

# Contents

<b>1</b>	<b>Graphical interface vs command line interface</b>	<b>1</b>
<b>2</b>	<b>Graphical interface</b>	<b>2</b>
2.1	Generate votes . . . . .	2
2.2	Run algorithm . . . . .	3
2.3	Error finder . . . . .	4
<b>3</b>	<b>Command line interface</b>	<b>5</b>

# 1. Graphical interface vs command line interface

If the program is run without any parameters, then it automatically start the graphical interface.

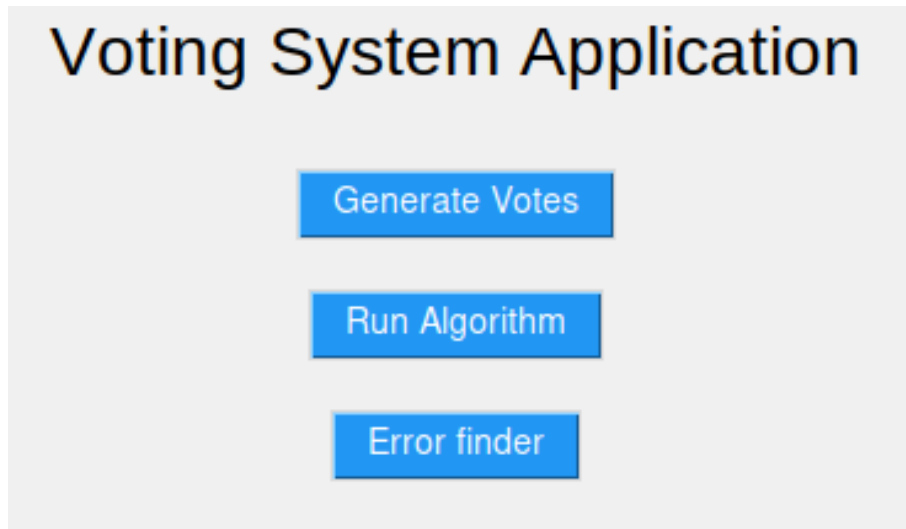
```
1 /bin/python ./vote_streams/src/main.py
```

More about that later. Otherwise if the program is run with any arguments it suggests that you didnt want to use graphical interface and its fully useable with command line.

```
1 /bin/python ./vote_streams/src/main.py -r ./TEST/soc/ ./TEST/TEST.txt stv
  1 0 2
```

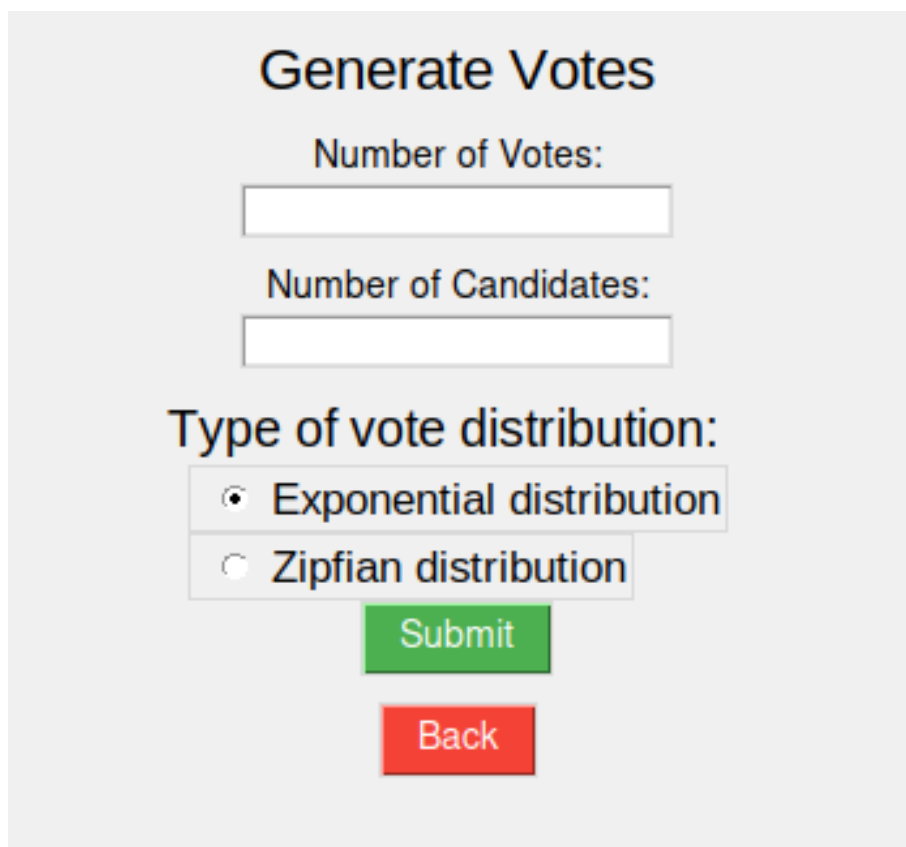
## 2. Graphical interface

If you open the graphical interface, this window will pop up:

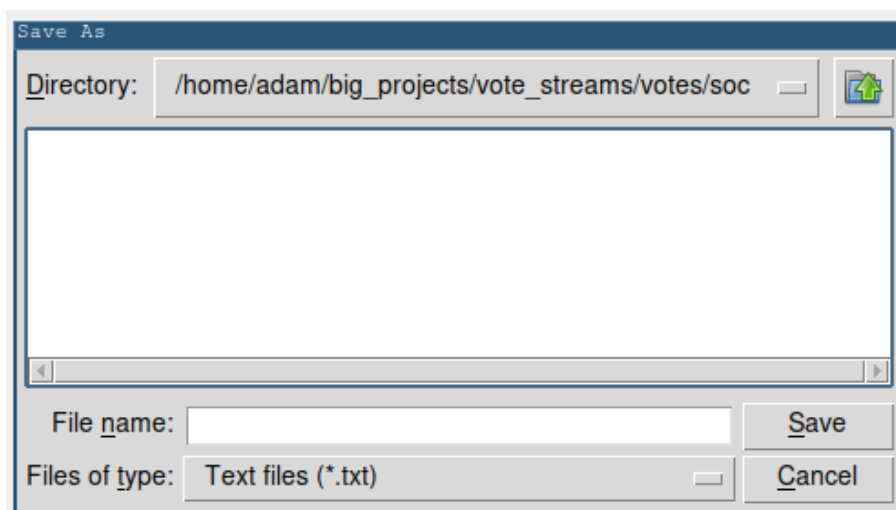


### 2.1 Generate votes

If you clicked in the main menu on the generate votes this window pop up:

A screenshot of a dialog box titled "Generate Votes". The title is in a bold, black font at the top. Below the title, there are two text input fields. The first is labeled "Number of Votes:" and the second is labeled "Number of Candidates:". Below these fields, there is a section titled "Type of vote distribution:" with two radio button options: "Exponential distribution" (which is selected) and "Zipfian distribution". At the bottom of the dialog, there are two buttons: a green "Submit" button and a red "Back" button.

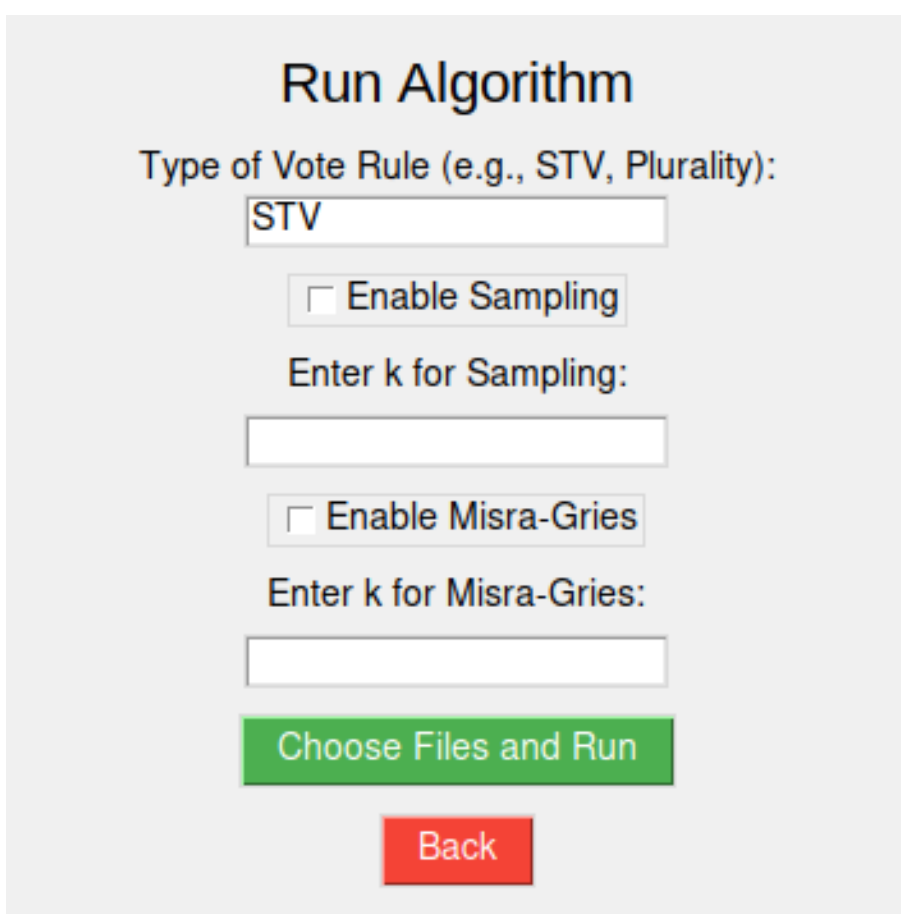
Clicking back will lead you back into main menu. If you put in all the informations and click submit the program will ask where it should save the generated vote file. That looks like this.



The file with generated votes will appear there.

## 2.2 Run algorithm

If you clicked run algorithm this will appear:



In the "Type od Vote Rule" you have these options:

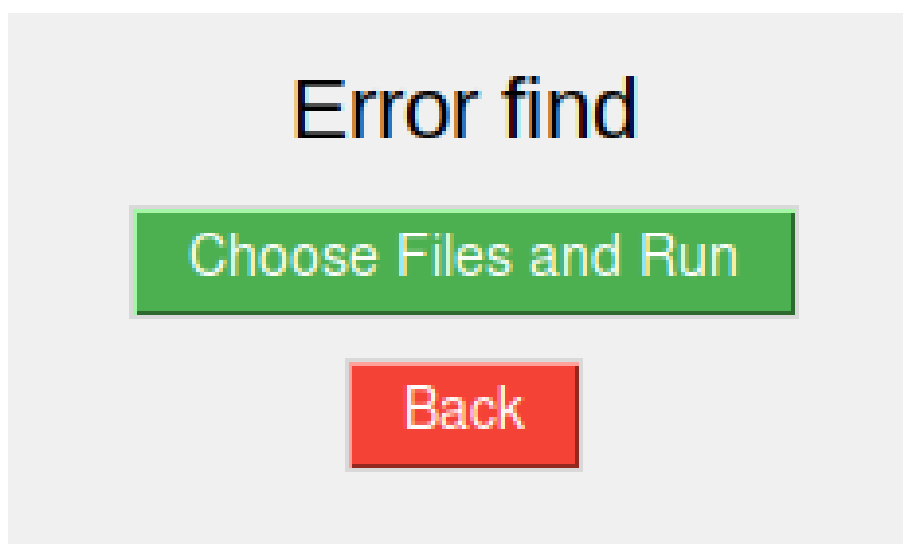
- STV

- Plurality
- Minimax
- Copeland

The capitalization of the letters in that doesn't matter. Then you choose if you want to have enabled Sampling or Misra Gries and put there the  $k$  for them. In Sampling  $k$  is that every  $k$ -th vote will be chosen, for  $k = 3$  it will be number of votes  $// k$ . In Misra Gries it is the classical parameter of Misra Gries algorithm. Make sure you don't choose both Sampling and Misra Gries at the same time, then the program won't let you further. Then you click Choose "Files and Run" same dialog as in Generate votes will appear for choosing the files, except now you must choose folder for an input, because in folder you can put more files that the algorithm will handle. And then save file same as before. And the program will generate the winners.

### 2.3 Error finder

If clicked the Error finder the only thing that will appear is this:



Again just click "Choose Files and run" and the program will ask you to put in the 2 files and 1 save file, choose them and the program will find the error between them.

## 3. Command line interface

When run with wrong number of arguments the program will output the help message which is:

```
1 Argumets are:
2 -g save_path num_votes num_candidates generate_distribution
3
4 -r load_path_folder save_path vote_type sampling_enable misra_enable k
5 (for the "enable" write 1 for true and 0 for false)
6 (k is used for either divide the number of votes when sampling for example
   k = 2 is "half of the votes" or for misra gries as a parametr)
7
8 -e load_path1 load_path2 save_path
```

It basicly works the same as Graphical interface but when -r is used than just one parametr of  $k$  is set, thats because we know that not both Sampling and Misra Gries can be used so we use just one.