Национальный исследовательский университет ИТМО
Факультет информационных технологий и программирования
Прикладная математика и информатика

Методы оптимизации

Отчёт по лабораторной работе №3

Работу выполнили:
Щербина К. А.
Чеканова П. А.
Викулаев И.А.

Преподаватель:
Станислав Ким

Санкт-Петербург
2023г.

Задание 1.

Перед нами стоит задача реализации методов Gauss-Newton и Powell Dog Leg для решения нелинейной регрессии.

Gauss-Newton

При решении какой-либо регрессии мы можем столкнуться с методом наименьших квадратов и поиска минимума полученной функции. Для этого, одним из оптимальных решений будет метод Гаусса-Ньютона.

Метод Гаусса-Ньютона - это итерационный численный метод нахождения решения задачи наименьших квадратов. Для этого метода требуется матрица Якоби - матрица первых производных функции по переменным.

На каждой итерации пересчёт весов будет происходить по формуле:

$$\boldsymbol{\beta}^{(s+1)} = \boldsymbol{\beta}^{(s)} - \left(\mathbf{J_r}^{\mathsf{T}} \mathbf{J_r}\right)^{-1} \mathbf{J_r}^{\mathsf{T}} \mathbf{r}(\boldsymbol{\beta}^{(s)})$$

Считать за квадрат - дорого и долго. Поэтому можем вспомнить, что этот метод - улучшенная версия метода Ньютона, а значит, что мы можем свести формулу выше к :

$$\boldsymbol{\beta}^{(s+1)} = \boldsymbol{\beta}^{(s)} - \mathbf{H}^{-1} \mathbf{g}$$

Где H - гессиан функции минимизации. Его можно посчитать по формуле:

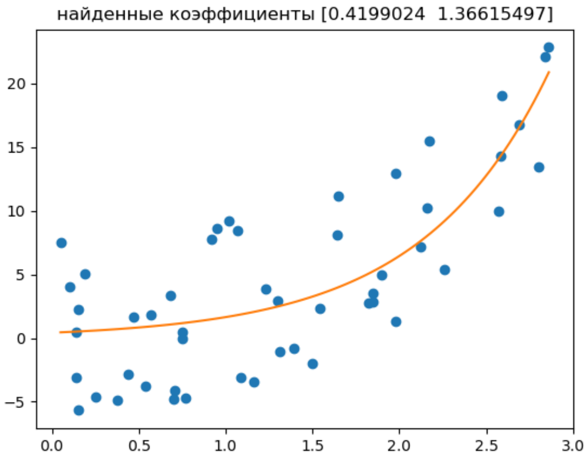
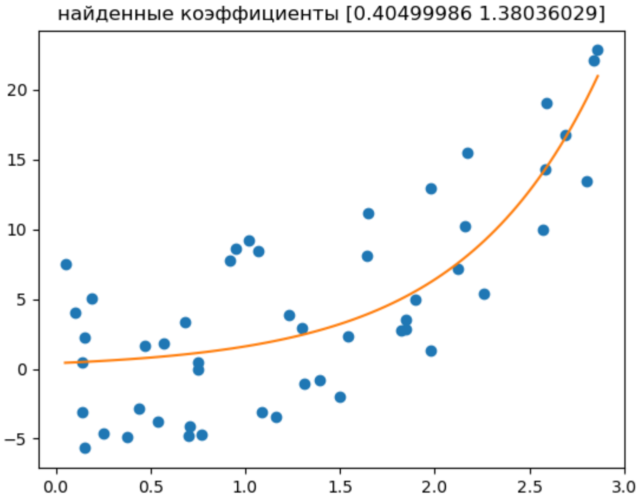
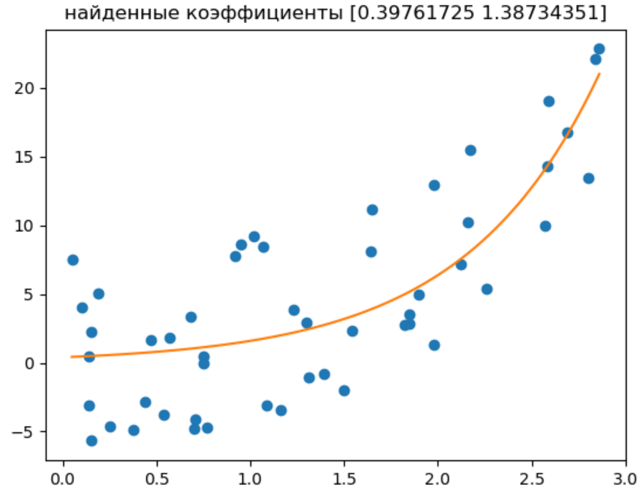
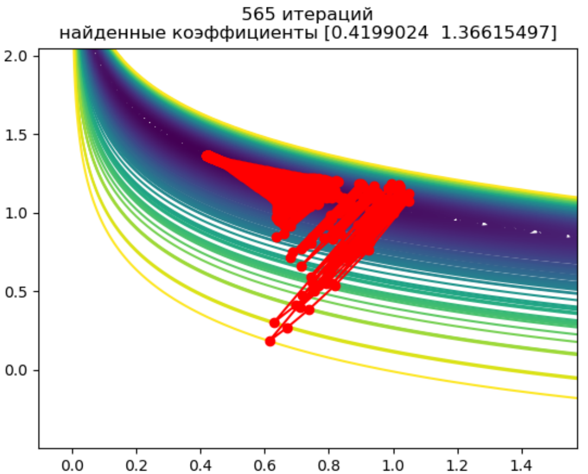
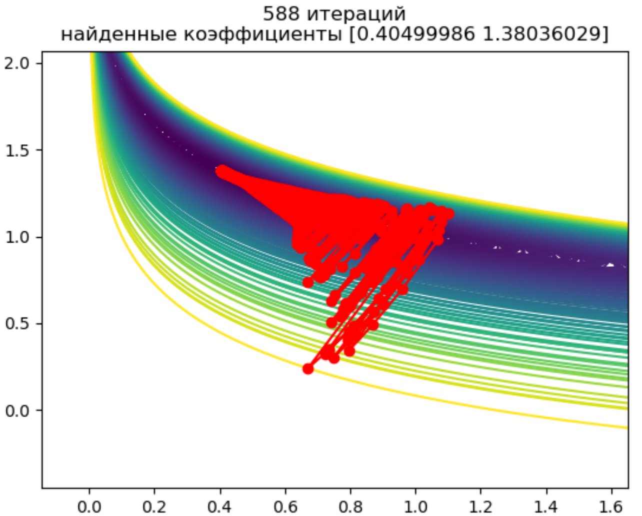
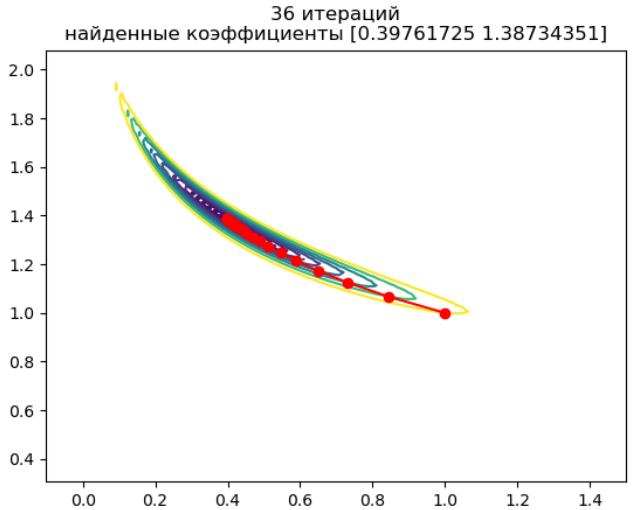
$$H_{jk} = 2 \sum_{i=1}^{m} \left(\frac{\partial r_i}{\partial \beta_j} \frac{\partial r_i}{\partial \beta_k} + r_i \frac{\partial^2 r_i}{\partial \beta_j \partial \beta_k} \right).$$

Однако предполагая, что второе слагаемое много меньше первого, метод будет его игнорировать, от чего достаточно тривиально выводится равенство первых двух формул.

В качестве нелинейной регрессии мы реализовали экспоненциальную.

Для сравнения будем рассматривать как с одним и тем же тестом справляются:

- Градиентный спуск
- ADAM
- Новый метод

<table>
<tr><th>Градиентный спуск</th><th>ADAM</th><th>Gauss-Newton</th></tr>
<tr>
<td>
</td>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td>
</td>
<td>
</td>
<td>
</td>
</tr>
</table>

Не смотря на то, что по результатам предыдущей лабораторной работы ADAM показал ошеломляющий результат, в этот раз он не отличается при наилучших входных параметрах от градиентного спуска.

Powell Dog Leg

Метод Powell Dog Leg, также называемый гибридным методом Пауэлла, представляет собой итеративный алгоритм оптимизации для решения нелинейных задач наименьших квадратов, представленный в 1970 году Майклом Дж. Д. Пауэллом. Аналогично алгоритму Левенберга–Марквардта, он сочетает алгоритм Гаусса–Ньютона с градиентным спуском, но использует trust region. На каждой итерации, если шаг из алгоритма Гаусса–Ньютона находится в пределах trust region, он используется для обновления текущего решения. Если нет, то алгоритм ищет минимум целевой функции вдоль самого крутого направления спуска, известного как точка Коши. Если точка Коши находится за пределами доверительной области, она усекается до границы последней и принимается в качестве нового решения. Если точка Коши находится внутри доверительной области, новое решение принимается на пересечении границы trust region и линии, соединяющей точку Коши и шаг Гаусса-Ньютона (dog leg step).

Метод Powell dogleg является одним из наиболее эффективных методов для решения задачи безусловной оптимизации с ограничениями на длину шага. Используется в случае, если матрица Гессе (или ее приближение) положительно определена.

Trust-region метод (TRM) является одним из самых важных численных методов оптимизации в решении проблем нелинейного программирования (nonlinear programming problems). Метод базируется на определении региона вокруг лучшего решения, в котором квадратичная модель аппроксимирует целевую функцию.

Trust-region метод использует квадратичную модель. На каждой итерации шаг вычисляется путем решения следующей квадратичной задачи:

$$min_{|p| \le \Delta k} \ m_k(p) = f(x_k) + p^T \nabla f(x_k) + \frac{1}{2} p^T \nabla^2 f(x_k) p \ ;$$

$$\Delta k > 0 - trust\, region \text{ радиус} ; \ \nabla^2 f(x_k) - \text{ Гессиан} ;$$

После нахождения шага (p_k) мы определяем следующее соотношение:

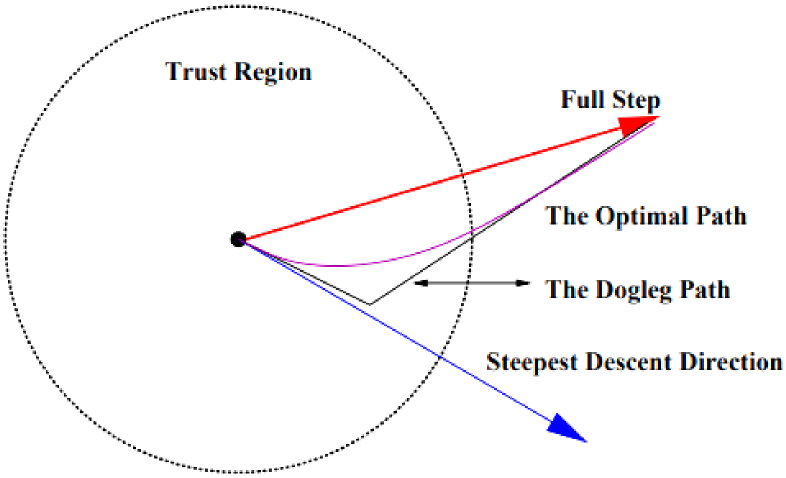
$$r_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)} \ ;$$

Знаменатель всегда должен быть неотрицателен, поскольку шаг p_k получается путем минимизации квадратичной модели m_k по региону, который включает в себя шаг $p = 0$. Отношение используется для определения преемственности шага и последующего обновления trust-region радиуса.

Если $p_k < 0$ или $p_k \approx 0$, то уменьшим trust-region область. Если же $p_k \approx 1$, тогда модель хорошо соответствует целевой функции. В данном случае следует расширить trust-region на следующей итерации. Иначе trust-region менять не будем.

Когда Δ_k весьма мало, ограничение $|p| \leq \Delta_k$ гарантирует, что квадратный член в модели m(p) оказывает небольшое влияние на решение. Реальное решение p(Δ) аппроксимируется также, как если бы мы оптимизировали линейную функцию $f + g^T p$ при условии $|p| \leq \Delta$, тогда: $p^*(\Delta) = -\Delta\frac{\nabla f}{|\nabla f|}$

Для средних значений Δ решение $p^*(\Delta)$, как правило, следует за криволинейной траекторией, как показано на картинке:



Dogleg метод аппроксимирует криволинейную траекторию $p^*(\Delta)$ линией состоящей из двух прямых:

1. Прямая вдоль направления наискорейшего спуска (steepest descent direction) и определяется следующим образом:

$$p^U = -\frac{\nabla f^T \nabla f}{\nabla f^T \nabla^2 f \nabla f}\nabla f$$

2. $p^{(\nabla^2 f)} = -(\nabla^2 f)^{-1}\nabla f$ – суть вектор решение для минимума квадратичной модели в один шаг считается из точки p^k.

Формально мы обозначим траекторию $\bar{p}(\tau)$, где $\tau \in [0, 2]$;

$$\bar{p}(\tau) = \begin{cases} \tau p^U, & 0 \le \tau \le 1, \\ p^U + (\tau - 1)(p^{(\nabla^2 f)} - p^U), & 1 \le \tau \le 2 \end{cases}$$

Для поиска τ необходимо решить квадратное уравнение, следующего вида:

$$\left| p^U + \tau * (p^{(\nabla^2 f)} - p^U) \right|^2 = \Delta^2$$

$$(p^U)^2 + 2\tau(p^{(\nabla^2 f)} - p^U)p^U + \tau^2(p^{(\nabla^2 f)} - p^U)^2 = \Delta^2$$

Находим дискриминант уравнения:

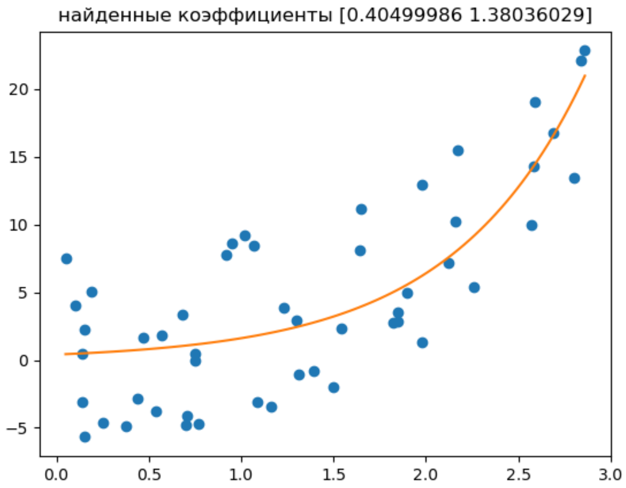
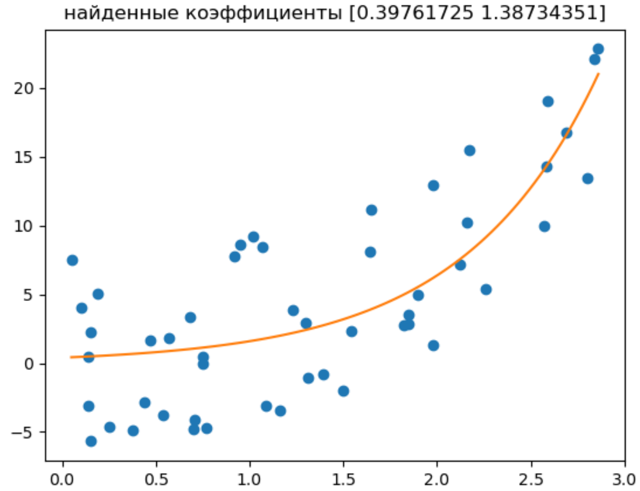
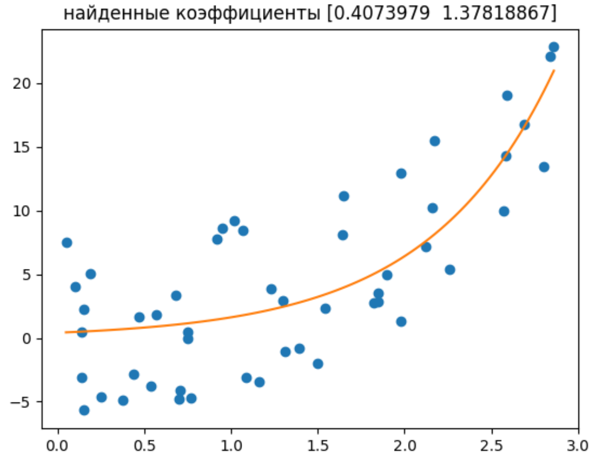
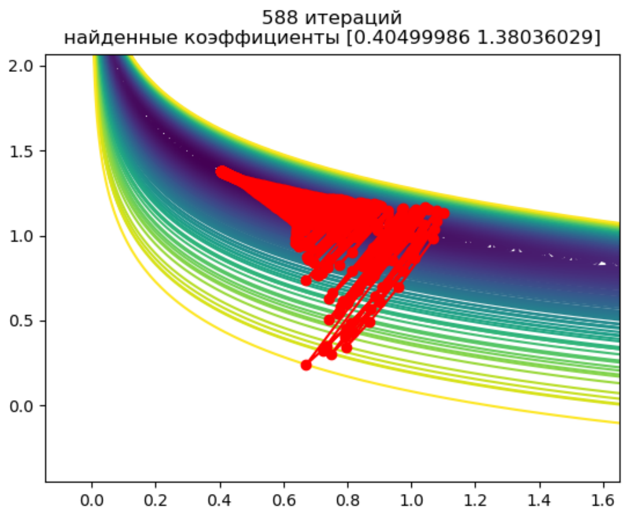
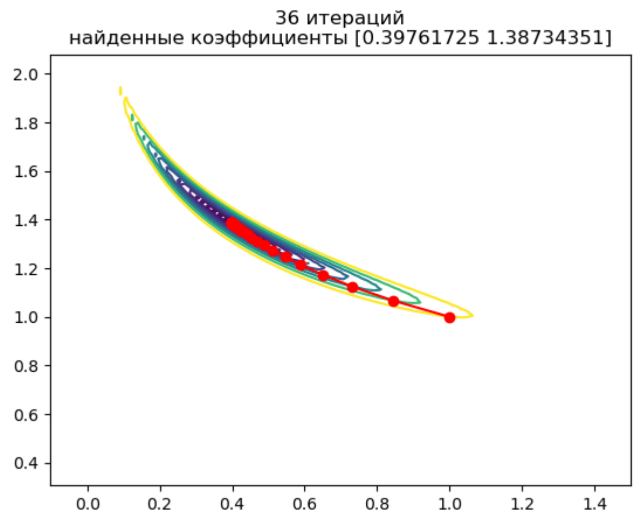
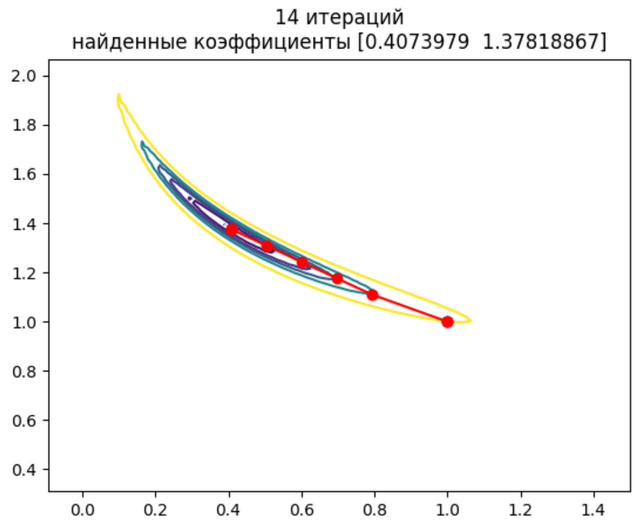
$$D = 4(p^{(\nabla^2 f)} - p^U)^2 (p^U)^2 - 4(p^{(\nabla^2 f)} - p^U)^2 ((p^U)^2 - \Delta^2)$$

$$\sqrt{D} = 2(p^{(\nabla^2 f)} - p^U)\Delta$$

Корень уравнения равен:

$$\tau = \frac{\Delta - p^U}{p^{(\nabla^2 f)} - p^U}$$

Dogleg метод выбирает p_k, чтобы минимизировать модель вдоль этого пути. В действительности нет необходимости выполнять поиск, поскольку dogleg путь пересекает границу trust-region только один раз и точка пересечения может быть найдена аналитически.

<table>
<tr><th>ADAM</th><th>Gauss-Newton</th><th>Powell Dog Leg</th></tr>
<tr>
<td>
</td>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td>
</td>
<td>
</td>
<td>
</td>
</tr>
</table>

Метод Powell dogleg может быть эффективным для поиска минимума функции в задачах нелинейной оптимизации. Однако он требует значительных вычислительных ресурсов и может страдать от сильной нелинейности функции или неудачного выбора начального приближения.

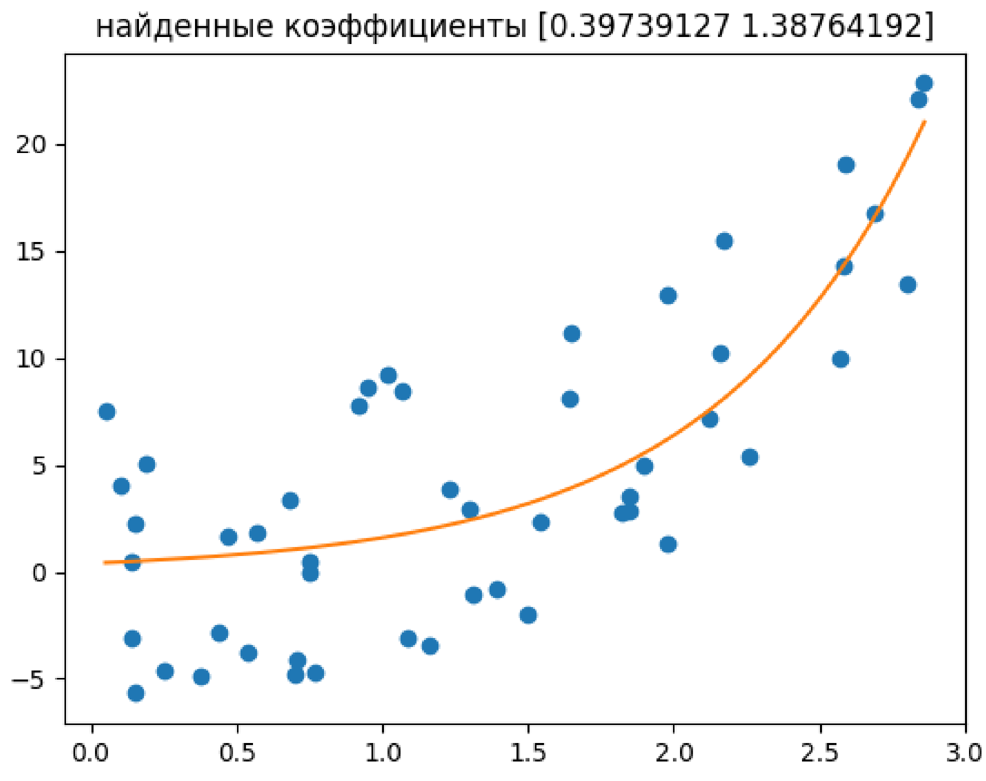
Задание 2.

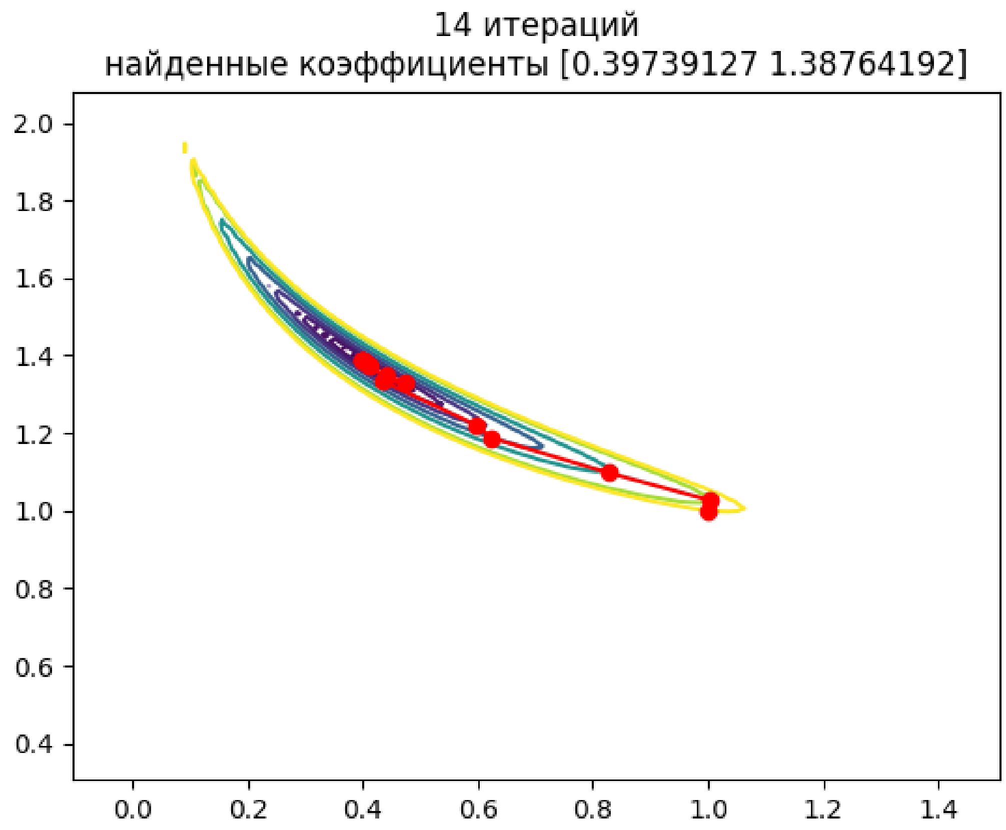
Метод BFGS - это итерационный метод для решения задач нелинейной оптимизации без ограничений. Он относится к классу квазиньютоновских методов, которые аппроксимируют матрицу Гессе функции потерь, используя только оценки градиента. Метод BFGS определяет направление спуска по формуле: $p_k = - B_k^{-1} \nabla f(x_k)$
,
где B_k – это приближение к матрице Гессе в точке x_k, которое обновляется на каждой итерации, а $\nabla f(x_k)$ – это градиент функции в точке x_k.

Затем используется одномерный поиск по направлению p_k для нахождения следующей точки x_{k+1} путем минимизации по скаляру α: $x_{k+1} = x_k + \alpha p_k$.
Обновление матрицы B_k удовлетворяет условию квазиньютоновости: $B_{k+1} s_k = y_k$, где $s_k = x_{k+1} - x_k$ и $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$. Формула обновления имеет вид: $B_{k+1} = B_k + ((y_k y_k^T) \div (y_k^T s_k)) - ((B_k s_k s_k^T B_k) \div (s_k^T B_k s_k))$.

Метод BFGS является одним из наиболее широко используемых второго порядка методов для численной оптимизации и часто применяется для обучения машинного обучения, таких как логистическая регрессия.

Для тестирования возьмём ту же функцию, что и в первом задании.





Как мы видим BFGS показал на нашей функции очень хороший результат. Этот метод явно выигрывает у всех остальных и при этом не уступает в точности методу Гаусса-Ньютона.

[Код реализации методов]