The 9th International Conference on Web Research
ICWR2023
May 3th-4th, 2023, Tehran, Iran

# Finding Optimal Parameters for ADBSCAN Clustering Algorithm using Genetic Algorithm

Motahareh Entezami*, Ali Shakiba

Department of Computer Science, Vali-e-Asr University of Rafsanjan, Iran
m.entezami.98@gmail.com, ali.shakiba@vru.ac.ir

## Abstract

Clustering is the process of partitioning a set of objects into disjoint groups, each partition is called a cluster. Intuitively, it is desirable that the members in each cluster are very similar to each other in terms of their characteristics. As well, it is desirable to have a low degree of similarity between members in different clusters. In general, clustering algorithms can be categorized to follow either a partitioning, a hierarchical, a density, a model-based or any combination of these approaches.
The ADBSCAN algorithm is a density-based clustering algorithm which presents a new method to identify high-density local instances considering the properties of the nearest neighbor graph. Two parameters are used in this algorithm, namely the parameter k representing the number of nearest neighbors, and the percentage of noise in the data set. These parameters have a significant effect on the quality of the output as well as the required time. Therefore, it is necessary to find optimal values for these parameters. Brute-force search is one of the naïve ways to this end. However, evolutionary-based algorithms such as genetic search methods can be used to make the search process easy and efficient. In this paper, we applied the genetic algorithm to get optimal values of the parameters. The proposed method led to an 11.46% improvement in the ARI criterion, on average.

**Keywords**:Density-Based Clustering, ADBSCAN, Genetic Algorithm.

The 9th International Conference on Web Research
ICWR2023
May 3th-4th, 2023, Tehran, Iran

# یافتن پارامترهای بهینه برای الگوریتم خوشه‌بندی ADBSCAN با استفاده از الگوریتم ژنتیک

مطهره انتظامی* [۱]، علی شکیبا [۲]

[۱] کارشناسی ارشد علوم کامپیوتر، علوم کامپیوتر، دانشگاه ولی عصر، رفسنجان
m.entezami.98@gmail.com

[2]استادیار گروه علوم کامپیوتر، دانشگاه ولی عصر، رفسنجان
ali.shakiba@vru.ac.ir

## چکیده

**خوشه‌بندی، فرآیندی است که مجموعه‌ای از اشیاء را به گروه‌های مجزا افراز می‌کند که هر افراز یک خوشه نامیده می‌شود. در یک خوشه‌بندی، مطلوب است تا اعضاء هر خوشه از لحاظ ویژگی‌ها، به یکدیگر شبیه باشند. همچنین، لازم است تا میزان شباهت بین نمونه‌هایی که در خوشه‌های متفاوت هستند، پایین باشد. به صورت کلی، الگوریتم‌های خوشه‌بندی از یکی از رویکردهای افرازی، سلسله‌مراتبی، چگالی، مبتنی بر مدل و یا ترکیبی از آن‌ها استفاده می‌کنند. الگوریتم ADBSCAN، الگوریتمی برای خوشه‌بندی دادگان و مبتنی بر چگالی است. این الگوریتم، یک روش جدید برای شناسایی نمونه‌های محلی با چگالی بالا با استفاده از خواص ذاتی گراف نزدیکترین همسایگی را ارائه می‌کند. در این الگوریتم، از دو پارامتر $k$ (تعداد نزدیکترین همسایگان) و درصد نویز در مجموعه داده استفاده می‌شود. این دو پارامتر، تأثیر به سزایی در نتیجه محاسبات و کیفیت خروجی دارند. بنابراین، لازم است تا این دو مقدار در بهینه‌ترین حالت ممکن تنظیم شوند. جستجوی فراگیر، یکی از راهکارهای یافتن مقدار بهینه است. به منظور کاهش زمان جستجو، در این مقاله از روش جستجوی ژنتیک برای یافتن مقادیر بهینه‌ی این پارامترها استفاده شده است. با به کارگیری روش پیشنهادی، به صورت متوسط، ۱۱/٤٦ درصد بهبود در معیار ARI حاصل شده است.**

## کلمات کلیدی

**خوشه‌بندی مبتنی بر چگالی، ADBSCAN ، الگوریتم ژنتیک.**

## ۱- مقدمه

خوشه‌بندی یک مسأله‌ی شناسایی الگوی بدون ناظر است که در چندین زمینه مانند موتورهای جستجو [۱]، سیستم توصیه‌گر [۲]، یافتن اجتماعات در شبکه‌های اجتماعی [۳]، تقسیم‌بندی تصویر، تشخیص شیء، جستجوی نزدیکترین همسایگان و پزشکی کاربرد دارد [۴]. مسأله‌ی خوشه‌بندی به معنی افراز داده‌ها به مجموعه‌ای از خوشه‌ها می‌باشد به طوری که نمونه‌های یک خوشه مشابه یکدیگر باشند و نمونه‌های خوشه‌های مختلف با یکدیگر متفاوت باشند.

برای تخمین پارامترهای الگوریتم‌های خوشه‌بندی اغلب از روش‌های آماری و کلاسیک یا ترکیب الگوریتم‌های داده‌کاوی با یکدیگر استفاده می‌شود که طولانی و پیچیده هستند.

در سال‌های اخیر محققین بسیاری با استفاده از الگوریتم‌های بهینه‌سازی فرا ابتکاری، بهبود مسائل خوشه‌بندی از جمله خوشه‌بندی K-means [5] و DBSCAN [6]را مورد بررسی و آزمایش قرار داده‌اند که موفقیت آمیز بوده است. برای نمونه در [۷]، روشی ارائه شده است که عملکرد الگوریتم خوشه‌بندی K-means در داده‌های با ابعاد بالا را با استفاده از الگوریتم بهینه‌سازی فرا ابتکاری کلونی مورچگان بهبود بخشیده است. در [۸] روشی

ارائه شده که با استفاده از الگوریتم NSGA-II[1] پارامترهای الگوریتم DBSCANرا از طریق تکرار و توابع تناسب تنظیم می‌کند تا دقت خوشه‌بندی را افزایش دهد.

در این مقاله قصد داریم تا با استفاده از الگوریتم بهینه‌سازی ژنتیک [۹]، مقادیر پارامترهای الگوریتم خوشه‌بندی ADBSCAN [10] را به گونه‌ای تخمین بزنیم که بتوان از آن‌ها در جهت بهبود عملیات خوشه‌بندی استفاده کرد.

سازمان مقاله بدین شرح است: در بخش ۲، ابتدا به بررسی الگوریتم ADBSCAN خواهیم پرداخت و سپس، الگوریتم ژنتیک مرور خواهد شد. در بخش ۳، رویکرد پیشنهادی برای جستجوی بهینه‌ی پارامترها تشریح می‌شود. عملکرد روش پیشنهادی در بخش ۴ مورد ارزیابی و تحلیل قرار خواهد گرفت. در بخش ۵، جمع بندی و سؤالات پژوهشی مطرح خواهند شد.

## ۲- پیش‌نیازها

### ۲-۱- الگوریتم ADBSCAN

در ابتدا تعاریف گراف $k$–نزدیکترین همسایه و نمونه‌ی هسته را که برای صحت الگوریتم خوشه‌بندی ADBSCAN حائز اهمیت می‌باشد را بیان می‌کنیم:

**تعریف ۱- گراف $k$–نزدیکترین همسایه**

یک گراف $k$–نزدیکترین همسایه گراف جهت‌داری است که با $G = (V.E)$ نشان داده می‌شود. $V$ مجموعه‌ای از نمونه‌ها است و $\forall(u\,.\,v) \in E$، $v$ یک $k$–نزدیکترین همسایه از $u$ است.

**تعریف ۲- نمونه‌ی هسته**

اگر $x_q$ شرایط (۱) را داشته باشد، $x_q$ یک نمونه‌ی هسته است:

$$\rho(x_q) \le \text{mean}(\rho(x)) + \varphi^{-1}\left(1 - noise_{persent}\right) * \sigma(\rho(x)), \quad (۱)$$

که mean تابع میانگین است و $\varphi^{-1}$ تابع کیفیت توزیع نرمال است. $\sigma$ نیز تابع انحراف معیار استاندارد است. $noise\_percent$ پارامتر الگوریتم A D B S C A N است و مقدار آن بین صفر و یک می‌باشد. الگوریتم ADBSCAN مبتنی بر یک رویکرد جدید برای شناسایی نمونه‌هایی است که در مناطق متراکم هستند و از گراف نزدیکترین همسایه استفاده می‌کند. این الگوریتم از دو پارامتر $k$ و درصد نویز استفاده می‌کند که به ترتیب تعداد نزدیکترین همسایگان و نسبت نویز مجموعه داده را نشان می‌دهند.

نحوه‌ی عملکرد آن به این صورت است که با مطالعه‌ی ماهیت نمودار نزدیکترین همسایه و استفاده از این خصوصیات، به طور مستقیم نمونه‌های متراکم محلی را بدون هیچ پارامتر اضافی شناسایی می‌کند و نمونه‌های هسته را با استفاده از فرض توزیع‌های آماری و گراف $k$–نزدیکترین همسایه پیدا می‌کند. سپس زیرگراف‌های نزدیک به هم را به خوشه‌های یکسان اختصاص می‌دهد و زیرگراف‌هایی را که در مناطق پراکنده هستند حذف می‌کند. از این رو، ADBSCAN به جای اینکه به پارامترهای استاتیک تعریف شده توسط کاربر برای تخمین چگالی هر نمونه و تعیین اینکه آیا نمونه هسته است، وابسته باشد، نمونه‌های محلی با چگالی بالا را به طور مستقیم از نمودار نزدیکترین همسایه شناسایی می‌کند.

این الگوریتم مزایای الگوریتم‌های خوشه‌بندی مبتنی بر چگالی را به ارث می‌برد و برخی از کاستی‌های الگوریتم‌های قبلی مانند عدم شناسایی خوشه‌های دارای تراکم متفاوت و حساس بودن به پارامترها را برطرف می‌کند. اما در این روش نمی‌توانستیم بهترین مقادیر را برای پارامترهای $k$ و درصد نویز تخمین بزنیم که در روش پیشنهادی این امکان فراهم شده است تا مقادیر بهینه‌ی پارامترهای الگوریتم ADBSCAN را با استفاده از الگوریتم ژنتیک بیابیم.

### ۲-۲- الگوریتم ژنتیک

ایده الگوریتم ژنتیک که توسط جان هالند در سال ۱۹۶۷ ابداع شده است از تکنیک‌های زیست‌شناسی مانند وراثت، جهش، انتخاب طبیعی و ترکیب الهام گرفته شده است.

الگوریتم‌های ژنتیک، فرآیند تکامل در طبیعت را شبیه‌سازی می‌کنند و با هدف یافتن بهترین جواب ممکن برای یک مسئله، در فضای جواب‌های کاندید به جستجو می‌پردازند. این عملیات در الگوریتم ژنتیک، با تولید یک جمعیت اولیه از رشته‌های تصادفی آغاز می‌شود. این رشته‌ها معادل کروموزوم‌ها یا جواب‌های کاندید مسئله هستند. در هر نسل از این الگوریتم، تغییرات خاصی در ژن‌های کروموزوم‌های تشکیل دهنده جمعیت ایجاد می‌شود. به طور معمول جمعیت جواب‌ها در هر نسل به سمت جواب بهینه همگرا می‌شوند.

در الگوریتم ژنتیک جمعیت اولیه تحت تأثیر سه دسته عملگر اصلی قرار می‌گیرند و جمعیت جدیدی در فضای جواب مسئله تولید می‌شود. عملگرهای اصلی الگوریتم ژنتیک عبارتند از: عملگر تولید مثل یا انتخاب[2]، عملگر ترکیب و عملگر جهش[3].

عملگر تولید مثل، بهترین رشته‌ها یا بهترین کروموزوم‌ها در یک جمعیت جدید را کپی می‌کند. عملگر ترکیب، دو رشته یا کروموزوم را ترکیب می‌کند که این کار، جهت تولید رشته‌ها یا کروموزوم‌های بهتر انجام می‌شود. در عملگر جهش، اطلاعات جدیدی به شکل تصادفی به فرایند جستجو اضافه می‌شود که به الگوریتم ژنتیک کمک می‌کند تا در دام بهینه محلی[4] قرار نگیرد. بنابراین استفاده از عملیات ترکیب و تولید مثل در نسل‌های متوالی باعث می‌شود که جمعیت کروموزوم‌ها یا جواب‌های کاندید به همگن شدن گرایش پیدا کنند.

## ۳- روش پیشنهادی

در روش پیشنهادی، ما سعی کردیم تا با استفاده از روش جستجوی ژنتیک مقادیر بهینه‌ی پارامترهای الگوریتم ADBSCAN را بیابیم. مراحل الگوریتم پیشنهادی به شرح زیر می‌باشد:

۱) در ابتدا با استفاده از الگوریتم ژنتیک یک جمعیت اولیه از مقادیر پارامترهای الگوریتم ADBSCAN را ایجاد کردیم و مقادیر معیار ارزیابی ARI [11] را برای مجموعه داده مورد نظر بر اساس جمعیت اولیه محاسبه کردیم.

۲) حال جمعیت جدید را بر اساس پارامترهایی که برای آنها بهترین مقادیر ARIبدست آمده بود انتخاب کردیم.
۳) در ادامه عملیات ترکیب را روی جمعیت فعلی به این صورت اعمال کردیم که برای مقادیر هر دو پارامتر الگوریتم ADBSCAN، از هر دو مقدار موجود در جمعیت میانگین گرفتیم و جمعیت جدیدی را تولید کردیم و مجدد مقادیر معیار ارزیابی ARI را بر اساس جمعیت جدید محاسبه کردیم و مرحله ۲ را تکرار کردیم.
۴) عملیات جهش را روی جمعیت جدید فعلی اعمال کردیم و با یک احتمال از مقادیر جمعیت فعلی یک بیت کم کردیم و جمعیت جدیدی را تولید کردیم و مقادیر معیار ارزیابی ARI را برای این جمعیت جدید محاسبه کردیم.
۵) این مراحل را ادامه دادیم تا جایی که در نهایت به بهترین مقدار ARI در تمام این جمعیت‌ها رسیدیم.
با استفاده از الگوریتم پیشنهادی دیدیم که این روش موجب بهبود دقت الگوریتم خوشه‌بندی ADBSCAN می‌شود.

## ۴- نتایج و ارزیابی

برای ارزیابی روش پیشنهادی ما چندین مجموعه داده‌ی دنیای واقعی را مورد ارزیابی قرار دادیم که از پایگاه دادگان UCI[5] [12] و مؤسسه‌ی امنیت سایبری کانادا بدست آمده است.

آزمون‌ها در پایتون و با استفاده از scikit-learn [13] و کتابخانه‌ی numpy پیاده‌سازی و اجرا شده‌اند. جدول (۱) جزئیات مجموعه داده‌های دنیای واقعی را نشان می‌دهد که در آزمایشات استفاده می‌شود و مجموعه داده‌ها به شرح زیر است:

Wine، Seeds، Iris، HTRU2، Banknote (مجموعه داده‌های احراز هویت اسکناس)، Ecoli و Leaf.

در روش پیشنهادی تعداد جمعیت اولیه را برای مجموعه داده banknoteو HTRU2 به ترتیب برابر با ۱۰۰۰ و ۵۰۰ و برای سایر مجموعه داده‌ها برابر با ۳۰۰۰ در نظر گرفتیم. نتایج بدست آمده از روش پیشنهادی و میزان بهبود آن نسبت به اجرای الگوریتم ADBSCAN، در جدول (۲) آمده است.

از نتایج بدست آمده درمیابیم که روش پیشنهادی نسبت به الگوریتم ADBSCAN در پنج از هفت مجموعه داده عملکرد بهتری داشته است که میزان بهبود آن در مجموعه داده Iris و Seeds درصد قابل توجهی می‌باشد.

در جدول (۳) زمان های اجرا برای دو روش فوق گزارش شده است.

در جدول (۴) عملکرد روش پیشنهادی را با الگوریتم‌های خوشه‌بندی DBSCAN(DBS) [14]، HDBSCAN(HDBS) [15] DP [16] و SNN [17] روی مجموعه داده دنیای واقعی مورد ارزیابی و مقایسه قرار دادیم. همانگونه که قابل مشاهده است روش پیشنهادی در چهار مجموعه داده بسیار بهتر از سایر الگوریتم‌ها عمل کرده است.

در شکل (۱) نیز میزان بهبود معیار ارزیابی ARI در روش پیشنهادی نسبت به الگوریتم ADBSCAN برای مجموعه داده‌های Iris، Ecoli، HTRU2، Leaf و Seeds نمایش داده شده است. همانطور که ملاحظه می‌کنید روش پیشنهادی مقادیر مختلف پارامترهای $k$ و $noise\_percent$ را به ترتیب در بازه‌های [۱۴۰، ۱] و [۱، ۰] مورد بررسی قرار می‌دهد و بهترین آن‌ها را تعیین می‌کند.

## ۵- جمع‌بندی

در این تحقیق با کمک الگوریتم بهینه‌سازی فرابتکاری ژنتیک مقادیر مختلف را برای پارامترهای الگوریتم ADBSCAN مورد بررسی و آزمایش قرار دادیم. سپس مقادیر بهینه‌ی پارامترها را بدست آوردیم و با استفاده از معیار ARI به ارزیابی نتایج پرداختیم. در نهایت توانستیم بهترین مقادیر را برای پارامترهای $k$ و $noise\_percent$ بدست آوریم که در نتیجه بیشترین میزان ARI برای آن مقادیر بدست آمد.

با به کارگیری روش پیشنهادی، به صورت متوسط، ۱۱/۴۶ درصد بهبود در معیار ARI حاصل شده است. بنابراین، از نتایج بدست آمده به این مهم رسیدیم که روش پیشنهادی کارا بوده و موجب بهبود عملکرد الگوریتم خوشه‌بندی ADBSCAN می‌شود.

در آینده تمایل داریم تا از روش پیشنهادی در کاربردهای پزشکی مانند قطعه‌بندی تصاویر پزشکی استفاده کنیم.

**جدول (۱) : جزئیات مجموعه داده دنیای واقعی**

| Dataset | Samples | Dimensions | Clusters |
|---|---|---|---|
| Wine | 178 | 13 | 3 |
| Iris | 150 | 4 | 3 |
| HTRU2 | 17898 | 8 | 2 |
| Seeds | 210 | 7 | 3 |
| Banknote | 1372 | 4 | 2 |
| Ecoli | 336 | 7 | 8 |
| Leaf | 340 | 15 | 30 |

**جدول (۲) : نتایج روش پیشنهادی روی مجموعه داده دنیای واقعی**

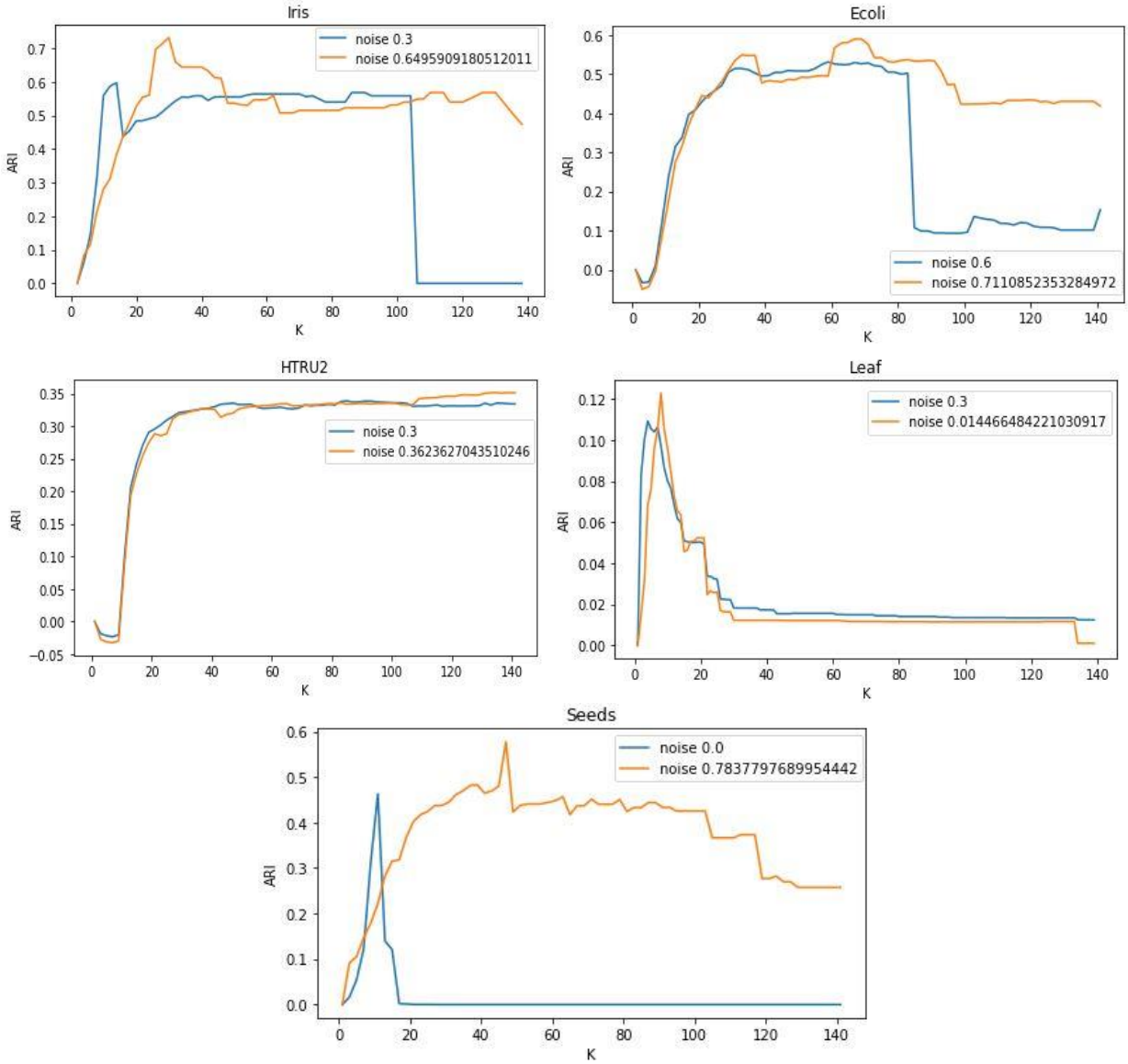
| Dataset | | ADBSCAN | ADBSCAN with Genetic | Percentage Improvement |
|---|---|---|---|---|
| Iris | ARI | 0.568 | 0.731 | 28.69 |
| Wine | ARI | 0.426 | 0.426 | 0 |
| Ecoli | ARI | 0.528 | 0.589 | 11.55 |
| HTRU2 | ARI | 0.331 | 0.350 | 5.74 |
| Leaf | ARI | 0.106 | 0.122 | 15.09 |
| Seeds | ARI | 0.484 | 0.577 | 19.21 |
| Banknote | ARI | 0.757 | 0.757 | 0 |

**جدول (۳) : گزارش زمان های اجرا**

| Dataset | ADBSCAN | ADBSCAN with Genetic |
|---|---|---|
| Iris | 163 s | 329 s |
| Wine | 204 s | 217 s |
| Ecoli | 226 s | 457 s |
| HTRU2 | 4486 s | 4118 s |
| Leaf | 223 s | 1104 s |
| Seeds | 170 s | 233 s |
| Banknote | 417 s | 1102 s |

جدول (٤) : مقایسه عملکرد روش پیشنهادی با تعدادی الگوریتم خوشه‌بندی روی مجموعه داده دنیای واقعی

| Dataset | | DBS | DP | SNN | HDBS | ADBSCAN with Genetic |
|---|---|---|---|---|---|---|
| Iris | ARI | 0.568 | 0.568 | 0.568 | 0.565 | **0.731** |
| Wine | ARI | 0.292 | 0.285 | 0.278 | 0.275 | **0.426** |
| Ecoli | ARI | 0.480 | 0.375 | **0.707** | 0.398 | 0.589 |
| HTRU2 | ARI | **0.375** | - | 0.071 | 0.298 | 0.350 |
| Leaf | ARI | 0.039 | 0.044 | 0.044 | 0.017 | **0.122** |
| Seeds | ARI | 0.447 | **0.712** | 0.622 | 0.344 | 0.577 |
| Banknote | ARI | 0.174 | 0.398 | 0.525 | 0.466 | **0.757** |




شکل (۱) : مقادیر ARI به عنوان تابعی از $k$ $(1 < k < 140)$

## زیرنویس ها

[1] Non-dominated Sorting Genetic Algorithm II
[2] Selection
[3] Mutation
[4] Local Optimum
[5] UCI Machine Learning Repository

## مراجع

[1] Ting Liu, Charles Rosenberg, and Henry A Rowley. 2007. *Clustering billions of images with large scale nearest neighbor search*. In 2007 IEEE workshop on applications of computer vision (WACV'07), 28.
[2] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. 2013. *Recommender systems survey*. *Knowledge-based Syst.* 46, (2013), 109–132.
[3] Mark S Handcock, Adrian E Raftery, and Jeremy M Tantrum. 2007. *Model-based clustering for social networks*. *J. R. Stat. Soc. Ser. A (Statistics Soc.* 170, 2 (2007), 301–354.
[4] Richard O Duda, Peter E Hart, and David G Stork. 1973. *Pattern classification and scene analysis*. Wiley New York.
[5] Edward W Forgy. 1965. *Cluster analysis of multivariate data: efficiency versus interpretability of classifications*. Biometrics 21, (1965), 768–769.
[6] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, and others. 1996. *A density-based algorithm for discovering clusters in large spatial databases with noise*. In Kdd, 226–231.
[7] K Aparna and Mydhili K Nair. 2014. *Enhancement of K-Means algorithm using ACO as an optimization technique on high dimensional data*. In 2014 International Conference on Electronics and Communication Systems (ICECS), 1–5.
[8] Elham Azhir, Nima Jafari Navimipour, Mehdi Hosseinzadeh, Arash Sharifi, and Aso Darwesh. 2021. *An efficient automated incremental density-based algorithm for clustering and classification*. Futur. Gener. Comput. Syst. 114, (2021), 665–678.
[9] Melanie Mitchell. 1998. An introduction to genetic algorithms. MIT press.
[10] Hao Li, Xiaojie Liu, Tao Li, and Rundong Gan. 2020. *A novel density-based clustering algorithm using nearest neighbor graph*. Pattern Recognit. 102, (2020), 107206..
[11] Lawrence Hubert and Phipps Arabie. 1985. *Comparing clusterings*. J. Classif. 2, 193–218 (1985), 24.
[12] Arthur Asuncion. 2007. Uci machine learning repository, university of california, irvine, school of information and computer sciences. http//www. ics. uci. edu/~ mlearn/MLRepository. html (2007).
[13] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and others. 2011. *Scikit-learn: Machine learning in Python*. J. Mach. Learn. Res. 12, (2011), 2825–2830.
[14] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, and others. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In Kdd, 226–231.
[15] Ricardo J G B Campello, Davoud Moulavi, and Jörg Sander. 2013. Density-based clustering based on hierarchical density estimates. In Pacific-Asia conference on knowledge discovery and data mining, 160–172.
[16] Alex Rodriguez and Alessandro Laio. 2014. Clustering by fast search and find of density peaks. Science (80-. ). 344, 6191 (2014), 1492–1496.
[17] Levent Ertöz, Michael Steinbach, and Vipin Kumar. 2003. Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data. In Proceedings of the 2003 SIAM international conference on data mining, 47–58.