Лекция 2: Введение в Нейронные сети. Deep Cross-Entropy Method

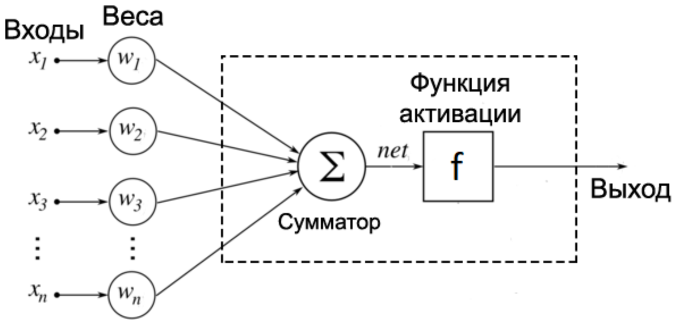
Антон Романович Плаксин

Организационные вопросы

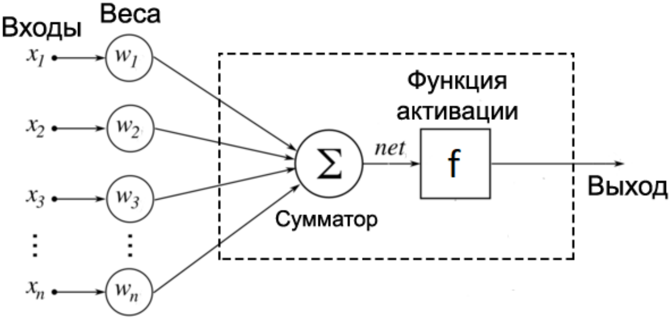
- Пятница, 17:50, аудитория 622
- Лекции и практики
- Отчетность: домашние работы
- Слайды: https://github.com/imm-rl-lab/UrFU_course
- E-mail для связи:
- Телефон (Telegram, Whatsapp) для связи:
- Вопросу по ходу можно и нужно!

Нейрон

Нейрон

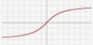

Входы
Веса
x_1
x_2
x_3
x_n
w_1
w_2
w_3
w_n
Функция активации
Σ
net
f
Сумматор
Выход

Нейрон

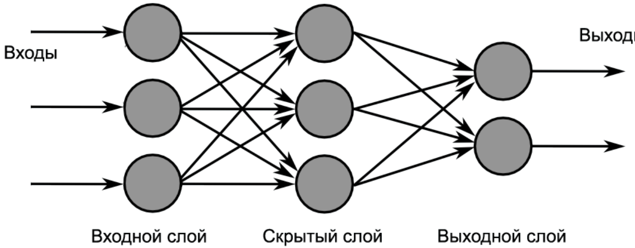


$$y = f\left(b + \sum_{i=1}^{n} w_i x_i\right)$$

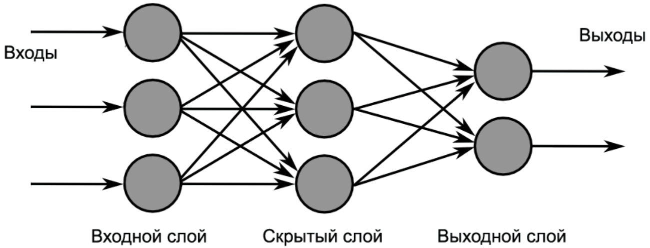
Функции активации

Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1+e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1+e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2+1}$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU) [2]		$f(x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Exponential Linear Unit (ELU) [3]		$f(x) = \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1 + e^x)$	$f'(x) = \frac{1}{1+e^{-x}}$

Нейронная есть



Нейронная есть

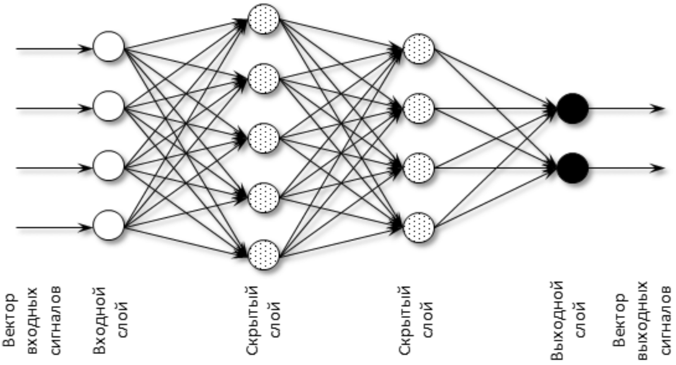


$$F_j^\theta(X) = f_{out}\left(b_j + \sum_{k=1}^{3} w_{j,k} f\left(\hat{b}_k + \sum_{i=1}^{3} \hat{w}_{k,i} x_i\right)\right), \quad j \in \overline{1,2}.$$

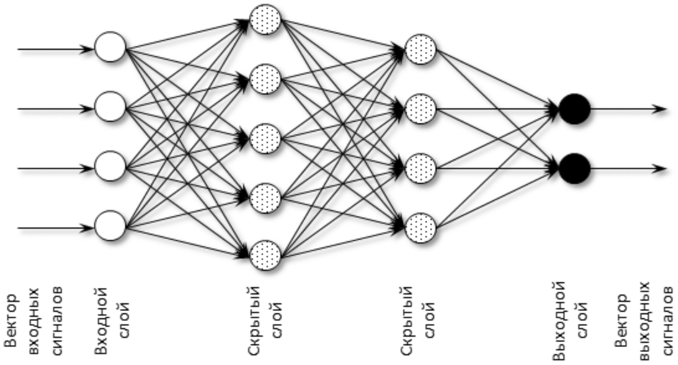
$$F^\theta(X) \in \mathbb{R}^2, \quad X = (x_1, x_2, x_3) \in \mathbb{R}^3,$$

$$\theta = (b_1, b_2, w_{1,1}, \dots, w_{2,3}, \hat{b}_1, \hat{b}_2, \hat{b}_3, \hat{w}_{1,1}, \dots, \hat{w}_{3,3}) \in \mathbb{R}^{20}$$

Нейронная есть



Нейронная есть



$$F_j^\theta(X) = f_{out}\left(b_j + \sum_{k=1}^{4} w_{j,k} f\left(\hat{b}_k + \sum_{l=1}^{5} \hat{w}_{k,l} f\left(\tilde{b}_l + \sum_{i=1}^{4} \tilde{w}_{l,i} x_i\right)\right)\right), \quad j \in \overline{1,2}.$$

$$F^\theta(X) \in \mathbb{R}^2, \quad X \in \mathbb{R}^4, \quad \theta \in \mathbb{R}^{59}$$

Задача регрессии

По данным $\{(X_1,Y_1),(X_2,Y_2),\dots,(X_k,Y_k)\}$, $X_i\in\mathbb{R}^n$, $Y_i\in\mathbb{R}^m$ решить задачу $\sum_{i=1}^{k}\|F(X_i)-Y_i\|^2\to\min\limits_{F\text{ is continuous}}$

Задача регрессии

По данным $\{(X_1, Y_1), (X_2, Y_2), \ldots, (X_k, Y_k)\}$, $X_i \in \mathbb{R}^n$, $Y_i \in \mathbb{R}^m$

решить задачу $\sum\limits_{i=1}^{k} \|F(X_i) - Y_i\|^2 \to \min\limits_{F \text{ is continuous}}$

Решение с использованием нейронной сети

- Зададим структуру F^θ и начальные веса θ_0 нейронной сети

Задача регрессии

По данным $\{(X_1, Y_1), (X_2, Y_2), \dots, (X_k, Y_k)\}$, $X_i \in \mathbb{R}^n$, $Y_i \in \mathbb{R}^m$ решить задачу $\sum\limits_{i=1}^{k} \|F(X_i) - Y_i\|^2 \to \min\limits_{F \text{ is continuous}}$

Решение с использованием нейронной сети

- Зададим структуру F^θ и начальные веса θ_0 нейронной сети
- Определим функцию ошибки $Loss(\theta) = \sum\limits_{i=1}^{k} \|F^\theta(X_i) - Y_i\|^2$

Задача регрессии

По данным $\{(X_1, Y_1), (X_2, Y_2), \ldots, (X_k, Y_k)\}$, $X_i \in \mathbb{R}^n$, $Y_i \in \mathbb{R}^m$ решить задачу $\sum_{i=1}^{k} \|F(X_i) - Y_i\|^2 \to \min_{F \text{ is continuous}}$

Решение с использованием нейронной сети

- Зададим структуру F^θ и начальные веса θ_0 нейронной сети
- Определим функцию ошибки $Loss(\theta) = \sum_{i=1}^{k} \|F^\theta(X_i) - Y_i\|^2$
- Решить задачу оптимизации $Loss(\theta) \mapsto \min_\theta$

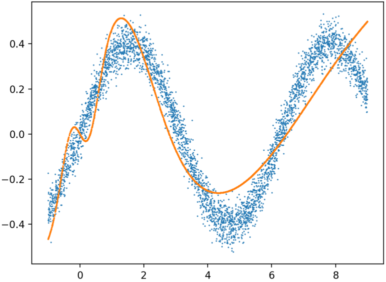
Задача регрессии

По данным $\{(X_1,Y_1),(X_2,Y_2),\dots,(X_k,Y_k)\}$, $X_i \in \mathbb{R}^n$, $Y_i \in \mathbb{R}^m$ решить задачу $\sum_{i=1}^{k} \|F(X_i) - Y_i\|^2 \to \min\limits_{F \text{ is continuous}}$

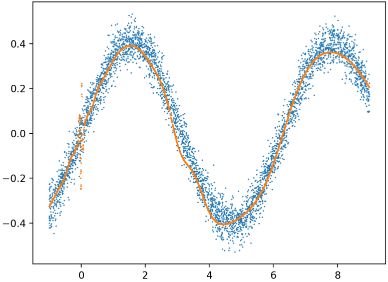
Решение с использованием нейронной сети

- Зададим структуру F^θ и начальные веса θ_0 нейронной сети
- Определим функцию ошибки $Loss(\theta) = \sum_{i=1}^{k} \|F^\theta(X_i) - Y_i\|^2$
- Решить задачу оптимизации $Loss(\theta) \mapsto \min\limits_{\theta}$
    - Метод градиентного спуска $\theta_{j+1} = \theta_j + \eta \nabla_\theta Loss(\theta)$, $\eta > 0$

Пример задачи регрессии



128 neurons



$32 \times 128 \times 32$ neurons

Теорема Цыбенко

Теорема

Для любой непрерывной функции $G\colon [0,1]^n \mapsto \mathbb{R}^m$ и любого $\varepsilon > 0$ найдутся такие N, w_i, $\hat{b}_i$, $\hat{w}_{i,j}$, $i \in \overline{1,N}$, $j \in \overline{1,n}$, что

$$\|F^\theta(X) - G(X)\| \le \varepsilon, \quad X \in [0,1]^n,$$

где

$$F^\theta(X) = \sum_{i=1}^{N} w_i \sigma\left(\hat{b}_i + \sum_{j=1}^{n} \hat{w}_{i,j} x_j\right),$$

$$X = (x_1, \ldots, x_n), \quad \theta = (w_1, \ldots w_N, \hat{b}_1, \ldots, \hat{b}_N, \hat{w}_{1,1}, \ldots, \hat{w}_{N,n})$$

Вычисление градиента

$$Loss(\theta) = \sum_{i=1}^{k} \|F^{\theta}(X_i) - Y_i\|^2,$$

$$F_j^{\theta}(X) = f_{out}\left(b_j + \sum_{k=1}^{3} w_{j,k} f\left(\hat{b}_k + \sum_{i=1}^{3} \hat{w}_{k,i} x_i\right)\right), \quad j \in \overline{1,2}.$$

Вычисление градиента

$$Loss(\theta) = \sum_{i=1}^{k} \|F^{\theta}(X_i) - Y_i\|^2,$$

$$F_j^{\theta}(X) = f_{out}\left(b_j + \sum_{k=1}^{3} w_{j,k} f\left(\hat{b}_k + \sum_{i=1}^{3} \hat{w}_{k,i} x_i\right)\right), \quad j \in \overline{1,2}.$$

$$\frac{\partial Loss(\theta)}{\partial \hat{w}_{2,3}} = ???$$

Вычисление градиента

$$Loss(\theta) = \sum_{i=1}^{k} \|F^{\theta}(X_i) - Y_i\|^2,$$

$$F_j^{\theta}(X) = f_{out}\left(b_j + \sum_{k=1}^{3} w_{j,k} f\left(\hat{b}_k + \sum_{i=1}^{3} \hat{w}_{k,i} x_i\right)\right), \quad j \in \overline{1,2}.$$

$$\frac{\partial Loss(\theta)}{\partial \hat{w}_{2,3}} = 2\sum_{i=1}^{k} \langle F^{\theta}(X_i) - Y_i, \frac{\partial F^{\theta}(X_i)}{\partial \hat{w}_{2,3}} \rangle$$

Вычисление градиента

$$\frac{\partial F_j^{\theta}(X_i)}{\partial \hat{w}_{2,3}} \approx \frac{F_j^{(\ldots,\hat{w}_{2,3}+\Delta w,\ldots)}(X_i) - F_j^{\theta}(X_i)}{\Delta w}$$

Вычисление градиента

$$\cancel{\frac{\partial F_j^{\theta}(X_i)}{\partial \hat{w}_{2,3}} \approx \frac{F_j^{(\dots,\hat{w}_{2,3}+\Delta w,\dots)}(X_i) - F_j^{\theta}(X_i)}{\Delta w}}$$

Вычисление градиента

$$\cancel{\frac{\partial F_j^{\theta}(X_i)}{\partial \hat{w}_{2,3}} \approx \frac{F_j^{(\ldots,\hat{w}_{2,3}+\Delta w,\ldots)}(X_i) - F_j^{\theta}(X_i)}{\Delta w}}$$

$$F_j^{\theta}(X) = f_{out}\left(b_j + \sum_{k=1}^{3} w_{j,k} f\left(\hat{b}_k + \sum_{i=1}^{3} \hat{w}_{k,i} x_i\right)\right), \quad j \in \overline{1,2}.$$

Вычисление градиента

$$\cancel{\frac{\partial F_j^{\theta}(X_i)}{\partial \hat{w}_{2,3}} \approx \frac{F_j^{(\dots,\hat{w}_{2,3}+\Delta w,\dots)}(X_i) - F_j^{\theta}(X_i)}{\Delta w}}$$

$$F_j^{\theta}(X) = f_{out}\left(b_j + \sum_{k=1}^{3} w_{j,k} f\left(\hat{b}_k + \sum_{i=1}^{3} \hat{w}_{k,i} x_i\right)\right), \quad j \in \overline{1,2}.$$

$$\frac{\partial F_j^{\theta}(X_i)}{\partial \hat{w}_{2,3}} = ???$$

Вычисление градиента

$$\cancel{\frac{\partial F_j^{\theta}(X_i)}{\partial \hat{w}_{2,3}} \approx \frac{F_j^{(\dots,\hat{w}_{2,3}+\Delta w,\dots)}(X_i) - F_j^{\theta}(X_i)}{\Delta w}}$$

$$F_j^{\theta}(X) = f_{out}\left(b_j + \sum_{k=1}^{3} w_{j,k} f\left(\hat{b}_k + \sum_{i=1}^{3} \hat{w}_{k,i} x_i\right)\right), \quad j \in \overline{1,2}.$$

$$\begin{aligned} \frac{\partial F_j^{\theta}(X_i)}{\partial \hat{w}_{2,3}} &= f'_{out}\left(b_j + \sum_{k=1}^{3} w_{j,k} f\left(\hat{b}_k + \sum_{i=1}^{3} \hat{w}_{k,i} x_i\right)\right) \\ &\times w_{j,2} f'\left(\hat{b}_2 + \sum_{i=1}^{3} \hat{w}_{2,i} x_i\right) x_3 \end{aligned}$$

Статистический градиентный спуск

Статистический градиентный спуск

Пусть $l \in \mathbb{N}$ — размер батча

Статистический градиентный спуск

Пусть $l \in \mathbb{N}$ — размер батча

- Случайным образом выбираем из данных батч $\{(X_1, Y_1), \ldots, (X_l, Y_l)\}$

Статистический градиентный спуск

Пусть $l \in \mathbb{N}$ — размер батча

- Случайным образом выбираем из данных батч $\{(X_1, Y_1), \ldots, (X_l, Y_l)\}$
- Определим функцию ошибки $Loss(\theta) = \sum\limits_{i=1}^{l} \|F^\theta(X_i) - Y_i\|^2$

Статистический градиентный спуск

Пусть $l \in \mathbb{N}$ — размер батча

- Случайным образом выбираем из данных батч $\{(X_1, Y_1), \ldots, (X_l, Y_l)\}$
- Определим функцию ошибки $Loss(\theta) = \sum_{i=1}^{l} \|F^{\theta}(X_i) - Y_i\|^2$
- Обновляем веса $\theta_{j+1} = \theta_j + \eta \nabla_{\theta} Loss(\theta)$

Задача классификации

По данным $\big\{(X_1, Y_1), \ldots, (X_k, Y_k)\big\}$, $X_i \in \mathbb{R}^n$, $Y_i \in \{1, \ldots, m\}$, найти такое отображение F, что $|\{i \in \overline{1, m} \mid F(X_i) \neq Y_i\}| \to \min\limits_{F}$.

Задача классификации

По данным $\{(X_1, Y_1), \ldots, (X_k, Y_k)\}$, $X_i \in \mathbb{R}^n$, $Y_i \in \{1, \ldots, m\}$, найти такое отображение F, что $|\{i \in \overline{1,m} \mid F(X_i) \neq Y_i\}| \to \min\limits_F$.

$$\operatorname{Softmax}(Z)_i = \frac{e^{z_i}}{\sum\limits_{j=1}^{m} e^{z_j}}, \quad i \in \overline{1,m}, \quad Z = (z_1, \ldots, z_m)$$

Задача классификации

По данным $\{(X_1, Y_1), \ldots, (X_k, Y_k)\}$, $X_i \in \mathbb{R}^n$, $Y_i \in \{1, \ldots, m\}$, найти такое отображение F, что $|\{i \in \overline{1,m} \mid F(X_i) \neq Y_i\}| \to \min\limits_F$.

$$\mathrm{Softmax}(Z)_i = \frac{e^{z_i}}{\sum\limits_{j=1}^{m} e^{z_j}}, \quad i \in \overline{1,m}, \quad Z = (z_1, \ldots, z_m)$$

Решение с использованием нейронной сети

- Зададим структуру нейронной сети F^θ и начальные веса θ_0

Задача классификации

По данным $\{(X_1, Y_1), \dots, (X_k, Y_k)\}$, $X_i \in \mathbb{R}^n$, $Y_i \in \{1, \dots, m\}$, найти такое отображение F, что $|\{i \in \overline{1,m} \mid F(X_i) \neq Y_i\}| \to \min\limits_F$.

$$\mathrm{Softmax}(Z)_i = \frac{e^{z_i}}{\sum\limits_{j=1}^{m} e^{z_j}}, \quad i \in \overline{1,m}, \quad Z = (z_1, \dots, z_m)$$

Решение с использованием нейронной сети

- Зададим структуру нейронной сети F^θ и начальные веса θ_0
- Определим функцию ошибки

$$Loss(\theta) = -\sum_{i=1}^{k} \ln \mathrm{Softmax}(F^\theta(X_i))_{Y_i}$$

Задача классификации

По данным $\{(X_1, Y_1), \ldots, (X_k, Y_k)\}$, $X_i \in \mathbb{R}^n$, $Y_i \in \{1, \ldots, m\}$, найти такое отображение F, что $|\{i \in \overline{1,m} \mid F(X_i) \neq Y_i\}| \to \min\limits_F$.

$$\mathrm{Softmax}(Z)_i = \frac{e^{z_i}}{\sum\limits_{j=1}^{m} e^{z_j}}, \quad i \in \overline{1,m}, \quad Z = (z_1, \ldots, z_m)$$

Решение с использованием нейронной сети

- Зададим структуру нейронной сети F^θ и начальные веса θ_0
- Определим функцию ошибки

  $Loss(\theta) = -\sum\limits_{i=1}^{k} \ln \mathrm{Softmax}(F^\theta(X_i))_{Y_i}$
- Решить задачу оптимизации $Loss(\theta) \mapsto \min\limits_\theta$

Задача классификации

По данным $\big\{(X_1, Y_1), \ldots, (X_k, Y_k)\big\}$, $X_i \in \mathbb{R}^n$, $Y_i \in \{1, \ldots, m\}$, найти такое отображение F, что $|\{i \in \overline{1,m} \,|\, F(X_i) \neq Y_i\}| \to \min\limits_F$.

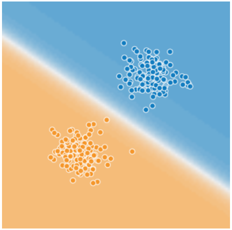
$$\mathrm{Softmax}(Z)_i = \frac{e^{z_i}}{\sum\limits_{j=1}^{m} e^{z_j}}, \quad i \in \overline{1,m}, \quad Z = (z_1, \ldots, z_m)$$

Решение с использованием нейронной сети

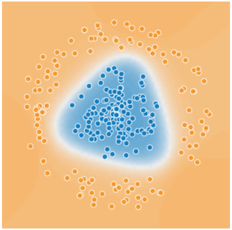
- Зададим структуру нейронной сети F^θ и начальные веса θ_0
- Определим функцию ошибки

  $$Loss(\theta) = -\sum_{i=1}^{k} \ln \mathrm{Softmax}(F^\theta(X_i))_{Y_i}$$
- Решить задачу оптимизации $Loss(\theta) \mapsto \min\limits_\theta$
  - Метод градиентного спуска $\theta_{j+1} = \theta_j + \eta \nabla_\theta Loss(\theta)$

Пример задачи классификации

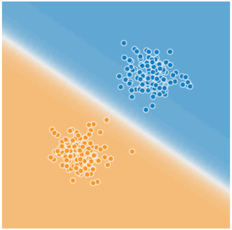


1 layer $\times$ 1 neuron

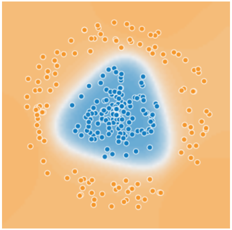


1 layer $\times$ 3 neuron

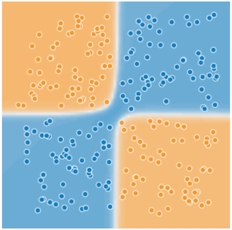
Пример задачи классификации



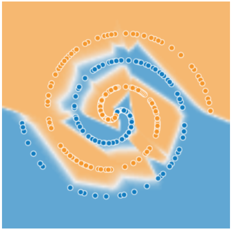
1 layer $\times$ 1 neuron



1 layer $\times$ 3 neuron

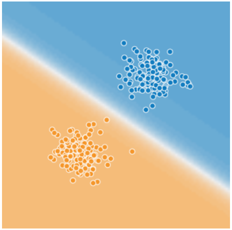


2 layer $\times$ 3 neuron

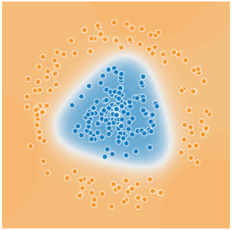


4 layer $\times$ 8 neuron

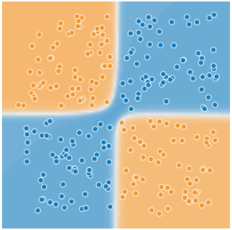
Пример задачи классификации



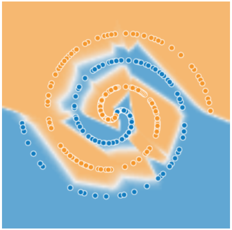
1 layer $\times$ 1 neuron



1 layer $\times$ 3 neuron



2 layer $\times$ 3 neuron



4 layer $\times$ 8 neuron

https://playground.tensorflow.org

Элементы работы нейронной сети

Элементы работы нейронной сети

- Структура (кол-во слоев и нейронов, функции активации)

Элементы работы нейронной сети

- Структура (кол-во слоев и нейронов, функции активации)
- Функция Loss

Элементы работы нейронной сети

- Структура (кол-во слоев и нейронов, функции активации)
- Функция Loss
- Метод градиентного спуска

Элементы работы нейронной сети

- Структура (кол-во слоев и нейронов, функции активации)
- Функция Loss
- Метод градиентного спуска
- Инициализация весов

Элементы работы нейронной сети

- Структура (кол-во слоев и нейронов, функции активации)
- Функция Loss
- Метод градиентного спуска
- Инициализация весов
- Регуляризация, Batch normalization, Dropout, ...

Элементы работы нейронной сети

- Структура (кол-во слоев и нейронов, функции активации)
- Функция Loss
- Метод градиентного спуска
- Инициализация весов
- Регуляризация, Batch normalization, Dropout, ...

С. Николенко, Е. Архангельская «Глубокое обучение. Погружение в мир нейронных сетей»

Cross-Entropy Method. Случай $\mathcal{S} \subset \mathbb{R}^n$ и $\mathcal{A} \subset \mathbb{R}^m$

Пусть $\pi^{\theta_0} \colon \mathbb{R}^n \mapsto \mathbb{R}^m$ — нейронная сеть с начальными весами θ_0,
N — количество итераций алгоритма,
$p \in [0, 1]$ — порог для элитных траекторий,
$\eta, \varepsilon > 0$ — параметры (можно уменьшать в процессе работы).
Для каждого $n \in \overline{0, N}$ делаем

Cross-Entropy Method. Случай $\mathcal{S} \subset \mathbb{R}^n$ и $\mathcal{A} \subset \mathbb{R}^m$

Пусть $\pi^{\theta_0} : \mathbb{R}^n \mapsto \mathbb{R}^m$ — нейронная сеть с начальными весами θ_0,
N — количество итераций алгоритма,
$p \in [0, 1]$ — порог для элитных траекторий,
$\eta, \varepsilon > 0$ — параметры (можно уменьшать в процессе работы).
Для каждого $n \in \overline{0, N}$ делаем

- (Policy evaluation) Действуя по правилу

$$A_t = \left[\pi^{\theta_n}(S_t) + Noise(\varepsilon)\right]_{\mathcal{A}}, \quad t \in \overline{0, T}$$

реализуем K сессий, получаем траектории τ_k, $k \in \overline{1, K}$ и награды $G(\tau_k)$ для каждой из них. Оцениваем policy π^{θ}:

$$\mathbb{E}_{\pi^{\theta_n}}[G] \approx V_{\pi^{\theta_n}} := \frac{1}{K} \sum_{k=1}^{K} G(\tau_k)$$

Если $V_{\pi^{\theta_n}} << V_{\pi^{\theta_{n-1}}}$, то break с ответом $\pi^{\theta_{n-1}}$

Cross-Entropy Method. Случай $\mathcal{S} \subset \mathbb{R}^n$ и $\mathcal{A} \subset \mathbb{R}^m$

Пусть $\pi^{\theta_0} : \mathbb{R}^n \mapsto \mathbb{R}^m$ — нейронная сеть с начальными весами θ_0,
N — количество итераций алгоритма,
$p \in [0, 1]$ — порог для элитных траекторий,
$\eta, \varepsilon > 0$ — параметры (можно уменьшать в процессе работы).
Для каждого $n \in \overline{0, N}$ делаем

- (Policy improvement) По значениям $G(\tau_k)$ выбираем $L = (1-p)K$ элитных траекторий $\mathcal{T}_n = \{\tau_1, \tau_2, \dots, \tau_L\}$, по ним определяем функцию потерь

$$Loss(\theta) = \sum_{(a|s) \in \mathcal{T}_n} \|\pi^{\theta_n}(s) - a\|^2$$

и методом градиентного спуска определяем следующие веса

$$\theta_{n+1} = \theta_n + \eta \nabla_\theta Loss(\theta_n)$$

Cross-Entropy Method. Случай $\mathcal{S} \subset \mathbb{R}^n$ и $|\mathcal{A}| = m$

Пусть $F^\theta \colon \mathbb{R}^n \mapsto \mathbb{R}^m$ — нейронная сеть. Обозначаем

$$\pi^\theta(i|s) = \mathrm{Softmax}(F^\theta(s))_i, \quad \pi_{\text{uniform}}(i|s) = 1/m, \quad i \in \overline{1,m}.$$

Пусть θ_0 — начальные веса, N — количество итераций алгоритма, $p \in [0,1]$ — порог для элитных траекторий, $\eta, \varepsilon > 0$ — параметры (можно уменьшать в процессе работы). Для каждого $n \in \overline{0,N}$ делаем

Cross-Entropy Method. Случай $\mathcal{S} \subset \mathbb{R}^n$ и $|\mathcal{A}| = m$

Пусть $F^\theta \colon \mathbb{R}^n \mapsto \mathbb{R}^m$ — нейронная сеть. Обозначаем

$$\pi^\theta(i|s) = \mathrm{Softmax}(F^\theta(s))_i, \quad \pi_{\text{uniform}}(i|s) = 1/m, \quad i \in \overline{1, m}.$$

Пусть θ_0 — начальные веса, N — количество итераций алгоритма, $p \in [0, 1]$ — порог для элитных траекторий, $\eta, \varepsilon > 0$ — параметры (можно уменьшать в процессе работы). Для каждого $n \in \overline{0, N}$ делаем

- (Policy evaluation) Действуя по правилу

  $$A_t \sim (1 - \varepsilon)\pi^{\theta_n}(\cdot|S_t) + \varepsilon\pi_{\text{uniform}}(\cdot|S_t), \quad t \in \overline{0, T},$$

  реализуем K сессий, получаем траектории τ_k, $k \in \overline{1, K}$ и награды $G(\tau_k)$ для каждой из них. Оцениваем policy π^θ:

  $$\mathbb{E}_{\pi^{\theta_n}}[G] \approx V_{\pi^{\theta_n}} := \frac{1}{K}\sum_{k=1}^{K} G(\tau_k)$$

  Если $V_{\pi^{\theta_n}} << V_{\pi^{\theta_{n-1}}}$, то break с ответом $\pi^{\theta_{n-1}}$

Cross-Entropy Method. Случай $\mathcal{S} \subset \mathbb{R}^n$ и $|\mathcal{A}| = m$

Пусть $F^\theta : \mathbb{R}^n \mapsto \mathbb{R}^m$ — нейронная сеть. Обозначаем

$$\pi^\theta(i|s) = \text{Softmax}(F^\theta(s))_i, \quad \pi_{\text{uniform}}(i|s) = 1/m, \quad i \in \overline{1, m}.$$

Пусть θ_0 — начальные веса, N — количество итераций алгоритма, $p \in [0, 1]$ — порог для элитных траекторий, $\eta, \varepsilon > 0$ — параметры (можно уменьшать в процессе работы). Для каждого $n \in \overline{0, N}$ делаем

- (Policy improvement) По значениям $G(\tau_k)$ выбираем $L = (1-p)K$ элитных траекторий $\mathcal{T}_n = \{\tau_1, \tau_2, \dots, \tau_L\}$, по ним определяем функцию потерь

$$Loss(\theta) = -\sum_{(a|s) \in \mathcal{T}_n} \ln \pi^\theta(a|s)$$

и методом градиентного спуска определяем следующие веса

$$\theta_{n+1} = \theta_n + \eta \nabla_\theta Loss(\theta_n)$$

ВОПРОСЫ?