```
### Polynomial Regression
### Use the “strongx” data.
### For "strongx" the forward and backward procedures pick the same d,
### but it is possible these two would pick different d values.

### Forward
> g1 = lm(crossx ~ energy, weights = sd^-2, strongx)
> round(summary(g1)$coef, dig=3)
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  148.473      8.079  18.378        0
energy       530.835     47.550  11.164        0
> g2 = lm(crossx ~ energy + I(energy^2), weights = sd^-2, strongx)
> round(summary(g2)$coef, dig=3)
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  183.830      6.459  28.461    0.000
energy         0.971     85.369   0.011    0.991
I(energy^2) 1597.505    250.587   6.375    0.000
> g3 =  lm(crossx ~ energy + I(energy^2) + I(energy^3) , weights = sd^-2,
strongx)
> round(summary(g3)$coef, dig=3)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   204.992     12.826  15.983    0.000
energy       -472.866    268.862  -1.759    0.129
I(energy^2)  4504.475   1600.789   2.814    0.031
I(energy^3) -5220.597   2848.373  -1.833    0.117
```

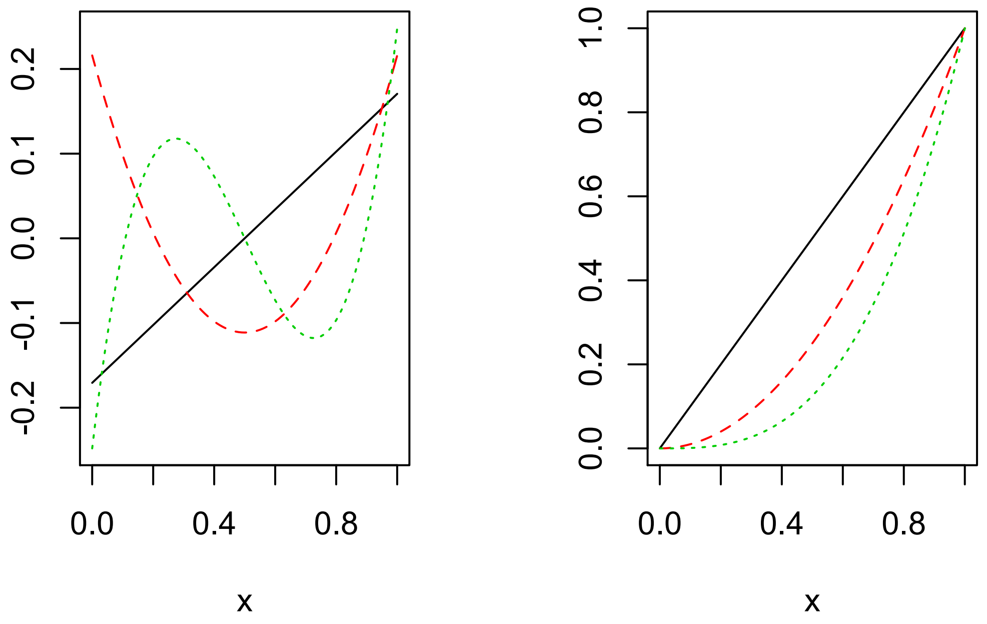
```
### Backward
> round(summary(lm(crossx ~ energy + I(energy^2) + I(energy^3) + I(energy^4)
+ I(energy^5), weights = sd^-2, strongx))$coef, dig=3)
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     162.505     82.098   1.979    0.119
energy         1104.881   2876.788   0.384    0.720
I(energy^2)  -16585.936  36586.764  -0.453    0.674
I(energy^3)  123023.612 214497.193   0.574    0.597
I(energy^4) -359391.947 586216.560  -0.613    0.573
I(energy^5)  375585.481 602771.524   0.623    0.567
> round(summary(lm(crossx ~ energy + I(energy^2) + I(energy^3) + I(energy^4),
weights = sd^-2, strongx))$coef, dig=3)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   208.663     33.156   6.293    0.001
energy       -579.586    921.559  -0.629    0.557
I(energy^2)  5511.600   8426.036   0.654    0.542
I(energy^3) -9010.875  31174.930  -0.289    0.784
I(energy^4)  4902.233  40118.865   0.122    0.908
> round(summary(lm(crossx ~ energy + I(energy^2) + I(energy^3) , weights =
sd^-2, strongx))$coef, dig=3)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   204.992     12.826  15.983    0.000
energy       -472.866    268.862  -1.759    0.129
I(energy^2)  4504.475   1600.789   2.814    0.031
I(energy^3) -5220.597   2848.373  -1.833    0.117
```

```
> round(summary(lm(crossx ~ energy + I(energy^2) , weights = sd^-2,
strongx))$coef, dig=3)
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  183.830      6.459  28.461    0.000
energy         0.971     85.369   0.011    0.991
I(energy^2) 1597.505    250.587   6.375    0.000

### Use the orthogonal polynomial bases
> round(summary(lm(crossx~poly(energy, 2),weights = sd^-2, strongx))$coef,
dig=3)
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       255.342      1.728 147.800        0
poly(energy, 2)1  168.081      6.673  25.190        0
poly(energy, 2)2   38.655      6.063   6.375        0
```

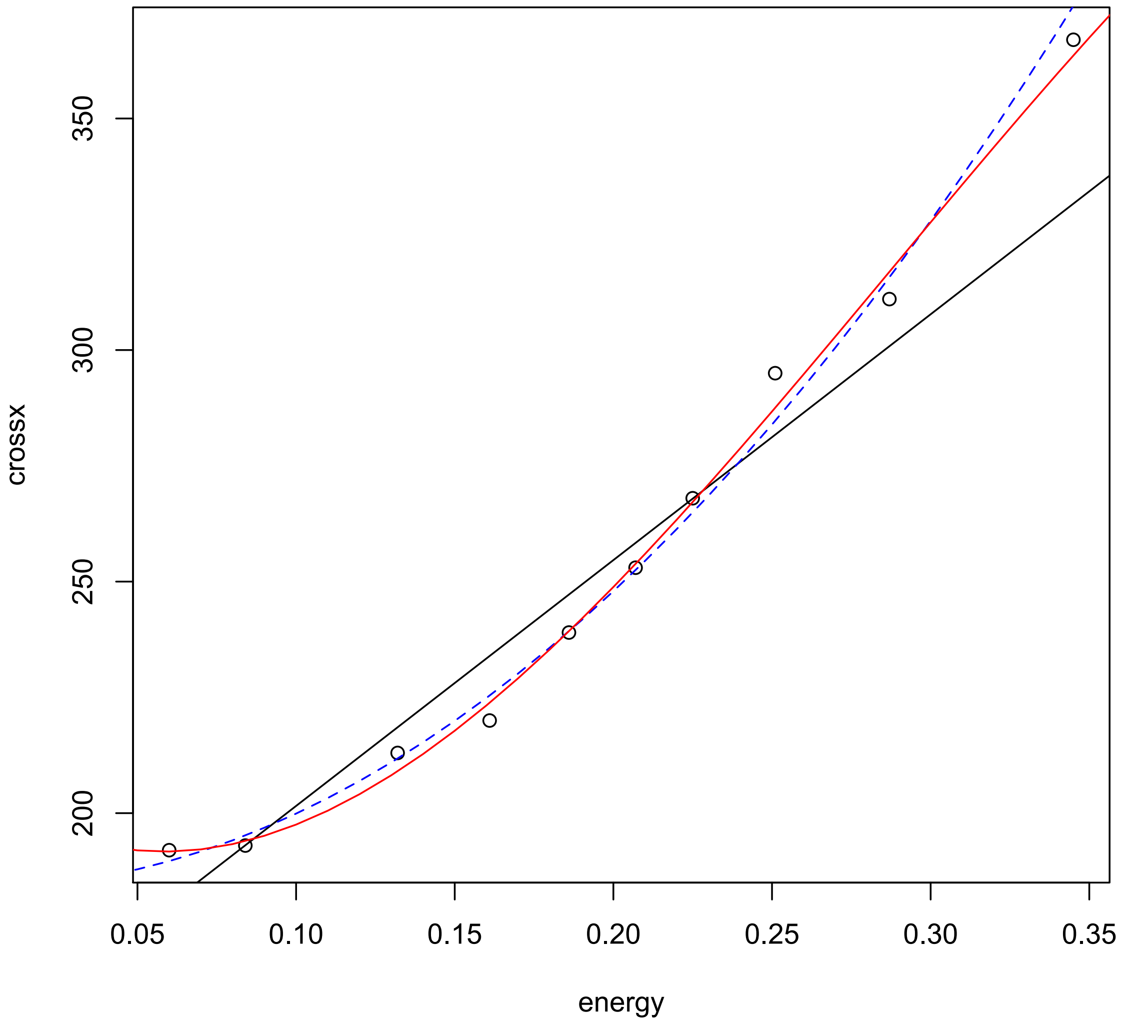
```
> round(summary(lm(crossx~poly(energy, 3),weights = sd^-2, strongx))$coef,
dig=3)
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       254.687      1.536 165.792    0.000
poly(energy, 3)1  163.660      6.254  26.167    0.000
poly(energy, 3)2   32.145      6.333   5.076    0.002
poly(energy, 3)3   -8.940      4.878  -1.833    0.117
```

```
> round(summary(lm(crossx ~ energy + I(energy^2) + I(energy^3), weights =
sd^-2, strongx))$coef, dig=3)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   204.992     12.826  15.983    0.000
energy       -472.866    268.862  -1.759    0.129
I(energy^2)  4504.475   1600.789   2.814    0.031
I(energy^3) -5220.597   2848.373  -1.833    0.117
```

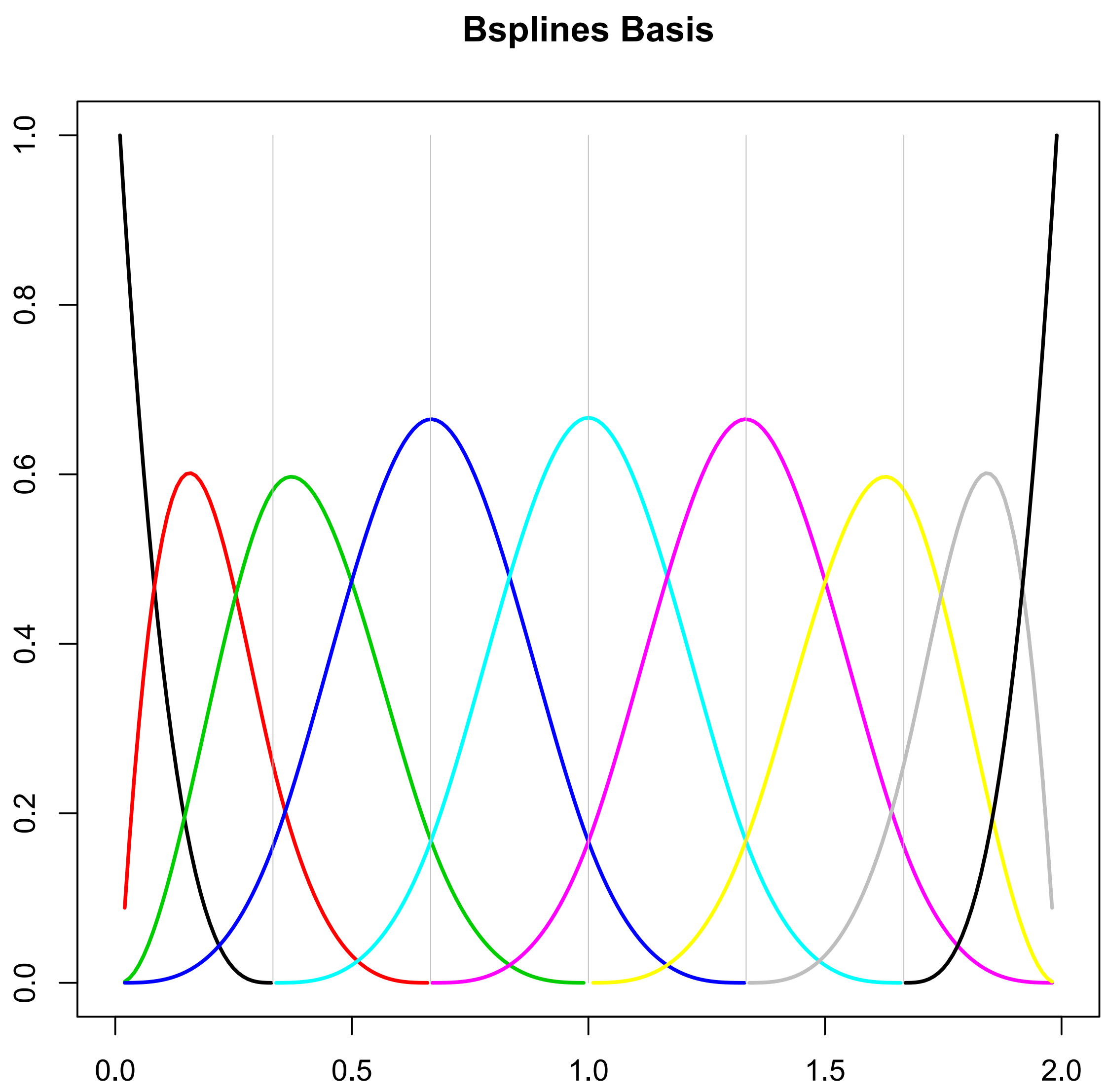
```
par(mfrow=c(1,2))
x=seq(0, 1, 0.01); out = poly(x, 3)
plot(range(x), range(out), type="n", xlab="x", ylab="")
for(i in 1:3) lines(x, out[,i], lty=i, col=i)
out = cbind(x, x^2, x^3);
plot(range(x), range(out), type="n",xlab="x", ylab="")
for(i in 1:3) lines(x, out[,i], lty=i, col=i)
```

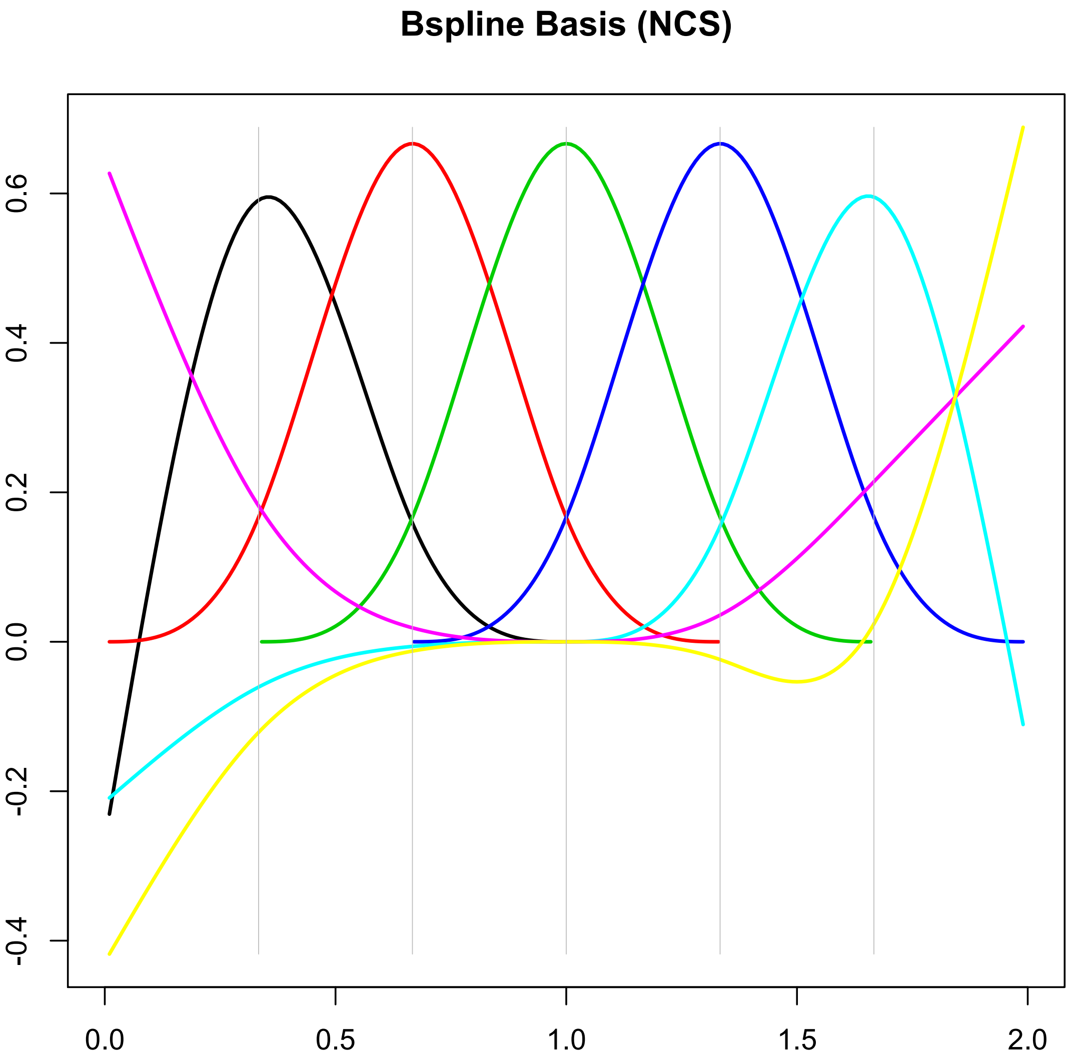


```
> g1 = lm(crossx ~ energy, weights = sd^-2, data=strongx)
> g2 = lm(crossx ~ energy + I(energy^2), weights = sd^-2,data=strongx)
> g3 = lm(crossx ~ energy + I(energy^2) + I(energy^3), weights = sd^-2,
data=strongx)
> newx=data.frame(energy=seq(0.02, 0.4, 0.01));
> plot(energy, crossx, xlab="energy", ylab="crossx")
> lines(newx$energy, predict(g1, newx), col="black", lty=1);
> lines(newx$energy,predict(g2, newx), col="blue", lty=2);
> lines(newx$energy,predict(g3, newx), col="red", lty=1);

> anova(g1,g2)

Model 1: crossx ~ energy
Model 2: crossx ~ energy + I(energy^2)
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1      8 21.9526
2      7  3.2255  1    18.727 40.641 0.0003761 ***
> anova(g2,g3)
Analysis of Variance Table

Model 1: crossx ~ energy + I(energy^2)
Model 2: crossx ~ energy + I(energy^2) + I(energy^3)
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1      7 3.2255
2      6 2.0678  1    1.1577 3.3593 0.1165
```

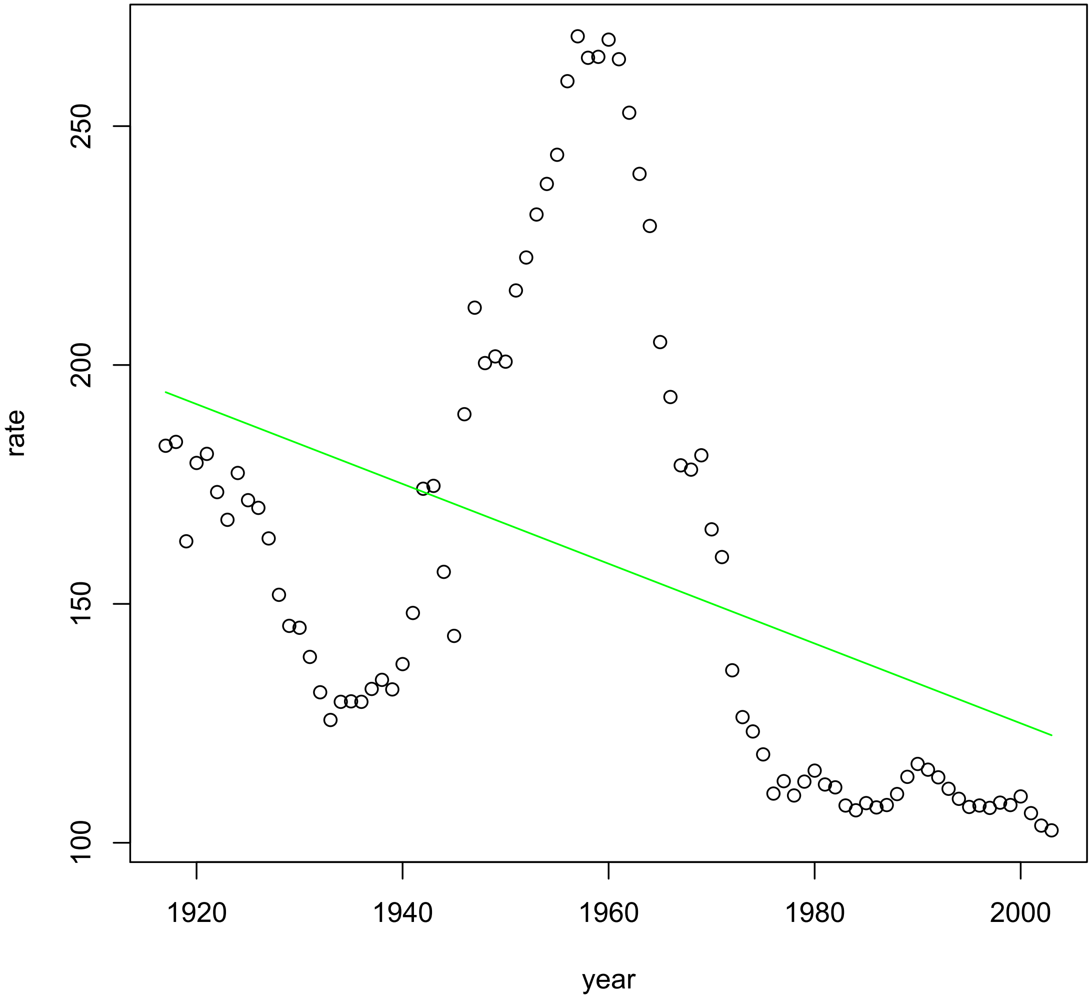

crossx
200
250
300
350
0.05
0.10
0.15
0.20
0.25
0.30
0.35
energy

```
> library(splines);
> help(bs)
> help(ns)
>
> x=(1:199)/100;
> m=5;myknots=2*(1:m)/(m+1)
>
> X=bs(x,knots=myknots, intercept=TRUE)
> plot(c(0,2), range(X), type="n", xlab="", ylab="")
> title("Bsplines Basis")
> for(i in 1:dim(X)[2]){
+   tmp=X[,i];
+   lines(x[tmp!=0], tmp[tmp!=0], col=i, lwd=2)}
> for(i in 1:m){
+   tmp=myknots[i];
+   lines(c(tmp,tmp), range(X), col="gray", lwd=0.5)}
> n=dim(X)[1]
> X=ns(x, knots=myknots, Boundary.knots=c(0,2), intercept=TRUE)
> plot(c(0,2), range(X), type="n", xlab="", ylab="")
> title("Bspline Basis (NCS)")

> for(i in 1:dim(X)[2]){
+   tmp=X[,i];
+   lines(x[tmp!=0], tmp[tmp!=0], col=i, lwd=2)}
> for(i in 1:dim(X)[2]){
+   tmp=myknots[i];
+   lines(c(tmp,tmp), range(X), col="gray", lwd=0.5)}
```

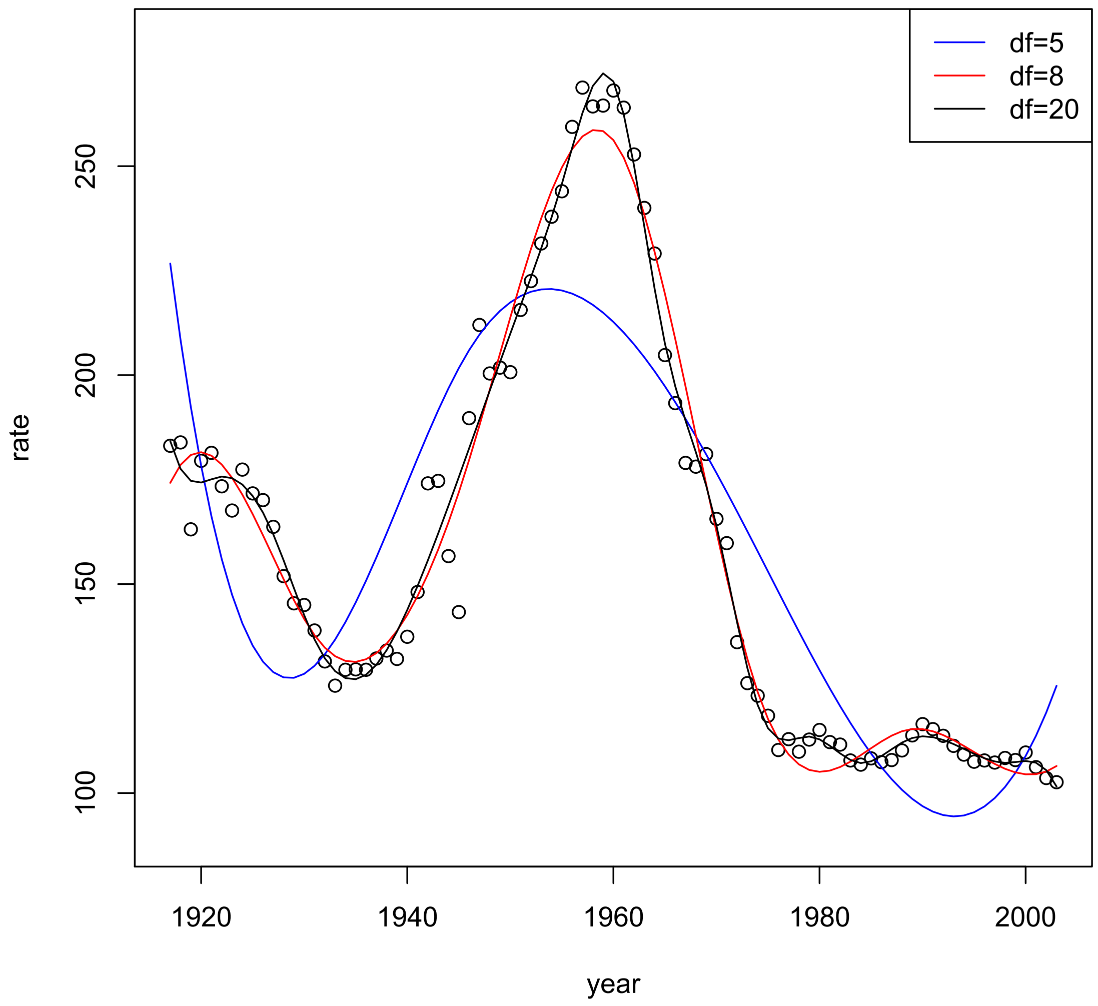

Bsplines Basis
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.5
1.0
1.5
2.0


Bspline Basis (NCS)
0.6
0.4
0.2
0.0
-0.2
-0.4
0.0
0.5
1.0
1.5
2.0

```
> ################################################################
> # An example: Birthrates data
> # This dataset lists the number of live births per 10,000
> # 23-year-old women in the United States between 1917 and 2003.
> ################################################################

> source("birthrates.txt");
> year=birthrates[,1]; rate=birthrates[,2];
> plot(year, rate);
> lines(year, predict(lm(rate~year)), col="green");
>
```


rate
100
150
200
250
year
1920
1940
1960
1980
2000

```
> plot(year, rate, ylim=c(90,280));
> lines(year, predict(lm(rate~bs(year, df=5))), col="blue");
> lines(year, predict(lm(rate~bs(year, df=8))), col="red");
> lines(year, predict(lm(rate~bs(year, df=20))), col="black");
> legend("topright", lty=rep(1,3),  col=c("blue", "red",  "black"),
legend=c("df=5", "df=8", "df=20"))
```


df=5
df=8
df=20
rate
250
200
150
100
year
1920
1940
1960
1980
2000