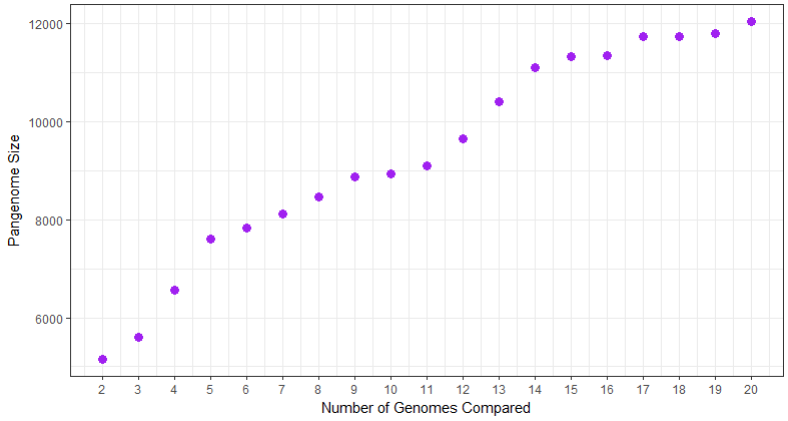
Last Week

- Reviewed Python and *attempted* to learn Java (key word attempt). Completed the "Learn Java" and "Python" courses on Code Academy.
- Completed various error tests for Roary with the *E. coli* pangenome:
  - Ran Roary for all 102 genomes from the tree just for the sake of it (testing extremities) - resulted in "Total genes" value of 18695:

Genome Designation	Percentage Present Within Strains	# of Genes
Core genes	(99% <= strains <= 100%)	2692
Soft core genes	(95% <= strains < 99%)	349
Shell genes	(15% <= strains < 95%)	2076
Cloud genes	(0% <= strains < 15%)	13578
Total genes	(0% <= strains <= 100%)	18695

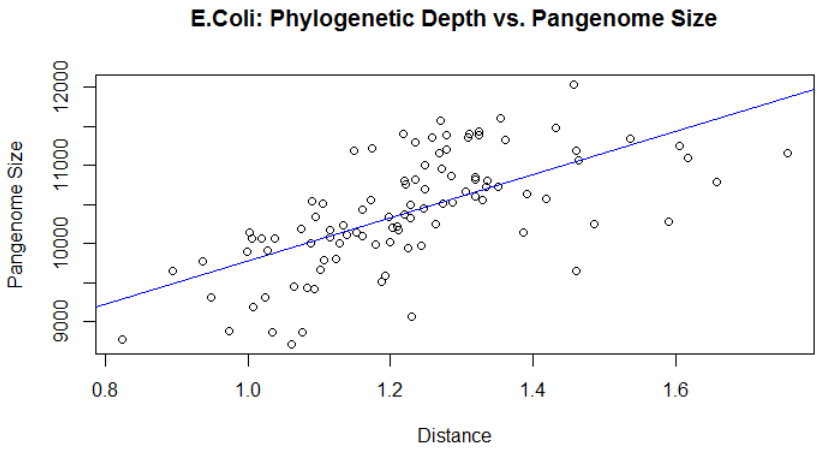
Table 1: Direct output from running Roary on all 102 *E. coli* samples

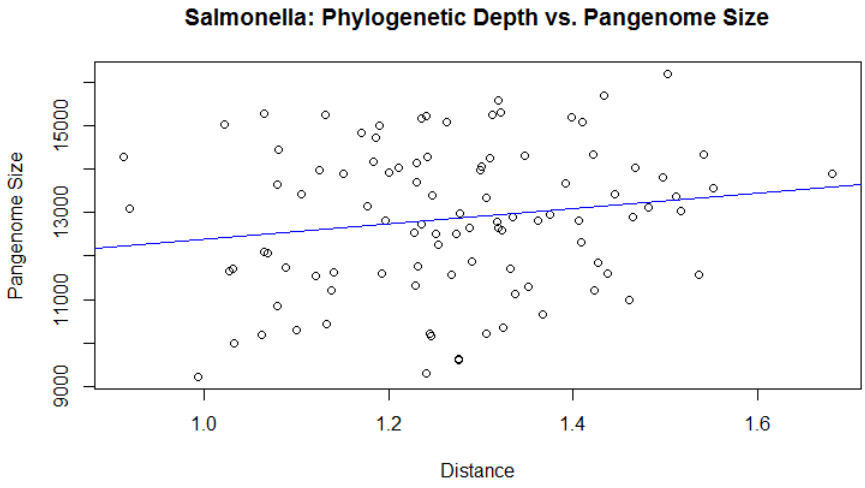
  - Ran Roary with two samples of *E. coli*, then three, then four...and so forth (Is there a proper name for this technique?) The results are as followed, and seem to be reasonable, steady rise in pangenome size:



  - Ran Roary with 19 genome samples rather than 20 - similar resuls for the most part (still need to do this for 5, 10,..etc. genomes to prove that selecting 20 is truly arbitrary.)

- Integrated R plots into LaTeX script through R-Sweave
- Played around with ggplot2 to make relevant plots
- Completed linear regression tests for *E. coli* and *Salmonella* data for phylogenetic distances in correlation to pangenome sizes. Results were decent for *E. coli*, but much less so for *Salmonella* (though the Salmonella data hasn't been as thoroughly checked for error or annotation)





According to R's significant codes, the data for *E. coli* Depth is extremely significant (***), whereas *Salmonella*'s is not significant at all (' ') - more on this in the "this week" section.

Table 2: Linear Regression Test for *E. coli*

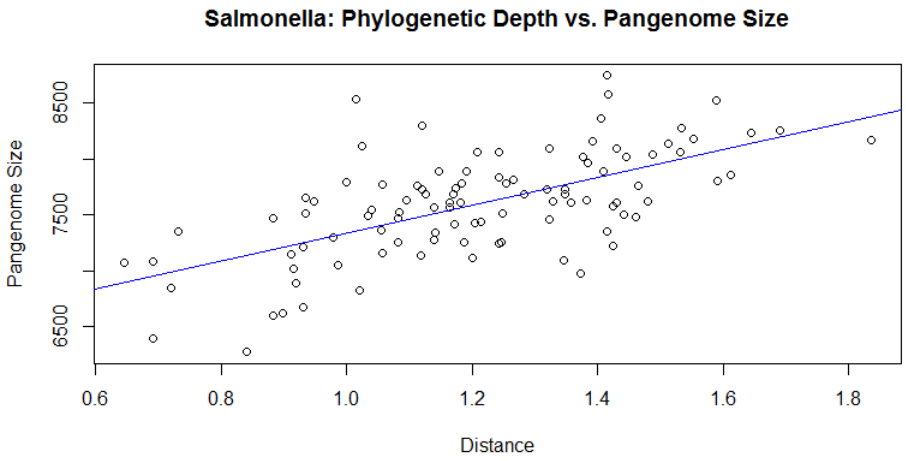
Min	1Q	Median	3Q	Max
-1402.98	-320.12	16.56	360.71	1050.86
Coefficient	Estimate	Std. Error	t-value	Pr(>|t|)
(Intercept)	7015.5	412.7	16.997	$< 2 \times 10^{-16}$
Depth	2759.8	333.6	8.274	6.58×10^{-13}
Res. Std. Error	Mult. R^2	Adj. R^2	F-statistic	p-value
559.6 (98 DF)	0.4113	0.4053	68.46(1 & 98 DF)	6.582×10^{-13}

This Week

- Played with HTML & CSS on Code Academy
- As suspected, the salmonella data was in fact not the complete genome, but just assemblies, so an alternative method of download was done, as instructed by https://www.ncbi.nlm.nih.gov/genome/doc/ftpfaq/#allcomplete.
- Downloaded from RefSeq database through NCBI's "Assembly" database
- Prokka files and Roary data regenerated with new data.
- The new salmonella data is as followed:

Table 3: Linear Regression Test for *Salmonella*

Min	1Q	Median	3Q	Max
-867.44	-248.82	34.71	208.10	1180.96
Coefficient	Estimate	Std. Error	t-value	Pr(>|t|)
(Intercept)	6083.9	190.8	31.887	2×10^{-16}
Depth	1252.6	154.9	8.088	1.65×10^{-12}
Res. Std. Error	Mult. R^2	Adj. R^2	F-statistic	p-value
369.3 (98 DF)	0.4003	0.3942	65.41(1 & 98 DF)	1.65×10^{-12}



- **Multiple FASTA titles within genome files** Though the downloads were coconfirmed to be the full genome from NCBI, it seems that it also comes with full plasmid sequences in addition to the original complete genome...and it seems that parsnp does not really work well with these.

  Attempts:

  - **Running Raw, Downloaded FNA Files Without Adjusting Anything:** This worked well for the case of *salmonella*, where I fed parsnp around 200 fna files, and ended up with a tree with around 100 branches or so. The data showed a similar correlation to the *E. coli* data, and was reported to be significant by R. However, when I tried the same things for bacteria that I had a smaller pool of download genomes, such as *S. aureus*, I ended up with a tree with very few branches. Naturally, I suspected this had something to do with the way the files were annotated so I made some adjustments
  - **Running Files Without Multiple Titles:** I extracted the files that had just one '>' title and ran those alone with parsnp. The tree was nearly all-inclusive, cutting out very few branches and provided useful data. However, I don't know if excluding the multiple title alignments would cause any bias in the data.
  - **Running Files With 'Complete Chromosome' Alignment as the First Read:** This was part of the process of me trying to

> extract/remove the plasmid data so that we would only be looking the the core, complete genome, however I decided to try and run parsnp with the files where the complete genome was listed first ONLY (i.e. I only ran parsnp with files that had > blah blah blah, complete genome on the first line). Surprisingly, this also gave a rather usable tree.

I have yet to try it with the complete genome data along (i.e. I have not yet tried ot actually cut out the plasmids entirely), but it seems like the result would definitely work. Right now I'm just worried about the consistency of everything, and if me selecting according to annotation will have any effect on the final data.

So in order to find out whether this does have any change, I will attempt to run Roary on the separate categories to see if there is any real, significant change in output data.

It doesn't seem either Prokka or Roary has the same issue (as the GFF files seem to be properly annotated and Roary displays similar input that is consistent to the *E. coli* data), so obtaining a proper tree seems to be the principal problem at this point

Tentatively, the plasmids will be cut out, and only the "complete genome" sequences will be looked at, despite that we know the plasmids could be a vital aspect of lateral gene transfer and may have to be addressed later.

Next Week

- Finish analyzing genome/plasmic data
- Begin looking at method to set up lateral gene transfer models so we can look at the bigger picture
- Get more samples (of which bacteria? which source?). Most likely to continue downloading NCBI genomes from Assembly
- Find the proper statistical tests for these because it kind of seems like I'm not just prodding in air as much):
- Continue separating the plasmid sequences *for now*
- Run more error tests on Roary, although it seems pretty(?) consistent at this point
- Write! Write! Write!