Handout 9: Smoothing— Some Basic Concepts

Oversmoothing and Undersmoothing and the \textit{density} Command

```r
> test<-c(rnorm(10000,10,3),rnorm(10000,20,3),rnorm(10000,30,3))
> hist(test)
> break1<-seq(min(test),max(test),length=3)
> break2<-seq(min(test),max(test),length=10)
> break3<-seq(min(test),max(test),length=50)
> break4<-seq(min(test),max(test),length=100)
> par(mfrow=c(2,2))
> hist(test,breaks=break1)
> hist(test,breaks=break2)
> hist(test,breaks=break3)
> hist(test,breaks=break4)

> t<-density(test,n=5)
> plot(t)
> t<-density(test,n=10)
> plot(t)
> t<-density(test,n=30)
> plot(t)
> t<-density(test,n=100)
> plot(t)
```
Figure 1. The Histogram of the Simulated Data. “R” Chooses an Optimal Number of Classes.
Figure 2. Histograms With Various Numbers of Bandwidths: 2, 10, 50, and 100, Clockwise.
Figure 3. The Density Estimation of the Original Data using the “density” Command.
Figure 4. The Function "density" With Different "n"s. "n is the number of equally spaced points at which the density is to be estimated".
Smoothing Splines

Example 1: The "Cars" Data

```r
> library(smooth)
> cars
   speed dist
 1     4     2
 2     4    10
 3     7     4
 4     7    22
...  ...
...  ...
46    24    70
47    24    92
48    24    93
49    24   120
50    25     85

> cars.spl <- smooth.spline(speed, dist)
> (cars.spl)
Call:
  smooth.spline(x = speed, y = dist)

  Smoothing Parameter spar= 0.7801305  lambda= 0.1112206 (11 iterations)
  Equivalent Degrees of Freedom (Df): 2.635278
  Penalized Criterion: 4187.776
  GCV: 244.1044

> ## This example has duplicate points, so avoid cv=TRUE
> lines(cars.spl, col = "blue")
> lines(smooth.spline(speed, dist, df=10), lty=2, col = "red")
> legend(5,120,c(paste("default [C.V.] => df =",round(cars.spl$df,1)), +                   "s( * , df = 10)"), col = c("blue","red"), lty = 1:2,
+            bg='bisque')
> detach()
```
Figure 5. The Smoothing Spline for the "cars" Data. Solid Line is the Default. We tend to Oversmooth as We Increase the Degrees of Freedom (Dashed Line).
data(cars) & smoothing splines

Figure 6. With Large df, We Get A Jagged Fit.
Example 2: Back to Our Simulated Data

```r
> s3 <- hist(test, breaks=break3)$counts
> s3
[1] 4 0 7 17 41 71 132 238 340 594 725 871 1044 1177 1123
[16] 1049 877 700 657 586 611 708 867 1009 1067 1154 1057 1018 78 0 717
[31] 614 591 646 727 949 1037 1074 1109 1064 857 764 507 356 218 130
[46] 51 47 14 4
> length(s3)
[1] 49
> d3 <- seq(1, range(test), length=49)
> length(d3)
[1] 49

> smooth.spline(d3, s3)
Call:
smooth.spline(x = d3, y = s3)

Smoothing Parameter spar= 0.3954784 lambda= 4.681822e-06 (13 iterations)
Equivalent Degrees of Freedom (Df): 21.20618
Penalized Criterion: 23392.77
GCV: 1483.820

> spline.here <- smooth.spline(d3, s3)

> plot(spline.here, type = "l", lwd = 3)
```
Figure 7. The Smoothing Splines for Our Normal Data.