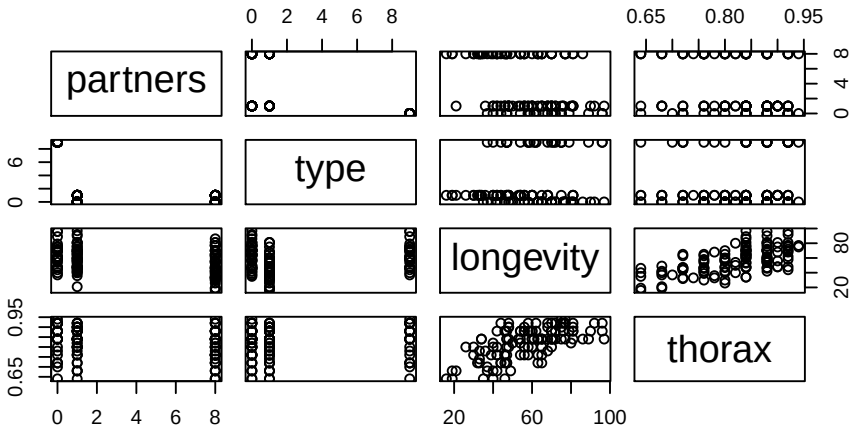
Solution to Series 3

1. **a)** At first we use the commands given in the exercise.

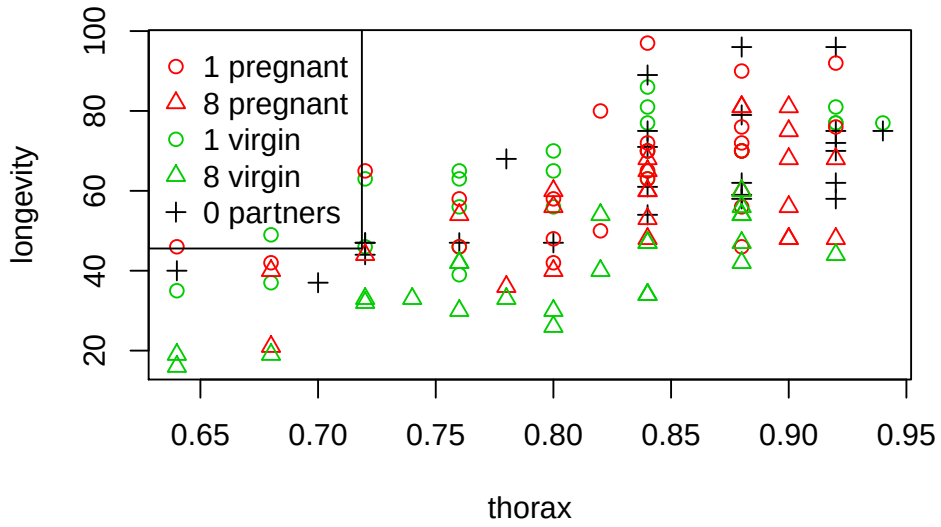
```
> url <- "https://ww2.amstat.org/publications/jse/datasets/fruitfly.dat.txt"
> data <- read.table(url)
> data <- data[,c(-1,-6)] # remove id and sleep
> names(data) <- c("partners","type","longevity","thorax")
> pairs(data)
```



From the pairs plot we see that there is a relation between `longevity` and `type`. Furthermore, thorax length is positively correlated with longevity.

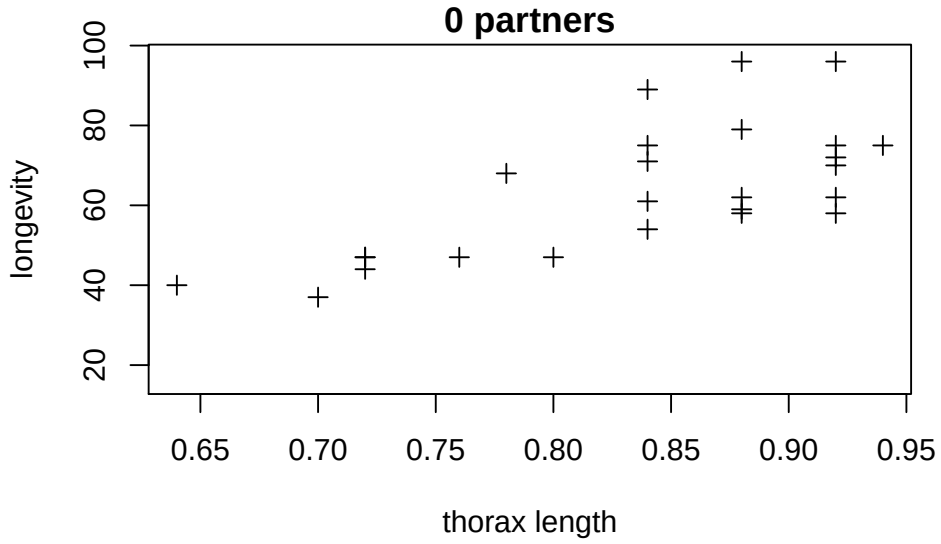
b)

```
> attach(data)
> # define vectors for the colors (col) and plotting characters (pch)
> col.partn <- 1*(partners==0) + 2*(partners==1) + 3*(partners==8)
>         # 1=black, 2=red, 3=green
> pch.type <- 1*(type==0) + 2*(type==1) + 3*(type==9)
>           # 1='circle', 2='triangle', 3='plus'
> # col.partn
> # pch.type
> par(mfrow=c(1,1))
> plot(thorax, longevity, pch=pch.type, col=col.partn,
      ylim=range(longevity),xlim=range(thorax))
> legend("topleft",c("1 pregnant","8 pregnant","1 virgin",
                     "8 virgin","0 partners"),
               pch=c(1,2,1,2,3),col=c(2,2,3,3,1))
```

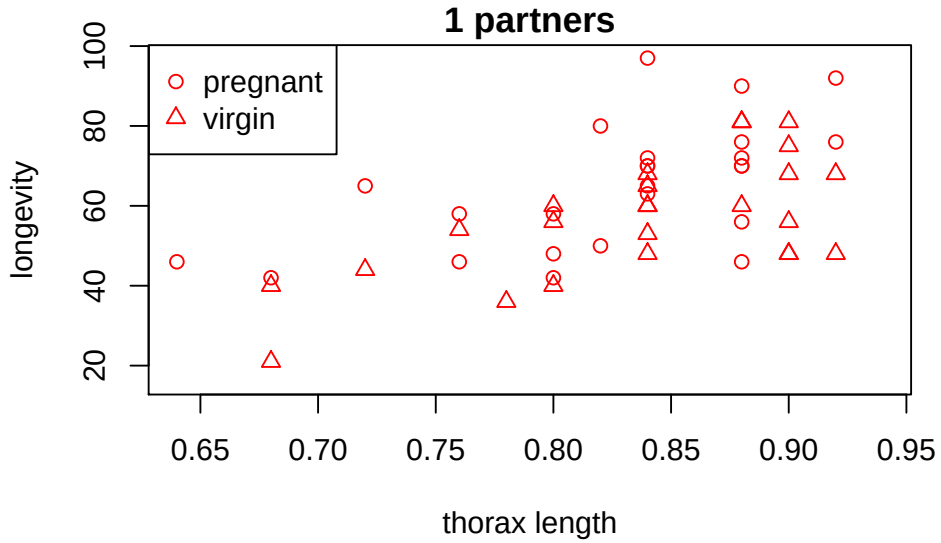


We can see that larger fruitflies tend to live longer. Furthermore, comparing fruitflies with similar thorax value, fruitflies with 8 virgins tend to live shorter.

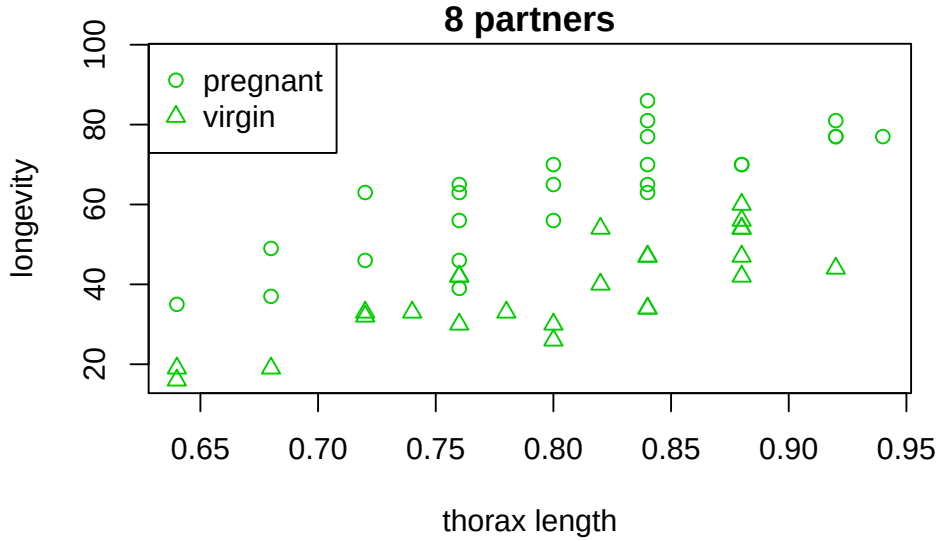
c)
```
> # par(mfrow=c(2,2))
> plot(thorax[partners==0],longevity[partners==0],pch=3,col=1,
      main="0 partners",ylim=range(longevity),xlim=range(thorax),
      xlab="thorax length", ylab="longevity")
```




```
> plot(thorax[partners==1],longevity[partners==1],
      pch=pch.type[partners==1],col=2,main="1 partners",
      ylim=range(longevity),xlim=range(thorax),
      xlab="thorax length", ylab="longevity")
> legend("topleft",c("pregnant","virgin"),pch=c(1,2),col=2)
```



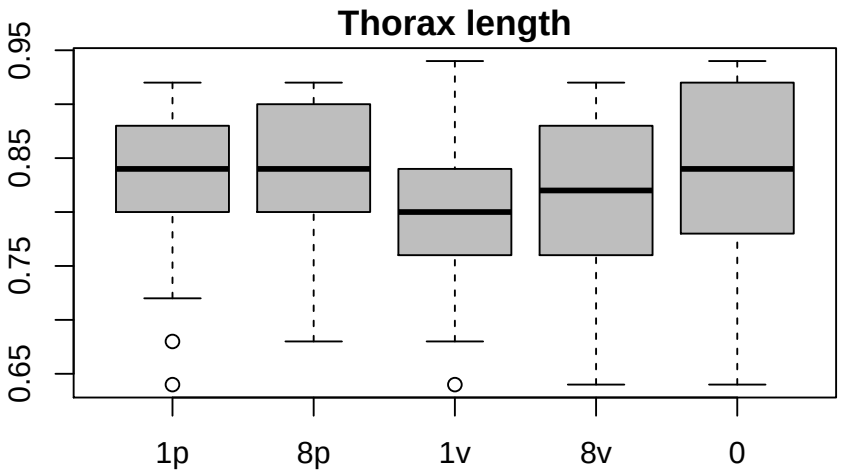

```
> plot(thorax[partners==8],longevity[partners==8],
      pch=pch.type[partners==8],col=3,main="8 partners",
      ylim=range(longevity),xlim=range(thorax),
      xlab="thorax length", ylab="longevity")
> legend("topleft",c("pregnant","virgin"),pch=c(1,2),col=3)
```



It seems that male fruitflies with pregnant females tend to live longer than those with virgin females. This difference in lifespan seems to be larger when `partners`=8 compared to `partners`=1. Hence, there seems to be an interaction effect between type and partners in their effect on longevity.

d)

```
> dummy.1.p <- (partners==1)*(type==0)*1
> dummy.1.v <- (partners==1)*(type==1)*1
> dummy.8.p <- (partners==8)*(type==0)*1
> dummy.8.v <- (partners==8)*(type==1)*1
> dummy.0   <- (partners==0)
> boxplot(thorax[dummy.1.p==1],thorax[dummy.1.v==1],
          thorax[dummy.8.p==1],thorax[dummy.8.v==1],
          thorax[dummy.0==1],
          main="Thorax length",
          names=c("1p","8p","1v","8v","0"),col="grey")
```



We can use an overall F-test to see if thorax is significantly different between at least two of the groups.

```
> fitfull<-lm(thorax~dummy.1.p+dummy.1.v+dummy.8.p+dummy.8.v)
> fitintercept<-lm(thorax~1)
> anova(fitintercept,fitfull)
Analysis of Variance Table

Model 1: thorax ~ 1
Model 2: thorax ~ dummy.1.p + dummy.1.v + dummy.8.p + dummy.8.v
  Res.Df     RSS Df Sum of Sq      F Pr(>F)
1    124 0.74388
2    120 0.71389  4  0.029997 1.2606 0.2893
```

The test is not significant. This was to be expected since the assignments to the groups were random, hence the distribution of thorax should be similar among the different groups.

e)

```
> fit_e1 <- lm(longevity[partners==1] ~ factor(type[partners==1]))
> summary(fit_e1)
Call:
lm(formula = longevity[partners == 1] ~ factor(type[partners ==
    1]))

Residuals:
   Min     1Q Median     3Q    Max
-35.76  -8.79   0.20  10.46  32.20

Coefficients:
                              Estimate Std. Error t value
(Intercept)                     64.800      3.059  21.184
factor(type[partners == 1])1    -8.040      4.326  -1.859
                              Pr(>|t|)
(Intercept)                     <2e-16 ***
factor(type[partners == 1])1    0.0692 .
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 15.29 on 48 degrees of freedom
Multiple R-squared:  0.06713,        Adjusted R-squared:  0.0477
F-statistic: 3.454 on 1 and 48 DF,  p-value: 0.06923
> fit_e2 <- lm((longevity)[partners==1] ~ thorax[partners==1] +
                                  factor(type[partners==1]))
> summary(fit_e2)
```

```
Call:
lm(formula = (longevity)[partners == 1] ~ thorax[partners ==
    1] + factor(type[partners == 1]))

Residuals:
    Min      1Q  Median      3Q     Max
-26.103  -9.123   1.092   7.273  30.267

Coefficients:
                             Estimate Std. Error t value
(Intercept)                   -46.038     20.799  -2.214
thorax[partners == 1]         134.252     25.019   5.366
factor(type[partners == 1])1   -9.651      3.456  -2.793
                             Pr(>|t|)
(Intercept)                   0.03175 *
thorax[partners == 1]        2.42e-06 ***
factor(type[partners == 1])1  0.00753 **
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.17 on 47 degrees of freedom
Multiple R-squared:  0.4215,        Adjusted R-squared:  0.3969
F-statistic: 17.12 on 2 and 47 DF,  p-value: 2.593e-06
```

We can see that type is much more significant in the second model which includes thorax. The t-value is obtained by dividing the point estimate by the estimate of the standard error. Note that the point estimates of the coefficient of type are slightly different in the two models, but we will leave this aside, and focus on the standard errors of the estimate. These are 3.456 in the model with thorax and 4.326 in the model without thorax. The ratio is 3.456/4.326=0.80. The smaller standard error in the model with thorax leads to a larger t-value and hence more significant results. Why is the standard error smaller in the model with thorax? The residual standard error is smaller in the model with thorax than in the model without thorax because thorax explains a significant amount of the variation in longevity. Moreover, thorax is not much correlated with type, so that we don't have to worry about large variance inflation factors.

f)
```
> partners.f <- as.factor(partners)
> type.f <- as.factor(type)
> fit_f1 <- lm(longevity ~ thorax + partners.f + type.f + partners.f*type.f)
> summary(fit_f1)

Call:
lm(formula = longevity ~ thorax + partners.f + type.f + partners.f *
    type.f)

Residuals:
    Min      1Q  Median      3Q     Max
-26.189  -6.599  -0.989   6.408  30.244

Coefficients: (4 not defined because of singularities)
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)          -49.984     10.609  -4.711 6.73e-06
thorax               135.819     12.439  10.919  < 2e-16
partners.f1            2.653      2.975   0.891 0.374483
partners.f8            3.929      2.997   1.311 0.192347
type.f1              -23.879      2.973  -8.031 7.83e-13
type.f9                   NA         NA      NA       NA
partners.f1:type.f1   14.210      4.210   3.375 0.000996
partners.f8:type.f1       NA         NA      NA       NA
partners.f1:type.f9       NA         NA      NA       NA
partners.f8:type.f9       NA         NA      NA       NA
```

```
(Intercept)         ***
thorax              ***
partners.f1
partners.f8
type.f1             ***
type.f9
partners.f1:type.f1 ***
partners.f8:type.f1
partners.f1:type.f9
partners.f8:type.f9
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.51 on 119 degrees of freedom
Multiple R-squared:  0.6564,        Adjusted R-squared:  0.6419
F-statistic: 45.46 on 5 and 119 DF,  p-value: < 2.2e-16
```

We only have 5 different groups of male fruitflies but there are 9 different combinations of the two three-level categorical predictors `type` and `partners`. The combinations (1,9), (8,9), (0,0) and (0,1) for (`partners`,`type`) do not appear in the dataset because they do not make sense. This is why we R reports "`Coefficients: (4 not defined because of singularities)`". We need to do the analysis more carefully, see the next subquestion.

g)
```
> fit_gFull <- lm((longevity) ~ thorax + dummy.1.p + dummy.1.v +
                        dummy.8.p + dummy.8.v)
> summary(fit_gFull)

Call:
lm(formula = (longevity) ~ thorax + dummy.1.p + dummy.1.v + dummy.8.p +
    dummy.8.v)

Residuals:
    Min      1Q  Median      3Q     Max
-26.189  -6.599  -0.989   6.408  30.244

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -49.984     10.609  -4.711 6.73e-06 ***
thorax       135.819     12.439  10.919  < 2e-16 ***
dummy.1.p      2.653      2.975   0.891   0.3745
dummy.1.v     -7.017      2.973  -2.361   0.0199 *
dummy.8.p      3.929      2.997   1.311   0.1923
dummy.8.v    -19.951      3.006  -6.636 1.00e-09 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.51 on 119 degrees of freedom
Multiple R-squared:  0.6564,        Adjusted R-squared:  0.6419
F-statistic: 45.46 on 5 and 119 DF,  p-value: < 2.2e-16
```

If there is no interaction then the difference in the predicted values between samples with `type`=0 and `type`=1 should be the same if the other predictors coincide, no matter whether `partners`=1 or `partners`=8. Mathematically, this means that $\gamma_{1,0} - \gamma_{1,1} = \gamma_{8,0} - \gamma_{8,1}$, which is equivalent to

$$\underline{\gamma_{1,0} = \gamma_{8,0} - \gamma_{8,1} + \gamma_{1,1}.}$$

If we plug this into the model, we get

$$\begin{aligned} \underline{y} &= \underline{\beta_0 + (\gamma_{8,0} - \gamma_{8,1} + \gamma_{1,1})p_1t_0 + \gamma_{1,1}p_1t_1 + \gamma_{8,0}p_8t_0 + \gamma_{8,1}p_8t_1} \\ &= \underline{\beta_0 + \gamma_{1,1}(p_1t_1 + p_1t_0) + \gamma_{8,0}(p_8t_0 + p_1t_0) + \gamma_{8,1}(p_8t_8 - p_1t_0)} \end{aligned}$$

We fit this model and conduct a partial F-test.

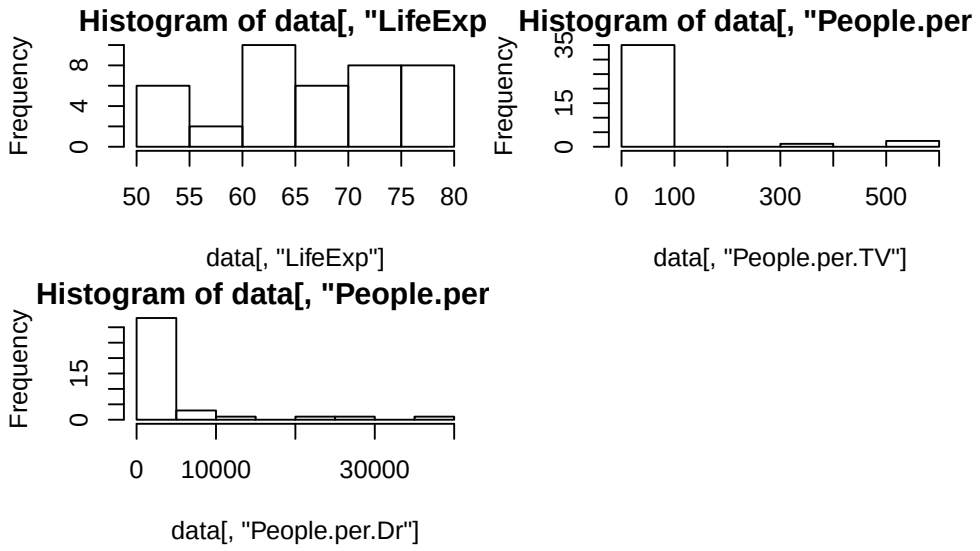
```
> fit_gPart <- lm((longevity) ~ thorax + I(dummy.1.v + dummy.1.p) +
                      I(dummy.8.p + dummy.1.p) +
                      I(dummy.8.v - dummy.1.p))
> anova(fit_gPart,fit_gFull)
Analysis of Variance Table

Model 1: (longevity) ~ thorax + I(dummy.1.v + dummy.1.p) + I(dummy.8.p +
    dummy.1.p) + I(dummy.8.v - dummy.1.p)
Model 2: (longevity) ~ thorax + dummy.1.p + dummy.1.v + dummy.8.p + dummy.8.v
  Res.Df   RSS Df Sum of Sq      F    Pr(>F)
1    120 14403
2    119 13145  1    1258.5 11.394 0.0009957 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

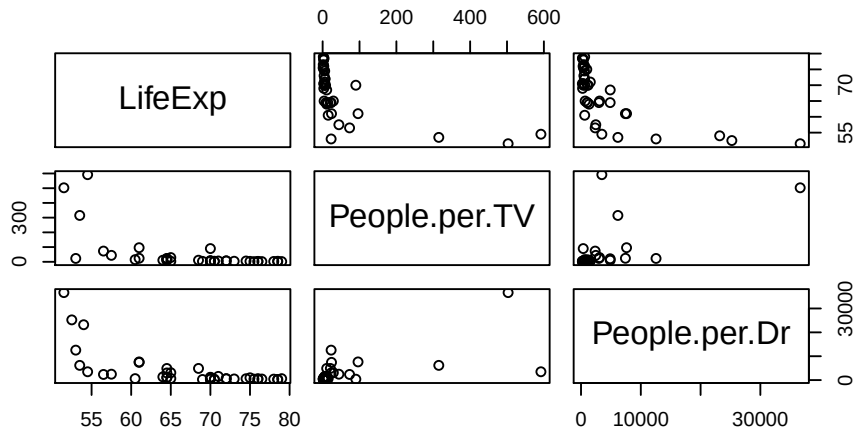
The partial F-test shows that the interaction between `type` and `partners` is significant.

2. **a)**

```
> url <- "https://raw.githubusercontent.com/jawj/coffeestats/master/lifeexp.dat"
> data <- read.table(url, sep="\t", header=T, row.names=1)
> data <- data[,c("LifeExp","People.per.TV","People.per.Dr")]
> par(mfrow=c(2,2))
> hist(data["LifeExp"])
> hist(data["People.per.TV"])
> hist(data["People.per.Dr"])
```




```
> plot(data)
```




```
> data[order(data[,"LifeExp"],decreasing=T)[1:3],]
      LifeExp People.per.TV People.per.Dr
Japan    79.0           1.8           609
Italy    78.5           3.8           233
Spain    78.5           2.6           275
> data[order(data[,"People.per.TV"],decreasing=T)[1:3],]
           LifeExp People.per.TV People.per.Dr
Burma         54.5           592          3485
Ethiopia      51.5           503         36660
Bangladesh    53.5           315          6166
> data[order(data[,"People.per.Dr"],decreasing=T)[1:3],]
         LifeExp People.per.TV People.per.Dr
Ethiopia    51.5           503         36660
Tanzania    52.5            NA         25229
Zaire       54.0            NA         23193
```

The countries with the highest life expectancy are Japan, Italy and Spain, the countries with the highest number of people per TV are Burma, Ethiopia and Bangladesh, the three countries with the highest number of people per doctor are Ethiopia, Tanzania and Zaire.

b)
```
> datanew <-data[complete.cases(data),]
> tv <- datanew$People.per.TV
> le <- datanew$LifeExp
> dr <- datanew$People.per.Dr
> l2tv=log2(tv)
> l2dr=log2(dr)
> fit<-lm(le~l2tv+l2dr)
> summary(fit)

Call:
lm(formula = le ~ l2tv + l2dr)

Residuals:
    Min      1Q  Median      3Q     Max
-7.7173 -2.7718  0.9026  2.9923  5.8553

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  90.6222     4.3557  20.806  < 2e-16 ***
l2tv         -2.0209     0.4094  -4.936 1.95e-05 ***
l2dr         -1.5657     0.5181  -3.022  0.00467 **
---
```

```
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.704 on 35 degrees of freedom
Multiple R-squared:  0.7868,        Adjusted R-squared:  0.7747
F-statistic:  64.6 on 2 and 35 DF,  p-value: 1.788e-12
```

In the original scale, the interpretation of the coefficients β_{tv} and β_{dr} of `l2tv` and `l2dr` is as follows: If we have two countries that have the same value for `People.per.TV` whereas the values for `People.per.Dr` differ by a factor of 2 (i.e. the `l2dr` values differ by 1) then the predicted values for life expectancy would differ by β_{dr}. Similarly for two countries with the same value for `People.per.Dr` and a difference in `People.per.TV` by a factor of 2 (i.e. the `l2tv` values differ by 1), the predicted values for life expectancy would differ by β_{tv}.

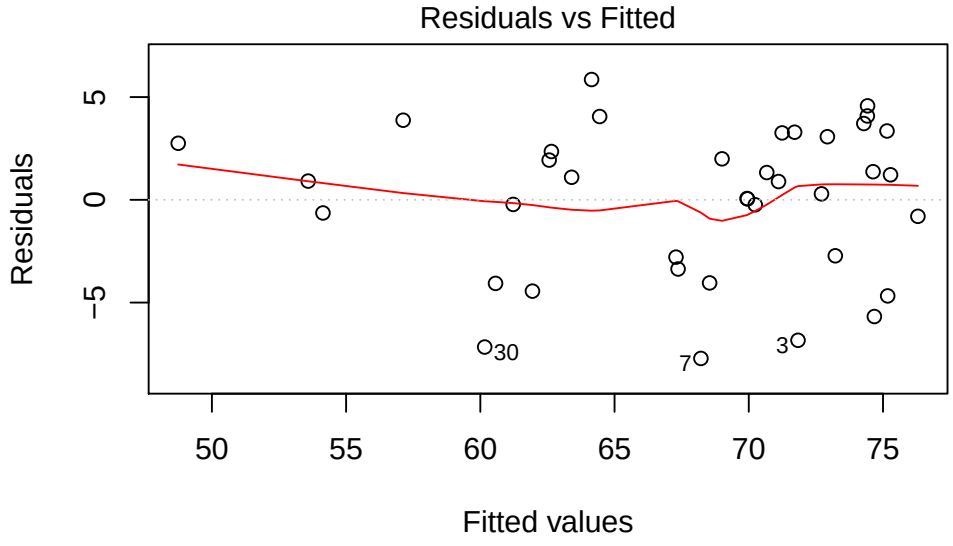
c) We cannot conclude that more TVs imply a higher life expectancy because we only have an observational study, which generally does not allow for conclusions about causal relations. However, we can use the number of people per TV to predict life expectancy for a new observation, i.e. country.

d) The confidence and prediction intervals can be computed as follows.

```
> newcountry=data.frame(l2tv=log2(50),l2dr=log2(3000))
> predict(fit, newdata=newcountry, interval="confidence")
       fit      lwr      upr
1 61.13099 59.41061 62.85137
> predict(fit, newdata=newcountry, interval="predict")
       fit      lwr      upr
1 61.13099 53.41774 68.84424
```

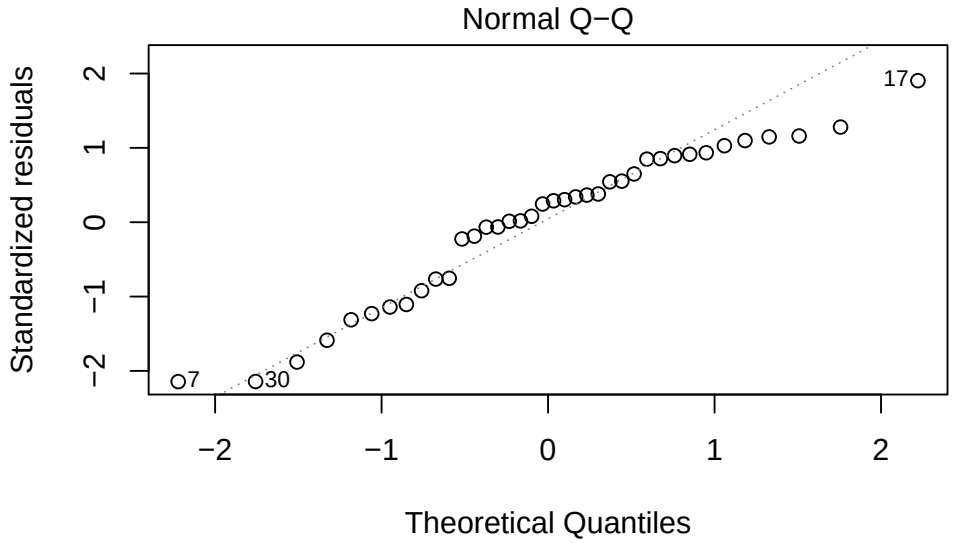
e)

```
> par(mfrow=c(1,1))
> plot(fit,which =1)
```



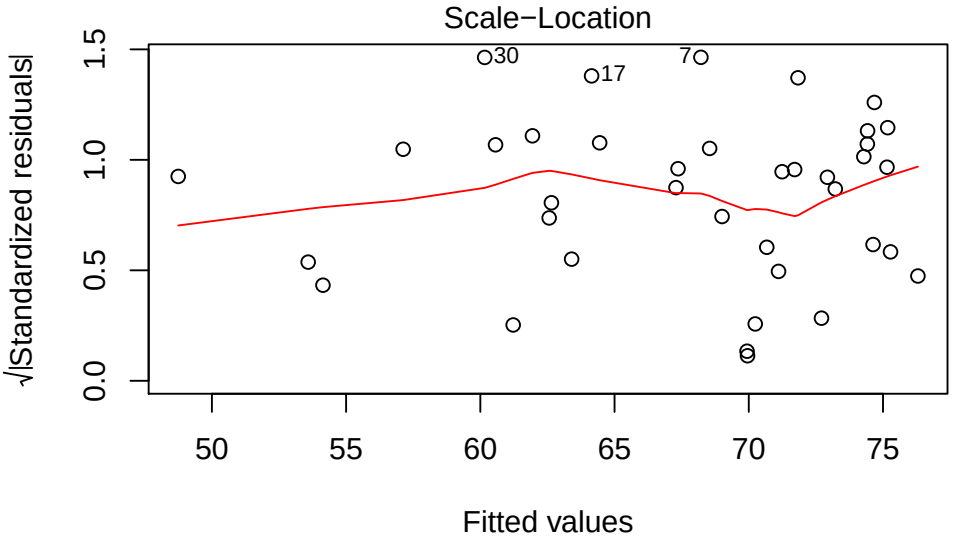
In the Tukey-Anscombe plot, we do not see any clear model violations.

```
> plot(fit,which =2)
```



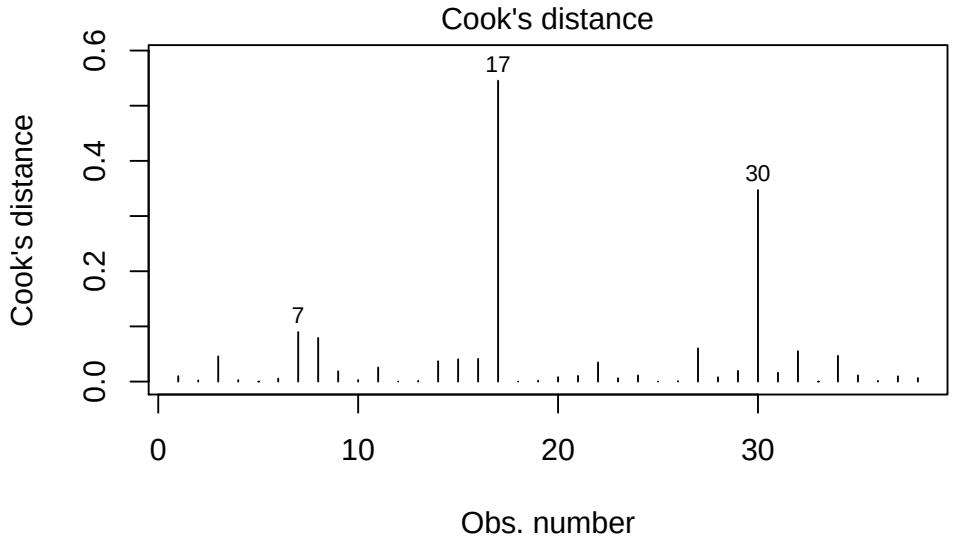
The QQ plot shows that the distribution of the residuals is somewhat left-skewed, but this is not very severe.

```
> plot(fit,which =3)
```



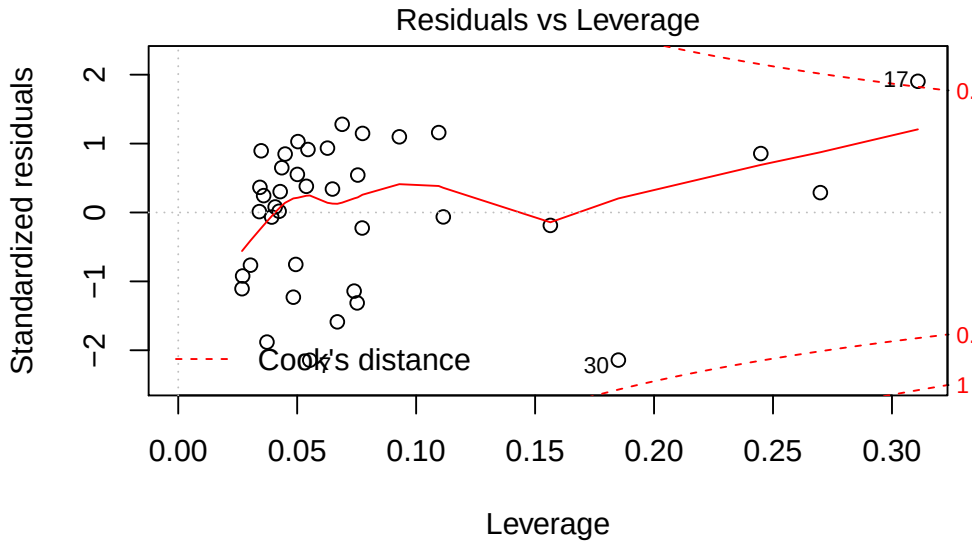
The scale-location plot shows that the variance of the residuals does not depend on the fitted values. There are no indications of heteroscedasticity (= non-equal error variance).

```
> plot(fit,which =4)
> rownames(datanew)[c(17,30)]
[1] "Korea.North" "Sudan"
```



There are two countries which have clearly larger Cook's distance (i.e. have a high impact on the fitted regression plane) than the others: North Korea and Sudan.

```
> plot(fit,which =5)
```



In the above residuals versus leverage plot the red dotted lines indicate the levels 0.5 and 1 of the Cook's distance. We can observe that North Korea has a very large standardized residual, leverage and Cook's distance, such that we should analyze how the confidence and prediction intervals change if we exclude this observation.

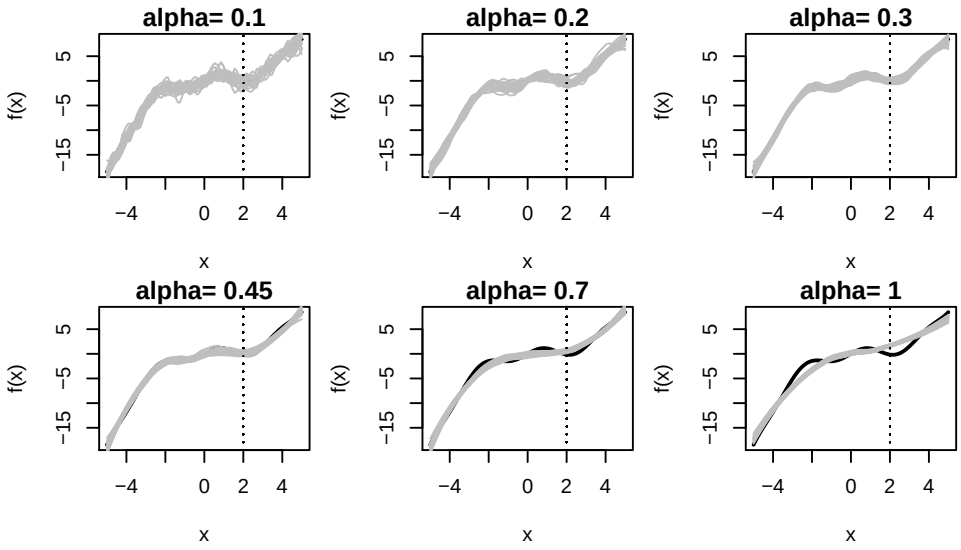
f) We exclude the two countries and fit another model.

```
> fit2<-lm(le[-c(17,30)]~l2tv[-c(17,30)]+l2dr[-c(17,30)])
> predict(fit2, newdata=newcountry, interval="confidence")
       fit      lwr      upr
1 60.95844 59.37872 62.53816
> predict(fit2, newdata=newcountry, interval="predict")
       fit      lwr      upr
1 60.95844 54.23489 67.68199
```

We see that the values for the confidence and prediction intervals do not differ too much from the ones in part d). This is a good sign, since we do not want our results to depend heavily on only two observations.

3. a) We use the following code

```
> f <- function(x){ .3* x - 0.2*x^2 + 0.1*x^3 + sin(2*x) }
> span <- c(0.1,0.2,0.3,0.45,0.7,1)   # smoothing parameter for loess:
> sigma <- 1.5                        # standard devation of noise
> n <- 100                            # sample size
> grid<-seq(-5,5,length=100)
> x <- seq(from=-5,to=5, length=n)    # x-values (fixed throughout simulation)
> xtest<-2
> par(mfrow=c(2,3))
> for (i in 1:length(span)){
    plot(x,f(x), type="l", lwd=2, main=paste("alpha=",span[i]))
    for(j in 1:25){
       y <- f(x) + rnorm(n=length(x),mean=0,sd=sigma)
       lo <- loess(y ~ x, span=span[i])
       lines(x, predict(object=lo, x),col="gray")
       abline(v=xtest, lty=3)
    }
 }
```

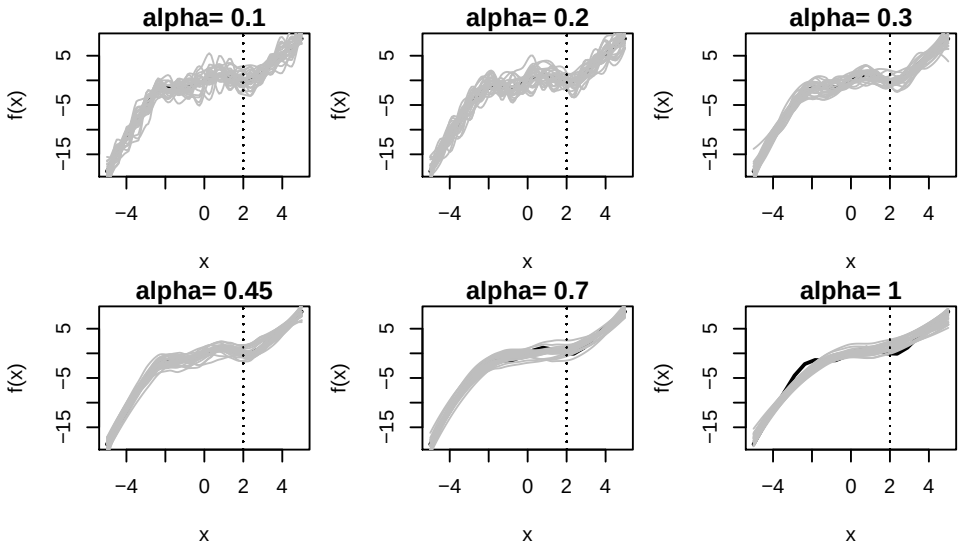


We can observe that for small values of `span` (complex models), there is more variance in the fitted values at `xtest`. This is because small `alpha` (bandwidth), only a close neighbourhood of samples with x-values around `alpha` are involved in the predicted value of the loess smoother at `xtest`. Therefore, the noise plays a greater role such that the variance of the predictions is larger. The larger the value of `span` the more data points in the x-neighbourhood of `xtest` are involved in the predicted value of the loess smoother such that the noise is not as important as for smaller values of `span`. At the same time, we have a positive bias at `x=xtest` because the smoother uses points with x-values in the neighbourhood that have mainly larger function values $f(x)$ (i.e. larger expectation for y). This is the bias-variance trade-off. If we increase the `span` for loess, the variance decreases but at the same time the squared bias increases.

b) At first we try a smaller sample size `n=20`.

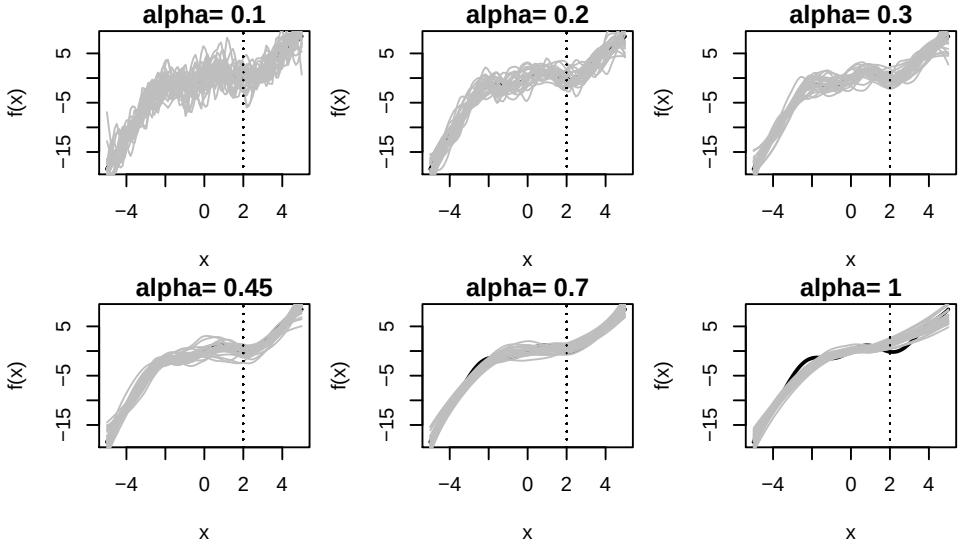
```
> plots<-function(){
 par(mfrow=c(2,3))
 for (i in 1:length(span)){
    plot(x,f(x), type="l", lwd=2, main=paste("alpha=",span[i]))
    for(j in 1:25){
       y <- f(x) + rnorm(n=length(x),mean=0,sd=sigma)
       lo <- loess(y ~ x, span=span[i])
       lines(grid, predict(object=lo, grid),col="gray")
       abline(v=xtest, lty=3)
    }
 }
 }
> sigma <- 1.5
```

```
> n <- 20
> x <- seq(from=-5,to=5, length=n)
> plots()
```



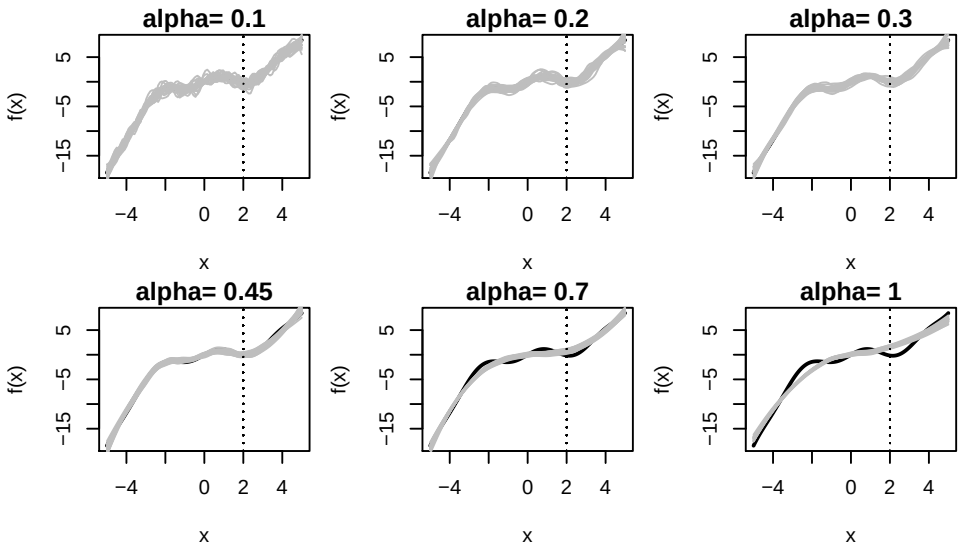
Now we take a larger `sigma` and set `n` back to 100.

```
> sigma <- 4
> n <- 100
> x <- seq(from=-5,to=5, length=n)
> plots()
```



Finally we use the increased `sigma` and use `n`=1000 .

```
> sigma <- 4
> n <- 1000
> x <- seq(from=-5,to=5, length=n)
> plots()
```

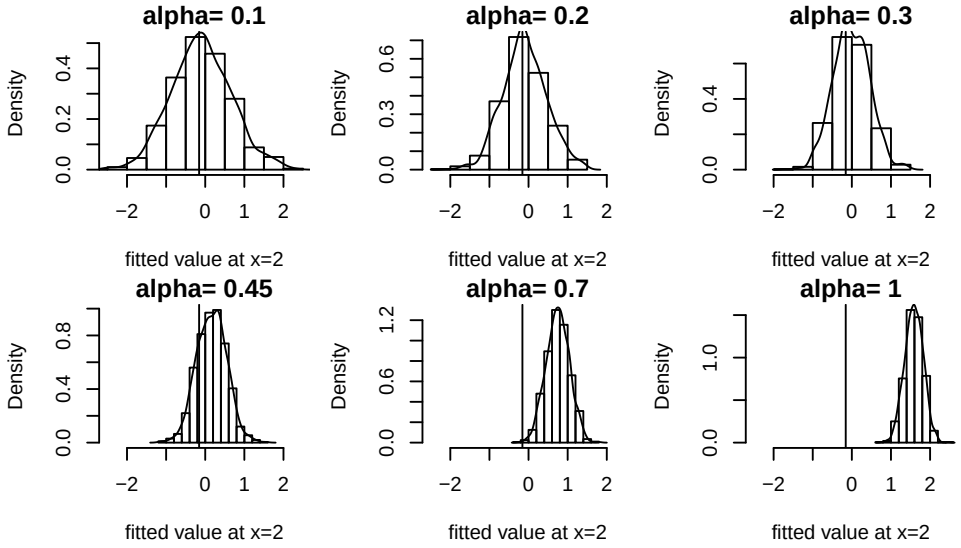


LOESS uses local polynomial regression based on αn observations. A larger n leads to more precise estimates. Increasing `sigma` leads of course to a larger variation in the fitted values.

c) We use the following parameters in Rcode3.R.

```
> nsim <- 1000            # number of simulations
> xtest <- 2            # x-value of test point
> sigma <- 1.5                    # standard devation of noise
> n <- 100                        # sample size
> grid<-seq(-5,5,length=n)
> x <- seq(from=-5,to=5, length=n)
```

Then we get the following histograms.



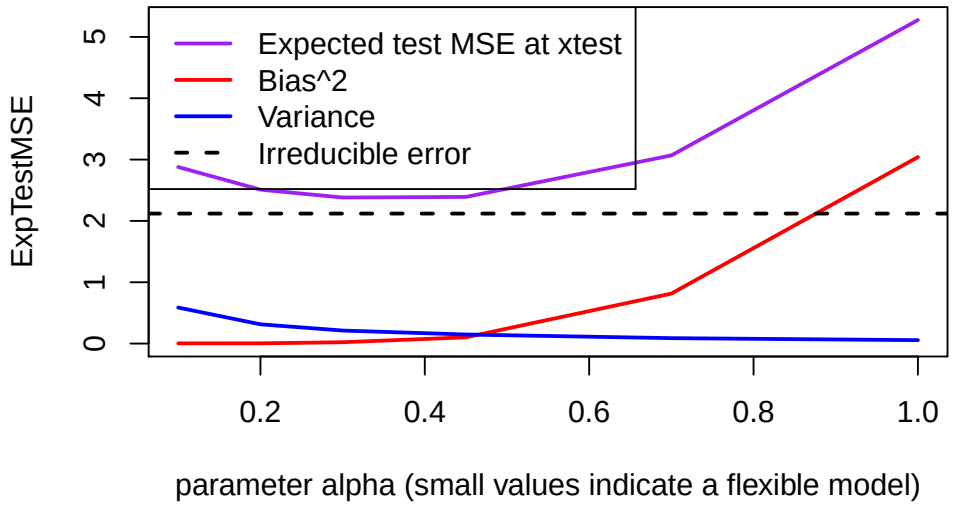
The histograms show that the fitted values at `xtest` are more concentrated for larger `span` values (there is a smaller variance) but at the same time, they have a systematic error (`f(xtest)=-0.157`), i.e. there is a larger bias. In other words, we observe the bias-variance trade-off.

d) The Rcode3.R produces the following output and plot for `xtest`$=2$:

```
> (ExpTestMSE <- apply((fit.test-y.test)^2,2,mean))
[1] 2.878689 2.509638 2.381170 2.391252 3.067762 5.275345
> (Bias2 <- (apply(fit.test,2,mean)-f(xtest))^2)
[1] 0.003212567 0.003689427 0.021888109 0.101381405
[5] 0.814906593 3.040387018
> (Var <- apply(fit.test,2,var))
[1] 0.58725226 0.31361867 0.21227947 0.14740228 0.08802842
[6] 0.05560389
```

```
> (VarY <- var(y.test))
[1] 2.119612
> Bias2+Var+VarY - ExpTestMSE
[1] -0.16861229 -0.07271820 -0.02739112 -0.02285669
[5] -0.04521516 -0.05974285
> # Note small errors because cross-terms do not fully disappear in simulation
```

The following plot visualizes the results.



4. Since $\hat{f}$ was constructed using only $(x_1, Y_1), \ldots, (x_n, Y_n)$ (and is thus a function of only $\varepsilon_1, \ldots, \varepsilon_n$), and since ε is independent of $\varepsilon_1, \ldots, \varepsilon_n$ through the iid assumption, it holds that $\hat{f}(x_0)$ and ε are independent. Thus

$$\begin{aligned} E\left[\left(Y_0 - \hat{f}(x_0)\right)^2\right] &= E\left[\left(f(x_0) + \varepsilon - \hat{f}(x_0)\right)^2\right] \\ &= E\left[\left[\left(f(x_0) - E(\hat{f}(x_0))\right) + \left(E(\hat{f}(x_0)) - \hat{f}(x_0)\right) + \varepsilon\right]^2\right]. \\ &= E\Big[\left(f(x_0) - E(\hat{f}(x_0))\right)^2 + \left(E(\hat{f}(x_0)) - \hat{f}(x_0)\right)^2 + \varepsilon^2 + \\ &\quad 2\left(f(x_0) - E(\hat{f}(x_0))\right)\left(E(\hat{f}(x_0)) - \hat{f}(x_0)\right) + 2\left(f(x_0) - E(\hat{f}(x_0))\right)\varepsilon + \\ &\quad 2\left(E(\hat{f}(x_0)) - \hat{f}(x_0)\right)\varepsilon\Big]. \end{aligned}$$

For the first two cross-terms we use that $f(x_0) - E(\hat{f}(x_0))$ is deterministic and $E(\varepsilon_i) = 0$ to obtain:

$$\begin{aligned} E\left[\left(f(x_0) - E(\hat{f}(x_0))\right)\left(E(\hat{f}(x_0)) - \hat{f}(x_0)\right)\right] &= \left(f(x_0) - E(\hat{f}(x_0))\right) E\left[\left(E(\hat{f}(x_0)) - \hat{f}(x_0)\right)\right] \\ &= \left(f(x_0) - E(\hat{f}(x_0))\right)\left(E(\hat{f}(x_0)) - E(\hat{f}(x_0))\right) = 0, \end{aligned}$$

and

$$E\left[\left(f(x_0) - E(\hat{f}(x_0))\right)\varepsilon\right] = \left(f(x_0) - E(\hat{f}(x_0))\right) E(\varepsilon) = 0.$$

Moreover, since ε and $\hat{f}(x_0)$ are independent, we have that

$$\begin{aligned} E\left[\left(E(\hat{f}(x_0)) - \hat{f}(x_0)\right)\varepsilon\right] &= E\left[E(\hat{f}(x_0)) - \hat{f}(x_0)\right] E(\varepsilon) \\ &= \left[E(\hat{f}(x_0)) - E(\hat{f}(x_0))\right] E(\varepsilon) = 0. \end{aligned}$$

Thus

$$E\left[\left(Y_0 - \hat{f}(x_0)\right)^2\right] = \left[f(x_0) - E(\hat{f}(x_0))\right]^2 + E\left[\left(E(\hat{f}(x_0)) - \hat{f}(x_0)\right)^2\right] + E(\varepsilon^2),$$

which is the desired result.