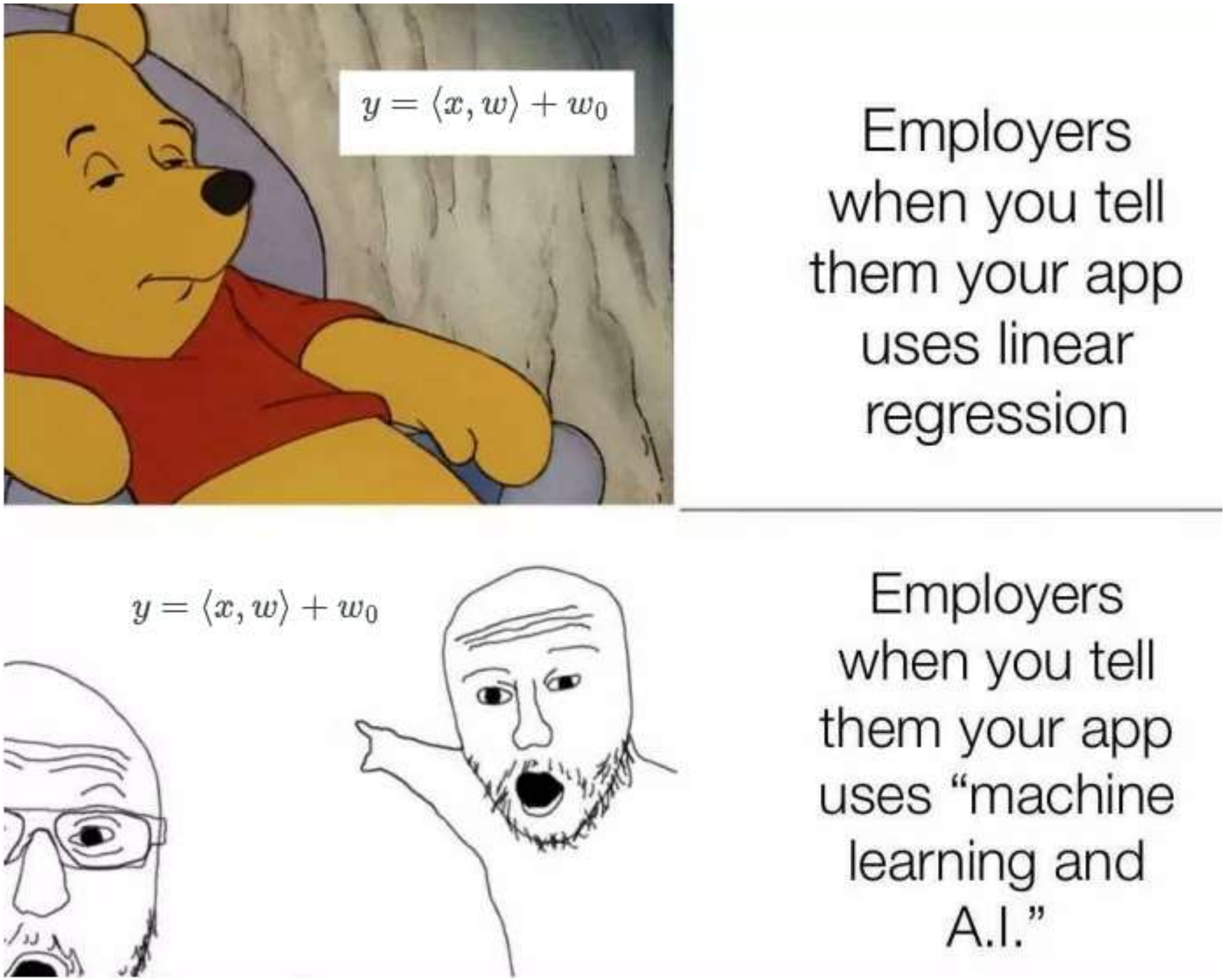
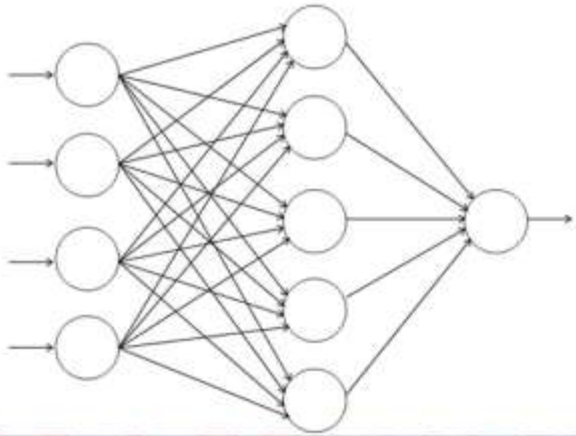
Тверской
государственный
технический
университет

Интеллектуальные информационные системы

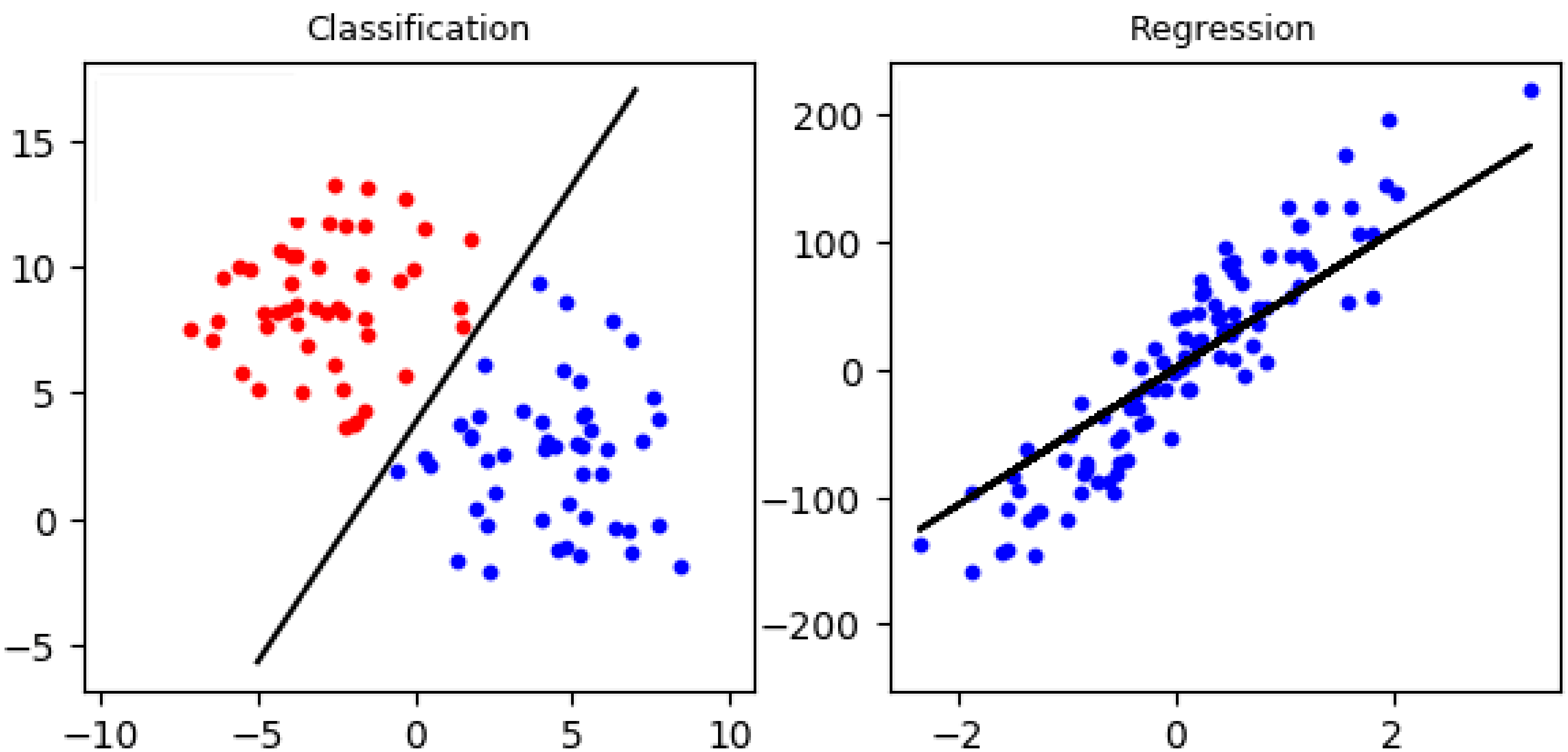
Линейные модели

2025 г.


$y = \langle x, w \rangle + w_0$
Employers when you tell them your app uses linear regression
$y = \langle x, w \rangle + w_0$
Employers when you tell them your app uses "machine learning and A.I."



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


Classification
15
10
5
0
−5
−10
−5
0
5
10
Regression
200
100
0
−100
−200
−2
0
2

Линейная регрессия

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

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

$\boldsymbol{w_0}$ - свободный коэффициент (сдвиг, bias)

$\boldsymbol{w_i}$ - параметры модели (коэффициенты, веса)

Обучение линейной регрессии

Задача оптимизации:

$$\frac{1}{l}\sum_{i=1}^{l}(\langle w, x_i\rangle - y_i)^2 \to \min_w$$

В матричном виде:

$$\frac{1}{l}\|Xw - y\|^2 \to \min_w$$

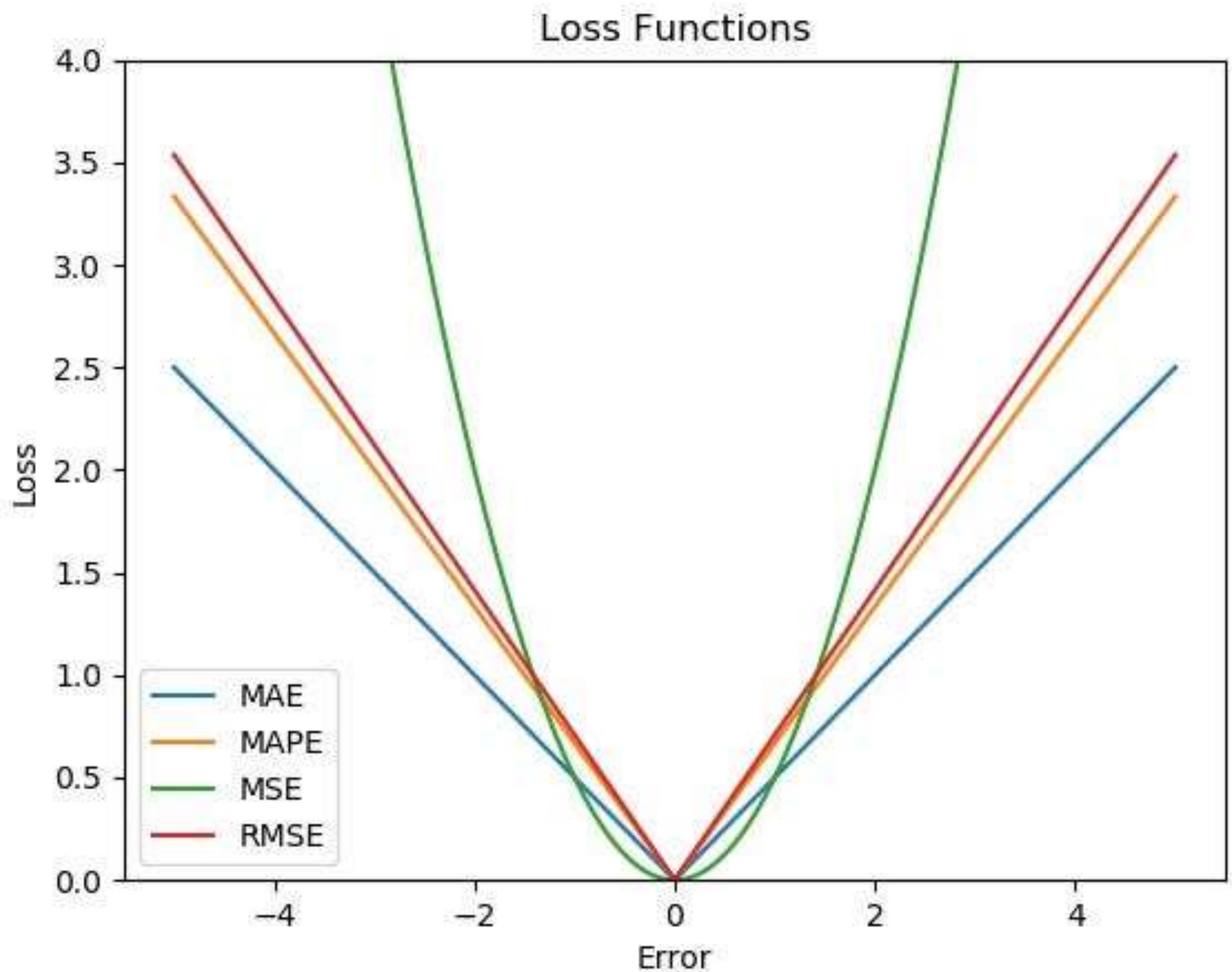
Решение:

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

Измерение ошибки в задачах регрессии

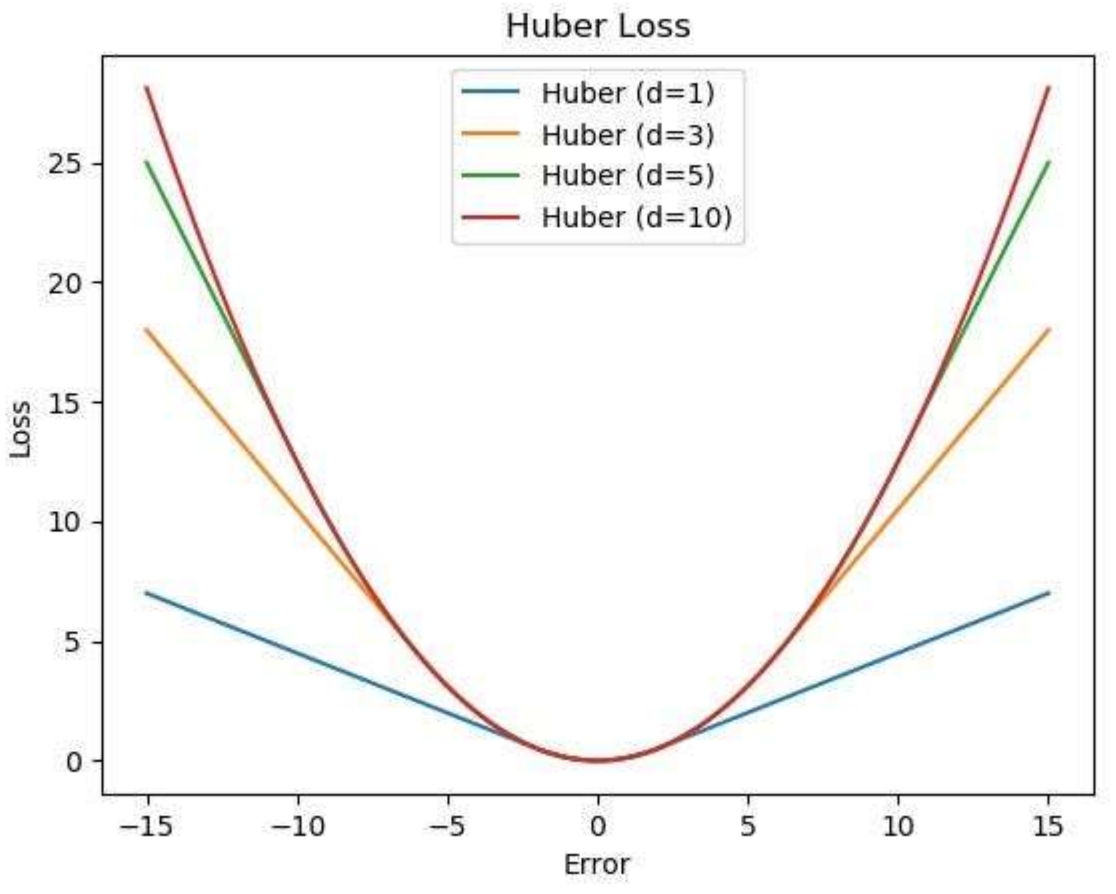
- $MSE(y, \widehat{y}) = \frac{1}{l}\sum_{i=1}^{l}(\widehat{y}_i - y_i)^2$
- $MAE(y, \widehat{y}) = \frac{1}{l}\sum_{i=1}^{l}|\widehat{y}_i - y_i|$
- $MAPE(y, \widehat{y}) = \frac{1}{l}\sum_{i=1}^{l}\frac{|\widehat{y}_i - y_i|}{y_i} \times 100\%$
- $RMSE(y, \widehat{y}) = \sqrt{\frac{1}{l}\sum_{i=1}^{l}(\widehat{y}_i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

Измерение ошибки в задачах регрессии



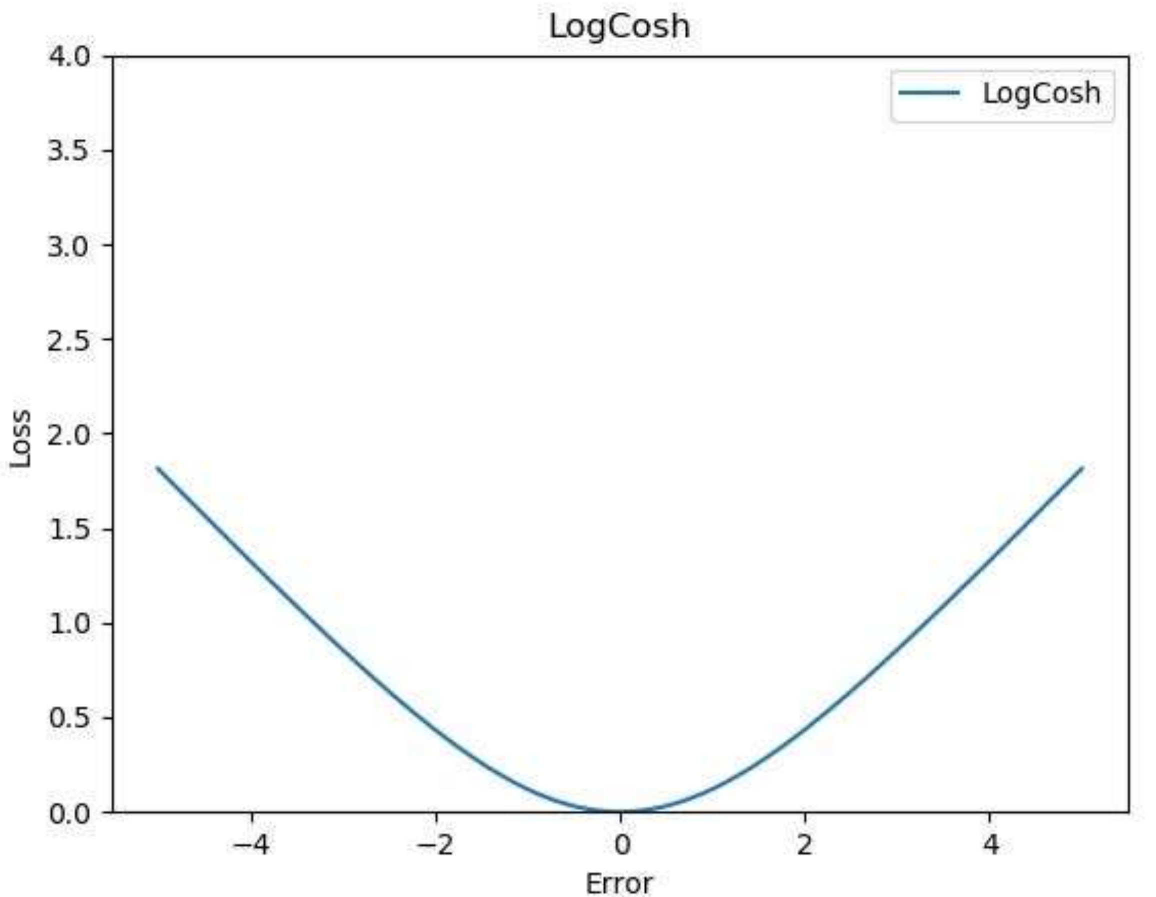
Измерение ошибки в задачах регрессии

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



Измерение ошибки в задачах регрессии

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

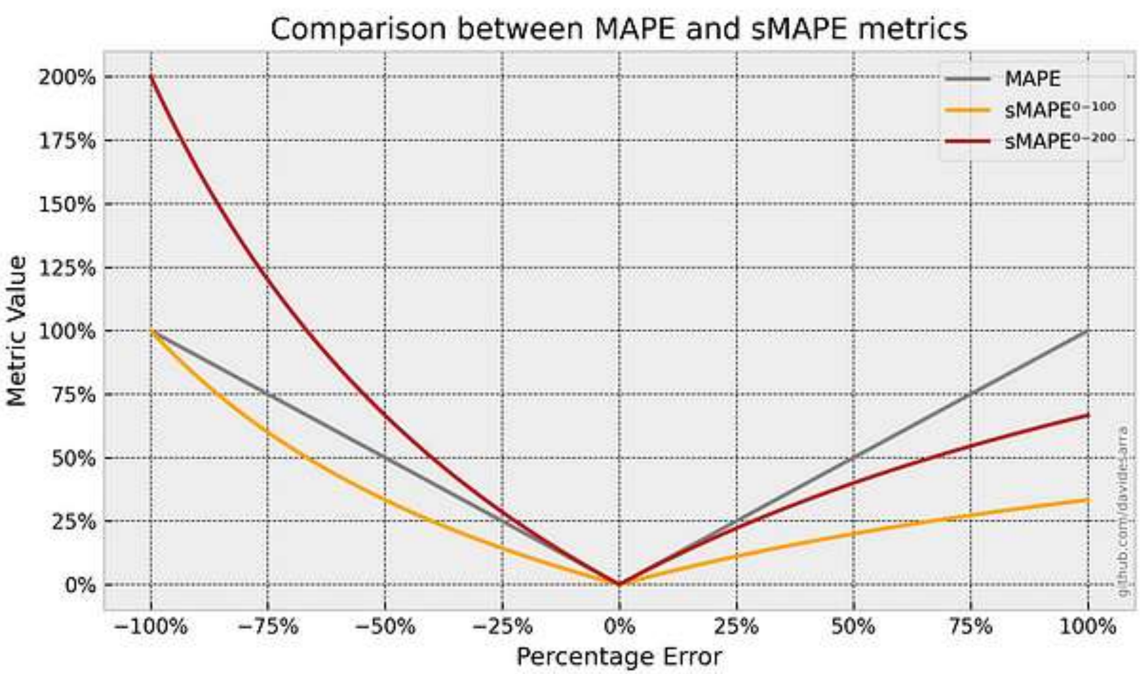


Измерение ошибки в задачах регрессии

Symmetric mean absolute percentage error:

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

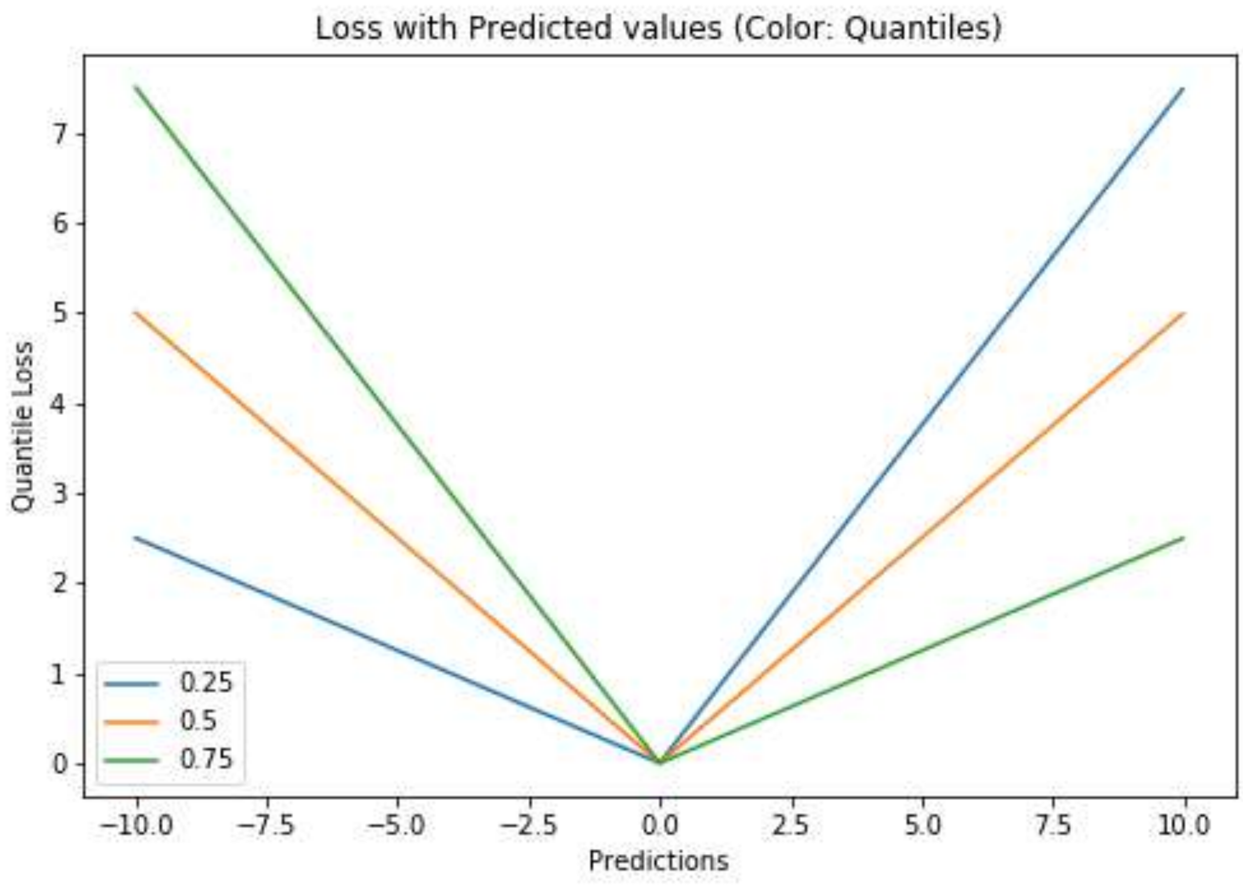
$$SMAPE^{0-200}(y,\hat{y}) = \frac{100\%}{l}\sum_{i=1}^{l}\frac{|\hat{y}_i - y_i|}{(|y_i| + |\hat{y}_i|)/2}$$



Измерение ошибки в задачах регрессии

Quantile Loss

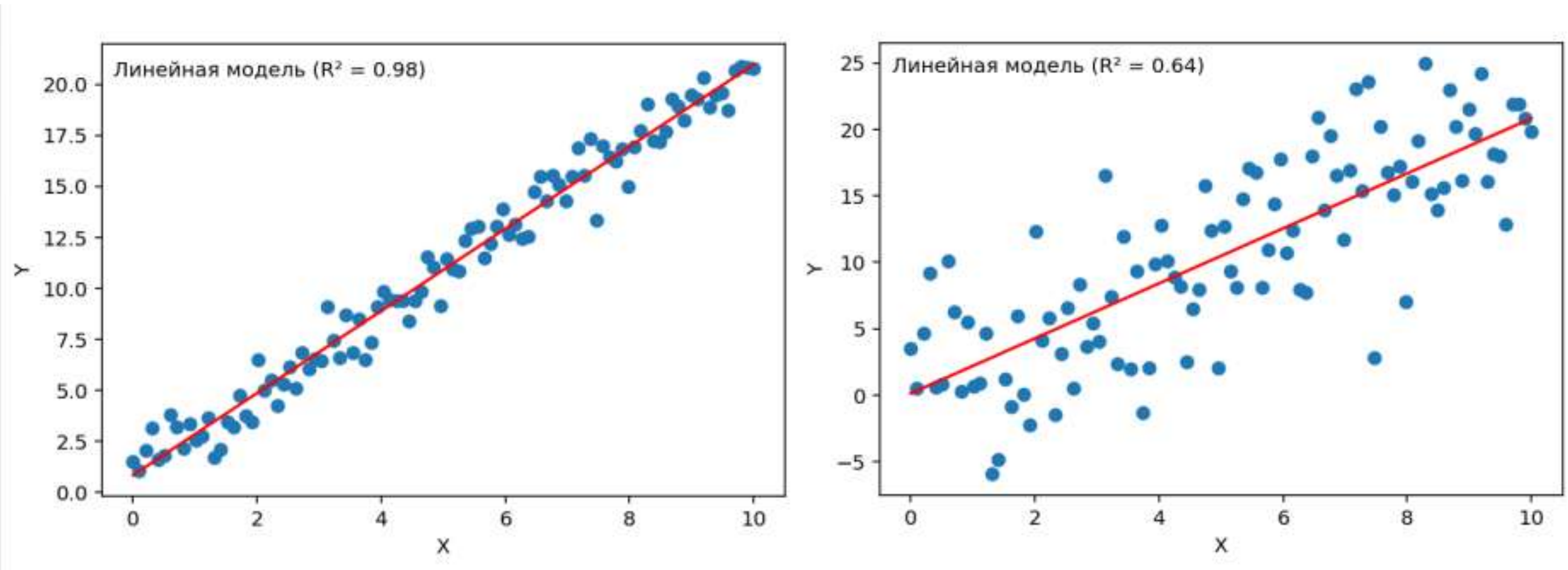
$$Q(y, \hat{y}) = \frac{1}{l} \sum_{i=1}^{l} \rho_\tau (y_i - \hat{y}_i)$$



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 $f : \mathbb{R}^n \to \mathbb{R}$ - <u>вектор частных производных</u>

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

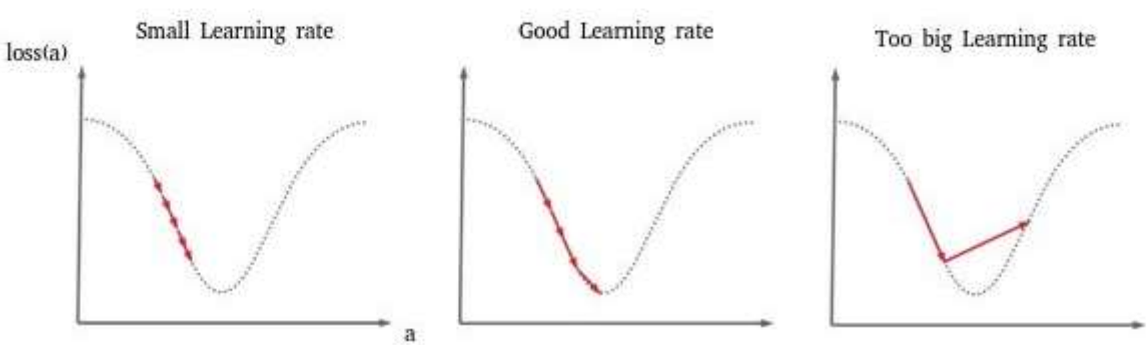
Градиентный шаг:

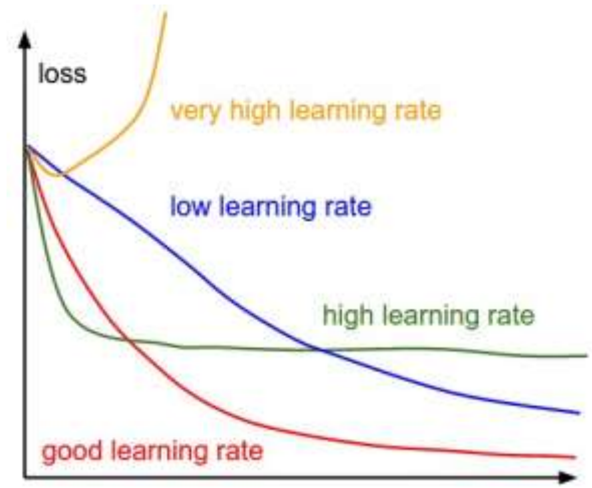
$$\boldsymbol{w^{(k)} = w^{(k-1)} - \alpha \nabla Q\left(w^{(k-1)}\right)}$$

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

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

$\boldsymbol{\alpha}$ - 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):

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

- Средний стохастический градиент (SAG):

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

Источник: https://education.yandex.ru/handbook/ml/article/optimizaciya-v-ml

Модификации градиентного спуска

- Метод инерции (momentum):

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

$$\boldsymbol{h}_k = \boldsymbol{\beta}\, \boldsymbol{h}_{k-1} + \boldsymbol{\alpha} \nabla_w \boldsymbol{Q}\left(\boldsymbol{w}^{(k-1)}\right)$$

вектор инерции

параметр метода, определяющий скорость затухания градиентов с предыдущих шагов

- Nesterov momentum:

$$\boldsymbol{h}_k = \boldsymbol{\beta}\, \boldsymbol{h}_{k-1} + \boldsymbol{\alpha} \nabla_w \boldsymbol{Q}\left(\boldsymbol{w}^{(k-1)} + \boldsymbol{\beta}\, \boldsymbol{h}_{k-1}\right)$$

Источник: https://education.yandex.ru/handbook/ml/article/optimizaciya-v-ml

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 $\boldsymbol{v}$, что для любого объекта $\boldsymbol{x}$:

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

Множество решений:

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

тоже решение оптимизационной задачи для любого $\boldsymbol{\alpha}$

оптимальный вектор весов (решение оптимизационной задачи)

<u>Подмена задачи:</u>

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

<u>коэффициент регуляризации</u>

<u>регуляризатор</u>

- L_1-регуляризация:

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

- L_2-регуляризация:

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

- Elastic net regularization:

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

Преобразования признаков

<u>Нелинейные преобразования</u> исходного признакового пространства вида

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

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

Масштабирование признаков

- Стандартизация (standard scaling):

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

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

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

- Robust scaling (standardization using median and IQR):

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