Karafan
Quarterly Scientific Journal of National University of Skills
Spring 2025, Vol. 22, No. 1, p.60-80
Journal Homepage: https://karafan.nus.ac.ir/article_229119.html
10.48301/kssa.2024.446587.2856

A Hybrid Algorithm for Cluster Analysis Based on Flower Pollination and Big Bang-Big Crunch Algorithms

Farzaneh Mahalleh [1] Abdolreza Hatamlou *[2]

[1]PhD Candidate, Department of Computer Engineering, Technical and Vocational University (TVU), Tehran, Iran.

[2]Associate Professor, Department of Computer Engineering, Khoy Branch, Islamic Azad University, Khoy, Iran.

ARTICLE INFO

Received: 25.03.2024
Revised: 20.09.2024
Accepted: 27.11.2024

Keyword:
Hybrid Algorithm,
Data Clustering,
Flower Pollination Algorithm,
Big Bang-Big Crunch Algorithm

***Corresponding Author:**
Abdolreza Hatamlou

Email:
rezahatamloo@gmail.com

ABSTRACT

The purpose of clustering is to identify natural categories in a large data set, which, by summarizing and simplifying, provides the possibility of analyzing a huge amount of data for other applications. To date, numerous algorithms have been proposed to address the clustering problem; however, no single algorithm consistently performs well across diverse conditions and data types. Each algorithm has its advantages and disadvantages. Therefore, the subject of current research is the design of hybrid algorithms to exploit the advantages of two or more algorithms in a single algorithm. The features of different algorithms are complementary. To achieve this goal, a hybrid meta-heuristic algorithm based on Flower Pollination and Big Bang-Big Crunch algorithms is presented in this thesis. In the proposed algorithm, the Flower Pollination Algorithm is used to search the problem space and find the optimal clusters, and the Big Bang-Big Crunch algorithm is used to solve the local optimal problem and the early convergence of the Flower Pollination Algorithm. The results of the simulations demonstrated the high efficiency of the proposed hybrid algorithm compared to non-hybrid algorithms.

E-ISSN: 2538-4430
ISSN: 2382-9796

EXTENDED ABSTRACT

Introduction

Density-based clustering algorithms are commonly used in machine learning and data mining due to their ability to identify clusters with different shapes and noisy objects. These algorithms are famous in data analysis and the use of their analysis output in industry and business. However, traditional clustering algorithms may have difficulty with datasets with different densities and overlapping neighboring clusters. To address these challenges, a new density-based clustering algorithm is proposed in this article. In this algorithm, the dependency matrix and the first-level search graph are used to find the dense points and the connection between the points. The concept of the relevant space is introduced to define the local and global density, and a central point identification method is used to identify the cluster structures. This algorithm also uses an allocation strategy based on the relevant space for the remaining objects to achieve accurate clustering results. Experimental results on real data sets show the effectiveness of the proposed method in clustering performance.

Clustering is a common type of unsupervised learning technique that plays an important role in data mining and machine learning [1]. Its main goal is to discover the inherent structure in a data set by grouping objects into clusters based on certain criteria. This process aims to minimize the differences in one cluster while maximizing the differences between different clusters. Clustering is a fundamental unsupervised learning approach that is used in various fields, including pattern recognition [2], image processing [3], bioinformatics [4], and information retrieval [5]. Currently, clustering is classified into different types based on different methods such as segmentation-based [6], model-based [7], hierarchy-based [8] and density-based clustering [9].

Methodology

The first step: formation of main sub-clusters.

The second step: creating a graph of the similarity of the reference point with its neighbors by the connection coefficient in the dependency matrix.

The third step: checking the connection of sub-clusters and non-cluster points in the graph.

Results and Discussion

Comparative experiments were designed to check the effectiveness and efficiency of the proposed algorithm in this section. ARI [31] and NMI [32] were used to evaluate the clustering results of different algorithms on real-world datasets. ARI and NMI values ranged from [-1,1] and [0,1], respectively, and the larger the value of these evaluation criteria, the better the clustering result.

The proposed algorithm was implemented in Google Colab with Python programming language, and its performance was evaluated on fourteen real-world datasets from the UCI Machine Learning Repository - Datasets site. Table 1 shows the details of the real-world dataset. There are two real-world data sources: one is the UCI repository containing biological datasets and the other is the KAGLE delta repository.

Table 1. Dataset on a real dataset.

DATASET	D	N	M	TYPE
Iris	4	150	3	REAL
Wine	13	178	3	REAL
Seed	7	210	3	REAL
Ecoil	8	336	8	REAL
WDBC	31	569	2	REAL
Dermatoloy	34	366	6	REAL
Frogs	22	7195	10	REAL
Heart	12	270	2	REAL
Yeast	8	1484	10	REAL
Libras	90	330	15	REAL
Segmentation	19	2100	7	REAL
Pen digits	16	7494	10	REAL
Nursery	8	12960	3	REAL
UJIIndoorLoc	520	21048	3	REAL

Table 2. The Clustering Results on UCI Datasets (ARI).

dataset	GCNN	NCAR	AFK	FKNN-DPC	NCARD	DPC	Our algorithm
Iris	0.90	0.6۲	0.90	0.88	0.90	0.88	**0.۹۱**
Wine	**0.91**	0.71	0.8۱	0.86	0.72	0.69	**0.91**
Seed	0.76	0.75	0.7۷	0.76	0.7۷	0.74	**0.78**
Ecoil	**0.75**	0.۵۷	0.۷۲	0.53	0.72	0.45	**0.75**
WDBC	0.82	0.7۵	0.73	0.69	0.5۲	0.50	**0.83**
Dermatoloy	**0.84**	0.73	0.8۵	0.79	0.7۶	0.72	**0.84**
Segmentation	0.62	0.5۸	0.5۸	0.65	0.56	0.66	**0.67**
Heart	0.33	0.۲0	0.۲5	0.32	0.28	0.31	**0.37**
Yeast	0.26	0.۲۸	0.2۸	0.07	0.14	0.09	**0.30**
Libras	0.36	0.۳۸	0.4۱	0.35	0.3۹	0.35	**0.42**
Pen digits	0.67	0.6۷	0.67	0.58	0.6۶	0.67	**0.6۸**
Frogs	0.81	0.6۸	0.7۴	0.65	0.6۸	0.72	**0.82**
Nursery	0.40	0.2۶	0.۴4	0.35	0.24	0.51	**0.53**
UJIIndoorLoc	0.40	0.5۶	**0.6۲**	0.57	0.۵3	0.43	**0.62**

Average	۰.۶۳	۰.۵۵	۰.۶۲	۰.۵۷	۰.۵۶	۰.۵۵	**۰.۶۷**

Table 3. The Clustering Results on UCI Datasets (NMI).

dataset	GCNN	NCAR	AFK	FKNN-DPC	NCARD	DPC	Our algorithm
Iris	**0.93**	0.68	0.88	0.86	0.89	0.86	**0.93**
Wine	**0.89**	0.79	0.80	0.84	0.81	0.72	0.79
Seed	0.75	0.74	0.75	0.74	0.74	0.72	**0.76**
Ecoil	0.69	0.67	0.67	0.57	0.66	0.59	**0.70**
WDBC	**0.72**	0.69	0.62	0.64	0.69	0.48	**0.72**
Dermatoloy	**0.91**	0.86	0.90	0.86	0.89	0.82	**0.91**
Segmentation	0.75	0.72	0.76	0.76	0.75	0.75	**0.77**
Heart	0.33	0.38	**0.40**	0.28	0.42	0.24	0.33
Yeast	0.29	0.30	0.27	0.15	0.27	0.21	**0.32**
Libras	0.63	0.58	0.68	0.64	0.66	0.64	**0.69**
Pen digits	0.78	0.78	**0.79**	0.77	0.78	0.77	0.72
Frogs	0.69	0.69	0.69	0.60	0.69	0.61	**0.70**
Nursery	0.29	0.29	0.33	0.45	0.51	0.64	**0.66**
UJIIndoorLoc	0.55	0.58	**0.72**	0.70	0.60	0.58	**0.72**
Average	۰.۶۵	۰.۶۲	۰.۶۶	۰.۶۳	۰.۶۴	۰.۶۱	**۰.۶۹**

Conclusions

In the present research, a new clustering algorithm was introduced, which finds clusters based on a dependency matrix and tree density search. A new neighborhood space was introduced to define the local and global density, and a cluster center point search method was used. The proposed algorithm provides a method based on the relevant space to allocate other unclustered data in the last step. The experimental results on the real data set showed the effective performance of the introduced clustering method.

کارافن
فصلنامه علمی دانشگاه ملی مهارت
بهار۱٤۰٤، دوره ۲۲، شماره ۱، ۸۰-٦۰
آدرس نشریه: https://karafan.nus.ac.ir/
10.48301/kssa.2024.446587.2856

یک الگوریتم ترکیبی برای تحلیل خوشه مبتنی بر الگوریتم های گرده افشانی گل و انفجار بزرگ

فرزانه محله[۱]، عبدالرضا حاتم لو*[۲]

۱- دانشجوی دکتری گروه مهندسی کامپیوتر، دانشگاه فنی و حرفه ای، تهران، ایران[1]

۲ *- دانشیار گروه کامپیوتر، واحد خوی، دانشگاه آزاد اسلامی، خوی، ایران[2]

اطلاعات مقاله

دریافت مقاله: ۱۴۰۳/۰۱/۰۶
بازنگری مقاله : ۱۴۰۳/۰۶/۳۰
پذیرش مقاله: ۱۴۰۳/۰۹/۰۷

کلید واژگان:
الگوریتم ترکیبی
خوشه بندی داده ها
الگوریتم گرده افشانی گل
الگوریتم انفجار بزرگ

***نویسنده مسئول:** عبدالرضا حاتم لو
پست الکترونیکی:
rezahatamloo@gmail.com

چکیده

الگوریتم‌های خوشه‌بندی مبتنی بر چگالی به دلیل توانایی آنها در شناسایی خوشه‌هایی با اشکال مختلف و اشیاء نویز معمولاً در یادگیری ماشین و داده‌کاوی استفاده می‌شوند. این الگوریتم‌ها در تحلیل داده‌ها و کاربرد خروجی تحلیل آنها در صنعت و تجارت معروف هستند. با این حال، الگوریتم‌های سنتی خوشه‌بندی ممکن است در مجموعه داده‌هایی با چگالی‌های مختلف و خوشه‌های همسایه درهم‌تنیده مشکل داشته باشند. برای پرداختن به این چالش‌ها، یک الگوریتم خوشه‌بندی مبتنی بر چگالی جدید در این مقاله پیشنهاد شده است. در این الگوریتم ماتریس وابستگی و گراف جستجوی سطح اول برای یافتن نقاط پر چگال و ارتباط بین آنها استفاده شده است، مفهوم فضای مربوطه برای تعریف چگالی محلی و سراسری معرفی می‌گردد، و از یک روش شناسایی نقاط مرکزی برای شناسایی ساختارهای خوشه‌ای استفاده شده‌است. این الگوریتم همچنین از یک استراتژی تخصیص بر اساس فضای مربوطه برای اشیاء باقی مانده برای دستیابی به نتایج خوشه‌بندی دقیق استفاده می‌کند. نتایج تجربی بر روی مجموعه داده‌های واقعی، اثربخشی روش پیشنهادی را در عملکرد خوشه‌بندی نشان می‌دهد.

شاپای الکترونیکی: ۴۴۳۰-۲۵۳۸
شاپای چاپی: ۹۷۹۶-۲۳۸۲

۱. مقدمه

خوشه‌بندی داده‌ها یکی از عملیات مهم در داده‌کاوی است و به یافتن گروه‌ها در مجموعه ای از مشاهدات اطلاق می‌شود به گونه ای که داده‌های متعلق به یک گروه تا حد امکان مشابه و در همان حال با داده‌های موجود در گروه‌های دیگر متفاوت باشند. برای مشابه بودن می‌توان معیارهای مختلفی را در نظر گرفت مثلا می‌توان معیار فاصله را برای خوشه‌بندی مورد استفاده قرار داد و اشیائی را که به یکدیگر نزدیکتر هستند را بعنوان یک خوشه در نظر گرفت که به این نوع خوشه‌بندی، خوشه‌بندی مبتنی بر فاصله نیز گفته می‌شود. خوشه‌بندی ابزار مهمی برای استخراج اطلاعات و دانش از حجم زیادی از داده است، چرا که می‌تواند الگوهای موجود را بدون هیچگونه نظارت و اطلاعات قبلی مانند برچسب داده‌ها تشخیص دهد. به همین دلیل از آن به عنوان یادگیری بدون نظارت هم یاد می‌شود. امروزه داده‌های ذخیره شده در پایگاه داده‌های مختلف با سرعت بسیار در حال رشد است. این داده‌ها حاوی اطلاعات و دانش مفید و پنهانی است که برای مقاصد مختلف قابل استخراج است. در نتیجه ، نیاز به استفاده از روشهای پیشرفته و کارامد برای تجزیه و تحلیل این حجم عظیم داده‌ها و کشف الگوها و اطلاعات پنهان در آنها پیش از پیش ضروری به نظر می رسد. با توجه به اهمیت موضوع محققان بسیاری در سراسر جهان در زمینه طراحی الگوریتم‌های کارامد جدید و همچنین بهبود عملکرد الگوریتم‌های موجود برای خوشه‌بندی فعالیت می‌کنند[2-1].

خوشه‌بندی با طبقه‌بندی متفاوت است. در طبقه‌بندی نمونه‌های ورودی برچسب گذاری شده‌اند ولی در خوشه‌بندی نمونه‌های ورودی دارای بر چسب اولیه نمی‌باشند و در واقع با استفاده از روشهای خوشه‌بندی است که داده‌های مشابه مشخص و بطور ضمنی برچسب گذاری می‌شوند. در واقع می‌توان قبل از عملیات طبقه‌بندی داده‌ها یک خوشه‌بندی روی نمونه‌ها انجام داد و سپس مراکز خوشه‌های حاصل را محاسبه کرد و یک بر چسب به مراکز خوشه‌ها نسبت داد و سپس عملیات طبقه‌بندی را برای نمونه‌های ورودی جدید انجام داد.

هدف خوشه‌بندی یافتن خوشه‌های مشابه از اشیاء در بین نمونه‌های ورودی می‌باشد اما چگونه می‌توان گفت که یک خوشه‌بندی مناسب است و دیگری مناسب نیست؟ می‌توان نشان داد که هیچ معیار مطلقی برای بهترین خوشه‌بندی وجود ندارد بلکه این بستگی به مساله و نظر کاربر دارد که باید تصمیم بگیرد که آیا نمونه‌ها بدرستی خوشه‌بندی شده‌اند یا خیر. با این حال معیار های مختلفی برای خوب بودن یک خوشه‌بندی ارائه شده است که می‌تواند کاربر را برای رسیدن به یک خوشه‌بندی مناسب راهنمایی کند. یکی از مسایل مهم در خوشه‌بندی انتخاب تعداد خوشه‌ها می‌باشد. در بعضی از الگوریتم‌ها تعداد خوشه‌ها از قبل مشخص شده است و در بعضی دیگر خود الگوریتم تصمیم می‌گیرد که داده‌ها به چند خوشه تقسیم شوند.

با وجود گوناگونی روش‌های خوشه‌بندی، هنوز روشی یکتایی وجود ندارد که بتواند تمام انواع خوشه‌ها را به خوبی شناسایی کند؛ از این رو، این کاربر است که باید با توجه به نیازهایش روش مناسب را برگزیند. الگوریتم‌های خوشه‌بندی موجود از یک لحاظ به دو گروه اصلی : روشهای کلاسیک و روشهای اکتشافی تقسیم

می‌شوند. الگوریتم‌های کلاسیک خود به زیرگروه‌های مختلفی از قبیل سلسله مراتبی، پارتیشن بندی، مبتنی بر شبکه و مبتنی بر تراکم تقسیم بندی می‌شوند. الگوریتم خوشه‌بندی k-means یکی از مهمترین و معروفترین الگوریتم‌های کلاسیک است که به طور گسترده ای در کاربردهای مختلف مورد استفاده قرار گرفته است. پیاده سازی این الگوریتم آسان و در اکثر موارد کارامد است ولی در عین حال کارایی آن به شدت وابسته به حالت اولیه مساله و مراکز اولیه خوشه‌ها است و به همین دلیل ممکن است در بهینه محلی گرفتار شود. در طول دو دهه گذشته، به منظور غلبه بر این مشکل، روش های اکتشافی و مبتنی بر جمعیت بسیاری بکار گرفته شده‌اند. از جمله این روشها می‌توان به الگوریتم تبریدی، روش جستجوی ممنوعه، الگوریتم ژنتیک، الگوریتم کلونی مورچه، الگوریتم بهینه‌سازی جمعیت ذرات و الگوریتم کلونی زنبورها اشاره کرد. هیچ کدام از این روشها به تنهایی جوابگوی تمامی کاربردها و انواع مختلف داده‌ها نمی‌باشند و هر کدام مزایا و معایب و محدودیتهایی را دارند به همین دلیل طراحی روشهای نوین که در جهت رفع معایب روشهای موجود عمل کنند یک موضوع باز و روز می‌باشد[3].

تاکنون تعداد زیادی از الگوریتم‌های فرا ابتکاری برای حل مسائل خوشه‌بندی بکار گرفته شده‌اند که هر کدام از آنها نارسائی های و معایبی از جمله: پیچیدگی طراحی و پیاده سازی، تعداد زیاد پارامترها و نیاز به تنظیم مقادیر پارامترها، همگرائی زودرس و گرفتار شدن در بهینه‌های محلی، زمان اجرای بالا، نادقیق بودن و تصادفی بودن و ... را دارند که برای حل این مشکلات تلاشهای زیادی انجام می‌گیرد و یکی از حوزه‌های تحقیقاتی فعال می‌باشد. مسائل زیر نمونه‌هایی برای بهبود عملکرد اکثر الگوریتم‌های بهینه‌سازی فراابتکاری است.

الگوریتم‌های مبتنی بر جمعیت از جمعیت اولیه راه‌حل‌های نامزد شروع می‌شوند. سپس، جمعیت بر اساس مقادیر برازش آنها به سمت راه‌حل بهینه اصلاح می‌شود. این روند تا زمانی که معیار خاتمه برآورده شود تکرار می‌شود. جمعیت اولیه می‌تواند بر کیفیت راه‌حل نهایی و تعداد تکرارهای مورد نیاز برای رسیدن به جواب بهینه تاثیر بگذارد. از این رو، ارتقای کیفیت جمعیت اولیه برای یافتن راه‌حل‌های با کیفیت بالا و کاهش تعداد تکرارها برای رسیدن به راه‌حل بهینه مطلوب است.

یکی از کاستی‌های عمده الگوریتم‌های فراابتکاری که می‌تواند بر عملکرد الگوریتم تاثیر بگذارد، تنظیم پارامترهای الگوریتم است. به عبارت دیگر، نتایج به دست آمده نسبت به تنظیمات پارامتر بسیار حساس هستند. به عنوان مثال، در الگوریتم تبرید شبیه سازی شده پارامترهای زیر درگیر هستند :دمای اولیه، قانون کاهش دما، دمای نهایی، حداکثر تعداد تلاش در یک دما، حداکثر تعداد موفقیت در یک دما. مشکل تنظیم پارامترها مسئله مهمی‌آست که هنوز به طور کامل حل نشده است. تعیین پارامترهای یک الگوریتم خاص وابسته به مسئله است و هیچ راه بهینه ای برای یافتن بهترین ترکیب ممکن از پارامترها وجود ندارد. از این رو، طراحی الگوریتم‌های جدیدی که کمتر به پارامترها وابسته هستند یا تعداد پارامترهای کمی دارند، مطلوب است.

یکی از اشکالات اصلی مرتبط با الگوریتم‌های فراابتکاری، همگرایی زودرس و مشکل بهینه محلی است. این مشکل به ویژه در مسائل با چندین بهینه محلی اتفاق میافتد. همگرایی زودرس به این معنی است که جمعیت در یک منطقه کوچک در فضای جستجو به دام میافتد، به طوری که نمی‌تواند مناطق جدیدی را در فضای جستجو کشف کند که در نتیجه منجر به یک راه‌حل بهینه محلی می‌شود. طراحی و بکارگیری مکانیسم‌ها و استراتژی‌های موثر برای جلوگیری از همگرایی زودرس و بهینه محلی یا کمک به الگوریتم برای فرار از چنین شرایطی مطلوب است.

یکی دیگر از کاستی‌های رفتار همگرایی الگوریتم‌های فراابتکاری این است که عموما در تکرارهای اولیه بسیار سریع همگرا می‌شوند، اما با افزایش تعداد تکرارها، همگرایی بسیار کند می‌شود. به عبارت دیگر، آنها نمی‌توانند کیفیت راه‌حل‌ها را در مجاورت بهینه سراسری بهبود بخشند. برای حل این مشکل، ترکیبی از الگوریتم جستجوی محلی مناسب برای ادامه فرایند جستجو در تکرارهای بعدی قابل استفاده است.

به منظور غلبه بر کاستی‌های ذکر شده در بالا در الگوریتم‌های خوشه‌بندی و بهبود عملکرد، اخیرا الگوریتم‌های خوشه‌بندی ترکیبی برای تحلیل خوشه‌ای توسعه داده شده‌اند. ویژگی‌های الگوریتم‌های مختلف مکمل یکدیگر هستند و ترکیب الگوریتم‌های مختلف می‌توانند نتایج بهتری را تولید نمایند. برای نیل به این هدف، در این تحقیق الگوریتم‌های گرده‌افشانی گل [8] و انفجار بزرگ [5] ترکیب شده‌اند.

روش پیشنهادی

در حالت کلی هدف از بهینه‌سازی یافتن بهترین جواب قابل قبول با توجه به محدودیتها و نیازهای مساله است. روشهای فراابتکاری قادر به حل طیف گسترده‌ای از مسائل بهینه‌سازی با دقت بالا و مناسب می‌باشند[18-4]. این نوع الگوریتم‌ها، از فرایندهای طبیعی و زیستی که در طبیعت وجود دارند، الهام گرفته شده‌اند. الهام از رشد گیاهان و گرده‌افشانی آنها یکی از روشهای جدید تکاملی برای حل مسائل سخت و دشوار است؛ که ایده اصلی الگوریتم گرده‌افشانی گلها بر این اساس ارائه شده است. در این الگوریتم از مفاهیم حرکت و پرواز گرده‌ها در فضای اطراف گلها، برای پخش شدن جمعیت اولیه گلها در فضای یک مساله بهینه‌سازی استفاده می‌شود. این روش از فرایند گرده‌افشانی گیاهان گلدار الهام گرفته شده است. در طبیعت دو نوع گرده‌افشانی خودالقایی و دگر القایی در گیاهان گلدار دیده می‌شود. در خودالقایی گرده‌های یک گل از یک گیاه بر روی یک گل دیگر از همان گیاه قرار می‌گیرد و در دگر القایی گرده‌های یک گل از یک گیاه بر روی یک گل از گیاه دیگر قرار می‌گیرد. الگوریتم گرده‌افشانی[1] (FPA) در سال 2012 به وسیله یانگ معرفی شد[18]. این روش با روشهای ژنتیک و بهینه‌سازی ازدحام ذرات مقایسه شده و بیان شده که عملکرد بهتری نسبت به این دو روش دارد. الگوریتم گرده‌افشانی گل به دلیل دارا بودن مکانیسم مناسب جستجوی محلی و سراسری و ایجاد یک تعادل در این دو نوع از جستجو، از توانایی بالایی در فرار از بهینه‌های محلی برخوردار است و در نتیجه از دقت و سرعت بالاتری در

[1] Flower Pollination Algorithm

همگرایی به جواب بهینه سراسری در مقایسه با الگوریتم ژنتیک، ازدحام گروه ذرات و حتی الگوریتم خفاش برخوردار می‌باشد. با توجه به ویژگیهای گرده‌افشانی در گیاهان گلدار می‌توان سه قانون ساده را برای مدلسازی الگوریتم گرده‌افشانی گل ارائه داد: ۱- گرده‌افشانی از نوع دگر القایی چون از طریق پرواز گرده‌های گل به وسیله حشرات حاصل می‌شود؛ به عنوان گرده‌افشانی سراسری در نظر گرفته می‌شود. ۲- گرده‌افشانی از نوع خودالقایی حالت گرده‌افشانی محلی در نظر گرفته می‌شود. ۳- نوع گرده‌افشانی محلی و گرده‌افشانی سراسری یک گل، به وسیله یک احتمال در بازه $p\in [0, 1]$ در نظر گرفته می‌شود. در دنیای واقعی هر گیاه می‌تواند چندین گل و هر گل تعداد زیادی گرده تولید نماید؛ که این فرض مدلسازی مساله را دشوار می‌نماید. لذا برای سادگی کار می‌توان فرض نمود هر گیاه دارای یک گل و هر گل فقط یک گرده تولید می‌نماید. با این پیش فرض می‌توان الگوریتم گرده‌افشانی گل را در دو گام اساسی گرده‌افشانی سراسری و محلی ایجاد نمود. در گام گرده‌افشانی سراسری، گرده هر گل توسط پرواز حشرات به مسافتهای دور برده می‌شود. این نوع گرده‌افشانی باعث می‌شود که گرده در محدوده وسیعی از مساله جابجا شوند. این رفتار را می‌توان در قالب قانونی که در رابطه ۱ آمده است؛ بیان نمود:

$$x_i^{t+1} = x_i^t + L \times (x_i^t - g_*) \qquad (1)$$

x_i^t مکان گرده i ام در تکرار t ام، x_i^{t+1} مکان گرده i ام در تکرار t+1 ام و g_* مکان بهترین گرده‌ای که تاکنون پیدا شده است. L قدرت گرده‌افشانی نامیده می‌شود که در اصل اندازه یک گام است و میزان جهش و پرش گرده‌ها را نشان می‌دهد. فرض بر این است که قدرت گرده‌افشانی یک عدد مثبت است و به شکل رابطه ۲ ، نشان داده می‌شود

$$L \sim \frac{\lambda \Gamma(\lambda) \sin(\frac{\pi\lambda}{2})}{\pi} \times \frac{1}{s^{1+\lambda}} \quad , s > 0 \qquad (2)$$

در این رابطه $\Gamma(\lambda)$ توزیع گامای استاندارد است و مقدار مناسب برای λ ۱,۵ و برای s ۰,۱ انتخاب می‌شود. گرده‌افشانی محلی یا همان خودالقایی گلها را می‌توان طبق رابطه ۳ مدلسازی نمود.

$$x_i^{t+1} = x_i^t + \varepsilon \times (x_j^t - x_k^t) \qquad (3)$$

در این رابطه، x_j^t و x_k^t دو گرده مختلفی هستند که از گلهای مشابه تولید شده‌اند. ε یک عدد تصادفی بین صفر تا یک است. اگر این عدد کمتر از ۰,۵ باشد گرده‌افشانی محلی و اگر بیشتر از این مقدار باشد

گرده‌افشانی از نوع سراسری انجام می‌شود. در واقع عدد تصادفی میزان وقوع گرده‌افشانی سراسری یا محلی را کنترل می‌نماید. لازم به ذکر است در ابتدا الگوریتم گرده‌افشانی گل برای حل مسائل پیوسته ارائه شد اما در ادامه جهت حل مسائل گسسته، الگوریتم گرده‌افشانی گل باینری ارائه شد که در حل این نوع از مسائل نیز موفق بود.

الگوریتم انفجار بزرگ۲(BB-BC) اولین بار توسط ایرول و اکسین در سال ۲۰۰۶ ارائه شد [5]. این الگوریتم از تکامل یکی از تئوری‌های فراگیر فیزیک و اخترشناسی الهام گرفته شده است که سعی در تشریح چگونگی پیدایش، تکامل و پایان گیتی با عنوان Big Bang Big Crunch دارد. در فاز Big- Bang جهان با استفاده از یک انفجار از حالت متراکم و بسیار ریز شکل گرفته است. پراکندگی و اتلاف انرژی در این فاز سبب بی نظمی و تصادفی بودن می‌شود. لیکن ذراتی که در این فاز به صورت تصادفی توزیع شده بودند در ادامه و در فاز Big Crunch به نظم ویژه‌ای در می‌آیند. الگوریتم انفجار بزرگ-تراکم بزرگ از این جهت که یک جمعیت اولیه را به صورت تصادفی تولید می‌کند، شبیه الگوریتم ژنتیک می‌باشد. ایجاد جمعیت اولیه تصادفی فاز انفجار بزرگ نامیده می‌شود، در این فاز جواب‌های کاندید شده در یک حالت یکنواخت در سرتاسر فضای مساله پراکنده می‌شود. از آنجایی که تولید کننده عدد تصادفی نرمال می‌تواند اعدادی بزرگتر از یک تولید کند، بنابراین ضروری است که مقادیر آنها را محدود کنیم به طوری که در داخل فضای مساله باشند.

به دنبال فاز انفجار بزرگ، فاز تراکم بزرگ پیگیری می‌شود. تراکم بزرگ یک عملگر همگرایی است با ورودی‌های زیاد اما تنها با یک خروجی، که می‌توان ان را مرکز جرم نامید، چرا که تنها خروجی مساله از محاسبه مرکز جرم ورودیها حاصل می‌شود. در اینجا منظور از جرم معکوس مقدار تابع شایستگی است، که برای هر عضو از جمعیت کاندید محاسبه می‌شود. نقطه ای که معرف مرکز جرم است با $\overline{x^c}$ نشان داده می‌شود و براساس رابطه زیر محاسبه خواهد شد:

$$\bar{x}^c = \frac{\sum_{i=1}^{N} \frac{1}{f^i} x^i}{\sum_{i=1}^{N} \frac{1}{f^i}} \qquad (٤)$$

به طوری که x^i نقطه ای در فضای جستجوی N بعدی و f^i مقدار تابع شایستگی متناظر با آن نقطه است. همچنین N اندازه جمعیت (تعداد نقاط) در فاز انفجار بزرگ است.

بعد از فاز تراکم بزرگ، الگوریتم بایستی اعضای جدیدی تولید کند که به عنوان فاز انفجار بزرگ برای تکرار بعدی به کار روند. این کار را از طرق مختلف می‌توان انجام داد. ساده ترین راه پرش به گام اول و تولید جمعیت

[2] Big Bang Big Crunch

اولیه تصادفی است. در این حالت الگوریتم BB- BC هیچ تفاوتی با روش جستجوی تصادفی نخواهد داشت، چرا که در این صورت تکرارهای بعدی از اطلاعات و دانش حاصل شده در تکرارهای قبلی هیچ استفاده‌ای نمی‌کنند، لذا همگرایی الگوریتمی با این توصیف به احتمال زیاد، خیلی کم خواهد بود. یک الگوریتم بهینه‌سازی باید به یک نقطه بهینه همگرا شود، و همزمان با ان، برای اینکه به عنوان یک الگوریتم کلی طبقه بندی شود، بایستی دارای نقاط مخلتف مشخصی در بین جمعیت جستجوی خود با احتمال رو به کاهش باشد. به عبارتی دقیقتر، بعد ار تعداد مشخصی از تکرارهای الگوریتم، تعداد زیادی از جوابهای تولید شده توسط الگوریتم، می‌بایست در پیرامون نقطه به اصطلاح بهینه قرار داشته باشند و تنها تعداد کم باقیمانده از جمعیت جوابها در تمام فضای جستجو پراکنده شده باشند.

نسبت نقاط دور از مقدار بهینه به نقطه جوابهای پیرامون مقدار بهینه باید با افزایش شمار تکرارها، کاهش یابد اما در هیچ حالتی نمی‌تواند مساوی صفر شود، چرا که صفر بودن این نسبت به معنی پایان جستجو است. این همگرایی با به کارگیری دانش قبلی (مرکز جرم) می‌تواند با پراکندن فرزندهای جدید در اطراف مرکز جرم با استفاده از یک عملگر توزیع نرمال در هر راستا انجام شود به طوری که انحراف معیار این تابع توزیع نرمال با افزایش شمار تکرارهای الگوریتم، کاهش یابد. کاندیدهای جدیدی پیرامون مرکز جرم، از طریق افزودن و یا کاستن یک عدد تصادفی نرمال که مقدارش با سپری شدن تکرارها کاهش خواهد یافت، محاسبه می‌شود که ان را می‌توان به شکل زیر فرمول بندی کرد.

$$x^{new} = x_{cm} + \frac{lr}{k} \quad (۵)$$

در این رابطه $\overline{x_{cm}}$ بیانگر مرکز جرم، l معرف حد بالایی پارامتر و r نماد یک عدد تصادفی نرمال هستند و k گام تکرار را نشان می‌دهد. بنابراین x جدید محدود شده از بالا و پایین می‌باشد.

بدیهی است که به لحاظ نظری هیچ محدودیتی در مورد ابعاد فضای جستجو وجود ندارد. این همگرایی می‌تواند به شکل زیر فرمول بندی شود به طوری که مرز فضای جستجو مجموع فواصل اقلیدسی تمام عضوها است.

$$\frac{D^k}{D^{k+1}} > ۱ \quad (۶)$$

که در ان $\overline{D^k}$ مرز فضای جستجو در تکرار k ام و $\overline{D^{k+1}}$ مرز فضای جستجو در تکرار k+1 ام می‌باشد.

بعد از انفجار دوم، مرکز جرم دوباره محاسبه می‌شود. این گام های انبساط و انقباض پی در پی تا زمانی که یک معیار متوقف کننده محقق شود، به طور مداوم تکرار می‌شوند. دو پارامتر لازم برای تولید نقاط تصادفی نرمال، مرکز جرم گام قبلی و انحراف معیار است. مقدار انحراف معیار می‌تواند ثابت در نظر گرفته شود، اما کاهش دادن مقدار آن با سپری شدن تکرارها نتایج بهتری را ایجاد خواهد کرد.

به طور کلی، اکثر روشهای خوشه‌بندی پایه روی جنبه‌های خاصی از داده‌ها تاکید می‌کنند، در نتیجه هر یک از آنها روی مجموعه داده‌های خاصی کارامد هستند. به همین دلیل، نیازمند روشهایی هستیم که بتواند با استفاده از ترکیب این الگوریتم‌ها و بهره گیری از نقاط قوت هر یک، نتایج بهتری را تولید کند. از سوی دیگر به علت بدون ناظر بودن مساله خوشه‌بندی، انتخاب یک الگوریتم خاص به عنوان خوشه‌بندی مناسب، جهت خوشه‌بندی یک مجموعه ناشناس، امری پرخطر و معمولا ناموفق است. همچنین عدم وجود یک معیار واحد در خوشه‌بندی برای ارزیابی، باعث گردیده که هیچ وقت نتوانیم به طور مطلق یک روش را بر روشی دیگر ترجیح دهیم. از طرفی عدم وجود یک روش قطعی برای تصمیم گیری در مورد این که چه نوع خوشه‌بندی برای یک مجموعه داده مناسب است، یکی دیگر از ضعفهای خوشه‌بندی است.

بنابراین، اخیرا توجه بیشتر به ساخت چارچوب‌هایی شده است که چند الگوریتم خوشه‌بندی را ترکیب کنند. در خوشه‌بندی به دلیل اینکه یک الگوریتم برای هر نوع مجموعه داده مورد استفاده قرار می‌گیرد و از آنجایی که هر مجموعه داده دارای پیچیدگی‌های خاص خود می‌باشد، لذا عموما هر الگوریتم خوشه‌بندی بر روی برخی مجموعه‌های داده‌ای عملکرد تا حدودی مطلوب و بر روی سایرین عملکرد ضعیفی دارد. لذا امروزه خوشه‌بندی ترکیبی جهت فائق آمدن بر این مساله مورد توجه قرار گرفته است. در خوشه‌بندی ترکیبی به نوعی با بهره‌مندی از خردجمعی و ترکیب کردن مجموع این راه‌حل‌ها، همواره می‌توان راه‌حل بهینه را پیدا کرد.

معمولا ترکیب چندین الگوریتم می‌تواند نتایج و راه‌حل‌های بهتری ارائه دهد. از آنجایی که ویژگی‌های الگوریتم‌های مختلف مکمل یکدیگر هستند، این الگوریتم‌ها را می‌توان به روش‌های مختلف ترکیب کرد تا عملکرد بهتری حاصل شود. الگوریتم پیشنهادی ترکیب الگوریتم‌های گرده‌افشانی گلها (FPA) و الگوریتم انفجار بزرگ (BB-BC) می‌باشد.

عملکرد قابل قبول یک الگوریتم جستجو به ایجاد تعادل مناسب بین استخراج و اکتشاف پاسخ‌ها در فضای مساله وابسته است. ویژگی استخراج یک الگوریتم به‌معنی بهره‌برداری درست از یک جواب خوب قبلا پیدا شده و اکتشاف به‌معنی یافتن نقاط جدید در فضای جستجو می‌باشد. قابلیت استخراج الگوریتم گرده‌افشانی گل در مقایسه با قابلیت اکتشاف ان بالا است که احتمال به دام افتادن در یک بهینه‌ی محلی را افزایش می‌دهد. از طرفی قابلیت اکتشاف الگوریتم BB-BC (فاز انفجار) بالا است و می‌تواند جمعیت جوابهای مساله را در بخش های وسیع و مختلفی از فضای مساله به حرکت در آورد. لذا در الگوریتم ترکیبی سعی کرده‌ایم از قابلیت اکتشاف الگوریتم انفجار بزرگ و قابلیت استخراج الگوریتم گرده‌افشانی گل به طور همزمان استفاده نماییم تا جوابهای بهتر و با کیفیت‌تری را برای مساله خوشه‌بندی پیدا نماییم.

الگوریتم پیشنهادی با الگوریتم گرده‌افشانی گل شروع می‌شود و شروع به جستجوی فضای مساله می‌نماید. بعد از تعدادی تکرار و بعد از اینکه جوابهای کاندید به سمت بهترین جواب کاندید حرکت نمودند و در اطراف آن جمع شدند تحرک جمعیت جوابها برای جستجوی قضای مساله کم می‌شود و جوابهای کاندید به کندی و با

سرعت پایینی حرکت می‌کنند و از جستجوی سایر بخشهای فضای مساله باز می‌مانند. در چنین شرایطی الگوریتم انفجار بزرگ برای حل این مشکل و افزایش تحرک جمعیت و پراکنده کردن جوابهای کاندید در فضای مساله بکار گرفته خواهد شد. سپس الگوریتم گرده‌افشانی گل دوباره با جمعیت جدیدی شروع به جستجوی فضای مساله برای پیدا کردن جواب بهینه خواهد نمود. این گامها و مراحل بصورت متوالی انجام خواهد شد تا شرط پایانی محقق شود. در ادامه توضیحات مراحل الگوریتم پیشنهادی شامل الگوریتم‌های گرده‌افشانی گل و انفجار بزرگ آورده می‌شود.

در الگوریتم ترکیبی، پس از هر بار تکرار الگوریتم FPA، با انتخاب نیمی‌از جمعیت که عملکرد ضعیف‌تری دارند، با استفاده از الگوریتم BB-BC دوباره در فضای مساله پراکنده شده و با دقت بالا حرکت می‌کنند. بنابراین، احتمال همگرایی زودرس و گیر افتادن در نقاط بهینه محلی کاهش می‌یابد. در ابتدا تعداد P جواب کاندید به عنوان جمعیت اولیه به صورت تصادفی تولید می‌شوند. این جوابها را می توان به عنوان گرده گل در الگوریتم FPA در نظر گرفت.

با توجه به توضیحات فوق شبه کد الگوریتم ترکیبی FPA-BB-BC بصورت زیر می‌باشد:

مقداردهی به پارامترهای الگوریتم

ساخت جمعیت اولیه گلها به صورت تصادفی

ارزیابی موقعیت هر گل و محاسبه شایستگی آن

شناسایی بهترین گل به عنوان g*

تا زمانی که شرط توقف برقرار نشده است مراحل ۶ تا ۱۴ تکرار شود.

مرتب‌سازی جمعیت جوابها بر اساس مقدار تابع تناسب

پراکنده کردن دوباره نصف جمعیت که مقدار تناسب کمتری دارند با استفاده از BB-BC

برای هر گل مراحل 7 تا 11 تکرار شود.

به احتمال P به روزرسانی موقعیت گل با گرده‌افشانی محلی انجام شود.

به احتمال 1-p به روزرسانی موقعیت گل با گرده‌افشانی سراسری انجام شود.

شایستگی گل جدید محاسبه شود.

اگر میزان شایستگی گل جدید بهتر از گل فعلی است، گل جدید جایگزین گل فعلی شود.

اگر شایستگی گل جدید بهتر از g* است، جایگزین g* شود.

اگر شرط توقف برقرار نشد به مرحله 5 بازگشته و در غیر اینصورت الگوریتم پایان می‌یابد.

ارزیابی و تحلیل نتایج

چندین مجموعه داده استاندارد و واقعی از سایت UCI برای تحلیل و ارزیابی عملکرد الگوریتم پیشنهادی استفاده شده است. تعداد رکوردها، فیلدها و کلاسترهای این مجموعه داده‌ها متفاوت هستند و با استفاده از آنها رفتار الگوریتم تحت شرایط مختلف قابل بررسی است. مشخصات اصلی این مجموعه داده‌ها در جدول ۱ خلاصه شده است[3].

جدول ۱ مشخصات مجموعه داده‌های استفاده شده

مجموعه داده	تعداد خوشه‌ها	تعداد خصیصه‌ها	تعداد رکورد ها
Iris	3	4	150 (50, 50, 50)
Wine	3	13	178 (59, 71, 48)
Cancer	2	9	683 (444, 239)
Glass	6	9	214 (70, 76, 17, 13, 9, 29)
Vowel	6	3	871 (72, 89, 172, 151, 207, 180)

مجموعه داده Iris شامل ۱۵۰ نمونه‌ی جمع‌آوری شده از گل‌های زنبق است که این نمونه‌ها ۵۰ نمونه از هر یک از سه نوع گل زنبق را شامل می‌شوند. برای هر یک از نمونه‌ها ۴ ویژگی گل زنبق اندازه‌گیری شده‌است. این ویژگی‌ها شامل طول و عرض کاسبرگ و گلبرگ، بر حسب سانتی‌متر است. این مجموعه داده به عنوان یک مثال پرکاربرد در زمینه‌های آماری و یادگیری ماشین مورد استفاده قرار گرفته‌است.

مجموعه داده Wine شامل پیش‌بینی کیفیت نوشیدنی‌ها است که با شاخص های شیمیایی، کیفیت هر نوشیدنی را می‌سنجد. این مجموعه داده شامل داده‌های مربوط به نتایج یک تجزیه شیمیایی از سه رقم مختلف شراب تولید شده در یک منطقه در ایتالیا است. این تجزیه و تحلیل مقدار ۱۳ ترکیب موجود در هر یک از سه نوع شراب را تعیین می‌کند.

مجموعه داده Cancer یا سرطان پستان ویسکانسین دارای ۶۸۳ نمونه با ۹ ویژگی است. این مجموعه داده دو خوشه به نام های بدخیم و خوش‌خیم دارد که نمونه های این مجموعه داده در آنها قرار می‌گیرند.

مجموعه داده Glass تعریف ۶ نوع شیشه توسط خدمات پزشکی قانونی ایالات متحده آمریکا بر حسب محتوای اکسید آنها می‌باشد. مطالعه طبقه‌بندی انواع شیشه با انگیزه تحقیقات جرم‌شناسی انجام می شود. در صحنه جنایت از شیشه باقی مانده در صورت شناسایی و طبقه‌بندی صحیح می توان به عنوان مدرک استفاده کرد.

مجموعه داده Vowel شامل ۸۷۱ آوای هندی است. مجموعه داده شامل ۶ کلاس است که با هم تداخل دارند و شامل نمونه‌هایی با ابعاد کم، متوسط و زیاد است.

الگوریتم پیشنهادی با استفاده از نرم افزار MATLAB 2018a شبیه‌سازی شده است. نتایج الگوریتم‌ها بر اساس معیار مربع مجموع فواصل درون خوشه‌ای گزارش شده‌اند. هدف این معیار که با نام مربع خطاها نیز معروف است کمینه کردن مجموع فواصل داده‌های موجود در داخل خوشه‌ها از مراکز خوشه‌ها می‌باشد که همان تلاش در جهت قرار دادن داده‌های مشابه و نزدیک به هم در یک خوشه واحد می‌باشد و بصورت زیر تعریف می‌شود:

(۷)

در این فرمول N تعداد داده‌های موجود در مجموعه داده مورد نظر و k تعداد خوشه‌ها را مشخص می‌کند. عبارت فاصله بین داده Oi و مرکز خوشه مربوطه (Zl) را تعیین می‌کند. از بین روشهای مختلفی که برای محاسبه فواصل بین دو شئ داده‌ای در بحث خوشه‌بندی وجود دارند فاصله اقلیدسی معروفترین و پر استفاده‌ترین آنها می‌باشد که در این کار هم ما از آن استفاده کرده‌ایم و بصورت زیر تعریف شده است:

(8)

که در آن Oi و Oj دو شئ داده‌ای و d تعداد ویژگیها یا خصایص آنها می‌باشد.

نتایج الگوریتم پیشنهادی در جداول شماره ۲ تا ۶ آورده شده‌اند. به خاطر اینکه پایه الگوریتم‌ها مبتنی بر تولید مراکز تصادفی و سپس بهبود این مراکز تصادفی می‌باشد و در اجراهای مختلف ممکن است نتایج متفاوتی حاصل شود، لذا هر کدام از الگوریتم‌ها را ۵۰ بار اجرا کرده و از میان ۵۰ جواب، بهترین، بدترین و متوسط جوابها را آورده ایم. همچنین انحراف معیار استاندارد جوابهای بدست آمده گزارش شده‌اند. واضح است که مقدار کمتر برای انحراف معیار مطلوب است و نشانگر آن است که الگوریتم مربوطه قابلیت اطمینان و پایداری بیشتری

دارد و بالعکس. نتایج الگوریتم پیشنهادی با الگوریتم‌های K-means, GSA[4], BB-BC[5], PSO[3] مقایسه شده‌اند[17].

جدول ۲ مجموع فواصل درون خوشه‌ای الگوریتم‌ها[17] برای مجموعه داده Iris

Algorithm	Best	Average	Worst	STD
K-mean	97.325920	105.72902	128.40420	12.38759
PSO	96.879350	98.142360	99.769520	0.842070
GSA	96.687940	96.731050	96.824630	0.027610
BB-BC	96.676480	96.765370	97.428650	0.204560
FPA- BB-BC	96.660124	96.738143	97.258061	0.14029

جدول ۲ شامل خلاصه‌ای از مجموع فاصله درون خوشه‌ای به دست آمده توسط الگوریتم‌ها در مجموعه دادهIris است. نتایج جدول نشان می‌دهد که چگونه الگوریتم پیشنهادی FPA-BB-BC بهترین نتایج را در بین همه الگوریتم‌های رقیب به دست آورده است. انحراف معیار کوچک نشان می‌دهد که نتایج در طول ۵۰ اجرا بسیار نزدیک به یکدیگر هستند که نشان دهنده رفتار پایدار الگوریتم پیشنهادی است. از جدول، به راحتی می توان نتیجه گرفت که نتایج به دست امده توسط الگوریتم FPA-BB-BC ، بر سایر الگوریتم‌ها غالب است.

به طور مشابه، جدول ۳ شامل خلاصه‌ای از مجموع فاصله درون خوشه‌ای به دست آمده برای مجموعه داده Wine است. بهترین راه‌حل الگوریتم FPA-BB-BC به طور قابل توجهی بهتر از الگوریتم‌های دیگر است. همچنین، بدترین و متوسط مجموع فواصل درون خوشه‌ای به دست آمده توسط FPA-BB-BC نیز بهتر از سایر الگوریتم‌ها می‌باشد. مقدار کوچکتر انحراف استاندارد نشان دهنده رفتار پایدار و همچنین توانایی بالای الگوریتم پیشنهادی برای همگرایی به راه‌حل‌های خوب است.

جدول ۳ مجموع فواصل درون خوشه‌ای الگوریتم‌ها [17]برای مجموعه دادهWine

Algorithm	Best	Average	Worst	STD

[3] Particle Swarm Optimization
[4] Gravitional Search Algorithm
[5] Big Bang-Big Crunch

K-mean	16,555.67942	16,963.04499	23,755.04949	1180.6942
PSO	16,304.48576	16,316.27450	16,342.78109	12.602750
GSA	16,313.87620	16,374.30912	16,428.86494	4 34.671220
BB-BC	16,298.67356	16,303.41207	16,310.11354	2.6619800
FPA- BB-BC	16,295.48216	16,299.82549	16,304.74231	1.96873

جدول ۴ نتیجه حاصل از آزمایش الگوریتم‌ها را بر روی مجموعه داده Glass خلاصه می‌کند. مجموع فواصل درون خوشه‌ای بدست آمده توسط الگوریتم پیشنهادی بهتر از الگوریتم‌های دیگر است. انحراف معیار استاندارد کوچکتر نشان دهنده ثبات و همچنین توانایی همگرایی الگوریتم FPA-BB-BC برای به دست آوردن راه‌حل‌های نزدیک به بهینه است.

جدول ۴ مجموع فواصل درون خوشه‌ای الگوریتم‌ها [17] برای مجموعه دادهGlass

Algorithm	Best	Average	Worst	STD
K-mean	215.67753	227.97785	260.83849	14.1388
PSO	223.90546	230.49328	246.08915	4.79320
GSA	224.98410	233.54329	248.36721	1 6.13946
BB-BC	223.89410	231.23058	243.20883	4.65013
FPA- BB-BC	217.47538	221.21637	231.40538	2.02187

نتایج آزمایشات بر روی مجموعه داده Cancer در جدول ۵ نشان داده شده است. در این مجموعه داده هم الگوریتم پیشنهادی FPA-BB-BC عملکرد بهتری را از نظر بهترین، میانگین و بدترین جوابهای بدست آمده نشان می‌دهد. یک رفتار پایدار از طریق مقدار کوچکتر انحراف معیار استاندارد برای الگوریتم پیشنهادی مشاهده می‌شود.

جدول ۵ مجموع فواصل درون خوشه‌ای الگوریتم‌ها[17] برای مجموعه دادهCancer

Algorithm	Best	Average	Worst	STD
K-mean	2986.96134	3032.24781	5216.08949	315.1456
PSO	2974.48092	2981.78653	3053.49132	10.43651
GSA	2965.76394	2972.66312	2993.24458	8.918600
BB-BC	2964.38753	2964.38798	2964.38902	0.000480
FPA- BB-BC	2964.36895	2964.37721	2964.38011	0.000394

جدول ۶ نتایج مجموعه داده‌های CMC را خلاصه کرده است. الگوریتم FPA-BB-BC دوباره بر همه الگوریتم‌های دیگر غلبه کرده و منجر به انحراف معیار استاندارد کوچک‌تر شد که نشان‌دهنده نزدیکی نتایج آزمایش‌ها به یکدیگر در هر اجرا است.

جدول ۶ مجموع فواصل درون خوشه‌ای الگوریتم‌ها[17] برای مجموعه دادهCMC

Algorithm	Best	Average	Worst	STD
K-mean	5542.18214	5543.42344	5545.33338	1.523840
PSO	5539.17452	5547.89320	5561.65492	7.356170
GSA	5542.27631	5581.94502	5658.76293	41.13648
BB-BC	5534.09483	5574.75174	5644.70264	39.43494
FPA- BB-BC	5533.00891	5540.48527	5598.63280	4.80754

بطور کلی نتایج بدست آمده در جداول فوق نشان می‌دهند ترکیب الگوریتم‌های FPA و BB-BC در مقایسه با سایر الگوریتم‌ها عملکرد بهتری را دارد چنانکه جوابهای یافته شده توسط الگوریتم پیشنهادی بهتر و با کیفیت‌ترهستند و علاوه بر این، انحراف معیار استاندارد جوابهای یافته شده در اجراهای مختلف توسط الگوریتم پیشنهادی کمتر از سایر الگوریتم‌ها است که نشانگر قابلیت اطمینان و پایدار بودن رویکرد پیشنهادی در شرایط مختلف می‌باشد.

نتیجه‌گیری و کارهای آتی

در این مقاله یک الگوریتم فراابتکاری ترکیبی برای حل مساله خوشه‌بندی ارائه گردید که مبتنی بر الگوریتم‌های گرده‌افشانی گل و انفجار بزرگ می‌باشد. در الگوریتم پیشنهادی الگوریتم گرده‌افشانی گل برای جستجوی فضای مساله و پیدا کردن خوشه‌های بهینه استفاده می‌شود و الگوریتم انفجار بزرگ برای کمک به خروج از بهینه‌های محلی و جلوگیری از همگرایی زودرس استفاده شده است. نتایج شبیه سازی ها با مجموعه داده‌های مختلف نشان داد که الگوریتم پیشنهادی در مقایسه با الگوریتم‌های معروف و پرکاربرد موجود عملکرد بهتری دارد. بطوریکه هم کیفیت جواب های پیدا شده و هم انحراف معیار جوابهای الگوریتم پیشنهادی در اجراهای مختلف بهتر از الگوریتم‌های مورد مقایسه می‌باشد که نشانگر قابلیت اطمینان بالا و پایداری الگوریتم در شرایط متفاوت می‌باشد. با توجه به اینکه ویژگیهای الگوریتم‌های فراابتکاری متفاوت و مکمل هم هستند برای تحقیقات بعدی پیشنهاد می‌شود ترکیب الگوریتم‌های دیگر و جدیدتر برای حل مساله خوشه‌بندی استفاده شود و با نتایج الگوریتم‌های موجود مقایسه شود.

منابع

[1] Han, J., Kamber, M., & Pei, J. (2012). Data mining concepts and techniques third edition. University of Illinois at Urbana-Champaign Micheline Kamber Jian Pei Simon Fraser University.

[2] Jain, A. K. (2010). Data clustering: 50 years beyond K-means. Pattern recognition letters, 31(8), 651-666.

[3] Hatamlou, A. (2013). Black hole: A new heuristic optimization approach for data clustering. Information sciences, 222, 175-184.

[4] Yang, X. S. (2020). Nature-inspired optimization algorithms: Challenges and open problems. Journal of Computational Science, 46, 101104.

[5] Erol, O. K., & Eksin, I. (2006). A new optimization method: big bang–big crunch. Advances in engineering software, 37(2), 106-111.

[6] Molina, D., Poyatos, J., Ser, J. D., García, S., Hussain, A., & Herrera, F. (2020). Comprehensive taxonomies of nature-and bio-inspired optimization: Inspiration versus algorithmic behavior, critical analysis recommendations. Cognitive Computation, 12, 897-939.

[7] Agrawal, P., Abutarboush, H. F., Ganesh, T., & Mohamed, A. W. (2021). Metaheuristic algorithms on feature selection: A survey of one decade of research (2009-2019). Ieee Access, 9, 26766-26791.

[8] Wang, S., Jia, H., Abualigah, L., Liu, Q., & Zheng, R. (2021). An improved hybrid aquila optimizer and harris hawks algorithm for solving industrial engineering optimization problems. Processes, 9(9), 1551.

[9] Abderazek, H., Yildiz, A. R., & Mirjalili, S. (2020). Comparison of recent optimization algorithms for design optimization of a cam-follower mechanism. Knowledge-Based Systems, 191, 105237.

[10] Guo, H., Gu, W., Khayatnezhad, M., & Ghadimi, N. (2022). Parameter extraction of the SOFC mathematical model based on fractional order version of dragonfly algorithm. International Journal of Hydrogen Energy, 47(57), 24059-24068.

[11] Ezugwu, A. E., Shukla, A. K., Nath, R., Akinyelu, A. A., Agushaka, J. O., Chiroma, H., & Muhuri, P. K. (2021). Metaheuristics: a comprehensive overview and classification along with bibliometric analysis. Artificial Intelligence Review, 54, 4237-4316.

[12] Darwish, A., Hassanien, A. E., & Das, S. (2020). A survey of swarm and evolutionary computing approaches for deep learning. Artificial intelligence review, 53(3), 1767-1812.

[13] Tzanetos, A., & Dounias, G. (2021). Nature inspired optimization algorithms or simply variations of metaheuristics?. Artificial Intelligence Review, 54(3), 1841-1862.

[14] Dhal, K. G., Das, A., Ray, S., Gálvez, J., & Das, S. (2020). Nature-inspired optimization algorithms and their application in multi-thresholding image segmentation. Archives of Computational Methods in Engineering, 27(3), 855-888.

[15] Ahmed, A. N., Van Lam, T., Hung, N. D., Van Thieu, N., Kisi, O., & El-Shafie, A. (2021). A comprehensive comparison of recent developed meta-heuristic algorithms for streamflow time series forecasting problem. Applied Soft Computing, 105, 107282.

[16] Pal, S. S., & Pal, S. (2020). Black hole and k-means hybrid clustering algorithm. In Computational Intelligence in Data Mining: Proceedings of the International Conference on ICCIDM 2018 . Springer Singapore.

[17] Deeb, H., Sarangi, A., Mishra, D., & Sarangi, S. K. (2022). Improved Black Hole optimization algorithm for data clustering. Journal of King Saud University-Computer and Information Sciences, 34(8), 5020-5029.
[18] Yang, X. S. (2012, September). Flower pollination algorithm for global optimization. In International conference on unconventional computing and natural computation. Berlin, Heidelberg: Springer Berlin Heidelberg.