МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА №  43

ОТЧЁТ

ЗАЩИЩЁН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

### ст.пр.                                   Поляк М.Д.

должность, уч. Степень, звание   подпись, дата           инициалы, фамилия

ОТЧЁТ О ЛАБОРАТОРНОЙ РАБОТЕ №3.

Применение методов бинарной классификации.

по курсу: Основы машинного обучения

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. 4136                                                                                Бобрович Н. С.

                                                                         подпись, дата                      инициалы, фамилия

Санкт-Петербург 2024

1. **Цель работы:**

Знакомство с алгоритмами классификации в машинном обучении.

1. **Индивидуальное задание:**

Задание 1.

Откройте Jupyter-ноутбук [classification\_assignment.ipynb](https://github.com/suai-mlb-2024/mlb-task3-OskolockKoli/blob/main/classification_assignment.ipynb) в этом репозитории. Скопируйте путь в адресной строке браузера. Перейдите в [Google Colab](https://colab.research.google.com/), в меню выберите "Файл" -> "Открыть ноутбук", в открывшемся окне слева выбрать "GitHub", затем:

* вставить в поле для поиска скопированный URL;
* поставить галочку "Показывать личные хранилища" ("Include private repos");
* и нажать на иконку с лупой. При необходимости разрешить Colab доступ к аккаунту GitHub, если откроется новое окно с таким приглашением. Среди результатов поиска выбрать classification\_assignment.ipynb и приступить к выполнению задания.

1. **Ход работы:**

Задание 1:

**Фрагмент кода из задания 1.**

### BEGIN YOUR CODE

#Второй алгоритм: SVM с ядром RBF

# Загрузить данныеdf = pd.read\_csv('Train.csv')

# Определить целевую переменную и признакиX = df.drop(['Reached.on.Time\_Y.N'], axis=1)y = df['Reached.on.Time\_Y.N']

# Преобразовать категориальные признаки в числовыеordinal\_encoder = OrdinalEncoder()columns\_to\_encode = ['Warehouse\_block', 'Mode\_of\_Shipment', 'Product\_importance', 'Gender']X[columns\_to\_encode] = ordinal\_encoder.fit\_transform(X[columns\_to\_encode])

# Разделить данные на обучающий и тестовый наборыX\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Создать и обучить SVM-классификатор с ядром RBFsvm\_classifier = SVC(kernel='rbf') # Используем ядро RBFsvm\_classifier.fit(X\_train, y\_train)

# Сделать предсказания на тестовом набореy\_pred = svm\_classifier.predict(X\_test)

# Вычислить метрику Accuracyalgo\_2\_accuracy = accuracy\_score(y\_test, y\_pred)

# Сохранение результатаprint("Accuracy of SVM with RBF kernel:", algo\_2\_accuracy)

### END YOUR CODE

**Результат выполнения кода из задания 1.**

Accuracy of SVM with RBF kernel: 0.6918181818181818

**Пояснения к коду из задания 1.**

Разработал модель машинного обучения, которая позволяет предсказывать целевое значение на основе различных характеристик товара и покупателя. Наша модель показала точность около 69%, что означает, что она правильно предсказывает результат в 69% случаев.

Текущая модель демонстрирует удовлетворительную точность, однако виден потенциал для улучшения за счет добавления дополнительных данных.

В будущем планируется провести тестирование других алгоритмов машинного обучения и оптимизацию гиперпараметров для повышения точности прогнозов. Также важно учитывать возможные ограничения данных и продолжать тестирование модели на новых наборах данных, чтобы убедиться в её надежности.

1. **Выводы:**

Познакомился с алгоритмами классификации в машинном обучении.