11. Using R to Support Simulation

Data Science for OR - J. Duggan

Using the tidyverse to support simulation

- Tidying input data
- Analysing simulation output
- Running sensitivity analysis

NOTES AND INSIGHTS

Input and output data analysis for system dynamics modelling using the tidyverse libraries of R

Jim Duggan*

Syst. Dyn. Rev. **34**, 438–461 (2018)

(1) Tidying input data

Table 2. Time series influenza data from the 1957 pandemic (U.K. data)

Week	Young	Child	Adult	Elderly
1	0	0	1	1
2	0	2	6	1
3	0	2	4	2
4	23	73	63	11
5	63	208	173	41
6	73	207	171	27
7	66	150	143	7
8	26	40	87	29
9	17	18	33	12
10	3	4	13	6
11	2	6	16	5
12	1	6	11	3
13	0	1	6	5
14	0	2	2	2
15	0	1	3	0
16	0	1	4	6
17	0	1	3	0
18	2	1	7	1
19	1	1	6	2

DOI: 10.1002/sdr

Using readr to access data

```
inc <- read_csv("../../11 simulation/code/sdr_paper1/data/Inci
```

```
## Parsed with column specification:
## cols(
##   Week = col_double(),
##   Young = col_double(),
##   Child = col_double(),
##   Adult = col_double(),
##   Elderly = col_double()
## )
```

```
slice(inc,1:2)
```

```
## # A tibble: 2 x 5
##    Week Young Child Adult Elderly
##   <dbl> <dbl> <dbl> <dbl>   <dbl>
## 1     1     0     0     1       1
```

Convert to Tidy Data

```
t_inc <- gather(inc,Cohort,Incidence,Young:Elderly)
slice(t_inc,1:8)
```

```
## # A tibble: 8 x 3
##    Week Cohort Incidence
##   <dbl> <chr>      <dbl>
## 1     1 Young          0
## 2     2 Young          0
## 3     3 Young          0
## 4     4 Young         23
## 5     5 Young         63
## 6     6 Young         73
## 7     7 Young         66
## 8     8 Young         26
```

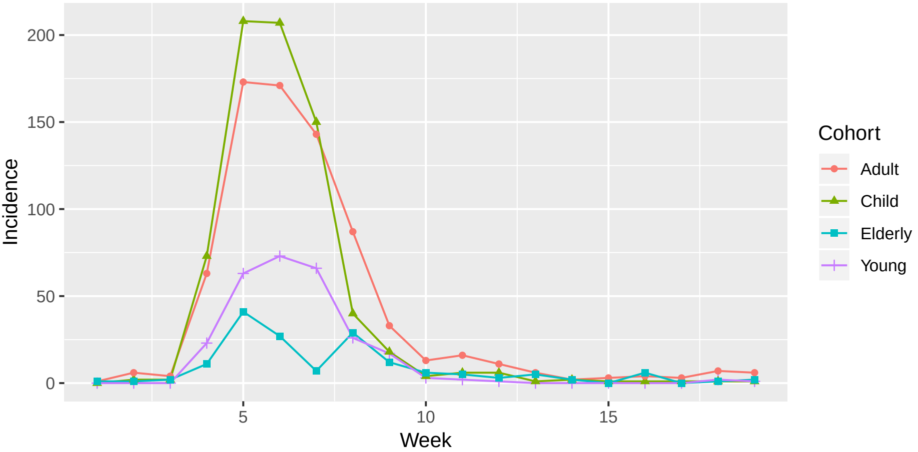
Summarise Data

```
wk_tot <- t_inc %>% group_by(Week) %>%
  summarise(Incidence=sum(Incidence)) %>%
  arrange(desc(Incidence))
slice(wk_tot,1:6)
```

```
## # A tibble: 6 x 2
##    Week Incidence
##   <dbl>     <dbl>
## 1     5       485
## 2     6       478
## 3     7       366
## 4     8       182
## 5     4       170
## 6     9        80
```

Plot tidy data

```
ggplot(t_inc,aes(x=Week,y=Incidence,color=Cohort,
      shape=Cohort)) + geom_line() + geom_point()
```

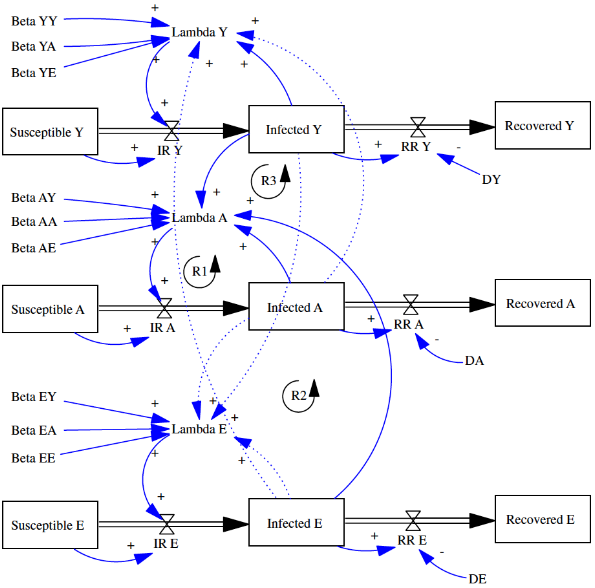


Descriptive Statistics

```
t_coh <- t_inc %>%
           group_by(Cohort) %>%
           summarise(TotalInfected=sum(Incidence),
                     PeakValue=max(Incidence),
    PeakWeek=Week[which(Incidence==max(Incidence))],
                     AvrValue=mean(Incidence),
                     SD=sd(Incidence))
t_coh
```

```
## # A tibble: 4 x 6
##   Cohort  TotalInfected PeakValue PeakWeek AvrValue    SD
##   <chr>           <dbl>     <dbl>    <dbl>    <dbl> <dbl>
## 1 Adult             752       173        5    39.6   59.3
## 2 Child             724       208        5    38.1   70.1
## 3 Elderly           161        41        5     8.47  11.4
## 4 Young             277        73        6    14.6   24.9
```

(2) Analysing Simulation Output



Simulation results - many columns

```
##  [1] "Time"               "Beta AA"           "Beta AE"
##  [4] "Beta AY"            "Beta EA"           "Beta EE"
##  [7] "Beta EY"            "Beta YA"           "Beta YE"
## [10] "Beta YY"            "CE AA"             "CE AE"
## [13] "CE AY"              "CE EA"             "CE EE"
## [16] "CE EY"              "CE YA"             "CE YE"
## [19] "CE YY"              "DA"                "DE"
## [22] "DY"                 "Infected A"        "Infected E"
## [25] "Infected Y"         "IR A"              "IR E"
## [28] "IR Y"               "Lambda A"          "Lambda E"
## [31] "Lambda Y"           "Pop A"             "Pop E"
## [34] "Pop Y"              "Prop A Infected"   "Prop E Infecte
## [37] "Prop Y Infected"    "Recovered A"       "Recovered E"
## [40] "Recovered Y"        "RR A"              "RR E"
## [43] "RR Y"               "Susceptible A"     "Susceptible E"
## [46] "Susceptible Y"      "Total Population"
```

Selecting the stocks

```
out <- res %>%
          select(Time,starts_with("Susceptible"),
                      starts_with("Infected"),
                      starts_with("Recovered"))
glimpse(out)
```

```
## Observations: 161
## Variables: 10
## $ Time            <dbl> 0.000, 0.125, 0.250, 0.375, 0.500,
## $ `Susceptible A` <dbl> 50000, 50000, 50000, 50000, 50000,
## $ `Susceptible E` <dbl> 25000, 25000, 25000, 25000, 25000,
## $ `Susceptible Y` <dbl> 25000, 25000, 25000, 25000, 25000,
## $ `Infected A`    <dbl> 0.00000, 0.00000, 0.01562, 0.05469
## $ `Infected E`    <dbl> 0.0000, 0.1250, 0.2891, 0.5083, 0.8
## $ `Infected Y`    <dbl> 1.000, 1.312, 1.738, 2.321, 3.124,
## $ `Recovered A`   <dbl> 0.00000, 0.00000, 0.00000, 0.00097
```

Convert to tidy format

```
out_td <- out %>%
         gather(key=Variable,value = Amount,
                `Susceptible A`:`Recovered Y`)
slice(out_td,1:5)
```

```
## # A tibble: 5 x 3
##    Time Variable       Amount
##   <dbl> <chr>           <dbl>
## 1 0     Susceptible A   50000
## 2 0.125 Susceptible A   50000
## 3 0.25  Susceptible A   50000
## 4 0.375 Susceptible A   50000
## 5 0.5   Susceptible A   50000
```

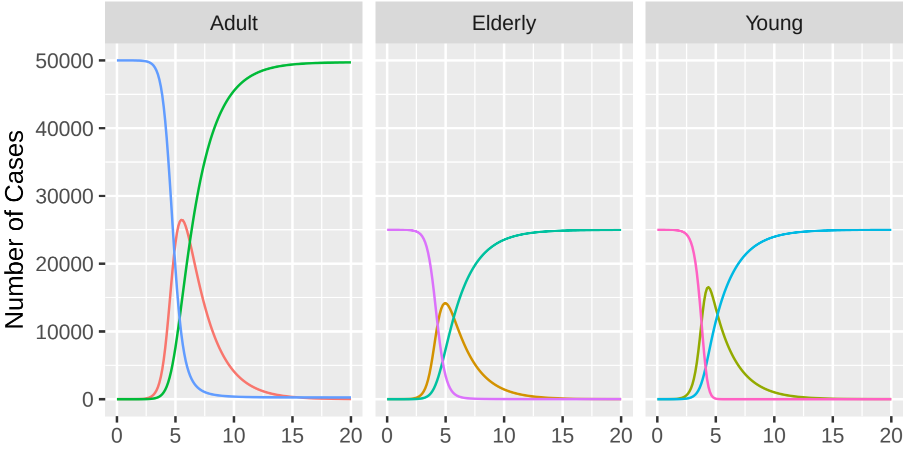
Add cohort and stock information

```
new_td <- out_td %>%
          mutate(Cohort=case_when(
                  grepl("A$",Variable) ~ "Adult",
                  grepl("E$",Variable) ~ "Elderly",
                  grepl("Y$",Variable) ~ "Young"),
                 Class=case_when(
                  grepl("^S",Variable) ~ "Susceptible",
                  grepl("^I",Variable) ~ "Infected",
                  grepl("^R",Variable) ~ "Recovered"))
slice(new_td,1:2)
```

```
## # A tibble: 2 x 5
##    Time Variable      Amount Cohort Class
##   <dbl> <chr>          <dbl> <chr>  <chr>
## 1 0     Susceptible A  50000 Adult  Susceptible
## 2 0.125 Susceptible A  50000 Adult  Susceptible
```

Display chart

```
ggplot(new_td) + geom_path(aes(x=Time,y=Amount,
  colour=Variable))+ylab("Number of Cases")+
  facet_wrap(~Cohort)+guides(colour=F)
```



(3) Exploring Sensitivity Data

```
d <- read_tsv("../../11 simulation/code/sdr_paper1/data/Sensit
dim(d)
```

```
## [1] 200 244
```

```
d[1:3,1:5]
```

```
## # A tibble: 3 x 5
##   Simulation    R0     VF `T1 Infected` `T2 Infected`
##        <dbl> <dbl>  <dbl>         <dbl>         <dbl>
## 1          1  2.76 0.0263             1          1.11
## 2          2  2.66 0.0739             1          1.10
## 3          3  4.06 0.159              1          1.19
```

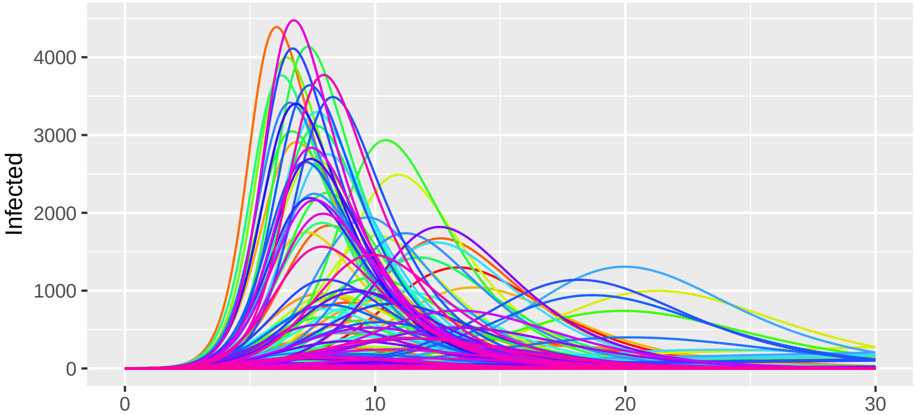
Convert to Tidy Data

```
START_TIME <- 0
DT <- 0.125

td <- gather(d,TimeVariable,Value,-(Simulation:VF)) %>%
  mutate(TSeq=parse_integer(
    str_extract(TimeVariable,"\\d+"))) %>%
  mutate(SimTime=START_TIME+(TSeq-1)*DT) %>%
  separate(TimeVariable,into = c("T","Variable")) %>%
  select(Simulation,SimTime,R0,VF,Variable,Value) %>%
  arrange(Simulation,SimTime)

slice(td,1:2)
```

```
## # A tibble: 2 x 6
##   Simulation SimTime    R0     VF Variable Value
##        <dbl>   <dbl> <dbl>  <dbl> <chr>    <dbl>
## 1          1     0    2.76 0.0263 Infected     1
```

Display simulation traces

```
ggplot(td,aes(x=SimTime,y=Value,color=Simulation)) +
  geom_path() +  ylab("Infected") +
  scale_colour_gradientn(colours=rainbow(10))+
  xlab("Time (Days)")  + guides(color=FALSE)
```



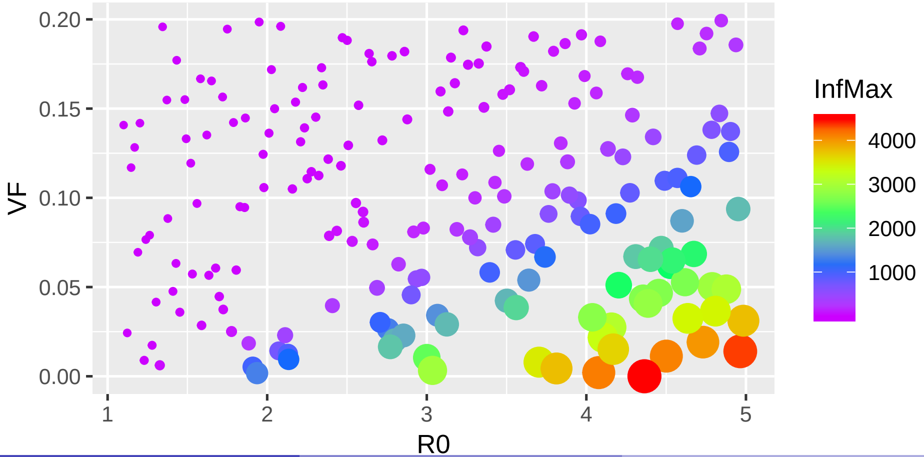
Calculate Summary Data

```
i_td <- td %>% group_by(Simulation) %>%
                 summarise(InfMax=max(Value),
                           R0=R0[1],
                           VF=VF[1])
slice(i_td,1:5)
```

```
## # A tibble: 5 x 4
##   Simulation InfMax    R0     VF
##        <dbl>  <dbl> <dbl>  <dbl>
## 1          1 1298.   2.76 0.0263
## 2          2   78.9  2.66 0.0739
## 3          3  127.   4.06 0.159
## 4          4  367.   2.69 0.0496
## 5          5   14.0  3.23 0.194
```

Explore Parameter Space

```
ggplot(data=i_td,aes(x=R0,y=VF,size=InfMax,colour=InfMax)) +
  geom_point() + guides(size=F) +
  scale_colour_gradientn(colours=rev(rainbow(5)))
```



Summary

Conclusion

While R is primarily viewed as a toolset to support data scientists, innovative new libraries such as the `tidyverse` can be leveraged to support the system dynamics model-building process. This paper has shown how time series data can be accessed and manipulated, and how the entire model output from a simulation run can be processed for informative summaries and for data visualisation. A further application of the `tidyverse` is to support the process of analysing large datasets produced through sensitivity analysis of system dynamics models.