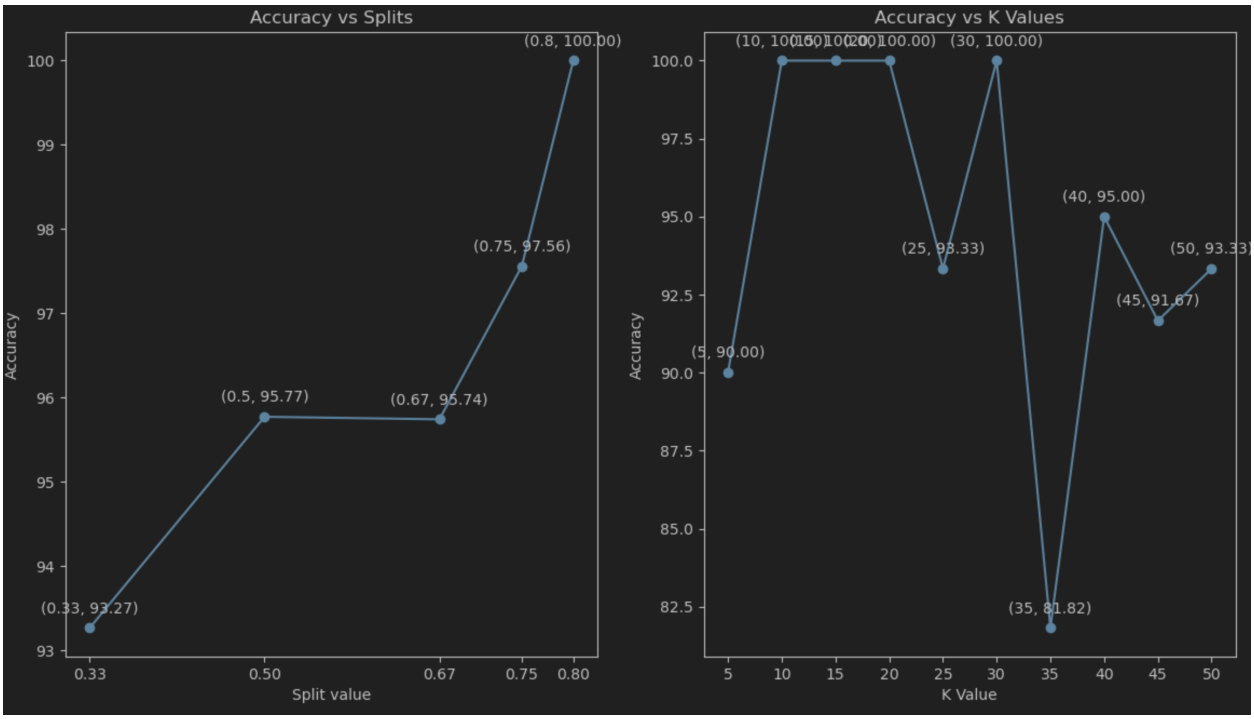
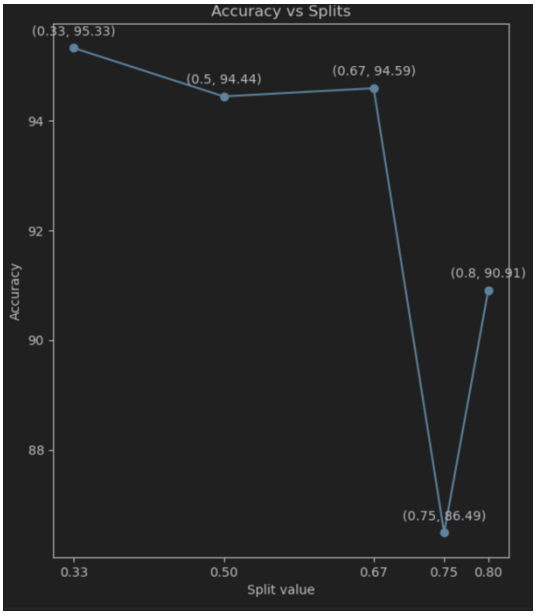
Lucas Sipos

Problem 1:



Accuracy vs Splits:

As we can see, the more test data we have, the bigger the accuracy gets. Logically speaking, this is correct, taking into consideration that maybe the data has more information about every aspect of the iris dataset. But that's not always the case. In the attached image on the right, we can see that even if the training set is 80%/75% out of the whole data, the accuracy of the run that only had 33% of the data is much bigger than both. The reason for that could be the following:



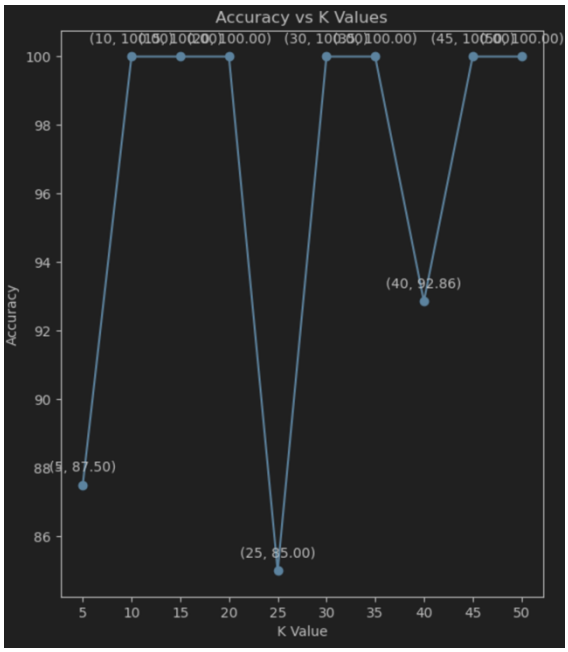
- The training set for the 33% slit contains a wider variety of data than the other 2 splits.
- The test set for the 33% split is more concentrated towards the training set than the other 2 splits.

Lucas Sipos

Accuracy vs K Values:

As expected, the graph is very random. In the first screenshot, we can see that only the 10 to 20 and 30 NN get 100% accuracy, as for the second screen attached on the left, only 5, 25 and 40 don't get 100% accuracy. A possible reason could be the following:

- When the data was split, we had imbalanced data, meaning more values of one type of result trained the model, which concluded with a model that was trained more on one of the outputs.



Problem 2:

```
last_letter = 'a'            female : male   =     35.8 : 1.0
last_letter = 'k'              male : female =     30.8 : 1.0
last_letter = 'f'              male : female =     15.9 : 1.0
last_letter = 'p'              male : female =     11.9 : 1.0
last_letter = 'v'              male : female =     11.2 : 1.0
last_letter = 'd'              male : female =      9.6 : 1.0
last_letter = 'm'              male : female =      8.6 : 1.0
last_letter = 'o'              male : female =      8.4 : 1.0
last_letter = 'r'              male : female =      6.6 : 1.0
last_letter = 'z'              male : female =      5.1 : 1.0
last_letter = 'w'              male : female =      5.1 : 1.0
last_letter = 'g'              male : female =      4.6 : 1.0
last_letter = 't'              male : female =      4.5 : 1.0
last_letter = 'b'              male : female =      4.4 : 1.0
last_letter = 's'              male : female =      4.2 : 1.0
last_letter = 'j'              male : female =      4.0 : 1.0
last_letter = 'i'            female : male   =      3.9 : 1.0
last_letter = 'u'              male : female =      3.2 : 1.0
last_letter = 'n'              male : female =      2.2 : 1.0
last_letter = 'x'              male : female =      1.9 : 1.0
last_letter = 'l'              male : female =      1.8 : 1.0
last_letter = 'e'            female : male   =      1.8 : 1.0
last_letter = 'h'              male : female =      1.5 : 1.0
last_letter = 'y'              male : female =      1.2 : 1.0
```

```
last_two_letters = 'vi'              male : female =      1.7 : 1.0
last_two_letters = 'ce'              male : female =      1.6 : 1.0
last_two_letters = 'ry'              male : female =      1.6 : 1.0
last_two_letters = 'ah'            female : male   =      1.5 : 1.0
last_two_letters = 'ss'              male : female =      1.5 : 1.0
last_two_letters = 'al'              male : female =      1.4 : 1.0
last_two_letters = 'ey'              male : female =      1.4 : 1.0
last_two_letters = 'yl'            female : male   =      1.4 : 1.0
last_two_letters = 'is'              male : female =      1.4 : 1.0
last_two_letters = 'my'              male : female =      1.4 : 1.0
last_two_letters = 'ty'            female : male   =      1.3 : 1.0
last_two_letters = 'un'              male : female =      1.3 : 1.0
last_two_letters = 'zy'              male : female =      1.3 : 1.0
last_two_letters = 'ly'            female : male   =      1.2 : 1.0
last_two_letters = 'th'            female : male   =      1.2 : 1.0
last_two_letters = 'wn'              male : female =      1.2 : 1.0
last_two_letters = 'cy'            female : male   =      1.2 : 1.0
last_two_letters = 'gy'              male : female =      1.1 : 1.0
last_two_letters = 'en'              male : female =      1.1 : 1.0
last_two_letters = 'py'            female : male   =      1.1 : 1.0
last_two_letters = 'dy'              male : female =      1.1 : 1.0
last_two_letters = 're'              male : female =      1.1 : 1.0
last_two_letters = 'et'              male : female =      1.0 : 1.0
last_two_letters = 'ny'            female : male   =      1.0 : 1.0
last_two_letters = 'ue'            female : male   =      1.0 : 1.0
last_two_letters = 'iz'            female : male   =      1.0 : 1.0
last_two_letters = 'Jo'            female : male   =      1.0 : 1.0
last_two_letters = 'rb'            female : male   =      1.0 : 1.0
```

Lucas Sipos

For this problem, the new feature I've used is classifying names by their last 2 letters.

a. The first thing that pops up when looking at the difference between these 2 lists of informative features is that if we are using only the last letter of a word, we are limited to having only 26 features in deciding what it's going to be. But, if we are using the last 2 letters, we are not as limited. Unfortunately, I cannot display a screenshot with all the features, but there are 100 features when using the last 2 letters and only 26 when using the last letters because it's the number of letters in the English alphabet. Yes, not always more is better, but in this case, I've attached a screenshot that will help you understand why we need the last 2 letters instead of only the last one. If you look closer, the first screenshot presents us with the fact that everything ending in "i" or "e" is a female, as for the second screenshot, we can see that the ones ending in "vi", "ce", and "re" are males, which just makes the model thing more about the decision he's making when looking at the possible gender of that name. We can also say the same for the male data. The first screenshot says that everything ending in "h", "l", "y", "z", "o", and "b" is a male, but the second screenshot implies that everything ending in "ah", "yl", "py" … (you get the point), is a female, which adds a little more perspective to classifying a name to the wrong gender.
b. For this part, I've looped through training and testing the data 1000 times twice and got the following accuracies:

Last Letter	Last 2 Letters
0.761854	**0.780704**
0.760038	**0.783168**

As we can see, and as expected, the last 2 letters feature is doing a little better than the last one. I thought it would be a bigger difference than this, but I guess that the last letter is pretty good, too. As said before, there is more information in the last 2 letters feature, and that's why it is doing better overall.