

Genome project tables in the genomes package

Chris Stubben

September 1, 2011

The number of genome sequencing projects submitted to public sequence databases is growing rapidly. In addition to the raw sequence data, the amount of associated metadata describing each project is also increasing. The **genomes** package collects genome project metadata from NCBI (<http://www.ncbi.nlm.nih.gov>) and provides tools to summarize, compare and plot the data in the R programming environment.

Genome tables are a defined class (*genomes*) and each table is a data frame where rows are genome projects and columns are the fields describing the associated metadata. At a minimum, the table should have a column listing the project name, status, and release date. A number of methods are available that operate on genome tables including **print**, **summary**, **plot** and **update**.

There are a number of ways to install this package. If you are running the most recent R version, you can use the **biocLite** command.

```
R> source("http://bioconductor.org/biocLite.R")
R> biocLite("genomes")
```

Since the format of online genome tables may change (and then **update** commands may fail), I would recommend downloading the development version for fixes in between the six month release cycle.

```
R> install.packages("genomes",
  repos="http://www.bioconductor.org/packages/devel/bioc", type="source")
```

Genome tables from the Genome Project database at NCBI include prokaryotic projects (**lproks**), eukaryotic projects (**leuks**), metagenomes (**lenvs**) and viruses (**virus**). The **print** methods displays the first few rows and columns of the table (either select less than seven rows or convert the object to a **data.frame** to print all columns). The **summary** function displays the download date, a count of projects by status, and a list of recent submissions. The **plot** method displays a cumulative plot of genomes by release date (Figure 1, use **lines** to add additional tables).

```
R> data(lproks)
R> lproks
```

A genomes data.frame with 6946 rows and 32 columns

	pid	name	status
1	33011	Abiotrophia defectiva ATCC 49176	Assembly
2	12997	Acaryochloris marina MBIC11017	Complete
3	16707	Acaryochloris sp. CCMEE 5410	Assembly
4	45843	Acetivibrio cellulolyticus CD2	Assembly
5	70153	Acetobacteraceae bacterium AT-5844	In Progress
...
6946	34927	Zymomonas mobilis subsp. pomaceae ATCC 29192	Complete
	released	...	
1	2009-03-17	...	
2	2007-10-16	...	
3	2011-06-03	...	
4	2010-08-11	...	
5	<NA>	...	
...	
6946	2011-06-17	...	

R> summary(lproks)

\$`Total genomes`

[1] 6946 genome projects on Aug 29, 2011

\$`By status`

	Total
In Progress	2904
Assembly	2324
Complete	1718

\$`Recent submissions`

RELEASED	NAME	STATUS
1 2011-08-26	halophilic archaeon DL31	Complete
2 2011-08-26	Helicobacter pylori Puno120	Complete
3 2011-08-26	Helicobacter pylori Puno135	Complete
4 2011-08-26	Helicobacter pylori Santal49	Complete
5 2011-08-25	Mycobacterium tuberculosis CTRI-2	Complete

R> plot(lproks, log = "y", las = 1)

R> data(leuks)

R> data(lenvs)

R> lines(leuks, col = "red")

R> lines(lenvs, col = "green3")

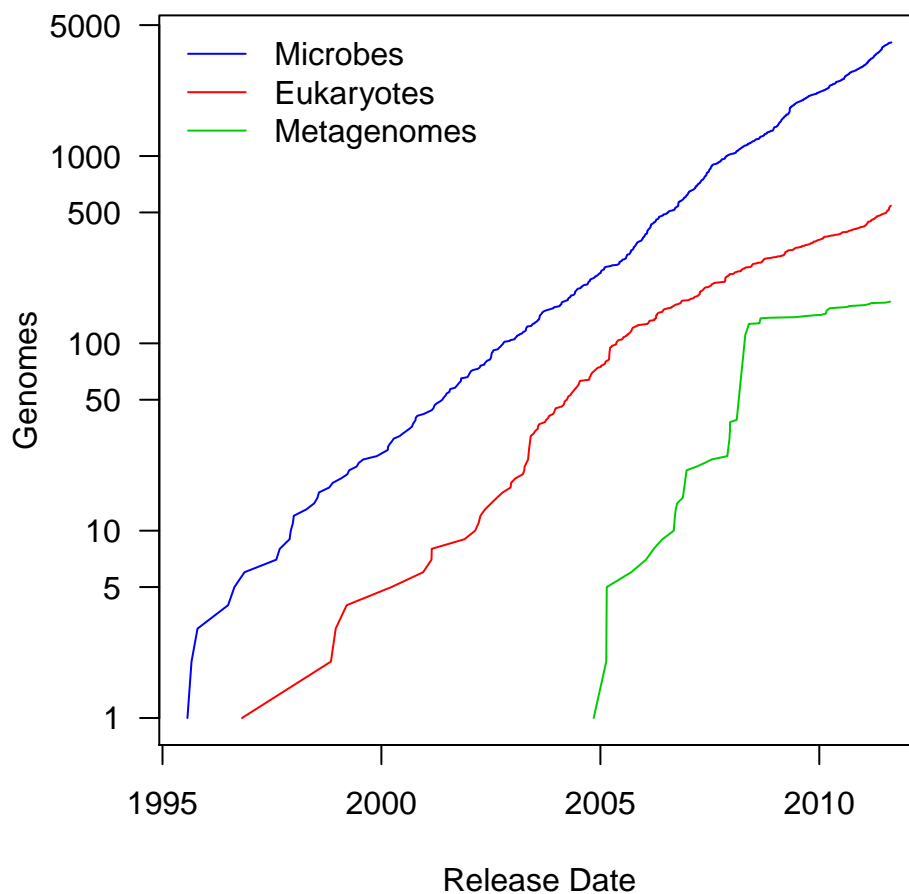


Figure 1: Cumulative plot of genome projects by release date at NCBI.

```
R> legend("topleft", c("Microbes", "Eukaryotes", "Metagenomes"),
      lty = 1, bty = "n", col = c("blue", "red", "green3"))
```

Most importantly, the `update` method downloads the latest version of the table from NCBI and displays a message listing the number of project IDs added and removed (not run).

```
R> update(lproks)
```

A number of additional functions assist in selecting, sorting and grouping genomes. The `species` and `genus` functions can be used to extract the species or genus from a scientific name. The `table2` function formats and sorts a contingency table by counts.

```
R> spp <- species(lproks$name)
R> table2(spp)
```

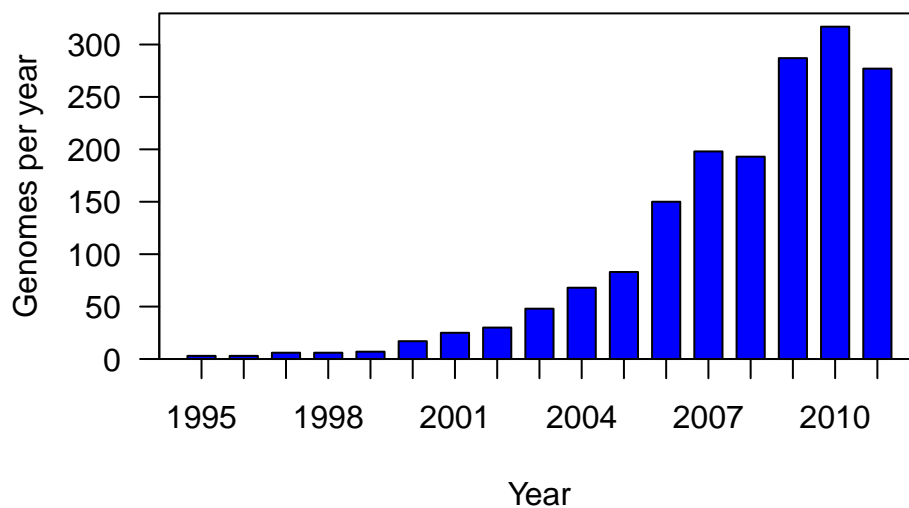


Figure 2: Number of complete microbial genomes released each year at NCBI

	Total
<i>Escherichia coli</i>	588
<i>Salmonella enterica</i>	217
<i>Staphylococcus aureus</i>	216
<i>Helicobacter pylori</i>	195
<i>Vibrio cholerae</i>	150
<i>Streptococcus mutans</i>	136
<i>Streptococcus pneumoniae</i>	102
<i>Yersinia pestis</i>	95
<i>Mycobacterium tuberculosis</i>	89
<i>Leptospira interrogans</i>	77

The `month` and `year` functions can be used to extract the month or year from the release date (Figure 2).

```
R> complete <- subset(lproks, status == "Complete")
R> x <- table(year(complete$released))
R> barplot(x, col = "blue", ylim = c(0, max(x) * 1.04), space = 0.5,
  las = 1, axis.lty = 1, xlab = "Year", ylab = "Genomes per year")
R> box()
```

Because subsets of tables are often needed, the binary operator `like` allows pattern matching using wildcards. The `plotby` function can then be used to plot the release dates by status using labeled points, in this case to identify complete and draft sequences of *Yersinia pestis* (Figure 3).

