



دانشگاه صنعتی اصفهان

دانشکده مهندسی برق و کامپیوتر

تکلیف عملی سری چهارم

مبانی هوش محاسباتی

استاد درس : دکتر سمانه حسینی

دستیار آموزشی: بهناز عالی‌پور

زمان تحويل تکلیف: 4 تیر

بخش اول: آشنایی ابتدایی با ACO و PSO و مراحل پیاده سازی هر کدام

سوال اول (آشنایی با PSO)_(15نمره)

در این قسمت برای آشنایی بیشتر شما با مراحل الگوریتم PSO_part1.ipynb فایلی تحت عنوان PSO در اختیار شما قرار گرفته است و هدف این سوال یافتن نقطه بهینه یک تابع ساده ($f(x)=x^2$) است. و از شما خواسته شده که بخش هایی از این فایل را تکمیل کنید و نتیجه نهایی را گزارش دهید.

پس لازم است در قسمت Main PSO Loop بخش های زیر را تکمیل کنید:

1. بروزرسانی سرعت ذرات.
2. بروزرسانی مکان ذرات.
3. بروزرسانی بهترین جوابی که ذره پیدا کرده است (pbest).
4. بروزرسانی بهترین جوابی که جمعیت یافته است (gbest).

سوال دوم(آشنایی با ACO)_(20نمره)

در این سوال برای آشنایی بیشتر شما با مراحل الگوریتم ACO_part1.ipynb یک فایل تحت عنوان ACO است. در اختیار شما قرار داده شده است و هدف این سوال حل مسئله TSP با چهار شهر است.

برای سادگی صرفاً از 4 مورچه و یک iteration استفاده شده است و از شما خواسته شده تا بخش های ناقص کد را کامل کنید.

1. پس لازم است در قسمت while unvisited، قسمت Construct one tour، قسمت while unvisited را کامل کنید. این تابع برای ساختن tour استفاده می‌شود و مورچه به صورت رندم از یک شهر شروع به حرکت می‌کند، در قسمت while unvisited شما باید مشخص کنید مورچه چگونه شهر های دیگر را انتخاب می‌کند و یک دور را می‌سازد.
2. سپس باید در قسمت update_pheromone، تابع Update pheromone matrix را کامل کنید. وظیفه شما این است که فرمون را روی مسیری که مورچه پیدا کرده است را بروزرسانی کنید.

بخش دوم

سوال اول (30نمره):

اکنون که با مراحل و نحوه پیاده سازی PSO آشنا شدید، در این سوال قصد داریم که به کمک این الگوریتم یک ساختار بهینه برای یک شبکه عصبی پیدا کنیم.

در این سوال قرار است یک مدل شبکه عصبی بر روی داده های mnist آموزش داده شود.

دیتاست mnist شامل عکس هایی از ارقام دستنویس است، که برای اینکه مدل سریع تر آموزش ببینید صرفا از 3000 تا داده استفاده کرده ایم، ای مدل دارای سه لایه است و در 3 ایپاک آموزش می بینند.

قصد داریم با استفاده از الگوریتم PSO پارامتر هایی نظیر موارد زیر را تنظیم کنیم :

1. تعداد نورون ها در هر لایه (در این مدل از سه لایه استفاده شده پس باید تعداد نورون ها برای هر لایه مشخص شود)
2. تابع activation مناسب (سه مورد پیشنهادی: relu, tanh, sigmoid)
3. نرخ dropout

پس particle ها نماینده ترکیب مختلفی از پارامتر های مسئله هستند و هر particle از 5 مولفه $p[0]$ تا $p[4]$ تشکیل شده است.

- $P[0]$: تعداد نورون لایه اول
- $P[1]$: تعداد نورون لایه دوم
- $P[2]$: تعداد نورون لایه سوم
- $P[3]$: شماره activation function (از صفرتا 2 هر شماره نماینده یکی از توابع relu, tanh, sigmoid dropout است)
- $P[4]$: نرخ dropout

فایلی تحت عنوان PSO_part2.ipynb در اختیار شما قرار داده شده است و شما باید بخش های ناقص را تکمیل نمایید.

وظیفه شما از این قرار است:

1. تعریف fitness function مناسب برای این مسئله، درنهایت این تابع باید منفی accuracy را برای

هر مدل بازگرداند ، دلیل اینکه از منفی دقت استفاده میکنیم این است که الگوریتم PSO به طور پیش فرض یک الگوریتم کمینه ساز است در حالی که ما تمایل داریم در این مسئله میزان دقت مدل را افزایش دهیم.(Define the fitness function for PSO کامل کنید)

2. تنظیم پارامتر های الگوریتم PSO نظیر $w, c1, c2, \dots$ و مقدار دهی اولیه با پارامتر هایی نظیر (set PSO hyperparameters positions, velocities,...

3. تکمیل حلقه اصلی PSO (قسمت PSO main loop)

سوال دوم(35 نمره):

در این سوال قصد داریم مسئله زمان بندی کلاس ها را با استفاده از الگوریتم ACO حل کنیم.

در این مسئله 12 کلاس ، 6 استاد (هر کلاس توسط یک استاد تدریس می شود) و 6 زمان ممکن داریم.

این مسئله محدودیت هایی دارد :

1. در هر زمان حداکثر 3 کلاس می تواند برگزار شود.

2. یک استاد همزمان نمی تواند در چند کلاس حضور داشته باشد.

3. برخی اساتید صرفا در برخی زمان ها قابل دسترسی هستند و باقی زمان ها جزو محدوده زمان بندی کاریشان نمی باشد.

4. به دلیل اینکه دانشجویان در کلاس های متفاوتی ثبت نام کرده اند، نباید برخی کلاس ها در یک زمان باهم برگزار شوند.

هدف این سوال ارائه یک زمان بندی برای کلاس ها می باشد که بتواند تا حد ممکن چهار قید بالا را برآورده کند (برای این کار می توانید وزن های متفاوتی برای هر قید در نظر بگیرید).

نکته: هر قید که برآورده نشود متناسب با آن قید باید متغیر conflicts افزایش یابد.

به عنوان مثال در صورتی که بیش از 3 کلاس در یک زمان برگزار شد باید به ازای هر کلاس بیشتر یک واحد به conflicts اضافه شود. مثلا اگر در یک راه حل تعداد 5 کلاس در یک زمان بود باید دو واحد به conflicts اضافه کنید).

یا اگر یک استاد فرضا در یک زمان مشخص در 3 کلاس تدریس میکند باید به اندازه دو واحد به conflict اضافه شود(چون یک استاد در یک زمان میتواند حداکثر در یک کلاس باشد، پس اگر راه حل طوری بوده که سبب شده یک استاد در زمانی مشخص در سه کلاس باشد ، پس باید این اختلاف در conflicts نشان داده شود).

البته مدیریت قید ها بر عهده خودتان بوده و موارد گفته شده صرفا برای راهنمایی میباشد ، میتوانید 4 محدودیت گفته شده را به شیوه خودتان مدیریت کنید.

✓ در نهایت هدف ما کمینه کردن مقدار conflicts است.

وظایف شما در این سوال تکمیل فایل ACO_part2.ipynb می باشد:

1. مقداردهی اولیه به pheromone , heuristic (and heuristic

2. تکمیل تابع construct_solution در این تابع شما باید راه حلی که مورچه میتواند برای این مسئله بسازد را مشخص کنید.(Ants build a solution)

3. سپس باید تابع evaluate_conflicts را تکمیل نمایید، در اینجا باید به ازای رعایت نشدن هر کدام از قیود به conflicts مقداری اضافه کنید.(Evaluate constraints)

4. سپس باید برای راه حل ساخته شده تابع update_pheromone را تکمیل کنید.

5. و در نهایت حلقه اصلی اجرای aco را کامل نمایید.

امتیازی(10 نمره):

حرکت ذرات را در طول زمان(از زمانی که به صورت پراکنده در فضا هستند تا زمانی که به نقطه ای همگرا میشوند به صورت انیمیشن برای حداقل یکی از چهار سوال بالا نشان دهید).

نکات تمرین:

1. این تمرین به صورت انفرادی می‌باشد و در صورت تقلب واضح نمره کسر خواهد شد.
2. برای گرفتن نمره کامل ، نوشتن کد و اجرای صحیح آن کافی می‌باشد اما نوشتن توضیحات و تحلیل کدها می‌تواند به عنوان نمره امتیازی در نظر گرفته شود.
3. برای هر سوال فایل ژوپیتر در اختیار شما قرار گرفته است، بخش‌های خواسته شده را تکمیل کنید و در نهایت همه فایل‌ها را در قالب یک فایل زیپ ارسال کنید.
4. بخش‌هایی از فایل‌ها را که برای شما نوشته شده را می‌توانید تغییر دهید، اما در صورت ایجاد تغییر اساسی باید موارد تغییر داده شده ذکر شود.
5. در صورت داشتن هرگونه سوال و ابهامی می‌توانید از طریق آیدی تلگرام زیر با دستیار آموزشی مربوطه در ارتباط باشید.

آیدی تلگرام: @Behnaz_Aa