Московский авиационный институт
(национальный исследовательский университет)

Факультет информационных технологий и прикладной математики
Кафедра вычислительной математики и программирования

Лабораторная работа №1 по курсу
«Искусственный интеллект (Машинное обучение»

Студент: Суханов Е. А.
Группа: М8О-406Б-19
Преподаватель: Ахмед Самир Халид
Дата:
Оценка:
Подпись:

Москва, 2022

Описание

Задача:
1) реализовать следующие алгоритмы машинного обучения: Linear/ Logistic Regression, SVM, KNN, Naive Bayes в отдельных классах

2) Данные классы должны наследоваться от BaseEstimator и ClassifierMixin, иметь методы fit и predict

3) Вы должны организовать весь процесс предобработки, обучения и тестирования с помощью Pipeline

4) Вы должны настроить гиперпараметры моделей с помощью кросс валидации, ывести и сохранить эти гиперпараметры в файл, вместе с обученными моделями

5) Проделать аналогично с коробочными решениями

6) Для каждой модели получить оценки метрик: Confusion Matrix, Accuracy, Recall, Precision, ROC_AUC curve

7) Проанализировать полученные результаты и сделать выводы о применимости моделей

8) Загрузить полученные гиперпараметры модели и обученные модели в формате pickle на гит вместе с jupyter notebook ваших экспериментов

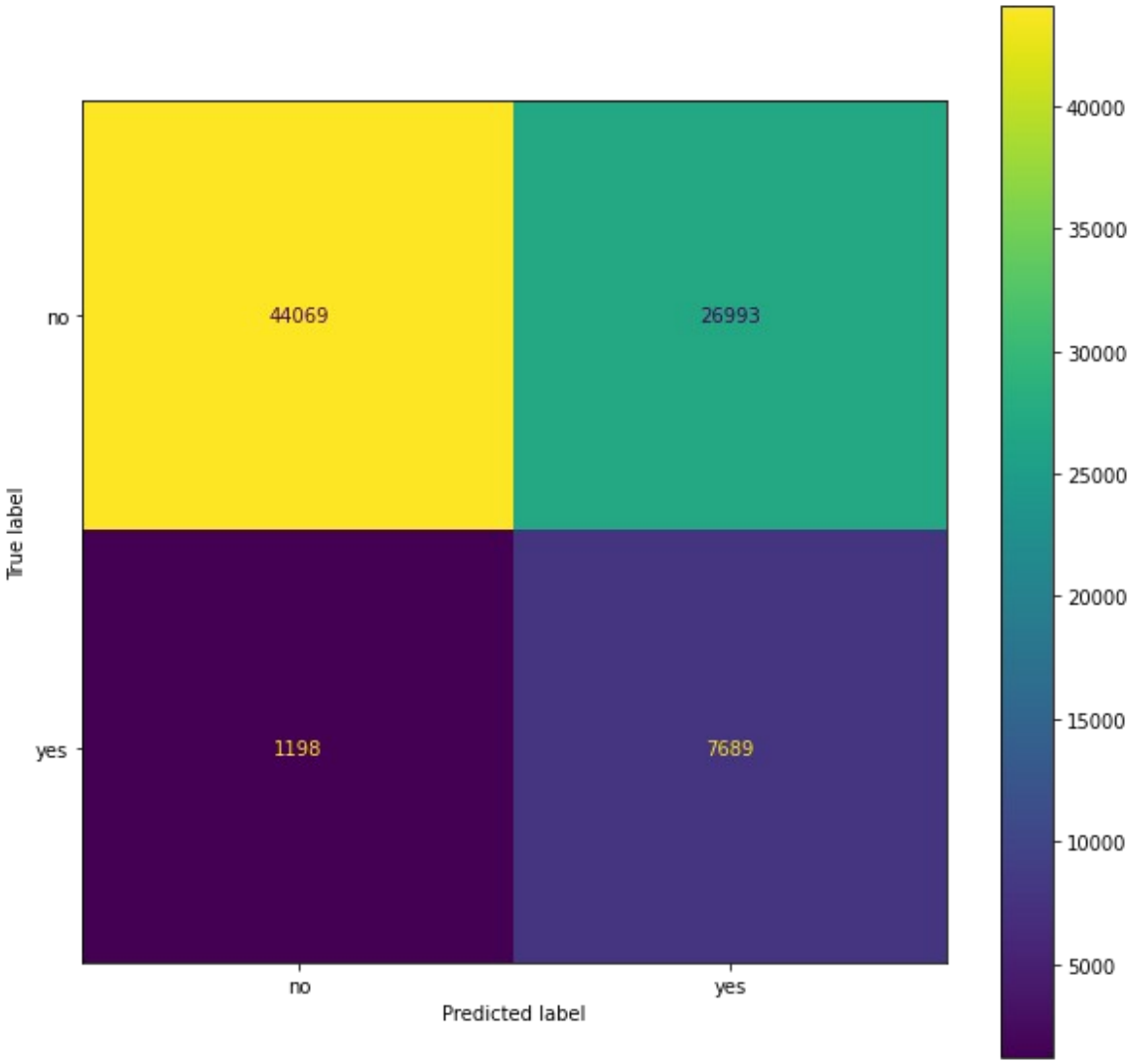
Я взял датасет «Personal Key Indicators of Heart Disease» (https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease). В нем содержится 18 полей и 400 тысяч записей. Я думаю, что для моих целей его хватит с лихвой. Поля могут быть как и числовые, так и символьные (с ограниченным числом вариантов).

На основе подготовленного датасета, взятого из прошлой лр, Я хочу научиться определять, с какой вероятностью у опрашиваемого человека есть проблемы с сердцем.

Ход выполнения лабораторной работы

Во-первых я написал алгоритмы Logistic Regression, SVM, KNN и Naive Bayes в отдельных классах. А так же обучил модели и сравнил их с библиотечными реализациями.

Linear Regression



Линейная регрессия плохо подходит для классификации. Так как результат предсказания модели нужно пропустить через функцию, которая будет давать окончательный ответ.

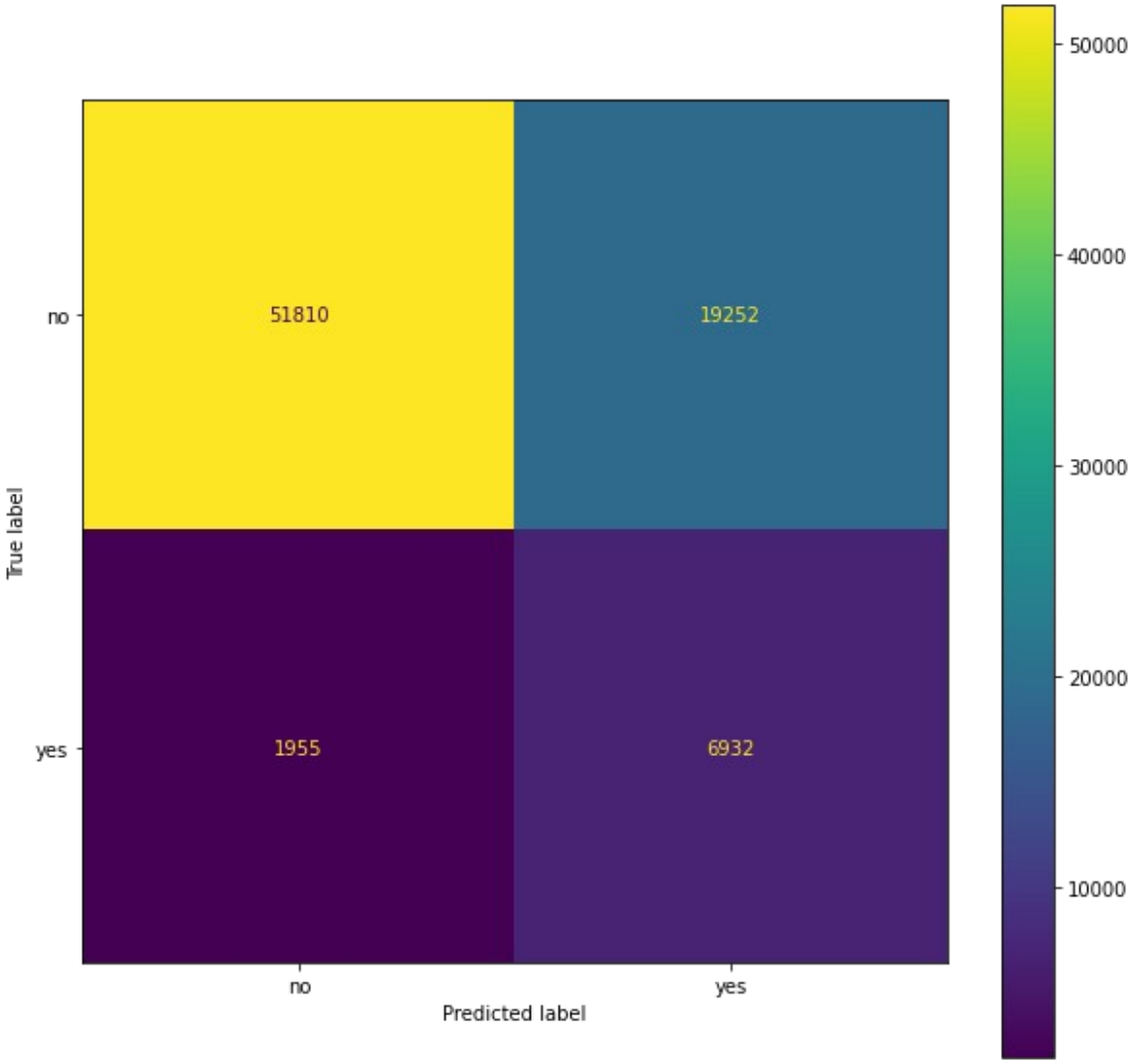
Несмотря на низкую точность, можно заметить, что модель имеет высокий recall. Это означает, что когда человек действительно болен ССЗ (сердечно-сосудистым заболеванием), то модель это с высокой вероятностью обнаруживает.

Однако у моделей много ложных срабатываний, о чем нам говорит precision.

Тем не менее, в нашей задаче важно иметь высокий recall.

Также я заметил что данная модель чувствительна к нормированию данных.

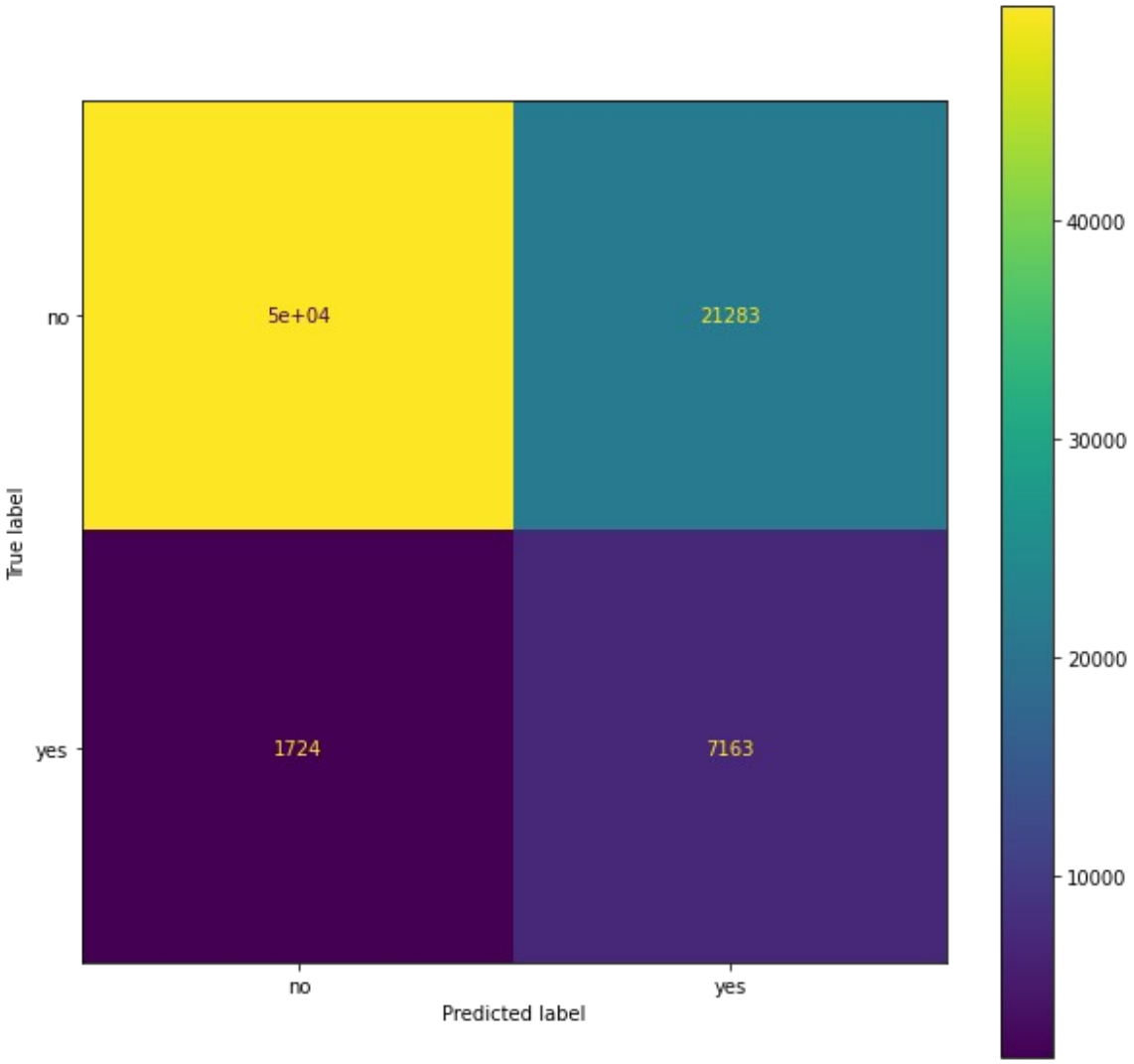
Logistic Regression



Логистическая регрессия больше подходит для классификации, так как она выдает значения в пределах четких границ, которые просто интерпретировать.

У модели более хороший recall и precision.
Также я заметил, что логистическая регрессия чувствительна к нормированию данных и балансу классов.

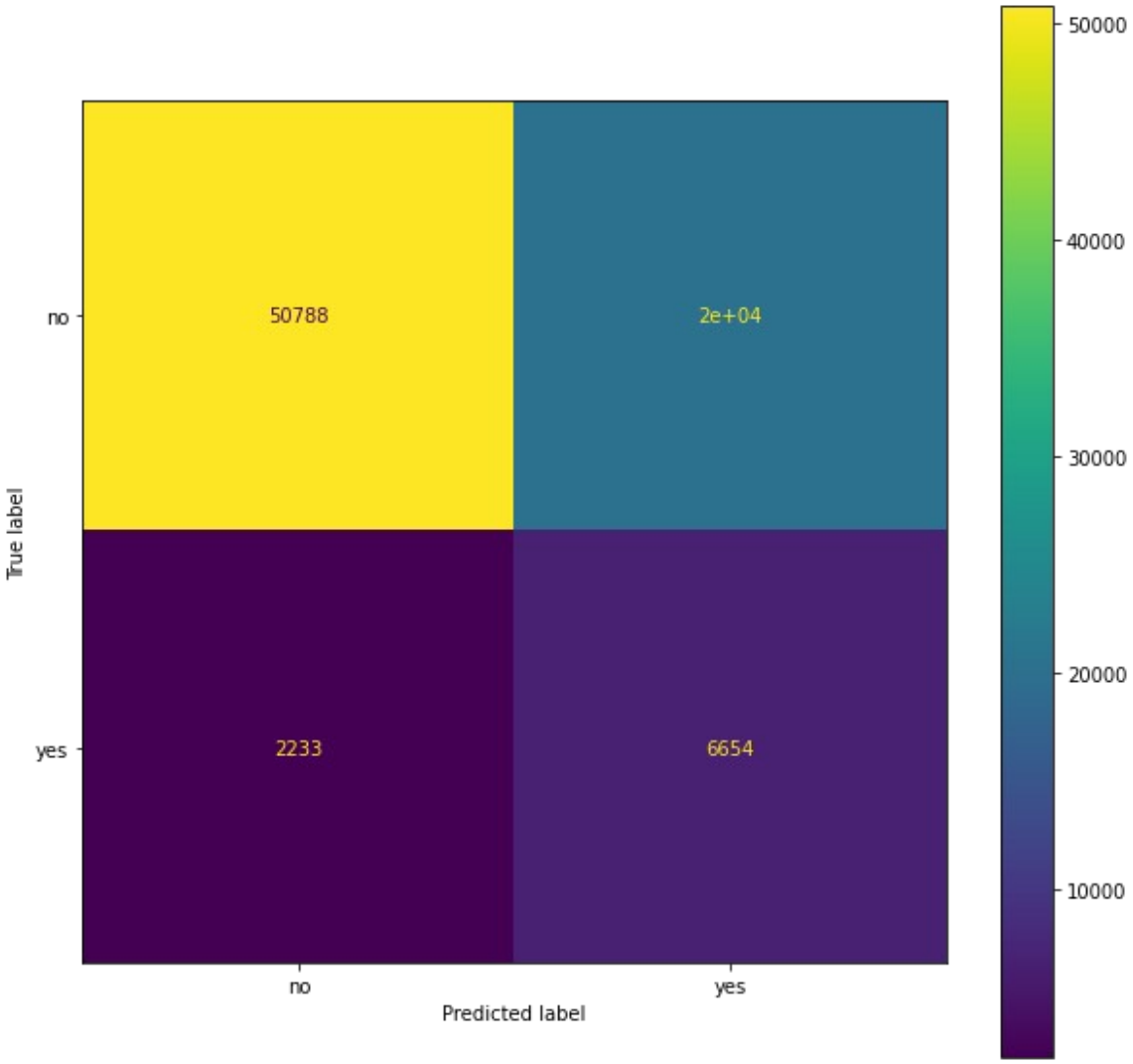
SVM



SVN имеет хороший recall, но и также высокий precision.

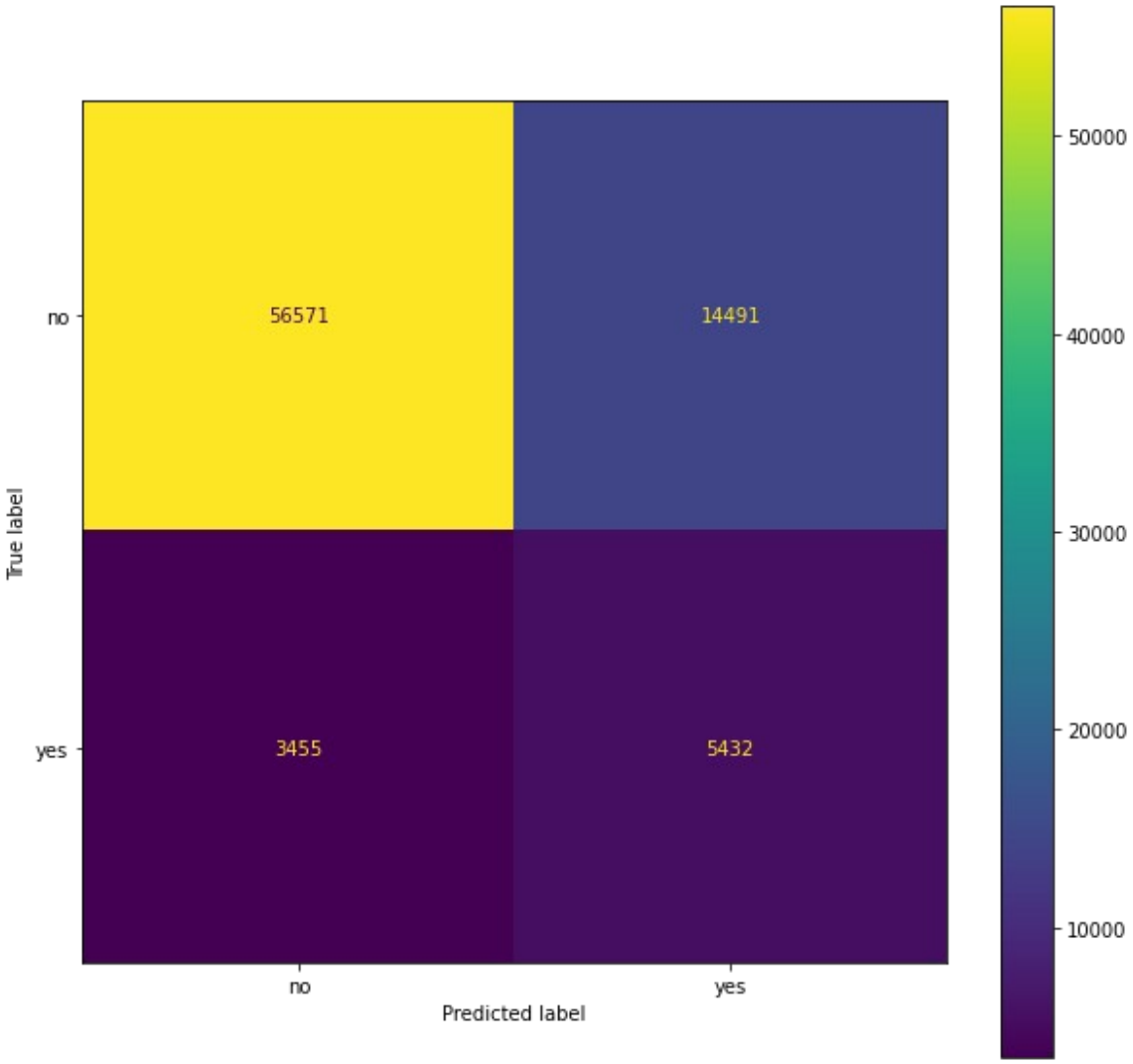
Модель чувствительна к нормированию и балансу классов.

KNN



Модель не очень чувстительна к нормировнию данных.
Показала более лучшие метрики на сбалансированных данных.

Naive Bayes



Naive Bayes имеет невысокий recall.
Показала более лучшие результаты на нормированных и сбалансированных данных.

	Linear Regression	Logistic Regression	SVN	KNN	Naive Bayes
Accuracy	0.65	0.73	0.71	0.72	0.78
Recall	0.87	0.78	0.80	0.75	0.61
Precision	0.22	0.27	0.25	0.25	0.27
Roc-auc	0.74	0.75	0.75	0.73	0.70

Как видно из таблицы, лучшим Recall обладает модель Linear Regression. Но при этом она имеет наихудший Precision, что для этой задачи не очень критично, а также имеет самую низкую точность. Кроме этого, линейная регрессия предназначена для задач регрессии, а не классификации.

Поэтому я бы выбрал SVN, так как она имеет хороший Recall, а так же неплохие Accuracy и Precision.

Выводы

С моей задачей лучше всего справляется линейная регрессия, если нам очень важно находить людей с ССЗ. Так как намного важнее узнать, есть ли у человека ССЗ, чем узнать, что у человека нет ССЗ.

На результат обучения моделей очень сильно влияет качество датасета. В данном случае Почти все модели стали работать лучше после нормирования и балансировки данных.