

Question 3:

Number of neurons in hidden layers :

Number of neurons in hidden layer 1 is 128

Number of neurons in hidden layer 2 is 10

The total number of neurons are 138.

Experimenting with different values of neurons:

Experimented with different values of neurons in the dense layer. The following table shows how the accuracy of the model changes when number of neurons are changed:

SL NO	No of neurons	Test Accuracy
1	128	88.38%
2	64	86.84%
3	32	87.19%
4	256	88.22%

Observations:

- For 128 neurons, the accuracy was 88.38%.
- The accuracy dropped when the number of neurons dropped from 128. For 64 and 32 neurons, the accuracy observed is 87.65% and 86.82% respectively.
- We can see a slight decrease in test accuracy when increasing the number of neurons from 128 to 256. The accuracy increased from 88.38% to 88.22%.
- Overall, the accuracy is highest with number of neurons equal to 128.

Experimenting with dropout layer with different values:

Experimented with different dropout rate values. The following table shows how accuracy changes when the dropout rate changes:

SL NO	Dropout Percentage	No of Neurons	Test Accuracy
1	0.2	128	88.02%
2	0.3	128	87.34%
3	0.4	128	86.44%
4	0.2	256	88.35%
5	0.3	256	88.13%

Observations:

- The accuracy of the model is highest when the dropout rate is 0.2 with an accuracy of 88.02% for 128 neurons.
- The accuracy decreases with an increase in dropout rate from 87.34% with 0.3 dropout rate to 86.44% with a 0.4 dropout rate for 128 neurons.

- The pattern is similar for 256 neurons as well. The accuracy is 88.35% for a 0.2 dropout rate and 88.13% for a 0.3 dropout rate.
- Overall the accuracy is highest for 256 neurons with a dropout rate 0.2 with an accuracy of 88.35%

Implementing CNN :

- Added two convolution layers before flatten layer. The first convolution layer has 32 filters with filter size (3,3). The second convolution layer has 64 filter size (3,3).
- The following is the final model:

```
model_10 = tf.keras.Sequential([
    tf.keras.layers.Conv2D(32,(3,3), input_shape=(28, 28, 1), data_format="channels_last"),
    tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(128, activation='relu'),
    tf.keras.layers.Dropout(0.2),
    tf.keras.layers.Dense(10)
])
```

- The model has 90.53% test accuracy.

Experimenting with different numbers of filters in CNN:

Tried different values for filters in CNN. The following table shows how the test accuracy of the model changes with the number of filters:

SL No	Dropout Rate	Number of filters in Conv Layer 1	Number of filters in Conv Layer 2	Test Accuracy
1	0.2	32	64	90.53%
2	0.2	64	128	89.89%
3	0.2	16	32	90.39%

Observations:

- When the number of filters in Conv layer 1 is 32 and the number of filters in Conv layer 2 is 64, the test accuracy is 90.53% which is the highest accuracy noted for far.
- The accuracy decreases for both cases where we are doubling the number of filters and dividing the number of filters by 2.
- The accuracy is 89.89% for 64 in Conv 1 and 128 in Conv 2 which is the least accuracy noted so far.

Experimenting different size of filters in CNN:

Tried different sizes for filters in a convolution layer. The following table shows how the test accuracy changes with the size of the filter.

SL NO	Filter Sxe for Conv Layer 1	Filter size for Conv Layer 2	Test Accuracy
1	3,3	3,3	90.53%
2	5,5	5,5	89.34%
3	7,7	7,7	89.62%

Observations:

- The accuracy is highest when the filter size for both the conv layers is 3,3 which is 90.53%.
- For both 5,5 and 7,7 filter sizes, the accuracy is approximately equal to 89%.