

Определение аномалий (Anomaly detection)

Методы машинного обучения и численной
оптимизации, часть 5

Центр математических финансов

Основная идея

Если количество наблюдений обучающей выборки, принадлежащих к одному из классов, крайне мало, то алгоритм классификации строится на изучении свойств доминирующего класса

При этом наблюдения из большего класса считаются «нормальными», а наблюдения из меньшего — «аномалиями»

Принцип определения аномалий

Производится проверка, похоже ли наблюдение из экзаменующей выборки $x_{test}^{(i)}$ на нормальные наблюдения из обучающей выборки

Рассчитывается «вероятность нормальности»:

$$p(\vec{x}_{test}^{(i)}) < \varepsilon \Rightarrow \text{аномалия}, \quad p(\vec{x}_{test}^{(i)}) \geq \varepsilon \Rightarrow OK$$

Обычно делается предположение, что $x_{test} \sim N(\mu, \sigma)$, тогда

$$p(x_j; \mu_j, \sigma_j) = \frac{1}{\sigma_j \sqrt{2\pi}} \exp\left(-\frac{(x_j - \mu_j)^2}{2\sigma_j^2}\right)$$

Оценки параметров находятся с помощью ММП:

$$\hat{\mu} = \frac{1}{m} \sum_{i=1}^m \vec{x}_{train}^{(i)}, \quad \hat{\sigma}^2 = \frac{1}{m-1} \sum_{i=1}^m \left(x_{train}^{(i)} - \hat{\mu}\right)^2$$

$$p(\vec{x}) = p(x_1; \mu_1, \sigma_1) \times \cdots \times p(x_n; \mu_n, \sigma_n) = \prod_{j=1}^n p(x_j; \mu_j, \sigma_j)$$

Алгоритм определения аномалий

1. Выбрать показатели x_j , которые могут быть индикаторами аномалии
2. Подобрать параметры $\vec{\mu}$ и $\vec{\sigma}$
3. Рассчитать $p(\vec{x}) = \prod_{j=1}^n \frac{1}{\sigma_j \sqrt{2\pi}} \exp\left(-\frac{(x_j - \mu_j)^2}{2\sigma_j^2}\right)$
4. $p(\vec{x}) < \varepsilon \Rightarrow$ аномалия

Значение ε может быть определено на валидационной выборке

Разделение выборки

На три части

	ОК	Аномалии
Обучающая	60 %	0 %
Валидационная	20 %	50 %
Тестовая	20 %	50 %

На две части

	ОК	Аномалии
Обучающая	80 %	0 %
Валидационная	20 %	100 %

Область применения алгоритма определения аномалий

Условия применения

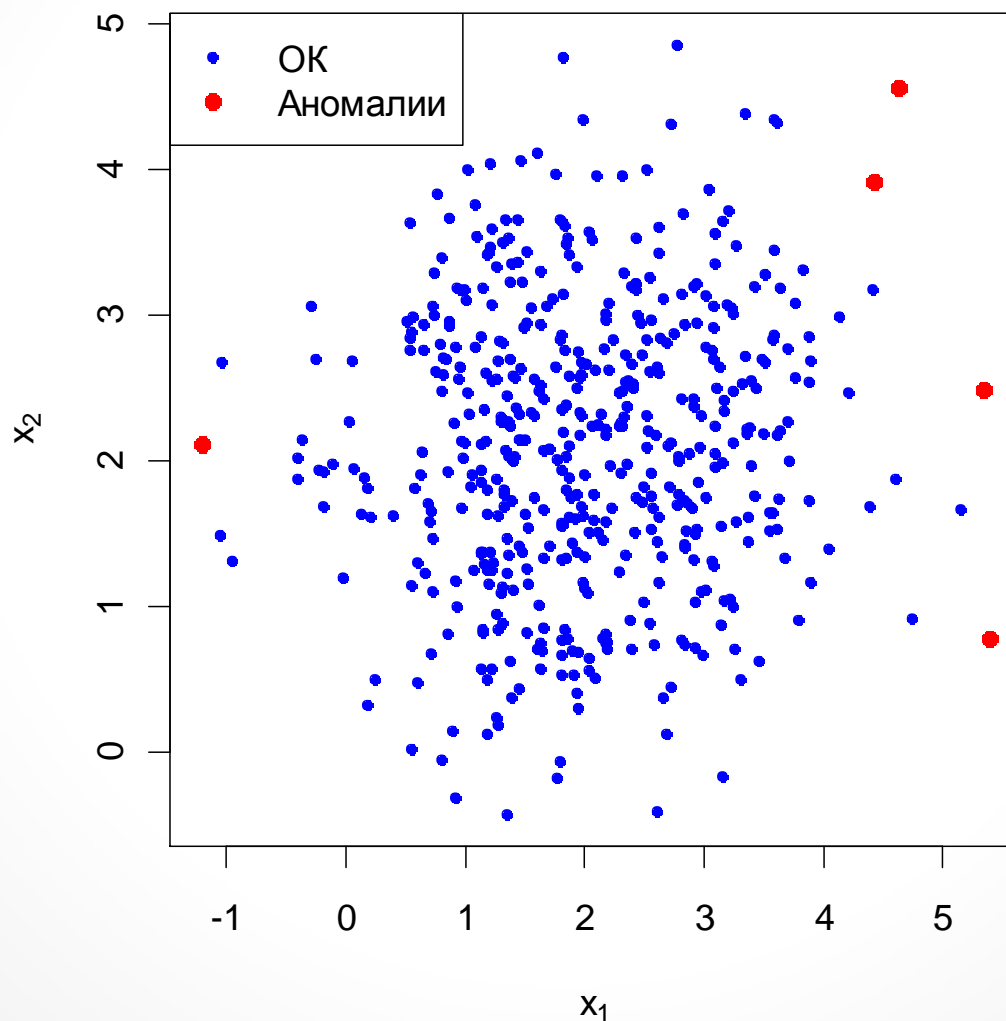
Определение аномалий	LR, NN, SVM
Малое количество (0 – 20) аномальных наблюдений	Большое количество наблюдений каждого класса
Большое количество возможных типов аномалий	Фиксированный тип аномалий

Области применения

Определение аномалий	LR, NN, SVM
Недобросовестное поведение пользователей	Спам в электронных письмах
Брак изделий	Прогноз погоды
Неисправность компьютеров в сети	Постановка диагноза

Определение аномалий в R

Пусть X — матрица наблюдений, y — классификатор



Разделение выборки

```
m <- nrow(X)
```

номера «аномальных» наблюдений

```
anom.obs <- (1:m)[y==1]; la <- length(anom.obs)
```

```
m.cv <- round(0.2*(m-la)) + la; m.train <- m - m.cv
```

номера экзаменующей и обучающей выборок

```
cv.obs <- c( sample((1:m)[-anom.obs],size=m.cv-la,replace=FALSE),  
            anom.obs )
```

```
train.obs <- (1:m)[-cv.obs]
```

разделение выборки

```
X.train <- X[train.obs,]; X.cv <- X[cv.obs,]
```

```
y.train <- y[train.obs]; y.cv <- y[cv.obs]
```


Подбор параметров и расчёт «вероятности нормальности»

оценки параметров

```
mu <- apply(X.train,2,mean)
sigma <- apply(X.train,2,sd)
```

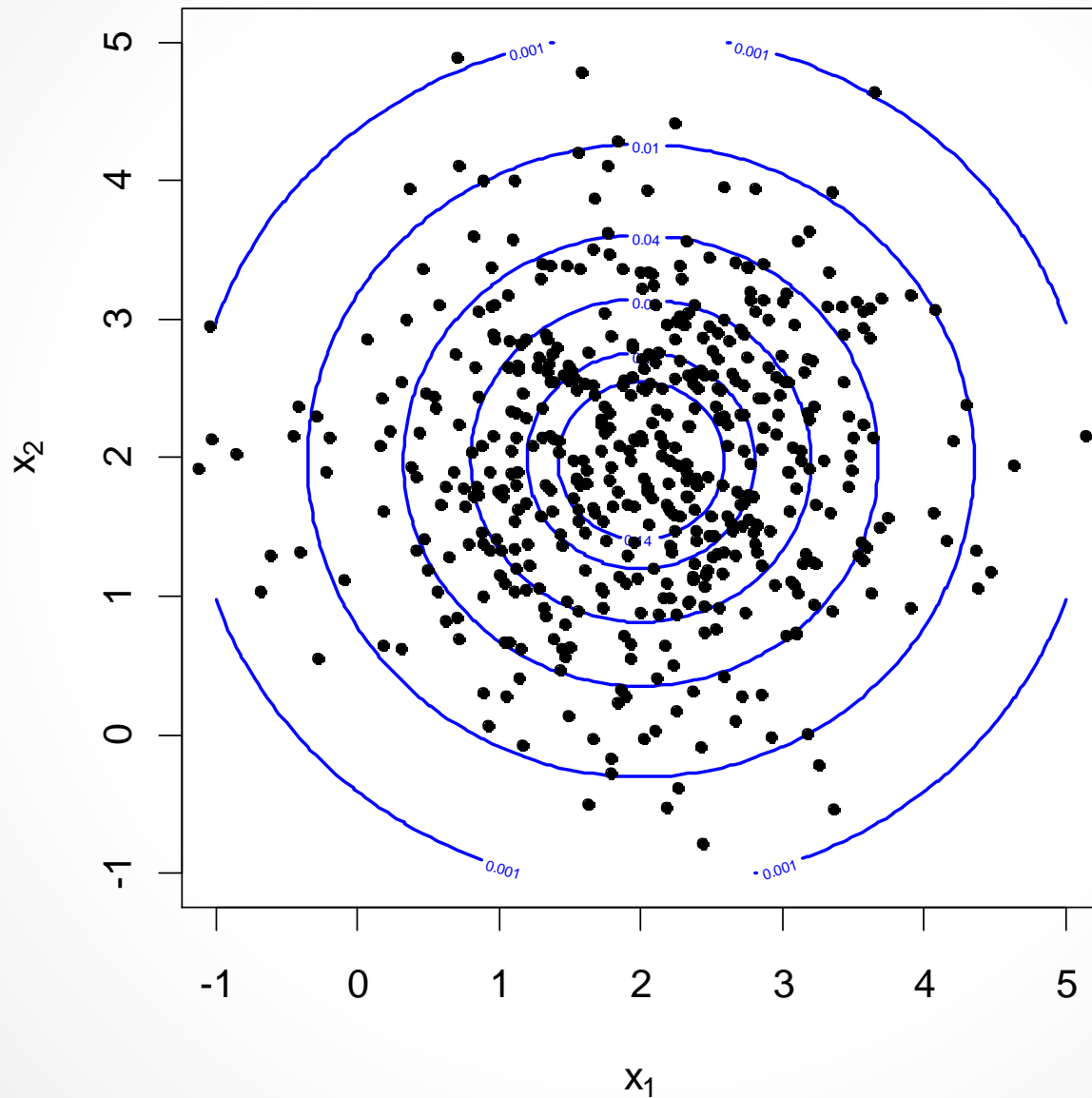
функция «вероятности»

```
p <- function(X,mu,sigma) {
  m <- nrow(X); n <- ncol(X)
  prob <- matrix(nrow=m,ncol=n)
  for (j in 1:n) prob[,j] <- dnorm(X[,j],mu[j],sigma[j])
  apply(prob,1,prod)
}
```

определение «вероятностей»

```
prob.train <- p(X.train,mu,sigma)
prob.cv <- p(X.cv,mu,sigma)
```

Линии уровня функции «вероятности»



Определение параметра ε

```
pr <- range(prob.cv) # границы возможных значений  $\varepsilon$ 
res <- NULL # в неё будут сохраняться результаты моделирования

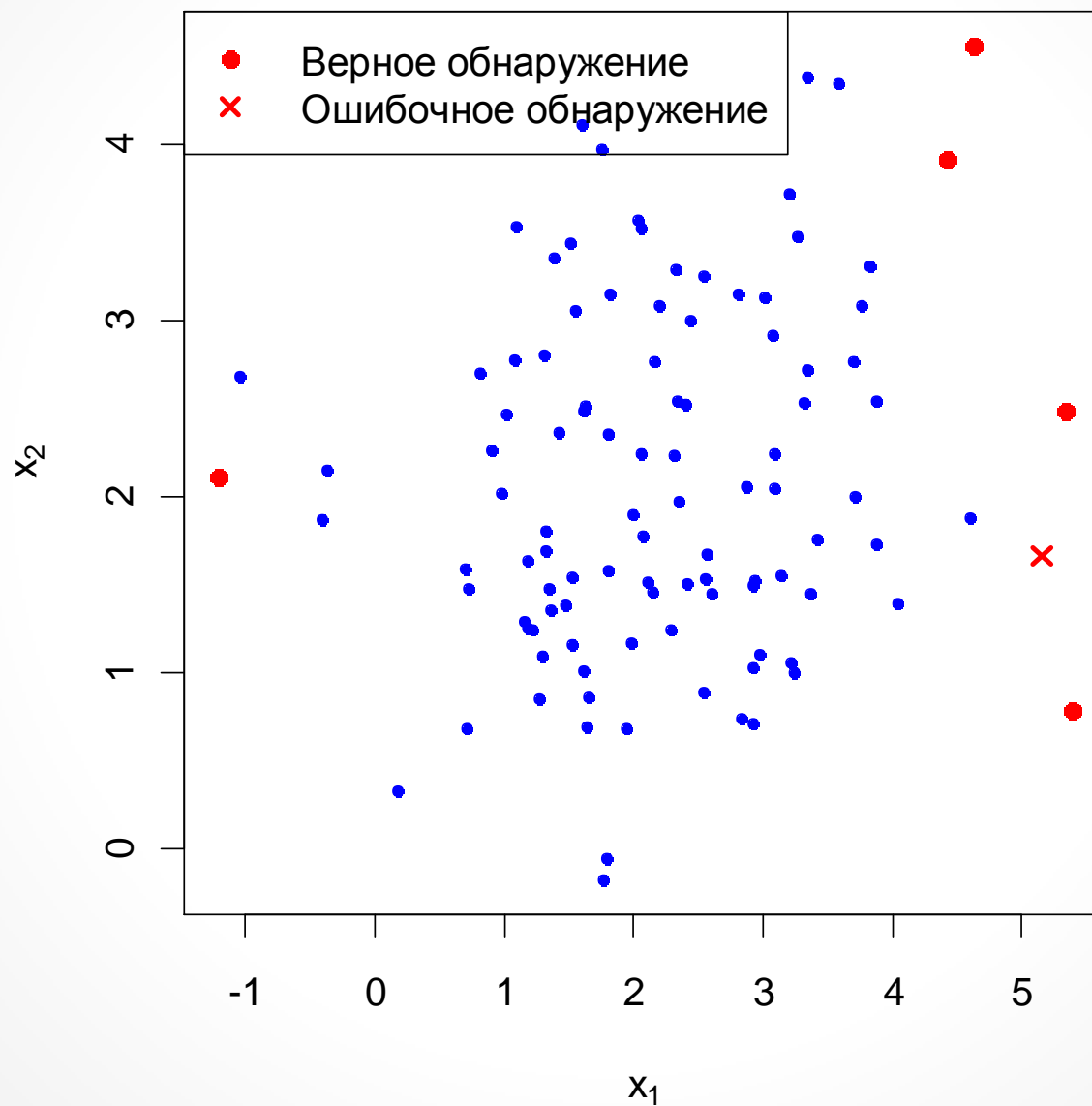
# для каждого наблюдения экзаменующей выборки рассчитываем
# прогноз при определённом значении  $\varepsilon$  и сравниваем его с фактом
for (eps in seq(pr[1], pr[2], length = 1000)) {
  y.pred <- 1*(prob.cv<eps)
  res <- rbind(res, c(eps, fitStats(y.pred, y.cv)) )
}

dimnames(res)[[2]][1] <- "epsilon" # заголовки

# выбор наиболее подходящего  $\varepsilon$ 
j <- which.max(res[, "f1.score"])
eps <- res[j, "epsilon"]

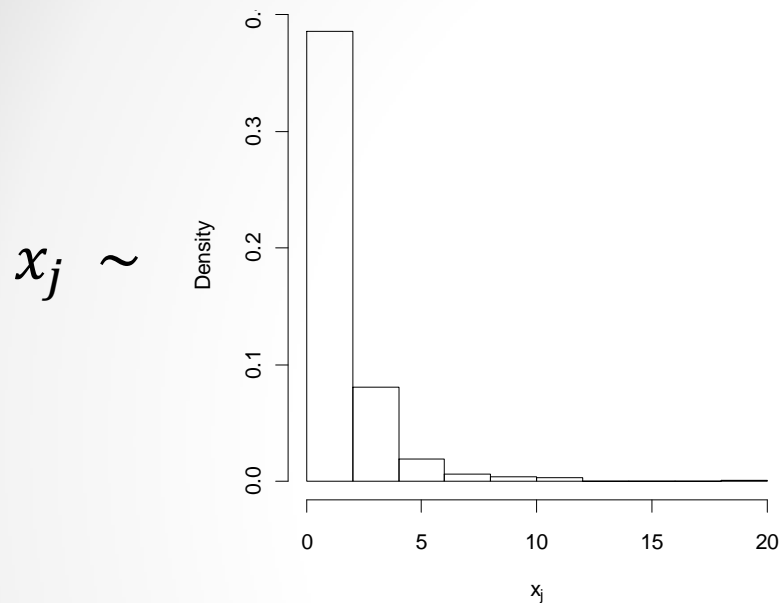
# окончательный прогноз
y.pred <- 1*(prob.cv<eps)
```

Результаты моделирования

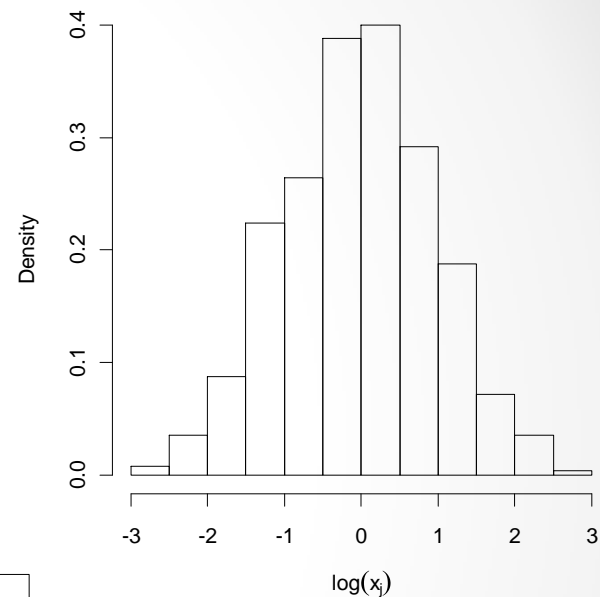


Случай негауссовских показателей

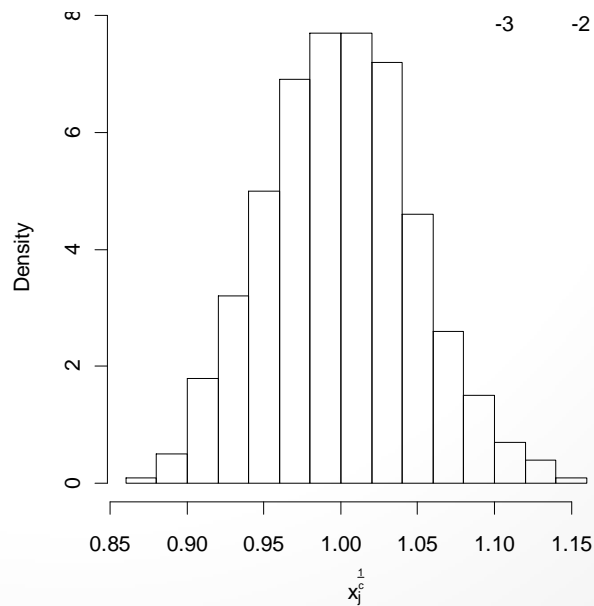
Применяется преобразование переменных



$$\log(x_j + c) \sim$$

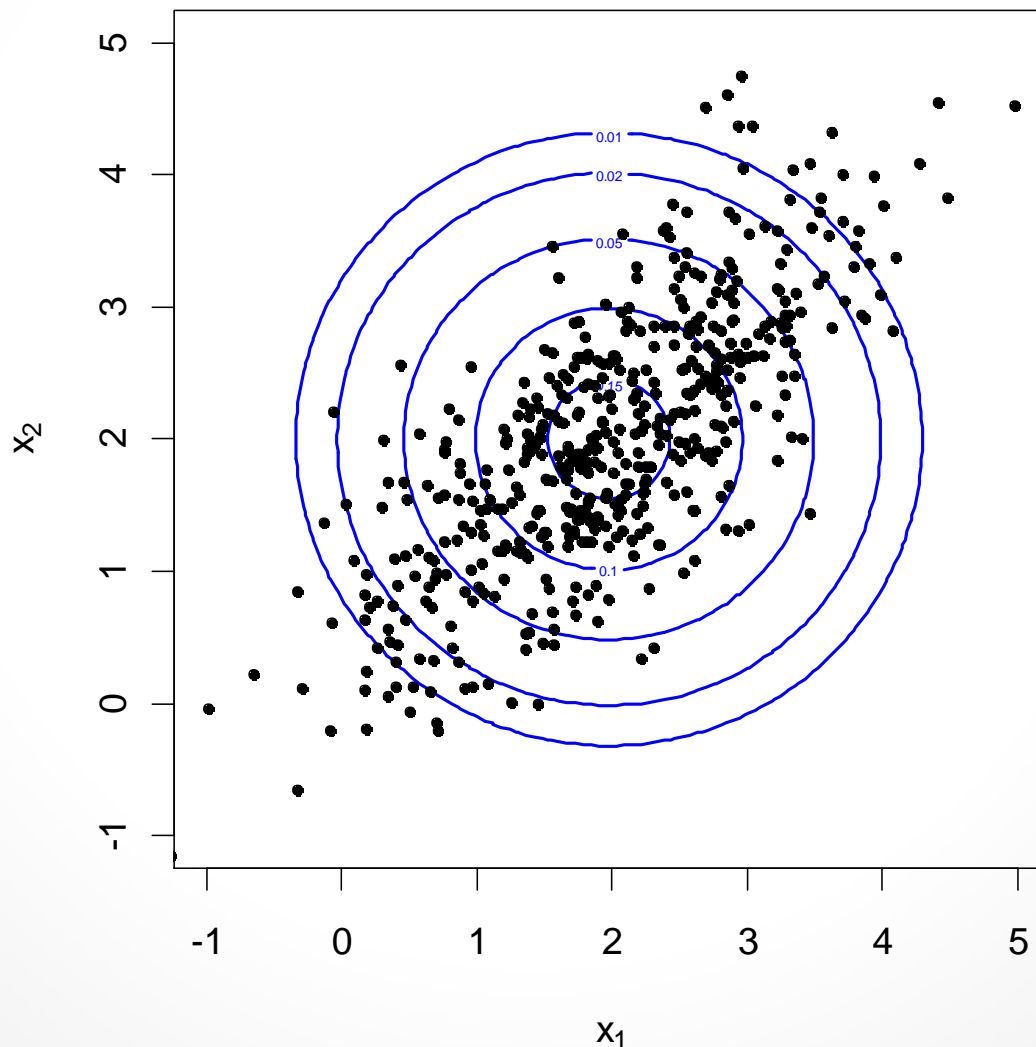


$$x_j^{\frac{1}{c}} \sim$$



Случай зависимых показателей

Для коррелированных признаков произведение «вероятностей» может давать неподходящий результат:

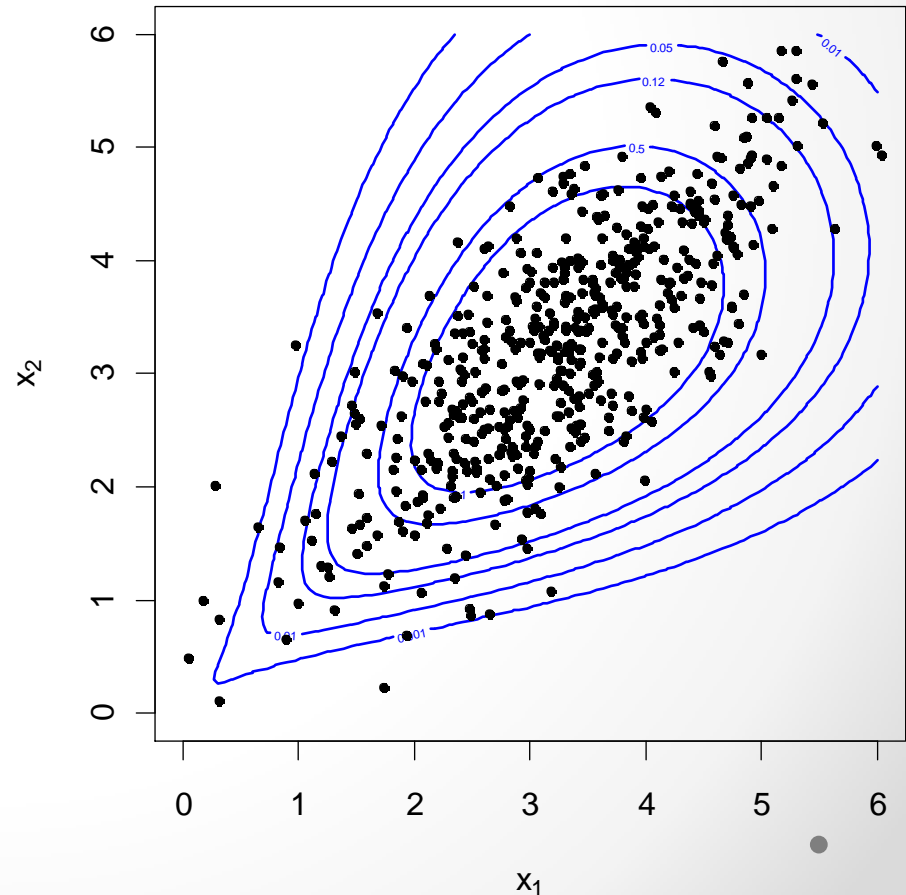


Создание новых признаков

Одна из возможностей справиться с коррелированным случаем — создание нового признака, способного уловить корреляцию

Для показанного примера можно использовать

$$\vec{x}_3 = \left(\frac{\vec{x}_2}{\vec{x}_1} \right)^{\frac{1}{c}}$$



Многомерные распределения

Другой способ заключается в использовании многомерных распределений (например, нормального)

$$\vec{\mu} \in R^n, \Sigma \in R^{n \times n}$$

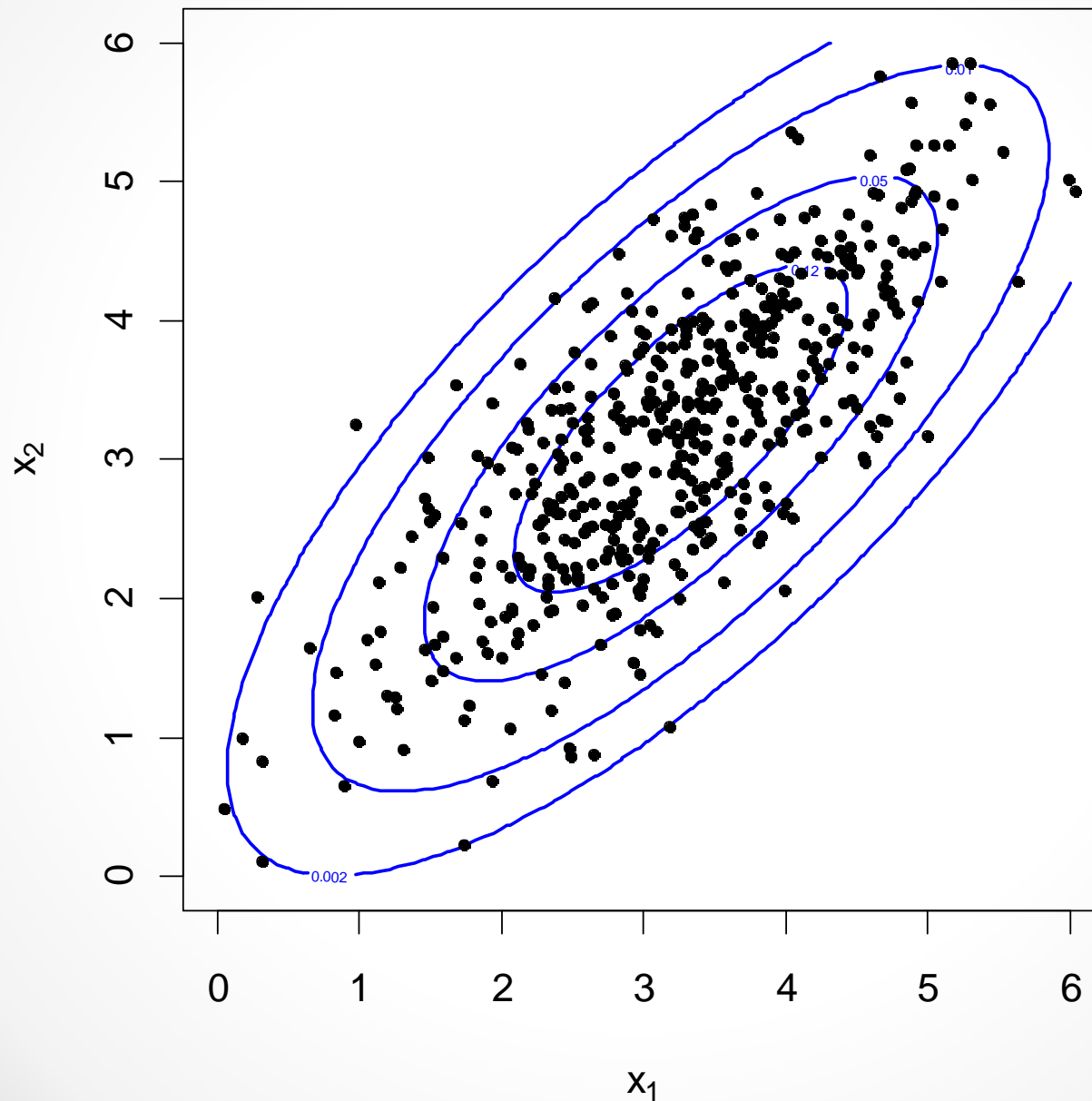
$$p(\vec{x}; \vec{\mu}, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} \exp \left(-\frac{1}{2} (\vec{x} - \vec{\mu})^T \Sigma^{-1} (\vec{x} - \vec{\mu}) \right)$$

ММП-оценки:

$$\hat{\vec{\mu}} = \frac{1}{m} \sum_{i=1}^m \vec{x}^{(i)}, \quad \hat{\Sigma} = \frac{1}{m-1} (X - \hat{\vec{\mu}})^T (X - \hat{\vec{\mu}})$$

В вычислительном плане этот способ — более тяжёлый, чем создание новых признаков

Многомерные распределения



Домашнее задание

Для стабильной работы дата-центра необходимо отслеживание неполадок в работе его компьютеров

В файле «[AD_comp_train.csv](#)» имеются данные о двух характеристиках работы компьютеров: загрузке процессора и использовании оперативной памяти

Вашей задачей является определение машин с необычным режимом работы в тестовой выборке «[AD_comp_test.csv](#)»

Домашнее задание

