

Модели согласования скрытого пространства в задаче прогнозирования*

Ф. Р. Яушев¹, Р. В. Исаченко², В. В. Стрижов³

Аннотация: В работе исследуется задача прогнозирования сложной целевой переменной. Под сложностью подразумевается наличие зависимостей, линейных или нелинейных. Предполагается, что исходные данные гетерогенны. Это значит, что пространства независимой и целевой переменных имеют разную природу. Предлагается построить предсказательную модель, которая учитывает зависимость в исходном пространстве независимой переменной, а также в пространстве целевой переменной. Согласование моделей предлагается производить в низкоразмерном пространстве. В качестве базового алгоритма используется метод проекции в скрытое пространство (PLS). В работе проводится сравнение линейного PLS и предложенных нелинейных моделей. Сравнение производится на гетерогенных данных в пространствах высокой размерности.

Ключевые слова: прогнозирование, модель частичных наименьших квадратов, задача декодирования, согласование скрытого пространства

DOI: 00.00000/0000000000000000

*Настоящая статья содержит результаты проекта Статистические методы машинного обучения, выполняемого в рамках реализации Программы Центра компетенций Национальной технологической инициативы «Центр хранения и анализа больших данных», поддерживаемого Министерством науки и высшего образования Российской Федерации по Договору МГУ им. М.В.Ломоносова с Фондом поддержки проектов Национальной технологической инициативы от 11.12.2018 № 13/1251/2018. Работа выполнена при поддержке РФФИ (проекты 19-07-01155, 19-07-00875).

¹Московский физико-технический институт, fyaush@mail.ru

²Московский физико-технический институт, roman.isachenko@phystech.edu

³Вычислительный центр имени А. А. Дородницына Федерального исследовательского центра «Информатика и управление» Российской академии наук, Московский физико-технический институт, strijov@phystech.edu

1 Введение

В данной работе решается задача прогнозирования целевой переменной с наличием зависимостей. Трудность задачи в том, что исходные данные имеют высокую размерность и в пространствах целевой и независимой переменных есть скрытые зависимости. Чрезмерно высокая размерность пространств и наблюдаемая множественная корреляция приводят к неустойчивости прогностической модели. Для решения задачи предлагается построить модель, которая бы учитывала обе эти зависимости. Она переводит данные в низкоразмерные пространства и согласование данных происходит в полученном скрытом пространстве.

Метод проекции в скрытое пространство (Projection to Latent Space, PLS) [1, 2] восстанавливает зависимости между двумя наборами данных. Он применяется в биоинформатике, медицине, социальных науках [3–6]. Алгоритм PLS строит матрицу совместного описания признаков и целевой переменной. Полученное пространство является низкоразмерным. Это позволяет получить простую, точную и устойчивую прогностическую модель. Наряду с PLS используется метод канонического анализа корреляций (Canonical Correlation Analysis, CCA) [7]. Метод CCA применяется для поиска зависимостей между двумя наборами данных и получения их низкоразмерного представления [8, 9]. Метод CCA максимизирует корреляции, а метод PLS — ковариации. Обзор и сравнение CCA и PLS приводится в [1]. Линейные методы PLS и CCA игнорируют сложные нелинейные зависимости.

Задачи, в которых между данными существует нелинейная зависимость, описаны в работе [2]. Аппроксимация этой зависимости линейной моделью PLS приводит к неудовлетворительным результатам. Разработаны нелинейные модификации PLS [10–12] и CCA [13, 14]. Например, модель Deep CCA [13] преобразует исходные данные с помощью нейронной сети таким образом, что результирующее представление становится согласованным. Метод Deep CCA используется для генерации текстового описания по изображениям в работе [15].

В данной работе исследуется сложность моделей для данных со сложноорганизованной целевой переменной. Для учёта зависимостей в целевом пространстве используются проекции в скрытое пространства с помощью моделей PLS и CCA. В случае наличия существенно нелинейных зависимостей между независимой и целевой переменными сложность линейной модели оказывается недостаточной. В работе предлагаются методы согласования проекций для нелинейных моделей.

В работе проведено два эксперимента. Первый эксперимент направлен на сравнение эффективности Deep CCA и CCA на задаче классификации зашумленных цифровых изображений MNIST [16]. Во втором эксперименте используется набор данных, полученный делением каждого изображения из MNIST на левую и правую части. На задаче регрессии правой части изображения по левой проводится сравнение нелинейных моделей с применением автоэнкодеров, моделей без преобразования данных и линейного PLS. На основании полученных результатов сделан вывод о точности и сложности нелинейных алгоритмов и о целесообразности использования той или иной модели.

2 Постановка задачи

Пусть дана выборка (\mathbf{X}, \mathbf{Y}) , где $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]^T \in \mathbb{R}^{n \times m}$ — матрица независимых переменных, $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_n]^T \in \mathbb{R}^{n \times k}$ — матрица целевых переменных.

Предполагается, что между \mathbf{X} и \mathbf{Y} существует зависимость

$$\mathbf{Y} = f(\mathbf{X}) + \boldsymbol{\varepsilon}, \quad (1)$$

где $f : \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^{n \times k}$ — функция регрессионной зависимости, $\boldsymbol{\varepsilon}$ — матрица регрессионных ошибок. Необходимо восстановить зависимость f по заданной выборке.

2.1 Линейная регрессия

Предположим, что зависимость (1) линейна. Требуется найти эту зависимость:

$$\mathbf{Y} = f(\mathbf{X}) + \boldsymbol{\varepsilon} = \mathbf{X}\mathbf{W}^T + \boldsymbol{\varepsilon}, \quad (2)$$

где $\mathbf{W} \in \mathbb{R}^{k \times m}$ — матрица параметров модели.

Оптимальные параметры определяются минимизацией функции потерь. Используется квадратичная функция потерь:

$$\mathcal{L}(\mathbf{W}|\mathbf{X}, \mathbf{Y}) = \left\| \mathbf{Y}_{n \times k} - \mathbf{X}_{n \times m} \cdot \mathbf{W}_{m \times k}^T \right\|_2^2 \rightarrow \min_{\mathbf{W}}. \quad (3)$$

Решение задачи (3) имеет вид:

$$\mathbf{W} = \mathbf{Y}^T \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1}.$$

Линейная зависимость столбцов матрицы \mathbf{X} приводит к неустойчивости решения задачи минимизации (3), так как в этом случае матрица $\mathbf{X}^T \mathbf{X}$ является плохо обусловленной. Для борьбы с линейной зависимостью используются методы снижения размерности, путем перехода в низкоразмерное латентное пространство.

Определение 2.1 Параметрическая функция $\varphi_1 : \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^{n \times p}$, переводящая исходные данные в латентное пространство, называется **функцией кодирования**.

Определение 2.2 Функция $\varphi_2 : \mathbb{R}^{n \times p} \rightarrow \mathbb{R}^{n \times m}$, переводящая данные из латентного пространства в исходное, называется **функцией восстановления**.

Определение 2.3 Функция $g : \mathbb{R}^{n \times p} \times \mathbb{R}^{n \times p} \rightarrow \mathbb{R}$, связывающая закономерности в низкоразмерных латентных представлениях, называется **функцией согласования**.

Определение 2.4 Согласование — алгоритмическая процедура максимизации функции согласования.

2.2 Снижение размерности

Коммутативная диаграмма процедуры выбора прогностической модели имеет вид:

$$\begin{array}{ccc}
 \mathbf{X}_{n \times m} & \xrightarrow{f} & \mathbf{Y}_{n \times k} \\
 \varphi_1 \left(\begin{array}{c} \uparrow \varphi_2 \\ \mathbf{T}_{n \times p} \end{array} \right) & & \left(\begin{array}{c} \downarrow \psi_2 \\ \mathbf{U}_{n \times p} \end{array} \right) \psi_1 \\
 & \xleftarrow{g} &
 \end{array} \tag{4}$$

где $\varphi_1 : \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^{n \times p}$ — функция кодирования независимых переменных; $\psi_1 : \mathbb{R}^{n \times k} \rightarrow \mathbb{R}^{n \times p}$ — функция кодирования целевых переменных; $\varphi_2 : \mathbb{R}^{n \times p} \rightarrow \mathbb{R}^{n \times m}$ — функция восстановления независимых переменных; $\psi_2 : \mathbb{R}^{n \times p} \rightarrow \mathbb{R}^{n \times k}$ — функция восстановления целевых переменных; $g : \mathbb{R}^{n \times p} \times \mathbb{R}^{n \times p} \rightarrow \mathbb{R}$ — функция согласования. Матрицы

$$\mathbf{T} = \varphi_1(\mathbf{X}) \in \mathbb{R}^{n \times p}; \quad \mathbf{U} = \psi_1(\mathbf{Y}) \in \mathbb{R}^{n \times p}$$

являются матрицами представлений данных в латентном пространстве низкой размерности.

Оптимальные параметры $\theta_{\varphi_1}^*, \theta_{\psi_1}^*$ для функций кодирования φ_1 и ψ_1 находятся из следующей задачи параметрической оптимизации:

$$(\theta_{\varphi_1}^*, \theta_{\psi_1}^*) = \arg \max_{(\theta_{\varphi_1}, \theta_{\psi_1})} g(\varphi_1(\mathbf{X}; \theta_{\varphi_1}), \psi_1(\mathbf{Y}; \theta_{\psi_1})). \tag{5}$$

Так как параметры функции кодирования подбираются из условия максимизации функции согласования (5), то после перехода в латентное пространство между \mathbf{T} и \mathbf{U} существует зависимость

$$\mathbf{U} = h(\mathbf{T}) + \boldsymbol{\eta}, \tag{6}$$

где $h : \mathbb{R}^{n \times p} \rightarrow \mathbb{R}^{n \times p}$ — функция регрессионной зависимости, $\boldsymbol{\eta}$ — матрица регрессивных ошибок. Оптимальная функция h выбирается минимизацией функции ошибки. Используется квадратичная функция потерь \mathcal{L} на \mathbf{T} и \mathbf{U} :

$$\mathcal{L}(h|\mathbf{T}, \mathbf{U}) = \left\| \mathbf{U} - h(\mathbf{T}) \right\|_2^2 \rightarrow \min_h. \tag{7}$$

Финальная прогностическая модель имеет вид: $\hat{\mathbf{y}} = \psi_2(h(\varphi_1(\mathbf{x})))$, то есть

$$f = \psi_2 \circ h \circ \varphi_1. \tag{8}$$

2.3 Метод главных компонент

Метод главных компонент (PCA) снижает размерность данных и сохраняет максимальную дисперсию. Линейная модель PCA представляет собой ортогональное линейное преобразование исходного признакового пространства в новое пространство меньшей размерности. Первый базисный вектор строится так, чтобы выборочная дисперсия столбцов проекций матрицы \mathbf{X} была максимальной:

$$\mathbf{p} = \arg \max_{\|\mathbf{p}\|_2=1} [\text{var}(\mathbf{X}\mathbf{p})], \quad (9)$$

где $\text{var}(\mathbf{X}\mathbf{p}) = \frac{1}{n}(\mathbf{X}\mathbf{p})^\top \mathbf{X}\mathbf{p}$ обозначает выборочную дисперсию. Последующие базисные векторы находятся итеративно после вычитания проекции на все найденные ранее.

Функция кодирования $\varphi_1 : \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^{n \times p}$ имеет вид:

$$\varphi_1(\mathbf{X}) = \underset{n \times m}{\mathbf{X}} \cdot \underset{m \times p}{\mathbf{P}}^\top, \quad (10)$$

где $\mathbf{P} = [\mathbf{p}_1, \dots, \mathbf{p}_p]$. Метод PCA не согласует независимые переменные и целевые переменные. Из-за этого зависимости в обоих пространствах не учитываются.

2.4 Метод частичных наименьших квадратов

Метод частичных наименьших квадратов восстанавливает связь между двумя наборами данных \mathbf{X} и \mathbf{Y} . Алгоритм проецирует \mathbf{X} и \mathbf{Y} на латентное пространство \mathbb{R}^p меньшей размерности. PLS находит матрицы исходных данных \mathbf{X} и \mathbf{Y} в латентном пространстве \mathbf{T} и \mathbf{U} соответственно. Матрица объектов \mathbf{X} и целевая матрица \mathbf{Y} проецируются на латентное пространство следующим образом:

$$\underset{n \times m}{\mathbf{X}} = \underset{n \times p}{\mathbf{T}} \cdot \underset{p \times m}{\mathbf{P}}^\top + \underset{n \times m}{\mathbf{F}}, \quad (11)$$

$$\underset{n \times k}{\mathbf{Y}} = \underset{n \times p}{\mathbf{U}} \cdot \underset{p \times k}{\mathbf{Q}}^\top + \underset{n \times k}{\mathbf{E}}, \quad (12)$$

где \mathbf{T} и \mathbf{U} — матрицы описания объектов и исходов в латентном пространстве; \mathbf{P} и \mathbf{Q} — матрицы перехода из латентного пространства в исходное; \mathbf{F} , \mathbf{E} — матрицы остатков.

В методе PLS функции кодирования имеют вид:

$$\varphi_1(\mathbf{X}) = \mathbf{X}\mathbf{W}_x, \quad \psi_1(\mathbf{Y}) = \mathbf{Y}\mathbf{W}_y, \quad (13)$$

где матрицы весов $\mathbf{W}_x \in \mathbb{R}^{m \times p}$, $\mathbf{W}_y \in \mathbb{R}^{k \times p}$ находятся путем максимизации функции согласования $g(\mathbf{X}\mathbf{W}_x, \mathbf{Y}\mathbf{W}_y) = \text{Cov}(\mathbf{X}\mathbf{W}_x, \mathbf{Y}\mathbf{W}_y)^2$:

$$(\mathbf{W}_x, \mathbf{W}_y) = \arg \max_{\mathbf{W}_y, \mathbf{W}_y} [\text{Cov}(\mathbf{X}\mathbf{W}_x, \mathbf{Y}\mathbf{W}_y)^2], \quad (14)$$

где $\text{Cov}(\mathbf{X}\mathbf{W}_x, \mathbf{Y}\mathbf{W}_y)$ — выборочная ковариация.

Функции восстановления принимают вид:

$$\varphi_2(\mathbf{T}) = \mathbf{T}\mathbf{P}^\top, \quad \psi_2(\mathbf{U}) = \mathbf{U}\mathbf{Q}^\top. \quad (15)$$

2.5 Канонический анализ корреляций

Канонический анализ корреляций находит два набора базисных векторов $\{\mathbf{w}_{\mathbf{x}_i}\}_{i=1}^p$, $\mathbf{w}_x \in \mathbb{R}^m$ и $\{\mathbf{w}_{\mathbf{y}_i}\}_{i=1}^p$, $\mathbf{w}_y \in \mathbb{R}^k$, один для матрицы \mathbf{X} , другой для матрицы \mathbf{Y} , так что коэффициент корреляции между проекциями переменных на эти базисные векторы был максимальным. Функция согласования для ССА имеет вид

$$g(\mathbf{XW}_x, \mathbf{YW}_y) = \text{corr}(\mathbf{XW}_x, \mathbf{YW}_y), \quad (16)$$

где $\text{corr}(\mathbf{XW}_x, \mathbf{YW}_y)$ – коэффициент корреляции между векторами.

Таким образом, функции кодирования имеют вид

$$\varphi_1(\mathbf{X}) = \mathbf{XW}_x, \quad \psi_1(\mathbf{Y}) = \mathbf{YW}_y, \quad (17)$$

где первые столбцы матриц весов находится, как вектора максимизирующие функцию согласования g . Далее ищутся вектора, максимизирующие g , но с ограничением, что они не коррелируют с первой парой векторов. Процедура продолжается до тех пор, пока число векторов не станет равным p .

2.6 Нелинейный канонический анализ корреляций

Нелинейный канонический анализ корреляций — нелинейная модификация ССА. Метод Deep ССА преобразует исходные данные с помощью нейронной сети таким образом, что результирующее представление становится согласованным. В данной работе рассматриваются следующие нелинейные функции кодирования и декодирования:

$$\begin{aligned} \mathbf{T} &= \varphi_1(\mathbf{X}) = \mathbf{W}_x^L \sigma(\dots \sigma(\mathbf{W}_x^2 \sigma(\mathbf{XW}_x^1)) \dots) \\ \mathbf{U} &= \psi_1(\mathbf{Y}) = \mathbf{W}_y^L \sigma(\dots \sigma(\mathbf{W}_y^2 \sigma(\mathbf{YW}_y^1)) \dots) \\ \mathbf{X} &= \varphi_2(\mathbf{X}) = \mathbf{W}_t^L \sigma(\dots \sigma(\mathbf{W}_t^2 \sigma(\mathbf{TW}_t^1)) \dots) \\ \mathbf{Y} &= \psi_2(\mathbf{Y}) = \mathbf{W}_u^L \sigma(\dots \sigma(\mathbf{W}_u^2 \sigma(\mathbf{UW}_u^1)) \dots) \end{aligned}$$

Каждая функция представляет нейронную сеть с L скрытыми слоями.

Требуется найти такие параметры, что функция согласования g достигает своего максимума

$$g(\mathbf{T}, \mathbf{U}) \rightarrow \max_{\mathbf{W}}, \quad (18)$$

где $\mathbf{W} = \{\{\mathbf{W}_x^i\}_{i=1}^L, \{\mathbf{W}_y^i\}_{i=1}^L, \{\mathbf{W}_t^i\}_{i=1}^L, \{\mathbf{W}_u^i\}_{i=1}^L\}$.

3 Вычислительный эксперимент

Целью вычислительного эксперимента является сравнительный анализ рассматриваемых моделей. Рассматриваются данные, для которых сложность класса линейных методов является неадекватно низкой. Нелинейные модели позволяют получить точный прогноз при адекватной сложности. В рамках вычислительного эксперимента написан программный комплекс для решения поставленных задач [17].

3.1 Анализ нелинейных зависимостей в задаче фильтрации шума

Проведем сравнение качества Deep CCA и CCA на задаче классификации зашумленных цифровых изображений Рис. 1. Для этого используется набор данных MNIST [16], который состоит из 70000 цифровых изображений 28×28 образцов рукописного написания цифр. Предлагается получить два новых набора данных \mathbf{X} и \mathbf{Y} следующим образом. Первый набор получим поворотом исходных изображений на угол в диапазоне $[-\frac{\pi}{4}, \frac{\pi}{4}]$. Для получения второго набора данных для каждой картинки из первого набора данных ставится в соответствие случайным образом картинка с той же цифрой, но с добавлением независимого случайного шума, распределенного равномерно на отрезке $[0, 1]$.

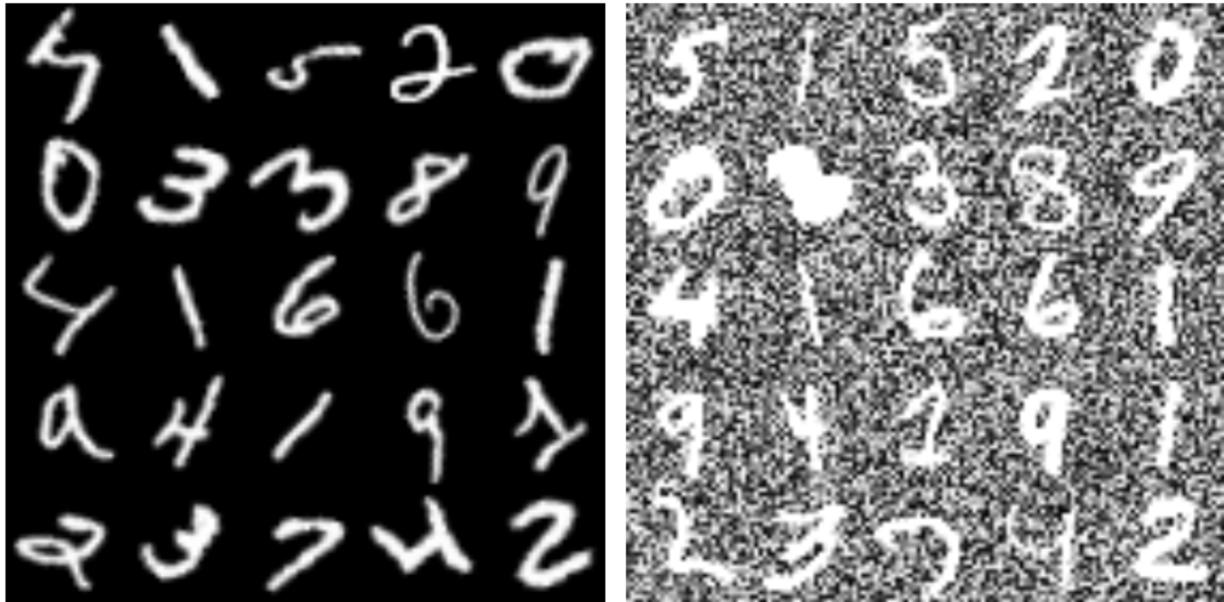


Рис. 1: Зашумленные изображения из набора данных MNIST

Применив к двум новым наборам данных DeepCCA или CCA, мы получаем новое низкоразмерное признаковое пространство, которое игнорирует шумы в исходных данных. Таким образом, получаем функции кодирования φ_1 и ψ_1 для исходных наборов данных. На новых признаках, полученных разными моделями (DeepCCA и CCA), для первого набора данных, то есть на данных после применения функции кодирования φ_1 к первому набору исходных данных, обучим линейный SVM классификатор. Показателем эффективности будет точность классификации линейного SVM на тестовых данных. В случае построения адекватного скрытого пространства полученные образы объектов будут линейно разделимы. Результаты эксперимента приведены в Таблице 1. Модель Deep CCA представляет из себя нейронную сеть с $L = 3$ скрытыми слоями. Точность классификации нелинейной модели существенно выше линейного

алгоритма CCA.

Таблица 1: Точность классификации линейного SVM для алгоритмов Deep CCA и CCA

Скользящий контроль	DeepCCA(L=3)	CCA
Валидация	92.74%	76.21%
Тест	92.14%	76.07%

3.2 Анализ нелинейных моделей для восстановления изображений

Для анализа процедуры согласования проведен вычислительный эксперимент с предложенными нелинейными моделями. Для снижения размерности пространства используются нейросетевые моделимодели автокодировщика с согласованием скрытого пространства (18). В качестве базовых моделей используются модель автокодировщика без согласования скрытых пространств, а также линейный PLS. В качестве исходного набора данных используется набор данных MNIST [16]. Каждое изображение поделено на левую и правую части, как показано на Рис. 2. Модель по левому изображению восстанавливает правое изображение.

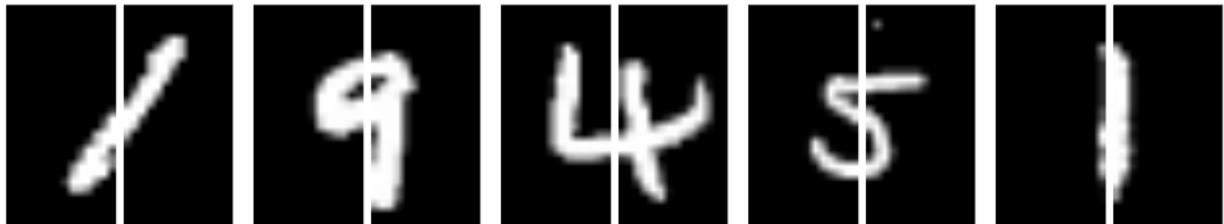


Рис. 2: Набор данных MNIST, в котором каждое изображение разделено пополам

Модель EncNet1 — нейронная сеть с нелинейными функциями активации, которая обучается на данных после преобразования их автоэнкодером. Модель LinNet1 — нейронная сеть с одним линейным слоем, которая также обучается на преобразованных данных. Для EncNet1 и LinNet1 автоэнкодеры для объектов и ответов используют совместную функцию потерь, которая связывает выходы энкодеров. Модели EncNet2 и LinNet2 устроены аналогично EncNet1 и LinNet1 соответственно, но в автоэнкодерах нет совместной функции потерь. Модель DumbNet — нейронная сеть, которая обучается на исходных данных и имеет такую же структуру, что и EncNet, то есть имеет такое же количество слоев и в каждом слое такое же количество нейронном, что и у EncNet.

Для измерения качества моделей выясняется среднеквадратичная ошибка. Примеры восстановленных изображений показаны на Рис. 3. Качество моделей, а также

их сложность представлены в Таблице 2. На Рис. 3 продемонстрировано, что предложенные модели EncNet и LinNet позволяют получить более четкие и различимые изображения в отличие от базовой нелинейной модели DumbNet и линейной модели PLS. Несмотря на заметное улучшение визуального качества изображений, ошибка предложенных моделей выше, чем у модели DumbNet. Авторы предполагают, что это связано с тем, что среднеквадратичная ошибка является неадекватной метрикой в пространстве изображений. Нахождение оптимальной метрики для оценки качества предложенных алгоритмов является одним из возможных направлений развития текущей работы.

Таблица 2: Квадратичная ошибка для нелинейных моделей в задаче восстановления правой части изображения по левой

	EncNet1	LinNet1	EncNet2	LinNet2	DumbNet	PLS
Число параметров, тыс.	283	239	283	239	283	-
Ошибка на тесте	0.147	0.235	0.149	0.236	0.128	0.188

4 Заключение

В работе рассмотрена задача восстановления для сложноорганизованной целевой переменной. Рассмотрены линейные модели согласования образов объектов в скрытом пространстве. В случае наличия сложных нелинейных зависимостей между независимой и целевой переменной сложности линейной модели оказывается недостаточно. Для построения точного прогноза приводятся нелинейные обобщения рассматриваемых линейных методов. В экспериментах на реальных данных изображений рукописных цифр показана адекватность рассматриваемых нелинейных моделей, а также проведен анализ различных способов согласования.

Список литературы

- [1] Rosipal R., Kramer N., Graves A. Overview and recent advances in partial least squares // International Statistical and Optimization Perspectives Workshop «Subspace, Latent Structure and Feature Selection», 2005. P. 34–51.
- [2] Rosipal R. Nonlinear partial least squares: An overview // Chemoinformatics and Advanced Machine Learning Perspectives, 2011. P. 169–189.
- [3] Nguyen D. V., Rocke D. M. Tumor classification by partial least squares using microarray gene expression data // Bioinformatics, 2012. Vol. 18. P. 39–50.
- [4] Worsley K. J. An overview and some new developments in the statistical analysis of pet and fmri data // Human Brain Mapping, 1997. Vol. 5. P. 254–258.

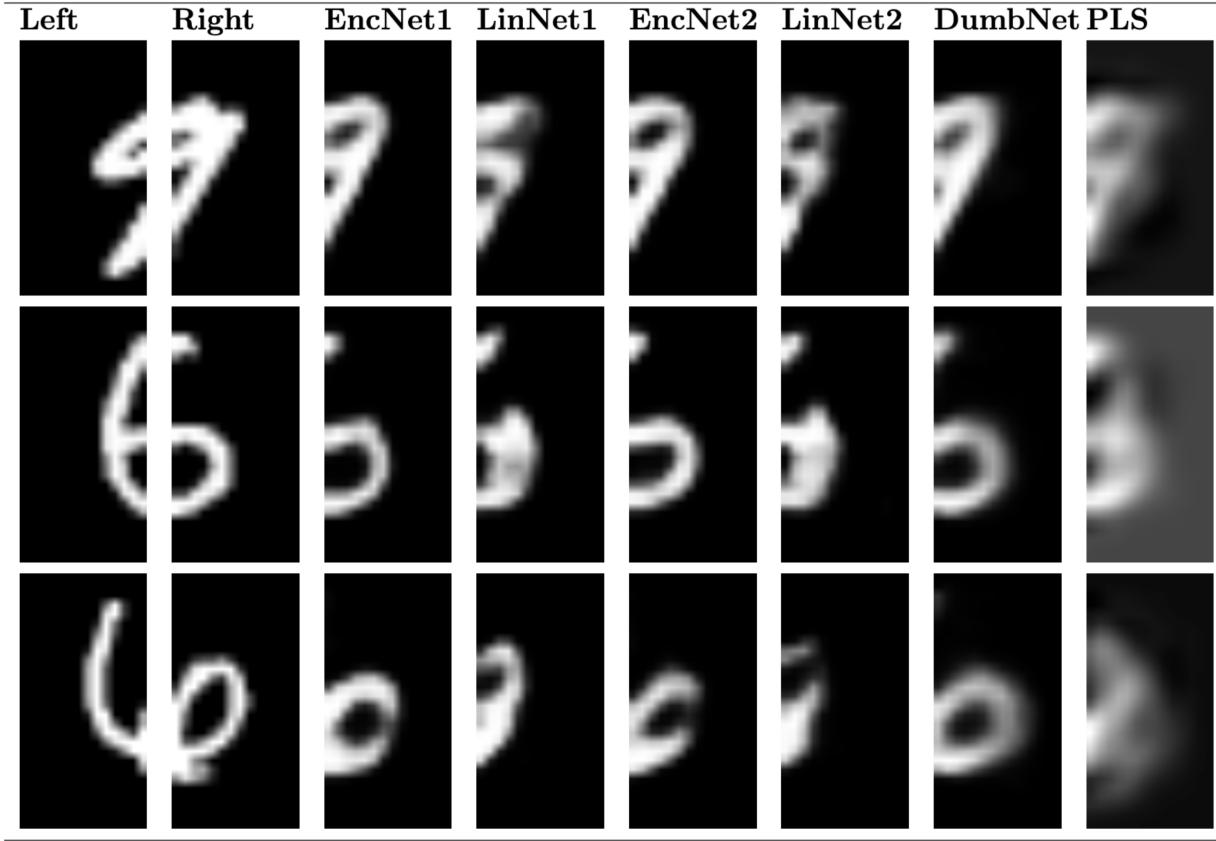


Рис. 3: Пример реконструкции правой части изображения по левой для рассматриваемых моделей

- [5] Hulland J. S. Use of partial least squares (pls) in strategic management research: A review of four recent studies // Strategic Management Journal, 1999. Vol. 20. P. 195–204.
- [6] Shalamu Abudu P. E., Pagano T. C. Application of partial least-squares regression in seasonal streamflow forecasting // Journal of Hydrologic Engineering, 2010. Vol. 15. P. 612–623.
- [7] Szedmak S. R., Hardoon D. R., Shawe-taylor J. R. Canonical correlation analysis: An overview with application to learning methods. // Neural computation, 2004. Vol. 16. P. 2639–2664.
- [8] Schechner Y. Y., Kidron E. Elad M. Pixels that sound // IEEE Computer Society, 2005. P. 88–95.
- [9] Sun S. Ji L., Ye J. A least squares formulation for canonical correlation analysis // International Conference on Machine Learning, 2008. P. 1024–1031.

- [10] *Qin S. J., McAvoy T. J.* Nonlinear pls modeling using neural networks // Computers and Chemical Engineering, 1992. Vol. 16. P. 379–391.
- [11] *Chen D. Z., Yan X. F., Hu S. X.* Chaos-genetic algorithms for optimizing the operating conditions based on rbf-pls model // Computers and Chemical Engineering, 2003. Vol. 27. P. 1393–1404.
- [12] *Hiden M., McKay B., Montague G.* Non-linear partial least squares using genetic programming // Genetic Programming, 1998. P. 128–133.
- [13] *Chen D. Z., Yan X. F., Hu S. X.* Deep canonical correlation analysis // International Conference on Machine Learning, 2013. P. 1247–1255.
- [14] *Lai P. L., Fyfe C.* Kernel and nonlinear canonical correlation analysis // International Journal of Neural Systems, 2000. Vol. 10. P. 365–377.
- [15] *Yan F., Mikolajczyk K.* Deep correlation for matching images and text // Computer Vision and Pattern Recognition, 2015. Vol. 4. P. 3441–3450.
- [16] *LeCun Y., Cortes C., Burges C.* The MNIST dataset of handwritten digits, 1998. Available at: <http://yann.lecun.com/exdb/mnist/index.html>.
- [17] *Yaushev F. Yu, Isachenko R. V.* Модели согласования скрытого пространства в задаче прогнозирования. 2020. Исходный код проекта: <https://github.com/Fyaushev/2020-Project-72>.