

# 1 Genome-visualization.py: A Comprehensive Guide

## 1.1 Visualizations

Genome-visualization.py is a python command line tool within the KREPE package that allows bioinformaticists to count the kmers with a .fasta, .fna, or .fastq file and lets the user create bar graphs that represent all of the kmers at length k present in a file along with their abundance. It also provides a simple uncompressed De Bruijn graph that is ideal for smaller values of k and is a nice visualization of the genome.

## 1.2 Installation

Genome-visualization is a one of KREPE's many python command line tools for bioinformaticists. Genome-visualization makes use of the powerful sourmash python library to make basic k-mer analysis easier than ever. There are two ways to install KREPE.

- pip install krepe
- git clone <https://RGBwalnut/KREPE.git>

```
$ python3 $PATH/genome-visualization.py (desired k-mer length) (file name)
(bar graph) (de bruijn graph) (de bruijn meta data)
```

## 1.3 Usage

If it was installed with pip you can use:

```
$ genome-visualization (k-mer length) (file) (plot_distribution) (plot_de_bruijn)
(file_type) (q, v) (meta_data)
```

Or if it was installed from github you can use:

```
$ python3 PATH/genome-visualization.py (k-mer length) (file) (plot_distribution)
(plot_de_bruijn) (file_type) (q, v) (meta_data(only if plot de bruijn =
True))
```

### 1.3.1 Argument Breakdown

The first argument is the an integer or the kmer length then followed by your input file. The plot\_distribution argument has two option not\_plot\_distribution and plot\_distribution. This will either plot an abundance graph or not. Then there is plot\_de\_bruijn which has two options: plot\_de\_bruijn and not\_plot\_de\_bruijn. If you plot\_de\_bruijn you will need to provide a meta data file as the last argument that has:

- Line 1: Width
- Line 2: Height
- Line 3: Circle Size
- Line 4: Stroke Width

The `q`, `v` argument will either be `-q` which will not print the dictionary of k-mers or the `-v` argument which will.

## 1.4 Tutorial

If you are using our pip package you can simply run the program with the `Tuberculosis.fna` in the same directory.

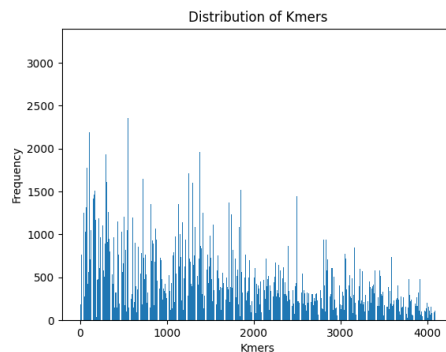
- `$ genome-visualization 6 lactobacillicus.fna plot_distribution plot_de_bruijn -fna -q meta-data.txt`

But if you are using the github version of our program you will need to `cd` into the comparison tutorial directory and give the absolute path for `organism-comparison.py`.

- `$ cd /KREPE/tex-tutorials/genome-tutorial/`
- `$ python3 /KREPE/tex-tutorials/genome-visualization 6 lactobacillicus.fna plot_distribution plot_de_bruijn -fna -q meta-data.txt`

## 1.5 Results

This is what your abundance graph should look like and the de bruijn graph will be saved as a pdf, although it is advised to save the html because `toyplot` doesn't save the connections to the nodes in the pdf.



Here is what the pdf version of the de bruijn graph looks like:

