PRAKTIKUM PEMODELAN STATISTIKA

MODUL 5

Disusun oleh :

Nama : Fidelia Ping

NIM : 245410012

Kelas : Informatika 1

PROGRAM STUDI INFORMATIKA

PROGRAM SARJANA

FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA

YOGYAKARTA

2025

MODUL 5
REGRESI DENGAN DUMMY VARIABEL

A. TUJUAN PRAKTIKUM

1. Memahami konsep dasar regresi dengan dummy variabel.
2. Melakukan analisis regresi dengan dummy variabel menggunakan R.

B. PEMBAHASAN LISTING

PRAKTIK

Kasus 1

Berikut ini contoh kasus untuk memprediksi gaji karyawan berdasarkan pengalaman kerja (variabel numerik) dan jenis kelamin (variabel kategori biner).

Variabel:

Dependen: Gaji (numerik)

Independen Numerik: Pengalaman kerja (tahun)

Independen Kategori Biner: Jenis kelamin (Laki-laki = 0, Perempuan = 1)oh t

Langkah 1: Persiapan Data

```
# Membuat dataset
data <- data.frame(
  Gaji = c(25, 30, 28, 35, 32, 33, 29, 34, 31, 35),
  Pengalaman = c(2, 5, 3, 7, 6, 4, 3, 8, 5, 6),
  JenisKelamin = c("Laki-laki", "Perempuan", "Laki-laki", "Perempuan", "Laki-
laki",
                   "Perempuan", "Laki-laki", "Perempuan", "Laki-laki",
"Perempuan")
)
# Mengubah JenisKelamin menjadi factor
data$JenisKelamin <- as.factor(data$JenisKelamin)
# Memeriksa tipe data
str(data)
```

Pembahasan : Kode di atas digunakan untuk membuat sebuah dataset sederhana yang berisi tiga variabel, yaitu **Gaji**, **Pengalaman**, dan **JenisKelamin**. Variabel *Gaji* berisi data numerik berupa gaji karyawan dalam satuan juta rupiah, *Pengalaman* berisi lama pengalaman kerja dalam tahun, sedangkan *JenisKelamin* berisi data kategorik berupa jenis kelamin karyawan. Setelah dataset dibuat menggunakan data.frame(), variabel *JenisKelamin* kemudian diubah menjadi tipe **factor** menggunakan as.factor() agar dapat dikenali sebagai variabel kategorik, khususnya jika dataset ini akan digunakan dalam analisis statistik atau pemodelan regresi. Terakhir, fungsi str(data) digunakan untuk melihat struktur data, termasuk tipe masing-masing variabel.

Output

```
'data.frame':   10 obs. of  3 variables:
 $ Gaji        : num  25 30 28 35 32 33 29 34 31 35
 $ Pengalaman  : num  2 5 3 7 6 4 3 8 5 6
 $ JenisKelamin: Factor w/ 2 levels "Laki-laki","Perempuan": 1 2 1 2 1 2 1 2 1 2
```

Pembahasan Output : Output dari fungsi str(data) menampilkan bahwa dataset terdiri dari **10 observasi dan 3 variabel**, di mana variabel *Gaji* dan *Pengalaman* bertipe numerik (num), sedangkan *JenisKelamin* bertipe faktor (Factor) dengan dua level, yaitu **"Laki-laki"** dan **"Perempuan"**. Hal ini menunjukkan bahwa data telah tersusun dengan benar dan tipe data masing-masing variabel sudah sesuai, sehingga dataset siap digunakan

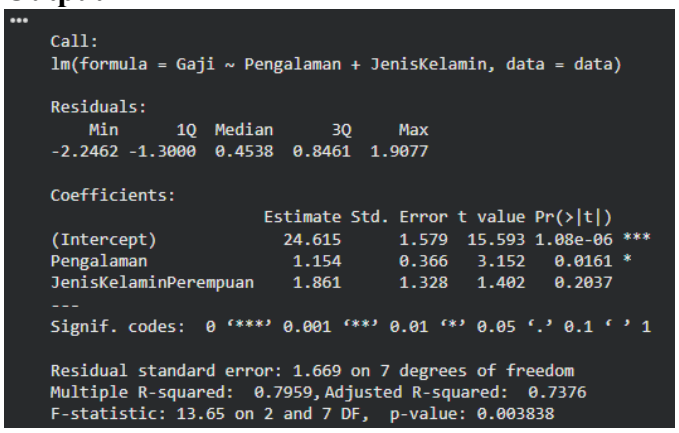
untuk analisis lanjutan seperti pembuatan model regresi linier yang melibatkan variabel numerik dan kategorik.

Langkah 2: Bangun model Regresi

```
# Membuat model regresi linier
model <- lm(Gaji ~ Pengalaman + JenisKelamin, data = data)

# Menampilkan ringkasan model
summary(model)
```

Pembahasan : Kode tersebut digunakan untuk membangun sebuah **model regresi linier** dengan variabel **Gaji** sebagai variabel dependen, serta **Pengalaman** dan **JenisKelamin** sebagai variabel independen. Fungsi lm() digunakan untuk membentuk hubungan linier antara gaji dengan pengalaman kerja dan jenis kelamin berdasarkan data yang telah dibuat sebelumnya. Setelah model terbentuk, fungsi summary(model) dipanggil untuk menampilkan ringkasan hasil regresi yang berisi informasi penting mengenai koefisien regresi, tingkat signifikansi, serta kualitas model secara keseluruhan.

Output



```
Call:
lm(formula = Gaji ~ Pengalaman + JenisKelamin, data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-2.2462 -1.3000  0.4538  0.8461  1.9077

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)          24.615      1.579  15.593 1.08e-06 ***
Pengalaman            1.154      0.366   3.152   0.0161 *
JenisKelaminPerempuan 1.861      1.328   1.402   0.2037
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.669 on 7 degrees of freedom
Multiple R-squared:  0.7959,Adjusted R-squared:  0.7376
F-statistic: 13.65 on 2 and 7 DF,  p-value: 0.003838
```

Pembahasan Output : Output dari summary(model) menampilkan nilai koefisien untuk setiap variabel, yaitu **intercept**, **Pengalaman**, dan **JenisKelamin**, yang menunjukkan seberapa besar pengaruh masing-masing variabel terhadap gaji. Koefisien *Pengalaman* menggambarkan perubahan rata-rata gaji untuk setiap penambahan satu tahun pengalaman kerja, sedangkan koefisien *JenisKelamin* menunjukkan perbedaan rata-rata gaji antara kategori jenis kelamin tertentu dengan kategori referensi. Selain itu, output juga menampilkan nilai **R-squared** dan **Adjusted R-squared** yang menunjukkan seberapa baik model menjelaskan variasi data gaji, serta nilai **p-value** yang digunakan untuk menilai signifikansi statistik dari setiap variabel dalam model.

Langkah 3: Validasi Model

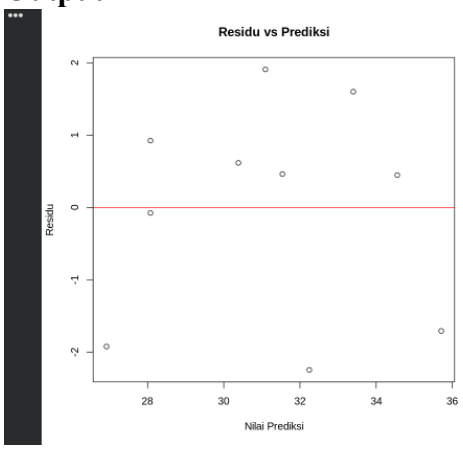
1. Linearitas

```
# Plot residu vs prediksi
   plot(fitted(model), residuals(model),
        main = "Residu vs Prediksi", xlab = "Nilai Prediksi", ylab = "Residu")
   abline(h = 0, col = "red")
```

Pembahasan : Kode tersebut digunakan untuk membuat grafik **residu terhadap nilai prediksi** pada model regresi linier yang telah dibangun. Fungsi fitted(model) menghasilkan nilai prediksi gaji dari model, sedangkan residuals(model) menghasilkan nilai residu, yaitu selisih antara nilai gaji aktual dan nilai gaji prediksi. Fungsi plot() digunakan untuk memvisualisasikan hubungan antara nilai prediksi dan residu, dengan

judul serta label sumbu yang menjelaskan isi grafik. Selanjutnya, fungsi abline(h = 0, col = "red") menambahkan garis horizontal berwarna merah pada nilai residu nol sebagai acuan untuk melihat apakah residu menyebar secara seimbang di sekitar nol.

Output



Pembahasan Output : Output berupa grafik menunjukkan sebaran titik-titik residu terhadap nilai prediksi. Jika titik-titik residu menyebar secara acak di sekitar garis nol tanpa membentuk pola tertentu, maka hal ini mengindikasikan bahwa asumsi linearitas dan homoskedastisitas pada model regresi telah terpenuhi. Sebaliknya, jika terlihat pola tertentu seperti bentuk kipas atau lengkungan, hal tersebut menandakan adanya kemungkinan pelanggaran asumsi model, sehingga model regresi linier perlu dievaluasi atau diperbaiki.

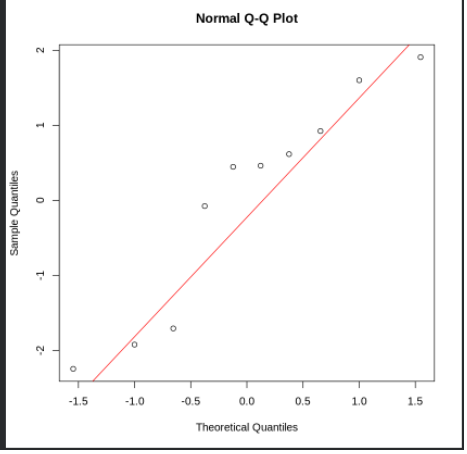
2. Normalitas Residu

```
# Q-Q plot
   qqnorm(residuals(model))
   qqline(residuals(model), col = "red")

   # Uji Shapiro-Wilk
   shapiro.test(residuals(model))
```

Pembahasan : Kode tersebut digunakan untuk menguji **asumsi normalitas residu** pada model regresi linier. Fungsi qqnorm(residuals(model)) membuat Q–Q plot yang membandingkan distribusi residu model dengan distribusi normal teoritis, sedangkan qqline(residuals(model), col = "red") menambahkan garis referensi berwarna merah untuk memudahkan pengamatan kesesuaian residu terhadap distribusi normal. Selanjutnya, fungsi shapiro.test(residuals(model)) digunakan untuk melakukan **uji Shapiro–Wilk**, yaitu uji statistik formal yang bertujuan untuk menilai apakah residu model berdistribusi normal atau tidak.

Output

```
        Shapiro-Wilk normality test

data:  residuals(model)
W = 0.90066, p-value = 0.2228
```



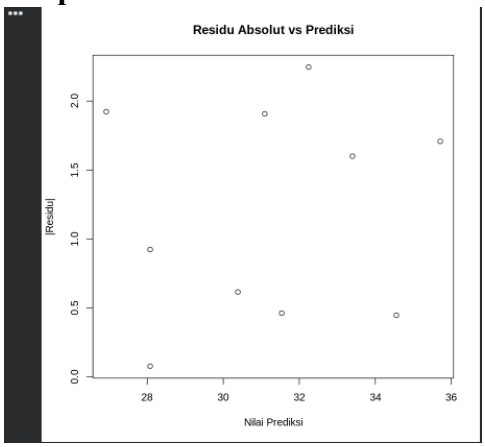
Pembahasan Output : Output Q–Q plot menunjukkan titik-titik residu yang diplot terhadap kuantil distribusi normal. Apabila titik-titik tersebut mengikuti dan berada di sekitar garis merah, maka residu dapat dianggap berdistribusi normal. Selain itu, hasil uji Shapiro–Wilk menampilkan nilai **W** dan **p-value**. Jika nilai p-value lebih besar dari 0,05, maka hipotesis nol yang menyatakan bahwa residu berdistribusi normal dapat diterima, sehingga asumsi normalitas terpenuhi. Sebaliknya, jika p-value kurang dari 0,05, maka residu tidak berdistribusi normal dan model regresi perlu ditinjau kembali.

3. Homoskedastisitas

```
   # Plot residu vs prediksi untuk memeriksa variansi konstan
   plot(fitted(model), abs(residuals(model)),
        main = "Residu Absolut vs Prediksi", xlab = "Nilai Prediksi", ylab
= "|Residu|")
```

Pembahasan : Kode tersebut digunakan untuk memeriksa **asumsi variansi konstan (homoskedastisitas)** pada model regresi linier. Fungsi fitted(model) menghasilkan nilai prediksi dari model, sedangkan abs(residuals(model)) menghitung nilai absolut dari residu. Grafik dibuat menggunakan fungsi plot() dengan sumbu-x berupa nilai prediksi dan sumbu-y berupa residu absolut, sehingga memudahkan pengamatan apakah besar kecilnya residu berubah seiring dengan perubahan nilai prediksi.

Output



Pembahasan Output : Output berupa grafik **residu absolut terhadap nilai prediksi** menunjukkan pola sebaran residu. Jika titik-titik pada grafik tersebar secara acak dan relatif merata di sepanjang sumbu-x tanpa membentuk pola tertentu, maka dapat disimpulkan bahwa variansi residu bersifat konstan dan asumsi homoskedastisitas terpenuhi. Namun, jika terlihat pola tertentu seperti semakin melebar atau menyempit (pola kipas), hal tersebut mengindikasikan adanya heteroskedastisitas, sehingga model regresi linier perlu dilakukan perbaikan atau transformasi data.

Evaluasi Performa Model

```
# Menghitung RMSE
prediksi <- predict(model, data)
rmse <- sqrt(mean((data$Gaji - prediksi)^2))
cat("RMSE: ", rmse, "\n")
```

Pembahasan : Kode tersebut digunakan untuk menghitung **Root Mean Square Error (RMSE)** sebagai ukuran tingkat kesalahan prediksi dari model regresi linier. Fungsi predict(model, data) digunakan untuk menghasilkan nilai gaji hasil prediksi berdasarkan data yang ada, kemudian selisih antara gaji aktual (data$Gaji) dan gaji prediksi dihitung serta dikuadratkan. Nilai kuadrat selisih tersebut dirata-ratakan menggunakan mean(), lalu diakarkan dengan sqrt() untuk memperoleh nilai RMSE. Terakhir, fungsi cat() digunakan untuk menampilkan nilai RMSE ke layar.

Output

```
••• RMSE:  1.396699
```

Pembahasan Output : Output yang dihasilkan berupa satu nilai RMSE yang menunjukkan rata-rata besar kesalahan prediksi model dalam satuan yang sama dengan variabel gaji, yaitu juta rupiah. Semakin kecil nilai RMSE, semakin baik kemampuan model dalam memprediksi gaji karena kesalahan prediksinya semakin rendah. Sebaliknya, nilai RMSE yang besar menandakan bahwa prediksi model masih kurang akurat dan perlu dilakukan evaluasi atau perbaikan model lebih lanjut.

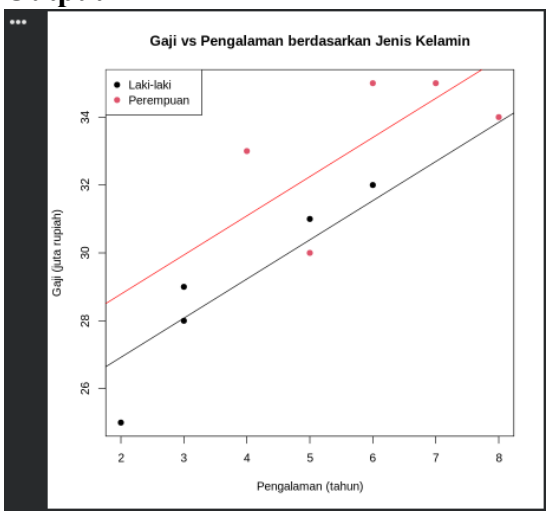
Langkah 4: Visualisasi dan Prediksi

```
# Visualisasi hubungan Pengalaman vs Gaji, dengan warna berbeda untuk
JenisKelamin
plot(data$Pengalaman, data$Gaji, col = data$JenisKelamin,
     pch = 19, main = "Gaji vs Pengalaman berdasarkan Jenis Kelamin",
     xlab = "Pengalaman (tahun)", ylab = "Gaji (juta rupiah)")
legend("topleft", legend = levels(data$JenisKelamin), col = 1:2, pch = 19)

# Menambahkan garis regresi untuk masing-masing kelompok
abline(a = coef(model)[1], b = coef(model)[2], col = "black") # Laki-laki
abline(a = coef(model)[1] + coef(model)[3], b = coef(model)[2], col = "red")
# Perempuan
```

Pembahasan : Kode tersebut digunakan untuk memvisualisasikan hubungan antara **Pengalaman kerja** dan **Gaji** dengan membedakan **Jenis Kelamin** menggunakan warna yang berbeda. Fungsi plot() menampilkan diagram pencar dengan sumbu-x berupa pengalaman (tahun) dan sumbu-y berupa gaji (juta rupiah), di mana argumen col = data$JenisKelamin memberikan warna berbeda untuk setiap kategori jenis kelamin, sedangkan pch = 19 digunakan agar titik data terlihat jelas. Fungsi legend() ditambahkan untuk menjelaskan arti warna pada grafik. Selanjutnya, fungsi abline() digunakan untuk menambahkan garis regresi, di mana garis pertama merepresentasikan hubungan gaji dan

pengalaman untuk kelompok referensi (Laki-laki), sedangkan garis kedua menunjukkan garis regresi untuk kelompok Perempuan yang diperoleh dari penyesuaian koefisien intersep pada model regresi.

Output



Pembahasan Output : Output berupa grafik menunjukkan sebaran titik-titik data gaji terhadap pengalaman kerja yang dibedakan berdasarkan jenis kelamin. Dari grafik tersebut dapat diamati kecenderungan bahwa gaji meningkat seiring bertambahnya pengalaman kerja, serta adanya perbedaan posisi garis regresi antara kelompok Laki-laki dan Perempuan yang mencerminkan perbedaan rata-rata gaji pada tingkat pengalaman yang sama. Visualisasi ini membantu memahami pengaruh pengalaman dan jenis kelamin terhadap gaji secara lebih intuitif serta memperjelas hasil yang diperoleh dari model regresi linier.

```
# Prediksi untuk data baru: Pengalaman = 4 tahun, JenisKelamin = Perempuan
new_data <- data.frame(Pengalaman = 4, JenisKelamin = "Perempuan")
prediksi <- predict(model, newdata = new_data)
cat("Prediksi gaji untuk pengalaman 4 tahun dan Perempuan: ", prediksi, "juta
rupiah\n")

# Prediksi untuk data baru: Pengalaman = 4 tahun, JenisKelamin = Laki-laki
# Perbaikan: Spasi pada " Laki-laki " dihapus agar sesuai dengan format
kategori data asli
new_data <- data.frame(Pengalaman = 4, JenisKelamin = "Laki-laki")
prediksi <- predict(model, newdata = new_data)
cat("Prediksi gaji untuk pengalaman 4 tahun dan Laki-laki: ", prediksi, "juta
rupiah\n")
```

Pembahasan : Kode tersebut digunakan untuk melakukan **prediksi gaji** menggunakan model regresi linier berdasarkan data baru dengan pengalaman kerja selama 4 tahun dan jenis kelamin tertentu. Pada bagian pertama, dibuat data baru dengan pengalaman 4 tahun dan jenis kelamin *Perempuan*, kemudian fungsi predict() digunakan untuk menghitung nilai gaji yang diperkirakan oleh model, dan hasilnya ditampilkan menggunakan cat(). Pada bagian kedua, proses yang sama dilakukan untuk jenis kelamin *Laki-laki*. Perbaikan pada kode dilakukan dengan memastikan penulisan kategori *JenisKelamin* sesuai dengan kategori yang terdapat pada data asli, sehingga tidak menimbulkan kesalahan saat proses prediksi.

Output

```
Prediksi gaji untuk pengalaman 4 tahun dan Perempuan:  31.09231 juta rupiah
Prediksi gaji untuk pengalaman 4 tahun dan Laki-laki:  29.23077 juta rupiah
```

Pembahasan Output : Output yang dihasilkan berupa nilai prediksi gaji dalam satuan juta rupiah untuk karyawan dengan pengalaman kerja 4 tahun berdasarkan jenis kelamin. Nilai prediksi tersebut mencerminkan hasil perhitungan model regresi yang mempertimbangkan pengaruh pengalaman kerja serta perbedaan jenis kelamin terhadap gaji. Perbedaan hasil prediksi antara Perempuan dan Laki-laki menunjukkan adanya kontribusi variabel jenis kelamin dalam model, sehingga model dapat digunakan untuk memperkirakan gaji pada kondisi tertentu sesuai dengan data yang telah dianalisis.

Kasus 2

Berikut ini akan dilakukan analisis regresi linier ganda untuk memprediksi penjualan toko (dalam juta rupiah) berdasarkan pengeluaran iklan, jumlah karyawan dan tipe toko, dengan

Variabel Dependen (Y): Penjualan (numerik)

Variabel Independen Numerik:

Pengeluaran iklan (juta rupiah)

Jumlah karyawan

Variabel Independen Kategorikal: Tipe toko (Kecil, Sedang, Besar).

```
# --- Langkah 1: Pengelolaan Data ---

# 1. Buat dataset hipotetis
data <- data.frame(
  Penjualan = c(50, 60, 55, 80, 75, 65, 70, 85, 90, 100, 95, 70),
  Iklan = c(5, 7, 6, 10, 8, 7, 6, 12, 15, 14, 13, 8),
  Karyawan = c(3, 4, 3, 6, 5, 4, 5, 7, 8, 9, 7, 5),
  TipeToko = c("Kecil", "Sedang", "Kecil", "Besar", "Sedang", "Kecil",
               "Sedang", "Besar", "Besar", "Besar", "Sedang", "Kecil")
)

# Mengubah TipeToko menjadi factor
data$TipeToko <- as.factor(data$TipeToko)

# Memeriksa tipe data
str(data)
```

Pembahasan : Kode tersebut digunakan pada tahap **pengelolaan data** dengan membuat sebuah dataset hipotetis yang merepresentasikan data penjualan toko. Dataset terdiri dari empat variabel, yaitu **Penjualan** sebagai variabel numerik yang menunjukkan jumlah penjualan, **Iklan** sebagai biaya atau intensitas iklan, **Karyawan** sebagai jumlah tenaga kerja, dan **TipeToko** sebagai variabel kategorik yang menunjukkan skala toko (Kecil, Sedang, Besar). Setelah dataset dibuat menggunakan data.frame(), variabel *TipeToko* diubah menjadi tipe **factor** dengan as.factor() agar dikenali sebagai data kategorik yang dapat digunakan dalam analisis statistik atau pemodelan regresi. Terakhir, fungsi str(data) digunakan untuk menampilkan struktur dataset dan memastikan setiap variabel memiliki tipe data yang sesuai.

Output

```
'data.frame':   12 obs. of  4 variables:
 $ Penjualan: num  50 60 55 80 75 65 70 85 90 100 ...
 $ Iklan    : num  5 7 6 10 8 7 6 12 15 14 ...
 $ Karyawan : num  3 4 3 6 5 4 5 7 8 9 ...
 $ TipeToko : Factor w/ 3 levels "Besar","Kecil",..: 2 3 2 1 3 2 3 1 1 1 ...
```

Pembahasan Output : Output dari fungsi str(data) menunjukkan bahwa dataset memiliki **12 observasi dan 4 variabel**, di mana variabel *Penjualan*, *Iklan*, dan *Karyawan* bertipe numerik (num), sedangkan variabel *TipeToko* bertipe faktor (Factor) dengan beberapa level, yaitu **Kecil**, **Sedang**, dan **Besar**. Hal ini menandakan bahwa data telah berhasil dikelompokkan dengan benar antara variabel numerik dan kategorik, sehingga dataset siap digunakan untuk analisis lanjutan, seperti pembuatan model regresi linier untuk melihat pengaruh iklan, jumlah karyawan, dan tipe toko terhadap penjualan.

Langkah 2: Bangun Model Regresi

```
# --- Langkah 2: Bangun Model Regresi ---

# Membuat model regresi linier
model <- lm(Penjualan ~ Iklan + Karyawan + TipeToko, data = data)

# Menampilkan ringkasan model
summary(model)
```

Pembahasan : Kode pada langkah ini digunakan untuk membangun **model regresi linier** dengan variabel **Penjualan** sebagai variabel dependen, serta **Iklan**, **Karyawan**, dan **TipeToko** sebagai variabel independen. Fungsi lm() digunakan untuk memodelkan hubungan antara jumlah penjualan dengan intensitas iklan, jumlah karyawan, dan tipe toko berdasarkan data yang telah disiapkan sebelumnya. Setelah model regresi terbentuk, fungsi summary(model) dipanggil untuk menampilkan ringkasan hasil analisis regresi yang memuat informasi penting terkait pengaruh masing-masing variabel terhadap penjualan serta kualitas model secara keseluruhan.

Output

```
Call:
lm(formula = Penjualan ~ Iklan + Karyawan + TipeToko, data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-4.4605 -1.5856 -0.0149  2.4055  4.8980

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)     21.3806     8.1282   2.630   0.0339 *
Iklan            0.8709     1.0869   0.801   0.4493
Karyawan         7.5021     2.1464   3.495   0.0101 *
TipeTokoKecil    4.8258     4.7582   1.014   0.3442
TipeTokoSedang   6.8309     3.5712   1.913   0.0974 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.729 on 7 degrees of freedom
Multiple R-squared:  0.9649,Adjusted R-squared:  0.9448
F-statistic: 48.11 on 4 and 7 DF,  p-value: 3.546e-05
```

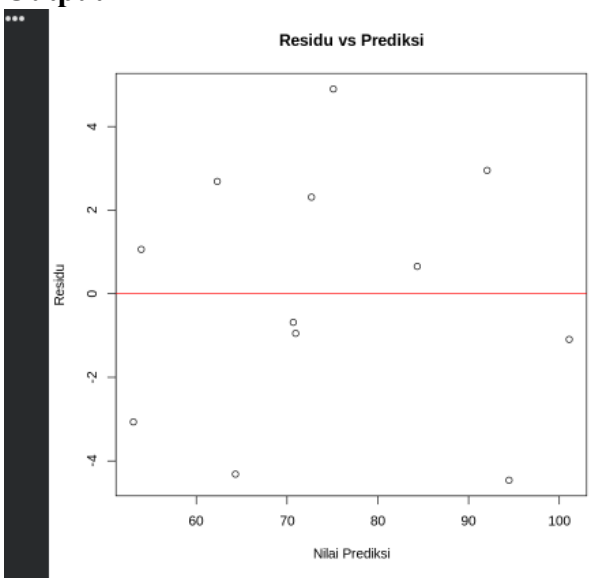
Pembahasan Output : Output dari summary(model) menampilkan nilai koefisien regresi untuk setiap variabel, termasuk **intercept**, **Iklan**, **Karyawan**, serta kategori **TipeToko** yang dibandingkan dengan kategori referensi. Koefisien tersebut menunjukkan besarnya perubahan rata-rata penjualan akibat peningkatan satu satuan pada variabel iklan atau jumlah karyawan, maupun perbedaan penjualan berdasarkan tipe toko. Selain itu, output juga menyajikan nilai **R-squared** dan **Adjusted R-squared** yang menunjukkan seberapa besar variasi penjualan dapat dijelaskan oleh model, serta nilai **p-value** yang digunakan untuk menilai signifikansi statistik dari masing-masing variabel dalam memengaruhi penjualan.

Langkah 4: Validasi Model

```
# a. Linearitas
# Plot residu vs prediksi
plot(fitted(model), residuals(model),
     main = "Residu vs Prediksi",
     xlab = "Nilai Prediksi",
     ylab = "Residu")
abline(h = 0, col = "red")
```

Pembahasan : Kode tersebut digunakan untuk memeriksa **asumsi linearitas** pada model regresi linier yang telah dibangun. Fungsi fitted(model) menghasilkan nilai prediksi penjualan dari model, sedangkan residuals(model) menghasilkan nilai residu, yaitu selisih antara nilai penjualan aktual dan nilai prediksi. Grafik dibuat menggunakan fungsi plot() dengan sumbu-x berupa nilai prediksi dan sumbu-y berupa residu, serta dilengkapi judul dan label sumbu agar mudah dipahami. Selanjutnya, fungsi abline(h = 0, col = "red") menambahkan garis horizontal pada nilai residu nol sebagai acuan visual.

Output



Pembahasan Output : Output berupa grafik **residu terhadap nilai prediksi** yang digunakan untuk mengevaluasi apakah hubungan antara variabel independen dan dependen bersifat linier. Jika titik-titik residu tersebar secara acak di sekitar garis nol tanpa membentuk pola tertentu, maka asumsi linearitas dapat dianggap terpenuhi. Sebaliknya, jika terlihat pola sistematis seperti lengkungan atau tren tertentu, hal tersebut mengindikasikan adanya pelanggaran asumsi linearitas, sehingga model regresi linier perlu ditinjau atau dimodifikasi lebih lanjut.

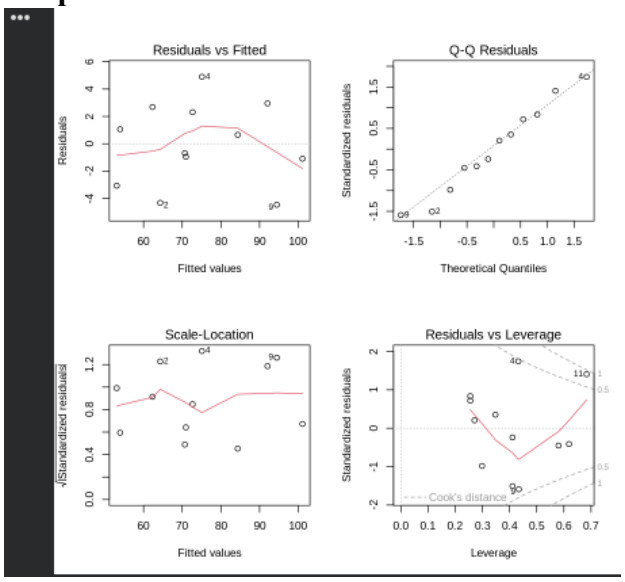
b. Normalitas Residu

```
# b. Normalitas Residu
# Visualisasi diagnostik (membagi layar menjadi 4 bagian)
par(mfrow = c(2, 2))
plot(model)
par(mfrow = c(1, 1)) # Mengembalikan layar ke pengaturan normal
```

Pembahasan : Kode tersebut digunakan untuk melakukan **pemeriksaan normalitas residu dan diagnostik model regresi linier** secara visual. Fungsi par(mfrow = c(2, 2)) digunakan untuk membagi tampilan grafik menjadi empat bagian dalam satu layar, sehingga beberapa grafik diagnostik dapat ditampilkan sekaligus. Perintah plot(model)

kemudian menampilkan empat grafik diagnostik standar dari model regresi linier, yaitu *Residuals vs Fitted*, *Normal Q–Q*, *Scale-Location*, dan *Residuals vs Leverage*. Setelah semua grafik ditampilkan, fungsi par(mfrow = c(1, 1)) digunakan untuk mengembalikan pengaturan tampilan grafik ke kondisi normal.

Output



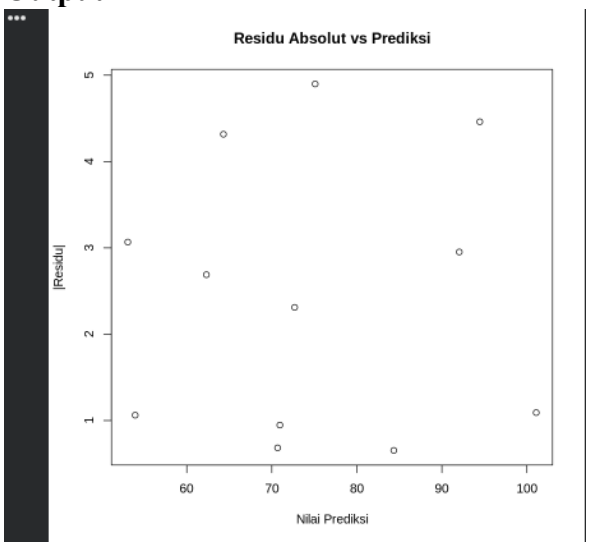
Pembahasan Output : Output berupa empat grafik diagnostik yang memberikan informasi penting mengenai kualitas model regresi. Grafik *Normal Q–Q* digunakan untuk menilai apakah residu berdistribusi normal, di mana titik-titik yang mengikuti garis diagonal menunjukkan normalitas residu yang baik. Grafik *Residuals vs Fitted* dan *Scale-Location* membantu mengevaluasi asumsi linearitas dan kesamaan variansi residu, sedangkan grafik *Residuals vs Leverage* digunakan untuk mengidentifikasi adanya pengamatan yang berpengaruh besar (*outlier* atau *influential points*). Secara keseluruhan, grafik-grafik ini membantu memastikan bahwa asumsi-asumsi regresi linier telah terpenuhi.

c. Homoskedastisitas

```
# c. Homoskedastisitas
# Plot residu absolut vs prediksi
plot(fitted(model), abs(residuals(model)),
     main = "Residu Absolut vs Prediksi",
     xlab = "Nilai Prediksi",
     ylab = "|Residu|")
```

Pembahasan : Kode tersebut digunakan untuk memeriksa **asumsi homoskedastisitas** pada model regresi linier. Fungsi fitted(model) menghasilkan nilai prediksi penjualan dari model, sedangkan abs(residuals(model)) digunakan untuk menghitung nilai absolut residu. Grafik dibuat menggunakan fungsi plot() dengan sumbu-x berupa nilai prediksi dan sumbu-y berupa residu absolut, serta dilengkapi judul dan label sumbu agar mudah dipahami. Visualisasi ini membantu melihat apakah besar kecilnya residu relatif konstan di sepanjang rentang nilai prediksi.

Output



Pembahasan Output : Output berupa grafik **residu absolut terhadap nilai prediksi** menunjukkan pola sebaran residu. Jika titik-titik pada grafik tersebar secara acak dan relatif merata tanpa membentuk pola tertentu, maka dapat disimpulkan bahwa variansi residu bersifat konstan dan asumsi homoskedastisitas terpenuhi. Namun, jika terlihat pola sistematis seperti bentuk kipas atau tren tertentu, hal tersebut mengindikasikan adanya heteroskedastisitas, sehingga model regresi linier perlu dievaluasi atau dilakukan transformasi data.

Evaluasi Performa Model

```
# --- Evaluasi Performa Model ---

# Menghitung RMSE
prediksi <- predict(model, data)
rmse <- sqrt(mean((data$Penjualan - prediksi)^2))
cat("RMSE: ", rmse, "\n")
```

Pembahasan : Kode tersebut digunakan untuk melakukan **evaluasi performa model regresi linier** dengan menghitung nilai **Root Mean Square Error (RMSE)**. Fungsi predict(model, data) digunakan untuk menghasilkan nilai penjualan hasil prediksi berdasarkan model regresi dan data yang sama. Selanjutnya, selisih antara nilai penjualan aktual (data$Penjualan) dan nilai prediksi dihitung, kemudian dikuadratkan dan dirata-ratakan menggunakan mean(). Hasil rata-rata kesalahan kuadrat tersebut kemudian diakarkan dengan sqrt() untuk memperoleh nilai RMSE, yang selanjutnya ditampilkan menggunakan fungsi cat().

Output

```
RMSE:  2.84786
```

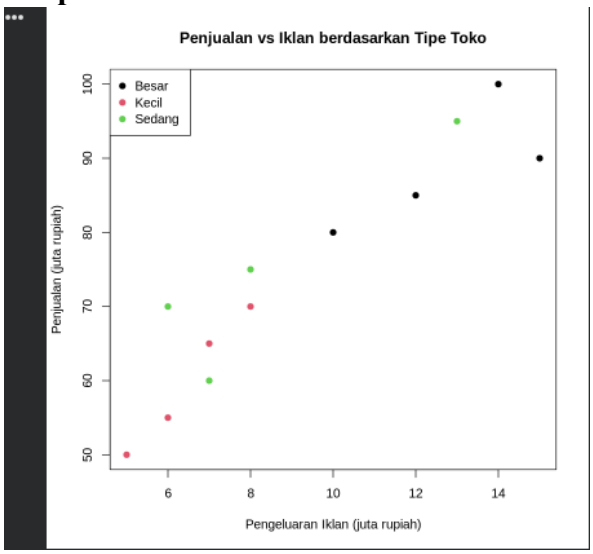
Pembahasan Output : Output yang dihasilkan berupa satu nilai RMSE yang menunjukkan rata-rata besar kesalahan prediksi model dalam satuan yang sama dengan variabel penjualan. Nilai RMSE yang kecil menandakan bahwa model memiliki tingkat akurasi prediksi yang baik, sedangkan nilai RMSE yang besar menunjukkan bahwa model masih memiliki kesalahan prediksi yang cukup tinggi, sehingga diperlukan evaluasi lebih lanjut atau perbaikan model.

Langkah 5: Visualisasi dan Prediksi

```
# --- Langkah 5: Visualisasi dan Prediksi ---

# Visualisasi Penjualan vs Iklan, dengan warna berdasarkan TipeToko
plot(data$Iklan, data$Penjualan, col = data$TipeToko,
     pch = 19, main = "Penjualan vs Iklan berdasarkan Tipe Toko",
      xlab = "Pengeluaran Iklan (juta rupiah)", ylab = "Penjualan (juta
rupiah)")
legend("topleft", legend = levels(data$TipeToko), col = 1:3, pch = 19)
```

Pembahasan : Kode pada langkah ini digunakan untuk melakukan **visualisasi hubungan antara pengeluaran iklan dan penjualan** dengan membedakan **tipe toko** menggunakan warna yang berbeda. Fungsi plot() digunakan untuk membuat diagram pencar dengan sumbu-x berupa pengeluaran iklan dan sumbu-y berupa penjualan, sementara argumen col = data$TipeToko memberikan warna yang berbeda untuk setiap kategori tipe toko (Kecil, Sedang, dan Besar). Simbol titik ditampilkan menggunakan pch = 19 agar data terlihat jelas, serta judul dan label sumbu ditambahkan untuk memperjelas informasi grafik. Selanjutnya, fungsi legend() digunakan untuk menampilkan keterangan warna yang mewakili masing-masing tipe toko pada grafik.

Output



Pembahasan Output : Output berupa grafik menunjukkan sebaran penjualan terhadap pengeluaran iklan yang dibedakan berdasarkan tipe toko. Dari visualisasi ini dapat diamati bahwa secara umum penjualan cenderung meningkat seiring dengan meningkatnya pengeluaran iklan, serta terdapat perbedaan pola penjualan antar tipe toko. Toko dengan skala lebih besar umumnya memiliki penjualan yang lebih tinggi dibandingkan toko kecil dan sedang pada tingkat pengeluaran iklan tertentu. Visualisasi ini membantu memahami pola data secara intuitif dan mendukung interpretasi hasil analisis regresi yang telah dilakukan.

```
# Prediksi untuk data baru: Iklan = 10 juta, Karyawan = 5, TipeToko = Sedang
new_data <- data.frame(Iklan = 10, Karyawan = 5, TipeToko = "Sedang")
prediksi_baru <- predict(model, newdata = new_data)

cat("Prediksi penjualan untuk Iklan = 10, Karyawan = 5, TipeToko = Sedang:
",
    prediksi_baru, "juta rupiah\n")
```

Pembahasan : Kode tersebut digunakan untuk melakukan **prediksi penjualan** berdasarkan data baru menggunakan model regresi linier yang telah dibangun. Data baru dibuat dengan nilai pengeluaran iklan sebesar 10 juta rupiah, jumlah karyawan sebanyak 5 orang, dan tipe toko *Sedang*. Selanjutnya, fungsi predict() digunakan untuk menghitung nilai penjualan yang diperkirakan oleh model berdasarkan kombinasi variabel tersebut. Hasil prediksi kemudian ditampilkan menggunakan fungsi cat() dalam format kalimat agar mudah dibaca.

Output

```
•••  Prediksi penjualan untuk Iklan = 10, Karyawan = 5, TipeToko = Sedang:  74.43076 juta rupiah
```

Pembahasan Output : Output yang dihasilkan berupa nilai prediksi penjualan dalam satuan juta rupiah untuk kondisi pengeluaran iklan 10 juta rupiah, jumlah karyawan 5 orang, dan tipe toko Sedang. Nilai ini mencerminkan estimasi penjualan yang dihasilkan oleh model regresi dengan mempertimbangkan pengaruh iklan, jumlah karyawan, dan tipe toko secara simultan. Hasil prediksi ini dapat digunakan sebagai bahan pertimbangan dalam pengambilan keputusan, misalnya untuk merencanakan strategi pemasaran atau pengelolaan sumber daya toko.

LATIHAN

```
# --- BAGIAN 1: PERSIAPAN DATA (Simulasi Dataset Kaggle) ---
set.seed(123)
n <- 50 # Jumlah data

# Membuat variabel
area <- round(runif(n, min = 50, max = 200), 0) # Luas tanah (m2)
kamar <- sample(2:5, n, replace = TRUE)         # Jumlah kamar
lokasi <- sample(c("PusatKota", "Pinggiran", "Desa"), n, replace = TRUE) #
Dummy Variable

# Rumus harga (Simulasi pola data)
# Harga dasar + (Area * 1.5) + (Kamar * 10) + (Efek Lokasi) + Error
harga_base <- 200 + (area * 1.5) + (kamar * 10)
efek_lokasi  <-  ifelse(lokasi  ==  "PusatKota",  100,  ifelse(lokasi  ==
"Pinggiran", 50, 0))
harga <- harga_base + efek_lokasi + rnorm(n, 0, 10) # Ditambah noise acak

# Membuat Data Frame
df_latihan <- data.frame(
  Harga = harga,
  LuasTanah = area,
  JmlKamar = kamar,
  Lokasi = as.factor(lokasi) # Penting: Ubah kategori jadi factor untuk Dummy
Variable
)

# Cek data
head(df_latihan)
str(df_latihan)
```

Pembahasan : Kode pada bagian ini digunakan untuk **menyiapkan data simulasi** yang menyerupai dataset harga rumah seperti yang sering dijumpai pada platform Kaggle. Proses diawali dengan set.seed(123) untuk memastikan data acak yang dihasilkan dapat direproduksi. Selanjutnya ditentukan jumlah observasi sebanyak 50 data, kemudian dibuat beberapa variabel independen, yaitu **LuasTanah** yang dihasilkan secara acak antara 50–200 m², **JmlKamar** yang diambil secara acak antara 2–5 kamar, serta **Lokasi** sebagai variabel kategorik dengan tiga kategori, yaitu PusatKota, Pinggiran, dan Desa. Variabel **Harga** dibentuk menggunakan rumus simulasi yang menggabungkan harga dasar, pengaruh luas tanah, jumlah kamar, serta efek lokasi, dan ditambahkan *noise* acak untuk menyerupai data nyata. Semua variabel kemudian digabungkan ke dalam sebuah data frame bernama df_latihan, dengan variabel Lokasi diubah menjadi tipe **factor** agar dapat diperlakukan sebagai *dummy variable* dalam analisis regresi. Fungsi head() dan str() digunakan untuk menampilkan sebagian data dan memastikan struktur serta tipe data sudah sesuai.

Output

A data.frame: 6 × 4

	Harga	LuasTanah	JmlKamar	Lokasi
	<dbl>	<dbl>	<int>	<fct>
1	486.3061	93	5	PusatKota
2	522.8454	168	2	Pinggiran
3	498.8153	111	4	PusatKota
4	536.7409	182	2	Pinggiran
5	617.4913	191	4	PusatKota
6	342.1373	57	5	Desa

```
'data.frame':   50 obs. of  4 variables:
 $ Harga    : num  486 523 499 537 617 ...
 $ LuasTanah: num  93 168 111 182 191 57 129 184 133 118 ...
 $ JmlKamar : int  5 2 4 2 4 5 3 2 3 2 ...
 $ Lokasi   : Factor w/ 3 levels "Desa","Pinggiran",..: 3 2 3 2 3 1 1 2 1 3 ...
```

Pembahasan Output : Output dari head(df_latihan) menampilkan enam baris pertama dari dataset yang menunjukkan contoh nilai harga rumah beserta luas tanah, jumlah kamar, dan lokasi. Sementara itu, output str(df_latihan) memperlihatkan bahwa dataset terdiri dari **50 observasi dan 4 variabel**, di mana variabel **Harga** dan **LuasTanah** bertipe numerik, **JmlKamar** bertipe numerik/integer, dan **Lokasi** bertipe faktor dengan tiga level, yaitu PusatKota, Pinggiran, dan Desa. Hal ini menunjukkan bahwa dataset telah tersusun dengan baik dan siap digunakan untuk analisis lanjutan, seperti pembuatan model regresi linier untuk memprediksi harga rumah.

```
# --- BAGIAN 2: ANALISA REGRESI LINIER GANDA ---
# Model: Harga dipengaruhi LuasTanah, JmlKamar, dan Lokasi
model_latihan <- lm(Harga ~ LuasTanah + JmlKamar + Lokasi, data = df_latihan)
```

Pembahasan : Kode pada bagian ini digunakan untuk melakukan **analisis regresi linier ganda** dengan tujuan memodelkan hubungan antara **Harga rumah** sebagai variabel dependen dengan tiga variabel independen, yaitu **LuasTanah**, **JmlKamar**, dan **Lokasi**. Fungsi lm() digunakan untuk membangun model regresi linier berdasarkan data pada df_latihan, di mana variabel Lokasi yang bertipe faktor secara otomatis akan dikonversi

menjadi variabel dummy oleh R. Model ini memungkinkan untuk mengetahui seberapa besar pengaruh masing-masing variabel, baik numerik maupun kategorik, terhadap harga rumah.

Output

```
Call:
lm(formula = Harga ~ LuasTanah + JmlKamar + Lokasi, data = df_latihan)

Residuals:
     Min       1Q   Median       3Q      Max
-19.4610  -4.9150  -0.4791   4.0914  16.6945

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     203.22700    5.21154   39.00  < 2e-16 ***
LuasTanah         1.47680    0.02555   57.80  < 2e-16 ***
JmlKamar         10.69569    1.00541   10.64 7.23e-14 ***
LokasiPinggiran  51.30091    2.48496   20.64  < 2e-16 ***
LokasiPusatKota  95.17014    3.02784   31.43  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

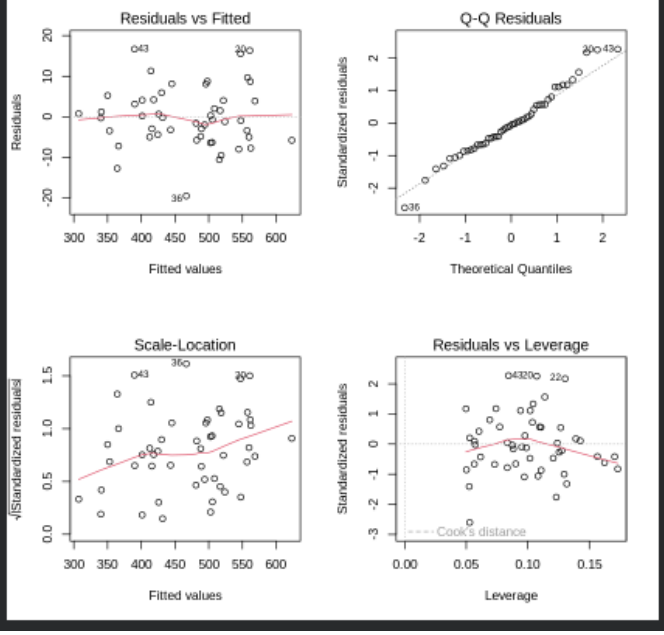
Pembahasan Output : Objek model_latihan yang dihasilkan berisi hasil estimasi koefisien regresi linier ganda, termasuk nilai intersep, koefisien untuk **LuasTanah** dan **JmlKamar**, serta koefisien untuk masing-masing kategori **Lokasi** dibandingkan dengan kategori referensi. Informasi ini dapat digunakan untuk menginterpretasikan pengaruh setiap variabel terhadap harga rumah, misalnya kenaikan harga rata-rata akibat bertambahnya luas tanah atau jumlah kamar, serta perbedaan harga rumah berdasarkan lokasi. Untuk melihat detail hasil estimasi dan evaluasi model, hasil ini dapat ditampilkan lebih lanjut menggunakan fungsi summary(model_latihan).

```
# --- BAGIAN 3: OUTPUT DAN ANALISA ---
summary(model_latihan)

# Plot sederhana untuk cek asumsi
par(mfrow = c(2, 2))
plot(model_latihan)
par(mfrow = c(1, 1))
```

Pembahasan : Kode pada bagian ini digunakan untuk menampilkan **output hasil regresi linier ganda** sekaligus melakukan **analisis diagnostik model**. Fungsi summary(model_latihan) digunakan untuk menampilkan ringkasan hasil regresi yang memuat koefisien, nilai signifikansi, serta ukuran kebaikan model seperti R-squared. Selanjutnya, fungsi par(mfrow = c(2, 2)) digunakan untuk membagi tampilan grafik menjadi empat bagian dalam satu layar, sehingga beberapa grafik diagnostik dapat ditampilkan secara bersamaan. Perintah plot(model_latihan) kemudian menghasilkan empat grafik diagnostik standar regresi linier, dan par(mfrow = c(1, 1)) digunakan untuk mengembalikan tampilan grafik ke pengaturan normal.

Output

```
Residual standard error: 7.676 on 45 degrees of freedom
Multiple R-squared:  0.9904, Adjusted R-squared:  0.9895
F-statistic:  1159 on 4 and 45 DF,  p-value: < 2.2e-16
```



Pembahasan Output : Output dari summary(model_latihan) menunjukkan nilai koefisien regresi untuk setiap variabel, termasuk **LuasTanah**, **JmlKamar**, dan kategori **Lokasi**, yang mengindikasikan besarnya pengaruh masing-masing variabel terhadap harga rumah. Nilai **R-squared** dan **Adjusted R-squared** menggambarkan seberapa baik model mampu menjelaskan variasi harga rumah berdasarkan variabel-variabel tersebut. Sementara itu, empat grafik diagnostik yang dihasilkan dari plot(model_latihan) memberikan gambaran visual mengenai pemenuhan asumsi regresi, seperti linearitas, normalitas residu, kesamaan variansi, dan keberadaan data berpengaruh. Grafik-grafik ini membantu memastikan bahwa model regresi linier yang dibangun layak digunakan untuk analisis dan prediksi.

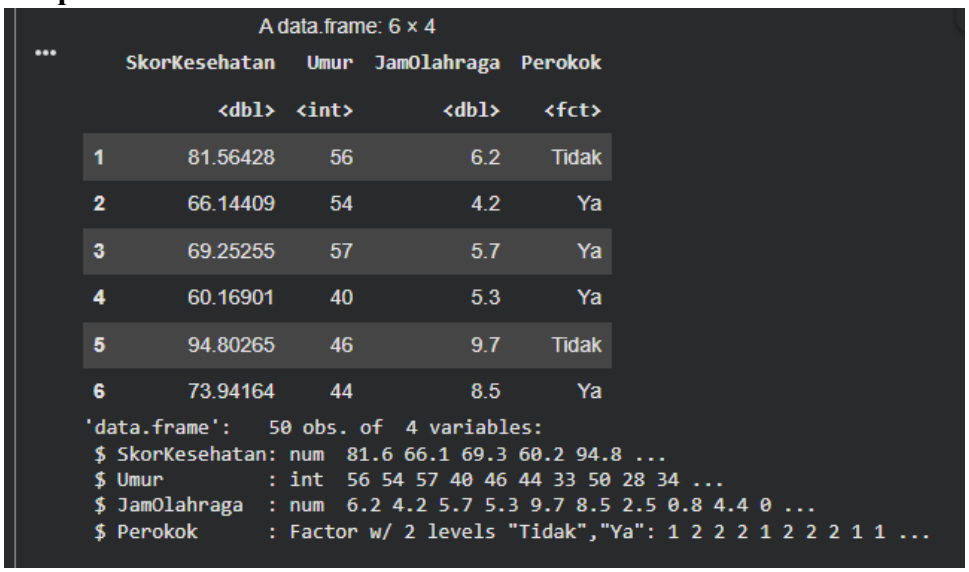
C. PEMBAHASAN TUGAS

```
# --- BAGIAN 1: PERSIAPAN DATA (Simulasi Open Data) ---
set.seed(456)
n_tugas <- 50

# Membuat variabel
umur <- sample(20:60, n_tugas, replace = TRUE)
olahraga <- round(runif(n_tugas, 0, 10), 1) # Jam per minggu
status_perokok <- sample(c("Ya", "Tidak"), n_tugas, replace = TRUE) # Dummy
Variable

# Rumus Kesehatan (Simulasi)
# Skor 80 - (Umur * 0.2) + (Olahraga * 2) - (Jika Perokok * 15)
skor_base <- 80 - (umur * 0.2) + (olahraga * 2)
efek_rokok <- ifelse(status_perokok == "Ya", -15, 0)
kesehatan <- skor_base + efek_rokok + rnorm(n_tugas, 0, 5)

# Membuat Data Frame
df_tugas <- data.frame(
```

```
  SkorKesehatan = kesehatan,
  Umur = umur,
  JamOlahraga = olahraga,
  Perokok = as.factor(status_perokok) # Penting: Ubah kategori jadi factor
)

# Cek data
head(df_tugas)
str(df_tugas)
```

Pembahasan : Kode pada bagian ini digunakan untuk **menyiapkan data simulasi** yang menyerupai data kesehatan dari sumber *open data*. Proses diawali dengan penggunaan set.seed(456) agar data acak yang dihasilkan bersifat konsisten ketika dijalankan ulang. Selanjutnya ditentukan jumlah data sebanyak 50 observasi, kemudian dibuat beberapa variabel, yaitu **Umur** sebagai usia responden, **JamOlahraga** sebagai jumlah jam olahraga per minggu, serta **Perokok** sebagai variabel kategorik yang menunjukkan status merokok (Ya atau Tidak). Variabel **SkorKesehatan** dibentuk menggunakan rumus simulasi yang mempertimbangkan pengaruh umur, aktivitas olahraga, dan status merokok, serta ditambahkan *noise* acak agar data lebih realistis. Seluruh variabel kemudian digabungkan ke dalam data frame df_tugas, dengan variabel Perokok diubah menjadi tipe **factor** agar dapat digunakan sebagai *dummy variable* dalam analisis regresi. Fungsi head() dan str() digunakan untuk menampilkan contoh data dan memastikan struktur serta tipe data sudah sesuai.

Output



A data.frame: 6 × 4

	SkorKesehatan	Umur	JamOlahraga	Perokok
	<dbl>	<int>	<dbl>	<fct>
1	81.56428	56	6.2	Tidak
2	66.14409	54	4.2	Ya
3	69.25255	57	5.7	Ya
4	60.16901	40	5.3	Ya
5	94.80265	46	9.7	Tidak
6	73.94164	44	8.5	Ya

```
'data.frame':   50 obs. of  4 variables:
 $ SkorKesehatan: num  81.6 66.1 69.3 60.2 94.8 ...
 $ Umur         : int  56 54 57 40 46 44 33 50 28 34 ...
 $ JamOlahraga  : num  6.2 4.2 5.7 5.3 9.7 8.5 2.5 0.8 4.4 0 ...
 $ Perokok      : Factor w/ 2 levels "Tidak","Ya": 1 2 2 2 1 2 2 2 1 1 ...
```

Pembahasan Output : Output dari head(df_tugas) menampilkan enam baris pertama dari dataset yang berisi nilai skor kesehatan beserta umur, jam olahraga, dan status perokok. Sementara itu, output str(df_tugas) menunjukkan bahwa dataset terdiri dari **50 observasi dan 4 variabel**, di mana **SkorKesehatan**, **Umur**, dan **JamOlahraga** bertipe numerik, sedangkan **Perokok** bertipe faktor dengan dua level, yaitu *Ya* dan *Tidak*. Hal ini menandakan bahwa dataset telah tersusun dengan baik dan siap digunakan untuk analisis lanjutan, seperti pemodelan regresi linier untuk mengetahui faktor-faktor yang memengaruhi skor kesehatan.

```
# --- BAGIAN 2: ANALISA REGRESI LINIER GANDA ---
# Model: SkorKesehatan dipengaruhi Umur, JamOlahraga, dan Perokok
model_tugas <- lm(SkorKesehatan ~ Umur + JamOlahraga + Perokok, data = df_tugas)
```

Pembahasan : Kode pada bagian ini digunakan untuk membangun **model regresi linier ganda** dengan **SkorKesehatan** sebagai variabel dependen dan **Umur**, **JamOlahraga**, serta **Perokok** sebagai variabel independen. Fungsi lm() digunakan untuk memodelkan hubungan linier antara skor kesehatan dengan faktor usia, aktivitas olahraga, dan status merokok berdasarkan data pada df_tugas. Variabel *Perokok* yang bertipe faktor secara otomatis akan diubah oleh R menjadi variabel dummy, sehingga pengaruh perokok dan bukan perokok terhadap skor kesehatan dapat dianalisis. Model ini bertujuan untuk mengetahui seberapa besar kontribusi masing-masing faktor terhadap perubahan skor kesehatan.

Output

```
Call:
lm(formula = SkorKesehatan ~ Umur + JamOlahraga + Perokok, data = df_tugas)

Residuals:
    Min      1Q  Median      3Q     Max
-9.2552 -3.0961  0.4017  3.6766  6.6481
```

Pembahasan Output : Output dari summary(model_tugas) menampilkan koefisien regresi untuk setiap variabel, yaitu **Umur**, **JamOlahraga**, dan **Perokok**, beserta nilai *intercept*. Koefisien **Umur** menunjukkan bahwa semakin bertambah usia, skor kesehatan cenderung menurun, sedangkan koefisien **JamOlahraga** menunjukkan bahwa semakin banyak jam olahraga per minggu, skor kesehatan cenderung meningkat. Koefisien variabel **Perokok** menunjukkan perbedaan skor kesehatan antara individu perokok dan bukan perokok, di mana perokok memiliki skor kesehatan yang lebih rendah dibandingkan kategori referensinya. Selain itu, nilai **R-squared** dan **Adjusted R-squared** menunjukkan seberapa besar variasi skor kesehatan dapat dijelaskan oleh ketiga variabel tersebut, sementara nilai **p-value** pada masing-masing variabel digunakan untuk menilai signifikansi pengaruhnya terhadap skor kesehatan. Jika p-value lebih kecil dari 0,05, maka variabel tersebut berpengaruh signifikan terhadap skor kesehatan.

```
# --- BAGIAN 3: OUTPUT DAN ANALISA ---
summary(model_tugas)

# Prediksi contoh: Umur 30, Olahraga 5 jam, Bukan Perokok
data_baru <- data.frame(Umur = 30, JamOlahraga = 5, Perokok = "Tidak")
prediksi_sehat <- predict(model_tugas, newdata = data_baru)
cat("Prediksi Skor Kesehatan (Umur 30, Olahraga 5 jam, Tidak Merokok):",
prediksi_sehat, "\n")
```

Pembahasan : Kode pada bagian ini digunakan untuk menampilkan **output hasil regresi linier ganda** dan melakukan **prediksi skor kesehatan** berdasarkan data baru. Fungsi summary(model_tugas) digunakan untuk menampilkan ringkasan hasil model regresi yang memuat informasi koefisien, signifikansi variabel, serta ukuran kebaikan model. Selanjutnya dibuat data baru dengan karakteristik umur 30 tahun, jam olahraga 5 jam per minggu, dan status **Tidak Merokok**. Fungsi predict() kemudian digunakan untuk menghitung nilai skor kesehatan yang diperkirakan oleh model berdasarkan data tersebut, dan hasilnya ditampilkan menggunakan fungsi cat().

Output

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  78.78949    2.74281  28.726  < 2e-16 ***
Umur         -0.15617    0.05648  -2.765  0.00816 **
JamOlahraga   1.94443    0.24100   8.068 2.35e-10 ***
PerokokYa   -15.15659    1.27404 -11.896 1.23e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.355 on 46 degrees of freedom
Multiple R-squared:  0.838, Adjusted R-squared:  0.8275
F-statistic: 79.33 on 3 and 46 DF,  p-value: < 2.2e-16
Prediksi Skor Kesehatan (Umur 30, Olahraga 5 jam, Tidak Merokok): 83.82667
```

Pembahasan Output : Output dari summary(model_tugas) menunjukkan bahwa variabel **Umur**, **JamOlahraga**, dan **Perokok** memiliki pengaruh yang berbeda terhadap skor kesehatan. Umur cenderung berpengaruh negatif terhadap skor kesehatan, sedangkan jam olahraga berpengaruh positif, dan status perokok menunjukkan bahwa individu yang tidak merokok memiliki skor kesehatan yang lebih baik dibandingkan perokok. Selain itu, nilai **R-squared** menunjukkan seberapa besar variasi skor kesehatan dapat dijelaskan oleh model. Output prediksi menampilkan nilai **Skor Kesehatan** yang diperkirakan untuk individu berusia 30 tahun, berolahraga 5 jam per minggu, dan tidak merokok, yang dapat digunakan sebagai gambaran kondisi kesehatan berdasarkan model yang telah dibangun.

D. KESIMPULAN

Berdasarkan hasil praktikum regresi dengan dummy variabel, dapat disimpulkan bahwa model regresi linier ganda mampu digunakan untuk menganalisis pengaruh variabel numerik dan kategorik secara bersamaan. Penggunaan dummy variabel memungkinkan variabel kategorik seperti jenis kelamin, tipe toko, lokasi, atau status perokok dimasukkan ke dalam model regresi sehingga perbedaan antar kategori dapat dianalisis secara kuantitatif. Hasil pengujian menunjukkan bahwa variabel numerik seperti pengalaman, pengeluaran iklan, luas tanah, atau jam olahraga memberikan pengaruh yang jelas terhadap variabel dependen, sementara dummy variabel menunjukkan adanya perbedaan nilai rata-rata variabel dependen antar kategori tertentu. Selain itu, evaluasi asumsi regresi melalui analisis residu dan nilai RMSE menunjukkan bahwa model yang dibangun cukup layak digunakan untuk prediksi. Dengan demikian, praktikum ini membuktikan bahwa regresi dengan dummy variabel efektif untuk memahami hubungan data yang melibatkan variabel kategorik dan numerik dalam satu model analisis.