Report coursework assignment A - 2021

CS4125 Seminar Research Methodology for Data Science

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

20/04/2021

Contents

```
library(foreign)
library(ggplot2)
library(plyr)
library(pander)
library(sm)
```

```
## Package 'sm', version 2.2-5.6: type help(sm) for summary information
```

```
library(AICcmodavg)
library(rethinking)
```

```
## Loading required package: rstan

## Loading required package: StanHeaders

## rstan (Version 2.21.2, GitRev: 2e1f913d3ca3)

## For execution on a local, multicore CPU with excess RAM we recommend calling
## options(mc.cores = parallel::detectCores()).
## To avoid recompilation of unchanged Stan programs, we recommend calling
## rstan_options(auto_write = TRUE)

## Do not specify '-march=native' in 'LOCAL_CPPFLAGS' or a Makevars file

## Loading required package: parallel
## rethinking (Version 2.13)

##
## Attaching package: 'rethinking'

## The following object is masked from 'package:AICcmodavg':
##
##     DIC

## The following object is masked from 'package:stats':
##
##     rstudent
```

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

1.1 The motivation for the planned research

(Max 250 words)

1.2 The theory underlying the research

(Max 250 words) Preferable based on theories reported in literature

1.3 Research questions

The research question that will be examined in the experiment (or alternatively the hypothesis that will be tested in the experiment)

1.4 The related conceptual model

This model should include: *Independent variable(s)* Dependent variable *Mediating variable (at least 1)* Moderating variable (at least 1)

1.5 Experimental Design

Note that the study should have a true experimental design

1.6 Experimental procedure

Describe how the experiment will be executed step by step

1.7 Measures

Describe the measure that will be used

1.8 Participants

Describe which participants will recruit in the study and how they will be recruited

1.9 Suggested statistical analyses

Describe the statistical test you suggest to care out on the collected data

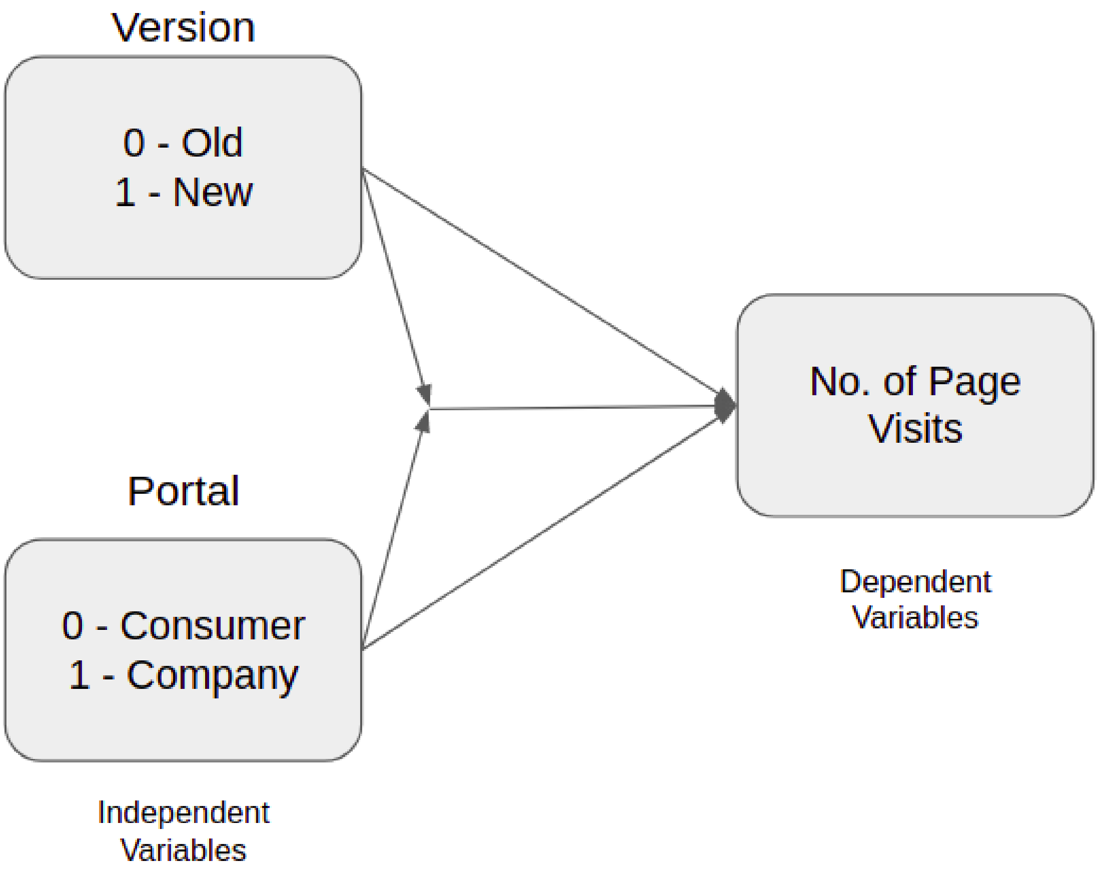


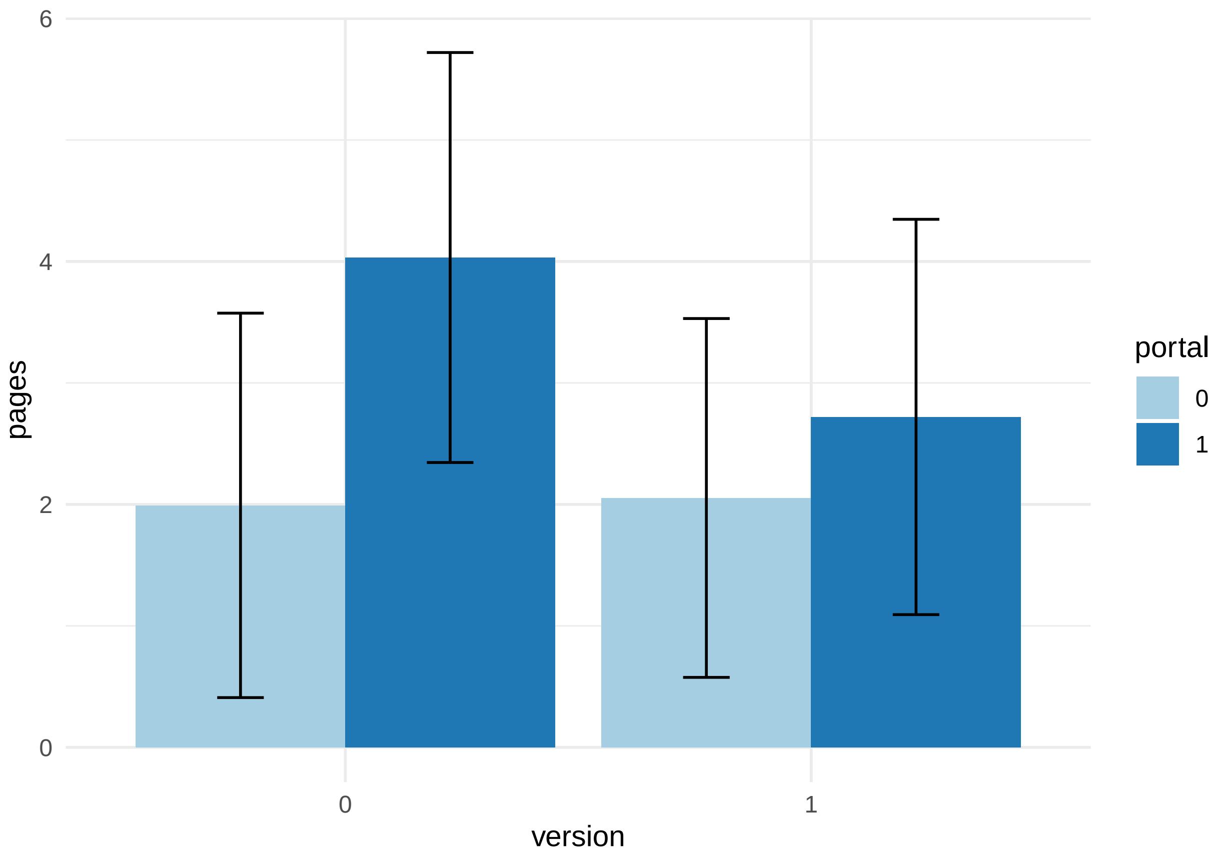
Figure 1: The conceptual model underlying the research question.

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

1.10.1 Conceptual model

1.10.2 Visual inspection

```r
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()
```

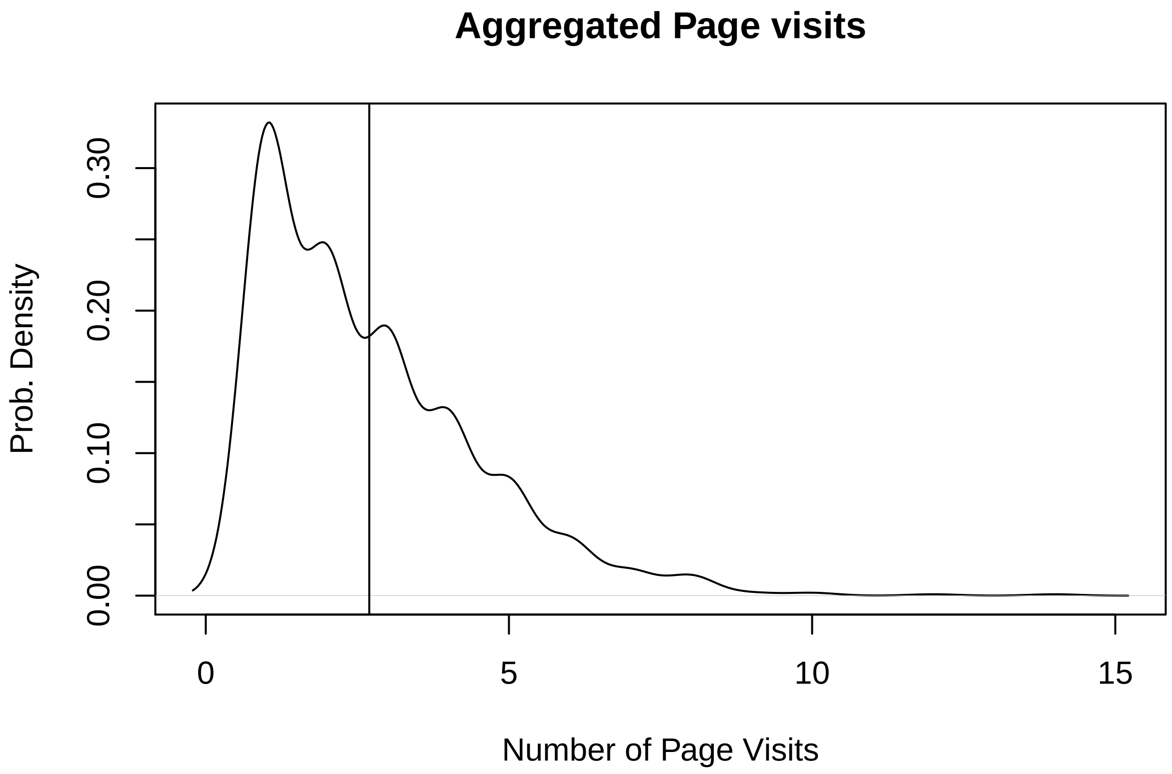


```
# Creating subsets of data for each combination of factors
subset00 <- subset(data, version == '0' & portal == '0')
subset01 <- subset(data, version == '0' & portal == '1')
subset10 <- subset(data, version == '1' & portal == '0')
subset11 <- subset(data, version == '1' & portal == '1')
```

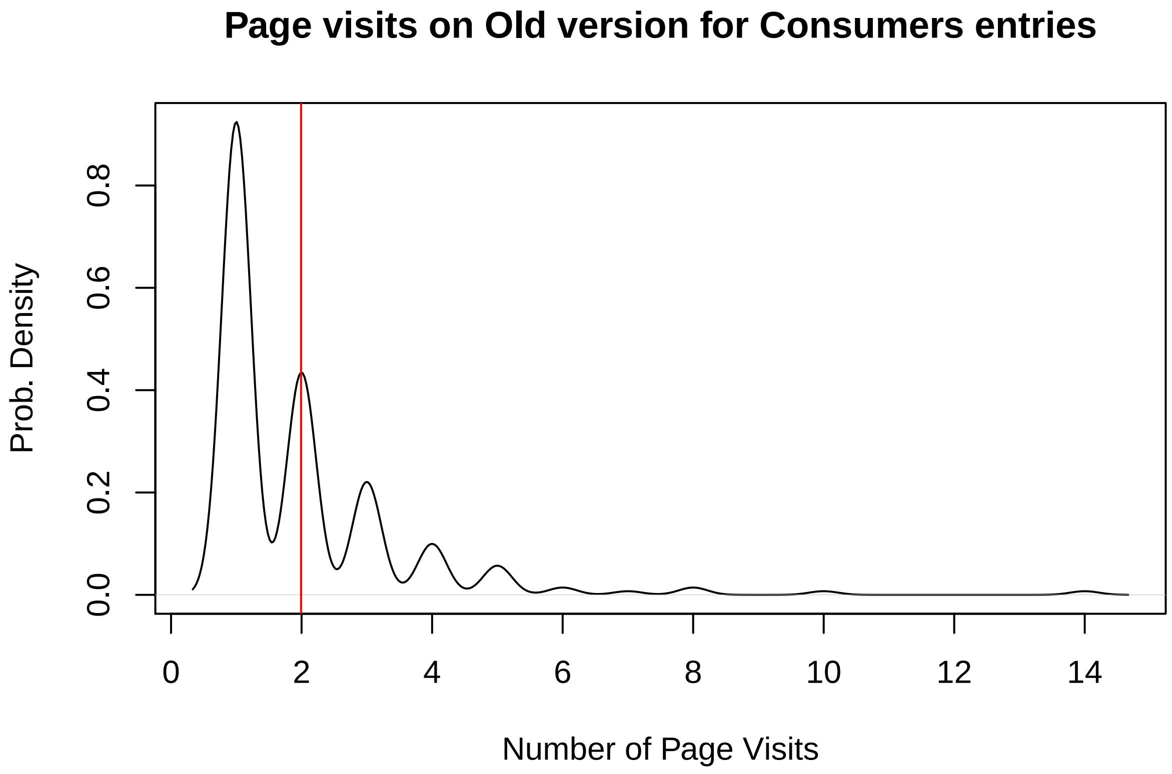
Notable observations from the bar-plot demonstrating the mean and standard deviation of the page visits are that the mean page visits across both the versions for the 0 - portal entries (Consumer) are almost the same but vary significantly for the 1 - portal entries (Company).

1.10.3 Normality check

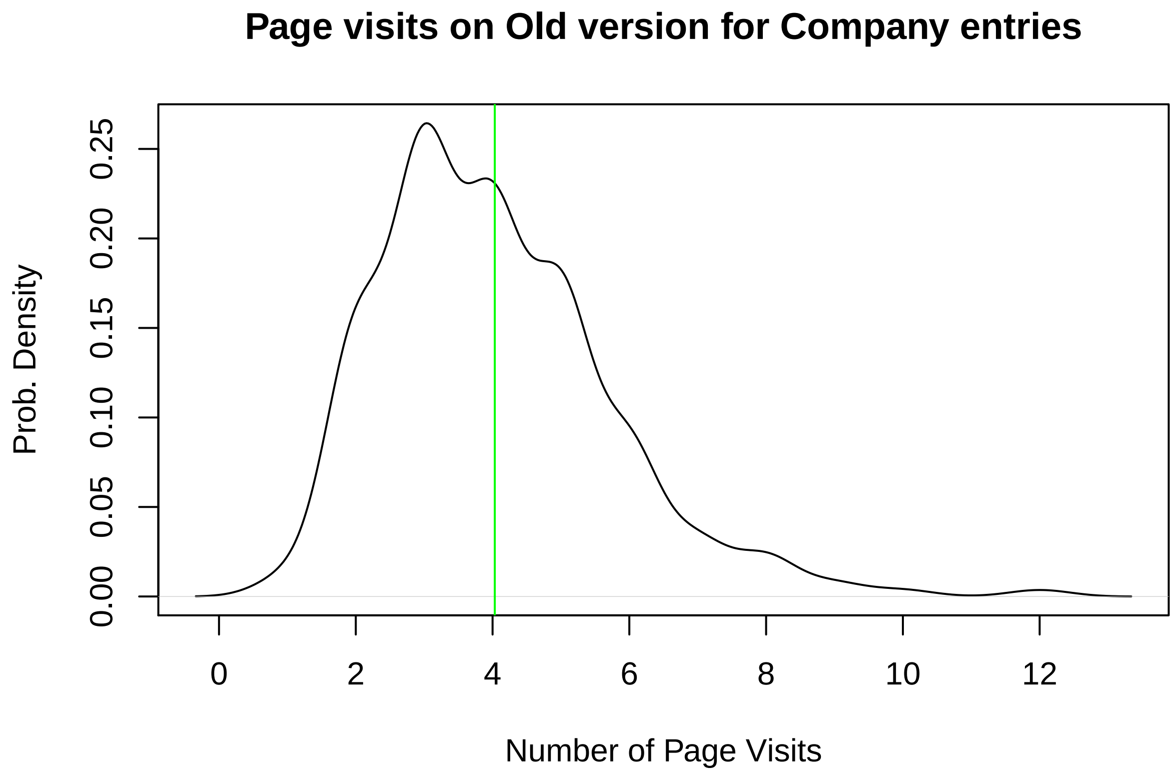
```
# Generating density plots

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")
```



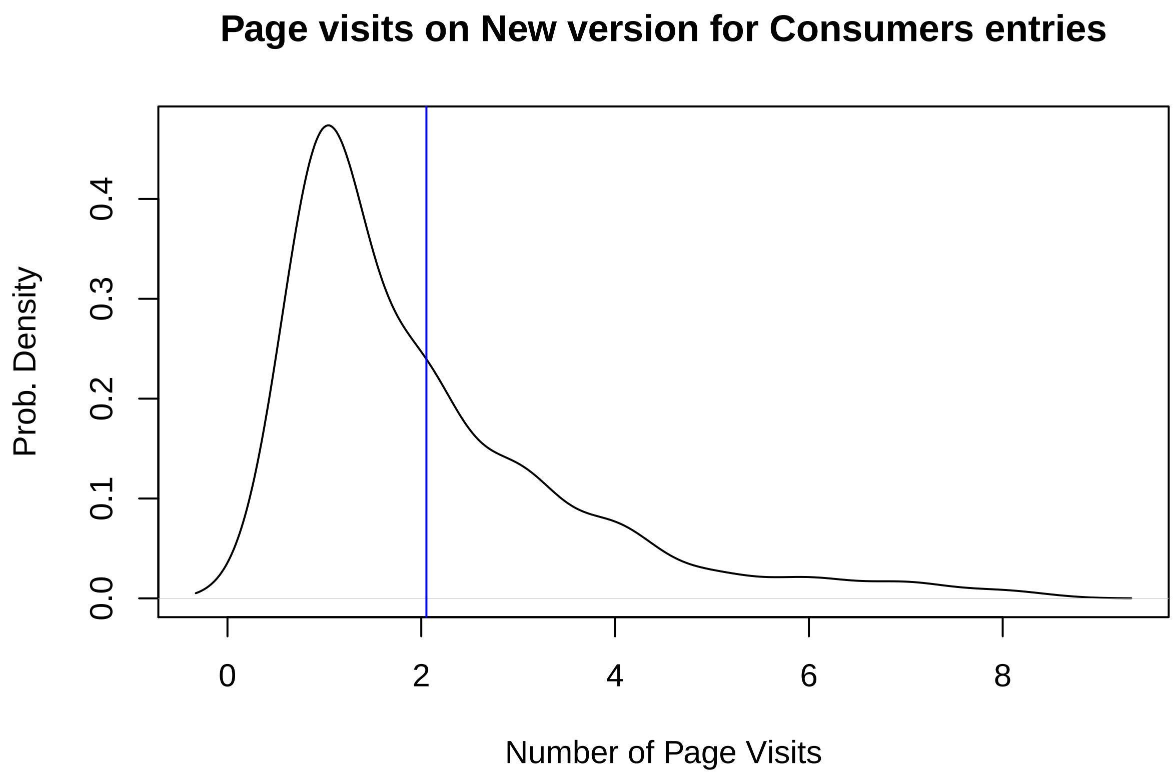
```
d <- density(subset00$pages)
plot(d, xlab='Number of Page Visits', ylab='Prob. Density', main='Page visits on Old version for Consume
abline(v = mean(subset00$pages), col = "red")
```



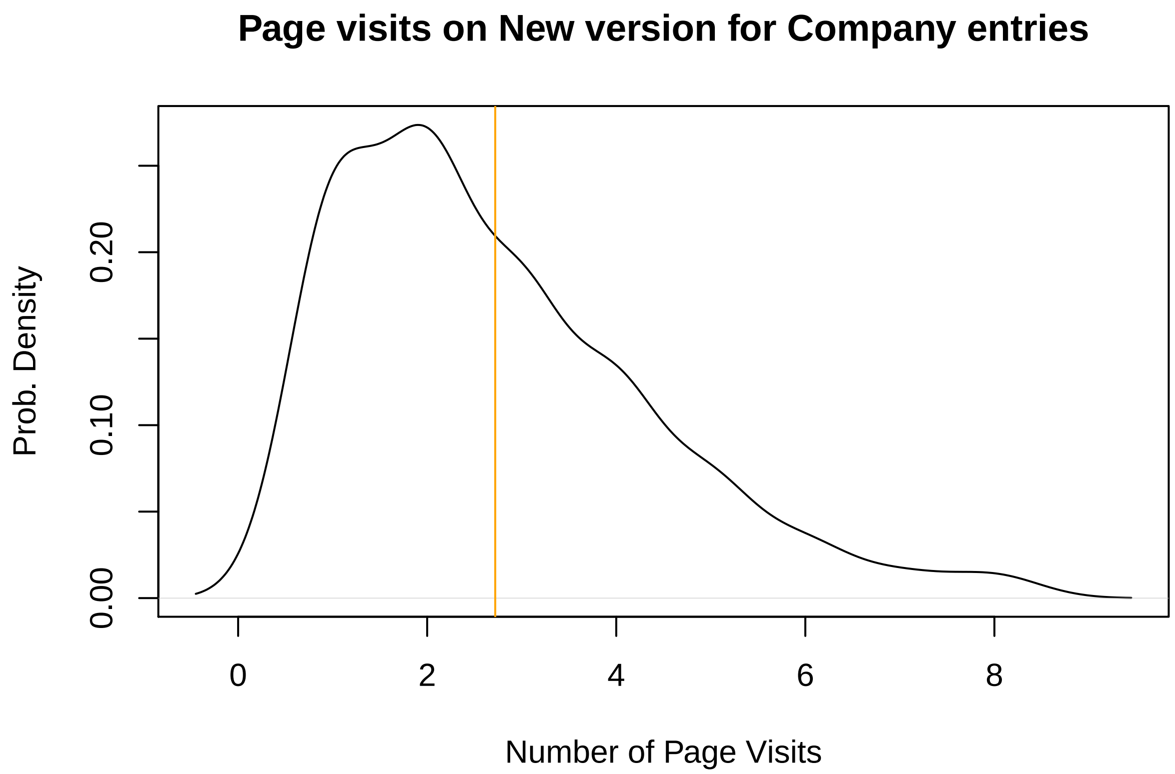
```
d <- density(subset01$pages)
plot(d, xlab='Number of Page Visits', ylab='Prob. Density', main='Page visits on Old version for Compan
abline(v = mean(subset01$pages), col = "green")
```



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

1.10.4 Frequentist Approach

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

model0 <- lm(pages ~ 1, data=data, na.action=na.exclude)
model1 <- lm(pages ~ version, data=data, na.action=na.exclude)
model2 <- lm(pages ~ portal, data=data, na.action=na.exclude)
model3 <- lm(pages ~ version + portal, data=data, na.action=na.exclude)
model4 <- lm(pages ~ version + portal + version:portal, data=data, na.action=na.exclude)

# ANOVA results of the effect of adding the factors

pander(anova(model0, model1), caption='Version as main effect on Page visits')
```

1.10.4.1 Model analysis

Table 1: 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(model0, model2), caption='Portal as main effect on Page visits')
```

Table 2: 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(model3, model4), caption='Interaction effect vs 2 main effects')
```

Table 3: 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(model4), caption='Version, Portal and interaction effect on Page visits')
```

Table 4: 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

```r
# AICc scores of the models

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	5	3775	0	1	1	-1882	1
4	model3	4	3818	43.37	3.824e-10	3.824e-10	-1905	1
3	model2	3	3853	78.07	1.117e-17	1.117e-17	-1923	1
2	model1	3	3975	199.6	4.451e-44	4.451e-44	-1984	1
1	model0	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.

```r
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))
```

1.10.4.2 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 7: 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.

1.10.4.3 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).

1.10.5 Bayesian Approach

1.10.5.1 Model description A gaussian model is fitted to each of the models. Model m0 is the base model with only an intercept. Model m1 is an extension of model m0 where the version in introduced as a predictor. Model m2 is again an extension of model m0 with portal as a predictor. In model m3, both predictors are added as main effects, and model m4 extends model m3 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

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

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

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

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

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

```r
#Fitting each variant of the model

m0 <-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
)
```

1.10.5.2 Model comparison

```
## Computing WAIC
```

```r
m1 <-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
```

```r
m2 <-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
```

```r
m3 <-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
```

```r
m4 <-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(m0,m1,m2,m3,m4))
```

	WAIC	SE	dWAIC	dSE	pWAIC	weight
m4	3780	85.85	0	NA	7.358	1
m3	3821	81.83	40.94	11.12	6.234	1.291e-09
m2	3855	81.89	75.53	16.96	5.185	3.978e-17
m1	3976	69.28	196.6	31.34	4.502	2.074e-43
m0	4003	69.9	223.6	33.11	3.427	2.821e-49

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

	mean	sd	2.5%	97.5%	n_eff	Rhat4
a	-0.6938	0.4486	-1.56	0.1702	3863	1.001
b	0.9457	0.2856	0.3868	1.507	3958	1.001
c	2.904	0.2847	2.351	3.453	3856	1.001
d	-1.058	0.1801	-1.41	-0.7054	3929	1.001
sigma	1.6	0.03626	1.531	1.673	6873	1

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).