2014-C12

調査研究報告書

南部アフリカにおける労働参加と失業
Unemployment and participation in the labor force in Southern Africa

2015 年 3 月
独立行政法人日本貿易振興機構
アジア経済研究所

調査研究報告書

地域 2014-C12

南部アフリカにおける労働参加と失業

目次

第 1 章

Reading data from QLFS 2013 and estimate rates under various labour market status

January 23, 2015

Seiro Ito

Abstract In this chapter, we show the nationally representative unemployment rate trends across gender and age groups. We use survey estimators with survey package in R. We will employ knitr to make the analysis reproducible. With knitr run on R, one can embed the computational commands and their outputs easily in a mark up language such as LaTeX, which can be then converted into a widely used format such as Adobe's pdf. In doing so, I will show the exact algorithm to classify the labour market status in 2013 QLFS sample. I will also add a programming memo in the survey and data.table packages.

Key Words Labour market status, discouraged job seekers, youth unemployment

I Introduction

As a short summary, this chapter shows the following:

- Read data from QLFS data file.
- Edit data: adjust numbers for "education" (starts with 0), turn numerically coded variables to factors as summarized in qops.R.
- Decipher QLFS questionnaire of various years and derive algorithms to define labour market status.
- Compute survey estimates: Use R's survey package (Lumley, 2014). Estimate directly the unemployment rates (narrow/wide) and discouraged rates using survey ratio estimators for various subpopulations.
- Plot unemployment rates, never been employed rates by age, gender, educational qualification.

```
library(data.table)
library(knitr)
library(survey)
library(bit64)
x ← fread("qlfs_annual_2013_final_F1.prn")
setnames(x, tolower(names(x)))
```

```
#       read response options
source("qops.R")
        #       attach a number
qops ← lapply(qops, attachnumber)
qops[["metro_code"]] ← metrocode
qops[["geo_type"]] ← geotype
qops[["sector1"]] ← sector1
qops[["sector2"]] ← sector2
        #       adjust numbers for "education"
qops[["q17education"]] ← cbind(asn(qops[["q17education"]][, "number"]) - 1,
        qops[["q17education"]][, "contents"])
qops[["q17education"]] ← rbind(qops[["q17education"]], c(98, NA))
        #       add 99 = NA in "reason stop working"
qops[["q314rsnstopwrk"]] ← rbind(qops[["q314rsnstopwrk"]], c(99, NA))
ii ← asn(lapply(as.list(names(qops)), function(y) grep(y, names(x))))
        #       There is a problem with education. Code is up to 26 but there are 27:30, 9
        #       And grades 12 (13, 17-20) is too few.
        #       There is a problem with q35 reason not working last week.
        #       Code is up to 7 but there are 8, 9.
names(qops)[c(4, 6)]
```

```
[1] "q17education" "q35ynotwrk"
```

```
#       substitute numbers with contents
for (i in 1:length(ii))
{
        if (i == 4) next
        codemat ← qops[[i]]
        x[, names(x)[ii[i]] := factor(asn(x[, ii[i], with = F]), labels = codemat[, "conte
}
#       add metro and nonmetro
x ← x[, metro := !whichgrep("non", x[, metro_code])]
```

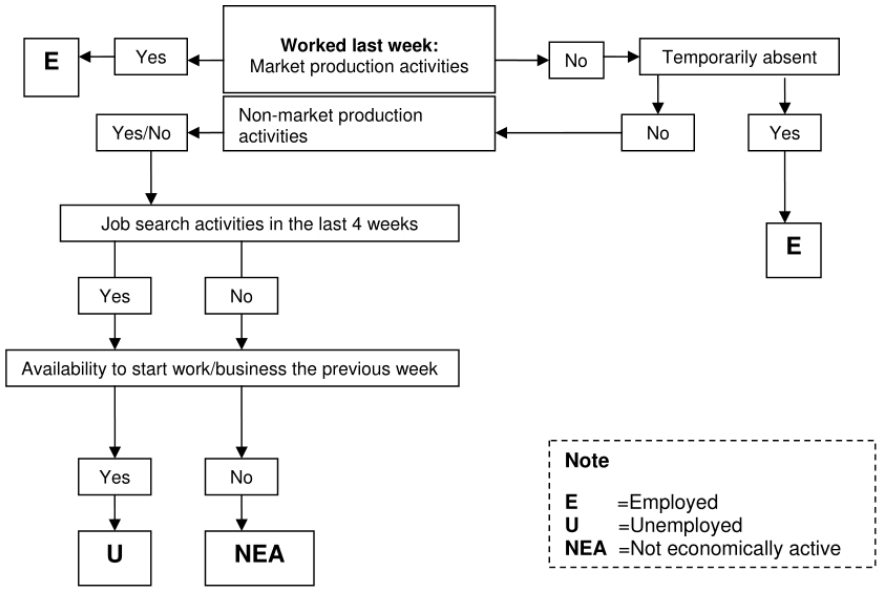
II Labour market status definitions: algorithm

Stats SA's *Quarterly Labour Force Survey* has a farily complicated way of defining the labour market status. Understanding the exact algorithm is not easy, not just because of its deep branching structure of questionnaire but also the questionnaire itslef changes often through years. This section is a modest attempt to clarify the classification algorithm QLFS uses in years 2008 and 2013 questionnaires. The basic definition is given in QLFS's guide (Statistics South Africa, 2008, Figure 2, p.6), however, this is not very useful in actually writing a code to define the labour maket status, because it is not a variable-by-variable guide.

In below we newly define a certain set of variables just to make the algorithm look more understandable. We then use these newly defined variables to define the various labour market status. In

each status, a serial number is given if there is more than one way to reach its status. For example, unemployed status **U** is reached through seven different ways, so there are **U1,. . . , U7**.

Figure 2: Labour force classification



Source: Statistics South Africa (2008, Figure 2, p.6).

II.1 2008

Result: worked, absent, lkforjob, wdaccptjob, XCosNEA → E, U, D.

if 2.4: worked Worked last week. **then**
  E
else
  if 2.5: absent Absent from work. **then**
    if 2.7: absentCosNEA Absent temporarily. **then**
      E
    else
      Go to 3.1a (Not employed.)
    end
  else
    if 3.1a: lkforjob Looked for jobs last 4 weeks. **then**
      3.2 → 3.6 → 3.7 →
      if 3.9, 3.10: wdaccptjob == T **then**
        3.12: everwrked →
        if 3.18: lkforjob2 **then**
          U (would accept suitable job, searched.)
        else
          U/D *(would accept suitable job, said searched in 3.1 but said not searched in 3.18.)*
        end
      else
        3.11: naCosNEA Not available for work because ineligible.
        if 3.12: everwrked **then**
          N (NEA [3.11] or not available for work, not searched.)
          D (EA [3.11] or available for work, not searched.)
        else
          if 3.18: lkforjob2 **then**
            D/N *(NEA [3.11] or not available for work, have not worked, searched.)*
            U *(EA [3.11] or available for work, have not worked, searched.)*
          else
            N (NEA [3.11] or not available for work, have not worked, not searched.)
            D (EA [3.11] or available for work, have not worked, not searched.)
          end
        end
      end
    else
      See next table.
    end
  end
end

Algorithm 1: Classification of labour market status

```
Not worked, not temporarily absent, not looked for job.
if 3.1b: startbiz Start business or 3.3: plantostart plan to start work. then
    3.6 → 3.7 →
    if 3.9, 3.10: wdaccptjob == T then
        U (Not worked, not searched, planned to start work, but would accept job.)
    else
        U (Not worked, not searched, planned to start work, would not accept job.)
    end
else
    if 3.4: wantedwrk Wanted to work. then
        3.8: notseekCosNEA Not looked for job because ineligible. (discouraged)
        if 3.9, 3.10: wdaccptjob Would have accepted suitable job. then
            if 3.12: everwrked then
                3.13 → ... →
                if 3.18: lkforjob2 then
                    U (EA [3.8] or available for work, have worked, wanted to work, would accept
                    suitable job, said not searched in 3.1 but said searched in 3.18.)
                    D/N/U (NEA [3.8] or not available for work, have worked, wanted to work,
                    would accept suitable job, said not searched in 3.1 but said searched in 3.18.)
                else
                    U (EA [3.8] or available for work, have worked, wanted to work, would accept
                    suitable job, said not searched in 3.1 and 3.18.)
                    D (EA [3.8] lost hope, have worked, wanted to work, would accept suitable job,
                    said not searched in 3.1 and 3.18.)
                    N (NEA [3.8] or not available for work, have worked, wanted to work, would
                    accept suitable job, said not searched in 3.1 and 3.18. Suitable jobs may not
                    exist for NEA.)
                end
            else
                if 3.18: lkforjob2 then
                    U (EA [3.8] or available for work, have not worked, wanted to work, would
                    accept suitable job, searched.)
                    N/U (NEA [3.8] or not available for work, have not worked, wanted to work,
                    would accept suitable job, searched. Suitable jobs may not exist for NEA.)
                else
                    D (EA [3.8] lost hope, have not worked, wanted to work, would accept suitable
                    job, not searched.)
                    U (EA [3.8] or available for work, have not worked, wanted to work, would
                    accept suitable job, not searched.)
                    N (NEA [3.8] or not available for work, have not worked, wanted to work, would
                    accept suitable job, not searched. Suitable jobs may not exist for NEA.)
                end
            end
        else
            N (Would not work for suitable job, not searched.)
        end
    else
        Go to next table.
    end
end
```

Algorithm 2: Classification of labour market status

Result: worked, absent, lkforjob, wdaccptjob, XCosNEA → E, U, D.

Not worked, not temporarily absent, not looked for job, not started business, not wanted to work.
3.5: notwantedwrkCosNEA Not wanted to work because ineligible.
if 3.12: everwrked **then**
    if 3.18: lkforjob2 **then**
        ***N** (NEA [3.5] or not available for work, not wanted to work, searched.)*
    else
        N (NEA [3.5] or not available for work, not wanted to work, not searched.)
    end
else
    if 3.18: lkforjob2 **then**
        ***N** (NEA [3.5] or not available for work, have not worked, not wanted to work, searched.)*
    else
        N (NEA [3.5] or not available for work, have not worked, not wanted to work, not searched.)
    end
end

Algorithm 3: Classification of labour market status

II.2 2013

Result: worked, absent, lkforjob, wdaccptjob, XCosNEA → E, U, D.

```
1  if 2.4: worked Worked last week. then
2      E
3  else
4      if 2.5: absent Absent from work. then
5          if 2.7: absentCosNEA Absent temporarily. then
6              E
7          else
8              Go to 3.1a (Not employed.)
9          end
10     else
11         if 3.1a: lkforjob Looked for jobs last 4 weeks. then
12             3.2 → 3.6 → 3.7 →
13             if 3.9, 3.10: wdaccptjob == T then
14                 3.12: everwrked →
15                 U1 (would accept suitable job, searched.)
16             else
17                 3.11: naCosNEA Not available for work because ineligible.
18                 if 3.12: everwrked then
19                     N1 (NEA [3.11] or not available for work, searched.)
20                     U2 (EA [3.11] or available for work, searched.)
21                 else
22                     N2 (NEA [3.11] or not available for work, have not worked,
                       searched.)
23                     U3 (EA [3.11] or available for work, have not worked,
                       searched.)
24                 end
25             end
26         else
27             See next table.
28         end
29     end
30 end
```

Algorithm 4: Classification of labour market status

```
Not worked, not temporarily absent, not looked for job.
if 3.1b: startbiz Start business or 3.3: plantostart plan to start work. then
    3.6 → 3.7 →
    if 3.9, 3.10: wdaccptjob == T then
        U4 (Not worked, not searched, planned to start work, but would accept job.)
    else
        U5 (Not worked, not searched, planned to start work, would not accept job.)
    end
else
    if 3.4: wantedwrk Wanted to work. then
        3.8: notseekCosNEA Not looked for job because ineligible. (discouraged)
        if 3.9, 3.10: wdaccptjob Would have accepted suitable job. then
            if 3.12: everwrked then
                3.13 → ... →
                D1 (EA [3.8] lost hope, have worked, wanted to work, would accept suitable
                job, not searched.)
                U6 (EA [3.8] or available for work, have worked, wanted to work, would
                accept suitable job, not searched.)
                N3 (NEA [3.8] or not available for work, have worked, wanted to work,
                would accept suitable job, not searched. Suitable jobs may not exist for NEA.)
            else
                D2 (EA [3.8] lost hope, have not worked, wanted to work, would accept
                suitable job, not searched.)
                U7 (EA [3.8] or available for work, have not worked, wanted to work, would
                accept suitable job, not searched.)
                N4 (NEA [3.8] or not available for work, have not worked, wanted to work,
                would accept suitable job, not searched. Suitable jobs may not exist for NEA.)
            end
        else
            N5 (Would not work for suitable job, not searched.)
        end
    else
        3.5: notwantedwrkCosNEA Not wanted to work because ineligible.
        if 3.12: everwrked then
            N6 (NEA [3.5] or not available for work, not wanted to work, not searched.)
        else
            N7 (NEA [3.5] or not available for work, have not worked, not wanted to work,
            not searched.)
        end
    end
end
```

Algorithm 5: Classification of labour market status

```
   Result: dd41, dd51 → e, u, d.
 1 if 2.4: worked Worked last week. then
 2 |  E
 3 else
 4 |  if 2.5: absent Absent from work. then
 5 |  |  if 2.7: absentCosNEA Absence due to ineligibility. then
 6 |  |  |  Go to 3.1a
 7 |  |  else
 8 |  |  |  Go to 3.6
 9 |  |  end
10 |  else
11 |  |  if 3.1a: lkforjob Looked for jobs last 4 weeks. then
12 |  |  |  Go to 3.6
13 |  |  else
14 |  |  |  if 3.1b: startbiz Start business. then
15 |  |  |  |  Go to 3.6
16 |  |  |  else
17 |  |  |  |  if 3.3: plantostart Plan to start work. then
18 |  |  |  |  |  3.6 → 3.7 →
19 |  |  |  |  |  if 3.9, 3.10: wdaccptwrk then
20 |  |  |  |  |  |  U
21 |  |  |  |  |  else
22 |  |  |  |  |  |  3.11 → Go to 3.12
23 |  |  |  |  |  end
24 |  |  |  |  else
25 |  |  |  |  |  if 3.4: wantedwrk Wanted to work. then
26 |  |  |  |  |  |  (confirmation only) 3.8: notseekCosNEA
   |  |  |  |  |  |  Not looked for job because ineligible.
27 |  |  |  |  |  |  if 3.9, 3.10: wdaccptjob Would have
   |  |  |  |  |  |  accepted suitable job. then
28 |  |  |  |  |  |  |  D
29 |  |  |  |  |  |  else
30 |  |  |  |  |  |  |  N
31 |  |  |  |  |  |  end
32 |  |  |  |  |  else
33 |  |  |  |  |  |  (confirmation only) 3.5:
   |  |  |  |  |  |  notwantedwrkCosNEA Not wanted to work
   |  |  |  |  |  |  because ineligible.
34 |  |  |  |  |  |  if 3.12: everwrked then
35 |  |  |  |  |  |  |  D
36 |  |  |  |  |  |  else
37 |  |  |  |  |  |  |  if 3.18: lkforjob2 then
38 |  |  |  |  |  |  |  |  U
39 |  |  |  |  |  |  |  else
40 |  |  |  |  |  |  |  |  D
41 |  |  |  |  |  |  |  end
42 |  |  |  |  |  |  end
43 |  |  |  |  |  end
44 |  |  |  |  end
45 |  |  |  end
46 |  |  end
47 |  end
48 end
49 ;
```

Algorithm 6: Classification of labour market status

- worked: Worked at least one hour for market activities. Yes to *any* of q24apdwrk, q24bownbusns, q24cunpdwrk. If yes, all the variables below will be NA.
- absent: Not worked and yes to *any* of q25apdwrk, q25bownbusns, q25cunpdwrk (all three ask if absent from regular work).
- absentCosNEA: Reason for absense is not caring family members, study leave, laid off, seasonal in q27rsnabsent.
- plantostart: Yes to q33havejob (Not worked because there was a prior arrangement to start later than last week).
- lkforjob: Yes to q31alookwrk (Were you looking for a job last 4 weeks?). Asked to all who "did not work last week."
- startbiz: Yes to q31bstartbusns (Were you starting business last 4 weeks?). Asked to all who "did not work last week" and "did not look for jobs in last 4 weeks."
- wantedwrk: Yes to q34wanttowrk (Would you have liked to work last week?) Asked to "not worked last week" and "not looking for a job nor started business in last 4 weeks."
- notwantedwrkCosNEA: q35ynotwrk is either student, homemaker, health, retired, no desire, too young. Asked to all who "did not work last week."
- notseekCosNEA: q38rsnnotseek is either student, homemaker, health, retired, no desire, too young, pregnancy, disabled, under training. Only this question provides a criteria for classifying discouraged job seekers. Asked to all who "did not work last week" and "did not look for jobs in last 4 weeks."
- wdaccptjob: Would have accepted to work if offered. Yes to *any* of q39joboffer, q310startbusns. Asked to "not worked last week" and "not looking for a job nor started business in last 4 weeks".

```
#       check age group
dim(x)
```

```
[1] 346687    164
```

```
setkey(x, q14age)
dim(x ← x[x[, q14age] ≥ 15 & x[, q14age] ≤ 60, ])
```

```
[1] 206409    164
```

```
#       check if first question is answered by everyone
table0(ii ← is.na(x[, q24apdwrk]) & is.na(x[, q24bownbusns]) & is.na(x[, q24cunpdwrk]))
```

```
 FALSE
206409
```

```
x ← x[!ii, ]
x ← x[, worked := whichgrep(1, x[, q24apdwrk]) |
        whichgrep(1, x[, q24bownbusns]) | whichgrep(1, x[, q24cunpdwrk])]
x ← x[, absent := whichgrep(1, x[, q25apdwrk]) |
                whichgrep(1, x[, q25bownbusns]) | whichgrep(1, x[, q25cunpdwrk])]
x ← x[, absentCosNEA := whichgrep("hea|caring|mater|obli|study", x[, q27rsnabsent])]
x ← x[, lkforjob := whichgrep(1, x[, q31alookwrk])]
x ← x[, startbiz := whichgrep(1, x[, q31bstartbusns])]
x ← x[, plantostart := whichgrep(1, x[, q33havejob])]
x ← x[, wdaccptjob := whichgrep(1, x[, q39joboffer]) | whichgrep(1, x[, q310startbusns])]
```

```
x ← x[, everwrked := whichgrep(1, x[, q312everwrk])]
x ← x[, neverwrked := whichgrep(2, x[, q312everwrk])]
x ← x[, wantedwrk := whichgrep(1, x[, q34wanttowrk])]
x ← x[, notwantedwrkCosNEA := whichgrep("hea|preg|disa|wif|scho|too|desir", x[, q35ynotwr
x ← x[, notseekCosNEA := whichgrep("hea|preg|disa|wif|train|scho|retir|old", x[, q38rsnn
x ← x[, naCosNEA := whichgrep("stude|wife|health|reti|young", x[, q311rsnnotavailable])]
x ← x[, CosNEA := notwantedwrkCosNEA | notseekCosNEA | naCosNEA]
x ← x[, losthope := whichgrep("no job|^unable|hope", x[, q38rsnnotseek])]
#       employed: worked or absent while eligible or preparing to start
x ← x[, employed := worked | (absent & !absentCosNEA)]
x ← x[, Ubeforedis :=
                #       U1
                ((lkforjob & wdaccptjob) |
                #       U2, U3
                (lkforjob & !wdaccptjob & !naCosNEA) |
                #       U4, U5
                (!lkforjob & (startbiz | plantostart)) |
                #       U6, U7
                (!lkforjob & !(startbiz | plantostart) & wantedwrk &
                 wdaccptjob & !notseekCosNEA & !losthope)) ]
#       discouraged: not employed and not looking for job and would accept to work if suit
x ← x[, discouraged := !employed & !lkforjob &
                #       D1, D2
                !(startbiz | plantostart) & wantedwrk & wdaccptjob]
#       unemployed: not employed and (looking for job|discouraged)
x ← x[, unemployed := !employed & (discouraged | Ubeforedis) ]
#       not economically active: would not accept job if suitable or not wanted to work
x ← x[, nea := !employed & !unemployed &
                #       N1, N2
                ((lkforjob & !wdaccptjob & naCosNEA) |
                #       N3, N4
                (!lkforjob & !(startbiz | plantostart) & wantedwrk & wdaccptjob &
                notseekCosNEA) |
                #       N5
                (!lkforjob & !(startbiz | plantostart) & wantedwrk & !wdaccptjob) |
                #       N6, N7
                (!lkforjob & !(startbiz | plantostart) & !wantedwrk &
                notwantedwrkCosNEA))]
#       narrower unemployed: not employed and looking for job
x ← x[, c("narrowU", "narrowNEA") := list(!employed & Ubeforedis, nea | discouraged) ]
x ← x[!is.na(x[, employed]) & x[, employed], lmstatus := 1]
x ← x[!is.na(x[, unemployed]) & x[, unemployed], lmstatus := 2]
x ← x[!is.na(x[, nea]) & x[, nea], lmstatus := 3]
#       drop who are neither NEA, U, E
dim(x ← x[!(is.na(employed) & is.na(unemployed) & is.na(nea)), ])
```

```
[1] 206409    187
```

```
table0(x[, c("q24apdwrk", "q24bownbusns"), with = F])
```

```
          q24bownbusns
q24apdwrk     1      2
        1    74  70361
        2 11083 124891
```

```
table0(x[, c("q24apdwrk", "q24cunpdwrk"), with = F])
```

```
          q24cunpdwrk
q24apdwrk    1      2
        1    3  70432
        2  547 135427
```

```
table0(x[, c("lmstatus", "employed"), with = F])
```

```
        employed
lmstatus FALSE  TRUE
    1        0 83657
    2    49615     0
    3    71486     0
    <NA>  1651     0
```

```
table0(x[, c("lmstatus", "unemployed"), with = F])
```

```
        unemployed
lmstatus FALSE  TRUE
    1    83657     0
    2        0 49615
    3    71486     0
    <NA>  1651     0
```

```
table0(x[, c("lmstatus", "nea"), with = F])
```

```
        nea
lmstatus FALSE  TRUE
    1    83657     0
    2    49615     0
    3        0 71486
    <NA>  1651     0
```

```
table(x[, c("employed", "unemployed", "nea"), with = F])
```

```
, , nea = FALSE

        unemployed
employed FALSE  TRUE
   FALSE  1651 49615
   TRUE  83657     0

, , nea = TRUE

        unemployed
employed FALSE  TRUE
   FALSE 71486     0
   TRUE      0     0
```

```
x[!x[, nea] & !x[, employed] & !x[, unemployed],
        c("q24apdwrk", "q24bownbusns", "q24cunpdwrk",
        "q25apdwrk", "q25bownbusns", "q25cunpdwrk", "q31alookwrk",
        "wdaccptjob", "q39joboffer", "q310startbusns",
        "wantedwrk", "notwantedwrkCosNEA", "lmstatus"), with = F]
```

```
      q24apdwrk q24bownbusns q24cunpdwrk q25apdwrk q25bownbusns q25cunpdwrk
   1:         2            2           2         2            2           2
   2:         2            2           2         2            2           2
   3:         2            2           2         2            2           2
```

```
   4:          2             2             2          2             2             2
   5:          2             2             2          2             2             2
  ---
1647:          2             2             2          2             2             2
1648:          2             2             2          1            NA            NA
1649:          2             2             2          2             2             2
1650:          2             2             2          1            NA            NA
1651:          2             2             2          2             2             2
      q31alookwrk wdaccptjob q39joboffer q310startbusns wantedwrk
   1:           2      FALSE          NA             NA     FALSE
   2:           2      FALSE          NA             NA     FALSE
   3:           2      FALSE          NA             NA     FALSE
   4:           2      FALSE          NA             NA     FALSE
   5:           2      FALSE          NA             NA     FALSE
  ---
1647:           2      FALSE          NA             NA     FALSE
1648:          NA      FALSE          NA             NA     FALSE
1649:           2      FALSE          NA             NA     FALSE
1650:          NA      FALSE          NA             NA     FALSE
1651:           2      FALSE          NA             NA     FALSE
      notwantedwrkCosNEA lmstatus
   1:              FALSE       NA
   2:              FALSE       NA
   3:              FALSE       NA
   4:              FALSE       NA
   5:              FALSE       NA
  ---
1647:              FALSE       NA
1648:              FALSE       NA
1649:              FALSE       NA
1650:              FALSE       NA
1651:              FALSE       NA
```

```
x[is.na(lmstatus), grep("q2|q3|sta|emp|nea|NEA|dis|wante|job|work|accp",
        colnames(x)), with = F][1:3, ]
```

```
   q16maritalstatus q20selfrespond q24apdwrk q24bownbusns q24cunpdwrk q25apdwrk
1:    never married              1         2            2           2         2
2:    never married              2         2            2           2         2
3:    never married              2         2            2           2         2
   q25bownbusns q25cunpdwrk q27rsnabsent q31alookwrk q31bstartbusns
1:            2           2           NA           2              2
2:            2           2           NA           2              2
3:            2           2           NA           2              2
   q3201register q3202enquire q3203jobads q3204jobsearch q3205assistance
1:            NA           NA          NA              NA              NA
2:            NA           NA          NA              NA              NA
3:            NA           NA          NA              NA              NA
   q3206startbusns q3207casual q3208finassist q3210nothing q33havejob
1:              NA          NA             NA           NA          2
2:              NA          NA             NA           NA          2
3:              NA          NA             NA           NA          2
   q34wanttowrk     q35ynotwrk q36timeseek q37actpriorjobseek q38rsnnotseek
1:            2 other, specify          NA                 NA            NA
2:            2 other, specify          NA                 NA            NA
3:            2 other, specify          NA                 NA            NA
   q39joboffer q310startbusns q311rsnnotavailable q311bwhnstart q312everwrk
1:          NA             NA                  NA            NA           2
2:          NA             NA                  NA            NA           2
3:          NA             NA                  NA            NA           2
   q313timeunemploy q314rsnstopwrk q315prevoccupation q316previndustry
```

```
1:               NA             NA                  NA               NA
2:               NA             NA                  NA               NA
3:               NA             NA                  NA               NA
   q317wrk4whom q319aoddjobs q319binhhpers q319cnothhpers q319dcharity q319euif
1:           NA            2             1              2            2        2
2:           NA            2             1              2            2        2
3:           NA            2             1              2            2        2
   q319fsavings q319gpension q319hgrants q319iwelfare q319jothr q41multiplejobs
1:            2            2           2            2         2              NA
2:            2            2           2            2         2              NA
3:            2            2           2            2         2              NA
   q44yearstart q44monthstart q416nrworkers q425startxwrk unempl_status status
1:           NA            NA            NA            NA             3      4
2:           NA            NA            NA            NA             3      4
3:           NA            NA            NA            NA             3      4
   education_status long_term_unempl underempl infempl status_exp worked
1:                2               NA         NA      NA          4  FALSE
2:                3               NA         NA      NA          4  FALSE
3:                4               NA         NA      NA          4  FALSE
   absentCosNEA lkforjob startbiz plantostart wdaccptjob wantedwrk
1:        FALSE    FALSE    FALSE       FALSE      FALSE     FALSE
2:        FALSE    FALSE    FALSE       FALSE      FALSE     FALSE
3:        FALSE    FALSE    FALSE       FALSE      FALSE     FALSE
   notwantedwrkCosNEA notseekCosNEA naCosNEA CosNEA employed Ubeforedis
1:              FALSE         FALSE    FALSE  FALSE    FALSE      FALSE
2:              FALSE         FALSE    FALSE  FALSE    FALSE      FALSE
3:              FALSE         FALSE    FALSE  FALSE    FALSE      FALSE
   discouraged unemployed   nea narrowNEA lmstatus
1:       FALSE      FALSE FALSE     FALSE       NA
2:       FALSE      FALSE FALSE     FALSE       NA
3:       FALSE      FALSE FALSE     FALSE       NA
```

```
table0(x[, c("employed", "lkforjob"), with = F])
```

```
        lkforjob
employed FALSE  TRUE
   FALSE 94362 28390
   TRUE  83634    23
```

```
table(x[, c("unemployed", "startbiz"), with = F])
```

```
          startbiz
unemployed  FALSE   TRUE
     FALSE 156793      1
     TRUE   49286    329
```

```
table0(x[, c("unemployed", "employed"), with = F])
```

```
          employed
unemployed FALSE  TRUE
     FALSE 73137 83657
     TRUE  49615     0
```

```
table(x[, c("unemployed", "discouraged"), with = F])
```

```
          discouraged
unemployed  FALSE   TRUE
     FALSE 156794      0
     TRUE   29025  20590
```

```
table0(x[, c("employed", "wdaccptjob"), with = F])
```

```
        wdaccptjob
employed FALSE  TRUE
   FALSE 73157 49595
   TRUE  83601    56
```

III Survey estimates

Programming memo:

- When estimating a mean of a subpopulation, one use svyby and set subpopulation with a formula format.
- When estimating a ratio of a subpopulation, one can still use svyby, but without numerator argument explicitly defined, define denominator, and set subpopulation with a formula format.
- For example, when one estimates $\frac{a}{b}$, one set svyby(~ a, by = ~ factor1 * factor2, denominator = ~ b, design = survey.design.object, svyratio, vartype = "ci", na.rm = T).

```
#       edulevel
x ← x[!is.na(q17education) & q17education ≤ 11, edulevel := 0]
x ← x[!is.na(q17education) & q17education == 12, edulevel := 1]
x ← x[!is.na(q17education) & q17education ≥ 13 & q17education ≤ 28, edulevel := 2]
x ← x[!is.na(q17education) & q17education ≥ 29, edulevel := NA]
x ← x[, edulevel := factor(x[, edulevel],
        labels = c("below matrics", "matrics", "above matrics"))]
table0(x[, edulevel])
```

```
below matrics       matrics above matrics          <NA>
       125552         49675         22807         8375
```

```
#       turn logical to binary
x ← x[, employed := (!is.na(employed) & employed) + 0]
x ← x[, unemployed := (!is.na(unemployed) & unemployed) + 0]
x ← x[, nea := (!is.na(nea) & nea) + 0]
x ← x[, discouraged := (!is.na(discouraged) & discouraged) + 0]
x ← x[, narrowU := (!is.na(narrowU) & narrowU) + 0]
x ← x[, narrowNEA := (!is.na(narrowNEA) & narrowNEA) + 0]
x ← x[, everwrked := (!is.na(everwrked) & everwrked) + 0]
x ← x[, neverwrked := (!is.na(neverwrked) & neverwrked) + 0]
dsgn ← svydesign(id = ~1, strata = ~stratum, weights = ~weight, pps = "brewer", data = x)
em ← svytotal(~ employed, dsgn, na.rm = T)
un ← svytotal(~ unemployed, dsgn, na.rm = T)
ne ← svytotal(~ nea, dsgn, na.rm = T)
nvr ← svytotal(~ neverwrked, dsgn, na.rm = T)
urt ← svyratio(~ unemployed, ~ I(employed + unemployed), dsgn, na.rm = T)
nrt ← svyratio(~ narrowU, ~ I(narrowU + employed), dsgn, na.rm = T)
vrt ← svyratio(~ neverwrked, ~ I(employed + unemployed + nea), dsgn, na.rm = T)
unbyage ← svyby(~ unemployed, ~ q14age * q13gender, dsgn, svytotal, na.rm = T)
embyage ← svyby(~ employed, ~ q14age * q13gender, dsgn, svytotal, na.rm = T)
neabyage ← svyby(~ nea, ~ q14age * q13gender, dsgn, svytotal, na.rm = T)
```

```
disbyage ← svyby(~ discouraged, ~ q14age * q13gender, dsgn, svytotal, na.rm = T)
nubyage ← svyby(~ narrowU, ~ q14age * q13gender, dsgn, svytotal, na.rm = T)
nvrbyage ← svyby(~ neverwrked, ~ q14age * q13gender, dsgn, svytotal, na.rm = T)
urtbyage ← svyby(~ unemployed, by = ~ q14age * q13gender,
        denominator = ~ I(employed + unemployed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
nrtbyage ← svyby(~ narrowU, by = ~ q14age * q13gender,
        denominator = ~ I(narrowU + employed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
vrtbyage ← svyby(~ neverwrked, by = ~ q14age * q13gender,
        denominator = ~ I(employed + unemployed + nea),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
unbyageedu ← svyby(~ unemployed, ~ q14age * q13gender * edulevel,
        dsgn, svytotal, na.rm = T)
embyageedu ← svyby(~ employed, ~ q14age * q13gender * edulevel,
        dsgn, svytotal, na.rm = T)
neabyageedu ← svyby(~ nea, ~ q14age * q13gender * edulevel,
        dsgn, svytotal, na.rm = T)
disbyageedu ← svyby(~ discouraged, ~ q14age * q13gender * edulevel,
        dsgn, svytotal, na.rm = T)
nubyageedu ← svyby(~ narrowU, ~ q14age * q13gender * edulevel,
        dsgn, svytotal, na.rm = T)
nvrbyageedu ← svyby(~ neverwrked, ~ q14age * q13gender * edulevel,
        dsgn, svytotal, na.rm = T)
urtbyageedu ← svyby(~ unemployed, by = ~ q14age * q13gender * edulevel,
        denominator = ~ I(employed + unemployed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
nrtbyageedu ← svyby(~ narrowU, by = ~ q14age * q13gender * edulevel,
        denominator = ~ I(narrowU + employed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
vrtbyageedu ← svyby(~ neverwrked, by = ~ q14age * q13gender * edulevel,
        denominator = ~ I(employed + unemployed + nea),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)

#       by metro vs. non-metro
unbyagemetro ← svyby(~ unemployed, ~ q14age * q13gender * metro,
        dsgn, svytotal, na.rm = T)
embyagemetro ← svyby(~ employed, ~ q14age * q13gender * metro,
        dsgn, svytotal, na.rm = T)
neabyagemetro ← svyby(~ nea, ~ q14age * q13gender * metro,
        dsgn, svytotal, na.rm = T)
disbyagemetro ← svyby(~ discouraged, ~ q14age * q13gender * metro,
        dsgn, svytotal, na.rm = T)
nubyagemetro ← svyby(~ narrowU, ~ q14age * q13gender * metro,
        dsgn, svytotal, na.rm = T)
nvrbyagemetro ← svyby(~ neverwrked, ~ q14age * q13gender * metro,
        dsgn, svytotal, na.rm = T)
urtbyagemetro ← svyby(~ unemployed, by = ~ q14age * q13gender * metro,
        denominator = ~ I(employed + unemployed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
nrtbyagemetro ← svyby(~ narrowU, by = ~ q14age * q13gender * metro,
        denominator = ~ I(narrowU + employed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
vrtbyagemetro ← svyby(~ neverwrked, by = ~ q14age * q13gender * metro,
        denominator = ~ I(employed + unemployed + nea),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
```

```
#       by metro vs. non-metro
unbyageedumetro ← svyby(~ unemployed, ~ q14age * q13gender * edulevel * metro,
        dsgn, svytotal, na.rm = T)
embyageedumetro ← svyby(~ employed, ~ q14age * q13gender * edulevel * metro,
        dsgn, svytotal, na.rm = T)
neabyageedumetro ← svyby(~ nea, ~ q14age * q13gender * edulevel * metro,
        dsgn, svytotal, na.rm = T)
disbyageedumetro ← svyby(~ discouraged, ~ q14age * q13gender * edulevel * metro,
        dsgn, svytotal, na.rm = T)
nubyageedumetro ← svyby(~ narrowU, ~ q14age * q13gender * edulevel * metro,
        dsgn, svytotal, na.rm = T)
nvrbyageedumetro ← svyby(~ neverwrked, ~ q14age * q13gender * edulevel * metro,
        dsgn, svytotal, na.rm = T)
urtbyageedumetro ← svyby(~ unemployed, by = ~ q14age * q13gender * edulevel * metro,
        denominator = ~ I(employed + unemployed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
nrtbyageedumetro ← svyby(~ narrowU,  by = ~ q14age * q13gender * edulevel * metro,
        denominator = ~ I(narrowU + employed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
vrtbyageedumetro ← svyby(~ neverwrked,  by = ~ q14age * q13gender * edulevel * metro,
        denominator = ~ I(employed + unemployed + nea),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)

#       by ethnic group ("population")
unbyageeth ← svyby(~ unemployed, ~ q14age * q13gender * q15population,
        dsgn, svytotal, na.rm = T)
embyageeth ← svyby(~ employed, ~ q14age * q13gender * q15population,
        dsgn, svytotal, na.rm = T)
neabyageeth ← svyby(~ nea, ~ q14age * q13gender * q15population,
        dsgn, svytotal, na.rm = T)
disbyageeth ← svyby(~ discouraged, ~ q14age * q13gender * q15population,
        dsgn, svytotal, na.rm = T)
nubyageeth ← svyby(~ narrowU, ~ q14age * q13gender * q15population,
        dsgn, svytotal, na.rm = T)
nvrbyageeth ← svyby(~ neverwrked, ~ q14age * q13gender * q15population,
        dsgn, svytotal, na.rm = T)
urtbyageeth ← svyby(~ unemployed, by = ~ q14age * q13gender * q15population,
        denominator = ~ I(employed + unemployed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
nrtbyageeth ← svyby(~ narrowU,  by = ~ q14age * q13gender * q15population,
        denominator = ~ I(narrowU + employed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
vrtbyageeth ← svyby(~ neverwrked,  by = ~ q14age * q13gender * q15population,
        denominator = ~ I(employed + unemployed + nea),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)

unbyageedueth ← svyby(~ unemployed, ~ q14age * q13gender * edulevel * q15population,
        dsgn, svytotal, na.rm = T)
embyageedueth ← svyby(~ employed, ~ q14age * q13gender * edulevel * q15population,
        dsgn, svytotal, na.rm = T)
neabyageedueth ← svyby(~ nea, ~ q14age * q13gender * edulevel * q15population,
        dsgn, svytotal, na.rm = T)
disbyageedueth ← svyby(~ discouraged, ~ q14age * q13gender * edulevel * q15population,
        dsgn, svytotal, na.rm = T)
nubyageedueth ← svyby(~ narrowU, ~ q14age * q13gender * edulevel * q15population,
        dsgn, svytotal, na.rm = T)
```

```
nvrbyageedueth ← svyby(~ neverwrked, ~ q14age * q13gender * edulevel * q15population,
        dsgn, svytotal, na.rm = T)
urtbyageedueth ← svyby(~ unemployed, by = ~ q14age * q13gender * edulevel * q15population
        denominator = ~ I(employed + unemployed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
nrtbyageedueth ← svyby(~ narrowU,  by = ~ q14age * q13gender * edulevel * q15population,
        denominator = ~ I(narrowU + employed),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
vrtbyageedueth ← svyby(~ neverwrked,  by = ~ q14age * q13gender * edulevel * q15populatio
        denominator = ~ I(employed + unemployed + nea),
        design = dsgn, svyratio, vartype = "ci", na.rm = T)
```

```
uba ← data.table(unbyage)
eba ← data.table(embyage)
neba ← data.table(neabyage)
dba ← data.table(disbyage)
nuba ← data.table(nubyage)
nvba ← data.table(nvrbyage)
urtba ← data.table(urtbyage)
nrtba ← data.table(nrtbyage)
vrtba ← data.table(vrtbyage)
uba ← uba[q14age ≥ 15 & q14age ≤ 60]
eba ← eba[q14age ≥ 15 & q14age ≤ 60]
neba ← neba[q14age ≥ 15 & q14age ≤ 60]
dba ← dba[q14age ≥ 15 & q14age ≤ 60]
nuba ← nuba[q14age ≥ 15 & q14age ≤ 60]
nvba ← nvba[q14age ≥ 15 & q14age ≤ 60]
urtba ← urtba[q14age ≥ 15 & q14age ≤ 60]
nrtba ← nrtba[q14age ≥ 15 & q14age ≤ 60]
vrtba ← vrtba[q14age ≥ 15 & q14age ≤ 60]
setkeyv(uba, c("q14age", "q13gender"))
setkeyv(eba, c("q14age", "q13gender"))
setkeyv(neba, c("q14age", "q13gender"))
setkeyv(dba, c("q14age", "q13gender"))
setkeyv(nuba, c("q14age", "q13gender"))
setkeyv(nvba, c("q14age", "q13gender"))
setkeyv(urtba, c("q14age", "q13gender"))
setkeyv(nrtba, c("q14age", "q13gender"))
setkeyv(vrtba, c("q14age", "q13gender"))
ue ← uba[eba]
ue ← ue[neba]
ue ← ue[dba]
ue ← ue[nuba]
ue ← ue[nvba]
ue ← ue[urtba]
ue ← ue[nrtba]
ue ← ue[vrtba]
ue ← ue[, edulevel := "all"]
ubae     ← data.table(unbyageedu)
ebae ← data.table(embyageedu)
nebae ← data.table(neabyageedu)
dbae ← data.table(disbyageedu)
nubae ← data.table(nubyageedu)
nvbae ← data.table(nvrbyageedu)
urtbae ← data.table(urtbyageedu)
nrtbae ← data.table(nrtbyageedu)
```

```
vrtbae ← data.table(vrtbyageedu)
ubae ← ubae[q14age ≥ 15 & q14age ≤ 60]
ebae ← ebae[q14age ≥ 15 & q14age ≤ 60]
nebae ← nebae[q14age ≥ 15 & q14age ≤ 60]
dbae ← dbae[q14age ≥ 15 & q14age ≤ 60]
nubae ← nubae[q14age ≥ 15 & q14age ≤ 60]
nvbae ← nvbae[q14age ≥ 15 & q14age ≤ 60]
urtbae ← urtbae[q14age ≥ 15 & q14age ≤ 60]
nrtbae ← nrtbae[q14age ≥ 15 & q14age ≤ 60]
vrtbae ← vrtbae[q14age ≥ 15 & q14age ≤ 60]
setkeyv(ubae, c("q14age", "q13gender", "edulevel"))
setkeyv(ebae, c("q14age", "q13gender", "edulevel"))
setkeyv(nebae, c("q14age", "q13gender", "edulevel"))
setkeyv(dbae, c("q14age", "q13gender", "edulevel"))
setkeyv(nubae, c("q14age", "q13gender", "edulevel"))
setkeyv(nvbae, c("q14age", "q13gender", "edulevel"))
setkeyv(urtbae, c("q14age", "q13gender", "edulevel"))
setkeyv(nrtbae, c("q14age", "q13gender", "edulevel"))
setkeyv(vrtbae, c("q14age", "q13gender", "edulevel"))
uee ← ubae[ebae]
uee ← uee[nebae]
uee ← uee[dbae]
uee ← uee[nubae]
uee ← uee[nvbae]
uee ← uee[urtbae]
uee ← uee[nrtbae]
uee ← uee[vrtbae]
ubam ← data.table(unbyagemetro)
ebam ← data.table(embyagemetro)
nebam ← data.table(neabyagemetro)
dbam ← data.table(disbyagemetro)
nubam ← data.table(nubyagemetro)
nvbam ← data.table(nvrbyagemetro)
urtbam ← data.table(urtbyagemetro)
nrtbam ← data.table(nrtbyagemetro)
vrtbam ← data.table(vrtbyagemetro)
ubam ← ubam[q14age ≥ 15 & q14age ≤ 60]
ebam ← ebam[q14age ≥ 15 & q14age ≤ 60]
nebam ← nebam[q14age ≥ 15 & q14age ≤ 60]
dbam ← dbam[q14age ≥ 15 & q14age ≤ 60]
nubam ← nubam[q14age ≥ 15 & q14age ≤ 60]
nvbam ← nvbam[q14age ≥ 15 & q14age ≤ 60]
urtbam ← urtbam[q14age ≥ 15 & q14age ≤ 60]
nrtbam ← nrtbam[q14age ≥ 15 & q14age ≤ 60]
vrtbam ← vrtbam[q14age ≥ 15 & q14age ≤ 60]
setkeyv(ubam, c("q14age", "q13gender", "metro"))
setkeyv(ebam, key(ubam))
setkeyv(nebam, key(ubam))
setkeyv(dbam, key(ubam))
setkeyv(nubam, key(ubam))
setkeyv(nvbam, key(ubam))
setkeyv(urtbam, key(ubam))
setkeyv(nrtbam, key(ubam))
setkeyv(vrtbam, key(ubam))
uem ← ubam[ebam]
uem ← uem[nebae]
```

```
uem ← uem[dbae]
uem ← uem[nubae]
uem ← uem[nvbae]
uem ← uem[urtbae]
uem ← uem[nrtbae]
uem ← uem[vrtbae]
```

```
ubat     ← data.table(unbyageeth)
ebat ← data.table(embyageeth)
nebat ← data.table(neabyageeth)
dbat ← data.table(disbyageeth)
nubat ← data.table(nubyageeth)
nvbat ← data.table(nvrbyageeth)
urtbat ← data.table(urtbyageeth)
nrtbat ← data.table(nrtbyageeth)
vrtbat ← data.table(vrtbyageeth)
ubat ← ubat[q14age ≥ 15 & q14age ≤ 60]
ebat ← ebat[q14age ≥ 15 & q14age ≤ 60]
nebat ← nebat[q14age ≥ 15 & q14age ≤ 60]
dbat ← dbat[q14age ≥ 15 & q14age ≤ 60]
nubat ← nubat[q14age ≥ 15 & q14age ≤ 60]
nvbat ← nvbat[q14age ≥ 15 & q14age ≤ 60]
urtbat ← urtbat[q14age ≥ 15 & q14age ≤ 60]
nrtbat ← nrtbat[q14age ≥ 15 & q14age ≤ 60]
vrtbat ← vrtbat[q14age ≥ 15 & q14age ≤ 60]
setkeyv(ubat, c("q14age", "q13gender", "q15population"))
setkeyv(ebat, key(ubat))
setkeyv(nebat, key(ubat))
setkeyv(dbat, key(ubat))
setkeyv(nubat, key(ubat))
setkeyv(nvbat, key(ubat))
setkeyv(urtbat, key(ubat))
setkeyv(nrtbat, key(ubat))
setkeyv(vrtbat, key(ubat))
uet ← ubat[ebat]
uet ← uet[nebae]
uet ← uet[dbae]
uet ← uet[nubae]
uet ← uet[nvbae]
uet ← uet[urtbae]
uet ← uet[nrtbae]
uet ← uet[vrtbae]
```

```
ubaem ← data.table(unbyageedumetro)
ebaem ← data.table(embyageedumetro)
nebaem ← data.table(neabyageedumetro)
dbaem ← data.table(disbyageedumetro)
nubaem ← data.table(nubyageedumetro)
nvbaem ← data.table(nvrbyageedumetro)
urtbaem ← data.table(urtbyageedumetro)
nrtbaem ← data.table(nrtbyageedumetro)
vrtbaem ← data.table(vrtbyageedumetro)
ubaem ← ubaem[q14age ≥ 15 & q14age ≤ 60]
ebaem ← ebaem[q14age ≥ 15 & q14age ≤ 60]
nebaem ← nebaem[q14age ≥ 15 & q14age ≤ 60]
dbaem ← dbaem[q14age ≥ 15 & q14age ≤ 60]
```

```
nubaem ← nubaem[q14age ≥ 15 & q14age ≤ 60]
nvbaem ← nvbaem[q14age ≥ 15 & q14age ≤ 60]
urtbaem ← urtbaem[q14age ≥ 15 & q14age ≤ 60]
nrtbaem ← nrtbaem[q14age ≥ 15 & q14age ≤ 60]
vrtbaem ← vrtbaem[q14age ≥ 15 & q14age ≤ 60]
setkeyv(ubaem, c("q14age", "q13gender", "edulevel", "metro"))
setkeyv(ebaem, key(ubaem))
setkeyv(nebaem, key(ubaem))
setkeyv(dbaem, key(ubaem))
setkeyv(nubaem, key(ubaem))
setkeyv(nvbaem, key(ubaem))
setkeyv(urtbaem, key(ubaem))
setkeyv(nrtbaem, key(ubaem))
setkeyv(vrtbaem, key(ubaem))
ueem ← ubaem[ebaem]
ueem ← ueem[nebaem]
ueem ← ueem[dbaem]
ueem ← ueem[nubaem]
ueem ← ueem[nvbaem]
ueem ← ueem[urtbaem]
ueem ← ueem[nrtbaem]
ueem ← ueem[vrtbaem]
```

```
ubaet ← data.table(unbyageedueth)
ebaet ← data.table(embyageedueth)
nebaet ← data.table(neabyageedueth)
dbaet ← data.table(disbyageedueth)
nubaet ← data.table(nubyageedueth)
nvbaet ← data.table(nvrbyageedueth)
urtbaet ← data.table(urtbyageedueth)
nrtbaet ← data.table(nrtbyageedueth)
vrtbaet ← data.table(vrtbyageedueth)
ubaet ← ubaet[q14age ≥ 15 & q14age ≤ 60]
ebaet ← ebaet[q14age ≥ 15 & q14age ≤ 60]
nebaet ← nebaet[q14age ≥ 15 & q14age ≤ 60]
dbaet ← dbaet[q14age ≥ 15 & q14age ≤ 60]
nubaet ← nubaet[q14age ≥ 15 & q14age ≤ 60]
nvbaet ← nvbaet[q14age ≥ 15 & q14age ≤ 60]
urtbaet ← urtbaet[q14age ≥ 15 & q14age ≤ 60]
nrtbaet ← nrtbaet[q14age ≥ 15 & q14age ≤ 60]
vrtbaet ← vrtbaet[q14age ≥ 15 & q14age ≤ 60]
setkeyv(ubaet, c("q14age", "q13gender", "edulevel", "q15population"))
setkeyv(ebaet, key(ubaet))
setkeyv(nebaet, key(ubaet))
setkeyv(dbaet, key(ubaet))
setkeyv(nubaet, key(ubaet))
setkeyv(nvbaet, key(ubaet))
setkeyv(urtbaet, key(ubaet))
setkeyv(nrtbaet, key(ubaet))
setkeyv(vrtbaet, key(ubaet))
ueet ← ubaet[ebaet]
ueet ← ueet[nebaet]
ueet ← ueet[dbaet]
ueet ← ueet[nubaet]
ueet ← ueet[nvbaet]
ueet ← ueet[urtbaet]
```

```
ueet ← ueet[nrtbaet]
ueet ← ueet[vrtbaet]
```

Programming memo

- In data.table, an assignment operator <- does not copy the object. So b <- a does not created a copy of a until there is an assignment on its element (subassign). It just references a when something is operated on b. So when b is modified, so is a!
- If needed to make an explicit copy, use b = copy(a).

```
uem = copy(ue)
uet = copy(ue)
uem ← uem[, metro := "all"]
uet ← uet[, q15population := "all"]
setkeyv(ue, c("q14age", "q13gender", "edulevel"))
setkeyv(uem, c(key(ue), "metro"))
setkeyv(uee, key(ue))
setkeyv(uet, c(key(ue), "q15population"))
setkeyv(ueem, key(uem))
setkeyv(ueet, key(uet))
ur ← rbind(ue, uee)
urm ← rbind(uem, ueem)
urt ← rbind(uet, ueet)
setkeyv(ur, key(ue))
setkeyv(urm, key(uem))
setkeyv(urt, key(uet))
ii ← grep("se", colnames(ur))
setnames(ur, colnames(ur)[ii], paste("se", colnames(ur)[ii-1], sep = "."))
ii ← grep("ci_l", colnames(ur))
setnames(ur, colnames(ur)[ii], paste("ci1", colnames(ur)[ii-1], sep = "."))
ii ← grep("ci_u", colnames(ur))
setnames(ur, colnames(ur)[ii], paste("ci2", colnames(ur)[ii-2], sep = "."))
setnames(ur, grepout("^une.*\\(", colnames(ur)), "urate")
setnames(ur, grepout("^ci1.une.*\\(", colnames(ur)), "ci1.urate")
setnames(ur, grepout("^ci2.une.*\\(", colnames(ur)), "ci2.urate")
setnames(ur, grepout("^na.*\\(", colnames(ur)), "nurate")
setnames(ur, grepout("^ci1.na.*\\(", colnames(ur)), "ci1.nurate")
setnames(ur, grepout("^ci2.na.*\\(", colnames(ur)), "ci2.nurate")
setnames(ur, grepout("^nev.*\\(", colnames(ur)), "neverrate")
setnames(ur, grepout("^ci1.nev.*\\(", colnames(ur)), "ci1.neverrate")
setnames(ur, grepout("^ci2.nev.*\\(", colnames(ur)), "ci2.neverrate")
ii ← grep("se", colnames(urm))
setnames(urm, colnames(urm)[ii], paste("se", colnames(urm)[ii-1], sep = "."))
ii ← grep("ci_l", colnames(urm))
setnames(urm, colnames(urm)[ii], paste("ci1", colnames(urm)[ii-1], sep = "."))
ii ← grep("ci_u", colnames(urm))
setnames(urm, colnames(urm)[ii], paste("ci2", colnames(urm)[ii-2], sep = "."))
setnames(urm, grepout("^une.*\\(", colnames(urm)), "urate")
setnames(urm, grepout("^ci1.une.*\\(", colnames(urm)), "ci1.urate")
setnames(urm, grepout("^ci2.une.*\\(", colnames(urm)), "ci2.urate")
setnames(urm, grepout("^na.*\\(", colnames(urm)), "nurate")
setnames(urm, grepout("^ci1.na.*\\(", colnames(urm)), "ci1.nurate")
setnames(urm, grepout("^ci2.na.*\\(", colnames(urm)), "ci2.nurate")
setnames(urm, grepout("^nev.*\\(", colnames(urm)), "neverrate")
setnames(urm, grepout("^ci1.nev.*\\(", colnames(urm)), "ci1.neverrate")
```

```
setnames(urm, grepout("^ci2.nev.*\\(", colnames(urm)), "ci2.neverrate")
ii ← grep("se", colnames(urt))
setnames(urt, colnames(urt)[ii], paste("se", colnames(urt)[ii-1], sep = "."))
ii ← grep("ci_l", colnames(urt))
setnames(urt, colnames(urt)[ii], paste("ci1", colnames(urt)[ii-1], sep = "."))
ii ← grep("ci_u", colnames(urt))
setnames(urt, colnames(urt)[ii], paste("ci2", colnames(urt)[ii-2], sep = "."))
setnames(urt, grepout("^une.*\\(", colnames(urt)), "urate")
setnames(urt, grepout("^ci1.une.*\\(", colnames(urt)), "ci1.urate")
setnames(urt, grepout("^ci2.une.*\\(", colnames(urt)), "ci2.urate")
setnames(urt, grepout("^na.*\\(", colnames(urt)), "nurate")
setnames(urt, grepout("^ci1.na.*\\(", colnames(urt)), "ci1.nurate")
setnames(urt, grepout("^ci2.na.*\\(", colnames(urt)), "ci2.nurate")
setnames(urt, grepout("^nev.*\\(", colnames(urt)), "neverrate")
setnames(urt, grepout("^ci1.nev.*\\(", colnames(urt)), "ci1.neverrate")
setnames(urt, grepout("^ci2.nev.*\\(", colnames(urt)), "ci2.neverrate")
```

Note that

$$f(a,b,c) \simeq f(\mu_a,\mu_b,\mu_c) + f_a \cdot (a-\mu_a) + f_b \cdot (b-\mu_b) + f_c \cdot (c-\mu_c),$$

hence, noting $\mathcal{V}[f] = \mathcal{E}[f^2 - \bar{f}^2]$,

$$\begin{aligned}
\mathcal{V}[f(a,b,c)] &\simeq \mathcal{V}\left[f_a \cdot (a-\mu_a) + f_b \cdot (b-\mu_b) + f_c \cdot (c-\mu_c)\right] \\
&= \mathcal{E}\left[\{f_a \cdot (a-\mu_a) + f_b \cdot (b-\mu_b) + f_c \cdot (c-\mu_c)\}^2\right] \\
&= f_a^2(\mu_a,\mu_b,\mu_c)\mathcal{V}[a] + f_b^2(\mu_a,\mu_b,\mu_c)\mathcal{V}[b] + f_c^2(\mu_a,\mu_b,\mu_c)\mathcal{V}[c] \\
&\quad + 2\,f_a(\mu_a,\mu_b,\mu_c)\,f_b(\mu_a,\mu_b,\mu_c)\mathrm{cov}[a,b] + 2\,f_a(\mu_a,\mu_b,\mu_c)\,f_c(\mu_a,\mu_b,\mu_c)\mathrm{cov}[a,c] \\
&\quad + 2\,f_b(\mu_a,\mu_b,\mu_c)\,f_c(\mu_a,\mu_b,\mu_c)\mathrm{cov}[b,c].
\end{aligned}$$

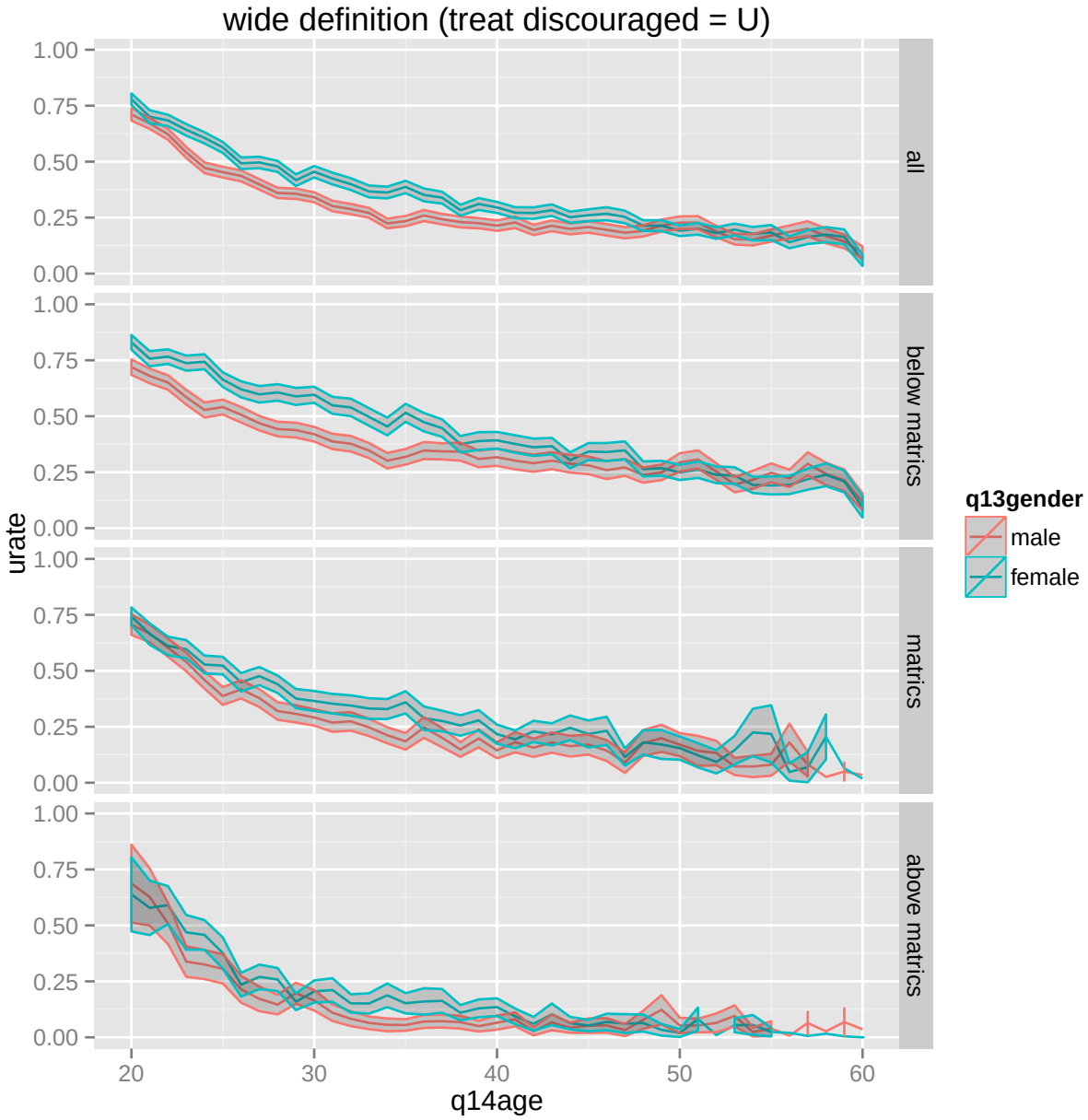
```
#       compute rates and their standard errors
ur ← ur[, urate := unemployed / (unemployed + employed)]
ur ← ur[, nurate := narrowU / (narrowU + employed)]
ur ← ur[, neverrate := neverwrked / (unemployed + employed + nea)]
# serate ← function(a, b, se.a, se.b) {
# fa ← 1 / mean(a + b) - mean(a) / (mean(a) + mean(b))^2
# fb ← - mean(a) / (mean(a) + mean(b))^2
# fa^2 * se.a^2 + fb^2 * se.b^2 + 2*fa * fb * cov(a, b)
# }
# serate2 ← function(a, b, c, d, se.a, se.b, se.c, se.d) {
# fa ← 1 / mean(b + c + d)
# fb ←  fc ← fd ← - mean(a) / (mean(b) + mean(c) + mean(d))^2
# fa^2 * se.a^2 + fb^2 * se.b^2 + fc^2 * se.c^2 + fd^2 * se.d^2 +
# 2*fa * fb * cov(a, b) + 2*fa * fc * cov(a, c) +
# 2*fa * fd * cov(a, d) + 2*fb * fc * cov(b, c) +
# 2*fb * fd * cov(b, d) + 2*fc * fd * cov(c, d)
# }
# ur ← ur[, se.urate := serate(unemployed, employed, se.unemployed, se.employed)]
# ur ← ur[, se.nurate := serate(narrowU, employed, se.narrowU, se.employed)]
# ur ← ur[, se.neverrate := serate2(neverwrked, unemployed, employed, nea,
#       se.neverwrked, se.unemployed, se.employed, se.nea)]
urm ← urm[, urate := unemployed / (unemployed + employed)]
urm ← urm[, nurate := narrowU / (narrowU + employed)]
urm ← urm[, neverrate := neverwrked / (unemployed + employed + nea)]
urt ← urt[, urate := unemployed / (unemployed + employed)]
```

```
urt ← urt[, nurate := narrowU / (narrowU + employed)]
urt ← urt[, neverrate := neverwrked / (unemployed + employed + nea)]
write.tablev(ur, "ur_survey_estimates.prn")
write.tablev(urm, "urm_survey_estimates.prn")
write.tablev(urt, "urt_survey_estimates.prn")
```

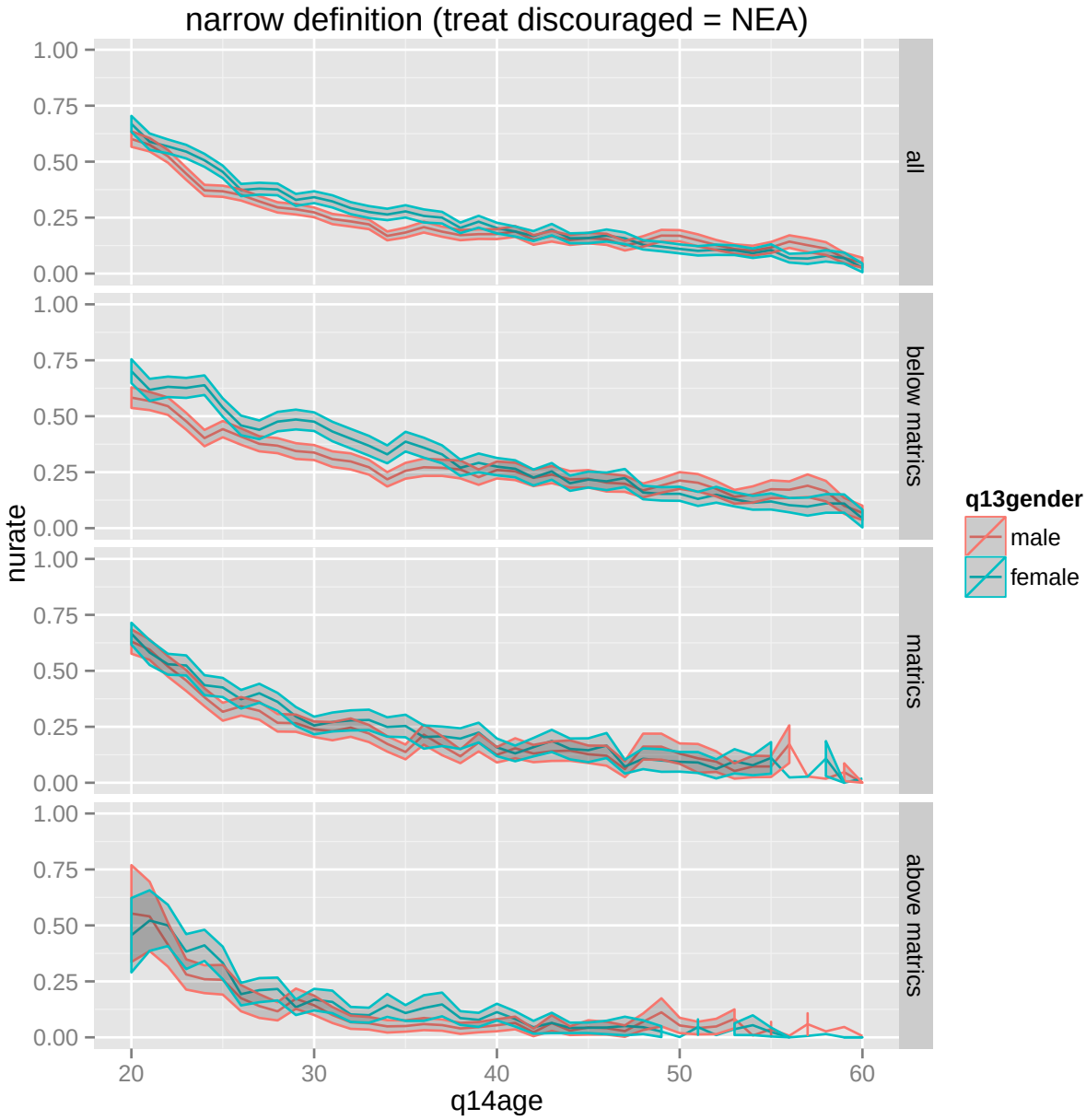
IV　Plotting various rates by group

From plots of unemployment rates (wider definition), we can see that, for the entire population, we see high unemployment rates at the age 20 of around 70% and 80%, for males and females, respectively. It takes almost 7 years for the males and 12 years for females for these rates to come down below 50%. When we plot the rates separately by educational qualification, we find individuals with educational qualification below matriculation have persistent unemployment rates both for males and females. For males, it takes about 10 years to come down below 50%, for females it takes almost 20 years. Having matriculation helps, but the additional gain in reducing unemployment rates is relatively small compared with the individuals with above matriculation qualification. While it takes about 2 to 3 years for males to go below 50% unemployment rates, it takes about 5 years for males with matriculation qualification. Plots of narrowly defined unemployment rates give similar observations, except that the level of unemployment rates are lower than with the wider definition.

```
library(ggplot2)
p ← ggplot(ur, aes(x = q14age, y = urate, group = q13gender, color = q13gender))
p ← p + geom_line() + ggtitle("wide definition (treat discouraged = U)") +
geom_ribbon(aes(ymin = ci2.urate, ymax = ci1.urate), alpha = .2) +
scale_y_continuous(limits = c(0, 1)) + scale_x_continuous(limits = c(20, 60))
p + facet_grid(edulevel ~ .)
```

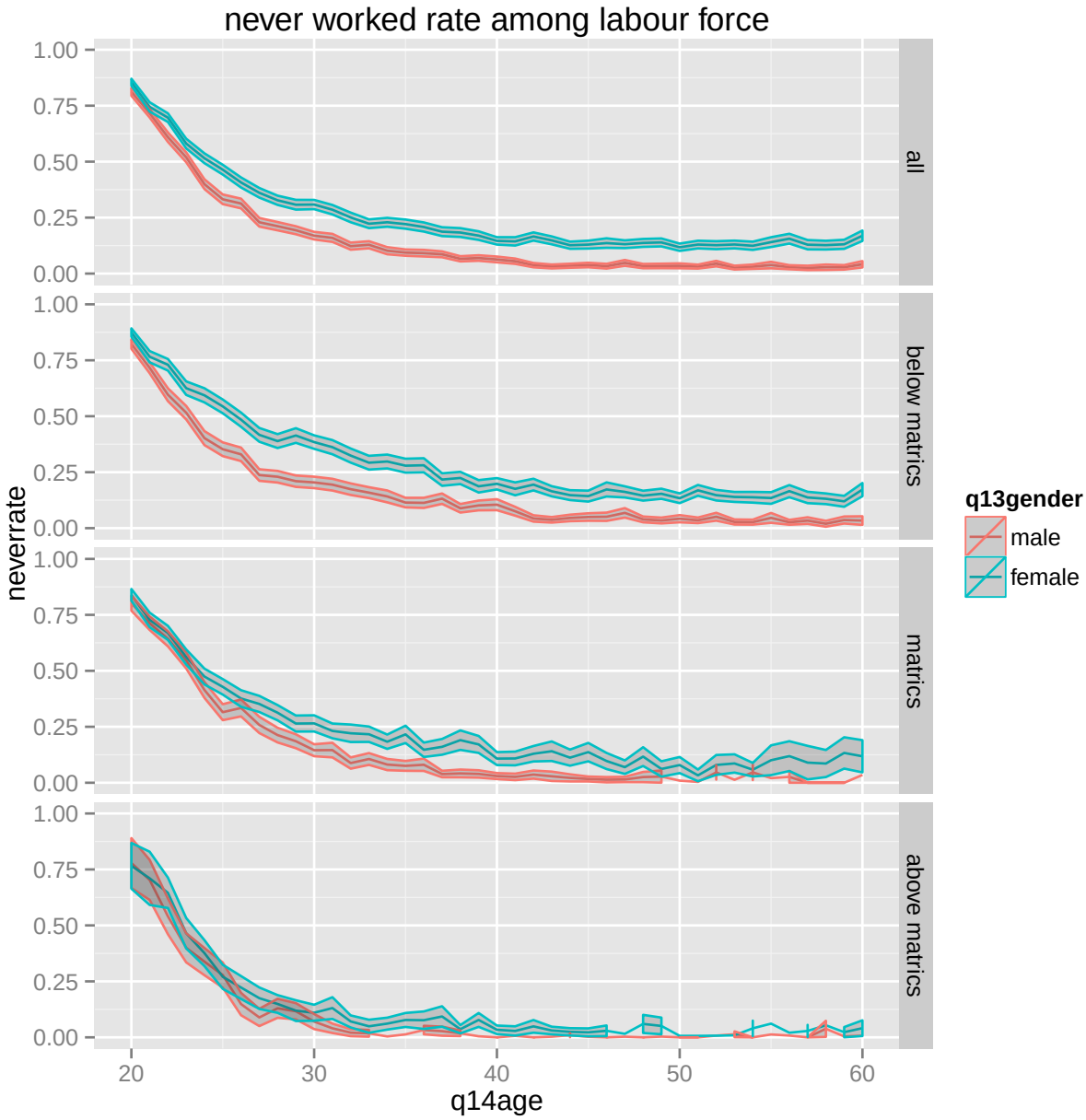



```
p ← ggplot(ur, aes(x = q14age, y = nurate, group = q13gender, color = q13gender))
p ← p + geom_line() + ggtitle("narrow definition (treat discouraged = NEA)") +
geom_ribbon(aes(ymin = ci2.nurate, ymax = ci1.nurate), alpha = .2) +
scale_y_continuous(limits = c(0, 1)) + scale_x_continuous(limits = c(20, 60))
p + facet_grid(edulevel ~ .)
```



The never worked rate, defined by the ratio of total affirmative responses to "have you ever worked before?" (Q312everwrk) sum of unemployed, employed, and not economically active individuals, gives interesting plots. For males, regardless of educational quaifications, the proportion of people who never worked rapidly decreases and reaches to around 20% at the age of 30. This is not the same with females where the individuals below matriculation qualification tend to stay out of the work longer, and 40% of them have not worked until the age of 30. This is most likely to be associated with pregnancy and child rearing, as most of high school dropouts tend to give birth before matriculation.

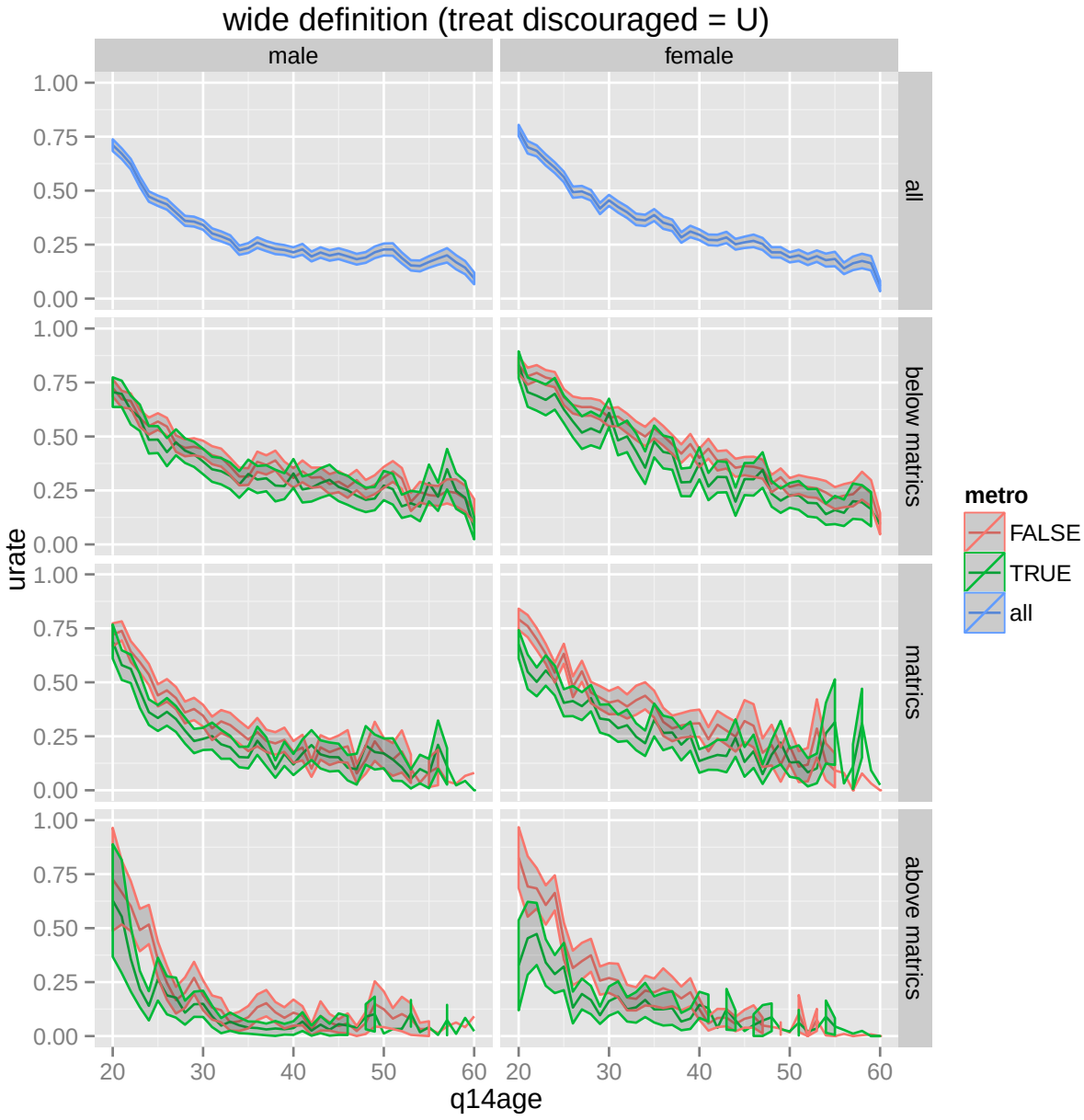
```
p ← ggplot(ur, aes(x = q14age, y = neverrate, group = q13gender, color = q13gender))
p ← p + geom_line() + ggtitle("never worked rate among labour force") +
geom_ribbon(aes(ymin = ci2.neverrate, ymax = ci1.neverrate), alpha = .2) +
scale_y_continuous(limits = c(0, 1)) + scale_x_continuous(limits = c(20, 60))
p + facet_grid(edulevel ~ .)
```



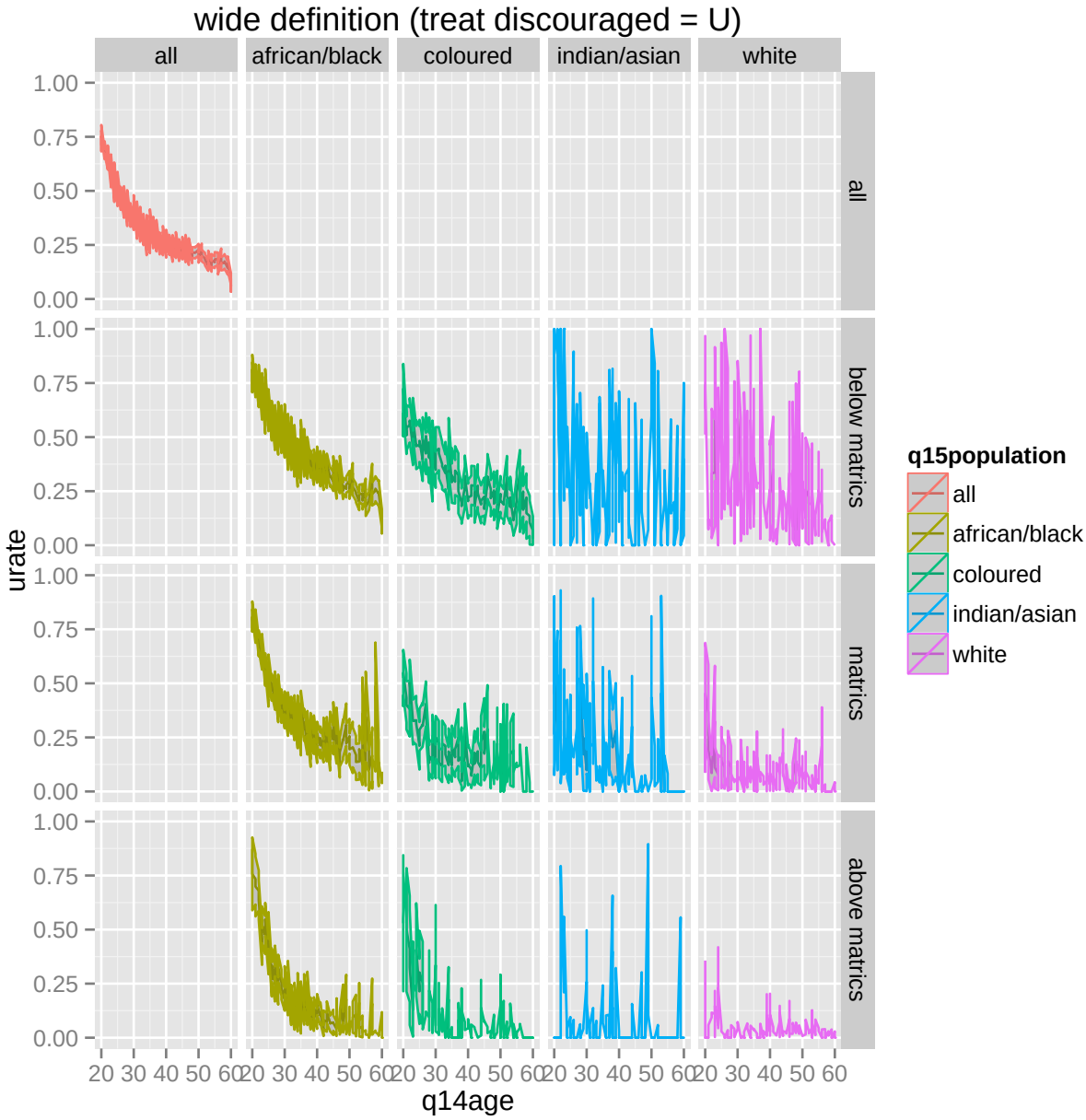
In the below plots, we show the unemployment rates by age, education, and geographical areas. We divide the sample into metro and non-metro regions*. For males below matriculation, we see little difference in unemployment rates regardless of the area they reside. Living closer to the metropolican areas does not increase the likelihood of getting a job significantly. If with matriculation qualification, we see a considerable gap of around 20% between metro and non-metro areas. For males above matriculation qualification, we see a rapid decline in unemployment rates if they live in metro areas. For non-metro males with above matriculation qualification, the reduction is slower but become rapid after passing the age of 25 and the rates become statistically indistinguishable with the counterpart group living in metro areas. Except for higher unemployment rates than males, females have similar patterns with males that geographic areas does not matter for below matriculation. It is only after matriculation that we see the difference between metro and non-metro population.

* Metro regions refer to Cape Town, eThekweni, edkhurhuleni, Johannesburg, Nelson Mandela Metro, Tshwane.

```
p ← ggplot(urm, aes(x = q14age, y = urate, group = metro, color = metro))
p ← p + geom_line() + ggtitle("wide definition (treat discouraged = U)") +
geom_ribbon(aes(ymin = ci2.urate, ymax = ci1.urate), alpha = .2) +
scale_y_continuous(limits = c(0, 1)) + scale_x_continuous(limits = c(20, 60))
p + facet_grid(edulevel ~ q13gender)
```




```
p ← ggplot(urt, aes(x = q14age, y = urate, group = q15population, color = q15population))
p ← p + geom_line() + ggtitle("wide definition (treat discouraged = U)") +
geom_ribbon(aes(ymin = ci2.urate, ymax = ci1.urate), alpha = .2) +
scale_y_continuous(limits = c(0, 1)) + scale_x_continuous(limits = c(20, 60))
p + facet_grid(edulevel ~ q15population)
```


wide definition (treat discouraged = U)
all
african/black
coloured
indian/asian
white
below matrics
matrics
above matrics
urate
1.00
0.75
0.50
0.25
0.00
20 30 40 50 60
q14age
q15population
all
african/black
coloured
indian/asian
white

参考文献

Acemoglu, Daron. 2001. “Good jobs versus bad jobs.” Journal of labor Economics, 19(1): 1–21.

Granovetter, Mark. 1983. “The strength of weak ties: A network theory revisited.” Sociological Theory, 1(1): 201–233.

Granovetter, Mark. 2005. “The impact of social structure on economic outcomes.” Journal of economic perspectives, 33–50.

Lumley, Thomas. 2014. “survey: analysis of complex survey samples.” R package version 3.30.

Navarro, Lucas. 2007. “Labor market policies in a sector specific search model with heterogeneous firms and workers.” Revista de Análisis Economico–Economic Analysis Review, 22(2): 29–45.

Petrongolo, Barbara, and Christopher A. Pissarides. 2001. “Looking into the Black Box: A Survey of the Matching Function.” Journal of Economic Literature, 39: 390–431.

Pissarides, Christopher A. 1985. “Short-run equilibrium dynamics of unemployment, vacancies, and real wages.” The American Economic Review, 676–690.

Pissarides, Christopher A. 2000. Equilibrium unemployment theory. MIT Press, Cambridge.

Statistics South Africa. 2008. Guide to Quarterly Labour Force Survey. Statistics South Africa.

第 2 章

Survey sampling and outcomes

March 9, 2015

Seiro Ito

Abstract In this chapter, we describe briefly about the sample size calculations and preliminary results of labour market status in the surveyed households. We will employ knitr to make the analysis reproducible. In the first section, we describe the sampling frame and sample size calculations. In the second section, we summarise the labour market status of the respondents.

Key Words Sampling, unemployment

I Sampling

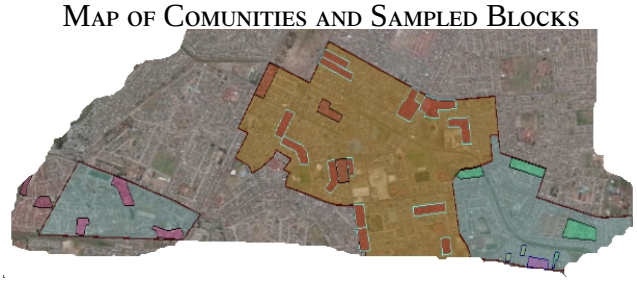
I.1 Sampling frame

Based on the understanding of youth unemployment in South Africa, we have decided that population is adults aged 20-35 in communities in Cape Town area. We have discussed if female population with small children may be dropped from survey sample but not from lab sample. However, this was ruled out since it takes an extra instruction and complications in data collection.

With the small sample size (or relatively small budget size), we did not attempt to get a representative sample. We purposefully chose only a few communities in Cape Town area. Stratification is made by taking into the account that the efficiency gain is related to differences in strata means. We have thus defined the strata as majority language groups in communities. Majority language group in communities can be grouped into Xhosa and Afrikaans. We took one community each from Xhosa and Afrikaans communities. Since this process was not randomised, we wish to claim almost nill external validity of future findings.

To make sampling operational, we further stratified the community into sections or wards. For a Xhosa speaking community, it turned out that councillors from some wards did not wish to give a permission for a survey without us paying a security fee. Because of limited budget and uncertainty in the quality of security services they wanted to provide, we skipped these wards. Hence the wards became primary sampling units in this community. Given that the permission was not granted for

Map of Comunities and Sampled Blocks



some wards, this process was also not randomised. In the Afrikaans community, sections are strata with the same probability weight.

In sampling households, we used residential blocks as clusters which we randomly chose (see the map). So the blocks are primary sampling units for Afrikaans speaking community and secondary sampling units for the Xhosa speaking community. Given a residential block, we employed systematic sampling. We started from a reference house and sampled every n-th house. We chose n for each cluster differently so we end up having approximately six houses per block. To do so, we counted all housing before sampling. In selecting household members, we identified the target individual members, and made an appointment. We then interviewed the household head or spouse to fill in the cover page and roster.

For informal settlements, systematic sampling was averted as it is difficult to define "the house next to a house" given its unstructured way of residential placement. We used satellite images, stratify the area to a small number of blocks, and sampled houses by generating random numbers.

I.2 Sampling theory considerations

In this subsection, we consider a simplified version of sample size calculation. In theory, until all strata reach the minimum visit size, one can keep on systematic sampling, while recording all HH types. Keeping all HH types is crucial in computing the inclusion probability $p_{ij} = \frac{n_{ij}}{n}$, $n = \Sigma n_{ij}$. For simplicity, use proportional allocation where inclusion probability $\pi_h = \frac{n_h}{N_h} = \pi$ is same across strata h. The grand mean is given by

$$\bar{y} = \sum_{h=1}^{H} \frac{N_h}{N} \bar{y}_h.$$

Variance of grand mean is

$$\mathcal{V}[\bar{y}] = \sum_{h=1}^{H} \left(1 - \tfrac{n_h}{N_h}\right) N_h^2 \frac{s_h^2}{n_h}, \quad s_h^2 = \sum_{i=1}^{n_h} \frac{(y_{hi} - \bar{y}_h)^2}{n_h - 1}.$$

Note that stratified sampling will make variance no larger than SRS (in almost all cases).

$$\mathcal{V}[\bar{y}_{str}] \leqslant \frac{1}{n} \sum_{h=1}^{H} \frac{n}{n_h} \left(\frac{N_h}{N}\right)^2 S_h^2 = \frac{v}{n}, \quad v = \sum_{h=1}^{H} \frac{n}{n_h} \left(\frac{N_h}{N}\right)^2 S_h^2.$$

If FPC is ignored (small n_h relative to N_h) and normal approximation is valid,

$$CI(\bar{y}_{str}) = \pm z_{\alpha/2} \sqrt{\tfrac{v}{n}}.$$

Then setting a desired error margin e at α corresponding to CI_α as:

$$e = z_{\alpha/2}\sqrt{\frac{v}{n}}$$

or inverting it will give the sample size

$$n = z^2_{\alpha/2}\frac{v}{e^2}.$$

If $\alpha = .05$, then $z_{.025} = 1.96$ or $z^2_{.025} = 3.8416$ and $n = 3.8416\frac{v}{e^2}$

- Example: Suppose that there are 100 people, of which 1/2 has employment days in the past 6 months given by `rnbinom(n = 50, size = 20, prob = .65) * 6`, the rest is `rnbinom(n = 50, size = 15, prob = .80) * 6` and replace with 0 if days are smaller than 30, the variances of two strata are below:

```
x1 ← rnbinom(n = 50, size = 20, prob = .65) * 6
x2 ← rnbinom(n = 50, size = 15, prob = .80) * 6
x1[x1 < 30] ← 0; x2[x2 < 30] ← 0
destat(cbind(x1, x2))
```

```
   min 25\\% median 75\\% max mean  std 0s NAs  n
x1   0    48     60    78 114 62.8 26.3  3   0 50
x2   0     0      0    30  54 13.1 18.1 32   0 50
```

```
var(x1); var(x2); var(c(x1, x2))
```

```
[1] 689.4514
```

```
[1] 328.6873
```

```
[1] 1127.185
```

For the employed days (continuous outcomes), assuming equal strata size $N_h = 1000$ and equal stratum sample size $n_h = 50$, with population stratum variance S^2_h measured as above and $H = 2$, we have:

$$v = \sum_{h=1}^{2}\frac{100}{50}\left(\frac{1000}{2000}\right)^2 S^2_h = (2)\cdot(.25)\cdot(S^2_1 + S^2_2) = \frac{S^2_1 + S^2_2}{2}.$$

So with $e = 5$ days,

$$n = 3.8416\frac{S^2_1 + S^2_2}{50} = .077(S^2_1 + S^2_2),$$

which is about 100 at most.

lohr3.3 is a function to compute the sample size given the stratum variances. vee is a function to compute stratum variances.

```
vee ← function(S2h, nh, Nh, H = NULL) {
  nhlen ← length(nh); S2hlen ← length(S2h); Nhlen ← length(Nh)
  if (nhlen != Nhlen) stop("length(nh) must be same as length(Nh).")
  if (nhlen != S2hlen) stop("length(nh) must be same as length(S2h).")
  if (Nhlen != S2hlen) stop("length(Nh) must be same as length(S2h).")
  if (is.null(H)) if(nhlen > 1) H ← nhlen else stop("H is NULL and length(nh) is 1.")
  if (nhlen > 1) {
```

```
    n ← sum(nh)
    N ← sum(Nh)
  } else {
    n ← nh * H
    N ← Nh * H
    S2h ← rep(S2h, nhlen)
  }
  sum((n/nh) * (Nh/N)^(2) * S2h)
}
lohr3.3 ← function(e, alpha = .05, ve = NULL, S2h, nh, Nh, H = NULL) {
  if (is.null(ve)) v ← vee(S2h, nh, Nh, H = NULL) else v ← ve
  zee ← qnorm(1-alpha/2)
  zee^(2) * v / e^(2)
}
# example variance
fakesample ← c(x1, x2)
es2h ← var(fakesample)
es2h
```

```
[1] 1127.185
```

```
eee ← 5
v ← vee(S2h = c(var(x1), var(x2)),
  nh = c(length(x1), length(x2)), Nh = c(length(x1), length(x2))*1000)
v
```

```
[1] 509.0694
```

```
lohr3.3(eee, ve = v)
```

```
[1] 78.22276
```

```
lohr3.3(eee, S2h = c(var(x1), var(x2)),
  nh = c(length(x1), length(x2)), Nh = c(length(x1), length(x2))*1000)
```

```
[1] 78.22276
```

- Another scenario: Continue working on the two group example. Let the number of strata to be 10, with random number generated by `rnbinom(n = 50, size = `s_h`, prob = `p_h`) * 6` with s_h, p_h given as:

```
sh ← seq(10, 30, length.out = 11); ph ← seq(.75, .6, length.out = 11)
sh; ph
```

```
 [1] 10 12 14 16 18 20 22 24 26 28 30
```

```
 [1] 0.750 0.735 0.720 0.705 0.690 0.675 0.660 0.645 0.630 0.615 0.600
```

Then we can summarise how the sample size changes as below. This also results in similar sample size of around 100.

```
for (h in 1:11) {
if (h == 1) xh ← rnbinom(n = 50, size = sh[h], prob = ph[h]) * 6 else
        xh ← cbind(xh, rnbinom(n = 50, size = sh[h], prob = ph[h]) * 6)
```

```
}
colnames(xh) ← paste0("x", 1:length(sh))
xh[xh < 30] ← 0
destat(xh)
```

```
    min 25\\% median 75\\% max  mean  std 0s NAs  n
x1    0   0.0      0   0.0  42   5.3 12.3 42   0 50
x2    0   0.0     15  36.0  72  20.8 22.7 25   0 50
x3    0   0.0     33  46.5  84  30.1 24.1 16   0 50
x4    0   0.0     36  48.0  96  33.7 26.4 15   0 50
x5    0  36.0     48  60.0 114  49.8 23.9  4   0 50
x6    0  42.0     60  84.0 126  62.0 26.3  2   0 50
x7   30  54.0     69  84.0 108  68.6 18.1  0   0 50
x8    0  60.0     78  96.0 162  76.7 28.6  1   0 50
x9   42  73.5     90 114.0 198  95.3 32.3  0   0 50
x10  42  84.0    108 126.0 210 109.2 34.6  0   0 50
x11  54  97.5    117 150.0 198 123.6 37.8  0   0 50
```

```
v ← vee(S2h = apply(xh, 2, var),
  nh = rep(nrow(xh), ncol(xh)),
  Nh = rep(nrow(xh), ncol(xh))*1000)
v
```

```
[1] 728.974
```

```
lohr3.3(eee, ve = v)
```

```
[1] 112.0129
```

I.3 Simple sample size and power calculations: One-stage cluster sampling

In this subsection, we consider sample size calculation under cluster sampling. Denote the clusters with $j = 1, \ldots, J$ and individuals within clusters as $i = 1, \ldots, N_j$ for each j. The total number of individuals are $N = \sum_{j=1}^{J} N_j$. Denote x_{ij} as the variable of interest for an individual i in cluster j. We sample m clusters which we denote with the set C.

In a one-stage cluster sampling, we sample all individual units in a chosen cluster. If we want the total sum q of it, say, production, the total in each cluster is simply $q_j = \sum_{i=1}^{N_j} x_{ij}$.

- Total: We can estimate the population total by multiplying the total of sampled clusters with the ratio of total clusters to sampled clusters:

$$\hat{q} = \frac{J}{m} \sum_{j \in C} q_j.$$

The variance estimator for this estimator is

$$\mathcal{V}[\hat{q}] = N \sqrt{\left(1 - \frac{m}{J}\right) \frac{s^2}{m}}$$

If we have the total number of individuals N, we can use:

$$\hat{q} = \frac{N}{\sum\limits_{j \in C} N_j} \sum_{j \in C} q_j.$$

(Or, I think we can also take the simple average of individual cluster estimates)

$$\hat{q} = \frac{1}{m}\sum_{j\in C}\frac{N}{N_j}q_j = \frac{N}{m}\sum_{j\in C}\bar{x}_j = N\bar{x},$$

where $\bar{x}$ is a simple average of cluster averages

$$\bar{x} = \frac{1}{m}\sum_{j\in C}\bar{x}_j.$$

- Mean: Population mean is given by $\frac{\sum_{j=1}^{J} q_j}{\sum_{j=1}^{J} N_j} = \frac{q}{N}$. Its estimator is:

$$\bar{x} = \frac{\hat{q}}{\hat{N}} = \frac{\frac{J}{m}\sum\limits_{j\in C} q_j}{\frac{J}{m}\sum\limits_{j\in C} N_j} = \frac{\sum\limits_{j\in C} q_j}{\sum\limits_{j\in C} N_j} = \frac{\sum\limits_{i\in C} N_j\bar{x}_j}{\sum\limits_{j\in C} N_j}.$$

Recall from (4.10) that the variance of ratio estimator $\hat{B} = \frac{\hat{a}}{\hat{b}}$ is:

$$\mathcal{V}\left[\hat{B}\right] = \frac{J-m}{JN\bar{b}^2}s^2,$$

where

$$s^2 = \frac{1}{N-1}\sum_{j\in C}\sum_{i\in C}\left(a_{ij} - \hat{B}b_{ij}\right)^2.$$

Then variance of mean estimator is:

$$\mathcal{V}\left[\bar{x}\right] = \frac{J-m}{JN\bar{N}^2}s^2$$

where

$$\hat{B} = \frac{\hat{q}}{\hat{N}}, \quad s^2 = \frac{1}{m-1}\sum_{j\in C}\left(q_j - \bar{B}N_j\right)^2, \quad \bar{N} = \frac{1}{m}\sum_{j\in C}N_j.$$

Here we see that, to decrease the variance of mean, we would want to increase numer of clusters m, and make per cluster sample size N_j as uniform as possible.

I.4 Sample size for continuous outcomes under SRS

In this subsection, we consider sample size calculation under simple random sampling for continuous outcomes. If we stratify the sample and we assume that distributions are different between strata, then the following will apply to each strata sample size, not the entire sample size.

Suppose we want to estimate the mean of data $x_1, \ldots, x_N$ of iid random sample whose mean is θ

and variance is σ_X^2. The unbiased mean estimator $\bar{x}$ has a variance:

$$\begin{aligned}
\mathcal{V}[\bar{x}] = \mathcal{E}[(\bar{x}-\theta)^2] &= \mathcal{E}\left[\left(\sum_i^N w_i x_i - \theta\right)^2\right], \\
&= \mathcal{E}\left[\left(\sum_i^N w_i (x_i - \theta)\right)^2\right], \\
&= \mathcal{E}\left[\left(\sum_i^N w_i (x_i - \theta)\right)\left(\sum_j^N w_j (x_j - \theta)\right)\right], \\
&= \mathcal{E}\left[\sum_i^N \sum_j^N w_i w_j \left(x_i - \theta\right)\left(x_j - \theta\right)\right], \\
&= \mathcal{E}\left[\sum_i^N w_i^2 \left(x_i - \theta\right)^2\right],
\end{aligned}$$

as x_i and x_j has a zero covariance. So if $w_i = \frac{1}{N}$,

$$\mathcal{V}[\bar{x}] = \frac{1}{N^2}\sum_i^N \sigma_X^2 = \frac{\sigma_X^2}{N},$$

so the standard error (standard deviation) of $\bar{x}$ is $\frac{\sigma_X}{\sqrt{N}}$. This, in turn, gives the sample size to achieve the target value of standard error. Suppose that one wants to attain the standard error to be no more than .5. Then $\sqrt{\mathcal{V}[\bar{x}]} = \frac{\sigma_X}{\sqrt{N}} \leqslant .5$, or:

$$N \geqslant \frac{\sigma_X^2}{.25} = 4\sigma_X^2.$$

So if we know the variance of X, say, assume it to be 100, then $N \geqslant 400$.

If we want a 80% power to distinguish θ from θ_0, or being able to test $\theta > \theta_0$ for some value θ_0, we need:

$$N \geqslant 2.8^2 \frac{\sigma_X^2}{(\theta - \theta_0)^2} = \left(2.8\frac{\sigma_X}{d}\right)^2,$$

where $d = \theta - \theta_0 > 0$ is an effect size. The multiplication factor 2.8 comes from the following

- Under a 95% CI, to have $\theta > \theta_0$, $\theta - \theta_0$ has to be $1.96\sigma_X$ away from zero (asymptotically).
- Assuming $\theta - \theta_0$ is normally distributed, for a 80% right tail at $1.96\sigma_X$ point, the true value of $\theta - \theta_0$ needs to be centered at $2.8\sigma_X$.

If we want to test $\bar{x_1} - \bar{x_2} = 0$, it follows that $\mathcal{V}[\bar{x_1} - \bar{x_2}] = \frac{\sigma_{X_1}^2}{N_1} + \frac{\sigma_{X_2}^2}{N_2}$ if we assume independent sampling between X_1 and X_2. To achieve the variance of $v > 0$ when $\frac{N_1}{N_2} = \frac{a}{1-a}$,

$$\frac{\frac{\sigma_{X_1}^2}{aN} + \frac{\sigma_{X_2}^2}{(1-a)N}}{N^2} = \frac{\frac{\sigma_{X_1}^2}{a} + \frac{\sigma_{X_2}^2}{(1-a)}}{N} \leqslant v,$$

or

$$N \geqslant \frac{1}{v}\left(\frac{\sigma_{X_1}^2}{a} + \frac{\sigma_{X_2}^2}{1-a}\right).$$

To detect an effect size of $d > 0$ at a 80% probability, we have:

$$N \geqslant \left(\frac{2.8}{d}\right)^2 \left(\frac{\sigma_{X_1}^2}{a} + \frac{\sigma_{X_2}^2}{1-a}\right).$$

If $a = \frac{1}{2}$ and $\sigma_{X_1}^2 = \sigma_{X_2}^2 = \sigma_X^2$, then $N \geqslant 4\left(\frac{2.8}{d}\right)^2 \sigma_X^2 = \left(\frac{5.6}{d}\right)^2 \sigma_X^2$.

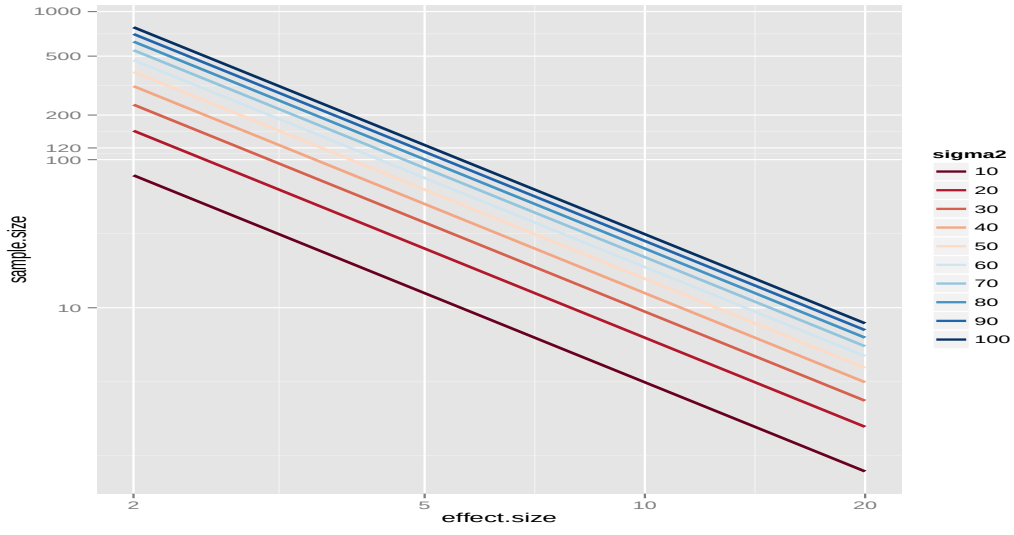
I.5 Simple sample size simulation for continuous outcomes under SRS

In this subsection, we simulate sample size calculation under SRS for continuous outcomes. Assume:

- $\sigma_X^2 = 10, 20, \ldots, 100$.
- $d = 2, 3, \ldots, 20$.

Note that the minimum sample size reduces if using covariates and having a multi period panel to control for individual fixed effects. Based on simulation outcomes, it looks like $N_j > 120 = 60 + 60$ is sufficient (for $d > 5$) at 80% power.

```
en ← NULL
for (sig in seq(10, 100, 10)) for (es in seq(2, 20, 1))
        en ← rbind(en, c(sig, es, (5.6/es)^{2}*sig))
colnames(en) ← c("sigma2", "effect.size", "sample.size")
en ← data.frame(en)
en[, "sigma2"] ← factor(en[, "sigma2"])
library(ggplot2)
p ← ggplot(en, aes(x = effect.size, y = sample.size))
p + geom_line(aes(group = sigma2, color = sigma2), size = 1) +
        #scale_y_log10(breaks = seq(50, 2000, 100)) +
        scale_y_log10(breaks = c(10, 100, 120, 200, 500, 1000)) +
        scale_x_log10(breaks = c(2, 5, 10, 20)) +
        scale_color_brewer(palette = "RdBu")
```



II Survey outcomes

In this section, we will examine the labour market status of the youth who are members of respondent households. Note that the data is still preliminary and is subject to change due to subsequent data cleaning process.

```
invisible(x ← fread("c:/data/stellenbosch/received/1/excel_all/status.prn"))
table0(x[, cstatus], ratio = T)
```

```
                  employed selfe-employed     unemployed
         0.004       0.183          0.009          0.804
```

Read roster and compute unemployment rates.

```
options(warn = -1)
invisible(x ← fread("c:/data/stellenbosch/received/1/excel_all/roster.prn"))
options(warn = 0)
invisible(x[whichgrep("01", x[, status]), status := "employed"])
invisible(x[whichgrep("02", x[, status]), status := "self-employed"])
invisible(x[whichgrep("03", x[, status]), status := "unemployed"])
invisible(x[whichgrep("04", x[, status]), status := "school"])
invisible(x[whichgrep("05", x[, status]), status := "retired/disabled"])
invisible(x[whichgrep("06", x[, status]), status := "preschool age"])
rr ← table0(x[x[, age] ≥ 20 & x[, age] ≤ 35 &
        whichgrep("emplo", x[, status]), status], ratio = T)
xb ← x[x[, age] ≥ 20 & x[, age] ≤ 35 &
        whichgrep("emplo", x[, status]) &
        whichgrep("^b", x[, com]), status]
        xg ← x[x[, age] ≥ 20 & x[, age] ≤ 35 &
        whichgrep("emplo", x[, status]) &
        whichgrep("^g", x[, com]), status]
rrb ← table0(xb, ratio = T)
rrg ← table0(xg, ratio = T)
```

Define youth as 20-35, and tabulate the labour market status. This gives unemployment rate of 68.7%, which is significantly higher than the QLFS statistic. By community wise, we see that 67.9% for the Afrikaans speaking community and 71.4% for Xhosa speaking community. The difference is statistically not significant by both proportions test and Fisher's exact test.

```
print(rr)
```

```
    employed self-employed    unemployed
       0.307          0.006          0.687
```

```
print(rrb)
```

```
    employed self-employed    unemployed
       0.315          0.005          0.679
```

```
print(rrg)
```

```
    employed self-employed    unemployed
       0.276         0.010         0.714
```

```
xb ← as.numeric(unlist(factor(xb)))
xb[xb < 3] ← 0; xb[xb == 3] ← 1
xg ← as.numeric(unlist(factor(xg)))
xg[xg < 3] ← 0; xg[xg == 3] ← 1
txb ← as.vector(rev(table(xb))); txg ← as.vector(rev(table(xg)))
nU ← c(txb[1], txg[1]); nAll ← c(sum(txb), sum(txg))
prop.test(nU, nAll, correct = T)
```

```
        2-sample test for equality of proportions with continuity correction

data:  nU out of nAll
X-squared = 0.318, df = 1, p-value = 0.5728
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.13888098  0.06928391
sample estimates:
   prop 1    prop 2
0.6794872 0.7142857
```

```
fisher.test(matrix(c(txb[1], sum(txb)-txb[1], txg[1], sum(txg)-txg[1]), ncol=2))
```

```
        Fisher's Exact Test for Count Data

data:
p-value = 0.554
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 0.5085691 1.3896283
sample estimates:
odds ratio
 0.8482946
```

```
#table0(x[x[, age] ≥ 20 & x[, age] ≤ 35, status], ratio = T)
#       drop preschool, retired/disabled, schoolers,
y ← x[x[, age] ≥ 20 & x[, age] ≤ 35, ]
invisible(y[, nummem := .N, by = c("hh", "com")])
ue ← rbind(table0(y[whichgrep("emplo", y[, status]), status]),
        table0(y[whichgrep("^ma", y[, sex]) & whichgrep("emplo", y[, status]), status]),
        table0(y[whichgrep("^fe", y[, sex]) & whichgrep("emplo", y[, status]), status]))
uer ← ue/apply(ue, 1, sum)
rownames(uer) ← c("all", "male", "female")
ii ← whichgrep("emplo", y[, status])
y ← y[ii, ]
invisible(y[, status := factor(y[, status])])
ii1 ← y[, age] ≤ 25
ii2 ← y[, age] ≥ 26 & y[, age] ≤ 30
ii3 ← y[, age] ≥ 31 & y[, age] ≤ 35
Ue0 ← Ue0r ← NULL
for (i in 1:3) {
        ii ← get(paste0("ii", i))
        ue0 ← rbind(table(y[ii , status]),
```

```
                table(y[ii & whichgrep("^ma", y[, sex]), status]),
                table(y[ii & whichgrep("^fe", y[, sex]), status]))
        ue0r ← ue0/apply(ue0, 1, sum)
        Ue0 ← rbind(Ue0, cbind(agegroup = i, ue0))
        Ue0r ← rbind(Ue0r, cbind(agegroup = i, ue0r))
}
rownames(Ue0) ← rep(c("all", "male", "female"), 3)
Ue0 ← data.table(Ue0)
invisible(Ue0[, agegroup := factor(agegroup, labels = c("20-25", "26-30", "31-35"))])
invisible(Ue0[, agegroup := factor(agegroup, labels = c("20-25", "26-30", "31-35"))])
invisible(Ue0[, gender := factor(rep(1:3, 3), labels = c("all", "male", "female"))])
rownames(Ue0r) ← rep(c("all", "male", "female"), 3)
Ue0r ← data.table(Ue0r)
invisible(Ue0r[, agegroup := factor(agegroup, labels = c("20-25", "26-30", "31-35"))])
invisible(Ue0r[, gender := factor(rep(1:3, 3), labels = c("all", "male", "female"))])
ue ← data.table(ue)
invisible(ue[, gender := factor(1:3, labels = c("all", "male", "female"))])
invisible(ue[, agegroup := "all"])
uer ← data.table(uer)
invisible(uer[, gender := factor(1:3, labels = c("all", "male", "female"))])
invisible(uer[, agegroup := "all"])
setkeyv(ue, c("agegroup", "gender"))
setkeyv(Ue0, c("agegroup", "gender"))
setkeyv(uer, c("agegroup", "gender"))
setkeyv(Ue0r, c("agegroup", "gender"))
Ue0 ← rbind(ue, Ue0)
Ue0r ← rbind(uer, Ue0r)
```

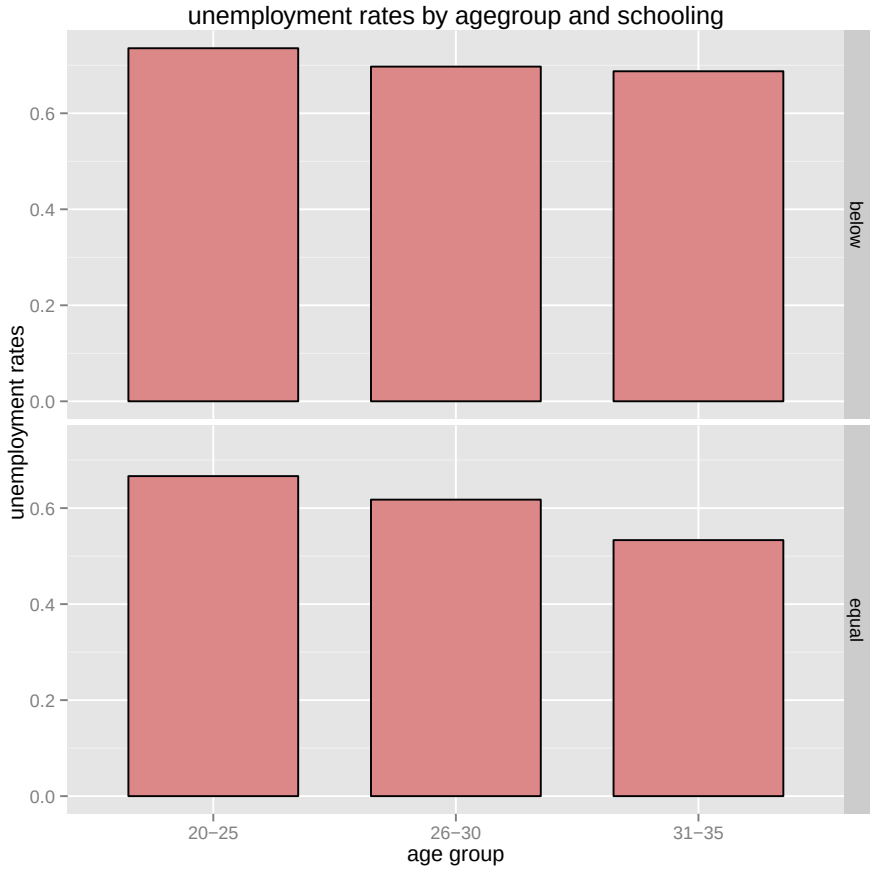
We then plot the unemployment rate by age group and gender. We see that unemployment rates are increasing in age group for men while decreasing for women. This is probably due to sample selection where working males at their prime age (31-35) tend to reject the interviews, so we end up having higher proportion of the unemployed in this group. In terms of sample selection process, the same should be true for women. However, we see a decrease in unemployment rates, indicating that women at 31-35 tend to be more employed than women of younger age groups. The all gender decrease in unemployment rates by age group is similar to one may find in QLFS data, hoowever, the underlying cause seems to be different as in our sample the deciine is driven by females.

```
library(ggplot2)
ggplot(data = Ue0r, aes(x = gender, y = unemployed, group = agegroup)) +
        geom_bar(colour="black", fill="#DD8888", width=.7, stat="identity") +
        guides(fill = FALSE) +
        xlab("sex") + ylab("unemployment rates") +
        ggtitle("unemployment rates by agegroup and gender") +
        facet_grid(agegroup ~ .)
```

Next, we plot unemployment against educational qualification.

```
invisible(y[whichgrep("diploma|certificate", y[, edu]), school := "above"])
invisible(y[whichgrep("13|14|", y[, edu]), school := "equal"])
invisible(y[!whichgrep("diploma|certificate|13|14", y[, edu]), school := "below"])
ii1 ← y[, age] ≤ 25
ii2 ← y[, age] ≥ 26 & y[, age] ≤ 30
ii3 ← y[, age] ≥ 31 & y[, age] ≤ 35
Ues0 ← Ues0r ← NULL
for (i in 1:3) {
        ii ← get(paste0("ii", i))
        ues0 ← y[ii, .SD[, table(status)], by = school]
        ues0r ← y[ii, .SD[, table(status)/length(status)], by = school]
        Ues0 ← rbind(Ues0, cbind(agegroup = i, ues0))
        Ues0r ← rbind(Ues0r, cbind(agegroup = i, ues0r))
}
Ues0 ← data.table(Ues0); Ues0r ← data.table(Ues0r)
invisible(Ues0[, status := factor(1:3, labels = c("employed", "self−employed", "unemployed
invisible(Ues0r[, status := factor(1:3, labels = c("employed", "self−employed", "unemploye
invisible(Ues0[, agegroup := factor(agegroup, labels = c("20−25", "26−30", "31−35"))])
```

```
invisible(Ues0r[, agegroup := factor(agegroup, labels = c("20-25", "26-30", "31-35"))])
Ues0r ← Ues0r[whichgrep("^un", Ues0r[, status]), ]
setnames(Ues0r, "V1", "urate")
setkeyv(Ues0r, c("agegroup", "school"))
ggplot(data = Ues0r, aes(x = agegroup, y = urate, group = school)) +
        geom_bar(colour="black", fill="#DD8888", width=.7, stat="identity") +
        guides(fill = FALSE) +
        xlab("age group") + ylab("unemployment rates") +
        ggtitle("unemployment rates by agegroup and schooling") +
        facet_grid(school ~ .)
```



We see that the level of unemployment rate is higher for the below matriculation in 20-25 age group. We also see that uneployment rates to be declining by age group, but more slowly for the below matriculation group. This contributes to a sustaining gap in unemployment rates between below and above matriculation. This is in a general agreement with QLFS data. The mechanism behind it is yet to be known to researchers. It is deemed that the job type may be different, so are the job search channels. With this data set, one should explore if this is due to lower match arrival rates or difference in search channels and intensity.

第 3 章

Heterogenous match efficiency

[§]

March 3, 2015

Rulof Burger[†] Seiro Ito[‡]

Abstract In this paper, we showed a model with one-sided endogenous match efficiency. It is assumed that schooling can enhance match efficiency, and people will choose the schooling level optimally to balance its costs and benefits of enhanced match efficiency. Assuming a financial market imperfection which limits individuals to borrow, we showed that, in equilibrium, when educational achievements can be characterised by dicohotomy (secondary vs. tertiary), tertiary education gives higher wages even it only has pure match efficiency (signalling) value with no human capital value. We also showed that relative match efficiency *vis-à-vis* its mean matters in wage levels.

Key Words equilibrium search model, match efficiency

I Introduction

In South Africa, it is casually observed that many individuals do not know where to search for the jobs. In the qualitative interviews undertaken by one of the authors, some respondents in low income areas reveal that they do not make use of the job creation centres nor employment agencies, they do not plan ahead to enquire about the job opening over the phone, but they simply go to the workplace and enquire directly. Majority of low income individuals cannot afford the internet usage, so they do not search over the web.[*1] The most cost-effective, active search method can be newspaper

[§] This paper was written when Seiro Ito visited the Faculty of Managerial and Economic Sciences, Stellenbosch Univeristy. He would like to thank deeply for their hospitality and the opportunities provided.

[†] Stellenbosch University, Stellenbosch, South Africa. rulof@sun.ac.za

[‡] Corresponding author. IDE, Chiba, Japan. seiroi@gmail.com

[*1] Not using phones and internet may sound irrational, but their non-use makes a perfect sense, given the price plans and complexity of services offered. In February 2015, with a leading carrier, data costs about monthly R.29 for 100 Mb (but pay a prohibitive, a seven times higher rate of R.2 per Mb after using 100 Mb allocation), so it is not just

advertisement, which may be subject to a limited employer base. Many individuals rely on word of mouth to get the job opening information. The quality of job information through word of mouth then may depend on the size and quality of network characterised by weak ties (Granovetter, 1983, 2005), which may be positively correlated with job searcher's own wealth levels.

This points to the questions of search efficiency impacts on labour market outcomes. A job search can be strategised to increase the rate of job match. A capacity to strategise may depend on schooling. First, strategisation requires careful thinking and planning, and schools are meant to capacitate the students in doing so. Second, alum networks of top schools can be of high quality due to its size and informational contents. The better your school friends do, the better your chances of getting the information will be.

We consider a model with heterogenous search efficiency in an equilibrium search framework of Pissarides (1985). The model treates "educational investments" (signal) as search efficiency. It derives steady state unemployment and vacancy under heterogeneity. The educational investments are assumed to carry no human capital value, and are optimally chosen by balancing the current costs and future benefits. Heterogeneity is introduced by heterogenous marginal costs of educational investments. In the search equilibrium, we naturaly see the job matching rate is greater with a greater value of educational investments.

Greater values of educational investments e can be considered to lead to a labour market advantage beyond traditional signaling function: More accurate revealation of individual traits. This is assumed to be achieved through better presentation skills and acquiring access to a better quality network which transmits information more efficiently and precisely. If $e_1 > e_2$, job matching rate is higher for individual 1 than individual 2.

Inspired by Acemoglu (2001); Navarro (2007), the model treats heterogenous individuals but do not assume sector specific employability. In fact, there is only one sector in the economy.

II Setup

II.1 Standard matching

Under the standard matching, it is assumed that an individual spends a unit time to search the jobs when unemployed, but not during employed. So the total number of job searchers in an economy is the number of unemployed uL where L is population size. There are vL vacancies in the economy. The employers and job searchers meet and examine the match of traits between individuals have and jobs require. The match of traits is "produced" in a production function-like process called a matching function. Following the previous works, the matching function is assumed to take arguments of u, v, and is homogenous of degree 1. The number of job matches x with the people under

expensive but also tremendously difficult for low income earners to plan the megabytes and use, even if you have a smart phone. A phone call costs R. 1.20 per minute, so it is about 4.8 times of cashier minimum wage (R.14.98 per hour) per minute. Phone calls, too, are expensive for low income earners.

unemployment uL and vacancies vL is given by $x(uL, vL)$. We normalize the population size L to 1. Then x is considered as the rate of job match per individual given unemployment rate u and vacancy rate v:

$$x = \tilde{x}(u, v) = \tilde{x}\left(\frac{u}{v}, 1\right) v = \tilde{x}\left(\theta^{-1}, 1\right) v \stackrel{\text{def}}{=} \tilde{q}(\theta) v, \tag{3.1}$$

where

$$\theta \stackrel{\text{def}}{=} \frac{v}{u}, \quad \tilde{q}' < 0.$$

Match arrival rates for vacancy position and the unemployed are expressed as:

$$\frac{\tilde{x}}{v} = \tilde{q}(\theta), \quad \frac{\tilde{x}}{u} = \frac{v}{u}\frac{\tilde{x}}{v} = \theta\tilde{q}(\theta). \tag{3.2}$$

II.2 Match efficiency

The above matching function has a microeconomic basis known as urn-ball matching (Petrongolo and Pissarides, 2001). Assuming that a vacancy is public knowledge and each unemployed sends one application, the probability that a vacancy receives at least one application is $1-(1-\frac{1}{vL})^{uL}$. Then the number of match is given by multiplying with total number of vacancies, or $vL\{1-(1-\frac{1}{vL})^{uL}\}$. Taking $L \to \infty$ while holding u, v fixed, $(1-\frac{1}{vL})^{uL}$ approaches to $\exp\left(\frac{u}{v}\right)$. Hence urn-ball matching function has a form

$$X(uL, vL) = vL\left\{1 - \exp\left(\frac{u}{v}\right)\right\}.$$

This function is homogenous of degree one. One way to define the efficiency in matching, from the job searcher's point of view, is to make vacancies vL variable. We can assume that the matching can incoporate efficiency by introducing $e \in [1, \infty)$ to be multiplied with the number of vancancies, giving evL. Then we have:

$$eX(uL, vL) = evL\left\{1 - \exp\left(\frac{u}{v}\right)\right\},$$

or its proportion form:

$$\frac{eX(uL, vL)}{L} \stackrel{\text{def}}{=} ex(u, v) = ev\left\{1 - \exp\left(\frac{u}{v}\right)\right\}. \tag{3.3}$$

We see that $x(u, v)$ is homogeneous of degree one, so is $ex(u, v)$.

II.3 Individuals

An individual is forward looking, infinitely lived, and maximizes the lifetime utility by choosing the labour market status and by choosing the education levels in childhood. An individual is assumed to be risk neutral, and invests in schooling e in childhood (time 0) to enhance the matching efficiency. In childhood, there is no consumption but there is a nonpecuninary cost for education.

After invested in e, an individual will search for the job and receives an offer if a firm decides to do so. An individual decides whether to accept the job. An individual will accept the offer only if it increases the expected lifetime utility. After observing the match, the matched individual and firm

will enter a generalized Nash-bargaining process where the threat points are unemployment and no production, respectively. The bargaining power for an individual is assumed to be unique and fixed at $\beta \in (0, 1)$.

The individuals receive the unemployment benefits $b > 0$ during unemployment, and firms will receive nothing if not producing. In each period, there is a fixed chance $s \in [0, 1]$ of job loss which hurts both the worker and the firm as they take away employment/production opportunities. Job loss is a random event that is not correlated with any parameters of the model. An individual will quit the job if doing so increases the expected lifetime utility. The problem that an individual faces at t under the discount rate r can be stated as maximizing the following function:

$$V(t) = \int_t^\infty \exp(-r\tau) y\{edu, m(\tau)\} d\tau$$

where $y\{edu, m(\tau)\}$ is net income in time τ with labour market status $m(\tau)$ a chosen education level edu.

We assume that matching becomes more efficient if an individual attains higher educational qualification. This is because of two related but potentially separate reasons. First, with better schooling comes with better presentation and a more matched focus, employers see the job candidate's traits more accurately, which makes them easier to hire. Secondly, higher educational qualification can grant access to higher quality networks. A network is of superior quality if it shares the information at a greater scale and speed, or with higher precision without much decay in informational contents. Or one can expect that, with better educational qualification, one can expect the peer to be closer to decision making positions of job applicants. This should give search efforts an extra efficiency in getting more offers. Thus even with the same information one sees between 1 and 2 except for e, employability of 1 is greater with the larger signaling value $e_1 > e_2$.

We assume there are $I > 0$ types of individuals. Types differ in their match efficiency $e_i \neq e_{i'}$ for $i' \neq i$, $\forall\, i' \in \mathbb{I}$. With match efficiency e_i, we redefine the matching function $\tilde{x}(\cdot) = ex(\cdot)$ as in (3.3):

$$\tilde{x}(u_i, v_i) = e_i x(u_i, v_i) = e_i x\left(\frac{u_i}{v_i}, 1\right) v = e_i x\left(\theta_i^{-1}, 1\right) v_i \stackrel{\text{def}}{=} e_i q(\theta_i) v_i, \quad q' < 0. \tag{3.4}$$

Note that now all u and v are indexed with the type i, because different level of educational investments distinguishes different types of individuals.[*2] Naturally, different values of e will result in different values of θ. Note also that an (exogenous) increase in e_i is purely welfare improving, better for both individuals and firms.

II.4 Firms

In production, a worker contributes one unit of labour which gives an output of y. We assume linear production technology, and each firm employs only one labourer. A firm can create a job to

[*2] So the number of matches x should also be indexed by i as well, but we do not do so as we use x for a function $x(\cdot)$, and it may conflate with the notion that the functional form is also different.

enjoy the profit opportunities, and can keep the worker as long as it wishes and fire at will. But the firms will keep on employing the same worker as much as they can, because we assume homogeneity in worker productivity and there is a fixed cost $\gamma > 0$ of creating a job which they must incurr had they decided to switch to a new worker. This fixed cost acts like an entry barrier and leads to a subsequent rent to be enjoyed.

The overall match for all firms becomes:

$$\overline{eq} = \sum_{i \in \mathcal{I}} \phi_i e_i q(\theta_i), \quad \sum_{i \in \mathcal{I}} \phi_i = 1,$$

where ϕ_i is proportion of type i workers.

II.5 Contrasts with search intensity model

Note that there is a close parallel with Pissarides (2000, Chapter 5)'s model with endogenous search intensity. In his model, an individual i can choose the "search units" s_i, which gives the search volume of $s_i u$. The matching function then becomes:

$$\ddot{x}(su, v) = \ddot{x}\left(s\tfrac{u}{v}, 1\right) v \overset{\text{def}}{=} \ddot{q}\left(\tfrac{\theta}{s}\right) v.$$

$$\theta \overset{\text{def}}{=} \tfrac{v}{u}, \ \ddot{q}' < 0.$$

Under variable search intensity, the match arrival rate for the unemployed shows negative externality of s, while it has positive impacts for match per vacancy.

$$\tfrac{\ddot{x}}{v} = \ddot{q}\left(\tfrac{\theta}{s}\right), \quad \tfrac{\ddot{x}}{su} = \ddot{q}\left(\tfrac{\theta}{s}\right)\tfrac{\theta}{s}.$$

This captures that searching with more search units has negative externality. An increase in s is good for firms but may not be good for individuals.

Contrasting two models may indicate:

- What changes: volume vs. efficiency.
- Notion: wander more vs. communicate better.
- Welfare: umbiguous vs. no worse.
- Choice variables: flow vs. stock.
- Timing: Contemporaneous vs. childhood.
- u: umbiguous vs. reduces.
- w: reduces (?) vs. increases.

III Equilibrium

III.1 Equilibrium Bellman equations

Individuals and firms have two potential states, respectively. Namely, employed or unemployed, and having a vacancy or a nonvacancy. These states have on going values represented by the following four Bellman equations. Following the literature, we assume firms incurr a fixed hiring cost $\gamma > 0$, there is a $s \in [0,1]$ chance of a job being destroyed (job destruction rate), the unemployed receive unemployment benefits $z > 0$, firms produce y while paying a wage w_i to the worker which results in a profit $y - w_i$, and individuals and firms discount the future with the factor $r > 0$.

Vacancy value J^V:

$$rJ^V = -\gamma + \overline{eq}\left(rJ^F - rJ^V\right). \tag{3.5}$$

Filled position value J^F:[*3]

$$rJ^F = y - w_i + (s + \delta_i)\left(rJ^V - rJ^F\right), \tag{3.6}$$

Note that e does not enter, because we assume that education has no productivity impact.[*4]Unemployment value J^U:

$$rJ_i^U = z + \theta_i e_i q(\theta_i)\left(rJ_i^E - rJ_i^U\right). \tag{3.7}$$

Employment value J^E:

$$rJ_i^E = w_i + (s + \delta_i)\left(rJ_i^U - rJ_i^E\right). \tag{3.8}$$

A firm may not need to differentiate wages across types, because they have the same productivity. However, it is assumed that a firm bargains wages to all workers. This can differentiate the wages due to different relative bargaining positions. The population increases by δ_i for each type. We assume δ_i differs across types. At each moment there will be δ_i more workers, hence matches, for type i. It reduces the value of filled positions by δ_i. Firms can offer lower wages by citing the larger number of applicants of the same type.[*5]

III.2 Individual choices in equilibrium

From (3.7) and (3.8):

$$rJ_i^E = \frac{(s + \delta_i)z + \{r + \theta_i e_i q(\theta_i)\}\, w_i}{r + s + \delta_i + \theta_i e_i q(\theta_i)}, \tag{3.9}$$

$$rJ_i^U = \frac{(r + s + \delta_i)z + \theta_i e_i q(\theta_i) w_i}{r + s + \delta_i + \theta_i e_i q(\theta_i)}. \tag{3.10}$$

[*3] δ_i is the rate of new labor market entries which reduces the asset value by δ_i because of more filled positions.

[*4]

[*5] A note on filled position value (3.6). I could have set $\delta_i = 0$ to keep things simpler.

Difference is proportional to relative benefits of employment:

$$J_i^E - J_i^U = \frac{w_i - z}{r + s + \delta_i + \theta_i e_i q(\theta_i)} \propto w_i - z. \tag{3.11}$$

Note (3.9) and (3.10) can be written as:

$$rJ_i^E = a_{i1} z + (1 - a_{i1}) w_i, \tag{3.12}$$

$$rJ_i^U = a_{i2} z + (1 - a_{i2}) w_i. \tag{3.13}$$

with

$$a_{i1} = \frac{s + \delta_i}{r + s + \delta_i + \theta_i e_i q(\theta_i)} < \frac{r + s + \delta_i}{r + s + \delta_i + \theta_i e_i q(\theta_i)} = a_{i2}.$$

$w_i > z$ shows that $rJ_i^E > rJ_i^U$ as it gives a larger weight on w_i.

III.3 Educational investments

A rational student will invest up to e^* that maximizes net expected values when initial employment probability is p:

$$\begin{aligned} e_i^* &= \text{argmax}\{pJ_i^E + (1-p)J_i^U - c(e_i)\}, \\ &= \text{argmax}\left\{(r + s + \delta_i)z + \theta_i e_i q(\theta_i) w_i + p(w_i - z) \right. \\ &\qquad \left. - \{r + s + \delta_i + \theta_i e_i q(\theta_i)\} c(e_i)\right\}, \\ &= \text{argmax}\left\{\theta_i e_i q(\theta_i) w_i - \{r + s + \delta_i + \theta_i e_i q(\theta_i)\} c(e_i)\right\}. \end{aligned} \tag{3.14}$$

We assume that $c(e_i)$ is a convex cost function. FOC is:

$$\theta_i q(\theta_i) \{w_i - c(e_i)\} - \{r + s + \delta_i + \theta_i e_i q(\theta_i)\} c'(e_i) = 0 \tag{3.15}$$

If $p = p(e_i)$ with $p' > 0$, e_i^* increases (with $c(\cdot)$ convex):

$$\theta_i q(\theta_i) \{w_i - c(e_i)\} + p'(e_i)(w_i - z) - \{r + s + \delta_i + \theta_i e_i q(\theta_i)\} c'(e_i) = 0 \tag{3.16}$$

In either case, e_i^* is increasing in w_i, which is considered as an expected wage rate. It implies that the higher the reservation wage, the longer the schooling they should acquire.

If $c(e) = c(e, \omega)$ with $\frac{\partial^2 c}{\partial e \partial \omega} < 0$, where ω is wealth, we get:

$$e^* = g(\omega), \quad g' > 0.$$

This assumption can be justified by the presence of a credit constraint, school (signal) quality $\propto \omega$, geographical sorting: distance to jobs $\propto \frac{1}{\omega}$[*6], network costs when e is a referral.

Usually, schooling is a discrete variable. Here, we assume $e = e_1, e_2$ with e_1 is a matriculation degree and e_2 is an advanced degree. Then

$$\exists\, \omega^* \in \mathbb{R}_{++} \quad \text{s.t.} \quad \omega \begin{cases} \leqslant \\ > \end{cases} \omega^* \Leftrightarrow \quad e^* = \begin{cases} e_1 \\ e_2 \end{cases}$$

[*6] This is not true in the US.

III.4 Firm choices in equilibrium

Free entry of firms imply:

$$rJ^V = 0. \tag{3.17}$$

We assume the generalized Nash barganing over matched rents. Given the bargaining power $\beta \in (0,1)$ of the individuals, this results in:

$$J_i^E - J_i^U = \frac{\beta}{1-\beta}\left(J^F - J^V\right). \tag{3.18}$$

Filled position value (3.6) can be writtenn as:

$$J^F = \frac{y - w_i}{r + s + \delta_i}.$$

Vacancy value (3.5) and free entry (3.17) give:

$$J^F = \frac{\gamma}{\overline{eq}}, \tag{3.19}$$

So

$$y - w_i - \gamma\frac{r + s + \delta_i}{\overline{eq}} = 0. \tag{3.20}$$

Job creation under free entry must yield a positive rent $y - w_i > 0$ to recover the cost γ. w_i is lower if there are more new entrants δ_i.

(3.9), (3.10), (3.23) give:

$$rJ_i^E = \frac{(s+\delta_i)z + \{r + \theta_i e_i q(\theta_i)\}\left\{\beta\left(y + \gamma\theta_i\frac{e_i q_i}{\overline{eq}}\right) + (1-\beta)z\right\}}{r + s + \delta_i + \theta_i e_i q(\theta_i)}, \tag{3.21}$$

$$rJ_i^U = \frac{(r+s+\delta_i)z + \theta_i e_i q(\theta_i)\left\{\beta\left(y + \gamma\theta_i\frac{e_i q_i}{\overline{eq}}\right) + (1-\beta)z\right\}}{r + s + \delta_i + \theta_i e_i q(\theta_i)}. \tag{3.22}$$

Note $\overline{eq} = \sum_j \phi_j \theta_j q(\theta_j)$. At $\theta_j = 0$ for $\forall j \neq i$, rJ_i^E is positve:

$$rJ_i^E\big|_{\theta_j=0} = \frac{(s+\delta_i)z + \{r + \theta_i e_i q(\theta_i)\}\{\beta(y + \gamma\theta_i) + (1-\beta)z\}}{r + s + \delta_i + \theta_i e_i q(\theta_i)}.$$

It can also be seen that:

$$\frac{\partial r J_i^E}{\partial \theta_i} > 0.$$

We see if $\theta_2 > \theta_1$

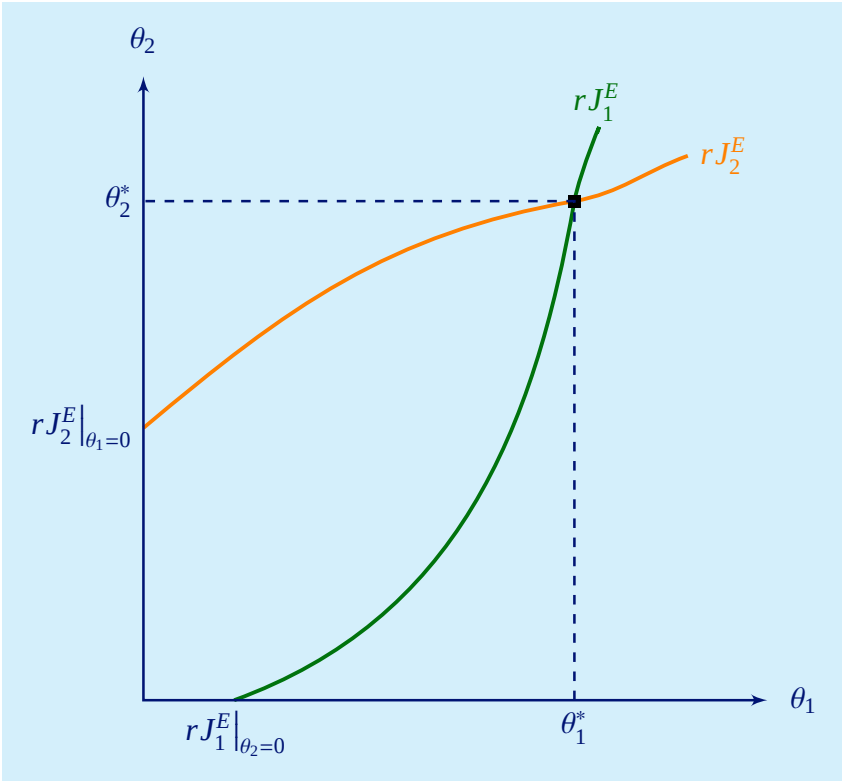
$$rJ_1^E\big|_{\theta_2=0} \gtrless rJ_2^E\big|_{\theta_1=0}, \quad \frac{\partial\, rJ_2^E\big|_{\theta_1=0}}{\partial\theta_2} < \frac{\partial\, rJ_1^E\big|_{\theta_2=0}}{\partial\theta_1}.$$

Use a short hand $dq = q(\theta_i) + \theta_i q'(\theta_i)$:

$$\begin{aligned}\frac{\partial r J_i^E}{\partial \theta_i} &= -\frac{\{num\}}{(denom)^2} e_i dq + \frac{e_i}{(denom)}\Bigg[\{wage\}\, dq \\ &\quad + \{r + \theta_i e_i q(\theta_i)\}\beta\gamma\left\{dq - \frac{\theta_i q(\theta_i)\phi_i e_i q'(\theta_i)}{\overline{eq}^2}\right\}\Bigg] \\ &= \frac{e_i dq}{(denom)^2}\Bigg[-\{num\} + (denom)\{wage\} \\ &\quad + (denom)\{r + \theta_i e_i q(\theta_i)\}\beta\gamma\left\{1 - \phi_i\theta_i \frac{e_i q(\theta_i)}{\overline{eq}^2}\frac{q'(\theta_i)}{dq}\right\}\Bigg].\end{aligned}$$

The last term is positive. Comparing the 1st and 2nd terms and one can show:

$$-\{num\} + (denom)\{wage\} = (s + \delta_i)\beta\left(y + \gamma\theta_i \frac{e_i q_i}{\overline{eq}} - z\right) > 0.$$



III.5 Steady state

(3.5), (3.6), (3.7), (3.8) and (3.17), (3.18) give:

$$w_i = \beta\left(y + \gamma\theta_i \frac{e_i q_i}{\overline{eq}}\right) + (1 - \beta)z. \tag{3.23}$$

So the higher the relative match efficiency $\frac{e_i q_i}{\overline{eq}}$, the higher the rent share. Looking at FOC in (3.15), $\frac{de_i}{dw_i} > 0$ and an increase in w_i encourages investments in e_i.

$$\theta_i q(\theta_i)\{w_i - c(e_i)\} - \{r + s + \delta_i + \theta_i e_i q(\theta_i)\}\, c'(e_i) = 0 \tag{3.15}$$

Note the externality: If $j \neq i$ invests more in e_j, i's rent share falls.

(3.7), (3.8) and (3.17), (3.18) give:

$$\begin{aligned}(r+s+\delta_i)(J_i^E - J_i^U) &= w_i - rJ_i^U,\\ J_i^E - J_i^U &= \tfrac{\beta}{1-\beta}J^F,\\ J^F &= \tfrac{1}{r+s+\delta_i}(y-w_i)\end{aligned}$$

So

$$w_i = \beta y + (1-\beta)rJ_i^U. \tag{3.24}$$

(3.7), (3.18), (3.19) give:

$$\begin{aligned}yJ_i^U &= z + \theta_i e_i q(\theta_i)\left(rJ_i^E - rJ_i^U\right),\\ &= z + \theta_i e_i q(\theta_i)\tfrac{\beta}{1-\beta}J^F,\\ &= z + \tfrac{\beta}{1-\beta}\gamma\theta_i\tfrac{e_i q_i}{\overline{eq}}.\end{aligned} \tag{3.25}$$

(3.24) and (3.25) give (3.23).

The steady state is characterised by the following equations. For signals:

$$\theta_i q(\theta_i)\{w_i - c(e_i)\} - \{r+s+\delta_i+\theta_i e_i q(\theta_i)\}c'(e_i) = 0 \tag{3.15}$$

Job creation:

$$y - w_i - \gamma\frac{r+s+\delta_i}{\overline{eq}} = 0. \tag{3.20}$$

Wage:

$$w_i = \beta\left(y + \gamma\theta_i\frac{e_i q_i}{\overline{eq}}\right) + (1-\beta)z. \tag{3.23}$$

Beveridge curve:

$$u_i = \frac{s+\delta_i}{s+\delta_i+\theta_i e_i q(\theta_i)} \tag{3.26}$$

Again, $\overline{eq}$ is a function of all θ_i's, so $4\times I$ equations must be solved simultaneously.

Alternatively, the steady state is $\{u_i, v_i, e_i\}$ determined by:

$$\begin{aligned}&\theta_i q(\theta_i)\left\{\beta\left(y+\gamma\theta_i\frac{e_i q_i}{\overline{eq}}\right) + (1-\beta)z - c(e_i)\right\}\\ &\quad - \{r+s+\delta_i+\theta_i e_i q(\theta_i)\}c'(e_i) = 0\end{aligned} \tag{3.16}$$

$$(1-\beta)(y-z) - \frac{\gamma}{\overline{eq}}\{r+s+\delta_i-\beta\theta_i e_i q(\theta_i)\} = 0. \tag{3.27}$$

$$u_i = \frac{s+\delta_i}{s+\delta_i+\theta_i e_i q(\theta_i)} \tag{3.28}$$

Three unknowns u_i, v_i (or $\theta_i = \frac{v_i}{u_i}$), e_i are solved with three equations provided that other types are in an equlibrium.

For $i = 1, 2$, (3.23) and (3.20) give:

$$(1-\beta)(y-z) - \frac{\gamma}{\overline{eq}}\{r+s+\delta_i-\beta\theta_i e_i q(\theta_i)\} = 0. \tag{3.29}$$

This gives θ_i. An equilibrium requires θ_1 and θ_2 to be determined simultaneously. (3.28), (3.29) give u_i, θ_i (or v_i). For $i = 1, 2$, it gives unique $\theta_1^* < \theta_2^*$.

IV Concluding remarks

In this paper, we showed a model with one-sided endogenous match efficiency. It is assumed that schooling can enhance match efficiency, and people will choose the schooling level optimally to balance its costs and benefits of enhanced match efficiency. Assuming a financial market imperfection which limits individuals to borrow, we showed that, in equilibrium, when educational achievements can be characterised by dicohotomy (secondary vs. tertiary), tertiary education gives higher wages even it only has pure match efficiency (signalling) value with no human capital value. We also showed that relative match efficiency *vis-à-vis* its mean matters in wage levels.

参考文献

Acemoglu, Daron. 2001. “Good jobs versus bad jobs.” Journal of labor Economics, 19(1): 1–21.

Granovetter, Mark. 1983. “The strength of weak ties: A network theory revisited.” Sociological Theory, 1(1): 201–233.

Granovetter, Mark. 2005. “The impact of social structure on economic outcomes.” Journal of economic perspectives, 33–50.

Lumley, Thomas. 2014. “survey: analysis of complex survey samples.” R package version 3.30.

Navarro, Lucas. 2007. “Labor market policies in a sector specific search model with heterogeneous firms and workers.” Revista de Análisis Economico–Economic Analysis Review, 22(2): 29–45.

Petrongolo, Barbara, and Christopher A. Pissarides. 2001. “Looking into the Black Box: A Survey of the Matching Function.” Journal of Economic Literature, 39: 390–431.

Pissarides, Christopher A. 1985. “Short-run equilibrium dynamics of unemployment, vacancies, and real wages.” The American Economic Review, 676–690.

Pissarides, Christopher A. 2000. Equilibrium unemployment theory. MIT Press, Cambridge.

Statistics South Africa. 2008. Guide to Quarterly Labour Force Survey. Statistics South Africa.

調査研究報告書
地域 2014-C12
南部アフリカにおける労働参加と失業

2015 年 3 月 31 日発行
発行所　独立行政法人日本貿易振興機構
アジア経済研究所
251-8545 千葉県千葉市美浜区若葉 3-2-2
電話 043-299-9500