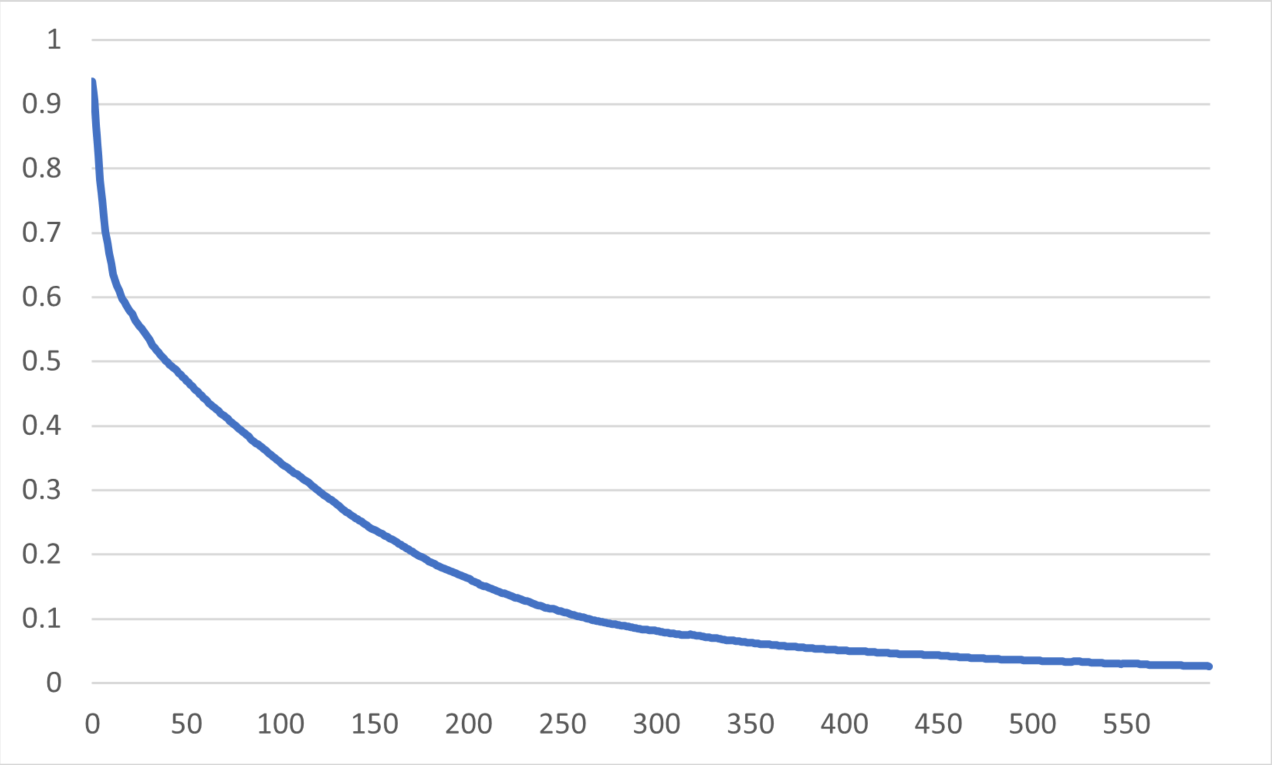
Terminal output:

```
   cough  cold  diarrhea  sore_throat  body_pain  headache  temperature  difficult_breathing  fatigue  travelled14  travel_covid  covid_contact  age  infection_probability
0      0     0         1            1          1         0            1                    0        0            3             0              0    2                      0
1      0     1         0            0          0         1            0                    0        0            0             0              3    2                      0
2      1     1         0            0          0         0            0                    0        0            0             0              3    2                      0
3      0     1         0            1          1         0            0                    2        2            0             0              3    0                      0
4      0     1         1            0          0         1            1                    0        2            3             3              0    2                      1

[5 rows x 14 columns]
Iteration 1, loss = 0.93542502
Iteration 2, loss = 0.90566317
Iteration 3, loss = 0.86604167
Iteration 4, loss = 0.81999387
Iteration 5, loss = 0.78118027
 ...
Iteration 591, loss = 0.02646653
Iteration 592, loss = 0.02638061
Iteration 593, loss = 0.02638143
Iteration 594, loss = 0.02633723
Iteration 595, loss = 0.02626700
Training loss did not improve more than tol=0.000100 for 10 consecutive epochs. Stopping.
[[2640    1]
 [  13 1350]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      2641
           1       1.00      0.99      0.99      1363

    accuracy                           1.00      4004
   macro avg       1.00      1.00      1.00      4004
weighted avg       1.00      1.00      1.00      4004

accuracy: 0.9965034965034965
precision: 0.9971798472352151
recall: 0.9950417856239008
confusion matrix:
[[2640    1]
 [  13 1350]]
prediction: [0, 1, 1, 0, 1, 1, 0, 2, 2, 3, 3, 0, 0] => [1]
```

Model properties

Property	Value
Accuracy	`0.9965034965034965`
Precision	`0.9971798472352151`
Recall	`0.9950417856239008`
Confusion matrix	`[2640 1], [13 1350]`

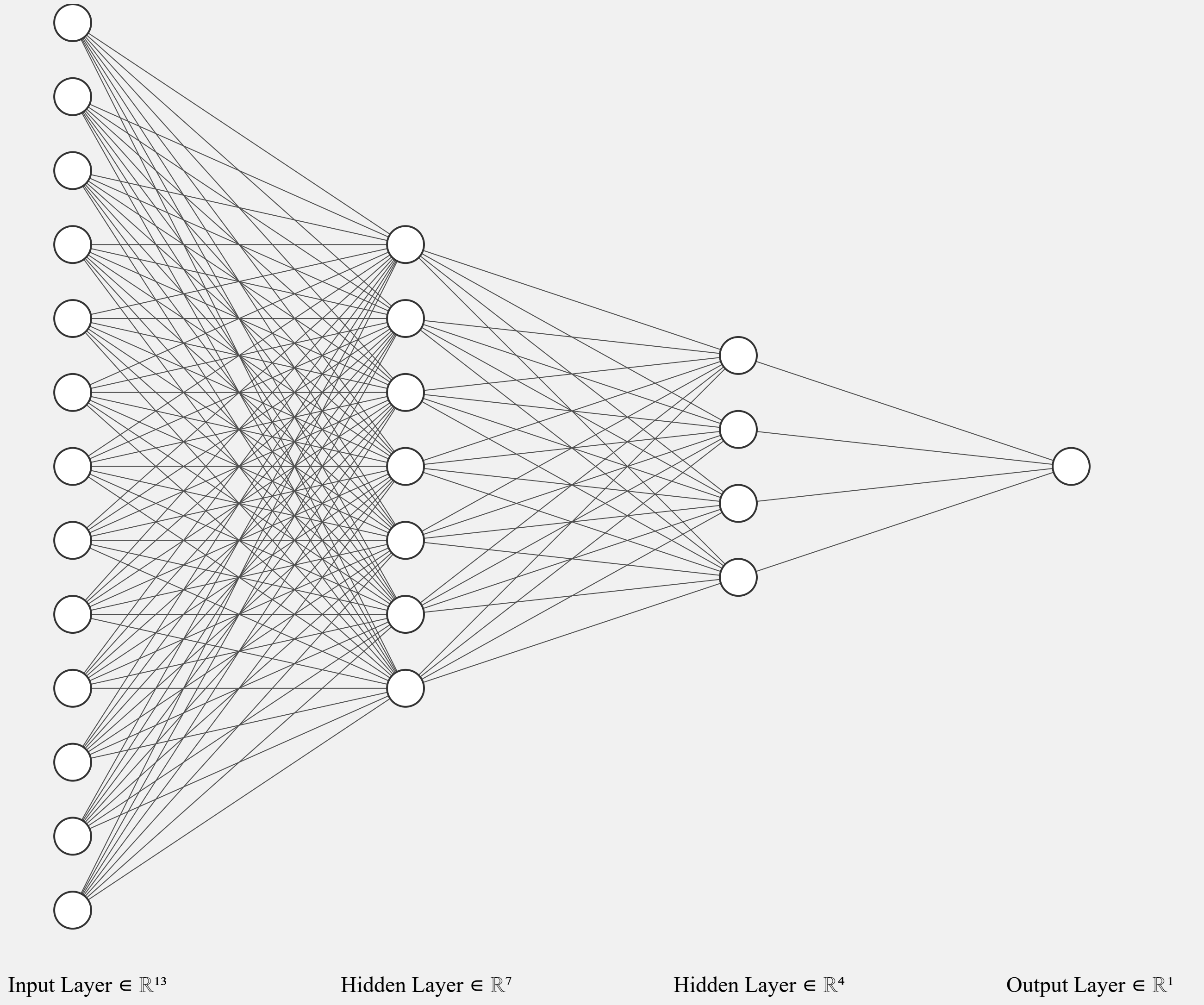
> Loss against iteration:
>
> 
>

Prediction

The model predicted the probability of infection given `[0, 1, 1, 0, 1, 1, 0, 2, 2, 3, 3, 0, 0]` to be `1`.

Network design

From the input/output relationship of the problem, the model is required to compile data from 13 input variables into 1 single output variable. Considering that the problem does not require very particularly complex computation, I decided that a simple funnel-shaped network would suffice.
Since there are 13 input data (symptoms/ risk factors), for the first hidden layer, I divided the input size in half, rounded up, and used 7 neurons. The same is done for the second hidden layer, which was allocated 4 neurons. I opted not to add a third hidden layer of size 2 since it increases the network complexity and computation time without much observable benefits.
In the end, the network has `hidden_layer_sizes` = `[7, 4]`, with a final network architecture of `13 -> 7 -> 4 -> 1`.

> Neural network architecture schematic, generated with NN-SVG:
>
> 
>

Hypermeter optimisation

I have identified a few hypermeters that can be subjected to optimisation.

For `test_size` in `model_selection.train_test_split()`, Gholamy et. al. (2018) proved that a test size of ~80% gives the best results for neural networks empirically. Hence, I have chosen `test_size` = `0.8`.

For `solver`, in preliminary trials, `"sgd"` seems to give the best resultant accuracy. Hence, I have chosen `solver`=`"sgd"`. The below hypermeters are optimised with respect to `"sgd"`.

For `max_iter`, this hypermeter is not expected to affect the training characteristics of the model at all unless in extreme cases. Empirically, the training process never exceeded 1000 iterations before it is stopped due to insignificant loss improvement. Hence, I have set `max_iter` = `1000`.