# Отчет о проделанной работе по практикуму

Гавриков Михаил

**MSU**

Москва, 2012

# Постановка задачи

**Задача**

Необходимо найти решение задачи, так чтобы оно показывало хороший результат и плюс описывало каким-то образом данные.

# Ход решения

## Основная Мысль

Мне не хотелось использовать стандартные решения, обычный **kNN** или линейный классификатор, тем более их все уже попробовали и просто маневрировали параметрами, метриками итп.

## Проблемы

У меня **32-x** битная версия Windows и 3gb оперативной памяти. Плюс ещё сколько не пытался, максимальный размер одного массива был не выше 1043 mb. Эту информацию я получил из команды **memory**. То есть всего доступно для всех массивов 2200 mb, но для одного 1043mb.

# Ход решения

## Метод главных компонент

Прежде всего мне хотелось уменьшить пространство. И я решил использовать метод главных компонент. Проблемы которые возникли при таком решении: во первых нельзя применить ко всему 26000-мерному признакову пространству. Т.к. необходимо создавать матрицу(не sparse, а обычную) 26000 на 26000, (максимальный размер матрицы 10000 на 10000 на моём компьютере). Тогда я решил сделать 83 таких пространства. Причем выкидывать признаки, которые нулевые у всех объектов принадлежащих заданному классу.

# Ход решения

## Метод главных компонент

Мне хватило ума сузить пространство для 3-х и 10-и мерного. Судя по тому что вы рассказали в понедельник(02.04.2012) надо было сделать что-то типа 700, но увы. В чём проблема: считается матрица преобразований долго. Результат меня ужаснул: бывают объекты принадлежащие классу и не принадлежащие классу попавшие почти в одну точку, с точностью до 8-ой цифры. kNN выдаёт ужасно плохой результат. Что-то типа .151, для $k = 15$ и метрики стандартной евклидовой. Есть смысл использовать именно её, а не какие-нибудь косинусные метрики или ещё того хуже, т.к. пространство получаемое выглядит вполне неплохо.

Пробовал следующие варианты строить уменьшенное пространство по объектам из заданного класса, по всем объектам из обучающей выборки, отсеив признаки не из этого класса, по обучающей и тестовой выборке, отсеив признаки не из этого класса.

После этого я решил, что это не очень хорошая идея и перекинулся на другое предположение.

# Ход решения

## Игра с признаками

Посмотрев на признаки, я заметил, что каждый признак принимает далеко не всю тысячу значений(диапазон от 0 до 1000), а только лишь какие-то выборочные значения. В этот момент у меня родилось несколько предположений.

## Первое

А что если для каждого класса сохранить все возможные значения признаков, и потом смотреть близость объекта из тестовой выборки не как близость к объектам класса, а как близость к классу. И к примеру если объект совпадает по признакам из выбранного класса(это те признаки которые не нулевые во всех объектах, из обучающей выборки, принадлежащих заданному классу.) с какой-то частью классовых признаков, то окей: объект принадлежит классу. Остаётся вопрос о порогах, ясно, что порог зависит от количества объектов из этого класса, от количества объектов с ненулевым этим признаком и от количества объектов не принадлежащих этому классу с ненулевым этим признаком.

То есть в какой-то степени это тоже метрический алгоритм, но достаточно хитрый по сравнению с теми, что мы рассматривали.

# Ход решения. Игра с признаками ч.2

## просто закономерности

- Есть признаки, которые выражены только у объектов принадлежащих заданному классу.

## Нулевое

Но перед тем как попробовать первое, я попробовал среднее и дисперсию по одному признаку из одного класса, попробовал повосстанавливать смесь распределений. Но, увы хороших результатов на первых тестах не получил, плюс ещё такой метод подразумевает очень большую независимость и между классами(хотя она неявно появляется, это никак не используется) и между признаками(с учётом разбиения по классам).

# Последний день Помпеи

Я решил, что как-то мой результат оставляет желать лучшего и надо сделать что-то из проделанного другими ребятами, но получивших получше результат: метрические алгоритмы и линейные классификаторы. Первое честно написал=) Второе не хотелось честно писать и решил установить 2012 matlab, в котором как выяснилось есть и kNN, и SVM, и даже решающие деревья. Но не тут-то было ограничения по памяти дали о себе знать....ни один из них не работает со sparse-матрицами. А это значит, что никакого обучения на 40, 44, да и более того первый раз он упал на 6 классе, не будет. Тогда я решил разбить множество признаков на какое-то количество частей(пересекающихся) и построить по ним композицию классификаторов, но не тут-то было за один день он такое не может посчитать, как выяснилось. За 8 часов работы он до 15-го класса дошёл.

## Что могу предоставить

- Код, по первому требованию
- Результат на любом промежуточном этапе, через некоторое время после требования.

## Советы новичкам(**НЕ** отсортировано в порядки значимости)

- Тратить больше времени на решение задачи.
- Сложность решения не всегда возрастает с качеством результата.
- Оперативная Память, вот где сила.

## Что узнал нового

- Существуют проблемы с памятью.
- Язык MatLab очень удобен и прост.
- MatLab 2012 обладает некоторым набором функций для datamining‘а.
- Если непонятна суть данных, то лучше её искать, когда будет хороший результат.
- Если for заменить на cellfun, то даже с учётом перевода матрицы в массив ячеек, cellfun работает быстрее.

## Помощь

Мне помогли Петя(описал как надо считывать файл) и Ильдар с Марией(в каком формате отправлять файл).

# Спасибо за внимание

В тексте присутствует много эмоциональных высказываний, по факту, не несущих никакой смысловой нагрузки, будьте бдительны.