# Задание по программированию: Анализ тональности отзывов на фильмы: строим простые модели

Зачет, личность идентифицирована · 6/6 баллов

**Срок сдачи** Сдайте это задание до май 28, 11:59 вечера PDT

1. **Инструкции**
2. Моя работа
3. Обсуждения

В этом задании вам предлагается начать разбираться с задачей анализа тональности отзывов на примере сентимент-анализа отзывов на фильмы.

Мы будем использовать стандартный датасет из nltk, уже возникавший в одном из примеров в предыдущих курсах. Для того, чтобы импортировать необходимый модуль, напишите:

```
from nltk.corpus import movie_reviews
```

Чтобы получить id-шники негативных и позитивных отзывов:

```
negids = movie_reviews.fileids('neg')

posids = movie_reviews.fileids('pos')
```

Чтобы получить список негативных отзывов:

```
negfeats = [movie_reviews.words(fileids=[f]) for f in negids]
```

## Инструкция по выполнению

В некоторых пунктах нужно получить ответ - число или строку, которые будет нужно набирать в текстовых файлах и прикреплять в ответах на вопросы. *Десятичные дроби записывайте через точку.*

1. Создайте список из текстов всех имеющихся отзывов, а также список с классами, которые будет использовать ваш классификатор - 0 для негативных отзывов и 1 для позитивных.
2. Подсчитайте количество отзывов в выборке.
3. Подсчитайте долю класса 1 в выборке.
4. Импортируйте CountVectorizer из sklearn.feature_extraction.text. Попробуйте использовать его с настройками по умолчанию для того, чтобы получить признаковое представление каждого текста. Скорее всего, попытка не увенчается успехом. Разберитесь, в чем причина, и добейтесь того, чтобы метод fit_transform у CountVectorizer успешно отрабатывал. Подсчитайте количество признаков в CountVectorizer. Никакой предварительной обработки текста (удаление стоп-слов, нормализация слов) на этом шаге делать не надо, в качестве признаков должны использоваться частоты слов.
5. Соберите pipeline из CountVectorizer и LogisticRegression с настройками по-умолчанию и с помощью cross_val_score (также со стандартными настройками) оцените получаемое "из коробки" качество по accuracy.

6. Аналогично accuracy, оцените качество по ROC AUC.

7. Обучите логистическую регрессию на всей доступной вам выборке и выведите 5 наиболее важных для модели признаков (подумайте, какие именно признаки стоит считать такими). Вам могут пригодиться метод get_feature_names() или поле vocabulary_ у класса CountVectorizer.