BCG - Task 3

Lingyu Tan

First, we need to load the data obtained in the last task.

```
hist_new <- read.csv("hist_new.csv")
train_new <- read.csv("train_new.csv")
```

We guess that whether a customer was churned depends on its average historical price of energy/power in 2015. In the last task, we have interpolated missing values in the historical dataset so every id has 12 records for 12 months respectively now. Also we can confirm that those id in training set and historical set are identical by sorting them and compare:

```
sum(sort(unique(hist_new$id)) == sort(unique(train_new$id))) == nrow(train_new)
```

```
## [1] TRUE
```

Then we can reorder these two dataset, calculate average annual historical prices and combine them with training dataset for convenience.

```
hist_order <- hist_new[order(hist_new$id), ]
train_order <- train_new[order(train_new$id), ]

p1_var <- rowMeans(matrix(hist_order$price_p1_var, ncol = 12, byrow = TRUE))
p2_var <- rowMeans(matrix(hist_order$price_p2_var, ncol = 12, byrow = TRUE))
p3_var <- rowMeans(matrix(hist_order$price_p3_var, ncol = 12, byrow = TRUE))

p1_fix <- rowMeans(matrix(hist_order$price_p1_fix, ncol = 12, byrow = TRUE))
p2_fix <- rowMeans(matrix(hist_order$price_p2_fix, ncol = 12, byrow = TRUE))
p3_fix <- rowMeans(matrix(hist_order$price_p3_fix, ncol = 12, byrow = TRUE))

new.df <- data.frame(p1_var, p2_var, p3_var, p1_fix, p2_fix, p3_fix)

data_combo <- cbind(train_order, new.df)
```

We guess that the duration time (`day = date_end - date_activ`) that a client has been using our products, the days left since the last modification until end of contract (`day_modif = date_end - date_modif_prod`), and days left from the next contract renewal to the end date (`day_renewal = date_end - date_renewal`) also have an impact.

```
day <- -as.duration(train_order$date_end %--% train_order$date_activ) / ddays(1)
day_modif <- -as.duration(train_order$date_end %--% train_order$date_modif_prod) / ddays(1)
day_renewal <- -as.duration(train_order$date_end %--% train_order$date_renewal) / ddays(1)

new.df <- data.frame(day, day_modif, day_renewal)
data_combo <- cbind(data_combo, new.df)
```

Also, we would like to convert boolean variables to numeric (False - 0, True - 1) as follows:

```
data_combo$has_gas <- as.integer(data_combo$has_gas)
data_combo$churn <- as.integer(data_combo$churn)
```

Finally, write the generated dataset with features of interest:

```
write.csv(data_combo, "data_combo.csv", row.names = FALSE)
```

Note: We have converted some characteristic variables to factors before but every time we write the dataset as csv file, this feature loses, so try to avoid read and write, and directly use the converted variables in R.