

**ATIVIDADE AVALIATIVA  
PACOTE RSNNS**

**ALUNO: FELIPPE VELOSO MARINHO  
MATRÍCULA: 2021072260  
DISCIPLINA: REDES NEURAIS ARTIFICIAIS**

Os exercícios retratados no relatório abaixo são para realizar estudos no pacote RSNNS (R Stuttgart Neural Networks Simulator). Para o exercício, nos é apresentado um problema de regressão simples com MLPs.

O problema considera uma aproximação de funções  $fg(x) = \text{seno}(x)$  no intervalo  $0, 2\pi$  em que um ruído gaussiano  $\epsilon_i$  é acrescido ao valor amostrado de  $e$   $y_i = fg(x_i)$ , ou seja,  $y_i = \text{seno}(x_i) + \epsilon_i$ .

Devido a este ruído, há uma incerteza nos dados. Portanto, o objetivo é aproximar a função geradora dos dados, apesar do ruído e ajustando a complexidade da rede neural observando casos que se aproximam de over-fitting ou under-fitting. Para isso, será utilizada a função de ativação linear (Função identidade).

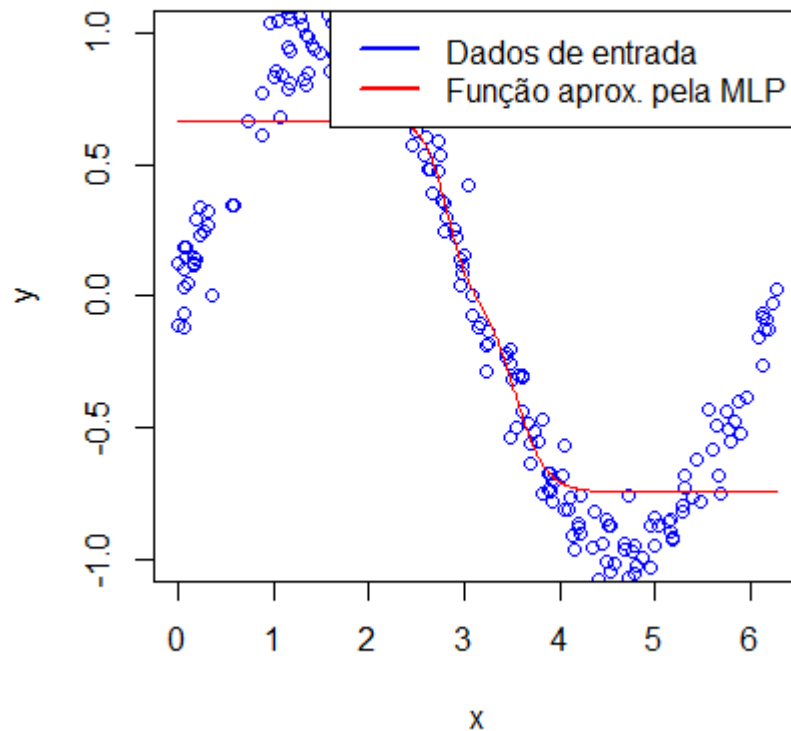
### **Implementação**

Para a implementação, foi utilizados os códigos dados como referência no enunciado com pequenas alterações.

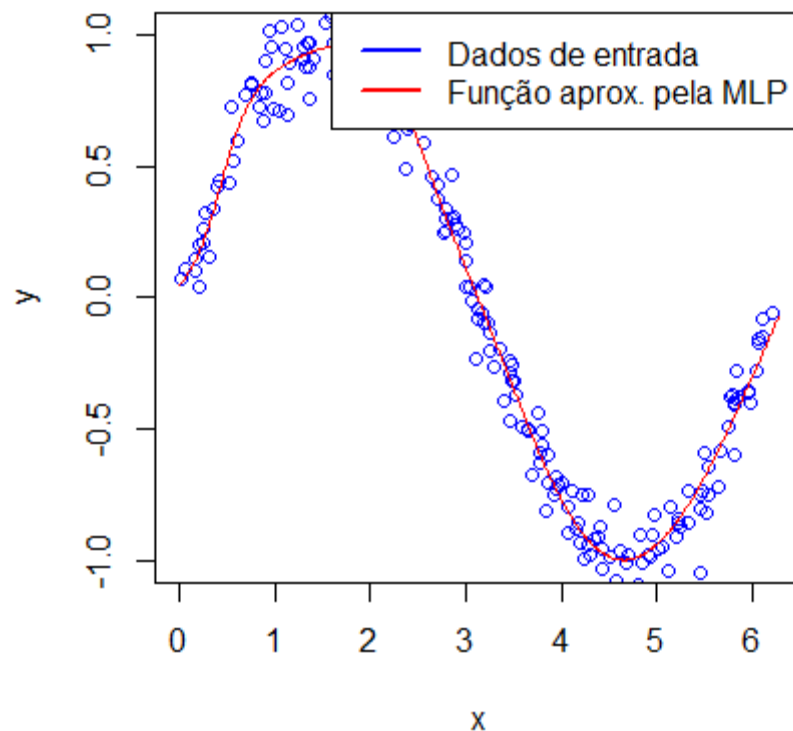
A função `mlp` foi colocada dentro da função chamada `treinar_mlp` que somente pega  $x$  e  $y$  pelo parâmetro e retorna a própria rede. Também foi adicionado a rede, o atributo “`outputActFunc`” que define a função de ativação na saída da rede.

### **Análise do comportamento da rede**

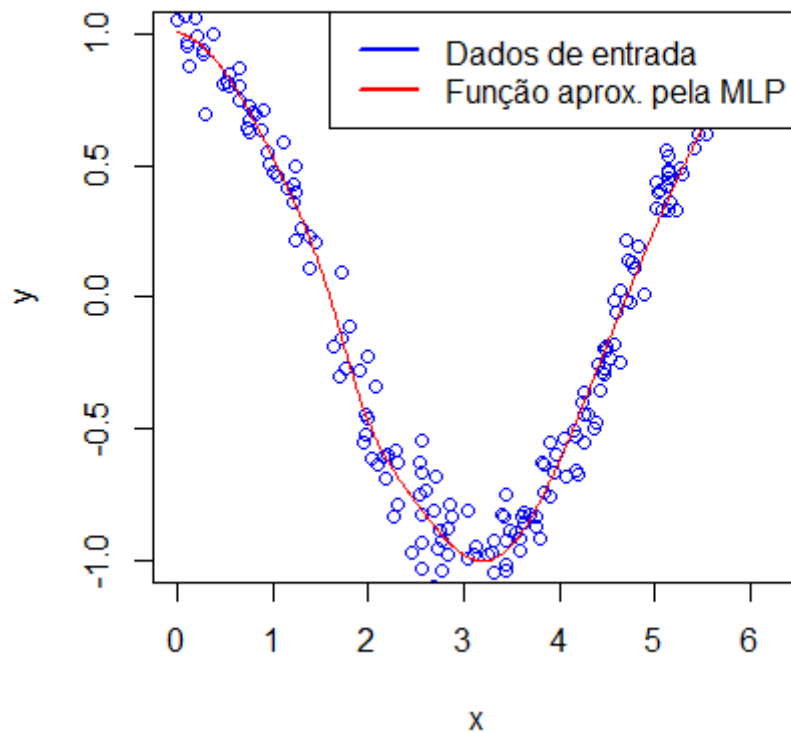
Utilizando a função identidade como função de ativação da saída da rede com somente 2 neurônios, temos um caso de under-fitting. Onde não temos a quantidade de neurônios suficientes para que o modelo realize uma aproximação satisfatória aos dados de entrada.



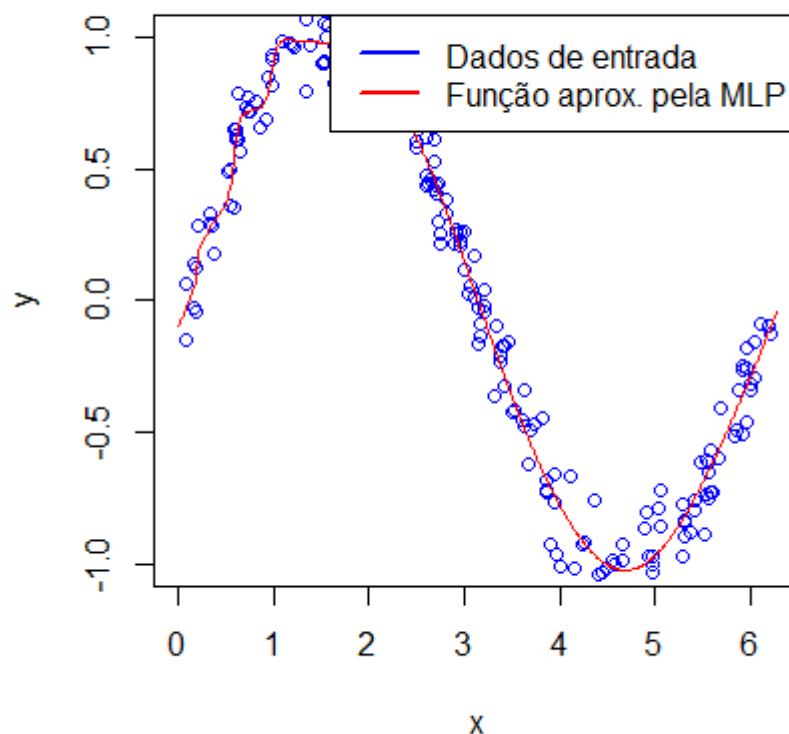
Ao testarmos com 10 neurônios, o cenário muda completamente. A curva se ajusta de maneira bem mais satisfatória ao problema do seno .



Realizando o teste com diferentes valores de entrada, no caso representando a função cosseno. Podemos ver que o modelo se ajusta bem a diferentes tipos de problema, apresentando assim uma boa generalização neste caso.



Ao aumentar gradativamente a quantidade de neurônios e número de epochs, os gráficos começam a demonstrar características de overfitting, com a função aproximada pela MLP generalizando pouco os dados de entrada.



## Conclusão

Por fim, é notável que treinar por muitas epochs pode levar ao overfitting. Monitorar o erro de validação pode ajudar a determinar o ponto ótimo para interromper o treinamento. Ao testar a rede com diferentes

conjuntos de dados (ex.: funções seno e cosseno) foi possível verificar melhor a capacidade de generalização da rede que se mostrou eficaz quando o número de epochs e neurônios estavam “adequados”.

Para evitar problemas de under e over fitting poderiam ser utilizadas algumas técnicas como o monitoramento do erro de validação. Se o erro de validação não diminuir após um certo número de epochs (patience), o treinamento é interrompido. Se não, o treinamento continua até que o erro de validação pare de melhorar.