Консолидация различных вариантов сетевых структур мозга при решении задачи классификации нормы и патологии

Сафиуллин А. Э.
Национальный исследовательский университет «Высшая школа экономики»
aesafiullin@edu.hse.ru

Беляев М. Г.
ИППИ РАН им. А.А. Харкевича
belyaevmichel@gmail.com

Додонова Ю. А.
Национальный исследовательский университет «Высшая школа экономики»
ya.dodonova@mail.ru

Аннотация

В данной работе решается задача классификации людей с расстройством аутистического спектра и нормальным развитием на основе сетевых структур головного мозга (коннектомов). Мы сравниваем различные подходы к построению алгоритмов машинного обучения в условиях, когда каждый объект представлен не одним коннектомом, а набором таких сетей, возникающим при применении к исходным данным различных способов взвешивания и нормировки. Мы рассматриваем алгоритмы двухуровневой классификации: стекинга и блендинга моделей, обученных на коннектомах с отдельными видами взвешивания и нормировки. На объединении различных вариантов коннектомов мы также строим discriminative fusion классификатор - логистическую регрессию на взвешенной комбинации данных. Наилучшее качество классификации (ROC AUC порядка 0.8) было получено для блендинга - взвешенной комбинации моделей логистических регрессий; построенная двухуровневая модель превосходит по качеству классификации индивидуальные модели первого уровня.

1. Введение

Современные неинвазивные технологии визуализации головного мозга, такие как диффузионно-взвешенная магнитно-резонансная томография (дМРТ), позволяют на макро-уровне проследить пучки волокон белого вещества, обеспечивая основу для построения карты структурных связей между отдельными регионами головного мозга. Представление структуры связей между зонами мозга в виде графа становится популярным инструментом исследований в нейронауках. В таком обобщенном графе мозга (коннектоме) вершины представляют собой области мозга, а ребра - структурные связи между регионами. При этом такие сети отличаются в топологии от случайных сетей [1].

С прикладной точки зрения актуальной является задача классификации таких объектов, в частности, различения нормального развития и патологии на основе структурных сетей мозга. Решение этой задачи осложняется многими особенностями, характерными для данных из области нейронаук. В частности, для данной предметной области типичны очень маленькие объемы выборок (несколько десятков объектов), что в сочетании с высокой размерностью данных существенно затрудняет построение алгоритмов машинного обучения (недавний обзор исследований с использованием машинного обучения в нейронауках может быть найден, например, в работе [2]).

В задаче классификации коннектомов существенным является также тот факт, что процедура порождения коннектома на основе данных нейровизуализации не является однозначной. Различные варианты коннектома могут возникать на основе одного и того же снимка дМРТ в результате выбора различных алгоритмов трактографии для реконструкции связей между зонами мозга, в результате различных процедур взвешивания и нормализации ребер коннектомов, или многих других выборов, которые может делать исследователь при построении

макро-уровневого графа, описывающего структуру мозга.

При решении задачи классификации это означает, что алгоритмы машинного обучения потенциально могут использовать информацию не от одной, а от нескольких представлений каждого из объектов (коннектомов). Эта идея, в частности, обсуждается в работе [4]. В статье предложена идея Discriminative Fusion (dFuse) - логистической регрессии, которая решает задачу классификации выпуклой комбинации различных данных (с поиском оптимальных коэффициентов выпуклой комбинации, минимизирующих логарифмическую функцию потерь). Авторы решали задачу классификации нормы и болезни Альцгеймера, комбинируя наборы коннектомов, полученные на основании одних и тех же снимков дМРТ с использованием различных алгоритмов трактографии. Такой подход показал существенное улучшение качества классификации именно из-за комбинации различных данных (ROC AUC 0.89±0.09 на комбинации данных по сравнению с лучшим результатом на отдельных данных 0.66±0.10).

В данной работе мы сравниваем три возможных подхода к решению задачи классификации в условиях, когда каждый из объектов представлен набором коннектомов. Два из них представляют собой алгоритмы двухуровневой классификации: стекинг и блендинг моделей, обученных на отдельных наборах данных. Третий подход - описанный в [4] discriminative fusion классификатор - логистическая регрессия на взвешенной комбинации данных. Мы сравниваем эффективность работы этих алгоритмов для одного и того же набора данных при решении задачи классификации нормы и расстройств аутистического спектра, порождая параллельные наборы данных использованием различных способов взвешивания и нормировки коннектомов.

2. Постановка задачи

Пусть $\widetilde{\mathscr{X}}$ - множество коннектомов (симметричных взвешенных матриц смежности), $\mathscr{Y}$ - конечное множество меток классов. Существует некоторая зависимость $\widetilde{f}: \widetilde{\mathscr{X}} \to \mathscr{Y}$ значения которой известны только на объектах конечной обучающей выборки $\{\widetilde{X}, Y\} = \{(\widetilde{x_1}, y_1), .., (\widetilde{x_n}, y_n)\}$. Однако процедура построения коннектома не является однозначной: это означает, что можно построить отображение $f': \widetilde{\mathscr{X}} \to \mathscr{X}^k$, где $\mathscr{X}^k$ - множество различных вариантов одного и того же коннектома $\widetilde{\mathscr{X}}$. В данной работе различные варианты коннектома возникают как результат применения нескольких схем взвешивания и нормировки исходных данных; в более общем виде речь идет о возможном применении нескольких различных алгоритмов на любом этапе построения коннектома.

Тогда есть зависимость $f: \mathscr{X}^k \to \mathscr{Y}$ для которой можно построить обучающую выборку $\{X_1, .., X_k, Y\} = \{(x_{11}, .., x_{1k}, y_1), (x_{21}, .., x_{2k}, y_2), .., (x_{n1}, .., x_{nk}, y_n)\}$, где $X_1, .., X_k$ - различные вариации одного коннектома, $x_{ij} \in \widetilde{\mathscr{X}} (i \in [1,k], j \in [1,n])$. Требуется построить алгоритм $\alpha: \mathscr{X}^k \to \mathscr{Y}$ способный классифицировать произвольный объект $x \in \mathscr{X}^k$.

3. Подходы к консолидации различных вариантов коннектомов при решении задачи классификации

В данной статье сравнивается три подхода к решению задачи классификации коннектомов, представленных набором различных вариантов (в данном случае - взвешиваний и нормировок):

1. Стекинг отдельных моделей логрегрессий моделью логистической регрессии на втором уровне
2. Блендинг - выпуклая линейная комбинация предсказаний отдельных моделей логрегрессий
3. Логистическая регрессия взвешенных данных (Discriminative Fusion классификатор).

Стекинг моделей представляет из себя двухуровневую модель. Первый уровень - модели логистических регрессий, каждая из которых обучена на отдельном варианте коннектомов, второй уровень - модель логистической регрессии, которая обучается на предсказаниях моделей перового уровня. Обучение стекинга происходит следущим образом: на втором уровне модель логистической регрессии обучается на предсказаниях моделей первого уровня по кросс-валидации (т.е. разбиваем данные на несколько частей, обучаемся на части, предсказываем на оставшейся части и так для всех частей, после чего объединяем все предсказания и обучаем на объединении модель логистической регрессии второго уровня). На первом уровне стекинга каждая отдельная модель логистической регрессии обучается на всем наборе соответствующих ей данных. После чего для предсказания результатов на новых данных достаточно объединить векторы предсказанных вероятностей, полученных моделями логистических регрессий первого уровня, и использовать полученные новые признаки для классификации моделью логистической регрессии второго уровня.

Приведем алгоритм обучения данной модели (наблюдения X - тензор N x K x D, где N - число наблюдений, K - число различных вариантов исходного коннектома, D - число вершин в коннектоме/число ребер) на языке python (с использованием методов из библиотеки sklearn для краткости изложения):

Алгоритм 1 Стекинг логистических регрессий

Вход: → X (тензор N x K x D), Y (вектор размерностью N), L - число разбиений
Выход: → обученная модель
Начало
$skf = KFold(L)$ #разбиваем данные на L частей
инициализировать массив (заполненный нулями) $|ans| = NxK$
инициализировать массив моделей лог.регрессий $|logr| = K$
for *train*, *test* **in** skf **do**
    for от $i = 1$ **до** K **do**
        $logr[i].fit(X[train,i,:],Y[train])$
#обучим лог.регрессию
        $ans[test,i] = logr[i].predict\_proba(X[test,i,:])$
#объединим предсказания
        $logr[i].fit(X[:,i,:],Y[train])$
#обучим лог.регрессию на всех данных
$logregr\_new.fit(ans,Y)$ #модель второго уровня
Конец.

Предсказание для X_{eval} происходит так:
инициализация нового массива *ans* размером *NxK* нулями
for от $i = 1$ **до** K **do**
    $ans[:,i] = logr[i].predict\_proba(X_{eval}[:,i,:])$
$logregr\_new.predict(ans)$ #ответ классификатора

Блендинг аналогичен стекингу, за исключением того, что на втором уровне вместо предсказаний логистической регрессии считается взвешенная сумма предсказаний (вероятность принадлежности классу 1) моделей первого уровня. Для нахождения этих весов решается задача поиска коэффициентов α выпуклой комбинации предсказаний вероятностей моделей первого уровня, минимизирующих функцию потерь кросс-энтропии:

$$logloss(x,y) =$$

$$= \sum_i y_i \cdot \log(\sum_j x_{i,j} \cdot \alpha_j) + (1-y_i) \cdot \log(1 - \sum_j x_{i,j} \cdot \alpha_j), \quad (1)$$

где $\sum_j \alpha_j = 1; \alpha_j \geq 0, \forall \alpha_j$, y_i - истинное значение класса ($y_i \in \{0,1\}$), $x_{i,j}$ - предсказанная j-ым классификатором вероятность принадлежности классу 1 i-го примера (см. Алгоритм 2).

Discriminative Fusion. Другой подход заключается не во взвешивании ответов классификаторов, а в использовании одним классификатором взвешенных данных. В статье [4] предложена идея Discriminative Fusion (dFuse) - логистической регрессии, которая решает задачу классификации выпуклой комбинации различных данных (с поиском коэффициентов выпуклой комбинации, минимизирующих логарифмическую функцию потерь). Авторы решали аналогичную поставленной нами задачу классификации нормы и аутизма,комбинируя в рамках одного алгоритма анализа коннектомы, полученные в результате использования различных алгоритмов трактографии. Такой подход показал существенное улучшение качества классификации именно из-за комбинации различных данных. В данной работе использован этот классификатор (однако, слегка видоизмененный: в оптимизационной задаче отсутствуют ограничения на коэффициенты и поэтому изменен вид штрафа) для наборов коннектомов, полученных с применением различных схем предварительного взвешивания и нормировки.

Алгоритм 2 Блендинг логистических регрессий

Вход: → X (тензор N x K x D), Y (вектор размерностью N), L - число разбиений
Выход: → обученная модель
Начало
$skf = KFold(L)$ #разбиваем данные на L частей (метод из библиотеки *sklearn*)
инициализировать массив (заполненный нулями) $|ans| = NxK$
инициализировать массив моделей лог.регрессий $|logr| = K$
for *train*, *test* **in** skf **do**
    for от $i = 1$ **до** K **do**
        $logr[i].fit(X[train,i,:],Y[train])$
#обучим лог.регрессию
        $ans[test,i] = logr[i].predict\_proba(X[test,i,:])$
#запишем предсказание
        $logr[i].fit(X[:,i,:],Y[train])$
#обучим лог.регрессию на всех данных
$\alpha = \arg\min_\alpha logloss(ans,Y)$ # уравнение (1)
Конец.

Предсказание для X_{eval} происходит так:
инициализация нового массива *ans* размером *NxK* нулями
for от $i = 1$ **до** K **do**
    $ans[:,i] = logr[i].predict\_proba(X_{eval}[:,i,:])$
$\sum_{i=1}^{K} \alpha_i * ans[:,i]$ #ответ классификатора

Постановка задачи следующая. Пусть N - число пациентов, K - число различных нормировок и взвешиваний коннектомов для каждого пациента. Пусть $x_i^{(k)}$ - векторное представление k-той нормировки и взвешивания графа для пациента i, y_i – значение диагноза для каждого i-того пациента ($y_i \in \{-1,1\}, y = [y_1,..,y_N]$).

Тогда нормировки-взвешивания для каждого пациента можно объединить в одну выпуклой комбинацией:

$$x_i(\tau) = \sum_{k=1}^{K} \tau_m \cdot x_i^{(k)}, \quad (2)$$

где τ_k – вектор коэффициентов ($\tau = [\tau_1, .., \tau_K]$).

Такая комбинация позволяет дать интерпретацию вклада каждой нормировки-взвешивания в результативный «коннектом». Далее решается следующая задача оптимизации:

$$min_{\omega,c,\tau} \sum_{i=1}^{N} l(\tau, \omega, c, x_i, y_i) + \lambda \cdot (||\omega|| \cdot ||\tau|| + |c|), \quad (3)$$

где λ - коэффициент регуляризации, τ - коэффициенты комбинации различных нормировок-взвешиваний, ω, c - параметры модели логистической регрессии, в качестве нормы используется L_2 норма, $\sum_{i=1}^{N} l(\tau, \omega, c, x_i, y_i)$ логарифмическая функция потерь:

$$l(\tau, \omega, c, x_i, y_i) = log(1 + exp(-y_i \cdot (x_i(\tau)^T \cdot \omega + c))). \quad (4)$$

Как отмечалось выше, мы предлагаем модификацию алгоритма, описанного в [4]: в оптимизационной задаче отсутствуют ограничения на коэффициенты и поэтому изменен вид штрафа. В силу выпуклости функции потерь данная оптимизационная задача может быть решена методом блочно-координатного спуска (см. Алгоритм 3).

Алгоритм 3 Discriminative fusion classifier

1: Вход: → X (тензор N x K x D), Y (|Y| = N), λ
2: Выход: → коэффициенты модели (w, c, τ)
3: Начало
4: Случайно инициализировать веса τ_{new}, τ_{old}, w_{new}, w_{old}
5: **while** $||\tau_{new} - \tau_{old}|| < \varepsilon$ and $||w_{new} - w_{old}|| < \varepsilon$ **do**
6: $x_w = X \otimes_3 w_{new}$
// произведение по третьей компоненте (x_w тензор N x D)
7: $\tau_{new} = \arg\min_\tau \{\log(1 + \exp(-y \cdot (x_w \cdot \tau))) + + \frac{\lambda}{||w_{new}||} \cdot ||\tau||\}$ //(обычная лог.регрессия)
8: $x_\tau = X \otimes_2 \tau_{new}$
// произведение по второй компоненте (x_w тензор N x K)
9: $w_{new}, c = = \arg\min_{w,c} \{log(1 + \exp(-y \cdot (x_\tau \cdot w + c))) + + \frac{\lambda}{||\tau_{new}||} \cdot (||w|| + ||c||)\}$ //(обычная лог.регрессия)
10: $c = c * ||\tau_{new}||$
11: Конец. (Выход: w_{new}, c, τ_{new})

Нормировка коэффициента регуляризации λ на $||\tau_{new}||$ и $||w_{new}||$ вводится для решения задачи (3) методом координатно-блочного спуска. Это позволяет проводить оптимизацию, учитывая в функции потерь регуляризацию совместного произведения коэффициентов τ и w.

Классификация тестовых данных производится следующим образом: считается выпуклая комбинация нормировок-взвешиваний на тестовых данных по формуле (2) с оптимально подобранными весами, затем применяется логистическая регрессия:

$$\hat{y}_i = sign(x_i(\tau)^T \cdot \omega + c). \quad (5)$$

Таким образом, данная постановка задачи dFuse классификатора сводится к решению задач поиска коэффициентов в задаче оптимизации модели логистической регрессии.

4. Данные

В данном разделе мы описываем исходный набор коннектомов, с которым мы работали, процедуру получения различных вариантов коннектомов на основе исходного набора, а также процедуру построения векторов признаков для классификаторов.

4.1. Набор коннектомов

Для данной задачи взят набор данных UCLA Autism [5] (получение данных описывается в статье [6]). Данные представляют собой коннектомы - симметричные взвешенные матрицы смежности между 264 зонами головного мозга, где веса пропорциональны количеству выявленных трактов между соответствующими зонами. Так же данные содержат координаты центров этих зон. Данные представлены по 94 пациентам: 51 человек с патологией (расстройства аутистического спектра), 43 пациента без выявленных отклонений (типичное развитие); для каждого пациента представлен один коннектом.

4.2. Взвешивание и нормировка исходных данных

Для каждого коннектома из исходного набора мы строим несколько его вариантов, применяя различные методы взвешивания и нормировки существующих ребер. Необходимо подчеркнуть, что применяемые процедуры взвешивания и нормировки затрагивают исключительно ненулевые ребра исходного графа, сохраняя при этом нулевые элементы исходных матриц смежности.

Под взвешиванием мы понимаем процедуру присвоения весов ребрам исходных коннектомов с учетом информации об их физических свойствах: «толщине» ребра (то есть количеству трактов, выявленных между соответствующими регионами мозга) или его «длине» (расстоянию между соответствующими регионами мозга). Под нормировкой мы понимаем процедуру модификации весов ребер на основе информации о топологических особенностях сети, в частности - о степенях вершин.

В [3] описаны отдельные виды взвешивания и нормировок структурных коннектомов в применении к этому набору данных. Мы используем описанные в этой работе алгоритмы препроцессинга данных, а также предлагаем несколько дополнительных алгоритмов. Для каждого типа взвешивания и нормировки в приведенном ниже описании в скобках указано сокращенное название, которое будет использоваться при представлении результатов:

1. Исходные веса без изменений (origw);
2. Бинаризация - все ненулевые ребра в исходных матрицах получают значение 1 (binar)

$$a_{ij}^{new} = 1 \ if \ a_{ij} > 0, else \ 0. \tag{6}$$

3. Корень из исходных весов (sqrtw)

$$a_{ij}^{new} = \sqrt{a_{i,j}}, \tag{7}$$

где a_{ij} - исходные веса ребер между зонами i и j;

4. Обратные значения расстояний между вершинами (invdist)

$$a_{ij}^{new} = \frac{1 \ if \ a_{ij} > 0 \ else \ 0}{l_{ij}}, \tag{8}$$

где $l_{i,j}$ - евклидовы расстояния между центрами соответствующих зон мозга (использовались координаты центров зон в стандартном MNI атласе, предоставленные авторами набора данных);

5. Корень из исходных весов деленный на расстояние между вершинами (rootwbydist)

$$a_{ij}^{new} = \frac{\sqrt{a_{ij}}}{l_{i,j}}. \tag{9}$$

6. Исходные веса, деленные на квадрат расстояния между вершинами (wbysqdist)

$$a_{ij}^{new} = \frac{a_{ij}}{l_{i,j}^2}. \tag{10}$$

Также для каждого набора взвешенных данных мы используем следующие нормировки:

1. Ненормированные данные - коннектомы оставлялись без изменений (nonnormed)
2. Нормировка на максимальное значение - веса делятся на максимальное значение в матрице (nonnormbymax)

$$a_{ij}^{new} = \frac{a_{ij}}{\max_{i,j}\{a_{i,j}\}}. \tag{11}$$

3. N1-нормировка: значения делятся на сумму всех весов в матрице (n1normed)

$$a_{ij}^{new} = \frac{a_{ij}}{\sum_{i,j} a_{i,j}}. \tag{12}$$

4. N2-нормировка: веса ребер в каждой матрице делятся на геометрическое среднее соответствующих смежных вершин (n2normed)

$$a_{ij}^{new} = \frac{a_{ij}}{\sqrt{\sum_j a_{i,j} \cdot \sum_i a_{i,j}}}. \tag{13}$$

5. N2-нормировка, затем полученные значения делятся на максимальное значение (n2bymax)

$$a_{ij}^{new} = \frac{a_{ij}}{\sqrt{\sum_j a_{i,j} \cdot \sum_i a_{i,j}} \cdot \max_{i,j}\{a_{i,j}\}}. \tag{14}$$

Таким образом, различные комбинации алгоритмов взвешивания и нормировки структурных коннектомов для каждого исходного коннектома порождают 30 вариантов коннектома. В данном исследовании мы, таким образом, работаем с 30 наборами коннектомов.

4.3. Построение признаков для классификатора

При решении задачи классификации нормы и расстройств аутистического спектра использовались следующие вектора признаков, построенные на основе коннектомов:

1. развернутые в вектор верхние треугольные части матриц смежности взвешенно-нормированных коннектомов (так называемый «мешок ребер»);
2. взвешенные степени вершин (weighted node degrees), представляющие собой для каждой вершины сумму весов ее смежных ребер;
3. взвешенные степени вершин коннектомов, предварительно приведенных к одинаковому уровню плотности 7% (от общего возможного числа ребер): в каждом коннектоме было оставлено 7% наиболее сильных связей, оставшиеся связи считались нулевыми.

Процедура приведения коннектомов к одинаковому уровню плотности традиционно применяется, в частности, чтобы исключить из анализа слабые ребра, являющиеся зачастую артефактом алгоритмов трактографии. В нашей работе мы использовали аналогичную процедуру и для предварительно взвешенных и нормированных коннектомов. Необходимо отметить также, что эта процедура в нашем случае вносит дополнительную вариативность в

данные: взвешенно-нормированные коннектомы, построенные на основании одного и того же исходного коннектома, после данной процедуры могут содержать ненулевые ребра в разных позициях.

4.4. Предыдущие результаты

Авторы этого набора данных не используют алгоритмы машинного обучения. Наилучший известный нам результат классификации, полученный для этого набора данных, приводится в [3]. В этой работе используется два вида нормировок геометрическая и топологическая. В лучшей модели (SVM с линейным ядром и elastic-net регуляризацией) на взвешенных степенях вершин было получено качество ROC AUC на кросс-валидации 0.77 ± 0.03.

5. Процедура

Так как линейные классификаторы чувствительны к масштабу данных, для каждого из описанных выше наборов признаков было использовано так называемое «max-min scaling» преобразование: $x^{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$.

При оценивании качества моделей использовалась процедура 10-fold кросс-валидации, т.е. бралось равномерное (относительно класса) разбиение данных на 10 частей, обучение производилось на 9 частях, а предсказание на оставшейся одной. Предсказания для каждой из тестовых частей объединялись, чтобы построить предсказание для всей выборки. Оценка качества предсказания строилась, таким образом, на основе объединенного предсказания всех тестовых подвыборок. В качестве метрики оценки качества алгоритма классификации использовалась метрика ROC AUC.

Оптимальные параметры моделей находились поиском по сетке на одном разбиении на 10 кросс-валидационных тестовых выборок. После этого тестирование моделей с выбранными фиксированными параметрами проводилось по 50 различным случайным разбиениям на 10 кросс-валидационных объединениях предсказаний. Таким образом, для каждой из моделей мы получаем 50 значений ROC AUC и оцениваем качество классификации на основе среднего значения ROC AUC и его стандартного отклонения.

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

6. Результаты

Для классификации различных взвешиваний и нормировок коннектомов были взята обычная модель логистической регрессии, с параметрами, подобранными на одном разбиении (Рис. 1, 2, 3)

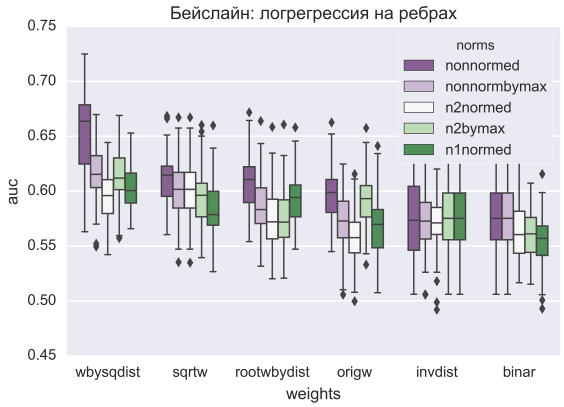

Рис. 1. Результаты логистической регрессии на весах ребер различных взвешенно-нормированных коннектомов

На «мешке ребер» отдельный лучший результат ROC AUC 0.65 ± 0.03 достигнут на взвешивании (9) и нормировке (12), на степенях вершин коннектомов отдельный лучший результат - 0.74 ± 0.03, а на степенях вершин с частично обнуленными ребрами - 0.75 ± 0.03, при этом на двух последних представлениях данных лучшие результаты получаются на взвешивании (10) (а лучшие результаты на этом взвешивании на нормировке (14)). На рис. 4 представлены результаты ROC AUC логистических регрессий для степеней вершин коннектомов и степеней вершин коннектомов с частично обнуленными ребрами. Видно, что взвешивание (10) практически устойчиво к обнулению ребер (в пределах погрешности).

Стекинг логистических регрессий показал себя

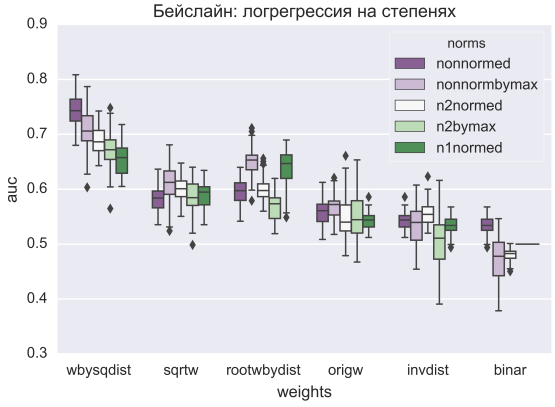

Рис. 2. Результаты логистической регрессии на взвешенных степенях вершин взвешенно-нормированных коннектомов

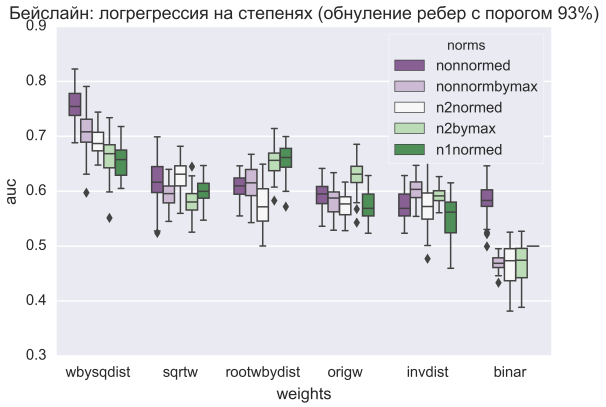

Рис. 3. Результаты логистической регрессии на взвешенных степенях вершин взвешенно-нормированных коннектомов (обнуленные 93% весов ребер с наименьшими значениями)

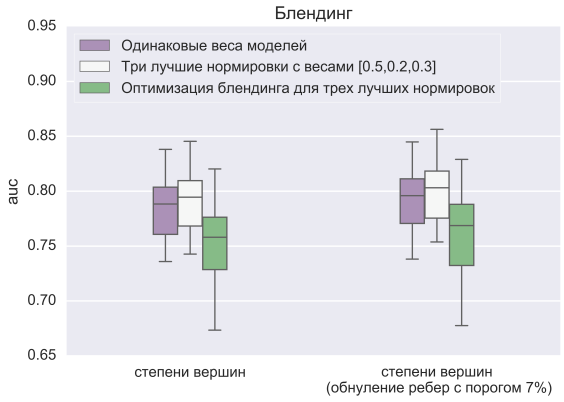

Рис. 4. Результаты блендинга

хуже, чем логистические регрессии индивидуально - лучший результат на стекинге при объединении трех моделей (взвешивание (10) и нормировки (14), (11), (13)) для степеней вершин равен 0.70 ± 0.03. На степенях вершин с частично обнуленными ребрами стекинг дает качество немного хуже, по сравнению со стекингом логистических регрессий на взвешенных степенях. Также стекинг склонен к переобучению и занижению качества классификации при использовании всех нормировок - взвешиваний (ROC AUC падает до 0.5).

На рис. 4 представлены результаты **блендинга** логрегрессий на трех индивидуально лучших нормировках-взвешиваниях (взвешивание (10) и нормировки (14), (11), (13) с весами соответственно 0.5, 0.2, 0.3), на которых видно, что блендинг на взвешенных степенях вершин, рассчитанных на коннектомах с обнуленной частью ребер с наименьшими весами, дает небольшое улучшение качества классификации, по сравнению со взвешенными степенями, рассчитанными на исходных ребрах (с 0.79 ± 0.03 до 0.80 ± 0.03). При этом видно, что оптимизация весов блендинга на втором уровне с минимизацией фукнции потерь кросс-энтропии дает значительное снижение качества классификации.

Результаты **dFuse классификатора** оставляют желать лучшего. При его обучении возникает эффект переобучения из-за небольшого числа данных большой размерности (94 объекта размерностью 30x264 в представлении данных в виде взвешенных степеней вершин коннектомов) и качество классификации по метрике ROC AUC находится около 0.70 ± 0.03.

7. Заключение

На данных UCLA (autism) [5] были опробованы двухуровневые модели классификации коннектомов с различными вариантами предварительной нормировки и взвешивания: стекинг логистических регрессий, блендинг логистических регрессий, dFuse-классификатор.

При сравнении работы одноуровневых моделей на отдельных видах взвешивания и нормировки были воспроизведены результаты работы [3].

В этой работе мы построили двухуровневую модель на основе выпуклой комбинации трех лучших моделей на отдельных видах нормировки и взвешивания (взвешивание (10) и нормировки (14), (11), (13)). Построенная модель дает лучшее качество классификации по метрике ROC AUC на взвешенных степенях вершин коннектомов по сравнению с лучшим результатом для одноуровневой модели (0.79 ± 0.03 против 0.77 ± 0.03). Дополнительная процедура приведения всех коннектомов к одинаковому уровню плотности (обнуления 93% ребер с наименьшим весом) позволила получить на тех же признаках ROC AUC 0.80 ± 0.03.

Важно, что мы получили наилучший результат двухуровневых моделей, взяв три лучшие модели на первом уровне и подобрав комбинацию их весов перебором. При этом стекинг и блендинг с оптимизацией коэффициентов переобучаются на втором уровне из-за маленького числа примеров и большой размерности данных. Результат dFuse классификатора также является низким из-за эффекта переобучения.

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] Duarte-Carvajalino J. M. et al. *Hierarchical topological network analysis of anatomical human brain connectivity and differences related to sex and kinship.* Neuroimage, 59 (4), 3784–3804 (2012)

[2] Arbabshirani M. R. et al. *Single subject prediction of brain disorders in neuroimaging: Promises and pitfalls.* NeuroImage (2016)

[3] Petrov D., Dodonova Y., Zhukov L., Belyaev M. *Boosting connectome classification via combination of geometric and topological normalizations*, Proc. Pattern Recognition in Neuroimaging (2016)

[4] Qi Wang et al. *Discriminative Fusion of Multiple Brain Networks for Early Mild Cognitive Impairment Detection.* Proc. IEEE International Symposium on Biomedical Imaging, 568–572 (2016)

[5] *UCLA autism datasets*, набор данных, доступен по ссылке: `http://umcd.humanconnectomeproject.org/umcd/default/update/1946`

[6] Brown J. A. et al. *The UCLA multimodal connectivity database: a web-based platform for brain connectivity matrix sharing and analysis.* Frontiers in neuroinformatics, 6, 28 (2012)