

Тятя! Тятя! Нейросети заменили продавца!

Ппилиф Ульяновкин

https://github.com/FUlyankin/neural_nets_prob

Листочек 3: пятьдесят оттенков градиентного спуска

Повторять до сходимости — это как жарить до готовности

Неизвестный студент Вышки

Упражнение 1 (50 оттенков спуска)

Маша Нестерова, хозяйка машин лёрнинга¹, собрала два наблюдения: $x_1 = 1, x_2 = 2, y_1 = 2, y_2 = 3$ и собирается обучить линейную регрессию $y = w \cdot x$. Маша очень хрупкая девушка, и ей не помешает помощь.

- а. Получите теоретическую оценку методом наименьших квадратов.
- б. Сделайте три шага градиентного спуска. В качестве стартовой точки используйте $w_0 = 0$. В качестве скорости обучения возьмите $\eta = 0.1$.
- в. Сделайте четыре шага стохастического градиентного спуска. Пусть в SGD сначала попадает первое наблюдение, затем второе.
- г. Если вы добрались до этого пункта, вы поняли градиентный спуск. Маша довольна. Начиная заниматься тупой технической бессмыслицей. Сделайте два шага Momentum SGD. Возьмите $\alpha = 0.9, \eta = 0.1$
- д. Сделайте два шага Momentum SGD с коррекцией Нестерова.
- е. Сделайте два шага RMSprop. Возьмите $\alpha = 0.9, \eta = 0.1$
- ж. Сделайте два шага Adam. Возьмём $\beta_1 = \beta_2 = 0.9, \eta = 0.1$

Упражнение 2 (логистическая регрессия)

Маша решила, что нет смысла останавливаться на обычной регрессии, когда она знает, что есть ещё и логистическая:

¹Лёрнинг ей папа подарил

$$z = w \cdot x \quad p = P(y = 1) = \frac{1}{1 + e^{-z}}$$

$$\text{logloss} = -[y \cdot \ln p + (1 - y) \cdot \ln(1 - p)]$$

Запишите формулу, по которой можно пересчитывать веса в ходе градиентного спуска для логистической регрессии.

Оказалось, что $x = -5$, а $y = 1$. Сделайте один шаг градиентного спуска, если $w_0 = 1$, а скорость обучения $\gamma = 0.01$.

Упражнение 3 (вопросики)

Убедитесь, что вы можете дать ответы на следующие вопросы:

- Как вы думаете, почему считается, что SGD лучше работает для оптимизации функций, имеющих больше одного экстремума?
- Предположим, что у функции потерь есть несколько локальных минимумов. Как можно адаптировать градиентный спуск так, чтобы он находил глобальный минимум чаще?
- Что будет происходить со стохастическим градиентным спуском, если длина его шага не будет уменьшаться от итерации к итерации?

Упражнение 4 (скорости обучения)

В стохастическом градиентном спуске веса изменяются по формуле

$$w_t = w_{t-1} - \eta_t \cdot \nabla L(w_{t-1}, x_i, y_i),$$

где наблюдение i выбрано случайно, скорость обучения зависит от номера итерации.

Условия Роббинса-Монро гарантируют сходимость алгоритма к оптимуму для выпуклых дифференцируемых функций. Они говорят, что ряд из скоростей $\sum_{t=0}^{\infty} \eta_t$ должен расходиться, а ряд $\sum_{t=0}^{\infty} \eta_t^2$ сходиться. То есть скорость спуска должна падать не слишком медленно, но и не слишком быстро. Какие из последовательностей, перечисленных ниже, можно использовать для описания изменения скорости алгоритма?

- $\eta_t = \frac{1}{t}$
- $\eta_t = \frac{0.1}{t^{0.3}}$
- $\eta_t = \frac{1}{\sqrt{t}}$
- $\eta_t = \frac{1}{t^2}$
- $\eta_t = e^{-t}$
- $\eta_t = \lambda \cdot \left(\frac{s_0}{s_0 + t} \right)^p$, где λ, p и s_0 — параметры