General description of my subtask2 code

My code for subtask 2 utilizes two functions which are described as follows:

1. createModelsAndEvaluate takes in all the lists of hyper parameters, loops through each to produce all combinations of parameters (270 combinations), produces a model from that combination, evaluates the accuracy of that model, and then adds those particular parameters and their associated accuracy, as a list, to a list of lists called “rows”. Once all combinations and their associated accuracies have been added to “rows”, “rows” is returned.
2. hyperparameter_ranking takes in the list of lists called “rows” produced by createModelsAndEvaluate, produces a list of column names, produces a pandas data frame from the rows and then sorts all the rows in descending order based on the “Accuracies” column. As the data frame is sorted the indexes of each row is reassigned to be the descending ordered position (row 0 has the highest accuracy, etc).

These two were separated so that if there are changes that need to be made to the sorting, they can be made without having to reproduce and reevaluate every model.

When createModelsAndEvaluate is called, the result is also immediately saved to a .csv so that the results can be loaded later, without having to reproduce and reevaluate every model.

Provide a screenshot of the first five rows and last 5 rows of the panda's data frame table that you outputted from subtask2. Summarize your findings in 2-3 sentences. Interpret the content of the generated data frame with respect to settings of the five parameters that are associated with higher accuracy (e.g. Do 4-layer neural networks have higher accuracies than 2/3-layer networks?). Discuss parameter settings that do not have much impact on accuracy (e.g. different chosen batch size parameters settings may not seem to have much impact on accuracy). Note that for some hyperparameters the reported findings might be inconclusive; in this case, state if that's inconclusive

	LAYERS	UNITS	BATCH_SIZE	LEARNING_RATE	EPOCHS	Accuracy
0	2.0	64.0	8.0	0.001	30.0	0.932833
1	2.0	64.0	32.0	0.010	30.0	0.927750
2	2.0	64.0	32.0	0.010	20.0	0.923583
3	2.0	64.0	8.0	0.001	20.0	0.922917
4	2.0	64.0	128.0	0.010	30.0	0.922083

	LAYERS	UNITS	BATCH_SIZE	LEARNING_RATE	EPOCHS	Accuracy
265	3.0	64.0	128.0	0.00010	10.0	0.093167
266	3.0	64.0	32.0	0.00001	30.0	0.091000
267	3.0	64.0	8.0	0.00001	10.0	0.087083
268	3.0	64.0	128.0	0.00010	20.0	0.085333
269	4.0	64.0	128.0	0.00010	20.0	0.060667

Low layer counts, and moderate learning rates seemed to have beneficial effects on accuracies. Some parameters had little effect on accuracy: both the best 5 and worst 5 models had all three possible batch sizes.
Some parameters had inconclusive effects on accuracy: both the best 5 and worst 5 models had unit counts all equal to 64. Additionally, all 5 of the best models had epoch counts of 20 or 30, while the worst 5 models had all 3 possible unit counts - seemingly suggesting that higher epoch counts may be beneficial for accuracy - but again, this is inconclusive.