

Laboratório Aprendizagem de Máquina

Lab5: Multilayer Perceptron (MLP) para classificação binária

Avalie a rede neural MLP para a classificação da base IMDB em anexo. A rede em anexo contém uma camada escondida com 50 neurônios. A saída com duas possibilidades contém uma função de ativação softmax. Escreva um relatório avaliando:

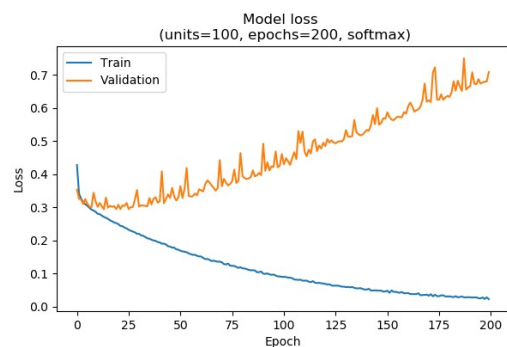
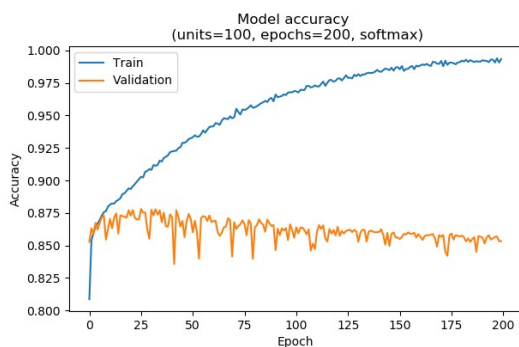
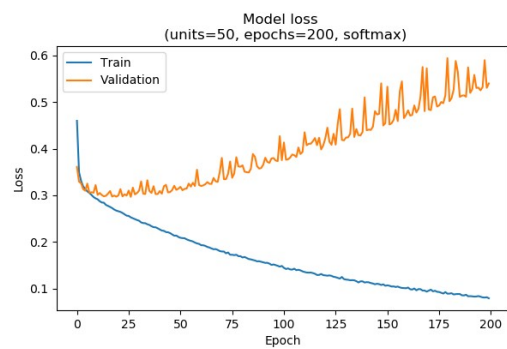
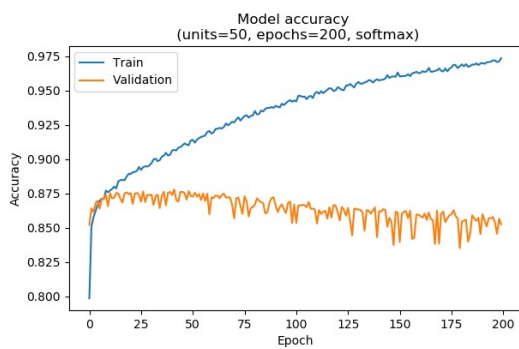
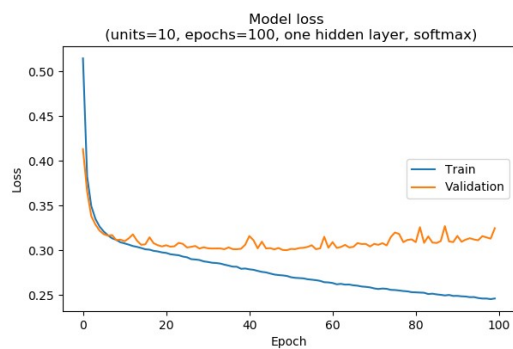
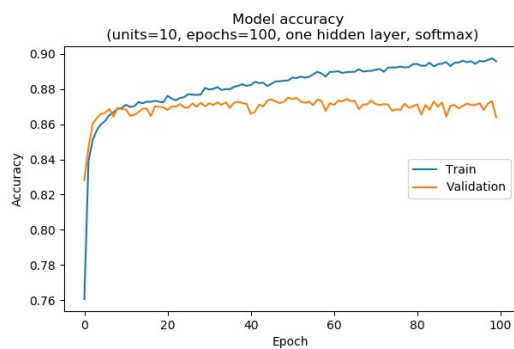
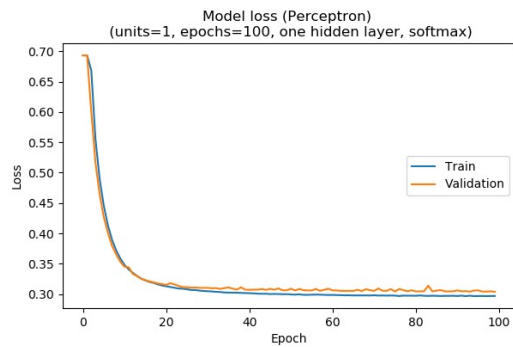
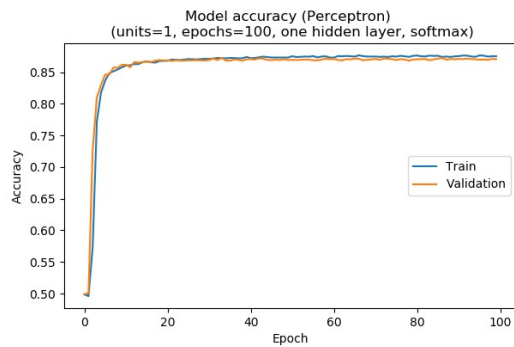
1. A quantidade de neurônios na camada escondida.
2. Quantidade de camadas escondidas.
3. Número de épocas de treinamento.
4. Overfitting da rede.
5. Qual arquitetura (simples ou complexa) entra em overfitting com mais facilidade?

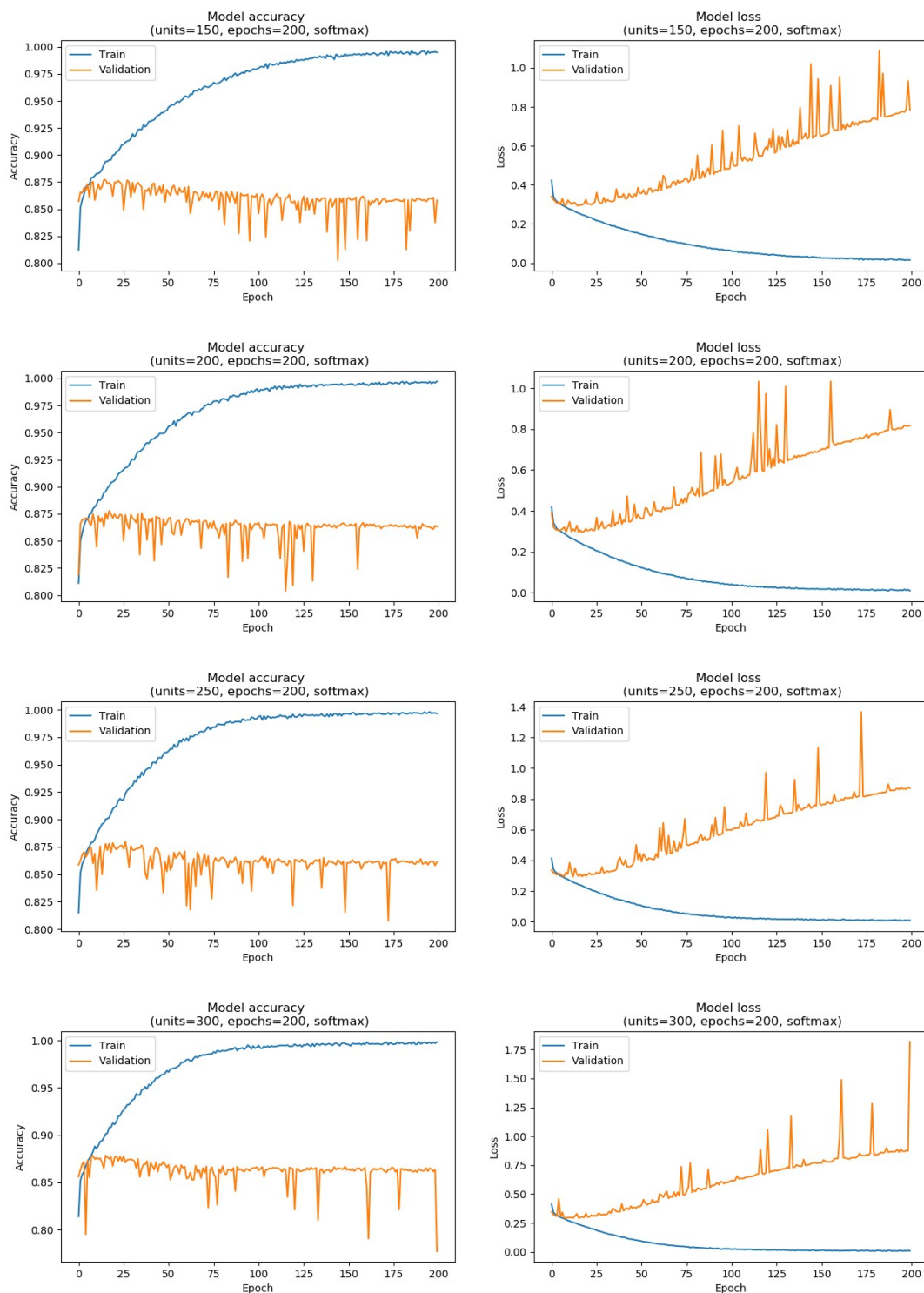
1) Variando o número de neurônios (units) na camada oculta, mantendo-se epochs = 200 e “softmax” na camada de saída.

Para avaliar o efeito da quantidade de neurônios na primeira camada escondida sobre a acurácia do classificador variou-se o argumento ‘units’ (1 [Perceptron], 10, 50, 100, 150, 200, 250, 300) na primeira camada escondida, mantendo-se constante o número de épocas (epochs = 200) e a função de ativação “softmax” na camada de saída. Nesta variação trabalhou-se com apenas uma camada escondida. Em relação as oito variações é fácil notar que usar apenas um neurônio com uma única camada escondida foi a que produziu o modelo com melhor acurácia, e sem sobreajuste. O aumento da quantidade de neurônios na camada escondida gerou modelos com overfitting, com boa performance no conjunto de treinamento, porém com inadequada capacidade de generalização. Da experimentação com número de épocas e camadas escondidas, constatou-se que submeter um modelo de rede a um excessivo número de iterações e também camadas escondidas pode levar a construção de modelos altamente especialistas (com overfitting). Portanto, as redes com arquiteturas mais complexas tendem a entrar mais facilmente em overfitting. Enfim, é razoável considerar que uma estrutura de rede com 1 neurônio (Perceptron) em uma única camada escondida, e com poucas épocas de treinamento (até 20) seja suficiente para se obter um bom classificador baseado em redes para o problema proposto.

Desempenho no conjunto de teste		
units	Test accuracy	Test loss
1	0.8674	0.3224
10	0.8564	0.3474
50	0.8446	0.6037
100	0.8442	0.7994
150	0.8482	0.8825
200	0.8515	0.9172
250	0.8516	0.9766
300	0.7704	1.9409

1) Variando o número de neurônios (units) na camada oculta, mantendo-se epochs = 200 e “softmax” na camada de saída.



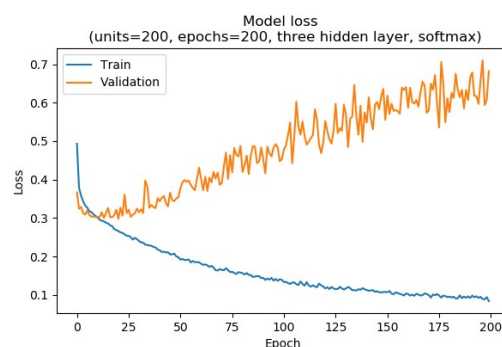
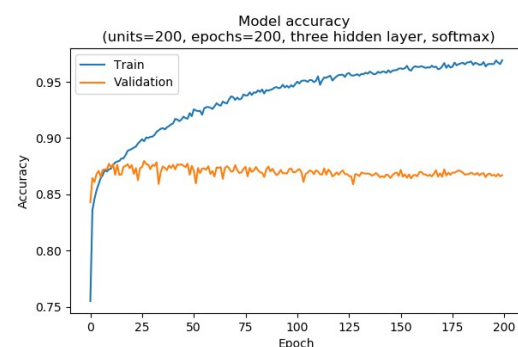
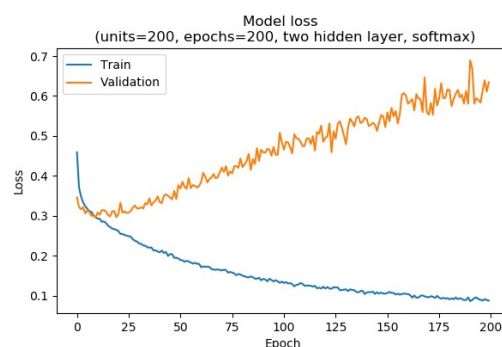
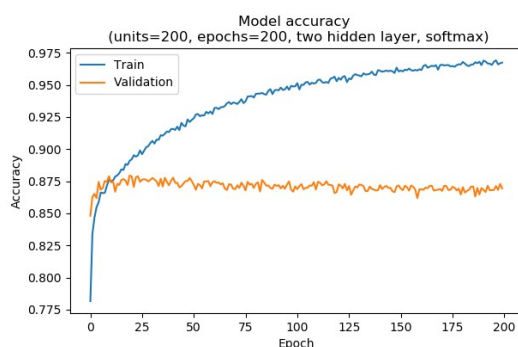
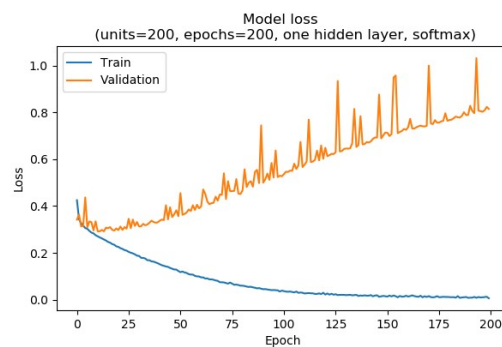
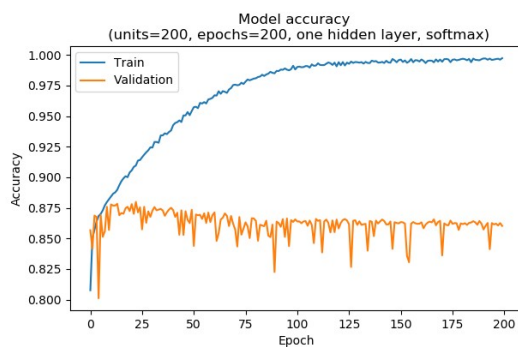


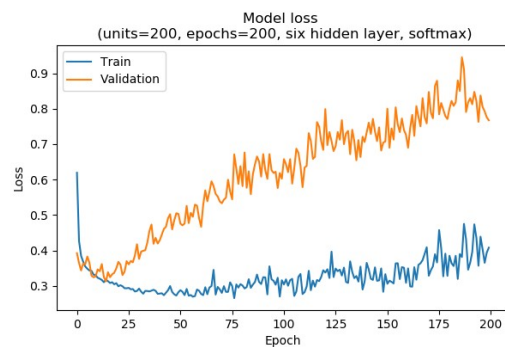
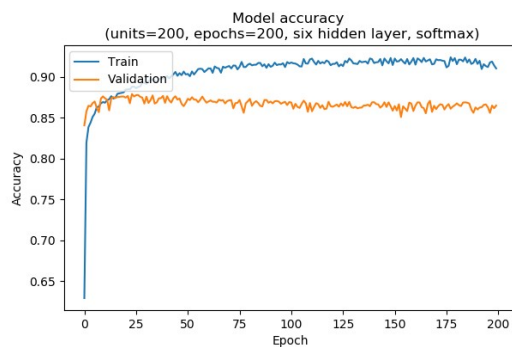
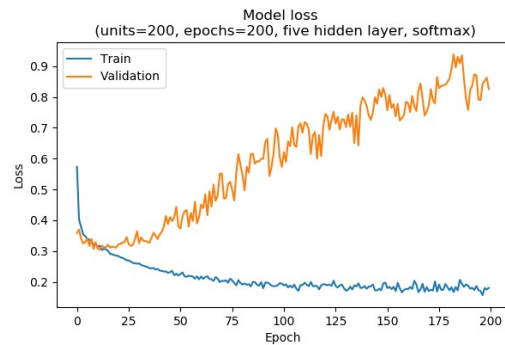
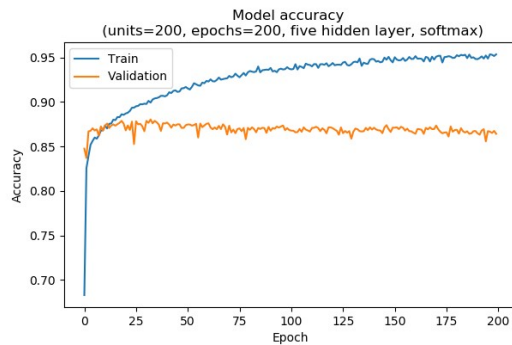
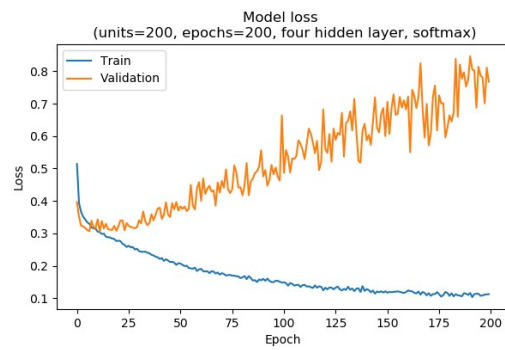
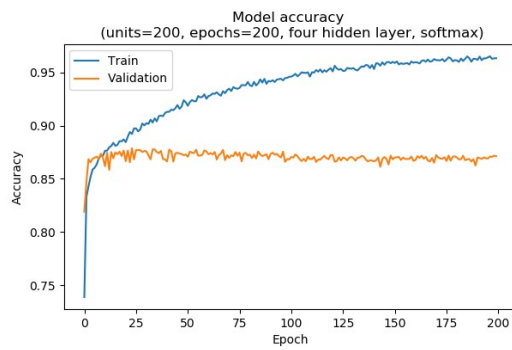
2) Variando a quantidade de camadas escondidas, mantendo-se epochs = 200 e “softmax” na camada de saída.

Para avaliar o efeito da quantidade de camadas escondidas sobre a acurácia do classificador adicionou-se gradativamente com o **método add()** novas camadas escondidas à estrutura da rede neural. A rede mais complexa foi constituída de seis camadas escondidas, todas com 200 neurônios (units), função de ativação “relu” e epochs = 200. Para todas as redes a função de ativação na camada de saída foi a “softmax”. Para avaliar o efeito do número de iterações variou-se o argumento “**epochs**” na função **fit()**, mantendo-se uma camada escondida, units = 100, função de ativação na camada de saída foi a softmax. De modo similar, o aumento do número de camadas

escondidas também tenderam a construir um modelo sobreajustado, com aprendizado de ruídos indesejáveis e com inadequada capacidade de generalização sobre novos dados. De certo, quanto maior o número de camadas escondidas na RN, maior foi sua capacidade de resolução do problema proposto. As curvas de aprendizado e validação mostraram tendência de sobreposição à medida que o número de camadas escondidas foi aumentado.

Desempenho no conjunto de teste		
Number of hidden layer	Test accuracy	Test loss
1	0.8525	0.9278
2	0.8599	0.7060
3	0.8594	0.7764
4	0.8576	0.8769
5	0.8549	0.9550
6	0.8590	0.8408





3) Variando o número de épocas de treinamento, mantendo-se uma camada escondida, units = 100, softmax

Para avaliar o efeito do número de épocas de treinamento sobre a acurácia do classificador variou-se na função **fit()** o argumento “**epochs**” (10, 25, 50, 75, 100, 150, 300, 600, 1000), mantendo-se uma camada escondida, units = 100, e função softmax. Da experimentação, constatou-se que submeter um modelo de rede a um excessivo número de iterações pode levar a construção de modelos altamente especialistas (com overfitting).

Desempenho no conjunto de teste		
Number of epochs	Test accuracy	Test loss
10	0.8596	0.3256
25	0.8564	0.3509
50	0.8627	0.3637
75	0.8554	0.4189
100	0.8474	0.5080
150	0.8491	0.6289
300	0.8453	1.0245
600	0.8448	1.3274
1000	0.8418	1.5900

