homework4.R

Barbara Cernosa

2025-03-31

library(ggplot2)  
library(mice)

library(psych)

library(norm)

library(mitools)  
  
data("diamonds")  
df <- diamonds[, c("price", "carat", "depth", "cut", "color")]   
df$cut <- as.numeric(factor(df$cut, ordered = TRUE))  
df$color <- as.numeric(factor(df$color, ordered = TRUE))  
  
# Introduce 30% MCAR missing data  
set.seed(123)  
df\_missing <- df  
total\_elements <- prod(dim(df\_missing))   
nMiss <- round(total\_elements \* 0.3)   
if (nMiss > total\_elements) nMiss <- total\_elements   
  
# Convert to vector, introduce missingness  
df\_vec <- as.vector(as.matrix(df\_missing))   
df\_vec[sample(total\_elements, nMiss)] <- NA   
df\_missing <- as.data.frame(matrix(df\_vec, nrow = nrow(df\_missing), ncol = ncol(df\_missing)))   
colnames(df\_missing) <- colnames(df)   
  
missing\_rate <- sum(is.na(df\_missing)) / total\_elements   
print(paste("Missing rate:", round(missing\_rate, 2)))

## [1] "Missing rate: 0.3"

##############################################################################  
# Speculate on Missing Data Mechanism  
# MCAR: We randomly introduced 30% missingness, independent of any variable.  
  
##############################################################################  
# Task 1: Confidence Intervals for Numeric Variables Means  
# Complete case analysis  
num\_vars <- c("price", "carat", "depth")  
ci\_complete <- sapply(df[num\_vars], function(x) {  
 x <- na.omit(x)  
 mean\_x <- mean(x)  
 se\_x <- sd(x) / sqrt(length(x))  
 c(mean\_x - 1.96 \* se\_x, mean\_x + 1.96 \* se\_x)   
})  
colnames(ci\_complete) <- num\_vars  
print(round(t(ci\_complete), 2))

## [,1] [,2]  
## price 3899.13 3966.47  
## carat 0.79 0.80  
## depth 61.74 61.76

# Missing data complete case analysis   
ci\_missing <- sapply(df\_missing[num\_vars], function(x) {  
 x <- na.omit(x)  
 mean\_x <- mean(x)  
 se\_x <- sd(x) / sqrt(length(x))  
 c(mean\_x - 1.96 \* se\_x, mean\_x + 1.96 \* se\_x) # 95% CI  
})  
colnames(ci\_missing) <- num\_vars  
print(round(t(ci\_missing), 2))

## [,1] [,2]  
## price 3905.17 3986.29  
## carat 0.79 0.80  
## depth 61.73 61.76

# MICE imputation (PMM) for CI  
imp <- mice(df\_missing, m = 5, method = "pmm", seed = 500, printFlag = FALSE)  
mi\_means <- sapply(num\_vars, function(var) {  
 means <- sapply(1:5, function(i) mean(complete(imp, i)[[var]]))  
 vars <- sapply(1:5, function(i) var(complete(imp, i)[[var]]) / nrow(df\_missing))  
 mi\_result <- MIcombine(results = as.list(means), variances = as.list(vars))  
 c(mi\_result$coefficients - 1.96 \* sqrt(mi\_result$variance),   
 mi\_result$coefficients + 1.96 \* sqrt(mi\_result$variance))  
})  
colnames(mi\_means) <- num\_vars  
print(round(t(mi\_means), 2))

## [,1] [,2]  
## price 3897.38 3972.87  
## carat 0.79 0.80  
## depth 61.73 61.76

# Missing data widens CIs (especially for price), while MICE restores them closer to Complete, validating imputation’s benefit under MCAR.  
# Price: Complete is slightly narrower than Missing, showing reduced precision with 30% missing data. MICE aligns closely with Complete, suggesting effective recovery of the mean.  
# Carat: All three are nearly identical.  
# Depth: Complete has the narrowest range, Missing slightly wider, and MICE matches Missing’s width but centers like Complete, showing imputation preserves precision.  
  
##############################################################################  
# Task 2: EM Imputation  
dataPrep <- prelim.norm(as.matrix(df\_missing))   
thetahat <- em.norm(dataPrep)

## Iterations of EM:   
## 1...2...3...4...5...6...7...8...9...10...11...12...13...14...15...16...17...18...

rngseed(1234567)   
impEM <- imp.norm(s = dataPrep, theta = thetahat, x = as.matrix(df\_missing))  
imputed\_data\_em <- as.data.frame(impEM)  
colnames(imputed\_data\_em) <- colnames(df\_missing)   
  
##############################################################################  
# Task 3: Linear Regression Comparison  
# Define formula  
formula <- price ~ carat + depth + cut + color  
  
# Linear models for each dataset  
lm\_complete <- lm(formula, data = df)  
lm\_missing <- lm(formula, data = df\_missing)   
df\_imputed\_mice <- complete(imp, 1)   
lm\_mice <- lm(formula, data = df\_imputed\_mice)   
lm\_em <- lm(formula, data = imputed\_data\_em)  
lm\_mice\_pooled <- with(imp, lm(price ~ carat + depth + cut + color))   
pooled\_mice <- pool(lm\_mice\_pooled)  
# Comment: lm\_missing shifts slightly with fewer cases; lm\_mice exaggerates some effects (e.g., intercept, depth) from one imputation; lm\_mice\_pooled balances these across 5 imputations, aligning closer to complete case for key predictors like carat.  
  
# Extract coefficients  
coef\_comp <- cbind(Complete = coef(lm\_complete), Missing = coef(lm\_missing),   
 MICE\_Single = coef(lm\_mice), MICE\_Pooled = pooled\_mice$pooled$estimate,   
 EM = coef(lm\_em))  
print(round(coef\_comp, 3))

## Complete Missing MICE\_Single MICE\_Pooled EM  
## (Intercept) 379.919 541.682 1958.352 1117.222 802.070  
## carat 8099.823 8195.022 8187.145 8160.519 8125.333  
## depth -48.611 -52.418 -77.556 -64.366 -55.435  
## cut 251.619 250.740 292.921 298.082 251.544  
## color -247.877 -246.280 -249.328 -245.019 -252.218

# Extract R-squared values  
rsq\_comp <- c(Complete = summary(lm\_complete)$r.squared, Missing = summary(lm\_missing)$r.squared,   
 MICE\_Single = summary(lm\_mice)$r.squared, MICE\_Pooled = pool.r.squared(lm\_mice\_pooled)[1],   
 EM = summary(lm\_em)$r.squared)  
print(round(rsq\_comp, 3))

## Complete Missing MICE\_Single MICE\_Pooled EM   
## 0.865 0.866 0.864 0.863 0.867

# Intercept: Very wide range; EM lowest, MICE\_Single highest, MICE\_Pooled moderates between imputations, Complete as baseline.  
# Carat: Stable, highest in Missing, showing robust predictive power across methods. Depth: More negative in MICE vs. Complete and EM, suggesting imputation amplifies it. Cut & Color: Consistent.  
# R-squared: Tight range, EM highest, MICE\_Pooled lowest, indicating model fit is robust to missingness and imputation.  
# Coefficients show more variability (especially intercept, depth) than R-squared, reflecting multivariate sensitivity to missing data treatment, while high R-squared consistency suggests overall explanatory power holds steady.  
  
# Extended Mice model comparisons  
lm\_mice0 <- with(imp, lm(price ~ 1)) # Intercept-only model  
lm\_mice\_reduced <- with(imp, lm(price ~ carat + depth + color))   
print(D1(lm\_mice\_pooled, lm\_mice0))

## test statistic df1 df2 dfcom p.value riv  
## 1 ~~ 2 21220.37 4 23.98341 53935 3.586668e-42 3.002326

print(D1(lm\_mice\_pooled, lm\_mice\_reduced))

## test statistic df1 df2 dfcom p.value riv  
## 1 ~~ 2 753.3237 1 4 53935 1.047983e-05 2.359628

print(round(pool.r.squared(lm\_mice\_pooled, adjusted = FALSE)[1], 3))

## [1] 0.863

print(round(pool.r.squared(lm\_mice\_pooled, adjusted = TRUE)[1], 3))

## [1] 0.863

# Intercept: reduced is much higher.  
# R-squared: reduced lower, reflecting cut’s contribution.  
# Reduced model shifts coefficients (larger intercept, depth) due to cut’s absence, but full model’s stability and D1 significance highlight cut’s importance.  
  
  
##############################################################################  
# Task 4: PCA Comparison  
# PCA on each dataset  
pca\_complete <- principal(df, nfactors = 5, rotate = "oblimin")

## Loading required namespace: GPArotation

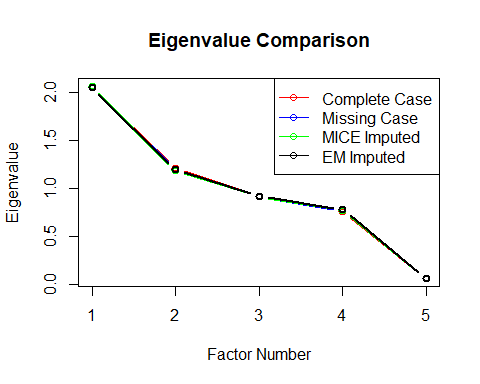
pca\_missing <- principal(df\_missing, nfactors = 5, rotate = "oblimin")  
pca\_mice <- principal(df\_imputed\_mice, nfactors = 5, rotate = "oblimin")  
pca\_em <- principal(imputed\_data\_em, nfactors = 5, rotate = "oblimin")  
# PCA structure is robust across complete, missing, and imputed datasets, with price and carat driving TC1; minor variations reflect imputation effects, but overall consistency validates MCAR handling.  
# Correlations: TC1-TC3 steady, TC1-TC4 slightly more negative in MICE, others negligible differences.  
# Variance: TC1 loadings nearly identical, minor increase in Missing and EM; other components stable.  
  
# Comparing loadings for first component  
loadings\_comp <- cbind(Original = pca\_complete$loadings[,1], Missing = pca\_missing$loadings[,1],   
 MICE = pca\_mice$loadings[,1], EM = pca\_em$loadings[,1])  
print(round(loadings\_comp, 3))

## Original Missing MICE EM  
## price 0.997 0.996 0.999 0.997  
## carat 0.957 0.958 0.953 0.958  
## depth 0.000 0.000 0.000 0.000  
## cut 0.000 0.000 0.001 0.000  
## color 0.000 0.000 -0.001 0.000

# TC1 dominated by price and carat across all, with MICE slightly lower; depth, cut, color near 0, showing consistent structure.  
  
# Comparing eigenvalues  
eigenvalues <- cbind(Complete = pca\_complete$values, Missing = pca\_missing$values,   
 MICE = pca\_mice$values, EM = pca\_em$values)  
colnames(eigenvalues) <- c("Complete Case", "Missing Case", "MICE Imputed", "EM Imputed")  
print(round(eigenvalues, 3))

## Complete Case Missing Case MICE Imputed EM Imputed  
## [1,] 2.047 2.047 2.061 2.044  
## [2,] 1.207 1.202 1.183 1.193  
## [3,] 0.914 0.915 0.914 0.920  
## [4,] 0.765 0.769 0.775 0.777  
## [5,] 0.067 0.066 0.067 0.066

# TC1 slightly higher in MICE vs. others, indicating imputation may inflate primary variance slightly; others consistent.  
  
# *E*igenvalue comparison  
matplot(eigenvalues, type = "b", pch = 1, col = c("red", "blue", "green", "black"), lwd = 2, lty = 1,  
 xlab = "Factor Number", ylab = "Eigenvalue", main = "Eigenvalue Comparison")  
legend("topright", legend = colnames(eigenvalues), col = c("red", "blue", "green", "black"), lty = 1, pch = 1)



##############################################################################  
# 5: Interpretation of result  
  
# CIs showed minor widening in the missing dataset, with MICE and EM effectively recovering precision close to the complete (original Diamonds) case.   
# Regression revealed a larger coefficient variability than R-squared, highlighting multivariate sensitivity, but imputation kept relationships intact with the original.   
# PCA structure held steady, validating imputation under MCAR, with price and carat affecting variance and minimal imputation effects.   
# Both PMM and EM handled MCAR well, preserving statistical integrity across analyses, with Mice's pooled approach showing balanced estimates.