

Qualint: Test for Qualitative Interactions from Complete Data

Lixi Yu
Eun-Young Suh
Guohua (James) Pan

October 6, 2014

1 Introduction

This is a short tutorial on using the “qualint” function. The “qualval” function could be used to test for qualitative interactions between treatment effects and patient subgroups. The term “treatment effect” here means the comparison between two treatments. It includes two testing methods: Interval Based Graphical Approach and Gail Simon Likelihood Ratio Test, among which the IBGA is the default. The type of response variable included inside are: continuous, binary and time to event. Besides, for continuous response, the treatment effect scale is the mean difference; for binary response, three scales are available: the risk difference, the relative risk and the odds ratio; for time to event data, hazard ratio is adopted.

2 Example

First, let’s install the “QualInt” package.

```
> library("QualInt")
```

2.1 Continuous Response

In order to show how to use function “qualint” to do testing for continuous responses, we need to make up the data first.

```
> ynorm <- rnorm(300)
> trtment <- sample(c(0, 1), 300, prob = c(0.4, 0.6),
+                 replace = TRUE)
> subgrp <- sample(c(0, 1, 2), 300, prob = c(1/3, 1/3, 1/3),
+                 replace = TRUE)
```

The default of the “qualint” function is `type = "continuous"`, `scale = "MD"`, `method = "IBGA"`, `alpha = 0.05` and `plotout = FALSE`, which means the function will use the Interval Based Graphical Approach to do the testing for continuous responses with the mean difference as the scale of treatment effects by default. Also, the significance level is set as 0.05 without outputting

the testing graph. Therefore, if we are good with the default settings, we could call our function now.

```
> out1 <- qualint(ynorm, trtment, subgrp)
```

If you prefer other setting for the parameters in the function, you need to specify them yourself. For example, if you want to let the function output the IBGA graph, you need to let `plotout = TRUE`.

```
> out2 <- qualint(ynorm, trtment, subgrp, type = "continuous", plotout = TRUE)
```

If instead of IBGA, you want to see the results of LRT, you could also let `test = "LRT"`. However, there will be no graph output for LRT.

```
> out3 <- qualint(ynorm, trtment, subgrp, test = "LRT", type = "continuous")
```

The “qualint” function will return an “qualint” object as the result. Now let’s see how the testing results are shown. The basic one is to type the name of the object. For example, type `out1` or `print(out1)`.

```
> print(out1)
```

Call:

```
qualint(y = ynorm, trtment = trtment, subgrp = subgrp)
```

Type:

```
continuous
```

Estimating Results for Mean Difference:

	Estimate	Std. Error	Lower CI	Upper CI
0	0.2703	0.2180	-0.1570	0.6976
1	0.1690	0.1898	-0.2031	0.5411
2	-0.1828	0.2092	-0.5927	0.2272

Test:

```
IBGA
```

p-value:

```
0.3457
```

Power:

```
0.04908
```

Alpha:

```
0.05
```

The main information of the testing results are listed as above. Reference means the reference treatment used in the comparison. For example, here the mean difference means the mean response of treatment 1 minus the mean response of treatment 0 since the treatment 0 is considered as the reference one. Another thing that needs to pay attention is Lower. Int and Upper. Int here is not the confidence intervals we used widely, they are actually the lower and upper limits of the testing interval from the IBGA. For LRT, the results are mostly the same except that it won’t have testing intervals.

```
> print(out3)
```

Call:

```
qualint(y = ynorm, trtment = trtment, subgrp = subgrp, type = "continuous",  
        test = "LRT")
```

Type:

continuous

Estimating Results for Mean Difference:

	Estimate	Std. Error	Lower CI	Upper CI
0	0.2703	0.2180	-0.1570	0.6976
1	0.1690	0.1898	-0.2031	0.5411
2	-0.1828	0.2092	-0.5927	0.2272

Test:

LRT

p-value:

0.3618

Power:

0.0512

Alpha:

0.05

If we also want to see what information an “qualint” object contain or we need to extract part of the information for further analysis, the function `summary` would be helpful.

```
> summary(out1)
```

	Length	Class	Mode
call	4	-none-	call
n	2	-none-	list
type	1	-none-	character
alpha	1	-none-	numeric
treatment	2	-none-	character
reference	1	-none-	character
nsbp	1	-none-	numeric
subgroup	3	-none-	character
scale	1	-none-	character
effect	3	-none-	numeric
se	3	-none-	numeric
LowerCI	3	-none-	numeric
UpperCI	3	-none-	numeric
test	1	-none-	character
index	1	-none-	numeric
LowerTI	3	-none-	numeric
UpperTI	3	-none-	numeric

```

pvalue    1      -none- numeric
power     1      -none- numeric
nobs      1      -none- numeric
missing   0      -none- numeric

```

Then you could just use the dollar sign \$ to get all the information you need from an “qualint” object. For example, if only p-value is what you need, just use `out$pvalue`.

```

> out1$pvalue

[1] 0.3457183

```

If you want to check the detailed explanation of all the elements in the “qualint” object, then go to the help page for the “qualint” function or type `?QualInt`.

```

> ?QualInt

```

If you only need the estimation results related to the testing, just type `coef(out1)` or `coefficients(out1)` which will return a matrix containing all the estimating results.

```

> coef(out1)

Estimation Results for Mean Difference:
  Estimate Std. Error Lower CI Upper CI
0   0.2703    0.2180  -0.1570  0.6976
1   0.1690    0.1898  -0.2031  0.5411
2  -0.1828    0.2092  -0.5927  0.2272

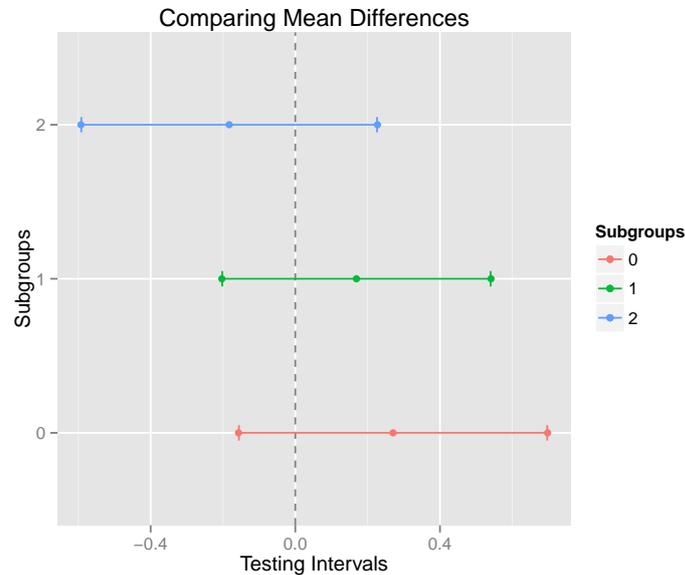
```

For IBGA, we also provide the testing graph. You could output it simply by specifying `plotout = TRUE` or type `plot(out1)`. Notice that the left dot and the right dot for each subgroup are the lower and upper limits of the confidence interval, and the middle dot is the location of the estimated treatment effect. Besides, the function “ibga” could be applied to the “qualint” object to produce the estimating results related with the plot. For example, the testing intervals of the scale chose for the IBGA graph. Notice that this two functions are not available for `test = "LRT"`.

```

> plot(out1)

```



```
> ibga(out1)
```

IBGA Results for Mean Difference:

	Estimate	Std. Error	Lower CI	Upper CI	Lower TI	Upper TI
0	0.2703	0.2180	-0.1570	0.6976	-0.1558	0.6965
1	0.1690	0.1898	-0.2031	0.5411	-0.2021	0.5401
2	-0.1828	0.2092	-0.5927	0.2272	-0.5916	0.2261

2.2 Binary Response

Since the default of the “qualint” function is for continuous responses, we need to specify the parameters ourselves when we want to doing the testing for binary responses. The default of the treatment effect scale for the binary response is `scale = "RD"` (Risk Difference).

```
> ybin <- sample(c(0, 1), 300, prob = c(0.3, 0.7), replace = TRUE)
> out4 <- qualint(ybin, trtment, subgrp, type = "binary", scale = "RD")
> print(out4)
```

Call:

```
qualint(y = ybin, trtment = trtment, subgrp = subgrp, type = "binary",
       scale = "RD")
```

Type:
binary

Estimating Results for Risk Difference:

	Estimate	Std. Error	Lower CI	Upper CI
0	-0.01148	0.09404	-0.1958	0.17283
1	-0.08156	0.07926	-0.2369	0.07379

```
2 0.02041 0.10614 -0.1876 0.22845
```

```
Test:  
IBGA
```

```
p-value:  
0.668
```

```
Power:  
0.01153
```

```
Alpha:  
0.05
```

If instead of the risk difference, you prefer the relative risk and LRT, then just specify `scale = "RR"` and `test = "LRT"`. Notice that the testing is based on the log of relative risk instead of relative risk itself.

```
> out5 <- qualint(ybin, trtment, subgrp, type = "binary",  
+               scale = "RR", test = "LRT")  
> print(out5)
```

```
Call:  
qualint(y = ybin, trtment = trtment, subgrp = subgrp, type = "binary",  
        scale = "RR", test = "LRT")
```

```
Type:  
binary
```

```
Estimating Results for log(Relative Risk):
```

	Estimate	Std. Error	Lower CI	Upper CI
0	-0.01984	0.1419	-0.2980	0.2583
1	-0.10566	0.1012	-0.3040	0.0927
2	0.02482	0.1445	-0.2584	0.3081

```
Test:  
LRT
```

```
p-value:  
0.6782
```

```
Power:  
0.0103
```

```
Alpha:  
0.05
```

The use of other “qualint” object related functions, like `coef`, `print` and `plot` are the same as for the continuous response.

2.3 Time to Event Response

The treatment effect scale for the time to event response is `scale = "HR"` (Hazard Ratio) by default, therefore you don't need to specify it specifically. Also, you don't need to specify the response type `type = "survival"` specifically since the function will recognize it through the "Surv" object of the response. However, it will signal a warning to remind the user. The use of other functions are the same as mentioned above.

```
> time <- rpois(300, 200)
> censor <- sample(c(0, 1), 300, prob = c(0.7, 0.3), replace = TRUE)
> out6 <- qualint(Surv(time, censor), trtment, subgrp, scale = "HR", type = "survival")
> print(out6)
```

Call:

```
qualint(y = Surv(time, censor), trtment = trtment, subgrp = subgrp,
        type = "survival", scale = "HR")
```

Type:

survival

Estimating Results for log(Hazard Ratio):

	Estimate	Std. Error	Lower CI	Upper CI
0	0.9059	0.3791	0.1629	1.6488
1	-0.2461	0.3646	-0.9608	0.4685
2	0.4598	0.4810	-0.4829	1.4024

Test:

IBGA

p-value:

0.4372

Power:

0.07345

Alpha:

0.05