

 branch: **master** ▾

Final_project / **slave_network.Rmd**



 **scollini** 3 days ago Added R markdown file

1 contributor

253 lines (214 sloc) | 12.278 kb

Raw

Blame

History



title	author	date	output
Network Analysis of the Enslaved Community at Mount Vernon, 1786-1799	Sara Collini	December 3, 2014	html_document

Introduction

This project uses network analysis in R to explore community amoung Mount Vernon slaves. The graph investigates two main communities: kinship relations and work relations. The information on the slaves draws from a compiled census created by the Mount Vernon staff at the Fred W. Smith National Library for the Study of George Washington.

First, we load the necessary libraries.

```
library(igraph)
library(dplyr)
library(tidyr)
library(stringr)
library(historydata)
library(ggplot2)
```

Data Manipulation

Read in the data

The data for this project exists in three csv files. The first file contains every slave with at least one kinship relation on Mount Vernon, from 1786-1799 .

```
slaverel <- read.csv("~/Desktop/Clio-3/finalproject/slaveRelationships.csv", stringsAsFactors = FALSE)
head(slaverel)
```


The second contains every slave and the respective static information, such as gender and owner, for 1786 and 1799.

```
total_slave <- read.csv("~/Desktop/Clio-3/finalproject/clean.csv", stringsAsFactors = FALSE) %>%
  select(id, Gender, Birth.Year, Owner)
head(total_slave)
```

This file contains both the 1786 and 1799 censuses with the relevant slaves. This means the slaves present in both













censuses are repeated but might have a different skill or farm.

```
slave_census <- read.csv("~/Desktop/Clio-3/finalproject/census.csv", stringsAsFactors = FALSE) %>%
  select(id, Gender, Birth.Year, Skill, Farm, Census, Owner)
head(slave_census)
```

Data Joining

Next, we must separate out the two most important and complete censuses: 1786 and 1799. We select only the variables that change between the two censuses.

```
slave_1786 <- slave_census %>%
  filter(Census == "1786") %>%
  select(id, Census, Skill, Farm)

slave_1799 <- slave_census %>%
  filter(Census == "1799") %>%
  select(id, Census, Skill, Farm)
```

Join the two censuses into one dataframe by the "id" variable in order to access that information on the network graph.

```
total_slave <- total_slave %>%
  left_join(slave_1786, by = "id") %>%
  left_join(slave_1799, by = "id")
head(total_slave)
```

Create a Graph Object

We create a graph object of the slaverel dataframe in order to graph the kinship relations between the slaves.

```
slave_graph <- graph.data.frame(slaverel, directed = "FALSE")
```

In order to add those slaves without kinship relations, we must add them as extra vertices to the graph object.

```
slave_graph <- slave_graph + vertex("Adam B", "Adam C", "Anthony A", "Austin A", "Bath A", "Bristol A", "Bruno A", "Bunbury A", "Carter A", "Carter B", "Carter C", "Carter D", "Carter E", "Carter F", "Carter G", "Carter H", "Carter I", "Carter J", "Carter K", "Carter L", "Carter M", "Carter N", "Carter O", "Carter P", "Carter Q", "Carter R", "Carter S", "Carter T", "Carter U", "Carter V", "Carter W", "Carter X", "Carter Y", "Carter Z", "Carter AA", "Carter AB", "Carter AC", "Carter AD", "Carter AE", "Carter AF", "Carter AG", "Carter AH", "Carter AI", "Carter AJ", "Carter AK", "Carter AL", "Carter AM", "Carter AN", "Carter AO", "Carter AP", "Carter AQ", "Carter AR", "Carter AS", "Carter AT", "Carter AU", "Carter AV", "Carter AW", "Carter AX", "Carter AY", "Carter AZ", "Carter BA", "Carter BB", "Carter BC", "Carter BD", "Carter BE", "Carter BF", "Carter BG", "Carter BH", "Carter BI", "Carter BJ", "Carter BK", "Carter BL", "Carter BM", "Carter BN", "Carter BO", "Carter BP", "Carter BQ", "Carter BR", "Carter BS", "Carter BT", "Carter BU", "Carter BV", "Carter BW", "Carter BX", "Carter BY", "Carter BZ", "Carter CA", "Carter CB", "Carter CC", "Carter CD", "Carter CE", "Carter CF", "Carter CG", "Carter CH", "Carter CI", "Carter CJ", "Carter CK", "Carter CL", "Carter CM", "Carter CN", "Carter CO", "Carter CP", "Carter CQ", "Carter CR", "Carter CS", "Carter CT", "Carter CU", "Carter CV", "Carter CW", "Carter CX", "Carter CY", "Carter CZ", "Carter DA", "Carter DB", "Carter DC", "Carter DD", "Carter DE", "Carter DF", "Carter DG", "Carter DH", "Carter DI", "Carter DJ", "Carter DK", "Carter DL", "Carter DM", "Carter DN", "Carter DO", "Carter DP", "Carter DQ", "Carter DR", "Carter DS", "Carter DT", "Carter DU", "Carter DV", "Carter DW", "Carter DX", "Carter DY", "Carter DZ", "Carter EA", "Carter EB", "Carter EC", "Carter ED", "Carter EE", "Carter EF", "Carter EG", "Carter EH", "Carter EI", "Carter EJ", "Carter EK", "Carter EL", "Carter EM", "Carter EN", "Carter EO", "Carter EP", "Carter EQ", "Carter ER", "Carter ES", "Carter ET", "Carter EU", "Carter EV", "Carter EW", "Carter EX", "Carter EY", "Carter EZ", "Carter FA", "Carter FB", "Carter FC", "Carter FD", "Carter FE", "Carter FF", "Carter FG", "Carter FH", "Carter FI", "Carter FJ", "Carter FK", "Carter FL", "Carter FM", "Carter FN", "Carter FO", "Carter FP", "Carter FQ", "Carter FR", "Carter FS", "Carter FT", "Carter FU", "Carter FV", "Carter FW", "Carter FX", "Carter FY", "Carter FZ", "Carter GA", "Carter GB", "Carter GC", "Carter GD", "Carter GE", "Carter GF", "Carter GG", "Carter GH", "Carter GI", "Carter GJ", "Carter GK", "Carter GL", "Carter GM", "Carter GN", "Carter GO", "Carter GP", "Carter GQ", "Carter GR", "Carter GS", "Carter GT", "Carter GU", "Carter GV", "Carter GW", "Carter GX", "Carter GY", "Carter GZ", "Carter HA", "Carter HB", "Carter HC", "Carter HD", "Carter HE", "Carter HF", "Carter HG", "Carter HH", "Carter HI", "Carter HJ", "Carter HK", "Carter HL", "Carter HM", "Carter HN", "Carter HO", "Carter HP", "Carter HQ", "Carter HR", "Carter HS", "Carter HT", "Carter HU", "Carter HV", "Carter HW", "Carter HX", "Carter HY", "Carter HZ", "Carter IA", "Carter IB", "Carter IC", "Carter ID", "Carter IE", "Carter IF", "Carter IG", "Carter IH", "Carter II", "Carter IJ", "Carter IK", "Carter IL", "Carter IM", "Carter IN", "Carter IO", "Carter IP", "Carter IQ", "Carter IR", "Carter IS", "Carter IT", "Carter IU", "Carter IV", "Carter IW", "Carter IX", "Carter IY", "Carter IZ", "Carter JA", "Carter JB", "Carter JC", "Carter JD", "Carter JE", "Carter JF", "Carter JG", "Carter JH", "Carter JI", "Carter JJ", "Carter JK", "Carter JL", "Carter JM", "Carter JN", "Carter JO", "Carter JP", "Carter JQ", "Carter JR", "Carter JS", "Carter JT", "Carter JU", "Carter JV", "Carter JW", "Carter JX", "Carter JY", "Carter JZ", "Carter KA", "Carter KB", "Carter KC", "Carter KD", "Carter KE", "Carter KF", "Carter KG", "Carter KH", "Carter KI", "Carter KJ", "Carter KK", "Carter KL", "Carter KM", "Carter KN", "Carter KO", "Carter KP", "Carter KQ", "Carter KR", "Carter KS", "Carter KT", "Carter KU", "Carter KV", "Carter KW", "Carter KX", "Carter KY", "Carter KZ", "Carter LA", "Carter LB", "Carter LC", "Carter LD", "Carter LE", "Carter LF", "Carter LG", "Carter LH", "Carter LI", "Carter LJ", "Carter LK", "Carter LL", "Carter LM", "Carter LN", "Carter LO", "Carter LP", "Carter LQ", "Carter LR", "Carter LS", "Carter LT", "Carter LU", "Carter LV", "Carter LW", "Carter LX", "Carter LY", "Carter LZ", "Carter MA", "Carter MB", "Carter MC", "Carter MD", "Carter ME", "Carter MF", "Carter MG", "Carter MH", "Carter MI", "Carter MJ", "Carter MK", "Carter ML", "Carter MM", "Carter MN", "Carter MO", "Carter MP", "Carter MQ", "Carter MR", "Carter MS", "Carter MT", "Carter MU", "Carter MV", "Carter MW", "Carter MX", "Carter MY", "Carter MZ", "Carter NA", "Carter NB", "Carter NC", "Carter ND", "Carter NE", "Carter NF", "Carter NG", "Carter NH", "Carter NI", "Carter NJ", "Carter NK", "Carter NL", "Carter NM", "Carter NN", "Carter NO", "Carter NP", "Carter NQ", "Carter NR", "Carter NS", "Carter NT", "Carter NU", "Carter NV", "Carter NW", "Carter NX", "Carter NY", "Carter NZ", "Carter OA", "Carter OB", "Carter OC", "Carter OD", "Carter OE", "Carter OF", "Carter OG", "Carter OH", "Carter OI", "Carter OJ", "Carter OK", "Carter OL", "Carter OM", "Carter ON", "Carter OO", "Carter OP", "Carter OQ", "Carter OR", "Carter OS", "Carter OT", "Carter OU", "Carter OV", "Carter OW", "Carter OX", "Carter OY", "Carter OZ", "Carter PA", "Carter PB", "Carter PC", "Carter PD", "Carter PE", "Carter PF", "Carter PG", "Carter PH", "Carter PI", "Carter PJ", "Carter PK", "Carter PL", "Carter PM", "Carter PN", "Carter PO", "Carter PP", "Carter PQ", "Carter PR", "Carter PS", "Carter PT", "Carter PU", "Carter PV", "Carter PW", "Carter PX", "Carter PY", "Carter PZ", "Carter QA", "Carter QB", "Carter QC", "Carter QD", "Carter QE", "Carter QF", "Carter QG", "Carter QH", "Carter QI", "Carter QJ", "Carter QK", "Carter QL", "Carter QM", "Carter QN", "Carter QO", "Carter QP", "Carter QQ", "Carter QR", "Carter QS", "Carter QT", "Carter QU", "Carter QV", "Carter QW", "Carter QX", "Carter QY", "Carter QZ", "Carter RA", "Carter RB", "Carter RC", "Carter RD", "Carter RE", "Carter RF", "Carter RG", "Carter RH", "Carter RI", "Carter RJ", "Carter RK", "Carter RL", "Carter RM", "Carter RN", "Carter RO", "Carter RP", "Carter RQ", "Carter RR", "Carter RS", "Carter RT", "Carter RU", "Carter RV", "Carter RW", "Carter RX", "Carter RY", "Carter RZ", "Carter SA", "Carter SB", "Carter SC", "Carter SD", "Carter SE", "Carter SF", "Carter SG", "Carter SH", "Carter SI", "Carter SJ", "Carter SK", "Carter SL", "Carter SM", "Carter SN", "Carter SO", "Carter SP", "Carter SQ", "Carter SR", "Carter SS", "Carter ST", "Carter SU", "Carter SV", "Carter SW", "Carter SX", "Carter SY", "Carter SZ", "Carter TA", "Carter TB", "Carter TC", "Carter TD", "Carter TE", "Carter TF", "Carter TG", "Carter TH", "Carter TI", "Carter TJ", "Carter TK", "Carter TL", "Carter TM", "Carter TN", "Carter TO", "Carter TP", "Carter TQ", "Carter TR", "Carter TS", "Carter TT", "Carter TU", "Carter TV", "Carter TW", "Carter TX", "Carter TY", "Carter TZ", "Carter UA", "Carter UB", "Carter UC", "Carter UD", "Carter UE", "Carter UF", "Carter UG", "Carter UH", "Carter UI", "Carter UJ", "Carter UK", "Carter UL", "Carter UM", "Carter UN", "Carter UO", "Carter UP", "Carter UQ", "Carter UR", "Carter US", "Carter UT", "Carter UY", "Carter UZ", "Carter VA", "Carter VB", "Carter VC", "Carter VD", "Carter VE", "Carter VF", "Carter VG", "Carter VH", "Carter VI", "Carter VJ", "Carter VK", "Carter VL", "Carter VM", "Carter VN", "Carter VO", "Carter VP", "Carter VQ", "Carter VR", "Carter VS", "Carter VT", "Carter VY", "Carter VZ", "Carter WA", "Carter WB", "Carter WC", "Carter WD", "Carter WE", "Carter WF", "Carter WG", "Carter WH", "Carter WI", "Carter WJ", "Carter WK", "Carter WL", "Carter WM", "Carter WN", "Carter WO", "Carter WP", "Carter WQ", "Carter WR", "Carter WS", "Carter WT", "Carter WY", "Carter WZ", "Carter XA", "Carter XB", "Carter XC", "Carter XD", "Carter XE", "Carter XF", "Carter XG", "Carter XH", "Carter XI", "Carter XJ", "Carter XK", "Carter XL", "Carter XM", "Carter XN", "Carter XO", "Carter XP", "Carter XQ", "Carter XR", "Carter XS", "Carter XT", "Carter XU", "Carter XV", "Carter XW", "Carter XX", "Carter XY", "Carter XZ", "Carter YA", "Carter YB", "Carter YC", "Carter YD", "Carter YE", "Carter YF", "Carter YG", "Carter YH", "Carter YI", "Carter YJ", "Carter YK", "Carter YL", "Carter YM", "Carter YN", "Carter YO", "Carter YP", "Carter YQ", "Carter YR", "Carter YS", "Carter YT", "Carter YU", "Carter YV", "Carter YW", "Carter YX", "Carter YY", "Carter YZ", "Carter ZA", "Carter ZB", "Carter ZC", "Carter ZD", "Carter ZE", "Carter ZF", "Carter ZG", "Carter ZH", "Carter ZI", "Carter ZJ", "Carter ZK", "Carter ZL", "Carter ZM", "Carter ZN", "Carter ZO", "Carter ZP", "Carter ZQ", "Carter ZR", "Carter ZS", "Carter ZT", "Carter ZU", "Carter ZV", "Carter ZW", "Carter ZX", "Carter ZY", "Carter ZZ")
```

Now, we create another dataframe that lists all the vertex attributes for each slave in order to visualize them on the network graph.

```
slave_vertex <- data_frame(name = V(slave_graph)$name) %>%
  left_join(total_slave, by = c("name" = "id"))
head(slave_vertex)
```

Plotting the First Network Graph

```
igraph.options(vertex.size=3,
               edge.arrow.size=.5)
set.seed(33)
```

```
plot(slave_graph, vertex.label = NA)
```

Immediately, the overall kinship structure of the enslaved community is apparent from this network graph. Four family groups comprise one large familial relation and several smaller groups also exist. Many slaves had no kinship relations (no known kinship relations) and some slaves were only connected to one or two other people. In order to analyze this information further, we must be able to visualize the vertex attributes and the different kinship relationships.

Gender

In order to visualize the gender of the slaves, we can change the shape of each vertex accordingly.

```
V(slave_graph)$shape <- ifelse(slave_vertex$Gender == "Female", "circle",
                                ifelse(slave_vertex$Gender == "Male", "square",
                                         "rectangle"))

set.seed(33)
plot(slave_graph, vertex.label = NA)
```

Females are circles, while males are squares.

Relationships

In order to visualize the different kinships relations, we can colorize the edges according to type of relation with a function.

```
lookup_edge <- function(relationship) {
  if(relationship == "Spouse") return("red")
  if(relationship == "Child") return("orange")
  if(relationship == "Sibling") return("blue")
  return("gray")
}

lookup_edge <- Vectorize(lookup_edge, USE.NAMES = FALSE)

E(slave_graph)$color <- lookup_edge(E(slave_graph)$relationship)
set.seed(33)
plot(slave_graph,
     vertex.label = NA,
     vertex.size = 3)
```

The network graph is easier to decipher. We can see that the one large family is connected by a slave marriage. Many of the two person families are slave marriages, while others are mother and child. With this graph, we can count the number of each relationship and know where they exist within each family.

Evolution of Kinship Networks in Space

Since both slave censuses taken by Washington list the farms that each slave lived and worked on, we can use that variable to locate the slave families within the estate. We use a function similar to the edge coloring function:

```
lookup_color <- function(type) {
  if(is.na(type)) return("gray")
  if(type == "Mansion House") return("purple")
  if(type == "Muddy Hole") return("blue")
  if(type == "Dogue Run") return("green")
}
```

```

    if(type == "Union Farm") return("orange")
    if(type == "River Farm") return("yellow")
    if(type == "Mr. Lears") return("brown")
    if(type == "Mrs. Washington's") return("brown")
    if(type == "Cedar Grove") return("brown")
    return("gray")
  }

lookup_color <- Vectorize(lookup_color, USE.NAMES = FALSE)

```

Five Farms made up the Mount Vernon estate: Mansion House, Muddy Hole, Dogue Run, Union Farm, and River Farm. Most slaves married and had kinship relations within Washington's slave community. However, some slaves married people owned by others. I have colored those non-Mount Vernon slave spouses brown. The gray color represents slaves that were not active on each census, meaning that they were not born yet or had died.

Slave Families Across Farms in 1786

```

V(slave_graph)$color <- lookup_color(slave_vertex$Farm.x)
set.seed(33)

plot(slave_graph,
     vertex.label = NA,
     vertex.size = 3)
title("Slave Families on Mount Vernon Farms in 1786")
legend("bottomleft", legend = c("Mansion House", "Muddy Hole", "Dogue Run", "Union Farm", "River Farm", "Other"),
     col = c("purple", "blue", "green", "orange", "yellow", "brown", "gray"), pch = 19,
     title = "Farm")

```

This graph tells us a lot more information. We can place the slave families in space on the estate and organize the various family structures.

1786 Family Structures

From this graph, four types of family structures arise: singles, one parent, two parent, and couples with no known children. We can plot the numbers of each family structure for the entire estate, as well as the individual farms. We will use ggplot2 bar graphs to plot this information drawn from the network graph into a csv.

```

structures <- read.csv("~/Desktop/Clio-3/finalproject/FamilyStructures.csv", stringsAsFactors = FALSE)
head(structures)

```

We can use a simple bar plot to show the percentages of each family structure:

```

structures_1786 <- structures %>%
  filter(Census == "1786")

ggplot(data = structures_1786, aes(x = Family.structure, y = Percent)) + facet_wrap(~Farm) + geom_bar(stat = 'sum')

```

From this plot, we can see that almost every farm and the entire estate had a majority of single person structures.

Slave Families Across Farms in 1799

```

V(slave_graph)$color <- lookup_color(slave_vertex$Farm.y)
set.seed(33)

```

```
plot(slave_graph,
     vertex.label = NA,
     vertex.size = 3)
title("Slave Families on Mount Vernon Farms in 1799")
legend("bottomleft", legend = c("Mansion House", "Muddy Hole", "Dogue Run", "Union Farm", "River Farm", "Other"),
      col = c("purple", "blue", "green", "orange", "yellow", "brown", "gray"), pch = 19,
      title = "Farm")
```

1799 Family Structures

Let's plot the four family structures for 1799 next to the 1786 figures:

```
ggplot(data = structures, aes(x = Family.structure, y = Percent, fill = Census)) + facet_wrap(~Farm) + geom_bar()
```

Immediately, we see the increase in the percentage of two-parent family structures in just 13 years. Single person structures still dominate on Mansion House and Union Farm, but every farm and the overall estate increases in two-parent families.

Community Detection

We can use the community detection methods in R in order to focus on each family grouping in 1799.

```
comm <- walktrap.community(slave_graph)
length(comm)
```

The walktrap.community DESCRIBE and calculates 119 slave family groups on Mount Vernon in 1799. This is not completely accurate, as the largest family is counted as 4 groups. A more nuanced analysis of the number and definition of family groupings is needed for this data, but the exploring the individual groups is helpful.

We can also calculate the size of each of the family groups.

```
sizes(comm)
```

We can inspect each community in the graph and list the names of each family member.

```
memb <- membership(comm)
names(memb[memb == 2])
```

We can also show the kinship network graph colored by each family group.

```
set.seed(33)
plot(slave_graph, vertex.label = NA, vertex.color = memb, vertex.size = 3)
title("Families detected by walktrap.community()")
```

```
set.seed(33)
plot(comm, slave_graph, vertex.label = NA, vertex.size = 2)
title("Families detected by walktrap.community()")
```

Family Tree

In order to see how each slave is connected within a family, we can show a subgraph of the larger network graph in a different layout. We will use community 2 for this example.

```
induced.subgraph(slave_graph, which(membership(comm) ==2)) %>%
  plot(layout = layout.reingold.tilford, vertex.label = names(memb[memb == 2]))
title("Community 2 among Mount Vernon")
legend("bottomleft", legend = c("Spouse", "Child", "Sibling"),
      col = c("red", "orange", "blue"), pch = 19,
      title = "Relationship")
```

This graph is not the most visually appealing, nor does it show a true family tree in chronological and hierarchical order. Suckey Bay and NA9 are married and have three children: Nancy G, Nancy H, and Rose A. Rose A has four children of her own: Hagar A, Joe D, Simon A, and Tom C. This entire family does reside on River Farm in 1799, except Suckey Bay's spouse, who is unknown. This subgraph does show a two generational family living and working on the same farm in 1799. They are all connected by their familial ties, as well as their working community.

Let's explore a smaller family

```
induced.subgraph(slave_graph, which(membership(comm) == 32)) %>%
  plot(layout = layout.reingold.tilford, vertex.label = names(memb[memb == 32]))
title("Community 32 among Mount Vernon")
legend("bottomleft", legend = c("Spouse", "Child", "Sibling"),
      col = c("red", "orange", "blue"), pch = 19,
      title = "Relationship")
```

This graph shows that Daphne B and Simms A are spouses with three children: Maria A, Matilda B, and Polly A. The women live on Union Farm, but Simms A lives on Mansion House farm. This means that Daphne B and Simms A would have traveled across these two farms in order to see each other. Simms A belongs to two different communities: his family and the Mansion House working community.