

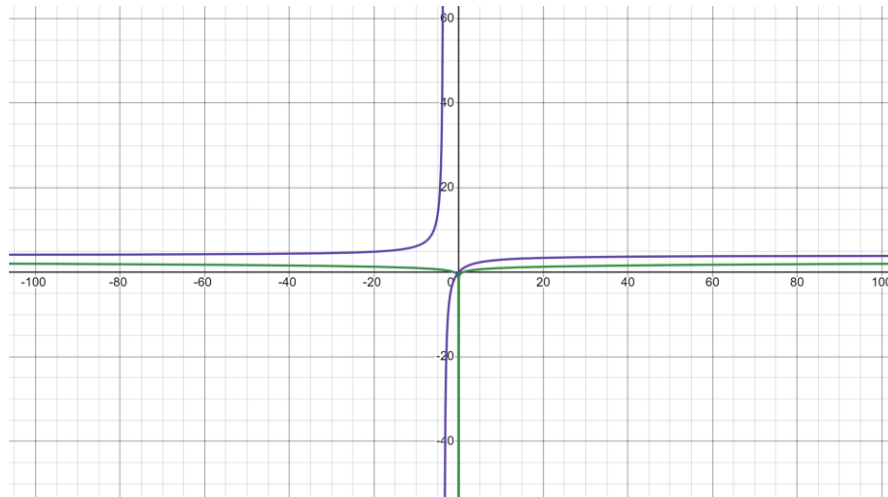
نکته ۱: اولین approach من به این مسئله این بود که طبق اسلاید های درس، هر بار به تعداد جمعیت، والد انتخاب میکردم و بچه های جدید میساختم و جمعیت قبلی را کلا پاک میکردم. جواب خوبی با این روش نگرفتم. با چاپ کردن مقدار fitness ها متوجه شدم که خیلی اوقات جمعیت جدید از جمعیت قبلی فیتنس بدتری دارد و کروموزوم های خوب که پتانسیل به جواب رسیدن را داشتند رو از دست میدادم. پس تصمیم گرفتم به جای این کار، تو هر مرحله بچه های جدید رو بسازم و اضافه کنم به جمعیت قبلی.

اما اگر فقط اضافه میکردم جمعیت با شیب تند زیاد میشد و از نظر زمانی کدم خیلی کند میشد، به همین دلیل بعد از اینکه به اندازه نصف جمعیت قبلی، بچه جدید اضافه کردم، همه کروموزوم ها را به ترتیب تابع فیتنس مرتب کردم و اضافه جمعیت را از کروموزوم هایی که فیتنس پایین داشتند حذف کردم. اینطوری تعداد جمعیت همیشه یک عدد ثابت و فیتنس اونها رو به بهبودی میره.

با این روش خیلی نتیجه بهتر شد، برای تمام توابع خطی یک بعدی که امتحان کردم جواب درست گرفتم.

نکته ۲: اول احتمال mutation را ۵ درصد گذاشته بودم، بعد از بررسی دیدم خیلی پیش میاد که ۱۰ تا ۲۰ جنریشن پشت سر هم هیچ بهبودی توی فیتنس نداشته باشن، به خاطر همین اون احتمال رو کردم ۱۵ درصد، و نتیجه خیلی بهتر شد. (سرعت همگرایی به جواب بیشتر شد.)

نکته ۳: نقاط ورودی رو از $\log(|x|)$ انتخاب کردم (تو عملگرهای درختم وجود نداره)، و کدم یک تابع خطی تحویل داد که به طرز عجیبی به تابع اصلی شبیهه. (در شکل زیر تابع سبز لگاریتمی هست و بنفش خروجی کد من هست)



نکته ۴: نقاط ورودی را از تابعی با رفتار متفاوت در بازه ها انتخاب کردم، تابع زیر:

```
def func3(values):  
    if values['x'] <= 0:  
        return 2**values['y']  
    if values['x'] <= 50:  
        return 3 - values['y']  
    return math.sin(values['y'])
```

و در نهایت خروجی به من $\sin(y)$ را بازگرداند.

نکته ۵: توابع ساده با ورودی ۳ بعدی هم امتحان کردم و اوکی بود (داخل کد هست). عموماً خطای کدم این بود که تابعی خروجی میداد که به اندازه یک عدد ثابت با تابع اصلی فاصله دارد (به نظرم چیز خوبییه)

نکته آخر: چون شروع الگوریتم از جمعیت تصادفی است و طی مسیر هم به شانس خیلی بستگی دارد، هر باری که اجرا میکنم جواب متفاوتی میدهد، من در این گزارش مشاهدات خودم رو نوشتم.