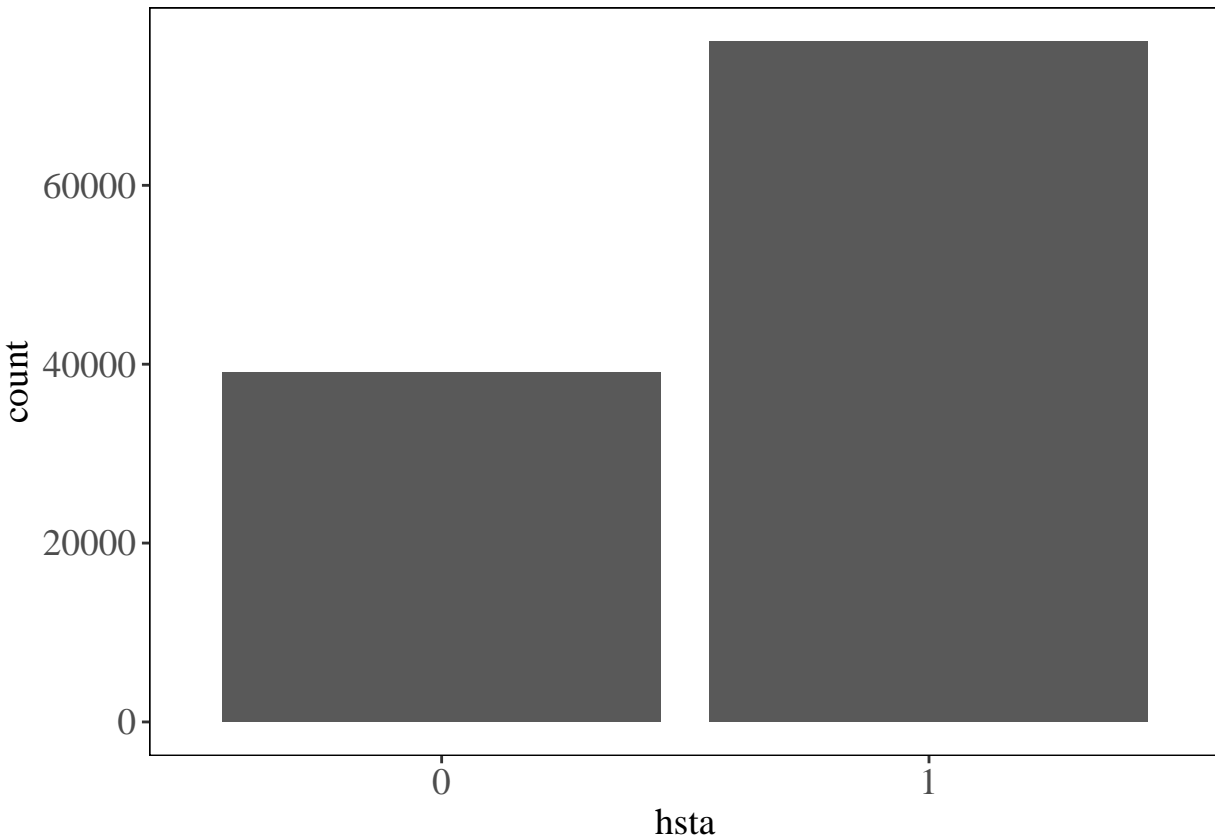
3 Teikningar og töflur

Fyrsta sem er gott að gera, er að skoða aðeins gögninn sjálf.

3.1 skoðað fyrir öll gögninn

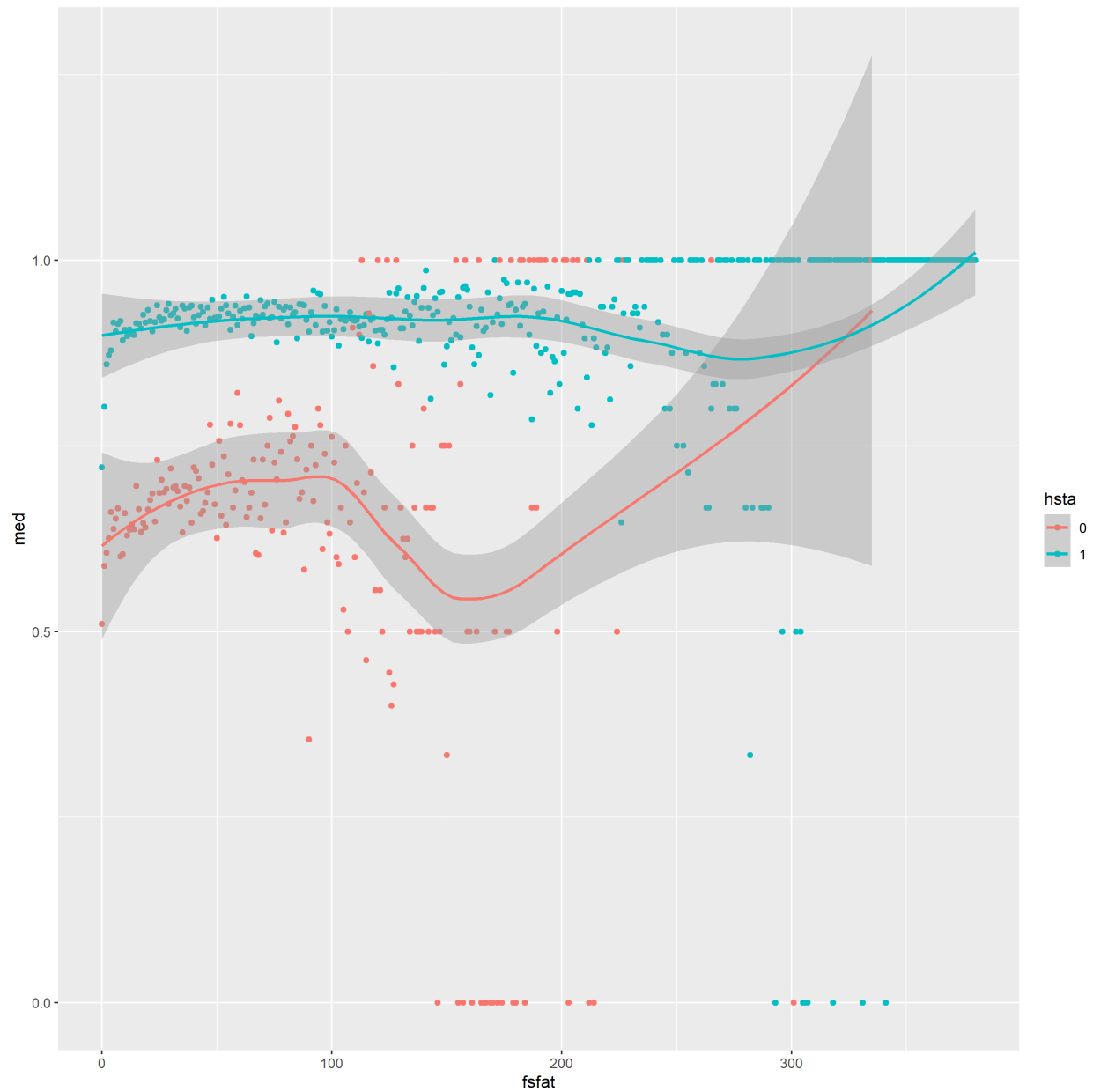
Byrjum aðeins á því að skoða hve mikið af gögnunum eru nemendur að sjá nýtt svar á móti gömlu svari.

```
ggplot(hashAnswer, aes(x = hsta)) +  
  geom_bar()
```



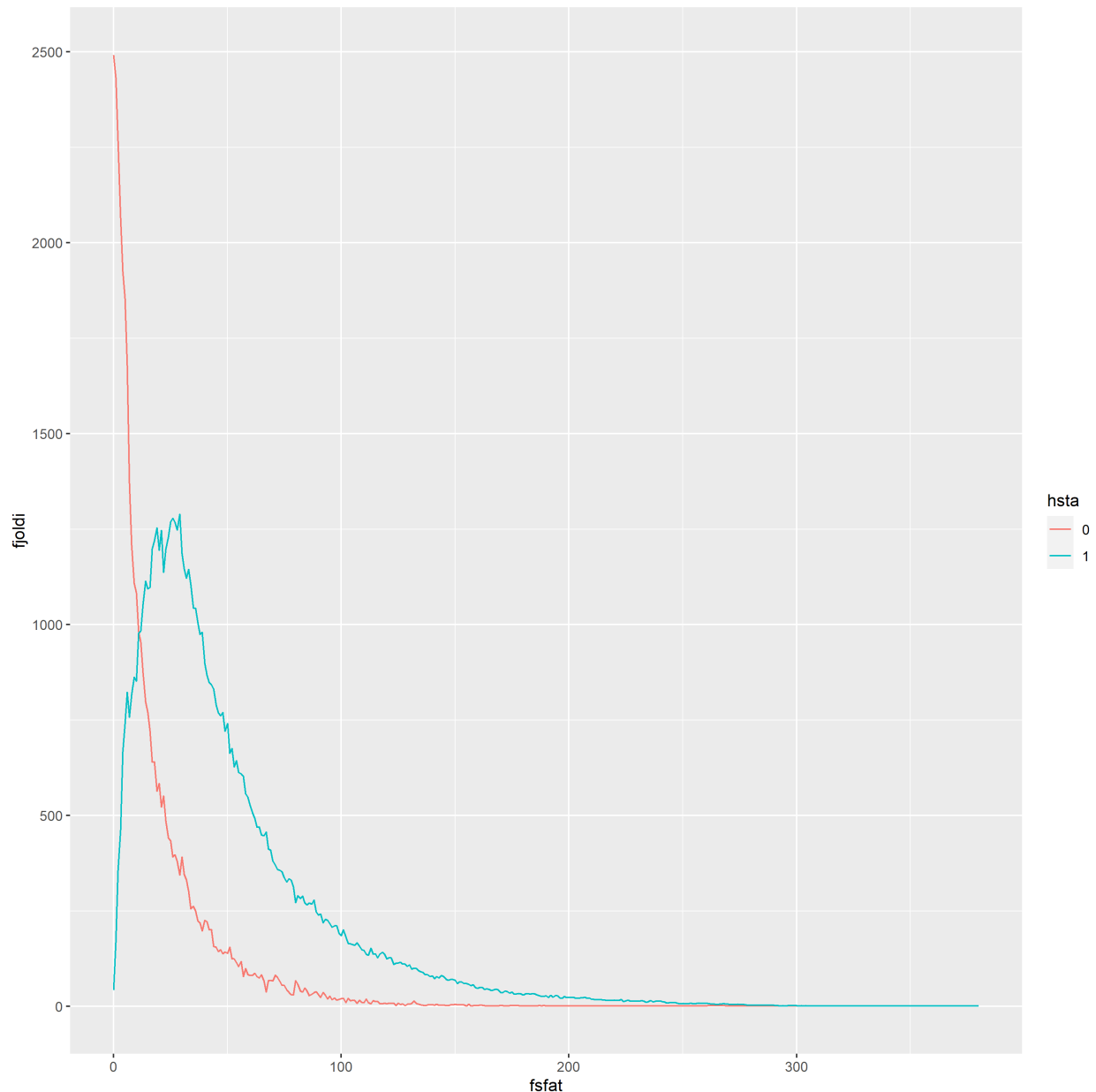
Það lítur út fyrir að vera mikið meira af gögnunum sem eru svör sem hafa sést áður á móti nýjum svörum. Þar sem u.þ.b. $2/3$ af gögnunum eru spurningar með rétt svör sem hafa sést áður á meðan $1/3$ af gögnunum eru spurningar með rétt svör sem eru að sjást í fyrsta sinn.

Nú markmiðið er að geta skoðað framfarir nemenda, ekki bara utanaðbókarlærdóm, svo það væri gott að setja upp graf sem inniheldur meðaltal réttra svara fyrir hvert fsfa og skipt eftir hsta. Samsagt meðaltal réttra svara hjá þeim sem hafa séð x margar spurningar hingað til, bæði fyrir þann fjölda sem hefur séð rétta svarið áður og þau sem hafa það ekki.



Mynd 1: plot by mean

Það er nú smá erfitt að lesa þessa mynd, það lítur út fyrir að vera smá vaxi hjá þeim sem eru að sjá þetta í fyrsta skiptið, svo fellur það niður og aftur upp. Svo er dreifnin að vaxa því lengra sem er farið. Skoðum aðeins stærð gagnanna fyrir hvern punkt til að sjá hvað er að gerast



Mynd 2: plot by sum

Hérna kemur skýringinn, það lítur út fyrir að vera stórt fall af gögnum sem eru að sjá svar í fyrsta sinn því lengra sem er farið, og líka hjá þeim sem hafa séð svarið áður. Þetta er líklega útaf því að hver nemandi fer mismunandi langt og sumir nemendur geta verið að fara lengur en hinir. Skoðum aðeins hlutföll gagnanna til að skoða þetta betur.

```
a <- hashAnswer %>% summarise("FY50" = sum(fsfat >= 50), "FY100" = sum(fsfat >= 100),  
                              "FY150" = sum(fsfat >= 150), "FY200" = sum(fsfat >= 200),  
                              "FY250" = sum(fsfat >= 250), "FY300" = sum(fsfat >= 300))
```

```

b <- hashAnswer %>% summarise("HY50" = mean(fsfat >= 50), "HY100" = mean(fsfat >= 100),
                              "HY150" = mean(fsfat >= 150), "HY200" = mean(fsfat >= 200),
                              "HY250" = mean(fsfat >= 250), "HY300" = mean(fsfat >= 300))
ab <- cbind(a, b)
FHbylim <- ab %>% pivot_longer(c('FY50', 'HY50', 'FY100', 'HY100', 'FY150', 'HY150', 'FY200', 'HY200',
                              names_to = "typewLim", values_to = "values") %>%
  separate(typewLim, into = c("type", "limit"), sep = 2) %>% pivot_wider(names_from = type, values_from = values)

FHbylim %>% kable(digits = 4)

```

limit	FY	HY
50	31964	0.2775
100	9553	0.0829
150	3204	0.0278
200	1149	0.0100
250	332	0.0029
300	84	0.0007

Lítur út fyrir að um 27.75% gagnanna eru af spurningum sem fara yfir 50 fsfat og bara u.þ.b. 8.29% gagnanna sem fara yfir 100. Hugmynd er þá að skera af gögninn miðað við 50 fsfat eða 100 fsfat. Skoðum þá aðeins hlutfall hsta og meðtalals grafið að ofan miðað við það.

3.2 Skoðað eftir limit

Fyrst er að setja upp gögninn. Að auki fjarlægji ég nemendur sem svöruðu ekki fleiri en 7 spurningum samtals

```

hashLim50 <- hashAnswer %>% group_by(studentId) %>% mutate("count" = n()) %>%
  filter(count > 7 & fsfat < 50)
hashLim100 <- hashAnswer %>% group_by(studentId) %>% mutate("count" = n()) %>%
  filter(count > 7 & fsfat < 100)

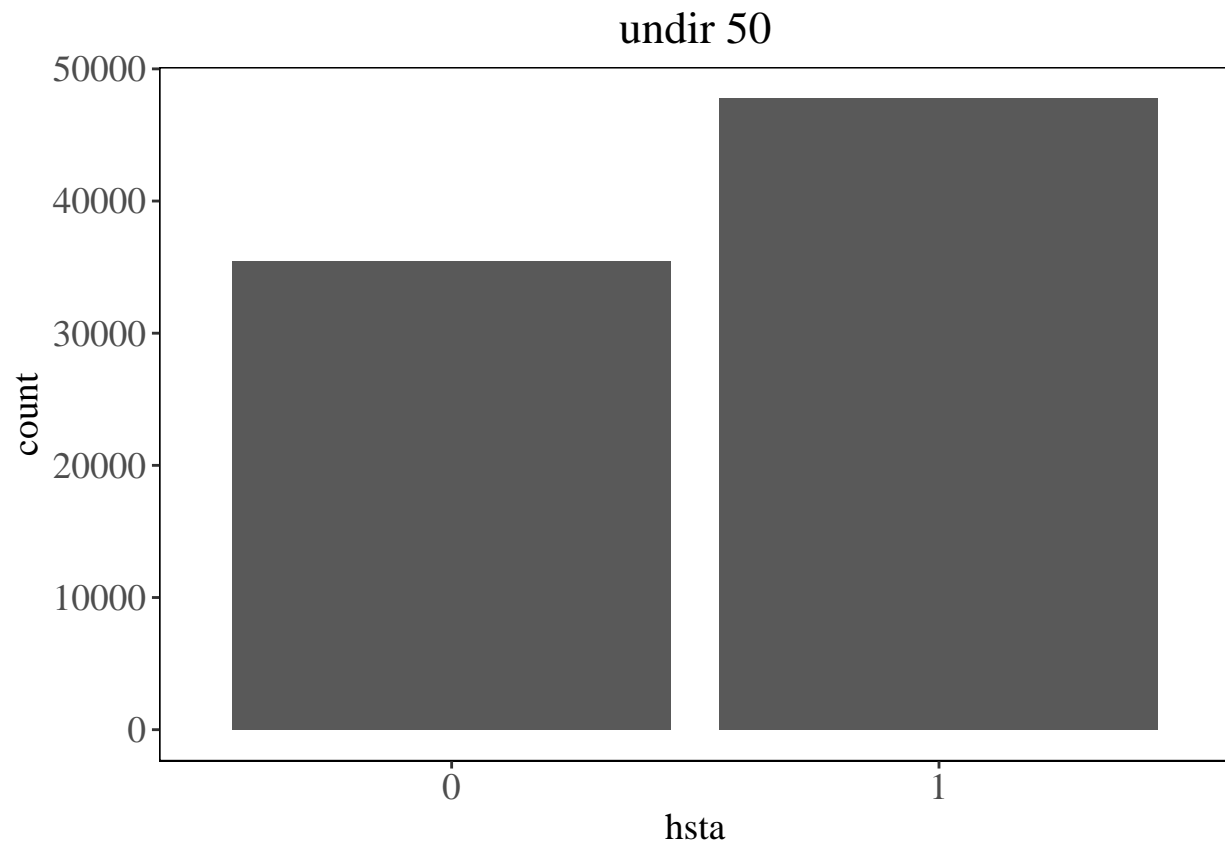
```

Skoðum hlutfallið aftur hjá hsta.

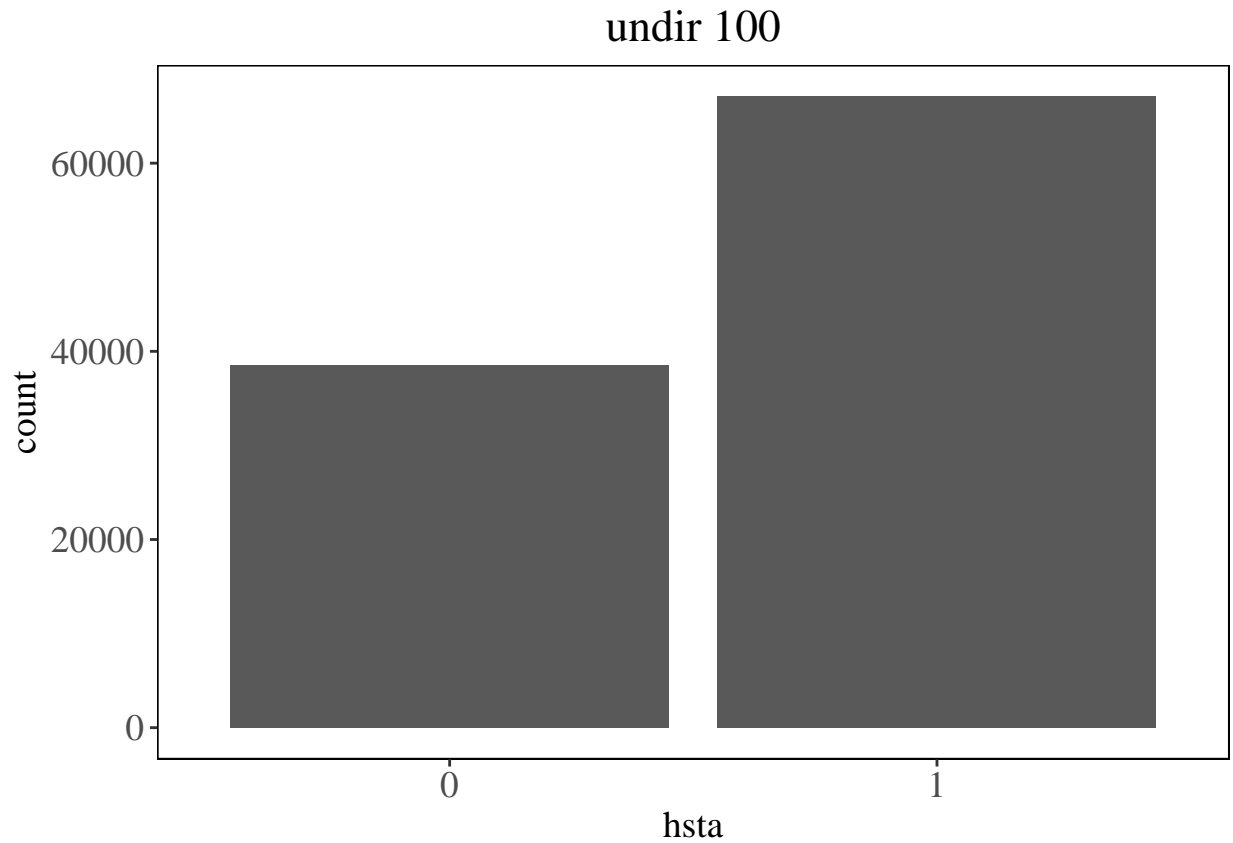
```

ggplot(hashLim50, aes(x = hsta)) +
  geom_bar() +
  ggtitle("undir 50")

```

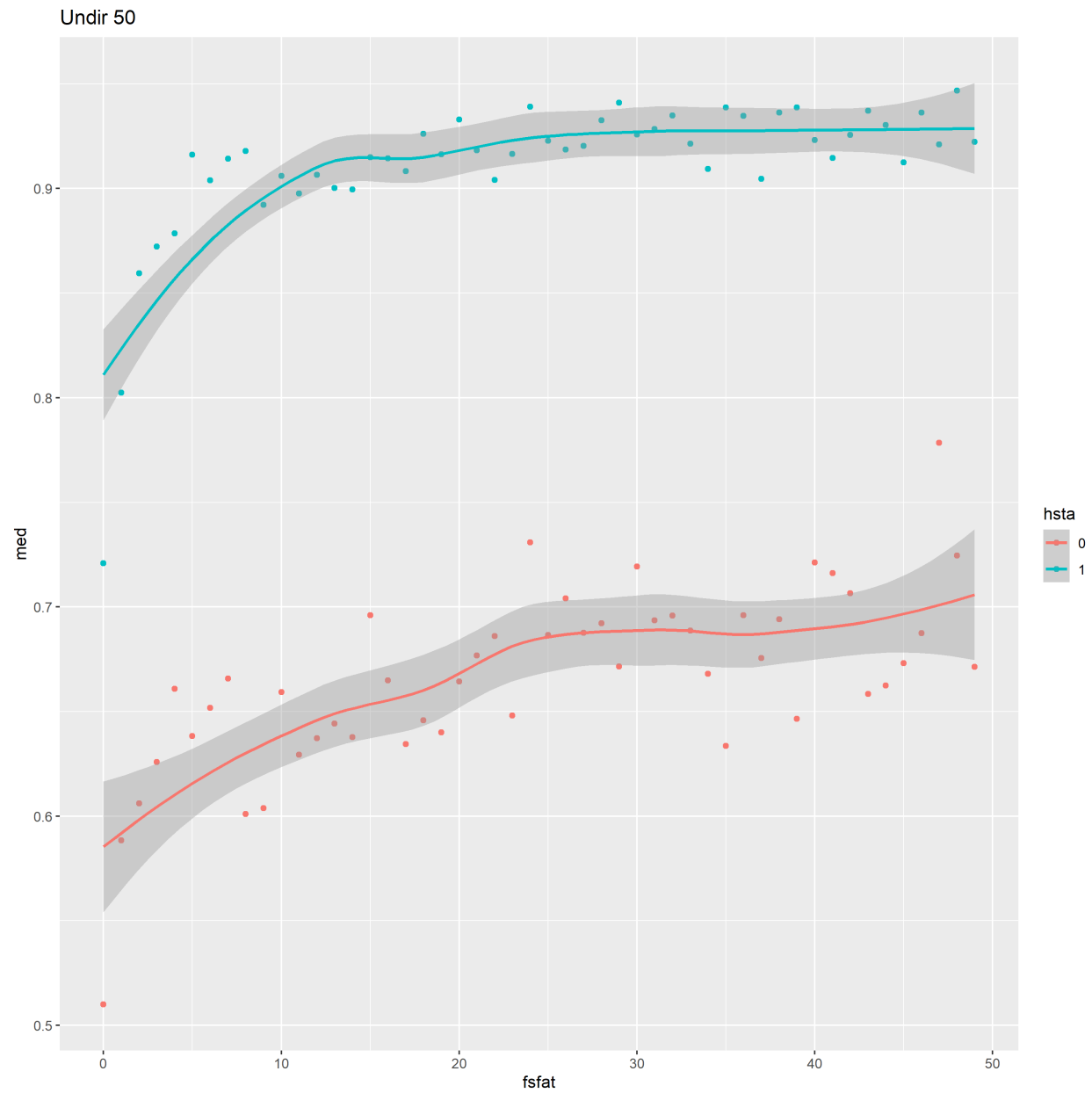


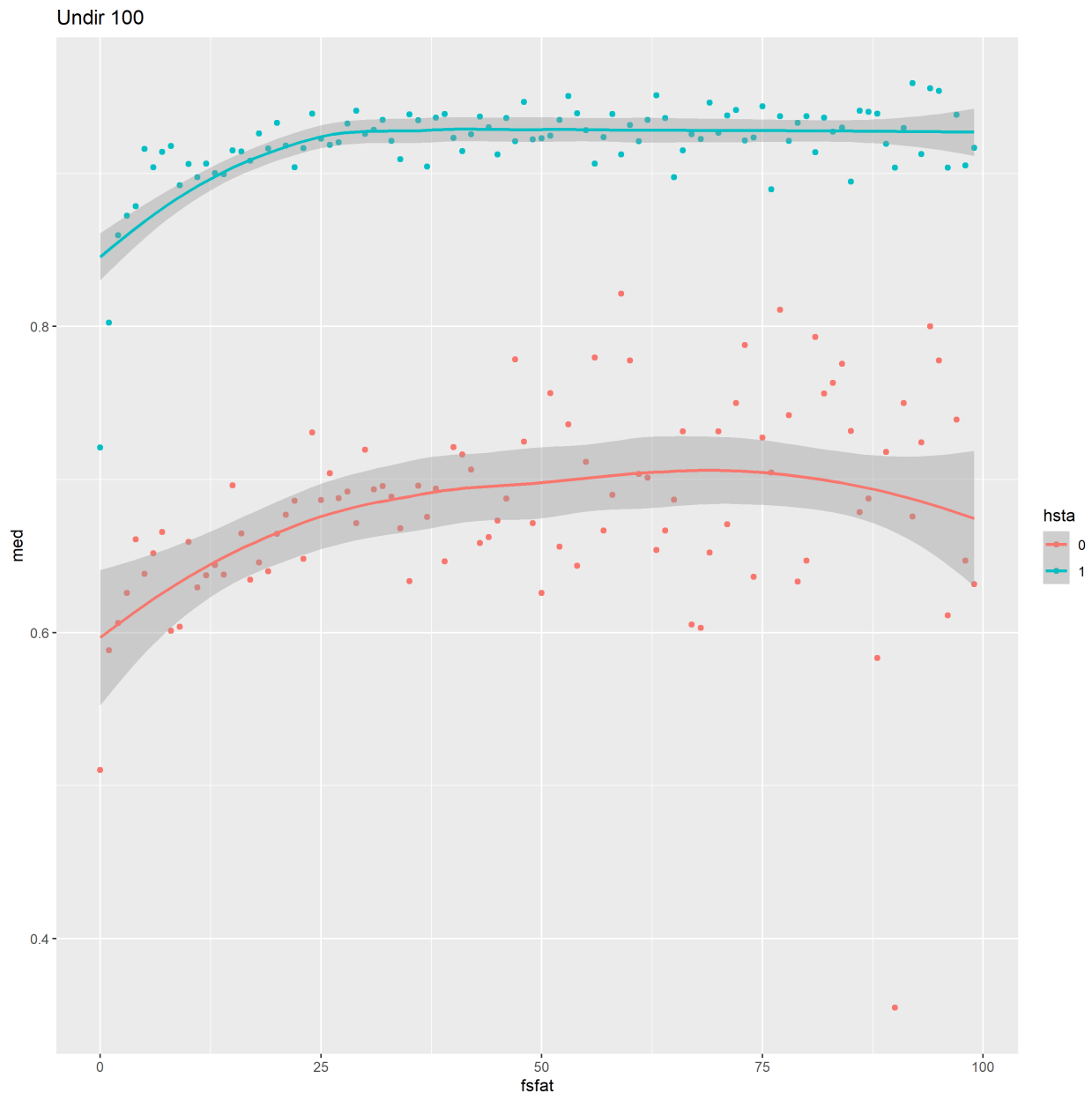
```
ggplot(hashLim100, aes(x = hsta)) +  
  geom_bar() +  
  ggtitle("undir 100")
```



Það sést að ef við tökum bara undir 50, þá er hsta orðið miklu nær og þegar við erum með undir 100, þá erum við kominn nálægt ójöfnunni eins og áðan.

Gott að skoða meðaltals myndina aftur, eftir að skera eftir limit.





Nú sést aðeins betur hvað er að gerast, í fyrri sjáum við að það lítur út fyrir að vera árangur og framfarir. Þar sem báðar línurnar eru að vaxa þegar fleiri spurninga hefur verið svarað. En það sést á seinni myndinni að það fer að jafna sig eftir það og framfarir hætta. Svo er það eitthvað smá fall hjá nýjum svörum.

3.3 umræður

Þá kemur spurningin hvort þetta er nokkuð framfarir eða utanaðbókarlærdómur, í fyrri þá lítur út fyrir að líkurnar vaxa. En það virðist svo jafna sig eftir það.

1. Ein hugmynd um hvað er að gerast þar er að þetta er ekki framfarir sem er að vaxa, heldur er það utanaðbókarlærdómur á röngu svarmöguleikunum, Það gæti verið aðal áhrifinn þar. Hugmynd til að skoða það betur væri kannski að skoða eins mynd, nema filterað bara fyrir spurningarnar sem hafa rétt svör sem eru að sjást í fyrsta skiptið og gert flokka fyrir hlutfall rangra svara sem hafa sést áður. Þá

mun koma ein lína fyrir hvert hlutfall, ef það er allt beinar láréttar línur, þá gæti það bent til að verið sé að ræða utanaðbókarlærdómur sem aðal áhrifin.

- Önnur hugmynd er að ef við gætum skipt nemendum upp í flokka, sem nemendur sem setja allt á minnið og nemendur sem eru að læra og fá framfarir. Svo er spurning hvort framfarirnar sjást betur þá.
- Loka hugmynd er að í staðinn fyrir að skoða fsfat eftir fyrirlestri, að skoða það kannski frekar eftir spurningahaus, þá gæti sést betur áhrif á sambærilegum spurningum eftir því hve margar tilraunir nemandinn hefur fengið á spurninguna sjálfa.

4 Modelgerð

Hér er smá grunnur af því sem ég er kominn með fyrir modelgerðina.

Ég byrjaði að skoða einföld glm, en þar sem það er verið að ræða um mixed effect model, þá hef ég fært yfir í glmer í staðinn. Ég ákvað að keyra allt miðað við 100 limitinn, ég hef ekki neina sterka ástæðu enn, þar sem ég veit ekki um leið til að sýna hvort einn gagnasett er betri en hitt, svo mér fannst 100 vera nægilega gott í bili.

4.1 gagnavinnsla

Fyrsta sem þarf að gera, er smá gagnavinnsla. Glmer tekur ekki vel við háu tölunum í fsfat, svo ég deili með 10 til að fá glmer til að keyra almennilega

```
hashLim100$fsfat <- hashLim100$fsfat/10
```

4.2 modelin sjálf

Modelinn sem ég er að vinna með á þessum tímamarki er

- `glmer(correct ~ fsfat*hsta + nicc + gpow + lectureId + (1 | studentId), family = binomial(link = "logit"), ...)`
- `glmer(correct ~ fsfat*hsta + nicc + gpow + lectureId + (1 + fsfat * hsta | studentId), family = binomial(link = "logit"), ...)`

Stóri munurinn milli þeirra er að ég hef meira í mixed effect modelinu seinna en í því fyrra

Ég keyri módelinn ekki hér, því þau eru of löng, en hef save-að þau til seinni notkunar, svo hægt sé að load-a þeim upp

```
#Þetta er fyrri modelið
load("Data/ans22")
```

```
## Registered S3 methods overwritten by 'lme4':
## method from
## cooks.distance.influence.merMod car
## influence.merMod car
## dfbeta.influence.merMod car
## dfbetas.influence.merMod car
```

```
#Þetta er seinna modelið
load("Data/ans42")
```

4.3 smá model Analysis

Summaríið þeirra er svo

```
summary(ans22)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula: correct ~ fsfat * hsta + nicc + gpow + lectureId + (1 | studentId)
## Data: hashTest2
## Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+05))
##
##      AIC      BIC    logLik deviance df.resid
## 78644.1 78873.8 -39298.1 78596.1 105586
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -11.2266  0.1580  0.2543  0.4171  3.7824
##
## Random effects:
## Groups Name Variance Std.Dev.
## studentId (Intercept) 0.7422 0.8615
## Number of obs: 105610, groups: studentId, 283
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.565915  0.060601 25.840 < 2e-16 ***
## fsfat        0.192314  0.005003 38.442 < 2e-16 ***
## hsta1       -0.952244  0.016145 -58.980 < 2e-16 ***
## nicc1        0.623305  0.135571  4.598 4.27e-06 ***
## nicc2        0.331299  0.054191  6.114 9.74e-10 ***
## nicc3       -0.247856  0.027513 -9.009 < 2e-16 ***
## nicc4       -0.014195  0.036763 -0.386 0.699402
## nicc5       -0.183080  0.035432 -5.167 2.38e-07 ***
## nicc6       -0.256486  0.035157 -7.295 2.98e-13 ***
## gpow         0.123848  0.033556  3.691 0.000224 ***
## lectureId1  -1.018070  0.024856 -40.959 < 2e-16 ***
## lectureId2   0.017850  0.033273  0.536 0.591628
## lectureId3   0.558622  0.044085 12.672 < 2e-16 ***
## lectureId4  -0.547061  0.026746 -20.454 < 2e-16 ***
## lectureId5  -0.170183  0.030146 -5.645 1.65e-08 ***
## lectureId6   0.138075  0.034613  3.989 6.63e-05 ***
## lectureId7   0.291268  0.037391  7.790 6.72e-15 ***
## lectureId8   0.447420  0.045693  9.792 < 2e-16 ***
## lectureId9   0.288598  0.044901  6.427 1.30e-10 ***
## lectureId10 -0.088576  0.036978 -2.395 0.016602 *
## lectureId11  0.062259  0.039052  1.594 0.110879
## lectureId12  0.145584  0.039316  3.703 0.000213 ***
```

```
## fsfat:hsta1 0.027901 0.004562 6.116 9.57e-10 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Correlation matrix not shown by default, as p = 23 > 12.
## Use print(x, correlation=TRUE) or
##      vcov(x)          if you need it
```

```
summary(ans42)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula: correct ~ fsfat * hsta + nicc + gpow + lectureId + (1 + fsfat *
##      hsta | studentId)
## Data: hashTest2
## Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+05))
##
##      AIC      BIC    logLik deviance df.resid
## 77962.3 78278.0 -38948.2 77896.3 105577
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -13.6590  0.1379  0.2467  0.4156  3.1668
##
## Random effects:
## Groups      Name                Variance Std.Dev. Corr
## studentId (Intercept) 0.675882 0.82212
##              fsfat      0.024972 0.15803  0.02
##              hsta1      0.072061 0.26844 -0.31 0.12
##              fsfat:hsta1 0.003271 0.05719  0.50 -0.01 -0.60
## Number of obs: 105610, groups: studentId, 283
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 1.477046 0.059899 24.659 < 2e-16 ***
## fsfat      0.259132 0.012380 20.931 < 2e-16 ***
## hsta1      -0.881182 0.025763 -34.203 < 2e-16 ***
## nicc1      0.626590 0.135552 4.623 3.79e-06 ***
## nicc2      0.340028 0.054450 6.245 4.24e-10 ***
## nicc3     -0.251857 0.027567 -9.136 < 2e-16 ***
## nicc4     -0.013454 0.036861 -0.365 0.71512
## nicc5     -0.185017 0.035551 -5.204 1.95e-07 ***
## nicc6     -0.257133 0.035274 -7.290 3.11e-13 ***
## gpow       0.118689 0.036084 3.289 0.00100 **
## lectureId1 -1.027480 0.025664 -40.036 < 2e-16 ***
## lectureId2 0.016464 0.033579 0.490 0.62392
## lectureId3 0.572888 0.044263 12.943 < 2e-16 ***
## lectureId4 -0.573858 0.027279 -21.036 < 2e-16 ***
## lectureId5 -0.166690 0.030503 -5.465 4.64e-08 ***
## lectureId6 0.140893 0.034812 4.047 5.18e-05 ***
## lectureId7 0.301169 0.037644 8.000 1.24e-15 ***
```

```
## lectureId8    0.452285    0.045846    9.865 < 2e-16 ***
## lectureId9    0.293143    0.045167    6.490 8.57e-11 ***
## lectureId10  -0.088364    0.037386   -2.364 0.01810 *
## lectureId11    0.062045    0.039242    1.581 0.11385
## lectureId12    0.148848    0.039586    3.760 0.00017 ***
## fsfat:hsta1    0.014892    0.007534    1.977 0.04808 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 23 > 12.
## Use print(x, correlation=TRUE) or
##      vcov(x)          if you need it
```

Sést að það er einhver munur hér á milli við styrkleika breytanna. Sem passar frá muninum þeirra.

Ef við skoðum aðeins Anova hjá þeim báðum

```
Anova(ans22, type = 3)
```

```
## Analysis of Deviance Table (Type III Wald chisquare tests)
##
## Response: correct
##              Chisq Df Pr(>Chisq)
## (Intercept)  667.695  1 < 2.2e-16 ***
## fsfat        1477.763  1 < 2.2e-16 ***
## hsta         3478.680  1 < 2.2e-16 ***
## nicc         172.253  6 < 2.2e-16 ***
## gpow         13.622  1 0.0002235 ***
## lectureId    2128.746 12 < 2.2e-16 ***
## fsfat:hsta    37.411  1  9.57e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Anova(ans42, type = 3)
```

```
## Analysis of Deviance Table (Type III Wald chisquare tests)
##
## Response: correct
##              Chisq Df Pr(>Chisq)
## (Intercept)  608.0674  1 < 2.2e-16 ***
## fsfat        438.1232  1 < 2.2e-16 ***
## hsta         1169.8764  1 < 2.2e-16 ***
## nicc         177.1123  6 < 2.2e-16 ***
## gpow         10.8193  1  0.001004 **
## lectureId    2102.7021 12 < 2.2e-16 ***
## fsfat:hsta    3.9071  1  0.048082 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Þá lítur út fyrir að allar breytur eru marktækar hjá þeim.

4.4 Model comparison

Hvor er svo sterkari? Skoðum aðeins anova milli þeirra

```
anova(ans22, ans42)
```

```
## Data: hashTest2
## Models:
## ans22: correct ~ fsfat * hsta + nicc + gpow + lectureId + (1 | studentId)
## ans42: correct ~ fsfat * hsta + nicc + gpow + lectureId + (1 + fsfat *
## ans42:      hsta | studentId)
##      npar   AIC   BIC logLik deviance  Chisq Df Pr(>Chisq)
## ans22   24 78644 78874 -39298    78596
## ans42   33 77962 78278 -38948    77896 699.83  9 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Lítur út fyrir að ans42 er betra, það getur passað þar sem það er meira leift. En vandinn er að seinni modelið tekur u.þ.b. tvöfalt lengri tíma að keyra í hvert sinn, svo þá kemur spurningin “hve mikið betri er seinni miðað við fyrri?”

Setjum upp fall fyrir BrierScore og reiknum svo það fyrir þau bæði

```
BrierScore <- function(modl, df ) {
  predicted <- predict(modl, type = "response")
  truth <- df$correct
  return(mean((predicted-truth)^2))
}

br22 <- BrierScore(ans22, hashLim100)
br42 <- BrierScore(ans42, hashLim100)

br22
```

```
## [1] 0.1144404
```

```
br42
```

```
## [1] 0.11211
```

```
br22-br42
```

```
## [1] 0.002330465
```

Nú er brierScore-ið hjá seinna minna, en bara um u.þ.b. 0.00233. Ég játa að ég veit ekki alveg enn hvað það þýðir milli styrkleikjanna þeirra, hvort það sé rosalega mikill munur eða bara smár munur.

Næst er að reikna AUC þeirra.

```
Auc22 <- AUC(predict(ans22, type = "response"), hashLim100$correct)
Auc42 <- AUC(predict(ans42, type = "response"), hashLim100$correct)

Auc22
```

```
## [1] 0.8205122
```

```
Auc42
```

```
## [1] 0.8311404
```

```
Auc42 - Auc22
```

```
## [1] 0.01062822
```

Hér sést að AUC er stærra hjá seinna modelinu en fyrra, um 0.010628. Ef ég skil Auc rétt, þá er seinni modelið með betra AUC.

En er þetta betra Brier og betra AUC nokkuð nógu mikið betra til að leifa tvöfalt lengri keyrslutíma fyrir eitthvað sem tekur svona langan tíma? Er ekki ennþá viss.