Exersices and solutions — A-not A and same-different tests in sensR

Rune Haubo B Christensen

September 10, 2013
file: `exerciseThursday.Rnw`

Topics:

Analysis of data from the protocols

- A-not A
- Same-different
- A-not A with sureness

Preliminaries

Before we you get started with the exercises, you need to make sure that you have a reasonably new version of sensR. When you run `sessionInfo()` you should have at least the version of the sensR package shown here:

```
R> sessionInfo()

R version 3.0.1 (2013-05-16)
Platform: x86_64-apple-darwin10.8.0 (64-bit)

locale:
[1] C

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods
[7] base

other attached packages:
[1] sensR_1.2-22      numDeriv_2012.9-1 ordinal_2013.8-25
[4] Matrix_1.0-12     lattice_0.20-15   ucminf_1.1-3

loaded via a namespace (and not attached):
[1] MASS_7.3-28     grid_3.0.1      multcomp_1.2-19 tools_3.0.1
```

If you don't have the newest version, you are probably able to get a newer version with the following command:

```
R> install.packages("sensR", repos="http://R-Forge.R-project.org")
```

Exercise 1

You have conducted an A-not A experiment with 100 subjects. The answers are summarized in table 1.

Table 1: A-not A data for exercise 1.

Sample	Response		Total
	"A"	"not A"	
A	57	43	100
Not A	42	58	100

1. Have you shown that the A and not-A products are different?
2. Have you shown that the A and not-A products are similar (using $d' = 0.75$ as the boundary of similarity) at the 5% level?
3. Have you shown that the A and not-A products are similar at the 1% level (again using $d' = 0.75$ as the boundary of similarity)?

Answer to the exercise:

To analyze the data, we use the `AnotA` function from the `sensR` package:

```
R> AnotA(57, 100, 42, 100)

Call:  AnotA(x1 = 57, n1 = 100, x2 = 42, n2 = 100)

Results for the A-Not A test:

         Estimate Std. Error      Lower     Upper    P-value
d-prime 0.3782676  0.1784076 0.02859508 0.7279402 0.02371745
```

1. Since the p-value is 0.024, i.e. less than 5%, we can conclude that the products are significantly different at the 5% level.
2. To evaluate if the products are similar on the 5% level, we can look at the 90% confidence interval. Here we use the `confint` method to change the default confidence level:

   ```
   R> confint(AnotA(57, 100, 42, 100), level=0.90)

                    5 %      95 %
   threshold -0.03044025 0.3843434
   d.prime    0.08538341 0.6723933
   ```

   since the confidence interval is entirely below $d' = 0.75$, we have shown similarity at this level.
3. We now change the confidence level to 0.98% and reevaluate the confidence interval:

   ```
   R> confint(AnotA(57, 100, 42, 100), level=0.98)
   ```

```
                   1 %      99 %
threshold -0.11588798 0.4709462
d.prime   -0.03567145 0.7946889
```

Since the upper confidence level is now *above* $d' = 0.75$, we have *not* shown similarity at this level.

Exercise 2

It has come to you knowledge that the technicians undertaking the test formulated the question in the following mannor:

"You are now given two samples and you are asked to determine if these samples are the same or not. Answer A if you believe the samples are the same and *not* A if you believe they are not the same"

As you may realize, these are the instructions for the same-different test rather than the A-not A test, and we will have to analyze the data assuming the same-different cognitive decision rule rather than that of the A-not A.

1. Estimate d' assuming the same-different protocol instead of the the A-not A protocol. Does the estimate of d' change? and if so, is it a large difference or a small unimportant one?
2. What is the p-value of the test of "no product difference"? Compare this to the p-value obtained using the A-not A test. Has the p-value changed? Is it important?
3. Make a similarity analysis similar to the one for the A-not A analysis. Are there any differences?
4. At what boundary-of-similarity (d'_0) are you able to declare the products 'similar' at the 5% level?

Answer to the exercise:

1. We analyze the data with the `samediff` function:

```
R> sd1 <- samediff(57, 43, 42, 58)
R> summary(sd1)

Call:
samediff(nsamesame = 57, ndiffsame = 43, nsamediff = 42, ndiffdiff = 58)

Coefficients
      Estimate Std. Error  Lower  Upper P-value
tau     1.1161     0.1198 0.8927 1.3616  <2e-16 ***
delta   1.2268     0.3098 0.3325 1.7659  0.0168 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Log Likelihood: -136.3607  AIC: 276.7214
```

Here $d' = 1.23$ which we should compare to $d' = 0.37$ from the A-not A test. This is a fairly large change. The assumption about which decision rule was used when generating the data is a really important one when it comes to the interpretation of d'.

2. The p-value for the "no product difference" test can also be read of the output above as the p-value for `delta` — it is $p = 0.017$. This p-value is not so different from the p-value from the A-not A test (which was $p = 0.024$); both are significant on the 5% level, but not on the 1% level.

3. Since even the same-different estimate of d' is larger than the similarity limit of 0.75, the same-different analysis does not indicate similarity at any reasonable level.

4. To compute the similarity boundary, d'_0 for the 5% level we want to look at the upper confidence limit of the 90% CI:

```
R> confint(sd1, level=0.90)
```

```
                Lower    Upper
0.9% tau   0.9270242 1.320723
0.9% delta 0.5688371 1.685508
```

This shows that we would be able to declare the products similar at the 5% level, if we only consider products with $d' \geq 1.69$ as different. However, that is an unreasonable large boundary of similarity.

Exercise 3

You have conducted a large-scale consumer study using the A-not A with sureness protocol and obtained the data in the following table.

```
      sureness
prod    1  2  3  4  5  6
  ref  10 40 70 50 20 10
  test 20 30 20 30 60 40
```

To get the data into R you may use the following commands:

```
R> wts <- c(10, 40, 70, 50, 20, 10, 20, 30, 20, 30, 60, 40)
R> dat <- data.frame(sureness = factor(rep(1:6, 2), ordered=TRUE),
+                    prod = factor(rep(c("ref", "test"), each = 6)),
+                    freq = wts)
R> dat
```

```
   sureness prod freq
1         1  ref   10
2         2  ref   40
3         3  ref   70
4         4  ref   50
5         5  ref   20
6         6  ref   10
7         1 test   20
8         2 test   30
9         3 test   20
```

```
10        4 test   30
11        5 test   60
12        6 test   40

R> ## Tabulate the data:
R> xtabs(freq ~ prod + sureness, dat)

      sureness
prod    1  2  3  4  5  6
  ref  10 40 70 50 20 10
  test 20 30 20 30 60 40
```

1. First assume that there i no difference in scale: compute d' using the `clm` function. Also compute the *sensitivity* and the overlap of the perceptual distributions.
2. Test if the products are different assuming the equal-variances model.
3. Now fit the model that allows for differences in scale/unequal variances for the two products.
4. Test if the variances are different or whether they can be assumed to be equal.
5. Test if products are different in the unequal-variances model and compare results to the question 2.
6. Compute d', the scale-ratio (i.e. the standard deviation of the 'test' distribution assuming the standard deviation of the 'reference' distribution is 1), the sensitivity and the overlap of the perceptual distributions.
7. Compare and discuss differences and similarities in d', sensitivity and distribution overlap.

Answer to the exercise:

1. We first compute d' using `clm` with a probit link:

```
R> fm1 <- clm(sureness ~ prod, data=dat, weights=freq, link="probit")
R> summary(fm1)

formula: sureness ~ prod
data:    dat

 link   threshold nobs logLik  AIC     niter max.grad cond.H
 probit flexible  400  -685.27 1382.55 4(0)  9.73e-10 2.9e+01

Coefficients:
         Estimate Std. Error z value Pr(>|z|)
prodtest   0.4768     0.1058   4.505 6.63e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2 -1.21759    0.10424 -11.681
2|3 -0.47012    0.08196  -5.736
```

```
3|4  0.15118    0.07912   1.911
4|5  0.69946    0.08590   8.143
5|6  1.43929    0.10470  13.747
```

So here $d' = 0.477$ with standard error 0.11.

The sensitivity is given by $\mathcal{S} = \Phi(d'/\sqrt{2})$ which we compute with

```
R> pnorm(fm1$beta / sqrt(2))

 prodtest
0.6319961
```

Notice that d' is stored in the `beta` element of the fit.

The degree of distribution overlap is given by $\lambda = 2\Phi(-d'/2)$ which we evaluate with

```
R> 2 * pnorm(-fm1$beta / 2)

prodtest
0.811573
```

2. To test if products are different, we can either look at the p-value from the Coefficient table in the summary output above or we can use the `anova` function to compute the (more accurate) likelihood ratio test. The p-value from the summary is already highly significant ($p < 0.001$) and the likelihood ratio test is not going to change that:

```
R> ## First estimate the null model:
R> fm0 <- clm(sureness ~ 1, data=dat, weights=freq, link="probit")
R> ## Then compare the models with anova:
R> anova(fm0, fm1)

Likelihood ratio tests of cumulative link models:

    formula:        link:  threshold:
fm0 sureness ~ 1    probit flexible
fm1 sureness ~ prod probit flexible

    no.par    AIC  logLik LR.stat df Pr(>Chisq)
fm0      5 1400.9 -695.45
fm1      6 1382.5 -685.27  20.348  1  6.457e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Here the p-value is even slightly smaller. The conclusion is that d' is highly significantly different from zero.

3. To estimate the unequal-variances model, we add `prod` to the `scale` formula in `clm`:

```
R> fm2 <- clm(sureness ~ prod, scale=~prod, data=dat, weights=freq, link="probit")
R> summary(fm2)

formula: sureness ~ prod
scale:   ~prod
data:    dat

 link   threshold nobs logLik  AIC     niter max.grad cond.H
 probit flexible  400  -666.92 1347.84 8(0)  2.53e-10 2.2e+01

Coefficients:
```

```
         Estimate Std. Error z value Pr(>|z|)
prodtest   0.6730     0.1525   4.414 1.01e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

log-scale coefficients:
         Estimate Std. Error z value Pr(>|z|)
prodtest   0.5290     0.0881   6.004 1.92e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Threshold coefficients:
    Estimate Std. Error z value
1|2 -1.60172    0.13752 -11.647
2|3 -0.60572    0.08960  -6.760
3|4  0.17736    0.08315   2.133
4|5  0.89787    0.09875   9.093
5|6  1.95877    0.15409  12.711
```

4. To test if we really need th unequal-variances model or if the equal-variance model is sufficient, we compare the two models with anova:

```
R> anova(fm1, fm2)
```

```
Likelihood ratio tests of cumulative link models:

    formula:        scale: link:  threshold:
fm1 sureness ~ prod ~1     probit flexible
fm2 sureness ~ prod ~prod  probit flexible

    no.par    AIC  logLik LR.stat df Pr(>Chisq)
fm1      6 1382.5 -685.27
fm2      7 1347.8 -666.92  36.708  1  1.372e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

   This test is highly significant, so we must retain the unequal-variances model.

5. To test if products are different, we want to test if both location and scale parameters, so we compare `fm0` and `fm2` in an anova test:

```
R> anova(fm0, fm2)
```

```
Likelihood ratio tests of cumulative link models:

    formula:        scale: link:  threshold:
fm0 sureness ~ 1    ~1     probit flexible
fm2 sureness ~ prod ~prod  probit flexible

    no.par    AIC  logLik LR.stat df Pr(>Chisq)
fm0      5 1400.9 -695.45
fm2      7 1347.8 -666.92  57.055  2  4.079e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

   Notice that this is a test on 2 degrees of freedom (one for d' and one for the scale). As in question 2 the product test is highly significant, but the p-value is even smaller than

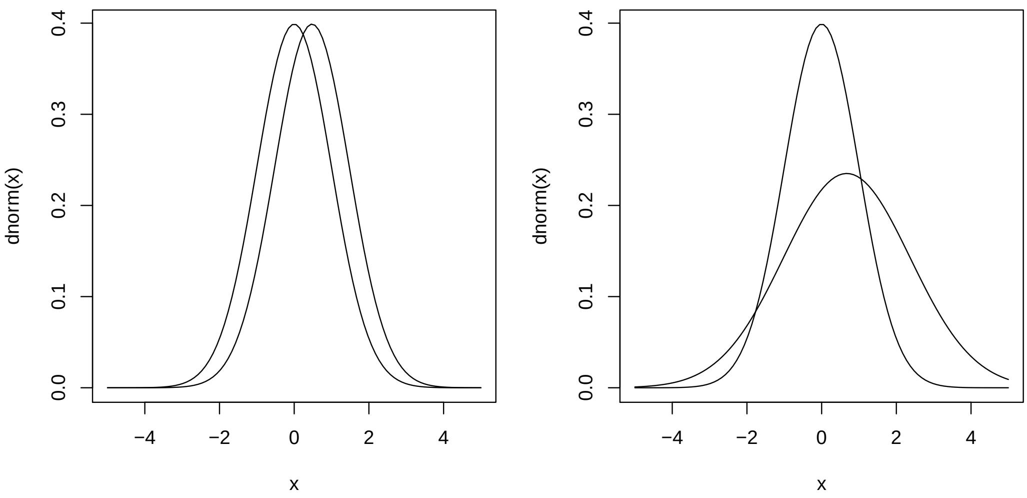

Figure 1: Perceptual distributions, left: equal variance model, right: unequal variance model.

in the equal-variance model since the unequal-variances model fits these data much better.

6. d' was computed in model `fm2` adn shown in the summary of that model above; $d' = 0.67$. Notice how `clm` reports the *log* of the scale component, so we have to take the antilog to get the scale (actually the scale ratio) itself:

```
R> exp(0.529)

[1] 1.697234

R> ## alternatively:
R> exp(fm2$zeta)

prodtest
1.697175
```

The sensitivity is now given as $\mathcal{S} = \Phi(d'/\sqrt{1 + \sigma^2})$ which we compute with

```
R> pnorm(fm2$beta, sqrt(1 + exp(fm2$zeta)^2))

 prodtest
0.0973413
```

The distribution overlap can be computed with the `overlap` function:

```
R> overlap(fm2$beta, exp(fm2$zeta))

[1] 0.7636128
```

7. It is interesting to notice that d' goes up from 0.477 to 0.67 and distrbution overlap goes down from 0.812 to 0.764 , but that sensitivity goes down from 0.17 to 0.097! Here d' and distribution overlap pulls in the same direction, but the sensitivity pulls in the opposite direction. The perceptual distributions for the two models are shown in figure 1.