1 Get Help

There are three ways to get help on any function or object in R

```
> help()
> help.start()
> help(range)
> help(hist)
> help(seq)
> help(rnorm)
> help(plot)
> help(density)
> help(lm)
> help(predict)
> help(data)
> help(names)
```

2 Some Exercises

Please try the following commands in R and see what outputs you obtain.

2.1 Exercise 1: Data Generalization and Visualization

```
y = rnorm(20,0,1)
mean(y)
var(y)
range(y)
hist(y)
hist(y, prob=T,labels=T)
plot(density(y))
```

2.2 Exercise 2: Linear Regression Model

```
x = seq(0,1,length.out=20)
x
z = 2*x + rnorm(20,0,0.5)
plot(x,z)
mylinear =  lm(z~x)
summary(mylinear)
zhat = predict(mylinear)
zresid =  z-zhat
```

2.3 Example 3: Multivariate Data Analysis

```
data(trees)
names(trees)
is.data.frame(trees)
# Use the variables Height and Volume to predict Girth
model1 =  lm(Girth~., data=trees)
summary(model1)
Ghat = predict(model1)
Gresid = trees$Girth-Ghat
```

2.4 Example 4: Generalized Additive Models (GAM)

The GAM model relaxes the linear assumption in the linear regression, by allowing the effect of each input variable to be arbitrarily nonlinear. The package *mgcv* is needed to run the **gam** . There is an option of choosing different basis functions by specifying **bs**

- bs="cr" for cubic regression spline (only for 1-d)
- bs="cc" for cyclic cubic regression spline (only for 1-d)
- bs="tp" for thin plate regression spline.

Smoothing is implemented by

- selecting smoothing parameters with GCV/UBRE
- using fixed degree of freedom (by specifying "fx=")

```
library(mgcv)
n<-400
sig2<-4
x0 <- runif(n, 0, 1)
x1 <- runif(n, 0, 1)
x2 <- runif(n, 0, 1)
x3 <- runif(n, 0, 1)
pi <- asin(1) * 2
f <- 2*sin(pi * x0)
f <- f+exp(2 * x1) - 3.75887
f <- f+0.2*x2^11*(10*(1-x2))^6+10*(10*x2)^3*(1-x2)^10-1.396
e <- rnorm(n, 0, sqrt(abs(sig2)))
y <- f + e
b <- gam(y~s(x0)+s(x1)+s(x2)+s(x3))
summary(b)
plot(b,pages=1)
lm <- lm(y~x0+x1+x2+x3)
anova(lm,b)
```