**AIRFOIL NOISE DATASET**

> library(splines)

> library("gam", lib.loc="~/R/win-library/3.2")

> data<- airfoil\_self\_noise

#renaming the predictor variable

> names(data) <- c("Freq","Angle", "Chord","Velocity","Suction", "Pressure")

> head(data)

Freq Angle Chord Velocity Suction Pressure

1 800 0 0.3048 71.3 0.00266337 126.201

2 1000 0 0.3048 71.3 0.00266337 125.201

3 1250 0 0.3048 71.3 0.00266337 125.951

4 1600 0 0.3048 71.3 0.00266337 127.591

5 2000 0 0.3048 71.3 0.00266337 127.461

6 2500 0 0.3048 71.3 0.00266337 125.571

#no NA values in dataset

> sum(is.na(data))

[1] 0

#plotting the correlation matrix

> cor(data)

Freq Angle Chord Velocity Suction Pressure

Freq 1.000000000 -0.27276454 -0.003660639 0.133663831 -0.230107353 -0.3907114

Angle -0.272764536 1.00000000 -0.504868150 0.058759565 0.753393785 -0.1561075

Chord -0.003660639 -0.50486815 1.000000000 0.003786629 -0.220842431 -0.2361615

Velocity 0.133663831 0.05875957 0.003786629 1.000000000 -0.003974013 0.1251028

Suction -0.230107353 0.75339378 -0.220842431 -0.003974013 1.000000000 -0.3126695

Pressure -0.390711412 -0.15610753 -0.236161512 0.125102801 -0.312669506 1.0000000

#There is high correlation between angle and suction. So, remove suction.

> data <- data[, -c(5)]

> names(data)

[1] "Freq" "Angle" "Chord" "Velocity" "Pressure"

> str(data)

'data.frame': 1503 obs. of 5 variables:

$ Freq : int 800 1000 1250 1600 2000 2500 3150 4000 5000 6300 ...

$ Angle : num 0 0 0 0 0 0 0 0 0 0 ...

$ Chord : num 0.305 0.305 0.305 0.305 0.305 ...

$ Velocity: num 71.3 71.3 71.3 71.3 71.3 71.3 71.3 71.3 71.3 71.3 ...

$ Pressure: num 126 125 126 128 127 ...

#fitting smooth spline on Freq predictor using cross validation for the degree of freedom

set.seed(100)

> smp\_size <- floor(0.5 \* nrow(data))

> train\_ind=sample(seq\_len(nrow(data)), size = smp\_size)

> data.train <- data[train\_ind, ]

> data.test <- data[-train\_ind, ]

> head(data.train)

Freq Angle Chord Velocity Pressure

463 5000 0.0 0.1524 71.3 126.457

388 4000 5.3 0.2286 39.6 115.801

830 4000 8.4 0.0508 39.6 118.955

85 800 1.5 0.3048 39.6 128.831

703 1600 12.6 0.1524 71.3 120.534

725 6300 12.6 0.1524 39.6 103.380

> timelims = range(data.train$Freq)

> time.grid = seq(from=timelims[1], to=timelims[2])

> plot(data.train$Freq,data.train$Pressure, xlim = timelims, col="darkgrey")

> title("Smooth Spline")

> train.fit=smooth.spline(data.train$Freq,data.train$Pressure, cv = TRUE)

Warning message:

In smooth.spline(data.train$Freq, data.train$Pressure, cv = TRUE) :

cross-validation with non-unique 'x' values seems doubtful

> train.fit$df

[1] 11.03061

#degree of freedom comes out to be around 11

> lines(train.fit,col="red", lwd=2)

> plot(data.train$Freq,data.train$Pressure, xlim = timelims, col="darkgrey")

> train.fit=smooth.spline(data.train$Freq,data.train$Pressure, df=9)

> train.fit=smooth.spline(data.train$Freq,data.train$Pressure, df=11)

> train.fit

Call:

smooth.spline(x = data.train$Freq, y = data.train$Pressure, df = 11)

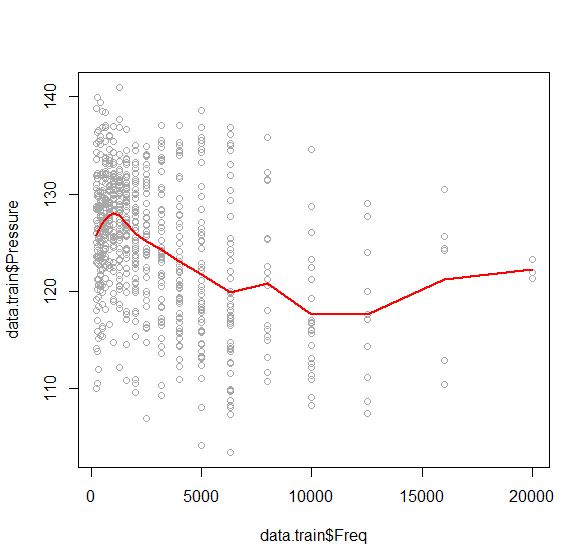
Smoothing Parameter spar= 0.6509996 lambda= 0.0002381952 (11 iterations)

Equivalent Degrees of Freedom (Df): 11.00088

Penalized Criterion: 197.6816

GCV: 38.83005

> lines(train.fit,col="red", lwd=2)



> timelims = range(data.test$Freq)

> time.grid = seq(from=timelims[1], to=timelims[2])

> plot(data.test$Freq,data.test$Pressure, xlim = timelims, col="darkgrey")

> title("Smooth Spline")

> test.fit=smooth.spline(data.test$Freq,data.test$Pressure, df=11)

> test.fit

Call:

smooth.spline(x = data.test$Freq, y = data.test$Pressure, df = 11)

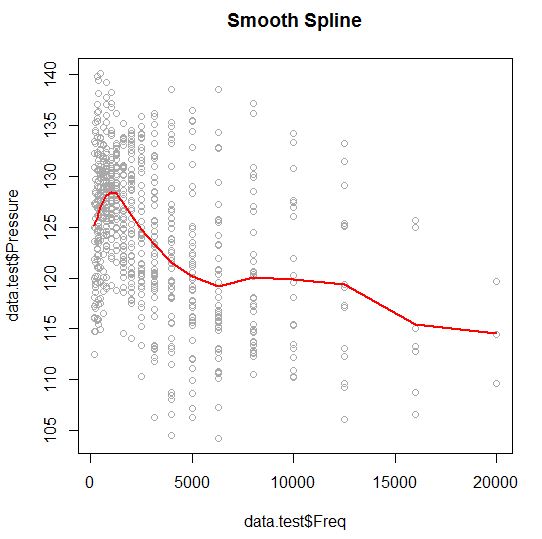
Smoothing Parameter spar= 0.6528187 lambda= 0.0002473264 (10 iterations)

Equivalent Degrees of Freedom (Df): 10.99924

Penalized Criterion: 267.8106

GCV: 38.23007

> lines(test.fit,col="red", lwd=2)



#using local regression for angle predictor

> anglelims = range(data$Angle)

> angle.grid = seq(from=anglelims[1], to=anglelims[2])

> angle=data$Angle

> pressure= data$Pressure

> plot(angle,pressure,xlim=timelims ,cex=.5,col="darkgrey ")

> title("Local Regression ")

> install.packages("bisoreg", dependencies=TRUE)

#cross validation for best model using local regression

> loess.wrapper(angle,pressure, span.vals = seq(0.3, .7, by = 0.05), folds = 5)

Call:

loess(formula = y ~ x, span = span)

Number of Observations: 1503

Equivalent Number of Parameters: 5.63

Residual Standard Error: 6.714

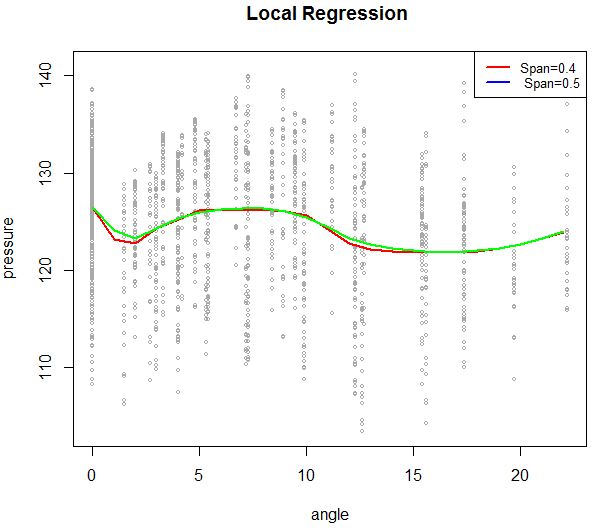
> fit=loess(pressure∼angle,span=.4,data=data)

> fit2=loess(pressure∼angle,span=.5,data=data)

> lines(angle.grid,predict(fit,data.frame(angle=angle.grid)),col="red",lwd=2)

> lines(angle.grid,predict(fit2,data.frame(angle=angle.grid)),col="green",lwd=2)

> legend("topright",legend=c("Span=0.4"," Span=0.5"), col=c("red","blue"),lty=1,lwd=2,cex=.8)

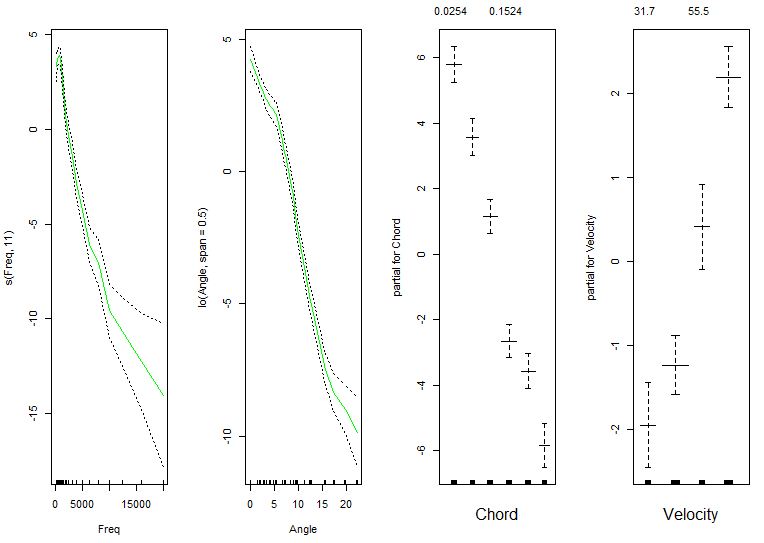


#final generalized additive model using smooth splines and local regression on Freq and Angle respectively. Chord and velocity are categorical variable.

> gam.m1=gam(Pressure ~ s(Freq,11)+lo(Angle,span=.5)+Chord+Velocity ,data=data)

> par(mfrow=c(1,4))

> plot.gam(gam.m1, se=TRUE, col="green")



This is the plot for final generalized additive model.