دانشگاه صنعتی اصفهان

دانشکده مهندسی برق و کامپیوتر

تکلیف تئوری سری چهارم

مبانی هوش محاسباتی

استاد درس : دکتر سمانه حسینی

دستیار آموزشی: بهناز عالی‌پور

زمان تحویل تکلیف:31 خرداد ساعت18

نکات تحویل تکلیف:

1.پاسخ های خود را حتما در سامانه یکتا آپلود کنید و از ارسال در بستر تلگرام خودداری کنید.

2.پاسخ ها ترجیحا تایپ شده و تمیز باشد که موجب کسر نمره به دلیل ناخوانایی نشود.

3.انجام تکلیف بصورت تک نفره است و در صورت مشاهده تقلب نمرات هم مبدأ و هم مقصد صفر میشود.

4.ساختار نامگذاری تکلیف باید بصورت روبرو باشد pdf.LastName_StudentID_HWX که X شماره تمرین،

StudentID شماره دانشجویی و LastName نام خانوادگی میباشد.

در صورت داشتن هرگونه ابهام یا سوالی می‌تواند از طریق آیدی تلگرام با دستیار آموزشی مربوطه در ارتباط باشید.

@Behnaz_Aa

Swarm intelligent

سوال اول:

چطور می‌توان یک الگوریتم هوش جمعی را برای کاربرد خاص به کار برد(به عنوان مثال الگوریتم pso برای بهینه سازی تخصیص منابع نوشته شده است در حالی که اکنون می‌خواهیم از این الگوریتم در مسئله بهینه سازی در حوزه مسائل مالی استفاده شود) توضیح دهید کدام بخش های الگوریتم ممکن است دچار تغیر شود و با مسئله جدید هماهنگ باشد؟

سوال دوم:

آیا می‌توان از الگوریتم هوش جمعی برای بهبود عملکرد مدل یادگیری ماشین استفاده کرد؟چگونه؟

پاسخ سوال اول:

برای هماهنگ سازی الگوریتم هوش جمعی که برای مسئله خاصی نوشته شده، برای استفاده در مسئله دیگر مهم ترین رکنی که باید در الگوریتم بررسی شود و در صورت نیاز تغیراتی در آن ایجاد گردد تابع هزینه می‌باشد، هر مسئله بهینه سازی طبیعتا مواردی در آن هست که باید بهینه شود که باید متناسب با آن تابع و محدودیت های لازم برای آن نوشته شود. به عنوان مثال در تخصیص منابع باید هزینه ها تا جای ممکن کاهش یافته و از طرفی بیشترین بهره برداری و استفاده از منابع را داشته باشیم ، این در حالی است که در مسائل مالی تابع هدف می‌تواند مربوط به کاهش ریسک و ضرر و بیشینه سازی سود باشد. پس نیاز است توابع مرتبط با این مسئله نوشته شود.

همچنین باید برای هر particle یا solution بلقوه نمایش خاصی در نظر گرفت. مثلا در تخصیص منایع یک particle می‌تواند به صورت مقدار یک منبع باشد در حالی که در مسائل مالی هر particle می‌تواند نشان دهنده درصد سود باشد. پس باید متناسب با هر مسئله نوع نمایش ذرات داخل الگوریتم هماهنگ شود.

از طرفی مکانیزم توقف نیز ممکن است در برخی موارد دچار تغیر شود، که این مورد را میتوان با تنظیم درست برخی پارامتر ها مثل : تعداد تکرار، و یا گذاشتن قید خاص مثل: تغیرات بسیار کم در جواب بعد از تعداد مشخصی تکرار کنترلر کرد.

پاسخ سوال دوم:

بله می‌توان، استفاده از الگوریتم های هوش جمعی برای تنظیم درست هایپرپارامترهای مدل های یادگیری ماشین و یا انتخاب feature های مناسب مفید باشد.

PSO

سوال اول:

برای هر یک از سوالات زیر توضیحات خواسته شده را کامل شرح دهید.

1. نقش هر یک از پارامترهای w,c1,c2 را توضیح دهید.
2. exploration و exploitation در PSO را توضیح دهید.
3. چه اتفاقی می‌افتد اگر c1=0,c2=0 باشد؟
4. PSO چگونه می‌تواند از local optimum ها فرار کند؟ آیا همیشه می‌تواند موفق باشد؟
5. رابطه بین اندازه جمعیت ذرات و دقت نهایی چیست؟

پاسخ سوال اول:

1. W: برای کنترل میزان وابستگی به سرعت قبلی استفاده میشود، در اصل ذرات نمی‌توانند ناگهان سرعت خودشان را تغیر دهند و به ذره خاصیت اینرسی اضافه می‌کند.
   C1: ضریبی است که تمایل به حرکت به سمت بهترین مکانی که خود ذره پیدا کرده است، را کنترل می‌کند.
   C2: این ضریب تمایل به حرکت به سمت بهترین مکانی که کل جمعیت پیدا کرده است را کنترل میکند

2. Exploration یعنی جستجوی نواحی ناشناخته در فضای جستجو برای جلوگیری از گیر افتادن در بهینه‌های محلی.
   Exploitation یعنی بهبود جواب با تمرکز بر روی نواحی که تاکنون کشف شده است، این ویژگی باعث می‌شود که به سمت جواب بهینه نهایی همگرا شویم.

3. این دو ضریب باعث میشوند تا ذرات به سمت بهترین نقاط پیدا شده حرکت کنند، در c1 ذره به سمت بهترین مکانی که خودش پیدا کرده است متمایل می‌شود و در c2 ذرات به سمت بهترین نقطه که جمعیت توانسته آن را پیدا کند. اما اگر هر دو اینها باهم صفر شوند در اصل ذرات نمی‌توانند بهترین موقعیت هایی را که پیدا کرده اند به خاطر بیاورند و احتمال همگرا شدن به جواب بهینه کاهش می‌یابد و صرفا ذرات ممکن است در یک مسیر مشخص قرار گیرند و یادگیری و تطبیق جدید برایشان صورت نخواهد گرفت.
4. تنوع بیشتر در جمعیت و داشتن رفتار تصادفی(r1,r2) باعث می‌شود که ذرات مسیر های متفاوتی را پیدا کنند و کم کم از مینیمم محلی فرار کنند. با این حال PSO تضمینی براینکه می‌تواند از مینیمم محلی به طور قطع فرار کند (مخصوصا در مسائل پیچیده با فضای جستجوی دشوار) نمی‌دهد.

5. هرچقدر اندازه جمعیت بزرگ تر باشد، باعث کشف مسیر های بیشتر خواهد شد و از طرفی دقت می‌تواند بیشتر شود، همچنین همانطور که بالاتر هم اشاره شده، جمعیت زیاد منجر می‌شود که از نقاط بهینه محلی خارج شویم و به سمت نقطه بهینه اصلی بیشتر همگرا شویم. اما نکته مهم این است که

افزایش بیش از حد جمعیت یک حجم محاسباتی سنگینی برای ما به همراه خواهد داشت که اگر بی رویه این کار صورت گیرد بهبود چندانی حاصل نخواهد شد.

سوال دوم:

فرض کنید موقعیت یک ذره در فضای دوبعدی به صورت x=[2, -1] است. سایر مقادیر هم به صورت زیر می‌باشند:

سرعت فعلی: $\vec{v} = [1,0.5]$

$$\overrightarrow{p_{best}} = [1,-2]$$

$$\overrightarrow{g_{best}} = [0,0]$$

w=0.6 , c1=2 , c2=2 , r1=0.5 , r2=0.5

الف) سرعت جدید برداری $\overrightarrow{v_{new}}$ را محاسبه کنید.

ب) موقعیت جدید $\overrightarrow{x_{new}}$ را بدست آورید.

پاسخ سوال دوم:

برای بروزرسانی سرعت ذرات می‌توان از فرمول زیر استفاده کرد:

$$v_i^{new} = w.v_i + c_1.r_1.\left(p_{best,i} - x_i\right) + c_2.r_2.\left(g_{best} - x_i\right)$$

پس اگر بخواهیم سرعت ذره را در بعد x بدست آوریم داریم:

$$v_1^{new,1} = 0.6 * 1 + 2 * 0.5 * (1 - 2) + 2 * 0.5 * (0 - 2) = -2.4$$

برای بعد *y* هم خواهیم داشت:

$$v_1^{new,2} = 0.6 * 0.5 + 2 * 0.5 * (-2 + 1) + 2 * 0.5 * (0 + 1) = 0.3$$

در نهایت سرعت ذره برابر است با:

$$v_1^{new} = [-2.4,0.3]$$

همچنین برای بدست آوردن مقدار X جدید نیز می‌توان از رابطه زیر استفاده کرد:

$$\overrightarrow{x_{new}} = \vec{x} + \overrightarrow{v_{new}} = [2,-1] + [-2.4,0.3] = [-0.4,-0.7]$$

سوال سوم:

تابع $f(x) = -x^2 = 5x + 20$ را در نظر بگیرید، دو مرحله از اجرای الگوریتم PSO برای یافتن نقطه بهینه این تابع را نشان دهید.

نکات:

تعداد particle ها را برابر با 5 در نظر بگیرید.

سرعت اولیه را برای همه particle ها برابر با صفر است.

پارامتر ها: r1=0.2, r2=0.8, c1=c2=w=1

پاسخ سوال سوم:

به عنوان مثال particle ها را به شکل زیر در نظر می‌گیریم:

$$x_1 = 0.6,\ x_2 = -6,\ x_3 = 2.3,\ x_4 = -1.1, x_5 = 10$$

در مرحله بعد باید تابع هزینه را برای هرکدام از *particle* ها بدست اوریم:

$$f_1 = 22.64,\ f_2 = -46,\ f_3 = 26.21, f_4 = 13.29, f_5 = -30$$

از انجایی که در گام اول هستیم Pbest را می‌توان برابر با مقادیرx در نظر گرفت.

$$p_{1,best} = 0.6,\ p_{2,best} = -6,\ p_{3,best} = 2.3,\ p_{4,best} = -1.1, p_{5,best} = 10$$

از طرفی Gbest برابر می‌شود با ذره ای که بیشترین مقدار تابع هزینه را دارد،که بیشترین مقدار تابع هزینه مربوط میشود به particle سوم .پس Gbest= X3

باقی پارامتر های مورد نیاز هم در صورت مسئله داده شده پس می‌توان سرعت های جدید را بدست آورد.

پس از فرمول بروزرسانی سرعت استفاده میکنیم:

$$v_i^{new} = w.v_i + c_1.r_1.\left(p_{best,i} - x_i\right) + c_2.r_2.(g_{best} - x_i)$$

$$v_1^{new} = 1 * 0 + 1 * 0.2 * (0.6 - 0.6) + 1 * 0.8 * (2.3 - 0.6) = 1.36$$

$$v_2^{new} = 1 * 0 + 1 * 0.2 * (-6 + 6) + 1 * 0.8 * (2.3 + 6) = 6.64$$

$$v_3^{new} = 1 * 0 + 1 * 0.2 * (2.3 - 2.3) + 1 * 0.8 * (2.3 - 2.3) = 0$$

$$v_4^{new} = 1 * 0 + 1 * 0.2 * (-1.1 + 1.1) + 1 * 0.8 * (2.3 + 1.1) = 2.72$$

$$v_5^{new} = 1 * 0 + 1 * 0.2 * (10 - 10) + 1 * 0.8 * (2.3 - 10) = -6.16$$

سپس باید مکان هر *particle* را بدست اوریم.

$$x^{new} = x + v^{new}$$

$$x_1^{new} = 0.6 + 1.36 = 1.96$$

$$x_2^{new} = -6 + 6.64 = 0.64$$

$$x_3^{new} = 2.3 + 0 = 2.3$$

$$x_4^{new} = -1.1 + 2.72 = 2.62$$

$$x_5^{new} = 10 - 6.16 = 3.84$$

و برای مرحله دوم هم به همین طریق عمل میکنیم ، برای مقادیر جدیدx تابع هزینه را محاسبه میکنیم و بر اساس آن تابع هزینه های بدست آمده را با تابع هزینه های قبلی برای هر ذره مقایسه میکنیم و در صورتی که تابع هزینه xجدید از x قبلی بیشتر باشد Pbest را بروز میکنیم.

برای Gbest هم متعلق به particle می باشد که بهترین مقدار تابع هزینه را داشته باش.

سپس مقادیر بدست آمده را در فرمول بروزرسانی سرعت میگذاریم و بعد از آن مقدار جدید x را محاسبه میکنیم.

این روند را تا جایی ادامه میدهیم تا به نقطه بهینه برسیم.

ACO

سوال اول :

الف) در چه شرایطی ممکن است الگوریتم ACO به یک جواب غیر بهینه همگرا شود؟

ب) نقش نرخ تبخیرρ وپارامتر های α و β در جلوگیری از این همگرایی را توضیح دهید.

پاسخ سوال اول:

الف) هنگامی که یک یا چند مورچه در تکرار های ابتدایی، تصادفی به مسیرهای بدی تمایل پیدا کنند و میزان انباشته شدن فرومون در این مسیرها زیاد شود، سبب می‌شود باقی مورچه‌ها نیز به این مسیر ها تمایل پیدا کنند و رفته رفته اکتشاف مسیر های بهتر متوقف خواهد شد.

ب)

نقش نرخ تبخیرρ :

اگر مقدار این متغیر خیلی کوچک باشد باعث می‌شود که فرومون خیلی آهسته پاک شود و همین امر موجب تقویت مسیر های اولیه خواهد شد به همین منظور بزرگ بودن این پارامتر می‌تواند باعث افزایش همگرایی به مسیر های بد شود(اگر مسیر های ابتدایی بهینه نباشند).

اما از طرفی اگر این پارامتر هم خیلی کوچک باشد، باعث بی ثباتی الگوریتم خواهد شد چونکه فرومون خیلی سریع پاک می‌شود و مثل این هست که مورچه‌ها دائما به صورت تصادفی در حرکت هستند.

نقش پارامتر α:

اگر زیاد باشد مورچه ها صرفا به فرومون وابسته می‌شوند و مسیر های دیگر را کمتر کاوش می‌کنند، اگر در ابتدا مسیر بدی انتخاب شود مورچه ها به احتمال زیاد به همان مسیر اکتفا می‌کنند.پس زیاد بودن این پارامتر می‌تواند باعث افزایش همگرایی به جواب غیر بهینه شود.

از طرفی اگر مقدار این پارامتر هم خیلی کم باشد سبب می‌شود که مورچه ها رفتار تصادفی تری داشته باشند.

نقش پارامتر β:

زیاد بودن این پارامتر سبب می‌شود که انتخاب مسیر بر اساس کیفیت محلی باشد(نه تجربه قبلی) اما کم بودن آن منجر می‌شود به اینکه مورچه ها به فرومون بیشتر اتکا کنند.پس افزایش این پارامترمی‌تواند در مواردی از همگرایی به مسیر های بد جلوگیری کند.

سوال دوم:

پیچیدگی زمانی ACO را برای حل یک مسئله TSP با n گره، m مورچه و t تکرار تحلیل کنید.

پاسخ سوال دوم:

مورچه از یک شهر شروع به حرکت می‌کند، در هر مرحله باید حداکثر برای n شهر احتمال را حساب کند و تصمیم بگیرد که به کدام شهر برود.

به عبارتی اگر گراف کامل باشد، در مرحله اول برای n-1 شهر باید این محاسبه احتمال صورت گیرد، در مرحله دوم برای n-2 شهر و در مرحله سوم برای n-3 و به همین صورت تا آخر. پس اگر n مرحله داشته باشیم، داریم:

$$O(n^2) = O(n) + O(n-1) + O(n-2) + \cdots + O(1)$$

پس چون m مورچه داریم پیچیدگی ساخت مسیر در نهایت برابر است با: m*n^2

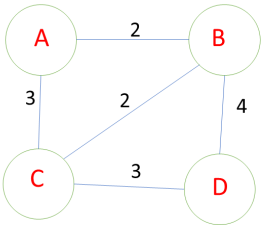
بعد از ساخت مسیر باید فرومون ها روی هر یال بروزرسانی شود، که مسیر متشکل از n یال است پس پیچیدگی برابر با n خواهد بود . اما نکته ای که وجود دارد این است که اگر به این شکل به مسئله نگاه کنیم که تمام یال ها (n^2 تا) باید بررسی شود یا بروزرسانی شود. پس در این حالت پیچیدگی بروزرسانی فرومون یال ها برابر با O(n^2)خواهد شد.

در نهایت داریم: $O(n^2) + O(m * n^2) = O(m * n^2)$ که چون t بار این مراحل تکرار خواهد شد پس داریم:

$$O(t*m*n^2)$$

سوال سوم:

فرض کنید گرافی را داریم که از 4 گره A,B,C,D تشکیل شده است، هدف این است که مسیر کوتاه را با استفاده از الگوریتم ACO از گره A به گره D پیدا کنیم.



- مقدار فرومون اولیه روی همه یال ها: $\tau_{ij} = 1$
- تابع heuristic تعریف شده به صورت زیر است :

$$\eta_{ij} = \frac{1}{d_{ij}}$$

- $\alpha=1$ (تاثیر فرومون)
- $\beta=2$ (تاثیر هیوریستیک)
- نرخ تبخیر: $\rho=0.2$
- پارامتر Q برای بروزرسانی فرومون : $Q=10$

الف) احتمال اینکه مورچه مسیر A->B یا A->C را انتخاب کند را محاسبه کنید.

ب) فرض کنید مورچه اول مسیر A->B->C->D را طی می‌کند و مورچه دوم مسیر

A-> C-> D را طی می‌کند.

1. مقدار افزایش فرومون را روی هر یال مسیر محاسبه کنید.
2. مقدار جدید فرومون را برای هر یک از یال ها پس از طی شدن این دو مسیر بدست آورید.

پاسخ سوال سوم:

الف:

برای محاسبه احتمال انتخاب مسیر از گره A می‌توان از فرمول زیر استفاده کرد:

$$p_{ij} = \begin{cases} \dfrac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{l \in T}[\tau_{lk}]^{\alpha}[\eta_{lk}]^{\beta}} & \text{if } j \in T \\ 0 & \text{otherwise} \end{cases}$$

برای استفاده از این فرمول لازم است تا پیش از آن تابع heuristic را برای هر دو یال گفته شده بدست آوریم.

$$\eta_{AB} = \frac{1}{2} = 0.5 \ , \qquad \eta_{AC} = \frac{1}{3} \approx 0.333$$

سپس مقادیر بدست آمده را در فرمول بالا جایگزین می‌کنیم:

برای A -> B داریم:

$$\mathrm{P}_{AB} = \frac{1^1 * (0.5)^2}{1^1 * (0.5)^2 + 1^1 * (0.333)^2} = \frac{0.25}{0.25 + 0.111} \approx \frac{0.25}{0.361} \approx 0.692$$

به همین ترتیب برای A -> C داریم:

$$\mathrm{P}_{AC} = \frac{1^1 * (0.333)^2}{1^1 * (0.5)^2 + 1^1 * (0.333)^2} = \frac{0.111}{0.25 + 0.111} \approx \frac{0.111}{0.361} \approx 0.3074$$

با توجه به احتمالات به دست آمده، احتمال انتخاب A->B بیشتر از یال دیگر خواهد بود.

ب -1)

طول مسیربرای مورچه اول برابر است با : d(A,B) + d(B,C) + d(C,D) = 2+2+3=7

افزایش فرومون : $\Delta\tau_{ij} = \frac{Q}{L} = \frac{10}{7} \approx 1.43$

طول مسیر برای مورچه دوم برابر است با: d(A,C)+d(C,D) = 3+3 =6

افزایش فرومون : $\Delta\tau_{ij} = \frac{Q}{L} = \frac{10}{6} \approx 1.67$

ب-2)

فرمول بروزرسانی فرومون به این شکل است:

$$\tau_{ij} = (1-\rho)\tau_{ij} + \Delta\tau_{ij}$$

از آنجایی که یال C->D در مسیر هر دو مورچه بوده است پس مقدارنهایی فرومون برای این یال برابر است با جمع $\Delta\tau$ های هر دو مورچه درنتیجه مقدار نهایی فرومون این یال می‌شود:

$$(1-0.2)*1 + 1.43+1.67=3.9$$

بعد از آن مقدار فرومون برای باقی یال ها به شکل زیر است:

مقدار نهایی فرومون برای مسیرهای طی شده توسط مورچه اول(یعنی یال های A–B، B–C) برابر است با:

$$(1-0.2)*1+1.43=2.23$$

همچنین مقدار نهایی فرومون برای مسیر های طی شده توسط مورچه دوم (یعنی یال های A->C) برابر است با:

$$(1-0.2)*1+1.67=2.47$$

و مقدار فرومون برای یال B-D که هیچ کدام از مورچه ها از روی آن رد نشده اند برابر با صفر است.

سوال چهارم:

مسئله TSP زیر را در نظر بگیرید.

$$d(A,B) = 2,\ d(A,C) = 3,\ d(A,D) = 5,\ d(B,C) = 3,\ d(B,D) = 3,\ d(C,D) = 4$$

الف) تعداد تمام مسیر های ممکن در این مسئله را بدست آورید.

ب) کدام مسیر بیشتر احتمال دارد تا توسط مورچه ها یافت شود؟

ج) تفاوت استفاده از یک مورچه برای حل این مسئله با حالتی که از چندین مورچه استفاده شود را به طور کامل شرح دهید.

پاسخ سوال چهارم:

الف) تمام جایگشت های ممکن برای 4 شهر برابر است با : 4!=24 (4 انتخاب برای شهر اول × 3انتخاب برای شهر دوم × 2انتخاب برای شهر سوم × 1 انتخاب برای شهر چهارم).

در مسئله TSPچون مسیر ها مدور هستند(به عنوان مثال: مسیر ABCDA با مسیر BADCB با هم یکی هستند اما نقطه شروع متفاوت دارند)، باید تمام جایگشت های ممکن را تقسیم بر تعداد شهر کنیم، پس در نهایت داریم:

$$\frac{4!}{4} = 6$$

البته اگر مسیر هایی که برعکس هم هستند را در نظر بگیریم(مثل ABCDA=ADCBA) آنگاه 3 مسیر متفاوت داریم.

اما نکته مهم این است که چون مورچه ها به صورت local heuristics عمل می‌کند(یعنی اطلاعات فوری و ساده‌ای که مورچه هنگام تصمیم‌گیری در یک مرحله خاص از حرکتش استفاده می‌کند تا انتخاب کند کجا برود)، اغلب همان 6 مسیر انتخاب می‌شود چون ترتیب انتخاب و گاها جهت حرکت مورچه ها اهمیت دارد.

ب) در ابتدا که هیچ فرومونی وجود ندارد،مورچه ها به صورت heuristic عمل می‌کنند، اما بعد از مدتی، معمولا مورچه ها مسیری را انتخاب می‌کنند که فاصله کمتری دارد(مثلا مورچه ها تمایل دارند تا مسیر A->B را که طول کمتری دارد انتخاب کنند) بعد از این مرحله تمایل دارند مسیر B->C یا B->D را که طول سه دارند را انتخاب کنند(احتمال انتخاب این دو مسیر به دلیل اینکه طول برابری دارند یکسان است.). اما مورچه‌ها به هر حال بعد از مدتی احتمال انتخاب مسیری مثل ABDCA که بین مسیر های دیگر طول کمتری دارد بیشتر خواهد شد.

نکته: احتمال انتخاب شدن مسیری که A->D در آن ظاهر شده است بسیار کم می‌باشد، چون طول بیشتری نسبت به دیگر فواصل دارد.

ج) در حالتی که برای حل مسئله صرفا از یک مورچه استفاده کنیم، احتمال دارد که مورچه در ابتدا نقطه شروع بدی را انتخاب کند(مثلا A->D را ابتدای حرکت خود انتخاب می‌کند) در این حالت این مسیر فرومون بیشتری خواهد گرفت و به همین جهت مورچه به اشتباه مسیر طولانی تری را برای رسیدن به مقصد نهایی انتخاب خواهد کرد.

اما اگر از چندین مورچه استفاده کنیم، مورچه های دیگر می‌توانند از مسیر های دیگری بروند و فرومون در دیگر مسیر ها نیز تقویت خواهد شد و به همین منظور احتمال پیدا کردن مسیر بهینه افزایش پیدا خواهد کرد.