유방암 진단 데이터를 활용한 암 분석 시각화

Cancer Analysis Visualization Using Breast Cancer Diagnosis Data

김수인(32230811), 김나연(32230418), 이윤서(32233391)

데이터 소개

id	diagnosis	radius_mea	texture_me	perimeter_	area_mear
842302	M	17.99	10.38	122.8	1001
842517	M	20.57	17.77	132.9	1326
84300903	M	19.69	21.25	130	1203
84348301	M	11.42	20.38	77.58	386.1
84358402	M	20.29	14.34	135.1	1297
843786	M	12.45	15.7	82.57	477.1
844359	M	18.25	19.98	119.6	1040

Figure 1: csv데이터셋

데이터 소개

컬럼 소개

-ID번호: 각 셈플에 부여된 식별번호
-Diagnosis: 샘들의 진단 결과-> “M”은 악성, “N”은 양성
-radius: 반지름
:세포 핵 중심에서 경계까지의 평균거리값
-texture: 세포의 표면 질감, 표면의 균일성
-perimeter: 세포 핵의 둘레길이(셀의 모양이나 크기와 관련)
-area: 세포 핵의 면적(세포 크기와 관련)

데이터셋

-데이터셋 구성: 357개의 양성(B)+212개의 악성(M)=총 569개의 샘플로 구성
-누락된 값 없이 특성값이 모두 존재
-출처: Kaggle-Breast Cancer Wisconsin (Diagnostic) Data Set

순서

1. 산점도
2. Q-Q플랏
3. 히스토그램 및 확률밀도함수 시각화
4. 상관계수 및 피어슨, 스피어만 상관계수 분석
5. 상자그림
6. 상자그림(ggplot,산점도)
7. 카이제곱 분포 시각화
8. 켄달의 타우 값 계산과 시각화
9. 선형회귀 분석과 시각화

산점도

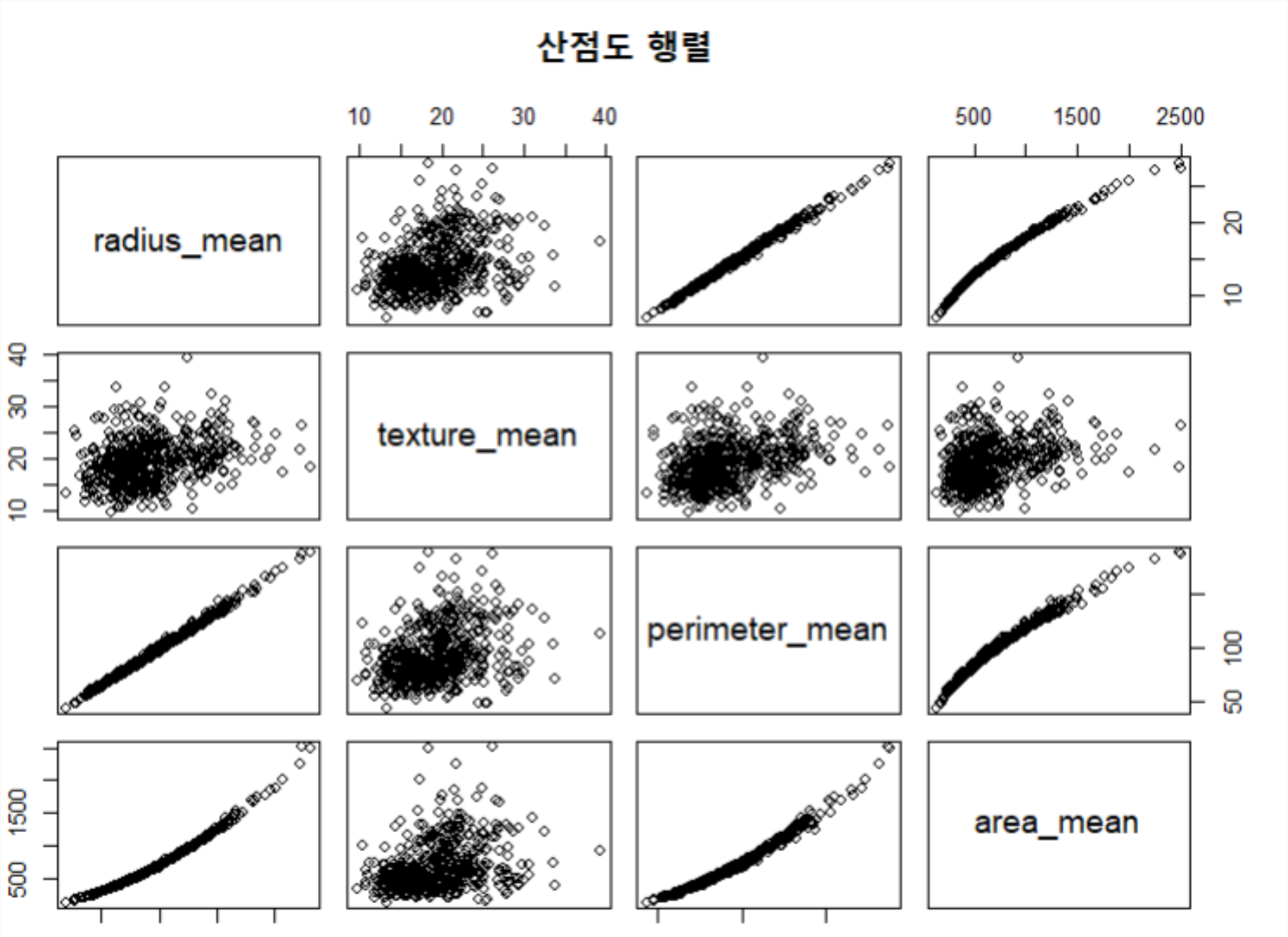
데이터 불러오기

data <- read.csv("C:\Users\User\OneDrive\바탕 화면\data.csv")

산점도 행렬 (몇 가지 주요 변수만 선택)

pairs(data[, c("radius_mean", "texture_mean", "perimeter_mean", "area_mean")],
main = "산점도 행렬")

결과


산점도 행렬
radius_mean
texture_mean
perimeter_mean
area_mean
10
20
30
40
500
1500
2500
10
20
30
40
50
100
500
1500

Q-Q플랏

qqnorm(data$radius_mean$, $main =$
"$Q - QPlotofradius_mean$")$qqline(data$radius_mean, col = "red")

한 화면에 확인(2x2 레이아웃)

par(mfrow = c(2, 2))
for (var in c("radius_mean", "texture_mean", "perimeter_mean",
"area_mean")) {
qqnorm(data[[var]], main = paste("Q-Q Plot of", var))
qqline(data[[var]], col = "red")
}

결과

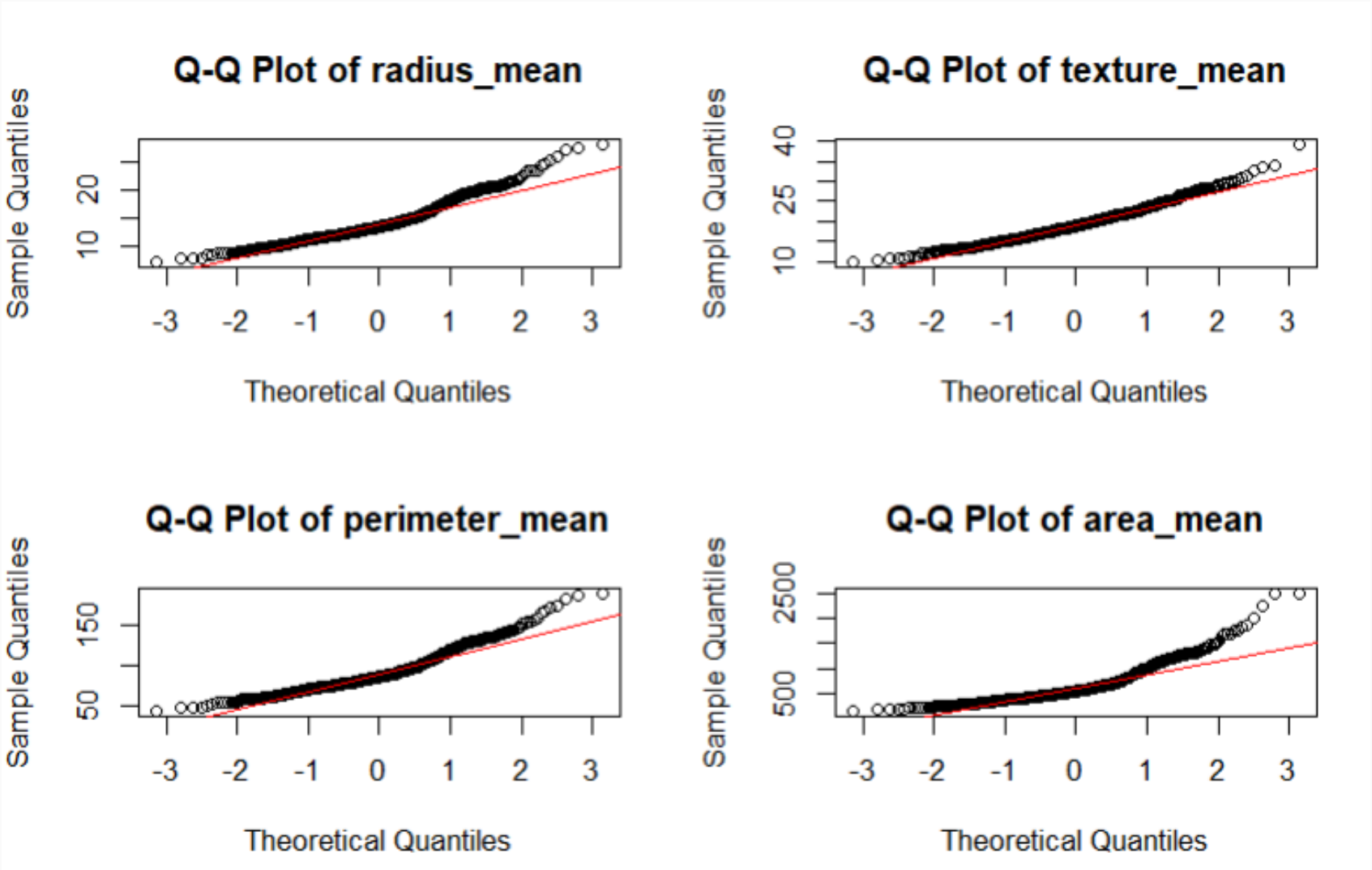


Figure 3: Q-Q플랏

히스토그램 및 확률밀도함수 시각화

필요한 패키지 로드

library(ggplot2)
library(gridExtra)

주요 변수 선택

selected_data <- data[, c("diagnosis", "radius_mean", "texture_mean", "perimeter_mean", "area_mean")]

diagnosis 변수를 양성과 악성으로 변환

selected_data$diagnosis <- ifelse(selected_data$diagnosis == "B", "Benign", "Malignant")

히스토그램 및 확률밀도함수 시각화

주요 변수 히스토그램 및 확률 밀도 함수

```
key_vars <- c("radius_mean", "texture_mean", "perimeter_mean",
"area_mean")
```

그래프 저장 리스트 생성

```
plot_list <- list()

for (var in key_vars){
p <- ggplot(selected_data, aes_string(x = var, fill = "diagnosis"))
+
geom_histogram(alpha = 0.5, bins = 30, position = "identity",
aes(y = ..density..)) +
geom_density(alpha = 0.7) +
labs(title = paste(var, "의 분포 및 확률 밀도 함수"),
x = var,
```

히스토그램 및 확률밀도함수 시각화

```
y = "밀도") +
theme_minimal() +
scale_fill_manual(values = c("skyblue", "coral"))

plot_list[[var]] <- p
}
```

모든 그래프를 한 화면에 출력

```
do.call(grid.arrange, c(plot_list, ncol = 2))
```

결과

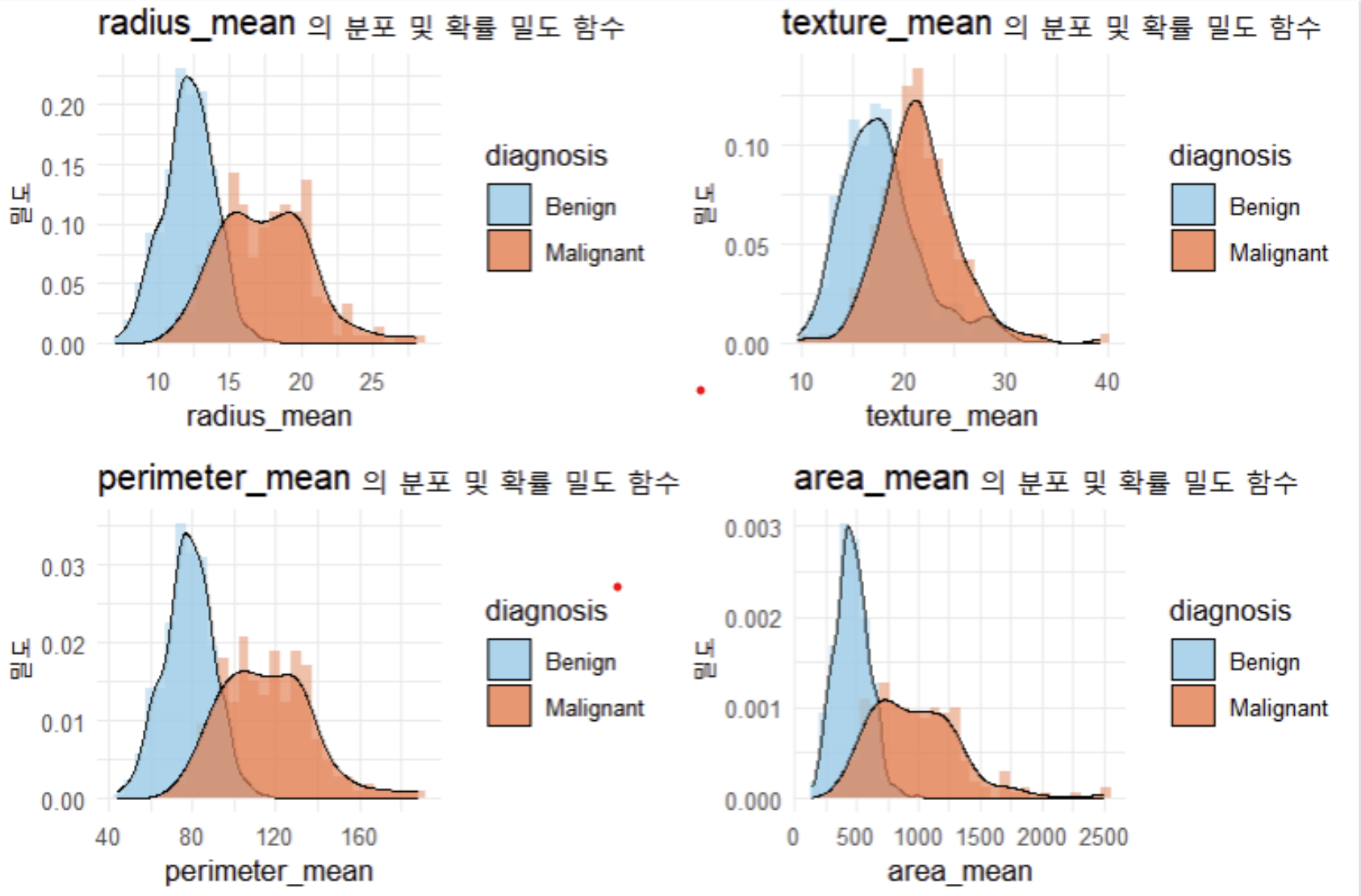


Figure 4: 히스토그램과 확률밀도함수

상관계수 및 피어슨, 스피어만 상관계수 분석

ID 컬럼 제외 (ID 컬럼 이름이 “id”로 가정)

numeric_data <- select(data, -id) %>% select(where(is.numeric))

피어슨 상관계수 분석

cor_matrix <- cor(numeric_data, method = “pearson”)

피어슨 상관계수 시각화

corrplot(cor_matrix, method = “circle”, type = “upper”, tl.col = “black”, tl.srt = 45)

스피어만 상관계수 분석과 시각화

cor_matrix_spearman <-cor(numeric_data,method=“spearman”)
corrplot(cor_matrix_spearman, method = “circle”, type = “upper”, tl.col = “black”, tl.srt = 45)

결과(피어슨 상관계수)

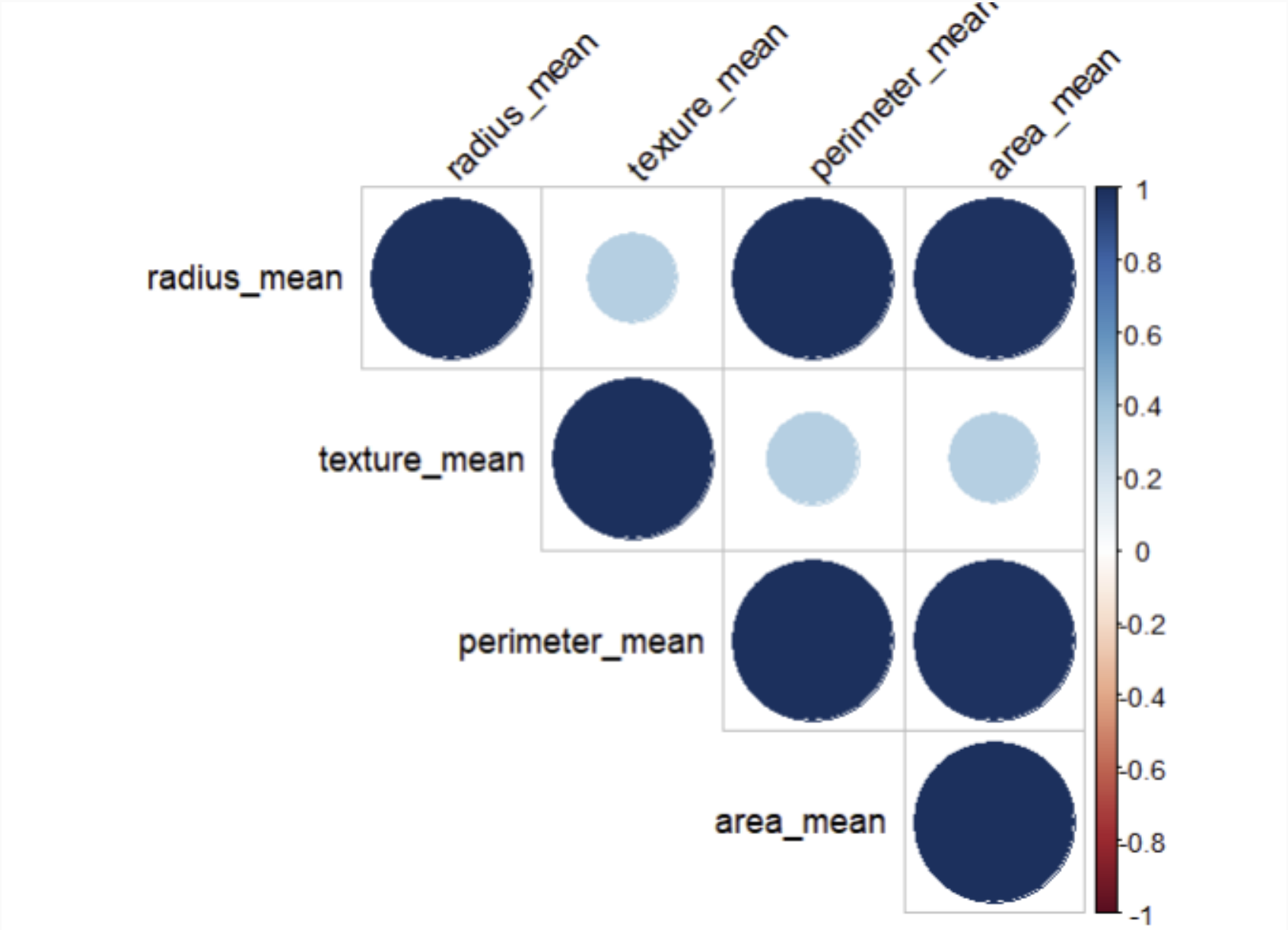
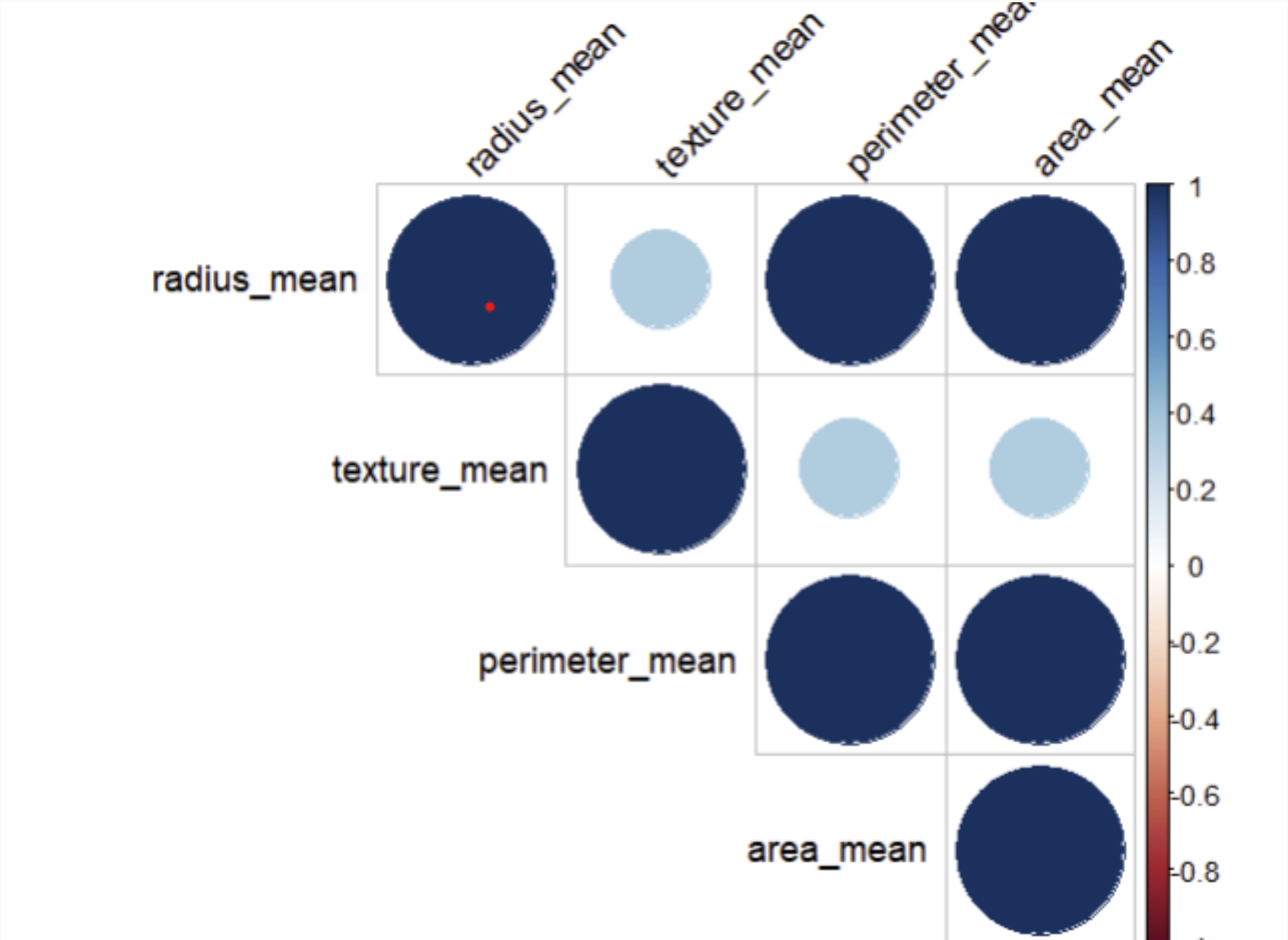


Figure 5. 피어슨 상관계수

결과


radius_mean
texture_mean
perimeter_mean
area_mean
radius_mean
texture_mean
perimeter_mean
area_mean
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8

상자그림

그래프 저장 리스트 생성

```
plot_list <- list()
for (var in key_vars) {
p <- ggplot(selected_data, aes_string(x = “diagnosis”, y = var, fill
= “diagnosis”)) +
geom_boxplot(outlier.color = “red”, outlier.size = 2) +
labs(title = paste(var, “의 양성과 악성 비교”),
x = “진단 결과”,
y = var) +
theme_minimal() +
scale_fill_manual(values = c(“skyblue”, “coral”))
plot_list[[var]] <- p}
```

모든 상자그림을 한 화면에 출력

```
do.call(grid.arrange, c(plot_list, ncol = 2))
```

결과

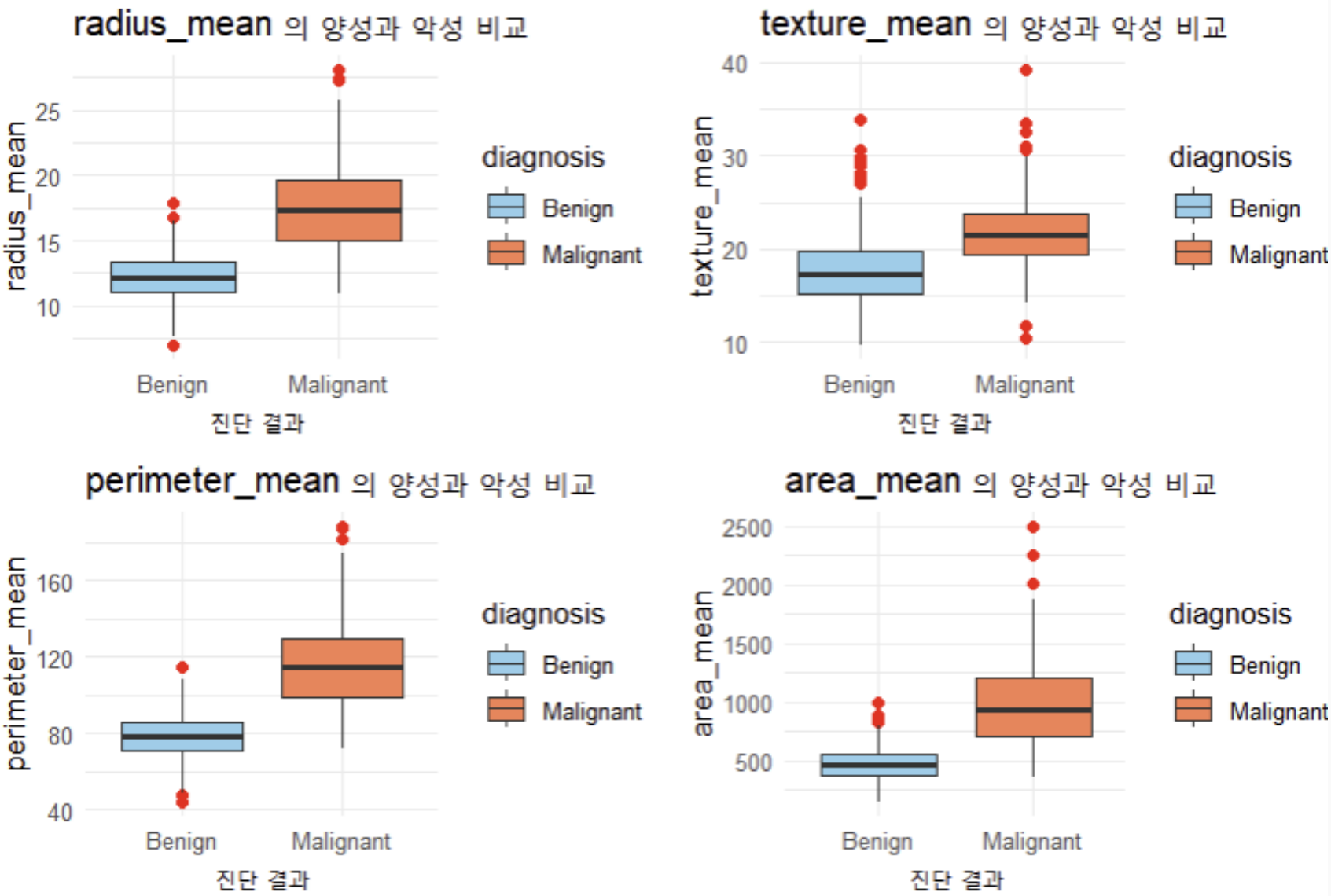


Figure 7: 상자그림

상자그림(ggplot,산점도)

1. Radius Mean by Diagnosis

```
plot1 <- ggplot(data, aes(x = diagnosis, y = radius_mean)) +
geom_jitter(width = 0.2, alpha = 0.6, color = "blue") +
geom_boxplot(alpha = 0.2, fill = "lightgray", outlier.shape = NA)
+
labs(title = "Radius Mean by Diagnosis",
x = "진단 결과 (B: 양성, M: 악성)",
y = "Radius Mean") +
theme_minimal()
```

상자그림(ggplot,산점도)

2. Texture Mean by Diagnosis

```
plot2 <- ggplot(data, aes(x = diagnosis, y = texture_mean)) +
geom_jitter(width = 0.2, alpha = 0.6, color = "green") +
geom_boxplot(alpha = 0.2, fill = "lightgray", outlier.shape = NA)
+
labs(title = "Texture Mean by Diagnosis",
x = "진단 결과 (B: 양성, M: 악성)",
y = "Texture Mean") +
theme_minimal()
```

상자그림(ggplot,산점도)

3. Perimeter Mean by Diagnosis

```
plot3 <- ggplot(data, aes(x = diagnosis, y = perimeter_mean)) +
geom_jitter(width = 0.2, alpha = 0.6, color = "orange") +
geom_boxplot(alpha = 0.2, fill = "lightgray", outlier.shape = NA)
+
labs(title = "Perimeter Mean by Diagnosis",
x = "진단 결과 (B: 양성, M: 악성)",
y = "Perimeter Mean") +
theme_minimal()
```

상자그림(ggplot,산점도)

4. Area Mean by Diagnosis

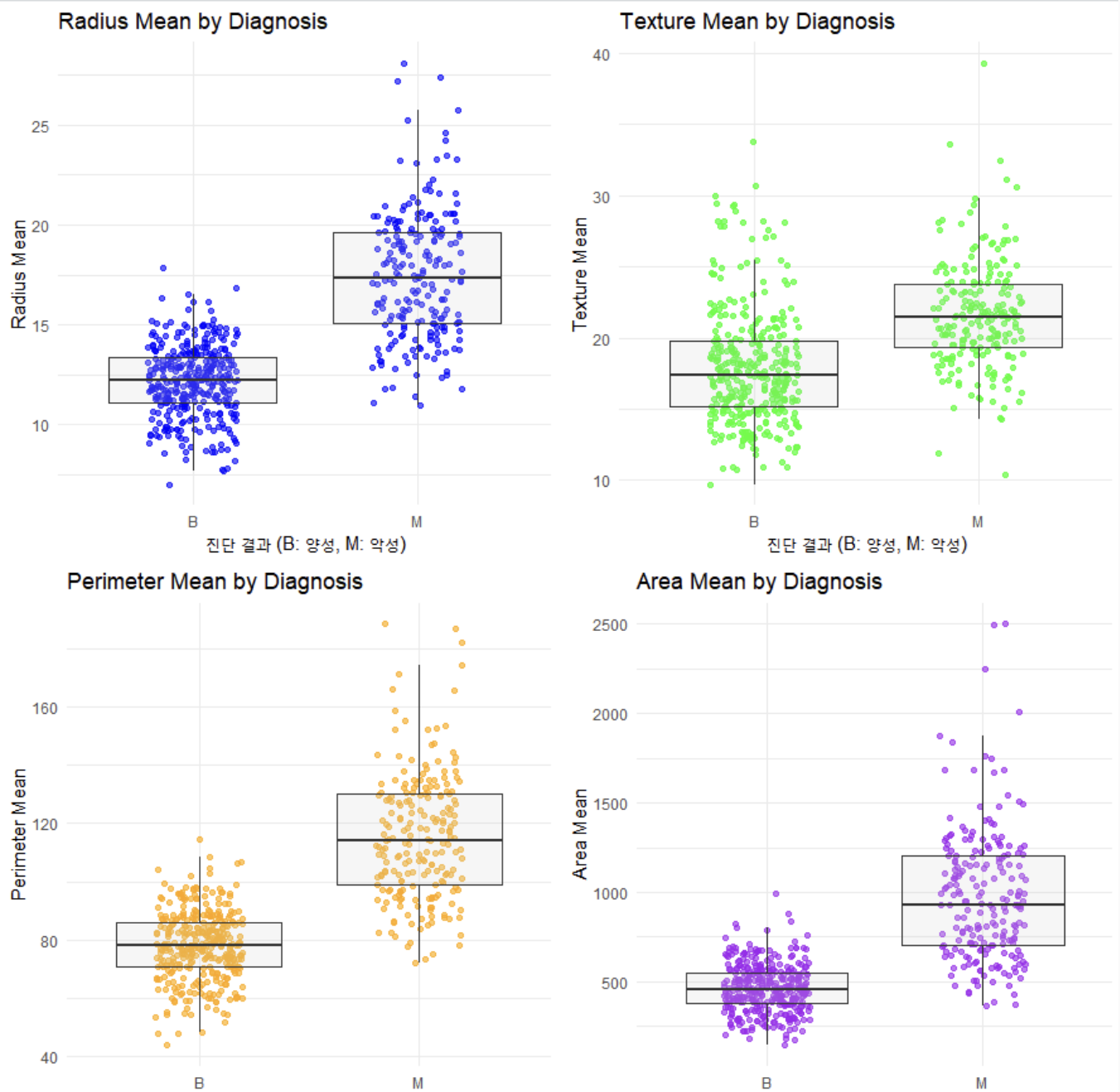
```
plot4 <- ggplot(data, aes(x = diagnosis, y = area_mean)) +
geom_jitter(width = 0.2, alpha = 0.6, color = "purple") +
geom_boxplot(alpha = 0.2, fill = "lightgray", outlier.shape = NA)
+
labs(title = "Area Mean by Diagnosis",
x = "진단 결과 (B: 양성, M: 악성)",
y = "Area Mean") +
theme_minimal()
```

상자그림(ggplot,산점도)

다중창 생성

```
if (!requireNamespace("gridExtra", quietly = TRUE)) {
install.packages("gridExtra")}
library(gridExtra)
grid.arrange(plot1, plot2, plot3, plot4, ncol = 2, nrow = 2)
```

결과


Radius Mean by Diagnosis
Radius Mean
25
20
15
10
B
M
진단 결과 (B: 양성, M: 악성)
Texture Mean by Diagnosis
Texture Mean
40
30
20
10
B
M
진단 결과 (B: 양성, M: 악성)
Perimeter Mean by Diagnosis
Perimeter Mean
160
120
80
40
B
M
Area Mean by Diagnosis
Area Mean
2500
2000
1500
1000
500
B
M

카이제곱 분포 시각화

ggplot2 로드

```
library(ggplot2)
```

필요한 변수 정의

```
variables <- c("radius_mean", "texture_mean", "perimeter_mean", "area_mean")
```

각 변수의 카이제곱 분포 시각화

```
for (var in variables) {
observed_data <- na.omit(data[[var]])

df_fixed <- 30

x <- seq(0, max(observed_data) * 1.5, length.out = 1000)
chisq_density <- dchisq(x, df = df_fixed)

plot_data <- data.frame(x = x, chisq_density = chisq_density)
```

카이제곱 분포 시각화

```
p <- ggplot(plot_data, aes(x = x, y = chisq_density)) +
geom_line(color = "red", linewidth = 1.2) +
labs(title = paste(var, ": Chi-Square Distribution"),
subtitle = paste("Fixed Degrees of Freedom (df):", df_fixed),
x = "Value",
y = "Density") +
theme_minimal() +
theme(plot.title = element_text(size = 14, face = "bold"),
plot.subtitle = element_text(size = 12))
print(p)}
```

결과

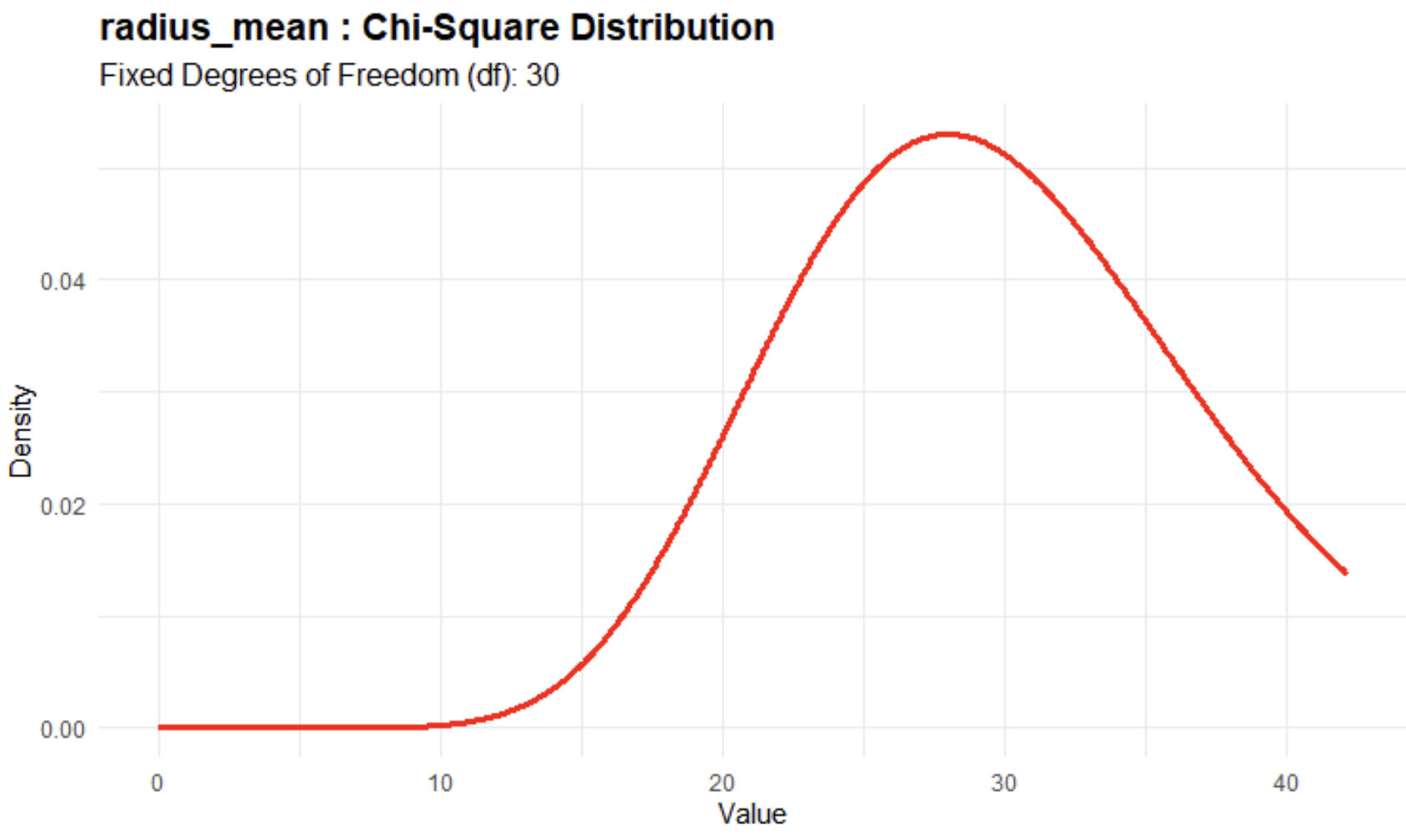


Figure 9: radius의 카이제곱분포

결과

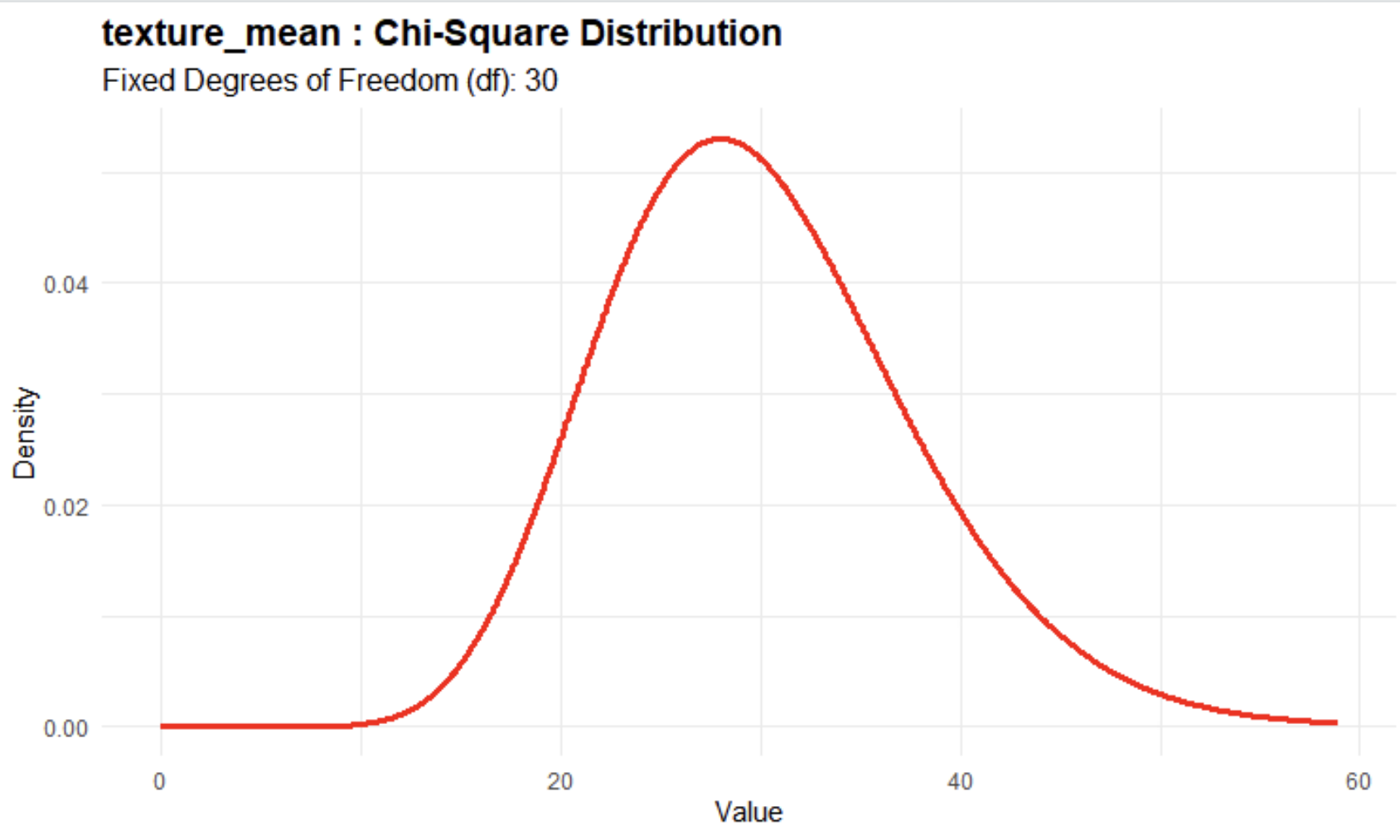


Figure 10: texture_mean의 카이제곱분포

결과

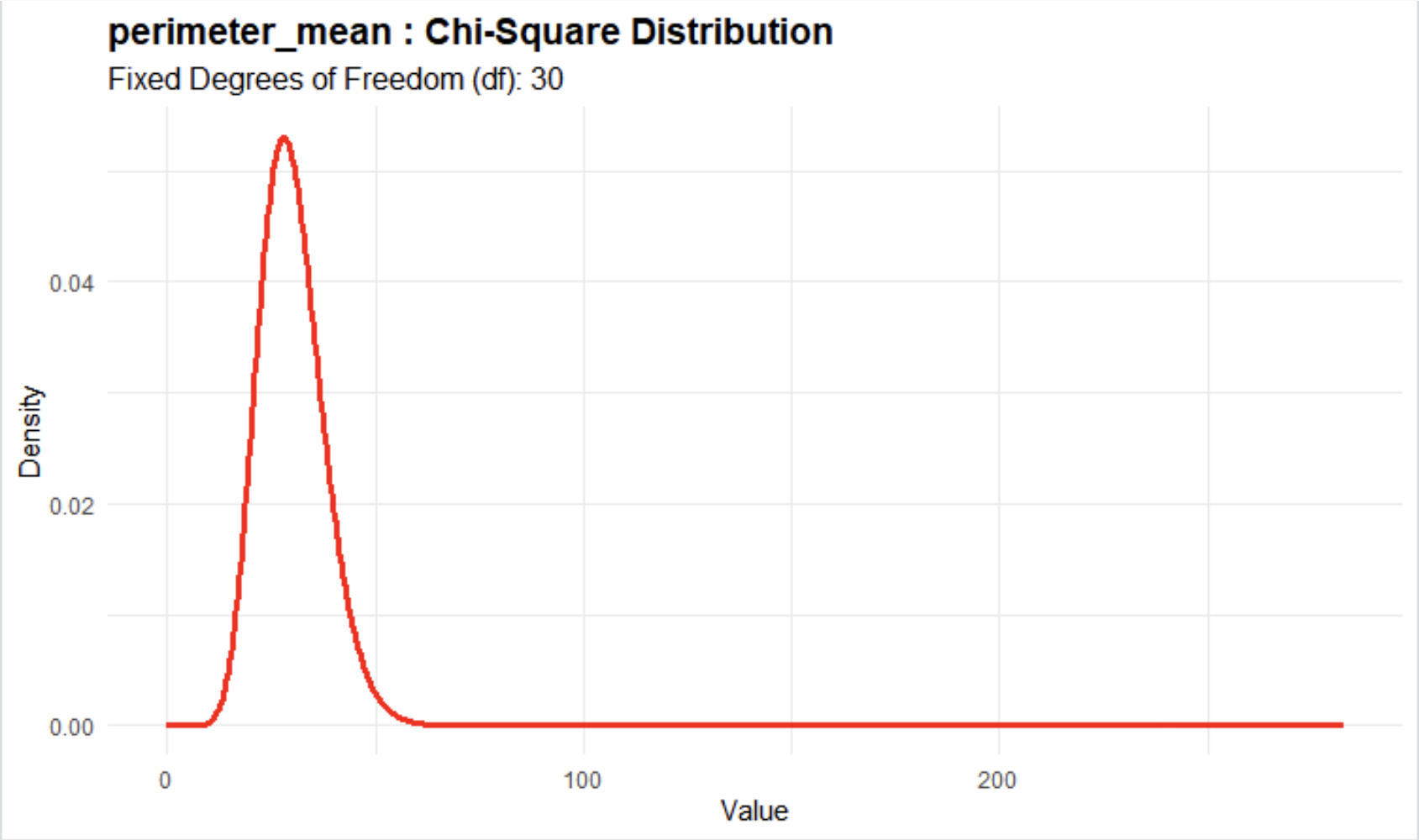


Figure 11: perimeter_mean의 카이제곱분포

결과

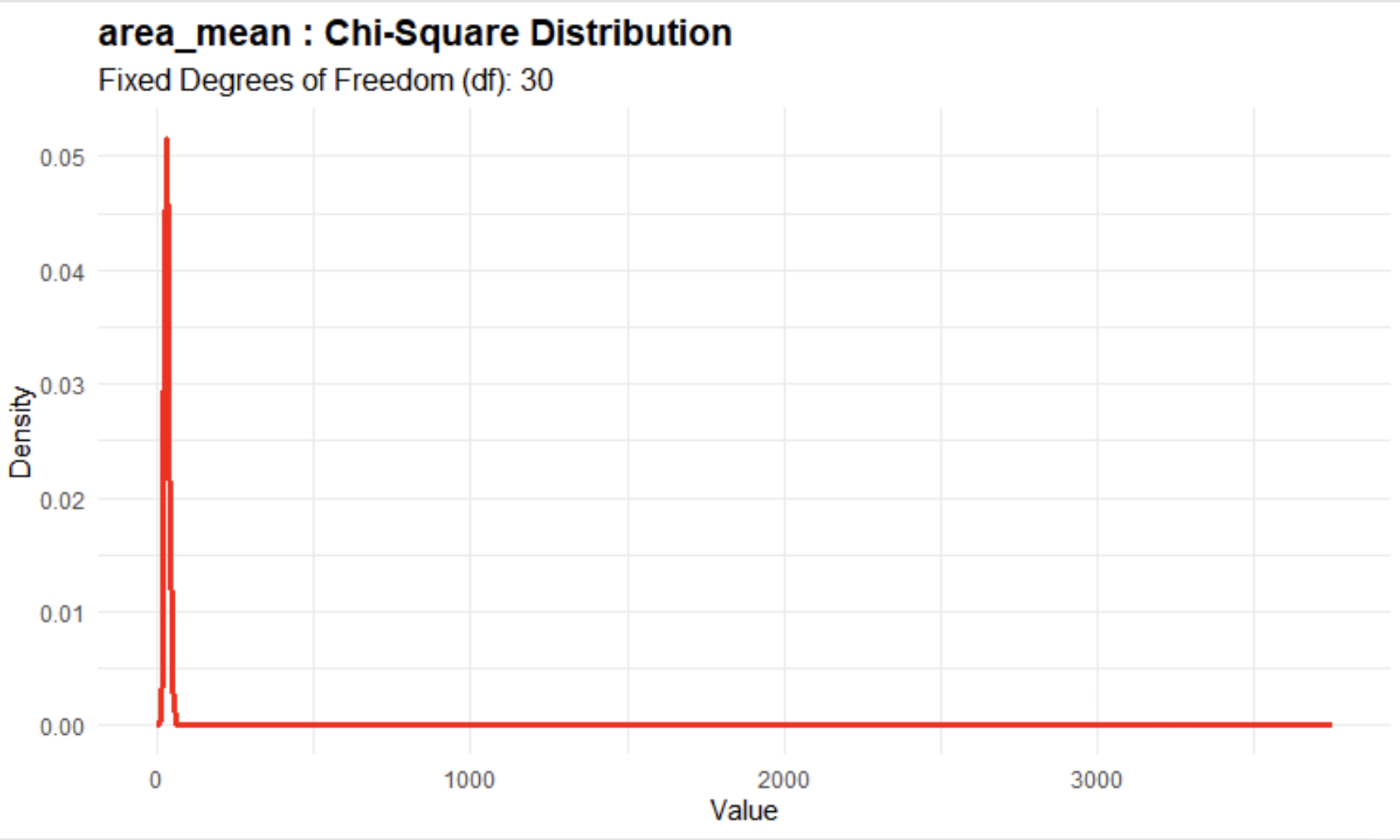


Figure 12: area_mean의 카이제곱분포

켄달의 타우 값 계산하고 시각화하기

필요한 패키지 설치

```
if (!require("corrplot")) install.packages("corrplot")
if (!require("readr")) install.packages("readr")
```

패키지 로드

```
library(corrplot)
library(readr)
```

데이터 불러오기

```
data <- read.csv("C:/Users/b612r/Desktop/real_data.csv")
```

켄달의 타우 값 계산하고 시각화하기

분석에 사용할 변수 선택

variables <- data[c("radius_mean", "texture_mean",
"perimeter_mean", "area_mean")]

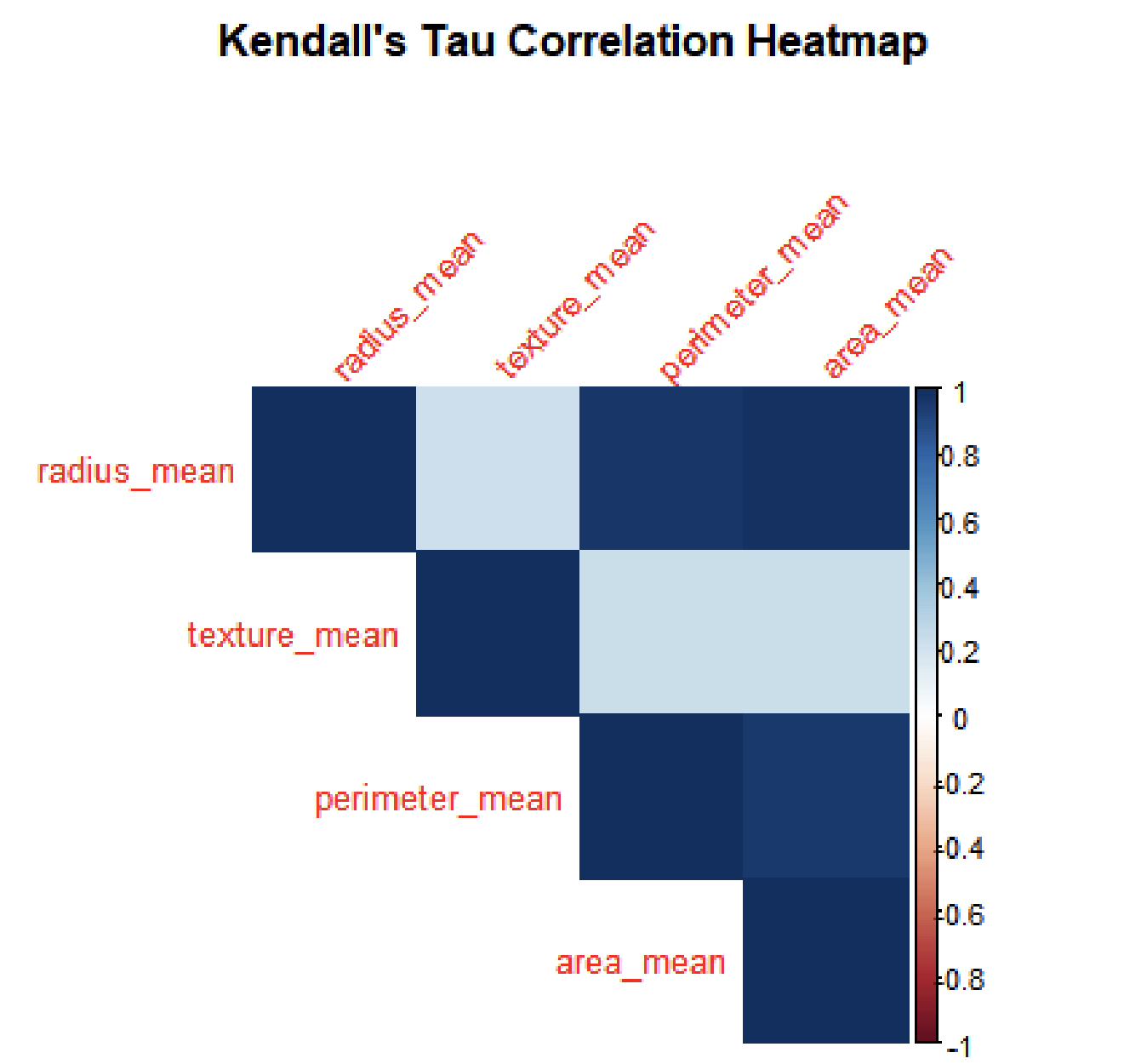
Kendall's Tau 계산

kendall_corr <- cor(variables, method = "kendall")

Kendall's Tau 히트맵 시각화

corrplot(kendall_corr, method = "color", type = "upper",
tl.col = "red", tl.srt = 45,
title = "Kendall's Tau Correlation Heatmap",
mar = c(0, 0, 2, 0))

결과


Kendall's Tau Correlation Heatmap
radius_mean
texture_mean
perimeter_mean
area_mean
radius_mean
texture_mean
perimeter_mean
area_mean
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1

선형 회귀 분석하고 시각화하기

필요한 패키지 설치 및 로드

```
if (!requireNamespace("ggplot2", quietly = TRUE)) {
install.packages("ggplot2")}
if (!requireNamespace("patchwork", quietly = TRUE)) {
install.packages("patchwork")}

library(ggplot2) library(patchwork)
```

선형 회귀 분석하고 시각화하기

1. Radius Mean vs Area Mean

plot1 <- ggplot(data, aes(x = radius_mean, y = area_mean)) + geom_point(color = “blue”, alpha = 0.6) + geom_smooth(method = “lm”, color = “red”, se = FALSE) + labs(title = “Radius Mean vs Area Mean”, x = “Radius Mean”, y = “Area Mean”) + theme_minimal()

선형 회귀 분석하고 시각화하기

2. Texture Mean vs Area Mean

plot2 <- ggplot(data, aes(x = texture_mean, y = area_mean)) +
geom_point(color = “green”, alpha = 0.6) +
geom_smooth(method = “lm”, color = “red”, se = FALSE) +
labs(title = “Texture Mean vs Area Mean”, x = “Texture Mean”, y
= “Area Mean”) + theme_minimal()

선형 회귀 분석하고 시각화하기

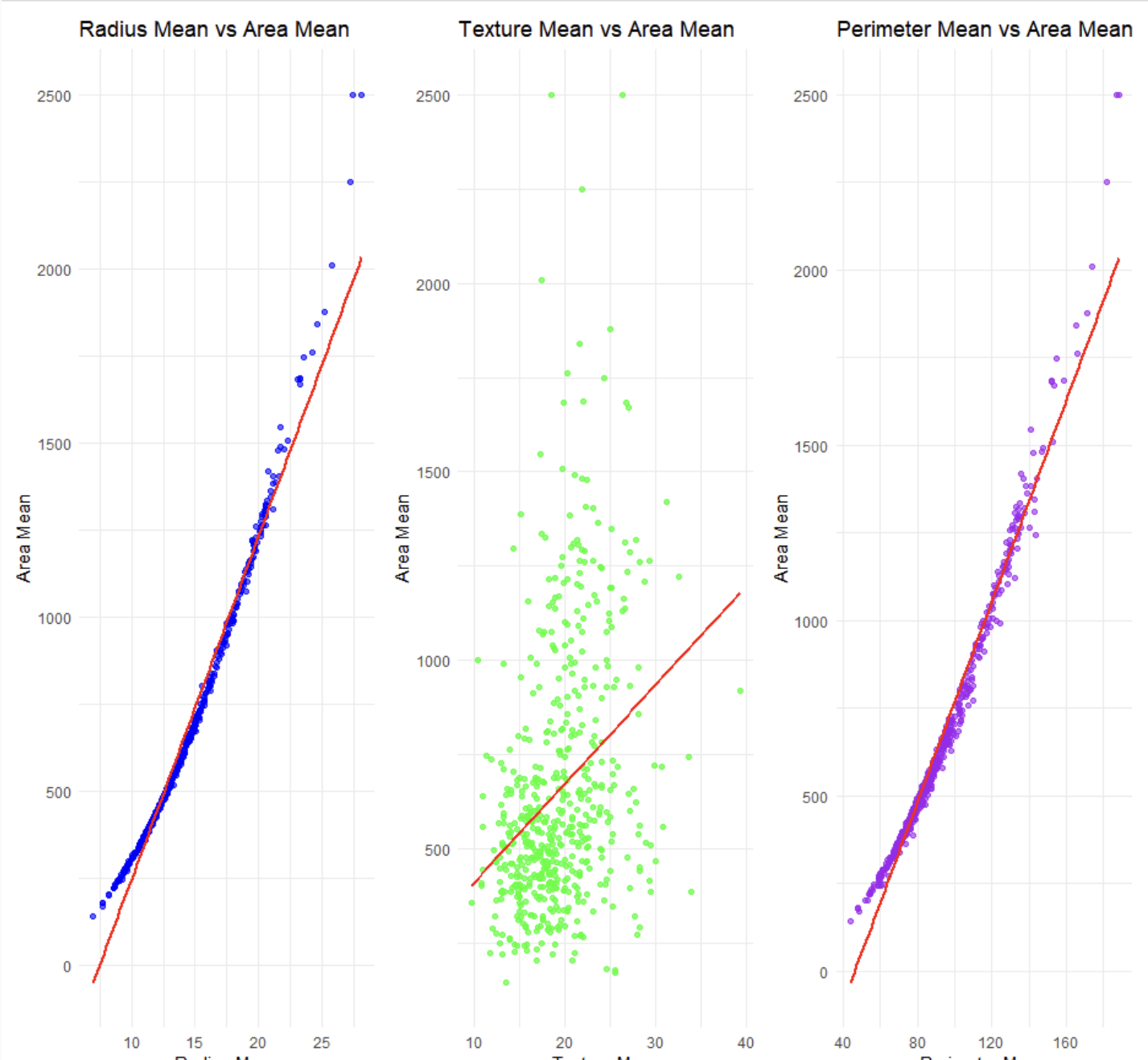
3. Perimeter Mean vs Area Mean

```
plot3 <- ggplot(data, aes(x = perimeter_mean, y = area_mean)) +
geom_point(color = "purple", alpha = 0.6) +
geom_smooth(method = "lm", color = "red", se = FALSE) +
labs(title = "Perimeter Mean vs Area Mean", x = "Perimeter
Mean", y = "Area Mean") + theme_minimal()
```

다중창 생성: 1행 3열로 정렬

```
plot1 + plot2 + plot3 + plot_layout(ncol = 3)
```

결과


Radius Mean vs Area Mean
Texture Mean vs Area Mean
Perimeter Mean vs Area Mean
Area Mean
2500
2000
1500
1000
500
0
10
15
20
25
30
40
80
120
160

이상 발표를 마칩니다.

감사합니다.