Лабораторная работа №5 «Сбор и подготовка данных»

Цель работы:

Исследование наборов данных, представленных на портале data.worldbank.org средствами R

**Порядок выполнения работы.**

1. Изучить теоретические сведения.
2. Получить задание у преподавателя, выполнить типовые задания:
3. Сделать выводы по результатам исследований.
4. Оформить отчет.

1 Выберите вариант задания.

Тематика

1 Agriculture & Rural Development

2 Aid Effectiveness

3 Climate Change

4 Economy & Growth

5 Education

6 Energy & Mining

7 Environment

8 External Debt

9 Financial Sector

10 Gender

11 Health

12 Infrastructure

13 Poverty

14 Private Sector

15 Public Sector

16 Science & Technology

17 Social Development

18 Social Protection & Labor

19 Trade 20 Urban Development

2 Найдите произвольный набор данных на портале data.worldbank.org по тематике, указанной в выбранном варианте задания.

3 Загрузите на компьютер найденный набор данных в формате XLS или csv.

4 На основе набора данных подготовьте выборку, содержащую значения показателя за 2015-2019 для 100 произвольно выбранных стран мира.

5 На основе подготовленной выборки для каждого года провеcти анализ с помощью R:

построить гистограмму распределения, подобрав оптимальное число интервалов разбиения;

проверить гипотезу о принадлежности к нормальному закону распределения;

привести описательную статистику.

для извлечения данных с WORLD BANK можно воспользоваться библиотекой WDI

<https://cran.r-project.org/web/packages/WDI/WDI.pdf>

<https://www.r-bloggers.com/2016/09/analyzing-world-bank-data-with-wdi-googlevis-motion-charts/>

<https://coderoad.ru/10191246/%D0%98%D0%BC%D0%BF%D0%BE%D1%80%D1%82-%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85-%D0%92%D1%81%D0%B5%D0%BC%D0%B8%D1%80%D0%BD%D0%BE%D0%B3%D0%BE-%D0%B1%D0%B0%D0%BD%D0%BA%D0%B0-%D0%B4%D0%BB%D1%8F-%D0%B2%D1%81%D0%B5%D1%85-%D1%81%D1%82%D1%80%D0%B0%D0%BD-%D1%80%D0%B5%D0%B3%D0%B8%D0%BE%D0%BD%D0%B0-%D1%81-%D0%B8%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5%D0%BC-%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%B0>

Тесты проверки на нормальность в R

Существует целый ряд статистических тестов, специально разработанных для проверки нормальности распределения данных. В общем виде проверяемую при помощи этих тестов нулевую гипотезу можно сформулировать так: «Анализируемая выборка происходит из генеральной совокупности, имеющей нормальное распределение». Если получаемая при помощи того или иного теста вероятность ошибки Р оказывается меньше некоторого заранее принятого уровня значимости (например, 0.05), нулевая гипотеза отклоняется.

В R реализованы практически все имеющиеся [тесты на нормальность](https://yadi.sk/i/IAmZaGE7rKCVs) — либо в виде стандарных функций, либо в виде функций, входящих в состав отдельных пакетов. Примером базовой функции является shapiro.test(), при помощи которой можно выполнить широко используемый **тест Шапиро-Уилка**:

shapiro.test(rnorm(500))  
Shapiro-Wilk normality test  
data: rnorm(500)  
W = 0.9978, p-value = 0.7653 # P > 0.05 - нулевая гипотеза не отвергается

Основные классические критерии проверки на нормальность собраны в пакете nortest. Пакет можно установить с CRAN при помощи вызова функции install.packages():

> install.packages(pkgs=c("nortest"))

Подключить установленный пакет можно при помощи функции library():

> library(nortest)

Может возникнуть вопрос: «А зачем столько много разных критериев для проверки одного факта? Нельзя ли выбрать наилучший и всегда его использовать?». Ответ на этот вопрос не утешителен: «В общем случае, к сожалению, нельзя».

Критерий Лиллифорса

Критерий Лиллифорса (Lilliefors) является вариантом известного классического критерия Колмогорова-Смирнова, специально модифицированного для проверки нормальности. Эта модификация существенна. Для проверки гипотезы нормальности нельзя использовать классический непараметрический критерий Колмогорова-Смирнова, реализованный в функции ks.test(). Критерий Лиллифорса реализован в функции lillie.test():

> lillie.test(rnorm(100, mean = 6, sd = 4));  
Lilliefors (Kolmogorov-Smirnov) normality test  
data: rnorm(100, mean = 6, sd = 4)  
D = 0.0463, p-value = 0.8621  
> lillie.test(runif(100, min = 2, max = 4));  
Lilliefors (Kolmogorov-Smirnov) normality test  
data: runif(100, min = 2, max = 4)  
D = 0.0732, p-value = 0.2089

Критерии Крамера-фон Мизеса и Андерсона-Дарлинга

Эти критерии менее известны, но обычно работают гораздо лучше, нежели критерий Лиллифорса. Они реализованы в функциях cvm.test() и ad.test() соответственно:

> cvm.test(rnorm(50, mean = 6, sd = 4));  
Cramer-von Mises normality test  
data: rnorm(50, mean = 6, sd = 4)  
W = 0.0321, p-value = 0.8123  
> ad.test(runif(50, min = 2, max = 4));  
Anderson-Darling normality test  
data: runif(50, min = 2, max = 4)  
A = 1.5753, p-value = 0.0004118

Критерий Шапиро-Франсиа

Этот критерий работает достаточно хорошо в большинстве не очень «сложных» случаев. Получить p-значение можно посредством функции sf.test():

> sf.test(rexp(50, rate = 2));  
Shapiro-Francia normality test  
data: rexp(50, rate = 2)  
W = 0.7803, p-value = 2.033e-06

Критерий хи-квадрат Пирсона

В отличие от задач проверки пропорций, критерий хи-квадрат обычно очень плохо работает в задачах проверки распределения на нормальность. Вероятность ошибки второго рода очень велика для достаточно широкого класса альтернативных распределений. В связи с этим, использовать его не рекомендуется. Тем не менее реализация его предоставлена функцией pearson.test(). У этой функции есть булевская опция adjusted, которая позволяет внести поправки в p-значение из-за наличия двух неизвестных параметров. Рекомендуемая последовательность действий такая: получить два p-значения, одно, соответствующее adjusted=TRUE, второе — adjusted=FALSE. Истинное p-значение обычно находится между. Кроме того, полезно поварьировать объем выборки и посмотреть, насколько сильно меняется p-значение. Если влияние объёма выборки сильное, то от использования критерия стоит отказаться во избежание ошибок.

> pearson.test(rnorm(50, mean = 6, sd = 4));  
Pearson chi-square normality test  
data: rnorm(50, mean = 6, sd = 4)  
P = 5.2, p-value = 0.6356  
> pearson.test(runif(50, min = -1, max = 1));  
Pearson chi-square normality test  
data: runif(50, min = -1, max = 1)  
P = 7.6, p-value = 0.3692