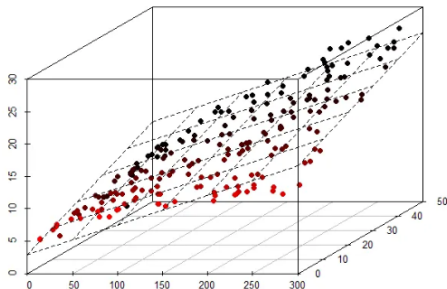


# Линейная регрессия и обобщения

Виктор Китов



# Содержание

- 1 Линейная регрессия
- 2 Регуляризация
- 3 Разные функции потерь
- 4 Взвешенный учет наблюдений
- 5 Другие типы регрессии

# Линейная регрессия

- Линейная регрессия

$$\hat{y} = x^T \hat{\beta} = \sum_{i=1}^D \hat{\beta}_i x^i$$

$$\hat{\beta} = \arg \min_{\beta} \sum_{n=1}^N \left( x_n^T \beta - y_n \right)^2$$

- Если смещение  $\hat{\beta}_0$  явно не указано, всегда включают константный признак в  $x$ .
- Предположения:
  - каждый  $x^i$  линейно влияет  $y$  с коэффициентом  $\hat{\beta}_i$
  - вклад каждого признака  $x^i$  не зависит от значений др. признаков.

## Анализ метода

### Преимущества:

- интерпретируемость
  - знак коэффициентов=направление влияния  $x^i$
  - модуль коэффициента=сила влияния  $x^i$  (при признаках из одной шкалы!)
  - $\hat{\beta}$  асимптотически нормальны (см. [ссылку](#)), можем тестировать:
    - значимость отличия коэффициентов (или группы коэффициентов) от нуля,
    - гипотезу положительного влияния признака на отклик (положительности коэффициента)
  - есть аналитическое решение
  - быстро и просто строятся прогнозы
  - меньше переобучается, чем сложные модели
    - для больших  $D$  может быть оптимальной моделью

### Недостатки: модельные предположения слишком простые

- признаки могут влиять нелинейно
- признаки могут иметь взаимозависимое влияние

# Признаки

- Можно использовать вещественные признаки и бинарные.
- Категориальные можно закодировать:
  - номером категории (плохо)
  - счетчиком встречаемости категории
  - в виде бинарных (one-hot encoding)
  - в виде вещественных (mean value encoding)

# One-hot кодирование

Row Number	Direction		Row Number	Direction_N	Direction_S	Direction_W	Direction_E	Direction_NW
1	North	➡	1	1	0	0	0	0
2	North-West		2	0	0	0	0	1
3	South		3	0	1	0	0	0
4	East		4	0	0	0	1	0
5	North-West		5	0	0	0	0	1
	North-West			0	0	0	0	1
	East			0	0	0	1	0
	South			0	1	0	0	0

## Mean value кодирование

- можно делать по вещественному признаку
- какие могут быть проблемы?

id	job	job_mean	target
1	Doctor	0,50	1
2	Doctor	0,50	0
3	Doctor	0,50	1
4	Doctor	0,50	0
5	Teacher	1	1
6	Teacher	1	1
7	Engineer	0,50	0
8	Engineer	0,50	1
9	Waiter	1	1
10	Driver	0	0

## Mean value кодирование

- можно делать по вещественному признаку
- если делаем по y, то на отдельной выборке!

id	job	job_mean	target
1	Doctor	0,50	1
2	Doctor	0,50	0
3	Doctor	0,50	1
4	Doctor	0,50	0
5	Teacher	1	1
6	Teacher	1	1
7	Engineer	0,50	0
8	Engineer	0,50	1
9	Waiter	1	1
10	Driver	0	0



## Решение

Определим  $X \in \mathbb{R}^{N \times D}$ ,  $\{X\}_{ij}$  - значение  $j$ -го признака  $i$ -го объекта,  $Y \in \mathbb{R}^N$ ,  $\{Y\}_i$  - отклик  $i$ -го объекта.

Метод наименьших квадратов (МНК, ordinary least squares):

$$L(\beta) = \sum_{n=1}^N \left( x_n^T \beta - y_n \right)^2 = \|X\beta - Y\|_2^2 \rightarrow \min_{\beta}$$

$$L'(\beta) = 2 \sum_{n=1}^N x_n \left( x_n^T \beta - y_n \right) = 0$$

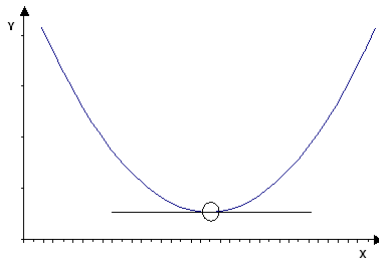
$$2X^T(X\beta - Y) = 0$$

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

Интуиция:  $\hat{\beta}_i$  пропорциональна ковариации  $x_n^i$  и  $y_n$ , нормализованная  $Var[x^i]$  и  $cov[x^i, x^j]$ .

## Глобальность минимума

- Это глобальный минимум, т.к. оптимизируемый критерий выпуклый.
  - выпуклая ф-ция от линейной выпукла<sup>1</sup>, сумма выпуклых - выпукла
  - для выпуклой ф-ции достаточное условие минимума - равенство нулю производной.



---

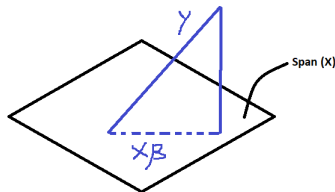
<sup>1</sup>Будет ли суперпозиция произвольных выпуклых ф-ций выпуклой?

## Геометрическая интерпретация

- Находится линейная комбинация признаков, чтобы приблизить  $Y$  в  $\mathbb{R}^N$ :

$$L(\beta) = \sum_{n=1}^N \left( x_n^T \beta - y_n \right)^2 = \|X\beta - Y\|_2^2 \rightarrow \min_{\beta}$$

- Решение - проекция на линейную оболочку признаков в  $\mathbb{R}^N$ .



## Линейно зависимые признаки - проблема

- Решение  $\hat{\beta} = (X^T X)^{-1} X^T Y$  существует, когда  $X^T X$  невырождена.
- Поскольку  $\text{rank}(X) = \text{rank}(X^T X) \forall X$ , проблема возникает при линейной зависимости признаков.
  - пример: константный признак и one-hot закодированные  $e_1, e_2, \dots, e_K$ , поскольку  $\sum_k e_k \equiv 1$
  - интерпретация: возникает неоднозначность  $\hat{\beta}$  для зависимых признаков:
    - линейная зависимость:  $\exists \alpha : x^T \alpha = 0 \forall x$
    - предположим  $\hat{\beta}$  - решение  $\sum_{n=1}^N (x_n^T \beta - y_n)^2 \rightarrow \min_{\beta}$
    - тогда  $\hat{\beta} + k\alpha$  - тоже решение  
 $\forall k \in \mathbb{R} : x^T \hat{\beta} \equiv x^T \hat{\beta} + kx^T \alpha \equiv x^T (\hat{\beta} + k\alpha).$
- При почти зависимых признаках ( $X^T X$  плохо обусловлена, т.е.  $\lambda_{\max}/\lambda_{\min}$  велико):
  - $\hat{\beta}$  неустойчиво и принимает большие по модулю значения.

## Линейно зависимые признаки - решение

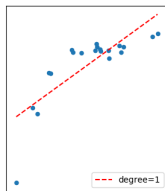
- Проблема может быть решена:
  - отбором признаков (feature selection)
  - снижением размерности (dimensionality reduction)
  - накладыванием доп. условий на решение (регуляризация)
    - $\|\beta\|$  должна быть мала
    - некоторые  $\beta_i$  должны быть неотрицательные
    - ...

## Нелинейные зависимости в линейной регрессии

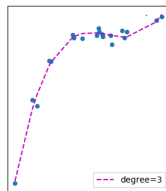
Перейдем от  $x \in \mathbb{R}^D$  к его нелинейному преобразованию  $\in \mathbb{R}^M$ :

$$x \rightarrow [\phi_1(x), \phi_2(x), \dots, \phi_M(x)]$$

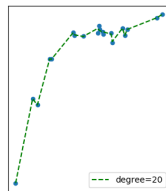
$$\hat{y}(x) = \phi(x)^T \hat{\beta} = \sum_{m=1}^M \hat{\beta}_m \phi_m(x)$$



Underfit  
High Bias



Correct Fit  
Low Bias



Overfit  
Low Bias

Линейная регрессия с полиномиальным преобразованием признака

## Анализ

$\hat{y}(x)$  уже нелинейно зависит от  $x$ . При этом преимущества лин. регрессии сохраняются:

- интерпретируемость (для несложных преобразований)
- аналитическое решение
- глобальный минимум потерь

# Популярные трансформации признаков

## Популярные преобразования признаков.

$\phi_k(x)$	примеры
$(x^i)^2, \sqrt{x^i}, \ln x^i$	учитываем нелинейное влияние расстояния до метро на стоимость квартиры
$\mathbb{I}\{x^i \in [a, b]\}$	принадлежит ли клиент определенному возрасту? (совершеннолетний, но не пенсионер)
$x^i \mathbb{I}[x^i \leq a], x^i \mathbb{I}[x^i > a]$	учесть изменения влияния $x^i$ при $x^i > a$
$(x^i)(x^j)$	длина x ширина участка = площадь
$x^i/x^j$	стоимость квартиры/метраж = стоимость одного метра
$F_{x^i}(x^i)$	приводим признак к равномерному распределению ( $F(\cdot)$ - ф-ция распределения)



## Популярные трансформации признаков

Использование метрических признаков (метод перестаёт быть линейным, нужна численная оптимизация).

$\phi_k(x)$	примеры
$\langle x, z \rangle / (\ x\  \ z\ )$	угол между объектом и репрезентативным объектом $z$
$\ x - z\ ^2$	расстояние от объекта до репрезентативного объекта $z$ (чаще используют близость)

## Нелинейная регрессия

- Можно исходные признаки подставлять в нелинейную ф-цию  $\hat{y} = f(x|\beta)$

$$L(\beta|X, Y) = \sum_{n=1}^N (f(x_n|\beta) - y_n)^2$$

$$\hat{\beta} = \arg \min_{\beta} L(\beta|X, Y)$$

- В общем случае не существует аналитического решения  $\hat{\beta}$ .
  - используем численные методы, например SGD.

# Содержание

- 1 Линейная регрессия
- 2 Регуляризация**
- 3 Разные функции потерь
- 4 Взвешенный учет наблюдений
- 5 Другие типы регрессии

# Регуляризация

- Для лучшей обобщающей способности важна не только точность, но и простота модели.
- Учтем простоту дополнительным регуляризатором  $R(\beta)$ :

$$\sum_{n=1}^N \left( x_n^T \beta - y_n \right)^2 + \lambda R(\beta) \rightarrow \min_{\beta}$$

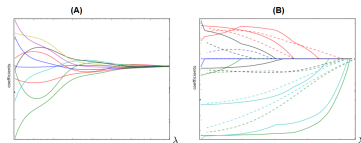
- $\lambda > 0$  - гиперпараметр, контролирующий сложность модели.
  - как он влияет на сложность?

$R(\beta) = \|\beta\|_1$ , Лассо регрессия (Lasso regression)

$R(\beta) = \|\beta\|_2^2$  Гребневая регрессия (Ridge regression)

## Зависимость $\hat{\beta}$ от $\lambda$

- Зависимость  $\hat{\beta}$  от  $\lambda$  для гребневой (A) и лассо (B) регрессии:



- Лассо регрессия может использоваться для автоматического отбора признаков.
- $\lambda$  находят по экспоненциальной сетке  $[10^{-6}, 10^{-5}, \dots, 10^5, 10^6]$ .
  - потом уточняют
- Всегда рекомендуется включать регуляризацию:
  - плавный контроль сложности модели
  - решение однозначно даже для линейно зависимых признаков
    - из набора решений выбирается с наименьшим  $\|\beta\|$ .

# ElasticNet

- ElasticNet - линейная комбинация  $L_1$  и  $L_2$  регуляризации:

$$R(\beta) = \alpha \|\beta\|_1 + (1 - \alpha) \|\beta\|_2^2$$

$\alpha \in [0, 1]$  — гиперпараметр.

- Если два признака  $x^i$  и  $x^j$  равны:
  - Гребневая регрессия выберет оба с равным весом
    - правильно, т.к. нет априорных предпочтений
  - Лассо регрессия выберет один из них (в общем случае)
    - зато отберет лишние признаки
- ElasticNet обладает обоими преимуществами.

# Аналитическое решение для гребневой регрессии

Критерий гребневой регрессии

$$\sum_{n=1}^N \left( x_n^T \beta - y_n \right)^2 + \lambda \beta^T \beta \rightarrow \min_{\beta}$$

Условие стационарности (равенство нулю производной):

$$2 \sum_{n=1}^N x_n \left( x_n^T \hat{\beta} - y_n \right) + 2\lambda \hat{\beta} = 0$$

$$2X^T(X\hat{\beta} - Y) + 2\lambda\hat{\beta} = 0$$

$$(X^T X + \lambda I) \hat{\beta} = X^T Y$$

поэтому

$$\hat{\beta} = (X^T X + \lambda I)^{-1} X^T Y$$

$X^T X + \lambda I$  всегда невырождена как сумма  $X^T X \succeq 0$  и  $\lambda I \succ 0$ .

## Учет разных признаков с разной силой

- Прогнозы обычной регрессии инвариантны к масштабированию признаков:

$$\hat{y} = \hat{\beta}_1 x^1 + \hat{\beta}_2 x^2 + \dots \xrightarrow{x^1 \rightarrow x^1/\alpha} \left( \alpha \hat{\beta}_1 \right) \left( \frac{x^1}{\alpha} \right) + \hat{\beta}_2 x^2 + \dots$$

- Но не регуляризованной:

$$\sum_{n=1}^N \left( x_n^T \beta - y_n \right)^2 + \lambda R(\beta) \rightarrow \min_{\beta}$$

- После изменения масштаба признаков, они будут вносить другой вклад в прогноз.
  - для большего учета признака как нужно изменить его масштаб?



## Агрегация разных моделей

- Пусть  $x_i = [x_i^1, \dots, x_i^D]$  состоит из прогнозов у  $D$  разными моделями, которые мы линейно объединяем.
- Веса  $\frac{1}{D}, \frac{1}{D}, \dots, \frac{1}{D}$  - разумный бейзлайн, но модели могут быть разной точности.
- Учтем их с настраиваемыми весами  $\hat{\beta}_1, \dots, \hat{\beta}_D$  (blending, linear stacking):

$$\hat{y} = x_n^T \hat{\beta}$$

- Логично предположить неотрицательность весов и несильное отклонение от бейзлайна.

$$\begin{cases} \sum_{n=1}^N (x_n^T \beta - y_n)^2 + \lambda \sum_{d=1}^D (\beta_d - \frac{1}{D})^2 \rightarrow \min_{\beta} \\ \beta_d \geq 0, \quad d = 1, 2, \dots, D \end{cases}$$

- Во избежание переобучения нужно базовые модели и  $\hat{\beta}$  настраивать на разных обучающих выборках.

# Содержание

- 1 Линейная регрессия
- 2 Регуляризация
- 3 Разные функции потерь**
- 4 Взвешенный учет наблюдений
- 5 Другие типы регрессии

## Обобщение функции потерь<sup>2</sup>

- Обобщим квадратичные потери на произвольные:

$$\sum_{n=1}^N \left( x^T \beta - y_n \right)^2 \rightarrow \min_{\beta} \quad \Rightarrow \quad \sum_{n=1}^N \mathcal{L}(x_n^T \beta - y_n) \rightarrow \min_{\beta}$$

### ФУНКЦИЯ ПОТЕРЬ

$$\mathcal{L}(\varepsilon) = \varepsilon^2$$

$$\mathcal{L}(\varepsilon) = |\varepsilon|$$

$$\mathcal{L}(\varepsilon) = \begin{cases} \frac{1}{2}\varepsilon^2, & |\varepsilon| \leq \delta \\ \delta \left( |\varepsilon| - \frac{1}{2}\delta \right) & |\varepsilon| > \delta \end{cases}$$

### НАЗВАНИЕ

квадратичная

абсолютная

Хубера

### СВОЙСТВА

дифференцируемая

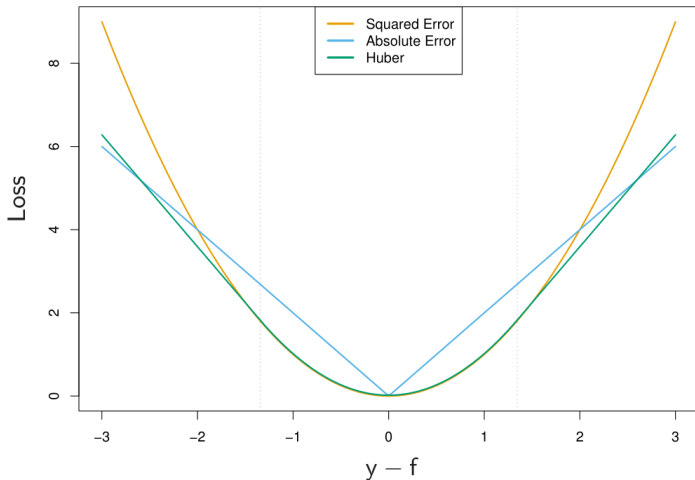
устойчивая

к выбросам

оба свойства

<sup>2</sup>Чему равен константный прогноз, минимизирующий квадратичные и абсолютные ошибки?

# Визуализация функций потерь



## Оптимальный прогноз для квадратичной ошибки

Константный прогноз  $\hat{y} \in \mathbb{R}$  при квадратичной ф-ции потерь:

$$L(\hat{y}) = \mathbb{E} \left\{ (\hat{y} - y)^2 \right\} \rightarrow \min_{\hat{y} \in \mathbb{R}}$$

$$\frac{\partial L(\hat{y})}{\partial \hat{y}} = \mathbb{E} \{ 2(\hat{y} - y) \} = 2\hat{y} - 2\mathbb{E}y = 0$$

$$\hat{y} = \mathbb{E}y$$

## Оптимальный прогноз для абсолютной ошибки

Константный прогноз  $\hat{y} \in \mathbb{R}$  при абсолютной ф-ции потерь:

$$\begin{aligned} L(\hat{y}) &= \mathbb{E} \{ |\hat{y} - y| \} = \int |\hat{y} - y| p(y) dy = \\ &= \int (\hat{y} - y) \mathbb{I}[\hat{y} \geq y] p(y) dy + \int (y - \hat{y}) \mathbb{I}[\hat{y} < y] p(y) dy \rightarrow \min_{\hat{y} \in \mathbb{R}} \\ \frac{\partial L(\hat{y})}{\partial \hat{y}} &= \int \mathbb{I}[\hat{y} \geq y] p(y) dy - \int \mathbb{I}[\hat{y} < y] p(y) dy = 0 \\ \frac{\partial L(\hat{y})}{\partial \hat{y}} &= \int_{y \leq \hat{y}} p(y) dx - \int_{y > \hat{y}} p(y) dy = 0 \\ \hat{y} &= \text{median}[y] \end{aligned}$$

## Влияние функции потерь на результат

- Следовательно, для фиксированного  $x$  оптимальный функциональный прогноз будет:

$$\arg \min_{\hat{y}(x)} \mathbb{E} \left\{ (\hat{y}(x) - y)^2 \mid x \right\} = \mathbb{E}[y|x]$$

$$\arg \min_{\hat{y}(x)} \mathbb{E} \{ |\hat{y}(x) - y| \mid x \} = \text{median}[y|x]$$

- При фиксированных обучающей выборке и модели результат будет получаться разный для различных ф-ций потерь!

## Содержание

- 1 Линейная регрессия
- 2 Регуляризация
- 3 Разные функции потерь
- 4 Взвешенный учет наблюдений**
- 5 Другие типы регрессии



## Взвешенный учет наблюдений<sup>3</sup>

- Взвешенный учет наблюдений

$$\sum_{n=1}^N w_n (x_n^T \beta - y_n)^2 \rightarrow \min_{\beta \in \mathbb{R}^D}$$

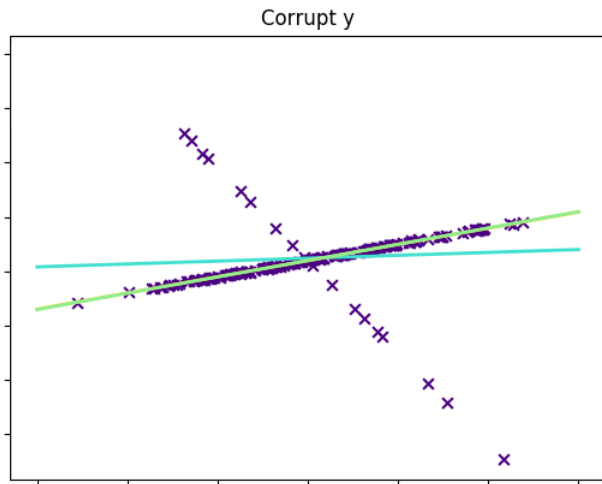
$$w_1 \geq 0, \dots, w_N \geq 0$$

- Неравномерные веса могут быть обусловлены:
  - разному доверию различным фрагментам обучающей выборки
  - желанием снизить влияние объектов-выбросов
  - желанием сделать сбалансированную выборку
    - Например, результаты голосования. Женщины много голосовали, мужчины мало. Хотим построить модель без перекоса на женские предпочтения.

---

<sup>3</sup>Выведите решение для взвешенной линейной регрессии.

## Проблема выбросов



## Робастная регрессия

- Инициализировать  $w_1 = \dots = w_N = 1/N$
- Повторять до сходимости:
  - оценить регрессию  $\hat{y}(x)$  используя  $(x_i, y_i)$  с весами  $w_i$ .
  - для каждого  $i = 1, 2, \dots, N$ :
    - переоценить  $\varepsilon_i = \hat{y}(x_i) - y_i$
    - пересчитать веса  $w_i = K(|\varepsilon_i|)$
  - нормализовать веса  $w_i = \frac{w_i}{\sum_{n=1}^N w_n}$

### Комментарии:

- $K(\cdot)$  - некоторая убывающая функция.
- Веса объектов-выбросов убывают, получаем устойчивое к выбросам решение.
- Алгоритм обобщается на любой метод, допускающий взвешенный учет наблюдений.

# Содержание

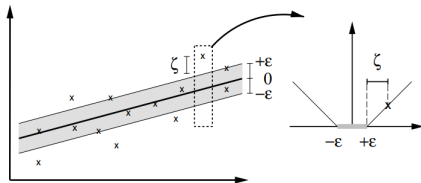
- 1 Линейная регрессия
- 2 Регуляризация
- 3 Разные функции потерь
- 4 Взвешенный учет наблюдений
- 5 Другие типы регрессии

## Регрессия опорных векторов

Эквивалентная формулировка (без ограничений неравенства):

$$\frac{1}{2} \|\beta\|_2^2 + C \sum_{n=1}^N \mathcal{L}(x_n^T \beta + \beta_0 - y_n) \rightarrow \min_{\beta \in \mathbb{R}^D}$$

$$\mathcal{L}(u) = \begin{cases} 0, & \text{if } |u| \leq \varepsilon \\ |u| - \varepsilon & \text{иначе} \end{cases} \quad \varepsilon - \text{нечувствительная ф-ция потерь}$$



Решение будет зависеть только от объектов с  $|\text{ошибка}| \geq \varepsilon$ ,  
называемых опорными векторами.

## Регрессия опорных векторов

Идея: допускаем небольшие  $\pm\varepsilon$  отклонения,  $L_2$  регуляризация.

$$\begin{cases} \frac{1}{2} \|\beta\|_2^2 \rightarrow \min_{\beta \in \mathbb{R}^D} & (\text{смещение } \beta_0 \text{ пишем явно}) \\ x_n^T \beta + \beta_0 - y_n \leq \varepsilon & n = \overline{1, N} \\ y_n - x_n^T \beta - \beta_0 \leq \varepsilon & n = \overline{1, N} \end{cases}$$

Если невозможно вписать все ошибки в интервал  $[-\varepsilon, \varepsilon]$ , воспользуемся методом общего вида:

$$\begin{cases} \frac{1}{2} \|\beta\|_2^2 + C \sum_{n=1}^N (\xi_n + \xi_n^*) \rightarrow \min_{\beta \in \mathbb{R}^D; \xi_n, \xi_n^* \in \mathbb{R}^N} \\ x_n^T \beta + \beta_0 - y_n \leq \varepsilon + \xi_n, & \xi_n \geq 0 & n = \overline{1, N} \\ y_n - x_n^T \beta - \beta_0 \leq \varepsilon + \xi_n^*, & \xi_n^* \geq 0 & n = \overline{1, N} \end{cases}$$

$C \geq 0$  - гиперпараметр, контролирующий противоречие между точностью и простотой модели.

## Orthogonal matching pursuit: задача

Метод Orthogonal Matching Pursuit решает задачу:

$$\begin{cases} \|X\beta - Y\|_2^2 \rightarrow \min_{\beta} \\ \|\beta\|_0 \leq K \end{cases}$$

или эквивалентную (с точностью до  $K(\varepsilon)$ ):

$$\begin{cases} \|\beta\|_0 \rightarrow \min_{\beta} \\ \|X\beta - Y\|_2^2 \leq \varepsilon \end{cases}$$

- $\|\beta\|_0 = \#[\text{число ненулевых весов}]$

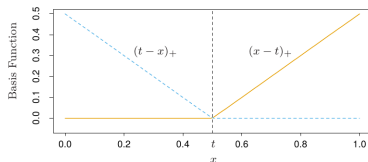
## Orthogonal matching pursuit: метод

- ❶ Инициализировать модель, равную константному нулю.
- ❷ Повторять, пока  $\|\beta\|_0 < K$  (или пока  $\|X\beta - Y\|_2^2 > \varepsilon$ )
  - ❶ добавить признак, максимально коррелирующий с ошибками прогноза последней модели.
  - ❷ переобучить линейную регрессию на данных (отобранные признаки, ошибки прогнозирования)
  - ❸ обновить ошибки прогнозирования
- Метод обобщается
  - на произвольный алгоритм прогнозирования
  - на произвольную меру взаимосвязи признаков и откликов



## MARS регрессия

- Рассмотрим  $S = \left\{ (x_j - t)_+, (t - x_j) \right\}_{t=x_j^1, \dots, x_j^N; j=1, \dots, D}$



- Multivariate Adaptive Regression Splines (MARS<sup>4</sup>):

$$\hat{y}(x) = \beta_0 + \sum_{m=1}^M \beta_m h_m(x)$$

где  $h_m(x) \in S$  или произведение элементов  $S$ .

<sup>4</sup> Др. название - Earth.

## Обучение MARS: наращивание модели

- Инициализация:  $M = 0$ ;  $\hat{y}(x) = \hat{\beta}_0 = \arg \min_{\beta_0} (y_n - \beta_0)^2$

## Обучение MARS: наращивание модели

- Инициализация:  $M = 0$ ;  $\hat{y}(x) = \hat{\beta}_0 = \arg \min_{\beta_0} (y_n - \beta_0)^2$
- Наращивание: последоват-но добавляем по 1 новой паре:
  - для каждого  $h_l(x)$  текущей модели, каждого  $j = 1, \dots, D$  и  $t \in \{x_j^1, \dots, x_j^N\}$  пробуем добавить в модель

$$\hat{\beta}_{M+1} h_l(x) (x_j - t)_+ + \hat{\beta}_{M+2} h_l(x) (t - x_j)_+$$

- $\hat{\beta}_{M+1}, \hat{\beta}_{M+2}$  находятся методом наименьших квадратов
- выбираем пару, приводящую к макс. уменьшению MSE
- ограничение: каждый признак присутствовать в произведении  $\leq 1$  раза (чтобы не было переобучения)
- пересчет MSE при сдвиге  $t$  можно пересчитывать не за  $O(N)$ , а за  $O(1)$  (т.к. сдвиг на константу)

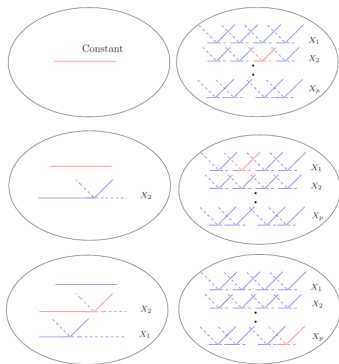
## Обучение MARS: наращивание модели

- Инициализация:  $M = 0$ ;  $\hat{y}(x) = \hat{\beta}_0 = \arg \min_{\beta_0} (y_n - \beta_0)^2$
- Наращивание: последоват-но добавляем по 1 новой паре:
  - для каждого  $h_l(x)$  текущей модели, каждого  $j = 1, \dots, D$  и  $t \in \{x_j^1, \dots, x_j^N\}$  пробуем добавить в модель

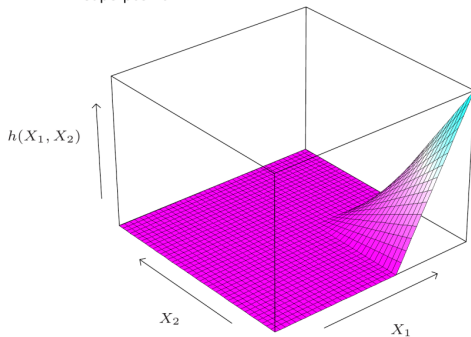
$$\hat{\beta}_{M+1} h_l(x) (x_j - t)_+ + \hat{\beta}_{M+2} h_l(x) (t - x_j)_+$$

- $\hat{\beta}_{M+1}, \hat{\beta}_{M+2}$  находятся методом наименьших квадратов
- выбираем пару, приводящую к макс. уменьшению MSE
- ограничение: каждый признак присутствовать в произведении  $\leq 1$  раза (чтобы не было переобучения)
- пересчет MSE при сдвиге  $t$  можно пересчитывать не за  $O(N)$ , а за  $O(1)$  (т.к. сдвиг на константу)
- Критерий остановки:
  - validation MSE не увеличивается
  - #слагаемых выше порога
  - степень мономов в слагаемых выше порога

# Обучение MARS: наращивание модели



superposition:



## Обучение MARS: упрощение модели<sup>5</sup>

- После окончания наращивания модели следует её упрощение:
- Последовательно выбрасывается по одному слагаемому, удаление которого меньше всего  $\downarrow$  MSE.
- Варианты:
  - смотреть на validation MSE
  - в MARS смотрят на train MSE, нормированный на  $\#$ степеней свободы:

$$\frac{\sum_{n=1}^N (y_n - \hat{y}_n)^2}{(1 - M_{\hat{y}}/N)^2}; \quad M_{\hat{y}} = \# \text{слагаемых} + \alpha \# \text{узлов}, \quad \alpha = 3$$

( $\#$ узлов= $\#$ задействованных порогов  $t$ )

---

<sup>5</sup>Hastie et al. The Elements of Statistical Learning.

# Нейросеть на основе MARS<sup>6</sup>

- Алгоритм MARS м. быть использован как начальное приближение для нейросети, которая будет потом донастраиваться.

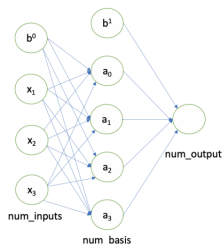
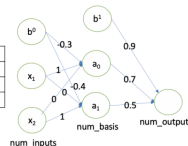


Figure 1: Structure of Converted Layers

Parameters of the network is calculated from the linear spline model in the form of (7). For the weight matrix  $W^{(1)}$  of the hidden layer,  $w_{i,j}^{(1)}$  equals 1 if basis  $h_i(X)$  is of the form  $R(X_j - t_i)$ , -1 if  $h_i(X)$  is of the form  $R(t_i - X_j)$  and 0 otherwise. The bias vector of the hidden layer is set to be  $(t_1, \dots, t_M)^T$ . The weight matrix (vector) of the output layer is calculated as  $(\beta_1, \dots, \beta_M)^T$  (here we assume the output dimension to be 1, as mentioned), and the bias of the output layer is equal to the intercept  $\beta_0$ . Shown in Figure 2 is a simple example of the described conversion.

Basis Func	Pruned	Coefficient
$h(x_2 - 0.4)$	No	0.5
$h(x_1 - 0.3)$	No	0.7
Intercept	No	0.9

Figure 2: One Example of Construction



<sup>6</sup><https://arxiv.org/pdf/2102.06554.pdf>

## Нейросеть на основе MARS

- $\hat{y}(x) = \beta_0 + \sum_{m=1}^M \beta_m h_m(x)$  - представляется в виде 2х слойной сети ( $h_m(x)$  вычисляются на скрытом слое с помощью сдвигов признаков, ReLU).
- рассматривается MARS сос лагаемыми 1го порядка (без произведений)
  - связь со скр. слоем каждый с каждым (просто где-то нулевой вес)
- Мультиплицируется скрытый слой несколько раз с единичной матрицей (без модификаций)
- Расширенная архитектура донастраивается BackProp.



## Заключение

- Линейная регрессия дает интерпретируемое аналитическое решение.
- Нелинейные закономерности моделируются:
  - добавлением нелинейных преобразований признаков
  - прогнозированием произвольной нелинейной функцией
- Регуляризация позволяет:
  - считать прогнозы для линейно-зависимых признаков
  - плавно настраивать сложность модели
  - отбирать признаки (лассо регрессия)
- Автоматический отбор признаков:
  - Лассо регрессия, orthogonal matching pursuit
- Различные функции потерь приводят к разным прогнозам.
- Устойчивость к выбросам достигается:
  - применением  $L_1$  потерь (лассо регрессия)
  - взвешенным учётом наблюдений (робастная регрессия)