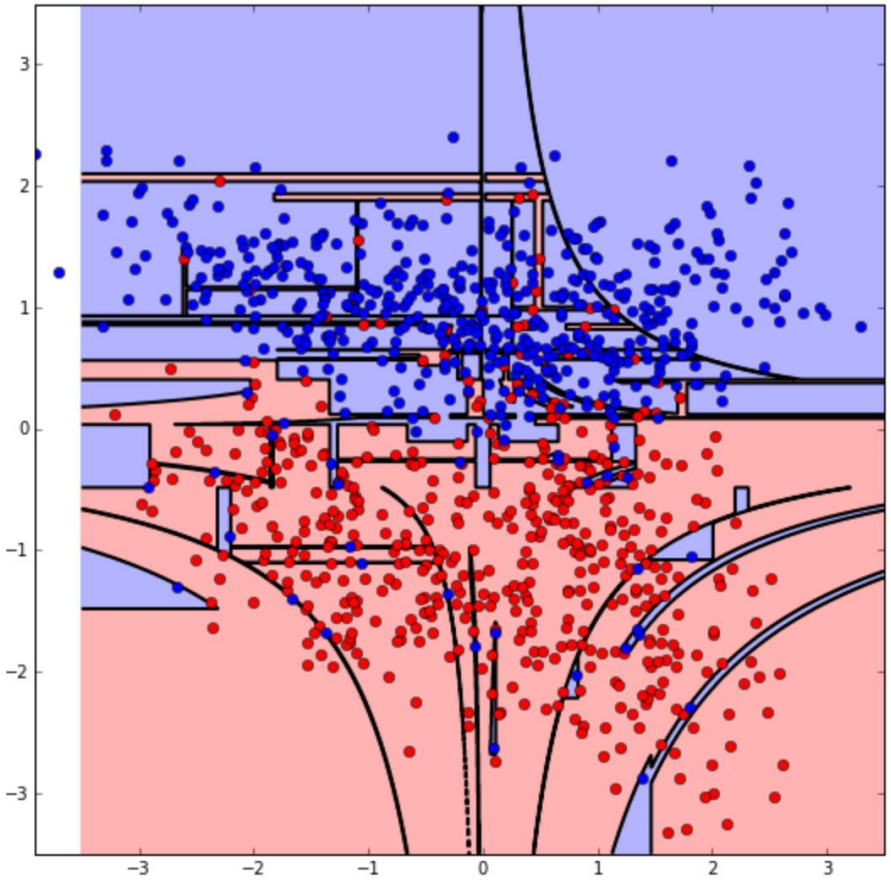
Введение в анализ данных

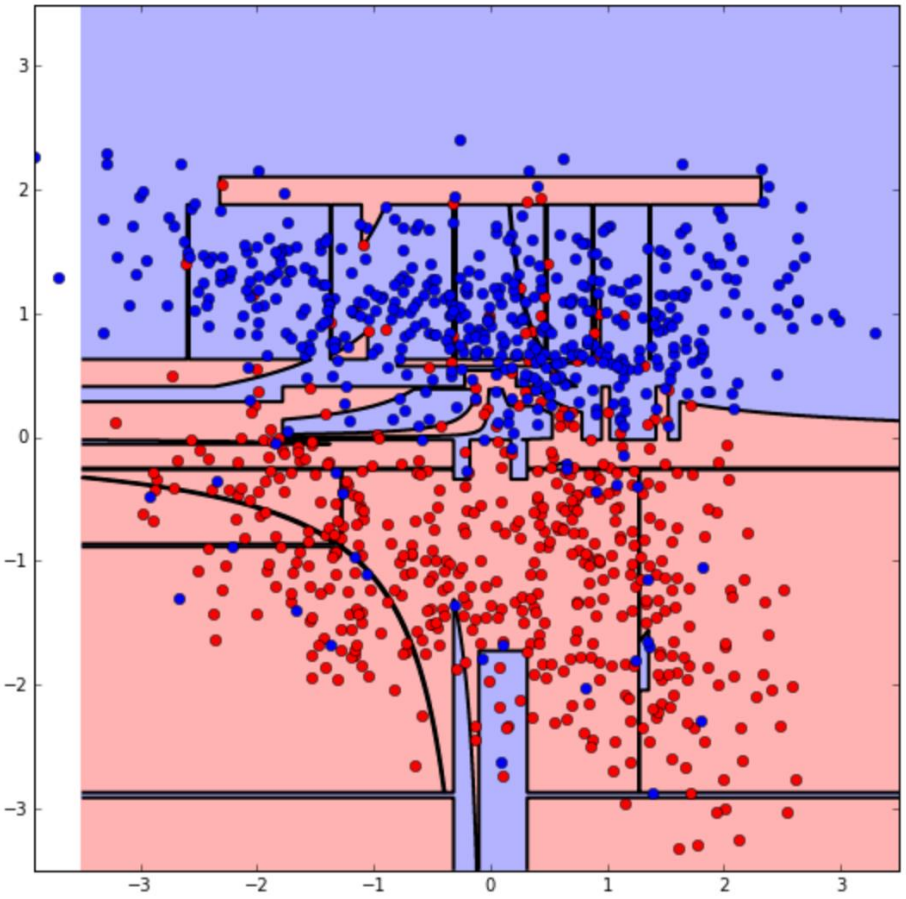
Решающие деревья и случайные леса

Слайды в основном Евгения Соколова

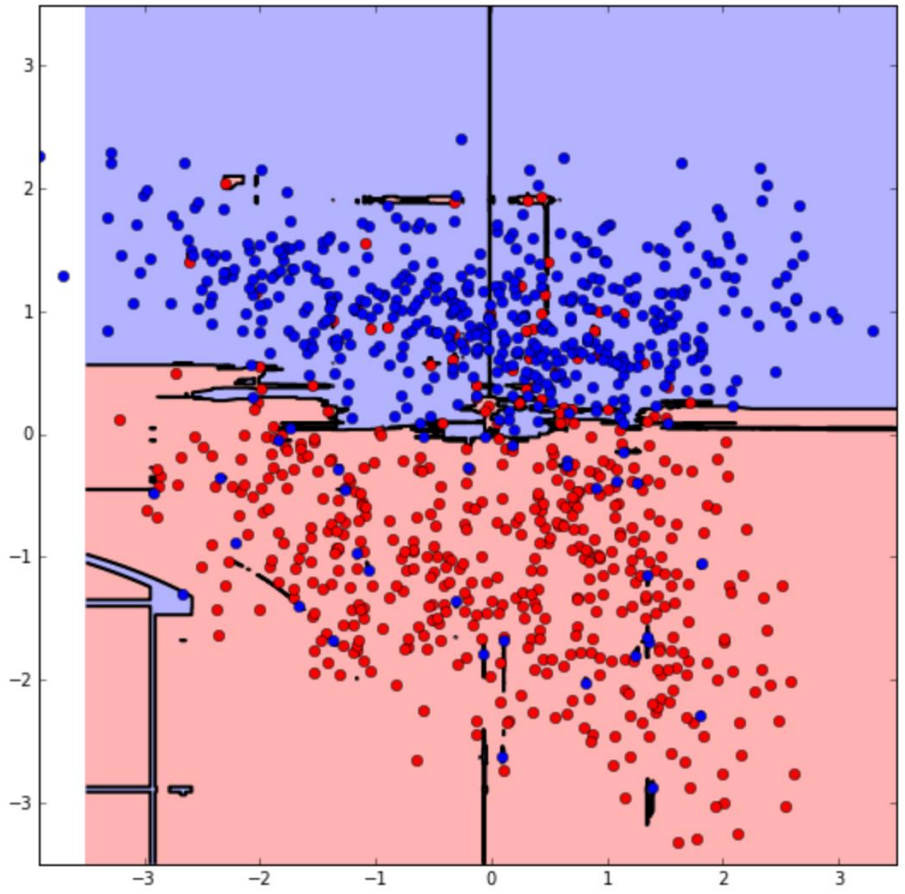
НИУ ВШЭ, 2020

Неустойчивость деревьев





Усреднение деревьев



Композиции алгоритмов

Основная идея

- Объединение нескольких моделей воедино может привести к созданию гораздо более мощной модели.

Majority Vote

Majority vote

- Дано: N базовых алгоритмов $b_1(x), \ldots, b_N(x)$
- Каждый хотя бы немного лучше случайного угадывания
- Композиция: класс, за который проголосовало больше всего базовых алгоритмов

$$a(x) = \arg\max_{y \in \mathbb{Y}} \sum_{n=1}^{N} [b_n(x) = y]$$

Усреднение наблюдений

- Наблюдение: усреднение результатов повышает их точность
- Измерение артериального давления
- Измерение скорости света
- Усреднение соседних пикселей изображения

Усреднение наблюдений

- Дано: N базовых алгоритмов $b_1(x), \ldots, b_N(x)$
- Каждый хотя бы немного лучше случайного угадывания
- Композиция:

$$a(x) = \frac{1}{N}\sum_{n=1}^{N} b_n(x)$$

Композиции алгоритмов

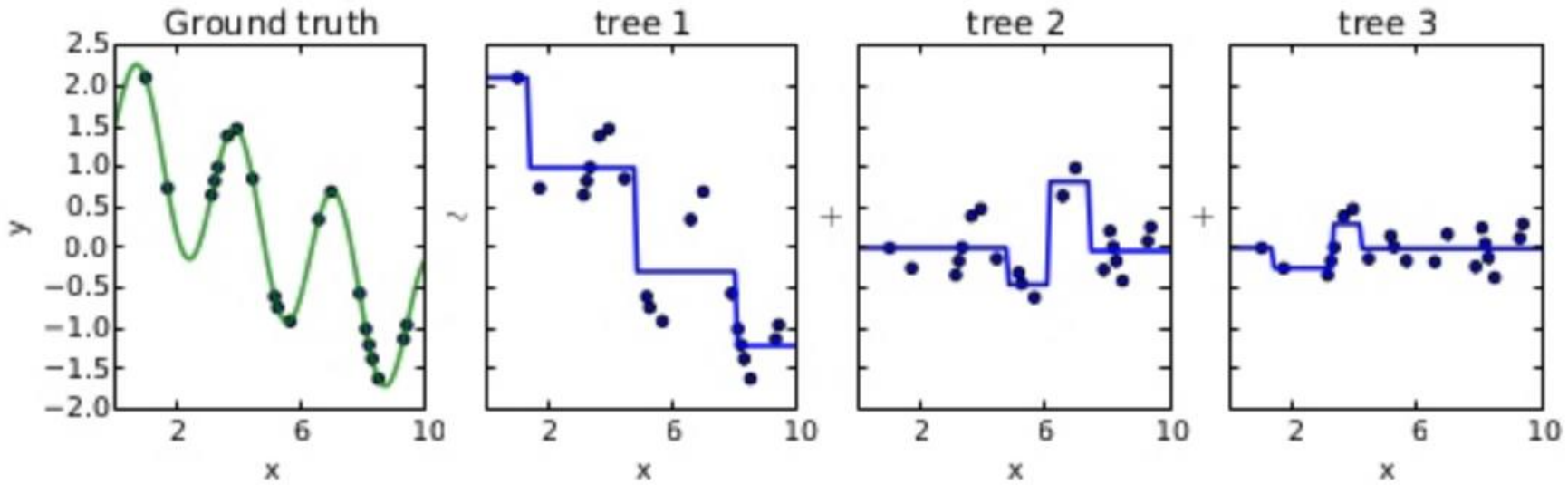
- Базовые алгоритмы: $b_1(x), \ldots, b_N(x)$
- Композиция: $a(x)$

- Как по одной и той же выборке обучить N различных моделей?

Бустинг

- Каждый следующий алгоритм исправляет ошибки предыдущих

- Яркий пример: градиентный бустинг над решающими деревьями
- В следующий раз



Бэггинг

- Bagging (Bootstrap Aggregation)
- Базовые алгоритмы обучаются независимо
- Каждый обучается на подмножестве данных
- Усреднение ответов или выбор по большинству

- Яркий пример: случайный лес (random forest)

Бэггинг

Идея:

- Обучим много деревьев $b_1(x), \dots, b_N(x)$
- Выберем ответ по большинству:

$$a(x) = \arg\max_{y \in \mathbb{Y}} \sum_{n=1}^{N} [b_n(x) = y]$$

Пример

- Прогнозы деревьев: $-1, -1, 1, -1, 1, -1$

$$a(x) = ?$$

Пример

- Прогнозы деревьев: $-1, -1, 1, -1, 1, -1$

$$a(x) = -1$$

Рандомизация

- Как сделать деревья разными?
- Обучать по подвыборкам!

Рандомизация

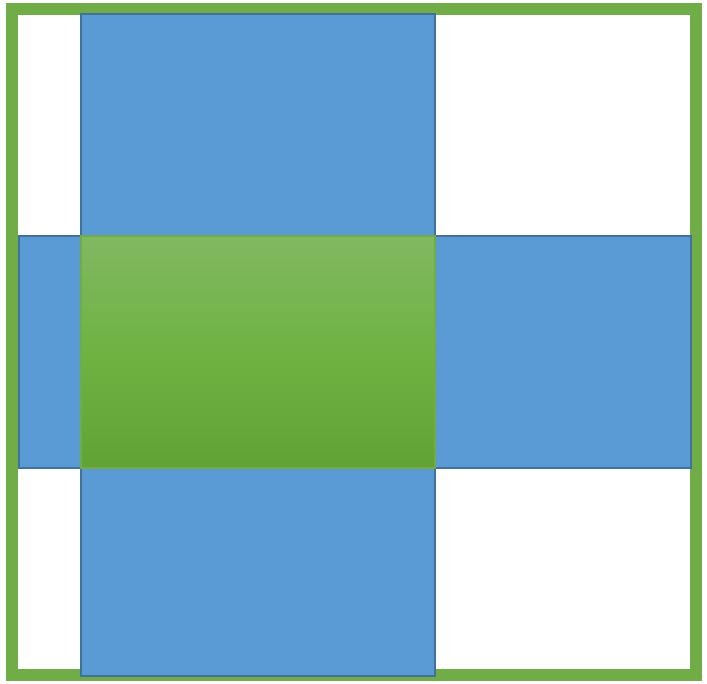
- Популярный подход: бутстрап
- Выбираем из обучающей выборки ℓ объектов с возвращением
- Пример: $\{x_1, x_2, x_3, x_4\} \to \{x_1, x_2, x_2, x_4\}$
- Примерно $0.632 * \ell$ различных объектов

Рандомизация

- Другой подход: выбор случайного подмножества объектов
- Гиперпараметр: размер подмножества

Виды рандомизации

- Бэггинг: обучаем на случайной подвыборке
- Метод случайных подпространств: обучаем на случайном подмножестве признаков
- Размер подвыборки/подмножества — гиперпараметр



Рандомизация

- Этого недостаточно
- Как можно рандомизировать сам процесс построения дерева?

Поиск разбиения

- Пусть в вершине m оказалась выборка X_m
- $Q(X_m, j, t)$ — критерий ошибки условия $[x^j \leq t]$
- Ищем лучшие параметры j и t перебором:

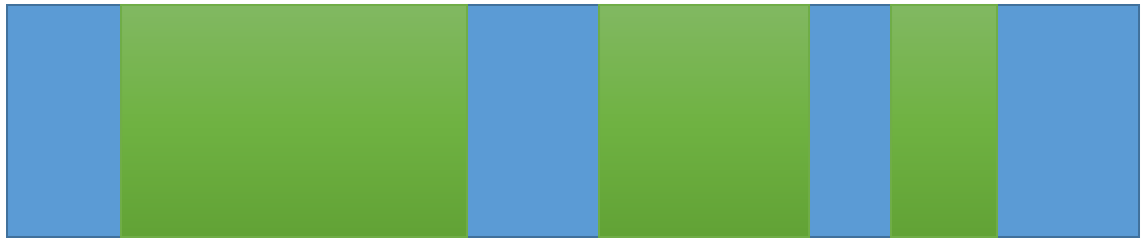
$$Q(X_m, j, t) \to \min_{j,t}$$

Поиск разбиения

- Пусть в вершине m оказалась выборка X_m
- $Q(X_m, j, t)$ — критерий ошибки условия $[x^j \leq t]$
- Ищем лучшие параметры j и t перебором:

$$Q(X_m, j, t) \to \min_{j,t}$$

- Случайный лес: выбираем j из случайного подмножества признаков размера q



Корреляция между деревьями

Рекомендации для q:

- Регрессия: $q = \frac{d}{3}$
- Классификация: $q = \sqrt{d}$

Случайный лес (Random forest)

1. Для $n = 1, \dots, N$:
2. Сгенерировать выборку $\tilde{X}$ с помощью бутстрапа
3. Построить решающее дерево $b_n(x)$ по выборке $\tilde{X}$
4. Дерево строится, пока в каждом листе не окажется не более n_{min} объектов
5. Оптимальное разбиение ищется среди q случайных признаков

Случайный лес (Random forest)

1. Для $n = 1, \dots, N$:
2. Сгенерировать выборку $\tilde{X}$ с помощью бутстрапа
3. Построить решающее дерево $b_n(x)$ по выборке $\tilde{X}$
4. Дерево строится, пока в каждом листе не окажется не более n_{min} объектов
5. Оптимальное разбиение ищется среди q случайных признаков

Выбираются заново при каждом разбиении!

Случайный лес

- Регрессия:

$$a(x) = \frac{1}{N}\sum_{n=1}^{N} b_n(x)$$

Среднее арифметическое ответов всех деревьев

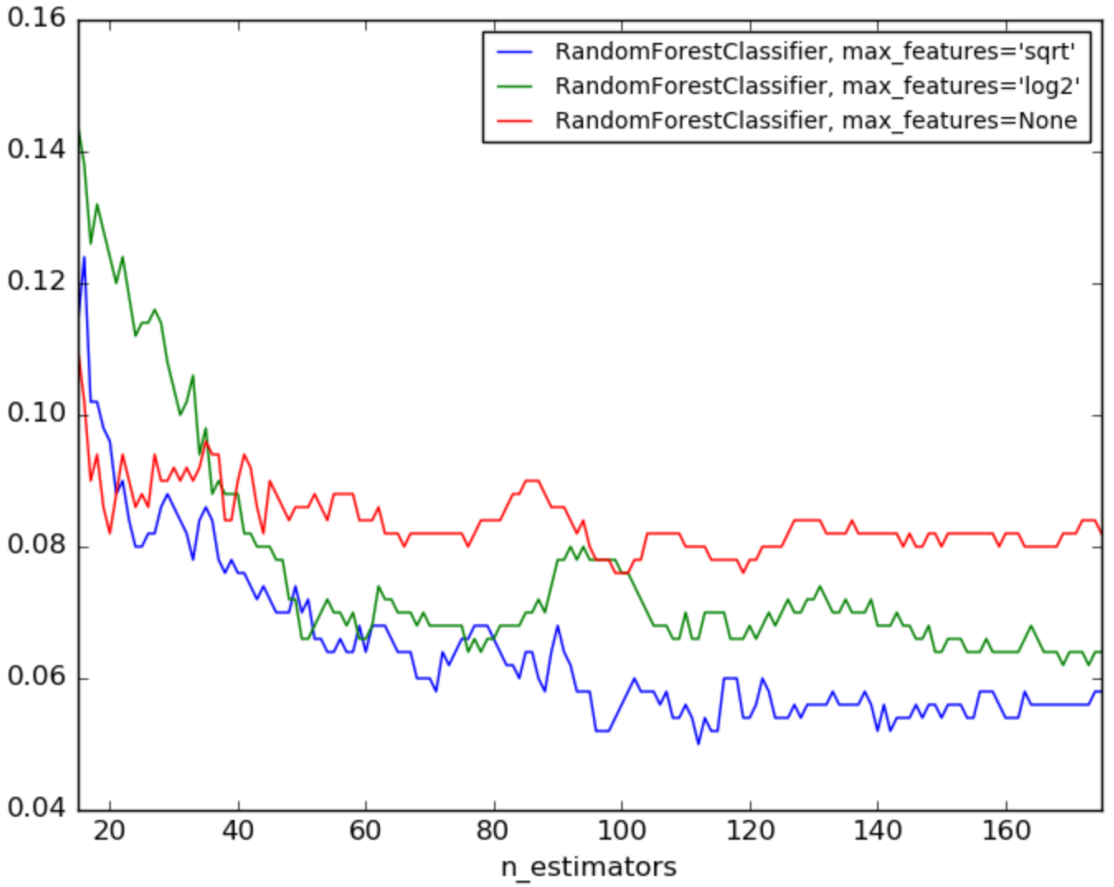
- Классификация:

$$a(x) = \arg\max_{y \in \mathbb{Y}} \sum_{n=1}^{N} [b_n(x) = y]$$

Класс, который предсказало большинстов деревьев

Ошибка на тесте

- Ошибка сначала убывает, а затем остаётся примерно на одном уровне
- Случайный лес не переобучается при росте N



Out-of-bag

- Каждое дерево обучается примерно на 63% данных
- Остальные объекты — как бы тестовая выборка для этого дерева
- X_n — обучающая выборка для $b_n(x)$
- Можно оценить ошибку на новых данных:

$$Q_{test} = \frac{1}{\ell} \sum_{i=1}^{\ell} \frac{1}{\sum_{n=1}^{N} [x_i \notin X_n]} \sum_{n=1}^{N} [x_i \notin X_n] \, b_n(x_i)$$

Out-of-bag

- Для каждого объекта обучающей выборки

  сделали предсказание всеми деревьями,

  в которых этот объект не был использован при обучении.
- Посчитали среднюю ошибку для каждого объекта по всем полученным предсказаниям.
- Посчитали среднюю ошибку по всем объектам.

Out-of-bag

- Оценить качество, если мало данных
- Подобрать значение гиперпараметров

Важность признаков

Перестановочный метод:

- Проверяем важность j-го признака
- Перемешиваем соответствующий столбец в матрице «объекты-признаки» для тестовой выборки
- Измеряем качество модели
- Если оно слабо изменилось, то признак не очень важный