

بسمه تعالی

گزارش تکلیف دوم مبانی هوش محاسباتی

حسین محمدی ۹۵۳۳۰۸۱

بازنمایی مسئله:

در این مسئله، تمام نقاط ورودی در لیستی به نام `Points` ذخیره می‌شوند. پس از آن لیستی از کروموزوم‌ها طبق قالب تعریف پروژه ساخته می‌شود. اندازه لیست به اندازه طول ورودی است. در هر کروموزوم ۲ ژن وجود دارد که ضرایب خط z را معلوم می‌کنند. هر z به کمک نقطه متناظر آن در آرایه `points` محاسبه می‌شود. هر کروموزوم متغیری برای ذخیره احراف معیار خود دارد (`score`).

`Chromosome.py`

این کلاس برای نگهداری اطلاعات هر کروموزوم، شامل ضرایب و انحراف معیار است. تابع `evaluate` برای محاسبه شایستگی (انحراف معیار) و تابع `normalize` برای نرمال سازی نقاط پس از هر جهش استفاده می‌شوند.

`Es.py`

الگوریتم تکاملی در این کلاس اجرا می‌شود. مراحل آن را در زیر بررسی می‌کنیم. تولید جمعیت اولیه: به تعداد نقاط ورودی به مسئله، کروموزوم ساخته و آن‌ها را در لیست `pop` ذخیره می‌کنیم.

مراحل پایین در حلقه‌ای و به تعداد نسل انجام می‌شوند.

انتخاب والدین: والدین جدیدی به صورت تصادفی به شکل جفت و به تعداد لامبدا انتخاب می‌شوند. نتیجه آن لیستی از جفت والدین خواهد شد.

`Cross over`: والدین انتخاب شده با احتمال `crossover_probability` که از پارامترهای مسئله است، با هم کراس اور می‌شوند. به این صورت که به شکل تصادفی ضرایب اول یا دوم

والد اول به عنوان ضریب اول فرزند و ضریب دوم هم به صورت تصادفی از والد دوم انتخاب می‌شود. اگر احتمال کراس اور در محدوده `crossover_probaility` وجود نداشت، کراس اور صورت نمی‌گیرد. نتیجه این کار، لیستی از فرزندان جدید است.

جهش: جمعیت این نسل و فرزندان تولید شده با استفاده از تابع نرمال گوسی جهش می‌یابند. به این صورت که ضرایب هر کدام با مقدار `noise` که خروجی تابع نرمال است جمع می‌شود. پارامترهای `sigma`, `mutation_rate` قابل کنترل است و احتمال جهش و قدم جهش را تغییر می‌دهد.

شایستگی: شایستگی جمعیت جهش یافته که شامل جمعیت کنونی و فرزندان است، محاسبه می‌شود. شایستگی رابطه مستقیم با انجراف معیار هر کروموزوم از داده‌های ورودی دارد.

انتخاب: روش انتخاب شده $\mu + \lambda$ خواهد بود. به این صورت که هر دو گروه جمعیت کنونی و فرزندان در یک لیست بر حسب شایستگی سورت شده و سپس، به اندازه نصف μ از بالای لیست و نیمی دیگر از پایین لیست برگردانده می‌شود. لیست جدید همان جمعیت نسل بعد است.

نکات:

- در هر نسل، شایستگی بهترین کروموزوم، بدترین آن و میانگین شایستگی چاپ می‌شود.
- شرط خاتمه الگوریتم تکاملی پایان یافتن تعداد نسل خواهد بود. در نسل‌های آخر شاهد همگرایی شایستگی میانگین هستیم.

تأثیر پارامترها:

پارامترهای مسئله به صورت زیر هستند:

- تعداد نسل ۴۰ در نظر گرفته شده است.

- اگر احتمال کراس اور را افزایش دهیم مسئله دیرتر همگرا می شود و اگر احتمال آن را پایین بیاوریم حرکتی نیز تغییر نسل ها نسبت به هم کم شده و ممکن است باعث همگرایی زودرس شود.
 - اگر تعداد فرزندان هر نسل کمتر شود حرکتی صورت نگرفته و ممکن است به جواب نرسیم.
 - اگر احتمال جهش بالا باشد حرکت بیشتری صورت می گیرد و فضای بیشتری جستجو می شود.
 - اگر قدم های جهش زیاد شوند. حرکت مشابه رندوم واک می شود.
 - جستجوی بهینه با مقدار میانه ای از ضرایب بالا انجام می گردد.
- ضرایب در نظر گرفته شده به شکل زیر است:

$\mu = 10$

$\lambda = 7 * \mu$

crossover_probability = 0.4

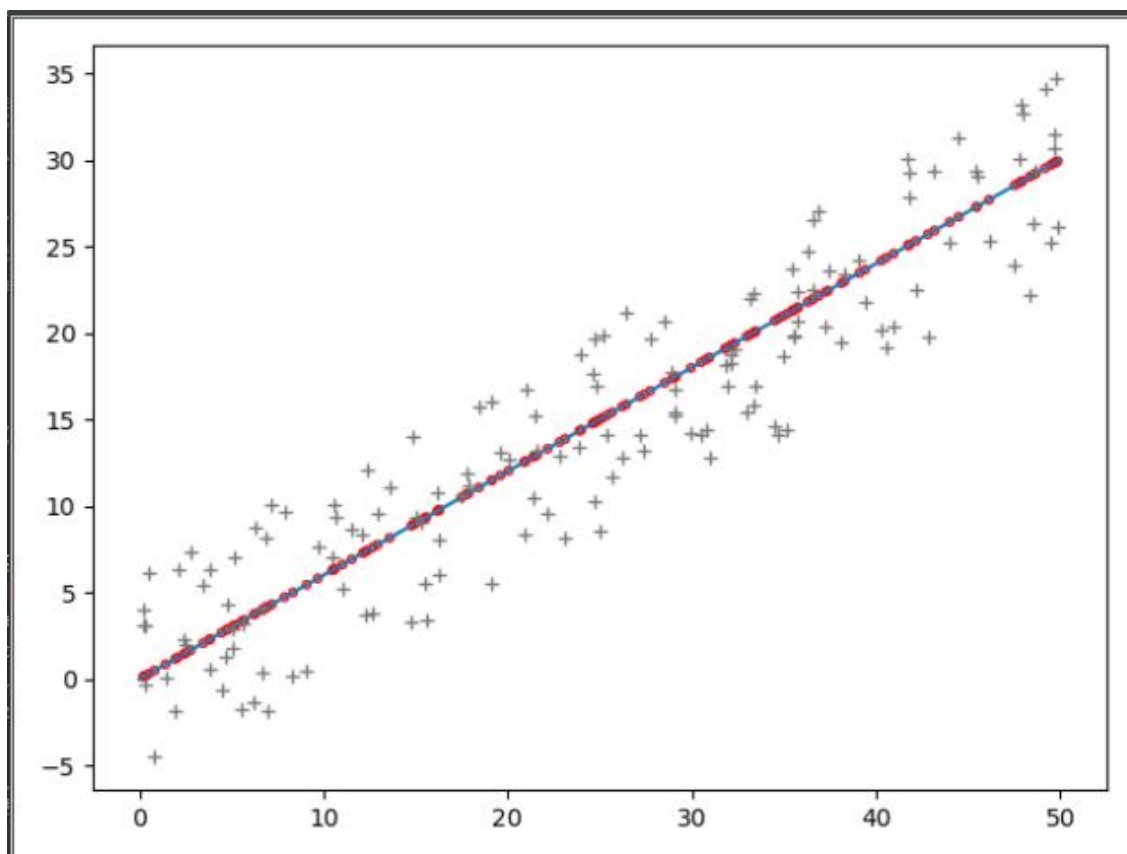
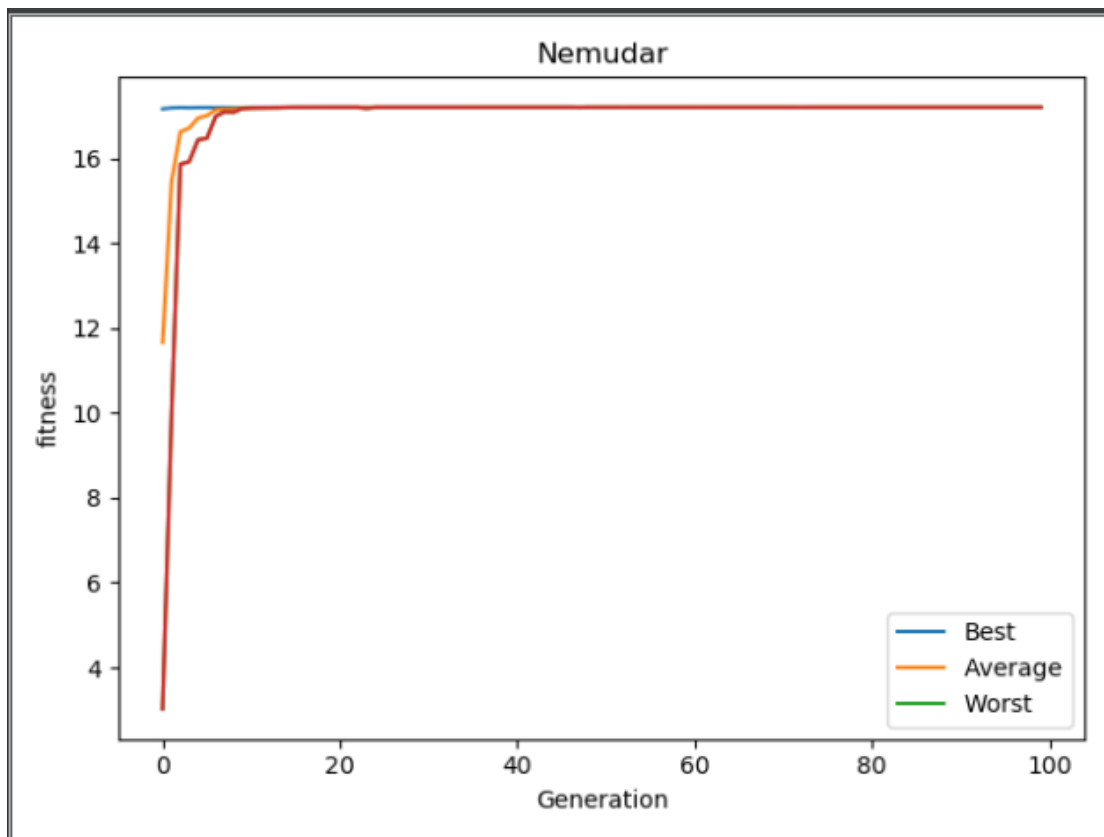
population_size = μ

chromosome_length = 2

iteration_size = 100

نتایج آزمایش:

Dataset1



Dataset 2

