

Lab9 : Daily demand forecast

Given the dataset of daily demand forecast

- The dataset was collected during 60 days, this is a real database of a Brazilian logistics company.
- The dataset has twelve predictors and a target that is the total of orders for daily treatment.

Experiment and create the best regression model for predicting daily

Value of Correlation every pair

	week_no	day_no	non_urgent_order	urgent_order	order_type_A	order_type_B	order_type_C	first_sector_orders	order_from_the_traffic_controller_sector	bank_order_1	bank_order_2	bank_order_3	target
week_no	1.000000000	-0.207791361	0.24347184	0.11825817	0.25611518	0.31276651	-0.041581501	0.0009644928	-0.1940876	0.392309707	0.14708608	-0.15705939	0.21170917
day_no	-0.207791361	1.00000000	-0.41633080	-0.51710646	-0.06889354	-0.37651221	-0.448823125	-0.1323968395	-0.3394851	-0.051815338	-0.57703533	-0.01225143	-0.43528649
non_urgent_order	0.2434718350	-0.41633080	1.00000000	0.56479650	0.56139741	0.82718576	0.752627435	-0.0571501771	0.2469369	0.732356705	0.78819200	0.13285748	0.93451783
urgent_order	0.1182581666	-0.51710646	0.56479650	1.00000000	0.41281707	0.50892879	0.767154718	-0.0114517083	0.2422678	0.230618058	0.66086821	0.03125973	0.72978474
order_type_A	0.2561151834	-0.06889354	0.56139741	0.41281707	1.00000000	0.43873351	0.218651152	0.0645251249	-0.1517624	0.675327748	0.29437362	0.23036926	0.55973505
order_type_B	0.3127665142	-0.37651221	0.82718576	0.50892879	0.43873351	1.00000000	0.523597758	-0.1224315524	0.1271113	0.592844716	0.71367402	0.06732478	0.90067287
order_type_C	-0.0415815015	-0.44882313	0.75262744	0.76715472	0.21865115	0.52359776	1.00000000	0.0079057268	0.4424042	0.330185827	0.71873880	0.03105275	0.80498379
first_sector_orders	0.0009644928	-0.13239684	-0.05715018	-0.01145171	0.06452512	-0.12243155	0.007905727	1.000000000	0.1976704	0.003522811	-0.05506611	0.29456725	-0.05211609
order_from_the_traffic_controller_sector	-0.1940876154	-0.33948507	0.24693693	0.24226777	-0.15176236	0.12711128	0.442404205	0.1976704001	1.0000000	-0.162309380	0.24044966	0.23161439	0.24471159
bank_order_1	0.3923097073	-0.05181534	0.73235671	0.23061806	0.67532775	0.59284472	0.330185827	0.0035228114	-0.1623094	1.000000000	0.26290454	0.22133454	0.63036504
bank_order_2	0.1470860785	-0.57703533	0.78819200	0.66086821	0.29437362	0.71367402	0.718738796	-0.0550661102	0.2404497	0.262904540	1.00000000	-0.10879711	0.79644702
bank_order_3	-0.1570593884	-0.01225143	0.13285748	0.03125973	0.23036926	0.06732478	0.031052750	0.2945672544	0.2316144	0.221334543	-0.10879711	1.00000000	0.10090063
target	0.2117091715	-0.43528649	0.93451783	0.72978474	0.55973505	0.90067287	0.804983790	-0.0521160909	0.2447116	0.630365036	0.79644702	0.10090063	1.00000000

Linear Regression have 4 experiments

Experiment 1

Code

```
#### model 1 ####
lmUrgent <- lm(urgent_order~order_type_A , data = L8_demand_data)
summary(lmUrgent) # review result
# plot graph
ggplot(L8_demand_data ,aes(x=order_type_A,y=urgent_order))+
  geom_point()+
  geom_smooth(formula = y~x,method="lm",color='DeepPink',size=0.3)+
  theme_minimal()
```

Result

```
> #### model 1 ####
> lmUrgent <- lm(urgent_order~order_type_A , data = L8_demand_data)
> summary(lmUrgent) # review result

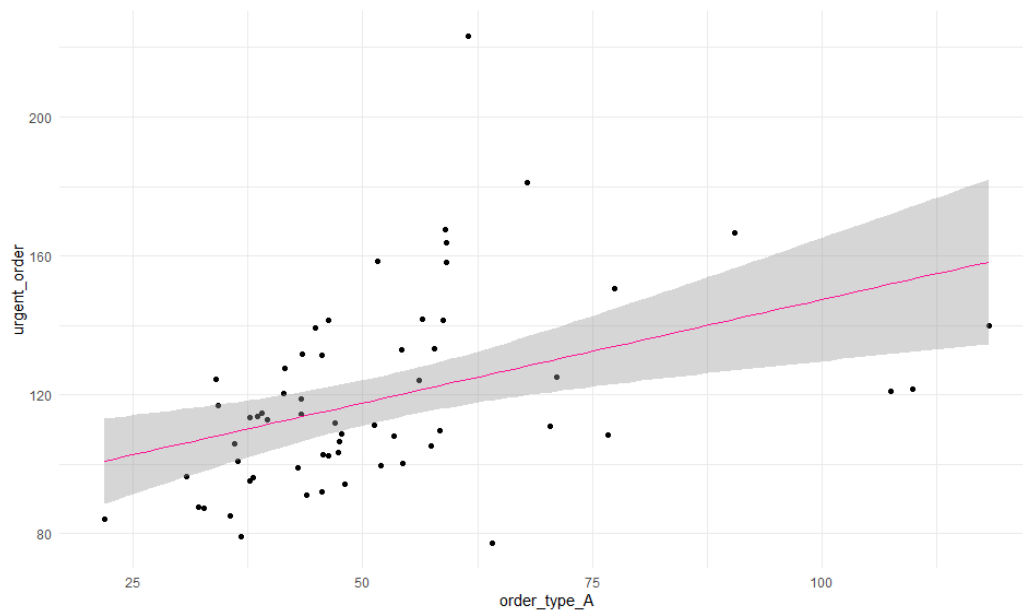
Call:
lm(formula = urgent_order ~ order_type_A, data = L8_demand_data)

Residuals:
    Min       1Q   Median       3Q      Max
-48.713 -17.187  -6.039  15.348  98.731

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   87.8786     9.5530   9.199 6.24e-13 ***
order_type_A    0.5957     0.1726   3.452  0.00105 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 24.96 on 58 degrees of freedom
Multiple R-squared:  0.1704,    Adjusted R-squared:  0.1561
F-statistic: 11.91 on 1 and 58 DF, p-value: 0.001046
```

Plot graph



Experiment 2

Code

```
#### model 2 ####
lmNonUrgent_OrderB <- lm(non_urgent_order~order_type_B ,data = L8_demand_data)
summary(lmNonUrgent_OrderB) # review result
# plot graph
ggplot(L8_demand_data ,aes(x=order_type_B,y=non_urgent_order))+
  geom_point()+
  geom_smooth(formula = y~x,method="lm",color='DeepPink',size=0.3)+
  theme_minimal()
```

Result

```
> #### model 2 ####
> lmNonUrgent_OrderB <- lm(non_urgent_order~order_type_B ,data = L8_demand_data)
> summary(lmNonUrgent_OrderB) # review result
```

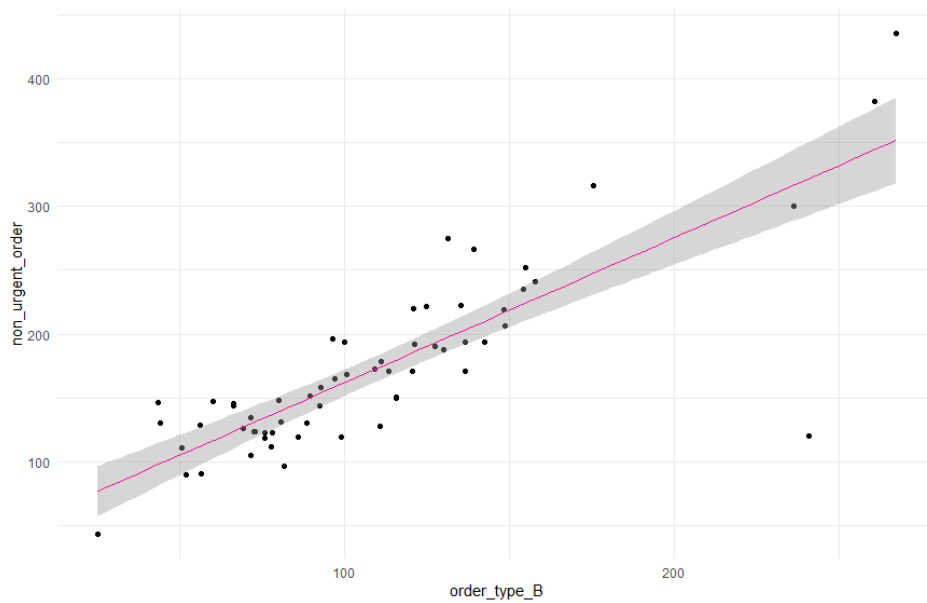
```
Call:
lm(formula = non_urgent_order ~ order_type_B, data = L8_demand_data)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-201.144  -16.616   -0.346   20.177   83.595
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  48.7885    12.1550   4.014 0.000174 ***
order_type_B   1.1331     0.1011  11.211 3.79e-16 ***
---
signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 39.39 on 58 degrees of freedom
Multiple R-squared:  0.6842,    Adjusted R-squared:  0.6788
F-statistic: 125.7 on 1 and 58 DF,  p-value: 3.793e-16
```

Plot graph



Experiment 3

Code

```
#### model 3 ####
lmTarget_OrderB <- lm(target~order_type_B,data = L8_demand_data)
summary(lmTarget_OrderB)
# plot graph
ggplot(L8_demand_data ,aes(x=order_type_B,y=target))+
  geom_point()+
  geom_smooth(formula = y~x,method="lm",color='DeepPink',size=0.4)+
  theme_minimal()
```

Result

```
> #### model 3 ####
> lmTarget_OrderB <- lm(target~order_type_B,data = L8_demand_data)
> summary(lmTarget_OrderB)
```

Call:

```
lm(formula = target ~ order_type_B, data = L8_demand_data)
```

Residuals:

Min	1Q	Median	3Q	Max
-107.717	-19.592	-6.371	21.272	133.167

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	127.1477	12.1159	10.49	5.04e-15 ***
order_type_B	1.5905	0.1007	15.79	< 2e-16 ***

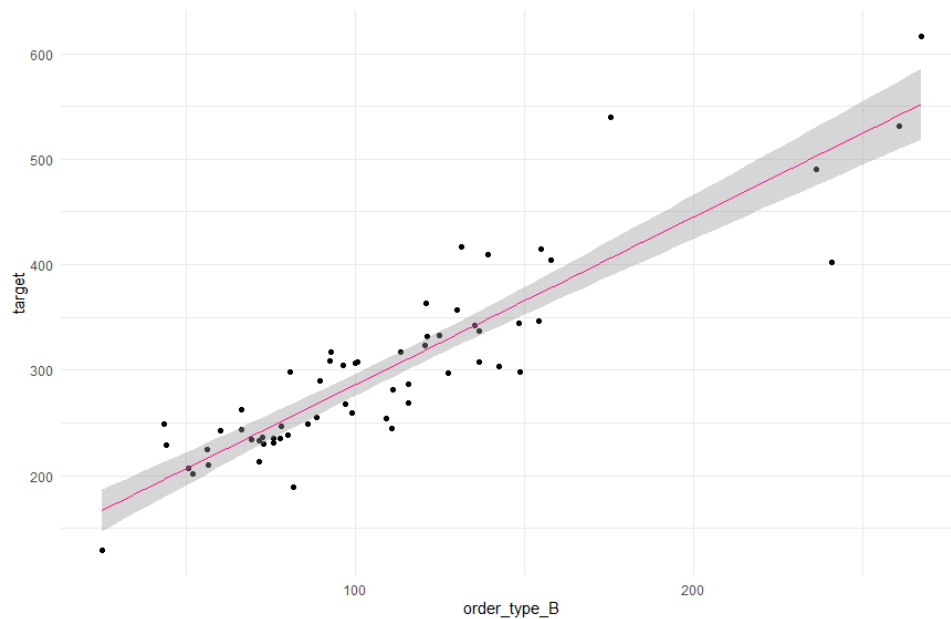
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 39.27 on 58 degrees of freedom

Multiple R-squared: 0.8112, Adjusted R-squared: 0.808

F-statistic: 249.2 on 1 and 58 DF, p-value: < 2.2e-16

Plot graph



Experiment 4

Code

```
#### model 4 : The Best Model in Simple Linear Regression####
lmTarget_nonUrgentOrder <- lm(target~non_urgent_order,data = L8_demand_data)
summary(lmTarget_nonUrgentOrder)
# plot graph
ggplot(L8_demand_data ,aes(x=non_urgent_order,y=target))+
  geom_point()+
  geom_smooth(formula = y~x,method="lm",color='DeepPink',size=0.4)+
  theme_minimal()
```

Result

```
> #### model 4 ####
> lmTarget_nonUrgentOrder <- lm(target~non_urgent_order,data = L8_demand_data)
> summary(lmTarget_nonUrgentOrder)
```

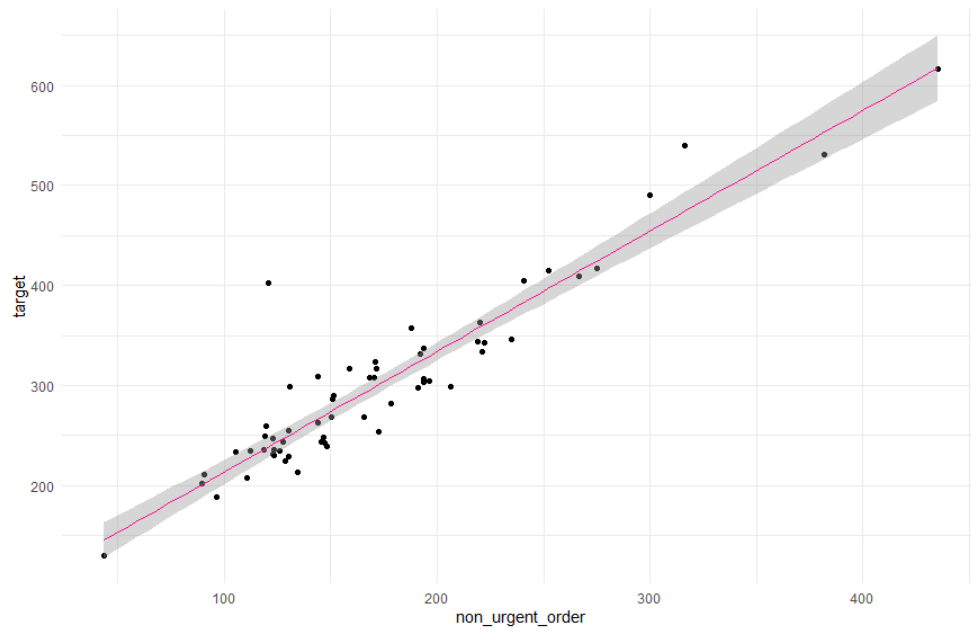
```
Call:
lm(formula = target ~ non_urgent_order, data = L8_demand_data)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-47.301 -21.302  -4.259  12.069 164.290
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   92.99370    11.19444     8.307 1.88e-11 ***
non_urgent_order  1.20472     0.06025    19.996 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 32.16 on 58 degrees of freedom
Multiple R-squared:  0.8733,    Adjusted R-squared:  0.8711
F-statistic: 399.9 on 1 and 58 DF,  p-value: < 2.2e-16
```

Plot graph



Discussion

Experiment 4 ถือว่าเป็น model ที่ดีที่สุดใน Simple Linear Regression เพราะว่ามีค่า adjusted R-squared ที่มากที่สุดเท่าที่ทดลองมาทั้ง 4 การทดลองในการทำ Linear Regression

Multiple & Interaction Regression have 2 experiments

Experiment 5

Code

```
#### model 5 ####
lmTarget_UrgentOrderA <- lm(target~ urgent_order + order_type_A ,
data = L8_demand_data)
summary(lmTarget_UrgentOrderA)
Type_A0 <- L8_demand_data$urgent_order+L8_demand_data$order_type_A
# plot graph
ggplot(L8_demand_data ,aes(x=Type_A0,y=target))+
  geom_point()+
  geom_smooth(formula = y~x,method="lm",color='DeepPink',size=0.4)+
  theme_minimal()
```

Result

```
> #### model 5 ####
> lmTarget_UrgentOrderA <- lm(target~ urgent_order + order_type_A ,data = L8_demand_data)
> summary(lmTarget_UrgentOrderA)

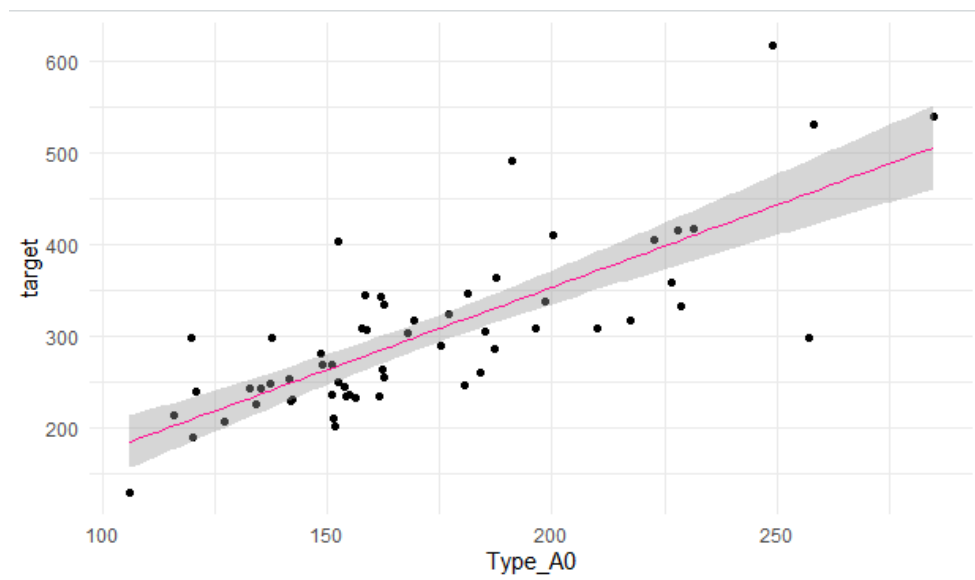
Call:
lm(formula = target ~ urgent_order + order_type_A, data = L8_demand_data)

Residuals:
    Min       1Q   Median       3Q      Max
-153.810  -32.394   -5.891   25.261  168.831

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -12.1445    34.0308  -0.357  0.72251
urgent_order    1.9825     0.2983   6.646 1.23e-08 ***
order_type_A    1.4826     0.4304   3.444  0.00108 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 56.7 on 57 degrees of freedom
Multiple R-squared:  0.6131,    Adjusted R-squared:  0.5995
F-statistic: 45.17 on 2 and 57 DF,  p-value: 1.762e-12
```

Plot graph



Experiment 6

Code

```
#### model 6 : The Best Model in Interaction Regression model ####
lmTarget_nonUrgentOrderA <- lm(target~non_urgent_order*order_type_A ,data = L8_demand_data)
summary(lmTarget_nonUrgentOrderA)

Type_A <- L8_demand_data$non_urgent_order*L8_demand_data$order_type_A
# plot graph
ggplot(L8_demand_data ,aes(x=Type_A,y=target))+
  geom_point()+
  geom_smooth(formula = y~x,method="lm",color='DeepPink',size=0.4)+
  theme_minimal()
# predict with linear regression
predict(lmTarget_nonUrgentOrderA ,L8_demand_data)
```

Result

```
> #### model 6 : The Best Model ####
> lmTarget_nonUrgentOrderA <- lm(target~non_urgent_order*order_type_A ,data = L8_demand_data)
> summary(lmTarget_nonUrgentOrderA)

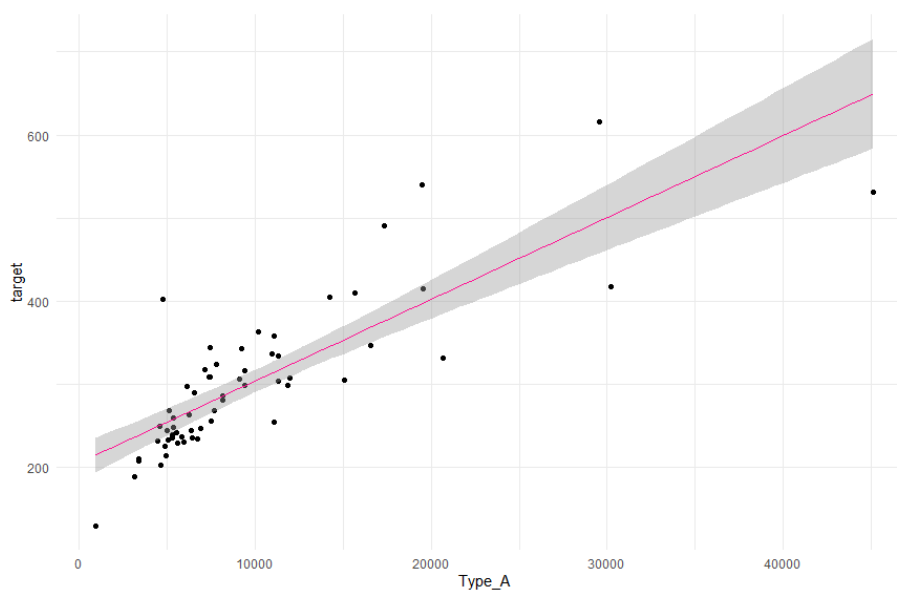
call:
lm(formula = target ~ non_urgent_order * order_type_A, data = L8_demand_data)

Residuals:
    Min       1Q   Median       3Q      Max
-53.56 -18.86  -3.14   12.91  167.82

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    53.46222   31.73716    1.685   0.0976 .
non_urgent_order  1.33952    0.16618    8.061 6.14e-11 ***
order_type_A     0.83979    0.58341    1.439   0.1556
non_urgent_order:order_type_A -0.00283    0.00246   -1.150   0.2549
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 32.12 on 56 degrees of freedom
Multiple R-squared:  0.878,    Adjusted R-squared:  0.8715
F-statistic: 134.3 on 3 and 56 DF,  p-value: < 2.2e-16
```

Plot graph



Discussion

Experiment 6 ถือว่าเป็น model ที่ดีที่กว่า Experiment 5 เพราะว่ามีค่า Adjusted R squared มากกว่า และมีค่า error น้อยกว่า ซึ่งถือว่า Experiment 6 เป็น model ที่ดีที่สุดจากที่ทดลองมาทั้ง 6 การทดลอง

Predict Value at Experiment 6

Code

```
predict
predict(lmTarget_nonUrgentOrderA ,L8_demand_data)
```

Result

```
> # predict
> predict(lmTarget_nonUrgentOrderA ,L8_demand_data)
      1      2      3      4      5      6      7      8      9     10     11     12     13
473.7539 243.8751 127.5667 297.6661 196.7205 218.2412 265.2229 232.7399 354.1128 265.2008 307.7801 267.5752 297.4145
      14      15      16      17      18      19      20      21      22      23      24      25      26
358.6140 329.4891 245.5286 268.7448 220.5048 385.2166 271.4500 248.0029 241.8405 318.6802 415.7526 236.1097 266.8591
      27      28      29      30      31      32      33      34      35      36      37      38      39
239.6986 266.2789 454.5146 274.0284 341.3515 298.5696 609.9404 380.6670 304.5985 307.4082 536.4097 360.9940 326.9438
      40      41      42      43      44      45      46      47      48      49      50      51      52
428.6689 401.1538 292.3139 248.0642 228.5665 242.2279 323.2390 236.4514 243.3175 234.7842 255.2650 359.9413 268.9096
      53      54      55      56      57      58      59      60
201.2228 203.8697 250.4080 289.1631 277.8385 329.7560 338.5162 342.6553
> |
```

Why is sometime adding predictors do not help prediction?

Answer บางทีการที่เราเพิ่มตัวทำนายเข้าไป อาจจะไม่ช่วยในการทำนาย เพราะว่า ตัวทำนายในโมเดลตอนแรกอาจจะดีอยู่แล้ว ที่จะสามารถอธิบายรายละเอียดต่าง ๆ ไปได้ดี โดยที่เราไม่ต้องเพิ่มตัวทำนายเข้าไป หรือบางทีการที่เราเพิ่มตัวทำนายเข้าไป เพื่อที่เราจะได้โมเดลที่มีการทำนายดีที่สุด