lab06-Regresja

Jakub Bryl

25 11 2019

Wczytanie Danych

Dane pochodzą z laboratorium o wizualizacji danych od dr.Ruska. Opisują zależności przejechanych km, cen paliwa, licznika i czy tankowanie było do pełna (zmienna kategoryczna). Wybrałem te dane ponieważ po lab02 zacząłem próbować rzeczy związane z regresją ale nie dokończYłem ich, dlatego wykorzystując wiedzę z ostaniego laboratorium chciałem jeszcze raz użyć tych danych.

```
data <- read_csv("Samochod.csv")%>%
mutate(Data.tankowania=as.Date(Data.tankowania))
```

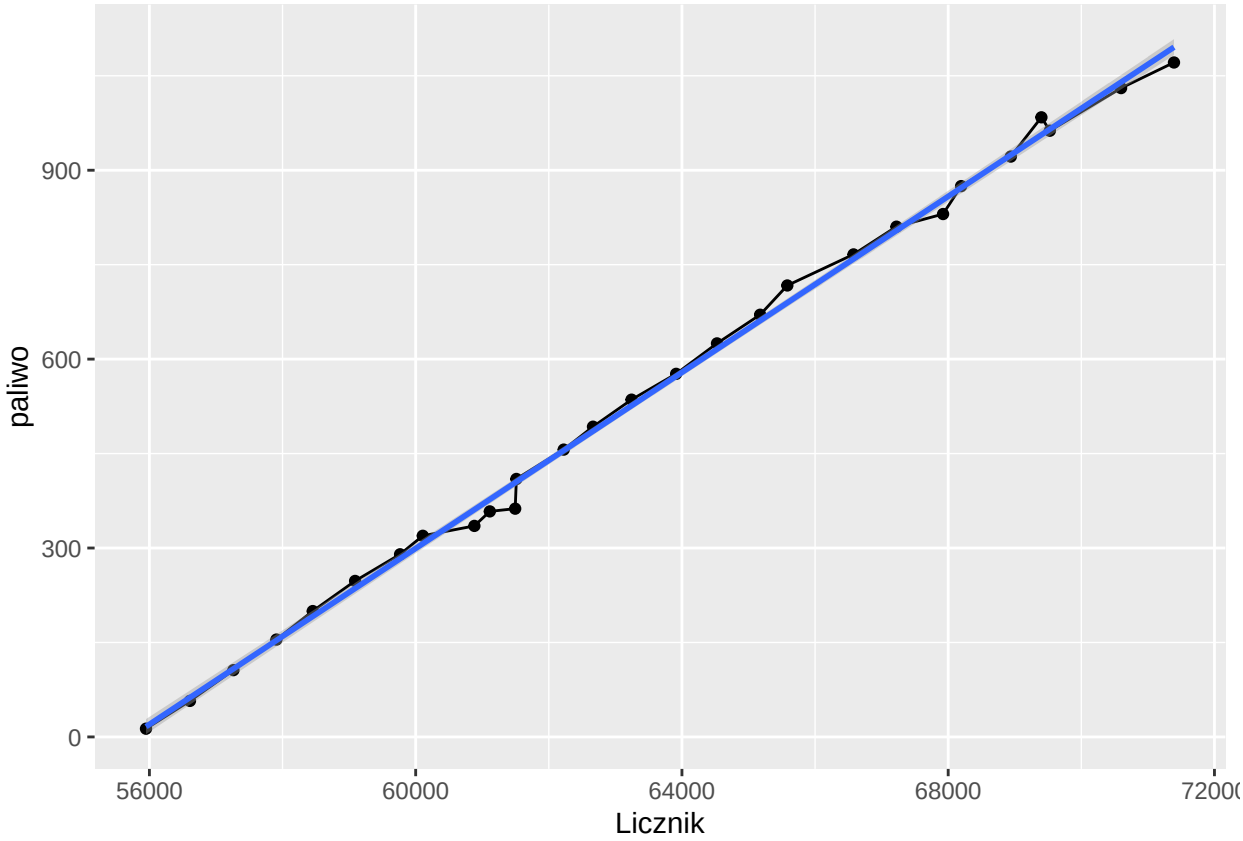
```
## Parsed with column specification:
## cols(
##   `Sygnatura czasowa` = col_datetime(format = ""),
##   Licznik = col_double(),
##   Cena.tankowania = col_double(),
##   Cena.jednostkowa = col_double(),
##   Do.pełna = col_character(),
##   Data.tankowania = col_date(format = "")
## )
```

```
print(data$Data.tankowania)
```

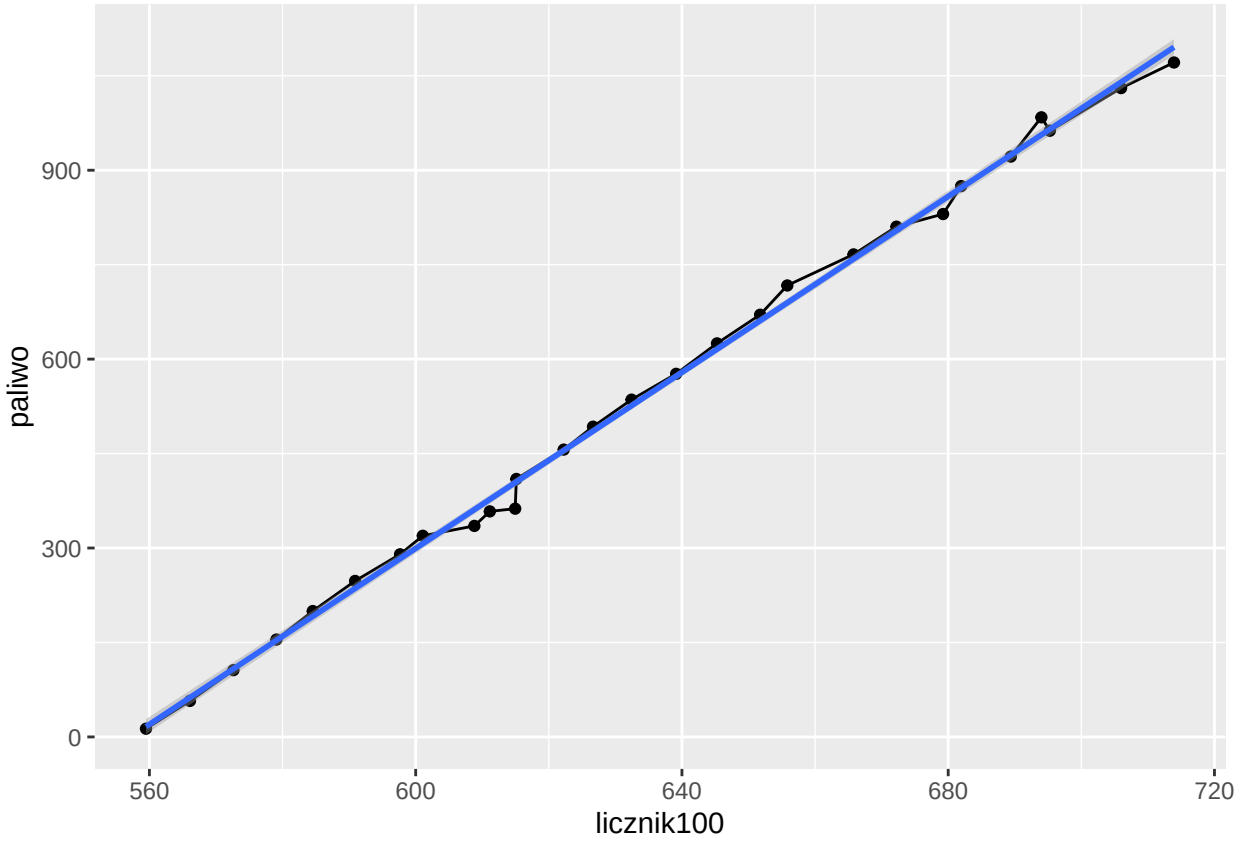
```
##  [1] "2015-06-13" "2015-07-23" "2015-09-07" "2015-11-10" "2015-12-23"
##  [6] "2016-01-29" "2016-03-05" "2016-04-30" "2016-07-31" "2016-08-15"
## [11] "2016-09-03" "2016-09-04" "2016-10-10" "2016-12-22" "2017-02-23"
## [16] "2017-06-03" "2017-07-03" "2017-08-03" "2017-09-09" "2017-11-10"
## [21] "2018-01-13" "2018-05-04" "2018-07-21" "2018-10-13" "2019-09-21"
## [26] "2019-06-15" "2019-02-10" "2019-10-15"
```

Dopasowanie regresji liniowej - ggplot:

```
data %>%arrange(Data.tankowania)%>%mutate(paliwo=cumsum(Cena.tankowania/Cena.jednostkowa))%>%
  ggplot(data=.,aes(x=Licznik,y=paliwo))+geom_point()+geom_line()+stat_smooth(method='lm')
```




```
# Dodanie dodatkowych zmiennych (mutate).
data%>%arrange(Data.tankowania)%>%mutate(paliwo=cumsum(Cena.tankowania/Cena.jednostkowa))%>%
  mutate(licznik100=Licznik/100)->data.100

# Nowy wykres do dopasowania regresji
plot1<-ggplot(data=data.100,aes(x=licznik100,y=paliwo))+
  geom_point()+geom_line()+stat_smooth(method='lm')

ggplot(data=data.100,aes(x=licznik100,y=paliwo))+geom_point()+
  geom_line()+stat_smooth(method='lm')
```



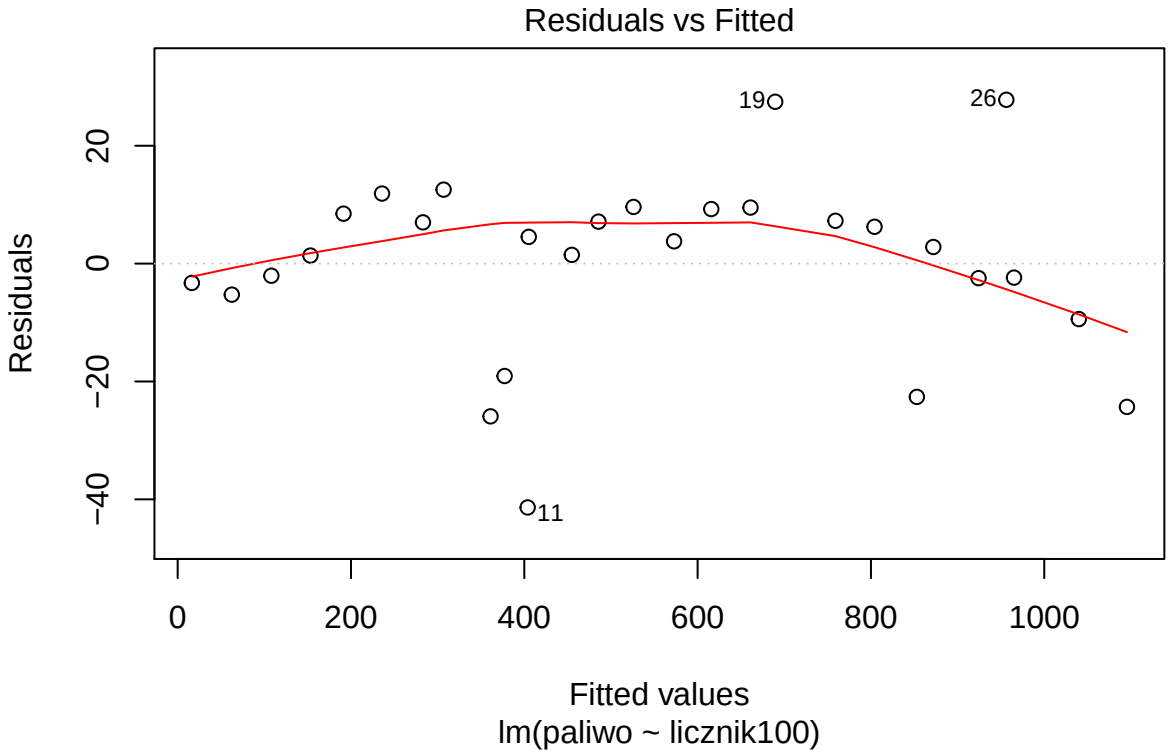
Dopasowanie regresji liniowej - manualnie:

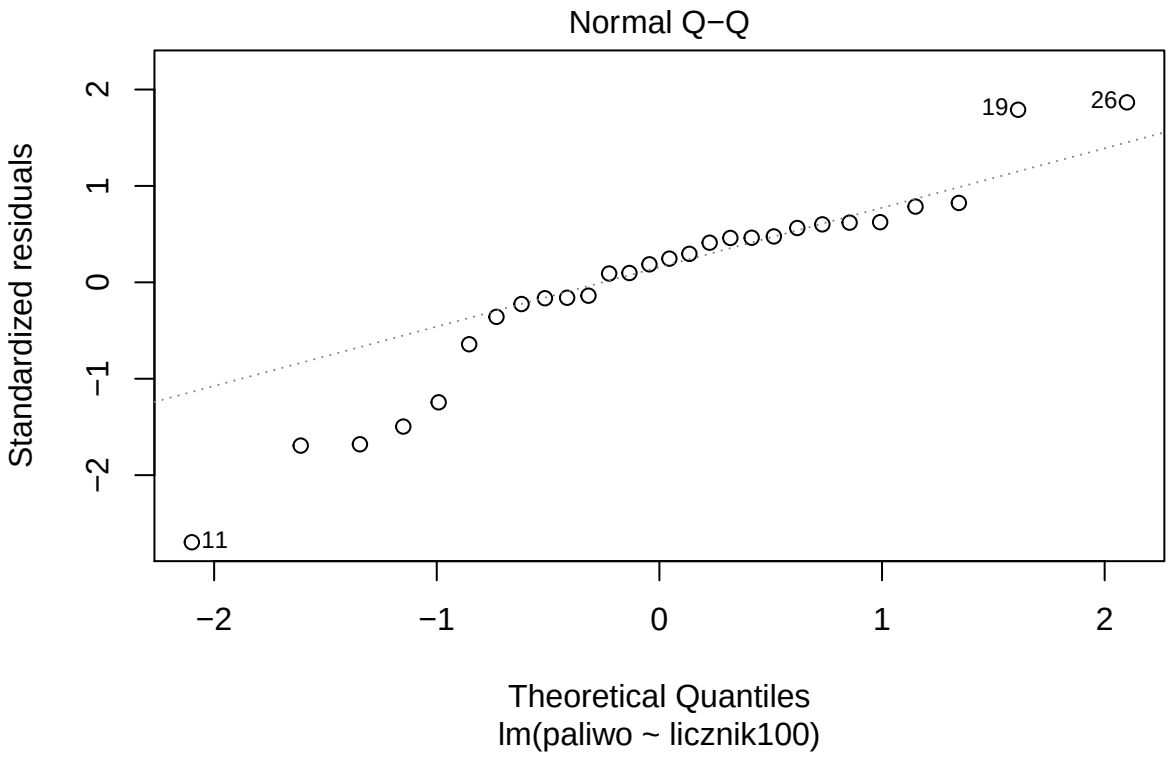
```
#Model okresla zaleznosc miedzy paliwem w litrach a licznikiem na 100km.
model <- lm(paliwo~licznik100, data=data.100)
summary(model)
```

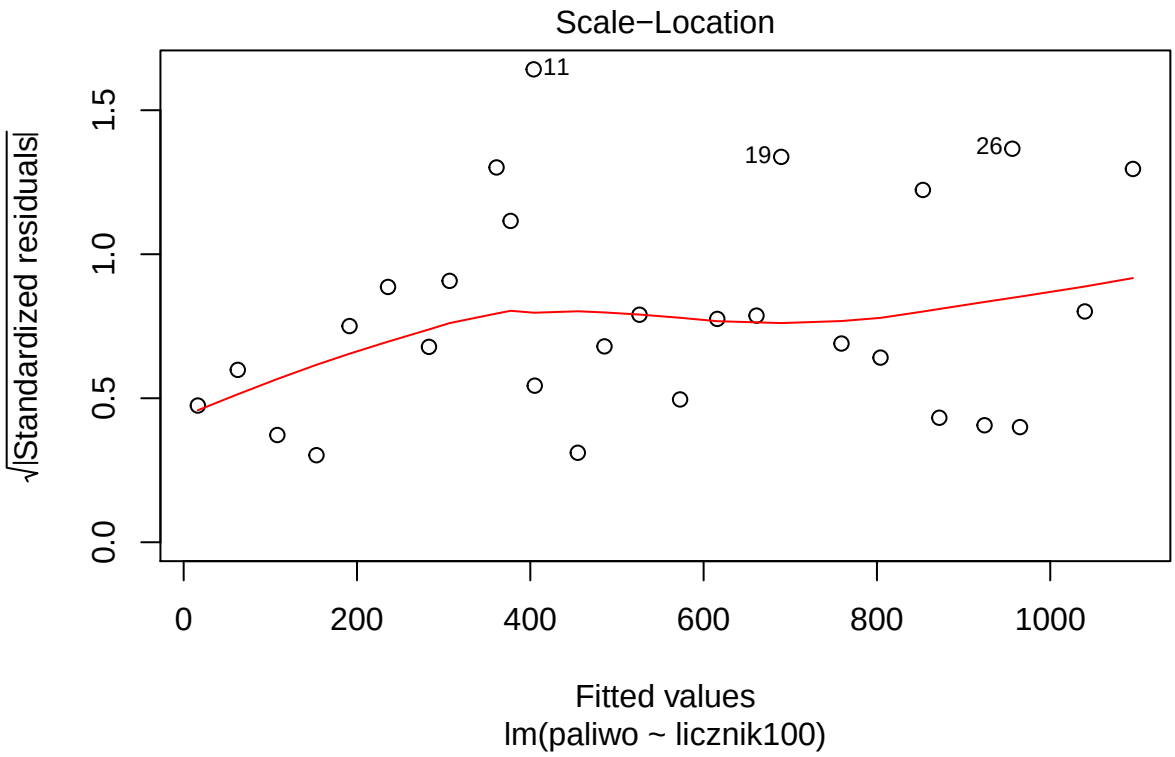
```
##
## Call:
## lm(formula = paliwo ~ licznik100, data = data.100)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -41.376  -3.778   3.302   8.667  27.798
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.893e+03  4.238e+01  -91.86   <2e-16 ***
## licznik100   6.987e+00  6.661e-02  104.91   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.69 on 26 degrees of freedom
## Multiple R-squared:  0.9976, Adjusted R-squared:  0.9976
```

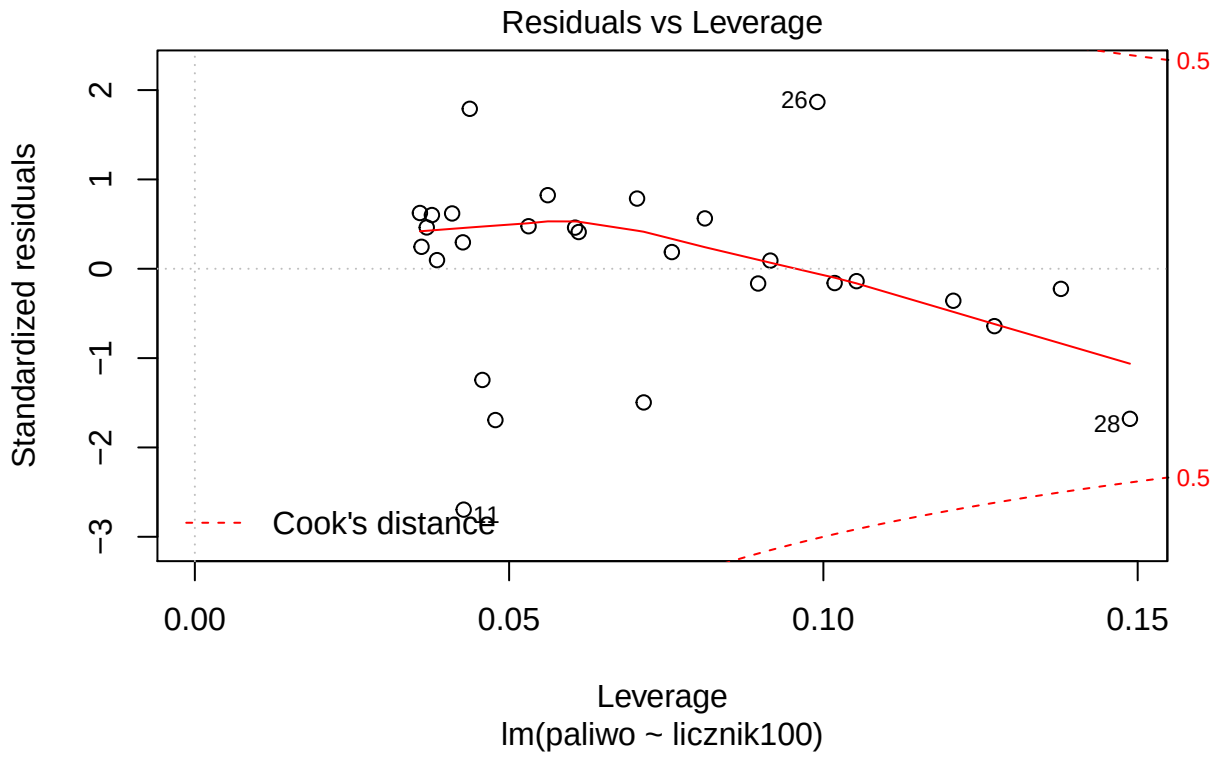
```
## F-statistic: 1.101e+04 on 1 and 26 DF,  p-value: < 2.2e-16
```

```r
#Wyliczenie predykcji i dodanie wartości jako dodatkowej kolumny
data.100$prediction = predict(model, newdata = data.100)

#Obliczenie epsilon'u - roznicy miedzy predykcja a cana paliwa
data.100$epsilon <- data.100$paliwo - data.100$prediction

plot(model)
```

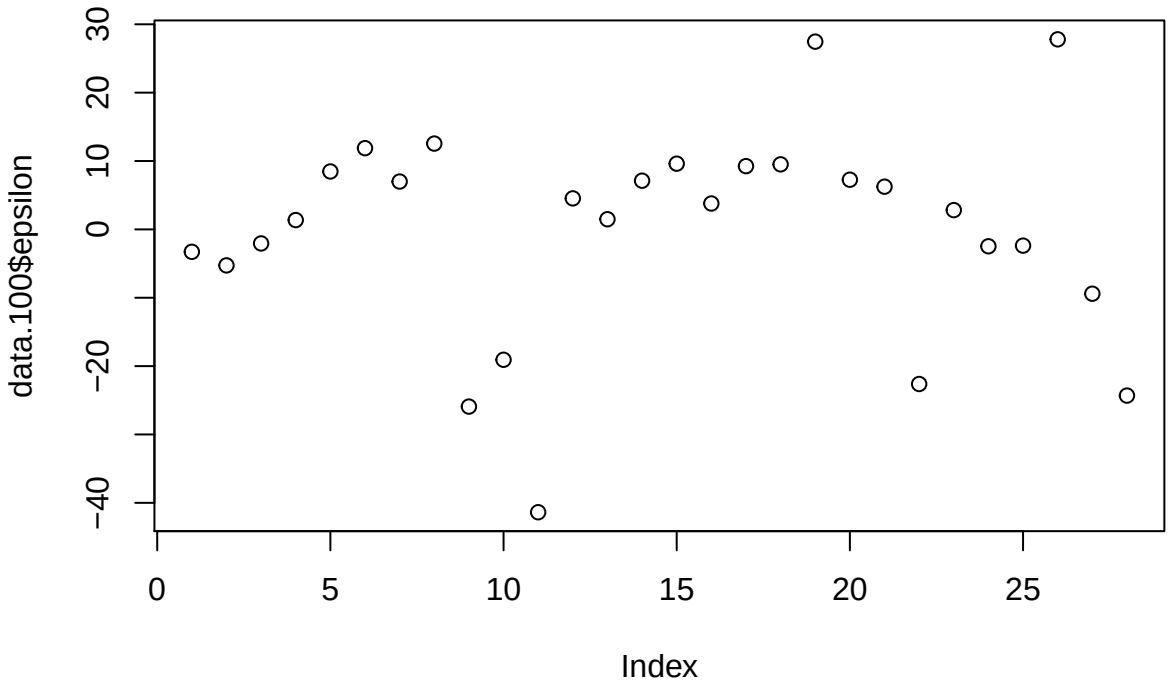



Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(paliwo ~ licznik100)
19
26
11
−2
−1
0
1
2


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
11
19
26
0
200
400
600
800
1000
Fitted values
lm(paliwo ~ licznik100)

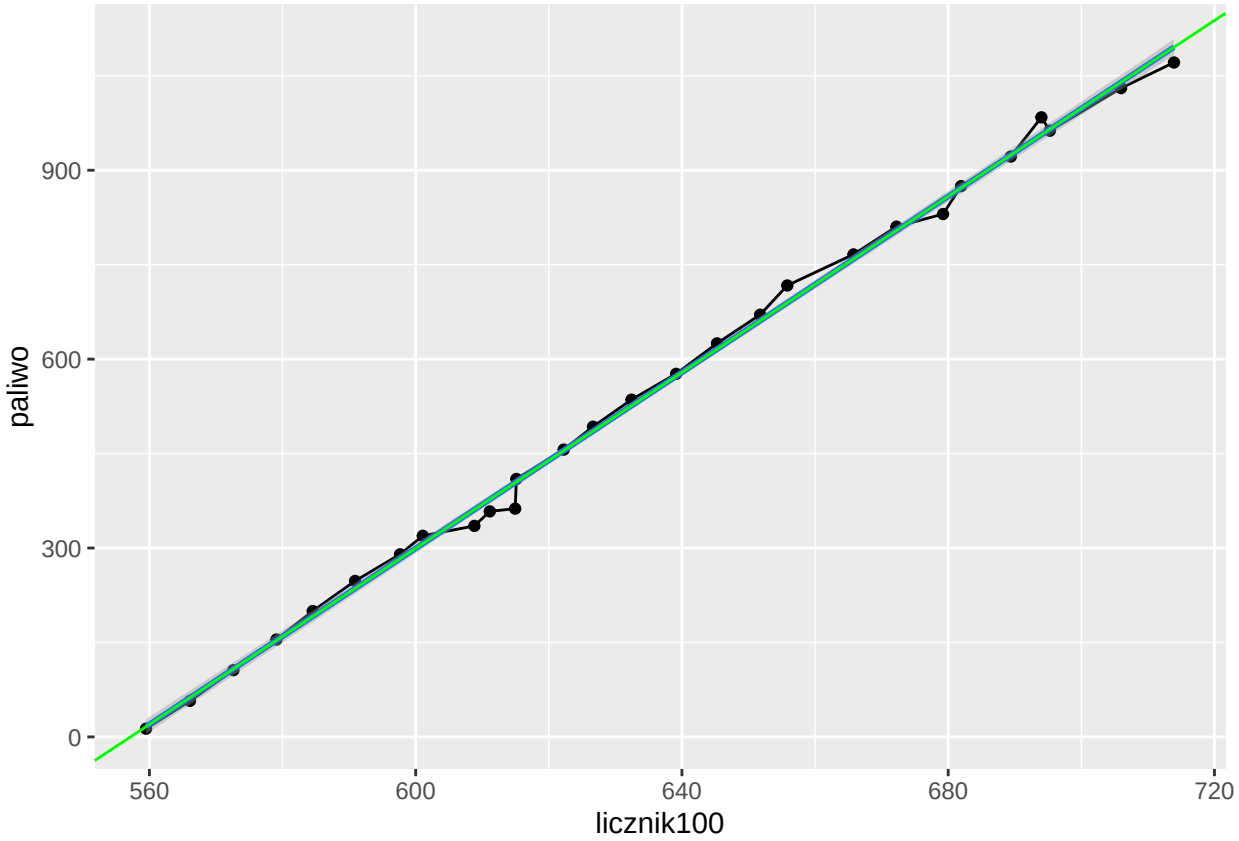



```
plot(data.100$epsilon)
```



Wizualizacja recznie dopasowanej regresji liniowej

```
plot2 <- plot1 + geom_abline(slope=model$coefficients[2],
                             intercept=model$coefficients[1], color = "green")
plot2
```



Sprwadzenie jakości modelu:

RSE - błąd standardowy odchyłek

```
summary(model)$r.squere
```

```
## NULL
```

R^2 - wariancja wyjaśniania przez model

```
summary(model)$sigma
```

```
## [1] 15.68621
```

Współczynniki:

```
A <- data.frame(summary(model)$coef)
A[,4] <- format.pval(summary(model)$coeff[,4], eps=0.001, digits=2)
kable(A, digits=2, col.names = c('Współczynnik', 'SE', 't', 'p-value'))
```

	Współczynnik	SE	t	p-value
(Intercept)	-3893.07	42.38	-91.86	<0.001
licznik100	6.99	0.07	104.91	<0.001

```
kable(cor(data.100[c(3,4,7,8)], method = "pearson"), digits=3)
```

	Cena.tankowania	Cena.jednostkowa	paliwo	licznik100
Cena.tankowania	1.000	0.139	0.277	0.254
Cena.jednostkowa	0.139	1.000	0.529	0.533
paliwo	0.277	0.529	1.000	0.999
licznik100	0.254	0.533	0.999	1.000

Biorąc wszystko pod uwagę można dokonać podsumowania, iż regresja jest dobrze dopasowana.