# Job Proficiency Analysis

**Your Name**

**2025-03-10**

## Problem 1

## Introduction

This vignette analyzes job proficiency data using a dataset stored in `data.xlsx`. We visualize the relationship between job proficiency and test scores using scatterplots.

## Load Data

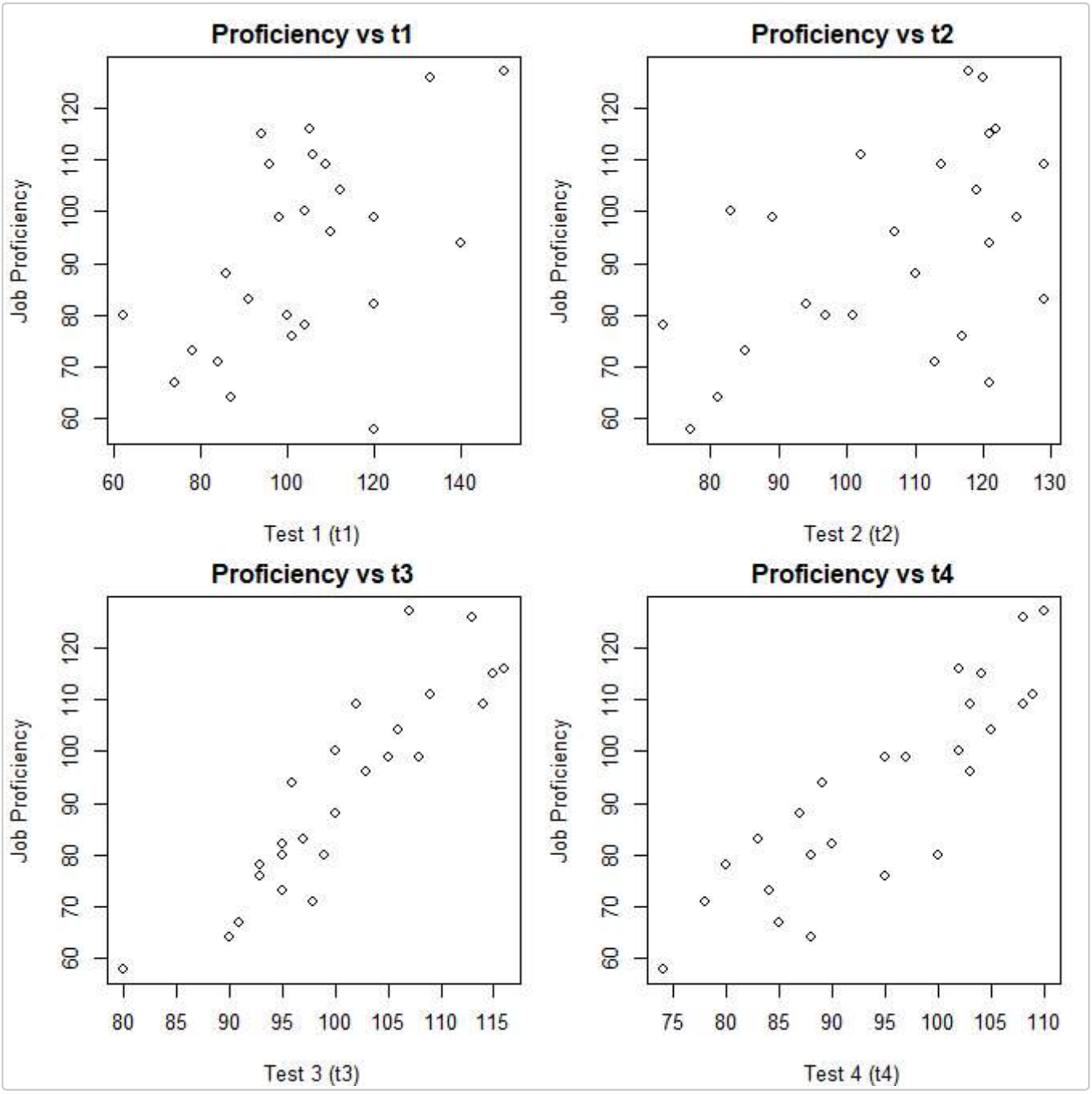
```
# Sử dụng thư viện readxl để đọc dữ liệu từ file Excel
library(readxl)

# Đọc dữ liệu từ file "data.xlsx"
# Tham số col_names = FALSE có nghĩa là file không có hàng tiêu đề nên chúng ta sẽ tự đặt tên sau
    này.
jobs <- read_excel("data.xlsx", col_names = FALSE)
#> New names:
#> • `` -> `...1`
#> • `` -> `...2`
#> • `` -> `...3`
#> • `` -> `...4`
#> • `` -> `...5`

# Gán tên cho các cột của dataframe:
# - "proficiency": độ chuyên môn của công việc
# - "t1", "t2", "t3", "t4": điểm số của các bài kiểm tra tương ứng.
colnames(jobs) <- c("proficiency", "t1", "t2", "t3", "t4")

# Hiển thị tóm tắt thống kê của dữ liệu (bao gồm min, max, median, mean, ...)
summary(jobs)
#>   proficiency          t1              t2              t3
#>  Min.   : 58.0   Min.   : 62.0   Min.   : 73.0   Min.   : 80.0
#>  1st Qu.: 78.0   1st Qu.: 91.0   1st Qu.: 94.0   1st Qu.: 95.0
#>  Median : 94.0   Median :104.0   Median :113.0   Median :100.0
#>  Mean   : 92.2   Mean   :103.4   Mean   :106.7   Mean   :100.8
#>  3rd Qu.:109.0   3rd Qu.:112.0   3rd Qu.:121.0   3rd Qu.:107.0
#>  Max.   :127.0   Max.   :150.0   Max.   :129.0   Max.   :116.0
#>        t4
#>  Min.   : 74.00
```

```
#>  1st Qu.: 87.00
#>  Median : 95.00
#>  Mean   : 94.68
#>  3rd Qu.:103.00
#>  Max.   :110.00
```

## Visualize Data

We create scatterplots to visualize the relationship between job proficiency and each test score.

```
# Thiết lập kích thước biểu đồ để hiển thị rõ ràng
options(repr.plot.width = 20, repr.plot.height = 20)

# Chia vùng vẽ thành 2 hàng x 2 cột để hiển thị 4 biểu đồ cùng lúc.
# Tham số mfrow = c(2, 2) điều chỉnh lưới vẽ biểu đồ.
# Tham số mar = c(4, 4, 2, 1) thiết lập lề (margin) cho biểu đồ: dưới, trái, trên, phải.
par(mfrow = c(2, 2), mar = c(4, 4, 2, 1))

# Biểu đồ phân tán: điểm số Test 1 (t1) vs độ chuyên môn (proficiency)
plot(jobs$t1, jobs$proficiency,
     xlab = "Test 1 (t1)",      # Nhãn trục X: điểm của bài kiểm tra t1
     ylab = "Job Proficiency",  # Nhãn trục Y: độ chuyên môn
     main = "Proficiency vs t1")# Tiêu đề biểu đồ

# Biểu đồ phân tán: điểm số Test 2 (t2) vs độ chuyên môn
plot(jobs$t2, jobs$proficiency,
     xlab = "Test 2 (t2)",
     ylab = "Job Proficiency",
     main = "Proficiency vs t2")

# Biểu đồ phân tán: điểm số Test 3 (t3) vs độ chuyên môn
plot(jobs$t3, jobs$proficiency,
     xlab = "Test 3 (t3)",
     ylab = "Job Proficiency",
     main = "Proficiency vs t3")

# Biểu đồ phân tán: điểm số Test 4 (t4) vs độ chuyên môn
plot(jobs$t4, jobs$proficiency,
     xlab = "Test 4 (t4)",
     ylab = "Job Proficiency",
     main = "Proficiency vs t4")
```




```
# Sau khi hoàn thành vẽ 4 biểu đồ, đặt lại vùng vẽ về mặc định (1 hàng, 1 cột)
par(mfrow = c(1, 1))
```

## Conclusion

The scatterplots provide insights into how each test score correlates with job proficiency. Further statistical analysis could quantify these relationships.

## Problem 2

```r
model_stats <- function(fit, MSE_full){
  # fit: đối tượng lm (mô hình hồi quy)
  # MSE_full: Mean Squared Error của mô hình đầy đủ dùng để tính Mallows' Cp

  n <- length(fit$residuals)    # Số lượng quan sát
  p <- length(coef(fit))        # Số tham số (bao gồm intercept)

  # Tính hệ số xác định R² và R² hiệu chỉnh từ kết quả summary của mô hình
  r2     <- summary(fit)$r.squared
  adjr2  <- summary(fit)$adj.r.squared

  # Tính tổng bình phương sai số (SSE) của mô hình fit
  sse_p  <- sum(resid(fit)^2)

  # Tính chỉ số PRESS: PRESS = Σ[(e_i / (1 - h_ii))^2]
  # Trong đó: e_i là phần dư, h_ii là giá trị hat (đo lường ảnh hưởng của mỗi quan sát)
  hat    <- lm.influence(fit)$hat  # Lấy giá trị hat của mô hình
  res    <- resid(fit)             # Lấy phần dư của mô hình
  press  <- sum((res/(1 - hat))^2)

  # Tính AIC và BIC của mô hình
  aic_val <- AIC(fit)
  bic_val <- BIC(fit)

  # Tính Mallows' Cp với công thức: Cp = SSE_p / MSE_full - (n - 2*p)
  cp_val  <- sse_p / MSE_full - (n - 2*p)

  # Trả về vector chứa các chỉ số
  return(c(p, r2, adjr2, press, aic_val, bic_val, cp_val))
}


# Xây dựng mô hình hồi quy đầy đủ với tất cả các biến dự báo
lm_full <- lm(proficiency ~ t1 + t2 + t3 + t4, data = jobs)

# Tính tổng bình phương sai số (SSE) của mô hình đầy đủ
SSE_full <- sum(resid(lm_full)^2)

# Số lượng quan sát trong dữ liệu
n <- nrow(jobs)

# Số tham số của mô hình đầy đủ (4 biến dự báo + intercept)
p_full <- length(coef(lm_full))

# Tính Mean Squared Error (MSE) của mô hình đầy đủ
MSE_full <- SSE_full / (n - p_full)


# Tạo dataframe rỗng để lưu kết quả
results <- data.frame(
```

```
  Model   = character(),
  p       = numeric(),  # Số tham số (bao gồm intercept)
  R2      = numeric(),
  AdjR2   = numeric(),
  PRESS   = numeric(),
  AIC     = numeric(),
  BIC     = numeric(),
  Cp      = numeric(),
  stringsAsFactors = FALSE
)

# Danh sách các biến dự báo
preds <- c("t1", "t2", "t3", "t4")

# Duyệt qua các tập con của biến dự báo với số lượng biến từ 0 đến 4
for (k in 0:4) {
  # Lấy tất cả tổ hợp k phần tử từ vector preds
  subset_list <- combn(preds, k)

  if (k == 0) {
    # Trường hợp k = 0: mô hình chỉ có intercept (không có biến dự báo)
    form <- as.formula("proficiency ~ 1")
    fit <- lm(form, data = jobs)
    stats <- model_stats(fit, MSE_full)

    results <- rbind(
      results,
      data.frame(
        Model = "Intercept Only",
        p     = stats[1],
        R2    = stats[2],
        AdjR2 = stats[3],
        PRESS = stats[4],
        AIC   = stats[5],
        BIC   = stats[6],
        Cp    = stats[7],
        stringsAsFactors = FALSE
      )
    )

  } else {
    # Trường hợp k > 0: duyệt qua từng tổ hợp của các biến dự báo
    for (i in 1:ncol(subset_list)) {
      vars <- subset_list[, i]  # Lấy tập hợp các biến dự báo cho mô hình hiện tại

      # Tạo công thức hồi quy từ các biến được chọn
      form <- as.formula(paste("proficiency ~", paste(vars, collapse = " + ")))

      # Xây dựng mô hình hồi quy với công thức trên
      fit <- lm(form, data = jobs)
      stats <- model_stats(fit, MSE_full)
```

```
      # Thêm kết quả của mô hình vào dataframe results
      results <- rbind(
        results,
        data.frame(
          Model = paste(vars, collapse = " + "),
          p     = stats[1],
          R2    = stats[2],
          AdjR2 = stats[3],
          PRESS = stats[4],
          AIC   = stats[5],
          BIC   = stats[6],
          Cp    = stats[7],
          stringsAsFactors = FALSE
        )
      )
    }
  }
}


# In kết quả các chỉ số thống kê của các mô hình con
knitr::kable(results, caption = "Bảng kết quả các mô hình dự báo")
```

Bảng kết quả các mô hình dự báo

| Model | p | R2 | AdjR2 | PRESS | AIC | BIC | Cp |
|---|---|---|---|---|---|---|---|
| Intercept Only | 1 | 0.0000000 | 0.0000000 | 9824.2188 | 222.2491 | 224.6868 | 515.964627 |
| t1 | 2 | 0.2646184 | 0.2326452 | 7791.5994 | 216.5649 | 220.2216 | 375.344689 |
| t2 | 2 | 0.2470147 | 0.2142762 | 7991.0964 | 217.1563 | 220.8130 | 384.832454 |
| t3 | 2 | 0.8047247 | 0.7962344 | 2064.5976 | 183.4155 | 187.0721 | 84.246496 |
| t4 | 2 | 0.7558329 | 0.7452170 | 2548.6349 | 189.0015 | 192.6581 | 110.597414 |
| t1 + t2 | 3 | 0.4641948 | 0.4154853 | 6444.0411 | 210.6495 | 215.5250 | 269.780029 |
| t1 + t3 | 3 | 0.9329956 | 0.9269043 | 760.9744 | 158.6741 | 163.5496 | 17.112978 |
| t1 + t4 | 3 | 0.8152656 | 0.7984716 | 2109.8967 | 184.0282 | 188.9037 | 80.565307 |
| t2 + t3 | 3 | 0.8060733 | 0.7884436 | 2206.6460 | 185.2422 | 190.1177 | 85.519650 |
| t2 + t4 | 3 | 0.7832923 | 0.7635916 | 2491.7979 | 188.0189 | 192.8944 | 97.797790 |
| t3 + t4 | 3 | 0.8772573 | 0.8660988 | 1449.6001 | 173.8075 | 178.6830 | 47.153985 |
| t1 + t2 + t3 | 4 | 0.9340931 | 0.9246779 | 831.1521 | 160.2613 | 166.3556 | 18.521465 |
| t1 + t2 + t4 | 4 | 0.8453581 | 0.8232664 | 1885.8454 | 181.5830 | 187.6774 | 66.346500 |
| t1 + t3 + t4 | 4 | 0.9615422 | 0.9560482 | 471.4520 | 146.7942 | 152.8886 | 3.727399 |
| t2 + t3 + t4 | 4 | 0.8789698 | 0.8616797 | 1570.5610 | 175.4562 | 181.5506 | 48.231020 |
| t1 + t2 + t3 + t4 | 5 | 0.9628918 | 0.9554702 | 518.9885 | 147.9011 | 155.2144 | 5.000000 |