# Naive Pseudoalignment Implementation

**Brendan Rossmango, 505370692**

## Introduction

Using Python, I implemented a pseudoalignment procedure that is given RNA-seq data in FASTA format (the reads and transcriptome files) and the k-mer length and outputs the vector of equivalence class counts in tsv format.

First, I used the BioPython library, specifically the SeqIO parser, to parse through the reads.fasta and chr11_transcriptome.fasta files to get the list of read sequences and a dictionary of isoform IDs to isoform sequences. The reads.fasta file had reads that had unknown bases (N) besides A, G, C, or T. When parsing the reads, I skipped these bases, so not all reads are 100 bases long.

Then, I made a k-mer map for all isoform sequences, where every possible k-mer from the isoform sequences were mapped to their respective isoforms IDs. For each read, I generated the equivalence classes of both the forward and reverse strands by taking the intersection of the equivalence classes of the read's k-mers, and then took the union of equivalence classes of the forward and reverse strands.

Then, for each read's set of equivalence classes, I made a dictionary that mapped each equivalence class set to the number of reads that are of the equivalence class set. I sorted this dictionary by the length of the equivalence class set and then produced the output.tsv file of equivalence class counts.

## Results

I took various statistics of the counts versus size of equivalence classes. The class with the highest number of counts was the 'NA' class at 231379 counts. The largest equivalence class

set was 54; this class had 55 counts. The equivalence class size with the highest mean and median was size 41, with mean count of 346 and median count of 384. Finally, the equivalence class size with the highest count over all of its sets was 2 - this count was 221,270.

| | Counts | Number of Items in Equivalence Class |
|---|---|---|
| Mean | 124.844 | 5.037 |
| Median | 9 | 3 |
| Mode | 1 | 1 |
| Minimum | 1 | 0 |
| Maximum | 231379 | 54 |

Table 1: Basic statistics of counts and size of equivalence class set

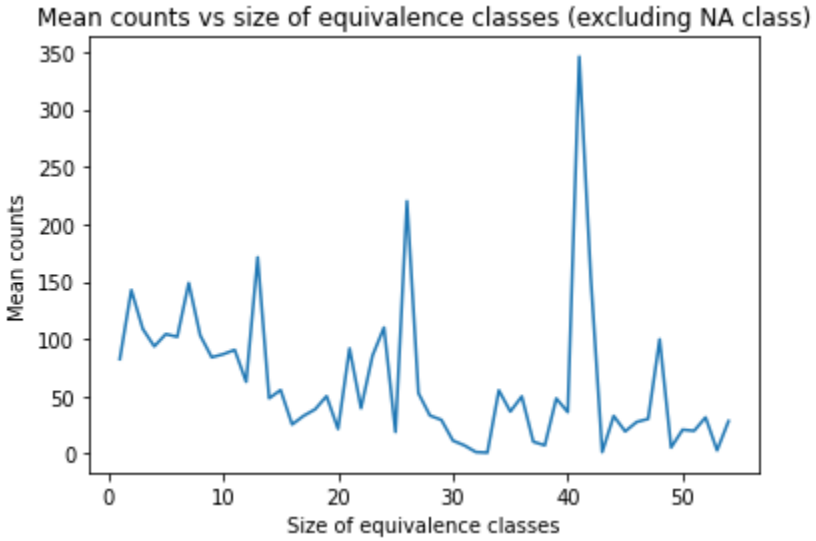


Figure 1: Mean counts vs size of equivalence classes (without ‘NA’ class)

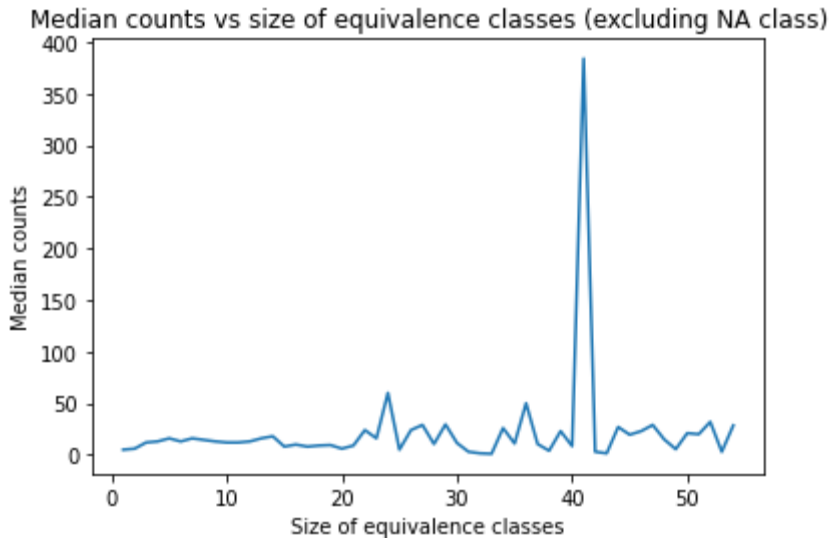


Figure 2: Median counts vs size of equivalence classes (without 'NA' class)

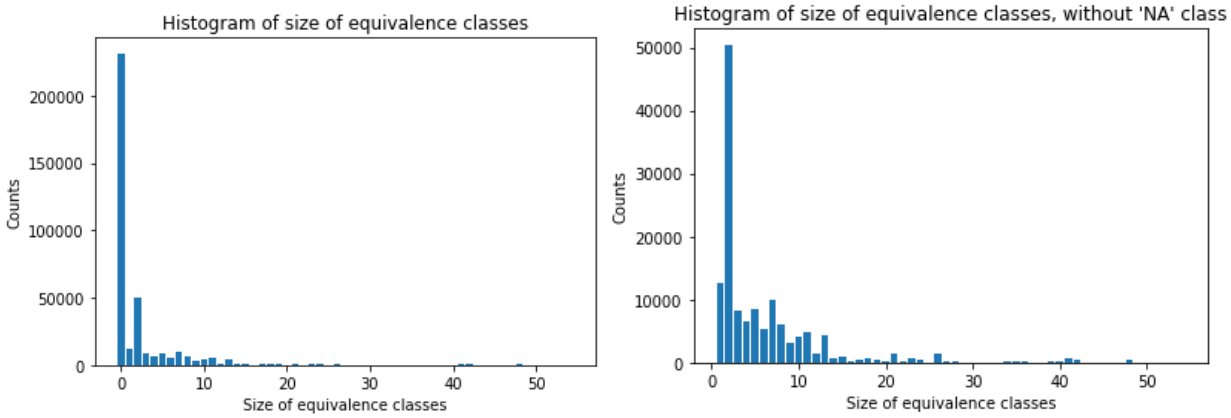


Figure 3ab: Histograms of size of equivalence classes

## Discussion

For reads that had erroneous bases (these were marked as N instead of A, G, C, or T), I chose to skip this unknown base and continue the pseudoalignment procedure as normal. Other methods I considered were to drop these reads entirely, or report the last non-empty equivalence class set (prior to the unknown base in the read sequence). Any of these methods are reasonable (since there is a sufficient number of total reads that dropping such reads entirely is fine to do).

However, I did not want to drop these reads entirely since there were 5610 reads that had unknown bases, which is quite a lot.

When handling the reverse complement, I simply took the union of equivalence classes of both the forward and reverse strands. Another method I considered was to only map the reverse strand if the forward strand's equivalence classes set was empty. However, I felt the method I used was more complete.

**References**

Introduction to SeqIO: Biopython. Retrieved March 24, 2023, from https://biopython.org/wiki/SeqIO