

ارایه الگویی برای خوشبندی داده‌ها مبتنی بر الگوریتم جستجوی هارمونی

محمد رضا شهریاری*

دانشکده مدیریت و حسابداری دانشگاه آزاد اسلامی واحد تهران جنوب، تهران، ایران

رسید مقاله: ۲۰ فروردین ۱۳۹۵
پذیرش مقاله: ۶ شهریور ۱۳۹۵

چکیده

خوشبندی، فرآیند طبقه‌بندی داده‌ها داخل گروه‌های یا خوش‌های خاص، بر اساس درجه شباهت بین آن‌ها است که یکی از روش‌های پر کاربرد در بسیاری از زمینه‌های علمی است. در ادبیات خوشبندی، در سال‌های اخیر محققان به منظور اجتناب از گرفتار شدن در بهینه محلی، الگوریتم‌های فرابنتکاری که از پدیده‌های اجتماعی و طبیعی الهام گرفته‌اند را برای حل مسایل خوشبندی ارایه کرده‌اند. در این مقاله الگوریتم جستجوی هارمونی (HSA) که مبتنی بر عملکرد سازه‌ای موسیقی است به عنوان یک الگوریتم توانا در حل مسایل خوشبندی در نظر گرفته می‌شود. به منظور ارزیابی توانایی الگوریتم چندین مجموعه داده استاندارد و واقعی ارایه می‌شود. نتایج شبیه‌سازی نشان می‌دهد که الگوریتم HSA از کارایی بالایی در بهدست آوردن جواب‌های مطلوب برخوردار است.

کلمات کلیدی: خوشبندی، الگوریتم فرابنتکاری، الگوریتم k-means، جستجوی هارمونی.

۱ مقدمه

خوشبندی داده‌ها که آنالیز خوشبندی نیز نامیده می‌شود، یکی از روش‌های مهم طبقه‌بندی بدون نظارت می‌باشد. در خوشبندی تلاش می‌شود تا مشاهدات واقع در هر خوشبندی بیشترین تشابه را از نظر متغیرهای مورد نظر باهم داشته باشند و مشاهدات هر گروه از مشاهدات دیگر خوشبندی بیشترین فاصله را داشته باشند. در سال‌های اخیر از این نگرش به عنوان یک فاکتور تأثیرگذار در علوم مختلف از جمله پردازش تصاویر، پژوهشی، انتخاب تأمین کننده و بازاریابی استفاده شده است.

به طور کلی می‌توان روش‌های خوشبندی را به بخش‌های سلسله مراتبی و تفکیکی تقسیم بندی کرد که روش سلسله مراتبی نیز به دو دسته ادغامی و شکافی تقسیم می‌شوند [۱]. در روش‌های تفکیکی که به آن مرکزگرانیز گفته می‌شود، تابع خطایی تعریف می‌شود که به دنبال حداقل کردن آن هستیم. یکی از پرکاربردترین روش‌های مرکزگرانیز K-means می‌باشد [۲]. در این روش تعداد خوشبندی‌ها به عنوان یک پارامتر ورودی برای الگوریتم، ثابت و از پیش تعیین شده است. از مهم‌ترین ویژگی‌های K-means می‌توان به ساده و سریع بودن آن اشاره کرد. اگرچه K-means سریع و ساده است ولی به موقعیت اولیه مراکزش بسیار وابسته است و به همین دلیل اغلب به بهینه محلی همگرا می‌شود. در سال‌های اخیر برای غلبه بر مشکل بهینه

* عهده‌دار مکاتبات

آدرس الکترونیکی: shahriari.mr@gmail.com

محلى، محققان تلاش کردند که با استفاده از الگوریتم‌های فرا ابتکاری که بسیاری از آن‌ها از پدیده‌های اجتماعی و طبیعی الهام گرفته‌اند، تابع هدف K-means را بهینه کنند، که از جمله می‌توان یه روش‌های SA، PSO، GA اشاره کرد [۴۳].

اگرچه الگوریتم‌های ذکر شده سعی بر بهبود تابع هدف K-means را دارند. ولی این نکته را باید پذیرفت که برخی از آن‌ها نیز از کاستی‌هایی همچون کیفیت نتایج و سرعت همگرایی پایین مانند GA و برخی دارای ساختار پیچیده و نرخ همگرایی پایین مانند PSO رنج می‌برند.

یکی از روش‌های فرا ابتکاری که در سال‌های اخیر مورد توجه قرار گرفته است، الگوریتم جستجوی هارمونی (HSA) می‌باشد که یک الگوریتم تکاملی مبنی بر عملکرد سازه‌ای موسیقی است که از آلات موسیقی الهام گرفته و از بداهه‌سرایی سرایندگان تقلید می‌نماید. در این مقاله ما از این الگوریتم به عنوان یک الگوریتم قدرتمند در حل مسایل خوشبندی استفاده می‌کنیم. ما نشان خواهیم داد که الگوریتم HSA قوی و مناسب برای خوشبندی داده‌ها است و همچنین نتایج به دست آمده توسط این الگوریتم دارای کیفیت بالایی است که برای این کار از چندین مجموعه داده واقعی و مشهور برای ارزیابی استفاده می‌شود.

باقي مانده این مقاله به شرح زیر سازماندهی می‌شود: در قسمت ۲، خوشبندی داده بحث می‌شود. در قسمت ۳، الگوریتم HSA معرفی می‌شود و سپس نتایج پیاده سازی در قسمت ۴ بحث می‌شود. سرانجام، ما در قسمت ۵ نتیجه گیری از مقاله می‌کنیم.

۲ خوشبندی داده‌ها

خوشبندی فرآیند خودکاری است که در طی آن، اشیا به دسته‌هایی که اعضای آن از نظر شاخص‌های مورد نظر مشابه یکدیگر می‌باشند تقسیم می‌شوند. بنابراین برای سنجش شباهت بین اشیا داده از اندازه گیری فاصله استفاده می‌شود. روش‌های مختلفی برای اندازه گیری فاصله بین دو شی وجود دارد که فاصله اقلیدسی معروف‌ترین و پرکاربردترین گونه فاصله است که به صورت رابطه (۱) تعریف می‌شود.

$$d(X, Y) = \sqrt{\sum_{i=1}^m (x_i - y_i)^2} \quad (1)$$

فرض کنید در فضای اقلیدسی N بعدی مجموعه ای از n شی به وسیله مجموعه $S = \{x_1, x_2, \dots, x_n\}$ و خوشبندی مجموعه $C = \{c_1, c_2, \dots, c_K\}$ نشان داده شود، از این رو خوشبندیها باید شرایط زیر را داشته باشند:

- هر خوشبندی حداقل باید شامل یکشی باشد، یعنی:
 $C_i \neq \emptyset \quad \forall i \in \{1, 2, \dots, K\}$
- دو خوشبندی مختلف باید اشیا مشترک داشته باشند، یعنی:
 $C_i \cap C_j = \emptyset \quad \forall i \neq j \text{ and } i, j \in \{1, 2, \dots, K\}$
- هر شی باید به یک خوشبندی تخصیص پیدا کند، یعنی:

$$\bigcup_{i=1}^k C_i = S$$

برای پیدا کردن مرکز K خوش، مساله به عنوان یک بهینه‌سازی (حداقل) یک تابع عملکرد تعريف می‌شود. تابع عملکرد مورد استفاده برای این هدف، کل میانگین مربع خطای تدریج (MSE) است که به صورت رابطه (۲) تعريف می‌شود [۵].

$$f(X, C) = \sum_{i=1}^N \text{Min}\{X_i - C_j | j = 1, 2, \dots, K\}. \quad (2)$$

هدف این معیار کمینه کردن مجموع فواصل داده‌های موجود در درون خوش‌ها از مرکز خوش‌ها می‌باشد.

عبارت $\|X_i - C_j\|$ فاصله بین داده X_i و مرکز مربوطه C_j را نشان می‌دهد [۱۳].

۳ الگوریتم جستجوی هارمونی (HSA)

الگوریتم جستجوی هارمونی، یک الگوریتم تکاملی مبتنی بر عملکرد سازهای موسیقی است که اولین بار توسط [۱۳] معرفی شد. این الگوریتم با مفهومی ساده و تعداد پارامترهای کم و آسان بر مبنای علم آمار و ریاضیات است. این الگوریتم از آلات موسیقی الهام گرفته و از بداهه‌سرایی سرایندگان تقليد می‌نماید. تابع هدف HSA تخمين و اندازه‌گیری هارمونی و اثری است که آن موسیقی در فرد می‌گذارد تا حالت مناسبی از هارمونی در شخص ایجاد شود. باید به این نکته اشاره شود که HSA فرآیند بداهه‌سرایی کیفی را به فرآیند بهینه‌سازی کمی تبدیل می‌کند هنگامی که یک سراینده یک آلت موسیقی را می‌نوازد با سه حالت روبرو است:

۱. نواختن بر اساس حافظه اش
۲. نواختن بر اساس تنظیم کوک
۳. نواختن تصادفی

این سه حالت در واقع سه عملگر کیفی مد نظر در HSA می‌باشد که به ترتیب عملگر حافظه هارمونی (HM)، تنظیم کوک، و تصادفی نواختن است. در واقع فرآیند بداهه‌سرایی HSA ترکیب این سه عملگر می‌باشد.

۱-۳ بردار هارمونی و فرآیند رمزگشایی

طریقه تعريف ساختار بردار هارمونی از جمله تاثیرگذارترین بخش‌ها برای افزایش کارایی الگوریتم‌های بهینه‌سازی ترکیباتی می‌باشد. که در این مقاله به صورت رشته‌ای تعريف شده است.

۲-۳ فرآیند بداهه‌سرایی

به منظور اجرای فرآیند بداهه‌سرایی در تکرارهای مختلف الگوریتم، یک عدد تصادفی تولید شده و سپس یک یا دو عملگر HSA جهت اجرای فرآیند بداهه‌سرایی مورد استفاده قرار می‌گیرند.

۳-۳ نرخ مورد نظر HM

استفاده از این عملگر در الگوریتم جستجوی هارمونی مشابه بحث نخبه گرایی در الگوریتم ژنتیک است. در واقع این عملگر تضمین می‌کند که بهترین هارمونی‌ها در طی فرآیند بهینه‌سازی از حافظه پاک نخواهد شد. این عملگر با نرخی به نام نرخ مورد نظر حافظه هارمونی (HMCR) کنترل می‌شود. مقادیر کوچک این نرخ باعث کندی همگرایی الگوریتم خواهد شد. علت نیز وجود تعداد کمی هارمونی برگزیده در فرآیند بداهه‌سرایی است. از طرفی دیگر، مقادیر بالای این نرخ باعث این می‌شود که تنها هارمونی‌های HM انتخاب شده و مورد استفاده قرار گیرند لذا الگوریتم در بهینه‌های محلی مستقر خواهد شد. بدین منظور [۶] پیشنهاد دادند که مقدار این نرخ در بازه $[0.75, 0.95]$ تنظیم شود.

$$P_{HMCR} = HMCR$$

۳-۴ عملگر تنظیم کوک

در دنیای موسیقی، تنظیم کوک به معنای تغییر ساختار فرکانس‌ها می‌باشد که معادل تولید جواب‌های همسایه در فرآیند بهینه‌سازی است. در واقع فضایی از جواب که توسط عملگرهای دیگر یافت نمی‌شود، کشف و مورد جستجو قرار می‌گیرد. این عملگر از نرخی به نام نرخ تنظیم کوک (r_{pa}) بهره جسته تا تنظیمات را کنترل نماید. کارکرد این عملگر مشابه عملگر جهش در الگوریتم ژنتیک است. بنابراین مقادیر بالای عملگر تنظیم کوک تنوع گرایی را در الگوریتم افزایش خواهد داد. افزایش این نرخ باعث این می‌شود که الگوریتم مانند یک روش جستجوی تصادفی عمل نماید. بدین منظور $[0.05, 0.1]$ در بازه $[0.05, 0.1]$ تنظیم شود. احتمال بهره جستن از این عملگر در رابطه (6) نمایش داده شده است. به منظور پیاده‌سازی این عملگر ابتدا یک یا چند جزء از بردار هارمونی به صورت تصادفی انتخاب می‌شود، باید در نظر داشت که می‌توان از عملگرهای مختلف در این زمینه می‌توان بهره جست.

$$P_{pa} = HMCR \times r_{pa} \quad (3)$$

۳-۵ عملگر تصادفی سرایی

مشابه عملگر تنظیم کوک، این نوع عملگر هم جهت افزایش تنوع گرایی مورد استفاده قرار می‌گیرد. این عملگر جواب‌های با تنوع گرایی بالاتری را کشف کرده و به احتمال قوی از بهینه محلی عبور می‌کند [6,9]. احتمال تصادفی سرایی نیز از رابطه زیر قابل محاسبه است:

$$P_{rand} = 1 - HMCR \quad (4)$$

ساختار کلی الگوریتم هارمونی به شکل زیر است:

Parameter Setting (number of iteration ,Pop Size ,HMCR , P_{pa} , β)

Best solution = []

for I = 1 to number of Pop Size do

 HM(I)=Randomly

 fitness HM (I)=evaluate(HM (I))

End

```

for it = 1 to number of iteration do
  for I = 1 to number of Pop Size do
    for j = 1 to nvar do
      if rand < HMCR
        HM=choose a solution randomly
      If rand <  $P_{pa}$ 
        HM(j)= HM(j)+uniform(-1,1).  $\beta$ 
      Else
        HM(j)= HM(j)
      end if
      Else
        HM(J)= Randomly
      end if
    End
  End
End

```

شبه کد الگوریتم HSA

۴ نتایج پیاده‌سازی

در این قسمت، عملکرد الگوریتم HSA با چندین الگوریتم مشهور که در ادبیات اخیر گزارش شده است مقایسه می‌شود. مقایسه نتایج برای هر مجموعه داده بر اساس بهترین، متوسط و بدترین جواب پیدا شده در بیش از ۲۰ شبیه‌سازی متفاوت برای هر الگوریتم می‌باشد. عملکرد الگوریتم‌ها بر اساس معیار مجموعه فواصل درون خوش ای، همان طور که در رابطه (۲) تعریف شده است، مقایسه می‌شوند. پارامترهای اصلی در الگوریتم HSA شامل تعداد نت‌ها (HM)، تعداد تکرار الگوریتم (Maximp)، نرخ حافظه‌ی هارمونی (HMCR) و پارامتر نرخ تنظیم کوک (Pac) است که مقدار هر یک از این پارامترهای در جدول (۱) نشان داده شده‌اند. نرم افزار MATLAB برای کد کردن الگوریتم پیشنهادی استفاده می‌شود.

جدول ۱. مقادیر پارامترهای الگوریتم HSA

پارامتر	مقدار
HM	۷
Maximp	۳۵۰۰
HMCR	۰/۹۱
Pac	۱/۵

برای ارزیابی دقیق‌تر روش پیشنهادی، ما از ۴ مجموعه داده واقعی استفاده کرده‌ایم. این مجموعه داده‌ها از ماشین آزمایشی آزمایشگاه گرفته شده است که تعداد زیادی از محققان برای ارزیابی الگوریتم‌شان به کار گرفته شده است [۷ و ۸]. این مجموعه داده‌ها به نام‌های Iris، Vowel، Wine و CMC هستند.

جدول ۲. مجموع فواصل درون خوشه‌ای الگوریتم‌ها

	مجموعه داده	معیار	K-means	SA	GA	PSO	HSA
Iris	میانگین	بهترین	۹۷/۳۳۳	۹۷/۴۵۷۳	۱۱۳/۹۸۶۵	۹۶/۸۹۴۲	۹۶/۶۵۰۴
		بدترین	۱۰۶/۰۵	۹۹/۹۵۴	۱۲۵/۱۹۷۰	۹۷/۲۳۲۸	۹۶/۷۳۱۲
		میانگین	۱۲۰/۴۵	۱۰۲/۰۱	۱۳۹/۷۷۸۲	۹۷/۸۹۷۳	۹۶/۹۵۳۰
Wine	میانگین	بهترین	۱۶۵۵۵/۶۸	۱۶۴۷۳/۴۸۲۵	۱۶۵۳۰/۵۳۳۸	۱۶۳۴۵/۹۶۷۰	۱۶۲۹۸/۷۷۹۴
		بدترین	۱۸۰۶۱	۱۷۵۲۱/۰۹۴	۱۶۵۳۰/۵۳۳۸	۱۶۴۱۷/۴۷۲۵	۱۶۲۹۹/۰۰۰۷
		میانگین	۱۸۵۶۳/۱۲	۱۸۰۸۳/۲۵۱	۱۶۵۳۰/۵۳۳۸	۱۶۵۶۲/۳۱۸۰	۱۶۲۹۹/۵۳۱۳
Vowel	میانگین	بهترین	۱۴۹۲۲/۲۶	۱۴۹۳۷۰/۴۷۰۰	۱۴۹۵۱۲/۷۳۵	۱۴۸۹۷۶/۰۱۵۲	۱۴۹۳۹۳/۰۸
		بدترین	۱۵۹۲۴۲/۸۹	۱۶۱۵۶۶/۲۸۱۰	۱۵۹۱۵۳/۴۹۸	۱۵۱۹۹۹/۸۲۵۱	۱۴۹۳۹۳/۵۸
		میانگین	۱۶۱۲۳۶/۸۱	۱۶۵۹۸۶/۴۲۰۰	۱۶۰۹۹۱/۶۵۲۰	۱۵۸۱۲۱/۱۸۳۴	۱۴۹۳۹۴/۱۷
CMC	میانگین	بهترین	۵۸۴۲/۲۰	۵۸۴۹/۰۳۸۰	۵۷۰۵/۶۳۰۱	۵۷۰۰/۹۸۵۳	۵۶۹۶/۳۰۱۵
		بدترین	۵۸۹۳/۶۰	۵۸۹۳/۴۸۲۳	۵۷۶۵/۵۹۸۴	۵۸۲۰/۹۶۴۷	۵۶۹۶/۹۴۶۰
		میانگین	۵۹۳۴/۴۳	۵۹۶۶/۹۴۷۰	۵۸۱۲/۶۴۸۰	۵۹۲۳/۲۴۹۰	۵۶۹۷/۸۸۱۲

نتایج شبیه‌سازی به دست آمده برای مقایسه روش پیشنهادی با روش‌های مختلف بر روی مجموعه داده‌های Iris، Wine و CMC در جدول (۲) نشان داده شده است. از مقادیر به دست آمده در جدول (۲) می‌توانیم بینیم که نتایج به دست آمده توسط الگوریتم پیشنهادی برای مجموعه Iris تقریباً برابر و نزدیک به ۹۶.۶۵ است، که به طور قابل توجهی از سایر الگوریتم‌ها بهتر می‌باشد. بر روی مجموعه داده Wine، الگوریتم HSA به بینه کلی ۱۶۲۹۸.۷۷۹۴ همگرا شده است. بر روی مجموعه داده Vowel الگوریتم پیشنهادی به بینه کلی ۱۴۹۳۹۳.۰۸ رسیده است. این در حالی است که سایر الگوریتم‌ها حتی با بیش از ۲۰ اجرا قادر به رسیدن به این مقدار نیستند. در مجموعه داده CMC الگوریتم HSA بهترین نتیجه را به دست آورده است که این مقدار برابر است با ۵۶۹۶.۳۰ است.

۵ نتیجه‌گیری

در سالهای اخیر محققان، الگوریتم‌های متعددی را برای یافتن جواب بهینه مساله در حل مسایل خوشه‌بندی به کار گرفته‌اند. این توسعه گاهی با ترکیب الگوریتم‌ها با یکدیگر و گاهی با ابداع یک روش جدید میسر شده است. این مطالعه از الگوریتم جستجوی هارمونی به منظور حل مسایل خوشه‌بندی داده‌ها استفاده می‌کند. نتایج شبیه‌سازی به دست آمده نشان می‌دهد که الگوریتم جستجوی هارمونی می‌تواند به عنوان یک روش مؤثر برای خوشه‌بندی داده‌ها در نظر گرفته شود. به طور خلاصه، با توجه به نتایج به دست آمده، می‌توان نتیجه گرفت که الگوریتم پیشنهادی HSA دقیق و قابل اعتماد بوده و می‌تواند جواب‌های با کیفیت را با انحراف استاندارد پایین به دست بیاورد. در حالی که سایر الگوریتم‌ها ممکن است به بینه محلی گرفتار شوند. از طرف دیگر می‌توان به عنوان پژوهش آتی ترکیب این الگوریتم را با K-means آزمود که باعث می‌شود الگوریتم K-

means به بهینه محلی گرفتار نشد و همچنین باعث افزایش کیفیت جواب به دست آمده نسبت به الگوریتم K-means می شود.

منابع

- [1] Hatamlou, A., Abdullah, A.,and Nezamabadi-pour, S. "A combined approach for clustering based on K-means and gravitational search algorithms" Swarm and Evolutionary Computation, (6), pp. 47–52, 2012.
- [2] Anil, K.J. "Data clustering: 50 years beyond K-means" Pattern Recognition Letters, (31), pp. 651–666, 2010.
- [3] Maulik U., and Bandyopadhyay, S. "Genetic algorithm-based clustering technique" Pattern Recognition, (33), pp. 1455–1465, 2000.
- [4] Paterlini, S., and Krink, T. "Differential evolution and particle swarm optimization in partitional clustering" Computational Statistics and Data Analysis, (50), pp.1220–1247, 2006.
- [5] Niknam, T., and Amiri, B., "An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis" Applied Soft Computing, (10), pp.183–197, 2010.
- [6] Geem Z.W., Kim J.H., and Loganathan G.V. "A new heuristic optimization algorithm, harmony search" Simulation, (76), pp. 60-68, 2001.
- [7] Merz, C.J., and Blake, C.L. UCI Repository of Machine Learning Databases. Available from: <http://www.ics.uci.edu/~mlearn/MLRepository.html>.
- [8] N. Sharma, A. Bajpai and R. Litoruya, "Comparison the various clustering algorithms of weka tools", International Jornal of Emerging technology and Advanced Engineering, vol. 2, no. 5, (2012) May.
- [9] M. E. Celebi, H. A. Kingravi, and P. A. Vela, "A comparative study of efficient initialization methods for the k-means clustering algorithm," Expert Systems with Applications, vol. 40, no. 1, pp. 200–210, 2013.
- [10] E. J. Dries and G. L. Peterson, "Scaling ant colony optimization with hierarchical reinforcement learning partitioning," in Proceedings of the 10th Annual Genetic and Evolutionary Computation Conference (GECCO '08), pp. 25–32, BiblioBazaar, Atlanta, Ga, USA, July 2008
- [11] P. Shen and C. Li, "Distributed information theoretic clustering," SIGNAL PROCESSING, IEEE Transactions on, vol. 62, no. 13, pp. 3442–3453, 2014.
- [12] G. Di Fatta, F. Blasa, S. Cafiero, and G. Fortino, "Fault tolerant decentralised k-means clustering for asynchronous large-scale networks," Journal of Parallel and Distributed Computing, vol. 73, no. 3, pp. 317– 329, 2013.
- [13] H. Mashayekhi, J. Habibi, S. Voulgaris, and M. van Steen, "Goscan: Decentralized scalable data clustering," Computing, vol. 95, no. 9, pp. 759–784, 2013.