Exercise 2

Linear regression: bias of empirical risk

G. Bontempi

Question

Let us consider the dependency where the conditional distribution of $\mathbf{y}$ is

$$\mathbf{y} = 1 - x + x^2 - x^3 + \mathbf{w}$$

where $\mathbf{w} \sim N(0, \sigma^2)$, $x \in \Re$ takes the values $\texttt{seq}(-1, 1, \texttt{length.out} = \texttt{N})$ (with $N = 50$) and $\sigma = 0.5$.

Consider the family of regression models

$$h^{(m)}(x) = \beta_0 + \sum_{j=1}^{m} \beta_j x^j$$

where p denote the number of weights of the polynomial model $h^{(m)}$ of degree m.

Let $\widehat{\text{MISE}}^{(m)}_{\text{emp}}$ denote the least-squares empirical risk and MISE the mean integrated empirical risk. We make the assumption that a Monte Carlo simulation with $S = 10000$ repetitions return an accurate estimation of the expectation terms (notably the MISE term).

By using Monte Carlo simulation (with $S = 10000$ repetitions) and for $m = 0, \dots, 6$

- plot $E[\widehat{\mathbf{MISE}}^{(m)}_{\text{emp}}]$ as a function of p,
- plot $\text{MISE}^{(m)}$ as a function of p,
- plot the difference $E[\widehat{\mathbf{MISE}}^{(m)}_{\text{emp}}] - \text{MISE}^{(m)}$ as a function of p and compare it with the theoretical result seen during the class.

For a single observed dataset:

- plot $\widehat{\text{MISE}}^{(m)}_{\text{emp}}$ as a function of the number of model parameters p,
- plot PSE as a function of p,
- discuss the relation between $\arg\min_m \widehat{\text{MISE}}^{(m)}_{\text{emp}}$ and $\arg\min_m \text{PSE}^{(m)}$

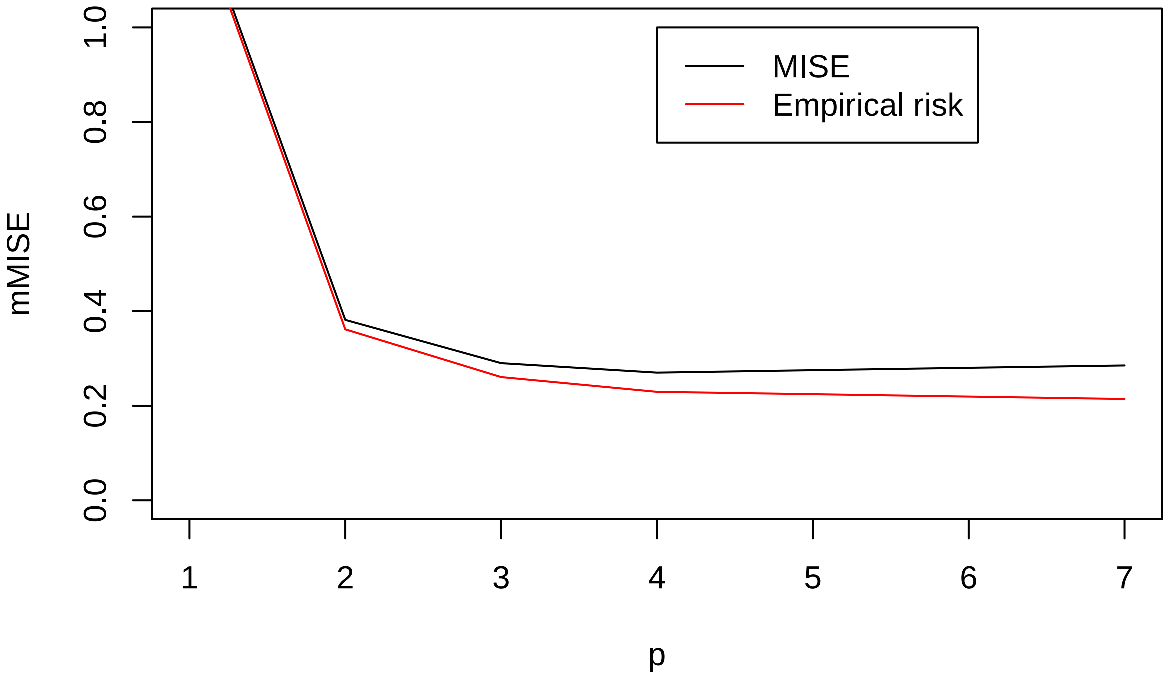
NOTA BENE: the use of the R command `lm` is NOT allowed.

Monte Carlo Simulation

```
N=50 ## number of samples

S=10000 ## number of MC trials
M=6  ## max order of the polynomial model
sdw=0.5 ## stanard deviation of noise

Emp<-array(NA,c(M+1,S))
MISE<-array(NA,c(M+1,S))
for (s in 1:S){
  X=seq(-1,1,length.out=N)
  Y=1-X+X^2-X^3+rnorm(N,sd=sdw)

  Xts=X
  Yts=1-Xts+Xts^2-Xts^3+rnorm(N,sd=sdw)

  for (m in 0:M){
    DX=NULL
    for (j in 0:m){
      DX=cbind(DX,X^j)
    }

    betahat=solve(t(DX)%*%DX)%*%t(DX)%*%Y
    Yhat=DX%*%betahat
    Emp[m+1,s]=mean((Y-Yhat)^2)
    MISE[m+1,s]=mean((Yts-Yhat)^2)

  }



}

  mMISE=apply(MISE,1,mean)
  mEmp=apply(Emp,1,mean)
```

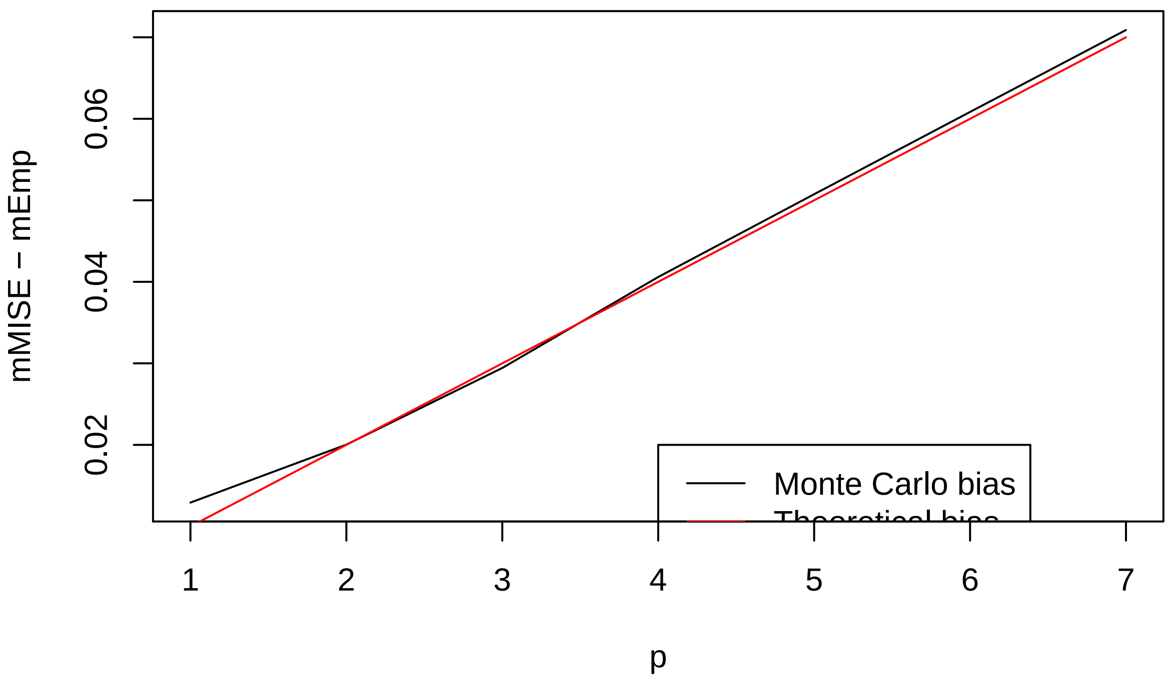
Plot expected empirical risk and MISE

```
plot(mMISE, ylim=c(0,1), type="l",xlab="p")

lines(mEmp,col="red")
legend(x=4,y=1,c("MISE","Empirical risk"),lty=1, col=c("black","red"))
```



Plot bias of empirical risk vs theoretical quantity

```
plot(mMISE-mEmp,  type="l",xlab="p")
p=1:(M+1)
lines(p,2*p*sdw^2/N,col="red")
legend(x=4,y=0.02,c("Monte Carlo bias","Theoretical bias"),lty=1, col=c("black","red"))
```

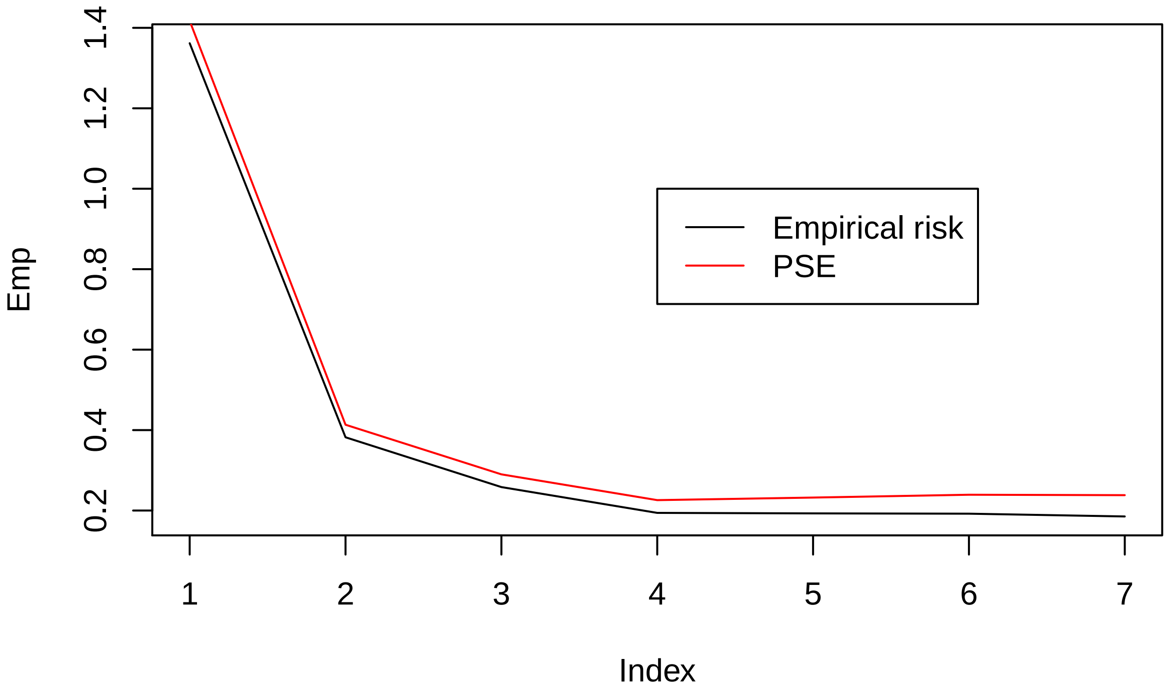


Single dataset

```r
set.seed(0)
N=50 ## number of samples

M=6  ## max order of the polynomial model
sdw=0.5 ## stanard deviation of noise

Emp<-numeric(M+1)
PSE<-numeric(M+1)

X=seq(-1,1,length.out=N)
Y=1-X+X^2-X^3+rnorm(N,sd=sdw)



  for (m in 0:M){
    DX=NULL
    for (j in 0:m){
      DX=cbind(DX,X^j)
    }

    betahat=solve(t(DX)%*%DX)%*%t(DX)%*%Y
    Yhat=DX%*%betahat
    Emp[m+1]=mean((Y-Yhat)^2)
    sdw=sd(Y-Yhat)
    PSE[m+1]=Emp[m+1]+2*sdw^2/N*(m+1)

  }

bestEmp=which.min(Emp)-1
bestPSE=which.min(PSE)-1

print(bestPSE)
```

```
## [1] 3
```

```r
plot(Emp,type="l")
lines(PSE,col="red")

legend(x=4,y=1,c("Empirical risk","PSE"),lty=1, col=c("black","red"))
```



The model degree **6** returned by minimizing the empirical risk corresponds to the highest order considered.

The model degree **3** returned by minimizing the empirical risk corresponds to the real degree of the regression function $E[\mathbf{y}|x]$.