Selected Statistical Methods in R
Vybrané statistické metody v R

Jiří Neubauer

Abstract: Statistical data analysis without computers is hardly imaginable nowadays. Trying to find a suitable statistical software, particular aspects, such as universality, being user friendly and not least price, play a significant role. It is possible to find a range of commercial products in the market (for example Statistica, Statgraphics, S-plus etc.) but their prices are too excessive for common users or students. The program R provides a potential alternative to these products. R was created on the basis of stripped down version of S language. The code for R was released in 1995 under a GPL (General Public License) which means it can be freely downloaded. The contribution is focused on using of R trying to show an application of selected statistical methods on particular examples with effort to demonstrate advantages and flexibility of R.

Abstrakt: Statistická analýza dat bez využití počítače je v současné době již nemyslitelná. Při hledání vhodného software hrají důležitou roli některé aspekty, jako jsou univerzálnost, uživatelská přístupnost a v neposlední řadě cena. Na trhu lze najít řadu komerčních produktů, např. Statistica, Statgraphics, S-plus a jiné, jejich cena je však pro normální uživatele či studenty nepřiměřeně vysoká. Možnou alternativou k uvedeným produktům může být program R, který vznikl na základě zjednodušené verze jazyka S. Program R se objevil v roce 1995 jako GPL licence (General Public License), což znamená, že je volně ke stažení. Článek ukazuje využití tohoto programu na některých konkrétních příkladech ze statistické analýzy dat se snahou ukázat jeho výhody a možnosti.

1. Introduction

Statistical data analysis without computers is hardly imaginable nowadays. Trying to find a suitable statistical software, particular aspects, such as universality, being user friendly and last, but not least price, play a significant role. It is possible to find a range of commercial products in the market (for example Statistica, Statgraphics, S-plus etc.) but their prices are too excessive for common users or students. The programming environment R offers a potential alternative to these products. R was created on the basis of a stripped down version of S language. The code for R was released in 1995 under a GPL (General Public License) which means it can be freely downloaded (http://www.r-project.org/). The contribution is focused on the use of R trying to show an application of selected statistical methods on particular examples with effort to demonstrate advantages and flexibility of R.

2. Descriptive Statistics

We will use several examples to describe how to use R. Necessary commands will be mentioned to complete statistical calculation. The commands will be written directly to "R console" window (a basic interface). In this paragraph we will focus on basic descriptive statistics.

Example 1: A large company buys thousands of lightbulbs every year. The company is currently considering four brands of lightbulbs to choose from. Before the company decides which lightbulbs to buy, it wants to investigate if the mean lifetimes of four types of lightbulbs are the same. The company's research department randomly selected a few bulbs of each type

and tested them. The following table lists the number of hours (in thousands) that each of the bulbs in each brand lasted before burning out.

Brand A	23	24	19	26	22	23	25
Brand B	19	23	18	24	20	22	19
Brand C	23	27	25	26	23	21	27
Brand D	26	24	21	29	28	24	28

First of all, we have to input data into program. The most common way to do this is to save data into text file (for example bulbs.txt, where the first column contains the type of bulbs – A, B, C, D – and in the second column are lifetimes) with the names of variables on the first line of the file. The commands `attach` and `names` allow us to call variables directly by names in the original text file. Using the command `plot` we create four box-plots of bulb lifetimes according to the four brands (categorical variables, factors), see figure 1. If all two variables were continuous, the command `plot` would produce a scatter-plot.

```
> data<-read.table("bulbs.txt",header=T)
> attach(data)
> names(data)
[1] "brand"    "lifetime"
```

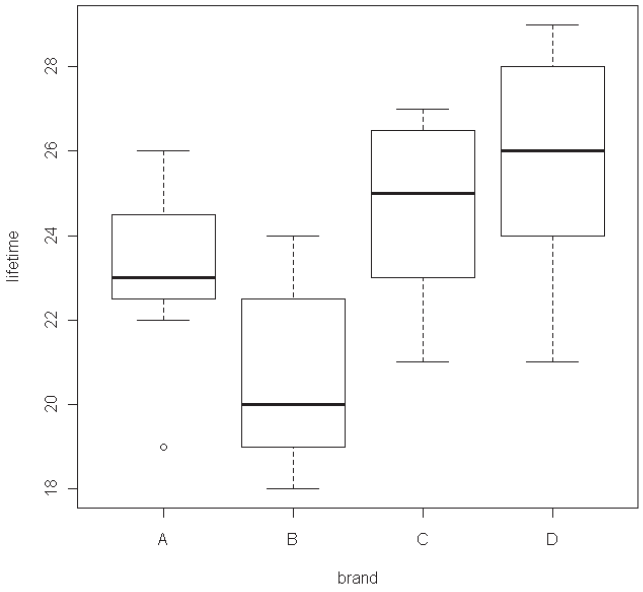

Fig. 1: Boxplots of lifetimes

```
> plot(data)
> table(data)
     lifetime
brand 18 19 20 21 22 23 24 25 26 27 28 29
    A  0  1  0  0  1  2  1  1  1  0  0  0
    B  1  2  1  0  1  1  1  0  0  0  0  0
    C  0  0  0  1  0  2  0  1  1  2  0  0
    D  0  0  0  1  0  0  2  0  1  0  2  1
```

The command `table` creates a frequency table of lifetimes and brands of bulbs. We list now some commands for elementary descriptive statistics of vector x: `mean(x)` – the arithmetic mean, `length(x)` – the number of elements in x, `var(x)` – the sample variation, `sd(x)` – the sample standard deviation, `quantile(x)` – quantiles, `IRQ(x)` – the interquartile deviation. The basic information about datasets is possible to get by the command `summary` (each column is taken as a variable). Nevertheless, this is not convenient for variable "lifetime" because it is a mixture of four brands of lightbulbs. To solve this problem we can use the command `tapply` which provides an application of some function (in our case the function `summary`) on the variable "lifetime" according to the factor "brand". The mean and the variance of the bulb lifetime according to the different brand can be computed in the same way.

```
> tapply(lifetime,brand,mean)
       A        B        C        D
23.14286 20.71429 24.57143 25.71429
> tapply(lifetime,brand,var)
       A        B        C        D
5.142857 5.238095 5.285714 8.238095
```

```
> summary(data)
 brand    lifetime
 A:7   Min.   :18.00
 B:7   1st Qu.:21.75
 C:7   Median :23.50
 D:7   Mean   :23.54
       3rd Qu.:26.00
       Max.   :29.00
```

```
> tapply(lifetime,brand,summary)
$A
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  19.00   22.50   23.00   23.14   24.50   26.00

$B
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  18.00   19.00   20.00   20.71   22.50   24.00

$C
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  21.00   23.00   25.00   24.57   26.50   27.00

$D
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  21.00   24.00   26.00   25.71   28.00   29.00
```

3. Hypothesis Testing

R contains a lot of different statistical tests. If they are not directly available (is not in our library), the package can be downloaded. If a package is not obtainable, run the R program, then from the command line use `install.packages("name of package")`. In this section we will mention some elementary statistical tests, such as one and two-sample t-test, one and two-sample tests of population proportions, a two-sample variation test and some normality tests. We set a significance level of all test to 0.05.

3.1 One-sample Tests

Provided we need to calculate a test of the mean of normal distribution, we usually use t-test. In program R we apply a function `t.test`. The following example shows the test of a null

hypothesis that the mean of the bulb lifetime of type "A" is equal to 20 thousands hours against an alternative hypothesis that the mean is greater than 20.

```
> t.test(lifetime[brand=="A"],mu=20,alternative="greater")

        One Sample t-test

data:  lifetime[brand == "A"]
t = 3.6667, df = 6, p-value = 0.005248
alternative hypothesis: true mean is greater than 20
95 percent confidence interval:
 21.47727      Inf
sample estimates:
mean of x
 23.14286
```

The function `t.test` returns not only the value of the test statistics and p-value but also a confidence interval corresponding to the chosen alternative hypothesis (alternative: "two.sided", "less", "greater") and a sample estimate of the mean. On the grounds of computed results we reject the null hypothesis and can say that the mean of lifetime for the brand A is greater than 20 thousand hours. We use the function `wilcox.test` if we intend to calculate a non-parametric test of the mean.

Sometimes we want to conduct a test of hypothesis about a population proportion. The following example describes how it can be calculated in R. Let us note that the sample size should be large.

Example 2: We tossed a coin 100 times and we got 52 heads. Can we say that this coin is fair?

For the purpose of testing of the population proportion it is possible to apply the function `pro.test` with or without continuity correction (correct=F).

```
> prop.test(52,100,p=0.5,correct=F)

        1-sample proportions test without continuity correction

data:  52 out of 100, null probability 0.5
X-squared = 0.16, df = 1, p-value = 0.6892
alternative hypothesis: true p is not equal to 0.5
95 percent confidence interval:
 0.4231658 0.6153545
sample estimates:
   p
0.52
```

The test does not reject the null hypothesis that the population proportion is 0.5, therefore we can consider the coin to be fair.

3.2 Two-sample Tests

In this paragraph we introduce function in R which allows to conduct two-sample tests. These test are useful in case we need to compare two population means, variances or proportions. Let us assume we have two independent samples from two normal distributions. The first test we are going to deal with is used to compare two variances. We would like to compare variances of brand "A" and brand "B" lifetime of bulbs (see the example 1).

```
> var.test(lifetime[brand=="A"],lifetime[brand=="B"])

        F test to compare two variances

data:  lifetime[brand == "A"] and lifetime[brand == "B"]
F = 0.9818, num df = 6, denom df = 6, p-value = 0.9828
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.1687043 5.7139428
sample estimates:
ratio of variances
         0.9818182
```

The result of the test is that we cannot reject the null hypothesis that the variances are equal (the ratio of the population variances is equal to 1).

More often than the comparison of two variances we need to compare two population means. In program R we use the function `t.test`, the same function which was mentioned when we calculated the one-sample test of the mean, but the syntax is a little different. We include the result of the previous test into the function options (var.equal=TRUE). We will test a null hypothesis that the population means of brand "A" and brand "B" lifetimes are equal against an alternative hypothesis that the brand "A" lifetime is greater than the brand "B".

```
> t.test(lifetime[brand=="A"],lifetime[brand=="B"],var.equal=TRUE,
alternative="greater")

        Two Sample t-test

data:  lifetime[brand == "A"] and lifetime[brand == "B"]
t = 1.9943, df = 12, p-value = 0.03467
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.2581336       Inf
sample estimates:
mean of x mean of y
 23.14286  20.71429
```

According to the results of the $t-$test we can say that the mean lifetime of the brand "A" bulbs is greater.

In case of two dependent (paired) samples from a normal distribution we also use the function `t.test` but we have to add one parameter – `t.test(x,y,paired=TRUE)`. A non-parametric version of the two-sample test of the mean is provided by the function `wilcox.test`.

The following test enables us to compare two population proportions. We should point out that the sample sizes must be sufficiently large.

Example 3: The management of a supermarket wanted to find out if the percentage of men and women who prefer to buy national brand products over store brand products is different. A sample of 600 male shoppers at the company's supermarkets showed that 246 of them prefer to buy national brand products over store brand products. Another sample of 700 female shoppers at the company's supermarkets showed 266 of them prefer to buy national brand products over store brand products. Can we conclude that the proportion of all male and all female shoppers at these supermarkets who prefer to buy national brand products over store brand products are different?

We apply the function `prop.test` (see example 2) to calculate appropriate tests.

```
> prop.test(c(246,266),c(600,700))

        2-sample test for equality of proportions with continuity
        correction

data:  c(246, 266) out of c(600, 700)
X-squared = 1.0956, df = 1, p-value = 0.2952
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.02485497  0.08485497
sample estimates:
prop 1 prop 2
  0.41   0.38
```

The result of this test is that there is no significant difference between the proportion of all male and female women shoppers at these supermarkets who prefer to buy national brand products over store brand products.

3.3 Normality Tests

The simplest test of normality is the 'quantile-quantile' plot. It plots the ranked samples from our distribution against a similar number of ranked quantiles taken from a normal distribution. If the sample is normally distributed, all plotted points will be in a straight line. The functions we need are `qqplot` and `qqline`.

```
> par(mfrow=c(2,2))
> qqnorm(lifetime[brand=="A"],main="Lifetime - Brand A")
> qqline(lifetime[brand=="A"],lty=2)
> qqnorm(lifetime[brand=="B"],main="Lifetime - Brand B")
> qqline(lifetime[brand=="B"],lty=2)
> qqnorm(lifetime[brand=="C"],main="Lifetime - Brand C")
> qqline(lifetime[brand=="C"],lty=2)
> qqnorm(lifetime[brand=="D"],main="Lifetime - Brand D")
> qqline(lifetime[brand=="D"],lty=2)
```

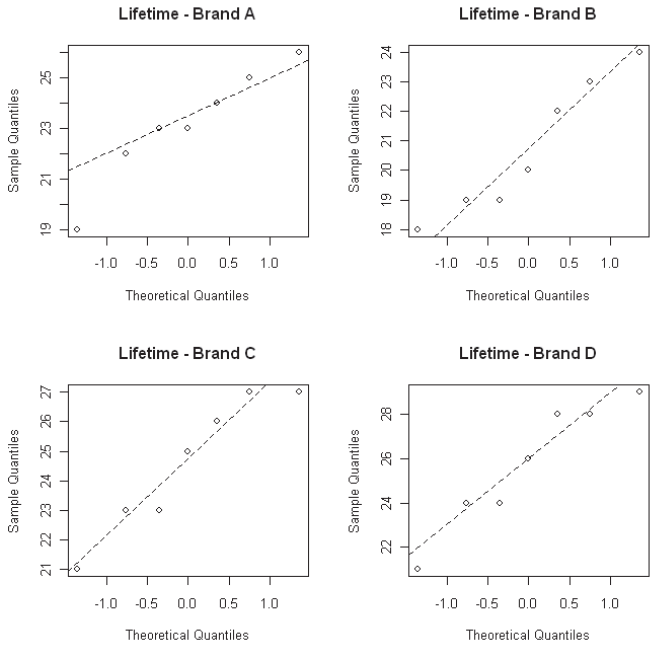

Fig. 2: Q-Q plots of lifetime

It is possible to find several normality tests, in which case we have decided to mention two of them: Lilliefors and Shapiro-Wilk normality test (functions `lillie.test` and `shapiro.test`).

```
> install.packages("nortest")
> library("nortest")
> lillie.test(lifetime[brand=="A"])

        Lilliefors (Kolmogorov-Smirnov)
                normality test

data:  lifetime[brand == "A"]
D = 0.1892, p-value = 0.6298
```

```
> shapiro.test(lifetime[brand=="A"])

        Shapiro-Wilk normality test

data:  lifetime[brand == "A"]
W = 0.9492, p-value = 0.7222
```

According to these tests we consider our datasets as samples form normal distributions.

4. Analysis of Variance

Analysis of variance (ANOVA) is a very useful statistical tool which provides comparison of two or more population means. Explanatory variables (called) factors are usually categorical. Using the function `aov` we can calculate not only one-way ANOVA (one factor), but also two or three-way analysis of variance. We will focus on the data from example 1 – lifetime of lightbulbs. The brand of bulbs plays a role of one factor.

```
> summary(aov(lifetime~brand))
            Df  Sum Sq Mean Sq F value   Pr(>F)
brand        3  97.536  32.512  5.4402 0.005344 **
Residuals   24 143.429   5.976
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

The p-value of F-test is smaller than 0.05; therefore, we reject the null hypothesis that all four population means are equal, the lifetime depends on the type of bulbs.

One assumption of ANOVA is that conditional variances (in our case four variances of bulb lifetime of brand "A", brand "B", brand "C" and brand "D") are equal. We can test it by using the Barlett's test of homogeneity of variances. (Another option is to use the Fligner-Killeen test `fligner.test(lifetime~brand)`).

```
> bartlett.test(lifetime~brand)

        Bartlett test of homogeneity of variances

data:  lifetime by brand
Bartlett's K-squared = 0.4701, df = 3, p-value = 0.9254
```

The assumption of equal variances is acceptable. If we want to compute two-way ANOVA with interactions, we can apply the command `oav(y~factor1*factor2)`, without interactions the command `oav(y~factor1+factor2)`.

5. Regression

Regression analysis is the statistical method we use when both the response variable and explanatory variable are quantitative (usually continuous).

Example: An insurance company wants to know how the amounts of life insurance depend on the incomes of persons. The research department at the company collected information on six persons. The table lists the annual incomes (in thousands of dollars) and amounts (in thousands of dollars) of life insurance policies for these six persons.

Annual income	62	78	41	53	85	34
Life insurance	250	300	100	150	500	75

We obtain least square estimates of linear regression function by using the function `lm`. The following script contains basic regression analysis including estimates of regression parameters, t-tests of these parameters, F-test of the model, correlation coefficient, residuals and confidence intervals of estimated parameters.

```
> income<-c(62,78,41,53,85,34)
> insurance<-c(250,300,100,150,500,75)
> plot(income,insurance)
> regmodel<-lm(insurance~income)
> summary(regmodel)
Call:
lm(formula = insurance ~ income)
Residuals:
      1       2       3       4       5       6
 -2.719 -71.717   3.467 -35.782  76.221  30.529
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -208.402     78.533  -2.654  0.05676 .
income         7.437      1.274   5.838  0.00429 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 57.41 on 4 degrees of freedom
Multiple R-squared: 0.895,      Adjusted R-squared: 0.8687
F-statistic: 34.08 on 1 and 4 DF,  p-value: 0.004292
> confint(regmodel1)
                  2.5 %    97.5 %
(Intercept) -426.443782  9.640335
income         3.900236 10.974610
```

The commands `plot(regmodel)` creates four plots describing quality of the model (a plot of residuals against fitted values, a normal Q-Q plot, a scale-location plot of $\sqrt{|\text{residuals}|}$) against fitted values, a plot of residuals against leverages including Cook's distances). All kinds of linear regressions can be computed in the same way.

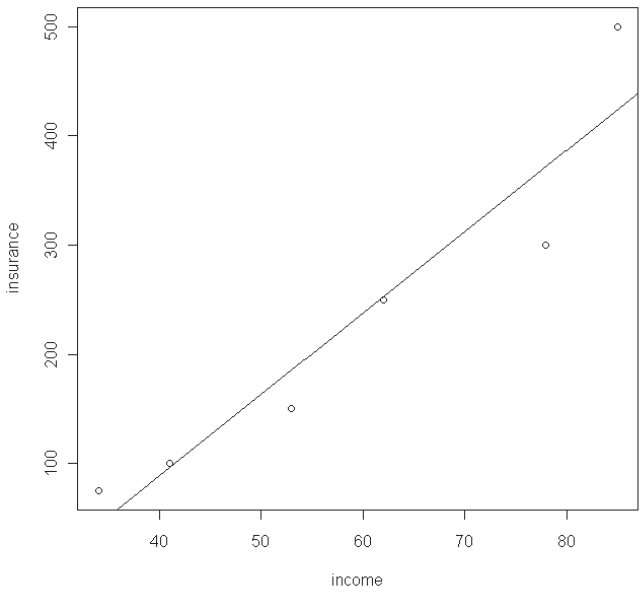

Fig. 3: Linear regression

6. Time Series Analysis

Time series data are vectors of numbers, typically regularly spaced in time. We will present elementary analysis of one-dimensional time series of real wages indices of the Czech Republic (quarterly data from 2001 to 2010). First of all, we have to create a time series object by the function `ts`.

```
> data.ts<-read.table("wages.txt",header=T);attach(data.ts);names(data.ts)
[1] "wages"
> wages<-ts(wages,frequency = 4, start = c(2001, 1))
```

If we want to get a graph of this time series, we simply write `plot(wages)`. Let us try to find an ARMA or an ARIMA model for the time series. It is reasonable to calculate an autocorrelation and a partial autocorrelation function (see figure 4).

```
> par(mfrow=c(2,1))
> acf(wages,main="");acf(wages,type="p",main="")
```

On the basis of the shape of these functions we choose the ARMA(1,0) model (see (3) for a detailed description). The commands `arima` and `predict` calculate estimates and predictions of ARIMA models (ARMA(1,0)=ARIMA(1,0,0)). If we want to create a graph of the time series and predictions, we can apply the function `plot.Arima`. It requires a TSA package.

```
model1<-arima(wages,order=c(1,0,0))
predict(model1, n.ahead = 4)
plot.Arima(model1,n.ahead=4)  #TSA package
```

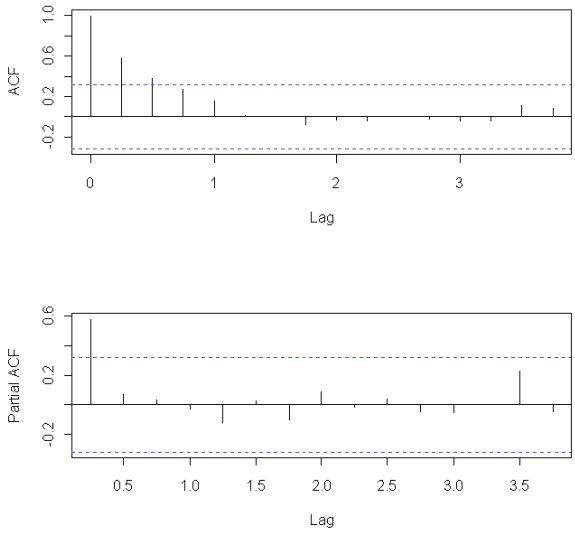

Fig. 4: Autocorrelation and partial autocorrelation function of indices of real wages

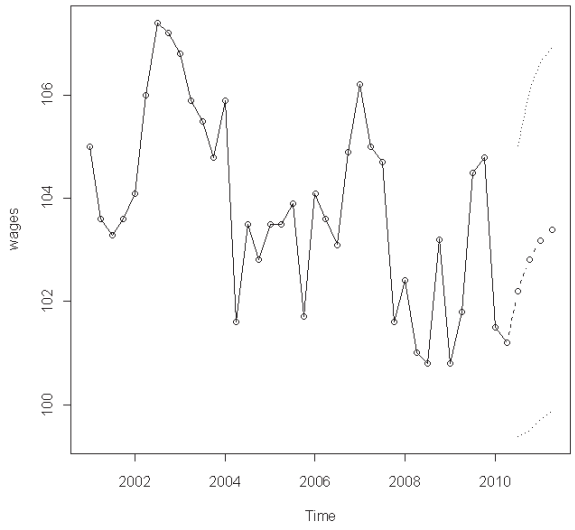

Fig. 5: Prediction of AR(1) model

7. Conclusion

The programming environment R offers a wide scale of statistical methods, some of the basic ones were briefly mentioned in this article. The main advantages of this tool are the growing community of users (a lot of useful packages can be freely downloaded) and of course the price. On the other hand, this software is not quite user friendly. It takes some time to learn how to

operate it. I would like to note that the package Rcmdr (R-commander, a basic-statistics graphical user interface) can help beginners with an elementary statistical data analysis. Program R belongs to the family of software, such as Gretl or JMulti, which are freely available. This aspect brings the possibility to analyze data and help understand statistical methods not only for students.

Acknowledgement: The paper was supported by the grant GAČR P402/10/P209.

References

1. CRAWLEY, M., J. *The R Book*. Wiley, 2007. ISBN 978-0-470-51024-7.
2. CRAWLEY, M., J. *Statistics : An Introduction using R*. Wiley, 2005. ISBN 978-0-470-02298-6.
3. CRYER, J., D., KUNG-SIK, CH. *Time Series Analysis : With Applications in R*. Wiley, 2005. ISBN 978-0-470-02298-6.
4. MANN, P., S. *Introductory Statistics*. Wiley, 2007. ISBN 978-0-471-75530-2.

Mgr. Jiří Neubauer, Ph.D.
Department of Econometrics, University of Defence
Kounicova 65, Brno, 612 00, Czech Republic
E-mail: Jiri.Neubauer@unob.cz
Telefon: + 420 973 442 029