ASSIGNMENT-3 REPORT

-ISHMEET KAUR(2015042)

Q1.

a) Sigmoid:

0.7767 accuracy on subset

0.6745 accuracy on full dataset

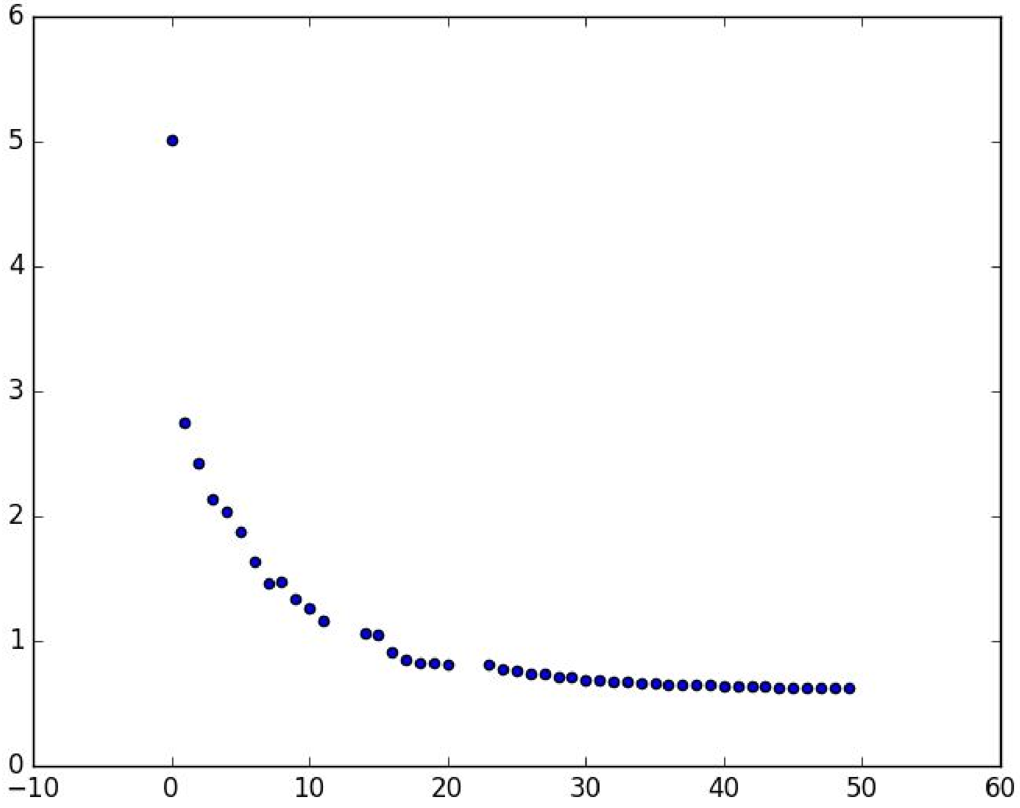
b) Relu:

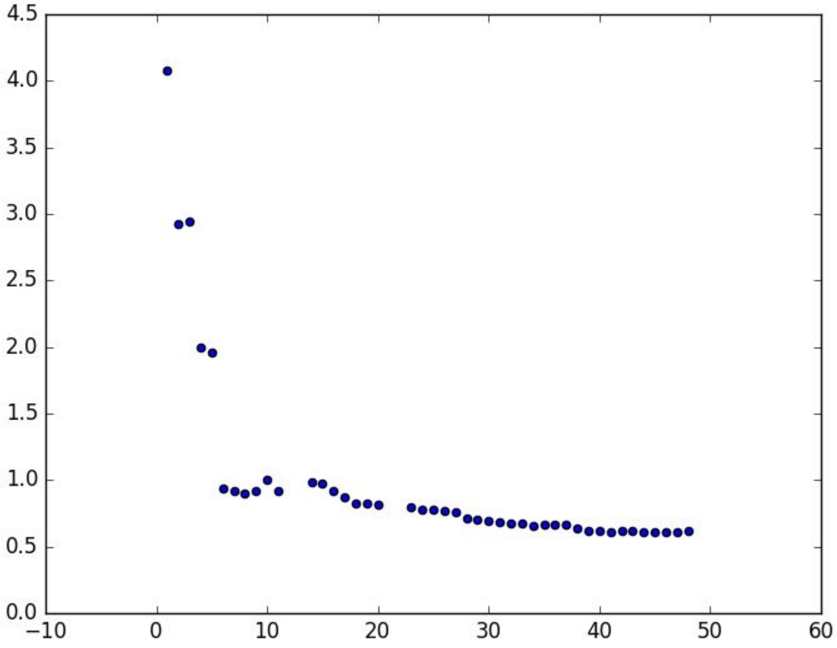
0.9723 accuracy on subset

0.9189 accuracy on full dataset

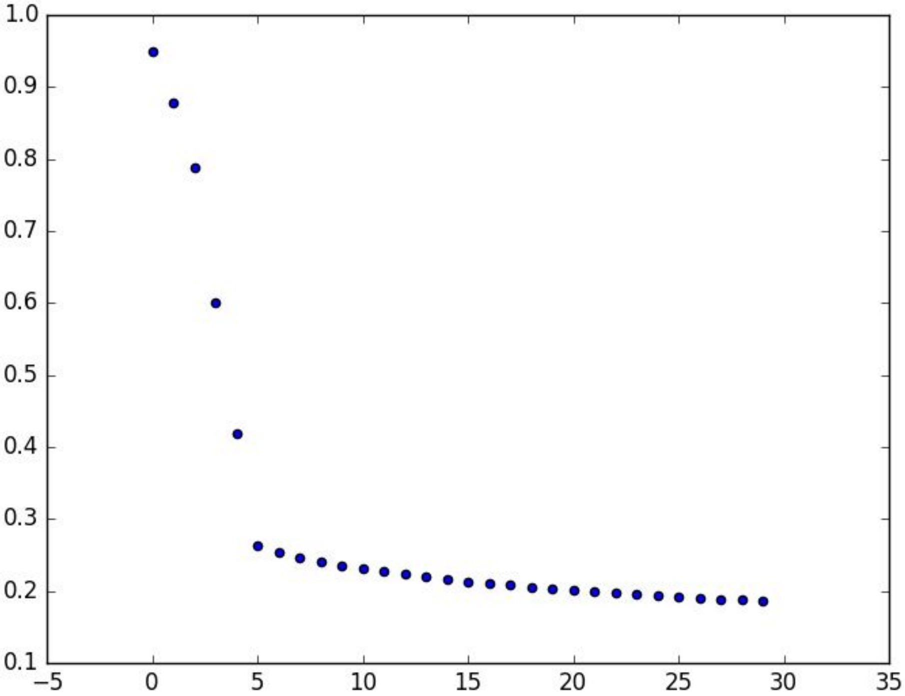
PLOTS:

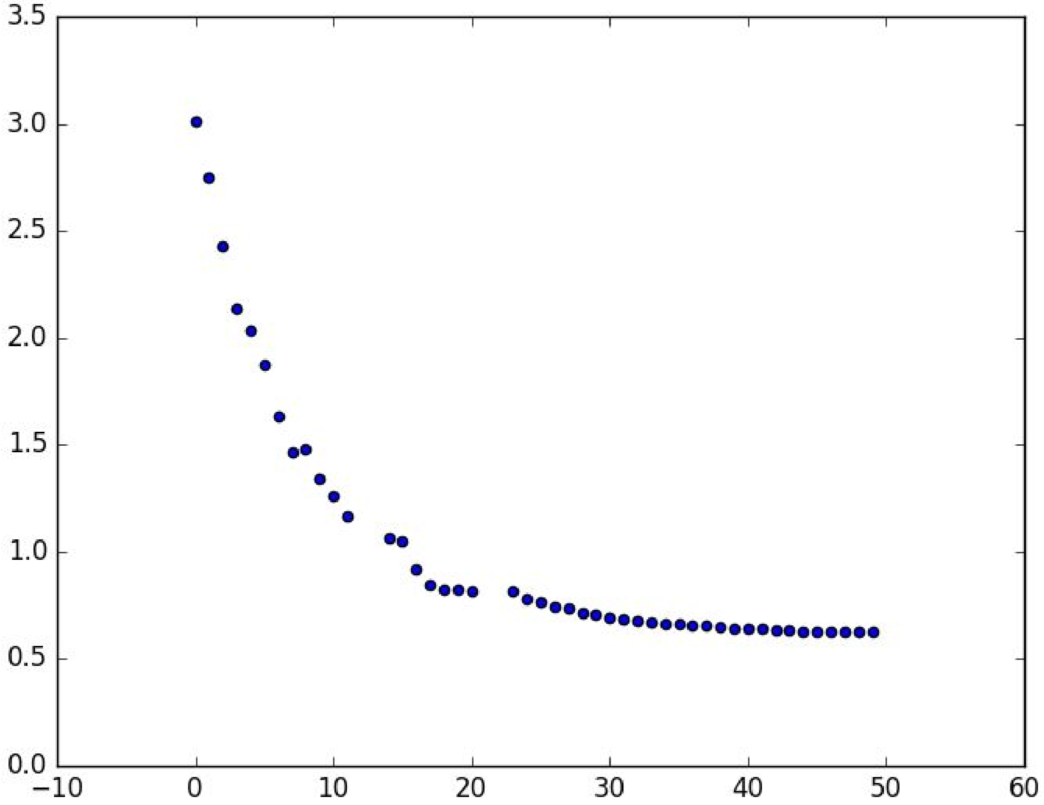
a) SIgmoid




4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
−10
0
10
20
30
40
50
60

b) Relu


1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
−5
0
5
10
15
20
25
30
35


3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
−10
0
10
20
30
40
50
60

Q2.

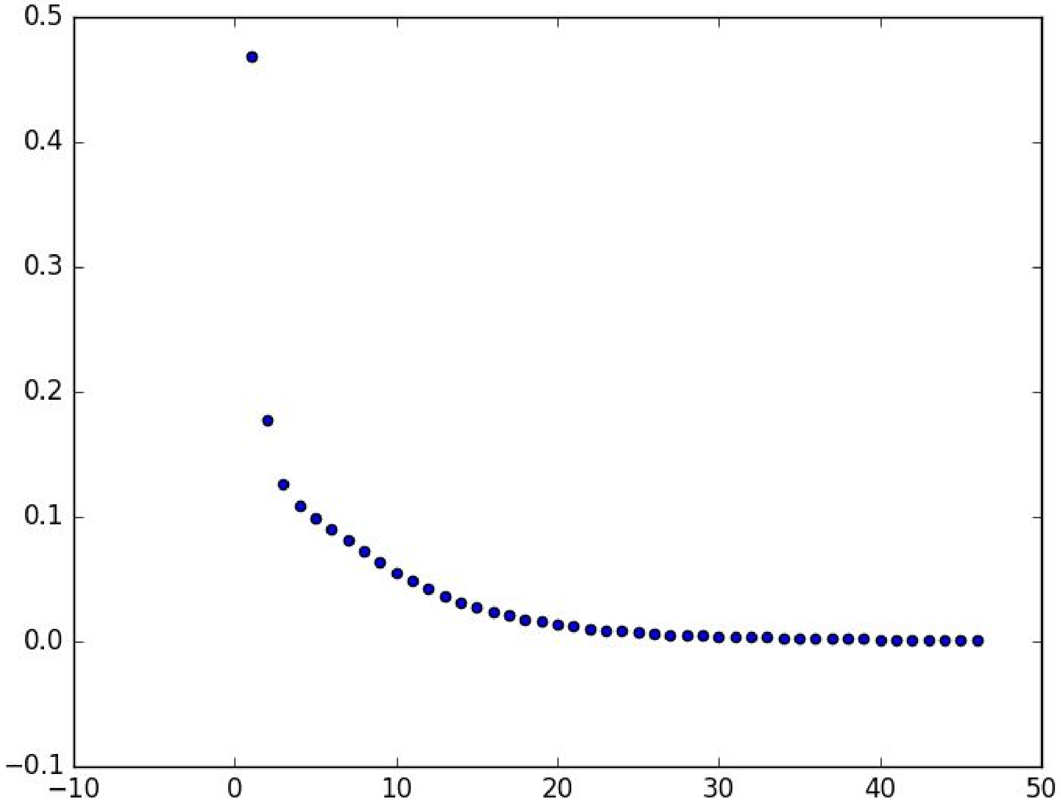
a)

Accuracy= 0.9867

Better than 1a by 0.9867- 0.9723 = 0.0144

MLP classifier takes a lot of different parameters which improves accuracy

This is the graph of epoch vs loss:



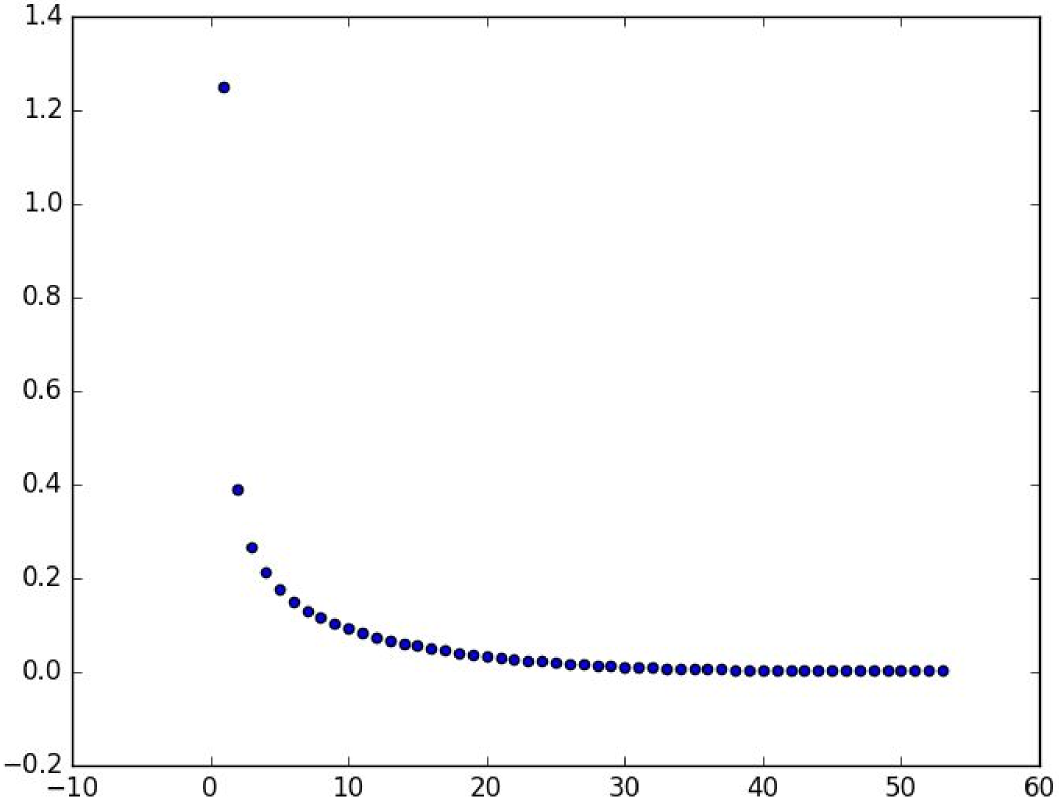
b)

Accuracy= 0.9774

Better than 1b by 0.9774- 0.9189 = 0.0585

MLP classifier takes a lot of different parameters which improves accuracy

Graph:



Q3.

First Model:

Simple 1 hidden layer network. n=100. No extra parameters.

```
 mlp = MLPClassifier(hidden_layer_sizes=(100), activation='relu',max_iter=100,verbose=10)
```

Second Model:

3 layer network. n=100,50,25. Some more parameters added

```
mlp = MLPClassifier(hidden_layer_sizes=(100,50,25),  activation='logistic', max_iter=100,
verbose=10, alpha=1e-4, solver='sgd',tol=1e-4, random_state=1,  learning_rate_init=.1)
```

Third Model:

2 layer network:

```
mlp = MLPClassifier(hidden_layer_sizes=(100,50),  activation='logistic', max_iter=100,
verbose=10, alpha=1e-4, solver='sgd',tol=1e-4, random_state=1,  learning_rate_init=.1)
```

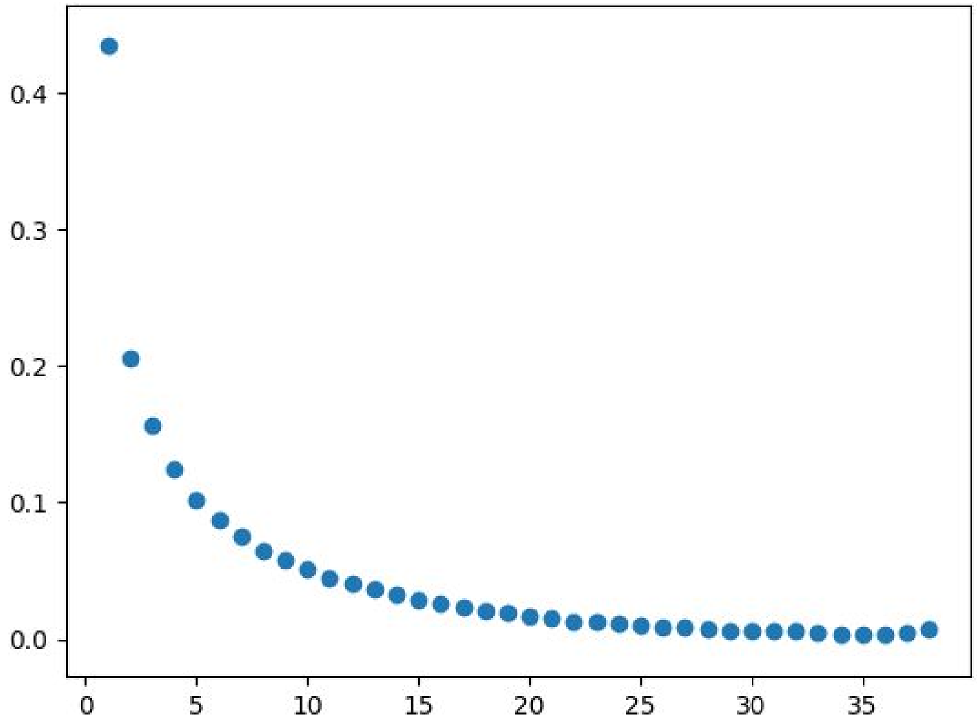
Fourth Model:

1 layer, n=1000 with extra parameters
mlp = MLPClassifier(hidden_layer_sizes=(1000), activation='logistic', max_iter=100, verbose=10, alpha=1e-4, solver='sgd',tol=1e-4, random_state=1, learning_rate_init=.1)

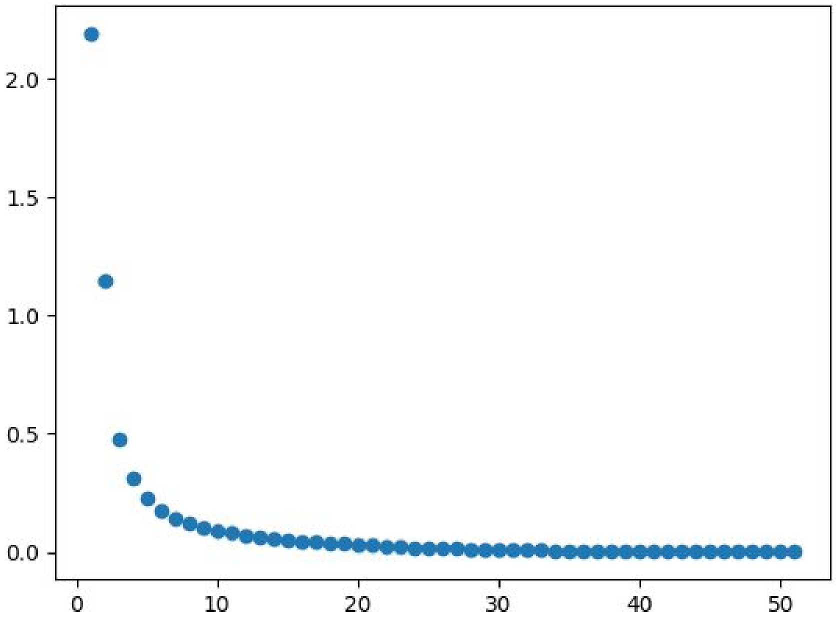
Accuracies:
0.9768, 0.9755,0.9783,0.981

Best model and Explanation:
Model 3
Model 1: underfitting
Model 2: overfitting
Model 3: ideal
Model 4: we should have gotten the max acc for 2-layer, but since n=1000 in model 1, it performs better. The machine took a lot of time to run with such high value for multiple layers.

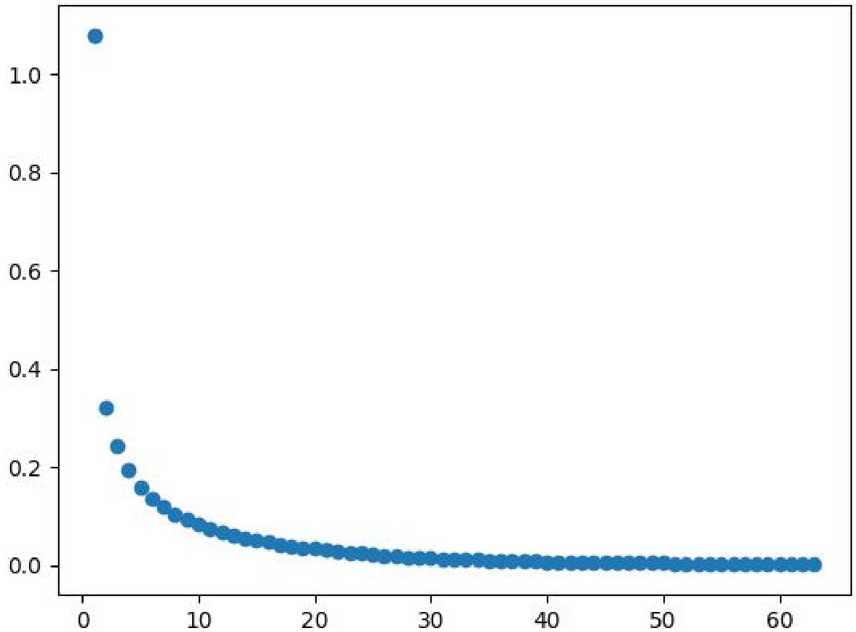
Plots:
Model 1:



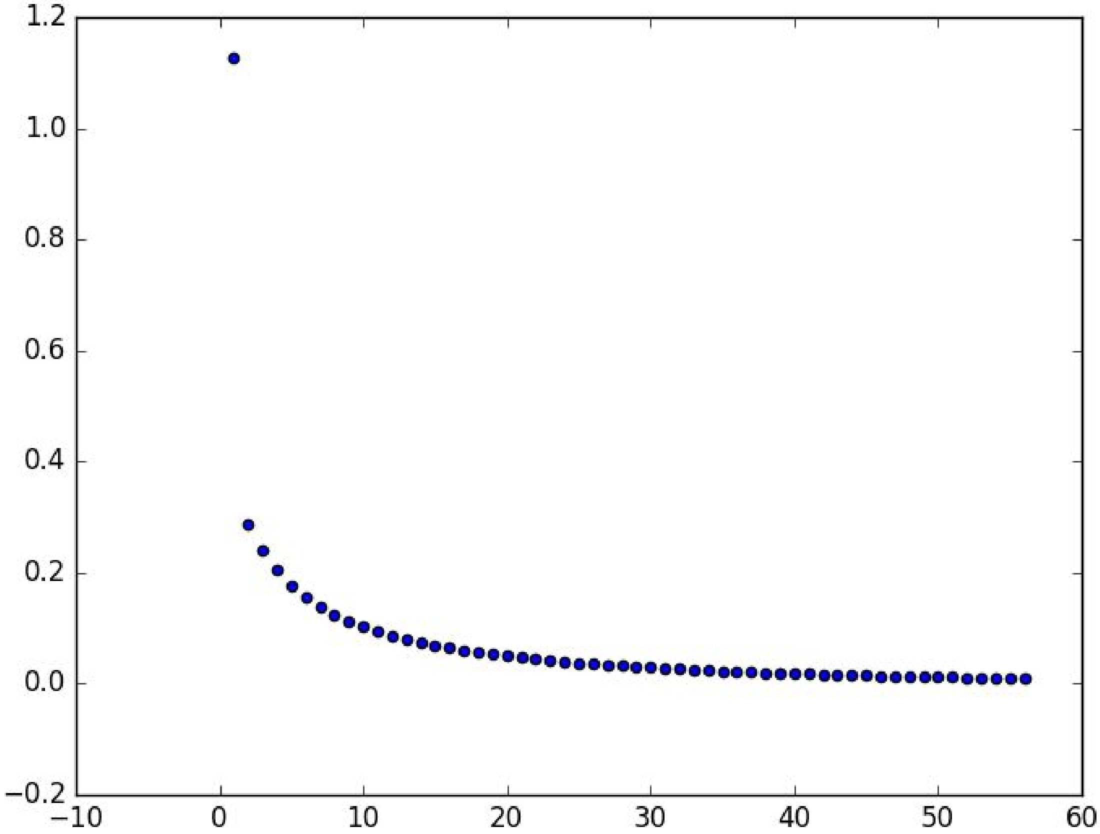
Model 2:



Model 3:



Model 4:


1.2
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−10
0
10
20
30
40
50
60