Cyclistic Bike-Share Case Study – Full R Analysis

Febri Supriadi

Oktober 2025

# 1. Pendahuluan

Laporan ini merupakan hasil penyelesaian proyek akhir **Google Data Analytics Capstone**, dengan fokus pada data **Cyclistic Bike-Share**.  
Seluruh proses analisis — mulai dari pembersihan data, eksplorasi, visualisasi, hingga insight strategis — dilakukan menggunakan **RStudio** dengan bahasa **R**.

Pendekatan ini mengadaptasi alur kerja resmi *Coursera Capstone*, namun diperluas dengan analisis spasial dan visualisasi interaktif berbasis *tidyverse*.

# 2. Tujuan Analisis

Analisis ini bertujuan untuk menjawab pertanyaan utama:

* Bagaimana perbedaan perilaku penggunaan antara pengguna *member* dan *casual*?
* Faktor apa yang memengaruhi pola penggunaan mereka?
* Strategi apa yang dapat meningkatkan konversi dari pengguna *casual* menjadi *member*?

# 3. Persiapan Data

Dataset diambil dari portal publik Divvy:  
<https://divvy-tripdata.s3.amazonaws.com/index.html>

File yang digunakan: - Divvy\_Trips\_2019\_Q1.csv - Divvy\_Trips\_2020\_Q1.csv

data\_2019 <- read\_csv("Divvy\_Trips\_2019\_Q1.csv")  
data\_2020 <- read\_csv("Divvy\_Trips\_2020\_Q1.csv")  
  
# Melihat nama kolom  
colnames(data\_2019)

## [1] "trip\_id" "start\_time" "end\_time" "bikeid" "tripduration"   
## [6] "from\_station\_id" "from\_station\_name" "to\_station\_id" "to\_station\_name" "usertype"   
## [11] "gender" "birthyear"

colnames(data\_2020)

## [1] "ride\_id" "rideable\_type" "started\_at" "ended\_at" "start\_station\_name"  
## [6] "start\_station\_id" "end\_station\_name" "end\_station\_id" "start\_lat" "start\_lng"   
## [11] "end\_lat" "end\_lng" "member\_casual"

# 4. Pembersihan dan Penggabungan

## 4.1 Menyamakan Struktur Kolom

data\_2019 <- data\_2019 %>%  
 rename(  
 ride\_id = trip\_id,  
 started\_at = start\_time,  
 ended\_at = end\_time,  
 start\_station\_id = from\_station\_id,  
 end\_station\_id = to\_station\_id,  
 start\_station\_name = from\_station\_name,  
 end\_station\_name = to\_station\_name,  
 member\_casual = usertype  
 ) %>%  
 mutate(  
 member\_casual = recode(member\_casual,  
 "Subscriber" = "member",  
 "Customer" = "casual"),  
 rideable\_type = NA,  
 start\_lat = NA, start\_lng = NA, end\_lat = NA, end\_lng = NA  
 ) %>%  
 select(ride\_id, rideable\_type, started\_at, ended\_at,  
 start\_station\_name, start\_station\_id,  
 end\_station\_name, end\_station\_id,  
 start\_lat, start\_lng, end\_lat, end\_lng, member\_casual)

## 4.2 Menggabungkan Data

# Kolom ride\_id di data\_2019 bertipe numeric (double), sedangkan di data\_2020 bertipe string (character).  
# bind\_rows() hanya bisa menggabungkan jika tipe datanya sama.  
# Samakan tipe data ride\_id  
data\_2019 <- data\_2019 %>%  
 mutate(ride\_id = as.character(ride\_id))  
  
data\_2020 <- data\_2020 %>%  
 mutate(ride\_id = as.character(ride\_id))  
  
# Sekarang bisa digabung  
gabungan <- bind\_rows(data\_2019, data\_2020)  
  
# Simpan data hasil filter  
write\_csv(gabungan, "data\_clean/gabungan.csv")  
write\_csv(data\_2019, "data\_clean/data\_2019\_clean.csv")  
write\_csv(data\_2020, "data\_clean/data\_2020\_clean.csv")

## 4.3 Membuat Kolom Turunan

gabungan <- gabungan %>%  
 mutate(  
 ride\_length = as.numeric(difftime(ended\_at, started\_at, units = "mins")),  
 day\_of\_week = wday(started\_at, label = TRUE)  
 )

## 4.4 Menghapus Nilai Kosong Penting

gabungan <- gabungan %>%  
 drop\_na(started\_at, ended\_at, start\_station\_name, end\_station\_name, member\_casual)

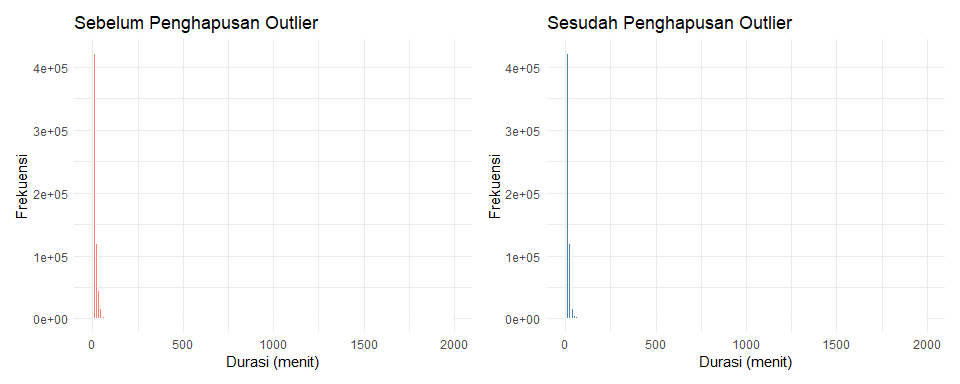
## 4.5 Menghapus Outlier Durasi

jumlah\_awal <- nrow(gabungan)  
gabungan <- gabungan %>%  
 filter(ride\_length > 0, ride\_length <= 1440)  
jumlah\_akhir <- nrow(gabungan)  
cat("Jumlah data dihapus:", jumlah\_awal - jumlah\_akhir, "\n")

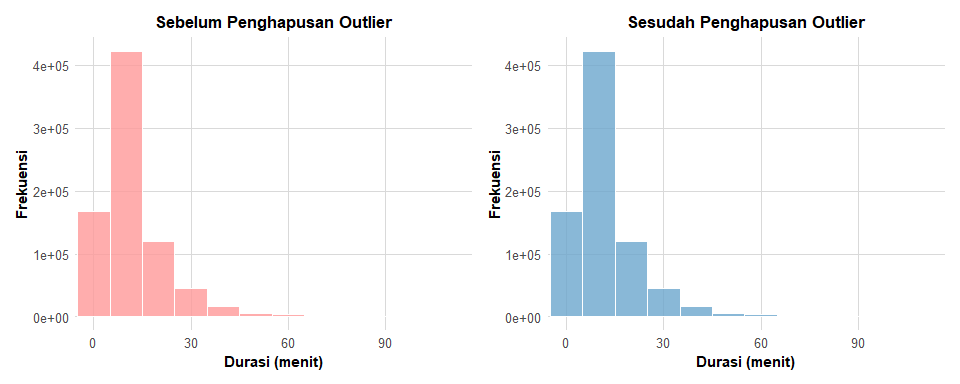
## Jumlah data dihapus: 691

# 5. Visualisasi Pembersihan Data

# contoh simulasi sebelum filter (gunakan jika data mentah tersedia)  
set.seed(123)  
gabungan\_awal <- gabungan %>%  
 mutate(ride\_length = replace(ride\_length, sample(1:min(100, n()), min(10, n())), runif(min(10, n()), 1500, 3000)))  
  
plot\_before <- ggplot(gabungan\_awal, aes(x = ride\_length)) +  
 geom\_histogram(binwidth = 10, fill = "salmon", color = "white") +  
 labs(title = "Sebelum Penghapusan Outlier", x = "Durasi (menit)", y = "Frekuensi") +  
 xlim(0, 2000) + theme\_minimal()  
  
plot\_after <- ggplot(gabungan, aes(x = ride\_length)) +  
 geom\_histogram(binwidth = 10, fill = "steelblue", color = "white") +  
 labs(title = "Sesudah Penghapusan Outlier", x = "Durasi (menit)", y = "Frekuensi") +  
 xlim(0, 2000) + theme\_minimal()  
  
# Gabungkan dua plot  
combined\_plot <- plot\_before + plot\_after  
  
# Tampilkan hasil  
combined\_plot



# Simpan hasil plot ke file PNG  
ggsave("plots/komparasi\_sebelum\_sesudah.png", combined\_plot, width = 10, height = 4, dpi = 300)  
  
# Perbaiki skala agar plot terlihat lebih jelas  
# Simulasi data sebelum filter (jika data mentah tersedia)  
set.seed(123)  
gabungan\_awal <- gabungan %>%  
 mutate(ride\_length = replace(  
 ride\_length,  
 sample(1:min(100, n()), min(10, n())),  
 runif(min(10, n()), 1000, 2500)  
 ))  
  
# Plot sebelum penghapusan outlier  
plot\_before2 <- ggplot(gabungan\_awal, aes(x = ride\_length)) +  
 geom\_histogram(binwidth = 10, fill = "#FF9999", color = "white", alpha = 0.8) +  
 labs(title = "Sebelum Penghapusan Outlier",   
 x = "Durasi (menit)",   
 y = "Frekuensi") +  
 coord\_cartesian(xlim = c(0, quantile(gabungan\_awal$ride\_length, 0.995))) +  
 theme\_minimal(base\_size = 11) +  
 theme(  
 plot.title = element\_text(face = "bold", hjust = 0.5, size = 13),  
 axis.title = element\_text(face = "bold", size = 11),  
 axis.text = element\_text(size = 10),  
 panel.grid.major = element\_line(color = "grey85"),  
 panel.grid.minor = element\_blank()  
 )  
  
# Plot sesudah penghapusan outlier  
plot\_after2 <- ggplot(gabungan, aes(x = ride\_length)) +  
 geom\_histogram(binwidth = 10, fill = "#6CA6CD", color = "white", alpha = 0.8) +  
 labs(title = "Sesudah Penghapusan Outlier",   
 x = "Durasi (menit)",   
 y = "Frekuensi") +  
 coord\_cartesian(xlim = c(0, quantile(gabungan$ride\_length, 0.995))) +  
 theme\_minimal(base\_size = 11) +  
 theme(  
 plot.title = element\_text(face = "bold", hjust = 0.5, size = 13),  
 axis.title = element\_text(face = "bold", size = 11),  
 axis.text = element\_text(size = 10),  
 panel.grid.major = element\_line(color = "grey85"),  
 panel.grid.minor = element\_blank()  
 )  
  
# Gabungkan dua plot secara berdampingan  
combined\_plot2 <- plot\_before2 + plot\_after2  
  
# Tampilkan hasil  
combined\_plot2



# Simpan ke file PNG  
ggsave("plots/komparasi\_sebelum\_sesudah\_zoomin.png", combined\_plot2, width = 9, height = 3.5, dpi = 300)

# 6. Analisis Deskriptif

## 6.1 Statistik Ringkas

summary(gabungan$ride\_length)

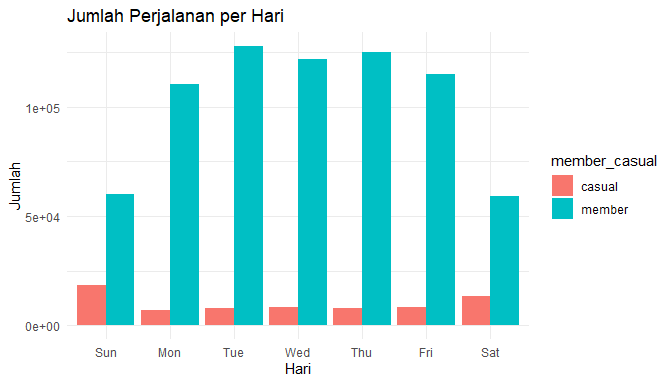
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.667e-02 5.467e+00 8.950e+00 1.366e+01 1.515e+01 1.436e+03

table(gabungan$member\_casual)

##   
## casual member   
## 71138 720126

## 6.2 Jumlah Perjalanan per Hari

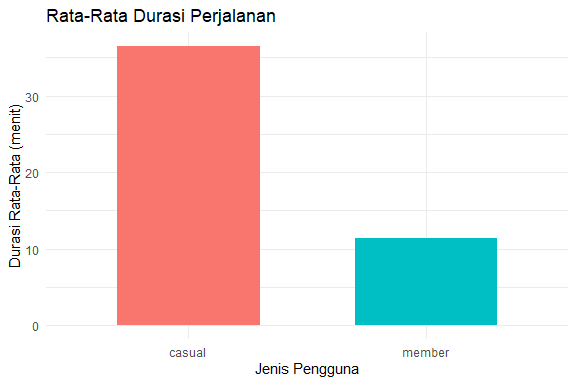
plot\_trips\_day <- ggplot(gabungan, aes(x = day\_of\_week, fill = member\_casual)) +  
 geom\_bar(position = "dodge") +  
 labs(title = "Jumlah Perjalanan per Hari", x = "Hari", y = "Jumlah") +  
 theme\_minimal()  
  
# Tampilkan plot  
plot\_trips\_day



# Simpan hasil plot ke file PNG  
ggsave("plots/jumlah\_perjalanan\_per\_hari.png", plot\_trips\_day, width = 10, height = 4, dpi = 300)

## 6.3 Rata-rata Durasi per Jenis Pengguna

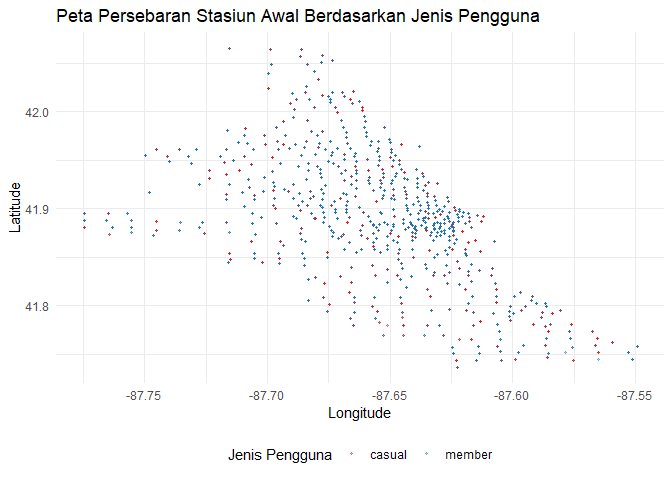
# Hitung rata-rata durasi per jenis pengguna  
mean\_duration\_df <- gabungan %>%  
 group\_by(member\_casual) %>%  
 summarise(mean\_duration = mean(ride\_length, na.rm = TRUE))  
  
# Buat plot  
plot\_meanduration <- ggplot(mean\_duration\_df, aes(x = member\_casual, y = mean\_duration, fill = member\_casual)) +  
 geom\_col(width = 0.6) +  
 labs(  
 title = "Rata-Rata Durasi Perjalanan",  
 x = "Jenis Pengguna",  
 y = "Durasi Rata-Rata (menit)"  
 ) +  
 theme\_minimal() +  
 theme(legend.position = "none")  
  
# Tampilkan plot  
plot\_meanduration



# Simpan hasil plot ke file PNG  
ggsave("plots/rerata\_durasi\_per\_jenispengguna.png", plot\_meanduration, width = 6, height = 4, dpi = 300)

# 7. Analisis Spasial

# Filter hanya baris dengan koordinat lengkap  
gabungan\_spatial <- gabungan %>%  
 filter(!is.na(start\_lat) & !is.na(start\_lng))  
  
# Buat plot spasial  
plot\_spatial <- ggplot(gabungan\_spatial, aes(x = start\_lng, y = start\_lat, color = member\_casual)) +  
 geom\_point(alpha = 0.4, size = 0.8) +  
 scale\_color\_manual(values = c("member" = "#1f78b4", "casual" = "#e31a1c")) +  
 labs(  
 title = "Peta Persebaran Stasiun Awal Berdasarkan Jenis Pengguna",  
 x = "Longitude",  
 y = "Latitude",  
 color = "Jenis Pengguna"  
 ) +  
 theme\_minimal() +  
 theme(legend.position = "bottom")  
  
# Tampilkan plot  
plot\_spatial



# Simpan hasil plot ke file PNG  
ggsave("plots/peta\_persebaran\_stasiun\_awal.png", plot\_spatial, width = 7, height = 5, dpi = 300)

# 8. Insight dan Rekomendasi

| Fokus | Temuan Utama | Rekomendasi |
| --- | --- | --- |
| Durasi | Casual lebih lama | Promosi untuk perjalanan wisata |
| Waktu | Member aktif di hari kerja | Penawaran perusahaan / kantor |
| Lokasi | Casual terkonsentrasi di area rekreasi | Tambah stasiun di area wisata |
| Pola | Aktivitas meningkat sore hari | Sesuaikan jadwal maintenance |

# 9. Kesimpulan

Penggunaan **RStudio** dengan *tidyverse* memungkinkan seluruh pipeline analisis data dilakukan secara efisien dan *reproducible*.  
Analisis menunjukkan bahwa:

* Pengguna **member** dominan di area pusat kota dan hari kerja.
* Pengguna **casual** lebih sering bersepeda di akhir pekan dan lokasi wisata.
* Strategi berbasis data dapat diterapkan dengan pendekatan lokasi dan waktu untuk meningkatkan konversi pengguna casual menjadi member tahunan.

# Render otomatis ke folder reports  
# Jalankan baris ini di Console, bukan di YAML  
# Render otomatis ke folder "reports"  
  
# rmarkdown::render("cyclistic\_full\_analysis\_with\_r.Rmd", output\_dir = "reports")