

Dimensionality reduction for time series decoding and forecasting problems*

R. V. Isachenko¹, M. R. Vladimirova², V. V. Strijov³

Abstract: (Решается задача обнаружения зависимостей в прогнозируемой переменной.) The problem is to detect the relation between independent and target variables. (Вместо построения прогноза одного момента времени предлагается прогнозировать многомерный вектор прогноза.) We propose to predict multidimensional target vector instead of predicting one timestamp point. (Рассматривается линейная модель метода частных наименьших квадратов.) We consider the linear model of partial least squares (PLS). (Описан метод нахождения матрицы совместного описания для исходных матриц объектов и ответов.) The described method finds a matrix of joint description for the design matrix and the outcome matrix. (Найденное описание является низкоразмерным, что позволяет построить устойчивую, простую модель.) New description is low-dimensional and allows to build simple, stable model. (Проводится вычислительный эксперимент на реальных данных объемов потребления электроэнергии и данных сигналов кортикограмм.) We conducted computational experiment on real data of energy consumption and electrocorticograms signals (ECoG).

Keywords: time series decoding, forecasting, partial least squares, dimensionality reduction

1 Introduction

(В работе рассматривается задача восстановления зависимости между независимой и прогнозируемой переменными.) Paper investigate the problem of dependency recovering between input data and model outcome. (Предлагаемая модель позволяет восстанавливать зависимость в случае многомерной прогнозируемой переменной.) Suggested

*The work was financially supported by the Russian Foundation for Basic Research (project 16-07-01155).

¹Moscow Institute of Physics and Technology, isa-ro@yandex.ru

²Moscow Institute of Physics and Technology, vladimirova.maria@phystech.edu

³A. A. Dorodnicyn Computing Centre, Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, strijov@ccas.ru

model is suitable for predicting multidimensional target variable. (В случае задачи прогнозирования пространства объектов и ответов имеют одну и ту же природу.) In the case of forecasting problem the object and target spaces have the same nature. (Для построения модели по исходному временному ряду строятся авторегрессионные матрицы объектов и ответов.) To build the model we need to construct autoregressive matrices for input objects and target variable. (Объектом является локальная история сигнала, ответом — значение сигнала в следующие моменты времени.) The object is the local signal history, the outcome is signal values in the next timestamps. (Авторегрессионная модель предполагает, что значение сигнала в данный момент времени линейно зависит от предыдущих значений этого же сигнала.) Autoregressive model makes a consumption that current signal values depends linearly on the previous signal values.

(В задаче декодирования временных рядов пространство объектов имеет значимо большую размерность.) (В этом случае пространства объектов и ответов имеют разную природу, ответом является отклик системы на входной сигнал.) In the case of time series decoding problem object and target spaces are different in nature, the outcome is a system response to the input signal. (Авторегрессионная матрица объектов содержит локальную историю входного сигнала, а авторегрессионная матрица ответов содержит локальную историю отклика.) Autoregressive design matrix contains local history of input signal. Autoregressive target matrix contains the local history of the response.

(Исходное признаковое пространство является многомерным.) The object space in time series decoding problems is high dimensional. (Избыточная размерность признакового пространства приводит к неустойчивости модели.) Excessive dimensionality of feature description leads to the model instability. (Для решения этой проблемы используются методы отбора признаков [1, 2].) To solve this problem the feature selection procedures are used [1, 2].

(В работе исследуется метод частных наименьших квадратов (PLS) [3–5].) Paper considers the partial least squares regression (PLS) model [3–5]. (Метод частных наименьших квадратов снижает размерности матрицы признаков и выделяет линейные комбинации признаков, которые оказывают наибольшее влияние на вектор ответов.) PLS model reduces the dimension of input data and extracts linear combination of features which have the greatest impact on the response vector. (Выделение признаков происходит итеративно, в порядке уменьшения их влияния на вектор ответов.) Feature extraction is iterative process in order of decreasing the influence on the response vector. (Рассматриваются только значимые комбинации признаков, незначительно потеряв в точности прогноза. Методы PLS регрессии подробно описаны в работах [6–8].) PLS regression methods are described in detail in [6–8]. (Разницу между методом PLS и связанными с ним подходами, различные разновидности регрессии PLS можно найти в [9].) The difference between various PLS approaches, the different kinds of PLS regression could be found in [9].

(Современное состояние области и обзор нелинейных модификаций метода PLS описаны [10].) The current state of the field and an overview of nonlinear PLS method modifications are described in [10]. Нелинейное расширение метода PLS регрессии впервые введено в [11]. Nonlinear PLS method extension was introduced in [11]. (В литерату-

49 ре были разработаны различные модификации PLS.) There have been developed the
50 variety of PLS modifications. (Предложены нелинейные методы PLS, основанные на
51 различных моделях: сглаживающих сплайнах [12], нейронных сетях [13], базисных
52 радиальных функциях [14], генетическом алгоритме [15].) The proposed nonlinear
53 PLS methods are based on smoothing splines [12], neural networks [13], radial basis
54 functions [14], genetic algorithm [15].

55 (Результатом отбора признаков является снижение размерности задачи и повы-
56 шение устойчивости моделей без существенной потери точности прогноза.) The result
57 of feature selection is the dimensionality reduction and increasing model stability without
58 significant loss of the prediction quality. (Метод решения задачи применяется к двух
59 наборам данных, имеющих избыточную информацию.) The proposed method is used
60 on two datasets with redundant input and target spaces. (Первый набор данных — по-
61 часовые временные ряды объёмов потребления электроэнергии. Временные ряды со-
62 ответствуют периоду 1999-2004 годов.) The first dataset consists of hourly time series
63 of energy consumption. Time series were collected in Poland from 1999 to 2004.

64 Второй набор данных взят из проекта NeuroTycho [16], в котором проектируется
65 нейрокомпьютерный интерфейс (BCI) [17, 18] для обмена информацией между моз-
66 гом и электронным устройством. Система BCI повышает умственные и физические
67 способности пользователя, обеспечивая прямую связь между мозгом и компьюте-
68 ром [19]. BCI восстанавливает поврежденную функциональность пациентов с нару-
69 шениями двигательного или когнитивного развития. Цель анализа моторных изоб-
70 ражений заключается в распознавании предполагаемых движений по записанной ак-
71 тивности мозга. Существуют различные методы измерения кортикальных данных для
72 BCI описанные в [20, 21]. В работе рассматриваются сигналы электрокортикографии
73 (ECoG) [22]. ECoG, как и другие инвазивные методы, обеспечивает более стабильную
74 запись и лучшее разрешение в временных и пространственных областях, чем ее неин-
75 вазивные аналоги. Данные ECoG являются многомерными и измерения коррелируют
76 как во временной, так и в пространственной областях. Задача состоит в предсказании
77 траектории движения руки по временным рядам напряжения кортикальной активнос-
78 ти. Сигналы ECoG измеряются через 32 канала, в то время как субъект перемещает
79 руку.

80 В работах по прогнозированию сложных пространственных временных рядов про-
81 гноз выполняется поточечно. При необходимости прогнозировать несколько точек од-
82 новременно задачу предлагается решать последовательно вычисляя прогноз по точ-
83 кам. При этом используются предыдущие спрогнозированные значения для получе-
84 ния последующих. Предлагаемый метод позволяет получать временной ряд прогнози-
85 руемых значений одновременно с учётом скрытых зависимостей не только в простран-
86 стве объектов, но и в пространстве ответов с согласованием этих двух зависимостей.
87 Предлагаемый метод позволил существенно снизить размерность признакового про-
88 странства.

2 Постановка задачи

Задана выборка $\mathfrak{D} = (\mathbf{X}, \mathbf{Y})$, где $\mathbf{X} \in \mathbb{R}^{m \times n}$ — матрица объектов, $\mathbf{Y} \in \mathbb{R}^{m \times r}$ — матрица ответов. Способ построения выборки под определенную прикладную задачу описан в разделе *Вычислительный эксперимент*.

Предполагается, что между объектами $\mathbf{x} \in \mathbb{R}^n$ и ответами $\mathbf{y} \in \mathbb{R}^r$ существует линейная зависимость

$$\underset{1 \times r}{\mathbf{y}} = \underset{1 \times n}{\mathbf{x}} \cdot \underset{n \times r}{\boldsymbol{\Theta}} + \underset{1 \times r}{\boldsymbol{\varepsilon}}, \quad (1)$$

где $\boldsymbol{\Theta}$ — матрица параметров модели, а $\boldsymbol{\varepsilon}$ — вектор регрессионных остатков.

Необходимо по известной выборке \mathfrak{D} восстановить матрицу параметров модели (1). Оптимальные параметры находятся минимизацией функции ошибки. Введем квадратичную функцию ошибки S на выборке \mathfrak{D} :

$$S(\boldsymbol{\Theta}|\mathfrak{D}) = \left\| \underset{m \times n}{\mathbf{X}} \cdot \underset{n \times r}{\boldsymbol{\Theta}} - \underset{m \times r}{\mathbf{Y}} \right\|_2^2 = \sum_{i=1}^m \left\| \underset{1 \times n}{\mathbf{x}_i} \cdot \underset{n \times r}{\boldsymbol{\Theta}} - \underset{1 \times r}{\mathbf{y}_i} \right\|_2^2 \rightarrow \min_{\boldsymbol{\Theta}}. \quad (2)$$

Линейная зависимость столбцов матрицы \mathbf{X} приводит к неустойчивому решению задачи оптимизации (2). Для устранения линейной зависимости применяются методы отбора признаков.

3 Метод частных наименьших квадратов

Для устранения линейной зависимости и снижения размерности пространства применяется метод главных компонент PCA. Основным недостатком данного метода является то, что он не учитывает взаимосвязь между объектами и ответами. Метод частных наименьших квадратов PLS проецирует матрицу объектов \mathbf{X} и матрицу ответов \mathbf{Y} в латентное пространство \mathbb{R}^l меньшей размерности ($l < r < n$). Алгоритм PLS находит в латентном пространстве матрицу $\mathbf{T} \in \mathbb{R}^{m \times l}$, наилучшим образом описывающую исходные матрицы \mathbf{X} и \mathbf{Y} .

Матрица объектов \mathbf{X} и матрица ответов \mathbf{Y} проецируются в латентное пространство следующим образом:

$$\underset{m \times n}{\mathbf{X}} = \underset{m \times l}{\mathbf{T}} \cdot \underset{l \times n}{\mathbf{P}^\top} + \underset{m \times n}{\mathbf{F}} = \sum_{k=1}^l \underset{m \times 1}{\mathbf{t}_k} \cdot \underset{1 \times n}{\mathbf{p}_k^\top} + \underset{m \times n}{\mathbf{F}}, \quad (3)$$

$$\underset{m \times r}{\mathbf{Y}} = \underset{m \times l}{\mathbf{T}} \cdot \underset{l \times r}{\mathbf{Q}^\top} + \underset{m \times r}{\mathbf{E}} = \sum_{k=1}^l \underset{m \times 1}{\mathbf{t}_k} \cdot \underset{1 \times r}{\mathbf{q}_k^\top} + \underset{m \times r}{\mathbf{E}}, \quad (4)$$

где \mathbf{T} — матрица совместного описания объектов и ответов в латентном пространстве, причём столбцы матрицы \mathbf{T} ортогональны; \mathbf{P} , \mathbf{Q} — матрицы перехода из латентного пространства в исходные пространства; \mathbf{E} , \mathbf{F} — матрицы невязок.

113 Псевдокод метода регрессии PLS приведен в алгоритме 1. Алгоритм итеративно на
 114 каждом из l шагов вычисляет по одному столбцу \mathbf{t}_k , \mathbf{p}_k , \mathbf{q}_k матриц \mathbf{T} , \mathbf{P} , \mathbf{Q} соответ-
 115 ственно. После вычисления следующего набора векторов из матриц \mathbf{X} , \mathbf{Y} вычитаются
 116 очередные одноранговые аппроксимации. Первым шагом необходимо произвести
 117 нормировку столбцов исходных матриц (вычесть среднее и разделить на стандарт-
 118 ное отклонение). На этапе тестирования необходимо провести нормировку тестовых
 119 данных, вычислить предсказание модели 1, а затем провести обратную нормировку.

Algorithm 1 Алгоритм PLSR

Require: $\mathbf{X}, \mathbf{Y}, l$;

Ensure: $\mathbf{T}, \mathbf{P}, \mathbf{Q}$;

```

1: нормировать матрицы  $\mathbf{X}$  и  $\mathbf{Y}$  по столбцам
2: инициализировать  $\mathbf{u}_0$  (первый столбец матрицы  $\mathbf{Y}$ )
3:  $\mathbf{X}_1 = \mathbf{X}; \mathbf{Y}_1 = \mathbf{Y}$ 
4: for  $k = 1, \dots, l$  do
5:   repeat
6:      $\mathbf{w}_k := \mathbf{X}_k^\top \mathbf{u}_{k-1} / (\mathbf{u}_{k-1}^\top \mathbf{u}_{k-1}); \quad \mathbf{w}_k := \frac{\mathbf{w}_k}{\|\mathbf{w}_k\|}$ 
7:      $\mathbf{t}_k := \mathbf{X}_k \mathbf{w}_k$ 
8:      $\mathbf{c}_k := \mathbf{Y}_k^\top \mathbf{t}_k / (\mathbf{t}_k^\top \mathbf{t}_k); \quad \mathbf{c}_k := \frac{\mathbf{c}_k}{\|\mathbf{c}_k\|}$ 
9:      $\mathbf{u}_k := \mathbf{Y}_k \mathbf{c}_k$ 
10:   until  $\mathbf{t}_k$  не стабилизируется
11:    $\mathbf{p}_k := \mathbf{X}_k^\top \mathbf{t}_k / (\mathbf{t}_k^\top \mathbf{t}_k), \quad \mathbf{q}_k := \mathbf{Y}_k^\top \mathbf{t}_k / (\mathbf{t}_k^\top \mathbf{t}_k)$ 
12:    $\mathbf{X}_{k+1} := \mathbf{X}_k - \mathbf{t}_k \mathbf{p}_k^\top$ 
13:    $\mathbf{Y}_{k+1} := \mathbf{Y}_k - \mathbf{t}_k \mathbf{q}_k^\top$ 

```

120 Вектора \mathbf{t}_k и \mathbf{u}_k из внутреннего цикла алгоритма 1 содержат информацию о мат-
 121 рице объектов \mathbf{X} и матрице ответов \mathbf{Y} соответственно. Блоки из шагов (6)-(7) и шагов
 122 (8)-(9) — аналоги алгоритма PCA для матриц \mathbf{X} и \mathbf{Y} [5]. Последовательное выполне-
 123 ние блоков позволяет учесть взаимную связь между матрицами \mathbf{X} и \mathbf{Y} .

124 Теоретическое обоснование алгоритма PLS следует из следующих утверждений.

125 **Утверждение 1.** *Наилучшее описание матриц \mathbf{X} и \mathbf{Y} с учётом их взаимосвязи*
 126 *достигается при максимизации ковариации между векторами \mathbf{t}_k и \mathbf{u}_k .*

127 Утверждение следует из равенства

$$\text{cov}(\mathbf{t}_k, \mathbf{u}_k) = \text{corr}(\mathbf{t}_k, \mathbf{u}_k) \cdot \sqrt{\text{var}(\mathbf{t}_k)} \cdot \sqrt{\text{var}(\mathbf{u}_k)}.$$

128 Максимизация дисперсий векторов \mathbf{t}_k и \mathbf{u}_k отвечает за сохранение информации об
 129 исходных матрицах, корреляция между векторами отвечает взаимосвязи между \mathbf{X}
 130 и \mathbf{Y} . ■

131 Во внутреннем цикле алгоритма вычисляются нормированные вектора весов \mathbf{w}_k и
 132 \mathbf{c}_k . Из данных векторов строятся матрицы весов \mathbf{W} и \mathbf{C} соответственно.

133 **Утверждение 2.** В результате выполнения внутреннего цикла вектора \mathbf{w}_k и \mathbf{c}_k
 134 будут собственными векторами матриц $\mathbf{X}_k^\top \mathbf{Y}_k \mathbf{Y}_k^\top \mathbf{X}_k$ и $\mathbf{Y}_k^\top \mathbf{X}_k \mathbf{X}_k^\top \mathbf{Y}_k$, соответству-
 135 ющими максимальным собственным значениям.

$$\begin{aligned}\mathbf{w}_k &\propto \mathbf{X}_k^\top \mathbf{u}_{k-1} \propto \mathbf{X}_k^\top \mathbf{Y}_k \mathbf{c}_{k-1} \propto \mathbf{X}_k^\top \mathbf{Y}_k \mathbf{Y}_k^\top \mathbf{t}_{k-1} \propto \mathbf{X}_k^\top \mathbf{Y}_k \mathbf{Y}_k^\top \mathbf{X}_k \mathbf{w}_{k-1}, \\ \mathbf{c}_k &\propto \mathbf{Y}_k^\top \mathbf{t}_k \propto \mathbf{Y}_k^\top \mathbf{X}_k \mathbf{w}_k \propto \mathbf{Y}_k^\top \mathbf{X}_k \mathbf{X}_k^\top \mathbf{u}_{k-1} \propto \mathbf{Y}_k^\top \mathbf{X}_k \mathbf{X}_k^\top \mathbf{Y}_k \mathbf{c}_{k-1},\end{aligned}$$

136 где символ \propto означает равенство с точностью до мультипликативной константы.
 137

138 Утверждение следует из того факта, что правила обновления векторов \mathbf{w}_k , \mathbf{c}_k сов-
 139 падают с итерацией алгоритма поиска максимального собственного значения. Данный
 140 алгоритм основан на следующем факте.

141 Если матрица \mathbf{A} диагонализуема, \mathbf{x} — некоторый вектор, то

$$\lim_{k \rightarrow \infty} \mathbf{A}^k \mathbf{x} = \lambda_{\max}(\mathbf{A}) \cdot \mathbf{v}_{\max},$$

142 где $\lambda_{\max}(\mathbf{A})$ — максимальное собственное значение матрицы \mathbf{A} , \mathbf{v}_{\max} — собственный
 143 вектор матрицы \mathbf{A} , соответствующий $\lambda_{\max}(\mathbf{A})$. ■

144 **Утверждение 3.** Обновление векторов по шагам (6)–(9) алгоритма 1 соответству-
 145 ет максимизации ковариации между векторами \mathbf{t}_k и \mathbf{u}_k .

Максимальная ковариация между векторами \mathbf{t}_k и \mathbf{u}_k равна максимальному соб-
 ственному значению матрицы $\mathbf{X}_k^\top \mathbf{Y}_k \mathbf{Y}_k^\top \mathbf{X}_k$:

$$\begin{aligned}\max_{\mathbf{t}_k, \mathbf{u}_k} \text{cov}(\mathbf{t}_k, \mathbf{u}_k)^2 &= \max_{\substack{\|\mathbf{w}_k\|=1 \\ \|\mathbf{c}_k\|=1}} \text{cov}(\mathbf{X}_k \mathbf{w}_k, \mathbf{Y}_k \mathbf{c}_k)^2 = \max_{\substack{\|\mathbf{w}_k\|=1 \\ \|\mathbf{c}_k\|=1}} \text{cov}\left(\mathbf{c}_k^\top \mathbf{Y}_k^\top \mathbf{X}_k \mathbf{w}_k\right)^2 = \\ &= \max_{\|\mathbf{w}_k\|=1} \text{cov}\left\|\mathbf{Y}_k^\top \mathbf{X}_k \mathbf{w}_k\right\|^2 = \max_{\|\mathbf{w}_k\|=1} \mathbf{w}_k^\top \mathbf{X}_k^\top \mathbf{Y}_k \mathbf{Y}_k^\top \mathbf{X}_k \mathbf{w}_k = \\ &= \lambda_{\max}\left(\mathbf{X}_k^\top \mathbf{Y}_k \mathbf{Y}_k^\top \mathbf{X}_k\right),\end{aligned}$$

146 где $\lambda_{\max}(\mathbf{A})$ — максимальное собственное значение матрицы \mathbf{A} . Применяя утвержде-
 147 ние 2, получаем требуемое. ■

После завершения внутреннего цикла на шаге (11) вычисляются вектора \mathbf{p}_k , \mathbf{q}_k проецированием столбцов матриц \mathbf{X}_k и \mathbf{Y}_k на вектор \mathbf{t}_k . Для перехода на следующий шаг необходимо вычесть из матриц \mathbf{X}_k и \mathbf{Y}_k одноранговые аппроксимации $\mathbf{t}_k \mathbf{p}_k^\top$ и $\mathbf{t}_k \mathbf{q}_k^\top$

$$\mathbf{X}_{k+1} = \mathbf{X}_k - \mathbf{t}_k \mathbf{p}_k^\top = \mathbf{X} - \sum_k \mathbf{t}_k \mathbf{p}_k^\top,$$

$$\mathbf{Y}_{k+1} = \mathbf{Y}_k - \mathbf{t}_k \mathbf{q}_k^\top = \mathbf{Y} - \sum_k \mathbf{t}_k \mathbf{q}_k^\top.$$

Тогда каждый следующий вектор \mathbf{t}_k оказывается ортогонален всем векторам \mathbf{t}_i , $i = 1, \dots, k$.

На Рис. 1 продемонстрирован результат работы алгоритма PLS для случая, когда размерности пространств объектов, ответов и латентного пространства равны 2 ($n = r = l = 2$). Синими и зелёными точками изображены строки матриц \mathbf{X} и \mathbf{Y} . Точки были сгенерированы из нормального распределения с нулевым матожиданием. Красным контуром показаны линии уровня матриц ковариаций распределений. Черным изображены единичные окружности. Красные стрелки соответствуют главным компонентам. Черные стрелки соответствуют векторам матриц \mathbf{W} и \mathbf{C} алгоритма PLS. Вектора \mathbf{t}_k и \mathbf{u}_k равны проекциям матриц \mathbf{X}_k и \mathbf{Y}_k на вектора \mathbf{w}_k и \mathbf{c}_k соответственно и изображены черными плюсами. Учёт взаимной связи между матрицами \mathbf{X} и \mathbf{Y} отклоняет вектора \mathbf{w}_k и \mathbf{c}_k от направления главных компонент. Вектора \mathbf{w}_k отклоняются незначительно. На первой итерации \mathbf{c}_1 близок к pc_1 , но вектора \mathbf{c}_k , найденные на следующих итерациях могут оказаться сильно коррелированными. Это происходит в следствие того, что из матрицы \mathbf{Y} на каждом шаге вычитается одноранговая аппроксимация, найденная в пространстве матрицы \mathbf{X}_k .

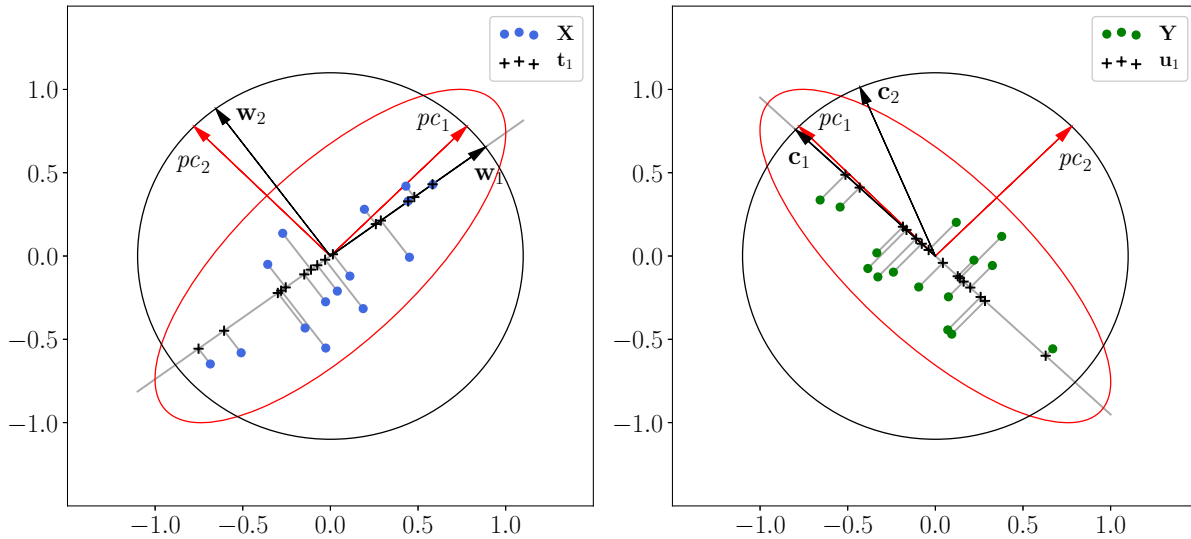


Рис. 1: Иллюстрация алгоритма PLS

Для получения прогнозов модели и нахождения параметров модели домножим справа формулу (3) на матрицу \mathbf{W} . Строки матрицы невязок \mathbf{E} ортогональны столбцам матрицы \mathbf{W} , поэтому

$$\mathbf{XW} = \mathbf{TP}^T \mathbf{W}.$$

Линейное преобразование между объектами в исходном и латентном пространстве имеет вид

$$\mathbf{T} = \mathbf{XW}^*, \quad (5)$$

где $\mathbf{W}^* = \mathbf{W}(\mathbf{P}^T \mathbf{W})^{-1}$.

Матрица параметров модели 1 находится из уравнений (4), (5)

$$\mathbf{Y} = \mathbf{T}\mathbf{Q}^\top + \mathbf{E} = \mathbf{X}\mathbf{W}^*\mathbf{Q}^\top + \mathbf{E} = \mathbf{X}\mathbf{\Theta} + \mathbf{E}.$$

Таким образом, параметры модели (1) равны

$$\mathbf{\Theta} = \mathbf{W}(\mathbf{P}^\top \mathbf{W})^{-1} \mathbf{Q}^\top. \quad (6)$$

После нахождения параметров модели на этапе тестирования для нахождения предсказания модели необходимо

- нормировать тестовые данные;
- вычислить предсказание модели с помощью линейного преобразования с матрицей $\mathbf{\Theta}$ из (6);
- провести обратную нормировку.

4 Вычислительный эксперимент

Временные ряды энергии состоят из почасовых записей (всего 52512 наблюдений), а погодные измерения проводились раз в день и содержат 2188 наблюдений. Строка матрицы \mathbf{X} — локальная история сигнала за одну неделю $n = 24 \times 7$. Строка матрицы \mathbf{Y} — локальный прогноз потребления электроэнергии в следующие 24 часа $r = 24$. В этом случае матрицы \mathbf{X} и \mathbf{Y} являются авторегрессионными матрицами (можно ссылку).

В случае данных ECoG матрица \mathbf{X} состоит из пространственно-временного спектрального представления временных рядов напряжения, а матрица \mathbf{Y} содержит информацию о положении руки. Процесс генерации матрицы \mathbf{X} из значений напряжения описан в [23]. Признаковое описание в каждый момент времени имеет размерность 864, положение руки описывается координатами по трём осям. Пример данных напряжения с разных каналов и соответствующих пространственных координат руки представлен на Рис. 2. Для прогнозирования движения руки используется авторегрессионный подход. Один объект состоит из признакового описания в несколько отсчётов времени. Ответом является положение руки в нескольких следующих моментах времени. Требуется предсказать положение руки в нескольких следующих моментах времени.

Введём среднеквадратичную ошибку для некоторых матриц $\mathbf{A} = [a_{ij}]$ и $\mathbf{B} = [b_{ij}]$

$$\text{MSE}(\mathbf{A}, \mathbf{B}) = \sum_{i,j} (a_{ij} - b_{ij})^2.$$

Для оценивания качества аппроксимации вычисляется значение нормированной среднеквадратичной ошибки

$$\text{NMSE}(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{\text{MSE}(\mathbf{Y}, \hat{\mathbf{Y}})}{\text{MSE}(\mathbf{Y}, \bar{\mathbf{Y}})}, \quad (7)$$

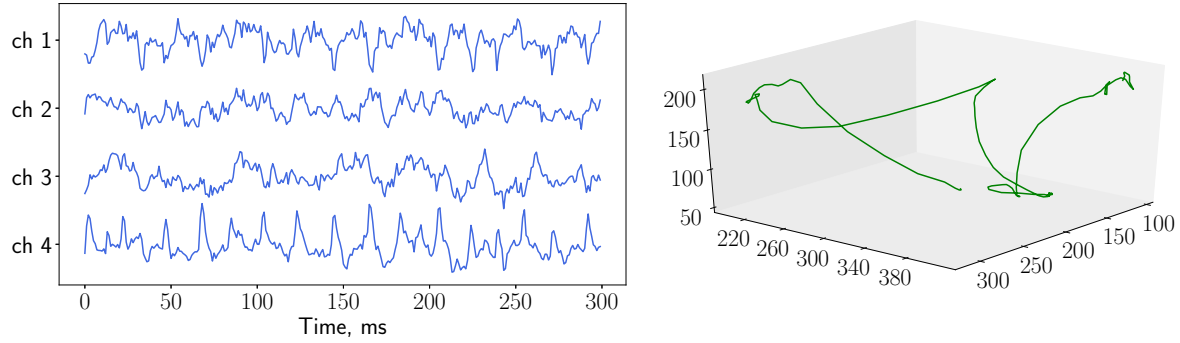


Рис. 2: Пример данных ECoG. Слева изображены данные напряжения, снятые с нескольких каналов, справа — координаты руки по трём осям.

198 где $\hat{\mathbf{Y}}$ — прогноз модели, $\bar{\mathbf{Y}}$ — константный прогноз средним значением по столбцам
 199 матрицы.

200 4.1 Данные потребления электроэнергии

201 Для нахождения оптимальной размерности l латентного пространства все данные
 202 потребления электроэнергии были разбиты на обучающую и валидационную части.
 203 Обучающая выборка состоит из 700 объектов, валидационная из 370. Зависимость
 204 нормированной квадратичной ошибки (7) от размерности l латентного пространства
 205 представлена на Рис. 3. Сначала ошибка резко падает при увеличении размерности
 206 скрытого пространства, а затем меняется незначительно.

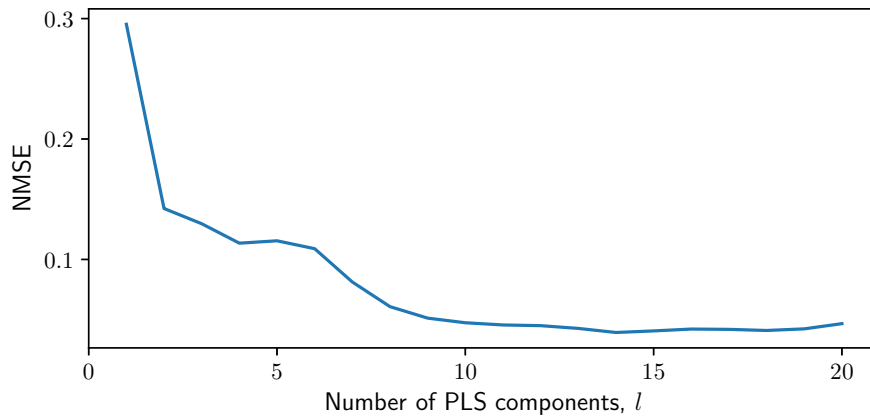


Рис. 3: Прогноз потребления электроэнергии алгоритмом PLS при размерности латентного пространства $l=14$

207 Минимальная ошибка наблюдается при $l = 14$. Построим прогноз потребления
 208 электроэнергии при данном l . Результат аппроксимации изображен на Рис. 4. Алго-

209 ритм PLS восстановил авторегрессионную зависимость и обнаружил дневную сезон-
210 ность.

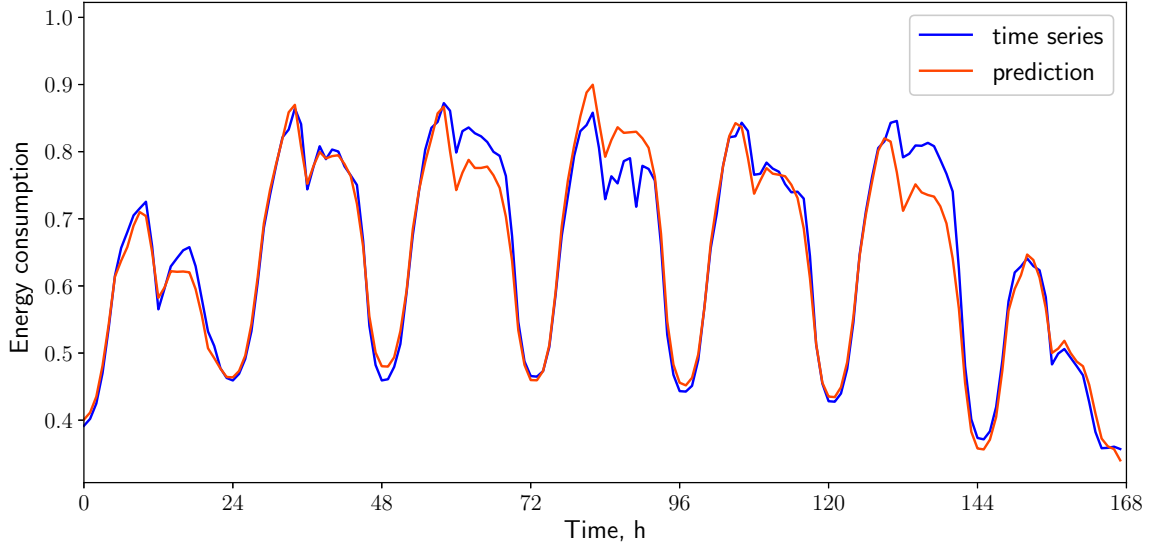


Рис. 4: Зависимость ошибки от размерности латентного пространства для данных потребления электроэнергии

211 4.2 Данные электрокортикограммы

212 На Рис. 5 представлена зависимость нормированной квадратичной ошибки (7) от раз-
213 мерности латентного пространства. Ошибка аппроксимации меняется незначительно
214 при $l > 5$. Таким образом совместное описание пространственно-временного спек-
215 трального представления объектов и пространственного положения руки может быть
216 представлено вектором размерности $l \ll n$. Зафиксируем $l = 5$. Пример аппроксима-
217 ции положения руки изображен на Рис. 6. Сплошными линиями изображены истин-
218 ные координаты руки по всем осям, пунктирными линиями показана аппроксимация
219 методом PLS.

220 5 Заключение

221 В данной работе предложен подход к решению задачи декодирования временных ря-
222 дов и прогнозирования. Алгоритм частичных наименьших квадратов позволяет по-
223 строить векторный прогноз целевой переменной. Латентное пространство содержит
224 информацию об объектах и ответах и снижает размерности исходных матриц на по-
225 рядки. Вычислительный эксперимент продемонстрировал применимость предложен-
226 ного метода к задачам прогнозирования временных рядов объёмов потребления элек-
227 троэнергии и проектирования нейрокомпьютерного интерфейса.

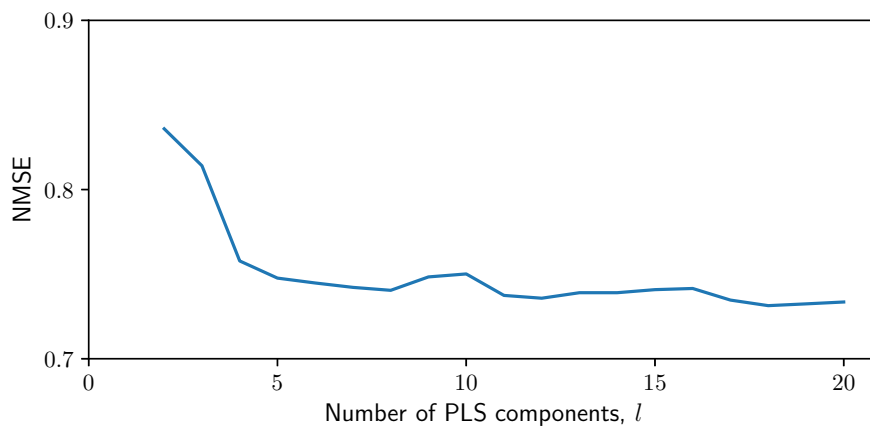


Рис. 5: Зависимость ошибки от размерности латентного пространства для данных ECoG

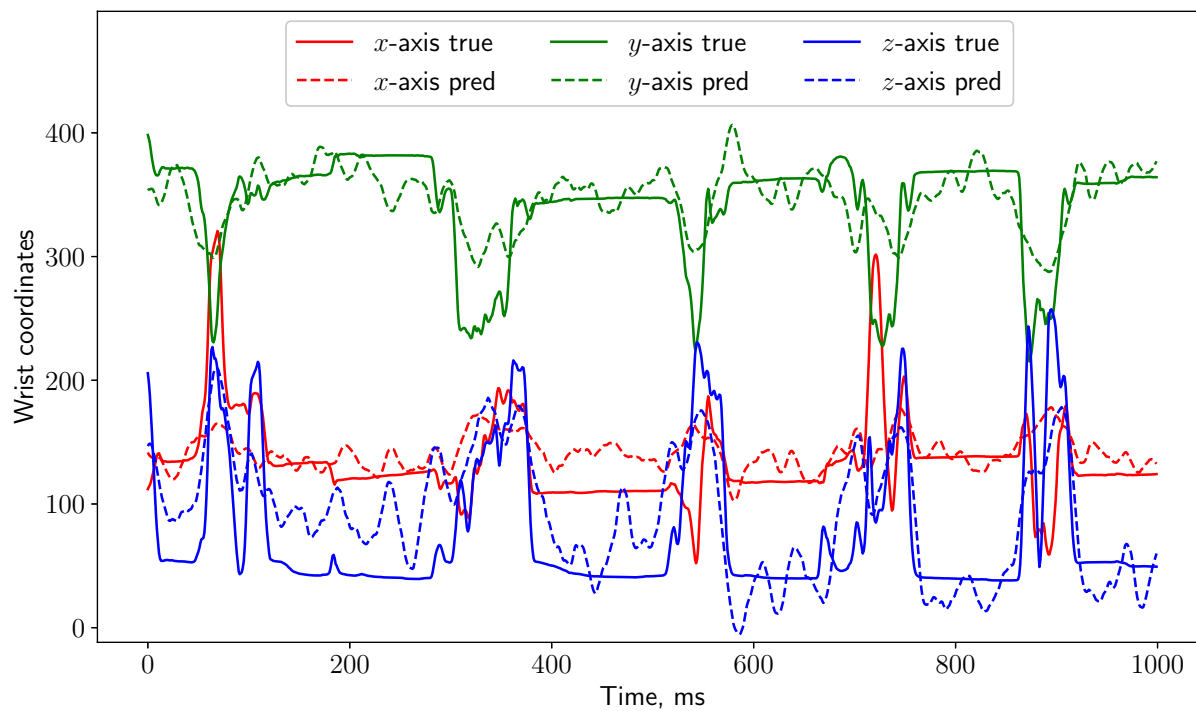


Рис. 6: Прогноз движения руки данных ECoG алгоритмом PLS при размерности латентного пространства $l = 5$

Список литературы

- [1] AM Katrutsa and VV Strijov. Stress test procedure for feature selection algorithms. *Chemometrics and Intelligent Laboratory Systems*, 142:172–183, 2015.

- [2] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P Trevino, Jiliang Tang, and Huan Liu. Feature selection: A data perspective. *arXiv preprint arXiv:1601.07996*, 2016.
- [3] Jacob A Wegelin et al. A survey of partial least squares (pls) methods, with emphasis on the two-block case. *University of Washington, Department of Statistics, Tech. Rep*, 2000.
- [4] Hervé Abdi. Partial Least Squares (PLS) Regression. *Encyclopedia for research methods for the social sciences*, pages 792–795, 2003.
- [5] Paul Geladi and Bruce R Kowalski. Partial least-squares regression: a tutorial. *Analytica chimica acta*, 185:1–17, 1986.
- [6] Paul Geladi. Notes on the history and nature of partial least squares (PLS) modelling. *Journal of Chemometrics*, 2(January):231–246, 1988.
- [7] Agnar Höskuldsson. PLS regression. *Journal of Chemometrics*, 2(August 1987):581–591, 1988.
- [8] Sijmen De Jong. Simpls: an alternative approach to partial least squares regression. *Chemometrics and intelligent laboratory systems*, 18(3):251–263, 1993.
- [9] Roman Rosipal and Nicole Kramer. Overview and Recent Advances in Partial Least Squares. *C. Saunders et al. (Eds.): SLSFS 2005, LNCS 3940*, pages 34–51, 2006.
- [10] Roman Rosipal. Nonlinear partial least squares: An overview. *Chemoinformatics and Advanced Machine Learning Perspectives: Complex Computational Methods and Collaborative Techniques*, pages 169–189, 2011.
- [11] Svante Wold, Nouna Kettaneh-Wold, and Bert Skagerberg. Nonlinear pls modeling. *Chemometrics and Intelligent Laboratory Systems*, 7(1-2):53–65, 1989.
- [12] Ildiko E. Frank. A nonlinear PLS model. *Chemometrics and Intelligent Laboratory Systems*, 8(2):109–119, 1990.
- [13] S Joe Qin and Thomas J McAvoy. Nonlinear pls modeling using neural networks. *Computers & Chemical Engineering*, 16(4):379–391, 1992.
- [14] Xuefeng F. Yan, Dezhao Z. Chen, and Shangxu X. Hu. Chaos-genetic algorithms for optimizing the operating conditions based on RBF-PLS model. *Computers and Chemical Engineering*, 27(10):1393–1404, 2003.
- [15] Hugo Hiden, Ben McKay, Mark Willis, and Gary Montague. Non-linear partial least squares using genetic. In *Genetic Programming 1998: Proceedings of the Third*, pages 128–133. Morgan Kaufmann, 1998.

- [16] Project tycho <http://neurotycho.org/food-tracking-task>.
- [17] José del R Millán, Rüdiger Rupp, Gernot Mueller-Putz, Roderick Murray-Smith, Claudio Giugliemma, Michael Tangermann, Carmen Vidaurre, Febo Cincotti, Andrea Kubler, Robert Leeb, et al. Combining brain–computer interfaces and assistive technologies: State-of-the-art and challenges. *Frontiers in Neuroscience*, 4:161, 2010.
- [18] SG Mason, A Bashashati, M Fatourehchi, KF Navarro, and GE Birch. A comprehensive survey of brain interface technology designs. *Annals of biomedical engineering*, 35(2):137–169, 2007.
- [19] José del R Millán, Frédéric Renkens, Josep Mouriño, and Wulfram Gerstner. Brain-actuated interaction. *Artificial Intelligence*, 159(1-2):241–259, 2004.
- [20] Luis Fernando Nicolas-Alonso and Jaime Gomez-Gil. Brain computer interfaces, a review. *Sensors*, 12(2):1211–1279, 2012.
- [21] Setare Amiri, Reza Fazel-Rezai, and Vahid Asadpour. A review of hybrid brain-computer interface systems. *Advances in Human-Computer Interaction*, 2013:1, 2013.
- [22] Andrey Eliseyev and Tetiana Aksenova. Penalized multi-way partial least squares for smooth trajectory decoding from electrocorticographic (ecog) recording. *PloS one*, 11(5):e0154878, 2016.
- [23] Motrenko A. Gasanov I. Creation of approximating scalogram description in a problem of movement prediction. *Journal of Machine Learning and Data Analysis*, 3(2):160–169, 2017.