Analysis for CCFP paper

Peter S. Hovmand

Feb 27, 2022

Overview

This document was written and generated as an R Markdown document and exported to Word. The document includes all the code and results for replicating the results in the paper.

Requirements

The following are required for replicating this simulation study:

1. Stella Architect (2.1.1 or higher) or Stella Simulator (2.1.1 or higher)
2. R (available at https://cran.r-project.org/)
3. RStudio (available at https://www.rstudio.com/)
4. Developmental-Transitions-2-2-9.stmx model

```
require(readr)    # package for reading in csv files as tibble data frame
```

```
## Loading required package: readr
```

```
require(readxl)   # reads Excel worksheets
```

```
## Loading required package: readxl
```

```
require(dplyr)    # for manipulating data frames, etc.
```

```
## Loading required package: dplyr
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
require(ggplot2)  # for creating interactive plots
```

```
## Loading required package: ggplot2
```

```
require(reshape2) # needed to reshape data for plotting with ggplot2
```

```
## Loading required package: reshape2
```

Setting up individual cases

The individual cases were generated by interactively varying the parameters and initial conditions to generate a representative set of developmental trajectories for cognitive vulnerabilities and family support. The values for initial conditions and parameters for each case were then exported from Stella Architect into an Excel worksheet that can be read by R. Having the values in an external file provides a flexible approach to add/modify cases as the model develops and replicate the analyses.

```
# read in the Excel file
cases.df <- read_excel("Cases.xlsx")

# print values of parameters and initial conditions used for each case
cases.df
```

```
## # A tibble: 6 x 7
##    Case Cognitive_Vulnera~ Family_Response_~ Family_Support_~ Initial_Cognitive~
##   <dbl>              <dbl>             <dbl>            <dbl>              <dbl>
## 1     1              0.125             0.125            0.125                 25
## 2     2              0.5              25               25                     25
## 3     3              1.25              0.1             12                     25
## 4     4              2                25               25                     25
## 5     5              2                25               25                     25
## 6     6              2                25               25                     25
## # ... with 2 more variables: Initial_Family_Support <dbl>,
## #   Onset_of_Maladaptive_Behaviors_Delay <dbl>
```

```
# create a vector that can be used later to iterate through the cases for
# simulation, plotting, etc
cases.vec<-1:nrow(cases.df)
```

Simulate a baserun for each case

The baserun for each case is simulated with the results imported via a .csv file and saved in the dataframe results.baserun.

```
results.baseruns <- NULL # create an empty objects to store results

# iterate over the list of cases
for (i in cases.vec) {
  # print progress
  cat("Case =",i,"\n")

  # pull the parameters from the row in the case data frame
  parms <- cases.df %>%             # start wtih the data frame
            filter(Case==i) %>%    # select row for case = i
            select(!Case)          # drop the Case column

  # write the parameters to the Stella import file
  write_csv(parms, file="Parms.csv")

  # run the simulation
  system2("stella_simulator",
        args = "-r Developmental-Transitions-2-2-10.stmx",
        env = "PATH=$PATH:/Applications/Stella_Simulator",
        wait=TRUE)

  # read the results and add them to the data frame

  results.baseruns<-rbind(results.baseruns,
                          tibble(Case=i,read_csv("Results.csv")))
}
```

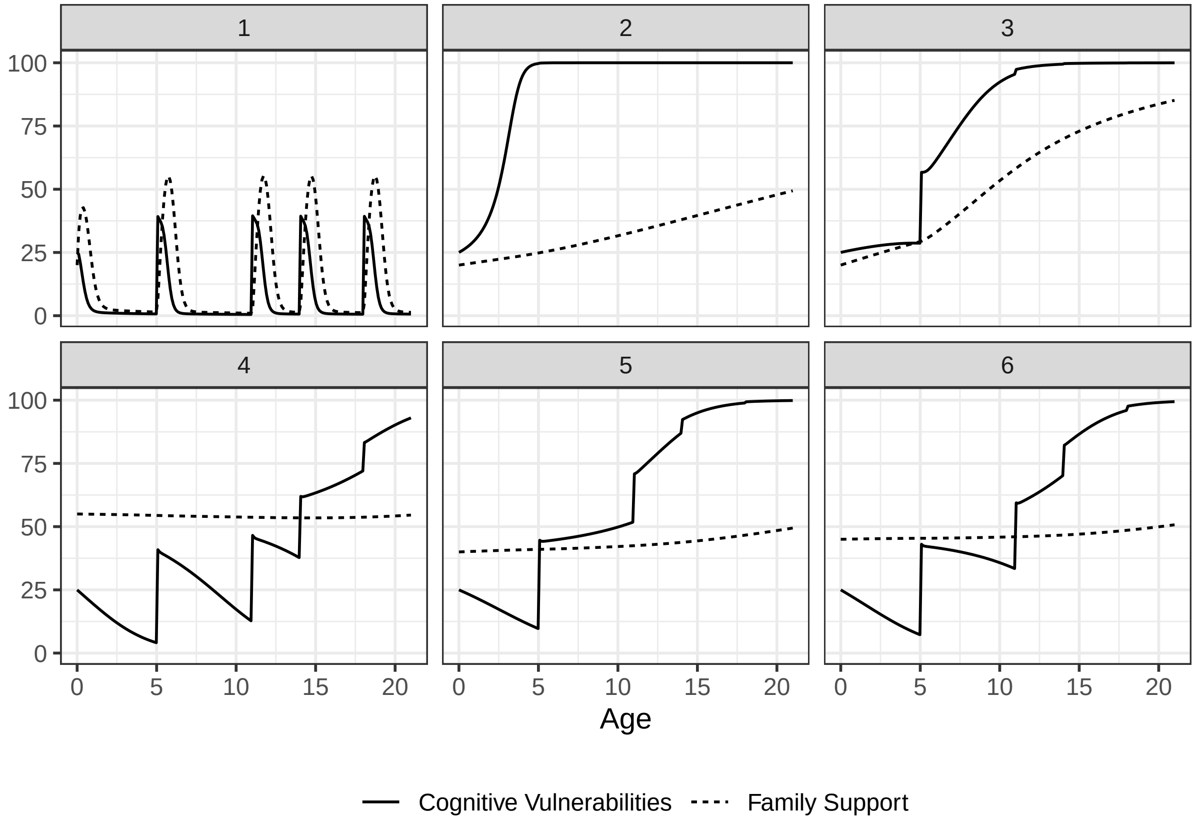
```
## Case = 1
## Case = 2
## Case = 3
## Case = 4
## Case = 5
## Case = 6
```

Plot results

Results are plotted using `ggplot2` package showing the dynamics of cognitive vulnerabilities and family support for each hypothetical case.

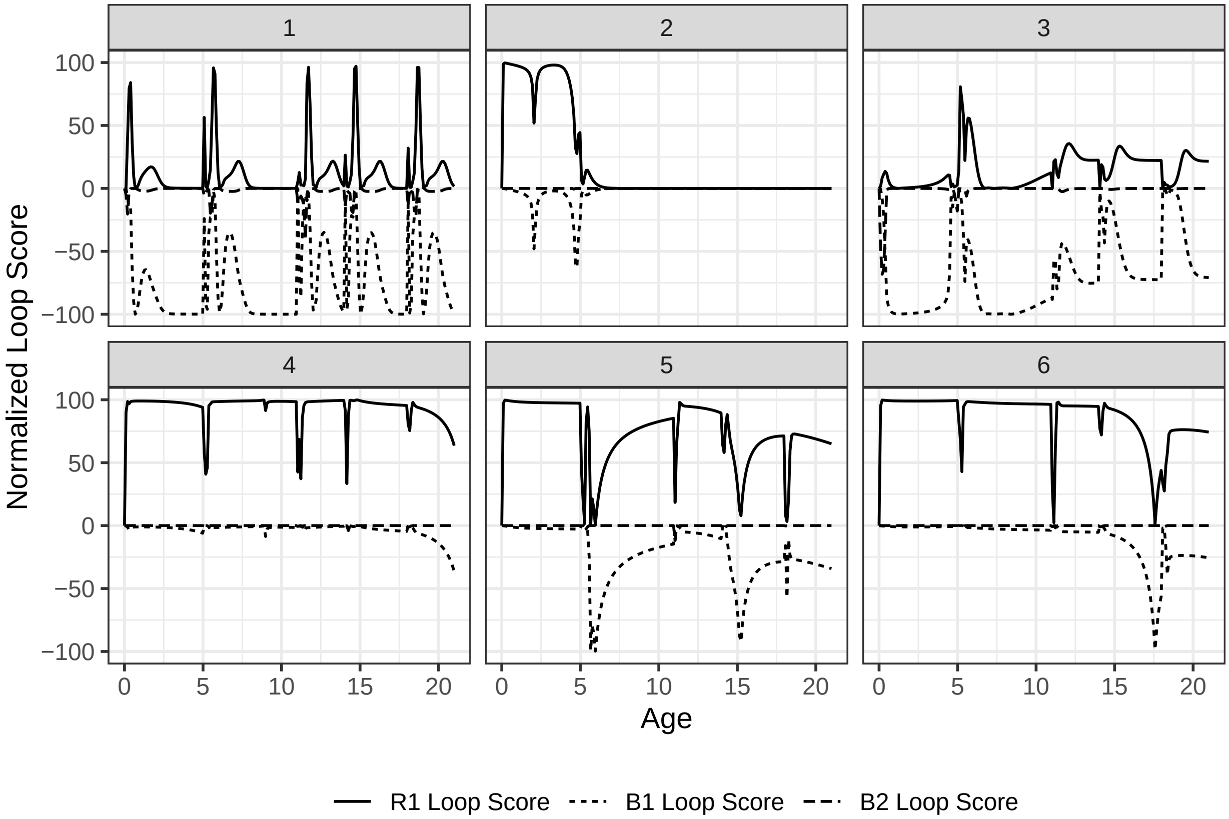
```
# Select a vector of time points to plot
t.plot <- unique(results.baseruns$Years)
t.plot <- t.plot[seq(1,length(t.plot),100)]

# Plot the results
results.baseruns %>%
  select(Case, Years, `Cognitive Vulnerabilities`, `Family Support`) %>%
  filter(Years %in% t.plot) %>%
  melt(id=c("Case","Years")) %>%
    ggplot(aes(x=Years, y=value)) +
    geom_line(aes(linetype = variable)) +
    ylab("") +
    xlab("Age") +
    facet_wrap(~Case) +
    theme_bw() +
    theme(legend.position = "bottom",
          legend.title = element_blank())
```



Loop scores for each of the three major loops are then plotted for each case.

```
results.baseruns %>%
  select(Case, Years, `R1 Loop Score`, `B1 Loop Score`, `B2 Loop Score`) %>%
  filter(Years %in% t.plot) %>%
  melt(id=c("Case","Years")) %>%
    ggplot(aes(x=Years, y=value)) +
    geom_line(aes(linetype = variable)) +
    ylab("Normalized Loop Score") +
    xlab("Age") +
    facet_wrap(~Case) +
    theme_bw() +
    theme(legend.position = "bottom",
          legend.title = element_blank())
```


1
2
3
4
5
6
Normalized Loop Score
100
50
0
−50
−100
0
5
10
15
20
Age
R1 Loop Score
B1 Loop Score
B2 Loop Score