Pert 8 - Data Modelling dan Shiny App

YOHANES FEBRYAN KANA NYOLA_123220198

2024-11-06

Data Modelling dan Shiny App

Data Modelling

Import Library

```
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ ggplot2   3.5.1     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.1
## ✓ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to be
come errors
```

```
library(dslabs)
library(tidymodels)
```

```
## ── Attaching packages ─────────────────────────────────────── tidymodels 1.2.0 ──
## ✓ broom        1.0.6     ✓ rsample      1.2.1
## ✓ dials        1.3.0     ✓ tune         1.2.1
## ✓ infer        1.0.7     ✓ workflows    1.1.4
## ✓ modeldata    1.4.0     ✓ workflowsets 1.1.0
## ✓ parsnip      1.2.1     ✓ yardstick    1.3.1
## ✓ recipes      1.1.0
## ── Conflicts ────────────────────────────────────── tidymodels_conflicts() ──
## ✖ scales::discard() masks purrr::discard()
## ✖ dplyr::filter()   masks stats::filter()
## ✖ recipes::fixed()  masks stringr::fixed()
## ✖ dplyr::lag()      masks stats::lag()
## ✖ yardstick::spec() masks readr::spec()
## ✖ recipes::step()   masks stats::step()
## • Use tidymodels_prefer() to resolve common conflicts.
```

```
library(vroom)
```

```
##
## Attaching package: 'vroom'
##
## The following object is masked from 'package:yardstick':
##
##     spec
##
## The following object is masked from 'package:scales':
##
##     col_factor
##
## The following objects are masked from 'package:readr':
##
##     as.col_spec, col_character, col_date, col_datetime, col_double,
##     col_factor, col_guess, col_integer, col_logical, col_number,
##     col_skip, col_time, cols, cols_condense, cols_only, date_names,
##     date_names_lang, date_names_langs, default_locale, fwf_cols,
##     fwf_empty, fwf_positions, fwf_widths, locale, output_column,
##     problems, spec
```

```
library(here)
```

```
## here() starts at D:/KULIAH IF/SEMESTER 5/PRAK DATA SCIENCE/Praktikum (Practice)/Pertemuan
8
```

Import Data

```
path = here('data-raw', 'un_smp.csv')
un_smp = vroom(path)
```

```
## Rows: 1409 Columns: 8
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (2): status, nama_sekolah
## dbl (6): tahun, jumlah_peserta, bahasa_indonesia, bahasa_inggris, matematika...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
un_smp = un_smp %>%
  mutate(tahun = as.character(tahun))
str(un_smp)
```

```
## tibble [1,409 × 8] (S3: tbl_df/tbl/data.frame)
##  $ tahun           : chr [1:1409] "2015" "2015" "2015" "2015" ...
##  $ status          : chr [1:1409] "Negeri" "Negeri" "Negeri" "Negeri" ...
##  $ nama_sekolah    : chr [1:1409] "SMP NEGERI 1 BANDUNG" "SMP NEGERI 2 BANDUNG" "SMP NEGER
I 3 BANDUNG" "SMP NEGERI 4 BANDUNG" ...
##  $ jumlah_peserta  : num [1:1409] 441 284 291 385 333 341 352 317 450 353 ...
##  $ bahasa_indonesia: num [1:1409] 86.5 86.3 86.2 84.5 89.2 ...
##  $ bahasa_inggris  : num [1:1409] 82.3 88.7 81.5 77.8 91.3 ...
##  $ matematika      : num [1:1409] 76.5 76.6 75.7 67 83.2 ...
##  $ ipa             : num [1:1409] 76.8 80.3 74.8 70.6 84 ...
```

Supervised Learning

Set seed untuk mengontrol pengecekan data sebelum splitting menjadi data training dan testing

```
set.seed(42)
un_smp_split = un_smp %>%
  initial_split(prop = 0.8)
un_smp_split
```

```
## <Training/Testing/Total>
## <1127/282/1409>
```

```
set.seed(5)
sample(1:10, 6)
```

```
## [1] 2 9 7 3 1 6
```

Buat resep

```
un_smp_recipe = training(un_smp_split) %>%
  recipe() %>%
  update_role(
    tahun,
    status,
    jumlah_peserta,
    bahasa_indonesia,
    bahasa_inggris,
    matematika,
    new_role = "predictor"
  ) %>%
  update_role(
    ipa,
    new_role = "outcome"
  ) %>%
  update_role(
    nama_sekolah,
    new_role = "ID"
  ) %>%
  step_corr(
    all_predictors(),
    -tahun,
    -status
  )

un_smp_recipe
```

```
##
```

```
## ── Recipe ──────────────────────────────────────────────────────────────────
```

```
##
```

```
## ── Inputs
```

```
## Number of variables by role
```

```
## outcome:   1
## predictor: 6
## ID:        1
```

```
##
```

```
## ── Operations
```

```
## • Correlation filter on: all_predictors(), -tahun, -status
```

Terapkan resep

```
un_smp_training = un_smp_recipe %>%
  prep() %>%
  bake(
    training(un_smp_split)
  )

un_smp_testing = un_smp_recipe %>%
  prep() %>%
  bake(
    testing(un_smp_split)
  )
```

Training Model

Training model dengan metode linear regression

```
un_smp_lm = linear_reg(mode = "regression") %>%
  set_engine("lm") %>%
  fit(
    ipa ~ . -nama_sekolah,
    data = un_smp_training
  )

un_smp_lm
```

```
## parsnip model object
##
##
## Call:
## stats::lm(formula = ipa ~ . - nama_sekolah, data = data)
##
## Coefficients:
##      (Intercept)         tahun2016         tahun2017         tahun2018
##        -0.104377          0.054595         -1.767860         -1.162246
##        tahun2019      statusSwasta    jumlah_peserta  bahasa_indonesia
##        -1.763483         -0.858112          0.003518          0.339712
##       matematika
##         0.608978
```

Prediksi dan Evaluasi

```
un_smp_lm %>%
  predict(un_smp_testing) %>%
  bind_cols(un_smp_testing) %>%
  metrics(
    truth = ipa,
    estimate = .pred
  )
```

```
## # A tibble: 3 × 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard       2.74
## 2 rsq     standard       0.954
## 3 mae     standard       2.10
```

Unsupervised Learning

Load Data

```
data(gapminder)
```

Preprocessing Data

Mengganti nilai NA menjadi rata rata dari kolom tersebut

```
gapminder$infant_mortality[is.na(gapminder$infant_mortality)] = mean(gapminder$infant_mortali
ty, na.rm = TRUE)

gapminder$life_expectancy[is.na(gapminder$life_expectancy)] = mean(gapminder$life_expectancy,
na.rm = TRUE)

gapminder$fertility[is.na(gapminder$fertility)] = mean(gapminder$fertility, na.rm = TRUE)

gapminder$gdp[is.na(gapminder$gdp)] = mean(gapminder$gdp, na.rm = TRUE)
```

Ambil data gapminder di tahun 2004

```
gapminder_2004 = gapminder %>%
  filter(year == 2004) %>%
  select(country, infant_mortality, life_expectancy, fertility, population, gdp)

head(gapminder_2004)
```

```
##              country infant_mortality life_expectancy fertility population
## 1            Albania             19.1            75.9      2.00    3103758
## 2            Algeria             30.1            74.4      2.45   32817225
## 3             Angola            122.8            54.5      6.70   17295500
## 4 Antigua and Barbuda             11.0            74.6      2.25      81718
## 5          Argentina             16.0            75.0      2.31   38728778
## 6            Armenia             21.9            71.8      1.40    3025982
##            gdp
## 1   4543619309
## 2  66189522629
## 3  12382535739
## 4    945770280
## 5 287258675094
## 6   2986190081
```

Scalling Data

```
gapminder_2004_scaled = gapminder_2004 %>%
  select(-country) %>% scale()

head(gapminder_2004_scaled)
```

```
##      infant_mortality life_expectancy  fertility   population        gdp
## [1,]       -0.5299630       0.7497195 -0.6482356 -0.237921468 -0.2083518
## [2,]       -0.1666519       0.5896845 -0.3726055 -0.009527032 -0.1404312
## [3,]        2.8950701      -1.5334472  2.2305676 -0.128835749 -0.1997150
## [4,]       -0.7974921       0.6110225 -0.4951078 -0.261150569 -0.2123159
## [5,]       -0.6323507       0.6536985 -0.4583571  0.035912492  0.1031398
## [6,]       -0.4374838       0.3122904 -1.0157424 -0.238519298 -0.2100678
```

Training Data

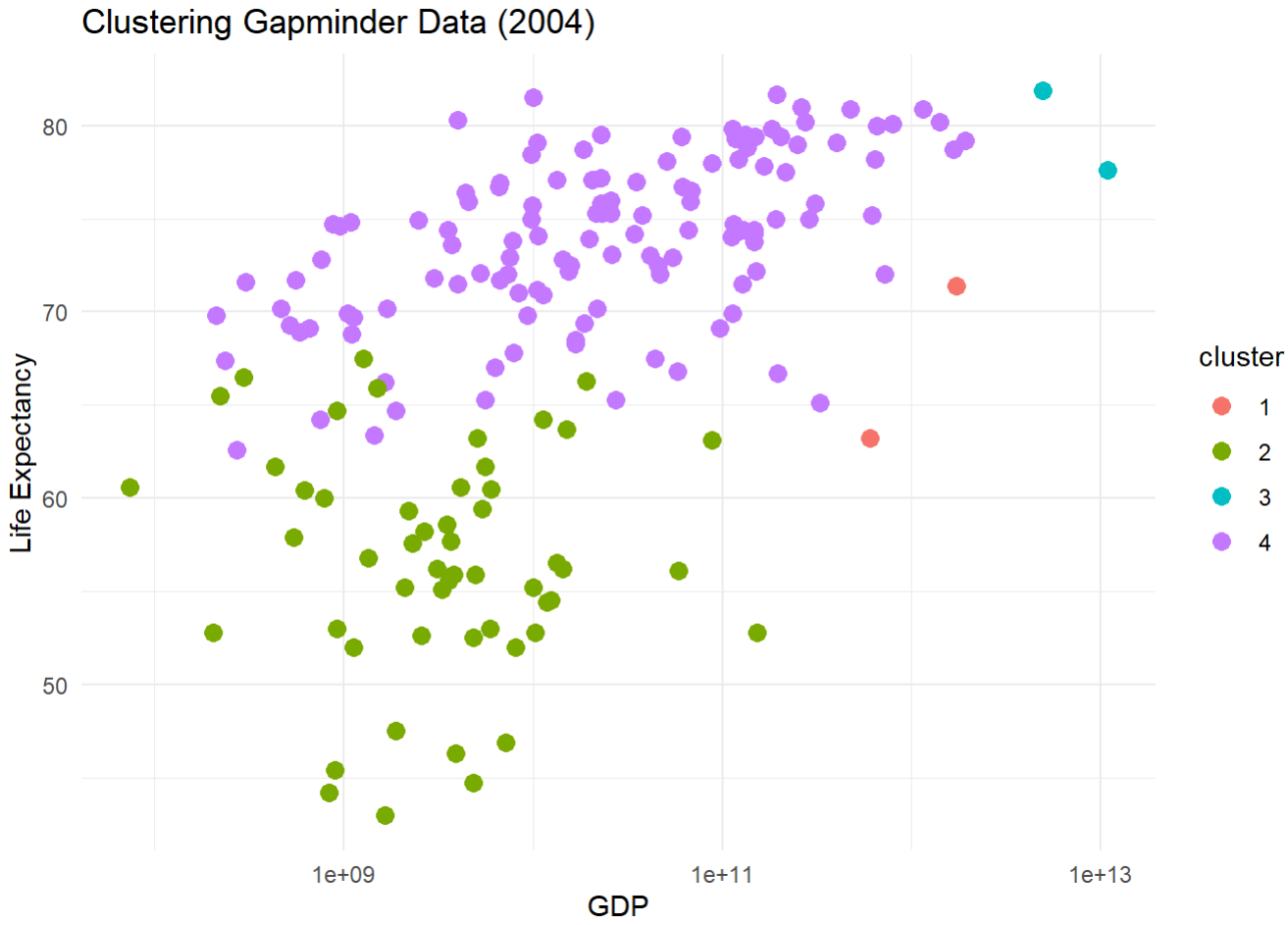
```
set.seed(123)
kmeans_result = kmeans(gapminder_2004_scaled,
                       center = 4,
                       nstart = 10)

gapminder_2004$cluster = as.factor(kmeans_result$cluster)
```

Elbow Method

Visualisasi Data

```
ggplot(
  gapminder_2004,
  aes(
    x = gdp,
    y = life_expectancy,
    color = cluster
  )
) + geom_point(
  size = 3
) + labs(
  title = "Clustering Gapminder Data (2004)",
  x = "GDP",
  y = "Life Expectancy"
) + theme_minimal() + scale_x_log10()
```


Clustering Gapminder Data (2004)
Life Expectancy
80
70
60
50
1e+09
1e+11
1e+13
GDP
cluster
1
2
3
4