Problem 1)

$$f(x,y) = \cos(\pi * x) * \cos(\pi * y)\ , (x,y) \in\ [50{,}54]^2$$

Training was done on a neural network with two hidden layers with 50 neurons per layer and a hyperbolic tangent function. Backpropagation was done on autograd by taking gradient of the loss function with respect to weights of each layer.

The neural network was trained for 50, 100, 250, 500 training points and the performance was noted. It is clearly evident that the neural network performs better as we increase the number of training points, as the approximation error in the relative L2 norm goes on reducing.

Sr no	Training points	Iterations	Learning rate	L2 error	Reference Fig
1	50	20000	0.0002	1.1284213083428591	Fig 2
2	100	20000	0.0002	0.6344937802226627	Fig 3
3	250	20000	0.0002	0.061863217097627925	Fig 4
4	500	20000	0.0002	0.021086139063713708	Fig 5
5	1000	20000	0.0002	0.016065990529665748	Fig 6

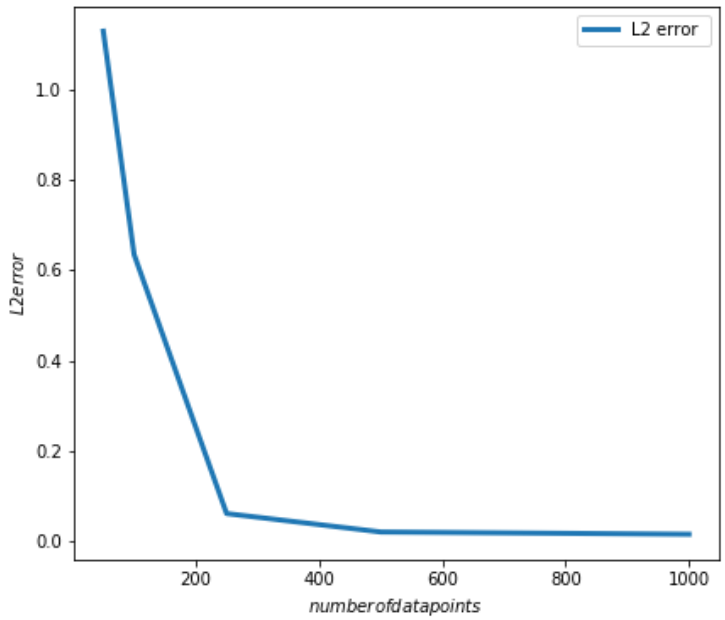


Figure 1 L2 error vs number of training points

The below graphs plot the exact output vs the predicted output. Ideal graph would be a straight line with 45^0 slope. We can see that as we increase the training points the graph becomes more linear.

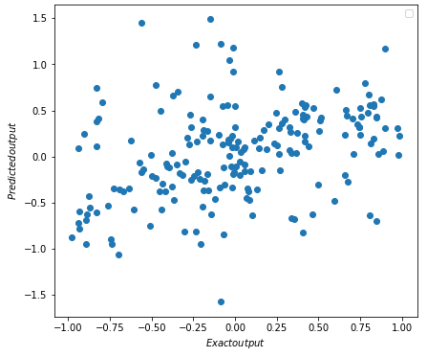


Figure 2: Training points = 50

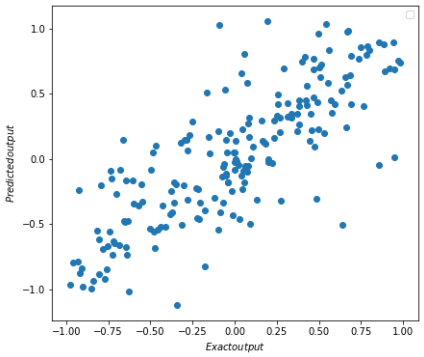


Figure 3: Training points = 100

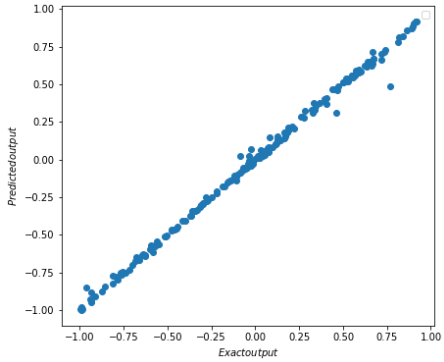


Figure 4 : Training points = 250

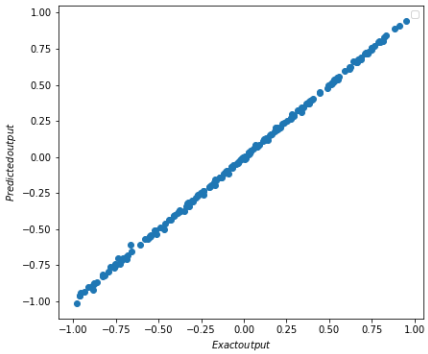


Figure 5: Training points = 500

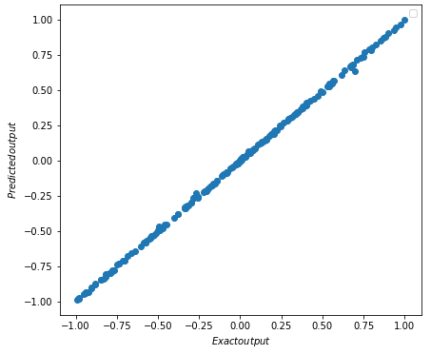
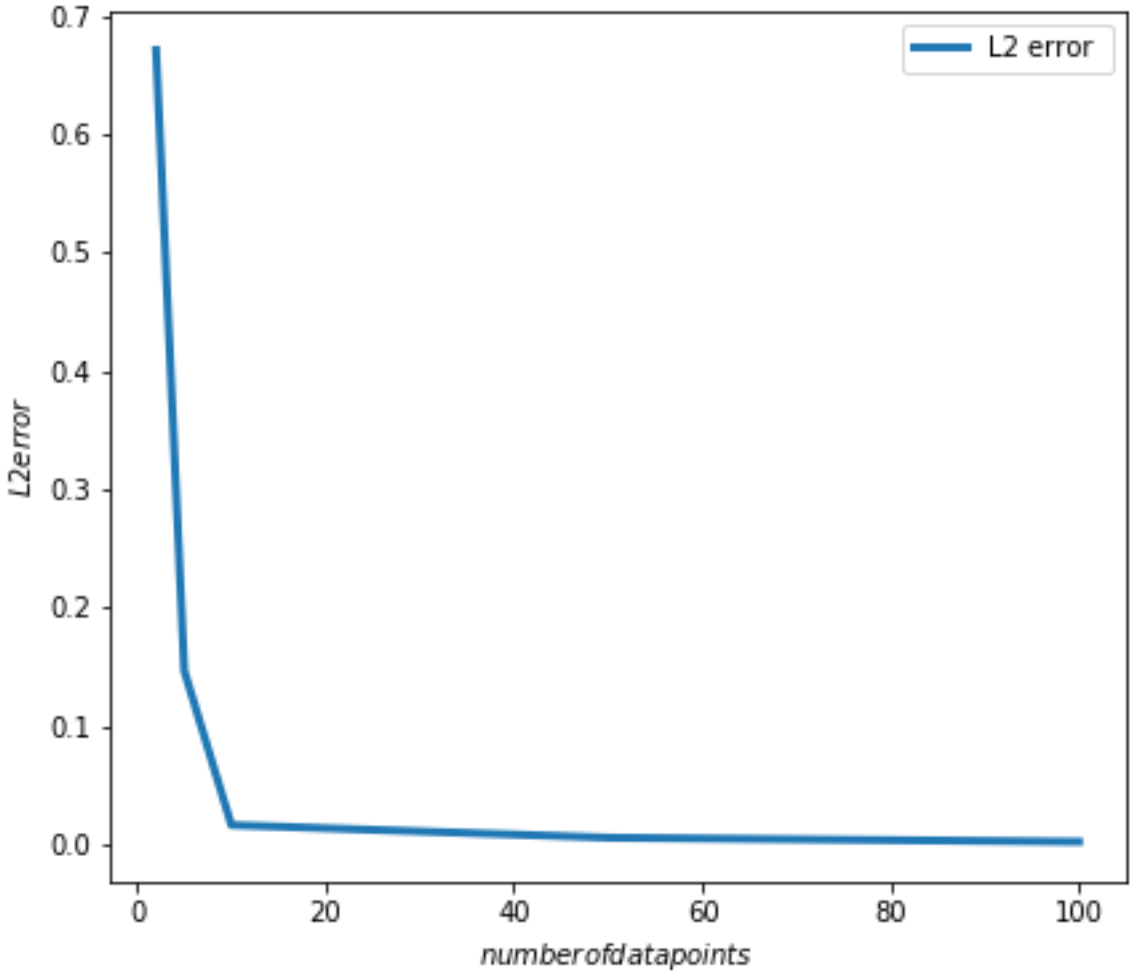


Figure 6: Training points = 1000

PROBLEM 2

On training the physics informed neural network for increasing values of N_f (training points) we see that the L2 error reduces. The advantage of this network is clearly visible that we need very few data points to train the network.

Sr no	Training points	Iterations	L2 loss
1	2	1000	0.6713805072995791
2	5	1000	0.1463593906044692
3	10	1000	0.01602994649456709
4	50	1000	0.005376072743316586
5	100	1000	0.001951188703966486



PROBLEM 3

When there are no hidden layers the neural network is basically a logistic regression and hence the accuracy in that case is similar to the accuracy of logistic regression.

When we increase the number of layers the accuracy increases. Also, the accuracy increases as we increase the number of nodes in each layer.

Sr no	Hidden Layers	Iterations	Nodes	Accuracy	
1	0	20000	-	0. 8010801080108011	$\begin{bmatrix} 834 & 389 \\ 274 & 1836 \end{bmatrix}$
2	1	20000	20	0. 9345934593459346	$\begin{bmatrix} 1100 & 123 \\ 95 & 2015 \end{bmatrix}$
3			50	0.9498949894989499	$\begin{bmatrix} 1128 & 95 \\ 72 & 2038 \end{bmatrix}$
4			100	0.9588958895889589	$\begin{bmatrix} 1143 & 80 \\ 57 & 2053 \end{bmatrix}$
5	2	20000	20	0.9375937593759376	$\begin{bmatrix} 1104 & 119 \\ 89 & 2021 \end{bmatrix}$
6			50	0.9486948694869487	$\begin{bmatrix} 1100 & 123 \\ 48 & 2062 \end{bmatrix}$
7			100	0.9537953795379538	$\begin{bmatrix} 1115 & 108 \\ 46 & 2064 \end{bmatrix}$
8	3	20000	20	0.9420942094209421	$\begin{bmatrix} 1106 & 117 \\ 76 & 2034 \end{bmatrix}$
9			50	0.9528952895289529	$\begin{bmatrix} 1125 & 98 \\ 59 & 2051 \end{bmatrix}$
10			100	0.9576957695769577	$\begin{bmatrix} 1133 & 90 \\ 51 & 2059 \end{bmatrix}$