Modele regresji i ich zastosowania - raport 2.

Patryk Krukowski(249824)

18 kwietnia 2021

Spis treści

1 Część praktyczna

1.1 Wstęp

Wczytujemy do pakietu statystycznego *R* dane z pliku *regresja wielokrotna.xlsx* zawierające 200 obserwacji jedenastu zmiennych: *X1*, *X2*, . . . , *X10* i *Y*, używając funkcji *read.xlsx2* z paczki *xlsx2*. Nim przystąpimy do wykonania zadań laboratoryjnych, zwróćmy uwagę na to, że danych nie ma wartości zakodowanych jako *NA*, dane są kompletne oraz poprawnie zakodowane. Ponadto załóżmy, że szum ϵ ma wielowymiarowy rozkład normalny $\mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}_n\right)$, gdzie oznaczenia są zgodne z oznaczeniami z wykładu.

```
## Loading required package: carData
## Registered S3 method overwritten by 'GGally':
##  method from
##  +.gg   ggplot2
##
## Attaching package: 'olsrr'
## The following object is masked from 'package:datasets':
##
##    rivers
```

```
##
## Attaching package: 'MASS'
## The following object is masked from 'package:olsrr':
##
##     cement
```

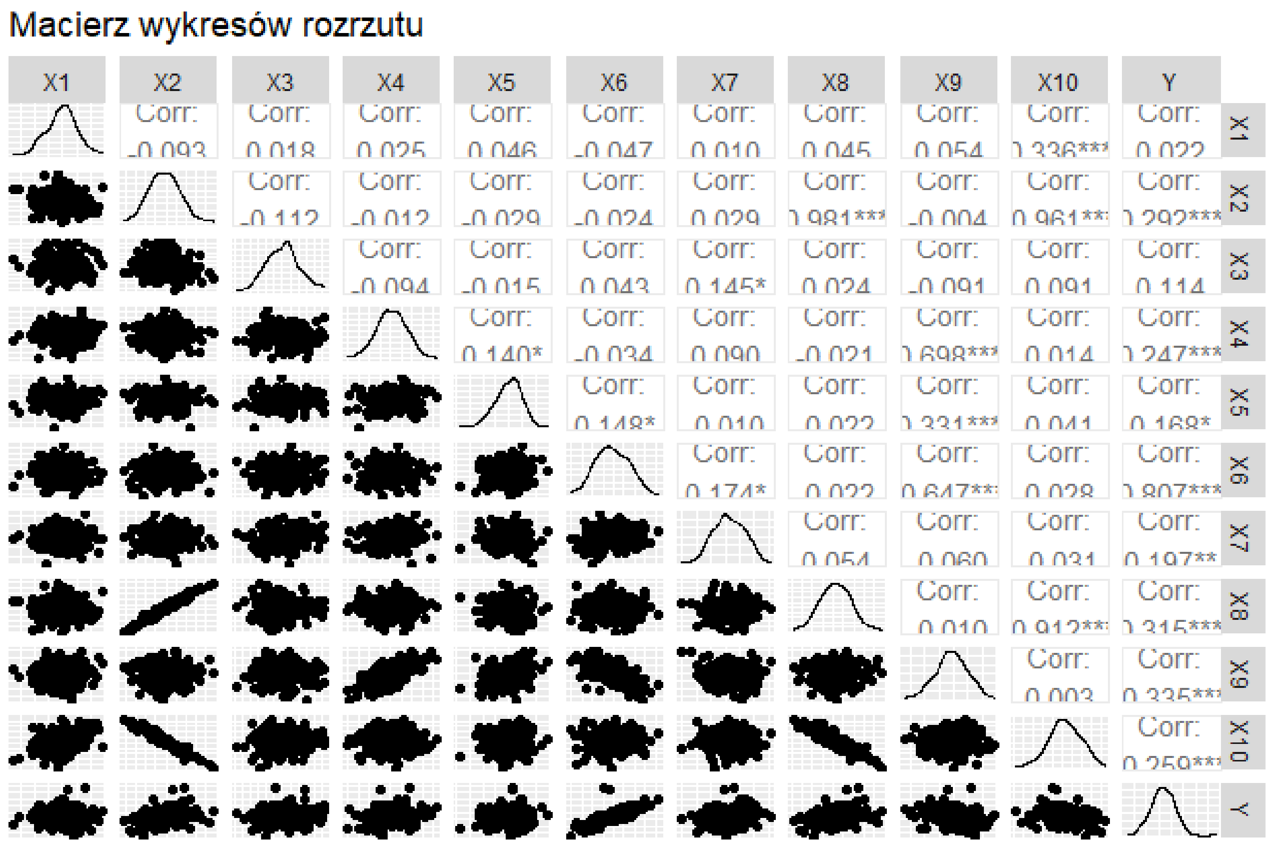
```
#Wczytujemy dane
dane <- read.xlsx2('C:/Users/Lenovo/Desktop/Modele regresji i ich zastosowania/regresja
lokrotna.xlsx',sheetName = 1)

#Konwertujemy typ 'list' do typu 'dobule' na potrzeby dzia■lania niektórych
#funkcji
dane <- apply(dane, MARGIN=2, function(x) return(as.numeric(x)))

#Sprawdzamy, czy istnieją w danych obserwacje zakodowane jako NA
apply(dane, 2, function(x) any(is.na(x))) #Nie istnieją

##    X1    X2    X3    X4    X5    X6    X7    X8    X9   X10     Y
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

Po wstępnym przyjrzeniu się danym, możemy przystąpić do analiy.

1.2 Zadanie 1.

Wykonajmy wykres rozrzutu dla każdej z $\binom{11}{2}$ par utworzonych przez zmienne *X1*, *X2*, ..., *X10* i *Y*, używając funkcji *ggpairs* z paczki *ggplot2*.

```
ggpairs(as.data.frame(dane), title = 'Macierz wykresów rozrzutu', axisLabels = "no-
ne")
```



Rysunek 1: Macierz wykresów rozrzutu dla zmiennych *X1*, *X2*, ..., *X10* i *Y*

Teraz przystąpmy do odpowiedzi na pytania z zadania.

(a) Najmocniejszy liniowy wpływ na zmienną objaśnianą Y wydaje się mieć zmienna *X6*.

(b) Wśród zmiennych objaśniających pojawia się problem współliniowości; pary zmiennych (*X2*,*X8*), (*X2*,*X10*) oraz (*X8*,*X10*) są silnie skorelowane.

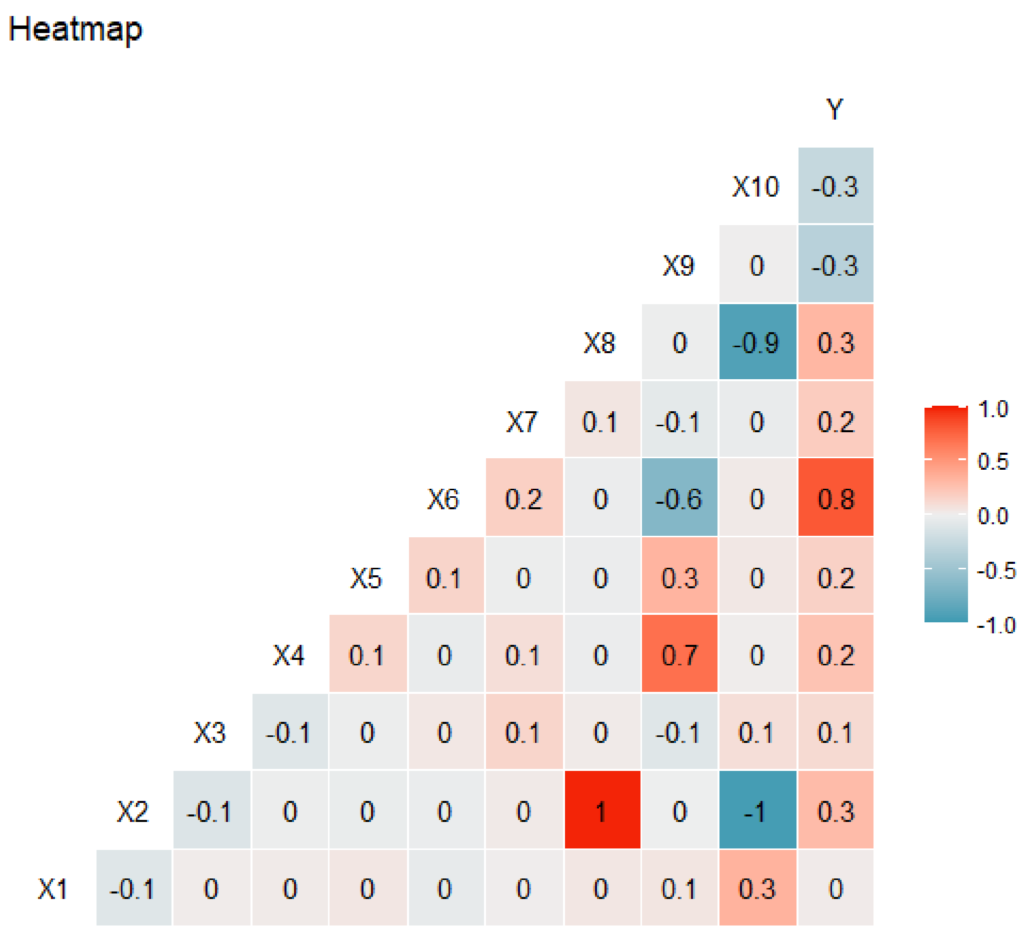
(c) Obserwacje odstające pojawiają się. Możemy to zaobserwować chociażby dla wykresu rozrzutu zmiennych (*X6*, Y).

1.3 Zadanie 2.

Wyznaczmy macierz korelacji próbkowych dla zmiennych Y, *X1*, *X2*, ..., *X10*, używając funkcji *ggcorr*, tak jak w poniższym kodzie. Dla lepszej wizualizacji używamy wykresu typu *heatmap*.

```
ggcorr(dane, method = c("everything", "pearson"), label =T) +
  ggtitle('Heatmap')
```

Odpowiedzi na podpunkty (a) oraz (b) z poprzedniego zadania są właściwie takie same.



Rysunek 2: Heatmap korelacji próbkowych dla zmiennych *X1*, *X2*, ..., *X10* i *Y*

1.4 Zadanie 3.

Konstruujemy model regresji liniowej między zmienną Y a zmiennymi objaśniającymi.

```
model <- lm(Y~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, data=as.data.frame(dane))
```

(a) Wyznaczamy estymator metodą najmniejszych kwadratów (OLS).

```
beta_hat <- model$coefficients
beta_hat

## (Intercept)          X1          X2          X3          X4          X5
##  0.52260442  2.84867658  1.82514674  3.64879728  3.95371546  0.21928094
##          X6          X7          X8          X9         X10
## 11.00583662 -0.03279499 -0.14514568  0.13848213 -0.73124055
```

(b) Spójrzmy na p-wartość testu F, którą możemy odczytać, korzystając z funkcji *summary*.

```
summary(model)

##
## Call:
## lm(formula = Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9 +
##     X10, data = as.data.frame(dane))
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.810 -1.972 -1.048  0.070 97.059
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.52260    6.65182   0.079   0.9375
## X1           2.84868    4.02874   0.707   0.4804
## X2           1.82515    7.20785   0.253   0.8004
## X3           3.64880    3.61818   1.008   0.3145
## X4           3.95372    2.37698   1.663   0.0979 .
## X5           0.21928    2.46797   0.089   0.9293
## X6          11.00584    2.38215   4.620 7.08e-06 ***
## X7          -0.03279    0.27664  -0.119   0.9058
## X8          -0.14515    3.59911  -0.040   0.9679
## X9           0.13848    2.34504   0.059   0.9530
## X10         -0.73124    1.50367  -0.486   0.6273
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.14 on 189 degrees of freedom
## Multiple R-squared:  0.8534,Adjusted R-squared:  0.8456
## F-statistic:   110 on 10 and 189 DF,  p-value: < 2.2e-16
```

Widzimy, że p-wartość testu F dla pełnego modelu jest $< 2.2e - 16$. Aby dobrze zinterpretować ten wynik, przypomnijmy, że hipoteza zerowa H_0 mówi o tym, że model zawiera jedynie stałą. Natomiast H_1 stwierdza, że H_0 jest fałszywa, tzn. istnieje co najmniej jedna zmienna objaśniająca, która ma liniowy wpływ na zmienną objaśnianą Y. Zatem w związku z otrzymanym wynikiem, odrzucamy H_0 na poziomie istotności $\alpha = 0.05$.

Wniosek 1 *W przyjętym modelu regresji liniowej istnieją zmienne objaśniajace mające liniowy wpływ na zmienną objaśnianą Y.*

(c) Wyznaczamy współczynniki determinacji R^2 oraz $AdjR^2$.

```
R_kwadrat <- summary(model)$r.squared
adj_R_kwadrat <- summary(model)$adj.r.squared
R_kwadrat

## [1] 0.8533544
```

```
adj_R_kwadrat
```

```
## [1] 0.8455954
```

Zatem, w przybliżeniu, $R^2 = 0.853$ oraz $AdjR^2 = 0.85$. Obie miary dopasowania modelu do danych dają podobny rezultat.

Wniosek 2 *Powyższe miary dopasowania (pełnego) modelu do danych wskazują na to, że przyjęty model nie sprawuje się najgorzej, ale też nie najlepiej.*

1.5 Zadanie 4.

W zadaniu tym zajmiemy się problemem współliniowości.

(a) Sprawdźmy, zgodnie z regułą kciuka, dla których zmiennych objaśniających współczynnik *VIF* (funkcja *vif* w pakiecie R) przekracza 10, następnie, spośród tych zmiennych, wybierzmy zmienną, dla której *VIF* jest największe i usuńmy ją z modelu (stwarzając nowy model *model_1*).

```
dane <- as.data.frame(dane)
vif(model) #X2 ma największą wartość wspó■lczynnika VIF, wiec usuwamy z mo-
delu
```

```
##          X1          X2          X3          X4          X5          X6
##   35.145297 1497.679681   27.256636   36.089952   11.758465   42.033020
##          X7          X8          X9         X10
##    1.093435 1469.489960   89.575184   73.671270
```

```
model_1 <- lm(Y~X1+X3+X4+X5+X6+X7+X8+X9+X10, data=as.data.frame(dane)) #no-
wy model
```

A zatem z modelu powinniśmy usunąć zmienną *X2*.

(b) Obliczamy wskaźniki podbicia wariancji dla zmiennych z *model_1* i, w razie potrzeby, usuwamy.

```
vif(model_1) #usuwamy X9
```

```
##        X1        X3        X4        X5        X6        X7        X8        X9
## 11.331815  1.852655 35.844845 11.605348 41.780951  1.074035 64.180276 88.932719
##       X10
## 72.840327
```

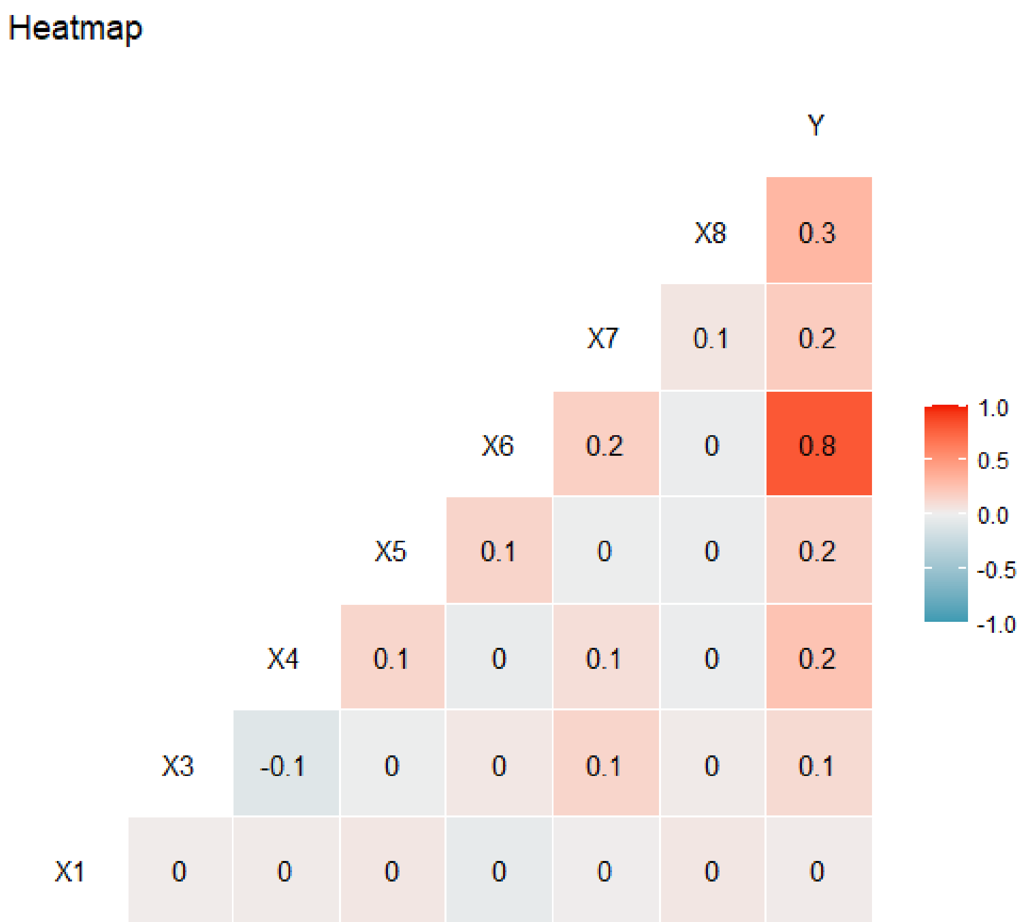
```
model_2 <- lm(Y~X1+X3+X4+X5+X6+X7+X8+X10, data=as.data.frame(dane))
vif(model_2) #Usuwamy X10
```

```
##        X1        X3        X4        X5        X6        X7        X8       X10
## 11.276908  1.842674  1.048843  1.051652  1.098026  1.072346 63.522539 72.147455
```

```
model_3 <- lm(Y~X1+X3+X4+X5+X6+X7+X8, data=as.data.frame(dane))
vif(model_3) #Rozwiązaliśmy problem współliniowości

##       X1       X3       X4       X5       X6       X7       X8
## 1.008049 1.034192 1.047199 1.050985 1.065881 1.071434 1.007002
```

Obliczenia pokazały, że musimy usunąć jeszcze zmienne *X9* oraz *X10*. Gdy to zrobimy, to *model_3*, składający się z pozostałych zmiennych, nie wykazuje się problemem współliniowości, co możemy sprawdzić jeszcze poniżej na wykresie typu heatmap.

```
dane_1 <- dane
dane_1$X2 <- NULL
dane_1$X9 <- NULL
dane_1$X10 <- NULL
ggcorr(dane_1, method = c("everything", "pearson"), label =T) +
  ggtitle('Heatmap')
```



Rysunek 3: Heatmap korelacji próbkowych dla zmiennych z modelu *model_3*

1.6 Zadanie 5.

Zidentyfikujmy teraz obserwacje wpływowe danych *dane_1*, na bazie których zbudowaliśmy model *model_3*, użwając poniższych miar.

(a) Wpływy (leverages) - zgodnie z regułą kciuka, jeśli i-ty element diagonali maicerzy daszkowej $\mathbf{H}$, h_{ii}, jest większy, niż $3\frac{p}{n}$, gdzie p jest śladem $\mathbf{H}$, a n liczbą obserwacji, to i-tą obserwację uznajemy za wpływową.

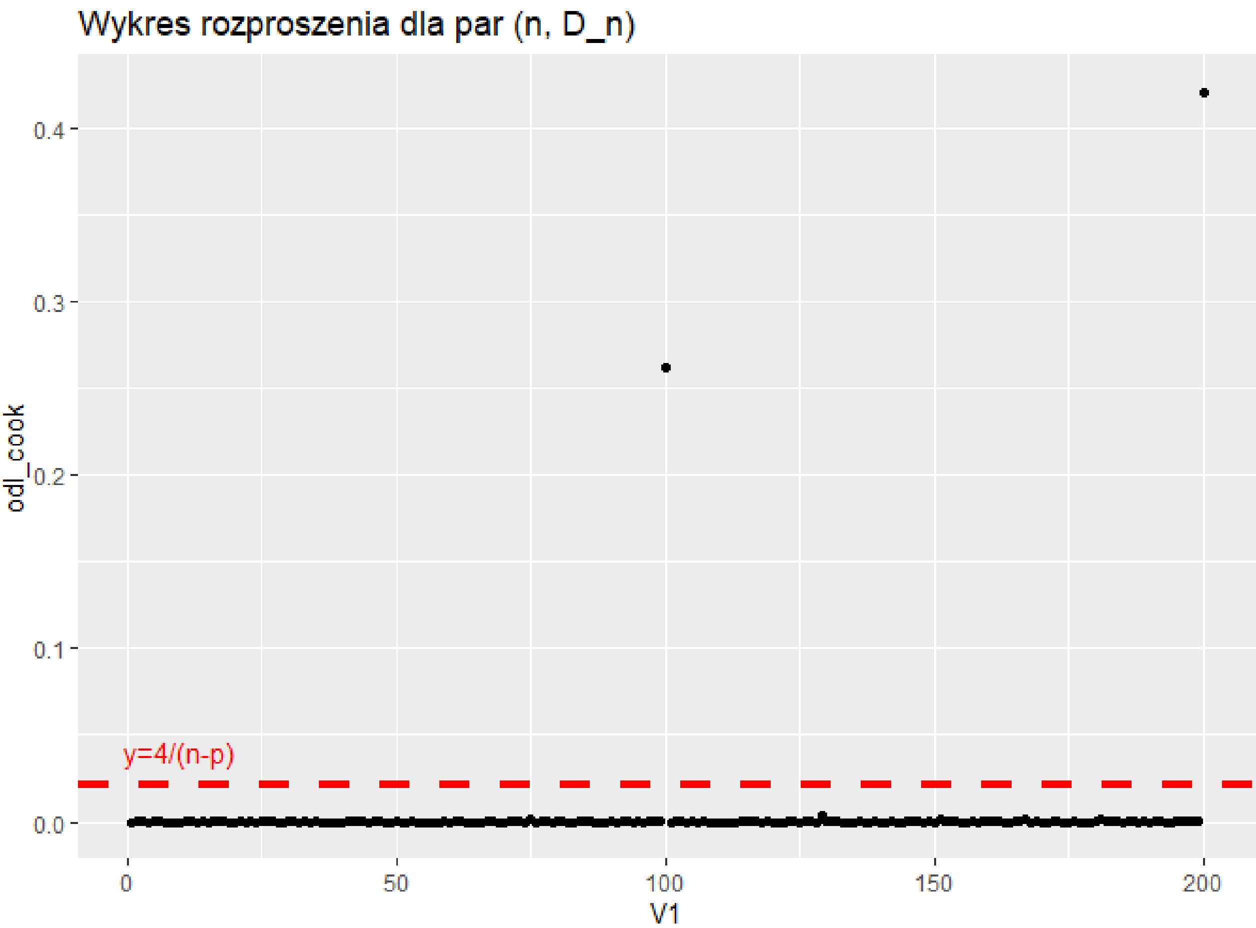
```
p <- sum(hatvalues(model_3)) #ślad macierzy H

#Szukamy obserwacji wp■lywowych ze względu na x
outliers <- hatvalues(model_3) > 3 * mean(hatvalues(model_3))
dane_1[outliers, ] #obserwacja nr 175 jest wp■lywowa ze względu na x

##             X1        X3       X4        X5       X6        X7       X8
## 175 -0.6157619 0.2297211 1.297732 -6.787715 6.700133 -5.188624 11.04227
##            Y
## 175 90.39774
```

Zatem obserwacja nr 175 jest wpływowa ze względu na $\mathbf{x}$.

(b) Odległości Cooke'a - w metodzie tej podejrzewamy, że i-ta obserwacja może być wpływowa, gdy, zgodnie z regułą kciuka, odległość Cooke'a dla i-tej obserwacji $D_i \geq \frac{4}{n-p}$. By ułatwić identyfikację obserwacji wpływowych, wykonamy wykres rozproszenia dla punktów $(1, D_1)$, $(2, D_2)$, $\ldots$, (n, D_n). Do policzenia odlgłości Cooke'a używamy funkcji *cooks.distance*.

```
odl_cook <- cooks.distance(model_3) #liczymy odleg■lości Cooke'a
D_n <- data.frame(cbind(1:length(odl_cook), odl_cook)) #tworzymy data fra-
me
cutoff <- data.frame(x = c(0, nrow(D_n)), y=4/(nrow(D_n)-p),
                     cutoff = factor(4/(nrow(D_n)-p)))

ggplot(data=D_n, mapping=aes(x=V1, y=odl_cook)) +
  geom_point(aes(colour='(n,D_n)'), lwd=1.5, color='black') +
  ggtitle('Wykres rozproszenia dla par (n, D_n)') +
  geom_hline(yintercept =4/(nrow(D_n)-p), color='red', linetype='dashed',
           lwd=1.5) +
  annotate("text",x=10, y=4/(nrow(D_n)-p),vjust=-1, label = "y=4/(n-p)", co-
lor='red')
```

Zatem z rysunku (4) wynika, że obserwacje nr 100 i 200 mogą być wpływowe (ze względu na $\mathbf{x}$ lub y - sprawdzimy to w kolejnym podpunkcie). Aby teraz sprawdzić, czy obserwacje te są rzeczywiście wpływowe, badamy rozbieżność estymatorów współczynników regresji liniowej dopasowanej do modelu danych odpowiednio bez 100. oraz, w drugim przypadku, bez 200. obserwacji oraz z tymi obserwacjami, uzyskanych metodą MNK.



Rysunek 4: Wykres rozproszenia dla (n, D_n) wraz z progiem odcięcia

```
dane_bez_100 <- dane_1[-100,]
dane_bez_200 <- dane_1[-200,]
model_bez_100 <- lm(Y~X1+X3+X4+X5+X6+X7+X8,
                    data=as.data.frame(dane_bez_100))
model_bez_200 <- lm(Y~X1+X3+X4+X5+X6+X7+X8,
                    data=as.data.frame(dane_bez_200))

beta_bez_100 <- model_bez_100$coefficients
beta_bez_200 <- model_bez_200$coefficients
beta <- model_3$coefficients
matrix(c(beta_bez_100, beta), nrow=length(beta), ncol=2) #cię■ko ocenić

##             [,1]       [,2]
## [1,]  1.07046280  2.0984896
## [2,]  0.71980969  0.8752426
## [3,]  1.79121850  2.4418012
## [4,]  4.27879841  4.1011174
## [5,]  0.10255598  0.3270729
## [6,] 10.87923174 10.8285302
## [7,]  0.05745917 -0.0370409
```

```
## [8,]  1.08691547  1.1307813

matrix(c(beta_bez_200, beta), nrow=length(beta), ncol=2) #cię■ko ocenić

##              [,1]       [,2]
## [1,]  1.26057088  2.0984896
## [2,]  0.20036456  0.8752426
## [3,]  2.66079157  2.4418012
## [4,]  3.82150870  4.1011174
## [5,]  0.25401146  0.3270729
## [6,] 10.93469928 10.8285302
## [7,] -0.09314298 -0.0370409
## [8,]  1.04094505  1.1307813
```

Jak możemy zauważyć, z macierzy przedstawiających porównania współczynników regresji z i bez odpowiednich obserwacji, dość ciężko jest ocenić, na ile równania odpowiednich hiperpłaszczyzn różnią się od siebie. Jednak w przypadku niektórych współczynników te rozbieżności są znaczące. W takm przypadku klasyfikujemy badane obserwacje jako **wpływowe**.

(c) Studentyzowane rezydua $r_1, \ldots, r_n$ - przyglądamy się obserwacjom, dla których studentyzowane rezyduum jest, co do modułu, większe, niż 2, ponieważ takie obserwacje są statystycznie istotne. Studentyzowane rezydua wykrywają obserwacje odstające ze względu na y.

```
student_rezydua <- rstudent(model_3) #obliczamy studentyzowane rezydua dla model_3
istotne <-as.numeric(c(rep(0,nrow(dane_1)))) #inicjalizacja wektora na
                                               #indeksy obserwacji istotnie statystycznych
for (k in 1:length(student_rezydua)) {
  if (abs(student_rezydua[k]) >= 2) {
    istotne[k] <- k #gdy obserwacja statystycznie istotna, to przypisujemy indeks k
  }
}
istotne[which(istotne > 0)] #obserwacje 100 i 200 są podejrzane, odstające ze względu na y

## [1] 100 200
```

Zatem obserwacje nr 100 i 200 są odstające ze względu na y. Z poprzedniego podpunktu wiemy także, że są wpływowe.

(d) $DFFTIS_1, \ldots, DFFITS_n$ - zgodnie z regułą kciuka, uznajemy, że i-ta obserwacja jest wpływowa, gdy $DFFITS_i > 2\sqrt{\frac{p}{n}}$. Jako że w naszym przypadku $n = 200$, to wybieramy próg odcięcia równy $2\sqrt{\frac{p}{n}}$, a nie 1.

```
#obserwacje 100 i 200 są wp■lywowe
dane_1[which(dffits(model_3) > 2*sqrt(p/nrow(dane_1))),]
```

```
##             X1       X3       X4        X5       X6        X7        X8        Y
## 100 0.3429141 3.442547 3.695635 -2.722313 9.517476 -2.842297  6.600038 233.2936
## 200 1.5500963 1.569503 7.107258 -2.769175 8.893082 -0.885525 12.178121 241.9999
```

Zatem miary $DFFITS_{100}$ i $DFFITS_{200}$ upewniają nas, że obserwacje nr 100 i 200 są wpływowe.

Wniosek 3 *Obserwacja nr* 175 *jest wpływowa i odstająca ze względu na* $\mathbf{x}$*, więc możemy ją bez większych przeszkód usunąć z danych. Obserwacje nr* 100 *i* 200 *są wpływowe i odstające ze względu na* y*, zatem podejmujemy decyzję o ich usunięciu z danych.*

```
dane_2 <- dane_1[-c(100,175,200),] #usuwamy obserwacje wp■lywowe
```

Uwaga 1 *Zwróćmy uwagę na to, że usunięcie z danych obserwacji nr* 100 *i* 200 *jest* ***nieuzasadnione****, ponieważ wartości te mogą wynikać z natury badanego zjawiska. Jeśli tak jest, to usuwając te obserwacje, tracimy ważne informacje. Problemu tego nie rozwiążemy bez konsultacji z osobą, która skompletowała te dane.*

1.7 Zadanie 6.

Budujemy model regresji liniowej dla *dane_2* i oznaczamy go jako *model_4*.

```
model_4 <- lm(Y~X1+X3+X4+X5+X6+X7+X8, data=dane_2)
```

(a) Wyznaczamy estymator najmniejszych kwadratów $\hat{\beta}$.

```
beta_model_4 <- model_4$coefficients #estymator najmniejszych kwadratów
beta_model_4
```

```
##  (Intercept)           X1           X3           X4           X5           X6
##  0.218971902  0.040256555  2.009059976  3.999296444  0.030359578 10.986713473
##           X7           X8
##  0.001794841  0.996155826
```

(b) Sprawdźmy, czy jakakolwiek zmienna objaśniająca ma liniowy wpływ na zmienną objaśnianą, używając testu F.

```
summary(model_4)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X3 + X4 + X5 + X6 + X7 + X8, data = dane_2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.66945 -0.22631  0.00045  0.22292  0.94691
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.218972   0.185894   1.178   0.2403
## X1           0.040257   0.022199   1.813   0.0713 .
## X3           2.009060   0.023122  86.890   <2e-16 ***
## X4           3.999296   0.013245 301.945   <2e-16 ***
## X5           0.030360   0.024605   1.234   0.2188
## X6          10.986713   0.012306 892.826   <2e-16 ***
## X7           0.001795   0.008878   0.202   0.8400
## X8           0.996156   0.003079 323.512   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3281 on 189 degrees of freedom
## Multiple R-squared:  0.9998,Adjusted R-squared:  0.9998
## F-statistic: 1.451e+05 on 7 and 189 DF,  p-value: < 2.2e-16
```

Z podsumowania odczytujemy, że p-wartość tego testu jest mniejsza, niż $2.2e-16$. Zatem na poziomie istotności $\alpha = 0.05$ odrzucamy hipotezę zerową H_0 na poczet hipotezy H_1, mówiącej o tym, że **istnieje** zmienna objaśniajaca mająca liniowy wpływ na zmienną objaśnianą.

(c) Wyznaczamy R^2 oraz $adjR^2$.

```
R_kwadrat_model_4 <- summary(model_4)$r.squared
adj_R_kwadrat_model_4 <- summary(model_4)$adj.r.squared
R_kwadrat_model_4

## [1] 0.999814

adj_R_kwadrat_model_4

## [1] 0.9998071
```

Wniosek 4 *Model model_4 jest o wiele lepiej dopasowany do danych po usunięciu niektórych zmiennych i obserwacji.*

1.8 Zadanie 7.

Stwórzmy teraz model regresji liniowej, używając kryteriów *stepwise regression*, *forward selection*, *backward elimination* dostępnych w *R* pod postacią funkcji *step*.

- Stepwise regression

```
step(model_4, direction = 'both')

## Start:  AIC=-431.25
## Y ~ X1 + X3 + X4 + X5 + X6 + X7 + X8
##
##         Df Sum of Sq   RSS     AIC
## - X7    1         0    20 -433.21
## - X5    1         0    21 -431.67
## <none>                 20 -431.25
## - X1    1         0    21 -429.85
## - X3    1       813   833  298.06
## - X4    1      9815  9835  784.37
## - X8    1     11267 11287  811.50
## - X6    1     85813 85834 1211.16
##
## Step:  AIC=-433.21
## Y ~ X1 + X3 + X4 + X5 + X6 + X8
##
##         Df Sum of Sq   RSS     AIC
## - X5    1         0    21 -433.65
## <none>                 20 -433.21
## - X1    1         0    21 -431.80
## + X7    1         0    20 -431.25
## - X3    1       830   850  300.12
## - X4    1      9931  9951  784.69
## - X8    1     11309 11330  810.24
## - X6    1     88589 88609 1215.43
##
## Step:  AIC=-433.65
## Y ~ X1 + X3 + X4 + X6 + X8
##
##         Df Sum of Sq   RSS     AIC
## <none>                 21 -433.65
## + X5    1         0    20 -433.21
## - X1    1         0    21 -432.08
## + X7    1         0    21 -431.67
## - X3    1       831   851  298.32
## - X4    1     10057 10077  785.17
## - X8    1     11310 11330  808.25
## - X6    1     90226 90247 1217.04
##
## Call:
```

```
## lm(formula = Y ~ X1 + X3 + X4 + X6 + X8, data = dane_2)
##
## Coefficients:
## (Intercept)           X1           X3           X4           X6           X8
##     0.09124      0.04142      2.00844      4.00132     10.98913      0.99620
```

Sugerowany model to model utworzony ze zmiennych X_1, X_3, X_4, X_6, X_8.

- Forward selection

```
step(model_4, direction = 'forward')

## Start:  AIC=-431.25
## Y ~ X1 + X3 + X4 + X5 + X6 + X7 + X8
##
## Call:
## lm(formula = Y ~ X1 + X3 + X4 + X5 + X6 + X7 + X8, data = dane_2)
##
## Coefficients:
## (Intercept)           X1           X3           X4           X5           X6
##    0.218972     0.040257     2.009060     3.999296     0.030360    10.986713
##          X7           X8
##    0.001795     0.996156
```

Sugerowany model to model początkowy, czyli utworzony ze zmiennych X_1, X_3, $\dots$, X_7, X_8.

- Backward elimination

```
step(model_4, direction = 'backward')

## Start:  AIC=-431.25
## Y ~ X1 + X3 + X4 + X5 + X6 + X7 + X8
##
##         Df Sum of Sq   RSS     AIC
## - X7     1         0    20 -433.21
## - X5     1         0    21 -431.67
## <none>                  20 -431.25
## - X1     1         0    21 -429.85
## - X3     1       813   833  298.06
## - X4     1      9815  9835  784.37
## - X8     1     11267 11287  811.50
## - X6     1     85813 85834 1211.16
##
## Step:  AIC=-433.21
## Y ~ X1 + X3 + X4 + X5 + X6 + X8
##
##         Df Sum of Sq   RSS     AIC
```

```
## - X5    1         0    21 -433.65
## <none>                  20 -433.21
## - X1    1         0    21 -431.80
## - X3    1       830   850  300.12
## - X4    1      9931  9951  784.69
## - X8    1     11309 11330  810.24
## - X6    1     88589 88609 1215.43
##
## Step:  AIC=-433.65
## Y ~ X1 + X3 + X4 + X6 + X8
##
##        Df Sum of Sq   RSS     AIC
## <none>                  21 -433.65
## - X1    1         0    21 -432.08
## - X3    1       831   851  298.32
## - X4    1     10057 10077  785.17
## - X8    1     11310 11330  808.25
## - X6    1     90226 90247 1217.04
##
## Call:
## lm(formula = Y ~ X1 + X3 + X4 + X6 + X8, data = dane_2)
##
## Coefficients:
## (Intercept)           X1           X3           X4           X6           X8
##     0.09124      0.04142      2.00844      4.00132     10.98913      0.99620
```

Sugerowany model to model utworzony ze zmiennych X_1, X_3, X_4, X_6, X_8.

Trzy powyższe metody doprowadziły nas do **dwóch** różnych modeli. Wybierzmy jeden z nich do dalszej analizy, używając współczynnika $adjR^2$ i oznaczmy nowy model jako M.

```
summary(lm(Y~X1+X3+X4+X6+X8, data=dane_2))$adj.r.squared

## [1] 0.9998076

adj_R_kwadrat_model_4

## [1] 0.9998071

M <- lm(Y~X1+X3+X4+X6+X8, data=dane_2)
```

Widzimy, że model utworzony ze zmiennych X_1, X_3, X_4, X_6, X_8 jest minimalnie lepszy, zatem przyjmijmy go za model M.

(a) Wyznaczamy estymator najmniejszych kwadratów dla modelu M.

```
beta_model_M <- M$coefficients
beta_model_M

## (Intercept)          X1          X3          X4          X6          X8
##  0.09124121  0.04141985  2.00843749  4.00132448 10.98913048  0.99620188
```

(b) Widzimy, że p-wartość testu (na poziomie istotności $\alpha = 0.05$) F dla pełnego modelu jest mniejsza, niż $2.2e - 16$. Oznacza to, że hipoteza H_0 (hipoteza H_0 ma postać taką, jak zadaniu 3.) jest fałszywa, czyli istnieje zmienna objaśniająca mająca liniowy wpływ na zmienną objaśnianą.

```
summary(M)

##
## Call:
## lm(formula = Y ~ X1 + X3 + X4 + X6 + X8, data = dane_2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.72629 -0.23005  0.00227  0.21664  0.95883
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.09124    0.14932   0.611    0.542
## X1           0.04142    0.02215   1.870    0.063 .
## X3           2.00844    0.02284  87.955   <2e-16 ***
## X4           4.00132    0.01307 306.018   <2e-16 ***
## X6          10.98913    0.01199 916.604   <2e-16 ***
## X8           0.99620    0.00307 324.518   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3277 on 191 degrees of freedom
## Multiple R-squared:  0.9998,Adjusted R-squared:  0.9998
## F-statistic: 2.037e+05 on 5 and 191 DF,  p-value: < 2.2e-16
```

(c) Niech H_0 oznacza, że $\beta_i = 0, i \in \{1, 3, 4, 6, 8\}$, natomiast H_1, że $\beta_i \neq 0$. Używając statystyki testowej o rozkładzie *t-Studenta* i odczytując odpowiednie p-wartości z podsumowania uzsykanego w poprzednim podpunkcie, wnioskujemy, że na poziomie istotności $\alpha = 0.05$ nie mamy podstaw do odrzucenia hipotezy H_0 dla $i = 1$. Natomiast dla pozostałych indeksów i odrzucamy H_0. Wobec tego usuwamy z modelu M zmienną X_1.

```
M <- lm(Y~X3+X4+X6+X8, data=dane_2)
summary(M)

##
## Call:
## lm(formula = Y ~ X3 + X4 + X6 + X8, data = dane_2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.72257 -0.23236 -0.00116  0.21130  1.03926
##
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.095115   0.150272   0.633    0.528
## X3           2.009134   0.022980  87.430   <2e-16 ***
## X4           4.001633   0.013159 304.096   <2e-16 ***
## X6          10.988080   0.012053 911.617   <2e-16 ***
## X8           0.996408   0.003088 322.703   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3298 on 192 degrees of freedom
## Multiple R-squared:  0.9998,Adjusted R-squared:  0.9998
## F-statistic: 2.513e+05 on 4 and 192 DF,  p-value: < 2.2e-16
```

(d) Wyznaczamy przedziały ufności, na poziomie ufności równym 0.95, dla poszczególnych współczynników regresji, odpowiadającym zmiennym z modelu M, używając funkcji *confint*.

```
PU <- confint(M,level=0.95)
data.frame(PU, M$coefficients)
```

```
##                  X2.5..    X97.5.. M.coefficients
## (Intercept) -0.2012815  0.3915124     0.09511546
## X3           1.9638091  2.0544598     2.00913444
## X4           3.9756784  4.0275884     4.00163341
## X6          10.9643063 11.0118544    10.98808033
## X8           0.9903177  1.0024980     0.99640780
```

Wniosek 5 *Współczynniki regresji odpowiadające zmiennym z modelu M należą do odpowiadających im przedziałów ufności na poziomie ufności równym* 0.95.

(e) Wyznaczamy współczynniki R^2 oraz $adjR^2$ dla modelu M.

```
summary(M)$r.squared

## [1] 0.9998091

summary(M)$adj.r.squared

## [1] 0.9998051
```

Wniosek 6 *Model M jest dobrze dopasowany do danych.*

1.9 Zadanie 8.

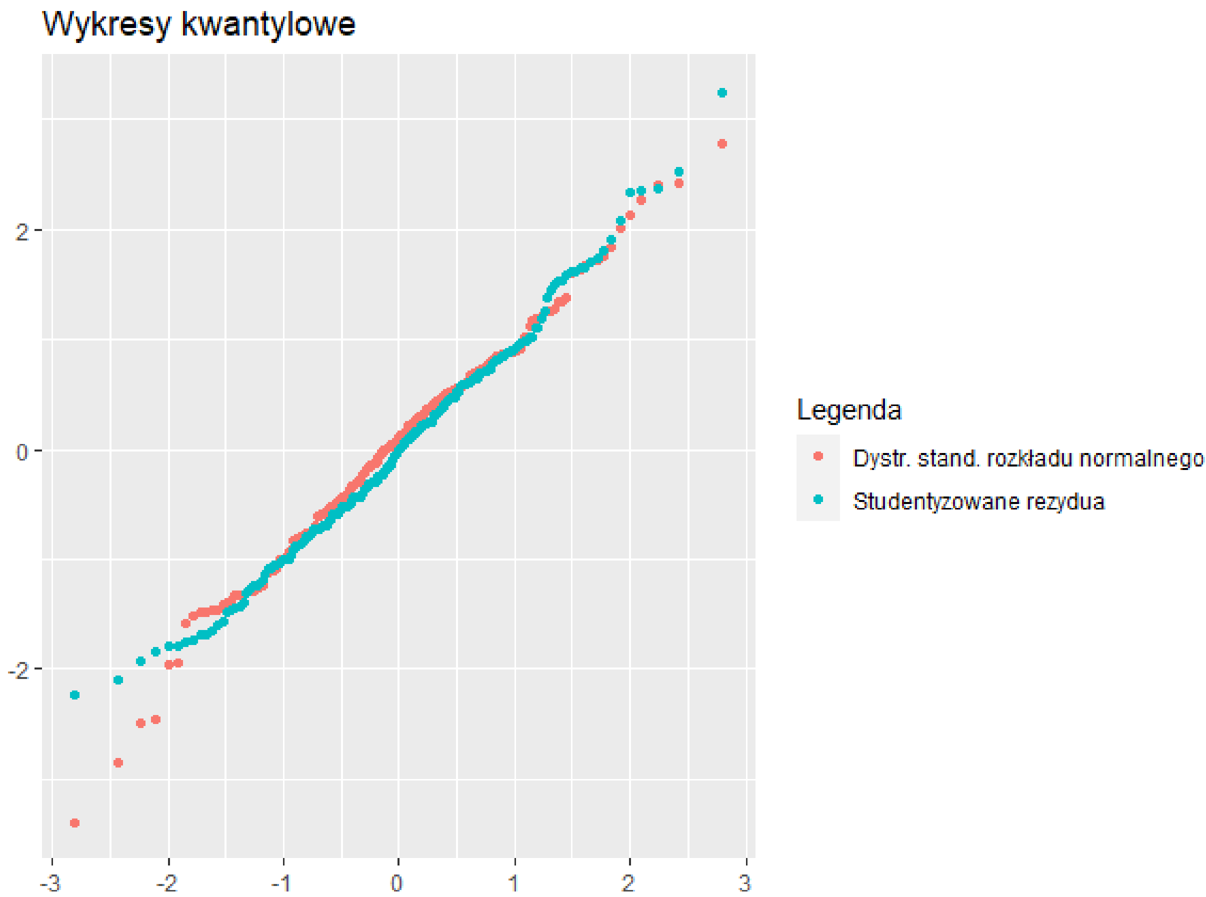
W zadaniu tym przeanalizujemy zachowanie reszt w modelu M, by sprawdzić, czy spełnione są założenia występujace w modelu regresji liniowej (tzn. czy błędy pochodzą z rozkładu normalnego, mają średnią zero i tą samą wariancję). W tym celu wykonamy wykresy:

(a) Wykresy kwantylowe dla reszt.

```
dane_3 <- dane_2[,-c(1,4,6)] #Usuwamy odpowiednie zmienne

#Tworzymy data frame do ggplot
d <- data.frame(group=rep(1:2, each=nrow(dane_3)),
                sample=c(rnorm(nrow(dane_3),0,1),studres(M)))
#Zastępujemy 1 i 2 odpowiednimi nazwami
d$group <- replace(d$group, c(1:197), 'Dystr. stand. rozk■ladu normalnego')
d$group <- replace(d$group, c(198:394), 'Studentyzowane rezydua')

#Rysujemy wykresy kwantylowe
qplot(sample=sample, data=d, color=as.factor(group)) +
  labs(colour='Legenda') +
  ggtitle("Wykresy kwantylowe")
```

Na rysunku nr 5 narysowaliśmy wykres kwantylowy dla studentyzowanych rezyduów oraz próby ze standardowego rozkładu normalnego, ponieważ o ile wektor rezyduów **e** powinien mieć (a przynajmniej tego oczekujemy) wielowymiarowy rozkład normalny, to jednak współrzędne tego wektora (nie są niezależne!) nie mają tej samej wariancji. Aby temu zaradzić, rozważamy studentyzowane rezydua, czyli n-tą współrzędną wektora rezyduów dzielimy przez jego wariancję równą $\hat{\sigma}\sqrt{1-h_{nn}}$, gdzie h_{nn} to element leżący na diagonali macierzy daszkowej $\mathbf{H}$, a $\hat{\sigma}$ to pierwiastek kwadratowy z estymatora wariancji. Obserwacje prowadzą nas do poniższego wniosku.

Wniosek 7 *Ciąg studentyzowanych rezyduów zachowuje się, w bardzo dobrym przybliżeniu, jak ciąg niezależnych zmiennych losowych o standardowym rozkładzie normalnym.*



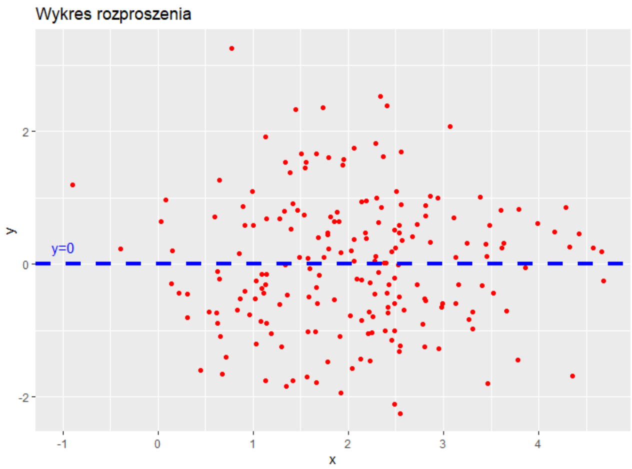
Rysunek 5: Porównanie wykresu kwantylowego dla studentyzowanych reszt oraz próby, o takim samym rozmiarze, ze standardowego rozkładu normalnego

(b) Teraz narysujemy wykresy rozproszenia studentyzowanych reszt względem każdej zmiennej objaśniajacej.

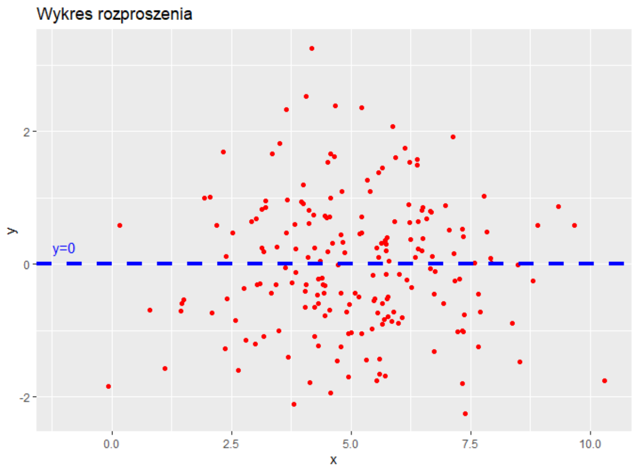
```
#budujemy data frame z dane_3 oraz ze  studentyzowanymi rezyduami
dane_rezyduum <- cbind(dane_3,studres(M))

names(dane_rezyduum)[6] <- 'stud_rezydua' #zmieniamy ich nazwe

dane_rezyduum %>%
  ggplot(aes(x = X3, y = stud_rezydua)) +
  geom_point(colour = "red") +
  ggtitle('Wykres rozproszenia') +
  labs(x='x', y='y') +
  geom_hline(yintercept =0, color='blue', linetype='dashed', lwd=1.5) +
  annotate("text",x=-1, y=0,vjust=-1, label = "y=0", color='blue')

dane_rezyduum %>%
  ggplot(aes(x = X4, y = stud_rezydua)) +
  geom_point(colour = "red") +
  ggtitle('Wykres rozproszenia') +
```

```
  labs(x='x', y='y') +
  geom_hline(yintercept =0, color='blue', linetype='dashed', lwd=1.5) +
  annotate("text",x=-1, y=0,vjust=-1, label = "y=0", color='blue')

dane_rezyduum %>%
  ggplot(aes(x = X6, y = stud_rezydua)) +
  geom_point(colour = "red") +
  ggtitle('Wykres rozproszenia') +
  labs(x='x', y='y') +
  geom_hline(yintercept =0, color='blue', linetype='dashed', lwd=1.5) +
  annotate("text",x=5, y=0,vjust=-1, label = "y=0", color='blue')

dane_rezyduum %>%
  ggplot(aes(x = X8, y = stud_rezydua)) +
  geom_point(colour = "red") +
  ggtitle('Wykres rozproszenia') +
  labs(x='x', y='y') +
  geom_hline(yintercept =0, color='blue', linetype='dashed', lwd=1.5) +
  annotate("text",x=-15, y=0,vjust=-1, label = "y=0", color='blue')
```

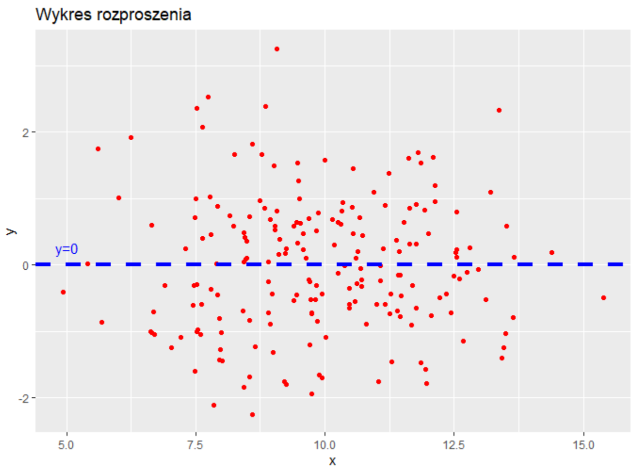
Z rysunków nr 6,7,8,9 wynika, że wykresy rozproszenia dla każdej z prób przypominają wahania losowe wokół osi Ox, bez żadnej zauważalnej tendencji. Ponadto amplituda wahań jest względnie stała.



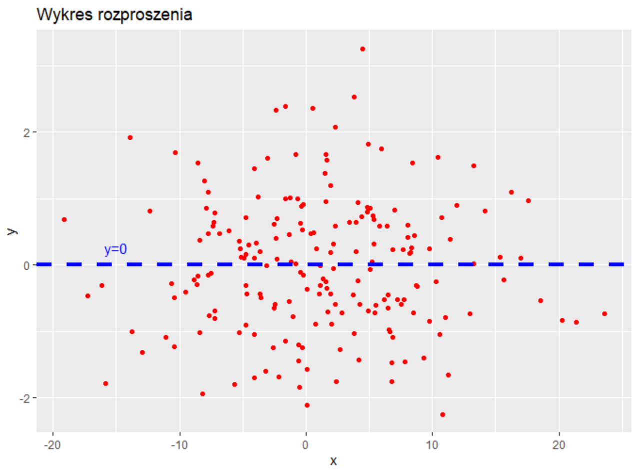
Rysunek 6: Wykres rozproszenia dla próby (x_{n3}, e_n)



Rysunek 7: Wykres rozproszenia dla próby (x_{n4}, e_n)



Rysunek 8: Wykres rozproszenia dla próby (x_{n6}, e_n)

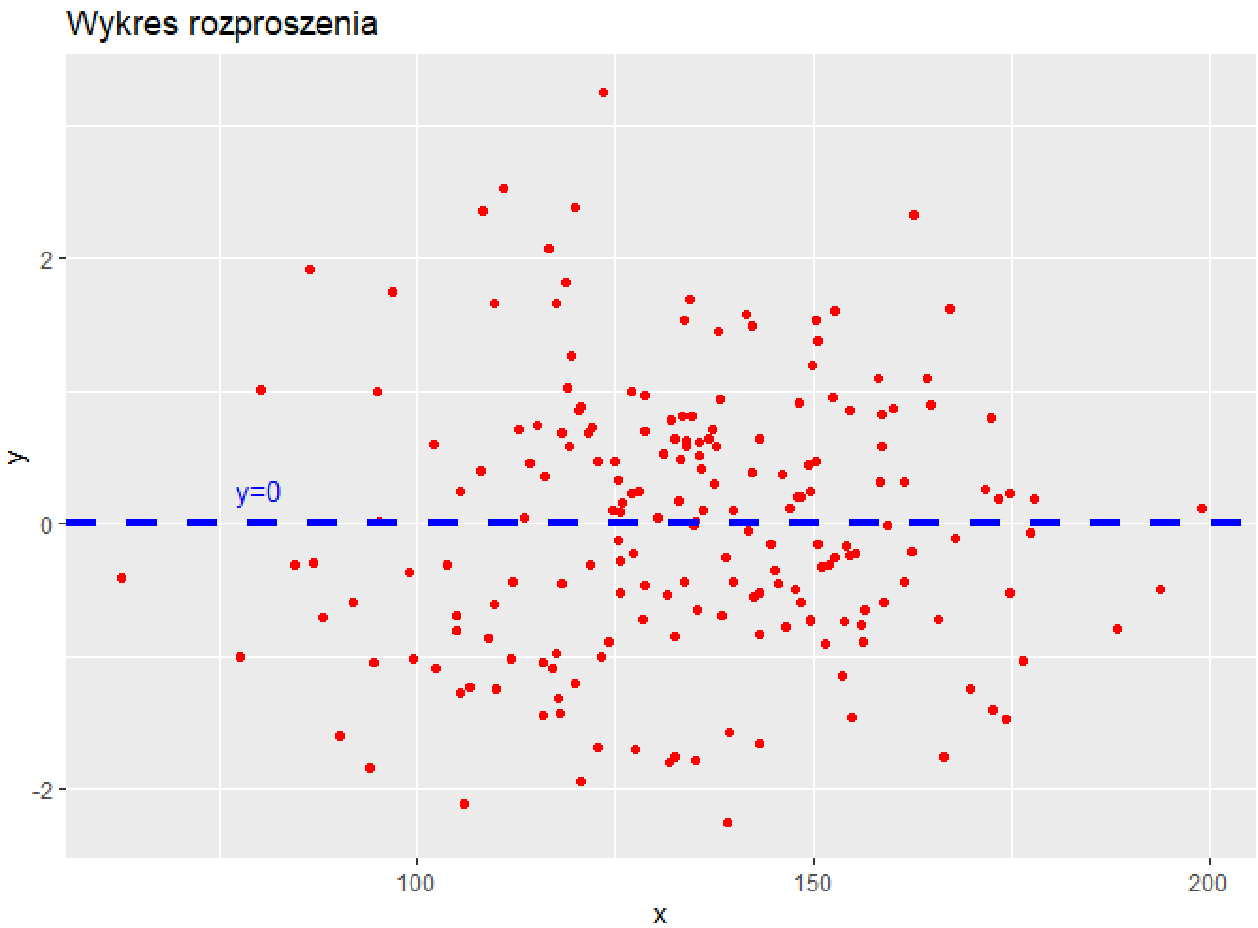


Rysunek 9: Wykres rozproszenia dla próby (x_{n8}, e_n)

(c) Wykresy studentyzowanych reszt względem wartości przewidywanych przez model.

```
dane_rezyduum %>%
  ggplot(aes(x = Y, y = stud_rezydua)) +
  geom_point(colour = "red") +
  ggtitle('Wykres rozproszenia') +
  labs(x='x', y='y') +
  geom_hline(yintercept =0, color='blue', linetype='dashed', lwd=1.5) +
  annotate("text",x=80, y=0,vjust=-1, label = "y=0", color='blue')
```

Z rysunku nr 10 widzimy, że wykres rozproszenia przypomina wahania losowe wokół osi Ox, o względnie stałej wariancji.



Rysunek 10: Wykres rozproszenia dla próby $(\hat{y_n}, e_n)$

Wniosek 8 *Z powyższych rozważań wynika, że dopasowany model jest adekwatny do danych.*

1.10 Zadanie 9.

Wyznaczamy przewidywaną przez model M wartość zmiennej objaśnianej Y, gdy zmienne objaśniające *X3*, *X4*, *X6*, *X8* mają wartości odpowiednio równe $3, 4, 6, 8$, używając funkcji *predict*.

```
predict(M, data.frame(X3=3, X4=4,X6=6,X8=8))

##       1
## 96.0288
```

Przewidywana wartość zmiennej objaśnianej wynosi 96.0288.

2 Część teoretyczna

Zadania wysyłam w formie PDF na eportal z uwagi na problemy z "obrotem" zdjęć.