U1

myMSE=function(lambda,pars){

res=loess(pars$Y~pars$X, enp.target=lambda)

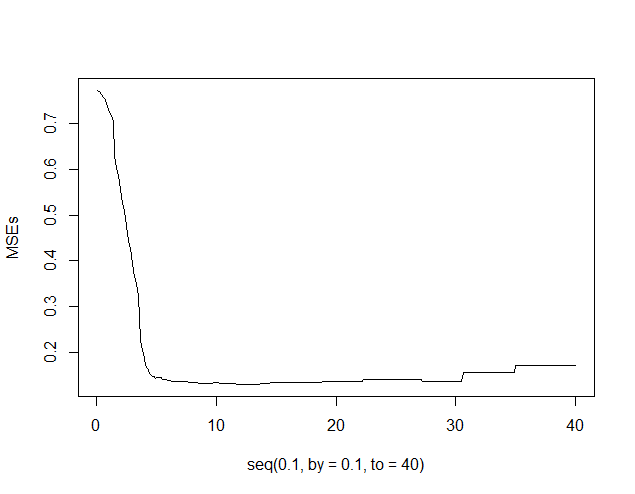
Yp=predict(res,newdata=pars$Xtest)

MSE=mean((pars$Ytest-Yp)^2)

print(MSE)

return(MSE)

}



> min(MSEs)

[1] 0.131047

> which.min(MSEs)

[1] 117

🡪optimal lambda is 11.7

$minimum

[1] 10.69361

$objective

[1] 0.1321441

18 iterations and pretty close to the optimum, but still a local minimum

> optim(30, myMSE, pars=list(X=train$Day, Y=train$LMR, Xtest=test$Day, Ytest=test$LMR), method="BFGS")

[1] 0.1380524

[1] 0.1380524

[1] 0.1380524

$par

[1] 30

$value

[1] 0.1380524

$counts

function gradient

1 1

$convergence

[1] 0

Only 3 function evaluations but we are pretty far to the global optimum

#uppgift 2

🡪 optimal estimators are sample mean and standard dev

> mu0

[1] 1.275528

> sigma0

[1] 2.016082

> optim(c(0,1), myfun, NULL, method = "CG")

$par

[1] 1.275528 2.005977

$value

[1] 211.5069

$counts

function gradient

719 101

$convergence

[1] 1

$message

NULL

> optim(c(0,1), myfun, gr=mygrad, method = "CG")

$par

[1] 1.275528 2.005976

$value

[1] 211.5069

$counts

function gradient

56 17

$convergence

[1] 0

$message

NULL

> optim(c(0,1), myfun, NULL, method = "BFGS")

$par

[1] 1.275528 2.005977

$value

[1] 211.5069

$counts

function gradient

37 15

$convergence

[1] 0

$message

NULL

> optim(c(0,1), myfun, gr=mygrad, method = "BFGS")

$par

[1] 1.275528 2.005977

$value

[1] 211.5069

$counts

function gradient

39 15

$convergence

[1] 0

$message

NULL