Problem_3

Hindy Rossignol, Riya Parikh, Mrugank Pednekar, Ioannis Panagiotopoulos

2025-09-11

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.4     v tibble    3.3.0
## v purrr     1.1.0     v tidyr     1.3.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts
↪  to become errors
```

```
# Load the mtcars dataset
data(mtcars)
# Display basic information about the dataset
head(mtcars)
```

```
##                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
summary(mtcars)
```

```
##       mpg             cyl             disp             hp
##  Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.   : 52.0
##  1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5
##  Median :19.20   Median :6.000   Median :196.3   Median :123.0
##  Mean   :20.09   Mean   :6.188   Mean   :230.7   Mean   :146.7
##  3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0
##  Max.   :33.90   Max.   :8.000   Max.   :472.0   Max.   :335.0
##       drat             wt             qsec             vs
##  Min.   :2.760   Min.   :1.513   Min.   :14.50   Min.   :0.0000
##  1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000
##  Median :3.695   Median :3.325   Median :17.71   Median :0.0000
```

```
##  Mean   :3.597   Mean   :3.217   Mean   :17.85   Mean   :0.4375
##  3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000
##  Max.   :4.930   Max.   :5.424   Max.   :22.90   Max.   :1.0000
##        am              gear            carb
##  Min.   :0.0000   Min.   :3.000   Min.   :1.000
##  1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000
##  Median :0.0000   Median :4.000   Median :2.000
##  Mean   :0.4062   Mean   :3.688   Mean   :2.812
##  3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :1.0000   Max.   :5.000   Max.   :8.000
```

Sanity Check for our Models

To make sure our models are correct we will use a function to visualize their equation explicity.

```r
# This is our function to visualize the equations
eq_print <- function(mod, digits = 4, mult_sign = " * ") {
  b <- coef(mod)
  fmt <- function(x) formatC(x, digits = digits, format = "f")
  parts <- Map(function(name, val) {
    if (name == "(Intercept)") return(fmt(val))
    nm <- gsub(":", mult_sign, name, fixed = TRUE)
    paste0(ifelse(val >= 0, " + ", " - "), fmt(abs(val)), " ", nm)
  }, names(b), b)
  intercept <- parts[[which(names(b) == "(Intercept)")[1]]]
  others <- parts[names(b) != "(Intercept)"]
  paste0("y_hat = ", intercept, paste0(others, collapse = ""))
}
```

Part a) Fit Model 1 and show summary

```r
# Model 1: mpg ~ hp + wt + hp*wt
model1 <- lm(mpg ~ hp + wt + hp*wt, data = mtcars)

summary(model1)
```

```
##
## Call:
## lm(formula = mpg ~ hp + wt + hp * wt, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0632 -1.6491 -0.7362  1.4211  4.5513
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 49.80842    3.60516  13.816 5.01e-14 ***
## hp          -0.12010    0.02470  -4.863 4.04e-05 ***
## wt          -8.21662    1.26971  -6.471 5.20e-07 ***
## hp:wt        0.02785    0.00742   3.753 0.000811 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.153 on 28 degrees of freedom
## Multiple R-squared:  0.8848, Adjusted R-squared:  0.8724
## F-statistic: 71.66 on 3 and 28 DF,  p-value: 2.981e-13
```

```r
cat(sprintf("Model %d's equation: %s\n", 1, eq_print(model1)))
```

```
## Model 1's equation: y_hat = 49.8084 - 0.1201 hp - 8.2166 wt + 0.0278 hp * wt
```

Part b) Interpretation of coefficients in Model 1

The interpretation of the coefficients in Model 1:

- **hp coefficient (-0.12010)**: This is the effect of horsepower when weight is equal to zero. While a car cannot realistically weigh zero, this coefficient combines with the interaction term to determine the true effect of hp at different weights. Formally, the marginal effect of horsepower on mpg is:

$$\frac{\partial mpg}{\partial hp} = -0.12010 + 0.02785 \cdot wt$$

  This means that for each additional unit of horsepower, mpg decreases by about 0.12 miles per gallon, but this negative effect becomes less severe as weight increases.

- **wt coefficient (-8.21662)**: This is the effect of weight when horsepower is equal to zero. Again, this is not realistic in practice, but together with the interaction term, it determines how the slope changes at different hp levels. Formally, the marginal effect of weight on mpg is:

$$\frac{\partial mpg}{\partial wt} = -8.21662 + 0.02785 \cdot hp$$

  This means that for each additional unit of weight (1000 lbs), mpg decreases by about 8.22 miles per gallon, but this negative effect becomes less severe as horsepower increases.

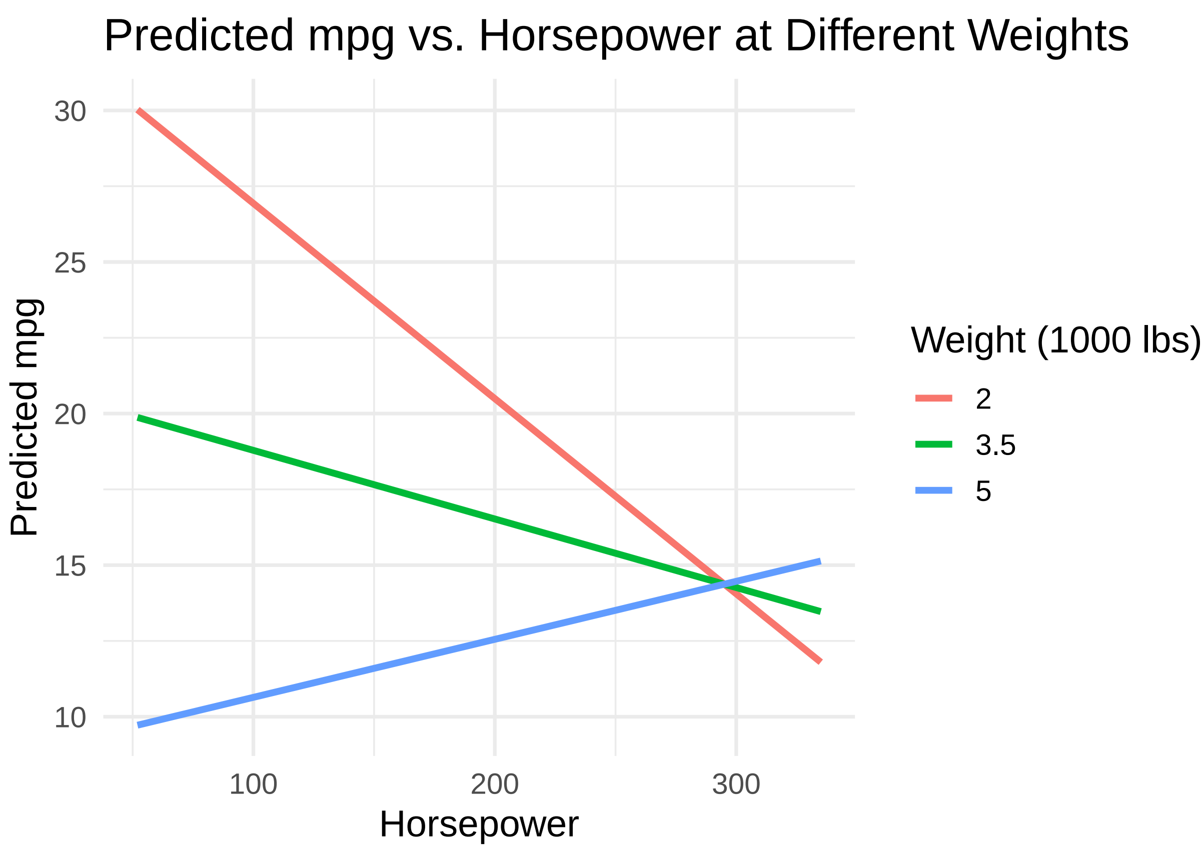
- **hp wt interaction coefficient (0.02785)**: The positive interaction means that the effect of horsepower depends on weight (and vice versa). Specifically:
  - For heavier cars, the negative effect of horsepower on mpg becomes *less severe*.
  - For more powerful cars, the negative effect of weight on mpg becomes *less severe*.

In other words, extra horsepower is more harmful for light cars than for heavy cars, while extra weight is more harmful for low-power cars than for high-power ones. To better understand this, we can plot the following graph:

```r
# Visualising the interaction effect

# Create a grid of values for predictions
newdata <- expand.grid(
  hp = seq(min(mtcars$hp), max(mtcars$hp), length.out = 100),
  wt = c(2, 3.5, 5) # choose three representative weights (1000 lbs units)
)

# Add predictions
```

```r
newdata$mpg_pred <- predict(model1, newdata)

# Plot
ggplot(newdata, aes(x = hp, y = mpg_pred, colour = factor(wt))) +
  geom_line(size = 1.2) +
  labs(
    title = "Predicted mpg vs. Horsepower at Different Weights",
    x = "Horsepower",
    y = "Predicted mpg",
    colour = "Weight (1000 lbs)"
  ) +
  theme_minimal(base_size = 14)
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



As you can see in the above, the effect of increasing horsepower for the lightest car (in red) is clearly leading to a bigger decrease in the predicted miles per gallon (mpg) than it is for a car slightly heavier (in green). Additionally, for a car much heavier, we can see that increasing power can even lead to an increase in the predicted mpg!

Part c) Model 2 with mean-centered variables

```r
# Create mean-centered variables
mtcars$hp_centered <- mtcars$hp - mean(mtcars$hp)
mtcars$wt_centered <- mtcars$wt - mean(mtcars$wt)

# Model 2: mpg ~ hp_centered + wt_centered + hp_centered*wt_centered
model2 <- lm(mpg ~ hp_centered + wt_centered + hp_centered*wt_centered, data = mtcars)

# Summary of stats of model2
summary(model2)
```

```
##
## Call:
## lm(formula = mpg ~ hp_centered + wt_centered + hp_centered *
##     wt_centered, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0632 -1.6491 -0.7362  1.4211  4.5513
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             18.898400   0.495703  38.124  < 2e-16 ***
## hp_centered             -0.030508   0.007503  -4.066 0.000352 ***
## wt_centered             -4.131649   0.529558  -7.802 1.69e-08 ***
## hp_centered:wt_centered  0.027848   0.007420   3.753 0.000811 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.153 on 28 degrees of freedom
## Multiple R-squared:  0.8848, Adjusted R-squared:  0.8724
## F-statistic: 71.66 on 3 and 28 DF,  p-value: 2.981e-13
```

```r
cat(sprintf("Model %d's equation: %s\n", 2, eq_print(model2)))
```

```
## Model 2's equation: y_hat = 18.8984 - 0.0305 hp_centered - 4.1316 wt_centered + 0.0278
↪  hp_centered * wt_centered
```

Comparison of coefficients between Model 1 and Model 2:

The underlying objective in centering the values taken by hp and wt around their mean is that, now, the hp coefficient represents the effect of horsepower on mpg when weight is at its average value, which is a much more realistic model than model1. Similarly, the wt coefficient represents the effect of weight on mpg when horsepower is at its mean value.

- The **interaction coefficient** remains exactly the same (0.02784815) in both models. This is a key property of mean-centering: it preserves the interaction coefficient (see question e).
- The **main effect coefficients** change significantly:
  - hp coefficient: -0.12010 (Model 1) vs -0.03051 (Model 2)
  - wt coefficient: -8.21662 (Model 1) vs -4.13165 (Model 2)

- The **intercept** changes from 49.80842 (Model 1) to 18.89840 (Model 2)

Key insight: The main effect coefficients in Model 2 represent the marginal effects at the mean values of the other variable. For example, the hp coefficient in Model 2 (-0.03051) is the effect of horsepower when weight equals its mean (3.21725), which can be verified by calculating the marginal effect from Model 1: $-0.12010 + 0.02785 \times 3.21725 = -0.03051$.

R-squared Analysis for Models 1 and 2: Both models have identical R-squared values (≈ 0.8848) and adjusted R-squared values (≈ 0.8724), which confirms that mean-centering does not change the model's explanatory power. This is expected because: - Mean-centering is a linear transformation that preserves the linear relationships - The interaction coefficient remains unchanged - The model fit is mathematically equivalent, just with different coefficient interpretations

Part d) Model 3 with only interaction term

```r
# Model 3: mpg ~ hp_centered:wt_centered (only interaction, no main effects)
model3 <- lm(mpg ~ hp_centered:wt_centered, data = mtcars)
summary(model3)
```

```
##
## Call:
## lm(formula = mpg ~ hp_centered:wt_centered, data = mtcars)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.3892  -4.0066  -0.3153   2.8009  12.7062
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)             19.41692    1.39189   13.95 1.19e-14 ***
## hp_centered:wt_centered  0.01574    0.02072    0.76    0.453
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.068 on 30 degrees of freedom
## Multiple R-squared:  0.01887,    Adjusted R-squared:  -0.01383
## F-statistic: 0.5771 on 1 and 30 DF,  p-value: 0.4534
```

```r
cat(sprintf("Model %d's equation: %s\n", 3, eq_print(model3)))
```

```
## Model 3's equation: y_hat = 19.4169 + 0.0157 hp_centered * wt_centered
```

Comparison of coefficients across all three models:

Model 3 results

- **Intercept:** 19.41692

- **hp_centered wt_centered:** 0.01574

```r
# Create a comparison table of R-squared values
r_squared_comparison <- data.frame(
  Model = c("Model 1", "Model 2", "Model 3"),
  R_squared = c(summary(model1)$r.squared, summary(model2)$r.squared,
  ↪  summary(model3)$r.squared),
  Adj_R_squared = c(summary(model1)$adj.r.squared, summary(model2)$adj.r.squared,
  ↪  summary(model3)$adj.r.squared),
  F_statistic = c(summary(model1)$fstatistic[1], summary(model2)$fstatistic[1],
  ↪  summary(model3)$fstatistic[1]),
  P_value = c(pf(summary(model1)$fstatistic[1], summary(model1)$fstatistic[2],
  ↪  summary(model1)$fstatistic[3], lower.tail = FALSE),
              pf(summary(model2)$fstatistic[1], summary(model2)$fstatistic[2],
              ↪  summary(model2)$fstatistic[3], lower.tail = FALSE),
              pf(summary(model3)$fstatistic[1], summary(model3)$fstatistic[2],
              ↪  summary(model3)$fstatistic[3], lower.tail = FALSE))
)

print(r_squared_comparison)
```

R-squared Comparison Across All Models

```
##     Model  R_squared Adj_R_squared F_statistic      P_value
## 1 Model 1 0.88476371    0.87241697  71.6596724 2.981394e-13
## 2 Model 2 0.88476371    0.87241697  71.6596724 2.981394e-13
## 3 Model 3 0.01887209   -0.01383218   0.5770528 4.533982e-01
```

Key R-squared Observations:

1. **Models 1 and 2 are mathematically equivalent::** Both achieve $R^2 \approx 0.8848$ and adjusted $R^2 \approx 0.8848$, confirming that mean-centering preserves model fit. They produce identical predictions despite different coefficient interpretations.

2. **Model 3 shows dramatic decline:** R^2 drops to ≈ 0.0189 (only 1.89% of variance explained), demonstrating the critical importance of including main effects.

3. **Model 2 coefficients represent marginal effects at means:** The hp coefficient in Model 2 exactly equals the marginal effect of hp when wt equals its mean, calculated from Model 1.

4. **Statistical significance:**

   - Models 1 and 2: Highly significant ($p < 0.001$)
   - Model 3: Not statistically significant ($p \approx 0.454$)

5. **Practical interpretation:** The interaction alone explains almost nothing about mpg variation, while the full model (with main effects) explains nearly 88% of the variance in mpg.

6. **Main effects are crucial:** Removing them causes an 86+ percentage point drop in R^2, demonstrating that individual effects of hp and wt are far more important than their interaction alone.

Comparison with Models 1 and 2

- **Interaction coefficient:** The interaction term changes significantly across models:
  - Model 1: 0.02784815
  - Model 2: 0.02784815 (identical to Model 1)
  - Model 3: 0.01573639 (significantly different!)

  This shows that **removing main effects DOES impact the interaction coefficient**, contrary to what one might expect. This is because the interaction term in Model 3 is forced to capture both the true interaction effect AND compensate for the missing main effects.

- **Main effects:**
  - Models 1 and 2 include main effects for hp and wt. In Model 1 these are defined at unrealistic baselines (hp = 0, wt = 0), while in Model 2 they are defined at realistic baselines (mean hp, mean wt).
  - Model 3 excludes the main effects entirely, meaning the model assumes hp and wt only affect mpg through their joint interaction. This is an oversimplification that costs the R-squared value to drop drastically, barely reaching 2% (i.e. 2% of the variations in mpg can be explained by the joint interaction of horsepower and weight).

- **Intercept:**
  - Model 1: 49.80842 (predicted mpg when hp = 0 and wt = 0, not meaningful).
  - Model 2: 18.89840 (predicted mpg at mean hp and mean wt, interpretable).
  - Model 3: 19.41692 (close to Model 2, but adjusted upward because the main effects are missing).

Interpretation Model 3 preserves the interaction effect but ignores the independent contributions of horsepower and weight. While its intercept represents predicted mpg at average hp and wt (similar to Model 2), excluding main effects makes it a weaker and less realistic specification. Importantly, the interaction coefficient in Model 3 (0.01574) is different from Models 1 and 2 (0.02785), indicating that the interaction term is absorbing some of the effect that should be attributed to the main effects.

Part e) Mathematical explanation

See the following:

Model 1:

$$mpg = \beta_0 + \beta_1 h_p + \beta_2 w_t + \beta_3 (h_p \cdot w_t) + \epsilon$$

Model 2:
Let ($h_{pc} = hp - \bar{h_p}$) and ($w_{tc} = wt - \bar{w_t}$) be the mean-centered variables.

$$mpg = \beta_0 + \beta_1 h_{pc} + \beta_2 w_{tc} + \beta_3 (h_{pc} w_{tc}) + \epsilon$$

$$mpg = \beta_0 + \beta_1 (hp - \bar{h_p}) + \beta_2 (wt - \bar{w_t}) + \beta_3 ((hp - \bar{h_p})(wt - \bar{w_t})) + \epsilon$$

$$mpg = \beta_0 + \beta_1(h_p - \bar{h_p}) + \beta_2(w_t - \bar{w_t}) + \beta_3(h_p - \bar{h_p})(w_t - \bar{w_t}) + \epsilon$$

$$mpg = h_p(\beta_1 - \beta_3\bar{w_t}) + w_t(\beta_2 - \beta_3\bar{h_p}) + \beta_0 - \beta_1\bar{h_p} - \beta_2\bar{w_t} + \beta_3\bar{h_p}\bar{w_t} + \epsilon$$

(2)

Let's match coefficients of model 1 with model2 . Let's set betas in (1) to alphas instead and keep the current notation for equation (2).

$$\begin{aligned}
\alpha_0 &= \beta_0 - \beta_1\bar{h_p} - \beta_2\bar{w_t} + \beta_3\bar{h_p}\bar{w_t} + \epsilon \\
\alpha_1 &= \beta_1 - \beta_3\bar{w_t} \\
\alpha_2 &= \beta_2 - \beta_3\bar{h_p} \\
\alpha_3 &= \beta_3
\end{aligned}$$

Then we solve the above for the betas in terms of alphas:

$$\begin{aligned}
\beta_0 &= \alpha_0 + \alpha_1\bar{h_p} + \alpha_2\bar{w_t} + \alpha_3\bar{h_p}\bar{w_t} - \epsilon \\
\beta_1 &= \alpha_1 + \alpha_3\bar{w_t} \\
\beta_2 &= \alpha_2 + \alpha_3\bar{h_p} \\
\beta_3 &= \alpha_3
\end{aligned}$$

By identification we see that the coefficient of the interaction term remains the exact same from model 1 to model 2. $\alpha_3 = \beta_3$ with α_3 and β_3 being the exact same value. For the rest of the coefficients we simply observe that there is a change of basis, which is why we get the same R^2 value.

Model 3:
(3)

$$mpg = \beta_0 + \beta_3 h_p c w_t c + \epsilon$$

$$mpg = \beta_0 + \beta_3(h_p - \bar{h_p})(w_t - \bar{w_t}) + \epsilon$$

$$mpg = \beta_0 + (\beta_3 h_p - \beta_3\bar{w_t})(w_t - \bar{w_t}) + \epsilon$$

$$mpg = \beta_0 + \beta_3 h_p w_t - \beta_3 h_p\bar{w_t} - \beta_3\bar{h_p}w_t + \beta_3\bar{h_p}\bar{w_t} + \epsilon$$

As we can see, the model 3 only contains 2 free parameters β_0 and β_3, lowering the overall flexibility and complexity of the model and decreasing R^2 value drastically.