



Тверской
государственный
технический
университет

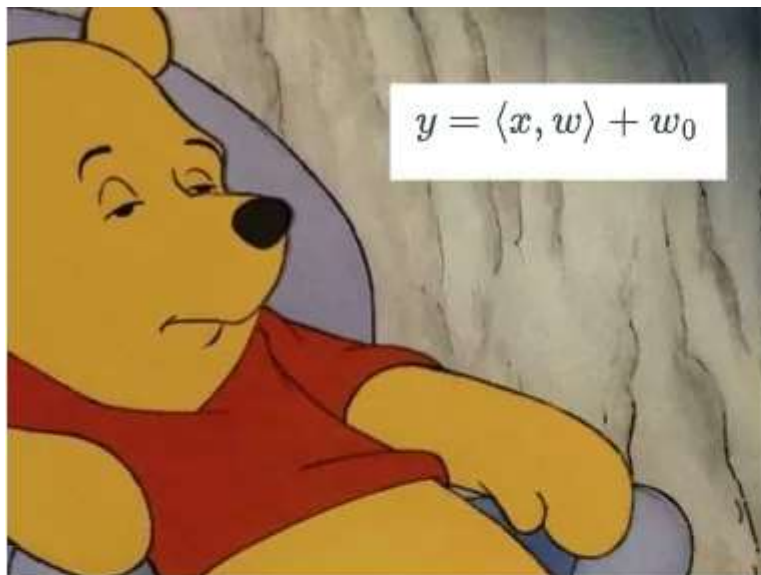
Интеллектуальные информационные системы

Линейные модели

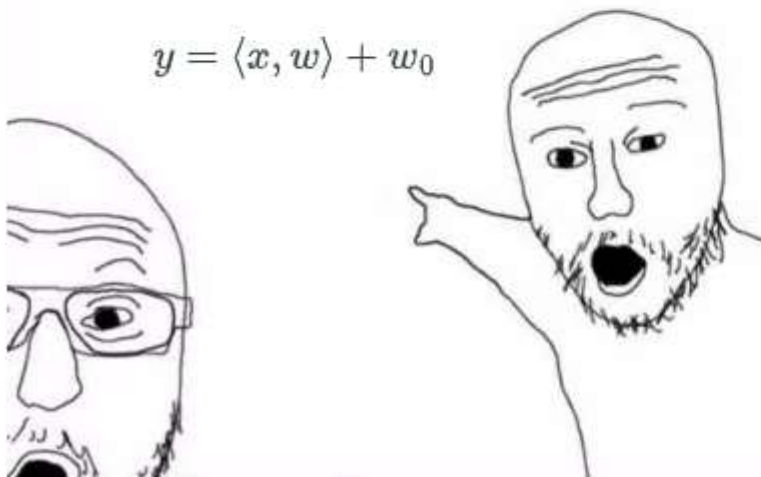
Материалы курса доступны по ссылке:

<https://github.com/AndreyShpigar/ML-course>

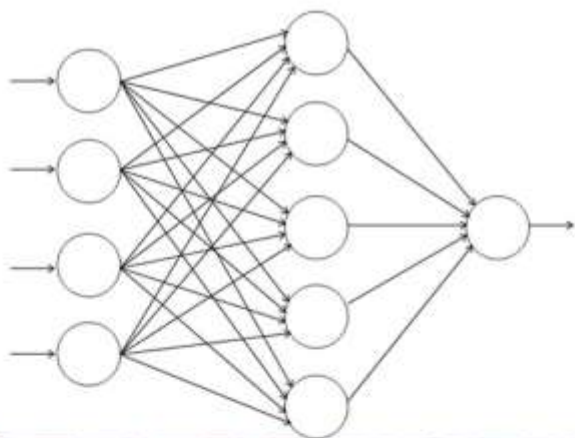
2024 г.



Employers
when you tell
them your app
uses linear
regression



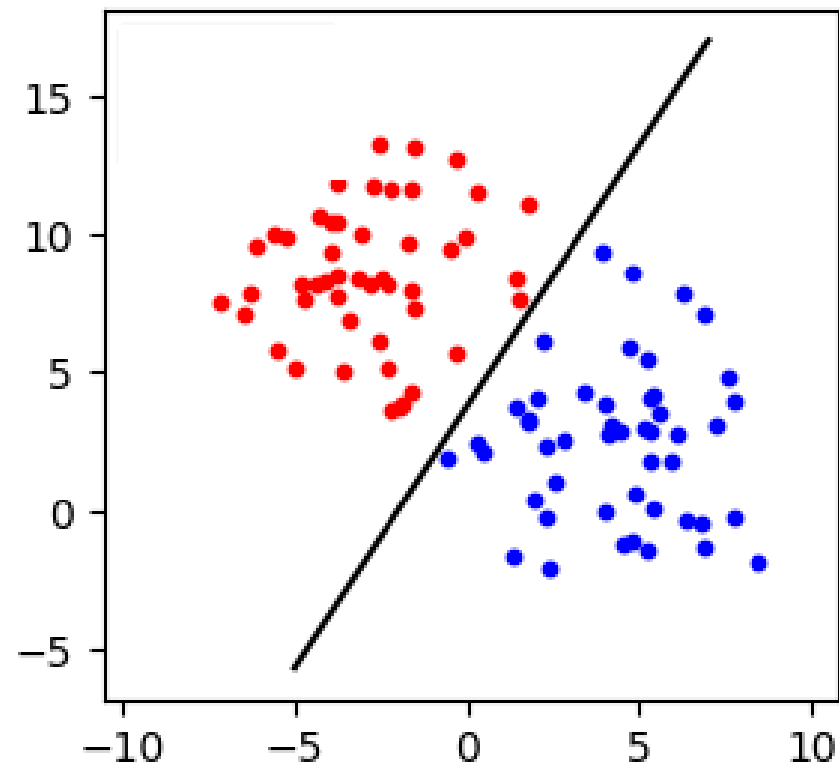
Employers
when you tell
them your app
uses “machine
learning and
A.I.”



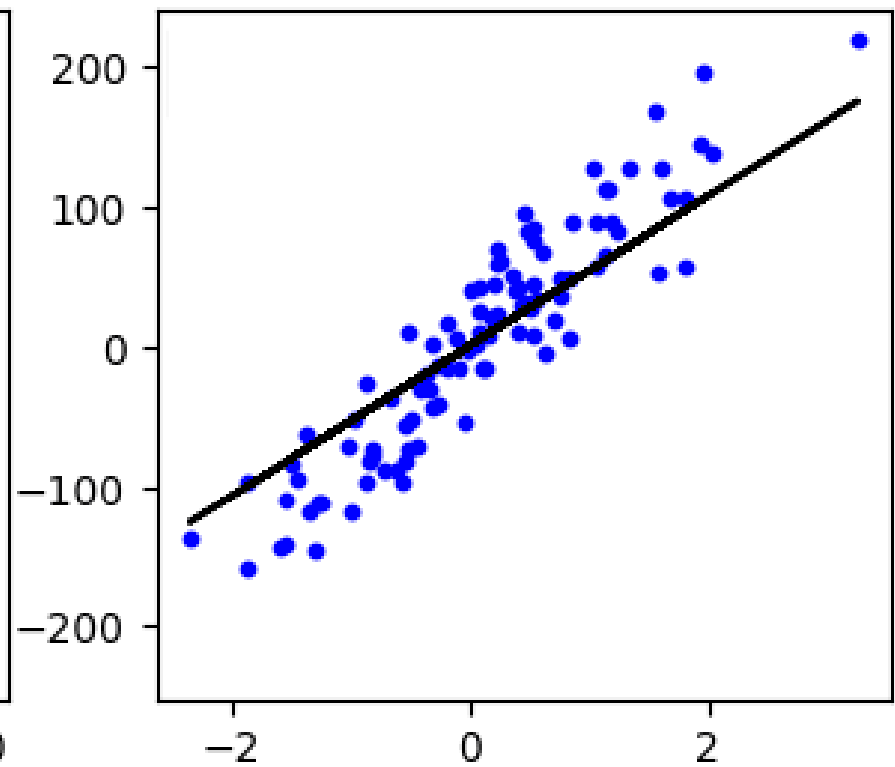
$$y = \langle x, w \rangle + w_0$$



Classification




Regression



Линейная регрессия

Линейная регрессия – метод предсказания вещественного выходного значения (целевой переменной) $y \in \mathbb{R}$ по вектору вещественных входных значений (признаков) $x \in \mathbb{R}^D$, при предположении, что ожидаемое выходное значение описывается линейной функцией входных значений

$$y(x) = w_0 + \sum_{i=1}^d w_i x_i$$



w_0 - свободный
коэффициент (сдвиг,
bias)

w_i - параметры модели
(коэффициенты, веса)

Обучение линейной регрессии

Задача оптимизации:

$$\frac{1}{l} \sum_{i=1}^l (\langle \mathbf{w}, \mathbf{x}_i \rangle - y_i)^2 \rightarrow \min_{\mathbf{w}}$$

В матричном виде:

$$\frac{1}{l} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 \rightarrow \min$$

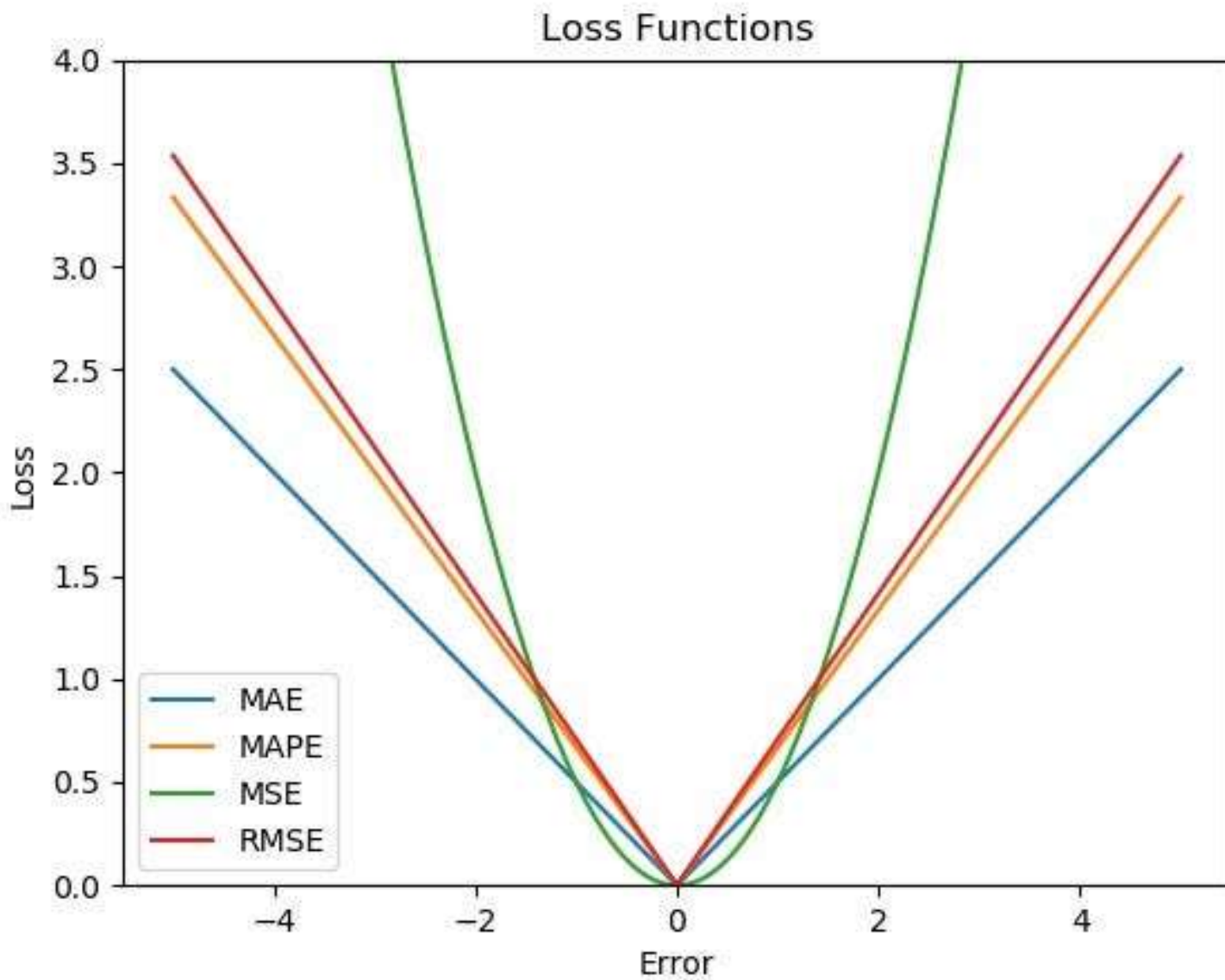
Решение:

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

Измерение ошибки в задачах регрессии

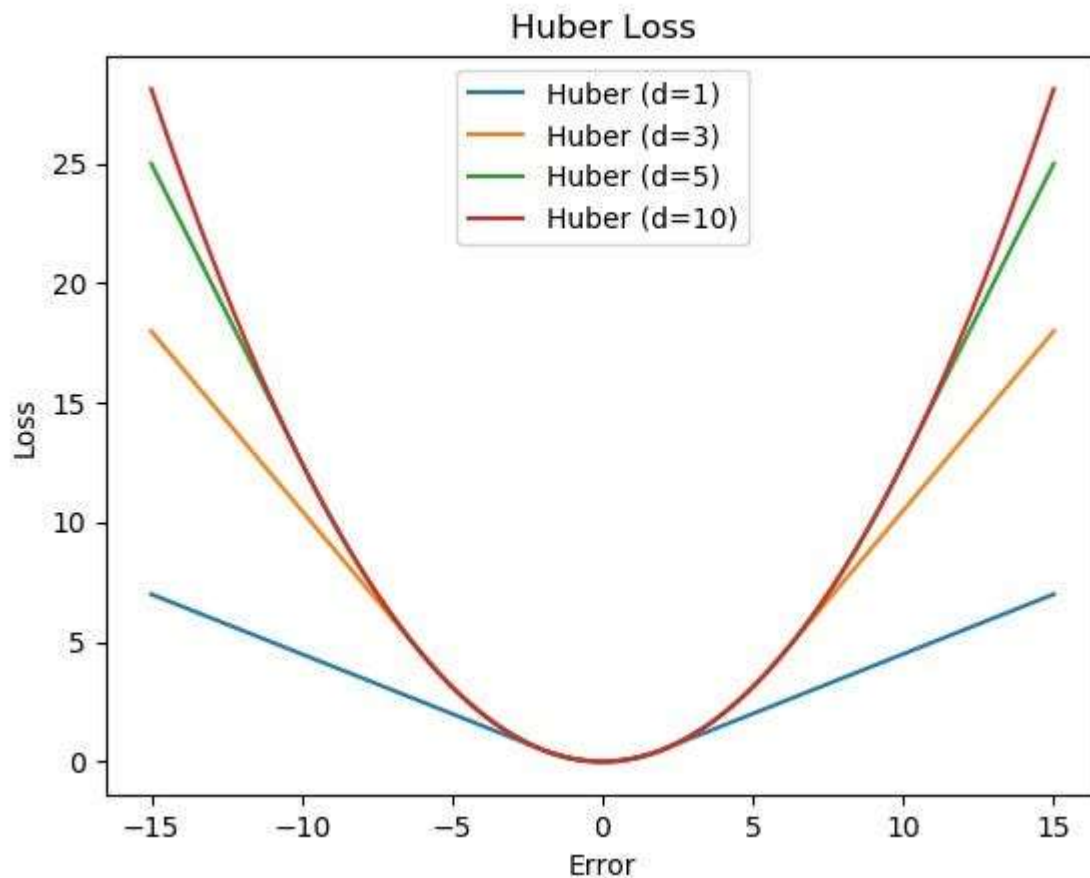
- $MSE(y, \hat{y}) = \frac{1}{l} \sum_{i=1}^l (\hat{y}_i - y_i)^2$
- $MAE(y, \hat{y}) = \frac{1}{l} \sum_{i=1}^l |\hat{y}_i - y_i|$
- $MAPE(y, \hat{y}) = \frac{1}{l} \sum_{i=1}^l \frac{|\hat{y}_i - y_i|}{y_i} \times 100\%$
- $RMSE(y, \hat{y}) = \sqrt{\frac{1}{l} \sum_{i=1}^l (\hat{y}_i - y_i)^2}$
- $MSLE(y, \hat{y}) = \frac{1}{l} \sum_{i=1}^l (\log(1 + \hat{y}_i) - \log(1 + y_i))^2$

Измерение ошибки в задачах регрессии



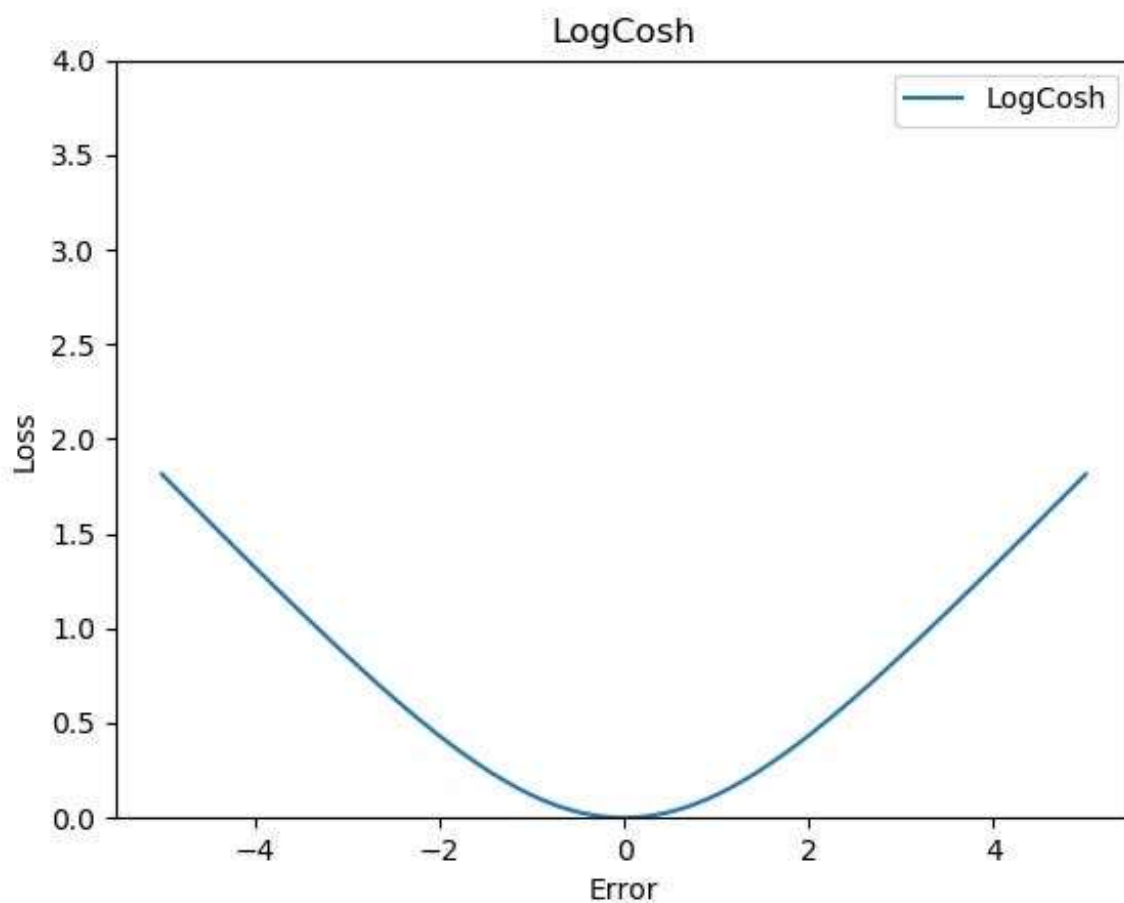
Измерение ошибки в задачах регрессии

$$\text{Huber}_{\delta}(y, \hat{y}) = \begin{cases} \frac{1}{2}(\hat{y} - y)^2, & |\hat{y} - y| < \delta \\ \delta \left(|\hat{y} - y| - \frac{1}{2}\delta \right), & |\hat{y} - y| \geq \delta \end{cases}$$



Измерение ошибки в задачах регрессии

$$\text{LogCosh}(y, \hat{y}) = \sum_{i=1}^l \log(\cosh(\hat{y}_i - y_i))$$

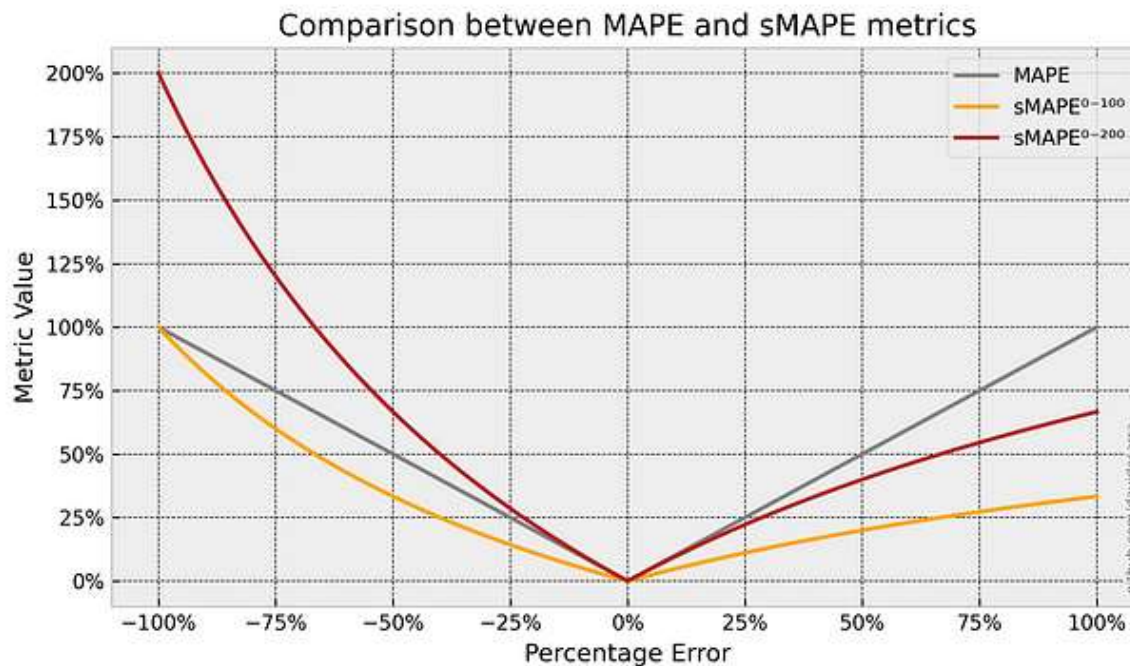


Измерение ошибки в задачах регрессии

Symmetric mean absolute percentage error:

$$SMAPE^{0-100}(y, \hat{y}) = \frac{100\%}{l} \sum_{i=1}^l \frac{|\hat{y}_i - y_i|}{(|y_i| + |\hat{y}_i|)}$$

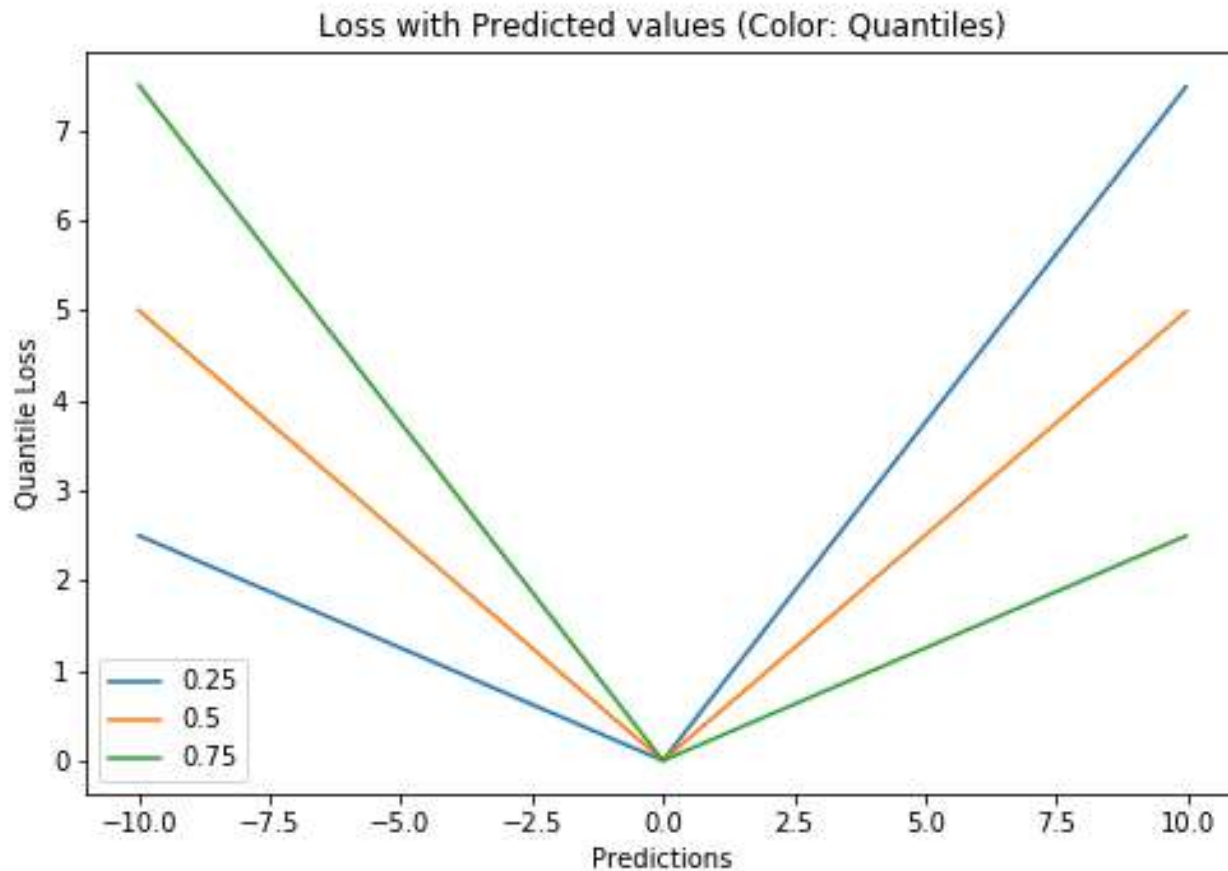
$$SMAPE^{0-200}(y, \hat{y}) = \frac{100\%}{l} \sum_{i=1}^l \frac{|\hat{y}_i - y_i|}{(|y_i| + |\hat{y}_i|)/2}$$



Измерение ошибки в задачах регрессии

Quantile Loss

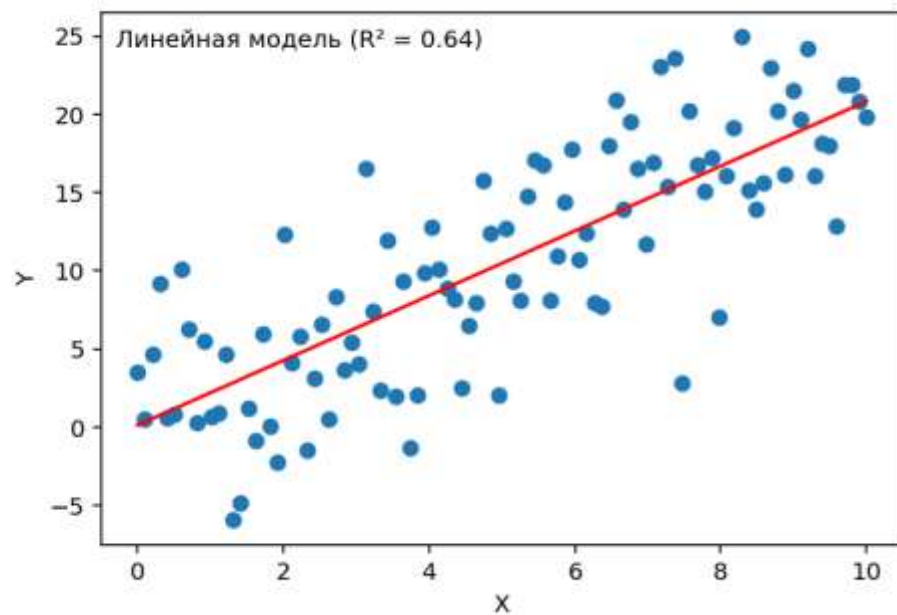
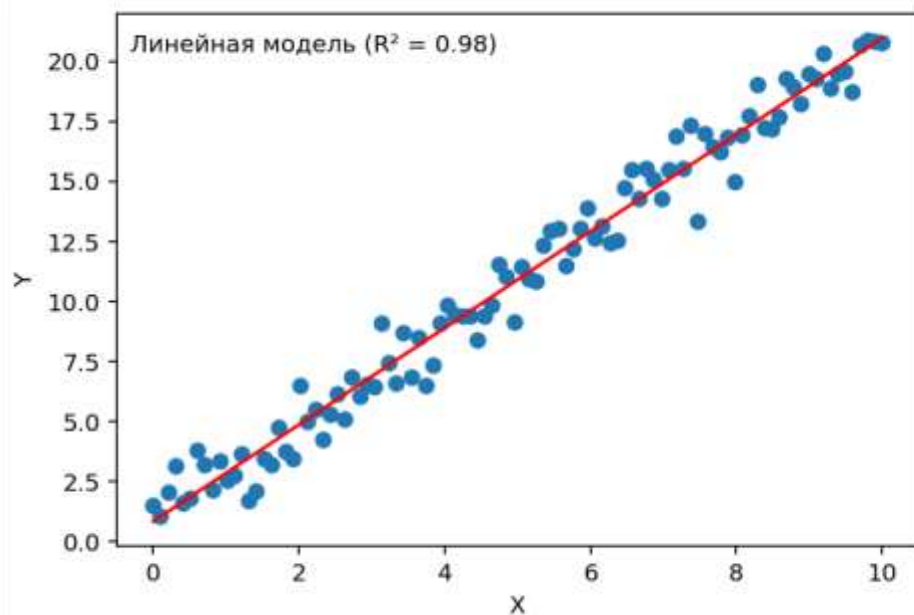
$$Q(y, \hat{y}) = \frac{1}{l} \sum_{i=1}^l \rho_{\tau}(y_i - \hat{y}_i)$$



Измерение ошибки в задачах регрессии

Coefficient of determination

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^l (\hat{y}_i - y_i)^2}{\sum_{i=1}^l (y_i - \bar{y})^2}$$



Градиентные методы оптимизации

Градиент функции $f : \mathbb{R}^n \rightarrow \mathbb{R}$ - вектор частных производных

$$\nabla f(x_1, \dots, x_n) = \left(\frac{\partial f}{\partial x_j} \right)_{j=1}^n$$

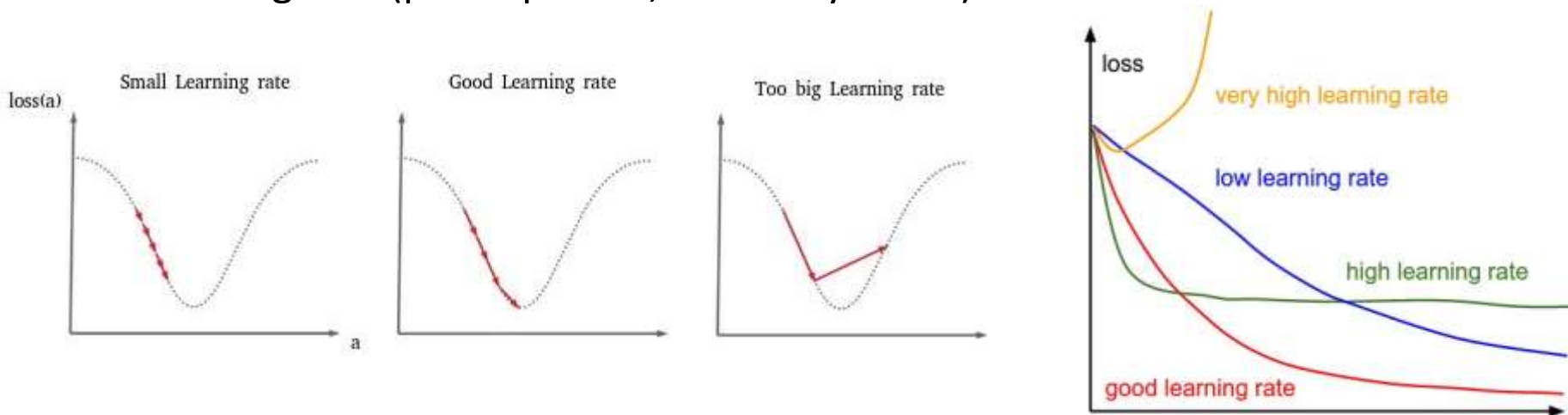
Градиентный шаг:

$$\mathbf{w}^{(k)} = \mathbf{w}^{(k-1)} - \alpha \nabla Q(\mathbf{w}^{(k-1)})$$

$\mathbf{w}^{(0)}$ - начальный набор параметров

$Q(\mathbf{w})$ - значение функционала ошибки для набора параметров \mathbf{w}

α - learning rate (размер шага, темп обучения)



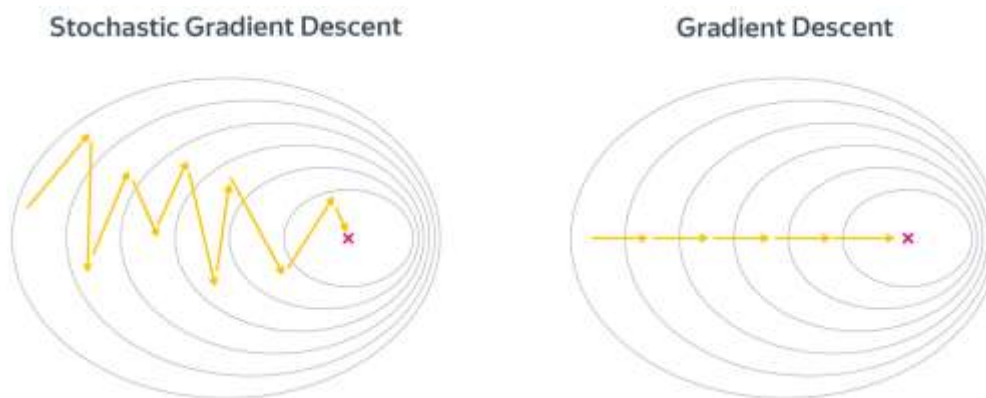
Оценивание градиента

- Полный градиент:

$$\nabla_w Q(w) = \frac{1}{l} \sum_{i=1}^l \nabla_w q_i(w)$$

- Стохастический градиентный спуск (SGD):

$$w^{(k)} = w^{(k-1)} - \alpha \nabla q_{i_k}(w^{(k-1)})$$



- Средний стохастический градиент (SAG):

$$w^{(k)} = w^{(k-1)} - \alpha \frac{1}{l} \sum_{i=1}^l z_i^{(k)}$$


Модификации градиентного спуска

- Метод инерции (momentum):

$$\mathbf{w}^{(k)} = \mathbf{w}^{(k-1)} - \mathbf{h}_k$$

$$\mathbf{h}_k = \beta \mathbf{h}_{k-1} + \alpha \nabla_{\mathbf{w}} Q(\mathbf{w}^{(k-1)})$$


вектор инерции


параметр метода, определяющий
скорость затухания градиентов с
предыдущих шагов

- Nesterov momentum:

$$\mathbf{h}_k = \beta \mathbf{h}_{k-1} + \alpha \nabla_{\mathbf{w}} Q(\mathbf{w}^{(k-1)} + \beta \mathbf{h}_{k-1})$$

Adaptive learning rate

- Adagrad

$$G_{k+1} = G_k + (\nabla f(x_k))^2$$
$$x_{k+1} = x_k - \frac{\alpha}{\sqrt{G_{k+1} + \varepsilon}} \nabla f(x_k).$$

- RMSProp

$$G_{k+1} = \gamma G_k + (1 - \gamma)(\nabla f(x_k))^2$$
$$x_{k+1} = x_k - \frac{\alpha}{\sqrt{G_{k+1} + \varepsilon}} \nabla f(x_k).$$

- Adam (ADaptive Momentum)

$$v_{k+1} = \beta_1 v_k + (1 - \beta_1) \nabla f(x_k)$$
$$G_{k+1} = \beta_2 G_k + (1 - \beta_2)(\nabla f(x_k))^2$$
$$x_{k+1} = x_k - \frac{\alpha}{\sqrt{G_{k+1} + \varepsilon}} v_{k+1}.$$

Регуляризация

- При наличии в выборке линейно зависимых (мультиколлинеарных) признаков всегда найдется такой вектор \mathbf{v} , что для любого объекта \mathbf{x} :

$$\langle \mathbf{v}, \mathbf{x} \rangle = 0$$

Множество решений:

$$\langle \mathbf{w} + \alpha \mathbf{v}, \mathbf{x} \rangle = \langle \mathbf{w}, \mathbf{x} \rangle + \alpha \langle \mathbf{v}, \mathbf{x} \rangle = \langle \mathbf{w}, \mathbf{x} \rangle$$

тоже решение
оптимизационной задачи
для любого α

оптимальный вектор весов
(решение оптимизационной задачи)

Подмена задачи:

$$Q_{\lambda}(\mathbf{w}) = Q(\mathbf{w}) + \lambda R(\mathbf{w})$$

коэффициент
регуляризации

регуляризатор

- L_1 -регуляризация:

$$R_1(\mathbf{w}) = \|\mathbf{w}\|_1 = \sum_{i=1}^d |\mathbf{w}|_i$$

- L_2 -регуляризация:

$$R_2(\mathbf{w}) = \|\mathbf{w}\|_2 = \sum_{i=1}^d w^2$$

- Elastic net regularization:

$$R_E(\mathbf{w}) = R_1(\mathbf{w}) + R_2(\mathbf{w})$$

Преобразования признаков

Нелинейные преобразования исходного признакового пространства вида

$$x = (x_1, \dots, x_d) \rightarrow \varphi(x) = (\varphi_1(x), \dots, \varphi_m(x))$$

- Полиномиальное (квадратичные или более высоких порядков): $\varphi(x) = (x_1, \dots, x_d, x_1^2, \dots, x_d^2, x_1 x_2, \dots, x_{d-1} x_d)$
- Логарифмическое: $\log(x_i)$
- Экспоненциальное: $\exp(\frac{\|x - \mu\|^2}{\sigma})$
- Синусоидальное: $\sin(x_i / T)$
- Любое другое, если вы можете обосновать и реализовать это преобразование

Масштабирование признаков

- Стандартизация (standard scaling):

$$z = \frac{x - \mu}{\sigma}$$

- Масштабирование к диапазону $[a, b]$ (min-max normalization):

$$x' = a + \frac{(x - \min(x))(b - a)}{\max(x) - \min(x)}$$

- Robust scaling (standardization using median and IQR):

$$x' = \frac{x - Q_2(x)}{Q_3(x) - Q_1(x)}$$