UNIVERSITY OF LEEDS

School of Mathematics

Declaration of Academic Integrity for Individual Pieces of Work

I declare that I am aware that as a member of the University community at the University of Leeds I have committed to working with Academic Integrity and that this means that my work must be a true expression of my own understanding and ideas, giving credit to others where their work contributes to mine.

I declare that the attached submission is my own work.

Where the work of others has contributed to my work, I have given full acknowledgement using the appropriate referencing conventions for my programme of study.

I confirm that the attached submission has not been submitted for marks or credits in a different module or for a different qualification or completed prior to entry to the University.

I have read and understood the University's rules on Academic Misconduct. I know that if I commit an academic misconduct offence there can be serious disciplinary consequences.

I re-confirm my consent to the University copying and distributing any or all of my work in any form and using third parties to verify that this is my own work, and for quality assurance purposes.

I confirm that I have declared all mitigating circumstances that may be relevant to the assessment of this piece of work and I wish to have taken into account.

Rajarshi Nandi

Student Signature: **Student Number: 201791225**

Student Name: Rajarshi Nandi **Date: 18/08/2024**

Please note:

When you become a registered student of the University at first and any subsequent registration you sign the following authorisation and declaration:

"I confirm that the information I have given on this form is correct. I agree to observe the provisions of the University's Charter, Statutes, Ordinances, Regulations and Codes of Practice for the time being in force. I know that it is my responsibility to be aware of their contents and that I can read them on the University web site. I acknowledge my obligation under the Payment of Fees Section in the Handbook to pay all charges to the University on demand.

I agree to the University processing my personal data (including sensitive data) in accordance with its Code of Practice on Data Protection http://www.leeds.ac.uk/dpa . I consent to the University making available to third parties (who may be based outside the European Economic Area) any of my work in any form for standards and monitoring purposes including verifying the absence of plagiarised material. I agree that third parties may retain copies of my work for these purposes on the understanding that the third party will not disclose my identity."'

Analysis of Factors Affecting Business Analyst Wages

Multiple Linear Regression Analysis

To analyze the factors affecting the wages of business analysts, I performed a multiple linear regression analysis using the available data. I started by examining the correlation between variables using correlation plots and a scatterplot matrix.

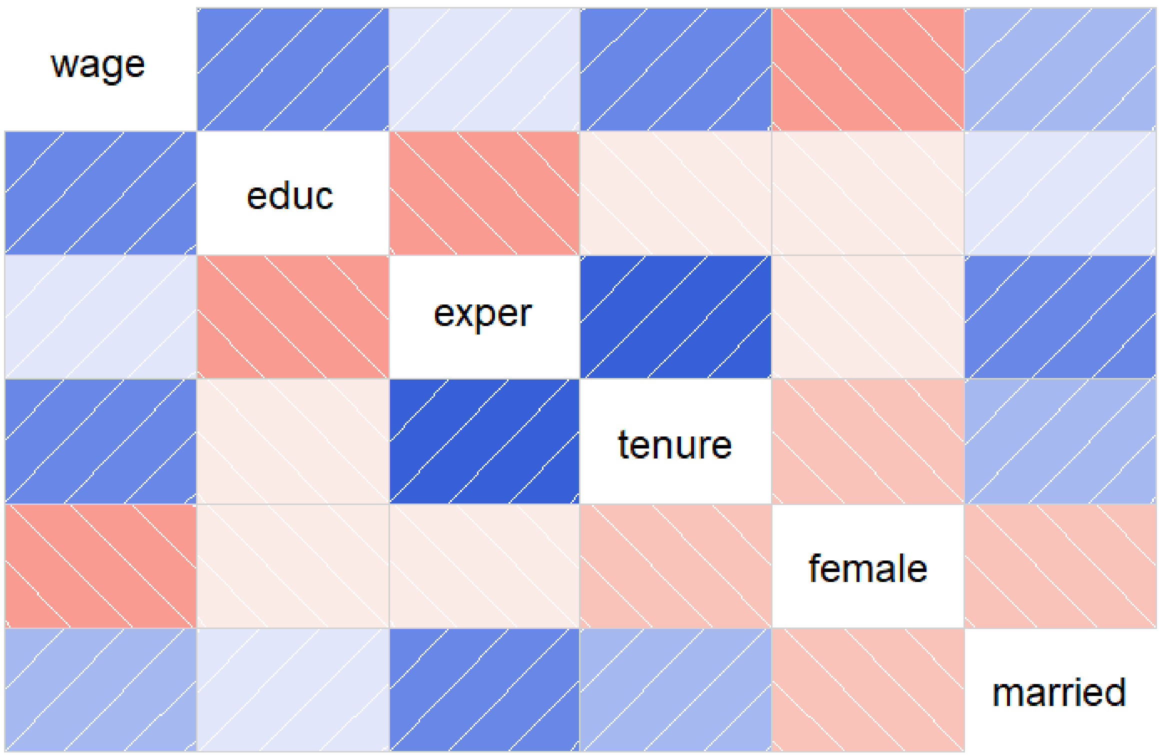

Figure 1.1: Correlation plot for every variable (dependent and independent).

From the correlation plot (Figure 1.1), I observed that:
1. Wage has a strong positive correlation with education (educ) and tenure.
2. There's a moderate negative correlation between wage and being female.
3. Experience (exper) shows a positive correlation with wage, but it's not as strong as education or tenure.
4. Being married has a weak positive correlation with wage.

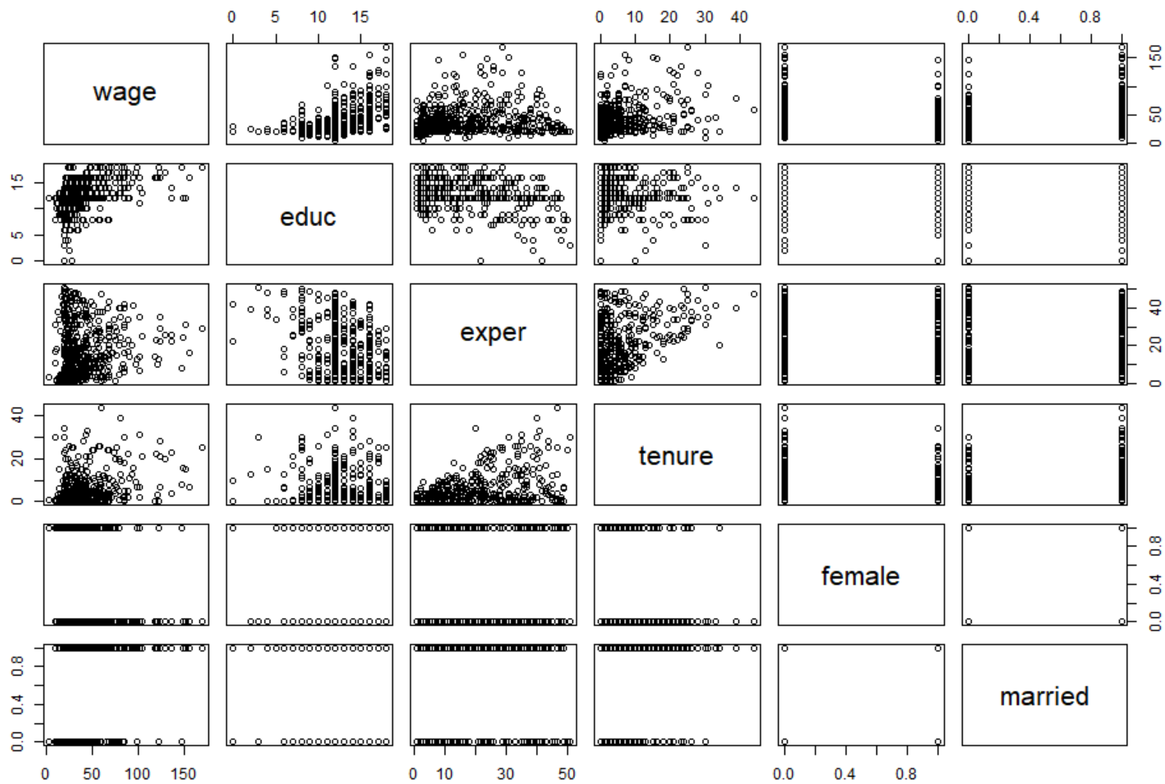

Figure 1.2: Scatter plot for every variable (dependent and independent).

The scatterplot matrix (Figure 1.2) provides additional insights:
1. The relationship between wage and education appears to be positive and roughly linear.
2. There's a positive but more scattered relationship between wage and experience/tenure.
3. The binary nature of the female and married variables is evident from their plots.

Based on these observations, I decided to include all available variables in the initial regression model:

fit <- lm(wage ~ educ + exper + tenure + female + married, data=data)

The results of this regression show:

1. Education (educ) has a highly significant positive effect on wage ($p < 2e-16$).
2. Tenure also has a highly significant positive effect ($p = 1.25e-10$).
3. Being female has a significant negative effect on wage ($p = 1.52e-10$).
4. Experience (exper) and being married are not statistically significant at the 0.05 level.

The adjusted R-squared value of 0.3621 indicates that the model explains about 36.21% of the variation in wages, which is moderate but not extremely high. This suggests that while these factors are important, there are likely other unmeasured factors influencing wages.

To refine the model, I removed the insignificant variables (experience and married status):

fit2 <- lm(wage ~ educ + tenure + female, data=data)

I then compared the two models using ANOVA:

anova(fit,fit2)

The ANOVA results show a significant difference between the models (p = 0.01358), suggesting that the full model might be slightly better. However, for simplicity and to focus on the most significant factors, I decided to proceed with the reduced model (fit2) for further analysis.

b) Estimating Expected Payment

To estimate my expected payment per hour, I used the reduced model (fit2) with the following characteristics:

- Education: 16 years (bachelor's degree)
- Tenure: 2 years (my tenure at IBM)
- Gender: Male (female = 0)

Using these values, I predicted the expected wage:

predict(fit2, list(educ=16, tenure=2, female=0))

The predicted wage is **$55.07** per hour. This estimate is based on the simplified model and assumes all other factors are average.

c) Interaction Effects Between Job Tenure and Gender

To analyze whether job tenure has the same effect on wage for male and female employees, I created a new model with an interaction term:

fit5 <- lm(wage ~ tenure * female, data=data)

The results show:

1. A significant positive effect of tenure on wage (p < 2e-16).
2. A significant negative effect of being female on wage (p = 1.08e-05).
3. A significant negative interaction between tenure and being female (p = 0.012).

The interaction term (tenure:female) is negative and statistically significant, indicating that the effect of tenure on wage is different for male and female employees. Specifically, the positive effect of tenure on wage is less pronounced for female employees compared to male employees.

For male employees (female = 0), each year of tenure increases wage by about $1.22 per hour. For female employees, the effect of each year of tenure is $1.22 - $0.78 = $0.44 per hour. This suggests that female employees benefit less from increased tenure compared to their male counterparts, indicating a potential gender disparity in wage growth over time.

In conclusion, this analysis reveals that education, tenure, and gender are significant factors affecting business analyst wages. However, the model's explanatory power is moderate, suggesting other unmeasured factors play a role. The analysis also uncovers a concerning gender disparity in how tenure affects wage growth. These findings could be valuable for understanding wage structures and addressing potential inequities in the field of business analysis.

Report on Topic Modeling of Amazon Mobile Phone Reviews

Introduction:

In this analysis, I performed topic modeling on a sample of Amazon mobile phone reviews to identify the key factors discussed in positive and negative reviews. I used a dataset of over 30,000 reviews and randomly selected a sample of 5,000 for analysis. The goal was to uncover the main topics that influence customer satisfaction and dissatisfaction.

Methodology:

1. Data Sampling:

I used the sample_n() function from the ‘dplyr’ package to randomly select 5,000 reviews from the dataset. To ensure reproducibility, I set the seed using set.seed(225) before sampling.

2. Identifying Positive and Negative Reviews:

To differentiate between positive and negative reviews, I used the rating system provided in the dataset. Reviews with ratings of 4 or 5 stars were classified as positive, while those with 1 or 2 stars were classified as negative. I excluded 3-star reviews as they were considered neutral and could potentially dilute the analysis of clearly positive or negative sentiments.

3. Text Preprocessing:

I performed several preprocessing steps to clean and prepare the text data:

- Converted text to UTF-8 encoding to handle special characters
- Removed punctuation and numbers
- Converted all text to lowercase
- Removed stop words (common words that don't contribute much to the meaning)
- Applied lemmatization to reduce words to their base form

4. Word cloud analysis:

To complement the topic modeling analysis, I generated word clouds for both positive and negative reviews. These visualizations provide an intuitive representation of the most frequent terms in each category, offering additional insights into customer sentiments. Following are the output for positive and negative word clouds respectively.

Figure 2.1: Positive word cloud.

Figure 2.2: Negative word cloud.

5. Document-Term Matrix Creation:
I created separate document-term matrices for positive and negative reviews using the tm package. This step converted the preprocessed text into a structured format suitable for topic modeling.

6. Determining the Optimal Number of Topics:
To select the appropriate number of topics for both positive and negative reviews, I used the 'ldatuning' package. This package implements several metrics to evaluate the quality of topic models with different numbers of topics. I used three metrics: Griffiths2004, CaoJuan2009, and Arun2010. The optimal number of topics is typically where these metrics converge or show significant changes. Each metric has different characteristics for determining the optimal number of topics:

1. Griffiths2004: This metric should be maximized. It shows a sharp increase up to 8-9 topics, then plateaus with slight fluctuations.

2. CaoJuan2009: This metric should be minimized. It shows high variability but has notable low points at 4 and 13 topics.

3. Arun2010: This metric should be minimized. It shows a steady decrease as the number of topics increases, with the lowest point at 18-19 topics.

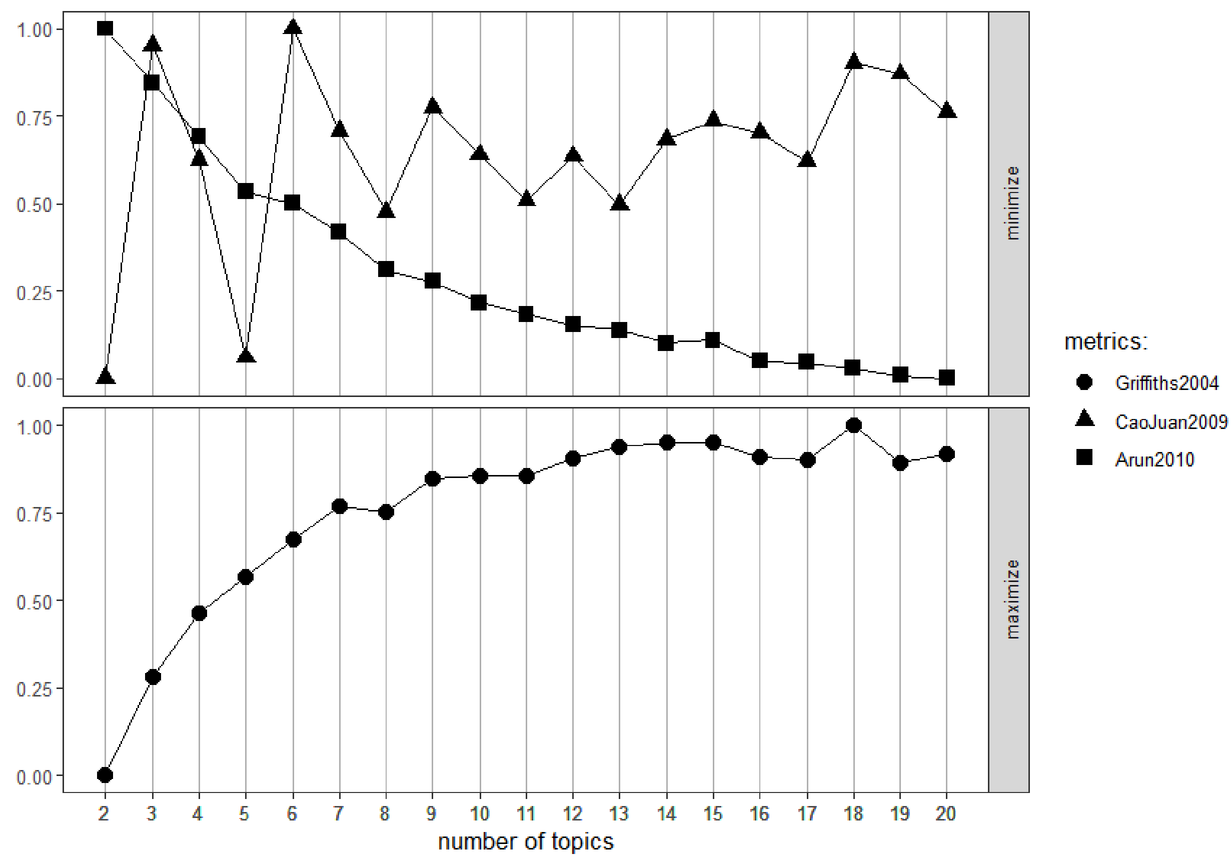

Figure 2.3: Positive topic number determination using Arun2010 &CaoJuan2009 (minimizing) and Griffiths2004 (maximizing).

For positive reviews, based on the metrics shown in Figure 2.3, I chose 7 topics. The Griffiths2004 metric plateaus around this point, while the CaoJuan2009 and Arun2010 metrics show relatively stable values.

Analyzing each metric:

1. Griffiths2004 (bottom graph): This metric should be maximized. It shows a sharp increase up to about 6-7 topics, then continues to increase more slowly, reaching its peak around 18-19 topics.
2. CaoJuan2009 (triangles in top graph): This metric should be minimized. It shows local minima at 2, 6, and 13 topics, with the global minimum at 6 topics.
3. Arun2010 (squares in top graph): This metric should also be minimized. It shows a sharp decrease until about 6-7 topics, then continues to decrease more slowly.

Considering all three metrics together, the optimal number of topics appears to be around 6-7. Here's why:

1. At 6-7 topics, Griffiths2004 has already shown significant improvement and is starting to level off.

2. CaoJuan2009 reaches its global minimum at 6 topics.
3. Arun2010 shows a sharp elbow at 6-7 topics, after which the rate of decrease slows considerably.

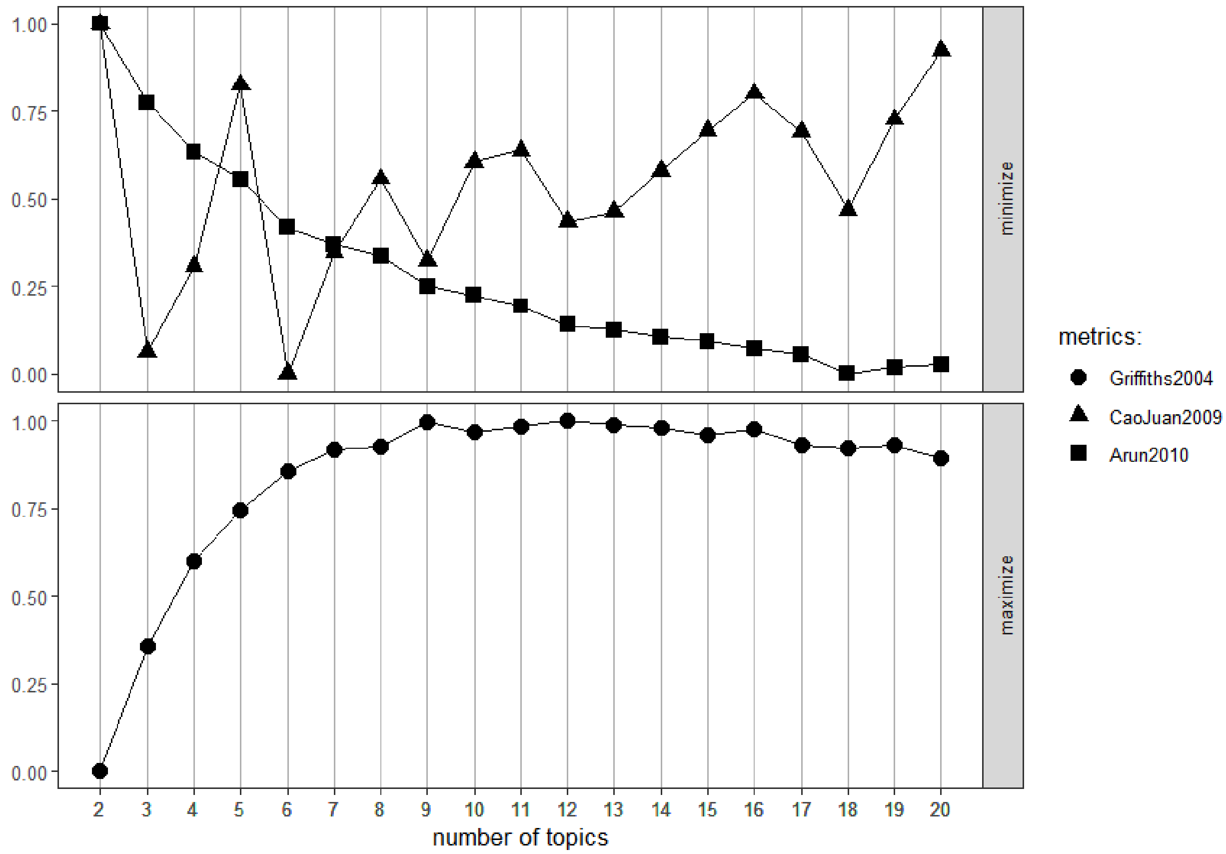

Figure 2.4: Negative topic number determination using Arun2010 &CaoJuan2009 (minimizing) and Griffiths2004 (maximizing).

For negative reviews, Figure 2.4 shows three different metrics (Griffiths2004, CaoJuan2009, and Arun2010) plotted against the number of topics, ranging from 2 to 20.

Analysis:

- Griffiths2004 suggests that the optimal number of topics is around 8-9, where it reaches its peak before plateauing.
- CaoJuan2009 has local minima at 4 and 13 topics, suggesting these could be optimal points.
- Arun2010 continuously decreases, which might suggest that more topics are better, but this needs to be balanced against the risk of overfitting.

Considering all three metrics:

1. 8-9 topics seem to be a good compromise. This is where Griffiths2004 reaches its peak, and it's also a point where CaoJuan2009 is relatively low.

2. 13 topics could be another option, as it's a local minimum for CaoJuan2009 and still has a high Griffiths2004 score.

3. 4 topics is a strong local minimum for CaoJuan2009, but it might be too few topics for complex text data.

7. Topic Modeling:
I used Latent Dirichlet Allocation (LDA) with the chosen number of topics for both positive and negative reviews. The LDA algorithm was run using the Gibbs sampling method with 1000 iterations to ensure convergence.

Results and Discussion:

Positive Reviews - Top 3 Factors Affecting Customer Satisfaction:

1. Camera Quality (Topic 6):
The words "camera", "quality", "good", "awesome", and "performance" suggest that camera quality is a significant factor in positive reviews. Customers seem to be highly satisfied with the camera performance of their mobile phones.

2. Battery Life (Topic 3):
Terms like "battery", "life", "fast", and "charging" indicate that battery performance is another crucial factor. Long battery life and fast charging capabilities appear to contribute significantly to customer satisfaction.

3. Value for Money (Topic 4):
Words such as "best", "price", "range", and "value" suggest that customers appreciate phones that offer good value for their price. The presence of "redmi" and "note" might indicate that certain models are perceived as particularly good value.

Negative Reviews - Top 3 Factors Affecting Customer Dissatisfaction:

1. Battery Issues (Topic 3):
The presence of "battery", "life", and "charger" in the negative context suggests that poor battery performance or charging issues are a major source of dissatisfaction.

2. Camera Problems (Topic 8):
Words like "camera", "quality", and "poor" indicate that subpar camera performance is a significant factor in negative reviews.

3. Software and Performance Issues (Topic 9):
Terms such as "problem", "working", "software", and "automatically" suggest that software bugs, performance issues, or unexpected behavior of the phone lead to customer dissatisfaction.

Additional Insights:

- The appearance of brand names like "oneplus", "redmi", and "nord" in both positive and negative topics suggests that these brands have mixed receptions among customers.
- Customer service and product reliability seem to be recurring themes in negative reviews, as evidenced by words like "issue", "return", and "service" across multiple topics.

Conclusion:

This topic modeling analysis reveals that camera quality, battery life, and value for money are the primary factors driving customer satisfaction in mobile phone reviews. Conversely, issues with battery, camera quality, and software/performance problems are the main sources of dissatisfaction.

These insights can be valuable for mobile phone manufacturers and retailers to focus on improving key areas that matter most to customers. Future research could involve a more detailed analysis of specific brands or price ranges to uncover more patterns in customer satisfaction and dissatisfaction.

APPENDIX

Code 1 (Analysis of Factors Affecting Business Analyst Wages):

```
data <- read.csv("B3.csv")

head(data)

summary(data)

## Correlation plot with corrplot

#install.packages("corrplot")
library("corrplot")

corrplot(cor(data[,c("wage", "educ", "exper", "tenure", "female", "married")]))

#install.packages("corrgram")
library("corrgram")
## Correlation plot with corrgram
corrgram(data[,c("wage", "educ", "exper", "tenure", "female", "married")])
```

```
## Scatterplot
pairs(data[,c("wage", "educ", "exper", "tenure", "female", "married")])

## Multiple Linear Regression----

## Run regression
fit <- lm(wage ~ educ + exper + tenure + female + married, data=data)

## See results
summary(fit)

## Remove insiginficant variables (income and frost)

fit2 <- lm(wage ~ educ + tenure + female, data=data)


## Compare models

anova(fit,fit2)

## Prediction----

predict(fit2, list(educ=15, tenure=2, female=0))

## Interaction----

fit5 <- lm(wage ~ tenure * female, data=data)
summary(fit5)
```

Output for Code 1:

```
> data <- read.csv("B3.csv")
>
> head(data)
  wage educ exper tenure female married
1 21.05   11     2      0      1       0
2 22.00   12    22      2      1       1
3 20.37   11     2      0      0       0
4 40.75    8    44     28      0       1
5 35.99   12     7      2      0       1
6 59.42   16     9      8      0       1
>
> summary(data)
     wage           educ          exper          tenure          female         married
 Min.   :  3.60   Min.   : 0.00   Min.   : 1.00   Min.   : 0.000   Min.   :0.0000   Min.   :0.0000
```

```
 1st Qu.: 22.61   1st Qu.:12.00   1st Qu.: 5.00   1st Qu.: 0.000   1st Qu.:0.0000   1st Qu.:0.0000
 Median : 31.58   Median :12.00   Median :13.50   Median : 2.000   Median :0.0000   Median
:1.0000
 Mean   : 40.04   Mean   :12.56   Mean   :17.02   Mean   : 5.105   Mean   :0.4791   Mean
:0.6084
 3rd Qu.: 46.72   3rd Qu.:14.00   3rd Qu.:26.00   3rd Qu.: 7.000   3rd Qu.:1.0000   3rd
Qu.:1.0000
 Max.   :169.64   Max.   :18.00   Max.   :51.00   Max.   :44.000   Max.   :1.0000   Max.   :1.0000
>
> ## Correlation plot with corrplot
>
> #install.packages("corrplot")
> library("corrplot")
corrplot 0.92 loaded
>
> corrplot(cor(data[,c("wage", "educ", "exper", "tenure", "female", "married")]))
>
> #install.packages("corrgram")
> library("corrgram")
> ## Correlation plot with corrgram
> corrgram(data[,c("wage", "educ", "exper", "tenure", "female", "married")])
>
> ## Scatterplot
> pairs(data[,c("wage", "educ", "exper", "tenure", "female", "married")])
>
> ## Multiple Linear Regression----
>
> ## Run regression
> fit <- lm(wage ~ educ + exper + tenure + female + married, data=data)
>
> ## See results
> summary(fit)

Call:
lm(formula = wage ~ educ + exper + tenure + female + married,
    data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-52.507 -12.332  -3.394   7.137  94.583

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -10.99020    4.91044  -2.238   0.0256 *
```

```
educ          3.77379    0.33864  11.144  < 2e-16 ***
exper         0.12730    0.08169   1.558   0.1198    
tenure        0.94247    0.14353   6.566 1.25e-10 ***
female      -11.82640    1.80979  -6.535 1.52e-10 ***
married       3.79810    1.94195   1.956   0.0510 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 20.03 on 520 degrees of freedom
Multiple R-squared:  0.3682,  Adjusted R-squared:  0.3621 
F-statistic: 60.61 on 5 and 520 DF,  p-value: < 2.2e-16

> 
> ## Remove insiginficant variables (income and frost)
> 
> fit2 <- lm(wage ~ educ + tenure + female, data=data)
> 
> 
> ## Compare models
> 
> anova(fit,fit2)
Analysis of Variance Table

Model 1: wage ~ educ + exper + tenure + female + married
Model 2: wage ~ educ + tenure + female
  Res.Df    RSS Df Sum of Sq      F  Pr(>F)  
1    520 208655                              
2    522 212133 -2   -3478.7 4.3348 0.01358 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> 
> ## Prediction----
> 
> predict(fit2, list(educ=15, tenure=2, female=0))
       1 
51.29819 
> 
> ## Interaction----
> 
> fit5 <- lm(wage ~ tenure * female, data=data)
> summary(fit5)

Call:
lm(formula = wage ~ tenure * female, data = data)
```

```
Residuals:
    Min      1Q  Median      3Q     Max 
-66.816 -12.093  -6.278   8.675 113.793 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)    40.2893     1.7123  23.529  < 2e-16 ***
tenure          1.2239     0.1620   7.554 1.90e-13 ***
female        -10.7356     2.4154  -4.445 1.08e-05 ***
tenure:female  -0.7810     0.3097  -2.522    0.012 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 22.4 on 522 degrees of freedom
Multiple R-squared:  0.2067,  Adjusted R-squared:  0.2021 
F-statistic: 45.33 on 3 and 522 DF,  p-value: < 2.2e-16
```

Code 2 (Report on Topic Modeling of Amazon Mobile Phone Reviews):

```
install.packages("dplyr")
install.packages("tm")
install.packages("stringr")
install.packages("RColorBrewer")
install.packages("wordcloud")
install.packages("topicmodels")
install.packages("ggplot2")
install.packages("LDAvis")
install.packages("servr")
install.packages("textcat")
install.packages("jsonlite")
install.packages("ldatuning")

library(stats)
library(dplyr) # basic data manipulation
library(tm) # package for text mining package
library(stringr) # package for dealing with strings
library(RColorBrewer)# package to get special theme color
```

```
library(wordcloud) # package to create wordcloud
library(topicmodels) # package for topic modelling
library(ggplot2) # basic data visualization
library(LDAvis) # LDA specific visualization
library(servr) # interactive support for LDA visualization
library(textcat)
library(jsonlite)
library(NLP)


#Downloading all data
data  <- fromJSON("https://query.data.world/s/4ria2tfww73wmhfzke5z2w4zlez2re")

#Fetching my data
set.seed(225)
reviews <-sample_n(data, 5000)

#Unique ratings
unique(reviews$review_rating)

#String to int conversion of ratings
reviews$rating <- as.numeric(str_sub(reviews$review_rating,1,1))
summary(reviews)

#Dropping Neutral Reviews and NULL values
reviews_final <- reviews[reviews$rating != 3, ]

reviews_final_corp <- reviews_final[, c("rating", "review_text")]

neg_set <- reviews_final_corp[reviews_final_corp$rating < 3,]
pos_set <- reviews_final_corp[reviews_final_corp$rating > 3,]

#Inspecting the reviews
head(pos_set,1)
head(neg_set,1)

#Correct encoding
pos_reviews <- stringr::str_conv(pos_set$review_text, "UTF-8")
pos_docs <- Corpus(VectorSource(pos_reviews))
neg_reviews <- stringr::str_conv(neg_set$review_text, "UTF-8")
neg_docs <- Corpus(VectorSource(neg_reviews))

pos_dtmdocs <- DocumentTermMatrix(pos_docs,
                        control = list(lemma=TRUE,removePunctuation = TRUE,
```

```
                                              removeNumbers = TRUE, stopwords = TRUE,
                                              tolower = TRUE))
pos_raw.sum=apply(pos_dtmdocs,1,FUN=sum)
pos_dtmdocs=pos_dtmdocs[pos_raw.sum!=0,]
neg_dtmdocs <- DocumentTermMatrix(neg_docs,
                                control = list(lemma=TRUE,removePunctuation = TRUE,
                                              removeNumbers = TRUE, stopwords = TRUE,
                                              tolower = TRUE))
neg_raw.sum=apply(neg_dtmdocs,1,FUN=sum)
neg_dtmdocs=neg_dtmdocs[neg_raw.sum!=0,]

#Positive word cloud
library(wordcloud)
pos_dtm.new <- as.matrix(pos_dtmdocs)
pos_frequency <- colSums(pos_dtm.new)
pos_frequency <- sort(pos_frequency, decreasing=TRUE)
pos_doc_length <- rowSums(pos_dtm.new)

pos_frequency[1:10]

pos_words <- names(pos_frequency)

wordcloud(pos_words[1:100], pos_frequency[1:100], rot.per=0.15,
        random.order = FALSE, scale=c(4,0.5),
        random.color = FALSE, colors=brewer.pal(8,"Dark2"))
title(main = "Positive review wordcloud")

#Negative word cloud
neg_dtm.new <- as.matrix(neg_dtmdocs)
neg_frequency <- colSums(neg_dtm.new)
neg_frequency <- sort(neg_frequency, decreasing=TRUE)
neg_doc_length <- rowSums(neg_dtm.new)

neg_frequency[1:10]

neg_words <- names(neg_frequency)

wordcloud(neg_words[1:100], neg_frequency[1:100], rot.per=0.15,
        random.order = FALSE, scale=c(4,0.5),
        random.color = FALSE, colors=brewer.pal(8,"Dark2"))
title(main = "Negative review wordcloud")

#Determining number of topics (positive)
library(ldatuning)
```

```
pos_result <- FindTopicsNumber(
  pos_dtm.new,
  topics = seq(from = 2, to = 20, by = 1),
  metrics = c("Griffiths2004", "CaoJuan2009", "Arun2010"),
  method = "Gibbs",
  control = list(seed = 430),
  mc.cores = 2L,
  verbose = TRUE
)
FindTopicsNumber_plot(pos_result)

#Determining number of topics (negative)
neg_result <- FindTopicsNumber(
  neg_dtm.new,
  topics = seq(from = 2, to = 20, by = 1),
  metrics = c("Griffiths2004", "CaoJuan2009", "Arun2010"),
  method = "Gibbs",
  control = list(seed = 430),
  mc.cores = 2L,
  verbose = TRUE
)
FindTopicsNumber_plot(neg_result)

#Topic modelling
library(dplyr)
pos_ldaOut <-LDA(pos_dtmdocs,7, method="Gibbs",
                control=list(iter=1000,seed=430))
neg_ldaOut <-LDA(neg_dtmdocs,9, method="Gibbs",
                control=list(iter=1000,seed=430))

#Positive topic labeling
pos_ldaOut.terms <- as.matrix(terms(pos_ldaOut, 10))
pos_ldaOut.terms

#Negative topic labeling
neg_ldaOut.terms <- as.matrix(terms(neg_ldaOut, 10))
neg_ldaOut.terms

#Top 3 factors in Positive reviews visualized
library(ggplot2)
pos_ldaOut.topics <- data.frame(topics(pos_ldaOut))
pos_ldaOut.topics$index <- as.numeric(row.names(pos_ldaOut.topics))
ggplot(pos_ldaOut.topics, aes(x = topics.pos_ldaOut.)) +
  geom_bar(fill = "skyblue", color = "black") +
```

```
  labs(title = "Count Plot for Positive Review topics", x = "Topic", y = "Count") +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 15, face = "bold"),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12)
  )

#Top 3 factors in Negative reviews visualized
neg_ldaOut.topics <- data.frame(topics(neg_ldaOut))
neg_ldaOut.topics$index <- as.numeric(row.names(neg_ldaOut.topics))
ggplot(neg_ldaOut.topics, aes(x = topics.neg_ldaOut.))+
  geom_bar(fill = "skyblue", color = "black") +
  labs(title = "Count Plot for Negative Review topics", x = "Topic", y = "Count") +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 15, face = "bold"),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12)
  )
```

Output for Code 2:

```
#Removing non-english reviews
> data  <- fromJSON("https://query.data.world/s/4ria2tfww73wmhfzke5z2w4zlez2re")
>
> #Fetching data
> set.seed(225)
> reviews <-sample_n(data, 5000)
>
> #Unique ratings
> unique(reviews$review_rating)
[1] "5.0 out of 5 stars" "2.0 out of 5 stars" "4.0 out of 5 stars" "1.0 out of 5 stars" "3.0 out of 5
stars"
>
> #String to int conversion of ratings
> reviews$rating <- as.numeric(str_sub(reviews$review_rating,1,1))
> summary(reviews)
   product          product_company    profile_name       review_title       review_rating
review_text
```

Length:5000 Length:5000 Length:5000 Length:5000 Length:5000
Length:5000
Class :character Class :character Class :character Class :character Class :character
Class :character
Mode :character Mode :character Mode :character Mode :character Mode :character
Mode :character

```
 helpful_count      total_comments   review_country     reviewed_at           url             crawled_at
 Length:5000        Min.   : 0.000   Length:5000        Length:5000        Length:5000        Min.
:1.603e+12
 Class :character   1st Qu.: 0.000   Class :character   Class :character   Class :character   1st
Qu.:1.603e+12
 Mode  :character   Median : 0.000   Mode  :character   Mode  :character   Mode  :character
Median :1.603e+12
                    Mean   : 0.077                                                            Mean   :1.603e+12
                    3rd Qu.: 0.000                                                            3rd Qu.:1.603e+12
                    Max.   :21.000                                                            Max.   :1.603e+12
```

```
    _id           verified_purchase     color           style_name        size_name          category
 Length:5000        Length:5000        Length:5000        Length:5000        Length:5000
Length:5000
 Class :character   Class :character   Class :character   Class :character   Class :character
Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character
Mode  :character
```

sub_category	images.Length	images.Class	images.Mode	rating
Length:5000	0	-none-	character	Min. :1.000
Class :character	0	-none-	character	1st Qu.:3.000
Mode :character	0	-none-	character	Median :4.000
	0	-none-	character	Mean :4.016
	0	-none-	character	3rd Qu.:5.000
	0	-none-	character	Max. :5.000
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	0	-none-	character	
	1	-none-	character	
	4	-none-	character	
	0	-none-	character	
	0	-none-	character	

```
                 0        -none-     character
                 0        -none-     character
                 5        -none-     character
                 0        -none-     character
                 0        -none-     character
                 2        -none-     character
                 0        -none-     character
                 0        -none-     character
                 0        -none-     character
                 2        -none-     character
                 0        -none-     character
                 0        -none-     character
                 0        -none-     character
                 0        -none-     character
                 0        -none-     character
                 0        -none-     character
                 0        -none-     character
                 0        -none-     character
                 1        -none-     character
                 0        -none-     character
                 0        -none-     character
                 0        -none-     character
                 0        -none-     character
                 0        -none-     character
 [ reached getOption("max.print") -- omitted 4953 rows ]
>
> #Dropping Neutral Reviews and NULL values
> reviews_final <- reviews[reviews$rating != 3, ]
>
> reviews_final_corp <- reviews_final[, c("rating", "review_text")]
>
> neg_set <- reviews_final_corp[reviews_final_corp$rating < 3,]
> pos_set <- reviews_final_corp[reviews_final_corp$rating > 3,]
>
> #Inspecting the reviews
> head(pos_set,1)
  rating     review_text
1      5 \n  Great.. !\n
> head(neg_set,1)
  rating                               review_text
3      2 \n  Yes quality is so third grade category\n
>
> #Correct encoding
> pos_reviews <- stringr::str_conv(pos_set$review_text, "UTF-8")
```

```
> pos_docs <- Corpus(VectorSource(pos_reviews))
> neg_reviews <- stringr::str_conv(neg_set$review_text, "UTF-8")
> neg_docs <- Corpus(VectorSource(neg_reviews))
>
> pos_dtmdocs <- DocumentTermMatrix(pos_docs,
+                             control = list(lemma=TRUE,removePunctuation = TRUE,
+                                            removeNumbers = TRUE, stopwords = TRUE,
+                                            tolower = TRUE))
> pos_raw.sum=apply(pos_dtmdocs,1,FUN=sum)
> pos_dtmdocs=pos_dtmdocs[pos_raw.sum!=0,]
> neg_dtmdocs <- DocumentTermMatrix(neg_docs,
+                             control = list(lemma=TRUE,removePunctuation = TRUE,
+                                            removeNumbers = TRUE, stopwords = TRUE,
+                                            tolower = TRUE))
> neg_raw.sum=apply(neg_dtmdocs,1,FUN=sum)
> neg_dtmdocs=neg_dtmdocs[neg_raw.sum!=0,]
>
> #Positive word cloud
> library(wordcloud)
> pos_dtm.new <- as.matrix(pos_dtmdocs)
> pos_frequency <- colSums(pos_dtm.new)
> pos_frequency <- sort(pos_frequency, decreasing=TRUE)
> pos_doc_length <- rowSums(pos_dtm.new)
>
> pos_frequency[1:10]
  good   phone  camera battery    best product    nice quality   price  mobile
  1521    1264     809     598     495     484     481     443     399     365
>
> pos_words <- names(pos_frequency)
>
> wordcloud(pos_words[1:100], pos_frequency[1:100], rot.per=0.15,
+          random.order = FALSE, scale=c(4,0.5),
+          random.color = FALSE, colors=brewer.pal(8,"Dark2"))
> title(main = "Positive review wordcloud")
>
> #Negative word cloud
> neg_dtm.new <- as.matrix(neg_dtmdocs)
> neg_frequency <- colSums(neg_dtm.new)
> neg_frequency <- sort(neg_frequency, decreasing=TRUE)
> neg_doc_length <- rowSums(neg_dtm.new)
>
> neg_frequency[1:10]
 phone  camera battery    good quality product  mobile problem     one     bad
   211     146     108     105      93      93      82      77      68      68
```

```
>
> neg_words <- names(neg_frequency)
>
> wordcloud(neg_words[1:100], neg_frequency[1:100], rot.per=0.15,
+          random.order = FALSE, scale=c(4,0.5),
+          random.color = FALSE, colors=brewer.pal(8,"Dark2"))
> title(main = "Negative review wordcloud")
>
> #Determining number of topics (positive)
> library(ldatuning)
> pos_result <- FindTopicsNumber(
+   pos_dtm.new,
+   topics = seq(from = 2, to = 20, by = 1),
+   metrics = c("Griffiths2004", "CaoJuan2009", "Arun2010"),
+   method = "Gibbs",
+   control = list(seed = 430),
+   mc.cores = 2L,
+   verbose = TRUE
+ )
fit models... done.
calculate metrics:
  Griffiths2004... done.
  CaoJuan2009... done.
  Arun2010... done.
> FindTopicsNumber_plot(pos_result)
>
> #Determining number of topics (negative)
> neg_result <- FindTopicsNumber(
+   neg_dtm.new,
+   topics = seq(from = 2, to = 20, by = 1),
+   metrics = c("Griffiths2004", "CaoJuan2009", "Arun2010"),
+   method = "Gibbs",
+   control = list(seed = 430),
+   mc.cores = 2L,
+   verbose = TRUE
+ )
fit models... done.
calculate metrics:
  Griffiths2004... done.
  CaoJuan2009... done.
  Arun2010... done.
> FindTopicsNumber_plot(neg_result)
>
> #Topic modelling
```

```
> library(dplyr)
> pos_ldaOut <-LDA(pos_dtmdocs,7, method="Gibbs",
+                 control=list(iter=1000,seed=430))
> neg_ldaOut <-LDA(neg_dtmdocs,9, method="Gibbs",
+                 control=list(iter=1000,seed=430))
>
> #Positive topic labeling
> pos_ldaOut.terms <- as.matrix(terms(pos_ldaOut, 10))
> pos_ldaOut.terms
      Topic 1     Topic 2    Topic 3      Topic 4     Topic 5    Topic 6     Topic 7
 [1,] "mobile"    "good"     "battery"    "best"      "phone"    "camera"     "one"
 [2,] "phone"     "nice"     "life"       "phone"     "good"     "quality"    "like"
 [3,] "awesome"   "product"  "fast"       "price"     "great"    "good"       "can"
 [4,] "money"     "overall"  "also"       "range"     "really"   "awesome"    "oneplus"
 [5,] "excellent" "super"    "fingerprint" "redmi"    "features" "performance" "using"
 [6,] "amazing"   "satisfied" "use"       "better"    "look"     "display"    "just"
 [7,] "value"     "backup"   "camera"     "note"      "colour"   "back"       "will"
 [8,] "budget"    "pubg"     "reader"     "buy"       "problem"  "processor"  "time"
 [9,] "everything" "happy"   "charging"   "smartphone" "looks"   "perfect"    "nord"
[10,] "amazon"    "improve"  "finger"     "pro"       "premium"  "design"     "plus"
>
> #Negative topic labeling
> neg_ldaOut.terms <- as.matrix(terms(neg_ldaOut, 10))
> neg_ldaOut.terms
      Topic 1       Topic 2     Topic 3   Topic 4   Topic 5   Topic 6   Topic 7     Topic 8      Topic 9
 [1,] "phone"       "using"     "battery" "one"     "phone"   "buy"     "mobile"    "camera"
"problem"
 [2,] "like"        "bad"       "good"    "nord"    "screen"  "product" "issue"     "quality"    "working"
 [3,] "even"        "day"       "days"    "plus"    "bad"     "dont"    "worst"     "poor"       "return"
 [4,] "product"     "issues"    "life"    "better"  "call"    "display" "time"      "pro"        "oneplus"
 [5,] "just"        "good"      "fast"    "also"    "product" "worst"   "amazon"    "redmi"
"automatically"
 [6,] "disappointed" "many"     "month"   "properly" "average" "network" "waste"    "note"
"use"
 [7,] "price"       "like"      "will"    "hang"    "heating" "oneplus" "service"   "times"      "can"
 [8,] "issue"       "bluetooth" "charger" "low"     "phones"  "mobile"  "money"     "video"      "one"
 [9,] "always"      "ever"      "nice"    "much"    "charge"  "charging" "experience" "replacement"
"issue"
[10,] "use"         "display"   "usage"   "getting" "started" "received" "got"      "charging"
"software"
>
> #Top 3 factors in Positive reviews visualized
> library(ggplot2)
> pos_ldaOut.topics <- data.frame(topics(pos_ldaOut))
```

```
> pos_ldaOut.topics$index <- as.numeric(row.names(pos_ldaOut.topics))
> ggplot(pos_ldaOut.topics, aes(x = topics.pos_ldaOut.)) +
+   geom_bar(fill = "skyblue", color = "black") +
+   labs(title = "Count Plot for Positive Review topics", x = "Topic", y = "Count") +
+   theme_minimal() +
+   theme(
+     plot.title = element_text(hjust = 0.5, size = 15, face = "bold"),
+     axis.title.x = element_text(size = 12),
+     axis.title.y = element_text(size = 12)
+   )
>
> #Top 3 factors in Negative reviews visualized
> neg_ldaOut.topics <- data.frame(topics(neg_ldaOut))
> neg_ldaOut.topics$index <- as.numeric(row.names(neg_ldaOut.topics))
> ggplot(neg_ldaOut.topics, aes(x = topics.neg_ldaOut.))+
+   geom_bar(fill = "skyblue", color = "black") +
+   labs(title = "Count Plot for Negative Review topics", x = "Topic", y = "Count") +
+   theme_minimal() +
+   theme(
+     plot.title = element_text(hjust = 0.5, size = 15, face = "bold"),
+     axis.title.x = element_text(size = 12),
+     axis.title.y = element_text(size = 12)
+   )
>
```