### PART 1

### Step 5: Training and validation errors

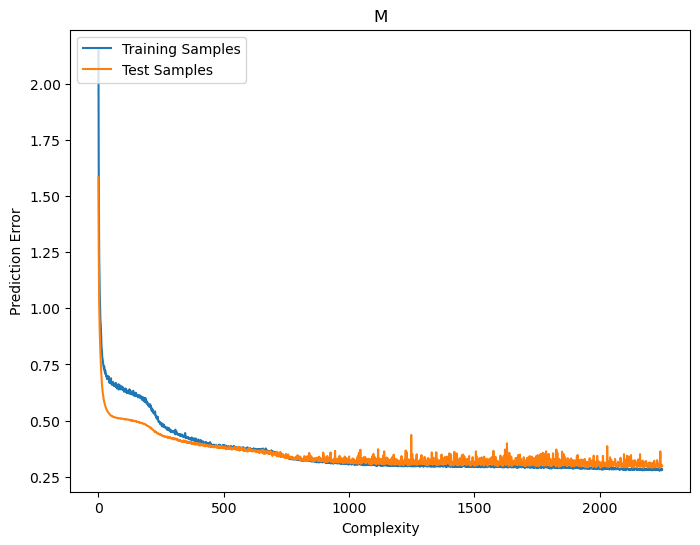
- Activation:relu, Learning Rate:0.1, Epoch:1000  
 Training and validation errors: 0.4189, 0.4102  
  
 - Activation:relu, Learning Rate:0.1, Epoch:1500  
 Training and validation errors: 0.4109, 0.4110  
  
 - Activation:relu, Learning Rate:0.1, Epoch:2250  
 Training and validation errors: 0.4475, 0.4262  
  
-----------------------------------------------------  
  
 - Activation:relu, Learning Rate:0.01, Epoch:1000  
 Training and validation errors: 0.4116, 0.4408  
  
 - Activation:relu, Learning Rate:0.01, Epoch:1500  
 Training and validation errors: 0.4276, 0.4470  
  
 - Activation:relu, Learning Rate:0.01, Epoch:2250  
 Training and validation errors: 0.4123, 0.4272  
  
-----------------------------------------------------  
  
 - Activation:relu, Learning Rate:0.001, Epoch:1000  
 Training and validation errors: 0.6761, 0.5218  
  
 - Activation:relu, Learning Rate:0.001, Epoch:1500  
 Training and validation errors: 0.6906, 0.5304  
  
 - Activation:relu, Learning Rate:0.001, Epoch:2250  
 Training and validation errors: 0.6286, 0.4886  
  
//////////////////////////////////////////////////////  
  
 - Activation:tanh, Learning Rate:0.1, Epoch:1000  
 Training and validation errors: 0.3740, 0.3865  
  
 - Activation:tanh, Learning Rate:0.1, Epoch:1500  
 Training and validation errors: 0.3797, 0.3562  
  
 - Activation:tanh, Learning Rate:0.1, Epoch:2250  
 Training and validation errors: 0.3644, 0.4170  
  
-----------------------------------------------------  
  
 - Activation:tanh, Learning Rate:0.01, Epoch:1000  
 Training and validation errors: 0.3934, 0.3621  
  
 - Activation:tanh, Learning Rate:0.01, Epoch:1500  
 Training and validation errors: 0.3652, 0.4034  
  
 - Activation:tanh, Learning Rate:0.01, Epoch:2250  
 Training and validation errors: 0.3831, 0.4033  
  
-----------------------------------------------------  
  
 - Activation:tanh, Learning Rate:0.001, Epoch:1000  
 Training and validation errors: 0.6706, 0.4939  
  
 - Activation:tanh, Learning Rate:0.001, Epoch:1500  
 Training and validation errors: 0.6033, 0.5012  
  
 - Activation:tanh, Learning Rate:0.001, Epoch:2250  
 Training and validation errors: 0.6104, 0.5130  
  
//////////////////////////////////////////////////////  
  
 - Activation:sigmoid, Learning Rate:0.1, Epoch:1000  
 Training and validation errors: 0.4583, 0.4584  
  
 - Activation:sigmoid, Learning Rate:0.1, Epoch:1500  
 Training and validation errors: 0.4607, 0.4835  
  
 - Activation:sigmoid, Learning Rate:0.1, Epoch:2250  
 Training and validation errors: 0.3817, 0.4198  
  
-----------------------------------------------------  
  
 - Activation:sigmoid, Learning Rate:0.01, Epoch:1000  
 Training and validation errors: 0.7820, 0.6317  
  
 - Activation:sigmoid, Learning Rate:0.01, Epoch:1500  
 Training and validation errors: 0.7002, 0.5304  
  
 - Activation:sigmoid, Learning Rate:0.01, Epoch:2250  
 Training and validation errors: 0.7560, 0.5918  
  
-----------------------------------------------------  
  
 - Activation:sigmoid, Learning Rate:0.001, Epoch:1000  
 Training and validation errors: 1.3462, 1.1671  
  
 - Activation:sigmoid, Learning Rate:0.001, Epoch:1500  
 Training and validation errors: 1.3464, 1.1625  
  
 - Activation:sigmoid, Learning Rate:0.001, Epoch:2250  
 Training and validation errors: 1.2918, 1.1248

### Step 7: My best parameters

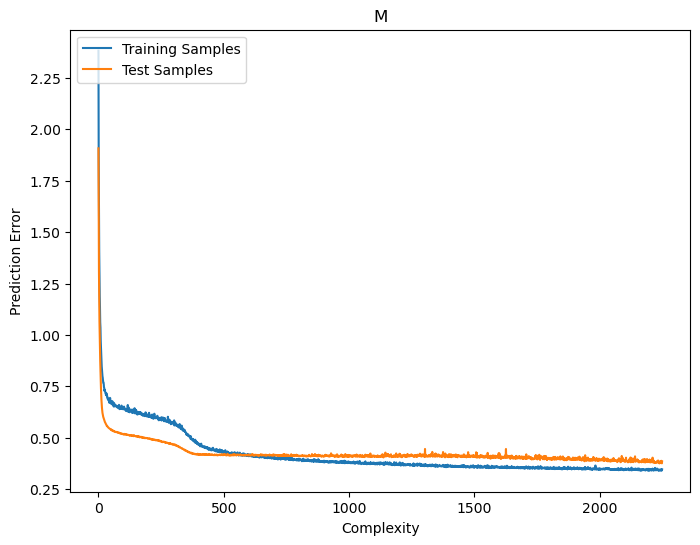
Best parameters: {'activation': '**tanh**', 'learning\_rate': **0.1**, 'epoch': **1500**}

### Step 8: Bias and variance curve

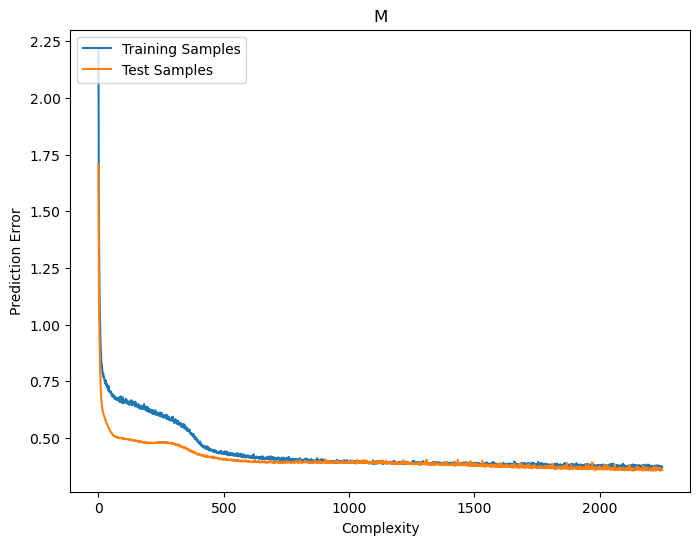
The first hidden layer with two nodes results.



The second hidden layer with two nodes results.

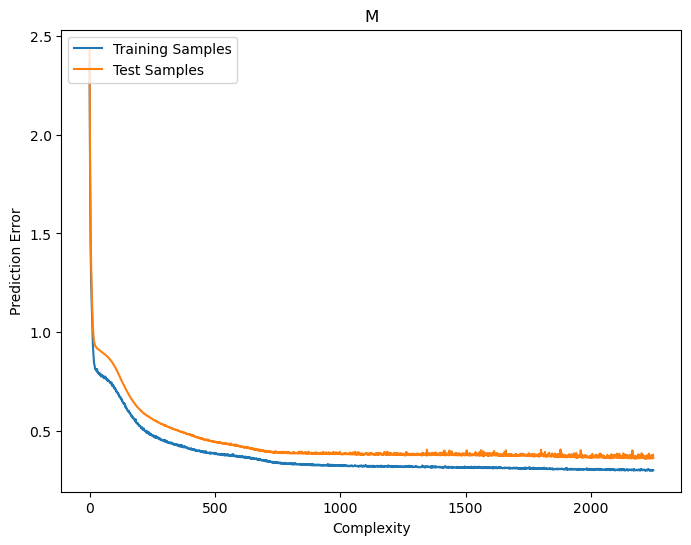


The third hidden layer with two nodes results.

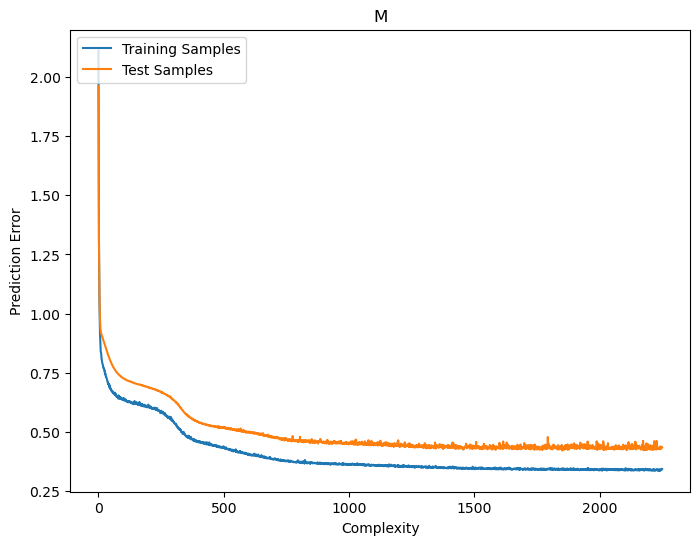


### Step 9: Increase 𝑁𝑡 by 10%

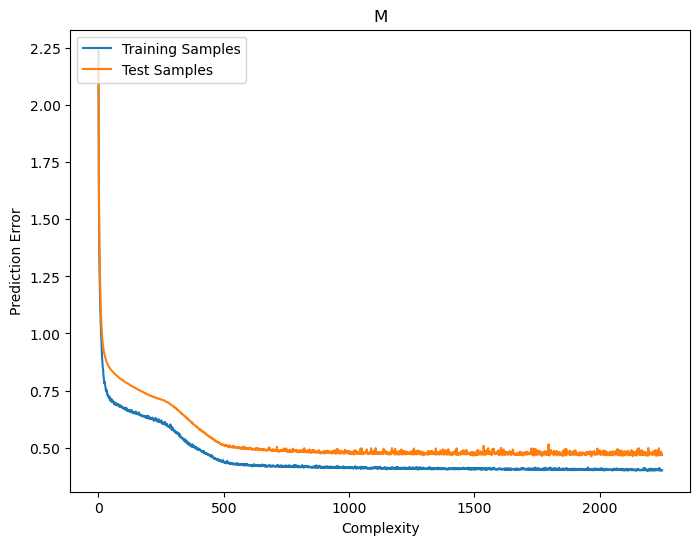
The first hidden layer with two nodes results.



The second hidden layer with two nodes results.



The third hidden layer with two nodes results.

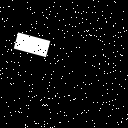
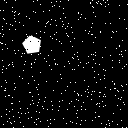
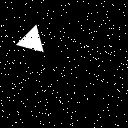
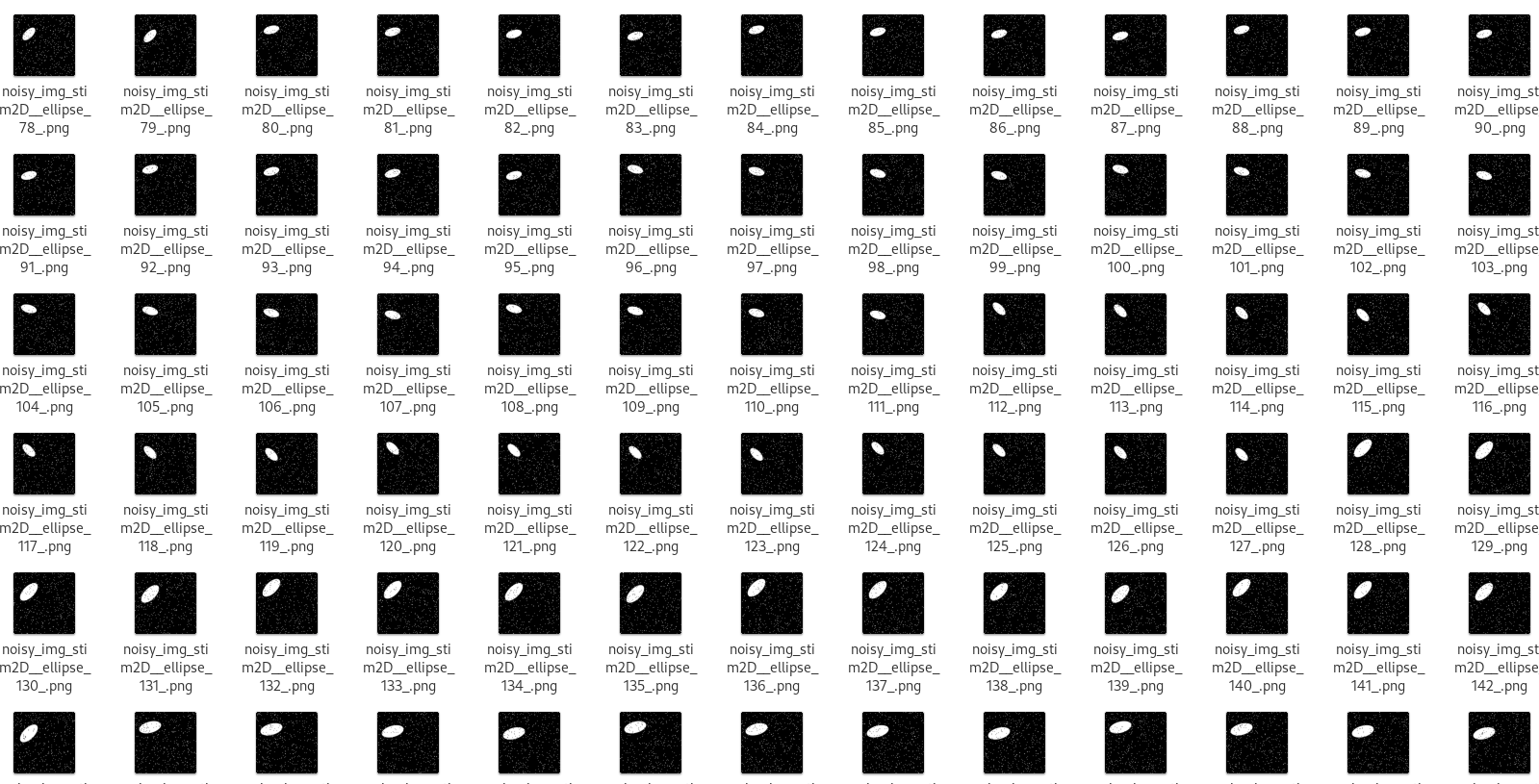


### PART 2

I generate 256 images for each class, so I have 2048 images via below code pieces.

*python main.py --shapes ellipse rect poly3 poly5 poly6 poly7 star5 star8 --to\_transform scale rota trx try --num\_transformations 4 -stim\_ellipse\_ratio 0.5 0.7*

My Sample Data Images, for example elipses



### Train Report