Archives of Current Research International
5(4): 1-9, 2016, Article no.ACRI.29429
*ISSN: 2454-7077*

**SCIENCEDOMAIN** *international*
www.sciencedomain.org

# Fast Graphs of Statistical Distributions Using $R$

**Steven T. Garren**[1*]

[1] *Department of Mathematics and Statistics, James Madison University, Harrisonburg, VA 22807, USA.*

***Author's contribution***

*The sole author designed, analyzed and interpreted and prepared the manuscript.*

***Article Information***

DOI: 10.9734/ACRI/2016/29429
*Editor(s):*
(1) Gongxian Xu, Associate Professor, Department of Mathematics, Bohai University, Jinzhou, China.
*Reviewers:*
(1) Olaiju Olusegun Adeyemi, Federal Polytechnic Ilaro, Nigeria.
(2) Wiktor Ejsmont, Wrocaw University of Economics, Ulica Komandorska, Poland.
Complete Peer review History: http://www.sciencedomain.org/review-history/16541

***Received: 9th September 2016***
***Accepted: 3rd October 2016***
***Published: 13th October 2016***

***Original Research Article***

## ABSTRACT

When teaching elementary statistics courses, instructors may find themselves frequently drawing by hand the probability density functions of normal and *t* distributions. The author presents the *R*-package fastGraph, containing new functions in *R*, allowing graphs of probability density functions and cumulative distribution functions to be drawn on the computer with few keystrokes. These functions also allow shading under the curve of the probability density function to illustrate concepts such as *p*-values and critical values. An additional function which fits the line for simple linear regression is also introduced.

*Keywords: Linear regression; normal distribution; power; p-value; t distribution.*

**2010 Mathematics Subject Classification:** 62-09

# 1 INTRODUCTION

Functions in *R* [1] written by the author are available online in the *R*-package named fastGraph through CRAN, so the reader can quickly generate graphs of statistical distributions and illustrate *p*-values. These quick graphs in *R* are especially useful in the classroom and are also useful when writing research reports. These functions are user-friendly, even for undergraduate students who are neophytes to *R*. If *R* is not downloaded on a computer, such

*Corresponding author: E-mail: garrenst@jmu.edu;

as a faculty classroom computer, the instructor can still access *R* using *Rweb* via http://pbil.univ-lyon1.fr/Rweb/. Many commonly-used elementary and calculus-based statistics textbooks use graphs for illustrating concepts [2],[3],[4],[5],[6]. In general, *R*-packages are quite useful, noting the popularity of *R* among statisticians, as exemplified by the existence of many textbooks focusing on *R* [7],[8]. Furthermore, some introductory statistics textbooks also focus on *R* [9],[10],[11]. Applications of *R* include experimental design [12], analyzing means [13], dose-response analysis [14], big data [15], and water resources [16], among many more.

The two main functions used herein are called `plotDist` and `shadeDist`. Additional functions used are `shadePhat` and `plotLine`. These four functions are available by typing

```
> install.packages( "fastGraph" )

> # Next, select a mirror site.

> library( fastGraph )
```

within *R*. Examples from executing these functions are available by typing

```
> example( fastGraph )

> example( plotDist )

> example( shadeDist )

> example( shadePhat )

> example( plotLine )
```

within *R*.

The function `plotDist` plots probability density functions (pdf*s*) and cumulative distribution functions (cdf*s*), and is described in Section 2. The function `shadeDist` shades area under the curve of a probability density function, while computing the corresponding probability, and is described in Section 3. Additional useful classroom-type examples based on `plotDist` and `shadeDist` are given in Section 4. The function `shadePhat`, which illustrates probabilities based on a sample proportion, $\widehat{p}$, and the function `plotLine`, which fits a simple linear regression line to a scatterplot and computes the equation of the fitted line, are described in Section 5. A brief discussion is given in Section 6.

## 2 USING THE $R$-FUNCTION plotDist

The function `plotDist` can plot one, two, or three probability density functions or cumulative distribution functions on the same graph. Both continuous and discrete distributions are permitted. The basic format is

```
> plotDist( distA, parmA1, parmA2, distB,

parmB1, parmB2, distC, parmC1, parmC2 )
```

where `distA` with parameters `parmA1` and `parmA2` is the distribution printed in black, `distB` with parameters `parmB1` and `parmB2` is the distribution printed in red, and `distC` with parameters `parmC1` and `parmC2` is the distribution printed in green. The default value of `distA` is `"dnorm"` (including the quotes).

When the `parm` values are not specified, then the default values are used. In general, the format of `plotDist` consists of the first distribution along with two parameters, the second distribution along with two parameters, and then the third distribution along with two parameters. For example, both

```
> plotDist( )   # Figure 1a
```

and

```
> plotDist( "dnorm", 0, 1 )  # Figure 1a
```

generate the graph of the standard normal $N(\mu = 0,\ \sigma = 1)$ pdf [17], as shown in Fig. 1a.

The endpoints of the graph are automatically determined by the `plotDist` function, based on small tail probabilities. However, these default values of the endpoints may be overridden by the user, when the user specifies values of `xmin` and `xmax`. Additional modifications to the graph may be made as well, such as the main title (`main`), the label on the $x$-axis (`xlab`), and the colors (`col`).

On the same graph, we generate the pdf*s* of the standard normal distribution in black, the $t_2$ distribution in red, and the $t_5$ distribution in green, using the command

```
> plotDist( , , , "dt", 2, , "dt", 5 )
# Figure 1b
```

noting that the default values of `distA`, `parmA1`, and `parmA2` are `"dnorm"`, `0` (which is $\mu$), and `1` (which is $\sigma$), respectively, as shown in Figure 1b. Therefore, the above command is equivalent to the

```
> plotDist( "dnorm", 0, 1, "dt", 2, 0,
"dt", 5 ) # Figure 1b
```

command. Without using the `plotDist` command, a similar graph can be generated using the following five commands:

```
> dt2 = function( x ){ dt(x,2) }
> dt5 = function( x ){ dt(x,5) }
> plot( dnorm, -4.5, 4.5 )
> curve( dt2, add=TRUE, col="red" )
> curve( dt5, add=TRUE, col="darkgreen" )
```

Hence, the `plotDist` command provides an efficient method for graphing distributions, especially for faculty in a classroom setting, and also for students.

Three or more parameters of a distribution can be defined by setting `parmA1` (or `parmB1` or `parmC1`) to be a vector of parameters. For example, the command

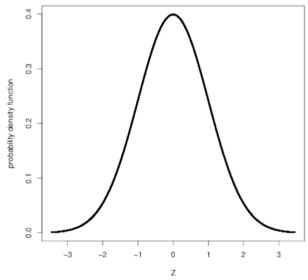
```
> plotDist( "df", 5, 20, "df",
c( 5, 20, 2 ) )   # Figure 1c
```

generates the pdfs of a central $F_{5,20}$ distribution in black, and simultaneously generates a non-central $F_{5,20}$ distribution with non-centrality parameter 2 in red, as shown in Figure 1c.

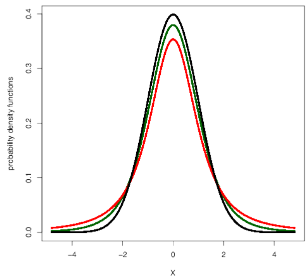
Cumulative distribution functions can also be constructed. The command

```
> plotDist( "pnorm", , , "pcauchy" )
# Figure 1d
```

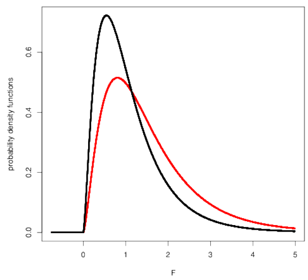
generates the cumulative distribution functions of the standard normal in black, and simultaneously generates the standard Cauchy in red, as shown in Fig. 1d.



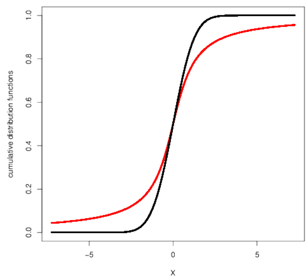

**(a)** Standard normal pdf




**(b)** Standard normal, $t_2$, and $t_5$ pdf$s$




**(c)** $F_{5,20}$ and noncentral $F_{5,20,2}$ pdf$s$




**(d)** Normal and Cauchy cdf$s$

**Fig. 1. Using the `plotDist` command**

## 3 USING THE $R$-FUNCTION shadeDist

The function `shadeDist` plots a continuous or discrete pdf, and shades in a particular region illustrating probability. The function even states the numerical value of this probability on the graph, and hence is especially useful for illustrating *p*-values. The basic format is

```
> shadeDist( xshade, ddist, parm1, parm2,
lower.tail )
```

The argument `xshade` is a scalar or a vector of size two, and specifies the region to be shaded. The logical argument `lower.tail` (whose default value is TRUE) works in conjunction with `xshade` by specifying whether or not the lower tail should be shaded to denote probability. The pdf is defined by `ddist` with parameters `parm1` and `parm2`, with `"dnorm"` as the default distribution. Similar to `plotDist`, the function `shadeDist` also allows additional modifications, such as `xmin`, `xmax`, `main`, and `xlab`.

Applying `shadeDist` to the $N(\mu = 50,\ \sigma = 10)$ pdf based on the commands

```
> shadeDist( c( 40, 60 ), "dnorm", 50,
10, lower.tail=FALSE )    # Figure 2a
```

```
> shadeDist( c( 40, 60 ), "dnorm", 50,
10 )    # Figure 2b
```

```
> shadeDist( 40, "dnorm", 50, 10 )
# Figure 2c
```

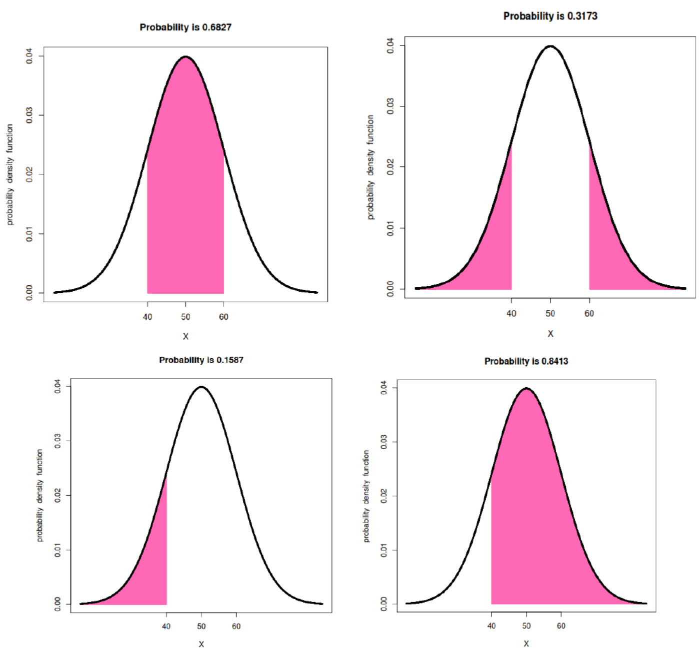
```
> shadeDist( 40, "dnorm", 50, 10,
lower.tail=FALSE )    # Figure 2d
```

produces Figures 2a-d. The above commands can be abbreviated even further, by noting that `"dnorm"` is the default value of `ddist` and that `lower.tail` is the fifth argument in `shadeDist`.

As an example without using `shadeDist`, a graph very similar to Figure 2c can be generated using the following commands:

```
> f = function( x ){ dnorm( x, 50, 10 ) }
```

```
> main = paste( "Probability is",
format( pnorm(40,50,10), digits=4 ) )
```




**(c)** Left-sided probability  **(d)** Right-sided probability

**Fig. 2. Using the `shadeDist` command for a Normal$(\mu = 50,\ \sigma = 10)$ distribution**

```
> plot( f, 20, 80, xlab="X",
ylab="probability density function",
main=main )
```

```
> curve( f, 20, 40, 1001, TRUE, "h",
col="hotpink" )
```

```
> curve( f, 20, 80, 1001, lwd=3,
add=TRUE )
```

Hence, the function `shadeDist` can save the user a lot of keystrokes.

# 4 ADDITIONAL EXAMPLES USING THE *R*-FUNCTIONS plotDist AND shadeDist

The following examples provide useful illustrations, many of which are appropriate for most elementary statistics courses. Figure 3a compares the pdf*s* of $\chi_1^2$, $\chi_2^2$, and $\chi_3^2$ distributions using

```
> plotDist( "dchisq", 1, , "dchisq", 2, ,
"dchisq", 3 )   # Figure 3a
```

while Figure 3b uses the command

```
> shadeDist( 5.8, "dchisq", 4,
lower.tail=FALSE )   # Figure 3b
```

to illustrate that the *p*-value based on $P(\chi_4^2 > 5.8)$ is 0.2146. Figure 4a illustrates the $N(\mu = 40,\ \sigma = \sqrt{24})$ approximation to a Binomial$(n = 100,\ p = 0.4)$ distribution via

```
> plotDist( "dnorm", 40,
sqrt(100*0.4*0.6), "dbinom", 100, 0.4 )
# Figure 4a
```

while Figure 4b illustrates the $N(\mu = 5,\ \sigma = \sqrt{5})$ approximation to a Poisson$(\mu = 5)$ distribution via the

```
> plotDist( "dnorm", 5, sqrt(5), "dpois",
5 )   # Figure 4b
```

command. Figure 4c compares three Binomial$(n = 100,\ p)$ pdf*s* using

```
> plotDist( "dbinom", 100, 0.4, "dbinom",
100, 0.5, "dbinom", 100, 0.7 )
# Figure 4c
```

for $p$ = 0.4, 0.5, and 0.7, respectively. Figure 4d compares three normal pdf*s* using the

```
> plotDist( "dnorm", 50, 5, "dnorm", 50,
10, "dnorm", 55, 5 )   # Figure 4d
```

command.

The size and power (based on the non-centrality parameter of 3) of a right-sided *t*-test for 15 degrees of freedom are illustrated using

```
> plotDist( "dt", 15, , "dt", 15, 3 )
# Figure 5a
```

```
> shadeDist( 1.753, "dt", 15,
lower.tail=FALSE )   # Figure 5b
```

```
> shadeDist( 1.753, "dt", 15, 3,
lower.tail=FALSE )   # Figure 5c
```

in Figures 5a-c.

# 5 ADDITIONAL *R*-FUNCTIONS: shadePhat AND plotLine

The function `shadePhat` effectively applies `shadeDist` to a sample proportion, $\widehat{p} = X/n$, where $X \sim$ Binomial$(n, p)$. The basic format is

```
> shadePhat( xshade, size, prob,
lower.tail )
```

noting that the functions `shadePhat` and `shadeDist` use the same definitions of the arguments `xshade` and `lower.tail`. The arguments `size` and `prob` are the values of $n$ and $p$, respectively. For $n = 10$ and $p = 0.4$, the commands

```
> shadePhat( 0.2, 10, 0.4 )  # Figure 6a
```

```
> shadeDist( 2, "dbinom", 10, 0.4 )
# Figure 6b
```

```
> shadePhat( c(0.2, 0.5), 10, 0.4 )
# Figure 6c
```

```
> shadeDist( c(2, 5), "dbinom", 10, 0.4 )
# Figure 6d
```

illustrate $P(\widehat{p} \le 0.2) = 0.1673$, $P(X \le 2) = 0.1673$, $P(\widehat{p} \le 0.2) + P(\widehat{p} > 0.5) = 0.3335$, and $P(X \le 2) + P(X > 5) = 0.3335$ in Figures 6a-d, respectively.

This function `shadePhat` also can be accessed, by executing `shadeDist` and setting the argument `ddist` equal to `"dprop"`. For example, Figure 6a also can be produced using the command

```
> shadeDist( 0.2, "dprop", 10, 0.4 )
# Figure 6a
```

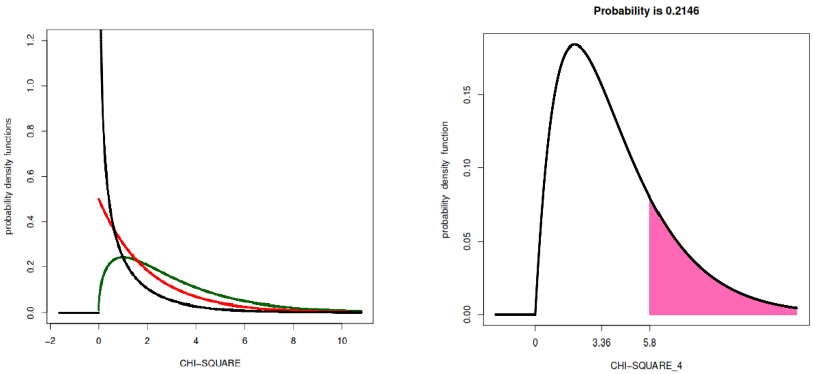
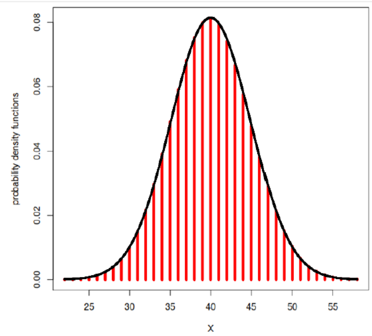
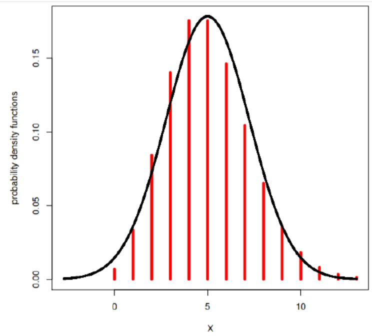


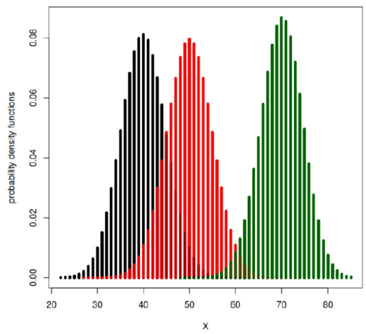

**Fig. 3. Using the `plotDist` and `shadeDist` commands for $\chi^2$ distributions**



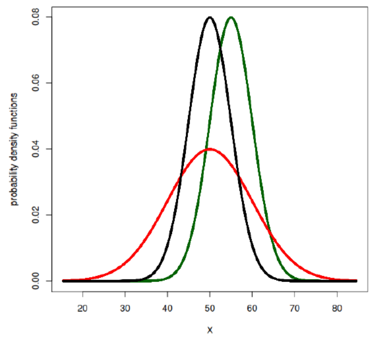

**(a)** Normal approximation to binomial pdf




**(b)** Normal approximation to Poisson pdf

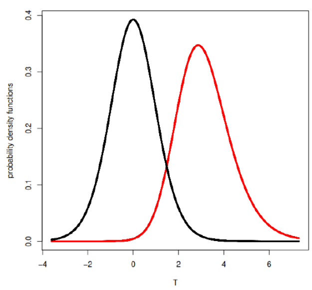



**(c)** Three binomial pdf$s$

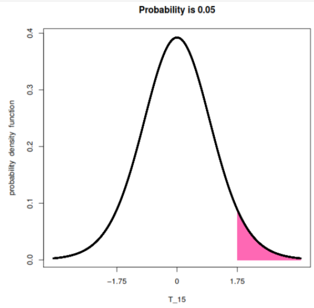



**(d)** Three normal pdf$s$

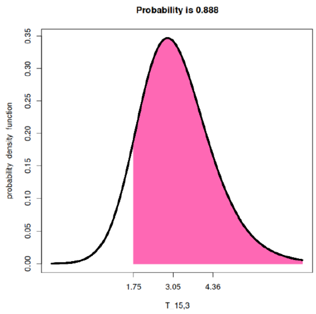
**Fig. 4. Using the `plotDist` command to produce multiple pdf*s* on the same graph**




**(a)** Central and non-central $t_{15}$ pdf$s$

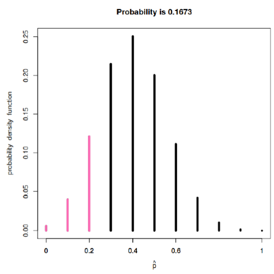



**(b)** Size using central $t_{15}$ pdf

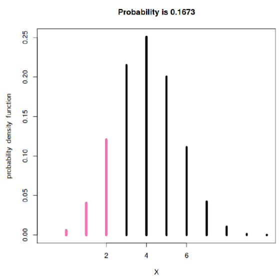



**(c)** Power using non-central $t_{15,3}$ pdf

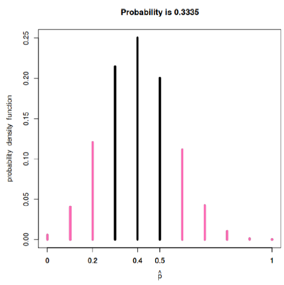
**Fig. 5. Using the `plotDist` and `shadeDist` commands to illustrate size and power (based on the non-centrality parameter of 3) of a right-sided $t$-test with 15 degrees of freedom**



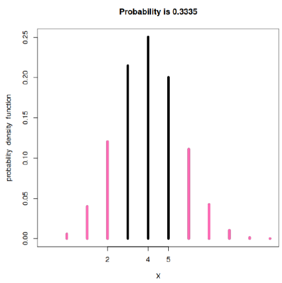

**(a)** One-sided probability for $\widehat{p}$




**(b)** One-sided probability for binomial




**(c)** Two-sided probability for $\widehat{p}$




**(d)** Two-sided probability for binomial

**Fig. 6. Using the `shadePhat` and `shadeDist` commands to compare the pdf of a sample proportion, $\widehat{p}$, with the pdf of a binomial, using $n = 10$ and $p = 0.4$ for both pdf$s$**

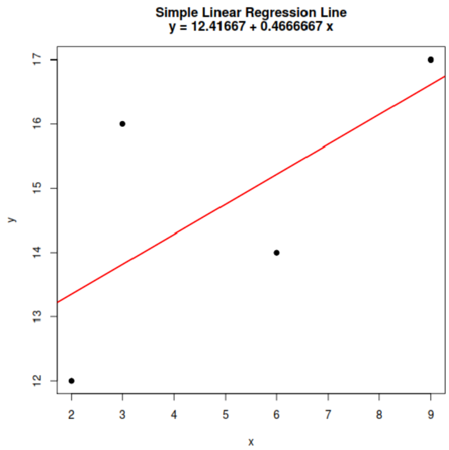



**Fig. 7. Using the `plotLine` command to list the equation and fit the line based on simple linear regression**

The function `plotLine` graphs a two-dimensional scatterplot along with its fitted linear regression line, and also states the equation of the line. The basic format is

```
> plotLine( x, y )
```

for vectors `x` and `y`. An example using data $(x, y) = \{(2, 12), (3, 16), (6, 14), (9, 17)\}$ based on the command

```
> plotLine( c( 2, 3, 6, 9 ),
c( 12, 16, 14, 17 ) )   # Figure 7
```

is shown in Figure 7. Suppose the data are in a data frame, such as the following:

```
> x = c( 2, 3, 6, 9 ) ; y = c( 12, 16,
14, 17 ) ; d = data.frame( x=x, y=y )
```

Then, Figure 7 can again be generated, this time using the format

```
> plotLine( y ~ x, data = d ) # Figure 7
```

Other CRAN packages, such as `ggplot2`, `openintro`, and `car`, plot the fitted regression line on a scatterplot as well, based on the following commands:

```
> library( ggplot2 )

> qplot( data = d, x = x, y = y ) +
stat_smooth( method = "lm", se = FALSE )

> library( openintro )

> lmPlot( d$x, d$y )

> library( car )

> lmPlot( y ~ x, data = d )
```

However, the function `plotLine` is somewhat unique in comparison to the three above methods, in that `plotLine` lists the equation of the least squares line in the title of the scatterplot.

# 6 CONCLUSION

The functions `plotDist`, `shadeDist`, `shadePhat`, and `plotLine` are available in the fastGraph package, for quickly printing graphs using the statistical software *R*. Printing a probability density function employs code used intrinsically in *R*, such as `dnorm`, `dt`, `dchisq`, `df`, `dcauchy`, `dbinom`, and `dpois`, corresponding to the probability density functions of normal, $t$, $\chi^2$, $F$, Cauchy, binomial, and Poisson distributions, respectively. Calculating and illustrating *p*-values also may be performed using the author's functions, for either continuous or discrete distributions.


# ACKNOWLEDGEMENTS

The author is very thankful to two anonymous reviewers and an editor, whose helpful suggestions added greatly to the quality of this paper.


# COMPETING INTERESTS

Authors has declared that no competing interests exist.

# References


[1] R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria ; 2014. Available: http://www.R-project.org/

[2] Agresti A, Franklin C, Klingenberg B. Statistics, The Art and Science of Learning from Data. 4th ed., Pearson, New York; 2017.

[3] Devore JL. Probability and Statistics for Engineering and the Sciences. 9th ed., Cengage Learning, Boston; 2016.

[4] Moore DS, Notz WI, Fligner MA. The Basic Practice of Statistics. 7th ed., W. H. Freeman, New York; 2015.

[5] Navidi W, Monk B. Essential Statistics. McGraw-Hill, New York; 2013.

[6] Peck R, Olsen C, Devore J. Introduction to Statistics and Data Analysis. 5th ed., Cengage, Boston; 2015.

[7] Berridge DM, Crouchley R. Multivariate generalized Linear Mixed models using R, Taylor and Francis, New York; 2011.

[8] Bretz F, Hothorn T, Westfall P. Multiple comparisons using R, Taylor and Francis, New York; 2011.

[9] Field A, Miles J, Field Z. Discovering statistics using R, Sage, London; 2012.

[10] Verzani J, Using R. for Introductory Statistics. 2nd ed., Taylor and Francis, New York; 2014.

[11] Ware WB, Ferron JM, Miller BM. Introductory Statistics: A Conceptual Approach Using R. Taylor and Francis, New York; 2013.

[12] Eudey TL, Kerr JD, Trumbo BE. Using R to simulate permutation distributions for some elementary experimental designs. Journal of Statistics Education. 2010;18(1):1-30. Accessed 25 September 2016. Available: http://ww2.amstat.org/publications/jse/v18n1/eudey.pdf

[13] Pallmann P, Hothorn LA. Analysis of means: a generalized approach using R. Journal of Applied Statistics. 2016;43(8):1541-1560. Accessed 25 September 2016. Available: http://dx.doi.org/10.1080/02664763.2015.1117584

[14] Ritz C, Baty F, Streibig JC, Gerhard D. Dose-Response Analysis Using R. Plos One. 2015;10(12):1-13. Accessed 25 September 2016. Available: http://journals.plos.org/plosone/article?id=10.1371%2Fjournal.pone.0146021

[15] Sharmila K, Manickam SAV. Efficient prediction and classification of diabetic patients from bigdata using R. International Journal of Advanced Engineering Research and Science. 2015;2(9):56-58. Accessed 25 September 2016. Available: http://www.academia.edu/16393703/Efficient_Prediction_and_Classification_of_Diabetic_Patients_from_bigdata_using_R

[16] Cisty M, Celar L. Using R in Water Resources Education. International Journal for Innovation Education and Research. 2015;3(10):97-117. Accessed 25 September 2016. Available: http://www.ijier.net/index.php/ijier/article/view/89

[17] Ejsmont W. A characterization of the normal distribution by the independence of a pair of random vectors. Statistics and Probability Letters. 2016;114:1-5.



*Peer-review history:*
*The peer review history for this paper can be accessed here:*
*http://sciencedomain.org/review-history/16541*