Lab 02: Decision Tree with scikit-learn

Decision Tree Classifier Evaluation

Interpretation

Interpreting the classification report

The classification report is of the form:

	precision	recall	f1-score	support
draw	***Pd***	***Rd***	***Fd***	***Sd***
loss	***PL***	***RL***	***FL***	***SL***
win	***Pw***	***Rw***	***Fw***	***Sw***
accuracy			***Accu***	***S***
macro avg	***Mp***	***Mr***	***Mf***	***S***
weighted avg	***Wp***	***Wr***	***Wf***	***S***

In which there are 22 calculated values as follows:

- ***Px***
  The precision of the category ***x***, with ***x*** $\in \{ draw, \ loss, \ win \}$.
  The value dictates the likelihood that the model is correct when predicting an observation as x.

  Is calculated by:

  $$Precision_x = \frac{TP_x}{TP_x + FP_x}$$

  In which:
  - $Precision_x$
    The precision value of category x.
  - TP_x
    Number of true positives of category x, that is, the number of observations that are of category x and is correctly predicted as x.
  - FP_x
    Number of false positives of category x, that is, the number of observations that are NOT of category x and is predicted as x.

- ***Rx***
  The precision of the category ***x***, with ***x*** $\in \{\, draw,\ loss,\ win \,\}$.
  The value dictates the sensitivity of the model to the category x.

  Is calculated by:

$$Recall_x = \frac{TP_x}{TP_x + FN_x}$$

  In which:
  - $Recall_x$
    The recall value of category x.
  - TP_x
    Number of true positives of category x, that is, the number of observations that are of category x and is correctly predicted as x.
  - FN_x
    Number of false negatives of category x, that is, the number of observations that are of category x and is incorrectly predicted as NOT x.

- ***Fx***
  The f1-score of the category ***x***, with ***x*** $\in \{\, draw,\ loss,\ win \,\}$.
  The value dictates how "good" is the model at predicting an observation as x.

  Is calculated by:

$$F_{1_x} = 2 \times \frac{Precision_x \times Recall_x}{Precision_x + Recall_x}$$

  In which:
  - F_{1_x}
    The f1-score value of category x.
  - $Precision_x$
    The precision value of category x.
  - $Recall_x$
    The recall value of category x.

- ***Sx***
  The number of observations with category ***x***, with ***x*** $\in \{\, draw,\ loss,\ win \,\}$.

- ***S***
  The total number of observations used to perform the test.

- ***Accu***
  The accuracy of the model.
  The value dictates the likelihood that the model makes a correct prediction.

  Is calculated by:
  $$Accuracy = \sum_x \frac{TP_x + TN_x}{TP_x + TN_x + FP_x + FN_x}$$
  In which:
  - $Accuracy$
    The accuracy value of the model.
  - x
    A category.
  - TP_x
    Number of true positives of category x, that is, the number of observations that are of category x and is correctly predicted as x.
  - TN_x
    Number of true negatives of category x, that is, the number of observations that are NOT of category x and is correctly predicted as NOT x.
  - FP_x
    Number of false positive of category x, that is, the number of observations that are NOT of category x and is incorrectly predicted as x.
  - FN_x
    Number of false negatives of category x, that is, the number of observations that are of category x and is incorrectly predicted as NOT x.
- ***My***
  The macro average of the metric ***y***, with ***y*** $\in \{Precision,\ Recall,\ F1\}$.

  Is calculated by:
  $$MacroAverage_y = \sum_x y_x$$
  In which:
  - $MacroAverage_y$
    The macro average of the metric y.
  - x
    The category x.
  - y_x
    The value of the metric y on the category x.

- ***Wy***
  The macro average of the metric ***y***, with ***y*** $\in \{Precision,\ Recall,\ F1\}$.

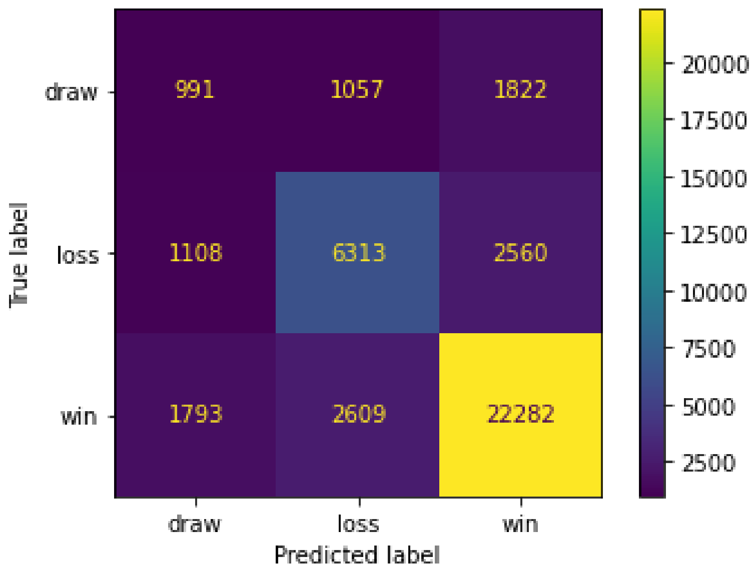
  Is calculated by:

$$MacroAverage_y = \sum_x W_x \times y_x$$

  In which:
  - $MacroAverage_y$
    The macro average of the metric y.
  - x
    The category x.
  - y_x
    The value of the metric y on the category x.
  - W_x
    The weight of the category x.

Interpreting the confusion matrix:

A sample of a confusion matrix:



The confusion matrix is an $N \times N$ matrix, with $N = number\ of\ categories$, whose value is the number of observations. The rows of the matrix represent observations that are defined as a certain category according to the dataset, that is, ground truth. The columns represent observations that are predicted by the model. As such, the value at $[0][1]$, or $[draw, loss]$, is the number of observations that are of the category $draw$ but is predicted as $loss$.

Note that in the confusion matrix, the diagonal values, values at indexes $[i, i]$, represents correct positives predictions, while the other values represent false predictions.

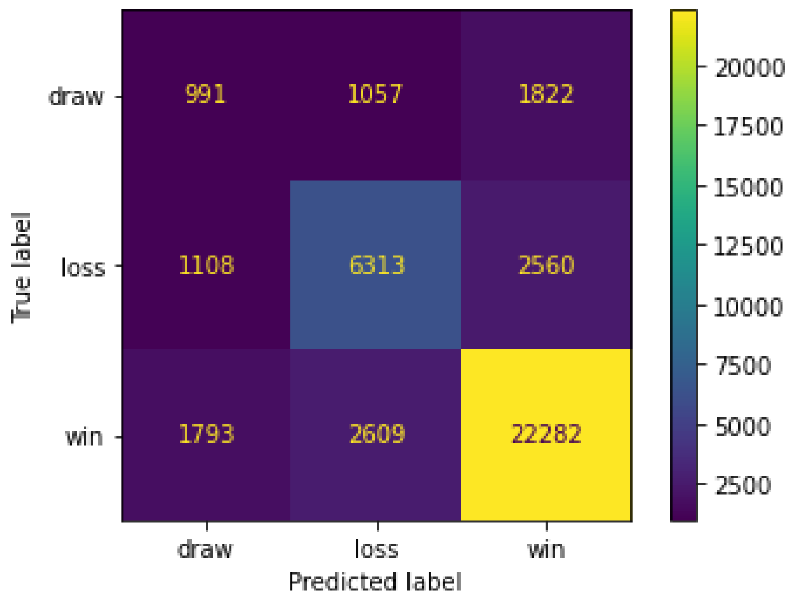
Results

The 40/60 Split

The classification report:

	precision	recall	f1-score	support
draw	0.25	0.26	0.26	3870
loss	0.63	0.63	0.63	9981
win	0.84	0.84	0.84	26684
accuracy			0.73	40535
macro avg	0.57	0.57	0.57	40535
weighted avg	0.73	0.73	0.73	40535

The confusion matrix:

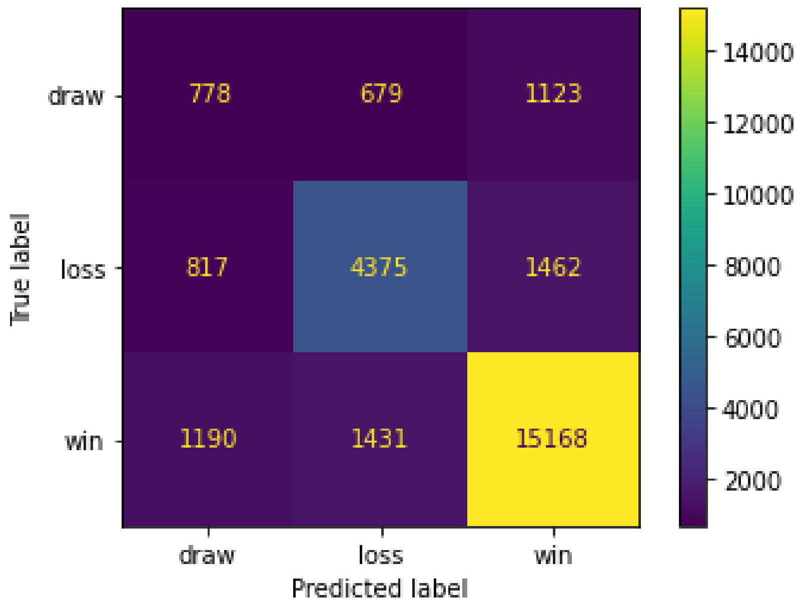


The 60/40 Split

The classification report:

	precision	recall	f1-score	support
draw	0.28	0.30	0.29	2580
loss	0.67	0.66	0.67	6654
win	0.85	0.85	0.85	17789
accuracy			0.75	27023
macro avg	0.60	0.60	0.60	27023
weighted avg	0.76	0.75	0.75	27023

The confusion matrix:

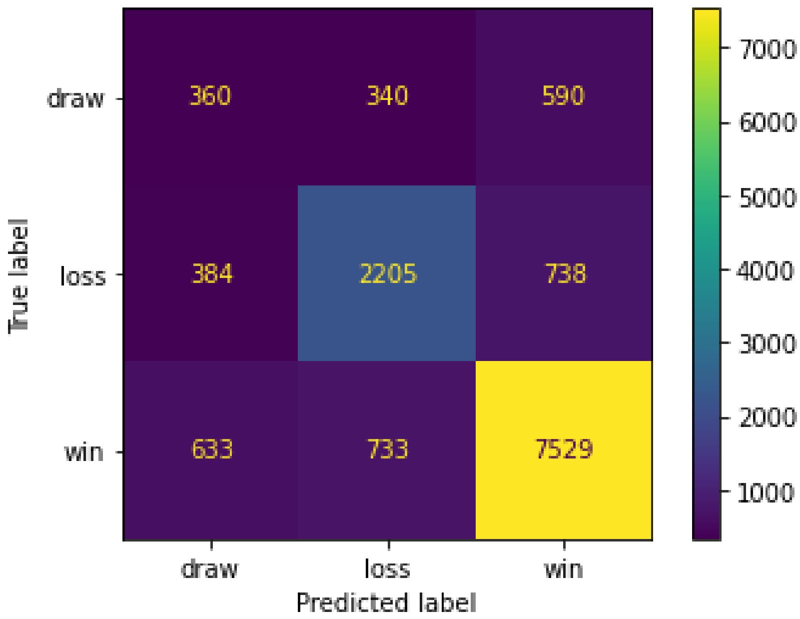


The 80/20 Split

The classification report:

	precision	recall	f1-score	support
draw	0.26	0.28	0.27	1290
loss	0.67	0.66	0.67	3327
win	0.85	0.85	0.85	8895
accuracy			0.75	13512
macro avg	0.59	0.60	0.60	13512
weighted avg	0.75	0.75	0.75	13512

The confusion matrix:

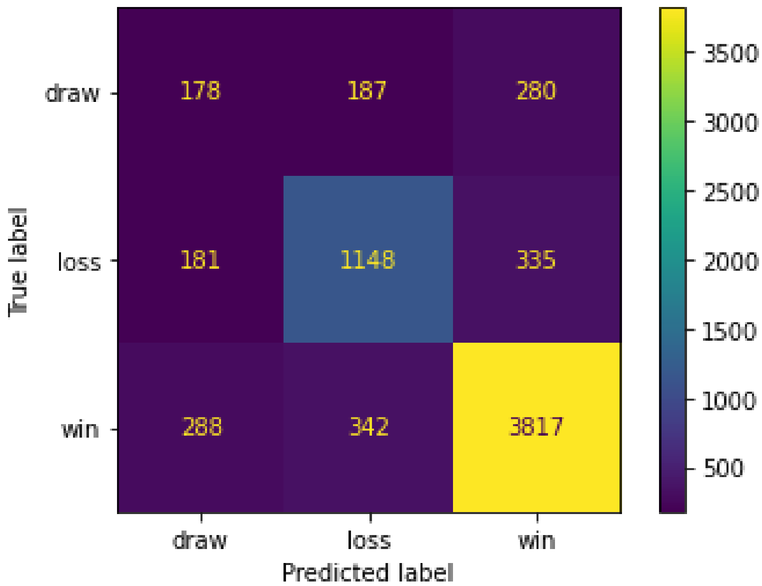


The 90/10 Split

The classification report:

	precision	recall	f1-score	support
draw	0.28	0.28	0.28	645
loss	0.68	0.69	0.69	1664
win	0.86	0.86	0.86	4447
accuracy			0.76	6756
macro avg	0.61	0.61	0.61	6756
weighted avg	0.76	0.76	0.76	6756

The confusion matrix:



Comment:

The metrics (precision, recall, and f1-score) are better as the proportion of the training set increases. This is as expected as the model performance are better when there are more training data.

For all train/test splitting proportion, the metrics (precision, recall, and f1-score) of the category "win" are higher than that of "loss", which are higher than that of "draw". This is due to the inconsistency in the proportion of observations for three classes: 65.83% for "win", 24.62% for "loss", and 9.55% for "draw". The lack of consistency in the number of observations per category in the original dataset leads to the model perform better on categories with higher data, and worse for others.

Accuracy of Decision Trees with Different Depths

max_depth	None	2	3	4	5	6	7
Accuracy	0.748	0.658	0.664	0.676	0.689	0.693	0.703

The accuracy increases by the order of the max depth [2, 3, 4, 5, 6, 7, None]. This is expected as an increasement in max depth would allow a tree with more branching, leading to more nodes to be considered. In other words, trees with higher depth would have a higher number of attributes (more data) being utilized, leading to higher accuracy.

Note that the configuration “max_depth=None” defines a tree without depth limitation.