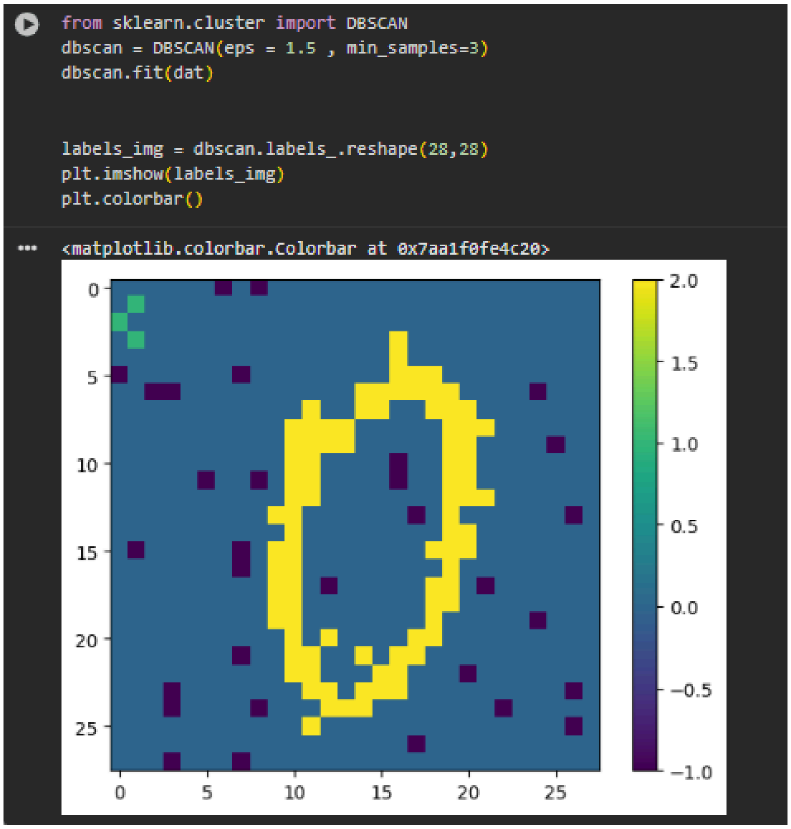
Mikolaj Kubowicz
1A)

```
(784, 3)
array([[ 0.,  0., -1.],
       [ 1.,  0., -1.],
       [ 2.,  0., -1.],
       [ 3.,  0., -1.],
       [ 4.,  0., -1.],
       [ 5.,  0., -1.],
       [ 6.,  0.,  1.],
       [ 7.,  0., -1.],
       [ 8.,  0.,  1.],
       [ 9.,  0., -1.]])
```

We mostly see –1 values because the majority of the image is dark background, black pixels. The few +1 values appear along the white region of the digit “0”, located roughly in the central area of the image, forming the circular outline of the number.

1b)

The pixels captured as noisy points by the algorithm are the scattered white pixels in the background of the image, labeled as –1. Yes, there are some noisy pixels that have been misidentified as small separate clusters. This happens because the chosen hyperparameters (eps = 2.0, min_samples = 3) make DBSCAN too lenient , the large radius and low minimum sample size cause nearby noise pixels to be grouped together as clusters. To fix this, we can decrease eps (e.g., to 1.0) and increase min_samples (e.g., to 4 or 5) so that DBSCAN becomes stricter and correctly labels all noisy pixels as noise.



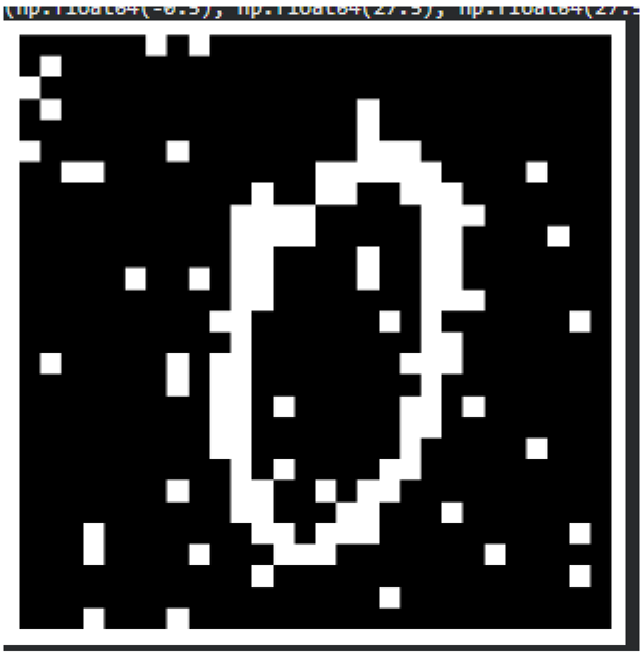
```
from sklearn.cluster import DBSCAN
dbscan = DBSCAN(eps = 1.5 , min_samples=3)
dbscan.fit(dat)


labels_img = dbscan.labels_.reshape(28,28)
plt.imshow(labels_img)
plt.colorbar()
```

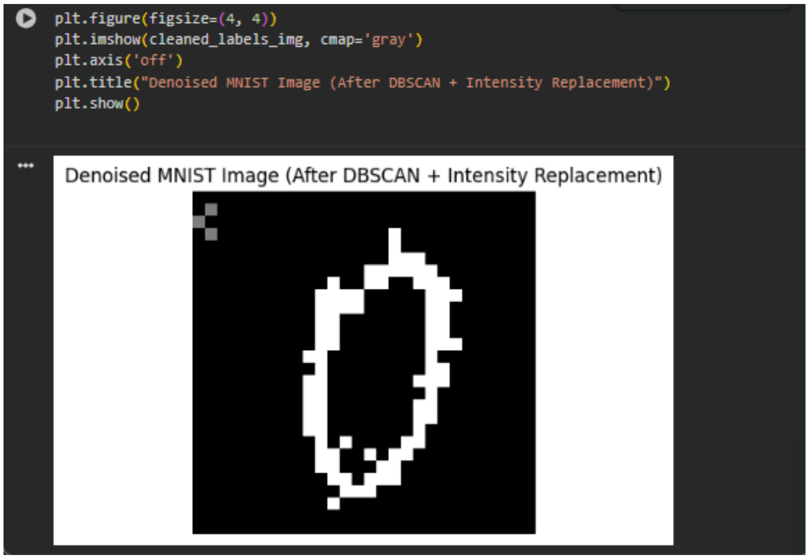
```
<matplotlib.colorbar.Colorbar at 0x7aa1f0fe4c20>
```

1c)

I think its way better however, the result is not completely clean , a few small gray pixels remain in the upper-left corner.This happens because, some noisy pixels did not have enough clean neighbors within the chosen 5×5 neighborhood to determine the correct majority intensity. DBSCAN may have slightly misclassified a few isolated pixels as clean instead of noise.
The fixed patch width may not fully capture the local structure of the digit in all regions.



```
(np.float64(-0.5), np.float64(27.5), np.float64(27.
```



```
plt.figure(figsize=(4, 4))
plt.imshow(cleaned_labels_img, cmap='gray')
plt.axis('off')
plt.title("Denoised MNIST Image (After DBSCAN + Intensity Replacement)")
plt.show()
```

2a)

$\boxed{A \perp S}$ Nothing observed, the only path A→C←S is a **collider** at CCC (blocked).

$\boxed{A \not\perp S \mid C}$ Condition on CCC only, conditioning on a collider **opens** the path (“explaining away”)

$\boxed{A \not\perp S \mid H}$Condition on HHH only. HHH is a **descendant of the collider CCC**; conditioning on a collider’s descendant also opens the path, inducing dependence.

2b)

```
Nodes: ['Age', 'Chol', 'Sex', 'HD', 'RestBP', 'FastingBS']
Edges: [('Age', 'Chol'), ('Chol', 'HD'), ('Sex', 'Chol'), ('RestBP',
'HD'), ('FastingBS', 'HD')]

First 5 independencies:
 (Sex ⊥ RestBP)
(HD ⊥ Age | Chol)
(FastingBS ⊥ Chol)
(FastingBS ⊥ Sex)
(FastingBS ⊥ Age)
```

: $(\text{FastingBS} \perp \text{Chol})$

The only path connecting **FastingBS** and **Chol** is $\text{Chol} \rightarrow \text{HD} \leftarrow \text{FastingBS}.$

Here, HD is a collider (two arrows pointing inward).In Bayesian Networks, a collider structure (A → C ← B) blocks information flow between A and B unless you condition on the collider (HD) or one of its descendants.Because we are not conditioning on HD, the path is blocked, making FastingBS and Chol independent.

2c)

The proportion of labels in `train_df['Sex']` is:

- fraction of 0’s ≈ **0.315854**
- fraction of 1’s ≈ **0.684146**

2d)

- A=35, S=0, Chol=172 → P(HD=0)=0.5232, P(HD=1)=0.4768
- A=35, S=1, Chol=172 → P(HD=0)=0.5232, P(HD=1)=0.4768
- A=60, S=1, Chol=172 → P(HD=0)=0.5232, P(HD=1)=0.4768

Explanation:
The probability of heart disease (HD) remains the same across all combinations of Age and Sex when cholesterol (Chol) is fixed.

This occurs because, in the Bayesian Network, Age and Sex only influence Heart Disease **through** Cholesterol via the paths:

- Age → Chol → HD
- Sex → Chol → HD

When we condition on Chol, these paths are **blocked**, meaning Age and Sex provide no additional information about HD once Chol is known.The model gives the same prediction for HD regardless of Age and Sex once Chol is included as evidence.

2f)

```
BN accuracy (manual): 0.9854
BN accuracy (sklearn): 0.9854
```