Ryan Burke

RNA Sequence Quantification using EM Algorithm

Design and Roadblocks

I implemented the Full-EM (No EC) model in python 3.7. I began heavy usage of python's dictionaries and lists but ran into some serious efficiency problems when accessing millions of maps of lists of lists of etc... The runtime was over 3 HOURS for even small files (alignments_small.cmsc423), so I had to re-write my code using flattened lists so Numba could digest it properly. I ran into plenty problems with Numba and my class structure (Numba wasn't a big fan of my implementation), so I pivoted to a more basic layout (below): a vanilla .py file with 5~ methods that work towards returning my results.

Ultimately, I was able to get alignments_small.cs423 to run in under 8 minutes and alignments.cs423 to run in under 40 minutes (which still isn't the best). I tried optimizing it here and there to no avail. I had a petty indexing error that took almost half a day to debug (it was just me accessing η with an incorrect index).

Other than that, the project was great. A couple serious roadblock that I'll avoid complaining about, but overall, I'm glad I finished this project, unlike the first! Professor Patro's supplementary readings/write-ups we're superb and unbelievably helpful.

A (Very) Basic Layout

```
imports
.
.
def effective_length(...):
  .
  .
def p_calculations():
  .
  .
@jit(nopython=True)
def runEM(...):
  .
  .
def EM_Algorithm():
  parsing transcripts and alignment blocks
  runEM()
  write results to output file
```

Results

My results we're maybe a little off. The scale of the plot was buggy and import matplotlib into my environment was unreasonably hard, so the results seem off. I think if I we're to scale them appropriately we'd see the results align relatively well, at extremely small numbers.

Mean Absolute Error: **21.25**

Spearman Correlation Plot: