Estimation procedure in Monthly retail trade survey in Serbia using R software

Sofija SUVOCAREV (sofija.suvocarev@stat.gov.rs)
Statistical Office of the Republic of Serbia

ABSTRACT

The objective of Monthly retail trade survey (MRTS), based on the sample and on the VAT reports received from Tax administration, is to provide the data on turnover of goods in retail trade in order to measure monthly changes in turnover. Indices, totals and standard errors are calculated for territory of the Republic of Serbia and the territorial units (NUTS 2). For the Republic of Serbia, these parameters are calculated also by two groups and eight classes of NACE Rev. 2. The calculation is based on stratified simple random sample. This paper shows how estimation procedure for these parameters is implemented in R software.

Keywords: *Retail trade, R software, parameter estimation.*

INTRODUCTION TO THE SURVEY

The aim of the survey is to provide data on turnover in order to measure monthly changes in turnover. Indices are calculated for territory of the Republic of Serbia and the territorial units (NUTS 2) for current to previous month. New demand in 2013 is to calculate indices Republic of Serbia by two classes and 8 groups of NACE Rev. 2 which belong to the division 47. They are shown in the Table 1.

Two classes and 8 groups of NACE Rev. 2 which belong to the division 47

Table no. 1.

4711	Retail sale in non-specialized stores with food, beverages or tobacco predominating
4719	Other retail sale in non-specialized stores
472	Retail sale of food, beverages and tobacco in specialized stores
473	Retail sale of automotive fuel in specialized stores
474	Retail sale of ICT equipment in specialized stores
475	Retail sale of other household equipment in specialized stores
476	Retail sale of cultural and recreation goods in specialized stores
477	Retail sale of other goods in specialized stores
478	Retail sale via stalls and markets
479	Retail trade not in stores, stalls or markets

POPULATION AND FRAME

The basic set of units for MRTS is created according to the data of the Statistical business register (SBR), in January of the current year. Units are all active enterprises from division 47, and in addition, 44 enterprises whose main activity was not retail trade, but are also engaged in retail trade activity.

The basic set consists of 5 parts, indicated by auxiliary variable DEO:

- small, medium and large enterprises according to financial report, DEO=1, 2 and 3, respectively;
- budget enterprises, DEO=B;
- enterprises included on purpose, whose main NACE Rev. 2 activity is not in 47, DEO=E.

The final frame for the year 2013 consists of 4641 units:

- all budget enterprises and those included on purpose;
- all medium and large enterprises with turnover >0;
- small enterprises with turnover >=1300 thousands RSD.

Table 2. shows fraction of units, turnover and number of employees by parts of the frame 2013, according to the variable DEO.

Fraction of units, turnover and number of employees by parts of the frame 2013

Table no. 2.

Description of the DEO	DEO	No. of units (%)	Turnover (%)	No. of employees (%)
Total		74,9	100,0	98,5
Small enterprises	1	73,7	99,7	93,8
Medium enterprises	2	99,3	100,0	99,8
Large enterprises	3	100,0	100,0	100,0
Budget enterprises	B	100,0	-	100,0
Enterprises that are not in NACE Rev. 2 division 47	E	100,0	100,0	100,0

STRATIFICATION AND ALLOCATION

Stratification of the frame units according to the part of the frame to which they belong is to five classes (defined by the values of the variable DEO):

- 1, 2, 3, B and E

Further stratification of the parts DEO=1, 2, 3, B is according to
- size:
  - into smaller (cens_m=0) and
  - larger (cens_m=1)
- NACE Rev 2 activity.

Census strata in the MRST are those for which:
- DEO=3,E or
- DEO=1, 2, B and cens_m=1.

Allocation is carried out by applying the Bethel algorithm. Total number of strata is 56 and 39 of them are census strata.

ESTIMATION PROCEDURE

The Horvitz-Thompson estimations of totals and indices are calculated pursuant to the standard procedure for stratified sample with random selection of units within the stratum. The R survey package is used for estimation procedure. Main concepts and parts of R code are presented in this chapter, and the complete code is given in the Annex.

Realized sample (*realiz_uzorak*) contains following variables:

- mb - statistical unit ID
- malo - NACE Rev. 2 classes
- vs – indicator of the type of turnover, takes six different values (1 = total turnover of the enterprise, 2 = turnover of the enterprise based on the retail trade, 3,4,5 and 6 = turnover of the enterprise based on the retail trade by territorial units (NUTS 2))
- ppromet – turnover in the previous month
- ipromet – turnover in the current month
- ppdv – VAT in the previous month
- ipdv – VAT in the current month
- pprometb – turnover in the previous month without VAT
- iprometb – turnover in the current month without VAT

Selected sample (*plan_uzorak*) contains following variables:

- mb - statistical unit ID
- mstrat – stratum ID
- mnh – size of the stratum
- mpnh – number of statistical units allocated in the stratum
- odziv – response information for statistical units (1 = unit has reported turnover and belongs to realized sample, 2 = refused to fill the questionnaire, 3 = not found on the address, 4 = closed, 5 = stationary / not operating, 6 = activity not in scope of the survey, 7 = in bankruptcy, 8 = in liquidation and 9 = other, state the reason)

If the variable odziv takes one of the values 1, 5, 6 or 7 for certain statistical unit, then that statistical unit is considered as part of the realized sample.

R code for the estimation procedure can be described in few steps. First of all, necessary packages are loaded using function library(), realized and selected sample which are given in excel file are read, using function readWorksheetFromFile(). After merging this two files merge() and the creation of new variable malo1 which is needed for domain estimation, path for excel file in which the results are going to be exported is defined file. path(). Subsets of the realized sample according to the type of turnover is taken subset(), for each subset is defined sampling design svydesign() and weights for such defined sampling design are calculated weights(). The R code produces further both unweighted and weighted totals for turnover in the previous month, turnover in the current month and indices. Note that turnover also include VAT. For such estimated parameters, standard errors, coefficients

of variation and confidence intervals are also calculated. At the end, all this parameters, standard errors, coefficients of variation and confidence intervals are calculated for the Republic of Serbia by two groups and eight classes of NACE Rev. 2 which belong to division 47.

For description of functions for import excel files in R and export from R to excel files see [4].

For descriptions of functions for handling with complex sampling designs see [3].

For descriptions of functions for handling with complex sampling designs and for insight in the corresponding theory see [1].

For description of functions which handle with frames and other R objects see [5].

CONCLUSION

R survey package offers very effective ways for implementing different estimation procedures in MRTS. I have chosen R functions that give Horvitz-Thompson estimations for totals and indices and R function that give estimation of variance. In case of non-linear parameters (indices) function uses Taylor linearization method for estimation of variance.

Since variance is one of the key indicators of quality in the sample surveys and helps the user to draw better conclusions about the statistics produced, I want to emphasize that other methods for variance estimation are developed in survey package such as: Balanced Repeated Replication or BRR, Fay's method, Jackknife and Bootstrap method.

ANNEX

COMPLETE CODE FOR ESTIMATION PROCEDURE

```
>library(survey)
>library(XLConnect)

>plan_uzorak<-file.path("C:/Documents and Settings/sofija/Desktop/puz10_2013_
april.xls")
>plan_uzorak<- readWorksheetFromFile(plan_uzorak, sheet="puz10_2013")
>realiz_uzorak<-file.path("C:/Documents and Settings/sofija/Desktop/TRG10N_za_
pondere_oktobar.xls")
>realiz_uzorak<- readWorksheetFromFile(realiz_uzorak, sheet="Sheet1")

>uparena<-merge(plan_uzorak,realiz_uzorak,by="mb")
>malo<-as.vector(uparena$malo)
>for(i in 1:length(malo)){
      if(malo[i]==4711 || malo[i]==4719)  malo[i]<-malo[i]
       else malo[i]<-substr(malo[i],1,3)
}
>uparena$malo1<-malo

>pomoc<- file.path("C:/Documents and Settings/sofija/Desktop/pom.xlsx")

>writeWorksheetToFile(pomoc,uparena,sheet='upar')

>slog1<-subset(uparena,vs==1)
>slog2<-subset(uparena,vs==2)
>slog3<-subset(uparena,vs==3)
>slog4<-subset(uparena,vs==4)
>slog5<-subset(uparena,vs==5)
>slog6<-subset(uparena,vs==6)

>options(survey.lonely.psu=»remove»)

>dstrat1<-svydesign(id=~1,strata=~mstrat, data=slog1, fpc=~mnh)
>dstrat2<-svydesign(id=~1,strata=~mstrat, data=slog2, fpc=~mnh)
>dstrat3<-svydesign(id=~1,strata=~mstrat, data=slog3, fpc=~mnh)
>dstrat4<-svydesign(id=~1,strata=~mstrat, data=slog4, fpc=~mnh)
>dstrat5<-svydesign(id=~1,strata=~mstrat, data=slog5, fpc=~mnh)
>dstrat6<-svydesign(id=~1,strata=~mstrat, data=slog6, fpc=~mnh)
```

```
>ponderi<-as.vector(weights(dstrat1))
>tabela_pon<-data.frame(slog1$mb,ponderi,slog1$rbr.x,slog1$mbops,slog1$ppromet,slog2$ppromet,
slog3$ppromet,slog4$ppromet,slog5$ppromet,slog6$ppromet,slog1$ipromet,slog2$ipromet,slog3$ipromet,slog4$ipromet,slog5$ipromet,slog6$ipromet,slog1$indeks_i_na_p,
slog2$indeks_i_na_p,slog3$indeks_i_na_p,slog4$indeks_i_na_p,slog5$indeks_i_na_p,
slog6$indeks_i_na_p,slog1$ppdv,slog2$ppdv,slog3$ppdv,slog4$ppdv,slog5$ppdv,
slog6$ppdv,slog1$ipdv,slog2$ipdv,slog3$ipdv,slog4$ipdv,slog5$ipdv,slog6$ipdv,slog1$mstrat)
>writeWorksheetToFile(pomoc,tabela_pon,sheet='tab_pon')

>unw_ppromet<-c(sum(slog1$ppromet),sum(slog2$ppromet),sum(slog3$ppromet),sum(slog4$ppromet),
sum(slog5$ppromet),sum(slog6$ppromet))
>unw_ipromet<-c(sum(slog1$ipromet),sum(slog2$ipromet),sum(slog3$ipromet),sum(slog4$ipromet),
sum(slog5$ipromet),sum(slog6$ipromet))
>unw_indeks_i_na_p<-c(sum(slog1$ipromet)/sum(slog1$ppromet),sum(slog2$ipromet)/sum(slog2$ppromet),
sum(slog3$ipromet)/sum(slog3$ppromet),sum(slog4$ipromet)/
sum(slog4$ppromet),
sum(slog5$ipromet)/sum(slog5$ppromet),sum(slog6$ipromet)/
sum(slog6$ppromet))*100
>vs<-c(1,2,3,4,5,6)
>naziv_vs<-c("Ukupan promet","Promet trg. na malo","Srbija-sever","Beogradski region","Region Vojvodine","Srbija jug")
>unw_promet<-data.frame(vs,naziv_vs,unw_ppromet,unw_ipromet,unw_
indeks_i_na_p)
>writeWorksheetToFile(pomoc,unw_promet,sheet='unw_prom')

#ESTIMATES WITH VAT

>w_ppromet<-c(round(svytotal(~slog1$ppromet,dstrat1),2),round(svytotal(~slog2$ppromet,dstrat2),2),
round(svytotal(~slog3$ppromet,dstrat3),2),round(svytotal(~slog4$ppromet,dstrat4),2),
round(svytotal(~slog5$ppromet,dstrat5),2),round(svytotal(~slog6$ppromet,dstrat6),2))
```

```
>w_ipromet<-c(round(svytotal(~slog1$ipromet,dstrat1),2),round(svytotal(~slog2$ip
romet,dstrat2),2),
round(svytotal(~slog3$ipromet,dstrat3),2),round(svytotal(~slog4$ipromet,dstrat4),2),
round(svytotal(~slog5$ipromet,dstrat5),2),round(svytotal(~slog6$ipromet,dstrat6
),2))
>w_indeks_i_na_p<-round(c(as.numeric(as.vector(svyratio(~slog1$ipromet,~slog1$
ppromet,dstrat1)))[1],
as.numeric(as.vector(svyratio(~slog2$ipromet,~slog2$ppromet,dstrat2)))[1],
as.numeric(as.vector(svyratio(~slog3$ipromet,~slog3$ppromet,dstrat3)))[1],
as.numeric(as.vector(svyratio(~slog4$ipromet,~slog4$ppromet,dstrat4)))[1],
as.numeric(as.vector(svyratio(~slog5$ipromet,~slog5$ppromet,dstrat5)))[1],
as.numeric(as.vector(svyratio(~slog6$ipromet,~slog6$ppromet,dstrat6)))
[1])*100,2)

>se_w_ppromet<-c( as.data.frame(svytotal(~slog1$ppromet,dstrat1))[,2], as.data.fr
ame(svytotal(~slog2$ppromet,dstrat2))[,2],
as.data.frame(svytotal(~slog3$ppromet,dstrat3))[,2], as.data.frame(svytotal(~slog
4$ppromet,dstrat4))[,2],
 as.data.frame(svytotal(~slog5$ppromet,dstrat5))[,2], as.data.frame(svytotal(~slog
6$ppromet,dstrat6))[,2])
>se_w_ipromet<-c( as.data.frame(svytotal(~slog1$ipromet,dstrat1))[,2], as.data.fra
me(svytotal(~slog2$ipromet,dstrat2))[,2],
 as.data.frame(svytotal(~slog3$ipromet,dstrat3))[,2], as.data.frame(svytotal(~slog4
$ipromet,dstrat4))[,2],
 as.data.frame(svytotal(~slog5$ipromet,dstrat5))[,2], as.data.frame(svytotal(~slog6
$ipromet,dstrat6))[,2])
>se_w_indeks_i_na_p<-c(round(sqrt(as.numeric((as.vector(svyratio(~slog1$iprome
t,~slog1$ppromet,dstrat1))[2])))*100,2),
round(sqrt(as.numeric((as.vector(svyratio(~slog2$ipromet,~slog2$ppromet,dstrat
2))[2])))*
100,2),
round(sqrt(as.numeric((as.vector(svyratio(~slog3$ipromet,~slog3$ppromet,dstrat
3))[2])))*
100,2),
round(sqrt(as.numeric((as.vector(svyratio(~slog4$ipromet,~slog4$ppromet,dstrat
4))[2])))*
100,2),
round(sqrt(as.numeric((as.vector(svyratio(~slog5$ipromet,~slog5$ppromet,dstrat
5))[2])))*
100,2),
```

```
round(sqrt(as.numeric((as.vector(svyratio(~slog6$ipromet,~slog6$ppromet,dstrat
6))[2])))*
100,2))

>cv_w_ppromet<-round(c(as.numeric(cv(svytotal(~slog1$ppromet,dstrat1))),
as.numeric(cv(svytotal(~slog2$ppromet,dstrat2))),
as.numeric(cv(svytotal(~slog3$ppromet,dstrat3))),
as.numeric(cv(svytotal(~slog4$ppromet,dstrat4))),
as.numeric(cv(svytotal(~slog5$ppromet,dstrat5))),
as.numeric(cv(svytotal(~slog6$ppromet,dstrat6))))*100,2)
>cv_w_ipromet<-round(c(as.numeric(cv(svytotal(~slog1$ipromet,dstrat1))),
as.numeric(cv(svytotal(~slog2$ipromet,dstrat2))),
as.numeric(cv(svytotal(~slog3$ipromet,dstrat3))),
as.numeric(cv(svytotal(~slog4$ipromet,dstrat4))),
as.numeric(cv(svytotal(~slog5$ipromet,dstrat5))),
as.numeric(cv(svytotal(~slog6$ipromet,dstrat6))))*100,2)
>cv_w_indeks_i_na_p<
round(c(as.numeric(cv(svyratio(~slog1$ipromet,~slog1$ppromet,dstrat1))),
as.numeric(cv(svyratio(~slog2$ipromet,~slog2$ppromet,dstrat2))),
as.numeric(cv(svyratio(~slog3$ipromet,~slog3$ppromet,dstrat3))), as.numeric(cv(s
vyratio(~slog4$ipromet,~slog4$ppromet,dstrat4))),
as.numeric(cv(svyratio(~slog5$ipromet,~slog5$ppromet,dstrat5))),
as.numeric(cv(svyratio(~slog6$ipromet,~slog6$ppromet,dstrat6))))*100,2)
```

#CONFIDENCE INTERVALS

```
>ci1_ppromet<-c(confint(svytotal(~slog1$ppromet,dstrat1)))
>ci2_ppromet<-c(confint(svytotal(~slog2$ppromet,dstrat2)))
>ci3_ppromet<-c(confint(svytotal(~slog3$ppromet,dstrat3)))
>ci4_ppromet<-c(confint(svytotal(~slog4$ppromet,dstrat4)))
>ci5_ppromet<-c(confint(svytotal(~slog5$ppromet,dstrat5)))
>ci6_ppromet<-c(confint(svytotal(~slog6$ppromet,dstrat6)))

>dci_ppromet<-c(ci1_ppromet[1],ci2_ppromet[1],ci3_ppromet[1],ci4_
ppromet[1],ci5_ppromet[1],
ci6_ppromet[1])
>gci_ppromet<-c(ci1_ppromet[2],ci2_ppromet[2],ci3_ppromet[2],ci4_
ppromet[2],ci5_ppromet[2],
ci6_ppromet[2])
```

```
>ci1_ipromet<-c(confint(svytotal(~slog1$ipromet,dstrat1)))
>ci2_ipromet<-c(confint(svytotal(~slog2$ipromet,dstrat2)))
>ci3_ipromet<-c(confint(svytotal(~slog3$ipromet,dstrat3)))
>ci4_ipromet<-c(confint(svytotal(~slog4$ipromet,dstrat4)))
>ci5_ipromet<-c(confint(svytotal(~slog5$ipromet,dstrat5)))
>ci6_ipromet<-c(confint(svytotal(~slog6$ipromet,dstrat6)))

>dci_ipromet<-c(ci1_ipromet[1],ci2_ipromet[1],ci3_ipromet[1],ci4_ipromet[1],ci5_
ipromet[1],ci6_ipromet[1])
>gci_ipromet<-c(ci1_ipromet[2],ci2_ipromet[2],ci3_ipromet[2],ci4_ipromet[2],ci5_
ipromet[2],
ci6_ipromet[2])

>ci1_indeks_i_na_p<-c(confint(svyratio(~slog1$ipromet,~slog1$ppromet,dstrat1)))
>ci2_indeks_i_na_p<-c(confint(svyratio(~slog2$ipromet,~slog2$ppromet,dstrat2)))
>ci3_indeks_i_na_p<-c(confint(svyratio(~slog3$ipromet,~slog3$ppromet,dstrat3)))
>ci4_indeks_i_na_p<-c(confint(svyratio(~slog4$ipromet,~slog4$ppromet,dstrat4)))
>ci5_indeks_i_na_p<-c(confint(svyratio(~slog5$ipromet,~slog5$ppromet,dstrat5)))
>ci6_indeks_i_na_p<-c(confint(svyratio(~slog6$ipromet,~slog6$ppromet,dstrat6)))

>dci_indeks_i_na_p<-c(ci1_indeks_i_na_p[1],ci2_indeks_i_na_p[1],ci3_indeks_i_
na_p[1],ci4_indeks_i_na_p[1],
ci5_indeks_i_na_p[1],ci6_indeks_i_na_p[1])*100
>gci_indeks_i_na_p<-c(ci1_indeks_i_na_p[2],ci2_indeks_i_na_p[2],ci3_indeks_i_
na_p[2],ci4_indeks_i_na_p[2],
ci5_indeks_i_na_p[2],ci6_indeks_i_na_p[2])*100
```

DOMAIN ESTIMATES

```
>d_ppromet<-as.vector(svyby(~ppromet,~malo1, dstrat2, svytotal, keep.
var=TRUE))[,2]
>d_ipromet<-as.vector(svyby(~ipromet,~malo1, dstrat2, svytotal, keep.var=TRUE))
[,2]
>d_indeks_i_na_p<-round(as.vector(svyby(~ipromet, by=~malo1,
denominator=~ppromet, design=dstrat2, svyratio))[,2]*100,2)

>d_se_ppromet<-as.vector(svyby(~ppromet,~malo1, dstrat2, svytotal, keep.
var=TRUE))[,3]
>d_se_ipromet<-as.vector(svyby(~ipromet,~malo1, dstrat2, svytotal, keep.
var=TRUE))[,3]
```

```
>d_se_indeks_i_na_p<-round(as.vector(svyby(~ipromet, by=~malo1,
denominator=~ppromet, design=dstrat2, svyratio))[,3]*100,2)

>d_cv_ppromet<-round(as.vector(cv(svyby(~ppromet,~malo1, dstrat2, svytotal,
keep.var=TRUE)))*100,2)
>d_cv_ipromet<-round(as.vector(cv(svyby(~ipromet,~malo1, dstrat2, svytotal,
keep.var=TRUE)))*100,2)
>d_cv_indeks_i_na_p<-round(as.vector(cv(svyby(~ipromet, by=~malo1,
denominator=~ppromet, design=dstrat2, svyratio)))*100,2)
```

#CONFIDENCE INTERVALS

```
>d_ci_ppromet<-c(confint(svyby(~ppromet, ~malo1, dstrat2, svytotal, keep.
var=TRUE)))
>d_ci_ipromet<-c(confint(svyby(~ipromet, ~malo1, dstrat2, svytotal, keep.
var=TRUE)))
>d_ci_indeks_i_na_p<-c(confint(svyby(~ipromet, by=~malo1,
denominator=~ppromet, design=dstrat2, svyratio)))

>dd_ci_ppromet<-d_ci_ppromet[1:(length(d_ci_ppromet)/2)]
>gd_ci_ppromet<-d_ci_ppromet[((length(d_ci_ppromet)/2)+1):length(d_ci_
ppromet)]

>dd_ci_ipromet<-d_ci_ipromet[1:(length(d_ci_ipromet)/2)]
>gd_ci_ipromet<-d_ci_ipromet[((length(d_ci_ipromet)/2)+1):length(d_ci_ipromet)]

>dd_ci_indeks_i_na_p<-d_ci_indeks_i_na_p[1:(length(d_ci_indeks_i_na_p)/2)]*100
>gd_ci_indeks_i_na_p<-d_ci_indeks_i_na_p[((length(d_ci_indeks_i_
na_p)/2)+1):length(d_ci_indeks_i_na_p)]*100

>d_ci_pprometb<-c(confint(svyby(~pprometb, ~malo1, dstrat2, svytotal, keep.
var=TRUE)))
>d_ci_iprometb<-c(confint(svyby(~iprometb, ~malo1, dstrat2, svytotal, keep.
var=TRUE)))
>d_ci_indeks_i_na_pb<-c(confint(svyby(~iprometb, by=~malo1,
denominator=~pprometb, design=dstrat2, svyratio)))

>dd_ci_pprometb<-d_ci_pprometb[1:(length(d_ci_pprometb)/2)]
>gd_ci_pprometb<-d_ci_pprometb[((length(d_ci_pprometb)/2)+1):length(d_ci_
pprometb)]
```

```
>dd_ci_iprometb<-d_ci_iprometb[1:(length(d_ci_iprometb)/2)]
>gd_ci_iprometb<-d_ci_iprometb[((length(d_ci_iprometb)/2)+1):length(d_ci_
iprometb)]

>dd_ci_indeks_i_na_pb<-d_ci_indeks_i_na_pb[1:(length(d_ci_indeks_i_na_
pb)/2)]*100
>gd_ci_indeks_i_na_pb<-d_ci_indeks_i_na_pb[((length(d_ci_indeks_i_na_
pb)/2)+1):length(d_ci_indeks_i_na_pb)]*100

>trg10_intpov_eu<-data.frame(d_indeks_i_na_p,dd_ci_indeks_i_na_p,gd_ci_
indeks_i_na_p,
d_ppromet,dd_ci_ppromet,gd_ci_ppromet,d_ipromet,dd_ci_ipromet,gd_ci_
ipromet,
d_indeks_i_na_pb,dd_ci_indeks_i_na_pb,gd_ci_indeks_i_na_pb,d_pprometb,
dd_ci_pprometb,gd_ci_pprometb,d_iprometb,dd_ci_iprometb,gd_ci_iprometb)
>writeWorksheetToFile(pomoc,trg10_intpov_eu,sheet='trg10_intpov_eu')
```

References

1. Thomas Lumley. *Complex Surveys – A Guide to Analysis Using R*. John Wiley & Sons, New York, 2010.
2. Olga Melovski Trpinac. *Monthly Retail Trade Survey – Working paper.* SORS, 2013.
3. Thomas Lumley. *Package – 'survey'*, available at http://cran.r-project.org/web/packages/survey/survey.pdf, 2013.
4. Thomas Lumley. *Package – 'XLConnect'*, available at http://cran.r-project.org/web/packages/XLConnect/XLConnect.pdf, 2014.
5. Michael J. Crawley . *The R Book.* John Wiley & Sons Ltd, England, 2007.