Corey Work

Extracting provirus sequences

1. changing all instances of | to -
2. Running code to separate provirus sequences

```
for file in "${work_dir}"/*provirus.fna;
do
while IFS= read -r line
do
if [[ ${line:0:1} == '>' ]]
then
outfile="split_fastas/${line#>}.fasta"
echo "$line" > "$outfile"
else
echo "$line" >> "$outfile"
fi
done < "$file"
done
```

3. Only using 'provirus' sequences
4. Some sequences have the exact same coordinates, need to check for duplicates

Sourmash

Version: 4.8.5

1. Running sourmash on sequences

```
sourmash sketch dna -p k=31,scaled=1000 *fasta
```

2. Comparing signatures

```
sourmash compare --ksize 21 --csv
CoreyProvirusSourmash_k21Scaled1000_JaccardIndex_23Jan24.csv -o
CoreyProvirusSourmash_k21Scaled1000_JaccardIndex_23Jan24.binary *sig
```

3. Getting plots

```
sourmash plot --pdf
CoreyProvirusSourmash_k21Scaled1000_JaccardIndex_23Jan24.binary
```

Manipulating Sourmash Output

1. Running 5.CreateEdgetable.py

- this changes the sourmash csv output from a matrix to a network edgetable (Source,Target,Connection format)

```
python3 ./../moon/5.CreateEdgetable.py -i
CoreyProvirusSourmash_k21Scaled1000_JaccardIndex_23Jan24.csv
```

2. Running 6.EdgetableVariations.py
- creates an edgetable with connections of *x* value or higher

```
python3 ./../moon/6.EdgetableVariations.py -i
CoreyProvirusSourmash_k21Scaled1000_JaccardIndex_23Jan24_Edgetable_2024-01-23_13-
59-22.csv
```

3. Running 8.FindElbow.py
- takes all edgetable variations, finds the number of communities for each one, calculates cumulative area under the curve for the number of communities as higher weight edges are removed

```
python3 ./../../moon/8.FindElbow.py
```

Finding AUC Curve Elbow

1. Take output from 8.FindElbow.py (ComsAUCByIteration_2024-01-23_14-16-16.csv)

```
# R
> install.packages("pathviewr")
> library(pathviewr)

> data <- read.csv("ComsAUCByIteration_2024-01-23_14-16-16.csv")
> find_curve_elbow(data, export_type="row_num", plot_curve=TRUE)
[1] 41

# Elbow of the curve == 41.csv
# Edgetable of connections with weight 0.41 or higher
```

Finding Elbow Edgetable Communities

1. Running 9.GetCommunityIDs.py
- takes the elbow edgetable, finds all communities within the network, outputs communties and IDs within them

```
python3 ./../moon/9.GetCommunityIDs.py -i 41.csv > AllCommunities_and_IDs.txt
```

2. Manipulating output to get usable format

```
# Changes .txt tp .csv format
cat AllCommunities_and_IDs.txt | sed 's/Community /Community_/g' | sed 's/\: \
[/,/g' | sed 's/ //g' | sed 's/\]//g' | sed "s/'//g" > AllCommunities.csv
```

```
# Splits into separate files containing each communities IDs
for i in `cat AllCommunities.csv`; do echo ${i#*,} > ${i%%,*}.csv; done
# Removes "Community_*" from the start of the line
for i in Community_*.csv; do sed -i 's/[^,]*,//' $i; done
# Changes commas in all files to newlines
for i in *; do sed -i 's/,/\n/g' $i; done
# Creates a file of every provirus ID and it's community
for i in *; do for j in `cat $i`; do echo "$j,${i%.csv}"; done; done >>
IDsandCommunities.csv
```

3. Added "Genome,Community" to the top of IDsandCommunities.csv to make it compatible with 10.CutOffInterEdges.py
4. Running 10.CutOffInterEdges.py

- removes edges connected nodes from different communites (inter-community edges), leaving only intra-community edges

```
# Had to run this, some reason ",IDsandCommunities" keeps getting added to the
Community/ID list
sed -i 's/,IDsandCommunities//g' IDsandCommunities.csv
```

```
python3 ./../moon/10.CutOffInterEdges.py -i 41.csv -c IDsandCommunities.csv
```