Введение в Машинное Обучение

Александр Сарачаков, Анастасия Кишкун,

Сколтех, Москва, Россия

Skoltech

Skolkovo Institute of Science and Technology

Содержание

Машинное Обучение (МО)

- большая подобласть Искусственного Интеллекта
- математическая дисциплина, направленная на создание численных методов, позволяющих решать задачи с накопленными данными на вычислительных машинах. Включает в себя глубокое обучение
- основано на данных $\rightarrow$ извлечение закономерностей из данных с применением математической статистики, численных методов, оптимизации, теории вероятностей, дискретного анализа, геометрии и т.д.
- результаты Компьютерных Наук используются для анализа обучающих алгоритмов, их сложности, теоретических гарантий
- **Пример**: предсказать метку изображения

Примеры задач МО

- Изображения: аннотация, сегментация, распознавание и верификация лиц, оптическое распознавание символов
- Автономные технические системы (роботы, машины)
- Медицинская диагностика, обнаружение мошенничества, обнаружение вторжений в сеть
- Обучение играм (шахматы, покер)
- Речь: распознавание, синтезирование, верификация
- Текст: моделирование, обнаружение спама

Математические задачи МО

- **Снижение размерности**: низко-размерные признаки, сохраняющие свойства данных
- **Регрессия**: предсказание вещественной выходной переменной по некоторому набору входных переменных (потребление топлива судном в зависимости от погодных условий, маршрута и т.д.)
- **Классификация**: размета объектов (например, классификация изображений)
- **Кластеризация**: разделение объектов на «однородные группы» (например, разбить документы на группы с похожими темами)
- **Ранжирование**: упорядочивание объектов в соответствии с некоторой метрикой

Что мы хотим?

- Алгоритмические задачи:
  - более эффективные и точные алгоритмы
  - применимость в задачах с большими данными (большие размерности, большие выборки)
  - обработка различных типов данных, включая неструктурированные данные, данные на графах и т.д.

- Теоретические задачи:
  - какие задачи можно решать? при каких условиях? ограничения?
  - теоретические гарантии?
  - качество работы обучающего алгоритма?

Данный курс: Цели

- **Модели и Алгоритмы**
  - **основные алгоритмы и их эффективность**
  - **современные темы**
- Теория
  - гарантии работы алгоритма
  - анализ алгоритма
- **Приложения**

Определения

- **Пример**: элемент данных, относящийся к одному объекту
- **Признаки**: входные значения (входные параметры, входной вектор, атрибуты, точка), описывающие объект
- **Метки**: выходные значения — категориальные (классификация) или вещественные (регрессия) — относящиеся к объекту
- **Данные**:
  - обучающие данные (обычно содержат метки)
  - тестовые данные (метки существуют, но неизвестны)
  - валидационные данные (содержат метки, используются для настройки гиперпараметров)

Определения

- пространство объектов (входное пространство) X
- пространство меток (выходное пространство) Y
- неизвестная целевая функция $f : X \to Y$
- **Дано**:
  - Обучающая выборка $S_m = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$
  - Объекты $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \in X$
  - Метки $\{y_1, \ldots, y_m\} \in Y$, $y_i = f(\mathbf{x}_i)$
- **Найти**:
  - $\widehat{f} : X \to Y$, аппроксимирующую f на X
- В основном МО — это о том, как
  - определить описание объекта $\mathbf{x}_i$
  - определить, что значит «$\widehat{f}$ аппроксимирует f»
  - построить $\widehat{f}$ используя S_m

Пример: Предсказание дохода ресторана

- **Объект**: место для открытия нового ресторана
- **Метка**: годовой доход
- **Признаки**: демографические свойства рассматриваемого района города, цены на аренду недвижимости в ближайшей окрестности, наличие офисов поблизости и т.д.
- **Трудности**:
  - маленькая обучающая выборка
  - много признаков ($d \gg 1$)
  - выбросы/неправильные измерения
  - неоднородные данные (большие мегаполисы и маленькие города)

Признаки

- Обычно для описания объекта $\mathbf{x}$ используется набор признаков, т.е. $\mathbf{x} = (x_1, \ldots, x_d)$
- Существует несколько типов признаков
  - бинарные, например $x_j \in \{0, 1\}$
  - номинальные, например $x_j \in \{\textit{red}, \textit{green}, \textit{blue}\}$
  - ординальные, например $x_j \in \{\textit{low}, \textit{middle}, \textit{high}\}$
  - вещественные, например $x_j \in \mathbb{R}$
- Таким образом, объекты $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \in X$ могут быть представлены в виде матрицы $m \times d$

$$\begin{pmatrix} x_{11} & \ldots & x_{1d} \\ \ldots & \ldots & \ldots \\ x_{m1} & \ldots & x_{md} \end{pmatrix}$$

Типы задач

- Классификация
  - двух-классовая классификация: $Y \in \{0, 1\}$
  - много-классовая классификация (моно-метки): $Y = \{1, \ldots, K\}$
  - мульти-классовая классификация (мульти-метки): $Y = \{0, 1\}^K$
- Регрессия: $Y = \mathbb{R}$ или $Y = \mathbb{R}^K$
- Ранжирование (обучение ранжированию): Y — конечное упорядоченное множество категорий

Сценарии обучения

- Постановки

батч: алгоритм получает полную выборку, обучает модель и делает предсказания для новых точек

онлайн: алгоритм получает по одной точке и делает предсказание для нее

- Обучение

пассивное: обучающая выборка с метками задана заранее и фиксирована

активное: алгоритм может запрашивать метки для произвольно выбранных точек

Батч постановка

- **Обучение без учителя**: данные без меток
- **Обучение с учителем**: обучение по размеченной выборке для предсказания меток для новых точек
- **Обучение с частичным привлечением учителя**: обучение по размеченной и неразмеченной выборкам для предсказания меток для новых точек
- **Трансдуктивное обучение**: использование размеченной и неразмеченной выборок для предсказания меток на неразмеченной части

Выборка данных «Ирисы» [Фишер, 1936]

- Определить морфологическую вариацию 3 видов цветов Ириса (Iris setosa, Iris virginica, Iris versicolor) по 4 признакам: длина и ширина лепестка и чашелистика в сантиметрах
- $d = 4$ признака, $K = 3$ класса (многоклассовый случай), $m = 150$

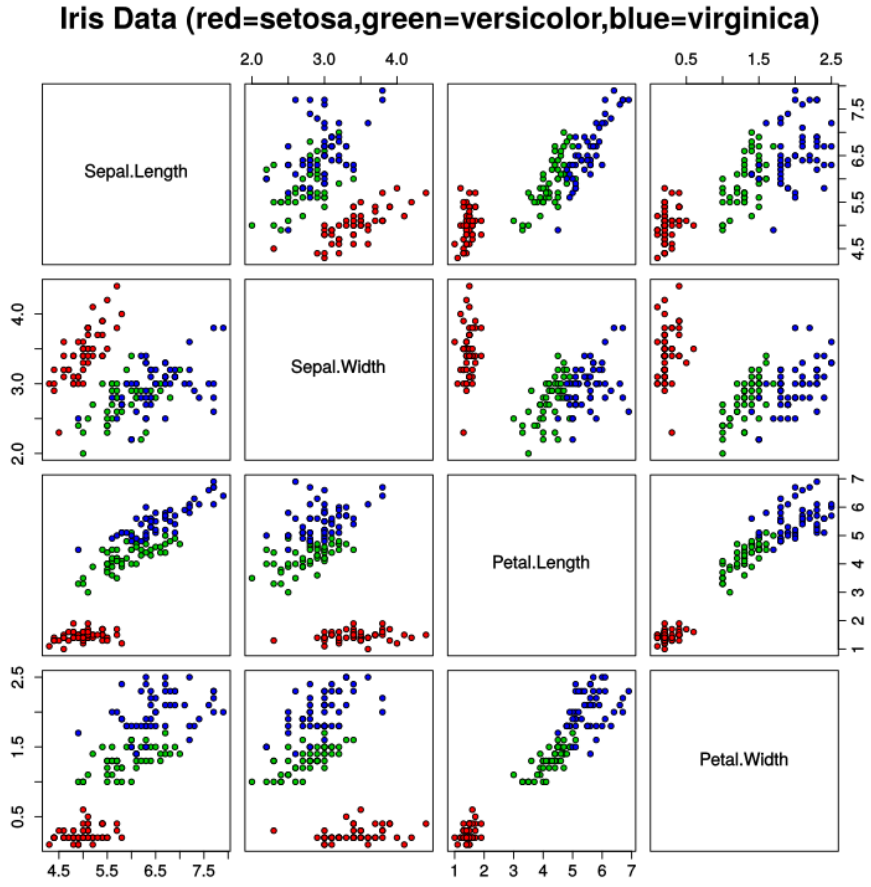


Figure – Диаграмма рассеяния обучающей выборки [Wikipedia]

Параметрическое множество гипотез

- **Множество гипотез** $F \subset Y^X$ — это подмножество функций, из которого обучающий алгоритм выбирает гипотезу
  - отображает априорное знание о конкретной задаче
  - зависит от заданных признаков
- Предсказательная модель принадлежит параметрическому семейству функций

$$F = \{f(\mathbf{x};\boldsymbol{\theta}) | \boldsymbol{\theta} \in \Theta\},$$

где
  - $f : X \times \Theta \to Y$ — фиксированное семейство функций,
  - Θ множество значений параметров
- Пример: в линейной модели $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$, $\Theta = \mathbb{R}^p$, а $\{\phi_j(\mathbf{x})\}_{j=1}^p$ — признаки:
  - регрессия $f(\mathbf{x};\boldsymbol{\theta}) = \sum_{j=1}^p \theta_j \phi_j(\mathbf{x})$, $Y = \mathbb{R}^1$
  - классификация $f(\mathbf{x};\boldsymbol{\theta}) = \operatorname{sign}\left(\sum_{j=1}^p \theta_j \phi_j(\mathbf{x})\right)$, $Y = \{-1, +1\}$

Параметрическое множество гипотез

- **Множество гипотез** $F \subset Y^X$ — это подмножество функций, из которого обучающий алгоритм выбирает гипотезу
  - отображает априорное знание о конкретной задаче
  - зависит от заданных признаков
- Предсказательная модель принадлежит параметрическому семейству функций

$$F = \{f(\mathbf{x}; \boldsymbol{\theta}) | \boldsymbol{\theta} \in \Theta\},$$

где
  - $f : X \times \Theta \to Y$ — фиксированное семейство функций,
  - Θ множество значений параметров
- Пример: в линейной модели $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$, $\Theta = \mathbb{R}^p$, а $\{\phi_j(\mathbf{x})\}_{j=1}^p$ — признаки:
  - регрессия $f(\mathbf{x}; \boldsymbol{\theta}) = \sum_{j=1}^p \theta_j \phi_j(\mathbf{x})$, $Y = \mathbb{R}^1$
  - классификация $f(\mathbf{x}; \boldsymbol{\theta}) = \operatorname{sign}\left(\sum_{j=1}^p \theta_j \phi_j(\mathbf{x})\right)$, $Y = \{-1, +1\}$

Параметрическое множество гипотез

- **Множество гипотез** $F \subset Y^X$ — это подмножество функций, из которого обучающий алгоритм выбирает гипотезу
  - отображает априорное знание о конкретной задаче
  - зависит от заданных признаков
- Предсказательная модель принадлежит параметрическому семейству функций

$$F = \{f(\mathbf{x}; \boldsymbol{\theta}) | \boldsymbol{\theta} \in \Theta\},$$

  где
  - $f : X \times \Theta \to Y$ — фиксированное семейство функций,
  - Θ множество значений параметров
- Пример: в линейной модели $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$, $\Theta = \mathbb{R}^p$, а $\{\phi_j(\mathbf{x})\}_{j=1}^p$ — признаки:
  - регрессия $f(\mathbf{x}; \boldsymbol{\theta}) = \sum_{j=1}^p \theta_j \phi_j(\mathbf{x})$, $Y = \mathbb{R}^1$
  - классификация $f(\mathbf{x}; \boldsymbol{\theta}) = \operatorname{sign}\left(\sum_{j=1}^p \theta_j \phi_j(\mathbf{x})\right)$, $Y = \{-1, +1\}$

Пример: Линейная регрессия

- $X = Y = \mathbb{R}^1$, $m = 200$, $d = 3$ признака: $\{x, x^2, 1\}$ или $\{x, \sin x, 1\}$

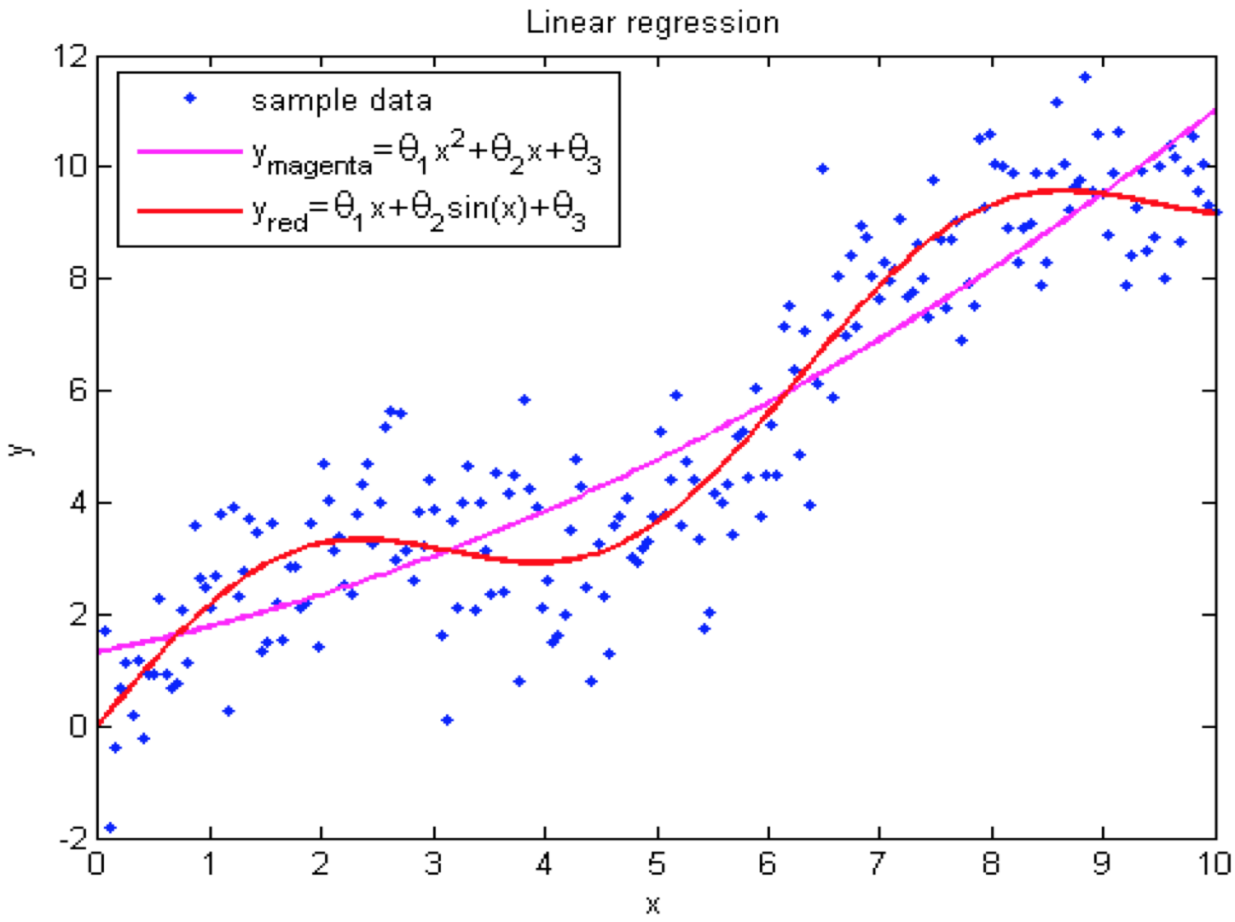


Figure – Линейная регрессия [Воронцов]

- **Обучающая выборка**: выборка S_m размера m независимо и одинаково распределенных (н.о.р.) точек в соответствии с распределением D на $X \times Y$

$$S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$$

- **Задача** построить гипотезу $\widehat{f} \in F$ с маленькой обобщающей ошибкой
  - детерминированный случай: $y = f(\mathbf{x})$ — детерминированная функция, только $\mathbf{x} \sim D$
  - стохастический случай: выход — это стохастическая функция от входа, например $y = f(\mathbf{x}) + \varepsilon$

- **Обучающая выборка**: выборка S_m размера m независимо и одинаково распределенных (н.о.р.) точек в соответствии с распределением D на $X \times Y$

$$S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$$

- **Задача** построить гипотезу $\widehat{f} \in F$ с маленькой обобщающей ошибкой
  - детерминированный случай: $y = f(\mathbf{x})$ — детерминированная функция, только $\mathbf{x} \sim D$
  - стохастический случай: выход — это стохастическая функция от входа, например $y = f(\mathbf{x}) + \varepsilon$

- **Обучающая выборка**: выборка S_m размера m независимо и одинаково распределенных (н.о.р.) точек в соответствии с распределением D на $X \times Y$

$$S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$$

- **Задача** построить гипотезу $\widehat{f} \in F$ с маленькой обобщающей ошибкой
  - детерминированный случай: $y = f(\mathbf{x})$ — детерминированная функция, только $\mathbf{x} \sim D$
  - стохастический случай: выход — это стохастическая функция от входа, например $y = f(\mathbf{x}) + \varepsilon$

- **Обучающая выборка**: выборка S_m размера m независимо и одинаково распределенных (н.о.р.) точек в соответствии с распределением D на $X \times Y$

$$S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$$

- **Задача** построить гипотезу $\widehat{f} \in F$ с маленькой обобщающей ошибкой
  - детерминированный случай: $y = f(\mathbf{x})$ — детерминированная функция, только $\mathbf{x} \sim D$
  - стохастический случай: выход — это стохастическая функция от входа, например $y = f(\mathbf{x}) + \varepsilon$

Схема обучения

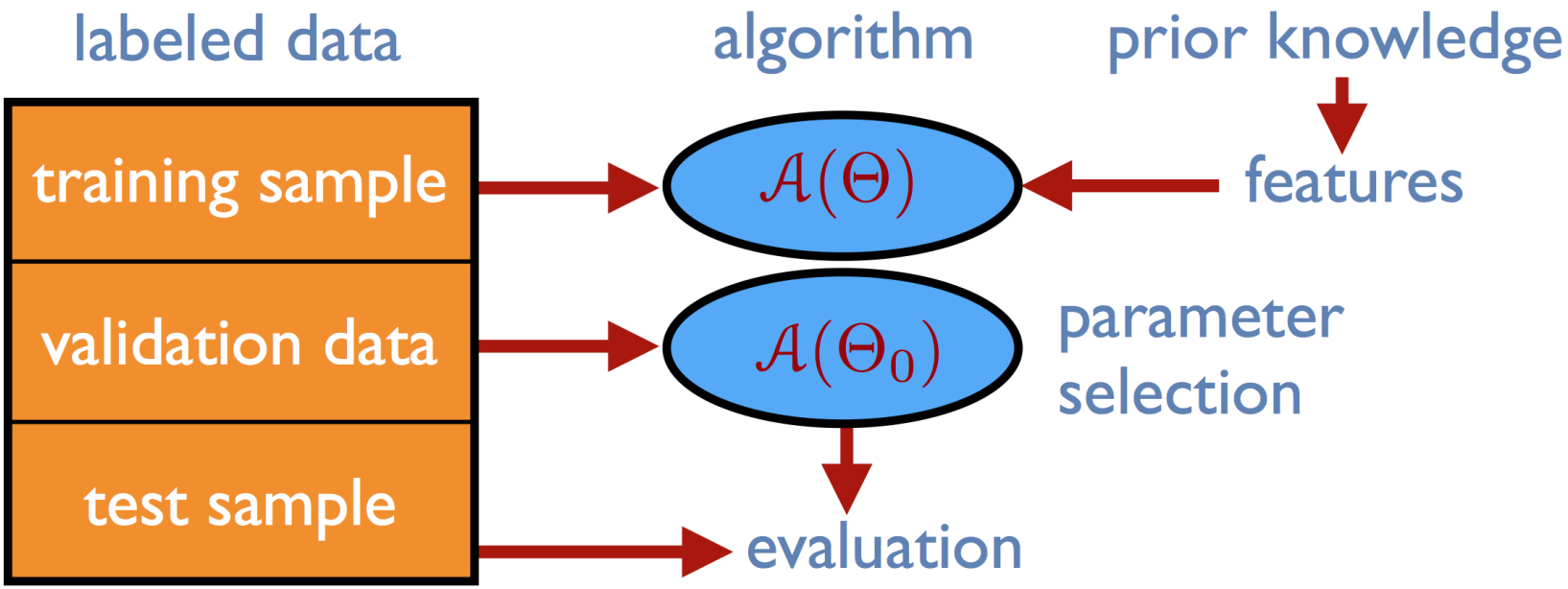


Figure – Схема обучения предсказательной модели [Mohri, 2016]

- Алгоритм $\mathcal{A} : (X \times Y)^m \to F$,
- Фаза обучения: применяя алгоритм $\mathcal{A}$ строим $\widehat{f} = \mathcal{A}(S_m)$,
- Фаза тестирования: вычислить предсказания $\widehat{f}(\mathbf{x}'_i)$ в новых точках $\mathbf{x}'_i$, $i = 1, \ldots, m'$ и сравнить с истинными значениями $y'_i = f(\mathbf{x}'_i)$.

Схема обучения

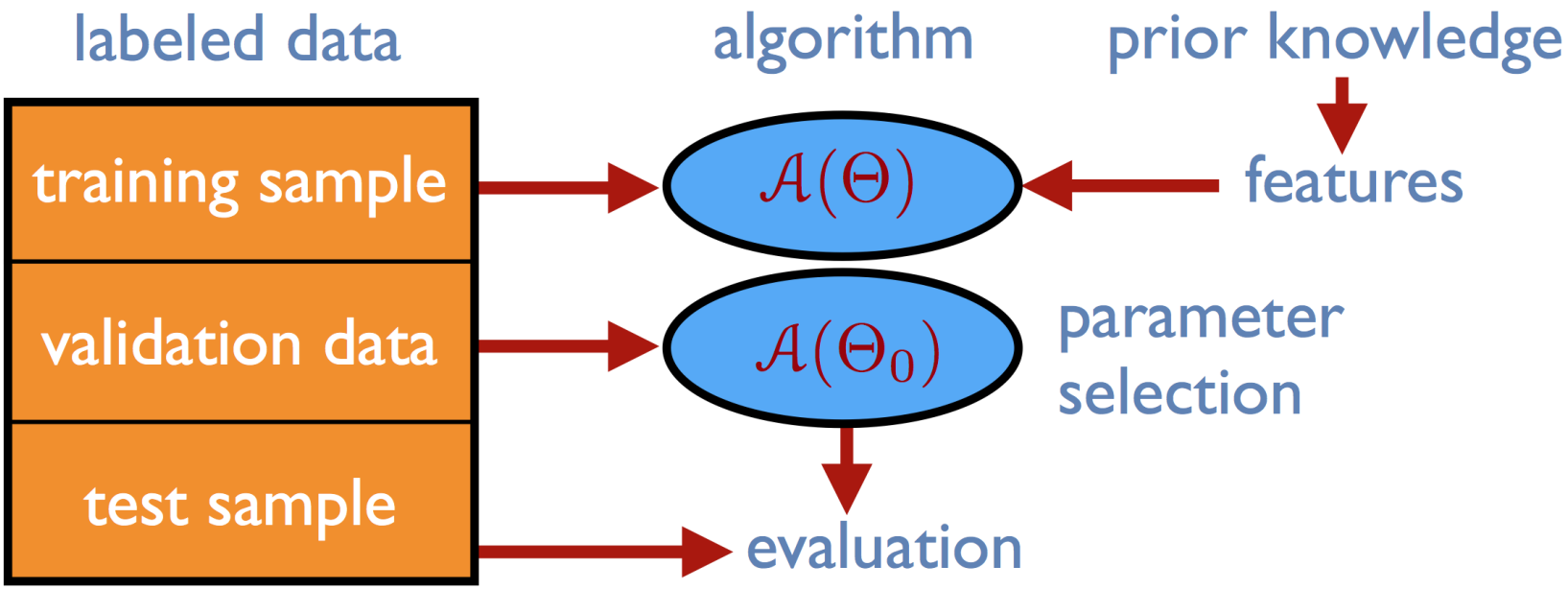


Figure – Схема обучения предсказательной модели [Mohri, 2016]

- Алгоритм $\mathcal{A} : (X \times Y)^m \to F$,
- Фаза обучения: применяя алгоритм $\mathcal{A}$ строим $\widehat{f} = \mathcal{A}(S_m)$,
- Фаза тестирования: вычислить предсказания $\widehat{f}(\mathbf{x}'_i)$ в новых точках $\mathbf{x}'_i$, $i = 1, \ldots, m'$ и сравнить с истинными значениями $y'_i = f(\mathbf{x}'_i)$.

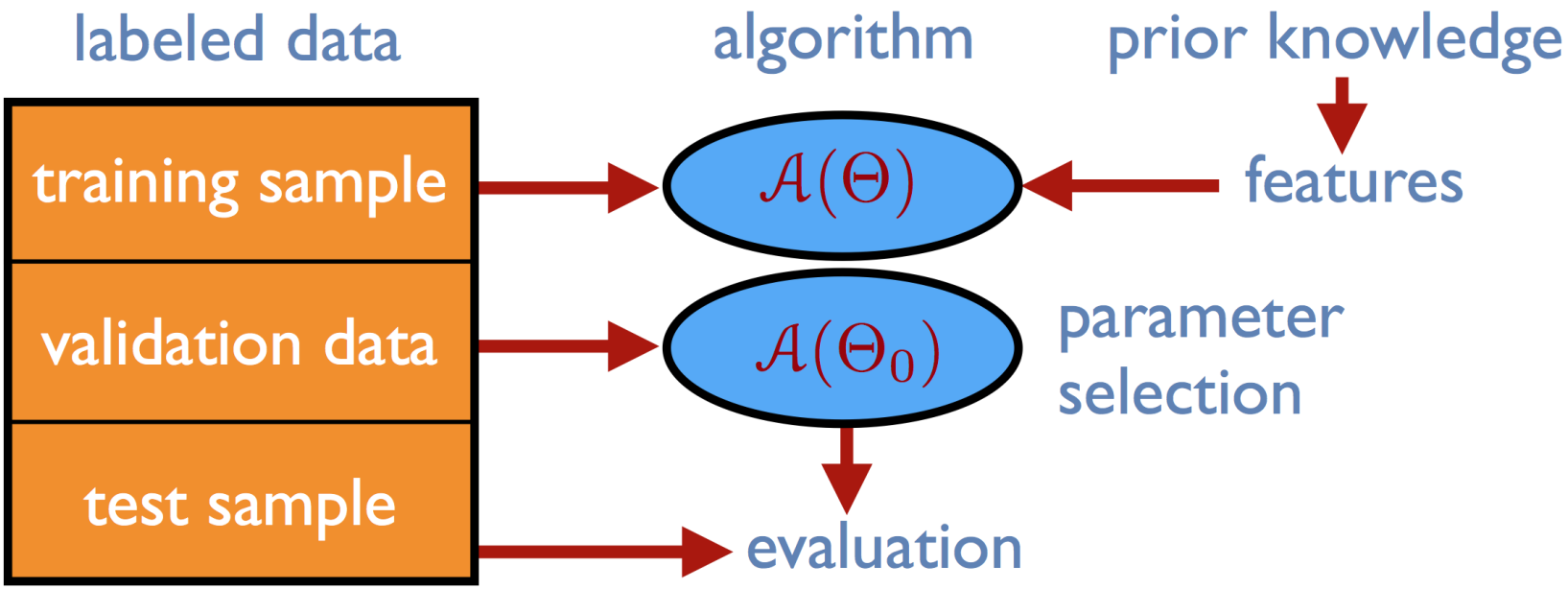


Figure – Схема обучения предсказательной модели [Mohri, 2016]

- Алгоритм $\mathcal{A} : (X \times Y)^m \to F$,
- Фаза обучения: применяя алгоритм $\mathcal{A}$ строим $\widehat{f} = \mathcal{A}(S_m)$,
- Фаза тестирования: вычислить предсказания $\widehat{f}(\mathbf{x}'_i)$ в новых точках $\mathbf{x}'_i$, $i = 1, \ldots, m'$ и сравнить с истинными значениями $y'_i = f(\mathbf{x}'_i)$.

Функция потерь (лосс) $L : Y \times Y \to \mathbb{R}$

- $L(y, \widehat{y})$ это ошибка предсказания $\widehat{y}$ при истинном значении y
- $0 - 1$ функция потерь $L(y, \widehat{y}) = 1_{y \neq \widehat{y}}$ в случае бинарной классификации
- $L(y, \widehat{y}) = (y - \widehat{y})^2$ в случае регрессии при $Y \subseteq \mathbb{R}$

Функция потерь (лосс) $L : Y \times Y \to \mathbb{R}$

- $L(y, \widehat{y})$ это ошибка предсказания $\widehat{y}$ при истинном значении y
- $0 - 1$ функция потерь $L(y, \widehat{y}) = 1_{y \neq \widehat{y}}$ в случае бинарной классификации
- $L(y, \widehat{y}) = (y - \widehat{y})^2$ в случае регрессии при $Y \subseteq \mathbb{R}$

Функция потерь (лосс) $L : Y \times Y \to \mathbb{R}$

- $L(y, \widehat{y})$ это ошибка предсказания $\widehat{y}$ при истинном значении y
- $0-1$ функция потерь $L(y, \widehat{y}) = 1_{y \neq \widehat{y}}$ в случае бинарной классификации
- $L(y, \widehat{y}) = (y - \widehat{y})^2$ в случае регрессии при $Y \subseteq \mathbb{R}$

- **Эмпирическая ошибка** для $f \in F$ и выборки S_m

$$\widehat{R}(f) = \frac{1}{m} \sum_{i=1}^{m} L(f(\mathbf{x}_i), y_i)$$

- **Обобщающая ошибка**: для $f \in F$

$$R(f) = \mathbb{E}_{(\mathbf{x},y)\sim D}[L(f(\mathbf{x}), y)]$$

- **Байесовская ошибка**

$$R^* = \inf_f R(f)$$

- **Эмпирическая ошибка** для $f \in F$ и выборки S_m

$$\widehat{R}(f) = \frac{1}{m} \sum_{i=1}^{m} L(f(\mathbf{x}_i), y_i)$$

- **Обобщающая ошибка**: для $f \in F$

$$R(f) = \mathbb{E}_{(\mathbf{x},y)\sim D}[L(f(\mathbf{x}), y)]$$

- **Байесовская ошибка**

$$R^* = \inf_f R(f)$$

- **Эмпирическая ошибка** для $f \in F$ и выборки S_m

$$\widehat{R}(f) = \frac{1}{m} \sum_{i=1}^{m} L(f(\mathbf{x}_i), y_i)$$

- **Обобщающая ошибка**: для $f \in F$

$$R(f) = \mathbb{E}_{(\mathbf{x},y)\sim D}[L(f(\mathbf{x}), y)]$$

- **Байесовская ошибка**

$$R^* = \inf_f R(f)$$

Минимизация Эмпирического Риска

- Пусть F — множество гипотез
- Построить гипотезу $f \in F$, минимизирующую эмпирическую ошибку

$$\widehat{f} = \arg\min_{f \in F} \widehat{R}(f)$$

- F может оказаться слишком сложным
- Размер выборки может быть не достаточно большим

Минимизация Эмпирического Риска

- Пусть F — множество гипотез
- Построить гипотезу $f \in F$, минимизирующую эмпирическую ошибку

$$\widehat{f} = \arg\min_{f \in F} \widehat{R}(f)$$

- F может оказаться слишком сложным
- Размер выборки может быть не достаточно большим

Минимизация Эмпирического Риска

- Пусть F — множество гипотез
- Построить гипотезу $f \in F$, минимизирующую эмпирическую ошибку

$$\widehat{f} = \arg\min_{f \in F} \widehat{R}(f)$$

- F может оказаться слишком сложным
- Размер выборки может быть не достаточно большим

Минимизация Эмпирического Риска

- Пусть F — множество гипотез
- Построить гипотезу $f \in F$, минимизирующую эмпирическую ошибку

$$\widehat{f} = \arg\min_{f \in F} \widehat{R}(f)$$

- F может оказаться слишком сложным
- Размер выборки может быть не достаточно большим

Обучение $\neq$ интерполяция данных

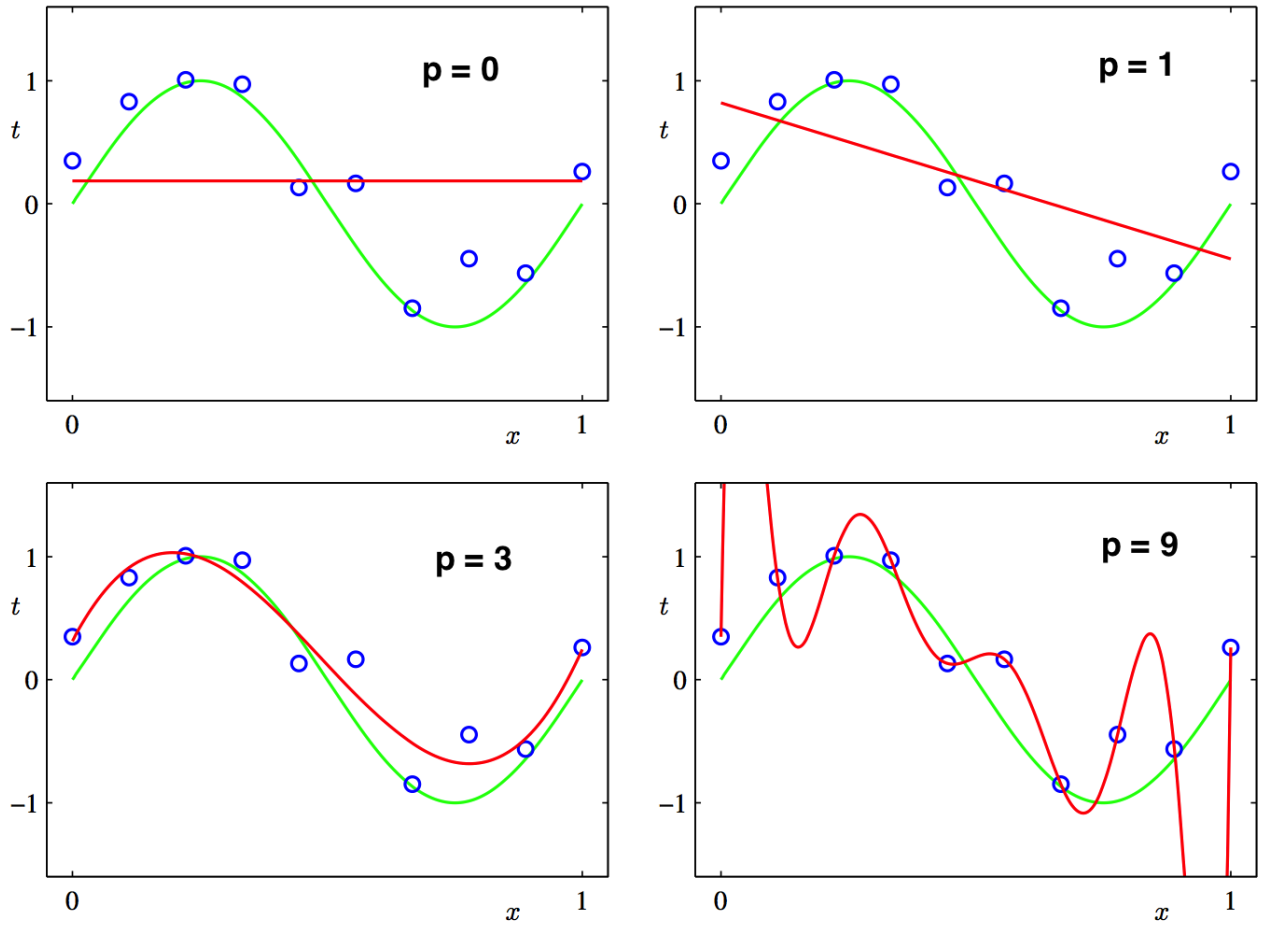


Figure – Понятие сложности модели. Как определить сложность? [Bishop]

- Множество многочленов $f(x;\boldsymbol{\theta}) = \sum_{j=0}^{p} \theta_j x^j$
- Квадратичная функция потерь $L(\widehat{y}, y) = (\widehat{y} - y)^2$

Обучение $\neq$ интерполяция данных

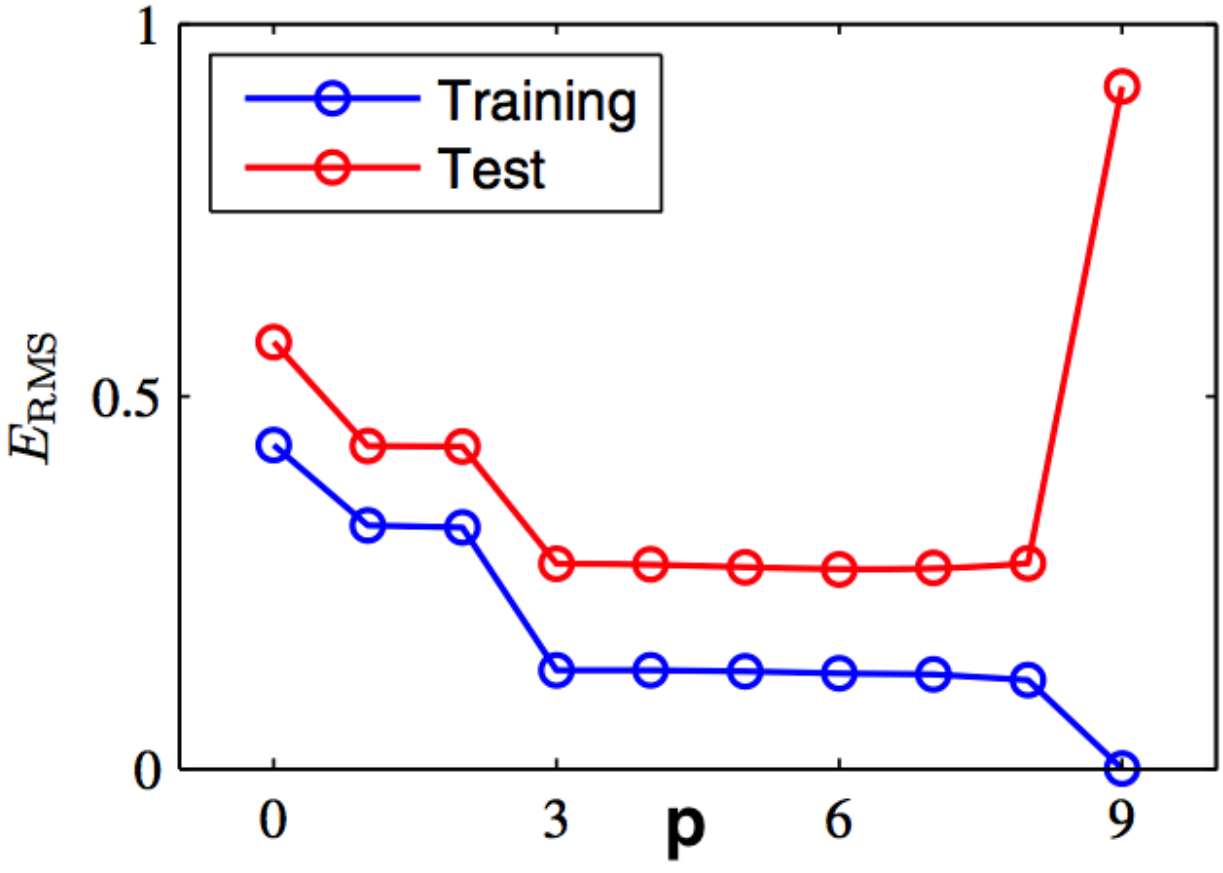


Figure – Понятие сложности модели. Как определить сложность? [Bishop]

Переобучение: ошибка на тестовой выборке $\gg$ ошибка на обучающей выборке

Эмпирические наблюдения

- наилучшая гипотеза на данной выборке может быть не наилучшей в целом
- обобщение — это не запоминание
- сложные правила могут давать плохие предсказания
- компромисс: сложность или размер выборки (недообучение/переобучение)
- как обнаружить: использовать кросс-валидацию на случайном разбиении выборки на обучающую/тестовую подвыборки

Эмпирические наблюдения

- наилучшая гипотеза на данной выборке может быть не наилучшей в целом
- обобщение — это не запоминание
- сложные правила могут давать плохие предсказания
- компромисс: сложность или размер выборки (недообучение/переобучение)
- как обнаружить: использовать кросс-валидацию на случайном разбиении выборки на обучающую/тестовую подвыборки

Эмпирические наблюдения

- наилучшая гипотеза на данной выборке может быть не наилучшей в целом
- обобщение — это не запоминание
- сложные правила могут давать плохие предсказания
- компромисс: сложность или размер выборки (недообучение/переобучение)
- как обнаружить: использовать кросс-валидацию на случайном разбиении выборки на обучающую/тестовую подвыборки

Эмпирические наблюдения

- наилучшая гипотеза на данной выборке может быть не наилучшей в целом
- обобщение — это не запоминание
- сложные правила могут давать плохие предсказания
- компромисс: сложность или размер выборки (недообучение/переобучение)
- как обнаружить: использовать кросс-валидацию на случайном разбиении выборки на обучающую/тестовую подвыборки

Эмпирические наблюдения

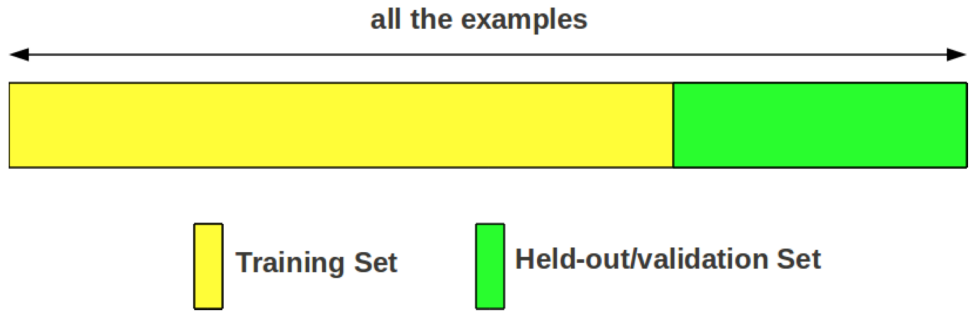
- наилучшая гипотеза на данной выборке может быть не наилучшей в целом
- обобщение — это не запоминание
- сложные правила могут давать плохие предсказания
- компромисс: сложность или размер выборки (недообучение/переобучение)
- как обнаружить: использовать кросс-валидацию на случайном разбиении выборки на обучающую/тестовую подвыборки

Валидационная выборка

- Отложить часть обучающей выборки (например, $10\% - 20\%$)
- Эта подвыборка называется валидационной выборкой
- Важно: валидационные данные это НЕ тестовые данные
- Обучаем модель, используя оставшуюся часть выборки
- Оцениваем ошибку на валидационной выборке
- Выбираем модель с наименьшей ошибкой на валидационной выборке
- Проблемы:
  - Уменьшается размер обучающей выборки, поэтому обычно используется при достаточно больших выборках
  - Валидационная выборка может быть не очень хорошей при неудачном разбиении (используйте случайное разбиение!)

Валидационная выборка

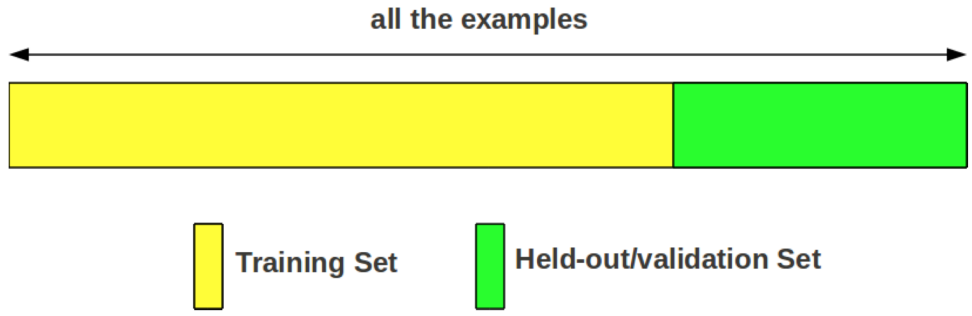
- Отложить часть обучающей выборки (например, $10\% - 20\%$)
- Эта подвыборка называется валидационной выборкой



- Важно: валидационные данные это НЕ тестовые данные
- Обучаем модель, используя оставшуюся часть выборки
- Оцениваем ошибку на валидационной выборке
- Выбираем модель с наименьшей ошибкой на валидационной выборке
- Проблемы:
  - Уменьшается размер обучающей выборки, поэтому обычно используется при достаточно больших выборках
  - Валидационная выборка может быть не очень хорошей при неудачном разбиении (используйте случайное разбиение!)

Валидационная выборка

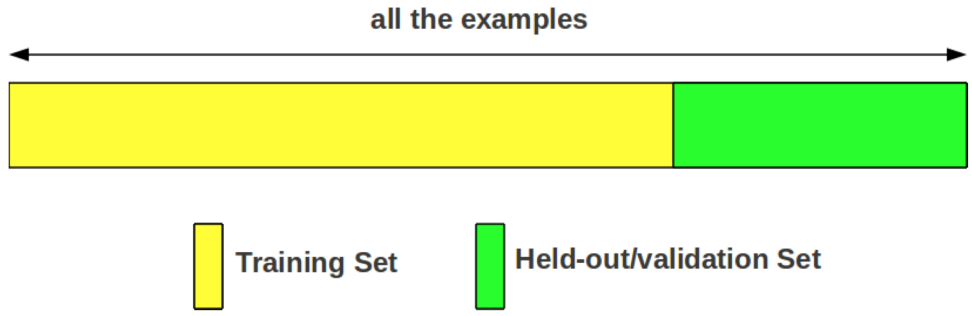
- Отложить часть обучающей выборки (например, $10\% - 20\%$)
- Эта подвыборка называется валидационной выборкой



- Важно: валидационные данные это НЕ тестовые данные
- Обучаем модель, используя оставшуюся часть выборки
- Оцениваем ошибку на валидационной выборке
- Выбираем модель с наименьшей ошибкой на валидационной выборке
- Проблемы:
  - Уменьшается размер обучающей выборки, поэтому обычно используется при достаточно больших выборках
  - Валидационная выборка может быть не очень хорошей при неудачном разбиении (используйте случайное разбиение!)

Валидационная выборка

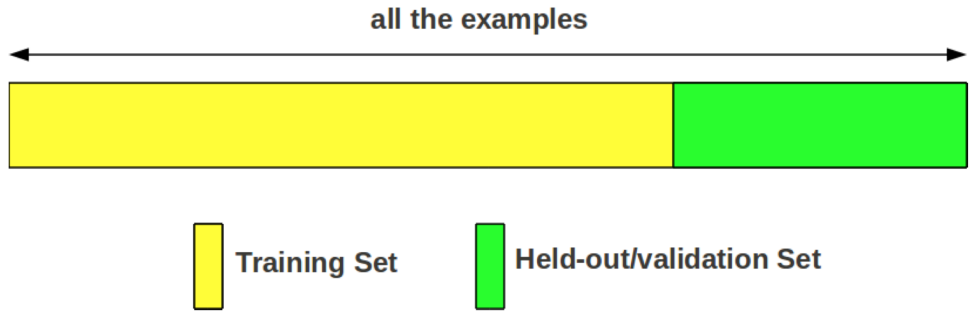
- Отложить часть обучающей выборки (например, $10\% - 20\%$)
- Эта подвыборка называется валидационной выборкой



- Важно: валидационные данные это НЕ тестовые данные
- Обучаем модель, используя оставшуюся часть выборки
- Оцениваем ошибку на валидационной выборке
- Выбираем модель с наименьшей ошибкой на валидационной выборке
- Проблемы:
  - Уменьшается размер обучающей выборки, поэтому обычно используется при достаточно больших выборках
  - Валидационная выборка может быть не очень хорошей при неудачном разбиении (используйте случайное разбиение!)

Валидационная выборка

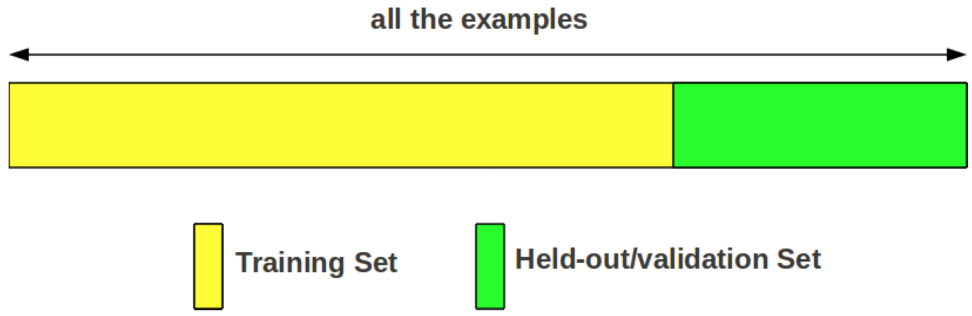
- Отложить часть обучающей выборки (например, $10\% - 20\%$)
- Эта подвыборка называется валидационной выборкой



- Важно: валидационные данные это НЕ тестовые данные
- Обучаем модель, используя оставшуюся часть выборки
- Оцениваем ошибку на валидационной выборке
- Выбираем модель с наименьшей ошибкой на валидационной выборке
- Проблемы:
  - Уменьшается размер обучающей выборки, поэтому обычно используется при достаточно больших выборках
  - Валидационная выборка может быть не очень хорошей при неудачном разбиении (используйте случайное разбиение!)

Валидационная выборка

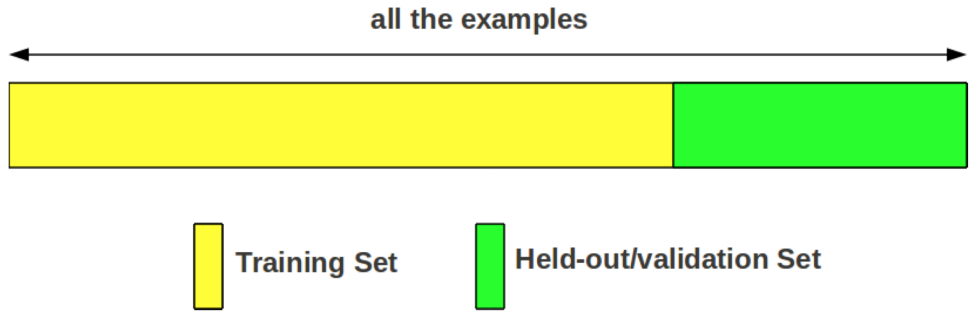
- Отложить часть обучающей выборки (например, $10\% - 20\%$)
- Эта подвыборка называется валидационной выборкой



- Важно: валидационные данные это НЕ тестовые данные
- Обучаем модель, используя оставшуюся часть выборки
- Оцениваем ошибку на валидационной выборке
- Выбираем модель с наименьшей ошибкой на валидационной выборке
- Проблемы:
  - Уменьшается размер обучающей выборки, поэтому обычно используется при достаточно больших выборках
  - Валидационная выборка может быть не очень хорошей при неудачном разбиении (используйте случайное разбиение!)

Валидационная выборка

- Отложить часть обучающей выборки (например, $10\% - 20\%$)
- Эта подвыборка называется валидационной выборкой



- Важно: валидационные данные это НЕ тестовые данные
- Обучаем модель, используя оставшуюся часть выборки
- Оцениваем ошибку на валидационной выборке
- Выбираем модель с наименьшей ошибкой на валидационной выборке
- Проблемы:
  - Уменьшается размер обучающей выборки, поэтому обычно используется при достаточно больших выборках
  - Валидационная выборка может быть не очень хорошей при неудачном разбиении (используйте случайное разбиение!)

Кросс-валидация I

M-кратная кросс-валидация

- Разбить выборку на M подмножеств одинакового размера
- Каждое подмножество содержит m/M примеров
- Обучить модель на $M-1$ подмножествах, провалидировать на оставшемся
- Повторить процедуру M раз, каждый раз выбирая новое валидационное подмножество
- Выбрать модель с наименьшей средней валидационной ошибкой
- Обычно M выбирают равным $4-10$

Кросс-валидация I

M-кратная кросс-валидация

- Разбить выборку на M подмножеств одинакового размера
- Каждое подмножество содержит m/M примеров
- Обучить модель на $M-1$ подмножествах, провалидировать на оставшемся
- Повторить процедуру M раз, каждый раз выбирая новое валидационное подмножество

- Выбрать модель с наименьшей средней валидационной ошибкой
- Обычно M выбирают равным $4-10$

Кросс-валидация I

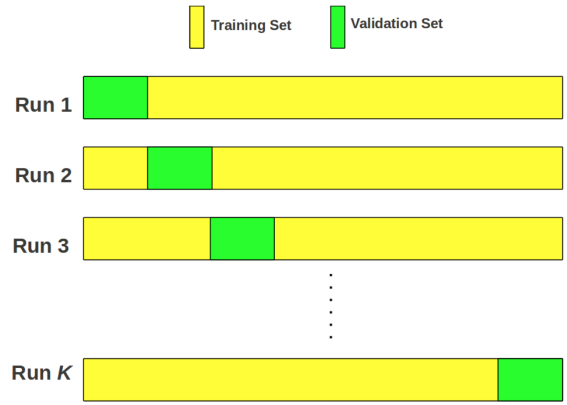
M-кратная кросс-валидация

- Разбить выборку на M подмножеств одинакового размера
- Каждое подмножество содержит m/M примеров
- Обучить модель на $M-1$ подмножествах, провалидировать на оставшемся
- Повторить процедуру M раз, каждый раз выбирая новое валидационное подмножество
- Выбрать модель с наименьшей средней валидационной ошибкой
- Обычно M выбирают равным $4-10$

Кросс-валидация I

M-кратная кросс-валидация

- Разбить выборку на M подмножеств одинакового размера
- Каждое подмножество содержит m/M примеров
- Обучить модель на $M-1$ подмножествах, провалидировать на оставшемся
- Повторить процедуру M раз, каждый раз выбирая новое валидационное подмножество

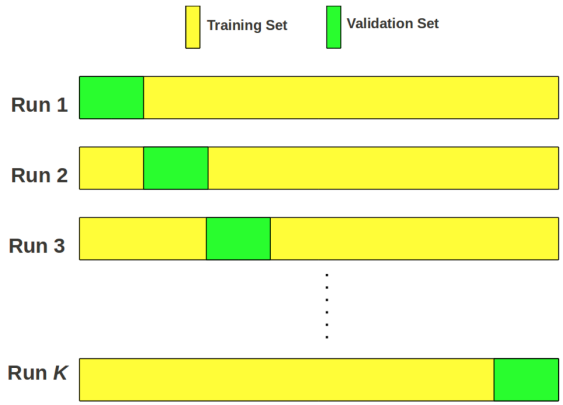


- Выбрать модель с наименьшей средней валидационной ошибкой
- Обычно M выбирают равным $4-10$

Кросс-валидация I

M-кратная кросс-валидация

- Разбить выборку на M подмножеств одинакового размера
- Каждое подмножество содержит m/M примеров
- Обучить модель на $M-1$ подмножествах, провалидировать на оставшемся
- Повторить процедуру M раз, каждый раз выбирая новое валидационное подмножество



- Выбрать модель с наименьшей средней валидационной ошибкой
- Обычно M выбирают равным $4-10$

M-кратная кросс-валидация

- Разбить выборку на M подмножеств одинакового размера
- Каждое подмножество содержит m/M примеров
- Обучить модель на $M-1$ подмножествах, провалидировать на оставшемся
- Повторить процедуру M раз, каждый раз выбирая новое валидационное подмножество

- Выбрать модель с наименьшей средней валидационной ошибкой
- Обычно M выбирают равным $4-10$

Кросс-валидация II

$N \times M$-кратная кросс-валидация

- Разбить обучающую выборку N раз на M подмножеств одинакового размера
- Агрегируем результаты всех N M-кратных кросс-валидаций (например, усредняем)
- Увеличивая N можно увеличить точность
- Каждый объект используется N раз для валидации
- Используя результаты N повторений можно строить доверительные интервалы

Кросс-валидация II

$N \times M$-кратная кросс-валидация

- Разбить обучающую выборку N раз на M подмножеств одинакового размера
- Агрегируем результаты всех N M-кратных кросс-валидаций (например, усредняем)
- Увеличивая N можно увеличить точность
- Каждый объект используется N раз для валидации
- Используя результаты N повторений можно строить доверительные интервалы

Кросс-валидация II

$N \times M$-кратная кросс-валидация

- Разбить обучающую выборку N раз на M подмножеств одинакового размера
- Агрегируем результаты всех N M-кратных кросс-валидаций (например, усредняем)
- Увеличивая N можно увеличить точность
- Каждый объект используется N раз для валидации
- Используя результаты N повторений можно строить доверительные интервалы

Кросс-валидация II

$N \times M$-кратная кросс-валидация

- Разбить обучающую выборку N раз на M подмножеств одинакового размера
- Агрегируем результаты всех N M-кратных кросс-валидаций (например, усредняем)
- Увеличивая N можно увеличить точность
- Каждый объект используется N раз для валидации
- Используя результаты N повторений можно строить доверительные интервалы

Кросс-валидация II

$N \times M$-кратная кросс-валидация

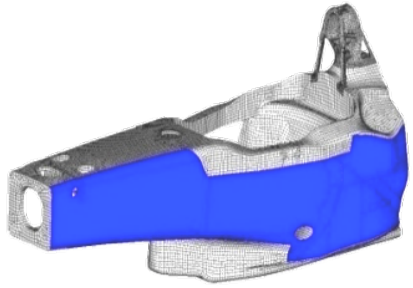
- Разбить обучающую выборку N раз на M подмножеств одинакового размера
- Агрегируем результаты всех N M-кратных кросс-валидаций (например, усредняем)
- Увеличивая N можно увеличить точность
- Каждый объект используется N раз для валидации
- Используя результаты N повторений можно строить доверительные интервалы

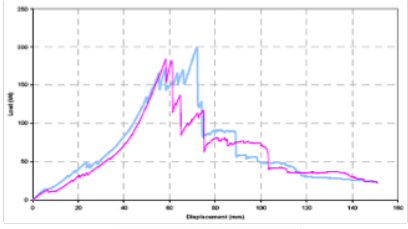
Послесловие

- Основные шаги для решения прикладной задачи
  - изучить постановку задачи и данные
  - построение признаков
  - выбор модели и формулировка задачи как задачи оптимизации
  - оценка качества модели
  - внедрение и использование модели
- При разработке алгоритмов МО
  - используйте искусственные данные для выявления ограничений алгоритма
  - используйте реальные данные (из доступных источников данных) для тестирования модели в реальных условиях

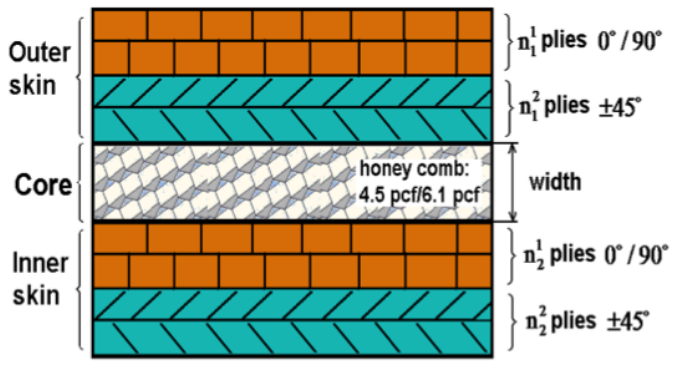
Проектирование боковой панели автомобиля Формулы 1

- **Цель**: минимизировать массу боковой панели автомобиля Формулы 1
  - по числу слоев, структуре слоев, толщине
  - при ограничении на прочность
- **Трудности**: Метод конечных элементов имеет ограниченную точность: эксперименты вычислительно-затратны
- **Решение**: построить регрессионную модель используя результаты симуляций и экспериментов (консолидация данных, data fusion) и использовать ее при решении задачи оптимизации;
- **Постановка задачи МО**:
  - **Цель**: боковая панель
  - **Признаки**: количество слоев, типы слоев, толщина
  - **Метка**: прочность панели (регрессия)
- **Результат**: уменьшение массы на 10% с использованием меньшего количества симуляций и натурных экспериментов



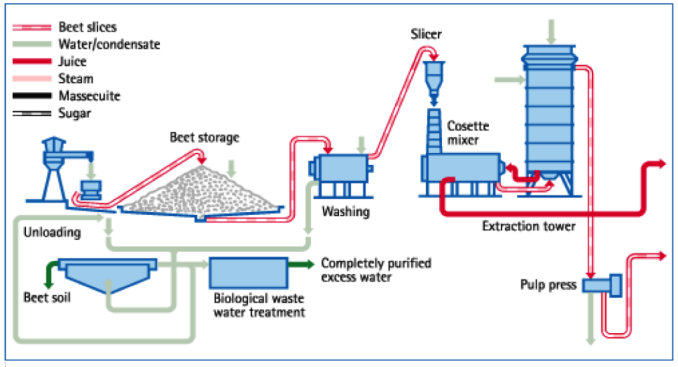


Load and Energy Plots



Управление и рекомендации на основе предсказательной модели

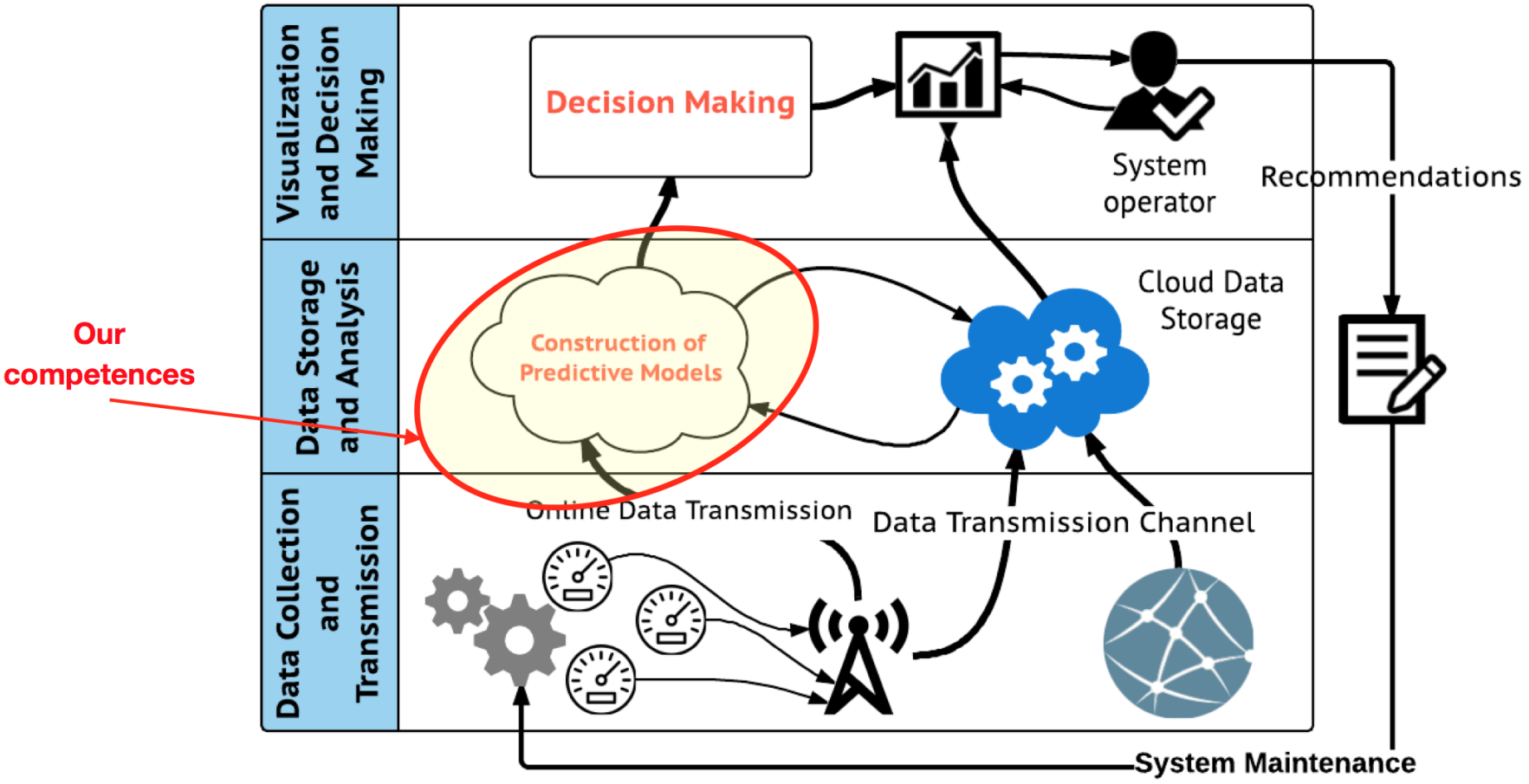
- **Задача**: Оптимизировать извлечение сахара
  - **Объект**: одна партия продукта
  - **Вход** $\mathbf{x}$: форма кусочков свеклы, температура, содержание сахара, температура омывающей воды, pH, температура внутри диффузора и т.д.
  - **Выход** y: затраты, потери, производительность извлечения сахара
- **Трудности**:
  - Неоднородные данные и шум,
  - Большие объемы высоко-размерных потоковых данных,
  - Пропуски в данных, выбросы и т.д.



Управление и рекомендации на основе предсказательной модели

- **Задача**: минимизировать расход топливо грузовым судном, детектировать утечки (кражу) топлива, оптимизировать маршрут
  - **Объект**: исторические данные расхода топлива судном при заданных погодных условиях на участке маршрута
  - **Признаки** $\mathbf{x}$: размеры (длина, высота, ширина), грузоподъемность, тип (паром, баржа и т.д.), количество двигателей и т.д.; маршрут; погодные условия, морские течения (исторические, текущие, прогнозируемые); управление (скорость судна и т.д.)
  - **Метка** y: расход топлива
- **Трудности**:
  - Неоднородные данные и шум,
  - Большие объемы высоко-размерных потоковых данных,
  - Пропуски в данных, выбросы и т.д.

Место в Индустриальном Процессе



Кредитный скоринг

- **Объект**: заявка на получение кредита от определенного кандидата
- **Метка**: $0/1$ (плохой кандидат/хороший кандидат)
- **Признаки**: пол, адрес, профессия, должность, образование, опыт работы, возраст, заработная плата и т.д.
- **Трудности**: оценить вероятность дефолта

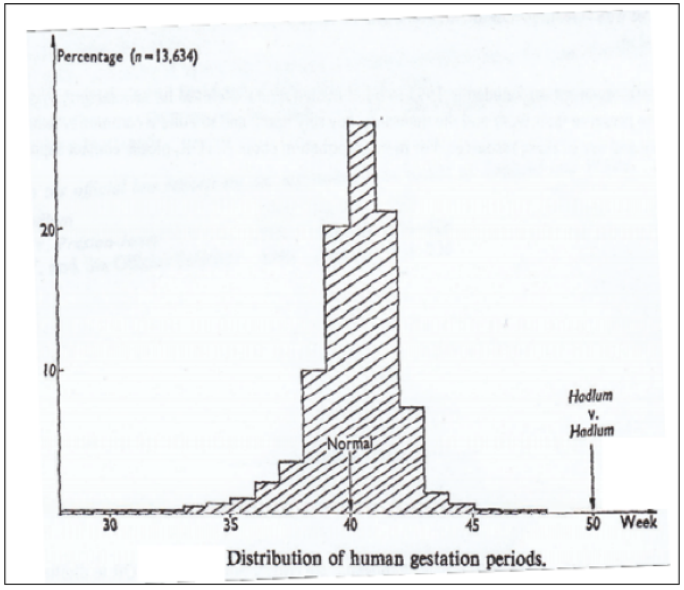
Предсказание оттока клиентов

- **Объект**: клиент в определенный момент времени
- **Метка**: $0/1$ (откажется от пользования услугами/не откажется)
- **Признаки**: пол, адрес, тип подписки, услуги, которыми пользовался, частота и продолжительность звонков и т.д.
- **Трудности**:
  - оценивание вероятности оттока
  - огромные выборки
  - трудоемкая генерация признаков

Предсказание переходов по контекстной рекламе

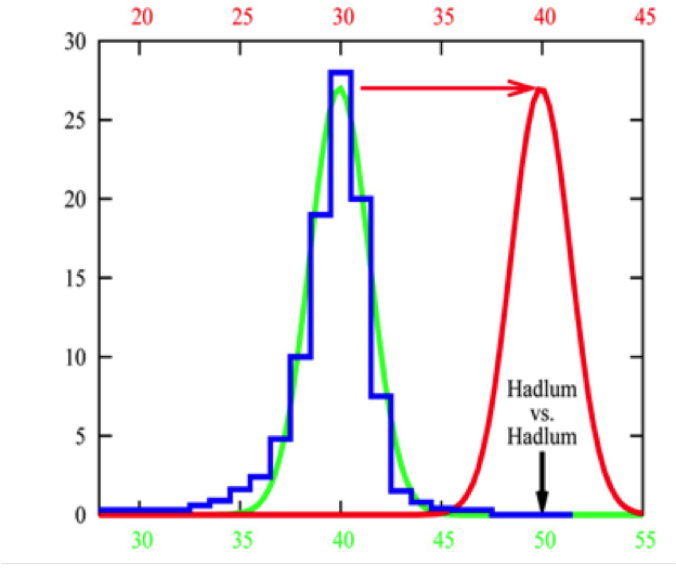
- **Объект**: тройка $\langle$*пользователь, реклама, баннер*$\rangle$
- **Метка**: переход по контекстной рекламе, показанной по запросу пользователя на сайте
- **Необработанные данные**: действия пользователя на сайте, профиль (браузер, устройство и т.д.), история действий пользователя, переходы других пользователей и т.д.
- **Трудности**:
  - огромные выборки (миллиарды показов рекламы)
  - выбросы/неправильные измерения
  - неоднородные данные (большие мегаполисы и маленькие города)
  - трудоемкая генерация признаков
  - основной критерий качества — прибыль от показа рекламы

Детектирование аномалий. Пример: Хэдлум против Хэдлум (1949) [Барнетт, 1978]



- У миссис Хэдлум родился ребенок через 349 дней после того, как мистер Хэдлум пошел служить в армию
- Средняя продолжительность беременности 280 дней (40 недель)
- Статистически, 349 дней это выброс

Хэдлум против Хэдлум (1949) [Барнетт, 1978]



- У миссис Хэдлум родился ребенок через 349 дней после того, как мистер Хэдлум пошел служить в армию
- Средняя продолжительность беременности 280 дней (40 недель)
- Статистически, 349 дней это выброс

Определение понятия аномалии [Хокинз, 1980]

«Выброс — это наблюдение, которое отличается от других наблюдений настолько сильно, что вызывает подозрение, будто оно было порождено другим механизмом»

Приложения:

- Раннее обнаружение мошеннических транзакций
- Обнаружение сбоев и их локализация в технических системах (например, самолетах, автомобилях)
- Кибер-безопасность, обнаружение атак, использующих недокументированные уязвимости

Принцип Минимизации Структурного Риска

- Рассмотрим последовательность множеств гипотез, упорядоченных по включению

$$F_1 \subset F_2 \subset \ldots \subset F_n \subset \ldots$$

$$f = \arg \min_{f \in F_n,\, n \in \mathbb{N}} \widehat{R}(f) + \text{penalty}(F_n, m)$$

- Строгие теоретические гарантии

- Решение как правило имеет высокую вычислительную сложность

Семейство алгоритмов

- Минимизация эмпирического риска

$$f = \arg\min_{f \in F} \widehat{R}(f)$$

- Минимизация структурного риска для $F_n \subseteq F_{n+1}$

$$f = \arg\min_{f \in F_n,\, n \in \mathbb{N}} \widehat{R}(f) + \text{penalty}(F_n, m)$$

- Алгоритмы, основанные на регуляризации

$$f = \arg\min_{f \in F} \widehat{R}(h) + \lambda \|f\|^2,\ \lambda > 0$$