



Списки содержания доступны на ScienceDirect

Распознавание образов

домашняя страница журнала: www.elsevier.com/locate/patcogНезависимый компонентный анализ с помощью оптимизации l_p -норм

Парк Сонхон, Ноджун Квак *

Высшая школа конвергенции науки и технологий, Сеульский национальный университет, 1 Gwanak-ro, Gwanak-gu, Сеул 08826, Республика Корея

articleinfo

История статьи:

Получено 18 октября 2016 г.

Исправлено 19 сентября 2017 г.

Принято 7 октября 2017 г. Доступно

онлайн 13 октября 2017 г.

Ключевые слова:

ICA

PCA

 l_p -Норма

Оценка максимального

правдоподобия Супергауссова

Субгауссовский

Аннотация

В этой статье предлагается несколько новых алгоритмов для независимого компонентного анализа (ICA). В предлагаемых методах предполагается, что независимые источники следуют заранее заданному распределению вида $f(s)$ знак равно $\alpha \exp(-\beta |s|^p)$ и оценка максимального правдоподобия используется для разделения источников. В первом методе для оценки максимального правдоподобия используется метод градиентного подъема, а во втором предложен неитерационный алгоритм, основанный на релаксации проблемы. Максимизация логарифма вероятности предполагаемого источника I_{KSTH} учитывая параметр p и данные I_{KSTH} показано, что эквивалентно минимизации l_p -норма прогнозируемых данных I_{KSTH} . Эта формулировка ICA очень тесно связана с L_p -PCA, где решается задача максимизации той же целевой функции. Предлагаемый алгоритм решает приближение l_p -норма минимизации как для супер- ($p < 2$) и субгауссовский ($p > 2$) случаев и демонстрирует более высокую производительность при разделении независимых источников, чем современные алгоритмы вычисления ICA.

© 2017 Elsevier Ltd. Все права защищены.

1. Введение

Поиск значимых прогнозов наблюдаемых данных является важной задачей при многомерном анализе данных. Для простоты его часто представляют как линейное преобразование наблюдений. Если обозначить исходное наблюдение и его преобразование как I и U соответственно, то задача состоит в том, чтобы найти матрицу проекции W так что U знак равно WX является значимым представлением I в некоторых аспектах. Для этой цели было предложено несколько алгоритмов, среди которых анализ главных компонент (PCA) [1] и независимый компонентный анализ (ICA) [2,3] являются наиболее широко используемыми методами для неконтролируемого поиска проекций.

Цель ICA - разложить наблюдаемый многомерный сигнал на набор независимых компонентов (источников). [2,3]. Этот метод изначально был разработан для слепого разделения источников и широко используется в области обработки сигналов и машинного обучения, таких как разделение источников звука [4,5], обработка данных электроэнцефалографии [6–8], и извлечение функций из данных изображения [9–11]. В ICA предполагается, что источники не являются гауссовыми, а наблюдаемые данные построены путем линейных комбинаций источников. Для ICA были предложены различные алгоритмы. [4,12–18], и их расширения или обобщения также предлагаются в литературе. [19–21]. В большинстве алгоритмов матрица несмешивания (проекция) W выбран, чтобы максимизировать негауссовость прогнозируемых данных

U , среди которых FastICA [12,13] были самым успешным алгоритмом более десяти лет.

С другой стороны, классический анализ главных компонент (PCA), который далее будет называться L_2 -PCA, пытается максимизировать дисперсию прогнозируемых данных. U [1]. Другими словами, он находит векторы проекции, которые максимизируют l_2 -норма U . В L_2 -PCA векторы проекции можно легко вычислить с помощью разложения по собственным значениям. Однако L_2 -PCA подвержен выбросам, как и другие l_2 -схемы оптимизации, основанные на нормах. Чтобы решить эту проблему, алгоритмы PCA на основе l_1 -нормы в проектируемом пространстве, устойчивые к выбросам, были предложены [22,23]. Недавно в [24], l_p -нормальное расширение [23] также был предложен для произвольного положительного числа p .

Несмотря на сходство концепций PCA и ICA, связь между ними практически не исследована. Интересно, что недавняя статья Мартина-Клемента и Зарзосо [25] показали, что ICA можно рассматривать как применение PCA- L_1 к побеленным данным. Вдохновленные этим, в этой статье предлагается несколько новых методов для ICA, использующих технику оптимизации L_p -нормы.

Хотя FastICA показывает удовлетворительный результат, производительность сильно зависит от выбора функции контрастности. Используя связь ICA с l_p -нормой оптимизация, таким образом, с L_p -PCA [24], формулировка ICA становится простой и интуитивно понятной. Хотя вычислительная сложность предложенного алгоритма

* Корреспондент.

Адрес электронной почты: suncheonpark@snu.ac.kr (Искра), nojunk@snu.ac.kr (Н. Квак).

¹ В этой статье L_1 -PCA относится к методу, который минимизирует l_1 ошибку реконструкции во входном пространстве, в то время как PCA- L_1 обозначает ту, которая максимизирует l_1 -норма в пространстве функций.

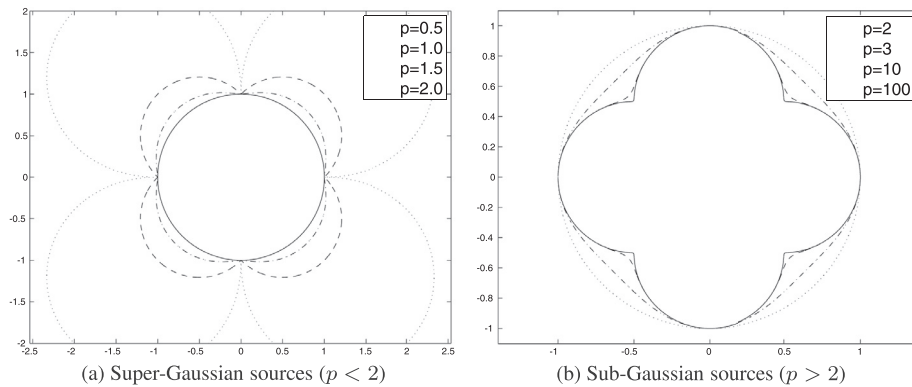


Рисунок 1. L_p -норма ($\|y\|_p$) точек на единичной окружности ($\|y\|_2 = 1$) для разных p ценности. Расстояние точки от начала координат - это L_p -норма единичного вектора с тем же направлением. Зар < 2 направление минимальной нормы совмещено с осью, что соответствует наиболее разреженному направлению. Зар > 2 , это соответствует $\{\pm 1\}^2$, самый плотный y направление.

является $O(p^2)$ где p - количество выборок, это быстро, потому что это неитеративный, детерминированный алгоритм. Результаты моделирования показывают, что эффективность восстановления предложенного алгоритма также превосходит современные алгоритмы вычисления ICA (FastICA и RobustICA[18]) особенно для супергауссовских источников. Одним из недостатков FastICA является то, что для схождения требуется много итераций, или иногда он не может сойтись, когда количество выборок довольно мало. С другой стороны, предлагаемый метод быстро сходится даже для небольшого количества выборок.

2. Краткий обзор ICA.

Предположим, что d -мерный случайный вектор с нулевым средним и единичной дисперсией s знак равно s_1, \dots, s_d обозначает набор d независимых источников и матрица S знак равно $s_1, \dots, s_d \in \mathbb{R}^{d \times n}$ состоит из p образцы s .² Матрица наблюдения $Икс$ предполагается, что это линейная комбинация исходных данных через $d \times d$ матрица смешивания A_0 , т.е. $Икс$ знак равно $A_0 S$.

В большинстве алгоритмов ICA для получения неизвестных источников S из наблюдения $Икс$, данные $Икс$ сначала трансформируются в сферическую (или беленую) версию $Икс$ таким образом, чтобы его выборочная матрица ковариаций была приходил единичная матрица, т.е. $\frac{1}{n} Икс Икс^T$ знак равно I_d . Эти сферические данные могут быть полученный масштабированной версией L2-PCA как $Икс$ знак равно $\frac{1}{\sqrt{2}} U^T Икс$ где U - пары собственных значений и собственных векторов выборочной ковариационной матрицы $\frac{1}{n} Икс Икс^T$.

Теперь целью ICA становится поиск матрицы несмешивания. W такой, что

$$W Икс \text{ знак равно } I \quad (1)$$

где $p \in \mathbb{R}^{d \times d}$ матрица перестановок и I_0 - W тот W ограничивается ортогональной матрицей, поскольку

$$W W^T \text{ знак равно } M \text{ где } M_{ij} = \frac{1}{n} Икс Икс^T \text{ знак равно } E[SS^T] \text{ знак равно } P E[SS^T] \text{ знак равно } I_d \quad (2)$$

Здесь, $E(\cdot)$ - операция ожидания, которую можно заменить выборочным средним. В(2), первое равенство связано с процессом отбеливания, а последнее равенство связано с предположением, что источники независимы с нулевым средним и единичной дисперсией.

Предполагается, что источники для задачи ICA не являются гауссовскими. Экссесс и отрицательная энтропия (негэнтропия) часто используются для измерения негауссовости распределения. Экссесс определяется как кумулянт четвертого порядка. Распределение с положительным эксцессом называется супергауссовым, а распределение с отрицательным эксцессом - субгауссовым. Негэнтропия измеряет разницу энтропии по сравнению с распределением Гаусса. Известно, что это

² В этой статье скаляр, вектор и матрица будут обозначаться строчными буквами (например, $Икс$), строчные буквы, выделенные жирным шрифтом (например, **Икс**) и заглавными буквами (например, $Икс$), соответственно. Одни и те же обозначения будут использоваться как для случайной величины (или случайного вектора), так и для ее выборки, значение которой будет очевидно в контексте.

статистически более устойчиво, но в вычислительном отношении более интенсивно, чем эксцесс, поскольку требуется оценка функции плотности вероятности. Здесь мы кратко опишем два типичных алгоритма ICA, FastICA [12] и RobustICA [18], которые используют соответственно негэнтропию и эксцесс. Эти методы будут использоваться в экспериментах для сравнения производительности с методом, предложенным в этой статье.

FastICA Hyvärinen [12] предложил функцию контраста на одну единицу J_f что приблизительно соответствует негэнтропии:

$$J_f(w) \text{ знак равно } (E[g(w^T Икс)] - E[g(v)])^2 \quad (3)$$

где g неквадратичная функция, и v представляет гауссову случайную величину с нулевым средним и единичной дисперсией. Выбор функции g важен как для устойчивости, так и для сходимости алгоритма FastICA [12]. Типичный выбор g либо $G(y) \text{ знак равно } \frac{1}{2} \cosh(ay)$ с $1 \leq a \leq 2$ или $G(y) = -\exp(-y^2/2)$. Алгоритм с фиксированной точкой, который повторяет следующие два шага до тех пор, пока сходимость не используется для максимизации(3):

$$w \leftarrow E[Икс g(w^T Икс)] - E[g'(w^T Икс)] w, \quad w \leftarrow \frac{w}{\|w\|} \quad (4)$$

Здесь, g и g' - первая и вторая производные от g , соответственно.

RobustICA Недавно Зарзосо и Комон [18] предложил RobustICA, который использует эксцесс в качестве функции контраста. Он максимизирует абсолютное значение следующего эксцесса:

$$K(w) \text{ знак равно } \frac{E|w^T Икс|^4 - 2E|w^T Икс|^2 - |E(w^T Икс)|^2}{E|w^T Икс|^2} \quad (5)$$

Метод выполняет точный линейный поиск по функции контраста. Оптимальный размер шага, который максимизирует(5) находится путем решения корней многочлена четвертой степени. Авторы утверждали, что RobustICA имеет меньшие вычислительные затраты и более быструю сходимость по сравнению с FastICA. RobustICA также может применяться к комплексным сигналам.

Помимо RobustICA и FastICA, в литературе предлагались различные алгоритмы ICA. Например, Ли и др.[14] расширенный алгоритм инфомакс для ICA. Theis et al.[26] разработан линейно-геометрический ИКА на основе гистограмм.

Ни один из вышеперечисленных алгоритмов не использует связь между ICA и L_p -нормальная оптимизация. Новый алгоритм ICA на основе L_p -нормальная оптимизация предлагается в данной статье. ВРаздел 3, мы объясним, как можно преобразовать ICA в L_p -нормальная задача оптимизации. Затем фактический алгоритм для медленной и быстрой версии предлагается вРазделы 4 и 5 соответственно.

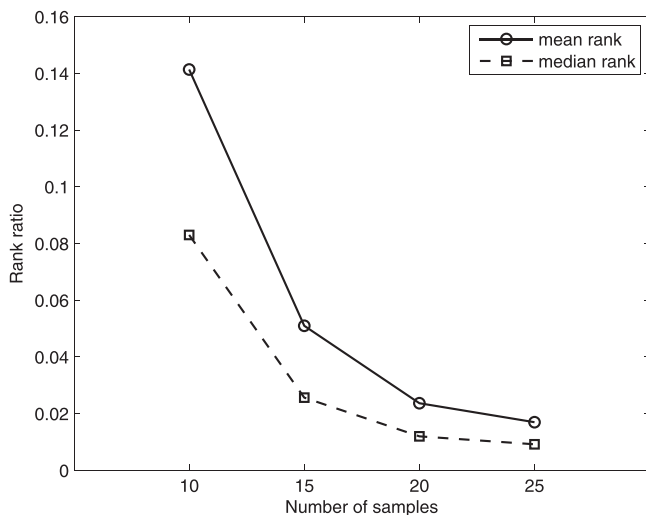


Рис. 2. Среднее и медианное отношения рангов жадного алгоритма Lp-ICA-F-. По мере увеличения количества выборок жадное решение приближается к глобальным оптимумам.

3. ICA Автор l_p -норма минимизация

3.1. Постановка проблемы

Наше ключевое наблюдение заключается в том, что, как и в [27–29] оценка максимального правдоподобия (MLE) может использоваться напрямую для разделения источников, а не для измерения эксцесса или негэнтропии. В нашем подходе, как и в [17], мы рассматриваем следующее обобщенное гауссово распределение с нулевым средним и единичной дисперсией как истинное распределение источников:

$$P(s) \propto \exp(-\beta |s|^\alpha). \quad (6)$$

Здесь, $\alpha > 0$ - константа, которая делает $P(s)$ распределение вероятностей, и $\beta > 0$ - константа, определяющая ширину распределения. Мы выбрали обобщенное гауссово распределение, потому что это параметрическое семейство распределений, которое охватывает широкий спектр типичных унимодальных и симметричных распределений реального мира. Учитывая, что предполагается, что источники имеют единичную дисперсию, α и β полностью зависят от p и может быть вычислено как $\alpha(p) = \frac{\beta p}{2^{1/p}}$ и $\beta(p) = \frac{\Gamma(p)}{\Gamma(p/2)^2}$, где $\Gamma(\cdot)$ это гамма-функция [17].

Интересно то, что, изменив p , распределение может быть супергауссовым или субгауссовым. Более конкретно, он становится супергауссовым, если $p < 2$, субгауссовым, если $p > 2$, и гауссовым, если p равно 2. Хювяринен [12] и Колдовский [17] Также упоминалось, что эта функция может использоваться как функция контраста, но алгоритм FastICA фактически использует приближение этого вида функции в качестве функции контраста. Предлагаемая в этом разделе схема не требует измерения эксцесса или негэнтропии.

Предположим источник s распространяется как (6) и побеленные данные $Икс$ равно $Икс_1, \dots, Икс_n$ даны. В ИЦА нам предстоит реконструировать от u знак равно $ш$ $Икс$ используя соответствующий вектор разложения $ш$ с ограничением $\|ш\|_2 = 1$. Тогда вероятность $ш$ и p данный $Икс$ является

$$L(ш, p | Икс) \propto \prod_{j=1}^n \exp(-\beta |ш_j Икс_j|^\alpha) \quad (7)$$

где $ш$ равно $Икс_1, \dots, Икс_n$ состоит из p независимых и одинаково распределенные (iid) выборки. Журнал (7), логарифмическая вероятность становится

$$L(ш, p | Икс) \propto \prod_{j=1}^n |ш_j Икс_j|^{-\beta/p} \quad (8)$$

здесь, $Икс \in \mathbb{R}^p$ - случайный вектор, p реализации $Икс$ равно $Икс_1, \dots, Икс_n$.

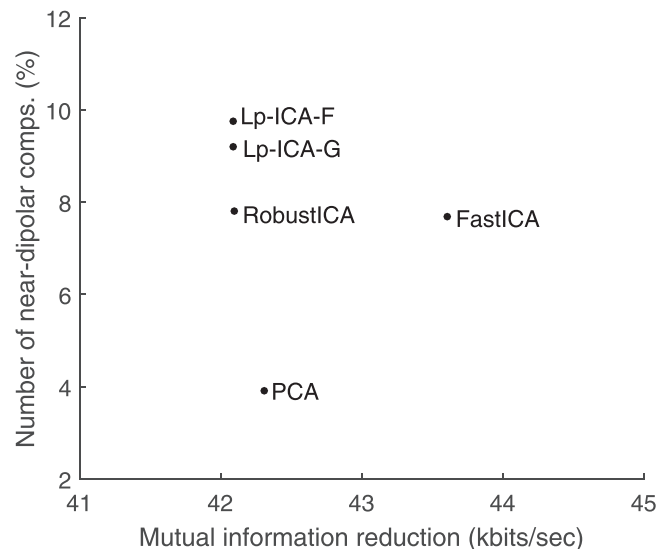


Рис. 3. Сравнение производительности алгоритмов ICA на наборе данных ЭЭГ. Взаимная редукция информации и доля околодипольных составляющих после разложения. Более высокие значения подразумевают большую независимость между разложенными источниками.

За фиксированный p , задача разделения источников сводится к нахождению вектора $ш^*$ который максимизирует вышеупомянутую логарифмическую вероятность, которую можно упростить как

$$\underset{ш}{\arg \max} \sum_{j=1}^n |ш_j Икс_j|^\alpha \quad \text{ш знак равно } 1, \quad (9)$$

так как α и β являются константами, если p фиксированный. Обратите внимание, что (9) является формой минимизации нормы для произвольных p знак равно 2, тогда как максимизация (9) сводится к проблеме Lp-PCA, которая была решена в [24].

3.2. Обсуждение

Учитывая $ш$ знак равно 1 и $Икс$ побелен, то $\|ш\|_2$ знак равно $ш$ знак равно $ш$ $Икс$ знак равно p . В настоящее время, (9) можно преобразовать в оптимизацию проблема в отношении $ш$ следующее:

$$\underset{ш}{\arg \max} \|ш\|_p \quad \text{ш знак равно } 1, \quad (10)$$

Если мы ослабим проблему, удалив второе ограничение в (10), это становится

$$\underset{ш}{\arg \max} \|ш\|_p \quad \text{ш знак равно } p. \quad (11)$$

Вышеупомянутая проблема является l_p минимизация нормы на p -размерная гиперсфера. На каждые $0 < p < 2$ (супергауссовский источников), оптимальное решение $ш^*$ этой расслабленной проблемы ось выровнен, т. е. $ш^* = \pm \frac{1}{\sqrt{p}} (1, \dots, 1)$ где e_j - единичный вектор на j -я координатная ось. С другой стороны, для каждого $p > 2$ (субгауссовский), он становится $ш^* = \pm 1$, т. е. каждый элемент $ш^*$ равно 1 или -1. Это показано в рисунке 1, для простого случая p равно 2.

Поскольку все супергауссовские источники имеют точно такое же решение с минимальной нормой (11), фиксированное значение p (например, p равно 1) можно использовать для поиска супергауссовских источников. Аналогичным образом другое фиксированное значение (например, p равно 3) также можно использовать для субгауссовских источников.

Решение (11) имеет очень тесную связь с оптимизацией эксцесса. Так как можно интерпретировать как p образцы

4 В такой постановке мы не можем гарантировать единственность оптимального решения и равенство в (9) следует заменить на \leq . Однако для простоты на протяжении всей статьи мы используем знак равенства.

5 Это можно доказать с помощью метода лагранжевых множителей.

Таблица 1

ICA, автор ℓ_p -нормальная оптимизация.

	минимизация	максимизация
$p < 2$	супер-G. (видеть Раздел 4 и 5) $\sum_{j=1}^n \mathbf{w}_j^T \mathbf{I}_{\text{Икс}} $ <small>например \mathbf{w}-знак равно аргумент мин $\mathbf{w} \cdot \mathbf{w} / \ \mathbf{w}\ = 1$ $j=1$</small>	к югу от G. (PCA-L1 [23] или Lp-PCA [24]) $\sum_{j=1}^n \mathbf{w}_j^T \mathbf{I}_{\text{Икс}} $ <small>например \mathbf{w}-знак равно арг макс $\mathbf{w} \cdot \mathbf{w} / \ \mathbf{w}\ = 1$ $j=1$</small>
$p > 2$	к югу от G. (видеть Раздел 4 и 5) $\sum_{j=1}^n \mathbf{w}_j^T \mathbf{I}_{\text{Икс}} $ <small>например \mathbf{w}-знак равно аргумент мин $\mathbf{w} \cdot \mathbf{w} / \ \mathbf{w}\ = 1$ $j=1$</small>	супер-G. (Lp-PCA [24]) $\sum_{j=1}^n \mathbf{w}_j^T \mathbf{I}_{\text{Икс}} $ <small>например \mathbf{w}-знак равно арг макс $\mathbf{w} \cdot \mathbf{w} / \ \mathbf{w}\ = 1$ $j=1$</small>

предполагаемый источник \mathbf{y} - s), алгоритм ICA путем оптимизации эксцесса \mathbf{u} можно рассматривать как частный случай (11) когда ℓ знак равно 4. Формально говоря, задача оптимизации эксцесса принимает следующий вид:

$$\mathbf{u}^* = \arg \max_{\mathbf{u}} \sum_{j=1}^n |\mathbf{u}_j| \quad \text{у.т.д.} \quad \|\mathbf{u}\|_2 \text{ знак равно } \ell. \quad (12)$$

Здесь задача максимизации состоит в том, чтобы найти супергауссов источник, а минимизация - для субгауссовских источников. Примечание: то решение, которое максимизирует (12) является $\mathbf{u}^* = \pm \ell \mathbf{e}_j$, в то время как тот, который сводит к минимуму (12) является \mathbf{u}^* знак равно $\{\pm 1\}_n$, которые точно такие же, как решения (11) за $p < 2$ и $p > 2$ соответственно.

Однако обратите внимание, что из-за ограничения \mathbf{u} знак равно $\mathbf{I}_{\text{Икс}} \mathbf{w}$, эти оптимальные точки ($\mathbf{u}^* = \pm \ell \mathbf{e}_j$ или $\{\pm 1\}_n$) не может быть достигнута вообще. Тем не менее, (11) играет решающую роль в разработке быстрого алгоритма ICA в [Раздел 5](#).

Еще одна интересная вещь, на которую следует обратить внимание в этой части, это то, что решение минимальной нормы (11) за $p < 2$ (см. [рисунок 1\(а\)](#)) совпадает с решением максимальной нормы для $p > 2$ (см. [рисунок 1\(б\)](#)) и наоборот. Следовательно, для субгауссовских источников проблема может быть переформулирована как ℓ_p -задача максимизации нормы с $p < 2$ где PCA-L1 [23] или Lp-PCA [24] алгоритмы могут использоваться напрямую. Таким образом, фиксированное значение ℓ может использоваться для разделения как супергауссовых, так и субгауссовых источников. Например, если мы исправим ℓ знак равно 1, Проблема ICA становится

$$\mathbf{w}^* = \arg \min_{\mathbf{w}} \sum_{j=1}^n |\mathbf{w}_j^T \mathbf{I}_{\text{Икс}}| \quad \text{у.т.д.} \quad \|\mathbf{w}\|_2 = 1 \quad (13)$$

для супергауссовских источников, а становится

$$\mathbf{w}^* = \arg \max_{\mathbf{w}} \sum_{j=1}^n |\mathbf{w}_j^T \mathbf{I}_{\text{Икс}}| \quad \text{у.т.д.} \quad \|\mathbf{w}\|_2 = 1 \quad (14)$$

для субгауссовских источников.

Таблица 1 резюмирует этот раздел. Чтобы разделить супергауссовы (субгауссовы) источники, необходимо решить ℓ_p -задача минимизации нормы с $p < 2$ ($p > 2$) или ℓ_p -задача максимизации нормы с $p > 2$ ($p < 2$). Задачу максимизации можно решить с помощью PCA-L1 [23] или Lp-PCA [24]. Решения задач минимизации рассматриваются в [Разделы 4 и 5](#).

4. Lp-ICA-G: алгоритм градиентного подъема.

В [24] градиент целевой функции (9) относительно \mathbf{w} рассчитывается как

$$\nabla_{\mathbf{w}} \text{знак равно } \ell \sum_{j=1}^n \text{подписать}(\mathbf{w}_j^T \mathbf{I}_{\text{Икс}}) / \|\mathbf{w}\|_2 \quad \text{у.т.д.} \quad \|\mathbf{w}\|_2 = 1 \quad (15)$$

где знак (\cdot) - это функция, которая выводит +1 или -1 в зависимости от знака ввода и выводит 0, когда ввод равен 0. Умножение $-\beta(\ell)$ к этому дает градиент $\nabla \mathbf{w}$ вероятности журнала (8) относительно \mathbf{w} . Когда $\ell > 1$ градиент (15) корректно определено даже для особых точек, где $\mathbf{w}_j^T \mathbf{I}_{\text{Икс}} = 0$ для некоторых $\{ \mathbf{I}_{\text{Икс}} \}_j$. С другой стороны, если $0 < \ell < 1$, градиент для особых точек не определен. Однако в этом случае можно рассматривать субдифференциальную $\partial \mathbf{w}$ функции $f(\mathbf{w})$ знак равно $|\mathbf{w}_j^T \mathbf{I}_{\text{Икс}}|$ и это очевидно

что он содержит ноль. Следовательно, точки данных, ортогональные \mathbf{w} можно игнорировать при вычислении (15).

В дополнении к $\nabla \mathbf{w}$, градиент логарифмического правдоподобия (8) относительно ℓ можно получить как

$$\frac{\alpha}{\alpha} - \beta \sum_{j=1}^n |\mathbf{w}_j^T \mathbf{I}_{\text{Икс}}| - \beta \sum_{j=1}^n |\mathbf{w}_j^T \mathbf{I}_{\text{Икс}}| \quad \text{у.т.д.} \quad \|\mathbf{w}\|_2 = 1 \quad (16)$$

где α и β градиенты $\alpha(\ell)$ и $\beta(\ell)$ относительно ℓ , соответственно. На практике метод численного дифференцирования используется для получения α и β так как производная гамма-функции не имеет замкнутой формы.

При поиске нескольких источников k й вектор размешивания \mathbf{w}_k должен быть ортогонален ранее найденным векторам $\mathbf{w}_1, \dots, \mathbf{w}_{k-1}$. Для этого можно использовать ортогонализацию Грама-Шмидта. В наш метод, чтобы уменьшить вычислительную сложность, вместо применяя ортогонализацию Грама-Шмидта к \mathbf{w}_k на каждой итерации процесс ортогонализации применяется к матрице данных $\mathbf{I}_{\text{Икс}}$ только один раз в поиске каждого источника. Один раз d векторы, окончательная матрица несмешивания может быть получена как \mathbf{W} знак равно $[\mathbf{w}_1, \dots, \mathbf{w}_d]$. Общая процедура описана в [Алгоритм 1](#). Здесь 'normal' - вектор

Алгоритм 1 Lp-ICA-G: алгоритм градиентного подъема для ICA от ℓ_p -оптимизация норм.

Требовать: отбеленные данные $\mathbf{I}_{\text{Икс}} (n \times \ell)$ знак равно $\mathbf{I}_{\text{Икс}} 1, \dots, \mathbf{I}_{\text{Икс}} \ell$

Гарантировать: матрица размешивания \mathbf{W} знак равно $[\mathbf{w}_1, \dots, \mathbf{w}_d]$

$\mathbf{W} \leftarrow \mathbf{I}$

за k знак равно 1 к d делать

Инициализировать \mathbf{w}_k как случайный вектор и $\ell \leftarrow 2$ \mathbf{w}

$k \leftarrow \text{нормальный}(\mathbf{w}_k - \mathbf{W} \mathbf{W}^T \mathbf{w}_k)$

$X(k) \leftarrow X(k-1) - \mathbf{W} \mathbf{W}^T X(k-1)$

повторение

$\ell \leftarrow \ell + \eta \nabla_{\ell}$ используя (16)

$\mathbf{w}_k \leftarrow \text{нормальный}(\mathbf{w}_k + \eta \nabla \mathbf{w}_k)$ используя (15)

пока не сойдутся

$\mathbf{W} \leftarrow [\mathbf{W}, \mathbf{w}_k]$

конец для

$\mathbf{W} \leftarrow \mathbf{W}^T$

операция нормализации, т. е. нормальная (\mathbf{a}) знак равно $\frac{\mathbf{a}}{\|\mathbf{a}\|_2}$ Скорость обучения η установлен как константа в нашей реализации.

Обратите внимание: если мы заранее знаем тип источников, мы можем установить ℓ как фиксированное значение и ℓ_p -задача минимизации нормы в (9) можно заняться. В этом случае, как показано на [Таблица 1](#), вместо ℓ_p минимизация нормы, ℓ_p -норма максимизации может быть решена, чтобы найти независимые компоненты. За ℓ_p максимизация нормы, PCA-L1 [23] или Lp-PCA [24], которые в основном представляют собой методы градиентного подъема. Более подробные сведения о взаимосвязи между ICA и PCA можно найти в [25].

5. Lp-ICA-F: неитеративный быстрый алгоритм.

5.1. Разделение супергауссовских источников ($\ell < 2$)

Lp-ICA-G - это простая и интуитивно понятная реализация для ICA.

Однако скорость сходимости этого метода намного ниже, чем у этого метода.

Что FastICA [12] и есть много шансов, что решение является локально оптимальным, что является неотъемлемым свойством методов градиентного спуска.

С этого момента мы сконцентрируемся на поиске решения (10). Предлагаемый в этой части метод тесно связан со стратегией в [26,30].

Рассмотрим супергауссовский ($\ell < 2$) случай первый. Как у нас уже видел, (11) имеет ℓ решения \mathbf{u}^* знак равно $\ell \mathbf{e}_j$, j знак равно $1, \dots, \ell$. Однако в нашей формулировке \mathbf{u} должно быть равно $\mathbf{I}_{\text{Икс}} \mathbf{w}$, где ранг $\mathbf{I}_{\text{Икс}}$

самое большое d . Следовательно, $\sqrt{\frac{1}{2}} \|\mathbf{w} - \mathbf{p}\|_2$ захватывает подпространство в \mathbb{R}^n . Другими словами, решение для $\mathbf{I}(\mathbf{w})$ знак равно \mathbf{p} может не существовать. В этом случае минимум квадратные решения $\mathbf{w}^{(j)}$ во-первых обнаружены, что делают $\mathbf{w}^{(j)}$ знак равно $\mathbf{I}(\mathbf{w})$ быть ближайшей точкой к \mathbf{p} за \mathbf{w} знак равно $1, \dots, p$ следующее:

$$\mathbf{w}^{(j)} = \begin{cases} \mathbf{p} & \text{знак равно аргумент мин } \|\mathbf{I}(\mathbf{w}) - \mathbf{p}\|_2 \\ \mathbf{0} & \text{иначе} \end{cases} \quad (17)$$

Затем окончательное решение \mathbf{w}^* установлен на один из $\mathbf{w}^{(j)}$, который дает минимальное значение целевой функции следующим образом:

$$\mathbf{w}^* = \arg \min_j \|\mathbf{I}(\mathbf{w}^{(j)}) - \mathbf{p}\|_2, \quad (18)$$

Вышеуказанная процедура (17) и (18) может использоваться для получения первого вектора разложения \mathbf{w}_1 . При нахождении нескольких векторов несмешивания k th вектор \mathbf{w}_k должен быть ортогонален ранее найденным векторам \mathbf{w}_{k-1} знак равно $1, \dots, \mathbf{w}_{k-1}$. В случае $k > 1$, (17) заменяется на

$$\mathbf{w}^{(j)} = \begin{cases} \mathbf{p} & \text{знак равно аргумент мин } \|\mathbf{I}(\mathbf{w}) - \mathbf{p}\|_2 \\ \mathbf{0} & \text{иначе} \end{cases} \quad (19)$$

После получения p решения (19), (18) снова применяется для получения k th вектор размешивания.

Оба (17) и (19) можно решить с помощью лагранжевых множителей. Путем введения лагранжевых множителей λ и μ знак равно $\mu_1, \mu_2, \dots, \mu_{k-1}$, (19) становится

$$L(\mathbf{w}, \lambda, \mu) = \frac{1}{2} \|\mathbf{I}(\mathbf{w}) - \mathbf{p}\|_2^2 + \frac{\lambda}{2} (\mathbf{w}^T \mathbf{w} - 1) + \sum_{j=1}^{k-1} \mu_j \mathbf{w}^T \mathbf{w}_j. \quad (20)$$

Установка производной по \mathbf{w} к нулю как

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{I}(\mathbf{w}) - \mathbf{p} + \lambda \mathbf{w} + \sum_{j=1}^{k-1} \mu_j \mathbf{w}_j = \mathbf{0}, \quad (21)$$

и переставляя термины, становится

$$\mathbf{w} = \frac{1}{\lambda} (\mathbf{I}(\mathbf{w}) - \mathbf{p} + \sum_{j=1}^{k-1} \mu_j \mathbf{w}_j). \quad (22)$$

где идентичности $\mathbf{w}^T \mathbf{w} = 1$ и $\mathbf{I}(\mathbf{w})^T \mathbf{w} = 1$ используются. Левое умножение \mathbf{w}^T с обеих сторон дает

$$\mu_j = \frac{1}{\mathbf{w}^T \mathbf{w}_j} (\mathbf{I}(\mathbf{w})^T \mathbf{w}_j - \mathbf{p}^T \mathbf{w}_j). \quad (23)$$

Подставляя (23) к (22) дает

$$\mathbf{w} = \frac{1}{\lambda} (\mathbf{I}(\mathbf{w}) - \mathbf{p} + \sum_{j=1}^{k-1} \frac{1}{\mathbf{w}^T \mathbf{w}_j} (\mathbf{I}(\mathbf{w})^T \mathbf{w}_j - \mathbf{p}^T \mathbf{w}_j) \mathbf{w}_j). \quad (24)$$

мы позволим $\mathbf{I}(\mathbf{w}) = \mathbf{Y}^T \mathbf{d}$ и $\mathbf{w} = \mathbf{W} \mathbf{z}$, тогда \mathbf{z} можно рассчитать как

$$\mathbf{z} = \frac{1}{\lambda} (\mathbf{Y}^T \mathbf{d} - \mathbf{W} \mathbf{z} + \sum_{j=1}^{k-1} \frac{1}{\mathbf{z}^T \mathbf{z}_j} (\mathbf{Y}^T \mathbf{d}^T \mathbf{z}_j - \mathbf{p}^T \mathbf{z}_j) \mathbf{z}_j). \quad (25)$$

Решение (17) может быть получено аналогичным образом, т.е. \mathbf{z} знак равно нормальный($\mathbf{I}(\mathbf{w})$). Поскольку мы не знаем, какой \mathbf{z} что ближе всего к \mathbf{p} (\mathbf{z} знак равно $1, \dots, p$) минимизирует функцию затрат, всего p нужны испытания. Решение гарантирует глобальный минимум для функции стоимости (17) и (19). Хотя это не гарантирует оптимальности (9), ожидается, что оно будет очень близко к глобальному оптимальному решению, поскольку приближение (17) близко к (9). Кроме того, поскольку он проверяет все p возможных вершин предлагаемый алгоритм не попадает в локальное оптимальное решение. Предлагаемый алгоритм Lp-ICA-F + описан в Алгоритм 2. Первая строка внешнего цикла

шаг ортогонализации, чтобы убедиться, что $\mathbf{w}^{(j)}$ ортогонален ранее найденному набору несмешивающих векторов.

Вычислительная сложность Lp-ICA-F составляет $O(dn^2)$. Обычно количество источников намного меньше количества отсчетов, т.е. $d \ll n$ а сложность квадратична по n . Однако предлагаемый алгоритм не является итеративным, в отличие от большинства других алгоритмов ICA. Следовательно, предлагаемый алгоритм может быть быстрым для небольших наборов данных.

Алгоритм 2 Lp-ICA-F + (super): быстрый алгоритм ICA для супергауссовских источников.

Требовать: отбеленные данные $\mathbf{I}(\mathbf{w})$ знак равно $\mathbf{I}(\mathbf{w})_1, \dots, \mathbf{I}(\mathbf{w})_p$, $p < 2$

Гарантировать: матрица размешивания \mathbf{W} знак равно $\mathbf{w}_1, \dots, \mathbf{w}_p$

```

W ← []
за k знак равно 1 к d делать
    X(k) ← X(k-1) - WW^T X(k-1)
за я знак равно 1 к p делать
    w(j) ← нормальный(I(w)_j), c(j) ← ||I(w)_j||_2
конец для
я ← аргумент мин c(j), W ← [W, w(j)]
я
конец для
W ← W^T
  
```

5.2. Разделение субгауссовских источников ($p > 2$)

Если тот же алгоритм, что и в Раздел 5.1 применяется к суб-Гауссовский случай, \mathbf{p} в (17) и (19) необходимо заменить на $\{\pm 1\}^p$, и тот, который минимизирует целевую функцию $\|\mathbf{I}(\mathbf{w}) - \mathbf{p}\|_2$ можно выбрать в качестве решения. Однако в этом случае временная сложность будет экспоненциальной, так как мы должны проверить 2^p комбинации ± 1 для возможного решения. Чтобы справиться с трудностями, присущими комбинаторной оптимизации, для быстрого субгауссовского разделения источников применяется жадный подход, который ищет локальный минимум. Существует множество алгоритмов, основанных на жадных подходах, и имеется достаточно доказательств того, что жадный подход практически находит приемлемое решение, даже если оно не самое лучшее. [31].

В Алгоритм 3, показан предлагаемый жадный алгоритм. В

Алгоритм 3 Lp-ICA-F- (sub): быстрый алгоритм ICA для субгауссовского источников.

Требовать: отбеленные данные $\mathbf{I}(\mathbf{w})$ знак равно $\mathbf{I}(\mathbf{w})_1, \dots, \mathbf{I}(\mathbf{w})_p$, $p > 2$

Гарантировать: матрица размешивания \mathbf{W} знак равно $\mathbf{w}_1, \dots, \mathbf{w}_p$

```

W ← []
за k знак равно 1 к d делать
    X(k) ← X(k-1) - WW^T X(k-1)
p ← randn(1, p), c ← Randperm(1, ..., p)^T, t ← X(k)_p, ш
    ← нормальный(t), за я знак равно 2 к p делать
        t' ← t - 2pc I(w)_c, ш' ← нормальный(t')
        если ||I(w)_ш'||_2 < v тогда
            t ← t', ш ← ш'
        конец, если
    конец для
W ← [W, ш]
конец для
W ← W^T
  
```

наш жадный алгоритм, сначала случайный вектор $\mathbf{p} \in \{\pm 1\}^p$ элементы которого равны 1 или -1, генерируются для формирования начальной суммы \mathbf{t} знак равно $\mathbf{I}(\mathbf{w}) \mathbf{p}$. потом \mathbf{t} нормирован на единичный вектор \mathbf{w} , и \mathbf{w} норма проекции $\mathbf{I}(\mathbf{w}) \mathbf{w}$ вычисляется и сохраняется как c . После этого образец $\mathbf{I}(\mathbf{w})$ случайным образом выбирается для формирования новой суммы \mathbf{t}' $\mathbf{I}(\mathbf{w})$ полярность s отрицается от исходного вектора суммы \mathbf{t} . Нормализованный вектор \mathbf{w}' из \mathbf{t}' проверяется, нужно ли уменьшить \mathbf{w} норма проекции $\mathbf{I}(\mathbf{w}) \mathbf{w}'$. Если да, то это более подходящий кандидат для оптимального вектора несмешивания и \mathbf{w} и \mathbf{t} заменены на \mathbf{w}' и \mathbf{t}' , соответственно. Эта процедура продолжается для всех образцов. Временная сложность предложенного алгоритма примерно в три раза выше, чем у супергауссова случая.

Чтобы измерить оптимальность предложенного жадного алгоритма, мы сравнили оптимальность жадного решения со всеми возможными комбинациями $\{\pm 1\}^p$ для небольшого количества образцов p следующим образом: Для смешанных субгауссовских источников равномерного распределения

ции, мы находим \mathbf{sh} в соответствии с [Алгоритм 3](#) с участием \mathbf{p} знак равно 3. Кроме того, \mathbf{sh} ближайшая к каждой точке $\{\pm 1\}^n$ рассчитывается для всевозможных баллов. Затем ранг Lp-ICA-F определяется как количество комбинаций, имеющих меньшее значение $||\mathbf{X}\mathbf{w}||_2$ чем жадное решение. Результат экспериментов для кейсов \mathbf{p} знак равно 10, 15, 20, а 25 проиллюстрировано на [Рис. 2](#). Соотношение рангов рассчитывается делением ранга на 2. \mathbf{p} и среднее и медианное соотношение рангов 100 испытаний для каждого случая представлены на рисунке. Показано, что соотношение рангов уменьшается с увеличением количества выборок. Следовательно, можно ожидать, что предложенный алгоритм Lp-ICA-F найдет решение, очень близкое к оптимальному для большого набора данных. Поскольку проблема ICA довольно часто касается случая \mathbf{d} \mathbf{n} , жадное решение даст многообещающий результат. Экспериментальный результат в следующем разделе подтверждает это утверждение.

Алгоритмы Lp-ICA-F + и Lp-ICA-F могут быть объединены для разделения произвольных источников. В этом случае \mathbf{k} -й вектор размешивания \mathbf{sh}_k можно найти путем сравнения логарифма правдоподобия (8) решений Lp-ICA-F + и Lp-ICA-F и установив его на то, которое выводит максимальное логарифмическое правдоподобие. Этот подход, который обозначается как Lp-ICA-F, используется во всех экспериментах в следующем разделе. Вычислительная сложность Lp-ICA-F примерно в четыре раза выше, чем у базового Lp-ICA-F +, показанного на [Алгоритм 2](#).

6. Моделирование

В этом разделе производительность предложенных алгоритмов Lp-ICA-G в [Раздел 4](#) и Lp-ICA-F в [Раздел 5](#), сравниваются с FastICA [12] и RobustICA [18]. Мы рассмотрели два типа широко используемых негауссовских распределений для моделирования. Распределение Лапласа рассматривается как представитель супергауссовских источников. Поскольку речевой сигнал приблизительно соответствует лапласовскому распределению, ICA для лапласовского распределения часто используется для слепого разделения источников. С другой стороны, равномерное распределение представляет субгауссовские источники, где ICA может применяться для удаления шума. В следующих экспериментах лапласиан и равномерное распределение генерируются с нулевым средним и единичной дисперсией. После того, как сгенерированные источники смешиваются с помощью матрицы случайного смешивания, смешанные сигналы отбеливаются с помощью процесса отбеливания, описанного в [Раздел 2](#). Затем на отбеленных данных выполняются алгоритмы Lp-ICA-G, Lp-ICA-F, FastICA и RobustICA. Эффективность разделения измеряется с точки зрения отношения сигнал / шум (SNR), которое вычисляется с помощью

$$\text{SNR (дБ)} = \frac{1}{d} \sum_{k=1}^d \left(\frac{s_k}{\sqrt{\frac{1}{d} \sum_{k=1}^d s_k^2}} \right)^2 \quad (26)$$

где s_k и y_k являются k -го первоисточника и k -й предполагаемый источник соответственно. Все эксперименты проводятся по 100 раз для каждого случая, и в этой статье приведены средние значения отношения сигнал / шум. Для алгоритмов FastICA и RobustICA использовались настройки по умолчанию исходной реализации MATLAB авторов. В FastICA функция контраста 'row3' (\mathbf{g} (и) знак равно $\frac{1}{2} \log \frac{1}{1 + \mathbf{g}^2}$), что является настройкой по умолчанию для реализации MATLAB. Мы выбрали функцию контраста, поскольку ее можно использовать для разделения как субгауссовых, так и супергауссовых источников. Все эксперименты проводились с использованием MATLAB на ядре Intel i7 на частоте 2,20 ГГц.

6.1. Бесшумный корпус

Во-первых, чтобы измерить производительность различных алгоритмов для базового случая, размер данных фиксируется на \mathbf{p} знак равно 500 и \mathbf{d} знак равно 8. Мы изменили количество суб- и супергауссовых источников от 0 до 8 и сообщили средние отношения сигнал / шум в [Таблица 2](#). Во всех послесловиях таблиц числа в скобках представляют собой стандартные отклонения. Для случая только субгаусса (8: 0) алгоритм PCA-L1, который использует метод Лагранжа-норма максимизация [23] был

Таблица 2

Средний SNR для \mathbf{p} знак равно 500 и \mathbf{d} знак равно 8 (в дБ). В крайнем левом столбце указано количество суб- и супергауссовых источников. Последняя строка - это среднее время в мс для всех экспериментов.

Sub: Супер	Lp-ICA-G	Lp-ICA-F	FastICA	RobustICA
0: 8	12,12 (3,27)	9,87 (1,51)	8,51 (2,02)	8,76 (2,06)
2: 6	14,55 (1,95)	13,65 (2,12)	10,97 (1,76)	10,56 (1,62)
4: 4	16,76 (1,92)	17,06 (1,36)	13,46 (1,87)	12,49 (1,56)
6: 2	18,67 (2,48)	18,50 (1,18)	16,75 (1,57)	15,74 (1,53)
8: 0	19,63 (4,08)	19,54 (1,07)	20,57 (1,25)	20,83 (1,10)
Время (мс)	1850 г.	19,5	6.5	13,5

Таблица 3

t-значения одностороннего t-тест для различных гипотез. Положительные значения t указывают на то, что Lp-ICA превосходит сравниваемый метод, в то время как отрицательные значения t указывают на неполноценность Lp-ICA.

Sub: Супер	Lp-ICA-G знак равно FastICA	Lp-ICA-F знак равно FastICA	Lp-ICA-G знак равно RobustICA	Lp-ICA-F знак равно RobustICA
0: 8	9,392	5,393	8,694	4,346
2: 6	13,63	9,727	15,74	11,58
4: 4	12,31	15,57	17,26	22,08
6: 2	6,541	8,910	10,05	14,28
8: 0	- 2,203	- 6,260	- 2,840	- 8,406

также протестированы, и полученное SNR составило 9,71 дБ со стандартным отклонением 3,66 дБ, что намного ниже показателей других алгоритмов. Для чистых субгауссовских источников (8: 0) лучше всего работал RobustICA, но другие алгоритмы также работали хорошо. В этом случае, поскольку жадный алгоритм Lp-ICA-F не проверяет все 2 \mathbf{p} возможных решений, он немного уступает другим алгоритмам. Только для супергауссовского случая (0: 8) мы измерили производительность Lp-PCA [24] с участием \mathbf{p} знак равно 3. Среднее SNR составило 14,21 дБ со стандартным отклонением 1,47, что превосходит Lp-ICA и все другие алгоритмы. Однако алгоритм Lp-PCA не устойчив к шуму, что будет обсуждаться в [Раздел 6.2](#). В других случаях предложенные алгоритмы, особенно Lp-ICA-G, работали лучше, чем FastICA и RobustICA. Однако Lp-ICA-G работает намного медленнее, чем другие алгоритмы, поскольку его скорость сходимости линейна. Время работы Lp-ICA-F в три раза больше, чем у FastICA, в то время как Lp-ICA-F дает лучший результат реконструкции, чем FastICA.

Мы провели t-тест на результат в [Таблица 2](#) для проверки статистической значимости. Односторонний t-тест используется для статистической проверки того, превосходят ли Lp-ICA-G и Lp-ICA-F FastICA и RobustICA или нет. Значения t для различных гипотез представлены в [Таблица 3](#). Мы выдвинули гипотезы, чтобы проверить превосходство алгоритмов Lp-ICA над FastICA, и RobustICA является статистически значимым. Пороги для односторонних t-тест составляют около 1,65 для 95% доверительного интервала и около 2,34 для 99% доверительного интервала соответственно. Большинство t-значений, представленных в [Таблица 3](#) превышают пороговые значения с большими запасами. Следовательно, результат статистически оправдывает, что Lp-ICA-G и Lp-ICA-F превосходят FastICA и RobustICA в том случае, когда существуют только супергауссовские источники и когда супер- и субгауссовские источники смешаны. С другой стороны, когда существуют только субгауссовские источники, FastICA и RobustICA работают лучше, чем алгоритмы Lp-ICA.

Затем алгоритмы были протестированы на различном количестве образцов. Количество отсчетов варьируется: 100, 300, 500, 700 и 900 с фиксированным количеством источников (3 субгауссовых, 3 супергауссовских). Результаты показаны в [Таблица 4](#). SNR увеличивается по мере увеличения размера выборки. Lp-ICA-F работает лучше всего во всех случаях. Более того, разрыв между Lp-ICA-F и другими алгоритмами имеет тенденцию увеличиваться по мере увеличения количества выборок. Помимо плохой производительности в случае смешанного источника, FastICA имеет еще один недостаток, заключающийся в том, что иногда не удается сходиться при небольшом размере выборки. В нашем эксперименте 7 из 100 испытаний не смогли сойтись за \mathbf{p} знак равно 100, и эти случаи были исключены из общего списка.

Таблица 4

Среднее отношение сигнал / шум и время с различным количеством выборок.

# Образцы Lp-ICA-G	Lp-ICA-F	FastICA	RobustICA
(a) SNR (в дБ)			
100	8,25 (3,27)	9,37 (2,80)	7,53 (2,91)
300	16,09 (2,82)	16,20 (2,25)	13,46 (2,30)
500	18,72 (2,23)	19,00 (1,46)	15,82 (2,15)
700	20,63 (1,89)	20,94 (1,72)	16,95 (2,58)
900	21,62 (2,37)	22,09 (1,60)	18,08 (2,23)
(б) Время (в мс)			
100	247 (11)	2,08 (0,97)	47,70 (85,21)
300	606 (11)	6,94 (0,20)	3,93 (10,65)
500	927 (14)	13,46 (0,55)	2,98 (1,08)
700	775 (14)	21,52 (1,65)	3,38 (1,35)
900	948 (22)	31,53 (2,89)	3,72 (0,92)

расчет среднего и стандартного отклонения. Во всех случаях FastICA работает лучше, чем RobustICA.

Глядя на время вычислений в Таблица 4(б) Lp-ICA-G выполнялся очень медленно по сравнению с другими алгоритмами, потому что это простой метод на основе градиента. Для небольшого количества образцов (100) Lp-ICA-F был самым быстрым, но по мере увеличения количества образцов он замедлялся. Найти эмпирическую степенную зависимость между количеством выборок n и время, мы использовали пять баллов и смогли получить $время(PC) \propto n^{1,23}$. Обратите внимание, что показатель степени 1,23 намного меньше, чем 2, что является результатом теоретического анализа. Скорость FastICA и RobustICA была почти постоянной независимо от количества образцов. Относительно большое количество для FastICA, когда n равно 100 из-за 7 неудачных испытаний.

Таблица 5 сообщает о выступлениях с разным количеством источников. В экспериментах количество выборок было зафиксировано на 500. Мы увеличили количество источников с 2 до 18, при этом количество суб- и супергауссовых источников было установлено одинаковым. Эффект увеличения количества источников аналогичен уменьшению количества выборок. Lp-ICA-F дал лучший результат, и одно испытание FastICA не удалось сойтись за d знак равно 18. Результаты показаны на Таблица 5(а).

Чтобы проверить характеристики предложенных методов для источников, которые не следуют обобщенному гауссовскому распределению, мы сгенерировали два типа несимметричного распределения следующим образом: (1) (Lap + Gau) Половина выборок была сгенерирована лапласианом и соответственно, гауссовские распределения и выборки из лапласовского распределения были вынуждены иметь отрицательные значения, а гауссовские - положительные значения путем применения операции абсолютного значения к выборкам. (2) (Uni + Gau) В процедуре генерации источников первого типа (Lap + Gau) лапласианское распределение было заменено равномерным распределением. В Таблица 5(б) мы сообщали о выступлениях с разным количеством источников. Во всех экспериментах количество источников (Uni + Gau) и (Lap + Gau) задавалось одинаковым. В таблице мы видим, что производительность алгоритмов ICA резко ухудшается по мере увеличения количества несимметрично распределенных источников. За исключением случая d знак равно 8, Lp-ICA-F показал себя лучше всех.

6.2. Шумный случай

В этом эксперименте к исходному источнику добавляются гауссовский шум и однородный шум. Шум моделируется как случайная величина с нулевым средним, не зависящая от источника. Мы варьировали дисперсию гауссова шума и поддержку равномерного шума. Для каждого эксперимента было установлено n знак равно 500 и d знак равно 6. Использовалось равное количество суб- и супергауссовских источников. Характеристики с аддитивным гауссовым шумом и однородным шумом показаны на Таблица 6. Результаты Lp-PCA [24] также включены, где n знак равно 1 и n знак равно 3 используются для разделения субгауса и супергауса соответственно. SNR уменьшается по мере увеличения дисперсии шума.

Таблица 5

Среднее отношение сигнал / шум при разном количестве источников (в дБ). (а) Лапласиан и равномерное (б) Распределения Lap + Gau и Uni + Gau.

# Источники Lp-ICA-G	Lp-ICA-F	FastICA	RobustICA
(а) Такое же количество источников из лапласовского и равномерного распределений.			
2	32,99 (7,88)	33,11 (6,68)	27,50 (8,78)
6	18,77 (2,07)	19,43 (1,77)	15,82 (2,43)
10	14,75 (2,00)	15,32 (1,23)	11,89 (1,46)
14	11,61 (2,37)	12,17 (1,71)	9,44 (1,32)
18	8,15 (2,34)	7,84 (2,45)	6,56 (1,61)
(б) Такое же количество источников из распределений Лап + Гау и Уни + Гау 2			
	26,91 (8,30)	29,11 (7,50)	25,90 (9,50)
4	8,23 (1,92)	8,25 (0,78)	7,96 (0,80)
6	4,77 (1,40)	5,02 (0,36)	4,67 (1,15)
8	3,27 (1,02)	3,21 (1,08)	3,09 (1,07)

Таблица 6

Среднее отношение сигнал / шум с аддитивным шумом (в дБ).

(а) Аддитивный гауссов шум					
Стд.	Lp-ICA-G	Lp-ICA-F	FastICA	RobustICA	Lp-PCA
10-3	18,78 (2,09)	18,93 (1,66)	15,87 (2,25)	14,39 (1,80)	14,53 (2,96)
10-2	16,06 (3,45)	16,12 (3,30)	13,50 (2,81)	12,69 (2,49)	13,34 (3,72)
10-1	7,00 (2,35)	7,04 (2,31)	6,92 (1,86)	6,73 (1,79)	6,11 (2,32)
(б) Аддитивный равномерный шум					
Интервал	Lp-ICA-G	Lp-ICA-F	FastICA	RobustICA	Lp-PCA
$\pm 5 \times 10^{-4}$	18,87 (2,16)	19,15 (1,59)	16,02 (2,54)	14,59 (1,93)	15,17 (2,84)
$\pm 5 \times 10^{-3}$	17,89 (2,73)	18,23 (2,66)	14,93 (2,75)	13,87 (2,31)	14,76 (2,83)
$\pm 5 \times 10^{-2}$	12,60 (3,45)	12,88 (3,47)	11,36 (2,73)	10,84 (2,52)	10,84 (3,27)

Таблица 7

Среднее отношение сигнал / шум источников с гауссовым смещением для различных соотношений гауссовских отсчетов (в дБ).

Коэффициент Га.	Lp-ICA-G	Lp-ICA-F	FastICA	RobustICA
0,1	16,76 (3,03)	17,96 (1,69)	14,60 (2,06)	13,66 (2,06)
0,2	16,13 (2,10)	16,47 (2,14)	13,71 (1,95)	13,23 (1,99)
0,3	13,78 (2,84)	14,79 (2,38)	12,35 (2,02)	12,16 (2,12)
0,4	11,92 (3,63)	13,24 (2,55)	11,42 (2,26)	11,14 (2,32)
0,5	9,72 (3,10)	10,58 (2,72)	9,72 (2,57)	9,54 (2,23)

Относительные характеристики между методами аналогичны случаю бесшумности. В этом эксперименте можно проверить, что и Lp-ICA-G, и Lp-ICA-F хорошо работают в присутствии шума, но Lp-ICA-F немного лучше, чем Lp-ICA-G. Lp-PCA во всех случаях работает хуже, чем Lp-ICA.

Наконец, чтобы сделать распределение источников более сложным, распределение источников изменяется с лапласовского (однородного) на смесь гауссовского и лапласовского (однородного) для супергауссовского (субгауссовского) источника. Другими словами, каждый супергауссовский (субгауссовский) источник содержит выборки из распределения Лапласа (равномерного) вместе с выборками из гауссова распределения с нулевым средним и единичной дисперсией. Распределение выборок по-прежнему суб- или супергауссово в зависимости от исходного несмешанного распределения, но распределение становится ближе к гауссовскому распределению по мере увеличения отношения гауссовых выборок, т. Е. Абсолютное значение нормированного эксцесса распределения становится меньше. SNR измерены для различных соотношений гауссовых отсчетов от 0,1 до 0,5. Результаты показаны в Таблица 7. Lp-ICA-F показал лучшую производительность. FastICA не удалось сойтись в 5 случаях, когда отношение гауссовых отсчетов составляет 0,5.

7. Эксперименты на реальных данных

Чтобы показать применимость предложенного метода на реальных данных, мы применили Lp-ICA в данных электроэнцефалографии (ЭЭГ). Мы использовали набор данных из [8] который содержит 71-канальный сигнал ЭЭГ

13 предметов.⁶ Предлагаются две схемы оценки для измерения производительности алгоритмов ICA в [8]. Во-первых, взаимное сокращение информации (MIR) между записанными каналами данных и восстановленными компонентами измеряет независимость с точки зрения взаимной информации. Во-вторых, поскольку независимые источники ЭЭГ дипольны [8], количество разложенных сигналов, которые являются почти дипольными, можно использовать для оценки качества разложенных сигналов. Каждый разложенный сигнал помещается в один эквивалентный диполь с помощью подключаемого модуля DIP-FIT из набора инструментов EEGLAB [32]. Затем вычисляется остаточная дисперсия, которая представляет собой ошибку дипольной подгонки. Компоненты, близкие к дипольным, определяются как сигналы, остаточная дисперсия которых составляет менее 5%. Мы отправляем читателей к [7,8] для получения подробной информации о методах оценки. Хорошее ICA-разложение имеет высокое значение MIR и большое количество источников, близких к дипольному разложению. Для эксперимента мы использовали первые 10 000 отсчетов исходных сигналов.

Мы измерили MIR (кбит / с) и долю ближних дипольных источников (%) для результатов различных алгоритмов ICA и PCA, и мы наносим результаты на 2D-плоскость (Рис. 3). И Lp-ICA-F, и Lp-ICA-G генерируют больше почти дипольных источников, чем FastICA и RobustICA. Между тем, значения MIR алгоритмов Lp-ICA меньше, чем FastICA, и аналогичны RobustICA. Таким образом, Lp-ICA превосходит RobustICA и конкурирует с FastICA в области декомпозиции данных ЭЭГ. Этот результат демонстрирует применимость алгоритмов Lp-ICA к данным реального мира.

8. Заключение

В этой статье ICA анализируется с другой точки зрения, а также пара новых алгоритмов ICA, основанных на l_p -предлагается минимизация нормы. Предлагаемые методы предполагают, что источники распределены согласно обобщенному распределению Гаусса, т. е. $p(s)$ знак равно $\alpha \exp(\beta |s|^\eta)$. Применяя оценку максимального правдоподобия, ICA сводится к l_p -проблема минимизации нормы, которая имеет очень тесную связь с PCA на основе l_p -норма максимизация [24]. Взаимосвязь между предлагаемыми l_p -также обсуждается минимизация нормы и оптимизация эксцесса. Кроме того, мы также показываем, что алгоритм ICA может быть сформулирован как l_p -задача минимизации или максимизации нормы в зависимости от типа источников, связанных с работой в [25].

В дополнение к градиентному методу были также предоставлены быстрые неитерационные алгоритмы для супер- и субгауссовских источников. Результаты моделирования показывают, что производительность предложенного алгоритма Lp-ICA-F совместима с традиционными алгоритмами или превосходит их. В частности, наш метод обеспечивает превосходные характеристики для супергауссовских источников и в присутствии шума.

В качестве будущей работы мы можем думать о неитеративной версии Lp-ICA-F или о решении (10) прямо без релаксации. Строгое доказательство оптимальности может быть еще одной проблемой исследования.

Подтверждение

Это исследование было поддержано Программой фундаментальных научных исследований через Национальный исследовательский фонд Кореи (NRF-2016R1A1A1A05005442).

использованная литература

- [1] И. Джоллифф, Анализ главных компонент, Интернет-библиотека Wiley, 2005.
- [2] П. Комон, Независимый компонентный анализ, новая концепция? Сигнальный процесс. 36 (3) (1994) 287–314.

- [3] А. Хювяринен, Дж. Карунен, Э. Оя, Независимый компонентный анализ, 46, John Wiley & Sons, 2004.
- [4] XL Li, T. Adali, Независимый компонентный анализ путем минимизации энтропийной границы, IEEE Trans. Сигнальный процесс. 58 (10) (2010) 5151–5164, DOI:10.1109 / TSP. 2010.2055859.
- [5] П. Комон, К. Юттен, Справочник по слепому разделению источников: независимый компонентный анализ и приложения, Academic Press, 2010.
- [6] NC Rogasch, RH Thomson, F. Farzan, BM Fitzgibbon, NW Bailey, JC Hernandez-Pavon, ZJ Daskalakis, PB Fitzgerald, Удаление артефактов из записей TMS-EEG с помощью независимого компонентного анализа: важность для оценки свойств префронтальной и моторной сети коры головного мозга, Neuroimage 101 (2014) 425–439.
- [7] С.-Х. Сюй, Т. Маллен, Т.-П. Юнг, Г. Каувербергс, Проверка онлайн-рекурсивного независимого компонентного анализа данных ЭЭГ, в: Neural Engineering (NER), 2015 г., 7-я Международная конференция IEEE / EMBS, IEEE, 2015, стр. 918–921.
- [8] А. Делорм, Дж. Палмер, Дж. Онтон, Р. Остенвельд, С. Макейг, Независимые источники ЭЭГ дипольны, PLoS ONE 7 (2) (2012) e30135.
- [9] Р. Йенсен, Т. Элтофт, Независимый компонентный анализ для сегментации текстуры, Распознавание образов. 36 (10) (2003) 2301–2315. doi: [https://doi.org/10.1016/S0031-3203\(03\)00131-6](https://doi.org/10.1016/S0031-3203(03)00131-6).
- [10] П.С. Юэн, Дж. Лай, Представление лиц с использованием анализа независимых компонентов, Распознавание образов. 35 (6) (2002) 1247–1257. doi: [https://doi.org/10.1016/S0031-3203\(01\)00101-7](https://doi.org/10.1016/S0031-3203(01)00101-7).
- [11] И. Даре, Р. Начар, Распознавание лиц с использованием алгоритма IPCA-ICA, IEEE Trans. Pattern Anal. Max. Intell. 28 (6) (2006) 996–1000, DOI:10.1109 / TPAMI.2006.118.
- [12] А. Хювяринен, Быстрые и надежные алгоритмы с фиксированной точкой для независимого компонентного анализа, Neural Netw. IEEE Trans. 10 (3) (1999) 626–634.
- [13] А. Хювяринен, Э. Оя, Независимый компонентный анализ: алгоритмы и приложения, Нейронные сети. 13 (4) (2000) 411–430.
- [14] Т.-В. Ли, М. Джиролами, Т. Дж. Сейновски, Независимый компонентный анализ с использованием расширенного алгоритма Infomax для смешанных субгауссовских и супергауссовских источников, Neural Comput. 11 (2) (1999) 417–441.
- [15] З. Ши, Х. Тан, Й. Тан, Новый алгоритм с фиксированной точкой для независимого компонентного анализа, Нейровычисления 56 (0) (2004) 467–473.
- [16] Ф. Р. Бах, М. И. Джордан, Анализ независимых компонентов ядра, J. Mach. Учиться. Res. 3 (2003) 1–48.
- [17] З. Колдовский, П. Тихавский, Э. Оя, Эффективный вариант алгоритма FastICA для независимого компонентного анализа с достижением нижней границы Крамера-Рао, Нейронные сети, IEEE Trans. 17 (5) (2006) 1265–1277.
- [18] В. Зарзосо, П. Комон, Робастный независимый компонентный анализ путем итерационной максимизации эксцесса контраста с алгебраическим оптимальным размером шага, IEEE Trans. Нейронная сеть. 21 (2) (2010) 248–261.
- [19] Дж. Ченг, К. Лю, Х. Лу, Ю.-В. Чен, Ансамблевое обучение для независимого компонентного анализа, Распознавание образов. 39 (1) (2006) 81–88. doi: <https://doi.org/10.1016/j.patcog.2005.06.018>.
- [20] А. Шарма, К. К. Паливал, Подпространственно независимый компонентный анализ с использованием векторного эксцесса, Распознавание образов. 39 (11) (2006) 2227–2232. doi: <https://doi.org/10.1016/j.patcog.2006.04.021>.
- [21] Z. Szab, B. Pczos, A. Lincz, Теорема разделения для независимого анализа подпространств и ее следствия, Распознавание образов. 45 (4) (2012) 1782–1791. doi: <https://doi.org/10.1016/j.patcog.2011.09.007>.
- [22] Ч. Дин, Д. Чжоу, Х. Хе, Х. Ча, R 1-PCA: анализ главных компонент с l_1 -нормой вращения для устойчивой факторизации подпространств, в: Труды 23-й Международной конференции по машинному обучению, ACM, 2006, С. 281–288.
- [23] Н. Квак, Анализ главных компонент на основе максимизации l_1 -нормы, Pattern Anal. Max. Intell. IEEE Trans. 30 (9) (2008) 1672–1680.
- [24] Н. Квак, Анализ главных компонент с помощью максимизации l_p -нормы, Cybern. IEEE Trans. 44 (5) (2014) 594–609.
- [25] Р. Мартин-Клементе, В. Зарзосо, О связи между l_1 -PCA и ICA, IEEE Trans. Pattern Anal. Mach. Intell. <https://doi.org/10.1109/TPAMI.2016.2557797>.
- [26] Ф. Дж. Тайс, А. Юнг, К. Г. Пунтонет, Э. У. Ланг, Линейный геометрический ICA: основы и алгоритмы, Neural Comput. 15 (2) (2003) 419–439.
- [27] Ж.-Ф. Кардосо, Контракты высокого порядка для независимого компонентного анализа, Neural Comput. 11 (1) (1999) 157–192.
- [28] Дж. В. Стоун, Анализ независимых компонент: Введение в учебное пособие, MIT Press, Кембридж, Массачусетс, Лондон, Англия, 2004.
- [29] RJ Samworth, M. Yuan, Независимый компонентный анализ с помощью непараметрической оценки максимального правдоподобия, Ann. Stat. 40 (6) (2012).
- [30] М. Аарон, М. Элад, А. Брукштейн, K-SVD: алгоритм для разработки переполненных словарей для разреженного представления, IEEE Trans. Сигнальный процесс. 54 (11) (2006) 4311–4322.
- [31] У. Файгл, Жадные алгоритмы в комбинаторной оптимизации, в: Протоколы исследования операций, 1993 г., в: Труды исследования операций, 1993 г., Springer Berlin Heidelberg, 1994 г., doi:10.1007 / 978-3-642-78910-6_161.
- [32] А. Делорм, Т. Маллен, К. Кот, З. А. Акар, Н. Бигдели-Шамло, А. Ванков, S. Makeig, EEGLAB, SIFT, NIFT, BCILAB и ERICA: новые инструменты для расширенной обработки ЭЭГ, Comput. Intell. Neurosci. 2011 (2011) 10.

⁶ <https://scn.ucsd.edu/wiki/BSSComparison>.

Парк Сунхон получил степень бакалавра и магистра информатики в Корейском институте науки и технологий (KAIST), Тэджон, Корея, в 2012 и 2014 годах соответственно. В настоящее время он защищает докторскую диссертацию. степень в области интеллектуальных систем в Лаборатории машинного интеллекта и анализа структур (MIPAL) Сеульского национального университета, Сеул, Корея. Его исследовательские интересы включают компьютерное зрение и глубокое обучение.

Ноджун Квак родился в Сеуле, Корея, в 1974 году. Он получил степени бакалавра, магистра и доктора философии в Школе электротехники и компьютерных наук Сеульского национального университета, Сеул, Корея, в 1997, 1999 и 2003 годах соответственно. С 2003 по 2006 год он работал в Samsung Electronics. В 2006 году он поступил в Сеульский национальный университет в качестве доцента BK21. С 2007 по 2013 год он был преподавателем кафедры электротехники и вычислительной техники Университета Аджу, Сувон, Корея. С 2013 года он работал в Высшей школе конвергенции науки и технологий Сеульского национального университета, Сеул, Корея, где в настоящее время является доцентом. Его текущие исследовательские интересы в основном связаны с обучением функций глубокими нейронными сетями и их приложениями в различных областях распознавания образов, компьютерного зрения, обработки изображений и так далее.