

FUSION Tools

A Few of Our Favorite Things

October, 2008

A Few of Our Favorite Things

Brief introduction to the following tools available on `snowwhite`.

<u>tool</u>	<u>purpose</u>
<code>pquery</code>	easy access to mySQL databases
<code>pocull</code>	add position & (optionally) restrict to a region
<code>metal2zoom.R</code>	“region plot” from metal (or similar) output
<code>(scriptR.pl)</code>	utility for making R scripts executable

pquery – parameterized query

pquery uses a 2-step process to get info from databases:

1. Construct a parameterized query template.

This step can be omitted if there is already a usable template in our library.

2. Provide pquery with the template and “fill in the blanks”.

Filling in the blanks can be done at the command line (useful for scripting) or interactively (useful for a quick look at some data).

Some queries in the current library

query	title
genes_near_snp	Find nearest gene(s) to a given SNP
ld_in_region	Get LD in a region
refFlat_in_region	Get RefFlat Info for a region
t2d_near_gene	Get T2D results near a given Gene
t2d_near_snp	Get T2D results near a given SNP
qt_with_genes	Find nearest gene(s) to top SNPs
snp_pos	Get SNP position and allele info
genes_near_region	Find nearest refSeq gene(s) to a region
qt_in_region	Compare QT p-values across FUSION/DGI/SAR
loris_hot_genes	Get results from T2D and QT meta-analyses
snpset_in_region	Positions for SNPs in SNP set
recomb_in_region	Get Recombination rates in a region

A complete list can be obtained using

```
pquery -list
```

Filling in the blanks – an example

To find what genes are near the SNP rs640742:

```
pquery genes_near_snp
```

```
Title : Find nearest gene(s) to a given SNP
```

```
Author: pchines@mail.nih.gov; rpruim@calvin.edu
```

```
SNP name? rs640742
```

```
Neighborhood for nearest gene search (bp)? [5mb]
```

```
Number of nearest genes to show? [1] 10
```

```
snp      chrom  chrpos strand nearest_gene dist_to_gene direction  txStart  txEnd
rs640742 chr1   20729860 -      DDOST      0          Within    20723565 20733343
rs640742 chr1   20729860 -      KIF17     5953       downstream 20735812 20789623
rs640742 chr1   20729860 +      PINK1     6551       downstream 20705253 20723309
rs640742 chr1   20729860 +      CDA      39156      downstream 20660749 20690704
```

```
⋮
```

Filling in the blanks via command line options

First we find out what the parameters are named.

```
$ pquery -copy genes_near_snp
Title : Find nearest gene(s) to a given SNP
Author: pchines@mail.nih.gov; rpruim@calvin.edu
## Title : Find nearest gene(s) to a given SNP
## Author: pchines@mail.nih.gov; rpruim@calvin.edu
#= Snp      : SNP name
#= Radius: Neighborhood for nearest gene search (bp) [5mb]
#= Limit  : Number of nearest genes to show [1]
```

- Parameters begin with a capital letter (by convention)
- Parameter names are consistent across current templates (Snp, Radius, Gene, Chr, Start, End, Build, ...)
- Even in templates that do not include `Limit`, you can limit the amount of output using `-limit n`.

Filling in the blanks via command line options

Now to find genes that are near the SNP rs640742:

```
$ pquery -defaults Snp=rs640742 genes_near_snp  
Title : Find nearest gene(s) to a given SNP  
Author: pchines@mail.nih.gov; rpruim@calvin.edu
```

```
snp      chrom  chrpos  strand  nearest_gene  dist_to_gene  direction  txStart  txEnd  
rs640742 chr1    20729860  -       -             DDOST        0         Within  2072356520733343
```

- `-defaults` tells `pquery` to use the default values for any parameters not set at the command line.
- Some parameters do not have defaults. Failing to specify them results in an error message.

```
$ pquery -defaults genes_near_snp  
Title : Find nearest gene(s) to a given SNP  
Author: pchines@mail.nih.gov; rpruim@calvin.edu  
No value provided for 'Snp' parameter
```

Creating new templates

Making query templates is fairly easy if you know some SQL.

- Make a copy of a template in the library:

```
| pquery -copy genes_near_snp > myGreatNewTemplate.psql
```

- More information about how pquery processes templates can be obtained via

```
| pquery -man
```

Or you can contact someone else for help . . .

pocull – position and culling tool

pocull was created because many of our output file formats do not include position information. pocull uses information in our SQL tables to add position information and optionally to cull the file to a particular region.

Examples:

```
pocull -reg chr2:12345-23456 all.metal  
pocull -reg rs5400 -flank 50k t1.metal t2.metal  
pocull -reg TCF7L2 -up_gene 25k -down_gene 5k *.metal
```

If no region is specified, position information is added to all rows of the input file.

pocull – position and culling tool

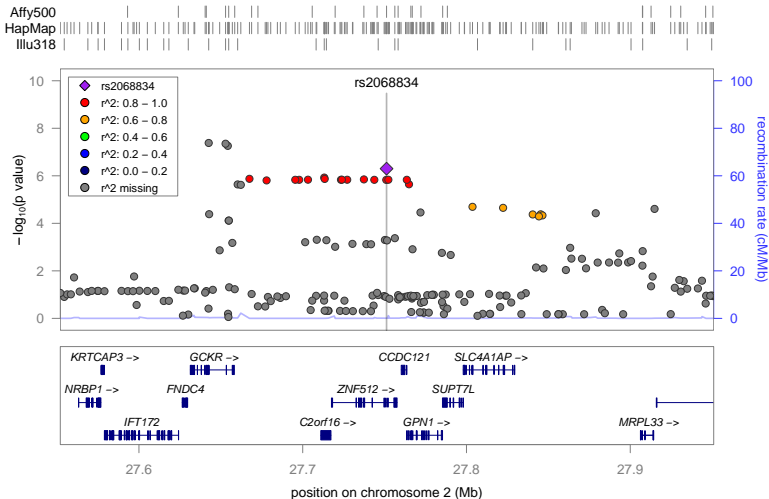
The (small) example below shows how the output compares to the input.

```
head -5 ../../MAGIC/2hr/gluc2hr_nobmi_07312008.metal1.tbl | delim2fixed
MarkerName Allele1 Allele2   Weight Zscore P-value Direction
rs2326918   a       g       13193.48 -1.025  0.3056 --+--+?-
rs2439906   c       g       13193.99 -0.900  0.3679 +-----?-
rs10760160  a       c       13193.71 -0.036  0.9715 -+++++?-
rs977590    a       g       13193.94  2.050  0.04039 -++-++?+
```

```
pocull -reg rs2326918 -flank 200 ../../MAGIC/2hr/gluc2hr_nobmi_07312008.metal1.tbl | delim2fixed
Culling to SNPs in region: chr6:130881584-130881984
chr      pos MarkerName Allele1 Allele2   Weight Zscore P-value Direction
6 130881784 rs2326918   a       g       13193.48 -1.025  0.3056 ---+--?-
6 130881691 rs11751575 a       t       13193.89 -1.054  0.2918 +-----?+
6 130881887 rs946299    t       g       13193.27  1.362  0.1732 +---+--?-
```

metal2zoom.R – plots of small regions

Example Plot



metal2zoom.R – work flow

Information taken from databases (using pquery):

- shallow LD (threshold = .6)
- refFlat info (we mirror UCSC and update monthly)
- SNP sets (HapMap, Affy500, Illu318, etc.)
- recombination rate

User input:

- output from `metal` with position added (or similar)
- any changes to default settings

```
metal2zoom.R metal=metal_2:27.3Mb-30.2Mb.tbl \  
  title="Example Plot" rfrows=3 \  
  refsnp=rs2068834 flank=200kb \  
  \
```

metal2zoom.R – to do list

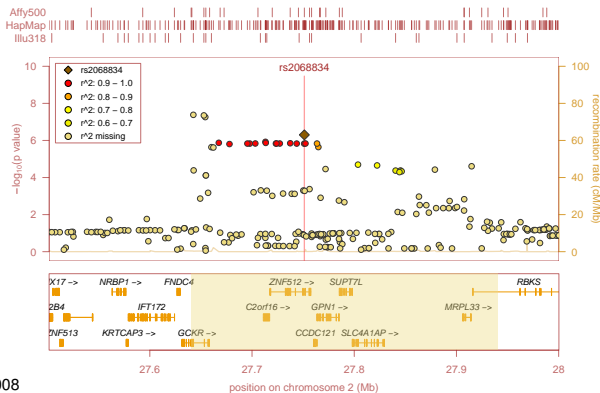
- Improve default settings and user options
- Write wrapper scripts for various batch jobs
- Get better LD into SQL tables
- Automate improving LD
- Documentation

- Apply tools and methods to other applications
 - `scriptR.pl`
 - `scriptR` R package?

Example with thematic color scheme

```
metal2zoom.R metal=metal_2:27.3Mb-30.2Mb.tbl \
  title="Thematic Colors" titleColor="red" \
  rfrows=3 geneColor=orange2 \
  recombColor=orange3 frameAlpha=.7 \
  hiColor=lightgoldenrod hiAlpha=.4 hiStart=27.64Mb hiEnd=27.94Mb \
  refsnp=rs2068834 refsnpTextColor=brown refsnpLineColor=brown1 \
  rugColor=brown rugAlpha=.7 \
  flank=250kb ldcuts=".6,.7,.8,.9,1" \
  ldcol="lightgoldenrod,yellow2,yellow1,orange1,red,orange4"
```

Thematic Colors



Available options for metal2zoom.R

```

ldTable = "results.ld_point6",      # LD Table (for SQL)
recombTable = "results.recomb_rate", # Recomb Rate Table (for SQL)
clean=TRUE,                          # remove temp files?
build = "hg17",                       # build to use for position information
metal = "metal.tbl",                 # metal output file
pval="P.value",                      # name for p-value column in metal output
chr = NULL,                          # chromosome
start = NULL,                       # start of region (string, may include Mb, kb, etc.)
end = NULL,                          # end of region (string, may include Mb, kb, etc.)
flank = "300kb",                    # surround refsnp by this much
refsnp = NULL,                      # snp name (string)
refsnpTextColor = "black",          # color for ref snp label
refsnpTextAlpha = 1,                # alpha for ref snp label
refsnpLineColor = "black",         # color for ref snp line
refsnpLineAlpha = .5,               # alpha for ref snp line
title = "",                          # title for plot
titleColor = "black",              # color for title
thresh = 1,                          # only get pvalues <= thresh
width = 10,                          # width of pdf (inches)
height = 7,                          # height of pdf (inches)
unit=1000000,                       # bp per unit displayed in plot
ld=NULL,                             # file for LD information
ldCuts = "0,.2,.4,.6,.8,1",         # cut points for LD coloring
ldColors = "gray50,navy,blue,green,orange,red,purple", # colors for LD
rfrows = 2,                          # reflat genes in how many rows?
geneFontSize = .8,                   # size for gene names
geneColor = "navy",                  # color for genes
snpset = "Affy500,Illu318,HapMap",   # SNP sets to show
snpsetFile = NULL,                  # use this file for SNPset data (instead of pquery)

```

Available options for metal2zoom.R

```

rugColor = "gray30",      # color for snpset rugs
rugAlpha = 1,            # alpha for snpset rugs
refFlat = NULL,         # use this file with refFlat info (instead of pquery)
showRecomb = TRUE,     # show recombination rate?
recomb=NULL,           # recombination rate file
recombColor='blue',    # color for recomb rate on plot
recombOver = FALSE,   # overlay recombination rate? (else underlay it)
recombFill = FALSE,   # fill recombination rate? (else line only)
frameColor='gray50',   # frame color for plots
frameAlpha=1,          # frame color for plots
legendAlpha=1,        # transparency of legend background
legend='left',        # legend? (left, right, or none)
hiStart=0,            # start of hilite region
hiEnd=0,              # end of hilite region
hiColor="blue",       # hilite color
hiAlpha=0.1,          # hilite alpha
clobber=TRUE,         # overwrite files?
reload=NULL,          # .Rdata file to reload data from
postlude=NULL,       # code to execute after plot is made
prefix=NULL,         # prefix for output files
dryRun=FALSE         # show a list of the arguments and then halt

```

metal2zoom.R can print the options and their defaults:

```

$ metal2zoom.R dryrun=true
Argument list:
  ldTable = results.ld_point6
  recombTable = results.recomb_rate
  metal = metal.tbl

```

The log file

The log file produced by `metal2zoom.R` includes the following:

```
metal summary:
  Zscore          P.value          Direction          rsquare
Min.   :-5.02600  Min.   :4.148e-08  -----?: 16  Min.   : 0.6829
1st Qu.:-1.64750  1st Qu.:1.995e-03  ???-?: 16  1st Qu.: 0.8000
Median : 0.27900  Median :9.423e-02  -----?: 14  Median : 0.9167
Mean   : 0.05522  Mean   :1.637e-01  +?+?+?: 12  Mean   : 0.8623
3rd Qu.: 1.70000  3rd Qu.:1.845e-01  ???+?: 11  3rd Qu.: 0.9167
Max.   : 5.48400  Max.   :8.750e-01  +++---?: 11  Max.   : 1.0000
                                (Other) :123  NA's   :180.0000

      date: Wed Oct 15 14:46:00 2008
working directory: /home/FUSION/qt-meta/ZoomPlotDevelop2008-09/Examples
      unit: 1000000
  display range: chr2:27551190-27951190 [27551190-27951190]
   hilite range: 0 - 0 [ 0 - 0 ]
  reference SNP: rs2068834
           log: example03.log
          reload: FALSE
      SNP set(s): Affy500,Illu318,HapMap
              ld: example03_ld.tbl
             metal: metal_2:27.3Mb-30.2Mb.tbl
             recomb: example03_recomb.tbl
  all data saved in: example03.Rdata
  metal data saved in: example03_metal.csv
    plot saved in: example03.pdf
number of SNPs plotted: 203
      best p-value: 4.148e-08 [rs1260326]
    p-value range: (4.148e-08, 0.875)
```

Page 2 of the plot

Some of the log information is also stored in a second page of the plot:

```
date: Fri Oct 31 09:27:34 2008
working directory: /home/FUSION/qt-meta/ZoomPlotDevelop2008-09/Examples
unit: 1000000
display range: chr2:27.5Mb-28.4Mb [27500000-28400000]
hilite range: 27.6Mb - 27.95Mb [ 27600000 - 27950000 ]
reference SNP: rs1260326
log: example02.log
SNP set(s): Affy500,Illu318,HapMap
id: example02_id.tbl
metal: metal_2:27.3Mb-30.2Mb.tbl
recomb: example02_recomb.tbl
all data saved in: example02.Rdata
plot saved in: example02.pdf
number of SNPs plotted: 517
best p-value: 4.148e-08 [rs1260326]
p-value range: (4.148e-08, 0.9938)
p-value cut-off: 1
postlude:
```