**Pima Diabetes**

*Question #1: Run the code 5 times, record the accuracy and AUC scores of each run. What do you notice about the scores?*

--ML Model Output--

Decision Tree Acc: 0.724907063197026

Decision Tree AUC: 0.7188107267780801

--ML Model Output--

Decision Tree Acc: 0.7323420074349443

Decision Tree AUC: 0.7253264691109995

--ML Model Output--

Decision Tree Acc: 0.7063197026022305

Decision Tree AUC: 0.6657303370786517

--ML Model Output--

Decision Tree Acc: 0.7063197026022305

Decision Tree AUC: 0.6844378698224851

--ML Model Output--

Decision Tree Acc: 0.7063197026022305

Decision Tree AUC: 0.6829703569267997

The scores of accuracy and AUC appear to oscillate. Accuracy fluctuates between 0.70 and 0.73 and tends to perform better in the first two rounds then become sort of stable, which stays around 0.7. AUC goes up and down, which varies from 0.68 to 0.73.

*Question #2: Run the code again 5 times, record the accuracy and AUC scores of each run. What do you notice about the scores? How do they compare to scores above in question 1?*

--ML Model Output--

Decision Tree Acc: 0.6802973977695167

Decision Tree AUC: 0.6436014741390265

--ML Model Output--

Decision Tree Acc: 0.6914498141263941

Decision Tree AUC: 0.6513767131744659

--ML Model Output--

Decision Tree Acc: 0.7063197026022305

Decision Tree AUC: 0.6849376648285783

--ML Model Output--

Decision Tree Acc: 0.7360594795539034

Decision Tree AUC: 0.7292929292929292

--ML Model Output--

Decision Tree Acc: 0.6951672862453532

Decision Tree AUC: 0.6615885875308206

The scores of accuracy and AUC also appear to oscillate. Accuracy seems improving in the first four rounds and drops back to the beginning level, which varies from 0.68 to 0.74. AUC also goes up and down, which varies from 0.64 to 0.73.

Compared to scores in question 1, entropy scores on average are slightly lower and varies over a wider range.

*Question #3: Run the code 5 times, record the accuracy and AUC scores of each run. What do you notice about the scores? How do they compare to the simple test/train split scores in question 1?*

Decision Tree Acc: 0.71 (+/- 0.08)

Decision Tree AUC: 0.69 (+/- 0.07)

CV Runtime: 0.025072813034057617

Decision Tree Acc: 0.71 (+/- 0.08)

Decision Tree AUC: 0.69 (+/- 0.07)

CV Runtime: 0.02891373634338379

Decision Tree Acc: 0.71 (+/- 0.08)

Decision Tree AUC: 0.69 (+/- 0.07)

CV Runtime: 0.027074337005615234

Decision Tree Acc: 0.71 (+/- 0.08)

Decision Tree AUC: 0.69 (+/- 0.07)

CV Runtime: 0.02409219741821289

Decision Tree Acc: 0.71 (+/- 0.08)

Decision Tree AUC: 0.69 (+/- 0.07)

CV Runtime: 0.02603316307067871

Both accuracy and AUC scores are stable over 5 times. Average scores of accuracy and AUC from cross validation fall in range observed in question 1.

*Question #4: Run the code once for each cv setting (3,8,10), record the accuracy and AUC scores. What do you notice about the scores? How do they compare to the CV performance above in question 3?*

cv=3

Decision Tree Acc: 0.69 (+/- 0.06)

Decision Tree AUC: 0.67 (+/- 0.05)

CV Runtime: 0.017046451568603516

cv=8

Decision Tree Acc: 0.70 (+/- 0.09)

Decision Tree AUC: 0.67 (+/- 0.09)

CV Runtime: 0.043140411376953125

cv=10

Decision Tree Acc: 0.71 (+/- 0.14)

Decision Tree AUC: 0.67 (+/- 0.16)

CV Runtime: 0.0521085262298584

As the number of folds increase, the model average accuracy increased about 3%, from 0.69 to 0.71 and standard deviation increased 133%, from 0.06 to 0.14. AUC remains the same when we change folds in setting (3,8,10) but it reaches its maximum when fold equals 5.

**Wine Quality Dataset**

*Question #5: Run the code 5 times, record the RMSE and Expl Variance scores of each run. What do you notice about the scores?*

--ML Model Output--

Decision Tree RMSE: 0.8056718580521123

Decision Tree Expl Var: 0.009864226422837907

--ML Model Output--

Decision Tree RMSE: 0.8605957239029253

Decision Tree Expl Var: -0.14168709632955268

--ML Model Output--

Decision Tree RMSE: 0.7659028285997793

Decision Tree Expl Var: 0.0729486978150855

--ML Model Output--

Decision Tree RMSE: 0.7682308060918746

Decision Tree Expl Var: 0.12439574605220738

--ML Model Output--

Decision Tree RMSE: 0.7831940098997547

Decision Tree Expl Var: 0.0352222294986172

The first two rounds we have RMSE higher than 0.8 which results in pretty low variance explained. I don’t know how I get a negative variance in the second run because it is the average squared deviation from the mean and: Anything squared is never negative. The last three run I got similar RMSE, which is around 0.77. Even though I have two similar RMSE from third and fourth run, 0.7659 and 0.7682 respectively, the variance explained differed by 71%, 0.05. It’s interesting that a slightly better RMSE (lower) has worse explained variance (lower).

*Question #6: Run the code again 5 times, record the RMSE and Expl Variance of each run. What do you notice about the scores? How do they compare to scores above in question 5?*

--ML Model Output--

Decision Tree RMSE: 0.8098169987278431

Decision Tree Expl Var: 0.005865736287239076

--ML Model Output--

Decision Tree RMSE: 0.8326485281489252

Decision Tree Expl Var: -0.11941614823665136

--ML Model Output--

Decision Tree RMSE: 0.8358592670334608

Decision Tree Expl Var: -0.060509345997995956

--ML Model Output--

Decision Tree RMSE: 0.7428228014040188

Decision Tree Expl Var: 0.12028475574830222

--ML Model Output--

Decision Tree RMSE: 0.8248376463626905

Decision Tree Expl Var: -0.0473312351783024

4 out of 5 times I got RMSE higher than 0.8, which is worse than using metric ‘mse’. The lowest RMSE is 0.74 which is slightly better than what I have in question 5. Again when I got a large RMSE, say beyond 0.8, the explained variance become negative. Change regressor criterion did not show explicit improvement.

*Question #7: Run the code 5 times, record the RMSE and Expl Variance scores of each run. What do you notice about the scores? How do they compare to the simple test/train split scores in question 5?*

Decision Tree RMSE:: 0.90 (+/- 0.10)

Decision Tree Expl Var: -0.31 (+/- 0.17)

CV Runtime: 0.04264545440673828

Decision Tree RMSE:: 0.90 (+/- 0.10)

Decision Tree Expl Var: -0.31 (+/- 0.17)

CV Runtime: 0.04267239570617676

Decision Tree RMSE:: 0.90 (+/- 0.10)

Decision Tree Expl Var: -0.31 (+/- 0.17)

CV Runtime: 0.04609417915344238

Decision Tree RMSE:: 0.90 (+/- 0.10)

Decision Tree Expl Var: -0.31 (+/- 0.17)

CV Runtime: 0.046125173568725586

Decision Tree RMSE:: 0.90 (+/- 0.10)

Decision Tree Expl Var: -0.31 (+/- 0.17)

CV Runtime: 0.04575705528259277

RMSE and variance explained are both stable and RMSE is much higher than the results I got in question 5. Explained variance has the lowest number which is negative 0.31. I assumed this is due to the poor performance of the model as the prediction is way off the actual number.

*Question #8: Run the code once for each cv setting (3,8,10), record the RMSE and Expl Variance. What do you notice about the scores? How do they compare to the CV performance above in question 7?*

cv=3

Decision Tree RMSE:: 0.98 (+/- 0.09)

Decision Tree Expl Var: -0.46 (+/- 0.27)

CV Runtime: 0.023065805435180664

cv=8

Decision Tree RMSE:: 0.93 (+/- 0.11)

Decision Tree Expl Var: -0.50 (+/- 0.64)

CV Runtime: 0.08072161674499512

cv=10

Decision Tree RMSE:: 0.91 (+/- 0.16)

Decision Tree Expl Var: -0.47 (+/- 0.85)

CV Runtime: 0.09626173973083496

RMSE is close to 1 when cross validation fold is set to 3. When fold number increased, RMSE dropped from 0.98 to 0.91, and the best result was achieved when fold equals to 5. Not surprising, I got variance in negative for all running.

*Question #9: Run the code once, record the RMSE and Expl Variance. What do you notice about the scores? How do they compare to the CV performance above in question 7? What features were selected, and which were removed?*

Decision Tree RMSE:: 0.96 (+/- 0.04)

Decision Tree Expl Var: -0.48 (+/- 0.23)

CV Runtime: 0.028123855590820312

--LOW VARIANCE FILTER ON--

Selected: ['fixed acidity', 'residual sugar', 'free sulfur dioxide', 'total sulfur dioxide', 'alcohol']

Features (total, selected): 11 5

RMSE increased from 0.9 to 0.96 and variance dropped from negative 0.31 to negative 0.48, which means that the model is performing worse compared with the one in question 7.

5 out of 11 features were selected. Selected features were: 'fixed acidity', 'residual sugar', 'free sulfur dioxide', 'total sulfur dioxide', 'alcohol'. Removed features were: 'volatile acidity', 'citric acid', 'chlorides', 'density', 'pH', 'sulphates'.

*Question #10: Run the code once, record the RMSE and Expl Variance. What do you notice about the scores? How do they compare to the CV performance above in question 7? What features were selected, and which were removed?*

Decision Tree RMSE:: 0.91 (+/- 0.08)

Decision Tree Expl Var: -0.35 (+/- 0.26)

CV Runtime: 0.023062705993652344

--FEATURE SELECTION ON--

Wrapper Select:

Selected: ['volatile acidity', 'sulphates', 'alcohol']

Features (total/selected): 11 3

RMSE is almost the same as the result in question 7 and explained variance is still negative and is slightly lower.

3 out of 11 features were selected. Selected features were: 'volatile acidity', 'sulphates', 'alcohol'. Removed features were: 'citric acid', 'chlorides', 'density', 'pH', 'fixed acidity', 'residual sugar', 'free sulfur dioxide', 'total sulfur dioxide'.

**Summary Questions**

*\*Question #11: Were there any notable differences in performance between the two datasets? Given that one was a classification problem and the other a regression problem, can we say that the Decision Tree model performed better on one of them? Why or why not?*

Yes. In the first classification problem, model average accuracy could achieve 0.71, which is acceptable but still could be improved. In the regression problem, RMSE of decision tree regression is relatively high and the explained variance is much lower than expected (with even negative number). I would not say decision tree model performed better on classification as no other algorithms were used on the same dataset for comparison. It would be too narrow to conclude on just one algorithm.

*\*Question #12: Based on the results you obtained, would you say that Decision Tree is a “good” model for these two datasets? If your boss or a customer asked you to build a decision tree for one of them, what would you tell him/her?*

*I would not agree and argue for a “good” model. Normally I would expect a higher accuracy in terms of classification. In this case, 0,71 is not up to my standard. Besides, we only tested one algorithm for the dataset. There are many other classification algorithms, like logistic regression, SVM and etc. that may produce even better prediction results than decision. If my boss asks me to build a decision tree for classification, I would do so and in the meantime test additional algorithms for comparison. Decision tree may help understand the flow of classification process (if we plot the tree). But if the goal is not for interpretation but prediction, I would suggest looking for better algorithms.*