1. GRAFİK- AYLARA GÖRE ORTALAMA TRAFİK YOĞUNLUĞU- ÇİZGİ GRAFİĞİ

library(ggplot2)

library(dplyr)

library(lubridate)

# 1. Veriyi yükle

df <- read.csv("C:/Users/Acer/Downloads/Sehir\_verileri.csv", fileEncoding = "UTF-8", stringsAsFactors = FALSE)

# 2. Tarih sütununu Date formatına çevir (gün-ay-yıl formatında)

df$analiz\_baslangic\_tarih <- as.Date(df$analiz\_baslangic\_tarih, format = "%d-%m-%Y")

# 3. Aylık ortalama yoğunluğu hesapla

df\_aylik <- df %>%

filter(!is.na(analiz\_baslangic\_tarih)) %>%

mutate(ay = month(analiz\_baslangic\_tarih, label = TRUE, abbr = FALSE)) %>%

group\_by(ay) %>%

summarise(yogunluk = mean(sonuc, na.rm = TRUE))

# 4. Aylık ortalama yoğunluğu gösteren çizgi grafiğini oluştur

ggplot(df\_aylik, aes(x = ay, y = yogunluk, group = 1)) +

geom\_line(color = "#6E7B8B", size = 1) + # Gri çizgi

geom\_point(color = "#87CEFA", size = 3) + # Açık mavi noktalar

labs(title = "Aylara Göre Ortalama Trafik Yoğunluğu",

x = "Ay", y = "Yoğunluk") +

theme\_minimal()

ggsave("harita\_600dpi.png", plot = p, width = 10, height = 8, dpi = 600)

2.GRAFİK- YEDİ BÖLGEYE GÖRE ORTALAMA SONUÇ DAĞILIMI- SÜTUN GRAFİĞİ

library(dplyr)

library(ggplot2)

library(stringi)

# 1. Veri setini UTF-8 olarak oku

Sehir\_verileri <- read.csv(

"C:/Users/Acer/Downloads/Sehir\_verileri.csv",

stringsAsFactors = FALSE,

fileEncoding = "UTF-8"

)

# 2. sehir\_adi sütununu ASCII’ye çevir (çÇğĞıİöÖşŞüÜ → cCgGiIoOsSuU)

Sehir\_verileri <- Sehir\_verileri %>%

mutate(

sehir\_ascii = stri\_trans\_general(sehir\_adi, "Latin-ASCII")

)

# 3. Bölge atamasını ASCII başlığına göre yap ve Diğer’i filtrele

Sehir\_verileri <- Sehir\_verileri %>%

mutate(

Bolge = case\_when(

sehir\_ascii %in% c("Istanbul","Tekirdag","Edirne","Kirklareli","Balikesir",

"Canakkale","Bursa","Bilecik","Sakarya","Kocaeli","Yalova")

~ "Marmara",

sehir\_ascii %in% c("Antalya","Burdur","Isparta","Mersin","Adana","Hatay",

"Osmaniye","Kahramanmaras")

~ "Akdeniz",

sehir\_ascii %in% c("Malatya","Erzincan","Elazig","Tunceli","Bingol",

"Erzurum","Mus","Bitlis","Sirnak","Kars","Agri",

"Ardahan","Van","Igdir","Hakkari")

~ "Doğu Anadolu",

sehir\_ascii %in% c("Izmir","Aydin","Mugla","Manisa","Denizli",

"Usak","Kutahya","Afyonkarahisar")

~ "Ege",

sehir\_ascii %in% c("Gaziantep","Kilis","Adiyaman","Sanliurfa",

"Diyarbakir","Mardin","Batman","Siirt")

~ "GüneyDoğu Anadolu",

sehir\_ascii %in% c("Eskisehir","Konya","Ankara","Cankiri","Aksaray",

"Kirikale","Kirsehir","Yozgat","Nigde","Nevsehir",

"Kayseri","Karaman","Sivas")

~ "Iç Anadolu",

sehir\_ascii %in% c("Bolu","Duzce","Zonguldak","Karabuk","Bartin",

"Kastamonu","Corum","Sinop","Samsun","Amasya",

"Tokat","Ordu","Giresun","Gumusghe","Trabzon",

"Bayburt","Rize","Artvin")

~ "Karadeniz",

TRUE

~ NA\_character\_

)

) %>%

filter(!is.na(Bolge))

# 4. Bölge faktör seviyelerini istediğimiz sıraya sok

bolge\_duzeyleri <- c(

"Marmara", "Ege", "Akdeniz",

"İç Anadolu", "Karadeniz",

"Doğu Anadolu", "GüneyDoğu Anadolu"

)

Sehir\_verileri <- Sehir\_verileri %>%

mutate(

Bolge = factor(Bolge, levels = bolge\_duzeyleri)

)

# 5. Bölgelere göre ortalama sonucu hesapla

Bolge\_avg <- Sehir\_verileri %>%

group\_by(Bolge) %>%

summarise(Ortalama\_Sonuc = mean(sonuc, na.rm = TRUE)) %>%

ungroup()

# 6. Sütun grafiğini çiz

ggplot(Bolge\_avg, aes(x = Bolge, y = Ortalama\_Sonuc, fill = Bolge)) +

geom\_col() +

scale\_fill\_manual(values = setNames(

rep(c("#A2C2E5", "#D3D3D3"), length.out = length(bolge\_duzeyleri)),

bolge\_duzeyleri

)) +

theme\_minimal(base\_size = 14) +

theme(

axis.text.x = element\_text(angle = 45, hjust = 1),

plot.title = element\_text(hjust = 0.5),

panel.grid = element\_blank()

) +

labs(

x = "Bolge",

y = "Ortalama Sonuc",

title = "Yedi Bolgeye Gore Ortalama Sonuc Dagilimi"

)

ggsave("harita\_600dpi.png", plot = p, width = 10, height = 8, dpi = 600)

3.GRAFİK ŞEHİRLERE GÖRE SONUÇ DAĞILIMI- SÜTUN GRAFİĞİ

library(ggplot2)

ggplot(Sehir\_verileri, aes(x=sehir\_adi, y=sonuc)) +

geom\_bar(stat="identity", fill="#A2C2E5") + # Barlar için pastel mavi renk

theme\_minimal(base\_size = 14) +

theme(

axis.text.x = element\_text(angle = 90, hjust = 1), # X eksenindeki yazıları döndür

plot.title = element\_text(hjust = 0.5), # Başlığı ortala

panel.grid = element\_blank() # Izgaraları kaldır

) +

labs(

x = "Sehirler",

y = "Sonuc Degeri",

title = "Sehirlere Gore Sonuc Dagilimi"

)

ggsave("harita\_600dpi.png", plot = p, width = 10, height = 8, dpi = 600)

4.GRAFİK- MEVSİMLERE GÖRE SONUÇ DAĞILIMI-KUTU GRAFİĞİ

library(dplyr)

library(ggplot2)

library(lubridate)

# 1. Tarihleri Date sınıfına çevir

Sehir\_verileri <- Sehir\_verileri %>%

mutate(

baslangic = as.Date(analiz\_baslangic\_tarih, format = "%d-%m-%Y"),

bitis = as.Date(analiz\_bitis\_tarih, format = "%d-%m-%Y")

)

# 2. Ayı çıkar ve mevsime eşle

Sehir\_verileri <- Sehir\_verileri %>%

mutate(

ay = month(baslangic), # veya bitis

Mevsim = case\_when(

ay %in% 3:5 ~ "İlkbahar",

ay %in% 6:8 ~ "Yaz",

ay %in% 9:11 ~ "Sonbahar",

ay %in% c(12,1,2) ~ "Kış"

)

)

# 3. Kutuphane faktör seviyelerini ayarla (okunur sırayla)

Sehir\_verileri <- Sehir\_verileri %>%

mutate(

Mevsim = factor(Mevsim, levels = c("İlkbahar","Yaz","Sonbahar","Kış"))

)

# 4. Kutu grafiği

ggplot(Sehir\_verileri, aes(x = Mevsim, y = sonuc, fill = Mevsim)) +

geom\_boxplot() +

scale\_fill\_manual(values = c(

"İlkbahar" = "#A2C2E5",

"Yaz" = "#D3D3D3",

"Sonbahar" = "#A2C2E5",

"Kış" = "#D3D3D3"

)) +

theme\_minimal(base\_size = 14) +

theme(

axis.text.x = element\_text(angle = 0, hjust = 0.5),

plot.title = element\_text(hjust = 0.5),

panel.grid = element\_blank()

) +

labs(

x = "Mevsim",

y = "Sonuc",

title = "Mevsimlere Gore Sonuc Dagilimi (Kutu Grafigi)"

)

ggsave("harita\_600dpi.png", plot = p, width = 10, height = 8, dpi = 600)

5.GRAFİK-AY VE BÖLGEYE GÖRE ORTALAMA SONUÇ-ISI HARİTASI

library(tidyverse)

library(lubridate)

# Türkçe ay adları için yerel ayar

Sys.setlocale("LC\_TIME", "Turkish")

# Veriyi oku

veri <- read\_csv("C:/Users/Acer/Downloads/Sehir\_verileri.csv")

# Bölge sütununu ekle ve "Diğer" olanları çıkar

veri <- veri %>%

mutate(Bolge = case\_when(

sehir\_adi %in% c("Malatya", "Erzincan", "Elazıg", "Tunceli", "Bingol", "Erzurum") ~ "Doğu Anadolu",

sehir\_adi %in% c("İstanbul", "Kocaeli", "Edirne", "Bursa", "Tekirdag") ~ "Marmara",

sehir\_adi %in% c("Ankara", "Konya", "Eskisehir", "Karaman", "Kayseri") ~ "İç Anadolu",

sehir\_adi %in% c("Antalya", "Mersin", "Adana", "Hatay") ~ "Akdeniz",

sehir\_adi %in% c("İzmir", "Aydın", "Mugla", "Manisa") ~ "Ege",

sehir\_adi %in% c("Trabzon", "Ordu", "Samsun", "Rize", "Giresun") ~ "Karadeniz",

sehir\_adi %in% c("Diyarbakır", "Sanlıurfa", "Mardin", "Batman", "Gaziantep") ~ "Güneydoğu Anadolu",

TRUE ~ NA\_character\_

)) %>%

filter(!is.na(Bolge))

# Ay bilgisini çıkar (Türkçe olarak)

veri <- veri %>%

mutate(ay = month(dmy(analiz\_baslangic\_tarih), label = TRUE, abbr = FALSE))

# Isı haritası için veriyi gruplama

heatmap\_data <- veri %>%

group\_by(Bolge, ay) %>%

summarise(ortalama\_sonuc = mean(sonuc, na.rm = TRUE), .groups = "drop")

# Grafik nesnesini oluştur

p <- ggplot(heatmap\_data, aes(x = ay, y = Bolge, fill = ortalama\_sonuc)) +

geom\_tile(color = "white") +

scale\_fill\_gradient(low = "#dce3f0", high = "#2b6cb0") +

labs(

title = "Ay ve Bölgeye Göre Ortalama Sonuç Isı Haritası",

x = "Ay",

y = "Bölge",

fill = "Ortalama\nSonuç"

) +

scale\_x\_discrete(labels = c("January" = "Ocak", "February" = "subat", "March" = "Mart",

"April" = "Nisan", "May" = "Mayis", "June" = "Haziran",

"July" = "Temmuz", "August" = "Agustos", "September" = "Eylul",

"October" = "Ekim", "November" = "Kasim", "December" = "Aralik")) +

theme\_minimal() +

theme(text = element\_text(size = 12))

# Grafiği göster

print(p)

# Yüksek çözünürlükle grafiği kaydet

ggsave("harita\_600dpi.png", plot = p, width = 10, height = 8, dpi = 600)

6.GRAFİK- TÜRKİYE BÖLGELERİNE GÖRE TRAFİK YOĞUNLUĞU-SAÇILIM GRAFİĞİ

veri <- read.csv("C:\Users\Acer\Downloads\Sehir\_verileri.csv", fileEncoding = "UTF-8")

View(veri) # Veriyi tablo şeklinde görmek için

install.packages("ggplot2")

install.packages("dplyr")

install.packages("sf")

install.packages("rnaturalearth")

install.packages("rnaturalearthdata")

install.packages("rgeos") # bazı sf işlemleri için gerekebilir

# Gerekli paketler

library(ggplot2)

library(dplyr)

library(sf)

library(rnaturalearth)

library(rnaturalearthdata)

# Türkiye haritasını al

turkey <- ne\_countries(scale = "medium", country = "Turkey", returnclass = "sf")

# Örnek veri (senin elindeki gerçek veriyle değiştirebilirsin)

# Bolgeler, enlem ve boylam değerleri ile yoğunluk oranı

veri <- data.frame(

bolge = c("Marmara", "Ege", "Akdeniz", "İç Anadolu", "Karadeniz", "Doğu Anadolu", "Güneydoğu Anadolu"),

lon = c(28.9, 27.1, 30.7, 33.5, 37.0, 39.5, 38.5), # yaklaşık boylamlar

lat = c(40.9, 38.4, 37.0, 39.0, 41.2, 39.5, 37.9), # yaklaşık enlemler

yogunluk = c(75, 60, 50, 40, 35, 20, 45) # örnek yoğunluk oranları

)

# Haritayı çiz

ggplot(data = turkey) +

geom\_sf(fill = "grey85", color = "white") + # gri harita

geom\_point(data = veri, aes(x = lon, y = lat, size = yogunluk),

color = "#80C1FF", alpha = 0.7) + # pastel mavi noktalar

scale\_size\_continuous(range = c(3, 10)) + # nokta büyüklük aralığı

theme\_minimal() +

labs(title = "Turkiye Bolgelerine Gore Trafik Yoğunluğu", size = "Yoğunluk")

+

theme(

plot.title = element\_text(hjust = 0.5, face = "bold", size = 16),

legend.position = "right"

)

ggsave("harita\_600dpi.png", plot = p, width = 10, height = 8, dpi = 600)