COL 774: Machine Learning
Assignment 3

Name: Suraj Reddy
Entry Number: 2025AIY7586

1 Decision Trees (and Random Forests)

Entropy:

$$H(Y) = -\sum_{i=1}^{k} P(y_i) \log_2 P(y_i)$$

where $P(y_i)$ is the probability of class y_i in the target variable Y.
Conditional Entropy:

$$H(Y|X) = \sum_{v \in \text{Values}(X)} P(X = v) \, H(Y|X = v)$$

Information Gain :

$$IG(Y, X) = H(Y) - H(Y|X)$$

Given:

- Max depths=[5,10,15,20]
- train, test and validation csv files

Table 1: Decision Tree Performance at Different Depths

Max Depth	Train Accuracy	Test Accuracy	Validation Accuracy
5	0.8839	0.5884	0.5828
10	0.9994	0.5905	0.5839
15	1.0000	0.5905	0.5839
20	1.0000	0.5905	0.5839

Comments

- As the tree depth increases, training accuracy reaches 1.0, indicating overfitting.
- However, test and validation accuracies remain around 0.59 and 0.58, showing poor generalization.
- The best performance is achieved at a **maximum depth of 10**,

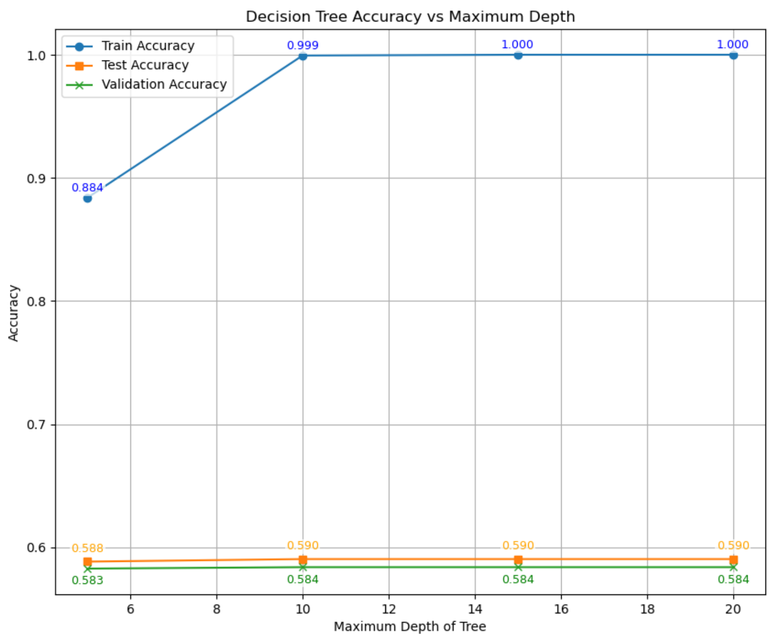


Figure 1: Accuracy vs max depth

2 Decision Tree One Hot Encoding

Given:

- **Categorical attributes:** ['team', 'opp', 'host', 'month']
- **Continuous attributes:** ['year', 'fow', 'score', 'rpo', 'toss', 'day_ match', 'bat_first', 'format']
- max depth: [15,25,35,45]
- performing one hot encoding on categorical attributes.

results:

Max Depth	Train Accuracy	Test Accuracy	Validation Accuracy
15	0.7109	0.5522	0.5724
25	0.8496	0.5812	0.6046
35	0.9221	0.5843	0.5805
45	0.9872	0.6122	0.5816

Table 2: Decision Tree performance for different maximum depths

Table 3: Comparison of Decision Tree Models (Q1 vs Q2)

Model	Best Depth	Train Acc	Test Acc	Validation Acc
Q1 (Without One-Hot)	5	0.9994	0.5905	0.5839
Q2 (With One-Hot)	25	0.8496	0.5688	0.6046

comments:

- In part Q1, the model achieved higher training accuracy with a shallow tree, suggesting it could effectively capture categorical patterns without additional encoding.
- In part Q2, one-hot encoding increased the feature dimensionality, requiring a deeper tree (depth 25) to fit the data.
- Although validation accuracy improved slightly (0.5839 $\rightarrow$ 0.6046), test accuracy decreased, indicating minor overfitting and limited generalization gain.
- Execution time for Q2 was higher due to increased input dimensionality from one-hot encoding.

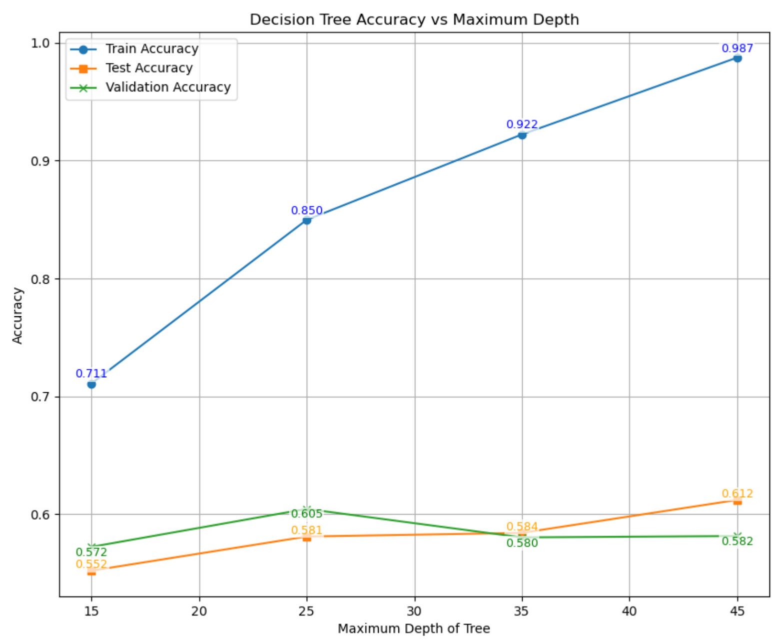


Figure 2: Accuracy vs max depth

3 Decision Tree Post Pruning

Given:

- **Categorical attributes:** ['team', 'opp', 'host', 'month']
- **Continuous attributes:** ['year', 'fow', 'score', 'rpo', 'toss', 'day_ match', 'bat_first', 'format']
- max depth: [15,25,35,45]
- performing one hot encoding on categorical attributes.
- Use entropy as criterion.

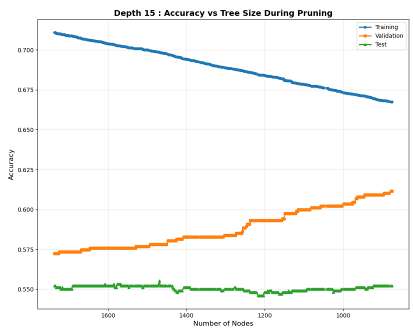
Results:

Table 4: Decision Tree Pruning Results for Different Max Depths

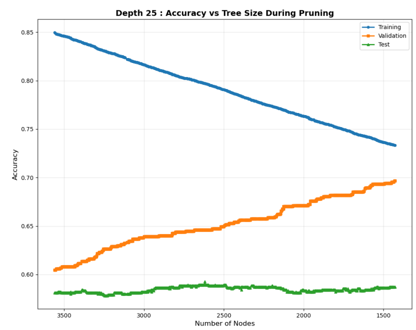
Max Depth	Before Acc	After Acc	Initial Nodes	Final Nodes	Pruned
15	0.5724	0.6115	1737	875	862
25	0.6046	0.6966	3561	1429	2132
35	0.5805	0.6989	4593	1849	2744
45	0.5816	0.7184	5641	2139	3502

Comments:

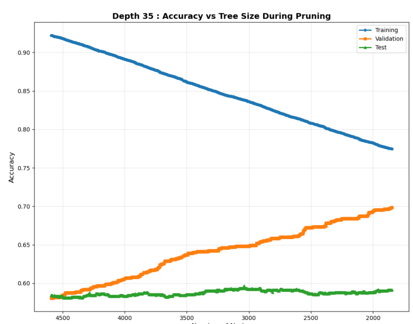
- Pruning improved validation accuracy across all tree depths,
- Indicating effective removal of overfitting branches.
- As depth increased, the number of nodes and nodes pruned also rose significantly.
- The best post-pruning accuracy was achieved at a max depth of 45.



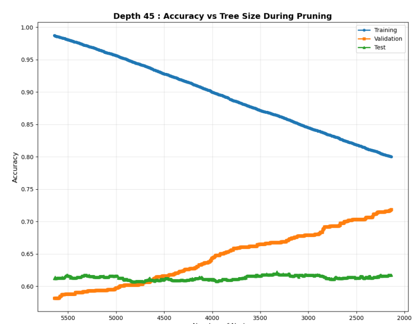
(a) Max Depth = 15



(b) Max Depth = 25



(c) Max Depth = 35



(d) Max Depth = 45

Figure 3: Training, validation, and test set accuracies against the number of nodes while successively pruning for different maximum depths.

4 Decision Tree Post Pruning with Gini Impurity

Given:

- **Categorical attributes:** ['team', 'opp', 'host', 'month']
- **Continuous attributes:** ['year', 'fow', 'score', 'rpo', 'toss', 'day_ match', 'bat_first', 'format']
- max depth: [15,25,35,45]
- performing one hot encoding on categorical attributes.
- Use gini index as criterion.

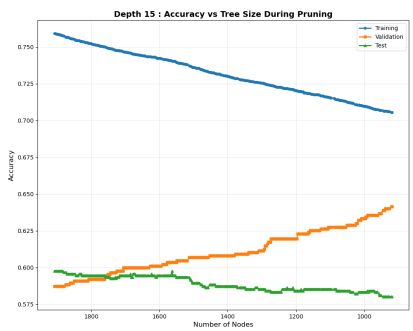
Results:

Table 5: Decision Tree Pruning Results for Different Max Depths using Gini Impurity

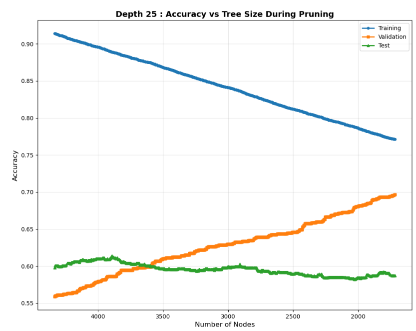
Max Depth	Before Acc	After Acc	Initial Nodes	Final Nodes	Pruned
15	0.5874	0.6414	1907	919	988
25	0.5586	0.6966	4335	1717	2618
35	0.5598	0.7046	5249	1953	3296
45	0.5598	0.7126	5705	2093	3612

Comments:

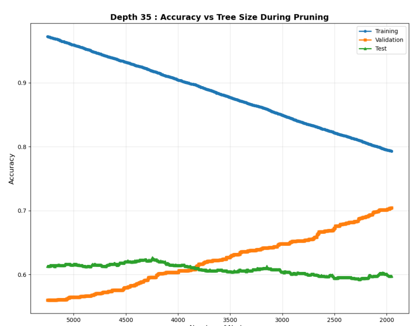
- Using gini Impurity, didn't impove much of the test accuracies after pruning.
- Pruning improved validation accuracy across all tree depths,
- Indicating effective removal of overfitting branches.
- As depth increased, the number of nodes and nodes pruned also rose significantly.
- The best post-pruning accuracy was achieved at a max depth of 45.



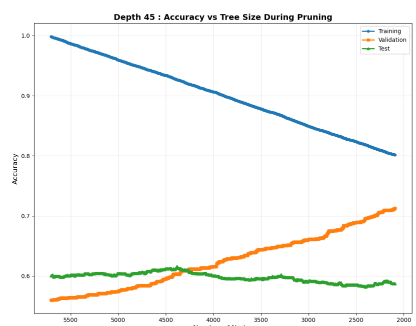
(a) Max Depth = 15



(b) Max Depth = 25



(c) Max Depth = 35



(d) Max Depth = 45

Figure 4: Training, validation, and test set accuracies against the number of nodes while successively pruning for different maximum depths.

5 Decision Tree sci-kit learn

Given: part (a)

- Use Entropy as criterion.
- max depth : [15,25,35,45]

Results:

Table 6: Decision Tree Performance for Different Max Depths

Max Depth	Train Accuracy	Test Accuracy	Validation Accuracy
15	0.8262	0.6329	0.6172
25	0.9845	0.6174	0.6092
35	0.9990	0.6163	0.6115
45	1.0000	0.6163	0.6230

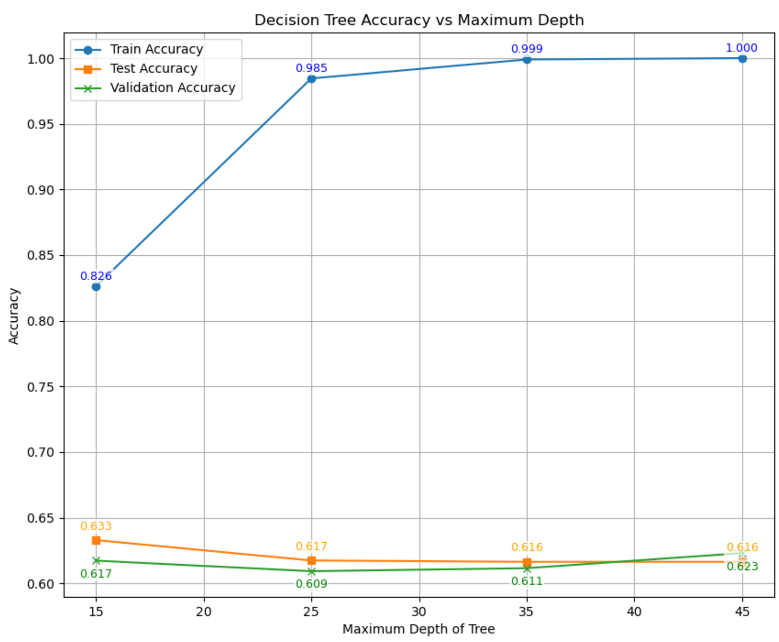


Figure 5: training, validation and test set accuracies against max depth

Given: part (b)

- Use Entropy as criterion.

- max depth: default value
- ccp alpha : [0.0,0.0001,0.0003,0.0005]

Results:

Table 7: Decision Tree Performance for Different Pruning Parameters

Pruning Parameter	Train Accuracy	Test Accuracy	Validation Accuracy
0.0	1.0000	0.6163	0.6230
0.0001	1.0000	0.6163	0.6230
0.0003	0.9782	0.6184	0.6218
0.0005	0.8635	0.6008	0.6414

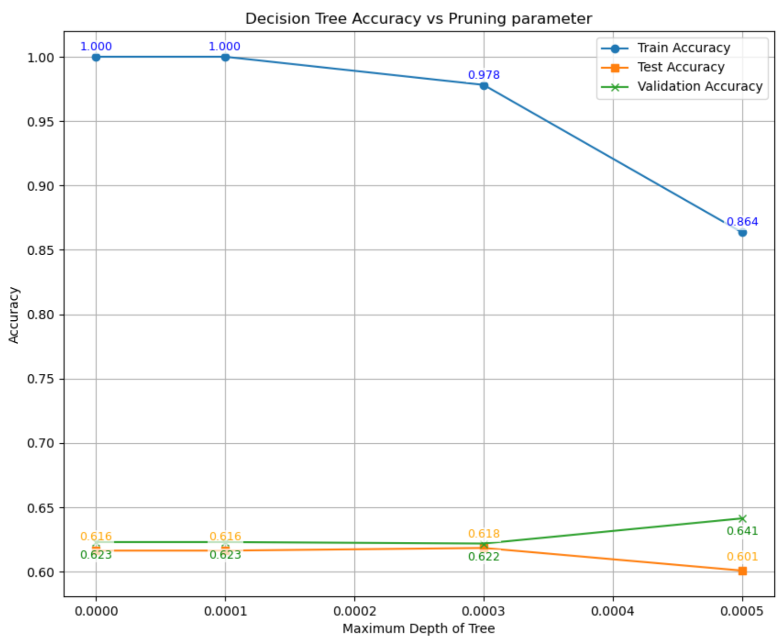


Figure 6: training, validation and test set accuracies vs ccp alphas

conclusion: Model with max depth 45 with val acc 0.6230 and model with cpp alpha 0.0005 with val acc **0.6413**.

comparision of best models with b and c:

Comment:

- Models from parts (b) and (c) achieved the highest validation accuracies after pruning.

Table 8: Comparison of Best Decision Tree Models from Parts (b), (c)

Part	**Criterion / Param**	**Max Depth**	**Train Acc**	**Test Acc**	**Val Acc**
(b)	Entropy+one-hot	25	0.8496	0.5688	0.6046
(c)	Entropy+Pruning	45	-	-	0.7184
(i)	Entropy(max_depth)	45	1.0000	0.6163	0.6230
(ii)	Entropy + `ccp_alpha`=0.0005	45	0.8635	0.6008	0.6414

- Confirming that pruning significantly enhances generalization.
- Between the two, the entropy-based model performed slightly better from 0.7184 to 0.7126.
- Introducing a pruning parameter (`ccp_alpha`) in part (e) improved validation accuracy compared to the unpruned model, but still underperformed compared to the manually pruned models in (b) and (c).

6 Random Forests: sci-kit learn

Given:

- Use Entropy as criterion.
- n_estimators : [50,150,250,350]
- max_features : [0.1,0.3,0.5,0.7,0.9]
- min_samples_split : [2,4,6,8,10]
- perform gridsearch

Results:

Table 9: Random Forest Results with Optimal Hyperparameters

Parameter	**Value**
Bootstrap	True
Criterion	Entropy
Max Features	0.3
Min Samples Split	10
Number of Estimators	350
Train Accuracy	0.9645
Test Accuracy	0.7146
Validation Accuracy	0.7172
Out-of-Bag Accuracy	0.7258
Execution Time (s)	437.2835

Comparison with Decision Tree Models from Parts (c) and (d)

Table 10: Comparison of Random Forest with Best Decision Tree Models

Model	**Train Acc**	**Test Acc**	**Val Acc**	**Remarks**
DT(Part c, entropy + Pruning)	-	-	0.7184	Improved after pruning
DT(Part d, gini + Pruning)	-	-	0.7126	Improved after pruning
RF(Optimal Params)	0.9645	0.7146	0.7172	Best generalization

Comments:

- The Random Forest model achieved the highest validation (0.7172) and test (0.7184) accuracies among all models.
- with training accuracy (0.9645) which was slightly lower than that of the fully grown tree, this reduction indicates better generalization.
- The out-of-bag accuracy (0.7258) confirms that the model has strong performance and reduced overfitting due to ensemble averaging.

COL 774: Machine Learning Assignment 3

Name: Suraj Reddy
Entry Number: 2025AIY7586

1 Neural Networks

(b) Implementing a simple neural network with single hidden layer.

given:

- number of hidden layer units from the set 1, 5, 10, 50, 100.
- learning rate = 0.01
- mini-batch size = 32
- input dimension = 3072
- output dimension = 36
- nof epochs=300
- Stopping criterion used = max epochs=300 and change in loss less than 0.0001 with patience 5.

Results: For hidden units = 1

Metric	Precision	Recall	F1-Score
Train	0.0307	0.0718	0.0303

Table 1: Training Results

Metric	Precision	Recall	F1-Score
Test	0.0361	0.0744	0.0342

Table 2: Testing Results

Class	Precision	Recall	F1-Score	Support
0	0.00	0.00	0.00	600
1	0.05	0.09	0.06	600
2	0.08	0.01	0.02	600
3	0.00	0.00	0.00	600
4	0.04	0.03	0.03	600
5	0.06	0.07	0.07	600
6	0.03	0.01	0.02	600
7	0.04	0.07	0.05	600
8	0.08	0.89	0.15	600
9	0.02	0.04	0.03	600
10	0.04	0.02	0.03	600
11	0.06	0.05	0.06	600
12	0.01	0.01	0.01	600
13	0.11	0.95	0.19	600
14	0.00	0.00	0.00	600
15	0.00	0.00	0.00	600
16	0.06	0.04	0.05	600
17	0.06	0.03	0.03	600
18	0.04	0.02	0.03	600
19	0.00	0.00	0.00	600
20	0.06	0.10	0.08	600
21	0.00	0.00	0.00	600
22	0.04	0.03	0.03	600
23	0.00	0.00	0.00	600
24	0.00	0.00	0.00	600
25	0.05	0.04	0.04	600
26	0.00	0.00	0.00	600
27	0.00	0.00	0.00	600
28	0.05	0.03	0.04	600
29	0.00	0.00	0.00	600
30	0.00	0.00	0.00	600
31	0.00	0.00	0.00	600
32	0.00	0.00	0.00	600
33	0.00	0.00	0.00	600
34	0.07	0.03	0.04	600
35	0.05	0.04	0.04	600
Accuracy			**0.0718**	21600
Macro Avg	0.03	0.07	0.03	21600
Weighted Avg	0.03	0.07	0.03	21600

Table 3: Classification Report for Training Data with hidden units 1

Class	Precision	Recall	F1-Score	Support
0	0.00	0.00	0.00	300
1	0.07	0.12	0.09	300
2	0.05	0.01	0.01	300
3	0.00	0.00	0.00	300
4	0.03	0.02	0.02	300
5	0.05	0.05	0.05	300
6	0.03	0.02	0.02	300
7	0.04	0.09	0.06	300
8	0.08	0.85	0.14	300
9	0.02	0.04	0.03	300
10	0.05	0.03	0.04	300
11	0.07	0.06	0.07	300
12	0.04	0.02	0.02	300
13	0.11	0.94	0.19	300
14	0.00	0.00	0.00	300
15	0.00	0.00	0.00	300
16	0.07	0.05	0.06	300
17	0.05	0.02	0.03	300
18	0.07	0.04	0.05	300
19	0.00	0.00	0.00	300
20	0.08	0.13	0.10	300
21	0.00	0.00	0.00	300
22	0.08	0.05	0.06	300
23	0.00	0.00	0.00	300
24	0.00	0.00	0.00	300
25	0.06	0.06	0.06	300
26	0.00	0.00	0.00	300
27	0.00	0.00	0.00	300
28	0.07	0.03	0.04	300
29	0.00	0.00	0.00	300
30	0.00	0.00	0.00	300
31	0.00	0.00	0.00	300
32	0.00	0.00	0.00	300
33	0.07	0.01	0.02	300
34	0.06	0.02	0.03	300
35	0.06	0.04	0.05	300
Accuracy			**0.0744**	10800
Macro Avg	0.04	0.07	0.03	10800
Weighted Avg	0.04	0.07	0.03	10800

Table 4: Classification Report for Test Data with hidden units 1

Results: For hidden units = 5

Metric	Precision	Recall	F1-Score
Train	0.4076	0.4256	0.3986

Table 5: Training Results for hidden units 5

Metric	Precision	Recall	F1-Score
Test	0.3673	0.3858	0.3584

Table 6: Testing Results for hidden units 5

Class	Precision	Recall	F1-Score	Support
0	0.65	0.69	0.67	600
1	0.27	0.09	0.14	600
2	0.41	0.60	0.48	600
3	0.31	0.38	0.34	600
4	0.23	0.10	0.13	600
5	0.62	0.61	0.62	600
6	0.38	0.26	0.30	600
7	0.52	0.64	0.57	600
8	0.58	0.61	0.60	600
9	0.49	0.77	0.60	600
10	0.52	0.80	0.63	600
11	0.52	0.36	0.43	600
12	0.38	0.47	0.42	600
13	0.51	0.75	0.61	600
14	0.46	0.37	0.41	600
15	0.47	0.63	0.54	600
16	0.36	0.30	0.33	600
17	0.38	0.36	0.37	600
18	0.41	0.55	0.47	600
19	0.38	0.20	0.27	600
20	0.39	0.27	0.32	600
21	0.64	0.80	0.71	600
22	0.42	0.49	0.45	600
23	0.41	0.65	0.50	600
24	0.27	0.08	0.12	600
25	0.20	0.06	0.09	600
26	0.47	0.45	0.46	600
27	0.43	0.28	0.34	600
28	0.40	0.12	0.18	600
29	0.27	0.28	0.28	600
30	0.34	0.58	0.43	600
31	0.28	0.21	0.24	600
32	0.34	0.25	0.29	600
33	0.33	0.56	0.41	600
34	0.29	0.12	0.17	600
35	0.35	0.58	0.43	600
Accuracy			**0.4256**	21600
Macro Avg	0.41	0.43	0.40	21600
Weighted Avg	0.41	0.43	0.40	21600

Table 7: Classification Report for Training Data for hidden units 5

Class	Precision	Recall	F1-Score	Support
0	0.62	0.65	0.64	300
1	0.16	0.06	0.08	300
2	0.37	0.56	0.45	300
3	0.30	0.38	0.33	300
4	0.22	0.09	0.13	300
5	0.53	0.53	0.53	300
6	0.40	0.29	0.34	300
7	0.48	0.62	0.54	300
8	0.48	0.48	0.48	300
9	0.46	0.72	0.56	300
10	0.52	0.77	0.62	300
11	0.49	0.31	0.38	300
12	0.32	0.42	0.36	300
13	0.45	0.74	0.56	300
14	0.38	0.28	0.32	300
15	0.40	0.63	0.49	300
16	0.32	0.26	0.28	300
17	0.34	0.31	0.33	300
18	0.36	0.48	0.41	300
19	0.28	0.13	0.17	300
20	0.35	0.20	0.26	300
21	0.61	0.76	0.68	300
22	0.39	0.46	0.42	300
23	0.38	0.62	0.47	300
24	0.24	0.06	0.09	300
25	0.26	0.07	0.11	300
26	0.47	0.42	0.45	300
27	0.37	0.25	0.30	300
28	0.27	0.07	0.11	300
29	0.28	0.28	0.28	300
30	0.31	0.55	0.40	300
31	0.29	0.22	0.25	300
32	0.24	0.17	0.20	300
33	0.25	0.44	0.32	300
34	0.33	0.13	0.18	300
35	0.31	0.49	0.38	300
Accuracy			**0.3858**	10800
Macro Avg	0.37	0.39	0.36	10800
Weighted Avg	0.37	0.39	0.36	10800

Table 8: Classification Report for Test Data for hidden units 5

Results: For hidden units = 10

Table 9: Train Results for hidden units 10

Metric	Precision	Recall	F1-Score
Macro Average	0.63	0.64	0.63
Weighted Average	0.63	0.64	0.63
Accuracy		0.6362	
Overall Scores	0.6321	0.6362	0.6319

Table 10: Test Data for hidden units 10

Metric	Precision	Recall	F1-Score
Macro Average	0.55	0.56	0.55
Weighted Average	0.55	0.56	0.55
Accuracy		0.5556	
Overall Scores	0.5508	0.5556	0.5507

Table 11: Classification Report for Train Data for hidden units 10

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.84	0.78	0.81	600
1	0.53	0.50	0.52	600
2	0.62	0.70	0.66	600
3	0.49	0.53	0.51	600
4	0.58	0.52	0.54	600
5	0.63	0.75	0.68	600
6	0.55	0.62	0.58	600
7	0.69	0.71	0.70	600
8	0.79	0.82	0.81	600
9	0.65	0.73	0.69	600
10	0.74	0.81	0.77	600
11	0.78	0.79	0.79	600
12	0.60	0.66	0.63	600
13	0.71	0.80	0.75	600
14	0.66	0.71	0.68	600
15	0.71	0.66	0.69	600
16	0.48	0.45	0.47	600
17	0.59	0.47	0.52	600
18	0.57	0.56	0.56	600
19	0.69	0.72	0.70	600
20	0.52	0.62	0.57	600
21	0.78	0.84	0.81	600
22	0.66	0.58	0.62	600
23	0.70	0.73	0.72	600
24	0.57	0.40	0.47	600
25	0.33	0.25	0.28	600
26	0.75	0.80	0.77	600
27	0.73	0.72	0.73	600
28	0.51	0.45	0.48	600
29	0.54	0.57	0.55	600
30	0.66	0.74	0.70	600
31	0.51	0.42	0.46	600
32	0.63	0.57	0.60	600
33	0.72	0.69	0.70	600
34	0.56	0.47	0.51	600
35	0.66	0.77	0.71	600

Table 12: Classification Report for Test Data for hidden units 10

Class	Precision	Recall	F1-Score	Support
0	0.73	0.71	0.72	300
1	0.39	0.35	0.37	300
2	0.55	0.64	0.59	300
3	0.49	0.54	0.51	300
4	0.47	0.44	0.45	300
5	0.53	0.66	0.59	300
6	0.47	0.52	0.49	300
7	0.60	0.64	0.62	300
8	0.71	0.76	0.73	300
9	0.61	0.70	0.66	300
10	0.73	0.76	0.74	300
11	0.73	0.73	0.73	300
12	0.51	0.53	0.52	300
13	0.59	0.73	0.65	300
14	0.62	0.60	0.61	300
15	0.60	0.64	0.62	300
16	0.40	0.41	0.40	300
17	0.52	0.41	0.46	300
18	0.54	0.53	0.54	300
19	0.58	0.58	0.58	300
20	0.47	0.55	0.51	300
21	0.68	0.77	0.72	300
22	0.60	0.51	0.55	300
23	0.58	0.64	0.61	300
24	0.45	0.32	0.38	300
25	0.28	0.20	0.23	300
26	0.66	0.68	0.67	300
27	0.65	0.56	0.60	300
28	0.46	0.38	0.41	300
29	0.49	0.57	0.53	300
30	0.57	0.65	0.61	300
31	0.44	0.35	0.39	300
32	0.53	0.48	0.50	300
33	0.62	0.54	0.58	300
34	0.46	0.40	0.42	300
35	0.52	0.55	0.54	300

Results: For hidden units = 50

Table 13: Training Results for hidden units 50

Metric	Precision	Recall	F1-score
Train	0.9057	0.9054	0.9054

Table 14: Testing Results for hidden units 50

Metric	Precision	Recall	F1-score
Test	0.7609	0.7607	0.7605

Table 15: Classification Report of Train Data for hidden units 50

	precision	recall	f1-score	support
0	0.94	0.94	0.94	600
1	0.86	0.89	0.88	600
2	0.90	0.90	0.90	600
3	0.85	0.86	0.86	600
4	0.88	0.84	0.86	600
5	0.91	0.92	0.91	600
6	0.93	0.88	0.90	600
7	0.92	0.93	0.92	600
8	0.97	0.96	0.97	600
9	0.94	0.92	0.93	600
10	0.94	0.96	0.95	600
11	0.92	0.93	0.92	600
12	0.86	0.88	0.87	600
13	0.93	0.94	0.93	600
14	0.92	0.93	0.92	600
15	0.91	0.94	0.93	600
16	0.87	0.90	0.89	600
17	0.94	0.89	0.91	600
18	0.91	0.87	0.89	600
19	0.93	0.89	0.91	600
20	0.86	0.90	0.88	600
21	0.95	0.95	0.95	600
22	0.88	0.86	0.87	600
23	0.92	0.91	0.91	600
24	0.88	0.86	0.87	600
25	0.85	0.87	0.86	600
26	0.94	0.93	0.93	600
27	0.93	0.93	0.93	600
28	0.87	0.86	0.87	600
29	0.87	0.92	0.90	600
30	0.91	0.93	0.92	600
31	0.85	0.88	0.87	600
32	0.91	0.91	0.91	600
33	0.93	0.93	0.93	600
34	0.89	0.89	0.89	600
35	0.91	0.94	0.93	600
accuracy			0.91	21600
macro avg	0.91	0.91	0.91	21600
weighted avg	0.91	0.91	0.91	21600

Table 16: Classification Report of Test Data for hidden units 50

	precision	recall	f1-score	support
0	0.82	0.83	0.83	300
1	0.67	0.71	0.69	300
2	0.76	0.73	0.75	300
3	0.65	0.66	0.66	300
4	0.73	0.70	0.72	300
5	0.81	0.76	0.78	300
6	0.74	0.70	0.72	300
7	0.82	0.85	0.83	300
8	0.85	0.84	0.84	300
9	0.84	0.85	0.85	300
10	0.90	0.90	0.90	300
11	0.86	0.83	0.84	300
12	0.73	0.78	0.75	300
13	0.83	0.84	0.83	300
14	0.77	0.80	0.79	300
15	0.83	0.85	0.84	300
16	0.67	0.70	0.68	300
17	0.76	0.75	0.76	300
18	0.68	0.63	0.66	300
19	0.73	0.69	0.71	300
20	0.72	0.78	0.75	300
21	0.84	0.86	0.85	300
22	0.72	0.65	0.68	300
23	0.78	0.77	0.78	300
24	0.75	0.70	0.72	300
25	0.64	0.64	0.64	300
26	0.86	0.84	0.85	300
27	0.85	0.85	0.85	300
28	0.64	0.64	0.64	300
29	0.71	0.73	0.72	300
30	0.74	0.79	0.77	300
31	0.65	0.65	0.65	300
32	0.72	0.73	0.72	300
33	0.77	0.80	0.78	300
34	0.75	0.75	0.75	300
35	0.80	0.79	0.79	300
accuracy			0.76	10800
macro avg	0.76	0.76	0.76	10800
weighted avg	0.76	0.76	0.76	10800

Results: For hidden units = 100

Table 17: Training Results for hidden units 100

Metric	Precision	Recall	F1-score
Train	0.9676	0.9675	0.9675

Table 18: Testing Results for hidden units 100

Metric	Precision	Recall	F1-score
Test	0.8250	0.8243	0.8241

Table 19: Classification Report of Train Data for hidden units 100

Class	Precision	Recall	F1-score	Support
0	0.98	0.98	0.98	600
1	0.97	0.97	0.97	600
2	0.97	0.96	0.97	600
3	0.95	0.95	0.95	600
4	0.96	0.96	0.96	600
5	0.97	0.98	0.97	600
6	0.97	0.96	0.97	600
7	0.96	0.97	0.97	600
8	0.99	0.99	0.99	600
9	0.97	0.97	0.97	600
10	0.97	0.97	0.97	600
11	0.97	0.98	0.98	600
12	0.97	0.95	0.96	600
13	0.97	0.96	0.97	600
14	0.96	0.98	0.97	600
15	0.97	0.98	0.97	600
16	0.96	0.96	0.96	600
17	0.98	0.95	0.97	600
18	0.97	0.96	0.96	600
19	0.96	0.95	0.95	600
20	0.97	0.97	0.97	600
21	0.98	0.99	0.98	600
22	0.96	0.95	0.95	600
23	0.98	0.97	0.98	600
24	0.97	0.95	0.96	600
25	0.94	0.96	0.95	600
26	0.97	0.97	0.97	600
27	0.98	0.97	0.97	600
28	0.95	0.96	0.95	600
29	0.95	0.98	0.97	600
30	0.96	0.96	0.96	600
31	0.97	0.95	0.96	600
32	0.98	0.98	0.98	600
33	0.98	0.98	0.98	600
34	0.96	0.96	0.96	600
35	0.97	0.98	0.97	600
Accuracy		0.9675 (21600 samples)		
Macro Avg	0.97	0.97	0.97	21600
Weighted Avg	0.97	0.97	0.97	21600

Table 20: Classification Report of Test Data for hidden units 100

Class	Precision	Recall	F1-score	Support
0	0.92	0.90	0.91	300
1	0.76	0.76	0.76	300
2	0.87	0.84	0.85	300
3	0.76	0.77	0.76	300
4	0.82	0.78	0.80	300
5	0.85	0.84	0.84	300
6	0.75	0.78	0.76	300
7	0.84	0.86	0.85	300
8	0.91	0.90	0.91	300
9	0.89	0.89	0.89	300
10	0.94	0.94	0.94	300
11	0.89	0.90	0.90	300
12	0.80	0.82	0.81	300
13	0.83	0.88	0.85	300
14	0.82	0.86	0.84	300
15	0.87	0.92	0.89	300
16	0.77	0.71	0.74	300
17	0.79	0.80	0.79	300
18	0.76	0.75	0.76	300
19	0.75	0.78	0.77	300
20	0.79	0.79	0.79	300
21	0.88	0.91	0.89	300
22	0.78	0.76	0.77	300
23	0.85	0.80	0.82	300
24	0.79	0.77	0.78	300
25	0.71	0.75	0.73	300
26	0.87	0.87	0.87	300
27	0.89	0.87	0.88	300
28	0.82	0.78	0.80	300
29	0.80	0.86	0.83	300
30	0.78	0.86	0.82	300
31	0.83	0.70	0.76	300
32	0.82	0.78	0.80	300
33	0.81	0.85	0.83	300
34	0.80	0.81	0.81	300
35	0.89	0.83	0.86	300
Accuracy	0.8243 (10800 samples)			
Macro Avg	0.83	0.82	0.82	10800
Weighted Avg	0.83	0.82	0.82	10800

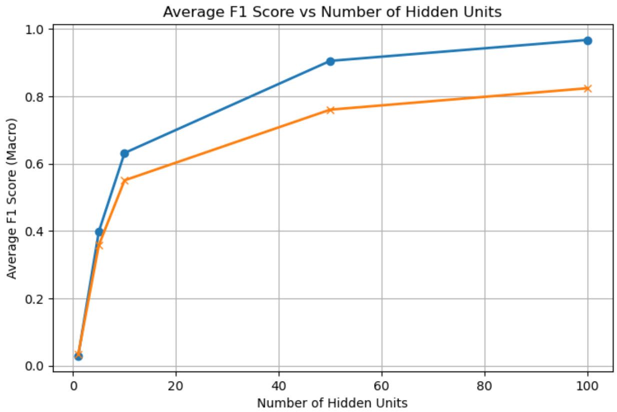


Figure 1: average f1 score vs number of hidden units

comments:

- As number of hidden units in a single hidden layer increase from 1 to 100 the eval metrics i.e., precision,recall and f1 score increase from 3.03 precent at hidden units 1 to 96.75 precent at hidden units 100 for training data.
- This show that model fails to learning patterns in the data with few hidden units.
- But overfits data with large number of hidden units.
- As we know, neural networks are universal approximators, they can approx any function, and with enough neurons can easily overfit the training data.
- Also single layer with enough hidden units is all that is needed to approximate any function.
- Model also generalises good with test data with more number of hidden units achieving 82.41 percent accuracy with 100 hidden units.

(c) Implementing a simple neural network with different number of hidden layer.

given:

- number of hidden depths = [[512],[512,256],[512,256,128],[512,256,128,64]]
- learning rate = 0.01
- mini-batch size = 32
- input dimension = 3072
- output dimension = 36
- nof epochs=350
- Stopping criterion used = max epochs=300 and change in loss less than 0.0001 with patience 5.

results:

Table 21: Performance Comparison for Different Network Architectures

Architecture	Train Metrics			Test Metrics		
	Precision	Recall	F1-score	Precision	Recall	F1-score
[512]	0.9991	0.9991	0.9991	0.8894	0.8894	0.8892
[512,256]	0.9997	0.9997	0.9997	0.8942	0.8939	0.8939
[512,256,128]	1.0000	1.0000	1.0000	0.8710	0.8705	0.8705
[512,256,128,64]	1.0000	1.0000	1.0000	0.8131	0.8129	0.8125

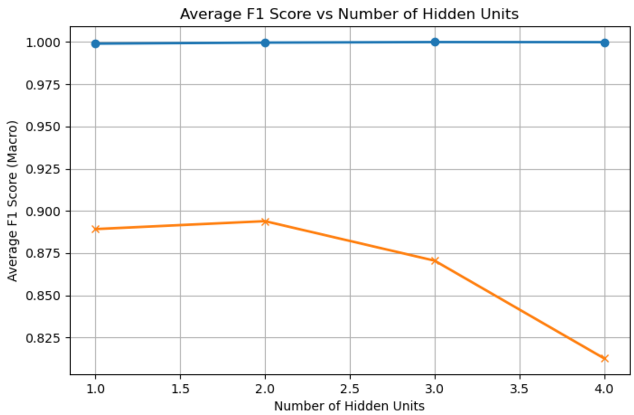


Figure 2: average f1 score vs depth

comments

- With increasing number of hidden layers and number of units in each layer, the model overfits the training data acheiving almost 100 percent F1 score.
- But test f1-score decreases from 88.92 to 81.25 percent.
- Because deeper networks are overfitting the training data, but simpler models generalize better.
- The decreasing test Performance with large model implies that capacity exceeds data regularization.

Table 22: Classification Report for Train and Test Data for [512]

Class	Train Data				Test Data			
	Precision	Recall	F1-score	Support	Precision	Recall	F1-score	Support
0	1.00	1.00	1.00	600	0.94	0.95	0.94	300
1	1.00	1.00	1.00	600	0.88	0.87	0.88	300
2	1.00	1.00	1.00	600	0.91	0.88	0.89	300
3	1.00	1.00	1.00	600	0.83	0.85	0.84	300
4	1.00	1.00	1.00	600	0.88	0.86	0.87	300
5	1.00	1.00	1.00	600	0.88	0.89	0.89	300
6	1.00	1.00	1.00	600	0.88	0.85	0.87	300
7	1.00	0.99	1.00	600	0.88	0.91	0.90	300
8	1.00	1.00	1.00	600	0.95	0.95	0.95	300
9	1.00	1.00	1.00	600	0.93	0.93	0.93	300
10	1.00	1.00	1.00	600	0.94	0.96	0.95	300
11	1.00	1.00	1.00	600	0.93	0.95	0.94	300
12	1.00	1.00	1.00	600	0.89	0.89	0.89	300
13	1.00	1.00	1.00	600	0.89	0.92	0.90	300
14	1.00	1.00	1.00	600	0.87	0.93	0.90	300
15	1.00	1.00	1.00	600	0.91	0.95	0.93	300
16	1.00	1.00	1.00	600	0.84	0.80	0.82	300
17	1.00	1.00	1.00	600	0.88	0.87	0.87	300
18	1.00	1.00	1.00	600	0.82	0.82	0.82	300
19	1.00	1.00	1.00	600	0.85	0.84	0.84	300
20	1.00	1.00	1.00	600	0.86	0.88	0.87	300
21	1.00	1.00	1.00	600	0.94	0.94	0.94	300
22	1.00	1.00	1.00	600	0.85	0.84	0.84	300
23	1.00	1.00	1.00	600	0.89	0.86	0.88	300
24	1.00	1.00	1.00	600	0.88	0.87	0.87	300
25	1.00	1.00	1.00	600	0.86	0.85	0.86	300
26	1.00	1.00	1.00	600	0.96	0.93	0.94	300
27	1.00	1.00	1.00	600	0.92	0.92	0.92	300
28	1.00	1.00	1.00	600	0.87	0.87	0.87	300
29	1.00	1.00	1.00	600	0.88	0.92	0.90	300
30	1.00	1.00	1.00	600	0.87	0.91	0.89	300
31	1.00	1.00	1.00	600	0.89	0.81	0.85	300
32	1.00	1.00	1.00	600	0.91	0.88	0.89	300
33	1.00	1.00	1.00	600	0.88	0.91	0.89	300
34	1.00	1.00	1.00	600	0.91	0.90	0.90	300
35	1.00	1.00	1.00	600	0.88	0.88	0.88	300
Accuracy			1.00	21600			0.89	10800
Macro Avg	1.00	1.00	1.00	21600	0.89	0.89	0.89	10800
Weighted Avg	1.00	1.00	1.00	21600	0.89	0.89	0.89	10800

Table 23: Classification Report for Train and Test Data for [512,256]

Class	Train Data				Test Data			
	Precision	Recall	F1-score	Support	Precision	Recall	F1-score	Support
0	1.00	1.00	1.00	600	0.94	0.94	0.94	300
1	1.00	1.00	1.00	600	0.89	0.88	0.88	300
2	1.00	1.00	1.00	600	0.89	0.89	0.89	300
3	1.00	1.00	1.00	600	0.80	0.85	0.83	300
4	1.00	1.00	1.00	600	0.87	0.88	0.87	300
5	1.00	1.00	1.00	600	0.89	0.89	0.89	300
6	1.00	1.00	1.00	600	0.88	0.87	0.87	300
7	1.00	1.00	1.00	600	0.90	0.90	0.90	300
8	1.00	1.00	1.00	600	0.97	0.93	0.95	300
9	1.00	1.00	1.00	600	0.93	0.94	0.94	300
10	1.00	1.00	1.00	600	0.96	0.95	0.95	300
11	1.00	1.00	1.00	600	0.92	0.94	0.93	300
12	1.00	1.00	1.00	600	0.89	0.89	0.89	300
13	1.00	1.00	1.00	600	0.92	0.92	0.92	300
14	1.00	1.00	1.00	600	0.90	0.94	0.92	300
15	1.00	1.00	1.00	600	0.91	0.96	0.93	300
16	1.00	1.00	1.00	600	0.83	0.81	0.82	300
17	1.00	1.00	1.00	600	0.90	0.87	0.88	300
18	1.00	1.00	1.00	600	0.85	0.83	0.84	300
19	1.00	1.00	1.00	600	0.84	0.86	0.85	300
20	1.00	1.00	1.00	600	0.87	0.88	0.88	300
21	1.00	1.00	1.00	600	0.95	0.94	0.94	300
22	1.00	1.00	1.00	600	0.84	0.84	0.84	300
23	1.00	1.00	1.00	600	0.89	0.87	0.88	300
24	1.00	1.00	1.00	600	0.86	0.86	0.86	300
25	1.00	1.00	1.00	600	0.87	0.86	0.86	300
26	1.00	1.00	1.00	600	0.96	0.94	0.95	300
27	1.00	1.00	1.00	600	0.91	0.93	0.92	300
28	1.00	1.00	1.00	600	0.88	0.89	0.88	300
29	1.00	1.00	1.00	600	0.88	0.91	0.90	300
30	1.00	1.00	1.00	600	0.85	0.91	0.88	300
31	1.00	1.00	1.00	600	0.88	0.85	0.87	300
32	1.00	1.00	1.00	600	0.90	0.86	0.88	300
33	1.00	1.00	1.00	600	0.90	0.92	0.91	300
34	1.00	1.00	1.00	600	0.93	0.90	0.91	300
35	1.00	1.00	1.00	600	0.91	0.89	0.90	300
Accuracy			1.00	21600			0.89	10800
Macro Avg	1.00	1.00	1.00	21600	0.89	0.89	0.89	10800
Weighted Avg	1.00	1.00	1.00	21600	0.89	0.89	0.89	10800

Table 24: Classification performance metrics for training and test data for [512,256,128]

Class	Train Data			Test Data		
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
0	1.00	1.00	1.00	0.92	0.93	0.92
1	1.00	1.00	1.00	0.90	0.82	0.86
2	1.00	1.00	1.00	0.91	0.85	0.88
3	1.00	1.00	1.00	0.78	0.82	0.80
4	1.00	1.00	1.00	0.85	0.84	0.85
5	1.00	1.00	1.00	0.85	0.87	0.86
6	1.00	1.00	1.00	0.88	0.89	0.88
7	1.00	1.00	1.00	0.90	0.89	0.89
8	1.00	1.00	1.00	0.92	0.94	0.93
9	1.00	1.00	1.00	0.92	0.91	0.91
10	1.00	1.00	1.00	0.96	0.93	0.94
11	1.00	1.00	1.00	0.92	0.93	0.92
12	1.00	1.00	1.00	0.86	0.88	0.87
13	1.00	1.00	1.00	0.90	0.93	0.91
14	1.00	1.00	1.00	0.87	0.87	0.87
15	1.00	1.00	1.00	0.90	0.93	0.92
16	1.00	1.00	1.00	0.81	0.82	0.81
17	1.00	1.00	1.00	0.89	0.85	0.87
18	1.00	1.00	1.00	0.81	0.82	0.81
19	1.00	1.00	1.00	0.84	0.85	0.85
20	1.00	1.00	1.00	0.82	0.85	0.84
21	1.00	1.00	1.00	0.93	0.91	0.92
22	1.00	1.00	1.00	0.82	0.78	0.80
23	1.00	1.00	1.00	0.88	0.85	0.86
24	1.00	1.00	1.00	0.79	0.83	0.81
25	1.00	1.00	1.00	0.79	0.80	0.80
26	1.00	1.00	1.00	0.92	0.93	0.93
27	1.00	1.00	1.00	0.89	0.92	0.91
28	1.00	1.00	1.00	0.86	0.83	0.84
29	1.00	1.00	1.00	0.89	0.90	0.90
30	1.00	1.00	1.00	0.84	0.88	0.86
31	1.00	1.00	1.00	0.87	0.83	0.85
32	1.00	1.00	1.00	0.87	0.82	0.84
33	1.00	1.00	1.00	0.84	0.86	0.85
34	1.00	1.00	1.00	0.86	0.88	0.87
35	1.00	1.00	1.00	0.89	0.90	0.90
Accuracy		1.00			0.87	
Macro Avg	1.00	1.00	1.00	0.87	0.87	0.87
Weighted Avg	1.00	1.00	1.00	0.87	0.87	0.87

Table 25: Classification performance metrics for training and test datasets for [512,256,128,64]

Class	Train Data			Test Data		
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
0	1.00	1.00	1.00	0.91	0.89	0.90
1	1.00	1.00	1.00	0.76	0.72	0.74
2	1.00	1.00	1.00	0.82	0.84	0.83
3	1.00	1.00	1.00	0.71	0.78	0.74
4	1.00	1.00	1.00	0.73	0.73	0.73
5	1.00	1.00	1.00	0.88	0.81	0.84
6	1.00	1.00	1.00	0.79	0.80	0.80
7	1.00	1.00	1.00	0.85	0.87	0.86
8	1.00	1.00	1.00	0.89	0.91	0.90
9	1.00	1.00	1.00	0.87	0.90	0.88
10	1.00	1.00	1.00	0.91	0.94	0.92
11	1.00	1.00	1.00	0.90	0.91	0.90
12	1.00	1.00	1.00	0.84	0.82	0.83
13	1.00	1.00	1.00	0.83	0.91	0.86
14	1.00	1.00	1.00	0.79	0.78	0.79
15	1.00	1.00	1.00	0.87	0.88	0.88
16	1.00	1.00	1.00	0.76	0.80	0.78
17	1.00	1.00	1.00	0.83	0.77	0.80
18	1.00	1.00	1.00	0.81	0.82	0.82
19	1.00	1.00	1.00	0.77	0.78	0.77
20	1.00	1.00	1.00	0.81	0.81	0.81
21	1.00	1.00	1.00	0.88	0.90	0.89
22	1.00	1.00	1.00	0.74	0.70	0.72
23	1.00	1.00	1.00	0.85	0.82	0.83
24	1.00	1.00	1.00	0.71	0.78	0.74
25	1.00	1.00	1.00	0.65	0.62	0.64
26	1.00	1.00	1.00	0.87	0.88	0.88
27	1.00	1.00	1.00	0.83	0.87	0.85
28	1.00	1.00	1.00	0.80	0.77	0.78
29	1.00	1.00	1.00	0.84	0.83	0.84
30	1.00	1.00	1.00	0.76	0.84	0.80
31	1.00	1.00	1.00	0.73	0.69	0.71
32	1.00	1.00	1.00	0.80	0.75	0.77
33	1.00	1.00	1.00	0.79	0.77	0.78
34	1.00	1.00	1.00	0.84	0.78	0.81
35	1.00	1.00	1.00	0.84	0.80	0.82
Accuracy		1.00			0.81	
Macro Avg	1.00	1.00	1.00	0.81	0.81	0.81
Weighted Avg	1.00	1.00	1.00	0.81	0.81	0.81

(d) Implementing a simple neural network with different number of hidden layer using relu activation function.

given:

- number of hidden depths = [[512],[512,256],[512,256,128],[512,256,128,64]]
- learning rate = 0.01
- mini-batch size = 32
- input dimension = 3072
- output dimension = 36
- nof epochs=350
- activation = ReLU
- Stopping criterion used = max epochs=300 and change in loss less than 0.0001 with patience 5.

results:

Table 26: Performance comparison for different network architectures

Architecture	Train Metrics			Test Metrics		
	Precision	Recall	F1-score	Precision	Recall	F1-score
[512]	1.0000	1.0000	1.0000	0.8947	0.8943	0.8942
[512,256]	1.0000	1.0000	1.0000	0.8960	0.8957	0.8956
[512,256,128]	1.0000	1.0000	1.0000	0.8986	0.8982	0.8982
[512,256,128,64]	1.0000	1.0000	1.0000	0.8952	0.8949	0.8949

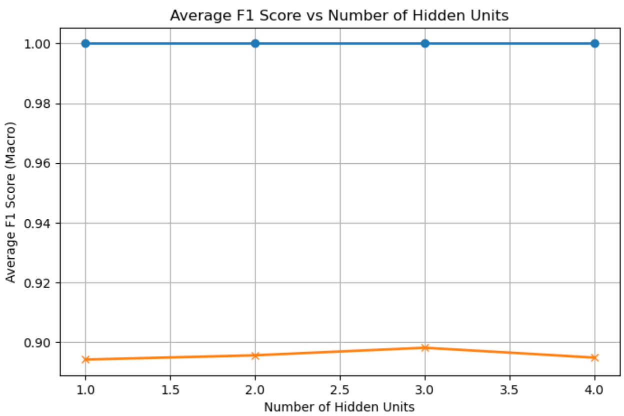


Figure 3: average f1 score vs depth

Comparison with part c:

Table 27: comparision of test Accuracy with part c

architectures	sigmoid	ReLU
[512]	0.8894	0.8943
[512,256]	0.8939	0.8957
[512,256,128]	0.8705	0.8982
[512,256,128,64]	0.8129	0.8949

Table 28: Classification Report for Train and Test Data using ReLU Activation for [512]

Class	Train Data				Test Data			
	Precision	Recall	F1-score	Support	Precision	Recall	F1-score	Support
0	1.00	1.00	1.00	600	0.97	0.95	0.96	300
1	1.00	1.00	1.00	600	0.91	0.88	0.89	300
2	1.00	1.00	1.00	600	0.91	0.89	0.90	300
3	1.00	1.00	1.00	600	0.82	0.85	0.83	300
4	1.00	1.00	1.00	600	0.91	0.87	0.89	300
5	1.00	1.00	1.00	600	0.89	0.92	0.90	300
6	1.00	1.00	1.00	600	0.91	0.86	0.89	300
7	1.00	1.00	1.00	600	0.87	0.91	0.89	300
8	1.00	1.00	1.00	600	0.94	0.94	0.94	300
9	1.00	1.00	1.00	600	0.94	0.94	0.94	300
10	1.00	1.00	1.00	600	0.95	0.96	0.95	300
11	1.00	1.00	1.00	600	0.92	0.94	0.93	300
12	1.00	1.00	1.00	600	0.87	0.88	0.87	300
13	1.00	1.00	1.00	600	0.91	0.92	0.91	300
14	1.00	1.00	1.00	600	0.87	0.93	0.90	300
15	1.00	1.00	1.00	600	0.92	0.97	0.94	300
16	1.00	1.00	1.00	600	0.85	0.78	0.81	300
17	1.00	1.00	1.00	600	0.88	0.87	0.88	300
18	1.00	1.00	1.00	600	0.85	0.84	0.84	300
19	1.00	1.00	1.00	600	0.83	0.87	0.85	300
20	1.00	1.00	1.00	600	0.87	0.90	0.88	300
21	1.00	1.00	1.00	600	0.96	0.95	0.95	300
22	1.00	1.00	1.00	600	0.87	0.85	0.86	300
23	1.00	1.00	1.00	600	0.90	0.87	0.88	300
24	1.00	1.00	1.00	600	0.88	0.88	0.88	300
25	1.00	1.00	1.00	600	0.82	0.84	0.83	300
26	1.00	1.00	1.00	600	0.98	0.93	0.96	300
27	1.00	1.00	1.00	600	0.93	0.92	0.93	300
28	1.00	1.00	1.00	600	0.85	0.85	0.85	300
29	1.00	1.00	1.00	600	0.88	0.92	0.90	300
30	1.00	1.00	1.00	600	0.87	0.91	0.89	300
31	1.00	1.00	1.00	600	0.90	0.85	0.87	300
32	1.00	1.00	1.00	600	0.90	0.86	0.88	300
33	1.00	1.00	1.00	600	0.88	0.92	0.90	300
34	1.00	1.00	1.00	600	0.94	0.90	0.92	300
35	1.00	1.00	1.00	600	0.89	0.89	0.89	300
Accuracy	1.00 (21600 samples)				0.89 (10800 samples)			
Macro Avg	1.00	1.00	1.00	21600	0.89	0.89	0.89	10800
Weighted Avg	1.00	1.00	1.00	21600	0.89	0.89	0.89	10800

Table 29: Classification Report for Train and Test Data (Using ReLU Activation) for [512,256]

Class	Train Data			Test Data			Support (Test)
	Precision	Recall	F1-score	Precision	Recall	F1-score	
0	1.00	1.00	1.00	0.97	0.96	0.96	300
1	1.00	1.00	1.00	0.90	0.87	0.88	300
2	1.00	1.00	1.00	0.92	0.90	0.91	300
3	1.00	1.00	1.00	0.83	0.82	0.82	300
4	1.00	1.00	1.00	0.90	0.86	0.88	300
5	1.00	1.00	1.00	0.87	0.90	0.89	300
6	1.00	1.00	1.00	0.87	0.87	0.87	300
7	1.00	1.00	1.00	0.88	0.91	0.89	300
8	1.00	1.00	1.00	0.96	0.95	0.95	300
9	1.00	1.00	1.00	0.94	0.95	0.95	300
10	1.00	1.00	1.00	0.95	0.96	0.96	300
11	1.00	1.00	1.00	0.93	0.95	0.94	300
12	1.00	1.00	1.00	0.88	0.90	0.89	300
13	1.00	1.00	1.00	0.91	0.95	0.93	300
14	1.00	1.00	1.00	0.92	0.94	0.93	300
15	1.00	1.00	1.00	0.94	0.97	0.95	300
16	1.00	1.00	1.00	0.84	0.80	0.82	300
17	1.00	1.00	1.00	0.91	0.87	0.89	300
18	1.00	1.00	1.00	0.84	0.85	0.85	300
19	1.00	1.00	1.00	0.85	0.87	0.86	300
20	1.00	1.00	1.00	0.83	0.89	0.86	300
21	1.00	1.00	1.00	0.94	0.94	0.94	300
22	1.00	1.00	1.00	0.84	0.83	0.83	300
23	1.00	1.00	1.00	0.90	0.87	0.88	300
24	1.00	1.00	1.00	0.88	0.86	0.87	300
25	1.00	1.00	1.00	0.84	0.83	0.83	300
26	1.00	1.00	1.00	0.97	0.94	0.95	300
27	1.00	1.00	1.00	0.94	0.92	0.93	300
28	1.00	1.00	1.00	0.84	0.88	0.86	300
29	1.00	1.00	1.00	0.89	0.94	0.92	300
30	1.00	1.00	1.00	0.88	0.90	0.89	300
31	1.00	1.00	1.00	0.88	0.84	0.86	300
32	1.00	1.00	1.00	0.91	0.86	0.88	300
33	1.00	1.00	1.00	0.89	0.93	0.91	300
34	1.00	1.00	1.00	0.92	0.89	0.91	300
35	1.00	1.00	1.00	0.90	0.89	0.90	300
Accuracy		1.0000			0.9000		
Macro Avg	1.0000	1.0000	1.0000	0.9000	0.9000	0.9000	
Weighted Avg	1.0000	1.0000	1.0000	0.9000	0.9000	0.9000	

Table 30: Classification Report for Train and Test Data (Using ReLU Activation) for [512,256,128]

Class	Train Data			Test Data			Support (Test)
	Precision	Recall	F1-score	Precision	Recall	F1-score	
0	1.00	1.00	1.00	0.97	0.94	0.96	300
1	1.00	1.00	1.00	0.88	0.87	0.88	300
2	1.00	1.00	1.00	0.91	0.89	0.90	300
3	1.00	1.00	1.00	0.85	0.84	0.84	300
4	1.00	1.00	1.00	0.87	0.87	0.87	300
5	1.00	1.00	1.00	0.89	0.92	0.91	300
6	1.00	1.00	1.00	0.90	0.87	0.89	300
7	1.00	1.00	1.00	0.90	0.92	0.91	300
8	1.00	1.00	1.00	0.93	0.95	0.94	300
9	1.00	1.00	1.00	0.95	0.94	0.94	300
10	1.00	1.00	1.00	0.96	0.95	0.96	300
11	1.00	1.00	1.00	0.92	0.93	0.93	300
12	1.00	1.00	1.00	0.89	0.90	0.90	300
13	1.00	1.00	1.00	0.91	0.93	0.92	300
14	1.00	1.00	1.00	0.89	0.94	0.91	300
15	1.00	1.00	1.00	0.91	0.96	0.94	300
16	1.00	1.00	1.00	0.89	0.85	0.87	300
17	1.00	1.00	1.00	0.88	0.89	0.89	300
18	1.00	1.00	1.00	0.85	0.85	0.85	300
19	1.00	1.00	1.00	0.85	0.88	0.87	300
20	1.00	1.00	1.00	0.85	0.89	0.87	300
21	1.00	1.00	1.00	0.95	0.96	0.95	300
22	1.00	1.00	1.00	0.83	0.85	0.84	300
23	1.00	1.00	1.00	0.91	0.86	0.89	300
24	1.00	1.00	1.00	0.87	0.86	0.87	300
25	1.00	1.00	1.00	0.88	0.82	0.85	300
26	1.00	1.00	1.00	0.98	0.95	0.96	300
27	1.00	1.00	1.00	0.93	0.92	0.92	300
28	1.00	1.00	1.00	0.84	0.87	0.85	300
29	1.00	1.00	1.00	0.89	0.93	0.91	300
30	1.00	1.00	1.00	0.88	0.91	0.89	300
31	1.00	1.00	1.00	0.89	0.83	0.86	300
32	1.00	1.00	1.00	0.94	0.87	0.91	300
33	1.00	1.00	1.00	0.89	0.91	0.90	300
34	1.00	1.00	1.00	0.90	0.89	0.89	300
35	1.00	1.00	1.00	0.91	0.89	0.90	300
Accuracy		1.0000			0.9000		
Macro Avg	1.0000	1.0000	1.0000	0.9000	0.9000	0.9000	
Weighted Avg	1.0000	1.0000	1.0000	0.9000	0.9000	0.9000	

Table 31: Classification Report for Train and Test Data for [512,256,128,64]

Class	Train Data			Test Data		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score
0	1.00	1.00	1.00	0.93	0.94	0.94
1	1.00	1.00	1.00	0.92	0.87	0.89
2	1.00	1.00	1.00	0.89	0.92	0.90
3	1.00	1.00	1.00	0.85	0.84	0.84
4	1.00	1.00	1.00	0.87	0.86	0.87
5	1.00	1.00	1.00	0.89	0.90	0.90
6	1.00	1.00	1.00	0.87	0.88	0.88
7	1.00	1.00	1.00	0.90	0.93	0.91
8	1.00	1.00	1.00	0.95	0.94	0.94
9	1.00	1.00	1.00	0.94	0.94	0.94
10	1.00	1.00	1.00	0.97	0.95	0.96
11	1.00	1.00	1.00	0.94	0.94	0.94
12	1.00	1.00	1.00	0.89	0.88	0.88
13	1.00	1.00	1.00	0.90	0.92	0.91
14	1.00	1.00	1.00	0.90	0.92	0.91
15	1.00	1.00	1.00	0.94	0.96	0.95
16	1.00	1.00	1.00	0.84	0.82	0.83
17	1.00	1.00	1.00	0.89	0.85	0.87
18	1.00	1.00	1.00	0.85	0.83	0.84
19	1.00	1.00	1.00	0.84	0.89	0.86
20	1.00	1.00	1.00	0.85	0.89	0.87
21	1.00	1.00	1.00	0.94	0.93	0.93
22	1.00	1.00	1.00	0.83	0.83	0.83
23	1.00	1.00	1.00	0.91	0.86	0.88
24	1.00	1.00	1.00	0.88	0.88	0.88
25	1.00	1.00	1.00	0.86	0.85	0.85
26	1.00	1.00	1.00	0.96	0.95	0.96
27	1.00	1.00	1.00	0.93	0.94	0.94
28	1.00	1.00	1.00	0.82	0.84	0.83
29	1.00	1.00	1.00	0.91	0.92	0.91
30	1.00	1.00	1.00	0.87	0.90	0.89
31	1.00	1.00	1.00	0.92	0.85	0.88
32	1.00	1.00	1.00	0.89	0.89	0.89
33	1.00	1.00	1.00	0.87	0.91	0.89
34	1.00	1.00	1.00	0.90	0.92	0.91
35	1.00	1.00	1.00	0.91	0.89	0.90
Accuracy		1.00			0.89	
Macro Avg		1.00			0.90	
Weighted Avg		1.00			0.90	

comments

- With increasing number of hidden layers and number of units in each layer, the model overfits the training data acheiving almost 100 percent F1 score.
- But test f1-score remains almost same by using ReLU activation function.
- Because deeper networks are overfitting the training data, but simpler models generalize better.
- ReLU activation enables faster convergence and effective learning of non-linear representations, improving generalization up to a moderate network depth.
- But with deeper networks model doesn't generalise better to test data even with ReLU activation function.

Conclusion:

- The model with the ReLU gives consistent performance across all architectures, achieving the highest test acc of 0.8982 with the [512, 256, 128] architecture.
- ReLU is better than sigmoid because it avoids vanishing gradients, enables faster convergence.

(e) Implementing a simple neural network with different number of hidden layer using relu activation function.

given:

- number of hidden depths = [[512],[512,256],[512,256,128],[512,256,128,64]]
- learning rate = 0.01
- mini-batch size = 32
- input dimension = 3072
- output dimension = 36
- nof epochs=350
- activation = ReLU
- Stopping criterion used = max epochs=300 and change in loss less than 0.0001 with patience 5.

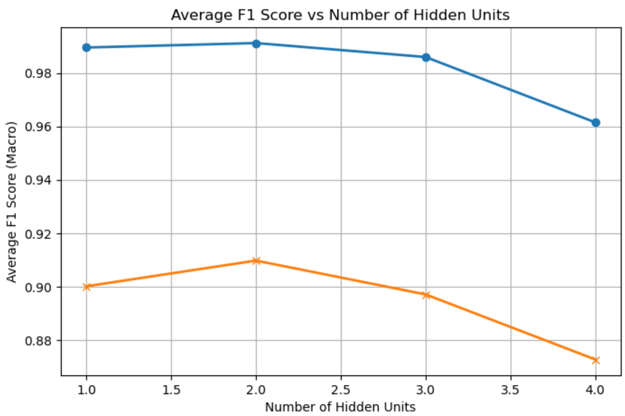


Figure 4: average f1 score vs depth

results:

Table 32: Performance Comparison of Different Network Architectures

Architecture	Train Metrics				Test Metrics			
	Acc	**Precision**	**Recall**	**F1-score**	**Acc**	**Precision**	**Recall**	**F1-score**
[512]	0.9896	0.9896	0.9896	0.9896	0.9004	0.9004	0.9004	0.9002
[512, 256]	0.9912	0.9913	0.9913	0.9912	0.9098	0.9102	0.9098	0.9098
[512, 256, 128]	0.9860	0.9862	0.9860	0.9860	0.8971	0.8994	0.8971	0.8971
[512, 256, 128, 64]	0.9614	0.9628	0.9614	0.9616	0.8723	0.8779	0.8723	0.8728

comparision:

Table 33: Comparison of Test Accuracy with Part (d) for Different Architectures

Architecture	Part (d) Accuracy (ReLU)	Current Accuracy (ReLU)	Improvement
[512]	0.8943	0.9004	+0.0061
[512, 256]	0.8957	0.9098	+0.0141
[512, 256, 128]	0.8982	0.8971	-0.0011
[512, 256, 128, 64]	0.8949	0.8723	-0.0226

comments:

- There is only slight improvement in initial model and model [512,256] shows best improvement over model from d(relu).

(f) Implementing a simple neural network with Digit dataset using relu activation function and implement transfer learning.

part(a): Training from Scratch on Digits Data

Given:

- hidden depths =[512,256,128,64]
- learning rate = 0.01
- mini-batch size = 32
- input dimension = 3072
- output dimension = 10
- nof epochs=20
- activation = ReLU
- Stopping criterion used = max epochs=20

results:

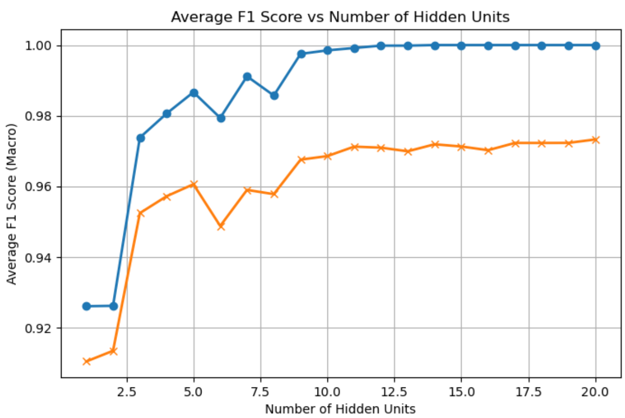


Figure 5: average f1 score vs depth

Table 34: Training and Testing Accuracy Results

Metric	Accuracy
Training Accuracy	1.0000
Testing Accuracy	0.9733
precision	0.9734
recall	0.9733
F1 score	0.9733

part(b): Transfer Learning from Consonants to Digits

results:

Table 35: Classification Performance after Transfer Learning

Dataset	Accuracy	Precision	Recall	F1-Score
Train	1.0000	0.9734	0.9733	0.9733
Test	0.9733	0.9735	0.9733	0.9733

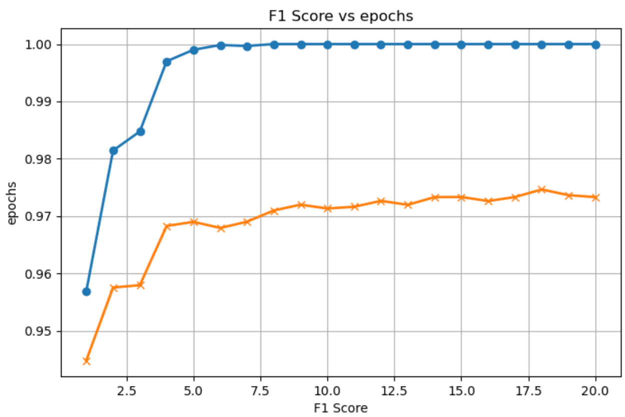


Figure 6: average f1 score vs depth

Table 36: Comparison of Training and Testing Metrics after Transfer Learning

Metric	Train	Test
Accuracy	1.0000	0.9733
Precision	0.9734	0.9735
Recall	0.9733	0.9733
F1-Score	0.9733	0.9733

Comments:

- Based on the results, the metrics remain almost similar before and after fine tuning for digit Classification using transfer learning.
- Based on the plots, after transfer learning the model has smoother f1-scores and achieves 100 percent accuracy on training set after just 5 epochs.
- which show performance of transfer learning is quite stable and sometimes even better that training from scratch.