

# Visualizing a co-occurrence network in Cytoscape

*Julia Gustavsen*

## Contents

<b>1</b>	<b>Description of vignette</b>	<b>2</b>
1.1	Tara Oceans . . . . .	2
1.2	How to examine organisms that occur together: co-occurrence networks . . . . .	2
1.3	What can we find out by creating co-occurrence networks? . . . . .	2
1.4	What kind of data are used in this vignette? . . . . .	2
<b>2</b>	<b>Set up Cytoscape and R connection</b>	<b>3</b>
2.1	Requirements . . . . .	3
<b>3</b>	<b>Read in data</b>	<b>3</b>
<b>4</b>	<b>Read in taxonomic classification</b>	<b>3</b>
<b>5</b>	<b>Add the taxonomic classifications to the network and then send network to Cytoscape</b>	<b>4</b>
<b>6</b>	<b>Send network to Cytoscape using RCy3</b>	<b>5</b>
<b>7</b>	<b>Colour network by prokaryotic phylum</b>	<b>7</b>
7.1	Set node shape to reflect virus or prokaryote . . . . .	9
<b>8</b>	<b>Colour edges of phage nodes</b>	<b>11</b>
<b>9</b>	<b>Do layout to minimize overlap of nodes.</b>	<b>13</b>
<b>10</b>	<b>Look at network properties</b>	<b>16</b>
<b>11</b>	<b>Size by degree</b>	<b>16</b>
<b>12</b>	<b>Select an interesting node and make a subnetwork from it</b>	<b>18</b>
12.1	Conclusion . . . . .	20
	<b>References</b>	<b>20</b>

# 1 Description of vignette

## 1.1 Tara Oceans

Many projects have collected samples from different regions and from different depths of the ocean. Some, such as the pioneering study by Craig Venter (Venter et al. 2004) have pioneered metagenomic sequencing and others, such as Tara Oceans Expedition have collected large amounts of data with global ecological questions in mind. On the Tara Oceans (8th and 9th expedition for this vessel researchers used a small sailboat outfitted with a lab and filtration supplies to collect samples from many different size fractions of microorganisms in the oceans over three years. They collected these samples to look at the different kinds of microorganisms present in different parts of the oceans to look at their composition and to observe their spatial patterns and distribution.

The scientists collected the samples and then used either targeted sequencing (amplicon approach using primers for specific targets such as the ribosomal genes and then amplifying these targets using PCR) or using metagenomic sequencing (where all the genetic material in a sample is sequenced) of each of the size fractions.

After the sequencing and quality checking of the samples was done, the sequences were taxonomically classified (different approaches for the different targets, see here for the details in Brum et al. (2015) and Sunagawa et al. (2015)). After that the data could be made into a species occurrence table where the rows are different sites and the columns are the observations of the different organisms at each site (Lima-Mendez et al. 2015).

## 1.2 How to examine organisms that occur together: co-occurrence networks

Many of these microbial species in these types of studies have not yet been characterized in the lab. Thus, to know more about the organisms and their interactions, we can observe which ones occur at the same sites or under the same kinds of environmental conditions. One way to do that is by using co-occurrence networks where you examine which organisms occur together at which sites. The more frequently that organisms co-occur at the same site, the stronger the interaction predicted among these organisms. For a review of some of the different kinds of techniques and software for creating interaction networks see: Weiss et al. (2016).

## 1.3 What can we find out by creating co-occurrence networks?

These kinds of analyses can be useful for studies where the organisms have not yet been characterized in the lab because these analyses can provide insights about the communities and how the organisms within them are interacting. These analyses can be exploratory, so that we can see which organisms warrant further insights and perhaps experimental work. We can also learn about how the overall community is organized (community structure) by looking at some of the network properties (that is the overall way that the organisms are co-occurring and the properties of the network seen this way).

## 1.4 What kind of data are used in this vignette?

In this analysis we are using a Tara Ocean data and we have data from the bacterial dataset (Sunagawa et al. 2015) and also from the viral dataset (Brum et al. 2015). They have been examined in Lima-Mendez et al. (2015) and we have used the original relative abundances to visualize the data. Data were retrieved from: <http://www.raeslab.org/companion/ocean-interactome.html>

## 2 Set up Cytoscape and R connection

We will run this example using RCy3 (P. T. Shannon et al. 2013) to drive the visualization of these networks in Cytoscape (P. Shannon et al. 2003) using CyREST (Ono et al. 2015).

### 2.1 Requirements

```
library(RCy3)
library(igraph)
library(RJSONIO)
library(RColorBrewer)
library(httr)
```

To run this example **Cytoscape software must be running**. In Cytoscape we will also need Allegro-plugin for this example.

To begin we create a connection in R that we can use to manipulate the networks and then we will delete any windows that were already in Cytoscape so that we don't use up all of our memory.

```
cy <- CytoscapeConnection()
deleteAllWindows(cy)
```

## 3 Read in data

We will read in a species co-occurrence matrix that was calculated using Spearman Rank coefficient. (If interested in seeing how this was done please see scripts and the raw data in inst/data-raw)

```
## scripts for processing located in "inst/data-raw/"
prok_vir_cor <- read.delim("./data/virus_prok_cor_abundant.tsv")
```

There are many different ways to work with graphs in R. We will use both the igraph (Csardi and Nepusz 2006) and the graph (Gentleman et al. 2016) package to work with our network with Cytoscape.

The igraph package is used to convert the co-occurrence dataframe into a network that we can send to Cytoscape. In this case our graph is undirected (so “directed = FALSE”) since we do not have any information about the direction of the interactions.

```
graph_vir_prok <- simplify(graph.data.frame(prok_vir_cor,
                                             directed = FALSE))
```

## 4 Read in taxonomic classification

Since these are data from small, microscopic organisms that were sequenced using shotgun sequencing, we rely on the classification of the sequences to know what kind of organisms are in the samples. In this case the bacterial viruses (bacteriophage), were classified by Basic Local Alignment Search Tool (BLAST <http://blast.ncbi.nlm.nih.gov/Blast.cgi>) by searching for their closest sequence in the RefSeq database (see methods in Brum et al. (2015)). The prokaryotic taxonomic classifications were determined using the SILVA database.

```
phage_id_affiliation <- read.delim("./data/phage_ids_with_affiliation.tsv")
bac_id_affi <- read.delim("./data/prok_tax_from_silva.tsv")
```

## 5 Add the taxonomic classifications to the network and then send network to Cytoscape

In preparation for sending the networks to Cytoscape we will add in the taxonomic data. Some of the organisms do not have taxonomic classifications associated with them so we have described them as “not\_class” for not classified. We do that because we have had problems sending “NA”s to Cytoscape from RCy3.

```
genenet.nodes <- as.data.frame(vertex.attributes(graph_vir_prok))

## not all have classification, so create empty columns
genenet.nodes$phage_aff <- rep("not_class", nrow(genenet.nodes))
genenet.nodes$Tax_order <- rep("not_class", nrow(genenet.nodes))
genenet.nodes$Tax_subfamily <- rep("not_class", nrow(genenet.nodes))

for (row in seq_along(1:nrow(genenet.nodes))){
  if (genenet.nodes$name[row] %in% phage_id_affiliation$first_sheet.Phage_id_network){
    id_name <- as.character(genenet.nodes$name[row])
    aff_to_add <- unique(subset(phage_id_affiliation,
                              first_sheet.Phage_id_network == id_name,
                              select = c(phage_affiliation,
                                          Tax_order,
                                          Tax_subfamily)))
    genenet.nodes$phage_aff[row] <- as.character(aff_to_add$phage_affiliation)
    genenet.nodes$Tax_order[row] <- as.character(aff_to_add$Tax_order)
    genenet.nodes$Tax_subfamily[row] <- as.character(aff_to_add$Tax_subfamily)
  }
}

## do the same for proks
genenet.nodes$prok_king <- rep("not_class", nrow(genenet.nodes))
genenet.nodes$prok_tax_phylum <- rep("not_class", nrow(genenet.nodes))
genenet.nodes$prok_tax_class <- rep("not_class", nrow(genenet.nodes))

for (row in seq_along(1:nrow(genenet.nodes))){
  if (genenet.nodes$name[row] %in% bac_id_affi$Accession_ID){
    aff_to_add <- unique(subset(bac_id_affi,
                              Accession_ID == as.character(genenet.nodes$name[row]),
                              select = c(Kingdom,
                                          Phylum,
                                          Class)))
    genenet.nodes$prok_king[row] <- as.character(aff_to_add$Kingdom)
    genenet.nodes$prok_tax_phylum[row] <- as.character(aff_to_add$Phylum)
    genenet.nodes$prok_tax_class[row] <- as.character(aff_to_add$Class)
  }
}
```

Add to the network the data related to the connections between the organisms, the edge data, and then prepare to send the nodes and edges to Cytoscape using the function `cyPlot()`.

```

genenet.edges <- data.frame(as_edgelist(graph_vir_prok))
names(genenet.edges) <- c("name.1",
                          "name.2")
genenet.edges$Weight <- edge_attr(graph_vir_prok)[[1]]

genenet.edges$name.1 <- as.character(genenet.edges$name.1)
genenet.edges$name.2 <- as.character(genenet.edges$name.2)
genenet.nodes$name <- as.character(genenet.nodes$name)

ug <- cyPlot(genenet.nodes, genenet.edges)

```

## 6 Send network to Cytoscape using RCy3

Now we will send the network from R to Cytoscape.

```

cw <- CytoscapeWindow("Tara oceans",
                      graph = ug,
                      overwriteWindow = TRUE)

```

```

displayGraph(cw)
layoutNetwork(cw)
fitContent(cw)

```

ph_11569	ph_12966	ph_13668	ph_12560	ph_1164	ph_12916	ph_1815	ph_12236	ph_15108	ph_14357	ph_1895	ph_589	ph_12363	ph_11600	ph_1870	ph_15307	ph_765	ph_53	EU686618	ph_155	ph_4200	ph_5361	ph_1077	ph_6632	ph_5256	ph_4384	ph_427	ph_2050	ph_1573
ph_11080	ph_12798	ph_12485	ph_5287	ph_7153	ph_3830	ph_1173	ph_7186	ph_12402	AY664104	ph_4393	ph_3408	ph_3980	ph_16266	ph_531	ph_1237	ph_424	ph_12231	ph_282	ph_157	ph_6111	ph_242	ph_2293	ph_5606	ph_2418	ph_4446	ph_3158	ph_3889	EF572710
ph_4386	ph_4714	ph_4972	ph_3105	ph_484	ph_253	ph_4344	ph_4405	ph_10747	ph_3767	ph_13832	ph_2857	ph_2739	AA_CY0200813	EU802434	ph_1018	ph_2036	ph_1434	ph_1218	EF573115	ph_3642	ph_1186	ph_2064	ph_3457	ACY02048743	ph_4448	ph_210568	ACY0205648	ph_8590
ph_4314	ph_8168	ph_3030	ph_5249	EU802893	EU805891	ph_2117	ph_7905	ph_5981	ph_1374	ph_3914	ph_3684	ph_3059	ph_864	ph_209	ph_1394	ph_3042	ph_121	ph_2467	ph_229	ph_2008	ph_7963	ph_3070	ph_7594	00061968	ph_16045	ph_674	ph_12136	ph_1285
ph_4452	ph_12530	ph_1471	ph_3621	ph_3028	ph_2424	ph_4841	ph_1524	ph_72	EF571973	ph_22428	ph_3844	ph_3493	ph_2405	ph_1868	ph_439	ph_2233	ph_4100	ph_553	ph_5286	ph_1388	ph_5063	ph_5023	ph_487	ph_3864	EU802893	ph_15150	ph_13413	ph_12287
ph_3301	ph_64	ph_8002	ph_11128	ph_20685	ph_8419	ph_276	AY664189	ph_2305	ph_741	ph_1683	ph_5302	EF574119	ph_646	ph_2021	ph_3204	ph_7902	ph_7389	ph_2175	ph_4165	ph_3211	ph_6185	ph_5924	ph_2172	ph_2246	ph_2352	ph_4384	ph_2297	ph_1895
ph_2452	ph_661	EF573463	ph_11258	ph_3571	ph_4091	ph_17340	ph_2947	ph_4447	ph_392	ph_1303	ph_318	ph_1687	ph_8315	ph_647	ph_2455	ph_7518	EU802971	ph_319	ph_3975	ph_4713	ph_675	ph_5360	ph_27043	ph_21720	ph_3365	ph_16719	ph_3280	ph_4480
ph_236	ph_13890	ph_3067	ph_3201	ph_2682	ph_4558	ph_4801	ph_2468	ph_2242	ph_1728	ph_2150	ph_449	EF571982	AY664012	ph_1188	ph_2143	ph_22514	ph_6280	IN166137	ph_2746	ph_1277	ph_4381	ph_13894	ph_21152	ph_3464	ph_3097	ph_6033	EF573363	ph_4327
ph_4205	ph_11060	ph_327	ph_1762	ph_2242	EF572471	ph_10738	ph_2193	EU759440	ph_300	ph_1715	ph_4150	ph_3246	ph_1909	ph_394	ph_3293	ph_8245	ph_4841	ph_3113	ph_8371	ph_22430	ph_4450	ph_7893	ph_15668	ph_4377	ph_1377	ph_3856	ph_4647	ph_8695
ph_4745	ph_1624	ph_1767	ph_11583	ph_4865	AA_CY0205496	ph_374	ph_1855	EU802437	ph_161	ph_3524	ph_15051	ph_6605	ph_3983	ph_182	AY664083	ph_2032	ph_6655	ph_1378	ph_8064	ph_3817	ph_8407	ph_1253	EU802833	00115296	ph_6235	ph_742	ph_734	ph_1487
ph_4929	ph_362	ph_1473	ph_5955	ph_8068	ph_2273	ph_1376	ph_5609	EF574440	ph_1635	ph_2142	ph_13481	ph_2524	ph_3768	ph_1313	ph_25328	ph_5516	ph_1095	ph_89	00246891	ph_4430	ACY02088370	ph_2108	ph_12858	ph_1634	ph_1222	ph_1561	ph_2807	ph_3219
ph_5321	ph_614	ph_362	ph_3952	ph_3660	ph_7000	ph_23743	ph_1485	ph_8161	ph_3477	ph_1402	ph_1522	ph_10396	AY534095	ph_573044	CY0203844	ph_16274	ph_4206	ph_7792	ph_2775	EF574264	ph_3959	ph_3098	0347414	ph_3626	ph_4287	ph_7880	ph_747	ph_4309
ph_636	ph_1247	ph_12279	ph_784	ph_4434	ph_20642	ph_3528	ph_15039	ph_1880	AA_CY0202554	ph_2730	ph_17518	ph_22720	ph_3267	EU803067	ph_10467	ph_3929	00061415	ph_2281	ph_5723	ph_655	ph_3542	ph_687	HQ672885	EF574484	ph_568	ph_371	ph_1422	ph_7520
ph_2764	ph_1483	ph_8066	ph_2435	00009101	ph_7616	ph_1843	ph_2677	ph_2087	ph_7350	ph_15726	ph_1465	ph_1346	ph_14679	ph_2233	ph_4813	ph_1837	ph_8243	ph_3519	ph_5110	ph_2122	ph_484	ph_3531	EU802701	ph_4892	ph_1205	ph_307	ph_7878	ph_14587
ph_2469	ph_2123	ph_2666	ph_2877	ph_3765	ph_21378	EU808616	ph_1452	ph_2105	ph_14213	ph_2017	00056479	ph_74	ph_1874	ph_7661	AY664000	ph_189	ph_5573	ph_5893	ph_1472	ph_8165	ph_1779	ph_13974	ph_2534	ph_5321	ph_547	ph_1431	ph_6178	ph_20
ph_11386	ph_2466	ph_1390	ph_11038	ph_2984	ph_1534	ph_5762	ph_8727	ph_1312	ph_2424	ph_11135	ph_3183	ph_1120	ph_5625	ph_2051	00227468	ph_1347	ph_15759	ph_8318	ph_15681	ph_325	U40230	ph_2563	ph_2857	ph_2568	AY704967	ph_2956	ph_2256	ph_2006
HQ671905	ph_5245	ph_3504	ph_4655	ph_1146	ph_2878	ph_5629	ph_3217	ph_11146	ph_4160	ph_1438	ph_2017	ph_5983	ph_331	ph_2222	ph_3840	ph_2159	ph_2165	ph_21467	ph_7789	ph_1142	ph_7842	X52163	ph_513	ph_339	ph_2323	ph_9237	ph_2180	ph_6112
ph_5724	ph_2125	ph_3998	ph_3241	00340897	ph_13708	ph_2253	ph_1343	ph_2757	ph_12263	ph_1601	ph_5977	ph_2844	ph_5617	ph_188	ph_2470	00239286	ph_588	ph_1308	ph_5861	ph_3711	ph_22390	ph_1392	ph_7271	ph_5625	ph_2762	ph_1619	ph_3892	ph_11145
ph_2523	ph_24577	ph_299	ph_1137	ph_2488	ph_4307	ph_8462	ph_2911	ph_1631	ph_78	HQ671891	ph_346	ph_3873	ph_2286	ph_2840	ph_4867	ph_3486	ph_1505	ph_6555	ph_2423	ph_18855	ph_872	ph_5727	ph_6133	ph_7564	AB01007386	ph_1800	ph_8962	ph_1668
ph_3240	ph_5626	ph_133	ph_4368	ph_973	00077772	ph_1251	ph_883	ph_1703	ph_8869	ph_11844	ph_1113	ph_4442	ph_14801	ph_4252	ph_12213	ph_2391	ph_1795	ph_2118	ph_4276	ph_4712	ph_51	ph_132212	ph_14784	ph_5010	ph_2368	ph_343	ph_3405	
ph_15141	ph_3606	ph_3901	ph_1575	ph_1443	ph_6800	EF572587	ph_603	ph_4441	ph_6025	ph_835	ph_11106	ph_2918	ph_1903	ph_5643	ph_2981	00145420	ph_5626	ph_6416	CY0202061	ph_13591	00340846	ph_3681	ph_22635	ph_287	ph_6551	ph_11307	ph_1274	ph_1145
ph_2957	ph_6635	ph_6011	ph_5001	ph_5631	ph_12362	ph_451	EU802707	ph_10377	ph_1633	ph_5898	EF574411	ph_407	ph_1427	ph_1325	ph_1879	ph_1727	ph_428	ph_2917	ph_1458	ph_2071	EU802606	ph_8039	ph_684	ph_2280	ph_17617	ph_1326	ph_17206	ph_6594
ph_5518	AY662993	ph_4453	ph_2346	ph_4279	ph_1371	ph_5668	ph_4530	ph_3050	ph_2345	ph_386	ph_11846	ph_4194	ph_1648	00341055	ph_6395	ph_20075	ph_2126	ph_3157	ph_2702	ph_5725	ph_408	ph_4358	ph_533	ph_4444	ph_3287	ph_1316	ph_3572	ph_2360
ph_11774	ph_5241	ph_5396	ph_2778	ph_896	ph_4805	ph_2061	ph_3407	ph_2152	AA_CY0205557	ph_18452	ph_14800	ph_4054	ph_2121	ph_7503	ph_452	ph_176	ph_93	ph_6454	AY663945	ph_4646	ph_1412	ph_2537	AF096371	ph_6441	ph_2612	ph_737	ph_1208	ph_637
ph_25342	EU677220	ph_5728	ph_308	ph_2921	ph_509	00235382	AA_CY0202072	ph_36155	ph_3628	ph_1531	ph_5627	ph_13207	ph_4380	ph_1318	ph_6391	ph_15881	ph_13192	ACY02048139	ph_10744	ph_3099	ph_1725	ph_2711	ph_218	ph_2350	ph_1723	ph_2994	ph_2254	ph_14208
ph_11602	ph_2886	ph_4602	ph_6110	ph_11724	ph_5453	ph_5790	ph_25845	ph_3450	ph_297	ph_18193	ph_8889	ph_1150	ph_2282	ph_1572	ph_5642	00089141	ph_15512	ph_12620	ph_1273	ph_8537	ph_11040	ph_115	ph_608	ph_366	ph_811	EU802865	IN547476	ph_12902
ph_134	ph_10931	ph_5265	00340773	ph_890	ph_4687	ph_8067	ph_2919	ph_4678	ph_3570	ph_7172	ph_4261	ph_1871	ph_7506	ph_3956	ph_8477	ph_1230	ph_865	EU804641	ph_3465	ph_4449	ph_13612	ph_6780	ACY02056232	ph_4199	ph_6439	AY662941	ph_2773	ph_4445
ph_430	EU802543	ph_7142	ph_1326	ph_2677	ph_386	ph_7930	ph_2495	ph_1046	ph_6874	ph_841	ph_2844	ph_892	ph_9222	ph_2447	ph_4541	ph_1534	ph_68	ph_4151	ph_7124	ph_2541	ph_4329	ph_2855	ph_1373	00001453	ph_4587	ph_1425	ph_12624	ph_2681
ph_3273	ph_3772	EF45158	ph_6109	ph_2955	ph_1367	ph_345	ph_4393	ph_58	ph_5944	ph_2140	ph_2860	ph_2698	ph_3111	ph_9185	ph_1041	ph_587	ph_17327	ph_6115	ph_4251	ph_11048	ph_4240	ph_2337	ph_3879	ph_5346	ph_1297	ph_4443	ph_5484	ph_3024
ph_357	ph_4396	ph_11738	ph_18998																									

## 7 Colour network by prokaryotic phylum

We would like to get an overview of the different phylum of bacteria that are in the network. One way is to colour the different nodes based on their phylum classification. The package Rcolorbrewer will be used to generate a set of good colours for the nodes.

```
families_to_colour <- unique(genenet.nodes$prok_tax_phylum)
families_to_colour <- families_to_colour[!families_to_colour %in% "not_class"]
node.colour <- brewer.pal(length(families_to_colour),
                          "Set3")
```

Use the colours from Rcolorbrewer to colour the nodes in Cytoscape.

```
setNodeColorRule(cw,
                 "prok_tax_phylum",
                 families_to_colour,
                 node.colour,
                 "lookup",
                 default.color = "#ffffff")
```

## Successfully set rule.

```
displayGraph(cw)
layoutNetwork(cw)
fitContent(cw)
```

ph_11569	ph_2966	ph_1369	ph_2560	ph_1364	ph_2916	ph_815	ph_2226	ph_5108	ph_4357	ph_895	ph_589	ph_2363	ph_1600	ph_1870	ph_15307	ph_765	ph_53	EU60618	ph_155	ph_4200	ph_5361	ph_1077	ph_6322	ph_5256	ph_4384	ph_627	ph_2050	ph_1573	
ph_1080	ph_2798	ph_2485	ph_5287	ph_7153	ph_3890	ph_1173	ph_7186	ph_2402	AY64104	ph_4993	ph_3408	ph_3980	ph_16266	ph_531	ph_1237	ph_424	ph_2231	ph_282	ph_157	ph_6111	ph_243	ph_2293	ph_5606	ph_2418	ph_4446	ph_3158	ph_3889	EF572710	
ph_4386	ph_4714	ph_4572	ph_3105	ph_604	ph_253	ph_4544	ph_4409	ph_10747	ph_3767	ph_13932	ph_3057	ph_2739	AA02004813	EU802434	ph_1018	ph_2036	ph_2434	ph_2318	EF572115	ph_3642	ph_1186	ph_2064	ph_3457	AA02004813	ph_4448	ph_21056	AY62056418	ph_8590	
ph_4314	ph_8168	ph_2030	ph_5249	EU802893	EU805091	ph_2117	ph_7906	ph_5981	ph_1374	ph_2014	ph_3684	ph_3059	ph_864	ph_209	ph_1354	ph_3042	ph_121	ph_2467	ph_229	ph_2008	ph_7363	ph_3070	ph_7594	EU061586	ph_16045	ph_674	ph_13126	ph_1385	
ph_4452	ph_2530	ph_1471	ph_3621	ph_3628	ph_2624	ph_6841	ph_1924	ph_72	EF571973	ph_22428	ph_2164	ph_3493	ph_2405	ph_1808	ph_493	ph_2233	ph_4100	ph_553	ph_5286	ph_1388	ph_5043	ph_5023	ph_487	ph_2866	EU803283	ph_15150	ph_13413	ph_32287	
ph_3301	ph_64	ph_8002	ph_1128	ph_20605	ph_8419	ph_276	AY64189	ph_2305	ph_741	ph_1682	ph_5302	EF574113	ph_646	ph_2031	ph_3204	ph_7902	ph_7389	ph_2175	ph_4165	ph_3311	ph_6185	ph_5924	ph_2172	ph_2846	ph_2352	ph_4384	ph_2297	ph_1895	
ph_2452	ph_661	EF573463	ph_1258	ph_3571	ph_4001	ph_17340	ph_2947	ph_4447	ph_392	ph_1033	ph_318	ph_1687	ph_8335	ph_647	ph_2455	ph_7518	EU802971	ph_319	ph_3979	ph_4713	ph_675	ph_5360	ph_27049	ph_21720	ph_3369	ph_16719	ph_3288	ph_4480	
ph_236	ph_13890	ph_3067	ph_2201	ph_2682	ph_4558	ph_4801	ph_2468	ph_2242	ph_1728	ph_2150	ph_449	EF571362	AY64012	ph_1188	ph_2143	ph_22514	ph_6280	IN166137	ph_2746	ph_1277	ph_4381	ph_13894	ph_2152	ph_3464	ph_2097	ph_6033	EF573363	ph_4327	
ph_4205	ph_1060	ph_327	ph_1762	ph_2242	EF572471	ph_10738	ph_2193	EU759440	ph_300	ph_1715	ph_4150	ph_3246	ph_1309	ph_394	ph_2293	ph_8245	ph_4841	ph_3113	ph_8371	ph_22430	ph_4450	ph_7893	ph_15668	ph_4377	ph_1377	ph_3856	ph_4647	ph_8635	
ph_4745	ph_6424	ph_1767	ph_11583	ph_4865	AA02054568	ph_374	ph_1855	EU802437	ph_161	ph_3524	ph_15051	ph_6605	ph_3983	ph_182	AY64083	ph_2032	ph_6455	ph_13578	ph_8064	ph_3817	ph_8407	ph_1293	EU802833	OU115296	ph_6235	ph_742	ph_734	ph_1487	
ph_4929	ph_362	ph_1473	ph_5935	ph_8068	ph_2273	ph_1376	ph_5609	EF574440	ph_1635	ph_2142	ph_12481	ph_2524	ph_3768	ph_1313	ph_25328	ph_5516	ph_1095	ph_89	OU046891	ph_4430	AA020288370	ph_2108	ph_12858	ph_1634	ph_222	ph_1561	ph_2807	ph_3219	
ph_5321	ph_614	ph_362	ph_10952	ph_3660	ph_7000	ph_33749	ph_1485	ph_8161	ph_3477	ph_1402	ph_1522	ph_10936	AY534095	ph_57304	OU0203944	ph_16274	ph_4206	ph_7792	ph_2775	EF574264	ph_3959	ph_3098	OU047416	ph_2626	ph_4207	ph_7880	ph_747	ph_4599	
ph_636	ph_1247	ph_2279	ph_784	ph_4434	ph_20642	ph_3538	ph_15039	ph_1800	AA02025554	ph_2730	ph_17518	ph_22728	ph_3267	EU803067	ph_10467	ph_3929	OU061415	ph_2281	ph_5723	ph_655	ph_3542	ph_687	HQ672883	EF574484	ph_368	ph_371	ph_1422	ph_7520	
ph_2764	ph_14481	ph_8066	ph_2435	OU009161	ph_7616	ph_1849	ph_2677	ph_2087	ph_7350	ph_15726	ph_1459	ph_1346	ph_14679	ph_2232	ph_4812	ph_1837	ph_8243	ph_3519	ph_5110	ph_2122	ph_484	ph_3531	EU802700	ph_5892	ph_1205	ph_307	ph_7878	ph_14587	
ph_2463	ph_2123	ph_2666	ph_2877	ph_3769	ph_21378	EU086616	ph_1452	ph_2105	ph_14213	ph_2017	OU056479	ph_74	ph_1874	ph_7661	AY64000	ph_189	ph_5973	ph_5893	ph_1472	ph_8185	ph_1779	ph_13974	ph_2534	ph_5921	ph_347	ph_1431	ph_6178	ph_20	
ph_11386	ph_2466	ph_1390	ph_11038	ph_2984	ph_1534	ph_5762	ph_8727	ph_1312	ph_2424	ph_11135	ph_3183	ph_1120	ph_5625	ph_2051	OU227468	ph_1347	ph_15759	ph_8318	ph_15681	ph_325	U40230	ph_3569	ph_3857	ph_3560	AY704567	ph_2956	ph_2056	ph_3066	
HQ671905	ph_5245	ph_3504	ph_4655	ph_1146	ph_2878	ph_5629	ph_3217	ph_11146	ph_4160	ph_1438	ph_2017	ph_5983	ph_331	ph_2222	ph_3840	ph_2159	ph_2165	ph_21467	ph_7789	ph_1142	ph_7842	X52165	ph_513	ph_339	ph_2323	ph_9237	ph_2180	ph_6112	
ph_5724	ph_2125	ph_3998	ph_3241	OU940897	ph_13708	ph_2253	ph_1343	ph_2757	ph_12263	ph_1601	ph_5577	ph_2844	ph_5617	ph_188	ph_2470	OU239286	ph_588	ph_1308	ph_5861	ph_3711	ph_22390	ph_1392	ph_7271	ph_5635	ph_2762	ph_1619	ph_3892	ph_11145	
ph_2523	ph_24577	ph_299	ph_1197	ph_2488	ph_4537	ph_8462	ph_2911	ph_1631	ph_78	HQ671891	ph_346	ph_3873	ph_2286	ph_2840	ph_4867	ph_3406	ph_1505	ph_6555	ph_2432	ph_18855	ph_872	ph_5727	ph_6113	ph_7504	AA01007386	ph_1800	ph_8962	ph_16688	
ph_3240	ph_5636	ph_193	ph_4368	ph_973	OU077772	ph_1291	ph_883	ph_1703	ph_8869	ph_11844	ph_1113	ph_4442	ph_14801	ph_4252	ph_12213	ph_2278	ph_2391	ph_1795	ph_2118	ph_18	ph_4276	ph_6712	ph_51	ph_12212	ph_4784	ph_5010	ph_2368	ph_343	ph_3405
ph_15141	ph_3606	ph_3901	ph_1575	ph_1441	ph_6800	EF572587	ph_601	ph_4441	ph_6025	ph_893	ph_11096	ph_2918	ph_1903	ph_5643	ph_2981	OU145420	ph_5626	ph_64164	AA0202963	ph_13931	OU940846	ph_3681	ph_20695	ph_287	ph_6551	ph_11307	ph_1274	ph_1143	
ph_2957	ph_6635	ph_6011	ph_5001	ph_5631	ph_12362	ph_451	EU802707	ph_10377	ph_1633	ph_5898	EF574411	ph_407	ph_1427	ph_1925	ph_1879	ph_1727	ph_428	ph_2917	ph_1458	ph_2071	EU802606	ph_8039	ph_684	ph_2280	ph_17617	ph_1326	ph_17206	ph_6594	
ph_5518	AY663939	ph_4453	ph_2346	ph_4279	ph_1371	ph_5668	ph_4530	ph_3050	ph_2345	ph_3886	ph_11846	ph_4194	ph_1648	OU941095	ph_6395	ph_20075	ph_2126	ph_3157	ph_2702	ph_5725	ph_408	ph_4358	ph_533	ph_4444	ph_3287	ph_116	ph_3572	ph_2360	
ph_11774	ph_5241	ph_5396	ph_2778	ph_896	ph_4805	ph_2061	ph_3407	ph_2192	AA02055577	ph_18452	ph_14800	ph_4054	ph_2121	ph_7593	ph_452	ph_176	ph_93	ph_6424	AY662945	ph_4646	ph_1412	ph_2537	AF096371	ph_6441	ph_2612	ph_737	ph_1208	ph_637	
ph_25342	HQ672720	ph_5728	ph_308	ph_2921	ph_509	OU2253204	AA02020721	ph_36195	ph_3628	ph_1531	ph_5627	ph_13207	ph_4380	ph_1318	ph_6391	ph_15881	ph_13192	AA0204813	ph_10744	ph_3039	ph_1725	ph_2711	ph_218	ph_2350	ph_1723	ph_2954	ph_2254	ph_14208	
ph_1662	ph_2806	ph_4602	ph_6110	ph_11724	ph_5453	ph_5790	ph_25845	ph_3450	ph_397	ph_15193	ph_8889	ph_1150	ph_2282	ph_1572	ph_5642	OU009141	ph_15512	ph_12620	ph_1273	ph_8537	ph_11040	ph_115	ph_608	ph_366	ph_811	EU802865	IN547476	ph_12902	
ph_134	ph_10931	ph_5265	OU040773	ph_890	ph_4687	ph_8067	ph_2919	ph_4678	ph_3570	ph_7172	ph_4261	ph_1871	ph_7506	ph_3936	ph_8477	ph_1230	ph_865	EU804641	ph_3465	ph_4449	ph_13612	ph_6780	AA0205232	ph_4199	ph_6439	EF663341	ph_2773	ph_4445	
ph_430	EU802543	ph_7043	ph_1326	ph_2677	ph_386	ph_7930	ph_2458	ph_1046	ph_6874	ph_381	ph_2844	ph_892	ph_9222	ph_2447	ph_4641	ph_1534	ph_68	ph_4151	ph_7124	ph_2541	ph_4339	ph_2855	ph_1773	EU061453	ph_4587	ph_1425	ph_12634	ph_2681	
ph_3373	ph_3772	EF745138	ph_6109	ph_2955	ph_1367	ph_345	ph_4393	ph_58	ph_3944	ph_2140	ph_2360	ph_2698	ph_3111	ph_3185	ph_1041	ph_387	ph_17327	ph_6115	ph_4251	ph_11048	ph_4240	ph_2337	ph_3879	ph_5346	ph_1297	ph_4443	ph_5484	ph_3024	
ph_357	ph_4936	ph_11730	ph_18998																										



## 7.1 Set node shape to reflect virus or prokaryote

Next we would like to change the shape of the node to reflect whether the nodes are viral or prokaryotic in origin. In this dataset all of the viral node names start with “ph\_”, thus we can set the viral nodes to be diamond-shaped by looking for all the nodes that start with “ph” in the network.

```
shapes_for_nodes <- c("DIAMOND")

phage_names <- grep("ph_",
                    genenet.nodes$name,
                    value = TRUE)

setNodeShapeRule(cw,
                 "label",
                 phage_names,
                 shapes_for_nodes)
```

## Successfully set rule.

```
displayGraph(cw)
fitContent(cw)
```

ph\_11569 ph\_2966 ph\_1969 ph\_2560 ph\_164 ph\_2916 ph\_815 ph\_7296 ph\_5108 ph\_4357 ph\_895 ph\_589 ph\_2963 ph\_1600 ph\_1870 ph\_15307 ph\_765 ph\_53 EU60618 ph\_155 ph\_4200 ph\_5361 ph\_1077 ph\_6632 ph\_5256 ph\_6284 ph\_627 ph\_2850 ph\_1572

ph\_1080 ph\_2798 ph\_2485 ph\_5287 ph\_7153 ph\_3830 ph\_1173 ph\_7184 ph\_2402 AY64104 ph\_4932 ph\_3408 ph\_3980 ph\_16266 ph\_531 ph\_1237 ph\_424 ph\_2231 ph\_382 ph\_157 ph\_6111 ph\_243 ph\_2293 ph\_5606 ph\_2418 ph\_4446 ph\_3158 ph\_3883 EF572710

ph\_4386 ph\_4714 ph\_4972 ph\_9105 ph\_654 ph\_253 ph\_4344 ph\_4405 ph\_10747 ph\_3767 ph\_18932 ph\_3057 ph\_2739 AACV02008017 AY62434 ph\_1016 ph\_2036 ph\_2434 ph\_3315 EF573115 ph\_3642 ph\_1184 ph\_2064 ph\_3457 AACV02048763 ph\_4448 ph\_2105 AACV02054409 ph\_8590

ph\_4314 ph\_8168 ph\_2632 ph\_5245 EU802892 EU805091 ph\_2117 ph\_7304 ph\_5381 ph\_1374 ph\_2914 ph\_3684 ph\_3050 ph\_864 ph\_309 ph\_3354 ph\_3042 ph\_121 ph\_2467 ph\_229 ph\_2008 ph\_7363 ph\_3070 ph\_7534 EU061586 ph\_16045 ph\_674 ph\_13126 ph\_1385

ph\_4452 ph\_3530 ph\_1471 ph\_3621 ph\_3028 ph\_2624 ph\_6841 ph\_1324 ph\_72 EF571973 ph\_22426 ph\_2184 ph\_3493 ph\_2405 ph\_1804 ph\_459 ph\_2231 ph\_4100 ph\_553 ph\_5286 ph\_1383 ph\_5043 ph\_5023 ph\_487 ph\_3866 EU802803 ph\_15151 ph\_1342 ph\_32287

ph\_3301 ph\_64 ph\_8902 ph\_1128 ph\_2869 ph\_8419 ph\_276 AY664189 ph\_2205 ph\_741 ph\_1682 ph\_5302 EF574119 ph\_646 ph\_2021 ph\_2204 ph\_7902 ph\_7389 ph\_3175 ph\_4165 ph\_3211 ph\_6185 ph\_5924 ph\_2172 ph\_2846 ph\_2952 ph\_4384 ph\_2297 ph\_1895

ph\_2452 ph\_561 EF573463 ph\_1228 ph\_2571 ph\_4091 ph\_1724 ph\_2941 ph\_4447 ph\_352 ph\_1035 ph\_318 ph\_1697 ph\_8335 ph\_647 ph\_2455 ph\_7518 EU802931 ph\_315 ph\_2975 ph\_4713 ph\_675 ph\_5360 ph\_2763 ph\_21735 ph\_3363 ph\_16715 ph\_3280 ph\_4480

ph\_236 ph\_13890 ph\_3067 ph\_3201 ph\_2692 ph\_4558 ph\_4801 ph\_2460 ph\_2242 ph\_1728 ph\_3150 ph\_449 EF571562 AY664012 ph\_1188 ph\_2143 ph\_22514 ph\_6280 IN166137 ph\_3746 ph\_1277 ph\_4381 ph\_19814 ph\_21152 ph\_3464 ph\_3097 ph\_6032 EF573263 ph\_4327

ph\_4205 ph\_1060 ph\_327 ph\_1762 ph\_2242 EF572471 ph\_10734 ph\_2153 EU759440 ph\_300 ph\_1715 ph\_4150 ph\_3246 ph\_1909 ph\_394 ph\_2233 ph\_8245 ph\_4841 ph\_9113 ph\_8971 ph\_2243 ph\_4450 ph\_7893 ph\_15606 ph\_4377 ph\_1377 ph\_3856 ph\_4647 ph\_8695

ph\_4745 ph\_6424 ph\_1167 ph\_11583 ph\_4865 AACV02054369 ph\_374 ph\_1855 EU802437 ph\_161 ph\_3534 ph\_15051 ph\_6665 ph\_3983 ph\_162 AY664083 ph\_2032 ph\_6655 ph\_3978 ph\_8064 ph\_3817 ph\_8407 ph\_1253 EU802833 ON115296 ph\_6235 ph\_742 ph\_734 ph\_1487

ph\_4929 ph\_363 ph\_1472 ph\_5595 ph\_8068 ph\_2273 ph\_1376 ph\_5600 EF574440 ph\_1635 ph\_2142 ph\_13481 ph\_2524 ph\_3768 ph\_131 ph\_2532 ph\_5516 ph\_1695 ph\_89 00246893 ph\_4420 AACV020289370 ph\_2104 ph\_1285 ph\_1634 ph\_222 ph\_1561 ph\_2807 ph\_3219

ph\_5321 ph\_614 ph\_342 ph\_3932 ph\_3640 ph\_7001 ph\_3374 ph\_1485 ph\_8161 ph\_3477 ph\_1493 ph\_1522 ph\_10936 AY536099 ph\_5710 AACV02039445 ph\_16230 ph\_4206 ph\_7791 ph\_2775 EF574264 ph\_3993 ph\_3093 10347414 ph\_2626 ph\_4287 ph\_7885 ph\_747 ph\_4399

ph\_636 ph\_1247 ph\_2273 ph\_784 ph\_4424 ph\_2064 ph\_3524 ph\_15023 ph\_1880 AACV020255495 ph\_2725 ph\_17914 ph\_12272 ph\_3267 EU803067 ph\_10467 ph\_3923 EU061419 ph\_2281 ph\_5723 ph\_655 ph\_3542 ph\_687 HQ672883 EF574484 ph\_968 ph\_371 ph\_1422 ph\_7520

ph\_2766 ph\_1648 ph\_8064 ph\_2435 00009161 ph\_7856 ph\_1843 ph\_2671 ph\_2087 ph\_7350 ph\_15725 ph\_1463 ph\_1246 ph\_1467 ph\_2223 ph\_4813 ph\_1827 ph\_8243 ph\_3915 ph\_3110 ph\_2122 ph\_484 ph\_3531 EU802700 ph\_6852 ph\_1205 ph\_307 ph\_7878 ph\_14587

ph\_2460 ph\_2123 ph\_2665 ph\_2877 ph\_9763 ph\_21378 EU086616 ph\_1452 ph\_2105 ph\_14215 ph\_2017 Q0156470 ph\_74 ph\_1874 ph\_7661 AY664000 ph\_189 ph\_9573 ph\_5895 ph\_1472 ph\_8185 ph\_1775 ph\_13974 ph\_2524 ph\_5921 ph\_547 ph\_1431 ph\_6178 ph\_20

ph\_11285 ph\_2466 ph\_1990 ph\_11039 ph\_2984 ph\_1534 ph\_5782 ph\_8727 ph\_1312 ph\_2424 ph\_11135 ph\_3183 ph\_1120 ph\_5625 ph\_2051 EU227468 ph\_1347 ph\_15759 ph\_9316 ph\_15681 ph\_376 U40230 ph\_3503 ph\_3057 ph\_3560 AY726367 ph\_2956 ph\_2056 ph\_3066

HQ671905 ph\_9245 ph\_2504 ph\_4655 ph\_1146 ph\_2878 ph\_5625 ph\_2217 ph\_11144 ph\_4160 ph\_1438 ph\_2017 ph\_5983 ph\_331 ph\_2222 ph\_3846 ph\_2155 ph\_2165 ph\_21467 ph\_7785 ph\_1142 ph\_7842 X52169 ph\_913 ph\_323 ph\_2323 ph\_9237 ph\_2180 ph\_6112

ph\_5724 ph\_2125 ph\_3938 ph\_3241 01340897 ph\_13704 ph\_2223 ph\_1343 ph\_2757 ph\_1225 ph\_1601 ph\_5977 ph\_2844 ph\_5617 ph\_188 ph\_2476 D0239286 ph\_588 ph\_1308 ph\_5841 ph\_3711 ph\_22290 ph\_1392 ph\_7271 ph\_5625 ph\_3763 ph\_1613 ph\_3892 ph\_11145

ph\_2593 ph\_2457 ph\_293 ph\_1197 ph\_2488 ph\_4537 ph\_8463 ph\_2311 ph\_1621 ph\_78 0671891 ph\_344 ph\_3571 ph\_2285 ph\_2846 ph\_4863 ph\_3406 ph\_1505 ph\_6555 ph\_2437 ph\_1805 ph\_872 ph\_572 ph\_6113 ph\_7564 IN01007230 ph\_1800 ph\_8943 ph\_16688

ph\_3240 ph\_5626 ph\_192 ph\_4364 ph\_573 00377772 ph\_1231 ph\_883 ph\_1703 ph\_8863 ph\_11844 ph\_1113 ph\_4442 ph\_14801 ph\_4252 ph\_12214 ph\_2278 ph\_1291 ph\_1795 ph\_2118 ph\_4276 ph\_6712 ph\_51 ph\_12212 ph\_4784 ph\_5010 ph\_3268 ph\_343 ph\_3405

ph\_15141 ph\_3605 ph\_3707 ph\_1575 ph\_1441 ph\_6800 EF672587 ph\_6301 ph\_4441 ph\_8025 ph\_393 ph\_11095 ph\_2918 ph\_1903 ph\_5643 ph\_3931 ON145420 ph\_5926 ph\_6416 AY0202963 ph\_13991 01946894 ph\_2601 ph\_28655 ph\_787 ph\_5951 ph\_11307 ph\_1274 ph\_1143

ph\_2257 ph\_6639 ph\_6011 ph\_5001 ph\_9631 ph\_12362 ph\_491 EU802707 ph\_10377 ph\_1623 ph\_5828 EF574411 ph\_407 ph\_1427 ph\_1325 ph\_1875 ph\_1727 ph\_428 ph\_2917 ph\_1458 ph\_2071 EU802606 ph\_8033 ph\_684 ph\_2280 ph\_17617 ph\_1326 ph\_17205 ph\_6594

ph\_5514 AY663935 ph\_4453 ph\_2246 ph\_4275 ph\_171 ph\_5608 ph\_4520 ph\_3050 ph\_2245 ph\_886 ph\_11844 ph\_4194 ph\_1648 00941095 ph\_6395 ph\_20075 ph\_2126 ph\_3197 ph\_2792 ph\_57735 ph\_400 ph\_4394 ph\_533 ph\_4444 ph\_3287 ph\_116 ph\_3572 ph\_2360

ph\_11774 ph\_5241 ph\_5395 ph\_2778 ph\_896 ph\_4805 ph\_2061 ph\_3401 ph\_2152 AACV020555723 ph\_18452 ph\_14801 ph\_4054 ph\_2121 ph\_7503 ph\_452 ph\_176 ph\_93 ph\_6424 AY663945 ph\_4646 ph\_1412 ph\_2527 AF096371 ph\_6441 ph\_2612 ph\_757 ph\_1208 ph\_6597

ph\_25342 HQ672720 ph\_5728 ph\_308 ph\_2921 ph\_569 00225232 AY02020722 ph\_36124 ph\_3626 ph\_15231 ph\_5627 ph\_13207 ph\_4380 ph\_1318 ph\_6393 ph\_15883 ph\_13152 AACV020481370 ph\_10744 ph\_3039 ph\_1725 ph\_2711 ph\_210 ph\_2350 ph\_1723 ph\_2994 ph\_2254 ph\_14208

ph\_1662 ph\_2806 ph\_4602 ph\_6110 ph\_11724 ph\_5453 ph\_5733 ph\_25845 ph\_3450 ph\_357 ph\_19197 ph\_8889 ph\_1150 ph\_2282 ph\_1572 ph\_5642 D0809141 ph\_15913 ph\_12630 ph\_1272 ph\_8537 ph\_11040 ph\_115 ph\_608 ph\_364 ph\_811 EU802865 IN547476 ph\_13902

ph\_134 ph\_10931 ph\_5265 00540773 ph\_890 ph\_4687 ph\_8067 ph\_2319 ph\_4678 ph\_3570 ph\_7172 ph\_4261 ph\_1871 ph\_7506 ph\_3936 ph\_8477 ph\_1230 ph\_865 EU804644 ph\_3465 ph\_4445 ph\_13612 ph\_6780 AACV02052322 ph\_4159 ph\_6435 AY663241 ph\_2773 ph\_4445

ph\_430 EU802543 ph\_7043 ph\_1926 ph\_2677 ph\_386 ph\_7923 ph\_2459 ph\_1046 ph\_6874 ph\_841 ph\_2044 ph\_892 ph\_9222 ph\_2441 ph\_4641 ph\_1554 ph\_68 ph\_1151 ph\_7134 ph\_3541 ph\_4328 ph\_2855 ph\_1773 00061453 ph\_4587 ph\_1425 ph\_12434 ph\_2681

ph\_2373 ph\_3772 EF745138 ph\_6105 ph\_2955 ph\_1367 ph\_345 ph\_4393 ph\_58 ph\_3944 ph\_2140 ph\_2360 ph\_2698 ph\_3111 ph\_9185 ph\_1041 ph\_987 ph\_17327 ph\_6115 ph\_4251 ph\_11048 ph\_4240 ph\_2327 ph\_3875 ph\_5346 ph\_1297 ph\_4443 ph\_5484 ph\_3024

ph\_357 ph\_4936 ph\_11730 ph\_18998

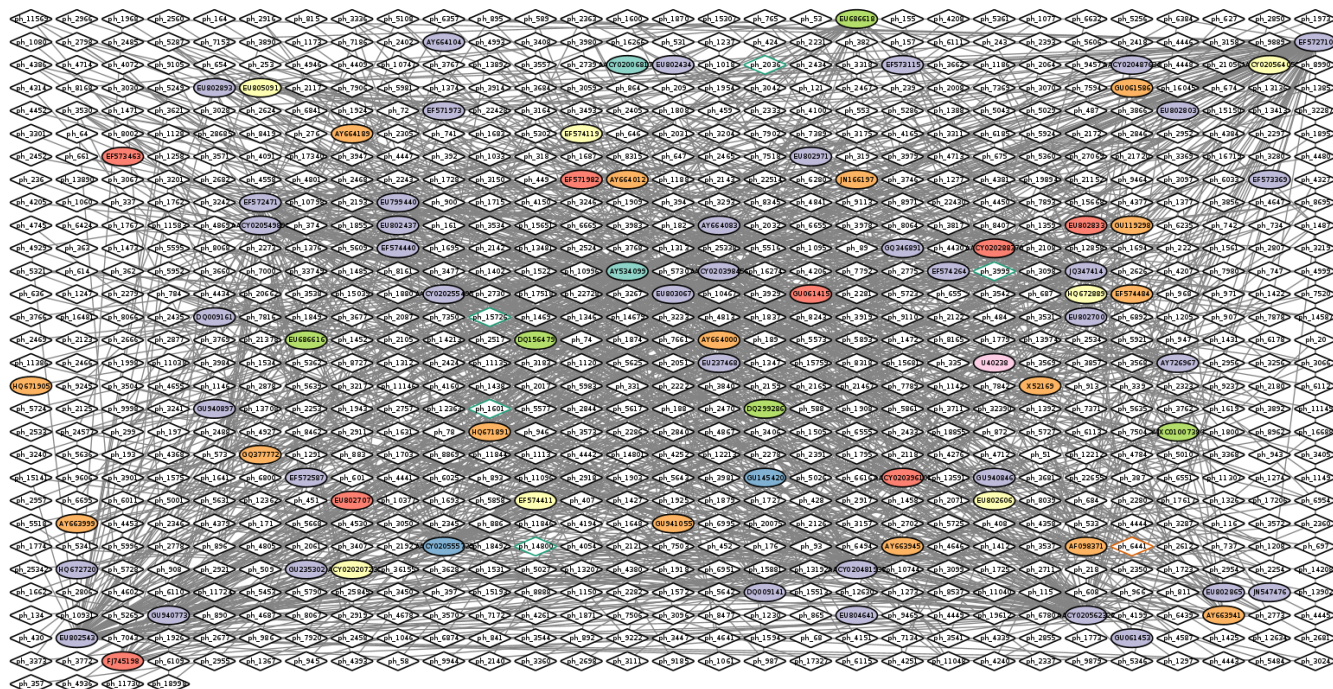
## 8 Colour edges of phage nodes

The classification of the viral data was done in a very conservative manner so not many of the viral nodes were identified. However, if we do want to add some of this information to our visualization we can colour the edges of the viral nodes by family. The main families that were identified in this dataset are the *Podoviridae*, the *Siphoviridae* and the *Myoviridae* (for more info see NCBI Podoviridae, NCBI Myoviridae, and NCBI Siphoviridae)

```
setDefaultNodeBorderWidth(cw, 5)
families_to_colour <- c(" Podoviridae",
                        " Siphoviridae",
                        " Myoviridae")
node.colour <- brewer.pal(length(families_to_colour),
                          "Dark2")
setNodeBorderColorRule(cw,
                       "Tax_subfamily",
                       families_to_colour,
                       node.colour,
                       "lookup",
                       default.color = "#000000")
```

```
## Successfully set rule.
```

```
displayGraph(cw)
fitContent(cw)
```



## 9 Do layout to minimize overlap of nodes.

After doing all of this colouring to the network we would like to layout the network in a way that allows us to more easily see which nodes are connected without overlap. To do the layout we will use the Cytoscape plugin Allegro.

When using RCy3 to drive Cytoscape, if we are not sure what the current values are for a layout or we are not sure what kinds of values are accepted for the different parameters of our layout, we can investigate using the RCy3 functions `getLayoutPropertyNames()` and then `getLayoutPropertyValue()`.

```
getLayoutNames(cw)
```

```
## [1] "attribute-circle"
## [2] "allegro-weak-clustering"
## [3] "allegro-edge-repulsive-fruchterman-reingold"
## [4] "stacked-node-layout"
## [5] "allegro-edge-repulsive-strong-clustering"
## [6] "allegro-strong-clustering"
## [7] "degree-circle"
## [8] "allegro-fruchterman-reingold"
## [9] "allegro-edge-repulsive-spring-electric"
## [10] "circular"
## [11] "attributes-layout"
## [12] "kamada-kawai"
## [13] "force-directed"
## [14] "allegro-edge-repulsive-weak-clustering"
## [15] "grid"
## [16] "hierarchical"
## [17] "allegro-spring-electric"
## [18] "fruchterman-rheingold"
## [19] "isom"
```

```
getLayoutPropertyNames(cw, layout.name = "allegro-spring-electric")
```

```
## [1] "randomize" "maxIterations"
## [3] "noOverlapIterations" "deviceSelection"
## [5] "componentProcessingSelection" "componentSorting"
## [7] "scale" "gravityTypeSelection"
## [9] "gravity"
```

```
getLayoutPropertyValue(cw, "allegro-spring-electric", "gravity")
```

```
## [[1]]
## [1] 100
```

```
getLayoutPropertyValue(cw, "allegro-spring-electric", "maxIterations")
```

```
## [[1]]
## [1] 2000
```

```
getLayoutPropertyValue(cw, "allegro-spring-electric", "noOverlapIterations")
```

```
## [[1]]  
## [1] TRUE
```

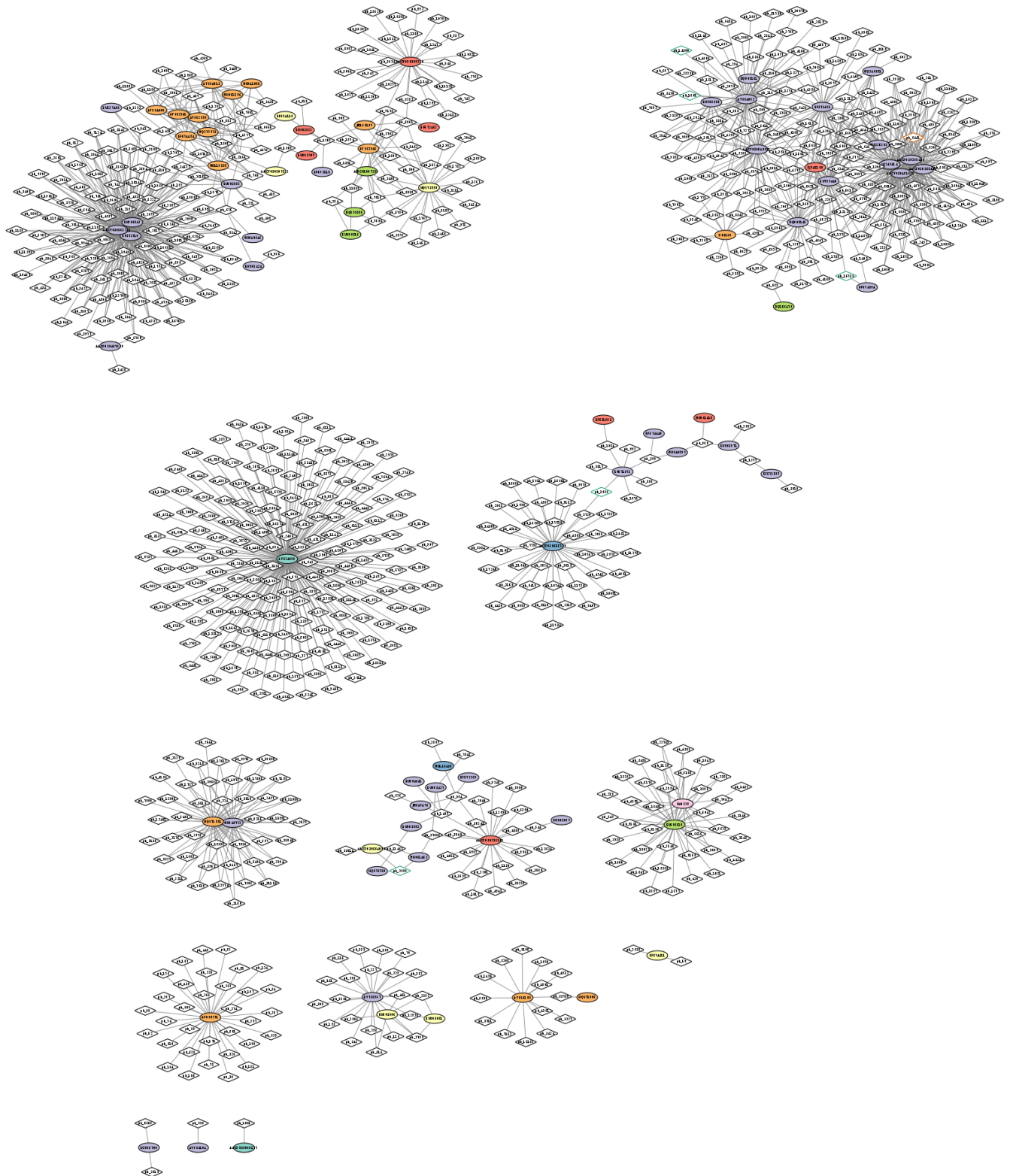
Once we decide on the properties we want, we can go ahead and set them like this:

```
setLayoutProperties(cw,  
  layout.name = "allegro-spring-electric",  
  list(gravity = 100,  
    scale = 6))
```

```
## Successfully updated the property 'gravity'.  
## Successfully updated the property 'scale'.
```

```
layoutNetwork(cw,  
  layout.name = "allegro-spring-electric")  
fitContent(cw)
```





## 10 Look at network properties

One thing that might be interesting to visualize is nodes that are connected to many different nodes and nodes that are connected to few other nodes. The number of other nodes to which one node is connected is called degree. We can use a gradient of size to quickly visualize nodes that have high degree.

```
## initiate a new node attribute
ug2 <- initNodeAttribute(graph = ug,
                        "degree",
                        "numeric",
                        0.0)

## degree from graph package for undirected graphs not working well,
## so instead using igraph to calculate this from the original graph
nodeData(ug2, nodes(ug2), "degree") <- degree(graph_vir_prok)

cw2 <- CytoscapeWindow("Tara oceans with degree",
                      graph = ug2,
                      overwriteWindow = TRUE)
```

```
displayGraph(cw2)
layoutNetwork(cw2)
```

## 11 Size by degree

```
degree_control_points <- c(min(degree(graph_vir_prok)),
                          mean(degree(graph_vir_prok)),
                          max(degree(graph_vir_prok)))

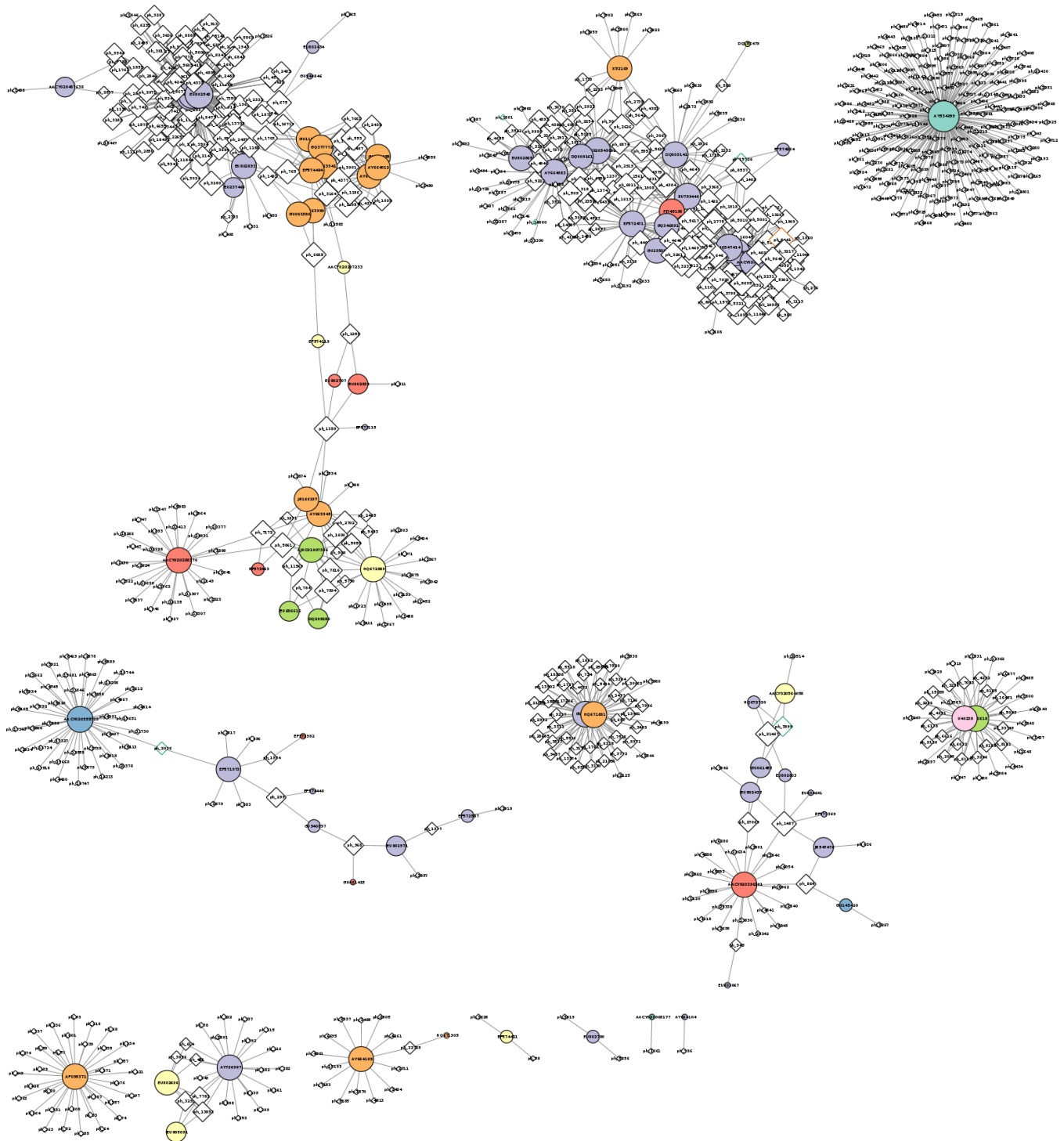
node_sizes <- c(20,
               20,
               80,
               100,
               110) # number of control points in interpolation mode,
                  # the first and the last are for sizes "below" and "above" the attribute seen.

setNodeSizeRule(cw2,
                "degree",
                degree_control_points,
                node_sizes,
                mode = "interpolate")

## Locked node dimensions successfully even if the check box is not ticked.
## Locked node dimensions successfully even if the check box is not ticked.
## Successfully set rule.

layoutNetwork(cw2,
              "force-directed")
```





## 12 Select an interesting node and make a subnetwork from it

The visualization displays several different areas where there are highly connected nodes that are in the same bacterial phylum. We will select one of these nodes, all of the nodes connected to this node, its first neighbours, and then the nodes connected to the first neighbours. One node that is in a group of highly connected nodes is the cyanobacterial node “GQ377772”. We will select it and its first and second neighbours and then make a new network from these nodes and their connections.

```
selectNodes(cw2,
            "GQ377772") # selects specific nodes
getSelectedNodes(cw2)
```

```
## [1] "GQ377772"
```

```
selectFirstNeighborsOfSelectedNodes(cw2)
getSelectedNodes(cw2)
```

```
## [1] "ph_3164" "ph_1392" "ph_1808" "ph_3901" "ph_407" "ph_4377"
## [7] "ph_553" "ph_765" "ph_7661" "GQ377772"
```

Now select the second neighbours of node “GQ377772”.

```
selectFirstNeighborsOfSelectedNodes(cw2)
getSelectedNodes(cw2)
```

```
## [1] "ph_3164"      "ph_1392"      "ph_1808"      "ph_3901"
## [5] "ph_407"      "ph_4377"      "ph_553"       "ph_765"
## [9] "ph_7661"      "AACY020207233" "AY663941"     "AY663999"
## [13] "AY664000"     "AY664012"     "EF574484"     "EU802893"
## [17] "GQ377772"     "GU061586"     "GU119298"     "GU941055"
```

This has only selected the nodes, but not the edges in Cytoscape, so we will need to select all of the edges before we make the new subnetwork.

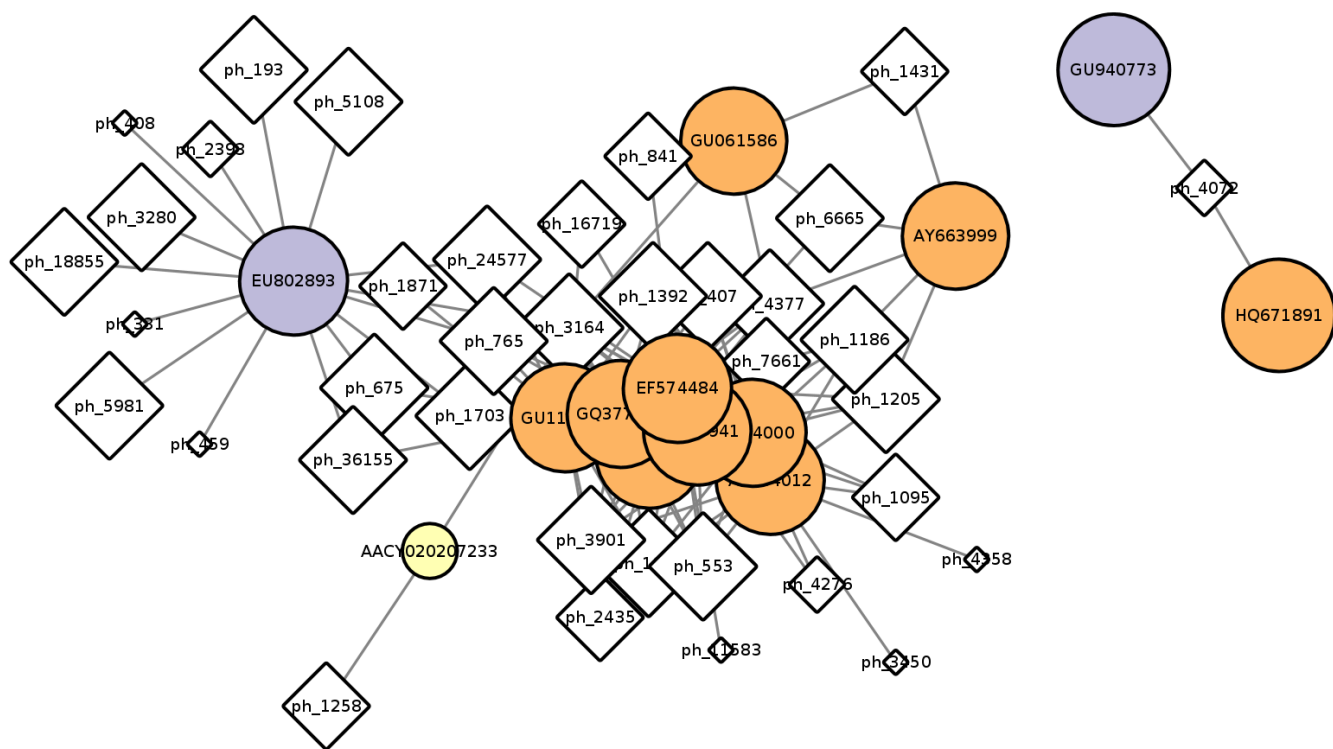
```
source("./functions_to_add_to_RCy3/subnetwork_stuff.R")
selectEdgesConnectedBySelectedNodes(cw2)
```

Now edges should appear selected in Cytoscape, in addition to the nodes.

```
newnet <- subnetwork_from_selected(cw2)
```

```
## [1] "Cytoscape window Tara oceans with degree(1) successfully connected to R session and graph copied"
```

```
layoutNetwork(newnet, "force-directed")
```



## 12.1 Conclusion

This has been a very basic introduction to exploring co-occurrence networks in Cytoscape using RCy3.

## References

- Brum, Jennifer R., J. Cesar Ignacio-Espinoza, Simon Roux, Guilhem Doucier, Silvia G. Acinas, Adriana Alberti, Samuel Chaffron, et al. 2015. “Patterns and Ecological Drivers of Ocean Viral Communities.” *Science* 348 (6237): 1261498. <http://www.sciencemag.org/content/348/6237/1261498.short>.
- Csardi, Gabor, and Tamas Nepusz. 2006. “The Igraph Software Package for Complex Network Research.” *InterJournal Complex Systems*: 1695. <http://igraph.org>.
- Gentleman, R., Elizabeth Whalen, W. Huber, and S. Falcon. 2016. *Graph: Graph: A Package to Handle Graph Data Structures*.
- Lima-Mendez, Gipsi, Karoline Faust, Nicolas Henry, Johan Decelle, Sébastien Colin, Fabrizio Carcillo, Samuel Chaffron, et al. 2015. “Determinants of Community Structure in the Global Plankton Interactome.” *Science* 348 (6237). doi:10.1126/science.1262073.
- Ono, K, T Muetze, G Kolishovski, P Shannon, and B Demchak. 2015. “CyREST: Turbocharging Cytoscape Access for External Tools via a RESTful API [Version 1; Referees: 2 Approved].” *F1000Research* 4 (478). doi:10.12688/f1000research.6767.1.
- Shannon, Paul T., Mark Grimes, Burak Kutlu, Jan J. Bot, and David J. Galas. 2013. “RCytoscape: Tools for Exploratory Network Analysis.” *BMC Bioinformatics* 14 (1): 1–9. doi:10.1186/1471-2105-14-217.
- Shannon, Paul, Andrew Markiel, Owen Ozier, Nitin S. Baliga, Jonathan T. Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. 2003. “Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks.” *Genome Research* 13 (11): 2498–2504. <http://genome.cshlp.org/content/13/11/2498.short>.
- Sunagawa, Shinichi, Luis Pedro Coelho, Samuel Chaffron, Jens Roat Kultima, Karine Labadie, Guillem Salazar, Bardya Djahanschiri, et al. 2015. “Structure and Function of the Global Ocean Microbiome.” *Science* 348 (6237): 1261359. <http://www.sciencemag.org/content/348/6237/1261359.short>.
- Venter, J. Craig, Karin Remington, John F. Heidelberg, Aaron L. Halpern, Doug Rusch, Jonathan A. Eisen, Dongying Wu, et al. 2004. “Environmental Genome Shotgun Sequencing of the Sargasso Sea.” *Science* 304 (5667): 66–74. doi:10.1126/science.1093857.
- Weiss, Sophie, Will Van Treuren, Catherine Lozupone, Karoline Faust, Jonathan Friedman, Ye Deng, Li Charlie Xia, et al. 2016. “Correlation Detection Strategies in Microbial Data Sets Vary Widely in Sensitivity and Precision.” *ISME J* 10 (7): 1669–81. <http://dx.doi.org/10.1038/ismej.2015.235>.