

گزارش پروژه درس شبکه‌های عصبی

پروژه‌ی اول

سید احمد نقوی نوزاد (۹۴۱۳۱۰۶۰)

دکتر رضا صفابخش

توجه: در ابتدا لازم به ذکر است که در این پروژه علاوه بر دسته‌بندی به روش آموزش، ارزیابی و تست داده‌ها از روشی شاید به نوعی ابداعی مبتنی بر رویکرد محاسبات یکپارچه‌ی برداری استفاده شده است که علاوه بر دقت بالا در محاسبه‌ی خطا از سرعت شدیداً بالاتری نیز نسبت به روش اول برخوردار است.

در این روش بجای تقسیم داده‌ها به سه دسته‌ی آموزشی، ارزیابی و تست، همگی داده‌ها را به یکباره در هر Epoch مورد ارزیابی قرار داده و از کلیه‌ی خطاها میانگین گرفته و با کمینه‌ی خطای تعریف‌شده توسط کاربر مقایسه می‌نمائیم و در صورت کم‌تر بودن خطای حاصل از خطای کمینه، محاسبات را خاتمه می‌دهیم.

منظور از محاسبات یکپارچه‌ی برداری این است که بجای محاسبه‌ی وزن‌های جدید برای هر نمونه و اعمال این وزن‌های حاصله در محاسبه‌ی وزن‌های جدید برای نمونه‌ی بعدی و ادامه‌ی این رویه تا انتهای مجموعه‌ی داده‌ها (Dataset) و به عبارتی پیشبرد مرحله به مرحله (gradual progress)، به یکباره تمامی ماتریس حاوی ویژگی‌های نمونه‌ها (مثلاً در سؤال اول یک ماتریس 4×150) را در ماتریس وزن اولیه‌ی تصادفی یا وزن حاصله از Epoch قبلی (مثلاً مطابق سؤال اول یک ماتریس 4×1) ضرب ماتریسی کرده و ماتریس حاصله را (در اینجا یک ماتریس 1×150) منهای مقدار بایاس تصادفی اولیه و یا بایاس به‌روزشده از Epoch قبلی نموده و در نتیجه مقادیر ورودی شبکه (net input) برای همگی نمونه‌ها همزمان محاسبه شده و در فرمول محاسبه‌ی وزن جدید اعمال می‌گردد و به عبارتی به ازای یک ماتریس وزن اولیه ورودی به هر Epoch برای تمامی نمونه‌ها به طور همزمان یک ماتریس وزن جدید و به دنبال آن یک ماتریس خطای حاصله از اعمال وزن‌های ورودی (در اینجا یک ماتریس 1×150) به دست می‌آید که همانطور که پیش از این نیز قید شد میانگین این خطا با کمینه‌ی خطای تعریف‌شده از قبل مقایسه گشته و در صورت کم‌تر بودن شرط پایان محاسبات خواهد بود. علاوه بر این ماتریس مقادیر مطلوب (target value) نیز در این روش تغییر یافته و در مورد جداسازی دو کلاس از نمونه‌ها به جای کلاس‌های صفر و یک از کلاس‌های یک و منفی یک بهره برده‌ایم و در مورد تابع فعالساز نیز بجای استفاده از تابع hardlim، از تابع sign استفاده کرده و **در مورد سؤال سوم** هم ناچاراً نمونه‌ها را با کم کردن مختصات هر کدام از میانگین هر بُعد به سمت مبدأ شیفت می‌دهیم که سبب افزایش بازه مجاز تغییر برای وزن‌های وارده شده و البته در هنگام رسم نمونه‌ها و نمودارها شیفت معکوس انجام می‌دهیم و باید قید کرد که همگی این‌ها نقش بسزائی در شدت همگرائی وزن‌ها به یک مقدار مطلوب در جداسازی یا همان دسته‌بندی نمونه‌ها را دارند.

همانطور که در ادامه مشاهده خواهید کرد برتری این روش از لحاظ سرعت انجام محاسبات سنگین ناشی از اعمال Epoch‌های بسیار زیاد (مثلاً در سؤال ۱ حدود $30,000$ Epoch طی حدود ۲۳ ثانیه) و البته بار محاسباتی و مصرف حافظه‌ی بسیار کمتر، کاملاً با روش معمول مطلوب مسئله (آموزش، ارزیابی و تست) قابل تفکیک خواهد بود.

سوال اول:

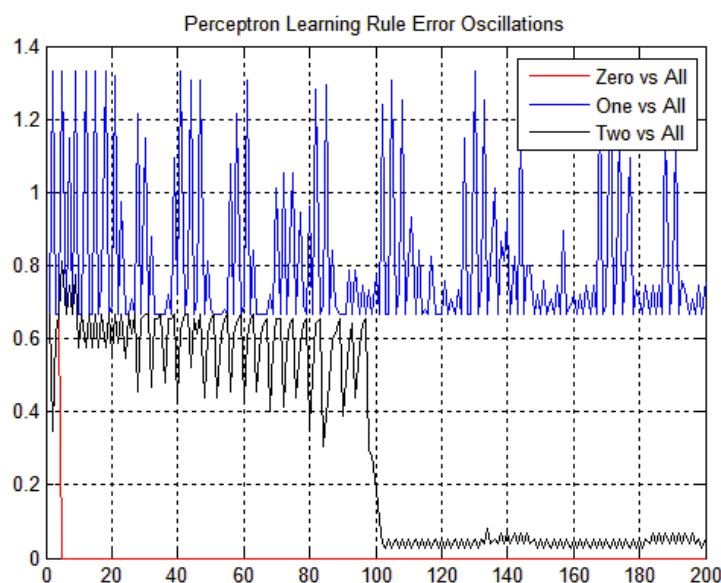
✓ { رویکرد محاسبات یکپارچه‌ی برداری }

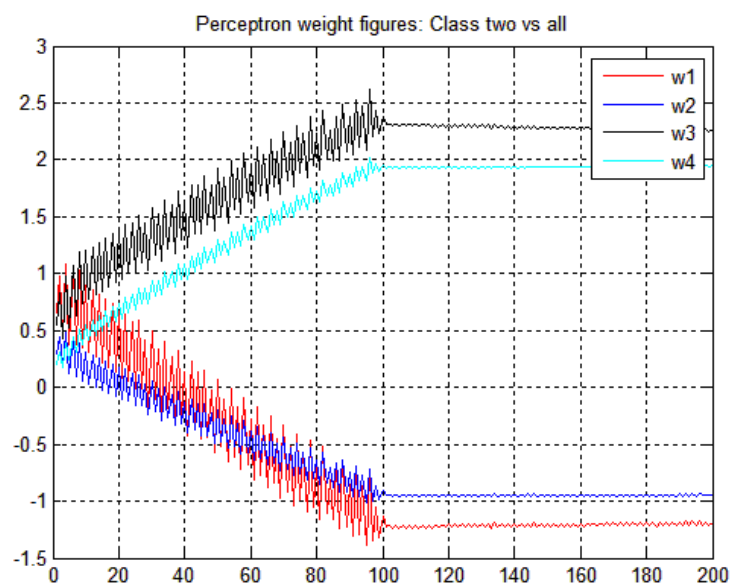
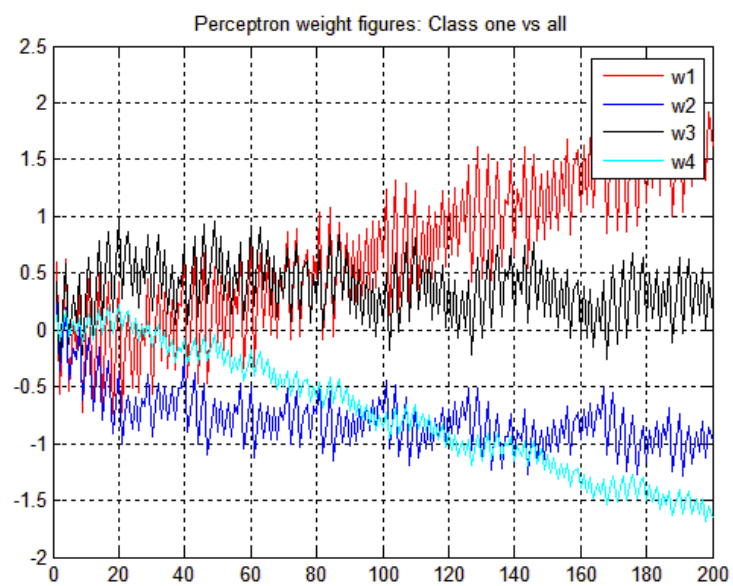
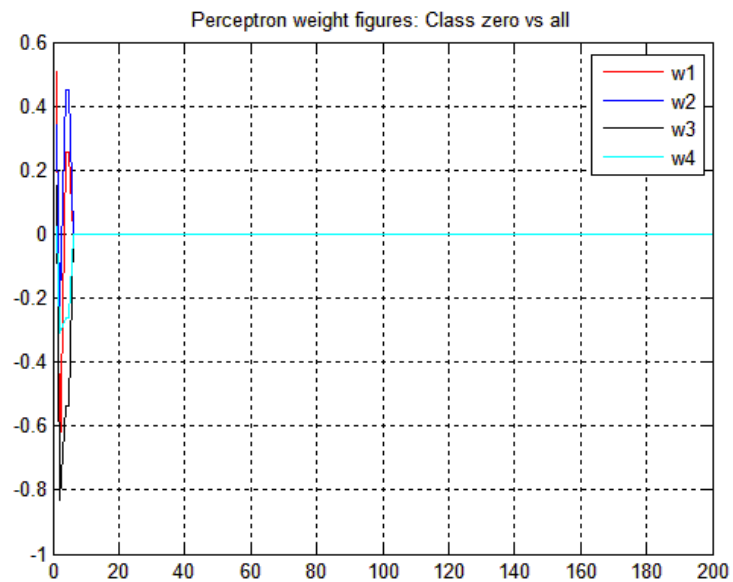
۱. شرایط آزمایش:

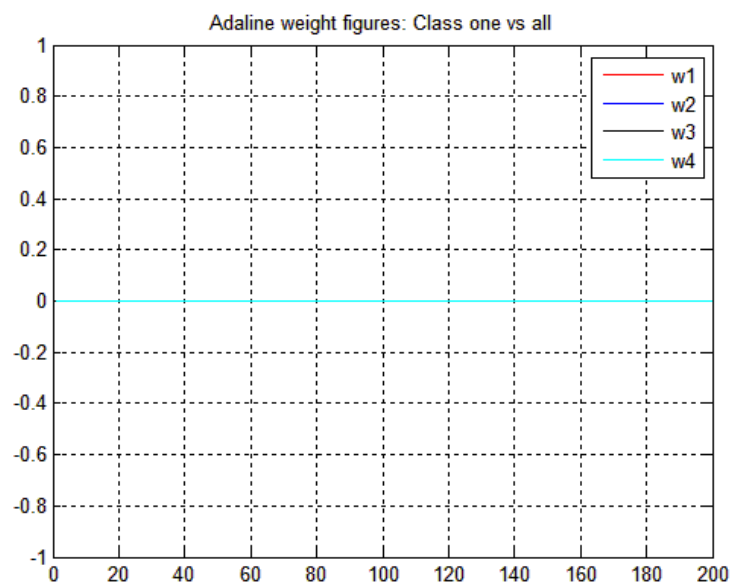
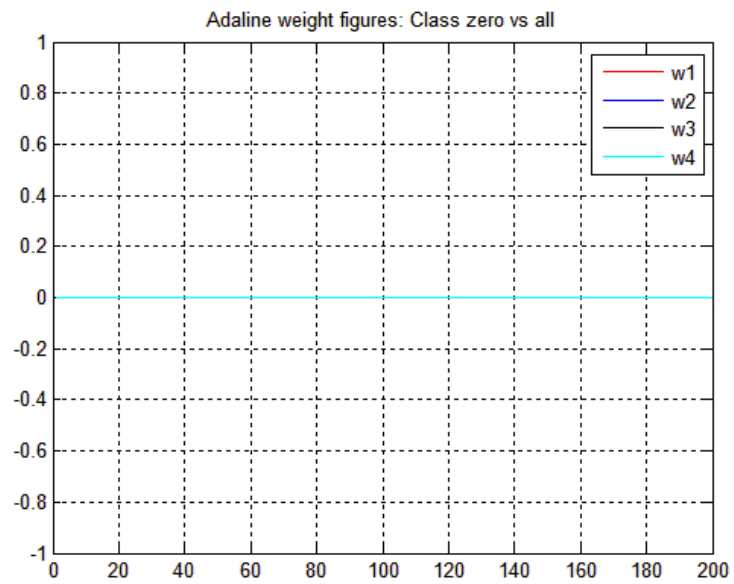
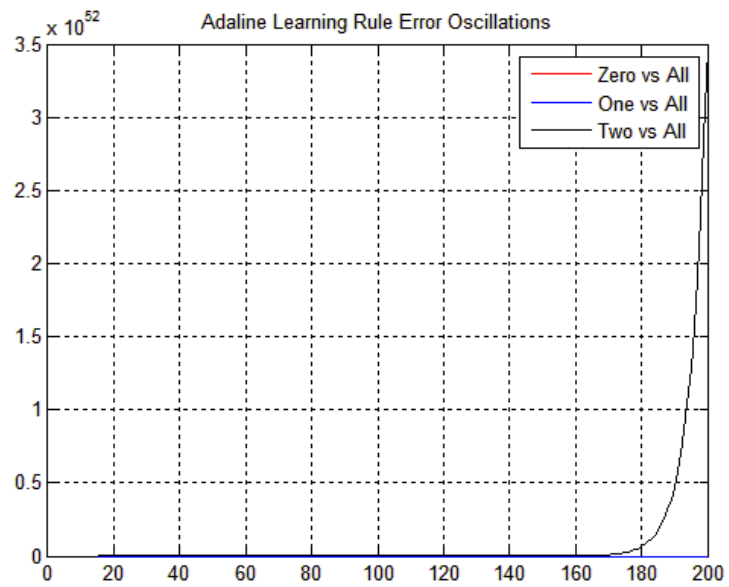
| No. | Approach Type | X vs Y | Initial Weights | Initial Threshold (Bias) | maxEpoch | minError | Learning Rate | Cessation Reason | Epoch No. | Error Quantity |
|-----|---------------|--------|-----------------------------|--------------------------|----------|----------|---------------|----------------------|-----------|----------------|
| 1 | Perceptron | 0vsAll | 0.0027 0.0018 0.0048 0.0019 | 0.6643 | 300 | 1e-7 | 0.1 | ErrorMean < minError | 11 | 0 |
| | | 1vsAll | 0.0060 0.0055 0.0026 0.0033 | | | | | epochNo > maxEpoch | 301 | 0.7200 |
| | | 2vsAll | 0.0046 0.0056 0.0041 0.0038 | | | | | epochNo > maxEpoch | 301 | 0.0933 |
| | Adaline | 0vsAll | 0.0022 0.0030 0.0048 0.0059 | 0.1228 | 300 | 1e-7 | 0.01 | epochNo > maxEpoch | 301 | NaN |
| | | 1vsAll | 0.0048 0.0001 0.0045 0.0029 | | | | | epochNo > maxEpoch | 301 | NaN |
| | | 2vsAll | 0.0029 0.0008 0.0054 0.0022 | | | | | epochNo > maxEpoch | 301 | NaN |
| 2 | Perceptron | 0vsAll | 0.0034 0.0010 0.0037 0.0000 | 0.9081 | 300 | 1e-4 | 0.1 | ErrorMean < minError | 10 | 0 |
| | | 1vsAll | 0.0051 0.0057 0.0061 0.0066 | | | | | epochNo > maxEpoch | 301 | 1.2666 |
| | | 2vsAll | 0.0034 0.0018 0.0007 0.0034 | | | | | epochNo > maxEpoch | 301 | 0.1200 |
| | Adaline | 0vsAll | 0.0039 0.0051 0.0006 0.0044 | 0.1080 | 300 | 1e-4 | 0.01 | epochNo > maxEpoch | 301 | NaN |
| | | 1vsAll | 0.0034 0.0011 0.0063 0.0039 | | | | | epochNo > maxEpoch | 301 | NaN |
| | | 2vsAll | 0.0029 0.0063 0.0044 0.0030 | | | | | epochNo > maxEpoch | 301 | NaN |
| 3 | Perceptron | 0vsAll | 0.0037 0.0045 0.0024 0.0016 | 0.8397 | 200 | 1e-3 | 0.01 | ErrorMean < minError | 7 | 0 |
| | | 1vsAll | 0.0039 0.0058 0.0027 0.0008 | | | | | epochNo > maxEpoch | 201 | 1.0266 |
| | | 2vsAll | 0.0030 0.0020 0.0027 0.0056 | | | | | epochNo > maxEpoch | 201 | 0.0933 |
| | Adaline | 0vsAll | 0.0027 0.0026 0.0024 0.0009 | 0.5326 | 200 | 1e-3 | 0.001 | epochNo > maxEpoch | 201 | 2.2944e+169 |
| | | 1vsAll | 0.0017 0.0006 0.0029 0.0017 | | | | | epochNo > maxEpoch | 201 | NaN |
| | | 2vsAll | 0.0020 0.0028 0.0008 0.0033 | | | | | epochNo > maxEpoch | 201 | NaN |
| 4 | Perceptron | 0vsAll | 0.0052 0.0005 0.0026 0.0000 | 0.7064 | 200 | 1e-3 | 0.001 | ErrorMean < minError | 5 | 0 |
| | | 1vsAll | 0.0015 0.0000 0.0013 0.0009 | | | | | epochNo > maxEpoch | 201 | 0.7600 |
| | | 2vsAll | 0.0018 0.0012 0.0009 0.0040 | | | | | epochNo > maxEpoch | 201 | 0.0533 |
| | Adaline | 0vsAll | 0.0060 0.0063 0.0015 0.0032 | 0.2436 | 200 | 1e-3 | 0.0001 | epochNo > maxEpoch | 201 | 4.9607e+16 |
| | | 1vsAll | 0.0025 0.0035 0.0018 0.0005 | | | | | epochNo > maxEpoch | 201 | 4.1576e+34 |
| | | 2vsAll | 0.0029 0.0012 0.0002 0.0064 | | | | | epochNo > maxEpoch | 201 | 3.4845e+52 |

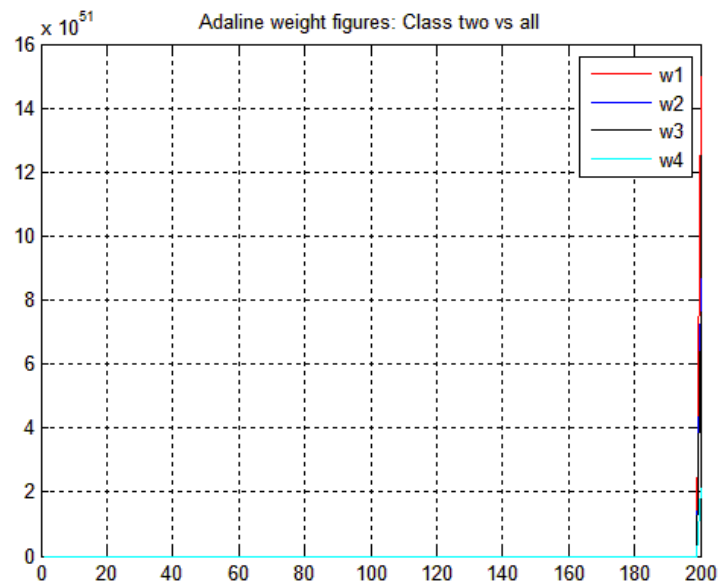
۲. نتیجه‌ی انجام آزمایش:

در اینجا فقط برای مورد آخر نمودارهای مورد نیاز قرار گرفته و توضیحات لازم نیز در ادامه می‌آید:









۳. نتیجه گیری:

همانطور که جداول و نمودارها نشان می‌دهند، روش یادگیری پرسپترون در این مسئله بسیار سریعتر از آدالاین به جواب رسیده و کلاس صفر را از مابقی کلاس‌ها سریعاً جداسازی می‌نماید که این مسئله از نمودار خطای مسئله قابل برداشت است و این در حالی است که روش یادگیری آدالاین طبق رویکرد محاسبات یکپارچه‌ی برداری شدیداً از پاسخ نهائی واگرا شده و قادر به یافتن پاسخ حتی در Epoch های بالاتر نیز نمی‌باشد.

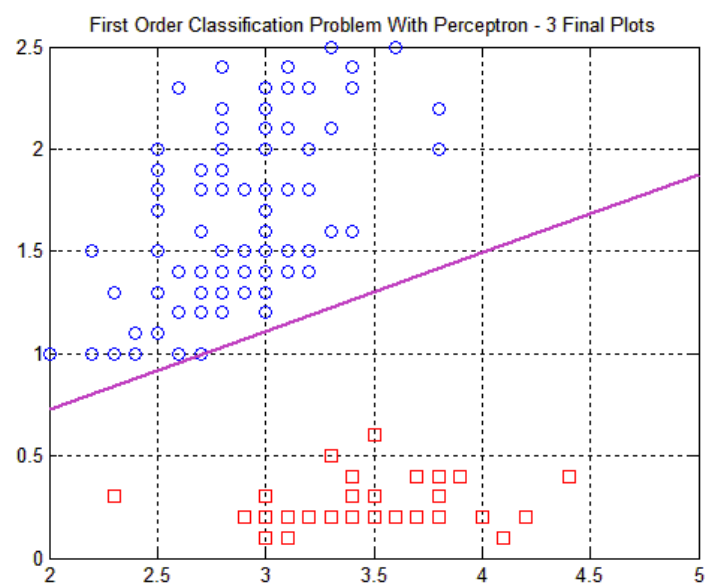
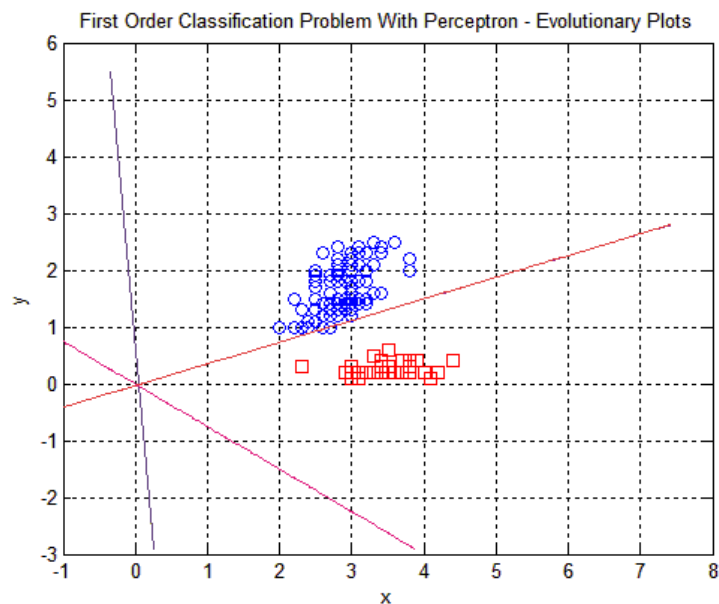
سوال دوم:

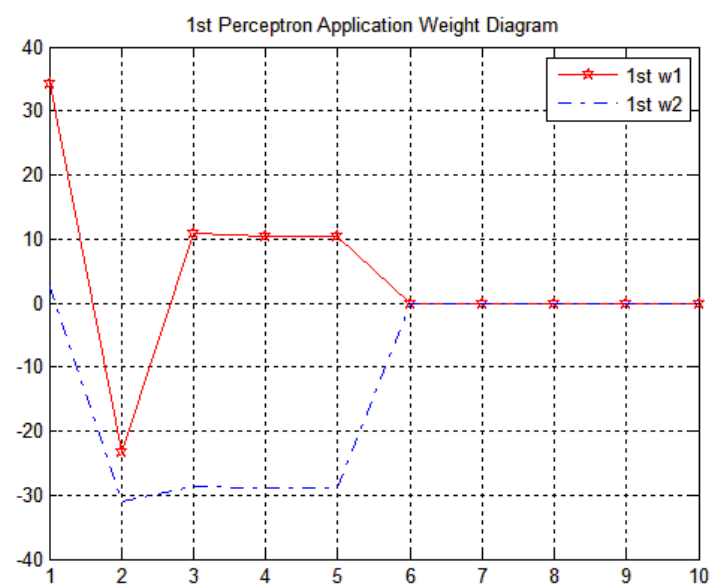
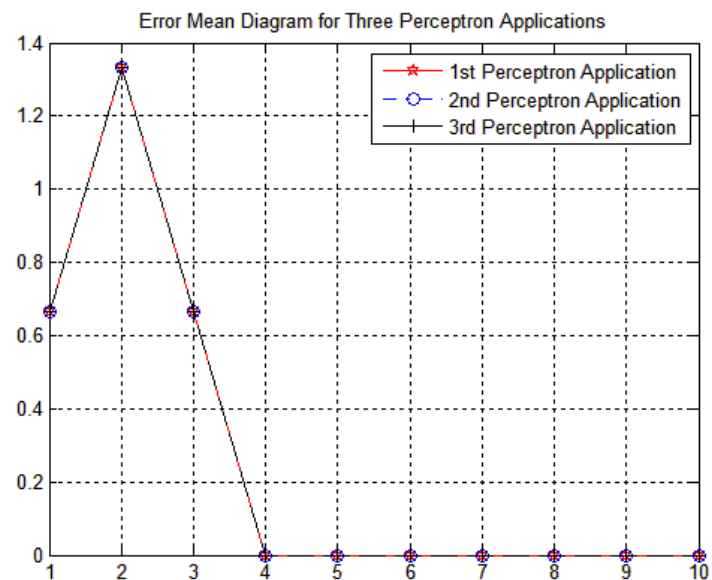
{ رویکرد محاسبات یکپارچه‌ی برداری }

۱. شرایط آزمایش:

| No. of Perceptron Application | Initial Weights | Initial Threshold (Bias) | maxEpoch | minError | Learning Rate | Stopping Epoch No. | Before Last Epoch Error Quantity |
|-------------------------------|-----------------|--------------------------|----------|----------|---------------|--------------------|----------------------------------|
| 1 | 0.0033 0.0051 | 0.2874 | 10 | 1e-7 | 0.1 | 5 | 0.0133 |
| 2 | 0.0035 0.0019 | 0.2874 | 10 | 1e-7 | 0.1 | 5 | 0.0133 |
| 3 | 0.0039 0.0008 | 0.2874 | 10 | 1e-7 | 0.1 | 5 | 0.0133 |

۲. نتیجه‌ی انجام آزمایش:





۳. نتیجه گیری:

در سه بار اجرای پرسپترون نتایج شدیداً مشابهی حاصل می شود که حکایت از شدیداً همگرا بودن رویکرد فعلی به جواب نهائی دارد.

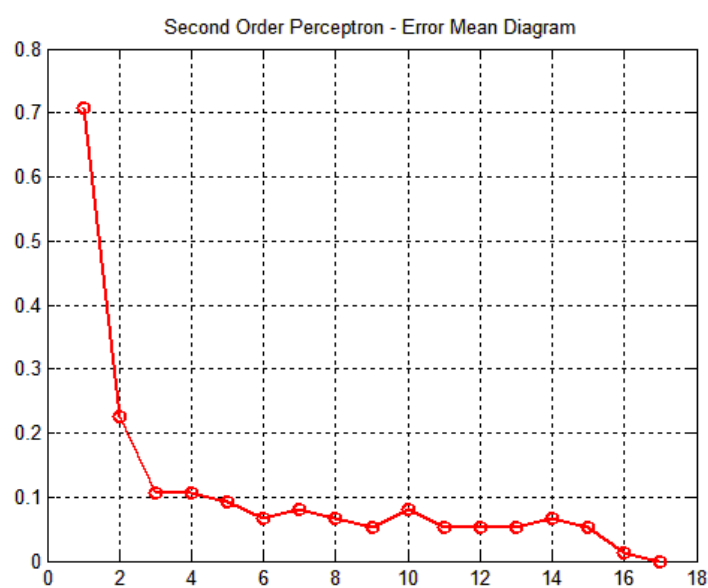
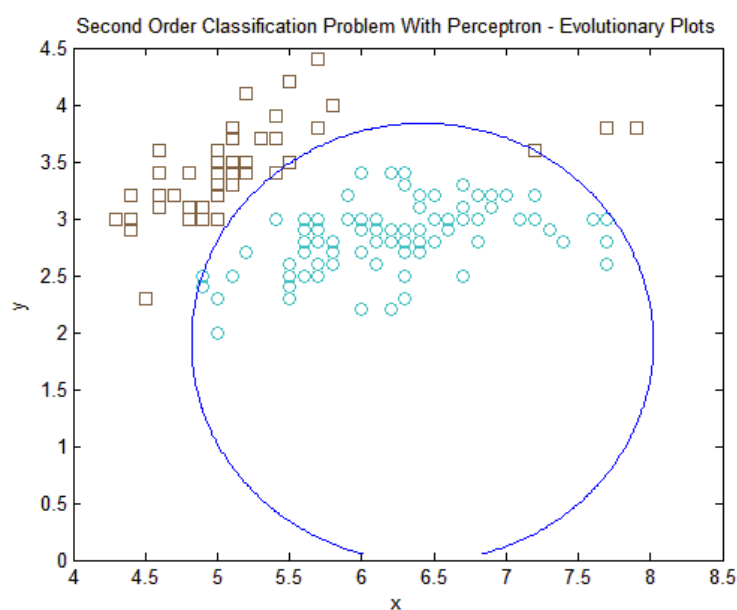
سوال سوم:

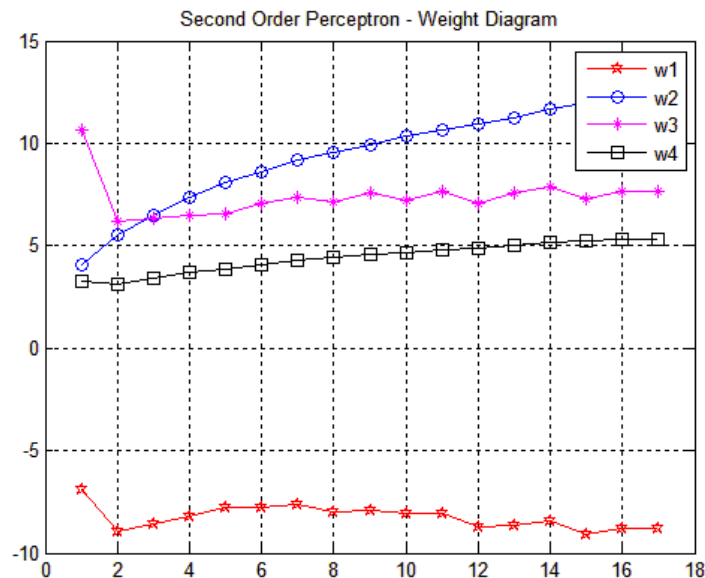
{ رویکرد محاسبات یکپارچه‌ی برداری }

۱. شرایط آزمایش:

| No. | Initial Weights | Initial Threshold (Bias) | maxEpoch | minError | Learning Rate | Stopping Epoch No. | Before Last Epoch Error Quantity |
|-----|-----------------------------|--------------------------|----------|----------|---------------|--------------------|----------------------------------|
| 1 | 0.0361 0.6181 0.5671 0.9620 | 10 | 3000 | 1e-5 | 0.1 | 20 | 0.0133 |
| 2 | 0.7461 0.6625 0.5233 0.2599 | 10 | 300 | 1e-5 | 0.1 | 17 | 0.0133 |
| 3 | 0.9620 0.5402 0.0303 0.6963 | 10 | 300 | 1e-7 | 0.1 | 17 | 0.0400 |

۲. نتیجه‌ی انجام آزمایش:





۳. نتیجه‌گیری:

از آنجا که تمامی محاسبات به صورت یکپارچه صورت می‌گیرد در نتیجه سرعت انجام کار حتی در Epoch های بالا شدیداً زیاد بوده و همانطور که از تصاویر مشهود است دقت دسته‌بندی نیز عملاً ۱۰۰ درصد می‌باشد. از آنجا که نرخ خطا در این روش با سرعت بالائی افت نموده و البته شرط توقف با خطا نیز در کد برنامه تعبیه شده است، لذا با اعمال پارامترهای مختلف نتایج مشابهی حاصل شده و مطلوب مسئله در عرض حدود ۲۰ Epoch و یا حتی کمتر به دست می‌آید.

نکته:

با عرض پوزش از حضور استاد عالی‌رتبه و البته تدریس‌یار محترم درس، اینجانب به دلیل مسائلی چون پیچیدگی فهم اولیه‌ی مسئله و البته عدم آشنائی اولیه‌ی کافی با نرم‌افزار MATLAB و نیز سنگینی وجود درس دیگری چون SML و در نتیجه‌ی آن ابتلای به بیماری، موفق به طراحی ظاهر گرافیکی برای همگی سؤالات نشدم و البته مورد طراحی شده نیز مطلوب نمی‌باشد و تأخیر فعلی نیز ناشی از همین موضوع می‌باشد. باشد که در پروژه‌های بعدی جبران گردد.

سپاسگزارم