Assignment Number 2

Question 1

```
# Question 1 ---------------------
closeness_centralities <- closeness(affiliation_graph)
max_closeness_node <- which.max(closeness_centralities)
closeness(affiliation_graph, vids = max_closeness_node)

shortestPaths <- data.table(shortest.paths(affiliation_graph))
#Replace Inf shortest path by number of vertices
shortestPaths[shortestPaths==Inf] <- gorder(affiliation_graph)
average_mean_short_paths <- shortestPaths[ , lapply(.SD, mean)]
which.min(average_mean_short_paths)
```

a. Which firm is the center of the venture capital firm network as of July 2014?

```
Intel Capital
  1.61524e-07
```

b. Firm with lowest average shortest path

```
> which.min(average_mean_short_paths)
Intel Capital
```

c. Mean shortest distance of all vertices

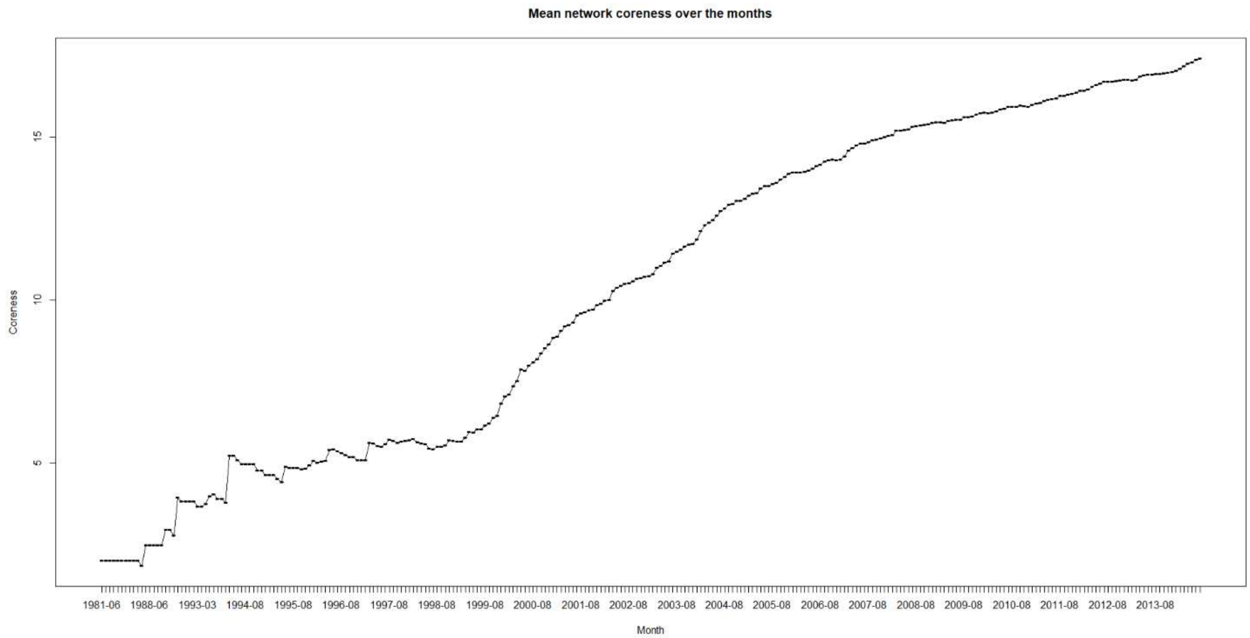
```
> average_mean_short_paths[ , sum_of_avg_paths := sum(.SD)]
> sum_of_avg_paths <- average_mean_short_paths[, sum_of_avg_paths]
> average_mean_short_paths[ , sum_of_avg_paths := NULL]
> number_of_nodes <- length(colnames(average_mean_short_paths))
> mean_shortest_distance <- sum_of_avg_paths/number_of_nodes
> mean_shortest_distance
[1] 969.6517
```

The average above is high because a large number of nodes have 'Inf' as shortest distances between nodes, as shown below.

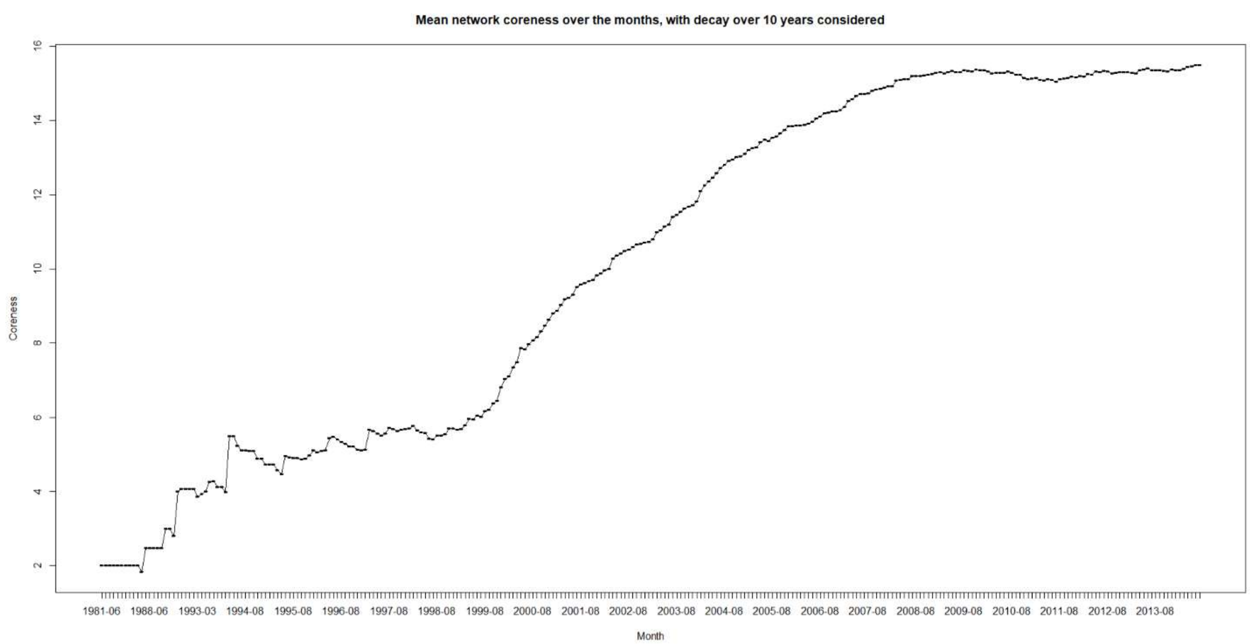
```
> shortestPaths <- data.table(shortest.paths(affiliation_graph))
> #Replace Inf shortest path by number of vertices
> sum(shortestPaths == Inf)
[1] 12086692
```

Question 2

a. Cumulative over months.



b. Decay considered



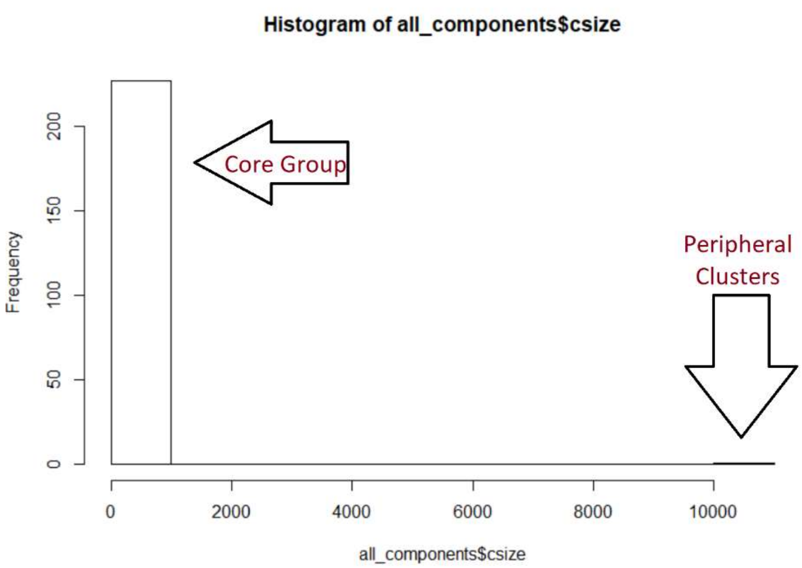
Question 3

Below we can the number of clusters and number of vertices in each cluster. As we can notice there is huge core clustered group with 10647 vertices(companies) in it.

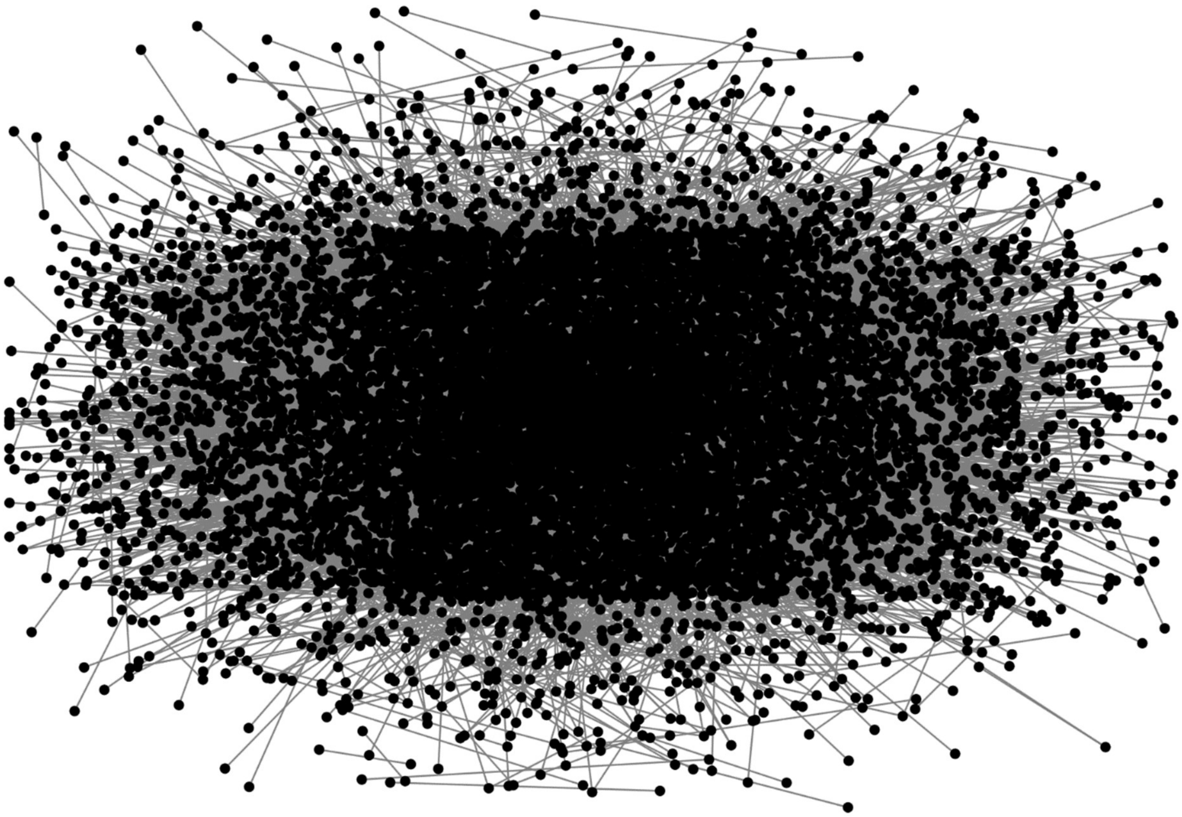
```
all_components <- components(cummulative.month.affiliation_graph)
gorder(cummulative.month.affiliation_graph)
```

```
$csize
  [1] 10647     2     2     4     2     2     2     2     2     3     2     3     3     2     2     2     2     5     2     2     2
 [22]     2     2     2     2     2     3     2     2     2     3     2     2     2     2     2     2     2     2     2     2     2
 [43]     2     2     3     3     2     2     3     4     2     2     2     2     5     2     3     2     2     2     2     2     3
 [64]     2     2     2     2     2     2     2     4     2     3     2     2     2     2     2     3     2     2     3     2     2
 [85]     3     2     3     2     3     2     2     2     2     2     4     2     4     2     2     2     2     3     2     2     2
[106]     2     2     2     2     2     4     2     3     2     2     2     3     2     2     4     3     2     2     2     2     2
[127]     2     2     2     2     3     4     3     2     2     2     2     2     2     3     2     2     2     2     3     3     2
[148]     2     2     2     2     2     2     2     2     2     2     2     2     2     2     2     2     2     2     2     2     2
[169]     3     2     2     2     2     2     2     2     2     2     2     3     2     2     2     2     2     2     2     2     2
[190]     2     3     2     2     2     2     2     2     2     2     2     2     2     2     2     2     2     2     2     3     3
[211]     2     2     2     2     2     2     2     2     2     2     2     3     3     2     2     2     2     2

$no
[1] 228

> gorder(cummulative.month.affiliation_graph)
[1] 11155
```

Number of Clusters



Graph in NodeXL

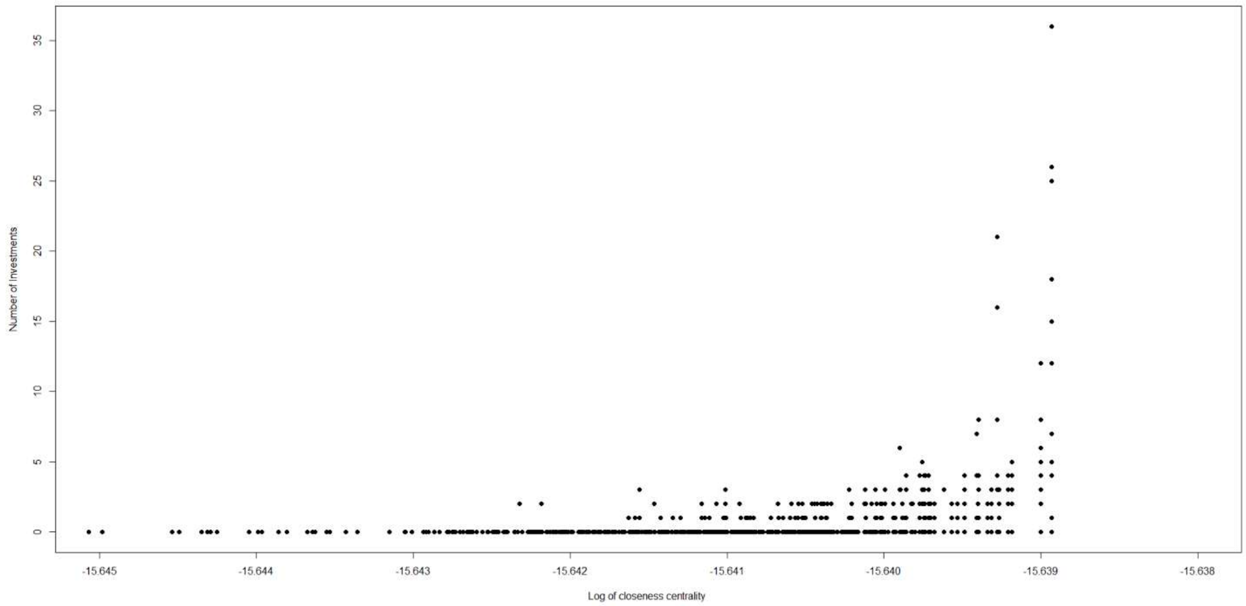


Question 4

a. As the below graph shows the ore central the company is in the network the more number of investments it does. I have taken log of closeness to make the results and pattern shown clearly.

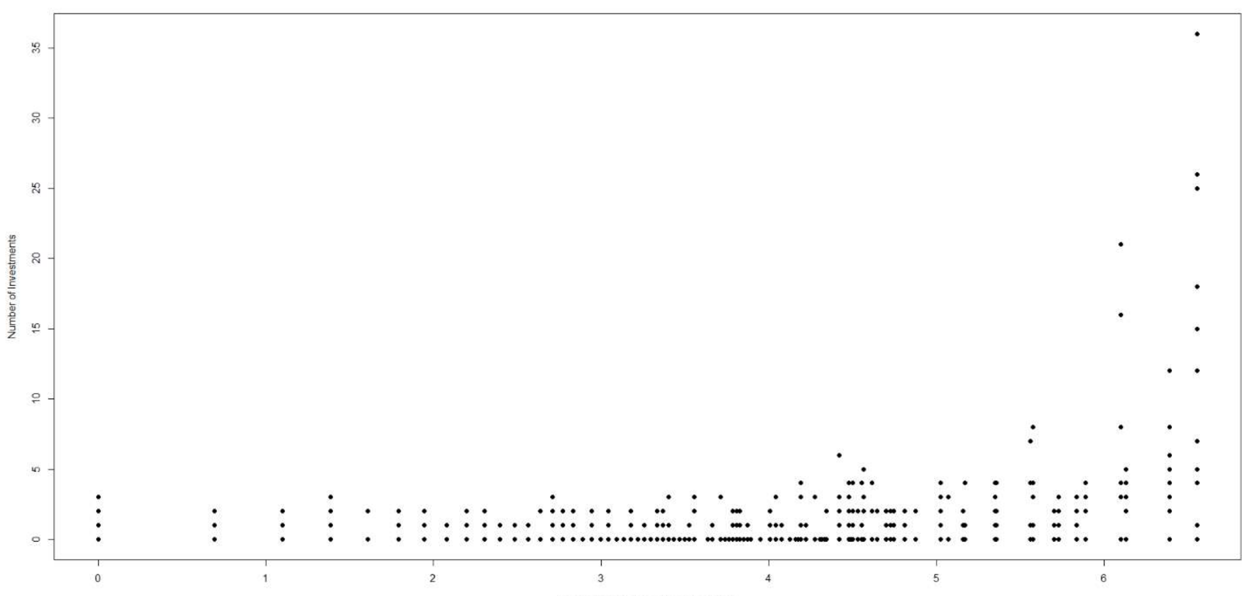
Closeness vs number of investments.

```
cc <- vc_outcomes[!is.na(Closensess_Centrality), .(Closensess_Centrality, Number_of_Investments)]
plot(log(cc[, Closensess_Centrality]), xlim = c(-15.645, -15.638), cc[, Number_of_Investments],
     xlab="Log of closeness centrality ", ylab="Number of Investments", pch=19)
```



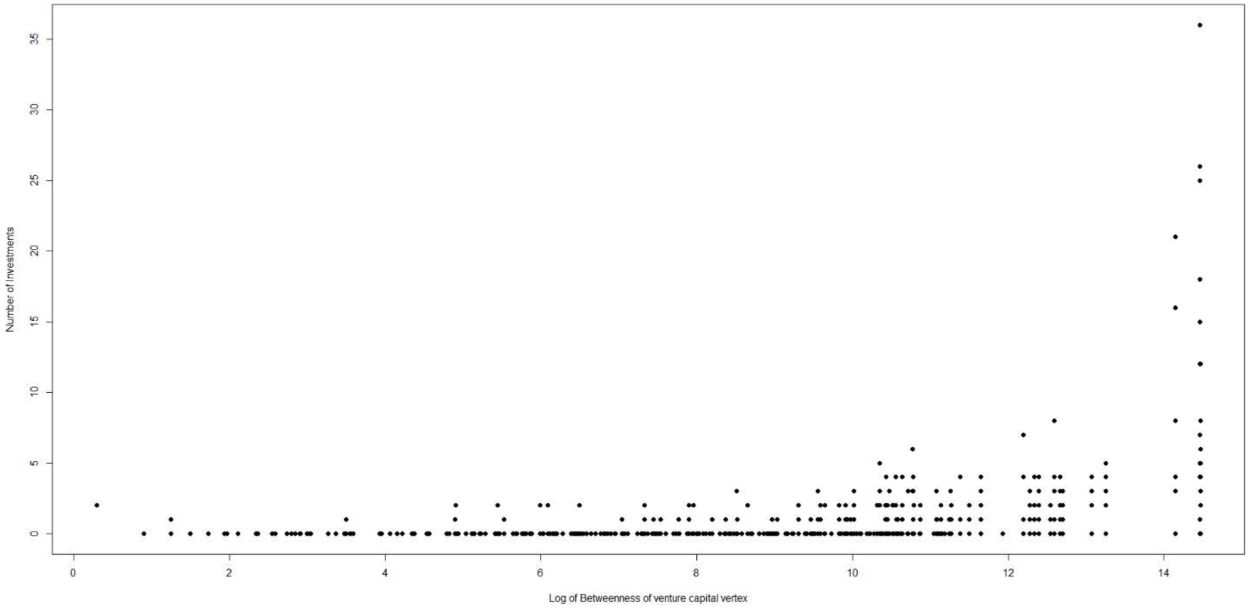
Degree vs Number of investments.

```
degrees <- vc_outcomes[!is.na(Degree_Value), .(Degree_Value, Number_of_Investments)]
plot(log(degrees[, Degree_Value]), degrees[, Number_of_Investments],
     xlab="Log of Degree of venture capital vertex", ylab="Number of Investments", pch=19)
```



Betweenness vs Number of investments made.

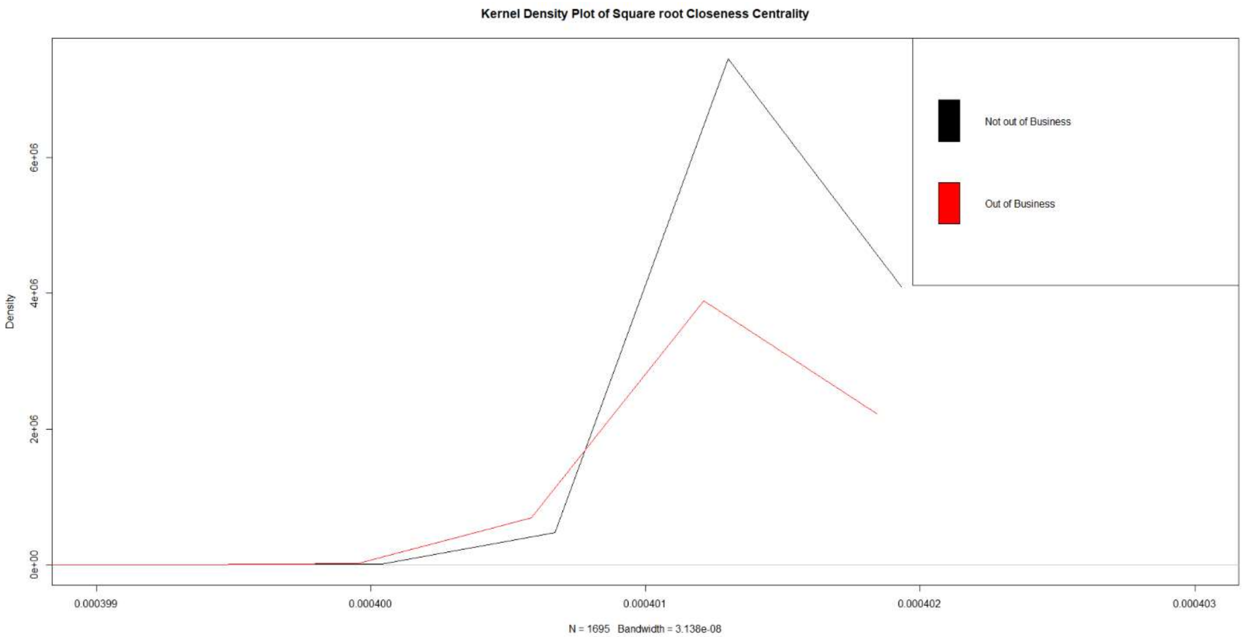
```
betweennesses <- vc_outcomes[!is.na(Betweenness_values), .(Betweenness_values, Number_of_Investments)]
plot(log(betweennesses[, Betweenness_values]), betweennesses[, Number_of_Investments],
     xlab="Log of Betweenness of venture capital vertex", ylab="Number of Investments", pch=19)
```



b. Success of the firm vs centrality measure.
   Below is the kernel density graphs for the centrality measures vs the success of firm(Not out of business)

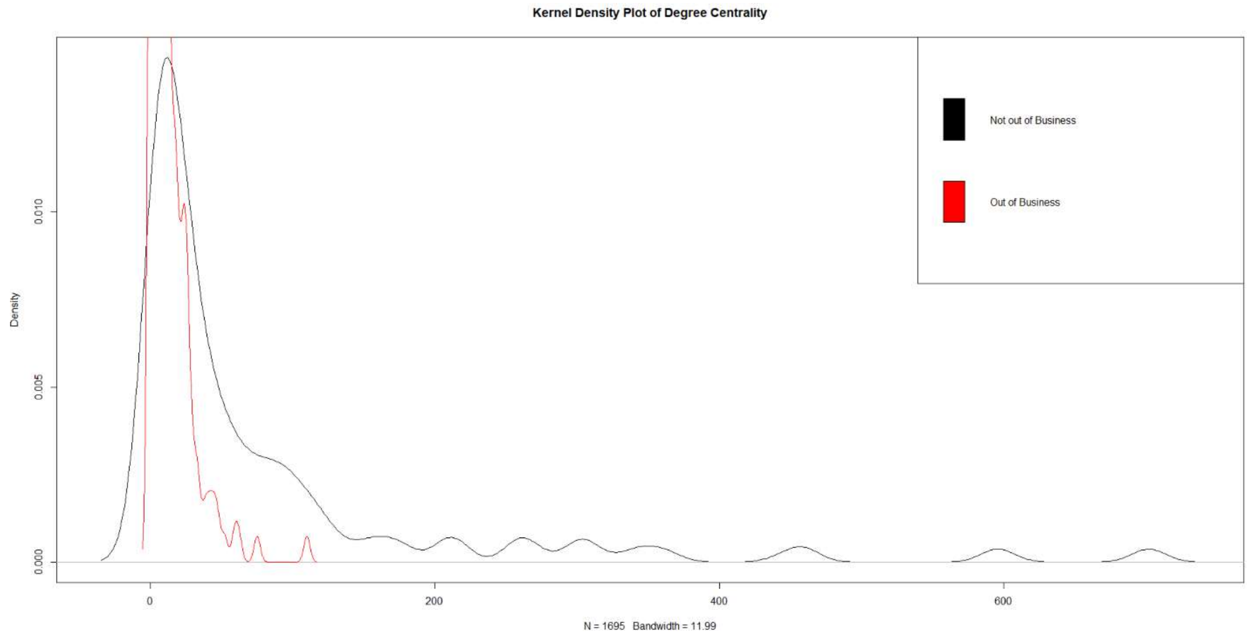
Closeness vs success of the firm

```
cc_success <- vc_outcomes[!is.na(Closensess_Centrality) & Out_of_Business == 0, Closensess_Centrality] #
cc_out_of_b <- vc_outcomes[!is.na(Closensess_Centrality) & Out_of_Business == 1, Closensess_Centrality]
plot(density(sqrt(cc_success)), xlim = c(0.000399, 0.000403),
     main = "Kernel Density Plot of Square root Closeness Centrality") # plots the results
lines(density(sqrt(cc_out_of_b)), col = "red")
legend("topright", c("Not out of Business","Out of Business"), fill=c("black", "red"))
```



Degree vs success of the firm.

```
cc_success <- vc_outcomes[!is.na(Degree_Value) & Out_of_Business == 0, Degree_Value] #
cc_out_of_b <- vc_outcomes[!is.na(Degree_Value) & Out_of_Business == 1, Degree_Value]
plot(density(cc_success), main = "Kernel Density Plot of Degree Centrality") # plots th
lines(density(cc_out_of_b), col = "red")
legend("topright", c("Not out of Business","Out of Business"), fill=c("black", "red"))
```



Betweenness vs Success of the firm.

```
cc_success <- vc_outcomes[!is.na(Betweenness_values) & Out_of_Business == 0, Betweenness_values] #
cc_out_of_b <- vc_outcomes[!is.na(Betweenness_values) & Out_of_Business == 1, Betweenness_values]
plot(density(log(cc_success)), main = "Kernel Density Plot of Log of Betweenness Centrality") # plc
lines(density(log(cc_out_of_b)), col = "red")
legend("topright", c("Not out of Business","Out of Business"), fill=c("black", "red"))
```