

lab07

Jakub Bryl

27 11 2019

Regresja Liniowa & Spline'y

Uzyte dane: - amazon.csv (dostarczone przed dr.Ruska)

Kontynuuję analizę tych danych ponieważ nie została skończona na zajęciach, oraz bardzo bym chciał się dowiedzieć jak poprawnie podejść do zagadnienia przedstawionego i opisanego w pkt.3 (przedstawiłem swój tok rozumowania i próbę dopasowania optymalnej f.regresji)

1). Wczytanie Danych

```
avocado <- read_csv("avocado.csv")  
  
## Warning: Missing column names filled in: 'X1' [1]  
  
## Parsed with column specification:  
## cols(  
##   X1 = col_double(),  
##   Date = col_date(format = ""),  
##   AveragePrice = col_double(),  
##   `Total Volume` = col_double(),  
##   `4046` = col_double(),  
##   `4225` = col_double(),  
##   `4770` = col_double(),  
##   `Total Bags` = col_double(),  
##   `Small Bags` = col_double(),  
##   `Large Bags` = col_double(),  
##   `XLarge Bags` = col_double(),  
##   type = col_character(),  
##   year = col_double(),  
##   region = col_character()  
## )  
  
colnames(avocado)[1] <- "Index"
```

2). Początek analizy LM

W tym kroku będę chciał zobaczyć wartości p-value dla wszystkich czynników, tak aby sprawdzić czy wybrane mają dostatecznie duży wpływ na Średnią kwotę Avocado. Następnie przeanalizuję graficznie zależność od Daty oraz Typu avocado dopasowując do wykresu regresję liniową (z zmienną kategoryczną 'typ')

```
#Sprawdzenie p-value dla wszystkich czynników (czy jest dostatecznie niskie -> wpływ na średnią cenę)
regr <- lm(AveragePrice~., avocado)
summary(regr)
```

```
##
## Call:
## lm(formula = AveragePrice ~ ., data = avocado)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.04821 -0.15535 -0.00159  0.14993  1.48620 
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)               6.187e+02  2.150e+01  28.773 < 2e-16 ***
## Index                     3.029e-03  2.279e-04 13.294 < 2e-16 ***
## Date                      1.054e-03  3.270e-05 32.233 < 2e-16 ***
## `Total Volume`          -1.730e-05 3.965e-05 -0.436 0.662577  
## `4046`                   1.731e-05 3.965e-05  0.437 0.662365  
## `4225`                   1.729e-05 3.965e-05  0.436 0.662763  
## `4770`                   1.729e-05 3.965e-05  0.436 0.662804  
## `Total Bags`            -3.179e-02 2.970e-02 -1.070 0.284412  
## `Small Bags`             3.181e-02 2.970e-02  1.071 0.284153  
## `Large Bags`             3.181e-02 2.970e-02  1.071 0.284154  
## `XLarge Bags`            3.181e-02 2.970e-02  1.071 0.284132  
## typeorganic              4.921e-01 4.020e-03 122.416 < 2e-16 ***
## year                     -3.152e-01 1.094e-02 -28.813 < 2e-16 ***  
## regionAtlanta            -2.226e-01 1.976e-02 -11.264 < 2e-16 ***  
## regionBaltimoreWashington -2.344e-02 1.978e-02 -1.185 0.236091  
## regionBoise               -2.128e-01 1.975e-02 -10.774 < 2e-16 ***  
## regionBoston              -2.697e-02 1.978e-02 -1.364 0.172701  
## regionBuffaloRochester    -4.379e-02 1.975e-02 -2.217 0.026613 *  
## regionCalifornia           -1.736e-01 2.020e-02 -8.593 < 2e-16 ***  
## regionCharlotte            4.548e-02 1.976e-02  2.302 0.021332 *  
## regionChicago              -1.912e-03 1.985e-02 -0.096 0.923266  
## regionCincinnatiDayton    -3.493e-01 1.977e-02 -17.674 < 2e-16 ***  
## regionColumbus             -3.089e-01 1.975e-02 -15.642 < 2e-16 ***  
## regionDallasFtWorth         -4.761e-01 1.979e-02 -24.060 < 2e-16 ***  
## regionDenver                -3.332e-01 1.987e-02 -16.772 < 2e-16 ***  
## regionDetroit               -2.900e-01 1.979e-02 -14.655 < 2e-16 ***  
## regionGrandRapids           -5.841e-02 1.975e-02 -2.957 0.003108 **  
## regionGreatLakes            -2.243e-01 2.051e-02 -10.934 < 2e-16 ***  
## regionHarrisburgScranton   -4.765e-02 1.975e-02 -2.413 0.015844 *  
## regionHartfordSpringfield   2.590e-01 1.975e-02 13.111 < 2e-16 ***  
## regionHouston                -5.112e-01 1.978e-02 -25.839 < 2e-16 ***  
## regionIndianapolis          -2.466e-01 1.975e-02 -12.485 < 2e-16 ***  
## regionJacksonville          -4.984e-02 1.975e-02 -2.523 0.011630 *  
## regionLasVegas                -1.782e-01 1.975e-02 -9.023 < 2e-16 ***  
## regionLosAngeles             -3.554e-01 2.006e-02 -17.712 < 2e-16 ***  
## regionLouisville              -2.739e-01 1.975e-02 -13.872 < 2e-16 ***  
## regionMiamiFtLauderdale     -1.325e-01 1.977e-02 -6.702 2.12e-11 ***  
## regionMidsouth                -1.466e-01 1.995e-02 -7.348 2.10e-13 ***  
## regionNashville               -3.491e-01 1.975e-02 -17.676 < 2e-16 ***
```

```

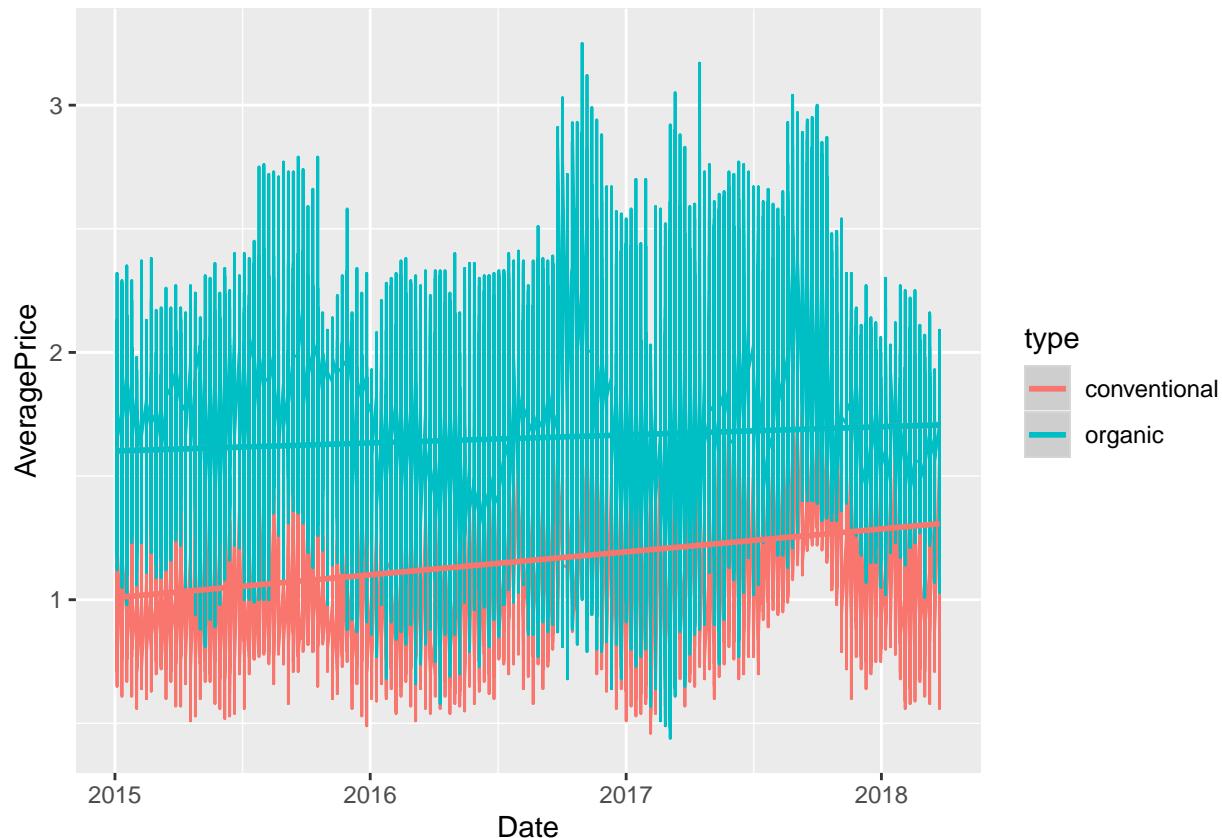
## regionNewOrleansMobile -2.585e-01 1.975e-02 -13.087 < 2e-16 ***
## regionNewYork 1.746e-01 1.993e-02 8.760 < 2e-16 ***
## regionNortheast 6.277e-02 2.136e-02 2.939 0.003296 **
## regionNorthernNewEngland -8.156e-02 1.976e-02 -4.128 3.68e-05 ***
## regionOrlando -5.486e-02 1.976e-02 -2.777 0.005500 **
## regionPhiladelphia 7.331e-02 1.976e-02 3.710 0.000208 ***
## regionPhoenixTucson -3.357e-01 1.981e-02 -16.949 < 2e-16 ***
## regionPittsburgh -1.965e-01 1.975e-02 -9.950 < 2e-16 ***
## regionPlains -1.260e-01 1.980e-02 -6.362 2.04e-10 ***
## regionPortland -2.395e-01 1.977e-02 -12.115 < 2e-16 ***
## regionRaleighGreensboro -5.387e-03 1.976e-02 -0.273 0.785129
## regionRichmondNorfolk -2.695e-01 1.975e-02 -13.646 < 2e-16 ***
## regionRoanoke -3.130e-01 1.975e-02 -15.847 < 2e-16 ***
## regionSacramento 6.050e-02 1.975e-02 3.063 0.002193 **
## regionSanDiego -1.622e-01 1.975e-02 -8.211 2.34e-16 ***
## regionSanFrancisco 2.444e-01 1.976e-02 12.364 < 2e-16 ***
## regionSeattle -1.146e-01 1.977e-02 -5.797 6.88e-09 ***
## regionSouthCarolina -1.578e-01 1.975e-02 -7.991 1.42e-15 ***
## regionSouthCentral -4.618e-01 2.045e-02 -22.584 < 2e-16 ***
## regionSoutheast -1.615e-01 2.030e-02 -7.954 1.92e-15 ***
## regionSpokane -1.153e-01 1.975e-02 -5.837 5.40e-09 ***
## regionStLouis -1.308e-01 1.975e-02 -6.622 3.65e-11 ***
## regionSyracuse -4.078e-02 1.975e-02 -2.065 0.038903 *
## regionTampa -1.519e-01 1.976e-02 -7.689 1.56e-14 ***
## regionTotalUS -1.857e-01 2.463e-02 -7.538 5.00e-14 ***
## regionWest -2.549e-01 2.059e-02 -12.380 < 2e-16 ***
## regionWestTexNewMexico -2.952e-01 1.983e-02 -14.882 < 2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2567 on 18183 degrees of freedom
## Multiple R-squared: 0.5951, Adjusted R-squared: 0.5936
## F-statistic: 411.1 on 65 and 18183 DF, p-value: < 2.2e-16

```

```

# Dopasowanie LM do wykresu zależności Daty od Ceny z zmienną kategoryczną typu
# awocado (organiczny / zwykły) które mają znaczący wpływ na AvaragePrice
ggplot(avocado, aes(x=Date, y= AveragePrice, color=type)) +geom_line() +stat_smooth(method = 'lm')

```



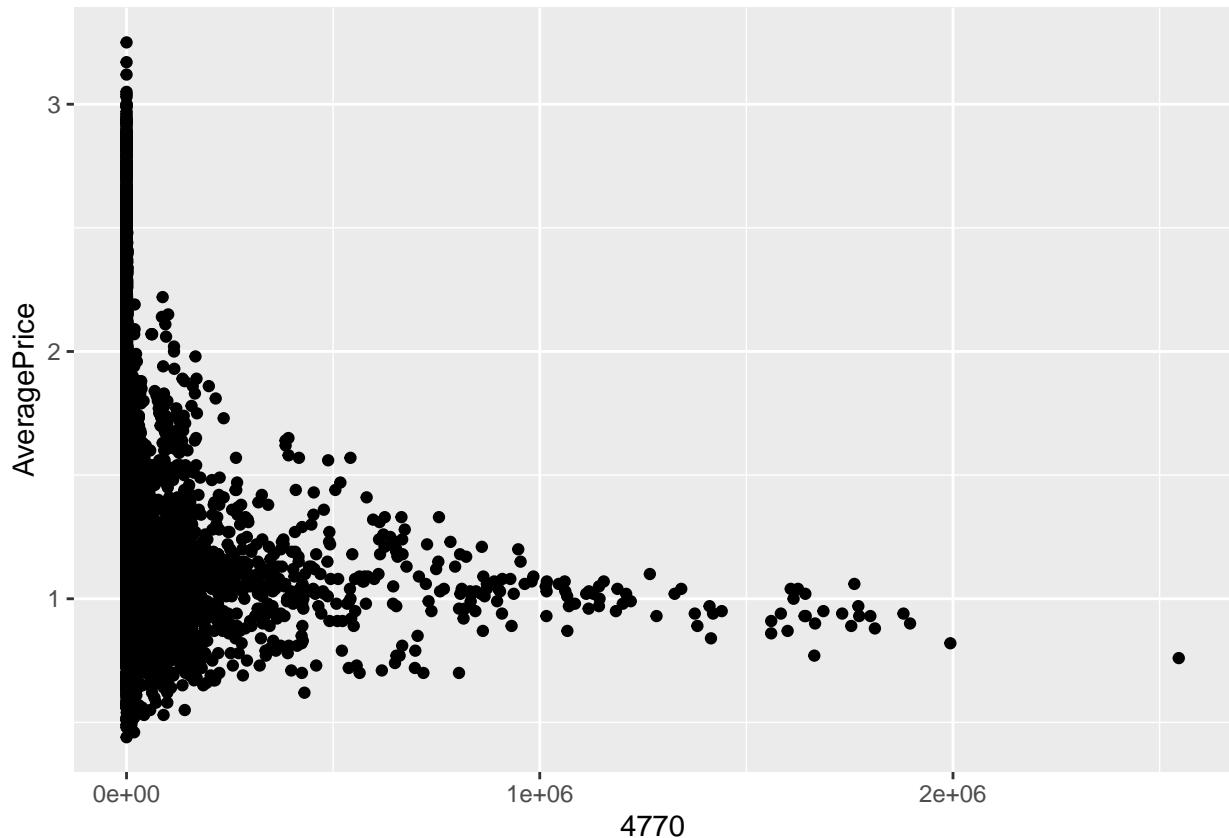
```
#Ponownie wykreslenie podsumowania dla dwóch parametrów które nakreśliliśmy (niskie p-val)
regr_2 <- lm(AveragePrice ~ Date + type, avocado)
summary(regr_2)
```

```
##
## Call:
## lm(formula = AveragePrice ~ Date + type, data = avocado)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.24884 -0.20040 -0.01754  0.18628  1.58278 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -1.764e+00  1.151e-01 -15.32   <2e-16 ***
## Date        1.716e-04  6.759e-06   25.39   <2e-16 ***
## typeorganic 4.960e-01  4.616e-03  107.44   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3118 on 18246 degrees of freedom
## Multiple R-squared:  0.4004, Adjusted R-squared:  0.4004 
## F-statistic: 6093 on 2 and 18246 DF,  p-value: < 2.2e-16
```

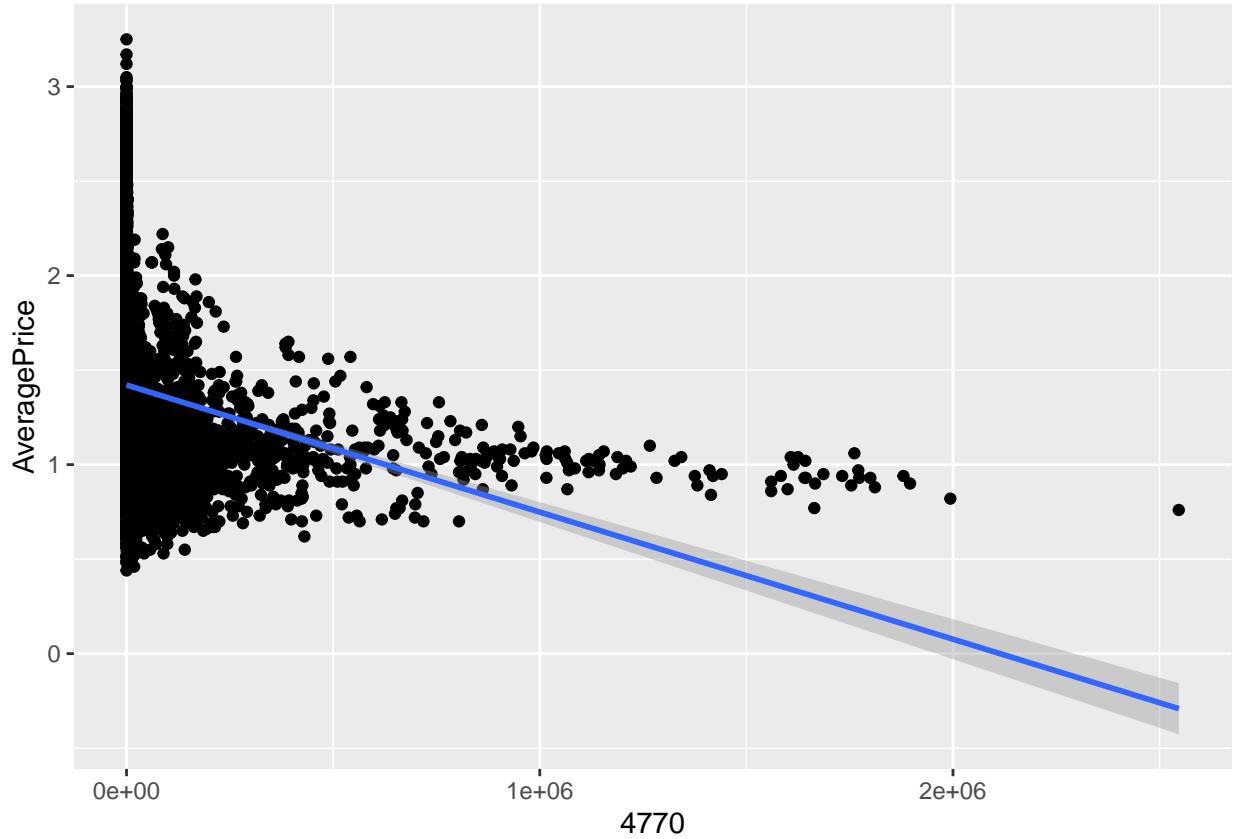
3). Spline'y oraz łamanie regresji liniowej

W tym kroku zająłem się analizą zależności kolumny 4770 (nr. seryjny Awokado) od Średniej Ceny, jako iż ten wykres jest ciekawy i ciężki przez swój kształt. Dzięki tej cesze będę w stanie przetestować i wybrać najlepszy sposób dopasowania regresji.

```
# Prezentacja wykresu
ggplot(avocado, aes(x = `4770`, y = AveragePrice)) + geom_point()
```

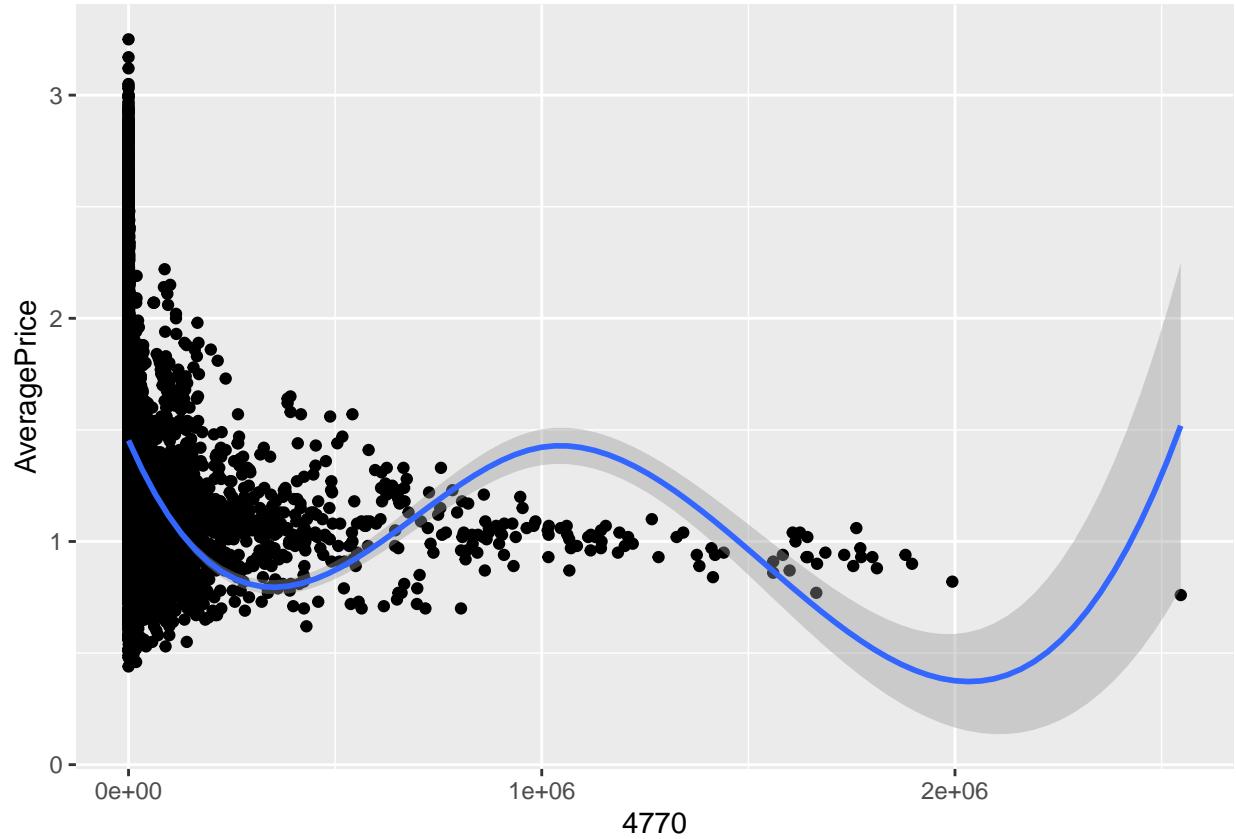


```
# Próba dopasowania regresji liniowej
ggplot(avocado, aes(x = `4770`, y = AveragePrice)) + geom_point() + stat_smooth(method = 'lm')
```

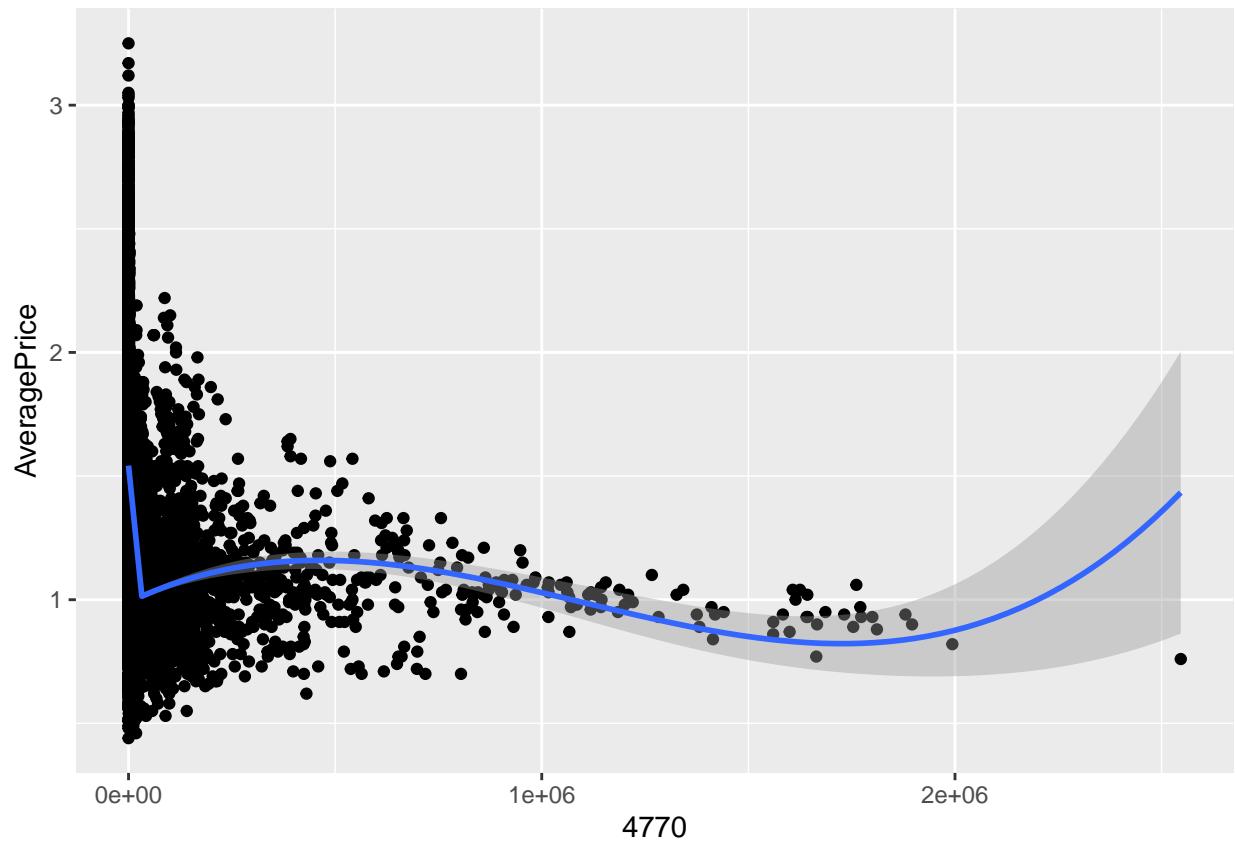


Jak widzimy regresja jest dopasowana źle, bardzo mocno opadająca, dany wykres prawdopodobnie wymaga dopasowania bardziej skomplikowanej funkcji.

```
# Dopuszczanie Base Spline'a z węzłem blisko wartości środkowej
ggplot(avocado, aes(x = `4770`, y = AveragePrice)) + geom_point() +
  stat_smooth(method = lm, formula = y~bs(x, knots = c(1e6)))
```

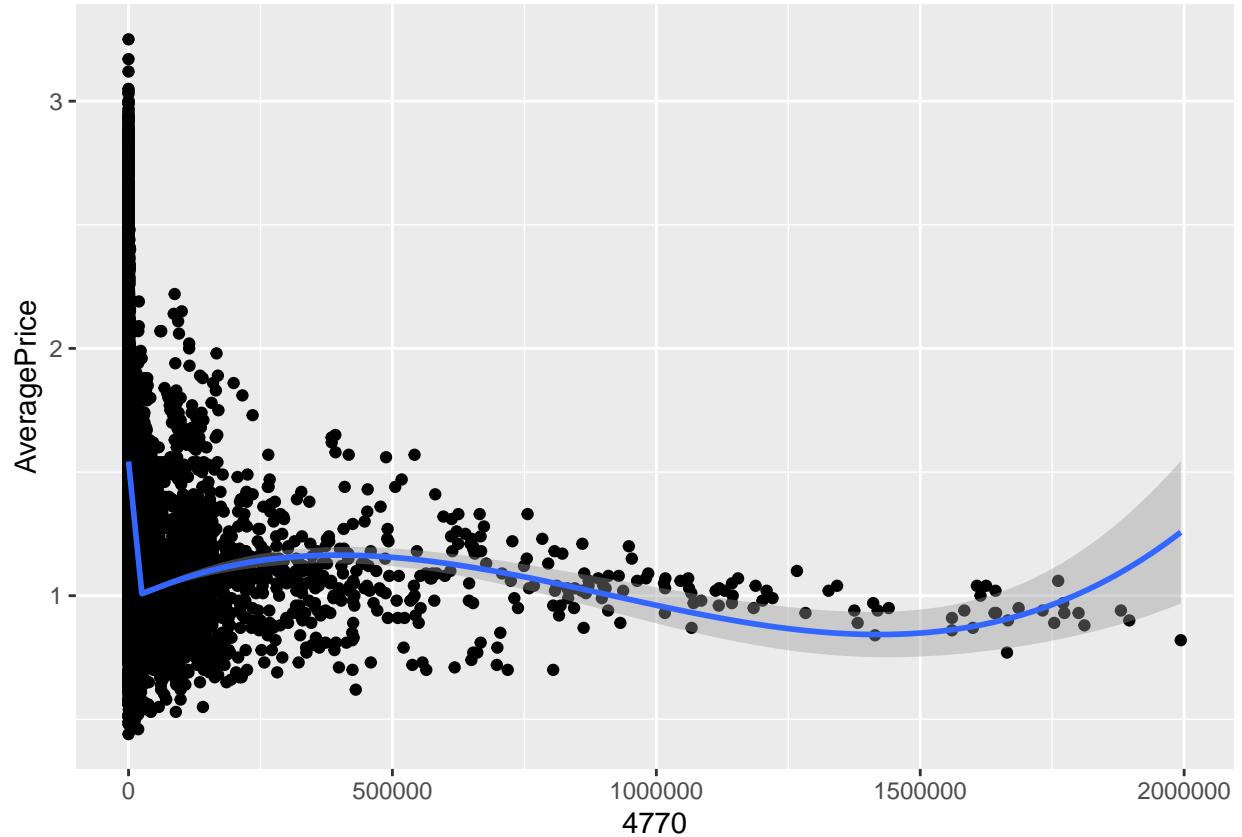


```
# Powyższy wykres nie trafił完全に zachowanie funkcji, wniosek ->
# przesunięcie bliżej punktu dla którego występuje największa wartość
ggplot(avocado, aes(x = `4770`, y = AveragePrice)) + geom_point() +
  stat_smooth(method = lm, formula = y~bs(x, knots = c(0.03e6)))
```

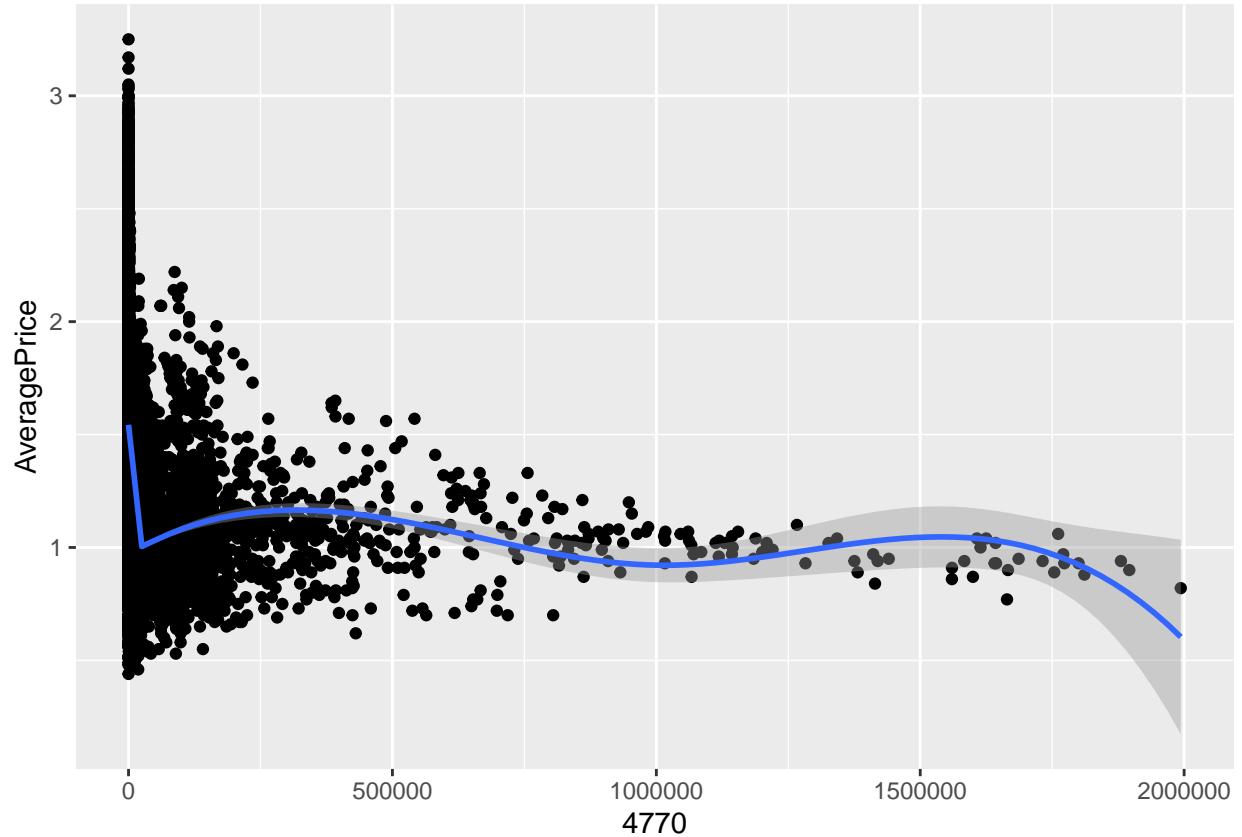


```
# Wycięcie odstającego elementu w celu sprawdzenia jego wpływu na otrzymywane wyniki
avocado_lastdrop <- filter(avocado, `4770` < max(`4770`))

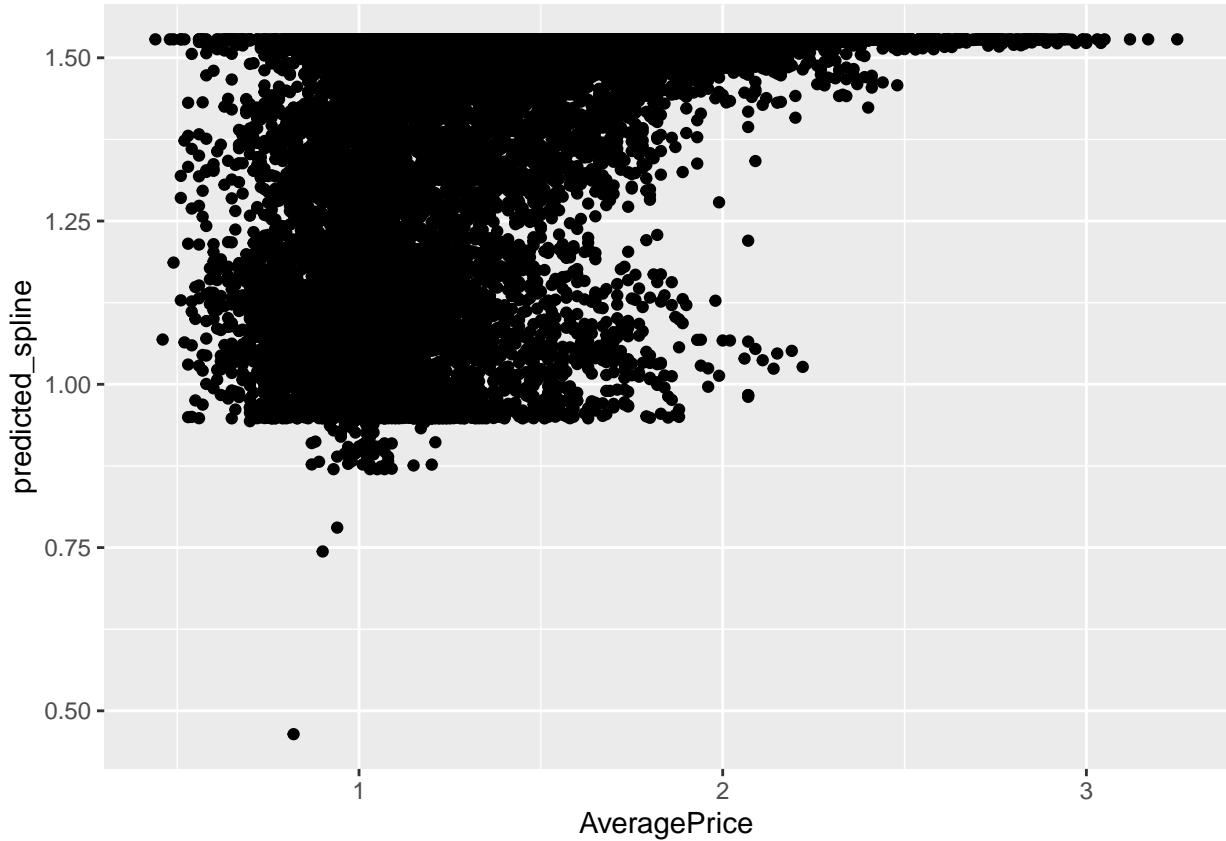
ggplot(avocado_lastdrop, aes(x = `4770`, y = AveragePrice)) + geom_point() +
  stat_smooth(method = lm, formula = y~bs(x, knots = c(0.03e6)))
```



```
# Funkcja w końcowym przedziale zaczyna nam rosnąć, w celu wyeliminowania tego zjawiska konieczne jest
#zwiększenie wymiaru o jeden ( przekształcenie w funkcje opadającą)
ggplot(avocado_lastdrop, aes(x = `4770`, y = AveragePrice)) + geom_point() +
  stat_smooth(method = lm, formula = y~bs(x, knots = c(0.03e6, 1e6)))
```



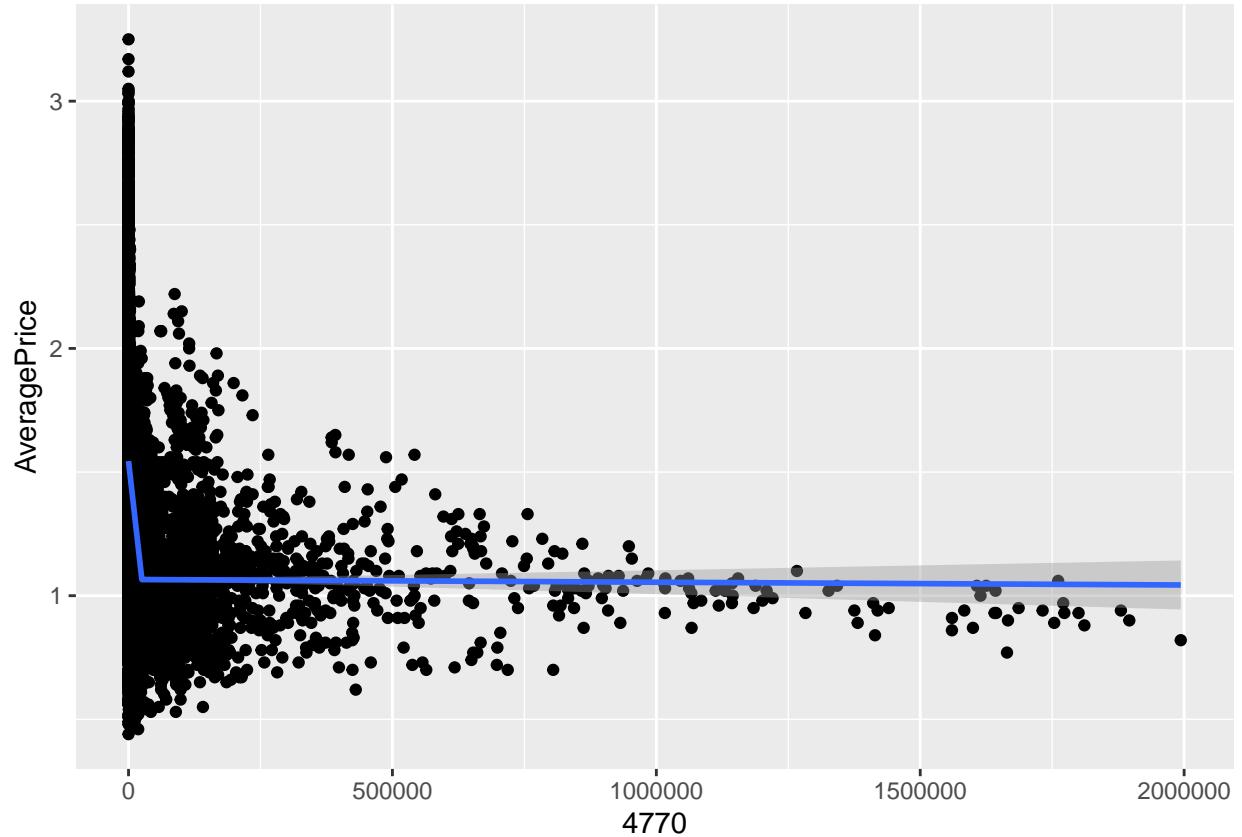
```
# W celu przetestowania poprawności danych wykonam predykcje
model_spline <- lm(AveragePrice ~ bs(`4770`, knots = c(0.05e6, 1e6)), avocado_lastdrop)
# Dodanie wartości przewidzianej z Spline'u do danych
avocado_lastdrop$predicted_spline <- predict(model_spline)
# Wykres zależności (czy predykcja jest poprawna)
ggplot(avocado_lastdrop, aes(x = AveragePrice, y = predicted_spline )) + geom_point()
```



Przy zastosowaniu Spline'ów nawet 2 stopnia nie otrzymaliśmy satysfakcjonujących wyników, dlatego postanowiłem zamiast spline'u spróbować zagiąć funkcję regresji w odpowiednim miejscu (węźle), przesunąć węzeł jeszcze bliżej skupiska danych a następnie przedstawić porównanie przewidzianych wartości

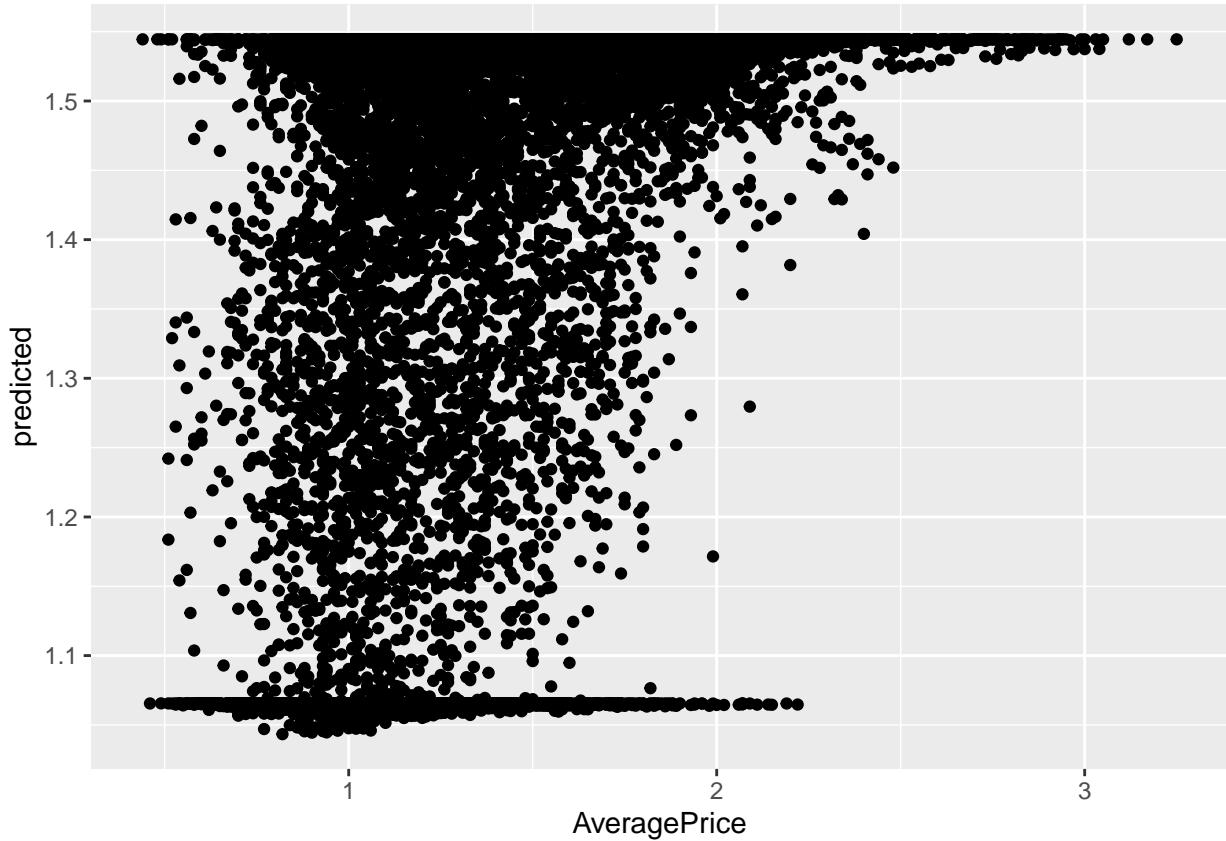
```
# "Łamanie" funkcji regresji przy wartości X ~ 0.01e6
ggplot(avocado_lastdrop, aes(x= `4770`, y= AveragePrice)) +geom_point()+
  stat_smooth(method = 'lm', formula = y ~ x + I(x-0.01e6):I(x>0.01e6))
```

```
## Warning in predict.lm(model, newdata = new_data_frame(list(x = xseq)), :
## se.fit = se, : prediction from a rank-deficient fit may be misleading
```



```
model <- lm(AveragePrice ~ `4770` + I(`4770`-0.01e6):I(`4770`>0.01e6), avocado_lastdrop)
avocado_lastdrop$predicted <- predict(model)

ggplot(avocado_lastdrop, aes(x = AveragePrice, y = predicted )) + geom_point()
```



Wydaje mi się iż tak dopasowana funkcja regresji jest bardziej poprawna niż za pomocą splajnów oraz przewidziane wartości są trafniejsze (brakuje mi pomysłu jak sprawić aby w końcowym etapie funkcja była bardziej zakrzywiona "w dół"). Niestety brakuje mi na tyle wiedzy i obeznania w tym temacie aby móc to stwierdzić na 100%, z tego powodu bardzo bym prosił o opinię czy moje rozumowanie oraz podejście do analizy jest poprawne.