

Documentação zhistogram

Dependências do Software

Implementado em Python 2.7

É necessário ter instalado o Python 2.7 e os pacotes não nativos:

numpy

mdtraj

matplotlib

Instalação usando pip:

```
pip install numpy matplotlib mdtraj
```

Modo de usar

```
zhistogram.py [-h] [-b BINS] [-c CUTOFF] [-f FRAMES] [-o OUTPUT] [-p] [-s] [-cv] [-v] traj  
topology index group1 group2
```

Argumentos obrigatórios:

traj	.xtc trajectory file.
Topology	.pdb topology file.
Index	.ndx index file.
group1	Name of group 1.
group2	Name of group 2.

Argumentos opcionais:

-h, --help	show this help message and exit
-b BINS, --bins BINS	Number of bins (default: numpy histogram 'auto').
-c CUTOFF, --cutoff CUTOFF	Cutoff distance.
-f FRAMES, --frames FRAMES	Range or number of frames to consider. Example 1-100 or 100.
-o OUTPUT, --output OUTPUT	File to output the histogram image and data (without extension)
-p, --plot	Show plot of histogram.
-s, --smooth	Smooth curve by applying Fourier Transform.
-cv, --converge	Converge curve to 1.
-v, --verbose	Make the software more verbose.

Exemplos

Descompacte o arquivo exemplo.zip na mesma pasta do script zhistogram.py

Você encontrará os seguintes arquivos:

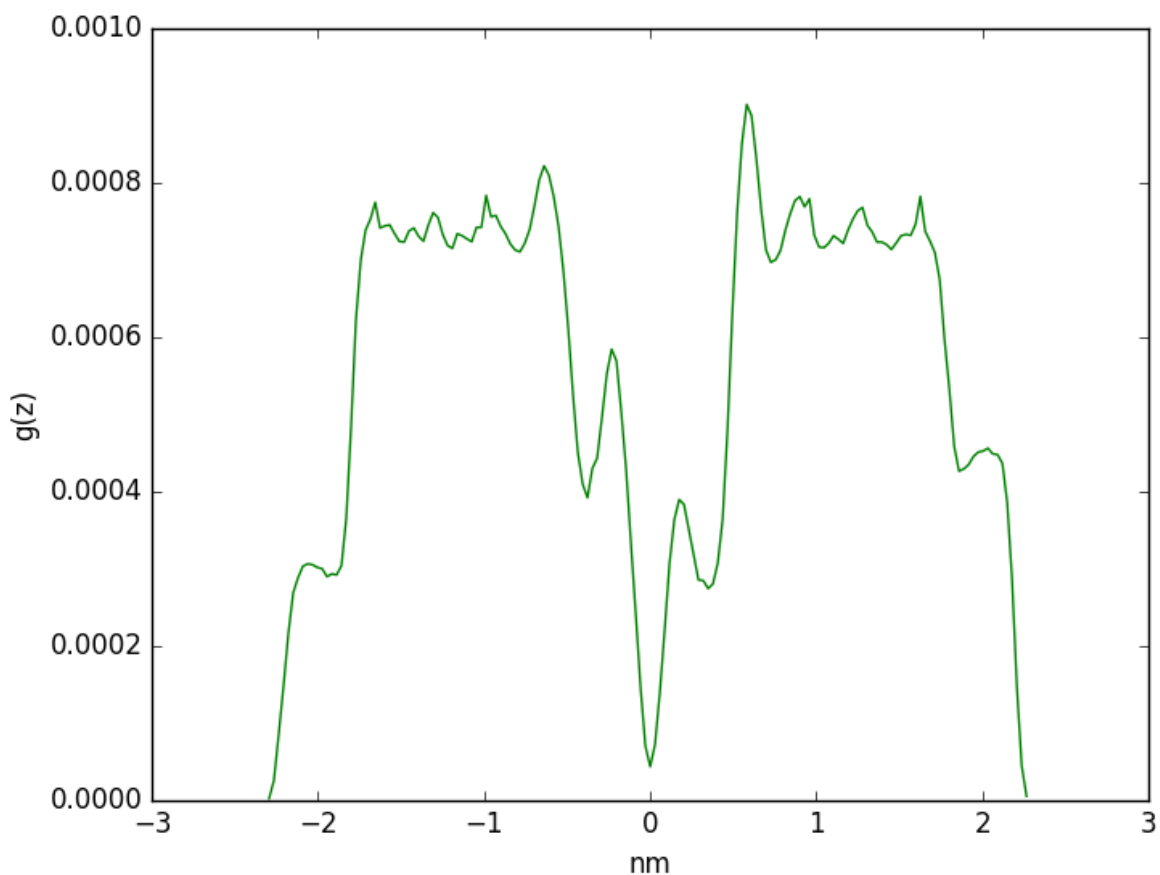
traj.xtc	arquivo binário com as trajetórias dos átomos
topol.pdb	arquivo com a topologia das moléculas (pode ser o primeiro frame de uma trajetória PDB)
index.ndx	arquivo com os indexes dos átomos separados por grupos

Exemplo 1

```
python zhistogram.py -v -f 100-120 -o "exemplo1" traj.xtc  
topol.pdb index.ndx HSH OW
```

Este comando vai ler a trajetória e calcular as distâncias entre os grupos HSH e OW nos frames entre 100 e 120 gerando dois arquivos (exemplo1.png e exemplo1.txt) e a saída abaixo:

```
Reading trajectory from traj.xtc and topology from topol.pdb...  
Trajectory data:  
# frames: 3201  
# atoms: 5055  
# residues: 1426  
Reading atoms indexes from index.ndx...  
Box mean size (nm): [ 3.40681748  3.68506747  3.98924763]  
Calculating distances between HSH and OW ...  
21 from 21 frames  
Calculating histogram...  
Normalizing...  
Smoothing...  
Converging...  
Saving histogram plot to exemplo1.png ...  
Saving histogram data to exemplo1.txt ...  
Done.
```



Exemplo 2

```
python zhistogram.py -v -s -cv -c 1.5 -b 200 -f 100-120 -o  
"exemplo2" data/traj.xtc data/topol.pdb data/index.ndx HSH OW
```

O comando acima lê os arquivos de trajetória e calcula as distâncias entre os grupos HSH e OW nos frames entre 100 e 120 (-f 100-120), utilizando 200 bins (-b 200), com um range de -1.5nm a 1.5nm (-c 1.5), aplica transformada de fourier para suavizar curva (-s) e converge as pontas para 1 (-cv). Ele gera os arquivos “exemplo2.png” e exemplo2.txt **Importante notar que a convergência é feita com base somente no primeiro valor do gráfico, sempre indicar um range onde o início da curva deve ser igual a 1.**

