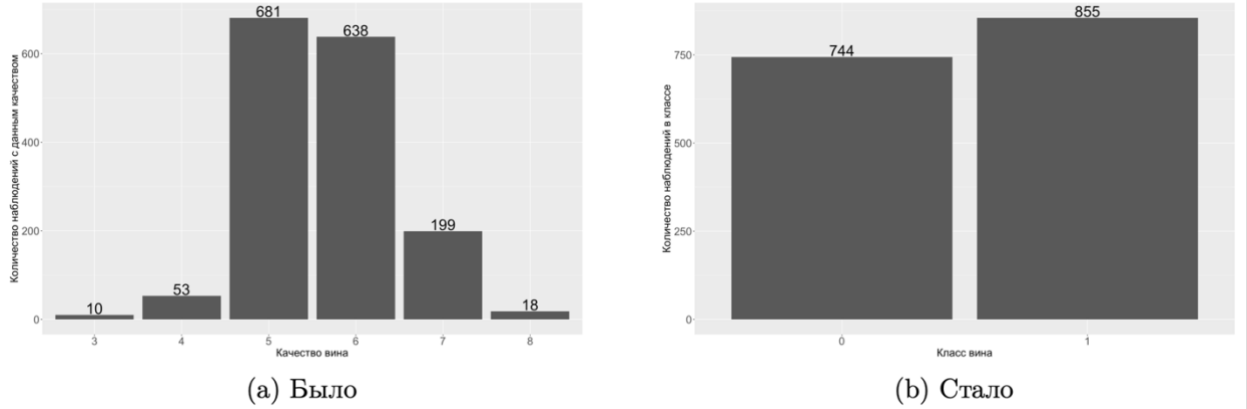
Домашнее задание №5 (Отчёт)

Пункт 1

Мы имеем 1599 наблюдений вина различного качества (quality) с определённым набором характеристик: fixed.acidity, volatile.acidity, citric.acid, residual.sugar, chlorides, free.sulfur.dioxide, total.sulfur.dioxide, density, pH, sulphates, alcohol. Ниже представлена описательная статистика данных переменных:

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
fixed.acidity	1,599	8.320	1.741	4.600	7.100	9.200	15.900
volatile.acidity	1,599	0.528	0.179	0.120	0.390	0.640	1.580
citric.acid	1,599	0.271	0.195	0	0.1	0.4	1
residual.sugar	1,599	2.539	1.410	0.900	1.900	2.600	15.500
chlorides	1,599	0.087	0.047	0.012	0.070	0.090	0.611
free.sulfur.dioxide	1,599	15.875	10.460	1	7	21	72
total.sulfur.dioxide	1,599	46.468	32.895	6	22	62	289
density	1,599	0.997	0.002	0.990	0.996	0.998	1.004
pH	1,599	3.311	0.154	2.740	3.210	3.400	4.010
sulphates	1,599	0.658	0.170	0.330	0.550	0.730	2.000
alcohol	1,599	10.423	1.066	8.400	9.500	11.100	14.900
quality	1,599	5.636	0.808	3	5	6	8

Для стандартной задачи классификации необходимо разделить вино по качеству на два класса. Ниже на рисунке (a) можно увидеть распределение наблюдений по качеству вина. Для пунктов 2-4 разделим наблюдения примерно поровну бинарной переменной: 0 – вино с качеством меньше либо равно 5, 1 – с качеством выше 5. Результат разделения виден на правом рисунке (b): мы имеем 744 наблюдения с плохим качеством и 855 наблюдений с хорошим качеством.



(a) Было (b) Стало

Пункт 2

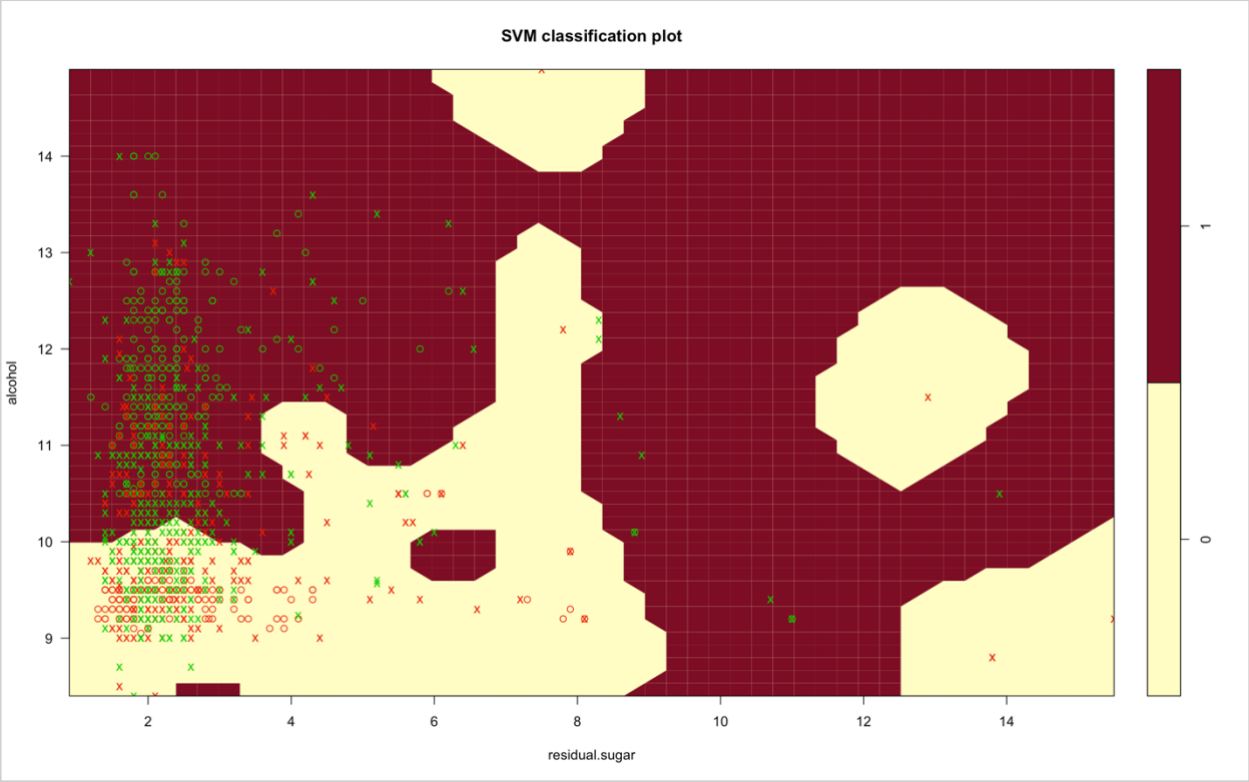
Разделяем выборку на тренировочную и тестовую (3:1). Обучим метод опорных векторов с RBF ядром и параметрами C = 5, gamma = 2. Ошибка на тестовой выборке составила 0.28. Всего алгоритм использовал 1049 опорных векторов: 493 класса 0 и 556 класса 1.

Пункт 3

Обучим аналогичный алгоритм с тем же набором параметров только с использованием двух характеристик (alcohol и residual sugar) и проведением кросс-валидации (k = 10). В результате обучения средняя ошибка предсказания на обучающей выборке составила 0.29 (всего 755 опорных векторов).

Также рассмотрим графическую иллюстрацию (представлена ниже): диаграмму рассеяния с выделенными областями, которые показывают, как алгоритм будет классифицировать (и уже классифицирует) вино по двум характеристикам. Зелёным точкам соответствует вино с хорошим качеством (1), красным – с плохим (0). Также крестиком прорисованы те наблюдения, которые были избраны опорными.

Мы можем наблюдать несколько возможных проблем, на которые намекает график. Во-первых, использование данных характеристик не лучший выбор для определения качества вина. Несмотря на то, что большинство точек классифицировано верно, ошибка достаточно велика, алгоритму пришлось набрать большое количество опорных векторов, часть из которых скорее правильнее считать выбросами (например точка с координатами (13, 11.5)), чем определяющими при создании областей для классификации. Возможно, тюнинг гиперпараметров сможет решить эту проблему (хотя бы частично). Во-вторых, с чем может быть связана проблема – некорректное деление на классы, которые было сделано на первом шаге. Возможно, хорошим вином нужно считать вино с более высоким качеством, чем казалось раньше при экзогенном делении. Эту проблему проработаем в пункте 5.



Пункт 4

Подберём лучшие гиперпараметры на обучающей выборке для SVM с RBF ядром с помощью кросс-валидации (k = 10). Диапазон для рассмотрения: C – от 1 до 30 с шагом 1, gamma – от 0,1 до 3 с шагом 0,1. Лучшими параметрами оказались C = 2 и gamma = 0.4. Используем их и получаем: ошибка на обучающей выборке – 0.09, ошибка на тестовой – 0.22. Количество опорных векторов равно 862: 412 принадлежат классу 0 и 450 – классу 1.

Пункт 5

Попытаемся подобрать лучшее деление. Опираясь на распределение качеств вина, отсекать нужно так, чтобы в каждом классе было как минимум два качества: очень мало наблюдений с качеством 3 и 8, обучение при делении по ним будет совсем неверным шагом. Поэтому будет рассматривать деления {34} {5678}, {345} {678}, {3456} {78} для классов 0 и 1 соответственно. Также стоит предположить, что для каждого деления будет свой лучший набор гиперпараметров, поэтому каждый раз будем рассматривать такие диапазоны: C – от 1 до 100 с шагом 3, gamma – от 0,1 до 10 с шагом 0,3. Параметр кросс-валидации немного снизим: k = 7 (чуть ускоримся). Лучшие гиперпараметры используем при обучении на всей тренировочной выборке, далее проверим ошибки прогноза на тренировочной и тестовой выборках.

Деление	Лучшие параметры	Ошибка на обучающей	Ошибка на тестовой
{34} {5678}	C = 4, gamma = 1.9	0.000	0.038
{345} {678}	C = 10, gamma = 0.1	0.139	0.243
{3456} {78}	C = 4, gamma = 1.0	0.001	0.098

Результаты цикла представлены в таблице выше. Как мы можем видеть, наше изначальное деление сильно проигрывает двум другим как на тренировочной выборке, так и на тестовой, что более важно. {34} {5678} лучше {3456} {78} на тестовой выборке, ошибка на 6 процентных пунктов меньше (на тренировочной оба хороши). Убедимся, что деление {34} {5678} на самом деле лучше {3456} {78}, позволив подобрать параметры с большей точностью. Изменим диапазон: C – от 1 до 8 с шагом 0.5, gamma – от 0,1 до 3 с шагом 0,1. Результаты таковы:

Деление	Лучшие параметры	Ошибка на обучающей	Ошибка на тестовой
{34} {5678}	C = 2, gamma = 1.6	0.001	0.038
{3456} {78}	C = 2.5, gamma = 1.0	0.001	0.098

Можно было бы сделать вывод о том, что предсказание класса наилучшее при делении {34} {5678}. Однако рассмотрение матриц ошибок даёт понимание того, что этот алгоритм предпочитает выбирать один класс, и так как наблюдений класса 0 при данном отсечении мало, ошибка выходит незначительной (хотя и была попытка избежать этой проблемы при не рассмотрении отсечений с одним качеством против остальных). Таким образом, остановлю свой выбор на делении {3456} {78}.