

# 1 Аннотация

Строится модель классификации, которая предсказывает, произойдет ли землетрясение с магнитудой выше заданного порога в конкретной локации во временной промежутке от 30 до 180 дней от фиксированного момента времени. Наиболее популярный подход заключается в том, чтобы использовать экспертные прогнозы, основанные на признаках таких, как Region—Time—Length (RTL: Регион—Время—Длина) характеристики. Предложенный подход использует модели машинного обучения, которые используют несколько RTL признаков как вход, чтобы учесть эффекты различных шкал и увеличить точность предсказаний. В качестве исторических данных используются данные о землетрясениях в Японии в период с 1992 по 2005 год. Точность лучшей модели  $\sim 0.95$ , а полнота  $\sim 0.98$ .

# Содержание

<b>1</b>	<b>Аннотация</b>	<b>2</b>
<b>2</b>	<b>Введение</b>	<b>4</b>
2.1	Постановка задачи . . . . .	4
2.2	Математическая постановка задачи . . . . .	4
2.3	Актуальность проблемы . . . . .	5
2.4	Методы решения задачи . . . . .	7
<b>3</b>	<b>Обзор литературы</b>	<b>10</b>
3.1	Различные типы признаков . . . . .	10
3.2	Различные методы обучения . . . . .	11
3.3	Полученные результаты . . . . .	13
3.4	Выводы . . . . .	13
<b>4</b>	<b>Методы</b>	<b>15</b>
4.1	Данные . . . . .	15
4.2	RTL признаки . . . . .	16
4.2.1	Нормализация RTL признаков . . . . .	19
4.3	Классификаторы . . . . .	20
4.4	Несбалансированная задача классификации . . . . .	21
4.5	Скольльзящий контроль . . . . .	24
4.6	Метрики . . . . .	24
<b>5</b>	<b>Эксперименты</b>	<b>27</b>
5.1	Выбор параметров для RTL . . . . .	27
5.2	Агрегация нескольких RTL . . . . .	28
5.3	Использование ресемплинга . . . . .	29
<b>6</b>	<b>Выводы</b>	<b>30</b>

## 2 Введение

### 2.1 Постановка задачи

Землетрясения — одно из самых разрушительных стихийных бедствий. В период между 1900 и 2015 годами причиной около 40% экономического ущерба были наводнения, около 25% — землетрясения, около 20% — штормы, 12% — засухи, 2% — лесные пожары и менее 1% — извержения вулканов.

На данный момент времени не существует достаточно точного способа предсказывать землетрясения из-за сильно нелинейного поведения сейсмичности. Сейчас для решения задачи прогнозирования землетрясений используют физические методы: эмпирические сейсмологические законы, такие как закон Гутенберга — Рихтера и закон Омори—Утсу, методы машинного обучения: ресэмплирование, классификаторы, нейронные сети, которые описаны в разделе 4. Однако, они работают недостаточно точно. В данной работе предлагается использовать методы машинного обучения для задачи классификации землетрясений.

Если магнитуа землетрясения больше порога в конкретное время в конкретном месте, то необходимо бить тревогу. В данной работе рассматриваются среднесрочные прогнозы: нужно предсказывать землетрясения в пределах от 30 до 180 дней. Мы работаем с сильно несбалансированной выборкой и используем методы машинного обучения с учетом этих особенностей: комбинация multiscale (нескольких) RTL признаков и балансировки классификаторов и данных.

### 2.2 Математическая постановка задачи

В используемом наборе данных у каждого землетрясения четыре параметра: пара координат  $(x, y)$ , время  $t$  и магнитуа  $M$ .

Определим через  $c(x, y, t)$  индикаторную функцию, которая определяет, что землетрясение  $e$  произошло в некоторой локации  $(x, y, t)$ . Значение  $c(x, y, t) = 1$ , если как минимум одно землетрясение магнитудой  $M_e \geq M_c$

(для некоторого порогового значения  $M_c$ ) с пространственно-временными координатами  $(x_e, y_e, t_e)$  удовлетворяет следующим ограничениям:

$$\|(x, y) - (x_e, y_e)\|_2 \leq R_c, \quad \delta_c < t_e - t < T_c, \quad (1)$$

где  $R_c$  — радиус на плоскости, для которого нужно предсказать землетрясение,  $\delta_c$  и  $T_c$  — соответственно начало и конец временного промежутка, внутри которого нужно предсказать землетрясение.

Иначе значение  $c(x, y, t) = 0$ .

Наша цель состоит в том, чтобы построить модель, которая предсказывает, произойдет ли землетрясение во временном цилиндре  $[T + \delta_c, T + T_c]$  для некоторого  $T$ , используя историческую информацию о землетрясениях, произошедших перед временем  $T$ .

Мы собираем информацию о землетрясениях, произошедших до момента времени  $T$ , в вектор признаков  $\mathbf{X}$  фиксированной длины. В частности мы генерируем RTL признаки, используя процедуру, описанную в разделе 4.2.

Окончательно для всех землетрясений, которые зафиксированы в используемых данных, мы получаем пары  $(\mathbf{X}_i, c_i)$ , где  $c_i$  определено с помощью ограничений (1) с пороговым значением для магнитуды, равным  $M_c = 5$ . Все такие пары формируют выборку  $D = \{(\mathbf{X}_i, c_i)\}_{i=1}^n$  размера  $n$ . Наша цель — разработать модель  $\hat{c}(x, y, t)$ , которая будет приближать с хорошей точностью реальность  $c(x, y, t)$ .

В работе мы рассматриваем исторические данные о землетрясениях в Японии в период с 1992 по 2005 года.

## 2.3 Актуальность проблемы

Ущерб от землетрясений рассчитывается на основе количества жертв и страховых выплат. Наибольшее разрушающее воздействие землетрясения оказывают на города и крупные технические объекты. Чем более густонаселен район землетрясения, тем больше в районе поражения высотных зданий, газо- и нефтепроводов, химических предприятий, складов опасных

химических веществ, плотин и дамб, тем выше число человеческих жертв и экономический ущерб. Землетрясения могут сопровождаться и техногенными катастрофами, возникающими в результате разрушения складов опасных химических веществ, разрывами газо- и продуктопроводов, разрушением транспортных магистралей и сооружений. Недавние крупные землетрясения:

1. Землетрясение на Гаити (12.01.2010) магнитудой 7.0. Его гипоцентр находился в море, в 25 км к юго-западу от столицы — города Порт-о-Пренса, на глубине 13 км. Погибли 316 тысяч человек, более 300 тысяч человек были ранены, 1.3 миллиона человек остались без жилья. Экономический ущерб составил \$7.9 млрд.
2. Землетрясение в китайской провинции Сычуань (12.05.2008) магнитудой 7.9. Эпицентр находился в уезде Вэньчуань в 80 км к северо-западу от столицы провинции города Чэнду. Землетрясение было настолько мощным, что в Пекине и Шанхае на расстоянии больше полутора тысяч километров ощущались толчки. Погибли 87.6 тысяч человек, более 370 тысяч были ранены. 15 миллионов человек были эвакуированы, более 5 миллионов лишились жилья. 5.36 миллионов зданий оказались полностью разрушенными, более 21 миллионов поврежденными. Экономический ущерб оценивается в \$86 млрд.
3. Землетрясение в Индийском океане (26.12.2004). Ученые оценивают магнитуду в пределах от 9.1 до 9.3. Гипоцентр находился в 160 километрах к западу от острова Суматра на глубине 30 километров. Предположительно, причиной являлся сдвиг тектонических плит, который привел к образованию цунами, достигшего Таиланда, Индонезии, Шри-Ланки, юга Индии и восточного побережья Африки. Количество погибших оценивается в пределах от 225 до 300 тысяч человек в 14 странах, раненых — около 2.2 миллионов. Землетрясение и цунами вызвали многочисленные разрушения, экономический ущерб Таиланда оценивается в \$5 миллиардов, Индии — в \$1.6 миллиардов,

Мальдивских островов — в \$1.3 миллиардов, Индонезии — в \$45 миллиардов, острова Суматра — в 675 миллионов.

Только за прошедший год произошло 20 землетрясений магнитудой больше 5. Число погибших в них — 200 человек, а пострадавших порядка 2000. Своевременное предсказание землетрясений может привести к нулевым человеческим потерям и к значительному снижению финансовых издержек.

## 2.4 Методы решения задачи

Есть несколько способов постановки задачи предсказания землетрясений:

- рассмотрение целевого региона и предсказание места и время следующего землетрясения;
- разбиение местности на сетку и предсказание землетрясения для каждой клетки;
- рассмотрение каждого землетрясения в качестве отдельного события с заданной локацией и временем и предсказание его магнитуды [1].

В данной работе мы рассматриваем третий подход к решению задачи и строим модель, предсказывающую, превосходит ли магнитуда заданный порог для землетрясения для фиксированной локации и времени.

Отправная точка построения модели для предсказания землетрясений — эмпирическое соотношение или физическая модель, которая соотносится с представлениями о реальности. Часто это представление недостаточно точное и требует применения подходов машинного обучения плюсом к описанию, основанному на физических моделях.

Для предсказаний землетрясений часто используются несколько эмпирических статистических соотношений, например закон Гутенберга — Рихтера и закон Омори — Утсу.

Закон Гутенберга — Рихтера [2] выражает соотношение между магнитудой и числом землетрясений, которые имеют магнитуду больше либо

равную данной для фиксированного региона:

$$\log N = a - bM,$$

где  $N$  является числом событий с магнитудой, выше данной  $M$ ,  $a$  и  $b$  — коэффициенты, который часто обозначается  $b - value$ . Оценки коэффициентов  $a$  и  $b$  получаются из исторических данных.

Закон Омори — Утсу [3] определяет задержку активности афтершоков во времени:

$$\dot{N}(t) = \frac{C_1}{(C_2 + t)^p},$$

где  $t$  — время,  $N$  — количество землетрясений,  $C_1$ ,  $C_2$  и степень задержки  $p$  (чаще всего обозначается как  $p - value$ ) являются коэффициентами, получающимися на основе исторических данных. Обе модели обеспечивают описание числа землетрясений в целевом регионе на хорошем уровне.

Из закона Гутенберга — Рихтера можно найти число землетрясений в регионе. Из закона Омори — Утсу можно установить связь между прошедшими и будущими сейсмическими активностями в регионе.

Еще один метод, основанный на физическом подходе — RTL признаки [4], которые позволяют объединить прошедшую сейсмическую активность в один признак, с учетом взвешивания признаков, учитывая их местоположение относительно целевого региона и давность относительно землетрясения, которое мы хотим предсказать. RTL признаки имеют набор гиперпараметров, которые оценивают с помощью исторических данных.

В добавление к этим признакам можно построить модель машинного обучения [5, 6, 7, 8]. Примером такой работы может служить [7], рассматривающая задачу предсказания землетрясений как задачу бинарной классификации. Авторы генерируют 51 сейсмический признак, основанный на известных сейсмических характеристиках, таких как «Стандартное отклонение  $b$ -value» или «Время ( $T$ ), за которое происходит  $n$  событий». В качестве моделей используются различные ансамблевые методы, такие как Random Forest, Rotation Forest и RotBoost.

В этой работе будет рассмотрена другая постановка задачи: предска-

зание появления землетрясений в конкретной локации в фиксированный временной интервал.

В качестве входных признаков используются нормализованные RTL признаки с разными параметрами и различными измерениями времени. Рассматриваются несколько различных подходов машинного обучения, которые позволяют оптимально использовать сгенерированные признаки.

При разработке моделей мы учитываем особенности задачи:

- несбалансированность классов (очень малое количество сильных землетрясений в датасете) [9, 10, 11];
- данные представлены в формате временных рядов;
- все внешние признаки отсутствуют и генерация необходимых признаков происходит по данным предыдущих землетрясений.



## 3 Обзор литературы

### 3.1 Различные типы признаков

В работе [12] описывается алгоритм построения RTL признаков, его эффективность и дальнейшие возможности улучшения. Признаки учитывают и пространственные, и временные характеристики. Мотивацией для использования RTL признаков являются следующие физические соображения. Первые аномалии в признаках начинают проявляться за 1 — 2.5 года. Следующая стадия — за несколько месяцев до землетрясения. Размер зоны покоя (где изменения в этот момент можно детектировать) в несколько раз превышает размер зоны главного удара (несколько сотен километров). Землетрясения происходят тогда, когда зона пройдет сначала через покой, а затем через активацию. Таким образом, физическая модель отражает то, как генерировать признаки и в каком диапазоне искать гиперпараметры при обработке данных.

Авторы [13] предлагают модификацию алгоритма получения RTL до алгоритма получения признаков, называемых RTM. В исходном алгоритме RTL расстояния используются для расчета два раза: для функций  $R$  и  $L$ . В этой статье предлагается избавиться от этого дублирования путем ввода нового фактора  $M$  (магнитуда):

$$M(x, y, t, M) = \sum_{e_i \in \mathcal{E}} M_i - M(x, y, t, M)_{trend}.$$

Тогда RTM признаки определяются как

$$RTM(x, y, t, M) = R(x, y, t, M) \cdot T(x, y, t, M) \cdot M(x, y, M).$$

Авторы статьи считают, что этот метод потенциально эффективен для предсказания на низких магнитудах.

Авторы [1] используют шесть сейсмологических параметров. Параметры связаны с законом Гутенберга — Рихтера, выбросами сейсмической энергии и частотой форшоков. С помощью оценки коэффициентов закона

Гутенбегра — Рихтера можно оценить максимальную ожидаемую амплитуду:

$$M_{max.expected} = \frac{a}{b}.$$

Сейсмическая энергия стабильно выделяется в разломах. Если выбросы прекращаются, то она начинает накапливаться, и это называется состоянием покоя. Накопившаяся энергия может превратиться в землетрясение. Выделяющаяся энергия считается так (коэффициенты в формуле рассчитаны специально для землетрясений в Пакистане, в рамках предсказаний землетрясений в других локациях они могут отличаться):

$$dE^{\frac{1}{2}} = \frac{\sum (10^{11.8+1.5M})^{\frac{1}{2}}}{T}.$$

Средняя магнитуда нескольких последних событий тоже является одним из признаков, форшоки зачастую предвещают крупные землетрясения, и поэтому считают частоту их появления. Еще одним признаком, который учитывают, является среднее время появления какого-либо из предыдущих признаков среди последних  $n$  событий.

### 3.2 Различные методы обучения

Примеры способов предсказания землетрясений, которые используют RTL признаки и методы машинного обучения, приведены ниже:

- детектирование аномальных значений во временном ряду RTL [12];
- классификация при помощи нейронных сетей прямого распространения на признаках — производных оригинальных данных [1];
- рекуррентные нейронные сети, работающие с временными рядами по подрегионам.

Алгоритм детектирования аномальных значений во временном ряду RTL в среднесрочной перспективе:

1. Отделение афтершоков от фоновых событий: фоновые события в среднем распределены равномерно.

2. Подземные толчки дают гауссово распределение на плоскости. Центры таких толчков распределяются во времени в соответствие с законом Омори (применяется требование равновероятных ошибок).
3. Получение афтершоков.
4. Дополняем каталог искусственными данными, с шагом, соизмеримым с величиной ошибки исходных данных, заполняем близкие окрестности линейно.
5. На графике все еще могут встречаться изгибы: из-за нерепрезентативной выборки или из-за естественных причин. В диапазоне с шагом дельта изгиб оценивается с помощью метода максимального правдоподобия. Далее этот график экстраполируется.
6. Сравниваем полученные результаты с известным количеством землетрясений. Проверяем соответствие.
7. Если мы предсказали не все землетрясения с запрошенной магнитудой, то процедура повторяется. Если попали в окрестность — заканчиваем.

Авторы [1] занимаются краткосрочными (2–3 дня до события) предсказаниями землетрясений. Сбор данных для предсказаний — сроком один месяц. Решают задачу классификации землетрясений на землетрясения с магнитудой большей некоторого порога. Далее нужно из всех этих признаков выбрать те, которые наиболее значимы для нас в данной ситуации. Можно посмотреть на то, какие признаки используются для предсказания землетрясений в других местах и понять, что не все они также применимы для Пакистана (после обработки осталось только шесть признаков). Для предсказаний авторы попробовали шесть различных алгоритмов: случайный лес, рекуррентные нейронные сети, многослойный перцептрон, радиально-базисную функцию, метод опорных векторов и нейронные сети прямого распространения. FFNN дает наилучшие результаты для Пакистана.

Авторы статьи [14] используют рекуррентные нейронные сети для среднесрочного предсказания землетрясений. В качестве входных данных ис-

пользуются векторы для подрегионов, из которых составляется временной ряд, поступающий в модель. Авторы использовали вектора размерности  $1 \times n$  и  $2 \times n$ , в результате чего пришли к выводу, что вектора размерности 2 дают лучшее качество обучения в целом, а также лучше детектируют целевой класс. Данный подход (LSTM) авторы проверяли на данных, собранных в шести разных регионах и пришли к выводу, что данная модель работает лучше, чем AdaBoost, метод  $k$  ближайших соседей, метод опорных векторов и другие виды нейронных сетей, которые тоже были проверены на этих данных.

### 3.3 Полученные результаты

В статье [1] авторы достигают точности в 78% для краткосрочных землетрясений. Помимо классических метрик в качестве метрики используется метрика R-score и Mathew's Correlation coefficient. Для проверки эффективности моделей используется тест Макнемара. Если значение коэффициента значимости достаточно мало (меньше 0.05), то результаты считаются значимыми, что выполнено для классификации с помощью нейронных сетей прямого распространения.

В статье [14] достигается точность 85.12%, а точность по целевому классу 77.07%.

### 3.4 Выводы

Таким образом, алгоритмы предсказания землетрясений состоят из следующих этапов:

1. Генерация признаков на основе данных о прошлых землетрясениях.
2. Применение методов машинного обучения.

Для этапа (1) используются RTL, RTM, сегментация, разделение на сетку и подрегионы, выделение временных рядов векторов состояний.

Для этапа (2) используются нейронные сети, случайный лес, RotBoost и классификаторы.

Однако, в литературе не обоснован выбор гиперпараметров для генерации признаков, не учитываются особенности задачи, такие как: несбалансированность классов, неоднородность данных (2.1).

## 4 Методы

Чтобы решить задачу классификации, необходимо создать матрицу входных признаков  $\mathbf{X}$  и вектор целевых меток  $\mathbf{Y}$ . Следовательно, мы хотим найти метки целевых событий  $Y$  для признаков  $X^j$ , каждая из которых обозначает пространственно-временной цилиндр для вектора  $(x, y, t)$ . Следовательно, мы создаем метки для индексов  $(x, y, t)$ . Получение  $X, Y$  на основе RTL:

1. Матрица  $\mathbf{X}$  генерируется, используя агрегацию  $R, T$  и  $L$  признаков для 10 моментов времени через различные усредняющие функции, такие как медиана, среднее и кумулятивная сумма.
2. Для фиксированной в пространстве-времени точки строится пространственно-временной цилиндр, и для каждого события, попавшего в него, определяется, превосходит ли магнитуда порог или нет. Если да, то компонента в векторе  $\mathbf{Y}$  будет равна 1, 0 иначе.

### 4.1 Данные

Целью задачи является предсказание сильных землетрясений в среднесрочной перспективе. Сильным землетрясением считается землетрясение магнитудой больше чем  $M_c = 5$ . Предсказание землетрясений непростая задача не только из-за того, что это сложный нелинейный процесс, но и из-за того, что исторически данные имеют неочевидную структуру:

- В Японии в период с 1990 по 2016 года произошли 247 204 землетрясений, однако выборка очень несбалансированна: на рисунке 2 приведено распределение магнитуд землетрясений. Большая часть классификаторов и их метрик точности заточена под баласированные выборки с равномерным распределением среди различных классов. Следовательно, в рамках нашей задачи необходимо настроить классификатор для большей чувствительности к целевому классу.
- Данные неоднородные, так как сеть сейсмических станций меняется

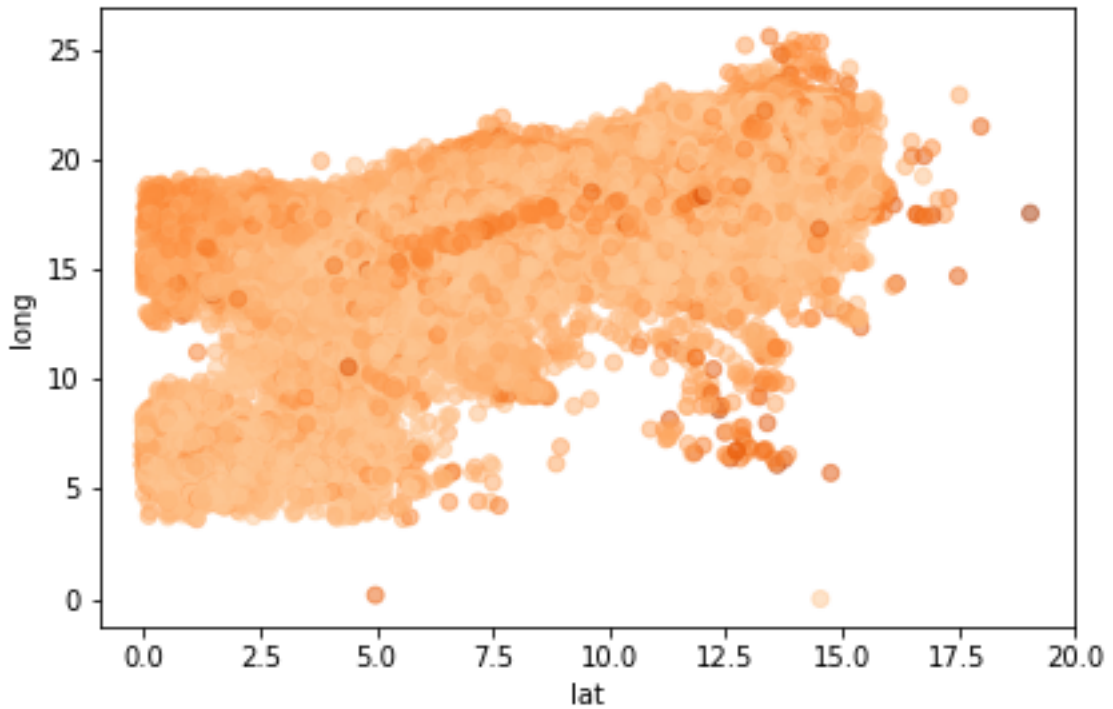


Рис. 1: Данные о землетрясениях в Японии с 1990 по 2016 года

с течением времени, они неравномерно расположены на территории Японии. Следовательно, необходимо учесть это при генерации признаков и оценке результатов предсказания модели.

## 4.2 RTL признаки

В качестве входных данных для модели машинного обучения используются RTL признаки. Алгоритм **Region – Time – Length (RTL)** (Регион — Время — Длина) [4] основан на предположении о том, что влияние каждого предыдущего события на текущее может быть учтено с использованием веса события. Веса становятся больше, если у землетрясения большая магнитуда или они расположены ближе к целевой точке в пространстве-времени. Следовательно, **RTL** характеризует уровень сейсмичности в конкретной локации в фиксированное время.

**RTL** учитывает взвешенные признаки, соотносящиеся с тремя параметрами (время, место и магнитуда) землетрясений.

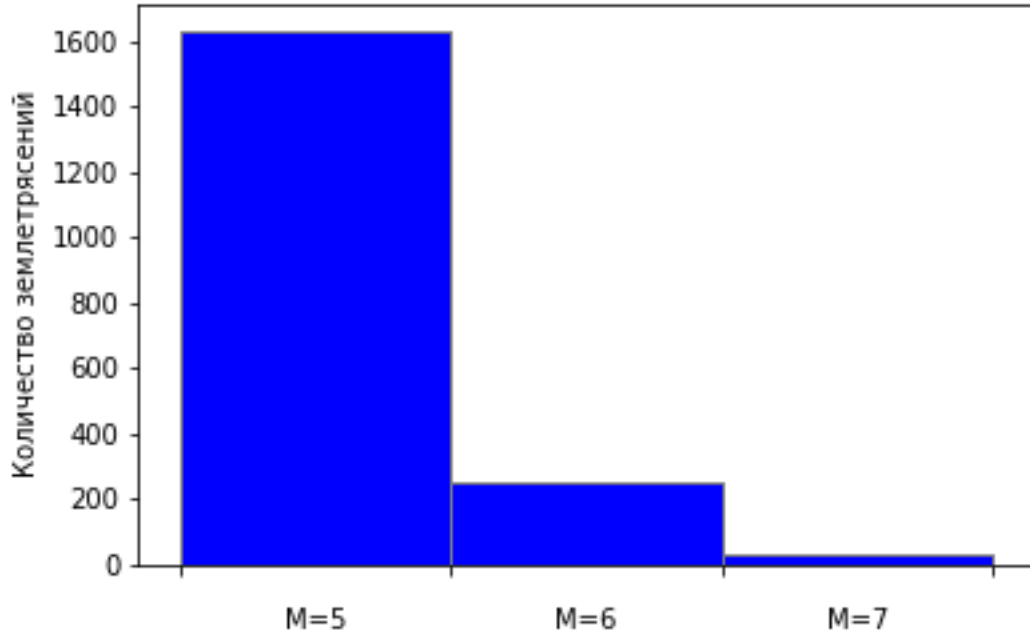


Рис. 2: Распределение магнитуд: итоговый размер выборки примерно 300000; из них только 2000 землетрясений с магнитудой большей или равной 5.

**RTL** признак определен как произведение следующих трех функций:

$$\text{RTL}(x, y, t, M) = R(x, y, t, M) \cdot T(x, y, t, M) \cdot L(x, y, t, M),$$

где  $R(x, y, t, M)$  — расстояние до эпицентра,  $T(x, y, t, M)$  — разница во времени и  $L(x, y, t, M)$  — длина разрыва. Они зависят от размера пространственно — временного цилиндра  $\mathcal{E}_{r_0, t_0}$ , который определяется радиусом  $r_0$  и временным промежутком  $t_0$ , как показано на рисунке 3:

$$\begin{aligned} R(x, y, t, M) &= \sum_{e_i \in \mathcal{E}} \exp \left( -\frac{r_i}{r_0} \right), \\ T(x, y, t, M) &= \sum_{e_i \in \mathcal{E}} \exp \left( -\frac{t - t_i}{t_0} \right), \\ L(x, y, t, M) &= \sum_{e_i \in \mathcal{E}} \left( \frac{l_i}{r_i} \right), \end{aligned}$$

где  $e_i$  — землетрясения  $(x_i, y_i, t_i, M_i)$  из  $\mathcal{E}$ ,  $l_i$  — эмпирическое соотношение,



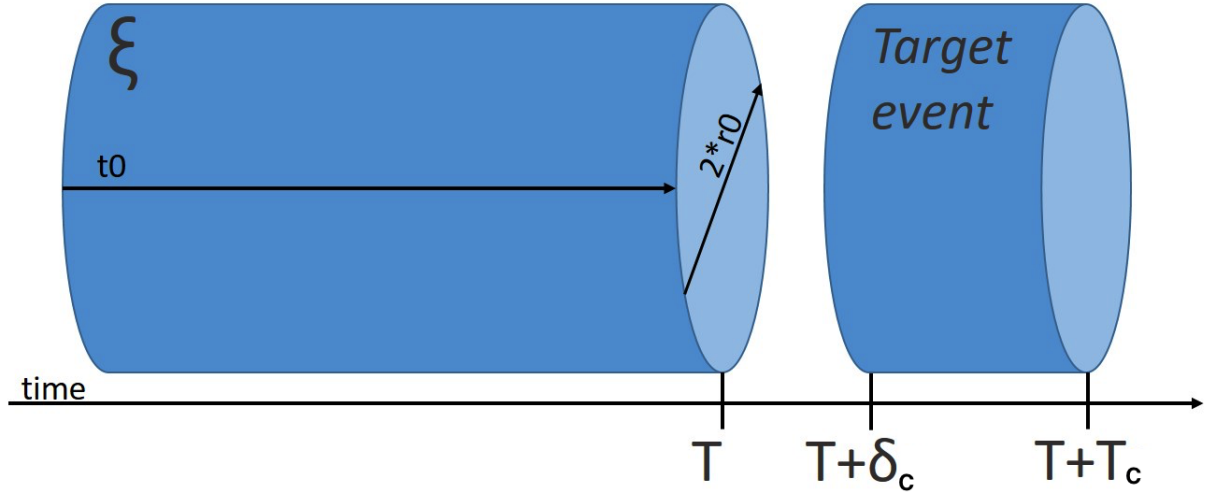


Рис. 3: Пространственно-временной цилиндр, использующийся для генерации RTL признаков и интервал, для которого нужно сделать предсказание

специфически определенное для Японии как

$$\log l_i = 0.5M_i - 1.8,$$

и  $r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$ . Мы учитываем только землетрясения, магнитуда которых  $M_i$  достигает  $M_c = 5$ .

Для **RTL** признаков, мы можем получить целый ряд признаков, меняя значение гиперпараметров  $r_0$  и  $t_0$ .

Опишем наш подход к предсказанию землетрясений.

- Конструируем сетку гиперпараметров  $(r_0^i, t_0^i)$ . Для каждого набора гиперпараметров считаем RTL статистику для каждого землетрясения и получаем матрицу признаков  $\mathbf{X}$ .
- Рассматриваем сетку гиперпараметров  $(M^i, R_c^i, \delta_c^i, T_c^i)$ . Для каждого набора параметров получаем метки целевых событий, откуда получаем вектор меток  $\mathbf{Y}$ .
- Строим модель, обучаясь на полученных признаках.

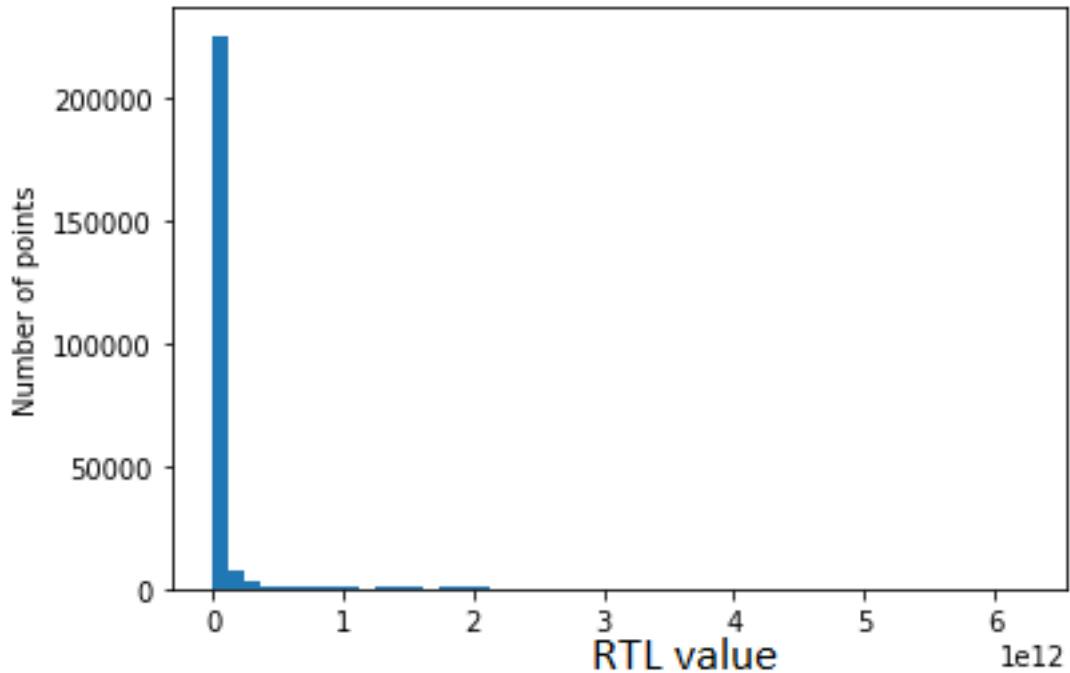


Рис. 4: Гистограмма значений RTL для каждой точки. Маленькие значения заметно преобладают.

#### 4.2.1 Нормализация RTL признаков

**RTL** — очень нестабильная статистика. В связи с этим, [12] предлагает нормализовать параметры.

В данной работе данные на стадии предобработки подлежат нормализации. Для нашего конкретного случая мы преобразуем данные так, чтобы среднее значение равнялось нулю, а дисперсия равнялась единице [15]:

- для каждого признака подсчитать среднее и дисперсию,
- из значений для каждого признака вычесть его среднее,
- поделить получившееся значение каждого признака на его дисперсию.

Также можно вычитать скользящее среднее вместо обычного среднего значения. Это позволяет учесть тенденцию изменений во времени (для длинных временных рядов порядка десятка лет).

Отрицательное значение **RTL** означает меньшую сейсмичность в сравнении с исходным уровнем в окрестностях исследуемой локации, а положительное значение **RTL** показывает большую сейсмичность в сравнении

с изначальным уровнем. На гистограмме 4 можно увидеть распределение значений RTL. С точки зрения предсказания землетрясений важны оба типа аномалий.

### 4.3 Классификаторы

В качестве классификаторов используются следующие методы машинного обучения:

- **Major RTL** (превалирующий RTL) — метод, который подбирает пороговое значение RTL признаков. Мы оцениваем оптимальное значение порога на основе обучающих данных. Если значение признака для конкретного случая меньше, чем этот порог, то он получает метку 0, 1 иначе. Используя эту оценку для порога, мы можем классифицировать примеры из новых данных. От результатов работы этого алгоритма мы и отталкиваемся, когда оцениваем вклад данной работы.
- **Logistic regression** (логистическая регрессия) — статистическая модель, используемая для предсказания вероятности того, что данное событие произойдет. Математически модель выражается как:

$$Pr(c = 1|\mathbf{X}) = f(\boldsymbol{\theta}^\top \mathbf{X} + b),$$

где  $f(z) = \frac{1}{1+e^{-z}}$  — логистическая функция, а  $\boldsymbol{\theta}$  и  $b$  — параметры модели, получающиеся оцениванием данных обучающей выборки [16].

- **Random Forest** (Случайный лес)— ансамблевый классификатор, работа которого основана на голосовании большинства нескольких решающих деревьев. Случайное число решающих деревьев генерируются в процессе бутстрэпа данных из обучающего датасета. Окончательное решение получается объединением предсказаний полученных всеми решающими деревьями. Следовательно, случайный лес позволяет найти сложные соотношения в данных, но при этом менее склонен к переобучению [17].

- **AdaBoost** — другой метод, позволяющий объединить классификаторы в ансамбль. Основная идея метода состоит в том, что он присваивает веса всем объектам. На каждой итерации веса каждого неправильно классифицированного объекта становятся больше, так что новый ансамбль классификаторов концентрирует свое внимание на объектах, ранее классифицированных неверно. AdaBoost чувствителен к шуму и менее склонен к переобучению, нежели другие алгоритмы машинного обучения [18].
- **Gradient Boosting** (градиентный бустинг) — ансамблевый метод, который строится на ансамблях деревьев, а затем предсказания отдельных деревьев объединяются. Следующее решающее дерево старается приблизить текущую функцию к целевой с помощью изменения остатков. Следовательно, процесс итеративный — на каждой итерации функция потерь минимизируется с помощью градиентного спуска [19]. Еще один из плюсов градиентного бустинга — способность классифицировать несбалансированные классы после несложной модификации [20].

#### 4.4 Несбалансированная задача классификации

Мы рассматриваем несбалансированную задачу классификации: землетрясений с большими магнитудами происходит немного, поэтому большинство землетрясений в выборке относятся к классу слабых. Естественно использовать подходы машинного обучения, которые работают с несбалансированной выборкой, как например, [10, 11].

Рассмотрим несколько простых техник работы с несбалансированной выборкой, мы модифицируем исходный процесс обучения так, чтобы меньший класс имел большее влияние на процесс:

- **Oversampling** — дополняет выборку объектами меньшего класса, генерация которых происходит на основе алгоритмов, описанных ниже.
- **Undersampling** — уменьшает количество объектов большего класса,

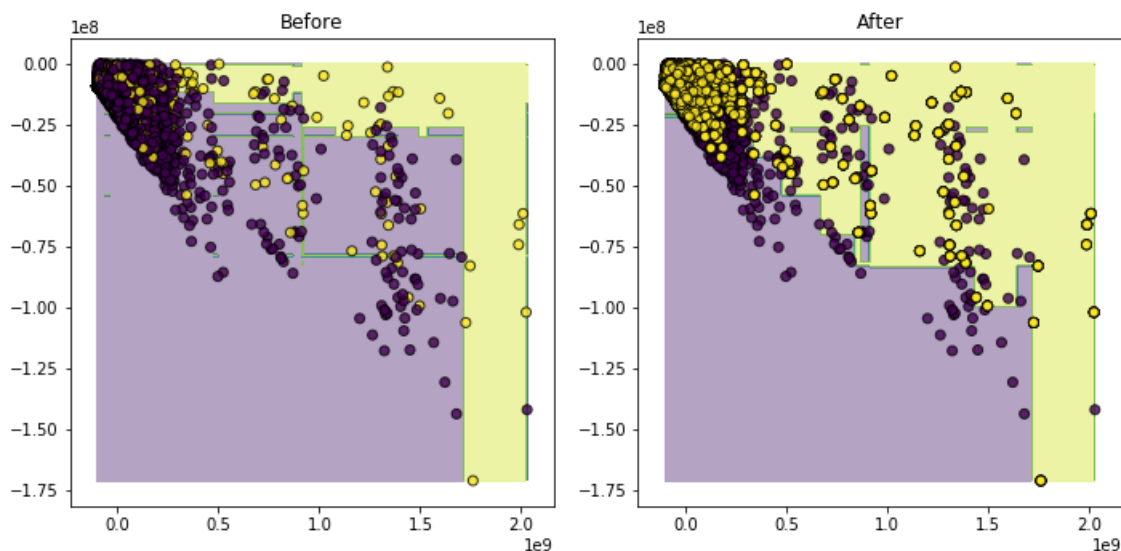


Рис. 5: Пример oversampling-a на наших данных по землетрясению

чтобы сбалансировать количество объектов обоих классов.

Используемые техники **oversampling-a**:

- SMOTE (синтетическая генерация данных меньшего класса) — основана на случайном выборе точки вдоль линейного сегмента между двумя признаками. Использует значения признаков ближайших соседей. Преимущества алгоритма в том, что он расширяет сферу данного класса, но при этом уменьшает специфичность.
- ADASYN (адаптивный подход к синтезированию искусственных данных) — алгоритм, основанный на пошаговой случайной генерации новых признаков на основе данных о ближайших соседях. В отличие от предыдущего алгоритма все  $k$  ближайших соседей используются для генерации. Преимущества состоят в том, что границы адаптивно сдвигаются в сторону наиболее сложных примеров.

Используемые техники **undersampling-a**:

- NearMiss — основывается на том, что убирать объекты большего класса надо из тех, которые ближе всего находятся к меньшему классу в пространстве признаков. Существуют различные подходы к вычислению таких объектов, как до ближайших объектов из меньшего класса или до дальних.

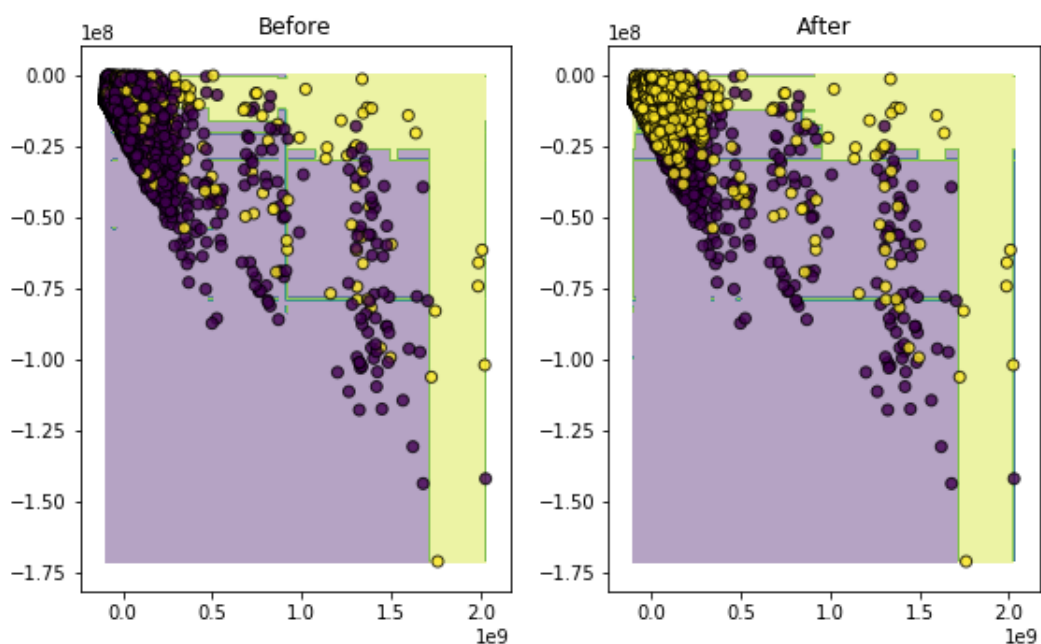


Рис. 6: Пример undersampling-a на наших данных по землетрясению

- Edited nearest neighbors — основная идея алгоритма состоит в том, чтобы даже уменьшенная выборка не теряла информацию, необходимую для алгоритма ближайших соседей. Если объект согласуется со своей классификацией по этому методу, то его можно убирать.

На Рисунке 5 показан пример oversampling-a с помощью алгоритма ADASYN на данных задачи. Для лучшей визуализации я взяла небольшую часть выборки (10000 объектов) и на них провела классификацию. В целях лучшей визуализации 10 признаков, которые получаются в результате генерации матрицы  $\mathbf{X}$  методом главных компонент, я свела до 2, которые и отложены по осям. По рисунку можно пронаблюдать изменение положения разделяющей поверхности в зависимости от применения данного метода к данным. Слева — состояние до применения алгоритма, справа — после. Можно заметить, что разделяющая поверхность лучше разделяет данные после применения алгоритма.

На Рисунке 6 показан пример undersampling-a при помощи алгоритма NearMiss. Данные были преобразованы аналогично тому, что указано в предыдущем пункте. Точно так же, слева — состояние до применения ресэмплинга, справа — после. Заметно более явное отношение элементов це-

левого класса к части поверхности.

Метод главных компонент — метод, применяющийся в задачах снижения размерности. Основная идея этого метода заключается в том, чтобы найти такую плоскость, ортогональная проекция исходных данных на которую сохраняет больше всего оригинальной информации. Математически сводится к сингулярному разложению или к вычислению собственных векторов.

## 4.5 Скользящий контроль

В целях борьбы с переобучением и, как следствие, улучшения качества классификации в стандартных задачах машинного обучения используют кросс-валидацию. Общая идея кросс-валидации такова: данные разбиваются на  $n$  блоков, каждый из которых бывает тестовыми данными в то время как классификатор обучается на остальных  $n - 1$  блоках, после чего результаты усредняются. Также, обычно это разбиение случайное, чтобы аккуратно обработать шум и различные выбросы в данных.

Во время работы с временными рядами такой принцип работы не представляется возможным: предсказание прошлых событий по будущим хоть и возможно, но не очень логично в рамках задачи. То есть данные для обучения должны всегда предшествовать в ряду данным для тестирования. Я для оптимальной кросс-валидации случайно выбирала точку, на которой будет учиться классификатор на данной итерации, выбирала длину обучающих данных, а потом выбирала длину тестовых данных, которые должны были следовать после обучающих, но при этом захватывать определенный временной промежуток, так как в рамках нашей задачи данные, отстоящие друг от друга больше, чем на 3 года, не предполагают корреляции.

## 4.6 Метрики

Задача предсказания землетрясений — задача несбалансированной классификации. Для оценивания качества моделей, которые решают эту задачу, используют следующие метрики:

- True Positive — если объект принадлежит целевому классу и предсказание определяет его как объект, принадлежащий целевому классу,
- True Negative — если объект не принадлежит целевому классу и предсказание определяет его как объект, не принадлежащий целевому классу,
- False Positive — если объект не принадлежит целевому классу, а предсказание определяет его как объект, принадлежащий целевому классу,
- False Negative — если объект принадлежит целевому классу, а предсказание определяет его как объект, не принадлежащий целевому классу.

**Precision** score (точность) отражает способность классификатора не отмечать нецелевые объекты как целевые. **Precision** — вероятность того, что положительное предсказание, сделанное классификатором действительно положительное. Значение данной метрики принадлежит промежутку  $[0, 1]$  и достигает 0 в худшем случае и 1 в идеальном.

Точность определяется как

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}.$$

**Recall** score (полнота) отражает способность классификатора находить все целевые объекты выборки. Она определяет, какая часть из положительных объектов была выбрана классификатором как положительные объекты. Значение метрики принадлежит промежутку  $[0, 1]$ , достигая 0 в худшем случае и 1 в лучшем.

Полнота определяется так:

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}.$$

**F1-score** — единая метрика, которая комбинирует значения точности и полноты через их среднее гармоническое. Она отражает, насколько точны предсказания на тесте и достигает своего лучшего значения в 1 (идеальные точность и полнота) и худшего в 0.



**F1-score** может быть определена так:

$$\mathbf{F1} = 2 \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

**ROC AUC score** — считает площадь под ROC—кривой. ROC—кривая — кривая, которая показывает, как доля True Positive зависит от доли False Positive. Эти характеристики определяются как:

$$\text{True Positive Rate} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}},$$

$$\text{False Positive Rate} = \frac{\text{False Positive}}{\text{False Positive} + \text{True Negative}}.$$

**ROC AUC score** измеряет качество бинарного классификатора. Лучшее значение равно 1, значение, равное 0.5, отвечает случайной классификации.

**PR AUC score** — считает площадь под кривой Precision-Recall (точность - полнота), которая характеризует как точность зависит от полноты. Precision-Recall — полезная метрика классификаторов, особенно в случаях с несбалансированной выборкой. Для идеального классификатора кривая кончается в (1.0, 1.0) и имеет площадь под графиком, равную 1.

## 5 Эксперименты

Выбор метрик и классификаторов обусловлен тем, что выборка, данная для задачи, крайне несбалансированна. В дополнение к обычно используемым Precision, Recall и ROC AUC мы используем F1-score и Precision-Recall AUC (PR AUC).

Для приведенных ниже экспериментов используются следующие параметры  $M_c = 5$ ,  $R_c = 50$  (км),  $\delta_c = 10$  (дней),  $T_c = 180$  (дней).

### 5.1 Выбор параметров для RTL

Мы используем следующие наборы гиперпараметров для расчета **RTL** признаков:

$$\vec{r}_0 = [10, 25, 50, 100], \quad \vec{t}_0 = [30, 90, 180, 365]. \quad (2)$$

То есть происходит генерация  $4 \times 4 = 16$  различных типов признаков для разных значений  $(r_0, t_0)$ . Для каждой пары  $(r_0, t_0)$  генерируются 10 различных авторегрессионных признаков: мы рассматриваем не только текущий момент времени  $t$ , но также  $t - 1, t - 2, \dots, t - 9$ . Мы используем эти 10 признаков в качестве входных данных для модели машинного обучения.

В таблице 1 мы приводим результаты для всех рассмотренных значений  $r_0$  и некоторых значений  $t_0$ . В таблицу включены результаты для тех значений времени  $t_0$  при фиксированном  $r_0$ , которые максимизируют ROC AUC для данной модели среди всех рассмотренных значений  $t_0$ , указанных в (2).

Заметим, что признаки с большим значением  $r_0$  и сравнительно большим значением  $t_0$  помогают достичь наилучшей точности модели. Это ожидаемо, так как мы получаем больший объем данных для обучения.

Лучшей моделью для наших экспериментов является градиентный бустинг [21]. Градиентный бустинг работает лучше, чем линейная регрессия, так как магнитуда землетрясений зависит от значений RTL признаков крайне нелинейным образом. Оба этих подхода работают эффективнее, чем превалирующее значение. Таким образом, более сложные подходы машин-

$r_0$	Лучший $t_0$	Алгоритм	Precision	Recall	F1	ROC AUC	PR AUC
10	180	Logistic Regression	0.54	0.36	0.43	0.64	0.53
		Random Forest	0.62	0.51	0.56	0.76	0.69
		AdaBoost	0.63	0.50	0.56	0.77	0.83
		Gradient Boosting	0.62	0.52	0.56	0.80	0.69
		Major_RTL	0.57	0.47	0.47	0.52	0.77
25	90	Logistic Regression	0.72	0.58	0.64	0.70	0.44
		Random Forest	0.79	0.51	0.62	0.74	0.68
		AdaBoost	0.75	0.67	0.71	0.77	0.53
		Gradient Boosting	0.82	0.6	0.69	0.79	0.77
		Major_RTL	0.67	0.52	0.62	0.70	0.60
50	180	Logistic Regression	0.91	0.84	0.88	0.83	0.67
		Random Forest	0.91	0.84	0.87	0.80	0.70
		AdaBoost	0.83	0.96	0.89	0.81	0.74
		Gradient Boosting	0.89	0.92	0.91	0.87	0.73
		Major_RTL	0.60	0.74	0.74	0.80	0.71
100	180	Logistic Regression	0.94	0.94	0.94	0.80	<b>0.94</b>
		Random Forest	<b>0.97</b>	0.90	0.94	0.89	0.90
		AdaBoost	0.96	0.96	0.96	0.92	0.90
		Gradient Boosting	0.95	<b>0.98</b>	<b>0.97</b>	<b>0.93</b>	<b>0.94</b>
		Major_RTL	0.90	0.86	0.89	0.83	0.89

Таблица 1: Результаты для различных значений гиперпараметров для сгенерированных RTL признаков: лучшие результаты получаются для большего размера цилиндра

ного обучения работают лучше простого порогового правила для предсказания землетрясений, вероятнее всего из-за сложности зависимости вероятности появления землетрясений от RTL признаков.

## 5.2 Агрегация нескольких RTL

В этом разделе мы используем 16 рассчитанных RTL признаков в качестве входных данных для моделей машинного обучения. В данном случае размер входных данных становится равен  $10 \times 16 = 160$ . Полученные результаты приведены в Таблице 2. Улучшение результатов в сравнении с предыдущим пунктом, где RTL признаки генерируются только один раз, незначительное. Можно заключить, что невозможно существенно улучшить точность модели, используя объединение нескольких RTL признаков для различных гиперпараметров.

Алгоритм	Precision	Recall	F1	ROC AUC	PR AUC
Gradient Boosting (best single RTL)	0.97	0.96	0.96	0.95	0.94
Major_RTL	0.90	0.85	0.89	0.82	0.89
Logistic Regression	0.94	0.95	0.94	0.81	0.94
Random Forest	0.97	0.90	0.94	0.89	0.90
AdaBoost	0.97	0.96	0.97	0.92	0.90
Gradient Boosting	0.95	0.98	0.97	0.93	0.93

Таблица 2: Результаты для нескольких объединенных RTL признаков, для генерации которых использовались разные гиперпараметры

Подход	Алгоритм	Precision	Recall	F1
No resampling	Gradient Boosting	0.84	0.59	0.59
Oversampling	Random Forest	0.94	0.99	0.97
Undersampling	Random Forest	0.96	0.87	0.91

Таблица 3: Качество обучения моделей при использовании подходов несбалансированной классификации, основанных на изменении пропорций исходной выборки.

### 5.3 Использование ресемплинга

Есть несколько подходов к несбалансированной классификации, основанных на идее изменений пропорций меньшего и большего классов в исходной выборке. [9, 10, 11], методы описаны выше в разделе 4.4. Рассмотрим классификацию для случаев oversampling-a и undersampling-a наряду со случаем, где баланс классов не изменялся. Результаты для каждого случая приведены в Таблице 3 для модели машинного обучения, которая показывает лучшие результаты на целевых метриках для каждого конкретного случая. В данных экспериментах использовалась выборка меньшего размера, в отличие от предыдущих пунктов.

Обе техники (и undersampling, и oversampling) улучшают качество моделей.

## 6 Выводы

Была рассмотрена задача среднесрочного прогнозирования землетрясений.

- Использование машинного обучения позволяет улучшить результат, достигаемый с помощью современных подходов (major RTL), основанных только на использовании экспертных знаний.
- Конкретная модель, основанная на градиентном бустинге, который принимает RTL признаки на вход, показывает лучший результат.
- Для большинства случаев для генерации RTL признаков использовался один набор гиперпараметров, и результат предсказаний получался не хуже, чем при использовании нескольких параметров для генерации multi-RTL признаков.
- Также было получено, что использование подходов несбалансированной классификации может увеличить точность предсказаний.

Дальнейшие пути развития модели, которые могут улучшить точность предсказаний:

- землетрясения с большой магнитудой — очень редкие события, по сути — аномалии. Следовательно, можно сначала установить последовательность различных аномалий среди исторических данных по землетрясениям и среди различных функций от этих данных (установить моменты, в которые происходят значительные изменения) [22, 23, 24, 25, 26], а затем соорудить ансамбль для предсказания появления редких событий [27, 28], используя найденные аномалии и их признаки в качестве предвестников сильных землетрясений, чтобы оптимизировать специфическое детектирование метрик, как в [29];
- использование привелегированной информации о будущих событиях, доступной на этапе обучения. Аналогичный подход был использован в [30, 31] для детектирования аномалий и показал значительное улучшение точности;

- исторические данные по землетрясениям содержат пространственную компоненту, следовательно, можно сконструировать граф зависимостей событий, зарегистрированных различными геологическими станциями, и использовать современные подходы, обучающиеся на графах в качестве входных данных [32]. Извлечение признаков временного ряда [33, 34] может применяться для обогащения набора входных признаков, используемых для обучения предсказательной модели;
- использование сверточных нейронных сетей для детектирования и обнаружения землетрясений по подрегионам, как в статье [35]. Благодаря сверткам требуется в разы меньший объем данных для обнаружения землетрясения с фактически не уменьшающимся качеством обнаружения.

По теме данного исследования была написана статья **Usage of multiple RTL features for Earthquakes prediction**, которая будет опубликована в "The 19th International Conference on Computational Science and its Applications".

## Список литературы

- [1] Khawaja M Asim, Muhammad Awais, F Martínez-Álvarez, and Talat Iqbal. Seismic activity prediction using computational intelligence techniques in northern Pakistan. *Acta Geophysica*, 65(5):919–930, 2017.
- [2] B. Gutenberg and C. Richter. *Seismicity of the Earth and associated phenomena*. 1951.
- [3] Tokuji Utsu, Yosihiko Ogata, et al. The centenary of the Omori formula for a decay law of aftershock activity. *Journal of Physics of the Earth*, 43(1):1–33, 1995.
- [4] G.A. Sobolev and Y.S. Tyupkin. Low-seismicity precursors of large earthquakes in Kamchatka. *Volcanol. Seismol.*, 18:433–446, 1997.

- [5] Ashif Panakkat and Hojjat Adeli. Neural network models for earthquake magnitude prediction using multiple seismicity indicators. *International journal of neural systems*, 17(01):13–33, 2007.
- [6] Bertrand Rouet-Leduc, Claudia Hulbert, Nicholas Lubbers, Kipton Barros, Colin J Humphreys, and Paul A Johnson. Machine learning predicts laboratory earthquakes. *Geophysical Research Letters*, 44(18):9276–9282, 2017.
- [7] KM Asim, F Martínez-Álvarez, A Basit, and T Iqbal. Earthquake magnitude prediction in Hindukush region using machine learning techniques. *Natural Hazards*, 85(1):471–486, 2017.
- [8] G Asencio-Cortés, Francisco Martínez-Álvarez, A Morales-Esteban, and Jorge Reyes. A sensitivity study of seismicity indicators in supervised learning to improve earthquake prediction. *Knowledge-Based Systems*, 101:15–30, 2016.
- [9] Evgeny Burnaev, Pavel Erofeev, and Dmitry Smolyakov. Model selection for anomaly detection. In *Eighth International Conference on Machine Vision (ICMV 2015)*, volume 9875, page 987525. International Society for Optics and Photonics, 2015.
- [10] Evgeny Burnaev, Pavel Erofeev, and Artem Papanov. Influence of resampling on accuracy of imbalanced classification. In *Eighth International Conference on Machine Vision (ICMV 2015)*, volume 9875, page 987521. International Society for Optics and Photonics, 2015.
- [11] D. Smolyakov, A. Korotin, P. Erofeev, A. Papanov, and E. Burnaev. Meta-learning for resampling recommendation systems. In *Proc. SPIE 11041, Eleventh International Conference on Machine Vision (ICMV 2018), 110411S (15 March 2019)*, 2019.
- [12] Qinghua Huang. Seismicity pattern changes prior to large earthquakes-an approach of the RTL algorithm. *TERRESTRIAL ATMOSPHERIC AND OCEANIC SCIENCES*, 15(3):469–492, 2004.

- [13] Kenji Nakamura Toshiyasu Nagao, Akihiro Takeuchi. A new algorithm for the detection of seismic quiescence: introduction of the rtm algorithm, a modified rtl algorithm. In *Earth Planets and Space*. Springer, LNCS, 2011.
- [14] Lixing Yu Qianlong Wang, Yifan Guo and Pan Li. Earthquake prediction based on spatio-temporal data mining: An lstm network approach. In *IEEE Transactions on Emerging Topics in Computing*. IEEE, 2017.
- [15] Selim Aksoy and Robert M Haralick. Feature normalization and likelihood-based similarity measures for image retrieval. *Pattern recognition letters*, 22(5):563–582, 2001.
- [16] Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2):337–407, 2000.
- [17] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE, 1995.
- [18] Y Freund and RE Schapire. A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence*, 14, 1999.
- [19] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.
- [20] Nataliia Kozlovskaya and Alexey Zaytsev. Deep ensembles for imbalanced classification. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 908–913. IEEE, 2017.
- [21] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM, 2016.
- [22] A. Artemov and E. Burnaev. Detecting performance degradation of software-intensive systems in the presence of trends and long-range



dependence. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pages 29–36, Dec 2016.

- [23] A. Safin and E. Burnaev. Conformal kernel expected similarity for anomaly detection in time-series data. *Advances in Systems Science and Applications*, 17(3):22–33, 2017.
- [24] E. V. Burnaev and G. K. Golubev. On one problem in multichannel signal detection. *Problems of Information Transmission*, 53(4):368–380, Oct 2017.
- [25] V. Ishimtsev, A. Bernstein, E. Burnaev, and I. Nazarov. Conformal k-nn anomaly detector for univariate data streams. In Alex Gammerman, Vladimir Vovk, Zhiyuan Luo, and Harris Papadopoulos, editors, *Proceedings of the Sixth Workshop on Conformal and Probabilistic Prediction and Applications*, volume 60 of *Proceedings of Machine Learning Research*, pages 213–227, Stockholm, Sweden, 13–16 Jun 2017. PMLR.
- [26] D. Volkhonskiy, E. Burnaev, I. Nouretdinov, A. Gammerman, and V. Vovk. Inductive conformal martingales for change-point detection. In Alex Gammerman, Vladimir Vovk, Zhiyuan Luo, and Harris Papadopoulos, editors, *Proceedings of the Sixth Workshop on Conformal and Probabilistic Prediction and Applications*, volume 60 of *Proceedings of Machine Learning Research*, pages 132–153, Stockholm, Sweden, 13–16 Jun 2017. PMLR.
- [27] A. Artemov and E. Burnaev. Ensembles of detectors for online detection of transient changes. In *Proc. SPIE*, volume 9875, pages 9875 – 9875 – 5, 2015.
- [28] D. Smolyakov, N. Sviridenko, V. Ishimtsev, E. Burikov, and E. Burnaev. Learning ensembles of anomaly detectors on synthetic data. In *Proc. of ISNN*. Springer, LNCS, 2019.
- [29] E. Burnaev, I. Koptelov, G. Novikov, and T. Khanipov. Automatic construction of a recurrent neural network based classifier for vehicle

passage detection. In *Proc. SPIE*, volume 10341, pages 10341 – 10341 – 6, 2017.

- [30] E. Burnaev and D. Smolyakov. One-class svm with privileged information and its application to malware detection. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pages 273–280, Dec 2016.
- [31] D. Smolyakov, N. Sviridenko, E. Burikov, and E. Burnaev. Anomaly pattern recognition with privileged information for sensor fault detection. In Luca Pancioni, Friedhelm Schwenker, and Edmondo Trentin, editors, *Artificial Neural Networks in Pattern Recognition*, pages 320–332, Cham, 2018. Springer International Publishing.
- [32] S. Ivanov and E. Burnaev. Anonymous walk embeddings. In *Proc. of ICML*, 2018.
- [33] R. Rivera, P. Pilyugina, A. Pletnev, I. Maksimov, W. Wyz, and E. Burnaev. Topological data analysis of time series data for b2b customer relationship management. In *Proc. of Industrial Marketing and Purchasing Group Conference (IMP19)*, The IMP Journal, 2019.
- [34] R. Rivera, I. Nazarov, and E. Burnaev. Towards forecast techniques for business analysts of large commercial data sets using matrix factorization methods. *Journal of Physics: Conference Series*, 1117(1):012010, 2018.
- [35] Marine Denolle Thibaut Perol, Michaël Gharbi. Convolutional neural network for earthquake detection and location. In *SCIENCE ADVANCES*. AAAS, 2018.