

УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Системы искусственного интеллекта»

Лабораторная работа №3

Вариант №2 (Оценки)

Студент

Цыпандин Н. П.

P33111

Преподаватель:

Бессмертный И. А.

Авдюшина А. Е.

Санкт-Петербург, 2022 г.

Задание

1. датасет с данными про оценки студентов инженерного и педагогического факультетов (для данного датасета нужно ввести метрику: студент успешный/неуспешный на основании грейда)
2. Отобрать случайным образом \sqrt{n} признаков
3. Реализовать без использования сторонних библиотек построение дерева решений
4. Провести оценку реализованного алгоритма с использованием Accuracy, precision и recall
5. Построить AUC-ROC и AUC-PR

Код

<https://github.com/kkkooolllyyaaa/DecisionTree>

Результат работы алгоритма

```
Ввод [423]: accuracy_score(y_test, y_pred) # Точность
```

```
Out[423]: 0.6333333333333333
```

```
Ввод [424]: precision_score(y_test, y_pred, average='binary')
```

```
Out[424]: 0.9411764705882353
```

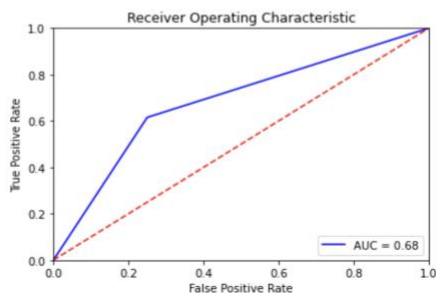
```
Ввод [425]: recall_score(y_test, y_pred, average='binary')
```

```
Out[425]: 0.6153846153846154
```

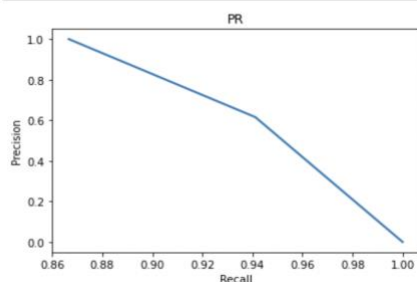
```
Ввод [426]: from sklearn.metrics import roc_curve
from sklearn.metrics import precision_recall_curve
from sklearn.metrics import auc
```

```
Ввод [427]: fpr, tpr, _ = roc_curve(y_test, y_pred)
roc_auc = metrics.auc(fpr, tpr)

plt.title('Receiver Operating Characteristic')
plt.plot(fpr, tpr, 'b', label = 'AUC = %0.2f' % roc_auc)
plt.legend(loc = 'lower right')
plt.plot([0, 1], [0, 1], 'r--')
plt.xlim([0, 1])
plt.ylim([0, 1])
plt.ylabel('True Positive Rate')
plt.xlabel('False Positive Rate')
plt.show()
```



```
Ввод [428]: precision, recall, _ = precision_recall_curve(y_test, y_pred)
plt.plot(precision, recall, linewidth=2)
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.title('PR')
plt.show()
```



Вывод

Я понял как строится дерево решений, также про эвристику с ограничением максимальной глубины и реализовал это самостоятельно используя Golang и Python3