Применение стохастического градиентного спуска в обобщенных линейных моделях

1 Теоретическая часть

1.1 Линейная регрессия

Модель: зависимая переменная $\mathbf{y}$ распределена нормально и

$$\mathsf{E}(\mathbf{y} \mid \mathbf{X}) = \boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta},$$

где $\mathbf{X}$ — матрица данных, $\boldsymbol{\beta}$ — вектор параметров. Плотность нормального распределения:

$$p(y; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}, \quad y \in \mathbb{R}.$$

Матожидание и дисперсия нормального распределения равны соответственно μ и $\sigma^2 = \phi$. Логарифм правдоподобия:

$$L(\mathbf{y}; \mu, \phi) = -\frac{n}{2} \ln 2\pi\phi - \frac{1}{2\phi} \sum_{i=1}^{n} (y_i - \mu)^2.$$

Тогда, поскольку параметр ϕ, вообще говоря, неизвестен, необходимо оптимизировать функцию потерь и по ϕ. Производные по вектору параметров $\boldsymbol{\beta}$ и ϕ:

$$\frac{\partial L}{\partial \boldsymbol{\beta}} = \frac{1}{\phi} \mathbf{X}^{\mathrm{T}} (\mathbf{y} - \boldsymbol{\mu}), \quad \frac{\partial L}{\partial \phi} = -\frac{n}{2\phi} + \frac{1}{2\phi^2} \sum_{i=1}^{n} (y_i - \mu_i)^2.$$

1.2 Гамма-регрессия

Модель: зависимая переменная $\mathbf{y}$ имеет гамма-распределение и

$$\mathsf{E}(\mathbf{y} \mid \mathbf{X}) = \boldsymbol{\mu} = g^{-1}(\mathbf{X}\boldsymbol{\beta}),$$

где g — линк-функция. Плотность гамма-распределения:

$$p(y; k, \theta) = \frac{y^{k-1}}{\Gamma(k)\theta^k} \exp\left\{-\frac{y}{\theta}\right\}, \quad y > 0, \quad k, \theta > 0.$$

Модель гамма-регрессии рассматривается, когда известно, что зависимая переменная принимает только положителные значения (в отличие от линейной модели, где зависимая переменная принимает любые значения), например, она представляет собой некоторый физический показатель.

Математическое ожидание и дисперсия гамма-распределения равны соответственно $\mu = \theta/\phi$ и $\mu^2\phi$, где $\phi = 1/k$. Перепараметризуем плотность:

$$p(y; \mu, \phi) = \frac{1}{y\Gamma(1/\phi)} \left(\frac{y}{\phi\mu}\right)^{1/\phi} \exp\left\{-\frac{y}{\phi\mu}\right\}, \quad y > 0, \quad \mu, \phi > 0.$$

Логарифм функции правдоподобия

$$L(\mathbf{y}; \mu, \phi) = -n \ln \Gamma(1/\phi) + \sum_{i=1}^{n} \left[\frac{1}{\phi} \ln \frac{y_i}{\phi\mu} - \ln y_i - \frac{y_i}{\phi\mu}\right].$$

Поскольку $\mu > 0$, в качестве линк-функции фозьмем $g(x) = \ln(x)$. Тогда производные по вектору параметров $\boldsymbol{\beta}$ и ϕ:

$$\frac{\partial L}{\partial \boldsymbol{\beta}} = \frac{1}{\phi} \mathbf{X}^{\mathrm{T}} (\mathbf{y} - \boldsymbol{\mu}) / \boldsymbol{\mu}, \quad \frac{\partial L}{\partial \phi} = \frac{1}{\phi^2} \left(n \cdot \psi(1/\phi) - n + \sum_{i=1}^{n} \left[-\ln \frac{y_i}{\phi\mu_i} + \frac{y_i}{\mu_i}\right]\right),$$

где деление векторов производится поэлементно, $\psi(x) = \Gamma'(x)/\Gamma(x)$ — дигамма-функция.

2 Что было сделано

Мной был реализован стохастический градиентный спуск (класс SGD) для обучения обобщенной линейной модели (GLM) на языке Python в двух версиях:

- Momentum:

$$v_{k+1} = \eta_1 v_k - (1-\eta_1) \cdot \alpha \nabla f(x_k),$$
$$x_{k+1} = x_k + v_{k+1};$$

- Adam (ADAptive Momentum):

$$v_{k+1} = \eta_1 v_k + (1-\eta_1) \nabla f(x_k),$$
$$G_{k+1} = \eta_2 G_k + (1-\eta_2) \left(\nabla f(x_k)\right)^2,$$
$$x_{k+1} = x_k - \frac{\alpha}{\sqrt{G_{k+1}+\varepsilon}} v_{k+1}.$$

Мной были написаны только варианты для линейной регрессии и GLM с гамма-распределением, но метод обобщается и на другие модели GLM определением соответствующего класса, наследующего класс Family.

3 Проверка работы метода

Рассмотрим синтетический и реальный пример и обучим SGD на этих данных. В качестве варианта SGD будем использовать Adam с $\eta_1 = 0.9$, $\eta_2 = 0.99$, $\varepsilon = 10^{-8}$.

3.1 Проверка на сгенерированных данных

Пусть количество наблюдений равно $n = 1000$, количество признаков (без учета константного члена) $p = 10$. Столбцы матрицы данных $\mathbf{X} = [X_1 : X_2 : \ldots : X_{10}]$ генерировалась из $\mathrm{U}(0, 1)$, вектор параметров $\boldsymbol{\beta}$ — из $\mathrm{U}(-10, 10)$. Параметр дисперсии ϕ равен 0.01 в случае линейной регрессии и 0.1 в случае гамма-регрессии.

Будем смотреть на близость оцененных параметров к истинным. Помимо этого, будем вычислять Adjusted R^2 в случае линейной модели и его аналог Pseudo R^2 в случае гамма-регрессии. В таблице 1 представлены результаты SGD с $\alpha = 0.03$.

Таблица 1: Результат SGD ($\alpha = 0.03$) на сгенерированных данных

	Оцененное значение	Истинное значение
Intercept	3.935208	3.929384
X_1	-4.254903	-4.277213
X_2	-5.470059	-5.462971
X_3	1.028168	1.026295
X_4	4.379706	4.389379
X_5	-1.544207	-1.537871
X_6	9.620079	9.615284
X_7	3.711597	3.696595
X_8	-0.383994	-0.381362
X_9	-2.180469	-2.157650
X_{10}	-3.131106	-3.136440
ϕ	0.008969	0.01

(a) Линейная регрессия (Adjusted $R^2 = 0.9995$)

	Оцененное значение	Истинное значение
Intercept	3.941278	3.929384
X_1	-4.263230	-4.277213
X_2	-5.452918	-5.462971
X_3	1.000984	1.026295
X_4	4.324310	4.389379
X_5	-1.535370	-1.537871
X_6	9.645510	9.615284
X_7	3.712122	3.696595
X_8	-0.379930	-0.381362
X_9	-2.164776	-2.157650
X_{10}	-3.133490	-3.136440
ϕ	0.092371	0.1

(b) Гамма-регрессия (Pseudo $R^2 = 0.9922$)

3.2 Проверка на реальных данных

В качестве данных рассмотрим датасет, содержащий информацию о сделках с недвижимостью. Он состоит из следующих столбцов:

- Transaction date — дата совершения сделки.
- House age — возраст дома в годах на момент совершения сделки.
- Distance to the nearest MRT station — близость к ближайшей станции общественного скоростного транспорта в метрах.
- Number of convenience stores — количество круглосуточных магазинов поблизости.
- Latitude — широта.
- Longitude — долгота.
- House price of unit area — цена за единицу площади объекта недвижимости.

Будем предсказывать цену недвижимости. Цена, очевидно, принимает только положительные значения, поэтому модель гамма-регрессии подходит. Помимо гамма-регрессии обучим и линейную регрессию с пониманием, что предсказанные значения могут быть и отрицательными.

Будем сравнивать самописный метод с методами из библиотек: для линейной регрессии был выбран класс SGDRegressor из библиотеки scikit-learn с параметрами learning_rate="adaptive", eta0=0.1, penalty=None, для гамма-регресси — класс GLM из библиотеки statsmodels. В таблицах 2 и 3 представлены результаты сравнения, получили примерно одинаковые результаты.

Таблица 2: Сравнение SGD ($\alpha = 0.1$) с SGDRegressor (линейная регрессия)

	SGD	SGDRegressor
Intercept	37.976997	37.985638
Transaction date	1.446827	1.461966
House age	-3.055246	-3.070485
Distance to the nearest MRT station	-5.660887	-5.635479
Number of convenience stores	3.373381	3.339555
Latitude	2.809412	2.805714
Longitude	-0.187202	-0.177384
Adjusted R^2	0.576197	0.576212

Таблица 3: Сравнение SGD ($\alpha = 0.01$) с GLM (гамма-регрессия)

	SGD	GLM
Intercept	3.591820	3.590622
Transaction date	0.046941	0.045534
House age	-0.073330	-0.073809
Distance to the nearest MRT station	-0.199475	-0.198254
Number of convenience stores	0.074807	0.074781
Latitude	0.087781	0.088450
Longitude	-0.004894	-0.004029
Dispersion (scale) parameter ϕ	0.048081	0.053814
Pseudo R^2	0.659948	0.659976