

# Exercise 4. Task 4: Interpreting CART Partitions

Daniil Koveh

2025-11-04

## Содержание

<b>1 Теория</b>	<b>1</b>
<b>2 Жизненный пример</b>	<b>1</b>
<b>3 Академическое решение</b>	<b>1</b>
3.1 План решения . . . . .	1
3.2 1. Учебный пример из рисунка слева . . . . .	2
3.3 2. Диаграмма по fitted tree . . . . .	3
3.4 3. Диаграмма регионов по дереву . . . . .	4
3.5 Итог . . . . .	5
3.6 Что запомнить . . . . .	5
3.7 Misc: Как читать сводку дерева . . . . .	5

## 1. Теория

Каждый прямоугольник на плоскости  $(x_1, x_2)$  соответствует листу дерева. Чтобы восстановить дерево, читаем разбиения: на каком признаке и пороге они происходят. Начиная с корня, рисуем ветки с неравенствами, пока не перечислим все прямоугольники. Для построения диаграммы регионов используем `geom_rect`, где для каждого листа задаём диапазоны  $x_1$ ,  $x_2$  и подпись со средним значением отклика.

## 2. Жизненный пример

Сначала разберём учебный пример из рисунка слева (значения 15, 5, 3, 10, 0), а затем воспроизведём реальные регионы по fitted tree. В прикладном контексте можно думать так:  $x_1$  — размер скидки, а  $x_2$  — интенсивность рекламы. Для fitted tree дерево вначале проверяет «реклама выше 2?» — тогда прогноз равен 2.5 условных единиц; если реклама ниже 2, оно смотрит на скидку и далее делит покупателей по уровням  $x_2$  и  $x_1$ , получая средние 0.40, -1.80, -1.10 и 0.12. Прямоугольники визуализируют эти правила.

## 3. Академическое решение

### 3.1. План решения

- Прочитать разбиения на плоскости и восстановить текстовое дерево, фиксируя порядок проверок и значения листьев.
- Построить диаграмму регионов через `geom_rect`, чтобы визуально связать прямоугольники и правила дерева.
- Сформировать таблицу кусочно-постоянной функции, которая показывает итоговые прогнозы в каждой области.

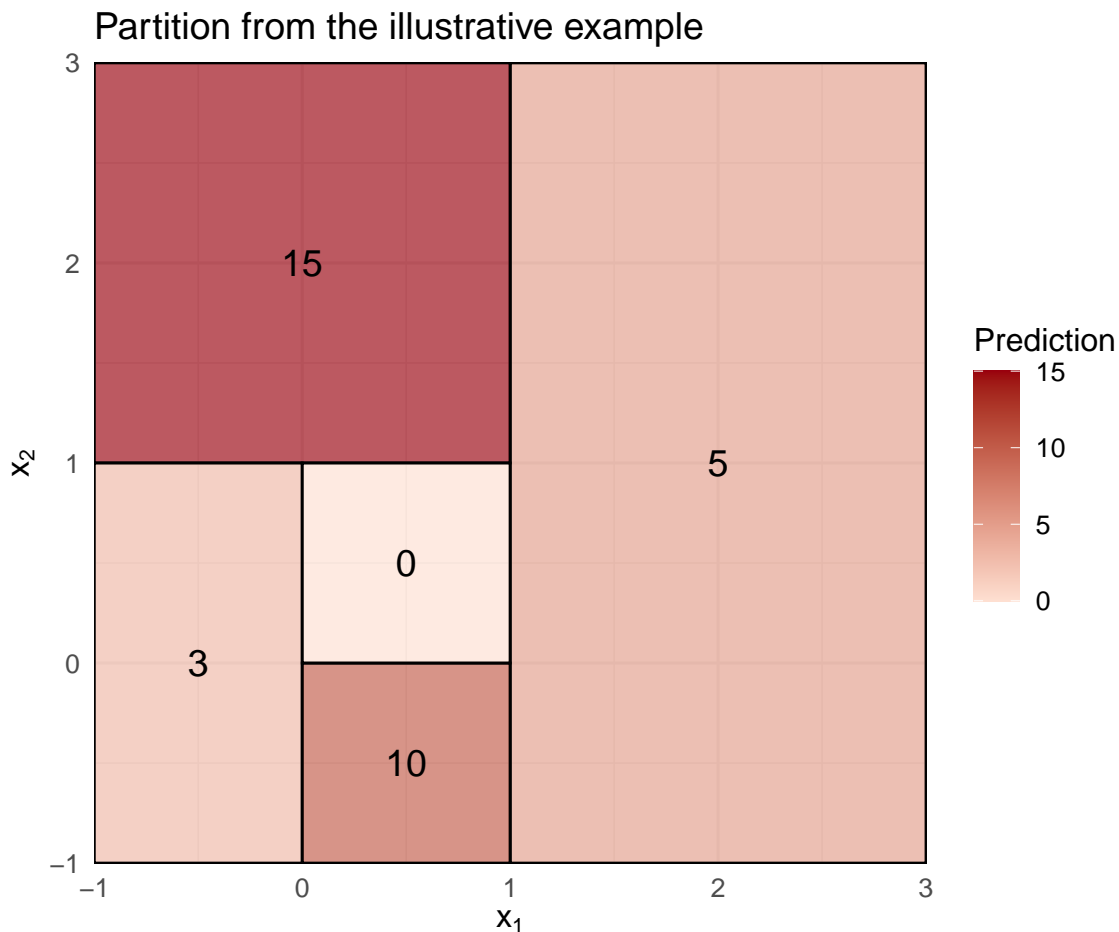
### 3.2. 1. Учебный пример из рисунка слева

Здесь каждая область имеет значения 15, 5, 3, 10 и 0. Логика дерева для этой иллюстрации:

Root: split on  $x_1 < 1$   $\vdash$  If  $x_1 \geq 1 \rightarrow$  predict 5  $\vdash$  If  $x_1 < 1$ :  $\vdash$  If  $x_2 \geq 1 \rightarrow$  predict 15  $\vdash$  If  $x_2 < 1$ :  $\vdash$  If  $x_2 < 0 \rightarrow$  predict 10  $\vdash$  If  $0 \leq x_2 < 1$ :  $\vdash$  If  $x_1 < 0 \rightarrow$  predict 3  $\vdash$  If  $0 \leq x_1 < 1 \rightarrow$  predict 0

```
library(ggplot2) # подключаем ggplot2
region_map_old <- data.frame(
  xmin = c(1, -1, -1, 0, 0),
  xmax = c(3, 1, 0, 1, 1),
  ymin = c(-1, 1, -1, -1, 0),
  ymax = c(3, 3, 1, 0, 1),
  value = c(5, 15, 3, 10, 0)
)
region_map_old$label <- format(region_map_old$value, trim = TRUE)

ggplot(region_map_old) +
  geom_rect(aes(xmin = xmin, xmax = xmax, ymin = ymin, ymax = ymax, fill = value),
    color = "black", alpha = 0.65) +
  geom_text(aes(x = (xmin + xmax) / 2, y = (ymin + ymax) / 2, label = label),
    size = 5) +
  scale_fill_gradient(
    low = "#fee0d2",
    high = "#99000d",
    name = "Prediction"
  ) +
  coord_cartesian(xlim = c(-1, 3), ylim = c(-1, 3), expand = FALSE) +
  labs(title = "Partition from the illustrative example",
    x = expression(x[1]), y = expression(x[2])) +
  theme_minimal(base_size = 12) +
  theme(legend.position = "right")
```



### 3.3. 2. Диаграмма по fitted tree

По структуре реального дерева (листья со средними -1.80, -1.10, 0.12, 0.40 и 2.50) получаем последовательность проверок:

Root: split on  $x_2 < 2$  — If  $x_2 \geq 2 \rightarrow$  predict 2.50 — If  $x_2 < 2$ : — If  $x_1 \geq 1 \rightarrow$  predict 0.40 — If  $x_1 < 1$ : — If  $x_2 < 1 \rightarrow$  predict -1.80 — If  $x_2 \geq 1$ : — If  $x_1 < 0.0003 \rightarrow$  predict -1.10 — If  $x_1 \geq 0.0003 \rightarrow$  predict 0.12

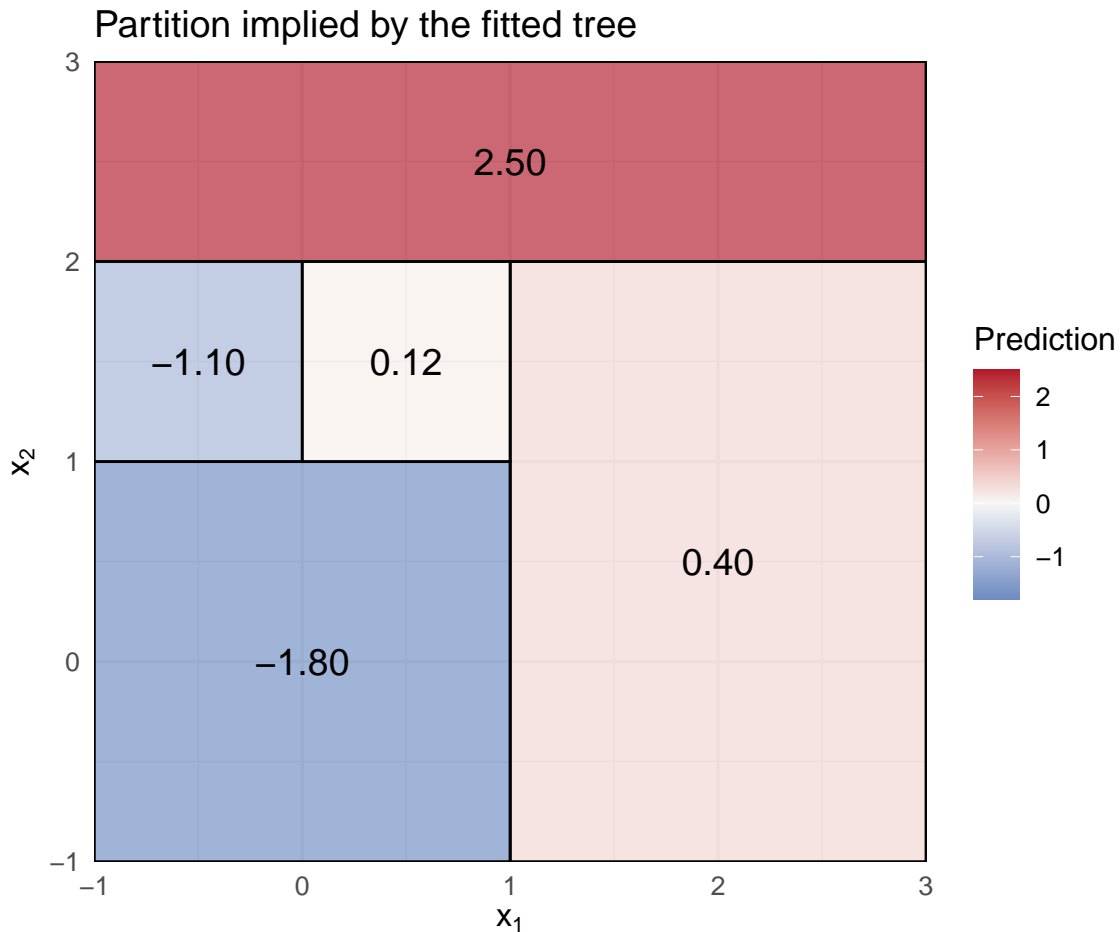
```
region_map <- data.frame(
  xmin = c(-1, 1, -1, -1, 0.0003),
  xmax = c(3, 3, 1, 0.0003, 1),
  ymin = c(2, -1, -1, 1, 1),
  ymax = c(3, 2, 1, 2, 2),
  value = c(2.50, 0.40, -1.80, -1.10, 0.12)
)
region_map$label <- format(region_map$value, trim = TRUE)

ggplot(region_map) +
  geom_rect(aes(xmin = xmin, xmax = xmax, ymin = ymin, ymax = ymax, fill = value),
    color = "black", alpha = 0.65) +
  geom_text(aes(x = (xmin + xmax) / 2, y = (ymin + ymax) / 2, label = label),
    size = 5) +
  scale_fill_gradient2(
    low = "#2166ac",
```

```

mid = "#f7f7f7",
high = "#b2182b",
midpoint = 0,
name = "Prediction"
) +
coord_cartesian(xlim = c(-1, 3), ylim = c(-1, 3), expand = FALSE) +
labs(title = "Partition implied by the fitted tree",
     x = expression(x[1]), y = expression(x[2])) +
theme_minimal(base_size = 12) +
theme(legend.position = "right")

```



### 3.4. 3. Диаграмма регионов по дереву

```

fitted_function <- data.frame(
  Region = c("x2 ≥ 2",
             "x2 < 2, x1 ≥ 1",
             "x1 < 1, x2 < 1",
             "x1 < 0.0003, 1 ≤ x2 < 2",
             "0.0003 ≤ x1 < 1, 1 ≤ x2 < 2"),
  MeanY = c(2.50, 0.40, -1.80, -1.10, 0.12)
)
fitted_function

```

```
##                               Region MeanY
## 1                               x2 ≥ 2  2.50
## 2                   x2 < 2, x1 ≥ 1  0.40
## 3                   x1 < 1, x2 < 1 -1.80
## 4      x1 < 0.0003, 1 ≤ x2 < 2 -1.10
## 5 0.0003 ≤ x1 < 1, 1 ≤ x2 < 2  0.12
```

### 3.5. Итог

- Обе диаграммы теперь сохранены: первая повторяет учебный пример (значения 15, 5, 3, 10, 0), вторая построена по реальному дереву с усреднениями (-1.80, -1.10, 0.12, 0.40, 2.50).
- Плоская диаграмма для fitted tree строится из порогов  $x_2 = 2$ ,  $x_1 = 1$ ,  $x_2 = 1$  и  $x_1 = 0,0003$ , что создаёт пять регионов с их средними.
- Оси растянуты до диапазона  $[-1, 3]$ , чтобы повторить масштаб исходной иллюстрации.
- Таблица fitted\_function перечисляет кусочно-постоянную модель, полученную из листьев дерева.

### 3.6. Что запомнить

- Геометрия разбиений и структура дерева эквивалентны: прямоугольники на плоскости полностью определяют последовательность сплитов.
- Визуализация через geom\_rect помогает объяснить бизнес-правила, лежащие внутри дерева.
- Таблица листьев — быстрый способ описать итоговую функцию без рисунков и графиков.

### 3.7. Misc: Как читать сводку дерева

Запись 1) root 500 1500.0 0.41 расшифровывается так:

- 1) — номер узла. Чётные и нечётные номера показывают левые и правые потомки: у узла 4 есть дети 8 и 9, у узла 2 — дети 4 и 5.
- root — подпись узла. Внутренние узлы содержат условие ( $x_2 < 2$ ), листья отмечены \*.
- 500 — сколько наблюдений попало в узел. В корне — вся выборка, в листьях — размер сегмента.
- 1500.0 — сумма квадратов ошибок (deviance) внутри узла. Чем меньше число, тем однороднее отклики.
- 0.41 — прогноз узла: среднее значение целевой переменной для наблюдений внутри.

Строка 8)  $x_1 < 0.0003$  52 6.0 -1.10 \* читается так: узел №8 проверяет условие  $x_1 < 0.0003$ , содержит 52 наблюдения, их внутренняя ошибка равна 6.0, а прогноз по узлу — -1.10. Звёздочка показывает, что это конечный лист.