ВЕБИНАРНЫЙ ФОРМАТ

Библиотеки Python для Data Science: продолжение

Урок 3.
Построение модели классификации.

Светлана Медведева

План урока

- Балансировка классов
- Схемы оценки обобщающей способности алгоритма
- Обзор алгоритмов классификации:

- Логистическая регрессия
- Метод опорных векторов
- k ближайших соседей
- Случайный лес и бустинговые алгоритмы

- Практическая часть

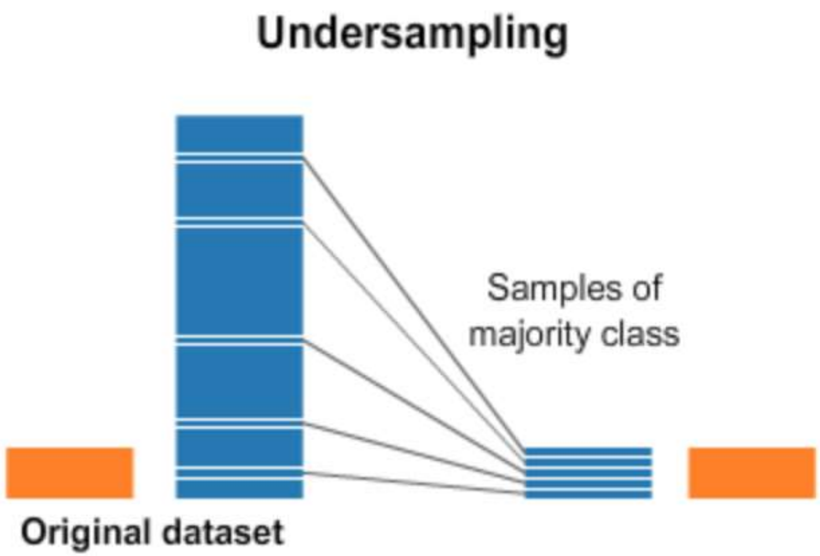
Балансировка классов

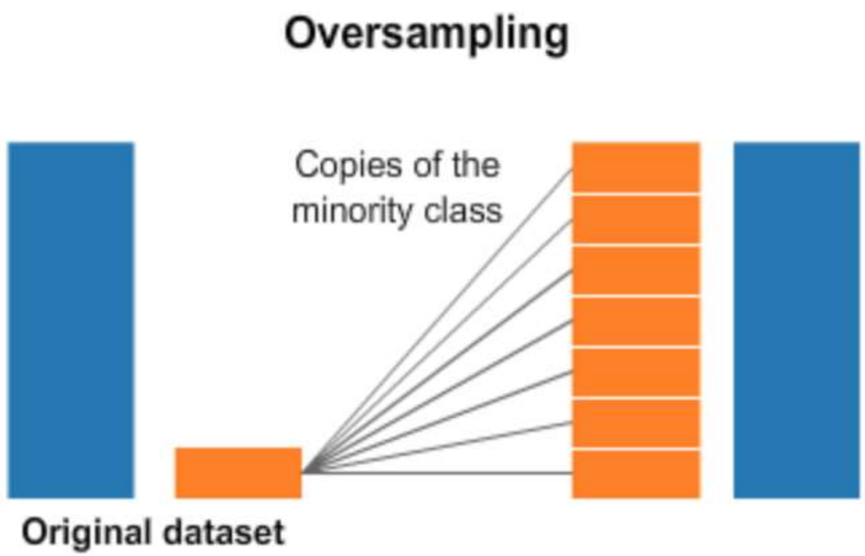
Собрать больше данных

Выбрать подходящую метрику качества

Попробовать разные модели, одни модели более устойчивы к несбалансированным данным, чем другие

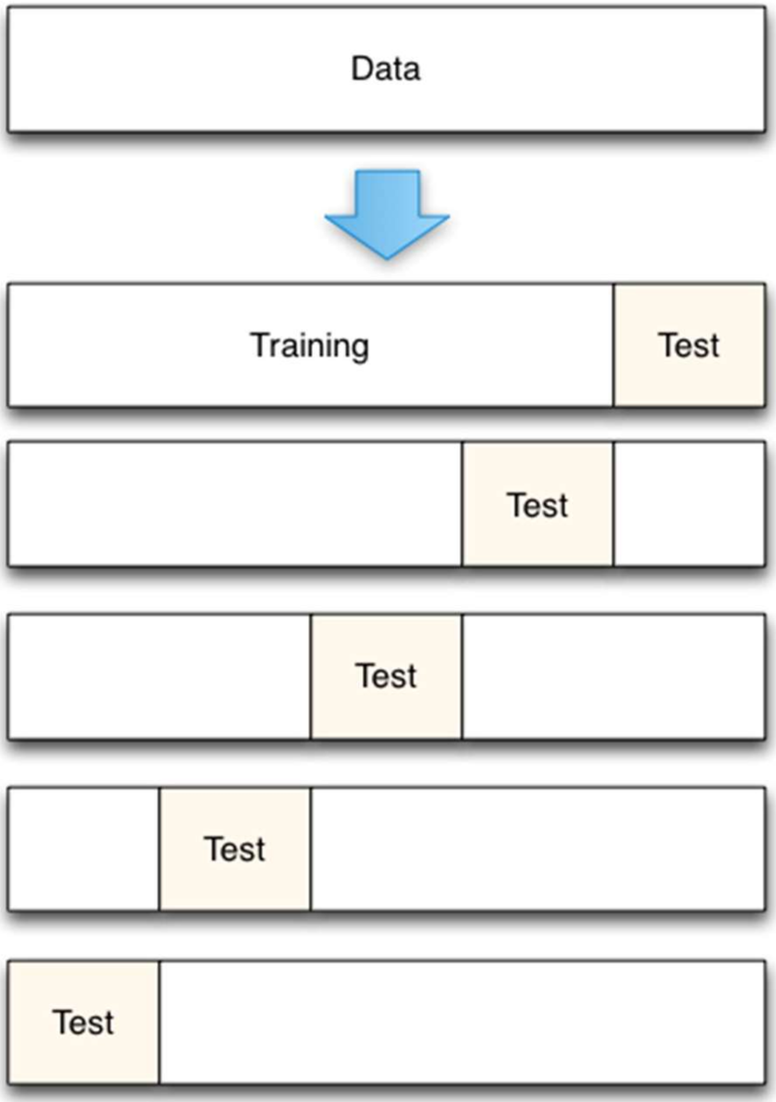
- Штраф за ошибки при прогнозе меньшего класса
- Undersampling и Oversampling
- Создание синтетических примеров для меньшего класса





Модель, обладающая хорошей **обобщающей способностью**, способна предсказывать примерно на одном и том же уровне качества, как на обучающем датасете, так и на новых данных.

- **Отложенная выборка**
- **Кросс-валидация**



Обзор алгоритмов классификации: Логистическая регрессия

Логистическая регрессия или логит-модель (англ. *logit model*) - это статистическая модель, используемая для прогнозирования вероятности возникновения некоторого события путём его сравнения с логистической кривой.

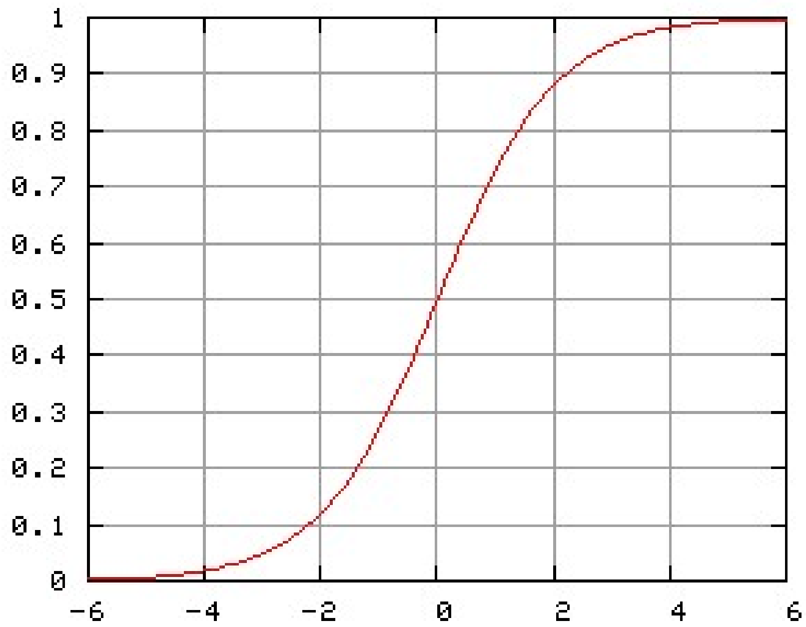
$$P\{y = 1|x\} = f(z),$$

где $z = \theta^T x = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$

$$f(x) = \frac{1}{1 + e^{-x}}$$

$$P\{y = 0|x\} = 1 - f(z) = 1 - f(\theta^T x)$$

$$P\{y|x\} = f(\theta^T x)^y \, (1 - f(\theta^T x))^{1-y}, y \in \{0,1\}$$



Логистическая регрессия: подбор параметров

Метод максимального правдоподобия

$$\hat{\theta} = argmax_{\theta}\mathrm{L}(\theta) = argmax_{\theta}\prod_{i=1}^{m} P\{y = y^{(i)} | x = x^{(i)}\}$$

$$ln\mathrm{L}(\theta) = \sum_{i=1}^{m} ln_{\theta} \prod_{i=1}^{m} P\{y = y^{(i)} | x = x^{(i)}\} = \sum_{i=1}^{m} y^{(i)} \ln f\left(\theta^{T} x^{i}\right) + (1 - y^{(i)}) \ln(1 - f\left(\theta^{T} x^{i}\right))$$

где $\theta^{T}\mathrm{x}^{(i)} = \theta_0 + \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)} + \cdots + \theta_n x_n^{(i)}$

Максимизация функции правдоподобия

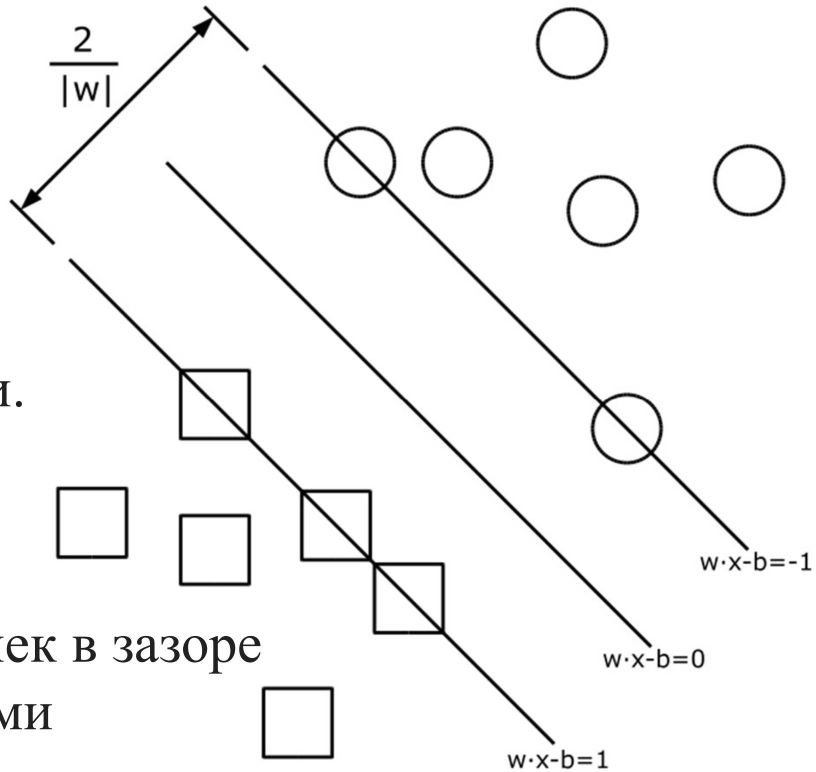
метод градиентного спуска:

$$\theta := \theta + \alpha \nabla lnL\theta$$

Обзор алгоритмов классификации: Метод опорных векторов

Постановка задачи

$$\{(\boldsymbol{x_1}, c_1), (\boldsymbol{x_2}, c_2), \dots, (\boldsymbol{x_n}, c_n)\}$$



$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$$
$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 1$$
$$\boldsymbol{w} \cdot \boldsymbol{x} + b = -1$$

Оптимальная разделяющая гиперплоскость и || ей плоскости.

$$\left[\begin{array}{l} \mathbf{w} \cdot \mathbf{x_i} - b \geq 1,\ c_i = 1 \\ \mathbf{w} \cdot \mathbf{x_i} - b \leq -1,\ c_i = -1 \end{array}\right.$$

Условие отсутствия точек в зазоре между гиперплоскостями

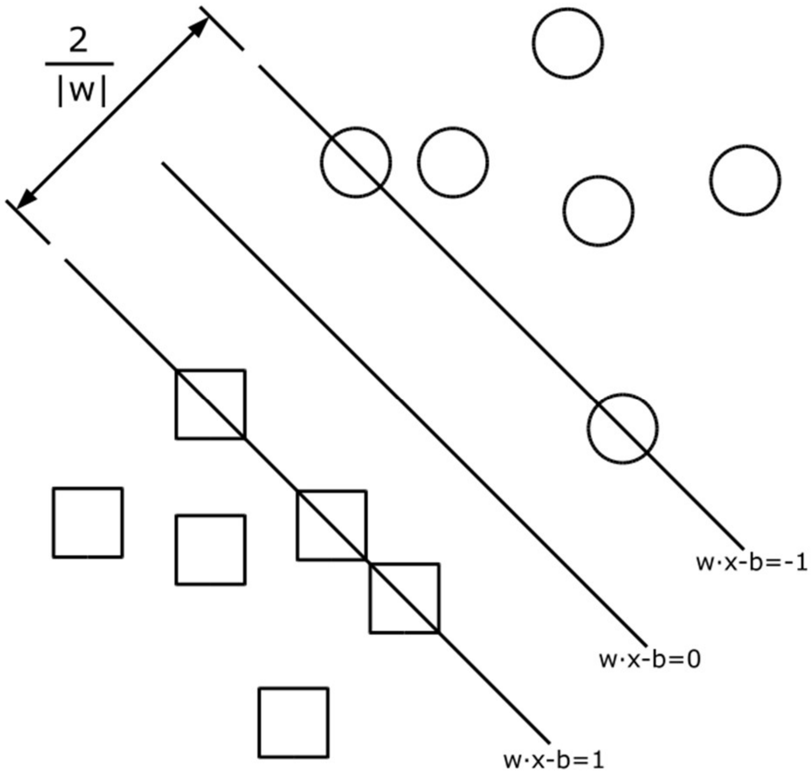
Обзор алгоритмов классификации: Метод опорных векторов

Случай линейной разделимости

$$\begin{cases} \|\mathbf{w}\|^2 \to \min \\ c_i(\mathbf{w}\cdot\mathbf{x_i} - b) \geq 1, \quad 1 \leq i \leq n. \end{cases}$$

Случай линейной неразделимости

$$\begin{cases} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^n \xi_i \to \min_{w,b,\xi_i} \\ c_i(\mathbf{w}\cdot\mathbf{x_i} - b) \geq 1 - \xi_i, \quad 1 \leq i \leq n \\ \xi_i \geq 0, \quad 1 \leq i \leq n \end{cases}$$



Обзор алгоритмов классификации: k ближайших соседей

Метод k-ближайших соседей (англ. k-nearest neighbors algorithm, k-NN) — метрический алгоритм для автоматической классификации объектов или регрессии.

Минимакс-нормализация:

$$x' = (x - \min[X])/(\max[X] - \min[X])$$

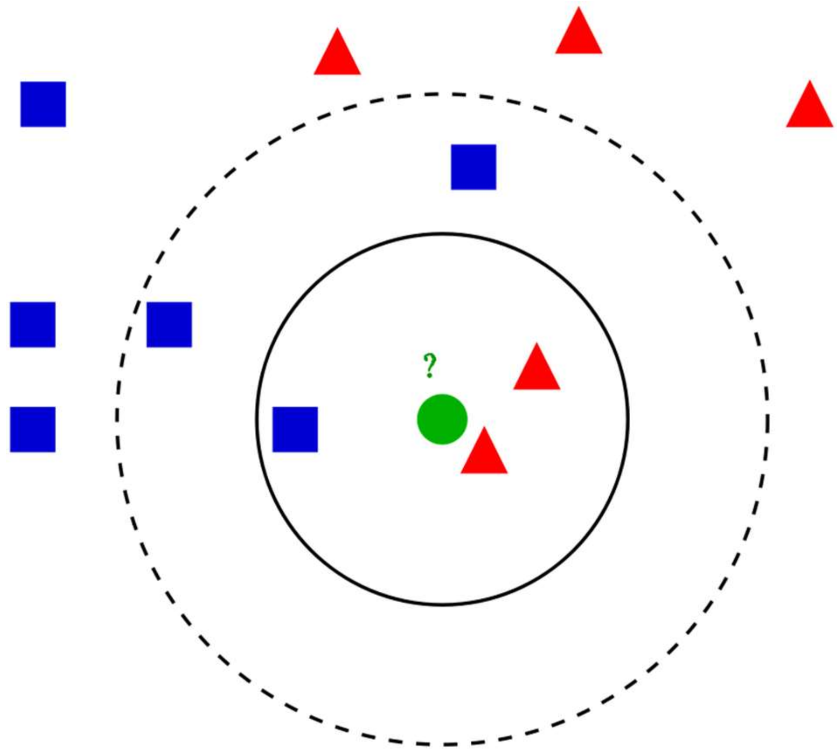
(0,1)

Z-нормализация:

$$x' = (x - M[X])/\sigma[X]$$

(-3σ, 3σ)

Z-нормализация:



Обзор алгоритмов классификации:
k ближайших соседей

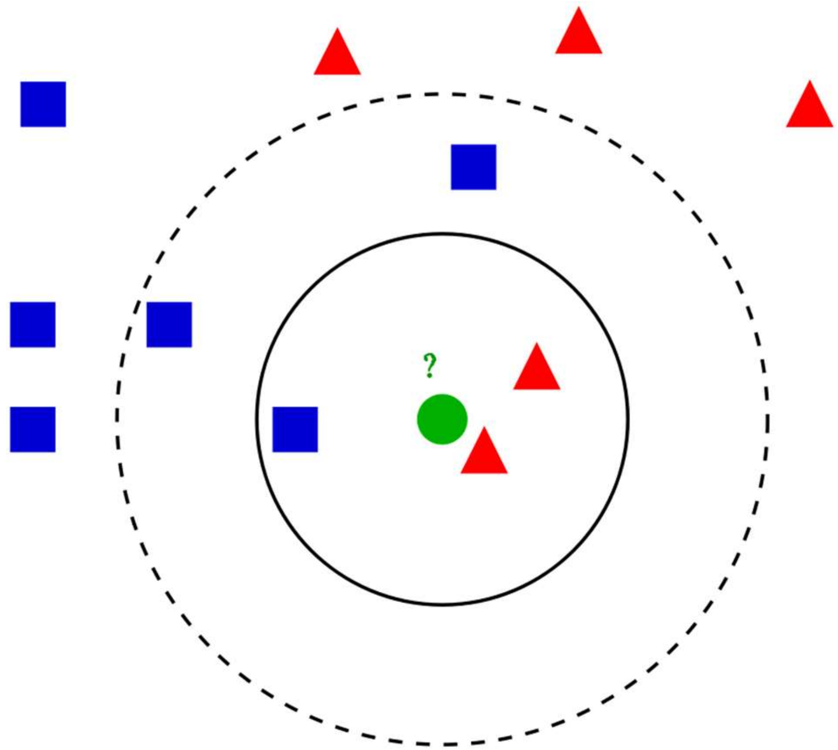
Метод k-ближайших соседей (англ. k-nearest neighbors algorithm, k-NN) — метрический алгоритм для автоматической классификации объектов или регрессии.

Минимакс-нормализация:

$$x' = (x - \min[X])/(\max[X] - \min[X])$$ (0,1)

Z-нормализация:

$$x' = (x - M[X])/\sigma[X]$$ $(-3\sigma, 3\sigma)$



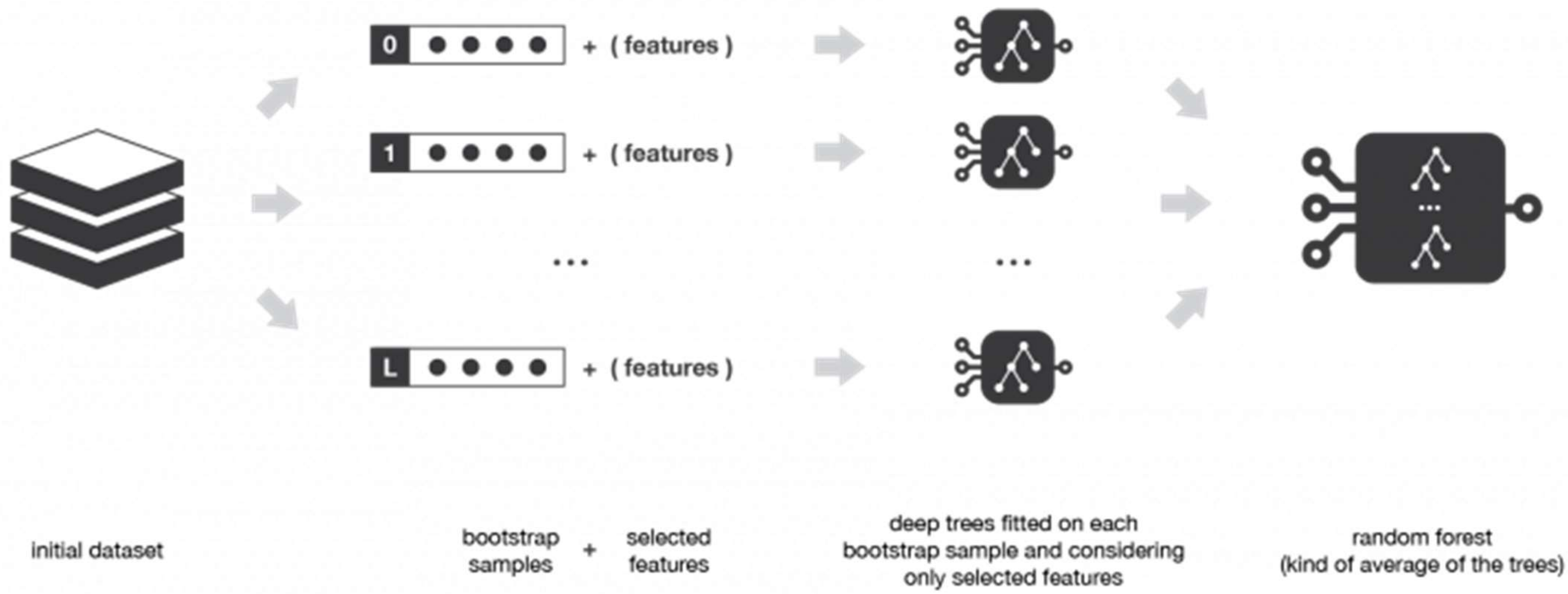
Алгоритмическая сложность для тестовой выборки:

$O(K \cdot N \cdot M)$

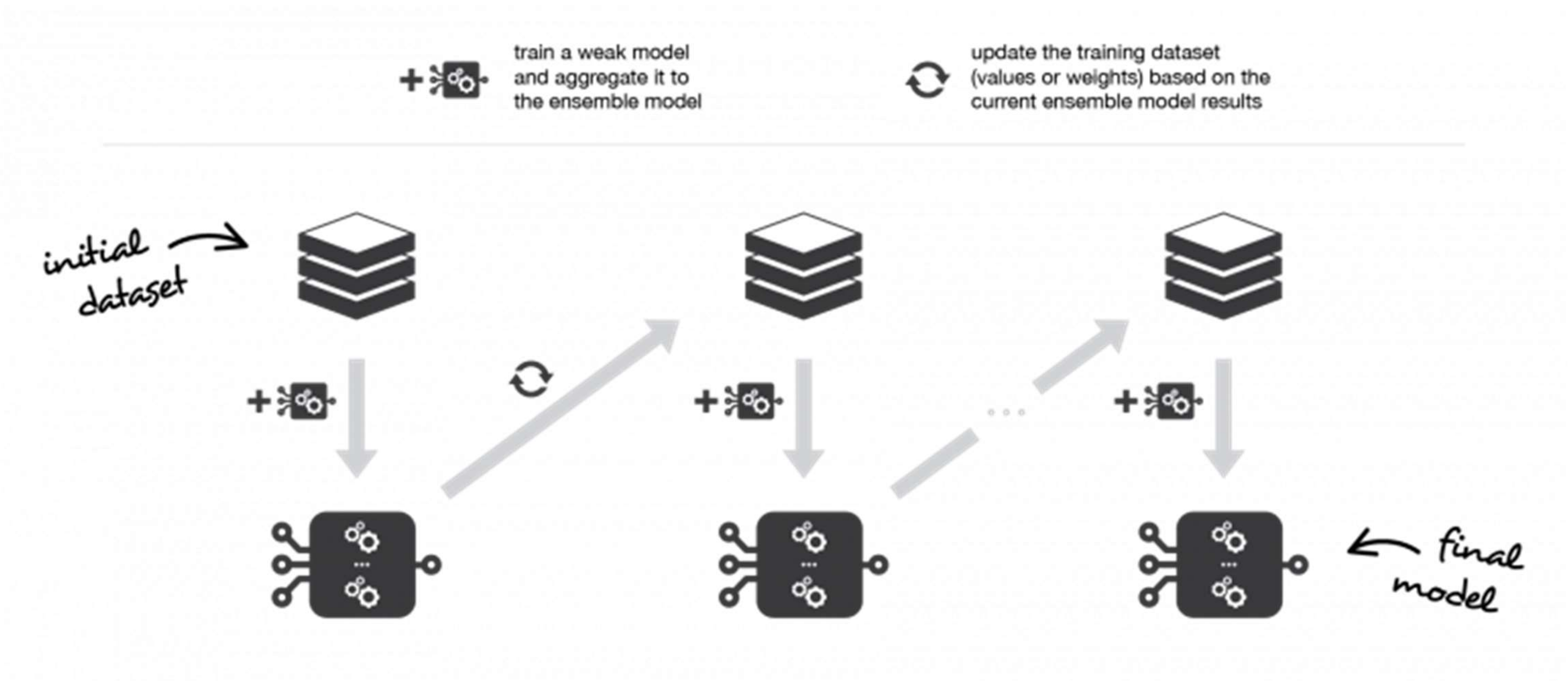
Обзор алгоритмов классификации: Случайный лес (бэгинг) и бустинговые алгоритмы

- Бэггинг
- Бустинг
- Стекинг

Обзор алгоритмов классификации:
Случайный лес (бэгинг) и бустинговые алгоритмы



Обзор алгоритмов классификации: Случайный лес (бэгинг) и бустинговые алгоритмы



Итоги урока

- Балансировка классов
- Схемы оценки обобщающей способности алгоритма
- Обзор алгоритмов классификации:

- Логистическая регрессия
- Метод опорных векторов
- k ближайших соседей
- Случайный лес и бустинговые алгоритмы

- Практическая часть

Практическая часть

Спасибо за внимание!