

گزارش مسائل سری سوم هوش محاسباتی

شیرین جعفرزاده خادم

۹۲۵۰۰۰۷

استاد: خانم موسوی

۹۵/۱۰/۲۵

استفاده از منطق فازی در رده‌بندی

برای این برنامه فایل program.m را نوشتیم که شامل توابع زیر است.

- program
- smallLength
- mediumLength
- largeLength
- smallWidth
- mediumWidth
- largeWidth
- FuzzyAND

تابع program تابع اصلی برنامه است و اجرای برنامه در آن قرار دارد. از تابع smallLength، تا تابع largeWidth همگی توابعی هستند که تابع عضویت فازی در آن‌ها پیاده سازی شده است. مثلاً تابع smallLength میزان کوچک بودن طول گلبرگ را برمی‌گرداند. تابع FuzzyAND نیز عملگر AND فازی را پیاده سازی کرده است.

با توضیح تابع program شروع می‌کنیم

۱. نمایش نمونه‌ها

ابتدا فایل iris2D.txt را باز می‌کنیم. اطلاعات درون این فایل به این صورت است.

pet_length	pet_width	Genre
1.4	0.2	Setosa
1.4	0.2	Setosa
1.3	0.2	Setosa
1.5	0.2	Setosa
...
5.1	1.8	Virginica

با استفاده از تابع fgets خط اول فایل را می‌خوانیم. این خط در جایی از برنامه به کار نمی‌آید.

```
fileID = fopen('iris2D.txt','r');  
names = fgets(fileID);
```

با استفاده از تابع textscan تمامی خطوط بعدی فایل را می‌خوانیم و اطلاعات آن‌ها را در یک لیست سه تایی از cell ها، برای هر ستون از داده‌ها یک سلول، ذخیره می‌کنیم.

```
Data = textscan(fileID, '%f %f %s');
x1 = Data{1};
x2 = Data{2};
label = Data{3};
fclose(fileID);
n = length(x1);
```

و سپس آن‌ها را در x1، x2 و label میریزیم. همین‌طور تعداد نمونه‌ها را در n قرار می‌دهیم.

Data{ [-] , [-] , [-] }

1.4	0.2	{Setosa}
1.4	0.2	{Setosa}
1.3	0.2	{Setosa}
1.5	0.2	{Setosa}
...
5.1	1.8	{Virginica}

حال با مقایسه‌ی label با نام هر یک از گونه‌ها اندیس نمونه‌های آن گونه را بدست می‌آوریم.

```
indxSetosa = strcmp(label, 'setosa');
indxVersicolor = strcmp(label, 'versicolor');
indxVirginica = strcmp(label, 'virginica');
```

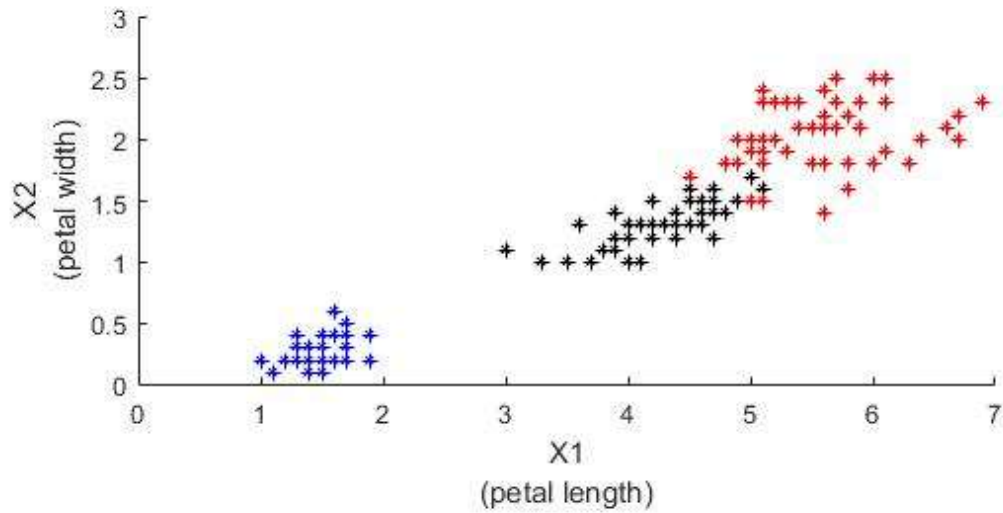
سپس با استفاده از اندیس‌ها، مقادیر نمونه‌های هر گونه را برای رسم کردن با رنگ متفاوت از هم جدا می‌کنیم.

```
x1Setosa = x1(indxSetosa);
x1Versicolor = x1(indxVersicolor);
x1Virginica = x1(indxVirginica);
x2Setosa = x2(indxSetosa);
x2Versicolor = x2(indxVersicolor);
x2Virginica = x2(indxVirginica);
```

سپس با استفاده از تابع scatter نقاط را رسم می‌کنیم. همین‌طور برچسب محورها و محدوده‌ی آن‌ها را نیز تنظیم می‌کنیم.

```
figure;
hold on;
scatter(x1Setosa, x2Setosa, 20, '*', 'blue');
scatter(x1Versicolor, x2Versicolor, 20, '*', 'black');
scatter(x1Virginica, x2Virginica, 20, '*', 'red');
xlabel({'X1', '(petal length)'});
ylabel({'X2', '(petal width)'});
axis equal;
axis([0 7 0 3]);
```

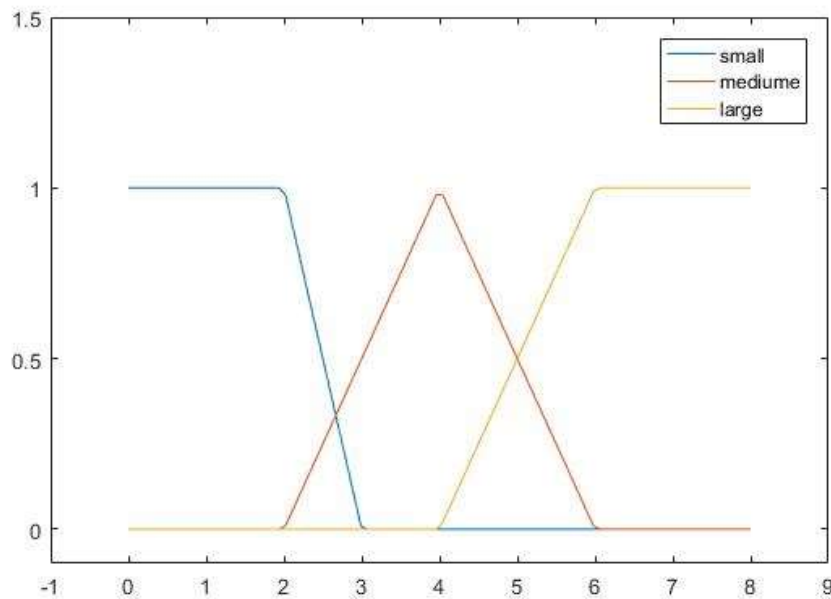
خروجی این بخش به این صورت است.



۲. توابع عضویت و قوانین فازی

با توجه به داده‌ها توابع عضویت را به صورت زیر تعریف می‌کنیم

توابع عضویت برای x_1 (طول گلبرگ)

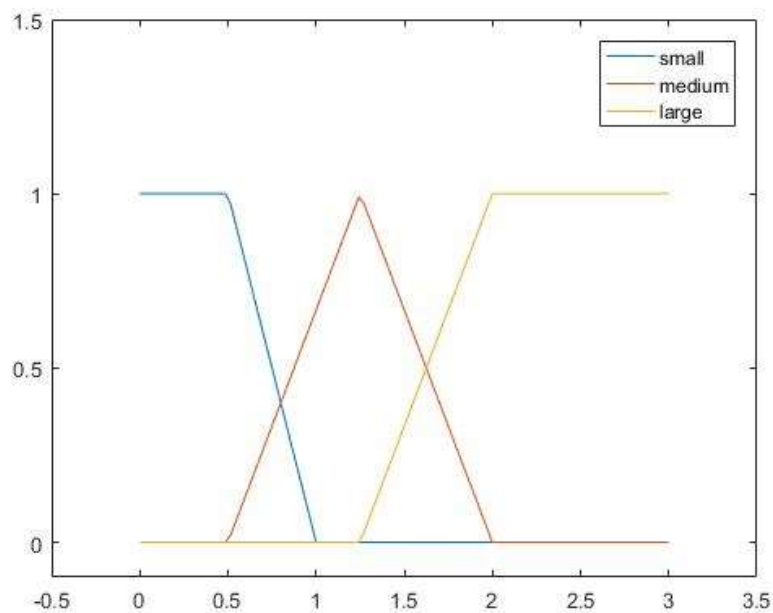


$$small(x) = \begin{cases} 1 & (0 \leq x < 2) \\ 3 - x & (2 \leq x < 3) \\ 0 & (x \geq 3) \end{cases}$$

$$medium(x) = \begin{cases} 0 & (0 \leq x < 2) \\ \frac{1}{2}x - 1 & (2 \leq x < 4) \\ 3 - \frac{1}{2}x & (4 \leq x < 6) \\ 0 & (x \geq 6) \end{cases}$$

$$large(x) = \begin{cases} 0 & (0 \leq x < 4) \\ \frac{1}{2}x - 2 & (4 \leq x < 6) \\ 1 & (x \geq 6) \end{cases}$$

توابع عضویت برای x_2 (عرض گلبرگ)



$$small(x) = \begin{cases} 1 & (0 \leq x < 2) \\ 3 - x & (2 \leq x < 3) \\ 0 & (x \geq 3) \end{cases}$$

$$medium(x) = \begin{cases} 0 & (0 \leq x < 2) \\ \frac{1}{2}x - 1 & (2 \leq x < 4) \\ 3 - \frac{1}{2}x & (4 \leq x < 6) \\ 0 & (x \geq 6) \end{cases}$$

$$large(x) = \begin{cases} 0 & (0 \leq x < 4) \\ \frac{1}{2}x - 2 & (4 \leq x < 6) \\ 1 & (x \geq 6) \end{cases}$$

حال با توجه به داده‌ها قوانین فازی را به این صورت تعریف می‌کنیم.

- اگر طول و عرض و گل‌برگ کوچک باشند، آن‌گاه گل زنبق متعلق به گونه‌ی **setosa** است.
- اگر طول و عرض و گل‌برگ متوسط باشند، آن‌گاه گل زنبق متعلق به گونه‌ی **Versicolor** است.
- اگر طول و عرض و گل‌برگ کوچک باشند، آن‌گاه گل زنبق متعلق به گونه‌ی **Virginica** است.

۳. اعمال توابع عضویت و قوانین فازی

در ادامه ابتدا خروجی توابع عضویت فازی را برای تمام نقاط حساب می‌کنیم. و در بردارهایی که در کد زیر تعریف شده‌اند ذخیره می‌کنیم.

```
smallLen = zeros(n, 1);
mediumLen = zeros(n, 1);
largeLen = zeros(n, 1);
smallWid = zeros(n, 1);
mediumWid = zeros(n, 1);
largeWid = zeros(n, 1);
for i=1:n
    smallLen(i) = smallLength(x1(i));
    mediumLen(i) = mediumLength(x1(i));
    largeLen(i) = largeLength(x1(i));
    smallWid(i) = smallWidth(x2(i));
    mediumWid(i) = mediumWidth(x2(i));
    largeWid(i) = largeWidth(x2(i));
end
```

حال با استفاده از این مقادیر هر سه قانون فازی را برای تمام نقاط حساب می‌کنیم در واقع میزان تعلق هر نمونه به هر گونه حساب می‌شود. برای ارزیابی این قانون‌ها به این صورت عمل می‌کنیم: چون هر سه قانون فقط **and** دو عبارت هستند، تابع **and** فازی را پیاده‌سازی کردیم و حاصل ارزیابی هر یک از قانون‌ها با اعمال این تابع بر روی خروجی توابع عضویت موجود در آن قانون بدست می‌آید. این مقادیر را برای تمام نقاط حساب کرده و در بردارهای تعریف شده در کد زیر ذخیره می‌کنیم.

```
SetosaValue = zeros(n, 1);
VersicolorValue = zeros(n, 1);
VirginicaValue = zeros(n, 1);
for i=1:n
    SetosaValue(i) = FuzzyAND(smallLen(i), smallWid(i));
    VersicolorValue(i) = FuzzyAND(mediumLen(i), mediumWid(i));
    VirginicaValue(i) = FuzzyAND(largeLen(i), largeWid(i));
end
```

۴. نمایش خروجی نهایی

حال این بردارها را در کنار هم قرار می‌دهیم، سپس با استفاده از تابع `max` قانونی که مقدار ارزیابی آن بیشتر بوده را برای هر نمونه بدست می‌آوریم. اگر قانون اول بود نمونه از گونه‌ی `setosa` است و الی آخر.

```
allValue = [SetosaValue, VersicolorValue, VirginicaValue];  
[maxValues, maxIndex] = max(allValue, [], 2);  
estimateIndxSetosa = (maxIndex == 1);  
estimateIndxVersicolor = (maxIndex == 2);  
estimateIndxVirginica = (maxIndex == 3);
```

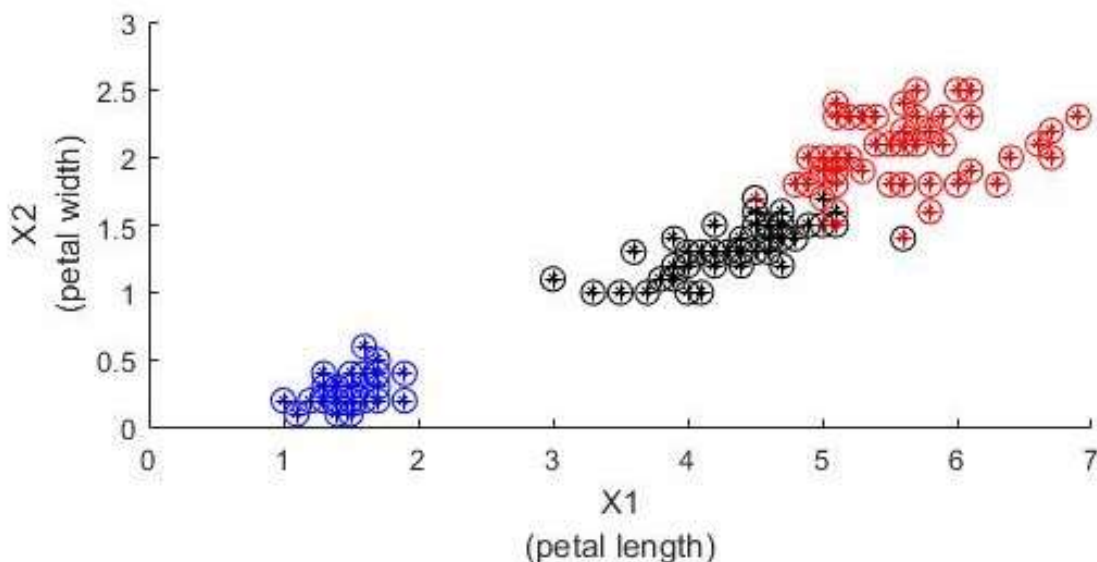
حال مشابه قبل نقاط با استفاده از اندیس‌ها، مقادیر نمونه‌های هر گونه را برای رسم کردن با رنگ متفاوت از هم جدا می‌کنیم.

```
x1Setosa = x1(estimateIndxSetosa);  
x1Versicolor = x1(estimateIndxVersicolor);  
x1Virginica = x1(estimateIndxVirginica);  
x2Setosa = x2(estimateIndxSetosa);  
x2Versicolor = x2(estimateIndxVersicolor);  
x2Virginica = x2(estimateIndxVirginica);
```

سپس با استفاده از تابع `scatter` نقاط را رسم می‌کنیم. این نقاط را به شکل دایره روی همان نمودار قبلی رسم می‌کنیم.

```
scatter(x1Setosa, x2Setosa, 70, 'o', 'blue');  
scatter(x1Versicolor, x2Versicolor, 70, 'o', 'black');  
scatter(x1Virginica, x2Virginica, 70, 'o', 'red');  
hold off;
```

در نهایت خروجی به این شکل می‌شود.



۵. محاسبه‌ی خطا

برای محاسبه‌ی تعداد نمونه‌هایی که اشتباه دسته‌بندی شده‌اند به این صورت عمل می‌کنیم که از روی گونه‌های تخمین زده شده یک بردار مانند بردار برچسب‌ها ایجاد می‌کنیم و با مقایسه‌ی برچسب‌های اصلی و تخمین زده شده خطا را محاسبه می‌کنیم.

```
estimateLabel = cell(n, 1);
estimateLabel(estimateIndxSetosa) = {'setosa'};
estimateLabel(estimateIndxVersicolor) = {'versicolor'};
estimateLabel(estimateIndxVirginica) = {'virginica'};
error = n - sum(strcmp(estimateLabel, label));
fprintf('error is : %d\n', error);
```

خروجی این قسمت به صورت زیر است:

```
error is : 7
```

۶. راه‌های پیشنهادی

اگر قوانین فازی را پیچیده‌تر کنیم می‌توانیم داده‌ها را دقیق‌تر دسته‌بندی کنیم. مثلاً برای گونه‌ی virginica بهتر است قانون به این صورت باشد: « یا طول و عرض گلبرگ بلند باشد و یا طول گل برگ خیلی بلند باشد ».