TDAP1

September 13, 2019

1 Enzo Rodriguez

2 Lower Star Image Filtrations for Scanning

Here I take the notebook Lower Star Image Filtrations from ripser.py and adjust it to create a left-to-right scan on a letter from letters.csv and I would also perform a right-to-left scan on each letter. First let's import everything they imported. I also import csv to read our csv file.

[3]:
```
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
```

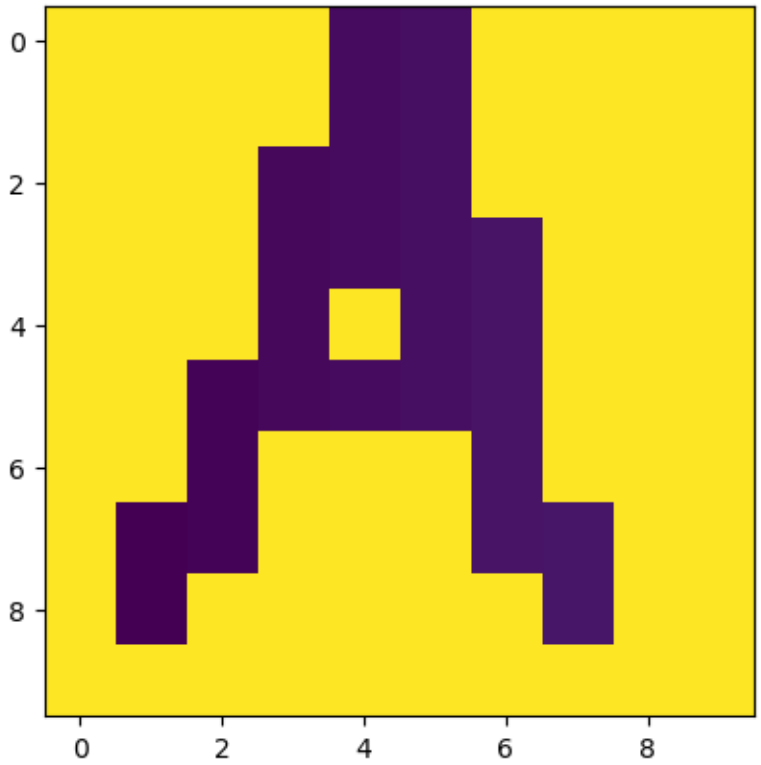
2.1 Left-to-right Scanning Example

Our function on the image will be very simple: if the value we see in the corresponding row is 0 then the value on the pixel is 100, if the value is 1 then the value is the x1 coordinate. I'll be scanning the leters in sequential order from A-Z first left-to-right.

[7]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[0,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
plt.show()
```

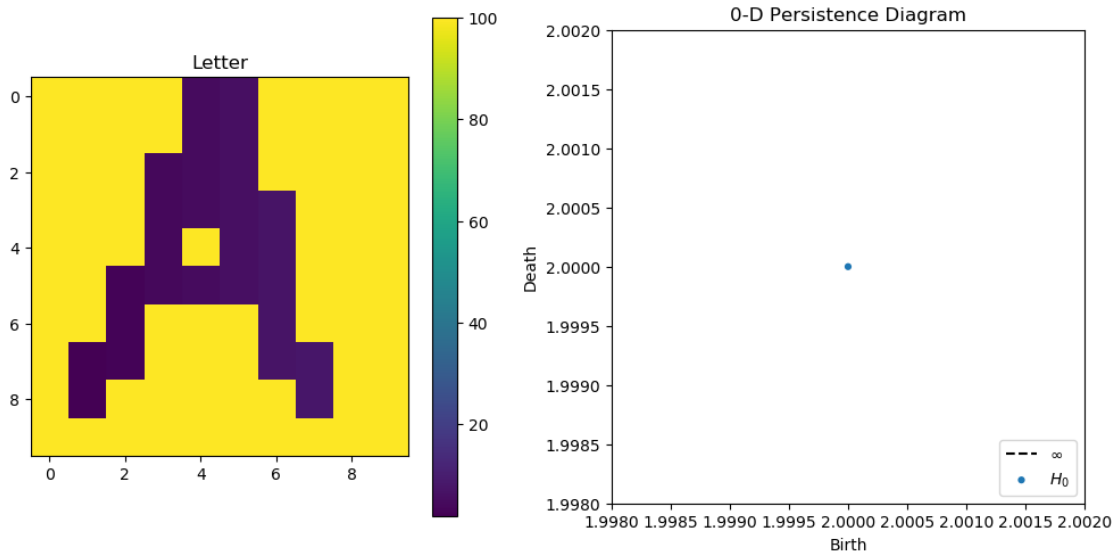
```
(10, 10)
[[100 100 100 100   5   6 100 100 100 100]
 [100 100 100 100   5   6 100 100 100 100]
 [100 100 100   4   5   6 100 100 100 100]
 [100 100 100   4   5   6   7 100 100 100]
 [100 100 100   4 100   6   7 100 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100   2   3 100 100 100   7   8 100 100]
 [100   2 100 100 100 100 100   8 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[8]:
```
dgm = lower_star_img(letter)
print(dgm)
```

```
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 2. inf]]
```



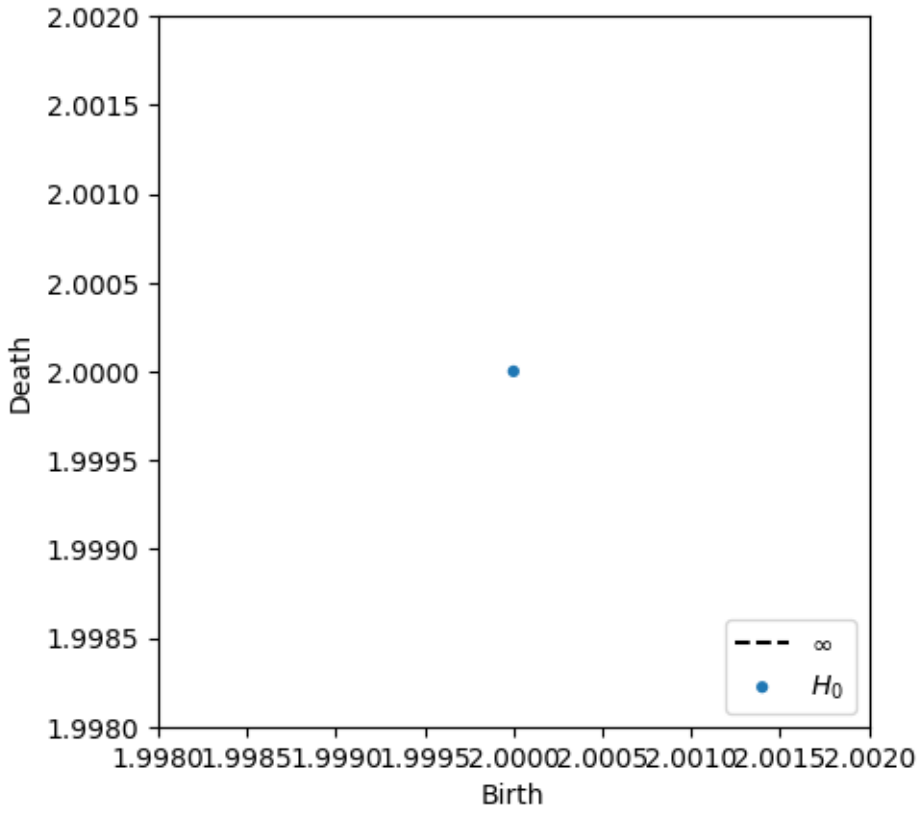
[9]:
```
help(lower_star_img)
```

```
Help on function lower_star_img in module ripser.ripser:

lower_star_img(img)
    Construct a lower star filtration on an image

    Parameters
    ----------
    img: ndarray (M, N)
        An array of single channel image data

    Returns
    -------
    I: ndarray (K, 2)
```

A 0-dimensional persistence diagram corresponding to the sublevelset filtration

[11]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

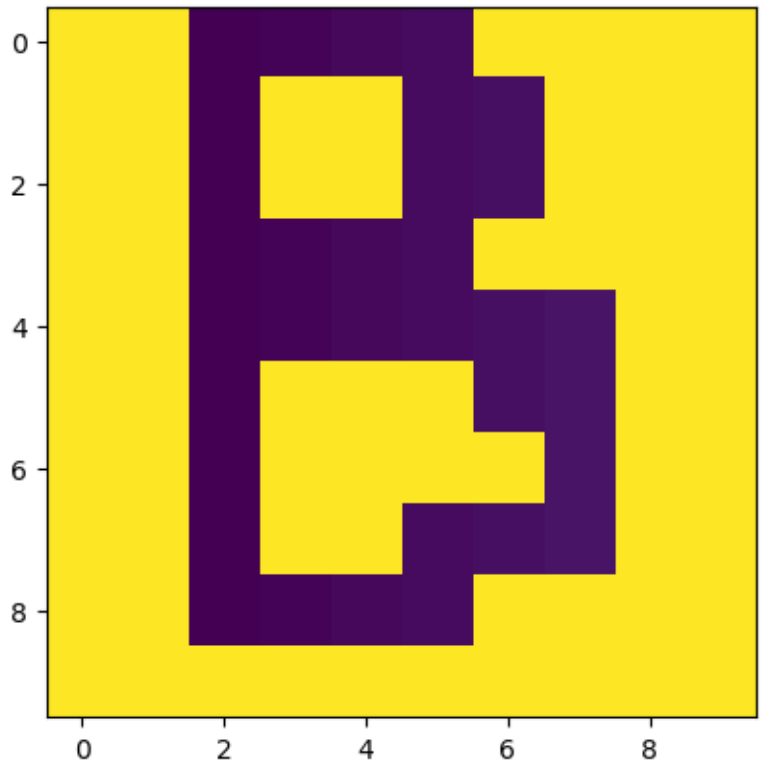
```
(1, 2)
[[ 2. inf]]
```



[12]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[1,:]
```

```
# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
plt.show()
```

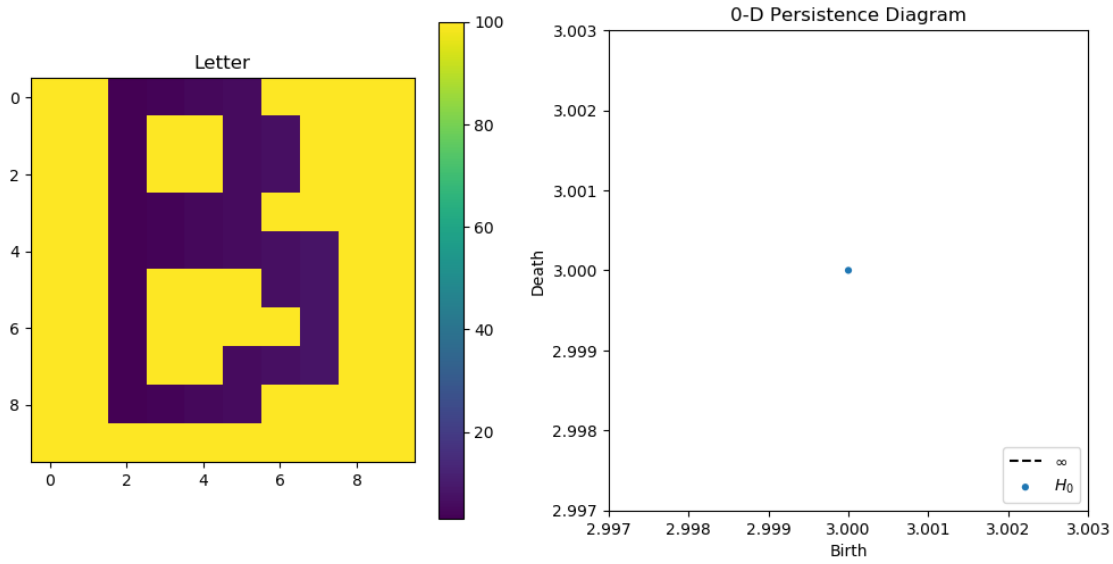
```
(10, 10)
[[100 100   3   4   5   6 100 100 100 100]
 [100 100   3 100 100   6   7 100 100 100]
 [100 100   3 100 100   6   7 100 100 100]
 [100 100   3   4   5   6 100 100 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100 100   3 100 100 100   7   8 100 100]
 [100 100   3 100 100 100 100   8 100 100]
 [100 100   3 100 100   6   7   8 100 100]
 [100 100   3   4   5   6 100 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[13]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

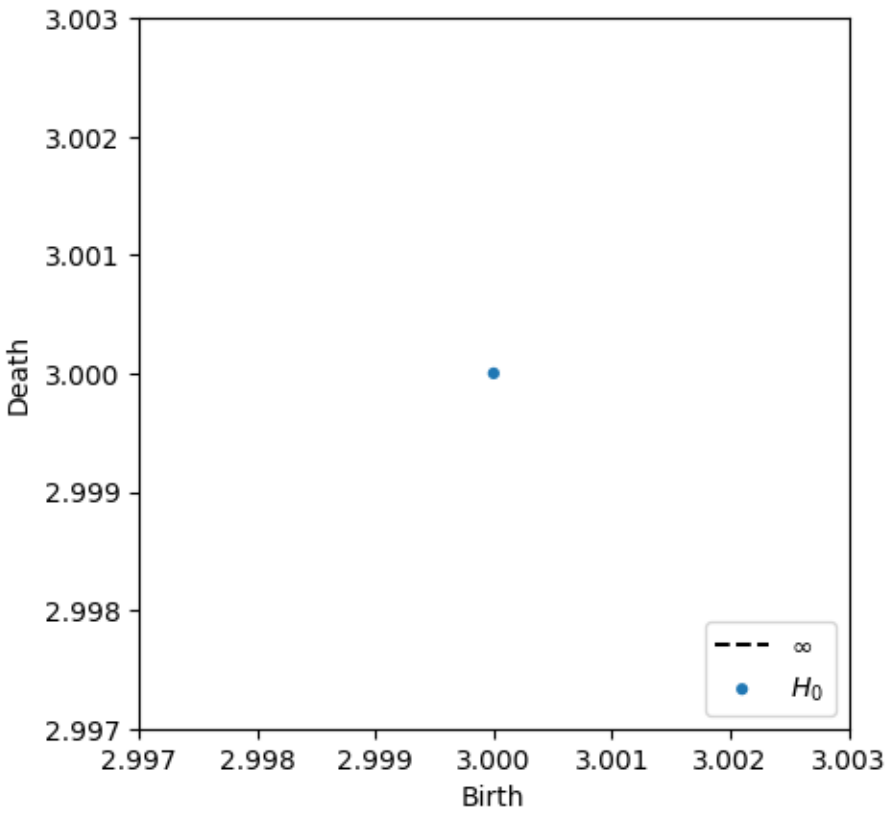
```
[[ 3. inf]]

/Users/enzo/anaconda2/lib/python2.7/site-packages/matplotlib/axes/_base.py:3152:
UserWarning: Attempting to set identical left==right results
in singular transformations; automatically expanding.
left=3.0, right=3.0
  'left=%s, right=%s') % (left, right))
/Users/enzo/anaconda2/lib/python2.7/site-packages/matplotlib/axes/_base.py:3471:
UserWarning: Attempting to set identical bottom==top results
in singular transformations; automatically expanding.
```

```
bottom=3.0, top=3.0
  'bottom=%s, top=%s') % (bottom, top))
```



[14]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

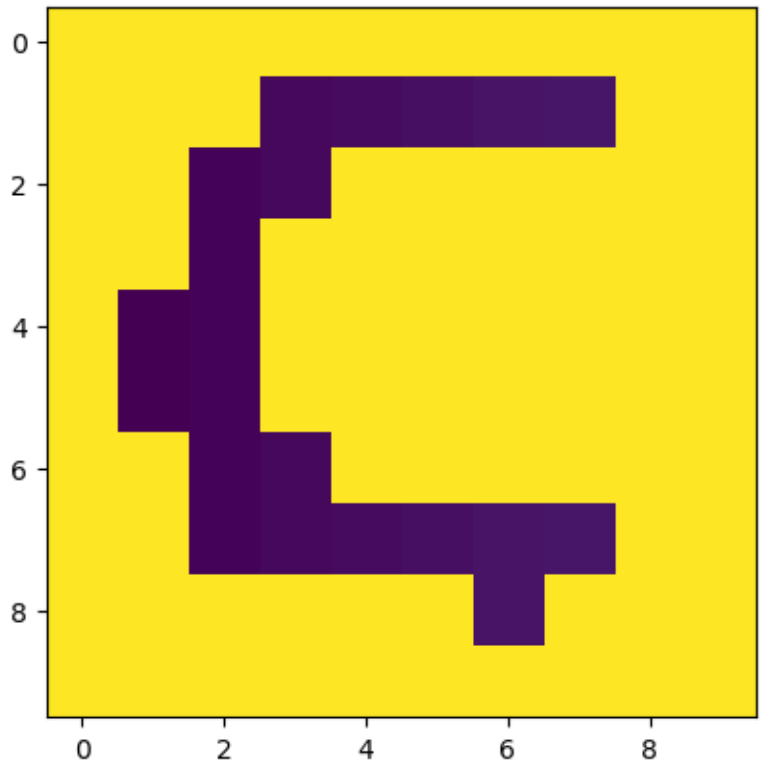
```
(1, 2)
[[ 3. inf]]
```



[15]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[2,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

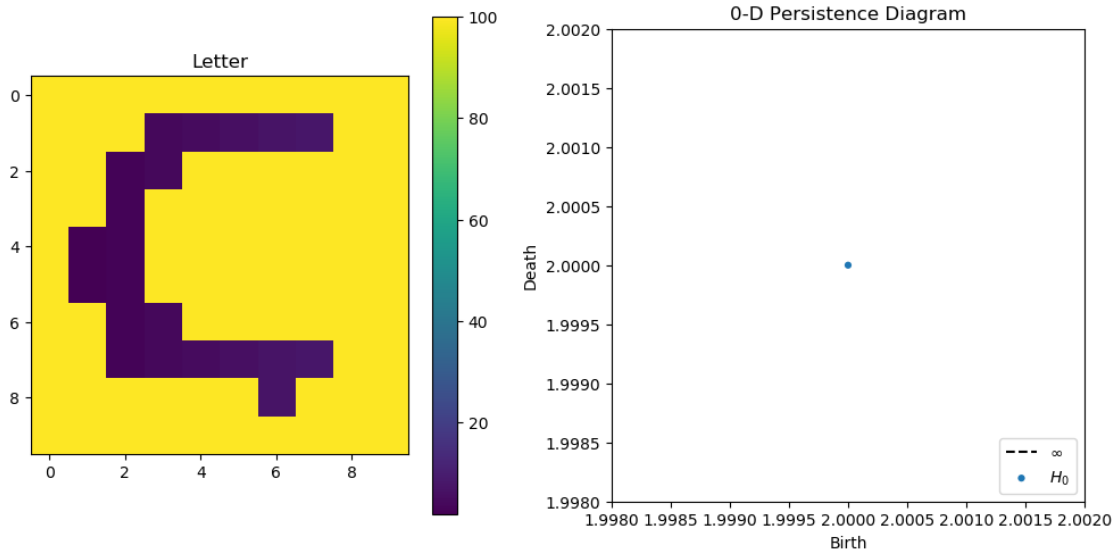
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100 100   4   5   6   7   8 100 100]
 [100 100   3   4 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100   2   3 100 100 100 100 100 100 100]
 [100   2   3 100 100 100 100 100 100 100]
 [100 100   3   4 100 100 100 100 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100 100 100 100 100 100   7 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[16]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

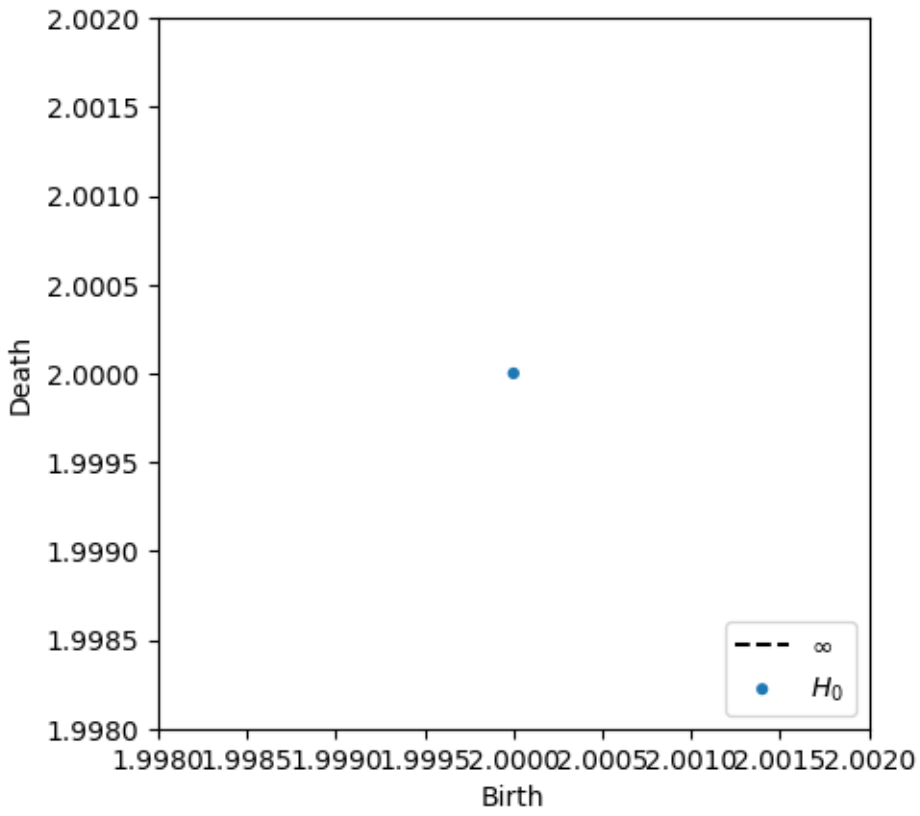
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 2. inf]]
```



[17]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

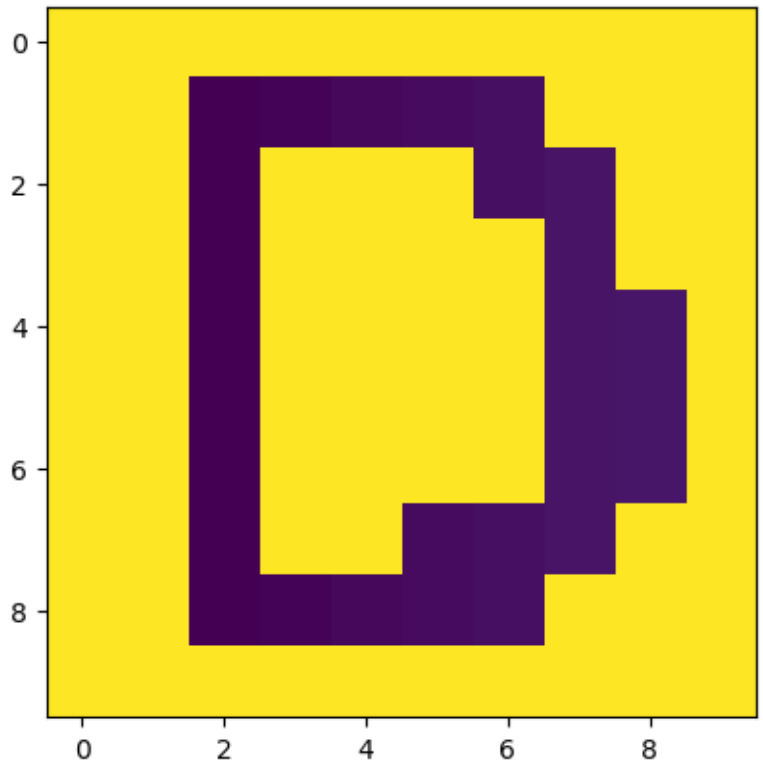
```
(1, 2)
[[ 2. inf]]
```



[18]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[3,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

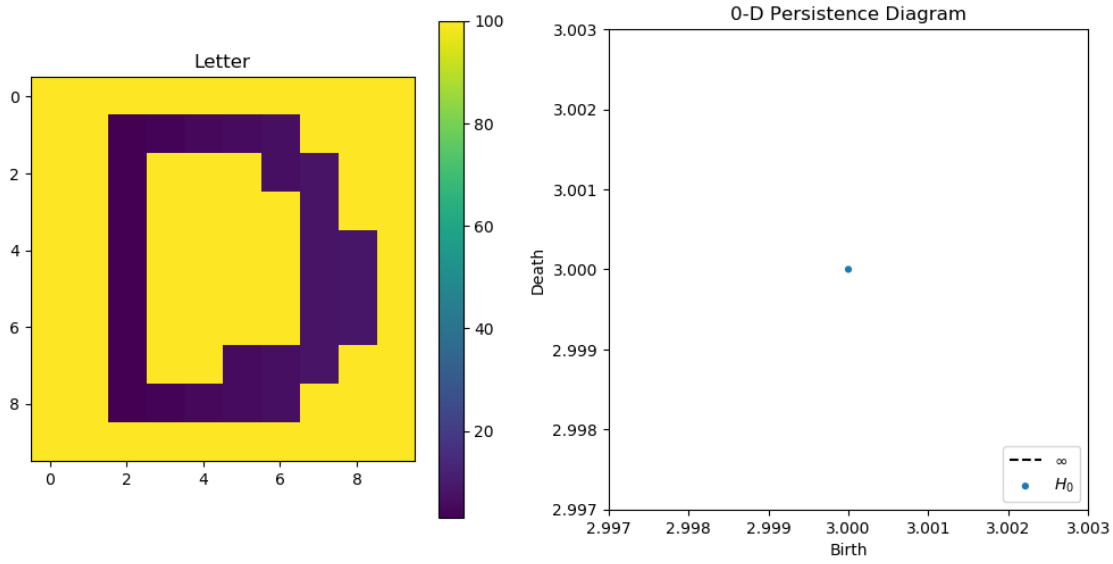
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100   3 100 100 100   7   8 100 100]
 [100 100   3 100 100 100 100   8 100 100]
 [100 100   3 100 100 100 100   8   9 100]
 [100 100   3 100 100 100 100   8   9 100]
 [100 100   3 100 100 100 100   8   9 100]
 [100 100   3 100 100   6   7   8 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[19]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

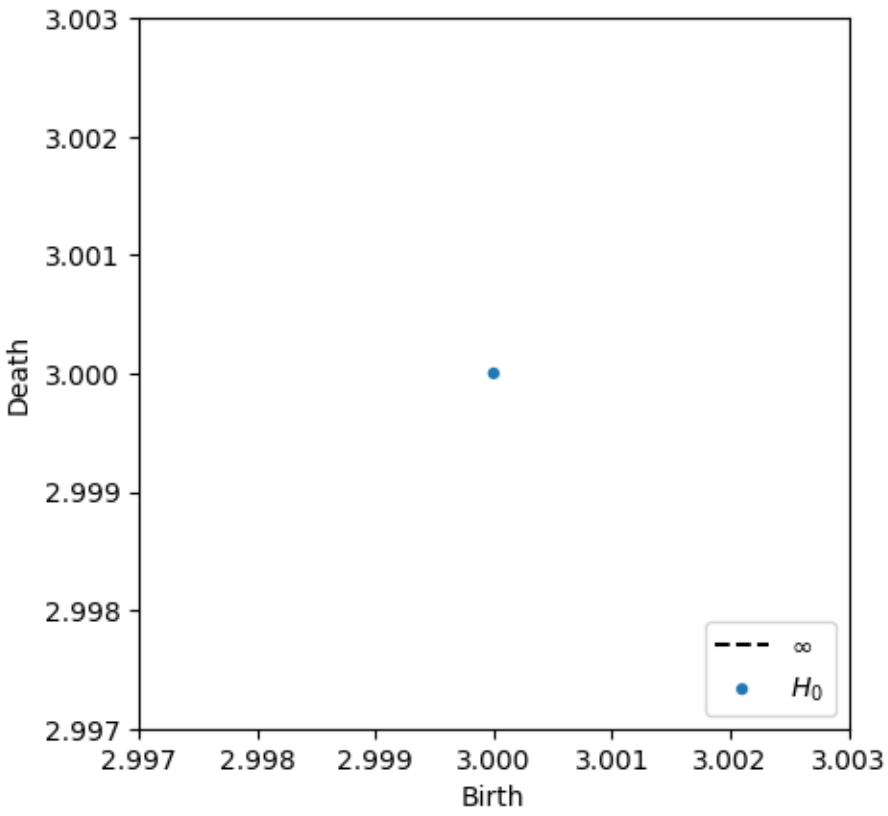
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3. inf]]
```



[20]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

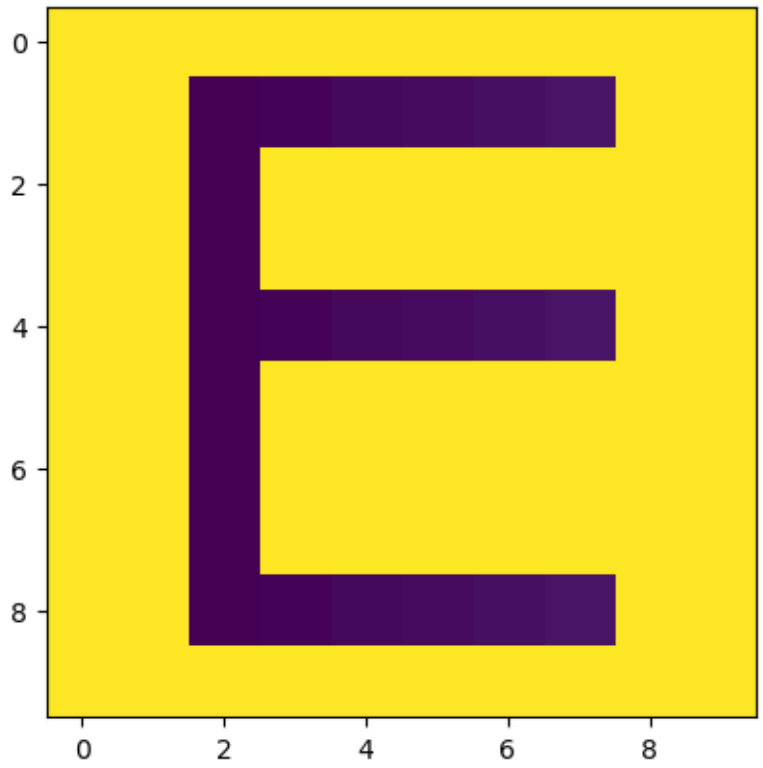
```
(1, 2)
[[ 3. inf]]
```



[21]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[4,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

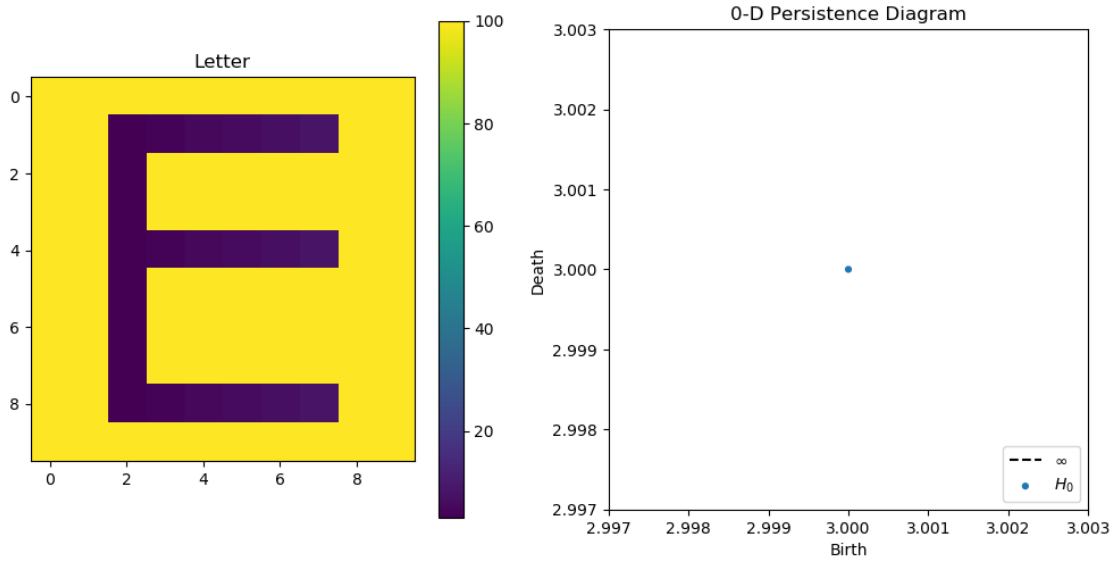
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[22]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

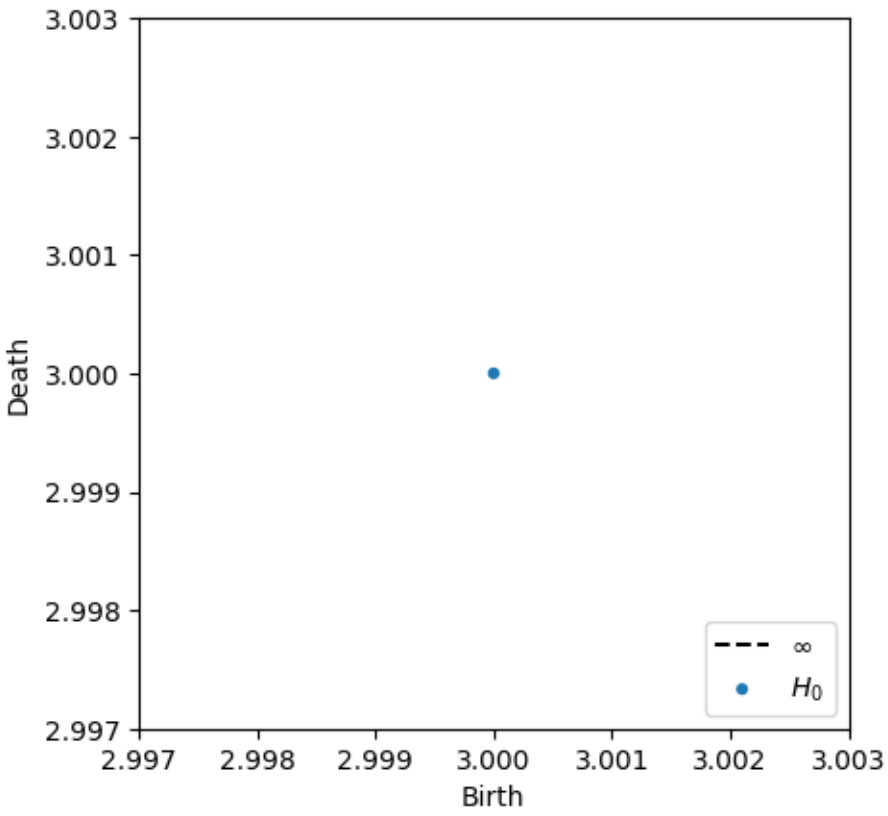
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3. inf]]
```



[23]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

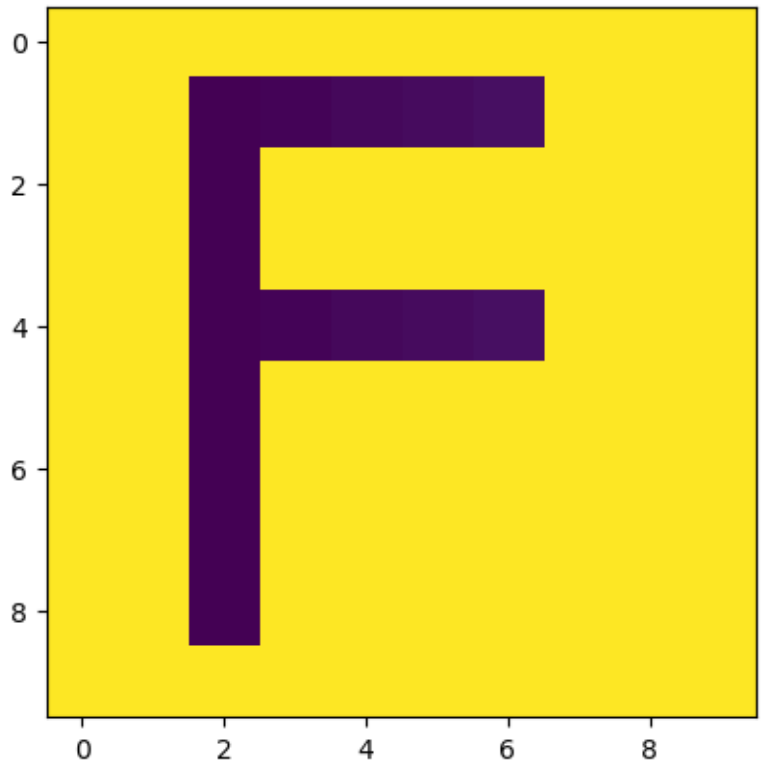
```
(1, 2)
[[ 3. inf]]
```



[24]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[5,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

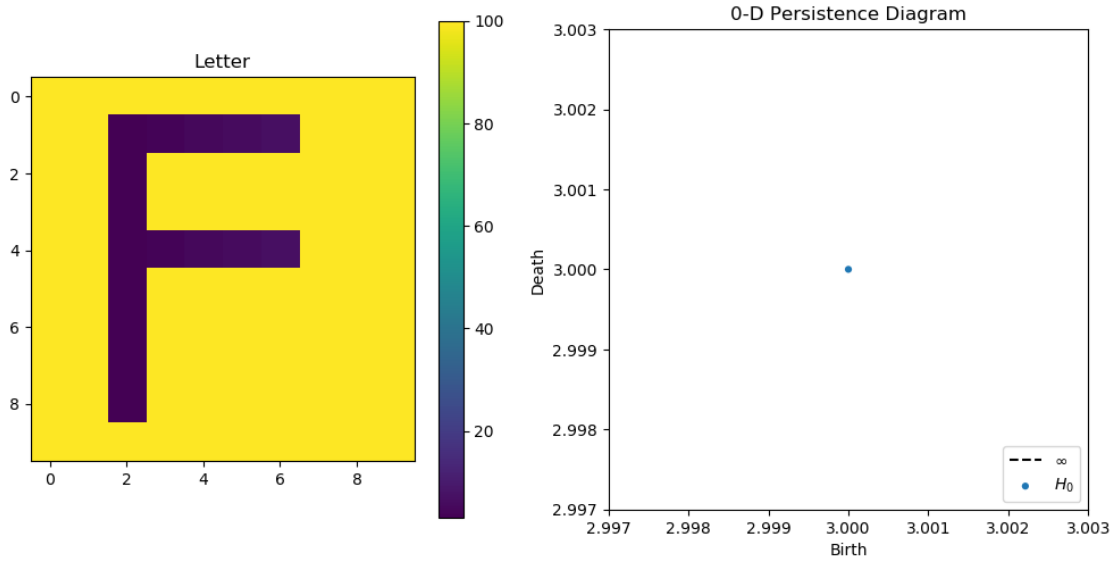
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[25]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

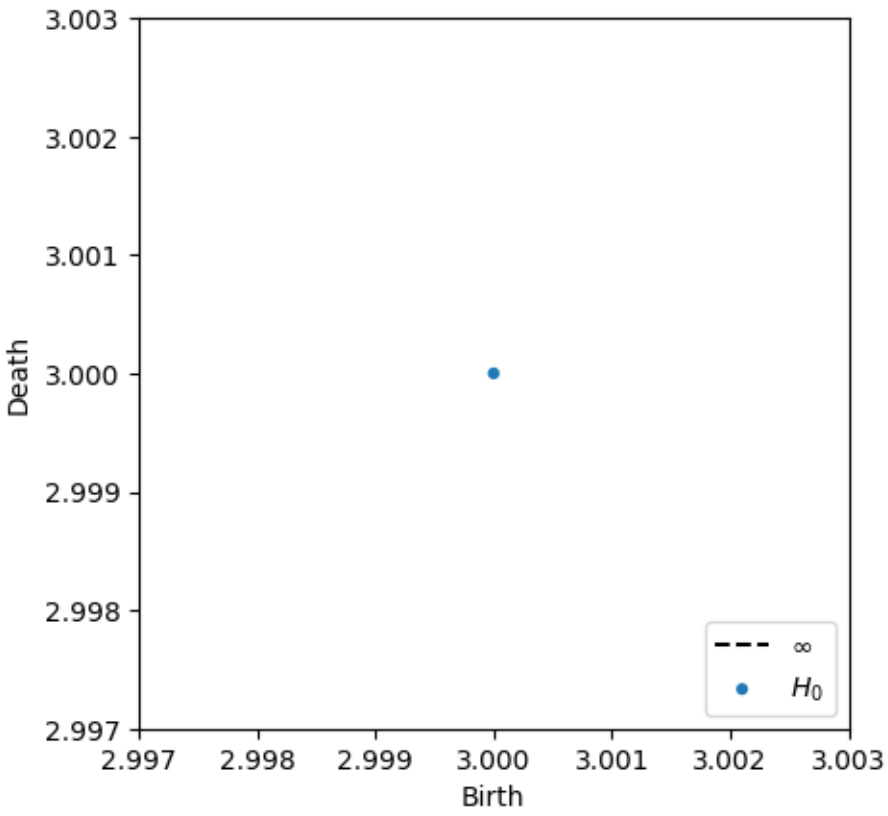
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3. inf]]
```



[26]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

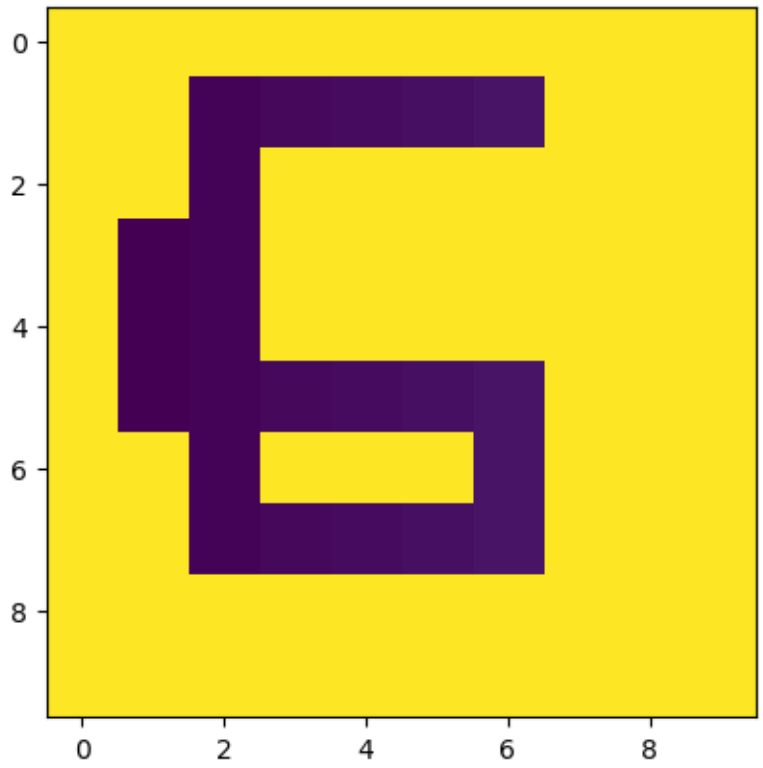
```
(1, 2)
[[ 3. inf]]
```



[27]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[6,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

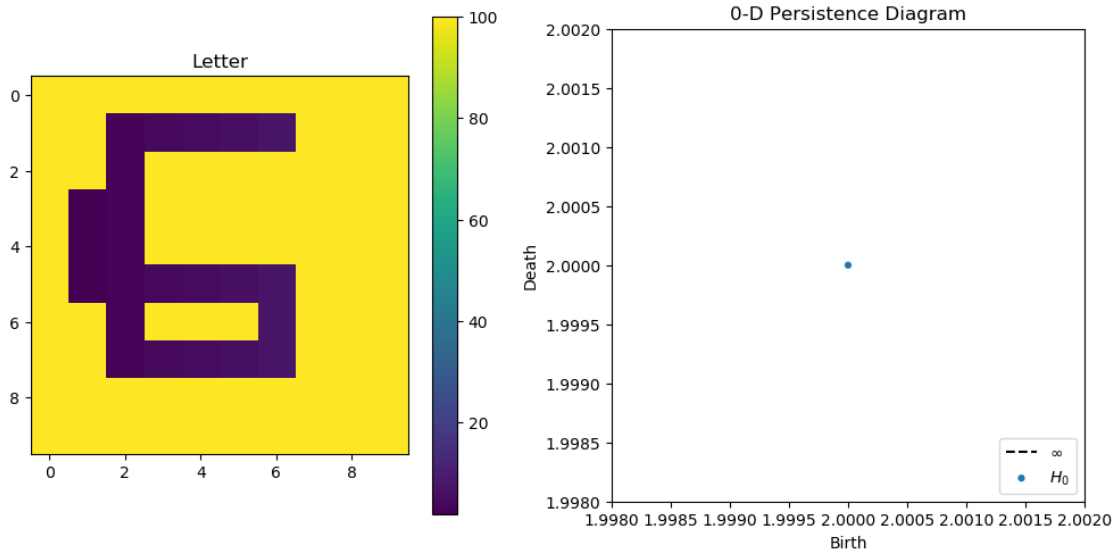
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100   2   3 100 100 100 100 100 100 100]
 [100   2   3 100 100 100 100 100 100 100]
 [100   2   3   4   5   6   7 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[28]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

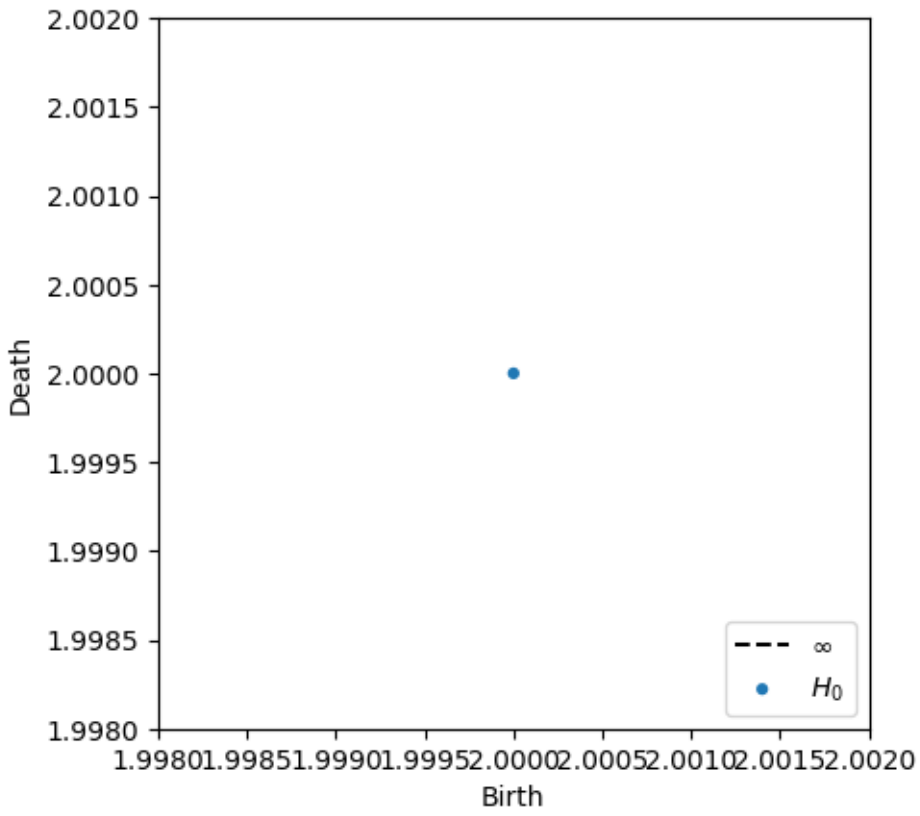
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 2. inf]]
```



[29]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

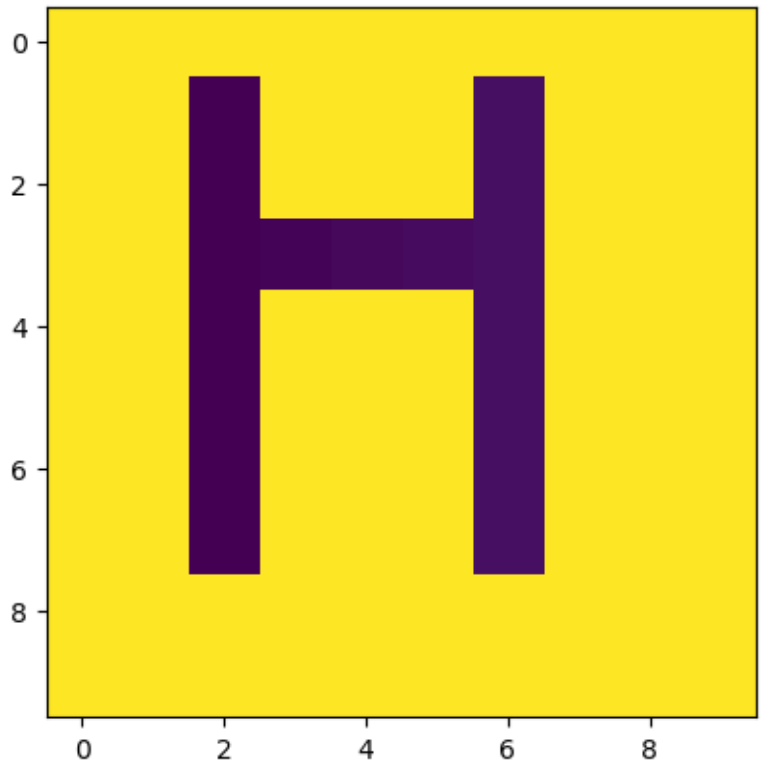
```
(1, 2)
[[ 2. inf]]
```



[30]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[7,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

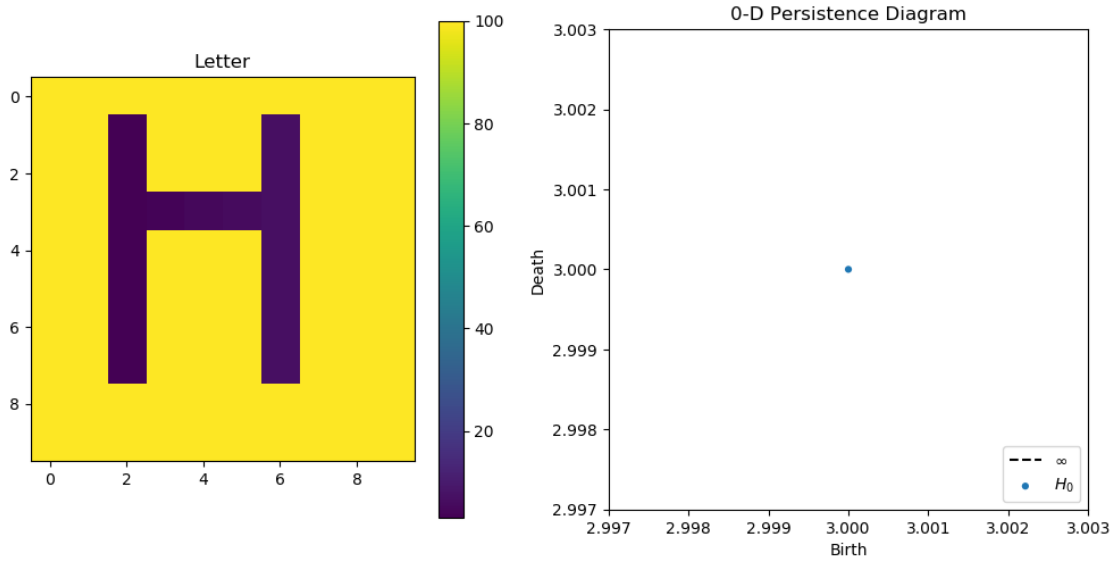
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[31]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

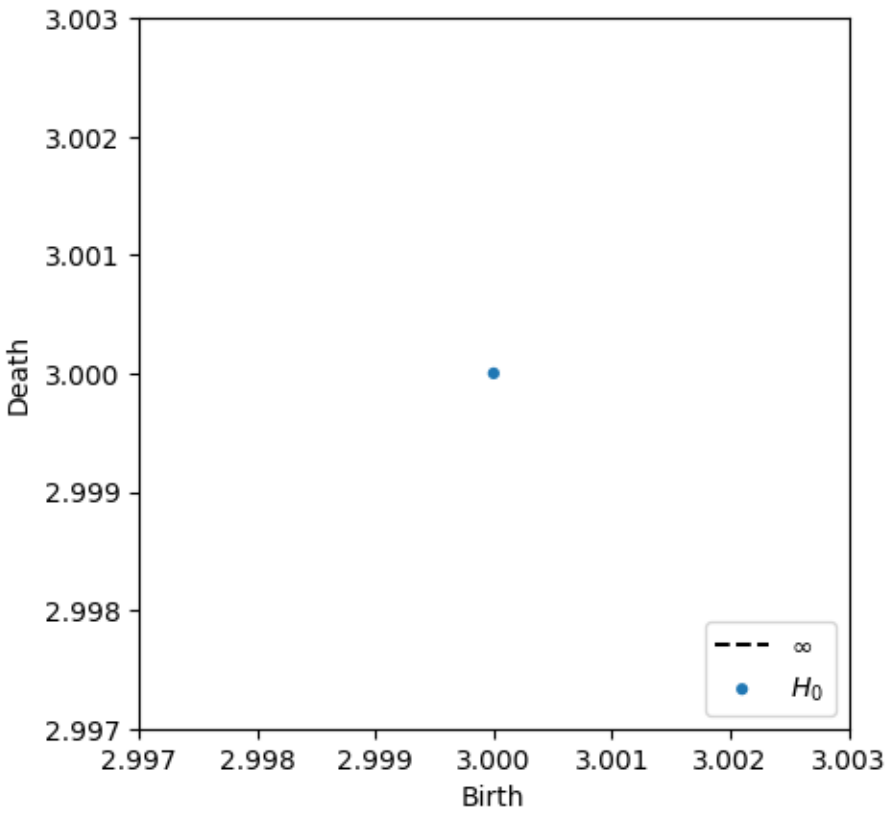
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3. inf]]
```



[32]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

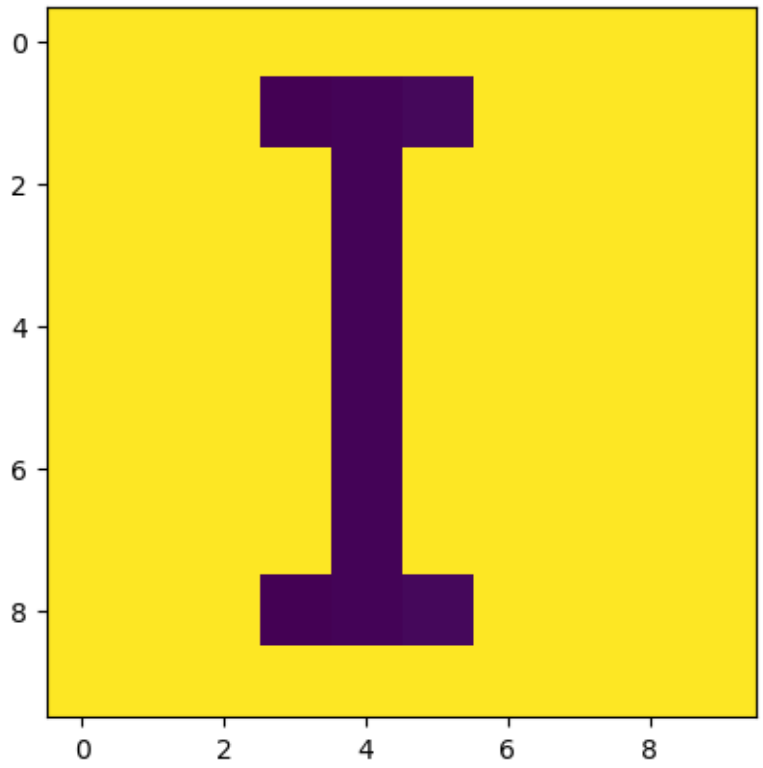
```
(1, 2)
[[ 3. inf]]
```



[33]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[8,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

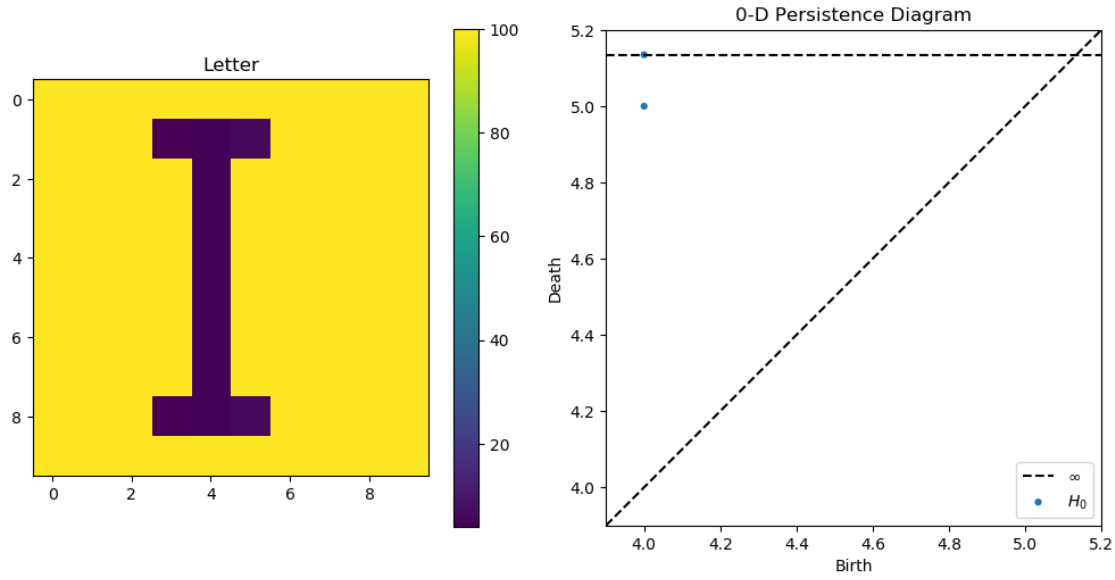
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100 100   4   5   6 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100   4   5   6 100 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[34]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

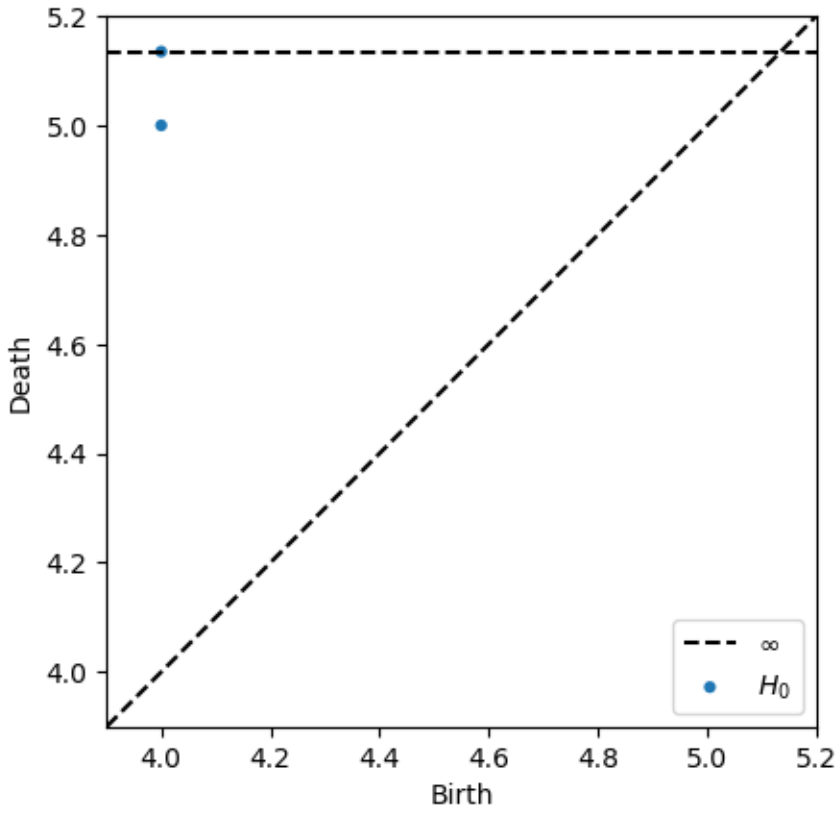
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 4.  5.]
 [ 4. inf]]
```



[35]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

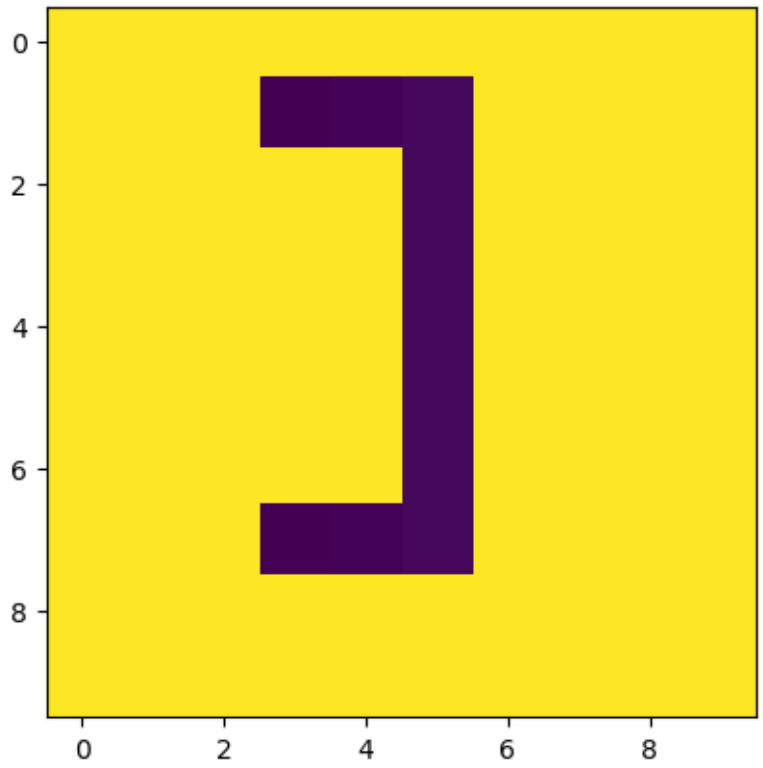
```
(2, 2)
[[ 4.  5.]
 [ 4. inf]]
```



[36]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[9,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

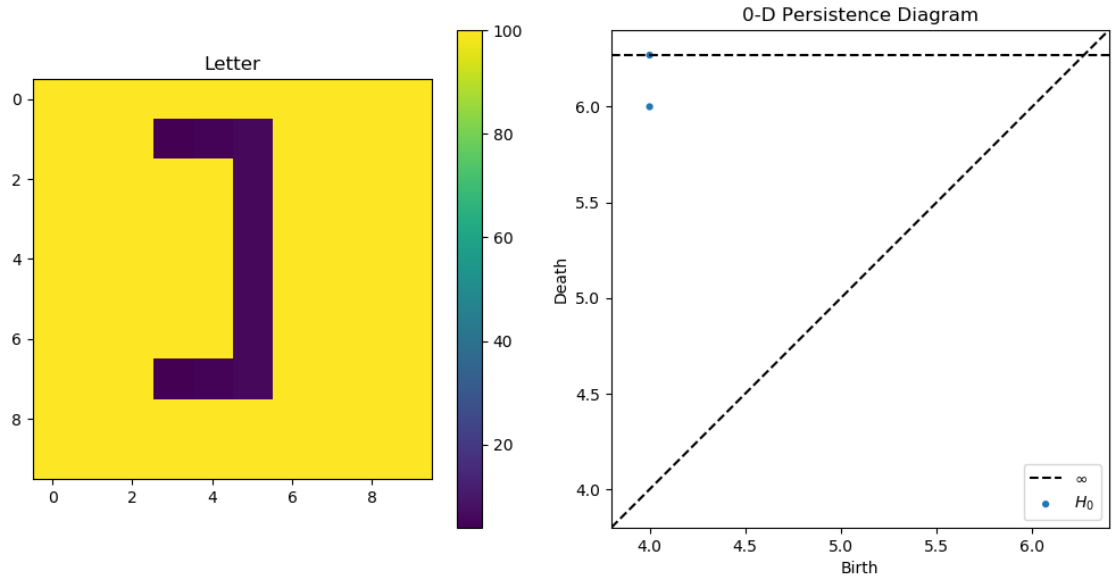
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100 100   4   5   6 100 100 100 100]
 [100 100 100 100 100   6 100 100 100 100]
 [100 100 100 100 100   6 100 100 100 100]
 [100 100 100 100 100   6 100 100 100 100]
 [100 100 100 100 100   6 100 100 100 100]
 [100 100 100 100 100   6 100 100 100 100]
 [100 100 100   4   5   6 100 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[37]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

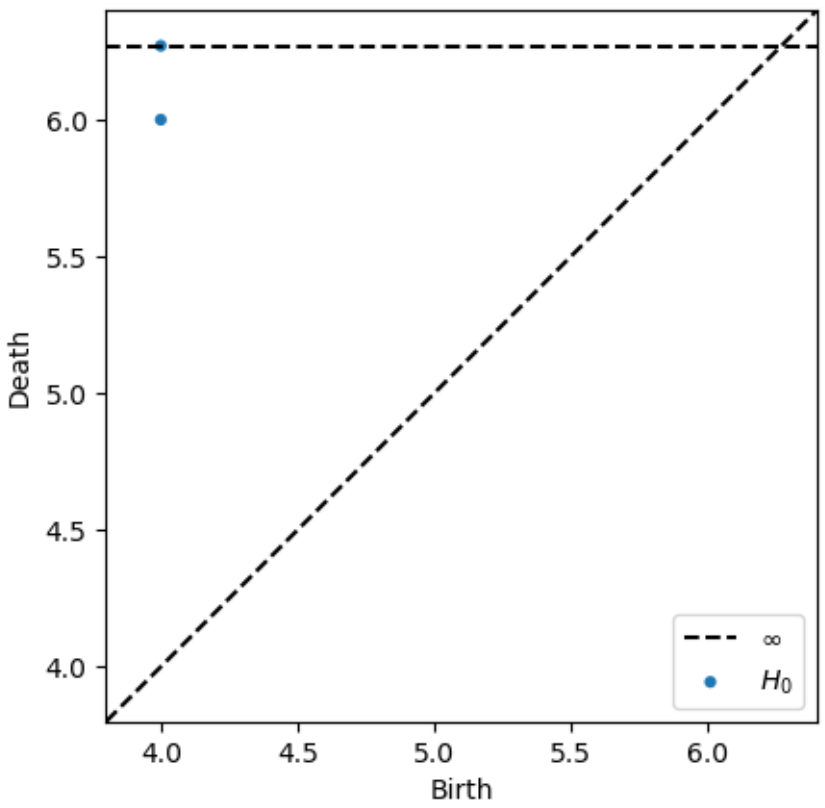
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 4.  6.]
 [ 4. inf]]
```



[38]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

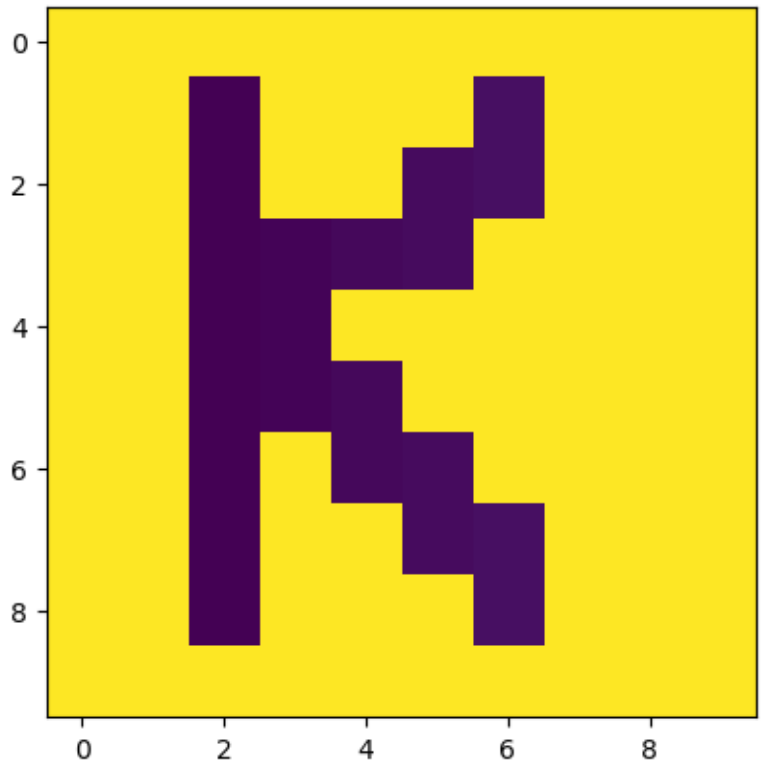
```
(2, 2)
[[ 4.  6.]
 [ 4. inf]]
```



[39]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[10,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

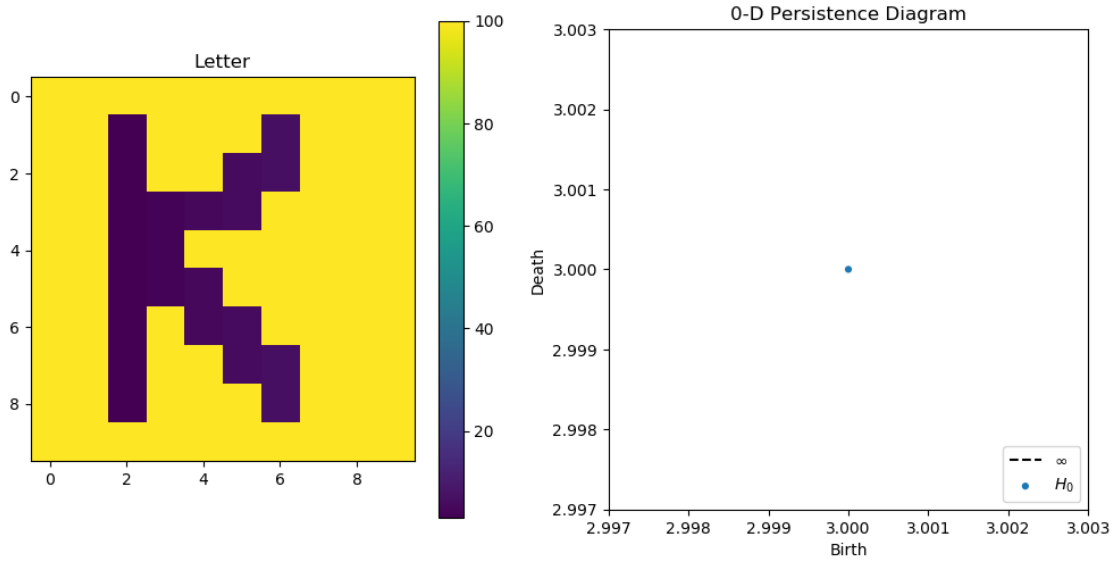
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100   3 100 100   6   7 100 100 100]
 [100 100   3   4   5   6 100 100 100 100]
 [100 100   3   4 100 100 100 100 100 100]
 [100 100   3   4   5 100 100 100 100 100]
 [100 100   3 100   5   6 100 100 100 100]
 [100 100   3 100 100   6   7 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[40]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

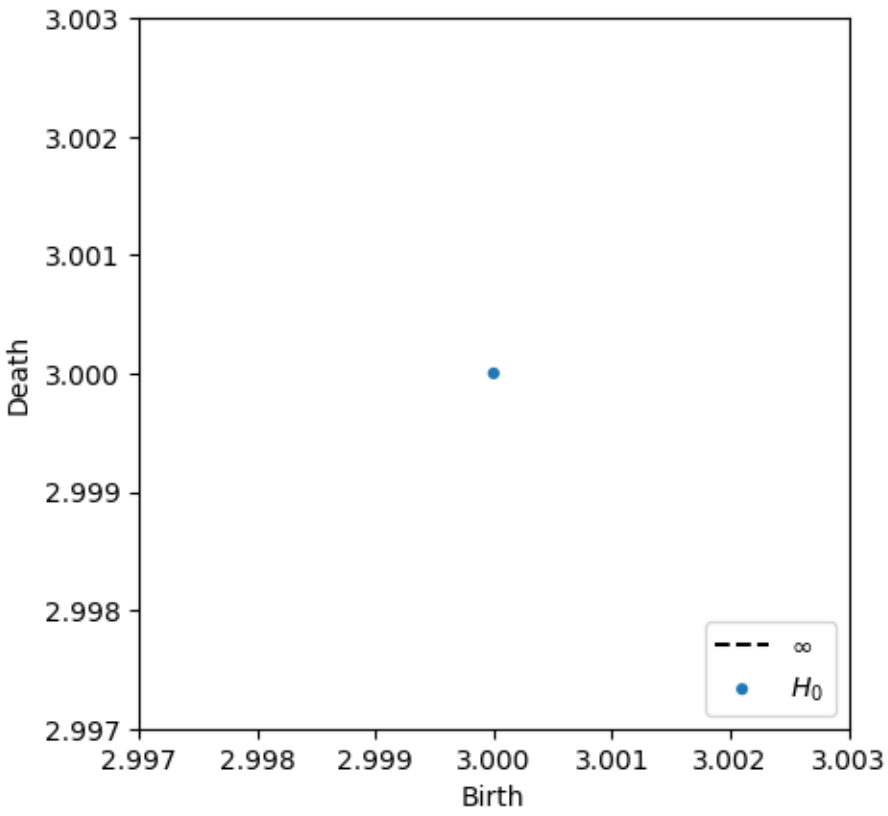
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3. inf]]
```



[41]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

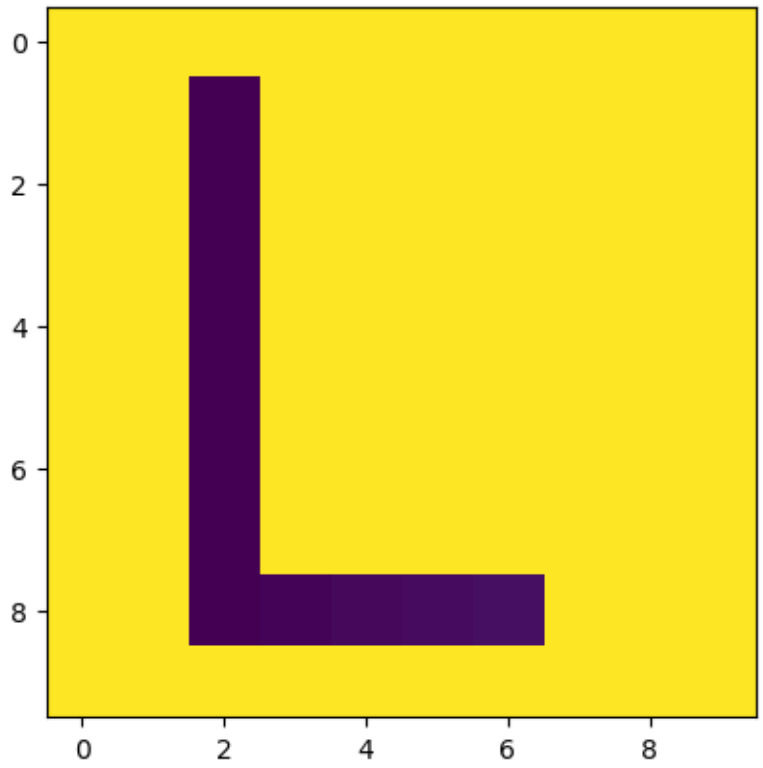
```
(1, 2)
[[ 3. inf]]
```



[42]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[11,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

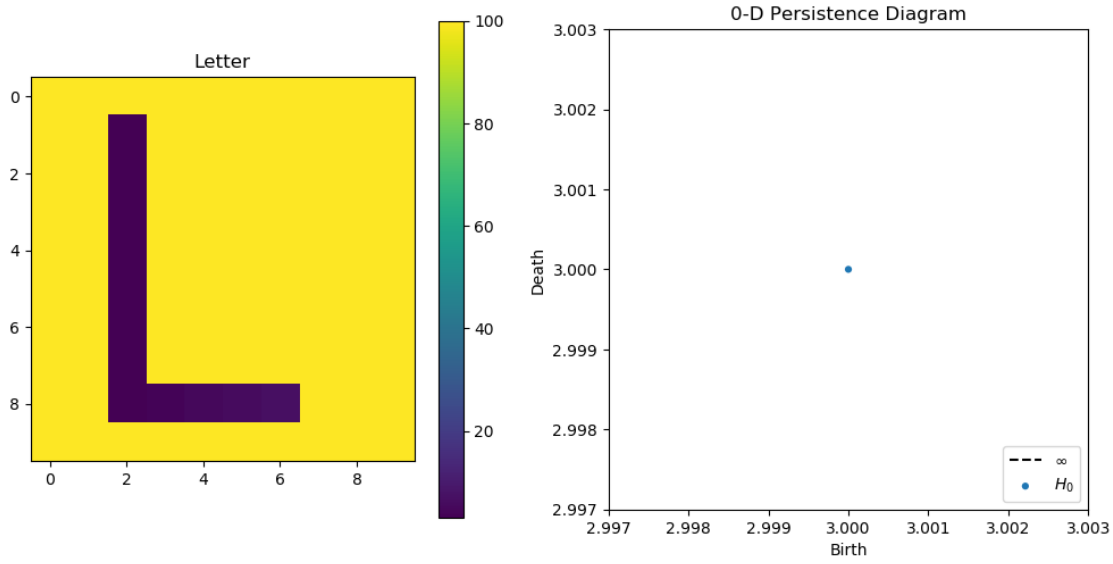
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[43]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

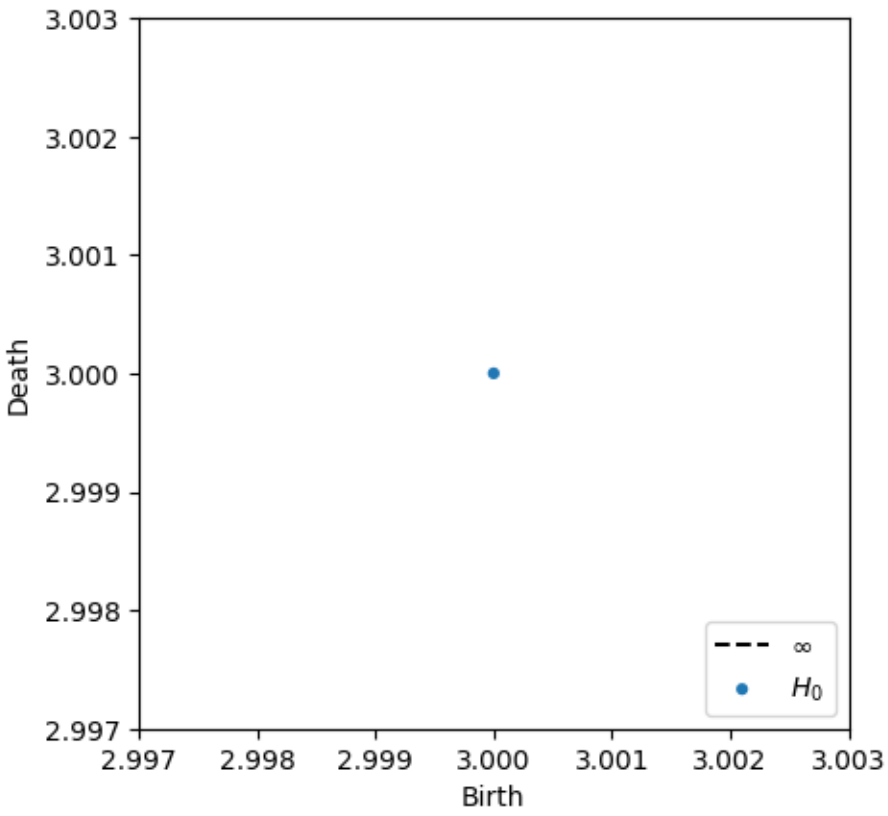
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3. inf]]
```



[44]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

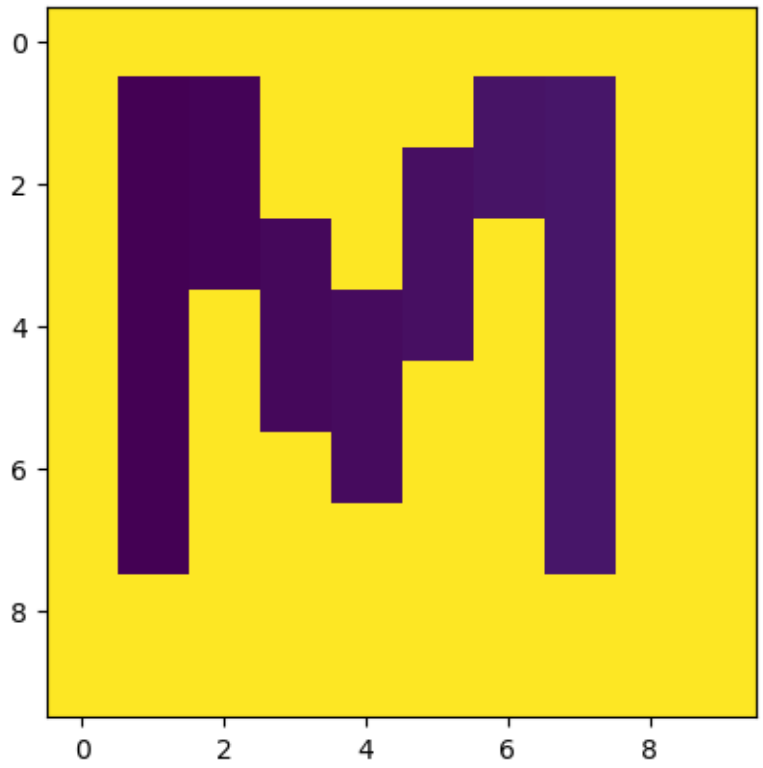
```
(1, 2)
[[ 3. inf]]
```



[45]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[12,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

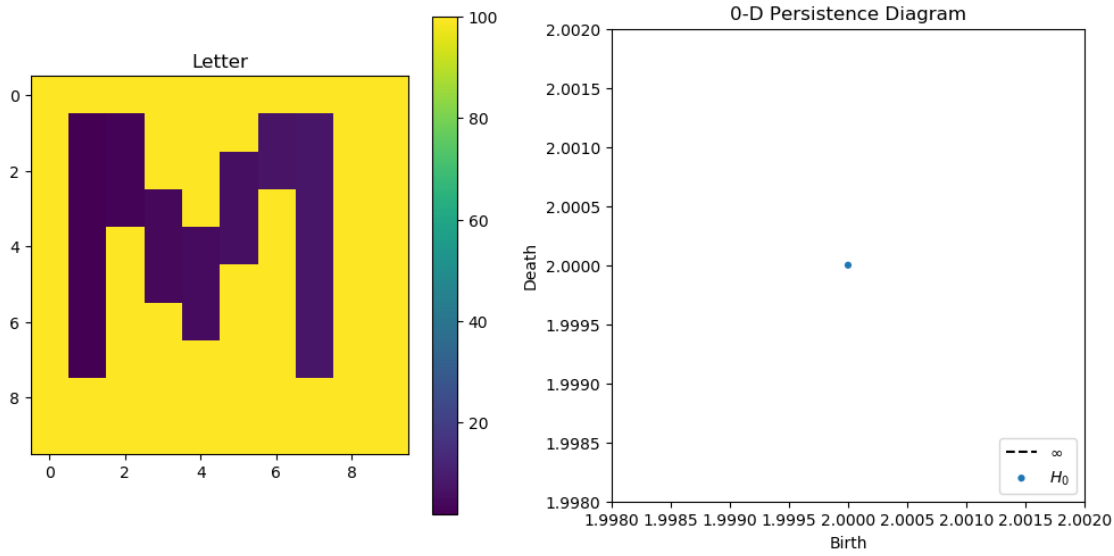
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100   2   3 100 100 100   7   8 100 100]
 [100   2   3 100 100   6   7   8 100 100]
 [100   2   3   4 100   6 100   8 100 100]
 [100   2 100   4   5   6 100   8 100 100]
 [100   2 100   4   5 100 100   8 100 100]
 [100   2 100 100   5 100 100   8 100 100]
 [100   2 100 100 100 100 100   8 100 100]
 [100 100 100 100 100 100 100 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[46]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

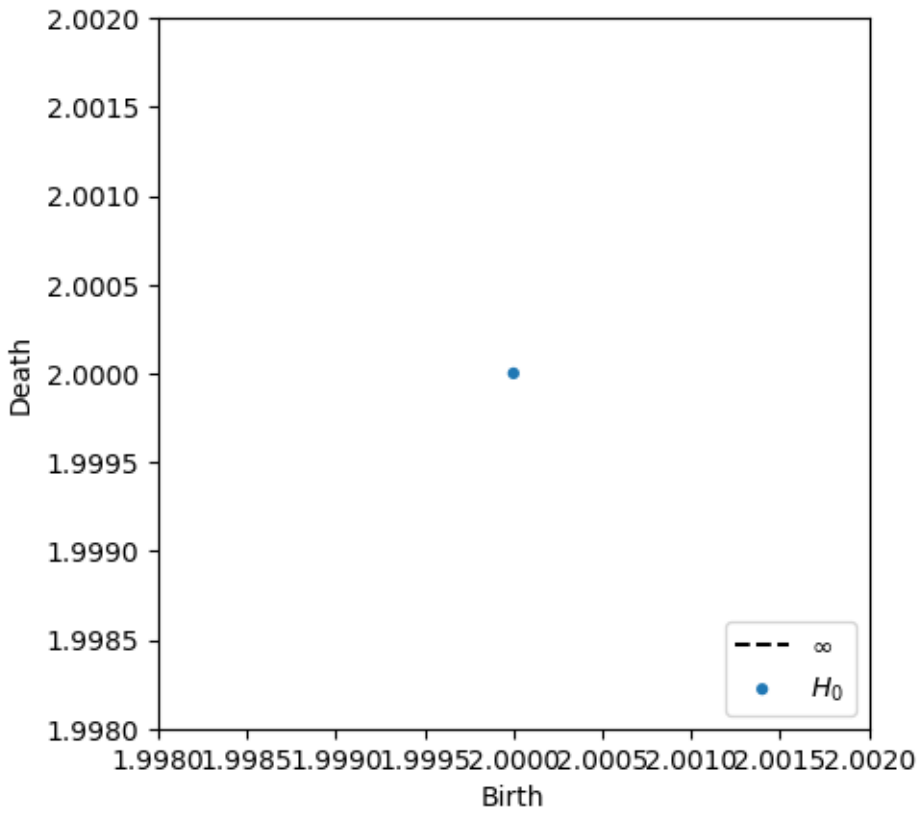
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 2. inf]]
```



[47]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

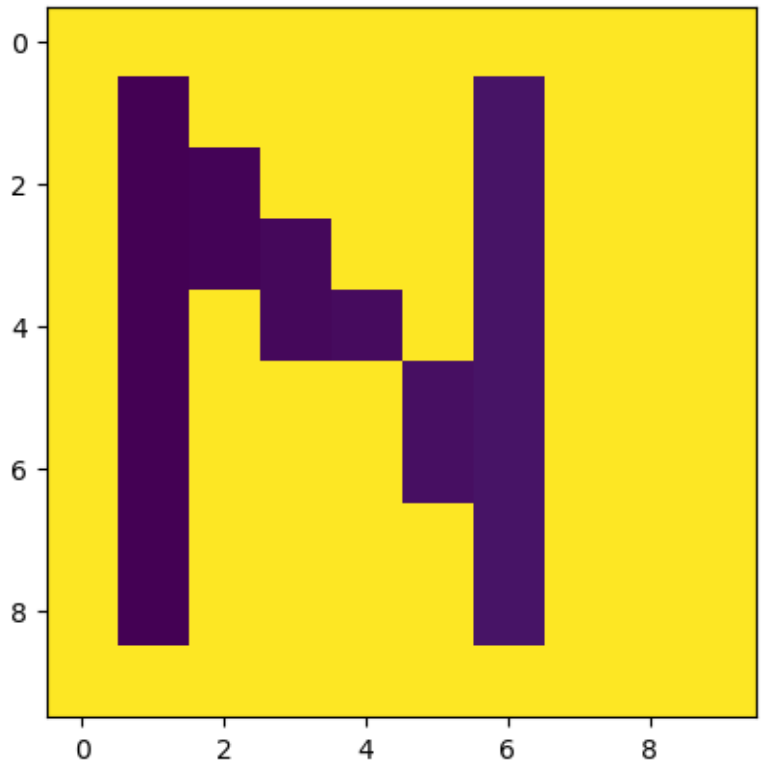
```
(1, 2)
[[ 2. inf]]
```



[48]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[13,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

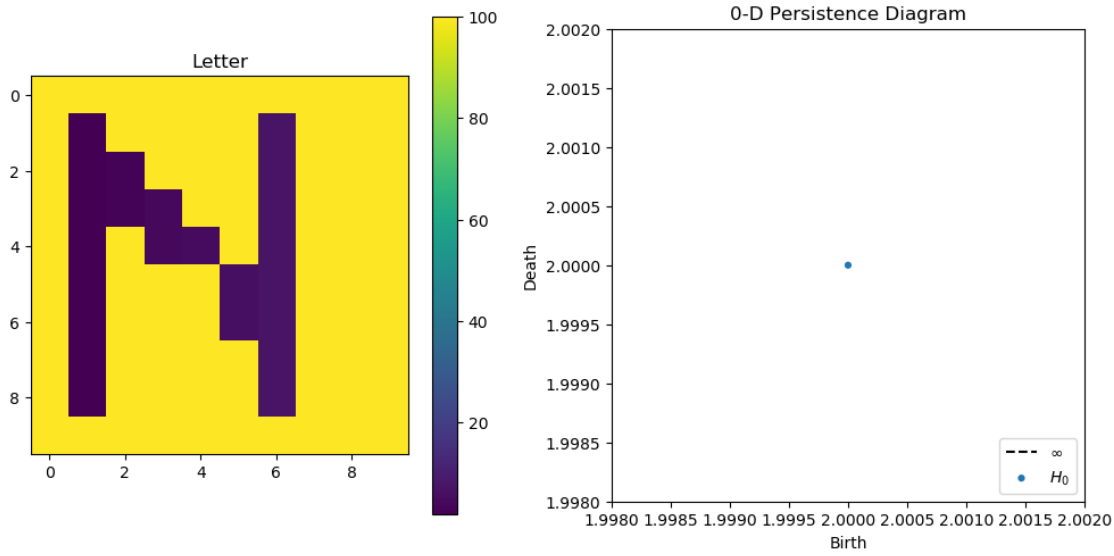
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100   2 100 100 100 100   7 100 100 100]
 [100   2   3 100 100 100   7 100 100 100]
 [100   2   3   4 100 100   7 100 100 100]
 [100   2 100   4   5 100   7 100 100 100]
 [100   2 100 100 100   6   7 100 100 100]
 [100   2 100 100 100   6   7 100 100 100]
 [100   2 100 100 100 100   7 100 100 100]
 [100   2 100 100 100 100   7 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[49]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

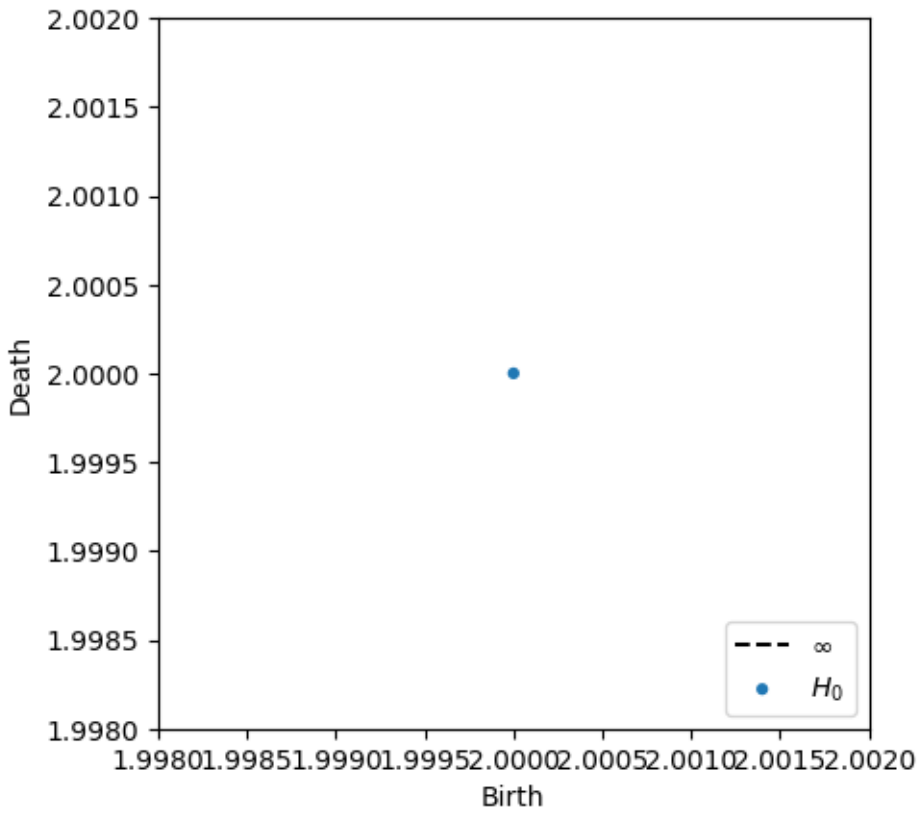
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 2. inf]]
```



[50]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

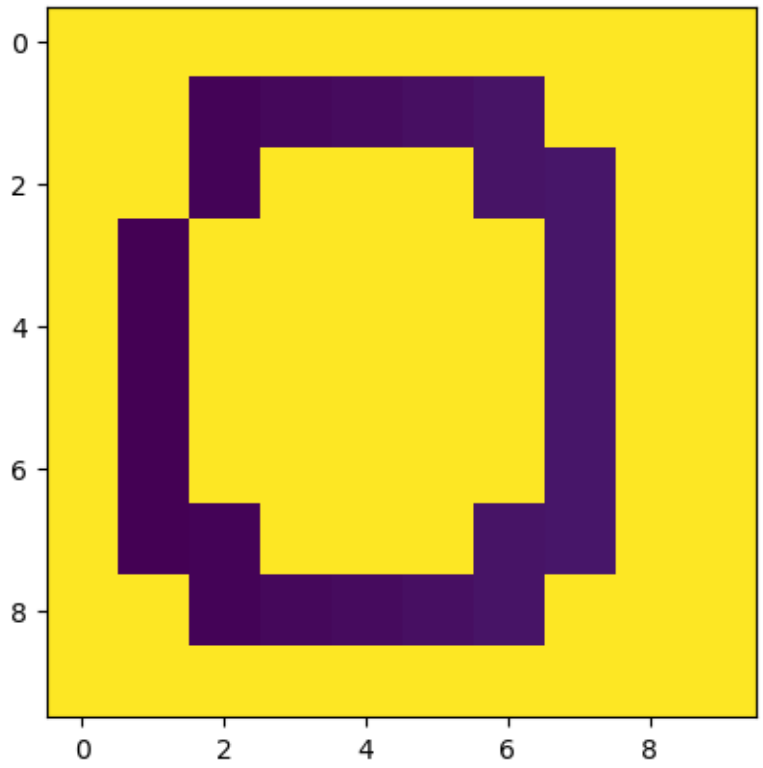
```
(1, 2)
[[ 2. inf]]
```



[51]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[14,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

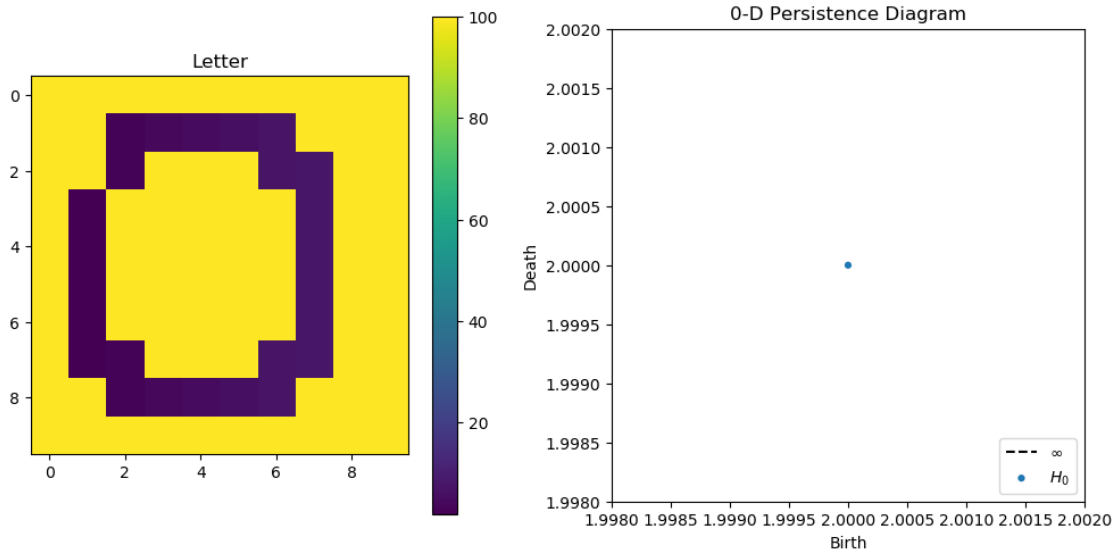
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100   3 100 100 100   7   8 100 100]
 [100   2 100 100 100 100 100   8 100 100]
 [100   2 100 100 100 100 100   8 100 100]
 [100   2 100 100 100 100 100   8 100 100]
 [100   2 100 100 100 100 100   8 100 100]
 [100   2   3 100 100 100   7   8 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[52]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

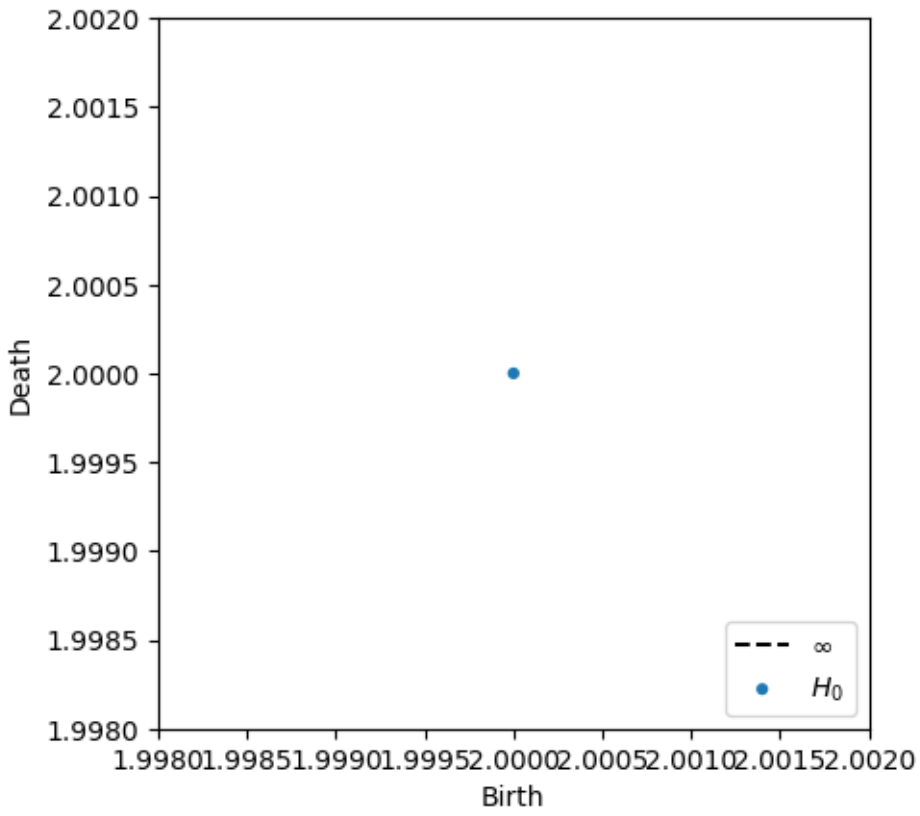
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 2. inf]]
```



[53]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

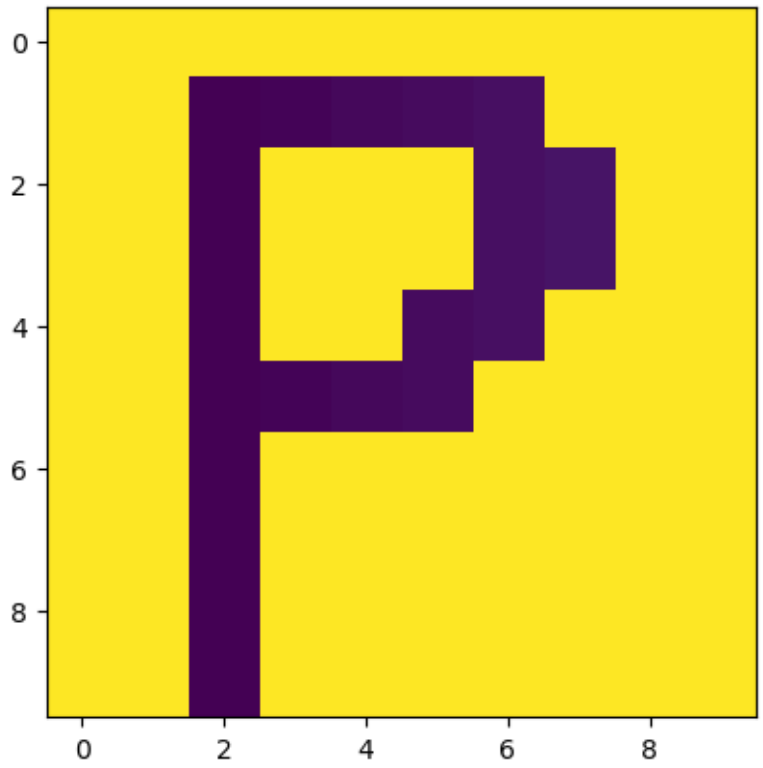
```
(1, 2)
[[ 2. inf]]
```



[54]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[15,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

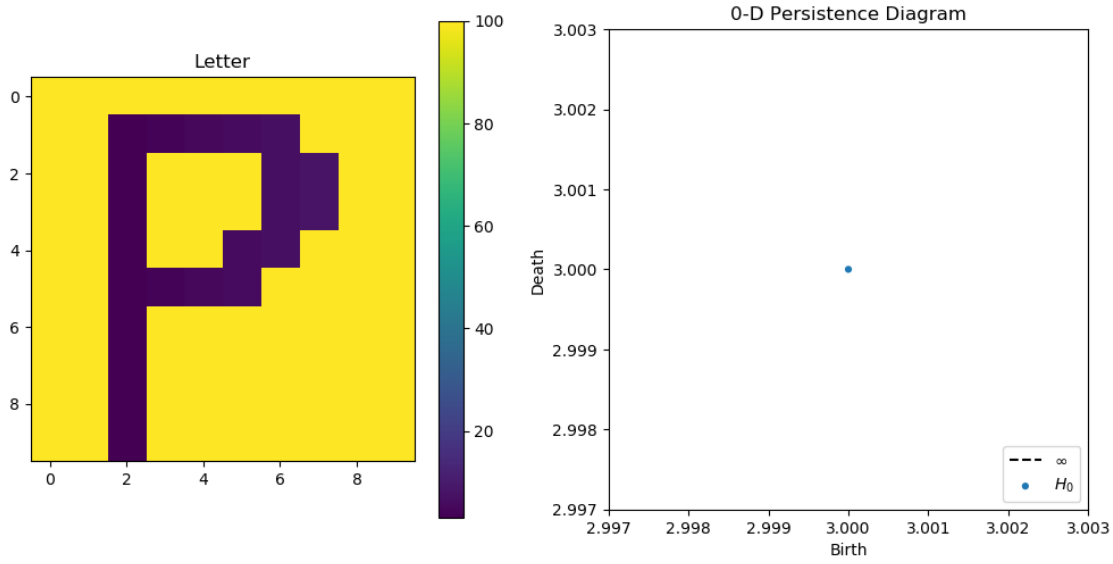
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7 100 100 100]
 [100 100   3 100 100 100   7   8 100 100]
 [100 100   3 100 100 100   7   8 100 100]
 [100 100   3 100 100   6   7 100 100 100]
 [100 100   3   4   5   6 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]]
```



[55]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

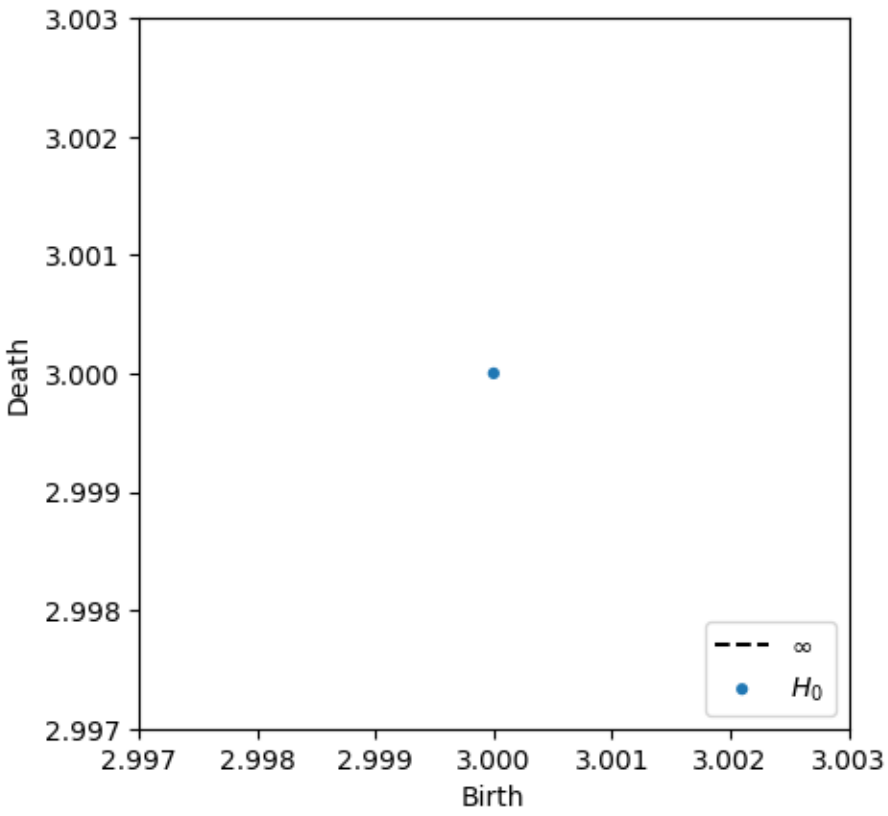
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3. inf]]
```



[56]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

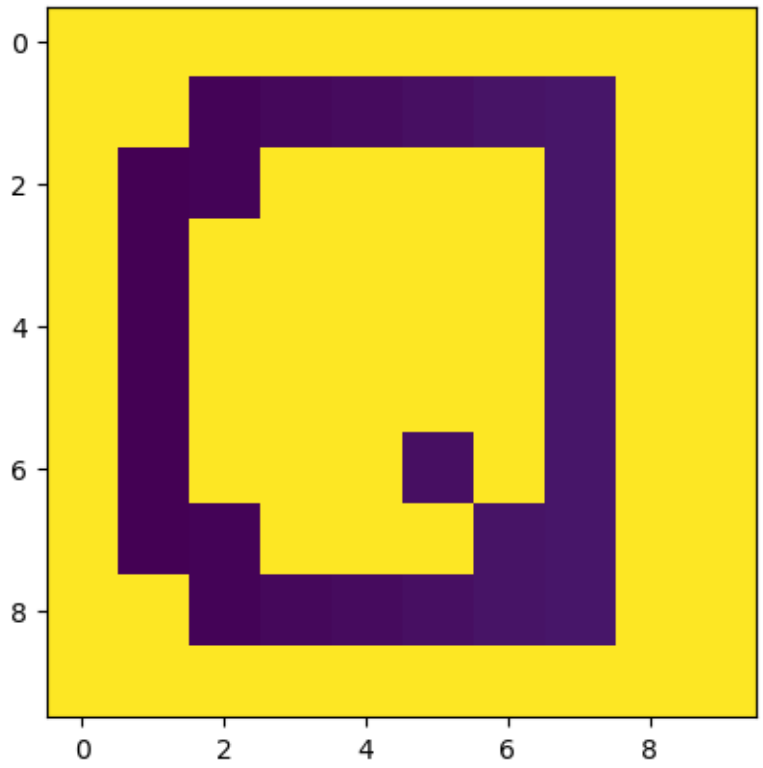
```
(1, 2)
[[ 3. inf]]
```



[57]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[16,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

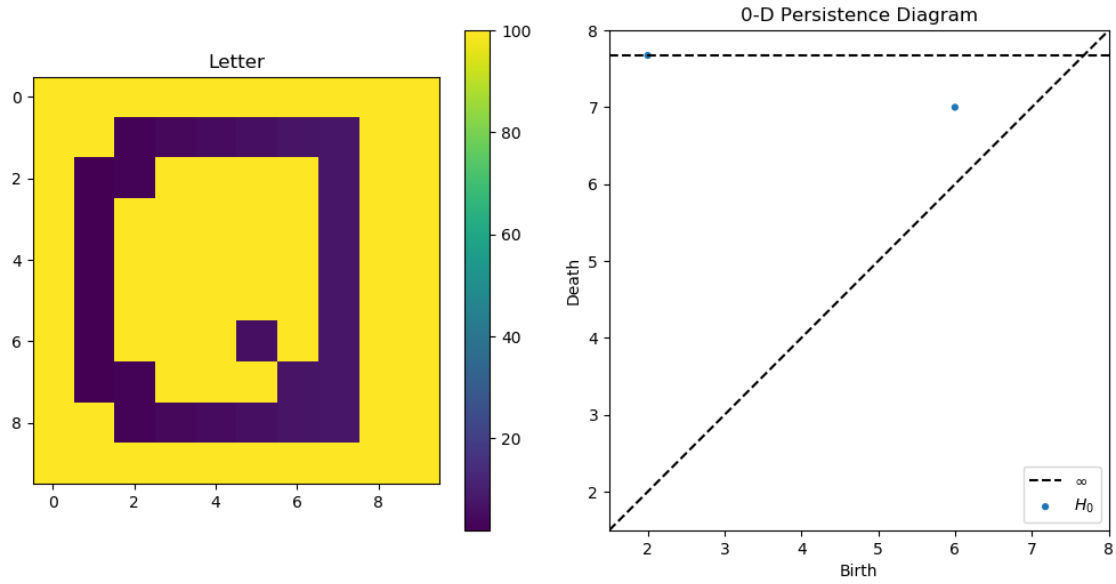
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100   2   3 100 100 100 100   8 100 100]
 [100   2 100 100 100 100 100   8 100 100]
 [100   2 100 100 100 100 100   8 100 100]
 [100   2 100 100 100 100 100   8 100 100]
 [100   2 100 100 100   6 100   8 100 100]
 [100   2   3 100 100 100   7   8 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[58]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

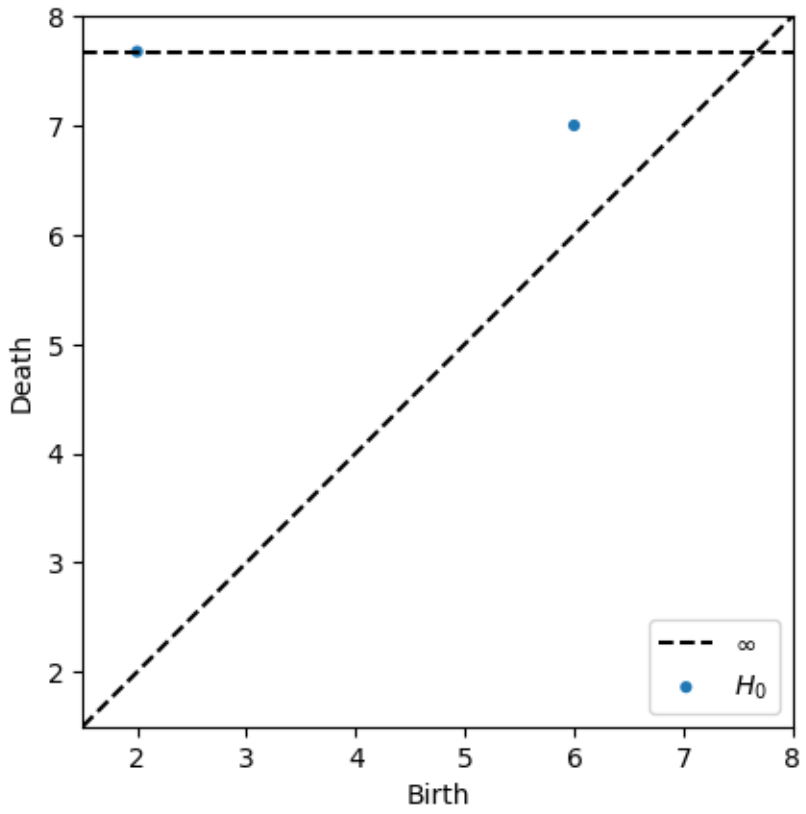
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 6.  7.]
 [ 2. inf]]
```



[59]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

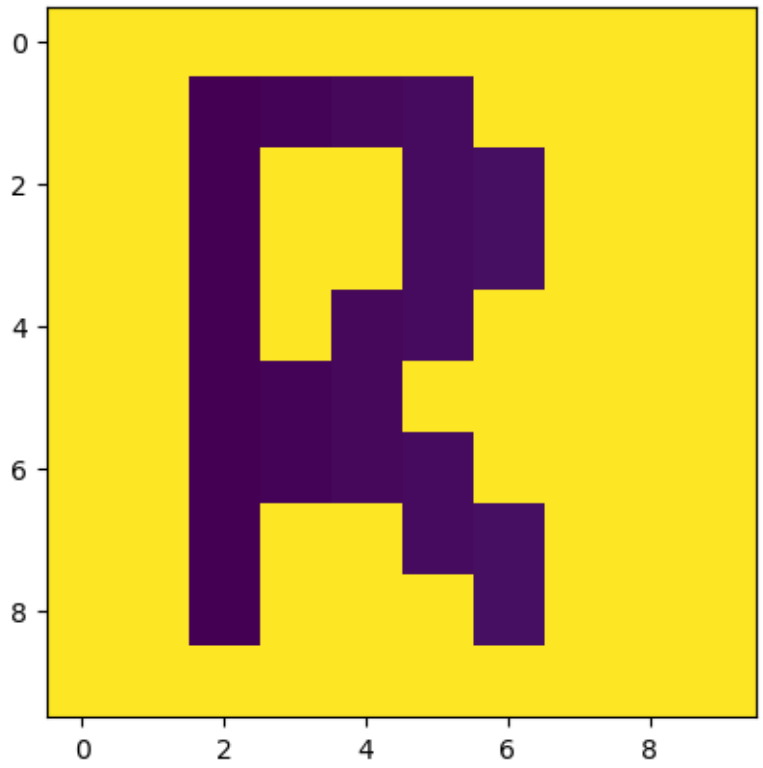
```
(2, 2)
[[ 6.  7.]
 [ 2. inf]]
```



[60]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[17,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

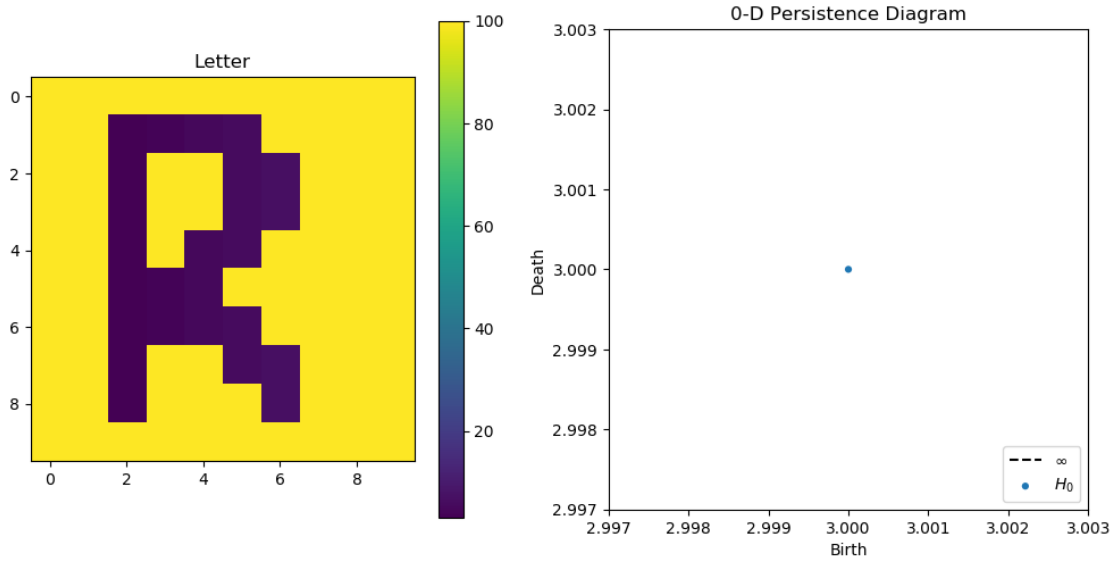
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3   4   5   6 100 100 100 100]
 [100 100   3 100 100   6   7 100 100 100]
 [100 100   3 100 100   6   7 100 100 100]
 [100 100   3 100   5   6 100 100 100 100]
 [100 100   3   4   5 100 100 100 100 100]
 [100 100   3   4   5   6 100 100 100 100]
 [100 100   3 100 100   6   7 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[61]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

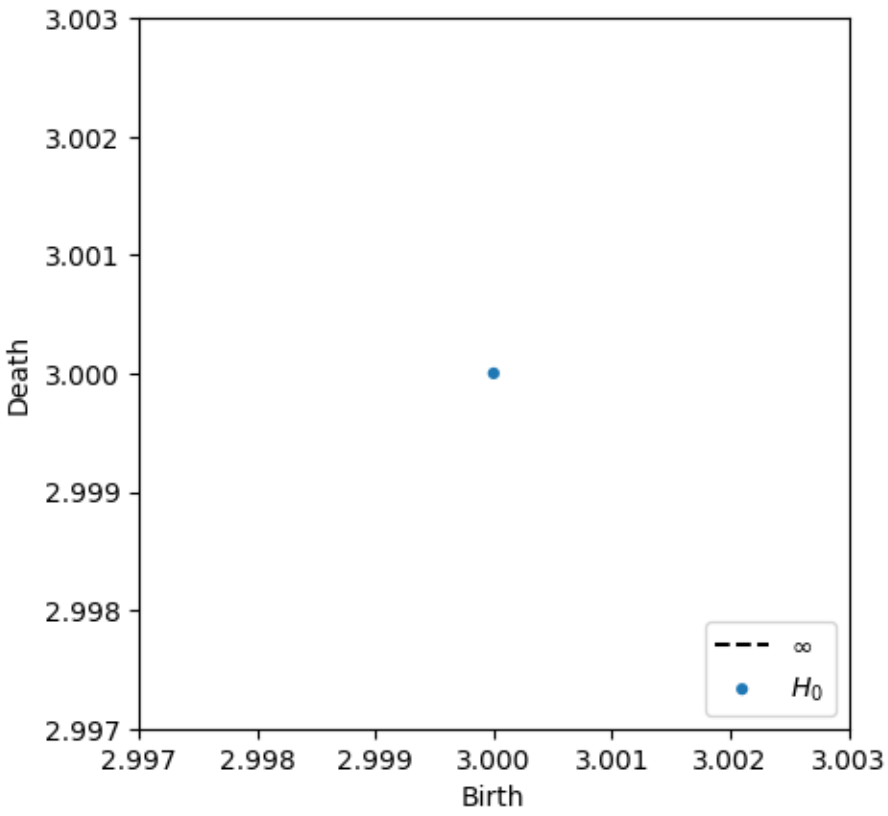
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3. inf]]
```



[62]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

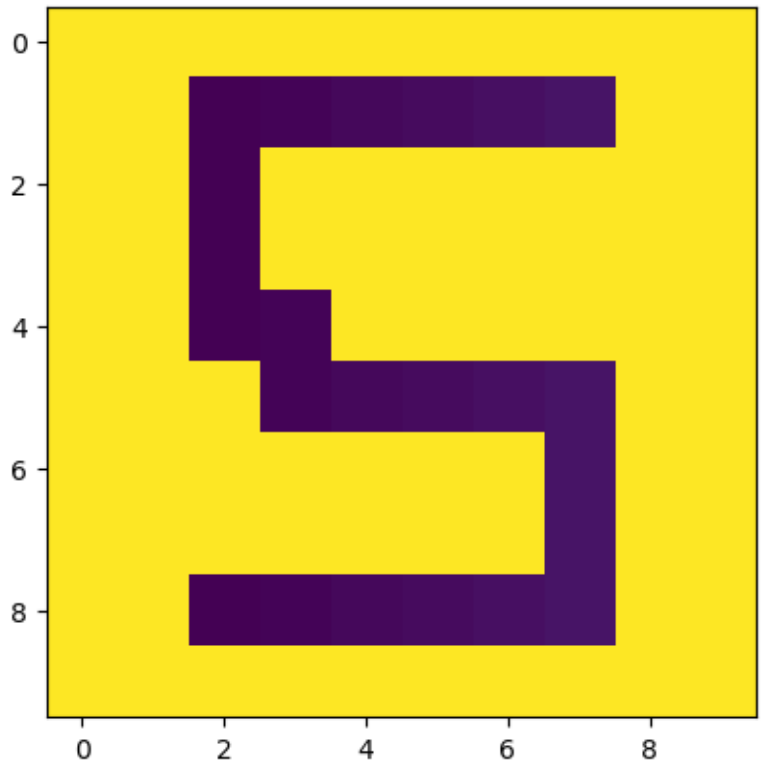
```
(1, 2)
[[ 3. inf]]
```



[63]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[18,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

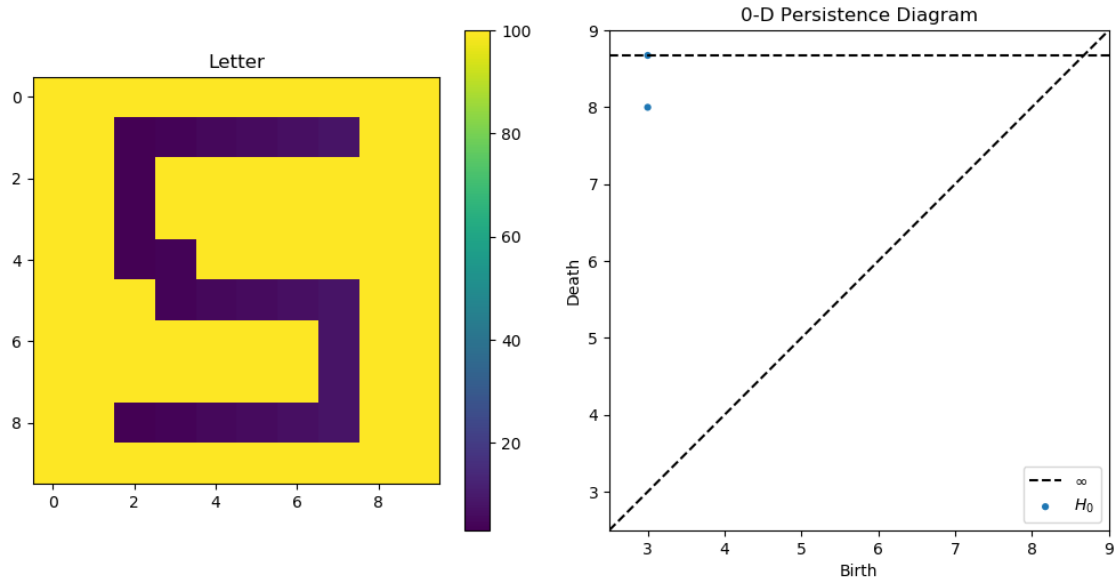
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100 100 100 100]
 [100 100   3   4 100 100 100 100 100 100]
 [100 100 100   4   5   6   7   8 100 100]
 [100 100 100 100 100 100 100   8 100 100]
 [100 100 100 100 100 100 100   8 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[64]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

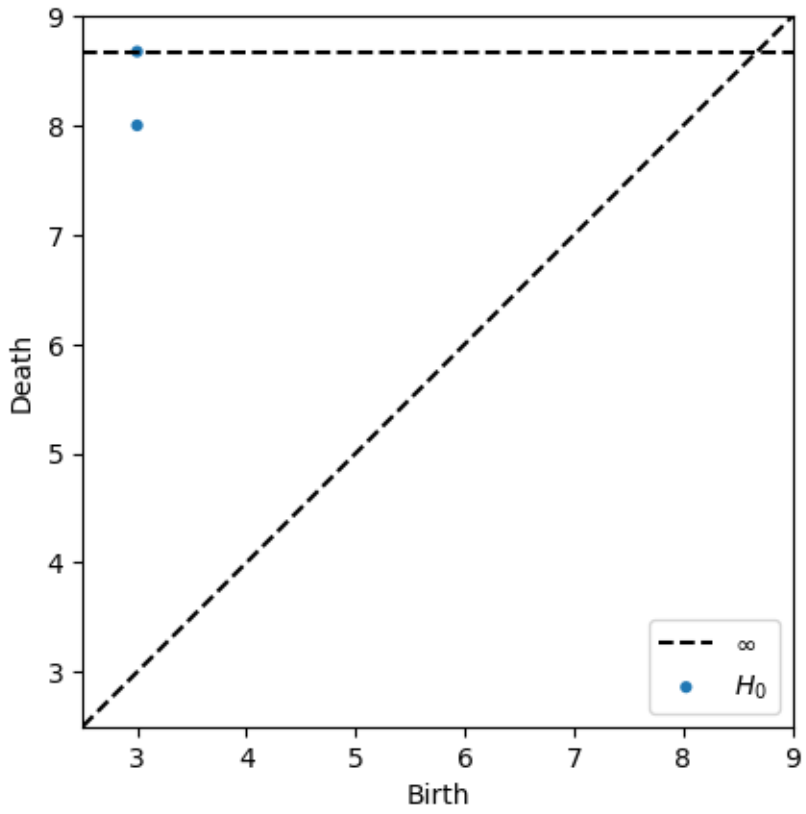
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3.  8.]
 [ 3. inf]]
```



[65]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

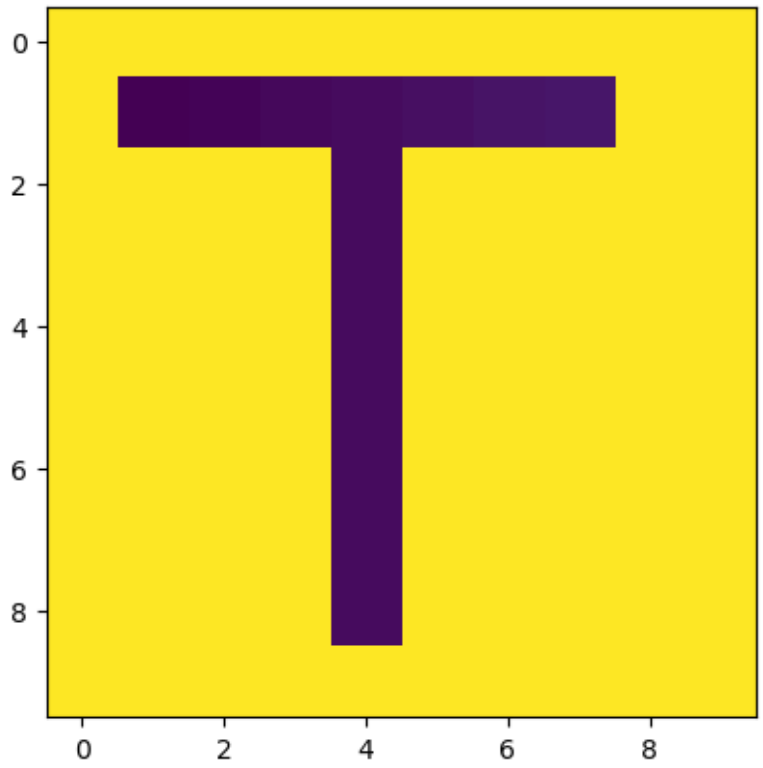
```
(2, 2)
[[ 3.  8.]
 [ 3. inf]]
```



[66]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[19,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

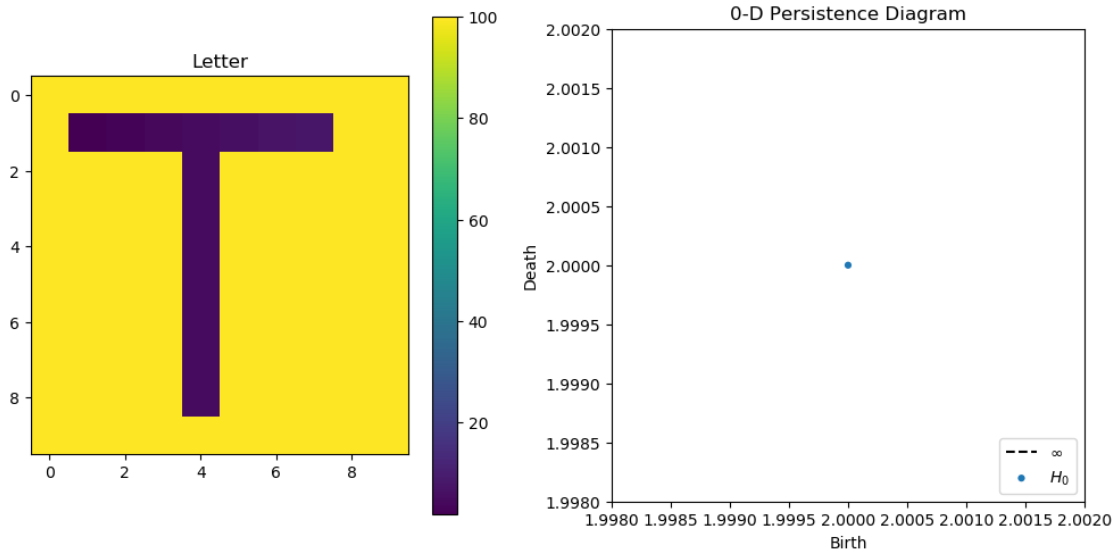
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100   2   3   4   5   6   7   8 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[67]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

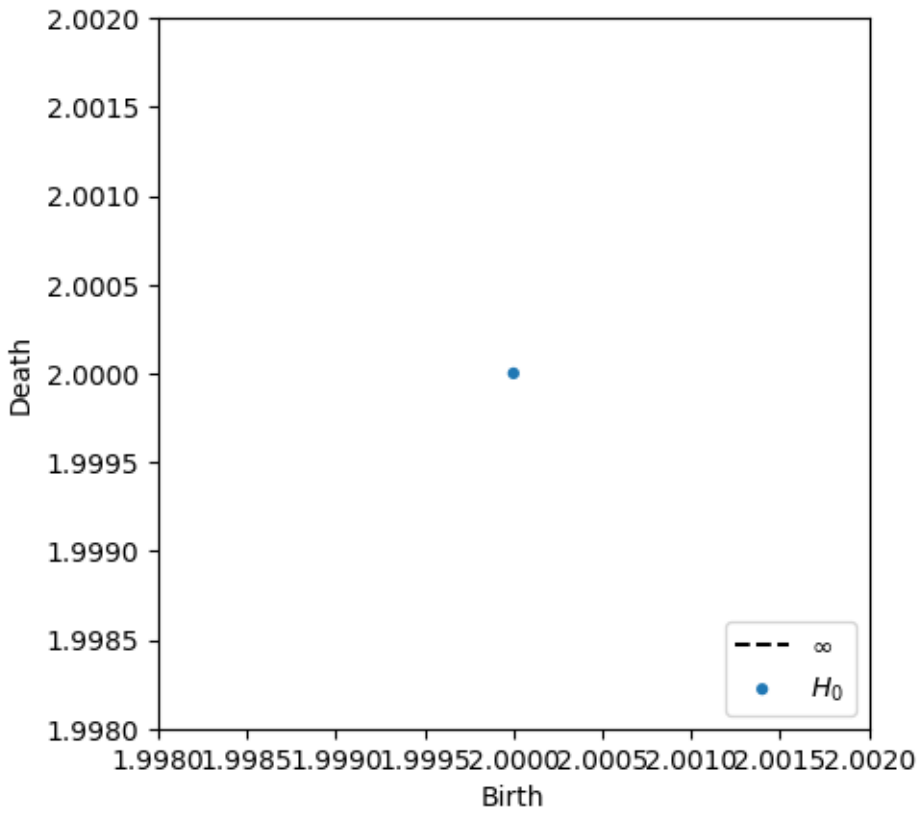
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 2. inf]]
```



[68]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

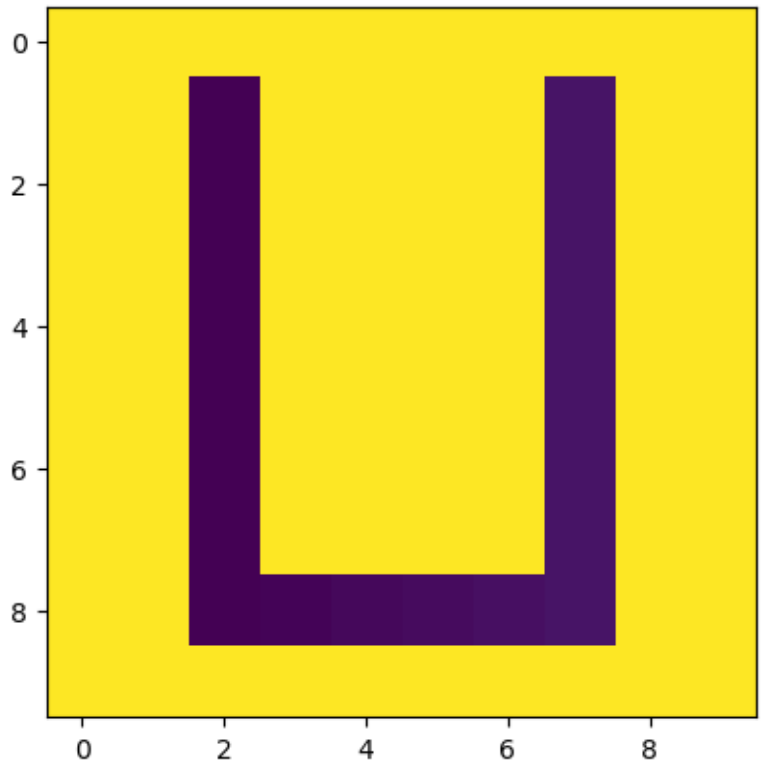
```
(1, 2)
[[ 2. inf]]
```



[69]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[20,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

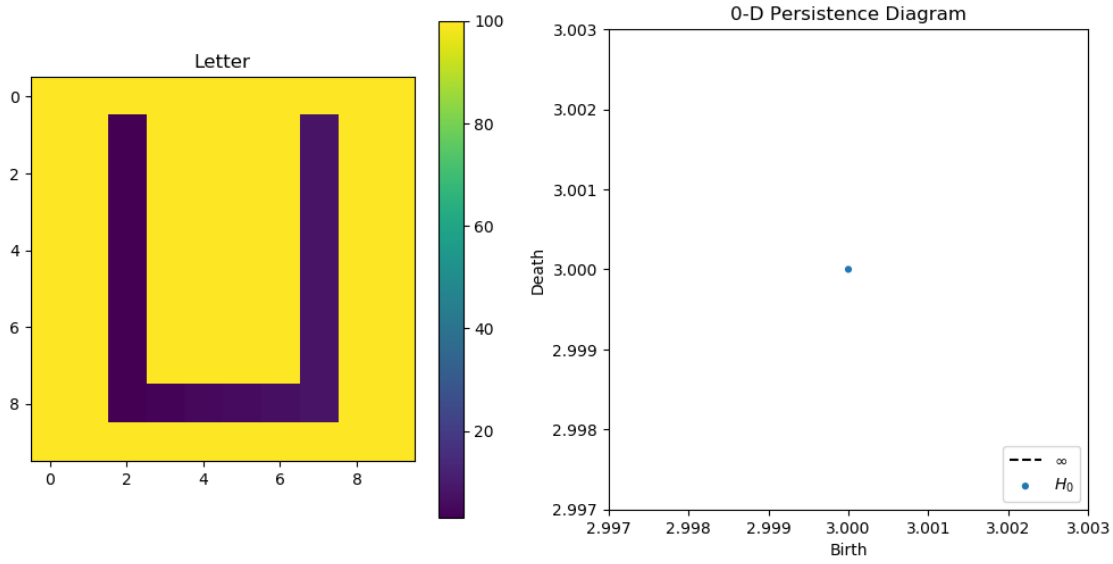
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3 100 100 100 100   8 100 100]
 [100 100   3 100 100 100 100   8 100 100]
 [100 100   3 100 100 100 100   8 100 100]
 [100 100   3 100 100 100 100   8 100 100]
 [100 100   3 100 100 100 100   8 100 100]
 [100 100   3 100 100 100 100   8 100 100]
 [100 100   3 100 100 100 100   8 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[70]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

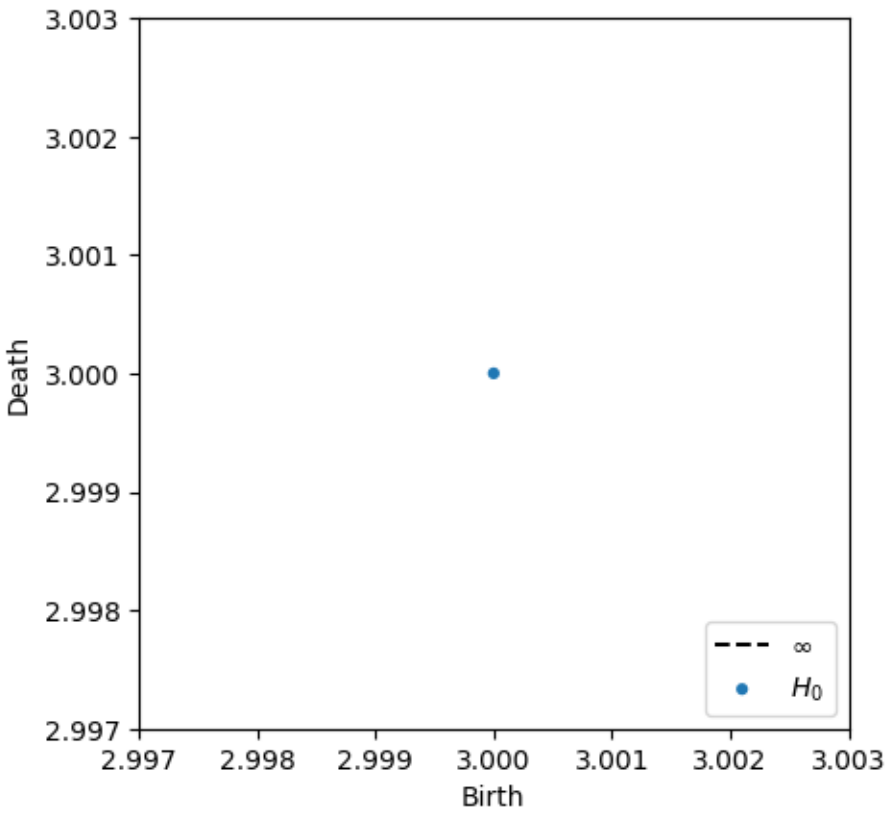
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3. inf]]
```



[71]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

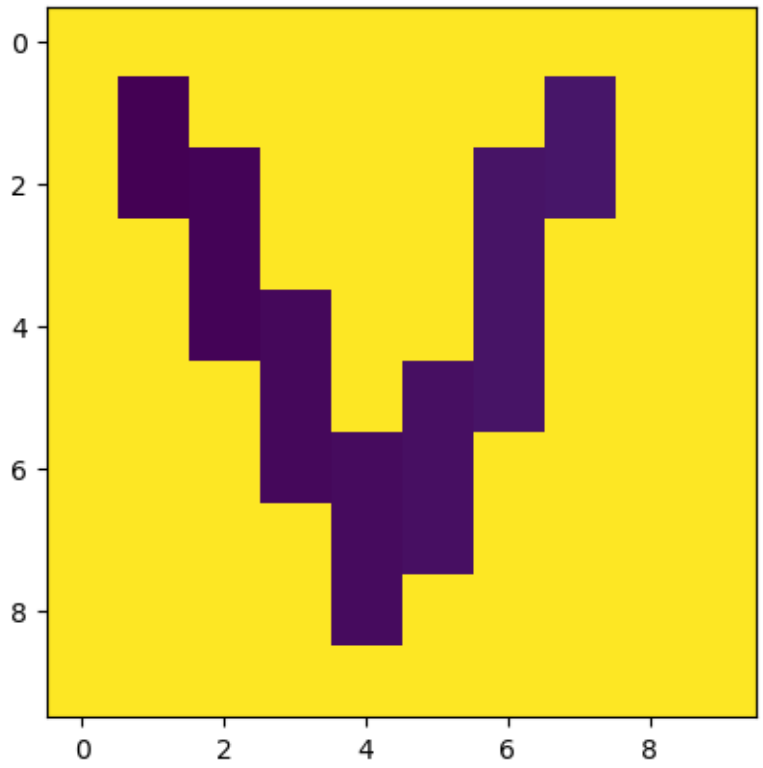
```
(1, 2)
[[ 3. inf]]
```



[72]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[21,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

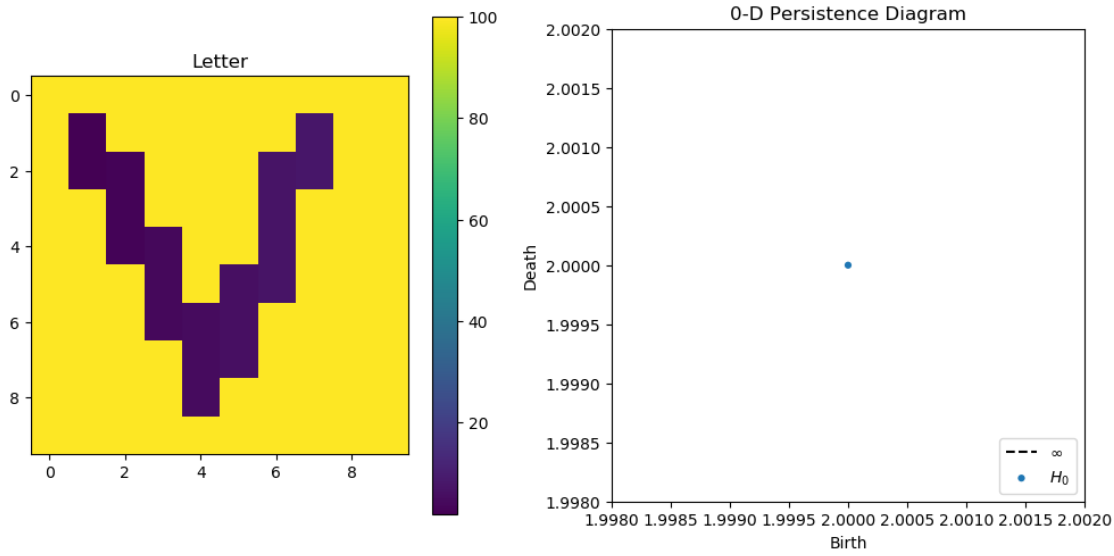
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100   2 100 100 100 100 100   8 100 100]
 [100   2   3 100 100 100   7   8 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100   3   4 100 100   7 100 100 100]
 [100 100 100   4 100   6   7 100 100 100]
 [100 100 100   4   5   6 100 100 100 100]
 [100 100 100 100   5   6 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[73]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

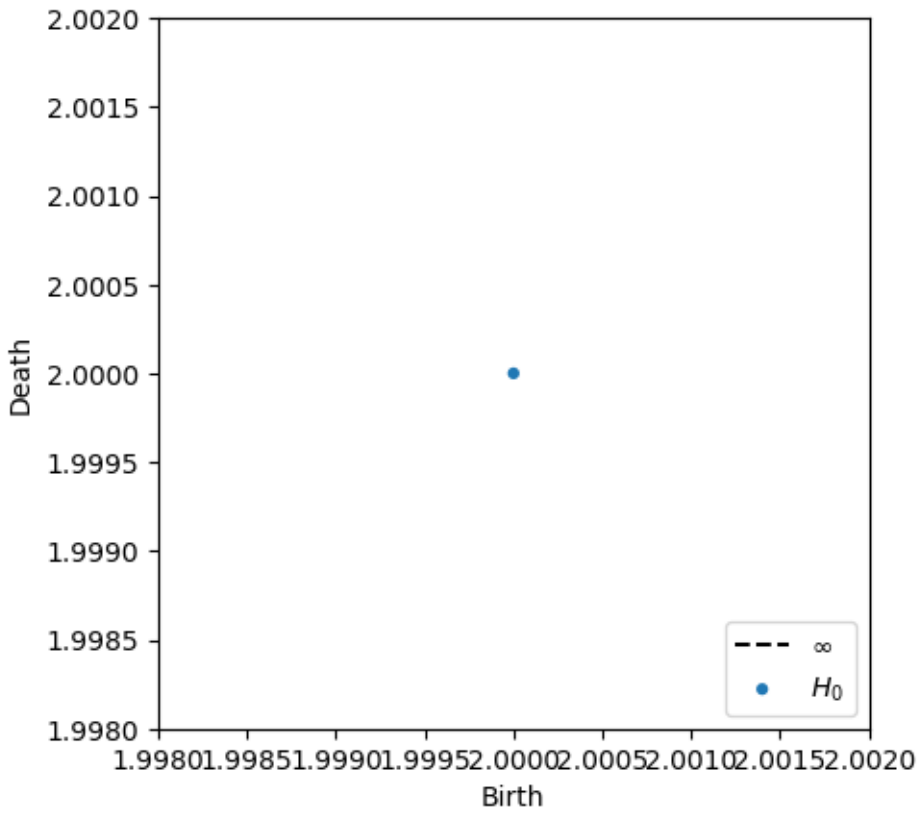
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 2. inf]]
```



[74]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

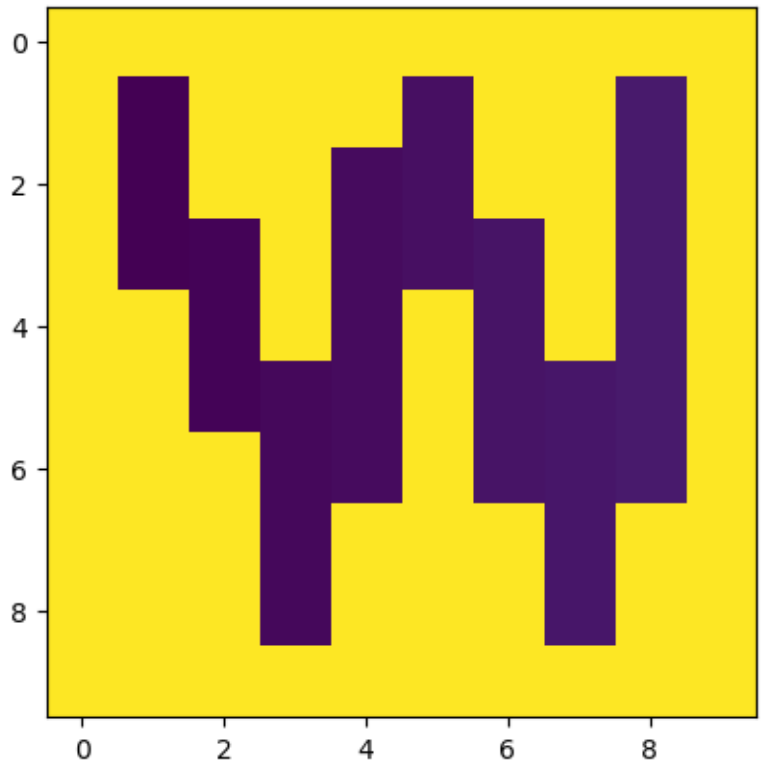
```
(1, 2)
[[ 2. inf]]
```



[75]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[22,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

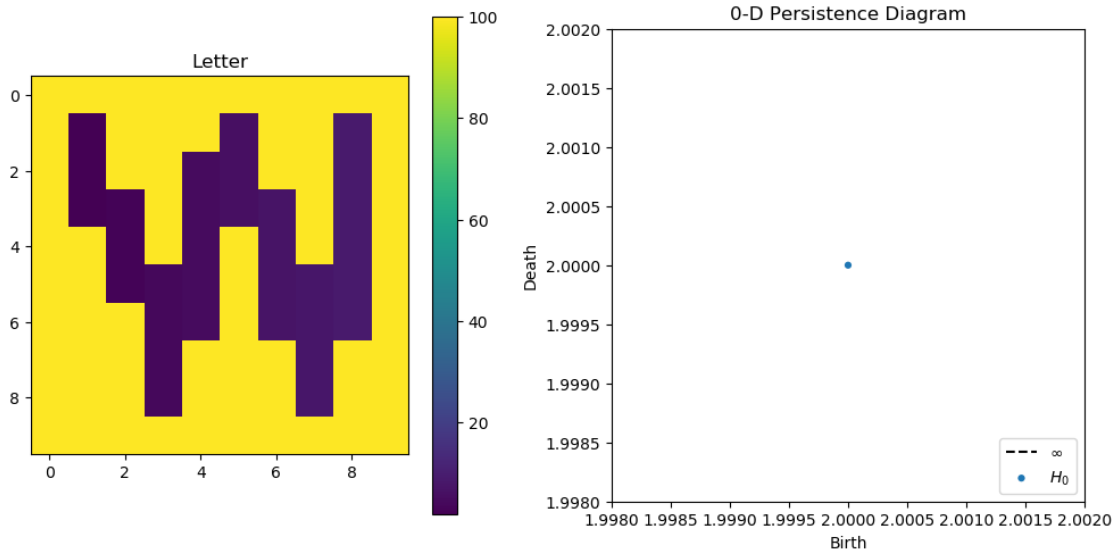
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100   2 100 100 100   6 100 100   9 100]
 [100   2 100 100   5   6 100 100   9 100]
 [100   2   3 100   5   6   7 100   9 100]
 [100 100   3 100   5 100   7 100   9 100]
 [100 100   3   4   5 100   7   8   9 100]
 [100 100 100   4   5 100   7   8   9 100]
 [100 100 100   4 100 100 100   8 100 100]
 [100 100 100   4 100 100 100   8 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[76]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

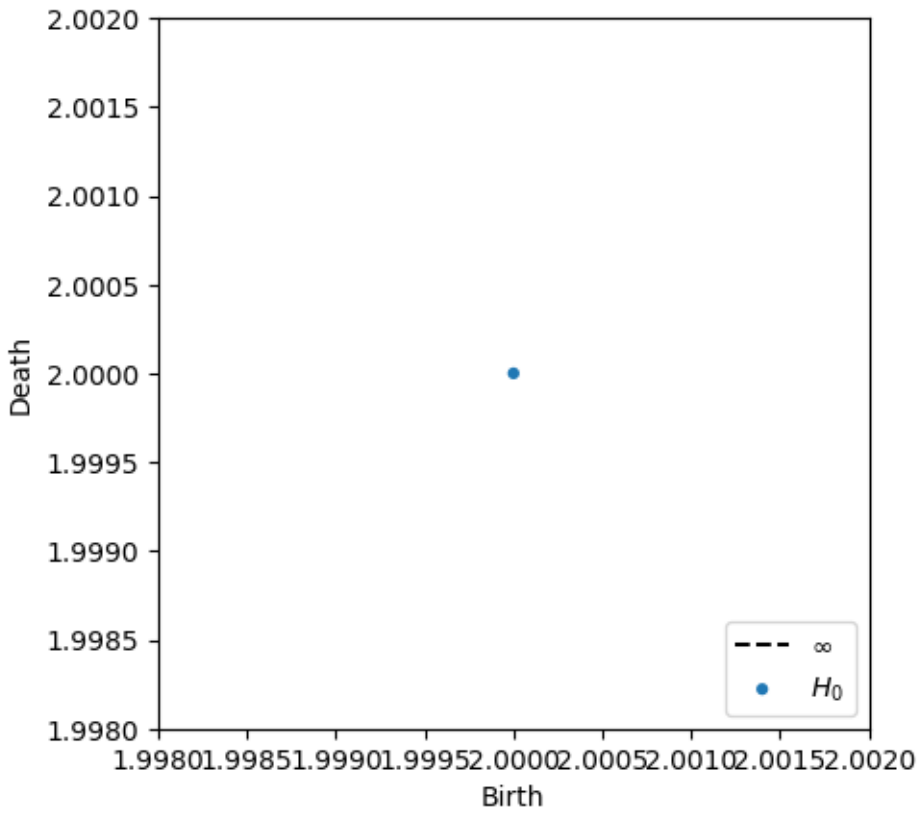
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 2. inf]]
```



[77]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

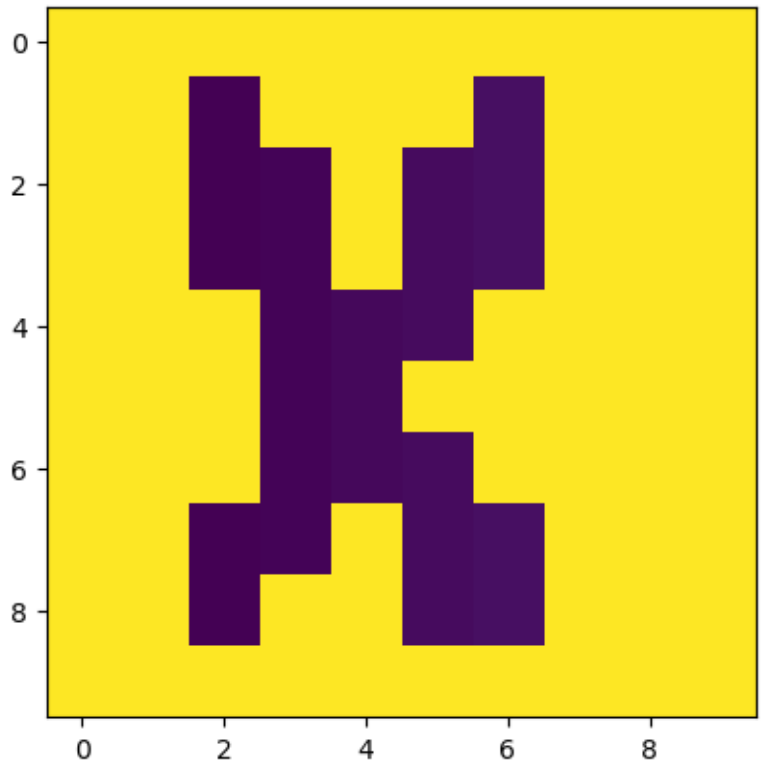
```
(1, 2)
[[ 2. inf]]
```



[78]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[23,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

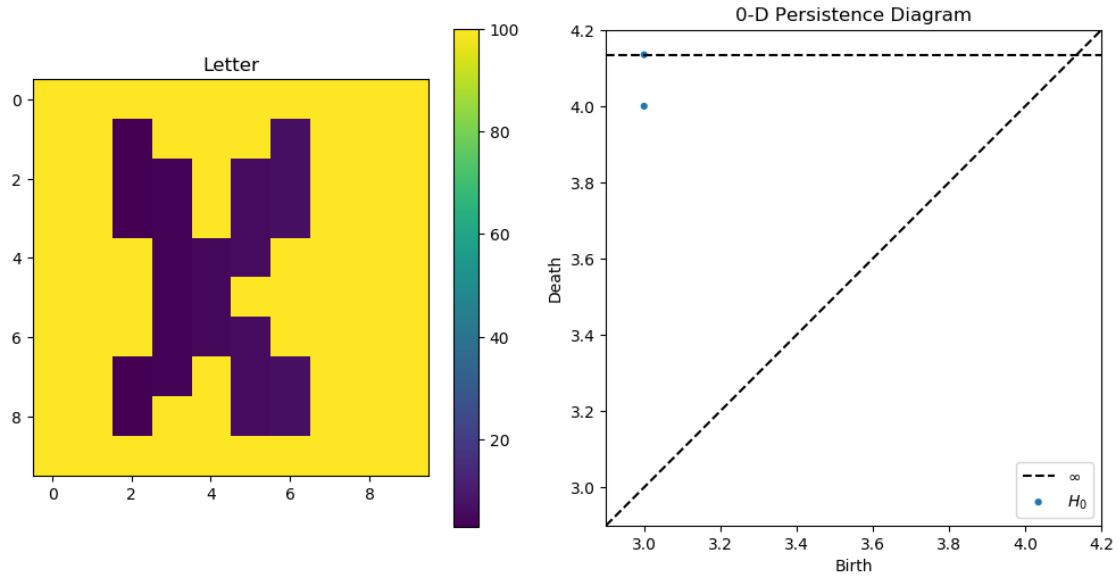
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100   3   4 100   6   7 100 100 100]
 [100 100   3   4 100   6   7 100 100 100]
 [100 100 100   4   5   6 100 100 100 100]
 [100 100 100   4   5 100 100 100 100 100]
 [100 100 100   4   5   6 100 100 100 100]
 [100 100   3   4 100   6   7 100 100 100]
 [100 100   3 100 100   6   7 100 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[79]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

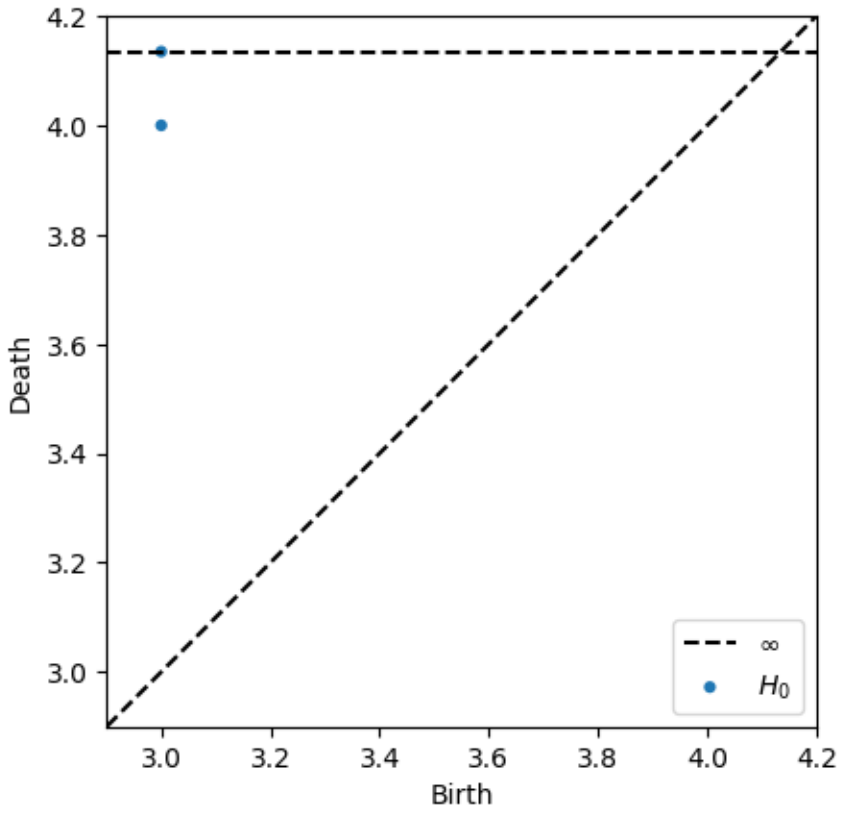
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3.  4.]
 [ 3. inf]]
```



[80]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

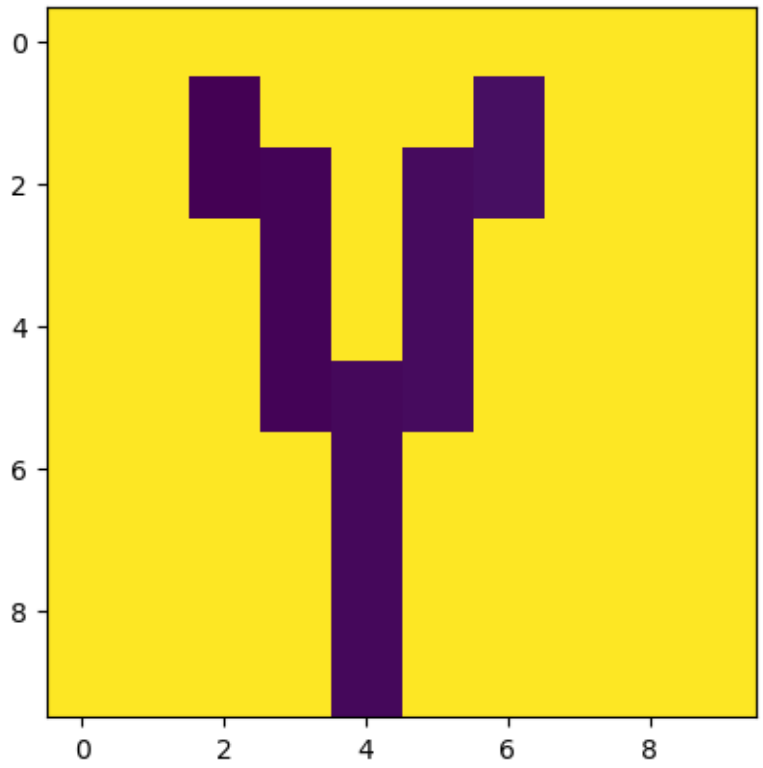
```
(2, 2)
[[ 3.  4.]
 [ 3. inf]]
```



[81]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[24,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

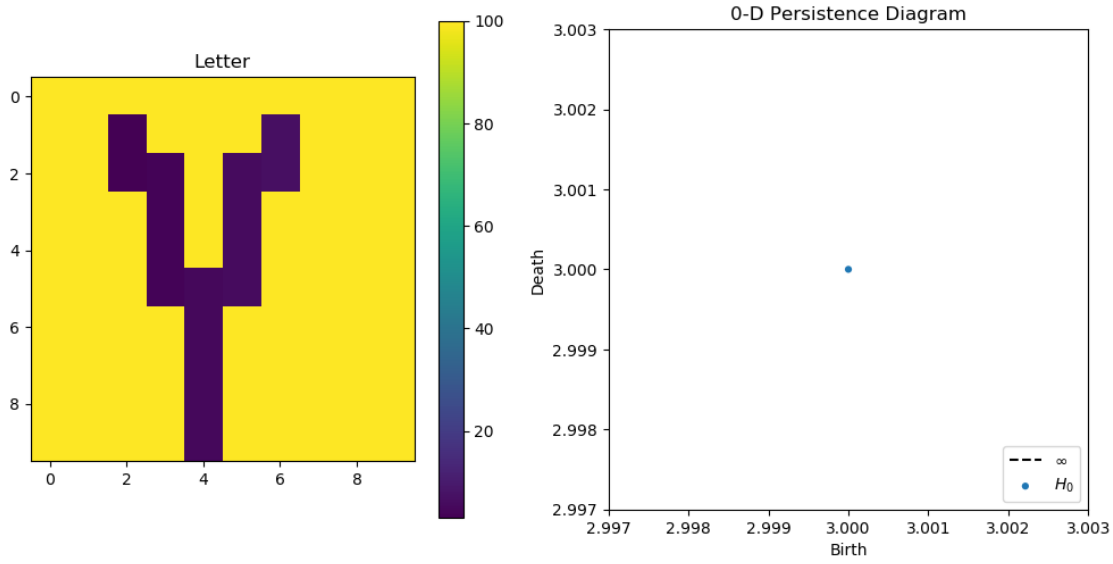
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100 100   3 100 100 100   7 100 100 100]
 [100 100   3   4 100   6   7 100 100 100]
 [100 100 100   4 100   6 100 100 100 100]
 [100 100 100   4 100   6 100 100 100 100]
 [100 100 100   4   5   6 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]]
```



[82]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

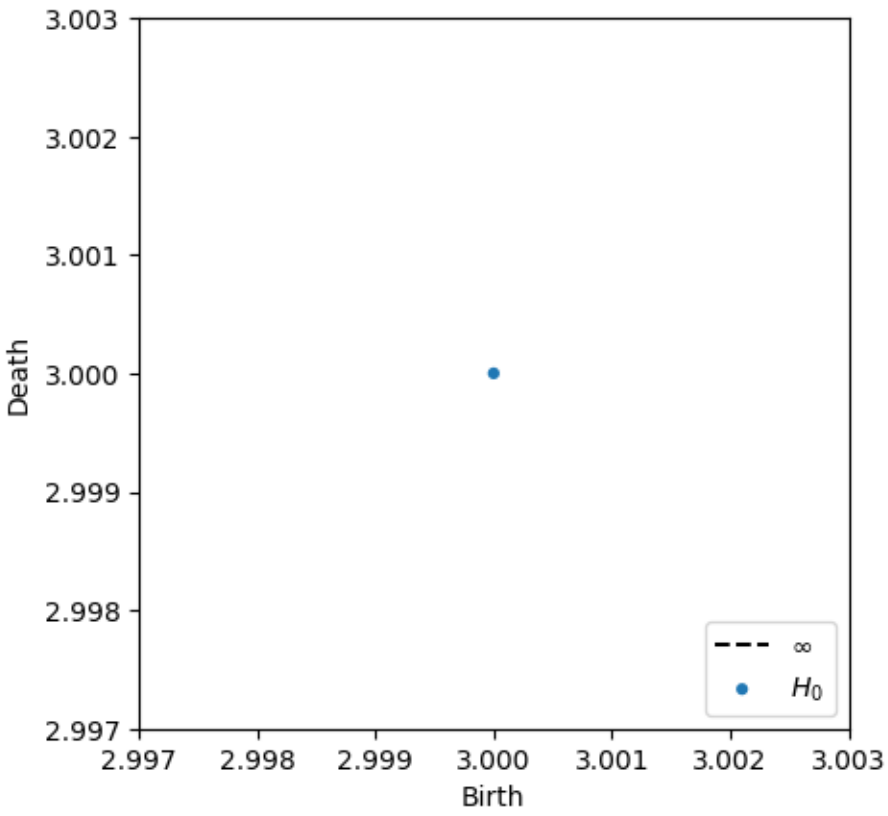
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3. inf]]
```



[83]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

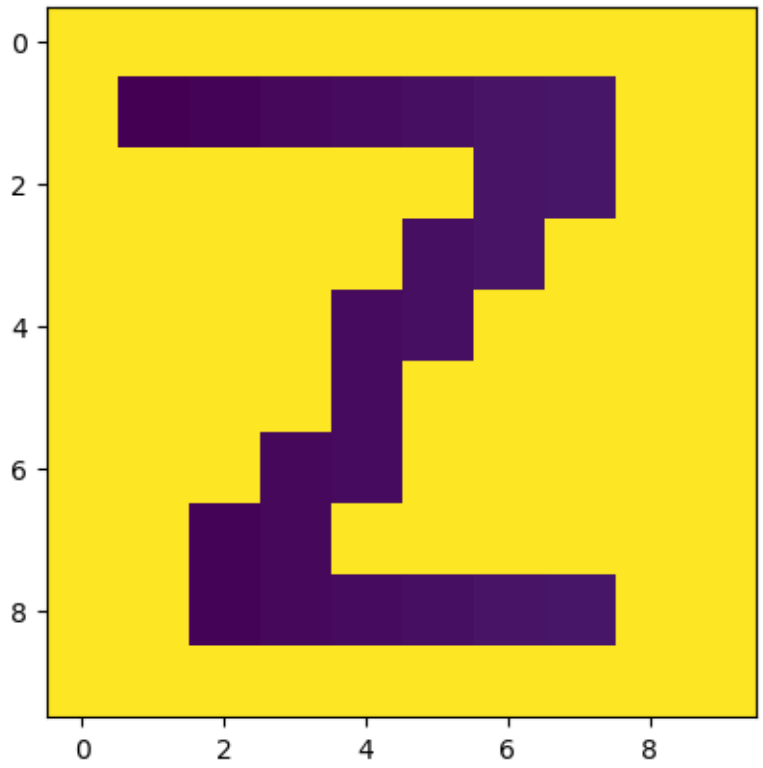
```
(1, 2)
[[ 3. inf]]
```



[84]:
```
from numpy import genfromtxt
import numpy as np
# read in file of letters

# read in file of letters
letters = genfromtxt('letters.csv', delimiter=',') # take first letter
letter_one_line=letters[25,:]

# 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]=k%10
print(letter.shape)
print(letter)

plt.imshow(letter)
```

```
plt.show()
```

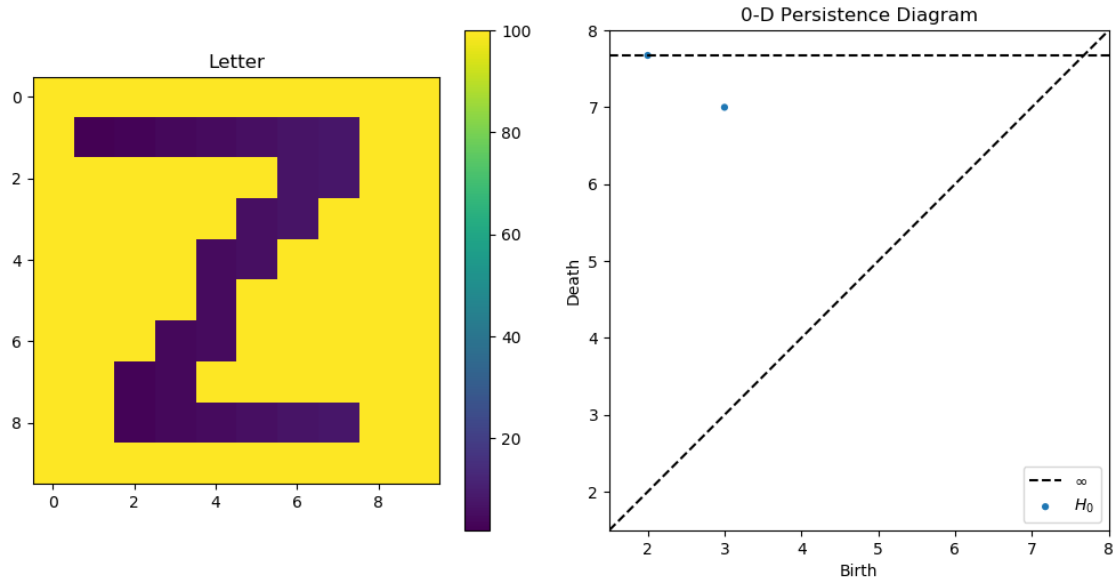
```
(10, 10)
[[100 100 100 100 100 100 100 100 100 100]
 [100   2   3   4   5   6   7   8 100 100]
 [100 100 100 100 100 100   7   8 100 100]
 [100 100 100 100 100   6   7 100 100 100]
 [100 100 100 100   5   6 100 100 100 100]
 [100 100 100 100   5 100 100 100 100 100]
 [100 100 100   4   5 100 100 100 100 100]
 [100 100   3   4 100 100 100 100 100 100]
 [100 100   3   4   5   6   7   8 100 100]
 [100 100 100 100 100 100 100 100 100 100]]
```



[85]:
```
dgm = lower_star_img(letter)
print(dgm)
plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(letter)
plt.colorbar()
plt.title("Letter")
plt.subplot(122)
```

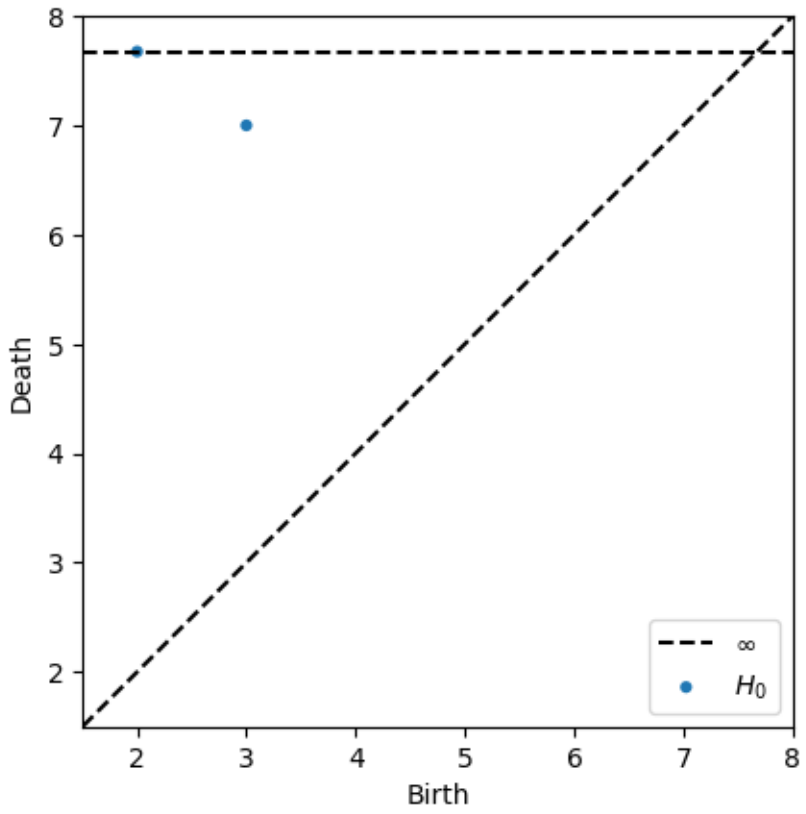
```
plot_diagrams(dgm)
plt.title("0-D Persistence Diagram")
plt.tight_layout()
plt.show()
```

```
[[ 3.  7.]
 [ 2. inf]]
```



[86]:
```
dgm = lower_star_img(letter)
print(dgm.shape)
print(dgm)
plot_diagrams(dgm)

plt.show()
```

```
(2, 2)
[[ 3.  7.]
 [ 2. inf]]
```


8
7
6
5
4
3
2
Death
2
3
4
5
6
7
8
Birth
∞
H_0

[ ]: