

Билет 14

1. Задача множественной классификации.

Задача классификации — задача, в которой имеется множество объектов (ситуаций), разделённых, некоторым образом, на классы. Задано конечное множество объектов, для которых известно, к каким классам они относятся. Это множество называется выборкой. Классовая принадлежность остальных объектов неизвестна. Требуется построить алгоритм, способный классифицировать произвольный объект из исходного множества.

Когда число классов достигает многих тысяч (например, при распознавании иероглифов или слитной речи), задача классификации становится существенно более трудной.

2. Искусственная нейронная сеть (ИНС) – это ?

Нейронная сеть (также искусственная нейронная сеть, ИНС) — математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма.

Практика

```
class DataAnalysis:
    """
    Дана строка, содержащая русскоязычный текст.
    Найти количество слов, начинающихся с буквы "е".
    В строке заменить все двоеточия (:) знаком процента (%).
    Подсчитать количество замен.
    """

    def __init__(self, string=None):
        self._string = string
        self._countReplace = 0
        self._countWordsWithNeedLetter = 0

    def findWordsWithNeedLetter(self, letterForFind):
        self._findWordsWithNeedLetter(letterForFind)
        return self._countWordsWithNeedLetter

    def _findWordsWithNeedLetter(self, letterForFind):
        self._countWordsWithNeedLetter = len([word for word in
        self._string.split() if word[0].lower() == letterForFind.lower()])

    @property
    def countWordsWithNeedLetter(self):
        return self._countWordsWithNeedLetter

    def replace(self, oldLetter, newLetter):
        self._replace(oldLetter, newLetter)
```

```
        return self.string, self.countReplace

    def _replace(self, oldLetter, newLetter):
        self._countReplace = self._string.count(oldLetter)
        self._string = self._string.replace(oldLetter, newLetter)

    @property
    def countReplace(self):
        return self._countReplace

    @property
    def string(self):
        return self._string

letter, oldLetter, newLetter = 'e', ':', '%'
task = DataAnalysis('На еримере: точность – это сколько из пойманных нами и
посажённых в психушку людей реально сумасшедшие.')
replaceData = task.replace(oldLetter, newLetter)
print(
    f'Строка: {replaceData[0]}',
    f'Кол-во слов начинающихся на "{letter}":
{task.findWordsWithNeedLetter(letter)}',
    f'Кол-во замен "{oldLetter}" на "{newLetter}": {replaceData[1]}',
    sep='\n'
)
```