Journal of Soft Computing and Information Technology (JSCIT)
Babol Noshirvani University of Technology, Babol, Iran
Journal Homepage: jscit.nit.ac.ir
Volume 10, Number 1, Spring 2021, pp. 52-62
Received: 07/23/2019, Revised: 04/26/2020, Accepted: 01/21/2021

Improving the Performance of the k-Nearest Neighbors Algorithm with Utilization of the PSO Metaheuristic Algorithm

Ahmad Hajimoradi [1], Aireza Naser Sadrabadi [*2], Seyed Mahmood Zanjirchi [3] and Habib Zare Ahmad Abadi [4]
1- PhD Student, Department of Industrial Management, Faculty of Economics, Management and Accounting, Yazd University, Yazd, Iran.
2,3,4- Department of Industrial Management, Faculty of Economics, Management and Accounting, Yazd University,Yazd, Iran.
[1]Ahajimoradi@stu.yazd.ac.ir, [2]Alireza_Naser@yazd.ac.ir, [3]Zanjirchi @ yazd.ac.ir, [4]Zarehabib @ yazd.ac.ir

*Corresponding author address: Aireza Naser Sadrabadi, Faculty of Economics, Management and Accounting, Yazd, Iran, Post Code: 89158_18411.

Abstract- The k-nearest neighbor's algorithm (KNN) is one of the most widely used and useful nonparametric classification algorithms. The classification mechanism of this algorithm involves computing the distance between new instances and the instances whole classes are known. When the dataset contains non-numerical (ordinal and nominal) attributes, the performance of the algorithm can be significantly affected by how this distance is measured. In this paper, we attempt to improve the performance of the KNN algorithm by presenting a new solution for computing the distance of non-numerical traits. For this purpose, the Particle Swarm Optimization (PSO) algorithm is used. The task of this algorithm is to determine the best value of the distance between two states in a non-integer trait so that the accuracy of the KNN algorithm is increased. UCI University Learning Repository Data is used to test this idea. The results obtained from the proposed algorithm are compared with several other improved algorithms and show the useful improvement of this mechanism.

Keywords- k-nearest neighbors, dynamic distance, practical swarm optimization, non-numerical attributes (nominal and ordinal).

مجله علمی پژوهشی رایانش نرم و فناوری اطلاعات
دانشگاه صنعتی نوشیروانی بابل
صفحه مجله: jscit.nit.ac.ir
جلد ۱۰، شماره ۱، بهار ۱۴۰۰ ، صفحه ۵۲-۶۲
دریافت: ۱۳۹۸/۰۵/۰۱، بازنگری: ۱۳۹۹/۰۲/۰۷، پذیرش: ۱۳۹۹/۱۱/۰۲

بهبود عملکرد الگوریتم KNN با استفاده از الگوریتم فرا ابتکاری PSO

احمد حاجی مرادی[۱] ، علیرضا ناصر صدرآبادی*[۲]، سید محمود زنجیرچی[۳] ، حبیب زارع احمدآبادی[۴]
۱- دانشجوی دکتری، دانشکده اقتصاد، مدیریت و حسابداری، دانشگاه یزد، یزد، ایران.
۲ و ۳ و ۴- دانشکده اقتصاد، مدیریت و حسابداری، دانشگاه یزد، یزد، ایران.
[1]Ahajimoradi@stu.yazd.ac.ir , [2]Alireza_Naser@yazd.ac.ir, [3]Zanjirchi @ yazd.ac.ir, [4]Zarehabib @ yazd.ac.ir

* نشانی نویسنده مسئول: علیرضا ناصر صدرآبادی، یزد، بلوار دانش، دانشگاه یزد، دانشکده اقتصاد، مدیریت و حسابداری ، کد پستی: ۸۹۱۵۸ – ۱۸۴۱۱

چکیده- الگوریتم KNN یکی از مهم‌ترین الگوریتم‌های نا پارامتری د سته‌بندی ا ست و جزء روش‌های اثربخش د سته‌بندی مح سوب می شود. سازوکار این الگوریتم برای تعیین د سته نمونه جدید، مبتنی بر محا سبه فا صله نمونه جدید تا سایر نمونه‌ها ست. زمانی که پایگاه داده شامل صفات غیر عددی (رتبه‌ای و ا سمی) با شد، نحوه محا سبه فا صله می‌تواند بر کارآیی الگوریتم اثرگذار با شد. در این مقاله تلاش شده ا ست با ارائه یک راه حل جدید برای محا سبه فا صله صفات غیرعددی عملکرد الگوریتم KNN بهبود یابد. برای این منظور از الگوریتم بهینه سازی انبوه ذرات (PSO) استفاده شده است. وظیفه این الگوریتم در مساله، تعیین بهترین مقدار فاصله بین دو وضعیت در یک صفت غیرعددی است به نحوی که میزان صحت الگوریتم KNN افزایش یابد. برای آزمایش این ایده از داده‌های مخزن یادگیری ما شین دان شگاه UCI ا ستفاده شده ا ست. نتایج بد ست آمده از مقای سه الگوریتم پی شنهادی با چند الگوریتم بهبود یافته، حاکی از اثربخشی قابل قبول الگوریتم پیشنهادی است.

واژه‌های کلیدی: K نزدیک‌ترین همسایه، فاصله پویا، الگوریتم بهینه‌سازی انبوه ذرات، صفات غیر عددی (اسمی و رتبه‌ای).

۱- مقدمه

الگوریتم K نزدیک‌ترین همسایه[۱] (KNN) به‌عنوان یکی از ده تکنیک برتر داده‌کاوی شناخته‌شده است[۱]. ایده اولیه این الگوریتم اولین بار در گزارش نیروی هوایی ایالات متحده توسط فیکس و هوگس به‌عنوان تکنیک تفکیک ناپارامتری برای طبقه‌بندی جمعیت‌هایی که توزیع آن‌ها ناشناخته بود، ارائه شد[۲]. بعدها توسط کاور و هات شکل اولیه الگوریتم اولیه KNN معرفی شد و در ۴ دهه بعد گسترش یافت[۳]. الگوریتم KNN یکی از ساده‌ترین ایده‌های دسته‌بندی را مورداستفاده قرار می‌دهد و درعین‌حال خودش را به‌عنوان جالترین و مؤثرترین الگوریتم‌های داده‌کاوی معرفی می‌کند[۴]. این الگوریتم نمونه‌ای از الگوریتم‌های مبتنی بر حافظه است که در آن مجموعه داده‌های آموزشی در حافظه ذخیره می‌شوند و برای پیش‌بینی دسته یک نمونه جدید، می‌توان به‌سادگی و از طریق مقایسه آن با رکوردهای مشابه در مجموعه آموزشی، دسته آن را پیش‌بینی نمود[۵]. در الگوریتم KNN ابتدا فاصله نمونه جدید با مجموعه نمونه‌های آموزشی محاسبه می‌شود. سپس با در نظر گرفتن K مورد از نزدیک‌ترین همسایه‌های نمونه جدید، دسته مربوط به این نمونه پیش‌بینی می‌شود[۶]. شبه کد این الگوریتم به در ادامه آمده است. به طور کلی عملکرد الگوریتم KNN به سه عامل بستگی دارد: اندازه نمونه، انتخاب معیار فاصله و تعیین ارزش K. انتخاب معیار مناسب فاصله به شدت روی میزان صحت الگوریتم KNN تاثیر می‌گذارد[۸]. اندازه گیری فاصله بین دو نقطه داده از الزامات اصلی برای الگوریتم KNN است[۹]. عموماً اندازه‌گیری فاصله در یک پایگاه داده با صفات عددی توسط توابع اقلیدسی یا

منهتن انجام می‌گیرد[۱۰]. اما یک پایگاه داده می‌تواند شامل داده‌هایی با چند صفت غیرعددی باشد. این امر تعاملات پیچیده تری را در مواجه با این صفات بوجود می‌آورد[۱۱].

شبه کد ۱: شبه کد الگوریتم KNN [۷]

Input:
Tr: *the set of training object*
T_{new}: *the test object*
L: *the set of classes used to label the objects*
LTr_j: *the label of training objects* Tr_j
Output: $LTNew \in L$, *the class label of* T_{new}
foreach *object* $Tr_j \in Tr$ ***do***
Compute $d(T_{new}, Tr_j)$, *the distance between* T_{new} *and* Tr_j;
End
Select $KNN \subseteq Tr$, *the set*$(neighborhood)$ *of* K *closest training objectsw*

$$LTnew = argmax \sum_{Tr_j \in KNN} I\left(l = class(LTr_j)\right) \qquad \forall l \in L$$

where $I(.)$*is an indicator function that returns the value* 1 *if its argument is true and* 0 *otherwise.*

مفهوم فاصله برای داده‌های غیرعددی به سادگی داده‌های عددی نیست و یک چالش بزرگ است. ارزشهای مختلف یک صفت غیر عددی ذاتاً منظم نیست، از این رو مقایسه مستقیم بین دو مقدار امکان‌پذیر نیست[۱۲]. سازوکار محاسبه فاصله در مواجه با صفات غیرعددی بسیار مهم است. اگرچه روشهای موجود توانایی بهبود الگوریتم KNN را از زوایای مختلف دارند اما آنها هنوز در مواجه با داده‌های غیرعددی، نمی‌توانند عملکرد مشابه داده‌های عددی داشته باشند[۱۳]. در پژوهش حاضر از یک ایده جدید به منظور بهبود عملکرد تابع فاصله در مواجه با صفات غیرعددی استفاده شده است. هدف آن است که با استفاده از الگوریتم‌های فرا ابتکاری، فاصله‌ای پویا برای مقادیر صفات غیر عددی ارائه شود تا صحت پیش‌بینی این الگوریتم افزایش یابد. ساختار مقاله در ادامه به‌این‌ترتیب است؛ بخش دوم به بیان مسئله اختصاص یافته است. در بخش سوم پیشینه تحقیق مورد بررسی قرارگرفته است. بخش چهارم به انواع داده‌ها و نحوه محاسبه فاصله اختصاص دارد و در بخش پنجم الگوریتم فراابتکاری بهینه‌سازی انبوه ذرات معرفی می‌شود. بخش ششم الگوریتم پیشنهادی مطرح می‌شود. در بخش هفتم عملکرد الگوریتم پیشنهادی با توجه به سنجه‌های مختلف با الگوریتم KNN کلاسیک و چند الگوریتم بهبود یافته دیگر مورد مقایسه قرار می‌گیرد و در بخش آخر به جمع‌بندی مطالب و نتیجه گیری پرداخته می‌شود.

۲- بیان مسئله

الگوریتم KNN از الگوریتم‌های کاربردی در مسائل دسته بندی است که تاکنون موفقیت‌های زیادی را در محیط‌های عددی بدست آورده است. اما به کارگیری این الگوریتم در پایگاه داده‌هایی با ارزش‌های غیرعددی به دلیل نبودن معیار مسافت غیرعددی برای ارزیابی روابط بین داده‌ها یک چالش بزرگ است. امروزه جمع آوری داده‌ها پیچیده‌تر شده است و داده‌های غیرعددی(اسمی و طبقه‌ای) به طور گسترده‌ای در حوزه‌هایی مانند تحلیل DNA، سیستم‌های پیشنهاد دهنده[۲]،پزشکی، صنعت بیمه و سیستم‌های خبره مورد استفاده قرار می گیرند. برای مثال در حوزه پزشکی معمولاً داده‌های یک بیمار حاوی چندین صفت اسمی و رتبه‌ای است که بیمار را توصیف می‌کند. در صنعت بیمه از صفاتی مثل نوع ماشین، سطح آموزش رانندگی، آب و هوا و... برای دسته بندی سطح مهارت رانندگان استفاده می شود. نکته مهم آنست که درک مقادیر غیرعددی برای انسان راحت است. در حالی که معنا و درکی که انسانها به راحتی از داده‌های غیرعددی بدست می‌آورند، برای اکثر الگوریتم‌های یادگیری ماشین بسیار دشوار است. بنابراین کم کردن درک فاصله بین الگوریتم‌ها و انسان، به یکی از فاکتورهای مهم برای بهبود عملکرد الگوریتم‌های دسته بندی در محیط‌های داده‌ای غیرعددی تبدیل شده است.

برای حل مساله دسته بندی در داده‌های غیرعددی، راهبرد معمول تبدیل داده‌های غیرعددی به ارزش‌های عددی و پردازش آن با استفاده از الگوریتم‌های یادگیری ماشین است. برای مثال در بیوانفورماتیک، نوکلئوتیدهای تشکیل دهنده یک زنجیره DNA هرکدام در ۴دسته اسمی A، G، T و C کد گذاری شده اند. با توجه به آنکه تعیین میزان فاصله بین وضعیتهای متفاوت کار دشواری است، به‌طورمعمول فاصله بین مقادیر در وضعیت‌های مختلف مانند این صفت اسمی، عدد یک نظر گرفته می‌شود. در حالیکه اختلاف بین نوکلئوتیدهای A وT با A و C می تواند اعداد مختلفی باشد. چنین داده‌هایی با این ویژگی ها، مانع اصلی استفاده از روشهای دسته بندی مانند KNN است که بر مبنای داده‌های عددی طراحی شده‌اند. جدول زیر یک مثال دیگر از داده‌های غیرعددی است.

جدول۱-یک نمونه از یک پایگاه داده

دسته	سرگرمی	تحصیلات	رشته	نام
C_1	موسیقی	دکتری	ریاضی	رضا
C_1	ورزش	کارشناسی ارشد	ریاضی	علی
C_1	مسافرت	کارشناسی	اقتصاد	حسین
C_2	موسیقی	کارشناسی	فلسفه	احمد
C_2	مسافرت	کارشناسی ارشد	کامپیوتر	امیر
C_2	ورزش	کارشناسی	مدیریت	محمد

در صفت رتبه‌ای تحصیلات در جدول ۱، اگر عدد یک را برای کارشناسی، عدد دو را برای کارشناسی ارشد و عدد سه را برای دکتری اختصاص دهیم، ماتریس زیر مقدار فاصله بین هر دو زوج صفت مدرک تحصیلی را نشان می‌دهد.

دکتری کارشناسی ارشد کارشناسی

$$\begin{matrix} \\ \text{کارشناسی} \\ \text{کارشناسی ارشد} \\ \text{دکتری} \end{matrix}\begin{bmatrix} ۰ & ۱ & ۲ \\ ۱ & ۰ & ۱ \\ ۲ & ۱ & ۰ \end{bmatrix} \quad (۱)$$

نکته قابل‌توجه در برابر دانستن فاصله بین مدارک تحصیلی پیاپی است. درحالی‌که برای اثربخش بودن الگوریتم‌های داده کاوی ممکن است فاصله بین مدرک کارشناسی و مدرک کارشناسی ارشد با فاصله مدرک کارشناسی ارشد از مدرک دکتری برابر نباشد. بنابراین زمانی که الگوریتم KNN داده‌های غیرعددی را پردازش می کند، مشکل اساسی چگونگی استخراج اطلاعات نهفته از ارزشهای غیرعددی است. به عبارت دیگر، بین ارزش‌های غیرعددی از صفات مختلف روابط و وابستگی مختلفی وجود دارد. از این رو سوال مهم آن است که نحوه نمایش و محاسبه فاصله بین داده‌های غیرعددی چگونه است؟ این چالش ها شرایط جدیدی را برای بهبود اندازه گیری فاصله برای داده‌های غیرعددی بوجود می آورد. به طور کلی بین داده ها روابط درهم تنیده‌ای وجود دارد که برای کشف آنها نیاز به الگوریتمی هوشمند است.

ایده زیربنایی این پژوهش، معرفی روشی جدید برای محاسبه فاصله بین مقادیر مختلف یک صفت غیر عددی است. برای افزایش قدرت تشخیص فاصله در این وضعیت از سازوکار یادگیری هوشمند الگوریتم بهینه‌سازی انبوه ذرات استفاده می‌شود. تعیین مقدار فاصله را می‌توان به یک مسئله بهینه‌سازی تبدیل نمود. در این مسئله بهینه‌سازی، تابع هدف با حداکثر کردن میزان صحت الگوریتم KNN تعریف می‌شود. با توجه به آنکه به ازای هر صفت غیر عددی یک ماتریس فاصله بهینه ایجاد می‌شود، متغیرهای تصمیم مسئله همان درآیه‌های ماتریس‌های فاصله صفات غیر عددی هستند. در این پژوهش هدف آن است که با استفاده از الگوریتم بهینه‌سازی انبوه ذرات، مقادیر این درآیه‌ها به نحوی مشخص شود که بتوان فاصله‌ای پویا را ایجاد نمود که بالاترین صحت الگوریتم KNN را به دست دهد. برای بیان ریاضی مساله از نمادهای جدول ۲ استفاده شده است:

جدول۲-تعریف نمادهای ریاضی بیان مساله

شرح	نماد	نماد شمارنده یا عضو
تعداد صفات عددی	M	m
مقدار صفت عددی m در نمونه k	A_{km}	
تعداد صفات اسمی	N	n
مقدار صفت غیرعددی n در نمونهk	B_{kn}	
مجموعه مقادیری که صفت غیرعددی n می تواند اختیار کند.	X_n	
عنصر شماره i از مجموعه آزمایش	T_i	
عنصر شماره j از مجموعه آموزش	Tr_j	
مجموعه طبقه های متغیر هدف	L	l
برچسب نمونه آزمایش i	LT_i	
برچسب نمونه آموزش j	LTr_j	
k نزدیک ترین همسایه نمونه آموزش i در مجموعه آزمایش	KNN_i	
تابع مشخصه به شرح توصیف شده در شبه کد شماره ۱	I(.)	
فاصله بین مقدار B و 'B در صفت غیرعددیn	$R_{(B,B')}$	متغیر تصمیم
برچسب پیش بینی شده نمونه آزمایش i	PT_i	متغیر تصمیم
فاصله بین نمونه آزمایش i و نمونه آموزش j	$d(T_i,Tr_j)$	متغیر تصمیم
متغیر صفر و یک نشان دهنده صحت پیش بینی برچسب نمونه آزمایش i	X_i	متغیر تصمیم

با توجه به نمادگذاری فوق، نمونه i از مجموعه آموزش (T_i) و نمونه j از مجموعه آزمایش (Tr_j) دارای ساختار برداری زیر هستند:

جدول۳-تعریف ساختار برداری صفات در داده‌های آموزش و آزمایش

نمونه	صفات غیرعددی				صفات عددی			
T_i	B_{i1}	B_{i2}	...	B_{iN}	A_{i1}	A_{i2}	...	A_{iM}
Tr_j	B_{j1}	B_{j2}	...	B_{jN}	A_{j1}	A_{j2}	...	A_{jM}

روابط ۳،۲ و ۴ نحوه محاسبه متغیرهای تصمیم محاسبه شونده و رابطه ۵ نیز مدل ریاضی مساله بهینه سازی را نشان می‌دهد.

$$X_i = \begin{cases} 0 & PT_i \neq LT_i \\ 1 & PT_i = LT_i \end{cases} \quad (۲)$$

$$d(T_i, Tr_j) = \sqrt{\sum_m (A_{im} - A_{jm})^2 + \sum_n R^2_{(B_{in}, B_{jn})}} \quad (۳)$$

$$PT_i = argmax \sum_{Tr_j \in kNN} I(l = LTrj) \quad \forall l \in L \quad (۴)$$

$$Max \quad z = \sum_i X_i$$

$$s.t \quad (۵)$$

$$R_{(B_{in}, B_{jn}) \geq 0} \quad \forall_n$$

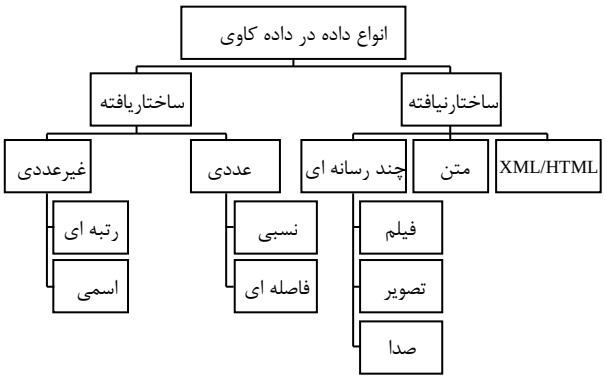
$$R_{(B_{in}, B_{jn}) \leq 1} \quad \forall_n$$

۳- پیشینه پژوهش

لئو و همکاران معتقدند معیار همپوشانی و انواع آن نمی تواند روابط درون صفات غیرعددی را اندازه گیری کند. آنها برای این چالش یک معیار اندازه گیری جدید با نام معیار سلسه مراتبی ارزش واقعی را ابداع کرده‌اند که قادر است روابط درون داده‌های غیرعددی را اندازه گیری کند و عملکرد الگوریتم KNN را بهبود دهد[۱۴]. چن و همکارش با تعریف یک مجموعه از توابع فاصله وزنی برای صفات رتبه‌ای و با بکارگیری آنها عملکرد الگوریتم KNN را ارتقاء داده اند[۱۵]. دومنیکونی و همکاران در تحقیق خود با استفاده از معیار کای دو برای تعیین فاصله و اعمال ضریب به عنوان وزن به آن سعی کرده‌اند که در انتخاب همسایه‌ها، نمونه‌های که قدرت بیشتری در تطابق با نمونه‌های جدید دارند وزن بیشتری بگیرند[۱۶]. شامسول و همکاران به‌منظور تجزیه‌وتحلیل مشکلات روش KNN، به معرفی الگوریتم K نزدیک‌ترین همسایه پویا[3] (DKNN) می‌پردازند[۱۷]. هوک و مارتینز برای وزن دهی به صفات یک برنامه‌ریزی خطی پیشنهاد دادند که در مقایسه با سایر روش‌های وزن دهی به صفات، در پیمایش اولیه ابعاد داده‌ها، عملکرد بهتری داشت[۱۸]. رافیل حسن و همکاران یک روش جدید برای تعیین تابع فاصله مناسب برای KNN ارائه داده است. در این روش سطح زیر منحنی(ROC[4]) که یک روش شناخته‌شده برای اندازه‌گیری کیفیت طبقه‌بندهای دو دسته‌ای است، در نظر گرفته می‌شود. بر اساس ویژگی‌های ROC دریک همسایگی مناسب برای نمونه‌هایی که فاصله آنها در حال محاسبه است، این وزن‌ها برای تابع فاصله محاسبه می‌شود[۱۹]. سیالیمن و همکاران در پژوهش خود، معتقدند که در KNN تعیین دسته برای داده جدید با یک سیستم رأی اکثریت ساده اجرا می‌شود که ممکن است شباهت بیان داده‌ها را نادیده بگیرد. آن‌ها روشی را برای حل این موضوع ارائه دادند که برای محاسبه فاصله وزنی به‌کاربرده شده از ترکیب میانگین محلی مبتنی بر KNN[5] و فاصله وزنی KNN[6] استفاده می‌کند. نتایج نشان می‌دهد که ترکیب این دو می‌تواند صحت KNN را افزایش دهد[۲۰]. توتز و همکارش معتقدند که K نزدیک‌ترین همسایه یک روش دسته‌بندی است که با فرضیات ضعیف کار می‌کند. آن‌ها دو تغییر برای بهبود عملکرد الگوریتم ارائه داده‌اند. اول، بجای به‌کارگیری وزن‌ها که به‌تنهایی در محاسبه فاصله استفاده می‌شود، وزن‌ها را با یک مدل منطقی تخمین زدند. با به‌کارگیری روشی مثل Lasso یا Boosting نزدیک‌ترین همسایه‌های مرتبط انتخاب می‌شود. گام دوم مبتنی بر تخمین و انتخاب است، آن‌ها فضای پیش‌بینی کننده‌ها را گسترش دادند. پیش‌بینی کننده‌ها و تعداد نزدیک‌ترین همسایه به‌عنوان ابعاد فرعی فضای پیش‌بینی کننده استفاده می‌شوند[۲۱]. گویو و همکاران در پژوهش خود با معرفی میانگین توزیع‌شده چندگانه فاصله‌ها و میانگین توزیع‌شده فاصله مبتنی بر ویژگی‌های عمومی میانگین ،یک میانگین فاصله توزیع شده دسته‌بند KNN را پیشنهاد داده‌اند. در روش پیشنهادی متوسط بردارهای چندگانه محلی از نمونه داده‌شده در هر دسته با در نظر گرفتن نزدیک‌ترین همسایه دسته مربوط به خود محاسبه می‌شود. با استفاده از متوسط بردارهای محلی، میانگین فاصله‌های متناظر K محاسبه می‌شود و سپس برای طراحی متوسط فاصله کلی مورداستفاده قرار می‌گیرد. در مرحله دسته‌بندی، متوسط فاصله کلی به‌عنوان قاعده تصمیم‌گیری برای دسته‌بندی مورداستفاده قرار می‌گیرد و نمونه با حداقل فاصله کلی بین همه دسته‌ها دسته‌بندی می‌شود[۲۲].

۴- انواع داده‌ها و محاسبه فاصله

داده یک مجموعه از حقایق است که به‌عنوان نتایج آزمایش‌ها، مشاهدات یا تجربیات به دست می‌آید. داده‌ها می‌توانند شامل اعداد، حروف، لغات، تصاویر، صداهای ضبط‌شده و سایر متغیرها باشند. داده اغلب به‌عنوان پایین‌ترین سطح انتزاع است که از آن اطلاعات و سپس دانش ناشی شده‌است. در بالاترین سطح انتزاع، داده‌ها می‌توانند به عنوان ساختاریافته و ساختار نیافته طبقه‌بندی شوند. داده ساختار نیافته از هر ترکیبی از متن، تصویر، صدا و محتوای وب تشکیل شده است. داده ساختاریافته، می‌تواند به داده عددی و غیر عددی (اسمی و رتبه‌ای) تقسیم شوند[۲۳].



شکل ۱-طبقه‌بندی داده‌ها درداده کاوی

در مبحث داده کاوی فاصله بین نقاط داده در فضای n بعدی یک مفهوم زیربنایی است و به شکل گسترده‌ای مورد استفاده قرار می‌گیرد. به همین دلیل در ادامه به نحوه محاسبه بین دو نقطه داده در فضای n بعدی اشاره می شود.

اگر دو نقطه داده X و Y را در فضای n بعدی در نظر گرفته شود:

$$X = (x_1, x_2, \cdots, x_n) \qquad (۶)$$

$$Y = (y_1, y_2, \cdots, y_n) \quad (۷)$$

تابع $d(X,Y)$ یک تابع محاسبه فاصله بین دو نقطه است اگر شرایط ذکر شده در روابط ۸ تا ۱۰ را داشته باشد[۲۴]:

$$d(X,Y) \geq 0 \quad (۸)$$

$$= d(Y,X)d(X,Y) \quad (۹)$$

$$=0 \Leftrightarrow X = Yd(X,Y) \quad (۱۰)$$

توابع متنوعی برای محاسبه فاصله در فضای n بعدی وجود دارد که شرایط فوق را دارد. یکی از آن‌ها تابع مینکوفسکی[۷](رابطه۱۱) است. در این تابع اگر $q = ۱$ باشد فاصله منهتن و اگر $q = ۲$ باشد فاصله اقلیدسی محاسبه می‌شود. فاصله اقلیدسی یکی از رایج‌ترین و محبوب‌ترین توابع فاصله است که به‌سادگی فاصله دو نمونه را در فضای n بعدی محاسبه می‌کند[۱۴].

$$d(X_i, Y_j) = \sqrt[q]{|x_{i1} - x_{j1}|^q + \cdots + |x_{in} - x_{jn}|^q} \quad (۱۱)$$

سایر معیارهای اندازه‌گیری فاصله شامل: اقلیدسی وزنی، بارایوکرتیس[۸]، واگرایی[۹]، باتاچاریا[۱۰] و چبی چف[۱۱] هستند[۲۵]. برای استفاده از تابع فاصله اقلیدسی باید به نکات زیر در مورد صفات توجه داشت:

اگر مقیاس اندازه‌گیری برای صفات از نوع عددی باشد ابتدا داده‌ها باید نرمال‌سازی شوند. در الگوریتم KNN، تفاوت وسیع دامنه متغیرها می‌تواند منجر به تأثیرات نامطلوب بر نتایج شود. بنابراین تحلیلگران برای استفاده از مقادیر متغیرهای عددی باید آن‌ها را ابتدا نرمال‌سازی کنند[۵].

مقادیر صفات رتبه‌ای مانند مدرک تحصیلی، دارای نظم هستند اما اندازه و بزرگی اختلاف میان دو مقدار مشخص نیست. در این نوع صفات با تبدیل هر صفت به داده‌های عددی می‌توان داده‌ها را به‌صورت نقاط در یک فضا نمایش داد؛ بنابراین اگر دونقطه ازلحاظ هندسی به یکدیگر نزدیک باشد بدان معنی است که رکوردها و داده‌های متناظر آن‌ها به یکدیگر شبیه هستند. چنانچه M تعداد وضعیت‌های ممکن برای یک صفت رتبه‌ای باشد، روش رایج آن است که M حالت به‌صورت مرتب و با $۱, \ldots, M$ کدگذاری می‌شود و پس‌ازآن مانند صفات عددی نرمال و در محاسبه فاصله مورداستفاده قرار می‌گیرد.

در صفات اسمی مثل جنسیت، تأهل و ... همگی از نوعی هستند که نمی‌توان آن‌ها را به سادگی مانند قد و وزن در فضا نمایش داد. به‌طور مرسوم اگر دو نمونه در یک صفت اسمی وضعیت یکسانی داشته باشند، مقدار فاصله بین آن‌ها برابر با صفر و در غیر این صورت مقدار فاصله برابر با یک در نظر گرفته می‌شود[۲۶]. درحالی‌که می‌توان برای نشان دادن اهمیت برخی از تفاوت‌ها بین دو نمونه در یک صفت اسمی، از سایر مقادیر بزرگتر از صفر نیز استفاده نمود. اگر فرض شود p تعداد وضعیت یک صفت اسمی باشد و l_{km} فاصله بین دو وضعیت m و k را نشان دهد می‌توان فاصله بین دو مقدار را در یک ماتریس متقارن که در آن $l_{km} = l_{mk}$ و $l_{kk} = 0$ و $l_{km} > 0$ است، سازماندهی نمود[۲۷].

$$L = \begin{bmatrix} 0 & \cdots & l_{1p} \\ \vdots & \ddots & \vdots \\ l_{p1} & \cdots & 0 \end{bmatrix} \quad (۱۲)$$

با توجه به مفهوم کلیدی محاسبه فاصله در الگوریتم KNN، در این پژوهش برای تعیین مقدار فاصله در صفات غیرعددی از الگوریتم بهینه‌سازی انبوه ذرات استفاده می‌شود.

۵- الگوریتم بهینه‌سازی انبوه ذرات[۱۲]

الگوریتم بهینه‌سازی انبوه ذرات (PSO) توسط کندی و ابرهات [۲۸] برای اولین بار ارائه شد. این الگوریتم از رفتار پروازجمعی پرندگان که در فضای چند بعدی در جستجوی مکان بهینه، حرکات و فواصل را برای جستجوی بهتر غذا تنظیم می‌کنند، الهام می‌گیرد و آن‌را شبیه‌سازی می‌کند[۲۹]. این الگوریتم با یک جمعیت از جواب‌های تصادفی مسئله که به آن‌ها ذرات گفته می‌شود، آغاز می‌شود. این ذرات در فضای حل مسئله که همان فضای جستجو است، پراکنده می‌شوند و مقدار تابع هدف هرکدام ذرات (جواب‌ها) مشخص می‌گردد. حرکت بعدی هرکدام از این ذرات در فضای جستجو با توجه به مکان فعلی ذره، بهترین موقعیتی که تاکنون توسط هر ذره یافت شده است (Pbest) و همچنین بهترین موقعیتی که تاکنون توسط همه ذرات به‌دست‌آمده است (Gbest)، مشخص می‌شود. تکرار بعدی زمانی انجام می‌شود که همه ذرات حرکت خود را انجام داده باشند. در بلندمدت این توده ذرات (جواب‌ها)، مشابه زمانی که توده‌ای از پرندگان در جستجوی غذا هستند، در فضای حل مسئله به سمت جواب بهینه نزدیک می‌شوند. رفتار ذره در این الگوریتم با استفاده از روابط زیر مشخص می‌شود:

$$V_{i,t+1} = V_{i,t} + C_1R_{1,t}(P_{pi,t} - X_{i,t}) + C_2R_{2,t}(P_{g,t} - X_{i,t}) \quad (۱۳)$$

$$X_{i,t+1} = X_{i,t} + V_{I,t+1} \quad (۱۴)$$

در اینجا$X_{i,t}$ مکان ذره i ام در تکرار t و $V_{i,t}$ نیز نرخ تغییر مکان ذره i ام در تکرار t را نشان می‌دهد. P_{it} و $P_{g,t}$ به ترتیب نشان‌دهنده Pbest ذره i ام و Gbest است. ضرایب C_1 و C_2، ضرایب شتاب[۱۳] هستند که میزان تمایل ذرات به سمت موقعیت Pbest یا Gbest را تعیین می‌کنند. $R_{1,t}$ و$R_{2,}$ نیز دو عدد تصادفی هستند که از توزیع یکنواخت بین صفر و یک در هر تکرار به کمک شبیه‌سازی تولید می‌شوند [۳۰].

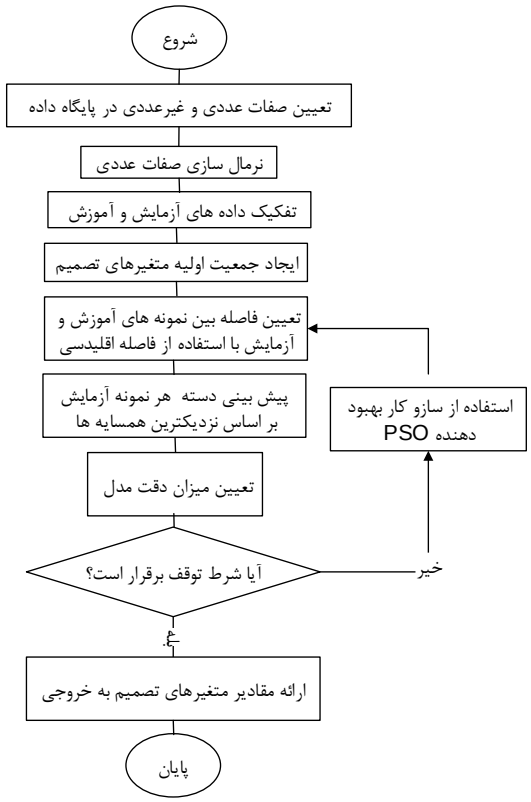
شبه کد این الگوریتم به‌طور خلاصه به شکل زیر است:

شبه کد ۲: شبه کد الگوریتم PSO [۳۱]

1. Randomly initialize positions and velocities all particles.
2. Do:
3. Set Pbest and Gbest.
4. Calculate particle velocity according to equation(13)
5. Update particle position according to equation(14)
6. Evaluate objective function value (fintness value).
7. While a satisfactory solution has been found

۶- الگوریتم پیشنهادی

در الگوریتم پیشنهادی این مقاله از سازوکار بهینه سازی الگوریتم PSO برای تعیین مقادیر مناسب متغیرهای تصمیم استفاده خواهد شد. متغیرهای اصلی تصمیم این مساله بهینه سازی همانطور که در مدل ریاضی مساله نیز به آن اشاره شد میزان فاصله بین وضعیتهای مختلف یک صفت غیرعددی است. گام های اجرای الگوریتم پیشنهادی در روندنمای شماره ۱ آمده است. همانطور که در این روندنما ملاحظه می‌شود در بخش مقدماتی صفات عددی و غیرعددی تفکیک شده و نرمال‌سازی صفات عددی انجام می‌شود. پس از آن داده‌ها به دو دسته آموزش و آزمایش تفکیک می گردند. پس از طی کردن گام‌های مقدماتی و آماده‌سازی زیرساخت‌های اجرای الگوریتم KNN مجموعه جوابهای تصادفی اولیه برای الگوریتم PSO تولید می‌شود. گام‌های تکرار شونده الگوریتم پیشنهادی از مرحله محاسبه فاصله بین داده‌های آموزش و آزمایش آغاز می‌شود. پس از محاسبه فاصله، پیش بینی دسته نمونه‌های آموزش انجام می‌شود. با مقایسه دسته پیش بینی شده با دسته واقعی نمونه‌های آزمایش، دقت مدل تعیین می‌شود. پس از تعیین دقت مدل شرط توقف بررسی می‌شود. اگر شرط توقف الگوریتم برقرار نباشد با استفاده از الگوریتم PSO مقادیر متغیرهای تصمیم بهبود یافته و این چرخه تا زمان برقراری شرط توقف ادامه می‌یابد. در پایان نیز پس از برقراری شرط توقف، مقادیر بدست آمده از الگوریتم PSO به خروجی ارسال خواهد شد.



روندنمای۱:فرآیند اجرای الگوریتم KNN-PSO

۷- آزمایش تجربی

یکی از روشهای رایج در ارزیابی کیفیت الگوریتم‌هایی که محققین مختلف در حوزه یادگیری ماشین و هوش مصنوعی ارائه می‌دهند، استفاده از داده‌های مخزن یادگیری ماشین دانشگاه UCI است. عموماً محققین مختلف الگوریتم‌های پیشنهادی خود را روی این مجموعه داده اجرا و نتایج آنرا باهم مقایسه می‌کنند. در پژوهش حاضر نیز از همین روش اعتبار سنجی استفاده شده است. الگوریتم پیشنهادی در این پژوهش نیز علاوه بر مقایسه نتایج با الگوریتم CLASSIC-KNN با دو الگوریتم لئو و همکاران[۱۴] و چن و همکارش[۱۵] که در پیشینه تحقیق به آنها اشاره شده نیز مورد مقایسه قرار می‌گیرد.

برای این منظور ۸ پایگاه داده از داده‌های مخزن یادگیری ماشین دانشگاه UCI[۱۴] برای آزمون روش پیشنهادی انتخاب‌شده است. که در جدول ۴ به آنها اشاره شده است.

جدول ۴:معرفی پایگاه داده‌ها

ردیف	نام پایگاه	تعداد نمونه	تعداد کل صفات	سهم صفات غیر عددی	تعداد دسته
۱	Statlog (German Credit Data)	1000	10	%50	2
۲	Credit Approval	653	15	%60	2
۳	Statlog (Heart)	270	13	%54	2
۴	Flags	194	28	%64	8
۵	Breast Cancer	286	9	%66	2
۶	Hayes-Roth	160	4	%100	3
۷	Tic-Tac-Toe Endgame	958	9	%100	2
۸	Car Evaluation	1729	6	%100	4

برای ارزیابی مدل از چهار سنجه صحت[۱۵]، حساسیت[۱۶] ،شفافیت[۱۷] و دقت[۱۸] ، استفاده شده است. این سنجه‌ها با استفاده از ماتریس خطای تشخیص (درهم‌ریختگی) برای طبقه‌بندی دو دسته‌ای و به شکل زیر تعریف می‌شود[۳۳]:

جدول ۵:ماتریس خطای تشخیص (درهم‌ریختگی)

		دسته پیش‌بینی		جمع ردیف
		A_1	A_2	
دسته واقعی	A_1	درست دسته‌بندی‌شده‌اند (TP)	غلط دسته‌بندی‌شده‌اند (FN)	P
	A_2	غلط دسته‌بندی‌شده‌اند (FP)	درست دسته‌بندی‌شده‌اند *(TN)*	N

نحوه محاسبه هریک از سنجه‌ها در دو وضعیت دو دسته‌ای و چند دسته‌ای در جدول زیر آمده است:

جدول ۶: روابط سنجه‌های ارزیابی [۳۴]

شاخص	مدل دو دسته‌ای	مدل چند دسته‌ای
حساسیت	$\frac{TP}{TP+FN}$	$\frac{\sum_{i=1}^{j} TP_i}{\sum_{i1}^{j} TP_i + \sum_{i1}^{j} FN_i}$
شفافیت	$\frac{TN}{TN+FP}$	$\frac{\sum_{i=1}^{j} TN_i}{\sum_{i1}^{j} TN_i + \sum_{i1}^{j} FP_i}$
دقت	$\frac{TP}{TP+FP}$	$\frac{\sum_{i=1}^{j} TP_i}{\sum_{i1}^{j} TP_i + \sum_{i1}^{j} FP_i}$
صحت	$\frac{\mathbf{TP+TN}}{\mathbf{TP+FP+TN+FN}}$	$\frac{\sum_{i=1}^{j} \mathbf{TP_i} + \sum_{i=1}^{j} \mathbf{TN_i}}{\sum_{i=1}^{j} \mathbf{TP_i} + \sum_{i=1}^{j} \mathbf{TN_i} + \sum_{i=1}^{j} \mathbf{FP_i} + \sum_{i=1}^{j} \mathbf{FN}}$

راه‌حل دیگر برای استفاده از دقت و حساسیت ترکیب و تبدیل آن‌ها به یک معیار دیگر است. این همان رویکردی است که در معیار F از آن استفاده‌شده است. این معیار به در رابطه (۱۵) تعریف‌شده است[۲۰]:

$$F = \frac{\text{۲} * \text{دقت} * \text{حساسیت}}{\text{دقت} + \text{حساسیت}} \qquad (۱۵)$$

برای تنظیم پارامترهای الگوریتم PSO ادبیات موضوع نشان می دهد که بهترین مقادیر اولیه و نهایی ضرائب شتاب C_1 و C_2 به صورت زیر تنظیم می‌شود[۳۲]. جدول زیر مقادیر پارامترهای مورداستفاده در الگوریتم PSO را نشان می‌دهد.

جدول ۷:مقادیر پارامترهای الگوریتم PSO

تعریف	نماد	مقادیر
تعداد جمعیت اولیه	N	15
حداکثر تعداد تکرارها	$maxiter$	100
حداکثر سرعت	V_{max}	.5
حداقل سرعت	V_{min}	-.5
مقدار اولیه ضریب شتاب C_1	C_{1S}	2.5
مقدار نهایی ضریب شتاب C_1	C_{1E}	.5
مقدار اولیه ضریب شتاب C_2	C_{2S}	.5
مقدار نهایی ضریب شتاب C_2	C_{2E}	2.5

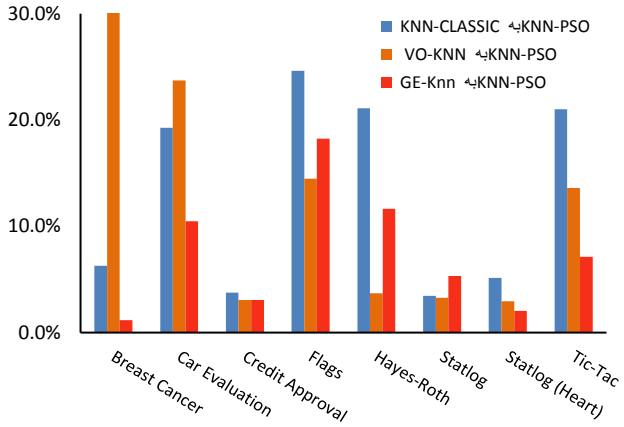
برای تفکیک داده‌های آموزشی و آزمایشی برای تمام الگوریتم‌های مورد مقایسه به ترتیب ۸۰٪ و ۲۰٪ در نظر گرفته‌شده است. مقدار K (تعداد همسایه) برابر یک لحاظ شده است. نتایج حاصل در جدول۸ آمده است که نتایج حاکی از بهبود صحت الگوریتم در کلیه پایگاه‌های داده است.

جدول ۸:مقایسه نتایج بر اساس سنجه صحت

نام پایگاه داده	KNN-PSO	KNN-CLASSIC	GE-kNN	VO-kNN
Breast Cancer	**67.8%**	63.5%	66.98%	45.46%
Car Evaluation	**95.7%**	77.3%	85.69%	73.00%
Credit Approval	**84.4%**	81.2%	81.82%	81.82%
Flags	**72.2%**	54.4%	59.05%	61.76%
Hayes-Roth	**84.2%**	66.5%	74.42%	81.11%
Statlog	**74.0%**	71.4%	70.02%	71.55%
Statlog (Heart)	**83.2%**	78.9%	81.45%	80.71%
Tic-Tac	**97.4%**	77.0%	90.47%	84.17%

از بین ۸ مجموعه داده مورد بررسی سه مجموعه داده Car Evaluation، Hayes-Roth و Tic-Tac تمام صفات آنها غیرعددی است. در سه مجموعه داده ذکر شده، الگوریتم KNN-PSO در مقایسه با الگوریتم KNN-CLASSIC دارای کارآیی کاملاً مناسبی است در حالیکه در سایر مجموعه داده‌ها که دارای ترکیبی از صفات عددی و غیرعددی هستند درصد میزان بهبود در مقایسه با سه الگوریتم گفته شده کمتر است. هرچند مجموعه داده flag که ۶۴٪

صفات آن غیرعددی است عملکرد مناسبی را نشان می دهد.



نمودار ۱: مقایسه نسبت بهبود درسنجه صحت

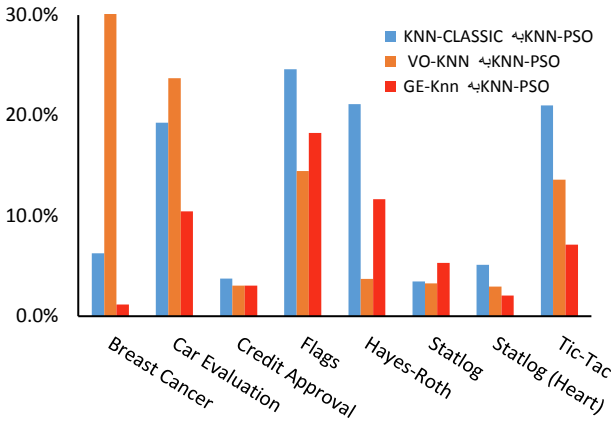
نمودار ۱ مقایسه نسبت بهبود درسنجه صحت الگوریتم KNN-PSO به سایرالگوریتم ها را نشان می‌دهد. همانطور که در نمودار ۱ مشخص شده‌است الگوریتم KNN-PSO نسبت به سه الگوریتم دیگر عملکرد بهتری را دارد و در تمام پایگاه داده ها برتری دارد.

جدول۹ مقایسه سنجه حساسیت در الگوریتم KNN-PSO با سایر الگوریتم دیگر را نشان می‌دهد.

جدول ۹:مقایسه نتایج بر اساس سنجه حساسیت

نام پایگاه داده	KNN-PSO	KNN-CLASSIC	GE-kNN	VO-kNN
Breast Cancer	**61.27%**	57.01%	61.11%	58.87%
Car Evaluation	**90.98%**	70.85%	90.00%	73.78%
Credit Approval	**83.48%**	81.36%	77.50%	83.42%
Flags	**63.71%**	54.32%	60.00%	60.00%
Hayes-Roth	**92.29%**	67.35%	70.30%	69.12%
Statlog	**69.04%**	65.44%	56.67%	66.11%
Statlog (Heart)	**83.84%**	79.43%	80.10%	82.91%
Tic-Tac	**99.85%**	83.58%	91.73%	86.07%

نتایج نشان می دهد که در سنجه حساسیت در مقایسه با سه مدل دیگر، الگوریتم پیشنهادی عملکرد مناسبی دارد. این عملکرد در سه پایگاه داده که تمام صفات آنها غیرعددی است دارای ثبات بهتری است. نمودار ۲ نسبت برتری الگوریتم KNN-PSO را به سایر الگوریتم ها نشان می‌دهد.



نمودار ۲: مقایسه نسبت بهبود درسنجه حساسیت

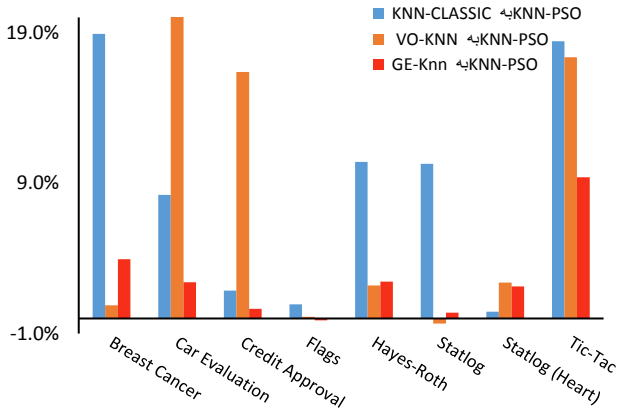
در جدول ۱۰ از سنجه شفافیت برای مقایسه عملکرد الگوریتم KNN-PSO با سه الگوریتم دیگر استفاده شده است.

جدول ۱۰:مقایسه نتایج بر اساس سنجه شفافیت

نام پایگاه داده	KNN-PSO	KNN-CLASSIC	GE-kNN	VO-kNN
Breast Cancer	**71.63%**	58.09%	68.80%	71.00%
Car Evaluation	**97.76%**	89.74%	95.40%	74.17%
Credit Approval	**85.62%**	84.04%	85.07%	71.60%
Flags	94.11%	93.23%	**94.23%**	94.00%
Hayes-Roth	**96.43%**	86.40%	94.07%	94.32%
Statlog	74.73%	67.05%	74.44%	**74.99%**
Statlog (Heart)	**87.66%**	87.26%	85.79%	85.58%
Tic-Tac	**99.25%**	80.96%	89.94%	82.04%

در سنجه شفافیت الگوریتم پیشنهادی در مقایسه با الگوریتم KNN-CLASSIC عملکرد بهتری دارد. تنها در دو مجموعه داده Flags و Statlog به ترتیب الگوریتم‌های GE-KNN و VO-KNN برتری جزیی در این سنجه دارند. نکته قابل ذکر نسبت صفات غیرعددی به کل صفات است که در دو مجموعه ذکر شده به ترتیب ۶۴٪ و ۶۶٪ است.

نمودار زیر مقایسه نسبت الگوریتم KNN-PSO به ۳ الگوریتم دیگر را در سنجه شفافیت را نشان می‌دهد. نسبت تغییرات الگوریتم پیشنهادی نسبت به ترتیب الگوریتم‌های GE-KNN و VO-KNN در پایگاه داده flag ناچیز است.



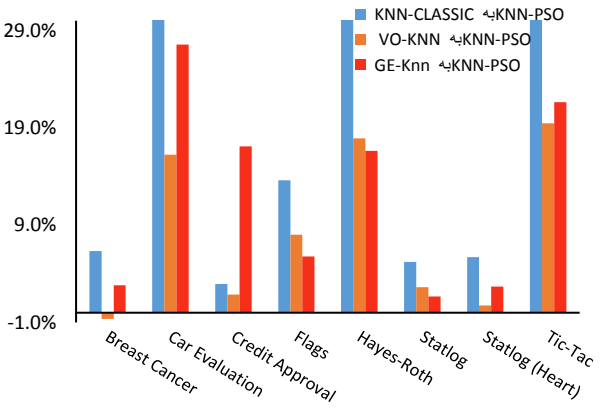
نمودار ۳: مقایسه نسبت بهبود درسنجه شفافیت

در جدول ۱۱ سنجه دقت ملاک مقایسه چهار الگوریتم است. الگوریتم KNN-PSO در مقایسه با الگوریتم KNN-CLASSIC بهبود را نشان می‌دهد و نسبت به دو الگوریتم دیگر برتری نسبی دارد و تنها در مجموعه داده Breast Cancer الگوریتم VO-KNN برتری کوچکی دارد.

جدول ۱۱:مقایسه نتایج بر اساس سنجه دقت

نام پایگاه داده	KNN-PSO	KNN-CLASSIC	GE-kNN	VO-kNN
Breast Cancer	60.59%	56.75%	58.87%	**60.98%**
Car Evaluation	**90.35%**	54.25%	65.46%	75.69%
Credit Approval	**83.38%**	80.93%	69.14%	81.82%
Flags	**63.20%**	54.60%	59.54%	58.13%
Hayes-Roth	**90.64%**	58.80%	75.58%	74.42%
Statlog	**67.79%**	64.24%	66.67%	66.02%
Statlog (Heart)	**83.08%**	78.33%	80.84%	82.45%
Tic-Tac	**99.92%**	67.38%	78.32%	80.47%

نمودار ۴ مقایسه نسبی درسنجه دقت الگوریتم KNN-PSO به سایرالگوریتم ها را نشان می‌دهد.



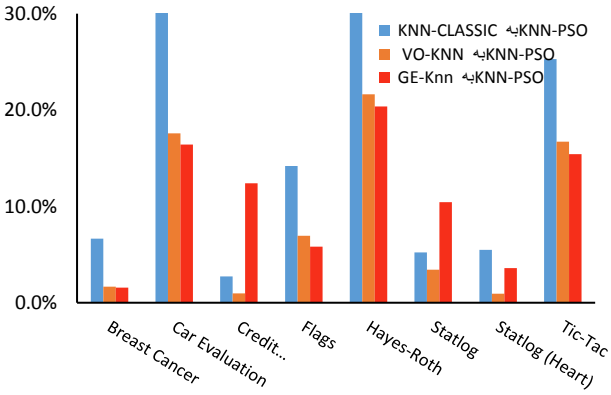
نمودار۴: مقایسه نسبت بهبود درسنجه دقت

سنجه F، میانگین هارمونیک دو سنجه دقت و حساسیت است. در جدول ۱۲ مقایسه نتایج بر اساس این سنجه را نشان می‌دهد. نتایج بهبود این سنجه در الگوریتم KNN-PSO نسبت به سه الگوریتم دیگر را نشان می‌دهد.

جدول ۱۲:مقایسه نتایج بر اساس سنجه f-score

نام پایگاه داده	KNN-PSO	KNN-CLASSIC	GE-KNN	VO-KNN
Breast Cancer	**60.93%**	56.88%	59.97%	59.91%
Car Evaluation	**90.66%**	61.45%	75.79%	74.73%
Credit Approval	**83.43%**	81.14%	73.08%	82.61%
Flags	**63.45%**	54.46%	59.77%	59.05%
Hayes-Roth	**91.46%**	62.79%	72.84%	71.67%
Statlog	**68.41%**	64.84%	61.26%	66.07%
Statlog (Heart)	**83.46%**	78.88%	80.47%	82.68%
Tic-Tac	**99.89%**	74.61%	84.50%	83.18%

نمودار زیر مقایسه نسبت PSO-KNN با سایرالگوریتم ها در سنجه F را نشان می‌دهد.



نمودار۵: مقایسه نسبت بهبود درسنجه f-score

آنچه در مقایسه مدل PSO-KNN با سایر مدلها در سنجه‌های مختلف قابل‌مشاهده است، بهبود سنجه‌ها در مدل پیشنهادی نسبت به سایر مدلهاست. متوسط میزان بهبود در هر یک از سنجه‌ها در جدول زیر مشخص‌شده است:

جدول۱۳: متوسط میزان بهبود در هریک از سنجه ها

به PSO-KNN	صحت	حساسیت	شفافیت	دقت	F-Score
KNN-CLASSIC	16.2%	15.4%	10.1%	25.8%	20.6%
VO-KNN	15.8%	11.1%	9.7%	9.8%	10.5%
GE-KNN	8.4%	10.2%	2.8%	15.0%	12.6%

۸- نتیجه گیری و پیشنهادات آتی

در تعیین مقادیر فاصله، در صورتی که روابط ذاتی بین وضعیتهای مختلف صفات غیرعددی در نظر گرفته شود، معیار فاصله معتبرتر و دقیق‌تر عمل می‌کند و در نتیجه اعتبار و اثربخشی الگوریتم KNN بیشتر می شود. در روش ارائه شده، تعیین میزان فاصله بین دو وضعیت یک صفت غیرعددی به عنوان مساله بهینه سازی در نظر گرفته شده‌است و با استفاده از الگوریتم فراابتکاری PSO مقادیر رضایت بخش به عنوان فاصله بهینه مشخص می‌شود. از آزمایش روی داده ها مشخص شد که راه حل پیشنهادی موثر و اثر بخش است و باعث ارتقاء عملکرد الگوریتم KNN می شود. ایده ارائه شده از چند جنبه ارزش تحقیقات بیشتری را دارد:۱.بکارگیری سایر مدلهای داده کاوی که از مفهوم فاصله استفاده می کنند تا با استفاده از الگوریتم‌های فراابتکاری عملکرد خود را بهبود دهند.۲. بررسی ارتباط بین تعداد صفات غیرعددی و تعداد وضعیتهای آن در یک پایگاه داده با میزان بهبود عملکرد الگوریتم پیشنهادی که خود مساله‌ای است که می تواند در آینده مورد توجه محققین قرار گیرد.۳. استفاده از سایر الگوریتم های فراابتکاری برای حل مساله و بررسی میزان قدرت اثربخشی آنها در بهبود الگوریتم های دسته بندی.۴. تغیر سازوکارهای PSO متناسب با الگوریتم KNN ۵. استفاده از سایر روشهای محاسبه فاصله بجز اقلیدسی در مسائلی که ترکیبی از صفات عددی و غیرعددی دارند. ضمن آنکه برای پایگاه دادهای که تنها دارای صفات اسمی هستند، روشهای محاسبه فاصله خاصی وجود دارد که می‌توان از ایده ارائه شده در این مقاله برای افزایش دقت از آن استفاده نمود.

مراجع

[1] X. V. Wu, "The top ten algorithm in data mining," *International Standard Book*, vol. 13, pp. 978-1, 2009.

[2] E. J. J. L. Fix, "Discriminatory analysis-nonparametric discrimination: consistency properties," California Univ Berkeley, Texas, 1951.

[3] T. P. Cover, "Nearest neighbor pattern classification," *IEEE transactions on information theory*, vol. 13, no. 1, pp. 21-27, 1967.

[4] A. N. Y. Papadopoulos, Nearest Neighbor Search: A Database Perspective, New York: Springer Science & Business Media, 2006.

[5] D. T. Larose, Discovering knowledge in data: an introduction to data mining, John Wiley & Sons, 2014.

[6] K. G. Q. L. J. Z. J. X. W. M. Zheng, "Applications of support vector machine and improved k-Nearest Neighbor algorithm in fault diagnosis and fault degree evaluation of gas insulated switchgear," in *2017 1st International Conference on Electrical Materials and Power Equipment*, Xian, PEOPLES R CHINA, 2017.

[7] M. P.-N. Steinbach, "kNN: k-nearest neighbors," in The top ten algorithms in data mining, Chapman and Hall/CRC, 2009, pp. 165-176..

[8] Y. Lin, J. Li, M. Lin and J. Chen, "A new nearest neighbor classifier via fusing neighborhood information," Neurocomputing, pp. 164-169, 2014.

[9] S. V. V. Boriah, "Similarity measures for categorical data: A comparative evaluation," in Proceedings of the 2008 SIAM international conference on data mining, 2008.

[10] Z. H. Šulc, "Comparison of Similarity Measures for Categorical Data in Hierarchical Clustering," Journal of Classification, vol. 36, no. 1, pp. 58-72, 2019.

[11] S. D. Z. Z. Luo, "Non-Numerical Nearest Neighbor Classifiers with Value-Object Hierarchical Embedding," Expert Systems with Applications, vol. 150, p. 113206, 2020.

[12] A. H. V. Desai, "Disc: Data-intensive similarity measure for categorical data," in Pacific-Asia Conference on Knowledge Discovery and Data Mining, Berlin, Heidelberg, 2011.

[13] L. Y. G. J. Chen, "Kernel-based linear classification on categorical data," Soft Computing, vol. 20, no. 8, pp. 2981-2993, 2016.

[14] D. Z. Z. Luo, "Non-Numerical Nearest Neighbor Classifiers with Value-Object Hierarchical Embedding," Expert Systems with Applications, p. 113206, 2020.

[15] L. G. Chen, "Nearest neighbor classification of categorical data by attributes weighting," Expert Systems with Applications, vol. 42, no. 6, pp. 3142-3149, 2015.

[16] J. P. G. C. Domeniconi, "Locally adaptive metric nearest-neighbor classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1281-1285, 2002.

[17] M. S. A. K. M. R. M. K. R. M. K. S. Huda and C. M. Rahman, "A dynamic k-nearest neighbor algorithm for pattern analysis problem," in *3rd International Conference on Electrical & Computer Engineering*, 2004.

[18] J. Hocke and T. Martinetz, "Feature weighting by maximum distance minimization," in *International Conference on Artificial Neural Networks*, 2013.

[19] M. R. H. M. M. B. J. R. K. Hassan, "Improving k-nearest neighbour classification with distance functions based on receiver operating characteristics," in *Machine Learning and Knowledge Discovery in Databases*, Berlin, Heidelberg, 2008.

[20] K. U. E. B. O. S. Syaliman, "Improving the accuracy of k-nearest neighbor using local mean based and distance weight," in *2nd International Conference on Computing and Applied Informatics 2017, ICCAI 2017*, Medan, INDONESIA, 2018.

[21] G. D. Tutz, "Improved nearest neighbor classifiers by weighting and selection of predictors," *Statistics and Computing*, vol. 26, no. 5, pp. 1039-1057, 2016.

[22] J. H. W. S. Y. H. Gou, "A generalized mean distance-based k-nearest neighbor classifier," *Expert Systems with Applications*, vol. 115, pp. 356-372, 2019.

[23] D. Delen, Real-world data mining: applied business analytics and decision making, Upper Saddle River, New Jersey: FT Press, 2014.

[24] A. Fargus, Optimisation of Correlation Matrix Memory Prognostic and Diagnostic Systems, University of York, 2015.

[25] P. Zerzucha and B. Walczak, "Concept of (dis) similarity in data analysis," *TrAC Trends in Analytical Chemistry*, vol. 38, pp. 116-128, 2012.

[26] J. Han, J. Pei and M. Kamber, Data mining: concepts and techniques, Elsevier, 2011

[27] T. J. T. R. J. J. H. Hastie, The elements of statistical learning: data mining, inference, and prediction, Second Edition ed., springer, 2011.

[28] R. J. Eberhart, "A new optimizer using particle swarm theory," in *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya,Japan, 1995.

[29] M. R. N. E. A. Imran, "An Overview of Particle Swarm Optimization Variants," *Procedia Engineering*, pp. 491-496, 2013.

[30] K. Kameyama, "Particle Swarm Optimization - A Survey," *IEICE Transactions on Information and Systems*, pp. 1354-1361, 2009.

[31] W. F. A. A.-S. M. A. Abd-El-Wahed, "Integrating particle swarm optimization with genetic algorithms for solving nonlinear optimization problems," Journal of Computational and Applied Mathematics, vol. 236, no. 5, pp. 1446-1453, 2011.

[32] A. S. K. H. C. Ratnaweera, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients,"*IEEE Transactions on evolutionary computation,* pp. 240-255, 2004.

[33] F. T. Provost, Data Science for Business: What you need to know about data mining and data-analytic thinking, United States of America: O'Reilly Media, Inc., 2013.

[34] N. L. A. S. Z. Z. N. E. A. Ghani, "Accuracy Assessment of Urban Growth Pattern Classification Methods Using Confusion Matrix and ROC Analysis," in *International Conference on Soft Computing in Data Science*, Singapore, 2015.

زیرنویس‌ها:

[1] K-nearest neighbor
[2] Recommendation systems
[3] Dynamic K-Nearest Neighbor
[4] Receiver Operating Characteristics
[5] Local mean based k-nearest neighbor
[6] Distance weight k-nearest neighbor
[7] Minkowski
[8] Bray and Curtis
[9] Divergence
[10] Bhattacharyya
[11] Bhattacharyya
[12] Particle Swarlerationm Optimization
[13] Acceleration Coefficients
[14] http://archive.ics.uci.edu/ml/index.php
[15] Accuracy
[16] Sensitivity
[17] Specificity
[18] Precision