Cameron Lee

4/29/2023

Professor Rohit Kate

COMPSCI-411

Dataset 1 – Airfoil (ID Number: 44957)

Overview: This dataset is obtained from NASA. It is obtained from a series of aerodynamic and acoustic tests of two and three-dimensional airfoil blade sections conducted in an anechoic wind tunnel. [Essentially, testing specific kind of blades to see how well it will move in a giant wind chamber.]

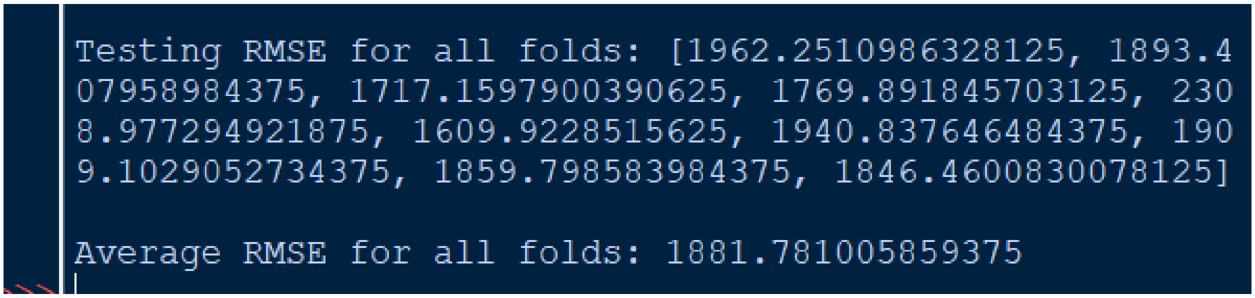
Task: The task of this dataset is to determine/predict the sound pressure of the airfoils depending on the different kind of features presented. This can be useful to make sure that people within a cabin do not get their hearing impaired. Additionally, this can be useful to figure out what kind of military airfoils may be needed to fly undetected by soundwaves/human ear capabilities.

Target: The target of this dataset is the specific sound pressure (in decibels) that is recorded depending on airfoil wind encounter.

Features: There were 5 numeric features. Frequency, angle of wind, chord length, free stream velocity, and displacement thickness were featured.

Results

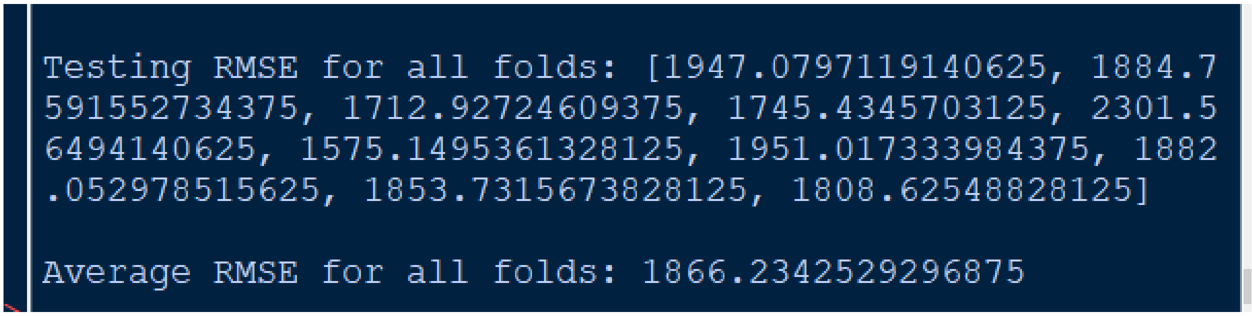
Neural Network Small # of Nodes [2] RMSE Result:



```
Testing RMSE for all folds: [1962.2510986328125, 1893.4
07958984375, 1717.1597900390625, 1769.891845703125, 230
8.977294921875, 1609.9228515625, 1940.837646484375, 190
9.1029052734375, 1859.798583984375, 1846.4600830078125]

Average RMSE for all folds: 1881.781005859375
```

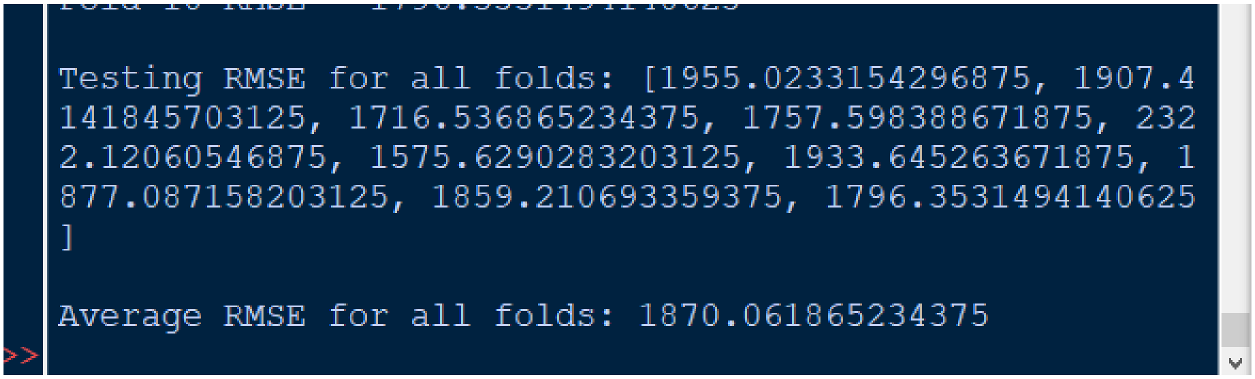
Neural Network Reasonable # of Nodes [10] RMSE Result:



```
Testing RMSE for all folds: [1947.0797119140625, 1884.7
591552734375, 1712.92724609375, 1745.4345703125, 2301.5
6494140625, 1575.1495361328125, 1951.017333984375, 1882
.052978515625, 1853.7315673828125, 1808.62548828125]

Average RMSE for all folds: 1866.2342529296875
```

Neural Network Too Many Nodes [50]:



```
Testing RMSE for all folds: [1955.0233154296875, 1907.4
141845703125, 1716.536865234375, 1757.598388671875, 232
2.12060546875, 1575.6290283203125, 1933.645263671875, 1
877.087158203125, 1859.210693359375, 1796.3531494140625
]

Average RMSE for all folds: 1870.061865234375
>>
```

Table of Average RMSE for Each Fold by Number of Nodes:

# Of Hidden Nodes	2	**10**	50
RMSE Average	1881.781	**1866.234**	1870.062

**Results in bold indicate best result

Table: According to the results generated, the best result was from using 10 hidden nodes which gave an average RMSE of 1866.234. As for the worst results, the "very few nodes" of 2 gave the worst RMSE of 1881.781 This would make sense as the results for the very few nodes would result in overfitting while the reasonable number of nodes should provide the best outcome.

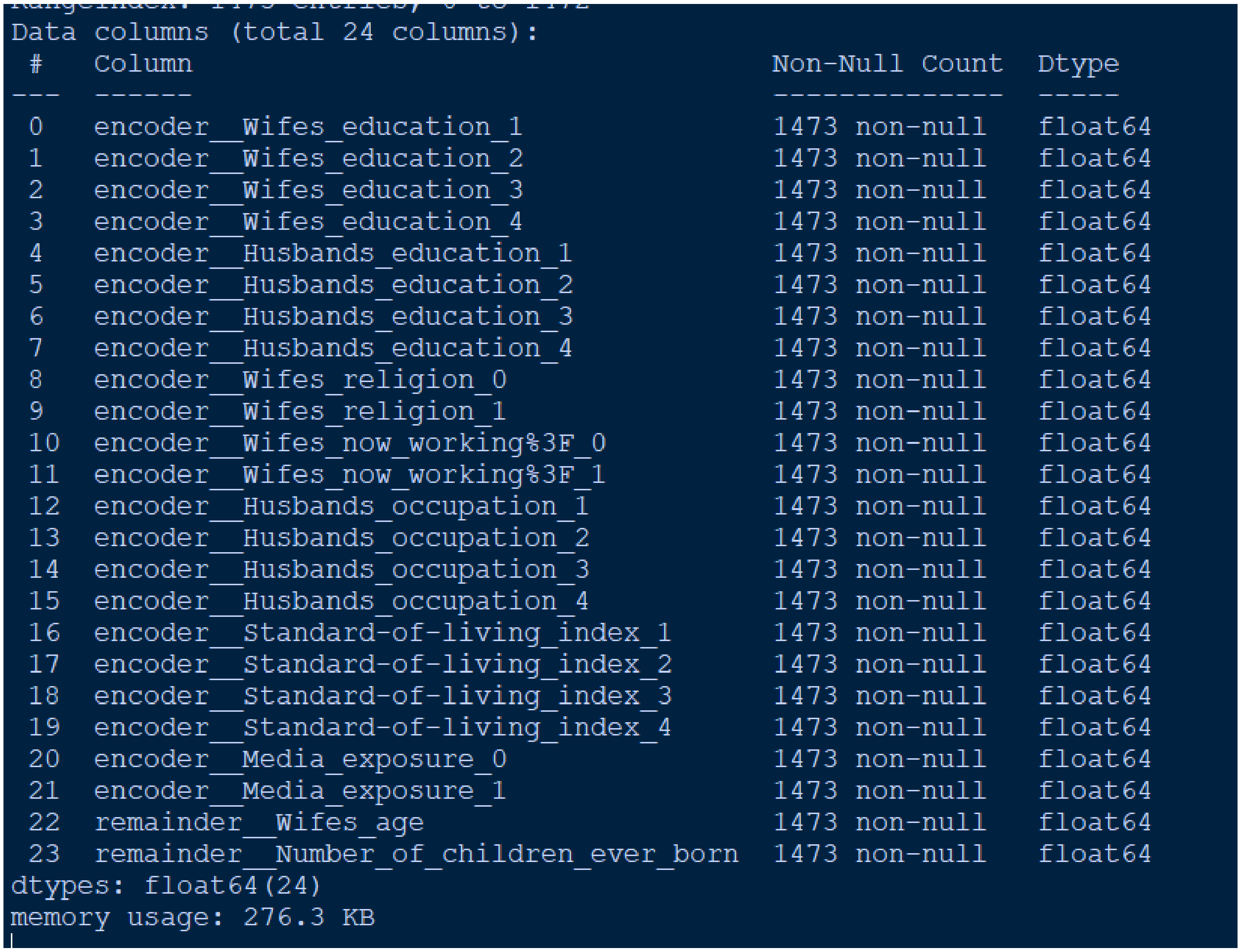
Dataset 2 – Indonesian Contraceptive Prevalence Survey (ID Number: 23)

Overview: This dataset is a compilation of survey results describing the Indonesian Contraceptive prevalence in 1987.

Task: The task of this dataset is to determine/predict the current contraceptive method choice of a woman based on her demographic and socio-economic characteristics. This can be useful when surveying an entire population (as Indonesia did here) in order to provide an accurate representation of what should be included in federal healthcare mandates for a country. Additionally, this would be good in determining the predicted population rate of a country.

Target: The target of this dataset is the contraceptive method used. It is coded with different numbers as 1 = No-Use, 2 = Long-Term Use, and 3 = Short-Term Use.

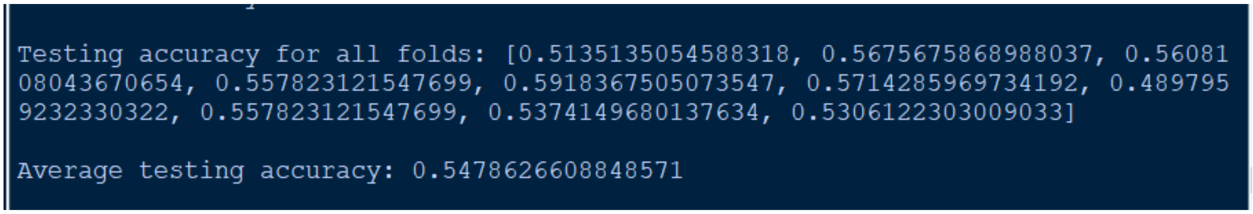
Features: There were two nominal features and seven categorical features. The nominal features were converted into numerical features by using one-hot encoding. So, the wife's education was split up into float64 by ranking of 1-4 with 4 being the highest, and 1 being the lowest. This is also the same for the husband's education. As for the wife's religion, that is a binary value that was converted to float64 to determine whether the person's religious beliefs were 0=Islam or 1=Not Islam. For the Husband's occupation, this was a categorical preference that I do not know of that just assigned a value of 1, 2, 3, or 4. As for the Standard-of-living, this was determining whether someone's standard of living was low=1 all the way up to 4=high. Lastly, this binary value was to determine whether there had ever been social media exposure of these people being either 0=good exposure or 1=not good exposure



```
RangeIndex: 1473 entries, 0 to 1472
Data columns (total 24 columns):
 #   Column                               Non-Null Count  Dtype  
---  ------                               --------------  -----  
 0   encoder__Wifes_education_1           1473 non-null   float64
 1   encoder__Wifes_education_2           1473 non-null   float64
 2   encoder__Wifes_education_3           1473 non-null   float64
 3   encoder__Wifes_education_4           1473 non-null   float64
 4   encoder__Husbands_education_1        1473 non-null   float64
 5   encoder__Husbands_education_2        1473 non-null   float64
 6   encoder__Husbands_education_3        1473 non-null   float64
 7   encoder__Husbands_education_4        1473 non-null   float64
 8   encoder__Wifes_religion_0            1473 non-null   float64
 9   encoder__Wifes_religion_1            1473 non-null   float64
 10  encoder__Wifes_now_working%3F_0      1473 non-null   float64
 11  encoder__Wifes_now_working%3F_1      1473 non-null   float64
 12  encoder__Husbands_occupation_1       1473 non-null   float64
 13  encoder__Husbands_occupation_2       1473 non-null   float64
 14  encoder__Husbands_occupation_3       1473 non-null   float64
 15  encoder__Husbands_occupation_4       1473 non-null   float64
 16  encoder__Standard-of-living_index_1  1473 non-null   float64
 17  encoder__Standard-of-living_index_2  1473 non-null   float64
 18  encoder__Standard-of-living_index_3  1473 non-null   float64
 19  encoder__Standard-of-living_index_4  1473 non-null   float64
 20  encoder__Media_exposure_0            1473 non-null   float64
 21  encoder__Media_exposure_1            1473 non-null   float64
 22  remainder__Wifes_age                 1473 non-null   float64
 23  remainder__Number_of_children_ever_born  1473 non-null   float64
dtypes: float64(24)
memory usage: 276.3 KB
```

Results

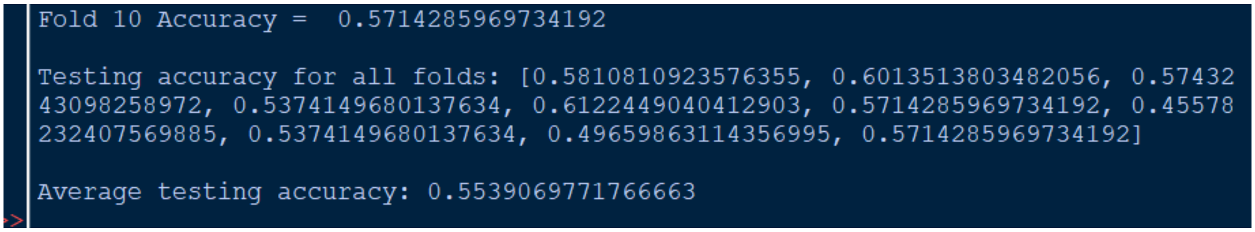
Neural Network Small # of Nodes [15] Accuracy Result:



```
Testing accuracy for all folds: [0.5135135054588318, 0.5675675868988037, 0.56081
08043670654, 0.557823121547699, 0.5918367505073547, 0.5714285969734192, 0.489795
9232330322, 0.557823121547699, 0.5374149680137634, 0.5306122303009033]

Average testing accuracy: 0.5478626608848571
```

Neural Network Normal # of Nodes [30] Accuracy Result:

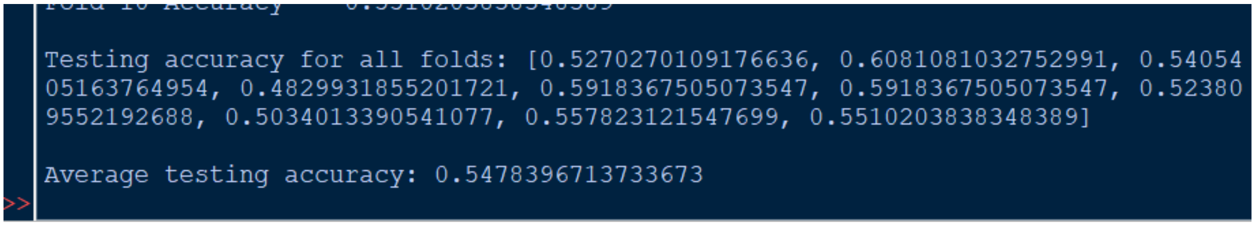


```
Fold 10 Accuracy =  0.5714285969734192

Testing accuracy for all folds: [0.5810810923576355, 0.6013513803482056, 0.57432
43098258972, 0.5374149680137634, 0.6122449040412903, 0.5714285969734192, 0.45578
232407569885, 0.5374149680137634, 0.49659863114356995, 0.5714285969734192]

Average testing accuracy: 0.5539069771766663
>>
```

Neural Network Large # of Nodes [45] Accuracy Result:



```
Fold 10 Accuracy =  0.5510203838348389

Testing accuracy for all folds: [0.5270270109176636, 0.6081081032752991, 0.54054
05163764954, 0.4829931855201721, 0.5918367505073547, 0.5918367505073547, 0.52380
9552192688, 0.5034013390541077, 0.557823121547699, 0.5510203838348389]

Average testing accuracy: 0.5478396713733673
>>
```

Table of Average Accuracy for Each Fold by Number of Nodes:

# of Hidden Nodes	15	**30**	45
Accuracy	0.5479	**0.5539**	0.5478

**Results in bold indicate best result

Table: According to the results generated, the best result was from using 30 hidden nodes which gave an average accuracy of 0.5539. As for the worst results, the “very few nodes” of 15 gave the worst accuracy of 0.5479. This would make sense as the results for the very few nodes would result in overfitting while the reasonable number of nodes should provide the best outcome. Additionally, the 45 hidden nodes’ accuracy was almost the exact same value as the 15 hidden nodes. This is proof that 30 truly is the best number of hidden nodes.

Conclusion:

For both datasets I was working on, it seemed that the “right” number of hidden nodes depended on the number of features and outputs included. In the airfoil dataset, there was a total of 5 numeric features and a single target. One of the best RMSE’s generated was from a hidden number of nodes that were a little larger than the sum of the feature and target outputs. If we were to compare this hypothesis to the second dataset, there were a total of 24 features (after one hot encoding) and 3 targets.

Using their sum of 27 and adding a few more nodes, the result comes to around 30-35 which is exactly what was used as the reasonable number of nodes in the neural network. Again, the 30 nodes that I used in my neural network generated the most accurate results allowing me to conclude that the number of features and target outputs are needed to determine the most effective number of nodes to use in a hidden layer.