Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики

Кафедра Интеллектуальных Технологий в Гуманитарной Сфере

Интеллектуальные системы и технологии

Лабораторная работа №2  
«Отбор признаков»

Автор:

Пыхтин Михаил

Факультет ФИКТ

Группа №K3342

Преподаватель:

Добренко.Н.В.

Санкт-Петербург  
2018

**Ход работы:**

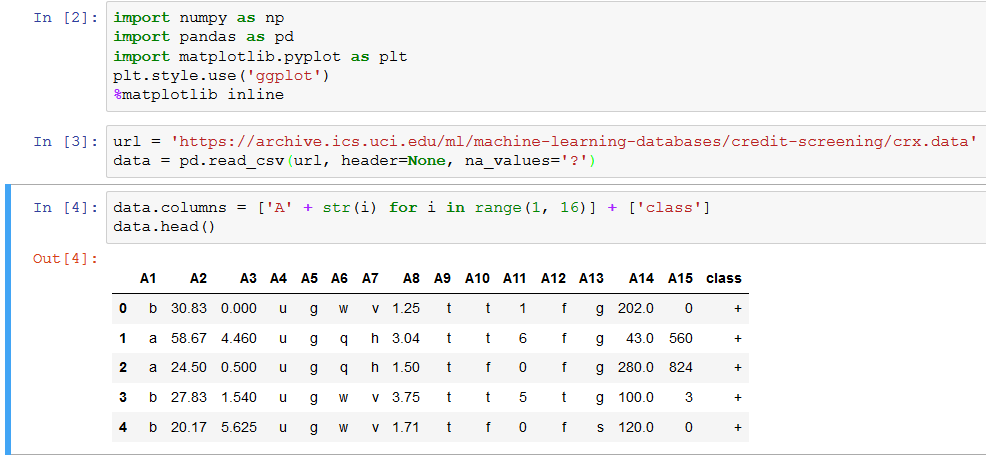
В качестве примера рассмотрим задачу определения кредитной платежеспособности (кредитного скрининга). Пусть имеются данные о клиентах, обратившихся за кредитом. Здесь объектами являются клиенты, а признаками могут являться их пол, уровень дохода, образование, информация о том, имеется или отсутствует у них положительная кредитная история и т.д.

В качестве выделенного признака (ответа) выступает информация о том, вернул ли клиент кредит в банк или нет. По этим данным требуется научиться предсказывать, вернет ли кредит новый клиент, обратившийся в банк.

Таким образом, речь идет о задаче классификации: требуется определить, какому классу: положительному (кредит будет возвращен) или отрицательному (кредит не будет возвращен) – принадлежит клиент.

Этот файл касается заявок на кредитные карты. Все имена атрибутов и значения были изменены на бессмысленные символы для защиты конфиденциальность данных.

Подключаем необходимые библиотеки и загружаем наши данные



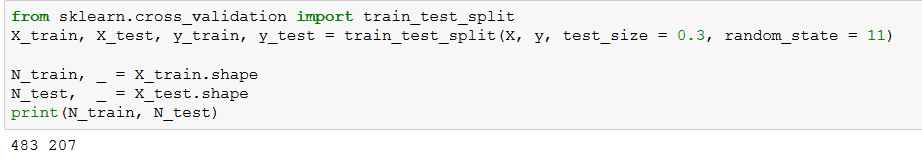
Затем нам нужно проанализировать и исправить некорректные аргументы для нас чтобы корректно начать отбор признаков

Обучаться, или, как говорят, строить модель, мы будем на обучающей выборке, а проверять качество построенной модели – на тестовой. В соревнованиях и конкурсах по анализу данных и машинному обучению участнику доступна только обучающая выборка, а тестовая неизвестна.

В рассматриваемой задаче мы сами разобьем имеющиеся у нас данные на обучающую и тестовую выборки (на самом деле, это больше соответствует реальной ситуации, с которой сталкиваются исследователи).

Разбиение на тестовую и обучающую выборку должно быть случайным. Обычно используют разбиения в пропорции 50%:50%, 60%:40%, 75%:25% и т.д.

Мы воспользуемся функцией train\_test\_split из модуля sklearn.cross\_validation. и разобьем данные на обучающую/тестовую выборки в отношении 70%:30%:



X\_train, y\_train – это обучающая выборка, X\_test, y\_test – тестовая.

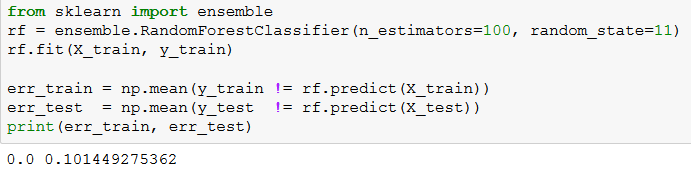
Воспользуемся одним из самых популярных алгоритмов машинного обучения – случайный лес – Random Forest.

Как работает алгоритм?  
Случайный лес - это множество решающих деревьев. В задаче регрессии их ответы усредняются, в задаче классификации принимается решение голосованием по большинству. Все деревья строятся независимо по следующей схеме:

* Выбирается подвыборка обучающей выборки размера samplesize (м.б. с возвращением) – по ней строится дерево (для каждого дерева — своя подвыборка).
* Для построения каждого расщепления в дереве просматриваем max\_features случайных признаков (для каждого нового расщепления — свои случайные признаки).
* Выбираем наилучшие признак и расщепление по нему (по заранее заданному критерию). Дерево строится, как правило, до исчерпания выборки (пока в листьях не останутся представители только одного класса), но в современных реализациях есть параметры, которые ограничивают высоту дерева, число объектов в листьях и число объектов в подвыборке, при котором проводится расщепление.

Понятно, что такая схема построения соответствует главному принципу ансамблирования (построению алгоритма машинного обучения на базе нескольких, в данном случае решающих деревьев): базовые алгоритмы должны быть хорошими и разнообразными (поэтому каждое дерево строится на своей обучающей выборке и при выборе расщеплений есть элемент случайности).

Алгоритм строит ансамбль случайных деревьев, каждое из которых обучается на выборке, полученной из исходной с помощью процедуры изъятия с возвращением.



Ошибка составила 10%

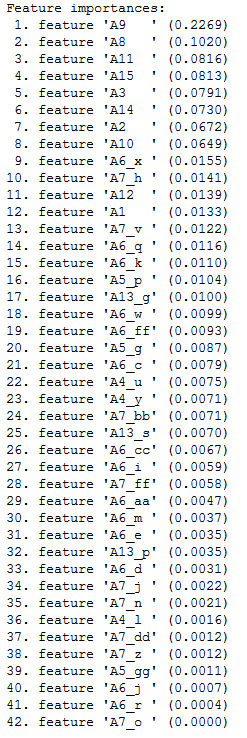
Теперь мы будем отбирать признаки.

Отбор признаков необходим для следующих целей:

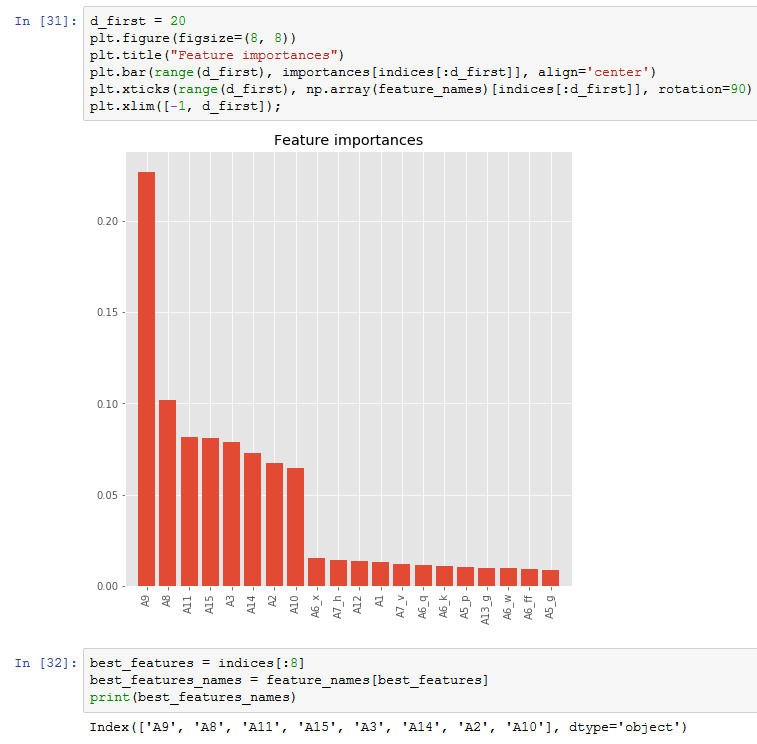
* Для лучшего понимания задачи. Человеку легче разобраться с небольшим количеством признаков, чем с огромным их количеством.
* Для ускорения алгоритмов.
* Для улучшения качества предсказания. Устранение шумовых признаков может уменьшить ошибку алгоритма на тестовой выборке, т.е. улучшить качество предсказания.

**Отбор признаков (Feature Selection) с помощью алгоритма случайного леса**

Произведем отбор, и выведем признаки в порядке их значимости

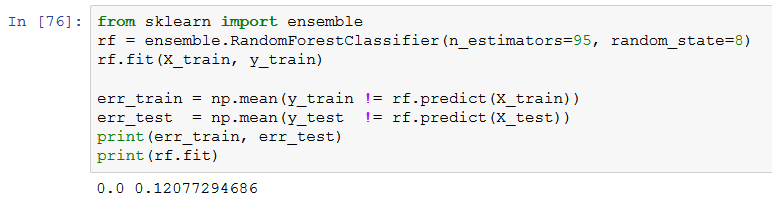


И покажем их на графике, чтобы получить более наглядную картину

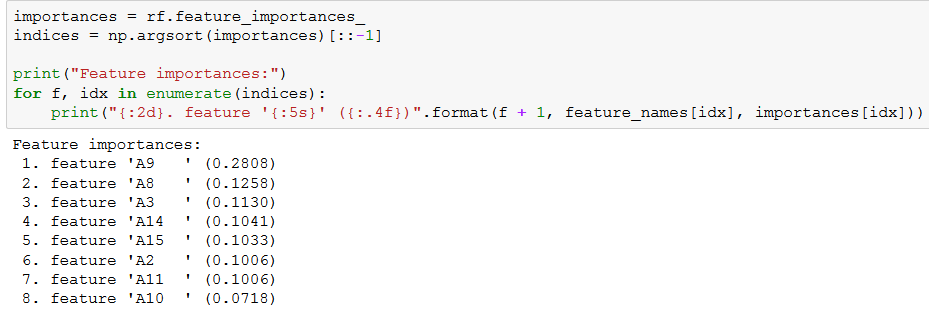


Мы видим, что основную роль играют признаки A9, A8, A11, A15, A3, A14, A2, A10

Теперь уберем все ненужные признаки, и оставим только важные, дабы проверить качество работы.



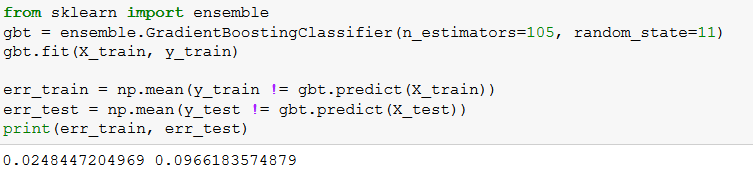
Как видим, ошибка незначительно увеличилась, что можно воспринимать как успех в отборе признаков



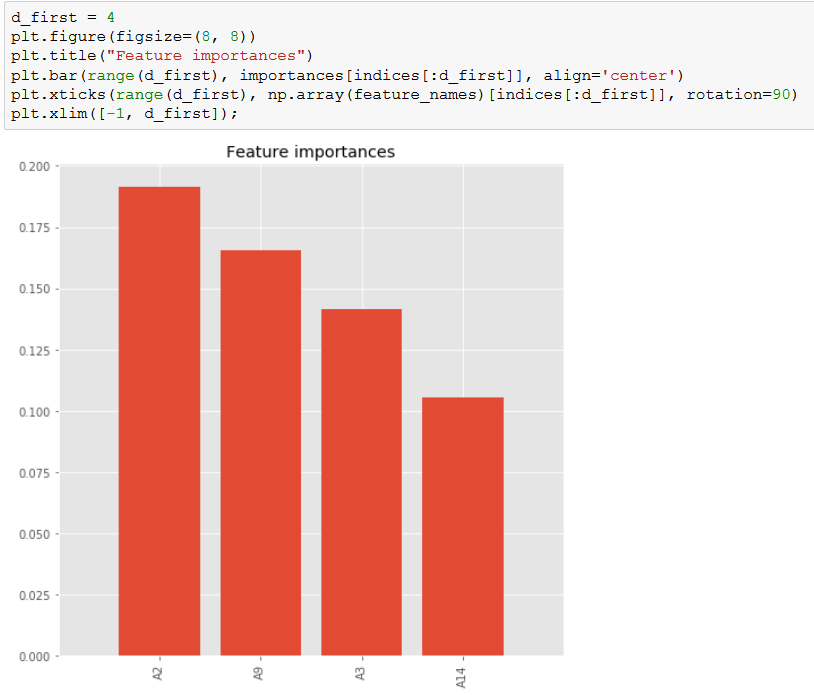
## GBT – градиентный бустинг деревьев решений

GBT – еще один метод, строящий ансамбль деревьев решений. На каждой итерации строится новый классификатор, аппроксимирующий значение градиента функции потерь.

Градиентный бустинг позволяет без серьезных модификаций алгоритма обучения использовать любой дифференцируемый штраф

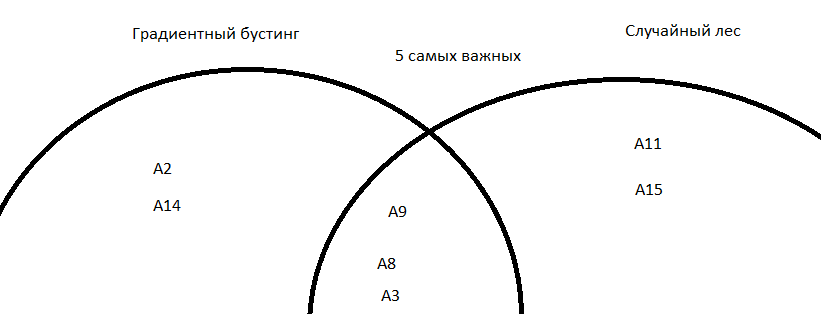


Как видим, ошибка уменьшилась, и соответственно важность признаков перераспределилась

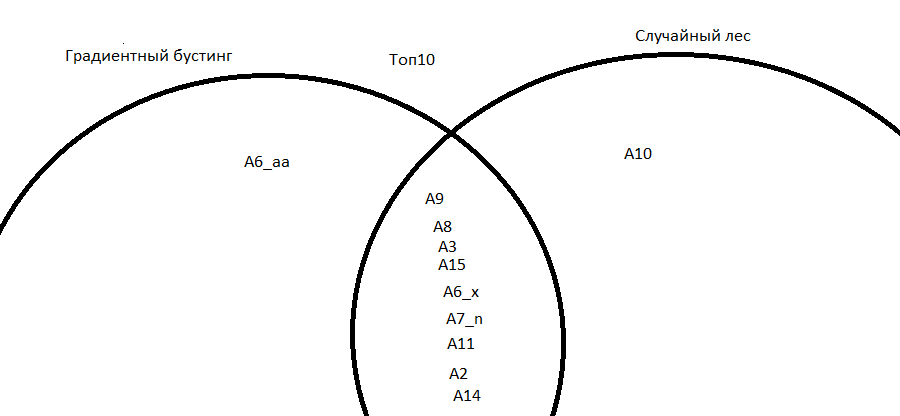


Используем диаграммы Венна для наглядности различия

Различие в 5 самых важных признаках(топ5):



Теперь рассмотрим 10 самых важных признаков(топ10):



Таким образом, метод Градиентного бустинга дополняет и улучшает качество работы случайного леса, уменьшая ошибку и перераспределяя значимость признаков, практически не внося поправок в общую картину.

**Вывод:**

Таким образом, мы научились использовать отбор признаков для машинного обучения, чтобы избежать переобучения нейросети и избежать лишних бесполезных данных.