

---

# WordCount Documentation

*Release 0.1*

**Harsha**

**May 19, 2021**



# CONTENTS

<b>1</b>	<b>Purpose</b>	<b>1</b>
1.1	Zipf's law . . . . .	1
<b>2</b>	<b>Dependencies</b>	<b>3</b>
2.1	Required . . . . .	3
2.2	Optional . . . . .	3
<b>3</b>	<b>Usage</b>	<b>5</b>
3.1	How to clone the code . . . . .	5
3.2	Make . . . . .	5
3.3	Snakemake . . . . .	5
3.4	Where to find the results . . . . .	5
<b>4</b>	<b>Credit and inspiration</b>	<b>7</b>
<b>5</b>	<b>Exercises (documentation lesson)</b>	<b>9</b>
5.1	Before you start . . . . .	9
5.2	Basic . . . . .	9
5.3	Advanced . . . . .	10
5.4	Meta . . . . .	10



PURPOSE

Write me ...

## 1.1 Zipf's law

Write me ...



## DEPENDENCIES

### 2.1 Required

- Python
- Numpy
- Matplotlib
- Make or Snakemake

### 2.2 Optional

- Docker





## 3.1 How to clone the code

Write me ...

## 3.2 Make

Generate all results:

```
$ make
```

## 3.3 Snakemake

Write me ...

## 3.4 Where to find the results

Write me ...



## CREDIT AND INSPIRATION

Inspired by and derived from <https://hpc-carpentry.github.io/hpc-python/> which is distributed under CC-BY 4.0 (<https://creativecommons.org/licenses/by/4.0/>).



## EXERCISES (DOCUMENTATION LESSON)

### 5.1 Before you start

- Discuss the exercise idea with the classroom.
- Distribute exercises among groups of 2-3 persons.
- Open a GitHub issue and inform the community about the problem and how you plan to solve it. Discuss why we do this.
- Fork this project.
- Commit to your fork. In your commit message auto-close the issue you have addressed.
- Submit a pull request.
- We then review the pull requests.
- After the pull requests are merged we verify that documentation updates itself.

### 5.2 Basic

- Document the purpose of this example code.
- Document how to clone the code.
- Describe the project tree structure.
- Write a sentence or two about Zipf's law and link to Wikipedia (coordinate with the group working on the previous exercise).
- Document how to check the code style with `pycodestyle`.
- Give other developers hints on how they can contribute to the documentation.
- Document how to build the documentation locally (coordinate with the group working on the previous exercise).
- Add an example output.
- Add an example plot (coordinate with the group working on the previous exercise).
- Document where/how to ask for help.
- Add a math equation somewhere.

## 5.3 Advanced

- Add a test and document how to run it.
- Add the possibility to auto-document Python code.

## 5.4 Meta

- Add new exercises ideas for future workshops (edit this file).