

1. Classification_TheodoreBeevers_XGBoostTree.txt:
 Algorithm: xgb.train
 Key HP values: max_depth = 10, eta = 0.1, num_rounds = 125
 HP Optimisation: trying different parameters and plotting train and val loss
 Parameters in model:
 Loss function and value on validation set: 0.07824 (Binary cross entropy)
 Training time: 47.0s
 Feature selection: Inbuilt feature importance
 Own evaluation: Good model with high accuracy and low training time
2. Classification_TheodoreBeevers_MLPClassifier_RandomSearchCV.txt:
 Algorithm: SKLearn MLPClassifier
 Preprocessing: Scaled using min max scaler
 Key HP values: hidden_Layer_sizes = (24, 91, 79), solver = Adam, eta = 0.001, activation = relu, alpha = 0.1
 HP Optimisation: Parameters selected using random search cv with k=2 for speed
 Parameters in model: 10127
 Loss function and value on validation set: 0.7575 (Binary cross entropy)
 Feature selection: Inbuilt feature importance
 Training time: 1:50s
 Own evaluation: Less accurate than XGBoost model with 10x the loss.
3. Classification_TheodoreBeevers_NN-TensorFlow1.txt:
 Algorithm: TF Keras Sequential
 Pre-processing: Scaled input features using MinMaxScaler
 Key HP values: layers = (70(rectified_linear), 55(tanh), 90(tanh), 70(tanh)), eta = 0.0001, optimiser = Adam
 HP Optimisation: random search / trial and error. Plotting al vs train loss to get an idea of eta.
 Parameters in model: 50,570 (197.54 KB)
 Loss function and value on validation set: 0.12195 (Binary cross entropy)
 Training time: 9:49
 Own evaluation: Same features used from mlp classifier.
 Better loss and a higher accuracy. Better neural network for classification but still not as good as the decision tree.
4. Regression_TheodoreBeevers_RandomForestRegressor.txt:
 Algorithm: RandomForestRegressor
 HP Optimisation: Exhaustive search using GridSearchCV with k = 3.
 Key HP values: max_depth = 20, min_samples_split = 2, min_samples_leaf = 2, n_estimators = 200
 HP Optimisation: Exhaustive search with GridSearchCV with k = 3
 Loss function and value on validation set: MAE of relative deviation on val set: 0.3001

Training time: 190.68s
Own evaluation: Simple model with decent accuracy. Easy to implement.

5. Regression_TheodoreBeevers_NN-TensorFlow2.txt:
Algorithm: TF keras.Sequential
Preprocessing: Scaled using MinMaxScaler
Key HP values: layers = (40, 50, 10), activation = relu,
epochs = 130, eta 0.01. solver = Adam
HP Optimisation: trial and error
Parameters in model: 10,835 (42.33 KB)
Loss function and value on validation set: MAE = 0.2889
Feature selection: same features used as RFR
Training time: 180.74
Own evaluation: Improvement on the RandomForestRegressor
but still not as accurate as one would want.

6. Clustering_TheodoreBeevers_KMeans.txt:
Algorithm: KMeans
Preprocessing: Scaled using MinMaxScaler, ten variables
with highest variance chosen
Key HP values: k = 10
Own evaluation: Visualised with UMAP. Seems to split up
into categories but it's hard to gain any information from this.