Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра автоматизации обработки информации (АОИ)

Вариант 2

Отчёт к Лабораторной работе №2

по дисциплине «Нейронные сети и их применение»

Выполнил:

Студент группы 422–M1

\_\_\_\_Белоус Г.В.

Принял:

К.т.н., Доцент кафедры АОИ

\_\_\_\_Аксёнов С.В.

Оглавление

[1 Введение 3](#_Toc130821209)

[1.1 Цель лабораторной работы 3](#_Toc130821210)

[1.2 Задание на лабораторную работу 3](#_Toc130821211)

[2 Ход выполнения работы 5](#_Toc130821212)

[2.1 Построение классифицирующих моделей с использованием GridSearch для алгоритмов RandomForest, LogisticRegression и XGBoost. 5](#_Toc130821213)

[2.2 Визуализация значения важности признаков для моделей, использующих деревья и логистической регрессии для сбалансированных и первоначальной выборок 7](#_Toc130821214)

[2.3 Выполнение K-блочной стратифицированной проверку для указанных алгоритмов со значениями гиперпараметров, полученных ранее 10](#_Toc130821215)

[2.4 Визуализация результаты ROC-анализа. Расчёт среднего ROC и отклонение 10](#_Toc130821216)

[3 Вывод 12](#_Toc130821217)

# Введение

## 1.1 Цель лабораторной работы

Построения бинарных классификаторов, подбора гиперпараметров моделей и анализа качества работы классифицирующих моделей.

## **1.2 Задание на лабораторную работу**

Написать программу на Python, которая загружает набор данных и выполняет задачи построения бинарных классификаторов, подбора гиперпараметров моделей и анализа качества работы классифицирующих моделей.

1. Выполнить предварительную обработку набора данных.

2. Построить классифицирующие модели с использованием GridSearch для алгоритмов RandomForest, LogisticRegression и XGBoost. Провести эксперименты с регуляризацией. Выполнить ROC-анализ и визуализировать результаты анализа.

3. Визуализировать значения важности признаков для моделей, использующих деревья и логистической регрессии для сбалансированных и первоначальной выборок.

4. Выполнить K-блочную стратифицированную проверку для указанных алгоритмов с значениями гиперпараметров, полученных ранее.

5. Получить усреднённую важность признаков для наборов моделей, использующих деревья и логистической регрессии

6. Визуализировать результаты ROC-анализа. Рассчитать среднюю ROC и отклонение (standard deviation).

7. Напишите короткое заключение о наиболее интересных зависимостях, которые Вы обнаружили в результате выполнения работы.

Содержание отчета.

1. Титульный лист
2. Задача по полученному варианту (выбор варианта осуществляется по последнему номеру в зачетной книжке, если он равен 0 или 5 – то берем вариант 1, если 2 или 6 – вариант 2, если 3 или 7 то вариант 3, если 4 или 8 - вариант 4, если 5 – вариант 5, если 9 – вариант 6)
3. Подобранные гиперпараметры для каждого алгоритма
4. Графики, полученные в ходе анализа набора данных
5. Вывод

# Ход выполнения работы

Датасет, данные из которого будут обрабатываться в данной работе, представляет из себя набор атрибутов для прогнозирования типа лесного покрова на основе картографических переменных. Исходя из варианта все записи из столбца Cover\_Type со значением “3”, получили новое значение “0”, а остальные “1”.

Для сокращения времени расчётов была взята выборка величиной 10000 записей.

## Построение классифицирующих моделей с использованием GridSearch для алгоритмов RandomForest, LogisticRegression и XGBoost.

GridSearch – позволяет создавать и обучать модели разных алгоритмов с разными гиперпараметрами, а потом выбрать лучшую для предсказания необходимых значений.

Для построения классифицирующих моделей были выбрани алгоритмы RandomForest, LogisticRegression и XGBoost, имеющие различные гиперпараметры, которын и подбирались с помощью GridSearch. На картинках 2.1, 2.2 и 2.3 показаны оптимальные гиперапраметры и точность предсказания типа лесного покрова для трех алгоритмов:

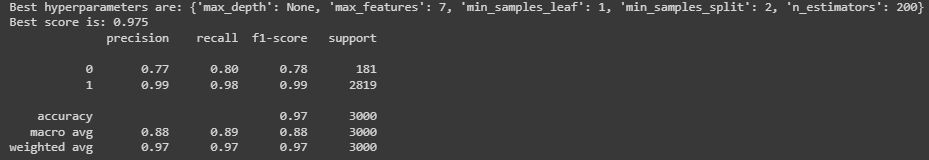


Рисунок 2.1 – Оптимальные гиперпараметры для алгоритма RandomForest

Изображение выглядит как календарь

Автоматически созданное описание

Рисунок 2.2 – Оптимальные гиперпараметры для алгоритма LogisticRegression

Изображение выглядит как текст, экран, табло

Автоматически созданное описание

Рисунок 2.3 – Оптимальные гиперпараметры для алгоритма XGBClassifier

Видно, что меньшую точность показывает LogisticRegression.

## Визуализация значения важности признаков для моделей, использующих деревья и логистической регрессии для сбалансированных и первоначальной выборок

В RandomForest важность признаков представляет собой вклад каждого признака в диапазоне от 0 до 1, где чем больше значение, тем выше важность данного признака. При этом LogisticRegression важность признака выражается в оценочном коэффициенте, который может принимать как положительное, так и отрицательное значение.

За сбалансированность выборки отвечает параметр stratify в методе train\_test\_split, в зависимости от которого выборки получаются с одинаковой долей значений.

На изображения 2.4, 2.5 и 2.6, 2.7 показаны значения важности признаков для моделей RandomForest и LogisticRegression со сбалансированной и первоначальной выборкой:

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 2.4 – Важность признаков для RandomForest и сбалансированных выборок

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 2.5 – Важность признаков для RandomForest и несбалансированных выборок

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2.6 – Важность признаков для LogisticRegression и сбалансированных выборок

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2.7 – Важность признаков для LogisticRegression и сбалансированных выборок

Как видно из полученных изображения, на важность признаков для алгоритма RandomForest не влияет сбалансированность выборок, в отличии от LogisticRegression.

## Выполнение K-блочной стратифицированной проверку для указанных алгоритмов со значениями гиперпараметров, полученных ранее

K-блочная кросс-валидация – процедура используемая для оценки навыков моделей машинного обучения на новых данных. Осуществлялось с помощью метода StratifiedKFold() на моделях с оптимальными гиперпараметрами, результаты показаны на 2.8:



Рисунок 2.8 – Средняя площадь под ROC кривой для алгоритмов

## Визуализация результаты ROC-анализа. Расчёт среднего ROC и отклонение

ROC-кривая (англ. Receiver Operating Characteristic curve) — это график, который используется для оценки качества бинарной классификации.

ROC-кривая показывает, как изменяется отношение количества верно классифицированных положительных объектов (True Positive Rate, TPR) к количеству неверно классифицированных отрицательных объектов (False Positive Rate, FPR) при изменении порогового значения классификатора.

TPR — это доля положительных объектов, которые были правильно классифицированы, то есть отношение числа верно классифицированных положительных объектов к общему числу положительных объектов.

FPR — это доля отрицательных объектов, которые были неправильно классифицированы, то есть отношение числа неверно классифицированных отрицательных объектов к общему числу отрицательных объектов.

ROC-кривая позволяет наглядно сравнить различные классификаторы между собой и выбрать наиболее эффективный в зависимости от задачи. В качестве метрики качества классификации на ROC-кривой используется площадь под кривой (Area Under Curve, AUC). Чем ближе значение AUC к 1, тем лучше качество классификации.

На рисунке 2.9 визуализированы результаты ROC-анализа:

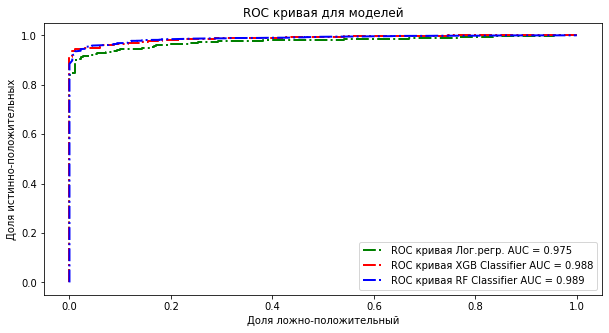


Рисунок 2.9 – Результаты ROC-анализа

# Вывод

В ходе выполнения лабораторной работы были построены бинарные классификаторы на основе RandomForest, LogisticRegression и XGBoost, подобраны гиперпараметры моделей и проведен анализ качества работы классифицирующих моделей.

Инструмент GridSearch позволяет не подбирать оптимальные гиперпараметры для модели самому, так как сам делает этот перебор, что позволяет построить наиболее подходящую модель под имеющийся набор данных.

Как видно из рисунков 2.4 и 2.5, сбалансированность выборок не влияет на классификацию алгоритмом RandomForest, что нельзя сказать про LogisticRegression, чьи коэффициенты меняются в зависимости от сбалансированности выборок.

По результату проведенного ROC-анализа, было определенно, что самым точным для выборки в 10000 записей оказался алгоритм RandomForest, возможно при обработки всего датасета размером 500000 результат мог быть и другим, но данное исследование не проводилось из-за высокого времени расчетов.

Листинг программы находится в соответствующей папке Lab2 по ссылке на репозиторий: https://github.com/peremichka256/NSiIP