

IA004 – Redes Neurais II

Primeira Lista de Exercícios – Peso 2

Data de Entrega: 12/09 Data de Devolução: 28/09

Aluno: Sergio Vieira Bueno, ra: 072372
Professor: Romis Ribeiro de Faissol Attux

As questões e os dados estão disponíveis em: <http://www.dca.fee.unicamp.br/~attux/>

Os programas bem como este relatório com as figuras em cores estão disponíveis em: <https://github.com/svbueno/RedesNeuraisII>

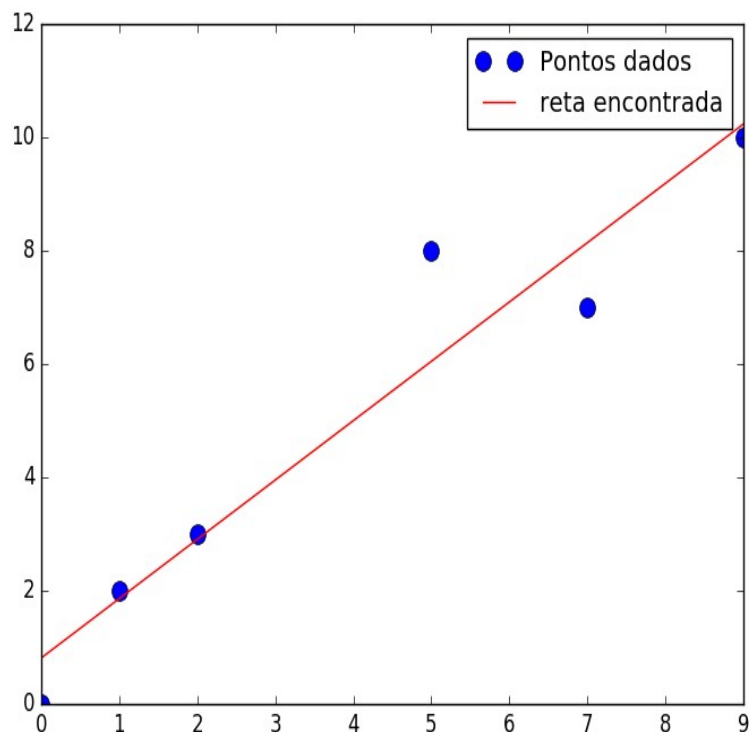


Figura 1 - Reta estimada pelo EQM ótimo,
 $a = 1.046875$, $b = 0.8125$, $EQM = 0.9765625$

Exercício 1

a e b) em anexo

c) Assumindo que o valor y observado advém de uma distribuição gaussiana quando de um dado valor de x , a reta é estimada ao maximizar a verossilhança dos valores observados ao se amostrar x , assim, tal reta apresenta um EQM ótimo, que pode ser calculado determinísticamente pelo método dos mínimos quadrados.

Exercício 2

a) Podemos ver pela figura 2a que os dados aparentam ser linearmente separáveis.

b) Uma vez que o algoritmo perceptron é capaz de separar, com erro zero, dados linearmente separáveis (Attux_2016), podemos constatar pela figura 2b1 que os padrões dos dados estudados em questão são linearmente separáveis, pois, a figura indica que os erros de classificação obtidos chegam a zero (e se mantêm neste patamar) após um número adequado de épocas. Constatamos ainda tal separabilidade, dada pelo treinamento do perceptron ao observarmos a figura 2b2, que mostra o hiperplano de separação (no caso um plano) o qual, ao ser estimado pelo referido algoritmo, separa perfeitamente o conjunto de dados e, portanto, deixa os pontos pertencentes a classes '+1' (em vermelho no gráfico) de um lado do plano e os pontos pertencentes a classe '-1' (em azul no gráfico) do outro lado.

c) Podemos ver pela figura 2b1 que, variando o α , apenas o tempo de convergência para zero se altera.

d) Ao se variar as condições iniciais, foi observado um comportamento da evolução do erro diferente para o caso de α igual a 0.01, contudo não foram observadas diferenças para os outros valores de α s. Tal observação indica, que a convergência é sensível às condições iniciais no que se refere ao seu tempo, mas não ao seu erro.

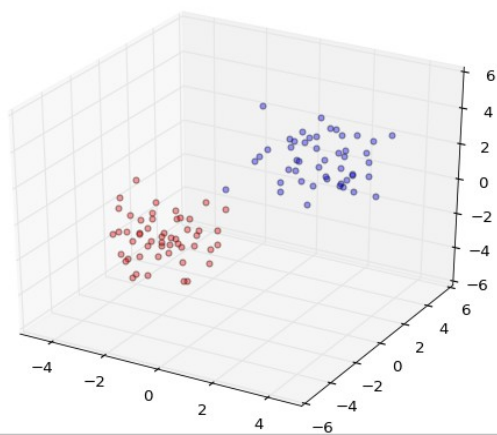


Figura 2a – Representação de “dados1” com “+1” em vermelho e “-1” em azul.

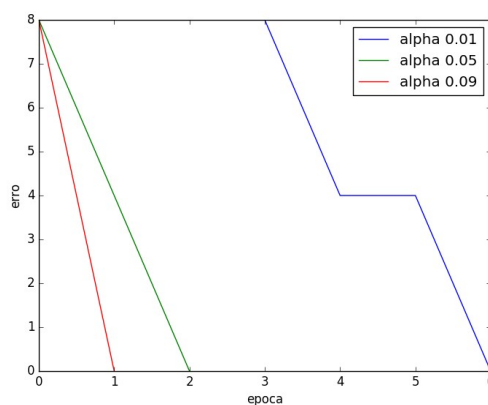


Figura 2b1 - Evolução do número de erros por época de algoritmo de treinamento do perceptron com diferentes α s (passos), convergentes à zero.

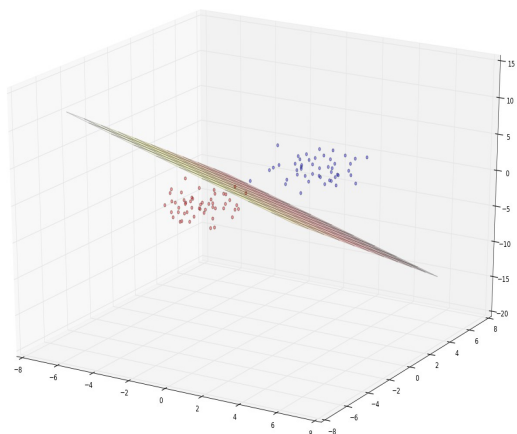


Figura 2b2 – Representação de “dados1” tendo como pano de fundo o hiperplano ótimo obtido pelo algoritmo perceptron, que separa perfeitamente os dados.

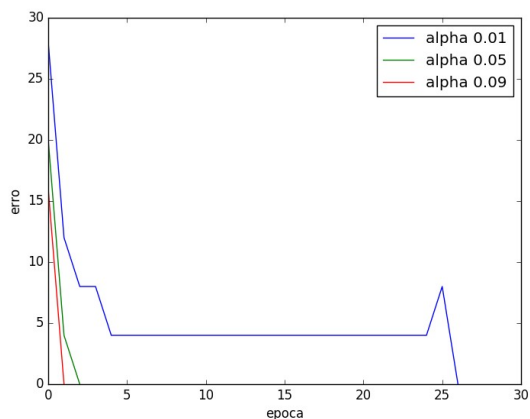


Figura 2d - Evolução do número de erros por época de algoritmo de treinamento do perceptron com diferentes α s (passos) com condição inicial distinta da que gerou a figura 2b1, também convergentes à zero.

Exercício 3

a) Para o projeto de uma rede neural MLP que utiliza o algoritmo do gradiente e atualiza os pesos em modo conhecido como batelada, primeiramente foi observado o conjunto dos valores desejados, que podem ser vistos na figura 3a1.

Como vemos que não se trata de uma superfície por demais complicada e como, segundo Haykin_1994, uma rede MLP com uma única camada já possui capacidade de aproximação universal, foi escolhido uma rede com apenas um camada intermediária.

Para definir um valor adequado para o número n de neurônios na camada intermediária, foi feita uma varredura na base 2, começando com dois neurônios e terminando com 128. Foram ainda utilizados valores de passo $\alpha = 0.1$, com um número máximo de iterações de 500 épocas. A varredura foi realizada 3 vezes e utilizou-se a média dos valores de EQM das 3 varreduras.

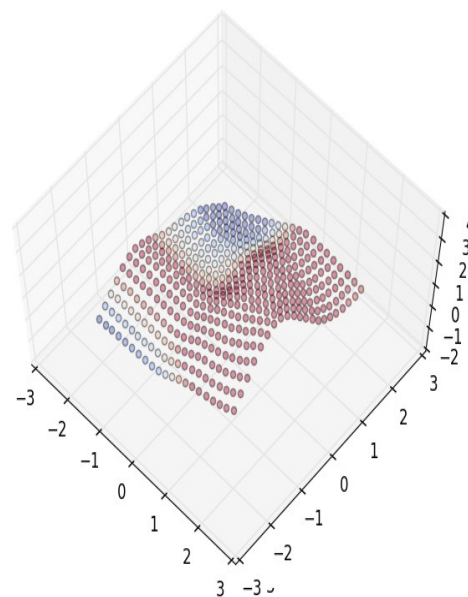


Figura 3a1 - Representação de “dados2”.

Para cada n foi utilizada a metodologia de validação cruzada com 3 partições. A figura 3a2 mostra os valores de erro de validação para a rede neural com 16 neurônios intermediários. Mais adiante, veremos o porquê da escolha deste número de neurônios como exemplo.

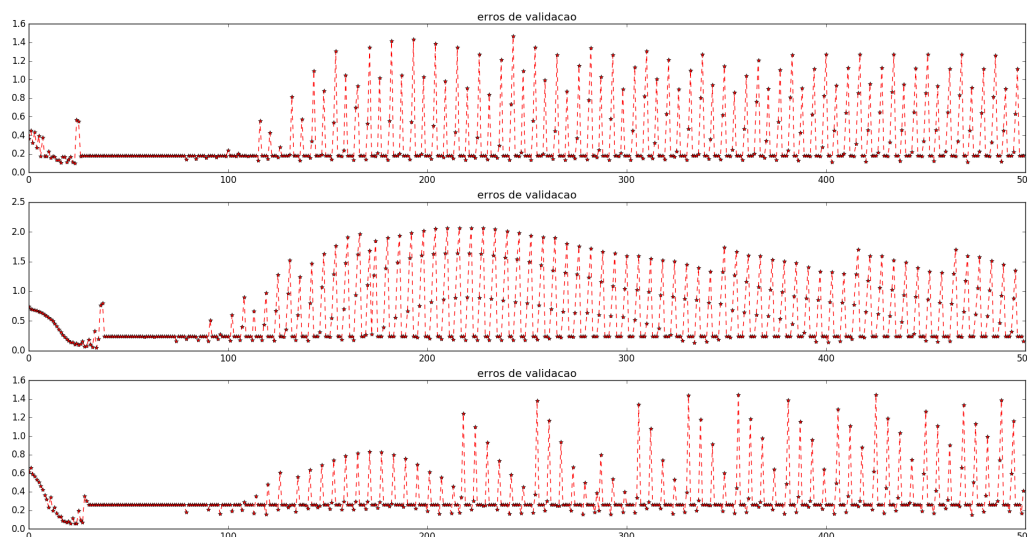


Figura 3a2 – Evolução dos EQMs de validação com o número de épocas das 3 partições da rede neural com 32 neurônios.

Nota-se que os valores de erro de validação começam decaindo com o passar das épocas, contudo, tais valores não diminuem até zero e passam a aumentar com o avançar do tempo. Nota-se ainda que para a rede em questão tal aumento ocorre por volta de 30 épocas. Os valores de EQM para todas as configurações para as redes avaliadas seguem na Tabela 1, que indica que a rede com 32 neurônios intermediários apresentou (em média) o menor valor de EQM de validação.

Tabela 1 – Valores de EQM para diferentes números de neurônios na camada intermediária da rede MLP.

rodadas/neuronios	2	4	8	16	32	64	128
primeira	0.022	0.011	0.036	0.011	0.009	0.155	0.088
segunda	0.038	0.021	0.014	0.017	0.029	0.117	0.091
terceira	0.022	0.012	0.012	0.010	0.024	0.116	0.091
media	0.027	0.015	0.021	0.013	0.020	0.130	0.090

Em Sheela_2013, após uma análise de mais de cem estratégias para se encontrar um número ótimo de neurônios na camada intermediária, os autores propõem a utilização da equação abaixo:

$$n = \frac{4q^2 + 3}{q^2 + 8} ,$$

onde a n é o número ótimo de neurônios na camada intermediária e q o número de atributos da entrada.

Uma vez que temos dois atributos na entrada, a utilização de tal formula resulta em n=2. Contudo, este número já foi contemplado na varredura, e resultou em um valor médio de EQM mais alto do que o advindo da rede com 16 neurônios. É interessante notar que o maior EQM obtido foi aquele através da rede com maior número de neurônios, o que indica que para um melhor desempenho não basta apenas se utilizar de mais neurônios na camada intermediária.

Por fim, achamos interessante notar que embora tenhamos obtido um baixo valor médio de EQM para MLP com n=16, vemos na figura 3a3 um exemplo (obtido em uma das três rodadas em uma das partições) no qual a superfície estimada não foi capaz de “se contorcer” a fim de acompanhar as não linearidades da região inferior direita da superfície a ser estimada.

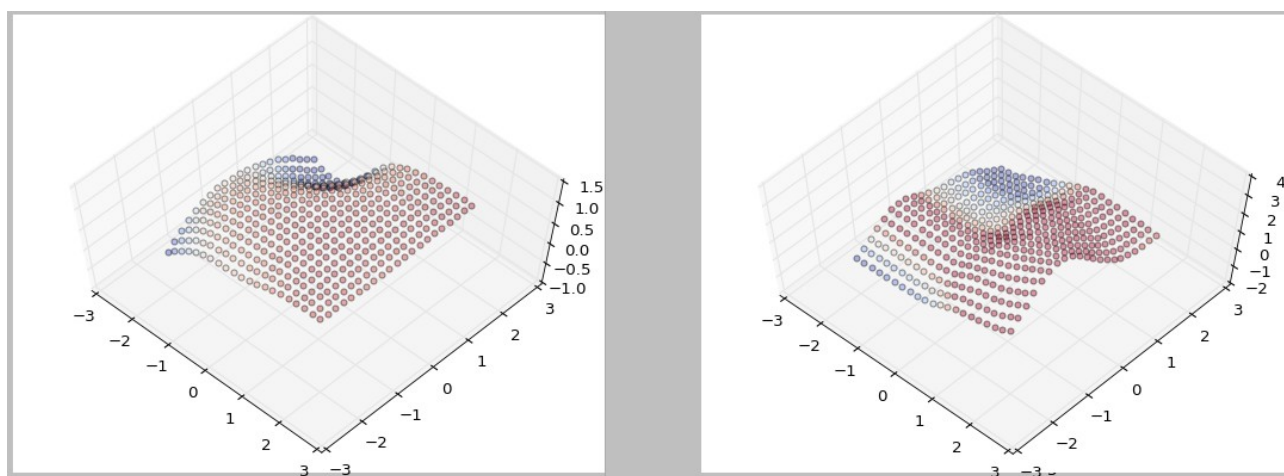


Figura3a3 – Superfície estimada pela MLP à esquerda e superfície que se deseja mapear à direita.

Também foram realizados testes com uma Extreme Learning Machine com regularização (conforme Huang et al., 2012), cujos valores médios das EQM obtidas (também por validação cruzada com três partições) podem ser vistos na tabela 2. Uma vez que os pesos entre os dados de entrada e a camada intermediária são aleatórios, faz-se útil rodar o algoritmo de treinamento algumas vezes. Escolhemos rodar 100 vezes para cada configuração de camada intermediária.

Tabela 2 – Valores de EQM para diferentes números de neurônios na camada intermediária da rede ELM.

rodadas/neurônios	2	4	8	16	32	64	128
	0.073	0.008	0.014	0.044	0.104	0.024	0.055
	0.024	0.094	0.020	0.040	0.049	0.026	0.013
	0.027	0.072	0.061	0.054	0.106	0.163	0.082
	0.041	0.058	0.032	0.046	0.086	0.071	0.050

Podemos observar que o melhor número de neurônios obtidos foi oito, onde obteve-se um erro maior do que o EQM apresentado pela rede MLP, porém o tempo de execução e a complexidade de implementação da ELM são consideravelmente menores. Além disso, vemos pela figura 3b que a ELM foi capaz de “contorcer” sua superfície estimada na região inferior direita da figura, sendo que, o mesmo nem sempre foi possível para a MLP, conforme vimos na figura 3a3.

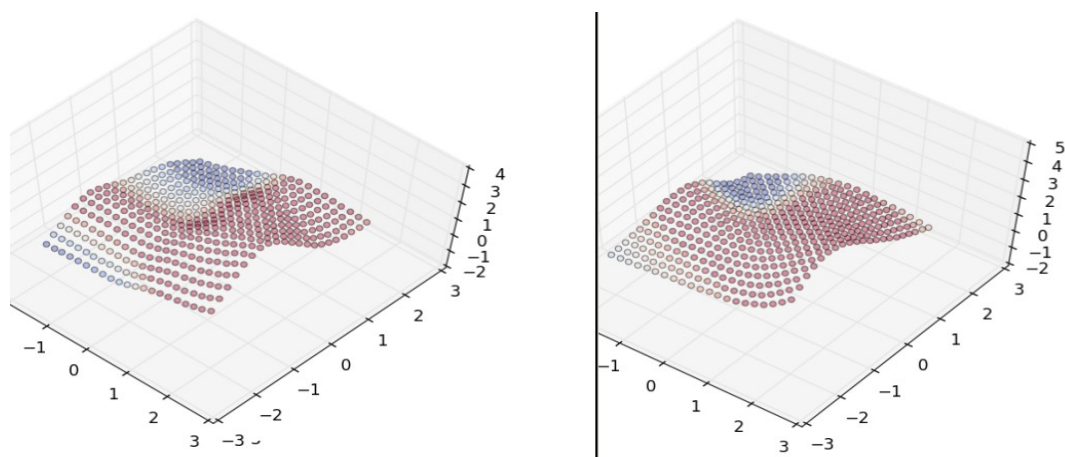


Figura3b – Superfície estimada pela ELM à esquerda e superfície que se deseja mapear à direita.

As redes projetadas nos itens a) e b) deste exercício 3 foram aplicadas aos dados contidos no arquivo “dados3”, que foi gerado com outra amostragem da mesma função. Obtivemos um EQM de 0.035 para a ELM e de 0.026 para a MLP, sendo estes, valores condizíveis com os valores obtidos anteriormente durante os cálculos dos erros de validação. Podemos ver as superfícies estimadas, bem como a que se deseja estimar na figura 3c.

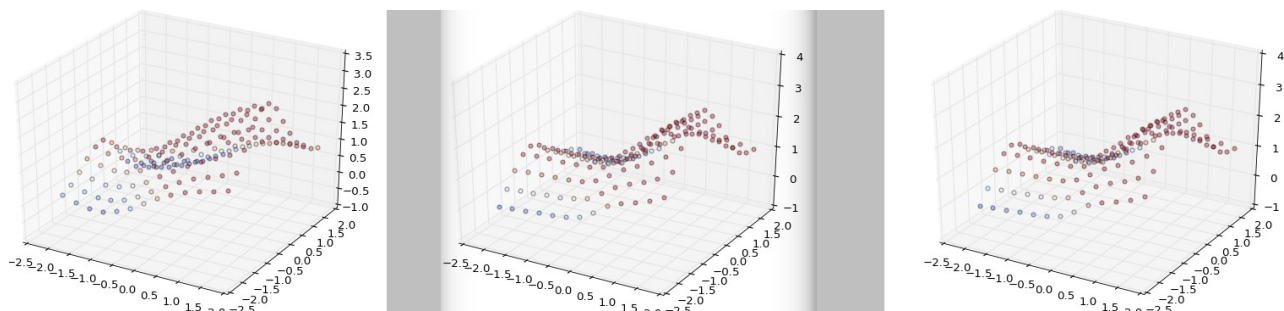


Figura3c – Superfícies que se deseja mapear à esquerda, superfície estimada pela MLP no centro e superfície estimada pela ELM à direita.

Para o exercício quatro em que se deseja projetar um mapeamento para os “dados_map”, foi projetada uma rede RBF, onde utilizou-se a estratégia de se adaptar, em separado, os parâmetros relativos à camada intermediária e os parâmetros relativos à camada de saída. Os primeiros foram utilizadas funções gaussianas como funções de base radial, cujas médias (ou centros) são nós obtidos por meio de um procedimento não-supervisionado conhecido com k-means, e, fixos estes, os segundos foram escolhidos por meio de uma estratégia de mínimos quadrados.

A busca pelo número de neurônios na camada intermediária se deu aos moldes do que fora realizado durante os projetos das redes MLP e ELM. Assim, utilizou-se da metodologia da validação cruzada com 3 partições. A tabela 3 mostra os valores de EQM obtidos.

Tabela 3 – Valores de EQM para diferentes números de neurônios na camada intermediária da rede RBF.

rodadas/neuronios	2	4	8	16	32	64	128
primeira	0.427	0.025	0.010	0.005	0.003	0.001	0.000
segunda	0.399	0.026	0.010	0.004	0.003	0.001	0.000
terceira	0.462	0.025	0.010	0.006	0.004	0.003	0.000
media	0.430	0.026	0.010	0.005	0.003	0.002	0.000

Vemos que os EQM ficaram muito baixos, o que leva a uma estimacão da superfície bastante próxima da desejada, conforme podemos ver no conjunto de imagens das figuras 3c1 e 3c2, onde utilizamos, respectivamente $n=8$ e $n=128$. Vemos que para $n=8$ as superfícies estimadas ficaram já bastante próximas das desejadas, enquanto que para $n=128$, ficaram visualmente quase idênticas.

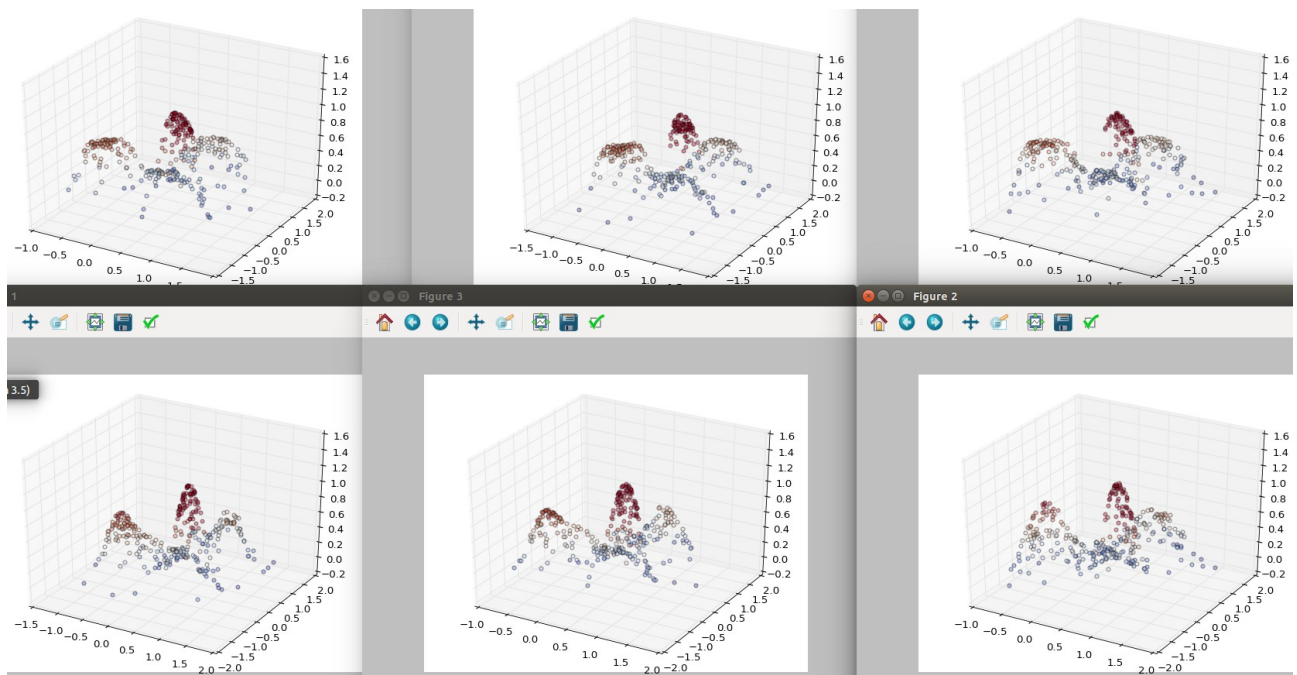


Figura3c1 – Superfícies que se deseja mapear para cada uma das três partições acima e, em baixo, as respectivas superfícies estimadas pela RBF com 8 neurônios na camada intermediária ajuste dos centros pelo método k-means.

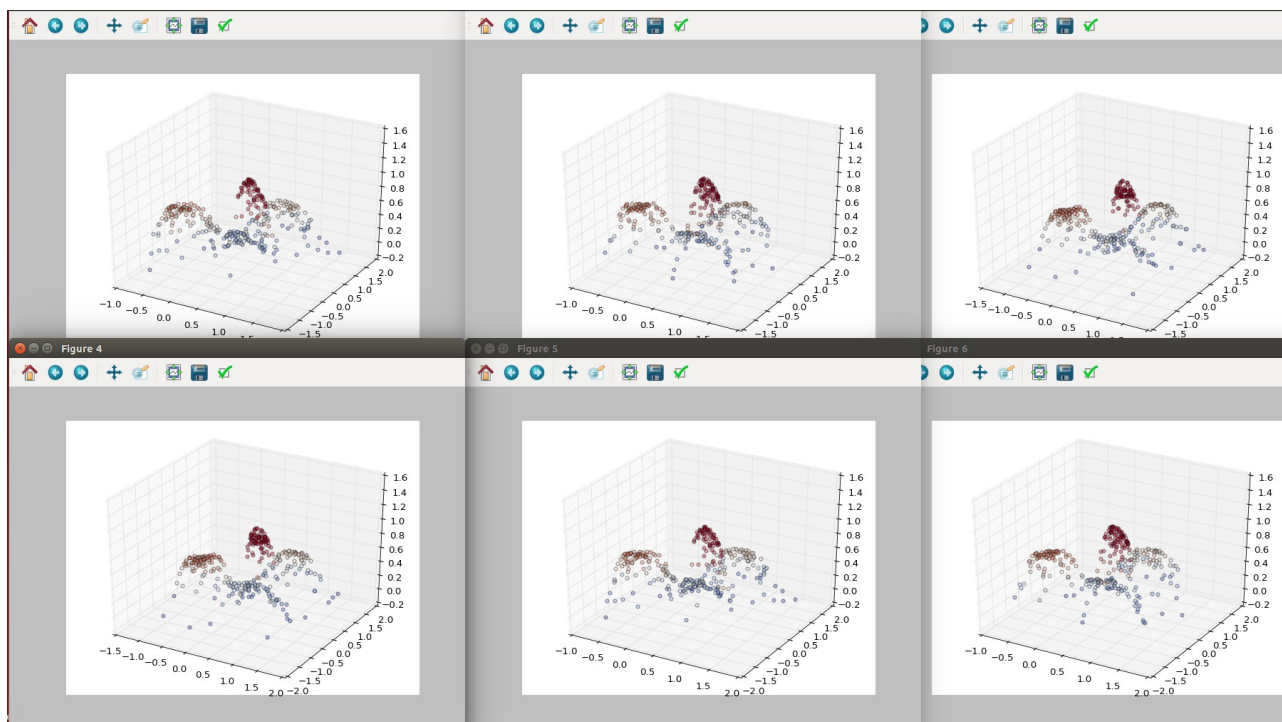


Figura3c2 – Superfícies que se deseja mapear para cada uma das três partições acima e, em baixo, as respectivas superfícies estimadas pela RBF com 128 neurônios na camada intermediária ajuste dos centros pelo método k-means.

Por fim, foram avaliadas redes RBF com ativação aleatória dos centros das gaussianas que descrevem em contraponto com a estratégia de iniciação pelo kmeans. A tabela 4 mostra os valores de EQM obtidos.

Tabela 4 – Valores de EQM para diferentes números de neurônios na camada intermediária da rede RBF.

rodadas/neuronios	2	4	8	16	32	64	128
primeira	0.199	0.155	0.039	0.024	0.008	0.003	0.009
segunda	0.371	0.062	0.126	0.026	0.003	0.001	0.001
terceira	0.142	0.146	0.072	0.058	0.007	0.000	0.001
media	0.237	0.121	0.079	0.036	0.006	0.001	0.003

Podemos ver, ao compararmos as tabelas 3 e 4, que apesar dos EQMs mínimos obtidos pelas duas estratégias de inicialização dos centros das gaussianas (kmeans e aleatório), a estratégia que se utiliza da técnica de kmeans já apresenta erros relativamente baixos para uma pequena quantidade de neuronios na camada intermediária, o que reduz consideravelmente o tempo de treinamento.

Referências:

Attux, R. R. F., Notas de Aula do Curso IA004 – Redes Neurais II, FEEC/UNICAMP, 2016.
 Haykin, S., “*Neural Networks: a Comprehensive Foundation*”, MacMillan, 1994.
 Huang, G., Q. Zhu, C. Siew, “Extreme Learning Machine: a New Learning Scheme of Feedforward Neural Networks”, Anais da IEEE International Joint Conference on Neural Networks, 2004.
 Sheela, K. Gnana, and S. N. Deepa. "Review on methods to fix number of hidden neurons in neural networks." Mathematical Problems in Engineering (2013),