Data Sources and Calculations of Initial Values

Talitha F. Speranza

2017-12-27

Number of Agents

The number of agents of each type in relation to other types will be based on the real proportions we observe in Brazilian time series and official records. We begin by reading data on Brazilian firms and employees[1], renaming columns and converting some data types of the resulting table:

[1] Brazilian Institute for Statistics and Geography, Central Registry of Enterprises (2015)

```
dist.firms <- read.csv2("distribuicao_empresas.csv",
    stringsAsFactors = F)

dist.firms[dist.firms == "X"] <- 0
names(dist.firms) <- c("type", "range", "n.firms",
    "n.workers", "wage")
dist.firms$n.workers = as.integer(dist.firms$n.workers)
dist.firms$n.firms = as.integer(dist.firms$n.firms)
dist.firms$wage = as.numeric(gsub(",", "\\.",
    dist.firms$wage))

knitr::kable(head(dist.firms, 17))
```

type	range	n.firms	n.workers	wage
A Agricultura, pecuária, produção florestal, pesca e aquicultura	0 a 4	88848	110599	1.6
A Agricultura, pecuária, produção florestal, pesca e aquicultura	5 a 9	6009	39090	1.6
A Agricultura, pecuária, produção florestal, pesca e aquicultura	10 a 19	3592	47512	1.8
A Agricultura, pecuária, produção florestal, pesca e aquicultura	20 a 29	1074	25712	1.9
A Agricultura, pecuária, produção florestal, pesca e aquicultura	30 a 49	848	0	0.0
A Agricultura, pecuária, produção florestal, pesca e aquicultura	50 a 99	601	0	0.0
A Agricultura, pecuária, produção florestal, pesca e aquicultura	100 a 249	379	56782	2.1
A Agricultura, pecuária, produção florestal, pesca e aquicultura	250 a 499	159	0	0.0
A Agricultura, pecuária, produção florestal, pesca e aquicultura	500 ou mais	136	0	0.0
B Indústrias extrativas	0 a 4	6254	0	0.0
B Indústrias extrativas	5 a 9	1680	11083	2.0
B Indústrias extrativas	10 a 19	1168	0	0.0
B Indústrias extrativas	20 a 29	482	11485	2.5
B Indústrias extrativas	30 a 49	363	13881	3.0
B Indústrias extrativas	50 a 99	288	19742	3.4
B Indústrias extrativas	100 a 249	145	21489	6.6
B Indústrias extrativas	250 a 499	42	14157	7.4

After that, we obtain the unemployement rate[2] in 2015, so that we can calculate the proportional number of unemployed workers that will be part of the model:

[2] IBGE, Monthly National Household Survey(PNAD-C)

```
unemp <- read.csv2("desemprego.csv", stringsAsFactors = F)
unemp.rate <- mean(unemp[grepl("2015", unemp[,
    1]), 2])/100
unemp.rate
```

```
## [1] 0.083
```

We set up filters to create two subsets of `dist.firms`: one for financial activities (in our model, banks) and other for industry, construction, and agriculture (firms). Finally, we calculate the number of agents of each type:

```
types <- unique(dist.firms$type)

# Financial Activities
fin.act <- types[10]

# Industry, Construction and Agriculture
ind.constr <- c(types[1], types[2], types[3],
    types[5])

# Fix the total number of workers
n.agents <- 2000

# Non-financial firms
n.firms <- sum(dist.firms[dist.firms$type != fin.act,
    "n.firms"])
n.firms
```

```
## [1] 4532385
```

```
# Financial firms
n.banks <- sum(dist.firms[dist.firms$type == fin.act,
    "n.firms"])
n.banks
```

```
## [1] 81043
```

```
# Number of workers in non-financial firms
n.employees <- sum(dist.firms[dist.firms$type !=
    fin.act, "n.workers"])
n.employees
```

```
## [1] 28966727
```

```
# Number of unemployed workers
n.unemp <- round(n.agents * unemp.rate, 0)
n.unemp
```

```
## [1] 166
```

```
# Number of employed workers
n.emp <- (1 - unemp.rate) * n.agents
n.emp
```

```
## [1] 1834
```

```
# Total number of workers
n.consumers <- n.employees/(1 - unemp.rate)
n.consumers
```

```
## [1] 31588579
```

The `params` dataset will include all the initial values of the model, so we start filling it with parameters that were already calculated.

```
# Fill the 'params' data.frame
params <- data.frame(Description = character(),
    Unit = character(), Value1 = numeric(), Value2 = numeric(),
    stringsAsFactors = FALSE)
params[1, ] <- list("Number of Workers", "workers",
    n.agents, NA)
params[2, ] <- list("Number of Unemployed Workers",
    "workers", n.unemp, NA)
params[3, ] <- list("Number of Firms", "firms",
    round(n.agents * (n.firms/n.consumers)), NA)
params[4, ] <- list("Number of Banks", "banks",
    round(n.agents * (n.banks/n.consumers)), NA)

knitr::kable(params[1:4, -4])
```

Description	Unit	Value1
Number of Workers	workers	2000
Number of Unemployed Workers	workers	166
Number of Firms	firms	287
Number of Banks	banks	5

Distribution of Firms' Sizes

In this version of the model, the metric for assessing the size of the firms will be the number of employees. This section defines how many workers each of the model's firms has to initially employ in order to reflect real proportions.

```
# Get different sizes in terms of number of
# workers
ranges <- unique(dist.firms$range)

# Eliminate financial activities from dataset
dist.firms = dist.firms[dist.firms$type != fin.act,
    ]

# Calculate totals

new.dist.firms <- data.frame(range = character(),
    n.firms = integer(), n.workers = integer(),
    med.wage = numeric(), stringsAsFactors = F)

for (i in 1:length(ranges)) {
    range = ranges[i]
    dist.range = dist.firms[dist.firms$range ==
        range, ]
    # Proportional number of workers in firms
    # sized 'range'
    prop.workers = round((sum(dist.range$n.workers)/n.employees) *
        n.emp)
    # Proportional number of firms sized 'range'
    prop.firms = round((sum(dist.range$n.firms)/n.firms) *
        params[3, 3])
    # Medium wage of firms sized 'range'
    med.wage = mean(dist.range$wage)
    new.dist.firms[i, ] <- list(range, prop.firms,
        prop.workers, med.wage)
}

new.dist.firms[new.dist.firms$n.firms == 0, "n.firms"] = 1
```

```
knitr::kable(new.dist.firms)
```

range	n.firms	n.workers	med.wage
0 a 4	205	384	1.46875
5 a 9	45	278	1.93750
10 a 19	22	295	2.09375
20 a 29	6	78	2.02500
30 a 49	4	118	2.30000
50 a 99	3	124	2.40000
100 a 249	1	186	3.08125
250 a 499	1	56	2.86875
500 ou mais	1	315	2.76875

We now establish the number of employees that will distinguish groups of firms. As defined by the Brazilian government[3], these are four groups: microfirms, small firms, medium firms and large firms. However, different thresholds are established for industry and commerce. Since our model do not differentiate between these two activities, we have to develop our own criteria to classify firms, but they will be based on Brazilian government's. The key is to find the percentages of industry and commerce in the total number of firms and weight thresholds appropriately.

[3] SEBRAE, Anuário do trabalho na micro e pequena empresa 2013, p.17

```
# Create a filter for commerce
com <- !(dist.firms$type %in% ind.constr) & dist.firms$type !=
    fin.act
# Calculate the weight of industry
w.ind <- sum(dist.firms[dist.firms$type %in% ind.constr,
    "n.firms"])/n.firms
# Calculate the weight of commerce
w.com <- sum(dist.firms[com, "n.firms"])/n.firms

# Find thresholds
micro <- round(19 * w.ind + 9 * w.com, 0)
small <- round(99 * w.ind + 49 * w.com, 0)
medium <- round(499 * w.ind + 99 * w.com, 0)

# Fill the 'params' data.frame
params[5, ] <- list("Microfirms maximum number of workers",
    "workers", micro, NA)
params[6, ] <- list("Small firms maximum number of workers",
    "workers", small, NA)
params[7, ] <- list("Medium firms maximum number of workers",
```

```
    "workers", medium, NA)

knitr::kable(params[5:7, -4])
```

	Description	Unit	Value1
5	Microfirms maximum number of workers	workers	11
6	Small firms maximum number of workers	workers	58
7	Medium firms maximum number of workers	workers	168

The next step is to add the quantities of firms and workers under each of the groups defined above.

```
# Sum the number of workers and firms across
# microfirms
mic <- new.dist.firms[1, 2:3] + new.dist.firms[2,
    2:3]
mic[1, 3] <- mic[2]/mic[1]
mic[1, 4] <- (new.dist.firms[1, 3] * new.dist.firms[1,
    4] + new.dist.firms[2, 3] * new.dist.firms[2,
    4])/mic[2]

knitr::kable(mic)
```

n.firms	n.workers	n.workers.1	n.workers.2
250	662	2.648	1.665597

Although several configurations yield these aggregate numbers for microfirms, it suffices to pick one of such settings randomly and assign it to the model.

```
# Define an arbitrary distribution which fits
# the aggregate numbers
params[8, ] <- list("Microfirms A", "1:workers 2:firms",
    2, 101)
params[9, ] <- list("Microfirms B", "1:workers 2:firms",
    3, 148)
params[10, ] <- list("Microfirms C", "1:workers 2:firms",
    6, 1)
params[11, ] <- list("Microfirms D", "1:workers 2:firms",
    10, 1)
params[12, ] <- list("Microfirms TOTALS", "1:workers 2:firms",
    mic[2], mic[1])
```

```
knitr::kable(params[8:12, ])
```

	Description	Unit	Value1	Value2
8	Microfirms A	1:workers 2:firms	2	101
9	Microfirms B	1:workers 2:firms	3	148
10	Microfirms C	1:workers 2:firms	6	1
11	Microfirms D	1:workers 2:firms	10	1
12	Microfirms TOTALS	1:workers 2:firms	662	250

The same aggregation and configuration process will be performed for small, medium and large firms:

```
# Sum the number of workers and firms across
# small firms
sm <- new.dist.firms[3, 2:3] + new.dist.firms[4,
    2:3] + new.dist.firms[5, 2:3]
sm[1, 3] <- sm[2]/sm[1]
sm[1, 4] <- (new.dist.firms[3, 3] * new.dist.firms[3,
    4] + new.dist.firms[4, 3] * new.dist.firms[4,
    4] + new.dist.firms[5, 3] * new.dist.firms[5,
    4])/sm[2]

# Define an arbitrary distribution which fits
# the aggregate numbers
params[13, ] <- list("Small Firms A", "1:workers 2:firms",
    15, 30)
params[14, ] <- list("Small Firms B", "1:workers 2:firms",
    20, 1)
params[15, ] <- list("Small Firms C", "1:workers 2:firms",
    21, 1)
params[16, ] <- list("Small Firms TOTALS", "1:workers 2:firms",
    sm[2], sm[1])

knitr::kable(params[13:16, ])
```

	Description	Unit	Value1	Value2
13	Small Firms A	1:workers 2:firms	15	30
14	Small Firms B	1:workers 2:firms	20	1
15	Small Firms C	1:workers 2:firms	21	1
16	Small Firms TOTALS	1:workers 2:firms	491	32

```
# Sum the number of workers and firms across
# medium firms
med <- new.dist.firms[6, 2:3] + new.dist.firms[7,
    2:3]
med[1, 3] <- med[2]/med[1]
med[1, 4] <- (new.dist.firms[6, 3] * new.dist.firms[6,
    4] + new.dist.firms[7, 3] * new.dist.firms[7,
    4])/med[2]

# Define an arbitrary distribution which fits
# the aggregate numbers
params[17, ] <- list("Medium Firms A", "1:workers 2:firms",
    77, 2)
params[18, ] <- list("Medium Firms B", "1:workers 2:firms",
    78, 2)
params[19, ] <- list("Medium Firms TOTALS", "1:workers 2:firms",
    med[2], med[1])

knitr::kable(params[17:19, ])
```

	Description	Unit	Value1	Value2
17	Medium Firms A	1:workers 2:firms	77	2
18	Medium Firms B	1:workers 2:firms	78	2
19	Medium Firms TOTALS	1:workers 2:firms	310	4

```
# Sum the number of workers and firms across
# large firms
lg <- new.dist.firms[8, 2:3] + new.dist.firms[9,
    2:3]
lg[1, 3] <- lg[2]/lg[1]
lg[1, 4] <- (new.dist.firms[8, 3] * new.dist.firms[8,
    4] + new.dist.firms[9, 3] * new.dist.firms[9,
    4])/lg[2]

# Define an arbitrary distribution which fits
# the aggregate numbers
params[20, ] <- list("Large Firms A", "1:workers 2:firms",
    185, 1)
params[21, ] <- list("Large Firms B", "1:workers 2:firms",
    186, 1)
params[22, ] <- list("Large Firms TOTALS", "1:workers 2:firms",
    lg[2], lg[1])
```

```
knitr::kable(params[20:22, ])
```

	Description	Unit	Value1	Value2
20	Large Firms A	1:workers 2:firms	185	1
21	Large Firms B	1:workers 2:firms	186	1
22	Large Firms TOTALS	1:workers 2:firms	371	2

Gross Domestic Product

The initial Gross Domestic Product (GDP) of the model will be defined in terms of the population of workers, just as every other model parameter. First, we collect the values of the real[4] and the nominal[5] GDP in 2015.

[4] World Bank, World Development Indicators (2017)

[5] Central Bank of Brasil (BCB), Department of Economics

```
real.gdp = 1817242668600
nom.gdp = 6.000572e+12
```

Then we obtain the percentage of salaries in GDP. The last available data on detailed GDP composition are from 2014[6], but these are surely acceptable values for 2015.

[6] IBGE, National Accounts System (2014)

```
# Percentage of salaries in GDP
perc.sal = 2516592/5778953
perc.sal
```

```
## [1] 0.4354754
```

We are thus able to determine the model's initial GDP and its composition with regard to profits and salaries.

```
# Definion of model parameters
gdp = round(real.gdp/n.consumers) * n.agents
gross.sal = perc.sal * gdp
gross.profit = gdp - gross.sal


params[23, ] <- list("Real GDP", "goods", gdp,
    NA)
params[24, ] <- list("Real GDP per employed worker",
    "goods", round(gdp/n.emp), NA)
params[25, ] <- list("Percentage of salaries in GDP",
    "percentage", 100 * round(perc.sal, 2), NA)
params[26, ] <- list("Profit per worker (last 12 months)",
    "goods", round(gross.profit/n.emp), NA)
params[27, ] <- list("Saving rate", "percentage",
```

```
    16.4, NA)

knitr::kable(params[23:27, ])
```

	Description	Unit	Value1	Value2
23	Real GDP	goods	115056000	NA
24	Real GDP per employed worker	goods	62735	NA
25	Percentage of salaries in GDP	percentage	44	NA
26	Profit per worker (last 12 months)	goods	35415	NA
27	Saving rate	percentage	16.4	NA

Distribution of Wealth and Income

Workers belong to social classes defined by income brackets. Each of these brackets, in turn, concentrates a certain amount of wealth and is delimited by monthly earnings in terms of minimum wages. In this session we attribute a social class to every worker and set his income and wealth accordingly.

Data on the distribution of wealth and income are readily available[7]. We now read and reframe them to facilitate manipulation.

[7] Federal Revenue Office, Grandes Números da DIRPF

```
# Read data
dist.inc <- read.csv("distribuicao_renda_riqueza.csv",
    stringsAsFactors = F)
dist.inc[, -1] <- data.frame(lapply(dist.inc[,
    -1], function(x) {
    y <- gsub(",", "", x)
    as.numeric(y)
}))

names(dist.inc) = c("faixa", "pessoas", "renda",
    "rend.trib.ex", "rend.ist", "bens")
dist.inc[, -1] <- data.frame(lapply(dist.inc[,
    -1], as.integer))

knitr::kable(dist.inc)
```

faixa	pessoas	renda	rend.trib.ex	rend.ist	bens
Até 1/2	1301366	254	46	113	136273
Mais de 1/2 a 1	573674	4487	92	341	38903
Mais de 1 a 2	1227268	14525	599	2553	135712
Mais de 2 a 3	3278035	73567	2159	6323	268682

faixa	pessoas	renda	rend.trib.ex	rend.ist	bens
Mais de 3 a 5	7403868	228922	16832	29606	526420
Mais de 5 a 7	4339708	192783	16498	32910	443328
Mais de 7 a 10	3352450	202073	18801	42627	496954
Mais de 10 a 15	2536352	211127	21922	58535	604905
Mais de 15 a 20	1180520	130938	15647	45710	445973
Mais de 20 a 30	1086611	157914	21739	69414	622922
Mais de 30 a 40	489421	92454	14777	51599	426299
Mais de 40 a 60	389811	89905	18318	69382	524434
Mais de 60 a 80	142916	37610	10550	44527	303922
Mais de 80 a 160	141451	40987	18427	84343	533681
Mais de 160 a 240	32329	11540	8269	39315	245037
Mais de 240 a 320	13753	6063	5447	24337	151526
Mais de 320	29311	27541	62826	207572	1288419
Total	27518844	1522690	252949	809206	7193391

Wealth is on the last column of `dist.inc`, and totals on the last row. Therefore, total wealth is element (18, 6) of `dist.inc` multiplied by 1 million (data are in R$ millions). To find the model's total wealth proportional to its GDP, we have to divide it by the nominal GDP, since income and wealth are in nominal values, and then multiply it by the model's initial GDP.

```
# Get total wealth
wealth = (dist.inc[nrow(dist.inc), ncol(dist.inc)]) *
    1e+06

# Find wealth in terms of the model's GDP
wealth = (wealth/nom.gdp) * gdp
```

Given that the table separates income that is exempt from taxes (column 5) from income that is not exempt (column 3), we collapse the two values by adding them. Column 4 is removed because it contains income from investments and real-state, but the current version of the model does not incorporate these.

```
# Collapse 'renda isenta' and 'renda não
# isenta'
dist.inc$renda <- dist.inc$renda + dist.inc$rend.ist
dist.inc <- dist.inc[, c(-4, -5)]

# Calculate totals, column-wise
tot <- dist.inc[nrow(dist.inc), ]

knitr::kable(tot)
```

	faixa	pessoas	renda	bens
18	Total	27518844	2331896	7193391

IBGE divides the population in social classes according to the number of minimum salaries workers earn each month[8]. We aggregate the lines of table `dist.inc` following this classsification.

[8] IBGE, Faixas Salariais e Classe Social

```
# Aggregate by social class
filt <- dist.inc$faixa %in% c("Até 1/2", "Mais de 1/2 a 1",
    "Mais de 1 a 2")
class.E = dist.inc[filt, ]
class.E = data.frame(lapply(class.E[, -1], sum))
class.E$faixa = "E - Between 0+ and 2 MS"
class.E[, -4] = class.E[, -4]/tot[, -1]

filt <- dist.inc$faixa %in% c("Mais de 2 a 3",
    "Mais de 3 a 5")
class.D = dist.inc[filt, ]
class.D = data.frame(lapply(class.D[, -1], sum))
class.D$faixa = "D - Between 2+ and 5 MS"
class.D[, -4] = class.D[, -4]/tot[, -1]

filt <- dist.inc$faixa %in% c("Mais de 5 a 7",
    "Mais de 7 a 10")
class.C = dist.inc[filt, ]
class.C = data.frame(lapply(class.C[, -1], sum))
class.C$faixa = "C - Between 5+ and 10 MS"
class.C[, -4] = class.C[, -4]/tot[, -1]

filt <- dist.inc$faixa %in% c("Mais de 10 a 15",
    "Mais de 15 a 20")
class.B = dist.inc[filt, ]
class.B = data.frame(lapply(class.B[, -1], sum))
class.B$faixa = "B - Between 10+ and 20 MS"
class.B[, -4] = class.B[, -4]/tot[, -1]

filt <- dist.inc$faixa %in% unique(dist.inc$faixa)[10:17]
class.A = dist.inc[filt, ]
class.A = data.frame(lapply(class.A[, -1], sum))
class.A$faixa = "A - More than 20 MS"
class.A[, -4] = class.A[, -4]/tot[, -1]

props <- rbind(class.A, class.B, class.C, class.D,
    class.E)
```

```
classes <- c("A", "B", "C", "D", "E")

df <- data.frame(Description = paste("Number of workers in class",
    classes), Unit = "employed workers - unemployed workers",
    Value1 = round(props$pessoas * n.emp), Value2 = round(props$pessoas *
        n.unemp))
params <- rbind(params, df)

df <- data.frame(Description = paste("Monthly wage of an employed worker of class",
    classes), Unit = "goods", Value1 = round((props$renda *
    gross.sal/round(props$pessoas * n.emp))/12),
    Value2 = NA)
params <- rbind(params, df)

df <- data.frame(Description = paste("Wealth of a worker of class",
    classes), Unit = "goods", Value1 = round((props$bens *
    wealth)/(props$pessoas * n.agents)), Value2 = NA)
params <- rbind(params, df)

knitr::kable(params[28:42, ])
```

	Description	Unit	Value1	Value2
28	Number of workers in class A	employed workers - unemployed workers	155	14
29	Number of workers in class B	employed workers - unemployed workers	248	22
30	Number of workers in class C	employed workers - unemployed workers	513	46
31	Number of workers in class D	employed workers - unemployed workers	712	64
32	Number of workers in class E	employed workers - unemployed workers	207	19
33	Monthly wage of an employed worker of class A	goods	12181	NA
34	Monthly wage of an employed worker of class B	goods	3222	NA
35	Monthly wage of an employed worker of class C	goods	1642	NA
36	Monthly wage of an employed worker of class D	goods	851	NA
37	Monthly wage of an employed worker of class E	goods	193	NA
38	Wealth of a worker of class A	goods	464694	NA
39	Wealth of a worker of class B	goods	74592	NA
40	Wealth of a worker of class C	goods	32250	NA
41	Wealth of a worker of class D	goods	19638	NA
42	Wealth of a worker of class E	goods	26438	NA

Debt

Non-corporate debt will be expressed as a percentage of household income[9]. Since families in our model are composed of one member

[9] BCB, Household Debt without Mortgage Loans (Series 20400)

only, this percentage will represent how much debt each worker has in relation to his income.

Corporate debt[10] will be split among firms in proportion to the their number of employees. It means that the whole amount of debt will be divided into the total number of workers, and the result of this division will be the unit of debt. For instance, a firm with 5 workers will have a debt of 5 times this unit. This initialization process occurs in the Netlogo program.

[10] BCB, Credit operations outstanding by type of borrower - Private sector (Series 22047)

As usual, initial values will be those of 2015 and compatibilization with model units will be achieved through simple proportions:

```
hh.debt <- read.csv2("divida_familias.csv", stringsAsFactors = F)
hh.debt <- round(mean(as.numeric(hh.debt[grepl("2015",
    hh.debt$Date), 2])), 2)
params[nrow(params) + 1, ] <- list("Mean household debt (% last 12 months earnings)",
    "goods", hh.debt, NA)

f.debt <- read.csv2("divida_empresas.csv", stringsAsFactors = F)
f.debt[, 2] <- gsub(",", "", f.debt[, 2])
f.debt <- round(mean(as.numeric(f.debt[grepl("2015",
    f.debt$Date), 2])), 2)

prop.f.debt <- round(f.debt * 1e+06/nom.gdp, 2)
params[nrow(params) + 1, ] <- list("Mean firms debt (% GDP)",
    "goods", 100 * prop.f.debt, NA)
params[nrow(params) + 1, ] <- list("Firms debt per worker",
    "goods", round(prop.f.debt * gdp/n.emp), NA)

knitr::kable(params[43:45, ])
```

	Description	Unit	Value1	Value2
43	Mean household debt (% last 12 months earnings)	goods	27.17	NA
44	Mean firms debt (% GDP)	goods	23	NA
45	Firms debt per worker	goods	14429	NA

Interest Rates

Interest rates charged on loans will be different across types of agents. The risk of a worker defaulting on his debt is greater than the risk of a firm going bankrupt. Therefore, a worker must pay higher interests on loans. The same logic applies to firms of different sizes. Smaller firms offers more risk, hence they must be charged higher interests.

In view of the lack of data on the spread of these different charges,

interest rates will be calibrated along with several other parameters in future versions of our model. For now, banks will choose values in an almost random fashion, but the mean of the distribution of rates will be the real mean interest rate on new credit operations[11] in 2015.

[11] BCB-Dstat, Monthly average interest rate of nonearmarked new credit operations - Non-financial corporations (Series 25437) and Households (Series 25462)

```
interest <- read.csv2("taxa_de_juros.csv", stringsAsFactors = F)
f.ints <- round(mean(as.numeric(interest[grepl("2015",
    interest$Date), 2])), 2)
h.ints <- round(mean(as.numeric(interest[grepl("2015",
    interest$Date), 3])), 2)

params[nrow(params) + 1, ] <- list("Mean interest rate on loans - workers",
    "monthly yield", h.ints, NA)
params[nrow(params) + 1, ] <- list("Mean interest rate on loans - firms",
    "monthly yield", f.ints, NA)

knitr::kable(params[46:47, ])
```

	Description	Unit	Value1	Value2
46	Mean interest rate on loans - workers	monthly yield	3.95	NA
47	Mean interest rate on loans - firms	monthly yield	2.08	NA