

---

# ВОССТАНОВЛЕНИЕ ТРАЕКТОРИИ ДВИЖЕНИЯ РУКИ ПО ВИДЕО

---

Владимиров Эдуард  
vladimirov.ea@phystech.edu

Курдюкова Антонина  
kurdiukova.ad@phystech.edu

Исаченко Роман  
roman.isachenko@phystech.edu

Стрижов Вадим  
strijov@ccas.ru

30 марта 2022 г.

## АННОТАЦИЯ

В работе решается задача прогнозирования временного ряда со сложной структурой. Под сложной структурой понимается наличие нелинейных зависимостей и варьирующийся период. Требуется найти причинно-следственные связи между временными рядами. Для этого предлагается снизить размерность траекторных пространств. В работе показано, что методы канонического корреляционного анализа, такие как метод главных компонент, метод частичных наименьших квадратов и другие, являются частным случаем метода перекрестных отображений Сугихары. Для демонстрации результатов работы решается задача восстановления траектории движения руки по видео.

**Ключевые слова:** временной ряд · фазовая траектория · траекторное подпространство · сходящееся перекрёстное отображение · частичные наименьшие квадраты · канонический корреляционный анализ

## 1 Введение

В данной работе решается задача прогнозирования временного ряда на основе других временных рядов. Одна из трудностей задачи заключается в обнаружении связи между временными рядами и исключении несвязанных временных рядов из прогностической модели. Решение этой проблемы повышает её качество.

В данной работе применяется метод сходящегося перекрёстного отображения (convergent cross mapping, CCM) [1, 2], который эффективен для временных рядов, порождённых динамической системой. Он основан на сравнении ближайших соседей в траекторном пространстве временного ряда  $\mathbf{x}$ , полученных с помощью ряда  $\mathbf{y}$ .

При построении прогностической модели используется траекторная матрица (или матрица сдвига), описывающая фазовое пространство временного ряда. Например, в методе анализа спектральных компонент (singular spectrum analysis, SSA) [3, 4, 5] прогноз временного ряда основан на спектральном разложении ковариационной матрицы, полученной по траекторной матрице. В ССМ матрицы сдвига используются для проверки наличия липшицева отображения между траекторными пространствами.

Однако размерность траекторного пространства может оказаться чрезмерно высокой, что приводит к неустойчивости прогностической модели. В таком случае необходимо снизить размерность траекторного пространства путём построения проекции фазовой траектории в некоторое подпространство. Для ССМ нет конкретного способа выбрать подпространство, в котором аппроксимируется фазовая траектория. В работе [6] эта проблема решается с помощью сферической регрессии. Согласно этому методу, информация об искомом подпространстве извлекается из множества эмпирических направлений  $\{\mathbf{x}_i - \mathbf{x}_j \mid i < j\}$ . В работе [7] используется автоматический выбор пары главных компонент. Идея заключается в сравнении спектральных плотностей главных компонент. Также используется простой перебор по главным компонентам [8].

Метод проекции в латентное пространство (partial least squares, PLS) [9, 10] отбирает наиболее значимые признаки и строит новые как их линейные комбинации. Это позволяет получить простую, точную и устойчивую прогностическую модель. Наряду с PLS используется метод канонического анализа корреляции (canonical correlation analysis, CCA) [11]. Он похож на PLS за исключением того, что первый метод максимизирует ковариацию между проекциями, а последний — корреляцию. Недостатком этих моделей является их низкая точность при оценивании нелинейных зависимостей между данными. Разработаны нелинейные модели PLS [12, 13] и CCA [14, 15]. В данной статье используется модель NNPLS [16], которая преобразует исходные данные с помощью нейронной сети.

В теоретической части работы показано, что методы снижения размерности CCA и PLS являются частными случаями ССМ. Для этого вводятся различные меры близости между целевой переменной и её аппроксимацией; при этом она является линейной комбинацией её ближайших соседей.

В качестве модели для предсказания временного ряда по набору временных рядов используется алгоритм локально взвешенного глобального линейного отображения (sequential locally weighted global linear map, SMap) [17].

Эксперимент проводится на наборе собранных вручную данных. Он представляет собой совокупность ключевых точек, полученных по видео движения человека, а также показания акселерометра и гироскопа, снятые с руки человека. В эксперименте строится прогноз временных рядов, использующий обнаруженные связанные компоненты временных рядов.

## 2 Постановка задачи

Пусть значения исходного временного ряда  $\mathbf{x}(t)$  доступны в моменты времени  $t = 1, 2, \dots, n$ . Предполагается, что на значения  $\mathbf{x}(t)$  оказывает влияние набор внешних факторов  $\mathbf{y}_1(t), \dots, \mathbf{y}_m(t)$ .

Для прогноза в момент времени  $n$  необходимо определить будущие значения исходного временного ряда  $\mathbf{x}(t)$  в моменты времени  $n + 1, \dots, n + p$ , учитывая влияние внешних факторов  $\mathbf{y}_1(t), \dots, \mathbf{y}_m(t)$ . При этом предполагается, что значения внешних факторов являются доступными в моменты времени:

$$\mathbf{y}_1(n + 1), \dots, \mathbf{y}_1(n + p), \dots, \mathbf{y}_m(n + 1), \dots, \mathbf{y}_m(n + p).$$

Для вычисления будущих значений временного ряда требуется определить функциональную зависимость, отражающую связь между прошлыми значениями  $\mathbf{x}$  и будущими, а также принимающую во внимание влияние внешних факторов  $\mathbf{y}_1, \dots, \mathbf{y}_m$ .

**Опр. 2.1.** Моделью прогнозирования с учётом внешних факторов называется зависимость:

$$\mathbf{x}(t) = F(\mathbf{x}(t - 1), \dots, \mathbf{y}_1(t), \mathbf{y}_1(t - 1), \dots, \mathbf{y}_m(t), \mathbf{y}_m(t - 1), \dots) + \varepsilon_t.$$

Требуется создать такую модель, для которой среднее квадратичное отклонение истинного значения от прогнозируемого стремится к минимальному для заданного  $p$ .

$$\hat{E} = \frac{1}{p} \sum_{i=n+1}^{n+p} \varepsilon_i^2 \rightarrow \min_F.$$

### 2.1 Метод ССМ

Определим траекторную матрицу временного ряда  $\mathbf{x} = [x_1, \dots, x_n]$  следующим образом:

$$\mathbf{H}_{\mathbf{x}} = \begin{bmatrix} x_1 & x_2 & \dots & x_\tau \\ x_2 & x_3 & \dots & x_{\tau+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_N & x_{N+1} & \dots & x_n \end{bmatrix},$$

где  $N$  — число задержек,  $\tau = n - N + 1$ .

Обозначим  $i$ -ый столбец матрицы  $\mathbf{H}_{\mathbf{x}}$  за  $\mathbf{x}_i$ . Матрица  $\mathbf{H}_{\mathbf{x}}$  принимает вид:

$$\mathbf{H}_{\mathbf{x}} = [\mathbf{x}_1, \dots, \mathbf{x}_\tau], \quad \mathbf{x}_i = [x_i, x_{i+1}, \dots, x_{i+N-1}]^T$$

Заметим, что все векторы  $\mathbf{x}_t$  принадлежат  $N$ -мерному траекторному пространству  $\mathbb{H}_{\mathbf{x}} \subseteq \mathbb{R}^N$  ряда  $\mathbf{x}$  и образуют фазовую траекторию  $\mathbf{x}(t) \in \mathbb{R}^N$ .

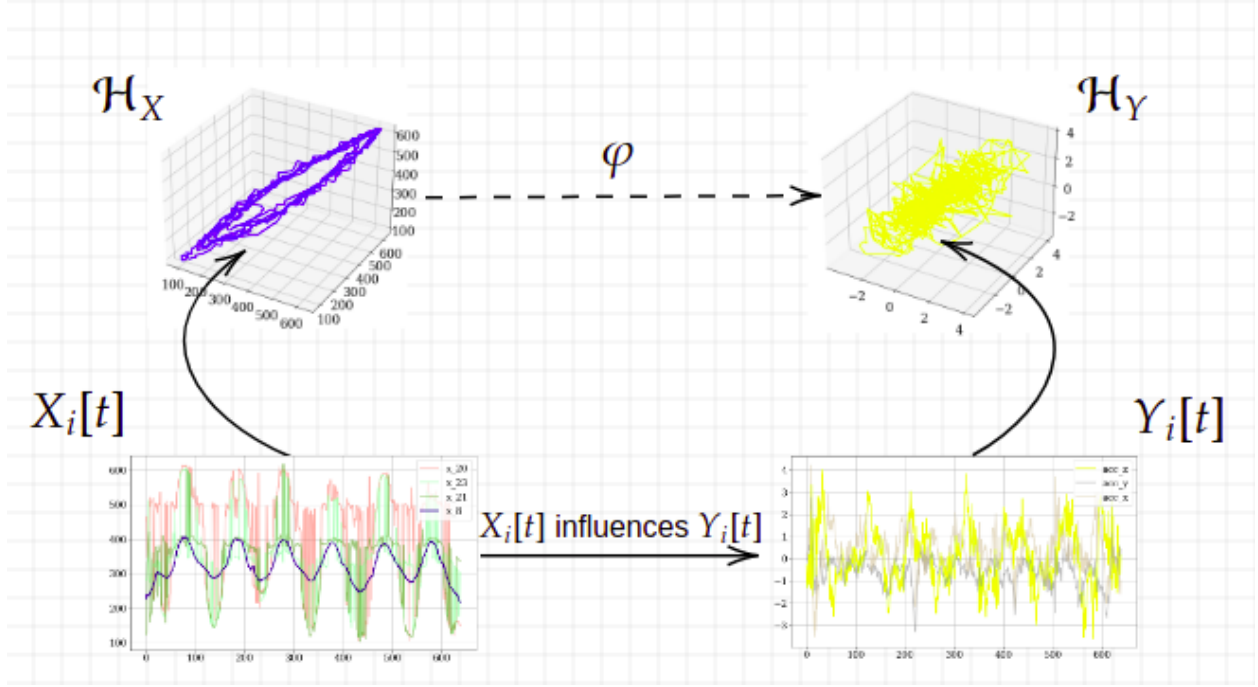


Рис. 1: Блок-схема

Для обнаружения зависимости между временными рядами  $\mathbf{x}$  и  $\mathbf{z}$  возьмём элемент  $\mathbf{x}_0$  из траекторного пространства  $\mathcal{H}_x$  и найдём  $k$  ближайших соседей в этом же пространстве. Обозначим их временные индексы (от ближнего к дальнему)  $t_1, \dots, t_k$ .

Так как оба временных ряда определены на одной временной оси, то по значению временного ряда  $\mathbf{x}$  в момент времени  $t_0 \in \{1, \dots, n\}$  можно однозначно получить значение временного ряда  $\mathbf{z}$  в тот же момент времени, и наоборот. Введём отображение из  $\mathcal{H}_x$  в  $\mathcal{H}_z$  следующим образом:

$$\varphi : \mathbf{x}_0 \mapsto \hat{\mathbf{z}}_0 = \sum_{i=1}^k w_i \mathbf{z}_{t_i}, \quad w_i = \frac{u_i}{\sum_{j=1}^k u_j}, \quad u_i = \exp(-\|\mathbf{x}_0 - \mathbf{x}_{t_i}\|)$$

.

**Опр. 2.2.** Будем считать временные ряды  $\mathbf{x}$  и  $\mathbf{z}$  **связанными**, если отображение  $\varphi$  является липшицевым:

$$\rho_{\mathcal{H}_z}(\varphi(\mathbf{x}_i), \varphi(\mathbf{x}_j)) \leq C \rho_{\mathcal{H}_x}(\mathbf{x}_i, \mathbf{x}_j) \quad \mathbf{x}_i, \mathbf{x}_j \in \mathcal{H}_x.$$

Для проверки наличия связанности введём меру близости векторов в окрестностях  $U_k(\mathbf{x}_{t_0})$  и  $U_k(\mathbf{z}_{t_0})$ :

$$L(\mathbf{x}, \mathbf{z}) = \frac{R(U_k(\mathbf{x}_{t_0}))}{R(U_k(\mathbf{z}_{t_0}))}, \quad R(U_k(\mathbf{x}_{t_0})) = \frac{1}{k} \sum_{i=1}^k \rho_{\mathcal{H}_x}(\mathbf{x}_{t_0}, \mathbf{x}_{t_i}). \quad (1)$$

Если  $L(x, z)$  больше некоторого порога  $C(n)$ , то временной ряд  $\mathbf{z}$  зависит от временного ряда  $\mathbf{x}$ .

## 2.2 Метод PLS

Метод частичных наименьших квадратов восстанавливает связь между наборами данных  $\mathbf{X}$  и  $\mathbf{Y}$ . Матрицы объектов  $\mathbf{X}$  и целевая матрица  $\mathbf{Y}$  проецируются на латентное пространство  $\mathbb{R}^l$  меньшей размерности следующим образом:

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

$n \times m$     $n \times l$     $l \times m$     $n \times m$     $n \times k$     $n \times l$     $l \times k$     $n \times k$

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

Псевдокод метода регрессии PLS приведен в алгоритме 1. Алгоритм итеративно на каждом из  $K$  шагов вычисляет по одному столбцу  $\mathbf{t}_k, \mathbf{u}_k, \mathbf{p}_k, \mathbf{q}_k$  матриц  $\mathbf{T}, \mathbf{U}, \mathbf{P}, \mathbf{Q}$  соответственно. После вычисления следующего набора векторов из матриц  $\mathbf{X}, \mathbf{Y}$  вычитаются очередные одноранговые аппроксимации.

Функция преобразования исходных данных имеет вид:

$$f(\mathbf{X}) = \mathbf{X}\mathbf{W}_x \quad g(\mathbf{Y}) = \mathbf{Y}\mathbf{W}_y,$$

где матрицы весов  $\mathbf{W}_x \in \mathbb{R}^{m \times l}$  и  $\mathbf{W}_y \in \mathbb{R}^{k \times l}$  находятся путём максимизации выборочной ковариации:

$$(\mathbf{W}_x, \mathbf{W}_y) = \operatorname{argmax}_{\mathbf{W}_x, \mathbf{W}_y} \operatorname{Cov}(\mathbf{X}\mathbf{W}_x, \mathbf{Y}\mathbf{W}_y)$$

Из алгоритма PLS можно получить явный вид матриц  $\mathbf{W}_x$  и  $\mathbf{W}_y$ . Заметим, что:

$$\mathbf{X} \cdot \mathbf{A}(\mathbf{P}^\top \mathbf{A})^{-1} = (\mathbf{T}\mathbf{P}^\top \mathbf{A} + \mathbf{E}\mathbf{A})(\mathbf{P}^\top \mathbf{A})^{-1} \approx \mathbf{T},$$

где матрица  $\mathbf{A}$  образована из столбцов  $\mathbf{a}_k$ . Аналогично,  $\mathbf{Y} \cdot \mathbf{B}(\mathbf{Q}^\top \mathbf{B})^{-1} \approx \mathbf{U}$ , где матрица  $\mathbf{B}$  образована из столбцов  $\mathbf{b}_k$ . Таким образом:

$$\mathbf{W}_x = \mathbf{A}(\mathbf{P}^\top \mathbf{A})^{-1} \quad \mathbf{W}_y = \mathbf{B}(\mathbf{Q}^\top \mathbf{B})^{-1}$$

## 2.3 Метод ССА

Канонический корреляционный анализ находит две матрицы перехода в латентные пространства для  $\mathbf{X}$  и  $\mathbf{Y}$  так, что коэффициент корреляции между проекциями является максимальным.

$$(\mathbf{w}_{x_1}, \mathbf{w}_{y_1}) = \operatorname{argmax}_{\mathbf{w}_x, \mathbf{w}_y} \operatorname{Corr}(\mathbf{X}\mathbf{w}_x, \mathbf{Y}\mathbf{w}_y) \quad (2)$$

Первые столбцы матриц весов  $\mathbf{W}_x, \mathbf{W}_y$  находятся путём решения задачи оптимизации (2). Затем ищутся векторы, максимизирующие корреляцию, но с ограничением, что они не коррелируют с первой парой векторов  $\mathbf{w}_{x_1}, \mathbf{w}_{y_1}$ . Процедура продолжается  $l$  шагов, где  $l$  — размерность латентного пространства.

---

**Algorithm 1** Canonical PLS

---

**Require:**  $\mathbf{X} \in \mathbb{R}^{n \times d}$ ,  $\mathbf{Y} \in \mathbb{R}^{n \times t}$ ,  $K \in \mathbb{N}$

**Ensure:**  $\mathbf{T}$ ,  $\mathbf{U}$ ,  $\mathbf{P}$ ,  $\mathbf{Q}$

Нормировать матрицы  $\mathbf{X}$  и  $\mathbf{Y}$  по столбцам

$\mathbf{X}_1 \leftarrow \mathbf{X}$

$\mathbf{Y}_1 \leftarrow \mathbf{Y}$

**for**  $k = 1, \dots, K$  **do**

**if**  $\mathbf{X}_k^T \mathbf{Y}_k = 0$  **then**

**break**

**end if**

    Вычислить  $\mathbf{a}_k \in \mathbb{R}^d$  и  $\mathbf{b}_k \in \mathbb{R}^t$ , первые

    левые и правые сингулярные вектора матрицы  $\mathbf{X}_k^T \mathbf{Y}_k$ .

    Из определения следует, что  $(\mathbf{a}_k, \mathbf{b}_k) = \underset{\mathbf{a}, \mathbf{b}}{\operatorname{argmax}} \operatorname{Cov}(\mathbf{X}_k \mathbf{a}, \mathbf{Y}_k \mathbf{b})$ .

$\mathbf{t}_k \leftarrow \mathbf{X}_k \mathbf{a}_k$

$\mathbf{u}_k \leftarrow \mathbf{Y}_k \mathbf{b}_k$

$\mathbf{p}_k^T = (\mathbf{t}_k^T \mathbf{t}_k)^{-1} \mathbf{t}_k^T \mathbf{X}_k$

$\triangleright$  Вектора  $\mathbf{p}_k$  образуют матрицу  $\mathbf{P}$

$\mathbf{q}_k^T = (\mathbf{u}_k^T \mathbf{u}_k)^{-1} \mathbf{u}_k^T \mathbf{Y}_k$

$\triangleright$  Вектора  $\mathbf{q}_k$  образуют матрицу  $\mathbf{Q}$

$\widehat{\mathbf{X}}_k(\mathbf{t}_k) = \mathbf{t}_k \mathbf{p}_k^T$

$\triangleright$  Регрессируем  $\mathbf{X}_k$  по  $\mathbf{t}_k$

$\widehat{\mathbf{Y}}_k(\mathbf{u}_k) = \mathbf{u}_k \mathbf{q}_k^T$

$\triangleright$  Регрессируем  $\mathbf{Y}_k$  по  $\mathbf{u}_k$

$\mathbf{X}_{k+1} \leftarrow \mathbf{X}_k - \widehat{\mathbf{X}}_k(\mathbf{t}_k)$

$\mathbf{Y}_{k+1} \leftarrow \mathbf{Y}_k - \widehat{\mathbf{Y}}_k(\mathbf{u}_k)$

**end for**

---

## 2.4 Алгоритм предсказания

Пусть задана обучающая выборка

$$\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid i = 1, \dots, d\} = (\mathbf{X}, \mathbf{y}),$$

где  $\mathbf{x}_i$  — элемент траекторного пространства, а  $y_i$  — будущее значение временного ряда. В одномерном случае:

$$\mathbf{x}_i = [x_i, x_{i+1}, \dots, x_{i+N-1}]^T, \quad y_i = x_{i+N}.$$

Для многомерного временного ряда его предсказанием служит  $k$ -ая компонента, а в качестве  $\mathbf{x}_i$  выступает его значение в момент времени  $i$ . Поэтому алгоритм запускается  $D$  раз, где  $D$  — размерность временного ряда.

Пусть алгоритм предсказывает значение временного ряда в момент времени  $t_0$ . Вначале каждому  $\mathbf{x}_i \in X \setminus \{x_{t_0}\}$  поставим в соответствие:

$$\mathbf{w}_i = \exp \left( - \frac{\theta \cdot \|\mathbf{x}_i - \mathbf{x}_{t_0}\|}{\frac{1}{d-1} \sum_{j=1, j \neq t_0}^d \|\mathbf{x}_j - \mathbf{x}_{t_0}\|} \right),$$

где  $\theta$  — коэффициент локализации. Таким образом, близкие к  $\mathbf{x}_{t_0}$  соседи имеют больший вес, чем дальние. Затем элементы  $\mathbf{x}_i$  и  $y_i$  умножаются на вычисленные веса  $\mathbf{w}_i$

Для прогнозирования временного ряда используется авторегрессионная модель порядка  $t_0 - 1$ :

$$X_{t_0} = \mu + \psi_1 X_{t_0-1} + \dots + \psi_{t_0-1} X_1 + u_{t_0}, \quad u_t \sim \mathcal{N}(0, \sigma^2), \quad \psi_p \neq 0.$$

Обозначим полученный прогноз  $\widehat{X}_{t_0}$ . Оптимальность прогноза будем понимать в смысле среднего квадратичного отклонения:

$$\min_{\psi_1, \dots, \psi_{t_0-1}} E(\widehat{X}_{t_0} - y_0)^2.$$

### 3 Вычислительный эксперимент

Целью эксперимента является сравнение различных стратегий снижения размерности целевого пространства. Важной его частью является изучение результатов алгоритма прогнозирования временного ряда, применённого к элементам пространства фазовых траекторий и траекторного подпространства меньшей размерности.

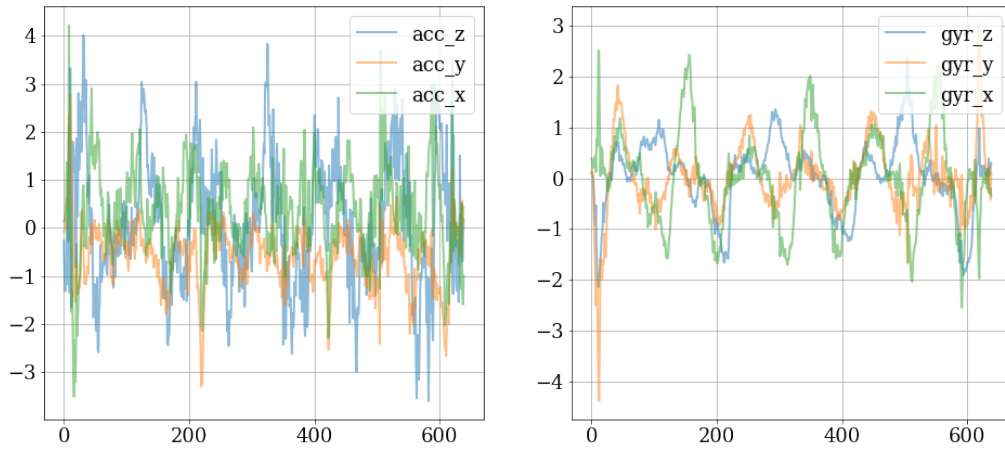


Рис. 2: Данные акселерометра и гироскопа

Первоначальные данные представляют собой набор видеороликов, на которых выполняются различные движения руками (циклические и хаотические), а также показания акселерометра и гироскопа частотой в 100 Герц, закреплённых на одной из рук. Далее по видеоряду с помощью фреймворка `alphapose` [links] получаются координаты конечностей, а именно 68 ключевых точек. Затем из полученного временного ряда исключаются сильно скореллированные компоненты. После этого полученные многомерные временные ряды приводятся к одной временной шкале с помощью удаления элементов более длинного временного ряда.

Перед началом эксперимента зафиксируем следующие переменные:  $N$  — размерность траекторного пространства,  $k$  — число ближайших соседей, рассматриваемых в ССМ,  $n_{tr}$  — размер обучающей выборки,  $E_{vid}$  — число признаков, полученных из видеоряда, которые будут использованы в алгоритме. Для каждой пары компонент, взятых из разных временных рядов, применим метод ССМ. В результате получим матрицу корреляций, в которой на  $i$ -ой строке и  $j$ -ом столбце стоит коэффициент корреляции Пирсона между  $i$ -ой компонентной "приборного" временного ряда и  $j$ -ой компонентой ряда, восстановленного по видео. После этого для каждого целевого признака выбираем  $E_{vid}$  видео-признаков, обладающих максимальной корреляцией в соответствующей строке. Затем на основе этих признаков обучается алгоритм.

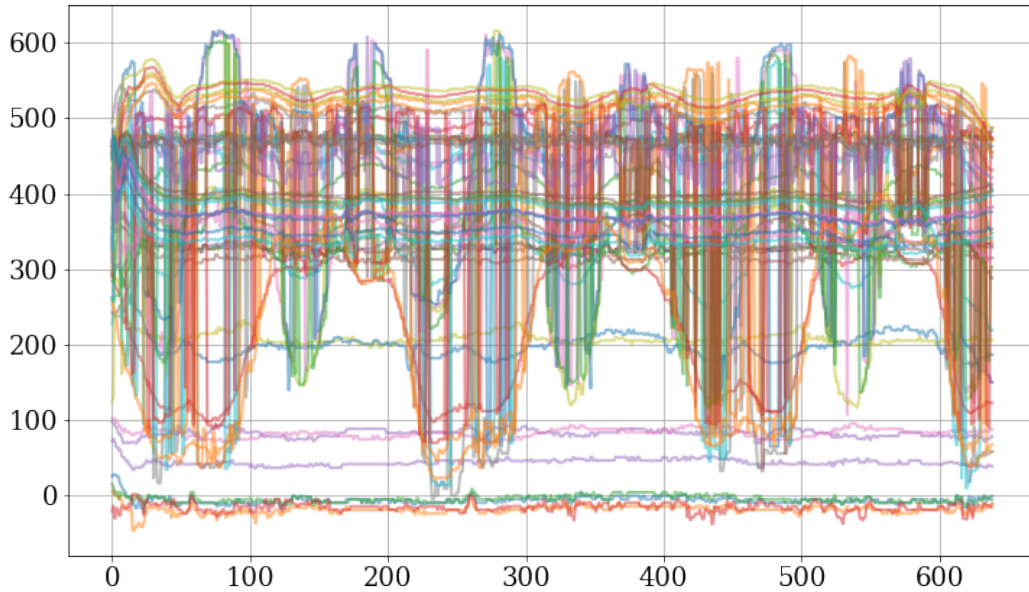


Рис. 3: Данные кейпоинтов, полученные по видео



Таблица 1: Сравнение ошибки (MSE) предсказательной модели, применённой в траекторном пространстве и в его подпространстве, полученном CCM

	acc_z	acc_y	acc_x	gyr_z	gyr_y	gyr_x
space	$1.053 \pm 2.223$	$0.401 \pm 0.833$	$0.483 \pm 0.825$	$0.084 \pm 0.537$	$0.090 \pm 0.094$	$0.063 \pm 0.29$
subspace	$0.315 \pm 0.461$	$0.043 \pm 0.051$	$0.150 \pm 0.177$	$0.001 \pm 0.001$	$0.015 \pm 0.031$	$0.001 \pm 0.00$

## 4 Анализ ошибки

Для начала сравним качество предсказаний прогностической модели, применённой в траекторном пространстве и его подпространстве. В таблице 1 представлены среднеквадратичная ошибки предсказаний значений акселерометра и гироскопа по каждой из осей и их стандартные отклонения. На ней видно, что прогностическая модель, применённая в траекторном подпространстве, даёт более точные предсказания, поскольку большинство признаков исходного признакового пространства неинформативно.

Далее, рассмотрим различные методы снижения размерности траекторного пространства.

Таблица 2: Среднеквадратичное отклонение между истинными показаниями устройств и предсказаниями, полученными с помощью одного из методов снижения размерности

Целевой признак \ Метод		Метод			
		CCM	PLS	CCA	Naive
cyclic	acc_z	0.163	<b>0.040</b>	0.116	0.141
	acc_y	0.009	<b>0.007</b>	0.011	0.008
	acc_x	<b>0.044</b>	0.045	0.089	0.049
	gyr_z	<b>0.000</b>	0.001	0.001	0.001
	gyr_y	<b>0.002</b>	0.004	0.005	0.003
	gyr_x	0.009	0.004	0.004	<b>0.003</b>
chaotic	acc_z	<b>0.315</b>	0.416	0.416	0.331
	acc_y	<b>0.043</b>	0.045	0.429	0.055
	acc_x	0.150	0.177	0.221	<b>0.143</b>
	gyr_z	<b>0.001</b>	0.002	0.003	0.003
	gyr_y	<b>0.015</b>	0.022	0.061	0.026
	gyr_x	<b>0.001</b>	0.013	0.015	0.008

## 5 Заключение

TODO

## Список литературы

- [1] G. Sugihara, B. Grenfell, and R. M. May. Distinguishing error from chaos in ecological time series. *Phil. Trans. Roy. Soc. London B*, 330(1257):235–51, 1990.

- [2] George Sugihara and Robert M May. Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. *Nature*, 344(6268):734–741, 1990.
- [3] Nina Golyandina and D Stepanov. Ssa-based approaches to analysis and forecast of multidimensional time series. In *proceedings of the 5th St. Petersburg workshop on simulation*, volume 293, page 298. St. Petersburg State University St. Petersburg, Russia, 2005.
- [4] Nina Golyandina, Vladimir Nekrutkin, and Anatoly A Zhigljavsky. *Analysis of time series structure: SSA and related techniques*. CRC press, 2001.
- [5] Anatoly Zhigljavsky. Singular spectrum analysis for time series: Introduction to this special issue. *Statistics and its Interface*, 3(3):255–258, 2010.
- [6] Карина Равилевна Усманова, ЮИ Журавлёв, КВ Рудаков, and ВВ Стрижов. Аппроксимация фазовой траектории квазипериодических сигналов методом сферической регрессии. *Вестник Московского университета. Серия 15: Вычислительная математика и кибернетика*, (4):40–46, 2020.
- [7] Th Alexandrov and N Golyandina. Automatic extraction and forecast of time series cyclic components within the framework of ssa. In *Proceedings of the 5th St. Petersburg Workshop on Simulation*, pages 45–50. St. Petersburg State University St. Petersburg, 2005.
- [8] Карина Равилевна Усманова and Вадим Викторович Стрижов. Модели обнаружения зависимостей во временных рядах в задачах построения прогностических моделей. *Системы и средства информатики*, 29(2):12–30, 2019.
- [9] Roman Rosipal. Nonlinear partial least squares an overview. *Chemoinformatics and advanced machine learning perspectives: complex computational methods and collaborative techniques*, pages 169–189, 2011.
- [10] Roman Rosipal and Nicole Kramer. Overview and recent advances in partial least squares. In *International Statistical and Optimization Perspectives Workshop Subspace, Latent Structure and Feature Selection*, pages 34–51. Springer, 2005.
- [11] David R Hardoon, Sandor Szedmak, and John Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural computation*, 16(12):2639–2664, 2004.
- [12] S Joe Qin and Thomas J McAvoy. Nonlinear pls modeling using neural networks. *Computers & Chemical Engineering*, 16(4):379–391, 1992.
- [13] Hugo Hiden, Ben McKay, Mark Willis, and M Tham. Non-linear partial least squares using genetic programming. *Genetic programming*, pages 128–133, 1998.
- [14] Pei Ling Lai and Colin Fyfe. Kernel and nonlinear canonical correlation analysis. *International Journal of Neural Systems*, 10(05):365–377, 2000.
- [15] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *International conference on machine learning*, pages 1247–1255. PMLR, 2013.
- [16] Elif Bulut and Erol Egrioglu. A new partial least square method based on elman neural network. *American Journal of Intelligent Systems*, 4(4):154–158, 2014.

- [17] George Sugihara. Nonlinear forecasting for the classification of natural time series. *Philosophical Transactions of the Royal Society of London. Series A: Physical and Engineering Sciences*, 348(1688):477–495, 1994.