



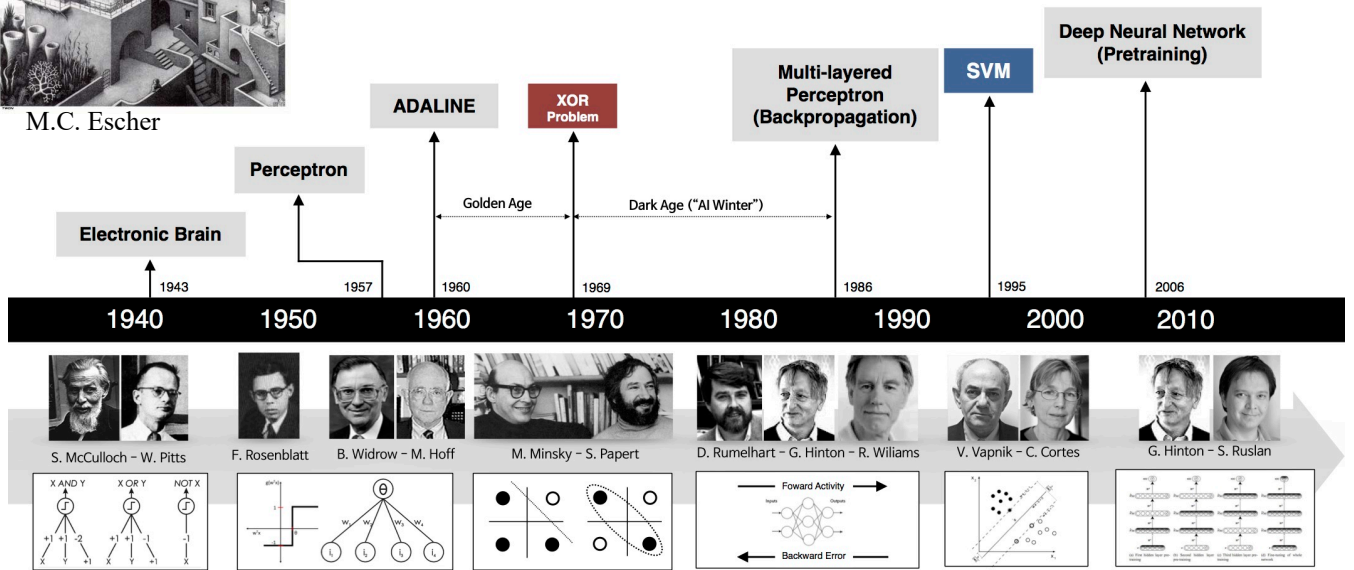
Nome: _____ Matrícula: _____



M.C. Escher

“Intelligence is the ability to adapt to change.”
“AI is likely to be either the best or the worst thing to happen to humanity.”
“The greatest enemy of Knowledge is not Ignorance, is the illusion of Knowledge.”

Stephen Hawking.



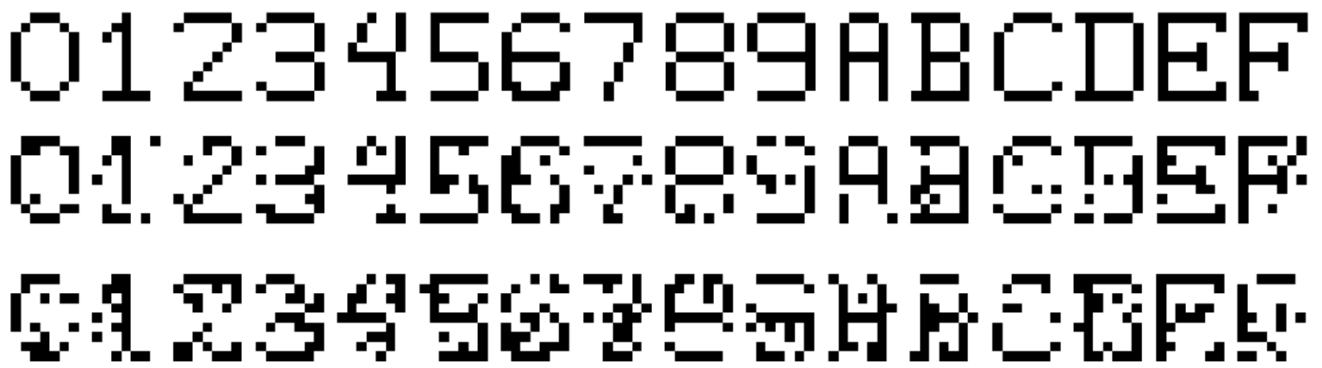
https://beamandrew.github.io/deeplearning/2017/02/23/deep_learning_101_part1.html

1ª PROVA ICIN 1º/2015

1ª Questão (2,5) O exercício 1 de ICIN, utilizou uma Rede Neural Artificial (RNA) para implementar o reconhecimento óptico de caracteres (OCR) de uma matriz binária de pixels 9 x 7. ($2^{63} = 9,2234e+18$).

- (0,5) Por quê uma rede neural treinada com ruído acerta mais do que uma rede que só foi treinada só com os caracteres originais?
- (0,5) Se o conjunto de treinamento é o mesmo, qual o propósito de se variar as condições iniciais?
- (0,5) Em que condições o treinamento pode ficar “paralisado”, isto é, os pesos da RNA não são alterados, apesar de o objetivo de treinamento ainda não ter sido alcançado?
- (1,0) Se em vez de uma matriz binária de pixels, fossem utilizados pixels em níveis de cinza (8 bits por pixel), que diferença haveria na arquitetura da RNA de reconhecimento de OCR? (Entradas, Saídas, Camadas, Conjuntos de Treinamento/Validação/Teste). A informação adicional ajudaria ou dificultaria o reconhecimento OCR? Por quê?

- Os 2^{63} padrões possíveis devem ser atribuídos às 16 classes assumindo-se a “proximidade” aos padrões de treinamento. Padrões ruidosos são “naturalmente” mais próximos aos padrões originais e, uma vez aprendidos, aumentam a capacidade de generalização da rede pois um padrão de validação pode ser visto como a interpolação entre padrões aprendidos.
- Como há mínimos locais em problemas de classificação não-lineares é interessante variar as condições iniciais na busca por bacias de gradiente descendente que tenham menores mínimos locais. O mínimo global raramente é necessário em problemas de classificação de padrões. Um “bom” mínimo resolve problemas típicos de RNAs.
- Em regiões em que o gradiente tende a zero, em um platô na superfície de erro médio quadrático.
- Nenhuma diferença. Entradas entre 0/1 seriam substituídas por 0:255. Ajudaria no reconhecimento, pois o processo de binarização (0/1) joga fora parte da informação analógica (continuidade do traço) dos caracteres manuscritos.



2ª Questão (2,5) LVQ – Learning Vector Quantization.

Considere a distribuição radial de 3 classes { . , x , * }, com 80 valores por classe. Cinco vetores de código são atribuídos inicialmente a cada classe de forma aleatória ($0.4 * \text{randn}()$). Com taxa de aprendizagem 0.1 têm-se, após 150 épocas, os vetores de código treinados. Com uma ‘grade de validação’ é possível visualizar o domínio de quantização dos vetores LVQ.

- (1,0) Apresente no 1º gráfico, considerando a sequência de treinamento indicada {1,2,3,4,5,6,7}, e uma taxa de aprendizagem de 0,5 do algoritmo LVQ1, a nova posição dos vetores de código.
- (1,0) Considerando classificadores de padrões, aponte as principais diferenças entre o LVQ e o MLP:
 - camadas;
 - velocidade de treinamento;
 - capacidade de generalização;
 - número de classes;
 - mapeamento de sistemas não-lineares.
- (0,5) Como a fronteira entre as classes poderia ser melhor definida?
 - Número de amostras por classe;
 - Número de vetores de código;
 - Algoritmo de treinamento.

b)

- A rede LVQ tem duas camadas: a 1a distribui vetores de Código para cobrir a distribuição de probabilidade das classes. A 2a camada agrega vetores da mesma classe. A rede MLP pode ter de uma a muito mais que 20 camadas. A “função” de cada neurônio não é clara (representação subsimbólica).
- O treinamento da MLP é muito mais lento e envolve a adaptação de todos os pesos para um par de treinamento. A LVQ só adapta um vetor de código por par de treinamento.
- A capacidade de generalização da MLP é melhor (foram propostas outras variantes do LVQ para definir melhor as fronteiras).
- O número de classes poderia ser equivalente entre LVQ e MLP, no entanto o treinamento de uma LVQ com 100 classes, exigiria, provavelmente uma quantidade de pontos de treinamento extremamente grande para que as fronteiras das classes sejam definidas de forma razoável.
- Tanto a MLP quanto a LVQ mapeiam sistemas não-lineares.

c)

- Aumentar o número de amostras por classe.
- Aumentar o número de vetores por classe.
- Sugestões propostas por Teuvo Kohonen: algoritmos que movam também o 2º colocado (LVQ2.1). Algoritmos que considerem a distância do ponto de treinamento em relação à classe vencedora e a 2ª colocada (LVQ3). Algoritmos que considerem uma taxa de aprendizagem individualizada (OLVQ).

Obs: LVQ1, Kohonen, 1990:

$W_c(t)$ – neurônio vencedor para o dado de treinamento $X(t)$;

$$W_c(t+1) = W_c(t) + \alpha(t)[X(t) - W_c(t)], \quad \text{se classe}(W_c) = \text{classe}(X)$$

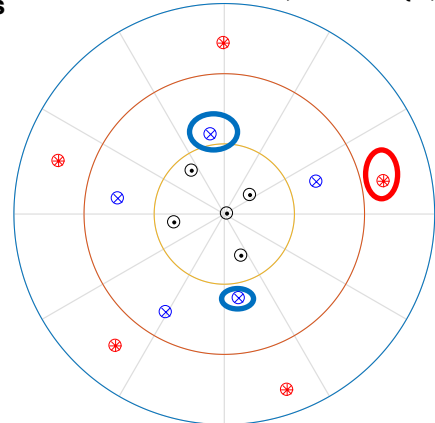
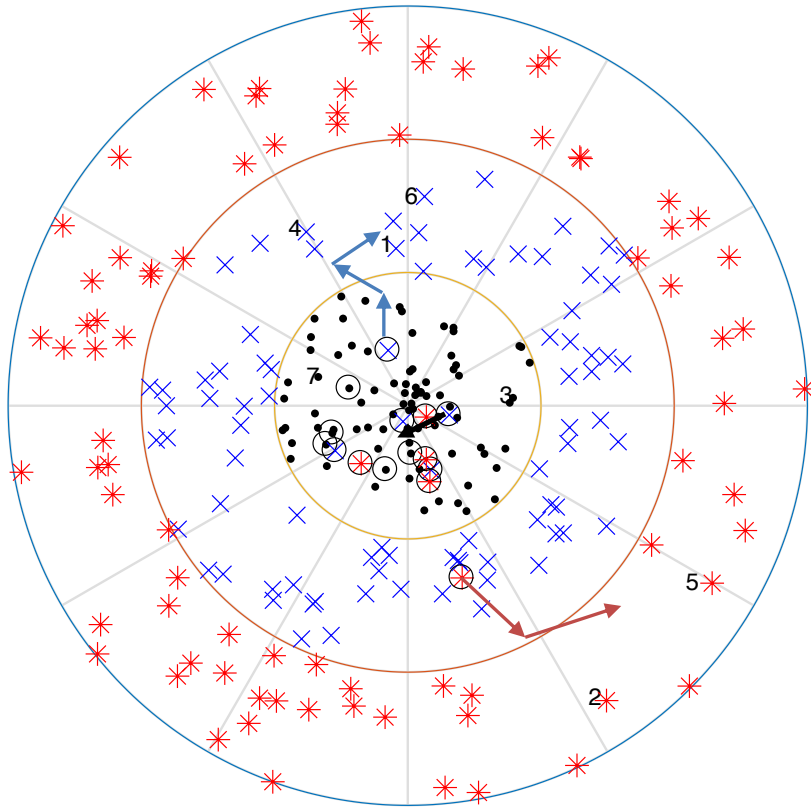
$$= W_c(t) - \alpha(t)[X(t) - W_c(t)], \quad \text{se classe}(W_c) \neq \text{classe}(X)$$

$$W_j(t+1) = W_j(t) \text{ para todo } j \neq c$$

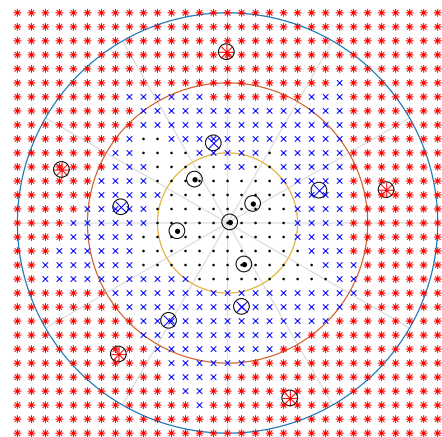
$0 < \alpha(t) < 1$, taxa de aprendizagem, usualmente $\alpha(0) = 0,1$ decaindo linearmente.

LVQ: train values, classes = { . , x , * }; o - Random Initialized Vectors

LVQ: o - trained vectors, classes = { . , x , * };



LVQ: o - trained vectors; validation grid, classes = { . , x , * };



- a) Em {1, 4, 6} o mesmo vetor de Código vencedor **x** é levado para cima. Em {2, 5} o vetor de Código vencedor ***** é levado para a direita. Em {3} o vetor de Código vencedor **.** é afastado, por ser de outra classe. À direita, em cima, mostra-se uma posição possível destes três vetores de código, ao final do treinamento.

3ª Questão (2,5) Uma memória associativa de Hopfield deve armazenar os seguintes padrões binários:

1	1	
1	1	

		1
		1
		1

		1
1	1	
		1

Cada padrão binário é dado por: $A_p = [a_1^p \ a_2^p \ \dots \ a_9^p]$, com $a_i^p = 0$ ou 1 , $i=1:9$, $p=1:3$. Considere o limiar de disparo dos neurônios $L_j = 0$. A equação de treinamento da rede de Hopfield binária é: $w_{ij} = \sum_{p=1}^m (2a_i^p - 1)(2a_j^p - 1)$

- a) Obtenha a matriz de pesos correspondente à rede de Hopfield.
 b) Considerando a operação seqüencial da rede de Hopfield, calcule o padrão que a rede fornece após a sua estabilização para o seguinte padrão inicial:

1		

$$W = \begin{bmatrix} 0 & 3 & -1 & -1 & -1 & 1 & 1 & 1 & -1 \\ 3 & 0 & -1 & -1 & -1 & 1 & 1 & 1 & -1 \\ -1 & -1 & 0 & -1 & -1 & 1 & -3 & -3 & 3 \\ -1 & -1 & -1 & 0 & 3 & -3 & 1 & 1 & -1 \\ -1 & -1 & -1 & 3 & 0 & -3 & 1 & 1 & -1 \\ 1 & 1 & 1 & -3 & -3 & 0 & -1 & -1 & 1 \\ 1 & 1 & -3 & 1 & 1 & -1 & 0 & 3 & -3 \\ 1 & 1 & -3 & 1 & 1 & -1 & 3 & 0 & -3 \\ -1 & -1 & 3 & -1 & -1 & 1 & -3 & -3 & 0 \end{bmatrix}$$

EP disparado	Soma do EP	Saída do EP	Novo vetor de saída								
1	-1	0	0	0	0	1	0	0	0	0	0
2	-1	0	0	0	0	1	0	0	0	0	0
3	-1	0	0	0	0	1	0	0	0	0	0
4	0	1	0	0	0	1	0	0	0	0	0
5	3	1	0	0	0	1	1	0	0	0	0
6	-6	0	0	0	0	1	1	0	0	0	0
7	2	1	0	0	0	1	1	0	1	0	0
8	5	1	0	0	0	1	1	0	1	1	0
9	-8	0	0	0	0	1	1	0	1	1	0
1	0	0	0	0	0	1	1	0	1	1	0
2	0	0	0	0	0	1	1	0	1	1	0
3	-8	0	0	0	0	1	1	0	1	1	0
4	5	1	0	0	0	1	1	0	1	1	0
5	5	1	0	0	0	1	1	0	1	1	0
6	-8	0	0	0	0	1	1	0	1	1	0
7	5	1	0	0	0	1	1	0	1	1	0
8	5	1	0	0	0	1	1	0	1	1	0
9	-8	0	0	0	0	1	1	0	1	1	0

Padrão final obtido:

0	0	0
1	1	0
1	1	0

4ª Questão (2,5) Assinale (V) Verdadeiro ou (F) Falso. Caso considere um item falso, justifique todos os aspectos que considerar incorretos.

- (0,5) Mostra-se que RNAs são aproximadores universais. Isto significa que, dado um conjunto de treinamento adequado, uma aproximação com precisão arbitrária pode sempre ser encontrada.
- (0,5) Um excelente programa que joga xadrez não pode ser considerado IA, pois não tem capacidade de se adaptar. Por exemplo, não consegue aprender a jogar damas.
- (0,5) Não são as redes neuronais que moldam os pensamentos, mas os pensamentos é que moldam as redes neuronais.
- (0,5) A memória nos seres vivos é auto-associativa, enquanto os computadores, só usam memória endereçada.
- (0,5) A interpolação e extrapolação são aprendidas. As ilusões de ótica (e.g., M.C. Escher), mostram que aprendemos “errado” ou de forma “incompleta”.

- F - Como Aproximador Universal entende-se que exista *uma Topologia* (número de camadas, número de neurônios por camada, escolha das funções de ativação), *um Algoritmo* (condições iniciais, forma de ajustar os pesos, número de épocas) e *Conjuntos de Treinamento/Teste* (suficientemente ricos em informação que abarquem o domínio da função mapeado) que permitam resolver o problema (encontrar pesos) com a precisão arbitrária desejada (Conjunto de Validação). Ser um Aproximador Universal não dá, no entanto, nenhuma indicação de **projeto**, *qual topologia*, *qual algoritmo* ou quais *conjuntos de Treinamento/Teste* permitem obter a *precisão desejada no conjunto de validação*.
- V – Segundo Sephen Hawkin.
F – O programa de Xadrez pode ter sido desenvolvido utilizando aprendizagem. Na fase de utilização, como numa RNA, os “pesos” são congelados, não havendo mais adaptação.
- V – “Your brain does not manufacture thoughts. Your thoughts shape neural networks.” — Deepak Chopra
Em RNAs os dados de treinamento/teste moldam os pesos da RNA. Os pesos não modificam os conjuntos de treinamento/teste.
- V – a nível de hardware toda a memória é endereçada.
F – é possível simular uma rede de Hopfield em um programa de computador. A operação deste simulador da rede de Hopfield é auto-associativa.
De forma mais genérica: Computadores são utilizados para realizar IA.
- F – Não aprendemos “errado” ou de forma “incompleta”. O conjunto de treinamento que utilizamos, nos prepara para lidarmos com o mundo “real” em que vivemos. As imagens “sintéticas” que Escher produziu não fazem parte deste conjunto e nunca ocorrem no mundo 3D real.
As imagens de Escher exploram a forma como o cérebro humano extrapola (generaliza) uma imagem 2D para o modelo 3D mais simples e tem propósito artístico. Tem coerência local, mas são um paradoxo global.