Backpropagation Algorithm & RBF

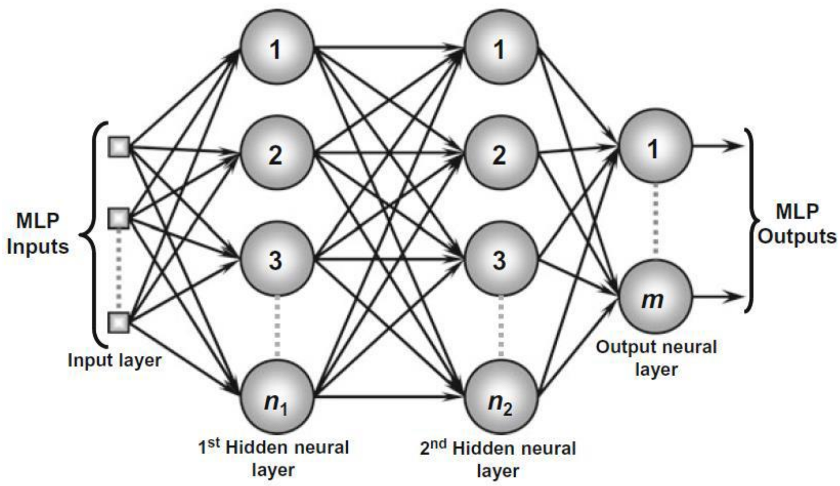
د. ياسر خضرا

مقدمة:

- في السابق كنا نقوم بتدريب شبكات عصبونية بسيطة، تحتوي على عصبون واحد أو طبقة واحدة مثلاً، حيث كُنّا نقوم بضرب الدخل بالأوزان وإضافة الانزياح ومن ثم ادخلنا القيمة إلى تابع التفعيل وقمنا بالتجريب على بعض الدارات المنطقية، وكانت تدعى هذه العملية ب Forward propagation.
- ولكن معظم المشاكل تكون أكثر تعقيداً وتحتاج إلى شبكات أكثر عمقاً (أي تحتوي على طبقات بينية)، ممكن أن تكون الطبقات البينية عبارة عن طبقة واحدة وممكن أن تصل إلى مئات الطبقات.
- أي أنه لم يعد التدرب باتجاه واحد كافياً (أي من الدخل إلى الخرج)، وإنما نحن بحاجة إلى التعديل على الأوزان حسب تأثير كل وزن، انطلاقاً من الخرج إلى الدخل.
- مثال:

في المثال التالي لدينا Multiplayer perceptron (MLP)، يحتوي على دخل وطبقتين مخفيتين وطبقة خرج.

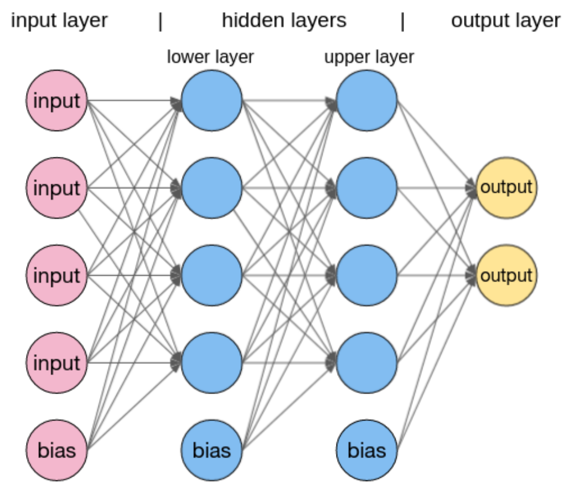
ملاحظة: أغلب المراجع لا تعتبر الدخل طبقة.



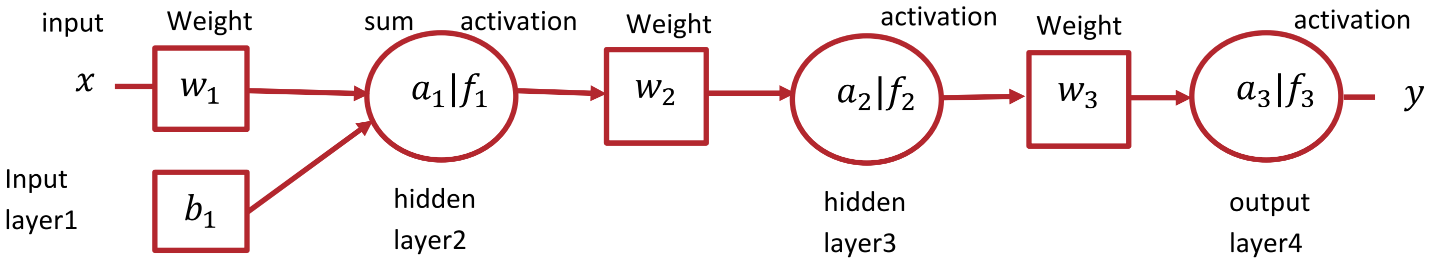
مبدأ عمل ال Multiplayer perceptron

على خلاف ال Simple perceptron و الADALINE فإن ال MLP ممكن أن تكون طبقة الخرج فيه تحتوي على العديد من العصبونات.

Fully connected: كل عصبون مرتبط مع جميع العصبونات الأخرى.



❖ حساب الخرج النهائي وخرج كل عصبون:



شرح الدلالات: نأخذ العقدة الأولى مثلاً:

x: تمثل الدخل

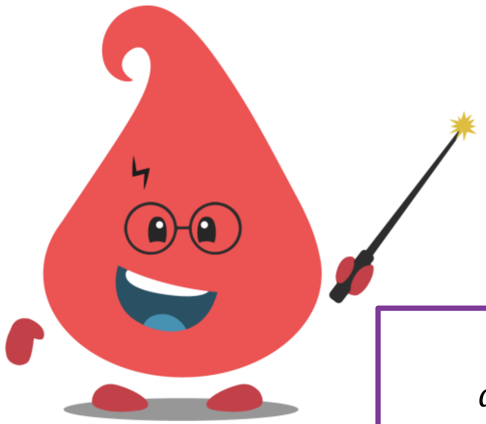
w_1: وزن الطبقة الأولى

b_1: الانزياح

a_1: كنا نسميها سابقاً υ وتمثل تابع الجمع

f_1: تابع التفعيل ويمثل خرج العصبون الأول

ويتم حساب توابع الجمع بطريقة تراكمية كالآتي:



$$a_1 = w_1 \,.\, x + b_1$$
$$a_2 = w_2 \,.\, f_1\,(w_1 \,.\, x + b_1)$$
$$a_3 = w_3 \,.\, f_2(w_2 \,.\, f_1\,(w_1 \,.\, x + b_1))$$

ويمثل الخرج النهائي نتيجة ضرب خرج الطبقة السابقة مضروبة بوزن الطبقة الأخيرة، ومن ثم إدخال هذه النتيجة إلى تابع التفعيل.

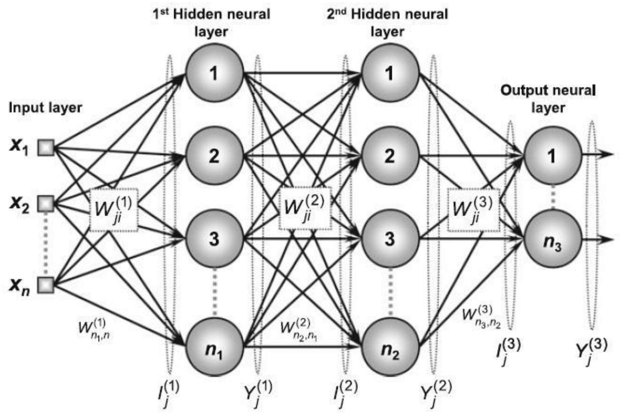
فيكون الخرج كالتالي:

$$\boxed{g = f_3\,(w_3\,.\,f_2(w_2\,.\,f_1\,(w_1\,.\,x + b_1)))}$$

- التدريب في هذه الشبكة سوف يتم عن طريق معرفة الخرج الحالي والخرج المثالي (لأنها supervise) ومن خلال حساب الخطأ الناتج بينهم القيام بعملية تعديل الأوزان بشكل عكسي، سيتم معرفة تأثير كل وزن.
- إذا أردنا دراسة تأثير العلاقات السابقة بالخرج، نلاحظ أن w_3 تؤثر بشكل مباشر على الخرج، w_2 تؤثر ولكن بعد تأثير w_3 وهكذا w_1 تؤثر ولكن بعد تأثير w_3, w_2 .
- إذاً عند تعديل الأوزان نبدأ بالوزن الأخير، وعند الانتهاء ننتقل إلى الوزن السابق، وهكذا حتى نصل إلى x وحساب تأثيرها بالعملية كاملة وتسمى هذه العلاقة بالانتشار الخلفي (Back propagation).

ملاحظة: الانزياح b قابل للتعديل أيضاً.

- عملية التدريب باستخدام Back propagation وشرح الرموز:



1. الأوزان: $W_{i,j}^{(l)}$

ويمثل المشبك الذي يربط العصبون j^{th} بالطبقة (L) مع العصبون j^{th} بالطبقة $(L-1)$.

2. الدخل: $I_j^{(L)}$

ويمثل دخل العصبون j^{th} في الطبقة (L).

$$I_j^{(1)} = \sum_{i=0}^{n} W_{i,j}^{(1)}\,.\,x_i \Leftrightarrow I_j^{(1)} = W_{1,0}^{(1)}\,.\,x_0 + W_{1,1}^{(1)}\,.\,x_1 + \cdots + W_{1,n}^{(1)}\,.\,x_n$$

وبنفس الطريقة نحسب $I_j^{(2)}$ مع ملاحظ اختلاف عدد العصبونات، وأن الدخل أصبح خرج الطبقة السابقة.

$$I_j^{(2)} = \sum_{i=0}^{n_1} W_{i,j}^{(2)}\,.\,y_i^{(1)} \Leftrightarrow I_j^{(2)} = W_{1,0}^{(2)}\,.\,\,y_0^{(1)} + W_{1,1}^{(2)}\,.\,y_1^{(1)} + \cdots + W_{1,n_1}^{(2)}\,.\,y_{n_1}^{(1)}$$

وبنفس الطريقة نحسب $I_j^{(3)}$

$$I_j^{(3)} = \sum_{i=0}^{n_2} W_{i,j}^{(3)}\,.\,y_i^{(2)} \Leftrightarrow I_j^{(3)} = W_{1,0}^{(3)}\,.\,\,y_0^{(2)} + W_{1,1}^{(3)}\,.\,y_1^{(2)} + \cdots + W_{1,n_2}^{(3)}\,.\,y_{n_2}^{(2)}$$

3. الخرج: $y_j^{(L)}$

ويمثل الخرج حيث أن j^{th} مرتبطة بترتيب الخرج للطبقة (L).

مثال:

$y_j^{(1)} = g(I_j^{(1)})$ هو خرج الطبقة الأولى، حيث تمثل ال g تابع التفعيل.

مثال عددي:

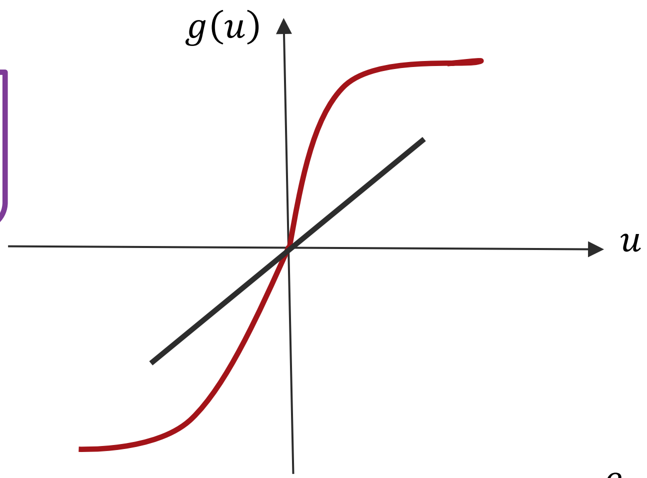
في البداية لدينا مصفوفات الأوزان:

$$W_{ji}^{(1)} = \begin{bmatrix} 0.2 & 0.4 & 0.5 \\ 0.3 & 0.6 & 0.7 \\ 0.4 & 0.8 & 0.3 \end{bmatrix}, \qquad W_{ji}^{(2)} = \begin{bmatrix} -0.7 & 0.6 & 0.2 & 0.7 \\ -0.3 & 0.7 & 0.2 & 0.8 \end{bmatrix},$$

$$W_{ji}^{(3)} = [0.1 \quad 0.8 \quad 0.5]$$

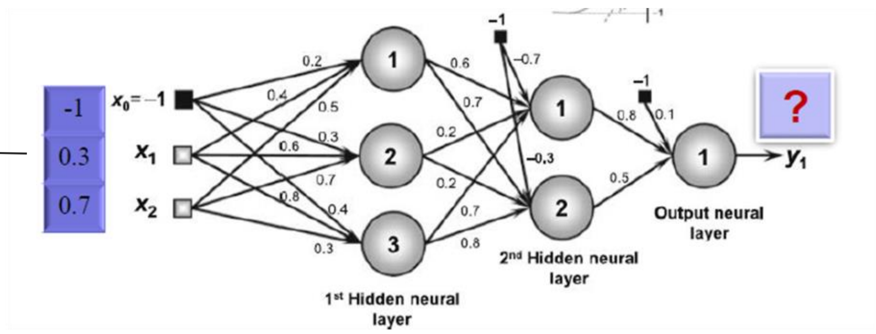
ولدينا تابع التفعيل:

$$\tan(u) = g(u) = \frac{1 - e^{-\beta u}}{1 + e^{-\beta u}}$$



نفرض $\beta = 1$

مصفوفة الدخل



في البداية نقوم بحساب مجموع مضروب الوزن بالدخل لكل عصبون في الطبقة الأولى:

$$I_j^{(1)} = \begin{bmatrix} I_1^{(1)} \\ I_2^{(1)} \\ I_3^{(1)} \end{bmatrix} = \begin{bmatrix} W_{1,0}^{(1)}.x_0 + W_{1,1}^{(1)}.x_1 + W_{1,2}^{(1)}.x_2 \\ W_{2,0}^{(1)}.x_0 + W_{2,1}^{(1)}.x_1 + W_{2,2}^{(1)}.x_2 \\ W_{3,0}^{(1)}.x_0 + W_{3,1}^{(1)}.x_1 + W_{3,2}^{(1)}.x_2 \end{bmatrix}$$

$$= \begin{bmatrix} 0.2\ (-1) + 0.4\ (0.3) + 0.5\ (0.7) \\ 0.3\ (-1) + 0.6\ (0.3) + 0.7\ (0.7) \\ 0.4\ (-1) + 0.8\ (0.3) + 0.3\ (0.7) \end{bmatrix} = \begin{bmatrix} 0.27 \\ 0.37 \\ 0.05 \end{bmatrix}$$

نقوم بحساب خرج الطبقة الأولى وذلك عن طريق إدخال القيم السابقة إلى تابع التفعيل:

$$Y_j^{(1)} = \begin{bmatrix} Y_1^{(1)} \\ Y_2^{(1)} \\ Y_3^{(1)} \end{bmatrix} = \begin{bmatrix} g(I_1^{(1)}) \\ g(I_2^{(1)}) \\ g(I_3^{(1)}) \end{bmatrix} = \begin{bmatrix} tanh(0.27) \\ tanh(0.37) \\ tanh(0.05) \end{bmatrix} = \begin{bmatrix} 0.26 \\ 0.35 \\ 0.05 \end{bmatrix} \xrightarrow{Y_0^{(1)}=-1} Y_j^{(1)} = \begin{bmatrix} Y_0^{(1)} \\ Y_1^{(1)} \\ Y_2^{(1)} \\ Y_3^{(1)} \end{bmatrix} = \begin{bmatrix} -1 \\ 0.26 \\ 0.35 \\ 0.05 \end{bmatrix}$$

في النهاية نقوم بإضافة الانزياح $Y_0^{(1)} = -1$ إلى خرج الطبقة الأولى، وذلك لنقوم بعدها باعتباره دخل للطبقة الثانية.

- وبنفس الطريقة نحسب خرج الطبقة الثانية:

$$I_j^{(2)} = \begin{bmatrix} I_1^{(2)} \\ I_2^{(2)} \end{bmatrix} = \begin{bmatrix} W_{1,0}^{(2)} \cdot y_0^{(1)} + W_{1,1}^{(2)} \cdot y_1^{(1)} + W_{1,2}^{(2)} \cdot y_2^{(1)} + W_{1,3}^{(2)} \cdot y_3^{(1)} \\ W_{2,0}^{(2)} \cdot y_0^{(1)} + W_{2,1}^{(2)} \cdot y_1^{(1)} + W_{2,2}^{(2)} \cdot y_2^{(1)} + W_{2,3}^{(2)} \cdot y_3^{(1)} \end{bmatrix} = \begin{bmatrix} 0.96 \\ 0.59 \end{bmatrix}$$

$$Y_j^{(2)} = \begin{bmatrix} Y_1^{(2)} \\ Y_2^{(2)} \end{bmatrix} = \begin{bmatrix} g(I_1^{(2)}) \\ g(I_2^{(2)}) \end{bmatrix} = \begin{bmatrix} tanh(0.96) \\ tanh(0.59) \end{bmatrix} = \begin{bmatrix} 0.74 \\ 0.53 \end{bmatrix} \xrightarrow{Y_0^{(2)}=-1} Y_j^{(2)} = \begin{bmatrix} Y_0^{(2)} \\ Y_1^{(2)} \\ Y_2^{(2)} \end{bmatrix} = \begin{bmatrix} -1 \\ 0.74 \\ 0.53 \end{bmatrix}$$

وبالنهاية نحسب خرج الطبقة الثالثة:

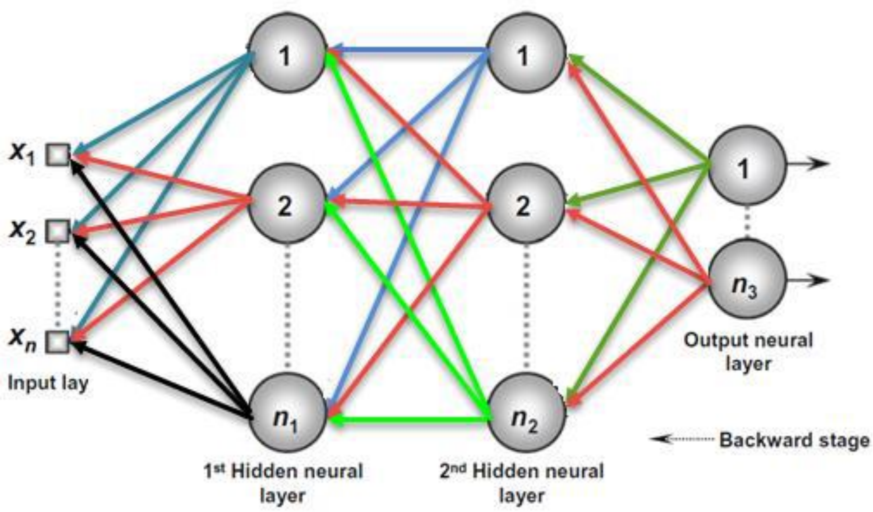
$$I_j^{(3)} = \left[I_1^{(3)}\right] = \left[W_{1,0}^{(3)} \cdot y_0^{(2)} + W_{1,1}^{(3)} \cdot y_1^{(2)} + W_{1,2}^{(3)} \cdot y_2^{(2)} + W_{1,3}^{(3)} \cdot y_3^{(2)}\right] = [0.76]$$

$$Y_j^{(3)} = \left[Y_1^{(3)}\right] = \left[g(I_1^{(3)})\right] = [tanh(0.76] = [0.64]$$

وتمثل هذه القيمة نهاية عملية التدريب بال Forward propagation.

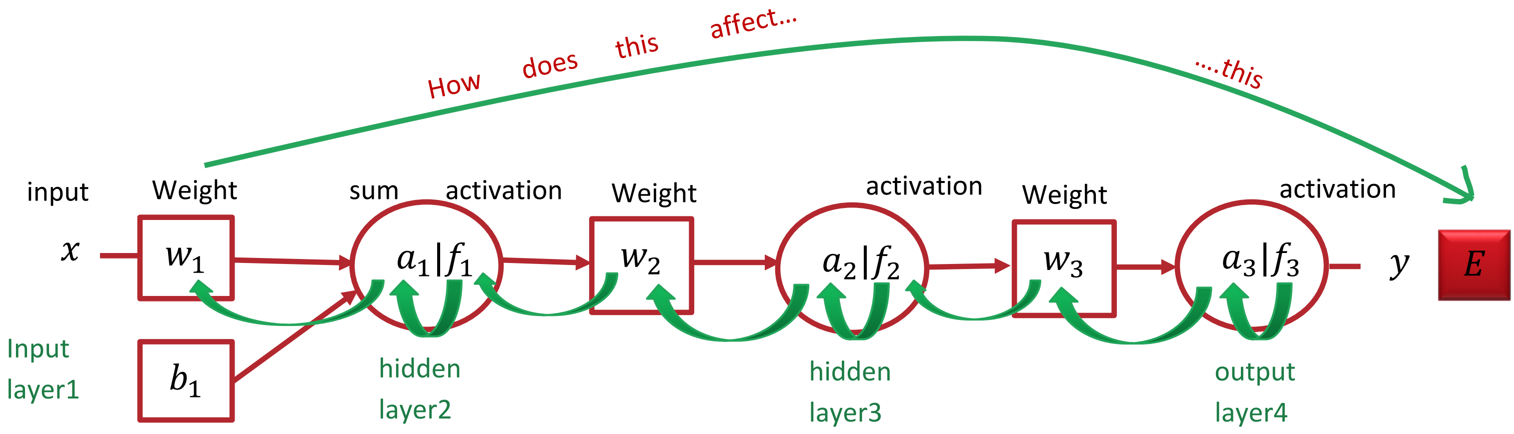
- الخطوة التالية ستكون هي عملية التراجع وتعديل الأوزان:

ويتم تكرار عملية التراجع والتعديل على الأوزان بناءً على قيمة تابع الخطأ، $E = \frac{1}{2}(d-y)^2$ وتمثل قيمته الفرق بين القيمة المرغوبة للشبكة والقيمة المحسوبة في الخطة السابقة.



The CHAIN Rule

$$\frac{\partial E}{\partial w_1}=\frac{\partial E}{\partial y}\times\frac{\partial y}{\partial f_3}\times\frac{\partial f_3}{\partial a_3}\times\frac{\partial a_3}{\partial w_3}\times\frac{\partial w_3}{\partial f_2}\times\frac{\partial f_2}{\partial a_2}\times\frac{\partial a_2}{\partial w_2}\times\frac{\partial w_2}{\partial f_1}\times\frac{\partial f_1}{\partial a_1}\times\frac{\partial a_1}{\partial w_1}$$



بناءً على العلاقة السابقة نجد أنه عملية التعديل على الأوزان بال Back propagation هي عملية تراكمية يتم فيها دراسة تأثير كل وزن من الأوزان بالخطأ، حيث أن العملية تبدأ بالتعديل على أوزان طبقة الخرج ومن ثم التراجع إلى الخلف وصولاً إلى طبقة الدخل، ودراسة تأثير أوزانها بالخطأ.

توابع الخطأ:

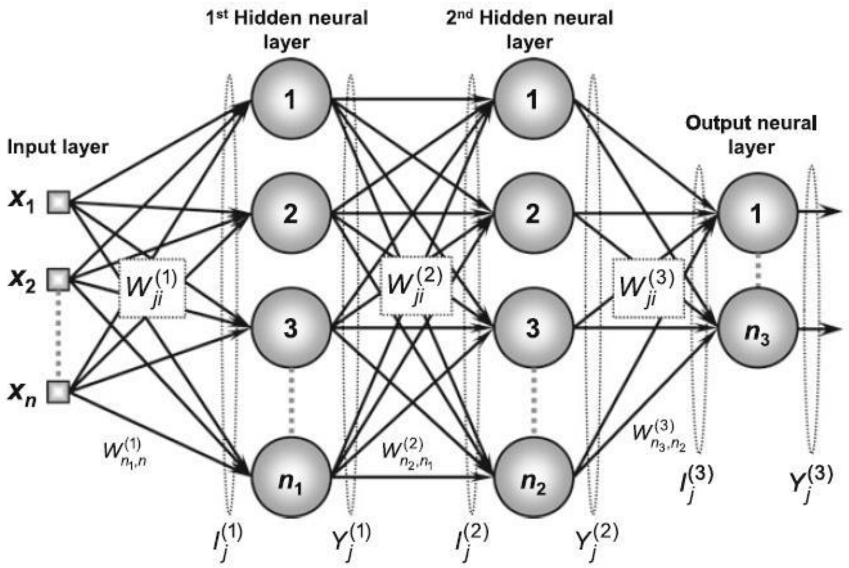
1. لدينا أولاً $E(k)=\frac{1}{2}\sum_{j=1}^{n_3}(d_j(k)-Y_j^{(3)}(k))$

حيث يمثل قيمة الخطأ عند كل عصبون من عصبونات طبقة الخرج و j^{th} تمثل رقم العصبون.

2. لدينا أيضاً الخطأ المتوسط $E_M=\frac{1}{p}\sum_{k=1}^{p}E(k)$ حيث أن p يمثل عدد العينات.

تتم عملية التعديل على الأوزان على مرحلتين:

1. التعديل على أوزان طبقة الخرج:



$$\nabla E^{(3)} = \frac{\partial E}{\partial Y_j^{(3)}} \cdot \frac{\partial Y_j^{(3)}}{\partial I_j^{(3)}} \cdot \frac{\partial I_j^{(3)}}{\partial W_{ji}^{(3)}}$$

بالاختصار نجد:

$$\nabla E^{(3)} = \frac{\partial E}{\partial W_{ji}^{(3)}}$$

$$I_j^{(3)} = \sum_{i=0}^{n_2} W_{ji}^{(3)} . Y_j^{(2)} \Rightarrow \frac{\partial I_j^{(3)}}{\partial W_{ji}^{(3)}} = Y_j^{(2)}$$

$$Y_j^{(3)} = g\left(I_j^{(3)}\right) \Rightarrow \frac{\partial Y_j^{(3)}}{\partial I_j^{(3)}} = \dot{g}(I_j^{(3)})$$

المشتق الأول لتابع التفعيل

$$E(k) = \frac{1}{2}\sum_{j=1}^{n_3}\left(d_j(k) - Y_j^{(3)}(k)\right)^2$$

بالاشتقاق

$$\Rightarrow \frac{\partial E}{\partial W_{ji}^{(3)}} = -(d_j - Y_j^{(3)})$$

$$\nabla E^{(3)} = \frac{\partial E}{\partial W_{ji}^{(3)}} = \frac{\partial E}{\partial Y_j^{(3)}} \cdot \frac{\partial Y_j^{(3)}}{\partial I_j^{(3)}} \cdot \frac{\partial I_j^{(3)}}{\partial W_{ji}^{(3)}}$$

$$\frac{\partial E}{\partial W_{ji}^{(3)}} = \underbrace{-\left(d_j - Y_j^{(3)}\right) . \dot{g}\left(I_j^{(3)}\right)}_{\text{التدرج المحلي } \delta_j^{(3)}} . Y_i^{(2)}$$

- $W_{ji}^{(3)}$: تعني الأوزان المرتبطة بالطبقة الثالثة.
- ∇: تدرج $E^{(3)}$
- وتعني تغير تدرج الخطأ بالطبقة الثالثة، وهو تغير الخطأ بالنسبة لتغير وزن من الأوزان.

تحديث الأوزان:

$$\Delta W_{ji}^{(3)} = -\eta . \frac{\partial E}{\partial W_{ji}^{(3)}} \Leftrightarrow \Delta W_{ji}^{(3)} = \eta . \delta_j^{(3)} . Y_i^{(2)}$$

الإضافة عكس الاتجاه.
ارتفع الخطأ ⟸ اطرح الوزن
قلّ الخطأ ⟸ اجمع
η: عامل التعلم

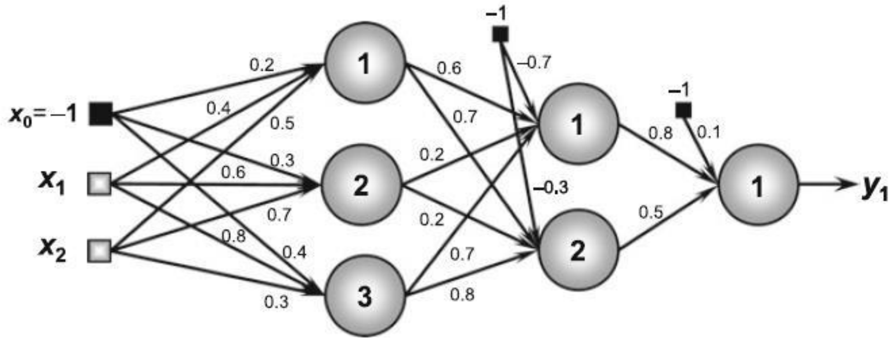
$$W_{ji}^{(3)} \leftarrow W_{ji}^{(3)} + \eta . \delta_j^{(3)} . Y_i^{(2)}$$

$$W_{ji}^{(3)} \leftarrow W_{ji}^{(3)} + \Delta W_{ji}^{(3)}$$

$$W_{ji}^{(3)} \leftarrow W_{ji}^{(3)} - \eta \frac{\partial E}{\partial W_{ji}^{(3)}}$$

2. التعديل على أوزان الطبقات المتوسطة:

- العصبونات في الطبقات المتوسطة ليس مرغوبة معروفة.
- تبدأ عملية تعديل الأوزان في الطبقات المتوسطة بعد الانتهاء من عملية تعديل أوزان طبقة الخرج.



❖ تعديل أوزان الطبقة الخفية الثانية:

كما تحدثنا سابقاً أن العملية تبدأ من النهاية إلى البداية، ولذلك تبدأ بالثانية.

$$\nabla E^{(2)} = \frac{\partial E}{\partial W_{ji}^{(2)}} = \frac{\partial E}{\partial Y_j^{(2)}} \cdot \frac{\partial Y_j^{(2)}}{\partial I_j^{(2)}} \cdot \frac{\partial I_j^{(2)}}{\partial W_{ji}^{(2)}} \Rightarrow . \frac{\partial I_j^{(2)}}{\partial W_{ji}^{(2)}} = Y_j^{(1)}$$

$$\frac{\partial Y_j^{(2)}}{\partial I_j^{(2)}} = \grave{g}(I_j^{(2)})$$

$$\frac{\partial E}{\partial Y_j^{(2)}} = \sum_{k=1}^{n_3} \frac{\partial E}{\partial I_k^{(3)}} \cdot \frac{\partial I_k^{(3)}}{\partial Y_j^{(2)}} = \sum_{k=1}^{n_3} \frac{\partial E}{\partial I_k^{(3)}} \cdot \frac{\partial \left(\sum_{k=1}^{n_3} W_{kj}^{(3)} . Y_j^{(2)}\right)}{\partial Y_j^{(2)}}$$

$$= \sum_{k=1}^{n_3} \frac{\partial E}{\partial I_k^{(3)}} \cdot W_{kj}^{(3)} = -\sum_{k=1}^{n_3} \delta_j^{(3)} \cdot W_{kj}^{(3)}$$

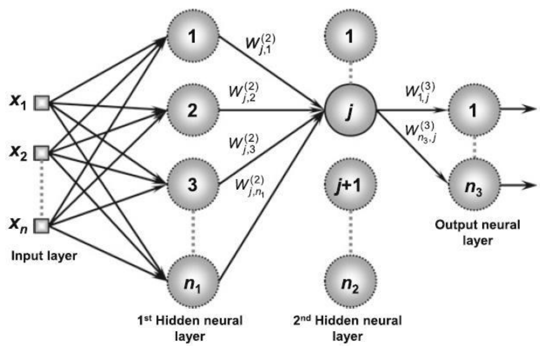
$$\frac{\partial E}{\partial W_{ji}^{(2)}} = -\left(\sum_{k=1}^{n_3} \delta_j^{(3)} \cdot W_{kj}^{(3)}\right) . \grave{g}\left(I_j^{(2)}\right) . Y_j^{(2)}$$

$$\Delta W_{ji}^{(2)} = -\eta \frac{\partial E}{\partial W_{ji}^{(2)}} \Leftrightarrow \Delta W_{ji}^{(2)} = \eta . \delta_j^{(2)} . Y_i^{(1)}$$

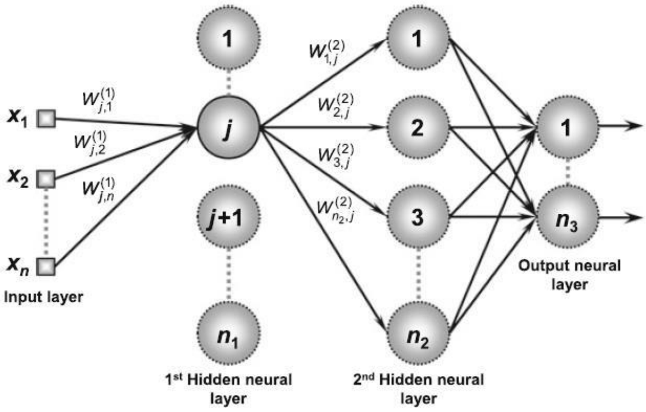
$$\delta_j^{(2)} = -\left(\sum_{k=1}^{n_3} \delta_k^{(3)} \cdot W_{kj}^{(3)}\right) . \grave{g}\left(I_j^{(2)}\right)$$

$$W_{ji}^{(2)} \leftarrow W_{ji}^{(2)} - \eta . \frac{\partial E}{\partial W_{ji}^{(2)}} \delta_j^{(3)} . Y_i^{(2)}$$

$$W_{ji}^{(2)} \leftarrow W_{ji}^{(2)} + \eta . \delta_j^{(2)} . Y_i^{(1)}$$



❖ تعديل أوزان الطبقة الخفية الأولى:



$$W_{ji}^{(1)} \leftarrow W_{ji}^{(1)} - \eta \,.\, \frac{\partial E}{\partial W_{ji}^{(1)}}$$

$$W_{ji}^{(1)} \leftarrow W_{ji}^{(1)} + \eta .\, \delta_j^{(1)} .\, x_i$$

$$\delta_j^{(1)} = -\left(\sum_{k=1}^{n_2} \delta_k^{(2)} \;.\; W_{kj}^{(2)}\right) .\, \dot{g}\left(I_j^{(1)}\right)$$

▪ ملاحظة:

$$\Omega^{(x)} = \begin{array}{c} \\ x_0 \\ x_1 \\ x_2 \\ x_3 \end{array} \begin{array}{cccc} x^{(1)} & x^{(2)} & x^{(3)} & x^{(4)} \\ \end{array} \begin{bmatrix} -1 & -1 & -1 & -1 \\ 0.2 & 0.1 & 0.9 & 0.6 \\ 0.9 & 0.3 & 0.7 & 0.4 \\ 0.4 & 0.5 & 0.8 & 0.3 \end{bmatrix}$$

وتمثل ال DATASET وتحتوي على أربعة أشعة دخل في كل شعاع ثلاثة قيم وانزياح.

$$\Omega^{d} = \begin{array}{c} \\ d_1 \\ d_2 \end{array} \begin{array}{cccc} d^{(1)} & d^{(2)} & d^{(3)} & d^{(4)} \end{array} \begin{bmatrix} 0.7 & 0.6 & 0.9 & 0.2 \\ 0.3 & 0.4 & 0.5 & 0.8 \end{bmatrix}$$

وتمثل مجموعة الخرج المرغوب، نلاحظ أن الخرج يتكون من طبقتين.

$$x^{(1)} = [-1 \quad 0.2 \quad 0.9 \quad 0.4]^T \;;\; with\;\; d^{(1)} = [0.7 \quad 0.3]^T$$
$$x^{(2)} = [-1 \quad 0.1 \quad 0.3 \quad 0.5]^T \;;\; with\;\; d^{(2)} = [0.6 \quad 0.4]^T$$
$$x^{(3)} = [-1 \quad 0.9 \quad 0.7 \quad 0.8]^T \;;\; with\;\; d^{(3)} = [0.9 \quad 0.5]^T$$
$$x^{(4)} = [-1 \quad 0.6 \quad 0.4 \quad 0.3]^T \;;\; with\;\; d^{(4)} = [0.2 \quad 0.8]^T$$

Implement the backpropagation algorithm

Begin {MLP Algorithm – Training Phase}

<1> Obtain the set of training samples $\{x^{(k)}\}$;

<2> Associate the vector with the desired output $\{d^{(k)}\}$ for each training sample;

<3> Initialize $W_{ji}^{(1)}$, $W_{ji}^{(2)}$ and $W_{ji}^{(3)}$ with small random values;

<4> Specify the learning rate $\{\eta\}$ and the required precision $\{\varepsilon\}$;

<5> Initialize the epoch counter $\{epoch \leftarrow 0\}$;

<6> Repeat:

- <6.1> $E_M^{previous} \leftarrow E_M$;
- <6.2> For all train samples $\{x^{(k)}, d^{(k)}\}$, do:
  - <6.2.1> Obtain $I_j^{(1)}$ and $Y_j^{(1)}$;
  - <6.2.2> Obtain $I_j^{(2)}$ and $Y_j^{(2)}$; *Forward steps*
  - <6.2.3> Obtain $I_j^{(3)}$ and $Y_j^{(3)}$;
  - <6.2.4> Determine $\delta_j^{(3)}$;
  - <6.2.5> Adjust $W_{ji}^{(3)}$;
  - <6.2.6> Determine $\delta_j^{(2)}$;
  - <6.2.7> Adjust $W_{ji}^{(2)}$; *Backward steps*
  - <6.2.8> Determine $\delta_j^{(1)}$;
  - <6.2.9> Adjust $W_{ji}^{(1)}$;
- <6.3> Obtain the adjusted $Y_j^{(3)}$;
- <6.4> $E_M^{current} \leftarrow E_M$;
- <6.5> $epoch \leftarrow epoch + 1$;
- Until: $\left|E_M^{current} - E_M^{previous}\right| \le \varepsilon$

End {MLP Algorithm – Training Phase}

$$E_M = \frac{1}{p}\sum_{k=1}^{p} E(k)$$

$$Y_j^{(1)} = g\left(I_j^{(1)}\right)$$

$$Y_j^{(2)} = g\left(I_j^{(2)}\right)$$

$$Y_j^{(3)} = g\left(I_j^{(3)}\right)$$

$$\delta_j^{(3)} = \left(d_j - Y_j^{(3)}\right).\acute{g}\left(I_j^{(3)}\right)$$

$$W_{ji}^{(3)} \leftarrow W_{ji}^{(3)} + \eta.\delta_j^{(3)}.Y_i^{(2)}$$

$$\delta_j^{(2)} = -\left(\sum_{k=1}^{n_3} \delta_k^{(3)} \,.\, W_{kj}^{(3)}\right).\acute{g}\left(I_j^{(2)}\right)$$

$$W_{ji}^{(2)} \leftarrow W_{ji}^{(2)} + \eta.\delta_j^{(2)}.Y_i^{(1)}$$

$$\delta_j^{(1)} = -\left(\sum_{k=1}^{n_2} \delta_k^{(2)} \,.\, W_{kj}^{(2)}\right).\acute{g}\left(I_j^{(1)}\right)$$

$$W_{ji}^{(1)} \leftarrow W_{ji}^{(1)} + \eta.\delta_j^{(1)}.x_i$$

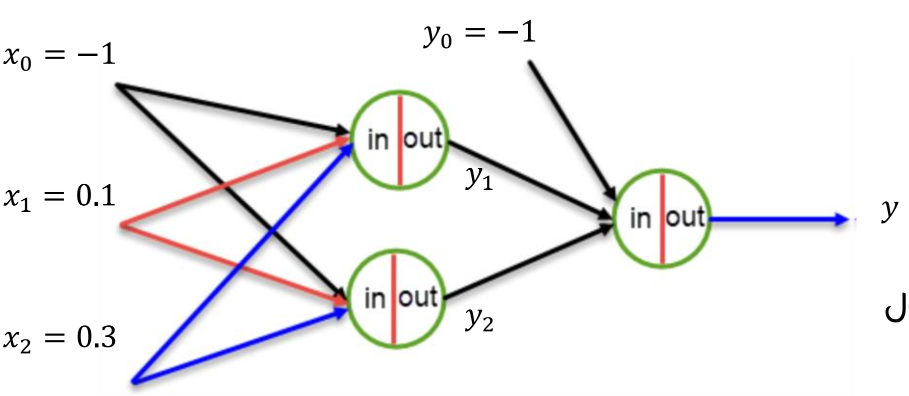
$$E_M = \frac{1}{p}\sum_{k=1}^{p} E(k)$$

مثال:

لدينا ثلاث قيم دخل

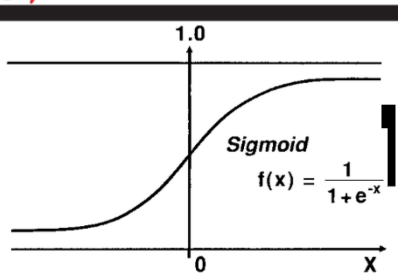
x_0	x_1	x_2
-1	0.1	0.3

ولدينا الخرج المرغوب: $d = 0.03$



- ❖ معدل التعلم $\eta = 0.1$
- ❖ نسبة الخطأ المقبول $\varepsilon = 0.001$
- ❖ الأوزان الابتدائية

w_0	w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8
-0.4	0.41	0.5	0.62	0.1	0.2	-1.83	-0.2	0.3



❖ تابع التفعيل:

❖ في البداية نقوم بحساب مجموع مضاريب الوزن بالدخل عند كل عصبون، ومن ثم ندخل الناتج إلى تابع التفعيل.

$$u_1 = w_0x_0 + w_2x_1 + w_4x_2 = (-0.4)(-1) + (0.5)(0.1) + (0.1)(0.3) = 0.48$$

$$y_1 = \frac{1}{1+e^{-u_1}} = \frac{1}{1+e^{-0.48}} = 0.618$$

$$u_2 = w_1x_0 + w_3x_1 + w_5x_2 = (0.41)(-1) + (0.62)(0.1) + (0.2)(0.3) = -0.288$$

$$y_2 = \frac{1}{1+e^{-u_2}} = \frac{1}{1+e^{0.288}} = 0.428$$

والآن نحسب مجموع خرج العصبونين السابقين ضرب الأوزان مع إضافة الانزياح:

$$u = w_6y_0 + w_7y_1 + w_8y_2 = (-1.83)(-1) + (-0.2)(0.618) + (0.3)(0.428) = 1.835$$

$$y = \frac{1}{1+e^{-u}} = \frac{1}{1+e^{-1.835}} = 0.862$$

والآن نقوم بحساب الخطأ بين الخرج المحسوب والخرج المرغوب:

$$E = \frac{1}{2}(d-y)^2 = \frac{1}{2}(0.03-0.862)^2 = 0.346 \Rightarrow E > \varepsilon$$

نلاحظ أن الخطأ الناتج أكبر من القيمة المقبولة ⇐ إذاً بعملية التعديل على الأوزان

$$\frac{\partial E}{\partial w_6} = \frac{\partial E}{\partial y} \times \frac{\partial y}{\partial u} \times \frac{\partial u}{\partial w_6}$$

$$\frac{\partial E}{\partial w_7} = \frac{\partial E}{\partial y} \times \frac{\partial y}{\partial u} \times \frac{\partial u}{\partial w_7}$$

$$\frac{\partial E}{\partial w_8} = \frac{\partial E}{\partial y} \times \frac{\partial y}{\partial u} \times \frac{\partial u}{\partial w_8}$$

$$\frac{\partial E}{\partial y} = \frac{\partial(1+e^{-u})}{\partial y} = 2\left(\frac{1}{2}\right)(d-y)(-1) = -(d-y) = 0.832$$

$$\frac{\partial y}{\partial u} = \frac{\partial\left(\frac{1}{1+e^{-u}}\right)}{\partial u} = \frac{e^{-u}}{(1+e^{-u})^2} = \frac{e^{-1.8351}}{(1+e^{-1.835})^2} = 0.118$$

$$\frac{\partial u}{\partial w_6} = \frac{\partial(w_6y_0 + w_7y_1 + w_8y_2)}{\partial w_6} = y_0 = -1$$

$$\frac{\partial u}{\partial w_7} = \frac{\partial(w_6y_0 + w_7y_1 + w_8y_2)}{\partial w_7} = y_1 = 0.618$$

$$\frac{\partial u}{\partial w_8} = \frac{\partial(w_6y_0 + w_7y_1 + w_8y_2)}{\partial w_8} = y_2 = 0.428$$

$$\frac{\partial E}{\partial w_6} = -0.0987, \qquad \frac{\partial E}{\partial w_7} = 0.061, \qquad \frac{\partial E}{\partial w_8} = 0.0423$$

وبعد الانتهاء من الطبقة الحالية ننتقل إلى الطبقة الطبقة التي تسبقها، نجد معادلات تأثير الوزن على الخرج في الطبقة الحالية هي

لدينا:

$$\frac{\partial E}{\partial w_1}=\frac{\partial E}{\partial y}\times\frac{\partial y}{\partial u}\times\frac{\partial u}{\partial y_2}\times\frac{\partial y_2}{\partial u_2}\times\frac{\partial u_2}{\partial w_1}$$
$$\frac{\partial E}{\partial w_3}=\frac{\partial E}{\partial y}\times\frac{\partial y}{\partial u}\times\frac{\partial u}{\partial y_2}\times\frac{\partial y_2}{\partial u_2}\times\frac{\partial u_2}{\partial w_3}$$
$$\frac{\partial E}{\partial w_5}=\frac{\partial E}{\partial y}\times\frac{\partial y}{\partial u}\times\frac{\partial u}{\partial y_2}\times\frac{\partial y_2}{\partial u_2}\times\frac{\partial u_2}{\partial w_5}$$

$$\frac{\partial E}{\partial w_0}=\frac{\partial E}{\partial y}\times\frac{\partial y}{\partial u}\times\frac{\partial u}{\partial y_1}\times\frac{\partial y_1}{\partial u_1}\times\frac{\partial u_1}{\partial w_0}$$
$$\frac{\partial E}{\partial w_2}=\frac{\partial E}{\partial y}\times\frac{\partial y}{\partial u}\times\frac{\partial u}{\partial y_1}\times\frac{\partial y_1}{\partial u_1}\times\frac{\partial u_1}{\partial w_2}$$
$$\frac{\partial E}{\partial w_4}=\frac{\partial E}{\partial y}\times\frac{\partial y}{\partial u}\times\frac{\partial u}{\partial y_1}\times\frac{\partial y_1}{\partial u_1}\times\frac{\partial u_1}{\partial w_4}$$

$$u=w_6y_0+w_7y_1+w_8y_2 \Rightarrow \frac{\partial u}{\partial y_0}=w_6=-1.83$$
$$\frac{\partial u}{\partial y_1}=w_7=-0.2,\qquad \frac{\partial u}{\partial y_2}=w_8=0.3$$
$$\frac{\partial y_1}{\partial u_1}=\frac{\partial(1+e^{-u_1})}{\partial u_1}=\frac{e^{-u_1}}{(1+e^{-u_1})^2}=0.236$$
$$\frac{\partial y_2}{\partial u_2}=\frac{\partial(1+e^{-u_2})}{\partial u_2}=\frac{e^{-u_2}}{(1+e^{-u_2})^2}=0.244$$

$$\frac{\partial u_2}{\partial w_1}=\frac{\partial(w_1x_0+w_3x_1+w_5x_2)}{\partial w_1}=x_0=-1$$
$$\frac{\partial u_2}{\partial w_3}=\frac{\partial(w_1x_0+w_3x_1+w_5x_2)}{\partial w_3}=x_1=0.1$$
$$\frac{\partial u_2}{\partial w_5}=\frac{\partial(w_1x_0+w_3x_1+w_5x_2)}{\partial w_5}=x_2=0.3$$

$$\frac{\partial u_1}{\partial w_0}=\frac{\partial(w_0x_0+w_2x_1+w_4x_2)}{\partial w_0}=x_0=-1$$
$$\frac{\partial u_1}{\partial w_1}=\frac{\partial(w_0x_0+w_2x_1+w_4x_2)}{\partial w_1}=x_1=0.1$$
$$\frac{\partial u_1}{\partial w_4}=\frac{\partial(w_0x_0+w_2x_1+w_4x_2)}{\partial w_4}=x_2=0.3$$

نعوض في معادلات حساب تأثير الوزن على الخرج فينتج لدينا تاثير كل وزن:

$\frac{\partial E}{\partial w_1}=-0.007$ $\frac{\partial E}{\partial w_3}=0.0007$ $\frac{\partial E}{\partial w_5}=0.002$ $\frac{\partial E}{\partial w_7}=0.061$

w_0	w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8
-0.4	0.41	0.5	0.62	0.1	0.2	-1.83	-0.2	$\mathbf{0.3}$

$\frac{\partial E}{\partial w_0}=0.004$ $\frac{\partial E}{\partial w_2}=0.0004$ $\frac{\partial E}{\partial w_4}=0.001$ $\frac{\partial E}{\partial w_6}=-0.098$ $\frac{\partial E}{\partial w_8}=0.042$

وأخيراً نقوم بتحديث الأوزان عن طريق المعادلة:

$$\boxed{w_j = w_j - \eta \frac{\partial E}{\partial w_j}}$$

وبمعدل تعلم $\eta = 0.1$:

epoch 1

w_0	w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8
-0.4	0.41	0.5	0.619	0.1	0.199	-1.82	-0.206	$\mathbf{0.295}$

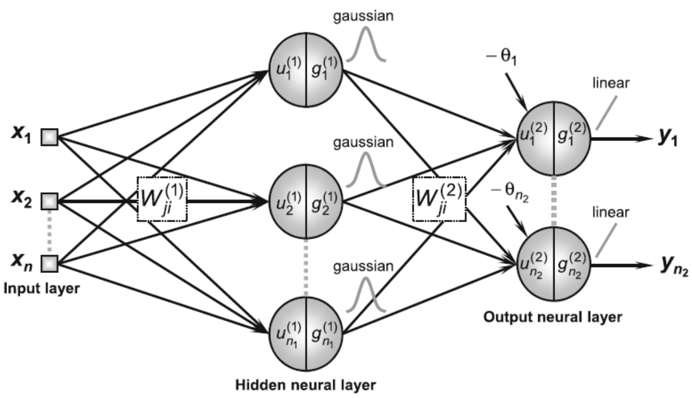
epoch 126

w_0	w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8
-0.629	0.405	0.522	0.620	0.168	0.201	-0.338	-1.154	$\mathbf{-0.332}$

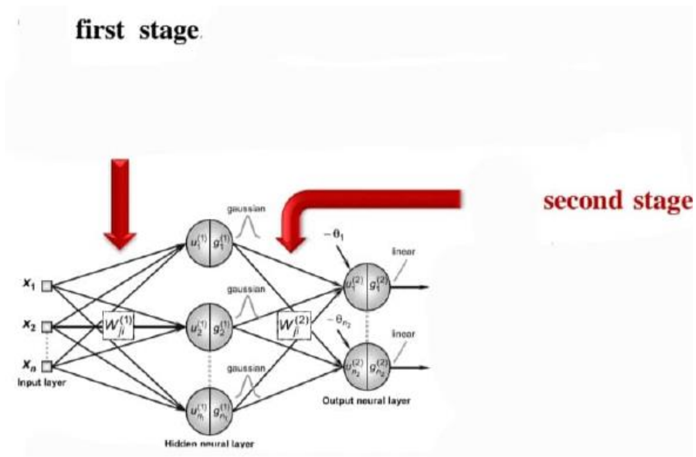
نلاحظ أننا احتجنا إلى 126 *epoch* وذلك من أجل أن تصبح قيمة الخطأ أقل من قيمة ε (قيمة الخطألمقبول)، وهو شرط التوقف.

Radial Basis Functions (RBF) Neural Networks

- أخذنا فيما سبق الشبكات التقليدية وسنرى في هذه المحاضرة شبكات أخرى تعتمد على توابع الـRBF
- شبكات RBF هي شبكات مؤلفة من طبقة مخفية واحدة، في هذه الشبكات يكون تابع التفعيل ليس التقليدي كما كان سابقاً، وإنما له طبيعة خاصة (مثل غاوص) حيث أن قيم الدخل لن تدخل للشبكة مباشرةً وإنما ستتعدل وفق تابع الغاوص، وخرج هذا التابع هو الدخل الحقيقي الذي سيدخل للشبكة.
- في النموذج التقليدي كان تابع التفعيل في الطبقة الأولى نفسه الموجود في الثانية، بينما شبكات RBF التابع الموجود في الطبقة الأولى مختلف عن طبقة الخرج.



Typical configuration of an RBF network



- الطبقة الأولى وحساباتها وتابع التفعيل بها هو stage 1.
- طبقة الخرج وحساباتها وتابع التفعيل بها هو stage 2.

تابع الغاوص:

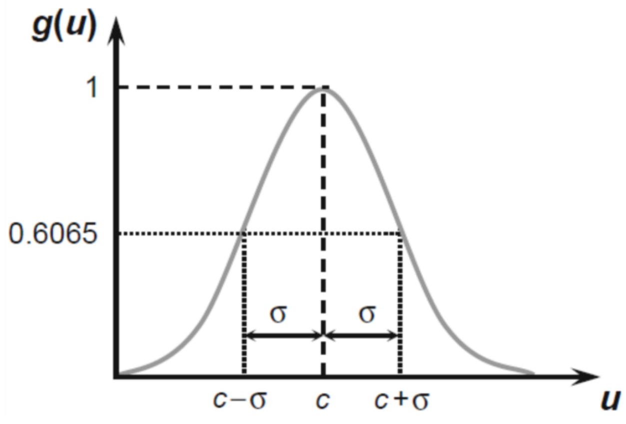
تختلف عن بعضها بمقدار المتوسط والانحراف المعياري (البعد عن المتوسط)، وهذين هما البارامترين الأساسيين في غاوص حيث تُعطى علاقة غاوص كالتالي:

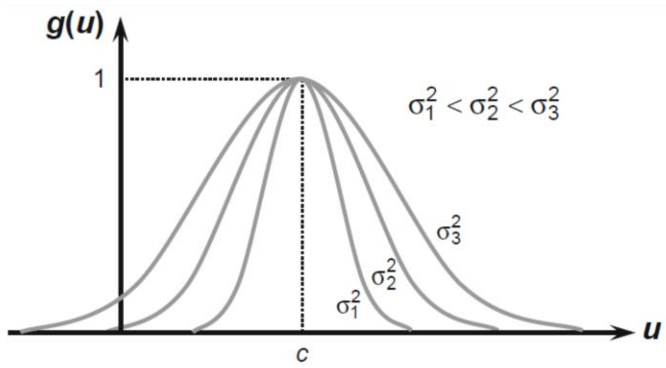
$$\boxed{g(u) = e^{-\frac{(u-c)^2}{2\sigma^2}}}$$

حيث: u القيمة المدروسة

c المتوسط

σ الانحراف المعياري





الفرق بين هذه المنحنيات هو بقيمة الانحراف المعياري.

- خرج الطبقة الأولى (طبقة دخل مرتبطة مع العصبونات):

$$g_j^{(1)}\left(u_j^{(1)}\right) = g_j^{(1)}(x) = e^{-\frac{\sum_{i=1}^{n}\left(x_i - w_{ji}^{(1)}\right)^2}{2\sigma_j^2}}$$

حيث $j = 1, \dots, n1$

نلاحظ أن العلاقة السابقة تكافئ علاقة غاوص أي كأننا نقول أن w هي المتوسطات أي أن الأوزان التي سنضعها في أول مرحلة هي المتوسطات التي ستحدد أو تحسب حساباً حيث $n1$ هو عدد العصبونات الموجودة في الطبقة الأولى، ومن أجل كل عصبون سنحسب له المتوسط والانحراف المعياري.

خوارزمية تديب شبكات RBF:

حيث أول مرحلة هي حساب المتوسطات عن طريق خوارزمية K-means (والتي سيرد شرحها بعد قليل)

Adjustment of the Neurons from the Intermediate Layer (Stage I)

BEGIN { RBF ALGORITHM – FIRST TRAINING PHASE}

<1> Obtain the set of training samples $\{x^{(k)}\}$;
<2> Initialize the weight vector of each neuron of the intermediate layer with the values of the n_1 first training samples;
<3> Repeat these instructions:
<3.1> For all samples $\{x^{(k)}\}$, do:
<3.1.1> Calculate the Euclidian distances between $x^{(k)}$ and $W_{ji}^{(1)}$, considering a single j-th neuron at each time;
<3.1.2> Select the neuron j with the shortest distance in order to group the given sample with the closest center;
<3.1.3> Attribute $x^{(k)}$ to group $\Omega^{(j)}$;
<3.2> For all $W_{ji}^{(1)}$, with $j = 1..n_1$, do:
<3.2.1> Adjust $W_{ji}^{(1)}$ according to the samples in $\Omega^{(j)}$:

$$W_{ji}^{(1)} = \frac{1}{m^{(j)}} \sum_{x^{(k)} \in \Omega^{(j)}} x^{(k)} \quad \{m^{(j)} \text{ is the number of samples in } \Omega^{(j)}\}$$

Until: there are no more changes in the groups $\Omega^{(j)}$ between two iterations;
<4> For all $W_{ji}^{(1)}$, with $j = 1..n_1$, do:
<4.1> Calculate the variance of each Gaussian activation function by using the mean squared distance criterion:

$$\sigma_j^2 = \frac{1}{m^{(j)}} \sum_{x^{(k)} \in \Omega^{(j)}} \sum_{i=1}^{n} \left(x_i^{(k)} - W_{ji}^{(1)}\right)^2$$

END { RBF ALGORITHM – FIRST TRAINING PHASE}

Center

$$g(u) = e^{-\frac{(u-c)^2}{2\sigma^2}}$$

Parameters to be adjusted

standard deviation

Centers: k-means clustering method

Standard deviation

وثاني مرحلة هي الشبكة التقليدية:

• Adjustment of Neurons of the Output Layer (Stage II)

BEGIN {RBF ALGORITHM – SECOND TRAINING PHASE}

<1> Obtain the original training sample set $\{x^{(k)}\}$;
<2> Obtain the desired output vector $\{d^{(k)}\}$ for each sample;
<3> Initialize $W_{ji}^{(2)}$ with small random values;
<4> Specify the learning rate $\{\eta\}$ and required precision $\{\varepsilon\}$;
<5> For all samples $\{x^{(k)}\}$, do:
<5.1> Obtain the values of $g_j^{(1)}$ with respect to the sample $x^{(k)}$;
<5.2> Assume $z^{(k)} = [g_1^{(1)} \; g_2^{(1)} \ldots g_{n_1}^{(1)}]^T$; {pseudo-samples}
<6> Initialize the epoch counter $\{epoch \leftarrow 0\}$;
<7> Repeat the instructions:
<7.1> $E_M^{previous} \leftarrow E_M$;
<7.2> For all training pairs $\{z^{(k)}, d^{(k)}\}$, do:
{Adjust $W_{ji}^{(2)}$ and θ_j by applying the same steps used on the adjustment of the synaptic weights of the MLP output layer}
<7.3> $E_M^{current} \leftarrow E_M$;
<7.4> $epoch \leftarrow epoch + 1$;
Until: $\left|E_M^{current} - E_M^{previous}\right| \le \varepsilon$

END {RBF ALGORITHM – SECOND TRAINING PHASE}

$$g_j^{(1)}(u_j^{(1)}) = g_j^{(1)}(x) = e^{-\frac{\sum_{i=1}^{n}\left(x_i - w_{ji}^{(1)}\right)^2}{2\sigma_j^2}}, \text{ where } j = 1, \ldots, n_1$$

$$E_M = \frac{1}{p}\sum_{k=1}^{p} E(k),$$

مثال لتوضيح فكرة حساب المتوسطات عن طريق خوارزمية K-means:

لنفرض لدينا صف من الطلاب ونريد تقسيمهم إلى مجموعتين ($k = 2$)، بحيث نوجد مركز للمجموعة الأولى يكون بوسطها ومركز للمجموعة الثانية بوسطها، بداية لا نعرف أين المراكز ولكن ما نعرفه هو إحداثيات كل طالب، لذلك نبحث عن أفضل طريقة لوضع مركزي المجموعتين، فيكونا هذان المركزان هما المتوسطات المطلوبة (ولكننا لم نحدد عدد العناصر بكل المجموعة).

- أصبح المعلوم هو عدد المجموعات $k = 2$
- والمطلوب هو تقسيم الطلاب إلى مجموعتين وتحديد مركز لكل مجموعة، وهذا ما يُدعى ب Clustering والذي تكلمنا عنه في المحاضرة الأولى.

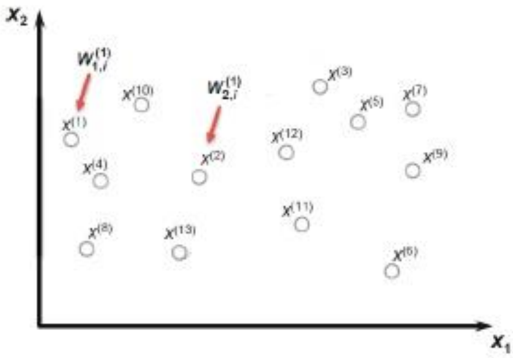
سنفترض بدايةً أن شخص ما وبشكل عشوائي هو مركز المجموعة الأولى، وشخص آخر هو مركز المجموعة الثانية ولتجميع العناصر إلى مجموعات يجب معرفة المسافة بين كل طالب مع المركزين المفترضين سابقاً، أصبح لكل طالب مسافة عن المركز الأول ومسافة عن المركز الثاني ومن أجل كل طالب نقارن المسافةعن المركز الأول فيما إذا كانت أصغر من المسافة عن المركز الثاني، فهذا الطالب مع المجموعة الأولى وإلاّ نصنفه مع المجموعة الثانية، الآن أصبح الصف عبارة عن مجموعتين بحيث قسم من الطلاب أقرب للمجموعة الأولى والقسم الثاني أقرب للمجموعة الثانية، الآن من أجل المجموعة الأولى نجمع اكسات ووايات كل عناصرها ونقسمهن على عدد المجموعة الأولى، وكذلك الأمر بالنسبة للمجموعة الثانية وبالتالي ينتج لدينا متوسطين جديدين، ندخل دورة جديدة ونعيد حساب بُعد كل الطلاب عن المتوسطين الجديدين فيمكن أن ينتج لدينا أن هناك أحد الطلاب أصبح أقرب لمجموعة غير مجموعته فينتقل إليها... وهكذا نعيد هذه الخطوات حتى تستقر المجموعات دون أي انتقال.

- تذكرة:
- قلنا سابقاً أنه في Machine Learning لدينا:
- Supervised Learning ⟵ Classification تصنيف
- Unsupervised Learning ⟵ Clustering تجميع
- و k-means هي أفضل خوارزميات التجميع.

إذاً في أول مرحلة الهدف هو حساب المتوسطين وهي مراكز المجموعات، ومن ثم أصبح لدينا حساب الانحراف المعياري (البعد عن المتوسط) وهو المسافة بين كل عنصر والمتوسط، أي أننا في أول مرحلة نأخذ جميع العمليات ونقوم بتطبيق خوارزمية k-means عليها، فينتج لدينا المتوسط والانحراف المعياري فندخل للمرحلة الثانية وهي الشبكة العصبونية التقليدية.

مثال عن k-means:

لدينا مجتمع العينات التالي:

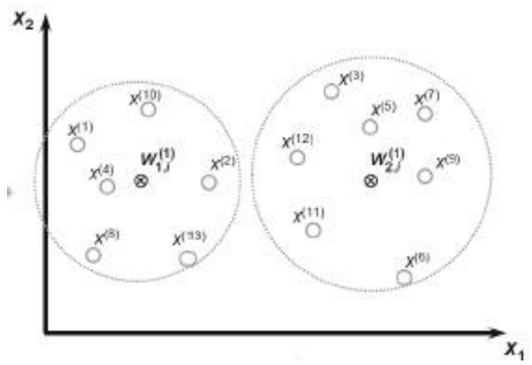


ونريد تقسيمه إلى مجموعات (تجميع)، نبدأ بمركزين افتراضيين $x^{(1)}, x^{(2)}$ ثم نطبق خوارزمية k-means لتجميعهم حسب بُعدهم عن المركز (حيث كل العناصر الموجودة في المجموعة الأولى هي أقرب للمركز الأول عن المركز الثاني وكل العناصر الموجودة في المجموعة الثانية هي أقرب للمركز الثاني عن المركز الأول) وفق تابع قياس المسافة.

$$\Omega_{initial}^{(1)} = \{x^{(1)}, x^{(4)}, x^{(8)}, x^{(10)}\}$$
$$\Omega_{initial}^{(2)} = \{x^{(2)}, x^{(3)}, x^{(5)}, x^{(6)}, x^{(7)}, x^{(9)}, x^{(11)}, x^{(12)}, x^{(13)}\}$$

ملاحظة هامة جداً:
المسافة ليست شرط أن تكون إحداثيات (x, y) حيث أنه من الممكن أن تكون لون (مثلاً عنصر 1 لونه 50 وعنصر 2 لونه 70 ولدينا لون 60، إذاً للعنصرين السابقين نفس المسافة عن 60 وهذه هي مسافة لون) أي أنه ليس دائماً المسافة إحداثيات ولكن افترضنا ذلك في المثال السابق لتوضيح الفكرة فقط لأن الإحداثيات هي أبسط مثال عن المسافة.
نتيجة: قياس المسافات ليس مرتبط فقط بالإحداثيات وإنما أصبح الفرق بين سمتين: لون، وزن، طول، ...

وهكذا أصبح شكل المجموعتين بعد تطبيق خوارزمية k-means



وهذه هي الداتا سيت الموافقة:

$$\Omega_{final}^{(1)} = \{x^{(1)}, x^{(2)}, x^{(4)}, x^{(8)}, x^{(10)}, x^{(13)}\}$$
$$\Omega_{final}^{(2)} = \{x^{(3)}, x^{(5)}, x^{(6)}, x^{(7)}, x^{(9)}, x^{(11)}, x^{(12)}\}$$

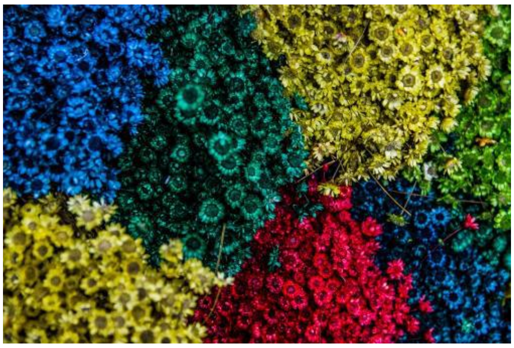
الآن إذا أتت عينة جديدة ونريد عمل clustering لها إلى إحدى هاتين المجموعتين نقيس مسافتها عن المركزين ثم نضيفها للمركز الأقرب، وبعد إضافتها نعدل الأوزان (حساب متوسط من جديد) ثم نعيد حساب مركز المجموعة التي أُضيف إليها.

- بعد حساب الأوزان (المتوسطات والانحراف المعياري) تطبقت الشبكة العصبونية بشكل كامل، أصبح أول مدخل هو عينات التدريب، الموجود بالعقد (العصبونات) الأولى هي توابع التفعيل الغاوص لل RBF والتي حسبنا σ, μ من أجلها (μ هو رمز المتوسط) انطلاقاً من عينات الدخل عن طريق عمليات clustering باستخدام خوارزمية k-means، والقسم الثاني من الشبكة هو حسابات تقليدية حيث أصبح لدينا عناصر دخل، أوزان، تابع تفعيل، خطأ، ثم نُحسّن الأوزان لنصل للنتيجة المطلوبة.

k-means clustering:

- k-means: هي عملية تجميع العينات وهي أحد أفضل خوارزميات clustering.
- Clustering: هو عملية تجميع الداتا إلى عدة أصناف، أمثلة عنه:

مجموعة أشخاص وتم تجميعهم حسب اللباس، الجنس (رسمي، تسوّق، ...)



صورة ورود وتم تجميعهم من خلال اللون، فاللون هنا هو حد الفصل.

قياس المسافة:

هذه الفكرة التي تم شرحها قبل قليل من خلال الملاحظة لها علاقة عامة تدعى "Minkowski distance" وتنص على أن المسافة بين شعاعين x, y

$$x = [x_1, x_2, x_3, \dots, x_d]$$
$$y = [y_1, y_2, y_3, \dots, y_d]$$

وفق المتغير r هي:

$$d_r(x, y) = \left(\sum_{i=1}^{d} |x_i - y_i|^r\right)^{\frac{1}{r}}$$

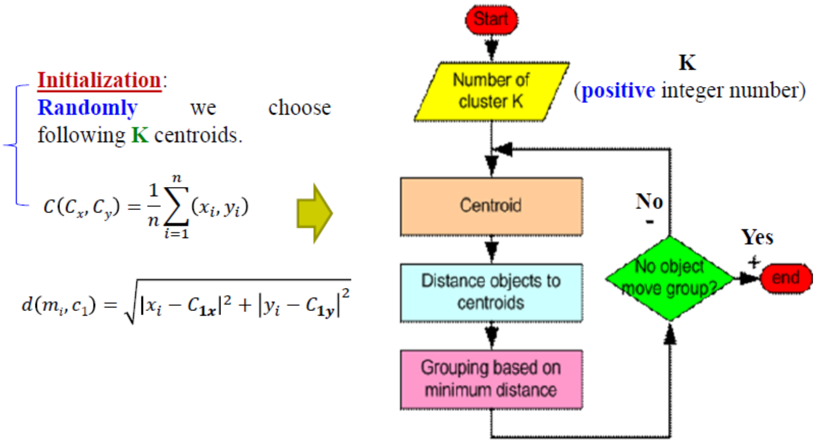
حيث d هي عدد العناصر بكل شعاع.

ينتج عن هذه العلاقة حالات خاصة هي:

$$r = 1 \Rightarrow Manhattan\ Distance\ (City\ Block\ Distance)$$
$$d_r(x, y) = \sum_{i=1}^{d} |x_i - y_i|$$

$$d_r(x, y) = \sqrt{\sum_{i=1}^{d} |x_i - y_i|^2}$$
$$r = 2 \Rightarrow Euclidean\ Distance$$

خوارزمية k-means:



حيث نبدأ من start ثم نحدد عدد الكلاسات ثم نحسب المراكز (نفرضهم فرضاً في المرة الأولى) ثم نحسب المسافة بين كل عنصر وهذه المراكز، ثم نجمّع العناصر حسب المسافة الأقصر (مثلاً لدينا مجموعتين فالعناصر الأقرب للمركز الأول أصبحوا المجموعة الأولى والعناصر الأقرب للمركز الثاني أصبحوا المجموعة الثانية).

- لو كان $k = 3$ لأصبح لكل عنصر ثلاث قيم (بُعده عن المركز الأول، بُعده عن الثاني، بُعده عن الثالث) وبالتالي نجمع حسب الأقرب ثم أخيراً نختبر فيما إذا انتقل أحد العناصر لمجموعة أخرى؟ إذا كان الجواب نعم فندور دورة ثانية وإلاّ نقف ويكون قد انتهى التجميع.

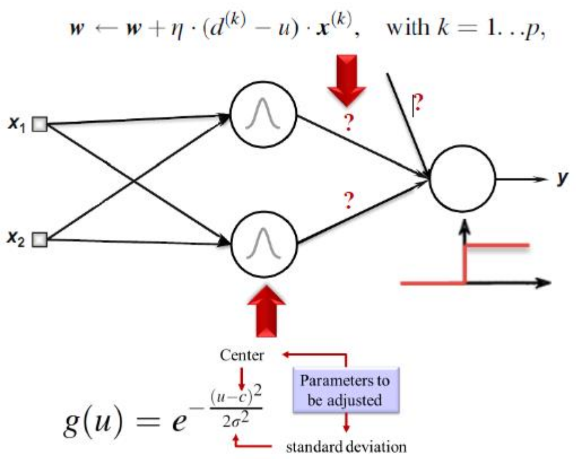
مثال شامل عن RBF:

نريد تدريب الشبكة العصبونية التالية

حيث المداخل هي x_1, x_2

المطلوب هو إشارات الاستفهام، أي كيف ستتغير القيم انطلاقاً من تابع التفعيل المفروض في الطبقة المخفية (غاوص).

لنحسب تغيّر القيم يجب حساب متغيرات تابع الغاوص σ, μ



Centers: k-means clustering method

الحل:

بما أن الشبكة التي هي RBF فالمطلوب كمرحلة أولى هو حساب خصائص تابع الغاوص وسنعتمد في ذلك على خوارزمية k-means.

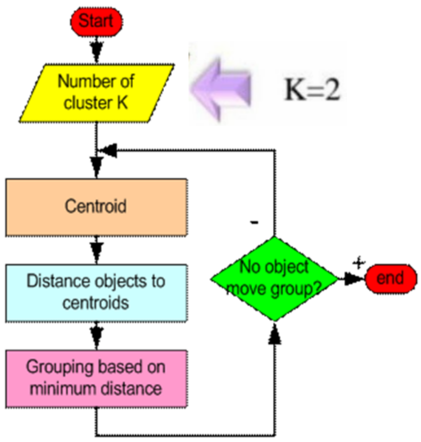
سنأخذ الأمثلة التالية من $x^{(1)}, \ldots, x^{(7)}$ عينات للدخل

	x_1	x_2	d
$x^{(1)}$	1.0	1.0	A
$x^{(2)}$	1.5	2.0	A
$x^{(3)}$	3.0	4.0	B
$x^{(4)}$	5.0	7.0	B
$x^{(5)}$	3.5	5.0	B
$x^{(6)}$	4.5	5.0	B
$x^{(7)}$	3.5	4.5	B

- ملاحظة: الانزياح لا يدخل معنا في هذه المرحلة بالحسابات لأن الهدف بالطقة الأولى ليس الانزياحات وإنما تعديل كل قيمة بالدخل وفق تابع الغاوص.

- لدينا 7 عينات ونريد تجميعهم إلى صنف A وصنف B وكأننا نقول أننا نريد تدريب هذه الشبكة لتنجح لاحقاً لتصنيف كل عينة إما بالصنف A أو بالصنف B، ولكن يجب الانتباه إلى مرحلة التدريب بالقسم الأول هي Unsupervised Learning (أي أننا لا نعرف الخرج وإنما نبحث عن تجميع لهم) ومرحلة التدريب بالقسم الثاني هي Supervised Learning (أي نعرف الخرج المرغوب وهو d)، أي اجتمع لدينا Clustering و Classification معاً.

نبدأ بتطبيق خوارزمية k-means:



كونه لدينا عصبونين في الطبقة الأولى ⟸ إذاً لدينا مركزين ⟸ وبالتالي $k = 2$

سنفترض بدايةً أن المركزين هما:

$$c_1 = x^{(1)} = (1.0, 1.0)$$
$$c_2 = x^{(4)} = (5.0, 7.0)$$

نحسب مسافة كل عنصر عن المراكز عن طريق العلاقة:

$$d\left(X^{(i)}, C^{(i)}\right) = \sqrt{\left|X_{x_1}^{(i)} - C_{x_1}^{(i)}\right|^2 + \left|X_{x_2}^{(i)} - C_{x_2}^{(i)}\right|^2}$$

مثال عن حساب مسافتين والباقي بنفس الطريقة:

$$d\left(X^{(2)}, C^{(1)}\right) = \sqrt{|1.5 - 1.0|^2 + |2.0 - 1.0|^2} = 1.12$$
$$d\left(X^{(2)}, C^{(2)}\right) = \sqrt{|1.5 - 5.0|^2 + |2.0 - 7.0|^2} = 6.10$$

بعد إكمال الباقي ينتج لدينا الجدول التالي

	x_1	x_2	$d\left(X^{(i)}, C^{(1)}\right)$	$d\left(X^{(i)}, C^{(2)}\right)$
$x^{(1)}$	1.0	1.0	0	7.21
$x^{(2)}$	1.5	2.0	1.12	6.10
$x^{(3)}$	3.0	4.0	3.61	3.61
$x^{(4)}$	5.0	7.0	7.21	0
$x^{(5)}$	3.5	5.0	4.72	2.5
$x^{(6)}$	4.5	5.0	5.31	2.06
$x^{(7)}$	3.5	4.5	4.30	2.92

حيث: $d(x^{(i)}, c^{(1)})$ هو المسافة بين العينة $x^{(i)}$ والمركز $c^{(1)}$

و $d(x^{(i)}, c^{(2)})$ هو المسافة بين العينة $x^{(i)}$ والمركز $c^{(2)}$

والآن من أجل كل عنصر وحسب المسافة الأصغر ينضم إلى المجموعة الموافقة فتنتج لدينا المجموعتين كالتالي:

	x_1	x_2	$d(X^{(i)}, C^{(1)})$	$d(X^{(i)}, C^{(2)})$
$x^{(1)}$	1.0	1.0	0	7.21
$x^{(2)}$	1.5	2.0	1.12	6.10
$x^{(3)}$	3.0	4.0	3.61	3.61
$x^{(4)}$	5.0	7.0	7.21	0
$x^{(5)}$	3.5	5.0	4.72	2.5
$x^{(6)}$	4.5	5.0	5.31	2.06
$x^{(7)}$	3.5	4.5	4.30	2.92

- نحسب المراكز الجديدة وذلك بجمع الاكسات وتقسيمهم على 3، وكذلك الأمر بالنسبة للوايات وذلك من أجل المجموعة الأولى (قسمنا على 3 لأن عدد عناصر المجموعة الأولى هو 3) وأيضاً بجمع الاكسات وتقسيمهم على 4 وكذلك الأمر بالنسبة للوايات وذلك من أجل المجموعة الثانية

	x_1	x_2	$d(X^{(i)}, C^{(1)})$	$d(X^{(i)}, C^{(2)})$
$x^{(1)}$	1.0	1.0	0	7.21
$x^{(2)}$	1.5	2.0	1.12	6.10
$x^{(3)}$	3.0	4.0	3.61	3.61
$x^{(4)}$	5.0	7.0	7.21	0
$x^{(5)}$	3.5	5.0	4.72	2.5
$x^{(6)}$	4.5	5.0	5.31	2.06
$x^{(7)}$	3.5	4.5	4.30	2.92

$$c^{(1)}\left(c_{x_1}^{(1)}, c_{x_2}^{(1)}\right) = \frac{1}{3}\sum_{i=1}^{3}\left(X_{x_1}^{(i)}, X_{x_2}^{(i)}\right)$$

$$c^{(2)}\left(c_{x_1}^{(2)}, c_{x_2}^{(2)}\right) = \frac{1}{4}\sum_{i=1}^{4}\left(X_{x_1}^{(i)}, X_{x_2}^{(i)}\right)$$

$$c^{(1)} = \frac{1}{3}\big((1.0 + 1.5 + 3.0), (1.0 + 2.0 + 4.0)\big) = (1.83, 2.33)$$

$$c^{(2)} = \frac{1}{4}\big((5.0 + 3.5 + 4.5 + 3.5), (7.0 + 5.0 + 5.0 + 4.5)\big) = (4.12, 5.38)$$

- ثم نعيد الحسابات بناءً على المراكز الجديدة فينتج الجدول التالي: $C^{(1)}(1.83, 2.33)$ $C^{(2)}(4.12, 5.38)$

	x_1	x_2	$d(X^{(i)}, C^{(1)})$	$d(X^{(i)}, C^{(2)})$
$x^{(1)}$	1.0	1.0	1.57	5.38
$x^{(2)}$	1.5	2.0	0.47	4.28
$x^{(3)}$	3.0	4.0	2.04	1.78
$x^{(4)}$	5.0	7.0	5.64	1.84
$x^{(5)}$	3.5	5.0	3.15	0.73
$x^{(6)}$	4.5	5.0	3.78	0.54
$x^{(7)}$	3.5	4.5	2.74	1.08

$$d\left(X^{(2)}, C^{(1)}\right) = \sqrt{|1.5 - 1.83|^2 + |2.0 - 2.33|^2} = 0.47$$

$$d\left(X^{(2)}, C^{(2)}\right) = \sqrt{|1.5 - 4.12|^2 + |2.0 - 5.38|^2} = 4.28$$

نعيد توزيع العناصر حسب المسافة الأقرب

	x_1	x_2	$d(X^{(i)}, C^{(1)})$	$d(X^{(i)}, C^{(2)})$
$x^{(1)}$	1.0	1.0	1.57	5.38
$x^{(2)}$	1.5	2.0	0.47	4.28
$x^{(3)}$	3.0	4.0	2.04	1.78
$x^{(4)}$	5.0	7.0	5.64	1.84
$x^{(5)}$	3.5	5.0	3.15	0.73
$x^{(6)}$	4.5	5.0	3.78	0.54
$x^{(7)}$	3.5	4.5	2.74	1.08

نلاحظ أنه تغير موضع بعض العناصر من مجموعة إلى أخرى فندخل دورة جديدة.

- نحسب المراكز الجديدة

$$c^{(1)} = \frac{1}{2}\big((1.0 + 1.5), (1.0 + 2.0)\big) = (1.25{,}1.5)$$

$$c^{(2)} = \frac{1}{5}\big((3 + 5.0 + 3.5 + 4.5 + 3.5), (4 + 7.0 + 5.0 + 5.0 + 4.5)\big) = (3.9, 5.1)$$

- ثم نعيد الحسابات بناءً على المراكز الجديدة فينتج الجدول التالي: $C^{(1)}(1.25{,}1.5)$ $C^{(2)}(3.9{,}5.1)$

	x_1	x_2	$d(X^{(i)}, C^{(1)})$	$d(X^{(i)}, C^{(2)})$
$x^{(1)}$	1.0	1.0	0.56	5.02
$x^{(2)}$	1.5	2.0	0.56	3.92
$x^{(3)}$	3.0	4.0	3.05	1.42
$x^{(4)}$	5.0	7.0	6.66	2.20
$x^{(5)}$	3.5	5.0	4.16	0.41
$x^{(6)}$	4.5	5.0	4.78	0.61
$x^{(7)}$	3.5	4.5	3.75	0.72

$$d(X^{(2)}, C^{(1)}) = \sqrt{|1.5 - 1.25|^2 + |2.0 - 1.5|^2} = 0.56$$

$$d(X^{(2)}, C^{(2)}) = \sqrt{|1.5 - 3.9|^2 + |2.0 - 5.1|^2} = 3.92$$

- نعيد توزيع العناصر حسب المراكز الألأقرب

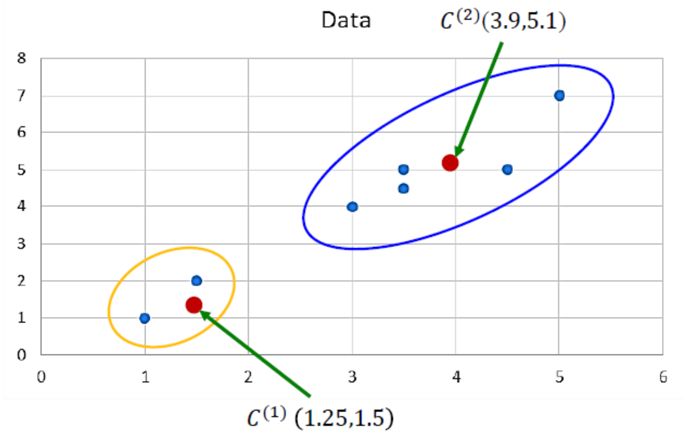
	x_1	x_2	$d(X^{(i)}, C^{(1)})$	$d(X^{(i)}, C^{(2)})$
$x^{(1)}$	1.0	1.0	0.56	5.02
$x^{(2)}$	1.5	2.0	0.56	3.92
$x^{(3)}$	3.0	4.0	3.05	1.42
$x^{(4)}$	5.0	7.0	6.66	2.20
$x^{(5)}$	3.5	5.0	4.16	0.41
$x^{(6)}$	4.5	5.0	4.78	0.61
$x^{(7)}$	3.5	4.5	3.75	0.72

نلاحظ انه نتجت قيم جديدة

ولكن لم يتغير موضع أي عنصر

$\Leftarrow$ إذاً هذه هي μ_1, μ_2. التي نبحث عنها

$$c^{(1)} = (1.25{,}1.5), \qquad c^{(2)} = (3.9, 5.1)$$



- النقاط الموجودة باللون الأزرق هي العناصر.
النقاط الموجودة باللون الأحمر هي المراكز.

حساب الانحراف المعياري من العلاقة:

$$\sigma_j^2 = \frac{1}{m^{(i)}} \sum_{x^{(k)} \epsilon \Omega^j} \sum_{j=1}^{n} \left(x^{(k)} - \mathrm{C}^{(j)}\right)^2$$

	x_1	x_2	$d\left(X^{(i)}, C^{(1)}\right)$	$d\left(X^{(i)}, C^{(2)}\right)$
$x^{(1)}$	1.0	1.0	0.56	5.02
$x^{(2)}$	1.5	2.0	0.56	3.92
$x^{(3)}$	3.0	4.0	3.05	1.42
$x^{(4)}$	5.0	7.0	6.66	2.20
$x^{(5)}$	3.5	5.0	4.16	0.41
$x^{(6)}$	4.5	5.0	4.78	0.61
$x^{(7)}$	3.5	4.5	3.75	0.72

$$\left(X^{(1)} - C^{(1)}\right)^2 = \left(d\left(x^{(1)}, C^{(1)}\right)^2\right) = \left|X_{x_1}^{(1)} - C_{x_1}^{(1)}\right|^2 + \left|X_{x_2}^{(1)} - C_{x_2}^{(1)}\right|^2$$

$$\left(X^{(1)} - C^{(1)}\right)^2 = |(1.0) - (1.25)|^2 + |(1.0) - (1.5)|^2 = 0.3125$$

$$\left(X^{(2)} - C^{(1)}\right)^2 = \left(d\left(x^{(2)}, C^{(1)}\right)^2\right) = \left|X_{x_1}^{(2)} - C_{x_1}^{(1)}\right|^2 + \left|X_{x_2}^{(2)} - C_{x_2}^{(1)}\right|^2$$

$$\left(X^{(2)} - C^{(1)}\right)^2 = |(1.5) - (1.25)|^2 + |(2.0) - (1.5)|^2 = 0.3125$$

$$\sigma_1^2 = \frac{1}{5}\left(\left(X^{(1)} - C^{(1)}\right)^2 + \left(X^{(2)} - C^{(1)}\right)^2\right) = \frac{1}{2}(0.3125 + 0.3125) = 0.3125$$

$$\boxed{\sigma_1^2 = 0.3125}$$

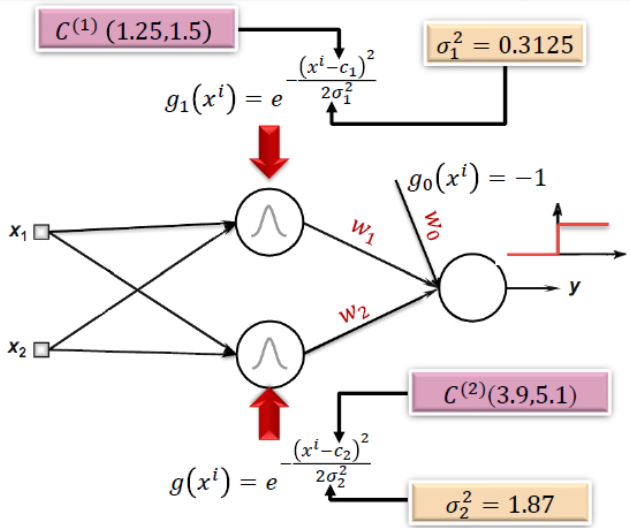
	x_1	x_2	$d\left(X^{(i)}, C^{(1)}\right)$	$d\left(X^{(i)}, C^{(2)}\right)$
$x^{(3)}$	3.0	4.0	3.05	1.42
$x^{(4)}$	5.0	7.0	6.66	2.20
$x^{(5)}$	3.5	5.0	4.16	0.41
$x^{(6)}$	4.5	5.0	4.78	0.61
$x^{(7)}$	3.5	4.5	3.75	0.72

$$\left(X^{(3)} - C^{(2)}\right)^2 = 2.02$$
$$\left(X^{(4)} - C^{(2)}\right)^2 = 4.84$$
$$\left(X^{(5)} - C^{(2)}\right)^2 = 1.618$$
$$\left(X^{(6)} - C^{(2)}\right)^2 = 0.372$$
$$\left(X^{(7)} - C^{(2)}\right)^2 = 0.518$$

$$\sigma_2^2=\frac{1}{5}\left(\left(X^{(3)}-C^{(2)}\right)^2+\left(X^{(4)}-C^{(2)}\right)^2+\left(X^{(5)}-C^{(2)}\right)^2+\left(X^{(6)}-C^{(2)}\right)^2+\left(X^{(7)}-C^{(2)}\right)^2\right)$$

$$\sigma_2^2=\frac{1}{2}(2.02+4.84+1.618+0.372+0.518)=1.87$$

$\sigma_2^2=1.87$



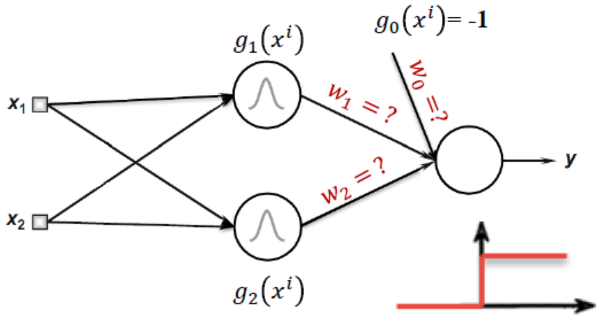
يبين الجدول التالي

$$g_1(x^i)=e^{-\frac{(x^i-c_1)^2}{19.53}} \qquad g_2(x^i)=e^{-\frac{(x^i-c_2)^2}{6.99}}$$

	x_1	x_2	$d(X^{(i)},C^{(1)})$	$d(X^{(i)},C^{(2)})$	$g_1(x^i)$	$g_2(x^i)$
$x^{(1)}$	1.0	1.0	0.56	5.02	0.984	0.027
$x^{(2)}$	1.5	2.0	0.56	3.92	0.984	0.11
$x^{(3)}$	3.0	4.0	3.05	1.42	0.62	0.74
$x^{(4)}$	5.0	7.0	6.66	2.20	0.1	0.5
$x^{(5)}$	3.5	5.0	4.16	0.41	0.41	0.976
$x^{(6)}$	4.5	5.0	4.78	0.61	0.85	0.948
$x^{(7)}$	3.5	4.5	3.75	0.72	0.486	0.928

أن العينة $x^{(1)}$ دخلت $1.0, 1.0$ وخرجت $0.984, 0.027$ أي تعدلت قيمتها وفق تابع الغاوص... وهكذا بالنسبة لبقية العينات.

الآن وصلنا إلى مرحلة الشبكة التقليدية



$$w\leftarrow w+\eta.\left(d^{(k)}-u\right).x^{(k)},$$

$$with\ k=1,\ldots,p$$

وهذه هي النتائج النهائية

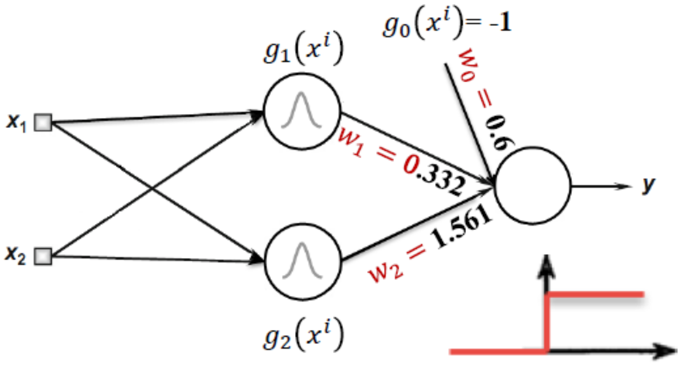
	$g_1(x^i)$	$g_2(x^i)$	d	
$x^{(1)}$	0.984	0.027	A	0
$x^{(2)}$	0.984	0.11	A	0
$x^{(3)}$	0.62	0.74	B	1
$x^{(4)}$	0.1	0.5	B	1
$x^{(5)}$	0.41	0.976	B	1
$x^{(6)}$	0.85	0.948	B	1
$x^{(7)}$	0.486	0.928	B	1

w
1.0
1.0
1.0

$\eta = 0.2$

7

Number Of Training Iteration = 7 ⇒ final w = [0.6, 0.3328,1.5618]



ملخص:

كان لدينا بداية العينات $x^{(1)}$ و $x^{(2)}$ مصنفتين ضمن A والباقي B ونريد البحث عن الأوزان w_0, w_1, w_2 والمتوسطات μ_1, μ_2 والانحراف σ_1, σ_2 التي تحقق هذا التصنيف، فقمنا بتدريب الشبكة كمرحلة أولى في Unsupervised Learning عن طريق خوارزمية k-means وثبتنا المتوسطات والانحراف المعياري ثم اكملنا بتدريب شبكة تقليدية ب Supervised Learning ب 3 مداخل (عصبوني الطبقة المخفية + الانزياح) ومخرج واحد حتى نتجت الأوزان النهائية w_0, w_1, w_2.

- ملاحظة:
- الشبكات العصبونية هذا أساسها والباقي أنواع من الشبكات، كل شبكة حالة خاصة من الحالة العامة، اخذنا في هذه المحاضرة مثال عن RBF وهذه الشبكة حالتها الخاصة أن توابع العصبونات الأولى هي توابع غاوص والشبكة مؤلفة من طبقة واحدة ونقوم بحساب بارامترات تابع الغاوص عن طريق Unsupervised Learning مثل خوارزمية k-means.

تمهيد للمحاضرة القادمة

سنتكلم في المحاضرة القادمة عن CNN وبما أنها تُطبّق على الصور لذلك سنُعطي مقدمة عن الصور.

0	3	2	5	4	7	6	9	8
3	0	1	2	3	4	5	6	7
2	1	0	3	2	5	4	7	6
5	2	3	0	1	2	3	4	5
4	3	2	1	0	3	2	5	4
7	4	5	2	3	0	1	2	3
6	5	4	3	2	1	0	3	2
9	6	7	4	5	2	3	0	1
8	7	6	5	4	3	2	1	0

What a computer sees

What we see

نحن نرى هذه الصورة على أنه قطار ساقط عن المحطة ولكن الكومبيوتر يراها على أنها مصفوفة أرقام.

❖ أنواع الصور:

1. Grayscale Images الصور بتدرجات الرمادي:

نمثلها ب 1 Byte أي 8 bit $\Longleftarrow$ **عدد الخيارات المتاحة** $256 = 2^8$

يتراوح من 0 (00000000) إلى 255 (11111111) حيث 0 يمثل اللون الأسود و 255 تمثل اللون الأبيض وما بينهما تدرج الرمادي حيث الرمادي هو 128 وما دونه هو رمادي غامق وما فوقه هو رمادي فاتح.

2. Binary Images ثنائية اللون (أبيض وأسود):

لتمثيل قيمة ثنائية نحتاج 1 bit إما 0 (أسود) أو 1 (أبيض).

3. Color Images الصور الملونة RGB:

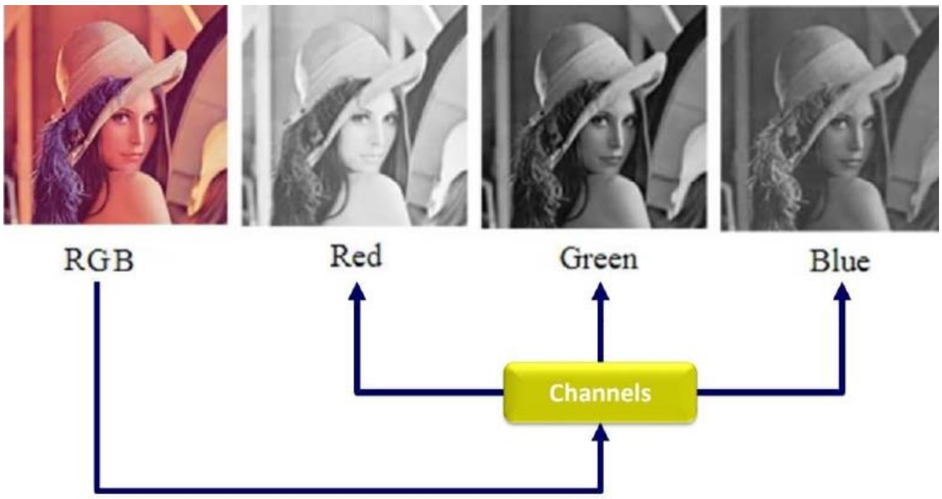
أصغر عنصر بالصورة نسميه pixel وهو اختصار ل picture element وهو عبارة عن 3 Byte (1 Byte للأحمر– 1 Byte للأخضر- 1 Byte للأزرق) أي 24 bits $\Longleftarrow$ **عدد الاحتمالات الممكنة** $2^8 \times 2^8 \times 2^8$ **لون أدناها (00 ... 00)(24 صفر) وهو اللون الأسود وأعلاها (11 ... 11)(24 واحد) وهو اللون الأبيض، وما بينهم جميع الألوان التي نراها والتي يمكن إظهارها.**

Color image 24 bits (RGB) | Gray image 8 bits | Binary image 1 bit

- ملاحظة:
- RGB هي الوان الشاشات بينما الوان الطباعة CMYK حيث:
  RGB: Red Green Blue
  CMYK: Cyan Magenta Yellow Black
- كل الإظهار ⟸ RGB مثال الشاشات
- كل الصباغ (الطباعة) ←CMYK مثال الوان الثياب.

ملخص:

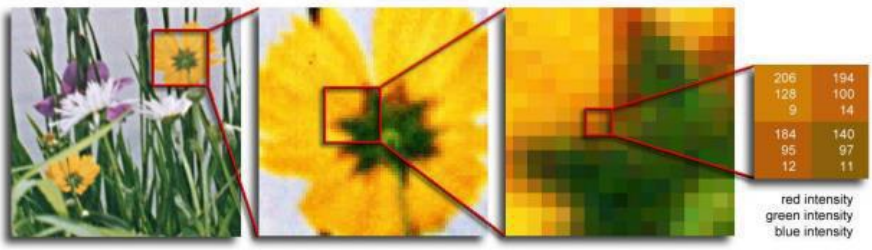
- أصغر عنصر بالصورة هو pixel
- إذا كانت الصورة ملونة ← كل pixel هو 3 Byte
- إذا كانت الصورة رمادية ← كل pixel هو 1 Byte
- أي أن كل صورة ملونة هي عبارة عن 3 صور (قنوات)
- قناة لل Red: قيمها محصورة بين 0-255 وهي صورة رمادية.
- قناة لل Green: قيمها محصورة بين 0-255 وهي صورة رمادية.
- قناة لل Blue: قيمها محصورة بين 0-255 وهي صورة رمادية.



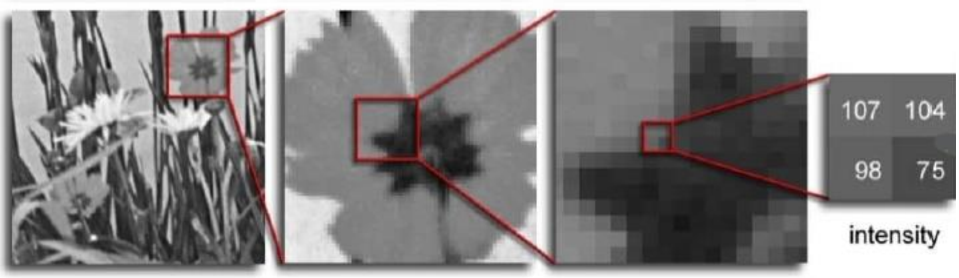
معنى اللون الأبيض في صورة Red أن قيمة المركبة الحمراء فيها عالية جداً

مثال:

كل قيمة عبارة عن 3 Byte لأنها ملونة



أربعة بكسلات كل قيمة محصورة بين 0-255



إذاً الصورة هي مصفوفة والصور الملونة هي ثلاث مصفوفات في الحقيقة (مصفوفة لل Red وأخرى لل Green وأخرى لل Blue)

- فائدة:
- Image Net هي قاعدة معطيات عالمية للصور المصنفة
- تُطبّق نتائج تطوير CNN على Image Net
- الرابط: https://www.image-net.org

انتهت المحاضرة