Report Template coursework assignment A - 2021

CS4125 Seminar Research Methodology for Data Science

Thomas Bos (4543408), Daniël van Gelder (4551028), Jessie van Schijndel (5407397)

20/04/2021

Contents

1 Part 1 - Design and set-up of true experiment

1.1 The motivation for the planned research

The coronavirus pandemic has had a great impact on many aspects of society. University education, in particular, has changed significantly. As education in many countries has shifted from physical lectures to online teleconferencing lectures, concerns have been raised with regards to the effectiveness of this method of education. While the technological developments surrounding teleconferencing have enabled an almost seamless transition from offline to online education, it may be that the lack of a physically present lecturer affects the comprehensibility of the lecture material for students. With this research, we aim to address whether the students' understanding of the lecture material is affected by a different learning setting (i.e., from home watching an online lecture). The results may reveal whether online education is a way to move forward out of the pandemic. Moreover, if the results indicate no significant change in student understanding of material it may open up the way for new form of education, where students could enroll into "digital universities" without needing to be present at any time.

1.2 The theory underlying the research

Figlio et al. (2013) presented, according to them, the first experimental evidence on the effects of live versus online instruction. In this research, participants took an entire microeconomics course either only attending live lectures or online lectures. Exam performance was then compared between both groups and all students which did not volunteer to participate in the experiment but did still follow the course. Result showed that there is a modest difference in exam scores in favour of the students only attending live lectures, although the authors state that the experiments had many limitations and that further research is necessary. In contrary, a more recent survey by Nguyen (2015), which summarizes results of multiple studies, has found that 92% makes online education to be at least as effective, if not better, than live education. However, it is also important to recognize other issues that may arise when switching teaching modalities, which becomes clear when such a shift is forced due to, for example, the onset of COVID-19. In a very recent study by Finnegan (2021), results showed that while results are marginally worse after the shift to online teaching, student experience has deteriorated when their learning environment is suddenly changed, especially with students with poor online access.

1.3 Research questions

Our research question is the following: "How is students' understanding of lecture material affected by attending the lecture live rather than online?". We describe our null hypothesis and alternative hypothesis in the section on suggested statistical analyses.

1.4 The related conceptual model

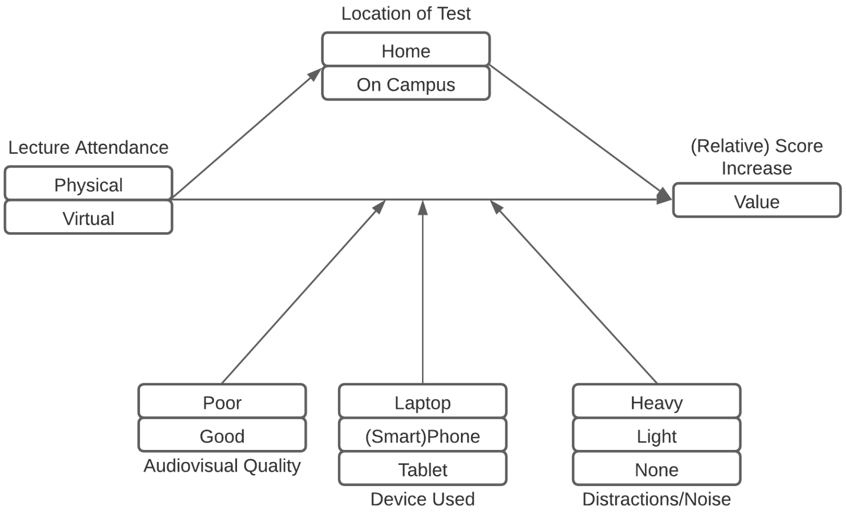

Figure 1: The conceptual model underlying the research.

The figure above displays the conceptual model for this research. The following sections describe the conceptual model for each type of variable:

1.4.1 Independent Variable (IV)

The IV of this research is whether the participant (student) attends the lecture physically or from home through online teleconferencing.

1.4.2 Dependent Variable (DV)

The DV of this research is the relative score increase on the test that students make. Before the experiment the participants make a small test regarding the lecture material for which the score is expected to be low as the participants are expected to have no prior knowledge regarding the material. Then after the lecture the students make the same test regarding the lecture material. The relative increase (or unlikely decrease) of score will be the DV.

1.4.3 Mediating Variable

As the students perform the test in a different setting (from home or on campus) depending on the IV. The change in setting is expected to have a mediating effect on the relationship between the IV and DV. The Mediating Variable is thus whether the students take the *test* at home or on campus.

1.4.4 Moderating Variable

There are several factors which may a moderating effect on the relationship between the IV and the DV which are difficult control on the experiment. These mostly have to do with the environment in which the lecture is attended. The following list describes the specific variables which are believed to have this moderating effect:

- (relates only to online lecture) video/audio quality
- (relates only to online lecture) device that is used to attend lecture (e.g. laptop, tablet, smartphone)
- (relates to both physical and online lecture) presence of noise and/or distraction in environment of watching lecture

1.5 Experimental Design

In order to determine the difference between live and online lectures on students with respect to acquired knowledge the experimental design Pre-test Post-test randomized controlled trail was chosen. This means the participants can be tested before and after the lecture so that the difference in test results, the dependent variable, can be used as an indicator of knowledge gained from said lectures. For the lecture itself, the participants will be divided randomly over live and online groups such that the live group will attend a lecture face-to-face with a lecturer, and the online group will attend the lecture via an online platform such as Zoom. In order to minimize the influence of moderating variables such as video/audio quality and distractions, the online group will watch the lecture in a quiet, moderated environment on identical systems specifically set up for the experiment.

1.6 Experimental procedure

First, we ask all students in the class who have agreed to participate in our experiment to perform a pre-test a day before the lecture. The pre-test will consist of questions composed by the teacher giving the lecture. The questions should reflect the main learning goals of the lecture. Ideally, this pre-test is done in a controlled setting on campus. If this is not possible due to governmental restrictions, the pre-test is performed online. All students perform the pre-test at the same time. After the pre-test, students are assigned to either the live lecture condition or the online lecture condition. To reduce unexplained variability, we will opt for a randomized block design. We will divide similar participants into blocks based on their pre-test scores. Then, we randomly assign participants from each block to the live condition or the online condition. Students in both conditions will follow the same lecture at the same time. A day after the lecture, the students perform a post-test. Just like the pre-test, the post-test will consist of questions composed by the teacher giving the lecture and should reflect the main learning goals of the lecture. However, the questions from the pre-test should not be repeated. Again, this post-test is ideally done in a controlled setting on campus, but may have to be performed online.

1.7 Measures

In the experiment, both participant groups will take a pre-test and a post-test. This test aims to evaluate the participants' comprehension of the lecture material. The pre-test is meant to serve as a baseline measurement to rule out any pre-existing knowledge of the participants. Both tests will be identical and will be in the form of a multiple choice exam of ten questions to be taken in a short time span (10 minutes). The score of the test is defined as the proportion of correct answers. The measure of the experiment is the ratio between these to tests for each particpant: the score of the post-test divided by the score of the pre-test.

1.8 Participants

Participants should be students and could be recruited by asking for volunteers on a university campus. In order to eliminate differences from different backgrounds, participants should be chosen from the same university programme (e.g. Computer Science or Mathematics). More participants is better, as this allows us to average out individual factors. Ideally, the live and online groups are about 300 participants each, both filling a lecture hall. There will be a variation in age, but because students are usually around the same age in a given univerisity program we can expect this variance to be small. A small compensation could be offered in return as a sign of appreciation.

1.9 Suggested statistical analyses

First, we determine our null hypothesis H_0 and alternative hypothesis H_1. Our null hypothesis states that there is no difference in student understanding of the lecture material between the two different conditions. Our alternative hypothesis states that there is a difference in student understanding. We create two linear models to predict student understanding of lecture material. First, we create a model which has only an intercept. This model does not use the information about which condition a participant was in. This model will be referred to as m_0. Second, we create a model which does include this information as a predictor. This model will be referred to as m_1. Then, we compare the fits of the two models to the data. We determine whether m_1 fits significantly better than m_0 through an ANOVA F-test. If this is not the case, we cannot reject our null hypothesis. We may also inspect the significance of the parameters of m_1. If the effect of the condition parameter is not significant, we cannot reject our null hypothesis.

2 Part 2 - Generalized linear models

2.1 Question 1 Twitter sentiment analysis (Between groups - single factor)

2.1.1 Conceptual model

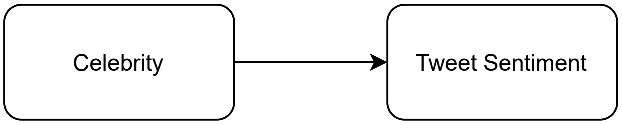

Figure 2: The conceptual model for the research question. The model is simple, as we only have tweet sentiment score and which celebrity the tweet is about.

2.1.2 Collecting tweets, and data preparation

The data used in this research has been collected on May 20th, 2021.

2.1.3 Homogeneity of variance analysis

From the boxplot containing the distribution of tweet sentiments for all three celebrities we can conclude that there is a visible difference in sentiment variance. Performing Levene's test verifies this. In the results of that test we can see that the effect is significant ($F_{2,897} = 6.79, p < 0.05$). This indicates that there is variance inequality between all three groups of tweets.

```
library(car)
library(pander)
```

```
boxplot(semFrame$score ~ semFrame$Candidate)
```

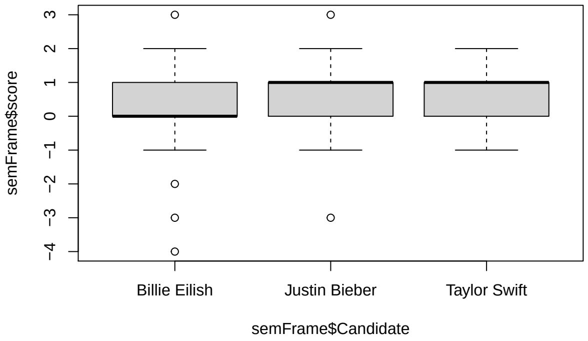

Figure 3: Boxplot of the sentiment values of the tweets for each celebrity.

```
pander(leveneTest(semFrame$score, semFrame$Candidate, center = median))
```

Table 1: Levene's Test for Homogeneity of Variance (center = median)

	Df	F value	Pr(>F)
group	2	6.79	0.001183
	897	NA	NA

2.1.4 Visual inspection Mean and distribution sentiments

Analysing the bar graph of sentiment counts for all tweets for each celebrity, we can see that in general tweets about Justin Bieber are the most positive, with most tweets having a sentiment around 1, while tweets about Taylor Swift are a bit less positive. Tweets about Billie Eilish are the most negative, or rather, most neutral with the most tweets having sentiment values of 0.

```
library(ggplot2)
p <- semFrame %>% ggplot( aes(x=score)) +
  geom_histogram( color="#e9ecef", alpha=0.6, position = 'identity') +
  facet_grid(. ~ Candidate)
```

```
plot(p)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

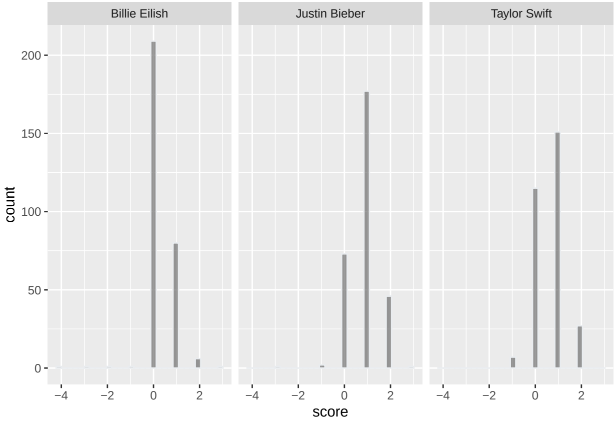

Figure 4: Bar plot for the tweet sentiment values for each celebrity.

2.1.5 Frequentist approach

2.1.5.1 Linear model Comparing the AIC of a null model (m_0, $AIC = 1942$) and a model where celebrity is added as a predictor (m_1, $AIC = 1827$) shows that there is an improvement with the latter model with respect to the quality of the fit, as the AIC is lower with m_1. Performing an F-test shows that there is a significant ($F = 63.49$, p. < 0.01) difference in fit between both models. This means that knowing the celebrity gives information about the distribution of tweet sentiments in the data set.

```
library(pander)
library(multcomp)
library(stats)
library(AICcmodavg)
semFrame$CandidateF <-factor(semFrame$Candidate,
                             levels=c("Justin Bieber", "Taylor Swift", "Billie Eilish"),
                             labels=c("Justin Bieber", "Taylor Swift", "Billie Eilish"))
m0 = lm(score ~ 1, data=semFrame)
m1 = lm(score ~ CandidateF, data=semFrame)
models = list(m0,m1)
model.names = c("Model 0", "Model 1")
```

```
pander(aictab(cand.set=models, modnames=model.names))
```

	Modnames	K	AICc	Delta_AICc	ModelLik	AICcWt	LL	Cum.Wt
2	Model 1	4	1827	0	1	1	-909.4	1
1	Model 0	2	1942	115.1	1.005e-25	1.005e-25	-969	1

```
pander(anova(m0, m1))
```

Table 3: Analysis of Variance Table

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
899	453.9	NA	NA	NA	NA

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
897	397.6	2	56.28	63.49	1.633e-26

2.1.5.2 Post Hoc analysis Performing Post Hoc analysis allows us to determine whether there is a significant difference in tweet sentiments between all three celebrities, and how the distribution changes between them. The results, obtained by performing Tukey's Honest Significant Difference test, show that there is a significant difference in tweet sentiments for all three celebrities. Tweets about Justin Bieber ($M = 0.89, SD = 0.69$) are the most positive, while those about Billie Eilish ($M = 0.28, SD = 0.63$) are most negative, confirming our findings in the visual inspection. Taylor Swift's ($M = 0.66, SD = 0.67$) are a bit less positive than Justin Bieber's.

```
library(stats)
mean(semFrame[semFrame$CandidateF=="Justin Bieber", "score"])
sd(semFrame[semFrame$CandidateF=="Justin Bieber", "score"])
mean(semFrame[semFrame$CandidateF=="Taylor Swift", "score"])
sd(semFrame[semFrame$CandidateF=="Taylor Swift", "score"])
mean(semFrame[semFrame$CandidateF=="Billie Eilish", "score"])
sd(semFrame[semFrame$CandidateF=="Billie Eilish", "score"])
res.aov <- aov(score~CandidateF, data = semFrame, na.action = na.exclude)
```

```
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = score ~ CandidateF, data = semFrame, na.action = na.exclude)
##
## $CandidateF
##                                   diff        lwr        upr    p adj
## Taylor Swift-Justin Bieber  -0.2300000 -0.3576174 -0.1023826 7.61e-05
## Billie Eilish-Justin Bieber -0.6066667 -0.7342841 -0.4790493 0.00e+00
## Billie Eilish-Taylor Swift  -0.3766667 -0.5042841 -0.2490493 0.00e+00
```

2.1.5.3 Report section for a scientific publication In order to determine whether sentiment of tweets depend on the celebrity they are about, a null model and a model with celebrity added as a independent variable predictor have been constructed. Results showed that the fit was significantly better ($F_{2,897} = 63.49$, p. < 0.01) when the null model ($AIC = 1942$) had celebrity added as predictor ($AIC = 1827$). We can therefore conclude that the Twitter user base tweets differently about Justin Bieber ($M = 0.89, SD = 0.69$), Taylor Swift ($M = 0.66, SD = 0.67$), and Billie Eilish ($M = 0.28, SD = 0.63$). Further analysis using Post Hoc analysis using Tukey's Honest Significant Difference test showed that there is a significant difference in tweet sentiment distribution in all three celebrity pairs, Taylor Swift - Justin Bieber (p. < 0.01), Billie Eilish - Justin Bieber (p. < 0.01), and Taylor Swift - Billie Eilish (p. < 0.01).

2.1.6 Bayesian Approach

2.1.6.1 Model description The most complex model tested uses the celebrity as an independent variable to predict the score. As we can expect the tweet sentiments to be normally distributed around 0 with almost all data between -5 and 5 due to the tweet character limit (280 characters), a normally distributed prior with $\mu = 0$ and $\sigma = 2$ is used for μ, and we allow σ to take a value between 0.0001 and 5. To investigate the influence of Celebrity we add it as a factor and transform the intercept parameter into an index variable, giving us the following model:

$$\begin{aligned} score &\sim Normal(\mu, \sigma) \\ \mu &= a_{Candidate} \\ a_{Candidate} &\sim Norm(0, 2) \\ \sigma &\sim Uniform(0.001, 5) \end{aligned}$$

2.1.6.2 Model comparison The figure below depicts the difference in WAIC score between the null model (m_0, $WAIC = 1943$) and the null model with Celebrity added as a factor (m_1, $WAIC = 1829$). It is clear that the fit is better when the Celebrity factor is added, as the WAIC score is lower. Looking at the next figure we can see the credibility intervals for the vector parameters ($a[1]$, $a[2]$, and $a[3]$) for the mean of model m_1. It is clear that the means of all three celebrities are different from one another in the model, indicating that people tweet differently about the different celebrities.

```
library("rstan")
library("rethinking")
m0 <-ulam(
  alist(
    score ~ dnorm(mu, sigma),
    mu <- a,
    a ~ dnorm(0, 2),
    sigma ~ dunif(0.0001, 5)),
  data =semFrame ,iter = 10000, chains = 4,
    cores = 4, control=list(adapt_delta=.99), log_lik=TRUE
)
m1 <-ulam(
 alist(
   score ~ dnorm(mu, sigma),
   mu <- a[CandidateF] ,
   sigma ~  dunif(0.0001, 5),
   a[CandidateF] ~ dnorm(0, 2)),
 data = semFrame ,iter = 10000, chains = 4,
  cores = 4, control=list(adapt_delta=.99), log_lik=TRUE
)
```

```
plot(compare(m0, m1))
```

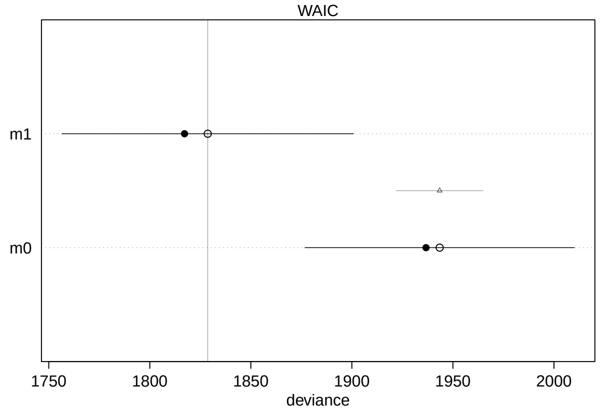

Figure 5: Comparison of the WAIC score for each model.

```
plot(precis(m1, depth=2, prob = .95))
```

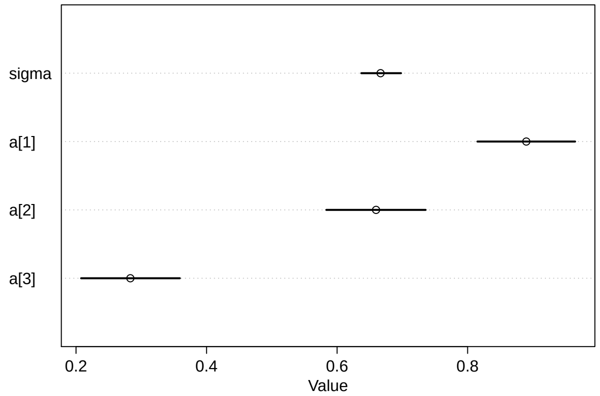

Figure 6: Confidence intervals for the parameters of the model with celebrity added as a predictor.

2.1.6.3 Comparison celebrity pair

Using _bayes.t.test_ we can analyse the difference in tweet sentiment distributions for each celebrity pair. From the plots we can deduce that for all three celebrity pairs there is a difference between the means, Justin Bieber ($Mean = 0.9$), Taylor Swift ($Mean = 0.66$), and Billie Eilish ($Mean = 0$). Furthermore, the credibility intervals for the effect size do not contain 0 for all celebrity pairs, indicating that knowing the celebrity gives information about the tweet sentiment distribution and thus the fit of a model is better if Celebrity is used as a predictor.

```
# Might have to do 'brew install jags' on Mac to make this work.
library(rjags)
devtools::install_github("rasmusab/bayesian_first_aid")
library(BayesianFirstAid)
jbSub <- subset(semFrame, (CandidateF == "Justin Bieber"))
tsSub <- subset(semFrame, (CandidateF == "Taylor Swift"))
beSub <- subset(semFrame, (CandidateF == "Billie Eilish"))
```

```
plot(bayes.t.test(jbSub$score, tsSub$score))
```

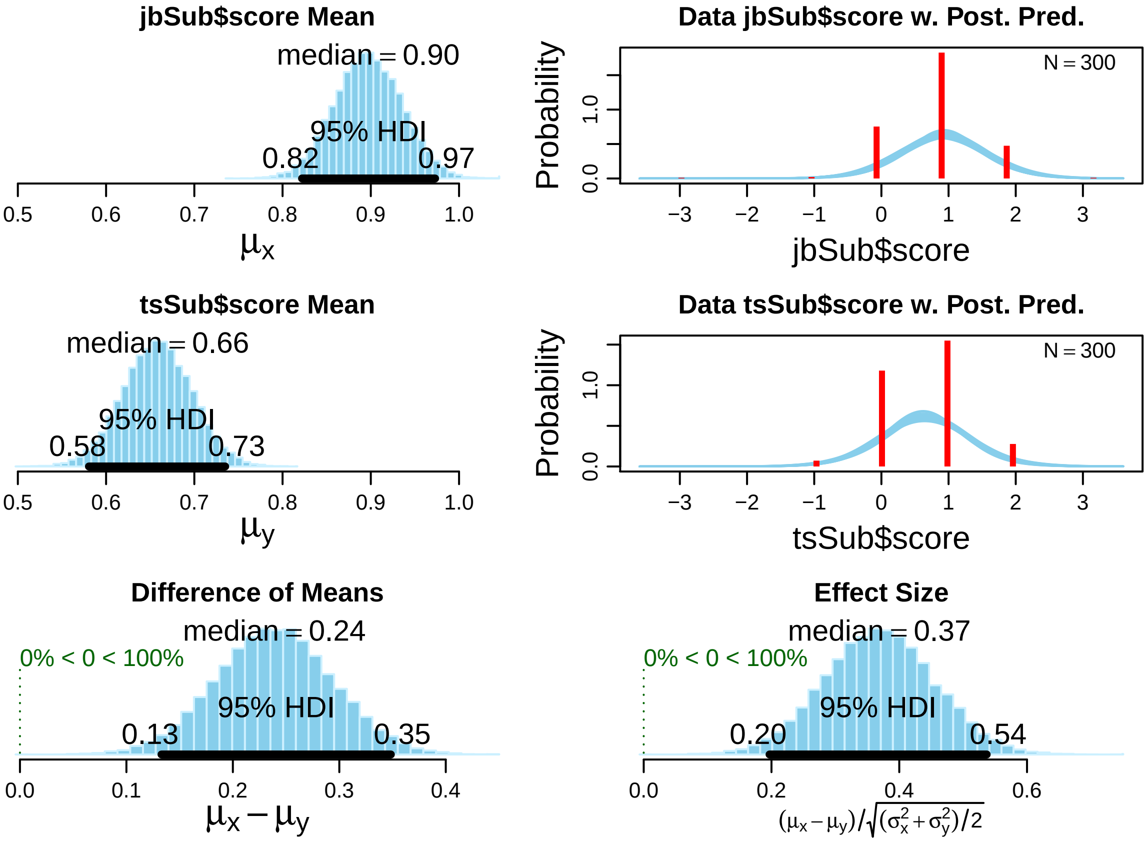

Figure 7: Results of performing the bayes t-test on the pair Justin Bieber - Taylor Swift.

```
plot(bayes.t.test(jbSub$score, beSub$score))
```

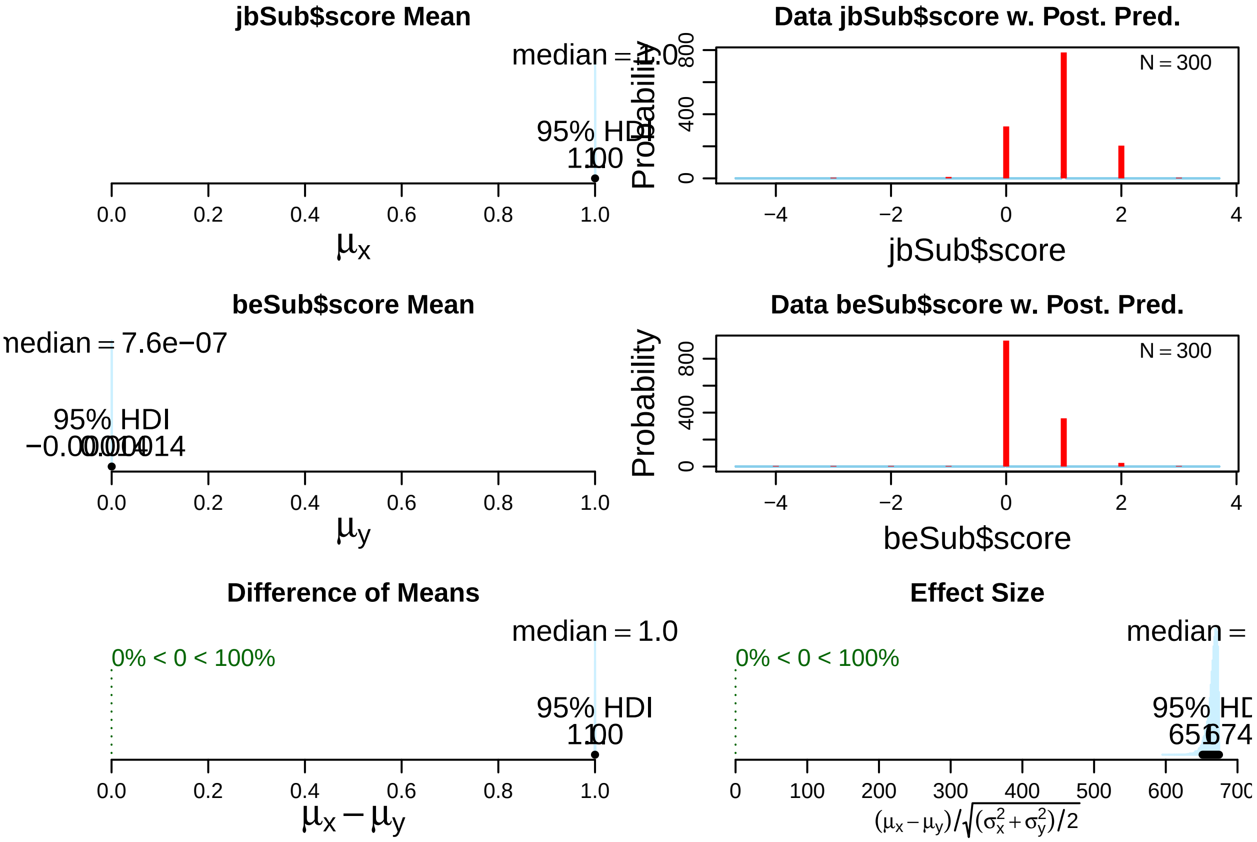

Figure 8: Results of performing the bayes t-test on the pair Justin Bieber - Billie Eilish.

```
plot(bayes.t.test(tsSub$score, beSub$score))
```

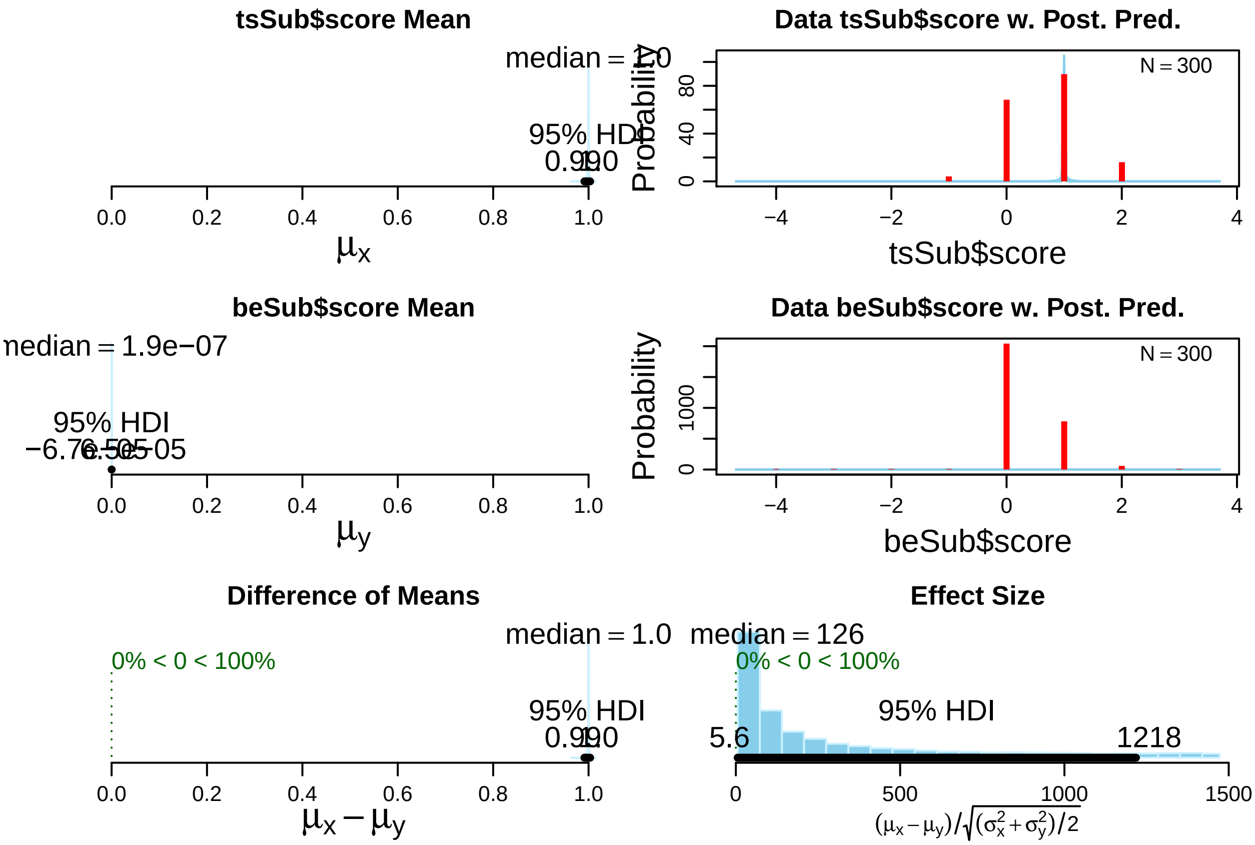

Figure 9: Results of performing the bayes t-test on the pair Taylor Swift - Billie Eilish.

2.2 Question 2 - Website visits (between groups - Two factors)

2.2.1 Conceptual model

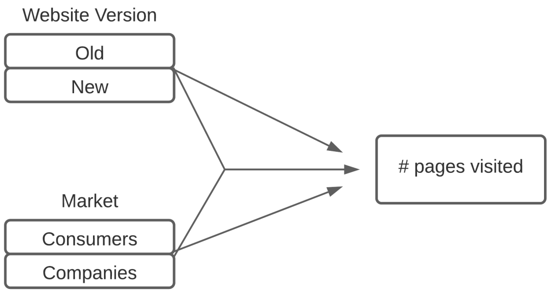

Figure 10: The conceptual for the research question about modeling the amount of page visits.

2.2.2 Visual inspection

The first step in our inspection is to observe the distribution of the dependent variable overall regardless of the factors. We plot a histogram of all the page visits:

```r
library(ggplot2)
library(magrittr)
library(dplyr)
web_visit <- read.csv("webvisit0.csv")
web_visit$version <- factor(x=web_visit$version, labels=c("old", "new"))
web_visit$portal <- factor(x=web_visit$portal, labels=c("consumers", "companies"))
```

```r
p <- web_visit %>% ggplot(aes(x=pages)) + geom_histogram(bins=10)
plot(p)
```

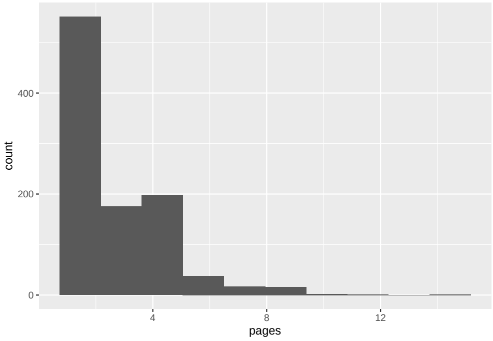

Figure 11: Histogram showing page visits on the website over all factors.

To determine whether either of the IV's have an effect on the distribution of the page visits we observe the distributions under different instantiations of the IV's in the following figure:

```r
p <- web_visit %>% ggplot( aes(x=pages)) + geom_histogram(bins=10) + facet_grid(portal ~ version)
plot(p)
```

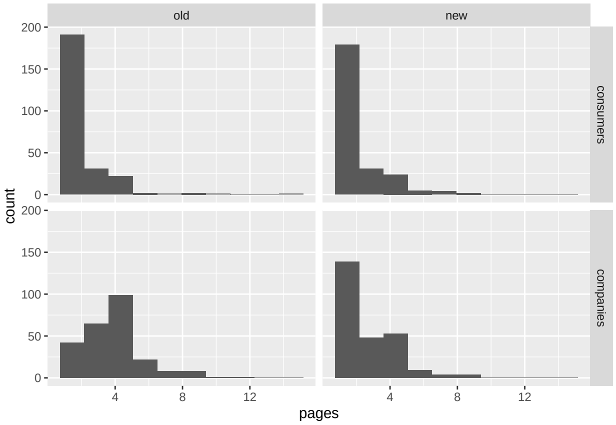

Figure 12: Histograms of page visits on the website per independent variable.

The following table shows summary statistics of the page visits across groups of the IV's:

```
grouped <- group_by(web_visit, version, portal)
pander(summarise(grouped, mean=mean(pages), sd=sd(pages)), caption = "Summary statistics across groups
```

```
## `summarise()` has grouped output by 'version'. You can override using the `.groups` argument.
```

Table 4: Summary statistics across groups of the IV's

version	portal	mean	sd
old	consumers	1.992	1.582
old	companies	4.033	1.688
new	consumers	2.053	1.477
new	companies	2.72	1.627

To further assist our understanding of the distribution we will also draw boxplots of the same data to observe differences in the distribution:

```
p <- web_visit %>% ggplot( aes(y=pages)) + geom_boxplot() + facet_grid(portal ~ version, scales = "free
plot(p)
```

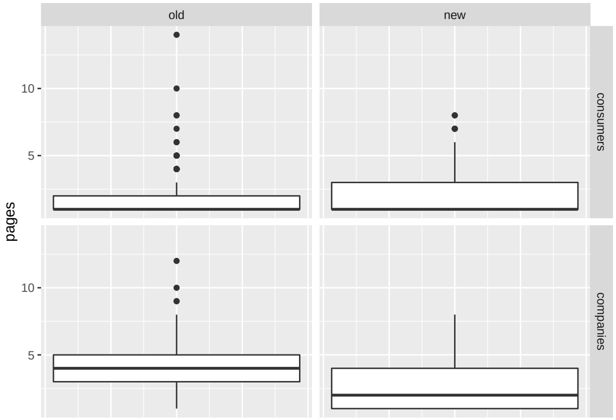

Figure 13: Boxplots of page visits on the website per independent variable

Based on the figures it seems that the IV's do affect the distribution of the DV. While the distributions between the old and new website for consumers do not show clearly different distributions, in the other cases there seems to be a significant difference. In particular, the distributions over the old and new website for companies shows a clearly different distribution in the histograms as well as in the boxplots.

2.2.3 Normality check

Again, we take the histograms over all the IV's and plot a density on top of it to see if a normal distribution can be observed in the data.

```
p <- web_visit %>% ggplot(aes(x=pages)) +
  geom_histogram(aes(y=..density..), bins=10, colour = "white", fill="grey75") +
  facet_wrap(portal~version, scales = "free") +
  geom_density(aes(y=..density..), colour="blue")
plot(p)
```

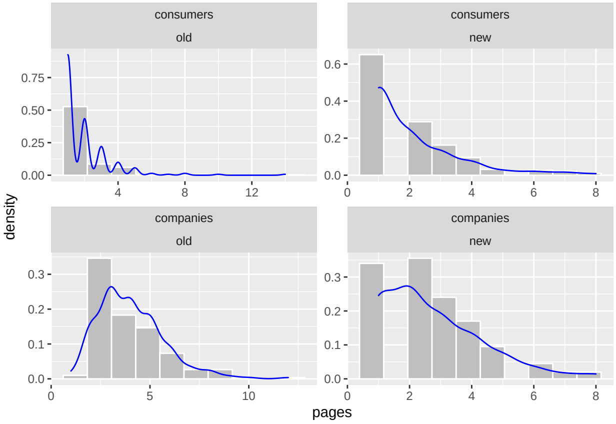

Figure 14: Density plots fitted to data of page visits

The figure shows that in almost all cases the distribution of the data does not fit to a normal distribution, except for the case of the old website on the company portal. This does not have to be an issue for the general linear model analysis as the assumption over the distribution of the data only applies to the errors. The assumption is that the errors are normally distributed. However, for the analysis we will consider a Poisson distribution which will be a better fit to the distributions visible in the density plots. The companies-old plot seems like a Poisson distribution with $\lambda = 4$ whereas the other plots seem more like a Poisson distribution with approximately $\lambda = 1$.

2.2.4 Frequentist Approach

2.2.4.1 Model analysis First we perform Levene's Test to determine the homogeneity of variance.

```
library(car)
library(pander)
pander(leveneTest(web_visit$pages, interaction(web_visit$version , web_visit$portal)))
```

Table 5: Levene's Test for Homogeneity of Variance (center = median)

	Df	F value	Pr(>F)
group	3	2.562	0.05358
	995	NA	NA

The p-value of Levene's Test in this case is larger than 0.05 and this thus indicates that the variance is homogeneous. Our assumptions on the data which allows for linear model fitting still hold.

The following tables show the results of fittings linear models and comparing them using ANOVA. To compare the models and test a significant deviance between the models, Pearson's chi-squared test is used.

The first table shows whether adding the website version as a predictor for the page visits has a significant effect. The results show indeed that there is a significant effect. The second table indicates the same, but in this case for the portal of the website (i.e. market), this effect is also significant. The third table indicates whether there is an interaction effect between the two factors, which is also significantly demonstrated. The final table shows the combined effect of the two factors as well as the interaction.

```
model0 <- glm(pages ~ 1 , data = web_visit, na.action = na.exclude, family = "poisson")
model1 <- glm(pages ~ version , data = web_visit, na.action = na.exclude, family = "poisson")
model2 <- glm(pages ~ portal , data = web_visit, na.action = na.exclude, family = "poisson")
model3 <- glm(pages ~ version + portal , data = web_visit, na.action = na.exclude, family = "poisson")
model4 <- glm(pages ~ version + portal + version:portal , data = web_visit, na.action = na.exclude, fam
```

```
pander(anova(model0,model1, test = "Chisq"), caption = "Website version as main effect on page visits")
```

Table 6: Website version as main effect on page visits

Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
998	1067	NA	NA	NA
997	1033	1	34.25	4.85e-09

```
pander(anova(model0,model2, test = "Chisq"), caption = "Portal type as main effect on page visits")
```

Table 7: Portal type as main effect on page visits

Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
998	1067	NA	NA	NA
997	898.8	1	168.2	1.871e-38

```
pander(anova(model3,model4, test = "Chisq"),caption = "Interation effect on top of two main effects")
```

Table 8: Interation effect on top of two main effects

Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
996	862	NA	NA	NA
995	834	1	28.01	1.205e-07

```
pander(anova(model4, test = "Chisq"),caption = "Effect of Website version, Portal type and interaction
```

Table 9: Effect of Website version, Portal type and interaction effect on page visits

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)
NULL	NA	NA	998	1067	NA
version	1	34.25	997	1033	4.85e-09
portal	1	170.8	996	862	5.015e-39

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)
version:portal	1	28.01	995	834	1.205e-07

The results justify a further investigation into the effects of the factors. However, we will first also evaluate the goodness of fit of the final statistical model through a Akaike Information Criterion (AIC) comparison:

```
library(AICcmodavg)
models <-list(model0, model1, model2, model3, model4)
model.names <-c("model0","model1","model2","model3","model4")
```

```
pander(aictab(cand.set = models, modnames=model.names))
```

	Modnames	K	AICc	Delta_AICc	ModelLik	AICcWt	LL	Cum.Wt
5	model4	4	3554	0	1	1	-1773	1
4	model3	3	3580	26	2.264e-06	2.264e-06	-1787	1
3	model2	2	3614	60.85	6.113e-14	6.113e-14	-1805	1
2	model1	2	3748	194.8	5.115e-43	5.115e-43	-1872	1
1	model0	1	3781	227	5.104e-50	5.104e-50	-1889	1

The analysis shows that the final model had the best goodness of fit and, in fact, captures all predictive power that could be found in the full set of models.

2.2.4.2 Simple effect analysis As we have previously found a two-way interaction effect between the version and portal factors of the experiment, we will conduct a Simple Effect analysis to explore this interaction effect.

The following bar plot shows how the page visits vary over the factors.

```
bar <- ggplot(web_visit, aes(portal , pages, fill = version))
bar + stat_summary(fun.y = mean, geom = "bar", position="dodge")
```

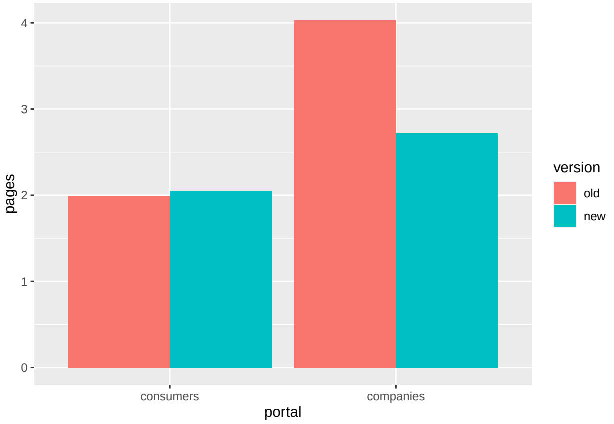

Figure 15: Bar plot showing the mean page visits per independent variable.

The plot clearly shows that there is a greater difference between the old and new versions of the website for the company portal. The Simple effect analysis will be conducted through a linear model fitted to contrasts on the consumer and company variables.

```
web_visit$simple <- interaction(web_visit$version, web_visit$portal)
levels(web_visit$simple)
```

```
## [1] "old.consumers" "new.consumers" "old.companies" "new.companies"
```

```
contrastConsumers <-c(1,-1,0,0) #Only the consumer portal data
contrastCompanies <-c(0,0,1,-1) #Only the company portal data
SimpleEff <- cbind(contrastConsumers,contrastCompanies)
contrasts(web_visit$simple) <- SimpleEff
```

```
simpleEffectModel <-lm(pages ~ simple , data = web_visit, na.action = na.exclude)
pander(summary.lm(simpleEffectModel))
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	2.699	0.0505	53.45	9.614e-295
simplecontrastConsumers	-0.03051	0.07166	-0.4258	0.6703
simplecontrastCompanies	0.6563	0.07117	9.222	1.695e-19
simple	1.354	0.101	13.4	8.88e-38

Table 12: Fitting linear model: pages ~ simple

Observations	Residual Std. Error	R^2	Adjusted R^2
999	1.596	0.2079	0.2056

The Simple effect analysis demonstrates no significant difference between page visits for users in the consumer portal using new and old versions of the website. However, it demonstrates that there is a significant difference for users in the company portal.

```
grouped <- group_by(web_visit, version, portal)
pander(summarise(grouped, mean=mean(pages), sd=sd(pages)), caption = "Summary statistics across groups
```

2.2.4.3 Report section for a scientific publication

```
## ‘summarise()‘ has grouped output by ’version’. You can override using the ‘.groups‘ argument.
```

Table 13: Summary statistics across groups of the IV’s

version	portal	mean	sd
old	consumers	1.992	1.582
old	companies	4.033	1.688
new	consumers	2.053	1.477
new	companies	2.72	1.627

A Poisson general linear model was fitted on the amount of page visits for users of a website, taking as independent variables the website version (new or old) and the user portal (consumer or company). The analysis found a significant main effect ($F(1, 995) = 36.2,\ p. < 0.01$) for the website version as well as for the portal ($F(995, 1) = 178.8,\ p. < 0.01$). Moreover, the analysis found a significant two-way interaction effect ($F(1, 995) = 46.2,\ p. < 0.01$). Overall, the old website version on the company portal had the highest mean page visits: 4.03 ($\sigma = 1.69$), followed by the new website version for the company portal: 2.72 ($\sigma = 1.63$). The consumer portals had a mean amount of page visits of 1.99 ($\sigma = 1.58$) and 2.05 ($\sigma = 1.48$) for the old and new website version respectively. A Simple Effect analysis was conducted to further examine the two-way interaction between the independent variables. It revealed a significant ($t = 9.22,\ p.0.01$) difference for website version in the company portal of the website but no significant effect ($t = 13.4,\ p = 0.67$) for the website version in the consumer portal.

2.2.5 Bayesian Approach

2.2.5.1 Model description Our most extensive Bayesian model captures each individual factor as well as the interaction of the factors and is defined as follows:

$$
\begin{aligned}
\text{PageVisits} &\sim \text{Poisson}(\lambda) \\
\lambda &= a + b \cdot \text{Version} + c \cdot \text{Portal} + d \cdot \text{Version} \cdot \text{Portal} \\
a &\sim N(1.5, 5) \\
b &\sim N(0, 5) \\
c &\sim N(0, 5) \\
d &\sim N(0, 5)
\end{aligned}
$$

Here the functions Poisson, N and U denote the probability functions of the Poisson, Normal and Uniform distributions respectively.

The Poisson distribution was chosen as a prior as previous plots of the page visits shows the distribution to follow a Poisson distribution. By means of visual inspection the parameter $\lambda = 1.5$ was chosen. See the figure below for a fit of the distribution to the data with these parameters. The other parameters are fitted to a normal distribution around 0 with a standard deviation of 5 to reflect the uncertainty of the effects that the variables may have.

```r
p <- web_visit %>%
  ggplot( aes(x=pages)) +
  geom_histogram(aes(y =..density..), bins=15) +
  stat_function(n = max(web_visit$pages), fun = dpois, geom="line", args = list(lambda=1.5))
plot(p)
```

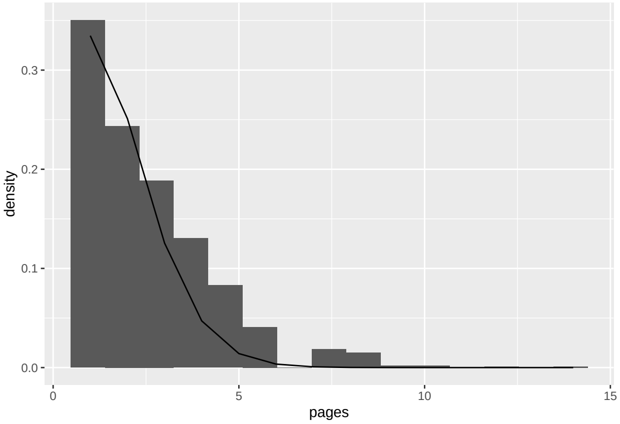

Figure 16: Poisson distribution (lambda = 1.5) fitted to data of page visits

2.2.5.2 Model comparison Besides the previously defined model, we defined simpler models which capture the effects of the factors on their own and their joint as well as interaction effect.

```
library("rstan")
library("rethinking")
web_visit <- subset(web_visit, select = c(pages,version, portal ))
web_visit$versionN <- as.numeric(web_visit$version)
web_visit$portalN <- as.numeric(web_visit$portal)
m0 <-map2stan( alist(
    pages ~ dpois(lam),
    lam <- a,
    a ~ dnorm(1.5,5)),
  data = web_visit, iter = 1000, chains = 4, cores = 4
)
m1 <-map2stan( alist(
    pages ~ dpois(lambda), lambda <- a + b*versionN,
    a ~ dnorm(1.5,5),
    b ~ dnorm(0, 5)),
  data = web_visit,iter = 10000, chains = 4, cores = 4
)

m2 <-map2stan( alist(
    pages ~ dpois(lambda), lambda <- a + c*portalN ,
    a ~ dnorm(1.5,5),
    c ~ dnorm(0, 5)),
  data = web_visit,iter = 10000, chains = 4, cores = 4
)
m3 <-map2stan( alist(
    pages ~ dpois(lambda), lambda <- a + b*versionN + c*portalN ,
   a ~ dnorm(1.5,5),
  b ~ dnorm(0, 5),
  c ~ dnorm(0, 5)),
  data = web_visit,iter = 10000, chains = 4, cores = 4
)
m4 <-map2stan( alist(
    pages ~ dpois(lambda),
  lambda <- a + b*versionN + c*portalN + d*versionN*portalN,
    a ~ dnorm(1.5,5),
  c(b,c,d) ~ dnorm(0, 5)),
  data = web_visit,iter = 10000, chains = 4, cores = 4
)
```

The figure below demonstrates the differences in WAIC between varying models. Here, `m0` is a model with just the prior, `m1` is a model capturing the effect of the website version, `m2` captures the effect of the portal, `m3` captures the effects of both the version and the portal and `m4` represents the model described earlier.

```
plot(compare(m0,m1,m2,m3,m4, func=WAIC))
```

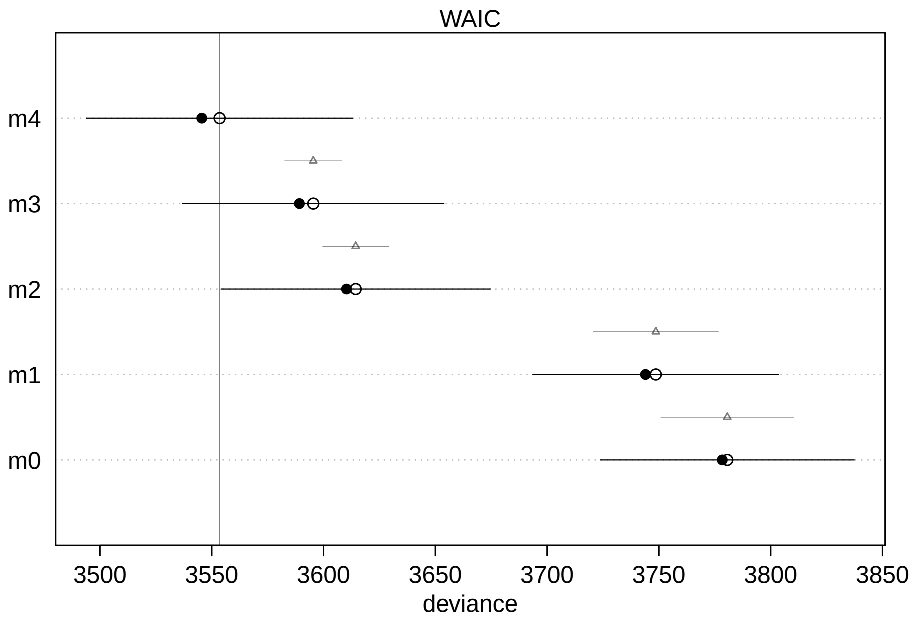

Figure 17: WAIC comparison between Bayesian models capturing effects of different variables.

```
pander(compare(m0,m1,m2,m3,m4))
```

	WAIC	SE	dWAIC	dSE	pWAIC	weight
m4	3554	59.59	0	NA	3.988	1
m3	3595	58.39	41.9	12.81	3.107	7.954e-10
m2	3614	60.25	60.87	14.75	2.049	6.064e-14
m1	3749	54.98	195.1	28.01	2.315	4.295e-43
m0	3781	56.83	227.1	29.76	1.122	4.898e-50

```
pander(precis(m4, prob= .95))
```

	mean	sd	2.5%	97.5%	n_eff	Rhat4
a	-1.433	0.4856	-2.4	-0.4641	2244	1.001

	mean	sd	2.5%	97.5%	n_eff	Rhat4
b	1.408	0.3025	0.8114	2.006	2268	1.001
c	3.382	0.3414	2.716	4.072	2264	1.001
d	-1.355	0.2082	-1.772	-0.9469	2253	1

The WAIC analysis above indicates that the extensive model described at the beginning of the section has the best goodness of fit according to the WAIC score. The table below the WAIC comparisons shows the 95% credible intervals for all the model parameters. As 0 is not present in any of the variables' 95% credible intervals, we can consider that all of the factors in the model contribute to the posterior distribution.

3 Part 3 - Multilevel model

```
library(pander)
library(car)
library(ggplot2)
library(dplyr)
library(nlme)
```

```
##
## Attaching package: 'nlme'

## The following object is masked from 'package:dplyr':
##
##     collapse
```

```
library(Rmisc)
```

```
## Loading required package: lattice
```

```
library(rethinking)
```

3.1 Visual inspection

For this exercise, we use `set1.csv`. Below we inspect the distribution of the score through a histogram of the score.

```
exp_data = read.csv("set1.csv")
p <- exp_data %>% ggplot( aes(x=score)) + geom_histogram(bins=15) + theme()
plot(p)
```

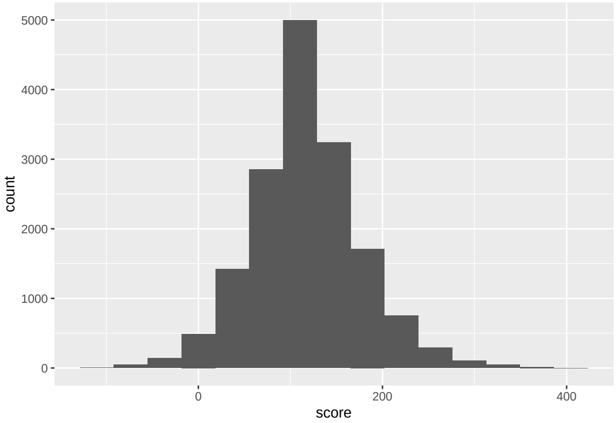

Figure 18: Histogram of score.

We also inspect the relationship between session and score through a scatterplot.

```
p <- exp_data %>% ggplot( aes(x=session, y=score)) + geom_point(alpha=0.25) + theme()
plot(p)
```

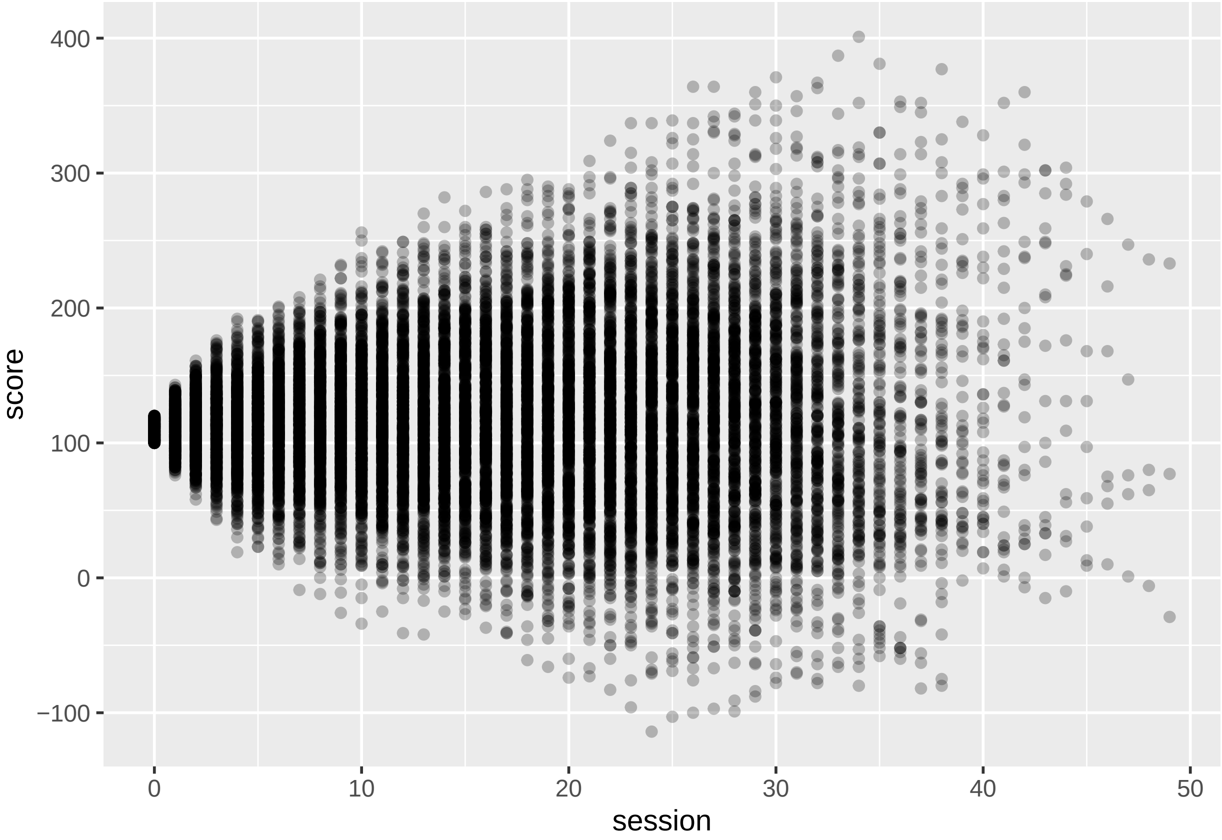

Figure 19: Scatterplot relation between score and session.

Lastly we inspect the same relationship through a plot showing the mean score and standard error over sessions.

```
tgc = summarySE(exp_data, measurevar="score", groupvars=c("session"))
p <- tgc %>% ggplot(aes(x=session, y=score, alpha=N*2)) +
    geom_errorbar(aes(ymin=score-se, ymax=score+se), width=.1) +
    geom_line() +
    geom_point()

plot(p)
```

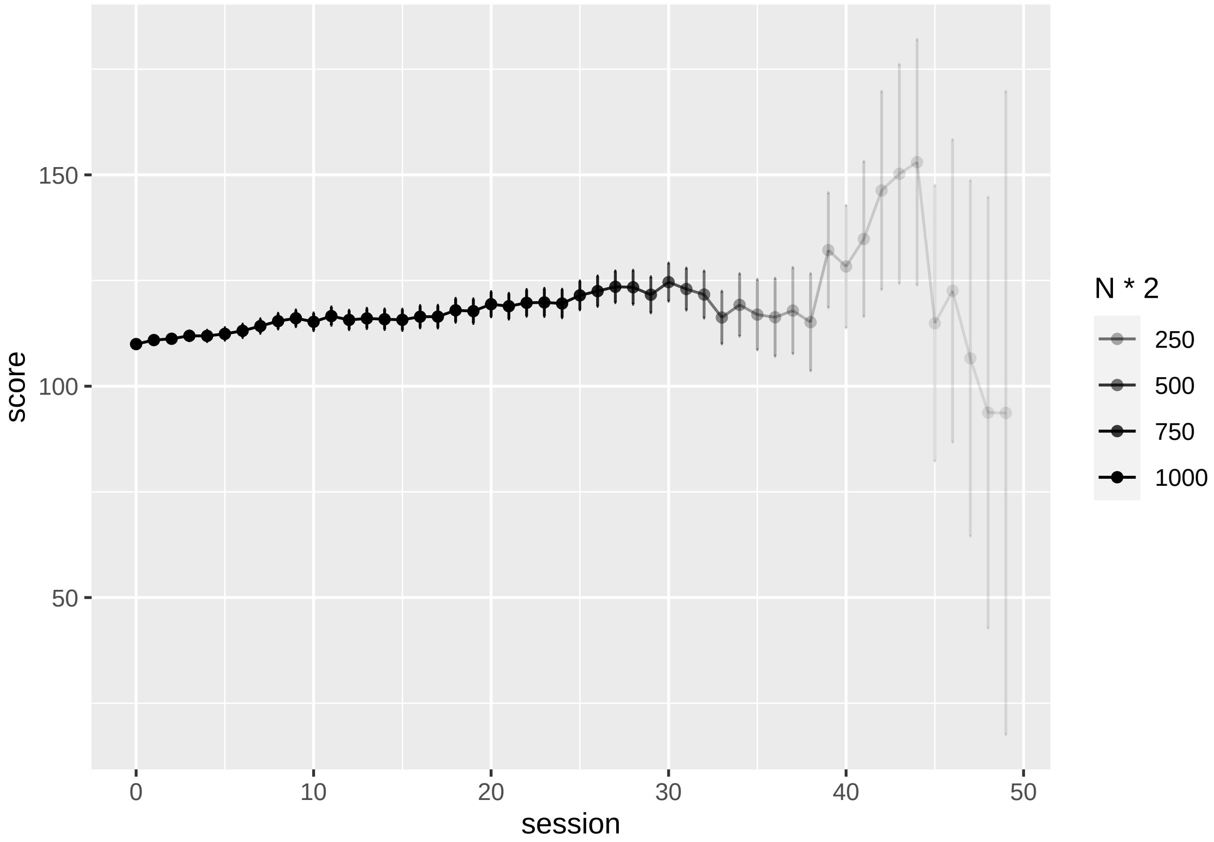

Figure 20: Mean score and standard error over sessions.

From the plots created above we conclude that there is a relation between session and score. This relation seems to be linear for up to 40-45 sessions.

3.2 Frequentist approach

3.2.1 Multilevel analysis

Below we conduct a multilevel analysis to determine if session has an impact on people's score and to determine if there is a significant variance between the participants in their score.

We check the assumption of homoskedacity using Levene's test, which determines the homogeneity of variance.

```
pander(leveneTest(exp_data$score, exp_data$session))
```

Table 16: Levene's Test for Homogeneity of Variance (center = median)

	Df	F value	Pr(>F)
group	49	80.26	0
	16078	NA	NA

The p-value of Levene's Test in this case is smaller than 0.05, indicating that there is a significant effect. This indicates that there is variance inequality between the sessions. This can also be seem in the plot showing the mean score and standard error over sessions. This means that the assumption of homoskedacity does not hold. Therefore, the results of the linear models should be interpreted with caution.

We start by comparing our two models. The first model, m_0, includes a fixed intercept (`~1`) and a random intercept, indicated by `random = 1|Subject`. This model will have a general intercept (the fixed effect intercept) and an intercept for each of the subjects. The second model m_1 includes the variable `session`. The table below shows the results of fitting these models on the data and comparing them using ANOVA.

```
model0 <- lme(score ~ 1 , random = ~1|Subject, data = exp_data, method="ML")
model1 <- lme(score ~ session , random = ~1|Subject, data = exp_data, method="ML")
pander(anova(model0, model1), caption = "Model comparison.")
```

Table 17: Model comparison. (continued below)

	call	Model	df	AIC	BIC	logLik	Test	L.Ratio	p-value
model0	lme.formula(fixed = score ~ 1, data = exp_data, random = ~1 | Subject, method = "ML")	1	3	162711	162734	-81352		NA	NA
model1	lme.formula(fixed = score ~ session, data = exp_data, random = ~1 | Subject, method = "ML")	2	4	162545	162576	-81269	1 vs 2	167.7	2.317e-38

As can be seen from these results, m_1 has the lowest AIC and thus the best goodness of fit. This result is significant ($p = 0$). We now further explore m_1.

The summary function shows an estimated fixed effect for session on the score of 0.37. With a p-value of 0.0 this fixed effect is significant.

```
summary(model1)
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: exp_data
##        AIC      BIC    logLik
##   162545.2 162575.9 -81268.58
##
## Random effects:
##  Formula: ~1 | Subject
##         (Intercept) Residual
## StdDev:     46.5146 35.06933
##
## Fixed effects:  score ~ session
##                Value Std.Error    DF  t-value p-value
## (Intercept) 111.0676  2.143371 15626 51.81911       0
## session       0.3682  0.028356 15626 12.98493       0
##  Correlation:
##         (Intr)
## session -0.206
##
## Standardized Within-Group Residuals:
##          Min           Q1          Med           Q3          Max
## -4.120041920 -0.613554431  0.009847298  0.627208531  3.952634923
##
## Number of Observations: 16128
## Number of Groups: 501
```

Inspecting the 95% confidence intervals, we can again confirm that the effects are significant, since the intervals do not include 0. The 95% confidence interval of the estimated fixed efffect for session is [0.313, 0.424]. We conclude that session has an impact on people's score.

```
intervals(model1, 0.95)
```

```
## Approximate 95% confidence intervals
##
##  Fixed effects:
##                   lower        est.       upper
## (Intercept) 106.8665678 111.0675622 115.2685566
## session       0.3126229   0.3682005   0.4237781
## attr(,"label")
## [1] "Fixed effects:"
##
##  Random Effects:
##   Level: Subject
##                    lower    est.    upper
## sd((Intercept)) 43.67471 46.5146 49.53914
##
##  Within-group standard error:
##    lower     est.    upper
## 34.68269 35.06933 35.46028
```

3.2.2 Report section for a scientific publication

To investigate whether session has an impact on people's scores and wheter there is a significant variance between the participants in their score, multilevel analysis was performed. A model with a random intercept only (m_0) and a model with a random intercept and fixed effects (m_1) were created. It was found that model m_1 showed the best goodness of fit, $p < 0.05$. There was a significant relationship between session and score, $M = 116.8139$ (95% CI 112.70, 120.93), $p = 0$.

3.3 Bayesian approach

3.3.1 Model description

In most extensive mathematical model, the score is drawn from a normal distribution with mean μ and standard deviation σ. Parameter μ is created through a linear equation involving parameters a (an intercept), a_{Subject} (a subject-specific intercept) and b multiplied by session.

$$\begin{aligned}
\text{Score} &\sim N(\mu, \sigma) \\
\mu &= a + a_{\text{Subject}}[Subject] + b * \text{session} \\
a_{\text{Subject}}[Subject] &\sim N(0, \sigma_{\text{Subject}}) \\
\sigma_{\text{Subject}} &\sim \text{Cauchy}(0, 100)T(0, \infty) \\
a &\sim N(0, 100) \\
b &\sim N(0, 100) \\
\sigma &\sim \text{Cauchy}(0, 100)T(0, \infty)
\end{aligned}$$

For the prior of the standard deviations σ_{Subject} and σ we use a half Cauchy distribution. We use a half Cauchy distribution with a large scale value of 100 for both, since we do not have much prior knowledge or belief about the true parameters. We use a half cauchy distribution as the standard deviation cannot be negative. With this large scale, the half Cauchy distribution has, unlike the normal distribution, fat tails. This makes the distribution over values away from the mean more uniform than in a normal distribution. For parameters a, a_{Subject} and b we chose to use a normal distribution as prior. We have no prior expectations on whether the intercepts a and a_{Subject} are positive or negative, therefore we choose a mean of 0 for both. For intercept a we choose a standard deviation of 100 to again show this uncertainty. For parameter b we again choose a mean of 0 as we do not know whether session has a positive or negative relation with μ. We choose a standard deviation of 100 to reflect this high uncertainty.

3.3.2 Model comparison

We select the first 100 participants from the data set. Then, we create three models with increasing complexity.

```
exp_data_1 = data.frame(exp_data)
exp_data_1 = exp_data_1[exp_data_1$Subject < 100,]
exp_data_1$Subject <- exp_data_1$Subject + 1

m0 <-ulam( alist(
    score ~ dnorm(mu, sigma),
    mu <- a ,
    a ~ dnorm(0, 100),
    sigma ~ dcauchy(0, 100)
  ),  data = exp_data_1, iter = 10000, chains = 4, cores = 4, control=list(adapt_delta=.99), log_lik=TRU
)

m1 <-ulam( alist(
    score ~ dnorm(mu, sigma),
    mu <- a + a_Subject[Subject] ,
    a_Subject[Subject] ~ dnorm(0, sigma_Subject),
    sigma_Subject ~ dcauchy(0, 100),

    a ~ dnorm(0, 100),
    sigma ~ dcauchy(0, 100)
  ),  data = exp_data_1, iter = 10000, chains = 4, cores = 4, control=list(adapt_delta=.99), log_lik=TRU
)

m2 <-ulam( alist(
    score ~ dnorm(mu, sigma),
    mu <- a + a_Subject[Subject] + b * session,

    a_Subject[Subject] ~ dnorm(0, sigma_Subject),
    sigma_Subject ~ dcauchy(0, 100),

    a ~ dnorm(0, 100),
    b ~ dnorm(0, 100),
    sigma ~ dcauchy(0, 100)
  ),  data = exp_data_1, iter = 10000, chains = 4, cores = 4, control=list(adapt_delta=.99), log_lik=TRU
)

compare(m0,m1,m2)
```

```
plot(compare(m0,m1,m2))
```

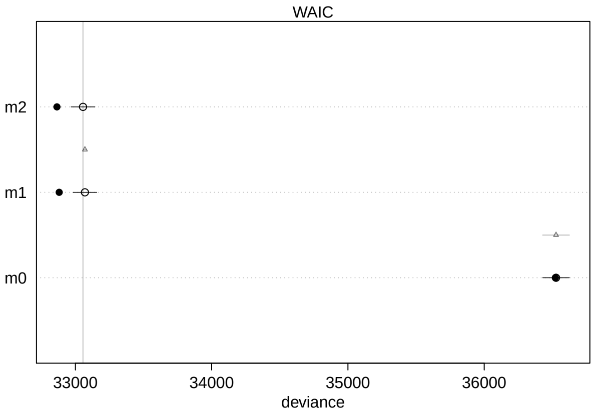

Figure 21: Comparing three Bayesian models of increasing complexity.

From the plot we can see that models m_1 and m_2 outperform model m_0. However, there is not much of a difference in WAIC scores between m_1 and m_2. The effective number of parameters ($pWAIC$) increases for both these models.

3.3.3 Estimates examination

We further investigate the difference between m_1 and m_2 by looking at the parameters.

```
pander(precis(m1))
```

```
## 100 vector or matrix parameters hidden. Use depth=2 to show them.
```

	mean	sd	5.5%	94.5%	n_eff	Rhat4
sigma_Subject	51.09	3.711	45.46	57.28	18769	1
a	114.3	5.104	106.1	122.5	408.7	1.002
sigma	36.82	0.465	36.08	37.57	19982	1

```
pander(precis(m2))
```

```
## 100 vector or matrix parameters hidden. Use depth=2 to show them.
```

	mean	sd	5.5%	94.5%	n_eff	Rhat4
sigma_Subject	51.2	3.7	45.62	57.43	17994	0.9999
a	110.6	5.415	102.1	119.6	414.3	1.011
b	0.2554	0.06492	0.1523	0.359	19392	1
sigma	36.73	0.4609	36	37.47	20850	1.001

As can be seen, the parameters of the two models are also quite similar. It seems that while b does have some effect, it is not a large one.

Below we visually examine the estimate of parameters of the model with best fit, which is m_2. We see that the uncertainty for parameters σ_{Subject} and a is relatively high compared to that for parameters b and σ.

```
plot(m2)
```

```
## 100 vector or matrix parameters hidden. Use depth=2 to show them.
```

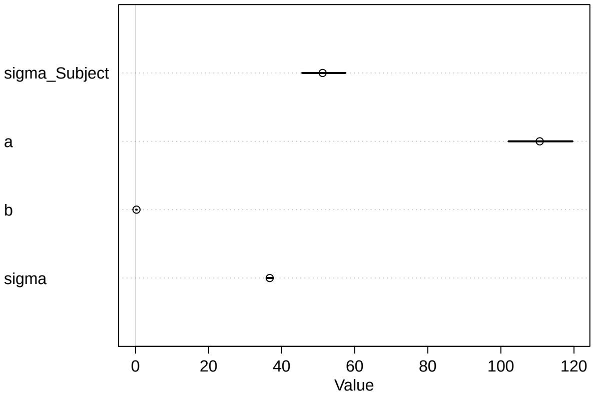

Figure 22: Parameters of m2.