

# Looping\_Diagonal

October 28, 2019

## 1 Looping through all the letters in diagonal direction

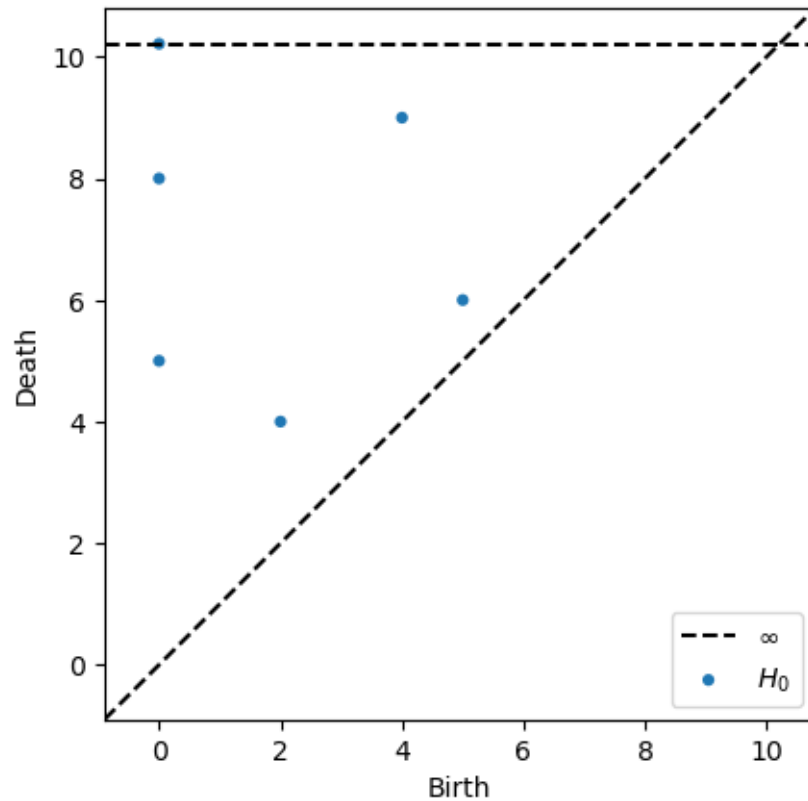
### 1.1 Importing notebooks

```
[9]: import numpy as np
import matplotlib.pyplot as plt
import scipy
from scipy import ndimage
import PIL
from persim import plot_diagrams
from ripser import ripser, lower_star_img
import csv
```

```
[10]: dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

```
(6, 2)
[[ 2.  4.]
 [ 0.  5.]
 [ 5.  6.]
 [ 0.  8.]
 [ 4.  9.]
 [ 0. inf]]
```



```
[11]: # Diagonal scanning through loops
letters = genfromtxt('letters.csv', delimiter=',') # Upload the file

dgmDiagonal = [None]*26 #Initialize an empty list
for i in range(26):
    letter_one_line=letters[i,:]

    # initialize matrix of size 10x10 with all values 100
    letter=np.full((10, 10), 100)

    # convert one line letter to 10x10 matrix replacing zeros with 100
    for k in range(1,101):
        if letter_one_line[k]==1.0:
            row=int((k-1)/10)
            column=(k-1)%10
            letter[row,column]=(column+row)*k%10
    dgmDiagonal[i] = lower_star_img(letter)

[12]: # Print A-Z diagrams
print(dgmDiagonal[0:25])
```

```
[array([[ 0.,  3.],
```

```

[ 0.,  4.],
[ 1.,  5.],
[ 0.,  5.],
[ 0., inf]], array([[ 1.,  2.],
[ 0.,  4.],
[ 0.,  4.],
[ 2.,  4.],
[ 2.,  5.],
[ 0.,  6.],
[ 0.,  6.],
[ 1.,  7.],
[ 0., inf]], array([[ 0.,  4.],
[ 1.,  5.],
[ 0.,  5.],
[ 4.,  9.],
[ 0., inf]], array([[ 1.,  2.],
[ 0.,  4.],
[ 5.,  6.],
[ 2.,  6.],
[ 1.,  7.],
[ 0.,  8.],
[ 0., inf]], array([[ 0.,  4.],
[ 0.,  4.],
[ 5.,  6.],
[ 1.,  7.],
[ 0.,  8.],
[ 0.,  8.],
[ 2.,  8.],
[ 4.,  9.],
[ 0., inf]], array([[ 0.,  4.],
[ 5.,  6.],
[ 1.,  7.],
[ 0.,  8.],
[ 2.,  8.],
[ 0., inf]], array([[ 1.,  4.],
[ 0.,  4.],
[ 0.,  5.],
[ 2.,  5.],
[ 5.,  6.],
[ 0., inf]], array([[ 1.,  7.],
[ 2.,  8.],
[ 1.,  8.],
[ 0., inf]], array([[ 0.,  5.],
[ 0.,  5.],
[ 0.,  5.],
[ 0., inf]], array([[ 2.,  5.],
[ 0.,  6.],
[ 2.,  8.],

```

```

[ 0., inf]], array([[ 1.,  2.],
[ 1.,  2.],
[ 2.,  5.],
[ 0.,  7.],
[ 2.,  8.],
[ 0., inf]], array([[ 0.,  4.],
[ 1.,  7.],
[ 2.,  8.],
[ 0., inf]], array([[ 0.,  2.],
[ 2.,  4.],
[ 0.,  5.],
[ 2.,  6.],
[ 2.,  8.],
[ 0.,  8.],
[ 0., inf]], array([[ 1.,  4.],
[ 2.,  4.],
[ 0.,  5.],
[ 0., inf]], array([[ 0.,  4.],
[ 0.,  6.],
[ 5.,  6.],
[ 2.,  6.],
[ 1.,  8.],
[ 0.,  8.],
[ 0., inf]], array([[ 1.,  5.],
[ 5.,  6.],
[ 2.,  6.],
[ 0.,  7.],
[ 0., inf]], array([[ 0.,  4.],
[ 0.,  6.],
[ 5.,  6.],
[ 0.,  8.],
[ 0.,  8.],
[ 2.,  8.],
[ 0., inf]], array([[ 1.,  2.],
[ 1.,  2.],
[ 0.,  2.],
[ 2.,  4.],
[ 2.,  6.],
[ 0.,  7.],
[ 0., inf]], array([[ 0.,  4.],
[ 2.,  5.],
[ 5.,  6.],
[ 0.,  7.],
[ 0.,  8.],
[ 2.,  8.],
[ 4.,  9.],
[ 0., inf]], array([[ 0.,  5.],
[ 0.,  5.],

```

```

[ 0., 5.],
[ 4., 9.],
[ 4., 9.],
[ 0., inf]], array([[ 0., 4.],
[ 1., 7.],
[ 0., 8.],
[ 0., 8.],
[ 2., 8.],
[ 0., inf]]), array([[ 0., 2.],
[ 2., 3.],
[ 2., 8.],
[ 0., inf]]), array([[ 1., 2.],
[ 0., 2.],
[ 0., 3.],
[ 0., 5.],
[ 0., 6.],
[ 5., 6.],
[ 4., 8.],
[ 0., 9.],
[ 0., inf]]), array([[ 1., 2.],
[ 0., 2.],
[ 2., 4.],
[ 0., 4.],
[ 0., inf]]), array([[ 0., 5.],
[ 0., 8.],
[ 2., 8.],
[ 0., inf]]])

```

```

[13]: # Print A digram
print(dgmDiagonal[0])

```

```

[[ 0. 3.]
 [ 0. 4.]
 [ 1. 5.]
 [ 0. 5.]
 [ 0. inf]]

```

```

[14]: # Print z diagram
print(dgmDiagonal[25])

```

```

[[ 2. 4.]
 [ 0. 5.]
 [ 5. 6.]
 [ 0. 8.]
 [ 4. 9.]
 [ 0. inf]]

```