Compare two groups in R

In this document we show how to compare 2 groups. We start with analysis of matched pairs and then show how make confidence intervals and tests for independent samples. We will show procedures for proportions and means of quantitative variables.

Matched Pairs

This section shows how to find confidence intervals and perform statistical testing for the difference between two dependent groups. Consider first the 'Skeleton' data set:

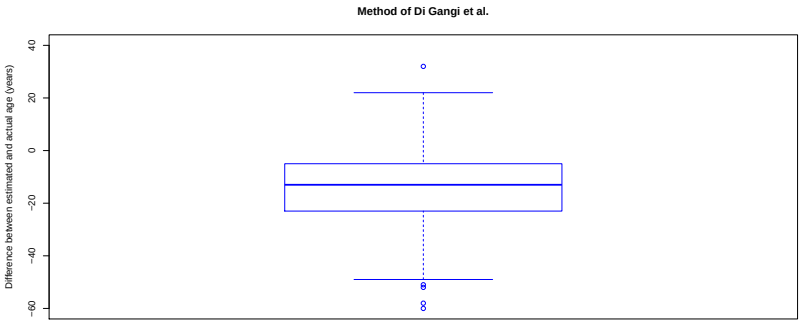
```
Skeleton.data=read.table("SkeletonDataComplete.txt",header=TRUE)
head(Skeleton.data)
attach(Skeleton.data)
```

```
  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
```

We have two methods of age estimation here. It is the method of Di Gangi and Suchey-Brooks method. The error of estimation is captured in two variables 'DGerror' and 'SBerror' respectively. The goal is to understand if both methods give the same results or one method is more precise than the other. First let's look at 'DGerror' variable. We get summary statistics for this variable and construct a boxplot ('border' argument in the 'boxplot' function specifies the color of the border of boxplot):

```
summary(DGerror)
boxplot(DGerror,border='blue',ylim=c(-60,40),ylab="Difference between estimated
  and actual age (years)", main="Method of Di Gangi et al.")
```

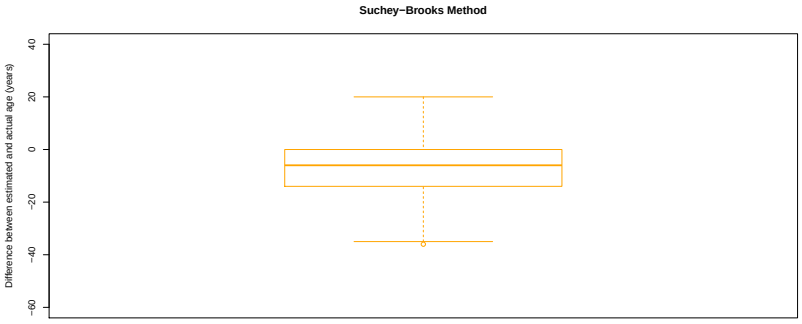
```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 -60.00  -23.00  -13.00  -14.15   -5.00   32.00
```



Similarly we do for the 'SBerror' variable:

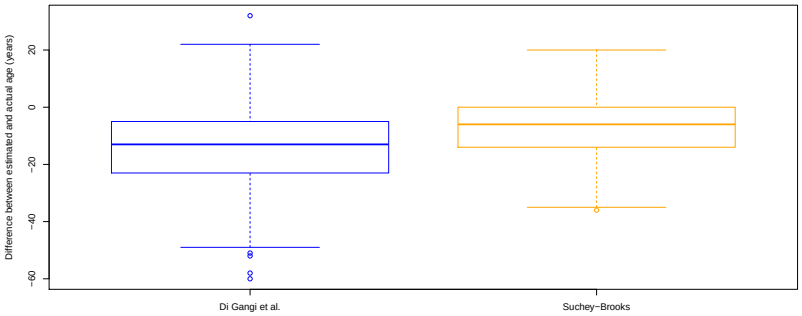
```
summary(SBerror)
boxplot(SBerror,border='orange',ylim=c(-60,40),ylab="Difference between estimated
  and actual age (years)", main="Suchey-Brooks Method")
```

```
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
-36.000 -14.000  -6.000  -7.259   0.000  20.000       2
```



Note that 2 variables are missing in the 'SBerror'. It seems that 'Suchey-Brooks' method is less biased than Di Gangi's method. Let's also make two boxplots side by side:

```
boxplot(list(DGerror,SBerror),ylab="Difference between estimated and actual age (years)",
  border=c('blue','orange'),names=c('Di Gangi et al.','Suchey-Brooks'))
```



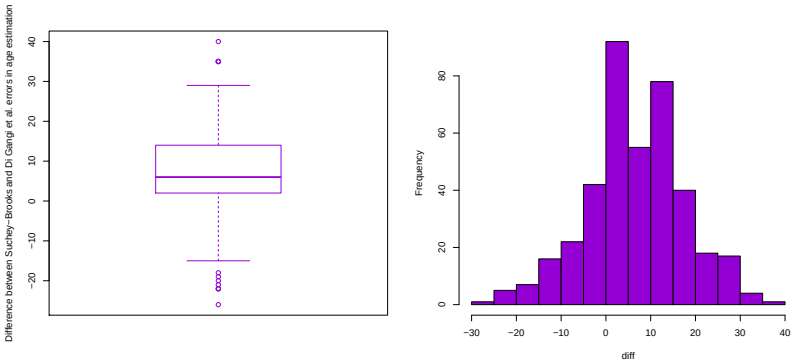
We want to analyse the difference between these two variables. Since these two variables are dependent (since two observations are taken from the same skeleton) we just find difference between 'SBerror' and 'DGerror' and call this new variable 'diff':

```
diff=SBerror-DGerror
```

Next let's make a boxplot, histogram and find summary statistics of the difference variable ('par(mfrow=c(1,2))' make two plots in one chart):

```
par(mfrow=c(1,2))
boxplot(diff,border='darkviolet',ylab="Difference between Suchey-Brooks and
  Di Gangi et al. errors in age estimation")
hist(diff,col='darkviolet',main='')
summary(diff)
```

```
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
-26.000   2.000   6.000   6.854  14.000  40.000       2
```



The distribution of the differences looks nearly normal and therefore we can use one sample t-test to check if the true mean of difference is zero or not (two sided alternative). So find sample mean, sample variance and sample size of 'diff' variable:

```
diff.bar=mean(diff,na.rm=TRUE)
diff.var=var(diff,na.rm=TRUE)
N=400-2
mu0=0
```

Note that since 'diff' variable contains two missed values we must put 'na.rm=TRUE' in the arguments of 'mean' and 'var' functions. Sample size of 'diff' is $400 - 2$, once again because two values are missing and we have a total of 400 skeletons. Finally we proceed in a usual way:

```
t.stat=(diff.bar-mu0)/sqrt( diff.var/N )
p.val=2*(pt(-abs(t.stat),df=N-1))
p.val
```

```
[1] 6.024137e-30
```

The p-value is very small and hence we reject null hypothesis that two methods are the same (have the same average error) and conclude that there is a difference between them. Instead of using the long way that we have implemented above, we can use 't.test' function with 'paired=TRUE' argument to indicate that we use matched pairs:

```
t.test(SBerror,DGerror,alternative="two.sided",paired=TRUE)$p.value
```

```
[1] 6.024137e-30
```

Two p-values are completely the same. If we are interested in the 95% confidence interval for the difference we can use the formulas from the one sample CI or use the 't.test':

```
t.test(SBerror,DGerror,conf.level=0.95,alternative="two.sided",paired=TRUE)
```

```
        Paired t-test

data:  SBerror and DGerror
t = 12.3681, df = 397, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 5.764761 7.943782
sample estimates:
mean of the differences
               6.854271
```

Hence we are 95% sure that the true mean difference of errors is between 5.76 and 7.94.

Comparing Two Proportions

In this section we show how two compare two independent proportions. We start with the 'Support for the Toronto mayor Rob Ford' example. We have two support surveys. In the first one the sample size was 1050 with sample proportion of support equals to 0.57 and another one with sample size of 1046 and sample proportion 0.42. The goal is to get 95% confidence interval for the difference in proportions. As usual we start with variable initiation and find the critical value:

```
N1=1050
p1.hat=0.57
N2=1046
p2.hat=0.42
conf.level=0.95
crit.val=qnorm( 1-(1-conf.level)/2 )
```

Next we use the formula for the margin of error for difference in proportions and then print the answer using 'cat' function:

```
ME=crit.val*sqrt( p1.hat*(1-p1.hat)/N1 + p2.hat*(1-p2.hat)/N2 )
cat('CI for p1-p2 is from ',p1.hat-p2.hat-ME,' to ',p1.hat-p2.hat+ME, '\n')
```

```
CI for p1-p2 is from  0.1076759  to  0.1923241
```

Hence we are 95% confident that the true proportion dropped from 0.10 to 0.19. To use the 'prop.test' here we first need to construct a 2 by 2 table. In the first row we have results from the first survey and in the second one from the second survey. The first column is total number of individuals that supported the mayor and in the second column number of people in a survey that did not support the mayor. To get this table use the following commands:

```
table=rbind(c(p1.hat*N1,(1-p1.hat)*N1),c(p2.hat*N2,(1-p2.hat)*N2))
prop.test(table,conf.level=0.95,correct=FALSE)$conf.int
```

```
[1] 0.1076759 0.1923241
attr(,"conf.level")
[1] 0.95
```

You see that we get exactly the same confidence interval. Now we repeat the same procedure but for the 'Support for US president Obama' example:

```
N1=1010
p1.hat=0.52
N2=563
p2.hat=0.48
table=rbind(c(p1.hat*N1,(1-p1.hat)*N1),c(p2.hat*N2,(1-p2.hat)*N2))
prop.test(table,conf.level=0.95,correct=FALSE)$conf.int
```

```
[1] -0.0115015  0.0915015
attr(,"conf.level")
[1] 0.95
```

In this case we cannot be sure that the support dropped since 0 is inside the confidence interval.

Let's return to the 'Rob Ford polls' example. Here we want to test equality of two proportions. The formula is completely the same as before but with a pooled sample proportion:

```
N1=1050
p1.hat=0.57
N2=1046
p2.hat=0.42
p.hat.pooled=(N1*p1.hat+N2*p2.hat)/(N1+N2)
z.stat=(p1.hat-p2.hat)/sqrt( p.hat.pooled*(1-p.hat.pooled)*(1/N1 + 1/N2) )
p.val=2*pnorm(-abs(z.stat))
p.val
```

```
[1] 6.527937e-12
```

The p-value is very small and we reject the null hypothesis that two proportions are the same and conclude that they are not the same. Equivalently we can use the 'prop.test' function with appropriate table in the argument.

```
table=rbind(c(p1.hat*N1,(1-p1.hat)*N1),c(p2.hat*N2,(1-p2.hat)*N2))
prop.test(table,alternative="two.sided",correct=FALSE)
```

```
        2-sample test for equality of proportions without continuity
        correction

data:  table
X-squared = 47.1643, df = 1, p-value = 6.528e-12
alternative hypothesis: two.sided
95 percent confidence interval:
 0.1076759 0.1923241
sample estimates:
prop 1 prop 2
  0.57   0.42
```

See that the p-value is the same as we got before. This approach is however much quicker and easier. Returning to the 'Obama polls' we want to test that two proportions are the same against the alternative that they are not the same. Using the 'prop.test' function we get:

```
N1=1010
p1.hat=0.52
N2=563
p2.hat=0.48
table=rbind(c(p1.hat*N1,(1-p1.hat)*N1),c(p2.hat*N2,(1-p2.hat)*N2))
prop.test(table,alternative="two.sided",correct=FALSE)
```

```
        2-sample test for equality of proportions without continuity
        correction

data:  table
X-squared = 2.3139, df = 1, p-value = 0.1282
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.0115015  0.0915015
sample estimates:
prop 1 prop 2
  0.52   0.48
```

The p-value for this data is 0.13 which is quite large and we do not have enough statistical evidence to reject null hypothesis. Two actual proportions can be the same.
To finish this section we consider the 'Patricia study'. The first sample proportion here is the proportion of women who got HPV vaccine and have HPV infection. The second sample proportion is proportion of women who got some other vaccine and have HPV infection. We first fill up the statistics for this study, and construct an appropriate table for 'prop.test' function:

```
N1=6163
p1.hat=23/N1
N2=6018
p2.hat=345/N2
table=rbind(c(p1.hat*N1,(1-p1.hat)*N1),c(p2.hat*N2,(1-p2.hat)*N2))
table
```

```
     [,1] [,2]
[1,]   23 6140
[2,]  345 5673
```

Hence we see that in the study with the HPV vaccine, 23 women had HIV infection and 6140 did not have and similarly for the second study (with some other vaccine) 345 had the infection while 5673 did not. Next we just use 'prop.test' function and extract only the p-value with the $ sign:

```
prop.test(table,conf.level=0.95,alternative="two.sided",correct=FALSE)$p.value
```

```
[1] 6.893919e-67
```

The p-value is very small and therefore we reject the null hypothesis and conclude that the HPV vaccine does make a difference.

Comparing Two Means

In this section we show how to compare two independent quantitative groups. Consider first the 'Skeleton' data set. We want to find the 95% confidence interval for the difference between DGerror for male and female. First we divide 'DGerror' variable into two variables corresponding to male and female and call them 'DGerror.male' and 'DGerror.female'. We can do that using the next commands:

```
DGerror.male=DGerror[Sex==1]
DGerror.female=DGerror[Sex==2]
```

The next step is to find sample mean, sample variance and sample sizes for each variable:

```
N1=length(DGerror.male)
N2=length(DGerror.female)
x1.bar=mean(DGerror.male)
x2.bar=mean(DGerror.female)
sam.var1=var(DGerror.male)
sam.var2=var(DGerror.female)
```

As usual for means, we will use student-t distribution with appropriate degrees of freedom. Two compare two means as we do here the formula for degrees of freedom is complicated but we still to it for completeness:

```
DF=(sam.var1/N1 + sam.var2/N2)^2 / ( (sam.var1/N1)^2/(N1-1) + (sam.var2/N2)^2/(N2-1) )
DF
```

```
[1] 200.0947
```

Finally we find critical value than get margin of error and print the answer:

```
conf.level=0.95
crit.val=qt(1-(1-conf.level)/2 , df=DF)
ME=crit.val*sqrt(sam.var1/N1 + sam.var2/N2)
cat('CI for mu1-mu2 is from ',x1.bar-x2.bar-ME,' to ',x1.bar-x2.bar+ME, '\n')
```

```
CI for mu1-mu2 is from  1.026367  to  7.374602
```

Hence we are 95% confident that error of estimation for male is from 1.03 to 7.37 larger than for female. Of course there is a shorter way to get this confidence interval using 't.test' function:

```
t.test(DGerror.male,DGerror.female,conf.level=0.95)$conf.int
```

```
1] 1.026367 7.374602
attr(,"conf.level")
[1] 0.95
```

We get exactly the same CI but much easier.

Consider next the 'Life Expectancy' data set:

```
LifeExp.data=read.table("LifeExpComplete.txt",header=TRUE)
head(LifeExp.data)
attach(LifeExp.data)
```

```
      Country Region LifeExp       GDP HIV
1 Afghanistan    SAs  48.673        NA  NA
2     Albania   EuCA  76.918        NA  NA
3     Algeria   MENA  73.131  6406.817 0.1
4      Angola    SSA  51.093  5519.183 2.0
5   Argentina   Amer  75.901 15741.046 0.5
6     Armenia   EuCA  74.241  4748.929 0.1
```

In this example we need 95% confidence interval for difference between 'LifeExp' for East Asia & Pacific and South Asia. First we construct two variables, one is 'LifeExp' for 'EAP' region and second for 'SAs' region:

```
LifeExp.EAP=LifeExp[Region=='EAP']
LifeExp.SAs=LifeExp[Region=='SAs']
```

To finish we use the 't.test' function:

```
t.test(LifeExp.EAP,LifeExp.SAs,conf.level=0.95)$conf.int
```

```
[1] -1.216372 13.323188
attr(,"conf.level")
[1] 0.95
```

So we are 95% sure that the true difference in averages of Life expectancy for these two regions is between -1.22 and 13.32.

The other important inference topic is testing. Consider again 'DGerror.male' and 'DGerror.female' variables. We have already found the confidence interval for the difference in means, now we want to test whether two means are the same or not (two sided alternative). We start with basic statistics and degrees of freedom calculation:

```
N1=length(DGerror.male)
N2=length(DGerror.female)
x1.bar=mean(DGerror.male)
x2.bar=mean(DGerror.female)
sam.var1=var(DGerror.male)
sam.var2=var(DGerror.female)
DF=(sam.var1/N1 + sam.var2/N2)^2 / ( (sam.var1/N1)^2/(N1-1) + (sam.var2/N2)^2/(N2-1) )
```

Finishing we find test statistic and two sided p-value:

```
t.stat=(x1.bar-x2.bar)/sqrt(sam.var1/N1 + sam.var2/N2)
p.val=2*pt(-abs(t.stat),df=DF)
p.val
```

```
[1] 0.009752372
```

The p-value is quite small and we reject the null hypothesis that error of estimation for male and female is the same. Using the 't.test' function we get:

```
t.test(DGerror.male,DGerror.female,alternative="two.sided")
```

```
        Welch Two Sample t-test

data:  DGerror.male and DGerror.female
t = 2.6095, df = 200.095, p-value = 0.009752
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 1.026367 7.374602
sample estimates:
mean of x mean of y
-12.90036 -17.10084
```

In the second row we observe the test statistic, the 'complicated' degrees of freedom and the p-value which is exactly the same that we have got before. Using the 't.test' we can compare 'LifeExp.EAP' with 'LifeExp.SAs':

```
t.test(LifeExp.EAP,LifeExp.SAs,alternative="two.sided")
```

```
        Welch Two Sample t-test

data:  LifeExp.EAP and LifeExp.SAs
t = 1.8809, df = 9.088, p-value = 0.09236
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -1.216372 13.323188
sample estimates:
mean of x mean of y
 73.08603  67.03262
```

Here the p-value is 0.09 which is considered as large and hence there is no statistical evidence to reject the hypothesis that 'Life Expectancy' for two region are same. So they can be the same.

Sometimes we can assume that the variances for each group are the same. Then calculation for degrees of freedom become very easy. Suppose that true variances of 'DGerror.male' and 'DGerror.female' are the same. Then we calculate the pooled variance and get the test statistic:

```
sam.var.pooled=( (N1-1)*sam.var1 + (N2-1)*sam.var2 )/ (N1+N2-2)
t.stat=(x1.bar-x2.bar)/sqrt(sam.var.pooled/N1 + sam.var.pooled/N2)
t.stat
```

```
[1] 2.740948
```

Once we know the test statistic we can easily find the two sided p-value (note that degrees of freedom in this case is straight forward):

```
p.val=2*pt(-abs(t.stat),df=(N1+N2-2))
p.val
```

```
[1] 0.006401581
```

The p-value is different than what we have got when we did not assume that two variances are the same. The conclusion however is the same, we reject null hypothesis. To use 't.test' with equal variance assumption we must put 'var.equal=TRUE' in the argument:

```
t.test(DGerror.male,DGerror.female,alternative="two.sided",var.equal=TRUE)
```

```
        Two Sample t-test

data:  DGerror.male and DGerror.female
t = 2.7409, df = 398, p-value = 0.006402
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 1.187691 7.213277
sample estimates:
mean of x mean of y
-12.90036 -17.10084
```

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()
qnorm()
qt() [df]
```

Confidence Intervals and Testing

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

Miscellaneous

```
rbind()
summary()
```

.