Homework 8

Kevin Linares and Jamila Sani

2024-11-15

```r
library(jtools)
library(faraway)
library(olsrr)
library(knitr)
library(gridExtra)
library(car)
library(performance)
library(sandwich)
library(corrplot)
library(tidyverse)

options(scipen=999)

#________________ handy functions __________________
kappa_fun <- function(mod_name){
  # model matrix w/o intercept
  x_mod = model.matrix(mod_name)[,-1]
  #  matrix multiplication
  e=eigen(t(x_mod)%*%x_mod)
  print("Inspect wide range in eigenvalues")
  print(e$val)

  # calculate kappa values
  print("Inspect if Kappa conditional number value > 30")
  kappa_mod = sqrt(max(e$val)/min(e$val))
  print(kappa_mod)
  print("Inspect condition index, of at least one linear combination")
  nu = sqrt(max(e$val)/e$val)
  print(nu) }
```

1. Faraway Chapter 7. Exercise 5.

- For the prostate data, fit a model with lpsa as the response and the other variables as predictors.

```r
data("prostate")

summ(mod <- lm(lpsa ~ lcavol + lweight + age +
              lbph + svi + lcp + gleason + pgg45, prostate))
```

```
MODEL INFO:
Observations: 97
Dependent Variable: lpsa
Type: OLS linear regression

MODEL FIT:
F(8,88) = 20.86, p = 0.00
R² = 0.65
Adj. R² = 0.62

Standard errors:OLS
------------------------------------------------
                     Est.   S.E.   t val.      p
----------------- -------- ------ -------- ------
(Intercept)          0.67   1.30     0.52   0.61
lcavol               0.59   0.09     6.68   0.00
lweight              0.45   0.17     2.67   0.01
age                 -0.02   0.01    -1.76   0.08
lbph                 0.11   0.06     1.83   0.07
svi                  0.77   0.24     3.14   0.00
lcp                 -0.11   0.09    -1.16   0.25
gleason              0.05   0.16     0.29   0.78
pgg45                0.00   0.00     1.02   0.31
------------------------------------------------
```

1.A. Compute and comment on Kappa and the condition numbers.

- We compute the conditional number kappa, which measures the relative sizes of the eigenvalues where $\kappa > 30$ is considered large. Our kappa value of 243 is 8 times larger than 30, but can only tell us that at least one of the eigenvalue is small relative to the rest. Therefore, we examine other conditional indices η_j because they indicate whether more than just one independent linear combination is to blame. We see that 6 of 8 conditional indices are above 30, suggesting several independent linear combinations are present in this model.

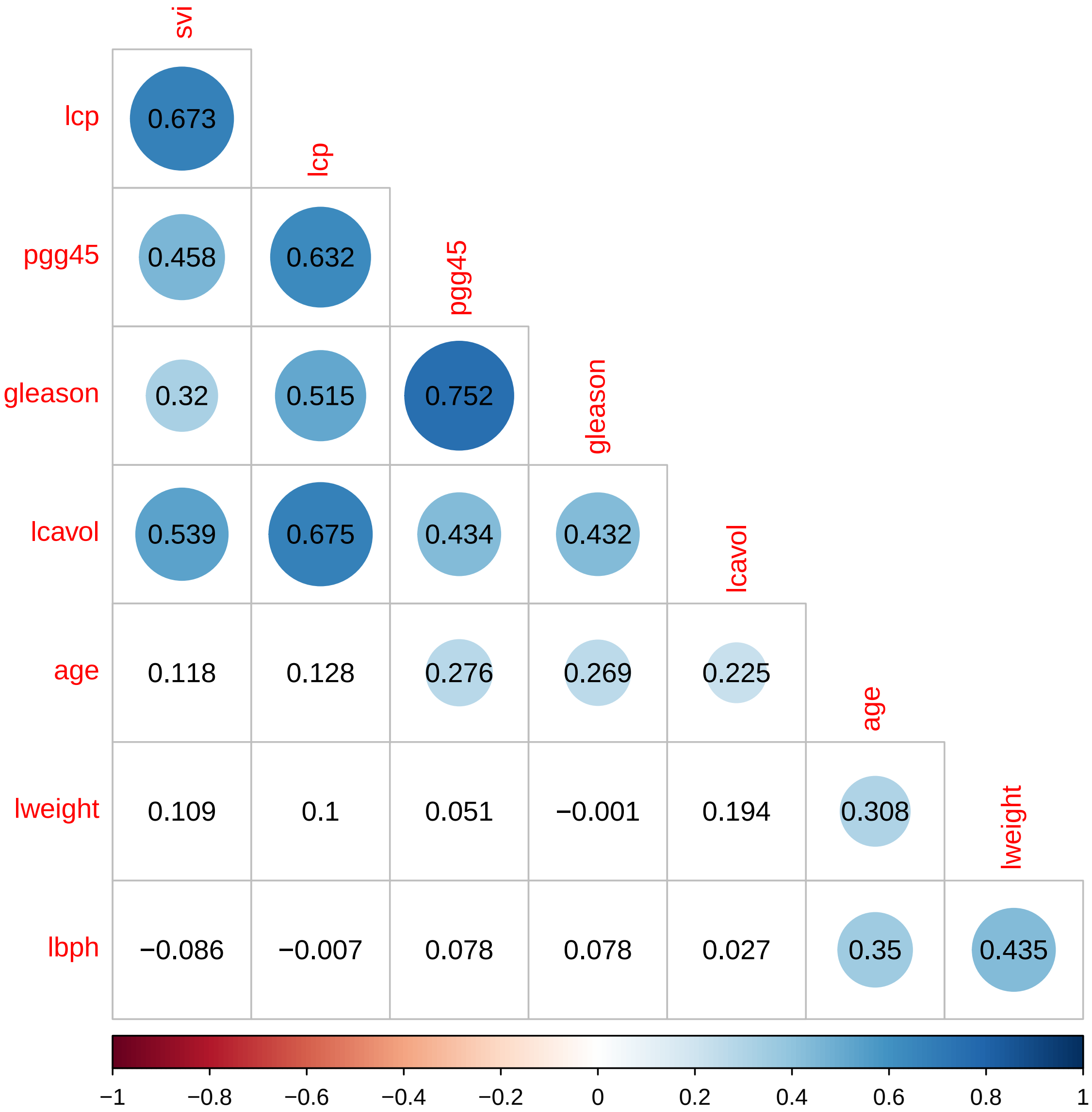
```
# pass the model object to this function
kappa_fun(mod)
```

```
[1] "Inspect wide range in eigenvalues"
[1] 479082.631078  61907.037212    210.904212    175.632866     64.798532
[6]     44.523793     20.239139      8.093145
[1] "Inspect if Kappa conditional number value > 30"
[1] 243.3025
[1] "Inspect condition index, of at least one linear combination"
[1]   1.00000   2.78186  47.66094  52.22787  85.98499 103.73114 153.85414
[8] 243.30248
```

1.B. Compute and comment on the correlations between the predictors. Round to 3 decimal places.

- In the correlation matrix contour plot below we take note on the strong positive association between gleason and pgg45 (r=.75), lcp lcavol (r=.68), svi and lcp (r=.67), and pgg45 and lcp (r=.63). The top correlated predictors also seem to have higher VIFs shown in 1.c.

```
testRes = cor.mtest(prostate |> select(-lpsa),
                    conf.level = 0.95)

corrplot(cor(prostate |> select(-lpsa)),
         p.mat = testRes$p, method = 'circle',
         type = 'lower', insig='blank',
         order = 'AOE', diag = FALSE,
         number.cex=0.05)$corrPos -> p1
text(p1$x, p1$y, round(p1$corr, 3))
```


svi
lcp
pgg45
gleason
lcavol
age
lweight
lcp
0.673
pgg45
0.458
0.632
gleason
0.32
0.515
0.752
lcavol
0.539
0.675
0.434
0.432
age
0.118
0.128
0.276
0.269
0.225
lweight
0.109
0.1
0.051
−0.001
0.194
0.308
lbph
−0.086
−0.007
0.078
0.078
0.027
0.35
0.435
−1
−0.8
−0.6
−0.4
−0.2
0
0.2
0.4
0.6
0.8
1

1.C. Compute the variance inflation factors. Comment on whether any appear problematic and why.

- The variance inflation factor (VIF) assesses multicollinearity, which is the ratio of the variance of $\hat{\beta}_j$: when fitting the full model divided by the variance of $\hat{\beta}_j$ if fit on its own. The smallest possible value is 1 indicating absence of collinearity. A rule of thumb of a VIF of over 5 is an indication of a problematic amount of collinearity. VIF is expressed as:

$$VIF_j = \frac{1}{1 = R_j^2}$$

- VIF ranged between 1.3 and 3.1. Of note, lcavol, lcp, gleason, and pgg45 have VIF>2, yet all VIF values are less than 5 so there does not appear to be any issues of collinearity.

```r
ols_vif_tol(mod) |> select(Variables, VIF) |> arrange(desc(VIF))
```

```
  Variables      VIF
1       lcp 3.097954
2     pgg45 2.974361
3   gleason 2.473411
4    lcavol 2.054115
5       svi 1.956881
6      lbph 1.375534
7   lweight 1.363704
8       age 1.323599
```

2. Faraway Chapter 8. Exercise 4.

- For the cars dataset, fit a linear model with distance as the response and speed as the predictor.

```
data("cars")
?cars
glimpse(cars)
```

```
Rows: 50
Columns: 2
$ speed <dbl> 4, 4, 7, 7, 8, 9, 10, 10, 10, 11, 11, 12, 12, 12, 12, 13, 13, 13~
$ dist  <dbl> 2, 10, 4, 22, 16, 10, 18, 26, 34, 17, 28, 14, 20, 24, 28, 26, 34~
```

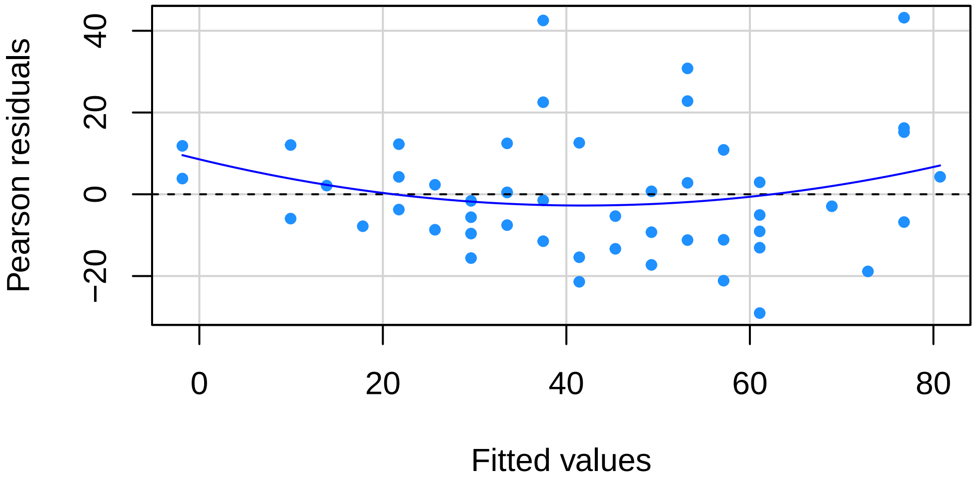
```
summ(mod_cars <- lm(dist ~ speed, cars))
```

```
MODEL INFO:
Observations: 50
Dependent Variable: dist
Type: OLS linear regression

MODEL FIT:
F(1,48) = 89.57, p = 0.00
R² = 0.65
Adj. R² = 0.64

Standard errors:OLS
-------------------------------------------------
                      Est.   S.E.   t val.      p
----------------- --------- ------ --------- ------
(Intercept)         -17.58   6.76    -2.60   0.01
speed                 3.93   0.42     9.46   0.00
-------------------------------------------------
```

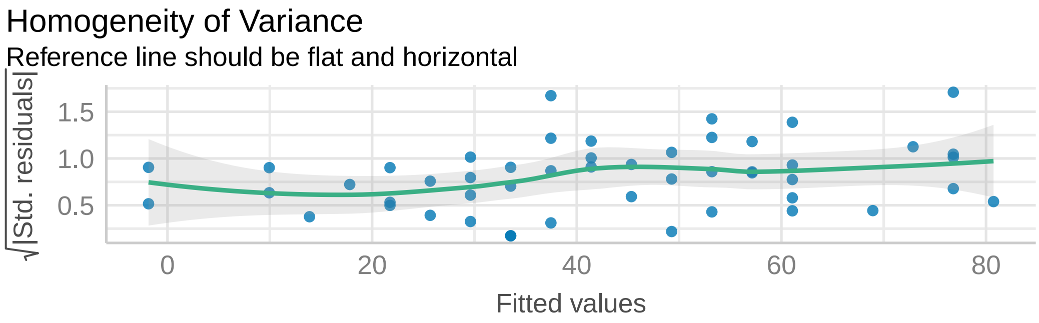
2.A. Test the homoscedasticity assumption using both a scatter plot between the residuals and fitted values and an F-test of equal variance below and above the fitted value of 30. What do you conclude about whether the assumption is met?

- There is a discernible pattern funnel-shape observed on the plot - more tightly clustered around the left half and more dispersed and spread apart on the right half of the plot. It is suggestive of non-constant variance or heteroscedasticity thus violating the constant variance assumption. The density and homogeneity of variance plots also show evidence of non-constant variance We can also see that lower fitted values have residuals higher than 0, while some mid point values are below 0. Therefore, we are seeing evidence for heteroscadasticity.

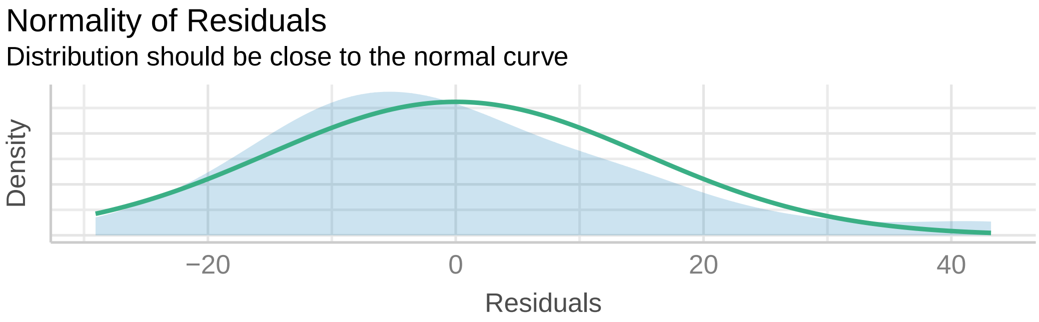
```r
residualPlot(mod_cars, pch=20, col= "dodgerblue")
```



```r
grid.arrange(
  # check for homogeneity: ref line is not flat
  plot(check_heteroskedasticity(mod_cars)),
  # check for normallity of residuals,
  plot(check_normality(mod_cars), type="density")
)
```


Homogeneity of Variance

Reference line should be flat and horizontal


Normality of Residuals

Distribution should be close to the normal curve

- We examine statistically equal variances for fitted values greater than 30 and values less than or equal to 30. The F-test for equal variances suggest suggests rejecting the null (95%$[1.53, 9.42]$) that the true ratio of variances of the two groups (>30 and <=30) are equal in favor of the alternative hypothesis. This further supports the observed violation of the constant error variance assumption on the residuals vs. fitted plot and homogeneity of variance plot, thus we conclude that we have evidence for non-constant variance both graphically and statistically. Residual variance for full model = 231.7045112, fitted values >30 = 303.4187299 fitted values <=30 = 73.4225165

```
var.test(resid(mod_cars)[fitted(mod_cars)>30],
         resid(mod_cars)[fitted(mod_cars)<=30])
```

```

    F test to compare two variances

data:  resid(mod_cars)[fitted(mod_cars) > 30] and resid(mod_cars)[fitted(mod_cars) <= 30]
F = 4.1325, num df = 34, denom df = 14, p-value = 0.006658
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 1.527594 9.415644
sample estimates:
ratio of variances
          4.132502
```

2.B. Report the estimate of the heteroscedastic consistent variance for the regression slope.

- The heteroscedastic consistent variance for the slope is .18.

```r
hetvar <- mod_cars |>
  # calculate Heteroscedasticity-consistent estimation
  ## of the covariance matrix for coefficients
  vcovHC()  |>
  # gives variances as they are the diagonal
  ## of the covariance matrix
  diag()

hetvar
```

```
(Intercept)       speed
 35.1862906   0.1827881
```

2.C. Construct 95% confidence interval of the regression slope assuming homoscedasticity and using the results in 2.B. How do they compare?

- 95% Confidence interval 1: assuming homoscedasticity [3.097, 4.7679]

```r
coef(mod_cars)[2]
```

```
   speed
3.932409
```

```r
confint(mod_cars)[2,]
```

```
   2.5 %   97.5 %
3.096964 4.767853
```

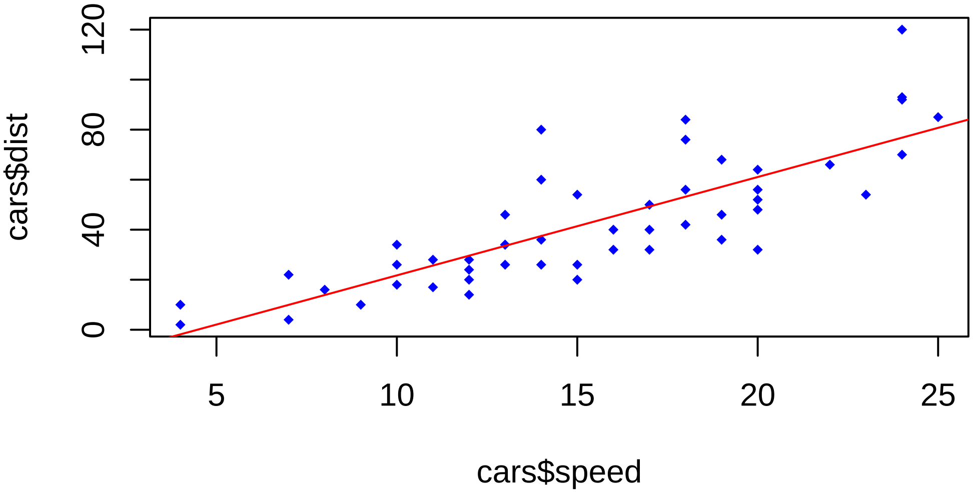
```
hetvar_cars <- hetvar |>
  # variances are the diagonal of the covariance matrix
  sqrt()

hetero_ci <- mod_cars$coefficients[2] + c(-1,1) *
  qt(0.975,mod_cars$df.residual)*sqrt(hetvar_cars[2])
```

- 95% confidence interval 2: heteroscedasticity [2.6177, 5.2471]
- CI heteroscedasticity is much wider and less precise than CI assuming homoscedasticity. This is because CI 2 is estimated using biased and inflated standard error reducing its precision.

2.D. Check for the lack of fit of the model.

- We use the function we created at the beginning of this rmarkdown and pass it two models, the original model and one where we convert the continuous variable into a factor.

```
plot(cars$speed,cars$dist,pch=18,col="blue",cex=.7) +
  abline(lm(dist~speed, cars),col="red")
```



```
integer(0)
```

```
summ(mod_cars_a<-lm(resid(mod_cars) ~ factor(speed), cars))
```

```
MODEL INFO:
Observations: 50
Dependent Variable: resid(mod_cars)
Type: OLS linear regression

MODEL FIT:
F(18,31) = 1.17, p = 0.34
R² = 0.40
Adj. R² = 0.06

Standard errors:OLS
------------------------------------------------------
                          Est.    S.E.   t val.      p
--------------------- -------- ------- -------- ------
(Intercept)               7.85   10.45     0.75   0.46
factor(speed)7           -4.80   14.77    -0.32   0.75
factor(speed)8           -5.73   18.09    -0.32   0.75
factor(speed)9          -15.66   18.09    -0.87   0.39
factor(speed)10          -3.59   13.49    -0.27   0.79
factor(speed)11         -11.03   14.77    -0.75   0.46
factor(speed)12         -15.96   12.79    -1.25   0.22
factor(speed)13          -6.39   12.79    -0.50   0.62
factor(speed)14           5.18   12.79     0.40   0.69
factor(speed)15         -15.92   13.49    -1.18   0.25
factor(speed)16         -17.19   14.77    -1.16   0.25
factor(speed)17         -16.45   13.49    -1.22   0.23
factor(speed)18           3.45   12.79     0.27   0.79
factor(speed)19         -14.99   13.49    -1.11   0.27
factor(speed)20         -18.52   12.36    -1.50   0.14
factor(speed)22         -10.78   18.09    -0.60   0.56
factor(speed)23         -26.72   18.09    -1.48   0.15
factor(speed)24           9.10   12.79     0.71   0.48
factor(speed)25          -3.58   18.09    -0.20   0.84
------------------------------------------------------
```

- The lack of fit F-test for the model's p-value of 0.29 is > .05. We fail to reject the null which is suggestive that the model does not fit the data well thus is not the best representation of the data.

```r
## use the olsrr package to conduct a lack of fit test
ols_pure_error_anova(mod_cars)
```

```
Lack of Fit F Test
------------------
Response :   dist
Predictor:   speed

                          Analysis of Variance Table
--------------------------------------------------------------------------------
                DF     Sum Sq     Mean Sq     F Value            Pr(>F)
--------------------------------------------------------------------------------
speed            1    21185.46    21185.46    97.08356     0.0000000000004102508
Residual        48    11353.52    236.5317
 Lack of fit    17    4588.738    269.9257     1.23695                 0.2948374
 Pure Error     31    6764.783    218.2188
--------------------------------------------------------------------------------
```