bio$_{b}itsDocumentation$

Release 1.2.0

Tyghe Vallard, Michael Panciera

January 07, 2016

Contents

Various bioinformatics scripts

All documentation is hosted at http://bio-bits.readthedocs.org/en/latest

CHAPTER 1

TODO

- Include existing scripts

Contents:

1.1 Installation

It is recommended to install into a virtualenv. If you know what you are doing and don't want to install into virtualenv, then you can skip right to step 3

1. Setup Virtualenv

   It is assumed you have virtualenv already installed. If not see https://virtualenv.pypa.io/en/latest/installation.html

```
virtualenv env
```

2. Activate virtualenv

```
. env/bin/activate
```

3. Install dependencies

```
pip install -r requirements.txt
```

   For python 2.6 you will need to also install some additional packages

```
pip install -r requirements-py26.txt
```

4. Install bio_bits

```
python setup.py install
```

1.2 Scripts

1.2.1 rename_fasta

Many times you find you have a fasta file where the identifiers are all wrong and you want to rename them all via some mapping file.

Take the example where you have the following fasta file(example.fasta):

```
>id1
ATGC
>id2
ATGC
>id3
ATGC
```

You want to rename each identifier(id1, id2, id3) based on a mapping you have. In a file called renamelist.csv you would have the following:

```
#From,To
id1,samplename1
id2,samplename2
id3,samplename3
```

Then to rename your fasta without replacing the original file you have two options:

1. Rename without replacing original file

```
        rename_fasta renamelist.csv example.fasta > renamedfasta.fasta
```

2. Rename replacing original file's contents

```
        reanme_fasta renamelist.csv example.fasta --inplace
```

Rename Mapping File Syntax

The file you specify as the rename map file is a simple comma separated text file.

The following rules apply to the format:

- The first entry is the identifier to find in the supplied fasta file.
- The second entry is what to replace the found identifier with
- Any line beginning with a pound sign(#) will be ignored by the renamer

Missing identifiers that are in fasta but not rename file

In the case where your fasta file contains an identifier that is not in the rename map file you supply, an error will be displayed in the console telling you as such:

```
idwhatever is not in provided mapping
```

1.2.2 beast_checkpoint

beast_checkpoint is a fork of https://gist.github.com/trvrb/5277297 that has been rewritten in python and slightly improved as the ruby script seemed to have a few errors.

It accepts any previously run or terminated beast run and will generate an xml file that essentially starts from the last generated tree/log state.

Since beast is random in nature, there does not appear to be a way to restart the run exactly from the same state that it left off.

Example

We will use the benchmark2.xml file that comes with Beast 1.8 This file is located in:

```
BEASTv1.8.0/examples/Benchmarks/benchmark2.xml
```

First you need to fix the benchmark2.xml because each taxa has a trailing space and that is annoying

```
$> sed 's/ "/"/' benchmark2.xml > beast.xml
```

Now run beast for about half of the iterations and hit CTRL-C to kill it This benchmark is set to run 1,000,000 iterations so around 500,000 you can kill it. Notice we are using a predefined seed

```
$> seed=1234567890
$> mkdir run1
$> cp beast.xml run1/beast.xml
$> beast -seed $seed -beagle_SSE beast.xml
```

Now we will want to re-run beast from that last state. We can use beast_checkpoint to do so by supplying the original xml and the produced trees and log files. We will put the new xml into a new directory since the .trees and .log files would create an error or possibly be overwritten.

NOTE If your fileLog and treeFileLog do not have the same logEvery then when beast exits you may end up with more/less tree states than log states. For now you will have to manually edit the files and ensure that the last tree state matches the last log state.

Todo

Could be possible to get beast_checkpoint to check for that scenario and use the last tree state that matches the last log state

---

```
$> mkdir run2
$> beast_checkpoint beast.xml *.trees *.log > run2/beast.xml
```

Now you can simply just re-run beast on the new xml using the same seed

```
$> cd run2
$> beast -seed $seed -beagle_SSE beast.xml
```

Tracer

If you name your runs sequentially as we did in the example(aka, run1, run2,...) then you can easily load all log files into tracer via the command line as follows

```
tracer run*/*.log
```

LogCombiner

After you have run all your beast checkpointed xml files you will probably want to combine them with logcombiner which comes with beast

1.2.3 beast_wrapper

Beast wrapper is intended as a helper script to run beast. At this point it just runs beast with the same arguments you would normally give to beast from the command line and just adds a estimated time left column to the console output

Example

```
$> beast_wrapper -beagle_SSE my_beast.xml
...
state   Posterior       Prior           Likelihood      rootHeight      my_beast.ucld.mean  location.
0   -86527.5880     -6850.8316      -79676.7564     57.6772         1.16103E-3      4.86012         1
20000   -29044.3753     -1123.5287      -27920.8466     288.102         3.02471E-4      0.11891
40000   -25517.9525     -979.5343       -24538.4182     211.705         1.35118E-4      0.25060
60000   -24212.1250     -1040.4103      -23171.7147     188.454         1.05572E-4      0.18908
80000   -24097.9354     -1019.8099      -23078.1256     182.242         1.53593E-4      0.12857
100000  -24121.5382     -1105.6545      -23015.8837     178.060         1.26907E-4      0.10367
120000  -23930.6897     -1105.7390      -22824.9507     187.411         1.01885E-4      0.34214
140000  -23869.4856     -1087.1915      -22782.2942     178.535         8.76375E-5      0.26128
```

1.2.4 group_references

group_references splits an alignment file by reference into seperate FASTQ files. group_references takes a SAM or BAM file as input, and can optionally be given an output directory where the FASTQ files will be saved. If not output directory name is provided, the files will be saved in the new folder group_references_out.

```
$> group_references contigs.bam
$> group_references contigs.bam --outdir split_fastqs
```

1.2.5 degen

Find genes where a sequence has degenerate bases.

How-to

Usage: degen.py <fasta> <options>

Options:

--gb-id=<accession_id> Accession id for reference

--gb-file=<gbfile> Local Genbank file for reference

--tab-file=<tabfile> TSV/CSV file for reference with fields name,start,end

Example:

```
    degen sequence.fasta --gb-id   12398.91
    degen sequence.fasta --gb-file  tests/testinput/sequence.gb
    degen sequence.fasta --tab-file tests/testinput/degen.tab
    degen sequence.fasta --tab-file tests/testinput/degen.csv
```

Output:

Gene name, degnerate position, degenerate base:

```
anchored capsid protein         85      R
anchored capsid protein         88      Y
membrane glycoprotein precursor 509     R
nonstructural protein NS5       8513    Y
nonstructural protein NS5       8514    Y
nonstructural protein NS5       8515    Y
anchored capsid protein         85      R
anchored capsid protein         88      Y
membrane glycoprotein precursor 509     R
nonstructural protein NS5       8513    Y
nonstructural protein NS5       8514    Y
nonstructural protein NS5       8515    Y
```

Gene/Tab File

degen.tab could look like:

```
genename    start    stop
foo    1    2
bar    9    33
```

The headers do not matter, but the start field must always come before the stop field, so the below example would also be valid:

```
start    GENEName    stop
1    foo    2
9    bar    33
```

or optionally without headers:

```
1    foo    2
9    bar    33
```

alternatively, with commas in place of tabs:

```
name,start,stop
foo,1,2
bar,9,33
```

You can also specify a coding region(CDS) in your file as well:

```
name,start,stop

CDS,3,33
foo,1,2
bar,9,33
```

Genbank File

As downloaded from NCBI's entrez database. Use this option if you don't have internet access.

An example

```
LOCUS       KJ189367               10452 bp ss-RNA     linear   VRL 10-FEB-2014
DEFINITION  Dengue virus 1 isolate DENV-1/PR/BID-V8188/2010, complete genome.
ACCESSION   KJ189367
VERSION     KJ189367.1  GI:582052497
DBLINK      BioProject: PRJNA31235
```

```
KEYWORDS    .
SOURCE      Dengue virus 1
  ORGANISM  Dengue virus 1
            Viruses; ssRNA viruses; ssRNA positive-strand viruses, no DNA
            stage; Flaviviridae; Flavivirus; Dengue virus group.
REFERENCE   1  (bases 1 to 10452)
  AUTHORS   Zody,M.C., Newman,R.M., Henn,M., Munoz-Jordan,J., McElroy,K.L.,
            Santiago,G., Poon,T.W., Charlebois,P., Weiner,B., Yang,X.,
            Piper,M.E., Fitzgerald,M., McCowan,C., Young,S., Gargeya,S.,
            Levin,J., Malboeuf,C., Qu,J., Ireland,A., Chapman,S.B., Murphy,C.,
            Wortman,J., Nusbaum,C. and Birren,B.
  CONSRTM   Genome Resources in Dengue Consortium; The Broad Institute Genomics
            Platform; The Broad Institute Genome Sequencing Center for
            Infectious Disease; Centers for Disease Control and Prevention
            Division of Vector Borne Infectious Diseases; CDC Dengue Branch
            Puerto Rico
  TITLE     Direct Submission
  JOURNAL   Submitted (22-JAN-2014) Broad Institute of MIT & Harvard, 7
            Cambridge Center, Cambridge, MA 02142, USA
COMMENT     ##Assembly-Data-START##
            Assembly Method       :: Vicuna v. 1
            Sequencing Technology :: Illumina
            ##Assembly-Data-END##
FEATURES             Location/Qualifiers
     source          1..10452
                     /organism="Dengue virus 1"
                     /mol_type="genomic RNA"
                     /isolate="DENV-1/PR/BID-V8188/2010"
                     /isolation_source="cell supernatant"
                     /host="Homo sapiens"
                     /db_xref="taxon:11053"
                     /country="Puerto Rico"
                     /collection_date="2010"
                     /note="cell passage history: C6/36 1; cohort population:
                     Dengue Surveillance;
                     type: 1"
     5'UTR           1..83
                     /note="indels in UTR have not been validated"
     CDS             84..10262
                     /codon_start=1
                     /product="polyprotein"
                     /protein_id="AHI43750.1"
                     /db_xref="GI:582052498"
                     /translation="MNNQRKKTGRPSFNMLKRARNRVSTGSQLAKRFSKGLLSGQGPM
                     KLVMAFIAFLRFLAIPPTAGILARWSSFKKNGAIKVLRGFKKEISSMLNIMNRRKRSV
                     TMLLMLLPTALAFHLTTRGGEPHMIVSKQERGKSLLFKTSAGVNMCTLIAMDLGELCE
                     DTMTYKCPRITEAEPDDVDCWCNATDTWVTYGTCSQTGEHRREKRSVALAPHVGLGLE
                     TRTETWMSSEGAWKQIQRVETWALRHPGFTVIAFFLAHAIGTSITQKGIIFILLMLVT
                     PSMAMRCVGIGNRDFVEGLSGATWVDVVLEHGSCVTTMAKNKPTLDIELLKTEVTNPA
                     VLRKLCIEAKISNTTTDSRCPTQGEATLVEEQDANFVCRRTFVDRGWGNGCGLFGKGS
                     LLTCAKFKCVTKLEGKIVQYENLKYSVIVTVHTGDQHQVGNETTEHGTIATITPQAPT
                     SEIQLTDYGALTLDCSPRTGLDFNEMVLLTMKEKSWLVHKQWFLDLPLPWTSGASTSQ
                     ETWNRQDLLVTFKTAHAKKQEVVVLGSQEGAMHTALTGATEIQTSGTTTIFAGHLKCR
                     LKMDKLTLKGMSYVMCTGSFKLEKEVAETQHGTVLVQVKYEGTDAPCKIPFSTQDEKG
                     VTQNGRLITANPIVTDKEKPVNIETEPPFGESYIVVGAGEKALKLSWFKRGSSIGKMF
                     EATARGARRMAILGDTAWDFGSIGGVFTSVGKLVHQIFGTAYGVLFSGVSWTMKIGIG
                     ILLTWLGLNSRSTSLSMTCIVVGMVTLYLGVMVQADSGCVINWKGRELKCGSGIFVTN
                     EVHTWTEQYKFQADSPKRLSAAIGKAWEEGVCGIRSATRLENIMWKQISNELNHILLE
```

```
                     NDMKFTVVVGDANGILAQGKKMIRPQPMEHKYSWKSWGKAKIIGADIQNTTFIIDGPD
                     TPECPDGQRAWNIWEVEDYGFGVFTTNIWLKLRDSYTQMCDHRLMSAAIKDSKAVHAD
                     MGYWIESEKNETWKLARASFIEVKTCTWPKSHTLWSNGVLESEMIIPKIYGGPISQHN
                     YRPGYFTQTAGPWHLGKLELDFDLCEGTTVVVDEHCGNRGPSLRTTTVTGKIIHEWCC
                     RSCTLPPLRFRGEDGCWYGMEIRPVKEKEENLVRSMVSAGSGEVDSFSLGILCVSIMI
                     EEVMRSRWSRKMLMTGTLAVFLLLIMGQLTWNDLIRLCIMVGANASDRMGMGTTYLAL
                     MATFKMRPMFAVGLLFRRLTSREVLLLTIGLSLVASVELPNSLEELGDGLAMGIMMLK
                     LLTEFQPHQLWTTLLSLTFVKTTLSLDYAWKTTAMALSIVSLFPLCLSTTSQKTTWLP
                     VLLGSFGCKPLTMFLITENKIWGRKSWPLNEGIMAIGIVSILLSSLLKNDVPLAGPLI
                     AGGMLIACYVISGSSADLSLEKAAEVSWEQEAEHSGASHSILVEVQDDGTMKIKDEER
                     DDTLTILLKATLLAVSGVYPMSIPATLFVWYFWQKKKQRSGVLWDTPSPPEVERAVLD
                     NGIYRILQRGLLGRSQVGVGVFQDGVFHTMWHVTRGAVLMYQGKRLEPSWASVKKDLI
                     SYGGGWRFQGSWNTGEEVQVIAVEPGKNPKNVQTTPGTFKTPEGEVGAIALDFKPGTS
                     GSPIVNREGKIVGLYGNGVVTTSGTYVSAIAQAKASQEGPLPEIEDEVFKKRNLTIMD
                     LHPGSGKTRRYLPAIVREAIKRKLRTLILAPTRVVASEMAEALKGMPIRYQTTAVKSE
                     HTGREIVDLMCHATFTMRLLSPVRVPNYNMIIMDEAHFTDPASIAARGYISTRVGMGE
                     AAAIFMTATPPGSVEAFPQSNAVIQDEERDIPERSWNSGYDWITDFPGKTVWFVPSIK
                     SGNDIANCLRKNGKRVIQLSRKTFDTEYQKTKNNDWDYVVTTDISEMGANFRADRVID
                     PRRCLKPVILKDGPERVILAGPMPVTAASAAQRRGRIGRNQNKEGDQYVYMGQPLNND
                     EDHAHWTEAKMLLDNINTPEGIIPALFEPEREKSAAIDGEYRLRGEARKTFVELMRRG
                     DLPVWLSYKVASEGFQYSDRRWCFDGERNNQVLEENMDVEIWTKEGERKKLRPRWLDA
                     RTYSDPLALREFKEFAAGRRSVSGDLILEIGKLPQHLTLRAQNALDNLVMLHNSEQGG
                     KAYRHAMEELPDTIETLMLLALIAVLTGGVTLFFLSGKGLGKTSIGLLCVTASSALLW
                     MASVEPHWIAASIILEFFLMVLLIPEPDRQRTPQDNQLAYVVIGLLFMILTVAANEMG
                     LLETTKKDLGIGYVAAENHQHATMLDVDLHPASAWTLYAVATTVITPMMRHTIENTTA
                     NISLTAIANQAAILMGLDKGWPISKMDIGVPLLALGCYSQVNPLTLTAAVLMLVAHYA
                     IIGPGLQAKATREAQKRTAAGIMKNPTVDGIVAIDLDPVVYDAKFEKQLGQIMLLILC
                     TSQILLMRTTWALCESITLATGPLTTLWEGSPGKFWNTTIAVSMANIFRGSYLAGAGL
                     AFSLMKSLGGGRRGTGAQGETLGEKWKRQLNQLSKSEFNTYKRSGIMEVDRSEAKEGL
                     KRGETTKHAVSRGTAKLRWFVERNLVKPEGKVIDLGCGRGGWSYYCAGLKKVTEVKGY
                     TKGGPGHEEPIPMATYGWNLVKLHSGKDVFFMPPEKCDTLLCDIGESSPNPTIEEGRT
                     LRVLKMVEPWLRGNQFCIKILNPYMPSVVETLERMQRKHGGMLVRNPLSRNSTHEMYW
                     VSCGTGNIVSAVNMTSRMLLNRFTMAHRKPTYERDVDLGAGTRHVAVEPEVANLDIIG
                     QRIENIKNEHKSTWHYDEDNPYKTWAYHGSYEVKPSGSASSMVNGVVRLLTKPWDVIP
                     MVTQIAMTDTTPFGQQRVFKEKVDTRTPRAKRGTTQIMEVTAKWLWGFLSRNKKPRIC
                     TREEFTRKVRSNAAIGAVFVDENQWNSAKEAVEDERFWDLVHRERELHKQGKCATCVY
                     NMMGKREKKLGEFGKAKGSRAIWYMWLGARFLEFEALGFMNEDHWFSRENSLSGVEGE
                     GLHKLGYILRDISKIPGGNMYADDTAGWDTRVTEDDLQNEAKITDIMEPEHALLATSI
                     FKLTYQNKVVRVQRPAKNGTVMDVISRRDQRGSGQVGTYGLNTFTNMEVQLIRQMESE
                     GIFLPSELETPNLAERALDWLEKHGAERLKRMAISGDDCVVKPIDDRFATALTALNDM
                     GKVRKDIPQWEPSKGWNDWQQVPFCSHHFHQLIMKDGREIVVPCRNQDELVGRARVSQ
                     GAGWSLRETACLGKSYAQMWQLMYFHRRDLRLAANAICSAVPVDWVPTSRTTWSIHAH
                     HQWMTTEDMLSVWNRVWIDENPWMENKTHVSSWEEVPYLGKREDQWCGSLIGLTARAT
                     WATNIQVAINQVRRLIGNENYLDYMTSMKRFKNESDSEGALW"
     mat_peptide     84..425
                     /product="anchored capsid protein"
     mat_peptide     426..923
                     /product="membrane glycoprotein precursor"
     mat_peptide     924..2408
                     /product="envelope protein"
     mat_peptide     2409..3464
                     /product="nonstructural protein NS1"
     mat_peptide     3465..4118
                     /product="nonstructural protein NS2A"
     mat_peptide     4119..4508
                     /product="nonstructural protein NS2B"
     mat_peptide     4509..6365
                     /product="nonstructural protein NS3"
```

```
     mat_peptide     6366..6746
                     /product="nonstructural protein NS4A"
     mat_peptide     6747..6815
                     /product="2K peptide"
     mat_peptide     6816..7562
                     /product="nonstructural protein NS4B"
     mat_peptide     7563..10259
                     /product="nonstructural protein NS5"
     3'UTR           10263..10452
                     /note="indels in UTR have not been validated"
ORIGIN
        1 catctggacc gacaagaaca gtttcgaatc ggaagcttgc ttaacgtagt tctaacagtt
       61 ttttattaga gagcagatct ctgatgaaca accaacggaa aaagacgggt cgaccgtctt
      121 tcaatatgct gaaacgcgcg agaaaccgcg tgtcaactgg ttcacagttg gcgaagagat
      181 tctcaaaagg attgctttca ggccaaggac ccatgaaatt ggtgatggct ttcatagcat
      241 ttctaagatt tctagccata cccccaacag caggaatttt ggctagatgg agctcattca
      301 agaagaatgg agcaattaaa gtgttacggg gtttcaaaaa agagatctca agcatgttga
      361 acataatgaa caggaggaaa agatccgtga ccatgctcct catgctgctg cccacagccc
      421 tggcgtttca tttgaccaca cgagggggag agccacacat gatagttagt aagcaggaaa
      481 gaggaaagtc actcttgttt aagacctctg cgggcgtcaa tatgtgcacc ctcattgcga
      541 tggacttggg agagttatgt gaggacacaa tgacctacaa atgcccccgg atcactgagg
      601 cggaaccaga tgacgttgac tgctggtgca atgccacaga cacatgggtg acctatggga
      661 cgtgttctca aaccggcgaa caccgacgag agaaacgttc cgtggcactg gccccacacg
      721 tgggacttgg tctagaaaca agaaccgaaa catggatgtc ctctgaaggc gcctggaaac
      781 aaatacaaag agtggaaact tgggctttga gacacccagg attcacggtg atagcctttt
      841 ttttagcaca tgctatagga acatccatca ctcagaaagg gatcattttc atcttgctga
      901 tgctggtgac accatcaatg gccatgcgat gcgtgggaat aggcaacaga gacttcgttg
      961 aaggactgtc aggagcaacg tgggtggacg tggtactgga gcacggaagc tgcgtcacca
     1021 ccatggcaaa aaataaacca acattggaca ttgaactctt gaagacggag gtcacgaacc
     1081 ctgccgtctt gcgcaaactg tgcattgaag ctaaaatatc aaacaccacc accgattcaa
     1141 gatgtccaac acaaggagag gccacactgg tggaagaaca agacgcgaac tttgtgtgtc
     1201 gccgaacgtt tgtggacaga ggctggggta atggctgcgg actattcgga aagggaagtc
     1261 tattgacgtg tgccaagttc aagtgtgtga caaaactaga aggaaagata gttcaatatg
     1321 aaaacctaaa atattcagtg atagtcactg tccacactgg ggaccagcac caggtgggaa
     1381 acgagaccac agaacatgga acaattgcaa ccataacacc tcaagctccc acgtcggaaa
     1441 tacagctgac cgactacgga gccctcacac tggactgctc acctagaaca gggctggact
     1501 ttaatgagat ggtgctattg acaatgaaag aaaaatcatg gcttgtccac aaacaatggt
     1561 ttctagactt gccactgcca tggacttcgg gggcttcaac atcccaagag acctggaaca
     1621 gacaagattt gctggtcaca ttcaagacag ctcatgcaaa gaaacaggaa gtagtcgtat
     1681 tgggatcaca ggaaggagca atgcatactg cgttgactgg ggcgacagaa atccagacgt
     1741 caggaacgac aacaatcttc gcaggacacc tgaaatgcag actaaaaatg gataaactga
     1801 ccttaaaggg gatgtcatat gtgatgtgca caggctcatt taagctagag aaggaagtgg
     1861 ctgagaccca gcatggaact gttctagtgc aggtcaaata tgaaggaaca gacgcgccat
     1921 gcaagatccc cttttcgacc caagatgaga aaggagtgac ccagaatggg agattgataa
     1981 cagccaatcc catagttact gacaaagaaa aaccagtcaa cattgagaca gaaccacctt
     2041 ttggtgagag ctacatcgtg gtaggggcag gcgaaaaagc tttgaaacta agctggttca
     2101 agagaggaag cagcataggg aaaatgttcg aagcaaccgc ccgaggagca cgaaggatgg
     2161 ctatcctggg agacaccgca tgggacttcg gttctatagg aggagtgttt acatctgtgg
     2221 gaaaattggt acaccagatt tttggaaccg catatggggt tctgtttagc ggtgtttctt
     2281 ggaccatgaa aataggaata gggattctgc tgacatggtt gggattaaat tcaaggagca
     2341 cgtcactttc gatgacgtgc attgtagttg gcatggtcac actgtaccta ggagtcatgg
     2401 ttcaagcgga ttcgggatgt gtgatcaact ggaagggcag agaacttaaa tgcggaagtg
     2461 gcatttttgt cactaatgaa gtccacactt ggacagagca atacaaattc caggctgact
     2521 ccccaaaaag actgtcagca gccattggaa aggcgtggga ggagggcgtg tgtggaattc
     2581 gatcagccac gcgtcttgag aacatcatgt ggaagcagat atcaaatgaa ttgaaccaca
     2641 ttttacttga gaatgacatg aaattcacag tggttgtagg agatgccaac ggaattttgg
     2701 cccaaggaaa aaaaatgatt aggccacaac ccatggaaca caaatactca tggaaaagct
     2761 ggggaaaagc taaaatcata ggagcagaca tacaaaatac caccttcatt atcgacggcc
```

```
2821 cagacacccc agaatgtcct gatggccaaa gagcatggaa catttgggaa gttgaggact
2881 atgggtttgg agttttcacg acaaacatat ggctgaaatt gcgtgactcc tacacccaaa
2941 tgtgtgacca ccggctaatg tcagctgcca tcaaggacag caaggcagtc catgctgaca
3001 tggggtactg gatagaaagt gaaaagaacg aaacctggaa gttggcgaga gcctccttca
3061 tagaagtcaa aacatgcacc tggccgaaat ctcacactct atggagcaat ggagttttgg
3121 aaagtgaaat gataatccca aagatatatg gaggaccaat atctcagcac aactacagac
3181 cagggtattt cacacaaaca gcagggccat ggcacctagg taagttggaa ctggattttg
3241 acttgtgtga aggcaccaca gttgttgtgg atgaacattg tggaaatcga ggtccatctc
3301 tcagaaccac aacagtcaca ggaaagataa tccatgaatg gtgttgcaga tcctgcacgc
3361 taccccccct acgtttcaga ggagaagacg ggtgttggta tggcatggaa atcagaccag
3421 tgaaggagaa ggaggagaat ctagttaggt caatggtctc tgcagggtca ggagaagtgg
3481 acagtttttc attaggaata ctatgcgtat caataatgat tgaagaagtg atgagatcca
3541 gatggagtag aaagatgctg atgactggaa cactggctgt cttcctcctt cttataatgg
3601 gacaactgac atggaatgat ctgattaggt tatgcatcat ggtcggagct aacgcttcag
3661 acaggatggg gatgggaaca acgtacctag ccttgatggc tactttcaaa atgagaccaa
3721 tgttcgctgt agggctatta ttccgcagac taacatccag agaagttctt ctcctaacga
3781 ttggattaag cctggtggca tccgtggagc taccaaattc cttggaggag ctaggggatg
3841 gacttgcaat gggtatcatg atgttaaaat tgttgactga atttcagcca caccagttat
3901 ggaccacctt attgtctctg acatttgtca aaacaactct ctcattggat tatgcatgga
3961 aaacaacggc tatggcactg tctatcgtat ctctctttcc tttatgcctg tctacgacct
4021 cccaaaaaac aacatggctt ccggtgctgt taggatcttt tggatgcaaa ccattaacca
4081 tgtttcttat aacagaaaat aaaatctggg gaaggaaaag ttggcccctc aatgaaggaa
4141 ttatggctat tggaatagtc agcattctac taagctcact cctcaaaaat gatgtgccgt
4201 tggccgggcc attaatagct ggaggcatgc taatagcatg ttatgtcata tccggtagct
4261 cagccgattt atcattggag aaagcggctg aagtatcctg ggaacaagaa gcagaacact
4321 ccggtgcctc acacagcata ttagtagagg tccaagatga tggaactatg aaaataaaag
4381 atgaagagag ggatgacaca ctcaccatac tccttaaagc aactttgctg gcagtctcag
4441 gagtgtaccc aatgtcaata ccagcaactc tttttgtgtg gtatttttgg cagaaaaaga
4501 aacagagatc aggagtgtta tgggacacac ccagccctcc ggaagtggaa agagcagttc
4561 ttgataatgg catctataga atcttgcaaa gaggattgtt gggcaggtcc caagtaggag
4621 tgggagtttt ccaagacggc gtgttccaca caatgtggca cgttaccagg ggagctgtcc
4681 ttatgtacca agggaagaga ctggaaccaa gctgggccag tgtgaaaaag gacttgatct
4741 catatggagg aggttggagg ttccaaggat catggaacac gggagaagaa gtgcaggtaa
4801 tagctgttga accaggaaaa aaccccaaaa atgtacagac aacgccgggc acctttaaga
4861 ctcctgaagg cgaagttgga gccatagctc tagatttcaa acccggcaca tctggatctc
4921 ccatcgtgaa cagagaggga aaaatagtgg gtctgtatgg aaatggagtg gtgacaacaa
4981 gtggaaccta cgtcagtgcc attgcccaag ctaaagcatc acaggaaggg cctctaccag
5041 agattgagga cgaggtattt aagaaaagaa acttaacaat aatggacctg cacccaggat
5101 cagggaaaac aagaagatat cttccagcca tagtccgtga ggccataaaa aggaaactgc
5161 gtacgttaat cctggctccc acaagagttg tcgcctctga aatggcagag gcactcaagg
5221 gaatgccaat aagatatcag acaacagcag tgaagagtga acacacagga agggagatag
5281 ttgacctcat gtgccacgct acttttacca tgcgtctctt atccccagtg agagttccca
5341 attacaacat gatcattatg gatgaagcac attttaccga tccagctagc atagcggcca
5401 gagggtacat ctcaacccga gtgggtatgg gtgaagcagc tgcgatcttt atgacagcca
5461 ctcccccagg atcggtggag gcctttccac agagcaatgc agttatccaa gatgaggaaa
5521 gagacattcc tgagagatca tggaactcag gctacgactg gatcactgac tttccaggta
5581 aaacagtctg gtttgttcca agcattaaat caggaaatga cattgccaac tgtttaagaa
5641 agaacggaaa acgggtaatc caattgagca gaaaaacctt tgacactgag taccagaaaa
5701 caaaaaacaa tgactgggac tatgttgtca caacagacat ttctgaaatg ggggcaaatt
5761 tccgggccga cagggtaata gacccaaggc ggtgcttgaa accggtaata ctaaaagatg
5821 gtccagagcg tgtcattcta gccggaccga tgccagtgac tgcggccagt gctgcccaga
5881 ggagaggaag aattggaagg aaccaaaaca aggaaggtga tcagtatgtt tatatgggac
5941 agcctttaaa taatgatgag gatcacgctc attggacaga agcaaaaatg ctccttgaca
6001 atataaacac accagaaggg atcatcccag ccctttttga gccagagaga gaaaagagtg
6061 cagcaataga cggggagtac agactgcggg gagaagcaag gaaaacgttc gtggagctca
6121 tgagaagagg agatctacca gtttggctat cctacaaagt agcctcagaa ggtttccagt
6181 actccgacag aaggtggtgc tttgatgggg aaaggaacaa ccaggtgttg gaggagaaca
6241 tggacgtgga gatctggaca aaggaaggag aaagaaagaa attgcgacct cgctggttgg
```

```
6301 acgccagaac atactctgat ccattggccc tgcgcgagtt taaagagttc gcagcaggaa
6361 gaagaagtgt ctcaggtgac ctgatattgg aaatagggaa acttccacaa catttgacgt
6421 taagagccca gaatgctctg gacaacttgg tcatgttgca caattccgaa caaggaggaa
6481 aagcctacag acatgccatg gaggaactac cagacaccat agaaacattg atgctactag
6541 ctttgatagc tgtgttgact ggtggagtga cgctgttctt cctatcagga aaaggcctag
6601 ggaaaacatc cattggcttg ctctgtgtga cggcctcaag cgcactgtta tggatggcca
6661 gtgtggagcc ccattggata gcggcctcca tcatactaga gttctttttg atggtgctgc
6721 tcattccaga gccagacaga cagcgcactc cacaggacaa ccagctagca tatgtggtga
6781 taggtttgtt attcatgata ctgacagtgg cagccaatga gatgggatta ttggaaacca
6841 caaagaaaga cctggggatt ggctatgtag ccgccgaaaa ccaccaacat gccacaatgc
6901 tggacgtaga cctacaccca gcttcagcct ggaccctcta tgcagtagcc acaacagtca
6961 tcactcccat gatgagacac acaattgaaa atacaacggc aaacatttcc ctgaccgcca
7021 ttgcaaatca ggcagctata ttgatgggac ttgacaaggg atggccaata tcgaagatgg
7081 acataggagt tccacttctc gccttagggt gctattccca ggtgaaccca ttgacactga
7141 cagcggcggt gttgatgtta gtggctcatt atgccataat tggaccagga ctgcaagcaa
7201 aggccactag agaagcccaa aaaaggacag cagccggaat aatgaaaaat ccaaccgtag
7261 acgggattgt tgcaatagac ttggatcctg tggtttatga tgcaaaattt gaaaaacaac
7321 taggccaaat aatgttactg atactttgta catcacagat cctcttgatg cggaccacat
7381 gggccttgtg tgaatccatc acactggcta ctggacccct gaccactctc tgggagggat
7441 ctccaggaaa attctggaat accacaatag cagtgtccat ggcaaatatt ttcaggggaa
7501 gttatctagc aggagcaggt ctggctttct cattgatgaa atctttagga ggaggtagga
7561 gaggcacggg agctcaaggg gaaacactgg gagagaaatg gaaaagacag ttgaaccaac
7621 tgagcaagtc agaattcaac acctacaaaa ggagtgggat tatggaggtg gacagatccg
7681 aagccaaaga gggactgaaa agaggagaaa caaccaaaca tgcagtgtca agaggaacag
7741 ccaaactgag gtggtttgtg gagaggaacc tcgtgaaacc agaaggaaaa gtcatagacc
7801 tcggttgtgg aagaggtggc tggtcatatt attgtgctgg gctgaagaaa gttactgaag
7861 tgaagggata cacaaaagga ggacctggac atgaggaacc tatcccaatg gcgacctatg
7921 gatggaacct agtaaaacta cactctggaa aggatgtatt ttttatgcca cctgagaaat
7981 gtgacactct tctgtgtgat attggtgagt cctctccgaa tccaactata gaagaaggaa
8041 gaacgttacg tgttctaaaa atggtggaac catggctcag aggaaaccaa ttctgcataa
8101 aaatcctaaa tccttacatg ccaagtgtgg tagaaactct ggagcgaatg caaagaaaac
8161 atggagggat gctagtgcga aacccactct caagaaattc tacccatgaa atgtattggg
8221 tttcatgtgg aacaggaaac attgtgtcgg cagtgaacat gacatccaga atgttactga
8281 accgattcac aatggctcac aggaagccaa catatgaaag agacgtggac ttaggcgctg
8341 gaacaagaca tgtggcagtg gaaccagagg tagccaacct agatatcatt ggccagagga
8401 tagaaaatat aaaaaatgaa cacaagtcaa catggcatta tgatgaggac aatccataca
8461 aaacatgggc ctatcatgga tcatatgagg tcaagccatc aggatcagcc tcatctatgg
8521 tgaatggagt ggtgagattg ctcacgaaac catgggatgt catccccatg gtcacacaaa
8581 tagctatgac tgataccaca ccctttggac aacagagagt gtttaaagag aaagttgaca
8641 cgcgcacacc aagagcaaaa cgaggcacaa cacagattat ggaggtgaca gccaagtggt
8701 tatggggttt cctttccaga aacaaaaaac ccagaatctg cacaagagag gagttcacaa
8761 gaaaggttag gtcaaacgcg gcaataggag cagtgttcgt tgatgaaaac caatggaact
8821 cagcaaaaga agcagtggaa gacgaaaggt tttgggatct tgtgcacaga gagagggagc
8881 ttcataaaca gggaaaatgt gccacgtgtg tctacaacat gatggggaag agagagaaaa
8941 aattaggaga gtttggaaag gcaaaaggaa gtcgtgcaat atggtacatg tggctgggag
9001 cacgctttct ggagttcgaa gcccttggtt ttatgaatga agatcactgg tttagtagag
9061 agaattcact cagtggagtg gaaggagaag gactgcacaa acttggatac atactcagag
9121 acatatcaaa gattccgggg ggaaatatgt atgcagatga tacagccgga tgggacacaa
9181 gagtaacaga ggatgacctc cagaatgagg ctaaaatcac tgacatcatg gagcctgaac
9241 atgctctatt ggctacgtca atttttaagc tgacttatca aaacaaggtg gtgagggtgc
9301 aaagaccagc aaaaaatgga accgtgatgg atgttatatc cagacgtgat cagagaggga
9361 gtggacaggt cggaacttat ggcttaaata ctttcaccaa tatggaggtc caactaataa
9421 gacaaatgga gtctgaggga atctttttac ccagcgaatt ggaaaccccc aacctagctg
9481 agagggctct tgactggtta gaaaaacatg gcgccgaaag gctgaaacga atggcaatca
9541 gcggagatga ttgcgtggtg aaaccaattg acgacaggtt cgcaacagcc ttaacagctc
9601 tgaatgacat gggaaaagta aggaaagaca taccgcagtg ggaaccttca aaaggatgga
9661 atgattggca gcaagtgcct ttttgttcac accatttcca ccaactgatc atgaaggatg
9721 ggagggaaat agtggtgcca tgccgcaacc aagatgaact tgtgggcagg gctagagtat
```

```
     9781 cacaaggcgc cggatggagc ctgagagaaa ctgcttgcct aggcaagtca tatgcacaaa
     9841 tgtggcagct gatgtacttc cacaggagag acctgagact agcggctaac gctatctgtt
     9901 cagccgtccc agttgattgg gtcccaacca gccgcacaac ctggtcaatc catgcccacc
     9961 accaatggat gacaacagaa gacatgttat cagtgtggaa tagggtttgg atagacgaaa
    10021 acccatggat ggagaacaaa actcatgtat ccagttggga agaagttcca tacctaggaa
    10081 aaagggaaga tcaatggtgt ggatccctga taggcttgac agcgagggcc acctgggcca
    10141 ccaacataca agtagccata aaccaagtga gaaggctcat cgggaatgag aattatttag
    10201 attacatgac atcaatgaag agattcaaga atgagagtga ttccgaagga gcactctggt
    10261 aagtcaacac actcatgaaa taaaggaaaa tagaagatca aacaaagtaa gaagtcaggc
    10321 cagattaagc catagcacgg aaagagctat gctgcctgtg agccccgtcc aaggacgtaa
    10381 aatgaagtca ggccgaaagc cacggattga gcaagccgtg ctgcctgtgg ctccatcgtg
    10441 gggatgtagc tc
//
```

1.2.6 parallel_blast

Parallel blast is a wrapper script around the blast commands as well as diamond. It utilizes GNU Parallel to run the commands in parallel by splitting up the input fasta files and distributes them across multiple subprocesses. If it detects that it is running inside of a PBS or SGE job it will run the job on multiple hosts that may be allocated to the job.

parallel_blast requires that you have gnu parallel installed and in your environments PATH as well as diamond and/or blastn/blastx/blastp.

- diamond
- blast
- GNU parallel

Usage

You can get all the arguments that can be supplied via the following

```
$> parallel_blast --help
```

Examples

For the examles below assume you have an input fasta in the current directory called `input.fasta`

Running blastn

```
$> parallel_blast input.fasta output.blast --ninst 4 --db /path/to/nt \
--blast_exe blastn --task megablast --blast_options "--evalue 0.01"
[cmd] /path/to/parallel -u --pipe --block 10 --recstart > --sshlogin 4/: /path/to/blastn -task megabl
```

Notice how we had to quote the additional `--blast_options`

Running diamond Diamond v0.7.9 is the version that was tested with parallel_blast. As diamond is still in development the options may change in future versions and parallel_blast may not run them correctly. Please submit a new issue if you find any issues.

```
$> parallel_blast input.fasta out.blast --ninst 4 --db /path/to/diamondnr \
--blast_exe diamond --task blastx --blast_options "--tmpdir dtmp"
[cmd] /path/to/parallel -u --pipe --block 10 --recstart > --cat --sshlogin 1/: /path/to/diamond blast
```

Notice how even though we specified `--ninst 4` that `--sshlogin 1/:` was used and `--threads 4` was set instead.

Note In recent versions of diamond, diamond outputs a daa binary file instead of a tab separated file. parallel_blast automatically converts the diamond output from daa to tab format for you but leaves the daa file behind(Same name as the output file you specify, but with the extension .daa)

Command that is run You will notice in the examples above that when you run parallel_blast that it outputs the command that it is running in case you want to copy/paste it and run it yourself sometime.

You might notice that the command does not include all the quoted arguments such as the `--recstart` argument which should be `--recstart ">"` as well as the `--outfmt` which should be quoted as `--outfmt "6 ..."`. If you intend on rerunning the command you will have to add the quotes manually.

Running inside of a PBS or SGE Job

parallel_blast is able to detect if it is running inside of a PBS or SGE job by looking to see if `PBS_NODEFILE` or `PE_HOSTFILE` is set in the environment's variables.

If it finds either of them it will run the job by supplying `--sshlogin` for each host it finds in the file.

`PBS_NODEFILE` and `PE_HOSTFILE` have different syntax so parallel_blast first builds a CPU,NODENAME list from them.

PBS_NODEFILE

This file is parsed and counts how many of each unique host is listed such that the following PBS_NODEFILE:

```
node1.localhost
node2.localhost
node2.localhost
node3.localhost
node3.localhost
node3.localhost
```

would run 1 instance on node1.localhost, 2 instances on node2.localhost and 3 instances on node3.localhost

PE_HOSTFILE

This file is almost in the exact syntax that parallel_blast uses so it is almost a 1-to-1 mapping.

Diamond and multiple hosts

Since diamond utilizes threads much more efficiently than blast, for each unique host in a job only 1 instance is launched but the `-p` option is set to the number of CPUS for each host listed in the `PE_HOSTFILE` or `PBS_NODEFILE`

1.2.7 degen_regions

Finds all degenerate bases in a given fasta input file that may contain multiple sequeces and reports their position as well as the annotated gene name that contains them.

The fasta file must be previously aligned to the query sequence. That is, if you are using a genbank annotation file or having the script download it for you, you should have aligned all your input sequences to that sequence.

The annotation is retrieved via supplied genbank accession, genbank file path or gene tab/csv file.

Usage

You can view the usage of degen_regions via:

```
degen_regions --help
```

Using Genbank Files

If you already have downloaded the genbank annotation file(typically the extension is .gb) you can use the *–gb-file* argument

The following will use the test input fasta file as well as the test input genbank file to find all degenerate bases and will put the output in a tab separated file called output.tsv

```
degen_regions -i tests/Den4_MAAPS_TestData16.fasta -o output.tsv --gb-file tests/testinput/sequence.g
```

Fetching Genbank Files Automatically

If you want the script to automatically fetch the Genbank annotation file from the internet you can use the *–gb-id* option and specify an accession number.

```
degen_regions -i tests/Den4_MAAPS_TestData16.fasta -o output.tsv --gb-id KJ189367
```

Using tab/csv file of gene annotation info

If you have a tab/csv file of gene annotations you can supply that using the *–tab-file* argument

You can read more about the format of the tab/csv annotation file in the degen docs

```
degen_regions -i tests/Den4_MAAPS_TestData16.fasta -o output.tsv --gb-file tests/testinput/sequence.g
```

Manually specify CDS

You can use the `--cds` argument to set the coding region. This argument should be comma separated such as `start,stop`. Specifying this argument will override any other cds found in the tab file, genbank file or fetched genbank file.

The following would mark all locations as NON-CODING as you are specifying that only position 1 is coding

```
degen_regions -i tests/Den4_MAAPS_TestData16.fasta -o output.tsv --gb-file tests/testinput/sequence.g
```

Output

The output is a simple tab separated file

seq id	nt Position	aa position	nt composition	aa compos
721	991	331	WCA	S/T
721	1307	436	AYA	I/T
721	1826	609	AYA	I/T
721	1865	622	GRA	E/G
721	7766	2589	ARA	K/R
2055_Den4/AY618992_1/Thailand/2001/Den4_1	1927	643	RAC	D/N
2055_Den4/AY618992_1/Thailand/2001/Den4_1	2833	945	YCG	P/S
2055_Den4/AY618992_1/Thailand/2001/Den4_1	3565	1189	YAT	H/Y
2055_Den4/AY618992_1/Thailand/2001/Den4_1	6271	2091	RAA	E/K
2055_Den4/AY618992_1/Thailand/2001/Den4_1	8656	2886	YAT	H/Y
2055_Den4/AY618992_1/Thailand/2001/Den4_1	8998	3000	YAG	*/Q
2055_Den4/AY618992_1/Thailand/2001/Den4_1	9811	3271	YCC	P/S
2055_Den4/AY618992_1/Thailand/2001/Den4_1	10542	3515	AGN	NON-CODING
2055_Den4/AY618992_1/Thailand/2001/Den4_1	10543	3515	NNN	NON-CODING
2055_Den4/AY618992_1/Thailand/2001/Den4_1	10541	3514	NNN	NON-CODING
2055_Den4/AY618992_1/Thailand/2001/Den4_1	10539	3514	NNN	NON-CODING
2055_Den4/AY618992_1/Thailand/2001/Den4_1	10546	3516	NNN	NON-CODING
2055_Den4/AY618992_1/Thailand/2001/Den4_1	10544	3515	NNN	NON-CODING
2055_Den4/AY618992_1/Thailand/2001/Den4_1	10542	3515	NNN	NON-CODING
1942_Den4/AY618992_1/Thailand/2001/Den4_1	4540	1514	RTA	I/V
1942_Den4/AY618992_1/Thailand/2001/Den4_1	10177	3393	MCA	P/T
1942_Den4/AY618992_1/Thailand/2001/Den4_1	10546	3516	NNN	NON-CODING
1942_Den4/AY618992_1/Thailand/2001/Den4_1	10544	3515	NNN	NON-CODING
1942_Den4/AY618992_1/Thailand/2001/Den4_1	10542	3515	NNN	NON-CODING
1875_Den4/AY618992_1/Thailand/2001/Den4_1	1514	505	AYG	M/T
1875_Den4/AY618992_1/Thailand/2001/Den4_1	3056	1019	ARA	K/R
1875_Den4/AY618992_1/Thailand/2001/Den4_1	3058	1020	KCA	A/S
1875_Den4/AY618992_1/Thailand/2001/Den4_1	3073	1025	WTT	F/I
1875_Den4/AY618992_1/Thailand/2001/Den4_1	3491	1164	AYC	I/T
1875_Den4/AY618992_1/Thailand/2001/Den4_1	3895	1299	RTG	M/V
1875_Den4/AY618992_1/Thailand/2001/Den4_1	7445	2482	GYA	A/V
948_Den4/AY618992_1/Thailand/2001/Den4_1	2819	940	ARC	N/S
871_Den4/AY618992_1/Thailand/2001/Den4_1	2947	983	RCC	A/T
871_Den4/AY618992_1/Thailand/2001/Den4_1	3058	1020	KCA	A/S
871_Den4/AY618992_1/Thailand/2001/Den4_1	3073	1025	WTT	F/I
871_Den4/AY618992_1/Thailand/2001/Den4_1	3116	1039	GYG	A/V
871_Den4/AY618992_1/Thailand/2001/Den4_1	3181	1061	RTW	I/V
871_Den4/AY618992_1/Thailand/2001/Den4_1	3179	1060	RTW	I/V
871_Den4/AY618992_1/Thailand/2001/Den4_1	3338	1113	ART	N/S
871_Den4/AY618992_1/Thailand/2001/Den4_1	3362	1121	ARA	K/R
871_Den4/AY618992_1/Thailand/2001/Den4_1	3373	1125	WCR	S/T
871_Den4/AY618992_1/Thailand/2001/Den4_1	3371	1124	WCR	S/T
871_Den4/AY618992_1/Thailand/2001/Den4_1	4314	1439	ATV	I/M
871_Den4/AY618992_1/Thailand/2001/Den4_1	7045	2349	WCC	S/T
871_Den4/AY618992_1/Thailand/2001/Den4_1	10536	3513	GAW	NON-CODING
871_Den4/AY618992_1/Thailand/2001/Den4_1	10537	3513	YCA	NON-CODING
947_Den4/AY618992_1/Thailand/2001/Den4_1	2971	991	YTY	F/L
947_Den4/AY618992_1/Thailand/2001/Den4_1	2969	990	YTY	F/L
947_Den4/AY618992_1/Thailand/2001/Den4_1	6763	2255	YTT	F/L
1793_Den4/AY618992_1/Thailand/2001/Den4_1	223	75	MAG	K/Q
1793_Den4/AY618992_1/Thailand/2001/Den4_1	556	186	RCC	A/T
1793_Den4/AY618992_1/Thailand/2001/Den4_1	586	196	RGT	G/S
1793_Den4/AY618992_1/Thailand/2001/Den4_1	613	205	YCA	P/S
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2875	959	YCG	P/S
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2943	982	AAN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2944	982	NNG	GAPFOUND

1793_Den4/AY618992_1/Thailand/2001/Den4_1	2942	981	NNG	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2976	993	ATN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2977	993	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2975	992	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2973	992	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2980	994	NTG	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2987	996	ANN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2986	996	ANN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2989	997	NGT	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2996	999	TNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2995	999	TNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3001	1001	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2999	1000	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	2997	1000	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3004	1002	NCC	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3073	1025	NTT	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3086	1029	ARC	N/S
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3095	1032	CNG	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3116	1039	GNG	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3144	1049	GAN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3159	1054	GAN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3160	1054	NNC	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3158	1053	NNC	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3206	1069	GNC	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3235	1079	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3233	1078	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3231	1078	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3238	1080	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3236	1079	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3234	1079	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3241	1081	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3239	1080	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3237	1080	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3244	1082	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3242	1081	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3240	1081	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3247	1083	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3245	1082	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3243	1082	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3250	1084	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3248	1083	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3246	1083	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3253	1085	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3251	1084	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3249	1084	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3256	1086	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3254	1085	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3252	1085	NNN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3316	1106	NGG	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3337	1113	NAT	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3341	1114	GNA	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3408	1137	ATN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3412	1138	NTG	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3493	1165	MCC	P/T
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3509	1170	ANT	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	3837	1280	TTN	GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1	6185	2062	ARG	K/R
1793_Den4/AY618992_1/Thailand/2001/Den4_1	6187	2063	RAR	E/K

```
1793_Den4/AY618992_1/Thailand/2001/Den4_1          6185          2062  RAR          E/K
1793_Den4/AY618992_1/Thailand/2001/Den4_1          6614          2205  TYT          F/S
1793_Den4/AY618992_1/Thailand/2001/Den4_1          6650          2217  ARA          K/R
1793_Den4/AY618992_1/Thailand/2001/Den4_1          8630          2877  ART          N/S
1793_Den4/AY618992_1/Thailand/2001/Den4_1          8844          2949  AAN          GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1          9938          3313  AYT          I/T
1793_Den4/AY618992_1/Thailand/2001/Den4_1          9941          3314  GRC          D/G
1793_Den4/AY618992_1/Thailand/2001/Den4_1         10015          3339  RTT          I/V
1793_Den4/AY618992_1/Thailand/2001/Den4_1         10087          3363  NGR          GAPFOUND
1793_Den4/AY618992_1/Thailand/2001/Den4_1         10085          3362  NGR          GAPFOUND
1901_Den4/AY618992_1/Thailand/2001/Den4_1            15             6  AAN          NON-CODING
1901_Den4/AY618992_1/Thailand/2001/Den4_1           111            38  TTN          GAPFOUND
1901_Den4/AY618992_1/Thailand/2001/Den4_1          2279           760  GYT          A/V
1901_Den4/AY618992_1/Thailand/2001/Den4_1          8798          2933  ARA          K/R
1901_Den4/AY618992_1/Thailand/2001/Den4_1         10195          3399  RAG          E/K
1901_Den4/AY618992_1/Thailand/2001/Den4_1         10366          3456  RGG          NON-CODING
1934_Den4/AY618992_1/Thailand/2001/Den4_1            15             6  AAN          NON-CODING
1934_Den4/AY618992_1/Thailand/2001/Den4_1           111            38  TTN          GAPFOUND
1934_Den4/AY618992_1/Thailand/2001/Den4_1           998           333  GMT          A/D
1934_Den4/AY618992_1/Thailand/2001/Den4_1          4515          1506  TTM          F/L
1934_Den4/AY618992_1/Thailand/2001/Den4_1          8798          2933  ARA          K/R
```

1.2.8 plot_muts

Usage

You can view the usage of degen_regions via:

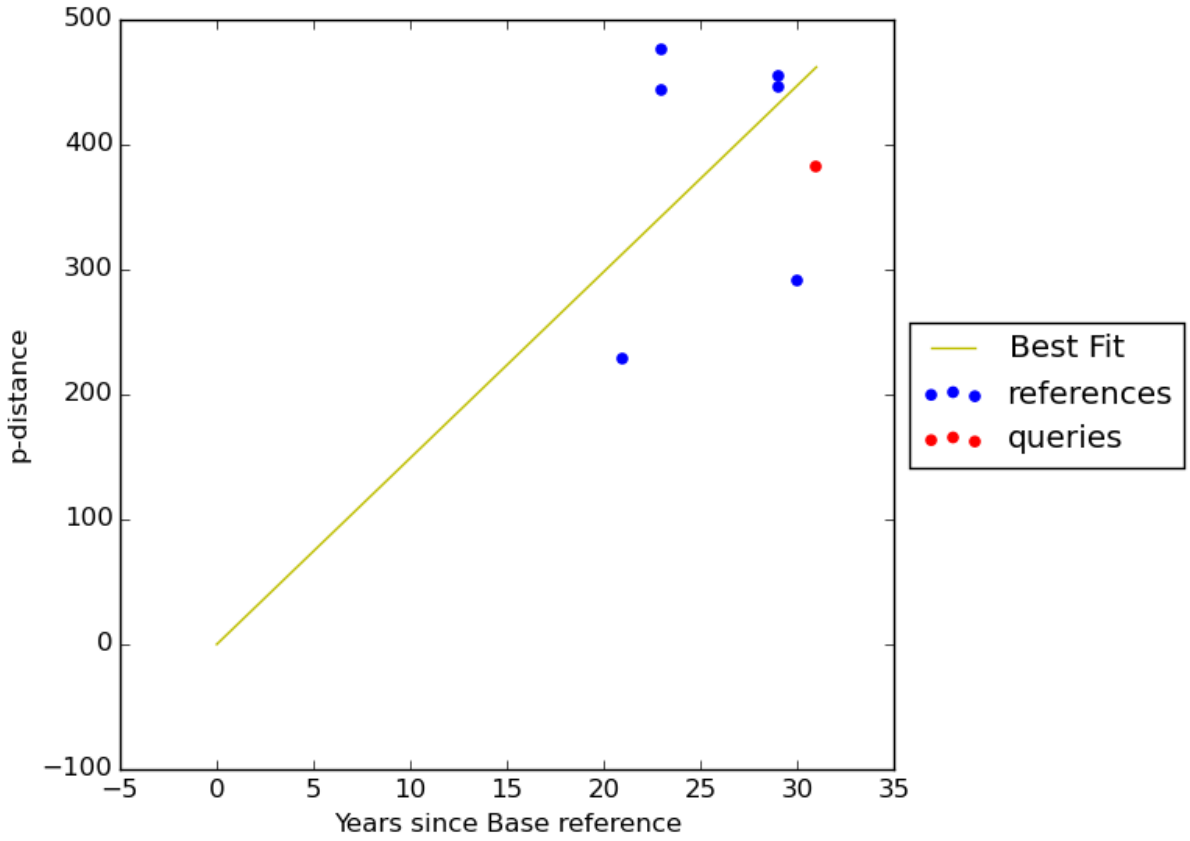
```
plot_muts --help
```

Example

```
plot_muts --refs tests/testinput/refs.fas --query tests/testinput/query.fas --out plot.png
```

The `--out` option is optional. If it is not provided, the plot will pop up on the user's screen automatically. If this does not work, try saving the image using `--out` instead.

Example Output



Input File Requirements

The input must be fasta format. Both the query and ref files can have any number of sequences.

The year should be the last part of the ID, preceded by a quadruple underscore. e.g.:

```
>some|info|blah_blah____2001_09_2010
>some____1995
>some____09/09/2012
```

If the ID uses '/' rather than underscore, plot_muts currently accepts the year as the *fourth* field. e.g.:

```
>some/info/blah/1995
>some/info/blah/1995/more/info
```

1.2.9 fasta

fasta is a very simple script to help mangle fasta files. Currently it only supports the ability to convert fasta files that have sequences that span multiple new lines into single lines.

Later on, it may be expanded more to include even more useful fasta features.

Usage

```
fasta --help
```

Examples

The following examples all use the test fasta file found under `tests/testinput/col.fasta`

```
>sequence1 some description !@#$%^&*()_+-=[]{}.,></?';:"
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
>sequence2
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

Read fasta input from standard input The following could output the fasta sequences as one line to your terminal(stdout) but reading from the pipe. This is useful if you want to use it in a pipeline.

```
$> cat tests/testinput/col.fasta | fasta -
```

Read fasta input from file The following could output the fasta sequences as one line to your terminal(stdout) as well, but reading straight from the file.

```
$> fasta tests/testinput/col.fasta
```

Simple shell pipeline using fasta The following is a simple shell pipeline to count how many A's there are in the sequence lines. There should be 160 since `col.fasta` is 80 characters per line and only the first line of each sequence has `A` and there are 2 sequences.

```
$> fasta tests/testinput/col.fasta | grep -v '>' | grep -Eo '[Aa]' | wc -l
160
```

1.3 AMOS

AMOS is a file format that is similar to any assembly file format such as ACE or SAM. It contains information about each read that is used to assemble each contig.

The format is broken into different message blocks. For the Ray assembler, it produces an AMOS file that is broken into 3 types of message blocks

- RED

```
{RED
iid:\d+
eid:\d+
seq:
[ATGC]+
.
```

```
qlt:
[A-Z]+
}
```

iid Integer identifier

eid Same as iid?

seq Sequence data

qlt Should be quality, but is only a series of D's from Ray assembler

- TLE

```
{TLE
src:\d+
off:\d+
clr:\d+,\d+
}
```

src RED iid that was used

off One would think offset, but unsure what it actually means

clr Not sure what this is either

- CTG

```
{CTG
iid:\d+
eid:\w+
com:
.*$
.
seq:
[ATGC]+
.
qlt:
[A-Z]+
.
{TLE
...
}
}
```

iid integer id of contig

eid contig name

com Communication software that generated this contig

seq Contig sequence data

qlt Supposed to be contig quality data, but for Ray it only produces D's

TLE 0 or more TLE blocks that represent RED sequences that compose the contig

1.3.1 Parsing

bio_bits contains an interface to parse a given file handle that has been opened on an AMOS file.

To read in the AMOS file you simply do the following

```
from bio_bits import amos
a = None
with open('AMOS.afg') as fh:
    a = amos.AMOS(fh)
```

CTG

To get information about the contigs(CTG) you can access the `.ctgs` attribute. The contigs are indexed based on their iid so to get the sequence of contig iid 1 you would do the following:

```
ctg = a.ctgs[1]
seq = ctg.seq
```

To retrieve all the reads(RED) that belong to a specific contig:

```
reads = []
for tle in ctg.tlelist:
    reads.append(a.reds[tle.src])
```

RED

To get information about the reads(RED) you can access the `.reds` attribute. The reds are indexed based on their iid so to get the sequence of red iid 1 you would do the following:

```
red = a.reds[1]
seq = red.seq
```

If you want to convert a RED entry into anything you can use the `.format` method. The `.format` method allows you to utilize any of the properties of a RED object such as `.iid`, `.eid`, `.seq`, `.qlt`. You can see in the examples below how to do this.

1.3.2 Examples

Here is an example of how to convert all RED blocks into a single fastq file

```
from bio_bits import amos

# Fastq format string
fastq_fmt = '@{iid}\n{seq}\n+\n{qlt}'

with open('amos.fastq','w') as fh_out:
    with open('AMOS.afg') as fh_in:
        for iid, red in amos.AMOS(fh_in).reds.items():
            fq = red.format(fastq_fmt)
            fh_out.write(fq + '\n')
```

1.4 CHANGELOG

1.4.1 Version 1.3.0

- Added fasta script that removes newlines from fasta sequences

1.4.2 Version 1.2.1

- Fixed some python3 and python2.6 incompatability issues
- Fixed some old bio_pieces references
- Added some simple tests for plot_muts

1.4.3 Version 1.2.0

- Renamed project to bio_bits to fix naming issue with other project
- GPL License added
- degen_regions script added
- parallel_blast added
- plot_muts script added

1.4.4 Version 1.1.0

- Renamed parse_contigs to group_references to better name functionality
- group_references now supports bam files

1.4.5 Version 1.0.0

- Version bump. Starting here we will employ semantic versioning
- Added version script to get version from project

1.4.6 Version 0.1.0

- Started project over to setup for Continuous Integration testing
- Added rename_fasta that can rename fasta sequence identifiers based on a input rename file
- Added travis, coveralls, readthedocs
- Added amos file parser that is specific to Ray assembler amos format
- Added format functionality for amos classes such that it is easy to convert to different formats
- Added amos2fastq to pull sequences out of AMOS files organized by their contigs.
- Added vcfcat.py, a commandline app for filtering and comparing vcf files.
- Completed documentation for vcfcat
- Added beast_checkpoint script and documentation
- Added beast_wrapper script that prints estimated time column in beast output
- Added beast_est_time script that allows you to easily get estimated time left from already running beast run

1.5 TODO

Todo

Could be possible to get beast_checkpoint to check for that scenario and use the last tree state that matches the last log state

(The original entry is located in /home/docs/checkouts/readthedocs.org/user_builds/bio-bits/checkouts/dev/docs/scripts/beast_checkpoint.rst, line 52.)

Indices and tables

- genindex
- modindex
- search