

Генеративная нейронная сеть быстрого обучения

А. Ю. Дорогов

Санкт-Петербургский государственный электротехнический университет
«ЛЭТИ» им. В.И. Ульянова (Ленина)

vaksa2006@yandex.ru

Аннотация. В статье предложен способ построения генеративных моделей на основе пирамидальных нейронных сетей быстрого обучения (БНС). В основе построения моделей положен вероятностный метод главных компонент (PPCA). Метод PPCA позволяет аналитически построить матрицы оптимальных декодеров способных восстанавливать образы из случайных латентных переменных. В работе представлены методы обучения БНС к матрицам декодеров. Обучение выполняется за конечное число шагов и не требует итерационных процедур. Приводятся примеры построения реализующих БНС для набора данных MNIST. Показаны результаты генерации образов стилистически подобных набору MNIST.

Ключевые слова: генеративная модель, вариационный автоэнкодер, вероятностный метод главных компонент, быстрая нейронная сеть, пирамидальная структура

I. ВВЕДЕНИЕ

К семейству генеративных моделей относятся вариационные автоэнкодеры (VAE) [1,2]. Классический автоэнкодер, состоит из двух основных частей: энкодера, преобразующего входные данные в сжатое представление (латентное или иначе скрытое пространство), и декодера, который восстанавливает данные из латентного пространства. Вариационный автоэнкодер добавляет в эту архитектуру случайный шум, предполагая, что в латентном пространстве данные распределены согласно предварительно выбранному вероятностному распределению, чаще всего нормальному с единичной матрицей ковариаций (рис. 1).

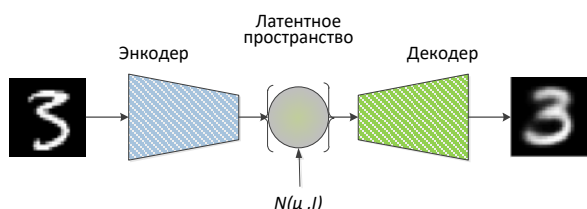


Рис. 1. Архитектура вариационного автоэнкодера

При обучении автоэнкодера одновременно решаются две оптимизационные задачи:

1) Модель энкодера стремится связать d мерный вектор обучающей выборки t с некоторым q мерным случайным вектором латентных переменных с нормальным законом вероятностного распределения:

$$x = f(t, w_E) \rightarrow N(\mu, I),$$

где w_E – параметры энкодера, $f(t, w_E)$ – функция отображения целевых образов в латентное пространство.

2) Модель декодера $t = \varphi(x; w_D)$ с параметрами w_D напротив обучается правильно восстанавливать образ t по случайной латентной переменной x при условии, что

она соответствует многомерному нормальному закону распределения $N(\mu, I)$ с математическим ожиданием μ .

Для реализации энкодера и декодера обычно используются многослойные нейронные сети, которые обучаются методом обратного распространения ошибок. После того как обучение завершено энкодер уже больше не требуется. Для генерации образов, на вход декодера достаточно подать значение латентной переменной, выбранной случайным образом из априорно заданного распределения $N(\mu, I)$.

Улучшенным вариантом VAE является условный автоэнкодер (Conditional VAE). В этой архитектуре существует дополнительный вход, на который поступает метка класса. По существу, для каждого класса формируется свой вариационный автоэнкодер, который обучается на примерах, принадлежащих только этому классу. Благодаря этому решению реалистичность генерируемых образов улучшается, но приводит к необходимости использования дополнительных ресурсов при реализации архитектуры CVAE. Для больших нейронных сетей процедура обучения VAE и CVAE занимает длительное время и требует большого объема выборки.

Вариационные автоэнкодеры имеют линейный прототип – это вероятностный метод главных компонент (англ. Probabilistic Principal Component Analysis – PPCA) [3]. Отличительная особенность метода PPCA состоит в том, что он позволяет найти аналитическое решение для функции отображения из пространства образов в пространство латентных переменных и обратно. Полученное решение можно непосредственно использовать для построения декодера генеративной модели, необходимость в энкодере в этом случае отпадает.

В данной работе будет рассмотрено построение генеративных моделей с использованием метода PPCA на основе многослойных пирамидальных нейронных сетей быстрого обучения [4]. Благодаря самоподобной структуре, сети данного класса способны быстро обучаться к произвольным функциям. Этими функциями могут быть, например, координатные компоненты многомерного отображения векторов из латентного пространства в пространство образов. Алгоритм обучения быстрых нейронных сетей принципиально отличается от метода обратного распространения ошибки отсутствием длительной итерационной процедуры по минимизации ошибки. Обучение всегда выполняется за конечное число шагов с гарантированной точностью.

II. ВЕРОЯТНОСТНЫЙ МЕТОД ГЛАВНЫХ КОМПОНЕНТ

Идея метода РРСА базируется на теоретических разработках конца 90-х годов, имеющих общее название байесовские методы машинного обучения [5, 6, 7]. Следуя работе [5], рассмотрим кратко основные положения данного решения. Согласно методу РРСА, связь между образами обучающей выборки t и латентными переменными x задаётся линейным отображением вида:

$$t = Wx + \mu + \varepsilon, \quad (1)$$

где W – это матрица размерностью $d \times q$, $\varepsilon \sim N(0, \sigma^2 I)$ случайный шум, соответствующий нормальному закону с нулевым средним, диагональной ковариационной матрицей и дисперсией σ^2 , параметр μ позволяет модели данных иметь ненулевое среднее значение. Для латентных переменных декларируется изотропный нормальный закон распределения с единичной дисперсией:

$$p(x) = (2\pi)^{-q/2} \exp\left\{-\frac{1}{2}x^T x\right\}. \quad (2)$$

Выражение (1) подразумевает распределение вероятностей по t -пространству для данного x в виде:

$$p(t|x) = (2\pi\sigma^2)^{-d/2} \exp\left\{-\frac{1}{2\sigma^2}\|t - Wx - \mu\|^2\right\}. \quad (3)$$

Используя (2) и (3) получим предельное (маргинальное) распределение для образов t :

$$p(t) = \int p(t|x)p(x) = \\ = (2\pi)^{-d/2} |C|^{-1/2} \exp\left\{-\frac{1}{2}(t - \mu)^T C^{-1}(t - \mu)\right\},$$

где $C = \sigma^2 I + WW^T$, ковариационная матрица (что также можно непосредственно получить из выражения (1)). Используя правило Байеса, можно вычислить апостериорное распределение скрытых переменных x при заданном наблюдаемом t ,

$$p(x|t) = \frac{p(t|x)p(x)}{p(t)} = (2\pi)^{-q/2} |\sigma^2 M|^{1/2} \times \\ \times \exp\left[-\frac{1}{2}\{x - M^{-1}W^T(t - \mu)\}^T (\sigma^{-2}M) \{x - M^{-1}W^T(t - \mu)\}\right],$$

где апостериорная ковариационная матрица задаётся выражением:

$$\sigma^2 M^{-1} = \sigma^2 (\sigma^2 I + W^T W)^{-1}.$$

Для определения оптимального отображения (1) запишем логарифмическую функцию правдоподобия по обучающей выборке размером N , и после преобразований получим:

$$L = \sum_{n=1}^N \ln\{p(t_n)\} = -\frac{N}{2} \{d \ln(2\pi) + \ln|C| + \text{tr}(C^{-1}S)\},$$

где матрица

$$S = \frac{1}{N} \sum_{n=1}^N (t_n - \mu)(t_n - \mu)^T,$$

является ковариационной матрицей обучающей выборки $\{t_n\}$, N – размер выборки. Для нахождения оптимального отображения, функция правдоподобия максимизируется по переменным μ и W , следуя следующим условиям экстремума:

$$\frac{dL}{d\mu} = 0, \quad \frac{dL}{dW} = 0.$$

В работе [5] показано, что оптимальные значения определяются выражениями:

$$\mu = \frac{1}{N} \sum_{n=1}^N t_n, \quad W = U_q (\Lambda_q - \sigma^2 I)^{1/2} R, \quad (4)$$

где матрица U_q (размером $d \times q$) содержит q векторов-столбцов, которые являются собственными векторами выборочной ковариационной матрицы S с собственными значениями $\lambda_1, \lambda_2, \dots, \lambda_q$ размещёнными в диагональной матрице Λ_q в порядке убывания, R – произвольная $q \times q$ ортогональная матрица вращения. Также показано, что при оптимальном выборе матрицы W , оценка максимального правдоподобия для σ^2 в t -пространстве определяется как

$$\sigma^2 = \frac{1}{d - q} \sum_{j=q+1}^d \lambda_j,$$

где $\lambda_{q+1}, \lambda_{q+2}, \dots, \lambda_d$ наименьшие собственные значения матрицы S . Для восстановления образа по латентной переменной может быть использовано математического ожидания для модели (1):

$$t = Wx + \mu.$$

В [5] отмечается, что такая оценка возможна, но не является оптимальной. Показано, что оптимальная оценка может быть получена с использованием следующего выражения:

$$t = W(W^T W)^{-1} Mx + \mu.$$

Таким образом функция декодера генеративной модели определяется матрицей W или матричным произведением $H = W(W^T W)^{-1} M$, где $M = \sigma^2 I + W^T W$. Напомним, что W является прямоугольной матрицей размерности $d \times q$, а M – квадратной матрицей размерности $d \times d$ и $q \ll d$.

III. ПИРАМИДАЛЬНЫЕ НЕЙРОННЫЕ СЕТИ БЫСТРОГО ОБУЧЕНИЯ

Теория данного класса сетей представлена в работах [4, 8]. Рассмотрим построение декодера на обратно-ориентированных пирамидальных нейронных сетях. Эти сети имеют глубокие связи с алгоритмом быстрого преобразования Фурье (БПФ). Возьмём за основу топологическую модель «Кули-Гьюки с прореживанием по времени» [9].

$$U^m = \langle u_{n-1} u_{n-2} \cdots u_{m+1} u_m v_{m-1} v_{m-2} \cdots v_1 v_0 \rangle,$$

$$V^m = \langle u_{n-1} u_{n-2} \cdots u_{m+1} v_m v_{m-1} v_{m-2} \cdots v_1 v_0 \rangle,$$

$$z^m = \langle u_{n-1} u_{n-2} \cdots u_{m+1} v_{m-1} v_{m-2} \cdots v_1 v_0 \rangle.$$

В этой модели кортежи представляют собой поразрядные представления позиционных номеров рецепторов – U^m , аксонов – V^m и нейронных ядер – z^m в пределах нейронного слоя m . В свою очередь, разрядные переменные u_m – это позиционный номер рецептора в пределах ядра, а v_m – позиционный номер аксона в пределах ядра слоя m (нумерация рецепторов, аксонов и нейронных ядер начинается с нуля). Сеть имеет послонную ядерную структуру с числом слоёв равным n . В терминах БПФ нейронное ядро соответствует базовой операции, например «бабочка». В общем случае нейронное ядро представляет собой матрицу весов размером $p_m \times g_m$. Функции активации и входы смещений отсутствуют. Совокупность чисел $p_0, p_1, \dots, p_{n-1}$ и $g_0, g_1, \dots, g_{n-1}$ определяют структурные характеристики сети. Сеть линейна и относится к категории быстрых нейронных сетей (БНС). Для построения пирамидальной сети выберем структурные характеристики сети следующим образом:

- размерности рецепторных полей сети положим равными $p_0 = p_1 = \dots = p_{n-2} = 1$, но значение $p_{n-1} \neq 1$ выбирается равным размерности латентного пространства. В этом случае размерность сети по входу будет равна $N = p_0 p_1 \cdots p_{n-1} = p_{n-1}$;
- размерности аксоновых полей сети зададим произвольными натуральными целыми числами $g_0, g_1, \dots, g_{n-1}$. Размерность сети по выходу в этом случае будет равна $M = g_0 g_1 \cdots g_{n-1}$. Это значение выбирается равным размерности t – пространства.

Топологическая модель при данных структурных характеристиках будет иметь вид:

$$U^m = \langle u_{n-1} 0_{n-2} \cdots 0_{m+1} 0_m v_{m-1} v_{m-2} \cdots v_1 v_0 \rangle,$$

$$V^m = \langle u_{n-1} 0_{n-2} \cdots 0_{m+1} v_m v_{m-1} v_{m-2} \cdots v_1 v_0 \rangle,$$

$$z^m = \langle u_{n-1} 0_{n-2} \cdots 0_{m+1} v_{m-1} v_{m-2} \cdots v_1 v_0 \rangle.$$

На рис. 2 приведён пример топологической модели трёхслойной пирамидальной нейронной сети для структурных параметров $[p_0 p_1 p_2] = [113]$, $[g_0 g_1 g_2] = [227]$.

Сеть имеет размерность по входу равную 3, а по выходу 28. В [8] показано, что пирамидальная сеть обратной ориентации может хранить произвольные дискретные функции, размерность которых определяется размерностью выхода сети, а их количество – размерностью входа. Например сеть, показанная на рис. 2, способна хранить три произвольные вектор-функции размерностью 28.

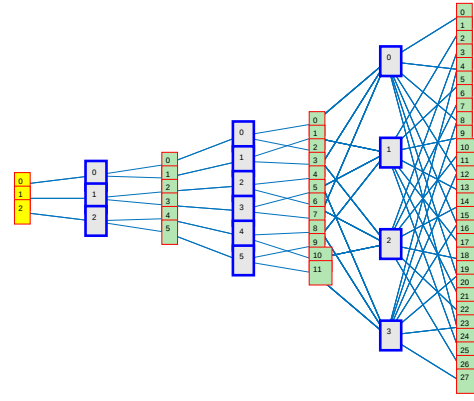


Рис. 2. Топология пирамидальной нейронной сети. Выделенные нумерованные вершины представляют нейронные ядра

Следуя [10] рассмотрим кратко способ обучения сети. В соответствии с обобщённой теоремой факторизации элементы матрицы БНС представляются в виде:

$$h(U, V) = w_{z^{n-1}}^{n-1}(u_{n-1}, v_{n-1}) w_{z^{n-2}}^{n-2}(0_{n-2}, v_{n-2}) \cdots w_{z^0}^0(0_0, v_0), \quad (5)$$

где U, V – позиция элемента в матрице БНС, $w_{z^m}^m(u_m, v_m)$ – элемент матрицы нейронного ядра с номером z^m в слое m , u_m, v_m – позиция элемента в матрице нейронного ядра. Пусть образы $f^k(V)$ представляет собой набор, состоящий из p_{n-1} дискретных функций, заданных на интервале длиной M . Выполним мультипликативную декомпозицию каждой функции по переменным v_i , начиная со старшего разряда [8], в результате функция образа представляется в виде:

$$f^k(V) = \varphi_{i^{n-1}}^k(v_{n-1}) \varphi_{i^{n-2}}^k(v_{n-2}) \cdots \varphi_{i^0}^k(v_0), \quad (6)$$

где $i^m = \langle v_{m-1} v_{m-2} \cdots v_1 v_0 \rangle$ – позиционный номер множителя уровня m , k – порядковый номер функции.

Сравнивая выражения (5) и (6) непосредственно получим следующее правило обучения нейронной сети:

$$w_{z^{n-1}}^{n-1}(u_{n-1}, v_{n-1}) = \varphi_{i^{n-1}}^k(v_{n-1}) \quad \text{для } m = n-1,$$

$$w_{z^m}^m(0_m, v_m) = \varphi_{i^m}^k(v_m) \quad \text{для } m < n-1,$$

$$z^m = \langle u_{n-1} 0_{n-2} \cdots 0_{m+1} v_{m-1} v_{m-2} \cdots v_1 v_0 \rangle,$$

$$i^m = \langle v_{m-1} v_{m-2} \cdots v_1 v_0 \rangle.$$

Предварительно, для упорядочивания хранимых функций, должно быть установлено взаимно-однозначное соответствие $k \leftrightarrow u_{n-1}$ между порядковым номером функции и разрядной переменной u_{n-1} . Считывание памяти выполняется установкой на входе нейронной сети унарного кода, в котором только один из разрядов равен 1, а остальные нули. Для сети, показанной на рис. 2, – это будут комбинации $[001]$, $[010]$ или $[100]$. При использовании кодов $[110]$ образы 1 и 2 складываются, а при кодах $[1-10]$ из первого образа вычитается второй, в общем случае при значениях на входе $[\alpha \beta \gamma]$ на выходе получим образ

$q = \alpha f^1 + \beta f^2 + \gamma f^3$. Заметим, что тот же результат можно получить, если умножить вектор строку $[\alpha \beta \gamma]$ на матрицу размерностью 3×28 , строки которой совпадают с вектор-функциями f^1 , f^2 и f^3 :

$$[\alpha \beta \gamma] \times \begin{bmatrix} f_0^1 & f_1^1 & \dots & f_{27}^1 \\ f_0^2 & f_1^2 & \dots & f_{27}^2 \\ f_0^3 & f_1^3 & \dots & f_{27}^3 \end{bmatrix} = \alpha f^1 + \beta f^2 + \gamma f^3.$$

Отсюда следует, что пирамидальная нейронная сеть является факторизованным представлением прямоугольной матрицы, строками которой являются хранимые образы. Это можно доказать, более строго используя понятие числа степеней свободы [11]. Хранимые образы – это произвольные функции, например, это могут быть строки матрицы РРСА декодера.

Факторизованное представление матрицы состоит из параллельных вычислительных элементов, организованных как нейронная сеть, что позволяет обеспечить высокое быстродействие при использовании специализированных процессоров с распараллеливанием операций. В этом состоит главное преимущество нейросетевой реализации.

Пирамидальную нейронную сеть не сложно обобщить на двумерный случай, удобный для обработки изображений. Сохраним для каждой координаты изображения тип топологии рассмотренной выше одномерной модели, тогда топологическая модель пирамидальной сети памяти для двумерного случая примет вид:

$$\begin{aligned} U_*^m &= \langle u_{n-1}^* 0_{n-2}^* \dots 0_{m+1}^* 0_m^* v_{m-1}^* v_{m-2}^* \dots v_1^* v_0^* \rangle, \\ V_*^m &= \langle u_{n-1}^* 0_{n-2}^* \dots 0_{m+1}^* v_m^* v_{m-1}^* v_{m-2}^* \dots v_1^* v_0^* \rangle, \\ z_*^m &= \langle u_{n-1}^* 0_{n-2}^* \dots 0_{m+1}^* v_{m-1}^* v_{m-2}^* \dots v_1^* v_0^* \rangle. \end{aligned}$$

Символ (*) здесь заменяет символы x и y означающих принадлежность к координатам изображения. Элементы четырёхмерной матрицы БНС для данной модели также выражаются через элементы нейронных ядер:

$$\begin{aligned} h(U_y U_x, V_y V_x) &= w_{z_x^{n-1}, z_y^{n-1}}^{n-1} (u_{n-1}^y, u_{n-1}^x; v_{n-1}^y, v_{n-1}^x) \cdot \\ &\cdot w_{z_x^{n-2}, z_y^{n-2}}^{n-2} (0_{n-2}^y, 0_{n-2}^x; v_{n-2}^y, v_{n-2}^x) \dots w_{z_x^0, z_y^0}^0 (0_0^y, 0_0^x; v_0^y, v_0^x), \end{aligned}$$

а мультипликативная декомпозиция сохраняемых изображений имеет вид:

$$f^k(V_y V_x) = \varphi_{i_x^{n-1}, i_y^{n-1}}^k (v_{n-1}^y, v_{n-1}^x) \varphi_{i_x^{n-2}, i_y^{n-2}}^k (v_{n-2}^y, v_{n-2}^x) \dots \varphi_{i_x^0, i_y^0}^k (v_0^y, v_0^x).$$

Сравнивая два последних выражения, получим правило обучения двумерной пирамидальной нейронной сети:

$$\begin{aligned} w_{z_x^{n-1}, z_y^{n-1}}^{n-1} (u_{n-1}^y, u_{n-1}^x; v_{n-1}^y, v_{n-1}^x) &= \varphi_{i_x^{n-1}, i_y^{n-1}}^k (v_{n-1}^y, v_{n-1}^x) \quad \text{для } m = n-1, \\ w_{z_x^{n-2}, z_y^{n-2}}^{n-2} (0_{n-2}^y, 0_{n-2}^x; v_{n-2}^y, v_{n-2}^x) &= \varphi_{i_x^{n-2}, i_y^{n-2}}^k (v_{n-2}^y, v_{n-2}^x) \quad \text{для } m < n-1, \\ z_*^m &= \langle u_{n-1}^* 0_{n-2}^* \dots 0_{m+1}^* v_{m-1}^* v_{m-2}^* \dots v_1^* v_0^* \rangle, \\ i_*^m &= \langle v_{m-1}^* v_{m-2}^* \dots v_1^* v_0^* \rangle. \end{aligned}$$

Предварительно, для упорядочивания хранимых функций, должно быть установлено взаимно-однозначное соответствие $k \leftrightarrow \langle u_{n-1}^y, u_{n-1}^x \rangle$ между порядковым номером функции и значениями переменных u_{n-1}^y, u_{n-1}^x . Считывание памяти выполняется установкой на входе нейронной сети двумерной маски размерностью $p_{n-1}^y \times p_{n-1}^x$, в которой только один из элементов равен 1, а остальные нули. На рис. 3. показана топологическая модель двумерной БНС.

По входу сеть имеет размерность 3×3 , а по выходу 28×28 , размерность латентного пространства равна 9. Сеть трёхслойная, структурные характеристики модели определяются размерностями рецепторных и аксоновых полей нейронных ядер по слоям: $p^x = p^y = [113]$, $g^x = g^y = [227]$, символы x и y обозначают пространственные координаты.

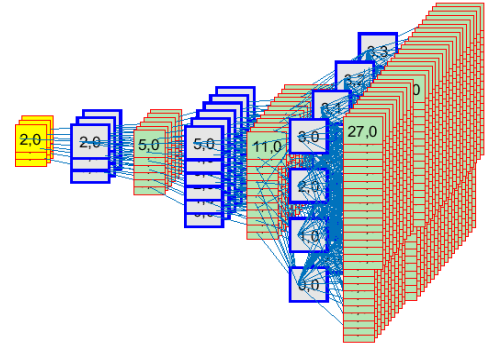


Рис. 3. Топологическая модель двумерной пирамидальной БНС

Сеть представляет собой факторизацию четырёхмерной матрицы размерностью $[3 \times 3, 28 \times 28]$ и способна сохранять 9 произвольных изображений размером 28×28 .

IV. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Эксперименты были проведены на наборе данных MNIST [12], который содержит образы рукописных цифр от 0 до 9 в виде пиксельных изображений размером 28×28 . Объем обучающей выборки равен 60000 изображений и 10000 содержит тестовая выборка. На рис. 4 показана выборка с десятью представителями для каждого класса. При проведении экспериментов все образы выборки предварительно нормировались по энергии к единичному уровню.

Размерность латентного пространства была выбрана равной 9. Для построения декодера в равной степени можно использовать как одномерную, так и двумерную реализацию сети.

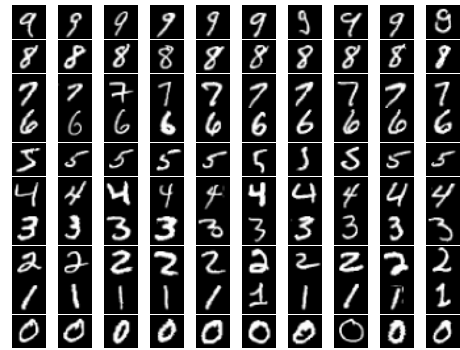


Рис. 4. Выборка из набора рукописных цифр

Вычисление матрицы декодера $H = \|h(V, U)\|$ удобно производить для одномерного случая, следуя выражению (4). По этой матрице можно непосредственно построить одномерную сеть, используя столбцы как функции обучающего множества для сети. Одномерная сеть будет иметь размерность 9 по входу и $784=28 \times 28$ по выходу. Для перехода к двумерному образу, выходной вектор перематрируется к матрице заменой одномерного индекса координат, двухразрядным числом $V = \langle V^y, V^x \rangle$ с основаниями разрядов равными размерностям изображения по осям x и y , в данном случае 28×28 .

Для построения двумерной нейронной сети матрицу H следует привести к четырёхмерной форме, это можно сделать представлением номеров строк и столбцов в двухразрядном виде $V = \langle V^y, V^x \rangle$ и $U = \langle u_{n-1}^y, u_{n-1}^x \rangle$. Обучающими образами сети будут являться двумерные плоскости четырёхмерной матрицы, определяемые разрядами u_{n-1}^y, u_{n-1}^x . Реализация двумерной сети показана на рис. 3.

Эксперименты показали, что метод PPCA позволяет получить вполне удовлетворительные результаты при реализации условного множественного декодера (аналог CVAE). В этом случае декодеры PPCA строятся для каждого класса отдельно, при этом используются все обучающие примеры конкретного класса. На рис. 5 показаны примеры множественной реконструкции на основе метода PPCA. Каждый столбец в таблице соответствует одной генеративной модели класса, а каждая строка – одной случайной реализации вектора латентных переменных.

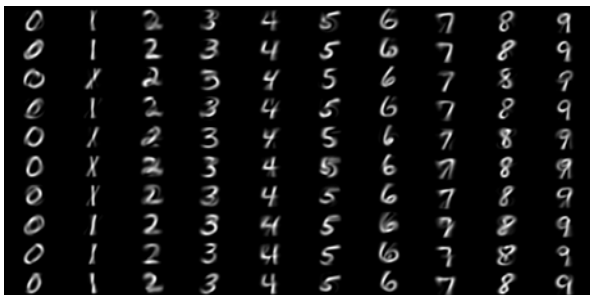


Рис. 5. Результаты реконструкции образов множественными генеративными моделями

В данной работе рассмотрен альтернативный вариант построения генеративных нейросетевых моделей на основе вероятностного метода главных компонент (PPCA). В отличие от классических VAE моделей, модель генеративного PPCA-декодера строится аналитически, и не требует длительного итерационного обучения, при этом сохраняется приемлемое качество генерации образов. В статье показано, что построенные матрицы декодеров могут быть эффективно реализованы в классе быстрых нейронных сетей. Настройка реализующей БНС выполняется за конечное число шагов, а число вычислительных операций при этом примерно такое же, как и число операции при генерации образа. Топология БНС задаётся аналитической моделью и генерируется автоматически. Генеративная модель для изображений может быть реализована как в одномерном, так и двумерном варианте БНС.

СПИСОК ЛИТЕРАТУРЫ

- [1] Carl Doersch. Tutorial on Variational Autoencoders, 2016, <https://arxiv.org/abs/1606.05908>.
- [2] Diederik P. Kingma, Max Welling, Auto-Encoding Variational Bayes, arXiv:1312.6114v11 [stat.ML] 10 Dec 2022.
- [3] Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. J. R. Statist. Soc. B (1999) 61, Part 3, pp. 611-622, Microsoft Research, Cambridge, UK
- [4] Дорогов А.Ю. Самоподобные нейронные сети быстрого обучения. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2024. 188 с. www.dorogov.su.
- [5] Tipping M.E. and C.M. Bishop. Mixtures of probabilistic principal component analyzers. Neural Computation 11(2), 443-482.
- [6] Tipping, M.E. and C.M. Bishop. Probabilistic principal component analysis. Journal of the Royal Statistical Society, Series B 21(3), 611-622.
- [7] Бишоп Кристофер М. Распознавание образов и машинное обучение / Пер. с англ. СПб.: ООО "Диалектика", 2020. 960 с.
- [8] Дорогов А.Ю. Быстрые преобразования и самоподобные нейронные сети глубокого обучения. Часть 3. Пирамидальные нейронные сети с глубокой степенью обучения // Информационные и математические технологии в науке и управлении. 2024. № 2(34). С. 19-32 DOI:10.25729/ESI.2024.34.2.002.
- [9] Дорогов А.Ю. Быстрые преобразования и самоподобные нейронные сети глубокого обучения. Часть 1. Стратифицированные модели самоподобных нейронных сетей и быстрых преобразований // Информационные и математические технологии в науке и управлении. 2023. № 4(32). С. 5-20. DOI: 10.25729/ESI.2023.32.4.001.
- [10] Дорогов А.Ю. Быстрые преобразования и самоподобные нейронные сети глубокого обучения. Часть 2. Методы обучения быстрых нейронных сетей // Информационные и математические технологии в науке и управлении. 2024. № 1(33). С. 5-19. DOI:10.25729/ESI.2024.33.1.001.
- [11] Дорогов А.Ю. Пластичность самоподобных нейронных сетей // Информационные и математические технологии в науке и управлении, 2024. № 3(35). С. 33-43. DOI: 10.25729/ESI.2024.35.3.003.
- [12] THE MNIST DATABASE of handwritten digits. <http://yann.lecun.com/exdb/mnist/>