The Process of Statistical Tests in R

This document shows how to perform statistical testing in R. We will show how to find p-values and make conclusions for proportions and means of distributions using a direct approach with appropriate formula and using build-in functions.

Hypothesis Testing for Proportions

We start this section with the Mayor support example. Remember that a survey of 1042 people was conducted and the sample proportion of support for Rob Ford mayor was 0.42. The goal is test whether the true proportion of the population is 0.5 (null hypothesis) versus that it is less than 0.5 (alternative). This test is clearly one directional. To use the formulas in a general way we first introduce variables like sample size, sample proportion, null hypothesis and then use these variables in an appropriate formula. Based on the survey we get:

```
N=1046
p.hat=0.42
p0=0.5
```

Here 'p.hat' is the sample proportion and 'p0' is the null hypothesis. Next we find z-statistic:

```
z.stat=(p.hat-p0)/sqrt( p0*(1-p0)/N )
z.stat
```

```
[1] -5.174708
```

In this case the z-statistic is -5.17 which is small and we expect that the p-value would also be very small. Let's find the exact p-value:

```
p.val=pnorm(z.stat)
p.val
```

```
[1] 1.14134e-07
```

Note that 'pnorm' function calculates probability below z-statistic which is what we want since the alternative hypothesis is that true proportion is less than 0.5. The p-value is clearly very small and therefore we reject the null hypothesis and conclude that the true proportion is smaller than 0.5.
Instead of using the above 'direct' method we can use the 'prop.test' function to get the p-value:

```
prop.test(x=p.hat*N,n=N,p=0.5,alternative="less",correct="FALSE")$p.value
```

```
[1] 1.14134e-07
```

The first argument is the number of successes (in this case number of people that supported Rob Ford), second argument is sample size (total number of people surveyed), the third one is the null hypothesis value. We also need to specify the 'alternative' which is less in this case and lastly we do not need here any corrections therefore we enter 'correct="FALSE"'. Since the prop.test function produces much information we extract the p-value from it using $ followed by 'p.value'. Note that the p-value is exactly the same as before. Hence both methods produce the same results, the second one is however is more convenient.
Now suppose that we want to test whether the true proportion is 0.44 versus that it is less than 0.42. We use exactly the same approach but with 'p0' equals to 0.44:

```
N=1046
p.hat=0.42
p0=0.44
z.stat=(p.hat-p0)/sqrt( p0*(1-p0)/N )
p.val=pnorm(z.stat)
p.val
```

```
[1] 0.09627147
```

The p-value in with this hypothesis is larger than 0.05 and therefore we do not have enough evidence to reject the null hypothesis.

Next let us move to the 'Flipping the bottle cap' example. Here a bottle cap was flipped 1000 times and proportion that we get red was 0.576. We want test whether the true proportion is 0.5 or it is not 0.5 (note that the test here is non-directional). First we get the z-statistic:

```
N=1000
p.hat=0.576
p0=0.5
z.stat=(p.hat-p0)/sqrt( p0*(1-p0)/N )
z.stat
```

```
[1] 4.806662
```

To get the p-value for non-directional alternative we must find probability above this z-statistic and multiply it by 2. The most convenient way to find it, is using the next command:

```
p.val=2* pnorm(-abs(z.stat))
p.val
```

```
[1] 1.534711e-06
```

Here the 'abs' function is the absolute value. Observe that the p-value is very small and hence we reject the null hypothesis that the true proportion is 0.5. Equivalently we can use the 'prop.test' function with 'alternative="two.sided"':

```
rop.test(p.hat*N,N,p=0.5,alternative="two.sided",correct="FALSE")$p.value
```

```
[1] 1.534711e-06
```

As before two p-values are completely the same.

Hypothesis Testing for Means

We start this section with the 'Age change' data set. Since these data consist of only one variable we use 'scan' function instead of 'read.table':

```
age.change=scan('agechange.txt')
head(age.change)
```

```
[1]  2.8  9.1  6.4  4.7  8.9 11.5
```

These data record by how many years a subject looks younger after a plastic surgery. We want to test whether the true mean is 0 (plastic surgery does not make any difference) versus that it is greater than 0 (people look younger after surgery). First we find sample mean, sample variance and number of observations (using 'length' function) and then compute t-statistic with the standard formula:

```
x.bar=mean(age.change)
sam.var=var(age.change)
N=length(age.change)
mu0=0
t.stat=(x.bar-mu0)/sqrt( sam.var/N )
t.stat
```

```
[1] 18.85627
```

The t-statistic here is larger than 18 which is very large hence we expect the p-value to be very tiny. Also since t-statistic follows a student-t distribution (under null) we must use this distribution with $N-1$ degrees of freedom to find p-value:

```
p.val=pt(t.stat,df=N-1.lower.tail=FALSE)
p.val
```

```
[1] 5.94943e-27
```

Since we need probability above the t-statistic we indicate 'lower.tail=FALSE'. The p-value is very small and hence we reject the null hypothesis and conclude that a plastic surgery does make people look younger on average. We can equivalently use the 't.test' function (here mu specifies the null hypothesis value):

```
t.test(age.change,mu=0,alternative="greater")
```

```
        One Sample t-test

data:  age.change
t = 18.8563, df = 59, p-value < 2.2e-16
alternative hypothesis: true mean is greater than 0
95 percent confidence interval:
 6.540651      Inf
sample estimates:
mean of x
 7.176667
```

The most important here is the second row: 't' represents the t-statistic, 'df' degrees of freedom and lastly the p-value. Here the output indicates that the p-value is less than $2.2 * 10^{-16}$ which is a very small number and therefore we should reject.
Next let's move to the 'Skeleton' data set:

```
Skeleton.data=read.table("SkeletonDataComplete.txt",header=TRUE)
head(Skeleton.data)
attach(Skeleton.data)
```

```
  Sex      BMIcat BMIquant Age DGestimate DGerror SBestimate SBerror
1   2 underweight    15.66  78         44     -34         60     -18
2   1      normal    23.03  44         32     -12         35      -9
3   1  overweight    27.92  72         32     -40         61     -11
4   1  overweight    27.83  59         44     -15         61       2
5   1      normal    21.41  60         32     -28         46     -14
6   1 underweight    13.65  34         25      -9         35       1
```

In this example the goal is to test whether the true mean of difference between actual and estimated age is 0 or not. We concentrate on 'DGerror' variable. As before we calculate the sample mean, sample variance then find t-statistic and finally get the p-value:

```
x.bar=mean(DGerror)
sam.var=var(DGerror)
N=length(DGerror)
mu0=0
t.stat=(x.bar-mu0)/sqrt( sam.var/N )
p.val=2*pt(-abs(t.stat),df=N-1)
p.val
```

```
[1] 2.735045e-62
```

Since this is a two sided test we multiply the p-value by 2. The p-value is extremely small and hence we reject null hypothesis and conclude that the true mean of 'DGerror' is not 0. Now we find the p-value with 't.test' function (do not forget to specify that the alternative is "two.sided"):

```
t.test(DGerror,mu=0,alternative="two.sided")$p.value
```

```
[1] 2.735045e-62
```

Finally let's analyze the 'Temperature' data set which consists of body temperatures of 130 subjects. First we download into R:

```
Temp=scan('TempData.txt')
```

We want test whether the true average body temperature is 37 or not. Using the same procedure as above we get the t-statistic and the p-value:

```
x.bar=mean(Temp)
sam.var=var(Temp)
N=length(Temp)
mu0=37
t.stat=(x.bar-mu0)/sqrt( sam.var/N )
p.val=2*pt(-abs(t.stat),df=N-1)
p.val
```

```
[1] 2.410632e-07
```

Note that the p-value is very small and therefore we reject the null hypothesis and conclude that the actual average body temperature is not 37 degrees Celsius. Equivalently we can use the 't.test' function:

```
t.test(Temp,mu=37,alternative="two.sided")
```

```
        One Sample t-test

data:  Temp
t = -5.4548, df = 129, p-value = 2.411e-07
alternative hypothesis: true mean is not equal to 37
95 percent confidence interval:
 36.73445 36.87581
sample estimates:
mean of x
 36.80513
```

Once again we obtain exactly the same p-value and hence the same conclusion.

Summary of R Functions

We give a short summary of all new and/or important R functions [and arguments] that we used in this Module:

Distributions

`pnorm()`
`pt() [df]`

Testing

`prop.test() [x,n,p,alternative,conf.level,correct]`
`t.test() [x,mu,alternative,conf.level]`

Miscellaneous

`sqrt()`
`abs()`