Классификация музыкальных композиций по инструментам

Николай Стулов
группа 473

6 марта 2017 г.

1 Цели и задачи

В работе рассматриваются временные ряды из датасета IRMAS, представляющие из себя аудиозаписи с определенным музыкальным инструментом.

Цель работы заключается в построении как можно более эффективного инструмента классификации композиций по музыкальным инструментам. Данная задача может быть эффективно решена с помощью методов машинного обучения, поскольку объем данных велик, при том, что их структура не является очевидной. Задача не является новой, как и получение информации из музыкальных композиций в целом, однако очень популярна, и, несмотря на кажущуюся простоту, имеет широкое применение — например, в распознавании голоса.

Данная работа состоит из двух основных частей: получение признаков из временного ряда и обучение различных классификаторов.

2 Используемые методы

Практические результаты получены с использованием языка программирования Python, а также библиотек Essentia (с реализацией алгоритмов получения признаков из временных рядов), scikit-learn и XGBoost (с реализацией алгоритмов машинного обучения).

3 Алгоритмическая справка

3.1 Алгоритмы Essentia

3.1.1 MonoLoader

Физически wav-файл содержит информацию об аудиоволне. Данный алгоритм извлекает эту информацию и представляет волну в виде массива вещественных чисел размера $time * framerate$, характеризующих амплитуду волны в каждой известной точке временного ряда.

3.1.2 ZeroCrossingRate

Временной ряд содержит амплитуды разных знаков. Соответственно, существуют точки, в которых волна пересекает ось абсцисс. Алгоритм вычисляет отношение числа пересечений оси абсцисс к длине ряда. Эта характеристика ряда является частотной и имеет прямое отношение к тональности (также частотная характеристика), поэтому включена в рассматриваемые.

3.1.3 Energy

По определению, энергией в обработке сигналов называется $E_s = \langle x(t), x(t) \rangle = \sum_{n=-\infty}^{\infty} |x(n)|^2$. Данная характеристика ряда является амплитудной и может быть интерпретирована как общая громкость композиции. Громкость также позволяет делать некоторые заключения об инструменте, поэтому энергия включена в рассматриваемые характеристики.

3.1.4 Windowing

Преобразование Фурье конечных импульсов хорошо работает с сигналами, которые могут быть продолжены непрерывно на бесконечность (в частности, равны нулю вне некоторого отрезка). Однако, конечно, не все сигналы имеют такой вид. Для того, чтобы преобразование работало лучше, применяются различные оконные функции, некоторым образом преобразующие сигнал для достижения указанной выше цели.

Различные оконные функции дают различный результат. Например, прямоугольная функция позволяет хорошо различать в сигнале синусоиды схожей длины волны, однако не позволяет различать синусоиды с сильно различающимися длинами волн. Существуют и функции, обладающие обратным эффектом. В данной работе применяется оконная функция Хэмминга, задаваемая уравнением $w(n) = \alpha - \beta \cos\left(\frac{2\pi n}{N-1}\right)$, где наилучшими параметрами являются $\alpha \approx 0.54, \beta \approx 0.46$.

3.1.5 Spectrum

Данный алгоритм получает частотный спектр сигнала с помощью преобразования Фурье. Частотный спектр $S(w)$ сигнала $x(t)$ получается из выражения $\int\limits_{-\infty}^{\infty} |x(t)|^2 dt = \int\limits_{-\infty}^{\infty} |F[x](w)|^2 dw = \int\limits_{-\infty}^{\infty} S(w) dw$ (теорема Парсеваля).

3.1.6 Centroid

Спектральный центроид определяется выражением $C = \frac{\sum\limits_{n=1}^{N-1} f(n)x(n)}{\sum\limits_{n=1}^{N-1} x(n)}$, где $x(n)$ — средняя амплитуда n-ой компоненты сигнала, а $f(n)$ — его главная частота. В данной работе N принято равным единице, то есть вычисляется произведение средней амплитуды сигнала на его главную частоту. Эта характеристика связывает частоту и амплитуду и может быть интерпретирована как тембр звука, поэтому включена в рассматриваемые.

3.1.7 CentralMoments и DistributionShape

Центральный момент — статистический термин. n-ый центральный момент распределения X с плотностью распределения f определяется выражением $\mu_n = \mathbb{E}[(X - \mathbb{E}(X))^n] = \int\limits_{-\infty}^{\infty} x^n f(x) dx$. Первые центральные моменты (нулевой равен единице как интеграл плотности распределения) имеют интерпретации:

1. $\mu_1 = \int\limits_{-\infty}^{\infty} x f(x) dx = 0$
2. $\mu_2 = \int\limits_{-\infty}^{\infty} x^2 f(x) dx = \sigma^2$ — разброс случайной величины X
3. μ_3 является характеристикой симметрии распределения
4. μ_4 показывает, насколько ярко выражена вершина распределения в окрестности среднего

Алгоритм DistributionShape возвращает вышеуказанные $\mu_0, \mu_1, \mu_2, \mu_3$ и μ_4. Они являются важными характеристиками спектра, и также будут использованы для классификации.

3.1.8 MFCC

MFCC расшифровывается как Mel-frequency Cepstrum Coefficients и представляет собой амплитуды спектра, полученного в результате следующей процедуры:

1. Получить преобразование Фурье $F[x]$ исходного сигнала $x(t)$.
2. Перевести частоты спектральных компонент в шкалу Мела по формуле $m = 1127 \ln(1 + \frac{w}{700})$

3. Взять от амплитуд логарифмы

4. Взять косинус-преобразование Фурье от полученных значений

Шкала Мела является результатом исследований в области психоакустики, и характеризует схожесть звуковых частот при восприятии их на значительном расстоянии от источника. Данный алгоритм обобщает информацию в спектре, уменьшая её объём с сохранением основной информации о композиции, и является основным в данной работе.

3.2 Алгоритмы scikit-learn и XGBoost

3.2.1 Preprocessing

Тестирование будет производиться на отложенной выборке, поэтому при разбиении необходимо провести стратификацию выборки — иначе возможно недообучение на одном из классов.

Кроме того, многие признаки имеют различный масштаб. Приведем их к одинаковому масштабу, вычтя из каждого элемента среднее по столбцу (таким образом уменьшив смещение) и поделив на стандартное отклонение (таким образом отнормировав разброс).

3.2.2 RandomForest

RandomForest представляет собой композицию базовых алгоритмов (простых решающих деревьев), где ответ композиции получается как среднее ответов базовых алгоритмов. Алгоритм работает особенно хорошо, когда корреляция между ответами различных базовых алгоритмов низкая. Для этого максимально рандомизируется подвыборка, на которой обучается каждый базовый алгоритм (с помощью бустрапа, например). Параметры композиции — число базовых алгоритмов, максимальная глубина каждого дерева и минимальное число объектов в листьях — подбираются по сетке, также стохастическим методом.

Этот алгоритм был выбран потому, что он даёт высокое качество при неизвестной (или неявной, как в данном случае) структуре данных.

3.2.3 XGBoost

XGBoost также является композиций алгоритмов, однако она применяются последовательно к результатам друг друга. При этом оптимизация производится приближением ответа следующего алгоритма к разности истинного ответа и ответа предыдущего алгоритма. Параметры композиции также подбираются по сетке.

Этот алгоритм был выбран потому, что он даёт более высокое качество при аналогичных RandomForest свойствам.

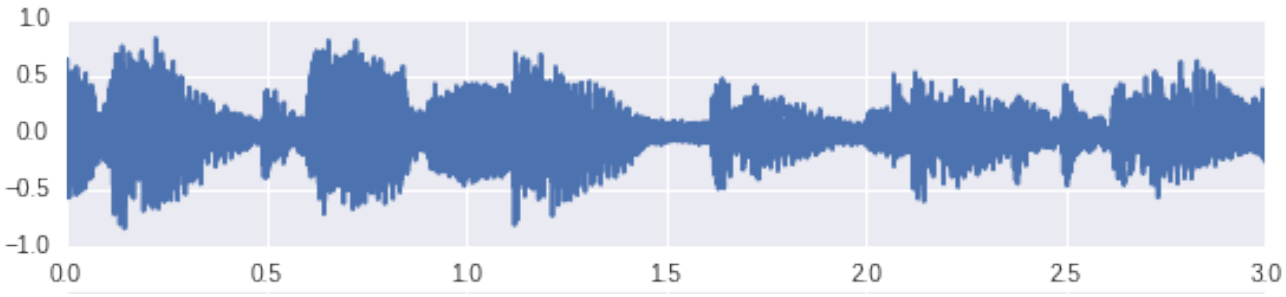
3.2.4 Multi-label classification

Предыдущие алгоритмы рассматривались как классификаторы, выдающие единственный ответ — метку класса, которому с наибольшей вероятностью принадлежит данный объект. Однако, если предположить, что данные могут принадлежать нескольким классам, можно выдавать n бинарных ответов по числу классов. Такой подход называется Multi-label classification, и может быть реализован стратегией One-vs-Rest. Базовым алгоритмом для этого классификатора выберем RandomForest в силу названных причин.

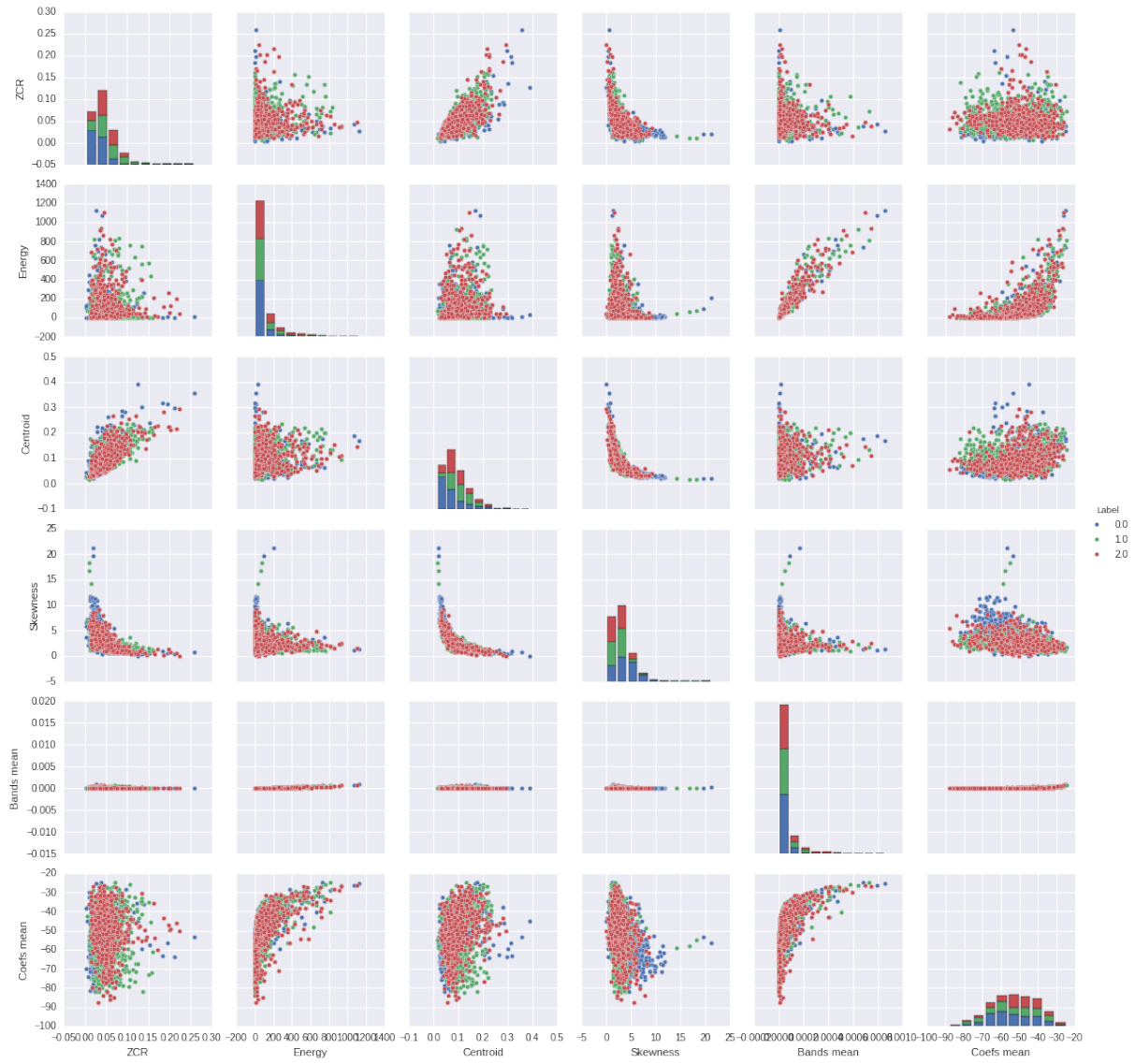
4 Практический эксперимент

Как уже было сказано, используется датасет IRMAS, а именно IRMAS-Training, состоящий из трехсекундных wav-файлов с указанными метками инструментов. Из всех десяти предлагаемых инструментов выбраны скрипка, фортепиано и труба как наиболее яркие представители соответственно струнных, клавишных и духовых инструментов. Для воспроизводимости результатов используем везде random_state = 0.

Посмотрим на результат открытия одного из wav-файлов экстрактором MonoLoader:

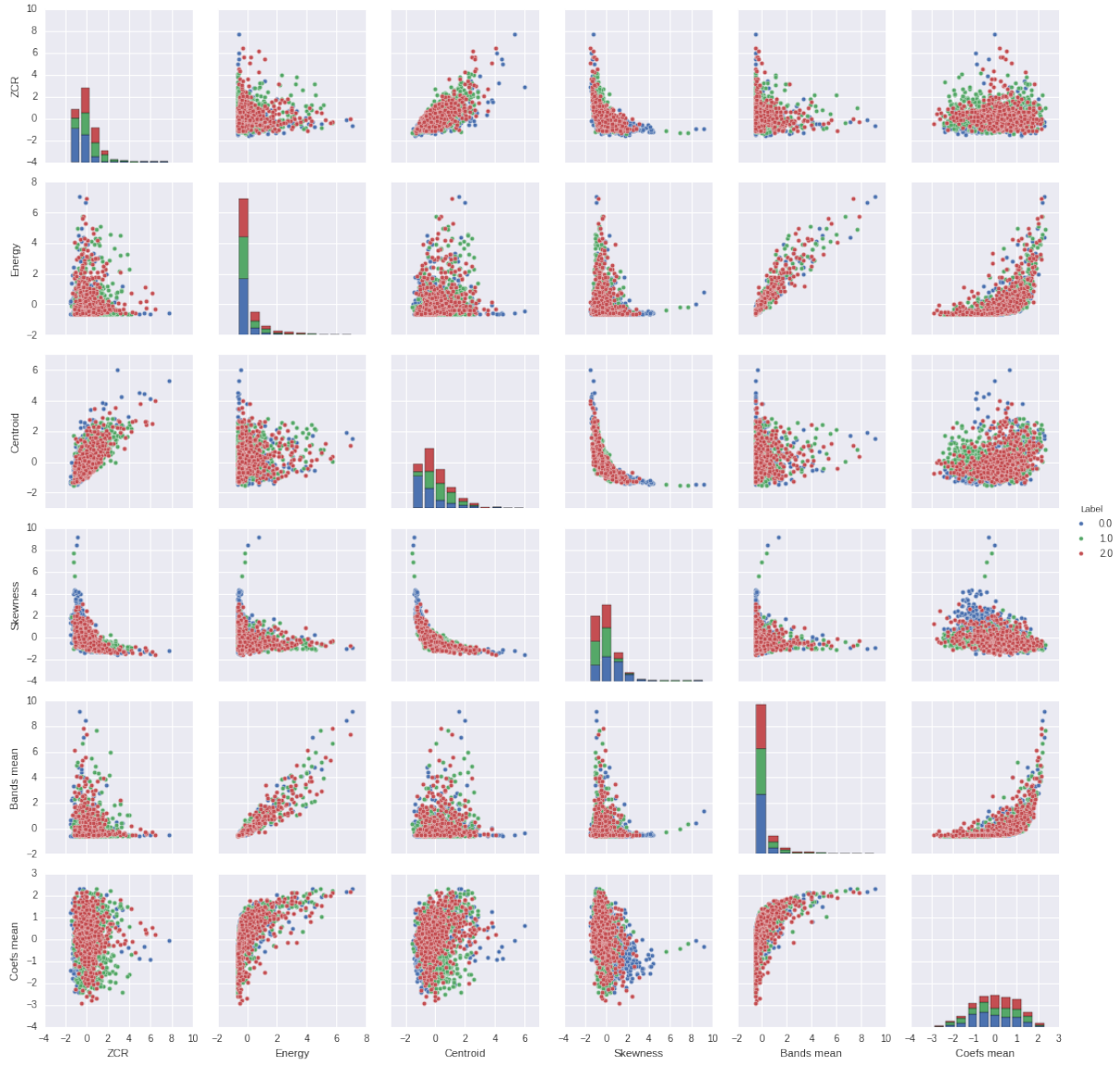


Произведем извлечение признаков, указаных в алгоритмической справке. Рассмотрим распределение данных по парам признакам (здесь не учтены непосредственно значения признаков MFCC в силу их большого числа, но учтены их средние, для полноты картины):



Видно, что классы сильно пересекаются. Из этих соображений и возникла идея использовать классификацию с множеством ответов.

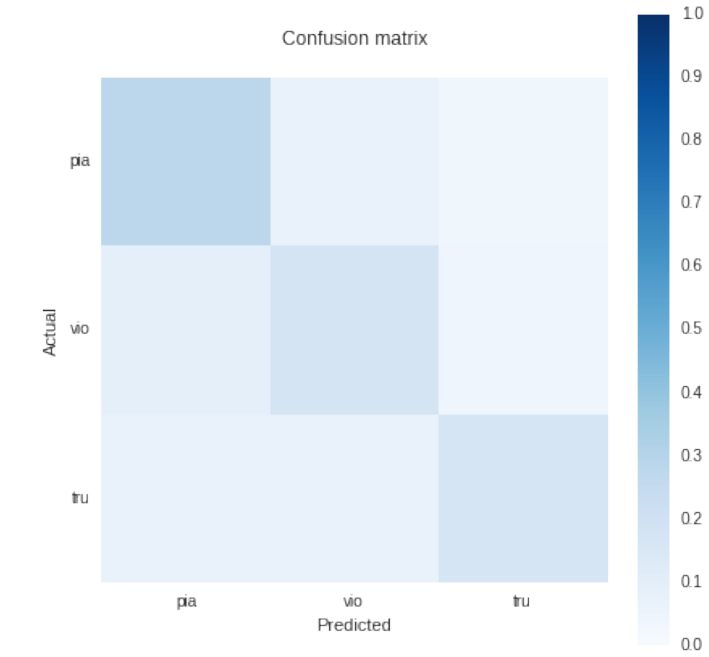
После генерации отложенной стратифицированной выборки и нормирования с помощью StandardScaler, распределение данных по парам признакам выглядит следующим образом:



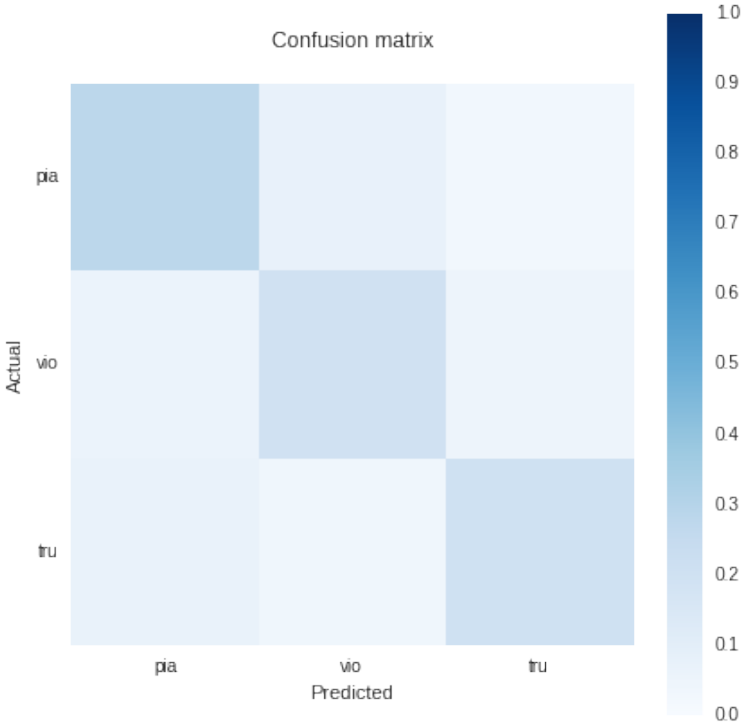
Начальной точкой исследования примем логистическую регрессию, как алгоритм, предсказывающий вероятности принадлежности ко всем классам. Кроме того, используя регуляризацию в норме ℓ_1, можно произвести отбор признаков. Оптимальный коэффициент при регуляризаторе равен 0.6 и получен поиском по сетке. При этом использовался регуляризатор ℓ_1, поэтому веса некоторых признаков оказались обнулены. Это признаки Energy, MFCC8, MFCC39. Для Energy причина, вероятно, в том, что он оказался линейно зависим от других признаков. Наиболее коррелирующими с итоговой переменной оказались признаки ZCR, Centroid, MFCC 15, 16, 18, 19, 21, 40, 43.

Для оценки эффективности работы алгоритма будем пользоваться метриками точности (accuracy, является долей правильных ответов) и кросс-энтропии (log_loss, вычисляется по формуле $logloss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij}\log p_{ij}$, где N — число объектов, M — число классов, y_{ij} — индикатор принадлежности объекта i классу j, p_{ij} — предсказанная вероятность принадлежности объекта i классу j). Для логистической регрессии они равны соответственно 0.624 и 0.991.

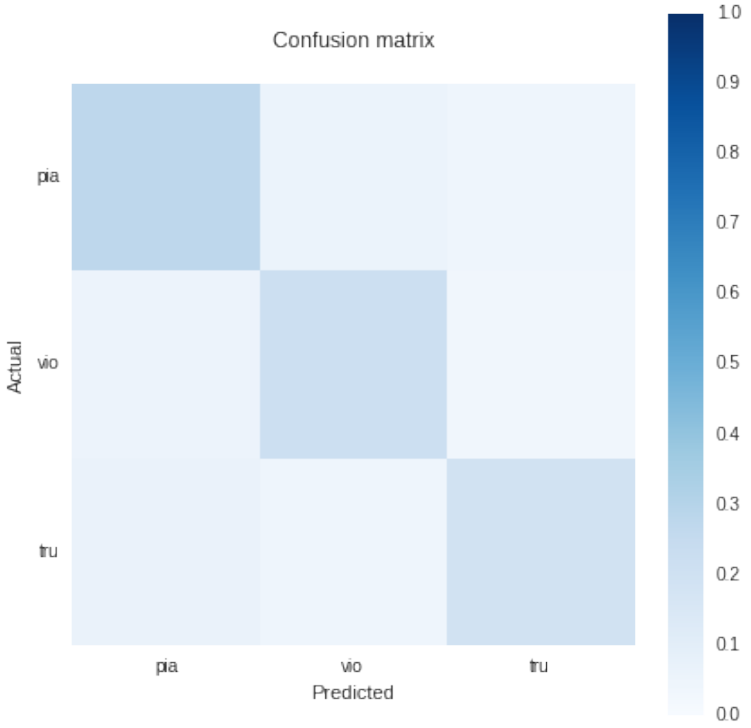
Посмотрим на ошибки классификатора, визуалилзируя Confusion Matrix:



После этого обучен алгоритм RandomForest. Оптимальные параметры следующие: 43 базовых алгоритма, максимальная глубина дерева 12, не более 3 объектов в листьях. Точность классификации 0.679, кросс-энтропия 0.753. Видно, что точность алгоритма значительно возросла, и функционал потерь уменьшился. Для иллюстрации вновь приведем Confusion Matrix:



Далее был обучен XGBoost. Оптимальные параметры подобраны стохастическим поиском по сетке: 46 базовых алгоритмов, максимальная глубина дерева 7. Рассмотрим качество алгоритма. Точность классификации составила 0.69, кросс-энтропия — 0.739. Вновь заметно улучшение качества. Наконец, посмотрим на Confusion Matrix:



Наконец, была испробована стратегия классификации с вектором ответов, так как отдельные классификаторы One-vs-Rest давали большое качество (до 0.80). Точность по стандартной метрике составила приблизительно 0.516. Это объясняется тем, что accuracy_score требует точного совпадения предсказанного ответа с истинным и не совсем подходит для данной задачи. Для более щадящей оценки качества была предложена метрика, дающая по 0.33 балла за совпадение в каждой компоненте. Затем берется отношение. В такой метрике точность классификатора составила приблизительно 0.775.

5 Реальные данные

Также в датасете IRMAS присутствуют не столь «идеальные»композиции (для тестирования). Протестируем полученный классификатор на нескольких композициях из предложенных.

Было выбрано пять композиций, имеющих разные ответы:

1. Фортепиано + труба
2. Фортепиано
3. Электрогитара
4. Фортепиано + труба
5. Фортепиано + скрипка

Используем несколько другой подход, чем при построении модели. В качестве основного инструмента используем только multilabel RandomForest (в силу особенностей выбранных композиций), обученный на всей обучающей выборке.

В результате, вторая и третья композиции были классифицированы алгоритмом корректно, а в четвертой и пятой совершена одна ошибка. Итого, по строгой метрике точность составила 0.4, по собственной метрике — 0.726.

6 Выводы и результаты

В результате проведенного исследования получен качественный инструмент классификации музыкальных композиций по инструментам, применимый на практике. В ходе работы изучены некоторые алгоритмы, используемые в работе с временными рядами, имеющими смысл музыкальной композиции, а также проведен сравнительный анализ наиболее популярных на данный момент алгоритмов классификации. Выдвинута гипотеза о необходимости использования подхода с множественными метками классов, и получен практический результат, согласующийся с ожиданиями.

7 Список источников

[1] «A Large Set of Audio Features for Sound Description», Geoffroy Peeters, 2004
[2] Документация библиотек Essentia и scikit-learn
[3] Wikipedia и Kaggle Wiki