02-750 Final Project

Carnegie Mellon University

Clinical Triage with An Active Feature Batch Algorithm

Author:
Handi Xu
Xiangzhu Long

Supervisor:
Prof. Christopher Langmead

December 15, 2015

1 Introduction

Features provide kinds of valuable information and are the key factors in predictive models. In reality, features acquisition is very tricky and costly. For example, it will spend a lot to acquire your life characteristics at hopital, and it is common that the customers refuse to provide more information that the store wants. One of the most common used solution is active feature acquisition, which acquires features incrementally in order to improve the predictive model with possible lowest cost.

The emergency care is one of the most expensive expense in healthcare in the United States. The main reason is the standard care process allow all the patients to take the ICU for 4 - 5 days before serious complications. However, the real data analyzed by J.-L. Vincent et al. told us that only 30% of these patients need such care, while the other 70% actually can be accommodated by normal hospital beds [1]. Therefore, it will save amount of money by identifying the patients with low risk. There are four tests to predict whether the patient needs the ICU, and each of them have different cost. To save cost at the optimal extent, the patient is suppsed to take one or some of the tests with the minimul expense on the condition of being confident to tell the result.

Our project is aim to build an accurate triage model on classifying the patients into two categories: those who need the Intensive Care Unit (ICU) and those who do not, with a possible optimal cost-efficient set of tests by implementing active feature value acquisition.

2 Background and Related Work

Patients are asked to take different tests with different expense. These tests could be used to determine whether the patient need to take ICU. The cost of different tests varies drastically, which leads to different expense by taking different tests. To minimize the cost while correctly classifying the patient, we need a mechanism to decide which tests the patient need take. If taking the combination of results from different test as features and whether the patient need ICU treatment as class value, this problem is addressed as active feature value acquisition (AFA) in academia. Melville et al[2] proposed a active feature value acquisition framework, however, this framework can not be applied directly to our task for the following reasons:

1. They only utilize active feature value acquisition during training,
2. It can only be applied on pool-based tasks,
3. We have no feature for every coming instance while they have some part of features,
4. Their framework must acquire all missing features once it decided the instance has a high utility score.

Sheng and Ling [3] proposed a active feature value acquisition algorithm during testing, called Sequential Batch Test Algorithm. The issue they addressed are very similar to ours, which is medical diagnose. However, they mainly focus on balancing the cost of testing, waiting time, and errors, while we only concern about testing cost and accuracy. Secondly, their approach uses the same model which can handle missing values for every tests. While we propose to use different models for features from different tests.

3 Dataset

The oringinal dataset[4] come from 10,000 patients labeled with the actual necessary of the ICU as shown in Figure 1.

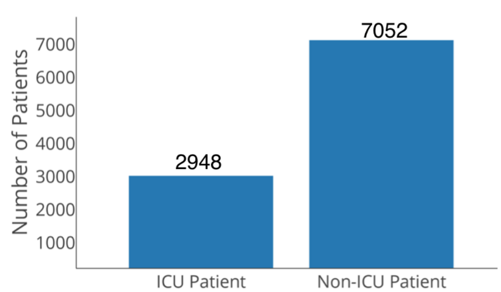
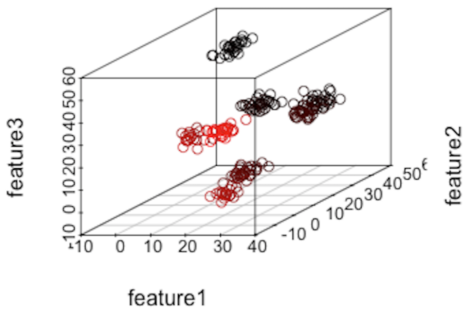
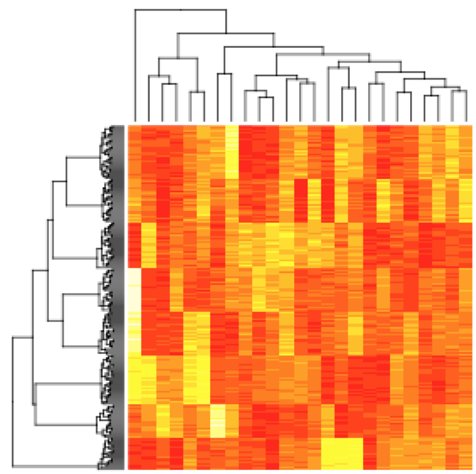


Figure 1: Numbers of ICU Patients and Non-ICU Patients.

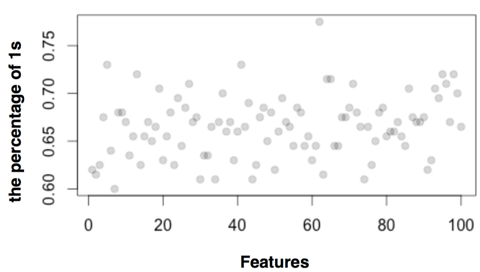
There are four tests, each of which have different cost and produce different features: an inexpensive proteomic panel, expensive proteomic panel, genotyping, and imaging. These four tests produce 130 features in all. Test 1 generate 3 numerical features, Test 2 generate 25 numerical features, Test 3 result in 100 binary classification features, and Test 4 result in 2 binary classification features. The feature visulizations of different test are shown as Figure 2.



(a) Test 1



(b) Test 2



(c) Test 3

feature	# 1	# 0
1	48	154
2	48	154

(d) Test 4

Figure 2: Feature Visulizations

4 Methodology

We uses an active-feature value acquisition learner with batch online to build the classifier with stream-data. Our algorithm, as shown in Algorithm 1, is based on the active feature-value acquisitions proposed by P. Melville et al.[5] and M. Saar-Tsechansky et al.[2]. First we took in 2% percent of data for training. We built 5 models based on different features obtained by different tests. And across the evaluation and cost of these models, we decide to excute which test for the online observaion by analyzing its cheapest testing features.

Algorithm 1: Active-Feature Learner.

1 Get 2% batch of data as training data.
2 Build 5 classifier about whether need the ICU or not on the training data with different features collected from 4 individual tests and tests in all.
3 Set the classifier with the cheapest feature cost as cheap classifier, and the one with the best performance ($\frac{accuracy}{avg(cost)}$) as optimal classifier.
4 Build a model selecter on the same training data, the label is whether the cheap model predict right or not.
5 For every online observation, first get their features by cheap test.
6 Check whether it need to get the optimal test by using the model selecter.
7   If yes, get optimal features and then implement optimal classifier;
8   If not, use the cheap classifier.
9 Optimize the classifier in addition of this observation every a certain number of observations.
10 Iterate 5-9 until no more observation.

4.1 Algotithm Analysis

There are mainly two goals in this project:

1. Classifier the patient to two categories: ICU or non-ICU;
2. Reduce the test costs;
3. Garantee the high recall of the model.

The most particular and unignorable thing in this project is that the recall [6] of the model is more important than the accuracy. Because the result of misclassifying a patient who does not need ICU to ICU need is much more cost, while the result of misclassifying a patient who needs ICU to non-ICU need is very serious, even leads to death. Thus, for the classifier performance, we give the recall higher weight when evaluating.

4.2 Training Part

4.2.1 Classifer

First we build 5 classifers on ICU patient and non-ICU patient based on different features collected from 4 individual tests and tests in all. And then we first select the one with the cheapest feature cost and then another one with highest performance in the rest classifers. Here the performance is defined as the ratio of accuracy and average cost of the model. The algorithm is shown as Algorithm 2.

Algorithm 2: Training.

```
Input: online instances, 5 tests
Output: cheap classifier, optimal classifier, model selector
for each instance of 2% batches do
    training_data[i] = readIn(instances)
    training_features[i] = testFeatures(5Tests, instances)
    training_labels = labels(instances)
for i: 1-5 do
    models[i] = trainModel_SVM(differentFeatures)
    evaluation[i] = getEvaluation(models[i])
    cost[i] = calCost(test)
cheap_classifier = minCostModel(cost[i], evaluation[i] ¿ 70_
optimal_classifier = maxEvalModel(evaluation[i])
for each data in training_data[i] do
    if cheap_classifier == training_label then
        selector_label[i] = 1
    else
        selector_label[i] = 0
model_selector = train_TreeBagger(cheap_classifier, training_data)
return cheap_classifier, optimal_classifier, model_selector
```

4.2.2 Model Selector

In order to decide whether the cheap classifer is confident enough to classify the observation or not, in other words, we need to decide whether the more expensive test in the optimal model needs to be applied or not, we build a model selector based on the cheap features, and categories the data which the cheap model predicted correctly or not.

For example, for a new patient with the cheap features, we compare with the features of those patients who were predicted correctly by cheap classifier, and predict whether the cheap classifier is good enough to predict this patient with high accuracy and recall.

4.3 Testing Part

For the instances for testing, we implement the algorithm discribed in Algorithm 3.

5 Result and Evaluation

Baseline results are shown in Figure 3 . If the model is based only on test 1, the model will classify all instances to 0, and the accuracy is low. All other baseline models can achieve 100% accuracy, with pretty high costs. The best one is model 2, because it has a lower cost, the accuracy per cost is 1/250.

The result of our AFA model with different size of batches to optimize model, is shown in Figure 4. In addition to achieving high accuracy, we reduced the accuracy per cost to 1/83.4.

6 Discussion

For baseline experiments, the result shows that strategy 1 (test 1) has the worst performance, while strategy 2,3,4 have similar performances. Therefore, it seem it doesn't make sense to do test 3, and test 4. From the performance curve, we can see that when batch size is 50, the performance is worse than baseline strategy 2. But when the batch size exceeds 100, the performance increases to as good as strategy 2 and with a much lower average cost. (94.9 compared to 250). As the batch size increase to 200, the performance doesn't increase a lot compared to the performance from batch size 100.

Algorithm 3: Online Testing.

```
Input: online instances, cheap classifier, optimal classifier, model
       selector, cheapest test, optimal test
Output: categories that classify the instances
for each instance do
    cheap_features ← cheap_TEST(instance)
    need_more_features ← model_selector(cheap_features)
    if need_more_features then
        more_features ← optimal_TEST(instance)
        categories[i] = optimal_classifier(more_features)
        batch_optimal += 1
        if batch == NUM then
            optimal_classifier =
            training_wt._extraNumData(optimal_classifier);
            batch_optimal = 0;
    else
        categories[i] = cheap_classifier(more_features)
        batch_cheap += 1
        if batch == NUM then
            optimal_classifier =
            training_wt._extraNumData(cheap_classifier);
            batch_cheap = 0;
return categories
```

Table 1: BaseLine

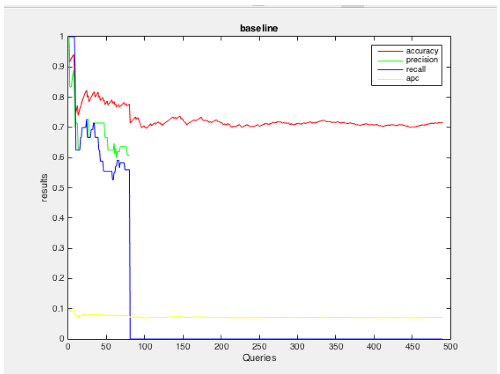
Num of Batch	50	100	200	400
Average Cost	61.3	94.9	114.505	149.705

Table 2: AFA

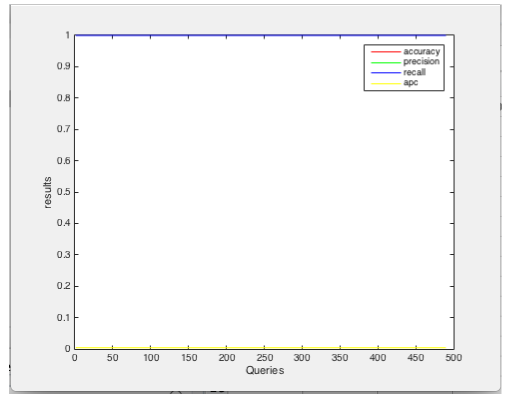
Strategy	1	2	3	4
Average Cost	10	250	500	1000

7 Conclusion and Future Work

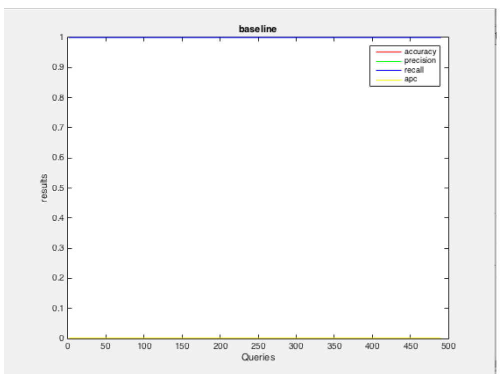
In this paper, we use streaming machine learning to model medical tests by which doctors decide whether the patient need be treated in ICU. In order



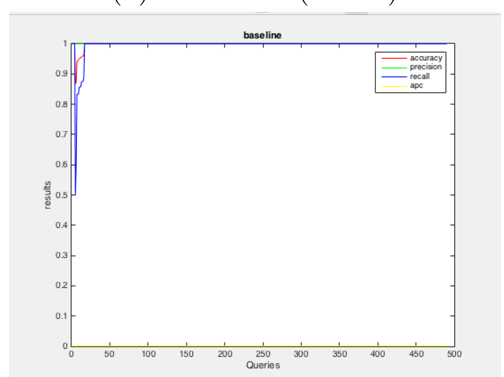
(a) Baseline 1 (Test 1)



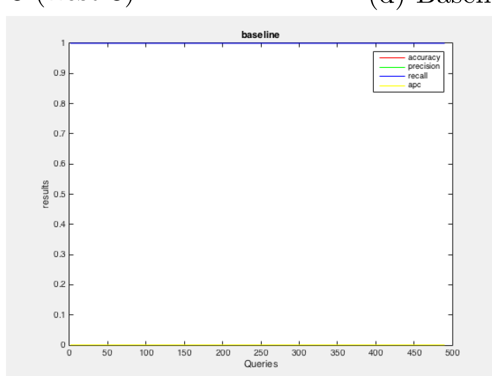
(b) Baseline 2 (Test 2)



(c) Baseline 3 (Test 3)

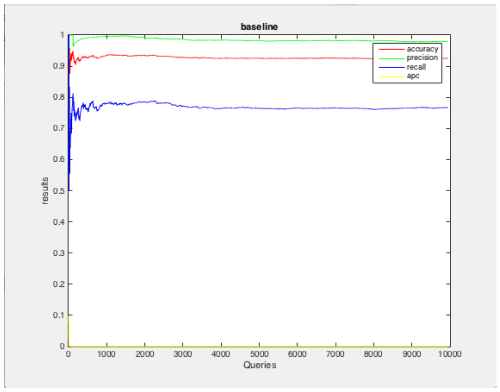


(d) Baseline 4 (Test 4)

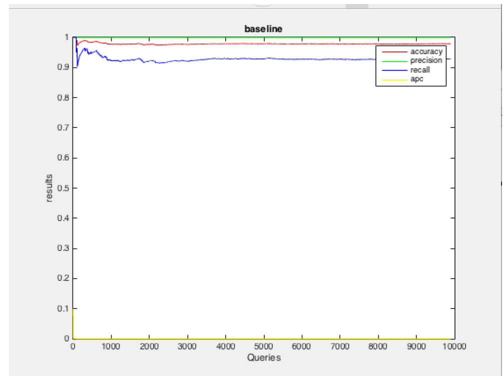


(e) Baseline 5 (Test 5)

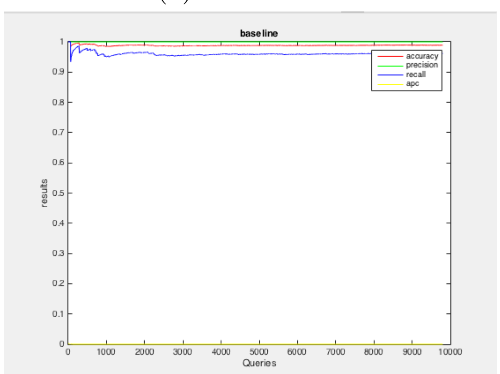
Figure 3: Baselines Performance



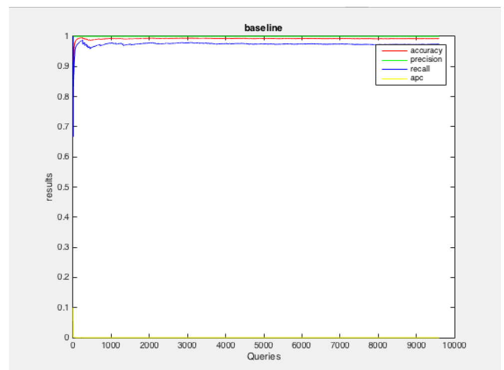
(a) NUM = 50



(b) NUM = 100



(c) NUM = 200



(d) NUM = 400

Figure 4: APA Performance

to increase the accuracy as well as decrease the cost, we propose the model selector to decide whether we need to extra test in addition to the least expensive test. From the discussion above, it can be safely concluded that our active feature acquisition algorithm significantly increase the accuracy with a reasonable cost increase. However, the dataset we played with is not real dataset, thus we still cannot guarantee it could work in real world.

In our future work, we plan to acquire real dataset from hospitals and apply our approaches on these real data. Moreover, we need do feature analysis and feature engineering on the dataset to help us pick a better base leaner.

8 Acknowledgment

The authors would like to thank teacher assistants Linglin Huang, and Bangrui Liu for helping with the references and algorithm advices.

References

[1] J.-L. Vincent and G. D. Rubenfeld, "Does intermediate care improve patient outcomes or reduce costs?," *Critical care*, vol. 19, no. 1, p. 89, 2015.

[2] M. Saar-Tsechansky, P. Melville, and F. Provost, "Active feature-value acquisition," *Management Science*, vol. 55, no. 4, pp. 664–684, 2009.

[3] V. S. Sheng and C. X. Ling, "Feature value acquisition in testing: a sequential batch test algorithm," in *Proceedings of the 23rd international conference on Machine learning*, pp. 809–816, ACM, 2006.

[4] C. J. Langmead, "Drug screening project," 1999.

[5] P. Melville, M. Saar-Tsechansky, F. Provost, and R. Mooney, "Active feature-value acquisition for classifier induction," in *Data Mining, 2004. ICDM'04. Fourth IEEE International Conference on*, pp. 483–486, IEEE, 2004.

[6] Wikipedia, "Precision and recall — wikipedia, the free encyclopedia," 2015. [Online; accessed 24-November-2015].