Школа анализа данных
Машинное обучение, часть 1
Домашнее задание №1

Решите предложенные задачи. Решения необходимо оформить в виде PDF документа. Каждая задача должна быть подробно обоснована, задачи без обоснования не засчитываются. Решения пишутся в свободной форме, однако так, чтобы проверяющие смогли разобраться. Если проверяющие не смогут разобраться в решении какой-нибудь задачи, то она автоматически не засчитывается.

Задача 1 (0.5 балла) Кроссвалидация, LOO, k-fold.

Объясните, стоит ли использовать оценку leave-one-out-CV или k-fold-CV с небольшим k в случае, когда:

- обучающая выборка содержит очень малое количество объектов;
- обучающая выборка содержит очень большое количество объектов.

Решение:

Распишем формулы для оценок leave-one-out-CV (LOO) и k-fold-CV (CV). Пусть рассматривается выборка объектов $X = \{x_1, \dots, x_L\}$. Тогда оценка leave-one-out-CV рассчитывается по следующей формуле:

$$LOO(a, X^L) = \frac{1}{L} \sum_{i=1}^{L} \mathcal{L}(a(X^L \backslash \{x_i\}, x_i))$$

Для подсчета оценки k-fold-CV нужно разбить выборку на k непересекающихся блоков одинаковой (или почти одинаковой) длины $l_1, \dots, l_k : X^L = X_1^{l_1} \sqcup X_2^{l_2} \sqcup \dots \sqcup X_k^{l_k}$, где $l_1 + l_2 + \dots + l_k = L$. Оценка k-fold-CV выглядит следующим образом:

$$CV(a, X^L) = \frac{1}{k} \sum_{i=1}^{k} \mathcal{L}(a(X^L \backslash X_i^{k_i}, X_i^{k_i}))$$

1. обучающая выборка содержит очень малое количество объектов

   В данном случае следует использовать оценку leave-one-out-CV. Так как размер выборки очень маленький, то можно быстро обучить модель L раз и подсчитать значение. Достоинтсвом этой оценки является то, что каждый объект выборки один раз участвует в контроле.

   С подсчетом k-fold-CV могут возникнуть проблемы, если k > L. Если k=L, то оценка превращается в оценку контроля по отдельным объектам (leave-one-out-CV).

2. обучающая выборка содержит очень большое количество объектов

   Так как выборка содержит очень большое количество объектов, то подсчёт оценки LOO достаточно ресурсоёмкий. Лучше всего в этом случае воспользоваться оценкой k-fold-CV, потому что приходится обучать модель небольшое количество k раз.

Ответ: Для выборки с малым количеством элементов лучше всего использовать оценку leave-one-out-CV, с большим количеством-k-fold-CV.

Задача 2 (1.5 балла). Линейная регрессия, решение с наименьшей нормой.

Рассмотрим задачу обучения линейной регрессии

$$\|Xw - y\|^2 \to \min_w . \tag{1}$$

Практика показывает, что для избежания переобучения можно использовать регуляризацию, которая дополнительно штрафует евклидову норму вектора весов $\|w\|$ в задаче оптимизации (1).

Положим, что система уравнений $Xw = y$ является *неопределенной*[1], т. е. существует бесконечно много w, которые доставляют точное решение этой системы. Кроме того, будем считать, что матрица X является прямоугольной матрицей полного ранга. Интуиция регуляризации, описанная выше, говорит нам о том, что среди всех точных решений необходимо выбрать такое решение w^*, которое будет иметь наименьшую норму. Докажите, что решением с наименьшей нормой является:

$$w^* = X^{\dagger}_{\text{right}} y, \quad X^{\dagger}_{\text{right}} = X^T (XX^T)^{-1}.$$

Матрица $X^{\dagger}_{\text{right}}$ называется правой псевдообратной. Считайте, что X — прямоугольная матрица полного ранга.

Решение:

Составим систему уравнений:

$$\begin{cases} Xw = y \\ ||w||^2 \to min_w \end{cases}$$

Применим метод множителей Лагранжа для данной системы:

$$L = ||w||^2 + \lambda^T (Xw - y) = 0,$$

где λ - вектор размера $n \times 1$, где n - число строк вектора y.

$$\begin{cases} \frac{\partial L}{\partial w} = 2w + X^T \lambda = 0 \\ Xw = y \end{cases}$$

$$\begin{cases} 2w = -X^T \lambda, \text{ домножим слева и справа на X} \\ Xw = y \end{cases}$$

$$\begin{cases} 2Xw = -XX^T \lambda \\ Xw = y \end{cases}$$

Из данной системы получаем равенство $2y = -XX^T \lambda$. Матрица X является прямоугольной матрицей полного ранга и неопределенной (underdetermined system), поэтому её ранг равен числу строк, следовательно XX^T можно обратить: $2(XX^T)^{-1} y = -\lambda$.

$$2X^T (XX^T)^{-1} y = -X^T \lambda = 2w => w = X^T (XX^T)^{-1} y$$

ч.т.д.

[1] https://en.wikipedia.org/wiki/Underdetermined_system

Задача 3 (1 балл). Линейная регрессия, точное решение.

Рассмотрим задачу обучения линейной регрессии с регуляризацией

$$\|(Xw + b \cdot \vec{1}) - y\|^2 + \lambda \|w\|^2 \to \min_{w,b} \tag{2}$$

где $\vec{1}$ - вектор-столбец, состоящий из единиц, а b - скаляр. Найдите точное решение для w и b.
Решение:

$$\begin{cases} \nabla_w L = \nabla_w \Big((Xw + b \cdot \vec{1})^T (Xw + b \cdot \vec{1}) + \lambda w^T w \Big) = 2X^T X w + 2X^T b \cdot \vec{1} - 2X^T y + 2\lambda w = 0 \\ \nabla_b L = \nabla_b \Big((Xw + b \cdot \vec{1})^T (Xw + b \cdot \vec{1}) + \lambda w^T w \Big) = w^T X^T \vec{1} + \vec{1}^T X w + 2\vec{1}^T \vec{1} b - \vec{1}^T y - y^T \vec{1} = 0 \end{cases}$$

Решаем 1 уравнение:

$$2X^T X w + 2X^T b \cdot \vec{1} - 2X^T y + 2\lambda w = 0$$
$$(X^T X + \lambda I) w = X^T (y - b\vec{1})$$
$$(X^T X + \lambda I) w = X^T (y - b\vec{1})$$
$$w = (X^T X + \lambda I)^{-1} X^T (y - b\vec{1})$$

Матрицу $X^T X + \lambda I$ можно обратить, потому что мы увеличиваем все собственные значения на λ, отодвинув их от нуля.

Решаем 2 уравнение, подставляя в него значение w, полученное выше:

$$w^T X^T \vec{1} + \vec{1}^T X w + 2bn - 2\vec{1}^T y =$$

$$= y^T X (X^T X + \lambda I)^{-T} X^T \vec{1} - b\vec{1}^T X (X^T X + \lambda I)^{-T} X^T \vec{1} +$$

$$+ \vec{1}^T X (X^T X + \lambda I)^{-1} X^T y - \vec{1}^T X (X^T X + \lambda I)^{-1} X^T \vec{1} b + 2nb - 2\vec{1}^T y = 0$$

$$b = \frac{y^T X (X^T X + \lambda I)^{-T} X^T \vec{1} + \vec{1}^T X (X^T X + \lambda I)^{-1} X^T y - 2\vec{1}^T y}{\vec{1}^T X (X^T X + \lambda I)^{-T} X^T \vec{1} + \vec{1}^T X (X^T X + \lambda I)^{-1} X^T \vec{1} - 2n} = \frac{\vec{1}^T (X (X^T X + \lambda I)^{-1} X^T - I) y}{\vec{1}^T X (X^T X + \lambda I)^{-1} X^T \vec{1} - n}$$

Ответ:

$$w = (X^T X + \lambda I)^{-1} X^T (y - b\vec{1}), b = \frac{\vec{1}^T (X (X^T X + \lambda I)^{-1} X^T - I) y}{\vec{1}^T X (X^T X + \lambda I)^{-1} X^T \vec{1} - n}$$

Задача 4 (1.5 балла). Логистическая регрессия, решение оптимизационной задачи.

1. (0.5 балла) Докажите, что в случае линейно разделимой выборки не существует вектора параметров (весов), который бы максимизировал правдоподобие вероятностной модели логистической регрессии в задаче двухклассовой классификации.

   Решение:

   Рассмотрим правдоподобие:

   $\prod\limits_{i=1}^{n} p_i^{[y_i=1]} (1-p_i)^{[y_i=-1]}) = \prod\limits_{i=1}^{n} \left(\frac{1}{1+e^{-\langle w, x_i \rangle}} \right)^{[y_i=1]} \left(\frac{1}{1+e^{\langle w, x_i \rangle}} \right)^{[y_i=-1]} = \prod\limits_{i=1}^{n} \frac{1}{1+e^{-y_i \langle w, x_i \rangle}}.$

Для линейно разделимой выборки $y_i\langle w, x_i\rangle > 0$. Заметим, что для $\forall c > 0$ $y_i\langle cw, x_i\rangle > 0$. Таким образом, увеличивая бесконечно норму весов, максимальное правдоподобие будет возрастать (но не достигать 1, те 1 - является асимптотой), получаем, что не существует вектора параметров (весов), который бы максимизировал правдоподобие.

2. (0.3 балла) Предложите, как можно модифицировать вероятностную модель, чтобы оптимум достигался.

Решение:

Чтобы оптимум достигался, нужно не дать весам принимать слишком большие значения, для этого добавляют регуляризатор.

На самом деле, чтобы получить регуляризованную модель для логистической регрессии мы можем предположить, что вместе в параметрической моделью имеется априорное распределение в пространстве параметров модели, тогда

$$p(X, y, w) = p(X, y|w)p(w; \sigma)$$

.

Предположим, что веса w_i имеют нормальное распределение $N(0, \sigma^2)$. Рассмотрим задачу максимального правдоподобия с такой точки зрения, тогда:

$$\log p(X, y, w) = \log p(X, y|w) + \log p(w; \sigma) = \log p(y|X, w) + \log p(X) + \log p(w; \sigma) \to \max_w$$

$$\log p(X, y, w) = \log p(y|X, w) + \log p(w; \sigma) \to \max_w$$

$$\log p(X, y, w) = \log p(y|X, w) + \log p(w; \sigma) = -\sum_{i=1}^{n} \log(1 + e^{-y_i\langle x_i, w\rangle}) + \log \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{||w||^2}{2\sigma^2}} =$$

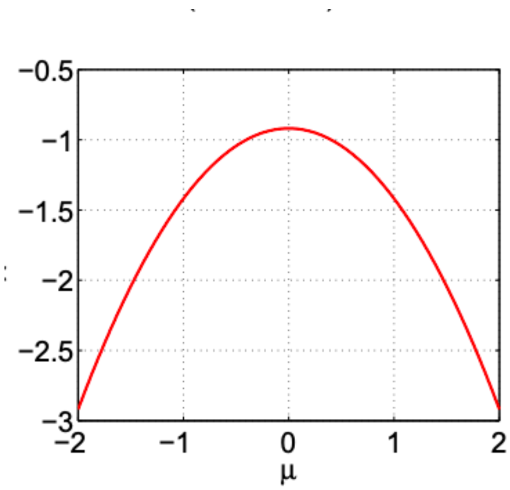
$$= -\sum_{i=1}^{n} \log(1 + e^{-y_i\langle x_i, w\rangle}) + \log C - \frac{||w||^2}{2\sigma^2} \to \max_w$$

Максимум не зависит от константы C, поэтому в итоге получаем следующую оптимизационную задачу:

$$\log p(X, y, w) = -\sum_{i=1}^{n} \log(1 + e^{-y_i\langle x_i, w\rangle}) - \frac{||w||^2}{2\sigma^2} \to \max_w$$

Заметим, что если взять минус логарифм от правдоподобия, то мы придём к задаче миним-мизации, которую обычно и пишут в материалах для L2 логистической регрессии.

Почему же будет точка оптимума? Если доказывать не строго, то график логарифма правдоподобия будет выглядеть следующим образом:



На графике видно, что существует точка оптимума (максимума) логарифма правдобододобия, поэтому существует и оптимум (максимум) у правдоподобия в этой же самой точке.

3. (0.7 балла) Что можно сказать о единственности решения L2-регуляризованной задачи? Почему?

   Решение:

   1) В предыдущем пункте было показано, что существует точка максимума, давайте покажем, что она единственна.

   2) Если продифференцировать логарифм правдоподобия по w_j дважды, то его значение будет отрицательным для любого значения w_j, таким образом функция по этому аргументу будет вогнута вниз, т.е. сначала значения будут возростать, а после убывать (получаем точку максимума). Так как это рассуждение выполняется для любого веса w_j, то мы и получаем единственную точку максимума.

$$\frac{\partial p(X,t,w)}{\partial w_j} = -\sum_i \frac{-y_i x_{ij}}{1+e^{y_i\langle x_i,w\rangle}} - \frac{w_j}{\sigma^2}$$

$$\frac{\partial^2 p(X,t,w)}{\partial w_j^2} = -\sum_i \frac{x_{ij}^2}{(1+e^{-y_i\langle x_i,w\rangle})^2} - \frac{1}{\sigma^2} < 0$$

   Ответ: L2-регуляризованная задача имеет единственное решение

Задача 5 (1 балл). Мультиномиальная регрессия.

В случае многоклассовой классификации логистическую регрессию можно обобщить: пусть для каждого класса k есть свой вектор весов w_k. Тогда вероятность принадлежности классу k запишем следующим образом:

$$P(y=k|x,W) = \frac{e^{\langle w_k,x\rangle}}{\sum_{j=1}^K e^{\langle w_j,x\rangle}}$$

Тогда оптимизируемая функция примет вид:

$$\mathcal{L}_{sm}(W) = -\sum_{i=1}^N\sum_{k=1}^K [y_i=k]\ln P(y_i=k|x_i,W),\ \text{где } [y_i=k] = \begin{cases}1, y_i=k,\\ 0, \text{иначе}\end{cases}$$

Пусть количество классов $K=2$. Для простоты положим, что выборка линейно неразделима.

1. (0.5 балла) Единственно ли решение задачи? Почему?

   Решение:

   На самом деле решение задачи не единственно, для этого покажем, что добавив один и тот же вектор z к каждому вектору весов w_j мы получим такие же вероятности принадлежности классам. Действительно,

$$P(y=k|x,W') = \frac{e^{\langle w_k+z,x\rangle}}{\sum_{j=1}^{K} e^{\langle w_j+z,x\rangle}} = \frac{e^{\langle w_k,x\rangle}e^{\langle z,x\rangle}}{\sum_{j=1}^{K}\left(e^{\langle w_j,x\rangle}e^{\langle z,x\rangle}\right)} =$$

$$= \frac{e^{\langle z,x\rangle}e^{\langle w_k,x\rangle}}{e^{\langle z,x\rangle}\sum_{j=1}^{K} e^{\langle w_j,x\rangle}} = \frac{e^{\langle w_k,x\rangle}}{\sum_{j=1}^{K} e^{\langle w_j,x\rangle}} = P(y=k|x,W)$$

   Ответ: Решение задачи не единственно.

2. (0.5 балла) Покажите, что предсказанные распределения вероятностей на классах в случае логистической и мультиномиальной регрессий будут совпадать.

   Решение:

   Распишем $\mathcal{L}_{sm}(W)$ для количества классов $K=2$:

$$\mathcal{L}_{sm}(W) = -\sum_{i=1}^{N}([y_i=1]\ln P(y_i=1|x_i,W) + [y_i=2]\ln P(y_i=2|x_i,W)) =$$

$$= -\sum_{i=1}^{N}([y_i=1]\ln\frac{e^{\langle w_1,x\rangle}}{e^{\langle w_1,x\rangle}+e^{\langle w_2,x\rangle}} + [y_i=2]\ln\frac{e^{\langle w_2,x\rangle}}{e^{\langle w_1,x\rangle}+e^{\langle w_2,x\rangle}}) =$$

$$= -\sum_{i=1}^{N}([y_i=1]\ln\frac{1}{1+e^{\langle w_2-w_1,x\rangle}} + [y_i=2]\ln\frac{1}{e^{\langle w_1-w_2,x\rangle}+1})$$

$$= -\sum_{i=1}^{N}([y_i=1]\ln\frac{1}{1+e^{-\langle w_1-w_2,x\rangle}} + [y_i=2]\ln\frac{1}{1+e^{\langle w_1-w_2,x\rangle}})$$

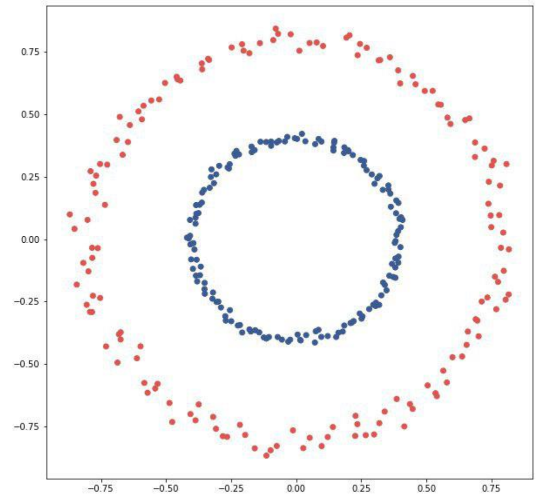
   Если перенумеровать 2 класс в -1, то получаем следующее:

$$-\sum_{i=1}^{N}([y_i=1]\ln\frac{1}{1+e^{-\langle w_1-w_2,x\rangle}} + [y_i=-1]\ln\frac{1}{1+e^{\langle w_1-w_2,x\rangle}}) =$$

$$= -\sum_{i=1}^{N}\ln\frac{1}{1+e^{-y_i\langle w_1-w_2,x\rangle}} = \sum_{i=1}^{N}\ln(1+e^{-y_i\langle w_1-w_2,x\rangle})$$

   На самом деле мы получили минус логарифм правдоподобия для логистической регрессии. Таким образом, мы получили, что распределения вероятностей на классах в случае логистической и мультиномиальной регрессии будут совпадать.

Задача 6 (0.5 балла) Нейронные сети.
Дана выборка из двух концентрических окружностей:



Допустим, что для классификации нужно обучить нейронную сеть — причем доступны только следующие слои: линейный $L(n,m)$ ($Wx+b,\ x \in \mathbb{R}^n,\ b \in \mathbb{R}^m$) и активация A (сигмоида или tanh), которые разрешено последовательно ставить друг после друга.

Вопрос: какие из приведенных ниже архитектур будут способны разделить выборку со 100% accuracy? Почему?

1. $L(2,2) \to A \to L(2,1)$
2. $L(2,2) \to A \to L(2,2) \to A \to L(2,1)$
3. $L(2,3) \to L(3,1)$
4. $L(2,3) \to A \to L(3,1)$
5. $L(2,3) \to L(3,3) \to L(3,1)$

Решение:

Понятно, что архитектуры под номерами 3 и 5 не подходят для данной задачи, так как они строят линейную разделяющую поверхность. Архитектуры под пунктами 1 и 2 тоже не будут распознавать окружности со 100% accuracy, потому что для получения такой оценки нужно "выгибать"двухмерное пространство (Рис. 2) в пространство большей размерности, а в этих пунктах преобразуется двухмерное пространство в двухмерное (Рис.1). Поэтому ответом будет архитектура под номером 4.

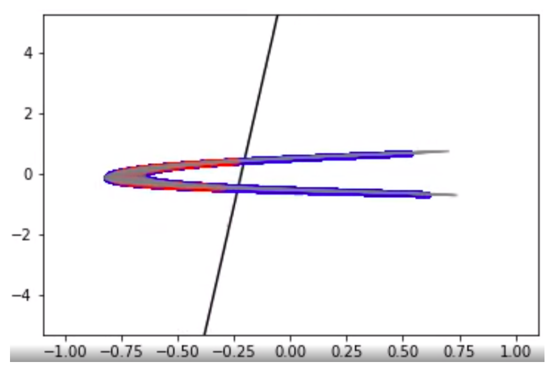


Рис. 1

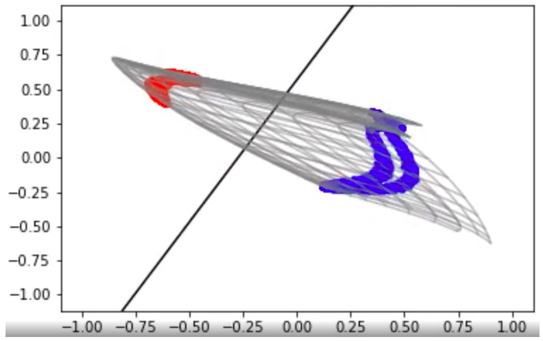


Рис. 2

Задача 7 (1.5 балла) Нейронные сети, back-prop.

Рассмотрим двуслойную полносвязную нейронную сеть, применяемую для задачи классификации. На вход нейронной сети подается вектор признаков x размерности n, полносвязный слой с матрицей весов W размерности $n \times d$ преобразует вектор x в скрытое представление h некоторой размерности d:

$$h = xW$$

Функции активации нет, еще один полносвязный слой с матрицей весов W' размерности $d \times m$ преобразует скрытое представление в вектор оценок a принадлежности к каждому классу. Чтобы вычисления были более устойчивыми, используется logsoftmax, который предсказывает логарифмы вероятностей:

$$o_j = a_j - \log \sum_{k=1}^{m} \exp(a_k)$$

Логарифм суммы экспонент здесь можно вычислить при помощи *трюка LogSumExp*[2]. Тогда функция потерь (кросс-энтропия) запишется так:

$$\mathcal{L} = -\sum_{j=1}^{m} y_j o_j,$$

где y – one-hot encoding истинной метки объекта.

Итак, мы полностью описали проход по нейронной сети вперед: как по входному вектору x найти логарифмы вероятности классов o_j и вычислить значение функции потерь, зная ответ y на рассматриваемом объекте. Опишите обратный проход по нейронной сети: выпишите формулы изменения матриц весов W и W' в стохастическом градиентном спуске для метода обратного распространения ошибки (backpropagation).

Решение:

$$h_l = \sum_{i=1}^{n} x_i w_{il} \qquad a_k = \sum_{i=1}^{d} h_i w'_{ik} \qquad o_j = a_j - \log \sum_{i=1}^{m} e^{a_i} \qquad \mathcal{L} = -\sum_{i=1}^{m} y_i o_i$$

$$\frac{\partial \mathcal{L}}{\partial w'_{lk}} = \frac{\partial \mathcal{L}}{\partial o_k} \frac{\partial o_k}{\partial a_k} \frac{\partial a_k}{\partial w'_{lk}} = -y_k \left(1 - \frac{e^{a_k}}{\sum\limits_{i=1}^{m} e^{a_i}}\right) h_l$$

$$\frac{\partial \mathcal{L}}{\partial w_{jl}} = \frac{\partial \mathcal{L}}{\partial h_l} \frac{\partial h_l}{\partial w_{jl}} = -\sum_{i=1}^{m} y_i \left(1 - \frac{e^{a_i}}{\sum\limits_{j=1}^{m} e^{a_j}}\right) w'_{li} x_j$$

[2] https://en.wikipedia.org/wiki/LogSumExp#log-sum-exp_trick_for_log-domain_calculations

Обратный проход:
$w'_{lk} = w'_{lk} - \mu \frac{\partial \mathcal{L}}{\partial w'_{lk}}$
$w_{jl} = w_{jl} - \mu \frac{\partial \mathcal{L}}{\partial w_{jl}}$
μ - размер шага, x_j - j-ый признак x
В матричной форме:

$$W' = W' + \mu h^T y \text{diag}\left(1 - \frac{e^{a_1}}{\sum\limits_{i=1}^{m} e^{a_i}}, ..., 1 - \frac{e^{a_m}}{\sum\limits_{i=1}^{m} e^{a_i}}\right)$$

$$W = W + \mu x^T y \text{diag}\left(1 - \frac{e^{a_1}}{\sum\limits_{i=1}^{m} e^{a_i}}, ..., 1 - \frac{e^{a_m}}{\sum\limits_{i=1}^{m} e^{a_i}}\right)(W')^T$$

где h, x, y - вектор-строки, diag(...) - диагональная матрица.

Задача 8 (2.5 балла) Нейронные сети, инициализация весов.

Рассмотрим полносвязный слой нейронной сети с матрицей весов W и свободным членом b, получающий на вход вектор x размерности n и вычисляющиющий скрытое представление размерности m

$$h = Wx + b.$$

Предложите, из какого невырожденного вероятностного распределения надо выбирать веса W и b, чтобы активации h имели нормальное распределение $N(0, \sigma^2)$, если

(a) **(0.5 балла)** Все признаки независимы и распределены по стандартному нормальному закону.

(b) **(2 балла)** Все признаки независимы и распределены равномерно от 0 до a.

Распределения W и b не обязаны совпадать, они могут быть из разных семейств.