На занятии мы обсуждали различные классические методы классификации и подбор порога с помощью ROC-кривых. Были разобраны примеры `iris`, `Smarket`, `Default`, `banknote`. `Default` и `Smarket` находятся в пакете `ISLR`, `banknote` в архиве, `iris` — стандартный датасет `R`

1 Домашнее задание

У меня попросили альтернативную задачу на линейную регрессию, их есть у меня.

Задание 1.1. Рассмотрим данные `concrete`. Описание и сами данные в архиве, кратко суть. Данные по прочности разных сортов бетона в зависимости от пропорций компонент и времени залития. Надо построить регрессию на прочность.

Предлагается использовать линейную регрессию и понять, что и как влияет на качество бетона. Обратите внимание, что параметры между собой зависимы, поэтому коэффициенты следует интерепретировать аккуратнее. Рекомендую построить одномерные графики зависимостей прочности от всего остального и попробовать определить характер. Возможно, что зависимость будет немонотонная, тогда стоит ввести дополнительный фактор “больше ли значение предиктора некоторого порога”. Некоторые предикторы могут оказаться не совсем непрерывными, возможно, что их стоит рассмотреть как факторы.

Попробуйте улучшить модель, добавляя эффекты взаимодействия, степени, логарифмы и прочее.

Разумеется, полученную модель нужно проверить с помощью test-train и/или кроссвалидации.

Ну и пара заданий для классификации. Сначала двухфакторная классификация:

Задание 1.2. Рассмотрим данные `parkinsons`. Данные и описание в архиве, кратко суть. Есть идея использовать записи речи для диагностики болезни Паркинсона. Для каждого пациента сделано несколько записей, записи обработаны и из каждой получено несколько фич. Нужно научиться по ним предсказывать, болен ли пациент.

Сначала имеет смысл вообще выкинуть столбец “name” и проанализировать записи независимо (т.е. вашим индивидом будет запись). Можно построить разные скаттеры, проверить разные известные методы, нарисовать ROC-кривые, определить значимые и малозначимые фичи.

Потом уже надо рассмотреть в качестве индивидов пациентов, т.е. каким-то образом сгруппировать характеристики записей по пациентам и для каждого пациента получить некоторый набор фич и строить обучение уже на них. Стоит, например, попробовать просто для каждого пациента взять среднее по соответствующей фиче.

Разумеется, полученную модель нужно проверить с помощью test-train и/или кроссвалидации.

И для многофакторной классификации:

Задание 1.3. Рассмотрим данные `seeds`. Данные и описание и архиве, кратко суть. Геометрические измерения зерен пшеницы трех различных сортов. Надо научиться определять сорт по измерениям.

Аналогично, стоит попробовать разные методы, попробовать убрать незначимые признаки, порисовать скаттерплоты, канонические направления, канонические направления мультиномиальной регресии etc.

Разумеется, полученную модель нужно проверить с помощью test-train и/или кросс-валидации.

Еще у меня просили задания на программирование в `R`. Ну вот вам:

Задание 1.4. Написать на `R` функцию `stepCV`, аналогичную `stepAIC`, но использующую кросс-валидацию для проверки значимости признака. Нужно следовать при этом принципу иерархии — нельзя выкидывать признаки более низкого порядка, если есть признаки более высокого. Функция должна работать со всеми методами, с которыми работает `stepAIC`.

Hint: Можно расковырять оригинальную функцию `stepAIC`, а кросс-валидацию взять из `e1071`.

Задание 1.5. Написать (используя `lattice`) функцию `bandplot`, которая вызывается:

```
# bandplot(lower + upper ~ x, data = ..., ...)
```

и рисует закрашенную "доверительную полоску", задаваемую двумя границами. Должно получаться что-то типа `http://stackoverflow.com/questions/14069629/plotting-confide`

Функция должна вести себя как `xyplot`, возвращать `trellis`-объект, понимать параметр `groups` и `|` в формуле. Результат должен корректно взаимодействовать с операциями из `latticeExtra`.

Hint: Достаточно просто дополнить и немного переработать пример `http://www.r-bloggers.com/confidence-bands-with-lattice-and-r/`.