



ft_linear_regression

Введение в машинное обучение

Резюме: В этом проекте вы реализуете свой первый алгоритм машинного обучения.

Версия: 4

Содержание

I	Предисловие	2
II	Введение	3
III	Объектив	4
IV	Общие инструкции	5
V	Обязательная часть	6
VI	Бонусная часть	7
VII	Представление и экспертная оценка	8

Глава I

Предисловие

Что, на мой взгляд, является лучшим определением для машинного обучения:

"Говорят, что компьютерная программа учится на опыте E в отношении некоторого класса задач T и меры производительности P , если ее производительность на задачах из T , измеряемая P , улучшается с опытом E ".

Том М. Митчелл

Глава II

Введение

Машинное обучение - это растущая область компьютерных наук, которая может показаться немного сложной и доступной только математикам. Возможно, вы слышали о нейронных сетях или кластеризации k-means, но не понимаете, как они работают и как кодировать подобные алгоритмы...

Но не волнуйтесь, на самом деле мы начнем с простого, базового алгоритма машинного обучения.

Глава III Цель

Цель этого проекта - познакомить вас с базовой концепцией машинного обучения. В этом проекте вам предстоит создать программу, предсказывающую цену автомобиля с помощью [линейной функции](#), построенной по [алгоритму градиентного спуска](#).

Мы будем работать над конкретным примером для проекта, но после завершения работы вы сможете использовать алгоритм с любым другим набором данных.

Глава IV

Общие инструкции

В этом проекте вы можете использовать любой язык.

Вы также можете использовать любые библиотеки, если они не делают всю работу за вас. Например, использование `numpy.polyfit` из `python` считается мошенничеством.



Вы должны использовать язык, который позволяет легко визуализировать данные: это очень поможет при отладке.

Глава V

Обязательная часть

Вам предстоит реализовать простую линейную регрессию с одним признаком - в данном случае пробегом автомобиля.

Для этого необходимо создать две программы:

- Первая программа будет использоваться для прогнозирования стоимости автомобиля при заданном пробеге. При запуске программы она должна предложить вам ввести пробег, а затем выдать предполагаемую цену для этого пробега. Программа будет использовать следующую гипотезу для прогнозирования цены:

$$\text{estimatePrice}(\text{mileage}) = \theta_0 + (\theta_1 * \text{mileage})$$

Перед запуском обучающей программы θ_0 и θ_1 будут установлены в 0.

- Вторая программа будет использоваться для обучения вашей модели. Она прочитает ваш файл набора данных и выполнит линейную регрессию данных. После завершения линейной регрессии вы сохраните переменные θ_0 и θ_1 для использования в первой программе. Вы будете использовать следующие формулы:

$$\text{tmp}\theta_0 = \text{learningRate} * \frac{1}{m} \sum_{i=0}^{m-1} (\text{estimatePrice}(\text{mileage}[i]) - \text{price}[i])$$

$$\text{tmp}\theta_1 = \text{learningRate} * \frac{1}{m} \sum_{i=0}^{m-1} (\text{estimatePrice}(\text{mileage}[i]) - \text{price}[i]) * \text{пробег}[i]$$

Позвольте вам угадать, что это за "m" :)

Обратите внимание, что `estimatePrice` такой же, как и в нашей первой программе, но здесь он использует ваши временные, вычисленные в последний раз θ_0 и θ_1 .

Также не забудьте одновременно обновить θ_0 и θ_1 .

Глава VI

Бонусная часть

Вот несколько бонусов, которые могут оказаться очень полезными:

- Нанесите данные на график, чтобы увидеть их распределение.
- Постройте линию, полученную в результате линейной регрессии, на том же графике, чтобы увидеть результат вашей тяжелой работы!
- Программа, которая вычисляет точность вашего алгоритма.



Бонусная часть оценивается только в том случае, если обязательная часть выполнена безупречно. Идеально - это значит, что обязательная часть выполнена целостно и работает без сбоев. Если вы не выполнили ВСЕ обязательные требования, ваша бонусная часть не будет оцениваться вообще.

Глава VII

Представление и экспертная оценка

Сдайте задание в свой Git-репозиторий, как обычно. На защите будет оцениваться только работа, находящаяся в вашем репозитории. Не стесняйтесь перепроверять названия папок и файлов, чтобы убедиться в их правильности.

Вот пункты, которые должен будет проверить ваш коллега-корректор:

- Отсутствие библиотек, которые делают работу за вас
- Использование указанной гипотезы
- Использование заданной функции обучения