**kernelSVM**

Before we just ran through the kernelized SVM, we normalized the dataset for both test/train, and cv sets.

* 1 vs all

SVM Multinomial Classification Train Accuracy :: 0.602323503128

SVM Multinomial Classification Test Accuracy :: 0.560416666667

SVM Multinomial CV-prediction error rate :: [0.46296296 0.58385093 0.56521739 0.4375 0.5375 0.68125

0.6125 0.575 0.60759494 0.56687898]

SVM Multinomial CV-prediction error mean :: 0.563025520354

SVM Multinomial CV-prediction error variance :: 0.00454286478151

* 1 vs 1

**<Linear>**

SVM Multinomial Classification Train Accuracy :: 0.597855227882

SVM Multinomial Classification Test Accuracy :: 0.604166666667

SVM Multinomial CV-prediction error rate :: [0.48148148 0.59006211 0.55900621 0.48125 0.53125 0.68125 0.6125 0.56875 0.63291139 0.58598726]

SVM Multinomial CV-prediction error mean :: 0.572444845801

SVM Multinomial CV-prediction error variance :: 0.00360750429466

**<Polynomial with degree=2, coef0=10>**

SVM Multinomial Classification Train Accuracy :: 0.630026809651

SVM Multinomial Classification Test Accuracy :: 0.6

SVM Multinomial CV-prediction error rate :: [0.50617284 0.54658385 0.52795031 0.53125 0.55 0.73125 0.6125 0.55 0.60759494 0.57324841]

SVM Multinomial CV-prediction error mean :: 0.573655034535

SVM Multinomial CV-prediction error variance :: 0.00377901124798

**<Polynomial with degree=3, coef0=10>**

SVM Multinomial Classification Train Accuracy :: 0.742627345845

SVM Multinomial Classification Test Accuracy :: 0.610416666667

SVM Multinomial CV-prediction error rate :: [0.50617284 0.55279503 0.54037267 0.46875 0.49375 0.69375

0.59375 0.55625 0.61392405 0.58598726]

SVM Multinomial CV-prediction error mean :: 0.560550185315

SVM Multinomial CV-prediction error variance :: 0.00386724332753

**<RBF with gamma=4>**

SVM Multinomial Classification Train Accuracy :: 0.998212689902

SVM Multinomial Classification Test Accuracy :: 0.545833333333

SVM Multinomial CV-prediction error rate :: [0.49382716 0.44720497 0.42857143 0.45 0.45625 0.43125

0.41875 0.4625 0.46835443 0.42038217]

SVM Multinomial CV-prediction error mean :: 0.447709015399

SVM Multinomial CV-prediction error variance :: 0.000507286849343

**<RBF with gamma=3>**

SVM Multinomial Classification Train Accuracy :: 0.992850759607

SVM Multinomial Classification Test Accuracy :: 0.60625

SVM Multinomial CV-prediction error rate :: [0.51234568 0.45341615 0.40993789 0.41875 0.4875 0.43125

0.425 0.51875 0.44936709 0.49044586]

SVM Multinomial CV-prediction error mean :: 0.459676266476

SVM Multinomial CV-prediction error variance :: 0.00142895308965

**<RBF with gamma=1>**

SVM Multinomial Classification Train Accuracy :: 0.948168007149

SVM Multinomial Classification Test Accuracy :: 0.614583333333

SVM Multinomial CV-prediction error rate :: [0.47530864 0.55279503 0.49689441 0.4625 0.5375 0.56875

0.53125 0.4875 0.58227848 0.60509554]

SVM Multinomial CV-prediction error mean :: 0.529987210538

SVM Multinomial CV-prediction error variance :: 0.00209009914564

**<RBF with gamma=10>**

SVM Multinomial Classification Train Accuracy :: 1.0

SVM Multinomial Classification Test Accuracy :: 0.564583333333

SVM Multinomial CV-prediction error rate :: [0.44444444 0.44099379 0.42857143 0.4125 0.4375 0.45

0.4 0.4625 0.42405063 0.42675159]

SVM Multinomial CV-prediction error mean :: 0.43273118871

SVM Multinomial CV-prediction error variance :: 0.000302164938751

**<RBF with gamma=0.1>**

SVM Multinomial Classification Train Accuracy :: 0.670241286863

SVM Multinomial Classification Test Accuracy :: 0.654166666667

SVM Multinomial CV-prediction error rate :: [0.48765432 0.62732919 0.53416149 0.53125 0.59375 0.725

0.63125 0.5875 0.62658228 0.60509554]

SVM Multinomial CV-prediction error mean :: 0.59495728241

SVM Multinomial CV-prediction error variance :: 0.00397048752315

**<Sigmoid with coef0=20>**

SVM Multinomial Classification Train Accuracy :: 0.415549597855

SVM Multinomial Classification Test Accuracy :: 0.45

SVM Multinomial CV-prediction error rate :: [0.42592593 0.42236025 0.42236025 0.425 0.425 0.425

0.425 0.425 0.43037975 0.43312102]

SVM Multinomial CV-prediction error mean :: 0.425914718876

SVM Multinomial CV-prediction error variance :: 1.01319436483e-05

**<Sigmoid with coef0=10>**

SVM Multinomial Classification Train Accuracy :: 0.430741733691

SVM Multinomial Classification Test Accuracy :: 0.414583333333

SVM Multinomial CV-prediction error rate :: [0.42592593 0.42236025 0.42236025 0.425 0.425 0.425

0.425 0.425 0.43037975 0.43312102]

SVM Multinomial CV-prediction error mean :: 0.425914718876

SVM Multinomial CV-prediction error variance :: 1.01319436483e-05

**<Sigmoid with coef0=1>**

SVM Multinomial Classification Train Accuracy :: 0.419124218052

SVM Multinomial Classification Test Accuracy :: 0.433333333333

SVM Multinomial CV-prediction error rate :: [0.43209877 0.38509317 0.42857143 0.44375 0.4625 0.55

0.53125 0.50625 0.51265823 0.42675159]

SVM Multinomial CV-prediction error mean :: 0.467892318191

SVM Multinomial CV-prediction error variance :: 0.0026218663724

**<Sigmoid with coef0=0.1>**

SVM Multinomial Classification Train Accuracy :: 0.485254691689

SVM Multinomial Classification Test Accuracy :: 0.5

SVM Multinomial CV-prediction error rate :: [0.42592593 0.50310559 0.45962733 0.43125 0.46875 0.56875

0.44375 0.46875 0.61392405 0.44585987]

SVM Multinomial CV-prediction error mean :: 0.482969276842

SVM Multinomial CV-prediction error variance :: 0.00347062655813

**<Sigmoid with coef0=0.01>**

SVM Multinomial Classification Train Accuracy :: 0.461126005362

SVM Multinomial Classification Test Accuracy :: 0.504166666667

SVM Multinomial CV-prediction error rate :: [0.45679012 0.52795031 0.49689441 0.4375 0.53125 0.55625

0.46875 0.475 0.48734177 0.48407643]

SVM Multinomial CV-prediction error mean :: 0.492180304923

SVM Multinomial CV-prediction error variance :: 0.00121087098842

Thus, through the observation, according to train/test accuracy and cross-validation result, linear, polynomial, and RBF kernels all produces similar results if the parameters are set properly, while sigmoid kernel seemed inefficient. Also, 1 vs 1 method seemed better both in performance and the time spent for calculation.

**Model Evaluation: alpha = 0.005 (Very Strict Test)**

For the model evaluation, we first ran 10 rounds of 10-fold cross validation for two different methods. Then, we have 100 (10\*10) accuracy values for both methods. We used these and conducted 2-independent sample difference of mean Welch’s T-test because:

1. we are not sure about the population for both samples.

2. we are not sure about population variances for both samples.

3. We are not sure whether two populations (for respective samples) have the same population variance.

The following are the results of the different model evaluations.

**kernelSVM vs adaBoost(n=10)**

*For adaBoost, we used the parameter of 10, considering both heuristic efficiency and the time consumed for calculation.*

New degree of freedom: 192.98644434745617

Test T-Score: 8.303210942193562

Comparable T-score: 2.60154434562431

Significantly, kernelSVM.py is better than adaBoost.py / kernelSVM.py mean accuracy: 0.594957282 / adaBoost.py mean accuracy: 0.514168987

**kernelSVM vs decisionTree**

New degree of freedom: 144.6621397054777

Test T-Score: 18.5734545545312

Comparable T-score: 2.6102419259465246

Significantly, kernelSVM.py is better than decisionTree.py / kernelSVM.py mean accuracy: 0.594957282 / decisionTree.py mean accuracy: 0.4644024291

**kernelSVM vs randomForest(n=8)**

*For randomForest, we used the parameter of 8, considering both heuristic efficiency and the time consumed for calculation.*

New degree of freedom: 170.8970672086863

Test T-Score: 8.152562194950022

Comparable T-score: 2.6049033087764633

Significantly, kernelSVM.py is better than randomForest.py / kernelSVM.py mean accuracy: 0.594957282 / randomForest.py mean accuracy: 0.5335187611

**kernelSVM vs MultinomialLogRegression**

New degree of freedom: 197.8093814878416

Test T-Score: 0.41469629349765436

Comparable T-score: 2.6009116454295658

Statistically, no difference detected. But in this sample, kernelSVM.pyis slightly better. kernelSVM.py mean accuracy: 0.594957282 / MultinomialLogRegression.py mean accuracy: 0.591203105

**kernelSVM vs GP**

*For GP, we used the RBF kernel with gamma value 1, considering both heuristic efficiency and the time consumed for calculation. Matern kernel produces a slightly better (better in terms of 0.002) result, but it took too much time and made Python run out of the RAM.*

New degree of freedom: 162.66801358426738

Test T-Score: 3.4043914446537302

Comparable T-score: 2.606390449865822

Significantly, kernelSVM.py is better than GP.py / kernelSVM.py mean accuracy: 0.594957282 / GP.py mean accuracy: 0.5699018470000001

Therefore, in conclusion, it seemed that kernelSVM will be the best method out of all the things that we used so far. However, the CV average of 0.594957282 is still low, so we need to figure out of a better method.