Семинар 9: алгоритмы классификации

На прошлом семинаре мы говорили про то, какими бывают метрики классификации. Обычно их используют, чтобы сравнивать между собой различные алгоритмы для классификации. Ни одного такого алгоритма мы пока ещё не знаем. Пришло время исправить это досадное упущение.

Упражнение 1 (KNN и кросс-валидация)

На плоскости расположены колонии рыжих и чёрных муравьёв. Рыжих колоний три и они имеют координаты $(-1,-1)$, $(1,1)$ и $(3,3)$. Чёрных колоний тоже три и они имеют координаты $(2,2)$, $(4,4)$ и $(6,6)$.

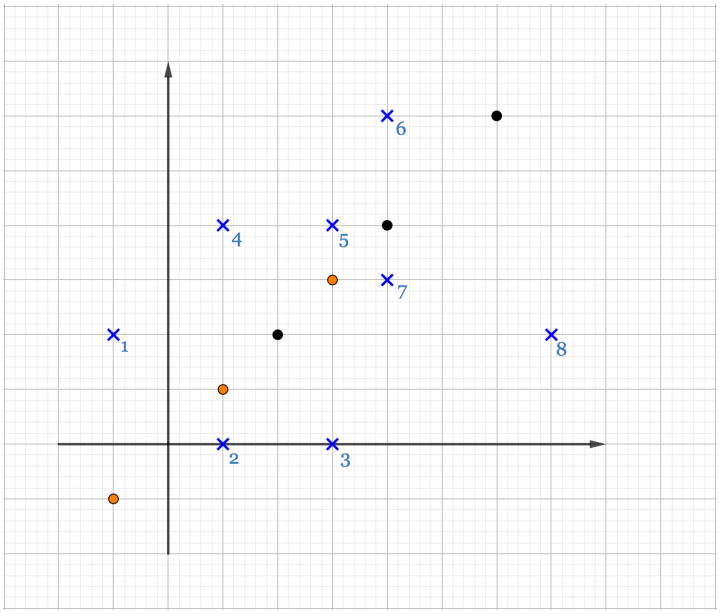
а) Поделите плоскость на «зоны влияния» рыжих и чёрных муравьёв, используя метод одного ближайшего соседа.

б) Поделите плоскость на «зоны влияния» рыжих и чёрных муравьёв, используя метод трёх ближайших соседей.

в) С помощью кросс-валидации с выкидыванием отдельных наблюдений выберите оптимальное число соседей k перебрав $k \in \{1,3,5\}$. Целевой функцией является количество верных предсказаний (accuracy).

Решение:

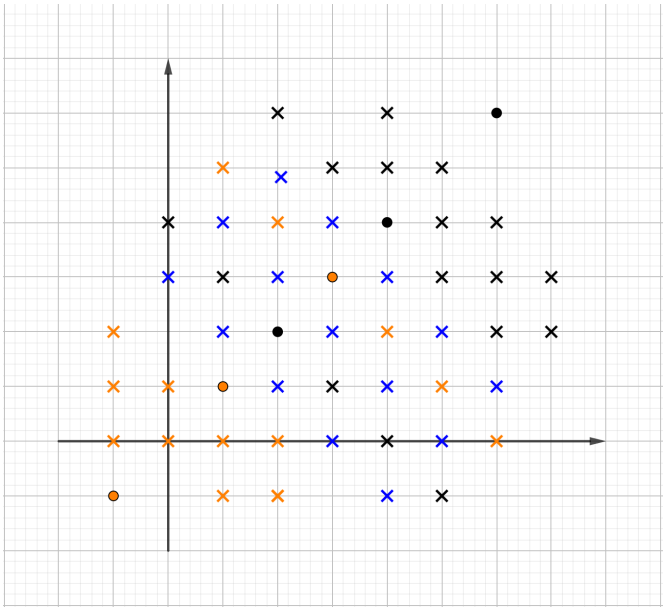
а) Будем ради удобства измерять расстояние между муравейниками в метрах. Давайте отметим на плоскости несколько рандомных точек и посмотрим к чьей зоне влияния они относятся.



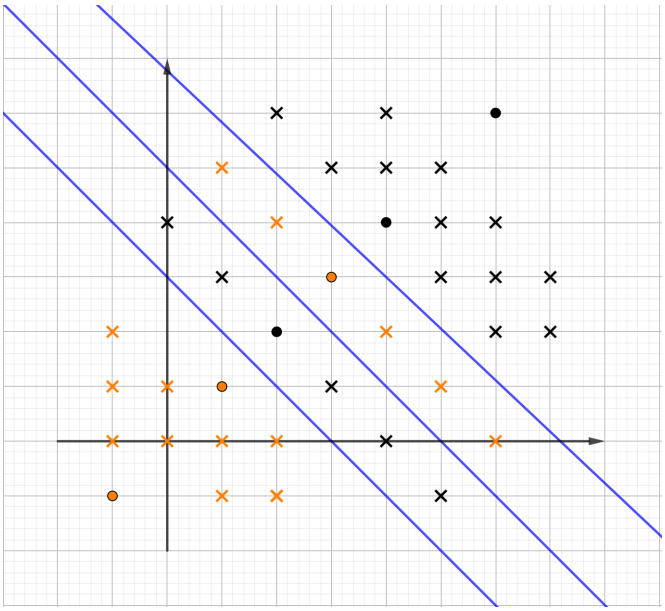
Точка номер один явно будет в зоне влияния рыжих муравьёв. До ближайшего рыжего муравейника нужно пройти $\sqrt{5}$ метров, до ближайшего чёрного 3 метра. Точка два тоже рыжая.

По аналогии точки восемь и шесть оказываются чёрными. С оставшимися точками возникают проблемы. Например, от точки номер пять одинаковое расстояние как до чёрного, так и до рыжего муравейников. Она является спорной. Судя по всему, именно через неё пройдёт граница. Давайте попробуем нащупать побольше подобных пограничных точек.

Если точка принадлежит рыжим муравьям, будем помечать её рыжим крестом. Если чёрным, то чёрным. Если это спорная точка, то синим.

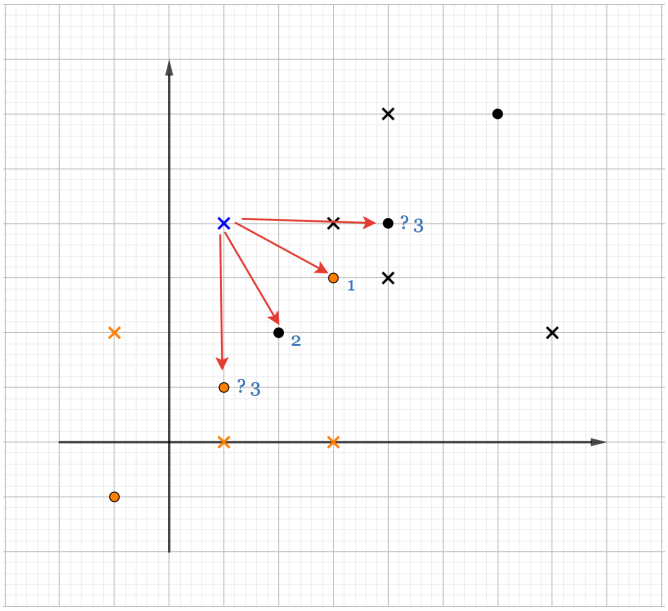


Кажется, что мы нащупали границы, вдоль которых находятся спорные территории. Осталось только прочертить их.

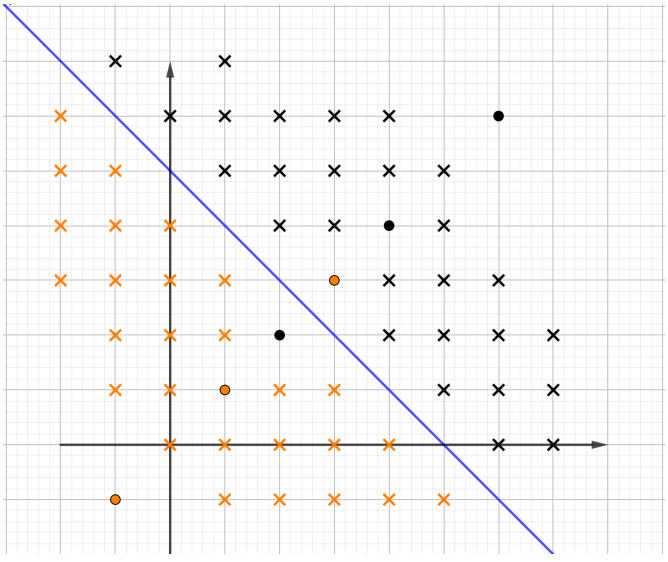


б) Теперь попробуем поделить плоскость на зоны влияния, используя метод трёх ближайших соседей. Посмотрим на самую первую картинку, где мы нанесли на плоскость рандомные точки, и попробуем порассуждать в чьей зоне влияния оказывается какая точка.

Для первой точки две из трёх ближайших — рыжие. Она находится в рыжей зоне влияния. По аналогии происходит со второй и третьей точками. Пятая, шестая, седьмая и восьмая точки оказываются в зоне влияния чёрных муравьёв и окрашиваются в чёрные цвета. Проблемы возникают только с четвёртой точкой. Ближайшие к ней две точки — рыжая и чёрная. Решение надо принимать по третьему ближайшему соседу. Третью ближайшую точку найти не удаётся, так как рыжая и чёрная точка находятся от неё н одинаковых расстояниях. Выходит, что мы оказались на границе.



Попробуем нащупать ещё пограничных точек и провести пограничную линию.



И это граница? У нас же есть ошибки! Да, есть. Давайте вспомним упражнение про собачек, кошек, пиццу и бургеры с прошлого семинара. Когда мы решали его, мы поняли, что слишком детализированная граница между классами приводит к переобучению.

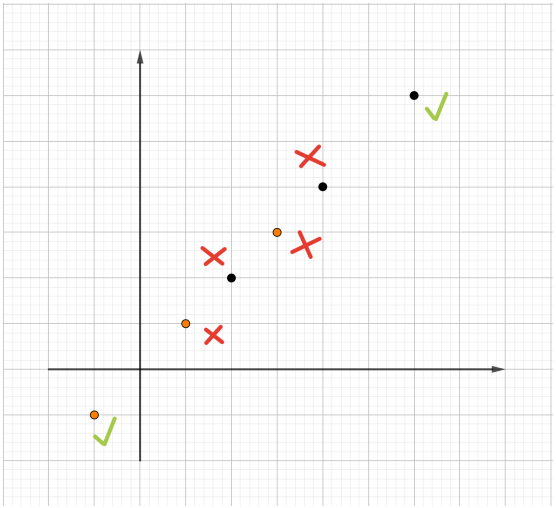
Порассуждаем в терминах джунглей. Есть поле, на нём селятся муравьи. Логично ли с их стороны селиться полосками? Конечно же нет. Намного логичнее было бы, что по историческим причинам на одной стороне поля живут рыжие муравьи, на второй чёрные.

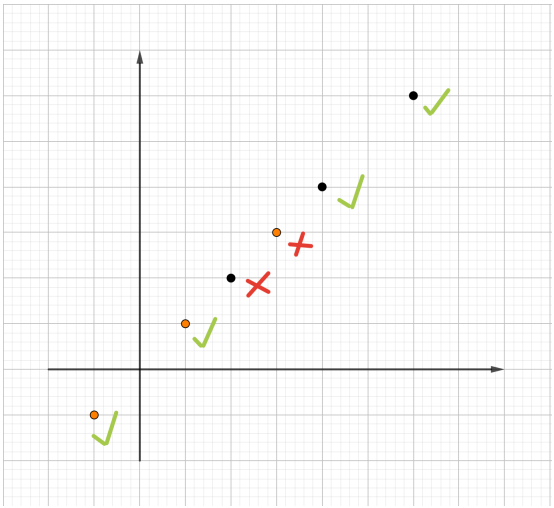
У нас в выборке оказалось несколько примеров муравейников. И по ним мы попытались нащупать границу для зон влияния. На границе вполне может происходить такое, что муравьи проникают на территорию друг-друга.

Проводя излишние полосы, мы переходим от выуживания реальных закономерностей, существующих в джунглях, к излишнему фрагментированию обучающей выборки, то есть переобучаемся под её особенности. Практически все алгоритмы машинного обучения страдают этим грехом. Нам надо будет как следует бить их за переобучение и исправлять.

в) Давайте убедимся в том, что алгоритм трёх ближайших соседей, проводящий одну разграничительную линию между муравьями, работает лучше, чем алгоритм одного ближайшего соседа. Для этого воспользуемся стратегией кросс-валидации.

Кросс-валидация состоит в следующем: давайте будем закрывать по очереди разные части выборки ладошкой. На оставшейся выборке будем обучать модель, а на скрытой проверять её качество. Будем делать так много раз и посмотрим на итоговое качество.





Закрываем ладошкой самую нижнюю точку. По оставшимся четырём расчерчиваем границы. Мы по методу одного ближайшего соседа относим эту точку к рыжим муравьям. Это оказывается правильным решением. Угадали.

Закроем ладошкой вторую снизу точку. Расчертим границы. Она окажется ближе всего к чёрным муравьям. Но на самом деле она рыжая. Ошибка... Также проделаем с остальными точками. В итоге получится, что мы совершаем целых 4 ошибки. По аналогии сделаем с методом трёх ближайших соседей и получим всего лишь 2 ошибки.

Чувствуете? Мы ошибаемся из-за излишней детализации, которую нам навязывает метод одного ближайшего соседа. Кросс-валидация позволяет это отследить. А что, если взять 5 ближайших соседей? Тогда мы ошибёмся абсолютно в каждой точке. Попробуйте проделать это.

На самом деле k это **гиперпараметр** метода ближайших соседей. Мы можем подобрать его оптимальным образом с помощью кросс-валидации. В данном примере оптимально будет выбрать $k = 3$.

В этом упражнении мы использовали для оптимизации алгоритма метрику accuracy. Понятное дело, что можно пытаться использовать и любую другую метрику с прошлого

семинара.

Упражнение 2 (древо для классификации)

У Маши есть инстаграмчик. На неё подписана целая куча парней. Недавно Дима поставил ей 10 лайков. Тогда Маша схватила ромашку и начала гадать. Она нагадала, что Дима плюнет в неё. То же самое она сделала с другими парнями.

Результат гадания — переменная y_i, количество лайков у фотки — переменная x_i.

y_i	x_i
плюнет	10
поцелует	11
поцелует	12
к сердцу прижмёт	13
к сердцу прижмёт	14

Сегодня в Машином инстаграмме Джонни Деп поставил 15 лайков. Маше очень хочется понадать что же с ней сделает Джонни Деп, но у неё кончились ромашки. Поэтому она решила обучить классификационное дерево, которое поможет ей спрогнозировать y_i по x_i.

Дерево строится до идеальной классификации. Критерий деления узла на два — минимизация числа допущенных ошибок[1]. Правило прогнозирования в каждой вершине: в качестве прогноза выдаем тот класс, представителей которого в вершине больше.

Решение:

Мы должны обучить дерево, которое будет по переменной x, число лайков от парня, прогнозировать переменную y, состояние отношений Маши. Обычно деревья учат по-жадному. Будем смотреть, какое разбиение по переменной x сильнее всего уменьшает ошибку, и выбирать его.

Ошибку мы договорились считать как долю неверных ответов. Обычно на практике при разбиении вершины на две используют не такой критерий, но мы для простоты используем его.

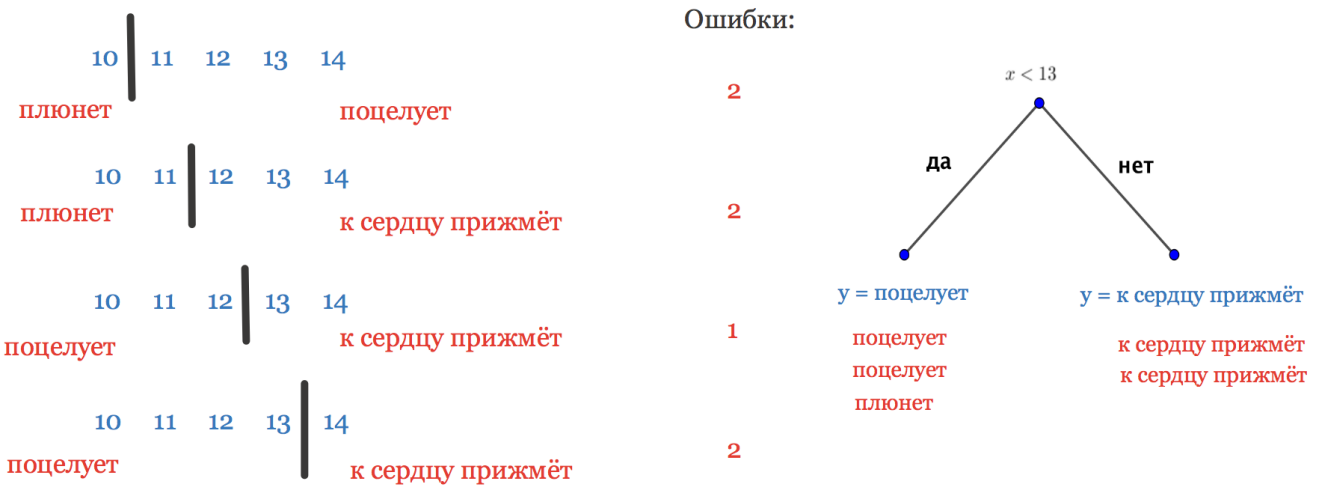
Будем перебирать все возможные пороги и смотреть что получится.

При делении вершины на две между 10 и 11 лайками, слева у нас окажется плюнет. Именно его мы и будем там прогнозировать. Справа окажется два поцелует и два к сердцу прижмёт. Надо спрогнозировать в этой вершине класс, представителей которого тут большинство, чтобы сделать поменьше ошибок. Так как у нас оба класса представлены в одинаковом объёме, неважно что мы спрогнозируем. В любом случае получим две ошибки.

При дроблении вершины на две между 11 и 12 лайками, слева оказывается плюнет и поцелует. Одна ошибка. Справа оказывается два к сердцу прижмёт и одно поцелует. Спрогнозируем

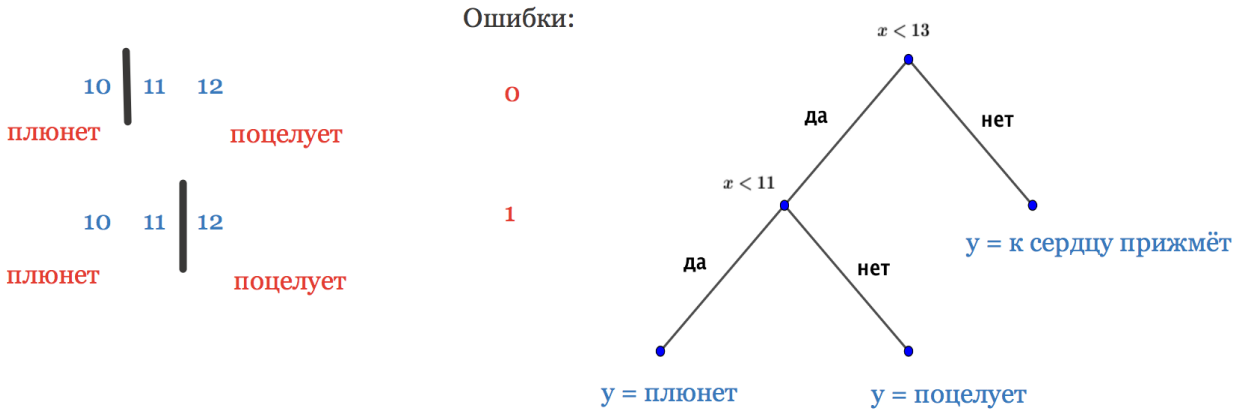
[1] На самом деле на практике так не делают. Обычно для разбиения узла при строительстве классификационных деревьев используют энтропию. О том, что это такое, можно погуглить.

к сердцу прижмёт, так как их большинство, и получи одну ошибку. В сумме у нас две ошибки.



Рассуждая аналогичном образом приходим к выводу, что самое классное разбиение между 12 и 13. При нём мы совершаем только одну ошибку. В дереве, мы будем задавать вопрос: «А количество лайков меньше 13?» Если да, будем идти налево и прогнозировать, что нас поцелуют. Если нет, будем идти направо и прогнозировать, что нас прижмут к сердцу.

Справа в листе дерева у нас оказались объекты одного класса. Слева в листе дерева соодержутся объекты разных классов. Можно сделать ещё одно разбиение.

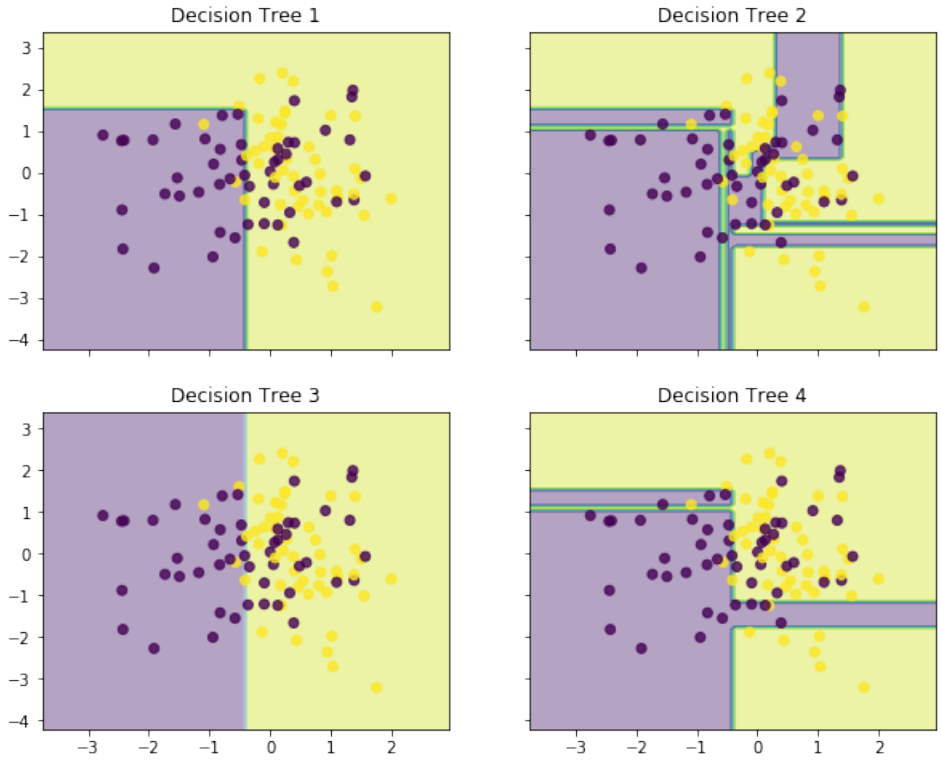


В итоге в нашем дереве окажется три листа, в каждом из которых мы будем делать прогноз. Обратите внимание, что дерево запомнило выборку. Деревья постоянно так делают. В этом их сушественный минус. Чтобы победить его, деревья стригут. Либо используют как части более сложных моделей. Например, как часть случайного леса.

Джонни поставил Маше 15 лайков. Что же ожидает их отношения? Начинаем идти по решающему дереву, чтобы сделать прогноз. Число лайков меньше 13? Нет. Идём направо. Кажется, Машу прижмут к сердцу. Это наш прогноз.

Упражнение 3 (ещё деревья)

Ниже изображены разделяющие поверхности для задачи бинарной классификации, соответствующие решающим деревьям разной глубины. Какое из изображений соответствует наиболее глубокому дереву? Какой примерной глубине дерева соответствует каждая из картинок?



Решение:

Чем глубже дерево, тем сильнее оно фрагментирует нашу выборку, и тем сильнее оно выделяет в ней самые микроскопические кусочки. Сильнее всего выборка фрагментирована на верхней правой картинке, значит это разбиение плоскости на части соответствует самому глубокому дереву.

На третьей картинке плоскость дробиться на части один раз. Значит в дереве есть один сплит. Его глубина равна единице. На первой картинке появляется ещё одно дополнительное разбиение по оси y, глубина дерева увеличивается до двух.

На картинке номер 4 мы делаем два дополнительных разбиения правой части и два левой. Глубина дерева уже не менее трёх. На второй картинке всё становится ещё глубже.

Ещё задачи

Тут лежит ещё несколько задач для самостоятельного решения. Возможно, похожие будут в самостоятельной работе...

Упражнение 4

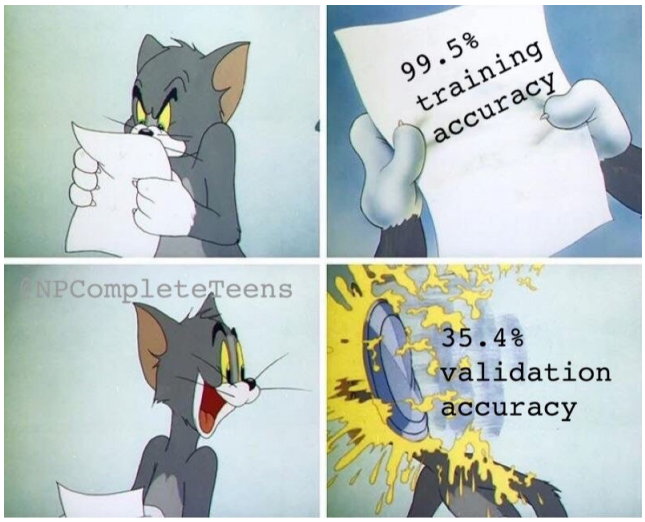
На плоскости расположены колонии рыжих и чёрных муравьёв. Рыжих колоний три и они имеют координаты $(-1, 1)$, $(1, -1)$ и $(1, 1)$. Чёрных колоний одна и она имеет координаты $(0, 0)$.

а) Поделите плоскость на «зоны влияния» рыжих и чёрных используя метод одного и трёх ближайших соседей.

б) С помощью кросс-валидации с выкидыванием отдельных наблюдений выберите оптимальное число соседей k перебрав $k \in \{1, 3\}$. Целевой функцией является количество несовпадающих прогнозов.

Упражнение 5

Объясните мемас:



Упражнение 6

Пятачок собрал данные о визитах Винни-Пуха в гости к Кролику. Здесь x_i — количество съеденного мёда в горшках, а y_i — бинарная переменная, отражающая застревание Винни-Пуха при входе

y_i	x_i
0	1
1	4
1	2
0	3
1	3
0	1

а) Пятачок собирается оценить дерево по всей выборке. Помогите очень маленькому существу сделать это.

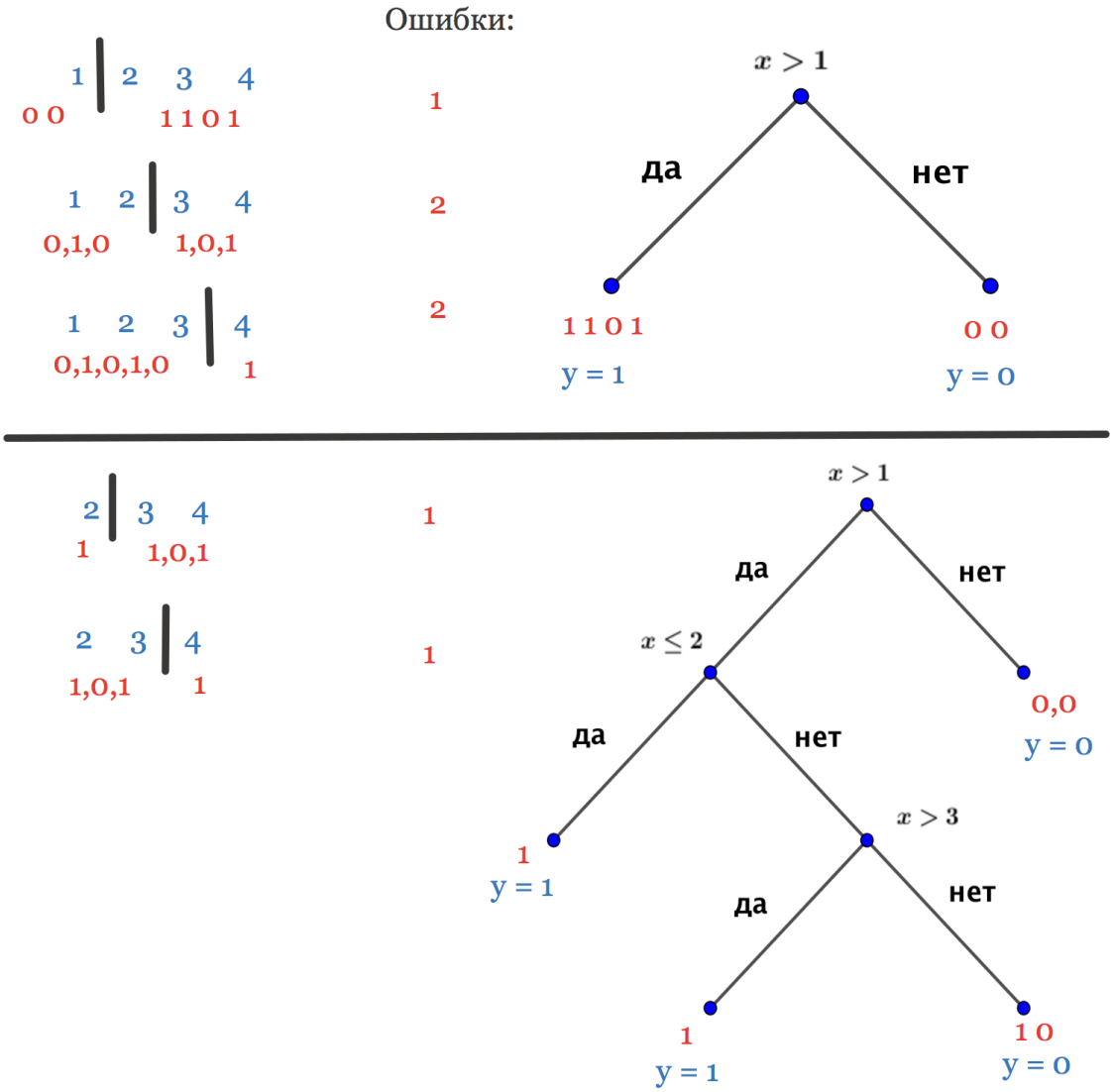
б) Пятачок узнал у Иа-Иа, что оказывается выборку надо делить на тренировочную и тестовую Поэтому он отложил последние два наблюдения для теста. Оцените дерево по первым четырём наблюдениям и проверьте его работоспособность по последним двум.

в) Пятачок поговорил с Совой и узнал, что деревья часто переобучаются. Она рассказала ему, что над деревьями надо строить ансамбли. Например, случайный лес. Пятачок решил построить лес из трёх деревьев. Первое дерево он строит на наблюдениях с первого

по третье, второе на наблюдениях со второго по четвёртое. Третье дерево на наблюдениях $1, 2, 4$. Помогите пяточку построить лес и оценить качество его работы на тестовой выборке.

Решение:

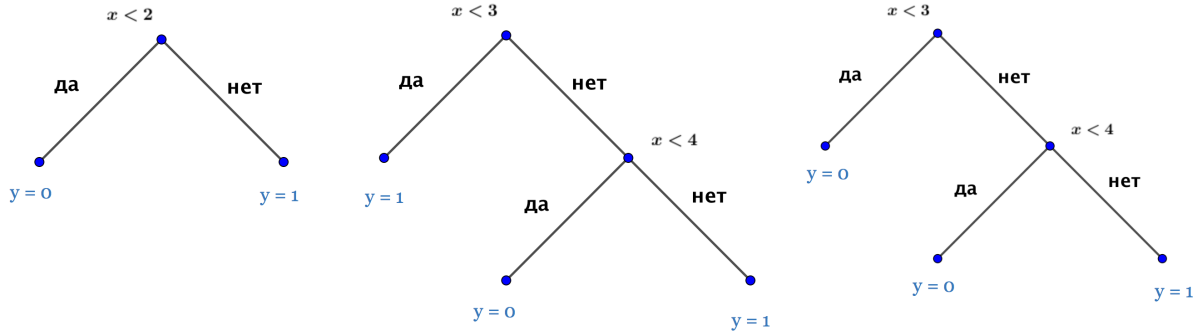
а) Бедный малютка Пяточок! Он даже не понимал, на какие муки он себя обрекает, когда собирался строить свою модель для прогнозирования того, что произойдёт с Винни! Как же хорошо, что мы оказались рядом и подставили маленькому существу своё большое дружеское плечо. Для начала построим дерево сразу на всей выборке.



Обратите внимание, что это дерево ошибается из-за того, что при $x = 3$ у нас есть как факт застревания медведя в норе, так и факт его прохождения сквозь нору. Когда мы оказываемся в вершине, где одинаковое число нулей и единиц, нам придется принимать решение случайно. Компьютер поступает именно так. На больших выборках это случаентся довольно редко.

б) Когда мы строим дерево на первых четырёх наблюдениях, первое разбиение можно сделать либо по единице, либо по четвёрке. В обеих ситуациях совершается одна ошибка. Для удобства выберем первый случай. Дальше снова неважно где делать разбиение. Сделаем его в двойке. В итоге получим дерево из пункта а). На тестовой выборке дерево делает одну ошибку при $x = 3$.

в) Лес, который должен получиться в ходе обучения, изображён на кратинке:

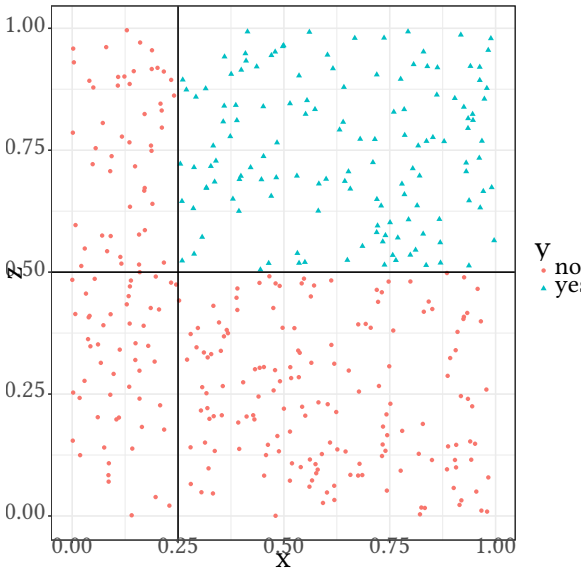


Для $x = 3$ первое дерево прогнозирует 1, второе 0, третье 0. Большая часть леса говорит, что прогнозом будет 0. Верим в торжество демократии и берём его в качестве итогового прогноза. Допускаем ошибку[2]. Для $x = 1$ первое и третье деревья прогнозируют 0, второе 1, берём ноль и не ошибаемся. Прогноз по лесу построен.

На практике случайный лес показывает себя как очень мощный алгоритм. Не стесняйтесь использовать его.

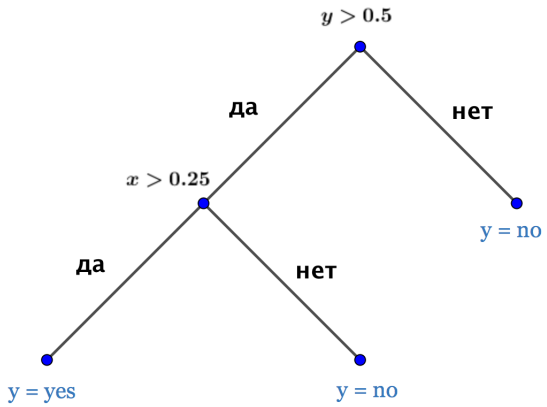
Упражнение 7

По данной диаграмме рассеяния постройте классификационное дерево для зависимой переменной y:



[2] Демократия не идеальна, но это лучшее, что придумало человечество

Решение:



Если мы дробим сначала по оси y, то мы сразу же довольно сильно уменьшаем неопределённость и ошибаемся только на верхнем левом прямоугольнике, прогнозируя, что он синего цвета.

Если мы дробим сначала по переменной x, то мы будем ошибаться на нижнем правом прямоугольнике. Там ошибка намного страшнее. Значит, сначала произойдёт разбиение по y, затем по x. Именно в этом состоит жадная процедура обучения дерева: уменьшить ошибку при каждом разбиении как можно сильнее.

Упражнение 8

Рассмотрим обучающую выборку для прогнозирования y с помощью x и z:

y_i	x_i	z_i
y_1	1	2
y_2	1	2
y_3	2	2
y_4	2	1
y_5	2	1
y_6	2	1
y_7	2	1

Будем называть деревья разными, если они выдают разные прогнозы на обучающей выборке. Сколько существует разных классификационных деревьев для данного набора данных?

Решение:

Либо мы сначала дробим по x, потом по z. Либо наоборот.