An introduction to sensR

Rune Haubo B Christensen

January 20, 2014

file: `sensR_intro.Rnw`

Abstract

The aim of this introduction is to provide a brief overview of the sensR package and explain how to use it with a small example. We assume that you are new to the sensR package and possibly also to R it self. While not needed to complete this introduction, it is good if you have heard of *sensory discrimination testing* before.

Contents

1 The sensR package — an overview.

The sensR package is a an R package for sensory discrimination testing developed by Rune Haubo Bojesen Christensen and Per Bruun Brockhoff. The package facilitates, among other things:

- Statistical tests of sensory discrimation data
- Power and sample size computations for discrimination tests
- Thurstonian analyses via d' (d-prime) estimation
- Improved confidence intervals via profile likelihood methods

Table 1 describes which sensory discrimination methods are supported by the sensR package and which features the sensR package provides for these discrimination methods. Absent check marks indicate that the feature is not implemented.

Table 2 provides a summary of the most important functions in (or related to) the sensR package. Use this table as an entry point if you are looking for a function to perform a particular task. You can always look at the help page for the function of interest to see exactly how to use it cf. section 2.

Table 1: Support for sensory discrimination methods available in sensR.

Discrimination	d' estimation	Difference test	Similarity test	Power	Sample size	Simulation	Likelihood CI	Replicated	Regression analysis
Duo-Trio, Triangle, Tetrad	✓	✓	✓	✓	✓	✓	✓	✓	✓
2-AFC, 3-AFC	✓	✓	✓	✓	✓	✓	✓	✓	✓
A-not A	✓	✓	✓				✓	✓	✓
Same-Different	✓	✓	(✓)	✓		✓	✓		
2-AC	✓	✓	✓	✓			✓	✓	✓
A-not A w. Sureness	✓	✓	(✓)				✓	✓	✓

Table 2: Which function to use for a particular tast; summary of the most important functions provided by or related to the sensR package.

d', CI, tests	Power & Sample size	Transformation	Illustration	*d'* comparisons	Miscellaneous
discrim	discrimPwr	rescale	plot	dprime_compare	findcr
AnotA	d.primePwr	psyfun	ROC	dprime_test	clm2twoAC
samediff	discrimSS	psyinv	AUC	dprime_table	SDT
twoAC	d.primeSS	psyderiv		posthoc	discrimR
betabin	twoACpwr	pd2pc			discrimSim
glm		pc2pd			samdiffSim
clm					
clmm					

2 Getting help

To get help on a particular function, e.g. `discrim`, you can write `help("discrim")` or equivalently `?discrim`. This works well if you know which function you want help on. Say you want to analyse some data from a same-different discrimination experiment, but cannot remember which function you should use (there is, for example, no function called `samedifferent`). One solution is to use two question marks and write `??"same different"` which in this case will lead you to the `samediff` function. As a last resort you can always google it or use the `RSiteSearch` function to look in R functions and mailing list archives (e.g. `RSiteSearch("same different sensory discrimination")`) which opens up a web browser and links to the relevant package/function or mailing list correspondence.

To browse the full entire manual for a package, go to its CRAN website (e.g. cran.r-project.org/package=sensR) where you will find the manual in pdf. Alternatively you can find a HTML version here: http://finzi.psych.upenn.edu/R/library/sensR/html/00Index.html.

3 Obtaining sensR

You need to have the R program installed on your computer before you think about the sensR package. You can download and install the latest version of R from http://cran.r-project.org/. Once you have installed R, you can start using it, but many people (and I encourage that) want to install a suitable GUI (graphical user interface) or IDE (integrated development environment). One suitable choice is the (also free) program Rstudio, which you can download and install from http://www.rstudio.com/ide/download/desktop.

To install the sensR package you open R, either directly or by opening Rstudio. Then you use the `install.packages` function to download the package from the internet and install

it on your local computer:

```
R> install.packages("sensR")
```

or you use the point-and-click facilites of Rstudio.

To actually use the facilities in the sensR package you have tell R that you *require* the sensR package:

```
R> require(sensR)
```

4 R is open source

The sensR package as well as R itself is free open source software. This means 1) that you don't have to pay for the software, but can download it freely from the internet, and that 2) source code is open to your inspection should have the desire to find out how a particular computation is being done. As R and sensR are licenced under the GPL licence (http://www.gnu.org/licenses/gpl.html) there are certain restrictions if you want to re-distribute the software, but if merely want to use it, you essentially don't have to worry about it.

5 Referencing sensR in you work

sensR is piece of scientific work and so we would really appreciate if you cite the package in your reports, papers and other work. To cite the version of sensR that you are using, use the `citation` function:

```
R> citation("sensR")
```

```
To cite the sensR-package in publications use:

  Christensen, R. H. B. & P. B. Brockhoff (). sensR---An
  R-package for sensory discrimination R package version 1.3-0
  http://www.cran.r-project.org/package=sensR/.

A BibTeX entry for LaTeX users is

  @Misc{,
    title = {sensR---An R-package for sensory discrimination},
    author = {R. H. B. Christensen and P. B. Brockhoff},
    note = {R package version 1.3-0 http://www.cran.r-project.org/package=sensR/},
  }
```

This function works for other packages as well — please use it abundantly. It also works for R itself, hence here is how to cite R in your work:

```
R> citation()
```

```
To cite R in publications use:

  R Core Team (2013). R: A language and environment for
  statistical computing. R Foundation for Statistical
```

```
  Computing, Vienna, Austria. URL http://www.R-project.org/.

A BibTeX entry for LaTeX users is

  @Manual{,
    title = {R: A Language and Environment for Statistical Computing},
    author = {{R Core Team}},
    organization = {R Foundation for Statistical Computing},
    address = {Vienna, Austria},
    year = {2013},
    url = {http://www.R-project.org/},
  }

We have invested a lot of time and effort in creating R,
please cite it when using it for data analysis. See also
'citation("pkgname")' for citing R packages.
```

6 Recommended Workflow

We distinguish between *examples* and *exercises*. In an example an exercise is presented followed by a suggested solution complete with R code, output and text. We strongly recommend that you not only *read* the example, but that you also *type* the R code into an R session to learn how to write R code: You will be making mistakes (like writing `lenght` when you mean `length`, and `,` (comma) when you mean `.` (period)) and getting unexpected output. Locating your errors and comparing outputs is an important source of learning!

In the exercises you are on your own: Read the exercises, do your own analyses and write your own answers.

Before you start writing R code at the R *prompt* (`>` or `R>` depending on your settings), we *strongly* encourage you to open a *script*. A script is just a text-file (e.g. `my_first_R_script.R`) in which you type your R commands. You can then *run* the commands directly by parsing the relevant lines into the R prompt where the commands are then executed. Working with R scripts is important because you then have a file with all the commands needed to do, and re-do, your analysis any time; your analyses, exercises and eventually work or research is now *reproducible* (see http://cran.r-project.org/web/views/ReproducibleResearch.html for more on this topic).

7 Example 1

7.1 Exercise: a Triangle experiment

You have conducted a Triangle experiment and it yielded 13 correct answers out of 30 tests (no replicates).

1. Have you proven that the products differ?
2. Have you proven that the products do *not* differ?

3. What range of values for d' (d-prime) and p_d (proportion discriminators) is acceptable given the data you have?

7.2 Suggested solution

7.2.1 Have you proven that the products differ?

To show that the products differ we could look at the p-value for a discrimination test using the Triangle method. If we use as null hypothesis that there is no difference between the products, the p-value will be the probability of observing 13 or more correct answers out of 30 if there really is no difference between the products (i.e. if the null hypothesis is true). We then use the p-value as a measure of evidence against the null hypothesis and implicitly in favour of the alternative hypothesis; that the products are different. Here small p-values, for example (and somewhat conventionally) less than 0.05, indicate evidence against the null hypothesis and in favour of the alternative hypothesis (that the products are different).

To actually compute this p-value, we use the `discrim` function:

```
R> discrim(correct=13, total=30, method="triangle")

Estimates for the triangle discrimination protocol with 13 correct
answers in 30 trials. One-sided p-value and 95 % two-sided confidence
intervals are based on the 'exact' binomial test.

        Estimate Std. Error  Lower  Upper
pc        0.4333    0.09047 0.3333 0.6257
pd        0.1500    0.13571 0.0000 0.4386
d-prime   1.0950    0.54688 0.0000 2.1070

Result of difference test:
'exact' binomial test:  p-value = 0.166
Alternative hypothesis: d-prime is greater than 0
```

Since $p > 0.05$ the null hypothesis, H_0 of the test (that the products are the same) is not rejected meaning that we have failed to show that there is a difference between the products.

7.2.2 Have you proven that the products do *not* differ?

We have already found out that test did not show a difference between the products, but this does not necessarily mean that they are the same! The null hypothesis is not rejected, but that does not mean we can conclude that the null hypothesis is true. It means that we do not have enough evidence to be able to reject the null hypothesis. Just because we cannot show that there is a difference, it does not mean that there is no difference: **Absence of evidence is not evidence of absence!**

7.2.3 What range of values for d' (d-prime) and p_d (proportion discriminators) is acceptable given the data you have?

This is a way to interpret the confidence intervals (CI); the CI helps us refine the answers from question 1.

The 95%-CI for d' (for the conducted test) can be read from the output above, the interval is $[0.00; 2.11]$. Since the lower limit of the interval contains the value 0 this means that "no product difference" is plausible. The upper limit shows that a large product difference with a d' larger than 2 is also plausible.

The lower limit of the 95%-CI for p_d shows (just like the lower limit for d') that there might as well be no difference between the products; the proportion of subjects in the population being able to tell the difference between the products could be 0. The upper limit of the 95%-CI for p_d states that there could be up to 44% (value = 0.44) of the population who could detect a difference — even though H_0 was not rejected in question 1!

8 List of symbols

8.1 General symbols

The following is a list of symbols and names used throughout the examples and exercises related to discrimination testing, and the sensR package.

α Significance level or Type I error rate.

β Type II error rate.

$1 - \beta$ Power (of a statistical test).

δ The Thurstonian measure of sensory or perceptual difference. Sometimes δ is used to denote the true, unknown quantity and d' the observed value (estimate or estimator) of δ.

d' The Thurstonian measure of sensory difference (see above) and the SDT (signal detection theory) measure of perceptual difference.

p_c Used to denote the probability of a correct answer in a sensory discrimination test. A small note: Technically p_c denotes the true, unknown quantity while $\hat{p}_c$ is an observed estimate of p_c, e.g. we can estimate $\hat{p}_c = 15/30 = 0.5$ if we observe 15 correct in 30 trials. Since p_c is the true probability of a correct answer, it cannot be any smaller than the guessing probability, e.g. $p_c \geq 1/3$ for the Triangle test. On the other hand, the *proportion* of correct answers can very well be smaller than the guessing probability; it is not unlikely to observe 9 correct out of 30 trials of product differences are negligible. In that case we would still estimate $\hat{p}_c = 1/3$ and not 9/30!

p_d Used to denote the *proportion of discriminators*. Since this concept implicitly assumes that any person either guesses in all cases or always correctly discriminates the sample, we sometimes interpret p_d as *the probability of discrimination* — a concept that applies to all assessors.

p_g Used to denote the guessing probability of a sensory discrimination method (e.g. $p_g = 1/3$ for the Triangle test).

protocol A sensory discrimination protocol, e.g. Triangle, tetrad or Duo-Trio. We sometimes also refer to it as, say, the Duo-Trio *method* or the Triangle *test*.

8.2 Often used sensR arguments

`alpha` The significance level, α. Used in sample size and power functions to denote the largest allowable type I error of the test.

`conf.level` The confidence level; $1 - \alpha$.

`d.prime0` The value of d' under the null hypothesis.

`d.primeA` The value of d' under the alternative hypothesis.

`method` The sensory discrimination *method* (or *protocol*/*test*). Common available choices are `duotrio`, `tetrad`, `threeAFC` (3-AFC), `twoAFC` (2-AFC), `triangle`.

`pd0` The value of p_d under the null hypothesis.

`pdA` The value of p_d under the alternative hypothesis.

`Pguess` see `pGuess`.

`pGuess` The guessing probability of the discrimination method (1/2 for Duo-Trio and 2-AFC, and 1/3 for Tetrad, Triangle and 3-AFC).

`size` The sample size.

`std.err` The standard error.

`target.power` In sample size computations, the desired (minimum allowable) power of the statistical test.

`test` Usually distinguishes between *similarity* (also known as *equivalence* or *parity*) tests and the conventional *difference* tests. Hence this argument controls the direction of the one-sided hypothesis test.

9 SessionInfo

`sessionInfo` shows the information about the R session that was used when generating this document. This includes the version of R used and versions of loaded ("available") packages.

```
R> sessionInfo()

R version 3.0.2 (2013-09-25)
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_1.3-0
```

```
loaded via a namespace (and not attached):
[1] MASS_7.3-29       multcomp_1.2-21   numDeriv_2012.9-1
[4] tools_3.0.2
```