Modele regresji - raport 1.

Patryk Krukowski(249824)

8 kwietnia 2021

Spis treści

1 Część praktyczna **1**
1.1 Wstęp . . . 1
1.2 Zadanie 1. . . . 1
1.3 Zadanie 2. . . . 3
1.4 Zadanie 3. . . . 4
1.5 Zadanie 4. . . . 4
1.6 Zadanie 5. . . . 4
1.7 Zadanie 6. . . . 5
1.8 Zadanie 7. . . . 5
1.9 Zadanie 8. . . . 6
1.10 Zadanie 9. . . . 7
1.11 Zadanie 10. . . . 8

2 Część teoretyczna **13**

1 Część praktyczna

1.1 Wstęp

Wczytujemy do pakietu statystycznego *R* dane z pliku *lab1.txt* zawierające 100 obserwacji dwóch zmiennych x i y, używając funkcji *read.table*. Nim przystąpimy do wykonania zadań laboratoryjnych, zwróćmy uwagę na to, że dane z pliku, w których część ułamkowa jest niezerowa, jest oddzielona od całości za pomocą przecinka, a nie kropki. Musimy zatem poprawić nasze dane, możemy to zrobić za pomocą poniższego kodu.

```
dane <- read.table('lab1.txt', skip = 1,col.names = c('x','y'))
dane <- data.frame(dane)
dane[,1] <- as.numeric(gsub(",", ".", gsub("\\.","", dane[,1])))
dane[,2] <- as.numeric(gsub(",", ".", gsub("\\.","", dane[,2])))
head(dane)

##        x       y
## 1 4.6149  9.4852
## 2 5.3829 13.1684
## 3 8.6515 19.3157
```

```
## 4 9.7658 21.0094
## 5 4.3198  9.2634
## 6 4.9352  9.0330
```

Zewnętrzne użycie funkcji *gsub* zapewnia zamianę przecinków na kropki, natomiast wewnętrzne odpowiada za zamianę pojawiającego się wówczas znaku podwójnego backslasha między danymi na spację. Używając funkcji *head*, widzimy, że dane są poprawnie określone. Warto również odnotować, że w danych nie występują wartości zakodowane jako *NA* oraz dane są oczywiście kompletne, co ułatwia zdecydowanie poniższe zadania. Przystąpmy zatem do analizy, równocześnie zakładając, że błędy losowe $\epsilon_1, \ldots, \epsilon_n$ są nieobserwowanymi zmiennymi losowymi i.i.d z rozkładu $N\left(0, \sigma^2\right)$ o nieznanej wariancji σ^2.

1.2 Zadanie 1.

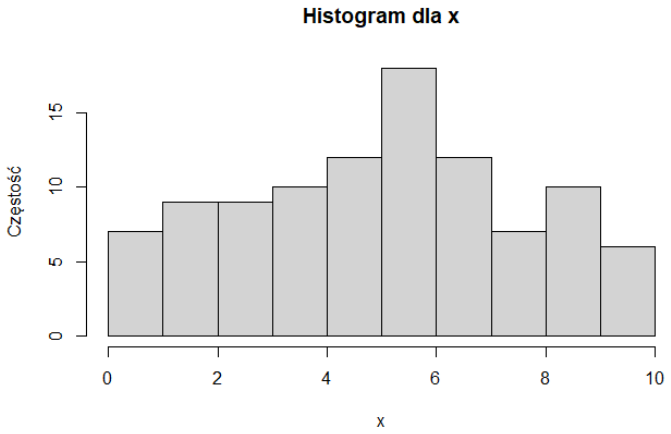
Podstawowe wskaźniki numeryczne uzyskujemy przy pomocy funkcji *summary*.

```
##        x                y         
##  Min.   :0.1052   Min.   : 0.2316  
##  1st Qu.:3.0166   1st Qu.: 7.3297  
##  Median :5.2358   Median :11.4087  
##  Mean   :5.0209   Mean   :11.1351  
##  3rd Qu.:6.8666   3rd Qu.:14.9667  
##  Max.   :9.7658   Max.   :21.0094
```

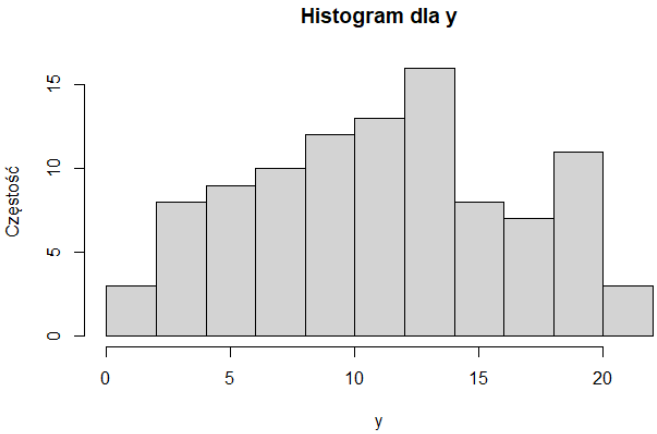
Następnie rysujemy histogramy i box-ploty.

Histogramy dla zmiennych x i y przedstawione na rysunkach (1) i (2) sugerują, że być może mamy do czynienia z rozkładem normalnym, gdybyśmy mieli więcej danych, to wówczas byłoby to nieco łatwiejsze do określenia.

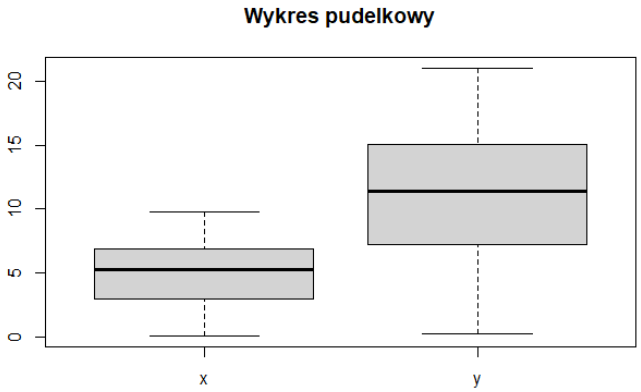
Z rysunku (3) możemy odczytać, że zmienna y charakteryzuje się większą zmiennością, niż zmienna x. Świadczy o tym chociażby większy rozmiar pudełka oraz większy rozrzut wartości odstających ("wąsy" wykresu).



Rysunek 1: Histogram zmiennej x



Rysunek 2: Histogram zmiennej y



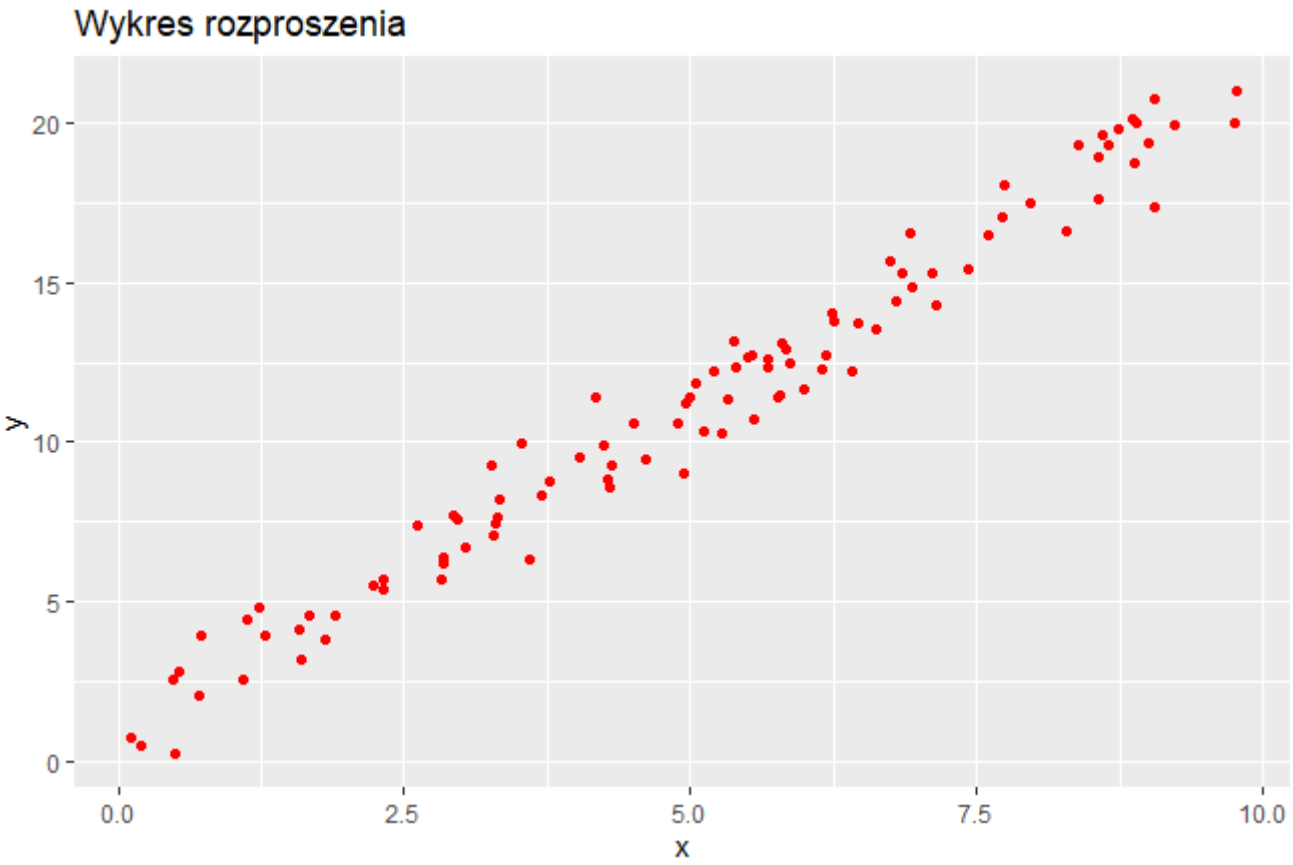
Rysunek 3: Box-plot dla zmiennych x i y

1.3 Zadanie 2.

Wykonujemy wykres rozproszenia zmiennych x i y przy użyciu paczki *ggplot*. Z rysunku (4) odczytujemy, że chmura punktów ma w przybliżeniu charakter liniowy. Policzmy współczynnik korelacji liniowej tych zmiennych.

```
wsp <- cor(dane$x, dane$y)
wsp

## [1] 0.9853143
```

Współczynnik korelacji liniowej jest bliski 1 oraz wykres rozproszenia potwierdza charakter liniowy obu zmiennych, zatem możemy wykorzystać model regresji liniowej w postaci $y = \beta_0 + \beta_1 x + \epsilon$ do opisu zależności między zmiennymi. Współczynniki β_0 i β_1 to nieznane współczynniki regresji, natomiast ϵ to wektor niezależnych zmiennych losowych o średniej 0 i nieznanej wariancji σ^2, tego samego wymiaru, co y.



Rysunek 4: Wykres rozproszenia dla zmiennych x i y

1.4 Zadanie 3.

Wyznaczamy estymatory najmniejszych kwadratów $\hat{\beta}_0$ i $\hat{\beta}_1$ parametrów β_0 i β_1, tworząc w R model regresji przy użyciu funkcji *lm* i odczytując odpowiednie współczynniki. Pojawia się jednak pytanie - dlaczego to są szukane estymatory? Dlatego, że współczynniki obliczone do modelu *lm* są liczone przy użyciu MNK, co możemy sprawdzić w dokumentacji.

```
model <- lm(y~x, data=dane) #wspolczynniki są liczone przy pomocy LSE
model$coefficients

## (Intercept)           x
##   0.8798193   2.0425174
```

Zatem $\hat{\beta}_0 = 0.8798193$ i $\hat{\beta}_1 = 2.0425174$.

1.5 Zadanie 4.

Zauważmy, że suma występująca w definicji estymatora to suma rezyduów podniesionych do kwadratu i podzielona przez 98 (w definicji w mianowniku mamy $n-2$, gdzie $n = 100$). Z kolei tę informację możemy uzyskać przy pomocy funkcji *anova*, tak jak w kodzie poniżej.

```
anova(model)
```

```
## Analysis of Variance Table
##
## Response: y
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## x          1 2769.30 2769.30  3263.3 < 2.2e-16 ***
## Residuals 98   83.17    0.85
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
sigma_hat_square <- anova(model)[2,3]
sigma_hat_square
```

```
## [1] 0.8486302
```

Wartość estymatora to 0.8486302.

1.6 Zadanie 5.

Naszym zadaniem jest zweryfikować (na poziomie istotności 0.05) hipotezę zerową $H_0 : \beta_1 = 0$ przy hipotezie alternatywnej $H_1 : \beta_1 \neq 0$. Użyjemy do tego testu opartego na statystyce *t-Studenta*, do którego dostęp możemy uzyskać poprzez użycie funkcji *summary*.

```
summary(model)
```

```
##
## Call:
## lm(formula = y ~ x, data = dane)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.00391 -0.68670  0.05062  0.55173  2.01107
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.87982    0.20178    4.36 3.21e-05 ***
## x            2.04252    0.03576   57.12  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9212 on 98 degrees of freedom
## Multiple R-squared:  0.9708,Adjusted R-squared:  0.9705
## F-statistic:  3263 on 1 and 98 DF,  p-value: < 2.2e-16
```

Z podsumowania odczytujemy, że p-wartość jest mniejsza, niż $2e^{-16}$. To z kolei jest mniejsze, niż poziom istotności 0.05, zatem powinniśmy odrzucić hipotezę zerową. Na podstawie tych wyników możemy powiedzieć, że model regresji rozpatrywany w naszym przykładzie ma sens, ponieważ, gdyby jednak $\beta_1 = 0$, to w równaniu prostej regresji wystąpiłaby jedynie stała, co nie miałoby u nas sensu. Powinniśmy wówczas jeszcze raz przeprowadzić analizę wyboru modelu do danych.

1.7 Zadanie 6.

Konstruujemy przedział ufności dla β_1 na poziomie ufności 0.99, używając funkcji *confint*. Wybieramy w niej opcję parm = 'x', by do rozważań wybrać β_1.

```
confint(model,parm='x', level=0.99)

##       0.5 %   99.5 %
## x 1.948591 2.136444
```

U nas $\hat{\beta}_1 = 2.04$ w przybliżeniu, zatem estymator ten należy do wyliczonego przedziału ufności $(1.948591, 2.136444)$. Zauważmy, że przedział ten jest dość dobry, tzn. długość tego przedziału jest dość mała.

1.8 Zadanie 7.

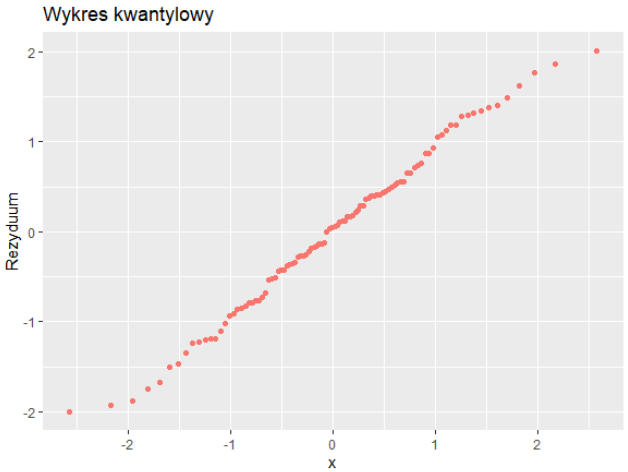
Obliczamy wartość $\hat{Y}(1)$ i znajdujemy przedział ufności na poziomie ufności 0.99 dla $Y(1)$, używając do tego funkcji *predict*.

```
predict(model, data.frame(x = 1), interval = 'confidence', level=0.99)

##        fit      lwr      upr
## 1 2.922337 2.473788 3.370886
```

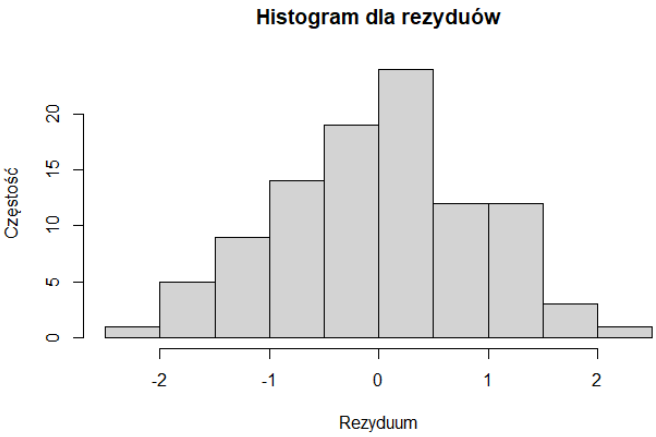
Wartość *fit* to $\hat{Y}(1)$ i wynosi 2.922337. Przedział ufności wynosi $(2.473788, 3.370886)$. Wniosek: prognozowana przez model wartość należy do przedziału ufności.

1.9 Zadanie 8.

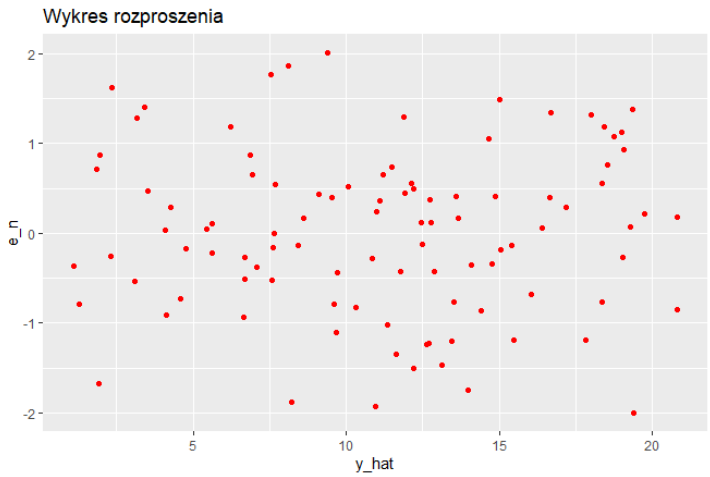
Narysujmy wykres kwantylowy i histogram dla rezyduów oraz wykres rozproszenia dla prób $(\hat{y}_1, e_1), \ldots, (\hat{y}_n, e_n)$ oraz $(x_1, e_1), \ldots, (x_n, e_n)$.



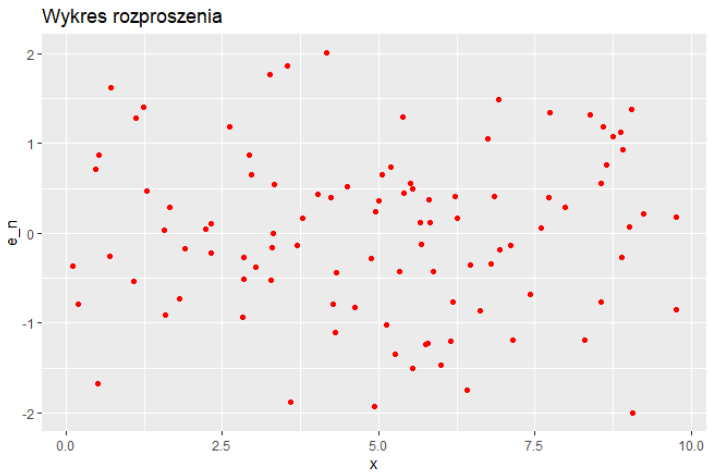
Rysunek 5: Wykres kwantylowy dla rezyduów



Rysunek 6: Histogram dla rezyduów



Rysunek 7: Wykres rozproszenia dla próby $(\hat{y_n}, e_n)$



Rysunek 8: Wykres rozproszenia dla próby (x_n, e_n)

Rysunki (5) i (6) pokazują, iż możemy przyjąć, że rozkład rezyduów jest w przybliżeniu normalny, co w szczególności oznacza, że mamy skończony drugi moment (więc na mocy nierówności Cauchy'ego-Schwarza także pierwszy moment), czyli estymacja współczynnika korelacji Pearsona ma sens. Z zadania 3. teoretycznego wiemy (jeszcze nie wiemy, ale udowodnimy), że jeśli model regresji liniowej poprawnie opisuje zależność między zmiennymi x i y, to współczynnik korelacji próbkowej dla prób z zadania jest równy zeru. Zatem gdyby wykresy rozproszenia dla tych prób miałyby charakter liniowy, to moglibyśmy wówczas stwierdzić, że współczynnik korelacji próbkowej jest różny od zera (przy założeniu, że nie jest też bardzo bliski zeru), a wtedy na mocy twierdzenia model regresji liniowej nie opisuje dobrze zależności między zmiennymi. W naszym przypadku jednak możemy wywnioskować, że współczynnik korelacji próbkowej dla obu prób wynosi 0, natomiast nie możemy z tego wnioskować, że model regresji liniowej dobrze opisuje zależności między zmiennymi, ponieważ nie wiemy, czy twierdzenie działa "w drugą stronę".

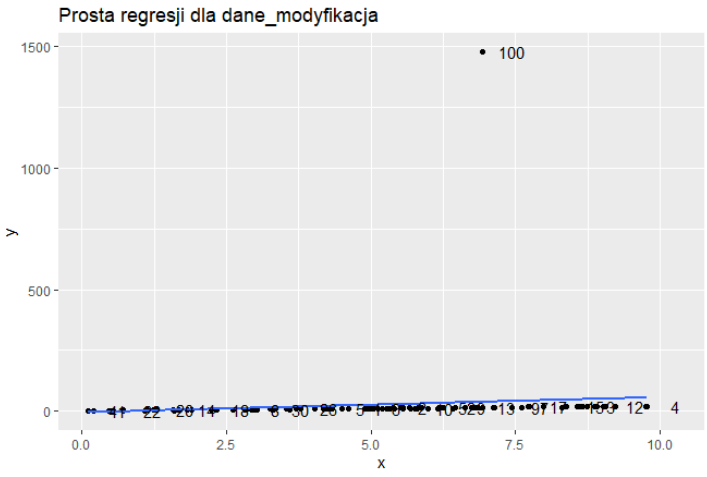
1.10 Zadanie 9.

Zmodyfikujmy dane, zamieniając w ostatnim wierszu wartość zmiennej y na 1480.64 oraz stwórzmy model regresji liniowej dla nowych danych. Oznaczyłem go jako *model_modyfikacja* oraz dane odpowiadające temu modelowi odpowiednio jako *dane_modyfikacja*. Ponadtwo stworzyłem dane o nazwie *dane_new* (pierwotne dane bez ostatniej obserwacji) oraz odpowiadający im model regresji liniowej *model_new*.

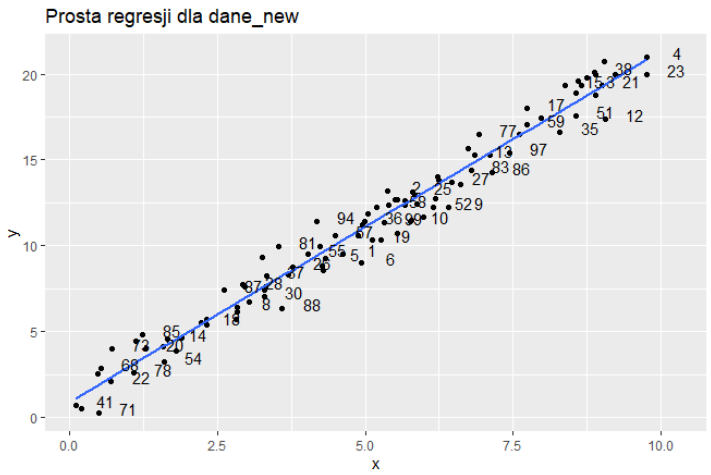
```
dane_modyfikacja <- dane
dane_modyfikacja[100,2] <- 1480.640

dane_new <- dane_modyfikacja[-100,]

model_new <- lm(y~x, data=dane_new)

model_modyfikacja <- lm(y~x, data=dane_modyfikacja)

beta_0_hat_mod <- model_modyfikacja$coefficients[1]
beta_1_hat_mod <- model_modyfikacja$coefficients[2]

beta_0_hat <- 0.8798193
beta_1_hat <- 2.0425174
```

```r
abs(beta_0_hat-beta_0_hat_mod) #Duża różnica

## (Intercept)
##    6.560399

abs(beta_1_hat-beta_1_hat_mod) #Duża różnica

##        x
## 4.225968
```

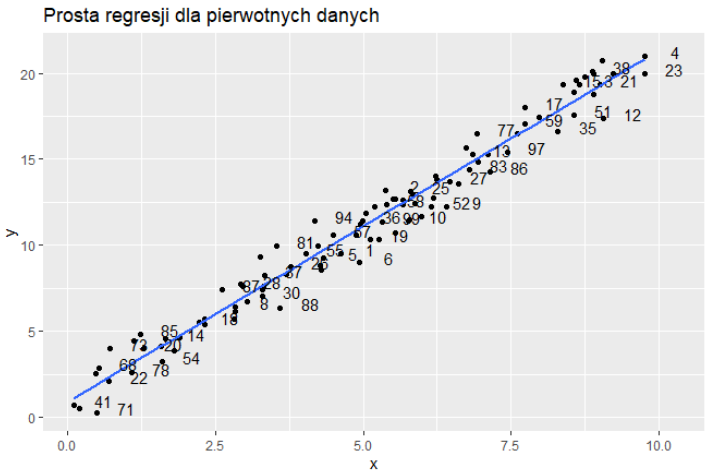
Jak możemy zaobserwować, różnice bezwzględne między estymatorami MNK pochodzącymi z modelu pierwotnego oraz z modelu *model_modyfikacja* są znaczące. Zastanówmy się, czym może to być spowodowane. Narysujmy proste regresji liniowej. Możemy to zrobić, ponieważ między danymi wciąż jest zależność liniowa, z tego samego względu powyższe modele mają sens. Spójrzmy na poniższe wykresy, gdzie punkty są ponumerowane w celu ułatwienia analizy.



Rysunek 9: Wykres prostej regresji dla *dane_modyfikacja*



Rysunek 10: Wykres prostej regresji dla *dane_new*



Rysunek 11: Wykres prostej regresji dla pierwotnych danych

Z rysunków (9), (10), (11) możemy wywnioskować, że obserwacja (6.9347,1480.64) **jest wpływowa**,

ponieważ prosta regresji odchyla się w stronę ostatniej, czyli setnej, obserwacji. To właśnie jest powodem, dla którego wartości estymatorów tak bardzo się różnią. Prosta regresji ulega odchyleniu, ponieważ zmieniają się jej współczynnik kierunkowy oraz wyraz wolny.

Wniosek: Estymatory uzyskane metodą MNK są wrażliwe na obserwacje wpływowe.

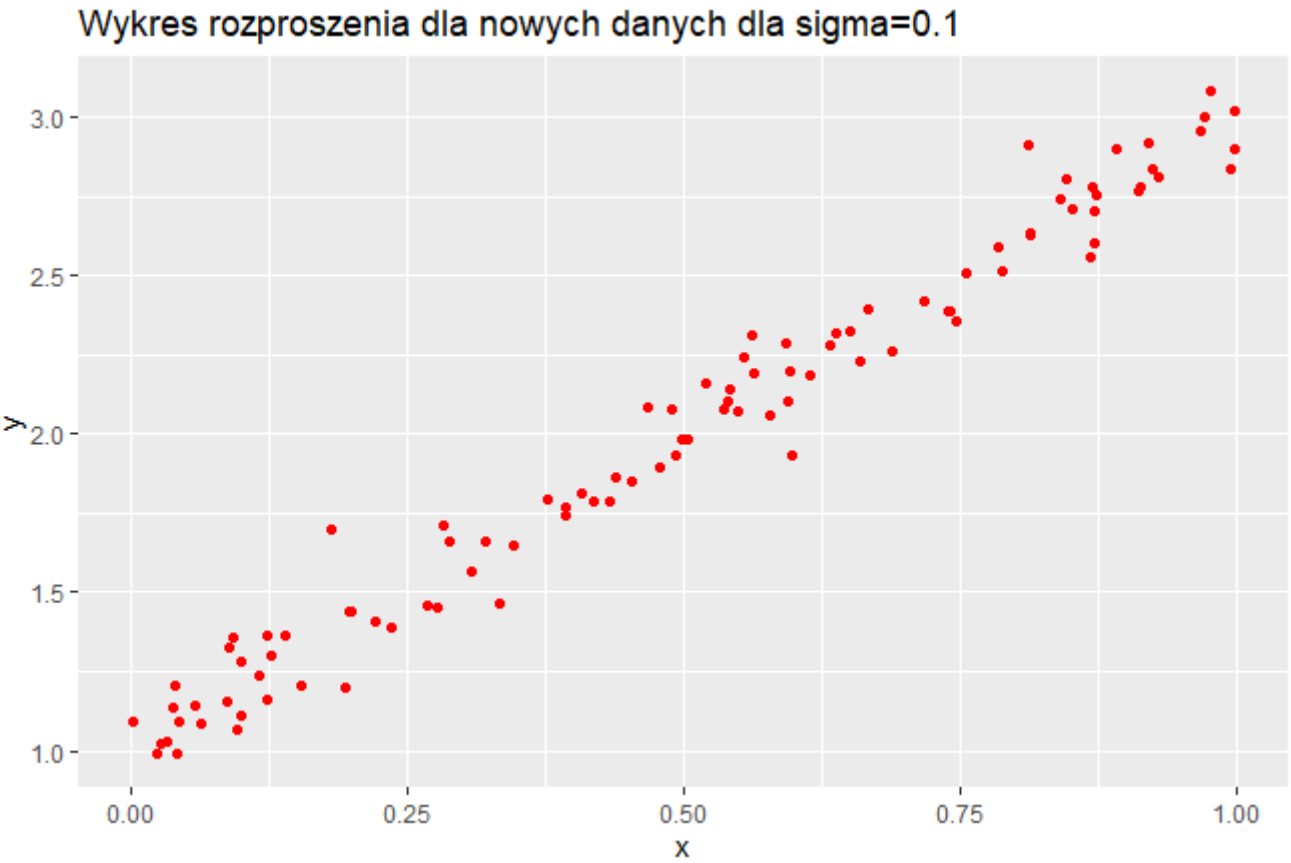
1.11 Zadanie 10.

Generujemy $x_1, x_2, \ldots, x_{100}$ iid z rozkładu $U(0,1)$, $\epsilon_1, \epsilon_2, \ldots, \epsilon_{100}$ iid z rozkładu $N\left(0, \sigma^2\right)$. Niech $\beta_0 = 1$, $\beta_1 = 2$ oraz:

(a) Generujemy $y_1, \ldots, y_{100}$, takie jak w poleceniu

```
x <- runif(100,0,1)
epsilon <- rnorm(100,0,0.1)
beta_0 <- 1
beta_1 <- 2

y <- c()

for (k in c(1:100)) {
y[k] <- beta_0 +beta_1*x[k] + epsilon[k]
}
```

(b) Wykonujemy wykres rozproszenia - chmura punktów ma charakter liniowy



Rysunek 12: Wykres rozproszenia dla nowych danych dla $\sigma = 0.1$

(c) Wyznaczamy estymatory MNK

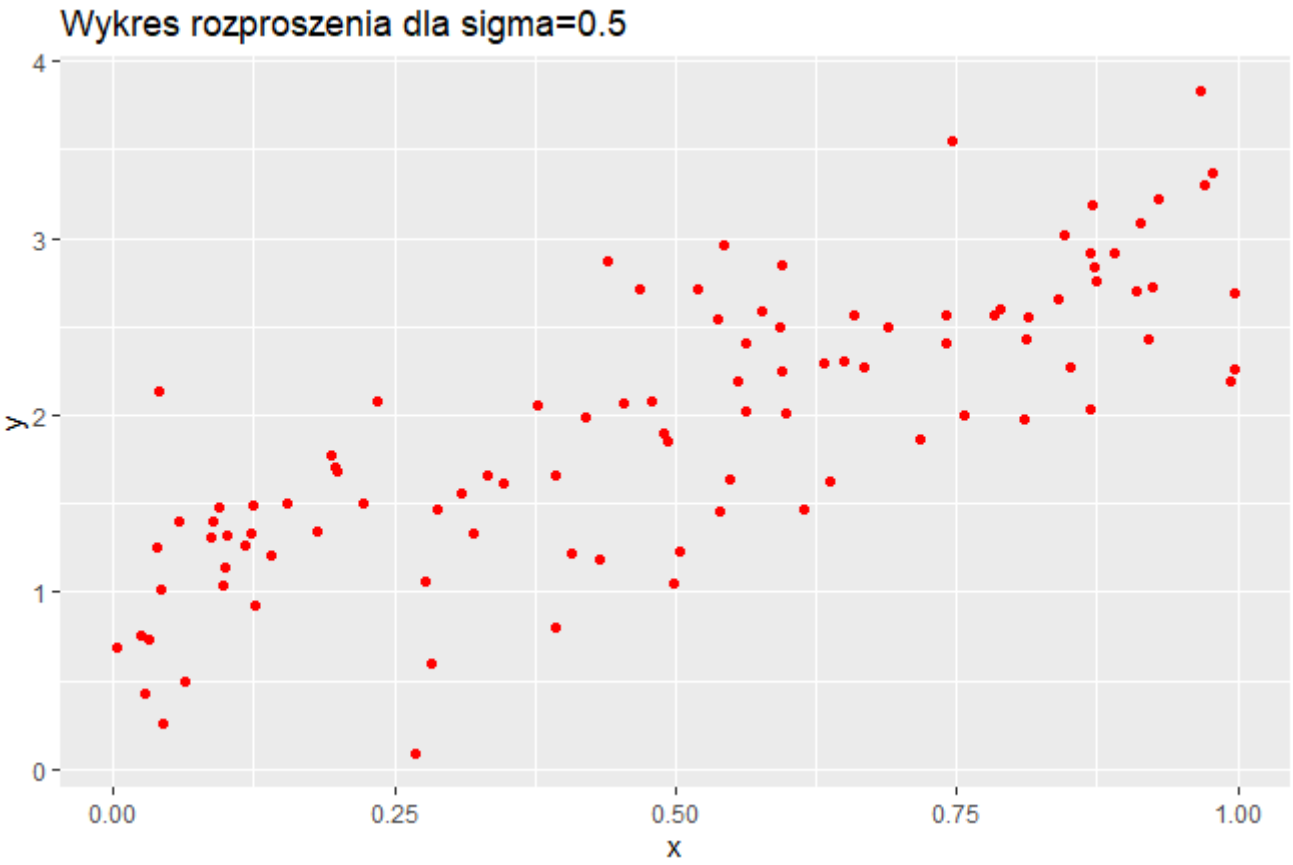
```
dane_1 <- as.data.frame(cbind(x,y))
model_1 <- lm(y~x, data=dane_1)
beta_0_hat_1 <- model_1$coefficients[1]
beta_1_hat_1 <- model_1$coefficients[2]
R_kwadrat <- summary(model)$r.squared #potrzebne do (d)
R_kwadrat

## [1] 0.9708442

abs(beta_0-beta_0_hat_1)

##  (Intercept)
## 9.033308e-05

abs(beta_1-beta_1_hat_1)

##           x
## 0.0002273477
```

Widzimy, że różnica bezwględna między β_0 a $\hat{\beta}_0$ oraz między β_1 a $\hat{\beta}_1$ jest znikoma.

(d) Powtórzmy obliczenia dla $\sigma = 0.5$ oraz $\sigma = 1$. Poniżej znajduje się kod programu, który to wykonuje oraz wykresy rozproszenia. Ostateczne wyniki i porównania przedstawione są przy pomocy tabeli.

- $\sigma = 0.5$; słabszy charakter liniowy, niż w poprzednim przypadku

```
#sigma = 0.5
epsilon_2 <- rnorm(100,0,0.5)
y_2 <- c()

for (k in c(1:100)) {
  y_2[k] <- beta_0 +beta_1*x[k] + epsilon_2[k]
}

dane_2 <- as.data.frame(cbind(x,y_2))
model_2 <- lm(y_2~x, data=dane_2)
beta_0_hat_2 <- model_2$coefficients[1]
beta_1_hat_2 <- model_2$coefficients[2]
abs(beta_0-beta_0_hat_2)

## (Intercept)
##  0.05857625

abs(beta_1-beta_1_hat_2)

##           x
## 0.03944848

R_kwadrat_2 <- summary(model_2)$r.squared
R_kwadrat_2

## [1] 0.5797511
```

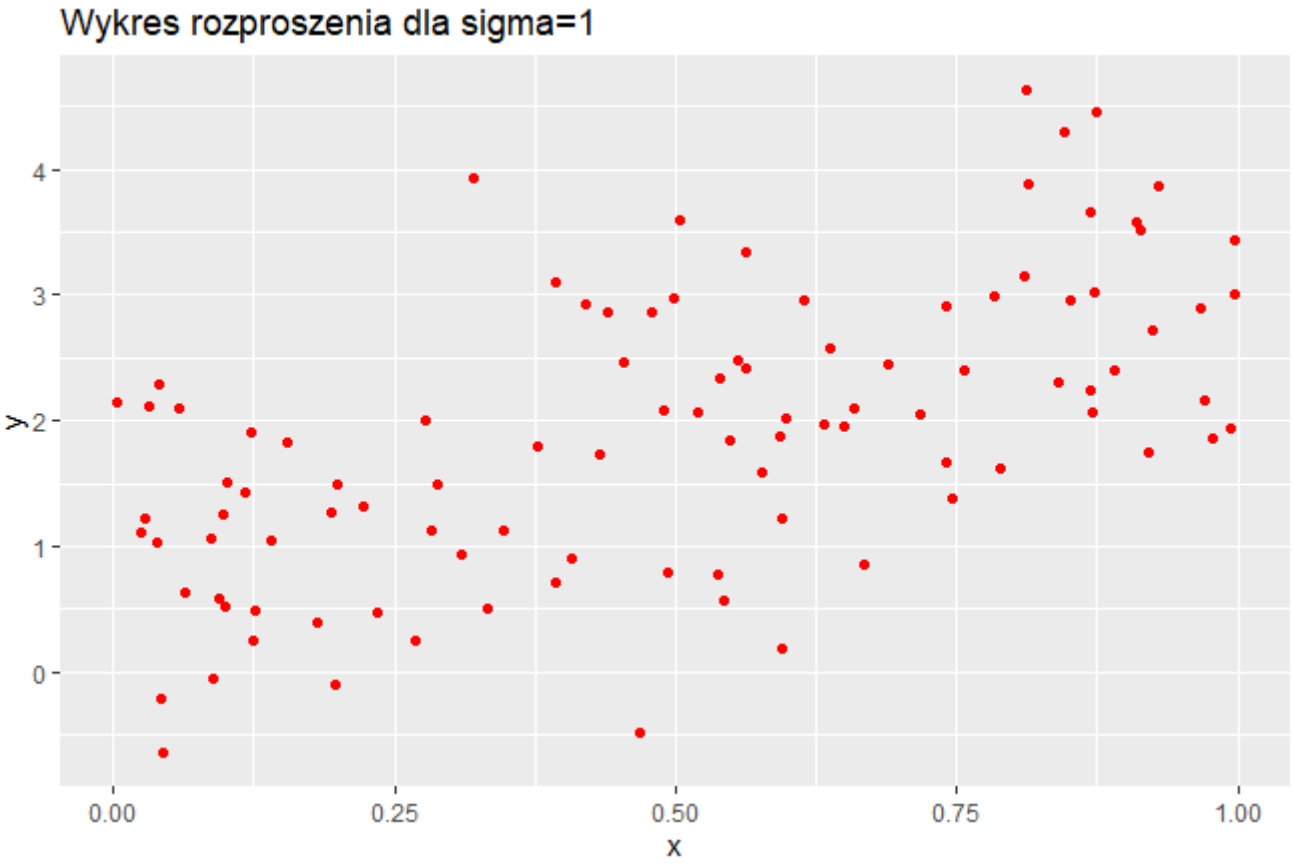


Rysunek 13: Wykres rozproszenia dla nowych danych dla $\sigma = 0.5$

- $\sigma = 1$; bardzo słaby charakter liniowy.

```
#sigma=1


epsilon_3 <- rnorm(100,0,1)
y_3 <- c()

for (k in c(1:100)) {
  y_3[k] <- beta_0 +beta_1*x[k] + epsilon_3[k]
}

dane_3 <- as.data.frame(cbind(x,y_3))
model_3 <- lm(y_3~x, data=dane_3)

beta_0_hat_3 <- model_3$coefficients[1]
beta_1_hat_3 <- model_3$coefficients[2]

abs(beta_0-beta_0_hat_3)

## (Intercept)
##   0.3290588
```

```
abs(beta_1-beta_1_hat_3)

##         x
## 0.5401501

R_kwadrat_3 <- summary(model_3)$r.squared
R_kwadrat_3

## [1] 0.310221
```



Rysunek 14: Wykres rozproszenia dla nowych danych dla $\sigma = 1$

Przedstawmy wyniki analizy w przejrzysty sposób.

σ	0.1	0.5	1
R^2	0.971	0.58	0.31
$\mid \beta_0 - \hat{\beta}_0 \mid$	0.00009	0.059	0.329
$\mid \beta_1 - \hat{\beta}_1 \mid$	0.0002	0.04	0.54

Tabela 1: Porównanie R^2 oraz estymatorów MNK dla różnych wartości σ

Przypomnijmy, że R^2 jest współczynnikiem determinacji, służącym jako miara jakości dopasowania modelu do danych. Mówi on o tym, jaki procent jednej zmiennej wyjaśnia zmienność drugiej zmiennej. Jak widzimy w tabeli (1), im mniejsza wartość σ, tym model jest słabiej dopasowany do danych oraz zmniejsza się precyzja estymatorów.

2 Część teoretyczna

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