

**دانشگاه تهران**

**دانشکده فنی-مهندسی کامپیوتر**

**دپارتمان الگوریتم ها و محاسبات**

**گزارش تمرین شماره ی دو**

**طراحی طبقه بند**

**نیلوفر آقایی ابیانه**

**810890001**

***چکیده***

در این پروژه ، الگوریتم های مختلف طبقه بند[[1]](#footnote-1) روی مجموعه های داده ای متعدد اجرا می شود و نتایج الگوریتم های تحلیل می شود. برای این کار از الگوریتم ***های طبقه بند اولین نزدیک ترین همسایه***[[2]](#footnote-2)، ***طبقه*** ***بند بیز***[[3]](#footnote-3)، ***k امین نزدیک ترین همسایه[[4]](#footnote-4) و پنجره ی پارزن*** [[5]](#footnote-5)استفاده می شود.

1. ***مقدمه***

در علم کامپیوتر، الگوریتم های مختلفی برای طبقه بندی وجود دارد، که بر اساس انجام کار به سه دسته ی اصلی تقسیم می شوند:

1. ***طبقه بندها بر اساس مفهوم شباهت***[[6]](#footnote-6)
2. ***طبقه بندها بر اساس روش های احتمالی***[[7]](#footnote-7)
3. ***طبقه بندها بر اساس ساخت مرز تصمیم گیری توسط بهینه سازی معیار های خطا***

در این گزارش از پنج مجموعه های داده ای[[8]](#footnote-8) تحت نام های: *مجموعه ای داده ای\_1 ، مجموعه ای داده ای\_2، مجموعه ای داده ای\_phoneme، مجموعه ای داده ای\_iris و مجموعه ای داده ای\_satimage* استفاده شده و روی الگوریتم های طبقه بند *اولین نزدیک ترین همسایه، طبقه بند بیز، k امین نزدیک ترین همسایه و پنجره ی پارزن* اجرا می شوند.

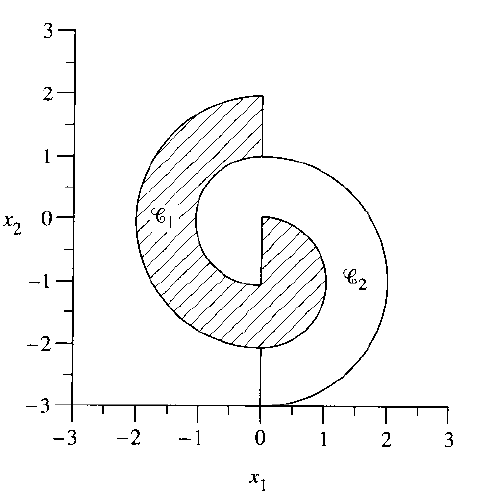
اندازه ی مجموعه ای داده ای\_1 ، مجموعه های داده ای\_2 در هر بار اجرای الگوریتم ها متغییر است در حالیکه اندازه ی مجموعه های داده ای\_phoneme، مجموعه های داده ای\_iris و مجموعه های داده ای\_satimage ثابت می باشند. در واقع از مجموعه ای داده ای\_1 ، مجموعه های داده ای\_2، تعدادی نقطه، به صورت تصادفی برای هر الگوریتم تولید شده و در آخر نتایج به دست آمده با هم مقایسه می شوند. برای پیاده سازی این الگوریتم ها و همچنین انتخاب نقاط از مجموعه ای داده ای\_1 ، مجموعه های داده ای\_2 از محیط MATLAB استفاده می شود.

***2.مجموعه های داده ای***

همانطور که گفته شد در این پروژه از پنج مجموعه ی داده ای استفاده می شود.

1. ***مجموعه ای داده ای\_1***

نقاط این مجموعه های داده ای در هر بار آزمایش از بازه ای که در شکل 1 نشان داده شده است، انتخاب می شوند.

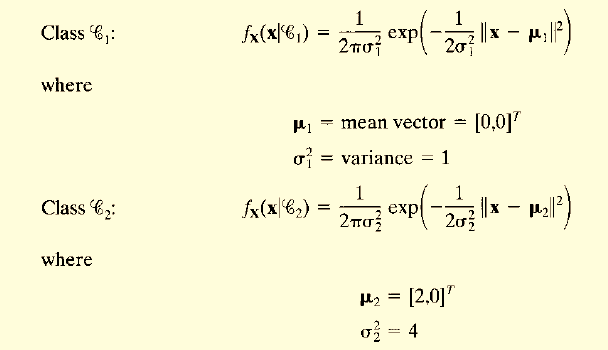


شکل1-بازه ی انتخاب نقاط

برای این کار در محیط MATLAB تابع dataset\_1 تعریف شده است؛ این تابع در هر بار اجرا، جداگانه برای هر کلاس با استفاده از توزیع یکنواخت ، به اندازه ی pointnumber نقطه انتخاب می کند و در ماتریس های class\_1 و class\_2 قرار می دهد. به ماتریس class\_1 برچسب A و به ماتریس class\_2 بر چسب B می زند . در نتیجه، یک مجموعه های داده ای با دو طبقه بدست می آید.

1. ***مجموعه های داده ای\_2***

نقاط این مجموعه های داده ای در هر بار آزمایش از بازه ای که درفرمول شکل 2 نشان داده شده است، انتخاب می شوند.



شکل2-فرمول بدست آوردن نقاط با استفاده از تابع گاوس

برای این کار در محیط MATLAB تابع dataset\_2 تعریف شده است؛ این تابع در هر بار اجرا، جداگانه برای هر کلاس با استفاده از توزیع نرمال ، به اندازه ی numberofpoint نقطه انتخاب می کند و در ماتریس های class\_1 و class\_2 قرار می دهد. به ماتریس class\_1 برچسب A و به ماتریس class\_2 بر چسب B می زند . در نتیجه، یک مجموعه های داده ای با دو طبقه بدست می آید.

1. ***مجموعه های داده ای\_phoneme***

نقاط این مجموعه ای داده ای از پایگاه داده ای ELENA تحت عنوان phoneme بدست آمده است [1]. این مجموعه ای داده ای متشکل از سه ماتریس به اندازه ی 6x540 است.

1. ***مجموعه های داده ای\_iris***

نقاط این مجموعه ای داده ای از پایگاه داده ای ELENA تحت عنوان Iris بدست آمده است [2]. این مجموعه ای داده ای، متشکل از یک ماتریس به اندازه ی 5x150 است.

1. ***مجموعه ای داده ای\_satimage***

نقاط این مجموعه ای داده ای از پایگاه داده ای ELENA تحت عنوان satimage بدست آمده است [3]. این مجموعه ای داده ای، متشکل از سه ماتریس به اندازه ی 37x6435 است.

***3.الگوریتم ها***

همانطور که گفته شد در این پروژه از سه مجموعه ی داده ای استفاده می شود؛ که در زیر آمده است.

1. ***طبقه بند اولین نزدیک ترین همسایه (OneNN)***

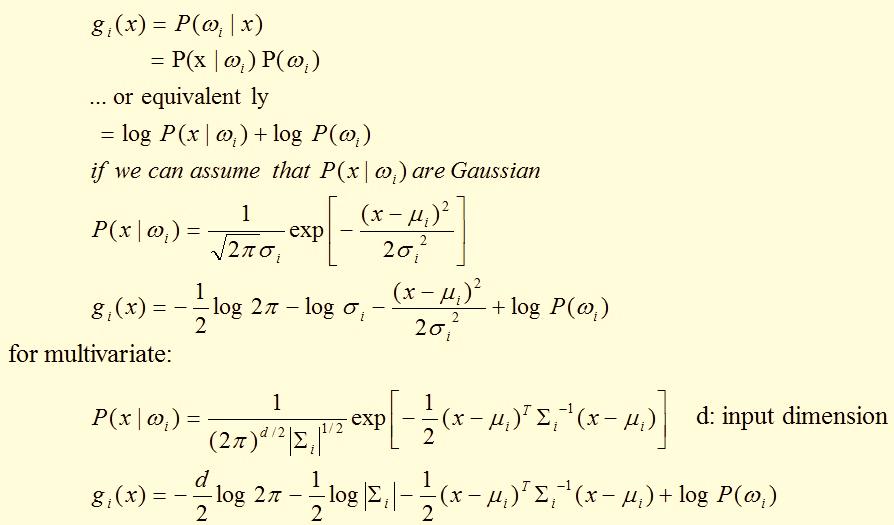
در این روش فاصله ی هر نمومه تست، با همه ی نمونه های train مقایسه می شود و در آخر بر چسب نزدیکترین نقطه را به خود می گیرد.

برای این الگوریتم، در MATLAB تا بع OneNN تعریف شده است. این تابع دو ماتریس train و test و شماره ی مجموعه ای داده ای را به عنوان ورودی می گیرد و حاصل را در خروجی را در ماتریس matrix قرار می دهد.

1. ***طبقه بند بیز(Bayes)***

در این روش تعلق یک نمونه تست، بر اساس تابع احتمال بیز (فرمول زیر) مشخص می شود.

که پس از محاسبات تعلق یک نمونه تست با استفاده از فرمول شکل 3 بدست می آید.



شکل3-فرمول بیز برای تعلق یک عنصر به یک کلاس

برای این الگوریتم، در MATLAB تا بع Bayes تعریف شده است. این تابع دو ماتریس train ، test و شماره ی مجموعه ای داده ای را به عنوان ورودی می گیرد و حاصل را در خروجی را در ماتریس matrix قرار می دهد.

1. ***طبقه بند k امین نزدیک ترین همسایه (KNN)***

این روش مشابه روش اولین نزدیک ترین همسایه می باشد. در این روش فاصله ی هر نمومه تست، با همه ی نمونه های train مقایسه می شود و در آخرkتا نزدیکترین عنصر به نمونه تست در نظر گرفته می شوند و بر چسب نمونه تست برچسب بیشترین را می گیرد اما در شرایطی که برابر باشد، برای بر چسب نمومه تست، روش های متعددی وجود دارد.

در این پروژه، برای بر چسب نمومه تست، میانگین فاصله kتا عنصر محاصبه می شود؛ سپس بر چسب عنصری که به میانگین نزدیکتر است به عنوان بر چسب نمومه تست در نظر گرفته می شود.

برای این الگوریتم، در MATLAB تا بع KNN تعریف شده است. این تابع دو ماتریس train ، test ، شماره ی مجموعه ای داده ای و K را به عنوان ورودی می گیرد و حاصل را در خروجی را در ماتریس matrix قرار می دهد.

در اینجا هر پنج مجموعه ای داده ای با دو مقدار K=5 و K=10 بررسی می شوند.

1. ***طبقه بند پنجره ی پارزن(ParzonWindows)***

در این روش تعلق یک نمونه تست، بر اساس تابع احتمال زیر مشخص می شود.

در واقع یک شعاع ثابت،H، در نظر گرفته می شود؛ که بر اساس آن برای نمونه تست آمده شده بر چسب نمونه تست انتخاب می شود.

برای این الگوریتم، در MATLAB تا بع ParzonWindows تعریف شده است. این تابع دو ماتریس train ، test ، شماره ی مجموعه ای داده ای ،H را به عنوان ورودی می گیرد و حاصل را در خروجی را در ماتریس matrix قرار می دهد.

H متغیری است که به صورت تجربی بدست می آید. بنا به ویژگی مجموعه های داده ای مقدار H برای هر کدام متفاوت است.

در اینجا برای هر مجموعه های داده ای،هر الگوریتم برای دو مقدار متعدد H اجرا می شود. برای مجموعه های داده ای 1 و2 مقدار H=0.5 و H=1 در نظر گرفته می شود در حالیکه برای مجموعه های داده ای phoneme ، iris و satimage به ترتیب H=1و H=2 ، H=0.8 و H=1.7 و H=70 و H=180 انتخاب می شوند.

همانطور که مشاهده شد ورودی همه ی این توابع دو ماتریس train و test است. برای ایجاد این دو ماتریس در محیط MATLAB تابع selectingtestandtrain تعریف شده است؛ که این تابع ماتریس مجموعه ای داده ای را بر اساس درصدی (percentage) که از کاربر می گیرد به دو ماتریس train و test تقسیم می کند.

برای اجرای این توابع با مجموعه های داده ای در MATLAB از تابع MAIN استفاده می شود؛ که در آن به

ازای هر الگوریتم دستور switch-case به کار رفته است؛ به طوری که case1 ، case2 ، case3 وcase4 به ترتیب به الگوریتم های OneNN ، Bayes ، KNN و ParzenWindows اشاره می کنند. از طرفی برای هر یک از این case ها به ازای هر مجموعه ای داده ای از دستور switch-case وجود دارد، به روشی که case1 ، case2 ، case3 ، case4 و case5 به ترتیب به مجموعه های داده ای مجموعه ای داده ای\_1 ، مجموعه ای داده ای\_2، مجموعه ای داده ای\_phoneme، مجموعه ای داده ای\_iris و مجموعه ای داده ای\_satimage دلالت می کنند.

این برنامه یکبار به ازای یک الگوریتم و مجموعه ای داده ای خاص که توسط کاربر مشخص می شود ، اجرا می گردد و نتیجه را به کاربر اعلام می کند. برای اجرای بعدی تابع MAIN مجددا باید فراخوانی شود.

در این برنامه از توابع زیر استفاده شده است:

* MAIN : تابع اصلی
* dataset\_1 : تولید اعداد مجموعه ای داده ای\_1 به تعداد pointnumber
* dataset\_2 : تولید اعداد مجموعه ای داده ای\_2 به تعداد numberofpoint
* OneNN : تابع اجرایی الگوریتم طبقه بند اولین نزدیک ترین همسایه
* Bayes : تابع اجرایی الگوریتم طبقه بند بیز
* KNN : تابع اجرایی الگوریتم طبقه بند k امین نزدیک ترین همسایه
* ParzenWindows: تابع اجرایی الگوریتم طبقه بند پنجره ی پارزن
* Repmatman : گسترش بردار ورودی به ابعاد خواسته شده
* removeonedimension : حذف بعد اول ماتریس ورودی
* insertionsort : مرتب سازی ماتریس ورودی بر اساس یک ستون خاص
* findclasses : بر گرداندن لیست همه ی کلاس ها ی trainingset
* selectingtestandtrain : انتخاب ماتریس های train و test با درصد مورد نظر کاربر
* insertionsoret\_onlabel : مرتب سازی ماتریس ورودی بر اساس کلاس ها
* efficiency : محاصبه ی درستی طبقه بندها ( عددی بین 0 -1)

***4.آزمایش ها و نتایج***

در این قسمت نتایج حاصل از اجرای الگوریتم های مختلف روی مجموعه های داده ای متفاوت با اندازه های متعدد در جداول 1-20 آمده است؛ که در هر جدول هر سطر بیانگر اندازه ی مجموعه های داده ای و هر ستون بیانگر درصد انتخاب عناصر تست و هر داریه جدول بیانگر کارایی آن الگوریتم روی اندازه ی مجموعه های داده ای خاص با درصد مشخص است.

1. با انجام آزمایش ها روی الگوریتم OneNN و مجموعه های داده ای: مجموعه ای داده ای\_1 ، مجموعه ای داده ای\_2، مجموعه ای داده ای\_phoneme، مجموعه ای داده ای\_iris و مجموعه ای داده ای\_satimage نتایج حاصل در جدول های 1-5 آمده است.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.3 | 0.2 | 0.15 | 0.1 | 0.05 |  |
| 0.8167 | 0.8250 | 0.7667 | 0.9000 | 0.9000 | **100x3** |
| 0.8417 | 0.8000 | 0.8000 | 0.7500 | 0.6500 | **200x3** |
| 0.8767 | 0.8550 | 0.8533 | 0.8000 | 0.8400 | **500x3** |
| 0.8133 | 0.8275 | 0.7967 | 0.8450 | 0.8700 | **1000x3** |
| 0.8108 | 0.8450 | 0.8433 | 0.7975 | 0.8300 | **2000x3** |

جدول 1-نتایج حاصل از اجرای الگوریتم OneNN روی مجموعه ای داده ای\_1 با درصد تست های مختلف

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.3 | 0.2 | 0.15 | 0.1 | 0.05 |  |
| 0.8167 | 0.8250 | 0.9000 | 0.9500 | 0.9000 | **100x3** |
| 0.9000 | 0.9000 | 0.8000 | 0.8000 | 0.8500 | **2000x3** |
| 0.8600 | 0.8300 | 0.8467 | 0.8200 | 0.8600 | **500x3** |
| 0.8300 | 0.8375 | 0.8100 | 0.8600 | 0.7900 | **1000x3** |
| 0.8583 | 0.8617 | 0.8650 | 0.8675 | 0.8550 | **2000x3** |

جدول 2-نتایج حاصل از اجرای الگوریتم OneNN روی مجموعه ای داده ای\_2 با درصد تست های مختلف

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.3 | 0.2 | 0.15 | 0.1 | 0.05 |  |
| 0.9112 | 0.8992 | 0.8964 | 0.9076 | 0.9004 | **5404x6** |

جدول 3-نتایج حاصل از اجرای الگوریتم OneNN روی مجموعه ای داده ای\_phoneme با درصد تست های مختلف

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.3 | 0.2 | 0.15 | 0.1 | 0.05 |  |
| 1 | 0.9333 | 0.9565 | 1 | 1 | **150x5** |

جدول 4-نتایج حاصل از اجرای الگوریتم OneNN روی مجموعه ای داده ای\_iris با درصد تست های مختلف

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.3 | 0.2 | 0.15 | 0.1 | 0.05 |  |
| 0.9006 | 0.9005 | 0.8851 | 0.8898 | 0.9037 | **6435x37** |

جدول 5-نتایج حاصل از اجرای الگوریتم OneNN روی مجموعه ای داده ای\_satimage با درصد تست های مختلف

1. با انجام آزمایش ها روی الگوریتم Bayes و مجموعه های داده ای: مجموعه ای داده ای\_1 ، مجموعه ای داده ای\_2، مجموعه ای داده ای\_phoneme، مجموعه ای داده ای\_iris و مجموعه ای داده ای\_satimage نتایج حاصل در جدول های 6-10 آمده است.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.3 | 0.2 | 0.15 | 0.1 | 0.05 |  |
| 0.8000 | 0.6350 | 0.6767 | 0.8000 | 0.9000 | **100x3** |
| 0.7917 | 0.8125 | 0.7833 | 0.8750 | 0.8000 | **200x3** |
| 0.7367 | 0.8000 | 0.7200 | 0.7500 | 0.6400 | **500x3** |
| 0.6967 | 0.7275 | 0.7367 | 0.7000 | 0.7300 | **1000x3** |
| 0.7142 | 0.7325 | 0.7367 | 0.7325 | 0.7250 | **2000x3** |

جدول 6-نتایج حاصل از اجرای الگوریتم Bayes روی مجموعه ای داده ای\_1 با درصد تست های مختلف

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.3 | 0.2 | 0.15 | 0.1 | 0.05 |  |
| 0.9000 | 0.9000 | 0.8667 | 0.9000 | 0.9000 | **100x3** |
| 0.9333 | 0.9000 | 0.9333 | 0.9000 | 0.8500 | **200x3** |
| 0.9167 | 0.9400 | 0.9200 | 0.9100 | 0.9200 | **500x3** |
| 0.9133 | 0.9175 | 0.9433 | 0.9350 | 0.9600 | **1000x3** |
| 0.9100 | 0.9225 | 0.9200 | 0.9175 | 0.9300 | **2000x3** |

جدول 7-نتایج حاصل از اجرای الگوریتم Bayes روی مجموعه ای داده ای\_2 با درصد تست های مختلف

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.3 | 0.2 | 0.15 | 0.1 | 0.05 |  |
| 0.7848 | 0.7983 | 0.8101 | 0.7911 | 0.7897 | **5404x6** |

جدول 8-نتایج حاصل از اجرای الگوریتم Bayes روی مجموعه ای داده ای\_phoneme با درصد تست های مختلف

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.3 | 0.2 | 0.15 | 0.1 | 0.05 |  |
| 0.9556 | 0.9667 | 0.9565 | 0.9333 | 0.8750 | **150x5** |

جدول 9-نتایج حاصل از اجرای الگوریتم Bayes روی مجموعه ای داده ای\_iris با درصد تست های مختلف

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0.3 | 0.2 | 0.15 | 0.1 | 0.05 |  |
| 0.8498 | 0.8570 | 0.8592 | 0.8478 | 0.8478 | **6543x37** |

جدول 10-نتایج حاصل از اجرای الگوریتم Bayes روی مجموعه ای داده ای\_satimage با درصد تست های مختلف

1. با انجام آزمایش ها روی الگوریتم KNN و انتخاب K=5 و K=10 روی مجموعه های داده ای: مجموعه ای داده ای\_1 ، مجموعه ای داده ای\_2، مجموعه ای داده ای\_phoneme، مجموعه ای داده ای\_iris و مجموعه ای داده ای\_satimage نتایج حاصل در جدول های 10-15 آمده است.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.3  K=10 | 0.3  K=5 | 0.2  K=10 | 0.2  K=5 | 0.15  K=10 | 0.15  K=5 | 0.1  K=10 | 0.1  K=5 | 0.05  K=10 | 0.05  K=5 |  |
| 0.7500 | 0.8167 | 0.7250 | 0.7500 | 0.8333 | 0.7667 | 0.9500 | 0.9000 | 1 | 1 | **100x3** |
| 0.8500 | 0.8583 | 0.8625 | 0.8875 | 0.9000 | 0.8667 | 0.8250 | 0.8000 | 0.7500 | 0.8000 | **200x3** |
| 0.7900 | 0.7600 | 0.8100 | 0.8050 | 0.8400 | 0.8800 | 0.8600 | 0.8100 | 0.8600 | 0.8600 | **500x3** |
| 0.8433 | 0.8317 | 0.8300 | 0.8275 | 0.8667 | 0.8367 | 0.8550 | 0.8800 | 0.8600 | 0.8800 | **1000x3** |
| 0.8283 | 0.8250 | 0.8075 | 0.8175 | 0.8183 | 0.8233 | 0.8000 | 0.8000 | 0.8500 | 0.8350 | **2000x3** |

جدول 11-نتایج حاصل از اجرای الگوریتم KNN روی مجموعه ای داده ای\_1 با درصد تست های مختلف به ازای K = 5 و K = 10 روی مجموعه های تست و train یکسان

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.3  K=10 | 0.3  K=5 | 0.2  K=10 | 0.2  K=5 | 0.15  K=10 | 0.15  K=5 | 0.1  K=10 | 0.1  K=5 | 0.05  K=10 | 0.05  K=5 |  |
| 0.7833 | 0.8500 | 0.7750 | 0.8250 | 0.6667 | 0.7000 | 0.9000 | 0.9500 | 0.8000 | 0.8000 | **100x3** |
| 0.8917 | 0.8750 | 0.8125 | 0.8125 | 0.8500 | 0.8667 | 0.9000 | 0.8750 | 0.9000 | 0.9000 | **200x3** |
| 0.8900 | 0.8633 | 0.8750 | 0.8600 | 0.8733 | 0.8800 | 0.9100 | 0.9100 | 0.8600 | 0.8400 | **500x3** |
| 0.9200 | 0.9167 | 0.9075 | 0.8900 | 0.9067 | 0.9000 | 0.9150 | 0.9000 | 0.9000 | 0.8800 | **1000x3** |
| 0.9017 | 0.8967 | 0.8950 | 0.8938 | 0.9167 | 0.9150 | 0.8900 | 0.8800 | 0.8900 | 0.8800 | **2000x3** |

جدول 21-نتایج حاصل از اجرای الگوریتم KNN روی مجموعه ای داده ای\_2 با درصد تست های مختلف به ازای K = 5 و K = 10 روی مجموعه های تست و train یکسان

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.3  k=10 | 0.3  k=5 | 0.2  k=10 | 0.2  k=5 | 0.15  k=10 | 0.15  k=5 | 0.1  k=10 | 0.1  k=5 | 0.05  k=10 | 0.05  k=5 |  |
| 0.8545 | 0.8755 | 0.8733 | 0.8927 | 0.8570 | 0.8866 | 0.8632 | 0.8706 | 0.8782 | 0.9041 | **5404x6** |

جدول 13-نتایج حاصل از اجرای الگوریتم KNN روی مجموعه ای داده ای\_phoneme با درصد تست های مختلف به ازای K =5 و K =10 روی مجموعه های تست و train یکسان

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.3  k=10 | 0.3  k=5 | 0.2  k=10 | 0.2  k=5 | 0.15  k=10 | 0.15  k=5 | 0.1  k=10 | 0.1  k=5 | 0.05  k=10 | 0.05  k=5 |  |
| 0.9778 | 0.9778 | 1 | 0.9667 | 0.9565 | 1 | 0.8667 | 0.9333 | 1 | 1 | **150x5** |

جدول 14-نتایج حاصل از اجرای الگوریتم KNN روی مجموعه ای داده ای\_iris با درصد تست های مختلف به ازای K =5 و K =10 روی مجموعه های تست و train یکسان

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.3  k=10 | 0.3  k=5 | 0.2  k=10 | 0.2  k=5 | 0.15  k=10 | 0.15  k=5 | 0.1  k=10 | 0.1  k=5 | 0.05  k=10 | 0.05  k=5 |  |
| 0.8990 | 0.9150 | 0.8842 | 0.8958 | 0.8964 | 0.8964 | 0.8897 | 0.9098 | 0.8788 | 0.8913 | **6435x37** |

جدول 15-نتایج حاصل از اجرای الگوریتم KNN روی مجموعه ای داده ای\_satimage با درصد تست های مختلف به ازای K = 5 و K =10 روی مجموعه های تست و train یکسان

1. با انجام آزمایش ها روی الگوریتم ParzenWindows و مجموعه های داده ای: مجموعه ای داده ای\_1 ، مجموعه ای داده ای\_2، مجموعه ای داده ای\_phoneme، مجموعه ای داده ای\_iris و مجموعه ای داده ای\_satimage نتایج حاصل در جدول های 16-20 آمده است.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.3  H=1 | 0.3  H=0.5 | 0.2  H=1 | 0.2  H=0.5 | 0.15  H=1 | 0.15  H=0.5 | 0.1  H=1 | 0.1  H=0.5 | 0.05  H=1 | 0.05  H=0.5 |  |
| 0.8333 | 0.7667 | 0.8000 | 0.8250 | 0.8333 | 0.8667 | 0.7500 | 0.7500 | 0.9000 | 1 | **100x3** |
| 0.8583 | 0.8333 | 0.8500 | 0.8750 | 0.7667 | 0.8333 | 0.6750 | 0.7250 | 0.8000 | 0.8500 | **200x3** |
| 0.7300 | 0.7900 | 0.8000 | 0.8100 | 0.8067 | 0.8600 | 0.8000 | 0.8700 | 0.7600 | 0.7800 | **500x3** |
| 0.8017 | 0.8317 | 0.7800 | 0.8425 | 0.8233 | 0.8400 | 0.7500 | 0.8150 | 0.7500 | 0.8100 | **1000x3** |
| 0.7833 | 0.8233 | 0.7800 | 0.8237 | 0.8200 | 0.8617 | 0.7825 | 0.8075 | 0.8000 | 0.8550 | **2000x3** |

جدول 61-نتایج حاصل از اجرای الگوریتم ParzenWindows روی مجموعه ای داده ای\_1 با درصد تست های مختلف وH=0.5 و H=1 روی مجموعه های تست و train یکسان

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.3  H=1 | 0.3  H=0.5 | 0.2  H=1 | 0.2  H=0.5 | 0.15  H=1 | 0.15  H=0.5 | 0.1  H=1 | 0.1  H=0.5 | 0.05  H=1 | 0.05  H=0.5 |  |
| 0.7333 | 0.5333 | 0.6000 | 0.4000 | 0.6000 | 0.4000 | 0.8000 | 0.5000 | 0.6000 | 0.5000 | **100x3** |
| 0.7417 | 0.6000 | 0.7750 | 0.6500 | 0.8000 | 0.7167 | 0.8000 | 0.6000 | 0.8500 | 0.7500 | **200x3** |
| 0.8700 | 0.7500 | 0.8650 | 0.7450 | 0.9000 | 0.7667 | 0.9000 | 0.7600 | 0.9000 | 0.7400 | **500x3** |
| 0.8800 | 0.8350 | 0.8700 | 0.8125 | 0.8967 | 0.8500 | 0.9000 | 0.8350 | 0.9300 | 0.8700 | **1000x3** |
| 0.8850 | 0.8467 | 0.9038 | 0.8738 | 0.8833 | 0.8650 | 0.8975 | 0.8650 | 0.9000 | 0.8550 | **2000x3** |

جدول 17-نتایج حاصل از اجرای الگوریتم ParzenWindows روی مجموعه ای داده ای\_2 با درصد تست های مختلف و H=0.5 و H=1 روی مجموعه های تست و train یکسان

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.3  H=2 | 0.3  H=1 | 0.2  H=2 | 0.2  H=1 | 0.15  H=2 | 0.15  H=1 | 0.1  H=2 | 0.1  H=1 | 0.05  H=2 | 0.05  H=1 |  |
| 0.7466 | 0.8126 | 0.7290 | 0.8076 | 0.7287 | 0.8076 | 0.7708 | 0.7893 | 0.7601 | 0.8192 | **5404x6** |

جدول 18-نتایج حاصل از اجرای الگوریتم ParzenWindows روی مجموعه ای داده ای\_phoneme با درصد تست های مختلف وH=1 و H=2 روی مجموعه های تست و train یکسان

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.3  H=1.7 | 0.3  H=0.8 | 0.2  H=1.7 | 0.2  H=0.8 | 0.15  H=1.7 | 0.15  H=0.8 | 0.1  H=1.7 | 0.1  H=0.8 | 0.05  H=1.7 | 0.05  H=0.8 |  |
| 0.8667 | 0.9111 | 0.9000 | 1 | 0.9565 | 1 | 0.9333 | 0.8667 | 0.7500 | 1 | **150x5** |

جدول 19-نتایج حاصل از اجرای الگوریتم ParzenWindows روی مجموعه ای داده ای\_iris با درصد تست های مختلف وH=0.8 و H=1.7 روی مجموعه های تست و train یکسان

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.3  H=60 | 0.3  H=20 | 0.2  H=60 | 0.2  H=20 | 0.15  H=60 | 0.15  H=20 | 0.1  H=60 | 0.1  H=20 | 0.05  H=60 | 0.05  H=20 |  |
| 0.8394 | 0.4505 | 0.8461 | 0.4848 | 0.8550 | 0.4668 | 0.8416 | 0.5124 | 0.8478 | 0.4813 | **6435x37** |

جدول 20-نتایج حاصل از اجرای الگوریتم ParzenWindows روی مجموعه ای داده ای\_satimage با درصد تست های مختلف و h=20 و h=60 روی مجموعه های تست و train یکسان

***7.تحلیل نتایج***

از جداول 1-5 مشاهده می شود که الگوریتم OneNN دارای کارایی بالایی می باشد؛ اگرچه این الگوریتم برای داده هایی نویزی مناسب نمی باشد و دارای سربار محاسباتی بالایی است، زیرا به ازای هر نمونه تست فاصله ی آن نمونه را با همه ی نمونه های train محاصبه می کند .

اگر فضای کار d بعدی باشد، آنگاه فاصله از فرمول زیر به دست می آید:

یک نمونه تست :

یک نمونه train :

که این کار را به ازای هر عنصر تست با همه ی عناصر train انجام می دهد؛ پس اگر تعداد عناصر train و تست به ترتیب برابر با n و m باشد، آنگاه پیچیدگی این روش برابر با است.

از جداول 6-10 مشاهده می شود که الگوریتم Bayes نسبت به الگوریتم از کارایی کمتری بر خوردار است. در این الگوریتم هم با افزایش داده های تست و همچنین با افزایش اندازه ی کل داده ها، کارایی الگوریتم کاهش می یابد.

از جداول 11-15 مشاهده می شود که الگوریتم KNN از کارایی مناسبی بر خوردار است. در این الگوریتم برای k=5 بهتر از k=10 جواب می دهد. این الگوریتم از هر دو الگوریتم OneNN و Bayes بهتر عمل می کند.

از جداول 16-20 مشاهده می شود که الگوریتم ParzenWindows از همه ی الگوریتم های دیگر بهتر عمل می کند گو اینکه کارایی این الگوریتم خیلی به مقدار H بستگی دارد. این الگوریتم در dataset\_1 برای H=0.5، در dataset\_2 برای H=1، در dataset\_phoneme برای H=1، در dataset\_iris برای H=0.8، و در dataset\_satimage برای H=60 خیلی بهتر ، جواب می دهد.

به طور کلی در بین این چهار الگوریتم، ParzenWinodws در شرایطی که مقدار H مناسب انتخاب شود از سایر الگوریتم ها کارایی بهتری دارد.

***6.منابع***

**1. http://sci2s.ugr.es/keel/dataset.php?cod=105**

**2.** **http://archive.ics.uci.edu/ml/datasets/Iris**

**3.http://www.dice.ucl.ac.be/neural-nets/Research/Projects/ELENA/databases/REAL/satimaget**

1. Classifier [↑](#footnote-ref-1)
2. One-nearest Neighbor [↑](#footnote-ref-2)
3. Bayes Classifier [↑](#footnote-ref-3)
4. K-nearest Neighbor [↑](#footnote-ref-4)
5. Parzen Window [↑](#footnote-ref-5)
6. Concept of Similarity [↑](#footnote-ref-6)
7. Probabilistic Approach [↑](#footnote-ref-7)
8. Dataset [↑](#footnote-ref-8)