Федеральное государственное бюджетное образовательное учреждение высшего образования

**«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»**

**Департамент анализа данных и машинного обучения**

Пояснительная записка к курсовой работе

по дисциплине “Технологии анализа данных и машинное обучение”

на тему:

«Машинное обучение в задачах акустической классификации типов дорожных условий для беспилотных автомобилей»

Выполнил:

студент группы ПМ21-5 факультета информационных технологий и анализа больших данных

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Астахов В.И.

Научный руководитель:

Старший преподаватель, Фурлетов Ю.М.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2023 г

**Введение**

Все файлы с кодом выложены в открытый репозиторий на github по ссылке: <https://github.com/LiSenPie/Course_work>

**Задача**

Собрать датасет для решения задачи классификации типов дорожных условий для беспилотных автомобилей, обработать данные, получить из них признаки для обучения моделей и обучить 7 моделей машинного обучения для классификации типов дорожных условий и выбрать лучшую из них.

**Цель**

Продемонстрировать владение основными навыками работы с методами машинного обучения с учителем и без учителя, владение основными инструментальными средствами библиотек языка программирования Python, методами и приемами подготовительного и описательного анализа данных, средствами визуализации данных, использования и усовершенствования обучаемых моделей, умение делать выводы из проведенного анализа.

**Этапы**

1. Сбор датасета в соответствии с темой курсовой работы, а именно аудиофайлов проезда автомобиля в разных дорожных условиях
2. Описание данных
3. Предварительный анализ, обработка и очистка данных, преобразование данных в числовые признаки
4. Получений метрик данных для обучения моделей
5. Описание данных метрик
6. Удаление ненужных столбцов, создание меток, масштабирование столбцов признаков и разделение данных на обучающий и тестовый наборы в соотношении 80/20
7. Обучение моделей и тестирование их точности. Улучшение данных моделей по средствам подбора наилучших гиперпараметров с помощью grid search

***Сбор датасета***

Для сбора данных для составления датасета мною были записаны звуки проезда автомобиля в разных дорожных условиях (сухая дорога, мокрая дорога и стоячая машина с заведенным двигателем). Микрофон находился на задней правой арке автомобиля.

***Описание данных***

Я получил звуковые дорожки различной длины формата wav. Более подробное описание количественных характеристик и графиков звуковых дорожек находятся в файле KR1.

***Предварительный анализ, обработка и очистка данных, преобразование данных в числовые признаки***

Предварительный анализ данных проведен в файле KR1. Я загружаю первые 20 минут каждого аудиофайла, что было сделано для эффективности и быстроты выполнения кода, а также для того, чтобы сбалансировать данные и данных по одному из типов дороги не было слишком много, так как все аудиодорожки имеют разный размер, а для мокрой дороги имеется два аудиофайла. После этого я делю каждый аудиофайл на равные отрезки по 1 секунде, чтобы в итоге получить достаточное кол-во данных для обучения моделей.

***Получение метрик данных для обучения модели***

Я использую следующие метрики аудиофайлов для обучения моделей машинного обучения: средние значения MFCC, спектральный центроид, частоту пересечения нуля, частоты цветности и спад спектра. Получаю я их с помощью методов библиотеки librosa. После этого я загружаю полученные данные в csv файл, из которого в дальнейшем планирую создать датаферйм.

***Описание данных метрик***

1. *MFCC* - это коэффициенты, полученные в результате преобразования Фурье над окном аудио-сигнала, переведенного в мел-шкалу. Они показывают, какие частоты присутствуют в сигнале, и насколько они важны для восприятия человеком.
2. *Cпектральный центроид* - то средневзвешенное значение частотного спектра звукового сигнала. Он отображает среднюю частоту звука, который мы слышим. Частота, которая ближе к спектральному центроиду, оказывает наибольшее влияние на восприятие высоты звука.
3. *Частота пересечения нуля* - это частота, на которой амплитуда звукового сигнала пересекает ось абсцисс. Эта характеристика используется для измерения количества изменений знака в сигнале. Чем больше частота пересечения нуля, тем более шумным является сигнал.
4. *Частоты цветности* - это характеристика звукового сигнала, которая отображает соотношение мощности высокочастотных компонентов к мощности низкочастотных компонентов в спектре сигнала. Частоты цветности помогают определить, как звук будет звучать на разных устройствах воспроизведения и как его можно оптимизировать для конкретных условий воспроизведения.
5. *Спад спектра* - это изменение амплитуды звукового сигнала в зависимости от частоты. Он может показать, как звуковой сигнал затухает на высоких или низких частотах. Спад спектра используется для анализа спектрального содержания звука и может помочь определить тип звука и его происхождение.

***Удаление ненужных столбцов, создание меток, масштабирование столбцов признаков и разделение данных на обучающий и тестовый наборы***

Я удаляю столбец ‘filename’, так как он не несет смысла для обучения моделей, после этого создаю метки классов при помощи LabelEncoder, после этого масштабирую столбцы признаков при помощи StandardScaler и разделяю данные на обучающие и тестовые выборки с помощью train\_test\_split в соотношении 80/20.

***Обучение моделей и тестирование их точности. Улучшение данных моделей по средствам подбора наилучших гиперпараметров с помощью grid search***

Заключительным этапом я обучаю модели и тестирую их с выводом характеристик каждой модели. Также применяю кроссвалидацию к каждой модели, после чего ищу наилучшие гиперпараметры при помощи grid search

**Желаемый результат**

Использовать полученные в течении семестра знания на практике и получить достаточно точные модели машинного обучения для определения типов дорожных условий на имеющихся данных и выяснить, какая из них является лучшей

**Метрики оценки качества моделей**

F1-score является метрикой оценки качества работы модели машинного обучения, используемой для бинарной классификации. Она является гармоническим средним между точностью (precision) и полнотой (recall).

Точность (precision) - это доля верно предсказанных положительных примеров относительно всех предсказанных положительных примеров.

Полнота (recall) - это доля верно предсказанных положительных примеров относительно всех положительных примеров в тестовом наборе данных.

F1-мера вычисляется как гармоническое среднее между точностью и полнотой:

F1-мера представляет собой баланс между точностью и полнотой и может быть использована для оценки качества работы модели, особенно в случаях, когда наблюдается дисбаланс классов. Например, если положительные примеры в тестовом наборе данных составляют только 10%, то модель может легко достичь высокой точности, но при этом не достигать высокой полноты. В этом случае, F1-мера может быть использована для оценки качества работы модели более объективно, чем точность или полнота по отдельности.

Таким образом, F1-мера является важной метрикой для оценки качества работы модели в задачах бинарной классификации, особенно в случаях с дисбалансом классов, что может наблюдаться в моем случае, так как я имею два файла со звуком проезда по влажной дороге и всего по одному файлу с остальными типами дорожных условий.

Дисбаланс в моем датасете нельзя назвать критичным, поэтому в качестве основной метрики я буду использовать accuracy.

**Методы машинного обучения**

В своей работе я использую следующие методы машинного обучения для классификации типов дорожных условий: Метод опорных векторов (SVM), Gaussian Naive Bayes, K-Nearest Neighbors, Decision Tree Classifier, Случайный лес (Random Forest Classifier), Gradient Boosting Classifier

В ходе работы я обучил каждую модель, вывел ее accuracy, вывел classification\_report, который хранит в себе такие параметры как precision, recall, f1-score и support. Метрики precision, recall, f1-score были описаны мной выше. support показывает, сколько строк с параметрами из датасета, соответствующих каждому классу, оказалось в тестовой выборке. Также я вывожу матрицу ошибок с помощью confusion\_matrix, по которой можно увидеть, с классификацией какого класса возникают ошибки в ходе применения модели.

После обучения и получения метрик качества, я провожу кросс-валидацию каждой модели, чтобы получить более объективные значения accuracy. Кол-во фолдов в каждом случае одинаково и равно 5.

И в качестве заключительного этапа я провожу grid search лучших гиперпараметров для каждой модели при помощи GridSearchCV, после чего вывожу лучшие найденные значения и лучшую accuracy, полученную в ходе поиска.

**Выводы**

В результате проделанной мной работы я выяснил, что все модели отлично справляются с поставленной задачей. Тяжело выделить безоговорочного лидера, но можно выделить однозначных аутсайдеров. В моем случае это Gaussian Naive Bayes, так как его точность является самой маленькой среди всех моделей (лучшая точность при grid search составила всего 0.876) и Gradient Boosting Classifier, так как времени на ее обучение и тестирование уходит гораздо больше, чем на другие (14.9 s без кросс-валидации, 16 s с кросс-валидацией и 5min 42s при grid search), но стоит отметить, что у нее отличная точность (0.977 при обычном обучении, 0.949 при кросс-валидации и 0.985 при grid search), хоть она и хуже лучшей модели.

Лучшею модель выбрать тяжело, так как можно сказать, что их оказалось две, а именно метод опорных векторов (SVM) (точность 0.967 при обычном обучении, 0.951 при кросс-валидации и 0.994 при grid search) и K-Nearest Neighbors (точность 0.985 при обычном обучении, 0.945 при кросс-валидации и 0.989 при grid search). При этом на их обучение и тестирование ушло сравнительно немного времени. В случае SVM это 80.2 ms при обычном обучении и тестировании, 3.9 s с кросс-валидацией и 2.16 s при grid search, а для K-Nearest Neighbors это 232 ms при обычном обучении и тестировании, 185 ms с кросс-валидацией и 899 ms при grid search. Но, поскольку одной из задач данной работы является выбор лучше модели, я бы отдал это звание методу опорных векторов (SVM), так как он показал лучшую точность после улучшения путем grid search и маленькое время обучения и тестирования модели.

Также хочется отметить, что точность всех моделей получилась весьма высокой, но это может измениться, если добавить еще несколько классов в наш датасет.