Содержание

1 Введение

1.1 Дерево

Деревом называют конечный, связанный граф со множеством вершин V, не содердащих циклов и имеющий выделенную вершину $v_0 \in V$, в которую не входит ни одно ребро. Эта вершина — корень дерева. Вершина, не имеющая выходящих рёбер — терминальная или лист. Остальные вершины — внутренние.

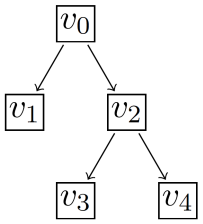

Рис. 1. дерево решений

1.2 Бинарное дерево

Дерево называется бинарным, если из любой его внутренней вершины выходит ровно два ребра. Выходящие ребра связывают каждую вершину v с левой дочерней вершиной L_v и правой дочерней вершиной R_v.

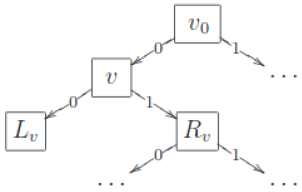

Рис. 2. построение бинарного дерева решений

1.3 Постановка задачи

Деревья решений применяются для классификации и регрессии. Пусть $\mathbf{X}$ — множество объектов, $\mathbf{y}$ — множество ответов.

Если $\mathbf{y}$ — бинарный или номинальный признак, то решаем задачу классификации.

Если $\mathbf{y}$ — количественный признак, то решаем задачу регрессии.

2 Дано

Набор данных

$$\mathbf{X} \in \mathbb{R}^{n \times p}.$$

Зависимые переменные

$$\mathbf{y} \in \mathbb{R}^{n}.$$

$\mathbf{x}_i \in \mathbb{R}^p$ — вектор–строки $\mathbf{X}$.
$X_j \in \mathbb{R}^n$ — вектор–столбцы $\mathbf{X}$.

$y \in \{1, \cdots, K\}$ — задача классификации.
$y \in \mathbb{R}$ — задача регрессии.

3 Вероятностная постановка

3.1 Генеральная постановка

Предполагаем, что η и $\boldsymbol{\xi}$ функционально зависимы:

$$\eta = \varphi(\boldsymbol{\xi}) + \varepsilon, \tag{1}$$

φ — неизвестная функция.
$\eta \in \mathbb{R}$ — случайная величина, зависимая переменная.
$\boldsymbol{\xi} \in \mathbb{R}^p$ — случайный вектор, признаки.
$\varepsilon \in \mathbb{R}$ — случайная величина, ошибка.

3.2 Выборочная постановка

$$y_i = \varphi(\mathbf{x}_i) + \varepsilon_i, \tag{2}$$

φ — неизвестная функция.
y_i — реализация случайной величины η, зависимая переменная.
$\mathbf{x}_i$ — реализация случайного вектора $\boldsymbol{\xi}$, признаки.
$\varepsilon_i \in \mathbb{R}$ — реализация случайной величины ε, ошибка.

4 Регрессионные деревья

Идея построения дерева решений заключается в разделении пространства признаков $X_1, \cdots, X_p$ на J непересекающиеся области $R_1, \cdots, R_J$.

Модель

$$\varphi(\mathbf{x}_i, \mathbf{\Theta}) = \sum_{j=1}^{J} c_j \mathbb{I}_{(\mathbf{x}_i \in R_j)}. \tag{3}$$

$\mathbf{x}_i \in \mathbb{R}^p$ — индивиды.
$\mathbf{\Theta} \in \mathbb{R}^p$ — вектор коэффициентов.
$\mathbf{\Theta} = \{c_1, \cdots, c_J\}$.
$\mathbb{I}_{(\mathbf{x}_i \in R_j)}$ — индикаторная функция принадлежности индивида $\mathbf{x}_i$ области R_j.

Функция потерь

$$\text{RSS} = \sum_{j=1}^{J} \sum_{\mathbf{x}_i \in R_j} (y_i - \varphi(\mathbf{x}_i, \mathbf{\Theta}))^2. \tag{4}$$

Оптимизация

$$\text{RSS} = \sum_{j=1}^{J} \sum_{\mathbf{x}_i \in R_j} (y_i - \varphi(\mathbf{x}_i, \mathbf{\Theta}))^2 \to \min_{R_1, \cdots, R_J}. \tag{5}$$

Тогда

$$\hat{c}_j = \frac{1}{|R_j|} \sum_{\mathbf{x}_i \in R_j} y_i. \tag{6}$$

4.1 Пример регрессии

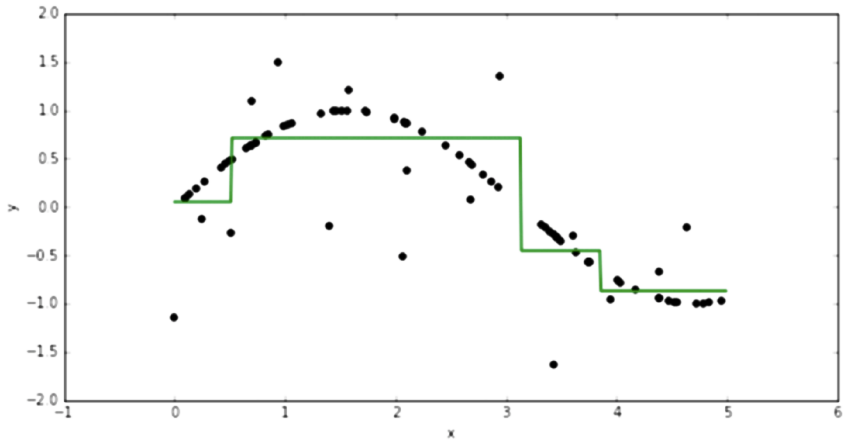

Рис. 3. модель регрессионного дерева решений

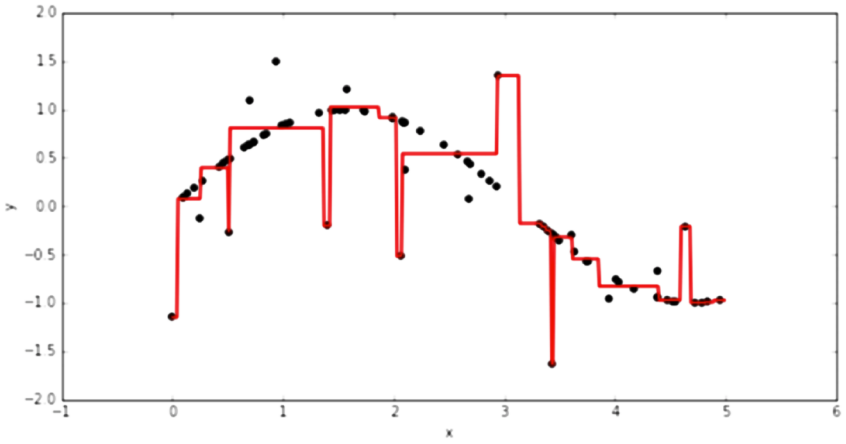

Рис. 4. модель переобученного регрессионного дерева решений

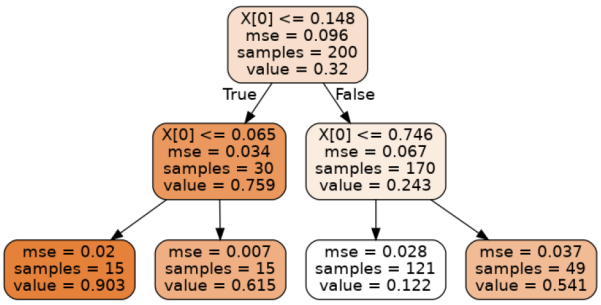

Рис. 5. регрессионное дерево решений

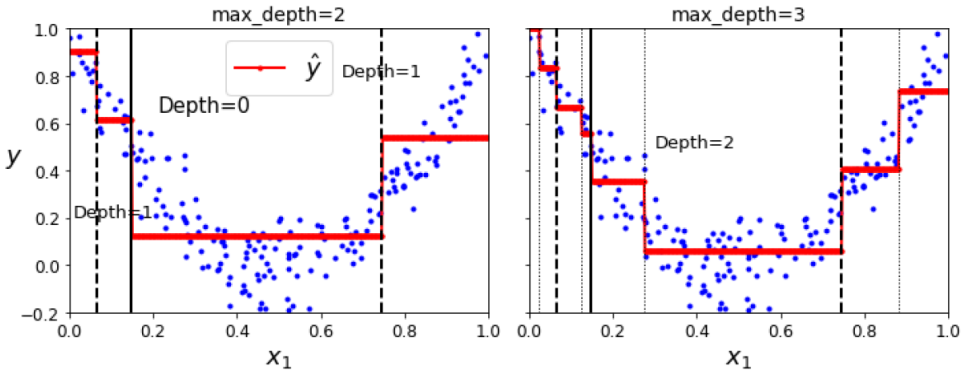

Рис. 6. модели регрессионного дерева решений

5 Классификационные деревья

Модель

$$\varphi(\mathbf{x}_i, \mathbf{\Theta}) = \sum_{j=1}^{J} c_j \mathbb{I}_{(\mathbf{x}_i \in R_j)}. \tag{7}$$

Функция потерь

Обозначим через p_{jk} долю тренировочных индивидов в области R_j из класса $k \in \{1, \cdots, K\}$

$$p_{jk} = \frac{1}{|R_j|} \sum_{\mathbf{x}_i \in R_j} \mathbb{I}_{(y_i = k)}. \tag{8}$$

5.0.1 Частота ошибок классификации

Естественной альтернативой RSS является частота ошибок классификации. это просто часть обучающих наблюдений в этой области, которые не принадлежат к наиболее распространенному классу:

$$E = \frac{1}{|R_j|} \sum_{\mathbf{x}_i \in R_j} \mathbb{I}_{(y_i \neq k)}. \tag{9}$$

5.0.2 Индекс Джинни

Интерпретация: Индекс Джинни является показателем того, как часто случайно выбранный элемент будет классифицировать неверно.

$$G = \sum_{k=1}^{K} p_{jk}(1 - p_{jk}). \tag{10}$$

Индекс Джинни узла измеряет его загрязненность. Узел "чист"($G = 0$) если все обучающие образцы, к которым он применяется, принадлежат одному и тому же классу.

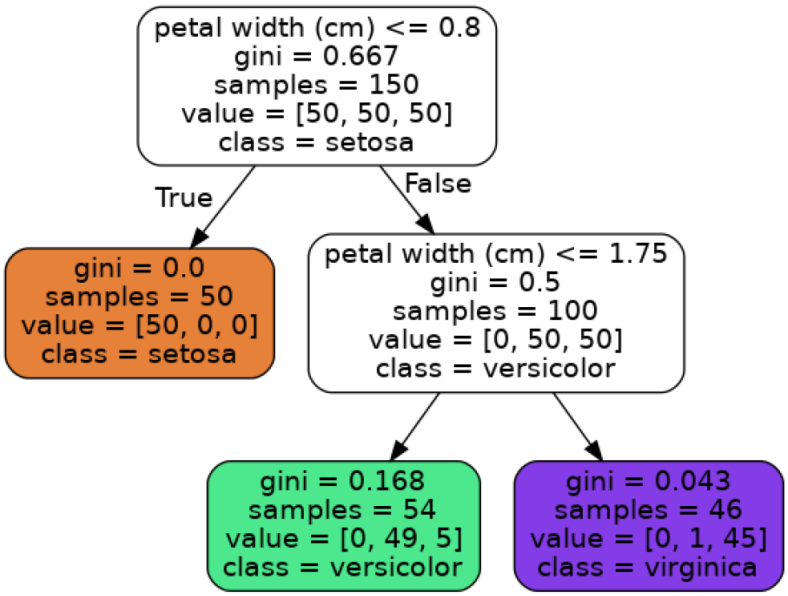

Рис. 7. дерево решений

Рассмотрим, как подсчитывается показатель Джини G_i для i-го узла

$$G = \sum_{k=1}^{K} p_{jk}(1 - p_{jk}),$$

$$G = 1 - \sum_{k=1}^{K} p_{jk}^2,$$

где p_{jk} — доля образцов класса k среди обучающих образцов в i-ом узле.

Узел на глубине 0 имеет Индекс Джини

$$1 - \left(\frac{50}{150}\right)^2 - \left(\frac{50}{150}\right)^2 - \left(\frac{50}{150}\right)^2 = 1 - 3 \cdot \left(\frac{1}{3}\right)^2 = 1 - \frac{1}{3} = 0.666.$$

Загрязненность $G = 0.666$, т.к. 50 индивидов неверно отнесены к *versicolor* и 50 неверно отнесены к *virginica*.

Узел на глубине 1 слева имеет Индекс Джини

$$1 - \left(\frac{50}{50}\right)^2 - 0 - 0 = 0.$$

Загрязненность $G = 0.0$, т.к. все 50 индивидов верно отнесены к *setosa*.

Узел на глубине 1 справа имеет Индекс Джини

$$1 - 0 - \left(\frac{50}{100}\right)^2 - \left(\frac{50}{100}\right)^2 = 0.5.$$

Загрязненность $G = 0.5$, т.к. 50 индивидов неверно классифицированы и отнесены к *virginica*.

Узел на глубине 2 слева имеет Индекс Джини

$$1 - 0 - \left(\frac{49}{54}\right)^2 - \left(\frac{5}{54}\right)^2 = 0.168.$$

Загрязненность $G = 0.168$, т.к. 5 индивидов неверно классифицированы и отнесены к *versicolor*.

Узел на глубине 2 справа имеет Индекс Джини

$$1 - 0 - \left(\frac{1}{46}\right)^2 - \left(\frac{45}{46}\right)^2 = 0.043.$$

Загрязненность $G = 0.043$, т.к. 1 индивид неверно классифицирован и отнесен к *virginica*.

5.0.3 Кросс-энтропия

$$CI = -\sum_{k=1}^{K} p_{jk} \log p_{jk}. \tag{11}$$

Интерпретация: из теории вероятностей известно, что энтропия ограничена снизу нулем, причем минимум достигается на вырожденных распределениях ($p_i = 1, pj = 0$ для $i \neq j$). Максимальное же значение энтропия принимает для равномерного распределения. Отсюда видно, что энтропийный критерий отдает предпочтение более «вырожденным» распределениям классов в вершине.

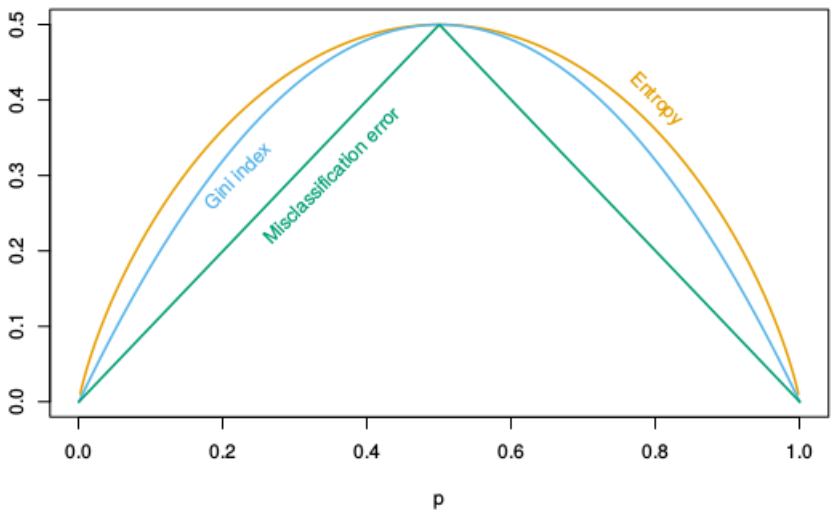

Рис. 8. Загрязненность *impurity* узла для двухклассовой классификации измеряется как доля индивидов p, отнесенных ко второму классу

6 Алгоритмы

6.1 CART (использует Индекс Джини)

Выбираем признак X_j и порог s так, чтобы разбиение $\mathbf{X}$ на

$$R_1(j,s) = \{\mathbf{x}_i \in \mathbf{X} | X_j < s\}$$

и

$$R_2(j,s) = \{\mathbf{x}_i \in \mathbf{X} | X_j \geq s\}$$

решало задачу

$$\sum_{i:\mathbf{x}_i \in R_1(j,s)} (y_i - \hat{c}_1) + \sum_{i:\mathbf{x}_i \in R_2(j,s)} (y_i - \hat{c}_2) \to \min_{j,s} \tag{12}$$

где оценка коэффициента $\hat{c}_j = \dfrac{1}{|R_j|} \sum\limits_{\mathbf{x}_i \in R_j(j,s)} y_i, \quad j = 1, 2.$

1. Перебираем все возможные s_j и выбираем то значение, при котором Индекс Джини минимален.
2. Разбиваем выборку на области R_1 и R_2, образуя две дочерние вершины L_v и R_v.
3. Повторяем процедуру, разбивая каждый из получившихся регионов, пока не будет достигнута максимальная глубина.
4. Алгоритм CART — жадный, он выбирает наилучшее расщепление на текущем уровне, что не обязательно приводит к наименьшей загрязненности на уровнях ниже. Алгоритм хорош, но не всегда оптимален.

6.2 ID3 (использует Кросс-энтропию)

Идея алгоритма заключается в последовательном дроблении выборки на две части до тех пор, пока в каждой части не окажутся объекты только одного класса. Нам необходимо выбирать такой предикат, чтобы ветвление дерева было максимально информативно

1. $\mathbf{X}$ — обучающая выборка, $\mathbf{y} \in \{1, \cdots, k\}$.
2. Если все $\mathbf{x}_i$ имеют класс k, ставим метку 1 в корень и выходим из цикла.
3. Если ни один $\mathbf{x}_i$ не имеет класс k, ставим метку 0 в корень и выходим из цикла.
4. Предикат $R(\mathbf{x}_i) := \{\mathbf{x}_i | X_j \lesseqgtr s_j\}$ для которого информационная выгода наибольшая.

5. Разбиваем $\mathbf{X}$ на $\mathbf{X}_0$ и $\mathbf{X}_1$ по предикату R

$$\mathbf{X}_0 := \{\mathbf{x}_i \in \mathbf{X} : R(\mathbf{x}_i) = 0\},$$

$$\mathbf{X}_1 := \{\mathbf{x}_i \in \mathbf{X} : R(\mathbf{x}_i) = 1\}.$$

6. Если $\mathbf{X}_0 = \varnothing$ или $\mathbf{X}_1 = \varnothing$, создаем новый лист v, k_v — класс, в котором находится большинство элементов $\mathbf{x}_i$.

7. Иначе создаем внутреннюю вершину v:

   (a) $R_v = R$;

   (b) L_v;

   (c) R_v.

6.3 Стрижка деревьев

Описанный выше процесс может дать хорошие прогнозы на обучающем наборе, но, вероятно, *переобучится*, что приведет к плохим результатам на тестовых наборах. Почему?

Меньшее дерево с меньшим количеством разбиений (то есть с меньшим количеством областей $R_1, \cdots, R_J$) может привести к меньшей дисперсии и лучшей интерпретации за счет небольшого смещения. Мы можем пожертвовать смещением, но получить меньшую дисперсию.

Одна из возможных альтернатив описанному выше процессу — выращивать дерево только до тех пор, пока уменьшение RSS из-за каждого разбиения превышает некоторый (высокий) порог.

Эта стратегия приведет к уменьшению размеров деревьев, но она слишком недальновидна: за кажущимся бесполезным разбиением в начале дерева может последовать очень хорошее разбиение — то есть разделение, которое в дальнейшем приводит к значительному сокращению RSS.

Лучшая стратегия — вырастить очень большое дерево T_0, а затем обрезать его, чтобы получить поддерево.

Cost complexity prunning — также называется сокращением наиболее слабых звеньев — используется для этого.

Мы рассматриваем последовательность деревьев, с настраиваемыми параметрами α. Каждому значению α соответствует поддерево $T \subset T_0$ (является подмножеством) такое, что

$$\sum_{j=1}^{|T|} \sum_{\mathbf{x}_i \in R_j} (y_i - \hat{y}_{R_j})^2 + \alpha |T|. \tag{13}$$

настолько мало насколько это возможно. Здесь $|T|$ указывает количество конечных узлов дерева T, R_j — это прямоугольник (**то есть подмножество пространства предикторов**), соответствующий j-му конечному узлу, а $\hat{y}_{R_j}$ — это среднее значение обучающих наблюдений в R_j.

Критерий остановки:

1. Ограничение макс. глубины дерева;
2. Ограничение мин. числа объектов в листе n_{min};
3. Ограничение макс. количества листьев в дереве;
4. Остановка в случае, если все объекты в листе относятся к одному классу.

6.4 Пример классификации: данные о бейсболе

Рассмотрим данные о зарплате в бейсболе. Заработная плата имеет цветовую маркировку от низкой (синий, зеленый) до высокой (желтый, красный).

Для данных **Hitters -** дерево регрессии для прогнозирования логарифма зарплаты бейсболиста на основе количества лет, которые он играл в высшей лиге, и количества попаданий, сделанных им в предыдущем году.

В данном внутреннем узле метка (в форме $X_j < t_k$) указывает левую ветвь, исходящую из этого разделения, а правая ветвь соответствует $X_j \geq t_k$. Например, разделение на вершине дерева приводит к образованию двух больших ветвей. Левая ветвь соответствует $Years < 4.5$, а правая ветвь соответствует $Years \geq 4.5$.

Дерево имеет два внутренних узла и три конечных узла, или три листа. Число на каждом листе — это среднее значение отклика на попадающие туда наблюдения.

В итоге, дерево разбивает или сегментирует игроков на три области пространства предикторов:

$$R_1 = \{X | Years < 4.5\},$$

$$R_2 = \{X | Years \geq 4.5, Hits < 117.5\},$$

$$R_3 = \{X | Years \geq 4.5, Hits \geq 117.5\}.$$

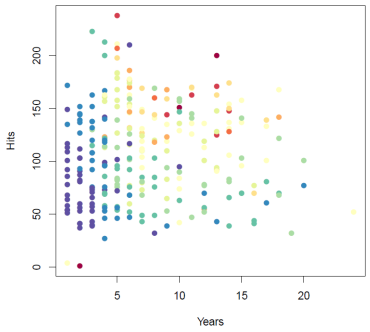

Рис. 9. данные о бейсболе

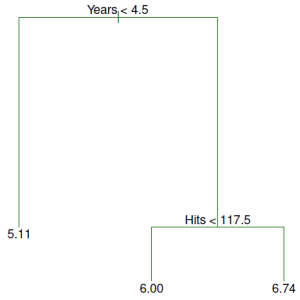

Рис. 10. классификационное дерево

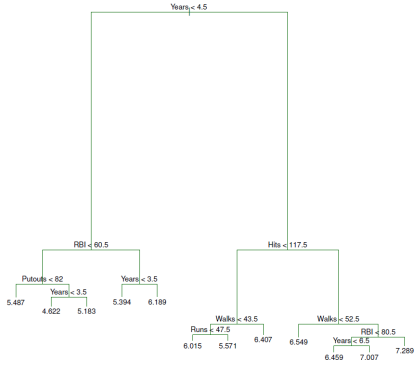

Рис. 11. переобученное классификационное дерево

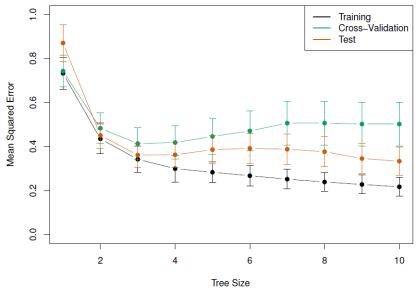

Рис. 12. средневквадратичная ошибка в зависимости от размера дерева

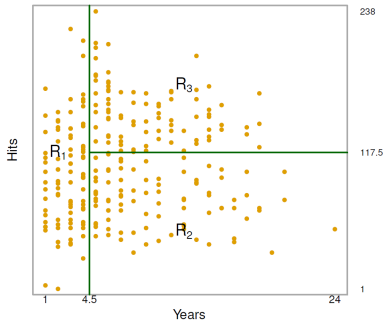

Рис. 13. результат классификации

6.5 Пример классификации: данные о сердце

Эти данные содержат бинарный результат HD для 303 пациентов с болью в груди.

Значение результата Yes указывает на наличие сердечного заболевания на основании ангиографического теста, в то время как No означает отсутствие сердечного заболевания.

Существует 13 предикторов, включая $Age, Sex, Chol$ (измерение холестерина) и другие измерения функции сердца и легких.

Перекрестная проверка дает дерево с шестью конечными узлами.

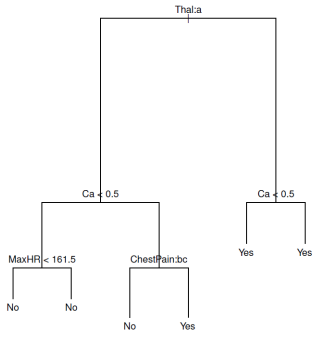

Рис. 14. классификационное дерево

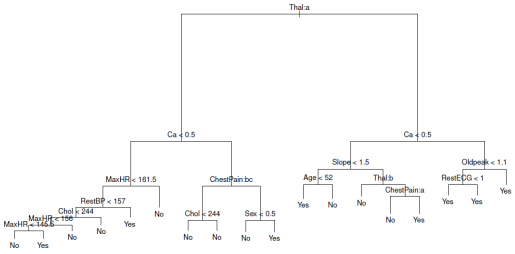

Рис. 15. переобученное классификационное дерево

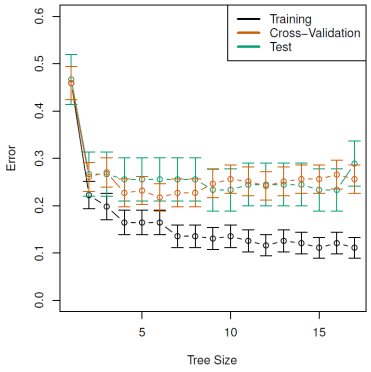

Рис. 16. среднеквадратичная ошибка в зависимости от размера дерева

7 Подведение итогов

7.1 Сравнение деревьев с линейными моделями

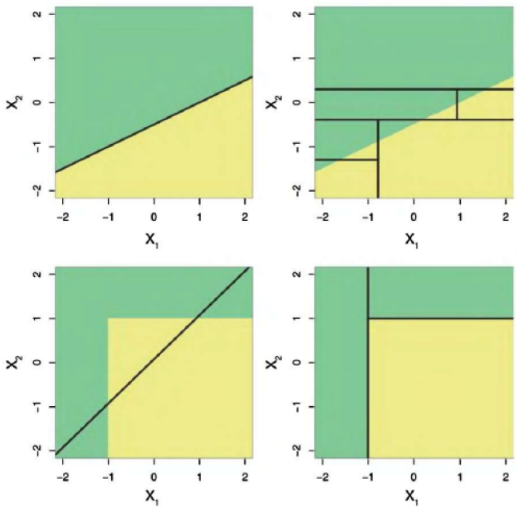

Рис. 17. сравнение моделей

7.2 Преимущества и недостатки рещающих деревьев

Преимущества:
1. Пригодность для задач как классификации, так и регрессии.
2. Легко визуализировать.
3. Легко интерпретировать.
4. Деревья отражают процесс принятия решения человеком.

Недостатки:
1. Невысокая точность.
2. Склонность к переобучению.