Complex Networks

Programming Assignment 1

Jason Thomas

s3907634

The code for producing these results is below the responses to assessment questions, as an appendix.

In [26]:

```
### Small bit of code to produce results that follow

packages = c('testit', 'network', 'sna', 'scales')
for(package in packages) {
    suppressMessages(
        library(package, character.only = TRUE))
}
source("assignment1_functions.R"); # lecturer provided
source("my_functions.R");
source("tests.R");
source("process_adj_matrices.R"); # load up the networks
```

```
[1] "Tests pass"
```

.

Question 1

.

In [27]:
```
plotAllNetworks(networks);
system("./makeGrid.sh"); # This was hard to do in R, bash was easy
```

chartGrid

\

Question 2

In the table, let "lc" be longest component

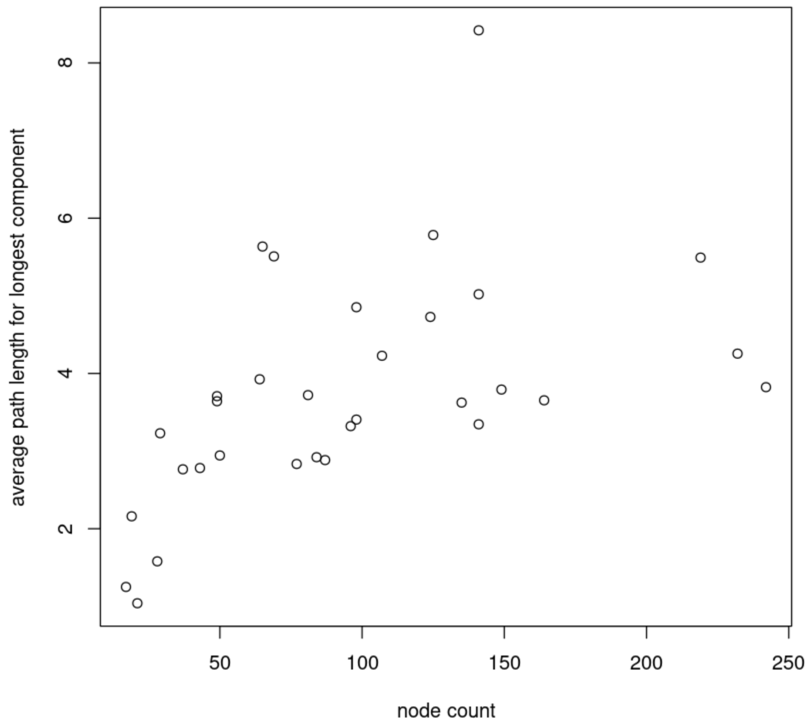
In [28]:
```
data = getDataForAllNetworks(networks);
data
```

A data.frame: 32 × 7

	net_id	node_count	edge_count	component_count	lc_node_share	average_degree	averag
	<int>	<dbl>	<int>	<int>	<dbl>	<dbl>	
32	1	29	33	7	0.6551724	2.2758621	
31	2	125	251	8	0.5920000	4.0160000	
30	3	164	657	2	0.9878049	8.0121951	
29	4	242	1083	1	1.0000000	8.9504132	
28	5	219	630	4	0.9634703	5.7534247	
27	6	232	612	11	0.4568966	5.2758621	
26	7	141	293	8	0.8865248	4.1560284	
25	8	49	122	1	1.0000000	4.9795918	
24	9	28	37	6	0.3571429	2.6428571	
23	10	64	120	7	0.8281250	3.7500000	
22	11	37	31	15	0.4864865	1.6756757	
21	12	50	102	4	0.8000000	4.0800000	
20	13	21	9	15	0.2380952	0.8571429	
19	14	17	3	14	0.2352941	0.3529412	
18	15	19	16	7	0.5263158	1.6842105	
17	16	96	275	4	0.9583333	5.7291667	
16	17	141	493	3	0.9858156	6.9929078	
15	18	107	292	6	0.9345794	5.4579439	
14	19	81	131	10	0.4938272	3.2345679	
13	20	49	71	6	0.6938776	2.8979592	
12	21	77	308	1	1.0000000	8.0000000	
11	22	98	264	4	0.9693878	5.3877551	
10	23	84	264	3	0.9761905	6.2857143	
9	24	135	466	5	0.9333333	6.9037037	
8	25	98	215	3	0.9591837	4.3877551	
7	26	69	118	2	0.9855072	3.4202899	
6	27	65	130	3	0.9692308	4.0000000	
5	28	43	118	1	1.0000000	5.4883721	
4	29	87	302	3	0.9770115	6.9425287	
3	30	149	525	2	0.9932886	7.0469799	
2	31	124	241	11	0.8548387	3.8870968	
1	32	141	351	6	0.9007092	4.9787234	

Question 3

In [7]:
```
plot(data$node_count,
     data$average_path_length_longest_component,
     xlab="node count",
     ylab="average path length for longest component");
```



We might also like to see the Pearson Correlation Coefficient and the level of confidence that we can have, given a default confidence level of 0.95.

In [8]:
```
result <- cor.test(data$node_count, data$average_path_length_longest_compone
print("Coefficient:")
result$estimate
print("p-value:")
result$p.value
```

```
[1] "Coefficient:"
```

cor: 0.510415252889794

```
[1] "p-value:"
```

0.00283838153769246

Given the confidence level of 0.95, then we apply $\alpha = 0.05$. Since the p-value is lower than the α then this result is statistically significant.

Note that this interpretation relies on their being a good sample without sampling errors.

Regarding the networks in this analysis: each edge represents a direct contact between one vole and another. If two voles are connected by a path that has more than one edge then those two voles have no direct contact.

The strong correlation between network size and average path length suggests that these animals are either territorial or limit their contact with others of their species for some other survival reason.

Appendix

My code

In [13]:
```
system("cat my_functions.R")
```

In [ ]:
```

```