Непараметрическое моделирование Часть 2: Адаптивные методы и многомерный случай

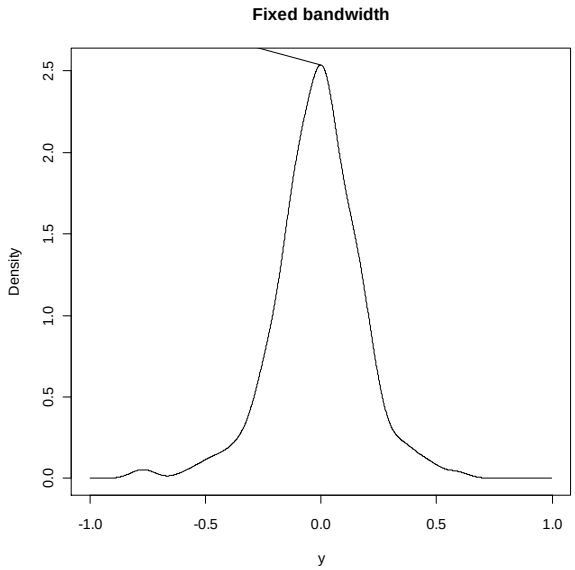
Финансовая эконометрика

Адаптивные методы

Адаптивные методы

Распределение данных может иметь различную концентрацию в центре и на хвостах, поэтому логично использовать широкий интервал *h* там, где они расположены редко (на хвостах), и меньший — в зонах высоких концентраций (в центре)

Ядерные оценки с постоянной шириной интервала в случае гетерогенной концентрации данных пересглаживают распределение в центре и недосглаживают на хвостах:



Метод ближайших соседей

Идея состоит в таком определении интервала, чтобы в него всегда попадало фиксированное количество наблюдений *k*

Рассмотрим расстояние от *i*-го наблюдения в выборке до некоторой точки *y*:

$$d_i(y) = |y_i - y| \quad (20)$$

Отсортируем эти расстояния по возрастанию так, что

$$d_1(y) \leq d_2(y) \leq \cdots \leq d_n(y) \quad (21)$$

k ближайших к точке *y* наблюдений находятся на расстоянии, не превышающем $d_k(y)$

Иными словами, отрезок $[y - d_k(y); y + d_k(y)]$ содержит *k* наблюдений из выборки

Оценка плотности

Положив $h = 2d_k(y)$, мы можем подставить эту величину в простую оценку плотности (3), которая примет вид

$$\hat{f}(y) = \frac{k}{2nd_k(y)} \quad (22)$$

В каждой отдельной точке *y* для любого значения *k* найдётся такое значение *h*, что оценки (3) и (22) дадут один и тот же результат, однако, рассматриваемая в целом, оценка по методу ближайших соседей отличается от простой оценки

Так же, как и в случае с простой оценкой, мы можем прибегнуть к помощи ядерных функций и получить обобщённую оценку по методу ближайших соседей:

$$\hat{f}(y) = \frac{1}{2nd_k} \sum_{i=1}^{n} K\left(\frac{y - y_i}{2nd_k(y)}\right) \quad (23)$$

Задача выбора оптимального значения *k* решается численно, сравнением критериев *ISE* (18) для различных значений *k*

Достоинства и недостатки метода

Достоинства:

- решается проблема недосглаженности и тонкости хвостов

Недостатки:

- оценка $\hat{f}(y)$ не дифференцируема там, где не дифференцируема функция $d_k(y)$;
- хвосты распределения могут казаться тяжелее, чем на самом деле, потому что $d_k(y)$ растёт очень медленно (как функция первой степени);
- в общем случае оценка $\hat{f}(y)$ не является функцией плотности

Адаптивный метод ближайших соседей

Оценка строится в следующем виде:

$$\hat{f}(y) = \frac{1}{n}\sum_{i=1}^{n} \frac{K\left(\frac{y-y_i}{hd_k(y_i)}\right)}{hd_k(y_i)} \quad (24)$$

Сглаживающий параметр разделяется на две части:

1. глобальная (*h*);
2. локальная концентрация наблюдений ($d_k(y_i)$)

Эта оценка лишена недостатков обобщённого метода ближайших соседей

Величину *h* определяют путём построения пилотной оценки плотности $\tilde{f}(y)$ с фиксированной шириной интервала

Часто вместо $d_k(y_i)$ используют показатель

$$\lambda_i = \left(\frac{g}{\tilde{f}(y_i)}\right)^{\alpha}, \; g = \left(\prod_{i=1}^{n} \tilde{f}(y_i)\right)^{\frac{1}{n}}, \; \alpha \in [0;1] \quad (25)$$

Пример 1. Острова

Ядерные оценки

```
library(np)
f.fix <- npudens(tdat=y,edat=x,
  ckertype="gaussian",bwtype="fixed")
```

- ***tdat*** — обучающая выборка
- ***edat*** — точки, в которых рассчитывается оценка
- ***ckertype*** — вид ядерной функции
  "gaussian", "epanechnikov", "uniform"
- ***bwtype*** определяет метод расчёта интервала *h*
  "fixed", "generalized_nn", "adaptive_nn"
- ***f$dens*** — искомые значения оценок

Пусть ***f.fix***, ***f.gen*** и ***f.ada*** — оценки плотности с фиксированным интервалом, по обобщённому методу ближайших соседей и по адаптивному методу ближайших соседей

Пример 1. Острова

Адаптивный метод с λ_i

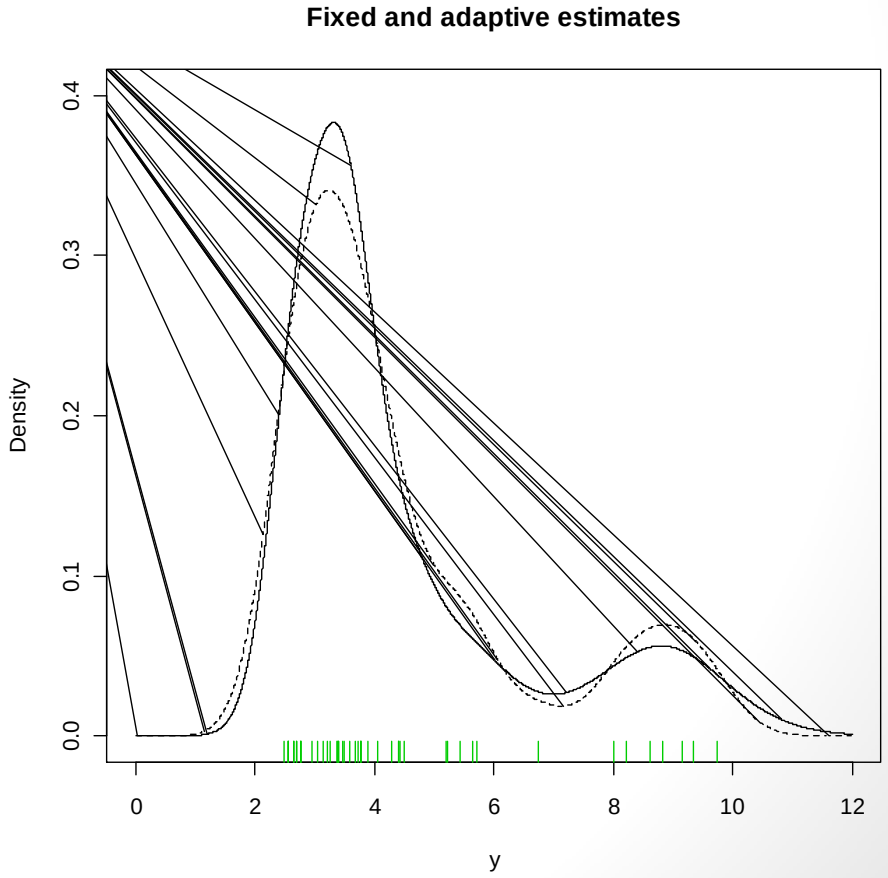
```
pilot <- npudens(tdat=y,ckertype="gaussian",bwtype="fixed")
h <- pilot$bws$bw   # оценка глобальной составляющей интервала

# среднегеометрическое пилотных оценок
g <- 1
for (i in 1:N) g <- g*pilot$dens[i]^(1/N)

# расчёт локальной концентрации наблюдений
alpha <- 0.5
lambda <- (g/pilot$dens)^alpha

kern <- function(u)  exp(-u^2/2)/sqrt(2*pi)   # ядро Гаусса

# расчёт оценок плотности
f <- numeric(L)
for (i in 1:L)  {
  f[i] <- sum(kern((x[i]-y)/(h*lambda))/(h*lambda)
}
f <- f / N
```

Пример 1. Острова

Сравнение адаптивной и фиксированной оценок

```
plot(x,f.fix$dens,type="l",lty="dashed",ylim=c(0,0.4),
main="Fixed and adaptive estimates",
xlab="y",ylab="Density")

lines(x,f)
```

- ***lty*** — тип линии
  "solid", "dashed", "dotted", "dotdash", "longdash", …
- ***ylim*** — границы по оси ординат
- ***lines*** — добавление кривых на существующий график



Пример 1. Острова

Нахождение квантилей оценки распределения, $\hat{F}^{-1}(\alpha)$

```
# оценка функции распределения
F.fix <- npudist(tdat=y,edat=x,ckertype="gaussian",bwtype="fixed")

# для адаптивного варианта
F <- rep(0,times=L)
for (i in 1:L)  F[i] <- sum(f[1:i])*dx

# поиск квантиля методом деления пополам
alpha <- 0.99
a <- 1; b <- L; ab <- trunc((a+b)/2)
while ((b-a)>2) {
  if  (F.fix$dist[ab]<=alpha)  a <- ab
  if  (F.fix$dist[ab]>=alpha)  b <- ab
  ab <- trunc((a+b)/2)
}
q.fix <- x[ab]
```

q.fix	10.00
q.ada	10.49

Пример 1. Острова

Генератор случайных чисел

фиксированный интервал

```
M <- 10^6
y.fix.sim <- sample(x,prob=f.fix$dens,size=M,replace=TRUE)
q.fix <- sort(y.fix.sim)[alpha*M]
```

для адаптивного варианта

```
y.ada.sim <- sample(x,prob=f,size=M,replace=TRUE)
q.ada <- sort(y.ada.sim)[alpha*M]
```

q.fix	10.01
q.ada	10.46

Многомерный случай

Оценки плотности

Простая оценка плотности в двумерной точке (y_1, y_2):

$$\hat{f}(y_1, y_2) = \frac{1}{nh^2}\sum_{i=1}^{n}\left(I\left(y_1 - \frac{h}{2} < y_{i,1} < y_1 + \frac{h}{2}\right)\cdot I\left(y_2 - \frac{h}{2} < y_{i,2} < y_2 + \frac{h}{2}\right)\right) \quad (26)$$

Заменим индикаторы на ядерные функции:

$$\hat{f}(y_1, y_2) = \frac{1}{nh^2}\sum_{i=1}^{n}\left(K\left(\frac{y_{1,i}-y_1}{h}\right)K\left(\frac{y_{2,i}-y_2}{h}\right)\right) \quad (27)$$

Оценка (27) не обязательно даёт одинаковый результат для всех точек, равноудалённых от пары (y_1, y_2)

Проблема решается с помощью многомерного ядра:

$$\hat{f}(y_1, y_2) = \frac{1}{nh^2}\sum_{i=1}^{n}K\left(\frac{y_{1,i}-y_1}{h}, \frac{y_{2,i}-y_2}{h}\right) \quad (28)$$

В качестве $K(x_1, x_2)$ обычно берутся одномодальные симметричные многомерные функции плотности

Двумерные ядерные функции

Ядро Гаусса:

$$K_G(x_1, x_2) = \frac{1}{2\pi} \exp\left(-\frac{x_1^2 + x_2^2}{2}\right) \quad (29)$$

Двумерное ядро Гаусса в точности равно произведению двух одномерных:

$$K_G(x_1, x_2) \equiv K_G(x_1) \cdot K_G(x_2)$$

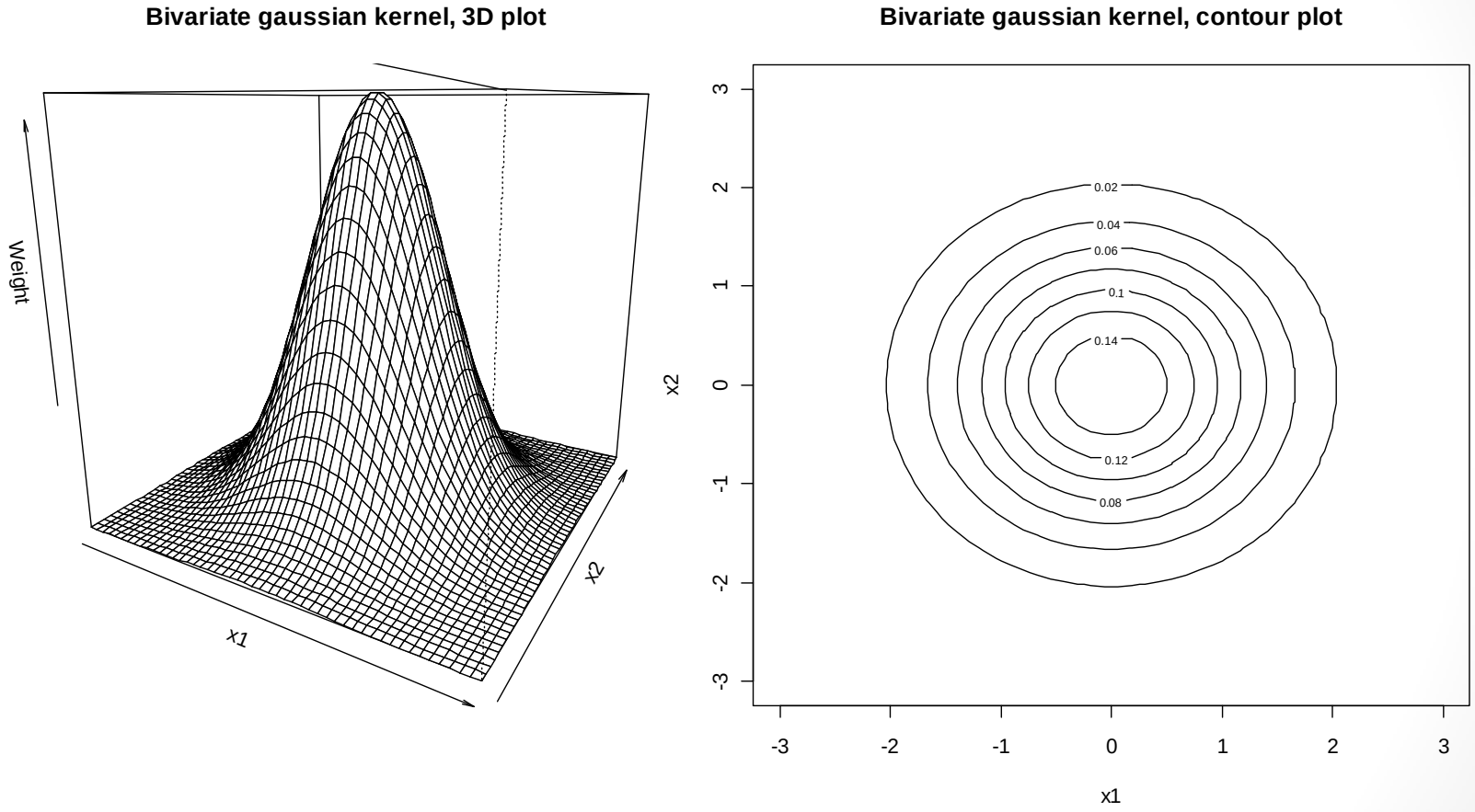
Ядро Епанечникова:

$$K_E(x_1, x_2) = \frac{2}{\pi}(1 - x_1^2 - x_2^2) \cdot I(x_1^2 + x_2^2 < 1) \quad (30)$$

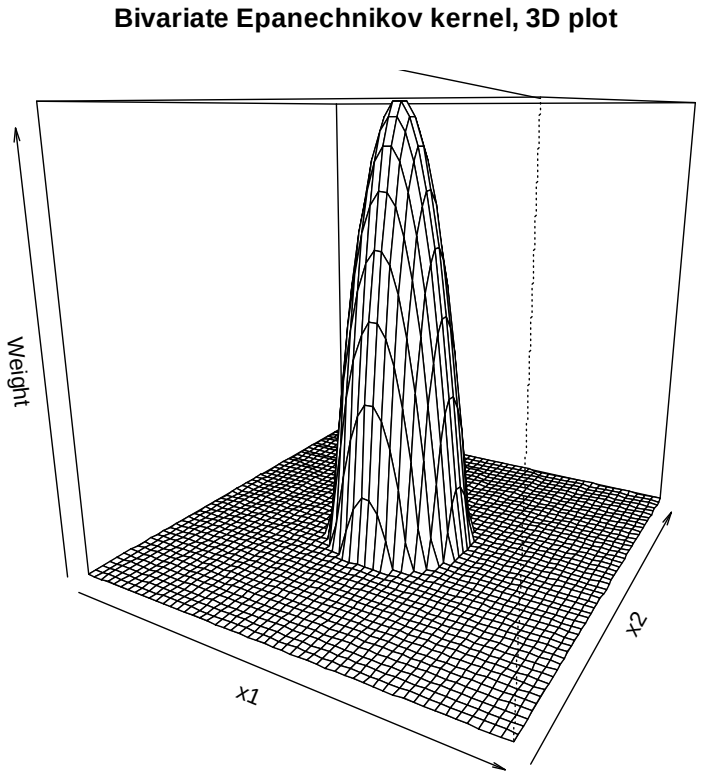
Двумерное ядро не равно произведению двух одномерных:

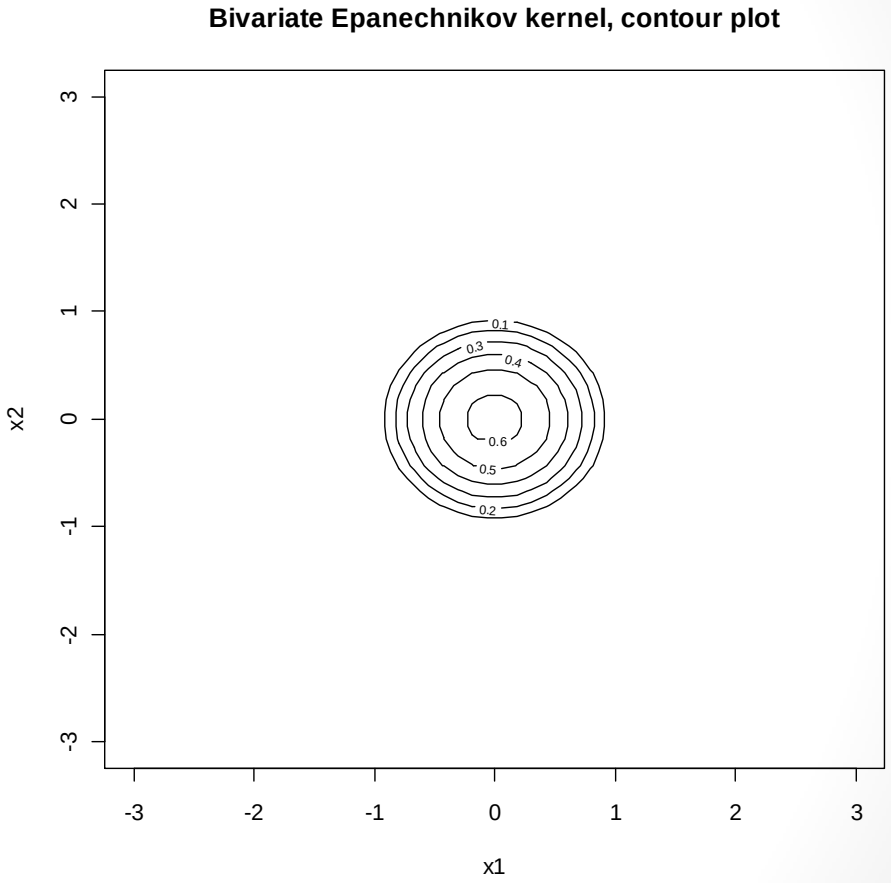
$$K_E(x_1, x_2) \neq K_E(x_1) \cdot K_E(x_2)$$

Двумерное гауссовское ядро

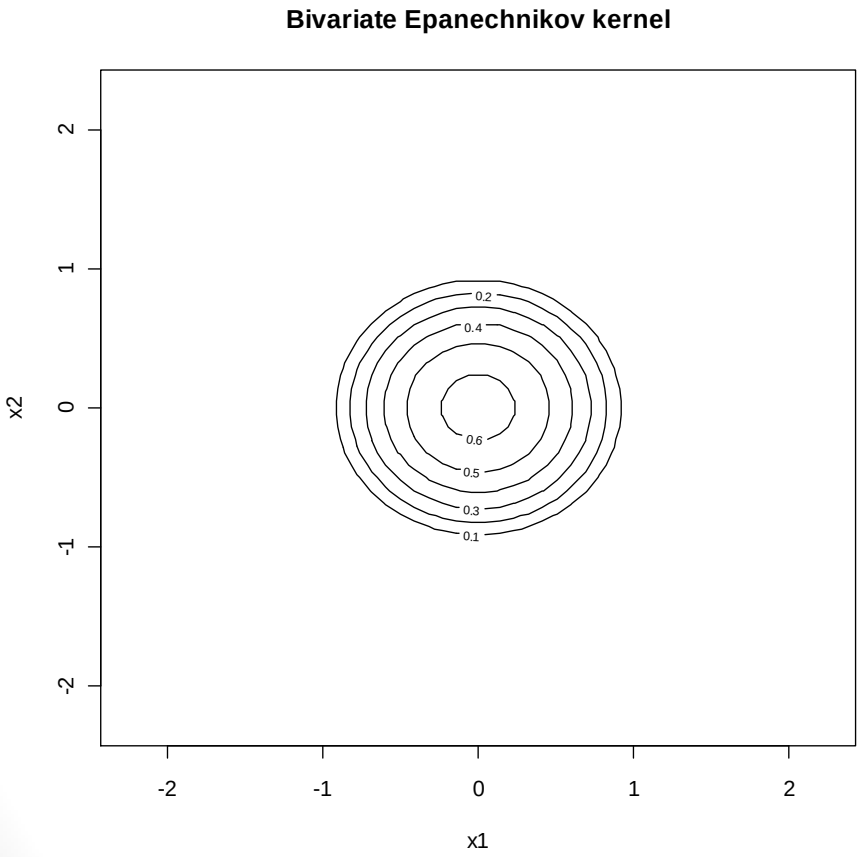


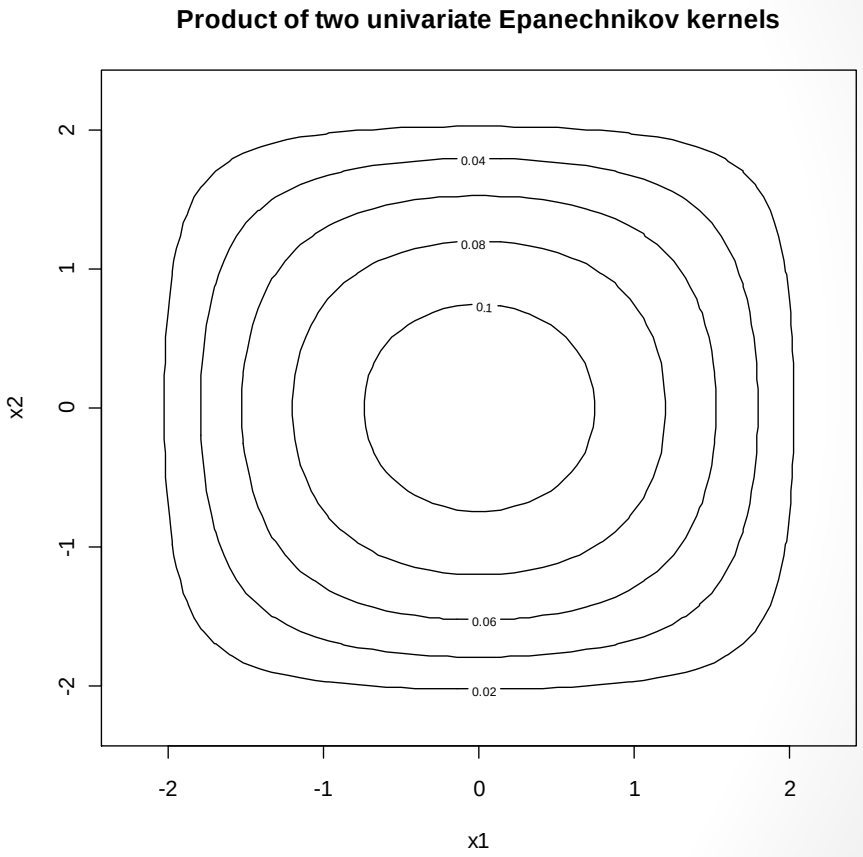
Двумерное ядро Епанечникова





Двумерные ядерные функции

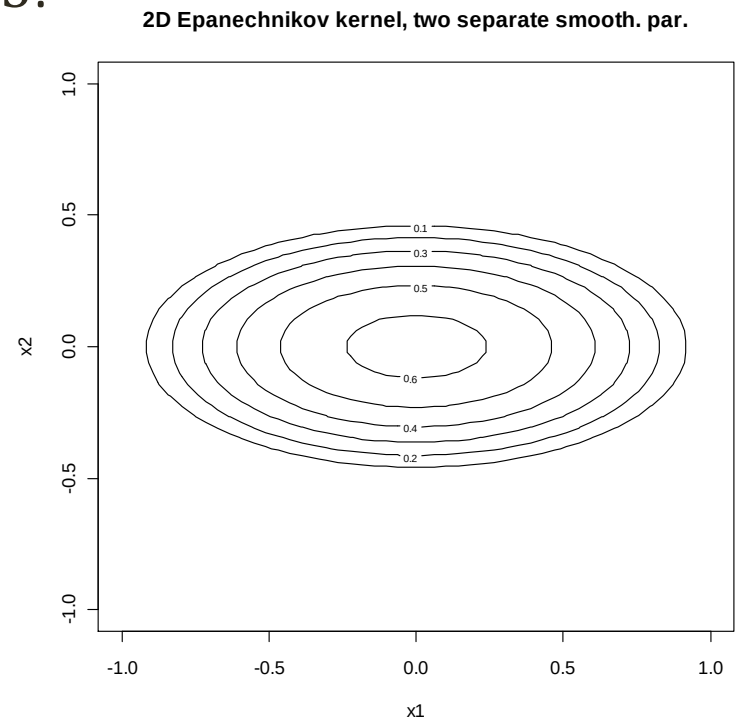




Различные сглаживающие параметры

Если разброс данных в первой и во второй выборке сильно отличаются, то можно использовать для этих выборок разные сглаживающие параметры $h_1 \neq h_2$

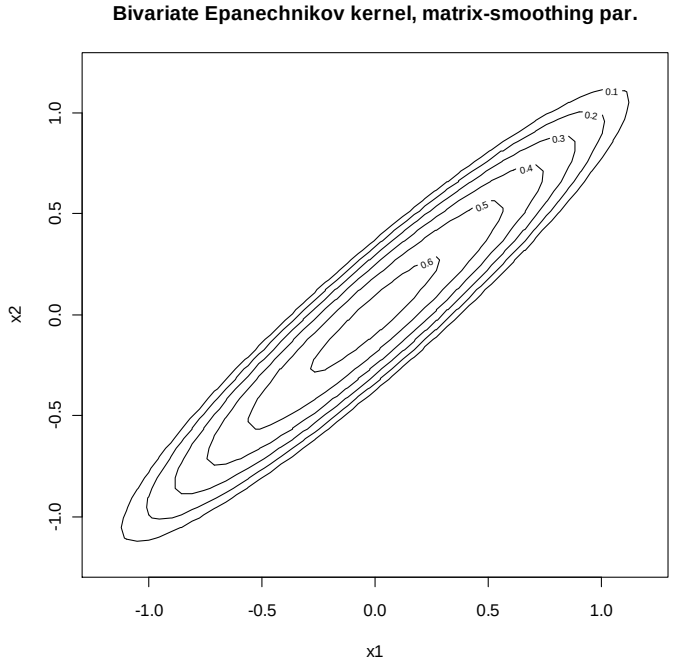
На рисунке ниже представлено двумерное ядро Епанечникова $K\left(\frac{x_1}{h_1}, \frac{x_2}{h_2}\right)$ со сглаживающими параметрами $h_1 = 1,\ h_2 = 0.5$:



Сглаживающая матрица

Если рассматриваемые величины коррелируют, это учитывается с помощью симметричной положительно определённой сглаживающей матрицы (matrix-smoothing parameter) *H*, которая в двумерном случае состоит их 4-х элементов: $H = \begin{pmatrix} h_1 & h_{12} \\ h_{21} & h_2 \end{pmatrix},\ h_{12} = h_{21}$

Для $h_1 = h_2 = 1,\ h_{12} = h_{21} = \sqrt{0.5}$, получим:



Общий случай

Ядерная оценка *d*-мерной плотности:

$$\hat{f}(\vec{y}) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{|H|} K(H^{-1}(\vec{y} - \vec{y}_i)), \quad (31)$$

$\vec{y} = (y_1, \dots, y_d)$, $|H|$ — определитель матрицы H

Пусть $\vec{x} = (x_1, \dots, x_d)$, тогда *d*-мерные ядра Гаусса и Епанечникова запишутся как

$$K_G(\vec{x}) = (2\pi)^{-\frac{d}{2}} \exp\left(-\frac{\vec{x}\vec{x}'}{2}\right), \quad (32)$$

$$K_E(\vec{x}) = (2c_d)^{-1}(d+2)(1 - \vec{x}\vec{x}') \cdot I(\vec{x}\vec{x}' < 1), \quad (33)$$

c_d — объём единичного *d*-мерного шара

Правило подстановки

Если в качестве подставляемого распределения использовать нормальное $N(\mu, \Sigma)$, $\Sigma = \left(\sigma_1^2, \dots, \sigma_d^2\right) \cdot Ъ$, $Ъ_{[d \times d]}$ — единичная матрица, то по критерию *MISE*, оптимальная диагональная матрица *H* состоит из элементов

$$h_j = \left(\frac{4}{d+2}\right)^{\frac{1}{d+4}} n^{-\frac{1}{d+4}} \sigma_j \quad (34)$$

Так как первый множитель при любых *d* приблизительно равен единице на практике используют правило

$$\hat{h} = n^{-\frac{1}{d+4}} \hat{\sigma}_j \quad (35)$$

Обобщённое правило подстановки:

$$\hat{H} = n^{-\frac{1}{d+4}} \hat{\Sigma}^{\frac{1}{2}} \quad (36)$$

Метод перекрёстной проверки

Метод перекрёстной проверки также может быть обобщён на многомерный случай, однако при этом он становится достаточно сложным, требующим ресурсоёмких вычислений

Алгоритм практического применения метода аналогичен правилу подстановки, но в этом случае, при предположении, что $K(\vec{x})$ — симметричная функция, оценка $\hat{h}$ находится путём минимизации выражения

$$CV(h) = \frac{1}{n^2 h^d} \sum_{i=1}^{n} \sum_{j=1}^{n} K^* \left(\frac{\vec{y}_i - \vec{y}_j}{h} \right) + \frac{2}{nh^d} K\left(0_{[1\times d]}\right), \quad (37)$$

$$K^*(\vec{x}) = \int_{R^d} K\left(\vec{t}\right) K\left(\vec{x} - \vec{t}\right) d\vec{t} - 2K(\vec{x}),$$

$$\vec{t} = (t_1, \dots, t_d)$$

Обобщённый метод ближайших соседей

Пусть $d_k(\vec{y})$ — евклидово расстояние от точки $\vec{y}$ до *k*-го ближайшего наблюдения в выборке, $V_k(\vec{y})$ — объём *d*-мерного шара радиусом $d_k(\vec{y})$, $V_k(\vec{y}) = c_d d_k^d(\vec{y})$

В этом случае простая оценка равна

$$\hat{f}(\vec{y}) = \frac{k}{nV_k(\vec{y})} = \frac{k}{nc_d d_k^d(\vec{y})}, \quad (38)$$

что аналогично одномерной оценке (22), так как $c_1 = 2$

Оценка (38) может быть обобщена с помощью ядер:

$$\hat{f}(\vec{y}) = \frac{1}{c_d n d_k^d(\vec{y})} \sum_{i=1}^{n} K\left(\frac{\vec{y}-\vec{y}_i}{c_d d_k(\vec{y})}\right), \quad (39)$$

что аналогично одномерной оценке (23)

Адаптивный метод ближайших соседей

Рассмотрим $d_k(\vec{y}_i)$ — расстояние от элемента $\vec{y}_i$ до *k*-го ближайшего элемента выборки

Оценка плотности по адаптивному методу равна

$$\hat{f}(\vec{y}) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h^d d_k^d(\vec{y}_i)} K\left(\frac{\vec{y}-\vec{y}_i}{h d_k(\vec{y}_i)}\right), \quad (40)$$

что аналогично одномерной оценке (24)

Как и в одномерном случае показатель локальной концентрации наблюдений $d_k(\vec{y}_i)$ часто заменяется на величину

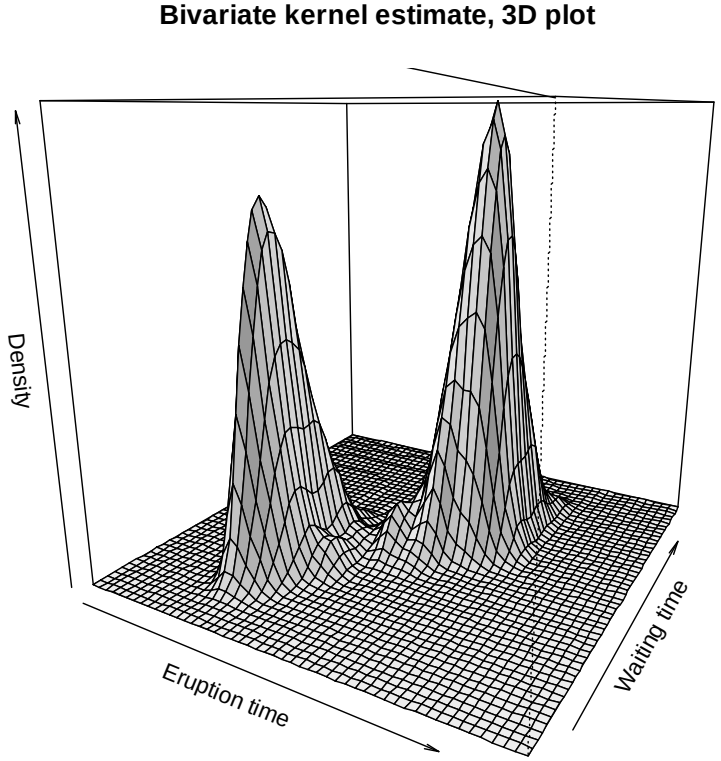
$$\lambda_i = \left(\frac{g}{\tilde{f}(\vec{y}_i)}\right)^{\alpha}, \ \alpha \in [0;1], \quad (41)$$

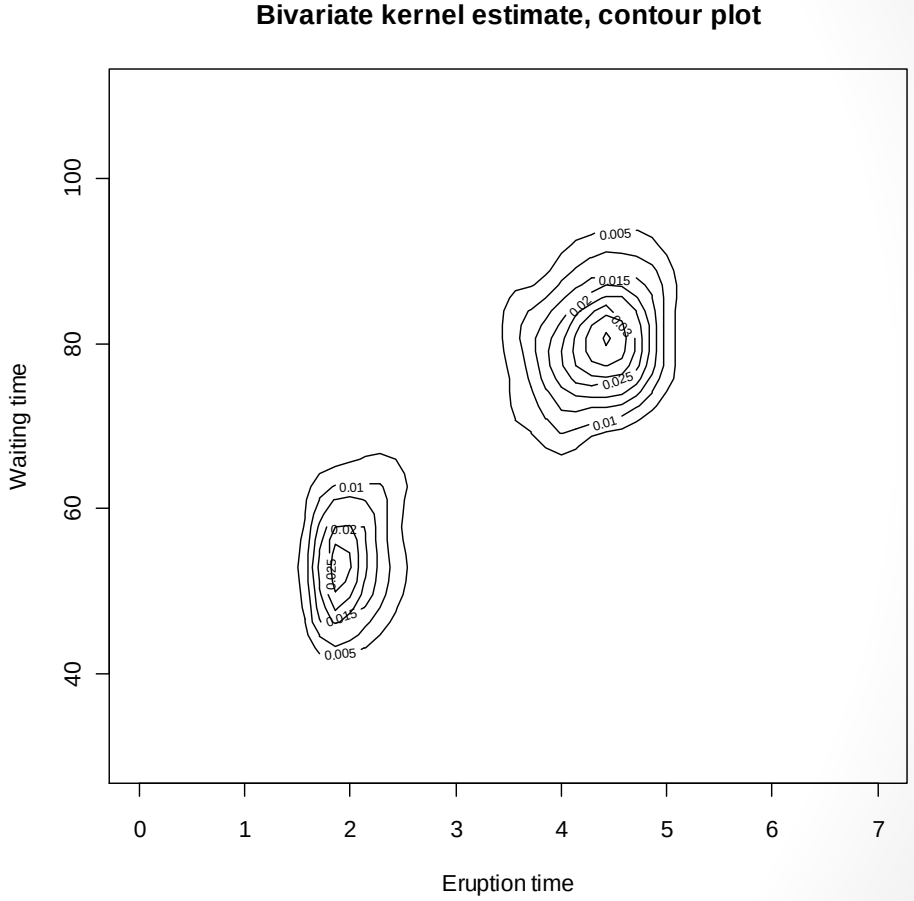
$g = \left(\prod_{i=1}^{n} \tilde{f}(y_i)\right)^{\frac{1}{n}}$ — геометрическое среднее пилотных оценок плотности

Пример 2. Старый служака

```
y <- faithful; N <- nrow(y)

# сетка для расчёта оценок плотности
L <- 50; u <- seq(0,7,length=L); v <- seq(30,110,length=L)
uv <- expand.grid(u,v)

# оценка плотности
f.fix <- npudens(tdat=y,edat=uv,ckertype="gaussian",bwtype="fixed")

# графики оценки
w <- f.fix$dens; dim(w) <- c(L,L)

persp(u,v,w,theta=30,main="Bivariate kernel estimate, 3D plot",
xlab="Eruption time",ylab="Waiting time",zlab="Density")

contour(u,v,w,nlevel=7,
main="Bivariate kernel estimate, contour plot",
xlab="Eruption time",ylab="Waiting time")
```

Пример 2. Старый служака



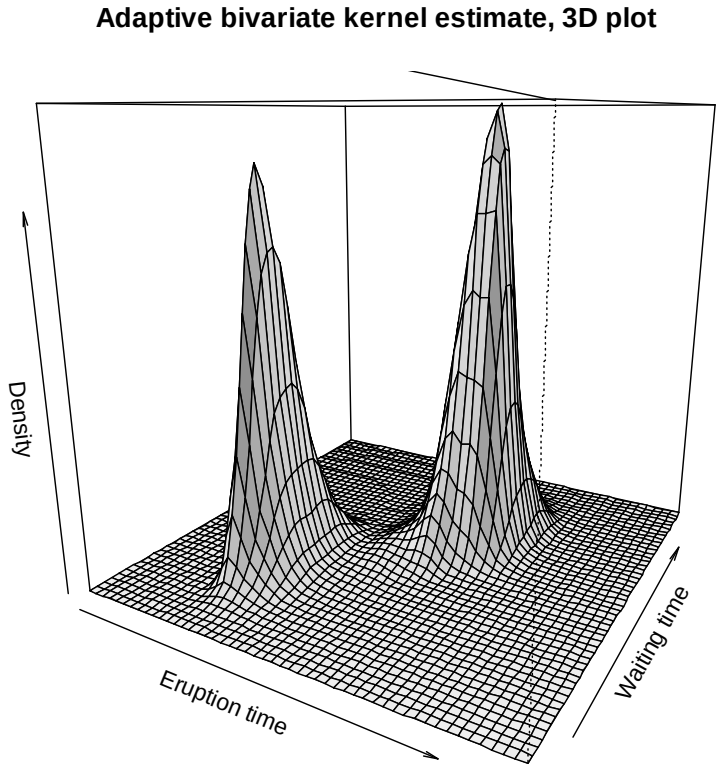


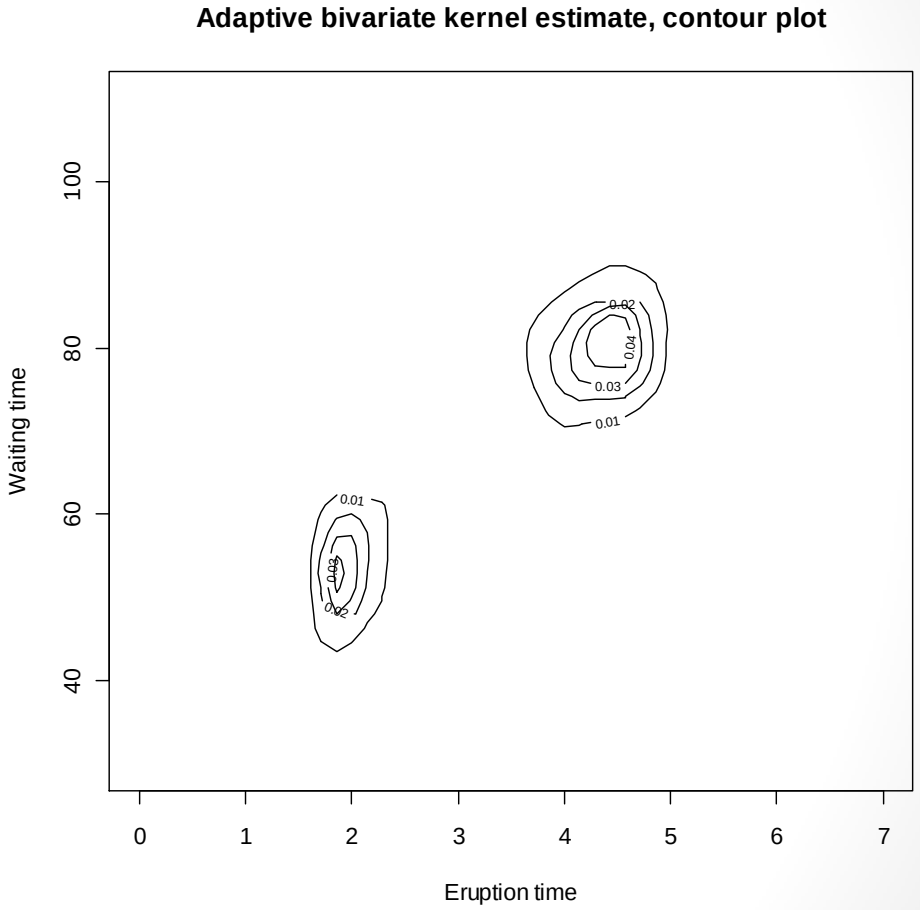
Пример 2. Старый служака

Адаптивный метод с λ_i, аналогично одномерному случаю

```
pilot <- npudens(tdat=y,ckertype="gaussian",bwtype="fixed")
h <- pilot$bws$bw

g <- 1
for (i in 1:N) g <- g*pilot$dens[i]^(1/N)

alpha <- 0.5
lmbd <- (g/pilot$dens)^alpha

kern <- function(x) exp(-(x[1]^2+x[2]^2)/2)/(2*pi)

f <- rep(0,times=L^2)
for (i in 1:(L^2)) {
  for (j in 1:N) f[i] <- f[i]+kern((uv[i,]-
    y[j,])/(h*lmbd[j]))/lmbd[j]^2
  f[i] <- f[i]/(N*h[1]*h[2])
}
```

Пример 2. Старый служака





Пример 2. Старый служака

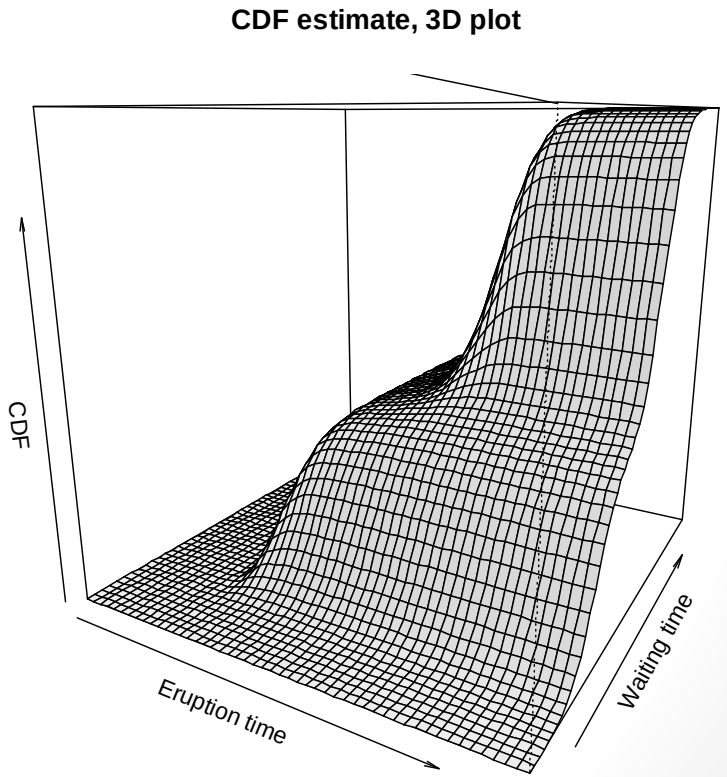
Расчёт функций распределения

фиксированный метод

```
F.fix <- npudist(tdat=y,edat=uv,ckertype="gaussian",bwtype="fixed")
```

адаптивный метод

```
du <- u[2]-u[1]; dv <- v[2]-v[1]
w <- f; dim(w) <- c(L,L)
F <- rep(0,times=L^2)
for (i in 1:L) {
  for (j in 1:L) F[j+(i-1)*L] <-
  sum(w[1:j,1:i])*du*dv
}
```

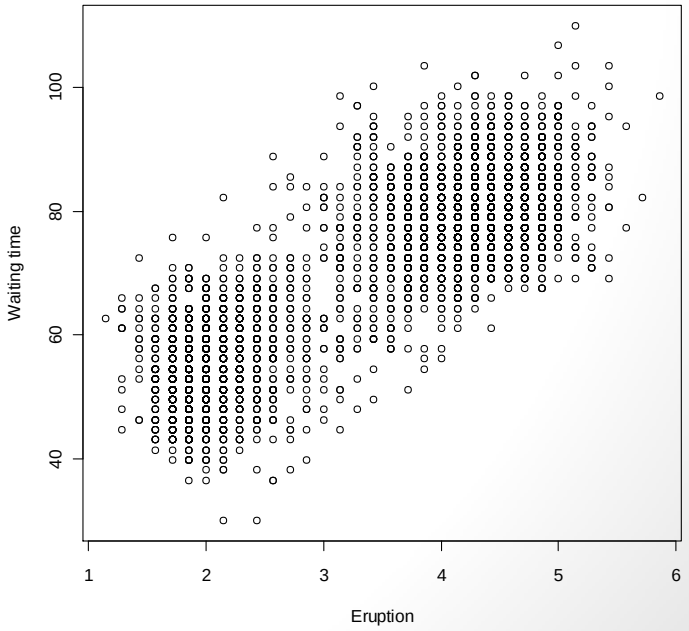


Пример 2. Старый служака

Генератор случайных чисел

для адаптивного метода

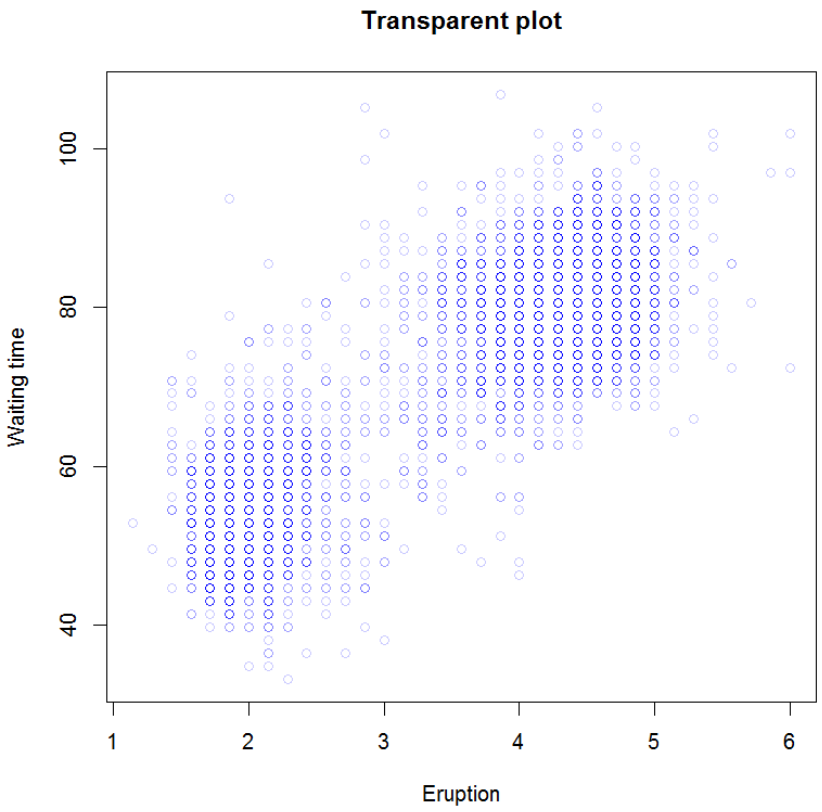
```
alpha <- 0.99
M <- 5000
smpl.ind <- sample(1:(L^2),prob=f,size=M,replace=TRUE)
y.ada.sim <- uv[smpl.ind,]
plot(y.ada.sim,xlab="Eruption",ylab="Waiting time")
```

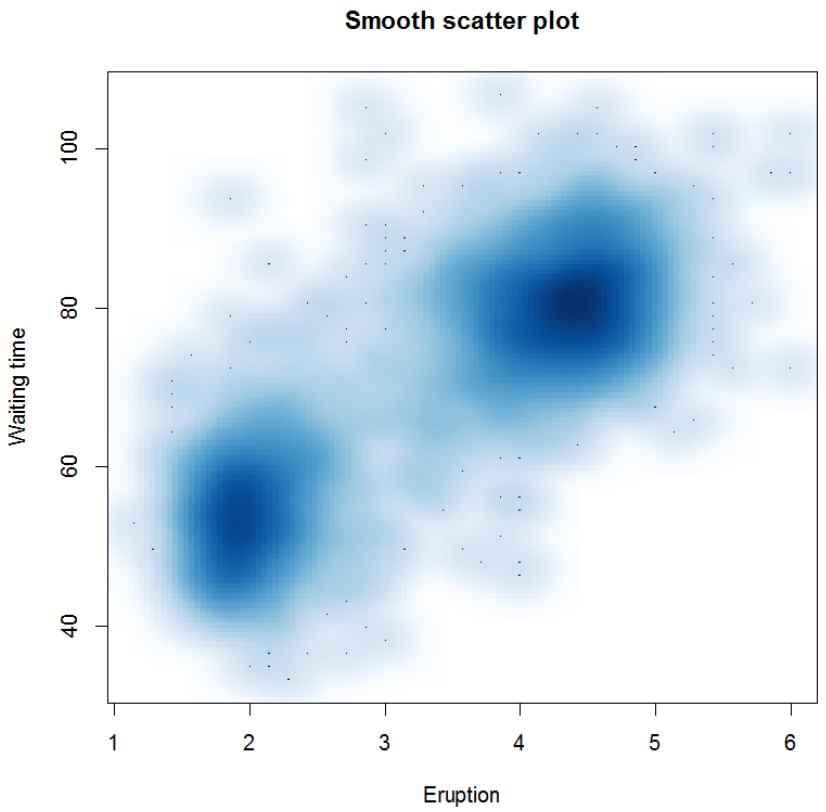


Пример 2. Старый служака

Рисование графиков с перекрывающими друг друга точками

```
plot(y.ada.sim,col=rgb(0,0,1,alpha=0.2))
smoothScatter(y.ada.sim)
```





Домашнее задание

- рассчитать оценки риска для портфеля из двух биржевых индексов с помощью многомерного адаптивного метода (с величинами λ_i)
- построить кривую VaR для портфеля и проверить качество оценок

Исходные данные — ежедневные котировки с сайтов finam.ru, finance.yahoo.com