Линейная непараметрическая регрессия

ЦМФ

Непараметрическая регрессия

$$y_t = f\left(x_{1,t}, \dots, x_{d,t}\right) + \varepsilon_t$$

Ядерная оценка Надарая – Ватсона

$$\hat{f}(x_1, \dots, x_d) = \frac{\sum_{t=1}^{T} K\left(\frac{x_1 - x_{1,t}}{h_1}, \dots, \frac{x_d - x_{d,t}}{h_d}\right) y_t}{\sum_{t=1}^{T} K\left(\frac{x_1 - x_{1,t}}{h_1}, \dots, \frac{x_d - x_{d,t}}{h_d}\right)},$$

$K(u_1, \dots, u_d)$ — ядерная функция

Непараметрическая регрессия в R

расчёт величины h

```
library(np)
bw <- npregbw(ozone ~ rad, ckertype="gaussian",
bwtype="fixed", data=data.frame(ozone=t.ozone,rad=t.rad))
h <- bw$bw
```

ядро и функция Надарая – Уотсона

```
kern <- function(x) exp(-(x^2/2))/sqrt(2*pi)

NW <- function(x, x.dat, y.dat, h) {
  K1 <- K2 <- 0
  N <- length(y.dat)
  for (i in 1:N) {
    K1 <- K1 + kern((x-x.dat[i])/h)*y.dat[i]
    K2 <- K2 + kern((x-x.dat[i])/h)
  }
  K1 / K2
}
```

График оценки

```
plot(t.rad,t.ozone,pch=16)
z <- order(t.rad)
lines(t.rad[z],NW(t.rad,t.rad,t.ozone,h)[z],col="blue")
```

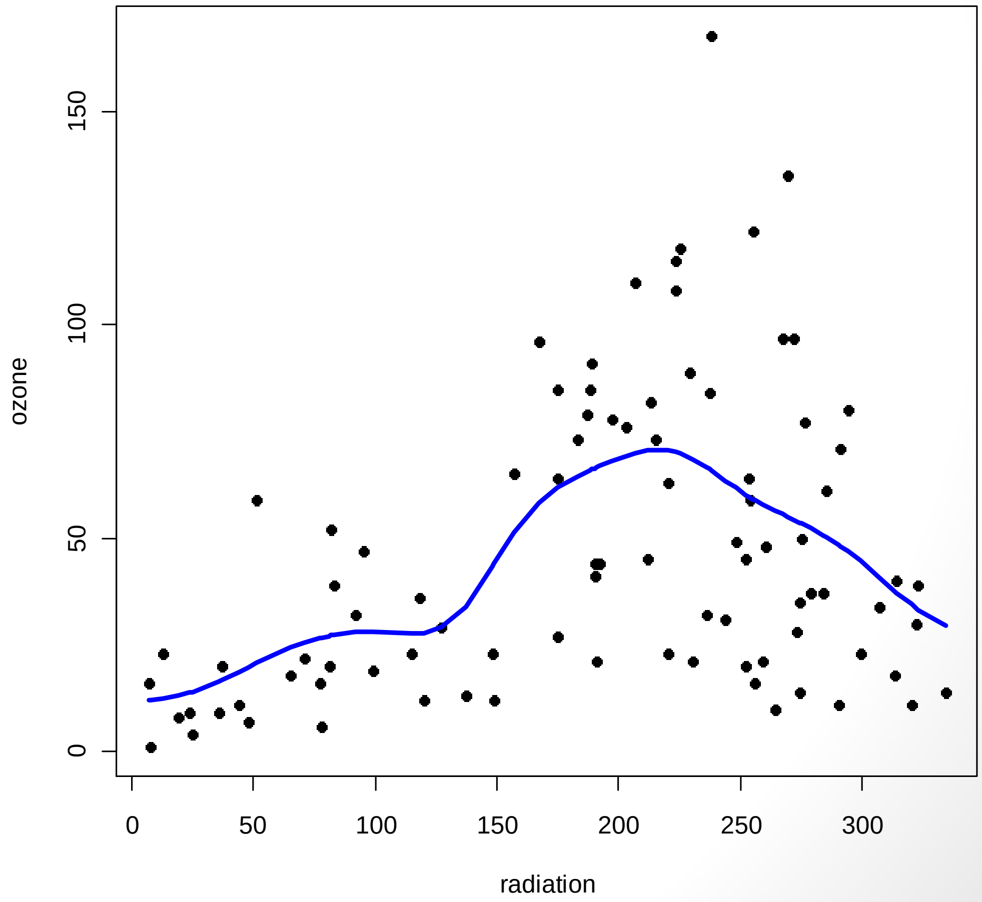
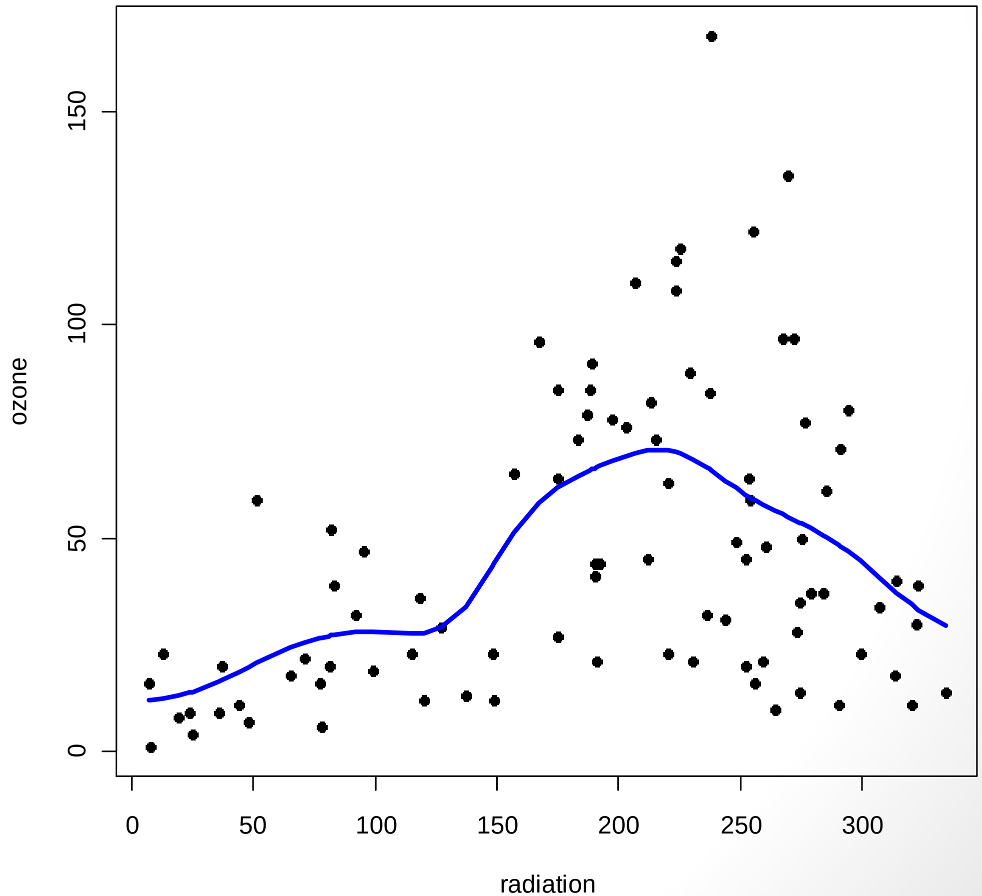


График оценки (другой вариант)

```
plot(t.rad,t.ozone,pch=16)
fit.npar <- npreg(bw)
ozone.hat <- predict(fit.npar)
lines(t.rad[z],ozone.hat[z],col="blue")
```



Построение прогноза, $\sigma^2 = const$

Метод бутстрапа

$\hat{\varepsilon}_t = e_t = y_t - \hat{f}(x_t)$ — оценки ошибок регрессии

$\widehat{\sigma^2} = s^2 = \frac{1}{T-1}\sum_{t=1}^{T}(e_t - \bar{e})^2$ — оценка дисперсии ошибок

$g = \left(\frac{4}{3T}\right)^{\frac{1}{5}} s$ — сглаживающий множитель

$e_i^* = e_{I_i} + g\xi_i,\ i \in \{1; \dots; b\},$

$I_i \sim \begin{pmatrix} 1 & \dots & T \\ \frac{1}{T} & \dots & \frac{1}{T} \end{pmatrix},\ \xi_i \sim N(0; 1)$

$e_{(i)}^* = \left(e_j^* \mid Rank\left(e_j^*\right) = i\right)$

$\left(\hat{f}(x_{T+1}) + e^*_{\left(\frac{\alpha}{2}b\right)}; \hat{f}(x_{T+1}) + e^*_{\left(\left(1-\frac{\alpha}{2}\right)b\right)}\right)$ — прогнозный интервал

Построение прогноза в R

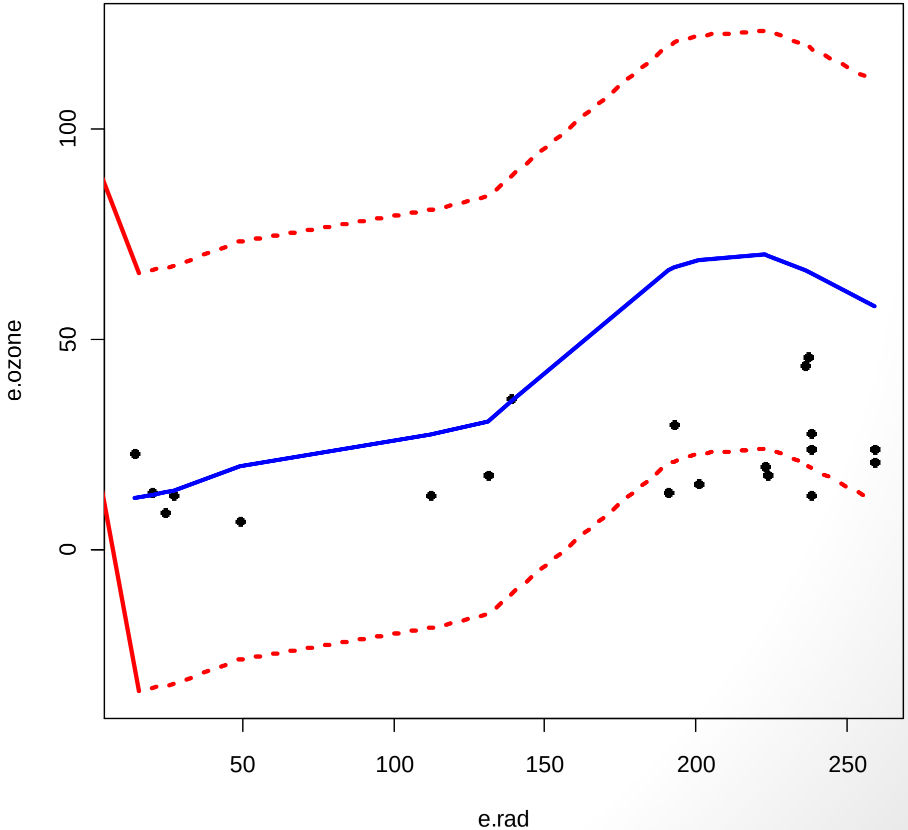
```
e <- t.ozone - ozone.hat
s2 <- var(e)
g <- (4/(3*T))^(1/5)*sqrt(s2)
```

```
# симулированные значения ошибок
b <- 10^4
e.star <- e[sample(1:T,size=b,replace=TRUE)]+g*rnorm(b)
e.star <- sort(e.star)
```

```
# прогноз и доверительные границы
alpha <- 0.1
y <- predict(fit.npar,newdata=data.frame(rad=e.rad))
bottom <- y + e.star[alpha/2*b]
top <- y + e.star[(1-alpha/2)*b]
```

Построение прогноза в R

```
z <- order(e.rad)
plot(e.rad,e.ozone,ylim=range(c(top,bottom)))
lines(e.rad[z],y[z],col="blue",lwd=2)
lines(e.rad[z],top[z],col="red",lty="dashed")
lines(e.rad[z],bottom[z],col="red",lty="dashed")
```



Построение прогноза, $\sigma^2 = \sigma^2(z)$

$s_t^2 = (e_t - \bar{e})^2$ — оценки условной дисперсии остатков

Рассмотрим зависимость условной дисперсии от экзогенных переменных:

$s_t^2 = f_s(z_t) + \eta_t$

$\widehat{s_t^2} = \hat{f}_s(z_t)$ — оценка условной дисперсии ошибок

$g_{T+1} = \left(\frac{4}{3T}\right)^{\frac{1}{5}} \hat{s}_{T+1}$ — сглаживающий множитель

Далее мы методом бутстрапа генерируем значения ошибок e_i^* и вычисляем прогнозный интервал аналогично случаю $\sigma^2 = const$

Построение прогноза в R

моделирование условной дисперсии

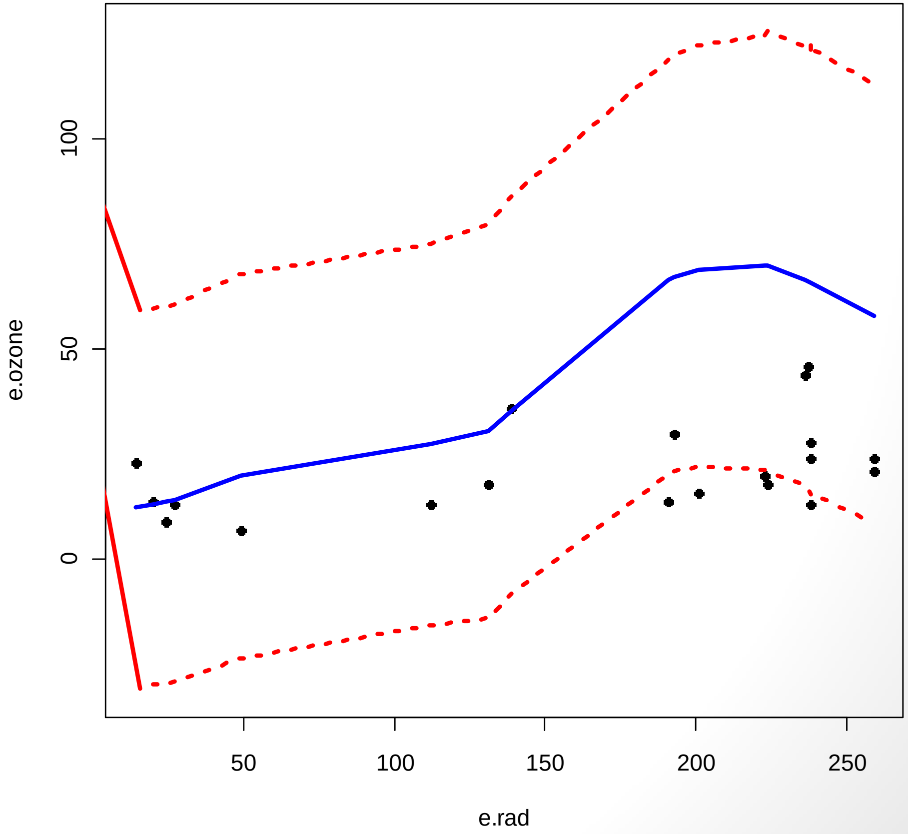
```
s2 <- (e-mean(e))^2
h.res <- npregbw(s2 ~ rad, ckertype="gaussian",
bwtype="fixed", data=data.frame(rad=t.rad))
res.npar <- npreg(h.res)
```

метод бутстрапа

```
b <- 10^4; alpha <- 0.1
top <- bottom <- numeric(E)
y <- predict(fit.npar,newdata=data.frame(rad=e.rad))

for (i in 1:E) {
  s2.hat <- predict(res.npar,newdata=data.frame(rad=e.rad[i]))
  g <- (4/(3*T))^(1/5)*sqrt(s2.hat)

  e.star <- e[sample(1:T,size=b,replace=TRUE)]+g*rnorm(b)
  e.star <- sort(e.star)

  bottom[i] <- y[i] + e.star[alpha/2*b]
  top[i] <- y[i] + e.star[(1-alpha/2)*b]
}
```

Построение прогноза в R

```
z <- order(e.rad)
plot(e.rad,e.ozone,ylim=range(c(top,bottom)))
lines(e.rad[z],y[z],col="blue",lwd=2)
lines(e.rad[z],top[z],col="red",lty="dashed")
lines(e.rad[z],bottom[z],col="red",lty="dashed")
```



Непараметрическая регрессия, двумерный случай

$$\hat{f}(x_1, \dots, x_d) = \frac{\sum_{t=1}^{T} K\left(\frac{x_1 - x_{1,t}}{h_1}, \dots, \frac{x_d - x_{d,t}}{h_d}\right) y_t}{\sum_{t=1}^{T} K\left(\frac{x_1 - x_{1,t}}{h_1}, \dots, \frac{x_d - x_{d,t}}{h_d}\right)}$$

\# ядерная оценка Надарая – Ватсона

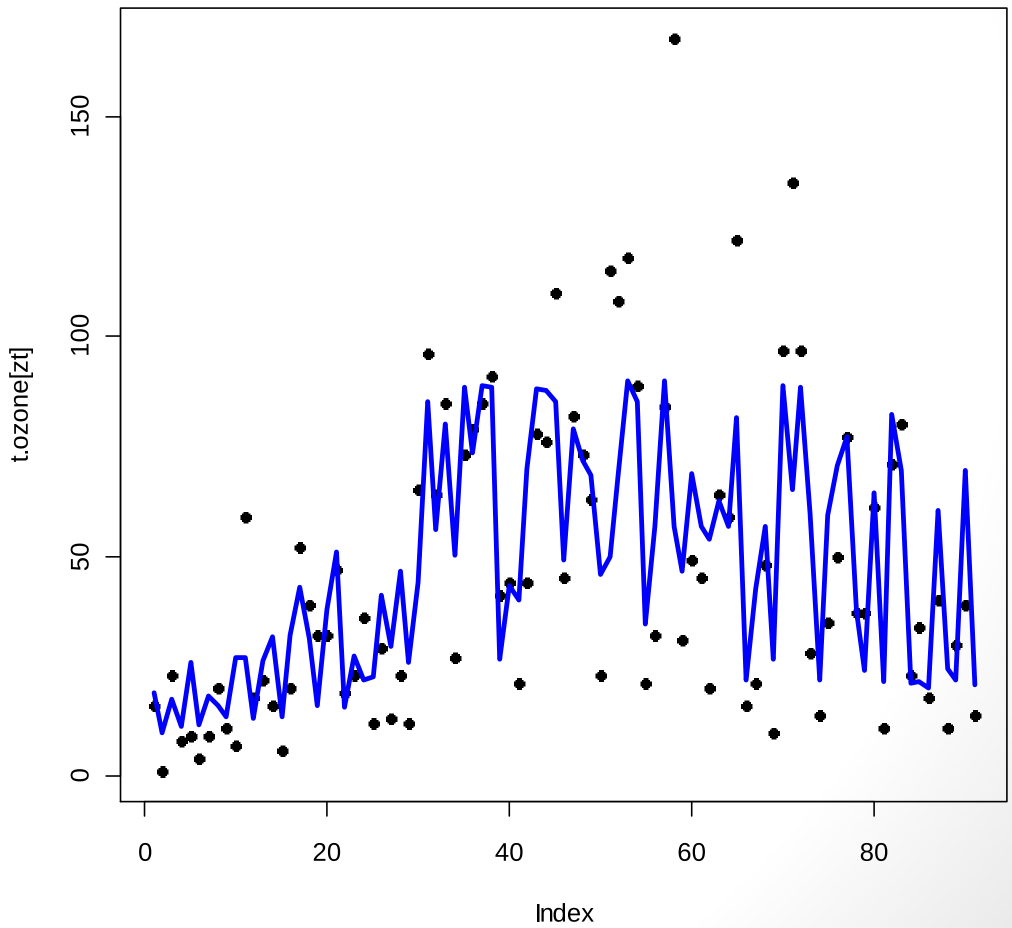
```
NW2 <- function(x,x.dat,y.dat,h) {
  N <- length(y.dat); M <- nrow(x); dim <- ncol(x)
  K1 <- K2 <- numeric(M)
  for (j in 1:M) {
    for (i in 1:N) {
      K1[j] <- K1[j] +
        kern2((x[j,]-x.dat[i,])/h,dim)*y.dat[i]
      K2[j] <- K2[j] + kern2((x[j,]-x.dat[i,])/h,dim)
    }
  }
  K1 / K2
}

kern2 <- function(x,dim=2) exp(-sum(x^2)/2)*(2*pi)^(-dim/2)
```

Множественная регрессия в R

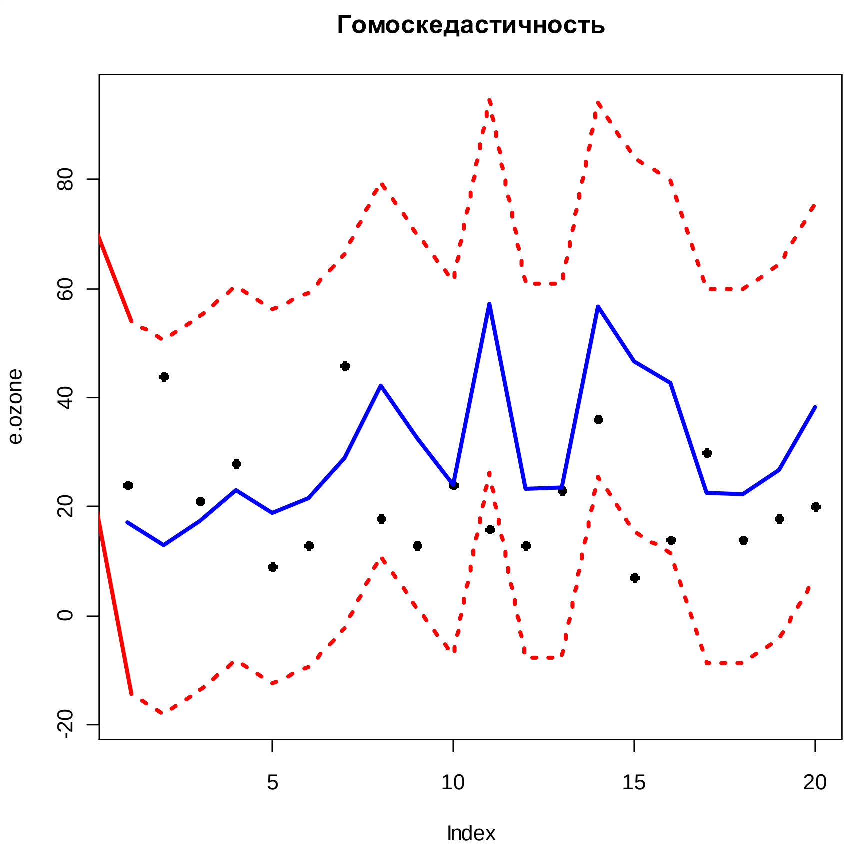
```
# добавочная объясняющая переменная
temp <- airquality$Temp
temp <- temp[!rem]
t.temp <- temp[train.obs]; e.temp <- temp[eval.obs]

# регрессионная модель
bw2 <- npregbw(ozone ~ rad + temp,
  ckertype="gaussian", bwtype="fixed",
  data=data.frame(ozone=t.ozone,rad=t.rad,temp=t.temp))
rad.temp <- cbind(t.rad,t.temp)
ozone.hat <- NW2(rad.temp,rad.temp,t.ozone,bw2$bw,2)

# альтернативный вариант
ozone.reg <- npreg(bw2)
ozone.hat <- predict(ozone.reg)
```

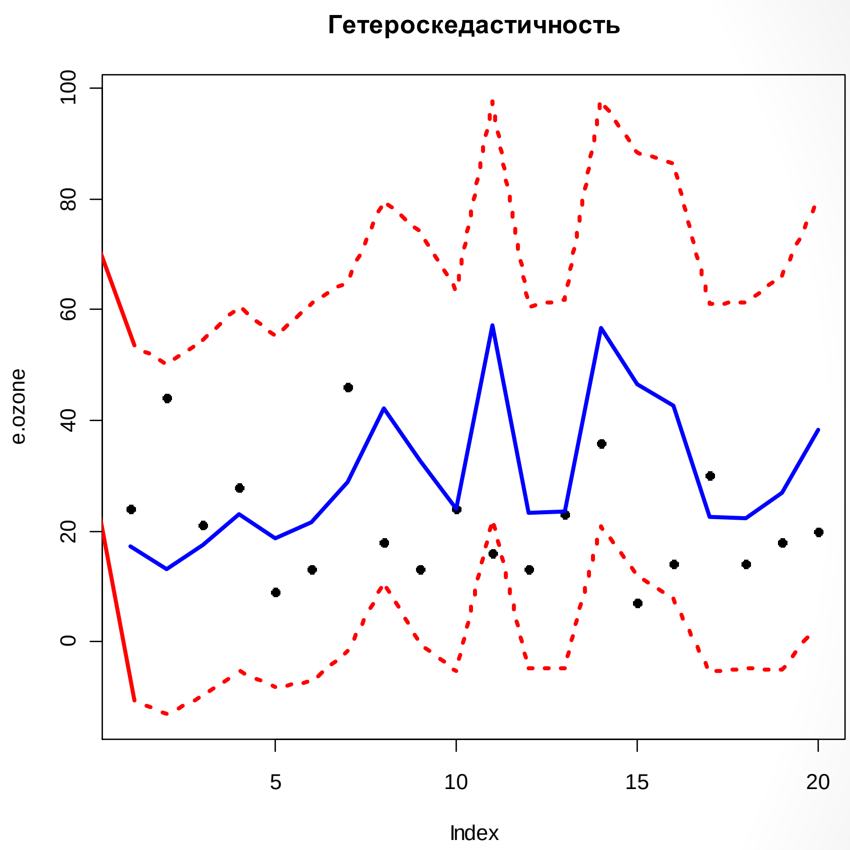
Множественная регрессия в R

```
zt <- order(t.rad)
plot(t.ozone[zt],pch=16)
lines(ozone.hat[zt],col="blue",lwd=3)
```



Построение прогноза





Домашнее задание

- рассмотреть данные о величине спроса на деньги в пакете lmtest: moneydemand
- разделить выборку на обучающую и экзаменующую части
- на пространстве обучающей выборки построить параметрическую и непараметрическую регрессионные модели с эндогенной переменной logM
- проверить качество параметрической модели с помощью тестов на нормальность, гетероскедастичность и автокорреляцию
- построить прогноз и доверительные интервалы для эндогенной переменной на экзаменующей выборке
- написать комментарии