



UNIVERSITY OF GUILAN

January 20, 2020

GENETIC ALGORITHM FRAME WORK

Arash Aghaei – Pegah Nakhsaz

موضوع پروژه ما طراحی فریم ورکی خلاصه برای پیاده سازی الگوریتم های ژنتیک در زبان پایتون می باشد؛ که با استفاده از function های موجود در این پروژه بتوان به پیاده سازی بهتر و راحت تر الگوریتم های ژنتیک در پایتون پرداخت .

خب نمونه هایی از این نوع فریم ورک ها وجود دارند که می توان به پرکاربرد ترین آن ها یعنی DEAP در پایتون اشاره کرد .

از آن جایی که هدف از پروژه درس هوش محاسباتی یاد گیری هرچند بهتر مباحث این درس و همچنین پیاده سازی آموخته ها بود؛ ما تصمیم گرفتیم که موضوع پروژه ما پیاده سازی الگوریتم های ژنتیک باشد که به جرأت می توان گفت یکی از اساسی ترین و مفهومی ترین بخش های این درس می باشد.

در ادامه با توضیحات پروژه همراه شما خواهیم بود:

در بخش import های پروژه اکثراً لیبراری های خود پایتون هستند مانند math و random و repeat و ... تنها لیبراری که خارج از پایتون باید import شود numpy می باشد که برای کار با اعداد و کارهای علمی و .. است

در کل در الگوریتم ژنتیک ما نیاز به مراحل زیر برای پیاده سازی داریم:

Initialization

Selection

Crossover

Mutation

ما در این پروژه سعی کردیم که برای هر یک از موارد فوق چند مدل مختلف ارایه کنیم

برای قسمت initialization یا مقدار دهی اولیه ما دو مدل repeat initialization و iteration initialization را پیاده سازی کردیم که در مدل اول به سری فانکشن به آن داده می شود و از آن n بلوک می سازیم و از بلوک ها یک لیست ایجاد می کنیم سپس آن را در container ریخته و به خروجی می دهیم

برای مدل دوم ما مانند مدل اولیه عمل کرده و تفاوت آن این است که تنها یک بار ورودی ها را گرفته و در کانتینر ریخته و به خروجی داده یا return می کنیم.

برای بخش selection نیز ما از سه مدل استفاده کردیم با نام های random selection و tournament selection و roulettselection

که در مدل اول به صورت رندم اشخاصی را در جمعیت انتخاب کرده و آن ها را به عنوان خروجی return می کنیم

در مدل دوم مانند مدل اول از رندم استفاده کرده با این تفاوت که بهترین آن ها را از نظر fitness انتخاب کرده برخی از آن ها را جنریت کرده و برای خروجی در تابع return می کنیم

در مدل سوم از چرخ رولت استفاده کرده و آن ها را از طریق فیتنس هایشان sort می کنیم و سپس به خروجی ارایه می کنیم

در crossover نیز در این پروژه ما از سه مدل onepoint و twopoint و uniform استفاده کردیم

که برای مدل اول مانند مثال اسلاید عمل شده و کراس اور بین دو کروموزوم انجام شده و قسمت دوم کروموزوم اول و قسمت اول کروموزوم دوم و برای کروموزوم دیگری عکس عمل فوق را انجام می دهیم

برای مدل دوم مانند مدل اول عمل کرده با این تفاوت که کروموزم ها را به سه قسمت تقسیم می کنیم و عمل جابه جایی را انجام می دهیم .
برای مدل سوم از یک آستانه استفاده کرده و با استفاده از رندم و مقایسه دو کروموزم و کروموزوم های ساخته شده به پیاده سازی crossover می پردازیم

برای mutation نیز gaussian و polynomial و shuffle ndex و flip bit استفاده شده است که در مدل های اول و دوم جهش با استفاده از برخی متغیر های پیچیده به صورت هوشمندان انجام شده و دو مدل آخر نیز جهش را به صورت ساده و همانند مفهوم آن در درس و مدل آخر جهش را فقط در کروموزوم های باینری پیاده سازی می کند.

در انتهای کد نیز مثالی از ادعای خود بنا بر اینکه با استفاده از پلنفرم ساخته شده تنها با چند خط کد می توان یک الگوریتم ژنتیک کامل برای تولید ژن جدید و... ساخت ارایه کرده ایم که در این مَثا کروموزوم ها به صورت رندم ساخته می شوند مانند زیر

[[۸, ۷, ۹, ۶, ۳, ۴, ۵, ۰, ۱, ۲], [۶, ۴, ۳, ۵, ۹, ۸, ۲, ۱, ۰, ۷]]

و از هر مدل یک نوع را می توان انتخاب کرده و پس از انجام مراحل خروجی زیر را می توان دریافت نمود.

([۶, ۴, ۴, ۶, ۳, ۰, ۵, ۱, ۳, ۷])

این frame work توانایی گسترش بالایی دارد و در صورت ادامه دادن کار بر روی آ» می توانیم بسیار کامل تر و با مدل های بیشتر برای پیاده سازی طیف های گسترده تر این الگوریتم پرداخت

