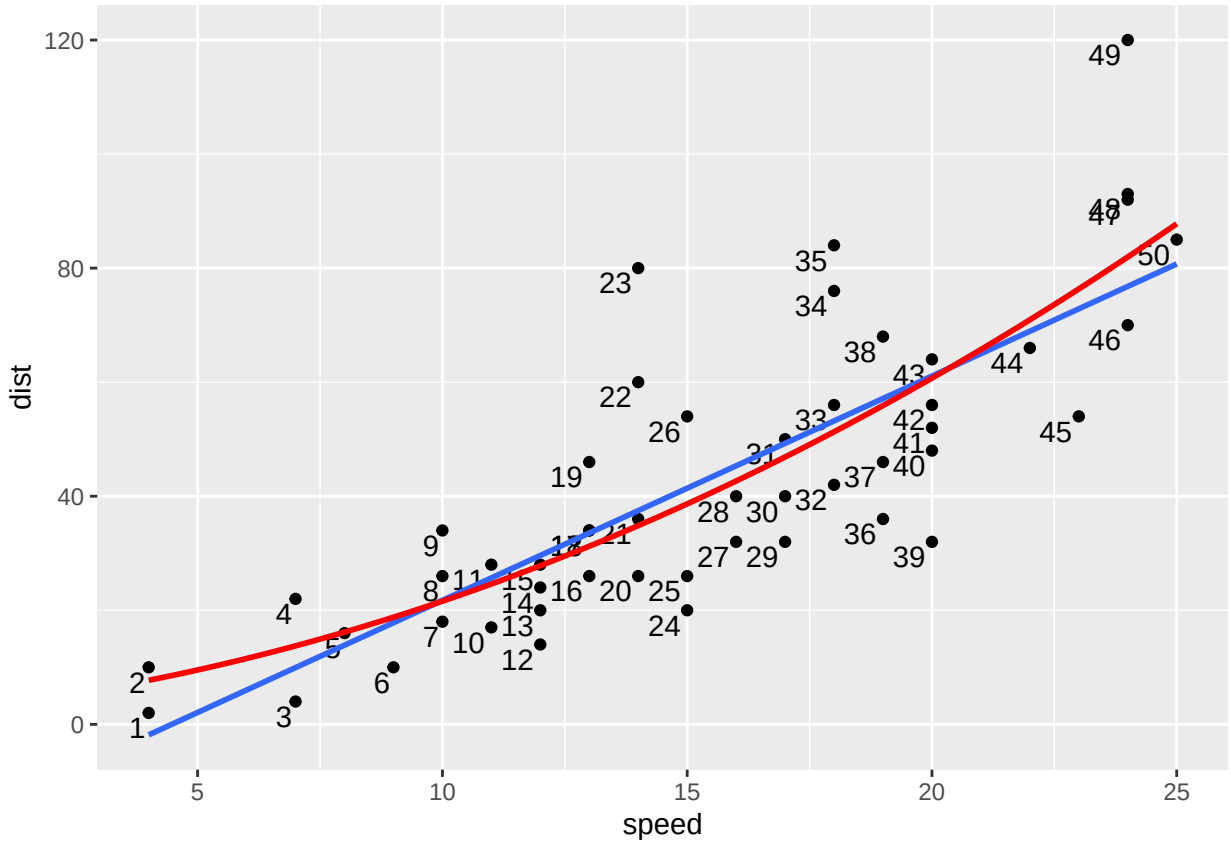
Ćwiczenia 4

2023-11-20

Zadanie 1

```
cars |>
  ggplot(aes(x = speed, y = dist)) +
  geom_point() +
  geom_text(aes(label = 1:NROW(cars)), vjust = 1.2, hjust = 1.2) +
  geom_smooth(method = "lm", se = FALSE) +
  geom_smooth(method = "lm", se = FALSE, formula = y ~ poly(x, 2), color = "red")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Zadanie 2

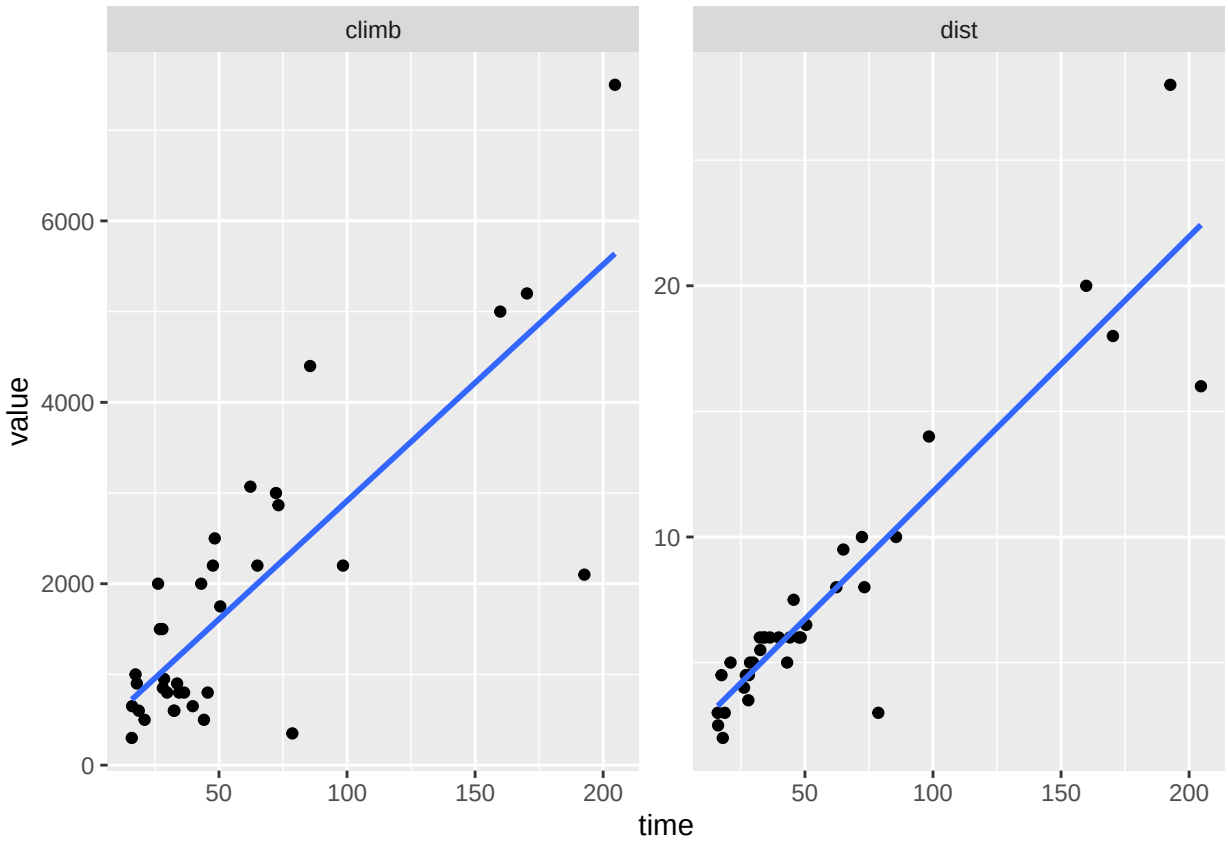
```
df <- MASS::hills
model <- lm(cbind(dist, climb) ~ time, data = df)
summary(model)
```

```
## Response dist :
```

```
##
## Call:
## lm(formula = dist ~ time, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.6374 -0.5558  0.0257  0.9714  6.7884
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.65347    0.57408    2.88  0.00693 **
## time         0.10151    0.00755   13.45 6.08e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.203 on 33 degrees of freedom
## Multiple R-squared:  0.8456, Adjusted R-squared:  0.841
## F-statistic: 180.8 on 1 and 33 DF,  p-value: 6.084e-15
##
##
## Response climb :
##
## Call:
## lm(formula = climb ~ time, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3227.3  -438.9   -76.4   549.6  1862.8
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   307.37     253.96   1.210    0.235
## time           26.05       3.34   7.801 5.45e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 974.5 on 33 degrees of freedom
## Multiple R-squared:  0.6484, Adjusted R-squared:  0.6378
## F-statistic: 60.86 on 1 and 33 DF,  p-value: 5.452e-09
```

```
df |>
  as_tibble(rownames = "Race") |>
  pivot_longer(cols = c("dist", "climb"),
               names_to = "variable", values_to = "value") |>
  ggplot(aes(x = time, y = value)) +
  geom_point() +
  facet_wrap(~ variable, scales = "free") +
  geom_smooth(method = "lm", se = FALSE)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Zadanie 3

```
df <- UsingR::homedata |> tibble()

model <- lm(y2000 ~ y1970, data = df)
summary(model)
```

```
##
## Call:
## lm(formula = y2000 ~ y1970, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -416665  -36308     809   34372  536605
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.040e+05  2.337e+03  -44.51   <2e-16 ***
## y1970        5.258e+00  3.147e-02  167.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 58000 on 6839 degrees of freedom
## Multiple R-squared:  0.8032, Adjusted R-squared:  0.8032
## F-statistic: 2.791e+04 on 1 and 6839 DF,  p-value: < 2.2e-16
```

```r
predict(model, data.frame(y1970 = 75000), interval = "prediction")
```

```
##        fit      lwr      upr
## 1 290343.2 176635.5 404050.9
```

Zad 4

```r
(df <- tibble(
  cena   = c(300, 250, 400, 550, 317, 389, 425, 289, 389, 559),
  pokoje = c(3, 3, 4, 5, 4, 3, 6, 3, 4, 5)
))
```

```
## # A tibble: 10 x 2
##     cena pokoje
##    <dbl>  <dbl>
##  1   300      3
##  2   250      3
##  3   400      4
##  4   550      5
##  5   317      4
##  6   389      3
##  7   425      6
##  8   289      3
##  9   389      4
## 10   559      5
```

```r
cat("------")
```

```
## ------
```

```r
model <- lm(cena ~ pokoje, data = df)
summary(model)
```

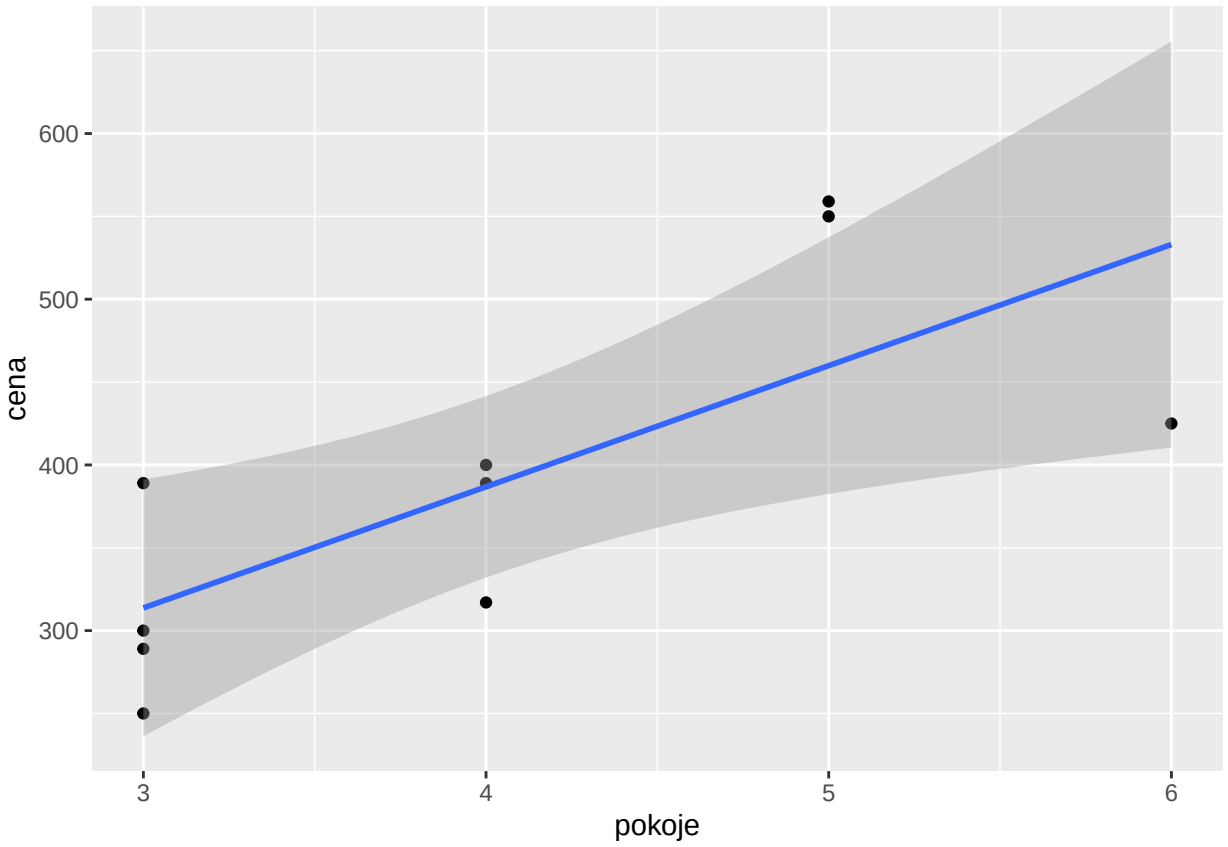
```
##
## Call:
## lm(formula = cena ~ pokoje, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -108.00  -53.95   -5.75   59.77   99.10
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    94.40      97.98   0.963   0.3635
## pokoje         73.10      23.76   3.076   0.0152 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 75.15 on 8 degrees of freedom
## Multiple R-squared:  0.5419, Adjusted R-squared:  0.4846
## F-statistic: 9.462 on 1 and 8 DF,  p-value: 0.01521
```

```r
predict(model, data.frame(pokoje = 2), interval = "prediction") |> print()
```

```
##     fit      lwr     upr
## 1 240.6 28.35795 452.842
```

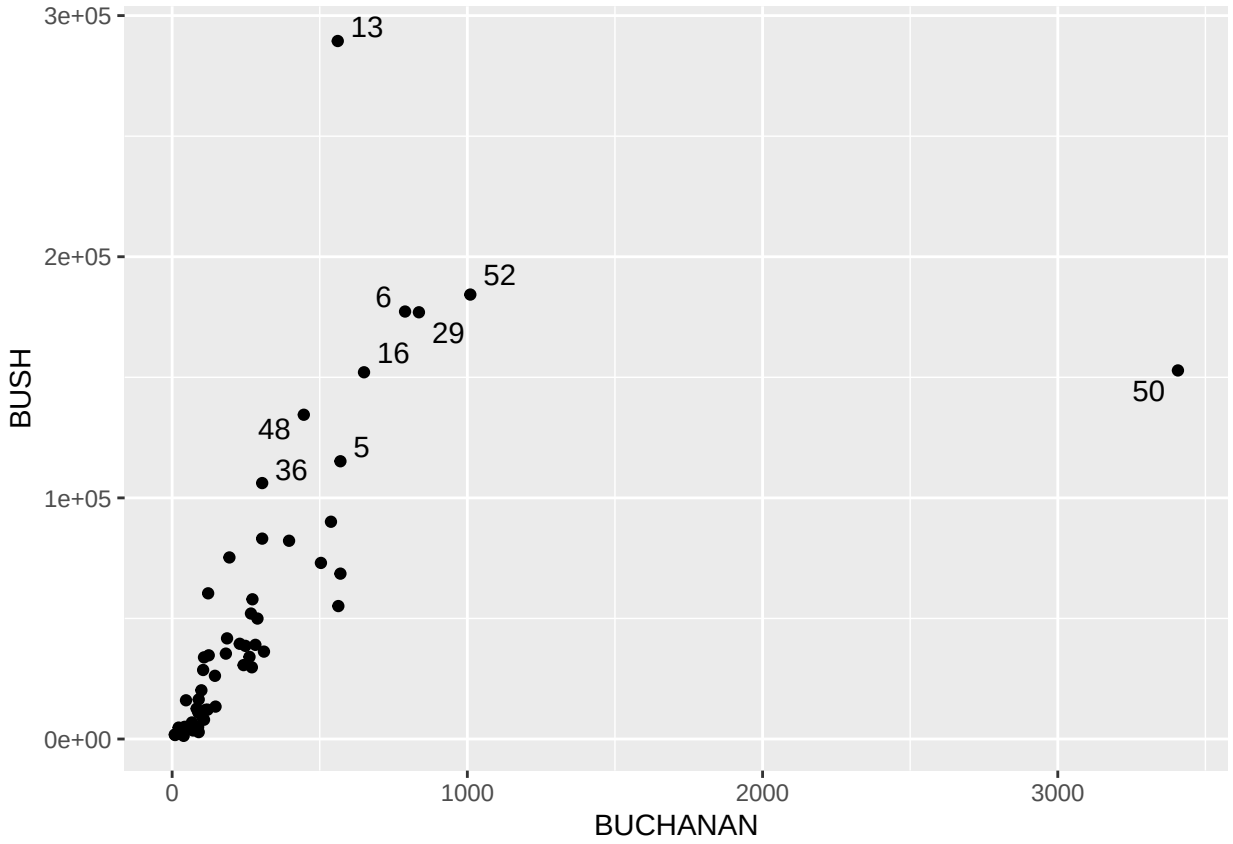
```
df |>
  ggplot(aes(y = cena, x = pokoje)) +
  geom_point() +
  geom_smooth(method = "lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Zadanie 5

```
df <- UsingR::florida
df |>
  transform(tt = ifelse(BUSH > 10^5 | BUCHANAN > 1000,
                        rownames(df), "")) |>
  ggplot(aes(y = BUSH, x = BUCHANAN)) +
  geom_point() +
  ggrepel::geom_text_repel(aes(label = tt))
```




```
m1 <- lm(BUCHANAN ~ BUSH, data = df[-c(13, 50),])
print(summary(m1))
```

```
##
## Call:
## lm(formula = BUCHANAN ~ BUSH, data = df[-c(13, 50), ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -200.94  -28.47  -11.06   27.52  281.67
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.854e+01  1.314e+01   2.934  0.00467 **
## BUSH        4.404e-03  2.193e-04  20.077  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 82.03 on 63 degrees of freedom
## Multiple R-squared:  0.8648, Adjusted R-squared:  0.8627
## F-statistic: 403.1 on 1 and 63 DF,  p-value: < 2.2e-16
```

```
cat("----------\n")
```

```
## ----------
```

```
print(predict(m1, df[df$County == "DADE", , drop = FALSE]))
```

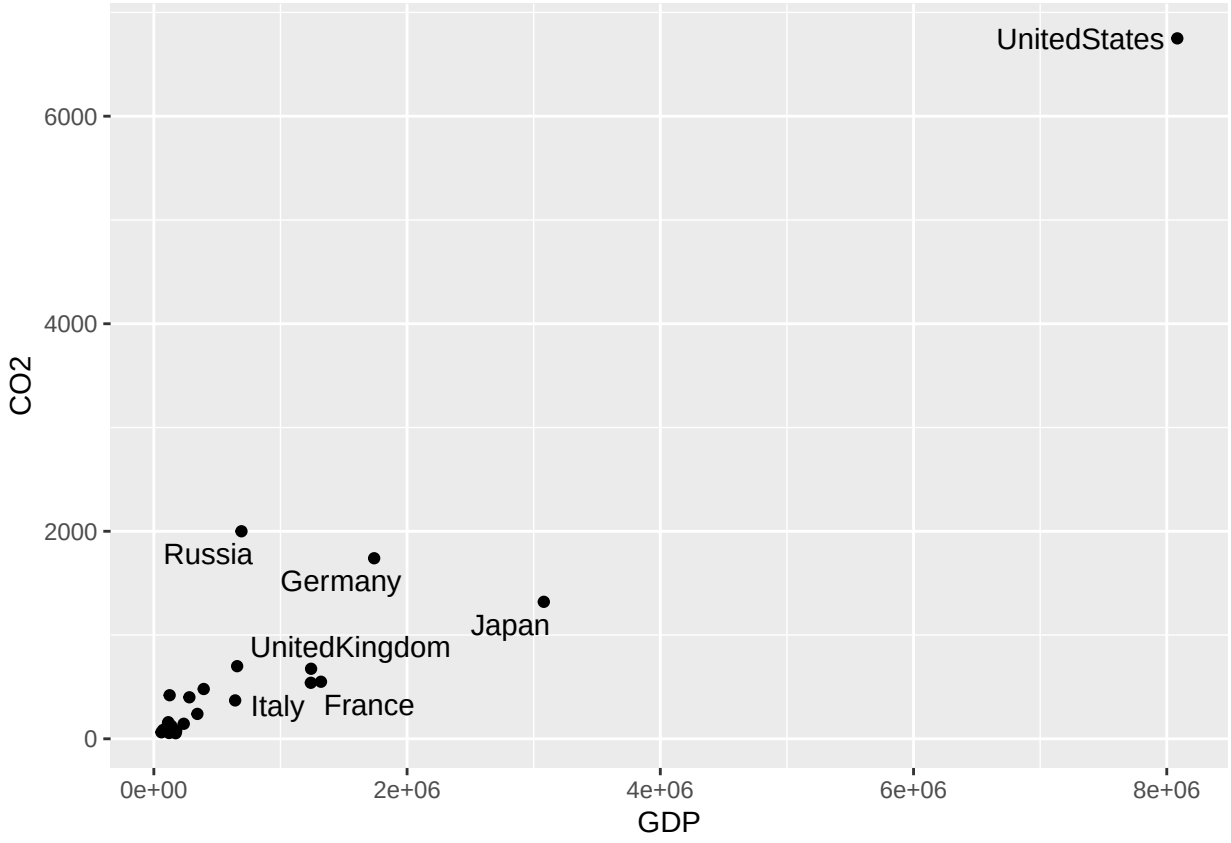
```
##     13
```

```
## 1313.2
```

Zadanie 6

```
df <- UsingR::emissions

df |>
  transform(tt = ifelse(CO2 > 1000 | GDP > 10 ^ 6, rownames(df), "")) |>
  ggplot(aes(y = CO2, x = GDP)) +
  geom_point() +
  ggrepel::geom_text_repel(aes(label = tt))
```




```
m_z_us   <- lm(CO2 ~ GDP, data = df)
m_bez_us <- lm(CO2 ~ GDP, data = df[-(which.max(df$GDP)), , drop = FALSE])
print(summary(m_z_us))
```

```
##
## Call:
## lm(formula = CO2 ~ GDP, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1107.35   -81.47   -32.69   126.33  1438.79
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.043e+01  9.441e+01   0.216     0.83
## GDP         7.815e-04  5.233e-05  14.933  1.2e-13 ***
```

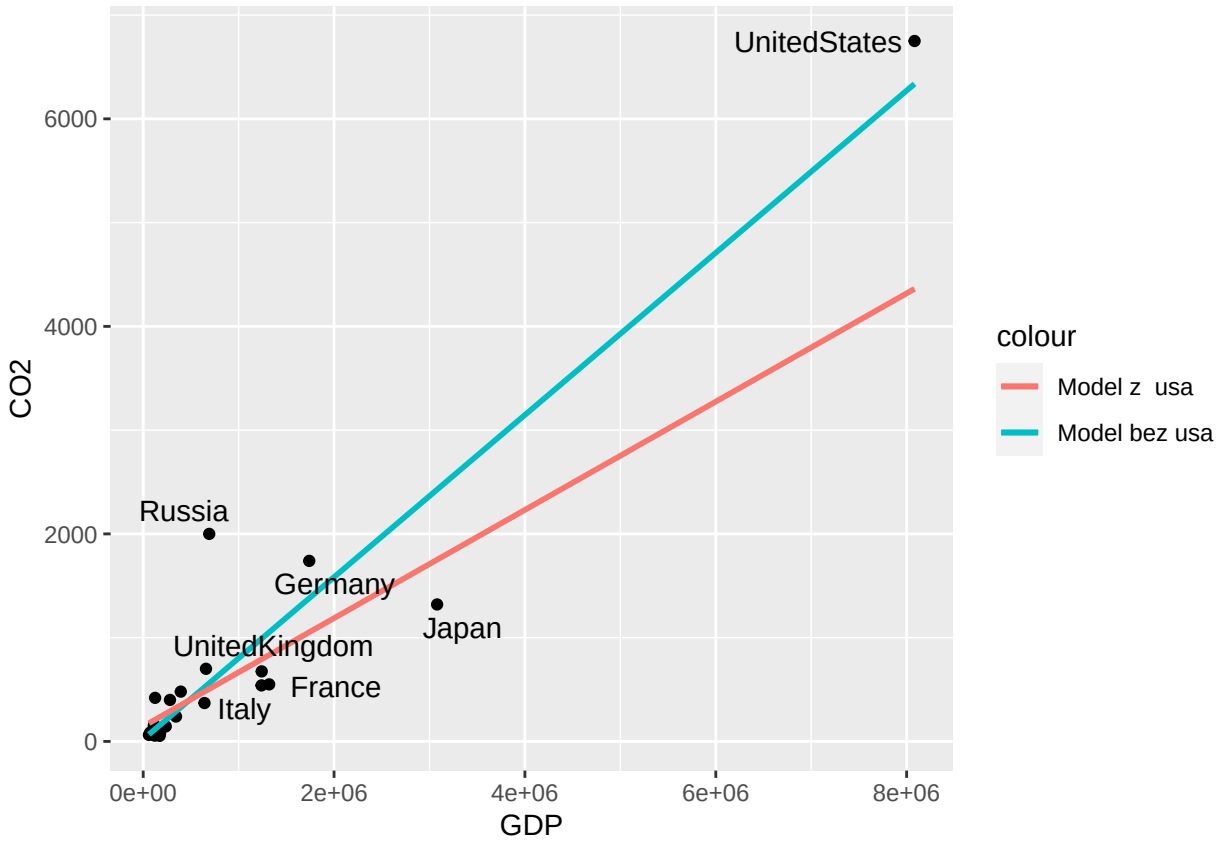
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 427.4 on 24 degrees of freedom
## Multiple R-squared:  0.9028, Adjusted R-squared:  0.8988
## F-statistic:   223 on 1 and 24 DF,  p-value: 1.197e-13
```

```
print(summary(m_bez_us))
```

```
##
## Call:
## lm(formula = CO2 ~ GDP, data = df[-(which.max(df$GDP)), , drop = FALSE])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -431.20 -151.12 -112.51  -43.84 1494.71
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.443e+02  9.749e+01   1.480    0.153
## GDP         5.217e-04  1.110e-04   4.701  9.8e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 384.3 on 23 degrees of freedom
## Multiple R-squared:   0.49,  Adjusted R-squared:  0.4679
## F-statistic:  22.1 on 1 and 23 DF,  p-value: 9.802e-05
```

```
df |>
  transform(tt = ifelse(CO2 > 1000 | GDP > 10 ^ 6, rownames(df), "")) |>
  ggplot(aes(y = CO2, x = GDP)) +
  geom_point() +
  geom_smooth(method = "lm", aes(col = "red"), se = FALSE,
              fullrange = TRUE) +
  geom_smooth(data = df[-(which.max(df$GDP)), , drop = FALSE],
              method = "lm", aes(col = "navy"),
              se = FALSE,
              fullrange = TRUE) +
  ggrepel::geom_text_repel(aes(label = tt)) +
  scale_color_discrete(label = c("Model z  usa", "Model bez usa"))
```

```
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
```



Zadanie 7

Usuwanie wyrazu wolnego nie ma sensu (bo dom bez toalet nie zostałby oddan za darmo)

```
df <- UsingR::homeprice

mm_bad <- lm(sale ~ half - 1, data = df)
print(summary(mm_bad))
```

```
##
## Call:
## lm(formula = sale ~ half - 1, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -222.62    6.44  117.70  215.00  450.00
##
## Coefficients:
##      Estimate Std. Error t value Pr(>|t|)
## half   242.56      39.36   6.163 1.18e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 196.8 on 28 degrees of freedom
## Multiple R-squared:  0.5757, Adjusted R-squared:  0.5605
## F-statistic: 37.98 on 1 and 28 DF,  p-value: 1.181e-06
```

```r
mm <- lm(sale ~ half, data = df)
summary(mm)
```

```
##
## Call:
## lm(formula = sale ~ half, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -180.27  -75.27  -22.34   72.66  246.58
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   228.27      28.78   7.932 1.59e-08 ***
## half           69.08      31.00   2.229   0.0344 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 109.8 on 27 degrees of freedom
## Multiple R-squared:  0.1554, Adjusted R-squared:  0.1241
## F-statistic: 4.966 on 1 and 27 DF,  p-value: 0.03436
```