

TXT2PED

17/November 2019

Version 1.1

Date 2019-11-17

Title TXT2PED

Requirements

R version: $\geq 3.6.1$

Required libraries for PedigreeEngine.R

Name	Version
quadprog	$\geq 1.5.7$
Matrix	$\geq 1.2.17$
kinship2	$\geq 1.8.4$

Required libraries for Shiny App

Name	Version
shiny	$\geq 1.4.0$
shinyFiles	$\geq 0.7.3$
shinyjs	≥ 1.0
dplyr	$\geq 0.8.3$
ggplot2	$\geq 3.2.1$
shinydashboard	$\geq 0.7.1$
leaflet	$\geq 2.0.2$

License GPL (≥ 2)

URL <https://github.com/cloudyaaron/6112project>

Authors

Ziqi Yang, Hujia Wang,

Jaiden Choy

Description

Pedigree charts are used in Medicine to record a patient's family history. They are particularly useful to document genetic diseases that can be inherited from one generation to the next.

In most clinical settings, pedigree charts are still drawn by hand on paper, which limits their utility. Pedigree drawing software tools exist but are complicated to use and/or record their data in proprietary formats that are unsuitable for medical research purposes.

In this project, we aim to develop a pedigree drawing engine that can create pedigree charts from simple textual descriptions. The concept is similar to Graphviz (<https://www.graphviz.org/>) and mermaid (<https://mermaidjs.github.io/>), two projects that allow creating graphs and diagrams using a simple text syntax.

Table of Contents

1.0 Instructions and Installation.....	4
1.1 Access through shinyapps.io	4
1.2 Clone from our GitHub repository.....	4
1.3 Notes	4
2.0 Input Commands	5
2.1 Attributes input:	5
2.2 Relation input:	5
2.3 Notes	6
3.0 Walkthrough.....	6
4.0 PedigreeEngine.R as a package	9
4.1 print_ped(df)	9
4.2 producePED(filepath)	9
4.3 Producegraph(df,d,position,arg,legendsize)	10

1.0 Instructions and Installation

There are currently two ways to use the TXT2PED.

1.1 Access through shinyapps.io

URL: https://6112project.shinyapps.io/Pedigree_project/

Limits to this method:

Due to the limitation of having a free account with shinyapps.io, our app can only be accessed for 25 hours per month.

Error feedback will not be given by this method.

We highly suggest deploying the server at local.

1.2 Clone from our GitHub repository.

URL: <https://github.com/cloudyaaron/6112project>

This can be run locally or deployed in LAN.

Run locally:

1. Clone the repository from GitHub.
2. Install all the required packages.
3. Open shiny.R file by RStudio.
4. Click "Run App" on the top right corner of the script window.
5. Open a web browser or use the built-in browser to start the app.

Deploy in LAN:

Deploy in LAN is possible however it is not integrated in our code. Use the provided link to modify the code and deploy.

URL: <https://shiny.rstudio.com/articles/deployment-local.html>

1.3 Notes

After creating a new file, this new file needs to be loaded to start creating the pedigree.

If the web page is accidentally closed and Pedigree Engine was run locally, the pedigree graph and ped file may be found in the ./out/ folder.

The python version is no longer in being developed. However, the python code is in the ice bucket folder for those who really need them.

2.0 Input Commands

Sequence of input is important.

Different sequence of input may result in different graph.

2.1 Attributes input:

1. A gender_is male
2. A,B,C gender_is female
3. A name_is "John Doe"
4. A DOB_is "13 Oct 1970"
5. A deceased_is true
6. A,B,C deceased_is true
7. A affected_is diseaseA
8. A affected_is diseaseA,B,C
9. A,B,C affected_is diseaseX,Y,Z
10. A ad_is "addition text "
11. A change_node_name B (will keep all attributes and relation)

2.2 Relation input:

1. A father_of B
2. A father_of B,C,D
3. A mother_of B
4. A mother_of B,C,D
5. A partner_of B (A will be B's children parents) (B must have children)
6. A brother_of B (Only can be use if B has parents)
7. A sister_of B (Only can be use if B has parents)
8. A m_twin B (monozygotic twin)
9. A d_twin B (dizygotic twin)
10. A unkonwn_twin B

2.3 Notes

Twin relation and Affected attribute must be assigned at last or after creating every node otherwise an error will rise.

All relation input will create new nodes if node is not in the graph.

For example, A father_of B,C,D will create 4 nodes named A,B,C,D.

But the graph won't show until B,C,D has a mother.

Assign B,C,D a mother Z can be done in:

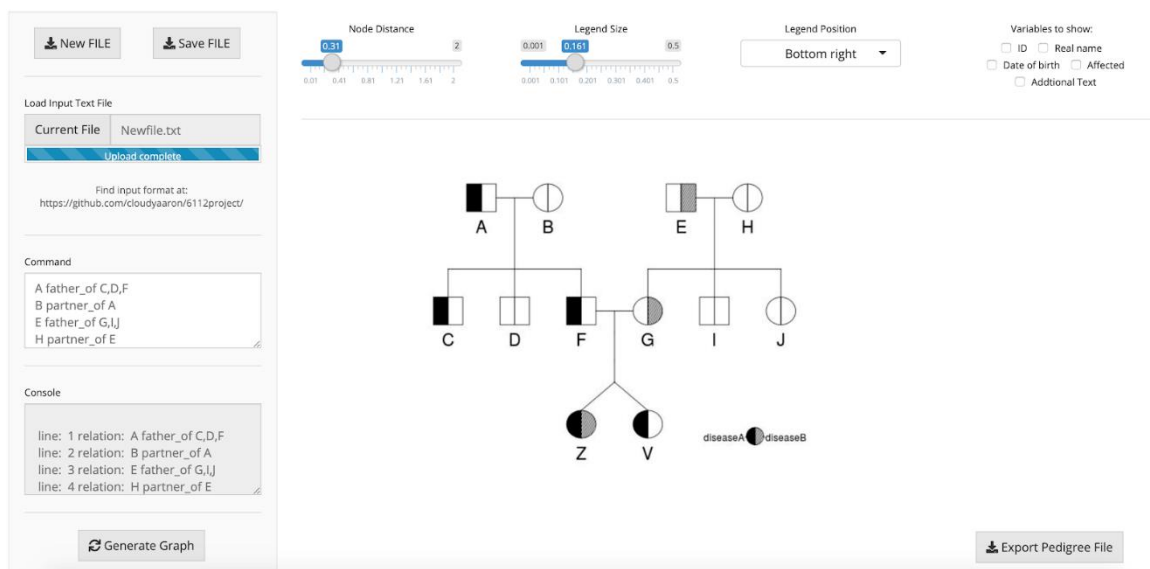
Z mother_of B,C,D

Or Z partner_of A

More sample inputs can be found in sample inputs folder.

3.0 Walkthrough

A walkthrough on how to use TXT2PED using a screenshot of the app as reference.



1. Either click on the “New FILE” button or upload an existing file using the “Current File” upload button. NOTE: The app requires a file to be loaded. Please upload the empty file created by the “New FILE” button.
2. Once a file has been loaded in, a pedigree can be created by editing the “Command” text box using the input commands shown in section 2.0 of this document.
3. If the user wants to save the commands that they have input into the app, they can click the “Save FILE” button which will save the commands as a new text file.
4. When ready to generate the pedigree, click the “Generate Graph” button. The pedigree shown was made using the following commands.

A father_of C,D,F

B partner_of A (Consanguine marriages cannot use this)

E father_of G,I,J

H partner_of E

F father_of Z,V

G mother_of Z,V

C,D,I gender_is male

J,Z,V gender_is female

A,C,F,Z,V affected_is diseaseA (define affected attributes)

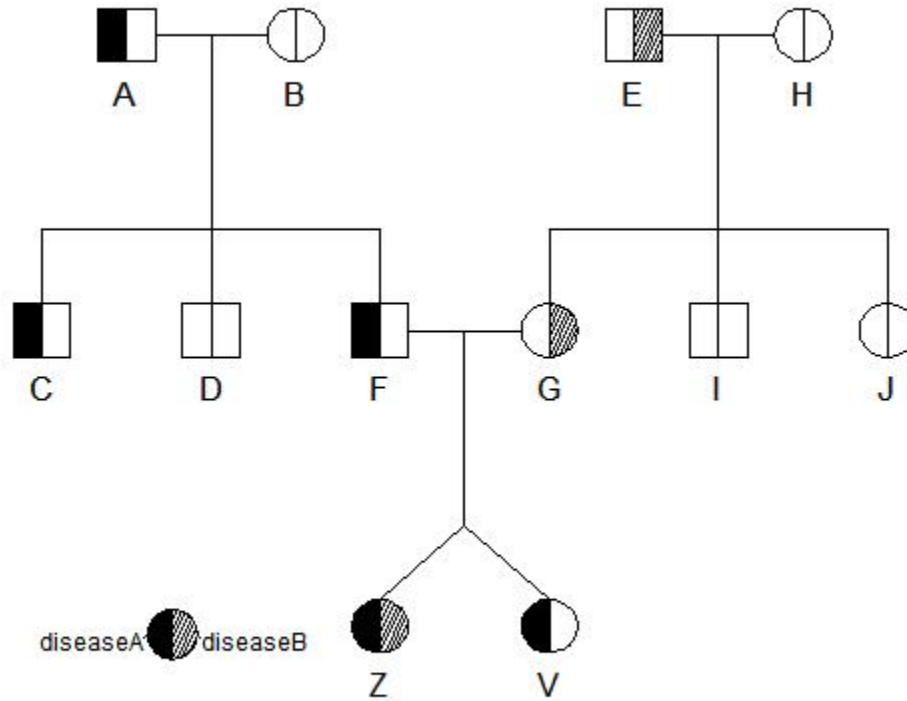
E,G,Z affected_is diseaseB

Z d_twin V (define twins' relationship)

5. The user can customize how large the pedigree is by using the “Node Distance” slider.
6. The user can also customize the legend size and position by using the “Legend Size” slider and “Legend Position” drop down menu respectively.
7. The user can choose to show extra information about each family member in the pedigree by selecting the variables they want using the “Variables to show” checkboxes.
8. When the user wants to save the pedigree, they can right click the image and choose the “Save Image As...” option. This will pull up a window where the user is prompted to give the pedigree a name.

9. The user can click the “Export Pedigree File” button to save the ped file that creates the pedigree made.

Above commands will generate:



4.0 PedigreeEngine.R as a package

PedigreeEngine.R file can be found in ./src/ folder. This file can be copied and used individually if needed. Below are the functions which can be called.

4.1 print_ped(df)

This function takes a data frame as an input and prints the data frame in ped format at ./output/output.ped

df -> A data frame (a data frame generated by the *producePED* function)

This function **returns nothing**.

4.2 producePED(filepath)

This function takes the text input file's path and turns the information into a data frame.

filepath -> Text input file's path (cannot be a relative path)

This function **returns df**.

df -> The data frame that contains the relationships and attributes. Initially 14 columns, max 18 columns.

14 columns are: "ped", "id", "father", "mother", "sex", "affected", "deceased", "twin", "node", "name", "dob", "partner", "sg", "ad"

The 4 columns added are for new diseases.

4.3 Producegraph(df,d,position,arg,legendsize)

This function takes a data frame and a set of arguments as an input and returns the pedigree.

df -> The data frame that is produced by producePED function.

d -> Distance between nodes, prefer from 0.01 to 2.

position -> Where the legend is located on the graph. Only accepted as "topright", "topleft", "bottomright", "bottomleft"

arg -> A list that may contain (name, id, dob, affect, ad) or empty. If the list contains any of these features, the graph will show extra content.

legendsize -> The legends' size on the graph. Preferred range between 0.001 to 0.05.