Data Visualization – Project 2 Write-up

1. What is the data that you chose? Why?

The data chosen were released as part of ACM's Knowledge Discovery & Data Mining Competition (KDD Cup 2003). The Stanford Linear Accelerator Center SPIRES-HEP database has been comprehensively cataloguing the **High Energy Particle Physics (HEP) literature online since 1974**, and the dataset I used for this project comes from the e-print arXiv (a superset of the SPIRES-HEP database), initiated in Aug 1991, and is the primary mode of research communication in multiple fields of physics, and some related disciplines. It currently contains over 225,000 full text articles.

The Cit-HepPh data I graphed comes from the arXiv and covers all the citations within a dataset of 34,546 papers with 421,578 citations. **If a paper i cites paper j, the graph contains a directed edge from i to j- If a paper cites, or is cited by, a paper outside the dataset, the graph does not contain any information about this.** The data covers papers in the period from January 1993 to April2003 (124 months).

Each paper is represented by a vertex (34,546 nodes total), and each citation is represented by an edge (421,578 edges total).

My rationale for choosing this dataset was two-fold: (1) I thought the connections between papers was interesting – would there be some overall structure to the graph? and (2) It was a challenge for me to construct a graph with this many vertices/edges. I wanted to understand how to do something like this.

2. Did you use a subset of the data? If so, what was it?

Each row of the dataset (421,578 records) consisted of one citation between papers. The details of the papers were not included in the dataset, and I could not discover a way to reasonably subset the data – so the entire dataset (34,546 vertices, 421,578 edges) was used for the graph.

3. Are there any particular aspects of your visualization to which you would like to bring attention?

The graph was created using the R software **igraph package**. Due to the large dataset it took a while for the graph to be created, but the code to create the graph is quite concise, here it is:

```
library(igraph)

citData <- read.table(file="Cit-HepPh1.txt", sep="", header=TRUE, quote="")

# Length of data frame should be # of edges in the graph
g <- graph.data.frame(citData, directed=FALSE)

# Look at vertices (esp. to count number to make sure vertex count is correct)
verts <- V(g)

# Extremely large graph - needs more room - reroute to large PDF
pdf( "CitGraph.pdf", width = 30, height = 9 )

# Use the "layout_nicely" function to organize the graph (see Section 4 of write for more info)
layorg <- layout_nicely(g, dim=2)
```

```
# Too many edges, change to white to blank out
E(g)$color <- "white"

# Plot the graph with 34,546 vertices
plot(g, layout=layorg, vertex.label=NA, vertex.size=1,     main="Citation Network - Arxiv High Energy Physics Papers
(34,546 papers, 421,578 citations)")

dev.off()
```

[NOTE: This generates a very large PDF (~34 MB) so a PNG snapshot of the PDF was taken to include in this project submission.]

The only specific aspect of the visualization to point out is that all edges between vertices have been “whited out”. Because there are over 400,000 edges, when printed the graph just became a large grey “blob”. That didn’t reveal anything – so for that reason all nodes are shown, but **all edges are not visible. See the following section (Section 4) for why this is not a problem.**

4. What do you think the data, and your visualization, shows?

The “layout_nicely” function to build the graph structure is a critical part of this code (due to the number of vertices). With this many vertices, printing a graph of this size to show anything meaningful becomes a challenge.

Recently (2008), S. Martin and colleagues at Sandia Labs published/implemented an algorithm for layout of very large graphs (reference 3 below). Their approach uses “simulated annealing” (see reference 4) as a method of placing nodes that are related (i.e. large number of interconnections) closer to each other.

So, instead of needing the edges to show the connection relationships, this “distributed recursive layout” approach shows the nodes/articles that have many citations/edges related to them closer to each other.

The outer part of the graph clearly shows a number (roughly 100-150) articles (represented by dots/vertices) that are rarely or never cited in any other papers. Towards the center of the graph, we see very tight clusters, meaning there are groups of papers that are closely related (and cited) together.

5. References/Citations

1. J. Leskovec, J. Kleinberg and C. Faloutsos. Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations. ACM SIG KDD International Conference on Knowledge Discovery and Data Mining (KDD), 2005.
2. J. Gehrke, P. Ginsparg, J. M. Kleinberg. Overview of the 2003 KDD Cup. SIGKDD Explorations 5(2): 149-151, 2003.
3. Martin, S., Brown, W.M., Klavans, R., Boyack, K.W., DrL:Distributed Recursive (Graph) Layout. SAND Reports (Sandia Lab Technical Report), 2008. 2936: p. 1-10.
4. Russell, S. & Norvig, P. – “Artificial Intelligence – A Modern Approach” 2nd Edition. P. 116.