

Обучение глубинных нейронных сетей на основе методов байесовской фильтрации

Сергей Павлов

Институт проблем передачи информации РАН
Московский физико-технический институт

2017

- 1 Постановка задачи
- 2 Интерпретация
- 3 Решение задачи
- 4 Эксперименты

Постановка задачи

- Измеримое пространство (Ω, \mathcal{F}) с фильтрацией $\mathcal{F} := (\mathcal{F}_r)_{r=0,1,2,\dots,T}$, $T \in N_+$
- Случайный процесс (управляющий процесс) \mathbf{a} : $\Omega \times \{0, \dots, T-1\} \rightarrow A$, где (A, \mathcal{B}) - измеримое пространство
- $\mathbf{a} = (a_0, a_1, \dots, a_{T-1}) \in \mathcal{A}$ - допустимое множество значений управлений; задавшись таким значением процесса \mathbf{a} , определим
- Марковский процесс $X_r : (\Omega, \mathcal{F}, P^{\mathbf{a}}) \rightarrow (S, \mathcal{S})$, $X_0 = x_0$ п.н., с

$$P^{\mathbf{a}}(X_{r+1} \in dy | X_r = x) = P^{ar}(x, dy), \quad 0 \leq r < T.$$

Пусть $\mathcal{T} \subset \{0, \dots, T\}$. Зададимся измеримыми функциями

$$f_r : S \times A \rightarrow \mathbb{R}$$

$$g_r : S \rightarrow \mathbb{R}$$

Рассмотрим оптимизационную задачу:

$$Y_0^* := \sup_{\mathbf{a} \in \mathcal{A}, \tau \in \mathcal{T}} E^{\mathbf{a}} \left[\sum_{r=0}^{\tau-1} f_r(X_r, a_r) + g_\tau(X_\tau) \right].$$

Интерпретируем данную общую задачу в терминах обучения нейронных сетей.

Рассмотрим нейронную сеть, состоящую из T слоёв с номерами $r = 0, \dots, T - 1$. $\mathcal{T} = \{T\}$ При этом:

- X_r отвечает за значение нейронов слоя r ;
- a_r соответствует паре $(\mathbf{W}^r, \mathbf{B}^r)$ - матрица переходов между слоями r и $r + 1$, и вектор порогов активаций слоя $r + 1$;
- Значения процессов \mathbf{W} и \mathbf{B} имеют независимые нормальные распределения:

$$\mathbf{B}^i \sim \mathcal{N}(\mathbf{b}^i, \mathbb{I}),$$

$$\mathbf{W}^i \sim \mathcal{N}(\mathbf{w}^i, \mathbb{I})$$

- Обозначим функцию потерь сети $L'(X_{T-1}, Y)$;
- $g_T(X_{T-1}) := -L'(X_{T-1}, Y) = L(X_{T-1}, Y)$
- $f_r = 0, r = 0, \dots, T - 1$

Оптимизационная задача переформулируется таким образом:

$$Y_0^* := \sup_{\mathbf{a} \in \mathcal{A}} E^{\mathbf{a}} [L(X_{T-1}, Y)]$$

Напомним, что здесь L есть функция потерь сети, взятая с обратным знаком ($L < 0$).

Обозначим \mathcal{A}_r допустимое множество управлений \mathbf{a} :

$$\Omega \times \{r, \dots, T-1\} \rightarrow \mathcal{A}$$

Определим случайные величины:

$$Y_r^* := \sup_{\mathbf{a} \in \mathcal{A}_r} E^{\mathbf{a}} [L(X_{T-1}, Y) | X_r], \quad 0 \leq r \leq T.$$

Зададим функции:

$$h_r^*(x) = \sup_{\mathbf{a} \in \mathcal{A}_r} E^{\mathbf{a}} [L(X_{T-1}, Y) | X_r = x], \quad 0 \leq r \leq T$$

Решение задачи

В соответствии с принципом Беллмана:

$$Y_r^* = h_r^*(X_r).$$

Если вычислена зависимость $h_{r+1}^*(x')$, $\forall x'$, то задача поиска зависимости $h_r^*(x)$ сводится к оптимизации по единственному параметру a_r :

$$h_r^*(x) = \sup_{a_r} \int P^{a_r}(x, dy) h_{r+1}^*(y)$$

Алгоритм обучения для фиксированного Y

- Значения X_0 и $Y_0(X_0)$ берутся из тренировочного набора данных;
- На шаге $i \in [1, \dots, T-1]$ оптимизационная задача решается перебором по следующей схеме:

- Сгенерировать $\{X_{T-i-1}^j\}_{j=0}^N$ из гиперкуба $[0, 1]^{dim(X_{T-i-1})}$;
- Для каждого из X_{T-i-1}^j генерировать по M значений параметров $\mathbf{w}_{jk}^{T-i-1}, \mathbf{b}_{jk}^{T-i-1}, k \in \{1, \dots, M\}$ из гиперкуба с базой $[-2, 2]$;
- Для каждой пары параметров $\mathbf{w}_{jk}^{T-i-1}, \mathbf{b}_{jk}^{T-i-1}$ получить K реализаций случайных величин: $B_{jkl}^{T-i-1}, W_{jkl}^{T-i-1}, l = 1, \dots, K$
-

$$h_{T-i-1,jk}^{*,emp}(X_{T-i-1}^j) := \frac{1}{K} \sum_{l=1}^K h_{T-i}^*(\sigma(W_{jkl}^{T-i-1})X_{T-i-1}^j + B_{jkl}^{T-i-1})$$

- Выбрать пару параметров $(w_j^{*,T-i-1}, b_j^{*,T-i-1}) = (w_{jk^*}^{T-i-1}, b_{jk^*}^{T-i-1}),$
 $k^* := \operatorname{argmax}_k h_{T-i-1,jk}^{*,emp}(X_{T-i-1}^j);$

- Построить регрессионную модель M_{T-i-1}^Y зависимости $(w_j^{*, T-i-1}, b_j^{*, T-i-1})$ от значения X_{T-i-1}^j ;

Результат обучения: $M^Y = (M_0^Y, \dots, M_{T-2}^Y)$

Вычисление прогноза

Имея обучение $M^Y = (M_0^Y, \dots, M_{T-2}^Y)$ для **фиксированного** Y , опишем процесс построение прогноза сети.

При вычислении прогноза будем полагать случайные величины $\mathbf{W}^i, \mathbf{B}^i$ равными своим мат.ожиданиям $\mathbf{w}^i(\mathbf{X}_{i-1}), \mathbf{b}^i(\mathbf{X}_{i-1})$;

Чтобы сделать прогноз по заданному входу X_0 необходимо произвести цепочку вычислений:

$$\begin{aligned} \overline{X}_0 &:= X_0 \xrightarrow{M_0^Y} \mathbf{w}^0, \mathbf{b}^0 \rightarrow \overline{X}_1 := \sigma(\overline{X}_0 \mathbf{w}^0 + \mathbf{b}^0) \\ &\xrightarrow{M_1^Y} \mathbf{w}^1, \mathbf{b}^1 \rightarrow \overline{X}_2 \dots, \overline{X}_{T-1} \end{aligned}$$

В зависимости от задачи (регрессия/классификация) прогнозом объявляется \overline{X}_{T-1} или $\sigma(\overline{X}_{T-1})$.

Объединение обучений для различных Y

Перейдём к задаче бинарной классификации(0/1).

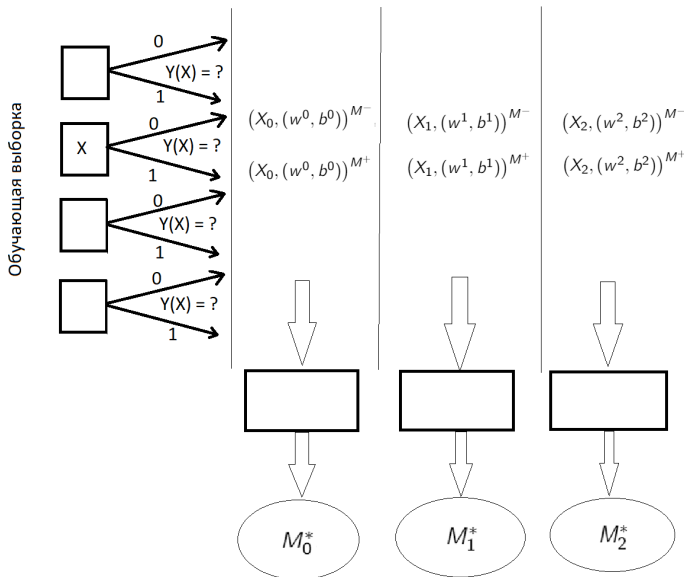
$\{(X_r, Y(X_r))\}_{r=1}^Q$ - обучающая выборка; $Y(X_r) \in \{0, 1\}$

Обучение для 1 : $M^+ = (M_0^+, \dots, M_{T-2}^+)$

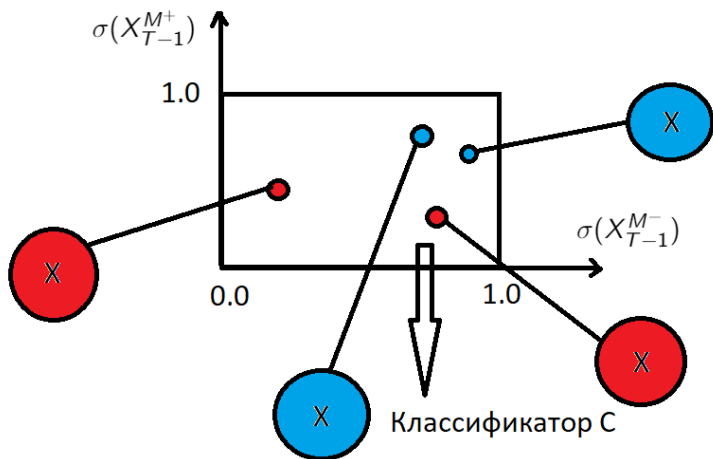
Обучение для 0 : $M^- = (M_0^-, \dots, M_{T-2}^-)$

Цель: Получить обучение $M^* = (M_0^*, \dots, M_{T-2}^*)$, отражающее зависимость $Y(X_{in})$

Подход 1



Подход 2



Эксперименты

$N = 15$, $M = 15$, $K = 5$, архитектура сети: $[X, 5, 5, 5, 5, 5, 1]$.

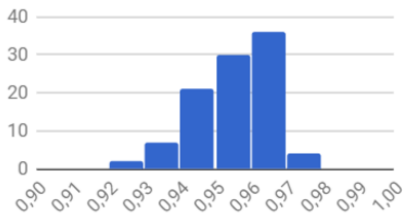
Данные:

- ① "diabetes"
- ② "breast-cancer"
- ③ "fourclass"
- ④ "german.numer"
- ⑤ "heart"
- ⑥ "liver-disorders"

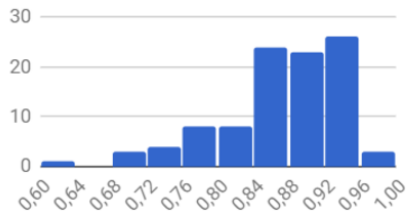
Результаты

Набор данных	Среднее(o/c)	Дисперсия(o/c)	Среднее(#1)	Дисперсия(#1)	Среднее(#2)	Дисперсия(#2)	Размер обуч. выб	Размер тест. выб
1	0,72	0,0006	0,69	0,003	0,54	0,014	400	368
2	0,96	0,0001	0,87	0,004	0,7	0,04	350	333
3	0,93	0,005	0,69	0,004	0,58	0,02	550	312
4	0,73	0,006	0,68	0,009	0,51	0,02	900	100
5	0,78	0,003	0,67	0,01	0,52	0,02	170	100
6	0,68	0,004	0,63	0,01	0,58	0,02	100	45

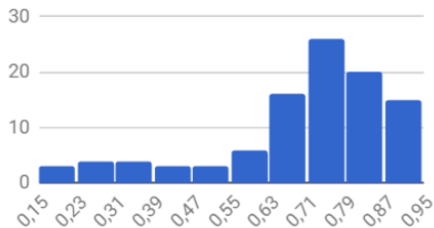
Обычная нейросеть

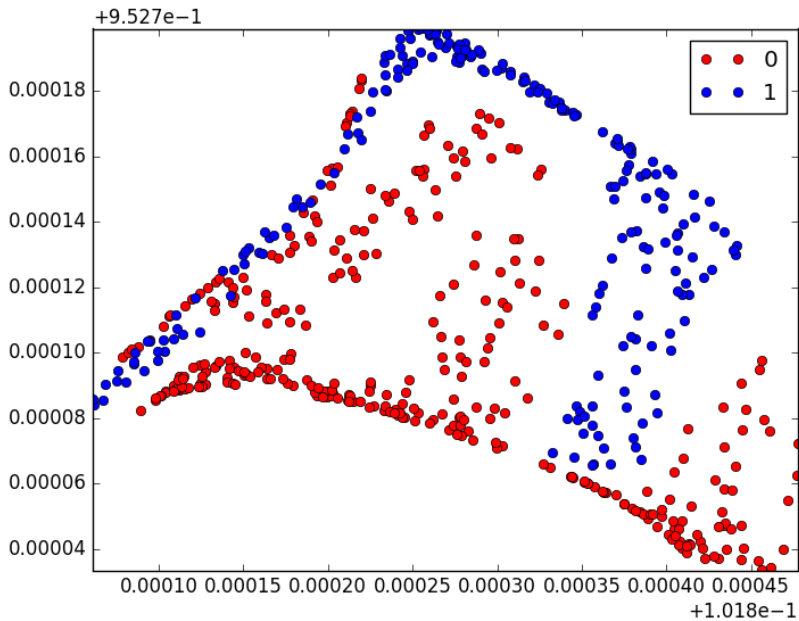


Подход #1



Подход #2





Спасибо за внимание!!

