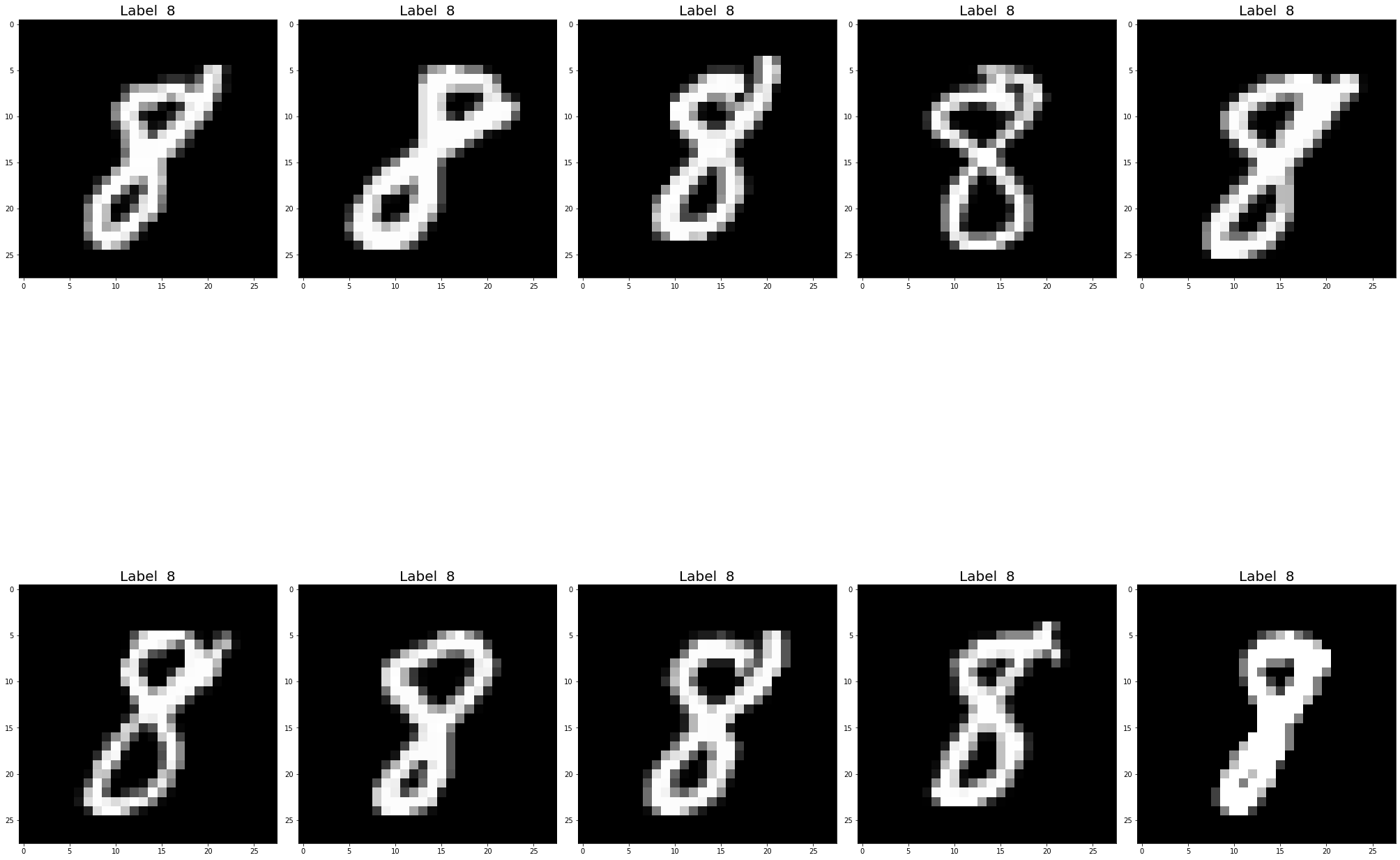
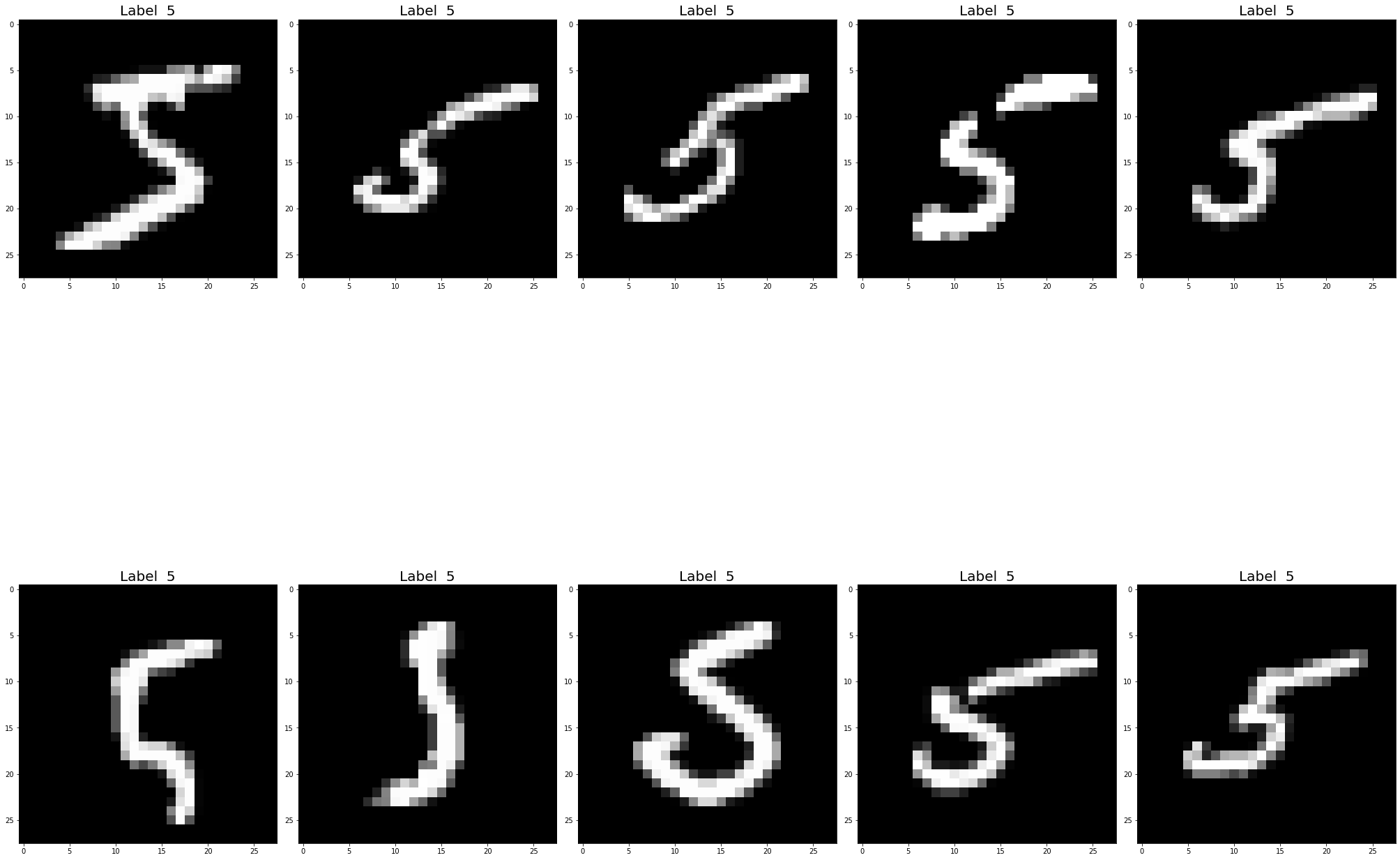
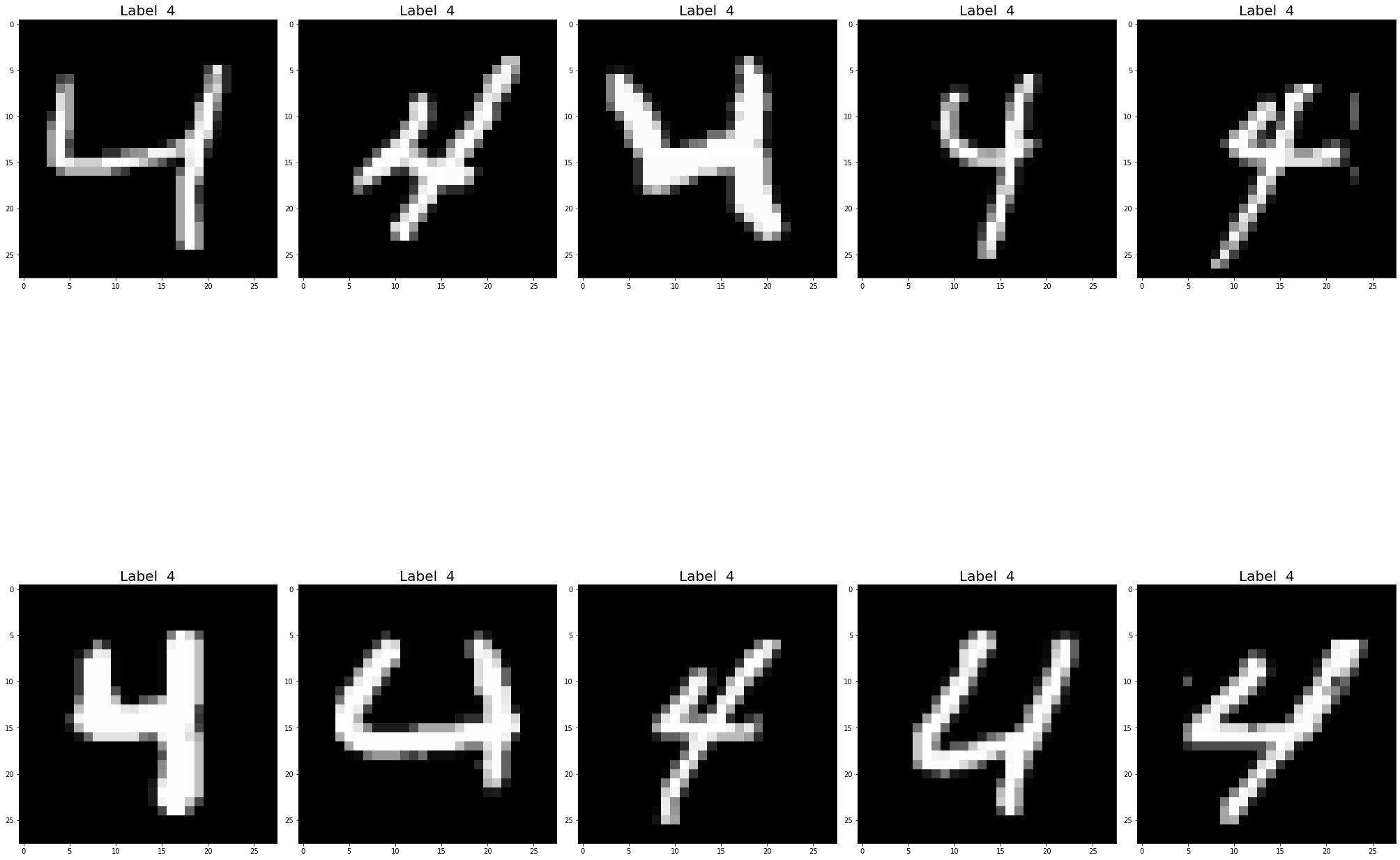
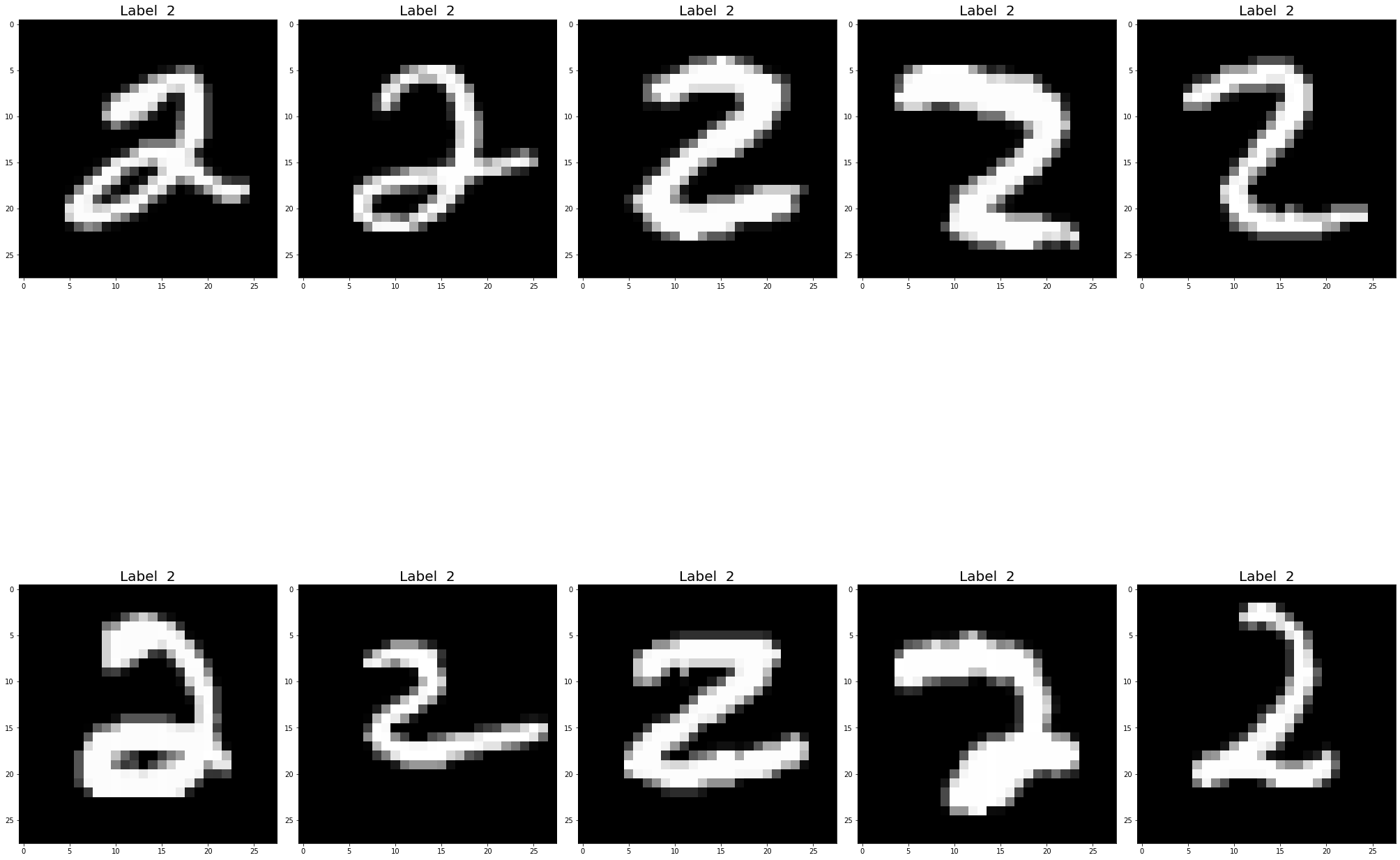
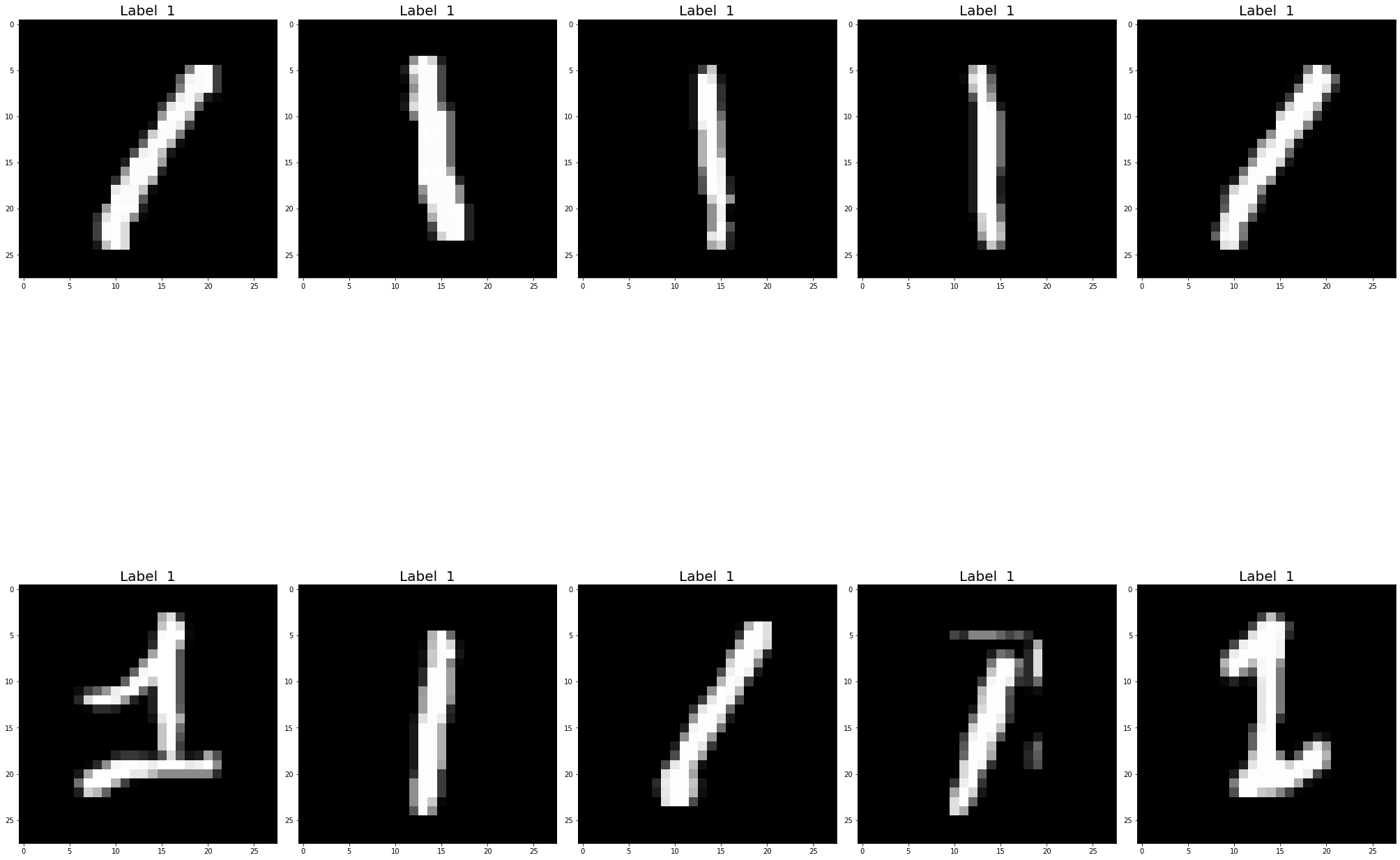
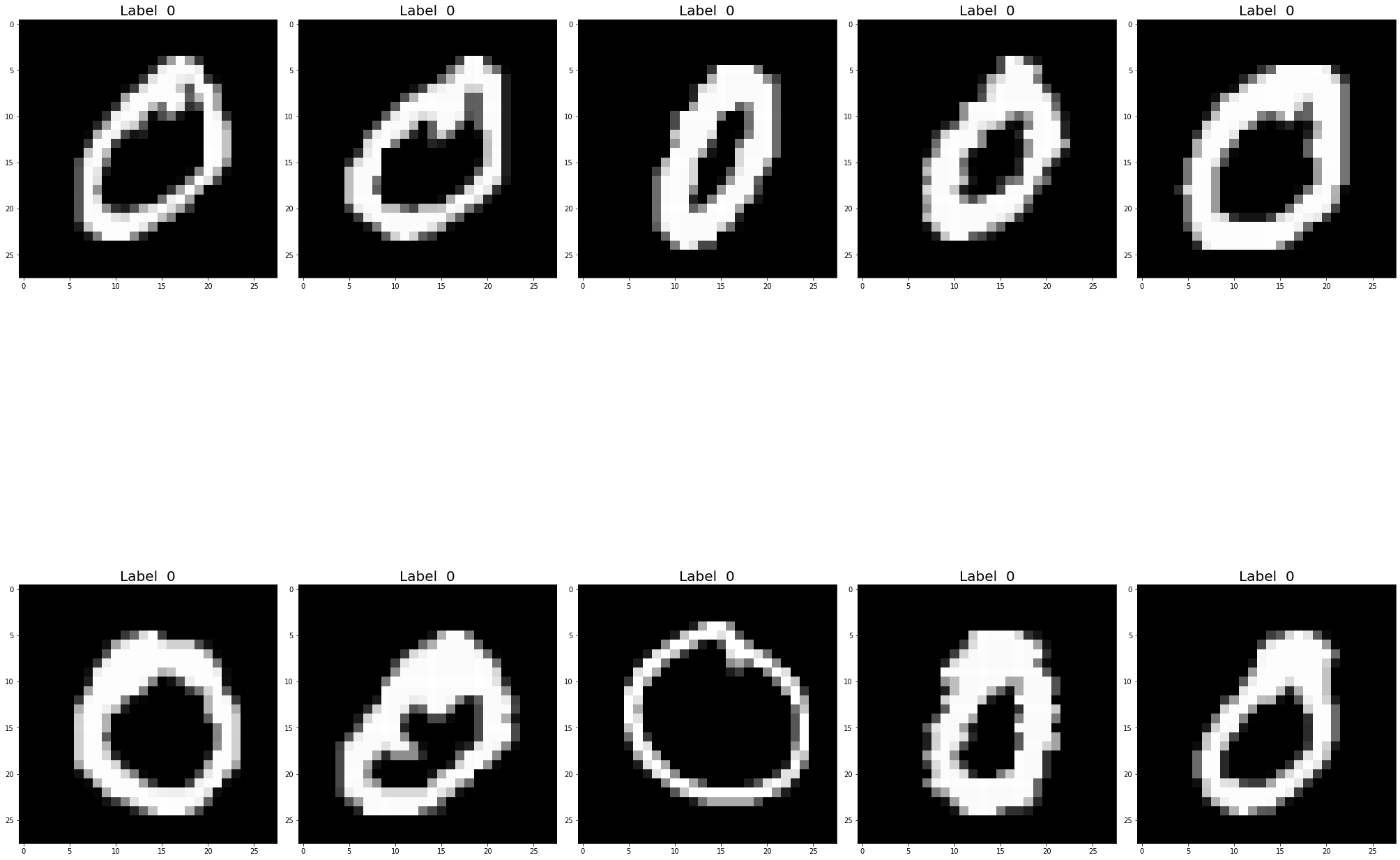
**Analysis**

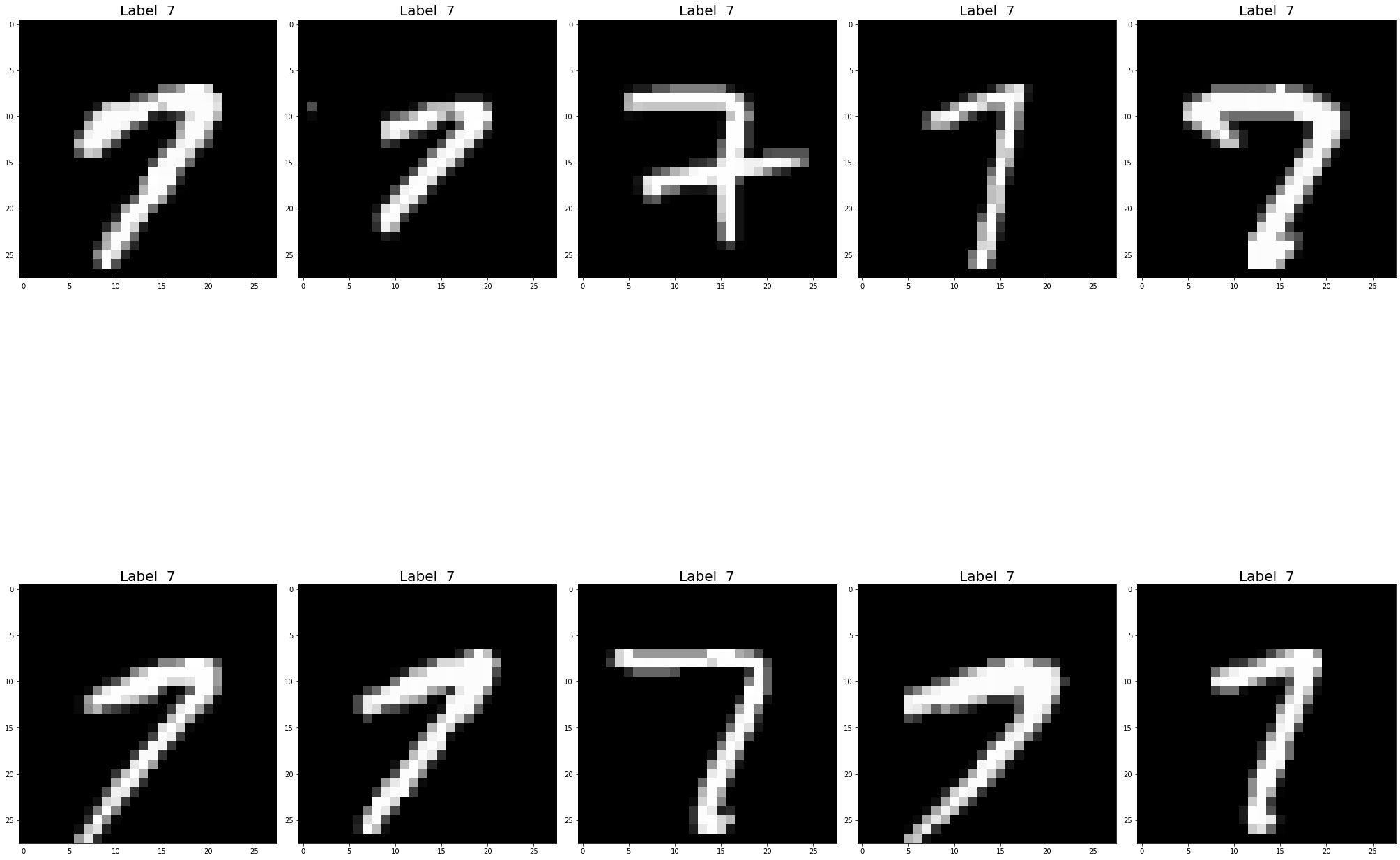
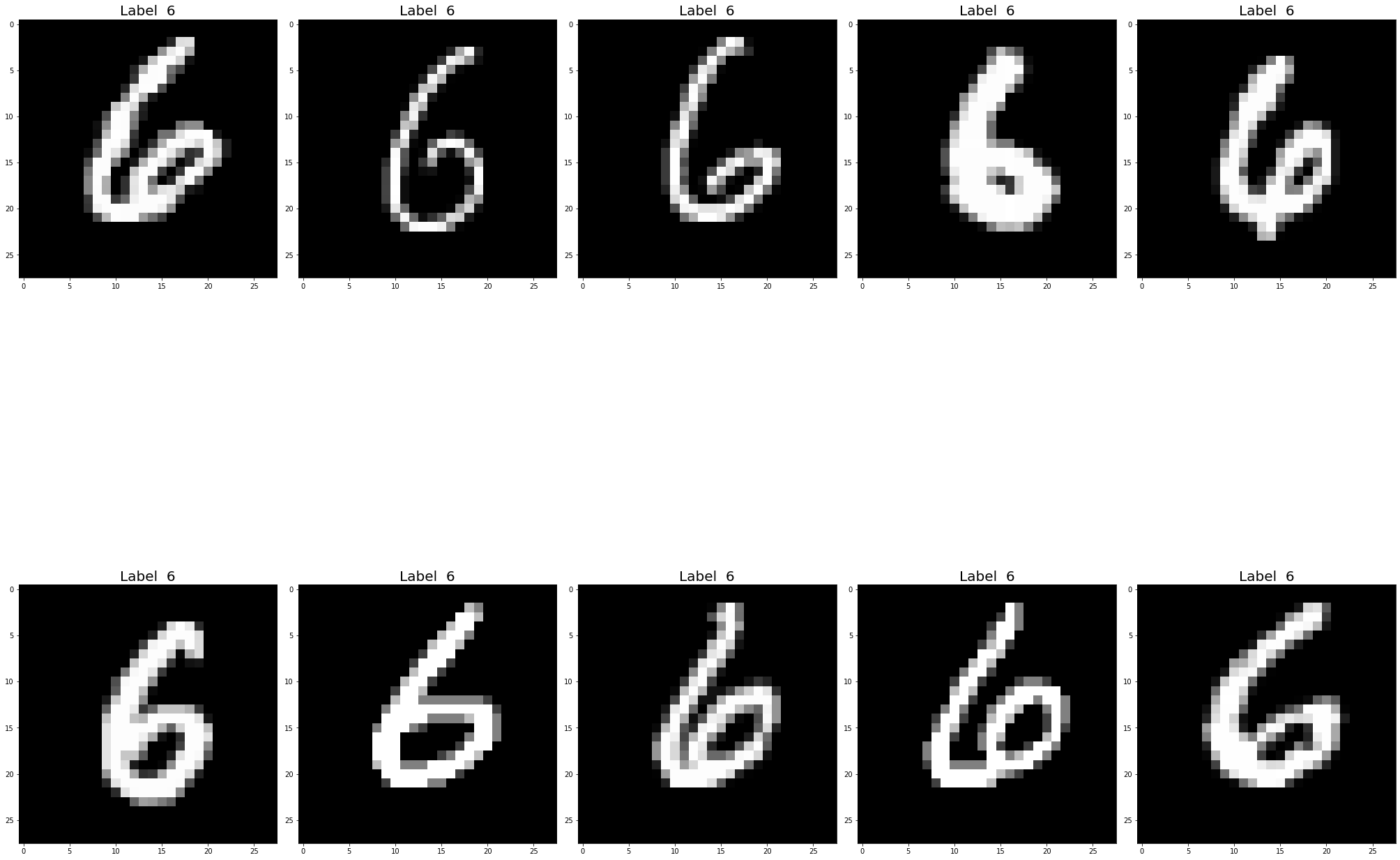
Question 1

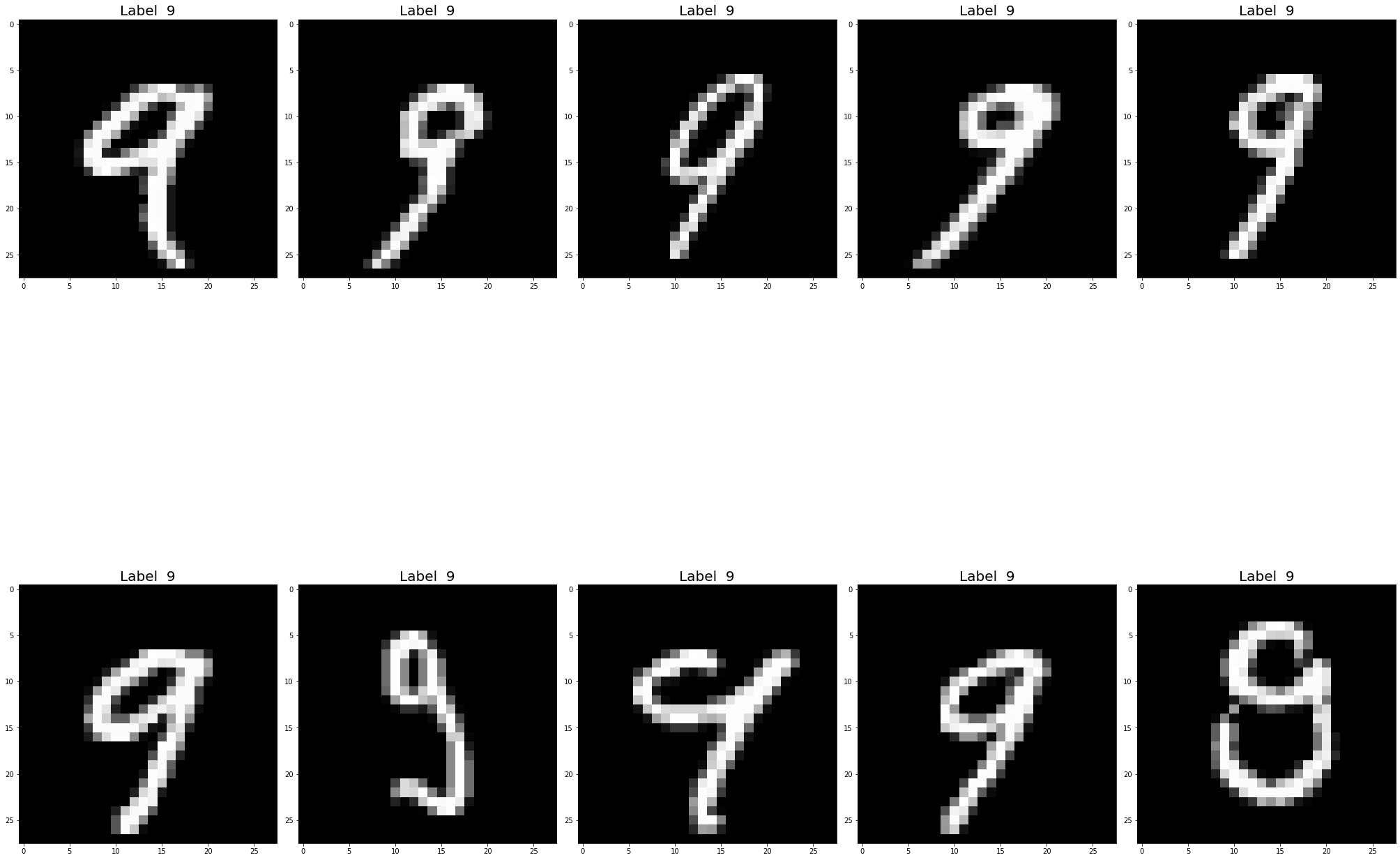
Part a

1. These are images of digits 0 to 9.
2. Each image is of size 28\*28 so each image is made up of 784 pixels
3. The images are in one channel giving it a range of colours between white and black colour.
4. The higher the value the brighter the pixels and the lower the value the darker the pixels.
5. The darker regions are having value set to 0 acting like a background and the brighter pixels have value more than 0 giving them the structure of each digit.
6. The pixels vale lie in the range 0 to 1. With 0 being the darkest pixels value and 1 being the brightest pixels value.
7. Generally the grayscale image values range from 0 to 255 but, I think the values here are being normalises between 0 to 1.
8. Different samples of each class has been depicting different writing styles/representation of a digit.
9. There has been quite a bit of noisy values as the image does not seem sharp and clear.
10. There may be certain structures which are not that similar or in context is not representing the number but has been labelled as that digit for example 0 may be na o or 5 or s.
11. Class Distribution is almost same with class 2 occurring the most.
12. The number of pixels with no zero and hence brightened value in each image of a particular class is different, representing several ways to write digits or presence of some noisy data.
13. Differentiating between numbers 3 and 8 or 4 and 9 or 7 and 9 is challenging as they both have overlapping pixels regions.
14. With 1 and 0 it is easier as the middle pixels of 1 are not zero but for 0 digit its mould be 0. 6 and 9 is also easier because of there opposite brightened pixels location.
15. 2 follows quite unique pattern than other and will not have much overlapping pixels, so will be easy to classify.
16. There is a challenge in identifying numbers like 1s and 7s as these have different writing notation as in a line at the Botton of 1 or cut in middle of 7. 1 can written diagonally or straight or left or right. Similarly the variation of 2 and 4 be taken care. So these notations should also be taken care while classifying them.
17. If the round part of 9 and line part is small is quite big it may get classified as 0.
18. Also depending on the language we are dealing with Hindi or English or other language digits they might need to take other aspects in consideration as well, for example 8 might be 4 in Hindi, 2 in some image is similar to 2 in Hindi.
19. If 9 is not written as the end line touching it head it might look like 8 and get wrongly get classified, so that need to be taking place.
20. If 5 head line is small and the the bottom turn is very long, it may get classified wrongly as 6, so that has to be taken care.





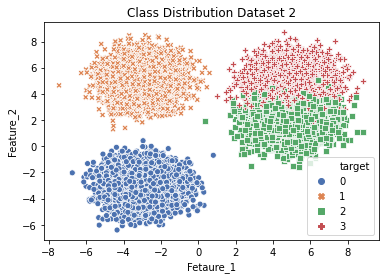
Part b





Q1 1.

1. There are 4 classes present in the given dataset.
2. All classes are equally distributed and hence the dataset is balanced.
3. I see few points in the dataset have properties very different from the rest of the members of its Classe and hence behaving as outliers. I see outlier in all the 4 classes. Presence of outlier can hamper the models which are sensitive to noise and can depreciate the accuracy
4. The classes 0 and 1 can be linearly separated if considered in isolation, similarly the classes 2 and 3.
5. But none of the classes are linearly separable from the rest of the classes. Hence Linear regression wont work in this example.
6. The classes 2 and 3 share a very close boundary with each other. This shows the feature set values of some datapoint of class 2 and 3 are quite same.
7. The decision tree boundary can very appropriately classify these datapoint.
8. The classes have a very distinct feature values and hence shows sharp boundaries except for few classes.
9. The datapoint of easy class are very densely populated clustered together hence representing very similar features. So basically the spread or variance of these points are low.
10. Each datapoint has 2 features describing them.
11. The range of one feature set for 0 and 1 classes are almost same and all are almost less than 0 . Similar the range of values for one of the feature for 2 and 3 classes are almost same and all almost greater than 0.
12. While class 0 has all negative values for the other feature but the class 1 has all positive value for the same feature set. The class 2 and 3 has mostly values positive for the other feature, although class 3 has some values less than 0. The other feature values for class 1 is similar to that of 2 and 3.
13. While none of the data points of class 0 and 1 clash into each other, so the chance of misclassification will be very minimal, hence the accuracy for classification of these two classes will be high. But in case of class 2 and 3 most of the points overlap into each other regions, so the chances of misclassification error could be high in this case hence accuracy would be effected.
14. The data points mid way to the boundary of the of the fours classes have high chance of miss classification into wrong class, if the model is not appropriate like the 0 class point into class 2 or class 2 point class 1.
15. The chances of class 0 elements to be classified into class 2 and vice versa is more due to their close boundary.
16. In all the classes the points close to the centre have more chances of getting classified correctly as compared to others as they are surrounded by data points of their own type, rather than the ones at the outer ends of the classes which are surrounded by other classes as well.
17. Learning feature set of class 0 and 1 and classifying them is quite easy as compared to learning feature set of class 2 and 3 as they overlap a lot.
18. In case of decision tree within 3 splits we can maximally classify the datapoint into each classes, although some error would be there due to overlap of class 2 and 3 and due to the outliers. So the depth would be 3 if we don’t want our tree to overfit.
19. The splitting could be first horizontally at 1 around along Feature\_1.Two regions would be created with 0 and 1 data points almost and 2 and 3 data points almost. The region with 1 and 0 classes can be split at feature 2 at value
20. But if want no outliers/all pure nodes then we have to go ahead and split further. Then the depth of tree would increase more than 3.

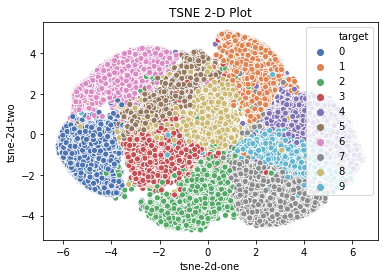


Q1 2.

Part c

1. The TSNE plot reveals all the 10 classes of the dataset quite appropriately.
2. The component selected by TSNE plot has great deal of information content cable enough to put the data points of same classes clustered together.
3. The classes can have decision boundaries although there would be certain data points not lying within own class boundaries. Hence the accuracy would be affected.
4. There are certain classes data points which are into the clustered region fo other data points. Hence separating them would be a challenge.
5. The classes 6, 0,2 ,4will have comparatively less misclassification euros as compares to other classes, as their data points seem to more clustered together.
6. 1 class will probably suffer more misclassification error on 2 feature set, as it shares overlapping boundaries with classes5,9,8. Similarly for Classe like 7,3 etc.
7. There are a lot of outliers in terms of 2 features set from every classes as the data points seems very much scattered all over.
8. Due to unclear boundaries between the classes the linear separation will not give good accuracy . The classification will be a challenge in this case.
9. For feature set values as pixels several classes may satisfy the test point and hence misclassification error could be more and accuracy will be decreased.
10. The feature set can be considered a better representation for certain classes like 6,4,0,2,3,8 as they seem more clustered and together.
11. Visualising all the points in the 2D space is quite challenging as most of the points are overlapping due to improper representation by the 2 feature set.

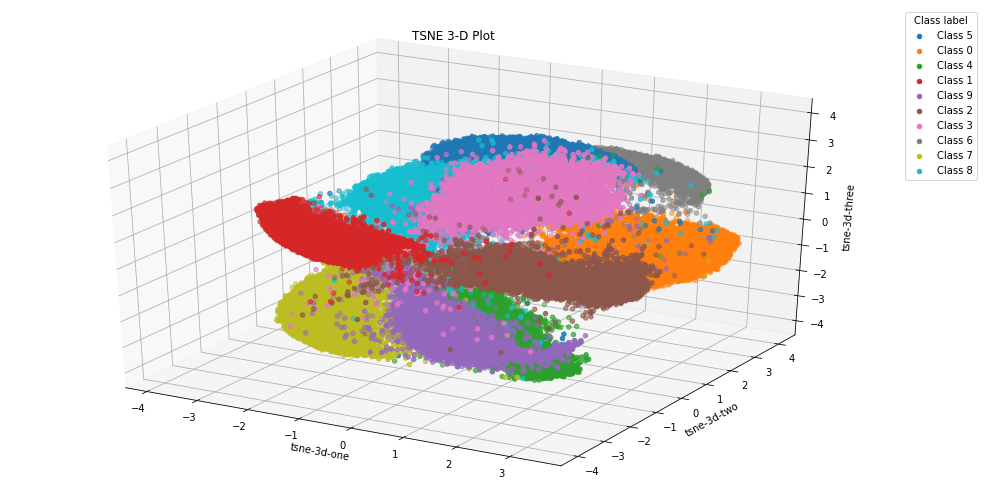
1. As discussed earlier the overlapping of the classes 4 and 9 which can also be visualised from here. Similarly the overlapping of 9 and 7 mentioned earlier can be seen appropriately.
2. 7 and 1 could also overlap due to missing top of 7 which can make it seem like 1 or as most structure of 7 is in 1 as well.
3. 0 and 6 and 2 and 8 and 3 do not have much overlapping nature in general and hence seems to out-stand than all the classes.
4. Some 5 can be miss written close to 6 structure hence overlapping with the 6 feature set.



Q1 3.

Part d

1. Below is the 3-D plot of the dataset 1:



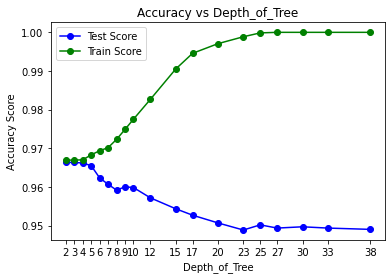
Q3 4.

1. Yes, there exists a distinctiveness between the results of part c and d.
2. In this now with three dimension the classes seem to be well separated and clustered.
3. This could help the decision tree classifiers to have a very good boundary separating the the classes.
4. Although there are still cases of overlapping which can lead to misclassification of the points.
5. The earlier overlap shown in pixels value of 4 and 9 or 7 and 1 can be easily depicted with the closeness of the the data points of the mentioned classes.

Question 2

Part a

1. The Decision Tree was trained on several depth [1,2,3,4,5,6,7,8,9,10,12,15,17,20,23,25,27,30,33,38] to see the difference .
2. Below are the results :



Q 2 1.

3. With the increase in the depth the as we see the accuracy seems to be improving in terms of training accuracy.

4. Whereas for the test accuracy the same increase and then decreases with the depth.

5. This depicts the undercutting and overfitting of the decision tress.

6. With starting at depth 1, we see the accuracy is 50 % , and the moment we move onto depth 2 that has increase considerably to 90 % in terms of both train and test accuracy.

7. To support the above fact we could look at the scatter plot shared earlier when classes 0 and 1 and 2 and 3 can mostly separated out with splits made twice.

8. For instance along feature 1 at 1 and the next split along feature 2 at 1. These two split alone can reach us to a very good accuracy value in both train and test terms.

9. This is so because the class seems most separated and hence in few splits the classification could be done appropriately.

10. Although the classes 2 and 3 has a lot of points overlapping which can be reason of not close to 100 accuracy.

11. So Whatever was visioned in the 1(b) part has been visioned her as well.

12. Moreover we could see that with increase in depth the train accuracy seem increasing and reaching 100 %. However the test accuracy seems decreasing.

13. This can be because of the overlapping points or the outliers which we could see in the scatter plot itself.

14. As the tree grow it tries to fit the training data and hence the new test data does not perform well.

15. The best depth could be 5 where the similarity between the score of train and test accuracy is quite good.

Part b

1. Below is the table obtained for train and test accuracy using my implemented accuracy function:

|  |  |  |  |
| --- | --- | --- | --- |
| Depth\_Of\_Tree | Train\_Accuracy | Test\_Accuracy | Stauts |
| 1 | 0.50221 | 0.49483 | Underfitting |
| 2 | 0.96693 | 0.96633 | Good fit |
| 3 | 0.96693 | 0.96633 | Good fit |
| 4 | 0.967 | 0.96633 | Good fit |
| 5 | 0.96829 | 0.96567 | Good fit |
| 6 | 0.96936 | 0.9625 | Good fit |
| 7 | 0.97014 | 0.96067 | Start to overfit |
| 8 | 0.97229 | 0.95917 | Start to overfit |
| 9 | 0.97486 | 0.96033 | Start to overfit |
| 10 | 0.9775 | 0.95967 | Start to overfit |
| 12 | 0.98257 | 0.95717 | Start to overfit |
| 15 | 0.99064 | 0.95283 | Overfitting |
| 17 | 0.9945 | 0.951 | Overfitting |
| 20 | 0.99707 | 0.95083 | Overfitting |
| 23 | 0.99886 | 0.9505 | Overfitting |
| 25 | 0.99986 | 0.95017 | Overfitting |
| 27 | 1 | 0.94933 | Overfitting |
| 30 | 1 | 0.948 | Overfitting |
| 33 | 1 | 0.9495 | Overfitting |
| 38 | 1 | 0.94883 | Overfitting |

2. At the start the at depth 1 the model was quite undercutting well seen form the low training and testing errors .

3. From the depth 2, the training and testing accuracy both increased a lot in comparison to depth 1. In fact the two accuracy are quite similar to each other and hence represents a good fit. This is same for depth like 3,4,5,6.

4. After the 6 depth from 7 depth we see a lot of deviation from the training and testing accuracy which keeps on increasing with the increase in depth.

5. Although the change is very minimal but still from the graph the deviation if quite visible which is a representation of overfitting.

6. Th essence of all can be explained through the dataset itself. In the scatter plot itself as we say the classes 0 and 1 data points are quite separable and in one depth classes 0,1 and 2 , 3 can be separated. And the in next step 0 and 1 can be separated and classes 2 and 3 can be separated . So since by depth 2 most of data points of all the classes are classified correctly.

7. The accuracy thus goes up considerably when we reach to a depth of 2.

8. With further increase in the depth from 3 or 4 or 5 or 6 there would be certain data points of classes 2 and 3 overlapping, which need to be classified properly. In a way at the depth 2 also the nodes would be pure completely mostly because of the overall between class 2 and class 3.

9. Hence with increase in depth further the node get pure and better accuracy results in them of training accuracy.

10. Although not much increase in the test accuracy could see, but training accuracy has increase as now the nodes are tried to be broken into pure nodes.

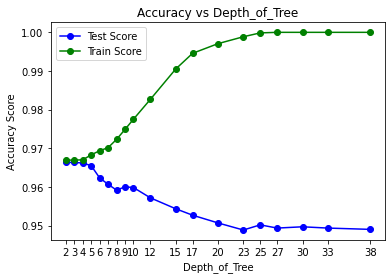
11. AS we keep on increasing the depth, the decision tree has started to learn training data points when it tries to split the node further trying to get the pure nodes.

12. This results in increase in training accuracy considerably and it reaches to 1 over the increase in depth. The data points has started to split up into leaves thus making Decision tree to learn the data. The test accuracy has been going down owing to the fact that the Decision tree structure is mainly build up to correctly classifying the training data and so on unseen data it fails.

13. The best depth could be 5.

Part c

1. There has been non change in the value of accuracy calculated by me or by the accuracy score implementation of sklearn.
2. Below are the results:



Q 2 3.

3. The table below:

|  |  |  |
| --- | --- | --- |
| Depth\_Of\_Tree | Train\_Accuracy | Test\_Accuracy |
| 1 | 0.50221 | 0.49483 |
| 2 | 0.96693 | 0.96633 |
| 3 | 0.96693 | 0.96633 |
| 4 | 0.967 | 0.96633 |
| 5 | 0.96829 | 0.96567 |
| 6 | 0.96936 | 0.9625 |
| 7 | 0.97014 | 0.96067 |
| 8 | 0.97229 | 0.95917 |
| 9 | 0.97486 | 0.96033 |
| 10 | 0.9775 | 0.95967 |
| 12 | 0.98257 | 0.95717 |
| 15 | 0.99064 | 0.95283 |
| 17 | 0.9945 | 0.951 |
| 20 | 0.99707 | 0.95083 |
| 23 | 0.99886 | 0.9505 |
| 25 | 0.99986 | 0.95017 |
| 27 | 1 | 0.94933 |
| 30 | 1 | 0.948 |
| 33 | 1 | 0.9495 |
| 38 | 1 | 0.94883 |

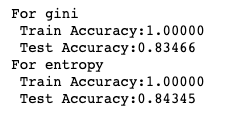
Question 3

Prepossessing:

1. The NULL values are present only on the column **pm2.5** and the same has been removed using the **back fill** method. This has been used because the pm2.5 value of today will be quite similar to the previous or future case in majority of the times. As the first values itself was missing so didi not use **first fill** instead used **back fill**.
2. The dataset has been split into 80: 20 ratio with training set with **35060** data points and testing set with **8764** data point using my **train\_test\_split** method.

Part a

1. I have run multiple times to check the accuracy is best at mini or at entropy. Found multiple time the **entropy** criteria gave the best test accuracy.



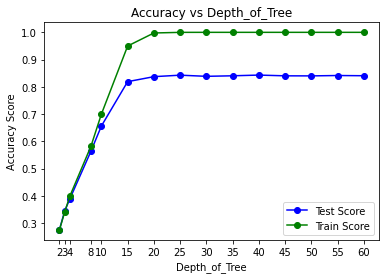
Q3. 1

2. Although the computational power of Gini is better than Entropy as Entropy involves log calculation and has to util 1 as compare to Gini which just have to go util 0.5, Entropy is better in this case with just minimal difference.

3. This might be the case because as there are 12 classes not all balanced, the weightage given by Gini for lower probability classes will be low as majority probability class is governing the Gini values due to squaring probability concept, but in case of Entropy the lower probabilities values also get some weight due to log multiplications and hence are providing better information gain.

Part b

1. The plots and results are below:



Q3 2.

|  |  |  |
| --- | --- | --- |
| Depth\_Of\_Tree | Train\_Accuracy | Test\_Accuracy |
| 2 | 0.27547 | 0.27681 |
| 3 | 0.34087 | 0.34493 |
| 4 | 0.40029 | 0.39012 |
| 8 | 0.58206 | 0.56401 |
| 10 | 0.70029 | 0.65666 |
| 15 | 0.94986 | 0.81926 |
| 20 | 0.99763 | 0.83752 |
| 25 | 0.99991 | 0.84299 |
| 30 | 1 | 0.83877 |
| 35 | 1 | 0.84071 |
| 40 | 1 | 0.84334 |
| 45 | 1 | 0.8406 |
| 50 | 1 | 0.84037 |
| 55 | 1 | 0.84174 |
| 60 | 1 | 0.84094 |

2. As we can see from the above plots and details the with the small depth like 2, 4, 8 the training and testing accuracy both are very low around 50 %. Owing to the fact that the model is still not trained properly and is under-fitting. Basically the model has not learned better and therefore need to grow more.

6. With increase in depth, however the model start learning and increasing both it training and test accuracy with reaching unto 60 % on increasing depth to 10 and further to 80% on reaching a depth of 15. As the tree grows more and more there is significant information gain and hence the accuracy keeps improving.

7. However, with increase in dept further after 15 there not much significant increase in the test accuracy of the model as with depth around 20 accuracy just increased by 2% to 3% and further on increasing to around 30 only in points accuracy seems to increase.

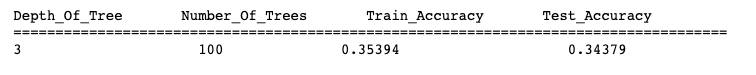
8. Above 30 if we go the accuracy now seems decreasing and the training accuracy is 100& which is owing to the fact that the tree has remembered the entire training set. And also it can be observed that the test accuracy has started to detoriate due to overfitting.

9. Owing to the above all facts, the best max\_depth could be somewhere between 15 to 20 if we want to accommodate the 2 to 3% rise in test accuracy. Since above 20 not much significant change has been seen in the test accuracy until 30 and also training data seems to be learnt with 100 % accuracy, will go with the **depth of 15 as the best depth**.

10. This is just to avoid over optimisations to my model by reaching 100% of trainings accuracy for just 2% to 3% rise in test accuracy. If we accept something like this then we might end up not having better accuracy for new set of test data.

Part c

1. Below are the results of ensemble on test and training dataset.



Q3 4.

2. We could clearly see that the test accuracy has improved although with very little amount.

3. Ensemble techniques de la in with lot of random in the data hence it performs better than a single decision tree.

4. The random in train data for each decision tree in ensemble helps each tree learn differently on different datasets.

5. This helps them to behave differently for new test data and hence they perform better.

6. Whereas in case of normal single decision tree, the tree learn on all train data and tries to fit them.

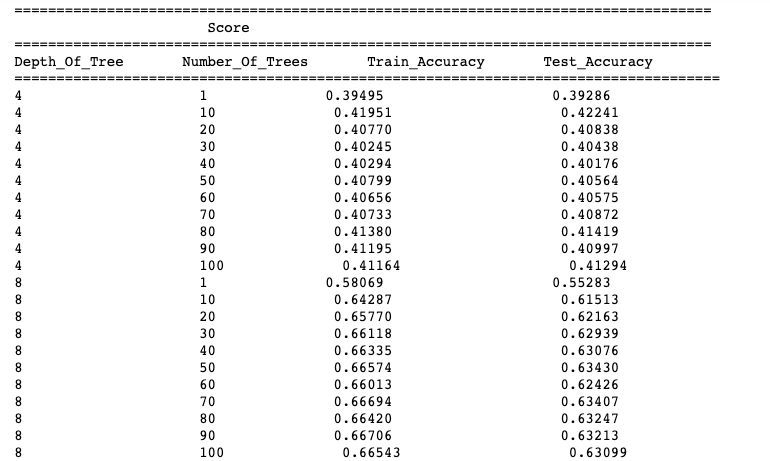
7. The training error would be greatly reduce in case of decision tree because one tree learns the entire training data.

8. However, the ensemble would not have that improvement in case of training accuracy because it learns from the subset of training dataset.

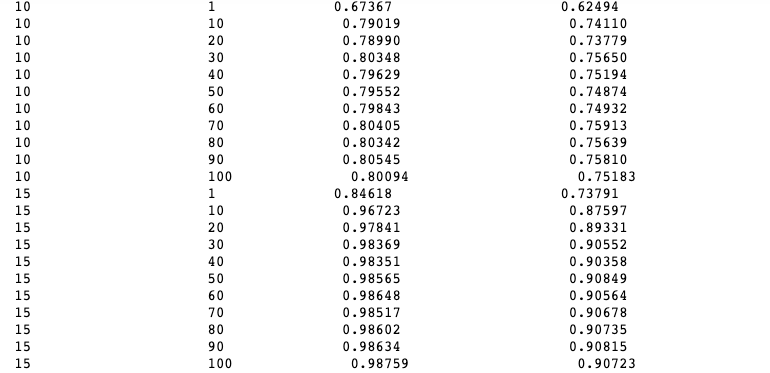
9. In a nutshell , when we need less variance then we should go for ensemble of decision tree and when we want less of bias we should go for single designs tree.

Part d

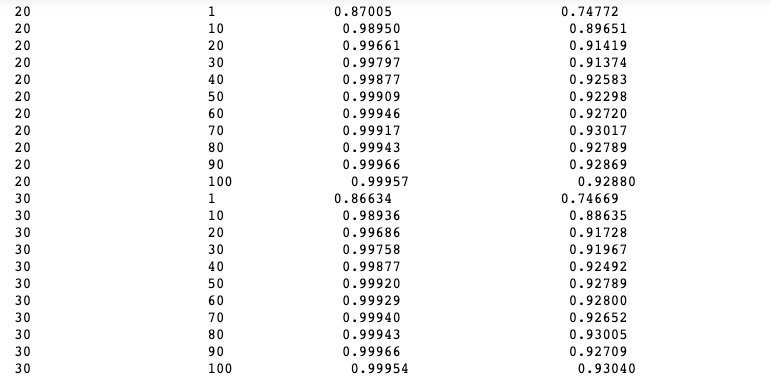
1. Below are the results of ensemble on test and training dataset for number of trees as hyper parameter.



Q 3 5



Q 3 6



Q 3 7

2. If we compare depth wise , we see that with increase in num of tress the train and test accuracy both has increased considerably.

3.