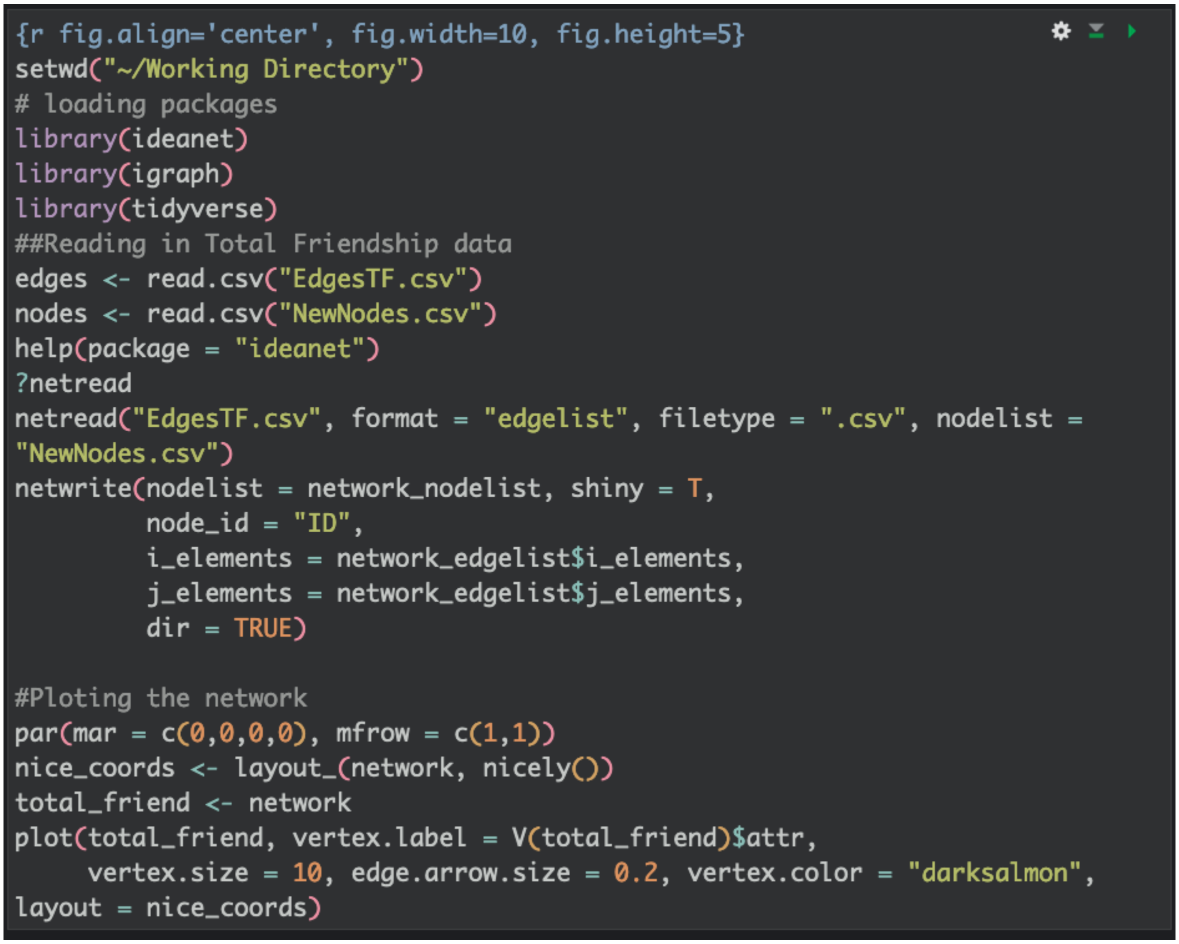
Analysis of Social Networks

Code Breakdown: Looking at the Networks in BoJack Horseman

Lauren Fowler

This project was completed with the use of RMarkdown.

Figure 1

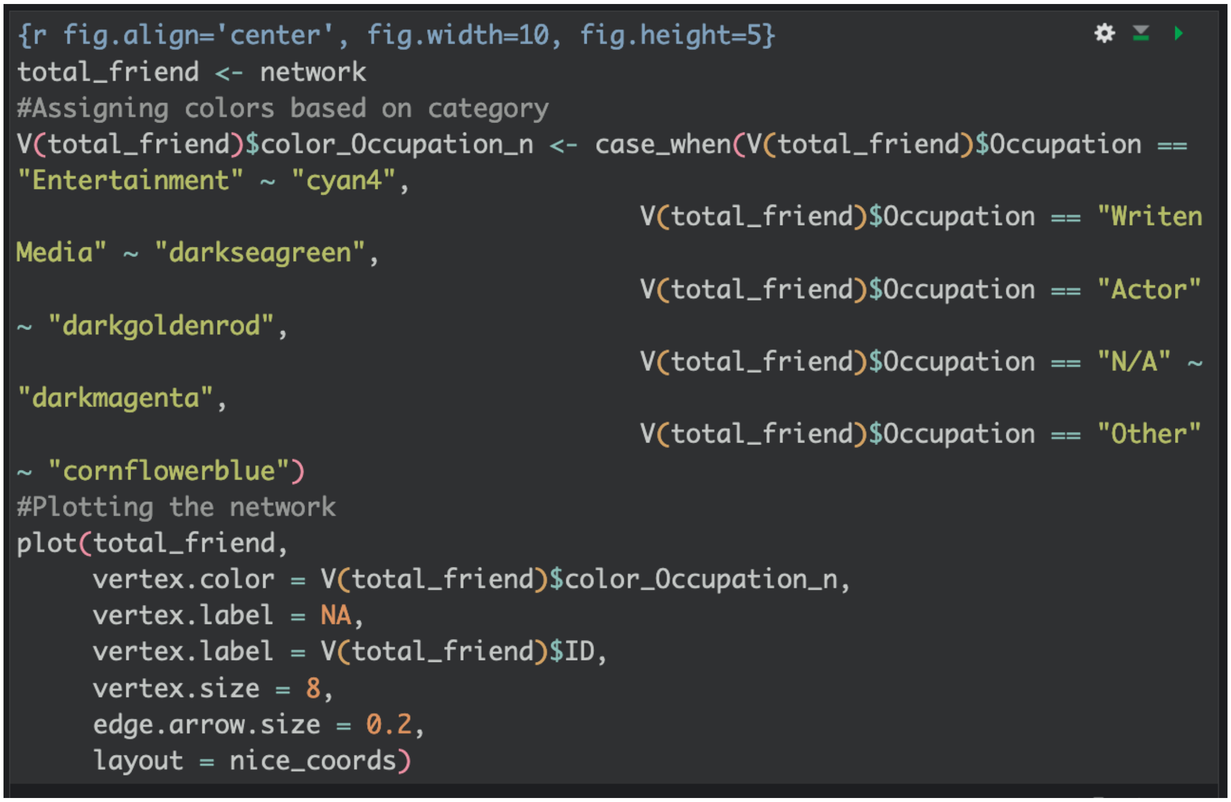


```
{r fig.align='center', fig.width=10, fig.height=5}
setwd("~/Working Directory")
# loading packages
library(ideanet)
library(igraph)
library(tidyverse)
##Reading in Total Friendship data
edges <- read.csv("EdgesTF.csv")
nodes <- read.csv("NewNodes.csv")
help(package = "ideanet")
?netread
netread("EdgesTF.csv", format = "edgelist", filetype = ".csv", nodelist =
"NewNodes.csv")
netwrite(nodelist = network_nodelist, shiny = T,
         node_id = "ID",
         i_elements = network_edgelist$i_elements,
         j_elements = network_edgelist$j_elements,
         dir = TRUE)

#Ploting the network
par(mar = c(0,0,0,0), mfrow = c(1,1))
nice_coords <- layout_(network, nicely())
total_friend <- network
plot(total_friend, vertex.label = V(total_friend)$attr,
     vertex.size = 10, edge.arrow.size = 0.2, vertex.color = "darksalmon",
layout = nice_coords)
```

Figure 1 is the first code block used for my project. First, I loaded in the necessary packages. Then I loaded in my data from excel using the ideanet package. I then plotted the network.

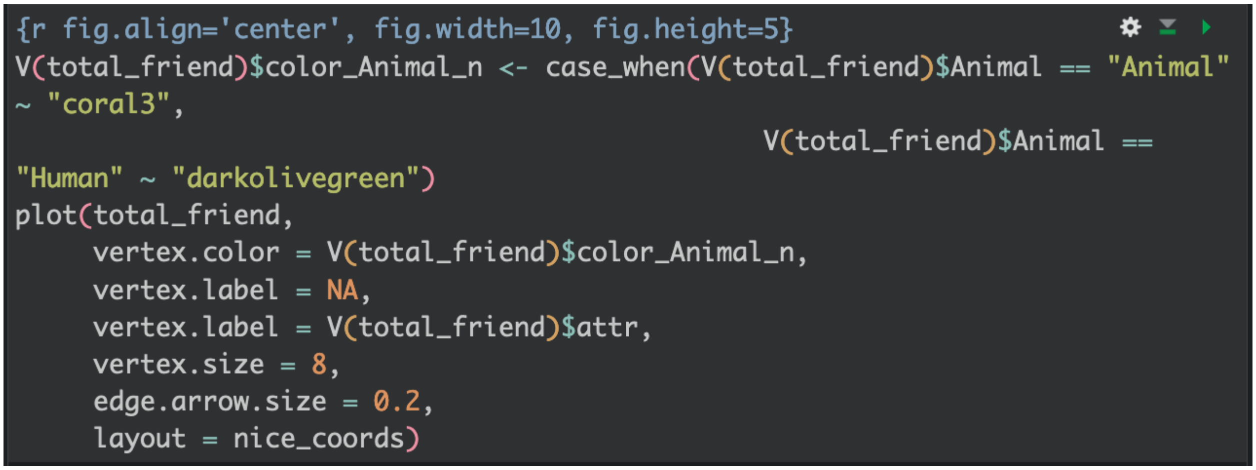
Figure 2



```
{r fig.align='center', fig.width=10, fig.height=5}
total_friend <- network
#Assigning colors based on category
V(total_friend)$color_Occupation_n <- case_when(V(total_friend)$Occupation ==
"Entertainment" ~ "cyan4",
                                        V(total_friend)$Occupation == "Writen
Media" ~ "darkseagreen",
                                        V(total_friend)$Occupation == "Actor"
~ "darkgoldenrod",
                                        V(total_friend)$Occupation == "N/A" ~
"darkmagenta",
                                        V(total_friend)$Occupation == "Other"
~ "cornflowerblue")
#Plotting the network
plot(total_friend,
     vertex.color = V(total_friend)$color_Occupation_n,
     vertex.label = NA,
     vertex.label = V(total_friend)$ID,
     vertex.size = 8,
     edge.arrow.size = 0.2,
     layout = nice_coords)
```

Here I grouped my network by color according to occupation and then plotted the network.

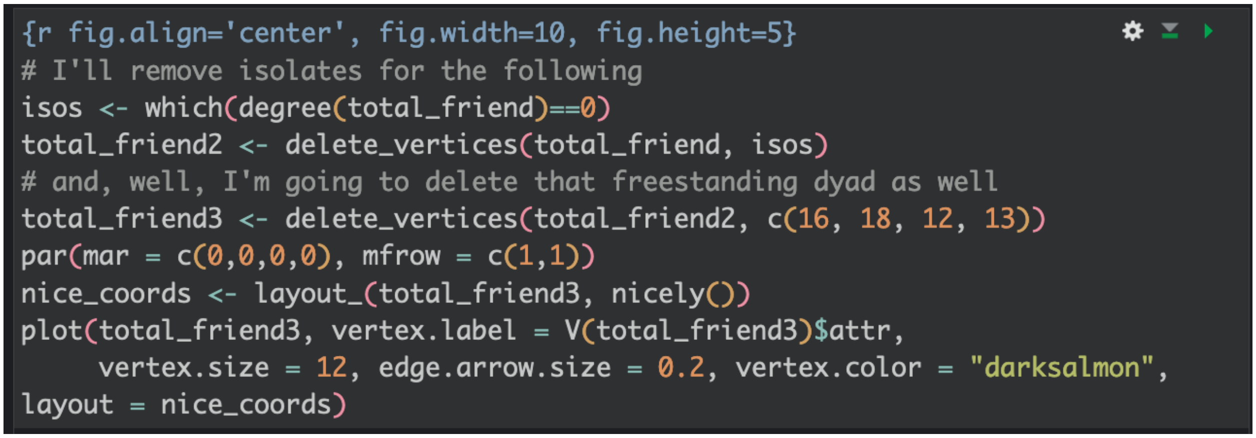
Figure 3



```
{r fig.align='center', fig.width=10, fig.height=5}
V(total_friend)$color_Animal_n <- case_when(V(total_friend)$Animal == "Animal"
~ "coral3",
                                              V(total_friend)$Animal ==
"Human" ~ "darkolivegreen")
plot(total_friend,
     vertex.color = V(total_friend)$color_Animal_n,
     vertex.label = NA,
     vertex.label = V(total_friend)$attr,
     vertex.size = 8,
     edge.arrow.size = 0.2,
     layout = nice_coords)
```

Again, the nodes were grouped by color according to species and plotted.

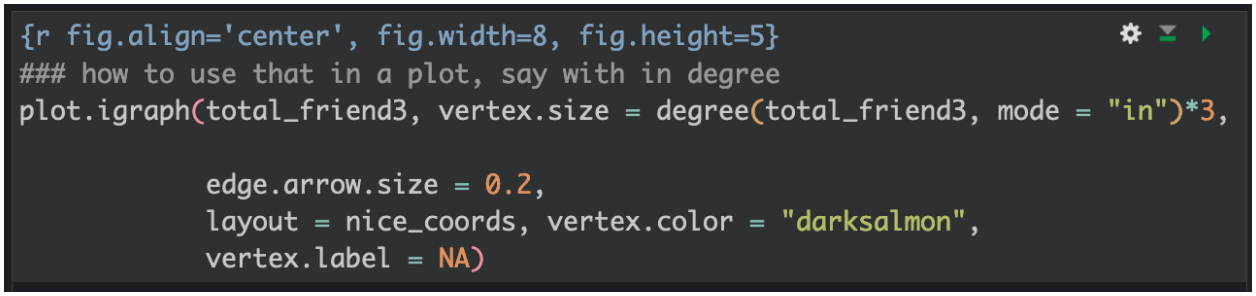
Figure 4



```
{r fig.align='center', fig.width=10, fig.height=5}
# I'll remove isolates for the following
isos <- which(degree(total_friend)==0)
total_friend2 <- delete_vertices(total_friend, isos)
# and, well, I'm going to delete that freestanding dyad as well
total_friend3 <- delete_vertices(total_friend2, c(16, 18, 12, 13))
par(mar = c(0,0,0,0), mfrow = c(1,1))
nice_coords <- layout_(total_friend3, nicely())
plot(total_friend3, vertex.label = V(total_friend3)$attr,
     vertex.size = 12, edge.arrow.size = 0.2, vertex.color = "darksalmon",
layout = nice_coords)
```

In Figure 4, I am removing the isolates and the free-standing dyad as so it will not affect later analysis. Then I am plotting the new network.

Figure 5

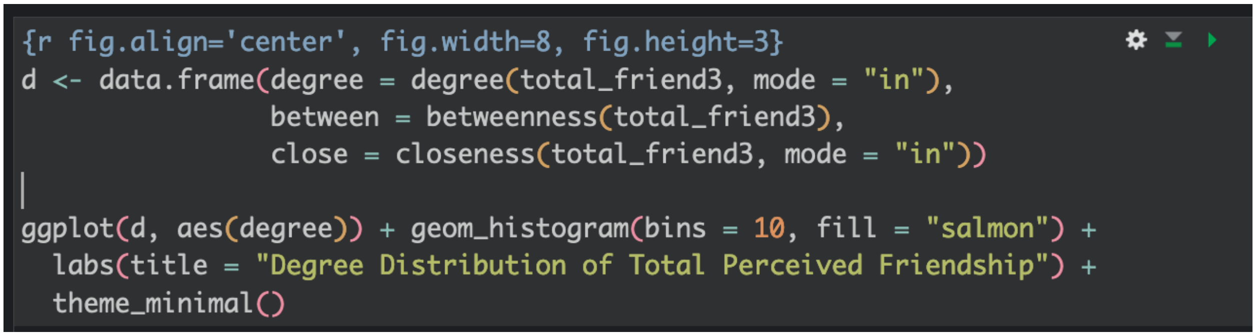


```
{r fig.align='center', fig.width=8, fig.height=5}
### how to use that in a plot, say with in degree
plot.igraph(total_friend3, vertex.size = degree(total_friend3, mode = "in")*3,

            edge.arrow.size = 0.2,
            layout = nice_coords, vertex.color = "darksalmon",
            vertex.label = NA)
```

Here, I am using the igraph package to plot degree centrality in my network. This will give me an image of my network with the size of the nodes based on their degree centrality.

Figure 6

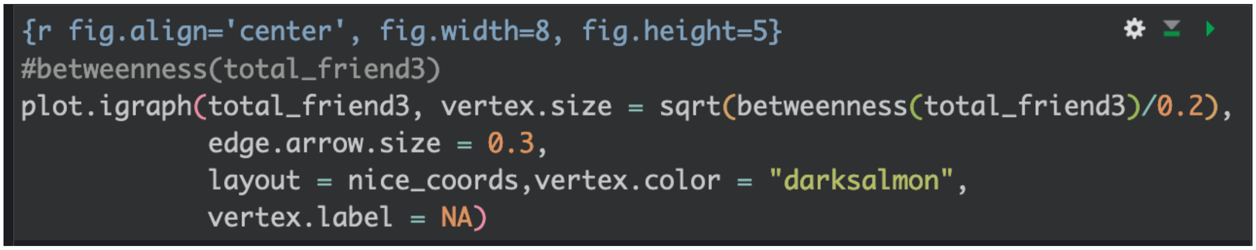


```
{r fig.align='center', fig.width=8, fig.height=3}
d <- data.frame(degree = degree(total_friend3, mode = "in"),
                between = betweenness(total_friend3),
                close = closeness(total_friend3, mode = "in"))

ggplot(d, aes(degree)) + geom_histogram(bins = 10, fill = "salmon") +
  labs(title = "Degree Distribution of Total Perceived Friendship") +
  theme_minimal()
```

With this I am creating a data frame with all modes of centrality. I am then making a histogram of just degree centrality as another way to show the centrality of the network.

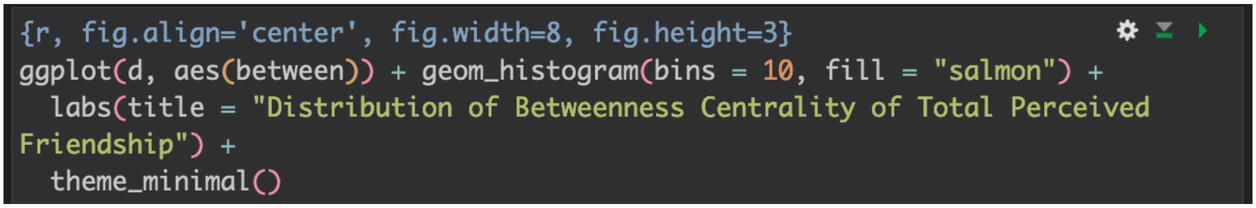
Figure 7



```
{r fig.align='center', fig.width=8, fig.height=5}
#betweenness(total_friend3)
plot.igraph(total_friend3, vertex.size = sqrt(betweenness(total_friend3)/0.2),
            edge.arrow.size = 0.3,
            layout = nice_coords,vertex.color = "darksalmon",
            vertex.label = NA)
```

Here, I am plotting betweenness centrality for the network.

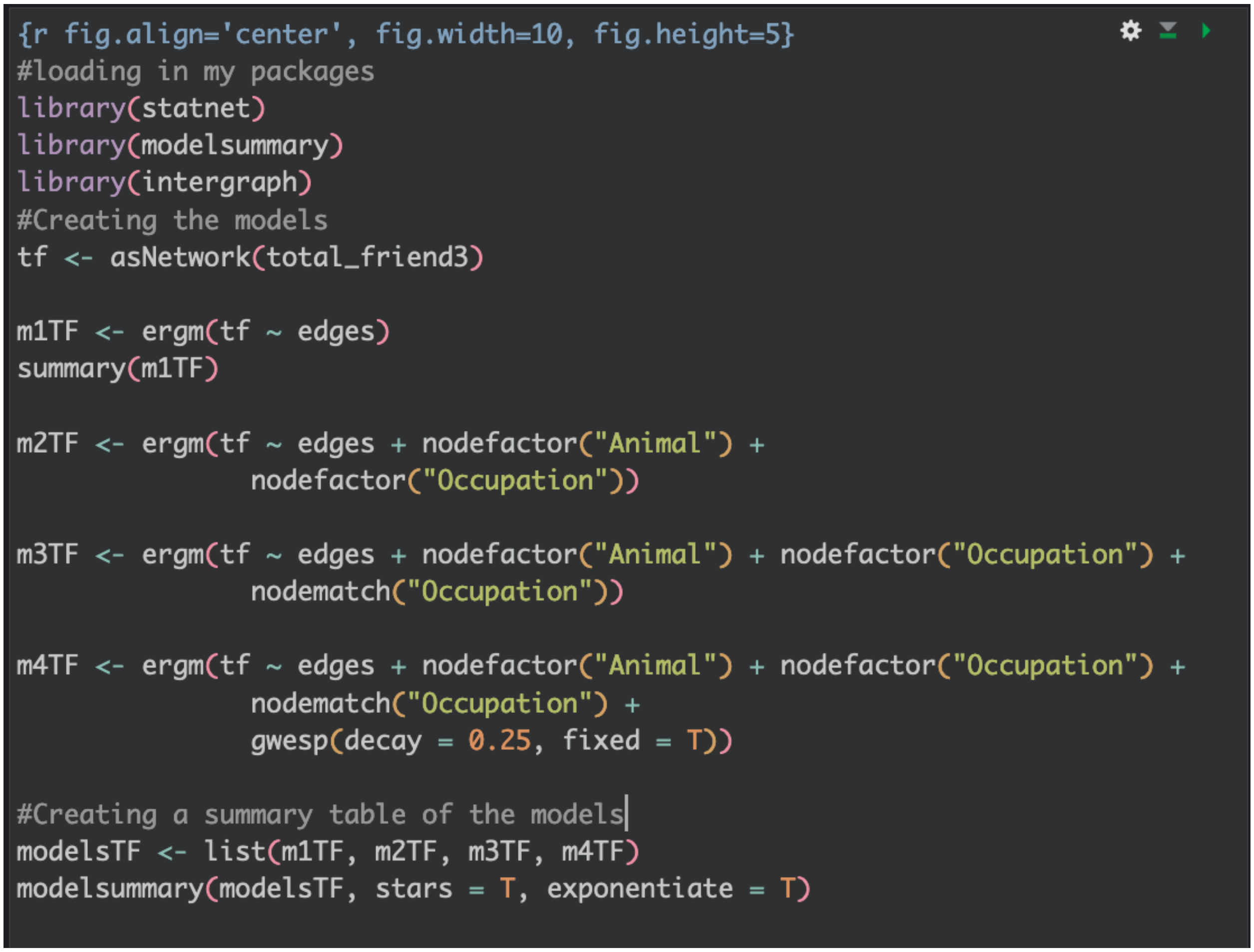
Figure 8



```
{r, fig.align='center', fig.width=8, fig.height=3}
ggplot(d, aes(between)) + geom_histogram(bins = 10, fill = "salmon") +
  labs(title = "Distribution of Betweenness Centrality of Total Perceived
Friendship") +
  theme_minimal()
```

Again, I create a histogram, but this time looking at betweenness centrality.

Figure 9



```
{r fig.align='center', fig.width=10, fig.height=5}
#loading in my packages
library(statnet)
library(modelsummary)
library(intergraph)
#Creating the models
tf <- asNetwork(total_friend3)

m1TF <- ergm(tf ~ edges)
summary(m1TF)

m2TF <- ergm(tf ~ edges + nodefactor("Animal") +
               nodefactor("Occupation"))

m3TF <- ergm(tf ~ edges + nodefactor("Animal") + nodefactor("Occupation") +
               nodematch("Occupation"))

m4TF <- ergm(tf ~ edges + nodefactor("Animal") + nodefactor("Occupation") +
               nodematch("Occupation") +
               gwesp(decay = 0.25, fixed = T))

#Creating a summary table of the models
modelsTF <- list(m1TF, m2TF, m3TF, m4TF)
modelsummary(modelsTF, stars = T, exponentiate = T)
```

This block of code is how I was able to get my ERGMS. First, I loaded in any packages I any need. Then I created 4 different models and put them in a table to summarize.

For my other network, economic exchange, I use the exact same code as in Figure 1 – 9, except I use a different network. Since the code is generally the same, there are no images of the code blocks.