Bioinformatic methods for CoBV ICTV Taxonomy Proposal (Whispovirus_1nsp)

This document describres the bioinformatic codes used to analyze the genome sequences of Chionoecetes opilio bacilliform virus (CoBV) and other nimaviruses in the ICTV Taxonomy Proposal (Whispovirus_1nsp) for creating a new species, "*Whispovirus lacteolymphae*", in genus *Whispovirus*.

Genome diagram of CoBV

Genome diagrams were drawn using gbdraw v0.2.0.

Circular genome diagram

Circular genome diagram generation (gbdraw)

```
micromamba activate gbdraw-0.2.0
gbdraw circular \
-i LC741431.gb -f svg \
--track_type middle \
--show_labels --separate_strands \
--block_stroke_width 1 \
--block_stroke_color gray \
-t custom_color_table.tsv -d modified_default_colors.tsv
micromamba deactivate
```

modified_default_colors.tsv

```
CDS #d3d3d3
```

custom_color_table.tsv

```
CDS protein_id  WKY18398.1  yellow  Horizontally transferred genes
CDS protein_id  WKY18439.1  yellow  Horizontally transferred genes
CDS protein_id  WKY18438.1  yellow  Horizontally transferred genes
CDS protein_id  WKY18437.1  yellow  Horizontally transferred genes

CDS protein_id  BDU62120.1  yellow  Horizontally transferred genes
CDS protein_id  BDU62121.1  yellow  Horizontally transferred genes
CDS protein_id  BDU62122.1  yellow  Horizontally transferred genes
CDS protein_id  BDU62187.1  yellow  Horizontally transferred genes

CDS product wsv.*-like protein  #47b8f8 WSSV-like proteins
CDS note    WSV.*-like protein  #47b8f8 WSSV-like proteins
```

Linear genome diagram

BLASTN comparsion (CoBV-CbBV)

In order to adjust the coordinates of the two circular genome sequences, the genome sequence of Chionoecetes bairdi bacilliform virus (CbBV; GenBank Accession no. OQ911497.1) was rotated counterclockwise using SnapGene Viewer v8.1 so that the new 5′ position corresponds to nucleotide 63,243 of the original record, yielding `CbBV_rotated.fa` and `CbBV_rotated.gbk`.

```
micromamba activate blast-2.16.0
blastn -query LC741431.fasta -subject CbBV_rotated.fa -outfmt 7 -out
LC741431.CbBV_rotated.blastn.out
micromamba deactivate
```

Linear genome diagram generation (gbdraw)

```
micromamba activate gbdraw-0.2.0

gbdraw linear \
-i LC741431.gb CbBV_rotated.gbk \
-b LC741431.CbBV_rotated.blastn.out \
-o CoBV_CbBV \
-f svg \
--block_stroke_width 1 \
--block_stroke_color gray \
--separate_strands --align_center \
-t custom_color_table.tsv -d modified_default_colors.tsv \
--bitscore 1000 --evalue 1e-30 --identity 90

micromamba deactivate
```

Average amino acid identity (AAI) between CoBV and WSSV

AAI was calculated with `aai.rb` script in the Enveomics collection (Commit 8660352) using BLAST+ v2.16.0 as the aligner.

AAI calculation (aai.rb)

```
micromamba activate blast-2.16.0
~/enveomics/Scripts/aai.rb -1 ../in_faa/CoBV.faa -2 ../in_faa/WSSV.faa &>aai.log
micromamba deactivate
```

aai.log

```
Temporal directory: /tmp/d20250613-11993-ddcv2i.
Creating databases.
  Reading FastA file: ../in_faa/CoBV.faa
    File contains 105 sequences.
  Reading FastA file: ../in_faa/WSSV.faa
    File contains 177 sequences.
Running one-way comparisons.
Warning: [blastp] Examining 5 or more matches is recommended
! One-way AAI 1: 34.67% (SD: 9.45%), from 103 proteins.
Warning: [blastp] Examining 5 or more matches is recommended
! One-way AAI 2: 33.78% (SD: 9.86%), from 177 proteins.
! Two-way AAI  : 34.74% (SD: 8.62%), from 78 proteins.
```

Ortholog clustering and phylogenetic analysis

Ortholog clustering

Ortholog analysis was performed using OrthoFinder v2.5.5.

```
cd ~/study/
mkdir 2025-05-26_nimaviridae
cd 2025-05-26_nimaviridae
mkdir analysis scripts
cd analysis/

cd in_gbk/
ls -1|grep ".gb"|sed 's/.gb//g' >in.list
cd ../

# Protein FASTA generation from GenBank files
cat ./in_gbk/in.list |while read line;do
  ../scripts/gb2faa.py -i ./in_gbk/${line}.gb -o ./in_faa/${line}.faa
done

micromamba activate orthofinder-2.5.5
orthofinder -f ./in_faa/
micromamba deactivate

../scripts/extract_fasta_by_orthogroup.py \
-t 2025-05-26_orthogroups_single_copy.tsv \
-i ./in_faa/ -d single_copy_orthologs/

mkdir single_copy_orthologs_renamed
../scripts/rename_single_copy_orthologs.py \
-i single_copy_orthologs \
-o single_copy_orthologs_renamed/ \
-t 2025-05-26_orthogroups_single_copy_mod.csv \
-s single_copy_orthologs.txt
```

gb2faa.py

```
#!/usr/bin/env python
# coding: utf-8

import sys
import os
import argparse
from Bio import SeqIO
from Bio.Seq import Seq
from Bio.SeqRecord import SeqRecord
from Bio.SeqFeature import SeqFeature

def _get_args():
    parser = argparse.ArgumentParser(
        description='Extract protein sequences from a genbank file')
    parser.add_argument(
        '-i',
        '--input',
        help='GenBank flat file format of the genomic sequence(s) (required)',
        type=str,
        required=True)
    parser.add_argument(
        '-o',
        '--output',
        help='output fasta file (default: out.faa)',
        type=str,
        default="out.faa")
    if len(sys.argv) == 1:
        parser.print_help(sys.stderr)
        sys.exit(1)
    args = parser.parse_args()
    return args

def main():
    args = _get_args()
    in_gb = args.input
    out_faa = args.output
    records = SeqIO.parse(in_gb, 'genbank')
    out_records = []
    for record in records:
        for feature in record.features:
            if feature.type == 'CDS':
                if 'protein_id' in feature.qualifiers.keys():
                    record_id = feature.qualifiers['protein_id'][0]
                else:
                    record_id = "{}_{}-
{}".format(record.id,feature.location.start,feature.location.end)
                out_record = SeqRecord(
                Seq(
                        feature.qualifiers['translation'][0]),
                    id=record_id,
```

```
                    description="{} [{}]".format(
                        feature.qualifiers['product'][0],
                        record.annotations['organism']))
                out_records.append(out_record)
    with open(out_faa, "w") as output_handle:
        for out_record in out_records:
            SeqIO.write(out_record, output_handle, "fasta")

if __name__ == "__main__":
    main()
```

extract_fasta_by_orthogroup.py

```
#!/usr/bin/env python
# coding: utf-8

import sys
import os
import argparse
import pandas as pd
from Bio import SeqIO

def _get_args():
    parser = argparse.ArgumentParser(description='Extract per-Orthogroup FASTA
files from multiple species FASTA using an orthogroup table.')
    parser.add_argument("-t", "--tsv", type=str, required=True,
                        help="Orthogroup matrix TSV file")
    parser.add_argument("-i", "--indir", type=str, required=True,
                        help="Directory containing input FASTA files (e.g.,
Species1.faa, Species2.faa, ...)")
    parser.add_argument("-d", "--outdir", type=str, required=True,
                        help="Directory to write per-Orthogroup FASTA files")
    parser.add_argument("--ext", type=str, default=".faa",
                        help="File extension of input FASTA files (default:
.faa)")
    if len(sys.argv) == 1:
        parser.print_help(sys.stderr)
        sys.exit(1)
    return parser.parse_args()

def load_fasta_index(fasta_path):
    return {record.id: record for record in SeqIO.parse(fasta_path, "fasta")}

def main():
    args = _get_args()
    df = pd.read_csv(args.tsv, sep="\t")
    os.makedirs(args.outdir, exist_ok=True)

    # Load all species FASTA files
    fasta_db = {}
    for column in df.columns[1:]:
```

```
        faa_file = os.path.join(args.indir, f"{column}{args.ext}")
        if not os.path.isfile(faa_file):
            print(f"WARNING: {faa_file} not found. Skipping.")
            continue
        fasta_db[column] = load_fasta_index(faa_file)

    print(f"Loaded {len(fasta_db)} FASTA files")

    for _, row in df.iterrows():
        orthogroup_id = row[0]
        records = []

        for column in df.columns[1:]:
            if column not in fasta_db:
                continue
            cell = row[column]
            if pd.isna(cell):
                continue
            ids = [x.strip() for x in str(cell).split(",") if x.strip()]
            for seq_id in ids:
                record = fasta_db[column].get(seq_id)
                if record:
                    records.append(record)
                else:
                    print(f"ID '{seq_id}' not found in {column}{args.ext}")

        if records:
            out_path = os.path.join(args.outdir, f"{orthogroup_id}.faa")
            with open(out_path, "w") as out_fh:
                SeqIO.write(records, out_fh, "fasta")

    print(f"Done. Per-orthogroup FASTA files saved to: {args.outdir}")

if __name__ == '__main__':
    main()
```

rename_single_copy_orthologs.py

```
#!/usr/bin/env python
# coding: utf-8

import sys
import argparse
import pandas as pd
from Bio import SeqIO

def _get_args():
    parser = argparse.ArgumentParser(description='Rename single-copy orthologue
sequences identified by OrthoFinder2')
    parser.add_argument("--input","-i", "--in",   metavar="DIR",
help="Single_Copy_Orthologue_Sequences directory", required=True)
```

```
    parser.add_argument("--output","-o", "--out",metavar="DIR",help="output
directory", required=True)
    parser.add_argument("-t", "--orthogroups", type=str ,help="Orthogroups.tsv",
required=True)
    parser.add_argument("-s", "--single_copy_orthologues", type=str
,help="Orthogroups_SingleCopyOrthologues.txt", required=True)
    if len(sys.argv) == 1:
        parser.print_help(sys.stderr)
        sys.exit(1)
    args = parser.parse_args()
    return args

def rename_fa(in_dir, out_dir, og_id, og_table):
    in_fa  = '{}/{}.faa'.format(in_dir,og_id)
    out_fa = '{}/{}.faa'.format(out_dir,og_id)
    with open(in_fa, 'r') as infh, open(out_fa, 'w') as outfh :
        records = SeqIO.parse(infh,'fasta')
        for r in records:
            r_name = r.id
            sp_name = og_table.loc[: , (og_table ==
r_name).any()].columns.values[0]
            r.id = sp_name
            r.description = ''
            SeqIO.write(r, outfh, 'fasta')

def main():
    args = _get_args()
    in_dir = args.input
    out_dir = args.output
    orthogroups = args.orthogroups
    single_copy_orthologues = args.single_copy_orthologues
    df_og = pd.read_table(orthogroups)
    with open(single_copy_orthologues) as sog:
        OGs = sog.read().splitlines()
        for OG in OGs:
            rename_fa(in_dir, out_dir, OG, df_og)

if __name__ == '__main__':
    main()
```

Phylogenetic analysis of nimaviral genomes

```
#!/bin/bash
# >>>>>>
# nimaviridae_phylogeny.sh
# >>>>>>

source /home/kawato/micromamba/etc/profile.d/mamba.sh

# Create/goto WDIR
WHOME=/home/kawato/study/2025-05-26_nimaviridae/
```

```
cd ${WHOME}/analysis/

# Set VARs
NUMTH=128

mkdir faa_aligned faa_aligned_trimmed

micromamba activate mafft-7.525

while read IN;do
mafft --maxiterate 1000 --globalpair --thread ${NUMTH}
single_copy_orthologs_renamed/${IN}.faa >faa_aligned/${IN}.aligned.faa
2>faa_aligned/${IN}.mafft.log
done <single_copy_orthologs.txt

micromamba deactivate

micromamba activate trimal-1.5.0

while read IN;do \
trimal -in faa_aligned/${IN}.aligned.faa -out
faa_aligned_trimmed/${IN}.aligned.trimmed.faa -automated1
2>faa_aligned/${IN}.trimal.err
done <single_copy_orthologs.txt

micromamba deactivate

micromamba activate iqtree-2.3.6

RUN=2025-05-28_iqtree
iqtree -s faa_aligned_trimmed/ -T ${NUMTH} -m MFP+MERGE -mset LG,WAG,JTT -B 1000 -
-prefix ${RUN}

micromamba deactivate
```

Jaccard clustering of niamviral genomes

```
import pandas as pd
import numpy as np
from scipy.spatial.distance import pdist, squareform
from scipy.cluster.hierarchy import linkage, leaves_list, dendrogram
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec

# ---------- Load XSLX ----------
# Ensure the file "Whispovirus_1nsp_File_S2.xlsx" is in the same directory
# or provide the full path.
file_path = "Whispovirus_1nsp_File_S2.xlsx"
try:
    df = pd.read_excel(file_path, header=2, index_col=0)
```

```
except FileNotFoundError:
    print(f"Error: The file '{file_path}' was not found.")
    print("Please make sure the xlsx file exists in the correct location.")


# ---------- Prepare binary presence/absence matrix ----------
meta_cols    = {"Orthogroup", "Description"}
genome_cols  = [c for c in df.columns if c not in meta_cols]

if not genome_cols:
    print("Error: No genome columns found in the TSV file.")
    print("Please check the column names and file format.")
    exit()

binary_matrix = df[genome_cols].notnull().astype(int)

# ---------- Compute Jaccard distance & clustering ----------
if binary_matrix.shape[1] < 2:
    print(f"Error: Need at least two genomes to calculate Jaccard distance and
cluster. Found {binary_matrix.shape[1]}.")
    exit()

dist_condensed   = pdist(binary_matrix.T, metric="jaccard")
linkage_matrix   = linkage(dist_condensed, method="average")

# Reorder by dendrogram leaves
leaves           = leaves_list(linkage_matrix)
sim_matrix       = 1 - squareform(dist_condensed) # Convert distance to similarity
sim_ordered      = sim_matrix[leaves][:, leaves]
labels_ordered   = np.array(genome_cols)[leaves]
n                = len(labels_ordered)

# ---------- Plot Dendrogram and Heatmap Together ----------
fig = plt.figure(figsize=(10, 11))

# Define GridSpec: 2 rows (dendrogram, heatmap), 2 columns (heatmap area, colorbar
area)
# Dendrogram height is smaller, colorbar width is smaller.
gs = gridspec.GridSpec(nrows=2, ncols=2,
                       figure=fig,
                       height_ratios=[0.2, 1],  # Dendrogram height is 0.2 of
heatmap height
                       width_ratios=[1, 0.05], # Colorbar width is 0.05 of heatmap
width
                       wspace=0.05, hspace=0.02) # Adjust spacing

ax_dendro = fig.add_subplot(gs[0, 0])
ax_heatmap = fig.add_subplot(gs[1, 0])
ax_colorbar = fig.add_subplot(gs[1, 1])

# 1. Plot Dendrogram on top
# The `dendrogram` function will use the leaf order determined by the linkage.
# `labels_ordered` is derived from `leaves_list`, which is consistent.
ddata = dendrogram(
```

```
    linkage_matrix,
    ax=ax_dendro,
    orientation='top',
    distance_sort='ascending', # Sort branches by distance
    no_labels=True # Hide labels on dendrogram itself, they will be on heatmap
)
ax_dendro.set_ylabel('Distance')
# Remove spines for a cleaner look, except the left one for the distance axis
ax_dendro.spines['top'].set_visible(False)
ax_dendro.spines['right'].set_visible(False)
ax_dendro.spines['bottom'].set_visible(False)
ax_dendro.spines['left'].set_visible(True) # Keep this if you want the y-axis line

# 2. Plot Heatmap using pcolormesh
# Create a coordinate grid for pcolormesh: 0..n along both axes
X, Y = np.meshgrid(np.arange(n + 1), np.arange(n + 1))

pcm = ax_heatmap.pcolormesh(
    X, Y,
    sim_ordered,
    cmap="viridis",
    shading="flat"
)

# Set tick positions at the center of each cell
ax_heatmap.set_xticks(np.arange(n) + 0.5)
ax_heatmap.set_yticks(np.arange(n) + 0.5)

# Label ticks with the reordered genome names
ax_heatmap.set_xticklabels(labels_ordered, rotation=90, fontsize=8)
ax_heatmap.set_yticklabels(labels_ordered, fontsize=8)

# Invert y-axis so that the top row corresponds to index 0 (matrix convention)
ax_heatmap.invert_yaxis()
ax_heatmap.set_aspect("equal") # Ensure square cells

# Edge color for heatmap cells (currently set to no visible lines by
linewidth=0.0)
pcm.set_edgecolor("black")
pcm.set_linewidth(0.0) # Set to >0 (e.g., 0.5) and a color (e.g. 'white' or
'gray') for visible grid lines

# Title for the heatmap
ax_heatmap.set_title("Jaccard similarity heatmap based on orthogroup contents",
y=1.21) # y to adjust title position slightly

# 3. Colorbar
cbar = fig.colorbar(pcm, cax=ax_colorbar) # Use cax for specific colorbar axis
cbar.set_label("Jaccard similarity", rotation=270, labelpad=15)

# Save as SVG and PNG
output_filename_base = "jaccard_clustermap"
fig.savefig(f"{output_filename_base}.svg", format="svg", bbox_inches='tight')
fig.savefig(f"{output_filename_base}.png", dpi=150, bbox_inches='tight')
```

```
plt.show()
```

Comparison of closely related nimaviral MAGs

Average nucleotide identity (ANI) between closely related nimaviral MAG sequences

ANI was calculated with `ani.rb` script in the Enveomics collection (Commit 8660352)using BLAST+ v2.16.0 as the aligner.

```
micromamba activate blast-2.16.0

# Penaeid endogenous nimaviruses (1) ("Majaniviruses")
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738868.fasta -2 ./in_fna/LC738874.fasta -
o MjeNMV_MlMJNV.ani.out &>MjeNMV_MlMJNV.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738868.fasta -2 ./in_fna/LC738870.fasta -
o MjeNMV_PemoMJNVA.ani.out &>MjeNMV_PemoMJNVA.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738868.fasta -2 ./in_fna/LC738871.fasta -
o MjeNMV_PemoMJNVB.ani.out &>MjeNMV_PemoMJNVB.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738868.fasta -2 ./in_fna/LC738873.fasta -
o MjeNMV_PeseMJNV.ani.out  &>MjeNMV_PeseMJNV.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738874.fasta -2 ./in_fna/LC738870.fasta -
o MlMJNV_PemoMJNVA.ani.out &>MlMJNV_PemoMJNVA.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738874.fasta -2 ./in_fna/LC738871.fasta -
o MlMJNV_PemoMJNVB.ani.out &>MlMJNV_PemoMJNVB.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738874.fasta -2 ./in_fna/LC738873.fasta -
o MlMJNV_PeseMJNV.ani.out  &>MlMJNV_PeseMJNV.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738870.fasta -2 ./in_fna/LC738871.fasta -
o PemoMJNVA_PemoMJNVB.ani.out &>PemoMJNVA_PemoMJNVB.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738870.fasta -2 ./in_fna/LC738873.fasta -
o PemoMJNVA_PeseMJNV.ani.out &>PemoMJNVA_PeseMJNV.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738873.fasta -2 ./in_fna/LC738871.fasta -
o PeseMJNV_PemoMJNVB.ani.out &>PeseMJNV_PemoMJNVB.ani.log

# Penaeid endogenous nimaviruses (2) ("Pemoniviruses")
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738869.fasta -2 ./in_fna/AP027152.fasta -
o PmeNMV_PesePMNV.ani.out &>PmeNMV_PesePMNV.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738869.fasta -2 ./in_fna/AP027202.fasta -
o PmeNMV_MjPMNV.ani.out &>PmeNMV_MjPMNV.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738869.fasta -2 ./in_fna/LC738875.fasta -
o PmeNMV_MlPMNV.ani.out &>PmeNMV_MlPMNV.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/AP027152.fasta -2 ./in_fna/AP027202.fasta -
o PesePMNV_MjPMNV.ani.out &>PesePMNV_MjPMNV.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/AP027152.fasta -2 ./in_fna/LC738875.fasta -
o PesePMNV_MlPMNV.ani.out &>PesePMNV_MlPMNV.ani.log
~/enveomics/Scripts/ani.rb -1 ./in_fna/AP027202.fasta -2 ./in_fna/LC738875.fasta -
o MjPMNV_MlPMNV.ani.out &>MjPMNV_MlPMNV.ani.log

# Sesarmid nimaviruses
~/enveomics/Scripts/ani.rb -1 ./in_fna/LC738884.fasta -2 ./in_fna/AP027155.fasta -
o SiNMV_CdNMV.ani.out &>SiNMV_CdNMV.ani.log
```

```
micromamba deactivate
```

Penaeid endogenous nimaviruses (1)

```
micromamba activate blast-2.16.0
cd ./in_fna/
# MjeNMV-MelaMJNV
blastn -query LC738868.fasta -subject LC738874.fasta -task blastn -outfmt 7 -out
MjeNMV.MelaMJNV.blastn.out
# MelaMJNV-PemoMJNVA
blastn -query LC738874.fasta -subject LC738870.fasta -task blastn -outfmt 7 -out
MelaMJNV.PemoMJNVA.blastn.out
# PemoMJNVA-PeseMJNV
blastn -query LC738870.fasta -subject LC738873.fasta -task blastn -outfmt 7 -out
PemoMJNVA.PeseMJNV.blastn.out
# PeseMJNV-PemoMJNVB
blastn -query LC738873.fasta -subject LC738871.fasta -task blastn -outfmt 7 -out
PeseMJNV.PemoMJNVB.blastn.out

umamba deactivate

cd ..
micromamba activate gbdraw-0.2.0

gbdraw linear \
-i \
./in_gbk/MjeNMV.gb \
./in_gbk/MlMJNV.gb \
./in_gbk/PemoMJNVA.gb \
./in_gbk/PeseMJNV.gb \
./in_gbk/PemoMJNVB.gb \
-b \
./in_fna/MjeNMV.MelaMJNV.blastn.out \
./in_fna/MelaMJNV.PemoMJNVA.blastn.out \
./in_fna/PemoMJNVA.PeseMJNV.blastn.out \
./in_fna/PeseMJNV.PemoMJNVB.blastn.out \
-o majani_ANI \
-f svg,png \
--block_stroke_width 1 \
--block_stroke_color gray \
--separate_strands --align_center \
-t majani_custom_color_table.tsv -d modified_default_colors.tsv \
--bitscore 1000 --evalue 1e-20 --identity 70
```

Penaeid endogenous nimaviruses (2)

```
micromamba activate blast-2.16.0
cd ./in_fna/
# PmeNMV-PesePMNV
blastn -query LC738869.fasta -subject AP027152.fasta -task blastn -outfmt 7 -out
PmeNMV.PesePMNV.blastn.out
# PesePMNV-PmeNMV
blastn -query AP027152.fasta -subject LC738869.fasta -task blastn -outfmt 7 -out
PesePMNV.PmeNMV.blastn.out
# PmeNMV-MjPMNV
blastn -query LC738869.fasta -subject AP027202.fasta -task blastn -outfmt 7 -out
PmeNMV.MjPMNV.blastn.out
# PesePMNV-MjPMNV
blastn -query AP027152.fasta -subject AP027202.fasta -task blastn -outfmt 7 -out
PesePMNV.MjPMNV.blastn.out
# MjPMNV-MlPMNV
blastn -query AP027202.fasta -subject LC738875.fasta -task blastn -outfmt 7 -out
MjPMNV.MlPMNV.blastn.out


umamba deactivate

cd ..
micromamba activate gbdraw-0.2.0

gbdraw linear \
-i \
./in_gbk/PesePMNV.gb \
./in_gbk/PmeNMV.gb \
./in_gbk/MjPMNV.gb \
./in_gbk/MlPMNV.gb \
-b \
./in_fna/PesePMNV.PmeNMV.blastn.out \
./in_fna/PmeNMV.MjPMNV.blastn.out \
./in_fna/MjPMNV.MlPMNV.blastn.out \
-o pemoni_ANI \
-f svg,png \
--block_stroke_width 1 \
--block_stroke_color gray \
--separate_strands --align_center \
-t majani_custom_color_table.tsv -d modified_default_colors.tsv \
--bitscore 1000 --evalue 1e-20 --identity 70
umamba deactivate
```

Sesarmid nimaviruses

```
micromamba activate blast-2.16.0
cd ./in_fna/
# SiNMV-ChdeNMV
blastn -query LC738884.fasta -subject AP027155.fasta -task blastn -outfmt 7 -out
SiNMV.ChdeNMV.blastn.out
```

```
umamba deactivate

cd ..
micromamba activate gbdraw-0.2.0

gbdraw linear \
-i \
./in_gbk/SiNMV.gb \
./in_gbk/ChdeNMV.gb \
-b \
./in_fna/SiNMV.ChdeNMV.blastn.out \
-o SiNMV_ChdeNMV_ANI \
-f svg,png \
--block_stroke_width 1 \
--block_stroke_color gray \
--separate_strands --align_center \
-t SiNMV_custom_color_table.tsv -d modified_default_colors.tsv \
-k CDS,tRNA,rRNA \
--bitscore 1000 --evalue 1e-20 --identity 70
umamba deactivate
```

modified_default_colors.tsv

```
CDS #d3d3d3
```

majani_custom_color_table.tsv

```
CDS product wsv.*-like protein  #47b8f8 WSSV-like proteins
CDS product baculoviral IAP repeat-containing protein   yellow  BIRP
CDS product tyrosine recombinase    red tyrosine recombinase
```

SiNMV_custom_color_table.tsv

```
CDS product wsv.*-like protein  #47b8f8 WSSV-like proteins
CDS product integrase-like protein  red retroviral integrase
CDS product retroviral integrase    red retroviral integrase
```