Developments in optimization tools for R

John C. Nash
Telfer School of Management
University of Ottawa
nashjc@uottawa.ca

Participation in UseR!2011 thanks to support from Nash Information Services Inc.

Thanks to Ravi Varadhan, Hans Werner Borchers, Kate Mullen, Doug Bates and ….

Background

- BIG subject
- Good overview in CRAN Task View: Optimization and Mathematical Programming (Stefan Theussl)
- Many (most?) statistics problems have at least a formal statement as an optimization problem
- For the novice (or not so novice) – too many choices

Attempts to help user

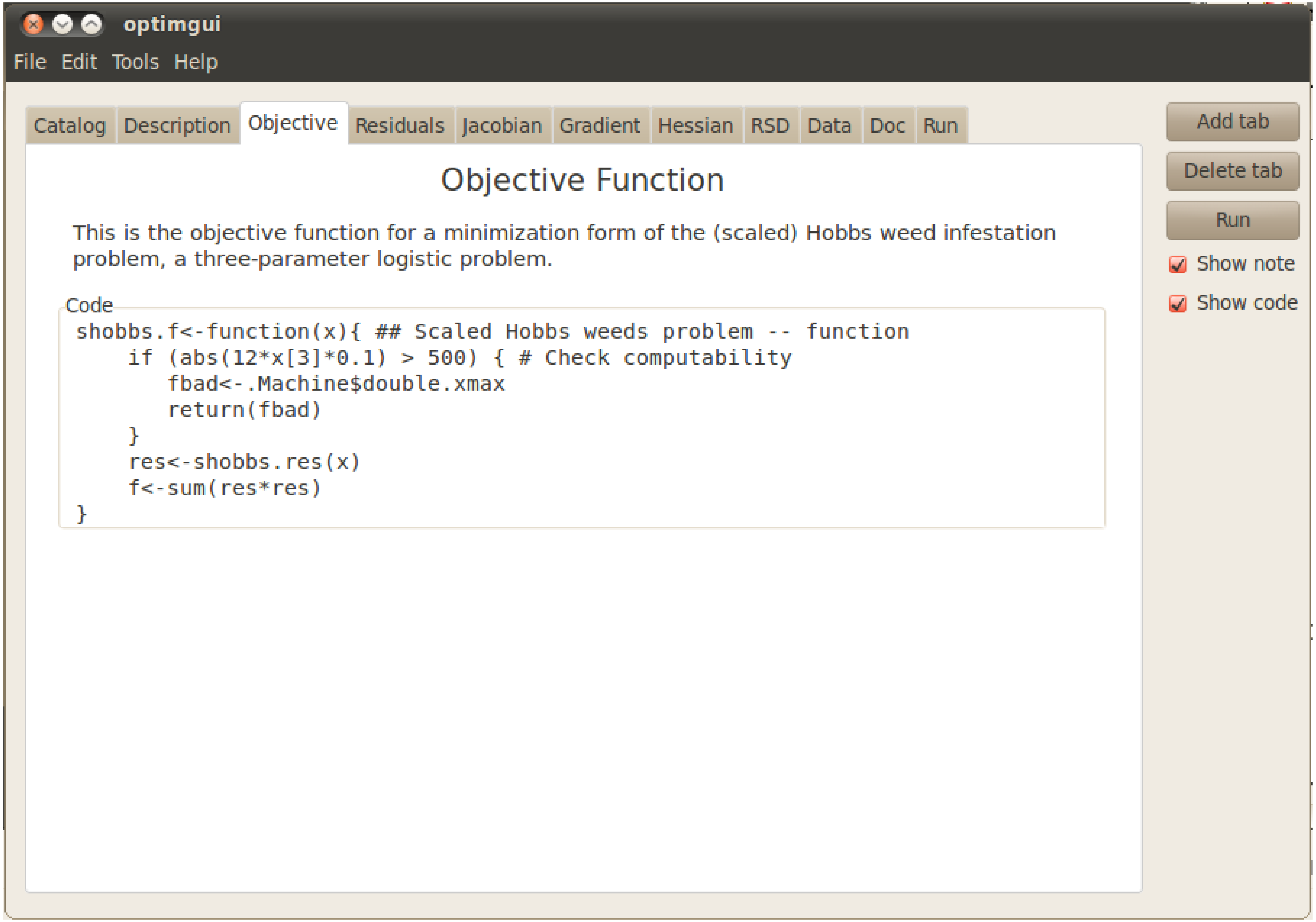
- Package **optimx**
  - Initially to unify a number of tools under a single calling syntax (JSS paper forthcoming with RV)
  - Opportunities for improvements led to
    - Optimality tests (Kuhn Karush Tucker)
    - Scaling tests
    - Bounds checking
    - Inadmissible user-function wrapper
- In-development package **dfoptim** for derivative free methods (some overlap with optimx)

Attempts to help user

- Package **optimx**
  - Initially to unify a number of tools under a single calling syntax (JSS paper forthcoming with RV)
  - Opportunities for improvements led to
    - Optimality tests (Kuhn Karush Tucker)
    - Scaling tests
    - Bounds checking
    - Inadmissible user-function wrapper
- In-development package **dfoptim** for derivative free methods (some overlap with optimx)

Attempts to help user (2)

- Forthcoming package **optimgui** to provide a template for building and running optimization tasks (Yixuan Qiu, Google Summer of Code 2011)
  - Catalog of existing problems that can be modified
  - Problem files combine data, objective, gradient etc. with documentation
  - Linkage to a decision tree to suggest methods
  - Google Summer of Code: student Yixuan Qui

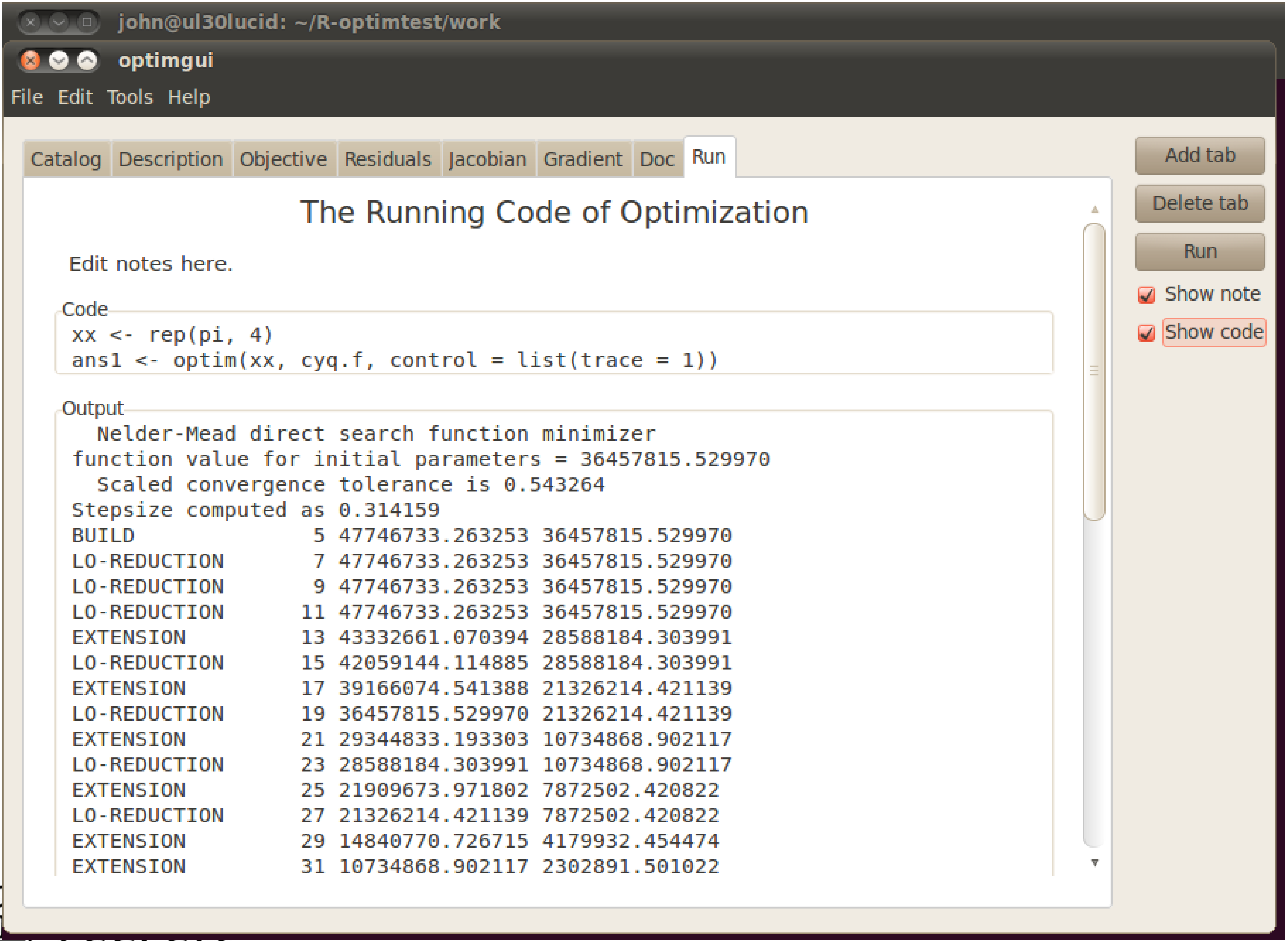

optimgui
File Edit Tools Help
Catalog Description Objective Residuals Jacobian Gradient Hessian RSD Data Doc Run
Add tab
Delete tab
Run
Show note
Show code

Objective Function

This is the objective function for a minimization form of the (scaled) Hobbs weed infestation problem, a three-parameter logistic problem.

Code

```
shobbs.f<-function(x){ ## Scaled Hobbs weeds problem -- function
    if (abs(12*x[3]*0.1) > 500) { # Check computability
       fbad<-.Machine$double.xmax
       return(fbad)
    }
    res<-shobbs.res(x)
    f<-sum(res*res)
}
```


john@ul30lucid: ~/R-optimtest/work
optimgui
File Edit Tools Help
Catalog Description Objective Residuals Jacobian Gradient Doc Run
Add tab
Delete tab
Run
Show note
Show code

The Running Code of Optimization

Edit notes here.

Code

```
xx <- rep(pi, 4)
ans1 <- optim(xx, cyq.f, control = list(trace = 1))
```

Output

```
  Nelder-Mead direct search function minimizer
function value for initial parameters = 36457815.529970
  Scaled convergence tolerance is 0.543264
Stepsize computed as 0.314159
BUILD              5 47746733.263253 36457815.529970
LO-REDUCTION       7 47746733.263253 36457815.529970
LO-REDUCTION       9 47746733.263253 36457815.529970
LO-REDUCTION      11 47746733.263253 36457815.529970
EXTENSION         13 43332661.070394 28588184.303991
LO-REDUCTION      15 42059144.114885 28588184.303991
EXTENSION         17 39166074.541388 21326214.421139
LO-REDUCTION      19 36457815.529970 21326214.421139
EXTENSION         21 29344833.193303 10734868.902117
LO-REDUCTION      23 28588184.303991 10734868.902117
EXTENSION         25 21909673.971802 7872502.420822
LO-REDUCTION      27 21326214.421139 7872502.420822
EXTENSION         29 14840770.726715 4179932.454474
EXTENSION         31 10734868.902117 2302891.501022
```

School of Management

My Active Tasks

- Refactoring of **optimx** / links with **optimgui**
  - Features useful to other optimization tools put in separate packages
  - More optimizers – see also dfoptim
- Under development for **optimx**
  - Axial search around minimum (2*n function evaluations)
  - Grid search to explore “nasty” situations – n^gstep fns – slow!
  - Measures of dispersion – “standard errors”
  - Masks – fixed parameters; need vignettes and examples
  - Extend box constraints to more methods
  - Measures of effort other than timing

My Active Tasks

- Refactoring of **optimx** / links with **optimgui**
  - Features useful to other optimization tools put in separate packages
  - More optimizers – see also dfoptim
- Under development for **optimx**
  - Axial search around minimum (2*n function evaluations)
  - Grid search to explore “nasty” situations – n^gstep fns – slow!
  - Measures of dispersion – “standard errors”
  - Masks – fixed parameters; need vignettes and examples
  - Extend box constraints to more methods
  - Measures of effort other than timing

JN: Medium – Long Term

- Automatic or symbolic derivatives
  - Existing tools awkward; “gaps” in function coverage
  - Need good tutorial material
- Linear & nonlinear constraints
  - Tools for penalty and barrier methods (few)
  - Math programming tools (some in R, but …)
- Measures of dispersion at constraint boundary
- “Noisy” functions – RSMIN
- Automated perfomance data gathering ==> R

JN: Medium – Long Term

- Automatic or symbolic derivatives
  - Existing tools awkward; “gaps” in function coverage
  - Need good tutorial material
- Linear & nonlinear constraints
  - Tools for penalty and barrier methods (few)
  - Math programming tools (some in R, but …)
- Measures of dispersion at constraint boundary
- “Noisy” functions – RSMIN
- Automated perfomance data gathering ==> R

Other work

- Non – R (and possibly not cross platform)
  - NLOpt: http://ab-initio.mit.edu/wiki/index.php/NLopt
  - Eigen: http://eigen.tuxfamily.org
- Multiple-minima → messy optimizations often approached by stochastic methods
  - rgenoud, DEoptim, soma, other developments
- Acceleration of iterations (SQUAREM, etc.)
- Many activities about which I should be better informed!

Other work

- Non – R (and possibly not cross platform)
  - NLOpt: http://ab-initio.mit.edu/wiki/index.php/NLopt
  - Eigen: http://eigen.tuxfamily.org
- Multiple-minima → messy optimizations often approached by stochastic methods
  - rgenoud, DEoptim, soma, other developments
- Acceleration of iterations (SQUAREM, etc.)
- Many activities about which I should be better informed!

Collaboration?

- "standardizing" the infrastructure
    - Cannot be too strict; need compromise
    - Measures of effort vs. timing – what counts?
    - Making constructing function and gradient easier
    - ALL methods fail sometimes – Why?
- Need help from users
    - Tell us what works and what does not (and why!)
    - Help build vignettes, documentation, "best practice", example problem sets

Collaboration? nashjc@uottawa.ca

- "standardizing" the infrastructure
  - Cannot be too strict; need compromise
  - Measures of effort vs. timing – what counts?
  - Making constructing function and gradient easier
  - ALL methods fail sometimes – Why?
- Need help from users
  - Tell us what works and what does not (and why!)
  - Help build vignettes, documentation, "best practice", example problem sets