Отчет

Лёва Паносян

28 ноября 2021 г.

0.1 Градиент и гессиан функции логистической регрессии

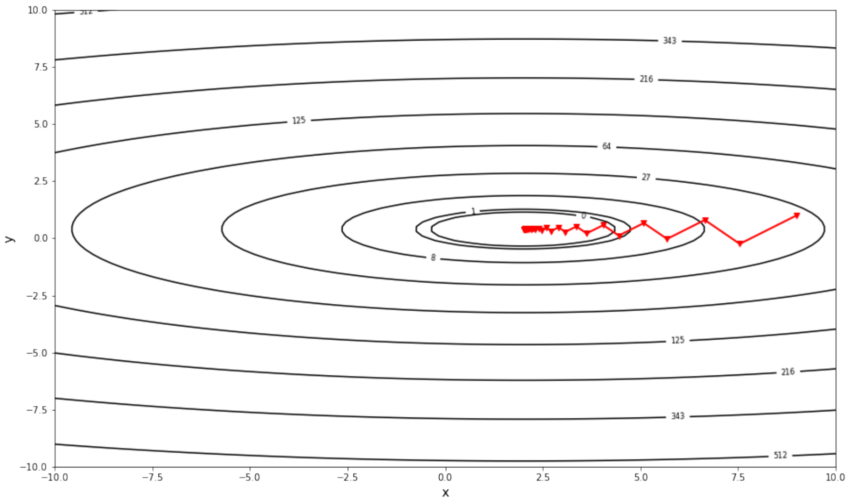
$\mathcal{F}(x) = \frac{1}{m}\sum_{i=1}^{m}\left(ln\left(1 + exp\left(-b_i < a_i, x >\right)\right)\right) + \frac{\lambda}{2}||x||_2^2$

$d\mathcal{F}(x) = \frac{1}{m}\sum_{i=1}^{m} d(ln(1 + exp(-b_i < a_i, x >))) + \lambda < x, dx >=$
$= \frac{1}{m}\sum_{i=1}^{m} \frac{d(exp(-b_i < a_i, x >))}{1 + exp(-b_i < a_i, x >)} + \lambda < x, dx >=$
$= -\frac{1}{m}\sum_{i=1}^{m} expit(-b_i < a_i, x >) b_i < a_i, dx > + \lambda < x, dx >=$
$=< \lambda x - \frac{1}{m} A^T Diag(b) expit(-Diag(b)Ax), dx >$
Следует что $\Delta\mathcal{F}(x) = \lambda x - \frac{1}{m} A^T Diag(b) expit(-Diag(b)Ax)$
$d(expit(x)) = (expit(x) - expit^2(x))dx$
$d^2\mathcal{F}(x) =< \lambda dx_2 - \frac{1}{m} A^T Diag(b) d(expit(-Diag(b)Ax)), dx_1 >=$
$=< \lambda dx_2 + \frac{1}{m} A^T Diag(b) expit(-Diag(b)Ax)(1 -$
$expit(-Diag(b)Ax)) Diag(b) A dx_2, dx_1 >$
$\bigtriangledown^2\mathcal{F}(x) =$
$\lambda E_n + \frac{1}{m} A^T Diag(b) expit(-Diag(b)Ax)(1 - expit(-Diag(b)Ax)) Diag(b) A$

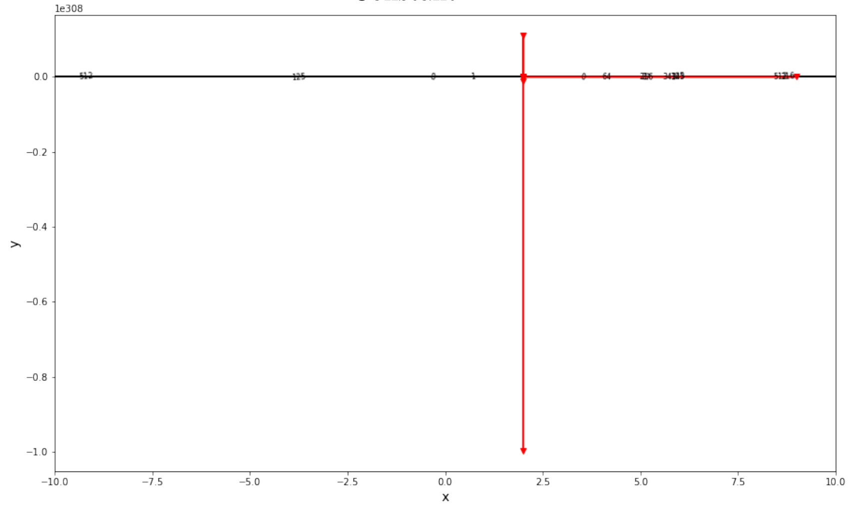
0.2 1-ий эксперимент

Результаты при $A = \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix}$, $b = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$,$x_0 = \begin{bmatrix} 9 \\ 1 \end{bmatrix}$

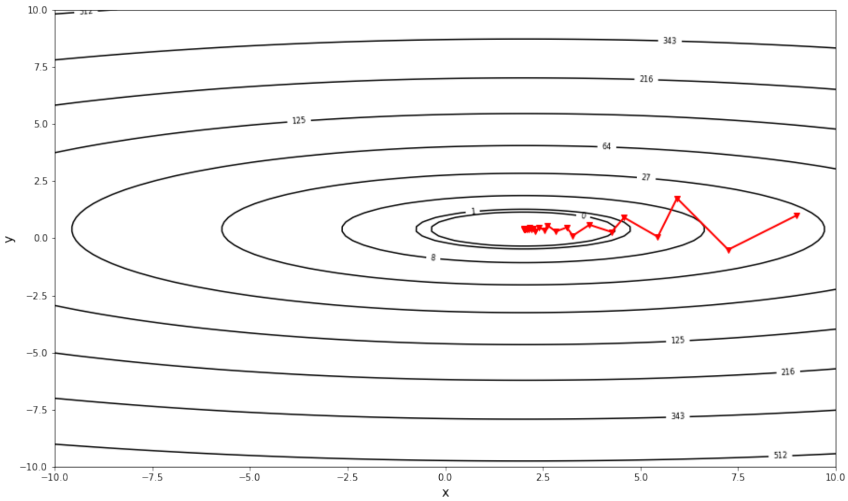
Wolfe



Constant

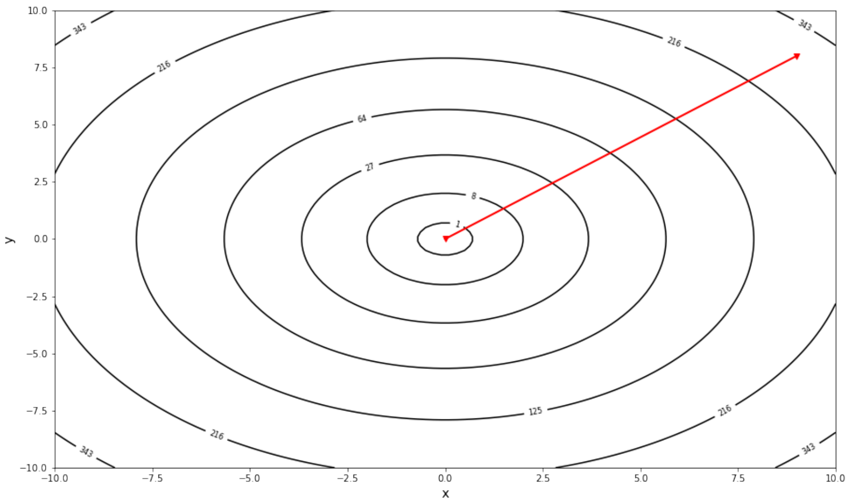


Armijo

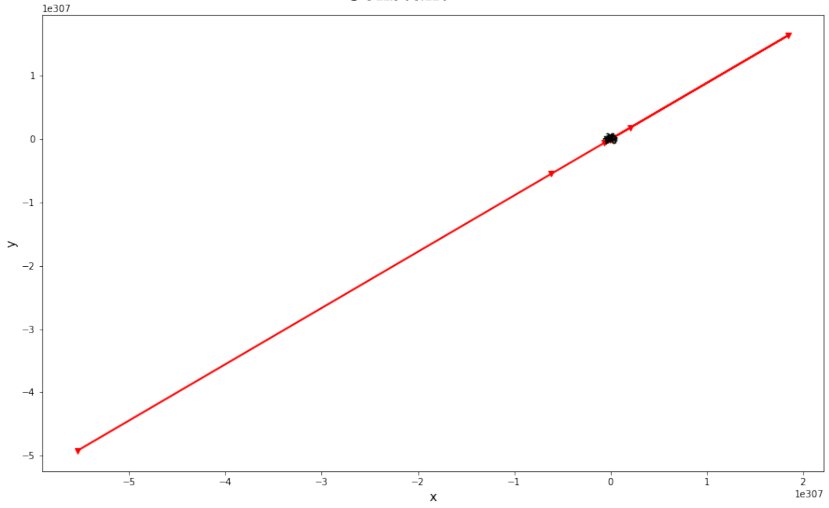


Результаты при $A = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}$, $b = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$,$x_0 = \begin{bmatrix} 9 \\ 8 \end{bmatrix}$

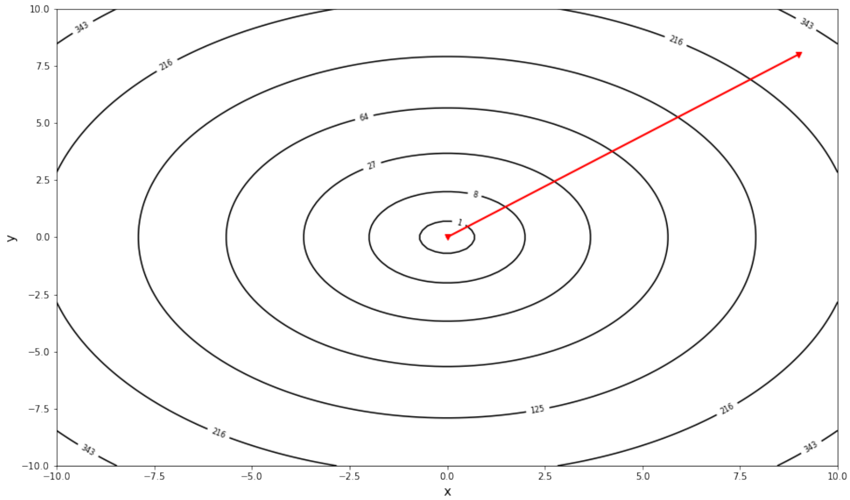
Wolfe



Constant

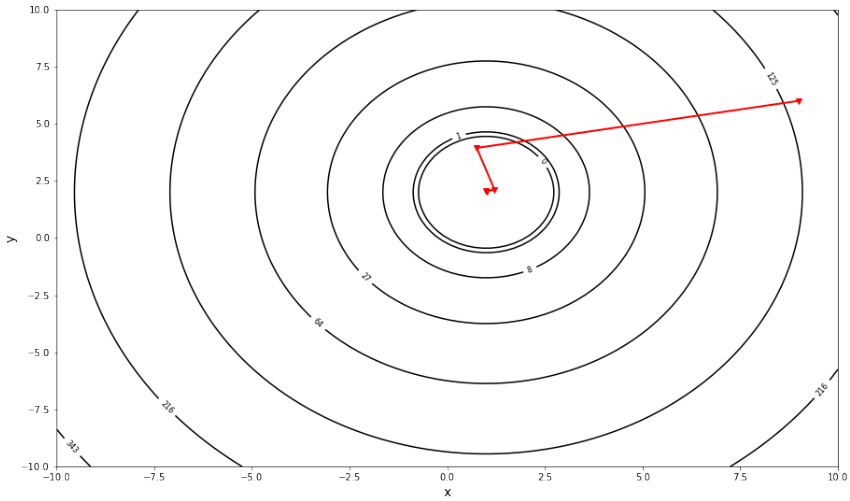


Armijo

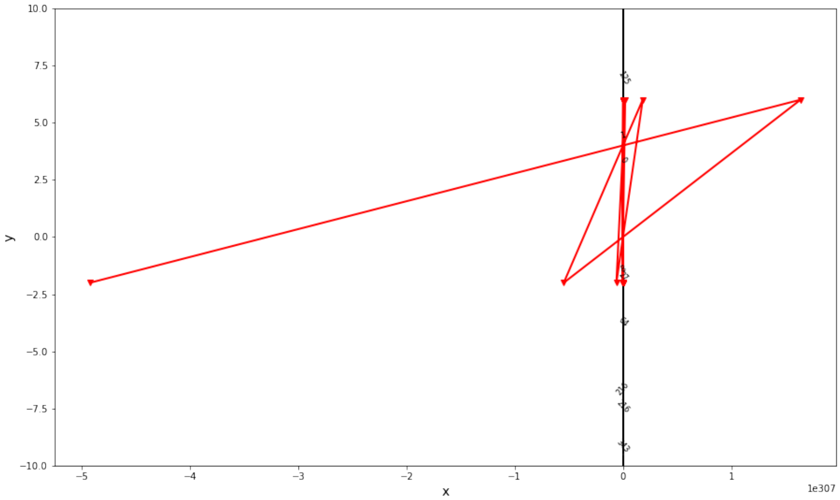


Результаты при $A = \begin{bmatrix} 4 & 0 \\ 0 & 2 \end{bmatrix}$, $b = \begin{bmatrix} 4 \\ 4 \end{bmatrix}$,$x_0 = \begin{bmatrix} 9 \\ 6 \end{bmatrix}$

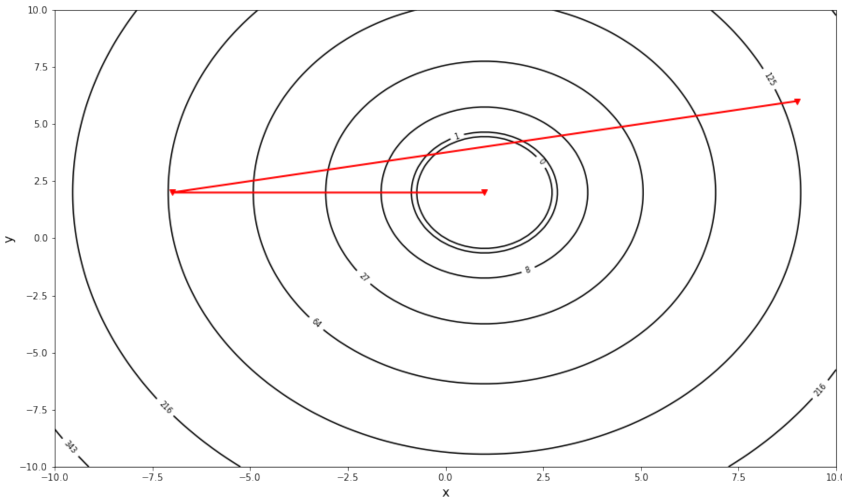
Wolfe



Constant



Armijo



Из этих графиков можно сказать, что наилучший результат дает метод линейного поиска Wolfe. пир таком методе точка равномерно приближаеться к целью. При методе Armijo точка не равномерно приближаеться, но при каждом шаге хотя бы приближатеься. А Константа самый плохой из трех, потому что все равно хочеться "подсказать"шаг, но это можно сделать только в начале алгоритма, а после этого ничего не гарантировано, даже в идельном(второй пример) случае, поэтому я не дал начальную точку, чтобы показать, что при больших A, b, x_0 метод ничего не будет гарантировать.

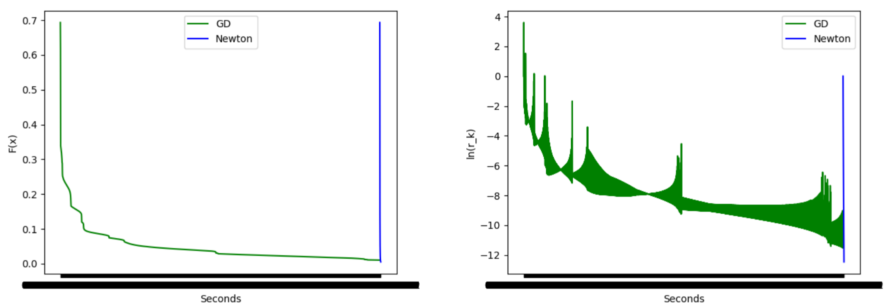
0.3 2-ой эксперимент

Размерности пространств: 10, 50, 100 и 200.
Величина κ: $100 + 10k \ \forall k \in \mathcal{N}_{90}$

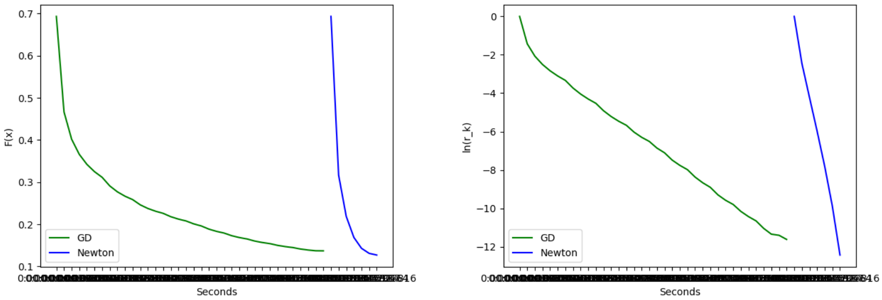
Из графика видно, что количесьтво итераций не сильно зависит от κ, и они "почти не зависимы". Еще из графика видно, что, если фиксируем маленькую δ, то вероятность того, что количество итераций пренадлежит на множестве $[m - \delta, m + \delta]$ самый высокий при m примерно равно 20. Ну можно и считать, что мат.ож. количество итераций $= 20$. Еще можно заметить, что при больших размерностей количество итераций более стабильно, чем маленьких. Но, при этом, при маленьких размерностей количество итераций стабилизируеться при увелечении κ.

0.4 3-ий эксперимент

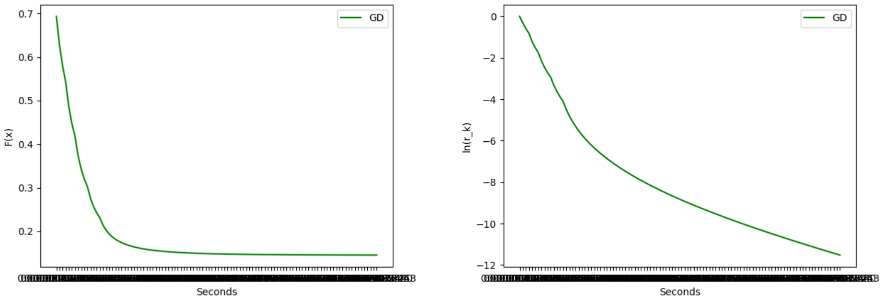
Gisette



W8A



Real-sim



В Real-sim-е метод Нютона не сходил, поэтому в графике нарисовал только градиентный спуск. Данные оказались такие, что во всех случаях Градентный спуск выигрывает метода Нютона. Асимптотика Градиентного спуска: $\mathcal{O}(nm)$, где m и n размерности матрицы A. Асимптотика так сложилась из-за вычитании вектора Ax. Асимптотика метода Нютона: $mathcalO(n^3 + mn)$. n^3 добавляеться из-за вычислении обратной матрицы.