پیاده سازی الگوریتم های parzen و knn با تعیین مناسبترین مقدار پارامتر ورودی

مبانی هوش محاسباتی دکتر ابراهیم پور بهار ۹۸
الهه شفیع پور (۹۴۹۹۱۲۶۵)
۱- دانشکده مهندسی برق و کامپیوتر - دانشگاه شهید رجایی - تهران - ایران – eshb195@gmail.com

چکیده: در این مقاله، به شرح الگوریتم های parzen (براساس شعاع همسایگی) و knn (k همسایه ی نزدیک) پرداخته و چون در این دو الگوریتم پارامتری وجود دارد که به عنوان ورودی به الگوریتم باید داده شود، یافتن مناسب ترین مقدار این دو پارامتر هم بیان می شود و تحلیل می گردد که مقادیر نامناسب چه تاثیری بر این الگوریتم ها دارند. در ادامه نتایج پیاده سازی این الگوریتم ها را می بینیم. لازم به ذکراست که هر کدام از این الگوریتم ها در مسایل مخصوص خود بسیار مفید هستند.

واژه‌های کلیدی: parzen و knn و شعاع همسایگی

۱- مقدمه

الگوریتم های parzen (براساس شعاع همسایگی) و knn (k همسایه ی نزدیک) هر براساس فاصله ی اقلیدسی عمل میکنند و به داده ی تست ما برچسب نزدیک ترین داده به آن، را میدهند. برای انتخاب نزدیک ترین داده ها از شعاع همسایگی و یا یک یا چند همسایه ی نزدیک استفاده می شود.

۲- الگوریتم ها

در این جا به شرح الگوریتم های parzen (براساس شعاع همسایگی) و knn (k همسایه ی نزدیک) پرداخته و در دو بخش مجزا به تفضیل این دو الگوریتم را بررسی می کنیم.

۲-۱- الگوریتم knn

این الگوریتم توسعه یافته ی الگوریتم 1nn است با این تفاوت که برای هر داده ی تست ما k همسایه ی نزدیک به داده ی مدنظر را براساس فاصله داده ها از هم یافته و در همسایه ها چک میکنیم برچسب شان چیست. هر چه یک برچسب در این همسایه ها بیشتر تکرار می شد یعنی احتمال بیشتری نسبت به سایر برچسب ها دارد و به عنوان برچسب داده ی تست ما پیشبینی میگردد.

معیار k اگر خیلی بزرگ باشد داده های زیادی را در بر میگیرد و نتیجه درستی نمیدهد. معمولا نهایت این معیار ۱۱ و یا ۱۵ است.

برای تعیین این پارامتر در مسئله مجموعه ی valid و داده های آموزش را درنظر گرفته و بر روی مجموعه ی valid برای k های مختلف روند الگوریتم knn را طی کرده و دقت (فاصله تا داده های آموزش) وکارایی آن را حساب می کنیم . هر چه معیار دقت ما بیشتر بود آن k را در نظر می گیریم. در نهایت با همین k برروی داده های تست الگوریتم را اعمال کرده و معیار کارایی را می یابیم.

۲-۲- الگوریتم parzen

در این الگوریتم یک شعاع همسایگی داریم که مرکز ما داده ی تست ما هست. داده هایی که فاصله یشان تا داده ی تست ما از شعاع کمتر باشد را درنظر گرفته و برچسبشان را بررسی می کنیم. تعداد و احتمال هر برچسبی بیشتر بود آن برچسب داده ی تست ما نیز می شود. پارامتر شعاع اگر خیلی کوچک باشد هیچ داده ای را در خود جا نمی دهد و نمیتوان پیشبینی کرد. شعاع بزرگ هم احتمال خطا را به علت وجود داده های زیاد افزایش می دهد پس برای هر مسئله شعاع مناسب باید تعیین گردد.
برای تعیین شعاع مانند تعیینk ، معیار کارایی را به ازای مقادیر مختلف شعاع یافته و بهترین را برای مسئله به کار می بریم.

۲-۳- معرفی دیتاست ها

دیتاست اول، دیتاست دو کلاسه با میانگین و واریانس معین برای هر کلاس است.

دیتاست دوم، دیتاست سه کلاسه با میانگین و واریانس معین برای هر کلاس است.

دیتاست سوم، دیتاست iris هست.

٤-٢- ماژول ها

در این بخش از یک ماژول که خود نو شتم به نام defs ا ستفاده شده که درون کدها از توابع این ماژول استفاده شده است. توابعی چون findk برای یافتن مناسبترین k و تابعی دیگر برای محاسبه دقت الگوریتم knn .

در کل ۴ تابع آماده در این فایل اســت که به کمک اینها هم k هم شعاع را یافته و دقت را محاسبه می کنیم.

٥-٢- پیاده سازی

داده های تســت وtrain با نسـبت ۰,۳ جدا شـده اند و از مجموعه ی train دو قسـمت train و valid با نسـبت ۰,۵ و به صـورت رندوم از بین داده ها انتخاب شده است.

۳- نتایج

نتایج اجرای الگوریتم ها بر روی هر دیتاست با بیان مقدار k و یا r مناسب که در کد6یافت شده و کدام دیتاست و تعداد نمونه های آن و زمانی برنامه در حال اجرا بوده در جدول زیر آمده است.

	precision	K or R	Dataset : count	time
knn	0.97222	11	1 : 100	**0.1512**
parzen	0.96875	1	1 : 100	**0.08172**
knn	0.97740	1	1 : 1000	23.3174
parzrn	0.95305	1	1 : 1000	**6.652**
knn	0.95570	0.1065	1:150	**0.1609**
parzen	0.8953	0.0180	2:150	**0.1260**
knn	0.928121	5	3 iris	**0.195049**
parzen	0.84210	2	3 iris	**0.13913**

معیار کارایی در الگوریتم knn دقیق تر از parzen اسـت با این که هر دو مبتنی بر فاصله اند . وقتی تعداد داده زیاد شود هر دو زمان بیشتری صرف میکنند اما در آزمایش برروی دیتاســت اول در دو حالت ۱۰۰ و ۱۰۰۰ داده تفاوت زمانی به شدت محسوس ا ست و معیار کارایی نسبتا برابری دارند.Knn مانند 1nn لود محاسباتی بالایی دارد.

در دیتاست ها زمان کمتر parzen و کارایی بیشتر knn به چشم می خورد که تفاوت اندکی دارند حداقل در داده هایی با تعداد کم.

Python 3.5.4 Shell

```
File  Edit  Shell  Debug  Options  Window  H
the best r is:  1

precision1 for parzen:
 0.96875
time:  0.08172154426574707
>>>
 RESTART: D:\terms\term7\doctor

the best k is:  11

precision1 for knn:
 0.9722222222222222
time:  0.15128874778747559
>>>
 RESTART: D:\terms\term7\doctor
n.py
the best r is:  1

precision1 for parzen:
 0.9530589543937709
time:  6.652256965637207
>>>
 RESTART: D:\terms\term7\doctor

the best k is:  1

precision1 for knn:
 0.977403156384505
time:  23.31740713119507
>>>
```