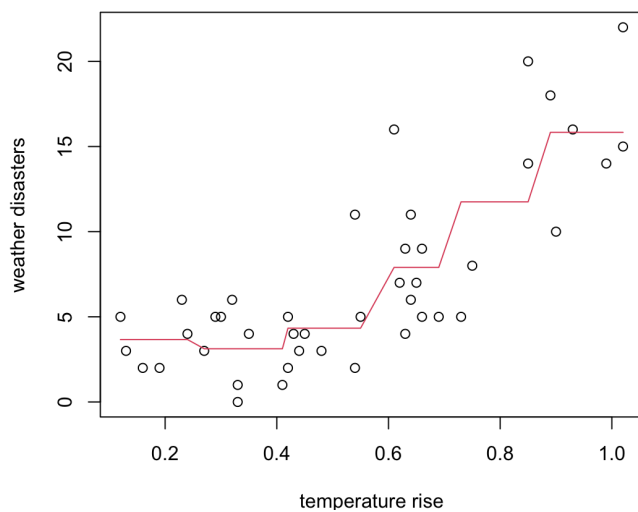
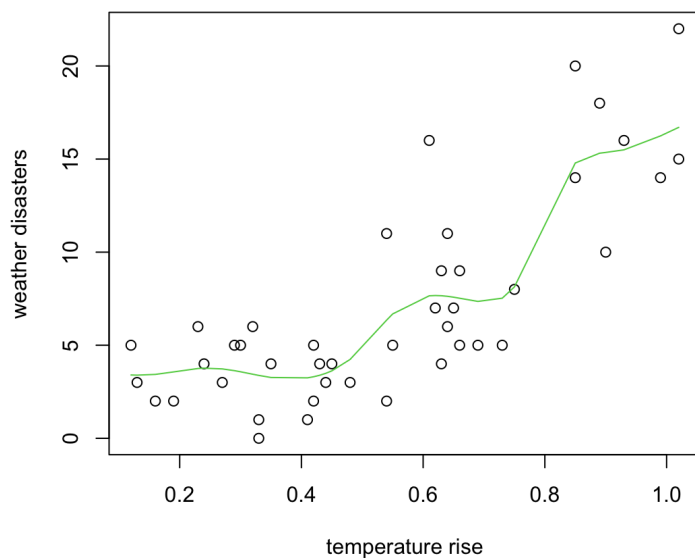


- Create a function that takes the output of each smoother in smoother.pck and calculates its press statistic. Simplest way to do it, put it in each smoother.
- Apply each to the NOAA.new data using the parameters given
- Now create a function that uses a greedy random search to find better values of the adjustable parameters, based on minimizing PRESS

```
>plot(NOAA1[,3],NOAA1[,2],xlab="temperature rise",ylab="rate of billion dollar
weather disasters")
>dumb<-bin.mean(NOAA1[,3],NOAA1[,2],6)
> dumb$press
[1] 418.0708
```



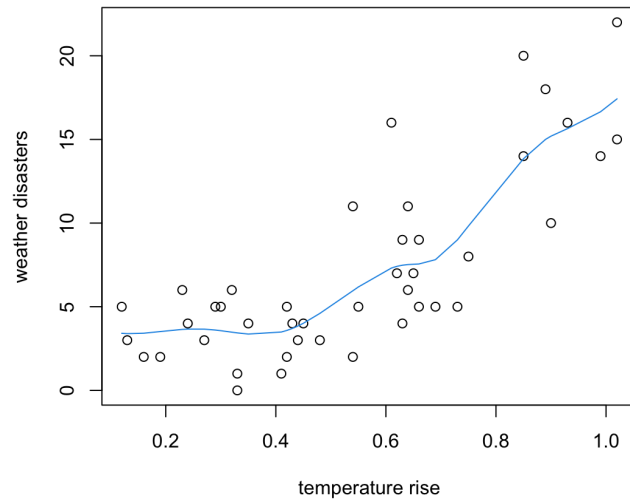
```
>dumg<-gauss.mean(NOAA1[,3],NOAA1[,2],.063)
> dumg$press
[1] 443.7311
```



```
>gauss.reg(NOAA1[,3],NOAA1[,2],.078,do.plot=T)
```

```
$press
```

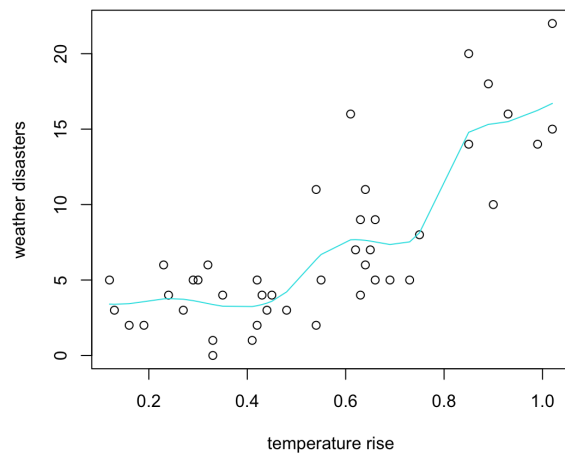
```
[1] 489.4819
```



```
>gauss.mean.trunc(NOAA1[,3],NOAA1[,2],.063,20,do.plot=T)
```

```
$press
```

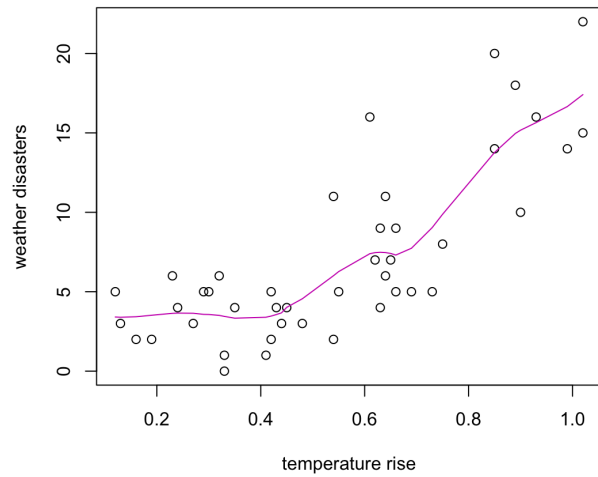
```
[1] 443.5786
```



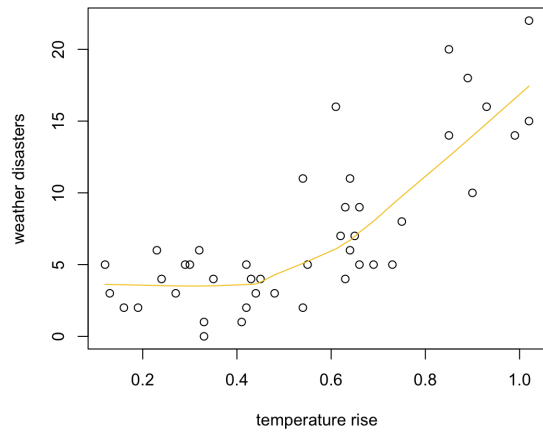
```
>gauss.reg.trunc(NOAA1[,3],NOAA1[,2],.08,17,do.plot=T)
```

```
$press
```

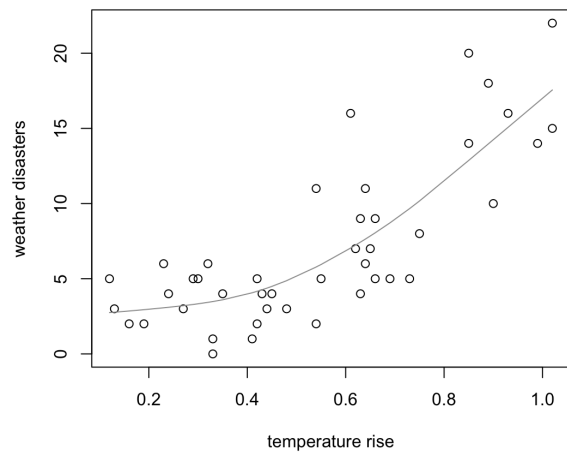
```
[1] 490.8155
```



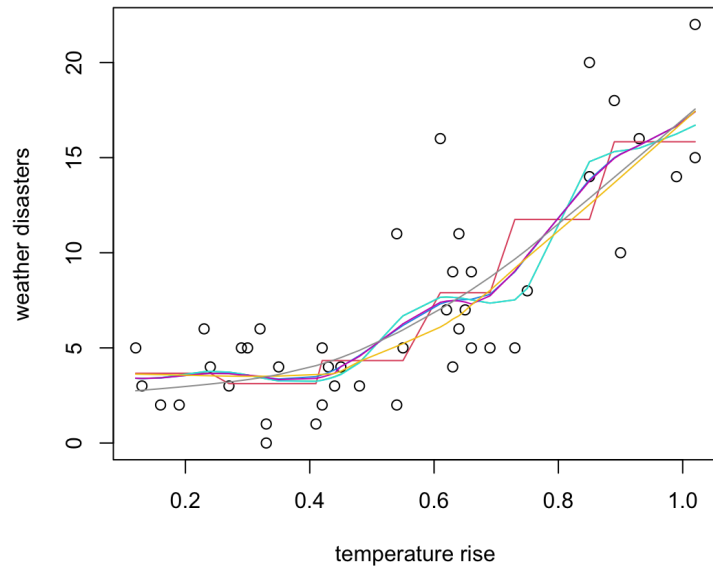
```
> lines(lowess(NOAA1[,3],NOAA1[,2]),col=7)
```



```
> lines(smooth.spline(NOAA1[,3],NOAA1[,2]),col=8)
```



```
> (smooth.spline(NOAA1[,3],NOAA1[,2]))$df  
[1] 3.314556  
ALL LINES:
```



Now create functions that uses a greedy random search to find better values of the adjustable parameters, based on minimizing PRESS

```
greedy<-function(func, data, xcol, ycol, theta){
  #press with original paramaters
  press0<- func(data[,xcol],data[,ycol],theta,F)$press
  #stores original press
  press00<-press0
  inc<- 0
  #stores original parameter
  theta0<-theta
  #loop runs up to 100 times with lowest current press val
  while(inc<100){
    epsilon<- rnorm(1,0,.01)
    #random value to the parameter lambda
    theta1 <- (theta+epsilon)
    #recalculate press statistic and store
    press1<- func(data[,xcol],data[,ycol],theta1)$press
    if (press1<press0){
      press0=press1
      inc=0
      theta=theta1
    }
    else{
      inc<-inc+1
    }
  }
}
```

```

}
list(theta = theta, theta0=theta0, press=press0, press0=press00)
print(list(theta = theta, theta0=theta0, press=press0, press0=press00))
}

```

```

greedytrunc<- function(func, data, xcol, ycol, theta,nnn){
  #press with original paramaters
  press0<- func(data[,xcol],data[,ycol],theta,nnn,F)$press
  #stores original press
  press00<-press0
  #stores original nnn value
  nnn0<-nnn
  inc<- 0
  #stores original parameter
  theta0<-theta
  #loop runs up to 100 times with lowest current press val
  while(inc<100){
    epsilon<- rnorm(1,0,.01)
    epsilon1<- sample(c(-1,0,1),1)
    #add random value epsilon to the parameter lambda
    theta1 <- (theta+epsilon)
    #add random value epsilon1 to parameter nnn
    nnn1<- nnn+epsilon1
    #recalculate press statistic and store
    press1<- func(data[,xcol],data[,ycol],theta1,nnn1)$press
    if (press1<press0){
      press0=press1
      inc=0
      theta=theta1
      nnn=nnn1
    }
    else{
      inc<-inc+1
    }
  }
  list(theta = theta, theta0=theta0,nnn0=nnn0,nnn=nnn, press=press0, press0=press00)
  print(list(theta = theta, theta0=theta0,nnn0=nnn0,nnn=nnn, press=press0,
press0=press00))
}

```

```

greedybin<- function(func, data, xcol, ycol,nbin){
  #press with original paramaters
  press0<- func(data[,xcol],data[,ycol],nbin)$press
  #stores original press
  press00<-press0
  #stores original nbin value
  nbin0<-nbin
  nbin00<-nbin
  inc<- 0
  #loop runs up to 100 times with lowest current press val
  while(inc<100){
    epsilon<- sample(-2:2,1)
    #add random value epsilon to parameter nbin
    nbin1<- nbin0+epsilon
    while(nbin1>9||nbin1<1){
      epsilon<-sample(-2:2,1)
      nbin1<- nbin0+epsilon
    }
    #recalculate press statistic and store
    press1<- func(data[,xcol],data[,ycol],nbin1)$press
    if (press1<press0){
      press0=press1
      nbin0=nbin1
    }
    else{
      inc<-inc+1
    }
  }
  list(nbin0=nbin00, nbin=nbin0, press=press0,press0=press00)
  print(list(nbin0=nbin00, nbin=nbin0, press=press0,press0=press00))
}

```

```

> bin<-greedybin(bin.mean,NOAA1,3,3,6)

```

```

$nbins

```

```

[1] 5

```

```

$press

```

```

[1] 0.4253857

```

```

$nbins0

```

```

[1] 6

```

```

$press0

```

```
[1] 0.5203199
```

```
> mean<-greedy(gauss.mean,NOAA1,3,2,.063)
```

```
$theta
```

```
[1] 0.06892099
```

```
$press
```

```
[1] 442.721
```

```
$theta0
```

```
[1] 0.063
```

```
$press0
```

```
[1] 443.7311
```

```
> reg<-greedy(gauss.reg,NOAA1,3,2,.078)
```

```
$theta
```

```
[1] 0.05657128
```

```
$press
```

```
[1] 484.1314
```

```
$theta0
```

```
[1] 0.078
```

```
$press0
```

```
[1] 489.4819
```

```
> meantrunc<-greedytrunc(gauss.mean.trunc,NOAA1,3,2,.063,20)
```

```
$theta
```

```
[1] 0.07062843
```

```
$nnn
```

```
[1] 20
```

```
$press
```

```
[1] 442.3098
```

```
$theta0
```

```
[1] 0.063
```

```
$nnn0
```

```
[1] 20
```

```
$press0
```

```
[1] 443.5786
```

```
> regtrunc<-greedytrunc(gauss.reg.trunc,NOAA1,3,2,.08,17)
```

```
$theta
```

```
[1] 0.05839566
```

```
$theta0
```

```
[1] 0.08
```

```
$nnn0
```

```
[1] 17
```

```
$nnn
```

```
[1] 16
```

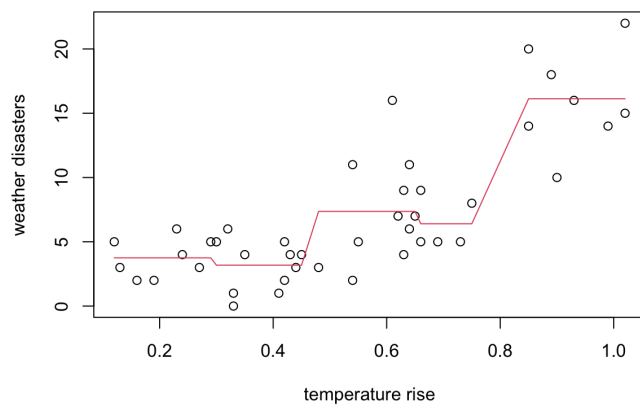
```
$press
```

```
[1] 484.9312
```

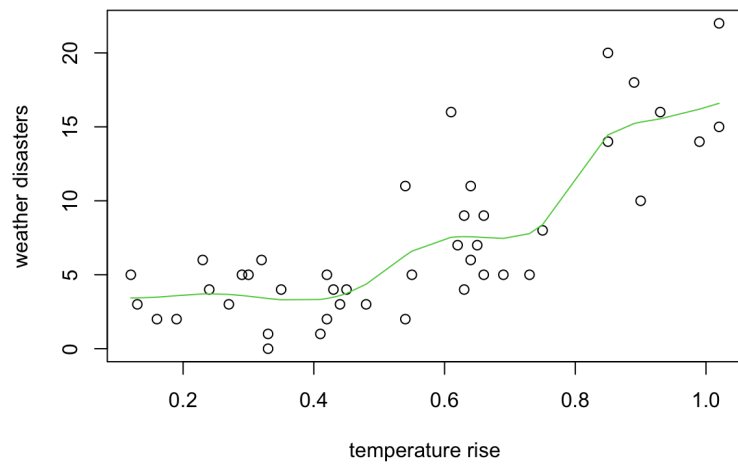
```
$press0
```

```
[1] 490.8155
```

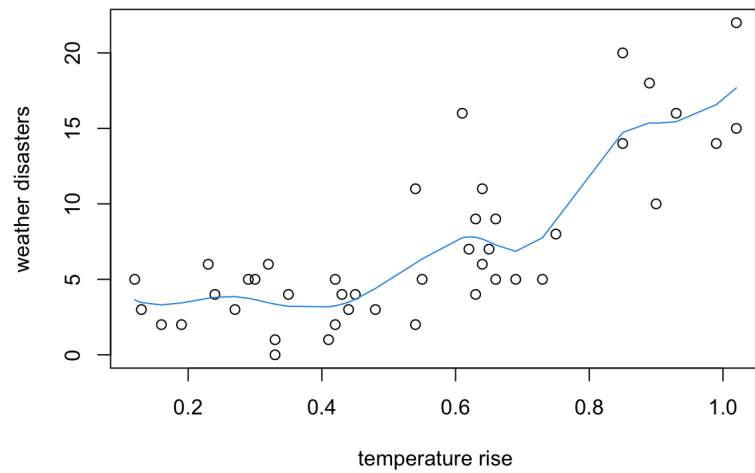
```
> newb<-bin.mean(NOAA1[,3],NOAA1[,2],bin$nbins)
```



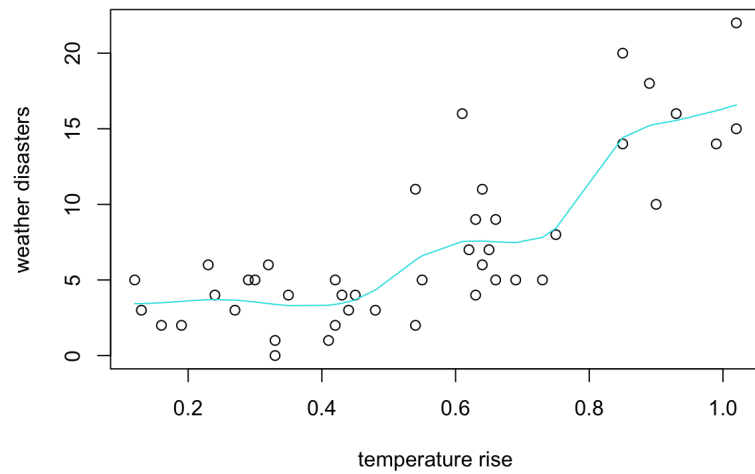
```
> newg<-gauss.mean(NOAA1[,3],NOAA1[,2],mean$theta)
```



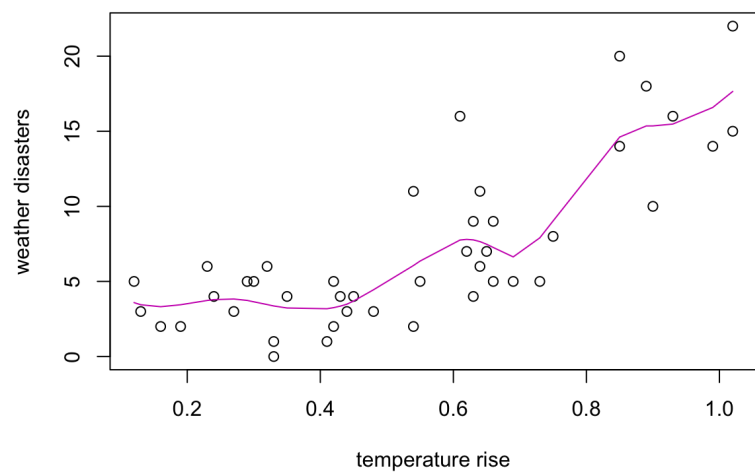
```
> gauss.reg(NOAA1[,3],NOAA1[,2],reg$theta,do.plot=T)
```

```
> gauss.mean.trunc(NOAA1[,3],NOAA1[,2],meantrunc$theta,meantrunc$nnn,do.plot=T)
```



```
> gauss.reg.trunc(NOAA1[,3],NOAA1[,2],regtrunc$theta,regtrunc$nnn,do.plot=T)
```



All lines:

