

SHORT USER GUIDE

28 июля 2021 г.

1 Введение

0 здравствуй, случайный странник Интернета!

Если ты читаешь эти строки, значит тебе очень повезло. Перед тобой открыт мир удивительных возможностей, который впустит тебя буквально через минуту. Но обо всём по порядку.

Математический бот. Их довольно много в интернете, но их функционал обычно ограничивается подсчётом указанного пользователем выражения или решением незамысловатых уравнений. Есть боты специального назначения - те, что могут решать дифференциальные уравнения, высчитывать **нули дзета-функции Римана** или решать задачи квадратичного программирования. Наш бот к ним не относится - в Ваших руках инструмент, с одной стороны, чрезвычайно простой, но в тоже время невероятно универсальный. А именно, Вы сможете:

- решать задачи одномерной и многомерной условной оптимизации
- находить корни алгебраических и трансцендентных уравнений
- строить графики одной или нескольких одномерных функций

Заинтересовались? Нет? Читайте дальше!

2 Интерфейс для Линуксоидов

Warning!

Эта глава - в основном для любителей командной строки. Если Вы по каким-то причинам не относитесь к этому множеству, используйте пошаговый GUI бота. Он приправит беседу достаточным количеством текста, что точно не позволит Вам запутаться.

Прежде чем перейти к основным возможностям бота, опишем его синтаксис.

2.1 Оптимизация

Чтобы провести процедуру оптимизации, нужно определить три обязательных параметра:

- целевую функцию (например, $(\sin(x) + 2) * (\cos(x))$)
- множество, на котором будете исследовать функцию (например, $x \in [-10, 1000]$). Не обязательно подмножество области определения функции.
- минорирующую константу для функции на заданном множестве, если таковая известна. Да, это может показаться странным, но это увеличит эффективность встроенных в бот алгоритмов. К примеру, про функцию $(\sin(x) + 2) * (\cos(x))$ известно, что она строго положительна на своей области определения, а потому можно задать этот параметр равным 0. Если Вы не хотите заморачиваться с этим (или точно знаете, что функция не ограничена снизу), пишите любое вещественное число.

Таким образом, для примера выше необходимо ввести команду:

```
» optimise (sin(x) + 2)*(cos(x)) for x in [-10, 1000] | minorant = 0
```

Ключевые слова **optimise**, **for**, **minorant** и разделитель **|** обязательны. В общем случае одномерной оптимизации, синтаксис следующий:

```
» optimise <function(arg)> for <arg bindings> | minorant = <real number>
```

Многомерная оптимизация? Не вопрос:

```
» optimise <function(arg1, ..., argN)> for <arg1 bindings>, ..., <argN bindings> | minorant = <real number>
```

Ограничения на переменные (**<arg bindings>**) можно задавать в различных формах:

- $\arg \in [a, b]$
- $\arg \in: a \rightarrow b$
- $a \leq \arg \leq b$

К сожалению, ограничения принимаются пока лишь в покоординатном интервальном виде. По поводу оптимизации без ограничений смотрите [ссылку](#)

2.2 Решение уравнений

Чтобы провести процедуру решения уравнения (по поводу решения системы смотрите [ссылку](#)), нужно определить лишь два обязательных параметра:

- уравнение, например $\sqrt{\lg(x)} = \ln(x) + 10$

- множество, на котором осуществляется поиск корней, например $-10 < x < 200$. Не обязательно подмножество области определения функции.

Для приведённого выше примера запрос должен выглядеть так:

```
» solve sqrt(tg(x)) = ln(x) + 10 for -10 < x < 200
```

Ключевые слова **solve** и **for** обязательны. В общем случае синтаксис следующий:

```
» solve <equation(arg1, ..., argN)> for <arg1 boundings>, ..., <argN boundings>
```

2.3 Построение графиков

Чтобы построить график одной или нескольких одномерных функций (увы, графиков дзета-функции здесь не будет) нужно определить два обязательных аргумента:

- функции, например $x^3 / (e^x - 1)$ и $\text{tg}(x)$
- множество, на котором будете строить график (например, $x \text{ in: } 0 \rightarrow 40$). Не обязательно подмножество пересечения областей определения функций.

Для приведённого выше примера запрос должен выглядеть так:

```
» plot x**3 / (e**x - 1) && tg(x) for x in: 0 → 40
```

Ключевые слова **plot**, **for** и разделитель **&&** обязательны. Синтаксис для общего случая:

```
» plot <function1(arg)> && ... && <functionN(arg)> for <arg boundings>
```

3 F.A.Q.

- **Q: Почему бот не может в безусловную оптимизацию?**

А: Может. Используйте суперпозицию функций. Чтобы оптимизировать $f(x)$ на \mathbb{R} , вводите в качестве целевой функции $f(\text{tg}(x))$ на интервале, содержащем $[-\pi/2, \pi/2]$.

- **Q: Почему нельзя вводить системы уравнений?**

А: И снова можно. Система $f_1(x) = g_1(x), \dots, f_N(x) = g_N(x)$ равносильна одному уравнению $(f_1(x) - g_1(x))^2 + \dots + (f_N(x) - g_N(x))^2 = 0$, которое и следует вводить.

- Q: Почему так многобукв в GUI?

A: Хотите меньше текста - используйте интерфейс командных запросов :)

- Q: Бот выдал оптимальную точку, выходящую за рамки ограничений на аргументы функции. Как это понимать?

A: Увы, такое может случаться. В основе бота - эвристические алгоритмы, в частности генетический. Отталкиваясь от заданных пользователем ограничений, алгоритм способен найти оптимум вне заданного множества, и сходится в него. Алгоритмы, использующие проецирование точек на заданное множество - предмет следующих обновлений.

- Q: Могу ли я использовать 'e' или 'pi' в качестве переменных?

A: Нет.

Warning!

Бот имеет массив встроенных математических констант. К ним относятся:

e	$e = 2.71828...$	fi	$\varphi = 0.61803...$
pi	$\pi = 3.14159...$	Fi	$\Phi = 1.61803...$

- Q: Почему не WolframAlpha?

A: На то есть несколько причин:

- totally free. Время вычислений не ограничено так жёстко, как в бесплатной версии WolframAlpha.
- В многомерной оптимизации бот превосходит WolframAlpha по эффективности. Не верите? Попробуйте!
- Вся история вычислений - вместе с Вами!
- Алгоритмы, которые постоянно совершенствуются. Более того, Вы можете повлиять на процесс создания бота, написав админам.
- Переписывайтесь с друзьями и ботом, не выходя из VK!
- Если у вас нет друзей и вам грустно, бот готов составить вам компанию в любое время суток, ведь ему тоже грустно...