B Code and Output

```
# ==================== 1. Data
# Exploration
# ====================
prostate <- read.table("~/Desktop/prostate.txt")

prostate <- prostate[, -1]
names(prostate) <- c("PSA", "cancer_volume",
    "weight", "age", "benign_prostatic_hyperplasia",
    "seminal_vesicle_invasion",
    "capsular_penetration", "gleason_score")

# Use pairwise scatter plot to
# distinguish quantitative
# variables and qualitative
# variables
pairs(prostate, pch = "*")

quantitative_vars <- c("PSA", "cancer_volume",
    "weight", "age", "benign_prostatic_hyperplasia",
    "capsular_penetration")
qualitative_vars <- c("seminal_vesicle_invasion",
    "gleason_score")
prostate$seminal_vesicle_invasion <-
as.factor(prostate$seminal_vesicle_invasion)
prostate$gleason_score <- as.factor(prostate$gleason_score)

# Draw the correlation matrix
# for the quantitatives
cor(prostate[quantitative_vars])
```

```
##                                       PSA
## PSA                            1.00000000
## cancer_volume                  0.62415059
## weight                         0.02621343
## age                            0.01719938
## benign_prostatic_hyperplasia -0.01648649
## capsular_penetration           0.55079252
##                               cancer_volume
## PSA                            0.624150588
## cancer_volume                  1.000000000
## weight                         0.005107148
## age                            0.039094423
## benign_prostatic_hyperplasia  -0.133209431
## capsular_penetration           0.692896688
##                                     weight
## PSA                           0.026213430
## cancer_volume                 0.005107148
## weight                        1.000000000
```

```
## age                           0.164323714
## benign_prostatic_hyperplasia 0.321848748
## capsular_penetration         0.001578905
##                                      age
## PSA                           0.01719938
## cancer_volume                 0.03909442
## weight                        0.16432371
## age                           1.00000000
## benign_prostatic_hyperplasia 0.36634121
## capsular_penetration         0.09955535
##                              benign_prostatic_hyperplasia
## PSA                                            -0.01648649
## cancer_volume                                  -0.13320943
## weight                                          0.32184875
## age                                             0.36634121
## benign_prostatic_hyperplasia                    1.00000000
## capsular_penetration                           -0.08300865
##                              capsular_penetration
## PSA                                     0.550792517
## cancer_volume                           0.692896688
## weight                                  0.001578905
## age                                     0.099555351
## benign_prostatic_hyperplasia           -0.083008649
## capsular_penetration                    1.000000000
```

```
# Plot histograms for each
# quantitative variables
par(mfrow = c(2, 3))
invisible(Map(function(x, y) hist(x,
    main = y), prostate[quantitative_vars],
    quantitative_vars))

# Use Box-Cox procedure to find
# the best transformation of
# the response variable
fit1 <- lm(PSA ~ ., data = prostate[quantitative_vars])
library(MASS)
par(mfrow = c(1, 1))
boxcox(fit1)

# Do a log transformation on
# the response variable
prostate$log_PSA <- log(prostate$PSA)
prostate <- prostate[c("log_PSA",
    "cancer_volume", "weight",
    "age", "benign_prostatic_hyperplasia",
    "seminal_vesicle_invasion",
    "capsular_penetration", "gleason_score")]
quantitative_vars <- c("log_PSA",
    "cancer_volume", "weight",
```

```r
    "age", "benign_prostatic_hyperplasia",
    "capsular_penetration")


# Plot a new pairwise scatter
# plot and histograms to see
# whether transformations are
# needed for predictors
pairs(prostate[!(names(prostate) %in%
    qualitative_vars)], pch = "*")

# From the scatter plot, it
# seems that the predictor
# cancer volume needs a log
# transformation
prostate$log_cancer_volume <- log(prostate$cancer_volume)
prostate <- prostate[c("log_PSA",
    "log_cancer_volume", "weight",
    "age", "benign_prostatic_hyperplasia",
    "seminal_vesicle_invasion",
    "capsular_penetration", "gleason_score")]
quantitative_vars <- c("log_PSA",
    "log_cancer_volume", "weight",
    "age", "benign_prostatic_hyperplasia",
    "capsular_penetration")

# Check multicollinearity
solve(cor(prostate[c("log_cancer_volume",
    "weight", "age", "benign_prostatic_hyperplasia",
    "capsular_penetration")]))
```

```
##                               log_cancer_volume
## log_cancer_volume                      1.69716394
## weight                                 0.06112201
## age                                   -0.27296326
## benign_prostatic_hyperplasia          -0.04594090
## capsular_penetration                  -1.02625401
##                                         weight
## log_cancer_volume                     0.06112201
## weight                                1.12144849
## age                                  -0.06615563
## benign_prostatic_hyperplasia         -0.34328385
## capsular_penetration                 -0.06147763
##                                             age
## log_cancer_volume                    -0.27296326
## weight                               -0.06615563
## age                                   1.22488972
## benign_prostatic_hyperplasia         -0.41982333
## capsular_penetration                  0.01211048
##                               benign_prostatic_hyperplasia
```

```
## log_cancer_volume                                -0.0459409
## weight                                           -0.3432838
## age                                              -0.4198233
## benign_prostatic_hyperplasia                      1.2800890
## capsular_penetration                              0.1770058
##                               capsular_penetration
## log_cancer_volume                       -1.02625401
## weight                                  -0.06147763
## age                                      0.01211048
## benign_prostatic_hyperplasia             0.17700579
## capsular_penetration                     1.64821553
```

```r
# Draw side-by-side boxplot to
# see whether the qualitative
# variables have effects on the
# response variable
par(mfrow = c(1, 2))
invisible(Map(function(x, y, z) boxplot(y ~
    x, main = z), prostate[qualitative_vars],
    data.frame(prostate$log_PSA,
        prostate$log_PSA), qualitative_vars))

par(mfrow = c(1, 1))
boxplot(prostate$log_PSA ~
prostate$seminal_vesicle_invasion:prostate$gleason_score,
    main = "log_PSA with seminal_vesicle_invasion and gleason_score")

# ================== 2. Model
# Selection
# ====================

# cross-validation split
set.seed(1)
train <- sample(nrow(prostate),
    70)
test <- setdiff(1:nrow(prostate),
    train)




# 2.1 First Order Model

fit2 <- lm(log_PSA ~ ., data = prostate,
    subset = train)
summary(fit2)
```

```
##
## Call:
## lm(formula = log_PSA ~ ., data = prostate, subset = train)
```

```
##
## Residuals:
##     Min      1Q  Median
## -1.4276 -0.5052  0.1234
##      3Q     Max
##  0.3451  1.3721
##
## Coefficients:
##                               Estimate
## (Intercept)                   2.364026
## log_cancer_volume             0.509867
## weight                        0.016241
## age                          -0.028436
## benign_prostatic_hyperplasia  0.066111
## seminal_vesicle_invasion1     0.907232
## capsular_penetration         -0.031721
## gleason_score7                0.144602
## gleason_score8                0.714372
##                              Std. Error
## (Intercept)                    0.676513
## log_cancer_volume              0.088720
## weight                         0.007795
## age                            0.011084
## benign_prostatic_hyperplasia   0.038260
## seminal_vesicle_invasion1      0.313160
## capsular_penetration           0.035328
## gleason_score7                 0.186805
## gleason_score8                 0.268939
##                              t value
## (Intercept)                    3.494
## log_cancer_volume              5.747
## weight                         2.084
## age                           -2.565
## benign_prostatic_hyperplasia   1.728
## seminal_vesicle_invasion1      2.897
## capsular_penetration          -0.898
## gleason_score7                 0.774
## gleason_score8                 2.656
##                              Pr(>|t|)
## (Intercept)                  0.000891
## log_cancer_volume             3.1e-07
## weight                       0.041394
## age                          0.012778
## benign_prostatic_hyperplasia 0.089063
## seminal_vesicle_invasion1    0.005225
## capsular_penetration         0.372760
## gleason_score7               0.441874
## gleason_score8               0.010069
##
## (Intercept)                  ***
```

```
## log_cancer_volume            ***
## weight                       *
## age                          *
## benign_prostatic_hyperplasia .
## seminal_vesicle_invasion1    **
## capsular_penetration
## gleason_score7
## gleason_score8               *
## ---
## Signif. codes:
##   0 '***' 0.001 '**' 0.01
##   '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6566 on 61 degrees of freedom
## Multiple R-squared:  0.7494, Adjusted R-squared:  0.7165
## F-statistic:  22.8 on 8 and 61 DF,  p-value: 1.161e-15

# Use forward stepwise and
# backward stepwise to find the
# best first-order model under
# AIC
model1aicf <- invisible(stepAIC(lm(log_PSA ~
    1, data = prostate, subset = train),
    scope = list(lower = lm(log_PSA ~
        1, data = prostate, subset = train),
        upper = lm(log_PSA ~ .,
            data = prostate, subset = train)),
    direction = "both"))

model1aicb <- invisible(stepAIC(lm(log_PSA ~
    ., data = prostate, subset = train),
    scope = list(lower = lm(log_PSA ~
        1, data = prostate, subset = train),
        upper = lm(log_PSA ~ .,
            data = prostate, subset = train)),
    direction = "both"))

# Use forward stepwise and
# backward stepwise to find the
# best first-order model under
# BIC
model1bicf <- invisible(stepAIC(lm(log_PSA ~
    1, data = prostate, subset = train),
    scope = list(lower = lm(log_PSA ~
        1, data = prostate, subset = train),
        upper = lm(log_PSA ~ .,
            data = prostate, subset = train)),
    direction = "both", k = log(length(train))))

model1bicb <- invisible(stepAIC(lm(log_PSA ~
```

```
    ., data = prostate, subset = train),
    scope = list(lower = lm(log_PSA ~
        1, data = prostate, subset = train),
        upper = lm(log_PSA ~ .,
            data = prostate, subset = train)),
    direction = "both", k = log(length(train))))

sapply(list(model1aicf = model1aicf,
    model1aicb = model1aicb, model1bicf = model1bicf,
    model1bicb = model1bicb), "[[",
    "call")

## $model1aicf
## lm(formula = log_PSA ~ log_cancer_volume + weight +
seminal_vesicle_invasion +
##     age + gleason_score + benign_prostatic_hyperplasia, data = prostate,
##     subset = train)
##
## $model1aicb
## lm(formula = log_PSA ~ log_cancer_volume + weight + age +
benign_prostatic_hyperplasia +
##     seminal_vesicle_invasion + gleason_score, data = prostate,
##     subset = train)
##
## $model1bicf
## lm(formula = log_PSA ~ log_cancer_volume + weight +
seminal_vesicle_invasion +
##     age, data = prostate, subset = train)
##
## $model1bicb
## lm(formula = log_PSA ~ log_cancer_volume + weight + age +
seminal_vesicle_invasion,
##     data = prostate, subset = train)

# 2.2 second order model

# Fit the second-order model
fit3 <- lm(log_PSA ~ .^2 + I(age^2) +
    I(log_cancer_volume^2) + I(weight^2) +
    I(benign_prostatic_hyperplasia^2) +
    I(capsular_penetration^2),
    data = prostate, subset = train)

# Use stepwise to find the best
# second-order model based on
# model1aic under AIC
model2aicaic <- invisible(stepAIC(model1aicb,
    scope = list(lower = lm(log_PSA ~
        1, data = prostate, subset = train),
        upper = fit3), direction = "both"))
```

```r
# Use stepwise to find the best
# second-order model based on
# model1bic under BIC
model2bicbic <- invisible(stepAIC(model1bicb,
    scope = list(lower = lm(log_PSA ~
        1, data = prostate, subset = train),
        upper = fit3), direction = "both",
    k = log(length(train))))

sapply(list(model2aicaic = model2aicaic,
    model2bicbic = model2bicbic),
    "[[", "call")

## $model2aicaic
## lm(formula = log_PSA ~ log_cancer_volume + weight + age +
benign_prostatic_hyperplasia +
##     seminal_vesicle_invasion + gleason_score +
I(benign_prostatic_hyperplasia^2) +
##     benign_prostatic_hyperplasia:seminal_vesicle_invasion, data =
prostate,
##     subset = train)
##
## $model2bicbic
## lm(formula = log_PSA ~ log_cancer_volume + weight + age +
seminal_vesicle_invasion,
##     data = prostate, subset = train)

# ================== 3. Model
# Diagnostic and Validation
# ==========================


# 3.1 model diagnostic
par(mfrow = c(1, 2))
plot(model1aicb, which = c(1, 2))

plot(model1bicb, which = c(1, 2))

plot(model2aicaic, which = c(1,
    2))

# 3.2 Model validation
model1aicb_mspe <- sum((prostate$log_PSA[test] -
    predict(model1aicb, prostate[test,
        ]))^2)/length(test)
model1bicb_mspe <- sum((prostate$log_PSA[test] -
    predict(model1bicb, prostate[test,
        ]))^2)/length(test)
model2aicaic_mspe <- sum((prostate$log_PSA[test] -
    predict(model2aicaic, prostate[test,
        ]))^2)/length(test)
```

```r
# ================ 4. Final
# Model Analysis
# ========================
model_final <- lm(formula = log_PSA ~
    log_cancer_volume + weight +
        age + benign_prostatic_hyperplasia +
        seminal_vesicle_invasion +
        gleason_score, data = prostate)

summary(model_final)
```

```
##
## Call:
## lm(formula = log_PSA ~ log_cancer_volume + weight + age +
benign_prostatic_hyperplasia +
##     seminal_vesicle_invasion + gleason_score, data = prostate)
##
## Residuals:
##      Min       1Q   Median
## -1.64579 -0.48659  0.02146
##       3Q      Max
##  0.48710  1.91447
##
## Coefficients:
##                                Estimate
## (Intercept)                    2.124108
## log_cancer_volume              0.504257
## weight                         0.001849
## age                           -0.014547
## benign_prostatic_hyperplasia   0.072115
## seminal_vesicle_invasion1      0.651022
## gleason_score7                 0.127880
## gleason_score8                 0.639699
##                               Std. Error
## (Intercept)                     0.657133
## log_cancer_volume               0.080054
## weight                          0.001700
## age                             0.010876
## benign_prostatic_hyperplasia    0.027562
## seminal_vesicle_invasion1       0.215670
## gleason_score7                  0.174490
## gleason_score8                  0.240184
##                               t value
## (Intercept)                     3.232
## log_cancer_volume               6.299
## weight                          1.087
## age                            -1.338
## benign_prostatic_hyperplasia    2.616
```

```
## seminal_vesicle_invasion1      3.019
## gleason_score7                 0.733
## gleason_score8                 2.663
##                               Pr(>|t|)
## (Intercept)                    0.00172
## log_cancer_volume             1.11e-08
## weight                         0.27979
## age                            0.18445
## benign_prostatic_hyperplasia   0.01044
## seminal_vesicle_invasion1      0.00331
## gleason_score7                 0.46556
## gleason_score8                 0.00918
##
## (Intercept)                   **
## log_cancer_volume             ***
## weight
## age
## benign_prostatic_hyperplasia  *
## seminal_vesicle_invasion1     **
## gleason_score7
## gleason_score8                **
## ---
## Signif. codes:
##   0 '***' 0.001 '**' 0.01
##   '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7092 on 89 degrees of freedom
## Multiple R-squared:  0.6497, Adjusted R-squared:  0.6221
## F-statistic: 23.58 on 7 and 89 DF,  p-value: < 2.2e-16
```

```
anova(model_final)
```

```
## Analysis of Variance Table
##
## Response: log_PSA
##                               Df
## log_cancer_volume              1
## weight                         1
## age                            1
## benign_prostatic_hyperplasia   1
## seminal_vesicle_invasion       1
## gleason_score                  2
## Residuals                     89
##                               Sum Sq
## log_cancer_volume             68.801
## weight                         1.891
## age                            0.032
## benign_prostatic_hyperplasia   1.796
## seminal_vesicle_invasion       6.618
## gleason_score                  3.868
```

```
## Residuals                     44.763
##                               Mean Sq
## log_cancer_volume              68.801
## weight                          1.891
## age                             0.032
## benign_prostatic_hyperplasia    1.796
## seminal_vesicle_invasion        6.618
## gleason_score                   1.934
## Residuals                       0.503
##                               F value
## log_cancer_volume            136.7917
## weight                         3.7600
## age                            0.0638
## benign_prostatic_hyperplasia   3.5700
## seminal_vesicle_invasion      13.1582
## gleason_score                  3.8451
## Residuals
##                                  Pr(>F)
## log_cancer_volume            < 2.2e-16
## weight                       0.0556592
## age                          0.8012397
## benign_prostatic_hyperplasia 0.0620865
## seminal_vesicle_invasion     0.0004769
## gleason_score                0.0250221
## Residuals
##
## log_cancer_volume            ***
## weight                       .
## age
## benign_prostatic_hyperplasia .
## seminal_vesicle_invasion     ***
## gleason_score                *
## Residuals
## ---
## Signif. codes:
##   0 '***' 0.001 '**' 0.01
##   '*' 0.05 '.' 0.1 ' ' 1
```

```r
par(mfrow = c(2, 3))
plot(model_final, which = 1:6)
```

```r
model_final_del <- lm(formula = log_PSA ~
    log_cancer_volume + weight +
        age + benign_prostatic_hyperplasia +
        seminal_vesicle_invasion +
        gleason_score, data = prostate[-32,
    ])
summary(model_final_del)
```

```
##
## Call:
## lm(formula = log_PSA ~ log_cancer_volume + weight + age +
benign_prostatic_hyperplasia +
##     seminal_vesicle_invasion + gleason_score, data = prostate[-32,
##     ])
##
## Residuals:
##      Min       1Q   Median
## -1.54232 -0.38308  0.05055
##       3Q      Max
##  0.44597  1.46036
##
## Coefficients:
##                                Estimate
## (Intercept)                    2.041883
## log_cancer_volume              0.488958
## weight                         0.011098
## age                           -0.017439
## benign_prostatic_hyperplasia   0.042962
## seminal_vesicle_invasion1      0.627956
## gleason_score7                 0.135673
## gleason_score8                 0.573952
##                               Std. Error
## (Intercept)                     0.646759
## log_cancer_volume               0.078991
## weight                          0.004797
## age                             0.010776
## benign_prostatic_hyperplasia    0.030561
## seminal_vesicle_invasion1       0.212156
## gleason_score7                  0.171449
## gleason_score8                  0.238096
##                               t value
## (Intercept)                     3.157
## log_cancer_volume               6.190
## weight                          2.313
## age                            -1.618
## benign_prostatic_hyperplasia    1.406
## seminal_vesicle_invasion1       2.960
## gleason_score7                  0.791
## gleason_score8                  2.411
##                               Pr(>|t|)
## (Intercept)                    0.00218
## log_cancer_volume             1.86e-08
## weight                         0.02303
## age                            0.10916
## benign_prostatic_hyperplasia   0.16331
## seminal_vesicle_invasion1      0.00395
## gleason_score7                 0.43088
## gleason_score8                 0.01801
```

```
##
## (Intercept)                   **
## log_cancer_volume             ***
## weight                        *
## age
## benign_prostatic_hyperplasia
## seminal_vesicle_invasion1     **
## gleason_score7
## gleason_score8                *
## ---
## Signif. codes:
##   0 '***' 0.001 '**' 0.01
##   '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6967 on 88 degrees of freedom
## Multiple R-squared:  0.6651, Adjusted R-squared:  0.6385
## F-statistic: 24.97 on 7 and 88 DF,  p-value: < 2.2e-16
```

```
anova(model_final_del)
```

```
## Analysis of Variance Table
##
## Response: log_PSA
##                               Df
## log_cancer_volume              1
## weight                         1
## age                            1
## benign_prostatic_hyperplasia   1
## seminal_vesicle_invasion       1
## gleason_score                  2
## Residuals                     88
##                               Sum Sq
## log_cancer_volume             68.720
## weight                         6.439
## age                            0.690
## benign_prostatic_hyperplasia   0.122
## seminal_vesicle_invasion       5.913
## gleason_score                  2.954
## Residuals                     42.710
##                               Mean Sq
## log_cancer_volume              68.720
## weight                          6.439
## age                             0.690
## benign_prostatic_hyperplasia    0.122
## seminal_vesicle_invasion        5.913
## gleason_score                   1.477
## Residuals                       0.485
##                                F value
## log_cancer_volume             141.5895
## weight                         13.2661
```

```
## age                              1.4214
## benign_prostatic_hyperplasia     0.2504
## seminal_vesicle_invasion        12.1839
## gleason_score                    3.0428
## Residuals
##                                    Pr(>F)
## log_cancer_volume               < 2.2e-16
## weight                          0.0004560
## age                             0.2363730
## benign_prostatic_hyperplasia    0.6180218
## seminal_vesicle_invasion        0.0007557
## gleason_score                   0.0527511
## Residuals
##
## log_cancer_volume               ***
## weight                          ***
## age
## benign_prostatic_hyperplasia
## seminal_vesicle_invasion        ***
## gleason_score                   .
## Residuals
## ---
## Signif. codes:
##   0 '***' 0.001 '**' 0.01
##   '*' 0.05 '.' 0.1 ' ' 1
```

```r
par(mfrow = c(2, 3))
plot(model_final_del, which = 1:6)
```