# 10601 Machine Learning Project
# Milestone 3:

*Team<Triple Chocolate>:*

*HongMing Xu (hongminx)*

*Pan Tan (pant)*

*Menglei Liang (mengleil)*

**Did you receive any help whatsoever from anyone in solving this assignment?**

No. Our team work alone.

**Did you give any help whatsoever to anyone in solving this assignment?**

No. Our team work alone.

# Report Description:

**(1) Classifier Description:**

The classifiers we use: (described in Wikipedia)

1. **Random Forest:**

Random forests are an used for classification (and regression) that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes output by individual trees.

2. **SMO(Sequential Minimal Optimization):**

SMO searches through the feasible region of the dual problem and maximizes the objective function.

3. **LMT(Logistic Model Tree)**

Logistic model tree (LMT) is a classification model with an associated supervised training algorithm that combines logistic regression (LR) and decision tree learning.

4. **AdaboostM1(Adaptive Boosting)**

AdaboostM1 is an algorithm for constructing a "strong" classifier as linear combination of simple "weak" classifiers. Once the weak classifier has been received, AdaBoostM1 chooses a parameter intuitively measures the importance that it assigns to this classifier. To use the weak learner to form a highly accurate prediction rule by calling the weak learner repeatedly on different distributions over the training examples.

**(2) Parameter we varied:**

**For AdaboostM1:**

Parameter "numIterations" and "weightThreshold" seem to be quite sensitive to the number or internal cross-validation folds used, with the most robust performance occurring for fairly large values. After experiments, we fixed "numIterations" at 100, and "weightThreshold" to 4, which seemed to strike a good balance between compute time and robustness.

**For Random Forest:**

Parameter "numTrees" and "numFeatures" seem to be quite sensitive to the number or internal cross-validation folds used, with the most robust performance occurring for fairly large values. After experiments, we fixed "numIterations" at 14, and "weightThreshold" to 3, which seemed to strike a good balance between compute time and robustness.

The change of "seed" seems not affect at all.

**For SMO:**

Under the condition that setting kernel to be the same, the complexity parameter C affects the result. After experiments, we fixed C at 1.0, which seemed to strike a good balance between compute time and robustness.

**For LMT:**
Parameter "numBoostingIterations" and "weightTrimBeta" seem to be quite sensitive to the number or internal cross-validation folds used, with the most robust performance occurring for fairly large values. After experiments, we fixed this "numBoostingIterations" at -1, and "weightTrimBeta" to 0.45, which seemed to strike a good balance between compute time and robustness.

**(3) Final settings:**

The comparison between the results we got from ms2 and the improved results is as follows:

| Data | AdM1 Improved | AdM1 Original | RF Improved | RF Original | SMO Improved | SMO Original | LMT Improved | LMT Original |
|---|---|---|---|---|---|---|---|---|
| anneal | 2.7059 | 2.7059 | 0.1765 | 0.3529 | 0.5294 | 0.5882 | 0.1764 | 0.2941 |
| audiology | 1.6818 | 1.6818 | 0.6818 | 0.7273 | 0.6364 | 0.6364 | 0.6818 | 0.6818 |
| autos | 1.0789 | 1.0789 | 0.8421 | 0.9474 | 0.9737 | 0.9737 | 1.0526 | 1.0526 |
| balance-scale | 1.3485 | 1.1667 | 0.9091 | 0.9545 | 0.4394 | 0.4091 | 0.4697 | 0.4545 |
| breast-cancer | 0.9655 | 0.9655 | 0.9310 | 1.0000 | 0.9310 | 0.8621 | 0.7241 | 0.7241 |
| colic | 1.0435 | 0.9565 | 0.8696 | 0.8261 | 1.0000 | 1.0000 | 0.826 | 1.0435 |
| credit-a | 0.7024 | 0.7024 | 0.6905 | 0.7024 | 0.7262 | 0.7262 | 0.6428 | 0.6786 |
| diabetes | 1.1290 | 1.1613 | 1.1290 | 1.2097 | 0.9839 | 0.9839 | 1.0484 | 0.9677 |
| glass | 0.8235 | 0.8235 | 0.4902 | 0.4314 | 0.5294 | 0.5490 | 0.4706 | 0.4510 |
| heart-c | 0.8333 | 0.6667 | 1.0000 | 0.7778 | 0.7222 | 0.8889 | 0.8333 | 0.8333 |
| hepatitis | 1.0000 | 1.0000 | 0.8333 | 0.8333 | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| hypothyroid | 0.8356 | 1.3836 | 0.2329 | 0.2192 | 1.1918 | 1.2603 | 0.0959 | 0.0959 |
| **Average** | **1.179** | **1.1911** | **0.7322** | **0.7485** | **0.8053** | **0.8232** | **0.6685** | **0.6898** |
| **Max** | **2.7059** | **2.7059** | **1.129** | **1.2097** | **1.1918** | **1.2603** | **1.0526** | **1.0526** |

From the table above, we can see that all improved classifiers' Average value decrease. Both the Average value and the Max value of LMT are the smallest, so we choose LMT classifier as the final one.

**(4) Summary:**
Among the classifiers we used, the best robustness was obtained by LMT, which was naturally quite robust without parameter tuning. Additionally, by using nested Cross Validation (CV) to set the three most critical values of LMT, and changing two default parameter setting, we were able to improve its average robustness from 0.6898 to 0.6685, but its maximum robustness remains 1.0526 . Actually, since LMT performances quite well in default parameters and the best parameter we get from CV does not mean the best for test data (bias test), in some special case, the robustness would get worse a little.