Домашнее задание 3

Задача

В этом задании вам предстоит провести анализ данных и обучить модель регрессии.

Вы будете прогнозировать **стоимость дома** на основе характеристик проданных домов.

Данные

В вашем распоряжении данные о домах в округе Кинг, штат Вашингтон, США. kc_house_data.csv. Данные содержат информацию о проданных домах в период с мая 2014 по май 2015.

- **id**: индетификатор дома
- **date**: дата продажи
- **price**: стоимость дома при продаже (таргет)
- **bedrooms**: количество спален в доме
- **bathrooms**: количество ванных в доме
- **sqft_living:** размер дома в квадратных метрах (общее)
- **sqft_lot**: размер участка в квадратных метрах
- **floors**: количество этажей/уровней в доме
- **waterfront**: дома с видом на воду (пруд-озеро-река и т.п.)
- **view**: дом был просмотрен
- **condition**: состояние дома в целом
- **grade**: оценка дома по системе, принятой в King County
- **sqft_above**: размер дома в квадратных метрах без подвала
- **sqft_basement**: размер подвала в квадратных метрах
- **yr_built**: год постройки
- **yr_renovated**: год ремонта
- **zipcode**: индекс почтового адреса дома
- **lat**: широта (географическая координата)
- **long**: долгота (географическая координата)
- **sqft_living15**: размер дома в квадратных метрах (общее) в 2015 году (в том числе после реноваций).Изменения могли коснуться размера участка.
- **sqft_lot15**: размер участка в квадратных метрах в 2015 году (в том числе после реноваций)

Задания

Часть 1: Исследование данных.

В этом блоке предложены минимальное необходимые задания, чтобы появилась тактика работы с данными в следующем блоке.

1. Изучите таргет: визуализируйте его с помощью гистограммы, оцените, нужна ли предобработка таргету.
2. Рассчитайте описательные статистики датасета, как для числовых переменных, так и для категориальных. Изучите полученные таблицы. Есть

ли что-то, что выбивается? Зафиксируйте переменные, которые нуждаются в предобработке.

3. Визуализируйте данные:
    a. Проиллюстрируйте все попарные взаимосвязи sns.pairplot
    b. Визуализируйте стоимость продажи дома в зависимости от даты продажи. Каким образом агрегировать информацию (и агрегировать ли вообще) – на усмотрение слушателя.
    c. Рассчитайте матрицу корреляций.
    d. При необходимости, сделайте интересующие вас визуализации.

Результатом этой части являются выводы, какие действия нужно совершить с данными. Зафиксируйте их в отдельной текстовой ячейке.

Часть 2: Предобработка данных.

В этой части реализуются все шаги, намеченные вами в предыдущем пункте. К примеру, можно прологарифмировать данные, отфильтровать по каким-то значениям, удалить пропущенные и т.п.

Часть 3: Подготовка к обучению.

1. При необходимости отшкалируйте данные стандартным шкалировщиком.
2. Разбейте данные на тренировочную (80%) и тестовую части (20%). Зафиксируйте random_state=42.

Часть 4: Модель линейной регрессии:

1. Обучите модель линейной регрессии. Рассчитайте метрику MAE, чтобы оценить качество модели.
2. Убедитесь, что модель демонстрирует обобщающую способность: проведите кросс-валидацию по 5 фолдам и оцените метрику качества MAE. Рассчитайте среднее и стандартное отклонение оценок на тестовой выборке и тренировочной. Оцените (словами) обобщающую способность модели.

Вам пригодится функция `cross_validate` модуля `sklearn.model_selection`. Вот пример ее применения:

```
scores = cross_validate(lasso, X, y, cv=3,
...                         scoring=('neg_mean_squared_error'),
...                         return_train_score=True)
```

В полях ‘`test_score`’ и ‘`train_score`’ объекта scores будут записаны значения метрики качества на тестовой или тренировочной выборке, которая указана в

параметре `scoring`. Обратите внимание, что в указанном примере используется MSE, только с префиксом 'neg_mean_squared_error'. Это особенность реализации всех функций для кросс-валидации в sklearn, в которых вшита логика «чем больше, тем лучше»:

```
greater_is_better : boolean, default=True

Whether score_func is a score function (default), meaning high is good,
or a loss function, meaning low is good. In the latter case, the scorer
object will sign-flip the outcome of the score_func
```

В то же время, оптимизируя MSE или MAE, мы придерживаемся принципа «чем меньше, тем лучше». Поэтому в кросс-валидации используются метрики 'neg_mean_squared_error' или 'neg_mean_absolute_error'.

Часть 5: Модели Lasso и Rigde:

1. Обучите модель Лассо-регрессии и подберите оптимальное значение гиперпараметра альфа. Используйте 5-фолд-кросс-валидацию, а потенциальные значения альф используйте, как в коде на занятии. Оцените качество прогноза при оптимальном гиперпараметре на тестовой выборке.
2. Аналогично, обучите модель Ридж-регрессии.
3. Сравните feature_importance моделей Lasso и Ridge. Есть ли признаки, которые обнулились?

Часть 6. Градиентный спуск

1. Реализуйте функцию стохастического градиентного спуска для функционала MSE. Для реализации вам понадобится градиент, который в матричной форме выглядит следующим образом:

$$\nabla Q(w) = 2X^T(Xw - y)$$

   Назовите функцию stochastic_gd, входные параметры:

   a. step_size – размер шага
   b. w – начальные веса
   c. cnt_steps – количество шагов градиентного спуска
   d. X – матрица признаков
   e. y – вектор целевой переменной

2. Примените написанную функцию к данным задачи. Сравните веса модели, полученные в части 4 и здесь. Прокомментируйте.