**Tree Portfolio Data Files Creation**

**Step 1:** the raw input files are tables of characteristics (T \* N). In this step we combine the returns, size and the 3 characteristics information (which are also transformed into quantiles) and saved as yearly files.

Inputs:

Table, Excel

Description automatically generated

Outputs:

Table

Description automatically generated

**Step 2:** We then use the outputs from the previous step to generate tree portfolios. We need value-weighted returns at all levels of the trees, and the characteristics min/max (for plotting portfolio boundary in heatmaps and other statistics purpose).

Outputs:

Tree portfolio value-weighted returns for one single tree

Application, table

Description automatically generated

Tree portfolio characteristic min/max for one tree (will have 2\*3 such files for one single tree)

A picture containing text, newspaper

Description automatically generated

**Step 3:** in this step we combine the tree portfolios from all trees into one file and calculate the excess return over risk free rates. We also need to remove the duplicated upper level tree portfolios.

Outputs: value-weighted returns for all trees combined, similar format for characteristics min/max

Application, table

Description automatically generated

**Step 4:** remove the columns where the level 4 portfolios are sorted solely based on one characteristic

**AP-Pruning**

The main file to run is AP-Pruning.R where the tree portfolios are imported and normalized according to the depth of each portfolio. Then the helper functions are called to: 1) split data into train/valid/test period 2) transform the optimization into regression according to the appendix 3) use lars for the regression 4) calculate sparsity and the Sharpe ratios in each period.

For each pair of hyperparamter lambda0 and lambda2, we have outputs as below where the first three columns are SRs, 4th column is sparcity and the remaining columns are the weights.

**Graphical user interface, application, table

Description automatically generated**