

УДК 004.415.532.2

## ОПРЕДЕЛЕНИЕ ПРИЧИНЫ ПЛАВАЮЩЕГО ДЕФЕКТА С ПРИМЕНЕНИЕМ КРИТЕРИЯ ХИ-КВАДРАТ И ОДНОФАКТОРНОЙ МОДЕЛИ ANOVA

Петросян Г.С., Титов В.А.

ФГБОУ ВО «Российский экономический университет имени Г.В. Плеханова», Москва,  
e-mail: grantp@rambler.ru, vitov213@yandex.ru

Управление дефектами является одной из ключевых областей в тестировании программного обеспечения. Существует класс дефектов, которые принято называть плавающими. Одной из наиболее сложных задач при работе с данными дефектами является их локализация. В настоящей работе описываются две модели поиска факторов, оказывающих влияние на частоту проявления плавающего дефекта. Первая модель представляет собой проверку гипотезы о независимости двух случайных величин (проявление дефекта и некоторый внешний фактор) при помощи критерия хи-квадрат. В основе второй модели лежит аппарат одномерного дисперсионного анализа (ANOVA). Каждая из математических моделей сопровождается примерами, а также программным кодом R. С помощью приведенных скриптов на языке R производится ввод исходных данных, тестирование статистических гипотез и построение графиков. Предложенные методы предназначены преимущественно для разработчиков программного обеспечения, также результаты работы могут быть использованы в практической деятельности специалистов по ручному, автоматизированному и нагрузочному тестированию.

**Ключевые слова:** тестирование ПО, плавающие дефекты, тестирование статистических гипотез, таблицы сопряженности, критерий хи-квадрат, одномерный дисперсионный анализ, язык программирования R

## INTERMITTENT SOFTWARE DEFECTS LOCALIZATION USING CHI-SQUARED TEST AND ONE-WAY ANOVA

Petrosyan G.S., Titov V.A.

Plekhanov Russian University of Economics, Moscow, e-mail: grantp@rambler.ru, vitov213@yandex.ru

Defect management is one of the key areas in software testing. There are defects that are referred to as intermittent. One of the most sophisticated activities related to intermittent defects is their localization. This article is dedicated to the development of two models for detecting the external factors, which influence the probability of intermittent defect reoccurrence. In the first model implemented in the article the hypothesis of two random variables (defect occurrence and some external factor) independence is tested using chi-squared criterion. The second model is based on one-way analysis of variance (ANOVA). Each of the mathematical models is provided by hands-on examples and R program code as well. R scripts presented in the article are implemented for working with input data, statistical hypothesis testing and plotting. The suggested mathematical methods are primarily aimed at software developers, also results of the work can be used in practical work of manual test engineers, automation test engineers and performance test engineers.

**Keywords:** software testing, intermittent defects, statistical hypothesis testing, contingency tables, chi-squared test, one-way analysis of variance, R programming language

Дефектом в программном обеспечении принято называть несоответствие ПО установленным или предполагаемым требованиям. Примерами дефектов ПО могут служить неверно рассчитанные платежи по кредитам, сбои в работе карточного процессинга в часы пиковой нагрузки, некорректная тарификация абонентов оператора сотовой связи. Одной из основных задач в тестировании и обеспечении качества ПО является работа с дефектами на всех этапах жизненного цикла: анализ и локализация, описание, проверка исправления. Особо сложной представляется задача анализа плавающих дефектов. Особенность данных дефектов заключается в том, что они воспроизводятся лишь с определенной частотой. Иными словами, отсутствует однозначный сценарий, который приводил бы к проявлению ошибки в 100% случаев.

Работа с плавающими дефектами является одной из наиболее неразработанных проблем в современной литературе по тестированию и обеспечению качества. Также в существующих на данный момент инструментальных средствах тестирования отсутствует функциональность для работы с данным видом дефектов.

Проблема плавающих дефектов подробно рассматривается в работе [2]: дано определение плавающих дефектов, приведена их классификация, рассмотрены основные особенности и риски, связанные с управлением данными дефектами.

Также в работе [2] предлагается решение следующих задач:

- Грамотное описание дефекта.
- Оценка критичности.

Для решения данных задач был использован аппарат теории вероятностей и математической статистики.

Таблица 1

Общий вид таблицы сопряжённости для анализа плавающего дефекта

	Дефект проявился	Дефект не проявился	Сумма
Фактор в состоянии 1	$n_{11}$	$n_{12}$	$n_{1.}$
Фактор в состоянии 2	$n_{21}$	$n_{22}$	$n_{2.}$
Сумма	$n_{.1}$	$n_{.2}$	
Общая сумма			$n$

Таблица 2

Статистика проявления дефекта в рамках примера с системой оценки кредитных рисков

	Дефект проявился	Дефект не проявился	Сумма
MDM система (новая)	12	38	50
MDM система (нецелевая)	7	43	50
Сумма	19	81	
Общая сумма			100

Настоящая статья является продолжением работы [2]. Цель статьи – разработка двух математических моделей для локализации плавающего дефекта. Актуальность статьи объясняется тем, что локализация любого дефекта является первым этапом в процессе его исправления.

Перейдём к рассмотрению первой модели.

Предположим, имеется некоторый внешний фактор, который может находиться только в одном из двух различных состояний. Требуется проверить предположение о том, что данный фактор оказывает влияние на вероятность проявления дефекта.

Пусть  $p_1$  – вероятность проявления дефекта при состоянии 1 внешнего фактора, а  $p_2$  – вероятность проявления дефекта, когда данный фактор находится в состоянии 2.

Проверяется гипотеза  $H_0$ , утверждающая, что вероятности проявления дефекта не отличаются в зависимости от рассматриваемого фактора.

$$H_0 : p_1 = p_2,$$

$$H_1 : p_1 \neq p_2.$$

Для проверки данной гипотезы осуществим несколько попыток воспроизвести дефект при каждом из двух состояний внешнего фактора.

Введём следующие обозначения:

- $n_{i1}$  – количество раз, когда дефект воспроизвёлся при состоянии  $i$  фактора ( $i = 1, 2$ );

- $n_{i2}$  – количество раз, когда дефект не воспроизвёлся при состоянии  $i$  фактора ( $i = 1, 2$ );

- $n_{i.}$  – общее число испытаний при состоянии  $i$  фактора ( $i = 1, 2$ );

- $n_1$  – общее число испытаний, в которых дефект проявился;

- $n_2$  – общее число испытаний, в которых дефект не проявился;

- $n = n_{11} + n_{12} + n_{21} + n_{22}$  – общее число испытаний.

Результаты экспериментов удобно представить в виде таблицы сопряжённости (табл. 1).

Для тестирования гипотезы о равенстве  $p_1 = p_2$  можно воспользоваться критерием хи-квадрат [1]. Данный статистический критерий можно использовать при выполнении условий

$$n > 20, \quad (1)$$

$$\frac{n_{i.}n_{.j}}{n} > 3i, j = 1, 2. \quad (2)$$

Расчётное значение критерия имеет следующий вид:

$$\chi_{\text{расч}}^2 = \frac{n(n_{11}n_{22} - n_{12}n_{21})^2}{(n_{11} + n_{12})(n_{21} + n_{22})(n_{11} + n_{21})(n_{12} + n_{22})}. \quad (3)$$

При  $\chi_{\text{расч}}^2 < \chi^2(1)$  гипотеза  $H_0$  согласуется с результатами наблюдений на уровне значимости  $\alpha$ ;

При  $\chi_{\text{расч}}^2 \geq \chi^2(1)$  гипотеза  $H_0$  отклоняется.

В данных формулах  $\chi^2(1)$  – это критическая точка распределения хи-квадрат, где  $\alpha$  – уровень значимости.

В качестве примера рассмотрим тестирование автоматизированной системы для оценки кредитных рисков. Периодически при попытке загрузки данных о клиентах из смежной MDM-системы (masterdatamanagement) возникает сбой. Система оценки рисков

и MDM-система интегрированы через ESB (enterpriseservicebus). За месяц до обнаружения дефекта была произведена миграция на новую систему MDM, разработанную другим вендором. Проверим предположение о влиянии системы MDM на плавающий дефект. Для этого совершим 50 попыток воспроизвести дефект, затем осуществим переключение на старую MDM и совершим ещё 50 попыток. Результаты эксперимента зафиксируем в табл. 2.

Проверим условия (1) и (2).

$$n = 100 > 20,$$

$$\min_{i,j} \frac{n_{i,j}}{n} = \frac{n_{1,1}}{n} = \frac{19 \times 50}{100} = 9,5.$$

Следовательно, для проверки гипотезы можно воспользоваться критерием хи-квадрат.

$$\chi_{\text{расч}}^2 = \frac{100(12 \times 43 - 7 \times 38)^2}{19 \times 81 \times 50 \times 50} = 1,624,$$

$$\chi_{0,05}^2(1) = 3,84,$$

$$\chi_{\text{расч}}^2 = 1,624 < 3,84 = \chi_{\text{крит}}^2.$$

Из последнего неравенства следует, что гипотеза  $H_0$  принимается. Это означает, что миграция данных о клиентах в новую MDM не оказывает значимого влияния на проявление плавающего дефекта. Дефект, вероятнее всего, относится к системе оценки рисков либо к ESB.

Продемонстрируем решение данной задачи в R при помощи расчета p-значения. Выполним следующий скрипт R [6]:

```
defects <- matrix(c(12, 38, 7, 43), nrow = 2, byrow = TRUE)
chisq.test(defects, correct=FALSE) # тест хи-квадрат
```

В результате исполнения скрипта будут получены следующие результаты (рис. 1).

Данные результаты необходимо интерпретировать следующим образом:

- X-squared – это величина  $\chi_{\text{расч}}^2$ , рассчитанная по формуле (3);

- p-value – это вероятность того, что случайная величина  $\chi^2(1)$  окажется больше либо равна  $\chi_{\text{расч}}^2$ .

Поскольку  $\alpha = 0,05 < 0,2025 = p\text{-value}$ , то гипотеза  $H_0$  принимается.

Проиллюстрируем тестирование гипотезы графически, выполнив следующий скрипт [6]:

```
X_test_statistic <- chisq.test(defects)$statistic # 1. Вычисление расчетного значения
X_critical_value <- qchisq(0.05, 1, lower.tail=FALSE) # 2. Вычисление критического значения
x <- seq(0, 6, by=0.01) # 3. Построение графика плотности распределения
yl <- dchisq(x, 1, ncp = 0, log = FALSE)
plot(x, yl, type="l", xlab="x", ylab="p(x)", ylim=c(0, 1), las=1, bty="n", axes = FALSE)
axis(1, pos=0) # 4. Построение оси OX
axis(2, pos=0, las=1) # 5. Построение оси OY
S <- seq(from = X_critical_value, to = 6, by = 0.001) # 6. Заливка критической области
x <- c(X_critical_value, S, 6)
y <- c(0, dchisq(S, 1, ncp = 0, log = FALSE), 0)
polygon(x, y, col = "grey", lty = 1, border = "black")
points(X_test_statistic, 0, pch=19) # 7. Отметим на графике расчетное значение
text(X_test_statistic, 0, "x^2расч", pos=3)
points(X_critical_value, 0, pch=19) # 8. Отметим на графике критическое значение
text(X_critical_value, 0, "x^2крит", pos=3, offset=0.7)
```

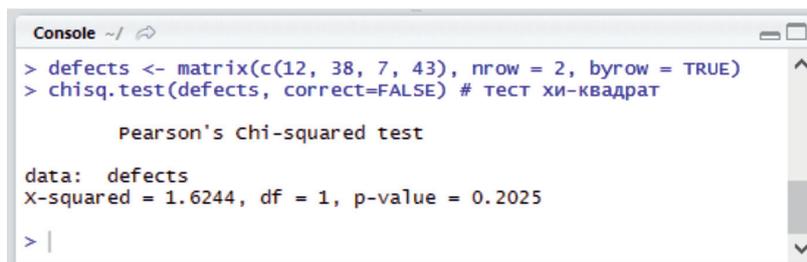


Рис. 1. Результаты исполнения скрипта теста хи-квадрат в среде RStudio

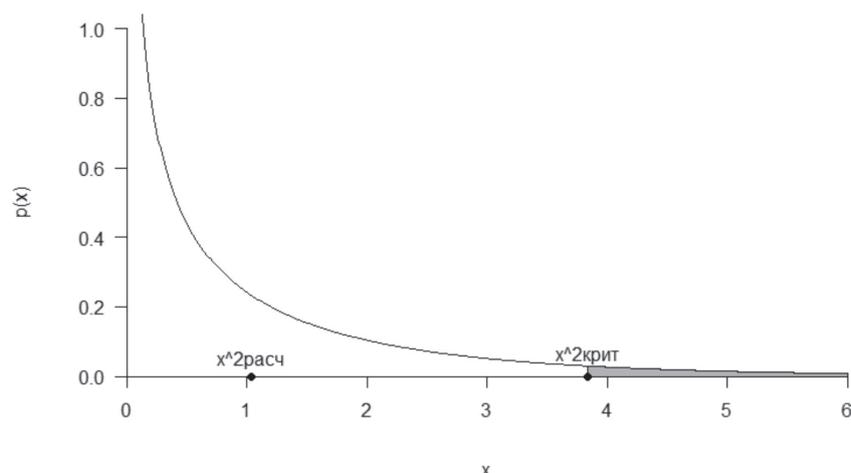


Рис. 2. График плотности вероятности  $\chi^2(l)$  с закрашенной критической областью, с отмеченными на графике расчётным и критическим значениями

В основе второй модели, разработанной в рамках данной статьи, лежит аппарат одномерного дисперсионного анализа (ANOVA).

Рассмотрим внешний фактор, который может находиться в одном из различных состояний. Обозначим через  $m_i$  – среднее число по попыток до воспроизведения дефекта при  $i$  состоянии фактора,  $i = 1, 2, \dots, l$ .

Требуется проверить следующую гипотезу:

$$H_0 : m_1 = m_2 = \dots = m_l,$$

$$H_1 : \exists i, j : m_i \neq m_j.$$

Для проверки гипотезы  $H_0$  осуществим  $n_k$  испытаний для каждого из возможных состояний  $k = 1, 2, \dots, l$  фактора. Результат каждого испытания – число попыток до проявления дефекта.

Введём следующие обозначения:

•  $n_k$  – число наблюдений при состоянии  $k$  фактора;

•  $n = \sum_{k=1}^l n_k$  – общее число наблюдений.

Доказано [4], что:

$$\frac{Q_1 / (l-1)}{Q_2 / (n-l)} \sim F(l-1, n-l),$$

где

•  $Q_1 = \sum_{k=1}^l n_k (\bar{x}_k - \bar{x})^2$  – сумма квадратов

отклонений выборочных средних  $\bar{x}_k$  от общего среднего  $\bar{x}$ ;

•  $Q_2 = \sum_{k=1}^l \sum_{i=1}^{n_k} (x_{ik} - \bar{x}_k)^2$  – сумма квадратов

отклонений наблюдений от выборочных средних групп (внутри групп);

•  $F(l-1, n-l)$  – распределение Фишера с  $l-1$  и  $n-l$  степенями свободы.

Расчётное значение может быть рассчитано по следующей формуле:

$$F_{\text{расч}} = \frac{Q_1 / (l-1)}{Q_2 / (n-l)}. \quad (4)$$

Если  $F_{\text{расч}} < F(l-1, n-l)$ , то гипотеза  $H_0$  не противоречит результатам наблюдений и принимается.

Если  $F_{\text{расч}} \geq F(l-1, n-l)$ , то гипотеза  $H_0$  отклоняется, и следует считать, что среди значений  $m_1, m_2, \dots, m_l$  имеется хотя бы два не равных друг другу.

В формулах выше  $F_\alpha(l-1, n-l)$  – критическая точка распределения Фишера с  $l-1$  и  $n-l$  степенями свободы при уровне значимости  $\alpha$ .

Вернёмся к примеру с риск-системой. Проверим предположение, что на вероятность проявления дефекта оказывает влияние нагрузка на систему (табл. 3).

Таблица 3

Статистика проявления дефекта в зависимости от уровня нагрузки

	Число попыток до проявления дефекта						
	4	2	3	5	5	3	
Уровень нагрузки 1	4	2	3	5	5	3	$n_1 = 6$
Уровень нагрузки 2	6	5	4	7	6	8	$n_2 = 6$
Уровень нагрузки 3	7	9	8	7	8	6	$n_3 = 6$
Уровень нагрузки 4	1	7	3	8	5	7	$n_4 = 6$
							$n = 24$

Осуществим необходимые расчёты для проверки гипотезы  $H_0$ :

$$l = 4,$$

$$l - 1 = 3,$$

$$n - 1 = 20,$$

$$Q_1 = 46,17,$$

$$Q_2 = 59,67,$$

$$F_{\text{расч}} = \frac{46,17/3}{59,67/20} \approx 5,158,$$

$$F_{0,05}(3, 20) = 3,89,$$

$$F_{\text{расч}} = 5,158 > 3,89 = F_{\text{крит}}.$$

$H_0$  в нашем случае отклоняется и принимается гипотеза  $H_1$ . Можно сделать вывод о значимо различных результатах испытаний при различных уровнях нагрузки. Это означает, что нагрузка на систему оказывает влияние на вероятность проявления дефекта.

Построим график плотности распределения  $F(2,15)$ :

```
F_test_statistic<- 5.158# 1. Вычисление расчетного значения
F_critical_value<-qf(0.05, 3, 20, lower.tail = FALSE, log.p = FALSE) # 2. Вычисление критического значения
x=seq(0,7,by=0.01)# 3. Построение графика плотности распределения
y1=df(x, 3, 20, log = FALSE)
plot(x,y1,type="l",xlab="x",ylab="p(x)",ylim=c(0,0.8), las=1, bty="n", axes = FALSE)
axis(1, pos=0) # 4. Построение оси OX
axis(2, pos=0, las=1) # 5. Построение оси OY
S<- seq(from = F_critical_value, to = 6, by = 0.001) # 6. Заливка критической области
x <- c(F_critical_value, S, 6)
y <- c(0, df(S, 3, 20, log = FALSE), 0)
polygon(x, y, col = "grey", lty = 1, border = "black")
points(F_test_statistic,0, pch=19) # 7. Отметим на графике расчетное значение
text(F_test_statistic, 0, "Fрасч", pos=3)
points(F_critical_value,0, pch=19) # 8. Отметим на графике критическое значение
text(F_critical_value, 0, "Fкрит", pos=3, offset=1.3)
```

Выполним следующий скрипт в R [5]:

```
Level1<-c(4,2,3,5,5,3) # вектор данных для уровня нагрузки 1
Level2<-c(6,5,4,7,6,8) # вектор данных для уровня нагрузки 2
Level3<-c(7,9,8,7,8,6) # вектор данных для уровня нагрузки 3
Level4<-c(1,7,3,8,5,7) # вектор данных для уровня нагрузки 4
# F-тест
Combined_Groups<-data.frame(cbind(Level1, Level2, Level3, Level4))
Stacked_Groups<-stack(Combined_Groups)
Anova_Results<-aov(values ~ ind, data = Stacked_Groups)
summary(Anova_Results)
```

Будут получены следующие результаты (рис. 3).

Интерпретация результатов выполнения скрипта:

- Fvalue – величина  $F_{\text{расч}}$ , вычисленная по формуле (4).
- $\text{Pr}( > F )$  – это вероятность того, что случайная величина  $F$  окажется больше либо равна  $F_{\text{расч}}$ .

$$\text{Pr}( > F ) = 0,00837 < 0,05 = \alpha.$$

Снова получаем, что принимается гипотеза  $H_1$ .

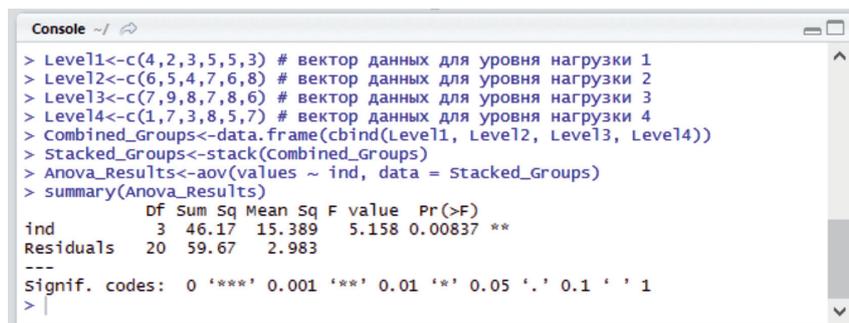


Рис. 3. Результаты выполнения скрипта для тестирования гипотезы в среде RStudio

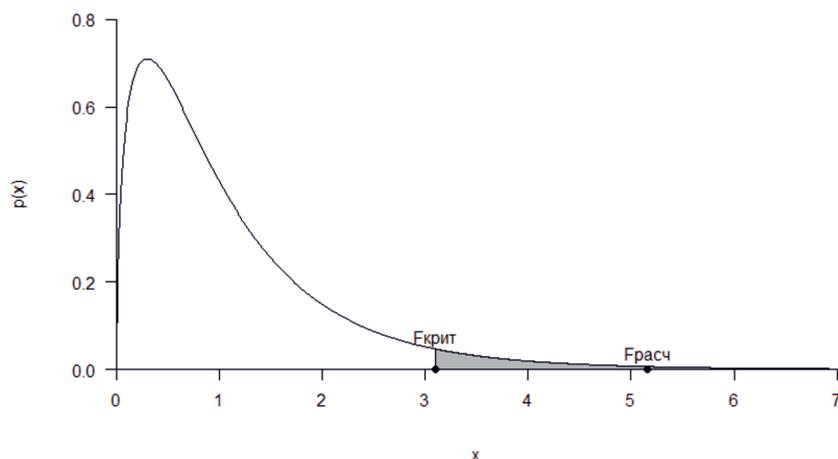


Рис. 4. Плотность распределения  $F(2, 15)$  с закрашенной критической областью, с отмеченными на графике расчётным и критическим значениями

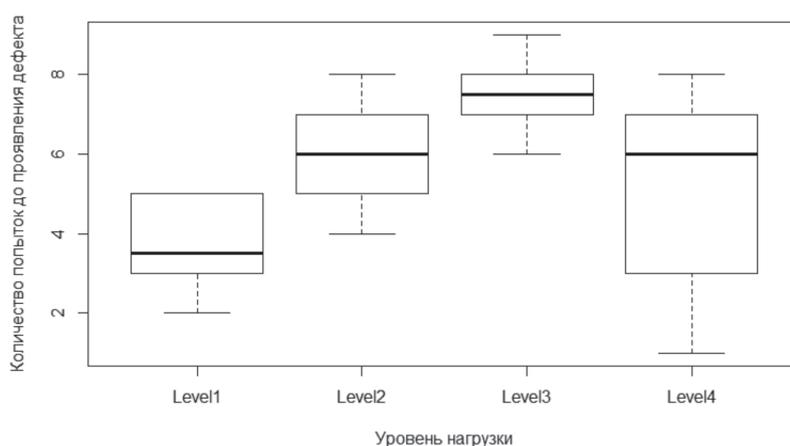


Рис. 5. Диаграмма размаха в рамках примера с уровнями нагрузки

Выполним следующий скрипт R для построения диаграммы размаха, чтобы показать неоднородность наблюдений при разных уровнях нагрузки:

```
boxplot(values ~ ind, data = Stacked_Groups,
        xlab="Уровень нагрузки", ylab="Количество
        попыток до проявления дефекта")
```

Таким образом, разработаны два инструментальных средства на языке программирования R, которые могут быть использованы для локализации плавающих дефектов в цепочке из нескольких автоматизированных систем либо в отдельной системе.

Данные инструментальные средства могут быть использованы в практической деятельности разработчиков и специалистов по тестированию, а также в учебной деятельности. Внедрение данных средств можно рассматривать как этап оптимизации процессов тестирования с использованием

методологии TPINext [3] (рассмотренные в статье модели и инструментальные средства соответствуют десятой ключевой области TPINext – «Управление дефектами»).

#### Список литературы

1. Браунли К.А. Статистическая теория и методология в науке и технике / К.А. Браунли; пер. с англ. М.С. Никулиной, науч. ред. Л.Н. Большевой. – М.: Наука, 1977. – 408 с.
2. Петросян Г.С., Титов В.А. Анализ плавающих дефектов в прикладном программном обеспечении с использованием вероятностных методов // *Фундаментальные исследования*. – 2016. – № 8–2. – С. 262–266.
3. Петросян Г.С., Титов В.А. Оптимизация процессов тестирования автоматизированных систем при помощи модели TPINEXT // *Международный журнал прикладных и фундаментальных исследований*. – 2016. – № 11–5. – С. 998–998.
4. Henry Scheffe. *The Analysis of Variance*. – New York: John Wiley and Sons, 1959. – 267 p.
5. Julian J. Faraway. *Practical Regression and Anovausing R* [Электронный ресурс]. – Режим доступа: <https://cran.r-project.org/doc/contrib/Faraway-PRA.pdf> (дата обращения: 20.02.2017).
6. *The R Manuals* [Электронный ресурс]. – Режим доступа: <https://cran.r-project.org/manuals.html> (дата обращения: 20.02.2017).