

Modul 08 - Simple Linear Regression

Roni Yunis

12/02/2024

Pengantar

Regresi linear sederhana adalah suatu metode statistik yang digunakan untuk memodelkan hubungan linier antara satu variabel bebas (independen) dengan satu variabel terikat (dependen). Dalam konteks ini, “linier” mengacu pada hubungan garis lurus antara variabel-variabel tersebut.

Dalam persamaan regresi linear sederhana, dapat dinyatakan sebagai berikut:

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

- Y adalah variabel terikat (dependen).
- X adalah variabel bebas (independen).
- β_0 adalah intercept (konstanta).
- β_1 adalah koefisien regresi yang mengukur tingkat perubahan dalam Y untuk setiap satu unit perubahan dalam X .
- ε adalah kesalahan acak yang tidak dapat dijelaskan oleh model dan diasumsikan mengikuti distribusi normal dengan mean nol.

Tujuan utama dari regresi linear sederhana adalah menemukan nilai-nilai β_0 dan β_1 yang meminimalkan jumlah kuadrat kesalahan (sum of squared errors) antara nilai prediksi yang diberikan oleh model dan nilai aktual dari variabel terikat.

Regresi linear sederhana dapat digunakan untuk memahami dan memodelkan hubungan antara dua variabel, serta melakukan prediksi berdasarkan data yang ada. Metode ini umum digunakan dalam statistika dan analisis data.

Contoh:

Anggaplah Anda bekerja dalam sebuah perusahaan ritel dan Anda ingin memahami hubungan antara jumlah uang yang dihabiskan oleh pelanggan dalam satu transaksi X dengan total pendapatan penjualan Y pada transaksi tersebut. Anda dapat menggunakan regresi linear sederhana untuk memodelkan hubungan tersebut.

Contoh Persamaan Regresi Linear Sederhana:

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

- Y : Total pendapatan penjualan dalam satu transaksi.
- X : Jumlah uang yang dihabiskan oleh pelanggan dalam satu transaksi.
- β_0 : Intercept (konstanta), mewakili pendapatan penjualan yang dihasilkan ketika X sama dengan nol.
- β_1 : Koefisien regresi, menunjukkan seberapa banyak pendapatan penjualan meningkat untuk setiap peningkatan satu unit dalam X .

- ε : Kesalahan acak.

Anda mengumpulkan data dari sejumlah transaksi dan kemudian menggunakan regresi linear sederhana untuk mengestimasi nilai β_0 dan β_1 . Setelah memperoleh model regresi, Anda dapat menggunakannya untuk membuat prediksi. Misalnya, jika seorang pelanggan menghabiskan \$50 dalam transaksi, berapa total pendapatan penjualan yang diperkirakan?

Contoh Implementasi dengan R

Load Packages

```
#Split dataset
library(caTools)
```

```
## Warning: package 'caTools' was built under R version 4.3.2
```

```
#Predicting result visualization
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
#Import dataset marketing pada library datarium
library(datarium)
```

```
## Warning: package 'datarium' was built under R version 4.3.2
```

```
#library manipulasi data
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.3.3
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##   filter, lag
```

```
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
marketing
```

##	youtube	facebook	newspaper	sales
## 1	276.12	45.36	83.04	26.52
## 2	53.40	47.16	54.12	12.48
## 3	20.64	55.08	83.16	11.16
## 4	181.80	49.56	70.20	22.20
## 5	216.96	12.96	70.08	15.48
## 6	10.44	58.68	90.00	8.64
## 7	69.00	39.36	28.20	14.16
## 8	144.24	23.52	13.92	15.84
## 9	10.32	2.52	1.20	5.76
## 10	239.76	3.12	25.44	12.72
## 11	79.32	6.96	29.04	10.32
## 12	257.64	28.80	4.80	20.88
## 13	28.56	42.12	79.08	11.04
## 14	117.00	9.12	8.64	11.64
## 15	244.92	39.48	55.20	22.80
## 16	234.48	57.24	63.48	26.88
## 17	81.36	43.92	136.80	15.00
## 18	337.68	47.52	66.96	29.28
## 19	83.04	24.60	21.96	13.56
## 20	176.76	28.68	22.92	17.52
## 21	262.08	33.24	64.08	21.60
## 22	284.88	6.12	28.20	15.00
## 23	15.84	19.08	59.52	6.72
## 24	273.96	20.28	31.44	18.60
## 25	74.76	15.12	21.96	11.64
## 26	315.48	4.20	23.40	14.40
## 27	171.48	35.16	15.12	18.00
## 28	288.12	20.04	27.48	19.08
## 29	298.56	32.52	27.48	22.68
## 30	84.72	19.20	48.96	12.60
## 31	351.48	33.96	51.84	25.68
## 32	135.48	20.88	46.32	14.28
## 33	116.64	1.80	36.00	11.52
## 34	318.72	24.00	0.36	20.88
## 35	114.84	1.68	8.88	11.40
## 36	348.84	4.92	10.20	15.36
## 37	320.28	52.56	6.00	30.48
## 38	89.64	59.28	54.84	17.64
## 39	51.72	32.04	42.12	12.12
## 40	273.60	45.24	38.40	25.80
## 41	243.00	26.76	37.92	19.92
## 42	212.40	40.08	46.44	20.52
## 43	352.32	33.24	2.16	24.84
## 44	248.28	10.08	31.68	15.48
## 45	30.12	30.84	51.96	10.20
## 46	210.12	27.00	37.80	17.88
## 47	107.64	11.88	42.84	12.72
## 48	287.88	49.80	22.20	27.84
## 49	272.64	18.96	59.88	17.76
## 50	80.28	14.04	44.16	11.64
## 51	239.76	3.72	41.52	13.68
## 52	120.48	11.52	4.32	12.84
## 53	259.68	50.04	47.52	27.12

## 54	219.12	55.44	70.44	25.44
## 55	315.24	34.56	19.08	24.24
## 56	238.68	59.28	72.00	28.44
## 57	8.76	33.72	49.68	6.60
## 58	163.44	23.04	19.92	15.84
## 59	252.96	59.52	45.24	28.56
## 60	252.84	35.40	11.16	22.08
## 61	64.20	2.40	25.68	9.72
## 62	313.56	51.24	65.64	29.04
## 63	287.16	18.60	32.76	18.84
## 64	123.24	35.52	10.08	16.80
## 65	157.32	51.36	34.68	21.60
## 66	82.80	11.16	1.08	11.16
## 67	37.80	29.52	2.64	11.40
## 68	167.16	17.40	12.24	16.08
## 69	284.88	33.00	13.20	22.68
## 70	260.16	52.68	32.64	26.76
## 71	238.92	36.72	46.44	21.96
## 72	131.76	17.16	38.04	14.88
## 73	32.16	39.60	23.16	10.56
## 74	155.28	6.84	37.56	13.20
## 75	256.08	29.52	15.72	20.40
## 76	20.28	52.44	107.28	10.44
## 77	33.00	1.92	24.84	8.28
## 78	144.60	34.20	17.04	17.04
## 79	6.48	35.88	11.28	6.36
## 80	139.20	9.24	27.72	13.20
## 81	91.68	32.04	26.76	14.16
## 82	287.76	4.92	44.28	14.76
## 83	90.36	24.36	39.00	13.56
## 84	82.08	53.40	42.72	16.32
## 85	256.20	51.60	40.56	26.04
## 86	231.84	22.08	78.84	18.24
## 87	91.56	33.00	19.20	14.40
## 88	132.84	48.72	75.84	19.20
## 89	105.96	30.60	88.08	15.48
## 90	131.76	57.36	61.68	20.04
## 91	161.16	5.88	11.16	13.44
## 92	34.32	1.80	39.60	8.76
## 93	261.24	40.20	70.80	23.28
## 94	301.08	43.80	86.76	26.64
## 95	128.88	16.80	13.08	13.80
## 96	195.96	37.92	63.48	20.28
## 97	237.12	4.20	7.08	14.04
## 98	221.88	25.20	26.40	18.60
## 99	347.64	50.76	61.44	30.48
## 100	162.24	50.04	55.08	20.64
## 101	266.88	5.16	59.76	14.04
## 102	355.68	43.56	121.08	28.56
## 103	336.24	12.12	25.68	17.76
## 104	225.48	20.64	21.48	17.64
## 105	285.84	41.16	6.36	24.84
## 106	165.48	55.68	70.80	23.04
## 107	30.00	13.20	35.64	8.64

## 108	108.48	0.36	27.84	10.44
## 109	15.72	0.48	30.72	6.36
## 110	306.48	32.28	6.60	23.76
## 111	270.96	9.84	67.80	16.08
## 112	290.04	45.60	27.84	26.16
## 113	210.84	18.48	2.88	16.92
## 114	251.52	24.72	12.84	19.08
## 115	93.84	56.16	41.40	17.52
## 116	90.12	42.00	63.24	15.12
## 117	167.04	17.16	30.72	14.64
## 118	91.68	0.96	17.76	11.28
## 119	150.84	44.28	95.04	19.08
## 120	23.28	19.20	26.76	7.92
## 121	169.56	32.16	55.44	18.60
## 122	22.56	26.04	60.48	8.40
## 123	268.80	2.88	18.72	13.92
## 124	147.72	41.52	14.88	18.24
## 125	275.40	38.76	89.04	23.64
## 126	104.64	14.16	31.08	12.72
## 127	9.36	46.68	60.72	7.92
## 128	96.24	0.00	11.04	10.56
## 129	264.36	58.80	3.84	29.64
## 130	71.52	14.40	51.72	11.64
## 131	0.84	47.52	10.44	1.92
## 132	318.24	3.48	51.60	15.24
## 133	10.08	32.64	2.52	6.84
## 134	263.76	40.20	54.12	23.52
## 135	44.28	46.32	78.72	12.96
## 136	57.96	56.40	10.20	13.92
## 137	30.72	46.80	11.16	11.40
## 138	328.44	34.68	71.64	24.96
## 139	51.60	31.08	24.60	11.52
## 140	221.88	52.68	2.04	24.84
## 141	88.08	20.40	15.48	13.08
## 142	232.44	42.48	90.72	23.04
## 143	264.60	39.84	45.48	24.12
## 144	125.52	6.84	41.28	12.48
## 145	115.44	17.76	46.68	13.68
## 146	168.36	2.28	10.80	12.36
## 147	288.12	8.76	10.44	15.84
## 148	291.84	58.80	53.16	30.48
## 149	45.60	48.36	14.28	13.08
## 150	53.64	30.96	24.72	12.12
## 151	336.84	16.68	44.40	19.32
## 152	145.20	10.08	58.44	13.92
## 153	237.12	27.96	17.04	19.92
## 154	205.56	47.64	45.24	22.80
## 155	225.36	25.32	11.40	18.72
## 156	4.92	13.92	6.84	3.84
## 157	112.68	52.20	60.60	18.36
## 158	179.76	1.56	29.16	12.12
## 159	14.04	44.28	54.24	8.76
## 160	158.04	22.08	41.52	15.48
## 161	207.00	21.72	36.84	17.28

```
## 162 102.84 42.96 59.16 15.96
## 163 226.08 21.72 30.72 17.88
## 164 196.20 44.16 8.88 21.60
## 165 140.64 17.64 6.48 14.28
## 166 281.40 4.08 101.76 14.28
## 167 21.48 45.12 25.92 9.60
## 168 248.16 6.24 23.28 14.64
## 169 258.48 28.32 69.12 20.52
## 170 341.16 12.72 7.68 18.00
## 171 60.00 13.92 22.08 10.08
## 172 197.40 25.08 56.88 17.40
## 173 23.52 24.12 20.40 9.12
## 174 202.08 8.52 15.36 14.04
## 175 266.88 4.08 15.72 13.80
## 176 332.28 58.68 50.16 32.40
## 177 298.08 36.24 24.36 24.24
## 178 204.24 9.36 42.24 14.04
## 179 332.04 2.76 28.44 14.16
## 180 198.72 12.00 21.12 15.12
## 181 187.92 3.12 9.96 12.60
## 182 262.20 6.48 32.88 14.64
## 183 67.44 6.84 35.64 10.44
## 184 345.12 51.60 86.16 31.44
## 185 304.56 25.56 36.00 21.12
## 186 246.00 54.12 23.52 27.12
## 187 167.40 2.52 31.92 12.36
## 188 229.32 34.44 21.84 20.76
## 189 343.20 16.68 4.44 19.08
## 190 22.44 14.52 28.08 8.04
## 191 47.40 49.32 6.96 12.96
## 192 90.60 12.96 7.20 11.88
## 193 20.64 4.92 37.92 7.08
## 194 200.16 50.40 4.32 23.52
## 195 179.64 42.72 7.20 20.76
## 196 45.84 4.44 16.56 9.12
## 197 113.04 5.88 9.72 11.64
## 198 212.40 11.16 7.68 15.36
## 199 340.32 50.40 79.44 30.60
## 200 278.52 10.32 10.44 16.08
```

View dataset

```
head(marketing)
```

```
##   youtube facebook newspaper sales
## 1  276.12   45.36    83.04 26.52
## 2   53.40   47.16    54.12 12.48
## 3   20.64   55.08    83.16 11.16
## 4  181.80   49.56    70.20 22.20
## 5  216.96   12.96    70.08 15.48
## 6   10.44   58.68    90.00  8.64
```

Melihat dimensi dari data, dengan menggunakan fungsi `dim()`

```
dim(marketing)
```

```
## [1] 200 4
```

Melihat struktur dari data, dengan menggunakan fungsi `glimpse()`

```
glimpse(marketing)
```

```
## Rows: 200
## Columns: 4
## $ youtube <dbl> 276.12, 53.40, 20.64, 181.80, 216.96, 10.44, 69.00, 144.24, ~
## $ facebook <dbl> 45.36, 47.16, 55.08, 49.56, 12.96, 58.68, 39.36, 23.52, 2.52~
## $ newspaper <dbl> 83.04, 54.12, 83.16, 70.20, 70.08, 90.00, 28.20, 13.92, 1.20~
## $ sales <dbl> 26.52, 12.48, 11.16, 22.20, 15.48, 8.64, 14.16, 15.84, 5.76, ~
```

Exploratory Data Analysis

```
# Melihat summary data
summary(marketing)
```

```
##      youtube      facebook      newspaper      sales
## Min.   : 0.84   Min.    : 0.00   Min.     : 0.36   Min.    : 1.92
## 1st Qu.: 89.25   1st Qu.:11.97   1st Qu.: 15.30   1st Qu.:12.45
## Median :179.70   Median :27.48   Median : 30.90   Median :15.48
## Mean   :176.45   Mean    :27.92   Mean     :36.66   Mean    :16.83
## 3rd Qu.:262.59   3rd Qu.:43.83   3rd Qu.: 54.12   3rd Qu.:20.88
## Max.   :355.68   Max.     :59.52   Max.     :136.80   Max.     :32.40
```

```
# Melihat korelasi atau hubungan antar variabel
cor(marketing)
```

```
##      youtube      facebook      newspaper      sales
## youtube  1.00000000  0.05480866  0.05664787  0.7822244
## facebook 0.05480866  1.00000000  0.35410375  0.5762226
## newspaper 0.05664787  0.35410375  1.00000000  0.2282990
## sales    0.78222442  0.57622257  0.22829903  1.0000000
```

Sekarang kita akan mencoba melihat korelasi antara facebook dengan sales

```
#menghitung korelasi antar variabel
korfacebook <- cor(marketing$facebook, marketing$sales)
korfacebook
```

```
## [1] 0.5762226
```

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

Y = Sales X = Facebook

$$Sales = \beta_0 + \beta_1 facebook + \varepsilon$$

Koefisien korelasi mengukur tingkat hubungan antara dua variabel x dan y. Nilainya berkisar antara -1 (korelasi negatif sempurna: ketika x meningkat, y menurun) dan +1 (korelasi positif sempurna: ketika x meningkat, y meningkat).

Nilai yang mendekati 0 menunjukkan hubungan yang lemah antara variabel. Korelasi yang rendah (-0,2 < x < 0,2) mungkin menunjukkan bahwa banyak variasi dari variabel hasil (y) tidak dijelaskan oleh prediktor (x). Dalam kasus seperti itu, kita mungkin harus mencari variabel prediktor yang lebih baik.

Dalam contoh ini, koefisien korelasinya antara variabel facebook dan sales adalah sebesar 0,57

Bagi dataset kedalam data training dan data testing

```
splitdata <- sample.split(marketing$sales, SplitRatio = 0.7)
trainingset <- subset(marketing, splitdata == TRUE)
testingset <- subset(marketing, splitdata == FALSE)
```

```
dim(trainingset)
```

```
## [1] 140  4
```

```
dim(testingset)
```

```
## [1] 60  4
```

Bisa dilihat bahwa untuk data training ada 140 baris data, dan untuk data testing ada 60 baris data

Model Regresi Sederhana

linier regresi sederhana pada data training

```
# Analisis regresi untuk variabel facebook terhadap sales

lm.r <- lm(sales ~ facebook,
           data = trainingset)
summary(lm.r)
```

```
##
## Call:
## lm(formula = sales ~ facebook, data = trainingset)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```



```
## -18.3018 -2.3058 0.8846 3.3228 10.4381
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.13494    0.79581  13.992 < 2e-16 ***
## facebook    0.19122    0.02461   7.768 1.63e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.196 on 138 degrees of freedom
## Multiple R-squared:  0.3043, Adjusted R-squared:  0.2992
## F-statistic: 60.35 on 1 and 138 DF, p-value: 1.628e-12
```

Nilai P-Value = 0.00000000000000057

$$Sales = 10.89708 + 0.21956 \cdot facebook + \varepsilon$$

Misalnya facebooknya Rp. 3500, berapakah nilai penjualan (sales)? Persamaan regresi linernya adalah:

$$Sales = 10.89708 + (0.21956 * 3500) + \varepsilon$$

```
sales_f = 10.89708 + (0.21956 * 3500)
sales_f
```

```
## [1] 779.3571
```

Kalau kita lihat dari model diatas bahwa facebook punya hubungan signifikan terhadap penjualan, artinya nilai penjualan dapat ditingkatkan dari anggaran iklan pada facebook.

Prediksi model regresi dengan data testing

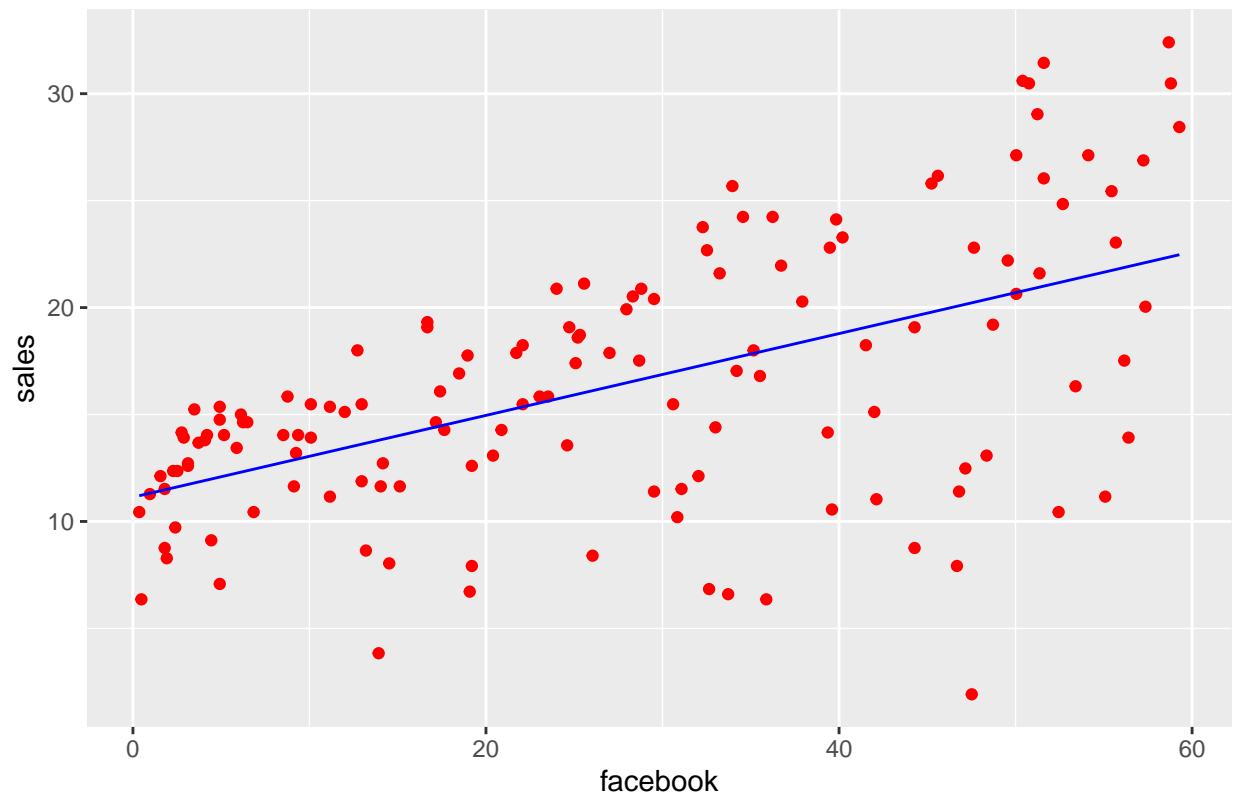
```
yprediksi <- predict(lm.r, newdata = testingset)
head(yprediksi)
```

```
##          1          6          9         11         17         18
## 19.80872 22.35578 11.61682 12.46584 19.53336 20.22176
```

Visualasi hasil data training

```
ggplot() + geom_point(aes(x = trainingset$facebook,
                          y = trainingset$sales), colour = 'red') +
  geom_line(aes(x = trainingset$facebook,
                y = predict(lm.r, newdata = trainingset)), colour = 'blue') +
  ggtitle('Pengaruh Facebook terhadap Sales (Data Training)') +
  xlab('facebook') +
  ylab('sales')
```

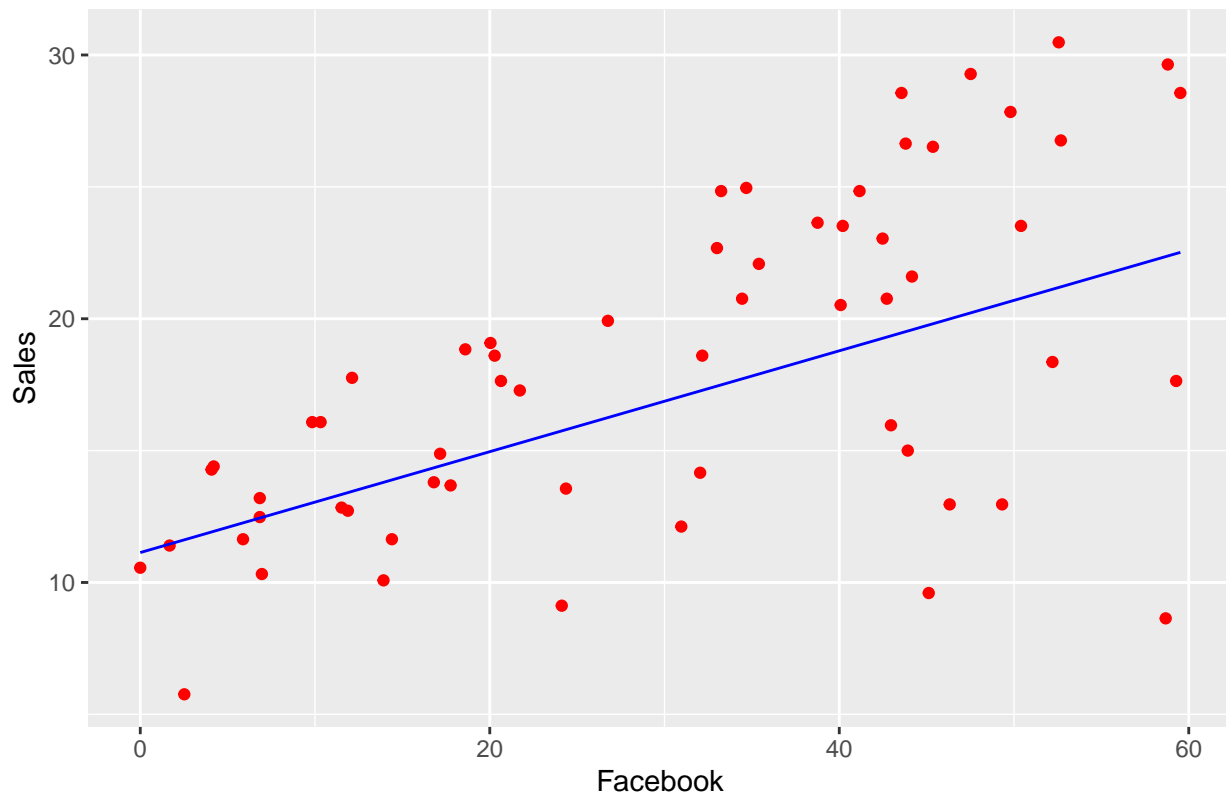
Pengaruh Facebook terhadap Sales (Data Training)



Visualasi hasil data testing

```
ggplot() + geom_point(aes(x = testingset$facebook,  
                           y = testingset$sales), colour = 'red') +  
  geom_line(aes(x = testingset$facebook,  
                y = predict(lm.r, newdata = testingset)), colour = 'blue') +  
  ggtitle('Pengaruh Facebook terhadap Sales (Data Testing)') +  
  xlab('Facebook') +  
  ylab('Sales')
```

Pengaruh Facebook terhadap Sales (Data Testing)



Evaluasi

Sum of Squared Errors (SSE) adalah metrik evaluasi yang mengukur jumlah kuadrat dari selisih antara nilai prediksi dari model regresi dan nilai aktual dalam data. SSE dapat memberikan gambaran tentang seberapa baik model sesuai dengan data.

Dalam konteks regresi linear sederhana, SSE dihitung dengan menjumlahkan kuadrat dari selisih antara nilai prediksi \hat{Y}_i dan nilai aktual Y_i untuk setiap observasi:

$$SSE = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$$

Dalam R, Anda dapat mengakses SSE langsung dari objek model regresi menggunakan fungsi `sum(model$residuals^2)`. Semakin kecil nilai SSE, semakin baik modelnya, karena SSE mencerminkan seberapa baik model dapat memperkirakan nilai aktual dalam dataset. Namun, penggunaan SSE sebaiknya selalu dikombinasikan dengan metrik evaluasi lainnya, seperti R-squared, MSE, atau uji lainnya, untuk mendapatkan gambaran yang lengkap tentang kinerja model.

```
# Menghitung SSE
sse_f <- sum(lm.r$residuals^2)
cat("Sum of Squared Errors (SSE):", sse_f, "\n")
```

```
## Sum of Squared Errors (SSE): 3725.37
```

R-Squared, Nilai R-squared mengindikasikan seberapa besar variasi dalam variabel terikat yang dapat dijelaskan oleh model. Nilai R-squared berkisar antara 0 (model tidak menjelaskan variasi sama sekali) hingga 1 (model menjelaskan seluruh variasi). Anda dapat menghitung R-squared (koefisien determinasi) dari model regresi linear sederhana menggunakan fungsi `summary()` pada objek model

```
# Mengitung R-Square
model.summary <- summary(lm.r)
r_squared_f <- model.summary$r.squared
cat("R-squared:", r_squared_f, "\n")
```

```
## R-squared: 0.3042571
```

Mean Squared Error (MSE) Menghitung MSE dapat memberikan gambaran tentang seberapa baik model dapat menjelaskan variasi dalam data. Semakin rendah MSE dan semakin tinggi R^2 , semakin baik modelnya.

```
# Menghitung MSE
mse <- mean(lm.r$residuals^2)
cat("Mean Squared Error (MSE):", mse, "\n")
```

```
## Mean Squared Error (MSE): 26.60978
```

Metode evaluasi ini memberikan pandangan holistik tentang kinerja model regresi linear sederhana, dan pemilihan metode yang sesuai tergantung pada kebutuhan spesifik analisis Anda.

Latihan

1. Buatlah model regresi untuk hubungan variabel `youtube` terhadap variabel `sales`
2. Buatlah model regresi untuk hubungan variabel `newspaper` terhadap variabel `sales`

1. Model Regresi Youtube terhadap Sales

```
# menghitung korelasi antar variabel
koryoutube <- cor(marketing$youtube, marketing$sales)
koryoutube
```

```
## [1] 0.7822244
```

Nilai korelasi dari kedua variabel adalah 0.7822244

Persamaan regresi dari youtube terhadap sales:

$$Sales = \beta_0 + \beta_1 youtube + \varepsilon$$

```
# liner regresi sederhana pada data training
lm.y <- lm(sales ~ youtube,
           data = trainingset)
summary(lm.y)
```

```
##
## Call:
## lm(formula = sales ~ youtube, data = trainingset)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -9.5608 -2.2749 -0.1501  2.5482  9.0311
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.385104   0.652557   12.85  <2e-16 ***
## youtube      0.046186   0.003267   14.14  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.981 on 138 degrees of freedom
## Multiple R-squared:  0.5916, Adjusted R-squared:  0.5886
## F-statistic: 199.9 on 1 and 138 DF,  p-value: < 2.2e-16
```

Berdasarkan output diatas bisa dijelaskan bahwa: Persamaan regresi linernya adalah

$$sales = 8.215839 + 0.049680 * youtube$$

Misal youtube = 3500

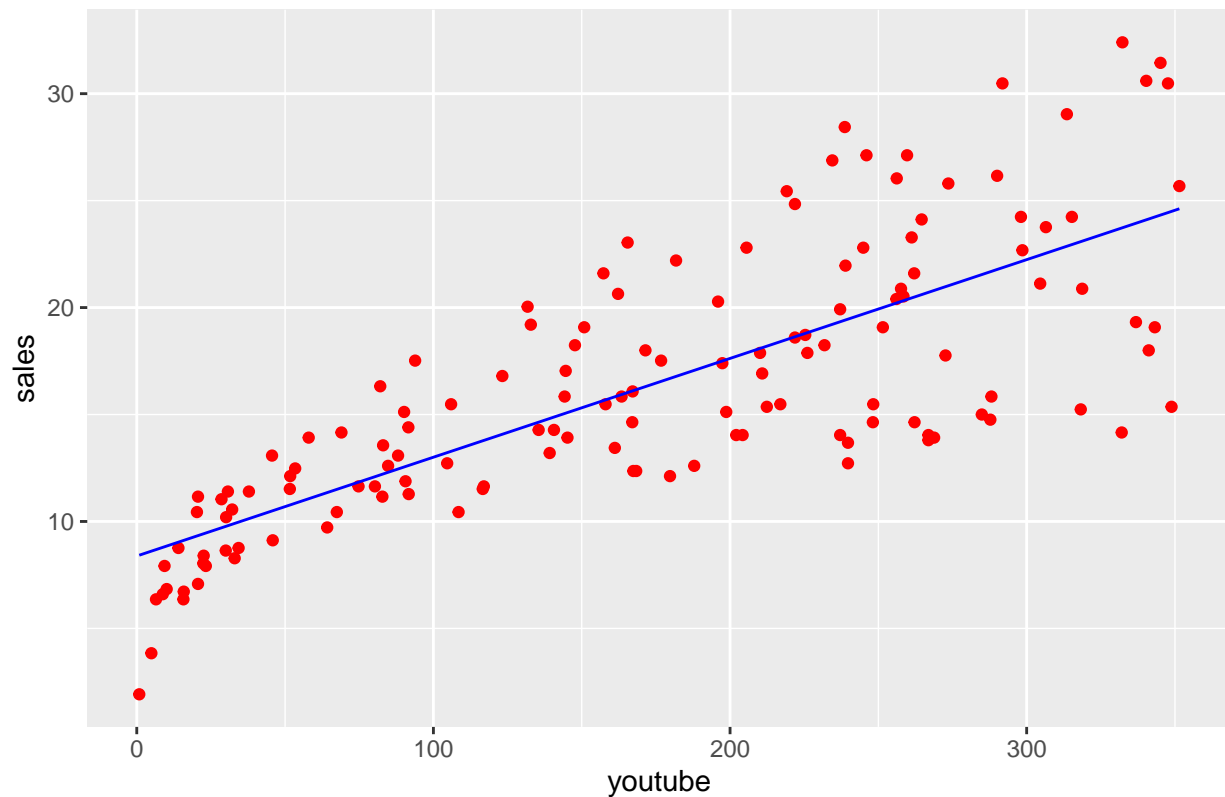
```
sales.y = 8.215839 + (0.049680*3500)
sales.y
```

```
## [1] 182.0958
```

Visualasi hasil data training youtube

```
ggplot() + geom_point(aes(x = trainingset$youtube,
                          y = trainingset$sales), colour = 'red') +
  geom_line(aes(x = trainingset$youtube,
                y = predict(lm.y, newdata = trainingset)), colour = 'blue') +
  ggtitle('Pengaruh Youtube terhadap Sales (Data Training)') +
  xlab('youtube') +
  ylab('sales')
```

Pengaruh Youtube terhadap Sales (Data Training)



Prediksi model regresi dengan data testing

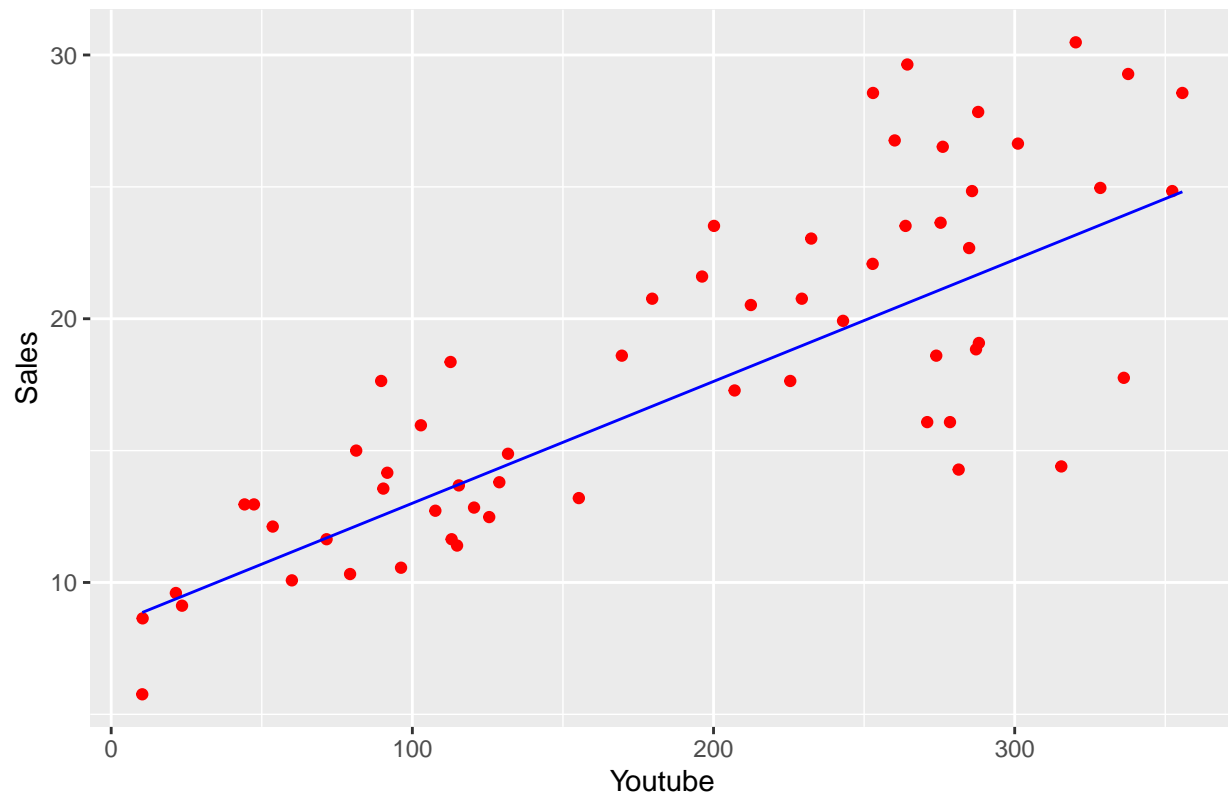
```
yprediksi.y <- predict(lm.y, newdata = testingset)
head(yprediksi.y)
```

```
##          1          6          9         11         17         18
## 21.138094  8.867290  8.861748 12.048610 12.142830 23.981329
```

Visualasi hasil data testing

```
ggplot() + geom_point(aes(x = testingset$youtube,
                           y = testingset$sales), colour = 'red') +
  geom_line(aes(x = testingset$youtube,
                y = predict(lm.y, newdata = testingset)), colour = 'blue') +
  ggtitle('Pengaruh Youtube terhadap Sales (Data Testing)') +
  xlab('Youtube') +
  ylab('Sales')
```

Pengaruh Youtube terhadap Sales (Data Testing)



Latihan ## Evaluasi Lakukan evaluasi dari model yang sudah dibuat dengan menggunakan SSE, MSE dan R-Squared

```
# Evaluasi SSE youtube terhadap sales
```

```
sse_y <- sum(lm.y$residuals^2)
cat("Sum of Squared Errors (SSE):", sse_y, "\n")
```

```
## Sum of Squared Errors (SSE): 2186.744
```

```
# Evaluasi RSquare youtube terhadap sales
```

```
model.summary_y <- summary(lm.y)
r_squared_y <- model.summary_y$r.squared
cat("R-squared:", r_squared_y, "\n")
```

```
## R-squared: 0.5916079
```

```
# Evaluasi MSE youtube terhadap sales
```

```
mse_y <- mean(lm.y$residuals^2)
cat("Mean Squared Error (MSE):", mse_y, "\n")
```

```
## Mean Squared Error (MSE): 15.6196
```

2. Model Regresi Newspaper terhadap Sales

Buatlah model regresi untuk hubungan variabel `newspaper` terhadap variabel `sales`

```
# Melihar korelasi newspaper dengan sales
```

```
kornewspaper <- cor(marketing$newspaper, marketing$sales)
kornewspaper
```

```
## [1] 0.228299
```

Persamaan regresi liner antara `newspaper` dengan `sales`

$$Sales = \beta_0 + \beta_1 \cdot newspaper + \varepsilon$$

```
# Model regresi liner newspaper terhadap sales
```

```
lm.n <- lm(sales ~ newspaper,
           data = trainingset)
summary(lm.n)
```

```
##
## Call:
## lm(formula = sales ~ newspaper, data = trainingset)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.5257  -3.6994  -0.7476   3.8596  15.0416
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  13.68008    0.91880  14.889  < 2e-16 ***
## newspaper     0.07333    0.02155   3.403 0.000872 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.983 on 138 degrees of freedom
## Multiple R-squared:  0.07743,    Adjusted R-squared:  0.07074
## F-statistic: 11.58 on 1 and 138 DF,  p-value: 0.0008718
```

$$Sales = 14.44853 + 0.07067 * newspaper + \varepsilon$$

Misal nilai `newspaper`nya: 3500 maka nilai:

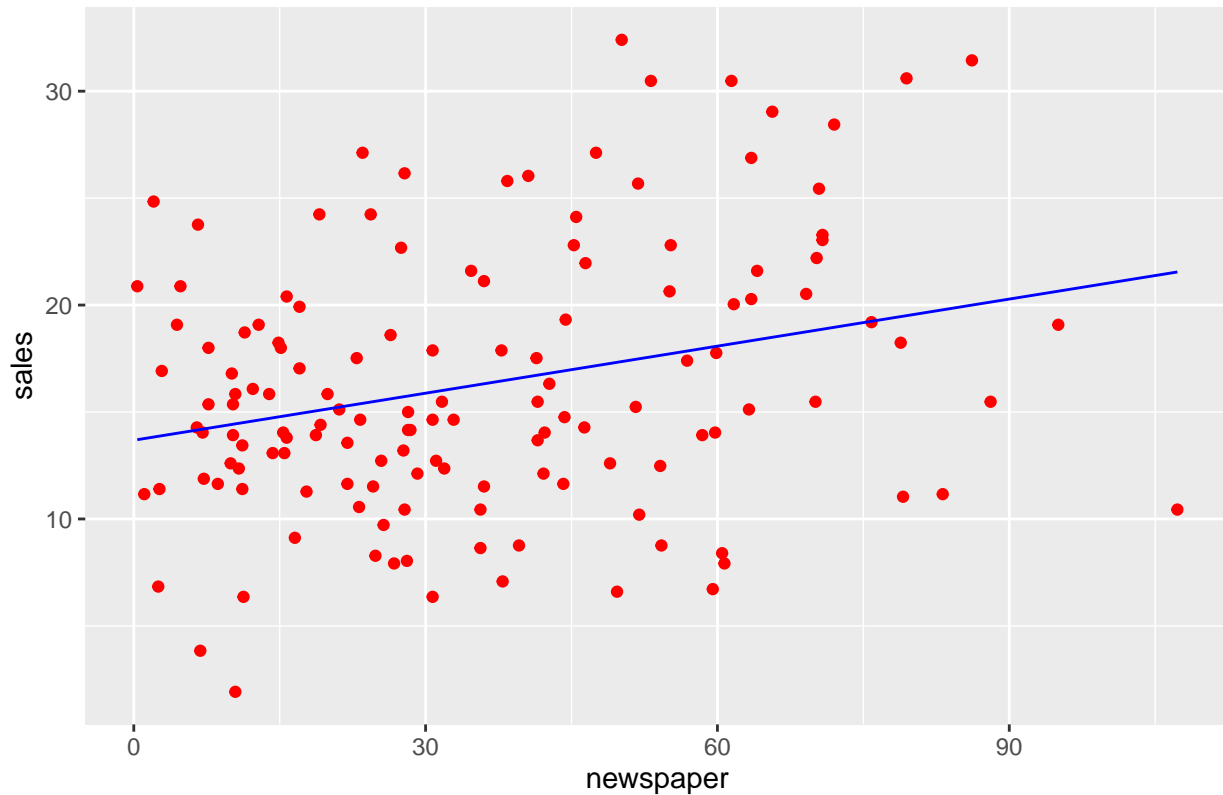
```
sales_n = 14.44853 + (0.07067 * 3500)
sales_n
```

```
## [1] 261.7935
```

Visualasi hasil data training `newspaper`


```
ggplot() + geom_point(aes(x = trainingset$newspaper,
                           y = trainingset$sales), colour = 'red') +
  geom_line(aes(x = trainingset$newspaper,
                y = predict(lm.n, newdata = trainingset)), colour = 'blue') +
  ggtitle('Pengaruh Newspaper terhadap Sales (Data Training)') +
  xlab('newspaper') +
  ylab('sales')
```

Pengaruh Newspaper terhadap Sales (Data Training)



Prediksi model dengan data testing

```
prediksi.n <- predict(lm.n, newdata = testingset)
head(prediksi.n)
```

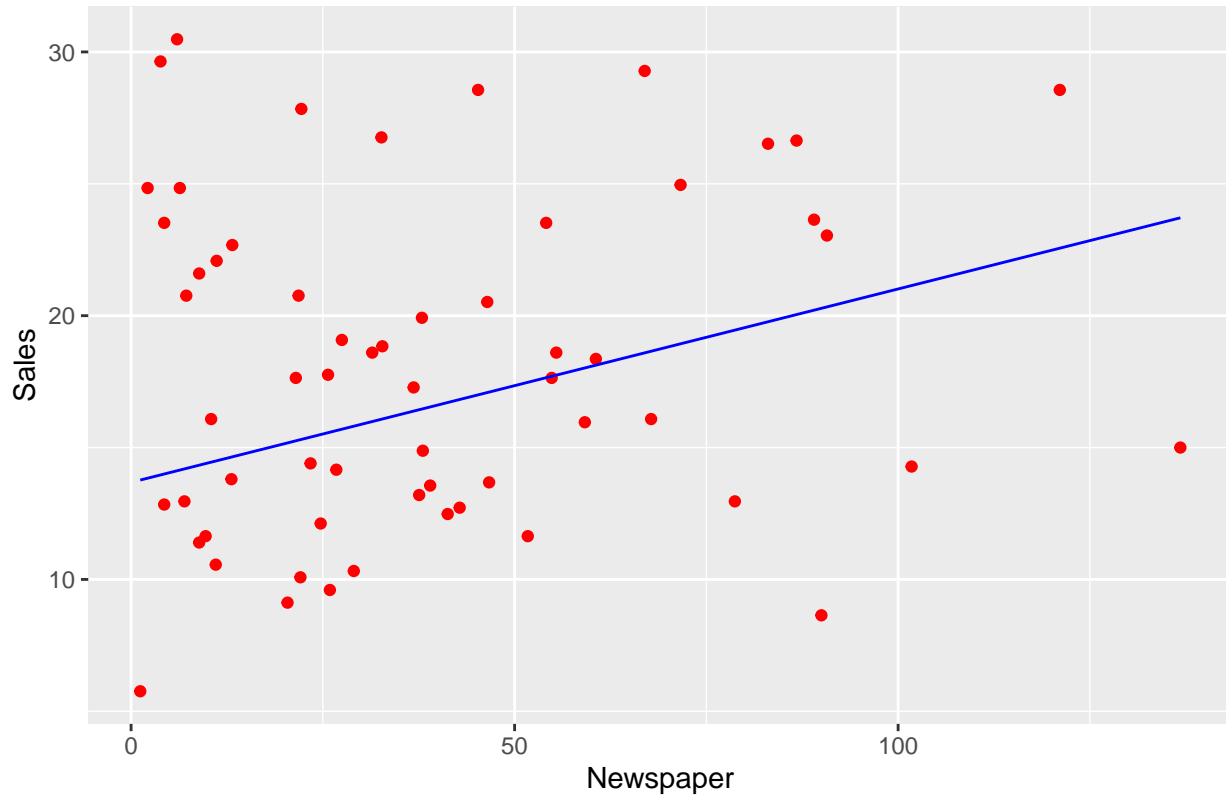
```
##      1      6      9     11     17     18
## 19.76954 20.27993 13.76808 15.80963 23.71185 18.59037
```

Visualasi hasil data testing

```
ggplot() + geom_point(aes(x = testingset$newspaper,
                           y = testingset$sales), colour = 'red') +
  geom_line(aes(x = testingset$newspaper,
                y = predict(lm.n, newdata = testingset)), colour = 'blue') +
  ggtitle('Pengaruh Newspaper terhadap Sales (Data Testing)')
```

```
xlab('Newspaper') +
ylab('Sales')
```

Pengaruh Newspaper terhadap Sales (Data Testing)



Evaluasi model

```
# Evaluasi SSE newspaper terhadap sales
```

```
sse_n <- sum(lm.n$residuals^2)
cat("Sum of Squared Errors (SSE)", sse_n, "\n")
```

Sum of Squared Errors (SSE) 4939.921

```
# Evaluasi RSquare newspaper terhadap sales
```

```
model.summary_n = summary(lm.n)
r_squared_n <- model.summary_n$r.squared
cat("R-squared:", r_squared_n, "\n")
```

R-squared: 0.07742982

```
# Evaluasi MSE newspaper terhadap sales
```

```
mse_n <- mean(lm.n$residuals^2)
cat("Mean Squared Error (MSE):", mse_n, "\n")
```

Mean Squared Error (MSE): 35.28515

```

model_performance <- data.frame(
  No = c(1:3),
  Variable = c("FACEBOOK", "YOUTUBE", "NEWSPAPER"),
  SSE = c(sse_f, sse_y, sse_n),
  Rsquare = c(r_squared_f, r_squared_y, r_squared_n),
  MSE = c(mse, mse_y, mse_n),
  stringsAsFactors = FALSE
)

model_performance

```

```

##   No Variable      SSE      Rsquare      MSE
## 1  1 FACEBOOK 3725.370 0.30425714 26.60978
## 2  2  YOUTUBE 2186.744 0.59160790 15.61960
## 3  3 NEWSPAPER 4939.921 0.07742982 35.28515

```

Kesimpulan: Berdasarkan 3 variabel prediktor yaitu facebook, youtube, dan newspaper yang paling berpengaruh terhadap peningkatan penjualan adalah variabel youtube. Hal ini bisa dijelaskan dari nilai evaluasi SSE dan MSE bahwa model youtube lebih kecil dan RSquare lebih besar dibandingkan dengan model regresi dari 2 variabel lainnya.