Homework 7

Charles Ancel

10/19/2023

Homework Instructions

Make sure to add your name to the header of the document. When submitting the assignment on Gradescope, be sure to assign the appropriate pages of your submission to each Exercise.

The point value for each exercise is noted in the exercise title.

For questions that require code, please create or use the code chunk directly below the question and type your code there. Your knitted pdf will then show both the code and the output, so that we can assess your understanding and award any partial credit.

For written questions, please provide your answer after the indicated *Answer* prompt.

You are encouraged to knit your file as you work, to check that your coding and formatting are done so appropriately. This will also help you identify and locate any errors more easily.

Homework Setup

We'll use the following packages for this homework assignment. We'll also read in data from a csv file. To access the data, you'll want to download the dataset from Canvas and place it in the same folder as this R Markdown document. You'll then be able to use the following code to load in the data.

```
library(ggplot2)
library(MASS)
library(leaps)
```

Exercise 1: Formatting [5 points]

The first five points of the assignment will be earned for properly formatting your final document. Check that you have:

- included your name on the document
- properly assigned pages to exercises on Gradescope
- selected **page 1 (with your name)** and this page for this exercise (Exercise 1)
- all code is printed and readable for each question
- all output is printed
- generated a pdf file

Exercise 2: A First Chick-fil-A Search [15 points]

For the first half of this assignment, we will analyze the nutritional value of menu items from Chick-fil-A, a fast food restaurant specializing in chicken sandwiches. This data is contained in the chickfila.csv file on Canvas.

We'll be interested in fitting a model to predict the Calories of a new menu item from the other nutritional characteristics of that menu item.

part a

Read in the chickfila.csv data file.

```
chickfila_data <- read.csv("chickfila.csv")
head(chickfila_data)
```

```
##   Calories Fat SatFat TransFat Cholesterol Sodium Carbs Fiber Sugar Protein
## 1      460  23    8.0      0.0          45   1510    45     2     6      19
## 2      360  13    4.0      0.0          60   1050    41     2     8      19
## 3      290   8    3.5      0.0          60    980    30     1     2      26
## 4      700  40   12.0      0.5         415   1750    51     3     2      34
## 5      470  30    9.0      0.0         415   1340    19     2     2      29
## 6      420  23   11.0      0.0         180   1290    38     2     4      15
##   Serving
## 1     153
## 2     127
## 3     172
## 4     302
## 5     233
## 6     145
```

part b

How many models predicting the number of Calories in a menu item are possible from this dataset? (Consider only first-order terms, which means include all of the variables once and exactly as they appear in the dataset.)

```
n <- ncol(chickfila_data) - 1

total_models <- 2^n - 1
total_models
```

```
## [1] 1023
```

Answer: There are $2^n - 1$ possible models that can be constructed to predict the number of Calories in a menu item from this dataset when considering only first-order terms, where n is the number of predictor variables. For the given dataset, this totals 1,023 possible models.

part c

We'll perform model selection in this exercise "by hand". That means you should not use the `step` function in `R` for this exercise; if you do, you will not receive credit. We will use a backward searching process and will use the coefficient p-values to determine which variables to remove from the model, with an α of **0.01**.

Show the starting model and any subsequent models fit during your searching process here.

```
alpha <- 0.01
current_formula <- Calories ~ .
```

```r
model <- lm(current_formula, data = chickfila_data)
cat("Starting Model:\n")
```

```
## Starting Model:
```

```r
print(summary(model))
```

```
## 
## Call:
## lm(formula = current_formula, data = chickfila_data)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -75.784  -2.973   0.569   4.505  53.618 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  1.558183   1.009078   1.544 0.123682    
## Fat          8.670134   0.065023 133.340  < 2e-16 ***
## SatFat       0.536353   0.152247   3.523 0.000499 ***
## TransFat    -1.866273   3.691147  -0.506 0.613531    
## Cholesterol  0.012125   0.009656   1.256 0.210291    
## Sodium       0.005877   0.002180   2.696 0.007445 ** 
## Carbs        3.894552   0.065406  59.544  < 2e-16 ***
## Fiber        0.775513   0.164901   4.703 4.04e-06 ***
## Sugar       -0.105815   0.070887  -1.493 0.136640    
## Protein      3.790689   0.104167  36.391  < 2e-16 ***
## Serving     -0.005730   0.001387  -4.131 4.77e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 11.46 on 279 degrees of freedom
## Multiple R-squared:  0.9998, Adjusted R-squared:  0.9998 
## F-statistic: 1.291e+05 on 10 and 279 DF,  p-value: < 2.2e-16
```

```r
cat("\n\n")
```

```r
while (TRUE) {
    predictors <- names(summary(model)$coefficients)[-1]
    pvals <- summary(model)$coefficients[predictors, 4]

    if (all(pvals <= alpha)) {
        cat("Final Model:\n")
        print(summary(model))
        cat("\n\n")
        break
    }

    predictor_to_remove <- names(which.max(pvals))

    predictors <- setdiff(predictors, predictor_to_remove)

    current_formula <- as.formula(paste("Calories ~", paste(predictors, collapse=" + ")))

    model <- lm(current_formula, data = chickfila_data)
    cat("After removing", predictor_to_remove, ":\n")
    print(summary(model))
    cat("\n\n")
}
```

```
## Final Model:
##
## Call:
## lm(formula = current_formula, data = chickfila_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -75.784  -2.973   0.569   4.505  53.618
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.558183   1.009078   1.544 0.123682
## Fat          8.670134   0.065023 133.340  < 2e-16 ***
## SatFat       0.536353   0.152247   3.523 0.000499 ***
## TransFat    -1.866273   3.691147  -0.506 0.613531
## Cholesterol  0.012125   0.009656   1.256 0.210291
## Sodium       0.005877   0.002180   2.696 0.007445 **
## Carbs        3.894552   0.065406  59.544  < 2e-16 ***
## Fiber        0.775513   0.164901   4.703 4.04e-06 ***
## Sugar       -0.105815   0.070887  -1.493 0.136640
## Protein      3.790689   0.104167  36.391  < 2e-16 ***
## Serving     -0.005730   0.001387  -4.131 4.77e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.46 on 279 degrees of freedom
## Multiple R-squared:  0.9998, Adjusted R-squared:  0.9998
## F-statistic: 1.291e+05 on 10 and 279 DF,  p-value: < 2.2e-16
```

part d

Report the predictor variables included in your selected model from part c.

Answer: The predictor variables included in the selected model are: `Fat`, `SatFat`, `Cholesterol`, `Sodium`, `Carbs`, `Fiber`, `Sugar`, `Protein`, and `Serving`.

part e

What was the first variable removed from the model?

Answer: The first (and only) variable removed from the model was `TransFat`.

Exercise 3: Systematic Chick-fil-A Searching Methods [20 points]

Now, we'll consider a more systematic way to select a good model to predict the Calories of a menu item at Chick-fil-A. For this exercise, we'll use the BIC as our selection metric.

part a

Perform model selection, using BIC as the metric and backward searching.

Report the predictor variables selected for the final model. No need to report the fitted coefficients.

```
full_model <- lm(Calories ~ ., data = chickfila_data)
current_model <- full_model
lowest_BIC <- BIC(current_model)

continue_searching <- TRUE

while (continue_searching) {
    continue_searching <- FALSE
    all_predictors <- names(coef(current_model))[-1]
    best_BIC <- lowest_BIC
    best_model <- current_model

    for (predictor in all_predictors) {
        reduced_formula <- as.formula(paste("Calories ~", paste(setdiff(all_predictors, predictor), coll
        reduced_model <- lm(reduced_formula, data = chickfila_data)
        reduced_BIC <- BIC(reduced_model)

        if (reduced_BIC < best_BIC) {
            best_BIC <- reduced_BIC
            best_model <- reduced_model
        }
    }

    if (best_BIC < lowest_BIC) {
        current_model <- best_model
        lowest_BIC <- best_BIC
        continue_searching <- TRUE
    }
}

final_predictors <- names(coef(current_model))[-1]
final_predictors
```

```
## [1] "Fat"     "SatFat"  "Sodium"  "Carbs"   "Fiber"   "Protein" "Serving"
```

```
backward_model <- current_model
```

Answer: The predictor variables selected for the final model are: `Fat`, `SatFat`, `Sodium`, `Carbs`, `Fiber`, `Protein`, and `Serving`.

part b

Perform model selection, using BIC as the metric and forward searching.

Report the predictor variables selected for the model after the first step and for the final model. No need to report the fitted coefficients.

```
null_model <- lm(Calories ~ 1, data = chickfila_data)
current_model <- null_model
lowest_BIC <- BIC(current_model)

all_predictors <- names(chickfila_data)[-1]
```

```
used_predictors <- c()

continue_searching <- TRUE
first_step_predictors <- NULL

while (continue_searching) {
    continue_searching <- FALSE
    best_BIC <- lowest_BIC
    best_model <- current_model

    for (predictor in setdiff(all_predictors, used_predictors)) {
        extended_formula <- as.formula(paste("Calories ~", paste(c(used_predictors, predictor), collapse
        extended_model <- lm(extended_formula, data = chickfila_data)
        extended_BIC <- BIC(extended_model)


        if (extended_BIC < best_BIC) {
            best_BIC <- extended_BIC
            best_model <- extended_model
        }
    }
    if (best_BIC < lowest_BIC) {
        current_model <- best_model
        lowest_BIC <- best_BIC
        continue_searching <- TRUE

        used_predictors <- names(coef(current_model))[-1]

        if (is.null(first_step_predictors)) {
            first_step_predictors <- used_predictors
        }
    }
}

list(first_step = first_step_predictors, final_model = used_predictors)
```

```
## $first_step
## [1] "Fat"
##
## $final_model
## [1] "Fat"     "Carbs"   "Protein" "Sugar"   "Serving" "SatFat"  "Fiber"
## [8] "Sodium"
```

```
forward_model <- current_model
```

Answer: - After the first step of forward selection, the predictor variable selected for the model is: `Fat`. - The predictor variables selected for the final model are: `Fat`, `Carbs`, `Protein`, `Sugar`, `Serving`, `SatFat`, `Fiber`, and `Sodium`.

part c

Perform model selection, using BIC as the metric and stepwise searching starting from the intercept-only model.

Report the predictor variables selected for the final model. No need to report the fitted coefficients.

```
start_model <- lm(Calories ~ 1, data = chickfila_data)

stepwise_model <- step(start_model, direction = "both", scope = list(lower = ~1, upper = ~Fat+SatFat+Tra

final_predictors_stepwise <- names(coef(stepwise_model))[-1]
final_predictors_stepwise
```

```
## [1] "Fat"     "Carbs"   "Protein" "Serving" "SatFat"  "Fiber"   "Sodium"
```

Answer: The predictor variables selected for the final model are: `Fat`, `Carbs`, `Protein`, `Serving`, `SatFat`, `Fiber`, and `Sodium`.

part d

First, do you select the same models using backward, forward, and stepwise searching?

Then, report the BIC for the final model(s) selected with the three searching methods. Based on the BIC, which model would you select overall?

```
# 1. Compare models
same_model_backward_forward <- isTRUE(all.equal(coef(backward_model), coef(forward_model)))
same_model_forward_stepwise <- isTRUE(all.equal(coef(forward_model), coef(stepwise_model)))
same_model_backward_stepwise <- isTRUE(all.equal(coef(backward_model), coef(stepwise_model)))

all_models_same <- same_model_backward_forward && same_model_forward_stepwise && same_model_backward_ste

# 2. Compute BIC for each model
bic_backward <- BIC(backward_model)
bic_forward <- BIC(forward_model)
bic_stepwise <- BIC(stepwise_model)

# 3. Decide which model to select based on BIC
lowest_bic <- min(bic_backward, bic_forward, bic_stepwise)
selected_model <- ifelse(lowest_bic == bic_backward, "backward",
                        ifelse(lowest_bic == bic_forward, "forward", "stepwise"))

list(same_model = all_models_same, BICs = c(backward = bic_backward, forward = bic_forward, stepwise = 
```

```
## $same_model
## [1] FALSE
##
## $BICs
## backward  forward stepwise
## 2281.672 2284.797 2281.672
##
## $selected_model
## [1] "stepwise"
```

Answer:

1. Do the backward, forward, and stepwise searching methods select the same models?
    - No, they do not select the exact same models.
2. BIC for the final models selected with the three searching methods are:
    - Backward searching: 2281.672
    - Forward searching: 2284.797
    - Stepwise searching: 2281.672
3. Based on the BIC, which model would you select overall?

- We would select either the model from backward searching or the model from stepwise searching, as they both have the lowest BIC value of 2281.672. Lower BIC values indicate better model fit. In this case, since the BIC values are the same for backward and stepwise, either model could be chosen.

Exercise 4: Comparing Chick-Fil-A Model Metrics [30 points]

We aren't sure if any of our searching methods from Exercise 3 identify the overall optimal model. In this exercise, we'll use exhaustive searching to identify the best optimal model of all possible models.

part a

First, run the exhaustive searching function. By default, the exhaustive searching function only selects models with up to 8 predictors included. Since we have more than 8 possible predictor variables, use the `nvmax` argument to specify the number of possible predictors to consider (we'd like to allow all predictors to be included). What metric is used to determine the optimal model at each p?

```
exhaustive_search <- regsubsets(Calories ~ Fat + SatFat + TransFat + Cholesterol + Sodium + Carbs + Fibe

summary(exhaustive_search)
```

```
## Subset selection object
## Call: regsubsets.formula(Calories ~ Fat + SatFat + TransFat + Cholesterol +
##     Sodium + Carbs + Fiber + Sugar + Protein + Serving, data = chickfila_data,
##     nvmax = 10, method = "exhaustive")
## 10 Variables  (and intercept)
##             Forced in Forced out
## Fat             FALSE      FALSE
## SatFat          FALSE      FALSE
## TransFat        FALSE      FALSE
## Cholesterol     FALSE      FALSE
## Sodium          FALSE      FALSE
## Carbs           FALSE      FALSE
## Fiber           FALSE      FALSE
## Sugar           FALSE      FALSE
## Protein         FALSE      FALSE
## Serving         FALSE      FALSE
## 1 subsets of each size up to 10
## Selection Algorithm: exhaustive
##           Fat SatFat TransFat Cholesterol Sodium Carbs Fiber Sugar Protein
## 1  ( 1 )  "*" " "    " "      " "         " "    " "   " "   " "   " "
## 2  ( 1 )  "*" " "    " "      " "         " "    "*"   " "   " "   " "
## 3  ( 1 )  "*" " "    " "      " "         " "    "*"   " "   " "   "*"
## 4  ( 1 )  "*" " "    " "      " "         " "    "*"   " "   "*"   "*"
## 5  ( 1 )  "*" "*"    " "      " "         " "    "*"   "*"   " "   "*"
## 6  ( 1 )  "*" "*"    " "      " "         " "    "*"   "*"   " "   "*"
## 7  ( 1 )  "*" "*"    " "      " "         "*"    "*"   "*"   " "   "*"
## 8  ( 1 )  "*" "*"    " "      " "         "*"    "*"   "*"   "*"   "*"
## 9  ( 1 )  "*" "*"    " "      "*"         "*"    "*"   "*"   "*"   "*"
## 10  ( 1 ) "*" "*"    "*"      "*"         "*"    "*"   "*"   "*"   "*"
##           Serving
## 1  ( 1 )  " "
## 2  ( 1 )  " "
## 3  ( 1 )  " "
## 4  ( 1 )  " "
## 5  ( 1 )  " "
## 6  ( 1 )  "*"
## 7  ( 1 )  "*"
## 8  ( 1 )  "*"
## 9  ( 1 )  "*"
## 10  ( 1 ) "*"
```

Answer:

The metric used by the `regsubsets()` function to determine the optimal model at each p (number of predictors) is the R^2 value, as stated earlier.

From the results:

- The best model with 1 predictor includes `Fat`.
- The best model with 2 predictors includes `Fat` and `Carbs`.
- The best model with 3 predictors includes `Fat`, `Carbs`, and `Protein`.
- The best model with 4 predictors includes `Fat`, `Carbs`, `Protein`, and `Sugar`.
- The best model with 5 predictors includes `Fat`, `SatFat`, `Carbs`, `Protein`, and `Fiber`.
- The best model with 6 predictors includes `Fat`, `SatFat`, `Carbs`, `Protein`, `Fiber`, and `Serving`.
- The best model with 7 predictors includes `Fat`, `SatFat`, `Sodium`, `Carbs`, `Protein`, `Fiber`, and `Serving`.
- The best model with 8 predictors includes `Fat`, `SatFat`, `Sodium`, `Carbs`, `Protein`, `Fiber`, `Sugar`, and `Serving`.
- The best model with 9 predictors includes `Fat`, `SatFat`, `Cholesterol`, `Sodium`, `Carbs`, `Protein`, `Fiber`, `Sugar`, and `Serving`.
- The model with all 10 predictors includes all the predictor variables.

part b

Do the optimal models at each p result in nested models for the Chick-fil-A data? What variables are included in the optimal model with 3 predictor variables?

Answer: 1. **Are the optimal models nested?** - Yes, the optimal models at each p result in nested models for the Chick-fil-A data. Nested models mean that the predictors in the model with p predictors are a subset of the predictors in the model with $p+1$ predictors. In the provided results, we can see that as we move from one level of complexity to the next (from 1 predictor to 2 predictors, from 2 predictors to 3 predictors, and so on), the predictors from the previous model are retained, and a new predictor is added. Thus, the models are nested.

2. **What variables are included in the optimal model with 3 predictor variables?**
    - The optimal model with 3 predictor variables includes `Fat`, `Carbs`, and `Protein`.

part c

Calculate the AIC for each of the models identified in part a. Based on AIC, which predictor variables should be included in the optimal model?

```r
calculate_AIC <- function(formula_string) {
  model <- lm(formula_string, data = chickfila_data)
  return(AIC(model))
}
formulas <- c(
  "Calories ~ Fat",
  "Calories ~ Fat + Carbs",
  "Calories ~ Fat + Carbs + Protein",
  "Calories ~ Fat + Carbs + Protein + Sugar",
  "Calories ~ Fat + SatFat + Carbs + Protein + Fiber",
  "Calories ~ Fat + SatFat + Carbs + Protein + Fiber + Serving",
  "Calories ~ Fat + SatFat + Sodium + Carbs + Protein + Fiber + Serving",
  "Calories ~ Fat + SatFat + Sodium + Carbs + Protein + Fiber + Sugar + Serving",
  "Calories ~ Fat + SatFat + Cholesterol + Sodium + Carbs + Protein + Fiber + Sugar + Serving",
  "Calories ~ Fat + SatFat + TransFat + Cholesterol + Sodium + Carbs + Protein + Fiber + Sugar + Servin
)

AIC_values <- sapply(formulas, calculate_AIC)

min_AIC_formula <- formulas[which.min(AIC_values)]

min_AIC_formula
```

```
## [1] "Calories ~ Fat + SatFat + Sodium + Carbs + Protein + Fiber + Sugar + Serving"
```

Answer: Based on the AIC criterion, the optimal model for predicting Calories includes the following predictor variables:

- `Fat`
- `SatFat`
- `Sodium`
- `Carbs`
- `Protein`
- `Fiber`
- `Sugar`
- `Serving`

This model offers the best trade-off between goodness-of-fit and model complexity among the models considered.

part d

Calculate the BIC for each of the models identified in part a. Based on BIC, which predictor variables should be included in the optimal model?

```
calculate_BIC <- function(formula_string) {
  model <- lm(formula_string, data = chickfila_data)
  return(BIC(model))
}

BIC_values <- sapply(formulas, calculate_BIC)

min_BIC_formula <- formulas[which.min(BIC_values)]

min_BIC_formula
```

```
## [1] "Calories ~ Fat + SatFat + Sodium + Carbs + Protein + Fiber + Serving"
```

Answer: Based on the BIC criterion, the optimal model for predicting Calories includes the following predictor variables:

- `Fat`
- `SatFat`
- `Sodium`
- `Carbs`
- `Protein`
- `Fiber`

- Serving

This model provides the best balance between goodness-of-fit and model complexity, as per the BIC metric. It's noteworthy that the BIC criterion selected a slightly simpler model than the AIC by excluding the Sugar variable. BIC tends to favor simpler models compared to AIC, especially when sample sizes are large.

part e

Calculate the adjusted R^2 for each of the models identified in part a. Based on the adjusted R^2, which predictor variables should be included in the optimal model?

```
calculate_adj_R2 <- function(formula_string) {
  model <- lm(formula_string, data = chickfila_data)
  return(summary(model)$adj.r.squared)
}

adj_R2_values <- sapply(formulas, calculate_adj_R2)

max_adj_R2_formula <- formulas[which.max(adj_R2_values)]

max_adj_R2_formula
```

```
## [1] "Calories ~ Fat + SatFat + Cholesterol + Sodium + Carbs + Protein + Fiber + Sugar + Serving"
```

Answer: Based on the adjusted R^2 criterion, the optimal model for predicting Calories includes the following predictor variables:

- Fat
- SatFat
- Cholesterol
- Sodium
- Carbs
- Protein
- Fiber
- Sugar
- Serving

This model explains the highest proportion of variance in the response variable (Calories) after adjusting for the number of predictors in the model, as per the adjusted R^2 metric. This suggests that including almost all the available predictor variables (except for TransFat) results in the best model fit when considering the adjusted R^2.

part f

Calculate the RMSE for each of the models identified in part a. Based on the RMSE, which predictor variables should be included in the optimal model?

```
calculate_RMSE <- function(formula_string) {
  model <- lm(formula_string, data = chickfila_data)
  predictions <- predict(model, chickfila_data)
  return(sqrt(mean((predictions - chickfila_data$Calories)^2)))
}

RMSE_values <- sapply(formulas, calculate_RMSE)

min_RMSE_formula <- formulas[which.min(RMSE_values)]
```

```
min_RMSE_formula
```

```
## [1] "Calories ~ Fat + SatFat + TransFat + Cholesterol + Sodium + Carbs + Protein + Fiber + Sugar + S
```

Answer: Based on the RMSE criterion, the optimal model for predicting Calories includes all of the available predictor variables:

- `Fat`
- `SatFat`
- `TransFat`
- `Cholesterol`
- `Sodium`
- `Carbs`
- `Protein`
- `Fiber`
- `Sugar`
- `Serving`

This means that when considering the RMSE, a model that uses all the predictor variables provides the most accurate predictions for Calories in the Chick-fil-A dataset.

part g

Are the same models selected for each of parts c through f? How many different models are selected from the different metrics but with the same exhaustive searching method? How do these selected models from exhaustive searching compare to the models selected from backwards, forwards, and stepwise searching in Exercise 3?

Answer: ### Selected Models:

From Exhaustive Searching:

1. **AIC:** `Calories ~ Fat + SatFat + Sodium + Carbs + Protein + Fiber + Sugar + Serving`
2. **BIC:** `Calories ~ Fat + SatFat + Sodium + Carbs + Protein + Fiber + Serving`
3. **Adjusted R^2:** `Calories ~ Fat + SatFat + Cholesterol + Sodium + Carbs + Protein + Fiber + Sugar + Serving`
4. **RMSE:** `Calories ~ Fat + SatFat + TransFat + Cholesterol + Sodium + Carbs + Protein + Fiber + Sugar + Serving`

From Backwards, Forwards, and Stepwise Searching (Exercise 3):

- **Backward Searching:** `Fat, SatFat, Sodium, Carbs, Fiber, Protein, Serving`
- **Forward Searching:** `Fat, Carbs, Protein, Sugar, Serving, SatFat, Fiber, Sodium`
- **Stepwise Searching:** `Fat, Carbs, Protein, Serving, SatFat, Fiber, Sodium`

Analysis:

1. **Are the same models selected for each of parts c through f?**
   - No, the models selected using AIC, BIC, Adjusted R^2, and RMSE from exhaustive searching are different.
2. **How many different models are selected from the different metrics but with the same exhaustive searching method?**
   - Four different models are selected using the four metrics.
3. **How do these selected models from exhaustive searching compare to the models selected from backwards, forwards, and stepwise searching in Exercise 3?**
   - The models from exhaustive searching include a wider range of predictors compared to the models from backward, forward, and stepwise searching.

- The BIC model from exhaustive searching closely matches the models from backward and stepwise searching, but with an additional predictor.
- The RMSE model from exhaustive searching includes all the predictors, while none of the models from Exercise 3 included all the predictors.
- The Adjusted R^2 model from exhaustive searching has a more varied set of predictors compared to the models from Exercise 3.

part h

For which of the metrics used in parts c through f is the comparison of models unfair? In other words, which metric would you not want to use in this situation?

Answer: When comparing models, we need to be cautious about metrics that naturally favor models with more predictors, as they might lead to overfitting. Among the metrics used in parts c through f:

1. **AIC (Akaike Information Criterion)**: It penalizes models based on the number of predictors, but not as heavily as BIC. Thus, while AIC aims to find the model that best predicts future observations, it can sometimes favor slightly more complex models. However, in the context of model selection, it is a commonly accepted metric.

2. **BIC (Bayesian Information Criterion)**: It penalizes the number of parameters more heavily than AIC. This makes BIC favor simpler models compared to AIC. Like AIC, BIC is also commonly used for model selection.

3. **Adjusted R^2**: Unlike the regular R^2, the adjusted R^2 takes into account the number of predictors in the model. It increases only if the new predictor improves the model more than expected by chance. It can decrease when a non-informative predictor is added, making it more reliable than the regular R^2 for model selection.

4. **RMSE (Root Mean Square Error)**: RMSE measures the model's prediction error. A lower RMSE indicates a better fit to the data. However, RMSE doesn't account for model complexity. As more predictors are added to a model, RMSE can decrease (indicating a better fit) even if those predictors don't genuinely improve the model's predictive power. This can lead to overfitting, where the model fits the training data very well but performs poorly on new, unseen data.

Answer:

Out of the metrics mentioned, **RMSE** would be the most "unfair" in the sense that it does not penalize for model complexity. Without a penalization term, RMSE can favor more complex models that might not generalize well to new data. Hence, in situations where we're trying to select the best model from a set of candidates, especially when there's a risk of overfitting, RMSE might not be the best choice on its own.

Exercise 5: Understanding Cats [30 points]

Now, we'll turn to understanding the heart weights of cats. We can use the `cats` data, which is contained in the MASS package. The data includes variables on the sex, body weight (Bwt) and heart weight (Hwt) of 144 adult cats.

part a

Fit a linear model to predict the heart weight of a cat using sex and body weight as the predictor variables. Print a summary of the model.

```r
cat_model <- lm(Hwt ~ Bwt + Sex, data = cats)

summary(cat_model)
```

```
##
## Call:
## lm(formula = Hwt ~ Bwt + Sex, data = cats)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5833 -0.9700 -0.0948  1.0432  5.1016
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.4149     0.7273  -0.571    0.569
## Bwt           4.0758     0.2948  13.826   <2e-16 ***
## SexM         -0.0821     0.3040  -0.270    0.788
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.457 on 141 degrees of freedom
## Multiple R-squared:  0.6468, Adjusted R-squared:  0.6418
## F-statistic: 129.1 on 2 and 141 DF,  p-value: < 2.2e-16
```

part b

Write out one general fitted model for the heart weight of a cat. Then, write out two specific fitted models for cats based on their sex.

Answer: The general model to predict the heart weight (Hwt) of a cat based on its body weight (Bwt) and sex is:

$$Hwt = -0.4149 + 4.0758 \times Bwt - 0.0821 \times SexM$$

Here, $SexM$ is a binary variable where: - $SexM = 1$ if the cat is male - $SexM = 0$ if the cat is female

Specific Fitted Models Based on Sex:

1. **For Female Cats ($SexM = 0$):** Substituting $SexM = 0$ into the general model:

$$Hwt_{\text{female}} = -0.4149 + 4.0758 \times Bwt$$

2. **For Male Cats ($SexM = 1$):** Substituting $SexM = 1$ into the general model:

$$Hwt_{\text{male}} = (-0.4149 - 0.0821) + 4.0758 \times Bwt$$

$$Hwt_{\text{male}} = -0.4970 + 4.0758 \times Bwt$$

So, the two specific models are:

- For female cats: $Hwt_{\text{female}} = -0.4149 + 4.0758 \times Bwt$
- For male cats: $Hwt_{\text{male}} = -0.4970 + 4.0758 \times Bwt$

part c

Interpret the coefficients from this model.

Answer: 1. **Intercept (-0.4149):** This is the estimated heart weight (in grams) for a female cat with a body weight of 0 kg. Practically, a body weight of 0 kg doesn't make sense for a cat, but the intercept helps set a baseline for our predictions. It's more meaningful in relation to the other coefficients in the model than as a standalone value.

2. **Body Weight (Bwt) Coefficient (4.0758):** For each additional kilogram in body weight, a cat's heart weight is expected to increase by approximately 4.0758 grams, holding the sex constant. This means that, regardless of whether the cat is male or female, for every kilogram increase in its body weight, its heart weight tends to increase by about 4.0758 grams.
3. **Sex ($SexM$) Coefficient (-0.0821):** Male cats, on average, have heart weights that are 0.0821 grams less than female cats of the same body weight. However, it's important to note that this coefficient is not statistically significant (given its p-value of 0.788), suggesting that there might not be a meaningful difference in heart weights between male and female cats after accounting for body weight.

part d

Fit a second linear model to predict the heart weight of a cat using only sex (not the body weight) as the predictor variable. Print a summary of the model.

```r
model_sex_only <- lm(Hwt ~ Sex, data = cats)
summary(model_sex_only)
```

```
##
## Call:
## lm(formula = Hwt ~ Sex, data = cats)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8227 -1.7227  0.0273  1.2273  9.1773
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.2021     0.3251  28.308  < 2e-16 ***
## SexM          2.1206     0.3961   5.354 3.38e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.229 on 142 degrees of freedom
## Multiple R-squared:  0.168,  Adjusted R-squared:  0.1621
## F-statistic: 28.66 on 1 and 142 DF,  p-value: 3.38e-07
```

part e

For the remaining parts, you will use one of your models from parts a and d to answer the questions. Make sure that for each question you choose the appropriate model to address that question.

Using information from one of these two models, what is the sample mean heart weight for male cats?

Answer: Using the model from part d, the sample mean heart weight for male cats can be determined by the intercept plus the coefficient for `SexM`.

Given:

$$\text{Intercept} = 9.2021\text{Coefficient for } SexM = 2.1206$$

$$\text{Sample mean heart weight for male cats } = \text{Intercept+Coefficient for `}SexM\text{` } = 9.2021 + 2.1206 = 11.3227 grams$$

The sample mean heart weight for male cats is 11.3227 grams.

part f

Using information from one of these two models, what is the difference in the sample mean heart weights between male and female cats? Which type of cat has the larger heart weight, on average.

Answer: The difference in the sample mean heart weights between male and female cats is given by the coefficient for `SexM`, which is 2.1206 grams. This means that male cats, on average, have a heart weight that is 2.1206 grams more than female cats.

part g

Using information from one of these two models, is there a statistically significant difference in the mean heart weights between all male and all female cats? Explain, and be sure to include numerical support.

Answer: The p-value for the `SexM` coefficient in the model from part d is 3.38×10^{-7}, which is much less than 0.05. This indicates that there is a statistically significant difference in the mean heart weights between all male and all female cats.

part h

Using information from one of these two models, is there a statistically significant difference in the mean heart weights between all male and all female cats after controlling for the body weights of cats? Explain, and be sure to include numerical support.

Answer: No, after controlling for the body weights of cats, there isn't a statistically significant difference in the mean heart weights between male and female cats, as evidenced by the p-value of 0.788.

part i

Using information from one of these two models, what is the average difference between the heart weights of male and female cats, after controlling for the body weight of a cat?

Answer: From the model in part a, after controlling for the body weight of a cat, the average difference between the heart weights of male and female cats is given by the coefficient for `SexM`, which is -0.0821 grams. This suggests that, on average and after accounting for body weight, male cats have slightly smaller heart weights than female cats.

After controlling for the body weight of a cat, the average difference between the heart weights of male and female cats is -0.0821 grams.

part j

Using information from one of these two models, what is the proportion of the variability of the heart weights of cats that can be explained by the linear relationship with body weight and sex?

Answer: Using the model from part a, the proportion of the variability of the heart weights of cats that can be explained by the linear relationship with body weight and sex is given by the R^2 value. The R^2 value from the model is 0.6468 or 64.68%.

64.68% of the variability of the heart weights of cats can be explained by the linear relationship with body weight and sex.