Введение в АД

Лекция 3

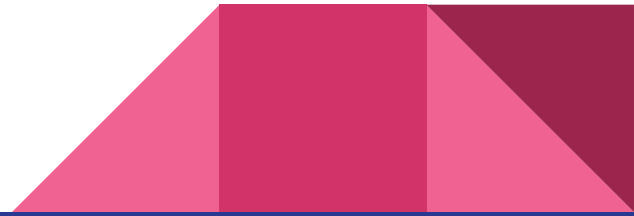
Задача классификации

ФЭФМ МФТИ

Весенний семестр 2023

Quick recap

- Train, validation, test stages
- Feature engineering (one-hot encoding)
- Linear regression
- Regularization (L1, L2)



Linear regression

$$f_w(x_i) = \langle w, x_i \rangle + w_0$$

to exclude w0

$$(x_{i1} \quad \dots \quad x_{iD}) \cdot \begin{pmatrix} w_1 \\ \vdots \\ w_D \end{pmatrix} + w_0 = (1 \quad x_{i1} \quad \dots \quad x_{iD}) \cdot \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_D \end{pmatrix}$$

Let's suppose we have Mean-squared error (MSE) as a Loss function

$$\text{MSE}(f, X, y) = \frac{1}{N}|y - Xw|_2^2$$

how to differentiate matrix functions?

$$||Ax - b||^2 = \langle Ax - b, Ax - b \rangle$$

$$[D_{x_0}\langle Ax - b, Ax - b\rangle](h) =$$

$$\langle [D_{x_0}(Ax - b)](h), Ax_0 - b\rangle + \langle Ax_0 - b, [D_{x_0}(Ax - b)](h)\rangle$$

$$= 2\langle Ax_0 - b, [D_{x_0}(Ax - b)](h)\rangle =$$

$$= 2\langle Ax_0 - b, Ah\rangle = \langle 2A^T(Ax_0 - b), h\rangle$$

$$X^T(y - Xw) = 0$$

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

Regularization

There are two problems regarding linear regression:

- Multicollinearity problem (features are nearly linear dependent, X^TX nearly singular)
- Low condition number (difference between eigenvalues of X^TX)

This result in unlikely big weights numbers in numerical solution

Thus we have regularization (L1, L2, or even both [ElasticNet])

$$L_2 = ||Y - Xw||_2^2 + \lambda^2||w||_2^2 \longrightarrow w = (X^TX + \lambda^2 I)^{-1}X^TY$$

$$L_1 = ||Y - Xw||_2^2 + \lambda^2||w||_1 \longrightarrow$$ No good analytical solution

Сегодня

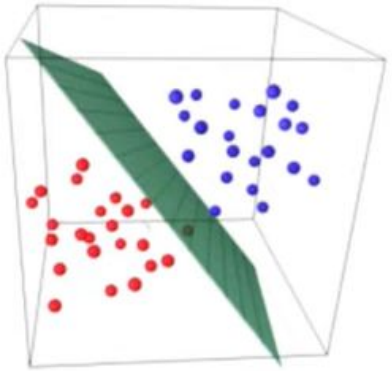
- Linear classifiers
- Multiple classes solution
- Logistic regression

Linear classifier

$$c(x) = \begin{cases} 1, & \text{if } f(x) \geq 0 \\ -1, & \text{if } f(x) < 0 \end{cases}$$

or equivalently

$$c(x) = \text{sign}(f(x)) = \text{sign}(x^T w)$$



Let's define linear model's Margin as

$$M_i = y_i \cdot f(x_i) = y_i \cdot x_i^T w$$

main property:
negative margin reveals misclassification

$$M_i > 0 \Leftrightarrow y_i = c(x_i)$$

$$M_i \leq 0 \Leftrightarrow y_i \neq c(x_i)$$

Remembering old paradigm

$$\text{Empirical risk} = \sum_{\text{by objects}} \text{Loss on object} \rightarrow \min_{\text{model params}}$$

Essential loss is misclassification

$$L_{\text{mis}}(y_i^t, y_i^p) = [y_i^t \neq y_i^p] = \\ = [M_i \leq 0]$$

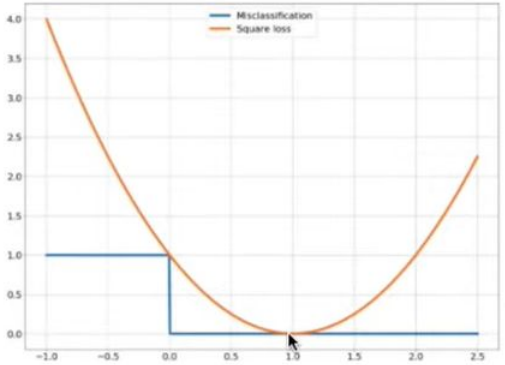
Disadvantages

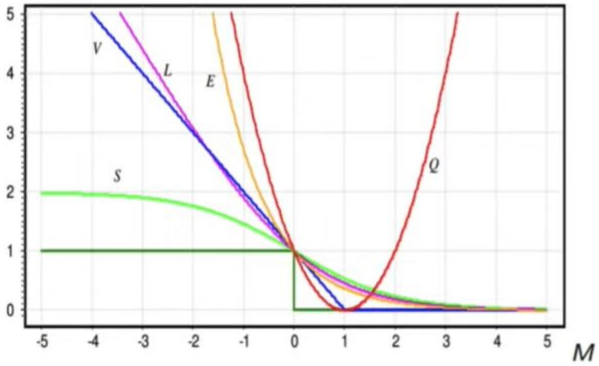
- Not differentiable
- Overlooks confidence

Let's treat classification problem as regression problem: $Y \in \{-1, 1\} \mapsto Y \in R$

thus we optimize MSE

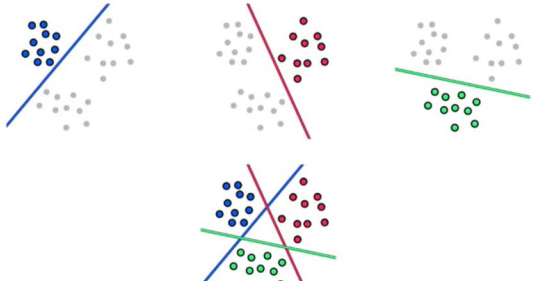
$$L_{\text{MSE}} = (y_i - x_i^T w)^2 = \frac{(y_i^2 - y_i \cdot x_i^T w)^2}{y_i^2} =$$
$$= (1 - y_i \cdot x_i^T w)^2 = (1 - M_i)^2$$

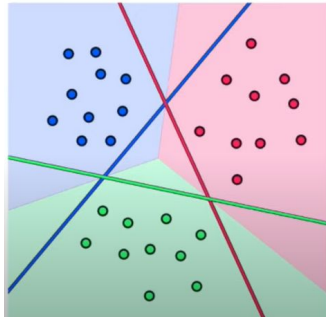


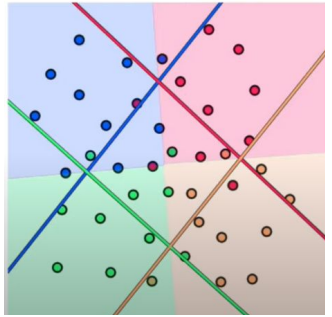


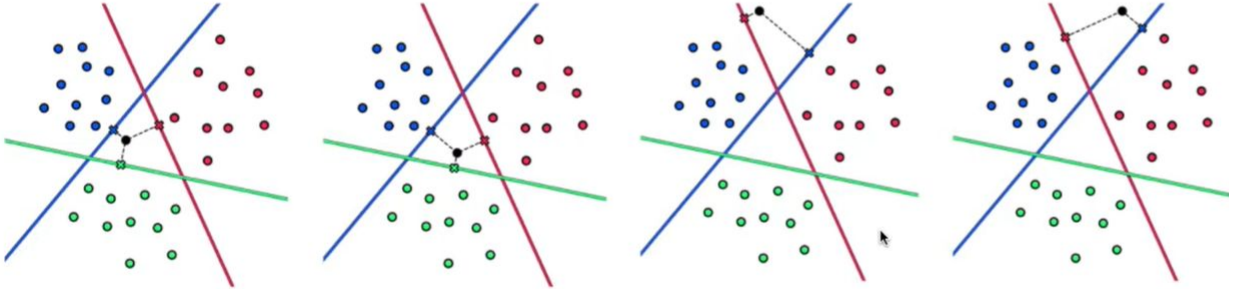
$Q(M) = (1 - M)^2$
$V(M) = (1 - M)_+$
$S(M) = 2(1 + e^M)^{-1}$
$L(M) = \log_2(1 + e^{-M})$
$E(M) = e^{-M}$

Multiclass problem: one vs rest

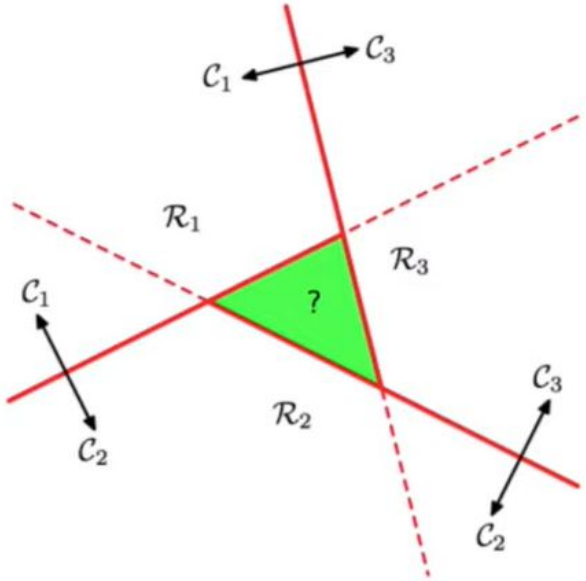








Multiclass problem: one vs one



	One vs Rest	One vs One
#classifiers	k	k(k-1)/2
dataset for each	full	subsampled

Logistic regression

I. Let's try to predict probability of an object to have positive class

$$p_+ = P(y = 1|x) \in [0, 1]$$

II. But all we can predict is a real number!

$$y = x^T w \in R$$

III. Time for some tricks

$$\frac{p_+}{1 - p_+} \in [0, +\infty)$$

$$\log \frac{p_+}{1 - p_+} \in R$$

Here is the match

IV. Reverse to closed form

$$\frac{p_+}{1 - p_+} = \exp(x^T w)$$

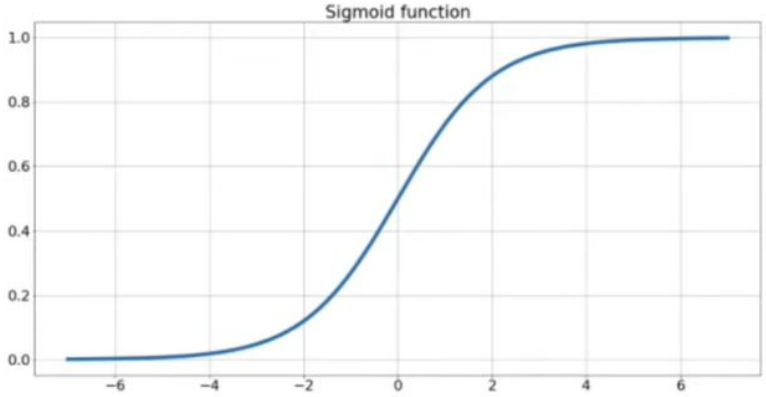
$$p_+ = \frac{1}{1 + exp(-x^T w)} = \sigma(x^T w)$$

$$\sigma(x) = \frac{1}{1 + exp(-x)}$$

Sigmoid is odd relative to (0, 0.5) point

Symmetric property:

$$1 - \sigma(x) = \sigma(-x)$$



Derivative: $\sigma'(x) = \sigma(x) \cdot (1 - \sigma(x))$

Maximum Likelihood Estimation (MLE)

Just to remind

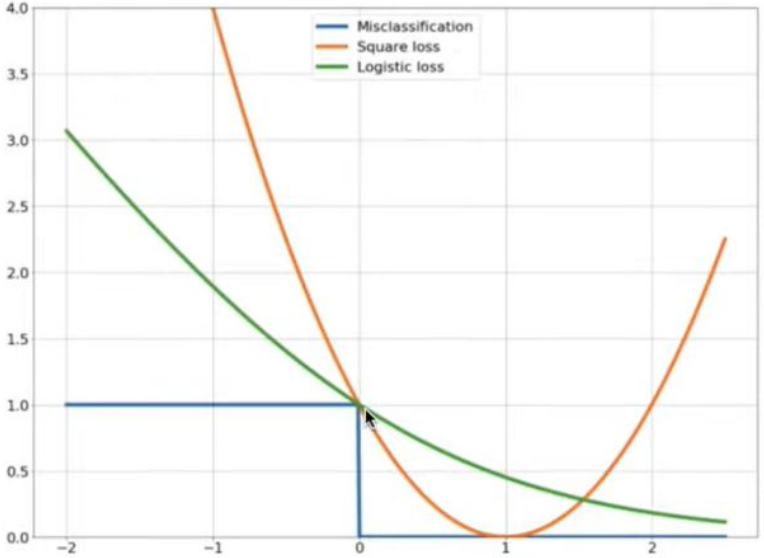
$$\log L(w|X,Y) = \log P(X,Y|w) = \log \prod_{i=1}^{n} P(x_i, y_i|w)$$

Calculating probabilities for objects

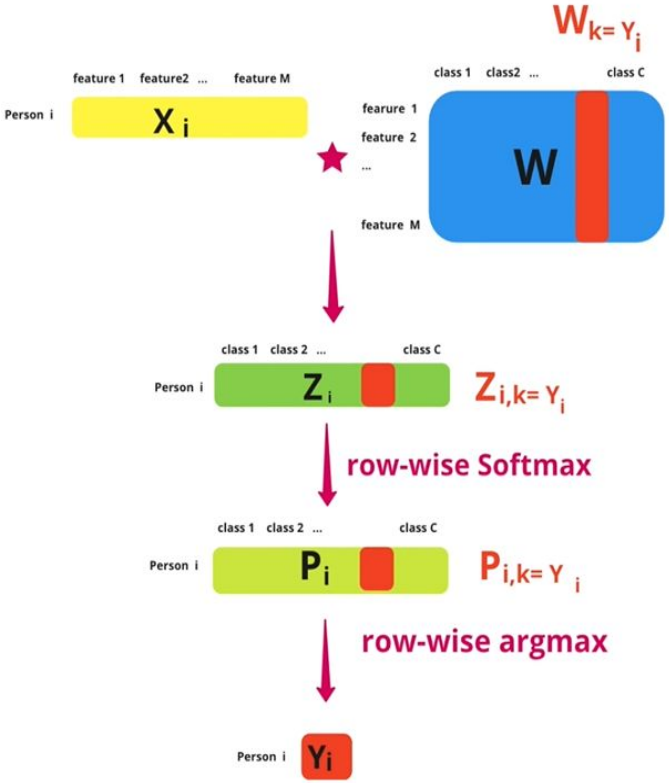
$$\text{if } y_i = 1: \quad P(x_i, 1|w) = \sigma_w(x_i) = \sigma_w(M_i)$$
$$\text{if } y_i = -1: \quad P(x_i, -1|w) = 1 - \sigma_w(x_i) = \sigma_w(-x_i) = \sigma_w(M_i)$$

$$\log L(w|X,Y) = \sum_{i=1}^{n} \log \sigma_w(M_i) = -\sum_{i=1}^{n} \log(1 + \exp(-M_i)) \to \max_w$$

$$L_{Logistic} = \log(1 + \exp(-M_i))$$



Multinomial logistic regression (softmax regression)



$$p(Y_i|X_i, W) = P_{i,k=Y_i} = \text{softmax}(Z_{i,k=Y_i}) = \frac{exp(Z_{i,k=Y_i})}{\sum_{k=0}^{C} exp(Z_{ik})} = \frac{\exp(-X_i W_{k=Y_i})}{\sum_{k=0}^{C} \exp(-X_i W_k)}$$

$$p(Y|X, W) = \prod_{i=1}^{N} \frac{\exp(-X_i W_{k=Y_i})}{\sum_{k=0}^{C} \exp(-X_i W_k)}$$

Loss

$$l(W)$$

$$= -\frac{1}{N}\log p(Y|X,W)$$

$$= \frac{1}{N}\left(\sum_{i=1}^{N}\left(X_i W_{k=Y_i} + \log\sum_{k=0}^{C}\exp(-X_i W_k)\right)\right)$$

$$= \frac{1}{N}\left(\sum_{i=1}^{N}\left(X_i W_{k=Y_i} + \sum_{i=1}^{N}\log\sum_{k=0}^{C}\exp(-X_i W_k)\right)\right)$$

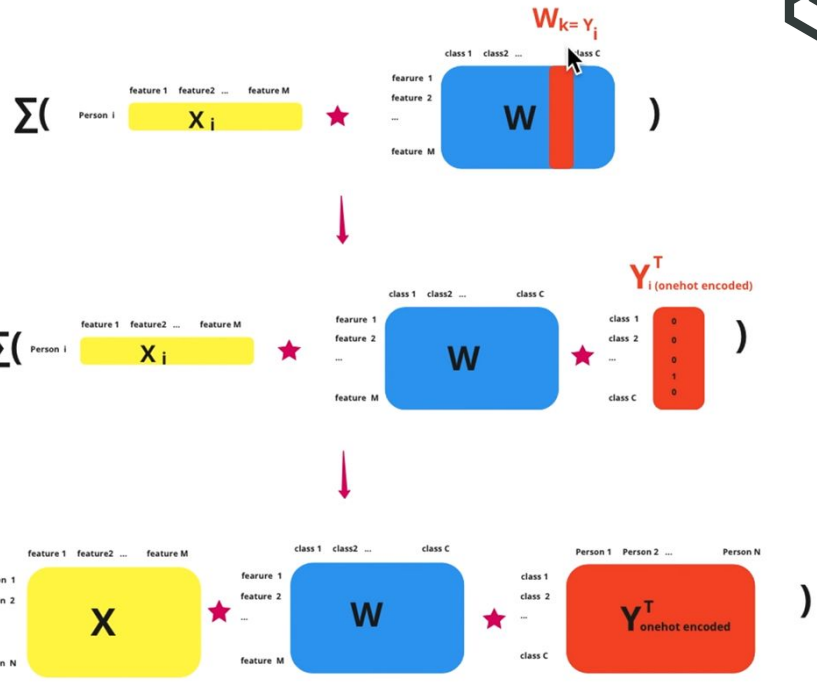
To write loss in matrix form:

$$l(W)$$

$$= \frac{1}{N}\left(\sum_{i=1}^{N}\left(X_i W Y^T_{i(onehot\_encoded)}\right) + \sum_{i=1}^{N}\log\sum_{k=0}^{C}\exp(-X_i W_k)\right)$$

$$= \frac{1}{N}\left(Tr(X W Y^T_{onehot\_encoded}) + \sum_{i=1}^{N}\log\sum_{k=0}^{C}\exp(-X_i W_k)\right)$$

$$= \frac{1}{N}\left(Tr(X W Y^T_{onehot\_encoded}) + \sum_{i=1}^{N}\log\sum_{k=0}^{C}\exp((-XW)_{ik})\right)$$



Loss gradient

$$f(W)$$

$$= \text{loss} + \text{regularization}$$

$$= \frac{1}{N} \sum_{i=1}^{N} (X_i W_{k=Y_i} + \log \sum_{k=0}^{C} \exp(-X_i W_k)) + \mu ||W||^2$$

$$\nabla_{W_k} f(W)$$

$$= \frac{1}{N} \sum_{i=1}^{N} (X_i^T I_{[Y_i=k]} - X_i^T \frac{\exp(-X_i W_k)}{\sum_{k=0}^{C} \exp(-X_i W_k)}) + 2\mu W$$

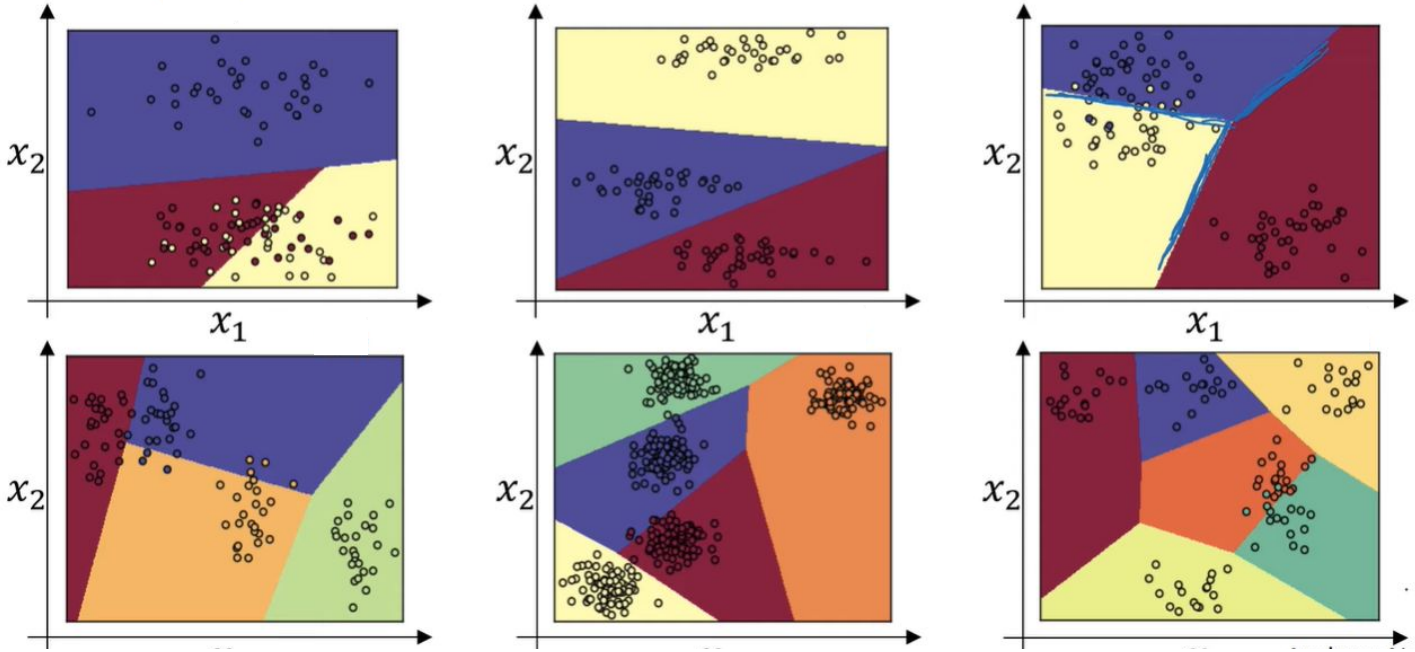
$$= \frac{1}{N} \sum_{i=1}^{N} (X_i^T I_{[Y_i=k]} - X_i^T P_i) + 2\mu W$$

$$= \frac{1}{N} (\sum_{i=1}^{N} X_i^T I_{[Y_i=k]} - \sum_{i=1}^{N} X_i^T P_i) + 2\mu W$$

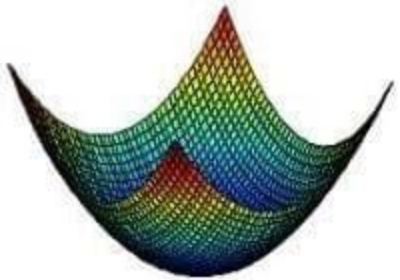
$$= \frac{1}{N} (X^T Y_{onehot\_encoded} - X^T P) + 2\mu W$$

$$= \frac{1}{N} (X^T (Y_{onehot\_encoded} - P)) + 2\mu W$$

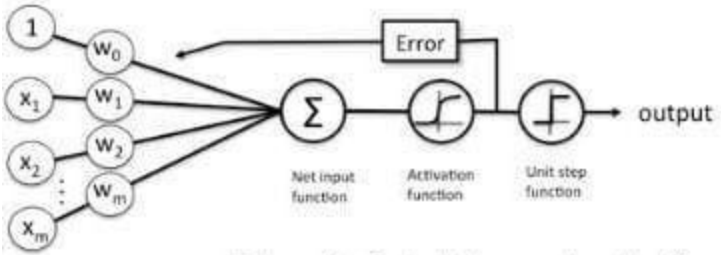
Examples



You

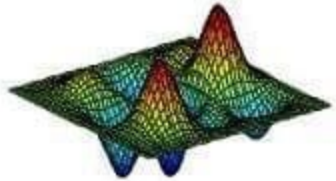


- Unique optimum: global/local.



Schematic of a logistic regression classifier.

The guy she tells you not to worry about



- Multiple local optima
- In high dimensions possibly

Deep neural network