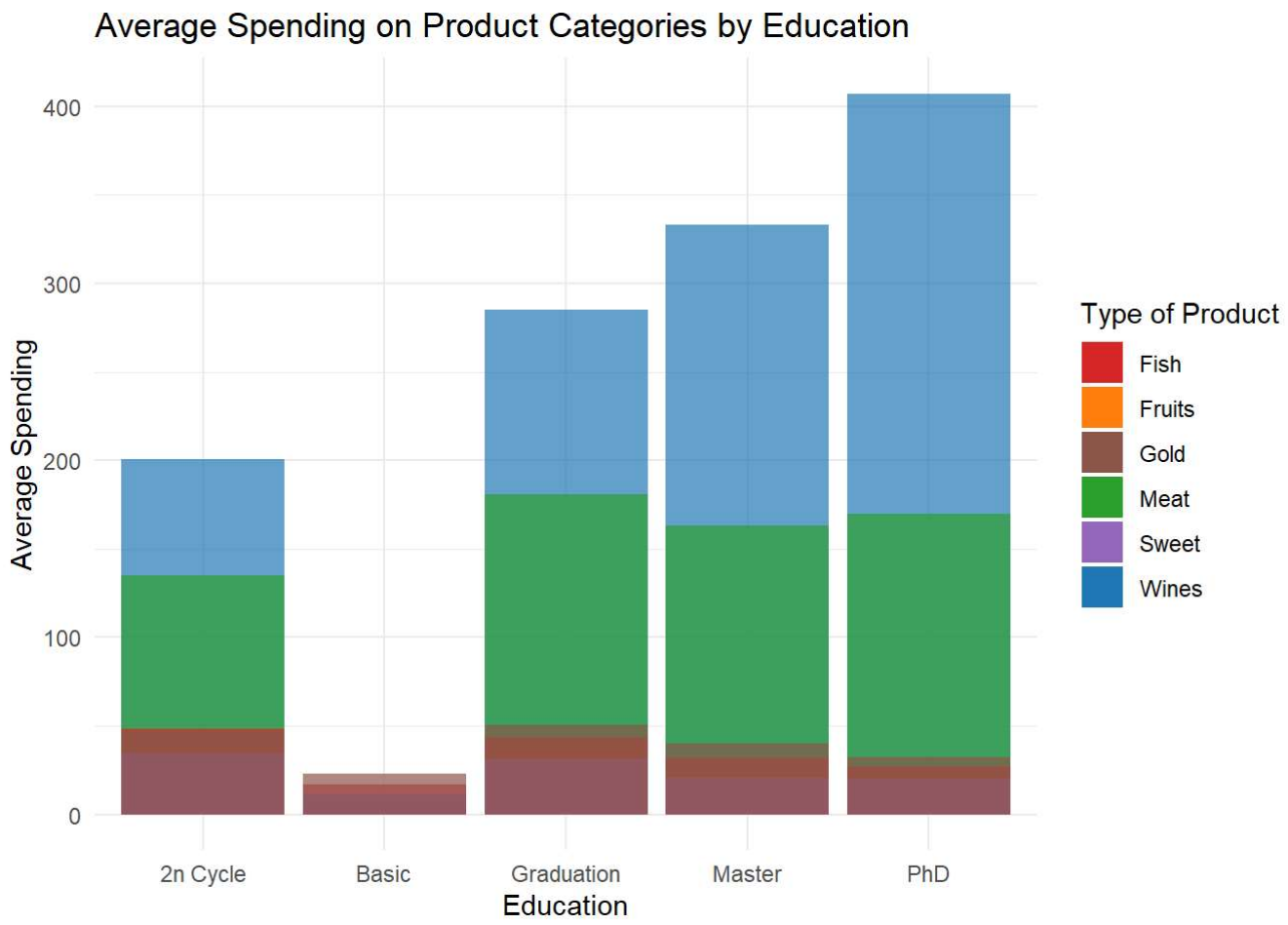


PhDs consume more wine also the fact they are older validates the the relation with age

Feature Engineering

We Create following features for Data Modelling

1. Age (already Created)
2. Total Purchase (Already Created) : Spending sum on all goods
3. Is_Parent: If customer has kids home
4. Education: Undergraduate, Graduate, Post-Graduate
5. Has_Partner: If living with someone.
6. Family Size:
7. Active Days: Number of days since enrollment to last buys.
8. Campaign: If Participated in campaign.

```
df %>%
  select(Kidhome, Teenhome)
```

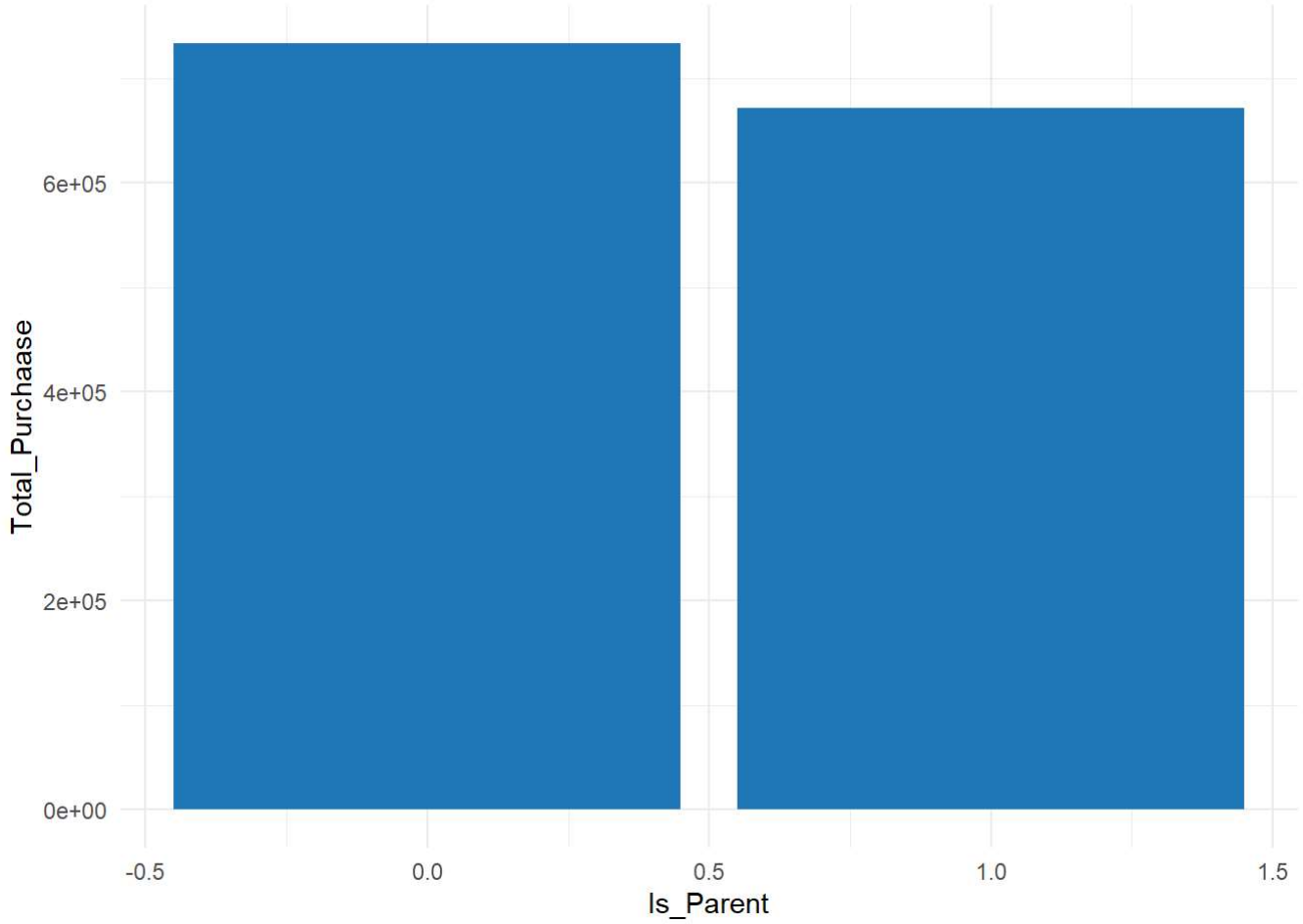
Kidhome <int>	Teenhome <int>
0	0
1	1

Kidhome <int>	Teenhome <int>
0	0
1	0
1	0
0	1
0	1
1	0
1	0
1	1

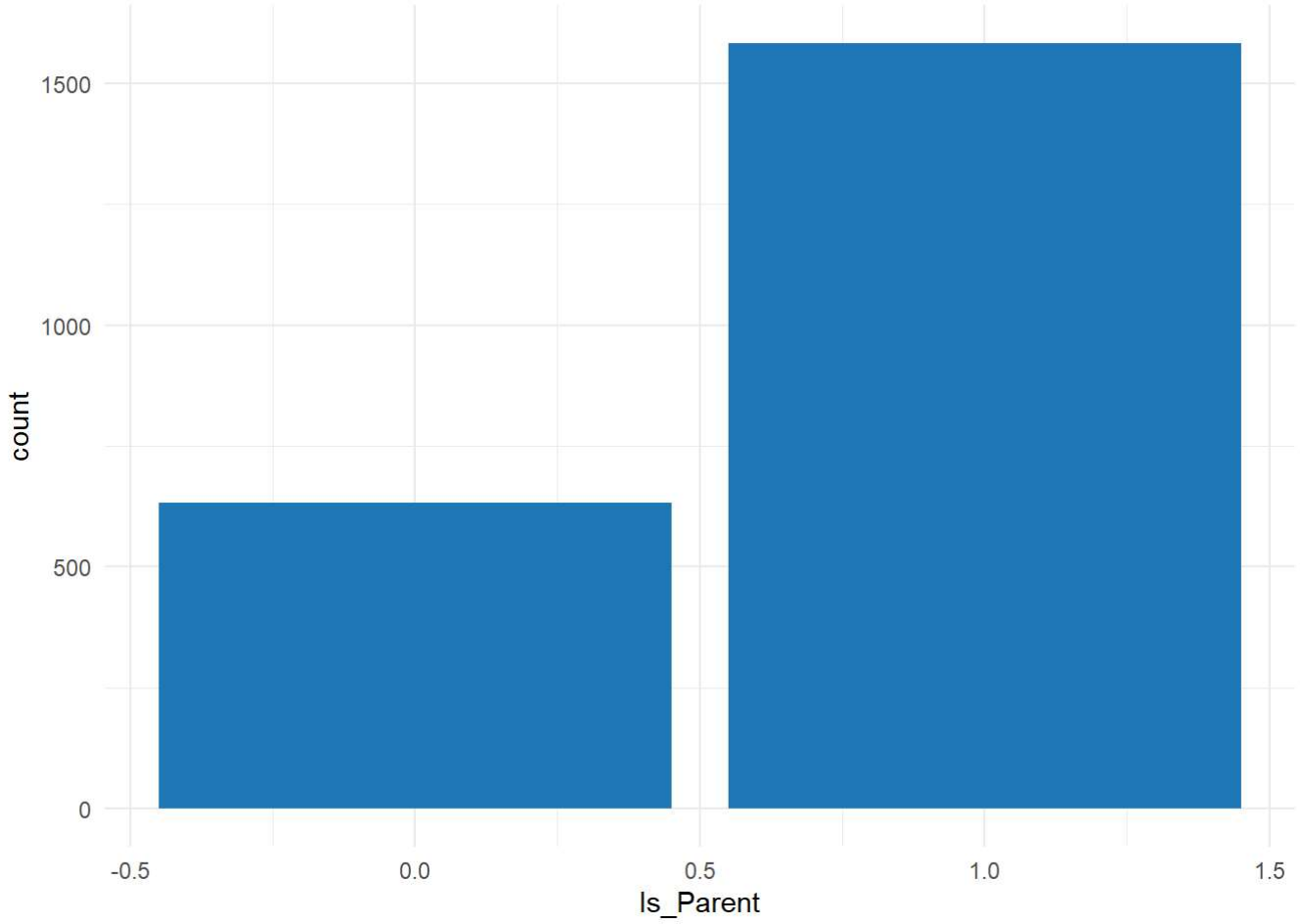
1-10 of 2,215 rows Previous **1** 2 3 4 5 6 … 222 Next

```
df <- df %>%
  mutate(Is_Parent = ifelse(Kidhome + Teenhome > 0, 1, 0))
```

```
df %>%
  ggplot(aes(x = Is_Parent, y = Total_Purchaase)) +
  geom_col(fill = "#1f77b7")+
  theme_minimal()
```



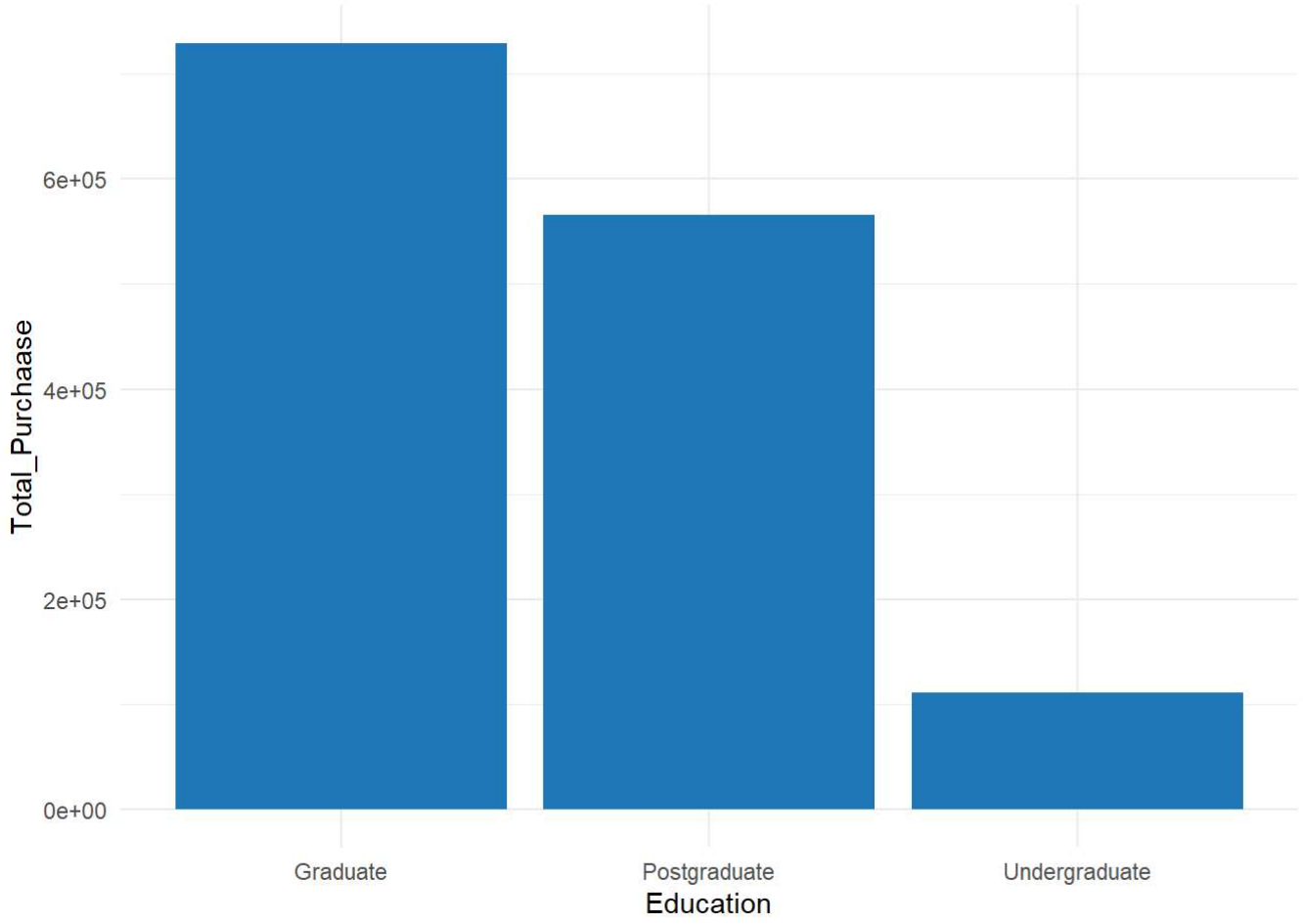

```
df %>%
  ggplot(aes(Is_Parent) ) +
  geom_bar(fill = "#1f77b7")+
  theme_minimal()
```



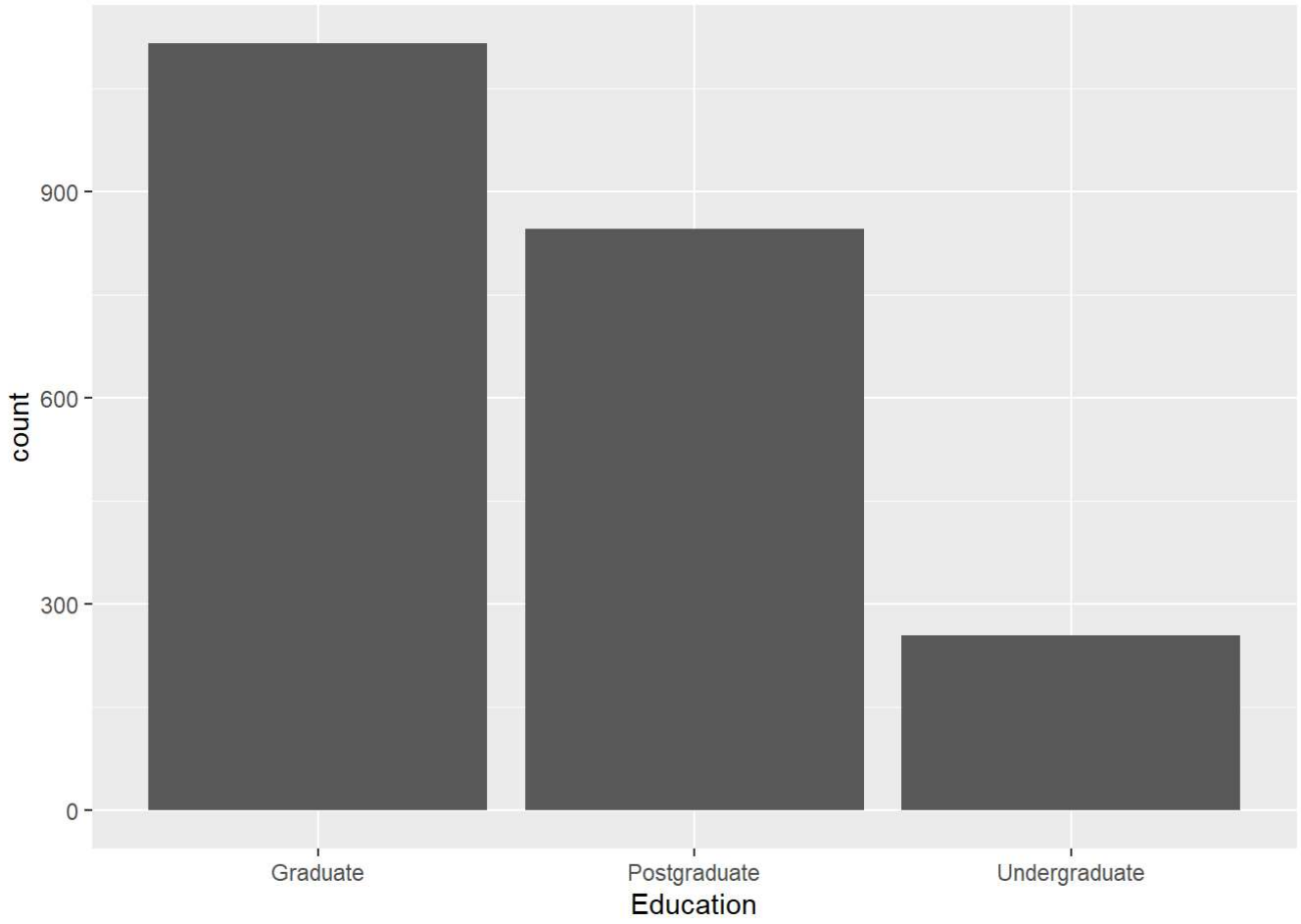
We see parents who have kids have spent relative more given then proportion in data.

```
df <- df %>%
  mutate(Education = case_when(
    Education == "Basic" ~ "Undergraduate",
    Education == "2n Cycle" ~ "Undergraduate",
    Education == "Graduation" ~ "Graduate",
    Education == "Master" ~ "Postgraduate",
    Education == "PhD" ~ "Postgraduate",
    TRUE ~ Education  # Keep the original value if none of the above conditions match
  ))
```

```
df %>%
  ggplot(aes(x = Education, y = Total_Purchaase)) +
  geom_col(fill = "#1f77b7")+
  theme_minimal()
```




```
df %>%
  ggplot(aes(x = Education)) +
  geom_bar()
```




```
df <- df %>%
  mutate(Has_Partner = case_when(
    Marital_Status %in% c("Married", "Together") ~ 1,
    Marital_Status %in% c("Absurd", "Widow", "YOLO", "Divorced", "Single", "Alone") ~ 0
  ))
```

```
df$Teenhome <- as.integer(df$Teenhome)
df$Kidhome <- as.integer(df$Kidhome)
df$Has_Partner <- as.integer(df$Has_Partner)
```

```
df <- df %>%
  mutate(Family_Size = Kidhome + Teenhome + Has_Partner)
```

```
df <- df %>%
  mutate(campaign_participation = ifelse(AcceptedCmp3 + AcceptedCmp1 + AcceptedCmp2 + AcceptedCm
p4 + AcceptedCmp5 + Response > 0, 1,0) )
```

```
features <- df %>%
  select(Age, Has_Partner, Is_Parent, Family_Size, Education, Income, Recency, campaign_particip
ation, Total_Purchaase
         )
```

```
features %>%
  head()
```

	A..	Has_Partner	Is_Parent	Family_Size	Education	Inco...	Rece...	campaign_particip
	<dbl>	<int>	<dbl>	<int>	<chr>	<int>	<int>	
1	57	0	0	0	Graduate	58138	58	
2	60	0	1	2	Graduate	46344	38	
3	49	1	0	1	Graduate	71613	26	
4	30	1	1	2	Graduate	26646	26	
5	33	1	1	2	Postgraduate	58293	94	
6	47	1	1	2	Postgraduate	62513	16	

6 rows | 1-9 of 10 columns

```
str(features)
```

```
## 'data.frame':    2215 obs. of  9 variables:
##  $ Age                  : num  57 60 49 30 33 47 43 29 40 64 ...
##  $ Has_Partner          : int  0 0 1 1 1 1 0 1 1 1 ...
##  $ Is_Parent            : num  0 1 0 1 1 1 1 1 1 1 ...
##  $ Family_Size          : int  0 2 1 2 2 2 1 2 2 3 ...
##  $ Education            : chr  "Graduate" "Graduate" "Graduate" "Graduate" ...
##  $ Income               : int  58138 46344 71613 26646 58293 62513 55635 33454 30351 5648
...
##  $ Recency              : int  58 38 26 26 94 16 34 32 19 68 ...
##  $ campaign_participation: num  1 0 0 0 0 0 0 0 1 1 ...
##  $ Total_Purchaase      : int  1705 28 797 56 449 758 639 170 49 50 ...
##  - attr(*, "na.action")= 'omit' Named int [1:24] 11 28 44 49 59 72 91 92 93 129 ...
##   ..- attr(*, "names")= chr [1:24] "11" "28" "44" "49" ...
```

```
features$Education <- as.integer(factor(features$Education, levels = c("Postgraduate","Graduat
e", "Undergraduate")))
```

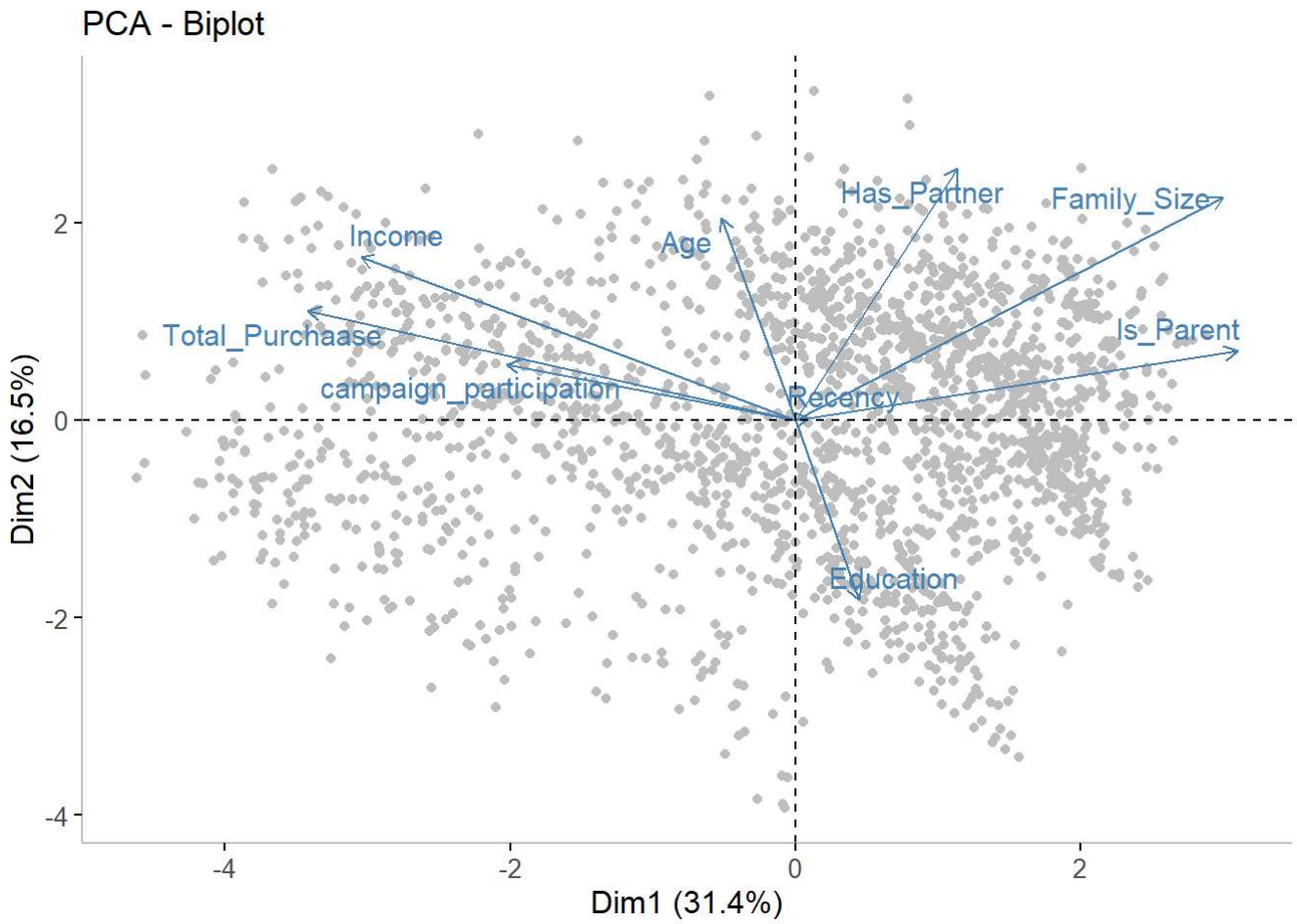
PCA

```
pca <- prcomp(features, scale = TRUE)
```

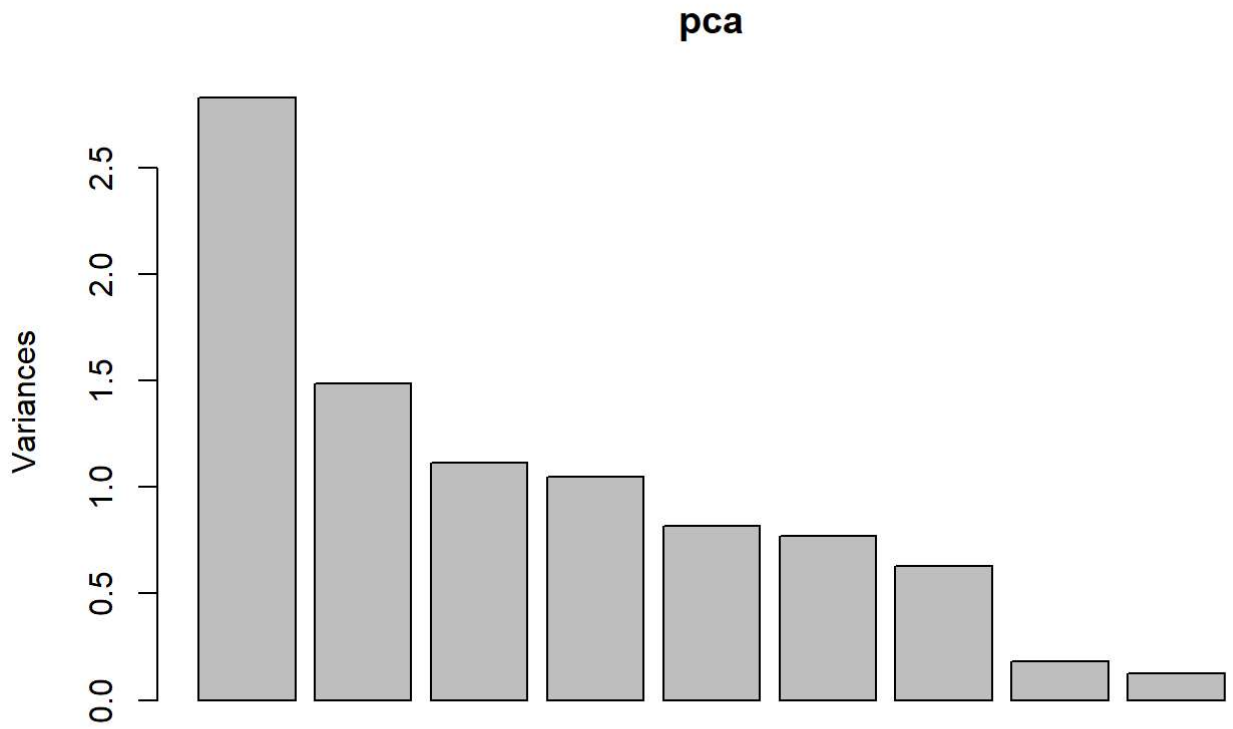
```
summary(pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4     PC5     PC6     PC7
## Standard deviation     1.6817 1.2183 1.0545 1.0237 0.90484 0.87862 0.79324
## Proportion of Variance 0.3142 0.1649 0.1235 0.1164 0.09097 0.08578 0.06991
## Cumulative Proportion  0.3142 0.4792 0.6027 0.7191 0.81010 0.89587 0.96579
##                            PC8     PC9
## Standard deviation     0.42727 0.35405
## Proportion of Variance 0.02028 0.01393
## Cumulative Proportion  0.98607 1.00000
```

```r
library(ggplot2)
library(factoextra)


# Create a biplot
biplot <- fviz_pca_biplot(pca,
                          geom.ind = "point",
                          col.ind = "grey",
                          palette = "jco",
                          repel = TRUE,
                          ggtheme = theme_classic() +
                                    theme(axis.line = element_line(colour = "grey"),
                                          axis.title = element_text(size = 12),
                                          axis.text = element_text(size = 10),
                                          panel.grid.major = element_blank(),
                                          panel.grid.minor = element_blank(),
                                          panel.border = element_blank(),
                                          panel.background = element_blank()))

# Display the biplot
biplot
```

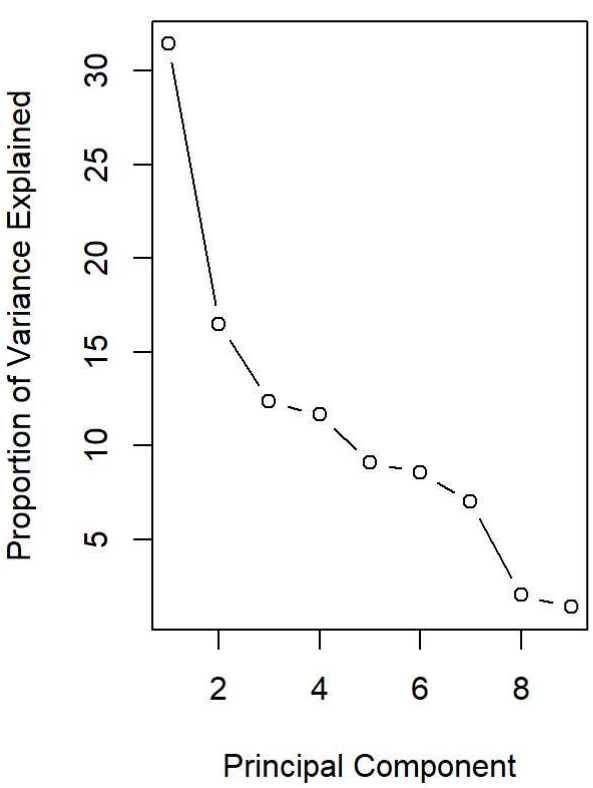


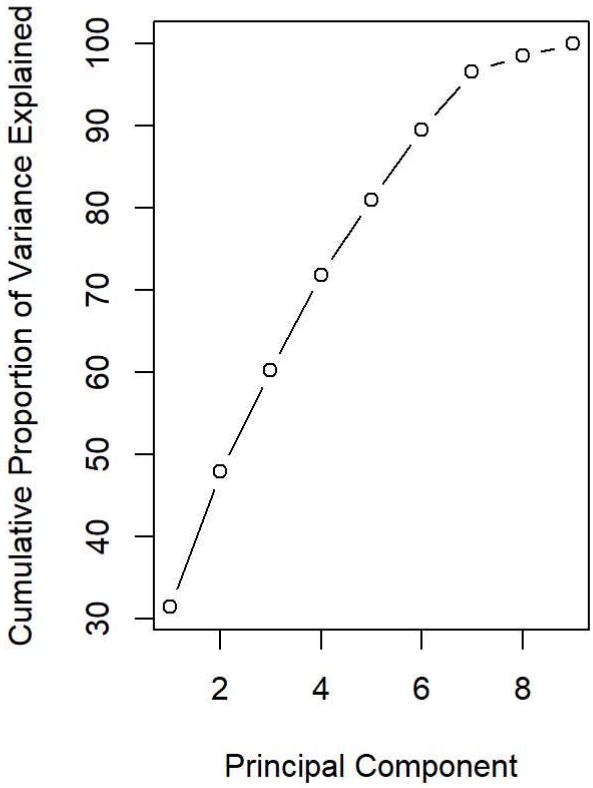

```
screeplot(pca)
```




```
pr.var <- pca$sdev^2
pve <- 100 * pr.var/ sum(pr.var)
```

```
par(mfrow = c(1, 2))
plot(pve, xlab = "Principal Component",
    ylab = "Proportion of Variance Explained",
    type = "b")
plot(cumsum(pve), xlab = "Principal Component",
    ylab = "Cumulative Proportion of Variance Explained",
     type = "b")
```

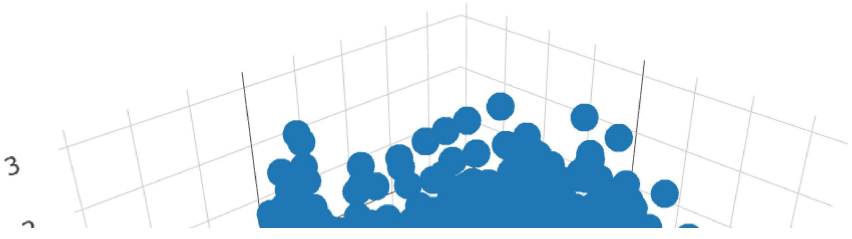


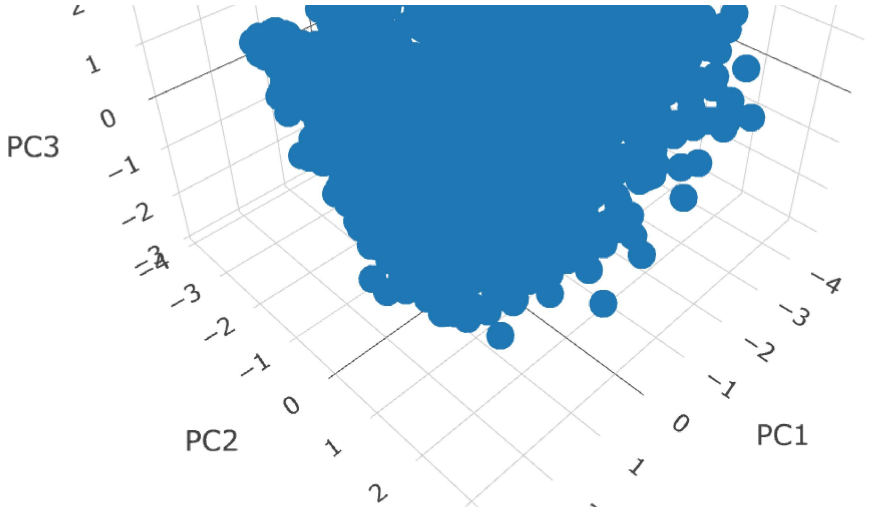


3 Principle component is good choice as it contributes to about 69% of the variation and there is an elbow point at 3,

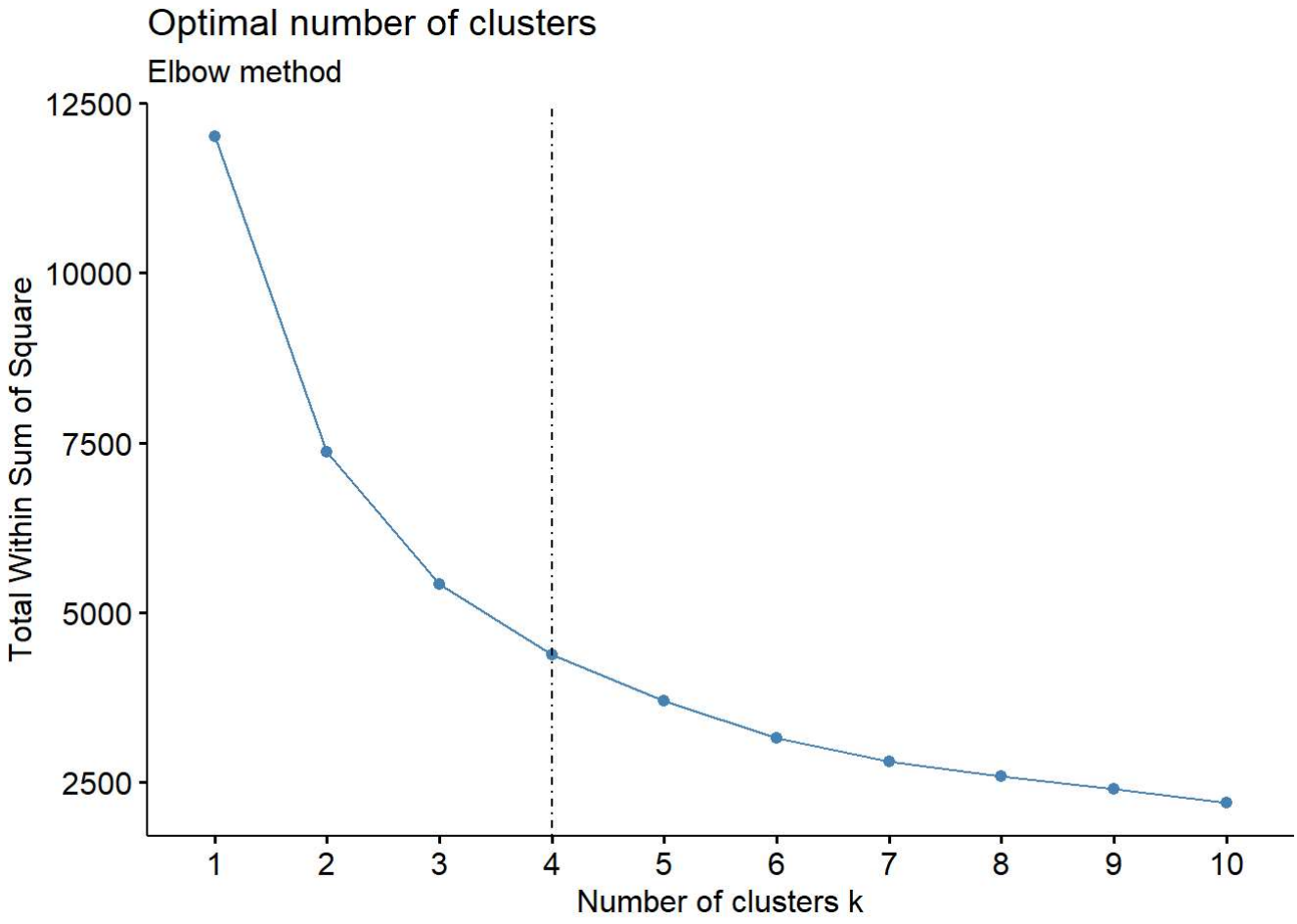
Clustering

```
library(plotly)

p <- plot_ly(x = pca$x[,1], y = pca$x[,2], z = pca$x[,3], type = "scatter3d",
             mode = "markers") %>%
  layout(scene = list(xaxis = list(title = "PC1"), yaxis = list(title = "PC2"),
                      zaxis = list(title = "PC3")))

# Display the plot
p
```






```
fviz_nbclust(pca$x[,1:3], kmeans, method = "wss",k.max=10, nstart=20, iter.max=20) +
  geom_vline(xintercept = 4, linetype = 4)+
  labs(subtitle = "Elbow method")
```



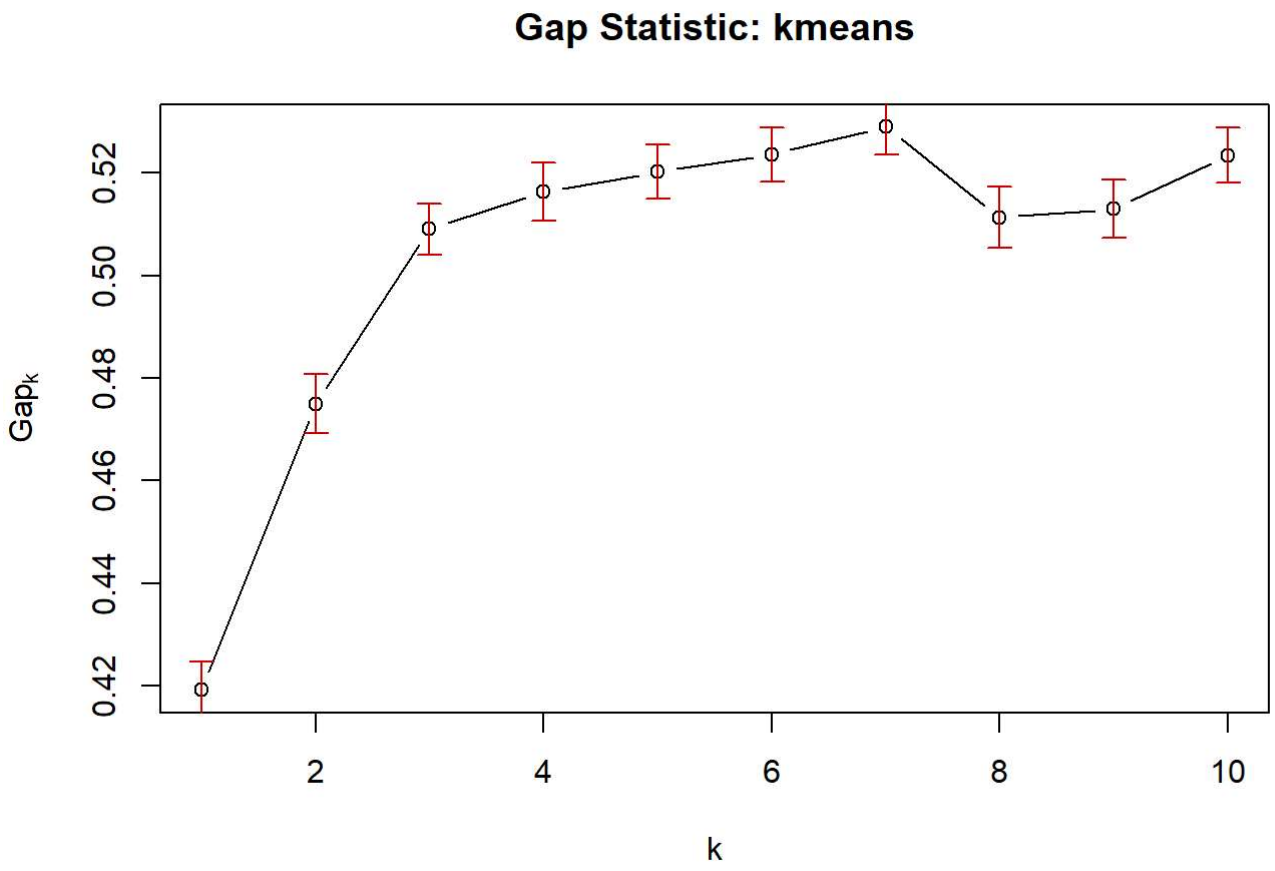

```
gap_kmeans <- clusGap(pca$x[,1:3], kmeans, nstart = 20, K.max = 10, B = 100)
```

```
## Warning: did not converge in 10 iterations

## Warning: did not converge in 10 iterations

## Warning: did not converge in 10 iterations

## Warning: did not converge in 10 iterations
```

```
## Warning: Quick-TRANSfer stage steps exceeded maximum (= 110750)
```

```
## Warning: did not converge in 10 iterations
```

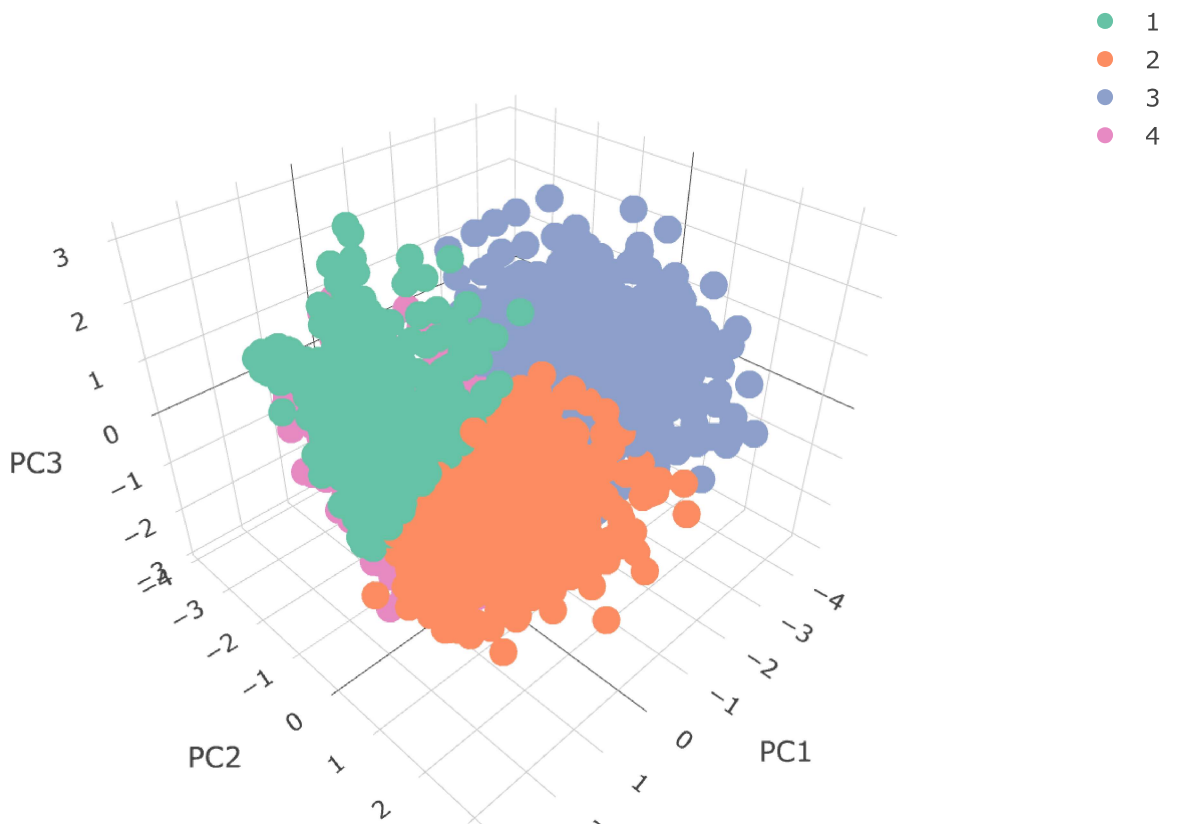
```
plot(gap_kmeans, main = "Gap Statistic: kmeans")
```



So, 4 seems like a good choice as the values post that do not add much to the curves.

```
km <- kmeans(pca$x[,1:3], 4)
```

```
# Add cluster assignment to the pca object
pca$cluster <- as.factor(km$cluster)


p <- plot_ly(x = pca$x[,1], y = pca$x[,2], z = pca$x[,3], type = "scatter3d",
              mode = "markers", color = pca$cluster) %>%
  layout(scene = list(xaxis = list(title = "PC1"), yaxis = list(title = "PC2"),
                      zaxis = list(title = "PC3")))

# Display the plot
p
```




```
df <- df %>%
  mutate(cluster = as.factor(km$cluster))
```

Profiling

```
ggplot(df, aes(x = cluster)) +
  geom_bar(fill = c("#3366CC", "#DC3912", "#FF9900", "#109618")) +
  ggtitle("Distribution Of The Clusters")+
  theme_minimal()
```