

Снижение размерности пространства зависимой переменной в задачах прогнозирования.*

Мария Владимировна

mrvladimirova@gmail.com

Московский физико-технический институт

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

БЕЗ ССЫЛОК НА ЛИТЕРАТУРУ, БЕЗ КАРТИНОК

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 [?], в результате чего повышается устойчивость модели без существенного снижения качества прогноза.

В работе исследуются методы отбора признаков: метод частных наименьших квадратов (PLS) [?] 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 [?]. Поэтому можно рассматривать только самые значимые комбинации, незначительно потеряв в точности прогноза.

Методы PLS и PLS регрессии подробно описаны в работах [?, ?]. Разницу между методом PLS и связанными с ним подходами, различные разновидности регрессии PLS можно найти в [?].

Нелинейное расширение метода PLSR впервые введено в [?]. В литературе были разработаны различные модификации PLS. Например, функции активации искусственных нейронных сетей используются в методе PLS [?]. Поскольку функции активации обеспечивают нелинейные преобразования, решается проблема мультиколлинеарности. Предложены нелинейные методы PLS, основанные на различных моделях: искусственных нейронных

сетей [?], функции активации радиальных оснований [?], логистическая функция активации и методы оптимизации роевых частиц [?], используют прямые нейронные сети [?], искусственную нейронную сеть Эльмана [?].

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 [?] — 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 сингулярной 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 объемы потребления электроэнергии в Варшаве. Результатом применения отбора признаков является снижение размерности задачи и повышение устойчивости модели без существенной потери точности прогноза.

Текст про кортиграмму

Постановка задачи прогнозирования

ОБЩАЯ ПОСТАНОВКА Рассматривается сигнал $\mathbf{x} = [x_t]$, $t = 1, \dots, n$, $x_t \in \mathbb{R}$. Необходимо спрогнозировать r значений сигнала или отклика: $\mathbf{y} = [y_t]$, $t = 1, \dots, r$, $y_t \in \mathbb{R}$. Прогнозом могут являться, например, r последующих значений временного ряда. Предполагается, что $r < n$.

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 сигнала предлагается использовать авторегрессионную модель. В авторегрессионной модели признаками являются предыдущие значения прогнозируемого сигнала. Предполагается, что значение сигнала в данный момент времени линейно зависит от предыдущих значений этого же сигнала.

Пусть $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^m \in \mathbb{R}^{m \times n}$ – матрица признаков описания, $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^m \in \mathbb{R}^{m \times r}$ – матрица ответов. Каждая строка \mathbf{x}_i матрицы \mathbf{X} – локальная история сигнала (n значений сигнала, начиная с момента i). Каждая строка \mathbf{y}_i матрицы \mathbf{Y} – локальный прогноз сигнала (например, r значений сигнала, начиная с момента $n + 1$).

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

В авторегрессионной модели матрица ответов представляется в виде

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

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

Пусть $\mathcal{D} = (\mathbf{X}, \mathbf{Y})$ – выборка, $\mathcal{I} = \{1, \dots, m\}$ – множество индексов элементов выборки. Введем функцию ошибки S на выборке \mathcal{D} :

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

Разобьем выборку \mathcal{D} на обучающую $\mathcal{D}_{\mathcal{L}}$ и контрольную $\mathcal{D}_{\mathcal{C}}$, где $\mathcal{L}, \mathcal{C} \in \mathcal{I}$ ($\mathcal{I} = \mathcal{L} \sqcup \mathcal{C}$).

Прогноз сигнала на r следующих значений ряда – это ответ модели при найденных оптимальных параметрах задачи авторегрессии: $\mathbf{y} = \mathbf{x}\boldsymbol{\Theta}^*$, где

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

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

Метод частных наименьших квадратов (PLS)

Идея метода частных наименьших квадратов состоит в том, чтобы перевести матрицу объектов \mathbf{X} и матрицу ответов \mathbf{Y} в пространство меньшей размерности. В отличие от метода главных компонент, метод частных наименьших квадратов учитывает взаимосвязь между матрицами \mathbf{X} и \mathbf{Y} .

Матрица плана \mathbf{X} и матрица ответов \mathbf{Y} проецируются на пространство меньшей размерности l следующим образом:

$$\begin{aligned}\mathbf{X} &= \mathbf{T}\mathbf{P}^\top + \mathbf{E}, \\ \mathbf{Y} &= \mathbf{U}\mathbf{Q}^\top + \mathbf{F},\end{aligned}$$

где $\mathbf{T} \in \mathbb{R}^{m \times l}$, $\mathbf{U} \in \mathbb{R}^{m \times l}$ — матрицы объектов и ответов в спроектированном пространстве, причём $\mathbf{T}^\top \mathbf{T} = \mathbf{I}_l$; $\mathbf{P} \in \mathbb{R}^{n \times l}$, $\mathbf{Q} \in \mathbb{R}^{r \times l}$ — 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.

Алгоритм регрессии PLS итеративно на каждом из l шагов вычисляет по одному столбцу \mathbf{t} , \mathbf{u} , \mathbf{p} , \mathbf{q} , из которых формируются матрицы \mathbf{T} , \mathbf{U} , \mathbf{P} , \mathbf{Q} соответственно. Предполагается, что вектора новых признаков \mathbf{t} и \mathbf{u} являются линейными комбинациями столбцов матриц \mathbf{X} и \mathbf{Y} соответственно.

Псевдокод метода регрессии PLS приведен в алгоритме 1. Во внутреннем цикле алгоритма вычисляются скрытые вектора \mathbf{w} и \mathbf{c} . Сюда про power iteration + ссылка Обновляя вектора по данным правилам, мы максимизируем ковариацию между векторами \mathbf{t} и \mathbf{u}

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

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.

Если ковариация между новыми признаками и ответами максимальна, то можно строить регрессионную модель в пространстве меньшей размерности с сохранением уровня точности прогноза. Параметр метода частных наименьших квадратов $l \in \mathbb{N}$ определяет размерность нового пространства. Отбор признаков осуществляется в виде замены исходных признаков на l новых признаков — линейные комбинации исходных признаков.

Чтобы получить модель регрессии, связывающую \mathbf{Y} и \mathbf{X} , на каждом шаге алгоритма 1 находятся коэффициенты β такие, что $\mathbf{u} = \mathbf{t}\beta$. Найденные коэффициенты образуют вектор параметров β размерности l . Подставляя вектор параметров в модель (??), получаем

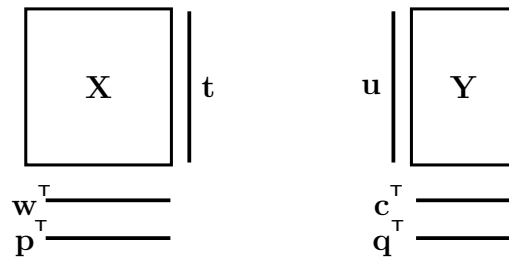
$$\mathbf{Y} = \mathbf{U}\mathbf{Q}^\top = \mathbf{T}\text{diag}(\beta)\mathbf{Q}^\top = \mathbf{X}\Theta.$$

ДАЛЬШЕ НЕКРАСИВЫЙ РИСУНОК Размеры векторов в алгоритме можно изобразить следующим образом:

Алгоритм 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: $\mathbf{w} := \mathbf{X}^T \mathbf{u} / (\mathbf{u}^T \mathbf{u})$
- 5: нормировать \mathbf{w} : $\|\mathbf{w}\| = 1$
- 6: $\mathbf{t} := \mathbf{X} \mathbf{w}$
- 7: $\mathbf{c} := \mathbf{Y}^T \mathbf{t} / (\mathbf{t}^T \mathbf{t})$
- 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: $\mathbf{p} := \mathbf{X}^T \mathbf{t} / (\mathbf{t}^T \mathbf{t}), \mathbf{q} := \mathbf{Y}^T \mathbf{u} / (\mathbf{u}^T \mathbf{u})$
- 13: регрессия (\mathbf{u} на \mathbf{t}): $\beta := \mathbf{u}^T \mathbf{t} / (\mathbf{t}^T \mathbf{t})$
- 14: $\mathbf{X} := \mathbf{X} - \mathbf{t} \mathbf{t}^T \mathbf{X} / (\mathbf{t}^T \mathbf{t})$
- 15: $\mathbf{Y} := \mathbf{Y} - \beta \mathbf{t} \mathbf{c}^T$

93

**Рис. 1.** Размерности векторов в алгоритме PLS

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.

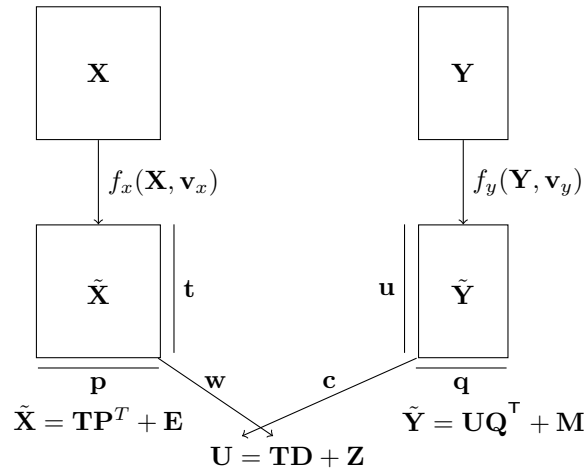


Рис. 2. Схема модифицированного метода частных наименьших квадратов

Преобразование зависимой переменной

Рассматриваются следующие преобразования пространства зависимой переменной \mathbf{Y} :

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

$$\check{\mathbf{Y}} = g(\mathbf{Y}, \mathbf{v}), \quad (1)$$

- нелинейное непараметрическое преобразование

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

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

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

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 (3) в ряд Тейлора до второго порядка:

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

№	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

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

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

110 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
 111 значении переменной \mathbf{y} . Далее $\Delta \mathbf{v}$ вычисляется решением задачи регрессии рассогласова-
 112 ния \mathbf{e} так, что

$$\Delta \mathbf{v} = (\mathbf{J}_y^T \mathbf{J}_y)^{-1} \mathbf{J}_y^T \mathbf{e}. \quad (2)$$

113 Преобразование независимой переменной

114 Аналогично преобразованию целевой переменной \mathbf{Y} , совершается преобразование за-
 115 висимой переменной \mathbf{X} для учета мультиколлинеарности в признаковом пространстве.

116 Рассмотрим преобразования \mathbf{X}

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

$$\check{\mathbf{X}} = g(\mathbf{X}, \mathbf{v}), \quad (3)$$

— нелинейное непараметрическое преобразование

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

— суперпозиция преобразований

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

118 Алгоритм cnPLSR

119 В данном разделе представлен модифицированный метод PLSR, содержащий шаги
 120 преобразования целевой переменной. Аналогично методу PLSR (алгоритм 1), алгоритм
 121 2 начинается с инициализации вектора \mathbf{u} , а обновления весов преобразования считается
 122 с помощью рассогласования \mathbf{e} для вектора \mathbf{u} , вычисленного в цикле и на предыдущей
 123 итерации.

Алгоритм 2 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 $f^{-1}(\tilde{\mathbf{Y}}, \mathbf{v})$
-

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

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. ???. Метод успешно делает краткосрочный прогноз (до 10 дней). С увеличением горизонта прогнозирования предсказание смещается.

Результаты вычислительного эксперимента для предложенного модифицированного алгоритма cnlPLS представлены на рис. ???. На графиках изображены сглаженные 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 использовании 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.

152 **Заключение**

153 В данной работе предложен новый подход к обнаружению зависимостей в пространстве
154 зависимой переменной задачи прогнозирования временных рядов. Сравнивались результа-
155 ты прогнозирования временных рядов, полученных с помощью метода частных наимень-
156 ших квадратов и предложенной модификации. Проведен вычислительный эксперимент на
157 реальных данных потребления электроэнергии в Варшаве. Построенная прогностическая
158 модель показала высокое качество предсказания электрической нагрузки.