جلسه دوم:

جلسه سوم:

اگر تابع فعالیت یکنوا افزایشی باشد به این صورت کاری میکنیم که خروجی ما بیشتر بشود یعنی اگر خروجی درست ما 1 بوده و خروجی ما کمتر بوده مثلا -1 کمک میکند بیشتر شود و همگرا میشود و بدین ترتیب تغییر وزن ها باعث کاهش خطا میشود. فرض جدایی پذیر خطی مهم هست در استفاده از پرسپترون. اگر وزن بهینه را در بیاوریم یعنی معادله خط یا مرز تصمیم گیری را بدست آورده ایم حالا بستگی به بعد هم دارد.

نحوه محاسبه مثال:

وزن را که ابتدا صفر قرار دادیم نظیر به نظیر با ورودی ضرب میکنیم از تابع فعالیت عبور میدهیم اگر خروجی با خروجی واقعی یکی بود که هیچی اگر نبود اپدیت میکنیم، چطوری؟ ورودی را در خروجی واقعی ضرب میکنیم بعلاوه یا منهای وزن فعلی میکنیم.

قضیه همگرا اگر مسئله جدایی پذیر خطی باشد در صورت وجود وزن ها مسئله به همگرایی میرسد و پاسخ همگرا خواهد بود.

Adaline تابع خروجی آن تابع خطی است چرا؟ در LMS میگفت میزان مقدار خطا مهم هست و اگر خطا بالا بود اپدیت بیشتر باشد بر خلاف پرسپترون که میزان خطا تاثیری نداشت و در نتیجه در LMS همگرایی سریع تر هست. Adaline تابع خطی دارد دیگر -1 و +1 نیست چرا؟ وقتی میخواهد تاثیر خطا در نظر بگیرد دیگر تابع نباید دو سطحی یا این یا اون باشد باید میزان خطا را در نظر بگیرد پس خروجی خطی هست. در Adaline از یک تابع فعالیت خطی رد میشود با وجود اینکه خروجی آن دو سطحی هست تنها نکته اینجا هست که بر خلاف پرسپترون اینجا مهم میشود که چه قدر خطا داریم و فقط داشتن یا نداشتن خطا مهم نیست.