

Классификация расстройств аутистического спектра и нормального развития на основе сходства разбиений сетевых структур мозга

Анвар Курмуков

*Национальный исследовательский университет «Высшая школа экономики»
kurmukovai@gmail.com*

Юлия Додонова

*Национальный исследовательский университет «Высшая школа экономики»
ya.dodonova@mail.ru*

Леонид Жуков

*Национальный исследовательский университет «Высшая школа экономики»
lzhukov@hse.ru*

Аннотация

Мы решаем задачу различения пациентов с расстройствами аутистического спектра и людей без патологии на основе графов структурных связей головного мозга (коннектомов). Для этого мы предлагаем использовать возможные различия в разбиениях графов на подграфы, характерные для коннектомов групп нормы и патологии. Мы используем четыре метода кластеризации, чтобы получить разбиения коннектомов на подграфы. Мы оцениваем попарные расстояния между полученными разбиениями и строим на их основе ядро для SVM классификатора. Полученные классификаторы мы объединяем в двухуровневую модель с использованием стэкинга. Качество классификации для двухуровневой модели достигает 0.73 в смысле площади под ROC-кривой (ROC AUC).

1. Введение

В последние годы в нейронауках активно развивается направление, в рамках которого данные нейровизуализации головного мозга человека представляются в виде графа, вершины которого репрезентируют зоны мозга, а ребра - связи между ними. Такие графы получили название коннектомы.

Для коннектомов характерны некоторые особенности. Это довольно маленькие графы - порядка сотен вершин, определяемых в соответствии с тем или иным стандартным атласом мозга. Важно, что каждая вершина всегда имеет уникальную метку; в случае, когда речь идет о наборе коннектомов разных

людей, набор меток одинаков для всех коннектомов.

Структурные коннектомы являются ненаправленными графами с симметричными матрицами смежности - алгоритмы трактографии, работающие на основе диффузионной магнитно-резонансной томографии, позволяют выявить наличие волокон (пучков белого вещества) между зонами мозга, то есть наличие структурной связи, но не ее направление. Получаемые коннектомы являются взвешенными, и веса пропорциональны количеству трактов, выявленных между соответствующими зонами мозга.

Одна из содержательных задач, возникающих в коннектомике - это задача классификации таких графов. Речь идет, например, о различении на основе структурных коннектомов людей с нормальным и патологическим развитием (расстройствами аутистического спектра, шизофренией, болезнью Альцгеймера и т.п.). Постановка проблемы при этом такова. Пусть X_i - коннектом (граф, представляющий сетевую структуру головного мозга человека), y_i - метка класса (в данной работе речь пойдет только о $y_i \in \{0, 1\}$). Имея обучающую выборку, состоящую из пар (X_i, y_i) , и тестовый набор объектов X_j , необходимо предсказать неизвестные метки класса y_j .

Один из простейших подходов к решению этой задачи состоит в том, чтобы превратить коннектом в вектора из \mathbb{R}^p и решать задачу классификации на этих векторах. Для этого могут использоваться различные приемы, от простого разворачивания в вектор верхнего треугольника матрицы смежности до подсчета на исходных графах различных сетевых метрик. Другой возможный подход состоит

в том, чтобы каким-либо способом оценивать расстояния между имеющимися графами и использовать матрицу таких попарных расстояний - например, для построения ядра SVM классификатора. В наших предыдущих работах мы анализировали поведение классификаторов для коннектомов, использующих ядра, основанные на случайном блуждании и длинах кратчайших путей [1], а также расстояниях Кульбака-Лейблера [2] и Васерштейна [3] между спектрами графов.

В данной работе мы предлагаем принципиально другой подход для оценки расстояний между графами. Для каждого коннектома мы оцениваем оптимальное для этого конкретного графа разбиение на подграфы (кластеры). После этого в качестве расстояния между графами мы рассматриваем сходство их оптимальных разбиений (векторов меток вершин, указывающих на их принадлежность к кластерам).

В контексте задачи классификации коннектомов это означает следующее. Мы хотим различать классы нормального и патологического развития на основе структуры соответствующих графов. Та или иная структура графа задает оптимальное для данного графа разбиение на подграфы. Мы предполагаем, что внутри класса (нормы или патологии) эти разбиения окажутся более похожи, чем между классами.

В данной статье мы демонстрируем работу предложенного алгоритма при решении задачи классификации расстройств аутистического спектра и нормального развития на основе коннектомов. Статья организована следующим образом. В Разделе 2 мы описываем подходы, которые мы использовали для разбиения коннектомов на подграфы, и меры для оценки сходства разбиений. В Разделе 3 мы описываем, каким образом мы организуем машинное обучение на полученных матрицах попарных расстояний между коннектомами. В Разделе 4 представлено описание данных. В Разделах 5 и 6 представлены результаты, их обсуждение и выводы.

2. Разбиение коннектомов на подграфы и оценка сходства разбиений

В этом разделе мы описываем использованные нами подходы к разбиению коннектомов на подграфы, а также метрики, которые мы будем использовать для оценки попарного сходства разбиений. Используемые методы сведены в Таблице 1.

2.1. Разбиение коннектомов на подграфы

Для разбиения коннектомов на подграфы мы используем четыре различных алгоритма кластеризации: Newman Leading eig vector method [5], Louvian method [4], Greedy modularity optimization [6], Label

Таблица 1. Используемые подходы к кластеризации и оценке сходства разбиений

Алгоритм кластеризации	Мера сходства
Newman EIG	ARI, MI
Newman EIG5	ARI, MI
Newman EIG9	ARI, MI
Louvain	ARI, MI
Greedy modularity optimization	ARI, MI
Label Propagation	ARI, MI

Propagation [7]. Первые три метода основаны на построении такой кластерной структуры графа, которая минимизирует его модулярность:

$$Q = \frac{1}{2m} \sum_{ij} [A_{ij} - \frac{d_i d_j}{2m}] \delta(i, j). \quad (1)$$

Здесь A_{ij} - матрица смежности графа, d_i, d_j - степени вершин i, j соответственно, m - общее число ребер, $\delta(i, j) = 1$, если вершины i, j принадлежат одному и тому же кластеру, и $\delta(i, j) = 0$, если разным. В целом все три метода дают схожие результаты. Различия заключаются в том, что Louvain был предложен как алгоритм для графов больших (несколько миллионов вершин) размерностей, и основан на эвристической оценке модулярности, Greedy modularity method - пошаговый иерархический метод, а Newman Leading eig vector - спектральный метод кластеризации, основанный на анализе собственных векторов оператора Лапласа матрицы смежности графа, где оператор Лапласа определяется следующим образом:

$$L = D - A, \quad (2)$$

где D - диагональная матрица степеней вершин графа:

$$d_i = \sum_j A_{ij}. \quad (3)$$

Последний метод использованный нами метод - Label Propagation - основан на матрице случайных блужданий:

$$P = D^{-1}A. \quad (4)$$

Суть метода состоит в том, что изначально всем вершинам графа приписываются уникальные метки, после чего метки меняются итерационно. Вершине приписывается та метка, которая чаще всего встречается среди ее соседей, в случае если таких оказывается несколько, выбирается случайная. Метод так же позволяет учитывать веса на ребрах.

Для построения базовых моделей, кроме 4 вышеописанных алгоритмов использовалась так же модификация Newman Leading eigvector с предопределенным числом кластеров 5 и 9. Таким образом всего

использовалось 6 подходов к кластеризации коннектомов.

2.2. Оценка сходства разбиений

Для сравнения разбиений графов коннектомов использовались меры Adjusted Rand Score (ARI) и Adjusted Mutual Information (AMI).

Для определения индекса ARI между двумя разбиениями $U = \{U_1, U_2, \dots, U_l\}$ и $V = \{V_1, V_2, \dots, V_m\}$, где l, m - число кластеров полученных в разбиениях U, V соответственно, составим следующую таблицу сопряженности:

Таблица 2. ARI

U, V	V_1	V_2	...	V_s	Сумма
U_1	n_{11}	n_{12}	...	n_{1s}	a_1
U_2	n_{21}	n_{22}	...	n_{2s}	a_2
\vdots	\vdots	\vdots	\ddots	\vdots	\vdots
U_l	n_{l1}	n_{l2}	...	n_{ls}	a_l
Сумма	b_1	b_2	...	b_s	

Тогда индекс ARI можно вычислить по следующей формуле:

$$\frac{\sum_{i,j} \binom{n_{i,j}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2} \right] / \binom{n}{2}}{\frac{1}{2} \left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2} \right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2} \right] / \binom{n}{2}}. \quad (5)$$

Adjusted Mutual Information определяется следующим образом:

$$AMI(U, V) = \frac{MI(U, V) - E(MI(U, V))}{\max(H(U), H(V)) - E(MI(U, V))}, \quad (6)$$

где $H(U)$ - энтропия:

$$H(U) = - \sum_{i=1}^l P(i) \log P(i). \quad (7)$$

И ARI и AMI принимают значение на отрезке $[0, 1]$ причем равенство 1 достигается только в том случае, когда два разбиения одинаковы. Мы строили матрицу попарных расстояний между разбиениями, используя в качестве элементов значения $(1 - ARI)$ и $(1 - MI)$ - одинаковые объекты имеют, таким образом, нулевое расстояние.

3. Классификация графов на основе сходства разбиений

В предыдущем разделе мы получили матрицы попарных расстояний между объектами выборки. В этом разделе мы описываем, каким образом мы организуем машинное обучение на основе этих матриц.

3.1. Ядерный SVM

Мы используем ядерный подход и выбираем метод опорных векторов (SVM) в качестве классификатора. Мы пользуемся тем, что в качестве ядра SVM-классификатора на множестве входных данных \mathbb{X} можно использовать любую положительно определенную функцию $K(\mathbf{x}_i, \mathbf{x}_j) : \mathbb{X}^2 \rightarrow \mathbb{R}$, то есть такую, что

$$\sum_{i=1}^n \sum_{j=1}^n K(\mathbf{x}_i, \mathbf{x}_j) c_i c_j \geq 0, \quad (8)$$

для любых конечных последовательностей $(x_1, x_2, \dots, x_n) \in \mathbb{X}$ и любых наборов коэффициентов $(c_1, c_2, \dots, c_n) \in \mathbb{R}$.

Имея матрицу попарных расстояний между объектами \mathbf{x}_i и \mathbf{x}_j с элементами $d(\mathbf{x}_i, \mathbf{x}_j)$, мы можем построить ядро экспоненцированием:

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\alpha d(\mathbf{x}_i, \mathbf{x}_j)}. \quad (9)$$

Положительная полуопределенность полученного ядра гарантирована в том случае, если расстояние $d(\mathbf{x}_i, \mathbf{x}_j)$ является метрикой. В нашем случае это неверно, для используемых нами расстояний между разбиениями не выполняется неравенство треугольника. Похожая проблема обсуждается, например, в работе [8], где авторы строят ядро на основе расстояния Кульбака-Лейблера (которое также не является метрикой). Авторы отмечают, что в данном случае ядро всегда можно сделать положительно определенным за счет соответствующего выбора параметра α , однако при выборе этого параметра важно также не потерять выразительность ядра и его информативность для классификатора. В данной работе мы варьируем параметр α . Кроме того, мы ищем наилучшие значения для параметра штрафа SVM-классификатора. Мы сообщаем результаты классификации для моделей с оптимальным образом подобранными параметрами.

3.2. Стекинг моделей на втором уровне

Выше мы описали процедуру классификации на основе одной матрицы попарных расстояний между объектами выборки. В нашем случае мы используем шесть различных алгоритмов разбиений, и на результатах работы каждого из них строим попарные расстояния двумя способами, получая таким образом 12 вариантов матриц попарного сходства между коннектомами. Кроме того, мы применяем три различных способа предварительной нормировки данных (описаны в следующем разделе), имея, таким образом, 36 матриц попарных расстояний между объектами. Мы сообщаем результаты работы SVM-классификатора для каждой из полученных 36 моделей.

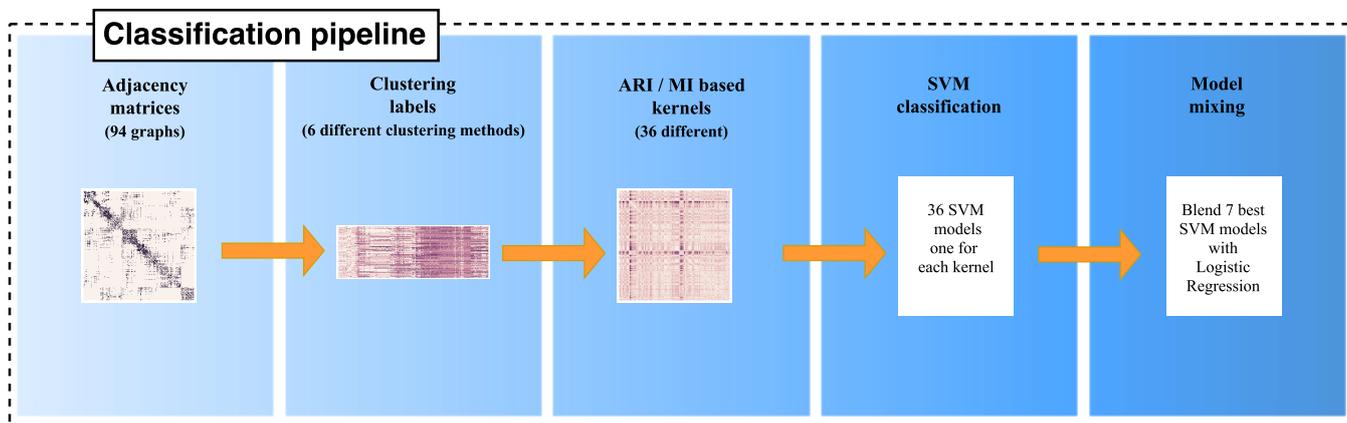


Рис. 1. Итоговая схема анализа.

Далее мы делаем следующий шаг: для каждого алгоритма кластеризации мы выбираем одну модель, дающую наилучшее качество классификации, и строим модель второго уровня. Такой подход может внести некоторое смещение в результаты кросс-валидации, поэтому рамках предварительного анализа мы также работали со всеми моделями первого уровня, реализуя автоматический отбор моделей на втором уровне с помощью регуляризации. Качество предсказания такой модели существенно не отличалось от качества предсказания двухуровневой модели, которую мы описываем в данной работе. При этом с содержательной точки зрения важным для нас оказывается тот факт, что в итоговой модели представлены ядра, полученные для каждого из различных алгоритмов разбиений (для label propagation было выбрано две одинаково хорошо работающие модели, см. секцию 5).

На втором уровне мы используем модель логистической регрессии, которая обучается на предсказаниях моделей первого уровня. При этом мы объединяем векторы предсказанных решений, полученных моделями с ядерным SVM на первом уровне, и используем их в качестве признаков для классификации с помощью логистической регрессии на втором уровне. В модели второго уровня использовалась l_2 регуляризация.

3.3. Итоговый алгоритм анализа и оценка качества

Итоговая схема анализа представлена на Рисунке 1. Мы работали с набором готовых коннектов и получили три параллельных набора данных, используя три алгоритма предварительной обработки. Для каждого коннекта в каждом из наборов данных мы применили каждый из шести алгоритмов разбиения графов на подграфы, описанных в Разделе 2.1, получив таким образом 18 наборов раз-

биений. К каждому из наборов мы применили два алгоритма оценивания сходства разбиений, получив таким образом 36 матриц попарного сходства между объектами.

Используя эти матрицы сходства между объектами, мы строим 36 моделей на основе ядерного SVM и сообщаем результаты работы каждой из моделей. На следующем шаге мы объединяем предсказания 7 лучших моделей и строим на них модель второго уровня, используя логистическую регрессию в качестве классификатора.

Во всех случаях для оценки качества модели используется 10-fold кросс-валидация. В качестве метрики качества классификации мы используем площадь под ROC-кривой (ROC AUC). Мы повторяем работу алгоритма 50 раз с различными разбиениями на обучающие и тестовые выборки и сообщаем среднее значение и стандартное отклонение значений ROC AUC по 50 различным разбиениям.

Все алгоритмы реализованы на языке Python, с использованием платформы IPython notebooks. Мы использовали библиотеки NumPy, SciPy, pandas, matplotlib, seaborn, scikit-learn.

4. Описание данных

Мы работали с открытым набором данных UCLA Autism [9]. В этом разделе мы приводим описание исходного набора данных и использованных нами алгоритмов их предварительной обработки.

4.1. Исходные данные

Набор данных UCLA Autism содержит готовые коннекты 94 человек: 51 человек с расстройствами аутистического спектра (6 женщин, средний возраст 13.0 лет, стандартное отклонение 2.8 лет), 43 человека с нормальным развитием (7 женщин, средний возраст 13.1 лет, стандартное отклонение 2.4 года).

Вершины для этого набора данных определены с использованием схемы, предложенной в работе [10]. Используемое разбиение мозга на зоны основано на большом мета-анализе исследований с использованием функциональной МРТ. Подход предполагает разбиение поверхности мозга на 264 региона и, таким образом, построение матриц размерности 264×264 .

Ребра коннектомов определяются на основе результатов работы детерминистского алгоритма трактографии (алгоритм FACT). Получаемые матрицы смежности представляют, таким образом, взвешенные графы, где веса ребер пропорциональны числу трактов, выявленных между соответствующими регионами мозга.

4.2. Предварительная обработка

Мы использовали три различных подхода к взвешиванию исходных коннектомов. Во-первых, мы работали с исходными коннектомами, веса которых пропорциональны «силе» связи между регионами мозга (числу трактов между ними).

Во-вторых, мы бинаризовали исходные данные:

$$a_{ij}^{binarized} = 1 \quad \text{if} \quad a_{ij} > 0, \quad 0 \quad \text{else}, \quad (10)$$

где a_{ij} - элементы матрицы смежности исходных графов.

В-третьих, мы использовали информацию о физическом расстоянии между регионами мозга и нормировали исходные веса ребер на квадрат расстояния между центрами смежных зон мозга:

$$a_{ij}^{weighted} = \frac{a_{ij}}{l_{ij}^2}, \quad (11)$$

где l_{ij} - евклидово расстояние между вершинами i и j (в качестве координат вершин мы рассматривали центры зон мозга в стандартных MNI координатах, предоставленные авторами набора данных).

Таким образом, мы использовали в анализе три параллельных набора данных. При объединении лучших одноуровневых моделей в двухуровневой модели классификации наша модель верхнего уровня получала на вход модели, работавшие на первом уровне с различным образом взвешенными данными.

5. Результаты и обсуждение

Результаты работы SVM-классификатора на всех используемых ядрах приведены на Рисунках 2 и 3. На Рисунке 2 представлены бокс-плоты значений ROC AUC по 50 повторам алгоритма на разных разбиениях 10-fold кросс-валидации. Цвета плотов соответствуют двум мерам сходства разбиений, в подписи на горизонтальной оси элемент до нижнего подчеркивания означает тип взвешивания данных:

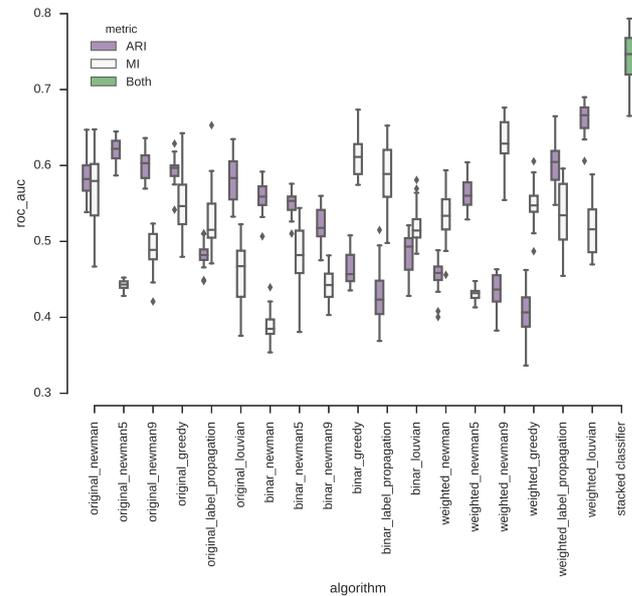


Рис. 2. Значения ROC AUC для одноуровневых моделей с ядерным SVM по 50 повторам алгоритма на разных разбиениях 10-fold кросс-валидации (подписи осей и легенда описаны в тексте)

orig - исходные данные, binar - бинаризованные в соответствии с (10), weighted - взвешенные в соответствии с (11). Элемент после нижнего подчеркивания указывает на использованный алгоритм кластеризации коннектомов. На Рисунке 3 те же результаты представлены в виде тепловой карты для удобства сравнения различных алгоритмов кластеризации и мер сходства на различных входных данных.

Как видно, для каждого из используемых алгоритмов кластеризации использование разных видов взвешивания исходных данных и различных способов подсчета схожести разбиений (ARI, AMI) по-разному влияет на качество классификации. Всего для построения ядер было использовано 6 алгоритмов кластеризации (3 из которых это - это один и тот же алгоритм Newman leading eigenvector, но либо без предопределенного числа кластеров, либо с заданным числом кластеров, равным 5 и 9), 3 вида нормировок и 2 способа сравнения разбиений. Таким образом использовалось $6 \times 3 \times 2 = 36$ различных ядер. Наиболее стабильные результаты, не зависящие от выбора алгоритма кластеризации, получаются с использованием ядер, построенных с помощью оригинальных данных с метрикой сходства ARI (за исключением алгоритма label propagation, который при этом хорошо работает на бинаризованных и оригинальных данных с метрикой AMI).

Лучшими базовыми ядрами оказались ядра, построенные с помощью алгоритмов Newman и Louvian на оригинальных и взвешенных данных для

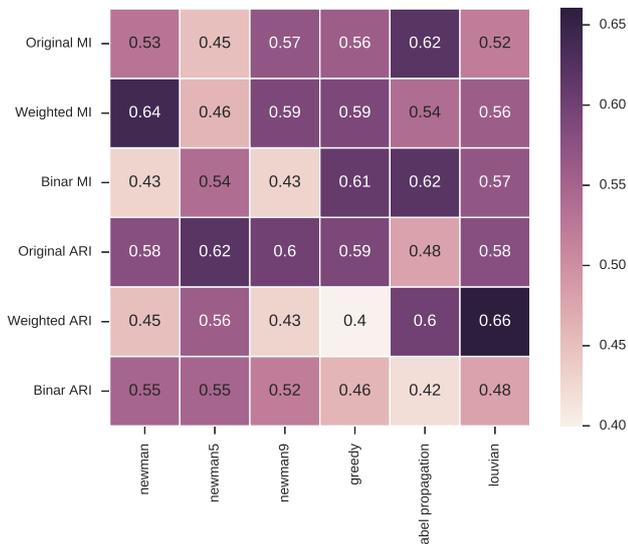


Рис. 3. Тепловая карта средних значений ROC AUC для SVM-классификаторов с ядрами, использующими различные алгоритмы кластеризации коннектов (горизонтальная ось), меры сходства разбиений и типы данных (вертикальная ось).

метрик ARI и AMI, соответственно. Лучший результат, полученный для индивидуальной модели - ROC AUC 0.66.

На следующем шаге для каждого алгоритма кластеризации была выбрана одна модель, дающая наилучшее качество классификации (для label propagation было выбрано две одинаково хорошо работающие модели). Двухуровневая модель классификации с логистической регрессией, основанной на предсказаниях 7 лучших SVM моделей, улучшила итоговое качество классификации в смысле площади под ROC кривой, достигнув результата ROC AUC 0.73 (стандартное отклонение 0.03). Бокс-плот значений ROC AUC для двухуровневой модели показан зеленым цветом на Рисунке 2.

На Рисунке 4 представлены ROC-кривые для лучших индивидуальных моделей по каждому из алгоритмов кластеризации, а также ROC-кривая для итоговой двухуровневой модели.

6. Выводы

Мы решали задачу классификации расстройств аутистического спектра и нормального развития на основе коннектов - графов, представляющих структуру головного мозга человека. Мы предположили, что если структура графов внутри класса (нормы или патологии) более похожа, чем между классами, то графы представителей одного класса будут более похожим образом разбиваться на кла-

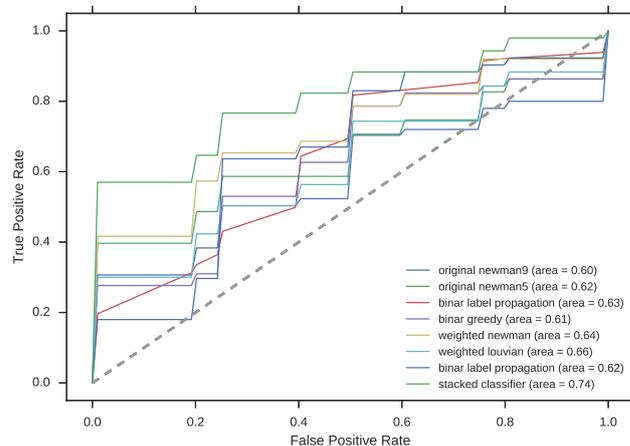


Рис. 4. ROC-кривые для лучших одноуровневых моделей с ядрами на основе каждого из алгоритмов кластеризации, а также для итоговой двухуровневой модели.

стеры, чем графы представителей разных классов.

Для проверки нашего предположения мы разбивали каждый коннектом на оптимальные подграфы, получая таким образом вектора меток принадлежности вершин к подграфам в рамках данного разбиения. Мы оценивали попарные расстояния между разбиениями и рассматривали их в качестве расстояния между соответствующими коннектами. Мы строили ядро на основе полученных матриц попарных расстояний и использовали SVM-классификатор с этим ядром для различения групп расстройств аутистического спектра и нормального развития.

В статье сравнивается поведение шести различных алгоритмов кластеризации коннектов, трех подходов к предварительному взвешиванию коннектов, и двух мер сходства получаемых разбиений. В целом ни один из алгоритмов кластеризации нельзя назвать предпочтительным для всех видов входных данных и мер сходства разбиений. Наилучшая одноуровневая модель позволила получить качество классификации ROC AUC 0.66.

Однако, по-видимому, различные алгоритмы кластеризации ловят различные аспекты структуры исходных коннектов и являются по-своему информативными. Построение двухуровневой модели, объединяющей предсказания лучших моделей для каждого из алгоритмов кластеризации, позволило улучшить качество классификации и получить ROC AUC 0.73.

В данной работе мы демонстрируем работу предложенного алгоритма на примере одной классификационной задачи и одного набора данных. Дальнейшие исследования могут быть направлены на анализ эффективности предложенного метода в раз-

личных классификационных задачах: как в контексте классификации сетевых структур мозга, так и в других предметных областях. В рамках данной предметной области, интересным также будет сравнение эффективности предложенных ядер с другими возможными ядерными методами. Наконец, продолжением данной работы может быть разработка модификаций предложенного алгоритма: в частности, вместо попарных расстояний между разбиениями возможно организовать принятие решения на основе близости между разбиением каждого "тестового" коннектома и оптимального разбиения для некоторого усредненного коннектома каждого из классифицируемых классов. Все перечисленные направления будут предметом нашей дальнейшей работы.

Благодарности

Статья подготовлена в ходе проведения исследования (№ 16-05-0050) в рамках Программы «Научный фонд Национального исследовательского университета «Высшая школа экономики» (НИУ ВШЭ)» в 2016 г. и в рамках государственной поддержки ведущих университетов Российской Федерации "5-100".

Список литературы

[1] Додонова, Ю., Петров, Д., Жуков, Л. (2015). Сравнение эффективности ядер SVM-классификатора для различения пола на основе структурных коннектом. Тр. конф. Информационные Технологии и Системы, 1155-1167.

[2] Dodonova, Y., Korolev S., Tkachev A., Petrov, D., Zhukov, L., Belyaev, M.: Classification of structural brain networks based on information divergence of graph spectra. Machine Learning for Signal Processing Proceedings (2016, accepted).

[3] Dodonova, Y., Belyaev, M., Tkachev, A., Petrov, D., Zhukov, L. : Kernel classification of connectomes based on earth mover's distance between graph spectra. Proceedings of the Workshop on Brain Analysis using Connectivity Networks (2016, accepted)

[4] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, Renaud Lefebvre : Fast unfolding of communities in large networks, Journal of Statistical Mechanics: Theory and Experiment 2008(10), P10008 (12pp)

[5] M. E. J. Newman : Finding community structure in networks using the eigenvectors of matrices, Phys. Rev. E 74, 036104 (2006).

[6] A. Clauset, M. E. J. Newman and C. Moore: Finding community structure in very large networks. Phys Rev E 70, 066111 (2004).

[7] Raghavan, U.N. and Albert, R. and Kumara, S. Near linear time algorithm to detect community structures in large-scale networks. Phys Rev E 76:036106, 2007

[8] Chan, A.B., Vasconcelos, N., Moreno, P.J. (2004) A family of probabilistic kernels based on information divergence. Univ. California, San Diego, CA, Tech. Rep. SVCL-TR-2004-1.

[9] Brown, J. A., Rudie, J. D., Bandrowski, A., Van Horn, J. D., Bookheimer, S. Y. (2012). The UCLA multimodal connectivity database: a web-based platform for brain connectivity matrix sharing and analysis. Frontiers in neuroinformatics, 6, 28.

[10] Power, J.D., Cohen, A.L., Nelson, S.M., Wig, G.S., Barnes, K.A., Church, J.A., Vogel, A.C., Laumann, T.O., Miezin, F.M., Schlaggar, B.L., Petersen, S.E. (2011) Functional network organization of the human brain. Neuron 72, 665–678.