Московский Государственный Университет имени М.В. Ломоносова

Факультет Вычислительной Математики и Кибернетики

Кафедра Математических Методов Прогнозирования

Задание по практикуму на ЭВМ:

Решение реальной задачи

«Topical Classification of Biomedical Research Papers»

Выполнила студентка 317 группы Любимцева Мария

2012 год

I. Постановка задачи

Имеем дело с задачей классификации медицинских статей по рубрикам. Каждая статья может относится более, чем к одной рубрике.

Даны обучающая матрица и ответы для нее, а также тестовая матрица.

Обучающая матрица имеет большой размер (10000 * 25640) и сильно разрежена. В качестве объектов выступают статьи. Что скрывается под признаками, нам неизвестно. Известно лишь, что они принимают целые неотрицательные значения из диапазона 0..1000.

Для обучающей матрицы для каждой статьи нам известен список рубрик (набор чисел 1..83), к которым она относится.

Необходимо найти список рубрик для объектов из тестовой матрицы, которая имеет такой же размер, как и обучающая.

II. Ход решения

1) Предобработка данных

1.1) Уменьшение количества признаков

Изначально были удалены бесполезные признаки, которые на всех объектах принимали нулевые значения. После этого количество признаков сократилось до 23917 штук.

Затем было сделано предположение, что у нас могут быть похожие признаки, которые принимают почти одинаковые значения на объектах. Чтобы их обнаружить, была использована функция MyFeatureSelection (код ниже). Параметр threshold подбирался в цикле по принципу, можем ли мы, используя полученное множество признаков, добиться значения F-меры (при работе обычного SVM в данном случае) не хуже, чем на предыдущем шаге цикла.

Удалось добиться сокращения количества признаков до 15069 штук при параметре threshold = 3.

```
function fs = MyFeatureSelection(X, threshold)
% X - обучающая матрица
% threshold - порог для оценивания сходства признаков
% fs - полученный набор признаков, после удаления похожих

    % изначально берем все признаки
    fs = 1:size(X,2);

    %нормируем матрицу по максимуму столбца
    X = bsxfun(@rdivide, X, max(X));

    for j = 1:(size(X,2)+1)
        % считаем меру отличия всех признаков от j-ого -
        % сумма модулей разности значений по всем объектам
        s = sum(abs(bsxfun(@minus, X, X(:,j))));

        % исключаем удаление самого j-ого признака
        s(j) = threshold;

        % удаляем признаки, которые отличаются от j-ого
        % меньше, чем на threshold
        X(:, s < threshold) = [];
        fs(:, s < threshold) = [];

        % условие досрочного выхода
        if (j >= size(X,2))
            return;
        end
    end
end
```

Не удалось использовать уже реализованные серьезные методы, так как возникала преследовавшая меня проблема «Out of memory».

1.2) Нормировка данных

Было опробовано несколько видов нормировки:

a) по сумме элементов каждого из столбцов/строки;

b) по максимуму столбца/строки;

c) по максимуму столбца, затем строк;

d) по среднему значению столбцов/строк;

e) по среднему значению ненулевых элементов столбцов;

f) X = 1 / (1 + exp(-(X – mean(по столбцам)) / std(по столбцам));

Результаты одного из экспериментов по выбору наилучшей нормировки данных. В данном случае использовались SVM с параметрами по умолчанию. Столбец w в таблице – без нормировки.

	a	b	c	d	e	f	w
F-мера	0.275/0.25	0.417/0.423	0.424	0.344/0.402	0.391	0.389	0.405

2) Использованные алгоритмы

Привожу описание только SVM, так как все остальное потерпело крах, наткнувшись на «Out of memory».

Строился классификатор для каждой рубрики, то есть решалась задача классификации на два непересекающихся класса – относится к рубрике или нет.

Был осуществлен подбор параметра C (мноого экспериментов) путем метода скользящего контроля, в котором использовалось пять разбиений в соотношении обучающая выборка к тестовой 9:1.

Сначала C подбиралось одним для всех классификаторов сразу и максимальный полученный предварительный результат был равен 0.455. Затем были сделаны попытки подобрать для каждого свое, но по результатам экспериментов получалось, что от выбора C вообще ничего не зависело, поэтому я расстроилась.

Не получилось придумать, как использовать оценки, выдаваемые SVM-ом, поэтому использовались только выдаваемые им уже метки принадлежности классам.

3) Финальное решение

Был произведено описанное выше удаление признаков, матрицы были пронормированы по максимуму строк, затем применены 83 SVM с одинаковым параметром C = 0.14.

Preliminary result: 0.455

Final result: 0.463

4) Могу предоставить

Могу предоставить использованные в ходе работы коды, файл с экспериментами по подбору C.

III. Советы новичкам

1) Пытайтесь уменьшить размерность задачи;

2) Не пытаться работать только с одним методом решения только потому, что кто-то именно с помощью него добился отличных результатов: совсем не факт, что это удастся вам, а время на «ну чем я хуже, ну еще вот это попробую и тогда точно за другое возьмусь» будет потрачено;

3) По возможности используйте уже реализованные методы;

4) Следите за временем;

IV. Про себя

Для себя я поняла, что:

- не стоит зацикливаться на чем-то одном при решении задачи, лучше попробовать что-то иное;

- решение реальной задачи слишком затягивает и поэтому времени всегда будет мало;

- у такой задачи может быть простое и красивое решение, но не мое☺

Если бы, то я:

- Поменьше смотрела бы на турнирную таблицу;

- Начала работу чуть раньше;

- Поработала бы более тщательно с kNN и отбором признаков;

О задании

Мне понравилось, так как задача с реальными данными и конкурс настоящий. Я была бы не против еще одного подобного задания, только хотелось бы немного иного формата обсуждения: помимо письменного еще и устного.

V. Первый этап

На первом этапе я не принесла никакой пользы, лишь исправила ошибку в коде функции для создания файла для отправки в систему, надеюсь, хоть кому-то было это пригодилось.

Мне помогли:

1) Андрей Остапец – нормировки, пример работы с Liblinear;

2) Петр Ромов – преобразование данных с сайта в m-файл;

3) Евгений Нижибицкий – код удаления ненулевых столбцов;

4) Дмитрий Кондрашкин – код для вычисления F-меры.