Data 622 Assignment 1

Keith DeNivo

Read in Data

```
file_url <- "https://raw.githubusercontent.com/division-zero/Data-622/refs/heads/main/assignm
# Download the file to a temporary location
temp_file <- tempfile(fileext = ".csv")
download.file(file_url, destfile = temp_file, mode = "wb")
# Read the csv file
fulldata <- read.delim(file_url, sep = ";", header = TRUE, stringsAsFactors = FALSE)
# View the data
head(fulldata)
```

```
  age       job marital   education default housing loan   contact month
1  56 housemaid married    basic.4y      no      no   no telephone   may
2  57  services married high.school unknown      no   no telephone   may
3  37  services married high.school      no     yes   no telephone   may
4  40    admin. married    basic.6y      no      no   no telephone   may
5  56  services married high.school      no      no  yes telephone   may
6  45  services married    basic.9y unknown      no   no telephone   may
  day_of_week duration campaign pdays previous    poutcome emp.var.rate
1         mon      261        1   999        0 nonexistent          1.1
2         mon      149        1   999        0 nonexistent          1.1
3         mon      226        1   999        0 nonexistent          1.1
4         mon      151        1   999        0 nonexistent          1.1
5         mon      307        1   999        0 nonexistent          1.1
6         mon      198        1   999        0 nonexistent          1.1
  cons.price.idx cons.conf.idx euribor3m nr.employed  y
1         93.994         -36.4     4.857        5191 no
2         93.994         -36.4     4.857        5191 no
```

```
3         93.994         -36.4     4.857        5191 no
4         93.994         -36.4     4.857        5191 no
5         93.994         -36.4     4.857        5191 no
6         93.994         -36.4     4.857        5191 no
```

```r
# Clean up the temporary file
unlink(temp_file)
```

```r
#smaller data set for training or testing
file_url <- "https://raw.githubusercontent.com/division-zero/Data-622/refs/heads/main/assignm
# Download the file to a temporary location
temp_file <- tempfile(fileext = ".csv")
download.file(file_url, destfile = temp_file, mode = "wb")
# Read the csv file
partdata <- read.delim(file_url, sep = ";", header = TRUE, stringsAsFactors = FALSE)
# View the data
head(partdata)
```

```
  age         job marital         education default housing    loan   contact
1  30 blue-collar married          basic.9y      no     yes      no  cellular
2  39    services  single       high.school      no      no      no telephone
3  25    services married       high.school      no     yes      no telephone
4  38    services married          basic.9y      no unknown unknown telephone
5  47      admin. married university.degree      no     yes      no  cellular
6  32    services  single university.degree      no      no      no  cellular
  month day_of_week duration campaign pdays previous    poutcome emp.var.rate
1   may         fri      487        2   999        0 nonexistent         -1.8
2   may         fri      346        4   999        0 nonexistent          1.1
3   jun         wed      227        1   999        0 nonexistent          1.4
4   jun         fri       17        3   999        0 nonexistent          1.4
5   nov         mon       58        1   999        0 nonexistent         -0.1
6   sep         thu      128        3   999        2     failure         -1.1
  cons.price.idx cons.conf.idx euribor3m nr.employed  y
1         92.893         -46.2     1.313      5099.1 no
2         93.994         -36.4     4.855      5191.0 no
3         94.465         -41.8     4.962      5228.1 no
4         94.465         -41.8     4.959      5228.1 no
5         93.200         -42.0     4.191      5195.8 no
6         94.199         -37.5     0.884      4963.6 no
```

```
# Clean up the temporary file
unlink(temp_file)
```

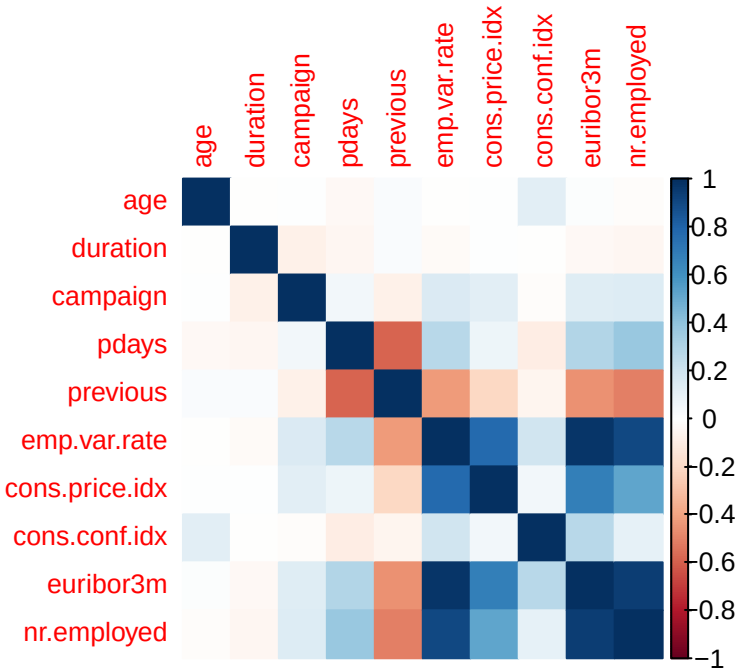
```
print(colSums(is.na(fulldata)))
```

```
          age           job       marital     education       default
            0             0             0             0             0
      housing          loan       contact         month   day_of_week
            0             0             0             0             0
     duration      campaign         pdays      previous      poutcome
            0             0             0             0             0
 emp.var.rate cons.price.idx  cons.conf.idx     euribor3m   nr.employed
            0             0             0             0             0
            y
            0
```

data does not have missing values.

Are the features (columns) of your data correlated?

```
numeric_df <- fulldata[sapply(fulldata, is.numeric)]
cor_matrix <- cor(numeric_df, use = "complete.obs")
corrplot(cor_matrix, method = "color", tl.cex = 0.8)
```



```r
cor_df <- as.data.frame(as.table(cor_matrix)) #put the correlations into a dataframe


names(cor_df) <- c("feature_1", "feature_2", "correlation") #name the columns of the correlat


cor_df <- cor_df[cor_df$feature_1 != cor_df$feature_2, ] #remove self correlations


cor_df <- cor_df[!duplicated(t(apply(cor_df[, 1:2], 1, sort))), ]
#remove the redundant pairs


cor_df <- cor_df[order(abs(cor_df$correlation), decreasing = TRUE), ]
# sort by absolute correlation


head(cor_df, 10)
```

```
        feature_1      feature_2 correlation
59      euribor3m   emp.var.rate   0.9722447
90    nr.employed      euribor3m   0.9451544
60    nr.employed   emp.var.rate   0.9069701
57 cons.price.idx   emp.var.rate   0.7753342
69      euribor3m cons.price.idx   0.6882301
35       previous          pdays  -0.5875139
70    nr.employed cons.price.idx   0.5220340
50    nr.employed       previous  -0.5013329
49      euribor3m       previous  -0.4544937
46   emp.var.rate       previous  -0.4204891
```

Most correlated numeric features:

Positive Correlation:

consumer price index, employment variation rate

number of employees, euribor 3 month rate (interest rates)

number of employees employment variation rate

negative:

previous(number of contacts performed before this campaign and for this client) pdays (number of days that passed by after the client was last contacted from a previous campaign)

previous, number of employees

```
cat_cols <- sapply(fulldata, function(x) !is.numeric(x))
head(cat_cols)
```

```
      age       job   marital education   default   housing
    FALSE      TRUE      TRUE      TRUE      TRUE      TRUE
```

```
#cat_cols <- names(cat_cols)
head(cat_cols)
```

```
      age       job   marital education   default   housing
    FALSE      TRUE      TRUE      TRUE      TRUE      TRUE
```

```
catdata <- fulldata[, cat_cols]
head(catdata)
```

```
        job marital   education default housing loan   contact month
1 housemaid married    basic.4y      no      no   no telephone   may
2  services married high.school unknown      no   no telephone   may
3  services married high.school      no     yes   no telephone   may
4    admin. married    basic.6y      no      no   no telephone   may
5  services married high.school      no      no  yes telephone   may
6  services married    basic.9y unknown      no   no telephone   may
  day_of_week    poutcome  y
1         mon nonexistent no
2         mon nonexistent no
3         mon nonexistent no
4         mon nonexistent no
5         mon nonexistent no
6         mon nonexistent no
```

```
combos <- combn(names(catdata), 2, simplify = FALSE)

#  Cramers V for comparing features
cramer_v <- lapply(combos, function(x) {
  v <- cramerV(catdata[[x[1]]], catdata[[x[2]]], bias.correct = TRUE)
```

```
  data.frame(var1 = x[1], var2 = x[2], cramers_v = v)
}) |>  bind_rows()


cramer_v <- cramer_v[order(abs(cramer_v$cramers_v), decreasing = TRUE), ]

head(cramer_v, 10)
```

```
                  var1      var2 cramers_v
Cramer V...35   housing      loan    0.7079
Cramer V...46   contact     month    0.6091
Cramer V...2        job education    0.3595
Cramer V...55  poutcome         y    0.3204
Cramer V...52     month         y    0.2740
Cramer V...48   contact  poutcome    0.2424
Cramer V...51     month  poutcome    0.2424
Cramer V...1        job   marital    0.1836
Cramer V...20 education   default    0.1704
Cramer V...3        job   default    0.1521
```

categorical variables:

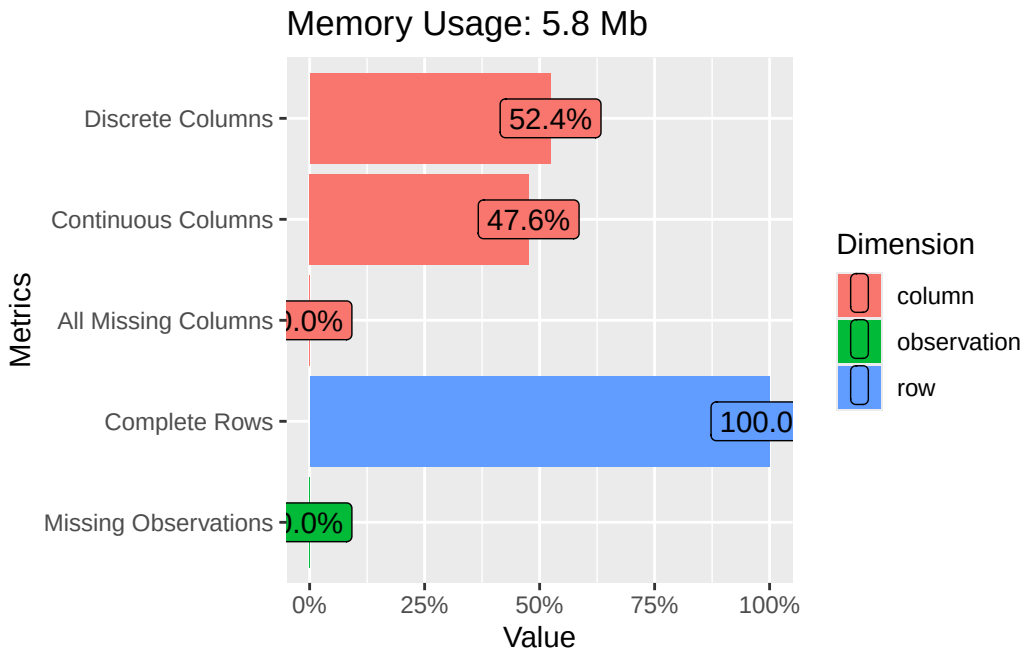
housing and loan, contact and month, job and education highly correlated

term deposit somewhat correlated with poutcome, and month

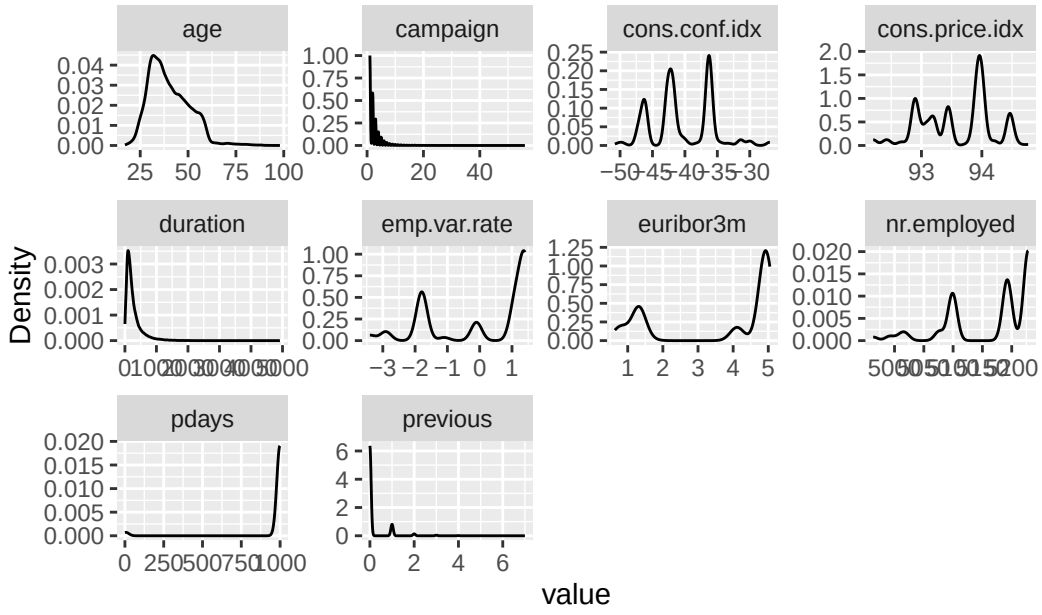
What is the overall distribution of each variable?

Distribution, central tendency and spread

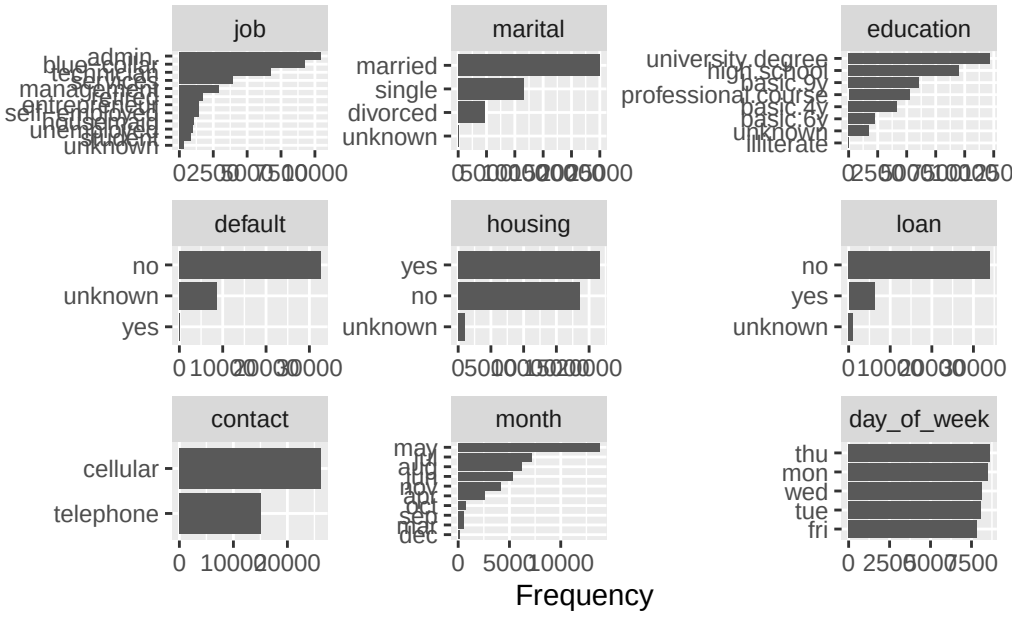
```
plot_intro(fulldata)         # summary info
```



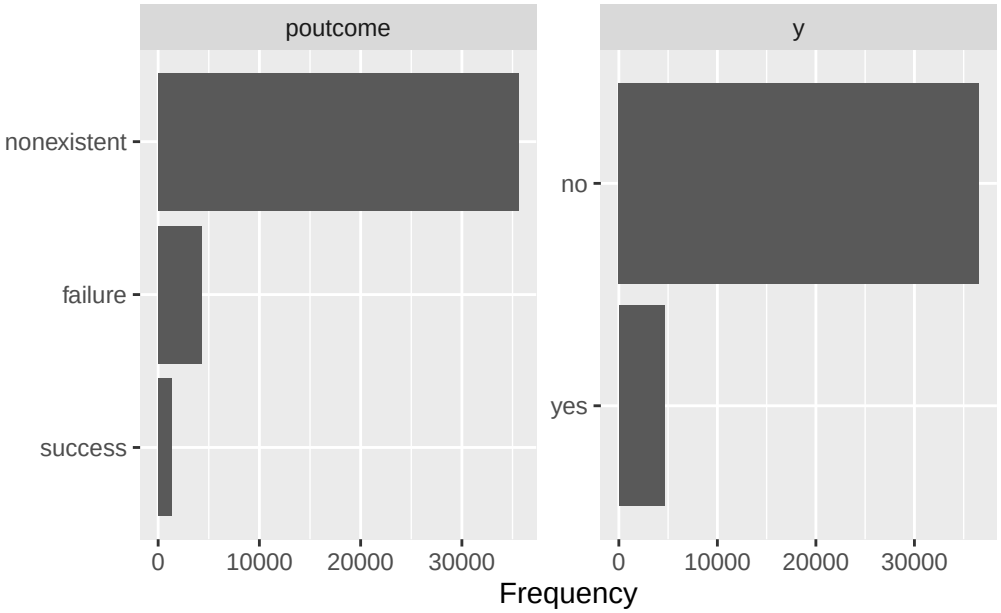

```
plot_density(fulldata)        # numeric distributions
```



```
plot_bar(fulldata)          # categorical distributions
```



Page 1



Page 2

```
summary(fulldata)
```

```
      age            job              marital           education
 Min.   :17.00   Length:41188       Length:41188       Length:41188
 1st Qu.:32.00   Class :character   Class :character   Class :character
 Median :38.00   Mode  :character   Mode  :character   Mode  :character
 Mean   :40.02
 3rd Qu.:47.00
 Max.   :98.00
   default            housing              loan             contact
 Length:41188       Length:41188       Length:41188       Length:41188
 Class :character   Class :character   Class :character   Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character


    month           day_of_week           duration         campaign
 Length:41188       Length:41188       Min.   :   0.0   Min.   : 1.000
 Class :character   Class :character   1st Qu.: 102.0   1st Qu.: 1.000
 Mode  :character   Mode  :character   Median : 180.0   Median : 2.000
                                       Mean   : 258.3   Mean   : 2.568
                                       3rd Qu.: 319.0   3rd Qu.: 3.000
                                       Max.   :4918.0   Max.   :56.000
     pdays          previous       poutcome          emp.var.rate
 Min.   :  0.0   Min.   :0.000   Length:41188       Min.   :-3.40000
 1st Qu.:999.0   1st Qu.:0.000   Class :character   1st Qu.:-1.80000
 Median :999.0   Median :0.000   Mode  :character   Median : 1.10000
 Mean   :962.5   Mean   :0.173                      Mean   : 0.08189
 3rd Qu.:999.0   3rd Qu.:0.000                      3rd Qu.: 1.40000
 Max.   :999.0   Max.   :7.000                      Max.   : 1.40000
 cons.price.idx  cons.conf.idx     euribor3m      nr.employed
 Min.   :92.20   Min.   :-50.8   Min.   :0.634   Min.   :4964
 1st Qu.:93.08   1st Qu.:-42.7   1st Qu.:1.344   1st Qu.:5099
 Median :93.75   Median :-41.8   Median :4.857   Median :5191
 Mean   :93.58   Mean   :-40.5   Mean   :3.621   Mean   :5167
 3rd Qu.:93.99   3rd Qu.:-36.4   3rd Qu.:4.961   3rd Qu.:5228
 Max.   :94.77   Max.   :-26.9   Max.   :5.045   Max.   :5228
      y
 Length:41188
 Class :character
 Mode  :character
```

```
describe(numeric_df)
```

```
               vars     n    mean     sd  median trimmed    mad     min     max
age               1 41188   40.02  10.42   38.00   39.30  10.38   17.00   98.00
duration          2 41188  258.29 259.28  180.00  210.61 139.36    0.00 4918.00
campaign          3 41188    2.57   2.77    2.00    1.99   1.48    1.00   56.00
pdays             4 41188  962.48 186.91  999.00  999.00   0.00    0.00  999.00
previous          5 41188    0.17   0.49    0.00    0.05   0.00    0.00    7.00
emp.var.rate      6 41188    0.08   1.57    1.10    0.27   0.44   -3.40    1.40
cons.price.idx    7 41188   93.58   0.58   93.75   93.58   0.56   92.20   94.77
cons.conf.idx     8 41188  -40.50   4.63  -41.80  -40.60   6.52  -50.80  -26.90
euribor3m         9 41188    3.62   1.73    4.86    3.81   0.16    0.63    5.04
nr.employed      10 41188 5167.04  72.25 5191.00 5178.43  55.00 4963.60 5228.10
                 range  skew kurtosis   se
age              81.00  0.78     0.79 0.05
duration       4918.00  3.26    20.24 1.28
campaign         55.00  4.76    36.97 0.01
pdays           999.00 -4.92    22.23 0.92
previous          7.00  3.83    20.11 0.00
emp.var.rate      4.80 -0.72    -1.06 0.01
cons.price.idx    2.57 -0.23    -0.83 0.00
cons.conf.idx    23.90  0.30    -0.36 0.02
euribor3m         4.41 -0.71    -1.41 0.01
nr.employed     264.50 -1.04     0.00 0.36
```

```
skim(fulldata)
```

Table 1: Data summary

Name	fulldata
Number of rows	41188
Number of columns	21
_____	
Column type frequency:	
character	11
numeric	10
_____	
Group variables	None

Variable type: character

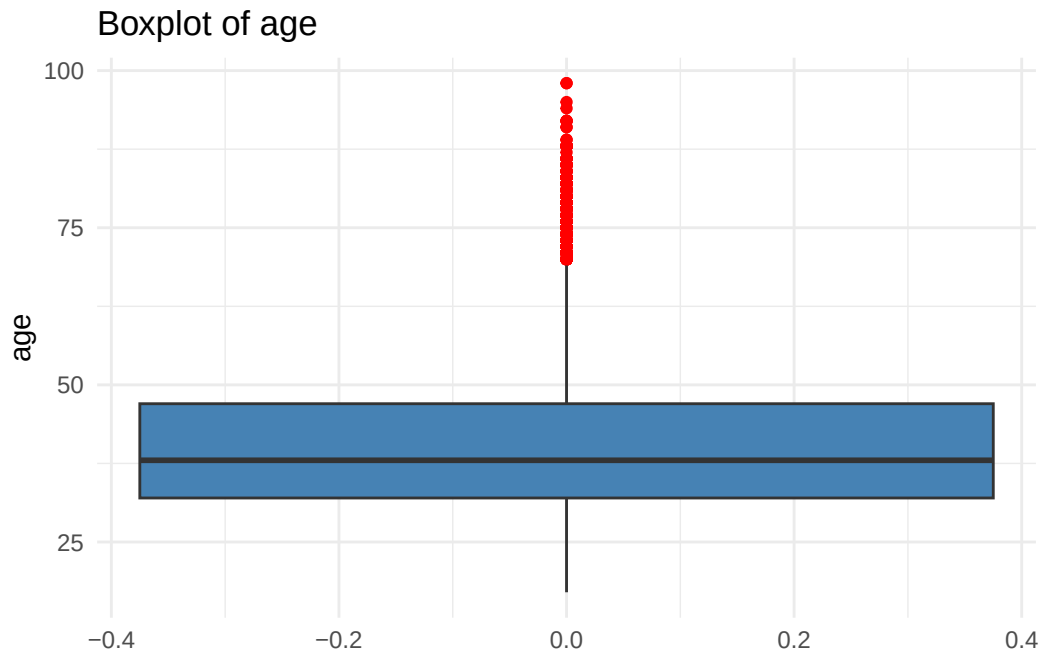
skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
job	0	1	6	13	0	12	0
marital	0	1	6	8	0	4	0
education	0	1	7	19	0	8	0
default	0	1	2	7	0	3	0
housing	0	1	2	7	0	3	0
loan	0	1	2	7	0	3	0
contact	0	1	8	9	0	2	0
month	0	1	3	3	0	10	0
day_of_week	0	1	3	3	0	5	0
poutcome	0	1	7	11	0	3	0
y	0	1	2	3	0	2	0

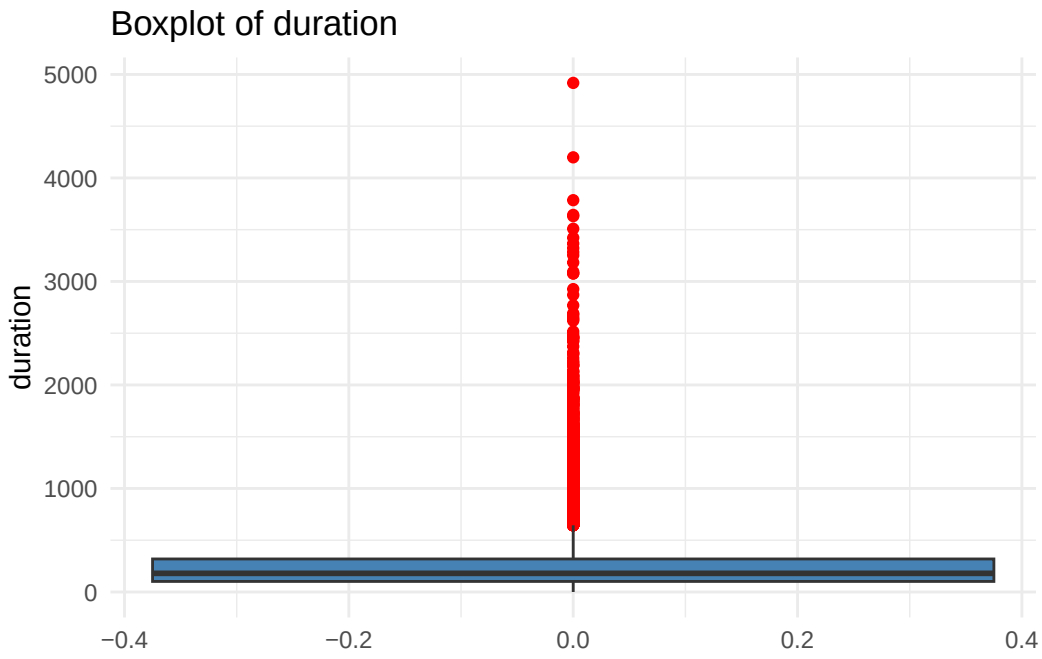
Variable type: numeric

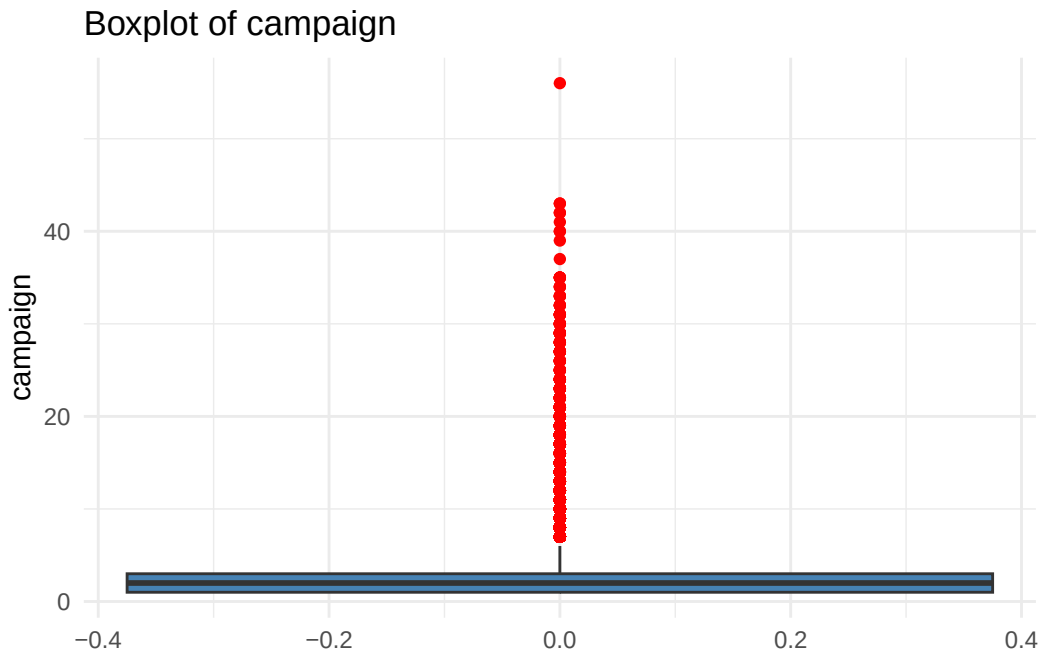
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
age	0	1	40.02	10.42	17.00	32.00	38.00	47.00	98.00	
duration	0	1	258.29	259.28	0.00	102.00	180.00	319.00	4918.00	
campaign	0	1	2.57	2.77	1.00	1.00	2.00	3.00	56.00	
pdays	0	1	962.48	186.91	0.00	999.00	999.00	999.00	999.00	
previous	0	1	0.17	0.49	0.00	0.00	0.00	0.00	7.00	
emp.var.rate	0	1	0.08	1.57	-3.40	-1.80	1.10	1.40	1.40	
cons.price.idx	0	1	93.58	0.58	92.20	93.08	93.75	93.99	94.77	
cons.conf.idx	0	1	-	4.63	-	-	-	-	-	
			40.50		50.80	42.70	41.80	36.40	26.90	
euribor3m	0	1	3.62	1.73	0.63	1.34	4.86	4.96	5.04	
nr.employed	0	1	5167.04	72.25	4963.60	5099.10	5191.00	5228.10	5228.10	

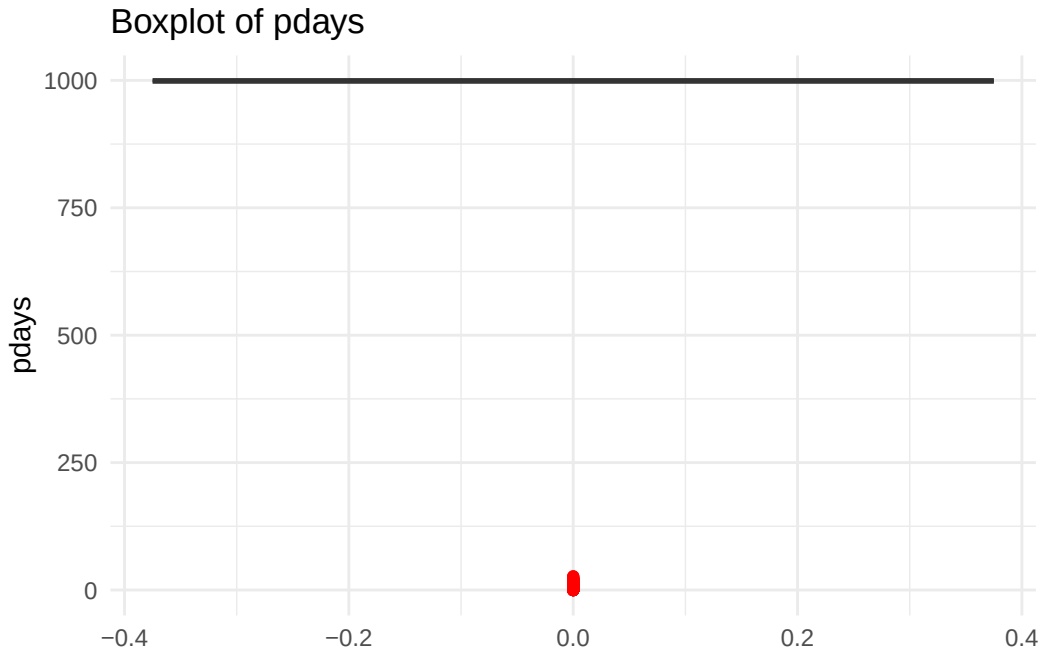
Are there any outliers present?

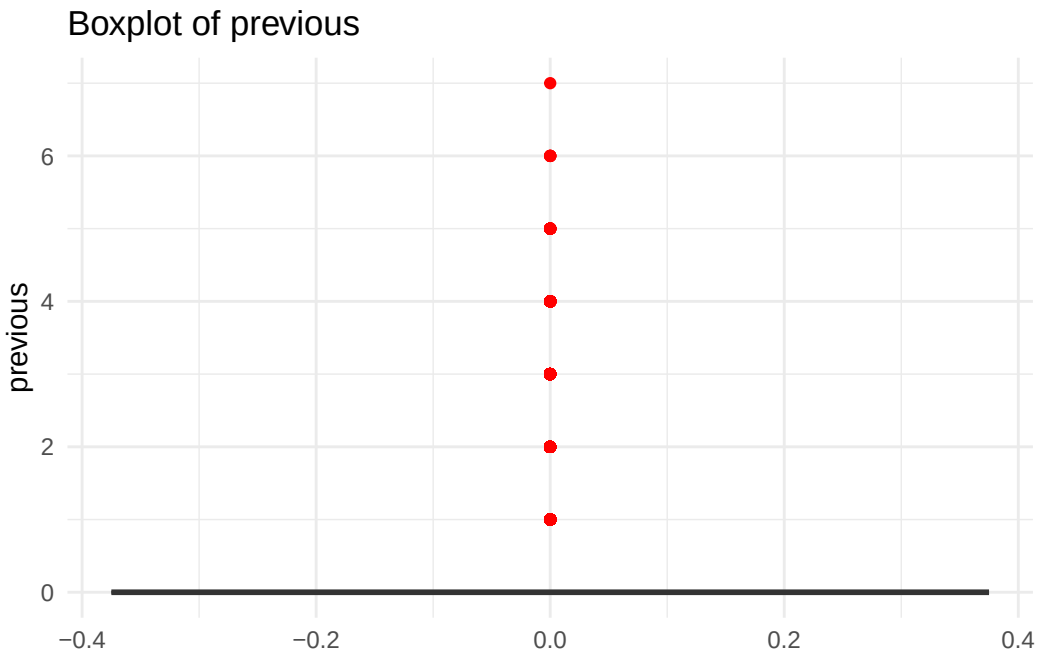
```
for (col in names(numeric_df)) {
  p <- ggplot(numeric_df, aes(y = .data[[col]])) +
    geom_boxplot(fill = "steelblue", outlier.color = "red") +
    labs(title = paste("Boxplot of", col), y = col) +
    theme_minimal()

  print(p)  # prints each plot in the viewer
}
```

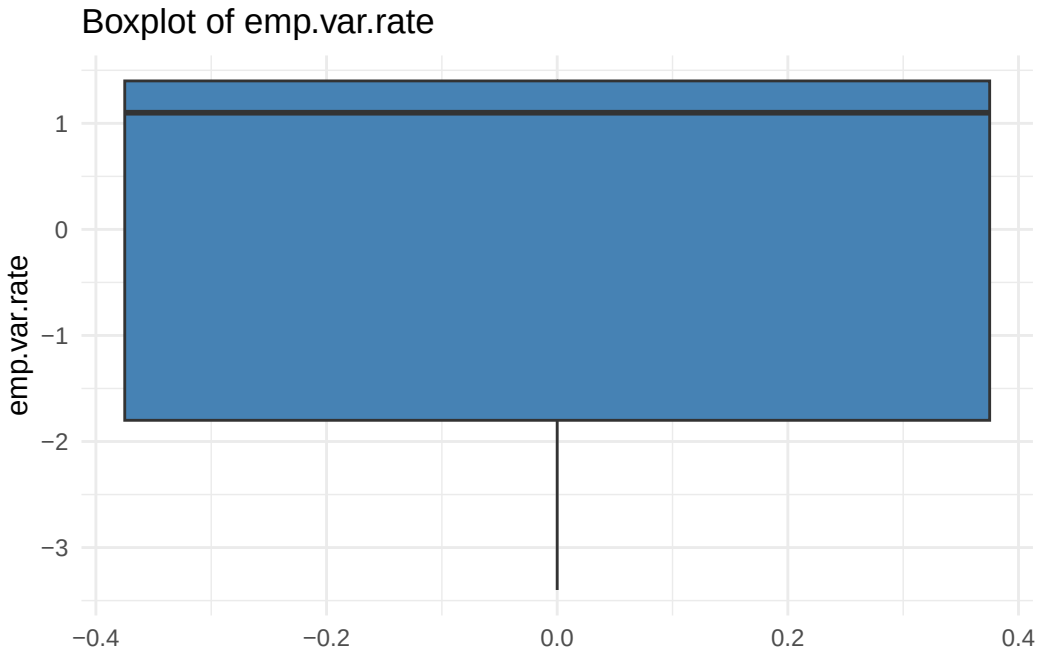

Boxplot of age
age
100
75
50
25
−0.4
−0.2
0.0
0.2
0.4

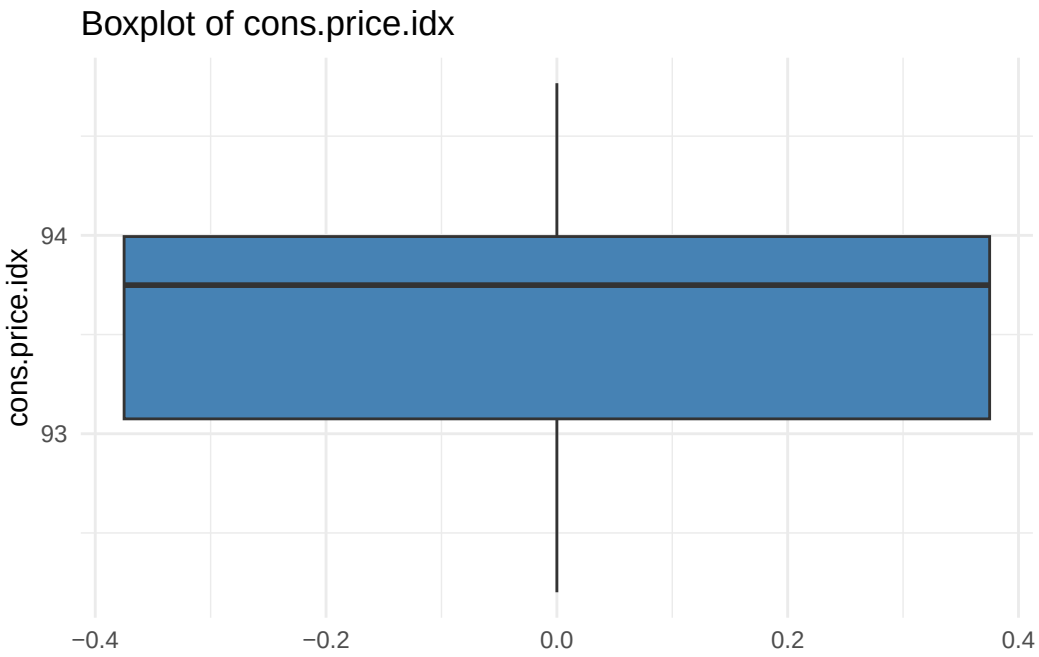

Boxplot of duration
duration
5000
4000
3000
2000
1000
0
−0.4
−0.2
0.0
0.2
0.4

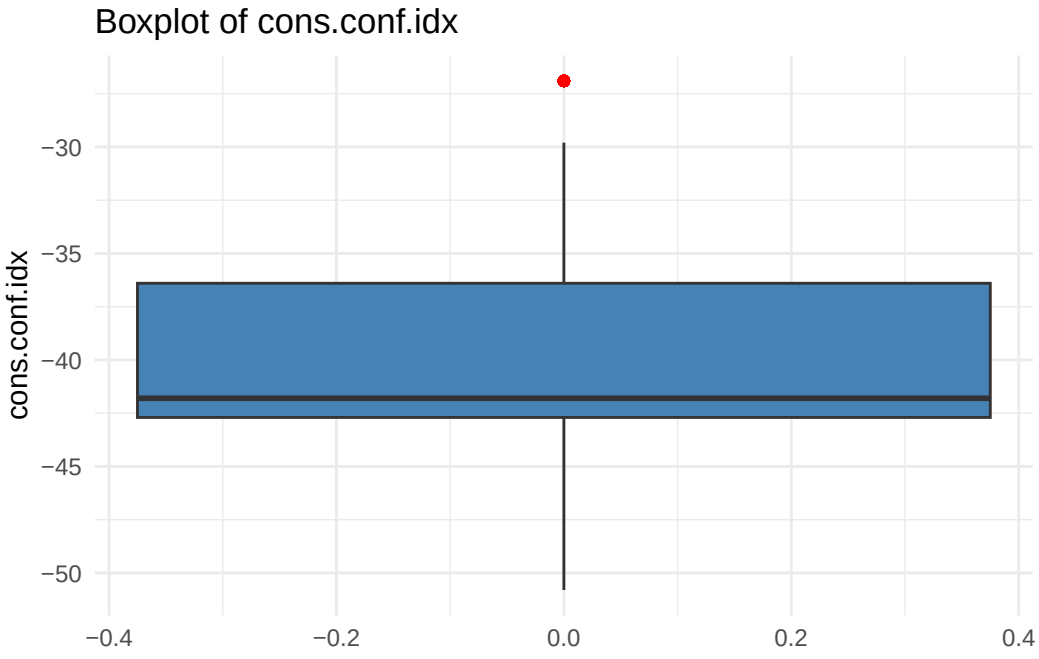

Boxplot of campaign
campaign
40
20
0
−0.4
−0.2
0.0
0.2
0.4

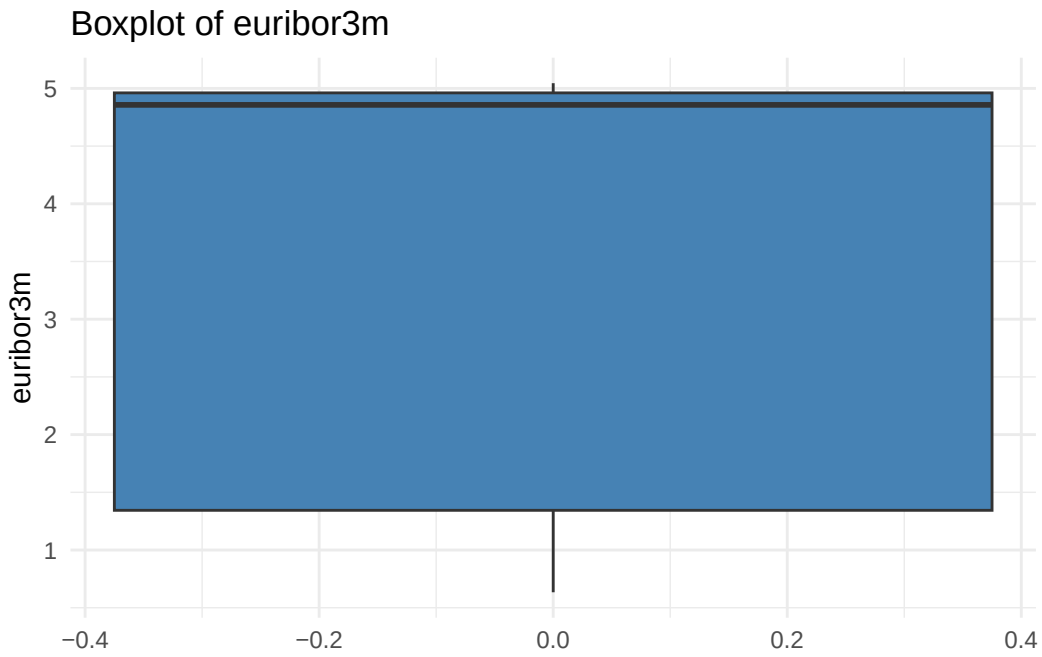

Boxplot of pdays
pdays
1000
750
500
250
0
−0.4
−0.2
0.0
0.2
0.4

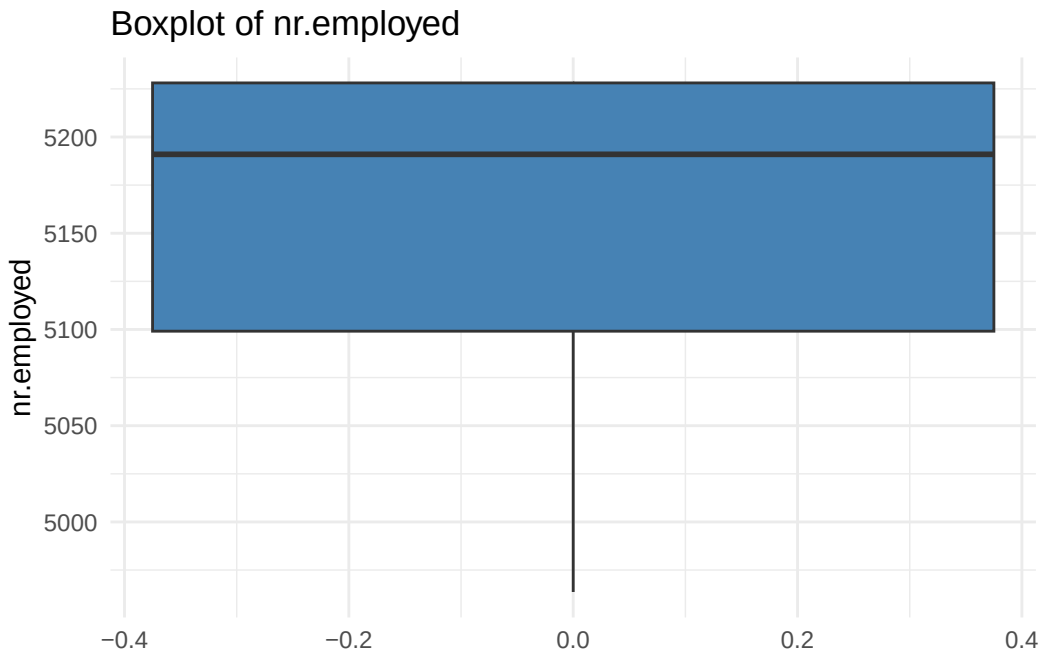

Boxplot of previous
previous
6
4
2
0
−0.4
−0.2
0.0
0.2
0.4


Boxplot of emp.var.rate
emp.var.rate
1
0
−1
−2
−3
−0.4
−0.2
0.0
0.2
0.4


Boxplot of cons.price.idx
cons.price.idx
94
93
−0.4
−0.2
0.0
0.2
0.4

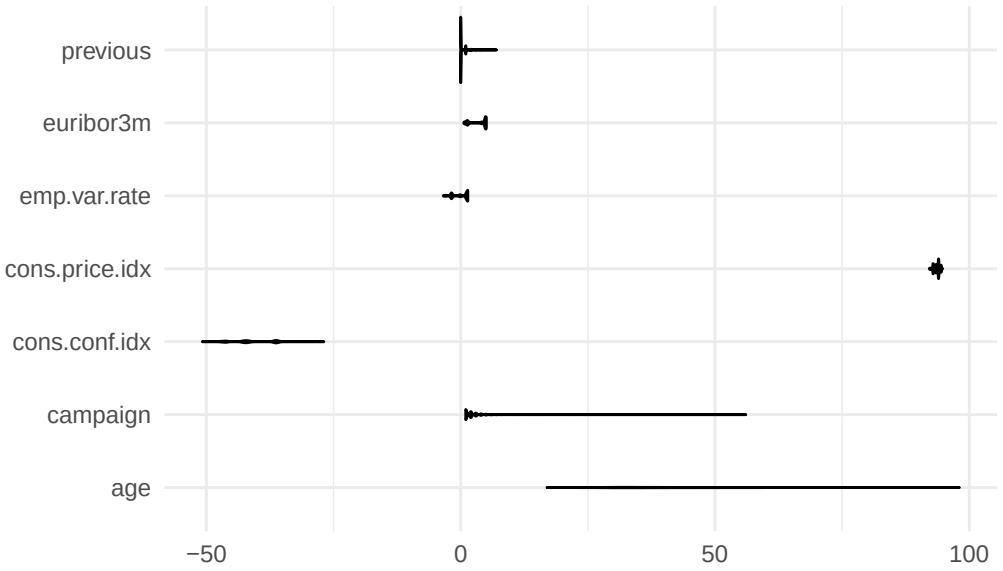

Boxplot of cons.conf.idx
cons.conf.idx
−30
−35
−40
−45
−50
−0.4
−0.2
0.0
0.2
0.4



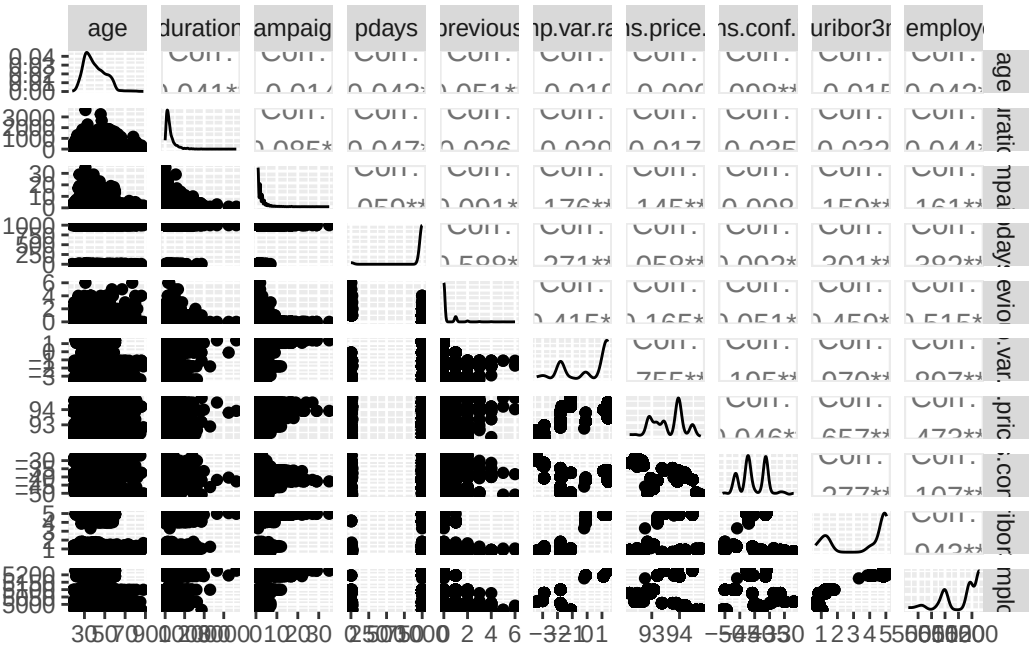



```
# Convert to long format for ggplot
numeric_long <- numeric_df  |> dplyr::select(-nr.employed, -duration,-pdays) |>
  pivot_longer(cols = everything(), names_to = "Variable", values_to = "Value")
```

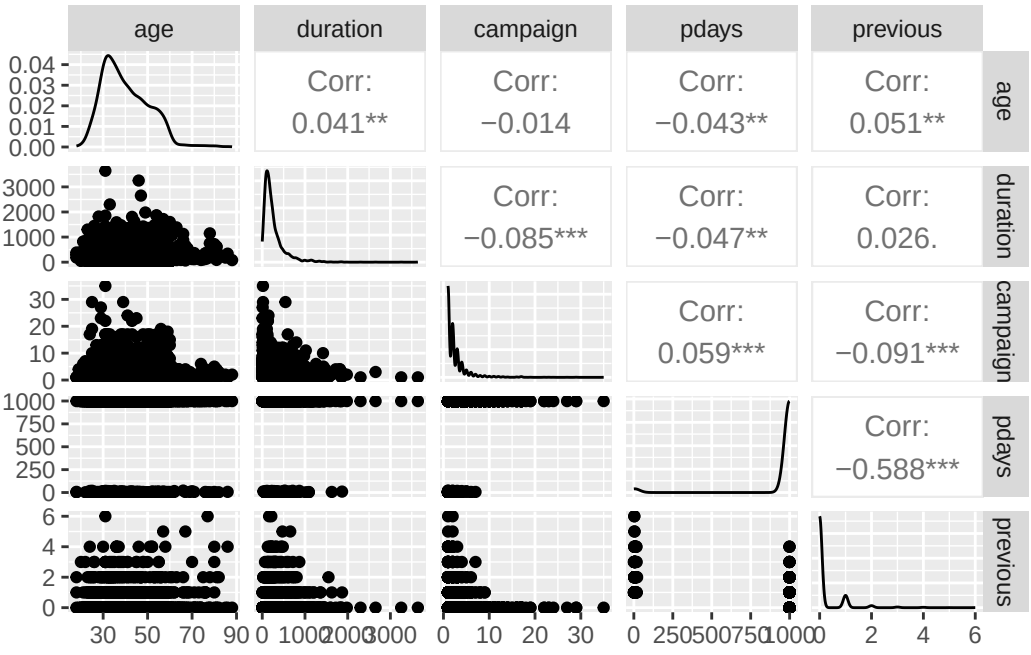
```
ggplot(numeric_long, aes(x = Variable, y = Value)) +
geom_violin(fill = "skyblue", color = "black") +
  #scale_y_continuous(limits = c(0, 500)) +
labs(title = "",
x = "",
y = "") +
theme_minimal()+
  coord_flip()
```



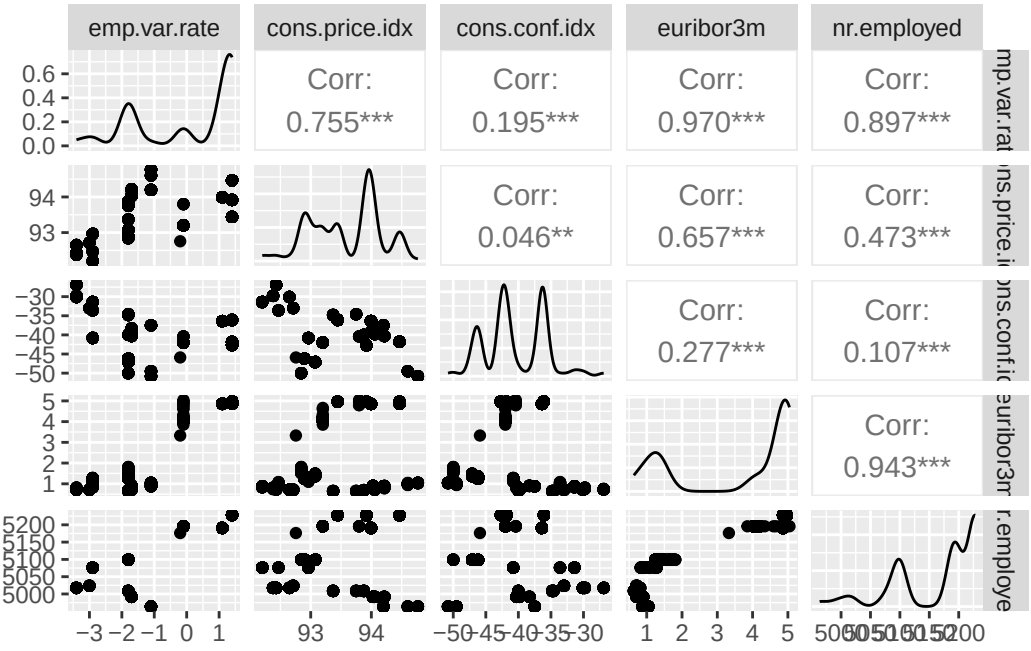

```
numpartdata <- partdata |> dplyr::select(where(is.numeric))
firsthalf <- numpartdata[ ,1:5]
secondhalf <- numpartdata[ ,6:10]

ggpairs(numpartdata)
```




```
ggpairs(firsthalf)
```



```
ggpairs(secondhalf)
```




```
unknown_counts <- sapply(fulldata, function(x) sum(x == "unknown", na.rm = TRUE))

# View the result
unknown_counts
```

```
          age           job       marital     education       default 
            0           330            80          1731          8597 
      housing          loan       contact         month   day_of_week 
          990           990             0             0             0 
     duration      campaign         pdays      previous      poutcome 
            0             0             0             0             0 
  emp.var.rate cons.price.idx  cons.conf.idx     euribor3m   nr.employed 
            0             0             0             0             0 
            y 
            0 
```

```
nrow(fulldata)
```

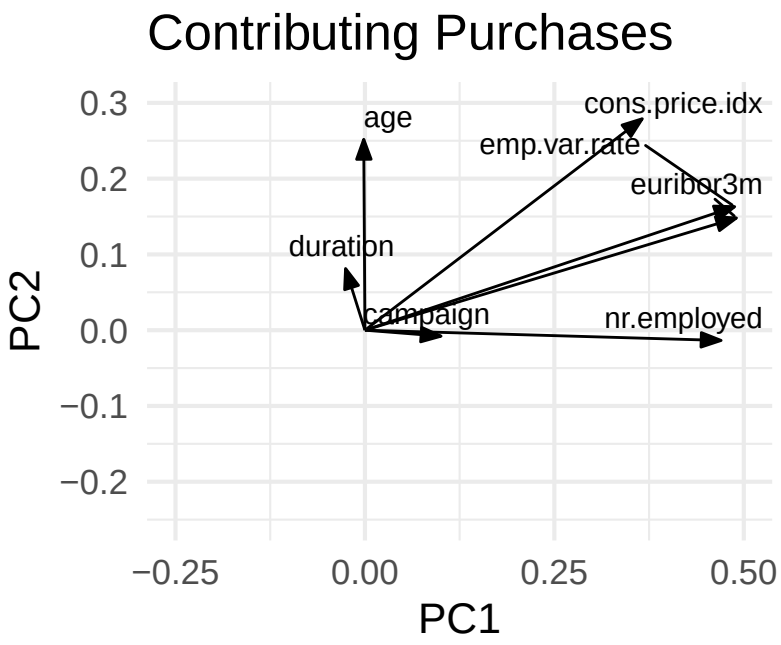
```
[1] 41188
```

```r
pca_fit <-  numeric_df |>
  dplyr::select(where(is.numeric))  |>   prcomp(scale = TRUE)

 kmodel <- pca_fit|>
    augment(numeric_df) |>
  dplyr::select(.fittedPC1:.fittedPC10) |>
  kmeans(centers=3,nstart = 10)

user_clusters = kmodel |> augment(numeric_df)
```

```r
arrow_style <- arrow(
  angle = 20, length = grid::unit(8, "pt"),
  ends = "first", type = "closed"
)
pca_fit |>
  tidy(matrix = "rotation")  |>
  pivot_wider(
    names_from = "PC", values_from = "value",
    names_prefix = "PC"
  )  |>
  ggplot(aes(PC1, PC2)) +
  geom_segment(
    xend = 0, yend = 0,
    arrow = arrow_style
  ) +
  geom_text_repel(aes(label = column), hjust = 1,vjust=1) +
  xlim(-0.25, 0.5) + ylim(-0.25, 0.3) +
  coord_fixed() +
  theme_minimal(base_size =16)+
  labs(title = "Contributing Purchases")
```

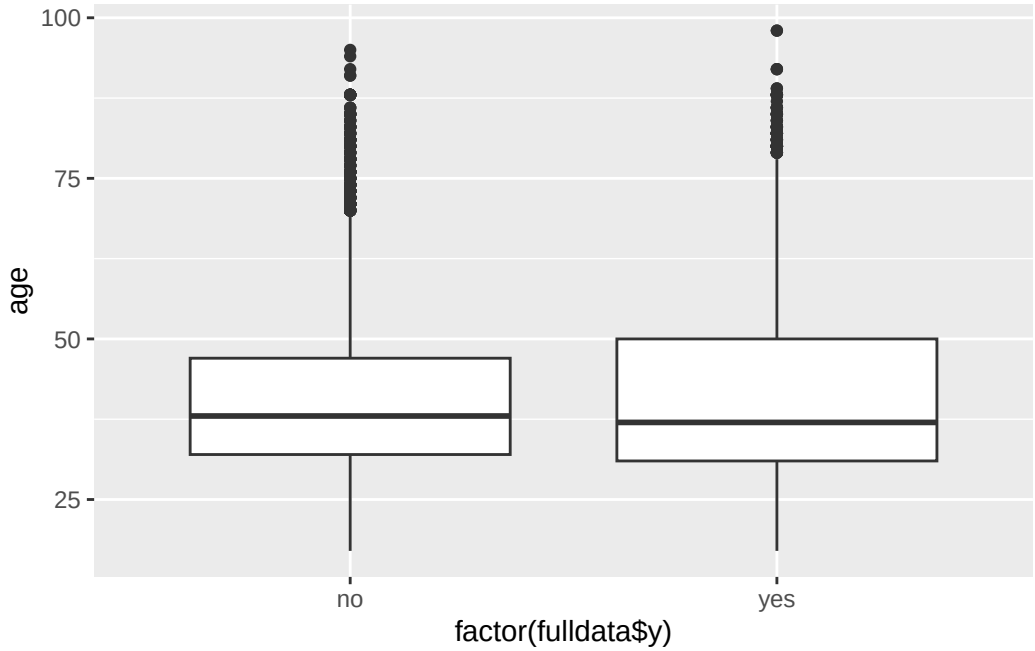
```
Warning: Removed 3 rows containing missing values or values outside the scale range
(`geom_segment()`).
```

```
Warning: Removed 3 rows containing missing values or values outside the scale range
(`geom_text_repel()`).
```



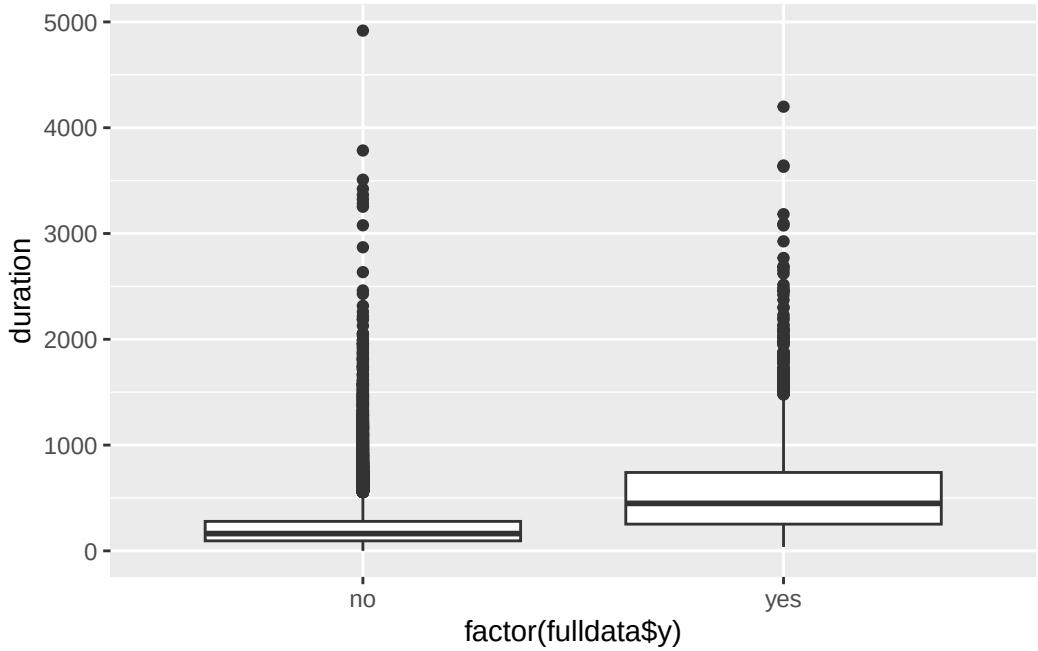

```
ggplot(fulldata, aes(x = factor(fulldata$y), y = age)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```



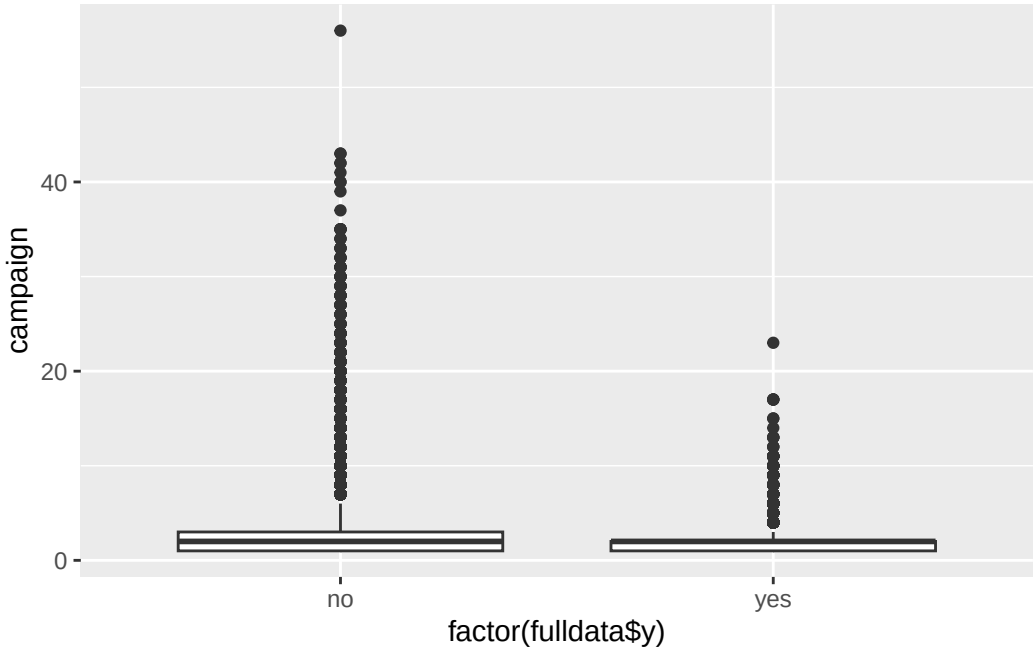

```
ggplot(fulldata, aes(x = factor(fulldata$y), y = duration)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```



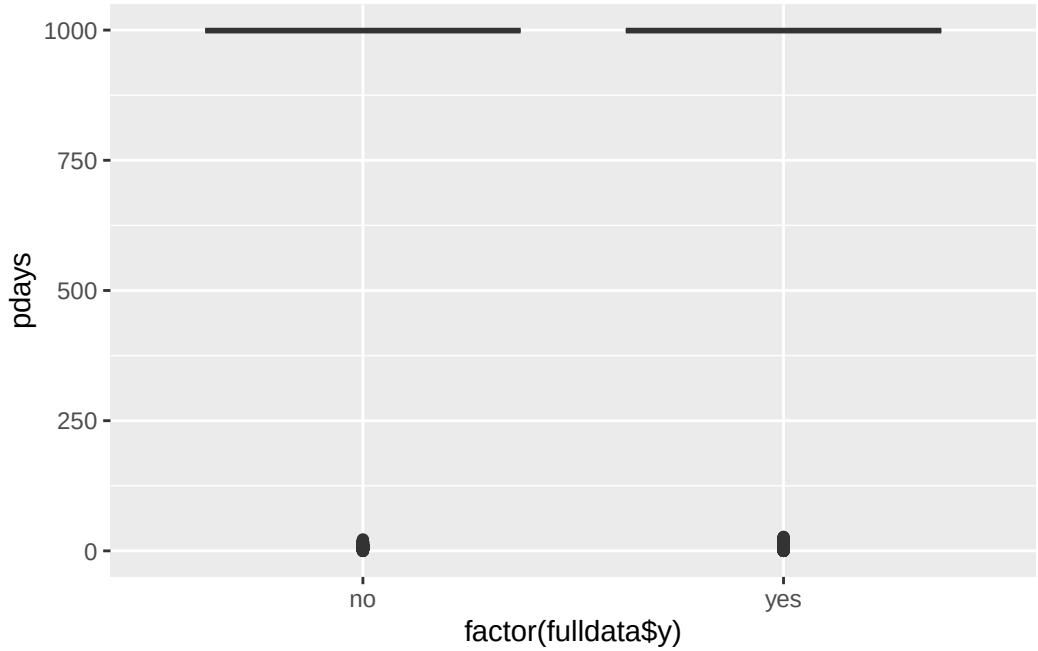

```
ggplot(fulldata, aes(x = factor(fulldata$y), y = campaign)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```



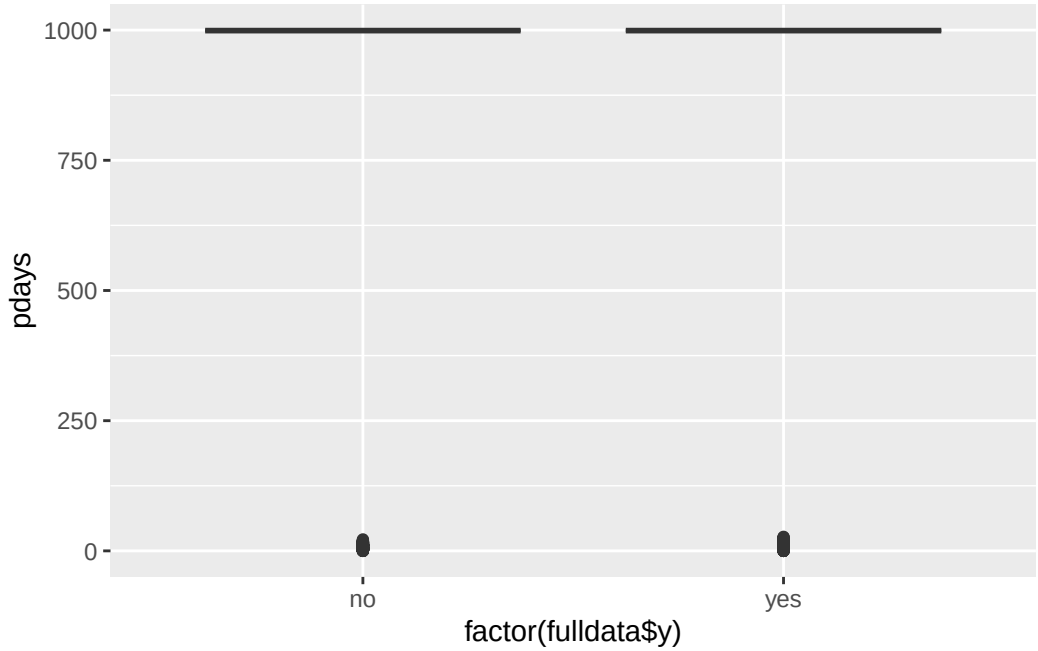

```
ggplot(fulldata, aes(x = factor(fulldata$y), y = pdays)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```



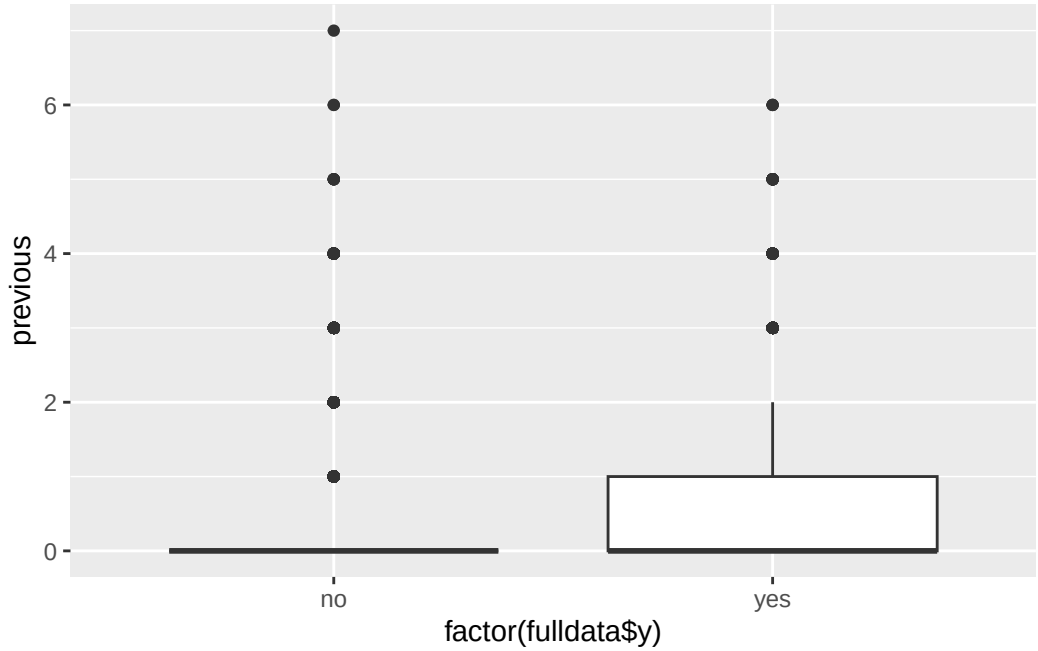

```
ggplot(fulldata, aes(x = factor(fulldata$y), y = pdays)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```



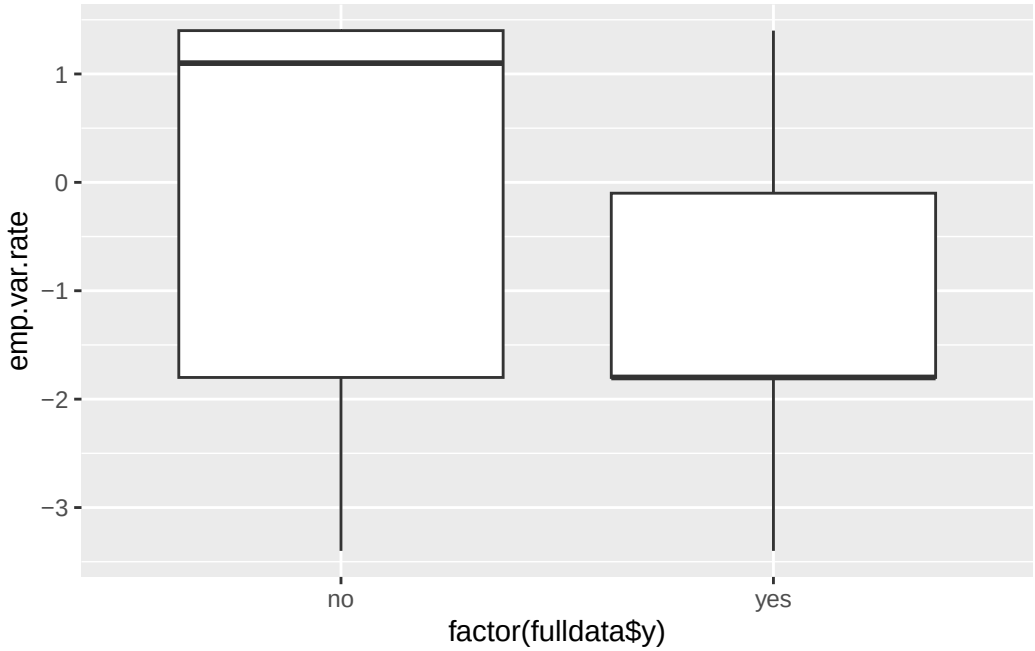

```
ggplot(fulldata, aes(x = factor(fulldata$y), y = previous)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```



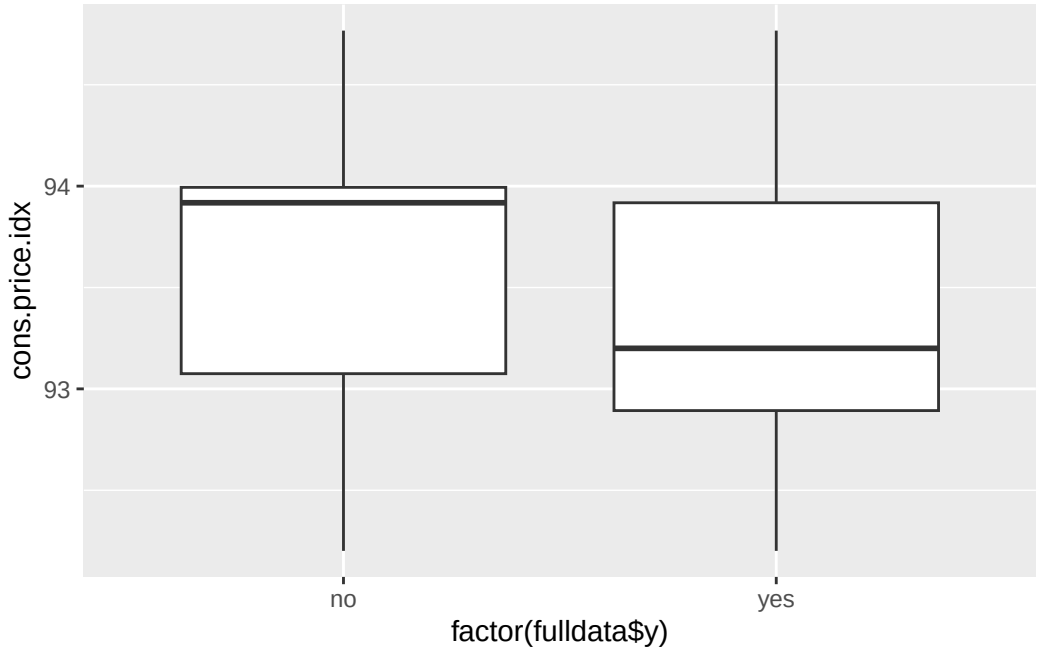

```
ggplot(fulldata, aes(x = factor(fulldata$y), y = emp.var.rate)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```



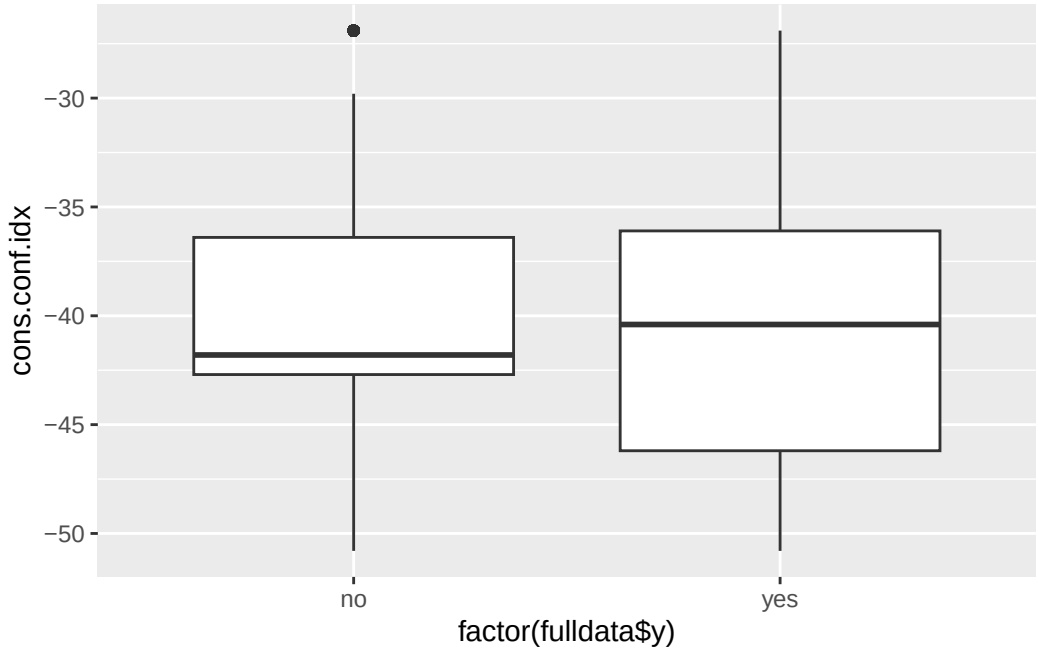

```
ggplot(fulldata, aes(x = factor(fulldata$y), y = cons.price.idx)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```



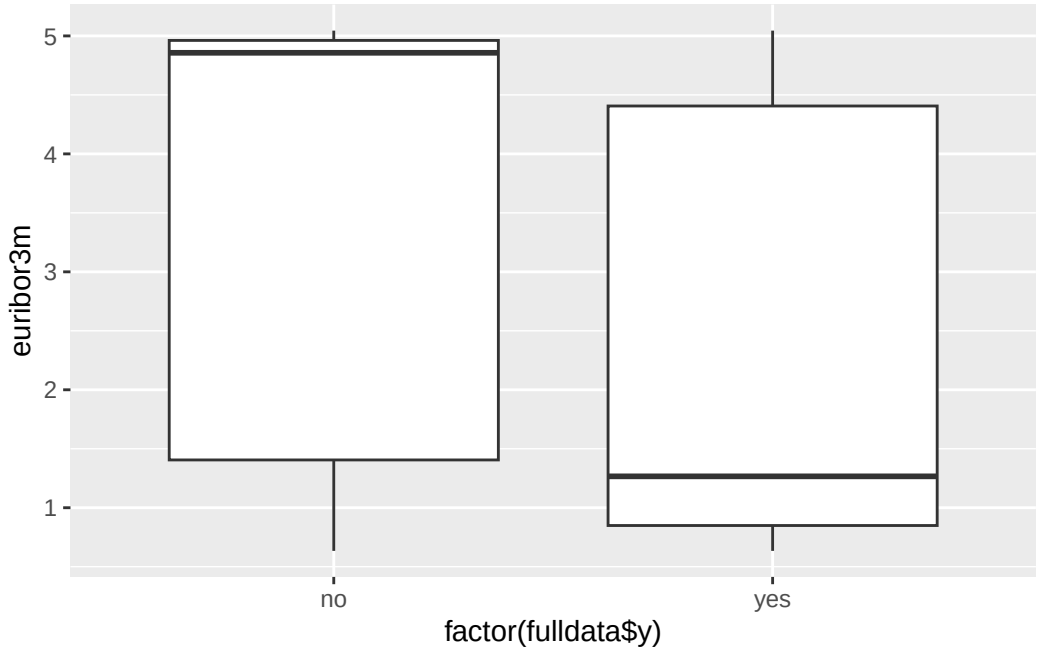

```
ggplot(fulldata, aes(x = factor(fulldata$y), y = cons.conf.idx)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```



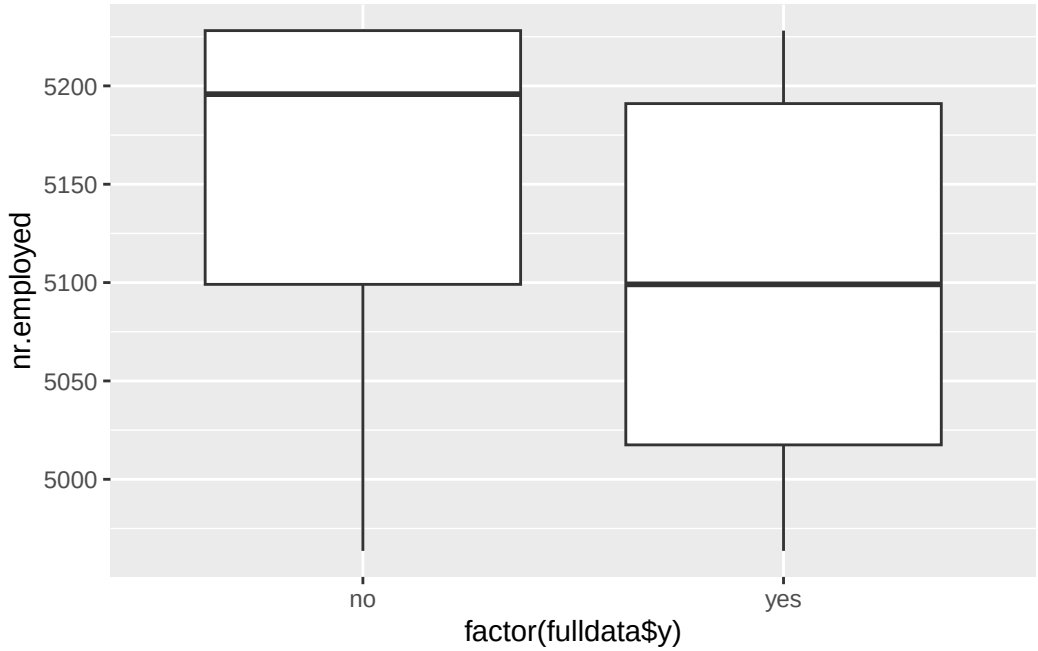

```
ggplot(fulldata, aes(x = factor(fulldata$y), y = euribor3m)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```




```
ggplot(fulldata, aes(x = factor(fulldata$y), y = nr.employed)) +
  geom_boxplot()
```

```
Warning: Use of `fulldata$y` is discouraged.
i Use `y` instead.
```




```
#emp.var.rate: employment variation rate - quarterly indicator (numeric) 17 - cons.price.idx
```

```
fulldata$term_deposit <- ifelse(fulldata$y == "yes", 1, 0)
lmodel <- glm(
  term_deposit ~ age + job + marital + education + default + housing + loan +
       contact + month + day_of_week +
       campaign + pdays + previous + poutcome +
       emp.var.rate + cons.price.idx + cons.conf.idx +
       euribor3m + nr.employed,
  data = fulldata,
  family = binomial
)

summary(lmodel)
```

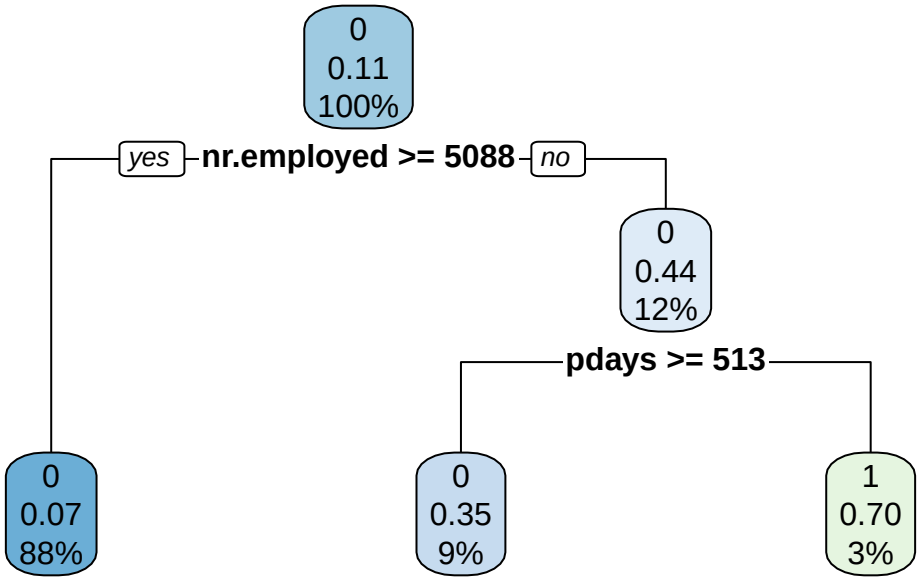
```
Call:
glm(formula = term_deposit ~ age + job + marital + education +
    default + housing + loan + contact + month + day_of_week +
    campaign + pdays + previous + poutcome + emp.var.rate + cons.price.idx +
    cons.conf.idx + euribor3m + nr.employed, family = binomial,
    data = fulldata)
```

```
Coefficients: (1 not defined because of singularities)
                             Estimate Std. Error z value Pr(>|z|)
(Intercept)                -2.285e+02  3.339e+01  -6.844 7.68e-12 ***
age                         4.392e-04  2.128e-03   0.206 0.836459
jobblue-collar             -1.515e-01  6.882e-02  -2.201 0.027704 *
jobentrepreneur            -7.367e-02  1.074e-01  -0.686 0.492598
jobhousemaid               -9.238e-02  1.285e-01  -0.719 0.472111
jobmanagement              -6.176e-02  7.520e-02  -0.821 0.411493
jobretired                  2.369e-01  9.460e-02   2.504 0.012281 *
jobself-employed           -6.599e-02  1.019e-01  -0.647 0.517390
jobservices                -1.288e-01  7.520e-02  -1.713 0.086788 .
jobstudent                  2.010e-01  1.005e-01   1.999 0.045594 *
jobtechnician              -1.596e-02  6.228e-02  -0.256 0.797699
jobunemployed              -2.144e-02  1.115e-01  -0.192 0.847486
jobunknown                 -1.740e-01  2.116e-01  -0.822 0.410920
maritalmarried              4.440e-02  6.010e-02   0.739 0.460069
maritalsingle               8.874e-02  6.847e-02   1.296 0.194968
maritalunknown              3.093e-01  3.724e-01   0.831 0.406187
educationbasic.6y           1.232e-01  1.037e-01   1.188 0.234735
educationbasic.9y          -2.216e-02  8.256e-02  -0.268 0.788383
educationhigh.school        3.548e-02  8.026e-02   0.442 0.658487
educationilliterate         9.020e-01  6.487e-01   1.391 0.164338
educationprofessional.course 4.688e-02  8.854e-02   0.529 0.596522
educationuniversity.degree  1.133e-01  8.039e-02   1.409 0.158746
educationunknown            1.072e-01  1.051e-01   1.020 0.307772
defaultunknown             -2.449e-01  5.753e-02  -4.257 2.07e-05 ***
defaultyes                 -8.629e+00  1.135e+02  -0.076 0.939387
housingunknown             -8.442e-02  1.193e-01  -0.708 0.479031
housingyes                 -2.936e-02  3.608e-02  -0.814 0.415741
loanunknown                        NA         NA      NA       NA
loanyes                    -2.636e-02  4.988e-02  -0.529 0.597112
contacttelephone           -7.369e-01  6.708e-02 -10.985  < 2e-16 ***
monthaug                    4.546e-01  1.080e-01   4.208 2.58e-05 ***
monthdec                    4.503e-01  1.892e-01   2.380 0.017305 *
monthjul                    5.507e-02  8.346e-02   0.660 0.509383
monthjun                   -6.472e-01  1.115e-01  -5.807 6.38e-09 ***
monthmar                    1.488e+00  1.303e-01  11.421  < 2e-16 ***
monthmay                   -4.061e-01  7.228e-02  -5.618 1.93e-08 ***
monthnov                   -4.498e-01  1.052e-01  -4.276 1.90e-05 ***
monthoct                    4.056e-02  1.353e-01   0.300 0.764349
monthsep                    2.373e-01  1.587e-01   1.496 0.134765
day_of_weekmon             -2.074e-01  5.796e-02  -3.579 0.000345 ***
```

```
day_of_weekthu                7.863e-02  5.578e-02   1.410 0.158638
day_of_weektue                5.814e-02  5.749e-02   1.011 0.311872
day_of_weekwed                1.616e-01  5.703e-02   2.833 0.004611 **
campaign                     -4.328e-02  9.185e-03  -4.712 2.45e-06 ***
pdays                        -1.094e-03  2.002e-04  -5.465 4.64e-08 ***
previous                     -6.269e-02  5.573e-02  -1.125 0.260643
poutcomenonexistent           4.442e-01  8.635e-02   5.144 2.70e-07 ***
poutcomesuccess               7.666e-01  1.958e-01   3.915 9.05e-05 ***
emp.var.rate                 -1.470e+00  1.241e-01 -11.841  < 2e-16 ***
cons.price.idx                2.068e+00  2.202e-01   9.393  < 2e-16 ***
cons.conf.idx                 2.930e-02  6.991e-03   4.192 2.77e-05 ***
euribor3m                     1.977e-01  1.146e-01   1.725 0.084590 .
nr.employed                   6.627e-03  2.717e-03   2.439 0.014708 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 28999  on 41187  degrees of freedom
Residual deviance: 22711  on 41136  degrees of freedom
AIC: 22815

Number of Fisher Scoring iterations: 10
```

```r
set.seed(42)
fulldata$term_deposit_factor <- factor(fulldata$term_deposit, levels = c(0, 1))
modeldata <- fulldata |> dplyr::select(-y, -term_deposit, -duration)
train_index <- sample(seq_len(nrow(modeldata)), size = 0.7 * nrow(modeldata))
train_data <- modeldata[train_index, ]
test_data  <- modeldata[-train_index, ]


tree_model <- rpart(
  term_deposit_factor ~ .,
  data = train_data,
  method = "class",
  control = rpart.control(
    cp = 0.01,
    minsplit = 20,
    maxdepth = 30
  )
)
```

```
rpart.plot(tree_model)
```




```
pred_class <- predict(tree_model, test_data, type = "class")
pred_prob <- predict(tree_model, test_data, type = "prob")

table(Predicted = pred_class, Actual = test_data$term_deposit_factor)
```

```
         Actual
Predicted     0     1
        0 10841  1134
        1   112   270
```

```
mean(pred_class == test_data$term_deposit_factor)
```

```
[1] 0.8991665
```

```
lmodel <- glm(
  term_deposit_factor ~ age + job + marital + education + default + housing + loan +
       contact + month + day_of_week +
```

```
       campaign + pdays + previous + poutcome +
       emp.var.rate + cons.price.idx + cons.conf.idx +
       euribor3m + nr.employed,
  data = modeldata,
  family = binomial
)

pred_prob <- predict(lmodel, test_data, type = "response")

pred_class <- ifelse(pred_prob >= 0.5, 1, 0)
pred_class <- factor(pred_class, levels = c(0, 1))

actual <- factor(test_data$term_deposit_factor, levels = c(0, 1))

conf_mat <- table(
  Predicted = pred_class,
  Actual = actual
)

conf_mat
```

```
         Actual
Predicted     0     1
        0 10789  1069
        1   164   335
```

```
accuracy <- sum(diag(conf_mat)) / sum(conf_mat)
accuracy
```

```
[1] 0.9002185
```

```
confusionMatrix(pred_class, actual, positive = "1")
```

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 10789  1069
         1   164   335
```

```
               Accuracy : 0.9002
                 95% CI : (0.8948, 0.9054)
    No Information Rate : 0.8864
    P-Value [Acc > NIR] : 4.337e-07

                  Kappa : 0.311

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.23860
            Specificity : 0.98503
         Pos Pred Value : 0.67134
         Neg Pred Value : 0.90985
             Prevalence : 0.11362
         Detection Rate : 0.02711
   Detection Prevalence : 0.04038
      Balanced Accuracy : 0.61182

       'Positive' Class : 1
```

```r
#QDA model

numeric_vars <- sapply(train_data, is.numeric)

train_qda <- train_data[, numeric_vars | names(train_data) == "term_deposit_factor"]
test_qda  <- test_data[,  numeric_vars | names(test_data) == "term_deposit_factor"]

qda_model <- qda(
  term_deposit_factor ~ .,
  data = train_qda
)

qda_pred <- predict(qda_model, test_qda)

table(
  Predicted = qda_pred$class,
  Actual    = test_qda$term_deposit_factor
)
```

```
         Actual
Predicted     0     1
```

```
         0 10127   704
         1   826   700
```

```
mean(qda_pred$class == test_qda$term_deposit_factor)
```

```
[1] 0.8761835
```

```
confusionMatrix(qda_pred$class, actual, positive = "1")
```

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 10127   704
         1   826   700

               Accuracy : 0.8762
                 95% CI : (0.8702, 0.8819)
    No Information Rate : 0.8864
    P-Value [Acc > NIR] : 0.999803

                  Kappa : 0.4077

 Mcnemar's Test P-Value : 0.001979

            Sensitivity : 0.49858
            Specificity : 0.92459
         Pos Pred Value : 0.45872
         Neg Pred Value : 0.93500
             Prevalence : 0.11362
         Detection Rate : 0.05665
   Detection Prevalence : 0.12349
      Balanced Accuracy : 0.71158

       'Positive' Class : 1
```

pre processing:

remove contact and duration

contact, default appears to be useless

housing and loan, contact and month, job and education highly correlated

term deposit somewhat correlated with poutcome, and month

term deposit is more correlated with job than education

There are many more no to term deposit than there are yes.

The dataset contains both numeric and categorical features. The numeric features of the data that are most correlated are: employment variation rate, number of employees, euribor 3 month rate (interest rates). Strong negative correlations exist between previous(number of contacts performed before this campaign and for this client) and pdays (number of days that passed by after the client was last contacted from a previous campaign),previous and number of employees. For the categorical features, the most correlated based on Cramer's V are housing and loan, contact and month, job and education.

With the exception of age none of the numeric data are normally distributed. data is either extremely skewed like with campaign or multimodal such as: consumer price index, consumer confidence index, employee variation rate, number employed, euribor3m (interest rates).

The target variable term deposits have significantly more "no's" than "yes'." This will have to be preprocessed to improve model performance. For the categorical data there are more married than single and divorced. A majority have a university degree for their education. about half have housing loans. there appears to be little to none of people who have defaulted on a loan. a majority of the clients do not have a personal loan. They seem to be mostly contacted in the month of may and clients are contacted each day of the week very evenly. There were very few successes from the last campaign, although it looks like a majority was "nonexistent." This may indicate that this campaign has more clients than before.

Numeric data with outliers include: previous, pdays, campaign, age. There appears to be only one outlier for consumer confidence index.

variables with high correlations tend to have clearer trends.

many features such as age, campaign, pdays and previous have little discernible trends.

There appears to be no missing data, however some features will be more useful than others.

Algorithm selection:

Since the output is binary a logistic model can be implemented by including the relevant features. logistic regression can use both numeric and categorical features. it has low variance when regularization is applied. Works well with limited data. In feature space it puts a linear decision boundary. sensitive to multicollinearity and struggles with complex nonlinear relationships. it assumes independence between features.

QDA requires numeric features and can create nonlinear decision boundaries which can fit the data well if the boundaries are not linear and the classes are very well separated. Sensitive to multicollinearity and outliers. It requires large sample sizes for each class. Low bias with high

variance due to the full covariance matrix per class. The features should be normalized before training.

Decision trees can also utilize categorical and numeric features. It can handle nonlinear relationships by using multiple boundaries in feature space. low bias with high variance.

So far without much preprocessing QDA performs the best in terms of predicting yes. I would like to try to optimize logistic regression or a decision tree to hopefully include a useful categorical variable.

With a low amount of data (~1000) a logisitic regression would be preferred, since it does not require a lot of data to be a decent model.

Data Cleaning:

Dimensions I would remove include: duration due to it not being a predictor, contact (provides no useful information since they are both phone methods), default (not enough data in the yes class for it to be useful). since housing and loan, contact and month, job and education are highly correlated, maybe it would be best to include housing, job and month without the others to reduce features. employment variation rate, number of employees, euribor 3 month rate (interest rates) are also highly correlated. it may be best to pick one or two of those numeric features to reduce the amount of features. QDA should have normalized data before implementation. Logistic regression may require regularization in order to improve predictions.

The training data should have equal amount of data points for no and yes in order to build a better classifier that is more sensitive to "yes." A large portion of the data has a classification of "no" the classifier will be bias towards "no" if the original ratio is kept.

In summary the dataset shows highly skewed and multimodal numeric features. There are categorical imbalances as well as correlated features. Logistic regression, QDA and decision trees are potential candidates for modeling the data. Pre processing is needed in order to build a model that can produce a reasonable accuracy.

reference:

Input variables: # bank client data: 1 - **age** (numeric) 2 - **job : type of job** (categorical: "admin.","blue-collar","entrepreneur","housemaid","management","retired","self-employed","services","student","technician","unemployed","unknown") 3 - **marital : marital status** (categorical: "divorced","married","single","unknown"; note: "divorced" means divorced or widowed) 4 - **education** (categorical: "basic.4y","basic.6y","basic.9y","high.school","illiterate","profes
5 - **default: has credit in default**? (categorical: "no","yes","unknown") 6 - **housing: has housing loan**? (categorical: "no","yes","unknown") 7 - **loan: has personal loan**? (categorical: "no","yes","unknown") # related with the last contact of the current campaign: 8 - **contact: contact communication type** (categorical: "cellular","telephone") 9 - **month: last contact month of year** (categorical: "jan", "feb", "mar", ..., "nov", "dec") 10 - **day_of_week: last contact day of the week** (categorical: "mon","tue","wed","thu","fri")

11 - duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration=0 then y="no"). Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model. # other attributes: 12 - **campaign: number of contacts performed during this campaign and for this client** (numeric, includes last contact) 13 - **pdays: number of days that passed by after the client was last contacted from a previous campaign** (numeric; **999 means client was not previously contacted**) 14 - **previous: number of contacts performed before this campaign and for this client (numeric)** 15 - **poutcome: outcome of the previous marketing campaign** (categorical: "failure","nonexistent","success") # social and economic context attributes 16 - **emp.var.rate: employment variation rate** - quarterly indicator (numeric) 17 - **cons.price.idx: consumer price index** - monthly indicator (numeric)
18 - **cons.conf.idx: consumer confidence index** - monthly indicator (numeric)
19 - **euribor3m: euribor 3 month rate** - daily indicator (numeric) 20 - **nr.employed: number of employees** - quarterly indicator (numeric)

Output variable (desired target): 21 - y - has the client subscribed a term deposit? (binary: "yes","no")