

به نام خدا



دانشگاه صنعتی شریف

دانشکده مهندسی برق

گزارش پروژه

دنیز رضاپور کیانی

98101628

هوش مصنوعی و محاسبات زیستی

دکتر حاجی پور

+ فاز اول:

○ مرحله اول: تعریف feature ها

در دو قسمت ابتدایی، feature های زمانی و سپس feature های فرکانسی داده های train و test به طور جداگانه بررسی شده و در نهایت نرمالیزه می شوند. در واقع ابتدا داده ها را لود کرده و تغییرات لازم را ایجاد کرده و ماتریس هایی لازم را استخراج می کنیم. در بخش بعد فیچر ها محاسبه شده و سپس برای داده های تست محاسبه می شوند و نتایج به صورت ماتریس هایی ذخیره می شوند تا در مراحل بعد مورد استفاده قرار گیرند. در ادامه feature های بررسی شده را لیست کرده ام.

نکته : label های 2 برای آسانتر شدن کار با داده ها به 0 تبدیل شدند

Feature های زمانی بررسی شده عبارتند از:

- Mean
- Variance
- Mode
- Median
- Skewness: asymmetry of the data around the sample mean
- Kurtosis:

Kurtosis is a measure of how outlier-prone a distribution is. The kurtosis of the normal distribution is 3. Distributions that are more outlier-prone than the normal distribution have kurtosis greater than 3; distributions that are less outlier-prone have kurtosis less than 3. Some definitions of kurtosis subtract 3 from the computed value, so that the normal distribution has kurtosis of 0. The kurtosis function does not use this convention.

- Max
- Energy: The sum of the square of the signal
- Average Power
- RMS
- Correlation

- Moment:

The central moment of order k for a distribution is defines as:

$$m_k = E(x - \mu)^k,$$

Where mu is the mean of x and E(t) represents the expected value of quantity t. The moment function computes a sample version of this population value.

$$m_k = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^k.$$

- FormFactor:

$$FF(x) = \frac{\sigma'_x / \sigma_x}{\sigma''_x / \sigma'_x}$$

- Nonlinear Energy
- AR

$$x(n) = \sum_{i=1}^P \alpha(i)x(n-i) + e(n)$$

Feature های فرکانسی بررسی شده عبارتند از:

- Frequency Mean
- Frequency Median
- Frequency Peak
- Max-Frequency-Index
- Band-Power

پس از محاسبه ویژگی های ذکر شده برای داده های test و train، مرحله نرمالیزه کردن feature های محاسبه شده را انجام می دهیم. با استفاده دو تابع mapstd و mapminmax نرمالیزاسیون انجام می شود:

○ Mapstd:

Process matrices by mapping each row's means to 0 and deviations to 1

$$y = (x - xmean) * (ystd / xstd) + ymean;$$

○ Mapminmax:

Process matrices by mapping row minimum and maximum values to [-1 1]

$$y = (ymax - ymin) * (x - xmin) / (xmax - xmin) + ymin.$$

○ مرحله دوم: انتخاب feature ها

ابتدا برای feature های زمانی و فرکانسی به طور جداگانه ماتریس پخش را محاسبه کردم و 50 رتبه اول برحسب این معیار انتخاب می شود. در انتها دوباره عمل نرمالیزاسیون به همان شیوه پیش برای ویژگی های انتخاب شده زمانی و فرکانسی، هر یک جدا، انجام می شود و کل ویژگی ها با عنوان ماتریس BestFeature برای داده های train و test به طور جداگانه ذخیره می شوند تا در مراحل بعدی استفاده شوند.

○ مرحله سوم: شبکه MLP

ابتدا داده ها با دستور cvpartition به دو دسته train و validation تقسیم می شوند (20 درصد داده ها به عنوان validation و مابقی داده ها، داده های train هستند).

در ابتدا شبکه MLP سه لایه (لایه خروجی تک نورون) train می شود بدین صورت که به ازای تعداد نورون های متفاوت لایه میانی و به کمک patternet مدل می شود. خروجی patternet با قرار دادن ترشهلد 0.5 به دو کلاس صفر و یک دسته بندی می شوند (در ابتدای کد label های 2 برای آسانتر شدن کار با داده ها به 0 تبدیل شدند) و با مقایسه با label های داده شده میزان دقت تعیین می شود. در نهایت به ازای نورون لایه میانی که بیشترین دقت را داشته است، شبکه منتخب تعیین می شود. در ادامه با استفاده از شبکه train شده منتخب، label های داده های آزمون داده شده را تعیین کرده و در فایل نتایج قرار دادم:

در بهترین حالت، مقدار دقت به ازای 62 نورون و برابر 81٪ است:

```
accuracy =  
  
    0.8091  
  
best_neuron_number =  
  
    65
```

در قسمت بعد، تمام مراحل ذکر شده فوق برای شبکه MLP با لایه خروجی با دو نورون تکرار می شود.

در بهترین حالت، مقدار دقت به ازای 20 نورون و برابر 83٪ است:

```
ans =  
  
    0.8364  
  
best_neuron_number =  
  
    20
```

در انتهای هر دو بخش با اجرای بهترین شبکه مدل شده روی داده های آزمون داده شده (50 داده) و label های آنها با عناوین MLP_TwoOutput_Labels و MLP_OneOutput_Labels ذخیره و گزارش شده اند.

○ مرحله چهارم: شبکه RBF

ابتدا داده ها با دستور cvpartition به دو دسته train و validation تقسیم می شوند (20 درصد داده ها به عنوان validation و مابقی داده ها، داده های train هستند). شبکه عصبی سه لایه (لایه خروجی تک نورون) به ازای تعداد نورون های و شعاع های متفاوت لایه میانی و به کمک newrb مدل می شود.

خروجی patternet با قرار دادن ترشهلد 0.5 به دو کلاس صفر و یک دسته بندی می شوند (در ابتدای کد label های 2 برای آسانتر شدن کار با داده ها به 0 تبدیل شدند) و با مقایسه با label های داده شده میزان دقت تعیین می شود. در نهایت به ازای نورون لایه میانی که بیشترین دقت را داشته است، شبکه منتخب تعیین می شود. در ادامه با استفاده از شبکه train شده منتخب، label های داده های آزمون داده شده را تعیین کرده و در فایل نتایج قرار دادم:

در بهترین حالت، مقدار دقت به ازای 10 نورون و شعاع 5 برابر 83٪ است:

```
best_neuron_number =  
    10  
  
best_spread =  
    5  
  
ans =  
    0.8364
```

در قسمت بعد، تمام مراحل ذکر شده فوق برای لایه خروجی با دو نورون تکرار می شود.

در بهترین حالت، مقدار دقت به ازای 10 نورون و شعاع 15 و برابر 85٪ است:

```
best_neuron_number =  
    10  
  
best_spread =  
    15  
  
ans =  
    0.8545
```

در انتهای هر دو بخش با اجرای بهترین شبکه مدل شده روی داده های آزمون داده شده (50 داده) و label های آنها با عناوین RBF_TwoOutput_Labels و RBF_OneOutput_Labels ذخیره و گزارش شده اند.

مقایسه MLP و RBF :

شبکه RBF زمان اجرا طولانی تری دارد اما دقت به طور میانگین می تواند بیشتر از MLP باشد.

+ فاز دوم:

○ الگوریتم تکاملی:

در این بخش انتخاب دسته ویژگی های مناسب با استفاده از الگوریتم تکاملی پیاده سازی می شود. کروموزوم به صورت یک رشته 20×200 تایی ژن می باشد. (20 برای نسل ها و 200 برای ویژگی های انتخاب شده) آلل های ژن های کروموزوم 0 و 1 می باشند. 0 به معنی عدم وجود آن ویژگی و 1 به معنی وجود آن ویژگی در ژن مورد نظر است.

مقدار اولیه کروموزوم ها به صورت انتخاب 100 ویژگی از feature های زمانی و 100 ویژگی از feature های فرکانسی می باشد و هدف نهایی انتخاب 100 feature از کل آنها است.

تابع سازگاری هم براساس ماتریس های پخش می محتسبه می شود.

Mutation: n بیتی

Crossover: pair swap

با اجرای الگوریتم تکاملی و مشخص شدن ویژگی ها، شبکه های MLP و RBF قبلی دوباره train می شوند اما با ویژگی های انتخاب شده جدید. نتایج زیر با اجرای شبکه های مدل شده و انتخاب بهترین شبکه حاصل می شوند:

MLP تک خروجی:

در بهترین حالت، مقدار دقت به ازای 10 نرون برابر 83٪ است:


```
accuracy =  
  
    0.8364  
  
best_neuron_number =  
  
    10
```

MLP دو خروجی:

در بهترین حالت، مقدار دقت به ازای 30 نورون برابر 59٪ است:

```
ans =  
  
    0.5909  
  
best_neuron_number =  
  
    30
```

در انتهای هر دو بخش با اجرای بهترین شبکه مدل شده روی داده های آزمون داده شده (50 داده) و label های آنها با عناوین EvolutionaryMLP_TwoOutput_Labels و MLP_OneOutput_Labels Evolutionary ذخیره و گزارش شده اند.

RBف تک خروجی:

در بهترین حالت، مقدار دقت به ازای 5 نورون و شعاع 15 و برابر 86٪ است:

```
best_neuron_number =  
    5  
  
best_spread =  
    15  
  
ans =  
    0.8636
```

RBف دو خروجی:

در بهترین حالت، مقدار دقت به ازای 50 نورون و شعاع 15 و برابر 67٪ است:

```
best_neuron_number =  
    50  
  
best_spread =  
    15  
  
ans =  
    0.6750
```

در انتهای هر دو بخش با اجرای بهترین شبکه مدل شده روی داده های آزمون داده شده (50 داده) و label های آنها با عنوان EvolutionaryRBF_TwoOutput_Labels و EvolutionaryRBF_OneOutput_Labels ذخیره و گزارش شده اند.