

Яндекс



**Skoltech**  
Skolkovo Institute of Science and Technology

# ICA и матричные разложения в нейронауках

ПМИ ФКН ВШЭ, 8 декабря 2018 г.

Максим Шараев<sup>1</sup>

<sup>1</sup> Сколтех

# Содержание лекции

- › Что делать после PCA?
- › Метод независимых компонент (ICA)
  - › независимость и декоррелированность
- › Неотрицательное матричное разложение (NNMF)

# PCA и декорреляция

# PCA и декорреляция

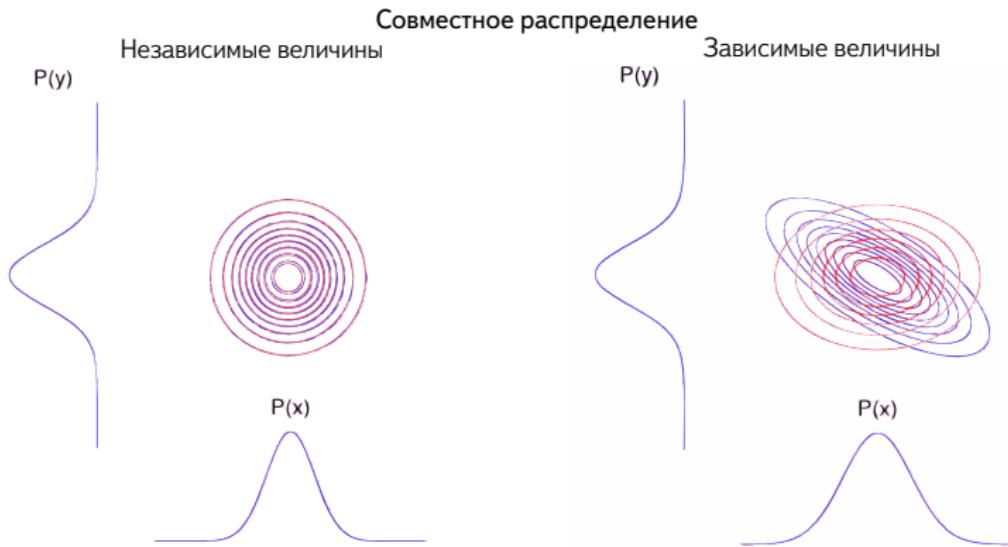
- › Цель PCA:
  - › Диагонализация ковариационной матрицы
    - › т.е. декорелировать активацию признаков
- › Зачем?
  - › Хотим иметь статистически независимую активацию признаков
    - › Чтобы они лучше отражали структуру данных

# Статистическая независимость

# Статистическая независимость

- Мы определяем статистическую независимость как

$$P(x, y) = P(x)P(y)$$



# Статистическая независимость

- › Мы определили статистическую независимость как

$$P(x, y) = P(x)P(y)$$

- › Что подразумевает

$$E\{f(x)g(y)\} = E\{f(x)\}E\{g(y)\}$$

$$\forall f, g$$

- › По сути независимость означает, что мы ничего не можем сказать про X, если наблюдаем Y

# Некоррелированность и независимость

# Некоррелированность и независимость

- › Некоррелированность не всегда влечет независимость!
  - › Некоррелированность:  $E\{xy\} = E\{x\}E\{y\}$
  - › Независимость:  $E\{f(x)g(y)\} = E\{f(x)\}E\{g(y)\}$
- › Но независимость всегда влечет некоррелированность
  - › Когда  $f(x) = x$  и  $g(y) = y$
  - › Некоррелированность - подмножество независимости

# Независимость и некоррелированность

- › Пример с дискретными величинами:
  - › Коррелируют ли они?

		$x = -1$	$x = 0$	$x = 1$
$y = -1$	0	1/4	0	
$y = 0$	1/4	0	1/4	
$y = 1$	0	1/4	0	

# Независимость и некоррелированность

- › Пример с дискретными величинами:
  - › Коррелируют ли они?

		$x = -1$	$x = 0$	$x = 1$
$y = -1$	0	$1/4$	0	
$y = 0$	$1/4$	0	$1/4$	
$y = 1$	0	$1/4$	0	

- ›  $x, y$  не коррелируют:

$$E\{xy\} = E\{x\}E\{y\} = 0$$

# Независимость и некоррелированность

- › Пример с дискретными величинами:
  - › Являются ли они зависимыми?

	$x = -1$	$x = 0$	$x = 1$
$y = -1$	0	1/4	0
$y = 0$	1/4	0	1/4
$y = 1$	0	1/4	0

# Независимость и некоррелированность

- › Пример с дискретными величинами:
  - › Являются ли они зависимыми?

	$x = -1$	$x = 0$	$x = 1$
$y = -1$	0	1/4	0
$y = 0$	1/4	0	1/4
$y = 1$	0	1/4	0

- › Да,  $x$  и  $y$  являются зависимыми:

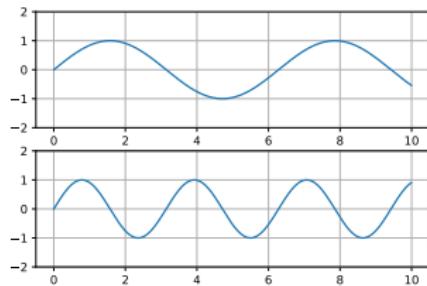
$$E\{x^2y^2\} = 0 \neq E\{x^2\}E\{y^2\} = 1/4$$

# Пример с сигналами

- › Являются ли  $x, y$  некоррелированными?

$$x = \sin(t)$$

$$y = \sin(2t)$$



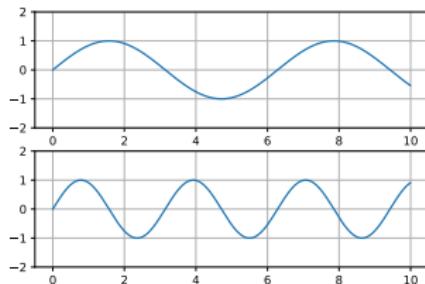
# Пример с сигналами

- › Являются ли  $x, y$  некоррелированными?

$$x = \sin(t)$$

$$y = \sin(2t)$$

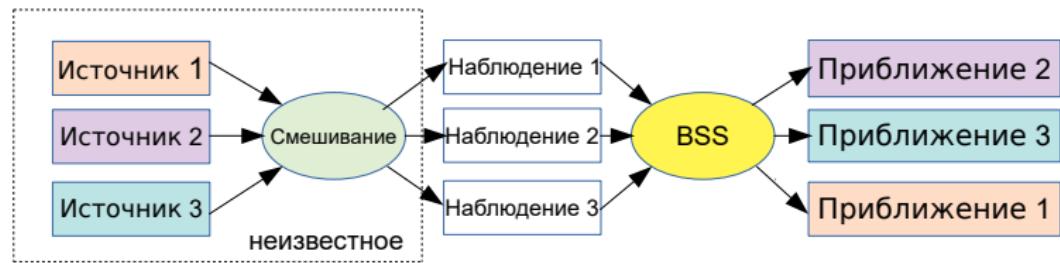
- › Да:  $E\{xy\} = 0$ 
  - › Но можно предсказать один, взглянув на другой
  - › Значит, они зависимы



Выделение  
независимых  
компонент

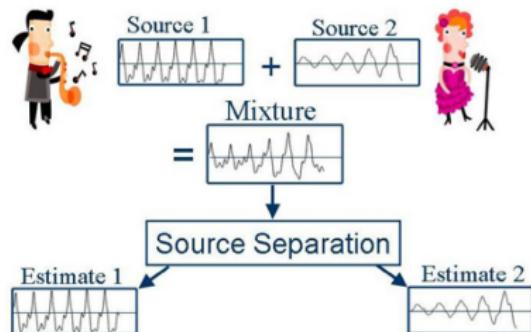
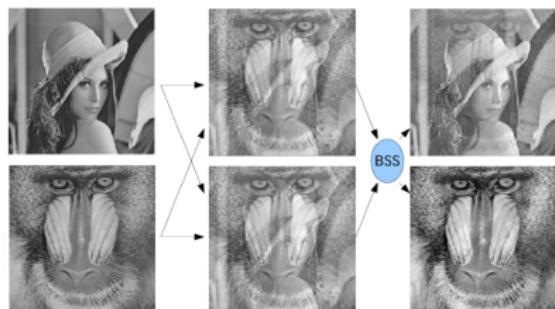
# Слепое разделение сигналов (Blind Source Separation)

- › BSS - это метод приближения исходных сигналов по наблюдаемым, которые могут содержать смесь исходных сигналов и шум



# Пример задачи - cocktail party

- › BSS часто используется для анализа речи и изображений



# Формальная постановка задачи

- › Пример коктейльной вечеринки

$$x_1(t) = a_{11}s_1(t) + a_{12}s_2(t) + a_{13}s_3(t)$$

$$x_2(t) = a_{21}s_1(t) + a_{22}s_2(t) + a_{23}s_3(t)$$

$$x_3(t) = a_{31}s_1(t) + a_{32}s_2(t) + a_{33}s_3(t)$$

- ›  $x$  - наблюдаемый сигнал, а  $s$  - исходный сигнал.
- › предполагаем, что  $s_1$ ,  $s_2$ ,  $s_3$  попарно независимы
- › Метод должен оценить независимые компоненты  $s(t)$  по  $x(t)$

$$x(t) = A \cdot s(t)$$

# Как же получить независимость?

- › Существует несколько способов
  - › Семейство алгоритмов ICA (independent component analysis)
- › Формальное определение:

$$y = W \cdot x,$$

где  $x$  - входной сигнал,  $W$  - обратная к матрице смешивания,  $y$  - оценка независимых компонент

$$P(y_i, y_j) = P(y_i)P(y_j) \forall i, j$$

# Подход 1

- › Нелинейная декорреляция (предполагаем, что матожидание входа нулевое)
  - › Цель:  $E\{f(y_i)g(y_j)\} = 0$  для фиксированных  $f, g$
- › Алгоритм Cichocki-Unbehauen
  - › Прекратить, когда достигнута независимость

do

$$\Delta W \propto (D - f(y_i) \cdot g(y_i^T)) \cdot W$$

$$W = W + \mu \Delta W$$

$$D = \begin{bmatrix} d_1 & & 0 \\ & \ddots & \\ 0 & & d_n \end{bmatrix}$$

repeat

$f(x), g(x)$  могут быть  $\tanh(x), x^3, \dots$

# Подход 2

- › "Диагонализация" высоких порядков
  - › В PCA мы диагонализуем ковариационную матрицу ( $N \times N$  - двумерный объект)

$$Cov(y)_{i,j} = E\{y_i y_j\} = \kappa_2(y_i, y_j)$$

- › В ICA мы диагонализуем "квадриковариационный" тензор (четырехмерный объект)

$$\begin{aligned} Q(y)_{i,j,k,l} &= \kappa(y_i, y_j, y_k, y_l) = E\{y_i y_j y_k y_l\} - \\ &E\{y_i y_j\} E\{y_k y_l\} - E\{y_i y_k\} E\{y_j y_l\} - E\{y_i y_l\} E\{y_j y_k\} \end{aligned}$$

# Подход 2

- › Идейно мы производим сингулярное разложение тензора
- › Алгоритм Комона (Comon P., 1994)
  - › Делаем РСА (декорреляция)
  - › Находим унитарное преобразование, минимизирующее кросс-кумулянты четвертого порядка

# Подход 3

- › Подход на основе теории информации
  - › Минимизируем взаимную информацию:

$$I(\mathbf{y}) = \sum H(y_k) - H(\mathbf{y})$$

$$H(y) = - \int p_y(\eta) \log p_y(\eta) d\eta$$

- › Что подразумевает минимизацию:

$$D(\mathbf{y}) = - \int P(\mathbf{y}) \log \frac{P(\mathbf{y})}{\prod P(y_k)}$$

## Подход 4

- › Негауссовость - мера независимости
- › Из ЦПТ следует, что гауссовость  $x(t)$  должна быть больше гауссности  $s(t)$
- › Хотим максимизировать негауссовость
- › Негэнтропия определяется как

$$J(\mathbf{y}) = H(\mathbf{y}_{Gauss}) - H(\mathbf{y}),$$

- ›  $\mathbf{y}_{Gauss}$  - вектор с гауссовым распределением с  $\mu = E(y)$  и  $\sigma = \sqrt{E((y - \mu)^2)}$
- › Если  $y$  - распределена по гауссу, то  $J(y) = 0$

# Подход 5, 6, ...

- › Метод максимального правдоподобия
- › FastICA
  - › Быстрый алгоритм с вычислениями фиксированной точности
- › Нейронные сети
  - › Напрямую оптимизируем KL-дивергенцию / взаимную информацию
- › Negentropy
  - › Мера негауссности

# Какой подход лучше?

- › Как обычно, конкретного ответа нет
- › Алгебраические алгоритмы
  - › тензорные разложения, etc.
  - › Вычислительно сложные
- › Итеративные алгоритмы
  - › Нелинейная декорреляция, infomax, etc.
  - › Простые и быстрые, но могут быть неустойчивы
- › FastICA
  - › Достаточно устойчивый и надежный

Как работает ICA?

# Как работает ICA?

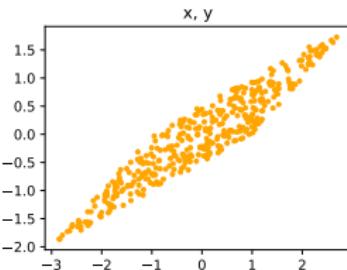
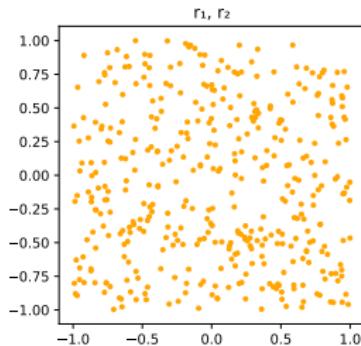
- › Возьмем две равномерно распределенные случайные величины и перемешаем их

$$r_1, r_2 \sim U(-1, 1)$$

$$x = 2r_1 + r_2$$

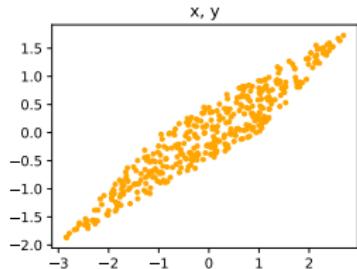
$$y = r_1 + r_2$$

- › Это создаст зависимые  $x$  и  $y$ 
  - › Это видно на графике как поворот и растягивание данных

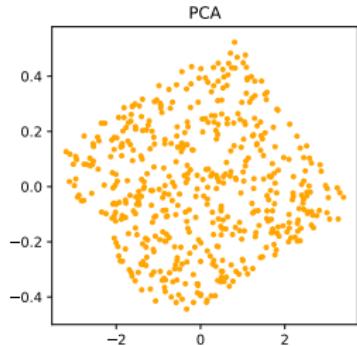


# Применение PCA

- › PCA: декорреляция
  - › PCA основывается на направлении максимального разброса
- › Полученная проекция не привела к независимости в данных

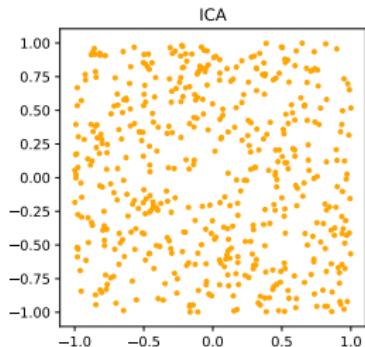
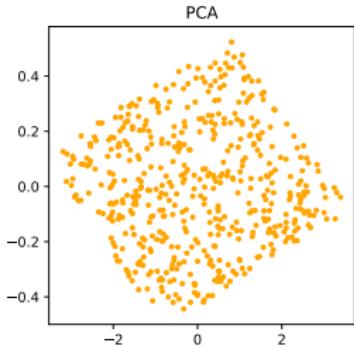
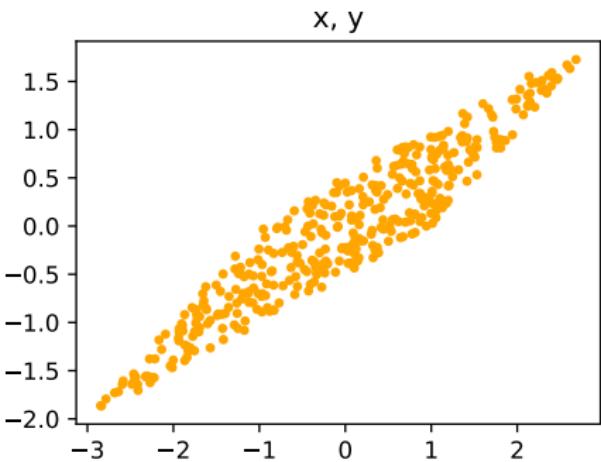


↓ PCA



# Что делает ICA?

- › Результат ICA независим
  - › Мы получили исходные СВ, которые подавали на вход



# Проблемы ICA

- › Большинство численных оценок (эстиматоры) приблизительны
  - › Результат работы не обязательно верен (находим не всегда то что искали)
  - › Может меняться между запусками и реализациями алгоритма
- › В данных может не быть независимости
  - › ICA возвращает максимально независимую проекцию, но не обязательно независимую
  - › В результате может получиться не то, что ожидали

# Ограничения ICA

- › Только линейные связи
- › Инвариантность к перестановкам на выходе

$$P(y_1, y_2, y_3) = P(y_1)P(y_2)P(y_3) = P(y_2)P(y_1)P(y_3) = \dots$$

- › Порядок результата не гарантируется и может различаться между запусками
- › Не упорядочивает компоненты
  - › PCA упорядочивает результаты по их значимости в дисперсии
  - › ICA никак не упорядочивает
    - › Как следствие мы не можем понизить размерность

# Совмещение PCA и ICA

- › Если нужно понизить размерность, перед ICA запускаем PCA
  - › 1) Используем PCA для понижения размерности
  - › 2) Используем ICA, чтобы установить независимость
    - › Применяем ICA к выходу PCA

# Выделение признаков

- › Чем различаются признаки, полученные ICA и PCA?
- › ICA признаки более компактные/разреженные
  - › PCA признаки часто имеют зависимости

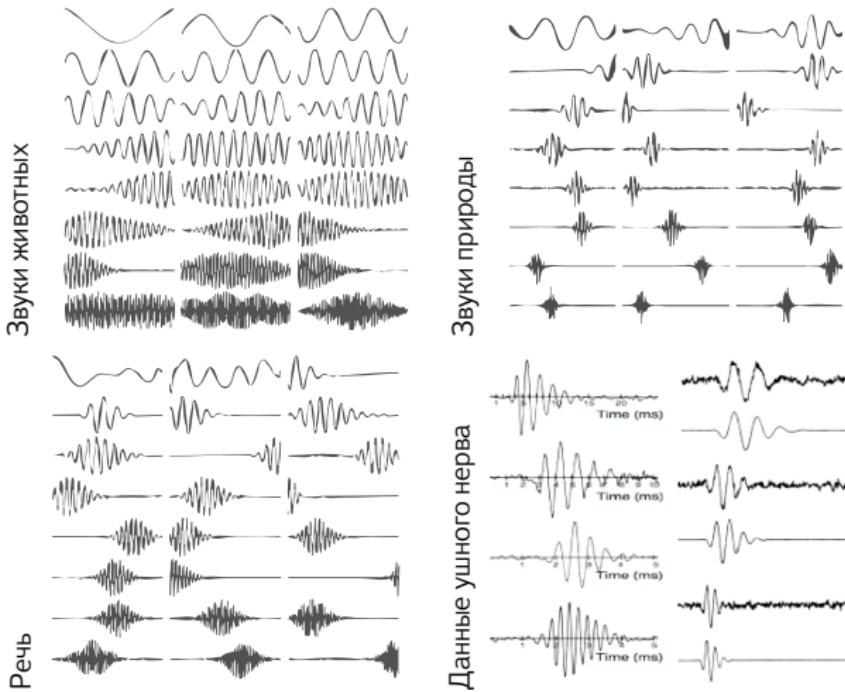
# Пример: признаки из звуков

- › Собираем много естественных звуков
  - › Звуки природы, птиц, ходьба по листьям и т.п.
- › Помещаем маленькие окна в большую матрицу
  - › Применяем PCA и ICA

$$Z = W \cdot \begin{bmatrix} x(t) & x(t+1) & \dots \\ \vdots & \vdots & \vdots \\ x(t+N) & x(t+1+N) & \dots \end{bmatrix}$$

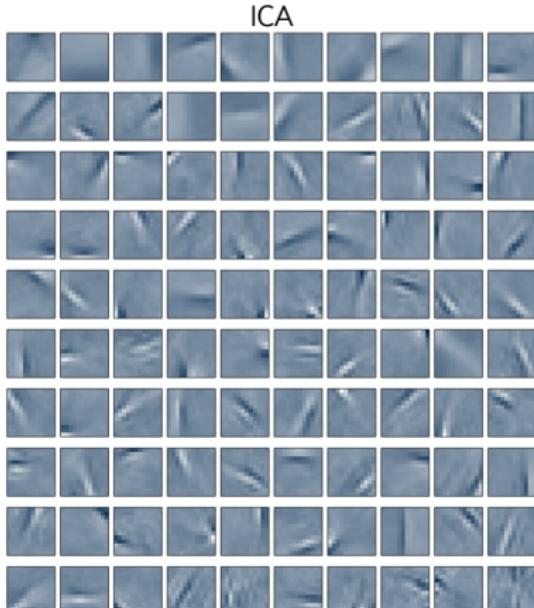
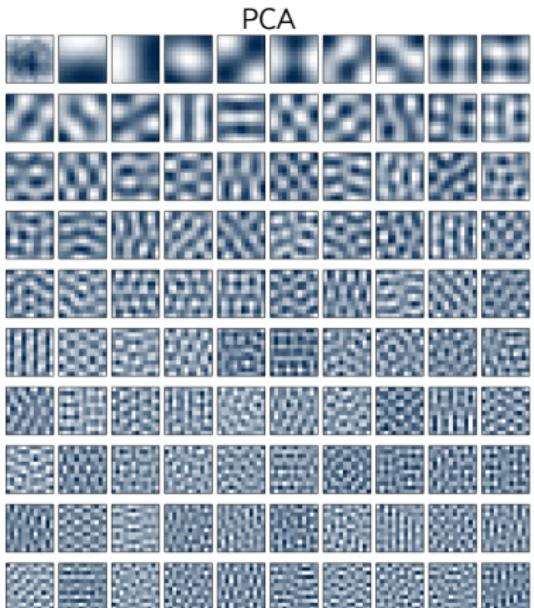
- › Мы знаем, что результат PCA - это синусоиды

# Пример: признаки из звуков



# То же с изображениями

- ICA компоненты похожи на первичные зрительные зоны

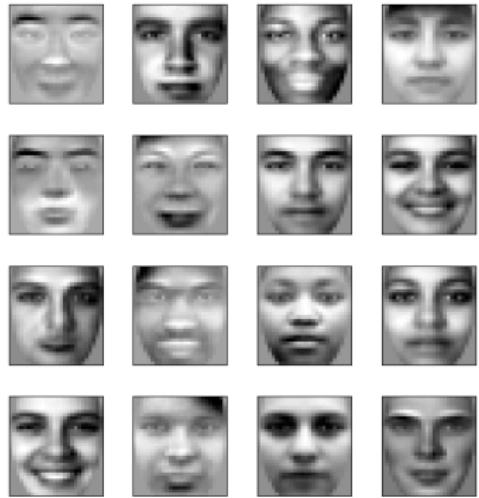


# Изображения лиц

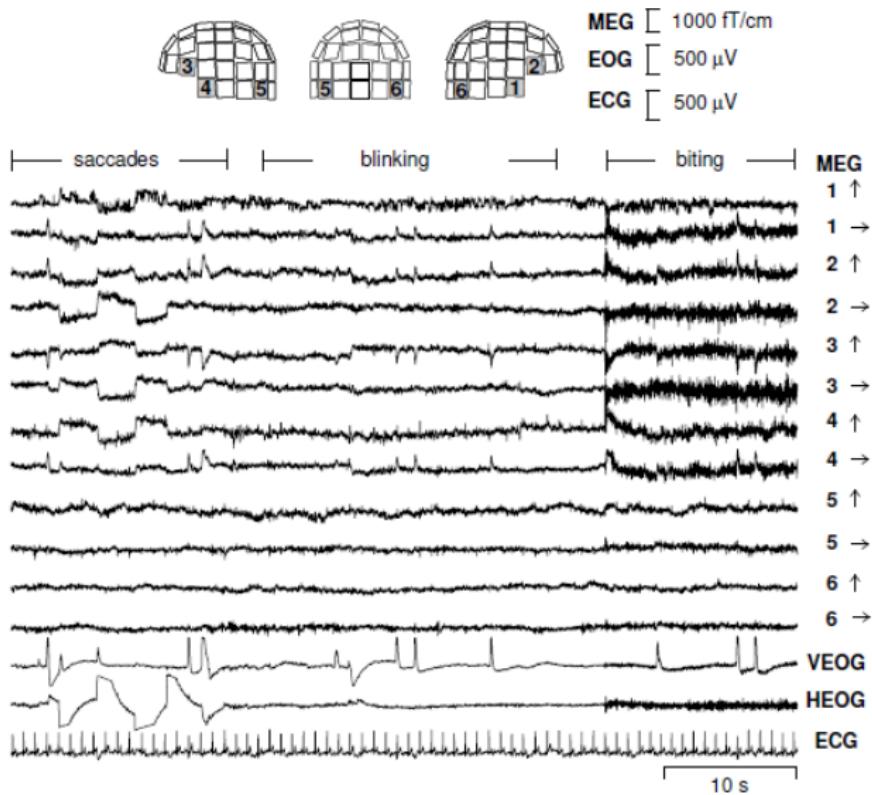
Eigenfaces



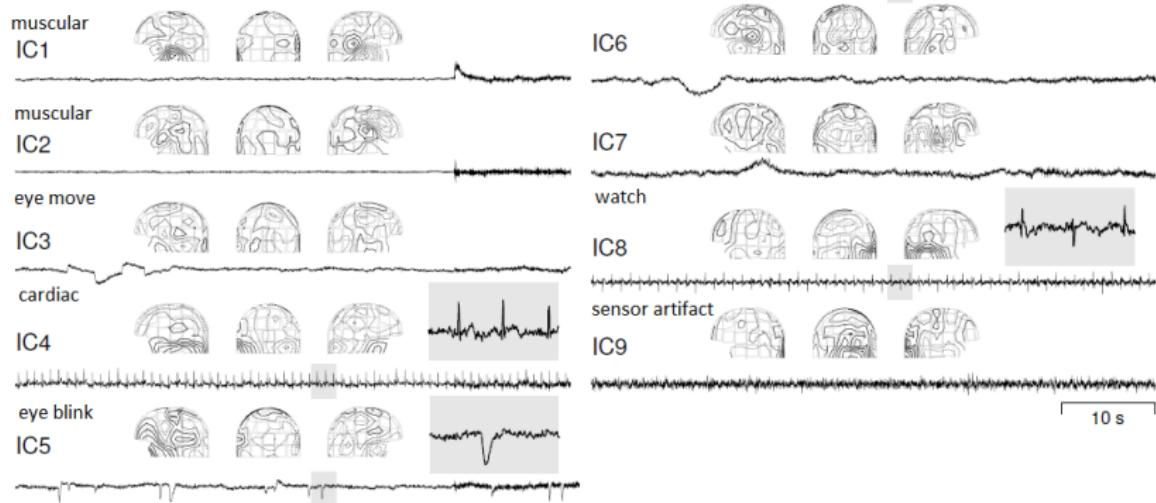
ICA-faces



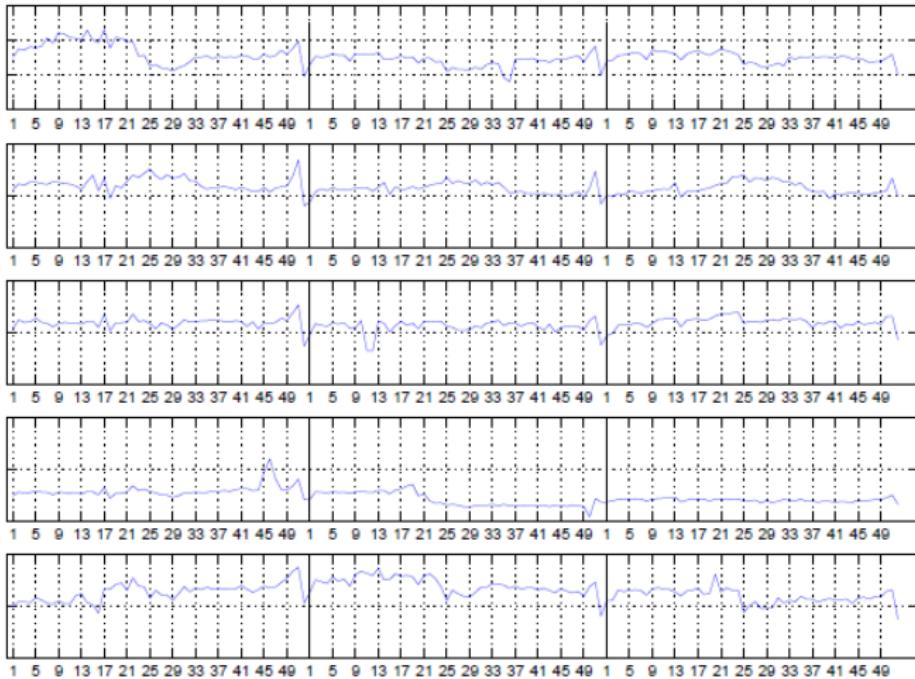
# Данные магнитоэнцефалографии (MEG)



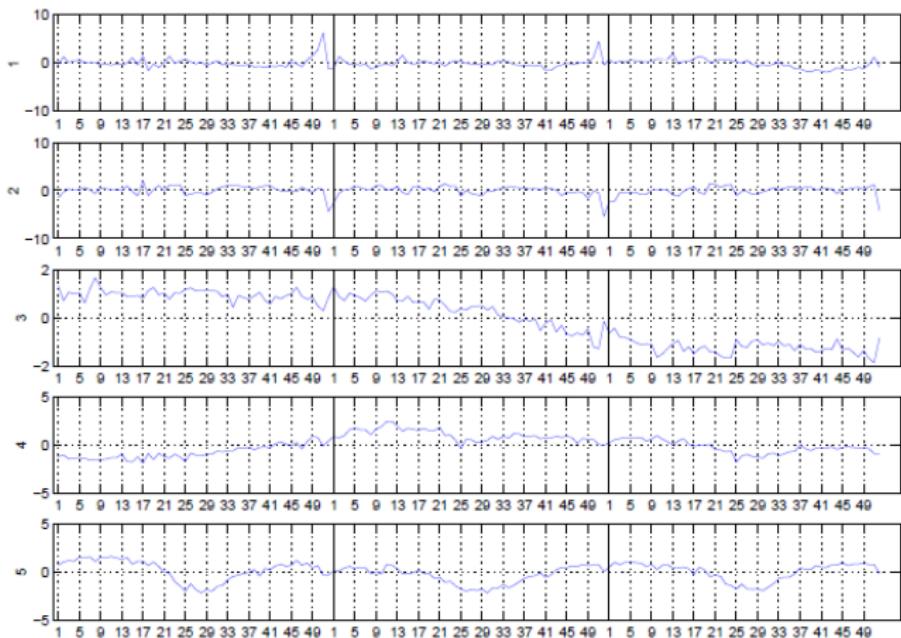
# Данные магнитоэнцефалографии (MEG)



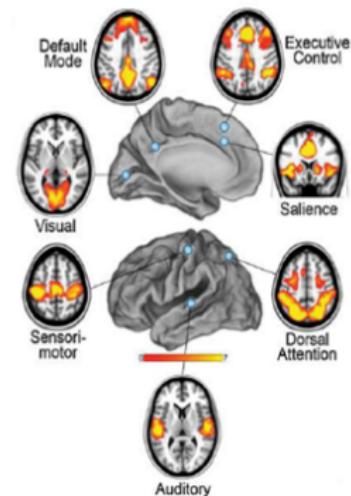
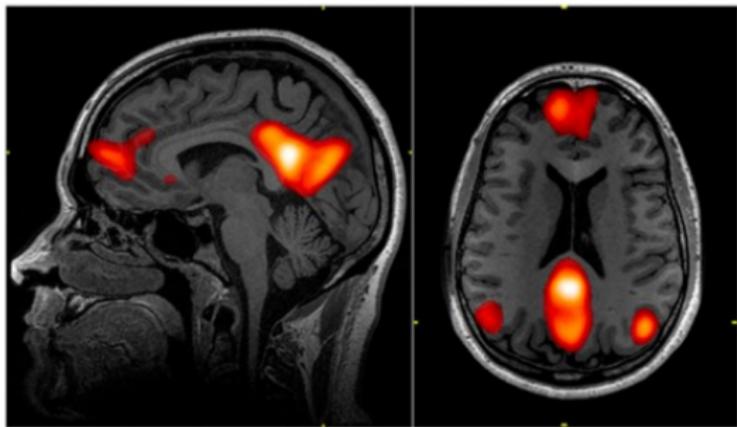
# Оборот магазинов сети



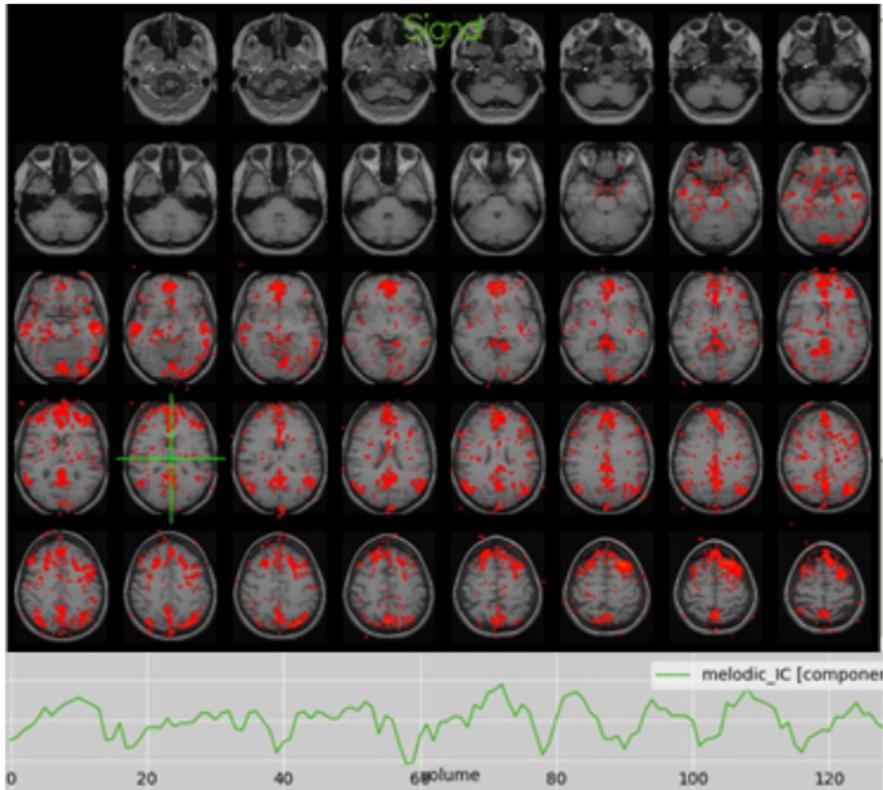
# Оборот магазинов сети



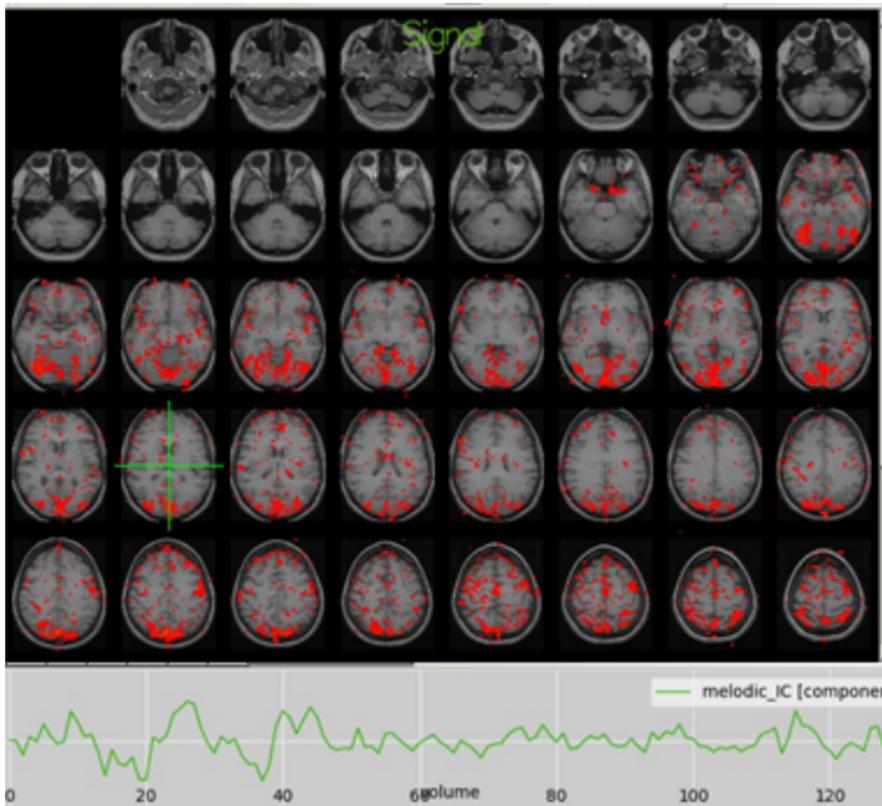
# фМРТ: сети состояния покоя



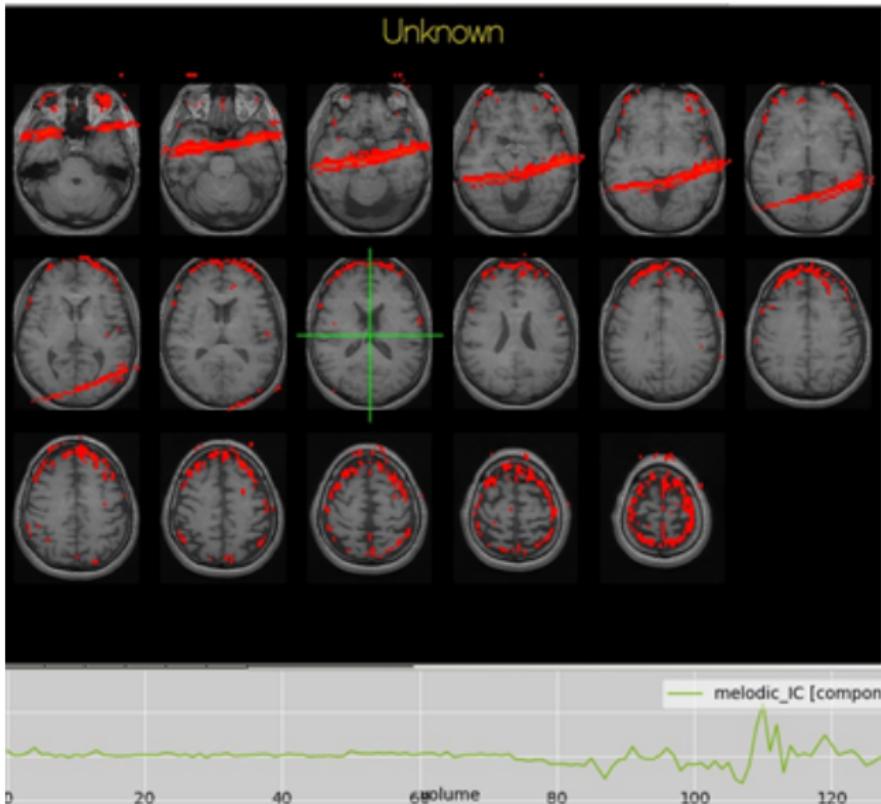
# фMPT: Default Mode Network



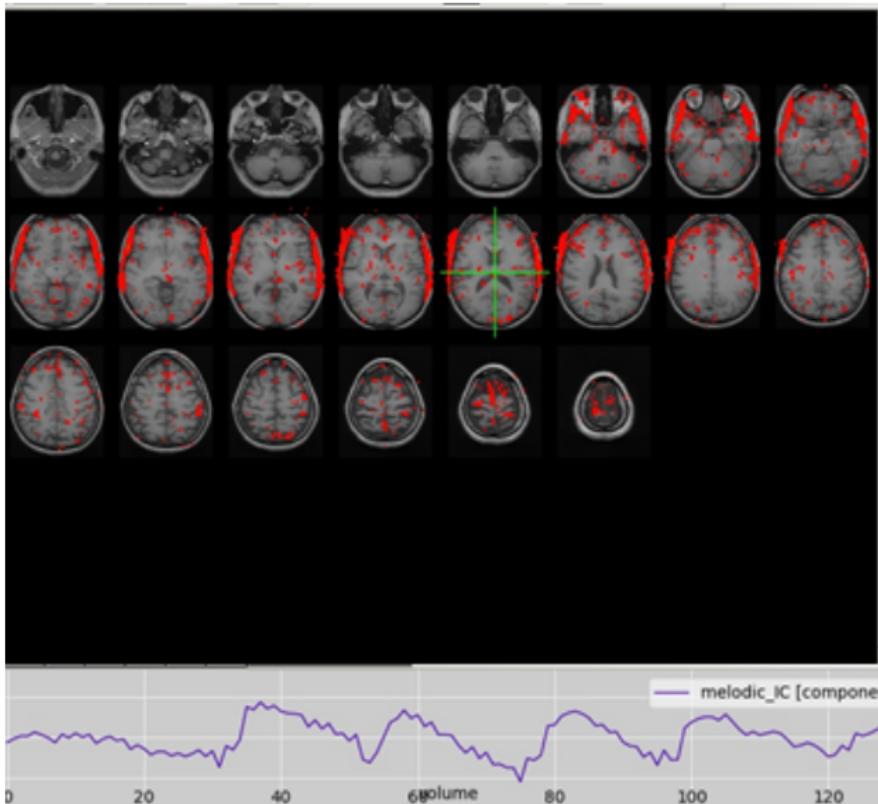
# ϕMPT: Visual Network



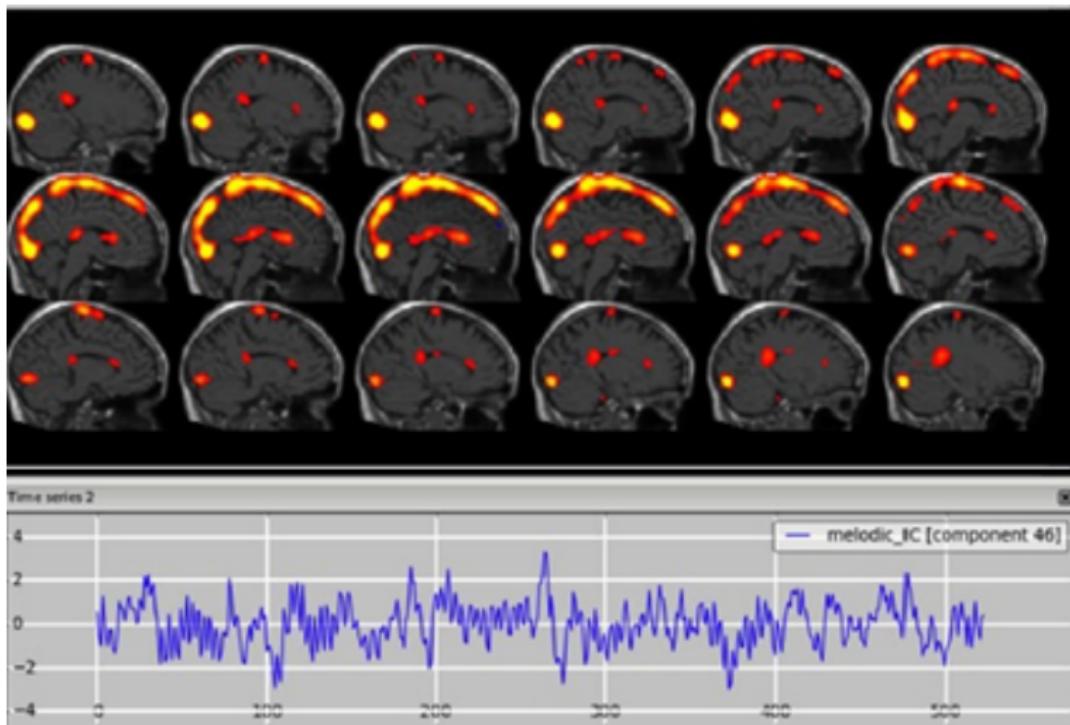
# ϕMPT: Noise (aquisition)



# фМРТ: Noise (motion)



# ϕMPT: Noise (veins)



# Общий вывод из ICA

- › PCA предполагает "нормальный" мир
  - › Для многомерного гауссова вектора он возвращает независимые компоненты
    - › Гауссовые моменты второго порядка уже равны нулю
- › ICA ослабляет предположение о гауссности и предполагает более сложные распределения
  - › Это больше похоже на реальный мир
  - › Это было большим откровением в машинном обучении

# NNMF: Non-negative matrix factorization

# Неотрицательная матричная факторизация (NMF)

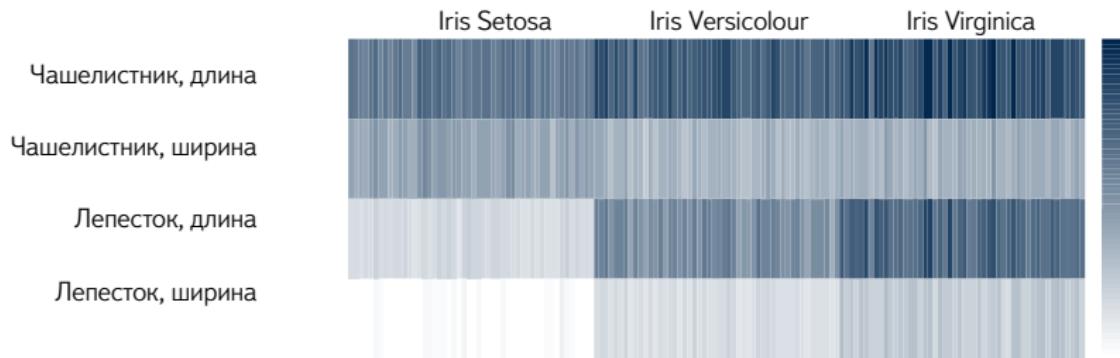
- › Недавний алгоритм (Lee & Seung 1999), близко связанный с компонентным анализом
- › У него есть одно хорошее свойство
  - › Часто он даёт то, что нужно
- › И одно плохое
  - › Часто не понятно, почему

# Неотрицательные данные

- › Мы часто имеем дело с “неотрицательными данными”
  - › Пиксели, энергия, композиции, звук и т. д.
- › Неотрицательные данные требуют особого обращения
  - › Отрицательные признаки могут противоречить действительности

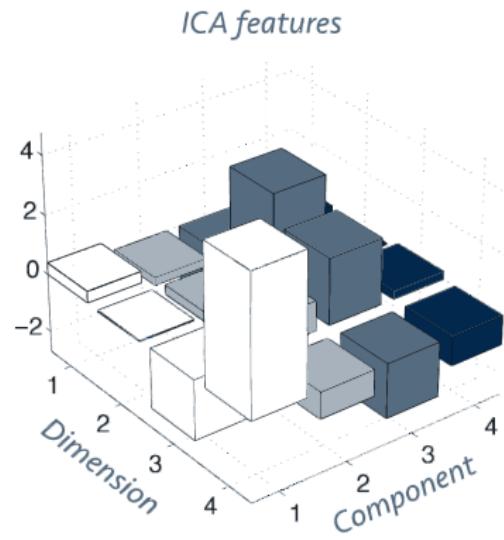
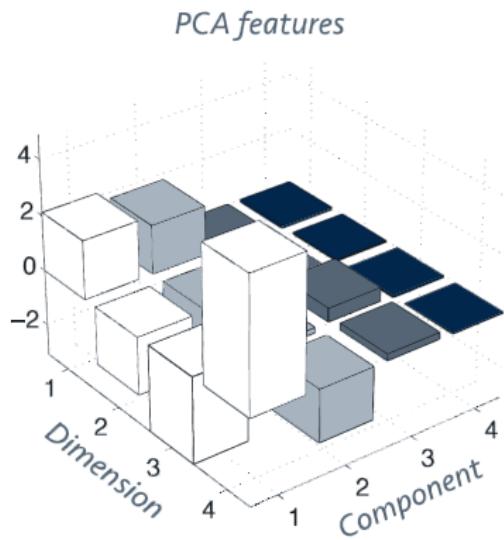
# Пример

- › Iris dataset
  - › Каждый ряд - измерение длины (положительные значения в см)



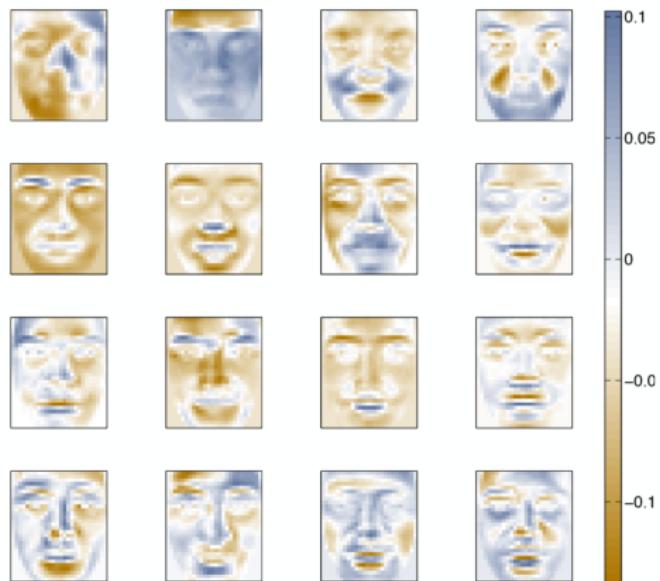
# PCA/ICA анализ на iris dataset

- › Оба дают признаки, которые частично отрицательны
  - › Что это значит?



# То же с eigenfaces

- › “Отрицательные” изображения как базис – почему?



# NNMF: постановка задачи

- › Формальная постановка:

$$X \approx W \cdot H$$

$$X \in \mathbb{R}^{M \times N, \geq 0}, W \in \mathbb{R}^{M \times R, \geq 0}, H \in \mathbb{R}^{R \times N, \geq 0}$$

- › Это схоже с постановкой PCA/ICA
  - ›  $W$  содержит признаки,  $H$  - их "активации"
  - ›  $R$  задает низкую размерность
- › Как это решать?
  - › Одна известная, две неизвестные

# Процесс факторизации

- › Надо оценить два фактора
  - › Чередуем их приближения
- › Пример алгоритма:
  - › Начнем со случайной  $W$
  - › Оцениваем  $H$  с данной  $W$
  - › Оцениваем  $W$  с данной  $H$
  - › повторяем, пока не сойдется

# Нахождение одного фактора при втором фиксированном

- › Эта задача проще
  - › Только одна неизвестная

$$\min_{W \text{ or } H} \sum_{i,j} |X - W \cdot H|^2$$

$$X \in \mathbb{R}^{M \times N, \geq 0}, W \in \mathbb{R}^{M \times R, \geq 0}, H \in \mathbb{R}^{R \times N, \geq 0}$$

- › Налагаем условие неотрицательности
  - › Неотрицательные наименьшие квадраты (медленно)
  - › Оптимизация с ограничениями (медленно)
  - › Применить метод наименьших квадратов и обрезать отрицательные числа (быстро)

# Простой NMF алгоритм

- › Начнем с произвольной  $W$

- › оценим  $H$  с данной  $W$ :

$$H = W^+ \cdot X$$

$$H = \max(H, 0)$$

- › оценим новую  $W$  с данной  $H$ :

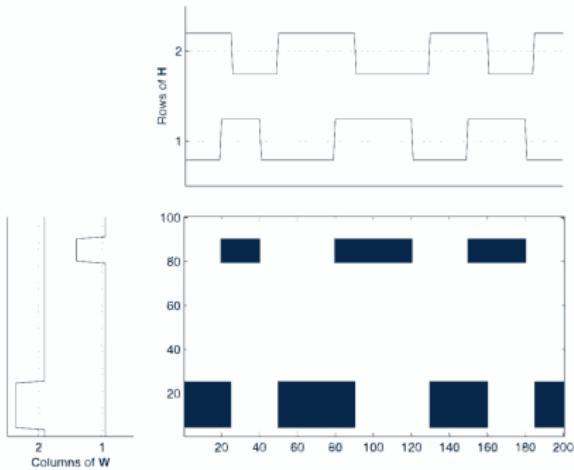
$$W = X \cdot H^+$$

$$W = \max(W, 0)$$

- › повторяем, пока не сойдется

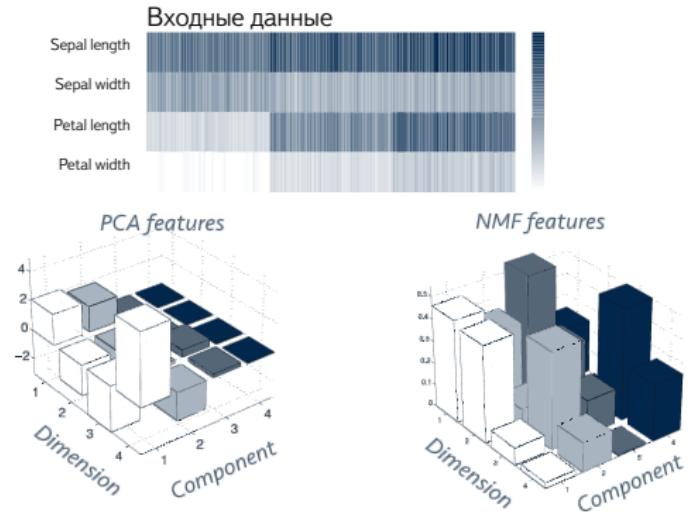
# Пример

- › Входной сигнал  $X$
- › NMF разложит как  
$$X \approx W \cdot H$$
- › Столбцы  $W$  будут  
содержать “вертикальную”  
информацию про  $X$
- › Строки  $H$  будут содержать  
“горизонтальную”  
информацию про  $X$



# Вернемся к ирисам

- › NMF:
  - интерпретируемые результаты
    - › Видно структуру данных
    - › Признаки имеют смысл как относительные величины
  - › PCA/ICA признаки
    - › Не так полезны

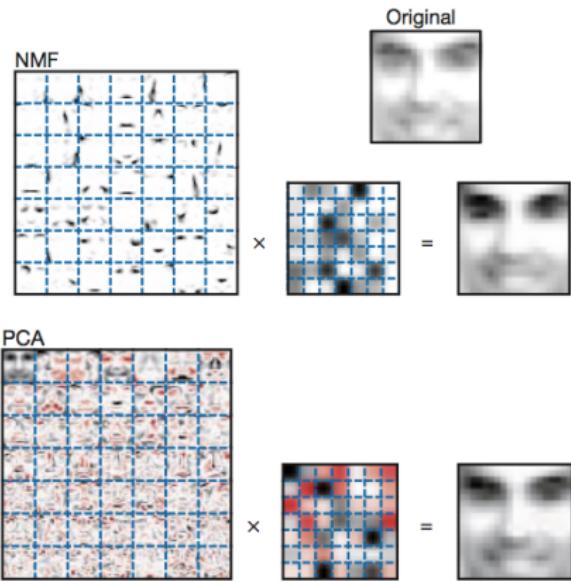


# Декомпозиция по частям

- › NMF производит “аддитивные декомпозиции”
  - › Объясняет данные в терминах добавленных признаков
- › Это соотносится с тем, как мы думаем
  - › Сцены создаются из объектов
    - › У нас никогда нет “отрицательного” присутствия объекта

# Пример с лицами

- › И PCA, и NMF хорошо описывают данные
  - › При этом eigenfaces не интерпретируемы (слишком абстрактно)
  - › NMF находит аддитивные части (носы, глаза и т. п.)
- › NMF более хороший способ объяснять структурированные данные



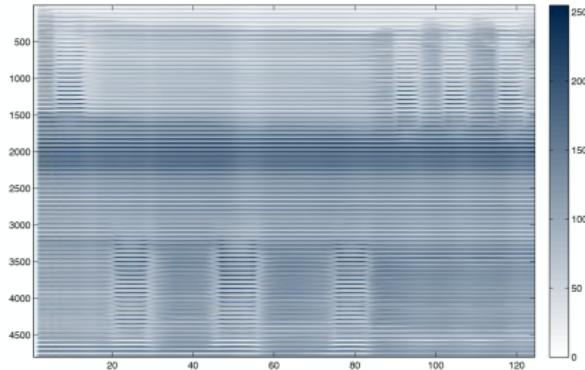
# Компонентный анализ видео

- › Видео - интересные данные для компонентного анализа
  - › Очень высокая размерность
    - › Нужно компактное представление
    - › PCA/NMF могут помочь
  - › Отдельные сцены состоят из элементов
    - › Полезно выделить эти элементы для последующего анализа
    - › PCA/NMF могут помочь
  - › Есть видео- и аудио- данные
    - › У обоих есть своя структура, часто они связаны
    - › Все методы анализа компонент/снижения размерности могут быть полезны

# Примеры

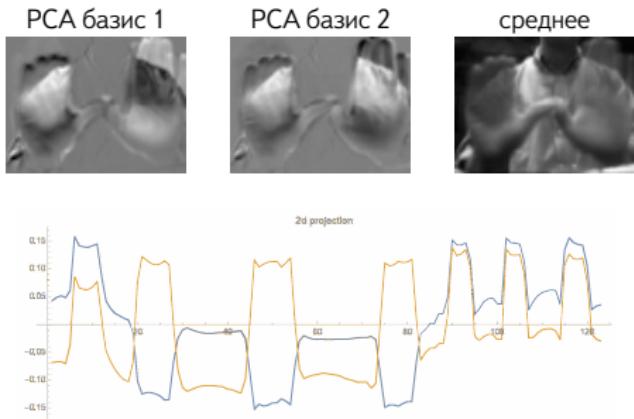
# Пример видео

- › Видео - серия кадров
  - › Каждый кадр - набор данных
  - ›  $126, 80 \times 60$  px кадров
  - › Итого  $4800 \times 126$  сэмплов
- › Попробуем разные подходы
  - › PCA, ICA, NMF
  - › Сравним признаки и веса



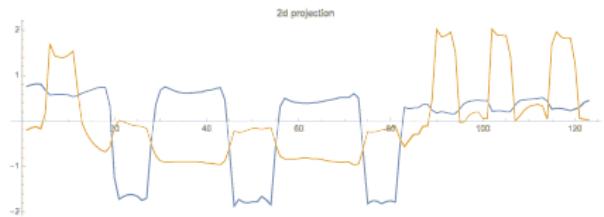
# Результаты РСА

- › Ничего особенного с визуальными компонентами
- › Это ортогональные изображения
  - › Это что-нибудь значит?
  - › Некоторая сегментация между статичными и подвижными частями
- › Некоторое представление движения в весах



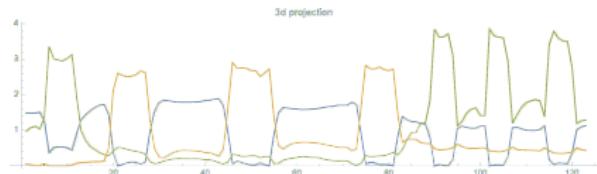
# Результаты ICA

- › Немного лучше
- › Больше независимости
  - › Два действия разделены
  - › Декомпозиция по частям
- › Веса компонент грубо описывают действия



# Результаты NMF

- › Кое-что интерпретируемо
  - › Описывают сумму возможных состояний видео
  - › Возможно, представление семантически более значимо
- › Некоторая избыточность
  - › Мы используем слишком много измерений



# Если использовать меньше измерений

- › NMF дает два состояния видео!

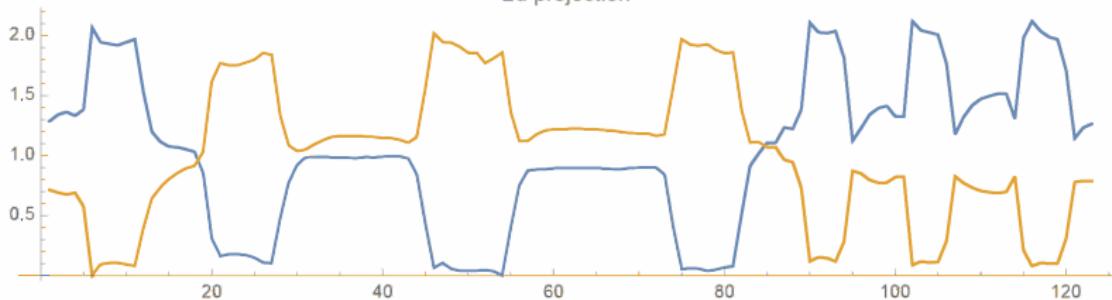
NMF базис 1



NMF базис 2



2d projection



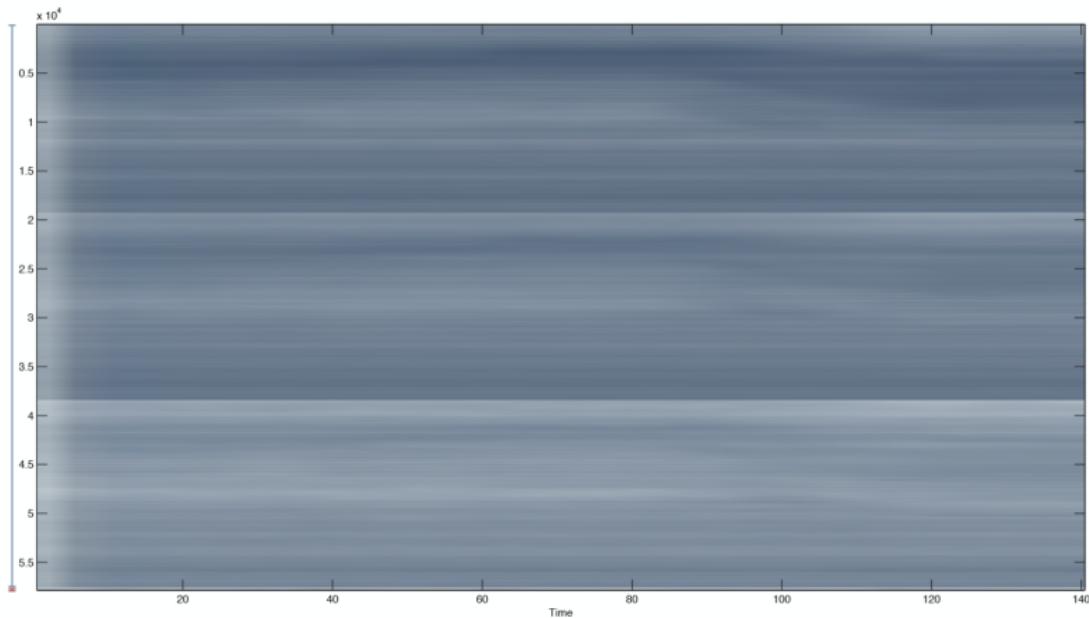
# Аудио/Визуальные компоненты

- › Мы можем одновременно обрабатывать оба вида данных
- › Сможем ли найти зависимость между аудио и видео?
- › Что такое аудио/видео признак?

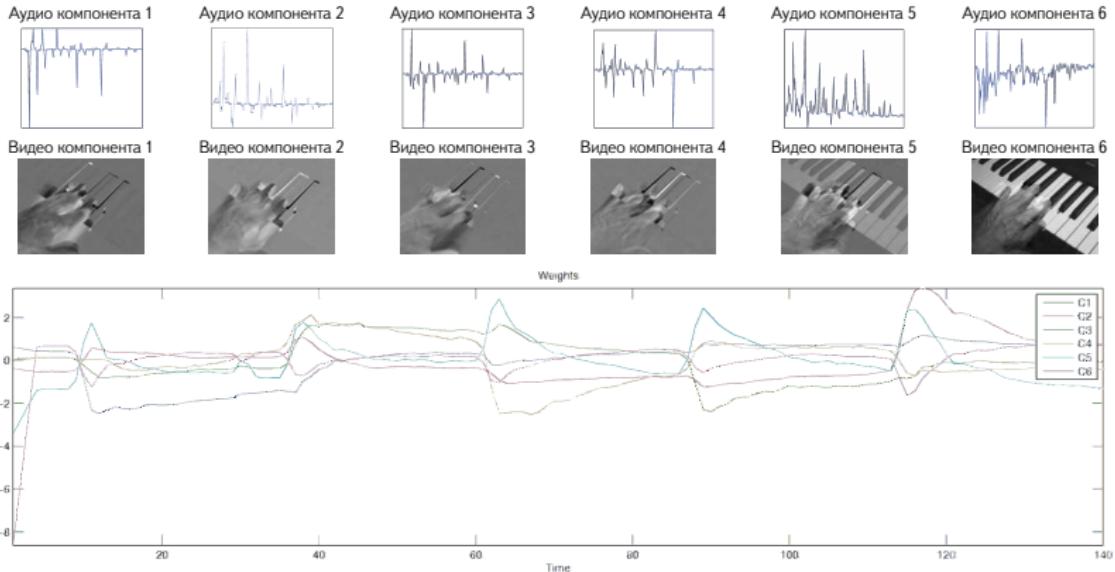


# Как выглядят данные?

57,600 пиксельных измерений, 257 аудио измерений



# Аудио/видео PCA компоненты



# Аудио/видео ICA компоненты

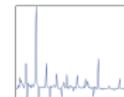
Аудио компонента 1



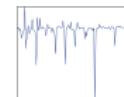
Аудио компонента 2



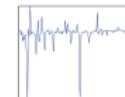
Аудио компонента 3



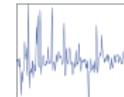
Аудио компонента 4



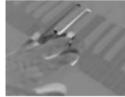
Аудио компонента 5



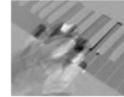
Аудио компонента 6



Видео компонента 1



Видео компонента 2



Видео компонента 3



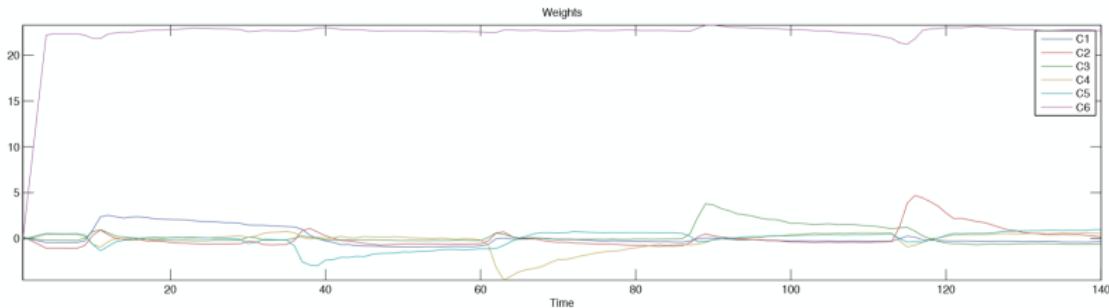
Видео компонента 4



Видео компонента 5

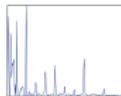


Видео компонента 6



# Аудио/видео NMF компоненты

Аудио компонента 1



Аудио компонента 2



Аудио компонента 3



Аудио компонента 4



Аудио компонента 5



Аудио компонента 6



Видео компонента 1



Видео компонента 2



Видео компонента 3



Видео компонента 4



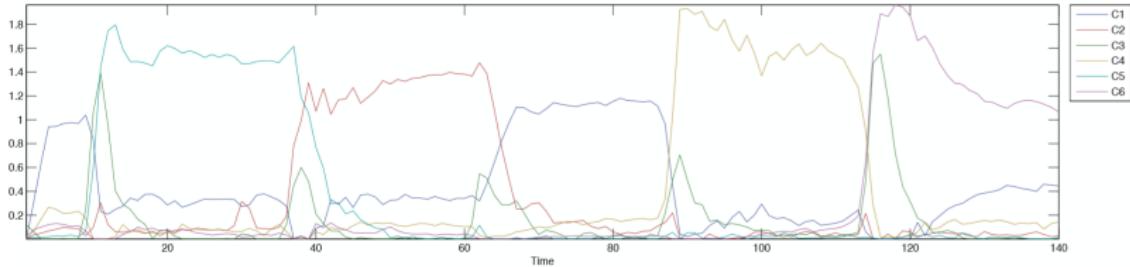
Видео компонента 5



Видео компонента 6



Weights



## Аудио/видео NMF компоненты

## Аудио компонента 1



## Видео компонента 1



## Аудио компонента 2



## Видео компонента 2



### Аудио компонента З



### Видео компонента 3



Аудио компонента 4



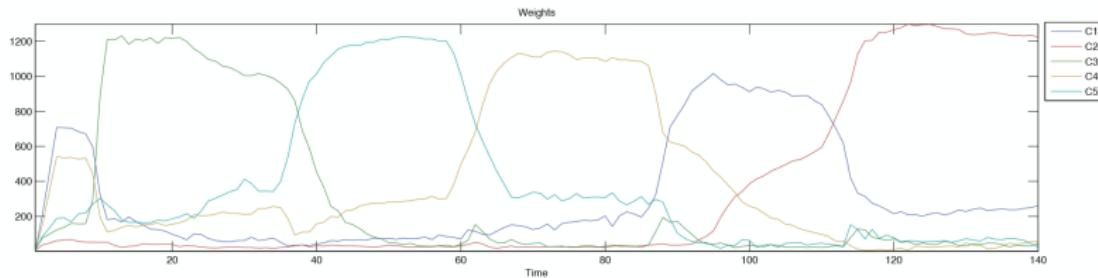
## Видео компонента 4



## Аудио компонента 5



## Видео компонента 5



# Заключение

# PCA, ICA или NMF?

- › Зависит от того, что вы хотите
  - › PCA отлично работает для понижения размерности
  - › ICA дает более разреженный результат
    - › И проще для понимания
  - › NMF дает более интерпретируемый результат
    - › Но только для неотрицательных данных
- › Как обычно, нет правильного ответа
  - › Когда в сомнении, стоит попробовать их все

# Итог

- › Independent Component Analysis
  - › Дает наибольшую независимость
  - › Не понижает размерность
- › Non-Negative Matrix Factorization
  - › Хорошо для анализа неотрицательных данных
    - › пиксели, энергий, счет и т.п....
  - › Но никаких особых статистических свойств

Спасибо за внимание!