

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ (государственный университет)
ФАКУЛЬТЕТ УПРАВЛЕНИЯ И ПРИКЛАДНОЙ МАТЕМАТИКИ
КАФЕДРА «ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ»
ПРИ ВЫЧИСЛИТЕЛЬНОМ ЦЕНТРЕ ИМ. А. А. ДОРОДНИЦЫНА РАН

Владимирова Мария Руслановна

Снижение размерности пространства зависимой переменной

010900 — Прикладные математика и физика

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

Научный руководитель:
д.ф.-м.н. Стрижов Вадим Викторович

Москва
2017 г.

Содержание

1	Введение	4
2	Обзор литературы	6
3	Постановка задачи прогнозирования	7
4	Метод частных наименьших квадратов (PLS)	9
5	Модификация метода частных наименьших квадратов (cnlPLS)	11
5.1	Преобразование зависимой переменной	11
5.2	Преобразование независимой переменной	13
5.3	Алгоритм cnlPLSR	13
6	Вычислительный эксперимент	15
7	Заключение	17

Аннотация

Решается задача обнаружения способов зависимостей в прогнозируемой переменной. Используется набор гомогенных моделей, восстанавливающих прогноз по общему для всех переменных описанию объектов. Анализируется различие в пространстве параметров моделей. По результатам анализа выбирается оптимальная структура каждой модели. Проводится эксперимент на реальных данных объемов потребления электроэнергии для сравнения предложенных методов.

Ключевые слова: *прогнозирование временных рядов; мультиколлинеарность; метод частных наименьших квадратов; PLS; нелинейный PLS.*

1 Введение

В работе решается задача прогнозирования потребления электроэнергии на основе исторических данных. Электрическая энергия является важной движущей силой экономического развития, а точность прогнозов спроса является важным фактором, который ведет к успешному эффективному планированию. По этой причине энергетическим аналитикам необходимо руководство для лучшего выбора наиболее подходящих методов прогнозирования, чтобы обеспечить точные прогнозы тенденций потребления электроэнергии. Для решения этой задачи используется авторегрессионная модель, то есть предполагается, что значение сигнала в данный момент времени линейно зависит от предыдущих значений этого же сигнала. Авторегрессионная модель является неустойчивой из-за наличия мультиколлинеарности в исторических данных. Для решения этой проблемы необходимо использовать методы отбора признаков [1], в результате чего повышается устойчивость модели без существенного снижения качества прогноза.

В работе исследуются методы отбора признаков: метод частных наименьших квадратов (PLS) [2] и предложенная его модификация (cnlPLS). Метод частных наименьших квадратов основан на снижении размерности матрицы признаков и выделяет линейные комбинации признаков, которые оказывают наибольшее влияние на вектор ответов. Выделение признаков происходит итеративно, в порядке уменьшения их влияния на вектор ответов [2]. Поэтому можно рассматривать только самые значимые комбинации, незначительно потеряв в точности прогноза. Предлагается провести модификацию алгоритма PLS: совершить криволинейное и нелинейное преобразования пространства целевой переменной для учета зависимостей между сигналами в разные моменты времени.

В работе проведено сравнение двух методов отбора признаков в задаче авторегрессионного прогнозирования сигналов (PLSR и cnlPLSR). Цель регрессии PLS [3] — предсказать \mathbf{Y} по \mathbf{X} и описать их общую структуру. Когда \mathbf{Y} — вектор, а \mathbf{X} — матрица полного ранга, эта цель может быть выполнена с использованием обычной линейной регрессии. Если число предикторов велико по сравнению с числом наблюдений, то \mathbf{X} будет сингулярным и регрессионный подход в этом случае невозможен из-за наличия мультиколлинеарности.

В качестве практической проверки данных методов в ходе вычислительного эксперимента решается задача прогнозирования на реальных данных, содержащих объ-

емы потребления электроэнергии в Варшаве. Результатом применения отбора признаков является снижение размерности задачи и повышение устойчивости модели без существенной потери точности прогноза.

2 Обзор литературы

Методы PLS и PLS регрессия описаны в работах [4, 5].

Разницу между методом PLS и связанными с ним подходами, различные разновидности регрессии PLS можно найти в [6]. Нелинейное расширение метода PLSR впервые введено в [7]. В литературе были разработаны различные модификации PLS. Например, функции активации искусственных нейронных сетей используются в методе PLS [8]. Поскольку функции активации обеспечивают нелинейные преобразования, решается проблема мультиколлинеарности. Предложены нелинейные методы PLS, основанные на различных моделях: искусственных нейронных сетей [9], функции активации радиальных оснований [10], логистическая функция активации и методы оптимизации роевых частиц [11], используют прямые нейронные сети [12], искусственную нейронную сеть Эльмана [13].

3 Постановка задачи прогнозирования

Для решения задачи прогнозирования сигнала предлагается использовать авторегрессионную модель. Рассматривается сигнал $\mathbf{x} = [x_t]$, $t = 1, \dots, n$, $x_t \in \mathbb{R}$. Необходимо спрогнозировать следующие r значений сигнала: $\mathbf{y} = [y_t]$, $t = 1, \dots, r$, $y_t \in \mathbb{R}$. Предполагается, что $r < n$. Постановка задачи прогнозирования представлена на рис. ??.

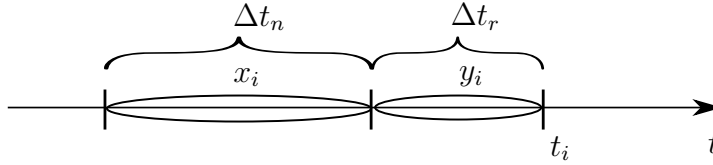


Рис. 1: Задача прогнозирования

Предполагается, что сигнал обладает следующими свойствами:

- значения сигнала получены через одинаковые промежутки времени,
- в сигнале нет пропущенных значений,
- сигнал имеет период $\tau > r$.

Для решения задачи прогнозирования используется авторегрессионная модель. В авторегрессионной модели предполагается, что значение сигнала в данный момент времени линейно зависит от предыдущих значений этого же сигнала. В этой модели признаками являются предыдущие значения прогнозируемого сигнала.

Пусть $\mathbf{X} \in \mathbb{R}^{m \times n}$ – матрица плана, $\mathbf{Y} \in \mathbb{R}^{m \times r}$ – матрица ответов.

$$[\mathbf{X} | \mathbf{Y}] = \left[\begin{array}{cccc|cccc} x_1 & x_2 & \dots & x_n & y_1 & y_2 & \dots & y_r \\ x_2 & x_3 & \dots & x_{n+1} & y_2 & y_3 & \dots & y_{r+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_m & x_{m+1} & \dots & x_{m+n-1} & y_m & y_{m+1} & \dots & y_{m+r-1} \end{array} \right] = \left[\begin{array}{c|c} \mathbf{x}_1 & \mathbf{y}_1 \\ \mathbf{x}_2 & \mathbf{y}_2 \\ \vdots & \vdots \\ \mathbf{x}_m & \mathbf{y}_m \end{array} \right].$$

В авторегрессионной модели матрица ответов представляется в виде

$$\mathbf{Y} = f(\mathbf{X}, \boldsymbol{\Theta}) + \epsilon(\mathbf{X}),$$

где $f(\mathbf{X}, \boldsymbol{\Theta}) = \mathbf{X}\boldsymbol{\Theta}$ – модель, $\boldsymbol{\Theta} \in \mathbb{R}^{n \times r}$ – матрица параметров модели, а $\epsilon(\mathbf{X}) \in \mathbb{R}^{n \times r}$ – вектор регрессионных остатков.

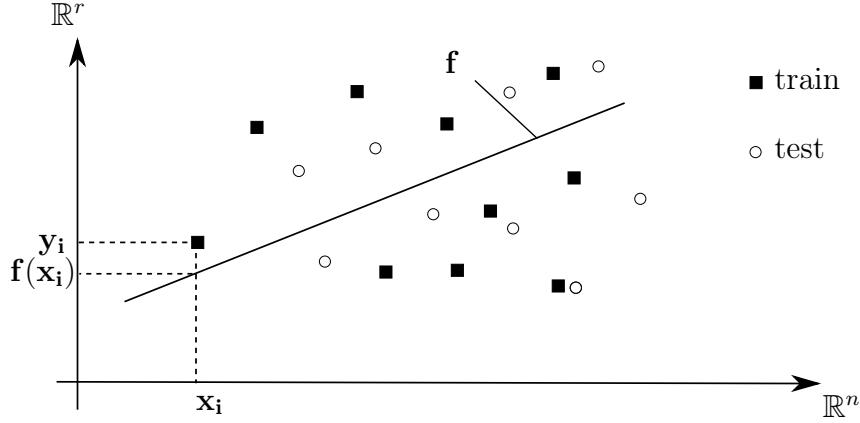


Рис. 2: Задача линейной регрессии временных рядов

Введем функцию ошибки S :

$$S(\Theta|\mathfrak{D}) = \|f(\mathbf{X}, \Theta) - \mathbf{Y}\|_2^2 = \|\mathbf{X}\Theta - \mathbf{Y}\|_2^2,$$

где \mathfrak{D} — обучающая выборка. Пусть $\mathcal{I} = \{1, \dots, m\}$ — множество индексов элементов выборки. Разобьем выборку $\mathfrak{D} = (\mathbf{X}, \mathbf{Y})$ на обучающую $\mathfrak{D}_{\mathcal{L}}$ и контрольную $\mathfrak{D}_{\mathcal{C}}$, где $\mathcal{L}, \mathcal{C} \in \mathcal{I}$ ($\mathcal{I} = \mathcal{L} \sqcup \mathcal{C}$). На рис. ?? проиллюстрировано решение задачи линейной регрессии, где в качестве объектов выступают временные ряды, а в качестве ответов — вектора предсказаний.

Прогноз сигнала на горизонте r — это ответ модели при найденных оптимальных параметрах задачи авторегрессии: $\mathbf{y} = \mathbf{x}\Theta^*$, где

$$\Theta^* = \arg \min_{\Theta \in \mathbb{R}^{n \times r}} S(\Theta|\mathfrak{D}_{\mathcal{L}}).$$

Значения объема потребления электроэнергии в соседние моменты времени не являются независимыми. Таким образом, наблюдается мультиколлинеарность между признаками в авторегрессионной модели прогнозирования сигнала. Для решения проблемы мультиколлинеарности применяются методы отбора признаков. Отбор признаков происходит на контрольной выборке $\mathfrak{D}_{\mathcal{C}}$. В следующем разделе описан базовый используемый метод отбора признаков.

4 Метод частных наименьших квадратов (PLS)

Идея метода частных наименьших квадратов состоит в том, чтобы перейти в пространство меньшей размерности с сохранением ковариации между признаками и ответами. Матрица плана \mathbf{X} и матрица ответов \mathbf{Y} проецируются на пространство меньшей размерности $m \times l$ так, что ковариация между новыми признаками и ответами максимальна:

$$\mathbf{X} = \mathbf{T}\mathbf{P}^\top + \mathbf{E}, \quad \mathbf{Y} = \mathbf{U}\mathbf{Q}^\top + \mathbf{F} \quad (4.1)$$

$\mathbf{T} \in \mathbb{R}^{m \times l}$, $\mathbf{U} \in \mathbb{R}^{m \times l}$ — значения признаков и ответов в спроектированном пространстве. $\mathbf{P} \in \mathbb{R}^{n \times l}$, $\mathbf{Q} \in \mathbb{R}^{r \times l}$ — матрицы перехода из нового пространства в старое. $\mathbf{E} \in \mathbb{R}^{m \times n}$, $\mathbf{F} \in \mathbb{R}^{m \times r}$ — матрицы невязок. Так как ковариация между новыми признаками и ответами максимальна, то можно строить регрессионную модель в пространстве меньшей размерности с сохранением точности прогноза. Параметр метода частных наименьших квадратов $l \in \mathbb{N}$ определяет размерность этого пространства. Отбор признаков осуществляется в виде замены исходных признаков $[\mathbf{x}_1, \dots, \mathbf{x}_n]$ на l новых признаков — линейные комбинации исходных признаков.

Чтобы получить модель регрессии, связывающую \mathbf{Y} и \mathbf{X} , сначала находятся параметры β_i такие, что $\mathbf{u}_i = \mathbf{t}_i \beta_i$, $i \in \{1, \dots, l\}$. Подставляя параметры в модель (4.1), получаем

$$\mathbf{Y} = \mathbf{U}\mathbf{Q}^\top = \mathbf{T}\text{diag}(\boldsymbol{\beta})\mathbf{Q}^\top = \mathbf{X}\boldsymbol{\Theta}.$$

Псевдокод метода PLSR приведен в алгоритме 4.1. Из сохраненных на каждом из l шагов значений \mathbf{t} , \mathbf{u} , \mathbf{p} , \mathbf{q} формируются матрицы \mathbf{T} , \mathbf{U} , \mathbf{P} , \mathbf{Q} соответственно.

Размеры векторов в алгоритме можно изобразить следующим образом:

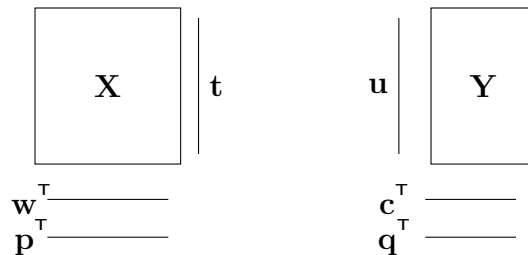


Рис. 3: Размерности векторов в алгоритме PLS

Алгоритм 4.1 Алгоритм PLSR

Вход: $\mathbf{X}, \mathbf{Y}, l$;

Выход: $\mathbf{T}, \mathbf{U}, \mathbf{P}, \mathbf{Q}$;

```

1: инициализировать  $\mathbf{u} := \mathbf{y}_1$  (вектор матрицы  $\mathbf{Y}$ )
2: для  $i = 1, \dots, l$ 
3:   повторять
4:      $\mathbf{w} := \mathbf{X}^\top \mathbf{u} / (\mathbf{u}^\top \mathbf{u})$ 
5:     нормировать  $\mathbf{w}$ :  $\|\mathbf{w}\| = 1$ 
6:      $\mathbf{t} := \mathbf{X}\mathbf{w}$ 
7:      $\mathbf{c} := \mathbf{Y}^\top \mathbf{t} / (\mathbf{t}^\top \mathbf{t})$ 
8:     нормировать  $\mathbf{c}$ :  $\|\mathbf{c}\| = 1$ 
9:      $\mathbf{u} := \mathbf{Y}\mathbf{c}$ 
10:   пока  $\mathbf{t}$  не перестанет меняться
11:   сохранить  $\mathbf{t}, \mathbf{u}, \mathbf{c}$ 
12:    $\mathbf{p} := \mathbf{X}^\top \mathbf{t} / (\mathbf{t}^\top \mathbf{t}), \mathbf{q} := \mathbf{Y}^\top \mathbf{u} / (\mathbf{u}^\top \mathbf{u})$ 
13:   регрессия ( $\mathbf{u}$  на  $\mathbf{t}$ ):  $\beta := \mathbf{u}^\top \mathbf{t} / (\mathbf{t}^\top \mathbf{t})$ 
14:    $\mathbf{X} := \mathbf{X} - \mathbf{t}\mathbf{t}^\top \mathbf{X} / (\mathbf{t}^\top \mathbf{t})$ 
15:    $\mathbf{Y} := \mathbf{Y} - \beta \mathbf{t}\mathbf{c}^\top$ 

```

В [?] показано, что вектора \mathbf{w} и \mathbf{c} во внутреннем цикле максимизируют ковариацию между новыми признаками и матрицей ответов:

$$[\text{cov}(\mathbf{t}, \mathbf{u})]^2 = [\text{cov}(\mathbf{X}\mathbf{w}, \mathbf{Y}\mathbf{c})]^2 = \max_{\|\mathbf{s}\|=1, \|\mathbf{r}\|=1} [\text{cov}(\mathbf{X}\mathbf{s}, \mathbf{Y}\mathbf{r})]^2,$$

где $\text{cov}(\mathbf{t}, \mathbf{u}) = \mathbf{t}^\top \mathbf{u} / n$ означает выборочную ковариацию.

5 Модификация метода частных наименьших квадратов (cnlPLS)

Предлагается провести модификацию алгоритма PLS: совершить криволинейное и нелинейное преобразования пространства целевой переменной и независимой переменной для учета мультиколлинеарности между сигналами в разные моменты времени. Схема модифицированного алгоритма представлена на рис. 4.

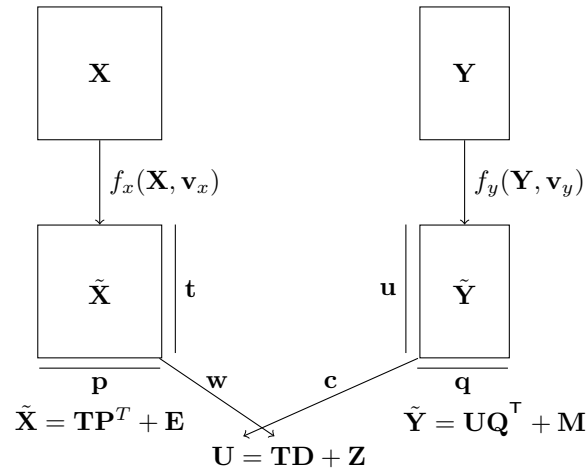


Рис. 4: Схема модифицированного метода частных наименьших квадратов

5.1 Преобразование зависимой переменной

Рассматриваются следующие преобразования пространства зависимой переменной Y :

- криволинейное с вектором параметров \mathbf{v} (примеры преобразований представлены в таб. 1)

$$\check{Y} = g(Y, \mathbf{v}), \quad (5.1)$$

- нелинейное непараметрическое преобразование

$$\hat{Y} = h(Y),$$

- суперпозиция преобразований $Y \xrightarrow{g(Y, \mathbf{v})} \check{Y} \xrightarrow{h(\check{Y})} \tilde{Y}$

$$\tilde{Y} = h(\check{Y}) = h(g(Y, \mathbf{v})) = f(Y, \mathbf{v}).$$

№	Функция	Параметры
1	$g(x) = \text{sign}(x) \exp(a)(\exp(b x) - 1)$	$a, b > 0$
2	$g(x) = \text{sign}(x) \exp(a)(\exp(b \ln(1 + x)) - 1)$	$a, b > 0$
3	$g(x) = \text{sign}(x) \exp(a)(\exp(b x ^{1/2}) - 1)$	$a, b > 0$
4	$g(x) = \text{sign}(x) \exp(a)(\exp(b x ^{1/3}) - 1)$	$a, b > 0$
5	$g(x) = \text{sign}(x) \exp(a)(\exp(b x ^{1/4}) - 1)$	$a, b > 0$
6	$g(x) = \text{sign}(x) \exp(a)(\exp(b x ^2) - 1)$	$a, b > 0$

Таблица 1: Криволинейные преобразования

Криволинейные преобразования выбирались таким образом, чтобы функции преобразования удовлетворяли следующим условиям:

- отображает множество действительных чисел во множество действительных чисел $g : \mathbb{R} \rightarrow \mathbb{R}$,
- принимает нулевое значение в нуле $g(0) = 0$,
- дифференцируется по параметрам,
- является обратимой, то есть существует g^{-1} .

Предлагается подход для обновления весов \mathbf{v} , основанный на линеаризации функции преобразования. Разложим (5.3) в ряд Тейлора до второго порядка:

$$\check{\mathbf{y}} \approx \check{\mathbf{y}}_0 + \left. \frac{\partial g}{\partial \mathbf{v}} \right|_0 \Delta \mathbf{v},$$

где $\check{\mathbf{y}}_0$ — это значение функции g при известном значении переменной \mathbf{y} . Для вычисления $\Delta \mathbf{v}$ предложены следующие шаги. Рассматривается разность $\check{\mathbf{y}} - \check{\mathbf{y}}_0 = \left. \frac{\partial g_y}{\partial \mathbf{v}_y} \right|_0 \Delta \mathbf{v}$. Определется рассогласование \mathbf{e}

$$\mathbf{e} = \check{\mathbf{y}} - \check{\mathbf{y}}_0 = \left. \frac{\partial g}{\partial \mathbf{v}} \right|_0 \Delta \mathbf{v} = \mathbf{J}_y \Delta \mathbf{v},$$

где матрица \mathbf{J}_y состоит из частных производных $\left\{ \left. \frac{\partial g}{\partial v_i} \right|_0 \right\}_{i=1}^N$, вычисленных при известном значении переменной \mathbf{y} . Далее $\Delta \mathbf{v}$ вычисляется решением задачи регрессии рассогласования \mathbf{e} так, что

$$\Delta \mathbf{v} = (\mathbf{J}_y^\top \mathbf{J}_y)^{-1} \mathbf{J}_y^\top \mathbf{e}. \quad (5.2)$$

5.2 Преобразование независимой переменной

Аналогично преобразованию целевой переменной \mathbf{Y} , совершается преобразование зависимой переменной \mathbf{X} для учета мультиколлинеарности в признаковом пространстве.

Рассмотрим преобразования \mathbf{X}

- криволинейное с вектором параметров \mathbf{v} (таб. 1)

$$\check{\mathbf{X}} = g(\mathbf{X}, \mathbf{v}), \quad (5.3)$$

- нелинейное непараметрическое преобразование

$$\hat{\mathbf{X}} = h(\mathbf{X}),$$

- суперпозиция преобразований

$$\tilde{\mathbf{X}} = h(\check{\mathbf{X}}) = h(g(\mathbf{X}, \mathbf{v})) = f(\mathbf{X}, \mathbf{v}).$$

5.3 Алгоритм snPLSR

В данном разделе представлен модифицированный метод PLSR, содержащий шаги преобразования целевой переменной. Аналогично методу PLSR (алгоритм 4.1), алгоритм 5.1 начинается с инициализации вектора \mathbf{u} , а обновления весов преобразования считается с помощью рассогласования \mathbf{e} для вектора \mathbf{u} , вычисленного в цикле и на предыдущей итерации.

Алгоритм 5.1 Алгоритм cnlPLSR

Вход: $\mathbf{X}, \mathbf{Y}, l$;

Выход: $\mathbf{T}, \mathbf{U}, \mathbf{P}, \mathbf{Q}$;

- 1: инициализировать \mathbf{v}
 - 2: определить \mathbf{u}_0 как вектор преобразования $f(\mathbf{Y}, \mathbf{v})$
 - 3: для $i = 1, \dots, l$
 - 4: **повторять**
 - 5: $\mathbf{w} := \mathbf{X}^\top \mathbf{u}_0 / (\mathbf{u}_0^\top \mathbf{u}_0)$
 - 6: нормировать \mathbf{w} : $\|\mathbf{w}\| = 1$
 - 7: $\mathbf{t} := \mathbf{X}\mathbf{w}$
 - 8: $\tilde{\mathbf{Y}} = f(\mathbf{Y}, \mathbf{v})$
 - 9: $\mathbf{c} := \tilde{\mathbf{Y}}^\top \mathbf{t} / (\mathbf{t}^\top \mathbf{t})$
 - 10: нормировать \mathbf{c} : $\|\mathbf{c}\| = 1$
 - 11: $\mathbf{u} := \tilde{\mathbf{Y}}\mathbf{c}$
 - 12: $\mathbf{e} := \mathbf{u} - \mathbf{u}_0$
 - 13: $\mathbf{J} := \partial \mathbf{u} / \partial \mathbf{v}$
 - 14: $\Delta \mathbf{v} = (\mathbf{J}^\top \mathbf{J})^{-1} \mathbf{J}^\top \mathbf{e}$
 - 15: $\mathbf{v} := \mathbf{v} + \Delta \mathbf{v}, \|\mathbf{v}\| = 1$
 - 16: $\mathbf{u}_0 := \mathbf{u}$
 - 17: **пока** \mathbf{t} не перестанет меняться
 - 18: сохранить $\mathbf{t}, \mathbf{u}, \mathbf{c}$
 - 19: вычислить $\tilde{\mathbf{Y}}$
 - 20: $\mathbf{p} := \mathbf{X}^\top \mathbf{t} / (\mathbf{t}^\top \mathbf{t}), \mathbf{q} := \tilde{\mathbf{Y}}^\top \mathbf{u} / (\mathbf{u}^\top \mathbf{u})$
 - 21: регрессия (\mathbf{u} на \mathbf{t}): $\beta := \mathbf{u}^\top \mathbf{t} / (\mathbf{t}^\top \mathbf{t})$
 - 22: $\mathbf{X} := \mathbf{X} - \mathbf{t}\mathbf{t}^\top \mathbf{X} / (\mathbf{t}^\top \mathbf{t})$
 - 23: $\tilde{\mathbf{Y}} := \tilde{\mathbf{Y}} - \beta \mathbf{t}\mathbf{c}^\top$
 - 24: вычислить \mathbf{Y} с помощью обратного преобразования $f^{-1}(\tilde{\mathbf{Y}}, \mathbf{v})$
-

6 Вычислительный эксперимент

В рамках вычислительного эксперимента строится прогноз временных рядов. В ходе эксперимента сравниваются методы PLSR, нелинейных автоэнкодеров и cnlPLS. Сравнение проводится на реальных данных объемов потребления электроэнергии в Польше.

Вычислительный эксперимент, продемонстрированный в этом разделе, основан на данных электроэнергии. Данные состоят из временного ряда польских электрических нагрузок и временных рядов погоды в Варшаве (долгота: 21,25, широта: 52,30, высота над уровнем моря: 94). Временные ряды энергии состоят из почасовых записей (всего 52512 наблюдений), а погодные измерения проводились раз в день и содержат 2188 наблюдений. Многомасштабные временные ряды соответствуют периоду 1999-2004 годов. Результаты, полученные с этим набором данных, являются иллюстрацией предлагаемых методов, поскольку данные содержат многомасштабные временные ряды, имеющие различный характер.

Примеры работы алгоритма приведены на рис. 5. Метод успешно делает краткосрочный прогноз (до 10 дней). С увеличением горизонта прогнозирования предсказание смещается.

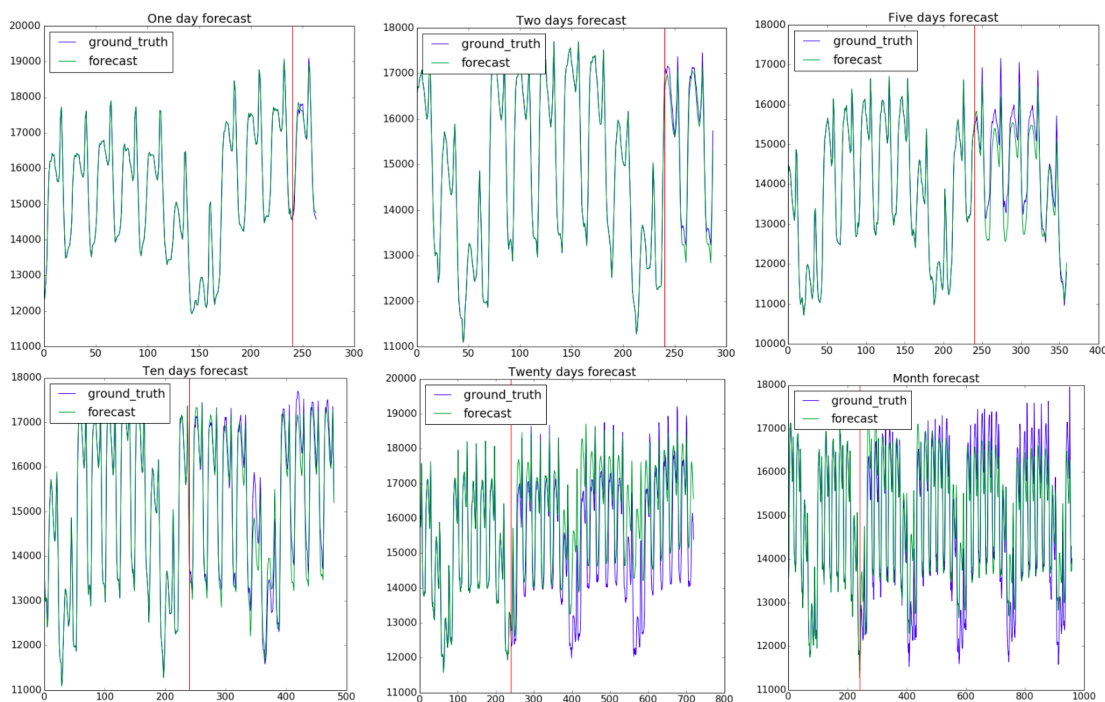
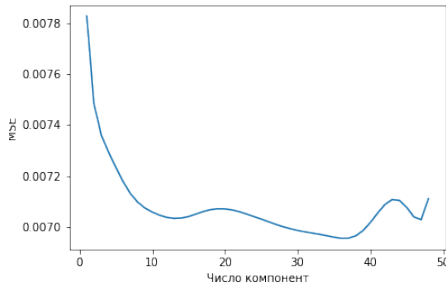
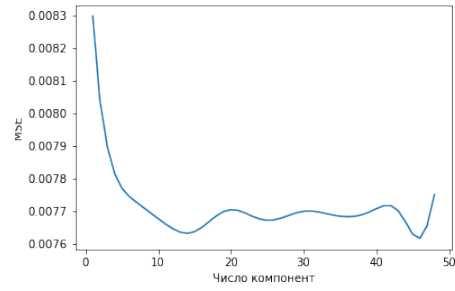


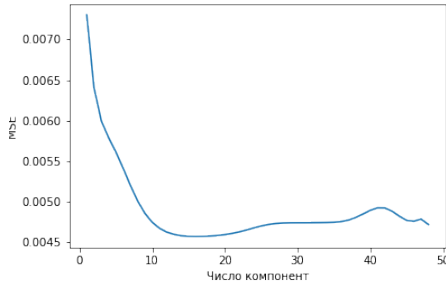
Рис. 5: Прогнозирование базового алгоритма на 1, 2, 5, 10, 20, 30 дней



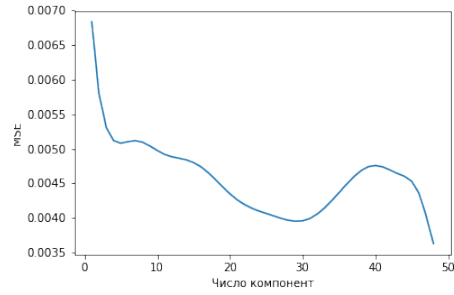
(a) $g(x) = \text{sign}(x)e^a(\exp(b|x|) - 1)$



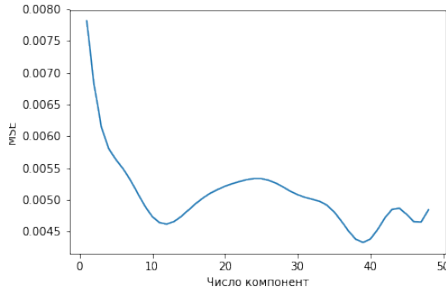
(b) $g(x) = \text{sign}(x)e^a(\exp(b \ln(1 + |x|)) - 1)$



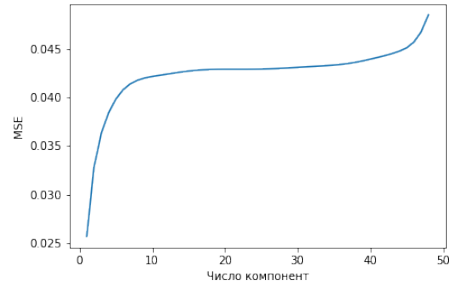
(c) $g(x) = \text{sign}(x)e^a(\exp(b|x|^{1/2}) - 1)$



(d) $g(x) = \text{sign}(x)e^a(\exp(b|x|^{1/3}) - 1)$



(e) $g(x) = \text{sign}(x)e^a(\exp(b|x|^{1/4}) - 1)$



(f) $g(x) = \text{sign}(x)e^a(\exp(b|x|^2) - 1)$

Рис. 6: Зависимость ошибки от числа компонент в алгоритме snlPLS для разных функций

Результаты вычислительного эксперимента для предложенного модифицированного алгоритма snlPLS представлены на рис. 6. На графиках изображены smoothed зависимости ошибки MSE от числа компонент в алгоритме для разных функций. Из графиков видно, что для функций (a) – (e) ошибка при увеличении числа компонент падает, затем колеблется, слабо меняясь. Ошибка алгоритма с функцией (f) увеличивается при увеличении числа компонент. Это означает, что преобразование, выполненное в пространстве целевой переменной с помощью функции (f), плохо описывает зависимость. Меньшую ошибку имеют функции, растущие медленнее, а именно (d) и (e).

В табл. 2 продемонстрировано увеличение точности прогнозирования при исполь-

Алгоритм	N=3	N=5	N=10	N=20
PLS	0,00404	0,00337	0,00151	0,00135
cnlPLS $g(x) = \text{sign}(x) \exp(a)(\exp(b x) - 1)$	0.00529	0.00514	0.00536	0.00506
cnlPLS $g(x) = \text{sign}(x) \exp(a)(\exp(b \ln(1 + x) - 1)$	0.00362	0.00386	0.00326	0.00317
cnlPLS $g(x) = \text{sign}(x) \exp(a)(\exp(b x ^{1/2}) - 1)$	0.00272	0.00236	0.00287	0.00128
cnlPLS $g(x) = \text{sign}(x) \exp(a)(\exp(b x ^{1/3}) - 1)$	0.00241	0.00233	0.00221	0.00173
cnlPLS $g(x) = \text{sign}(x) \exp(a)(\exp(b x ^{1/4}) - 1)$	0.00796	0.00768	0.00737	0.00803
cnlPLS $g(x) = \text{sign}(x) \exp(a)(\exp(b x ^2) - 1)$	0.00816	0.00798	0.00796	0.00775

Таблица 2: Значения ошибки MSE для разных чисел компонент и разных функций

зовании криволинейного преобразования в пространстве зависимой переменной, но увеличение точности в пределах погрешности алгоритма (0.0005-0.0010). Функции с быстрым ростом не позволяют описать зависимость.

7 Заключение

В данной работе предложен новый подход к обнаружению зависимостей в пространстве зависимой переменной задачи прогнозирования временных рядов. Сравнивались результаты прогнозирования временных рядов, полученных с помощью метода частных наименьших квадратов и предложенной модификации. Проведен вычислительный эксперимент на реальных данных потребления электроэнергии в Варшаве. Построенная прогностическая модель показала высокое качество предсказания электрической нагрузки.

Список литературы

- [1] Karen E Adolph. *Learning to learn*.
- [2] Kee Siong Ng. A Simple Explanation of Partial Least Squares. pages 1–10, 2013.
- [3] Hervé Abdi. Partial Least Squares (PLS) Regression. *Encyclopedia for research methods for the social sciences*, pages 792–795, 2003.
- [4] Paul Geladi. Notes on the history and nature of partial least squares (PLS) modelling. *Journal of Chemometrics*, 2(January):231–246, 1988.
- [5] Agnar Höskuldsson. PLS regression. *Journal of Chemometrics*, 2(August 1987):581–591, 1988.
- [6] Sidney R. Lehky, Roozbeh Kiani, Hossein Esteky, and Keiji Tanaka. Dimensionality of object representations in monkey inferotemporal cortex. *Neural computation*, 1872(10):1840–1872, 2014.
- [7] Ildiko E. Frank. A nonlinear PLS model. *Chemometrics and Intelligent Laboratory Systems*, 8(2):109–119, 1990.
- [8] E.C. Malthouse, A.C. Tamhane, and R.S.H. Mah. Nonlinear partial least squares. *Computers & Chemical Engineering*, 21(8):875–890, 1997.
- [9] J Mcavovt and Chemical Process. USING. 16(4):379–391, 1992.
- [10] Xuefeng F. Yan, Dezhaoh 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.
- [11] Yan Ping Zhou, Jian Hui Jiang, Wei Qi Lin, Lu Xu, Hai Long Wu, Guo Li Shen, and Ru Qin Yu. Artificial neural network-based transformation for nonlinear partial least-square regression with application to QSAR studies. *Talanta*, 71(2):848–853, 2007.
- [12] Yan Xuefeng. Hybrid artificial neural network based on BP-PLSR and its application in development of soft sensors. *Chemometrics and Intelligent Laboratory Systems*, 103(2):152–159, 2010.

- [13] Elif Bulut and Erol Egrioglu. A New Partial Least Square Method Based on Elman Neural Network. 4(4):154–158, 2014.
- [14] Il Gyo Chong and Chi Hyuck Jun. Performance of some variable selection methods when multicollinearity is present. *Chemometrics and Intelligent Laboratory Systems*, 78(1):103–112, 2005.
- [15] Выбор признаков в задаче авторегрессионного прогнозирования биомедицинских сигналов *. pages 1–10, 2017.
- [16] Roman Rosipal. Nonlinear partial least squares: An overview. *Chemoinformatics and Advanced Machine Learning Perspectives: Complex Computational Methods and Collaborative Techniques*, pages 169–189, 2011.
- [17] Svante Wold, Nouna Kettaneh-Wold, and Bert Skagerberg. Nonlinear PLS modeling. *Chemometrics and Intelligent Laboratory Systems*, 7(1-2):53–65, 1989.
- [18] 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.
- [19] Wen-Cong Lu, Nian-Yi Chen, Guo-Zheng Li, and Jie Yang. Multitask Learning Using Partial Least Squares Method. *Proceedings of the Seventh International Conference on Information Fusion; International Society of Information Fusion*, 1:79–84, 2004.
- [20] Rich Caruana and Virginia R. de Sa. Benefitting from the Variables that Variable Selection Discards. *Journal of Machine Learning Research*, 3(7-8):1245–1264, 2003.
- [21] Alexandre Varnek and Igor Baskin. Machine learning methods for property prediction in chemoinformatics: Quo Vadis? *Journal of Chemical Information and Modeling*, 52(6):1413–1437, 2012.