ONLINE MASTERS IN **DATA SCIENCE**

DSC 255 - MACHINE LEARNING FUNDAMENTALS

NEURAL NET EXAMPLES

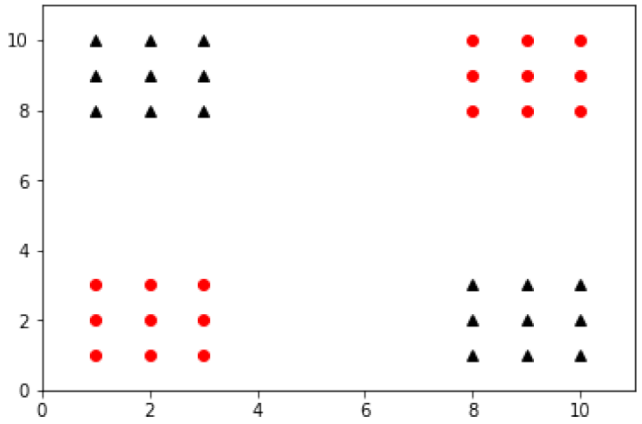
SANJOY DASGUPTA, PROFESSOR

UC San Diego

COMPUTER SCIENCE & ENGINEERING

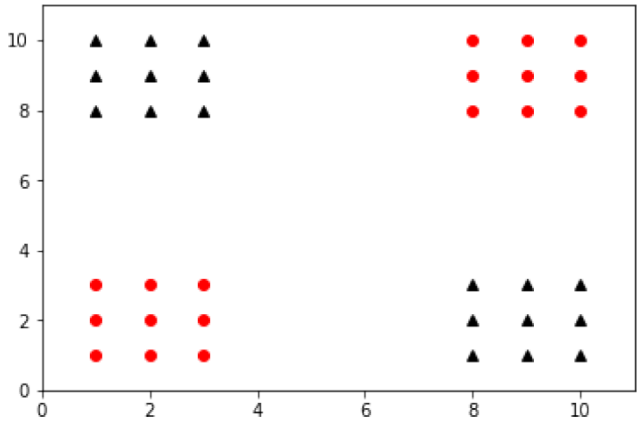
HALICIOĞLU DATA SCIENCE INSTITUTE

Example 1



What kind of net to use for this data?

Example 1

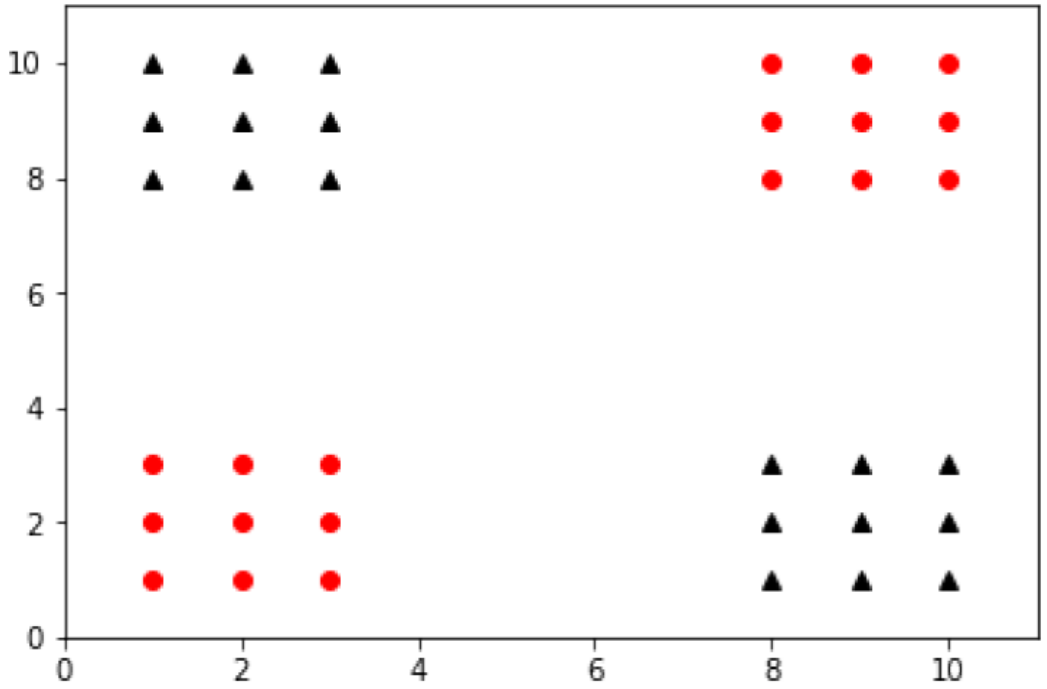


What kind of net to use for this data?

- Input layer: 2 nodes
- One hidden layer: H nodes
- Output layer: 1 node
- Input $\longrightarrow$ hidden: linear functions, $ReLU$ activation
- Hidden $\longrightarrow$ output: linear function, sigmoid activation

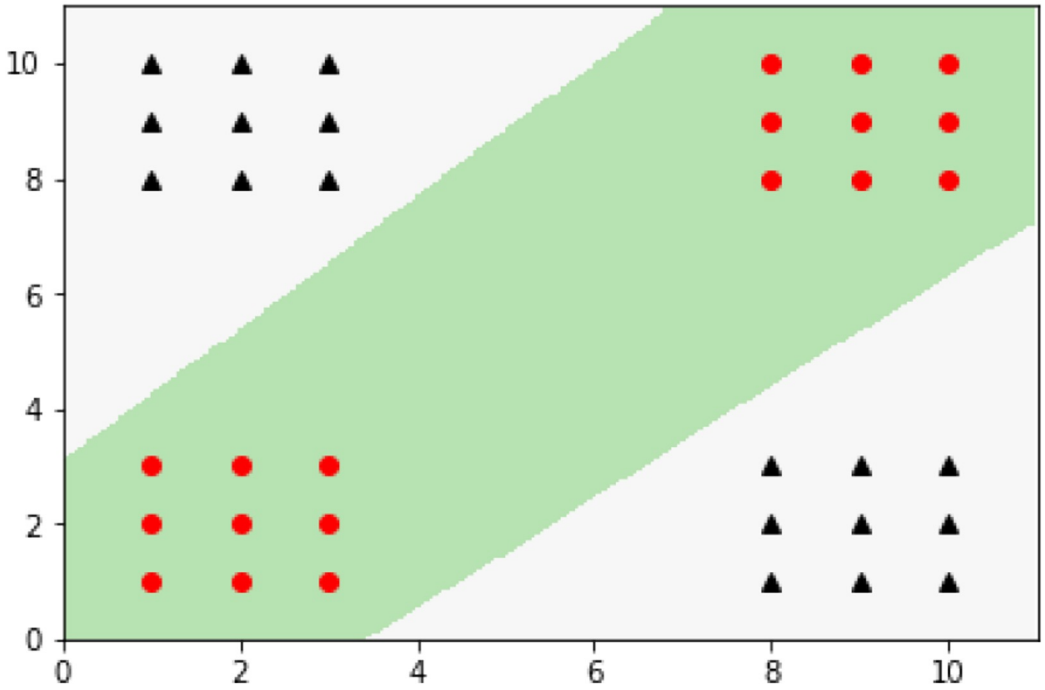
Example 1

How many hidden units should we use?



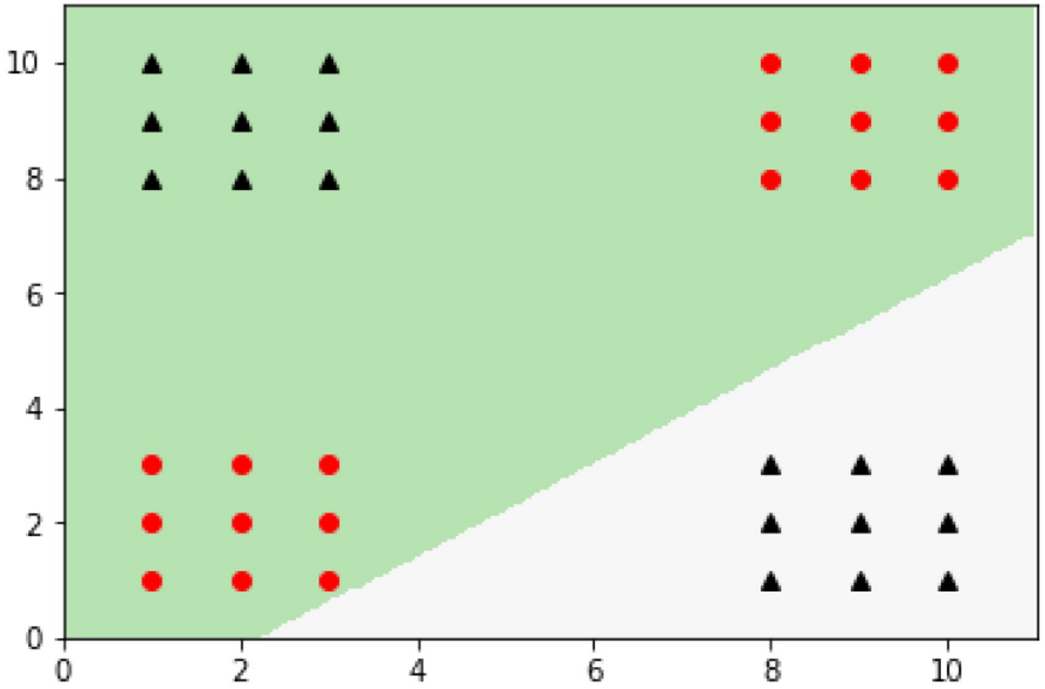
Example 1

$H = 2$



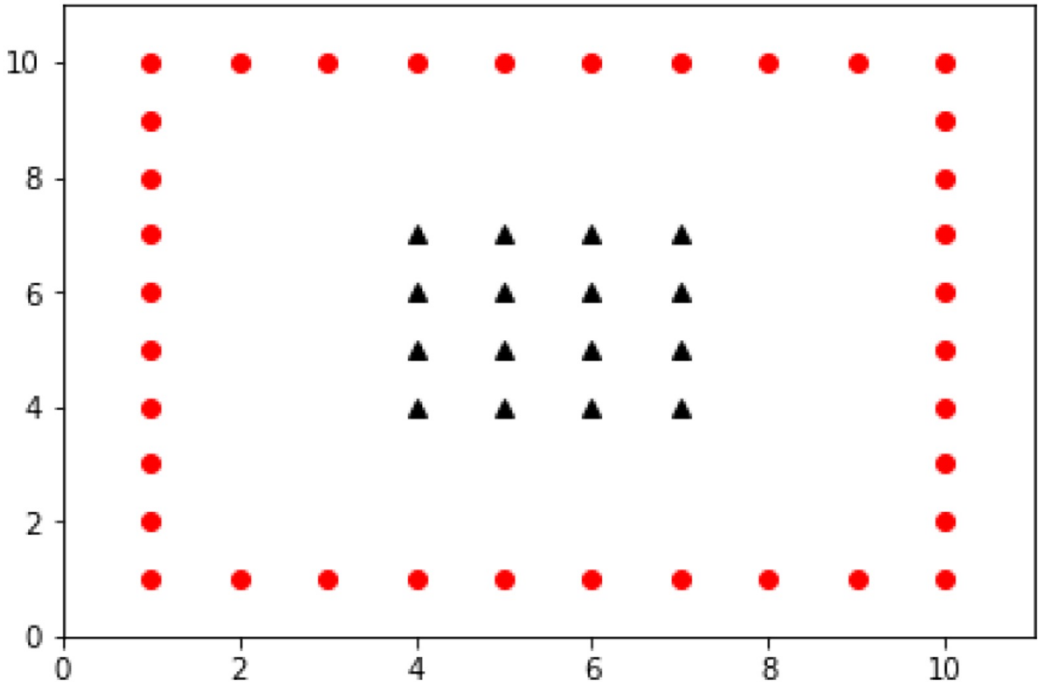
Example 1

$H = 2$



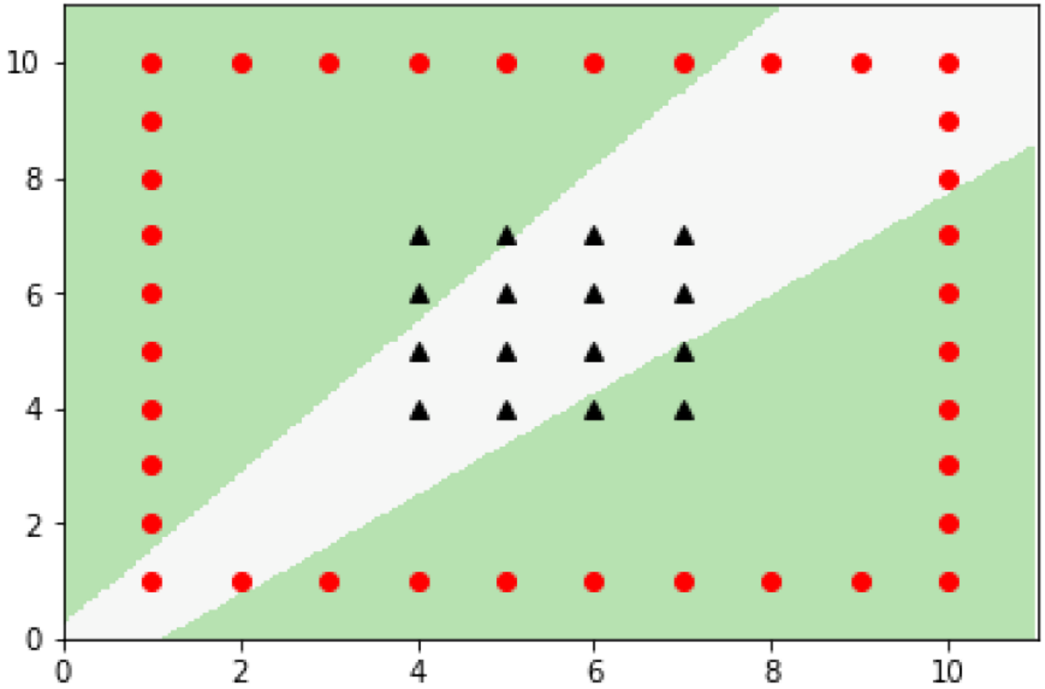
Example 2

How many hidden units should we use?



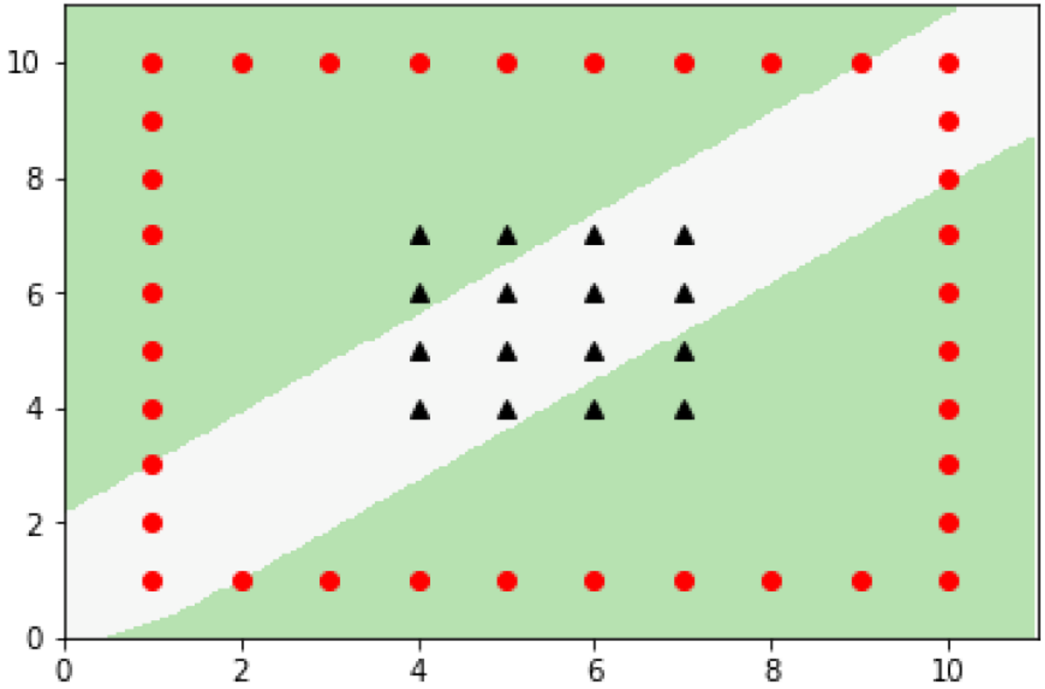
Example 2

$H = 4$



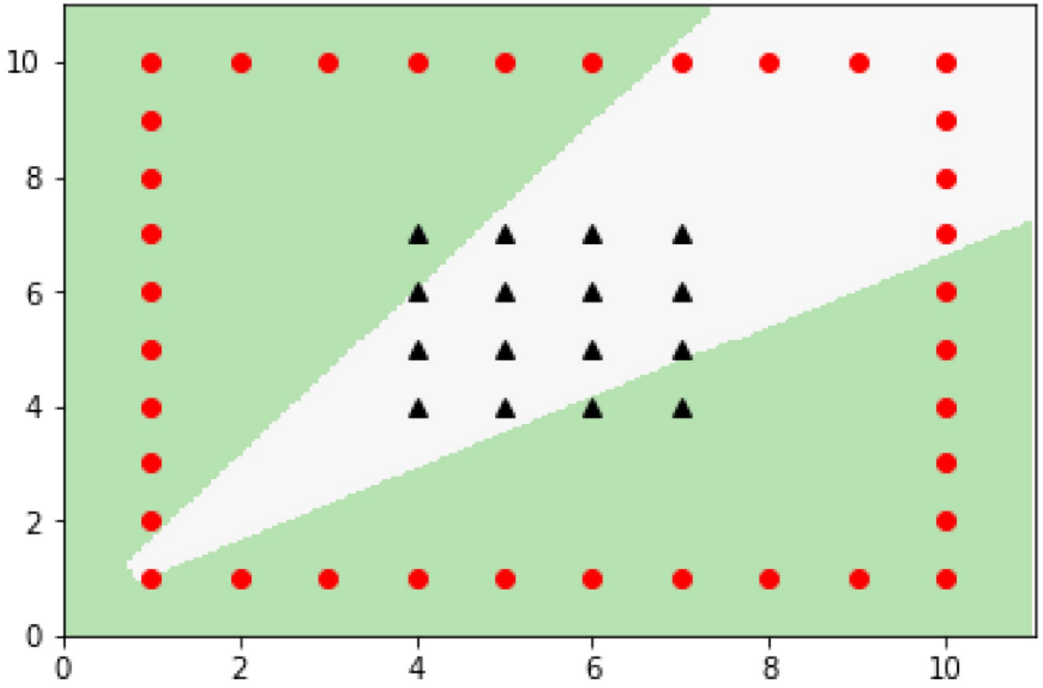
Example 2

$H = 4$



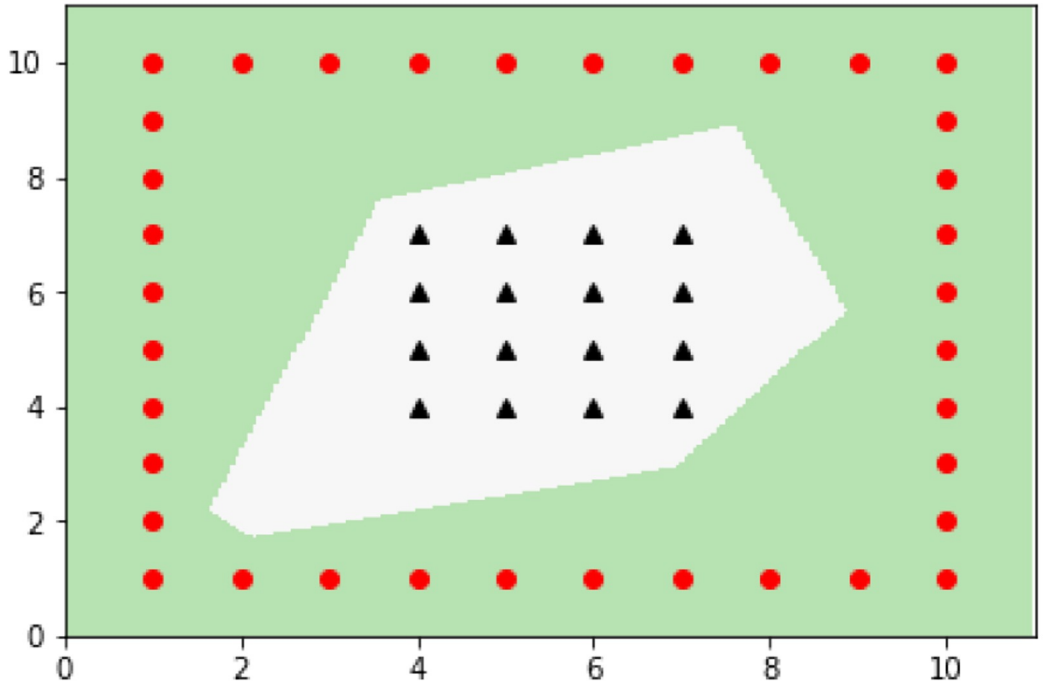
Example 2

$H = 4$



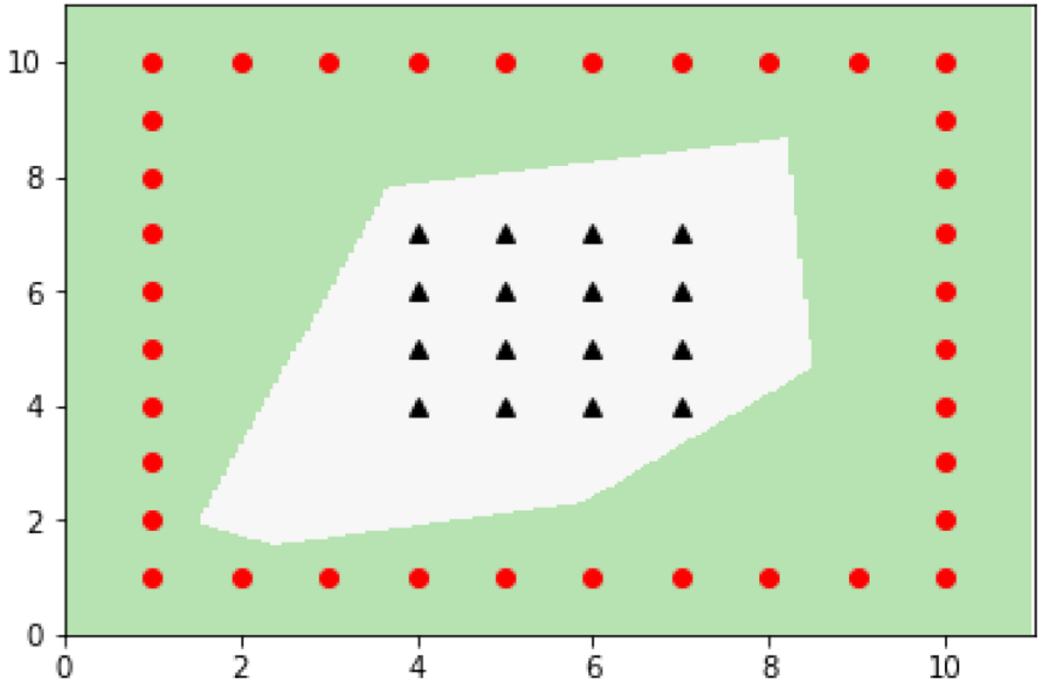
Example 2

$H = 4$



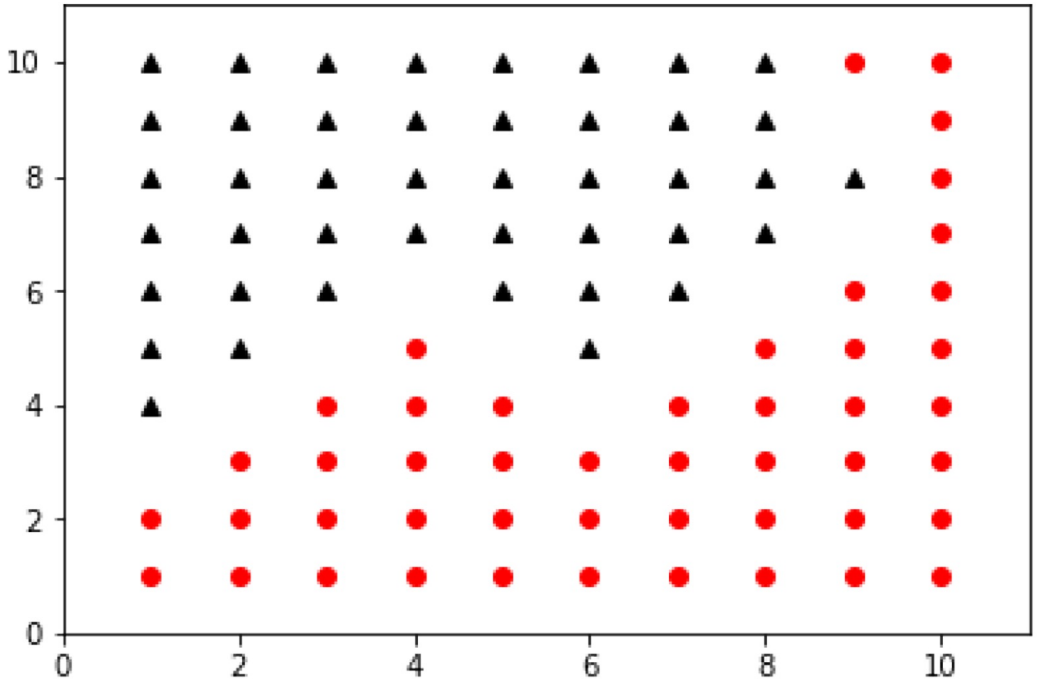
Example 2

$H = 8$: overparametrized



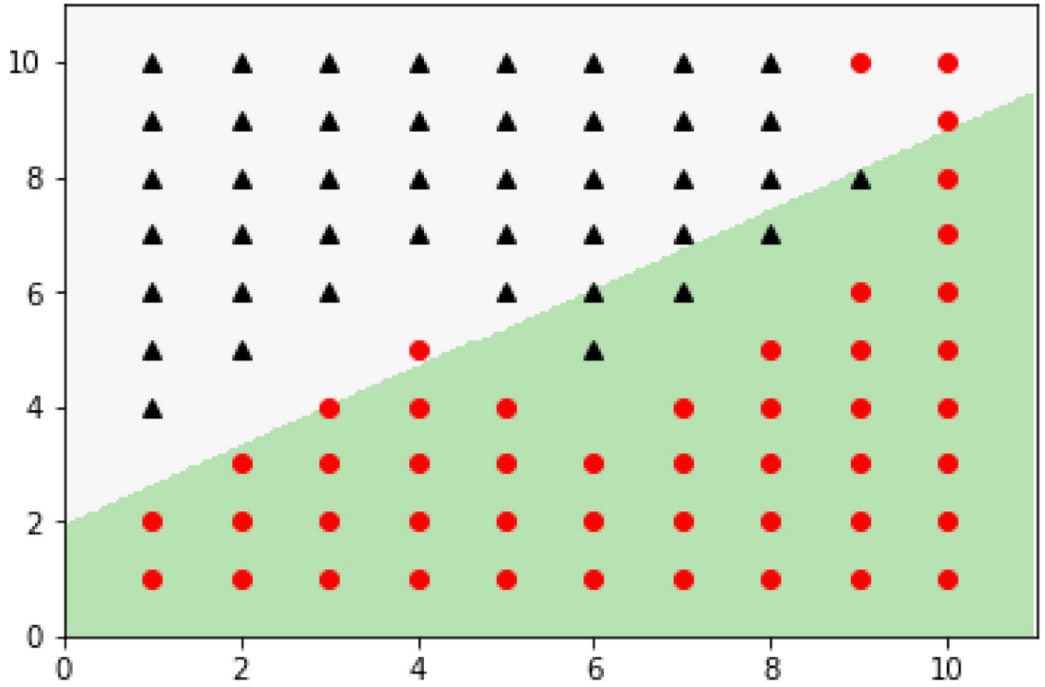
Example 3

How many hidden units should we use?



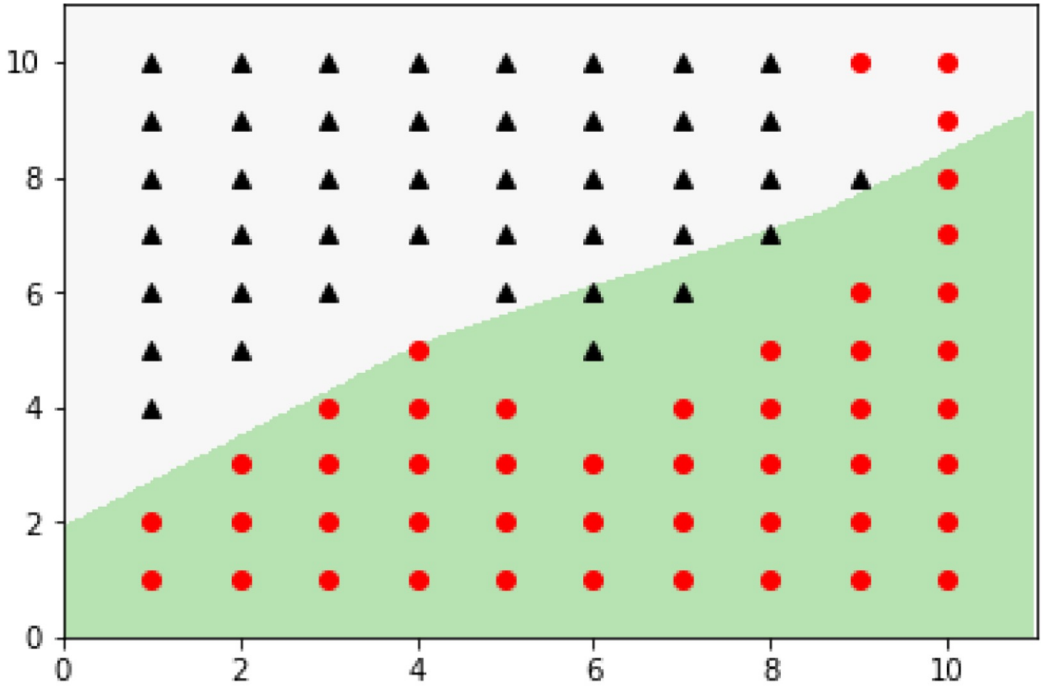
Example 3

$H = 4$



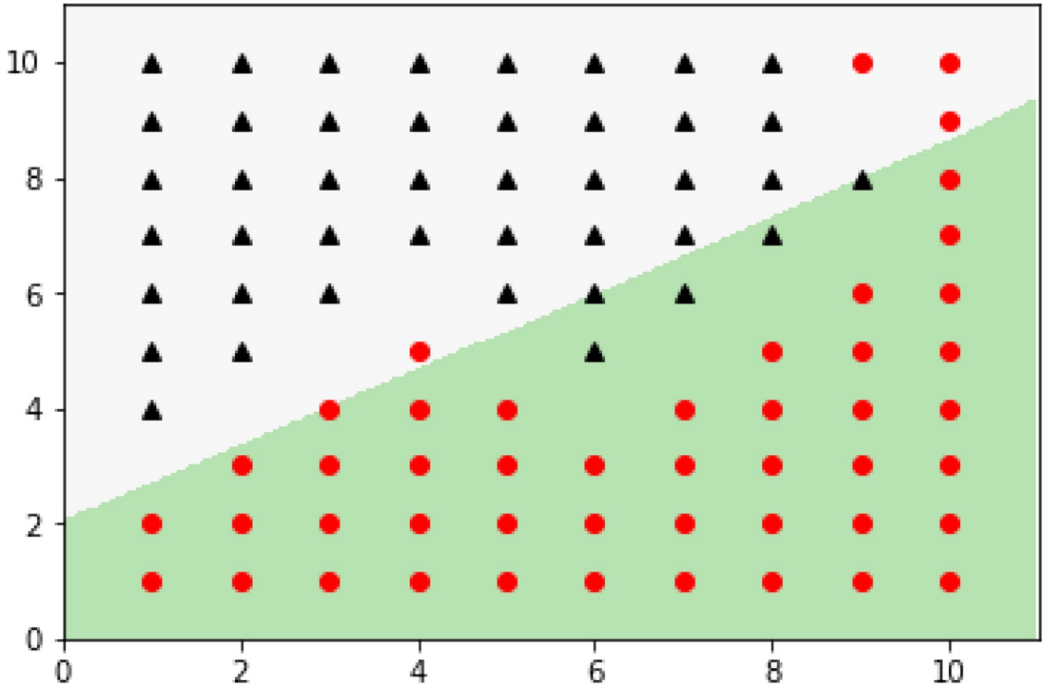
Example 3

$H = 8$



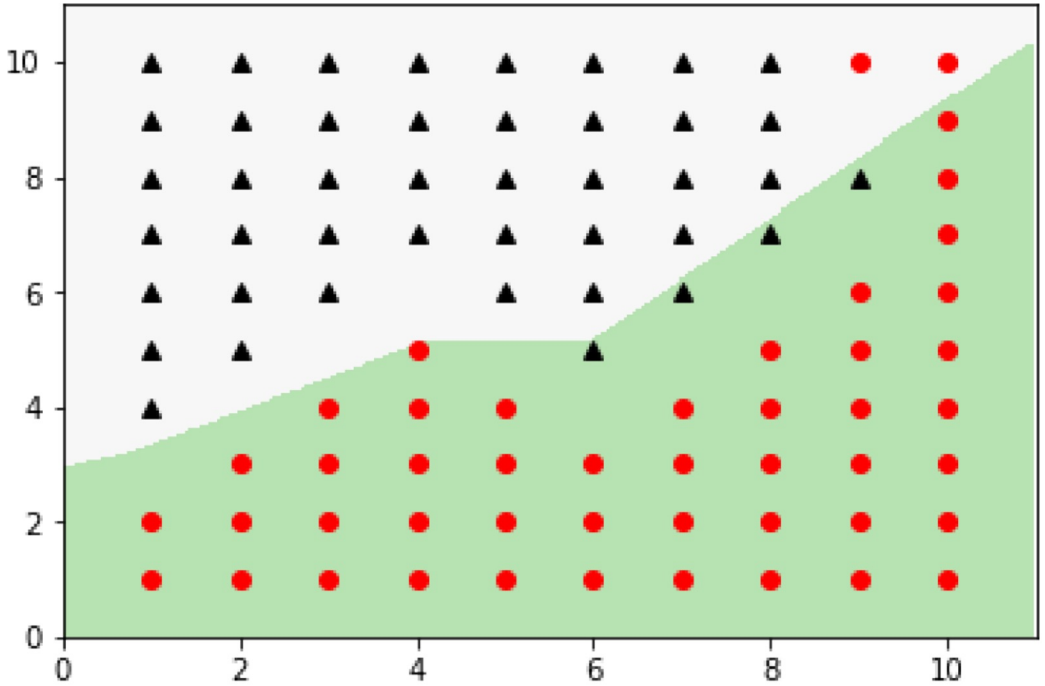
Example 3

$H = 16$



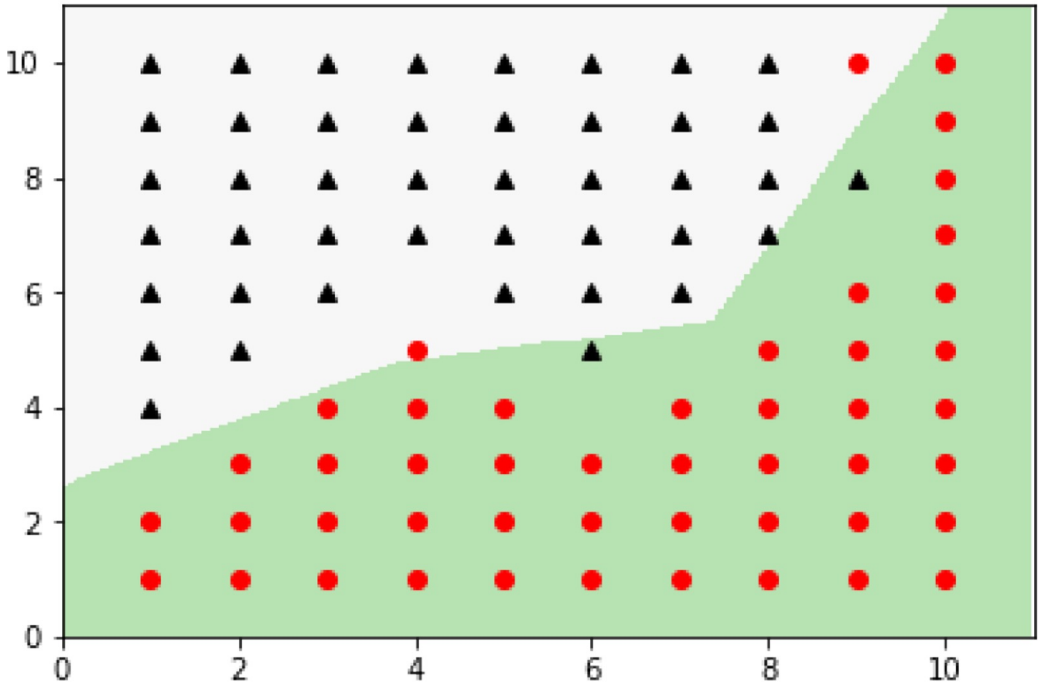
Example 3

$H = 16$



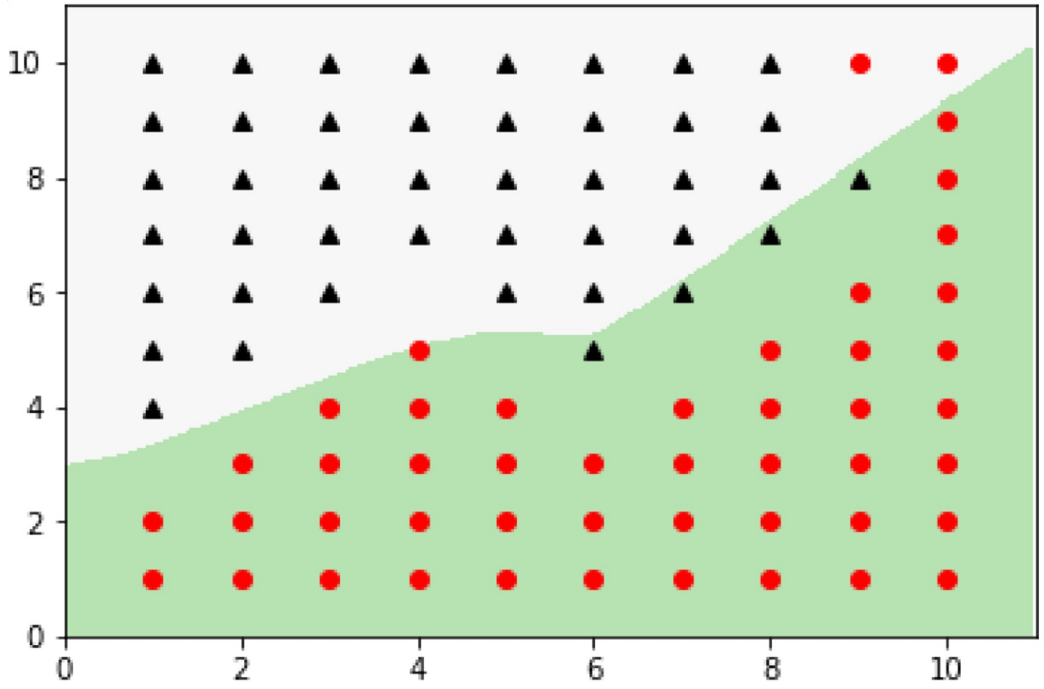
Example 3

$H = 16$



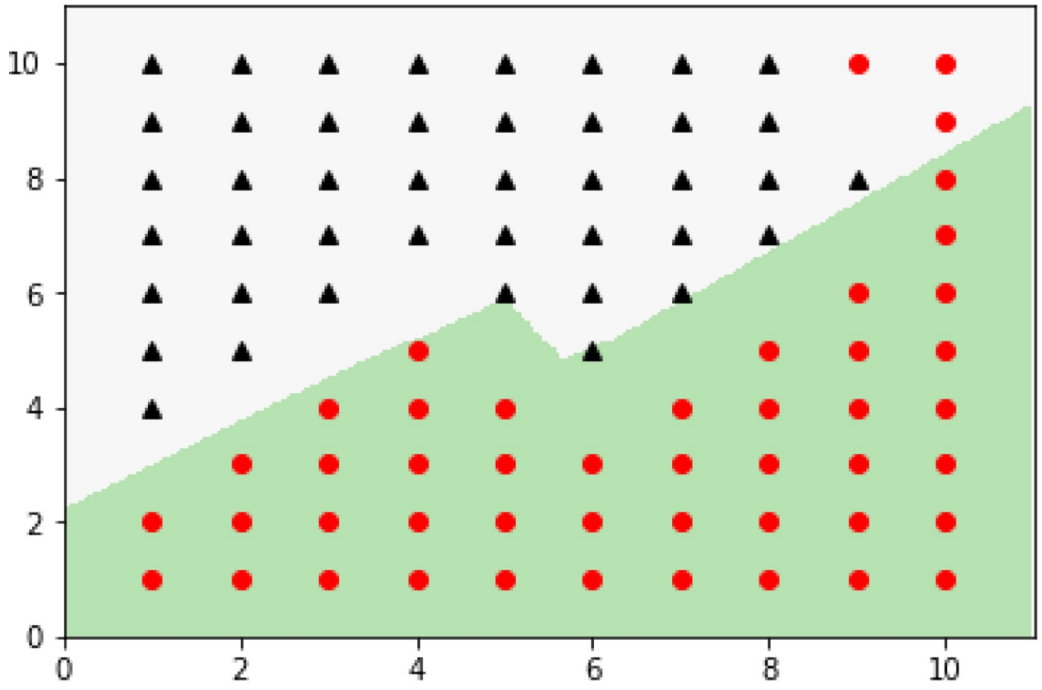
Example 3

$H = 32$



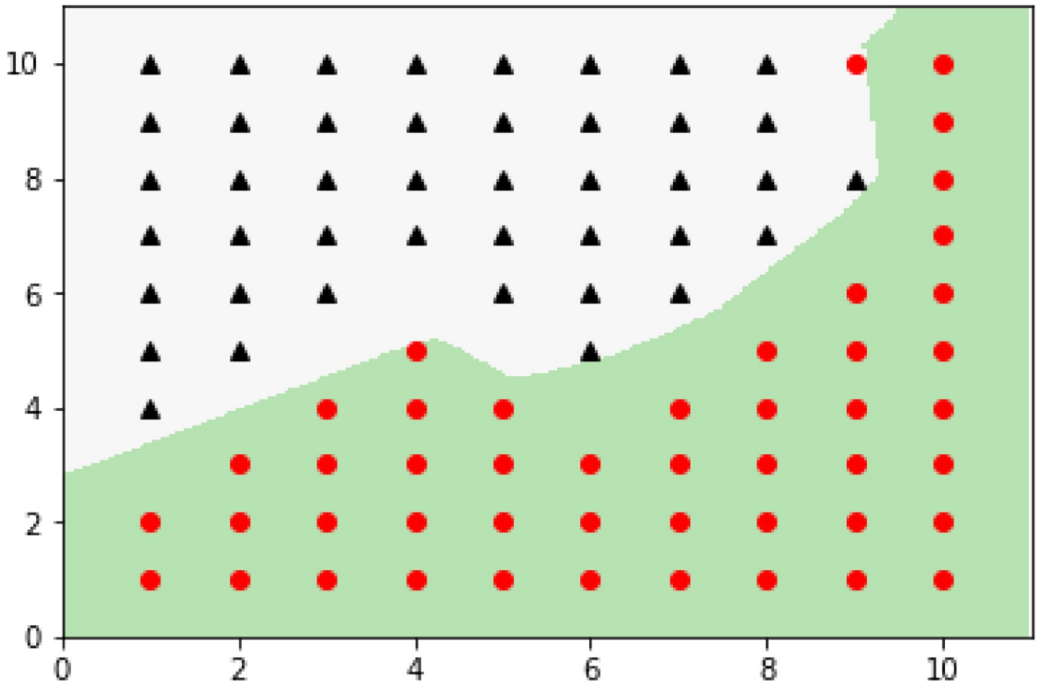
Example 3

$H = 32$



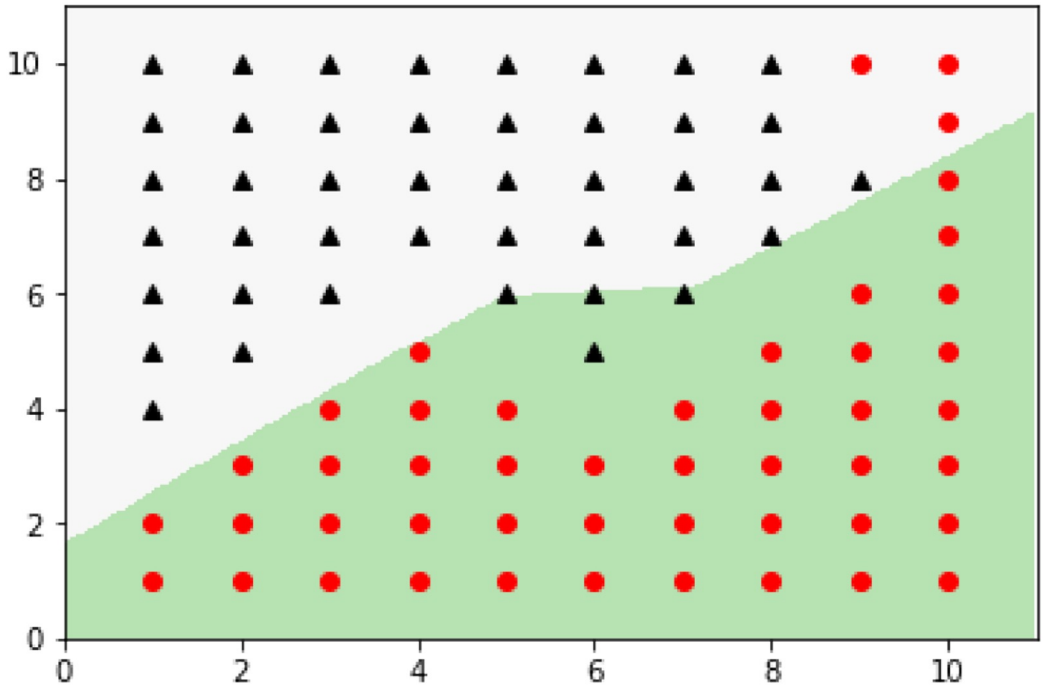
Example 3

$H = 32$



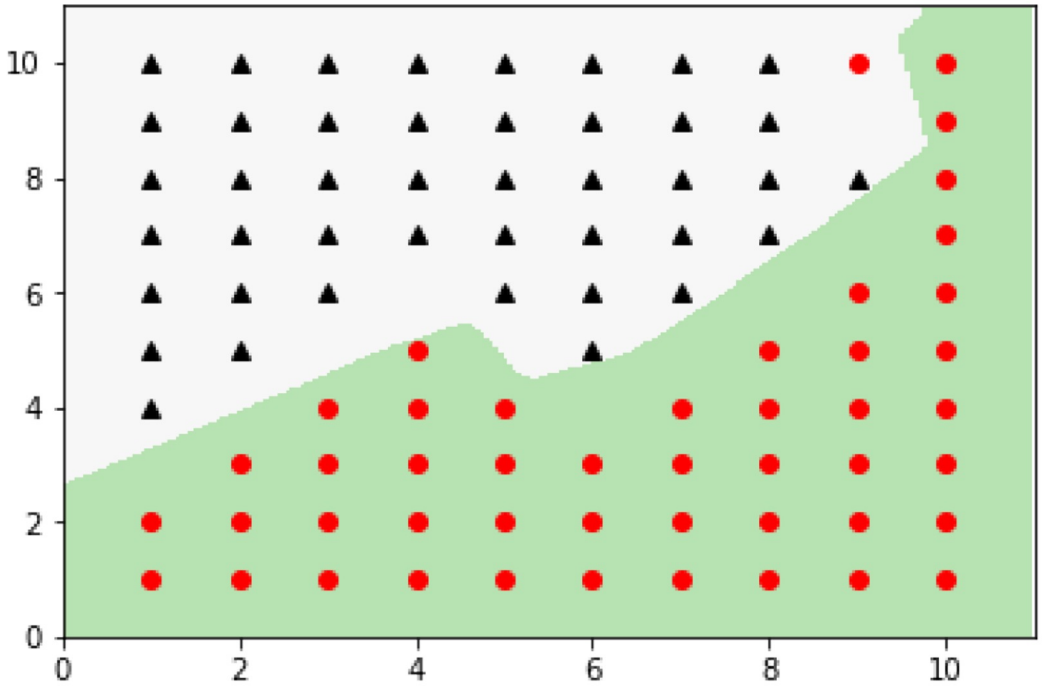
Example 3

$H = 64$



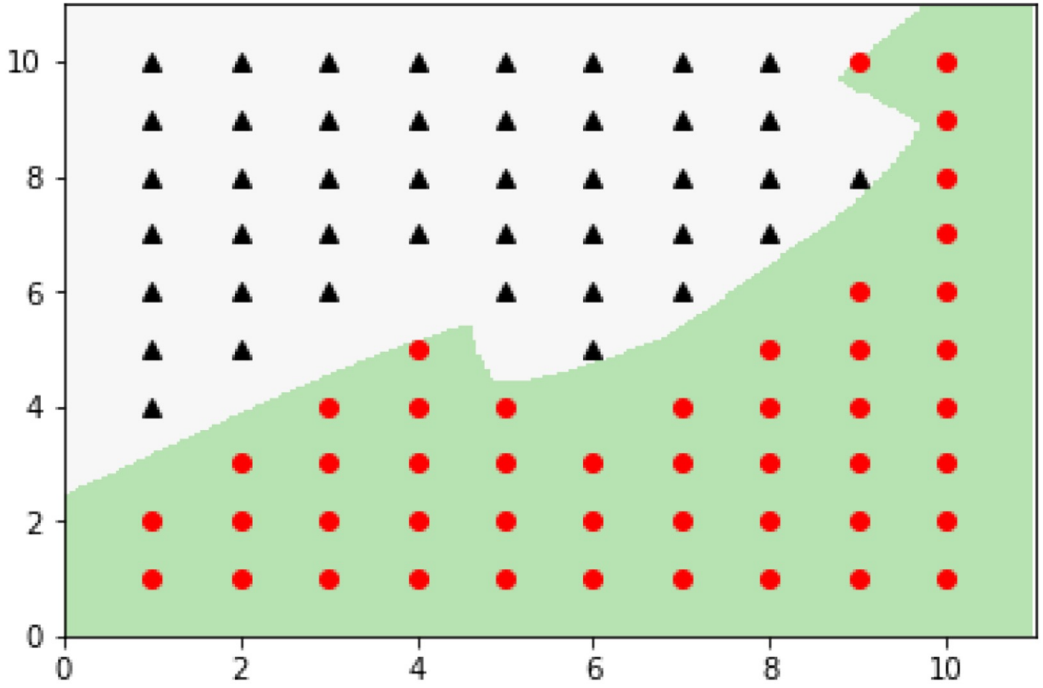
Example 3

$H = 64$



Example 3

$H = 64$



PyTorch Snippet

Declaring & initializing the network:

```
d, H = 2, 8
model = torch.nn.Sequential(
    torch.nn.Linear(d, H),
    torch.nn.ReLU(),
    torch.nn.Linear(H, 1),
    torch.nn.Sigmoid())
lossfn = torch.nn.BCELoss()
```

A gradient step:

```
ypred = model(x)
loss = lossfn(ypred, y)
model.zero_grad()
loss.backward()
with torch.no_grad():
    for param in model.parameters():
        param -= eta * param.grad
```