Ionotropic receptors as the driving force behind human synapse establishment

Supplementary Material

Lucas H. Viscardi Danilo O. Imparato Maria Cátira Bortolini
Rodrigo J. S. Dalmolin

Abstract

Model uncertainty and limited data are fundamental challenges to robust management of human intervention in a natural system. These challenges are acutely highlighted by concerns that many ecological systems may contain tipping points, such as Allee population sizes. Before a collapse, we do not know where the tipping points lie, if they exist at all. Hence, we know neither a complete model of the system dynamics nor do we have access to data in some large region of state-space where such a tipping point might exist.

Contents

Project structure

This is the title page

Preprocessing

This topic refers mainly to data wrangling done before the actual analysis with the intent of making it simpler.

Eukaryota species tree

We opted to use the TimeTree database in order to obtain an standardized Eukaryota species tree. However, some species were not present in it, so we devised a way to fill them in based on NCBI Taxonomy data.

NCBI Taxonomy tree

First we preprocess NCBI Taxonomy data to leave only STRING eukaryotes, thus making the task easier.

Downloading data

```
download_if_missing("http://ftp.ncbi.nlm.nih.gov/pub/taxonomy/taxdump.tar.gz")
download_if_missing("stringdb-static.org/download/species.v11.0.txt")

untar("download/taxdump.tar.gz", exdir = "download/taxdump")
```

Loading data

Table 1: Lists all organisms in STRING v11.

Location: data-raw/download/species.v11.0.txt Source: stringdb-static.org/download/species.v11.0.txt					
#	Col. name	Col. type	Used?	Example	Description
1	taxid	character	yes	9606	NCBI Taxonomy identifier
2	string_type	character	no	core	if the genome of this species is core or periphery
3	string_name	character	yes	Homo sapiens	STRING species name
4	ncbi_official_name	character	no	Homo sapiens	NCBI Taxonomy species name

Table 2: Links outdated taxon IDs to corresponding new ones.

Location: data-raw/download/taxdump/merged.dmp Source: ftp.ncbi.nlm.nih.gov/pub/taxonomy/taxdump.tar.gz					
#	Col. name	Col. type	Used?	Example	Description
1	taxid	character	yes	9606	id of node that has been merged
2	new_taxid	character	yes	core	id of node that is the result of merging

Table 3: Represents taxonomy nodes.

Location: data-raw/download/taxdump/nodes.dmp Source: ftp.ncbi.nlm.nih.gov/pub/taxonomy/taxdump.tar.gz					
#	Col. name	Col. type	Used?	Example	Description
1	taxid	character	yes	2	node id in NCBI taxonomy database
2	parent_taxid	character	yes	131567	parent node id in NCBI taxonomy database
3	rank	character	no	superkingdom	rank of this node
4	...		no		(too many unrelated fields)

Table 4: Links taxon IDs to actual species names.

Location: data-raw/download/taxdump/names.dmp Source: ftp.ncbi.nlm.nih.gov/pub/taxonomy/taxdump.tar.gz					
#	Col. name	Col. type	Used?	Example	Description
1	taxid	character	yes	2	the id of node associated with this name
2	name	character	yes	Monera	name itself
3	unique_name	character	no	Monera <bacteria>	the unique variant of this name if name not unique
4	name_class	character	yes	scientific name	type of name

```
string_species <- read_tsv(
  "download/species.v11.0.txt",
  skip = 1,
  col_names = c(
    "taxid",
    "string_type",
    "string_name",
    "ncbi_official_name"
  ),
  col_types = cols_only(
    taxid = "c",
    string_name = "c"
  )
)

# these .dmp files are very tricky to read
# the following read_delims are very hacky
ncbi_merged_ids <- read_delim(
  "download/taxdump/merged.dmp",
  delim = "|",
  trim_ws = TRUE,
  col_names = c("taxid","new_taxid"),
  col_types = "cc"
)

ncbi_edgelist <- read_delim(
  "download/taxdump/nodes.dmp",
  skip = 1,
  delim = "|",
```

```
  trim_ws = TRUE,
  col_names = c("n1","n2"),
  col_types = "cc"
)

ncbi_taxon_names <- read_delim(
  "download/taxdump/names.dmp",
  delim = "|",
  trim_ws = TRUE,
  col_names = c("name","ncbi_name","type"),
  col_types = "cc-c"
)
```

Updating STRING taxon IDs
Some organisms taxon IDs are outdated in STRING. We must update them to work with the most recent NCBI Taxonomy data.

```
string_species %<>%
  left_join(ncbi_merged_ids) %>%
  mutate(new_taxid = coalesce(new_taxid, taxid))
```

Creating tree graph
The first step is to create a directed graph representing the NCBI Taxonomy tree.

```
# leaving only "scientific name" rows
ncbi_taxon_names %<>%
  filter(type == "scientific name") %>%
  select(name, ncbi_name)

# finding Eukaryota taxid
eukaryota_taxon_id <- subset(ncbi_taxon_names, ncbi_name == "Eukaryota", "name", drop = TRUE)

# creating graph
g <- graph_from_data_frame(ncbi_edgelist[,2:1], directed = TRUE, vertices = ncbi_taxon_names)

# easing memory
rm(ncbi_edgelist, ncbi_merged_ids)
```

Traversing the graph
The second step is to traverse the graph from the Eukaryota root node to STRING species nodes. This automatically drops all non-eukaryotes and results in a species tree representing only STRING eukaryotes (476).

```
eukaryote_root <- V(g)[eukaryota_taxon_id]
eukaryote_leaves <- V(g)[string_species[["new_taxid"]]]

# not_found <- subset(string_species, !new_taxid %in% ncbi_taxon_names$name)

eukaryote_paths <- shortest_paths(g, from = eukaryote_root, to = eukaryote_leaves, mode = "out")$vpath

eukaryote_vertices <- eukaryote_paths %>% unlist %>% unique
```

```
eukaryote_tree <- induced_subgraph(g, eukaryote_vertices, impl = "create_from_scratch")
```

Saving
Saving `ncbi_tree` and `string_eukaryotes` for package use. These data files are documented by the package. We also create a plain text file `476_ncbi_eukaryotes.txt` containing the updated names of all 476 STRING eukaryotes. This file will be queried against the TimeTree website.

```
ncbi_tree <- treeio::as.phylo(eukaryote_tree)

# plot(ncbi_tree %>% ape::ladderize(), type="cladogram")

string_eukaryotes <- string_species %>%
  filter(new_taxid %in% ncbi_tree$tip.label) %>%
  inner_join(ncbi_taxon_names, by = c("new_taxid" = "name"))

write(string_eukaryotes[["ncbi_name"]],"476_ncbi_eukaryotes.txt")

usethis::use_data(ncbi_tree, overwrite = TRUE)
```

```
## <U+2714> Setting active project to 'C:/R/neuro'
## <U+2714> Saving 'ncbi_tree' to 'data/ncbi_tree.rda'
```

```
usethis::use_data(string_eukaryotes, overwrite = TRUE)
```

```
## <U+2714> Saving 'string_eukaryotes' to 'data/string_eukaryotes.rda'
```

Duplicated Genera

Some species from different kingdoms may share the same genus name. These genera must be noted down because one of the ways we fill in missing species is by looking at genera names.

Hybrid tree

Explanation

Gene selection and annotation

Gene selection and annotation

Neuroexclusivity

Explanation

Expression

Pathways

COG data

Network

Analysis

Analysis

```
library(magrittr)
library(neurotransmissionevolution)
```

```
## Warning: replacing previous import 'dplyr::union' by 'igraph::union' when
## loading 'neurotransmissionevolution'
```

```
## Warning: replacing previous import 'dplyr::as_data_frame' by
## 'igraph::as_data_frame' when loading 'neurotransmissionevolution'
```

```
## Warning: replacing previous import 'dplyr::groups' by 'igraph::groups' when
## loading 'neurotransmissionevolution'
```

Test tab2

```
tibble::tribble(
  ~col_name,   ~col_type, ~used, ~ex,                  ~desc,
  "string_id", "numeric", "yes", "9606.ENSP000000666", "This represents the string protein identifier."
  "length",    "numeric", "yes", "666",                "This represents the protein size"
) %>% print_dataframe_specification("a", "b", "caption")
```

```
## \begin{table}[!h]
##
## \caption{\label{tab:unnamed-chunk-1}caption}
## \centering
## \resizebox{\linewidth}{!}{
## \begin{tabular}[t]{rllll>{\raggedright\arraybackslash}p{20em}}
## \toprule
## \multicolumn{6}{c}{\cellcolor[HTML]{EEEEEE}{\textbf{Location: a}}} \\
## \multicolumn{6}{c}{\cellcolor[HTML]{EEEEEE}{\textbf{Source: b}}} \\
## \cmidrule(l{3pt}r{3pt}){1-6}
## \# & Col. name & Col. type & Used? & Example & Description\\
## \midrule
## \rowcolor{gray!6}  1 & string\_id & numeric & yes & 9606.ENSP000000666 & This represents the string p
## 2 & length & numeric & yes & 666 & This represents the protein size\\
## \bottomrule
## \end{tabular}}
## \end{table}
```