

Снижение размерности пространства зависимой переменной в задачах прогнозирования.*

Мария Владимировна, Роман Исаченко

vladimirova.maria@phystech.edu, isa-ro@yandex.ru

Московский физико-технический институт

В работе решается задача обнаружения зависимостей в прогнозируемой переменной. Используется набор гомогенных моделей, восстанавливающих прогноз по общему для всех переменных описанию объектов. Рассматривается линейная модель метода частных наименьших квадратов и ее предложенная нелинейная модификация. Находятся оптимальные параметрические преобразования исходных пространств объектов и ответов. Проводится вычислительный эксперимент на реальных данных объемов потребления электроэнергии и данных сигналов кортикограмм.

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. Предполагается, что значение сигнала в данный момент времени линейно зависит от предыдущих значений этого же сигнала. Авторегрессионная модель является неустойчивой в случае наличия мультиколлинеарности в исторических данных. Для решения этой проблемы необходимо используются 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) [11] и предложенная его нелинейная 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 [11]. Рассматриваются только значимые комбинации признаков, незначительно потеряв в точности прогноза.

Методы PLS регрессии подробно описаны в работах [8,9]. Разницу между методом PLS и связанными с ним подходами, различные разновидности регрессии PLS можно найти в [17].

Нелинейное расширение метода PLS регрессии впервые введено в [6]. В литературе были разработаны различные модификации PLS. Например, функции активации искусственных нейронных сетей используются в методе PLS [?]. **ПОДРОБНЕЕ** Поскольку функции активации обеспечивают нелинейные преобразования, решается проблема мультиколлинеарности. Предложены нелинейные методы PLS, основанные на различных моделях: ис-

кусственных нейронных сетей [4], функции активации радиальных оснований [5], логистическая функция активации и методы оптимизации роевых частиц [7], используют прямые нейронные сети [3], искусственную нейронную сеть Эльмана [10].

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 [18] — 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

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 $\mathcal{D} = (\mathbf{X}, \mathbf{Y})$ — выборка, $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^m \in \mathbb{R}^{m \times n}$ — матрица объектов, $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^m \in \mathbb{R}^{m \times r}$ — 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 (например, r значений сигнала, начиная с момента $n+1$). Матрицы \mathbf{X} и \mathbf{Y} имеют следующую структуру:

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

В авторегрессионной модели матрица ответов представляется в виде

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

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

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 $\mathcal{D}_{\mathcal{L}} \cap \mathcal{D}_{\mathcal{C}} = \emptyset$, $\mathcal{D}_{\mathcal{L}} \cup \mathcal{D}_{\mathcal{C}} = \mathcal{D}$.

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

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

Одним из основных препятствий эффективного применения регрессионного анализа является мультиколлинеарность. Она связана с линейной зависимостью между признаками объектов. В результате мультиколлинеарности матрица парных коэффициентов корреляции и матрица $\mathbf{X}^T \mathbf{X}$ становятся слабообусловленными, т.е. их определители близки к нулю. Это приводит к неустойчивости оценок коэффициентов регрессии, так как в их выражения входит обратная матрица $(\mathbf{X}^T \mathbf{X})^{-1}$, получение которой связано с делением на определитель матрицы $\mathbf{X}^T \mathbf{X}$. Значения сигнала в соседние моменты времени являются зависимыми. Таким образом, наблюдается мультиколлинеарность между признаками

70 авторегрессионной модели прогнозирования сигнала. Для решения проблемы мультикол-
71 линеарности применяются методы отбора признаков. Оптимальные параметры задачи ре-
72 грессии Θ настраиваются на обучающей выборке $\mathcal{D}_{\mathcal{L}}$. Отбор признаков происходит на
73 контрольной выборке $\mathcal{D}_{\mathcal{C}}$.

Метод частных наименьших квадратов (PLS)

Метод частных наименьших квадратов проецирует 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. Размеры векторов в алгоритме можно изобразить следующим образом:

$$\begin{array}{c} \boxed{\mathbf{X}} = \boxed{\mathbf{T}} \times \boxed{\mathbf{P}^\top} + \boxed{\mathbf{E}} \\ \boxed{\mathbf{Y}} = \boxed{\mathbf{U}} \times \boxed{\mathbf{Q}^\top} + \boxed{\mathbf{F}} \end{array}$$

Рис. 1. Размерности векторов в алгоритме PLS

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 вектора \mathbf{w} и \mathbf{c} . Правила обновления векторов \mathbf{w} , \mathbf{c} совпадают с итерацией Power method [20]. Вектора \mathbf{w} , \mathbf{c} являются собственными векторами матриц $\mathbf{X}^\top \mathbf{Y} \mathbf{Y}^\top \mathbf{X}$ и $\mathbf{Y}^\top \mathbf{X} \mathbf{X}^\top \mathbf{Y}$ соответственно.

$$\mathbf{w} \propto \underbrace{\mathbf{X}^\top \mathbf{Y} \mathbf{Y}^\top \mathbf{X}}_{\mathbf{u}} \underbrace{\mathbf{w}}_{\mathbf{t}}$$

Обновляя вектора по данным правилам, мы максимизируем ковариацию между векторами \mathbf{t} и \mathbf{u}

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

87 После завершения внутреннего цикла вычисляются вектора \mathbf{p} , \mathbf{q} . Для перехода на
88 следующий шаг необходимо вычесть из матриц \mathbf{X} и \mathbf{Y} одноранговые аппроксимации.

89 Если ковариация между новыми признаками и ответами максимальна, то можно стро-
90 ить регрессионную модель в пространстве меньшей размерности с сохранением уровня
91 точности прогноза. Параметр метода частных наименьших квадратов $l \in \mathbb{N}$ определяет
92 размерность нового пространства. Отбор признаков осуществляется в виде замены исход-
93 ных 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 $\beta \in \mathbb{R}$ такие, что $\mathbf{u} \approx \mathbf{t}\beta$. Найденные коэффициенты образуют вектор параметров $\beta \in \mathbb{R}^l$. Подставляя вектор параметров в модель (??), получаем

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

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: $\mathbf{w} := \mathbf{X}^\top \mathbf{u} / (\mathbf{u}^\top \mathbf{u})$
 - 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: $\mathbf{X} := \mathbf{X} - \mathbf{t}\mathbf{p}^\top$
 - 15: $\mathbf{Y} := \mathbf{Y} - \beta \mathbf{t}\mathbf{q}^\top$
-

№	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

Модификация метода частных наименьших квадратов (cnPLS)

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.

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

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

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:

$$\check{\mathbf{u}} \approx \check{\mathbf{u}}_0 + \frac{\partial \mathbf{u}}{\partial \mathbf{v}} \Delta \mathbf{v},$$

где $\check{\mathbf{u}}_0$ — это значение функции g при известном значении переменной \mathbf{u} . Для вычисления $\Delta \mathbf{v}$ предложены следующие шаги. Рассматривается разница $\check{\mathbf{u}} - \check{\mathbf{u}}_0 = \frac{\partial \mathbf{u}}{\partial \mathbf{v}} \Delta \mathbf{v}$. Определится рассогласование \mathbf{e}

$$\mathbf{e} = \check{\mathbf{u}} - \check{\mathbf{u}}_0 = \frac{\partial \mathbf{u}}{\partial \mathbf{v}} \Delta \mathbf{v} = \mathbf{J}_u \Delta \mathbf{v},$$

где матрица \mathbf{J}_u состоит из частных производных $\left\{ \frac{\partial \mathbf{u}}{\partial v_i} \right\}_{i=1}^N$, вычисленных при известном значении переменной \mathbf{u} . Таким образом,

$$\begin{aligned} \mathbf{J}_u &= \frac{\partial \mathbf{u}}{\partial \mathbf{v}} = \frac{\partial(\tilde{\mathbf{Y}}\mathbf{c})}{\partial \mathbf{v}} = \frac{\partial(\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^\top \mathbf{t})}{\partial \mathbf{v}} \frac{1}{(\mathbf{t}^\top \mathbf{t})} = \frac{\partial(f(\mathbf{Y}, \mathbf{v})f^\top(\mathbf{Y}, \mathbf{v})\mathbf{t})}{\partial \mathbf{v}} \frac{1}{(\mathbf{t}^\top \mathbf{t})} \\ &= \frac{\partial(f(\mathbf{Y}, \mathbf{v}))}{\partial \mathbf{v}} \frac{f^\top(\mathbf{Y}, \mathbf{v})\mathbf{t}}{(\mathbf{t}^\top \mathbf{t})} + f(\mathbf{Y}, \mathbf{v}) \frac{\partial(f^\top(\mathbf{Y}, \mathbf{v})\mathbf{t})}{\partial \mathbf{v}} \frac{1}{(\mathbf{t}^\top \mathbf{t})}. \end{aligned}$$

110 Далее $\Delta \mathbf{v}$ вычисляется решением задачи регрессии рассогласования \mathbf{e} так, что

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

111 Преобразование независимой переменной

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

114 Рассмотрим преобразования \mathbf{X}

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

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

— нелинейное непараметрическое преобразование

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

— суперпозиция преобразований

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

116 Алгоритм cnPLSR

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

Алгоритм 2 Алгоритм cnlPLSR с преобразованием пространства ответов

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 3 Алгоритм snlPLSR с преобразованием пространства объектов

Вход: $\mathbf{X}, \mathbf{Y}, l$;**Выход:** $\mathbf{T}, \mathbf{U}, \mathbf{P}, \mathbf{Q}$;

- 1: инициализировать \mathbf{v}_x
 - 2: определить \mathbf{t}_0 **КАК**
 - 3: определить \mathbf{u} как вектор матрицы \mathbf{Y}
 - 4: **для** $i = 1, \dots, l$
 - 5: **повторять**
 - 6: $\tilde{\mathbf{X}} := f(\mathbf{X}, \mathbf{v}_x)$
 - 7: $\mathbf{w} := \tilde{\mathbf{X}}^\top \mathbf{u} / (\mathbf{u}^\top \mathbf{u})$
 - 8: нормировать \mathbf{w} : $\|\mathbf{w}\| = 1$
 - 9: $\mathbf{t} := \tilde{\mathbf{X}} \mathbf{w}$
 - 10: $\mathbf{c} := \mathbf{Y}^\top \mathbf{t} / (\mathbf{t}^\top \mathbf{t})$
 - 11: нормировать \mathbf{c} : $\|\mathbf{c}\| = 1$
 - 12: $\mathbf{u} := \mathbf{Y} \mathbf{c}$
 - 13: ? $\mathbf{e} := \mathbf{t} - \mathbf{t}_0$
 - 14: ? $\mathbf{J}_t := \partial \mathbf{t} / \partial \mathbf{v}_x$
 - 15: ? $\Delta \mathbf{v}_x = (\mathbf{J}_t^\top \mathbf{J}_t)^{-1} \mathbf{J}_t^\top \mathbf{e}$
 - 16: $\mathbf{v}_x := \mathbf{v}_x + \Delta \mathbf{v}_x, \|\mathbf{v}_x\| = 1$
 - 17: ? $\mathbf{t}_0 := \mathbf{t}$
 - 18: **пока** \mathbf{t} не перестанет меняться
 - 19: сохранить $\mathbf{t}, \mathbf{u}, \mathbf{c}$
 - 20: вычислить $\tilde{\mathbf{Y}}$
 - 21: $\mathbf{p} := \mathbf{X}^\top \mathbf{t} / (\mathbf{t}^\top \mathbf{t}), \mathbf{q} := \tilde{\mathbf{Y}}^\top \mathbf{u} / (\mathbf{u}^\top \mathbf{u})$
 - 22: регрессия (\mathbf{u} на \mathbf{t}): $\beta := \mathbf{u}^\top \mathbf{t} / (\mathbf{t}^\top \mathbf{t})$
 - 23: $\mathbf{X} := \mathbf{X} - \mathbf{t} \mathbf{t}^\top \mathbf{X} / (\mathbf{t}^\top \mathbf{t})$
 - 24: $\tilde{\mathbf{Y}} := \tilde{\mathbf{Y}} - \beta \mathbf{t} \mathbf{c}^\top$
 - 25: 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.

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.

Литература

- [1] *Thrun, Sebastian and Pratt, Lorien* Learning to learn // Springer Science & Business Media, 2012.
- [2] *Chong, Il Gyo and Jun, Chi Hyuck* Performance of some variable selection methods when multicollinearity is present // *Chemometrics and Intelligent Laboratory Systems*, 2005. Vol. 78. No. 1. P. 103–112.
- [3] *Xuefeng, Yan* Hybrid artificial neural network based on BP-PLSR and its application in development of soft sensors // *Chemometrics and Intelligent Laboratory Systems*, 2010. Vol. 103. No. 2. P. 152–159.
- [4] *Mcavovt, J. and Process, Chemical* Title // *Journal name*, 2005. Vol. 16. No. 4. P. 379–391.
- [5] *Yan, Xuefeng F. and Chen, Dezhao Z. and Hu, Shangxu X.* Chaos-genetic algorithms for optimizing the operating conditions based on RBF-PLS model // *Computers and Chemical Engineering*, 2003. Vol. 27. No. 10. P. 1393–1404.
- [6] *Frank, Ildiko E.* A nonlinear PLS model // *Chemometrics and Intelligent Laboratory Systems*, 1990. Vol. 8. No. 2. P. 109–119.
- [7] *Zhou, Yan Ping and Jiang, Jian Hui and Lin, Wei Qi and Xu, Lu and Wu, Hai Long and Shen, Guo Li and Yu, Ru Qin* Artificial neural network-based transformation for nonlinear partial least-square regression with application to QSAR studies // *Talanta*, 2007. Vol. 71. No. 2. P. 848–853.
- [8] *Chong, Il Gyo and Jun, Chi Hyuck* Notes on the history and nature of partial least squares (PLS) modelling // *Journal of Chemometrics*, 1988. Vol. 2. No. January. P. 231–246.
- [9] *Höskuldsson, Agnar* PLS regression // *Chemometrics and Intelligent Laboratory Systems*, 1987. Vol. 2. No. August. P. 581–591.
- [10] *Bulut, Elif and Egrioglu, Erol* A New Partial Least Square Method Based on Elman Neural Network // *Chemometrics and Intelligent Laboratory Systems*, 2005. Vol. 4. No. 4. P. 154–158.
- [11] *Ng, Kee Siong* A Simple Explanation of Partial Least Squares // *Journal title*, 2013. Vol. volume. No. number. P. 1–10.
- [12] *Rosipal, Roman* Nonlinear partial least squares: An overview // *Chemoinformatics and Advanced Machine Learning Perspectives: Complex Computational Methods and Collaborative Techniques*, 2011. Vol. number. No. number. P. 169–189.
- [13] *Wold, Svante and Kettaneh-Wold, Nouna and Skagerberg, Bert* Nonlinear PLS modeling // *Chemometrics and Intelligent Laboratory Systems*, 1989. Vol. 7. No. 1-2. P. 53–65.
- [14] *Rosipal, Roman and Kramer, Nicole* Overview and Recent Advances in Partial Least Squares // *????? C. Saunders et al. (Eds.): SLSFS 2005, LNCS 3940*, 2006. Vol. ?. No. ?. P. 34–51.
- [15] *Lu, Wen-Cong and Chen, Nian-Yi and Li, Guo-Zheng and Yang, Jie* Multitask Learning Using Partial Least Squares Method // *Proceedings of the Seventh International Conference on Information Fusion; International Society of Information Fusion*, 2004. Vol. 1. P. 79–84.
- [16] *Varnek, Alexandre and Baskin, Igor* Machine learning methods for property prediction in chemoinformatics: Quo Vadis? // *Journal of Chemical Information and Modeling*, 2012. Vol. 52. No. 6. P. 1413–1437.
- [17] *Lehky, Sidney R. and Kiani, Roozbeh and Esteky, Hossein and Tanaka, Keiji* Dimensionality of object representations in monkey inferotemporal cortex // *Neural computation*, 2014. Vol. 1872. No. 10. P. 1840–1872.
- [18] *Abdi, Hervé* Partial Least Squares (PLS) Regression // *Encyclopedia for research methods for the social sciences*, 2003. P. 792–795.
- [19] *Caruana, Rich and de Sa, Virginia R.* Benefitting from the Variables that Variable Selection Discards // *Journal of Machine Learning Research*, 2003. Vol. 3. No. 7-8. P. 1245–1264.

- 202 [20] *Mises R. V., Pollaczek-Geiringer H.* Praktische Verfahren der Gleichungsauflösung // ZAMM-
203 Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und
204 Mechanik, 1929. Vol. 9. No. 1. P. 58–77.