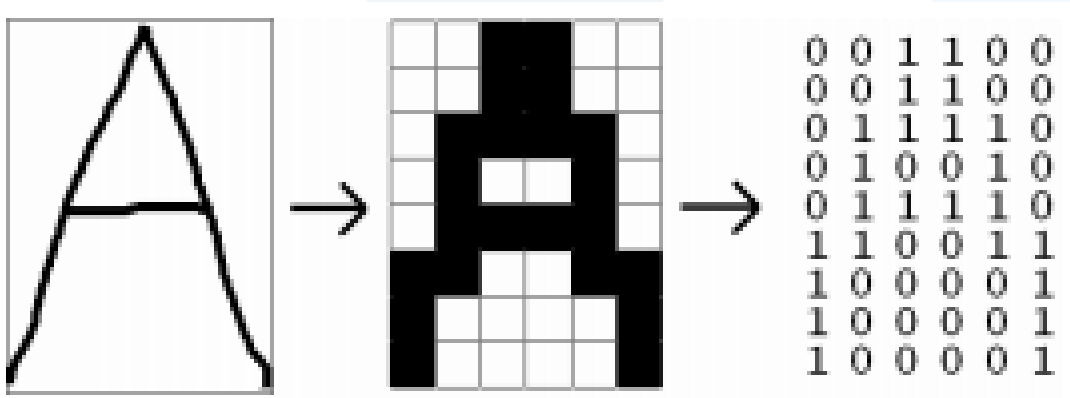
ذكاء صنعي 2

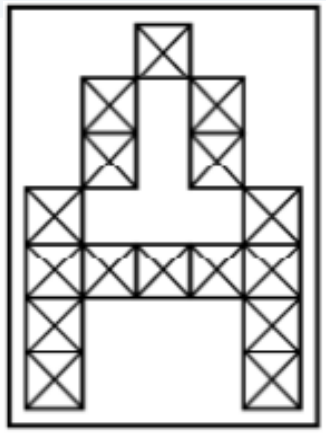
جلسة عملي/5/

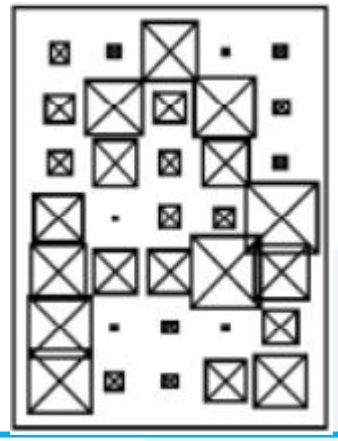
د.فادي متوج



المسألة:

- نريد تصميم شبكة عصبونية للتعرف على الأحرف الأبجدية الإنكليزية ال26.
- يتم تحجيم كل صورة دخل إلى 7 أسطر و 5 أعمدة، ثنائية القيم.
- كل حرف مكون من مصفوفة 5×7 (35 عنصر).





- ليست صور كل الأحرف مثالية، قد تعاني بعض الصور من التشويش.

المسألة-تابع:

- نرتب كل مصفوفة 5×7 كشعاع ببعد وحيد مؤلف من 35 عنصر فيصبح لدينا 26 شعاع، كل شعاع يعبر عن حرف أبجدي محدد.
- هذه الأشعة معرفة في التابع prprob كمصفوفة دخل اسمها alphabet.
- تعرف أشعة الخرج بالاسم target، وهي عبارة عن 26 شعاع كل شعاع يحوي 26 قيمة، 25 قيمة منها صفرية، وقيمة فقط 1 هي رقم الحرف.
- شعاع الخرج الموافق للحرف A: الرقم 1 في الخانة الأولى، الخانات 25 المتبقية أصفار.

الشبكة العصبونية:

- تستقبل الشبكة العصبونية قيم الدخل ال35 لكل شعاع كمداخل، ثم تقوم بتحديد الخرج المناسب لهذه المداخل.
- اي أن الشبكة يجب أن تستجيب عند شعاع الخرج الموافق للحرف الصحيح فقط، وتعطي استجابة صفرية عند جميع الأحرف المتبقية.
- في البداية نستدعي التابع prprob بمداخله ومخارجه.

```
[alphabet,targets] =prprob;

P = alphabet;
```

1- إنشاء الشبكة:

- شبكة عصبونية مكونة من طبقة مخفية وحيدة (10 عصبونات) إضافة لطبقة الخرج (26 عصبون).
- تابع التفعيل لكل من الطبقتين هو log-sigmoid اللوغاريتم القطعي، خيار جيد بما أن مجال قيمه يتراوح بين 0 و 1 فهو مناسب لتطبيقنا الحالي.
- نستخدم إحدى الطريقتين التاليتين:

```
net = newff(minmax(P),[10 26],{'logsig' 'logsig'},'traingdx');

net=patternnet(10,'traingdx');
net.layers{1}.transferFcn='logsig';
net.layers{2}.transferFcn='logsig';
```

2- تدريب الشبكة:

- لتصبح الشبكة قادرة على التعامل مع أشعة دخل مثالية أو مشوشة يجب تدريبها **بعدة مراحل**.

- في البداية ندرب الشبكة على اشعة الدخل المثالية (بدون تشويش)، بعدد مرات كافٍ (5000) للوصول إلى قيمة خطأ (وفق معيار sse) لا تتجاوز 0.1:

```
T = targets;
net.performFcn = 'sse';
net.trainParam.goal = 0.1;
net.trainParam.epochs = 5000;
[net,tr] = train(net,P,T);
```

- ملاحظة: يمكن أيضاً تغيير بارامتر معدل التعلم lr أو بارامتر ثابت الزخم mc (قيم كل منهما بين 0 و 1).

2- تدريب الشبكة - تابع:

- الخطوة التالية هي تدريب الشبكة لـ10 **مرات**، في كل مرة على مجموعتي بيانات مثالية إضافة إلى مجموعتي بيانات مشوشة مختلفة عن التشويش السابق، وفق البارامترات التالية:

```
net.trainParam.goal = 0.6;
net.trainParam.epochs = 300;
T = [targets targets targets targets]
 for pass = 1:10
 P = [alphabet ,alphabet, (alphabet+randn(35,26)*0.1), (alphabet+ randn(35,26)*0.2)];
 [net,tr] = train(net,P,T);
 end
```

- الهدف من التدريب بمجموعتي البيانات المشوشة هو تعليم الشبكة وضمان قدرتها على التعامل مع التشويش، بينما التدريب على مجموعتي البيانات المثالية هدفه الحفاظ على قدرتها الأساسية (التعامل مع بيانات غير مشوشة).

2- تدريب الشبكة-تابع:

- الخطوة التالية هي تدريب الشبكة مرةً أخرى على بيانات مثالية غير مشوشة لضمان قدرتها على تمييز المحارف المثالية في حال عدم وجود تشويش، وفق البارامترات التالية:

```
P = alphabet;
T = targets;
net.trainParam.goal = 0.1;
net.trainParam.epochs = 5000;
[net,tr] = train(net,P,T);
```

3- اختبار الشبكة:

- لاختبار كفاءة عمل الشبكة نختار على سبيل المثال الحرف الأبجدي العاشر J. ونضيف له تشويش عشوائي بنسبة 20%، ثم نختبر الشبكة عنده.

```
noisyJ = alphabet(:,10)+randn(35,1)*0.2;
A1 = sim(net,noisyJ);
```

- نستخدم التعليمة compet لانتخاب أعلى رقم من الشعاع الناتج عن الاختبار وحذف البقية، أي لتقريب خرج الشبكة إلى رقم حرف ابجدي محدد، ثم نبحث عن الشعاع الموافق لهذا الرقم ونرسمه لنقارن دخل الاختبار مع نتيجة الشبكة.

```
A2 = compet(A1);
answer = find(compet(A2) == 1);
figure, subplot(1,2,1), plotchar(noisyJ)
        subplot(1,2,2), plotchar(alphabet(:,answer))
```

نتيجة الاختبار: