Working with gene features and genomes

Typical workflow when working with sequence data (e.g., genomes)

1. Download sequence data from NCBI
2. Save on local harddrive
3. Do analysis on locally stored sequence data
4. Repeat from step 3
   - to fix bugs
   - to run a different analysis

We can do the initial download in Biopython

```
# retrieve record from Entrez
handle = Entrez.efetch(db="nucleotide", id="KT220438", rettype="gb", \
                                                          retmode="text")
# read from the handle using regular Python, not SeqIO.read()!
gb_file_contents = handle.read()
handle.close()

# the entire genbank file is now in the variable `gb_file_contents`
print(gb_file_contents)
```

```
LOCUS       KT220438                1701 bp    cRNA    linear   VRL 20-JUL-2015
DEFINITION  Influenza A virus (A/NewJersey/NHRC_93219/2015(H3N2)) segment 4
            hemagglutinin (HA) gene, complete cds.
ACCESSION   KT220438
VERSION     KT220438.1  GI:887493048
KEYWORDS    .
SOURCE      Influenza A virus (A/New Jersey/NHRC_93219/2015(H3N2))
  ORGANISM  Influenza A virus (A/New Jersey/NHRC_93219/2015(H3N2))
            Viruses; ssRNA viruses; ssRNA negative-strand viruses;
            Orthomyxoviridae; Influenzavirus A.
REFERENCE   1  (bases 1 to 1701)
```

We can do the initial download in Biopython

```
# retrieve record from Entrez
handle = Entrez.efetch(db="nucleotide", id="KT220438", rettype="gb", \
                                                        retmode="text")
# read from the handle using regular Python, not SeqIO.read()!
gb_file_contents = handle.read()
handle.close()

# write out to disk
with open("KT220438.gb", "w") as out_handle:
    out_handle.write(gb_file_contents) # writes everything at once


# now we can read back in and process with Biopython
with open("KT220438.gb", "r") as in_handle:
    record = SeqIO.read(in_handle, format="gb") # use SeqIO to parse
```

Features in genbank files

```
FEATURES             Location/Qualifiers
     source          1..1701
                     /organism="Influenza A virus (A/New
                     Jersey/NHRC_93219/2015(H3N2))"
                     /mol_type="viral cRNA"
                     /strain="A/NewJersey/NHRC_93219/2015"
                     /serotype="H3N2"
                     /isolation_source="nasopharyngeal swab"
                     /host="Homo sapiens"
                     /db_xref="taxon:1682360"
                     /segment="4"
                     /lab_host="MDCK"
                     /country="USA: New Jersey"
                     /collection_date="17-Jan-2015"
     gene            1..1701
                     /gene="HA"
     CDS             1..1701
                     /gene="HA"
                     /function="receptor binding and fusion protein"
                     /codon_start=1
                     /product="hemagglutinin"
                     /protein_id="AKQ43545.1"
                     /db_xref="GI:887493049"
                     /translation="MKTIIALSYILCLVFAQKIPGNDNSTATLCLGHHAVPNGTIVKT
                     ITNDRIEVTNATELVQNSSIGEICDSPHQILDGENCTLIDALLGDPQCDGFQNKKWDL
                     FVERSKAYSNCYPYDVPDYASLRSLVASSGTLEFNNESFNWTGVTQNGTSSACIRRSS
                     SSFFSRLNWLTHLNYTYPALNVTMPNNEQFDKLYIWGVHHPGTDKDQIFLYAQSSGRI
                     TVSTKRSQQAVIPNIGSRPRIRDIPSRISIYWTIVKPGDILLINSTGNLIAPRGYFKI
                     RSGKSSIMRSDAPIGKCKSECITPNGSIPNDKPFQNVNRITYGACPRYVKHSTLKLAT
                     GMRNVPEKQTRGIFGAIAGFIENGWEGMVDGWYGFRHQNSEGRGQAADLKSTQAAIDQ
                     INGKLNRLIGKTNEKFHQIEKEFSEVEGRIQDLEKYVEDTKIDLWSYNAELLVALENQ
```

The source feature

```
source          1..1701
                /organism="Influenza A virus (A/New
                Jersey/NHRC_93219/2015(H3N2))"
                /mol_type="viral cRNA"
                /strain="A/NewJersey/NHRC_93219/2015"
                /serotype="H3N2"
                (...)
```

- identifies the biological source of the specified span of the sequence; mandatory
- more than one source key per sequence is allowed
- every entry/record will have, as a minimum, either a single source key spanning the entire sequence or multiple source keys, which together, span the entire sequence

The gene feature

```
gene            1..1701
                /gene="HA"
```

- region of biological interest identified as a gene and for which a name has been assigned

The CDS feature

```
CDS             1..1701
                /gene="HA"
                /function="receptor binding and fusion protein"
                /codon_start=1
                /product="hemagglutinin"
                (...)
```

- coding sequence
- sequence of nucleotides that corresponds with the sequence of amino acids in a protein (location includes stop codon)

The `mat_peptide` feature

```
mat_peptide     49..1035
                /gene="HA"
                /product="HA1"
```

- mature peptide or protein coding sequence
- coding sequence for the mature or final peptide or protein product following post-translational modification
- the location does not include the stop codon (unlike the corresponding CDS)

Influenza HA has one CDS but two mature peptides

```
CDS             1..1701
                /gene="HA"
                /function="receptor binding and fusion protein"
                /codon_start=1
                /product="hemagglutinin"
                /protein_id="AKQ43545.1"
                (...)

mat_peptide     49..1035
                /gene="HA"
                /product="HA1"

mat_peptide     1036..1698
                /gene="HA"
                /product="HA2"
```

Many more feature types exist

- `centromere`
- `exon`
- `gap`
- `LTR`
- `mRNA`
- `tRNA`

Full list with explanations at:

http://www.insdc.org/files/feature_table.html#7.2

Working with features in Biopython

Features are stored as a list in Biopython genbank records

```
In [1]: print(record.features)


Out[1]: [SeqFeature(FeatureLocation(ExactPosition(0),
ExactPosition(1701), strand=1), type='source'),
SeqFeature(FeatureLocation(ExactPosition(0),
ExactPosition(1701), strand=1), type='gene'),
SeqFeature(FeatureLocation(ExactPosition(0),
ExactPosition(1701), strand=1), type='CDS'),
SeqFeature(FeatureLocation(ExactPosition(48),
ExactPosition(1035), strand=1), type='mat_peptide'),
SeqFeature(FeatureLocation(ExactPosition(1035),
ExactPosition(1698), strand=1), type='mat_peptide')]
```

Features are stored as a list in Biopython genbank records

```
In [1]: for feature in record.features:
            print(feature)

Out[1]: type: source
        location: [0:1701](+)
        qualifiers:
            Key: collection_date, Value: ['17-Jan-2015']
            Key: country, Value: ['USA: New Jersey']
            Key: db_xref, Value: ['taxon:1682360']
            (...)

        type: gene
        location: [0:1701](+)
        qualifiers:
            Key: gene, Value: ['HA']

        type: CDS
        location: [0:1701](+)
        qualifiers:
            Key: codon_start, Value: ['1']
            Key: db_xref, Value: ['GI:887493049']
```

Features have member variables `type`, `location`, and `qualifiers`

```
In [1]: print("Type:", feature.type)
        print("\nLocation:", feature.location)
        print("\nQualifiers:", feature.qualifiers)

Out[1]: Type: source

        Location: [0:1701](+)

        Qualifiers: {'organism': ['Influenza A virus
(A/New Jersey/NHRC_93219/2015(H3N2))'], 'lab_host':
['MDCK'], 'strain': ['A/NewJersey/NHRC_93219/2015'],
'db_xref': ['taxon:1682360'], 'host': ['Homo sapiens'],
'segment': ['4'], 'isolation_source': ['nasopharyngeal
swab'], 'collection_date': ['17-Jan-2015'], 'mol_type':
['viral cRNA'], 'serotype': ['H3N2'], 'country': ['USA:
New Jersey']}
```

The `qualifiers` are stored as a dictionary

```
In [1]: print(feature.qualifiers["organism"])

Out[1]: ['Influenza A virus (A/New
Jersey/NHRC_93219/2015(H3N2))']
```

Note: The `qualifiers` dictionary returns a list! We need to extract the first element of the list if we want to just get the contents string.

```
In [2]: organism = feature.qualifiers["organism"][0]
        print(organism)

Out[2]: Influenza A virus (A/New
Jersey/NHRC_93219/2015(H3N2))
```