WUM PD6

Wojciech Celej

1. Załadowanie danych, budowa modelu

```
# Załadowanie danych

data <- read_dta("HCMST 2017 fresh sample for public sharing draft v1.1.dta")
features <- c("time_from_rel_to_cohab",
              "w6_q24_length",
              "hcm2017q24_church",
              "hcm2017q24_btwn_I_sig_other", "S1")
train_df <- data %>%
  select(features) %>%
  mutate(hcm2017q24_church = as.numeric(hcm2017q24_church),
         hcm2017q24_btwn_I_sig_other = as.numeric(hcm2017q24_btwn_I_sig_other),
         S1 = as.numeric(S1-1)) %>%
  na.omit() %>%
  unique() %>%
  as.data.frame()

X_train <- as.matrix(select(train_df, -S1))
y_train <- pull(train_df, S1)


# Dobór nrounds metodą CV

params = list("objective" = "binary:logistic",
              "eta" = 0.05,
              "max_depth" = 4)

# dtrain <- xgb.DMatrix(data = X_train, label = y_train)
#
# cv <- xgb.cv(params = params,
#              data = dtrain,
#              nrounds = 200,
#              nfold = 5,
#              showsd = TRUE,
#              eval_metric = "auc",
#              print_every_n = 5,
#              early_stopping_rounds = 20)
#
# print(cv$best_iteration)

nrounds <- 40
xgb_model <- xgboost(data = X_train,
                     label = y_train,
                     params = params,
                     nrounds = nrounds,
                     verbose = 0)
```

```r
## AUC modelu dla zbioru uczącego

y_pred <- predict(xgb_model, X_train)
auc <- performance(prediction(y_pred, y_train), measure = "auc")
auc@y.values
```

```
## [[1]]
## [1] 0.6923084
```

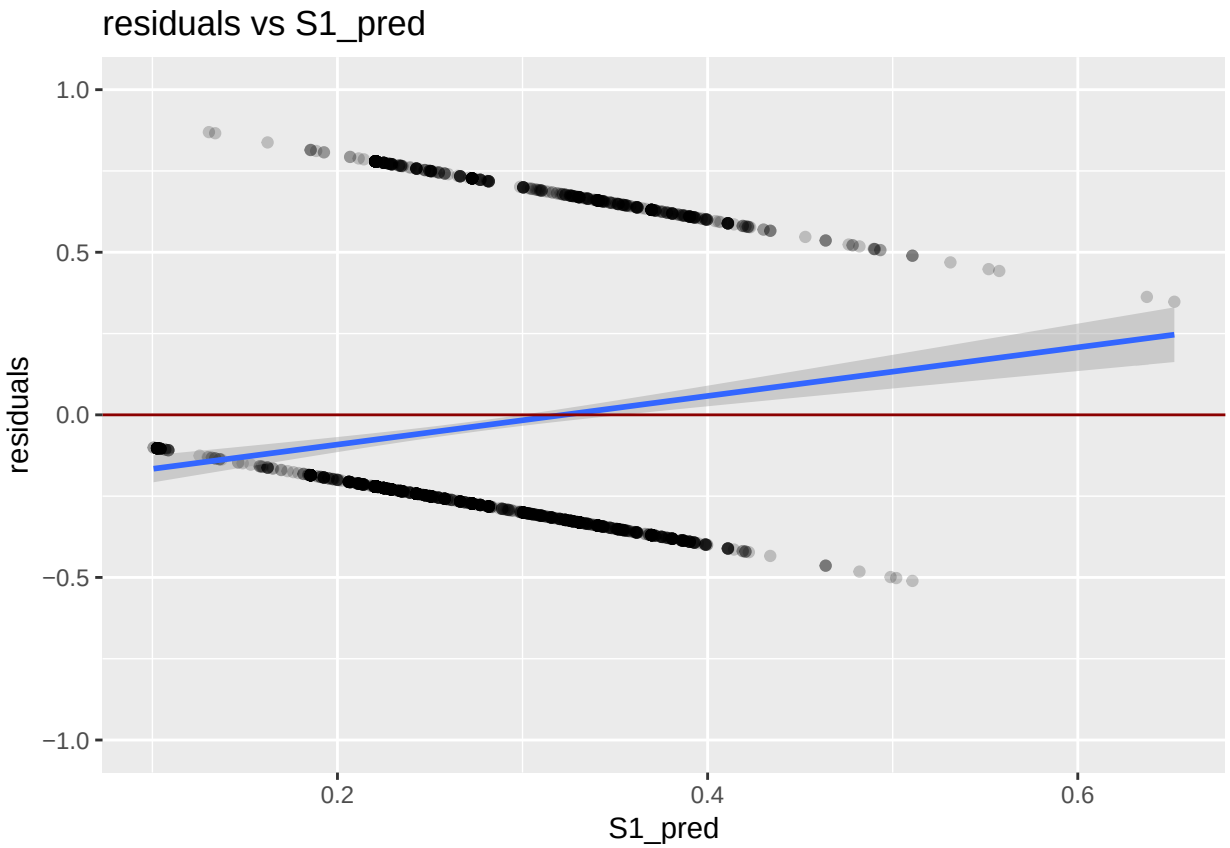
2. Wyznaczenie reszt na zbiorze uczącym

```r
train_df_ext <- train_df %>%
  mutate(S1_pred = y_pred,
         residuals = y_train - y_pred)

plot_residuals <- function(data, x, y) {
  p <- ggplot(data = train_df_ext, aes_string(x = x, y = y)) +
    geom_point(alpha = 0.2) +
    geom_smooth(method = "gam") +
    geom_hline(yintercept = 0, colour = "darkred") +
    labs(title = paste(y, "vs", x)) +
    scale_y_continuous(limits = c(-1, 1)) +
    theme_grey()
  return(p)
}
```

3. Zależność między resztą a wynikiem modelu

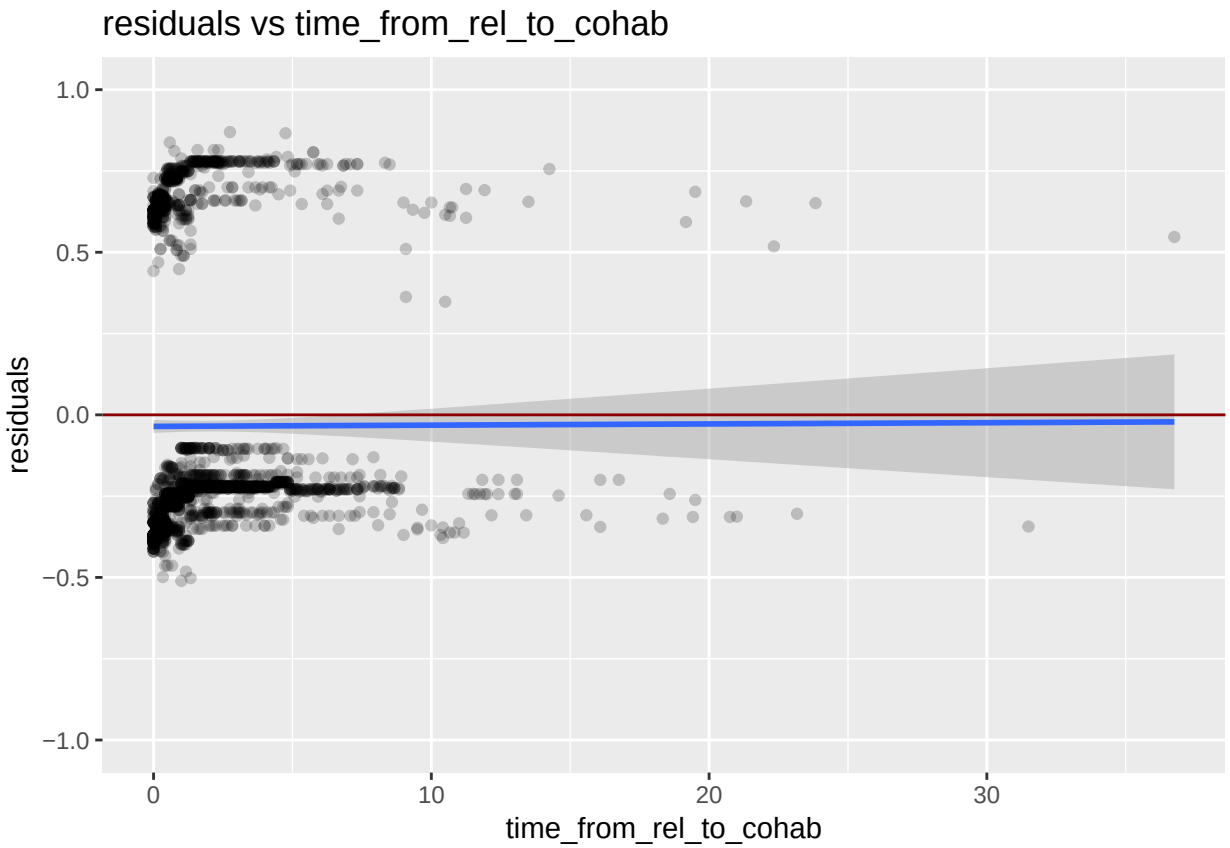
```r
plot_residuals(train_df_ext, "S1_pred", "residuals")
```



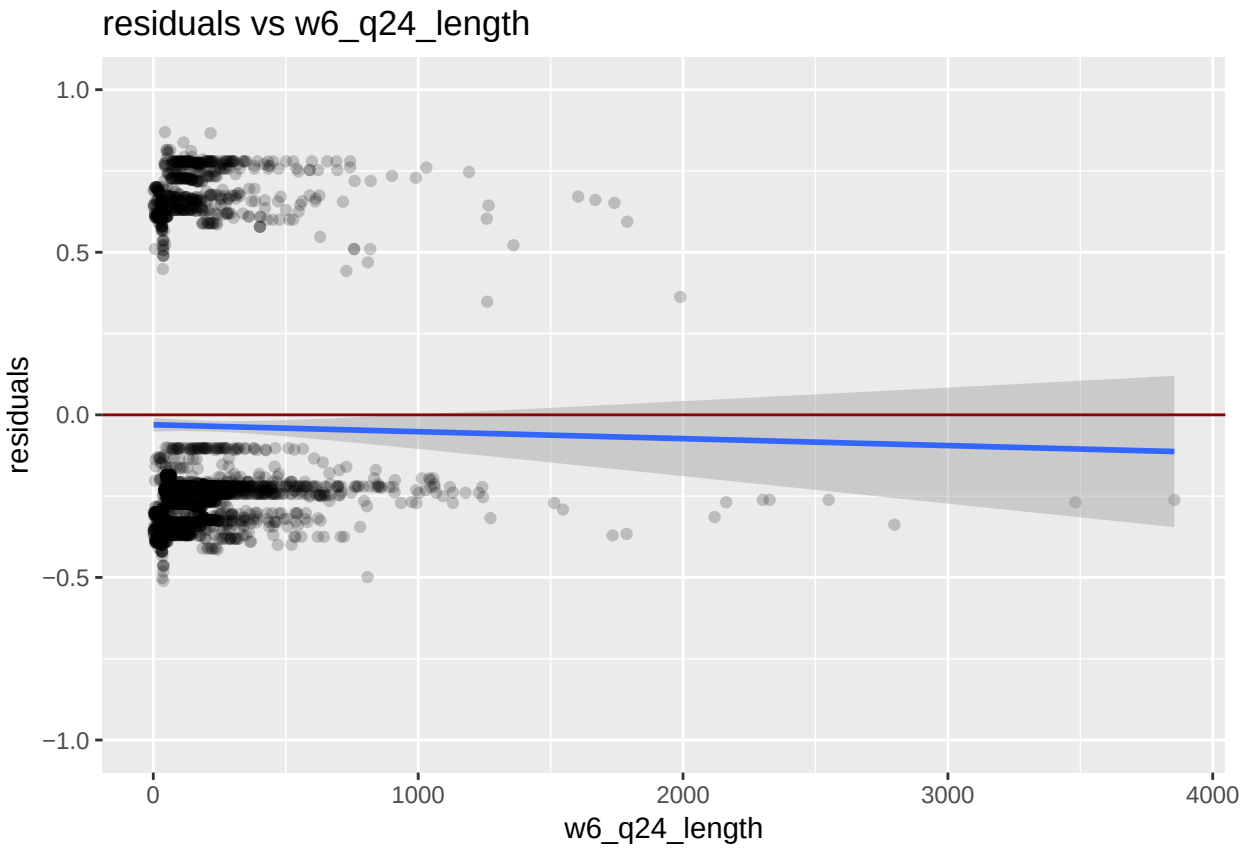
Krzywa lokalnego trendu różni się od funkcji stale równej 0 - jest rosnąca. Jest to zgodne z intuicją - obserwacji z wartościami predykcji mniejszymi od 0.5 i należącymi do klasy 0 (wartość `residuals <= 0`) powinno być więcej niż tych należących do klasy 1. Z kolei obserwacji z wartościami predykcji nie mniejszymi od 0.5 i należącymi do klasy 1 (wartość `residuals >= 0`) powinno być więcj niż tych z klasy 0, co powoduje trend rosnący.

4. Zależność między reszą a wybranymi zmiennymi zależnymi

```
plot_residuals(train_df_ext, "time_from_rel_to_cohab", "residuals")
```




```
plot_residuals(train_df_ext, "w6_q24_length", "residuals")
```



Krzywe lokalnego trendu w obu przypadkach nieznacznie różnią się od funkcji stale równej 0. W przypadku zmiennej `w6_q24_length` krzywa lokalnego trednu nieznacznie maleje, i dla dużych wartości x leży o ok. 0.1 poniżej 0, czyli przeszacowuje prawdopodobieństwo.

5. Obliczenie wartości Cooka dla obserwacji

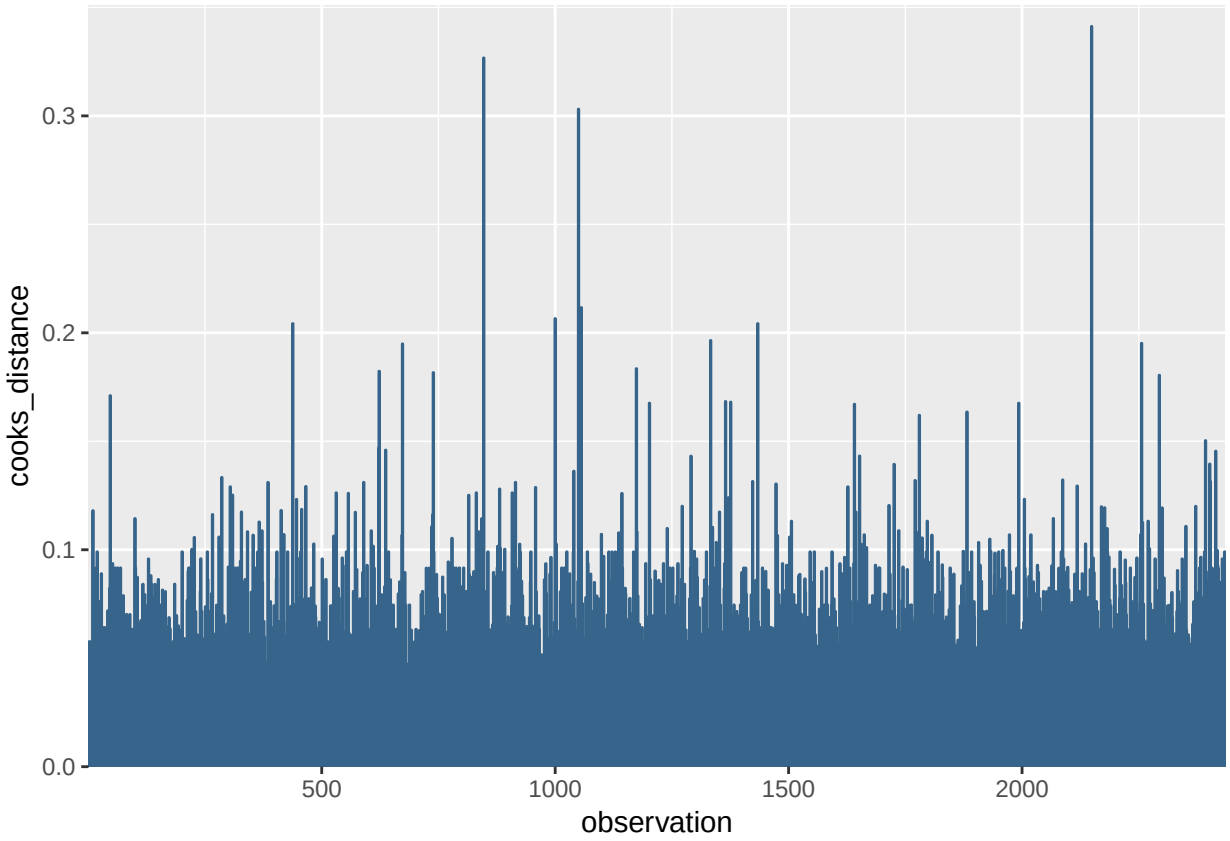
```
cooks_dist <- numeric(nrow(X_train))

for (i in seq(nrow(X_train))) {
  X_train_mod <- X_train[-i,]
  y_train_mod <- y_train[-i]
  xgb_model_mod <- xgboost(data = X_train_mod,
                           label = y_train_mod,
                           params = params,
                           nrounds = nrounds,
                           verbose = 0)
  y_pred_mod <- predict(xgb_model_mod, X_train_mod)
  cooks_dist[i] <- sum((y_pred[-i] - y_pred_mod)^2)
}

train_df_ext <- train_df_ext %>%
  mutate(cooks_distance = cooks_dist)

ggplot(data = train_df_ext, aes(x = seq(nrow(train_df_ext)), y = cooks_distance)) +
  geom_bar(width = 1,  stat = "identity", color = "steelblue4") +
  labs(x = "observation") +
```

```
  scale_x_continuous(expand = expand_scale(add = c(0, 0))) +
  scale_y_continuous(expand = expand_scale(add = c(0, 0.01))) +
  theme_grey()
```



Obserwacja z największą wartością Cooka

```
max_cooks_dist <- train_df_ext[which.max(train_df_ext[["cooks_distance"]]),]
kable(max_cooks_dist)
```

	time_from_rel_to_cohab	w6_q24_length	hcm2017q24_church	hcm2017q24_btwn_I_sig_other	S1	S1_
2149	0.083374	1604	0	0	1	0.328