Zac Lindsey

Dr. Botts

Honors 2018

30 March 2018

A novel tool for the identification of plasmid backbone gene labels

ABSTRACT

Plasmids carry antibiotic resistance genes which have a devastating impact on health and medicine. The plasmid backbone genes are also critical in understanding the effect plasmids have on a genome. NCBI stores plasmid backbone gene information in the nucleotide database, but the product names are often redundant, ambiguous, or incorrect. Incorrect naming makes assessment of the characteristics of the plasmid difficult. Untangling the mess of incorrect names is essential to the genomic analysis of plasmids. We present an interactive tool that assists in identifying and quantifying backbone gene annotation inconsistencies and provides a way to help fix the problem.

TABLE OF CONTENTS

INTRODUCTION

In 2013, the CDC reported that antibiotic resistant bacterial infections are the cause of approximately 23,000 deaths each year in the United States. The economic impact has been estimated to be up to multiple billions of dollars per year (McGowan). Antibiotics have become less effective due to antibiotic resistance genes. One of the most important mechanisms in the rise and spread of antibiotic resistance genes between bacteria are plasmids. Plasmids are circular, self-replicable mobile genetic elements that are often found within bacteria. A mobile genetic element is a type of genetic element (such as DNA) that can easily move around within a genome. The study of plasmids is a crucial piece in the puzzle of understanding the transmission of antibiotic resistance genes.

In addition to the antibiotic resistance genes, plasmids carry other types of genes. For example, backbone genes are vital to the replication and transmission of plasmids. Genes are sequences of DNA which encode for specific proteins. Proteins are the molecules that are used as the building blocks of structures inside of cells. Therefore, the function of a cell is determined by which genes are present. Plasmids have the ability to introduce new genes into the genome, making them critical to the overall function of cells, and significant in the rise of novel resistance phenotypes.

PROBLEM STATEMENT

The process of identifying the function of each gene on a plasmid is referred to as *plasmid gene annotation*. To assist in annotation, many plasmid gene sequences are stored in online databases such as the Nucleotide database at the National Center for Biotechnology Information (NCBI). In order to easily annotate genes, the NCBI protein product label should be consistent across

identical genes. Consistent labels allow one to accurately infer the role of the gene in plasmid function. But this is not the case; in fact, many products are labeled ambiguously (Thomas et al. 62). For example, many products are simply labeled as "hypothetical protein" rather than being given a name that suits its function. Although some of these truly are unknown, many are actually known, but have not been updated in the database. The problem of inconsistent product names within the NCBI Nucleotide database must therefore be quantified in order to prevent further hindrance of study, and new consistent naming conventions should be applied, such as those suggested in "Annotation of plasmid genes."

Thesis

This paper describes a novel tool that assists in identifying and quantifying backbone gene annotation inconsistencies within the NCBI database, and provides a way to help fix the problem. Two primary types of inconsistencies will be studied: (1) Sequences with inferred functional similarity based on sequence identity, but with different product names and (2) Sequences with inferred functional diversity, but the same product name.

Criteria for similar sequences

A pairwise alignment is a way to match two sequences in order to find similarities and differences. There are two properties of alignments that collectively measure similarity between the protein sequences: coverage and identity. Target coverage refers to the percentage of the query sequence covered by the alignment sequence. Within the covered portion, the sequences have identity to the extent that their letters match. For example, "AAGG" and "AAGGTT" have 100% identity within the covered portion, which is composed of the first four characters. "AAGC" would have 75% identity within the covered portion because of differing character "C."

We assume that sequences with a certain fraction of target coverage and identity encode the same protein and therefore have a similar function. Similar sequences within the NCBI database will often have different product names. Frequently, two proteins that do not have the required fraction of identity and coverage will be placed in different groups. This could imply that they have different functions. However, their labels are often the same.

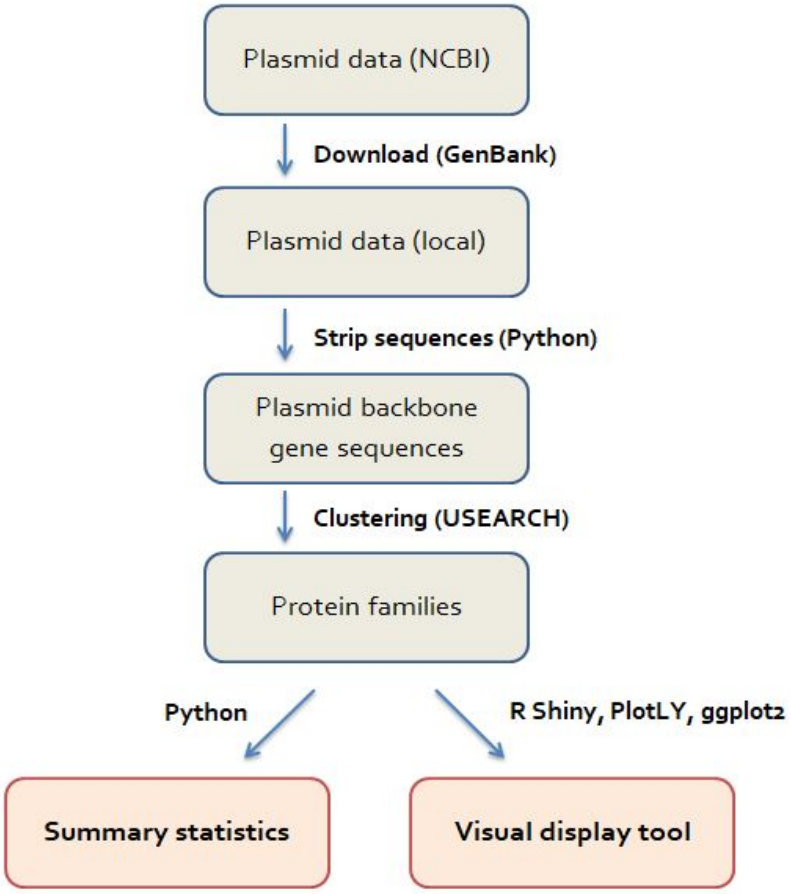
An interactive tool

In order to help solve the problem of inconsistent product labels, it would be useful to have (1) an interactive tool that displays the problem using multiple visualizations and (2) summary statistics which estimate the number of inconsistencies in the database.

METHODS

In order to display the problem using multiple visualizations, we need to obtain the protein sequences and organize similar proteins into groups. We will use clustering to accomplish this task. The visualization will be accomplished using R, and the summary table will be created in Python. For a summary flow chart of the process, see Flow Map 1 (next page).

Flow Map 1



Obtaining backbone genes

The first step in building the interactive tool was to collect the plasmid backbone gene sequences from the NCBI GenBank nucleotide database. The search term used on January 22, 2018 was "plasmid complete." In addition, filters were selected to only find sequences which are (1) between 20 and 200 kilo-base pairs in length and (2) located on plasmids only. We selected plasmids between 20 and 200 kilo-base pairs in order to avoid engineered cloning vectors labeled as complete plasmids, and to avoid large non-mobile plasmids that act as second chromosomes. The results were downloaded as a full GenBank file.

Initially, the plasmid backbone genes found within the GenBank file were not organized in any meaningful way. We used Python scripts to reorganize the data. See Table 2 for a step-by-step breakdown of how the clusters were obtained. The scripts are available for download in the Supplementary Materials.

Creating protein families

Having obtained the plasmid backbone genes, the next step was to put the genes into similar groups. If a group of genes are similar in sequence identity and coverage, we can infer their function to be similar. If their function is the same they should have the same product name, but this is often not true of GenBank protein products. This problem is clearly seen by examining the product names of genes that were placed in the same *cluster*.

The process of placing similar data into groups is known as *clustering*. A cluster of similar genes will be referred to as a *protein family*. We chose the UCLUST algorithm for its speed and ease of use. Additionally, UCLUST allows the user to define what percentage of sequence identity and target coverage should be used as the criteria for making a cluster. Each cluster will have a *centroid sequence*, which is a gene sequence that serves as the representative for a cluster. Any gene that has the minimum identity and target coverage with a centroid will be placed in the centroid's protein family; otherwise, it will become the centroid of a new protein family.

UCLUST works in a linear fashion, starting with the first input sequences and ending with the last (Edgar 2010). The first input genes are more likely to become centroids. For this reason, we selected genes from a list of well-studied plasmids found in "Annotation of plasmid genes" to be the first inputs into UCLUST. It is important to note that any two or more sequences

in a protein family do not necessarily have the minimum criteria of similarity with each other. Rather, they are in the same protein family because they both have similarity with a protein family's centroid.

We also want to have an idea of what the correct label for a particular protein family should be. Protein families were placed into broader categories based on a list of proposed four-letter backbone labels in "Annotation of plasmid genes." Families associated with one of the proposed labels were placed in the label's category. A family is associated with a proposed label when the proposed label can be found in at least one of the product names within that family. For example, a protein called "ParAprotein" along with its entire protein family would be placed within the "ParA" category. This means that proteins which may be similar to "ParAprotein" will be associated with the "ParA" name. Although "ParAprotein" and "ParA" are very similar labels, it would be better to follow a standardized naming convention.

Protein families with well-studied plasmids

We are interested in creating protein families because we would like to identify inconsistencies in labeling and work toward applying consistent names. In order to find inconsistent labels, we allow the user to search for protein families in which the function is known with reasonable confidence based on the presence of a well-studied plasmid. The well-studied plasmids described in "Annotation of plasmid genes" have been characterized consistently and their functions are well-known. The protein families containing well-studied plasmids can be analyzed with reasonable confidence, so it may be possible to assign a name in these cases based on the suggested scheme in "Annotation of plasmid genes."

The protein families were divided into two groups based on the presence of well-studied plasmids. The first group (Group 1) consists of only the protein families containing at least one protein encoded in a well-studied plasmid found in "Annotation of plasmid genes" (Thomas et al.). The second group (Group 2) contains all protein families generated from the NCBI database. The second group helps the user to identify protein families which may not use standardized naming because it does not contain well-studied plasmids. The interactive tool has functionality that allows a user to choose which group to view. The difference between Group 1 and Group 2 shows that many proteins grouped with a particular backbone gene by name may not be functionally related, which demonstrates the problem of ambiguous naming.

Computing a summary table

In addition to the tool, a summary table showing statistical information is available in Table 1a, 1b and 1c. Table 1a shows Group 1, 1b shows Group 2, and 1c shows the difference between the two groups. For each group, the table shows the number of clusters, the number of genes, the average number of genes per cluster, the mean sequence length of all genes, and the standard deviation of sequence length.

Graphs and tools

We want to look at each cluster and examine the product names on each gene. This allows us to see the problem of similar genes having different product names. UCLUST outputs a tabbed file with the cluster information. It is easier to visualize the relationships using a bar graph, which can show each gene next to each other at a glance. The open statistical programming language R was chosen for the task of data visualization. The data was formatted for R using Python scripts. The scripts are available in the Supplementary Materials.

A user would also like to see which backbone proteins are shared by each plasmid. The tool gives users the ability to look at the genomic context of plasmids carrying genes using the selected name. We created a summary matrix which shows the presence of a potential backbone protein for each plasmid in the database. The R heatmap allows the user to see which of the currently selected plasmids may have the currently selected protein. In addition, multiple sequence alignments were computed on each protein family using the MUSCLE package. Finally, the alignments were input into the ape package to compute phylogenetic trees for each cluster.

Visual display and ease of use

In order to make the graphs and tools viewable online without having to download or install packages, three R packages were used in conjunction - Shiny, PlotLY, and ggplot2.

RESULTS

The main section of the resulting interactive tool allows the user to select a reference protein of interest. Under this selection box are five tabs containing each different part of the tool. The tool can be found at the following address: zaclindsey.shinyapps.io/plasmidbackbone2/ . See Table 1 for the summary table.

Selecting a protein of interest

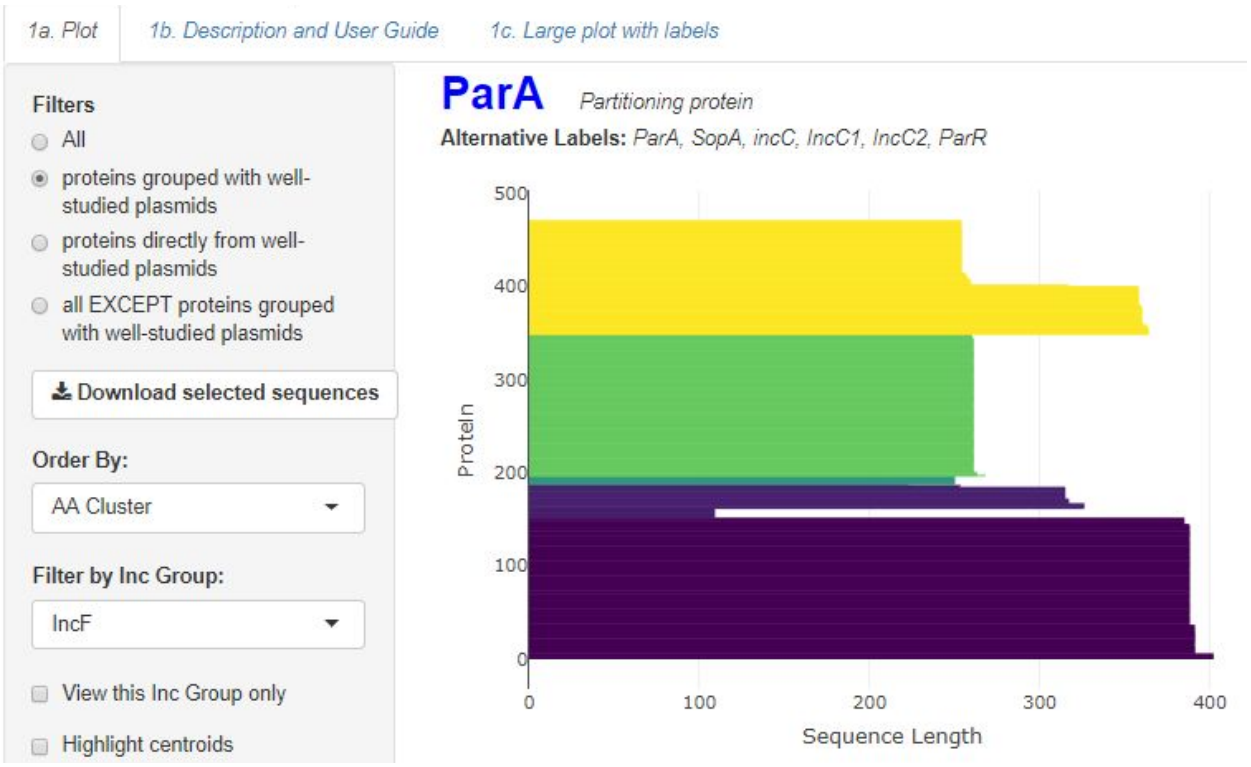
Each four-letter label comes from a table of proposed backbone gene names from “Annotation of plasmid genes.” The user can select any one of these four-letter labels from the list. For example, choosing “ParA” will bring up all of the relevant data for the partitioning protein. Each alternative label for this proposed label is listed above the bar graph. According to “Annotation

of plasmid genes," the ParA protein is associated with the following labels: ParA, SopA, incC, IncC1, IncC2, and ParR. Therefore, selecting ParA will display those labels above the bar graph for ParA.

Tab 1a: Interactive bar graph

Selecting a 4-letter label will display a bar graph (Tab 1a) of the lengths of proteins associated with this label. The goal was to find genes associated with a particular backbone gene name and examine their product names. This graph shows multiple distinct protein families in which at least one member had the selected label contained within the product name. Each protein family is represented by a different color. Each bar on the graph is a different protein. Hovering over a bar will display information, including: the NCBI product name, the incompatibility group, and the plasmid on which the protein is encoded. On the left-hand side, the control panel gives options to filter out large amounts of proteins based on user-selected parameters. The control panel also lets the user choose to download a FASTA file containing the amino acid sequences of all of the displayed sequences. This file can be used for reannotation or to establish a new database.

Tab 1a



The user can see six distinct protein families associated with "ParA." Hovering over the yellow family shows each product label associated with this family. The most commonly used product label for this family is "IncC." A few instances of "partitioning protein" and "hypothetical protein" appear, but the "ParA" name is used less than 10 times.

Tab 1c: Large plot with labels

The large plot found in Tab 1c shows the same information as the interactive plot, but expanded so that all of the product names can be viewed at a glance.

Tab 2a: Heat maps

The heat map gives the user a snapshot of the overall genome for each plasmid. If at least one four-letter label is associated with the plasmid, the map will highlight the protein-plasmid pair in

yellow. Each plasmid is displayed on the y-axis. Each four-letter label is displayed on the x-axis. This also allows users to identify which portions of the plasmid backbone are shared by the other plasmids in the genome.

Tab 2a

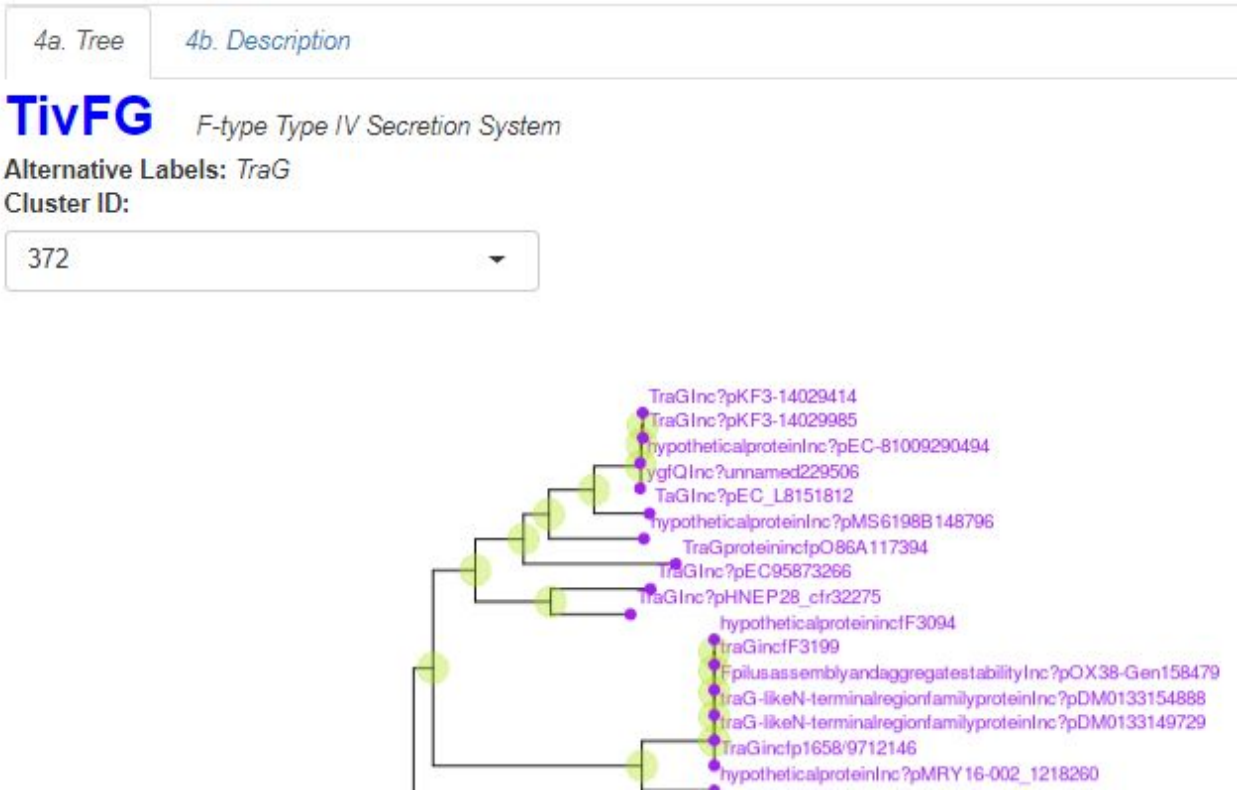
Tab 3: Multiple sequence alignments

Multiple sequence alignments for each protein family were computed using MUSCLE. These can be downloaded in FASTA format and allow the user to view potential evolutionary relationships for each protein, as well as the potential labeling inconsistencies.

Tab 4a: Phylogenetic trees

Phylogenetic trees allow the user to visualize and infer evolutionary relationships and homology between protein families. These trees are available in Tab 4a.

Tab 4a



Examples of problems

In the summary table, the protein Sfx has the lowest difference between the number of protein families between Group 1 and Group 2. Only one family is associated with Sfx in Group 1 but six with Group 2. This means there may be up to five protein families which had names similar to "Sfx," but may not be functionally similar to Sfx. In one of the worst cases, the partitioning protein ParA has a difference of 163 protein families between Group 1 and Group 2. Hovering over the proteins in the interactive tool reveals that many names associated with ParA have nothing to do with the keyword "ParA." Some of the names include "Resolvase," "ParD," "MinD," and "Soj," none of which are recommended in "Annotation of plasmid genes."

CONCLUSION

The original purpose of the tool was to identify and quantify inconsistencies in product labels within the NCBI Database. The summary statistics table quantifies the potential problems with labeling proteins. The interactive tool sufficiently allows the user to view multiple distinct groups of plasmid backbone proteins and their NCBI product labels.

Suggestions for use

The tool also provides information that can assist biologists moving forward in labeling plasmid backbone proteins. For example, a user may be interested in having a reference point for the labeling of Type IV secretion system proteins. In order to accomplish this, one might select a Type IV secretion system protein from the drop-down list, possibly "TivFW." Within TivFW, there is only one protein family associated with plasmid "F," so this family will show up on the interactive bar graph upon selection. Upon inspecting the various product names, the user may find this protein family to be a reliable reference point for the future annotation of TivFW proteins sequences. Clicking "download selected sequences" will result in a FASTA file containing each protein from this family. This FASTA file could be used as a curated database for annotation of plasmid genomes using tools such as Prokka.

Experts may also be interested in creating a new naming convention for plasmid backbone genes. This tool provides a way to view thousands of grouped proteins as well as suggested labels for each based on the recommended names in "Annotation of plasmid genes." The user can see potential labeling inconsistencies by selecting the filter labeled "all EXCEPT proteins grouped with well-studied plasmids." Using the tool and the data from the summary table, one may observe that there are potentially up to 2704 mislabeled products for "ParA."

Table 1a - Group 1

Label	# Clusters	# Genes	# Mean genes per Cluster	Mean SeqLength	SD of SeqLength
Cpl	12	640	53.33	472.42	244.3
DsbC	4	331	82.75	244.15	58.15
Dtr	11	528	48	221.7	142.4
Eex	2	73	36.5	107.96	46.34
MpfPL-O	3	118	39.33	340.87	179.28
Nac	2	185	92.5	247.99	20.05
ParA	6	470	78.33	310.93	67.44
ParB	13	2666	205.08	322.27	236.69
ParC	3	248	82.67	240.06	161.3
Pep	6	402	67	272.2	52.01
Pri	9	1012	112.44	688.96	335.08
Rep	24	8869	369.54	236.75	209.69
Rlx	7	359	51.29	910.6	666.04
Sfx	1	117	117	247.13	8.43
Slt	1	33	33	215.3	12.88
Ssb	9	1937	215.22	159.74	53.93
TivB10	3	353	117.67	365.42	159.83
TivB11	2	157	78.5	227.55	65.96
TivB2	4	428	107	154.92	45.31
TivB3	6	254	42.33	131.26	63.54
TivB4	14	1194	85.29	614.7	378.97
TivB5	7	528	75.43	280	172.63
TivB6	1	44	44	571.61	31.43

TivB7	3	85	28.33	182.64	35.01
TivB8	3	152	50.67	200.64	67.22
TivB9	6	314	52.33	203.43	95.96
TivFA	2	248	124	124.6	28.64
TivFC	2	180	90	196.69	27.18
TivFF	3	243	81	241.63	42.8
TivFG	4	125	31.25	763.14	188.62
TivFH	3	171	57	388.25	140.02
TivFI	3	353	117.67	365.42	159.83
TivFN	3	135	45	438.15	180.95
TivFU	1	157	157	330.28	4.47
TivFW	1	195	195	213.23	11.66

Table 1b - Group 2

Label	# Clusters	# Genes	# Mean genes per Cluster	Mean SeqLength	SD of SeqLength
Cpl	114	1978	17.35	469.51	285.65
DsbC	28	839	29.96	271.26	90.23
Dtr	110	2435	22.14	298.74	233.68
Eex	38	913	24.03	117.38	53.73
MpfPL-O	28	593	21.18	235.31	155.59
Nac	19	485	25.53	224.13	33.85
ParA	169	3174	18.78	287.3	82.13
ParB	185	7048	38.1	307.01	194.7
ParC	41	1492	36.39	216.99	164.37

Pep	78	2350	30.13	207.62	104.04
Pri	120	2461	20.51	615.05	407.53
Rep	459	27113	59.07	205	175.94
Rlx	69	1279	18.54	696.1	493.2
Sfx	6	246	41	246.59	23.83
Slt	19	437	23	313.66	97.75
Ssb	43	2560	59.53	193.86	148.33
TivB10	44	667	15.16	379.69	124.62
TivB11	20	585	29.25	309.91	72.62
TivB2	59	1138	19.29	248.43	257.07
TivB3	69	1175	17.03	296.79	317.52
TivB4	100	2661	26.61	593.75	408.1
TivB5	46	1324	28.78	308.56	203.05
TivB6	12	181	15.08	419.25	98.87
TivB7	25	435	17.4	374.68	350.88
TivB8	21	445	21.19	212.67	54.79
TivB9	48	830	17.29	217.69	151.3
TivFA	33	509	15.42	149.88	183.96
TivFC	21	602	28.67	337.92	281.51
TivFF	25	615	24.6	262.75	89.27
TivFG	49	900	18.37	598.56	302.51
TivFH	29	591	20.38	257.65	143.45
TivFI	40	606	15.15	379.61	130.71
TivFN	27	441	16.33	459.44	196.72
TivFU	18	465	25.83	544.06	336.8

TivFW	18	453	25.17	308.07	94.23

Table 1c - Difference between Group 1 and Group 2

Label	# Clusters	# Genes	# Mean genes per Cluster	Mean SeqLength	SD of SeqLength
Cpl	102	1338	-35.98	-2.91	41.35
DsbC	24	508	-52.79	27.11	32.08
Dtr	99	1907	-25.86	77.04	91.28
Eex	36	840	-12.47	9.42	7.39
MpfPL-O	25	475	-18.15	-105.56	-23.69
Nac	17	300	-66.97	-23.86	13.8
ParA	163	2704	-59.55	-23.63	14.69
ParB	172	4382	-166.98	-15.26	-41.99
ParC	38	1244	-46.28	-23.07	3.07
Pep	72	1948	-36.87	-64.58	52.03
Pri	111	1449	-91.93	-73.91	72.45
Rep	435	18244	-310.47	-31.75	-33.75
Rlx	62	920	-32.75	-214.5	-172.84
Sfx	5	129	-76	-0.54	15.4
Slt	18	404	-10	98.36	84.87
Ssb	34	623	-155.69	34.12	94.4
TivB10	41	314	-102.51	14.27	-35.21
TivB11	18	428	-49.25	82.36	6.66
TivB2	55	710	-87.71	93.51	211.76
TivB3	63	921	-25.3	165.53	253.98
TivB4	86	1467	-58.68	-20.95	29.13
TivB5	39	796	-46.65	28.56	30.42
TivB6	11	137	-28.92	-152.36	67.44
TivB7	22	350	-10.93	192.04	315.87
TivB8	18	293	-29.48	12.03	-12.43
TivB9	42	516	-35.04	14.26	55.34
TivFA	31	261	-108.58	25.28	155.32
TivFC	19	422	-61.33	141.23	254.33
TivFF	22	372	-56.4	21.12	46.47
TivFG	45	775	-12.88	-164.58	113.89
TivFH	26	420	-36.62	-130.6	3.43
TivFI	37	253	-102.52	14.19	-29.12

TivFN	24	306	-28.67	21.29	15.77
TivFU	17	308	-131.17	213.78	332.33
TivFW	17	258	-169.83	94.84	82.57

Table 2

Step	**Tool**	**Output**
1) Obtain Gene Data	**NCBI Database**	**Plasmids.gb**
2) Strip Sequences	**Python Code**	**Plasmids.csv, Seqs.faa**
3) Cluster Sequences	**USEARCH**	**Clusters.uc (tab)**
4) Reorganize Data for R	**Python Code**	**Clusters.csv**
5) Display Data	**R Code & Packages**	**HTML Display**

BIBLIOGRAPHY

Edgar, RC, 2004. MUSCLE: multiple sequence alignment with high accuracy and high throughput Nucleic Acids Res. 32(5):1792-1797

Edgar, RC, 2010. Search and clustering orders of magnitude faster than BLAST, Bioinformatics 26(19), 2460-2461.

McGowan, J. E., 2001. Economic Impact of Antimicrobial Resistance. *Emerging Infectious Diseases*, *7*(2): 286-292. Web. 30 March 2018.

Suzuki H., Yano H., Brown C. J., Top E. M., 2010. Predicting plasmid promiscuity based on genomic signature. J. Bacteriol. 192: 6045–6055.

Thomas, Christopher M., Nicholas R. Thomson, Ana M. Cerdeno-Tarraga, Celeste J. Brown, Eva M. Top, and Laura S. Frost. "Annotation of Plasmid Genes." Plasmid 91 (2017): 61-67. Web. 28 June 2017.

SUPPLEMENTARY MATERIALS

Links

Interactive tool	zaclindsey.shinyapps.io/plasmidbackbone2/
Project source code	https://github.com/rbotts/ProteinNaming
R Package: "ape"	https://cran.r-project.org/web/packages/ape/index.html
R Package: "MUSCLE"	http://www.bioconductor.org/packages/release/bioc/html/muscle.html