R and Rcpp

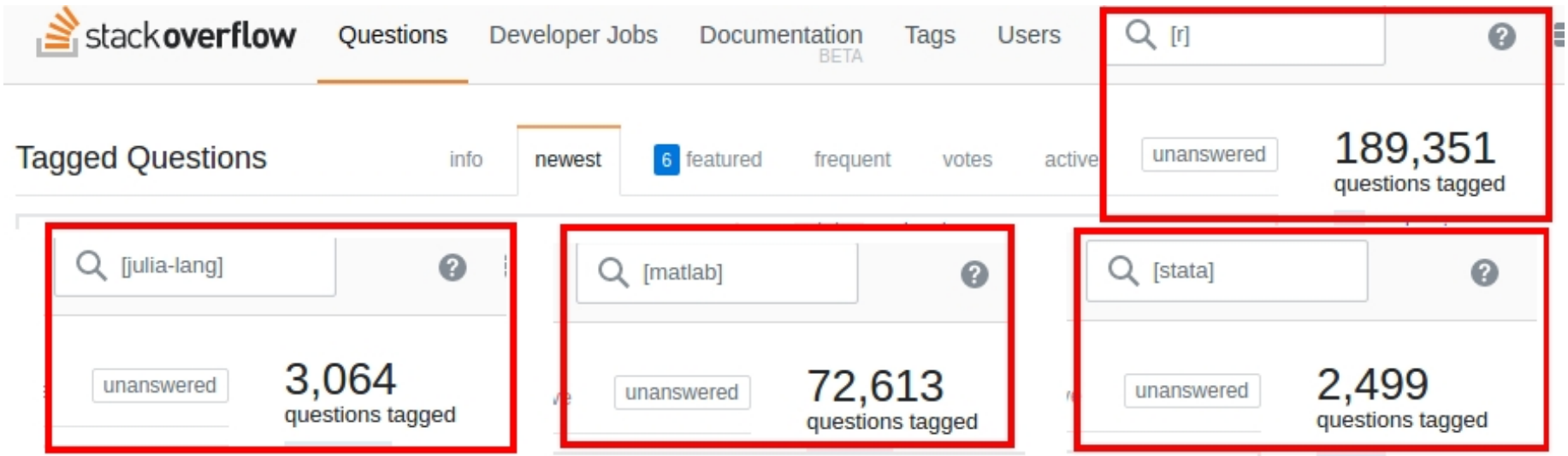
David Zarruk Valencia

October 11, 2017

Why Use R?

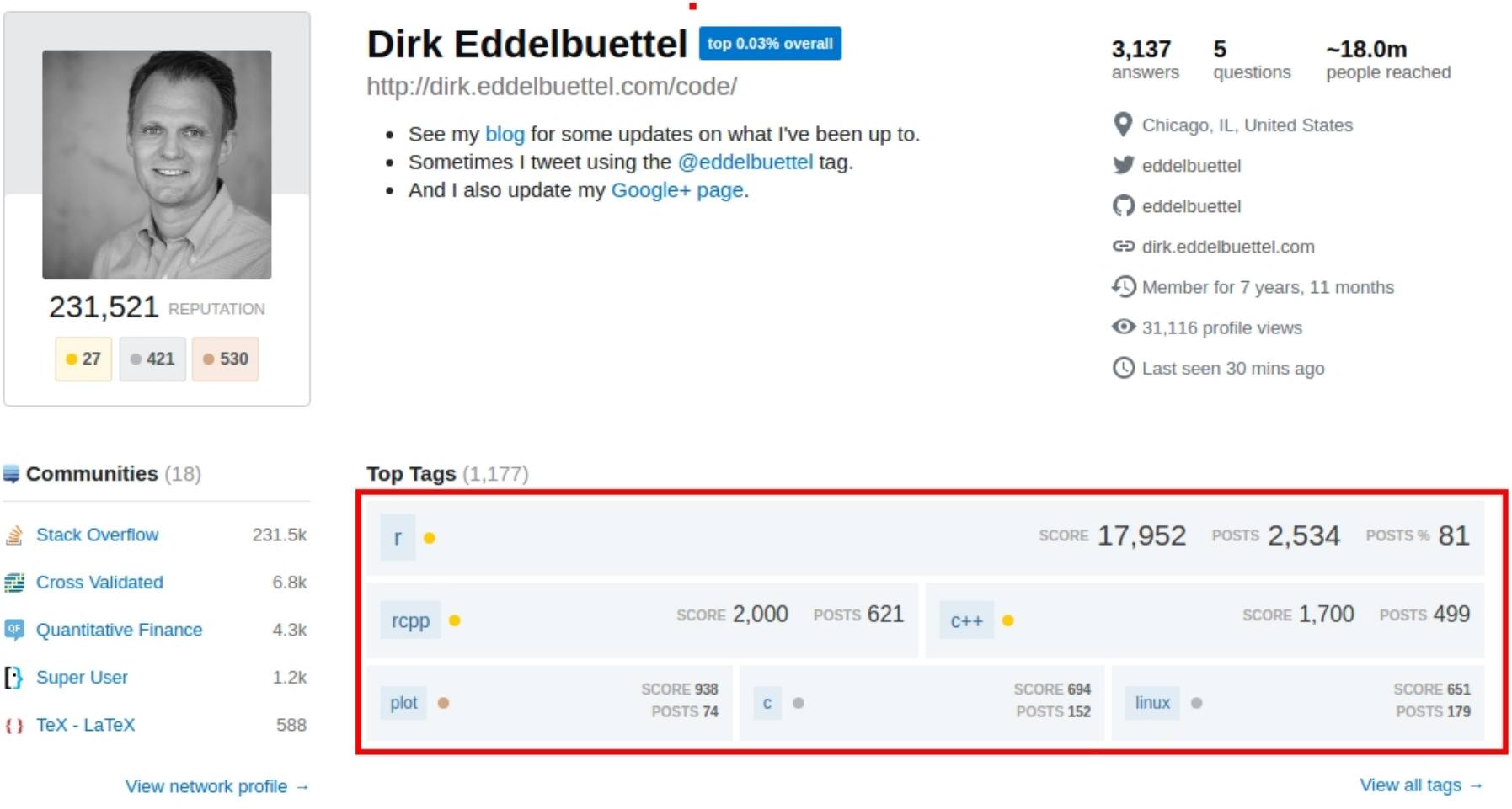
1. Open Source:
   - No license - free
   - Packages for every single task
   - Tons of documentation

   "Am I a developer or just a good googler?"



Why Does Open Source Work?

- Signaling device:



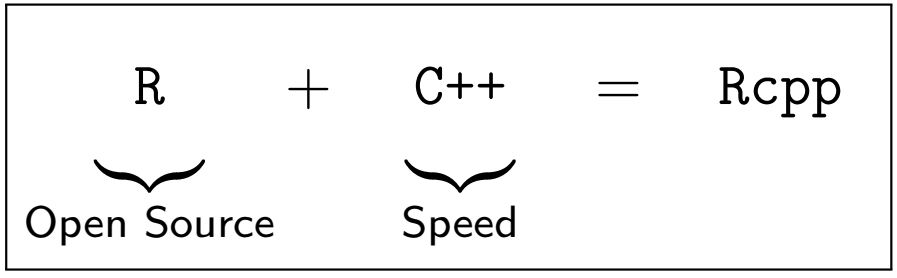
Why Use R?

2. Leading tool for statistics and data processing
   - Coordination game
   - Estimation: regression, RD, non-parametric, semi-parametric, etc.
   - Time series: forecasting, filters, ARIMA, etc.
   - Optimization: global, local, etc.
   - Graphing packages: `ggplot2`, `dplyr`
   - Ongoing research:

3. Additional tools: Machine learning, Data scraping , GIS tools

R and Rcpp

- Only problem of R...
- Very slow at some tasks
- Eg: nested loops
- Solution:



Life-Cycle Model

- Households solve:

$$V(t,e,x) = \max_{\{c,x'\}} \frac{c^{1-\sigma}}{1-\sigma} + \beta \mathbb{E} V(t+1,e',x') \quad s.t.$$

$$c + x' \leq (1+r)x + ew$$

$$\mathbb{P}(e'|e) = \Gamma(e)$$

$$x' \geq 0$$

$$t \in \{1,\ldots,T\}$$

Computing the Model

1. Choose grids for assets $X = \{x_1, \ldots, x_{n_x}\}$ and shocks $E = \{e_1, \ldots, e_{n_e}\}$.

2. Backwards induction:

   2.1 For $t = T$ and every $x_i \in X$ and $e_j \in E$, solve the static problem:

$$V(t, e_j, x_i) = \max_{\{c\}} u(c) \quad s.t. \quad c \leq (1 + r)x_i + e_j w$$

   2.2 For $t = T - 1, \ldots, 1$, use $V(t + 1, e_j, x_i)$ to solve:

$$\begin{aligned} V(t, e_j, x_i) \quad = \quad & \max_{\{c, x' \in X\}} \quad u(c) + \beta \mathbb{E} V(t + 1, e', x') \quad s.t. \\ & c + x' \leq (1 + r)x_i + e_j w \\ & \mathbb{P}(e' \in E | e_j) = \Gamma(e_j) \end{aligned}$$

Code Structure

```
for(age = T:-1:1)
  for(ix = 1:nx)
    for(ie = 1:ne)

      VV = -10^3;
      for(ixp = 1:nx)

        expected = 0.0;
        if(age < T)
          for(iep = 1:ne)
            expected = expected + P[ie, iep]*V[age+1, ixp, iep];
          end
        end

        cons  = (1 + r)*xgrid[ix] + egrid[ie]*w - xgrid[ixp];
        utility = (cons^(1-ssigma))/(1-ssigma) + bbeta*expected;

        if(cons <= 0)
          utility = -10^5;
        end

        if(utility >= VV)
          VV = utility;
        end

      end

      V[age, ix, ie] = VV;

    end
  end
end
```