Rbrul workshop

Lynn Clark (l.clark6@lancaster.ac.uk)

****Big thanks go to Daniel Ezra Johnson for kindly donating data for this workshop, giving a workshop on Rbrul & R at NWAV 39 and, of course, for creating Rbrul in the first place! If you have any suggestions for improvement to Rbrul, Dan would love to hear them (email: danielezrajohnson@gmail.com)****

1. Getting R and Rbrul

If R is not already installed on your machine, get it from here: http://cran.r-project.org/ . Follow the link for "Download packages", then select a UK CRAN, select the machine type you're installing R on (windows, mac or linux), select "base" then follow the link for "download 2.12.0 for windows (if you have a windows machine). NB: these instructions follow a windows version of R; the mac version looks a little different but is essentially the same).

IMPORTANT: when installing R on your machine, when prompted "curtomise start-up options?" select "yes" then when asked which internet connection you want, select "internet 2". This is important if you want to connect to R on the university internet server.

Once R is installed on your machine, it will be necessary to install several packages that Rbrul will use. One simple way to get these packages is to type the command

```
>update.packages()
```

Into the command line. R will then ask you to select a CRAN mirror (choose a UK one) and then it will ask you if you want to install a series of packages (type 'y' for yes). It will then install the basic packages called 'cluster', 'codetools' , 'Matrix' , 'mgcv' , 'rpart' and 'survival'.

You will also need to install several other packages for Rbrul to work, the most important of which is lme4 which is the package underlying the type of regression analysis Rbrul performs. To access the packages you need in order to run Rbrul you can Source and run Rbrul:

```
> source("http://www.danielezrajohnson.com/Rbrul.R")
> rbrul()
```

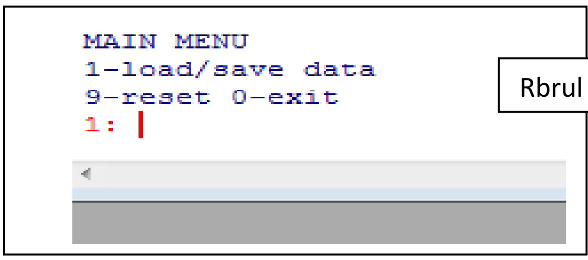
This *should* automatically install the packages that you need but will only work if you have a connection to the internet and admin rights on the machine that you are using (I have had some problems doing this on the university network). If you are having problems, you can also install these packages manually by doing the following:

- Open R and under Packages, choose "Install package(s)" . Choose a mirror near you. Hold down Ctrl and select the following four packages: "boot", "Hmisc", "lattice", and "lme4".
- Run the following four commands in the R window:
```
> library(boot)
> library(Hmisc)
> library(lattice)
> library(lme4)
```

It is worth pointing out that these packages may change from time to time and they get updated. To check for updates and install new versions of already installed packages, simply run the command again...

```
>update.packages()
```

You should now see the following screen and be ready to load some data:



Rbrul main menu

TIP: I keep 2 text files in a folder next to my R icon on my desktop. One contains only the commands to easily load Rbrul (so I don't have to remember them or always look them up)

```
>source("http://www.danielezrajohnson.com/Rbrul.R")
> rbrul()
```

The other is a text file and contains the R script (download this from here http://www.ling.upenn.edu/~johnson4/Rbrul.R)
Copying and pasting this into R, followed by the command >rbrul() will allow you to run Rbrul even if you don't have internet access.

2. Loading data

On a windows version of R, before you can load data, you need to tell R where to look for it.

```
File > change dir...> select a folder that contains your data
```

R can read data in a number of formats (e.g. text files, spss files, Goldvarb token files & excel/.csv files). I always use .csv files because they can be created in excel and are the most transparent way to look at data (I think). To begin with, we'll work with some simple data from Labov's department store study (file 'ds'). To load the data in Rbrul, follow the menu on the screen:

```
MAIN MENU
1-load/save data
9-reset 0-exit
1: 1

No data loaded.

What separates the columns in the data file to open?
(c-commas s-semicolons t-tabs tf-token file)
Press Enter to exit, keeping current data file, if any.
1: c

Current data file is: C:\Users\Lynn\Users\r data files\ds2.csv

Current data structure:
r (factor with 2 values): non-rhotic rhotic
store (factor with 3 values): Saks Macy's Klein's
emphasis (factor with 2 values): normal emphatic
word (factor with 2 values): fouRthflooR
```

Commas separate the data in a .csv file so choose option "c"

Response/dependant variable is presence of absence of post-vocalic r so we have to variants, "rhotic" and "non-rhotic". Notice that I can use strings to code these variants (i.e. no need to create obscure coding systems)

Predictor/independent variables in this study were department store, emphasis and word.

Before running a statistical analysis, I find it very useful to simply 'eyeball' the data and make sure that there are enough tokens filling each cell. To do this in Rbrul, you can use the crosstabs function on the main menu (no. 4) and cross-tabulate your response variable with each of your independent variables in turn.

```
MAIN MENU
1-load/save data 2-adjust data
4-crosstabs 5-modeling 6-plotting
8-restore data 9-reset 0-exit
1: 4
Cross-tab: factor for columns?  (1-r 2-store 3-emphasis 4-word)
1: 1
Cross-tab: factor for rows? (2-store 3-emphasis 4-word Enter-none)
1: 2
Cross-tab: factor for 'pages'? (3-emphasis 4-word Enter-none)
1: 3
Cross-tab: variable for cells? (1-response proportion/mean Enter-counts)
1:
counts
, , emphasis = emphatic

         r
store     non-rhotic rhotic total
  Klein's         73     13    86
  Macy's          68     44   112
  Saks            36     37    73
  total          177     94   271

, , emphasis = normal

         r
store     non-rhotic rhotic total
  Klein's        122      8   130
  Macy's         143     81   224
  Saks            57     47   104
  total          322    136   458

, , emphasis = total

         r
store     non-rhotic rhotic total
  Klein's        195     21   216
  Macy's         211    125   336
  Saks            93     84   177
```

These are the actual token numbers in the data set for each variant of the dependant variable cross-tabulated with store and then by emphasis

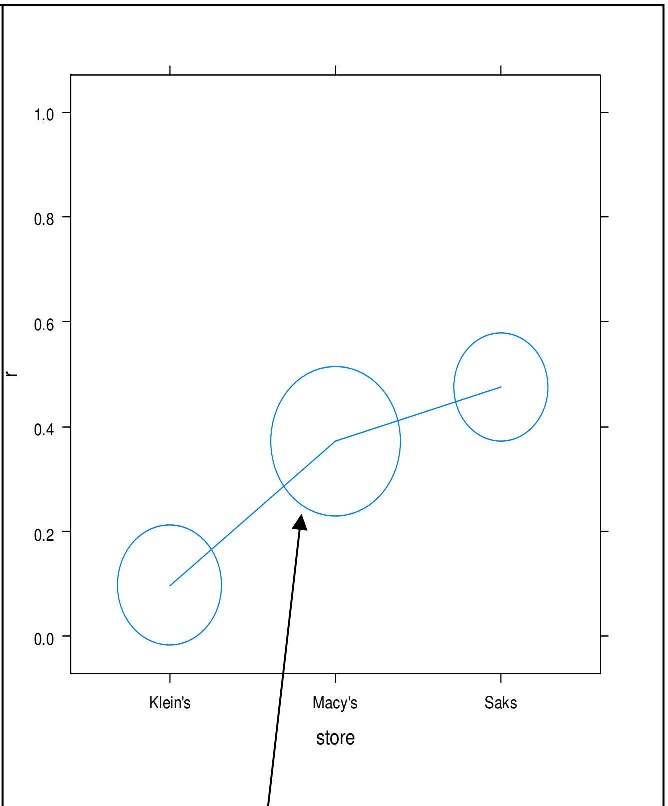
These are the total token numbers in the data set for each variant of the dependant variable cross-tabulated with store

TIP: these are raw token numbers but to get percentages (and so get a better idea of underlying patterns in the data, when prompted "variable for cells?", choose "1 – response proportion/mean")

Some of these counts are quite small but none are empty so that's a good start! [NB: Rbrul will still run with empty cells (unlike Goldvarb) but it's questionable whether the results will be reliable (empty cells imply no variation!)

Another useful function of Rbrul is that you can easily plot your data to see if there are any visible underlying patterns before you run the regression. You can do this using the plot function on the main menu (number 6).

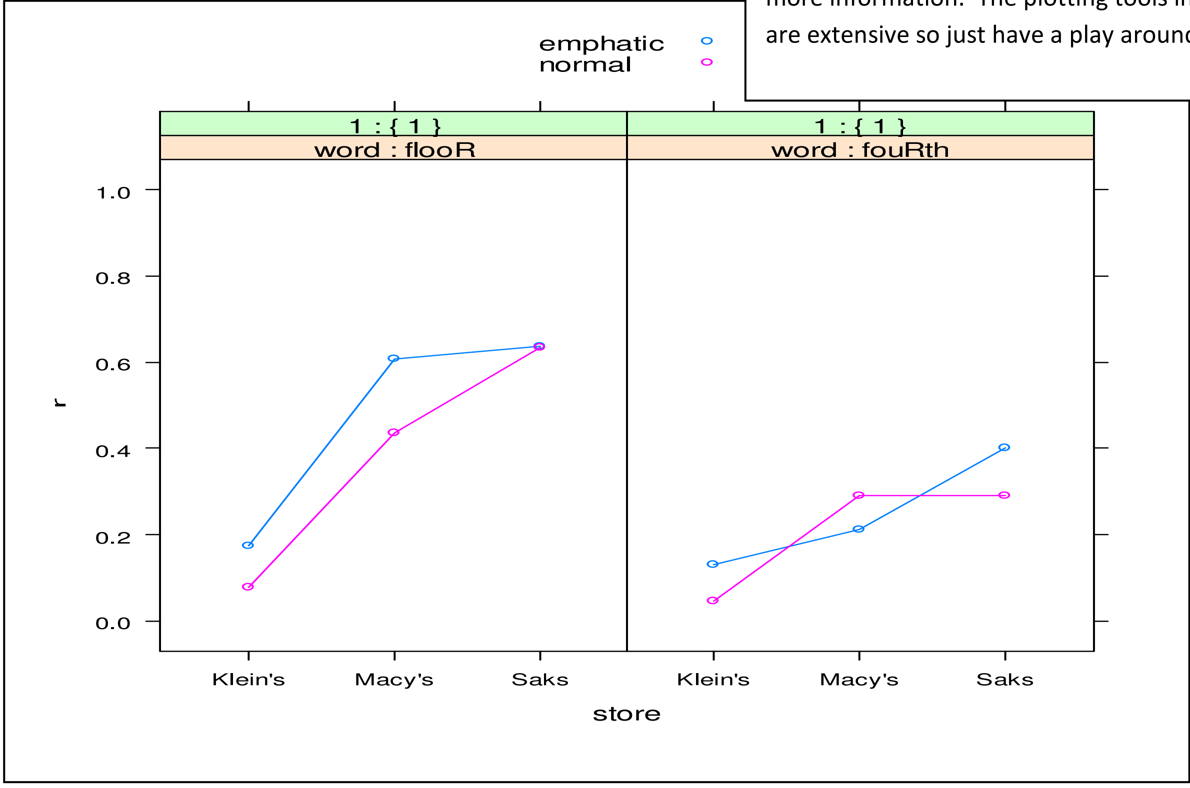
```
PLOTTING MENU
1-custom scatterplot
5-modeling 7-save plot 9-main menu 0-exit
1: 1

Current data structure:
r (factor with 2 values): non-rhotic rhotic
store (factor with 3 values): Saks Macy's Klein's
emphasis (factor with 2 values): normal emphatic
word (factor with 2 values): fouRth flooR

Data variables: 1-r 2-store 3-emphasis 4-word

No model loaded.

Choose variable for y-axis?
1: 1
Choose variable for x-axis?
1: 2
Separate (and color) data according to the values of which variable? (press
Enter to skip)
1:
Split data into horizontal panels according to which variable? (press Enter to
skip)
1:
Split data into vertical panels according to which variable? (press Enter to
skip)
1:
Type of points to plot? (raw points not recommended for binary data)
(0-no points 1-raw points Enter-mean points)
1:
Scale points according to the number of observations?
Enter size factor between 0.1 and 10 (1 = Enter = default)
or 0 to not scale points
1:
Type of lines to plot (raw lines not recommended for binary data)?
0-no lines 1-raw lines Enter-mean lines)
1:
Add a reference line? (1-diagonal [y=x] 2-horizontal [y=0] Enter-none)
1: |
```



This graph has scaled the points according to the number of observations (so we have more data from Macy's)

```
PLOTTING MENU
1-custom scatterplot
5-modeling 7-save plot 9-main menu 0-exit
1: 1

Current data structure:
r (factor with 2 values): non-rhotic rhotic
store (factor with 3 values): Saks Macy's Klein's
emphasis (factor with 2 values): normal emphatic
word (factor with 2 values): fouRth flooR

Data variables: 1-r 2-store 3-emphasis 4-word

No model loaded.

Choose variable for y-axis?
1: 1
Choose variable for x-axis?
1: 2
Separate (and color) data according to the values of which variable? (press
Enter to skip)
1: 3
Also show data (in black) averaged over all values of
emphasis? (1-yes Enter-no)
1:
Split data into horizontal panels according to which variable? (press Enter to
skip)
1: 4
Split data into vertical panels according to which variable? (press Enter to
skip)
1:
Type of points to plot? (raw points not recommended for binary data)
(0-no points 1-raw points Enter-mean points)
1:
Scale points according to the number of observations?
Enter size factor between 0.1 and 10 (1 = Enter = default)
or 0 to not scale points
1: 0
Type of lines to plot (raw lines not recommended for binary data)?
0-no lines 1-raw lines Enter-mean lines)
1:
```

This graph is a little more complicated and shows more information. The plotting tools in Rbul (and R) are extensive so just have a play around with them!



3. Running a (fixed effect) logistic regression analysis

It looks like all of these predictor variables could be having an effect on our response/dependent variable of rhoticity. In a simple data set like this where our response variable is binary and our predictor variables are categorical, a logistic regression analysis can help us to model the extent to which our predictor variables are influencing variation in our response variable. **Logistic regression** is well-suited to the type of data we usually have in sociolinguistics because it is a method that is nonparametric - it doesn't require equal variance in the cells of a model, and doesn't require that the data be normally distributed (K. Johnson 2009). A simple logistic regression of this sort will tell us (a) how much variation there is in our data set, (b) how much variation our predictor variables account for and (c) the effect size of each predictor variant.

Before you run a model like this (number 5, modelling), Rbrul will first ask you which variables you want to include in the regression [handy if you don't want to include all at once].

```
MAIN MENU
1-load/save data 2-adjust data
4-crosstabs 5-modeling 6-plotting
8-restore data 9-reset 0-exit
1: 5

No variables chosen.

MODELING MENU
1-choose variables 2-one-level 3-step-up 4-step-down 5-step-up/step-down
6-plotting 8-settings 9-main menu 0-exit
10-chi-square test
1: 1
Choose response (dependent variable) by number (1-r 2-store 3-emphasis 4-word)
1: 1
Type of response? (1-continuous Enter-binary)
1:
Choose application value(s) by number?  (1-non-rhotic 2-rhotic)
1: 2
Choose predictors (independent variables) by number (2-store 3-emphasis 4-word)
1: 2
2: 3
3: 4
Are any predictors continuous? (2-store 3-emphasis 4-word Enter-none)
1:
Any grouping factors (random effects)? (2-store 3-emphasis 4-word Enter-none)
1:
Consider a(nother) pairwise interaction between predictors? Choose two at a time. (2-store 3-emphasis 4-word Enter-done)
1:

Current variables are:
response.binary: r (rhotic vs. non-rhotic)
fixed.factor: store emphasis word
```

To start modelling, you need to select no. 5

Choose your variables

Your response/dependent variable is rhoticity (the column label is "r")

The new feature is rhoticity in NYC so select "2" as your application value (i.e. the thing you are interested in)

No continuous predictors or random effects here (more on this later)

We'll ignore interaction effects for the moment too

Now that you've defined your variables, you're ready to run the analysis. There are 4 options available to you now – you can run a one level analysis, a step up analysis, a step down analysis or a step-up/step down analysis. When you're staring out, it's a good idea to run a step up/step down analysis because you can see the individual stages of the model-build and if there are any weird stages (e.g. if you don't have enough data, Rbrul will say 'error'; you'll miss this stage out in a one-level analysis and jump straight to the output)...so here goes:

Select modelling, then step up/step down.

Rbrul will then run the step up analysis followed by the step down analysis and (hopefully) they should match!

```
BEST STEP-UP MODEL WAS WITH store (1.08e-18) + word (8.18e-09) [A]

STEP-UP AND STEP-DOWN MATCH!

STEPPING DOWN:

$store
  factor logodds tokens rhotic/rhotic+non-rhotic centered factor weight
    Saks   0.900    177                    0.475                  0.711
  Macy's   0.436    336                    0.372                  0.607
 Klein's  -1.337    216                    0.097                  0.208

$word
 factor logodds tokens rhotic/rhotic+non-rhotic centered factor weight
  flooR   0.493    347                    0.412                  0.621
 fouRth  -0.493    382                    0.228                  0.379

$misc
 deviance df intercept grand mean centered input prob Nagelkerke R2
  793.002  4     -0.97      0.316                0.275         0.206

Current variables are:
response.binary: r (rhotic vs. non-rhotic)
fixed.factor: store emphasis word
```

These figures are the p values associated with adding each of these predictors to the model (v. Small p values hence very significant effect)

Logodds are raw co-efficients for the regression model. The range from negative infinity to positive infinity and the larger the number, the bigger the effect size

Degrees of freedom

Factor weights are the Goldvarb-style co-efficients which takes the logodds (- to + infinity) and converts them to a range of 0 – 1). FWs above 0.5 favour the application value (in this case, rhoticity) and the closer to 1, the bigger the effect). The only difference here is that Rbrul uses centered FWs and Goldvarb uses uncentered FWs (uncentered FWs are dependent on cell size so it's not advisable to use these)

The `deviance` is a measure of how well the model fits the data, or how much the actual data deviates from the predictions of the model. The larger the deviance, the worse the fit. As we add terms (predictors) to the model, we will see this number decrease.

Grand mean = proportion of response application value as total of all data (so in this model 31.6% of the tokens are of rhotic forms)

Rbrul reports the R-squared for linear models (continuous responses) as `R2` and a pseudo-R-squared for logistic models (binary responses) as `Nagelkerke.R2`. The R-squared or pseudo-R-squared represents the proportion of variation explained by the model(s).

How to report these results? I tend to use a table format and show something like this...

Deviance				793.002
df				4
Grand mean				0.316
Factors	Log Odds	Tokens (N)	Proportion of application value [rhoticity]	Uncentered weight
STORE				
Saks	0.900	177	0.475	0.711
Macy's	0.436	336	0.372	0.607
Klein's	-1.337	216	0.097	0.208
WORD				
flooR	0.493	347	0.412	0.621
fouRth	-0.493	382	0.228	0.379

NB: if you're not presenting to a sociolinguistics audience, probably best not to show the factor weights (they're only there so that people previously familiar with Goldvarb would be able to compare across studies easily).

4. Running a mixed effect logistic regression analysis

The previous example worked only with a very small number of predictor variable, all of which were categorical. But what if you have some variables which are measured on a continuous scale (e.g. lexical frequency or formant measurements)? Rbrul can handle these too. It can also, to some extent, test for interactions between predictor variables (i.e. situations where the predictor variables are not independent of each other but pattern in a similar way). And it can handle random predictor variables (i.e. predictor variables which are usually not replicable but are expected to randomly vary in some unique way such as the individual speaker or individual word in a particular study...more on this later).

Load the data file called t-to-r_archiveliv_rbrulwkshop.

```
What separates the columns in the data file to open?
(c-commas s-semicolons t-tabs tf-token file)
Press Enter to exit, keeping current data file, if any.
1: c

Current data file is: F:\Liverpool project folder 29.11.10\scouse DATA 20.10.10\t-to-r\ALL DATA\RBRUL analysis\t-to-r_archiveliv_rbrulwkshop$

Current data structure:
speaker (factor with 8 values): LIV_ArchiveM01 LIV_ArchiveF07 LIV_ArchiveM04 LIV_ArchiveM10 LIV_ArchiveF03 ...
time (factor with 539 values): 10:25.3 13:44.7 19:53.2 17:14.2 33:22.1 ...
context (factor with 533 values): a dentist had been at a set of teeth there were houses missing you know .. at a fellow and knocked him out$
t.to.r (factor with 5 values): tapped r t d glottal stop approximant r
preceding.phon (factor with 6 values): TRAP Schwa KIT FOOT LOT ...
following.phon (factor with 15 values): TRAP START Schwa LOT THOUGHT ...
word (factor with 12 values): at bit but get got ...
word.grammatical.category (factor with 13 values): AT (preposition) BIT (noun) but (conjunction) but (conjunction/discourse particle) GET (v$
word.frequency.raw.in.BNC (integer with 12 values): 4000 1253 6499 5275 7271 ...
word.frequency.log (numeric with 11 values): 3.6 3.1 3.81 3.72 3.86 ...
gender (factor with 2 values): male female
year.of.birth (integer with 8 values): 1925 1930 1935 1900 1919 ...

MAIN MENU
1-load/save data 2-adjust data
4-crosstabs 5-modeling 6-plotting
8-restore data 9-reset 0-exit
```

Time and **context** are irrelevant not predictor variables but are perhaps useful for other reasons so we'll keep them here for now.

The first thing to notice is that the predictor variable (here labelled t.to.r) has 5 values. A regression analysis such as this will only work is the predictor variable is (a) binary or (b) continuous. To make the variable binary, you can run separate regressions for each variant modelled against the rest (e.g. tapped r vs. the rest; t vs. the rest etc.) In our data, however, it became clear when coding the data that tapped and approximant r were restricted to certain phonological environments but the other variants weren't so we decided to collapse t,d and glottal stop as T and tapped & approximant r as R. To do this select 2 from the main menu (adjust data) and follow the instructions...

```
MAIN MENU
1-load/save data 2-adjust data
4-crosstabs 5-modeling 6-plotting
8-restore data 9-reset 0-exit
1: 2

ADJUSTING MENU
1-change class 2-rename 3-exclude 4-retain 5-recode
6-relevel 7-center/transform 8-count 9-main menu 0-exit
10-make interaction group
1: 5
Factor group to recode? (press Enter to exit) (1-speaker 2-time 3-context 4-t.to.r 5-preceding.phon 6-following.phon 7-word 8-word.grammatic$
1: 4
Factor(s) of t.to.r to recode together? (1-approximant r 2-d 3-glottal stop 4-t 5-tapped r Enter-done)
1: 1
2: 5
3:
Recode approximant r tapped r as what?
1: R
Factor(s) of t.to.r to recode together? (1-R 2-d 3-glottal stop 4-t 5-R Enter-done)
1: 2
2: 3
3: 4
4:
Recode d glottal stop t as what?
1: T
Factor(s) of t.to.r to recode together? (1-R 2-T 3-T 4-T 5-R Enter-done)
1:
Recode to new column? (Yes-type new column name No-press Enter)
1:
```

We also have some other adjustments to make to the data before we can proceed. We have to continuous variables in the data this time – log word frequency (ignore the raw data) and year of birth. It's useful to manually change these to continuous variables because sometimes Rbrul thinks they're factors and it tries to run them as such (taking AGES!).

```
ADJUSTING MENU
1-change class 2-rename 3-exclude 4-retain 5-recode
6-relevel 7-center/transform 8-count 9-main menu 0-exit
10-make interaction group
1: 1

Current data structure:
speaker (factor with 8 values): LIV_ArchiveM01 LIV_ArchiveF07 LIV_ArchiveM04 LIV_ArchiveM10 LIV_ArchiveF03 ...
time (factor with 539 values): 10:25.3 13:44.7 19:53.2 17:14.2 33:22.1 ...
context (factor with 533 values): a dentist had been at a set of teeth there were houses missing you know .. at a fellow and knocked him out$
t.to.r (factor with 2 values): R T
preceding.phon (factor with 6 values): TRAP Schwa KIT FOOT LOT ...
following.phon (factor with 15 values): TRAP START Schwa LOT THOUGHT ...
word (factor with 12 values): at bit but get got ...
word.grammatical.category (factor with 13 values): AT (preposition) BIT (noun) but (conjunction) but (conjunction/discourse particle) GET (v$
word.frequency.raw.in.BNC (integer with 12 values): 4000 1253 6499 5275 7271 ...
word.frequency.log (numeric with 11 values): 3.6 3.1 3.81 3.72 3.86 ...
gender (factor with 2 values): male female
year.of.birth (integer with 8 values): 1925 1930 1935 1900 1919 ...

Change class of which variable? (1-speaker 2-time 3-context 4-t.to.r 5-preceding.phon 6-following.phon 7-word 8-word.grammatical.category 9-$
1: 10
Change word.frequency.log to which class? (f-factor c-continuous [integer/numeric])
1: c
```

In the adjusting menu, select the variable you want to change, then select "c" for continuous. Do this for all continuous variables in the data set.

I usually start with a fixed effect model and build up the complexity of the model as I go. So model the t-to-r data in the way described above using only the fixed effect predictors (preceding phon, following phon, word/grammatical category & gender). You should get something like this:

```
BEST STEP-UP MODEL WAS WITH gender (2.76e-28) + word.grammatical.category (1.26e-31) + preceding.phon (0.00246) + following.phon (0.0456) [A]

STEP-UP AND STEP-DOWN MATCH!

STEPPING DOWN:

$preceding.phon
 factor logodds tokens R/R+T centered factor weight
   TRAP   2.616     52 0.538                   0.932
    LOT   0.435    192 0.651                   0.607
  Schwa   0.087     54 0.315                   0.522
   FOOT  -0.115    143 0.490                   0.471
  DRESS  -1.168     67 0.716                   0.237
    KIT  -1.855     98 0.276                   0.135

$following.phon
  factor logodds tokens R/R+T centered factor weight
    FACE  13.549      3 1.000                  >0.999
   FORCE  13.463      1 1.000                  >0.999
  FLEECE   0.988     17 0.765                   0.729
    TRAP  -0.180     56 0.750                   0.455
   MOUTH  -0.544     17 0.529                   0.367
   PRICE  -0.671     27 0.630                   0.338
   START  -0.773     34 0.559                   0.316
   Schwa  -0.883    165 0.612                   0.293
    GOAT  -1.087     21 0.571                   0.252
     KIT  -1.113    121 0.529                   0.247
     LOT  -1.291     50 0.260                   0.216
   DRESS  -1.639     63 0.222                   0.163
 THOUGHT  -1.675      9 0.111                   0.158
    FOOT  -1.860     20 0.300                   0.135
   NURSE -16.282      2 0.000                  <0.001

$word.grammatical.category
                                  factor logodds tokens R/R+T centered factor weight
                              BIT (noun)   2.651     18 0.833                   0.934
                              GET (verb)   2.182     63 0.746                   0.899
                              LET (verb)   1.690      7 0.429                   0.844
                       but (conjunction)   1.159     93 0.624                   0.761
                              PUT (verb)   0.628     31 0.581                   0.652
                              GOT (verb)   0.440     75 0.653                   0.608
           WHAT (pronoun/deteriner(wh))   0.350     46 0.696                   0.587
                            NOT (adverb)   0.339     25 0.720                   0.584
                              LOT (noun)  -0.256     49 0.571                   0.436
                            IT (pronoun)  -0.477     83 0.157                   0.383
                         THAT (conj+det)  -2.785     43 0.465                   0.058
 but (conjunction/discourse particle)  -2.882     37 0.081                   0.053
                       AT (preposition)  -3.037     36 0.306                   0.046

$gender
 factor logodds tokens R/R+T centered factor weight
   male   1.473    363 0.700                   0.814
 female  -1.473    243 0.251                   0.186

$misc
 deviance df intercept grand mean centered input prob Nagelkerke R2
  497.354 33       0.8       0.52                0.69          0.575


Current variables are:
response.binary: t.to.r (R vs. T)
fixed.factor: preceding.phon following.phon word.grammatical.category gender
```

NOTE: don't just accept the output without looking very carefully at it. The values for following phon are weird – why? FACE, FORCE & NURSE have very low token numbers and no variation in the cell which massively skews the rest of the data. Goldvarb would not allow a regression with data like this to proceed; Rbrul will but you need to be cautious.

In cases like this, I remove the offending cells (because there's no point having cells with no variation in an analysis of variation!).

To remove these cells, return to the main menu, select adjust data, exclude then exclude the numbers corresponding to FACE, FORCE and NURSE in the following phon category.

Re-run the basic analysis and you should now see something like this:

```
BEST STEP-UP MODEL WAS WITH gender (8.93e-28) + word.grammatical.category (1.02e-31) + preceding.phon (0.00388) [A]

STEP-UP AND STEP-DOWN MATCH!

STEPPING DOWN:

$preceding.phon
 factor logodds tokens R/R+T centered factor weight
   TRAP   2.705     47 0.511                   0.937
    LOT   0.414    192 0.651                   0.602
  Schwa   0.100     54 0.315                   0.525
   FOOT  -0.032    142 0.493                   0.492
  DRESS  -1.118     67 0.716                   0.246
    KIT  -2.068     98 0.276                   0.112

$word.grammatical.category
                                  factor logodds tokens R/R+T centered factor weight
                              BIT (noun)   2.930     18 0.833                   0.949
                              GET (verb)   2.109     63 0.746                   0.892
                              LET (verb)   1.659      7 0.429                    0.84
                       but (conjunction)   1.261     93 0.624                   0.779
                            NOT (adverb)   0.694     25 0.720                   0.667
           WHAT (pronoun/deteriner(wh))   0.568     46 0.696                   0.638
                              GOT (verb)   0.356     75 0.653                   0.588
                              PUT (verb)   0.253     31 0.581                   0.563
                              LOT (noun)  -0.167     49 0.571                   0.458
                            IT (pronoun)  -0.374     83 0.157                   0.408
                         THAT (conj+det)  -2.710     39 0.436                   0.062
                       AT (preposition)  -3.162     35 0.286                   0.041
 but (conjunction/discourse particle)  -3.416     36 0.083                   0.032

$gender
 factor logodds tokens R/R+T centered factor weight
   male   1.462    358 0.698                   0.812
 female  -1.462    242 0.252                   0.188
```

Following phon is no longer included in the model.

The word/grammatical category FG came about because we noticed that the word BUT behaves differently when used as a conjunction and when used as a discourse particle (usually in the filler "but er…"). There is very little different between the FGs word and word/grammatical category except that word/grammatical category is more descriptive. Because they are so similar, it's unwise to include both in the same regression so we'll stick with the more detailed word/grammatical category for the moment.

Next, let’s try including the continuous predictors in the model. Include the previous significant predictors but this time also include log word frequency and year of birth.

```
BEST STEP-UP MODEL WAS WITH gender (8.93e-28) + word.grammatical.category (1.02e-31) + preceding.phon (0.00388) + year.of.birth (0.00653) [A]

STEP-UP AND STEP-DOWN MATCH!

STEPPING DOWN:

$preceding.phon
 factor logodds tokens R/R+T centered factor weight
   TRAP   2.680     47 0.511                  0.936
    LOT   0.405    192 0.651                    0.6
  Schwa  -0.045     54 0.315                  0.489
   FOOT  -0.133    142 0.493                  0.467
  DRESS  -1.040     67 0.716                  0.261
    KIT  -1.867     98 0.276                  0.134

$word.grammatical.category
                              factor logodds tokens R/R+T centered factor weight
                          BIT (noun)   2.824     18 0.833                  0.944
                          GET (verb)   2.008     63 0.746                  0.882
                          LET (verb)   1.622      7 0.429                  0.835
                   but (conjunction)   1.336     93 0.624                  0.792
                        NOT (adverb)   0.733     25 0.720                  0.675
         WHAT (pronoun/deteriner(wh))  0.579     46 0.696                  0.641
                          PUT (verb)   0.360     31 0.581                  0.589
                          GOT (verb)   0.332     75 0.653                  0.582
                          LOT (noun)  -0.138     49 0.571                  0.466
                        IT (pronoun)  -0.558     83 0.157                  0.364
                     THAT (conj+det)  -2.554     39 0.436                  0.072
                    AT (preposition)  -3.004     35 0.286                  0.047
 but (conjunction/discourse particle) -3.540     36 0.083                  0.028

$gender
 factor logodds tokens R/R+T centered factor weight
   male   1.434    358 0.698                  0.808
 female  -1.434    242 0.252                  0.192

$year.of.birth
 continuous logodds
         +1  -0.024

$misc
 deviance df intercept grand mean Nagelkerke R2
  508.749 20    46.204      0.518         0.554


Current variables are:
response.binary: t.to.r (R vs. T)
fixed.factor: preceding.phon word.grammatical.category gender
fixed.continuous: word.frequency.log year.of.birth
```

The continuous variable **year of birth** is return as significant. Notice that there are no factor weights for continuous predictors (which are not factors); instead we get a single regression co-efficient. In this case, the value is a negative which suggests a negative correlation between frequency of R and year of birth (as year of birth increases, frequency of t-to-r decreases). With a much larger data set, this could indicate change in progress but here the range contained in year of birth is very small (only a generation) – it’s included here simply as a way of showing how continuous predictors are returned as significant effects in the model.

So far, we’ve been treating word/grammatical category as a fixed effect. However, a variable should be treated as ***random*** if we can think of the levels that we observe as being drawn from a larger population (and not one defined by the analyst). In linguistics, individual speaker and individual word are often considered random effects because the data set that we use represents a much larger random sample of people and words. We would expect some unpredictable ‘noise’ in the system from these variables because we expect them to behave (to a certain extent) randomly – “Including a speaker random effect takes into account that some individuals might favor a linguistic outcome while others might disfavor it, over and above (or ‘under and below’) what their gender, age, social class, etc. would predict.” (Johnson 2009: 365). In models such as this, if we code random effects as fixed effects (as we may have done here), we risk committing a Type I error i.e. we can end up observing a significant difference when in fact there is none or at least none that couldn’t be accounted for by random variation). Let’s re-run the model, this time including word/grammatical category and individual speaker as random effects in the model:

Only 2 significant p values now (preceding phon & gender) because you don't get a p value for a random effect.

```
BEST STEP-UP MODEL WAS WITH speaker (random) + word.grammatical.category (random) + preceding.phon (0.00465) + gender (0.0302) [A]

STEP-UP AND STEP-DOWN MATCH!

STEPPING DOWN:

$preceding.phon
 factor logodds tokens R/R+T centered factor weight
   TRAP   1.759     47 0.511                  0.853
    LOT   0.727    192 0.651                  0.674
  DRESS   0.507     67 0.716                  0.624
  Schwa  -0.582     54 0.315                  0.358
   FOOT  -1.064    142 0.493                  0.257
    KIT  -1.348     98 0.276                  0.206

$gender
 factor logodds tokens R/R+T centered factor weight
   male   1.552    358 0.698                  0.825
 female  -1.552    242 0.252                  0.175

$word.grammatical.category
                               random logodds tokens R/R+T centered factor weight std dev
                    but (conjunction)   2.158     93 0.624                  0.896   1.589
                           BIT (noun)   2.048     18 0.833                  0.885   1.589
                           PUT (verb)   1.537     31 0.581                  0.822   1.589
                           GET (verb)   0.930     63 0.746                  0.716   1.589
          WHAT (pronoun/deteriner(wh))  0.337     46 0.696                  0.582   1.589
                           GOT (verb)   0.216     75 0.653                  0.552   1.589
                         NOT (adverb)   0.199     25 0.720                  0.548   1.589
                           LOT (noun)  -0.033     49 0.571                   0.49   1.589
                           LET (verb)  -0.149      7 0.429                  0.461   1.589
                         IT (pronoun)  -1.103     83 0.157                  0.248   1.589
                      THAT (conj+det)  -1.782     39 0.436                  0.143   1.589
 but (conjunction/discourse particle)  -2.106     36 0.083                  0.108   1.589
                      AT (preposition) -2.175     35 0.286                  0.101   1.589
```

```
$speaker
         random logodds tokens R/R+T centered factor weight std dev
 LIV_ArchiveF05   2.932     52 0.654                  0.949   1.689
 LIV_ArchiveF01   1.266     52 0.423                  0.778   1.689
 LIV_ArchiveM01   0.329     95 0.821                  0.579   1.689
 LIV_ArchiveM04   0.219     83 0.639                  0.552   1.689
 LIV_ArchiveM10  -0.004    126 0.714                  0.496   1.689
 LIV_ArchiveM07  -0.818     54 0.537                  0.304   1.689
 LIV_ArchiveF07  -1.545     61 0.049                  0.174   1.689
 LIV_ArchiveF03  -2.292     77 0.026                  0.091   1.689

$misc
 deviance df intercept grand mean centered input prob
  457.036  9    -0.353      0.518                0.413


Current variables are:
response.binary: t.to.r (R vs. T)
fixed.factor: preceding.phon gender
fixed.continuous: word.frequency.log year.of.birth
random.intercept: speaker word.grammatical.category
```

Including speaker as a random effect has eliminated year of birth which means that all of the variation accounted for by year of birth can be accounted for by simple individual speaker variation

The default setting in Rbrul is to show estimates of the individual effect for each variant in the random effects. The Rbrul manual has this to say: "these numbers resemble and are comparable with the fixed effect coefficients, although in a technical sense they are not parameters of the model in the same way". If you're not especially interested in the behaviour of the random effects but you just want a way of taking the variation of that group into account, you can change the settings to hide these coefficients (see below).

6. Changing the Settings in Rbrul

```
MODELING MENU
1-choose variables 2-one-level 3-step-up 4-step-down 5-step-up/step-down
6-plotting 8-settings 9-main menu 0-exit
10-chi-square test
1: 8
Run silently? (1-yes Enter-no)
1:
Hide intermediate models' details? (1-yes Enter-no)
1:
Hide factor weights? (1-yes Enter-no)
1:
Center factors [recommended]? (1-no Enter-yes)
1:
Show random effect estimates (BLUPs)? (1-no Enter-yes)
1: 1
Threshold p-value for fixed effect significance? (1-automatic, Enter-0.05, or type another value)
1:
Use slow but more accurate [?] simulation method for fixed effect significance? (1-yes Enter-no)
1:
Number of significant digits/decimal places to display? (Enter-3)
1:
Type of contrasts to use for factors? (1-treatment Enter-sum)
1:
```

By selecting 'no', all of the 'working' (i.e. tables of coefficients) produced in the step up/down analysis are hidden

This means that Rbrul will only show the details of the best model from each step-up or step-down run.

If you're not interested in Goldvarb-style factor weights, they can be hidden

Random estimates can be hidden (see new output below)

Change P value sig threshold (e.g. from 0.05 to 0.01)

From Rbrul manual...
"Sum contrasts operate similarly to (centered) factor weights; for any predictor, they are centered around zero. For example, in one of the department store models above, we saw `emphatic: 0.115` and `normal: -0.115`; values for `store` and `word` also summed to zero. Treatment contrasts appear quite different, although they are really just a different way of conveying the same information about the different effects of factors on a response variable. With treatment contrasts, one level of each factor (one factor in each factor group) is chosen as the baseline. The effects of the other factors are expressed in terms of their difference from the baseline. So if `normal` was the baseline level of `emphasis`, it would appear with a coefficient of `0.000` while `emphatic` would appear with `0.330`...Note that using treatment contrasts...will not affect the output in the factor weights column. Rbrul allows us, as sociolinguists, to have our cake and eat it too."

```
STEPPING DOWN:

$preceding.phon
 factor logodds tokens R/R+T centered factor weight
   TRAP   1.759     47 0.511                  0.853
    LOT   0.727    192 0.651                  0.674
  DRESS   0.507     67 0.716                  0.624
  Schwa  -0.582     54 0.315                  0.358
   FOOT  -1.064    142 0.493                  0.257
    KIT  -1.348     98 0.276                  0.206

$gender
 factor logodds tokens R/R+T centered factor weight
   male   1.552    358 0.698                  0.825
 female  -1.552    242 0.252                  0.175

$word.grammatical.category
 random std..dev
          1.589

$speaker
 random std..dev
          1.689

$misc
 deviance df intercept grand mean centered input prob
  457.036  9    -0.353      0.518                0.413
```

Above model with random effect estimates hidden and only standard deviation shown.

7. Testing for interactions in Rbrul

One final thing that we should do before the model is complete is test for interactions. Interaction effects arise from a situation where the influence of one independent variable is dependent on the influence of another. A nice real world example (from Wikipedia!) is an intuitive interaction between adding sugar to coffee and stirring the coffee. Neither of the two individual variables has much effect on sweetness but a combination of the two does. NOTE: Interactions between independent variables should not be confused with multicollinearity, which is when substantial correlations exist between two or more of the independent variables in a regression (e.g. the two methods of coding 'word' in the above regression were almost identical and so were collinear). It is only possible to test for interaction effects between categorical independent variables (in Rbrul...not sure about elsewhere). The only two categorical predictor models left are gender and preceding phonological environment so let's test for an interaction effect here and see what happens:

```
MODELING MENU
1-choose variables 2-one-level 3-step-up 4-step-down 5-step-up/step-down
6-plotting 8-settings 9-main menu 0-exit
10-chi-square test
1: 1
Choose response (dependent variable) by number, or Enter to keep t.to.r (1-speaker 2-time 3-context 4-t.to.r 5-preceding.phon 6-following.phon 7-word 8-w$
1:
Choose predictors (independent variables) by number, or Enter to keep speaker & preceding.phon & word.grammatical.category & word.frequency.log & gender $
1: 1
2: 5
3: 8
4: 10
5: 11
6:
Are any predictors continuous? (1-speaker 5-preceding.phon 8-word.grammatical.category 10-word.frequency.log 11-gender Enter-none)
1: 10
2:
Any grouping factors (random effects)? (1-speaker 5-preceding.phon 8-word.grammatical.category 11-gender Enter-none)
1: 1
2: 8
3:
Consider a(nother) pairwise interaction between predictors? Choose two at a time. (1-speaker 5-preceding.phon 8-word.grammatical.category 10-word.frequen$
1: 5
2: 11
Consider a(nother) pairwise interaction between predictors? Choose two at a time. (1-speaker 5-preceding.phon 8-word.grammatical.category 10-word.frequen$
1:

Current variables are:
response.binary: t.to.r (R vs. T)
fixed.factor: preceding.phon gender
fixed.continuous: word.frequency.log
fixed.interaction: preceding.phon:gender
random.intercept: speaker word.grammatical.category
```

Include interaction variables here

```
BEST STEP-UP MODEL WAS WITH speaker (random) + word.grammatical.category (random) + preceding.phon (0.00465) + gender (0.0302) + preceding.phon:gender (0$

STEP-UP AND STEP-DOWN MATCH!

STEPPING DOWN:

$preceding.phon
 factor logodds tokens R/R+T centered factor weight
   TRAP   1.726     47 0.511                 0.849
  DRESS   0.695     67 0.716                 0.667
    LOT   0.636    192 0.651                 0.654
  Schwa  -0.583     54 0.315                 0.358
   FOOT  -0.728    142 0.493                 0.326
    KIT  -1.746     98 0.276                 0.149

$gender
 factor logodds tokens R/R+T centered factor weight
   male    1.53    358 0.698                 0.822
 female   -1.53    242 0.252                 0.178

$`preceding.phon:gender`
 factor:factor logodds tokens R/R+T centered factor weight
      KIT:male   0.712     63 0.413                 0.671
   FOOT:female   0.655     60 0.367                 0.658
      LOT:male   0.498    117 0.915                 0.622
  Schwa:female   0.484     31 0.323                 0.619
   TRAP:female   0.384     19 0.211                 0.595
    DRESS:male   0.313     45 0.933                 0.578
  DRESS:female  -0.313     22 0.273                 0.422
     TRAP:male  -0.384     28 0.714                 0.405
    Schwa:male  -0.484     23 0.304                 0.381
    LOT:female  -0.498     75 0.240                 0.378
     FOOT:male  -0.655     82 0.585                 0.342
    KIT:female  -0.712     35 0.029                 0.329

$word.grammatical.category
 random std..dev
          1.459

$speaker
 random std..dev
          1.735

$misc
 deviance df intercept grand mean centered input prob
  443.985 14      -0.4      0.518                0.401
```

It looks like we also have an interaction effect for preceding phon/gender. For preceding TRAP and DRESS vowels, gender doesn't seem to be a relevant factor (the factor weights for males & females hover around 0.5 mark). However for preceding KIT, FOOT, LOT & schwa vowels, these seem to behave differently according to gender. A preceding schwa and preceding FOOT vowel favours R among the females (and disfavours R among the men). A preceding KIT & LOT vowel favours R among the men (and disfavours R among the women). This could be indicative of something else going on with these vowels that is socially meaningful in this community.

It might not be immediately clear whether the difference between modal A (e.g. model without interaction effects) is better than model B (e.g. model with interaction effects). You can test this very simply in Rbrul using the chi square test on the main menu. Select chi square test, input the deviance value for each model then input the difference in degrees of freedom for each model and the output will give you a P value which will tell you if the difference between the models is significant (i.e. whether model A is significantly different, and so better, than model B). Try this using the deviance and df values from the two models above (with and without the interaction effect included). If P is less than or equal to 0.05, the difference between the models is significant and shouldn't be ignored.

```
MODELING MENU
1-choose variables 2-one-level 3-step-up 4-step-down 5-step-up/step-down
6-plotting 8-settings 9-main menu 0-exit
10-chi-square test
1: 10
Enter first deviance or log likelihood.
1: 457.036
Enter second deviance or log likelihood.
1: 443.985
If these were log likelihood values, press 1. Press Enter if they were deviances.
1:
Enter difference in degrees of freedom.
1: 5
Chi-square = 13.051, df = 5, p = 0.023
```

8. Over to you...

If you have brought along a data set of your own to work on, feel free to do this now. The best way to learn how to use Rbrul in particular (and statistical programs in general) is by trial and error so feel free to play with Rbrul/R and see how it goes. If you have any further questions, please don't hesitate to get in touch. **HAVE FUN!!!**