Report coursework assignment A - 2021

CS4125 Seminar Research Methodology for Data Science

Nikki Bouman (4597648), Anuj Singh (5305926), Gwennan Smitskamp (4349822)

12/05/2021

Contents

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

1.1 The motivation for the planned research

The recent outbreak of the COVID-19 pandemic has changed our daily lives significantly. People are obligated to stay at home, disrupting their usual social interactions in both work and private life. The situation compels people to meet online. Typically, in such digital interactions, interlocutors can see each other by means of webcam streaming. However, this may not always be the case. Some or all interlocutors may not be visible during online dialogue, which could affect the quality of the conversation and the mutual understanding.

An important effect of the shift from face-to-face to online interaction can be revealed by studying laughter, as it is extremely contagious social behavior (Provine, 1992). Humans are very prone to unintentionally or unconsciously laugh as a social signal in any form; from a minor smile to laughing out loud. Additionally, laughing is one of the most important social signals for lubricating the flow of social interaction (Griffin et al., 2015).

1.2 The theory underlying the research

The effect of visibility on the use of gestures as a communicative function has been studied broadly (Alibali, Heath, & Myers, 2001; J. B. Bavelas, Chovil, Lawrie, & Wade, 1992; Cohen & Harrison, 1973; Cohen, 1977; Emmorey & Casey, 2001; Krauss, Dushay, Chen, & Rauscher, 1995; Rim ´e, 1982). , J. Bavelas, Gerwing, Sutton, and Prevost (2008) provide a summary of previous experiments where rate and form of gestures were compared under two conditions: where the addressee could see the speaker and where the addressee could not see the speaker. These experiments show that speakers gestured at higher rate when they communicated with mutual visibility than without. J. Bavelas et al. (2008) extended these experiments by focusing on both visibility and dialogue as a variable, finding similar results. Furthermore, they found that speakers gestured at a significantly higher rate in a telephone dialogue than in a monologue to a tape recorder, confirming that visibility is not the only variable operating in telephone conversations. These experiments showed us that visibility plays a major role in the rate of gesturing, but that people also gesture when they are not visible to each other. As laughter can be seen as a form of gesturing, these findings are relevant for this study.

Laughing together is found to be essentially collaborative (Mehu & Dunbar, 2008; Coates, 2007). Joint laughter therefore serves important means to achieve effective team meetings (Ponton, Osbourne, Greenwood, & Thompson, 2018), considering that people who laugh on video are perceived with a higher likeability than people who do not (Reysen, 2006). This social function of joint laughter emphasises the relevance of studying the occurrence, now that the majority of meetings take place online.

1.3 Research questions

We will aim to answer the following research question:

What is the effect of webcam visibility during online dialogue on the frequency of joint laughter?

When recognizing laughter we do not focus on the reason why someone is laughing. We consider anything from an awkward laugh in a moment of silence to laughing out loud about a joke as a laughter episode regardless of the context.

1.4 The related conceptual model

The conceptual model related to the research question can be viewed in Figure 1. The main question is about the effect of mutual visibility on joint laughter. The mediating variable is familiarity, which we can define as the level of friendliness or intimacy between people. This can be caused when people know each other (which we aim to avoid), but also when people find similarities in their interests and behaviours. It is even possible that people of the same gender will feel more familiar with each other. A moderating variable is the duration of the experiment. The participants will do the experiment for about one hour, which can have a negative effect on the frequency of laughter, thus on the frequency of joint laughter.

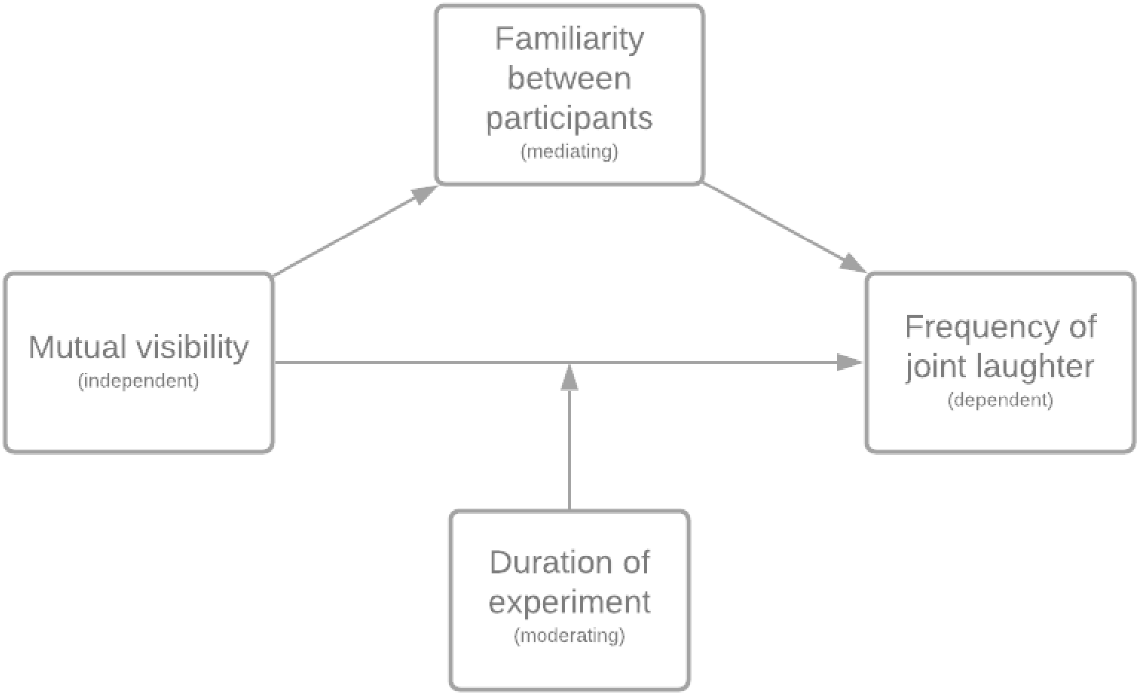


Figure 1: The conceptual model to test the effect of mutual webcam visibility on joint laughter

1.5 Experimental Design

One of the most important requirements for the setup of the experiment, was the creation of a comfortable and pleasant ambiance so that people would laugh. Therefore, it was decided that a game would comply, as the participants get the chance to interact with each other in a undemanding setting where the attention of the participants would be drawn to a task. It was reasoned that this would contribute to a reduction of awkwardness and give all the participants the option to speak and laugh. Additionally, the game needed to have a smooth flow that would automatically keep going to keep the interference of the researchers to a minimum.

The game that was chosen is called 30 Seconds. During the game, participants work together in teams (in the case of the experiment: two teams of two people) and gain points by guessing what the team member is describing. These descriptions include concepts such as famous persons, locations, movies and brands. Every team gets 30 seconds to guess as many concepts on the card as possible. Who is describing and who is guessing switches after every card.

1.6 Experimental procedure

The experiment will be set up in an online setting in a Zoom meeting. The host, one of us (not visible), will be able to send private messages containing the five words, share their screen and sound for a 30 second timer, and turn the participants' webcams on and off.

The following will repeat for every card of five words:

1. The host sends the words to the participant who has the turn to describe.
2. The host starts the 30 second timer.
3. The participant will try to describe as many words as possible, while his/her teammate will try to guess the words.
4. The timer rings, the host puts the score in the chat.

During the experiment, multiple things will happen. Each time all players have guessed and described a card (i.e. after four cards total), their webcams will switch on or off. After each player has guessed and described four cards (i.e. after sixteen cards total), the final score will be displayed and the teams will be rearranged. The previous will then repeat until every participant has been in a team with every other participant. An example of such an experiment is displayed in Figure 2. To counterbalance the experiments, half of the experiments will start with the webcams on, while the other half will start with the webcams off.

1.7 Measures

The data that has been collected includes audio and video of participants. The first step in data analysis involves annotating the signals. This will be done with a program in which we can manually select timesteps in which the participant is laughing. In the end we will have annotations for every person that contains the total amount of laughter (frequency), the amount of laughter with their webcam off, and the amount of laughter with their webcam on..

In a case where different people will annotate this data, we will first let every one of them annotate the same data sample. Then we can calculate the consistency in the annotations with for example Krippendorf's Alpha or Cohen's Kappa. When this is high enough, they can start annotating separate data.

1.8 Participants

The research is not specifically about a certain group of people, but more in general. However, we do want to aim for people with experience in an online setting and people who speak the same language. To be exact, we will conduct the experiment with people from the ages of 18 to 50 who speak dutch. The number of patricipants depends on the acceptable margin or error, which we do not know since we would have to go much more in-depth. However, since the population size basically includes more than 10.000 people and the

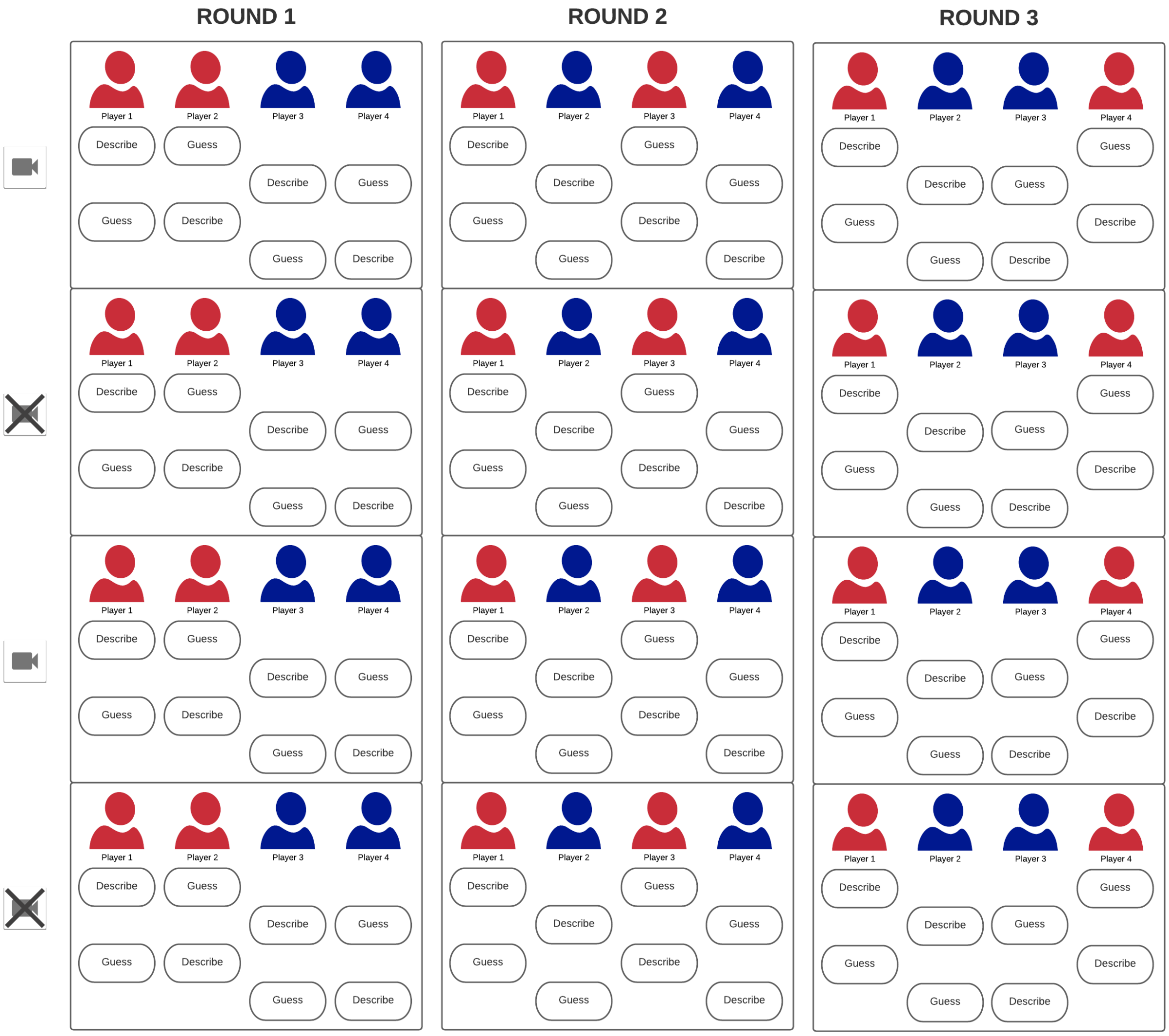


Figure 2: An example of an experimental setup.

independent variable is categorical, we should aim for around 385 participants (W.P. Brinkman, 2009). This results into around 96 experiments.

The experiments should be easy to conduct since the participants participate online; there is no need to travel. Moreover, the whole experiment should not take long. There will be 48 cards and for each card they have 30 seconds to explain. Taking some talking afterwards into account, the experiment should take 45 to 60 minutes.

To find the participants we can thus search online. Using a medium on which we state the experiment, we can find dutch people from all around the Netherlands of different age groups. They could for example sign up for a specific date, and if four people have signed up, the experiment can be held.

1.9 Suggested statistical analyses

To determine the significance of the results we will subject the data to statistical tests. To test the effect of mutual visibility (categorical, since it is either on or off) on the frequency (numerical) we will use the Wilcoxon signed rank test. The Wilcoxon signed rank test is a suitable statistical test when the measurements for a single variable are taken under two different conditions. It is similar to the paired t-test, however the paired t-test assumes that the data is normally distributed which we cannot assume for frequency of laughter. It tests the null hypothesis that the median difference of a two sets of observations is zero.

2 Part 2 - Generalized linear models

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

2.1.1 Conceptual model

Make a conceptual model for the following research question: Is there a difference in the sentiment of the tweets related to the different celebrities?

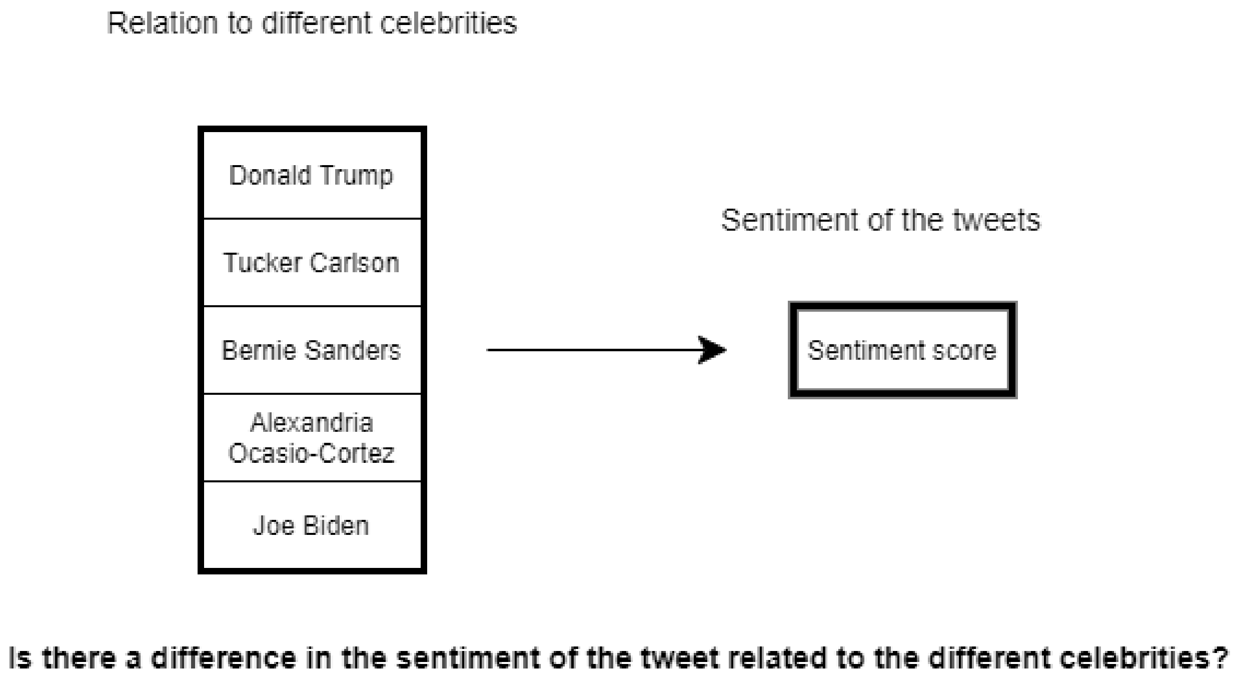

Relation to different celebrities
Donald Trump
Tucker Carlson
Bernie Sanders
Alexandria Ocasio-Cortez
Joe Biden
Sentiment of the tweets
Sentiment score
Is there a difference in the sentiment of the tweet related to the different celebrities?

Figure 3: The conceptual model for the research question: Is there a difference in the sentiment of the tweets related to the different celebrities?r

2.1.2 Collecting tweets, and data preparation

We found five celebrities in US politics: Donald Trump, Tucker Carlson, Bernie Sanders, Alexandria Ocasio-Cortez, Joe Biden. As dutch students we are not well-versed in the popular English twitter celebrities, so US politics was the best option for us to find celebrities that had enough recent Tweets for the Twitter API.

2.1.3 Homogeneity of variance analysis

```
pander(leveneTest(semFrame$score, semFrame$Celeb))
```

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

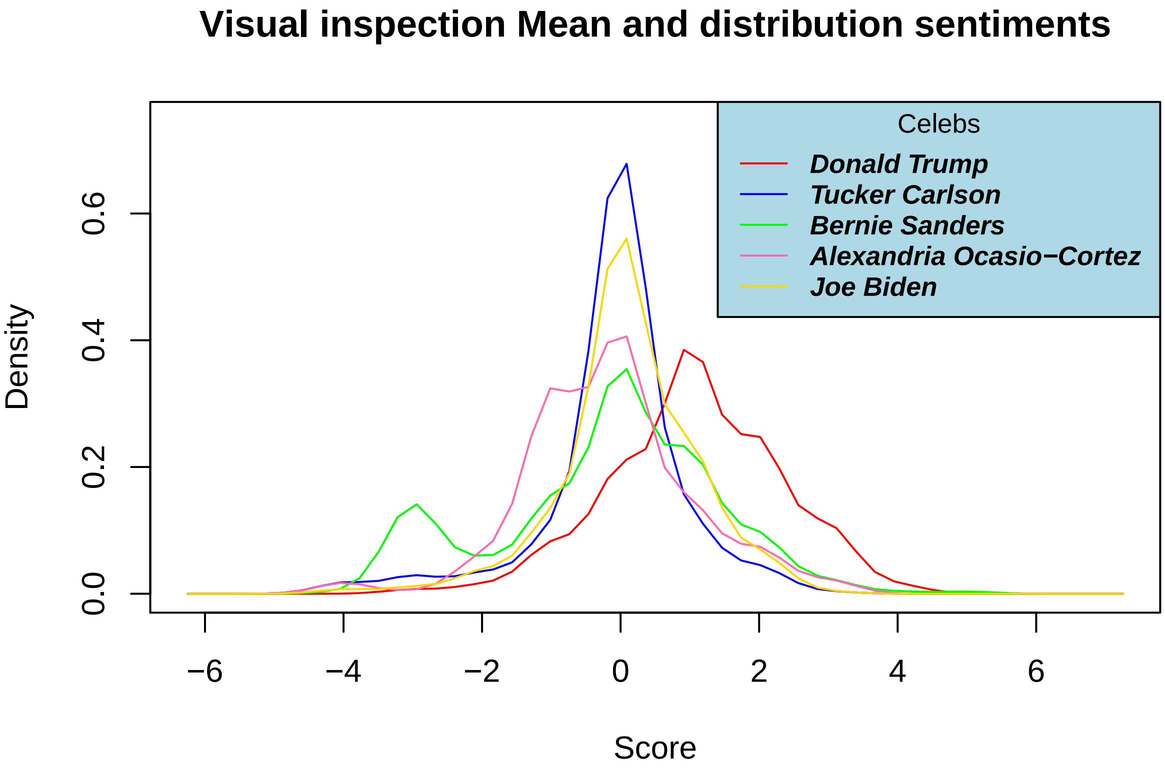
	Df	F value	Pr(>F)
group	4 1465	23.64 NA	5.714e-19 NA

The Levene test reveals a p-value smaller than 0.001, indicating that there is significant difference between the group variances in sentiment score. We conclude that the variance among the five groups is not equal.

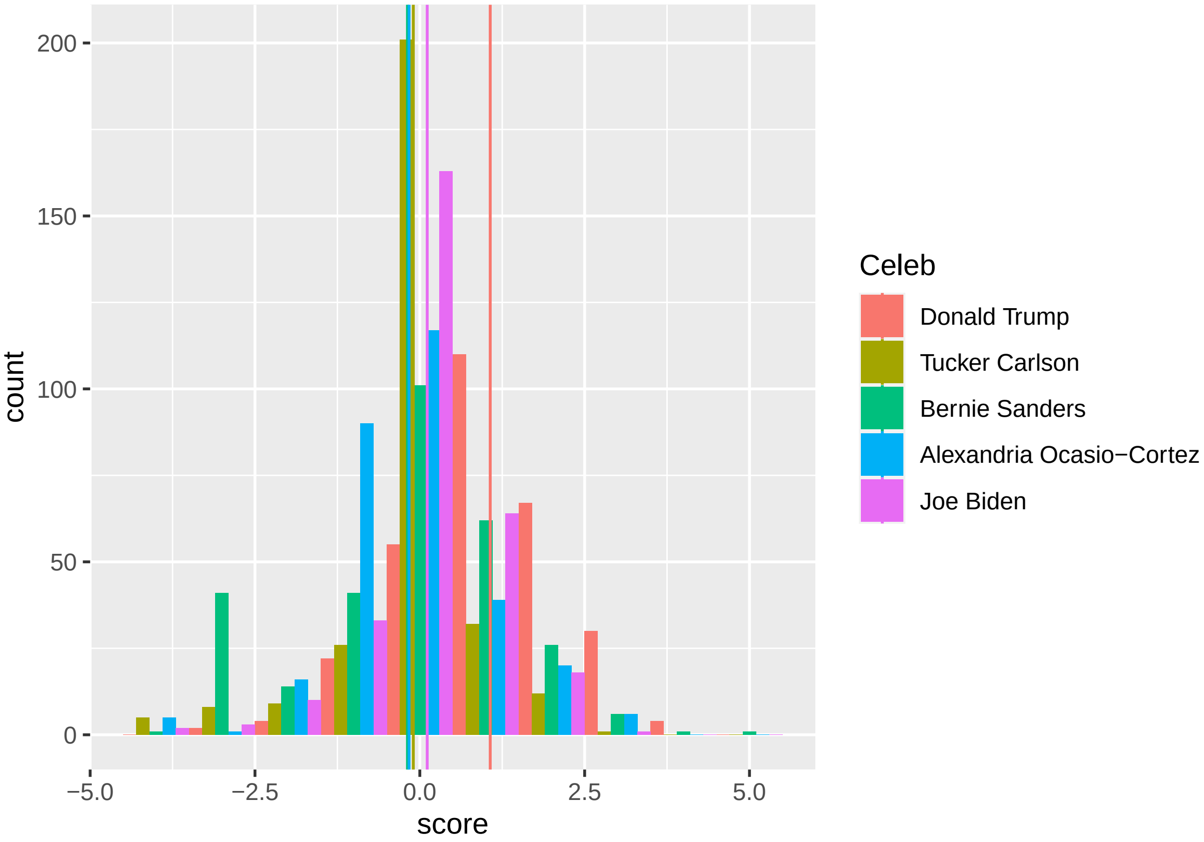
2.1.4 Visual inspection Mean and distribution sentiments

We plot both a line density and a distribution histogram which includes a mean line.

```
#boxplot(score ~ Celeb, data = semFrame, text.font = 4)
sm.density.compare(semFrame$score, semFrame$Celeb, xlab = "Score",
                   col=c('red', 'blue', 'green', 'hotpink', 'gold'), lty=c(1,1,1,1,1))
  title(main="Visual inspection Mean and distribution sentiments")
  legend('topright', legend = levels(semFrame$Celeb),
         col=c('red', 'blue', 'green', 'hotpink', 'gold'),
         title="Celebs", lty=1, cex=0.8, text.font = 4, bg='lightblue')
```



```
cdat <- ddply(semFrame, "Celeb", summarise, score.mean=mean(score))
ggplot(semFrame, aes(x=score, fill=Celeb)) +
  geom_histogram(binwidth=1, position="dodge") +
  geom_vline(data=cdat, aes(xintercept=score.mean,  colour=Celeb),
               linetype="solid", size=0.5)
```



We see all US politic Celebs have a mean around 0. #trump has the highest sentiment mean and the largest difference with the rest.

2.1.5 Frequentist approach

2.1.5.1 Linear model A one-way between subjects ANOVA was conducted to compare the effect of relation to celebrities on sentiment score in five conditions.

```r
modelTwitter0 <- lm(formula = score ~ 1 , data = semFrame)
modelTwitter1 <- lm(formula = score ~ Celeb , data = semFrame)
pander(anova(modelTwitter0, modelTwitter1, test = "F"))
```

Table 2: Analysis of Variance Table

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1469	2476	NA	NA	NA	NA
1465	2147	4	328.6	56.06	4.859e-44

There was a significant effect of relation to celebrities on sentiment score at the p<.001 level for the five conditions [F(4, 1465) = 56.06, p < 0.001].

```r
#AIC
modelsTwitter <- list(modelTwitter0, modelTwitter1)
modelTwitter.names <- c("modelTwitter0", "modelTwitter1")
pander(aictab(cand.set = modelsTwitter, modnames=modelTwitter.names),
       caption="Model selection based on AICc.")
```

Table 3: Model selection based on AICc. (continued below)

	Modnames	K	AICc	Delta_AICc	ModelLik	AICcWt	LL	Cum.Wt
2	modelTwitter1	6	4741	0	1	1	-2364	1
1	modelTwitter0	2	4942	201.3	1.939e-44	1.939e-44	-2469	1

A lower AIC indicates a better fit, which is the model with the predictor.

```
pander(pairwise.t.test(semFrame$score, semFrame$Celeb, paired = FALSE, p.adjust.method = "bonferroni"))
```

2.1.5.2 Post Hoc analysis

```
## Warning in pander.default(pairwise.t.test(semFrame$score, semFrame$Celeb, : No
## pander.method for "pairwise.htest", reverting to default.
```

- **method**: t tests with pooled SD
- **data.name**: semFrame*scoreandsemFrame*Celeb
- **p.value**:

Table 5: Table continues below

	Donald Trump	Tucker Carlson	Bernie Sanders	Alexandria Ocasio-Cortez
Tucker Carlson	3.197e-29	NA	NA	NA
Bernie Sanders	1.727e-33	1	NA	NA
Alexandria Ocasio-Cortez	2.575e-32	1	1	NA
Joe Biden	4.257e-20	0.3485	0.02765	0.05864

- **p.adjust.method**: bonferroni

Post hoc comparisons using the Bonferroni correction indicated that the corrected p-value for the trump condition was significantly different than the other conditions (p<0.001). However, between the others condition it does not show a significantly difference.

2.1.5.3 Report section for a scientific publication A one-way between subjects ANOVA was conducted to compare the effect of relation to celebrities on sentiment score in five conditions. There was a significant effect of relation to celebrities on sentiment score at the p<.001 level for the five conditions [F(4, 1465) = 56.06, p < 0.001]. However, post hoc comparisons using the Bonferroni correction indicated that only the corrected p-value for the trump condition was significantly different than the other conditions (p<0.001), between the others condition it does not show a significantly difference. Taken together, these results suggest that some celebrities really do have an effect on the sentiment in Tweets.

2.1.6 Bayesian Approach

2.1.6.1 Model description The sentiment scores seem to center around 0, and all seem to be single digits.

$$score \sim Norm(\mu, \sigma)$$

$$\mu = \alpha + b * Celeb$$

$$\alpha = Norm(0, 10)$$

$$\sigma = Uniform(0.001, 10)$$

```
mTwitter0 <-map2stan(alist(
  score ~ dnorm(mu, sigma),
  mu <-a,
  a ~ dnorm(0, 10),
  sigma ~ dunif(0.001, 10)),
  data =  semFrame ,  iter= 10000, chains = 4,   cores = 4 )
```

2.1.6.2 Model comparison

```
## Computing WAIC
```

```
mTwitter1 <-map2stan(alist(
  score ~ dnorm(mu, sigma),
  mu <-a[Celeb] ,
  a[Celeb] ~ dnorm(0, 10),
  sigma ~ dunif(0.001, 10)),
  data =  semFrame ,iter= 10000, chains = 4, cores = 4 )
```

```
## Computing WAIC
```

```
pander(compare(mTwitter0, mTwitter1, func=WAIC))
```

	WAIC	SE	dWAIC	dSE	pWAIC	weight
mTwitter1	4741	70.18	0	NA	6.698	1
mTwitter0	4942	67.72	201.2	28.94	2.593	2.01e-44

Lower WAIC indicates a better performing model, so with predictors (mTwitter1) is the winning model.

```
pander(precis(mTwitter1, depth=2, prob = .95))
```

2.1.6.3 Comparison celebrity pair

	mean	sd	2.5%	97.5%	n_eff	Rhat4
a[1]	1.068	0.07005	0.9316	1.205	32580	1
a[2]	-0.0994	0.07053	-0.2381	0.03762	31507	1
a[3]	-0.1872	0.0714	-0.3268	-0.04528	32950	0.9998

	mean	sd	2.5%	97.5%	n_eff	Rhat4
a[4]	-0.163	0.07088	-0.3021	-0.02317	32402	0.9998
a[5]	0.1124	0.07034	-0.02366	0.2498	30536	0.9999
sigma	1.212	0.0226	1.169	1.257	32906	1

Looking at the credibility intervals of the celebrities effects, We see the conditions where the mean of a condition does not fall within a credibility interval of an other condition. This holds for the a[1] (Trump) condition and a couple other combinations. We can again conclude that some celebrities really do have an effect on the sentiment in Tweets.

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

2.2.1 Conceptual model

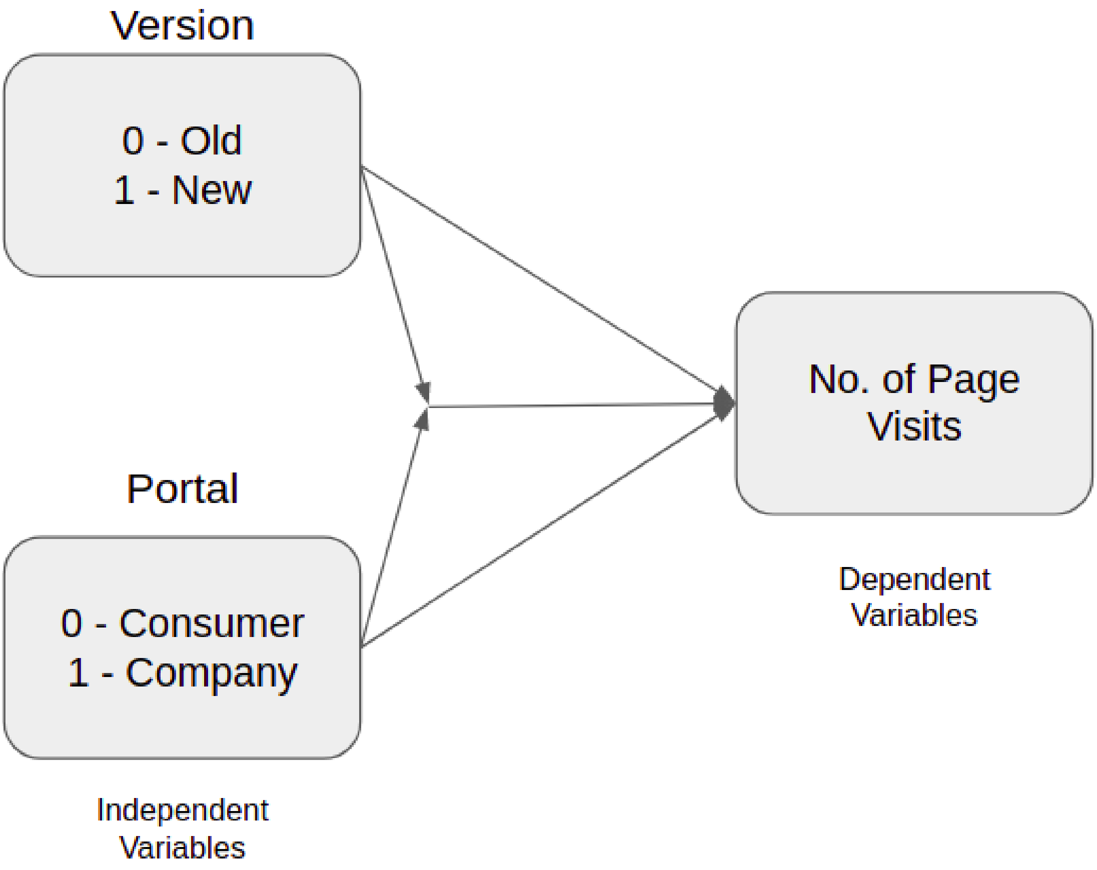
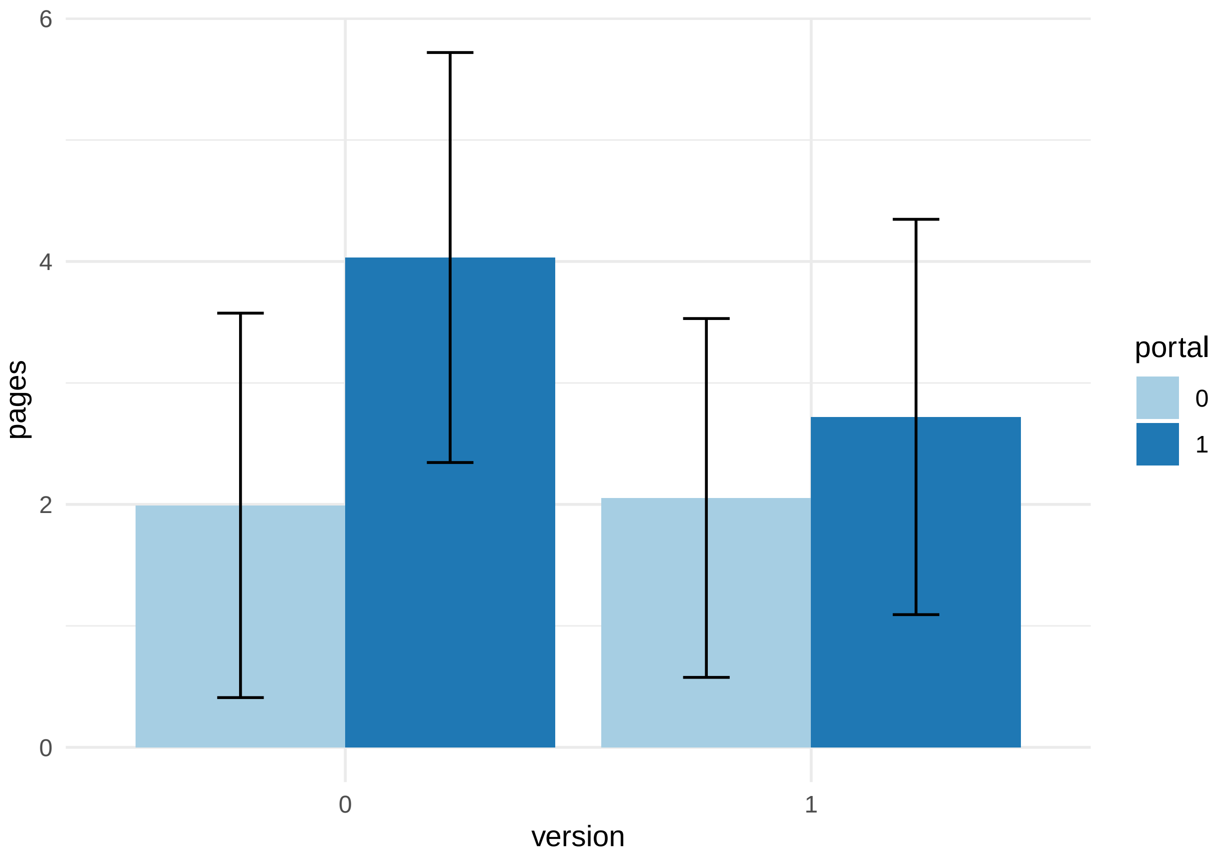


Figure 4: The conceptual model underlying the research question.

2.2.2 Visual inspection

```
filepath <- ("webvisit0.csv")
data <- read.csv(file=filepath, header=TRUE)
```

```
# changing dtype of the factors
data$portal = as.factor(data$portal)
data$version = as.factor(data$version)

# Function to calculate the mean and the standard deviation for each factor group

data_summary <- function(data, varname, groupnames){
  require(plyr)
  summary_func <- function(x, col){
    c(mean = mean(x[[col]], na.rm=TRUE),
      sd = sd(x[[col]], na.rm=TRUE))
  }
  data_sum<-ddply(data, groupnames, .fun=summary_func,
                  varname)
  data_sum <- rename(data_sum, c("mean" = varname))
 return(data_sum)
}

df3 <- data_summary(data, varname="pages",
                    groupnames=c("version", "portal"))

p <- ggplot(df3, aes(x=version, y=pages, fill=portal)) +
   geom_bar(stat="identity", position=position_dodge()) +
  geom_errorbar(aes(ymin=pages-sd, ymax=pages+sd), width=.2,
                 position=position_dodge(.9))

p + scale_fill_brewer(palette="Paired") + theme_minimal()
```

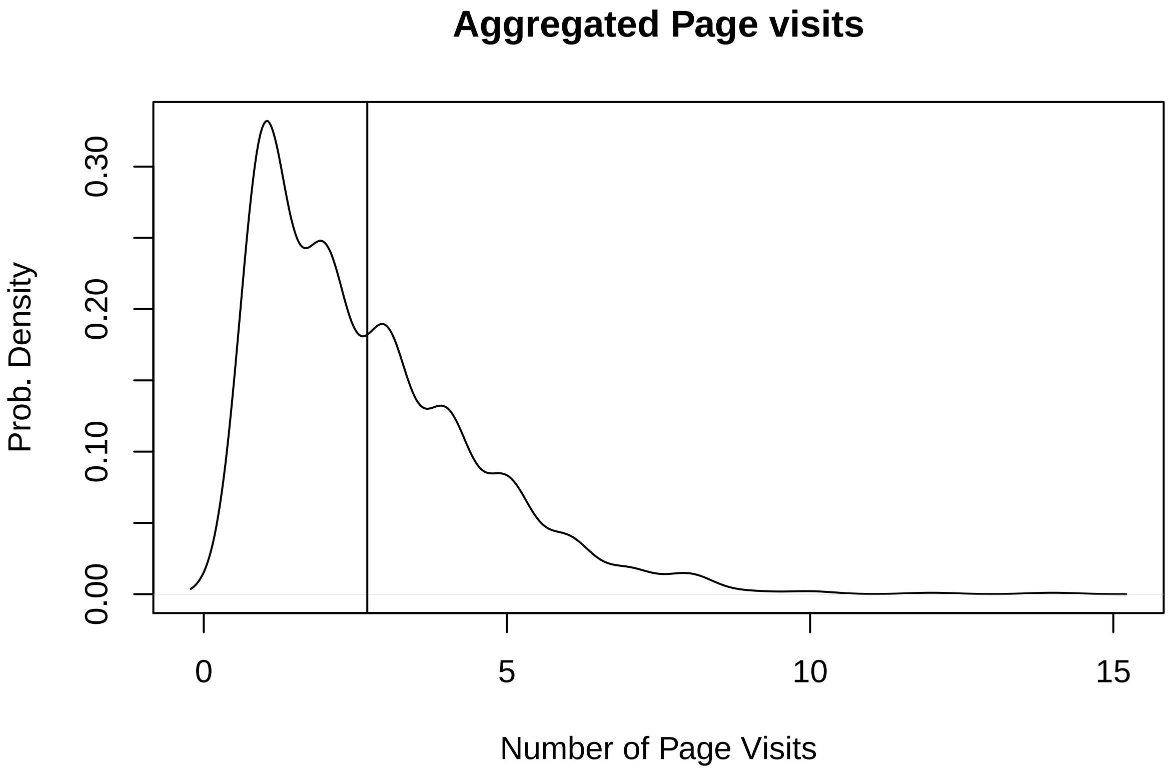


```
# Creating subsets of data for each combination of factors
subset00 <- subset(data, version == '0' & portal == '0')
subset01 <- subset(data, version == '0' & portal == '1')
subset10 <- subset(data, version == '1' & portal == '0')
subset11 <- subset(data, version == '1' & portal == '1')
```

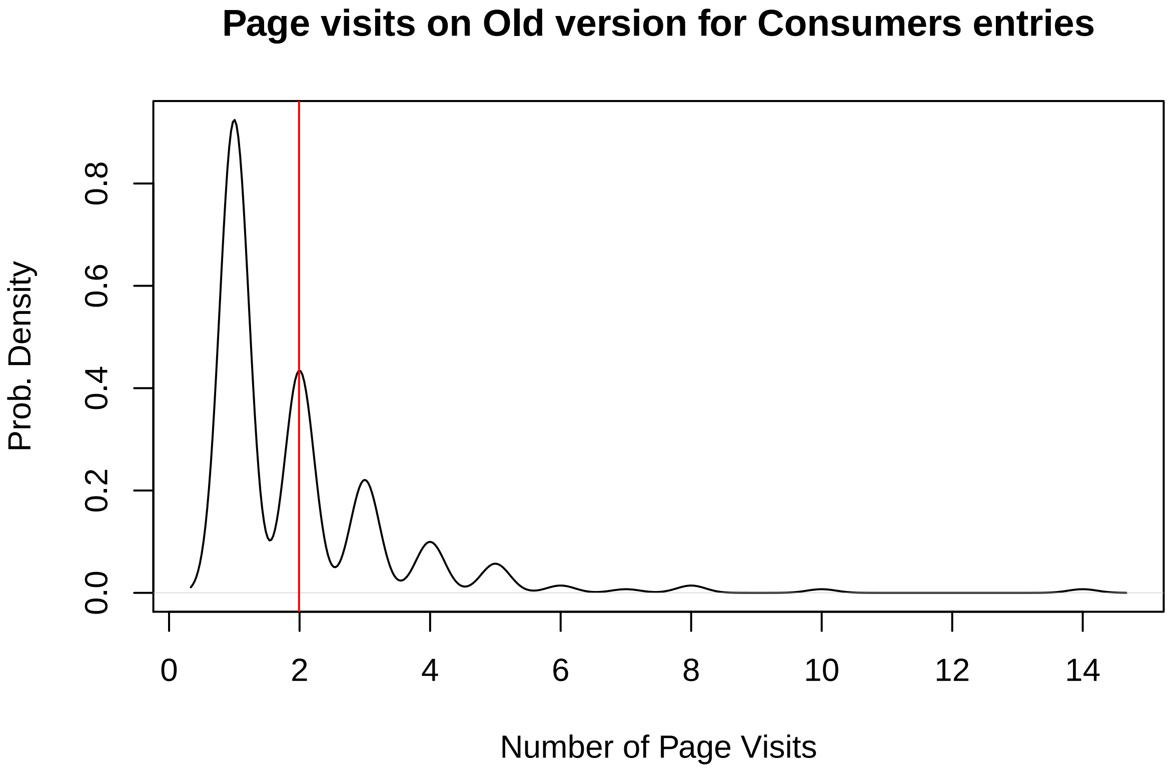
Notable observations from the bar-plot demonstrating the mean and standard deviation of the page visits are that the mean page visits across both the versions for the 0 - portal entries (Consumer) are almost the same but vary significantly for the 1 - portal entries (Company).

2.2.3 Normality check

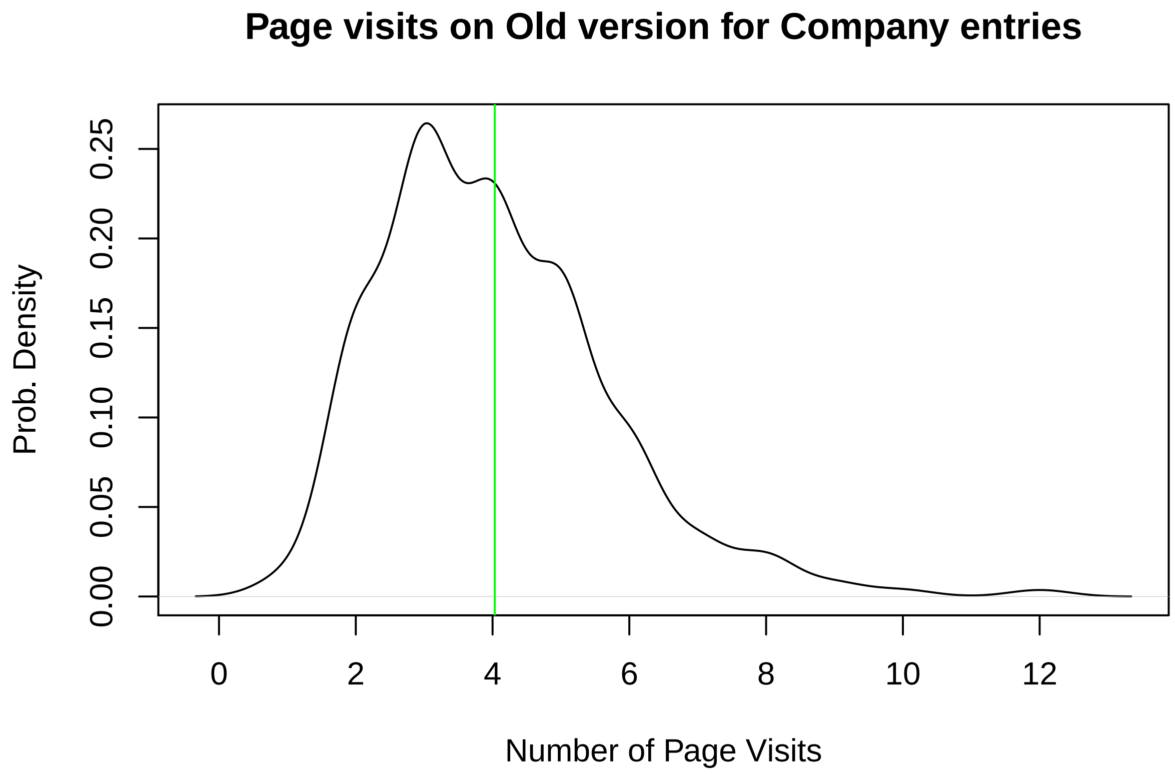
```
# Generating density plots

d <- density(data$pages)
plot(d, xlab='Number of Page Visits', ylab='Prob. Density', main='Aggregated Page visits')
abline(v = mean(data$pages), col = "black")
```



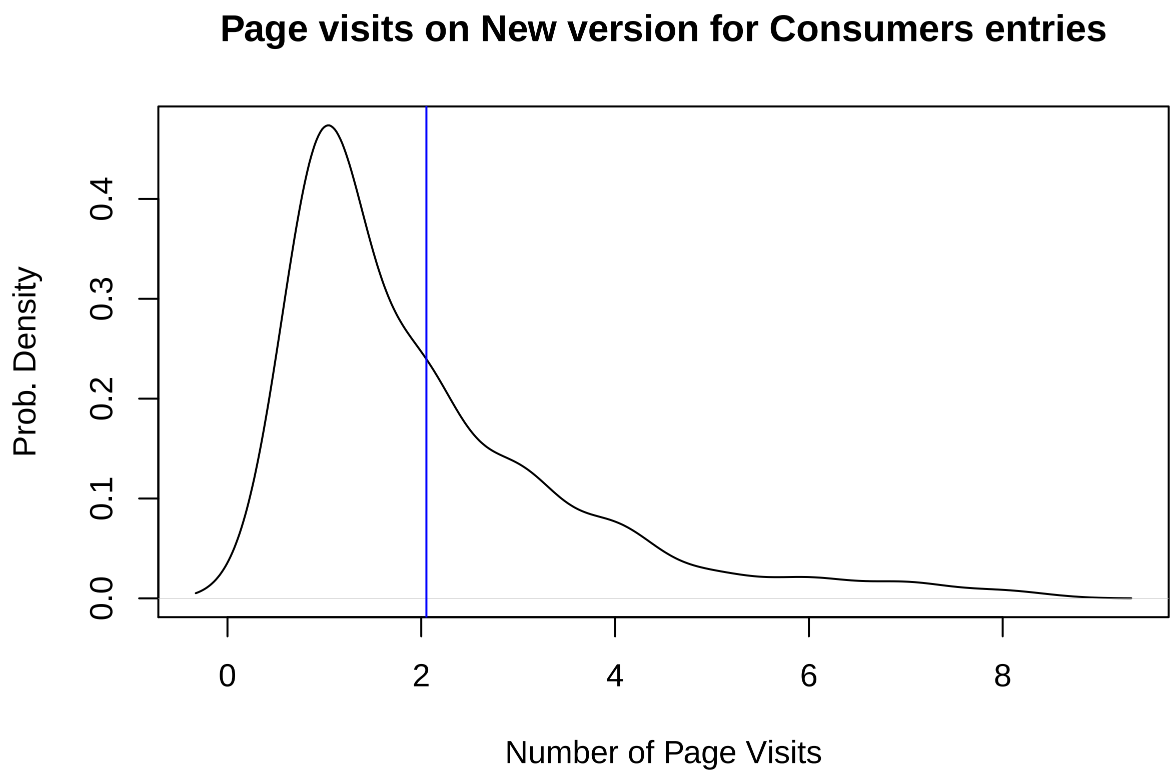
```
d <- density(subset00$pages)
plot(d, xlab='Number of Page Visits', ylab='Prob. Density', main='Page visits on Old version for Consume
abline(v = mean(subset00$pages), col = "red")
```



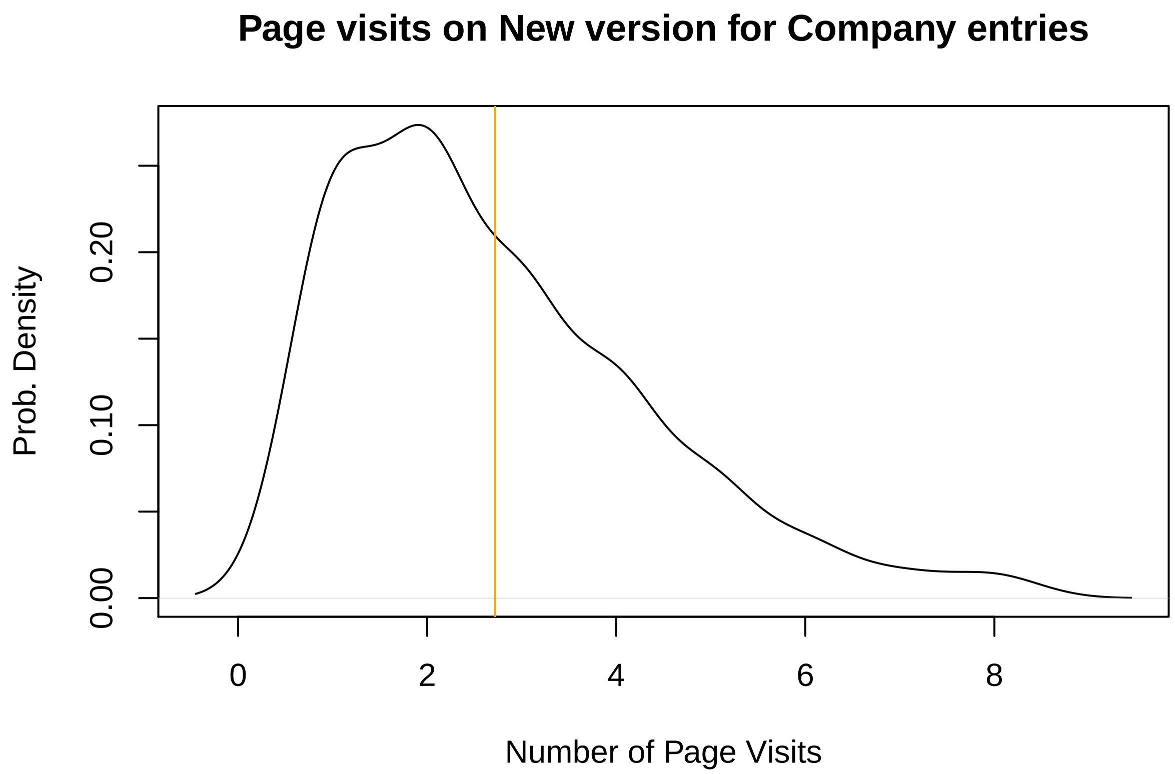
```
d <- density(subset01$pages)
plot(d, xlab='Number of Page Visits', ylab='Prob. Density', main='Page visits on Old version for Company
abline(v = mean(subset01$pages), col = "green")
```



```
d <- density(subset10$pages)
plot(d, xlab='Number of Page Visits', ylab='Prob. Density', main='Page visits on New version for Consume
abline(v = mean(subset10$pages), col = "blue")
```



```
d <- density(subset11$pages)
plot(d, xlab='Number of Page Visits', ylab='Prob. Density', main='Page visits on New version for Compan
abline(v = mean(subset11$pages), col = "orange")
```



The Density plots indicate that none of the Page visit densities resemble a Gaussian distribtuion, apart from the page visits for New version and Old version company entries. The rest have skewed distributions and the Page visits for Old version consumer entries resembles a mixture of densities. General Linear Model analysis assumes the fact that the target continuous variable has Gaussian-error distribution and thus uses appropriate log-likelihoods for the best MLE regression fit. Since the densities do not resemble Normal distributions, this might hamper the interpretability of the results.

2.2.4 Frequentist Approach

```
# Model fitting for each factor and a combination of them

modelWeb0 <- lm(pages ~ 1, data=data, na.action=na.exclude)
modelWeb1 <- lm(pages ~ version, data=data, na.action=na.exclude)
modelWeb2 <- lm(pages ~ portal, data=data, na.action=na.exclude)
modelWeb3 <- lm(pages ~ version + portal, data=data, na.action=na.exclude)
modelWeb4 <- lm(pages ~ version + portal + version:portal, data=data, na.action=na.exclude)

# ANOVA results of the effect of adding the factors

pander(anova(modelWeb0, modelWeb1), caption='Version as main effect on Page visits')
```

2.2.4.1 Model analysis

Table 9: Version as main effect on Page visits

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
998	3199	NA	NA	NA	NA
997	3107	1	92.2	29.59	6.731e-08

```
pander(anova(modelWeb0, modelWeb2), caption='Portal as main effect on Page visits')
```

Table 10: Portal as main effect on Page visits

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
998	3199	NA	NA	NA	NA
997	2751	1	448.2	162.4	1.409e-34

```
pander(anova(modelWeb3, modelWeb4), caption='Interaction effect vs 2 main effects')
```

Table 11: Interaction effect vs 2 main effects

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
996	2652	NA	NA	NA	NA
995	2534	1	117.8	46.25	1.793e-11

```
pander(anova(modelWeb4), caption='Version, Portal and interaction effect on Page visits')
```

Table 12: Version, Portal and interaction effect on Page visits

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
version	1	92.2	92.2	36.2	2.495e-09
portal	1	455.3	455.3	178.8	1.283e-37
version:portal	1	117.8	117.8	46.25	1.793e-11
Residuals	995	2534	2.547	NA	NA

```
# AICc scores of the models

modelsWeb <-list(modelWeb0, modelWeb1, modelWeb2, modelWeb3, modelWeb4)
modelWeb.names <-c("modelWeb0","modelWeb1","modelWeb2","modelWeb3","modelWeb4")
pander(aictab(cand.set = modelsWeb, modnames=modelWeb.names))
```

	Modnames	K	AICc	Delta_AICc	ModelLik	AICcWt	LL	Cum.Wt
5	modelWeb4	5	3775	0	1	1	-1882	1
4	modelWeb3	4	3818	43.37	3.824e-10	3.824e-10	-1905	1
3	modelWeb2	3	3853	78.07	1.117e-17	1.117e-17	-1923	1
2	modelWeb1	3	3975	199.6	4.451e-44	4.451e-44	-1984	1
1	modelWeb0	2	4002	226.8	5.517e-50	5.517e-50	-1999	1

The ANOVA results for the comparison of each model type indicate that the added values by including the factors individually, together and their interaction effect is statistically significant since all their p-values are <0.001. The AICc results show that model4 has the best goodness of fit since its corrected-AIC value is the least with the best log-likelihood score too.

```
data$simple <- interaction(data$version, data$portal)
contrast0 <-c(1,-1,0,0) #Only the 0-portal data
contrast1 <-c(0,0,1,-1) #Only the 1-portal data

SimpleEff <- cbind(contrast0,contrast1)
contrasts(data$simple) <- SimpleEff

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

2.2.4.2 Simple effect analysis

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	2.699	0.0505	53.45	9.614e-295
simplecontrast0	-0.03051	0.07166	-0.4258	0.6703
simplecontrast1	0.6563	0.07117	9.222	1.695e-19
simple	1.354	0.101	13.4	8.88e-38

Table 15: Fitting linear model: pages ~ simple

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

After fitting a linear model on the data, it can be observed that the company portal entries (1) have a statistically significant difference and not the consumer portal entries (0). This observation also agrees with the first plot indicating the variation in page visits for the 2 factors. The 1-portal page visits have a larger difference than the 0-portal page visits for the 0 and 1 - versions.

2.2.4.3 Report section for a scientific publication A linear model was fitted on the number of page visits by users, taking website version and web portal entires as independent variables, and including a two-way interaction between these variables. The analysis found a significant main effect (F (1, 995) = 36.2, p. < 0.01) for the version factor and (F (1, 995) = 178.8, p. < 0.01) for portal factor. The analysis also found a significant two-way interaction effect (F (1, 76) = 46.25, p. < 0.01) between these two variables. A Simple Effect analysis further examined the two-way interaction. It revealed a significant (t = 9.222, p. < 0.01) difference for the web portal entries by companies (1), but no significant effect (t = -0.4258, p. = 0.6703) was found for the web portal entries by consumers (0).

2.2.5 Bayesian Approach

2.2.5.1 Model description A gaussian model is fitted to each of the models. Model m0 is the base model with only an intercept. Model mWeb1 is an extension of model mWeb0 where the version in introduced as a predictor. Model mWeb2 is again an extension of model mWeb0 with portal as a predictor. In model mWeb3, both predictors are added as main effects, and model mWeb4 extends model mWeb3 by adding a two-way interaction effect between version and portal in the model. The priors are chosen with a normal distribution of N(0,1) for each of the model types.

The most complete model is the one which uses both the factors (Version and Portal) to determine the mean of the Gaussian distribution to model the dependent variable of Page Visits. The prior for the first variable 'a' is chosen to be a normal distribution with the mean as the mean of the page visits from the data and the uncertainty in this estimate is reflected by the standard deviation of the mean page visits as 2. The priors for the coefficients of Version and Portal are chosen to be normal distributions of mean 0 and deviation 1 since these will be anyway adjusted by the counts of the factors. The prior for the standard deviation of the of the number of page visits is chosen to be uninformed between 0.1 and 2 visits with all values within this interval having an equal chance to be

$$pages \sim Norm(\mu, \sigma)$$

$$\mu = a + b * versionN + c * portalN$$

$$alpha = Norm(0, 1)$$

$$\sigma = Uniform(0.1, 2)$$

```r
datasub <- subset(data, select = c(pages, version, portal))
datasub$versionN <- as.numeric(datasub$version)
datasub$portalN <- as.numeric(datasub$portal)
```

```
#Fitting each variant of the model

mWeb0 <-map2stan(
    alist(
        pages ~ dnorm(mu, sigma),
        mu <- a ,
        a ~ dnorm(1, 2),
        sigma ~ dunif(0.1, 2)
    ), data = datasub, iter = 10000, chains = 4, cores = 4
)
```

2.2.5.2 Model comparison

```
## Computing WAIC
```

```
mWeb1 <-map2stan(
    alist(
        pages ~ dnorm(mu, sigma),
        mu <- a + b*versionN ,
        a ~ dnorm(1, 2),
        b ~ dnorm(0, 1),
        sigma ~ dunif(0.1, 2)
    ), data = datasub, iter = 10000, chains = 4, cores = 4
)
```

```
## Computing WAIC
```

```
mWeb2 <-map2stan(
    alist(
        pages ~ dnorm(mu, sigma),
        mu <- a + b*portalN ,
        a ~ dnorm(1, 2),
        b ~ dnorm(0, 1),
        sigma ~ dunif(0.1, 2)
    ), data = datasub, iter = 10000, chains = 4, cores = 4
)
```

```
## Computing WAIC
```

```
mWeb3 <-map2stan(
    alist(
        pages ~ dnorm(mu, sigma),
        mu <- a + b*versionN + c*portalN ,
        a ~ dnorm(1, 2),
        b ~ dnorm(0, 1),
        c ~ dnorm(0, 1),
        sigma ~ dunif(0.1, 2)
    ), data = datasub, iter = 10000, chains = 4, cores = 4
)
```

```
## Computing WAIC
```

```
mWeb4 <-map2stan(
    alist(
        pages ~ dnorm(mu, sigma),
        mu <- a + b*versionN + c*portalN + d*versionN*portalN ,
        a ~ dnorm(1, 2),
        b ~ dnorm(0, 1),
        c ~ dnorm(0, 1),
        d ~ dnorm(0, 1),
        sigma ~ dunif(0.1, 2)
    ), data = datasub, iter = 10000, chains = 4, cores = 4
)
```

```
## Computing WAIC
```

```
pander(compare(mWeb0,mWeb1,mWeb2,mWeb3,mWeb4))
```

	WAIC	SE	dWAIC	dSE	pWAIC	weight
mWeb4	3780	85.86	0	NA	7.305	1
mWeb3	3821	81.89	41.03	11.08	6.338	1.232e-09
mWeb2	3855	81.97	75.53	16.95	5.17	3.977e-17
mWeb1	3976	69.23	196.2	31.36	4.279	2.513e-43
mWeb0	4003	69.91	223.5	33.11	3.426	2.889e-49

```
pander(precis(mWeb4, prob= .95))
```

	mean	sd	2.5%	97.5%	n_eff	Rhat4
a	-0.6848	0.4518	-1.581	0.186	3585	1.001
b	0.9397	0.2864	0.3915	1.5	3562	1
c	2.899	0.2869	2.349	3.47	3501	1.001
d	-1.055	0.1811	-1.412	-0.7058	3467	1
sigma	1.599	0.03563	1.531	1.671	6819	1.001

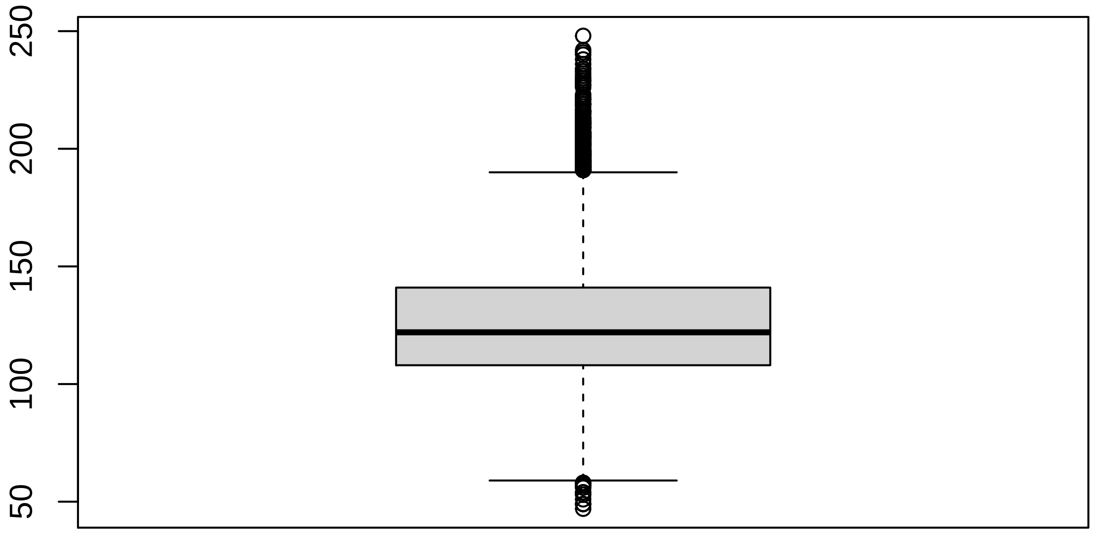
The compare() function indicates the best goodness of fit has been observed for the model m4 with the least WAIC value. For further investigation of the 95% credibility intervals, the precis() function indicates that the mean value of the coefficient of version is approximately 0, unlike for the coefficients of all the other variables (c, d for portal and two-way interaction respectively).

3 Part 3 - Multilevel model

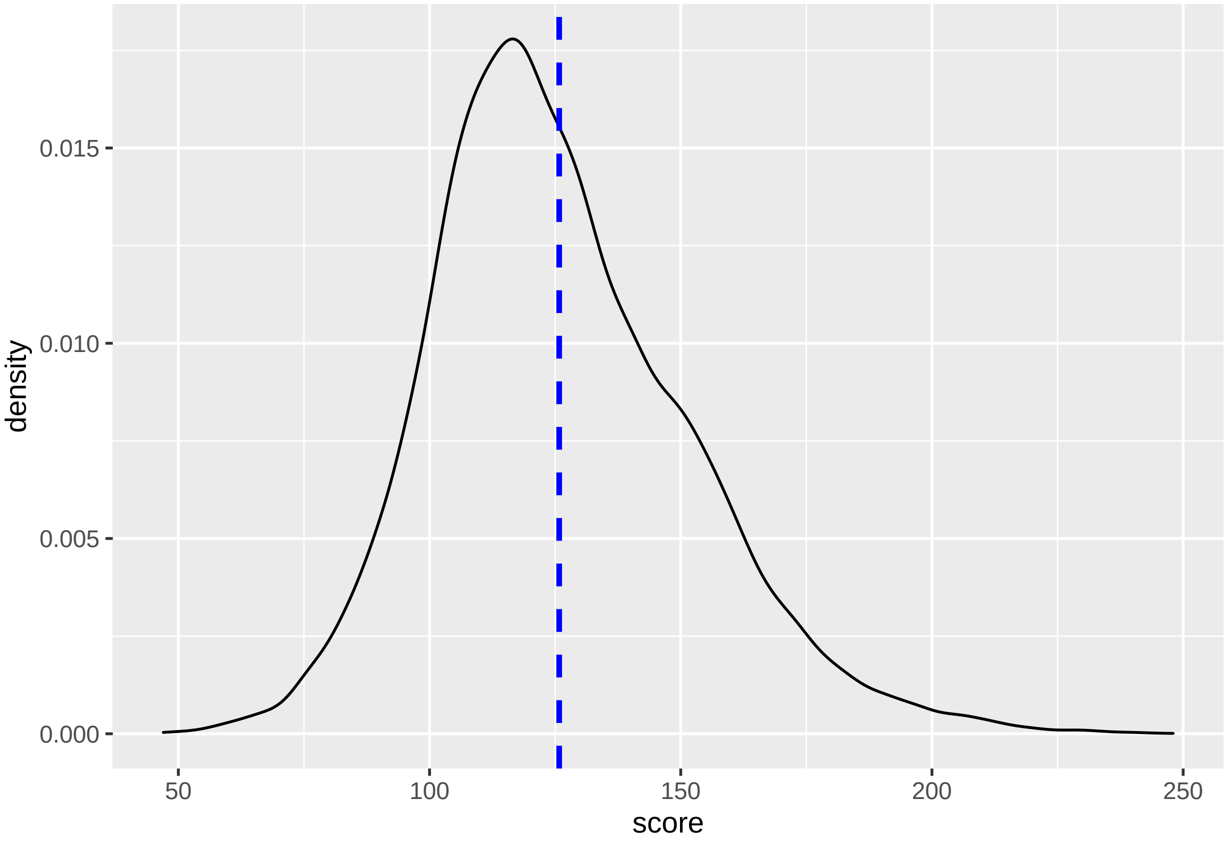
3.1 Visual inspection

The boxplot and density plot show the distribution of the score. We can see that the mean score is 122 points. The minimum is set at 59, with outliers until 46, while the maximum is set at 190, with outliers until 248.

```
# Get data
filepath <- ("set0.csv")
ds <- read.csv(file=filepath, header=TRUE)
ds <- data.frame(ds)

# boxplot score overall distribution (session independent)
boxplot(ds$score)
```



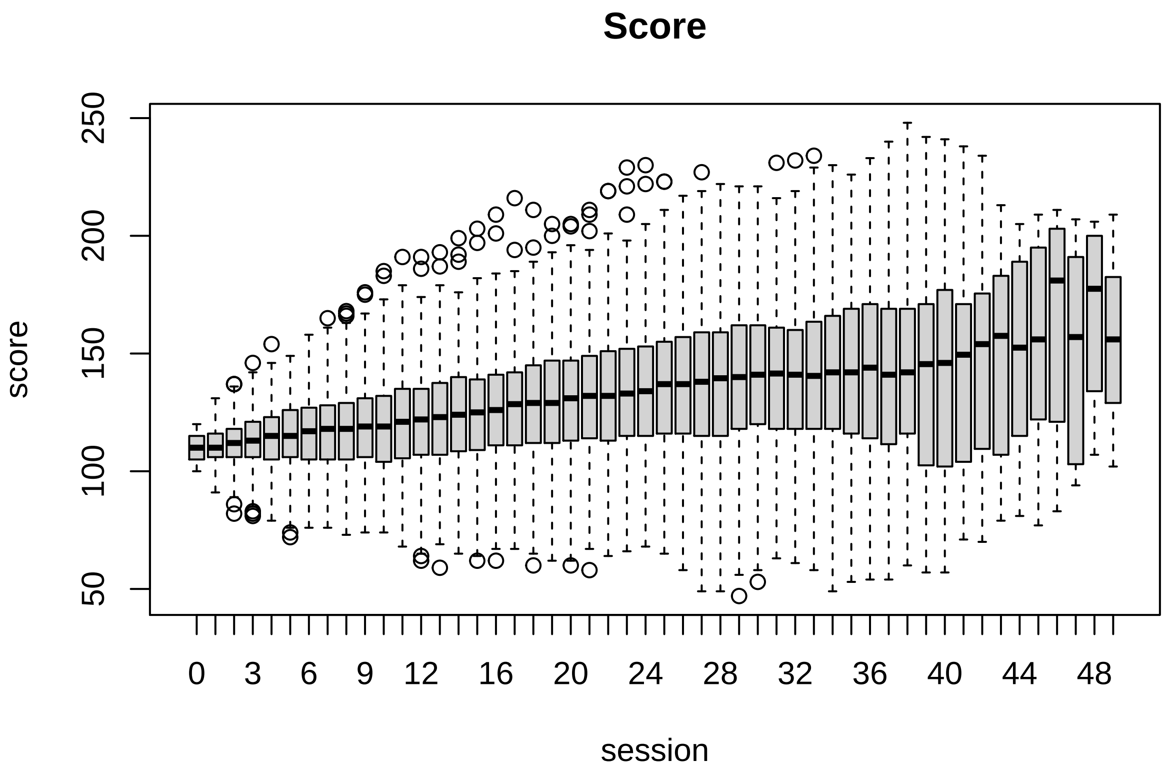
```r
# density score overall distribution (with mean line)
p <- ggplot(ds, aes(x=score)) + geom_density()
p + geom_vline(aes(xintercept=mean(score)), color="blue", linetype="dashed",size=1)
```



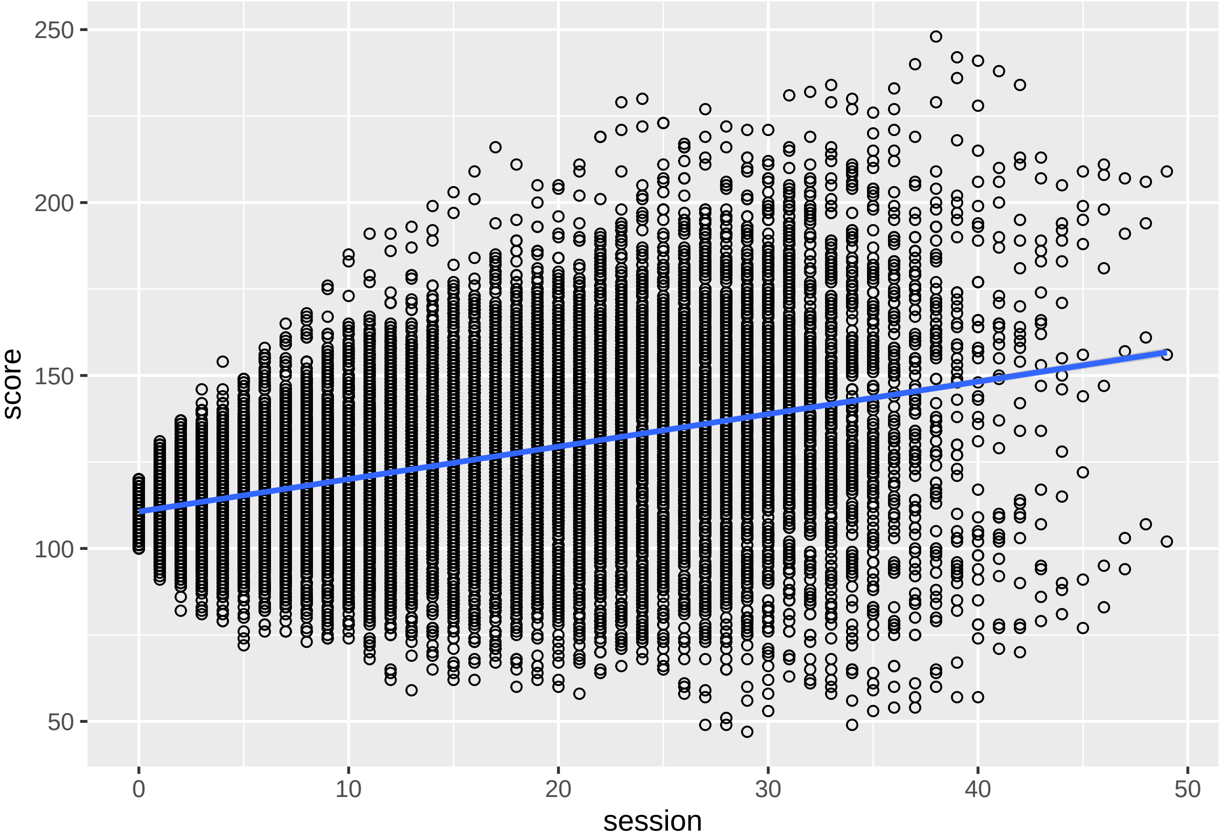
The relationship between the score and the session can be observed with the next two figures. The regression line (blue) in the scatterplot clearly shows how the score rises with the amount of sessions. This can also be observed in the box plot when looking at the mean (black line) for every box.

```r
# set labels
ds$sessionF <- factor(ds$session, levels=c(0:49), labels=c(0:49))
```

```
# boxplot score per session
boxplot(score~sessionF, data=ds, main="Score", xlab="session", ylab="score")
```



```
# ggplot score per session
hp <- ggplot(ds, aes(x=session, y=score)) + geom_point(shape=1) +
  geom_smooth(formula = y ~ x,method=lm)
hp
```



3.2 Frequentist approach

3.2.1 Multilevel analysis

We have conducted a multilevel analysis. We have an intercept only model (model0) which we compare to a model that includes a predictor parameter for the session (model1). By comparing these two models, we will know whether there is a difference in the score over the sessions.

```
# create models as given in slides lecture 4
model0 <- lm(formula=score~1, data=ds, na.action=na.exclude)
model1 <- lm(formula=score~sessionF, data=ds, na.action=na.exclude)

# analysis, see if predictor improves fitting
pander(anova(model0,model1))
```

Table 18: Analysis of Variance Table

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
16127	10713477	NA	NA	NA	NA
16078	9228641	49	1484836	52.79	0

```
pander(anova(model1))
```

Table 19: Analysis of Variance Table

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sessionF	49	1484836	30303	52.79	0
Residuals	16078	9228641	574	NA	NA

From this analysis we can see there is a significant variation between the sessions. We take a further look at the summary results.

```
pander(summary(model1))
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	110.3	1.07	103	0
sessionF1	0.7166	1.514	0.4734	0.636
sessionF2	1.908	1.514	1.261	0.2075
sessionF3	2.96	1.514	1.955	0.05054
sessionF4	3.838	1.514	2.536	0.01123
sessionF5	5.004	1.514	3.306	0.0009494
sessionF6	5.972	1.514	3.945	8.005e-05
sessionF7	7.016	1.514	4.635	3.599e-06
sessionF8	7.637	1.514	5.045	4.585e-07
sessionF9	8.551	1.514	5.649	1.642e-08
sessionF10	9.097	1.515	6.004	1.969e-09
sessionF11	10.36	1.515	6.836	8.453e-12
sessionF12	11.22	1.515	7.402	1.41e-13
sessionF13	12.44	1.515	8.209	2.407e-16
sessionF14	13.6	1.515	8.974	3.161e-19
sessionF15	14.45	1.515	9.539	1.644e-21
sessionF16	15.63	1.515	10.31	7.277e-25
sessionF17	16.69	1.516	11.01	4.212e-28
sessionF18	18.07	1.518	11.9	1.62e-32
sessionF19	19.18	1.521	12.61	2.751e-36
sessionF20	19.8	1.521	13.02	1.539e-38
sessionF21	20.85	1.525	13.67	2.489e-42
sessionF22	21.35	1.529	13.96	5.131e-44
sessionF23	22.39	1.536	14.58	7.593e-48
sessionF24	23.32	1.542	15.12	2.555e-51
sessionF25	25.06	1.555	16.11	5.923e-58
sessionF26	26.11	1.574	16.59	2.818e-61
sessionF27	26.54	1.597	16.62	1.578e-61
sessionF28	27.37	1.64	16.69	5.045e-62
sessionF29	28.76	1.68	17.12	4.195e-65
sessionF30	29.52	1.73	17.07	9.765e-65
sessionF31	30.66	1.804	16.99	3.552e-64
sessionF32	30.02	1.908	15.73	2.406e-55
sessionF33	30.19	2.034	14.85	1.568e-49
sessionF34	30.08	2.213	13.59	7.575e-42
sessionF35	30.84	2.388	12.92	5.712e-38
sessionF36	31.17	2.571	12.12	1.116e-33
sessionF37	29.23	2.825	10.35	5.185e-25
sessionF38	32.42	3.076	10.54	6.942e-26
sessionF39	31.77	3.767	8.434	3.618e-17
sessionF40	32.78	4.031	8.131	4.553e-16

	Estimate	Std. Error	t value	Pr(>|t|)
sessionF41	32.06	4.503	7.119	1.13e-12
sessionF42	34.55	5.109	6.763	1.397e-11
sessionF43	37.34	5.748	6.496	8.474e-11
sessionF44	38.8	6.492	5.976	2.33e-09
sessionF45	43.17	8.057	5.358	8.536e-08
sessionF46	50.16	9.118	5.5	3.846e-08
sessionF47	40.13	10.77	3.727	0.0001948
sessionF48	56.73	12.03	4.717	2.417e-06
sessionF49	45.39	13.87	3.272	0.00107

Table 21: Fitting linear model: score ~ sessionF

Observations	Residual Std. Error	R^2	Adjusted R^2
16128	23.96	0.1386	0.136

The summary compares the first session (intercept) with the other sessions. Looking at the estimates, we can see that compared to the first session, the scores are higher every later session. Next, we will take a look at the Akaike Information Criterion (AIC) to compare the models on the goodness-of-fit concering the out-of-sample deviance.

```
#AIC
models <- list(model0, model1)
model.names <- c("model0", "model1")
pander(aictab(cand.set = models, modnames=model.names),
       caption="Model selection based on AICc.")
```

Table 22: Model selection based on AICc.

	Modnames	K	AICc	Delta_AICc	ModelLik	AICcWt	LL	Cum.Wt
2	model1	51	148277	0	1	1	-74087	1
1	model0	2	150584	2308	0	0	-75290	1

Here we can see that model 1 has the best goodness-of-fit as it has the smallest AICc value. Lastly, we will obtain a 95% confidence interval of the estimates we have obtained earlier.

```
# gives CI95%
pander(confint(model1), caption="95% confidence interval of the estimates.")
```

Table 23: 95% confidence interval of the estimates.

	2.5 %	97.5 %
(Intercept)	108.2	112.4
sessionF1	-2.251	3.684
sessionF2	-1.059	4.875
sessionF3	-0.007004	5.927
sessionF4	0.8712	6.805
sessionF5	2.037	7.971
sessionF6	3.005	8.939
sessionF7	4.049	9.983

	2.5 %	97.5 %
sessionF8	4.67	10.6
sessionF9	5.584	11.52
sessionF10	6.127	12.07
sessionF11	7.388	13.33
sessionF12	8.245	14.19
sessionF13	9.468	15.41
sessionF14	10.63	16.57
sessionF15	11.48	17.42
sessionF16	12.66	18.6
sessionF17	13.72	19.67
sessionF18	15.09	21.04
sessionF19	16.2	22.16
sessionF20	16.82	22.79
sessionF21	17.86	23.84
sessionF22	18.35	24.34
sessionF23	19.38	25.41
sessionF24	20.3	26.34
sessionF25	22.01	28.11
sessionF26	23.02	29.19
sessionF27	23.41	29.67
sessionF28	24.15	30.58
sessionF29	25.47	32.06
sessionF30	26.13	32.91
sessionF31	27.12	34.19
sessionF32	26.28	33.76
sessionF33	26.2	34.18
sessionF34	25.74	34.42
sessionF35	26.16	35.52
sessionF36	26.13	36.21
sessionF37	23.69	34.76
sessionF38	26.39	38.45
sessionF39	24.39	39.16
sessionF40	24.88	40.68
sessionF41	23.23	40.89
sessionF42	24.54	44.57
sessionF43	26.07	48.6
sessionF44	26.07	51.52
sessionF45	27.38	58.96
sessionF46	32.28	68.03
sessionF47	19.02	61.23
sessionF48	33.15	80.3
sessionF49	18.2	72.59

Here we can again see an increase in score related to the sessions. From this we can conclude that the session has a positive effect on people's score. Also, it seems there is a significant variance between the participants in their score when the sessions increase.

3.2.2 Report section for a scientific publication

A Linear Model analysis was conducted to test the difference between sessions on the score. The results found a significant effect (F(49,16078) = 52.79, p < .001) for the sessions on the score. From the results we can conclude that over sessions the score per participant generally increases. Moreover, the variance between the participants increases when the sessions increase, which we think is caused by missing scores on later sessions or the ground truth. We cannot make any conclusions on this until the missing data is available.

3.3 Bayesian approach

3.3.1 Model description

For model 2, the model with session as a factor, we take as prior a normal distribution of N(125,30). This comes from the mean of the score, 125, and a bit more than the standard deviation, which is around 27. Our sigma is set at a uniform distribution of U(0.001,30).

$$score \sim Norm(\mu, \sigma)$$

$$\mu = a + b * sessionF + c * subjF$$

$$alpha = Norm(125, 30)$$

$$\sigma = Uniform(0.001, 50)$$

3.3.2 Model comparison

```
ds <- ds[!(ds$Subject>99),] # select first 100 subjects
ds$Subject <- ds$Subject +1 # increase subject number with 1 to overcome Stan zero index problem

mean(ds$score) # check mean
```

```
## [1] 125.5142
```

```
sd(ds$score) # check standard deviation
```

```
## [1] 27.402
```

```
ds$sessionF <- factor(ds$session, levels=c(0:49), labels=c(0:49))
ds$subjF <- factor(ds$Subject, levels=c(1:100), labels=c(1:100))

da <- subset(ds, select=c(score, sessionF))
da1 <- subset(ds, select=c(score, sessionF, subjF))

# create model with fixed intercept (i)
m0 <- map2stan(
  alist(
    score ~ dnorm(mu, sigma),
    mu <- a,
    a ~dnorm(125,30), # mean and sd from what we found above
    sigma ~dunif(0.001,50)
  ), data = da, iter = 10000, chains = 4, cores = 4
)
```

```
## Computing WAIC
```

```
# create model extended with an adaptive prior for subject id (ii)
m1 <- map2stan(
  alist(
    score ~ dnorm(mu, sigma),
    mu <- a + a_subj[subjF],
    a_subj[subjF] ~ dnorm(0,sigma_subj),
    sigma_subj ~ dcauchy(0,10),
    a ~ dnorm(125,30),
    sigma ~ dcauchy(0.001,50)
  ), data = da1, iter = 10000, chains = 4, cores = 4
)
```

```
## Computing WAIC
```

```
# create model with session as a factor (iii)
m2 <- map2stan(
  alist(
    score ~ dnorm(mu, sigma),
    mu <- a[sessionF],
    a[sessionF] ~ dnorm(125,30),
    sigma ~dunif(0.001,50)
  ), data = da, iter = 10000, chains = 4, cores = 4
)
```

```
## Computing WAIC
```

```
pander(compare(m0,m1,m2,func=WAIC))
```

	WAIC	SE	dWAIC	dSE	pWAIC	weight
m1	28321	96.43	0	NA	92.86	1
m2	30578	98.54	2256	120.9	67.43	0
m0	31039	98.24	2717	104.5	2.421	0

We have created and compared the three described models. From the results we can see that model 1, the model with the adaptive prior for subject id, has the best fit since it has the smallest WAIC value and largest Akaike weight.

3.3.3 Estimates examination

From the previous comparison we could see that model 1 is the best fit model. We will further examine this model with 95% credible intervals of the parameters of this model.

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

	mean	sd	2.5%	97.5%	n_eff	Rhat4
a_subj[1]	-6.696	5.746	-18.01	4.503	7252	1
a_subj[2]	-19.59	3.606	-26.73	-12.47	2675	1.001
a_subj[3]	44.36	3.682	37.03	51.51	3141	1.002
a_subj[4]	14.11	3.556	7.026	21	2769	1.001
a_subj[5]	-5.959	3.637	-13.17	1.106	2727	1.002
a_subj[6]	-10.73	3.799	-18.38	-3.439	2928	1.002
a_subj[7]	-4.632	3.726	-11.9	2.566	3285	1.001
a_subj[8]	9.319	3.645	2.248	16.43	2857	1.001
a_subj[9]	-2.608	4.522	-11.54	6.202	4321	1.001
a_subj[10]	20.58	3.713	13.23	27.88	3010	1.002
a_subj[11]	13.13	3.321	6.559	19.59	2416	1.002
a_subj[12]	-19.77	3.726	-27.08	-12.57	3082	1.002

	mean	sd	2.5%	97.5%	n_eff	Rhat4
a_subj[13]	9.445	3.524	2.585	16.44	2732	1.001
a_subj[14]	10.19	3.369	3.501	16.7	2434	1.002
a_subj[15]	-25.71	3.461	-32.44	-18.91	2571	1.002
a_subj[16]	26.5	3.583	19.48	33.45	2653	1.002
a_subj[17]	2.346	3.441	-4.345	9.039	2621	1.002
a_subj[18]	-42.49	3.911	-50.15	-34.85	3209	1.002
a_subj[19]	-1.237	3.88	-8.766	6.405	3238	1.001
a_subj[20]	-2.549	3.847	-10.15	4.977	3234	1.002
a_subj[21]	-13.35	3.518	-20.3	-6.489	2705	1.002
a_subj[22]	10.81	4.375	2.163	19.29	4158	1.001
a_subj[23]	-6.649	3.513	-13.61	0.1163	2623	1.002
a_subj[24]	-6.301	4.129	-14.43	1.749	3450	1.001
a_subj[25]	-15.62	3.802	-23.06	-8.137	3087	1.002
a_subj[26]	-7.67	3.363	-14.31	-1.179	2448	1.002
a_subj[27]	2.155	3.475	-4.731	8.901	2608	1.002
a_subj[28]	7.797	3.734	0.4367	15.1	3064	1.002
a_subj[29]	-30.16	3.625	-37.24	-23.11	2826	1.002
a_subj[30]	23.72	3.748	16.42	31.09	2945	1.002
a_subj[31]	23.41	3.963	15.6	31.02	3354	1.001
a_subj[32]	-43.26	3.619	-50.43	-36.2	2780	1.001
a_subj[33]	34.17	3.454	27.36	40.85	2509	1.002
a_subj[34]	-29.1	3.821	-36.66	-21.66	2990	1.002
a_subj[35]	4.815	3.821	-2.728	12.21	2938	1.002
a_subj[36]	-32.23	3.317	-38.76	-25.82	2373	1.002
a_subj[37]	-4.897	3.667	-12.1	2.21	2995	1.002
a_subj[38]	20.22	3.468	13.38	26.98	2549	1.002
a_subj[39]	-12.65	3.852	-20.41	-5.199	3258	1.001
a_subj[40]	-10.88	3.836	-18.39	-3.299	3088	1.002
a_subj[41]	-24.67	3.787	-32.1	-17.23	2895	1.002
a_subj[42]	14.05	3.86	6.424	21.63	3348	1.002
a_subj[43]	-2.73	3.73	-10.02	4.537	2949	1.002
a_subj[44]	-29.5	3.769	-36.94	-22.18	3002	1.002
a_subj[45]	-17.07	3.639	-24.28	-10.08	2607	1.002
a_subj[46]	1.34	3.442	-5.471	8.011	2348	1.002
a_subj[47]	9.101	3.62	1.918	16.17	2808	1.002
a_subj[48]	-10.76	3.632	-17.9	-3.712	2556	1.002
a_subj[49]	38.65	3.525	31.61	45.56	2493	1.002
a_subj[50]	19.25	3.544	12.28	26.13	2631	1.002
a_subj[51]	1.18	3.688	-6.044	8.349	2829	1.002
a_subj[52]	-8.129	3.437	-14.87	-1.34	2496	1.002
a_subj[53]	16.97	3.723	9.587	24.23	2877	1.002
a_subj[54]	7.856	3.657	0.6465	14.99	2839	1.002
a_subj[55]	-13.61	3.945	-21.33	-5.994	3208	1.002
a_subj[56]	-29.05	3.995	-36.81	-21.26	3615	1.001
a_subj[57]	10.47	3.672	3.223	17.58	2863	1.001
a_subj[58]	14.11	3.752	6.74	21.39	3179	1.002
a_subj[59]	-25.01	4.154	-33.16	-16.97	3634	1.001
a_subj[60]	21.25	3.507	14.3	28.13	2708	1.001
a_subj[61]	-18	3.55	-24.95	-11.2	2947	1.001
a_subj[62]	72.77	3.351	66.19	79.24	2361	1.002
a_subj[63]	-34.21	3.754	-41.69	-26.84	3117	1.002
a_subj[64]	-19.68	3.855	-27.22	-12.2	2972	1.001

	mean	sd	2.5%	97.5%	n_eff	Rhat4
a_subj[65]	-27.82	3.55	-34.83	-20.86	2744	1.002
a_subj[66]	-20.44	3.464	-27.22	-13.61	2768	1.001
a_subj[67]	17.03	3.844	9.43	24.59	3081	1.002
a_subj[68]	-15.69	3.513	-22.58	-8.876	2790	1.002
a_subj[69]	13.26	3.728	5.895	20.58	2627	1.003
a_subj[70]	33.54	3.607	26.46	40.61	2748	1.002
a_subj[71]	-27.86	3.701	-35.12	-20.66	2910	1.002
a_subj[72]	-16.58	3.681	-23.83	-9.384	2951	1.001
a_subj[73]	5.878	3.833	-1.72	13.34	2796	1.002
a_subj[74]	13.26	3.562	6.199	20.14	2624	1.002
a_subj[75]	-17.94	3.315	-24.49	-11.46	2261	1.002
a_subj[76]	12.83	3.591	5.632	19.77	2651	1.002
a_subj[77]	29.6	3.45	22.9	36.41	2554	1.002
a_subj[78]	-5.608	3.628	-12.7	1.424	2950	1.001
a_subj[79]	-16.03	3.766	-23.56	-8.595	2897	1.001
a_subj[80]	-3.644	3.514	-10.58	3.269	2787	1.002
a_subj[81]	6.803	3.646	-0.3159	13.91	2764	1.002
a_subj[82]	-9.806	3.558	-16.82	-2.791	2622	1.002
a_subj[83]	-4.049	3.863	-11.7	3.494	2997	1.002
a_subj[84]	-33.48	3.612	-40.65	-26.41	2932	1.001
a_subj[85]	-7.919	3.709	-15.23	-0.6285	3047	1.001
a_subj[86]	20.32	3.382	13.66	26.9	2393	1.002
a_subj[87]	7.409	3.766	-0.04578	14.7	3159	1.001
a_subj[88]	21.17	3.915	13.52	28.9	3260	1.001
a_subj[89]	-1.338	3.442	-8.104	5.388	2517	1.002
a_subj[90]	19.07	3.692	11.86	26.32	2963	1.002
a_subj[91]	-0.8418	3.594	-7.926	6.113	2668	1.002
a_subj[92]	15.86	3.738	8.523	23.13	3090	1.001
a_subj[93]	3.358	3.584	-3.652	10.34	2705	1.002
a_subj[94]	2.896	3.52	-4.101	9.785	2667	1.002
a_subj[95]	-8.43	4.191	-16.72	-0.1597	3824	1.001
a_subj[96]	1.071	3.859	-6.516	8.556	3091	1.002
a_subj[97]	29.2	3.862	21.54	36.76	3114	1.002
a_subj[98]	24.21	3.455	17.51	31.02	2503	1.002
a_subj[99]	0.2123	3.52	-6.63	7.047	2791	1.002
a_subj[100]	14.57	3.501	7.767	21.42	2620	1.002
sigma_subj	20.49	1.48	17.83	23.67	28458	0.9999
a	125	2.018	121.1	129.2	869.2	1.006
sigma	17.86	0.2252	17.43	18.31	36654	0.9999

We can observe that the mean between the subjects has a high variance. This means that although the scores increase per session, the subjects have very different prior skills, achieving relatively higher or lower scores in their first session than average.