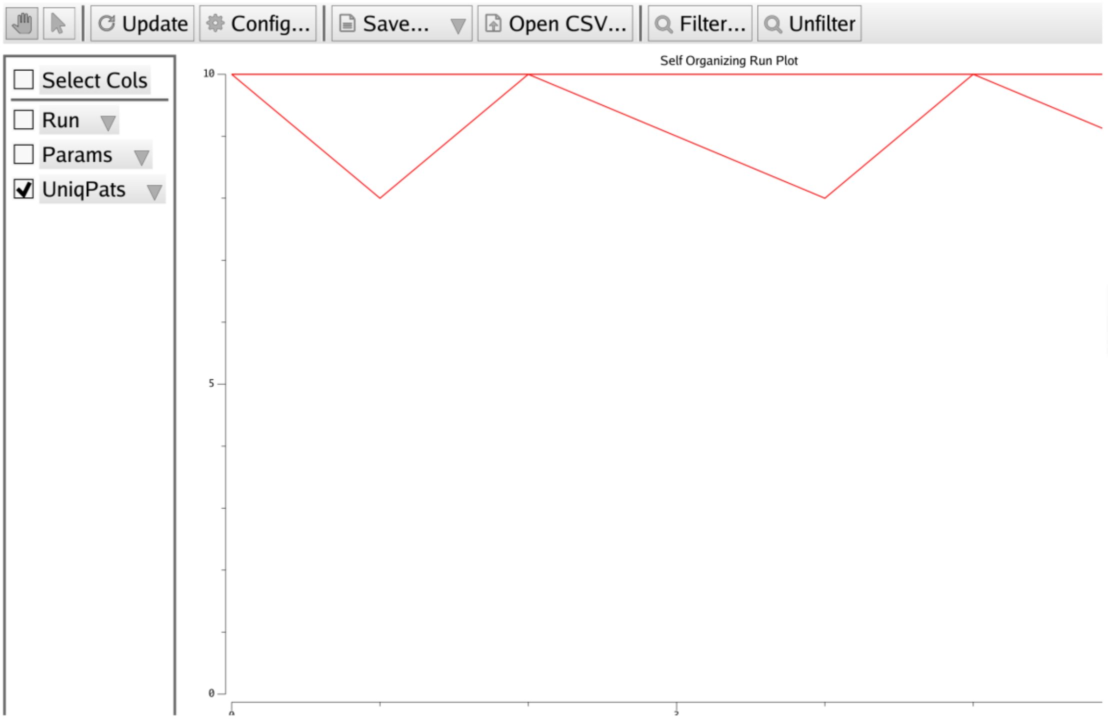
2.75 2.75 4.5

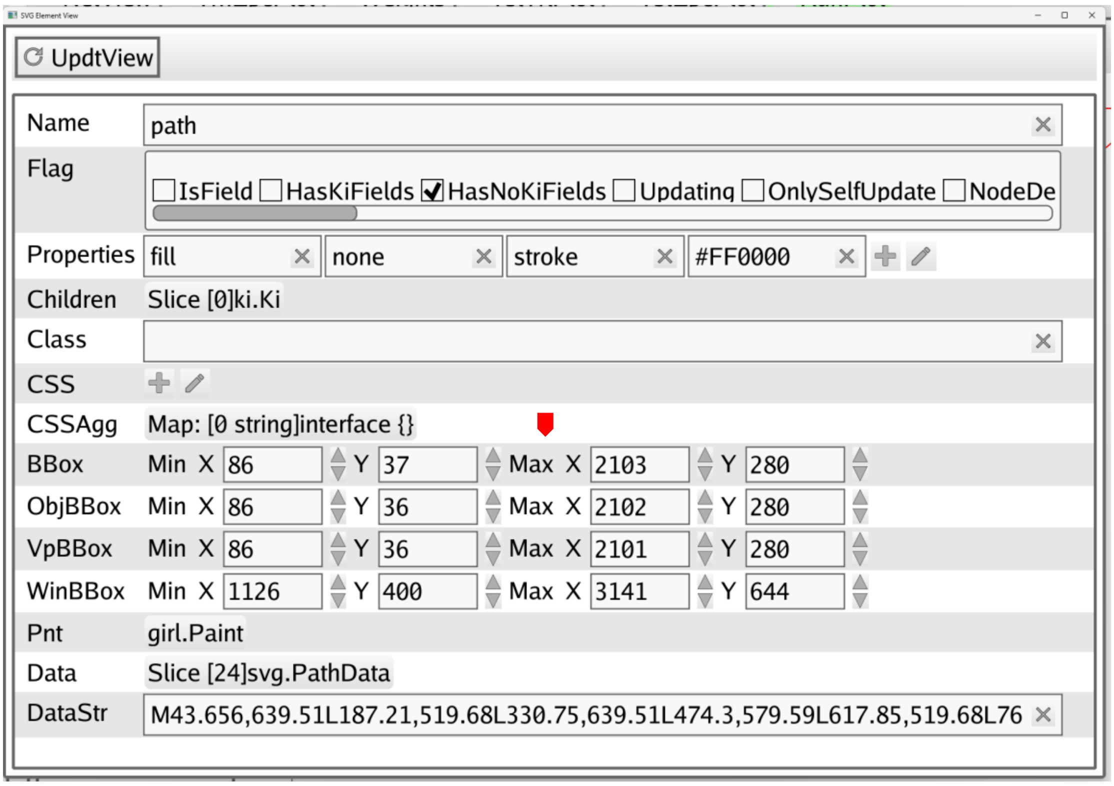
Computational Cognitive Neuroscience – CH4

Zhuo Wang ScM BME Brown ID# 140641091

Question 4.1

0.5





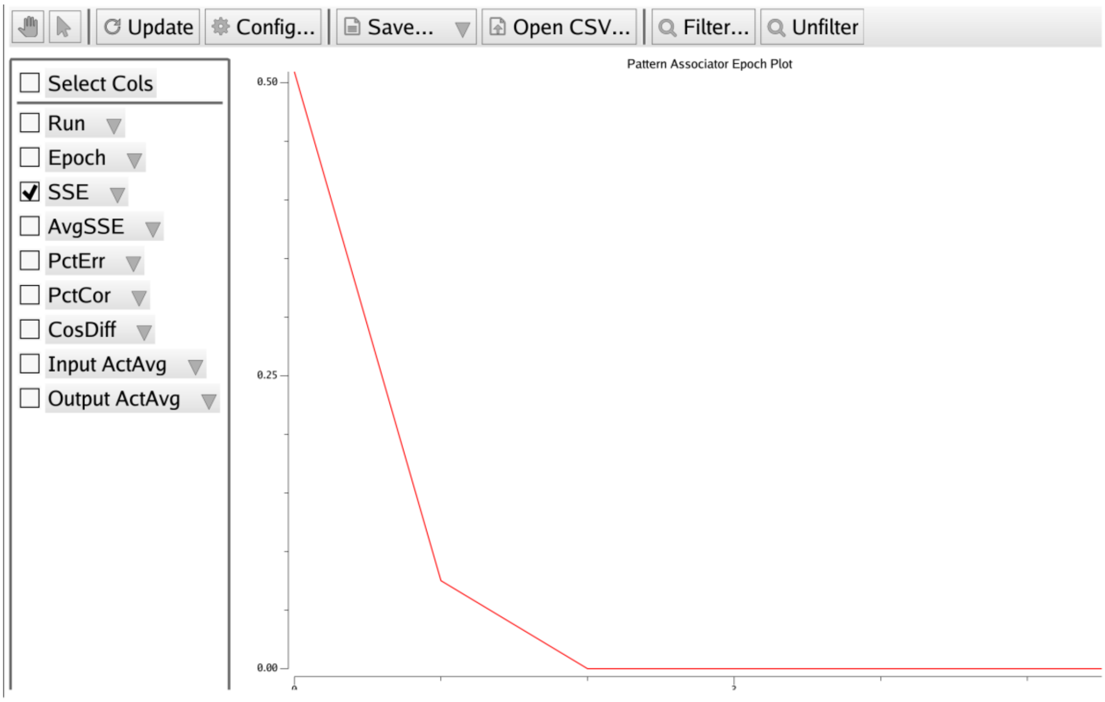
Min and max.

Question 4.2

The number of lines that are uniquely represented as the network learns is measured by the UniqPats statistic in the RunStats. Depending on the value of AvgLGain, adding input noise has a different effects. Adding input noise (InputNoise set to 0.2) does not significantly change the UniqPats statistic when AvgLGain is at its default value of 2.5, and the effect is not immediately noticeable. However, when input noise is present, the
0.25 UniqPats statistic may increase if AvgLGain is decreased to 1.5 or 1. Lower AvgLGain values give the network more latitude to create different patterns, which helps improve the learning of separate representations.

More erratic activation patterns in the input layer may result from the addition of input noise. The learning process may be affected by this variation. The network may have trouble telling apart comparable inputs in the no-noise simulation, when input is deterministic and devoid of unpredictability, and it may not build as many separate representations. The network experiences input diversity in the noise simulation, which can aid in producing more varied activation patterns. This variety promotes the development of distinctive representations for various input patterns, improving UniqPats statistics.

Question 4.3



1 Strong weights from the leftmost input units and weaker weights from the rightmost input units are present in the left output unit. Strong weights from the rightmost input units and weaker weights from the leftmost input units are present in the right output unit.

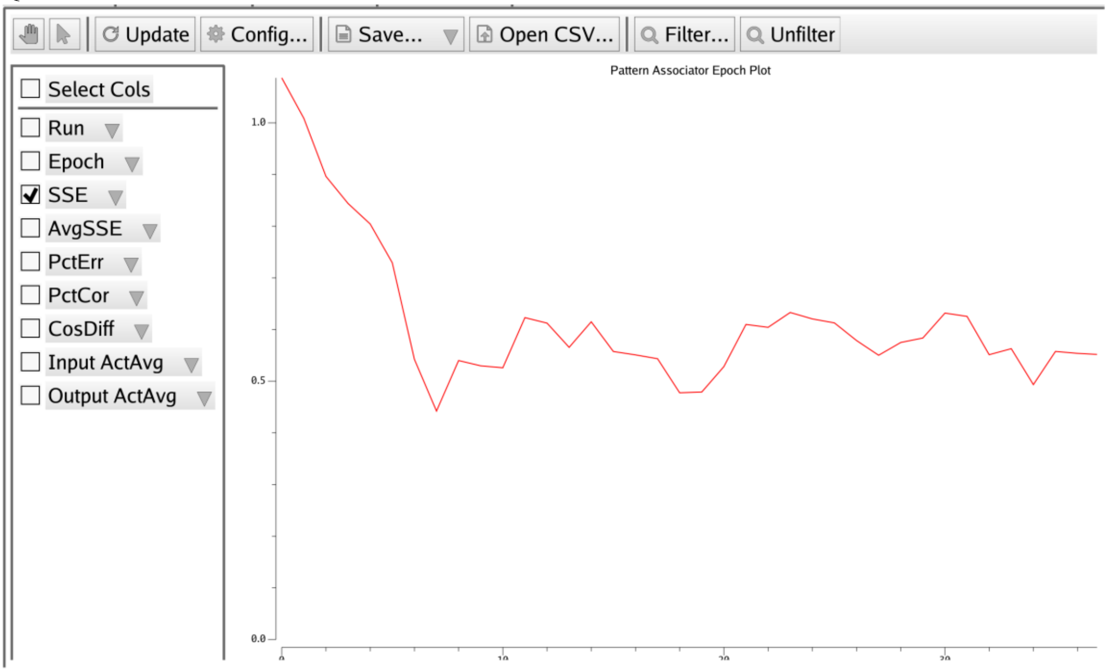
Question 4.4

The observed pattern of weights would result from a learning mechanism in the Hebbian sense since it strengthens the links between input units and output units that are active together during training. When the leftmost input units are active, the left output unit is clamped with a value of 1, and when the rightmost input units are active, it is clamped with a value of 0. Because they frequently co-activate during training and

1 produce strong weights, the Hebbian learning mechanism strengthens the connections between the leftmost input units and the left output unit. When the rightmost input units are active, the right output unit is clamped with a value of 1, and when the leftmost

good! input units are active, it is clamped with a value of 0. Through Hebbian learning, the connections between the rightmost input units and the right output unit are reinforced because of their continuous co-activation throughout training, which produces strong weights.

Question 4.5

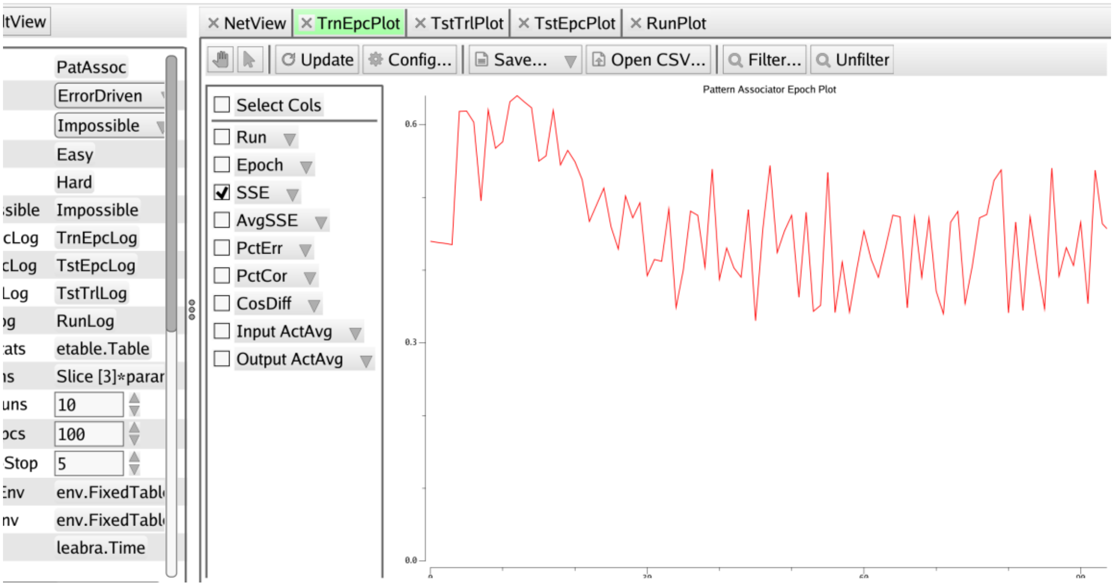


0.75

The task does not appear to be entirely resolved by hebbian learning. The SSE value is around 0.5. This result indicates that the input patterns are not correctly categorized by the network in accordance with the task criteria.

Question 4.6

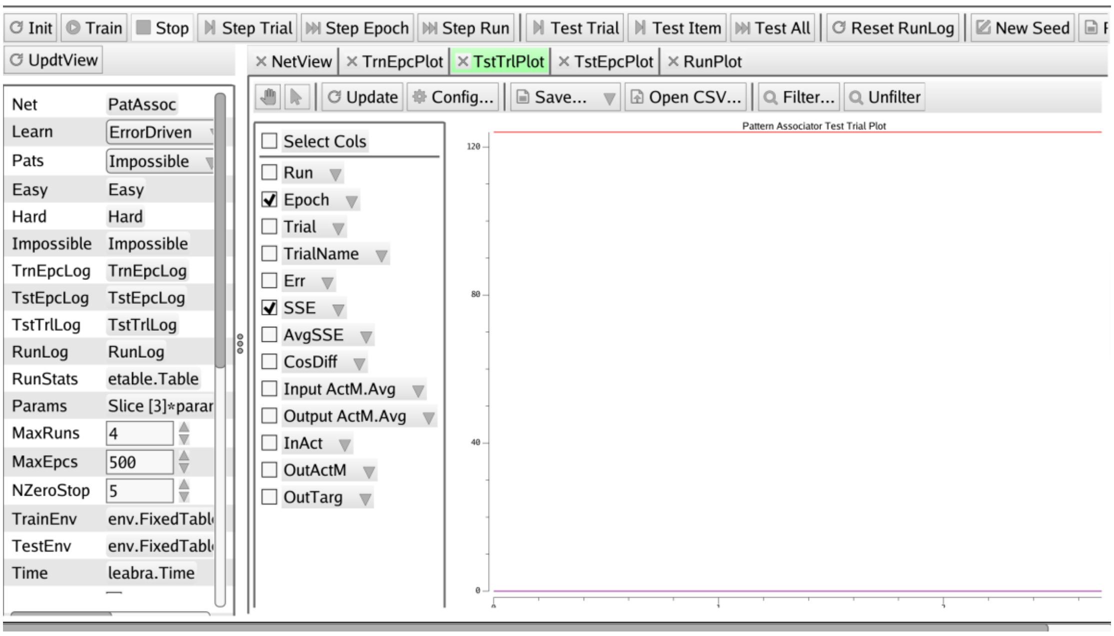
0.5



The "Impossible" problem does not to be learned by the network utilizing error-driven learning. The network appears to have difficulty classifying the input patterns in accordance with the task criteria, according to the final SSE values. This highlights the intricacy of the issue and the difficulties involved in employing error-driven learning to solve ambiguous cue situations.

Question 4.7a

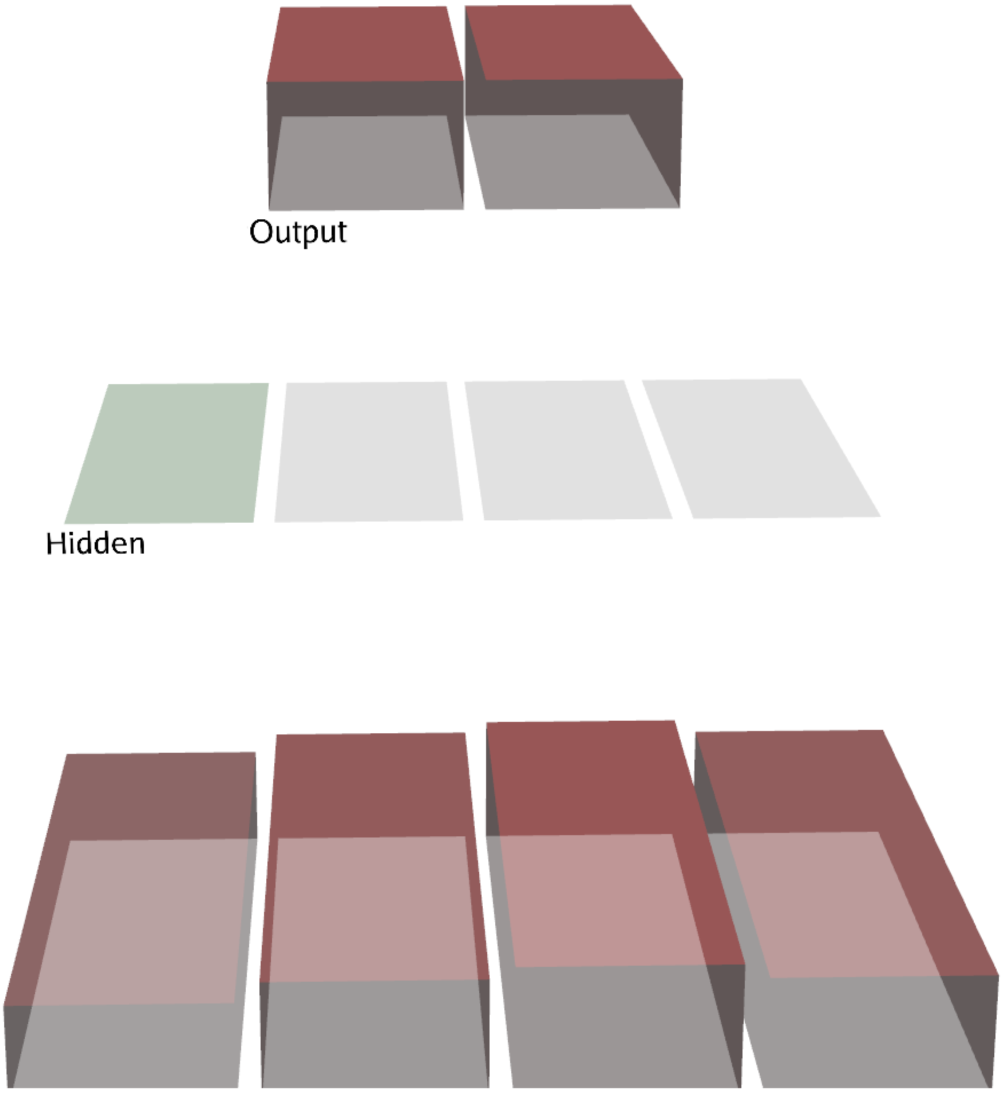
0.5

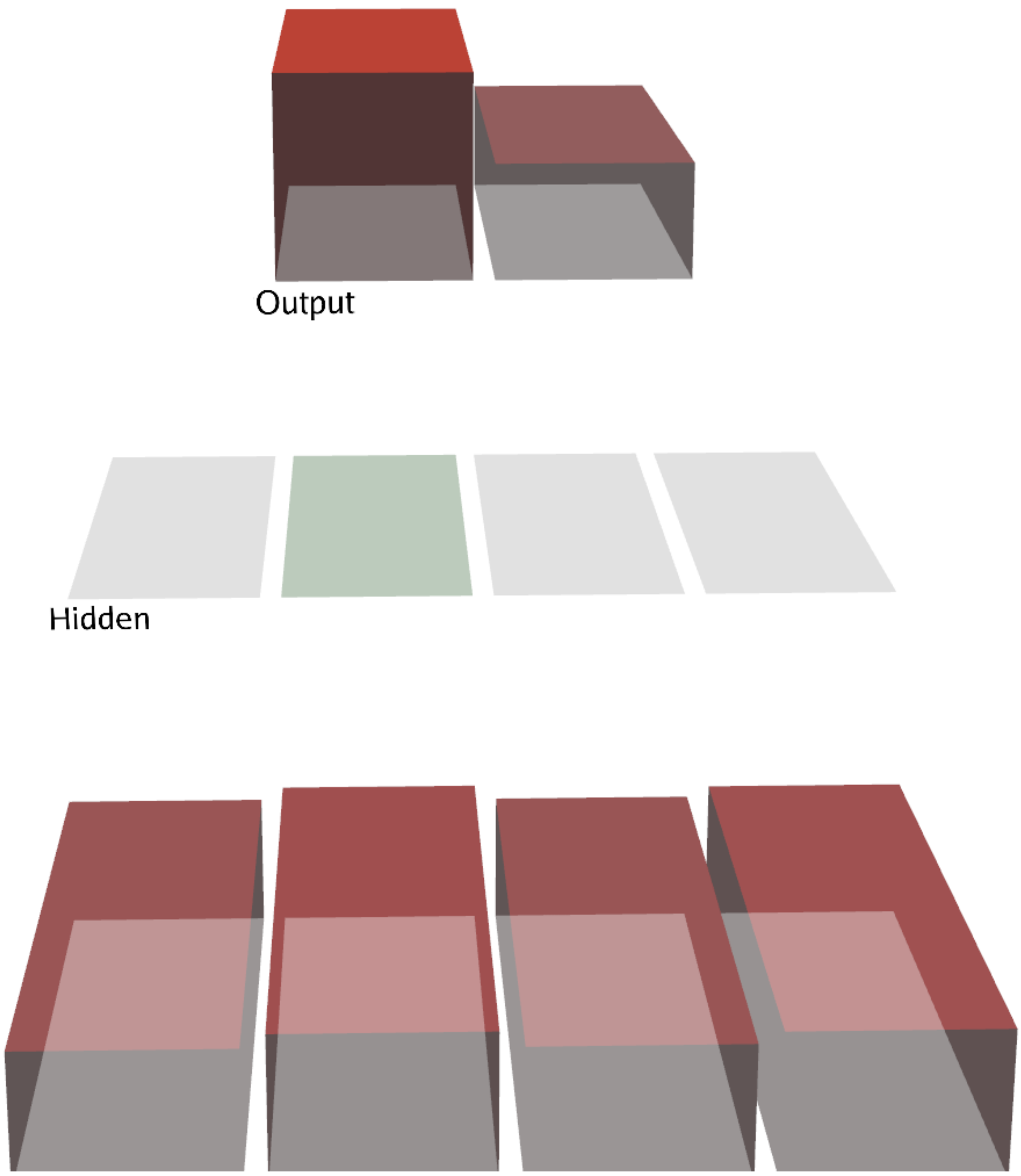


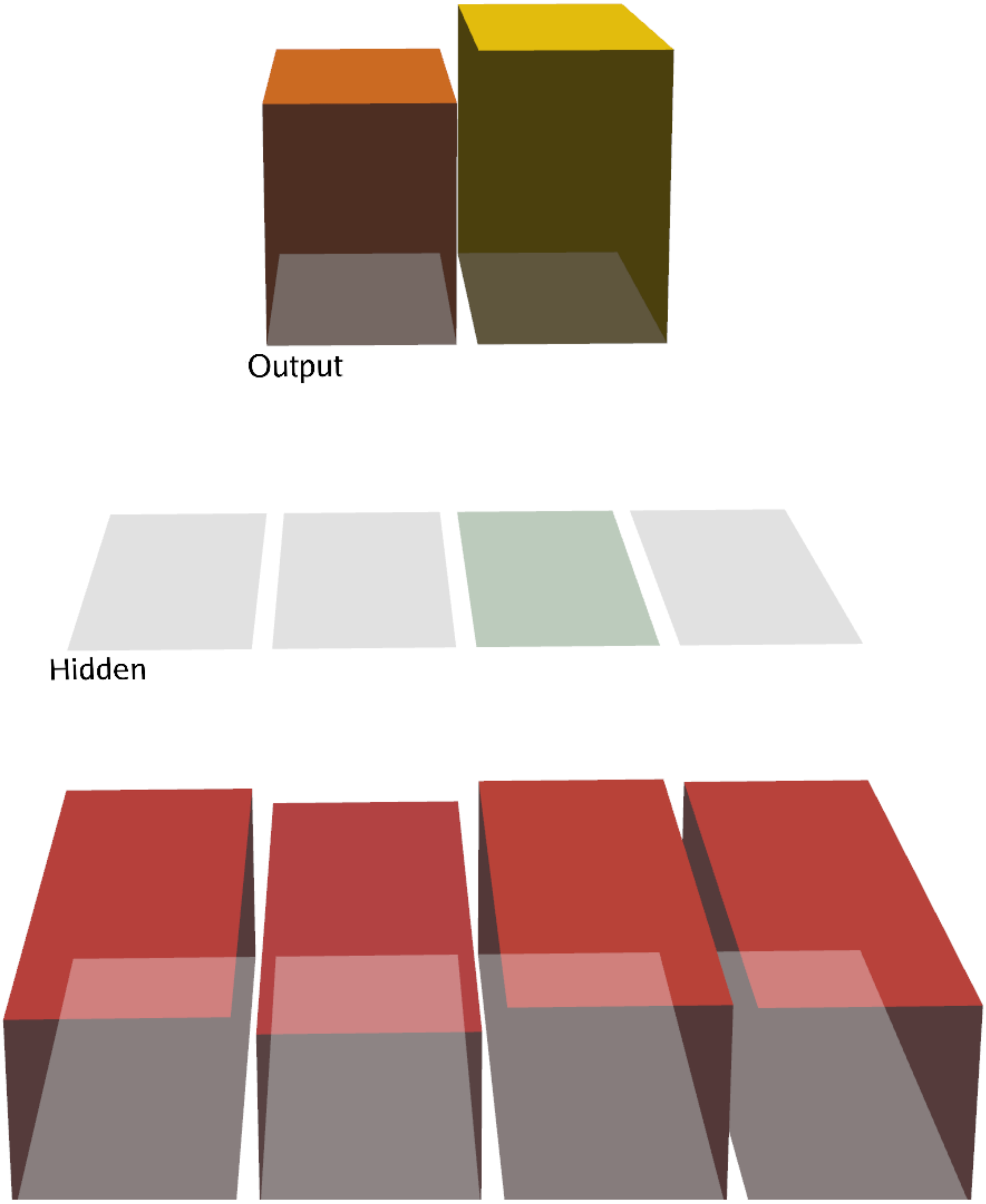
Around 120.

Question 4.7b

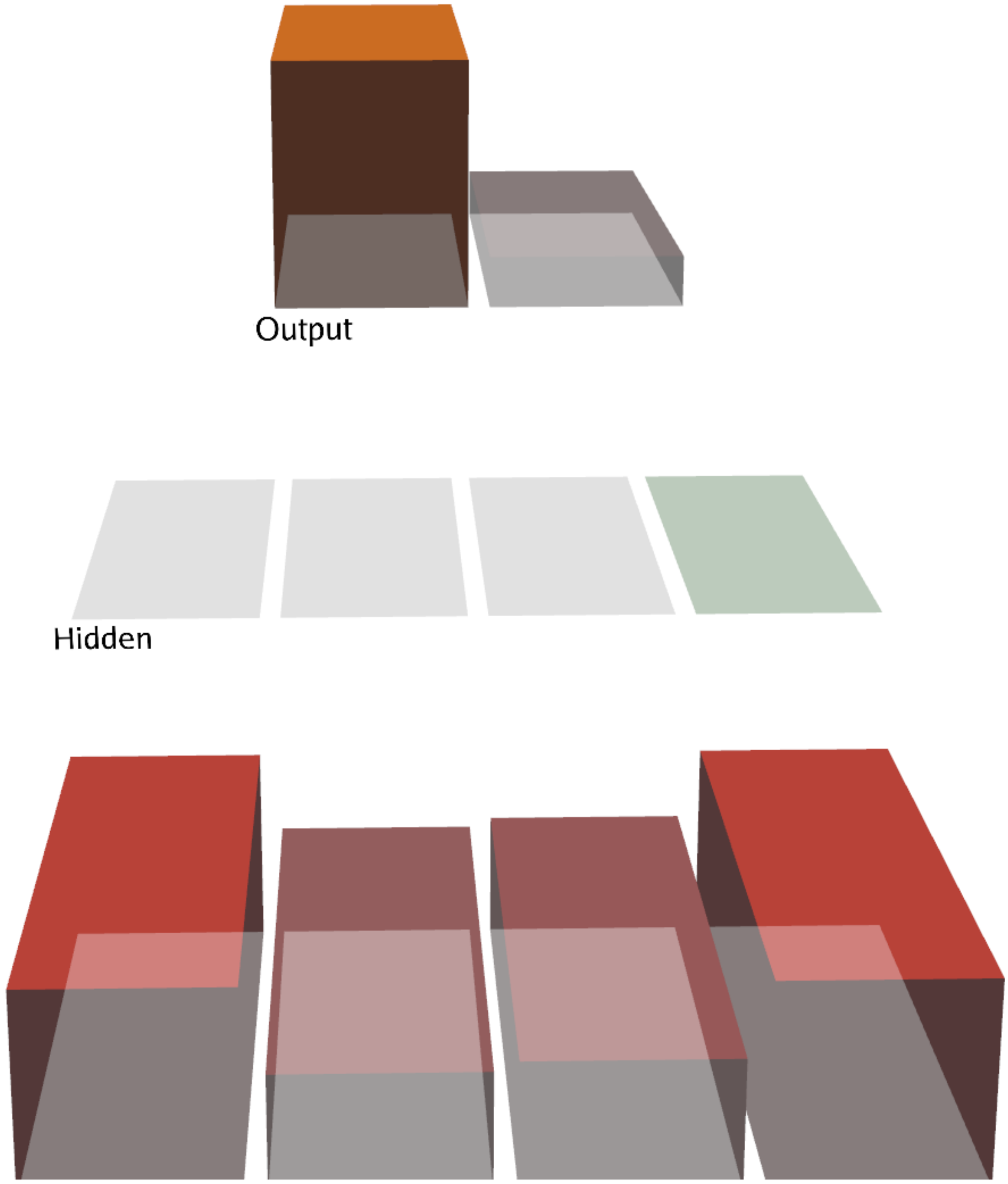
The hidden units of a network with a hidden layer are vital in altering or classifying the input patterns to make the task achievable. The hidden layer successfully simplifies the process by grouping two of the non-overlapping input patterns into the same representation, enabling the network to drive the proper output unit for each input pattern. The features of the input data are abstracted into another dimensional space to show its more abstract features.




Output
Hidden



Output
Hidden



The four units of the hidden layer appear to be active as well as the four events, but the weights from these units to the output units vary.

Question 4.8

The results pattern suggests that the relationship component is the main component along which the hidden layer is organizing the input patterns. With a reasonably high
1 degree of similarity across inputs with the same relationship, the similarity matrix for the Hidden layer is clearly arranged according to the type of relationship encoded. Because it mirrors the intrinsic structure in the input patterns, this organization helps the model fulfill the task. The relationship component captures how people interact or

are described more similarly when they are in similar functional relationships. The network gains the ability to detect and classify people according to their functional functions as a result of learning to distinguish between various connection categories.

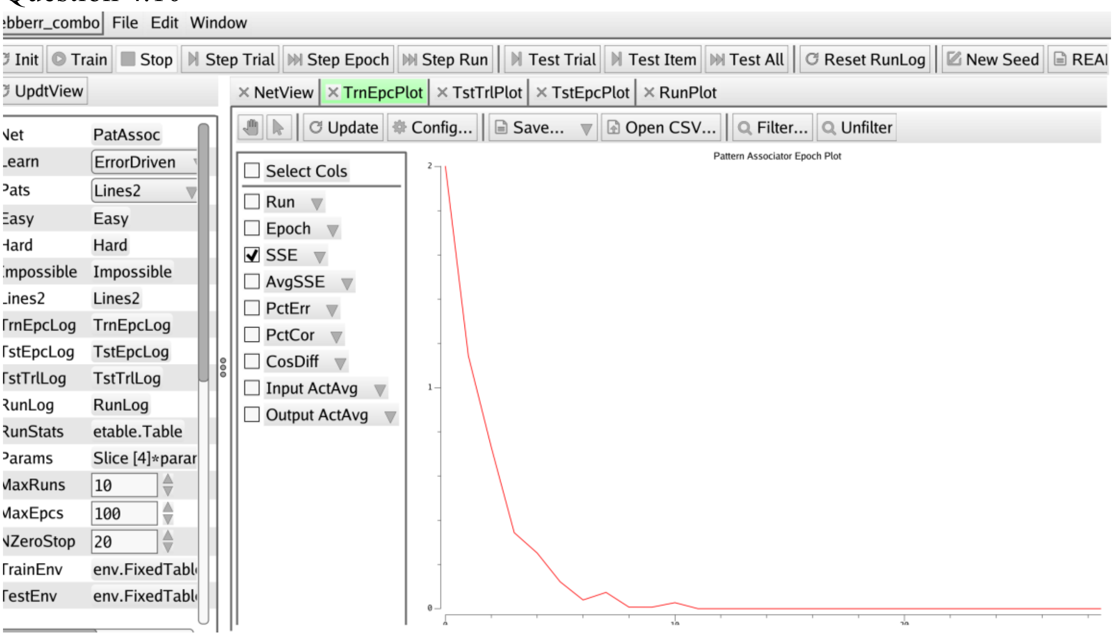
The agent component, in comparison, is probably less central across all the patterns even though it still contributes to the structure. Despite the fact that individuals in various partnerships may share similar roles or traits, the connection itself seems to be a more pronounced and distinctive element of the input patterns. The hidden layer's relationship-based structure corresponds to the task's requirement to classify individuals according to their functions in these relationships.

Question 4.9

The arrangement of the representations in the Hidden layer, particularly with regard to relationships, seems more orderly in the trained network with defined weights. In order
1 to indicate shared characteristics among various connection categories, similar relationship categories are grouped together to form separate groupings.

The untrained representations, on the other hand, do show some structure, but it is not as systematic as in the trained network. The trained network improves this initial structure and creates more systematic representations, even though some structure still exists as a result of the input patterns' intrinsic commonalities. Less limited and lacking in organization compared to the taught representations are the untrained representations.

Question 4.10

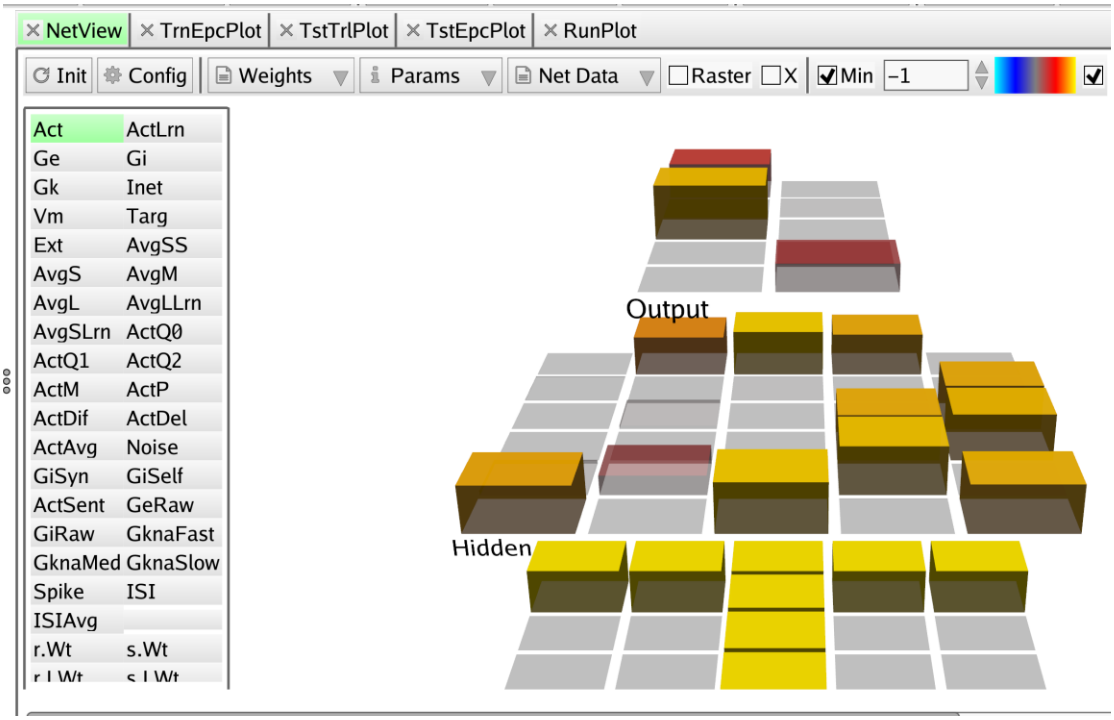


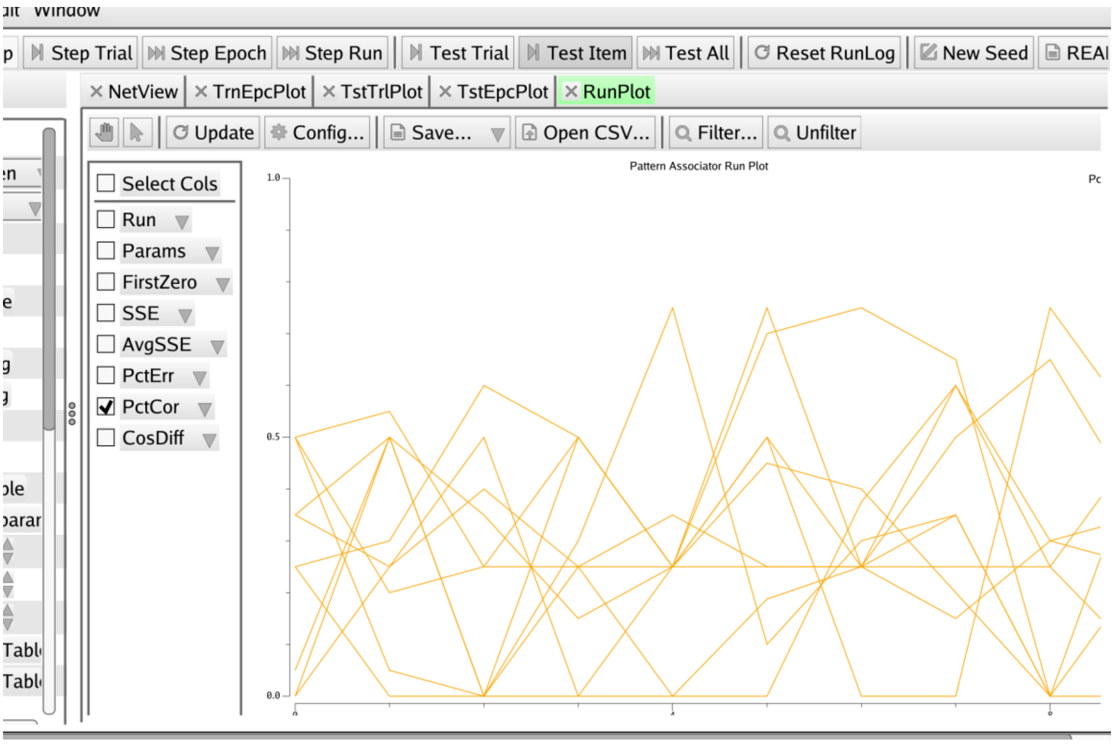
In contrast to the self_org mode, the hidden units in this network do not learn to represent distinct lines. The nature of the learning rule and the job at hand are the main
1 causes of this discrepancy. To create customized representations for each line, the self_org mode sought to encode distinct hidden unit representations for each of the

several lines. The self_org mode's Hebbian-style learning rule encourages the development of distributed representations in which individual hidden units become selective to particular lines. This is so that the network may be trained to recognize the distinctiveness of each line by activating particular hidden units in response to particular lines.

The objective of this study, in contrast, is to learn how to label combinations of lines (vertical and horizontal lines), not to encode distinct representations for individual lines. These pairings must be accurately categorized and labeled by the network. The goal of the error-driven learning rule is to reduce the discrepancy between the network's educated predictions and the accurate labels. As a result, it motivates the network to modify its weights so that it can produce the appropriate labels in response to the shown line combinations.

Question 4.11





The underlying hidden representations that the network has learnt are what allow for this generalization.

0.5 The hidden layer of the network has created representations that take higher-level characteristics or patterns relating to the line combinations into account. These representations encode information about the connections between lines rather than being particular to any one line. As a consequence, using the patterns and correlations it has gleaned from the training data, the network can correctly recognize and classify unique combinations of lines.

Question 4.12

The hidden layer's hebbian learning does improve the network's ability to generalize to new objects. The network may produce rich, abstract representations in the hidden layer, capturing complex relationships and patterns in the input, thanks to the combination of

1 Hebbian learning and error-driven learning. Because they are not bound to particular examples but rather encode more universal properties and connections, these acquired representations aid generalization.

Combining these learning principles in the brain might help people learn specific tasks and develop more broad cognitive abilities. The brain can form perceptual representations, find patterns in the environment, and retain memories via associations with the aid of hebbian learning.