Stats744: HW6

Tell a Story With Graphs

Data: Thesis Again. . .

As part of my thesis I have information about substitutions and their variation across 6 *E. coli* genomes. The data is binary in nature: determining at each site in the genome if there is a substitution (1), or there is not (0). This was also broken down into protein coding and non-protein coding sections of the genome. We expect that non-protein coding segments of the genome to be undergoing more substitutions because they are typically "less important" than protein-coding segments of the genome.

In bacteria, genomic reorganization such as rearrangements, inversions, and duplications happen often. This means that genes change position in bacterial genomes often. Therefore, it is possible for one gene to be present at the beginning of the genome in one bacterial strain, and the same gene could be found towards the end of the genome in another bacteria.

My data takes this genome reorganization into account by adding a phylogenetic component to this analysis. I reconstructed both the sequence and the genomic position of that sequence in the ancestors of the 6 *E. coli* genomes. This allows for a gene, or more generally a segment of sequence to be found in various genomic positions between the different strains of *E. coli*. This also means that for a particular site in the genome, I have 11 data points, one for each of the nodes in the phylogenetic tree of these bacteria. a.k.a millions of data points.

Again, this takes time to read in and to run on my laptop so I have condensed the data into total number of substitutions over 10,000bp segments of the genome. This binned data can be found here.

```r
chrom_datafile <- "binned_dat.csv"
chrom_data <- read.csv(chrom_datafile)
chrom_data <- chrom_data[, -1]
colnames(chrom_data) <- c("position", "cod_sub", "ncod_sub", "cod_tot",
    "ncod_tot", "cod_weight", "ncod_weight")
# subset data
chrom_data_raw <- chrom_data[, c("position", "cod_sub", "ncod_sub")]

# tidy df
chrom_data_raw <- chrom_data_raw %>% gather(genom_cat, val, cod_sub:ncod_sub) %>%
    mutate(genom_cat = gsub("genom_cat", "", genom_cat)) %>% arrange(position,
    genom_cat)

# scale data to look better on graph
chrom_data_raw$position <- chrom_data_raw$position/10000
chrom_data_raw$val <- chrom_data_raw$val/1000
# change names so direct labels will be nicer
```

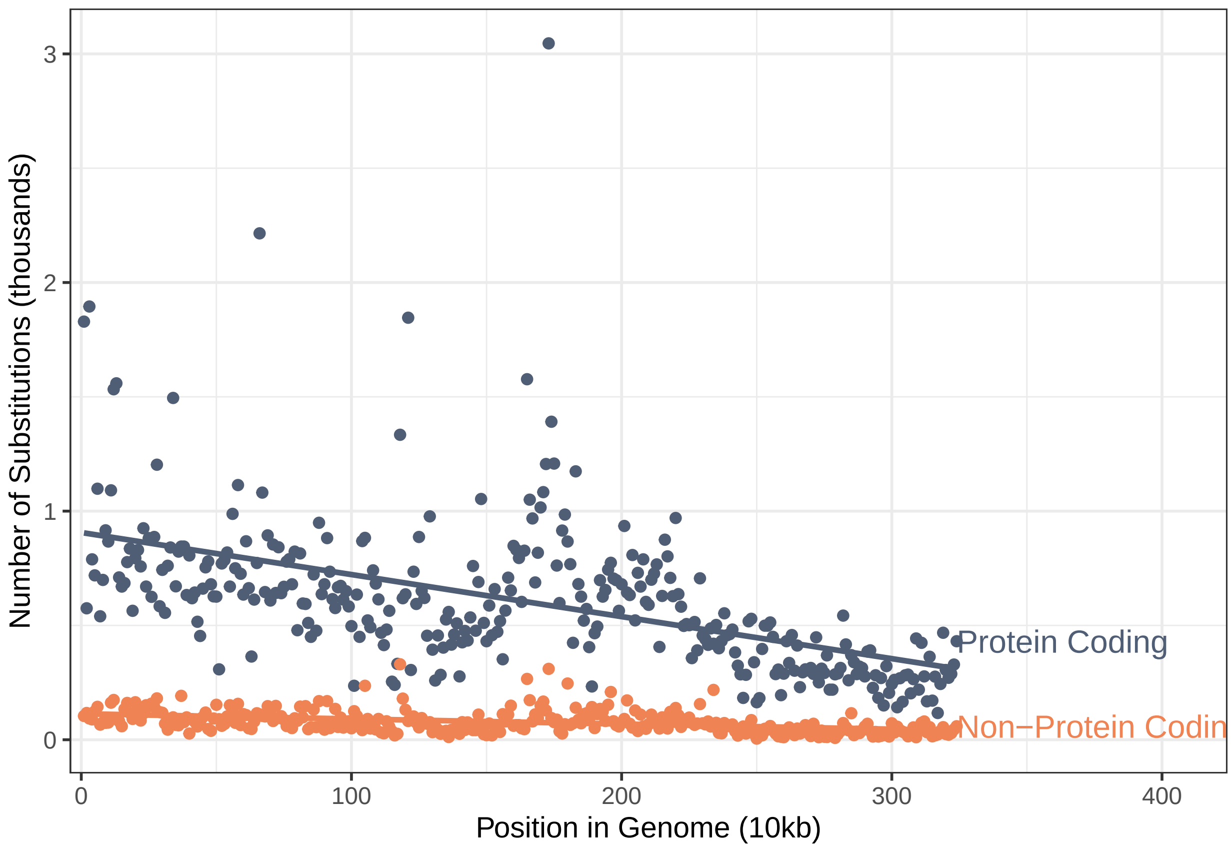
```
chrom_data_raw <- chrom_data_raw %>% mutate_if(is.character, str_replace_all,
    pattern = "^cod_sub", replacement = "Protein Coding")
chrom_data_raw <- chrom_data_raw %>% mutate_if(is.character, str_replace_all,
    pattern = "^ncod_sub", replacement = "Non-Protein Coding")
# is there a better way to do the above?

# defining colours
cols <- c("#EF8354", "#4F5D75")
```

Graph 1

```
#base graph
base_g <- ggplot(data=chrom_data_raw, aes(x=position, y=val, color= genom_cat))

rg <- (base_g
       + geom_point()
       + geom_smooth(method = 'lm', se = FALSE)
       + theme_bw()
       + theme(legend.position="top")
       + labs(x = "Position in Genome (10kb)", y = "Number of Substitutions (thousands)"
       + scale_color_manual(values = cols)
       #expand axis so direct labels can be seen
       + scale_x_continuous(expand = c(0.01,0),limits=c(0,420))
)
direct.label(rg,method="last.points")
```



Discription of Graph 1:

In this graphic my main goal was to show the difference in distribution of protein coding substitutions and non-protein coding substitutions. The above graph is looking at the "raw" total counts of substitutions in each 10,000 bp (or 10kb) region of the genome. According to the Cleveland hierarchy, we are best at identifying positions along a common scale. Here I am showing how protein coding and non-protein coding substitutions vary with genomic position (a common scale). I believe that the font size and aspect ratio are all fine. I chose to re-scale the data so that the numbers along the x and y axis look "prettier". Since the genome and number of substitutions are integers, they can be scaled while still having the graphs make sense. This also necessitates the use of units so the viwer can determine the magnitude of the values on the graph. Since it is possible to have zero substitutions at a particular site as well as having the genome begin at zero, anchoring the graph at zero in both the x and y direction is appropriate and accurate.

I decided to use direct labels to avoide having the viewer rely on looking at a legend for reference. I do not really like all the white space that this creates from having to extend the x-axis to make the labels fit, so I am not sure if it was better to have the legend. (see graph at end for version with the legend) I suppose this might be a preference and asthetic issue and not a "bad graph" issue. Although, having the axis extend could mislead the viewer in thinking that the *E. coli* genome is longer than it actually is and that I am "missing data" from the end of the genome.

I tried to choose colours that are asthetically pleasing but also easy to distinguish between the two categorical variables. I additionally chose colours that would be colour-blind and black-and-white friendly (which is quite challenging when you want it to look pretty!). My favourite website to choose complementary colours from is coolors.

I also added a trend line to help guide the viewer in seeing how each of these genomic classifications varies with genomic position. It is clear that the protein coding substitutions have a negative trend, indicating that there are higher numbers of substitutions near the beginning of the genome than at the end. Biologically this is curious, we expect there to be more essential genes located near the beginning of the genome, and these genes should therefore have fewer substitutions because any mutation could be detrimental to the organism. For the non-protein coding genes it is less clear if this trendline is positive, negative or relatively flat.

Another biologically curious aspect to this graph is that overall the number of protein coding substitutions seems to exceede the number of non-protein coding substitutions. This should not be happening because again, we expect genes that are coding for proteins to be more important to the organisms survival than DNA that does not code for proteins. Therefore, non-protein coding regions should have more substitutions than protein coding regions.

Proportional Data:

I thought that the above graph was deceiving because it makes it look like there are more protein coding substitutions than non-protein coding substitutions. However, there is a disproportional amount of protein coding sites in bacterial genomes compared to non-protein coding sites. Bacterial genomes are almost exclusivly protein coding! So, I decided to divide the total number of protein coding or non-protein coding substitutions in a 10,000bp segment by the total number of protein coding or non-protein coding sites respectively.

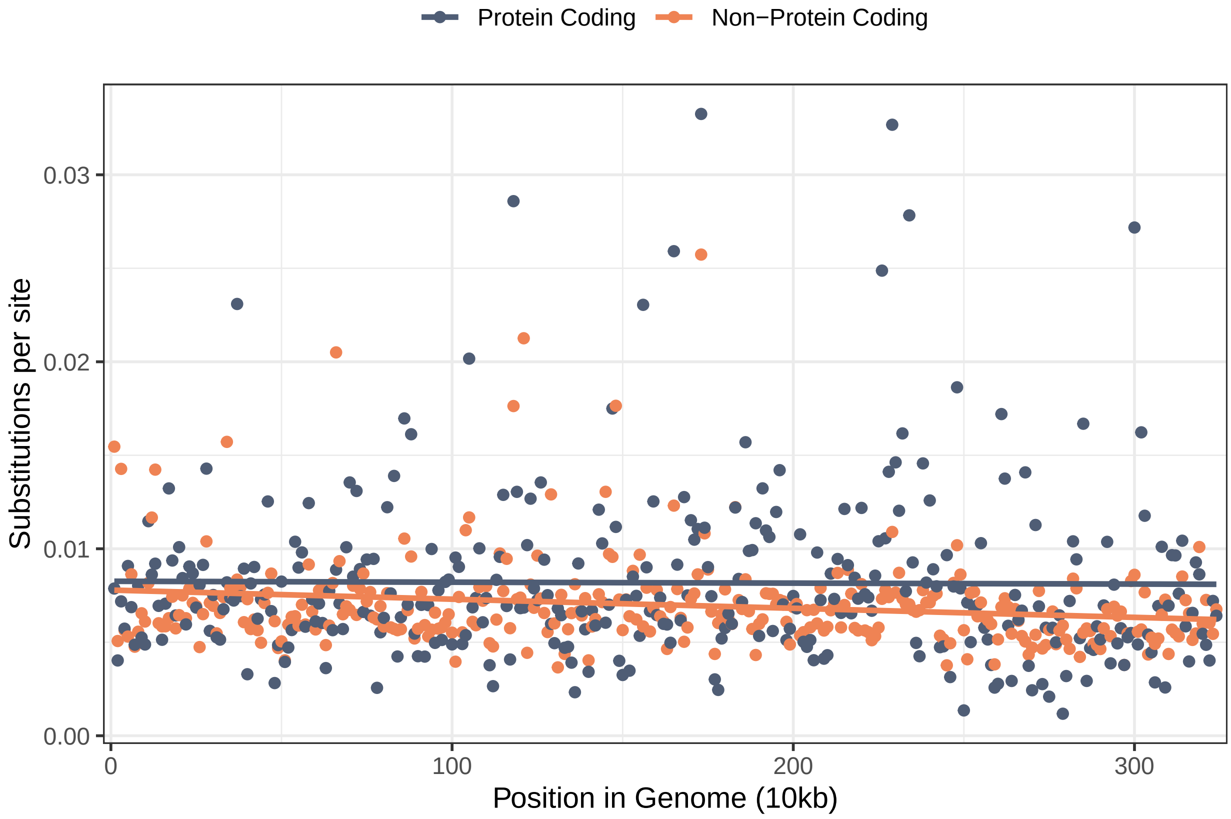
```
# subset data
chrom_data_weight <- chrom_data[, c("position", "cod_weight", "ncod_weight")]

# tidy df
chrom_data_weight <- chrom_data_weight %>% gather(genom_cat, val, cod_weight:ncod_weight
    mutate(genom_cat = gsub("genom_cat", "", genom_cat)) %>% arrange(position,
    genom_cat)

# scale data to look better on graph
chrom_data_weight$position <- chrom_data_weight$position/10000
# chrom_data_weight$val <- chrom_data_weight$val / 1000 change names so
# direct labels will be nicer
chrom_data_weight <- chrom_data_weight %>% mutate_if(is.character, str_replace_all,
    pattern = "^cod_weight", replacement = "Protein Coding")
chrom_data_weight <- chrom_data_weight %>% mutate_if(is.character, str_replace_all,
    pattern = "^ncod_weight", replacement = "Non-Protein Coding")
```

```
#defining colours again because it switched which was protein coding and which was non
cols <- c("#4F5D75","#EF8354")
#base graph
base_g2 <- ggplot(data=chrom_data_weight, aes(x=position, y=val, color= genom_cat))

wg <- (base_g2
       + geom_point()
       + geom_smooth(method = 'lm', se = FALSE)
       + theme_bw()
       + theme(legend.position="top")
       + labs(x = "Position in Genome (10kb)", y = "Substitutions per site")
       + scale_color_manual(values = cols, name="",labels = c("Protein Coding","Non-Prot
       #expand axis so direct labels can be seen
       + scale_x_continuous(expand = c(0.01,0))
)
wg
```



Discription of Graph 2:

The drastic trend of protein coding substitutions being higher than non-protein coding substitutions is reduced in this graph. However, it still appears as though there are still more protein coding substitutions per site than non-protein coding substitutions. This could be an artifact about how the data was collected. We had to implement a length cutoff for our sequences to ensure that there was enough information in a particular segment of sequence to ancestrally reconstruct the nucleotides and respective genomic positions. Non-protein coding regions of bacterial genomes are less conserved than the protein coding regions. Therefore, a significant portion of these non-protein coding regions may be missing due to the sequence length restriction. This could potentially be why the number of substitutions per site appear to be higher than non-protein coding regions.

With respect to the visualization of this graph, most of the same points that I mentioned before appily. I decided to include a legend here instead of direct labeling because the points are not organized in a way that clearly separates the protein coding and non-protein coding categories. So I thought a legend would be clearer than having direct labels.

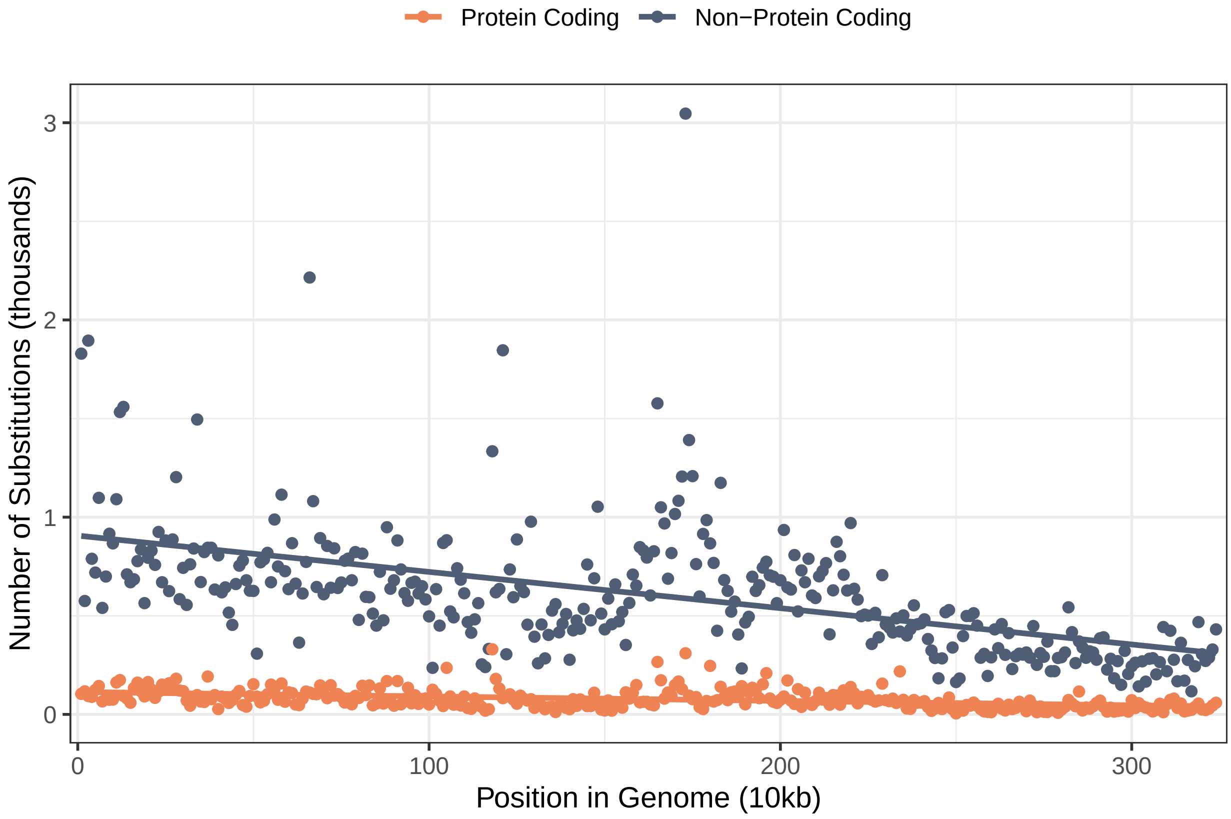
I keept the colouring of each genomic category the same as the previous graph so that anyone reading this report would begin to associate protein coding with dark blue and non-protein coding with orange. This makes for quicker recognition when presented with multiple graphs of the same categories.

I additionally ensured that the labels in the legend matched the order of the trend lines in the graph (left to right coensides with top to bottom).

Graph 1 with legend

```
#defining colours
cols <- c("#EF8354","#4F5D75")
#base graph
base_g <- ggplot(data=chrom_data_raw, aes(x=position, y=val, color= genom_cat))

rgl <- (base_g
       + geom_point()
       + geom_smooth(method = 'lm', se = FALSE)
       + theme_bw()
       + theme(legend.position="top")
       + labs(x = "Position in Genome (10kb)", y = "Number of Substitutions (thousands)"
       + scale_color_manual(values = cols, name="",labels = c("Protein Coding","Non-Prot
#       #expand axis so direct labels can be seen
       + scale_x_continuous(expand = c(0.01,0))
)
rgl
```


Protein Coding
Non−Protein Coding
Number of Substitutions (thousands)
3
2
1
0
0
100
200
300
Position in Genome (10kb)