Classificação e regressão com redes MLP

Leonardo Loureiro Costa
leonardo.costa@unifesp.br

***Resumo*—Redes neurais como a MLP — MultiLayer Perceptron — são vastamente utilizadas para classificação e regressão. Neste relatório exploraremos o uso desse modelo neural em dois problemas: reconhecimento e classificação de dígitos em uma imagem, e a previsão da nota final de um candidato do ENEM com base em seu cartão de respostas.**

Mostraremos que é possível adquirir uma acurácia superior a 98% em problemas de classificação como o do dataset MNIST, e um desvio padrão de apenas 3 pontos na predição da nota do ENEM com base nas respostas assinaladas.

I. Introdução

A. Classificação

Neste relatório será apresentado um modelo neural treinado para classificar dígitos em uma imagem, numerais de 0 a 9 — o dataset MNIST.

Cada elemento do dataset em questão consiste em uma imagem monocromática, de dimensões 28 por 28 píxeis, representando um algarismo em branco sobre um fundo preto, como ilustrado na figura 1.

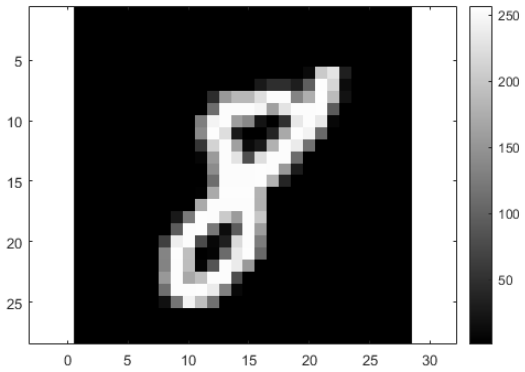

Figura 1

Uma rede MLP, implementada em Python por meio da biblioteca TensorFlow, será treinada para reconhecer os padrões que formam um dígito e o classificar de 0 a 9.

B. Regressão

O ENEM, Exame Nacional do Ensino Médio, aplicado pelo INEP, no Brasil, não atribui as notas finais para cada uma das 4 áreas de conhecimento proporcionalmente ao número de questões acertadas.

A prova utiliza de uma ferramenta chamada TRI, que pondera as questões ao nível de dificuldade e penaliza o candidato que acerta as difíceis, mas erra as fáceis: sinal de um possível chute.

O modelo em questão tem como meta treinar em cima da base de dados fornecida pelo INEP e, considerando as questões assinaladas e a nota final, prever o resultado dado um determinado gabarito.

1) Dados utilizados: Os dados utilizados para o treinamento da rede foram coletados da seção de "Microdados"do site do INEP, um dataset que contém informações socioeconômicas e de desempenho de cada um dos candidatos.

Foram considerados para este relatório somente os dados do caderno azul de matemática dos alunos que não tiveram seu gabarito anulado: $533,925$ participantes.

II. Objetivos

1) Treinar modelos MLP para problemas de classificação
2) Treinar modelos MLP para problemas de regressão
3) Avaliar o impacto de diferentes topologias, variando número de camadas e de neurônios.
4) Avaliar o impacto do uso do Momentum
5) Avaliar o impacto do uso da regularização L2

III. Resultados — Classificação

A. Diferentes topologias

Todos os modelos foram treinados usando um "batch size"de valor 25.

1) Modelo 1: Este modelo foi, primeiramente, treinado por 10 épocas. O modelo aprendeu bem, porém muito rapidamente atingiu um plateau, estagnando em uma acurácia de validação de $\approx 90\%$.

Camada	Nº Perceptrons	Ativação
1ª camada	10	SoftMax

Tabela I

2) Modelo 2: Este modelo treinou, primeiramente, por 10 épocas. Como apresentou uma curva de acurácia com uma derivada maior do que o modelo anterior, ele foi treinado novamente em 100 épocas. Obteve uma acurácia de validação de aproximadamente 96.9%.

Camada	Nº Perceptrons	Ativação
1ª camada	100	ReLu
2ª camada	10	SoftMax

Tabela II

3) Modelo 3: A fim de melhorar a acurácia do modelo 2, a complexidade da rede foi aumentada, como ilustra a tabela III. A acurácia de validação obtida foi maior que a anterior, aproximadamente 97.47%.

Camada	Nº Perceptrons	Ativação
1ª camada	512	ReLu
2ª camada	10	SoftMax

Tabela III

4) Modelo 4: Com a meta de aumentar novamente a acurácia do modelo, a topologia cresceu em mais uma camada intermediária de 512 neurônios, como ilustra a tabela IV, apresentando a menor taxa de erros até então, uma acurácia de aproximadamente 97.6%

Camada	Nº Perceptrons	Ativação
1ª camada	512	ReLu
2ª camada	512	ReLu
3ª camada	10	SoftMax

Tabela IV

5) Modelo 5: O modelo 5 mantém a mesma topologia de camadas do anterior, mas dobra o número de neurônios em cada uma delas. O modelo obteve a mesma acurácia do anterior.

Camada	Nº Perceptrons	Ativação
1ª camada	1024	ReLu
2ª camada	1024	ReLu
3ª camada	10	SoftMax

Tabela V

B. Regularização e uso de Momentum

1) Modelo 6: O modelo 6 da rede introduz uma regularização dos dados de valor 0.01, baseando-se na rede anterior. A regularização aumentou a acurácia final de validação para 98%.

Camada	Nº Perceptrons	Ativação
1ª camada	1024	ReLu
2ª camada	1024	ReLu e Regularização L2
3ª camada	10	SoftMax

Tabela VI

2) Modelo 7: Este modelo é idêntico em topologia ao modelo anterior, porém apresenta a adição do parâmetro "momentum", de valor 0.9. Esta adição acelera a taxa de aprendizagem do modelo, o fazendo convergir em menor tempo. Este modelo atingiu uma acurácia de validação superior a 97% em apenas 3 épocas de treinamento, cerca de 6 vezes mais rápido que o modelo anterior, que só alcançou tal métrica em 18 épocas.

A acurácia obtida é a melhor até então, 98.15%.

C. Análise gráfica

A figura 2 ilustra a acurácia no conjunto de validação dos modelos 2 ao 7 em função do tempo. A figura 3 é um recorte do gráfico 1 com um filtro de média móvel aplicado, nele observa-se melhor quais gráficos obtiveram melhor desempenho. Nota-se que os modelos 4 e 5 obtiveram desempenho quase idêntico.

O modelo 1 não está incluso, pois treinou apenas por 10 épocas.

Observa-se que o modelo 7 é o que melhor performa na tarefa de classificar os dígitos do dataset utilizado.

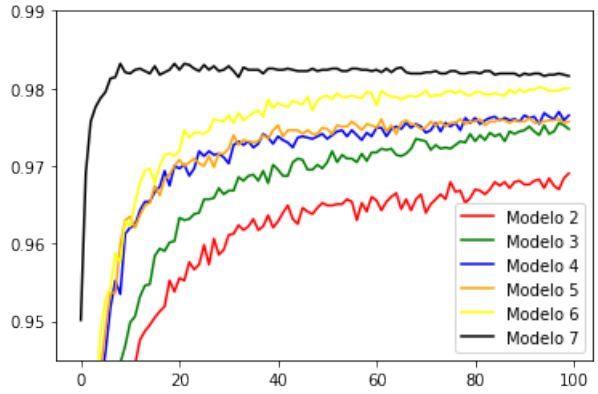

Figura 2: Acurácia no conjunto de treinamento em função do tempo para os modelos de 2 a 7.

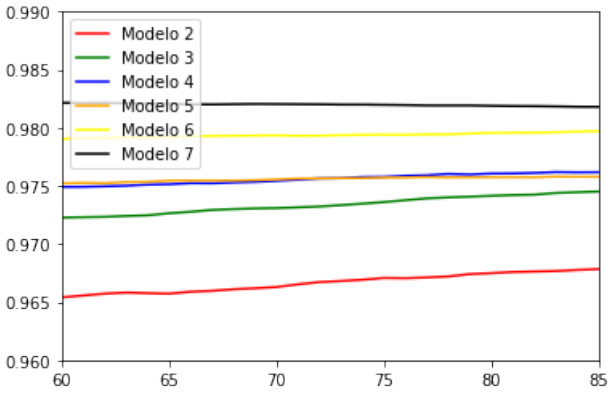

Figura 3: Gráfico da figura II, porém ampliado e com um filtro de média móvel aplicado.

Na figura 4 é mais claro notar o desempenho da rede na tarefa de classificar os algarismos. Nas temos os valores verdadeiros, reais, das imagens analisadas e nas colunas as predições do modelo neural.

Observa-se que os poucos erros existentes se dão em algarismos com formatos semelhantes: confunde-se mais um "3"com um "5"do que com um "1", visto que eles possuem uma maior semelhança visual. Esse efeito se evidencia quando comparamos a figura 4 com a figura 5, a segunda possui mais erros que a primeira, visto que é resultado da rede que apresentou pior desempenho durante o treino.

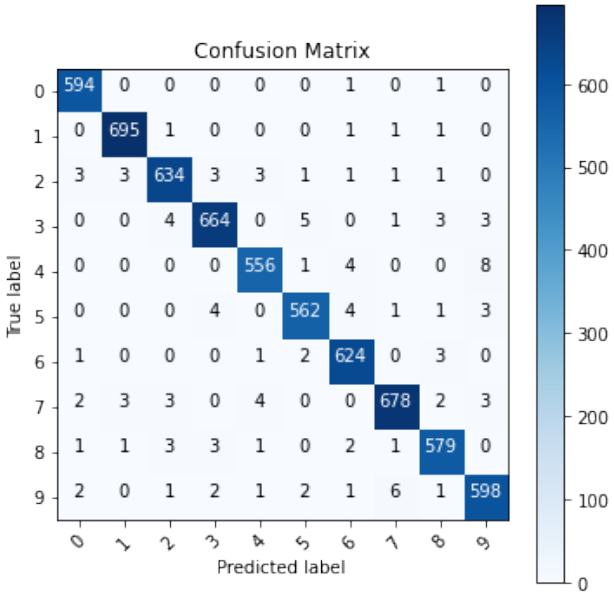

Figura 4: Matriz de confusão da predição do conjunto de testes da rede 7

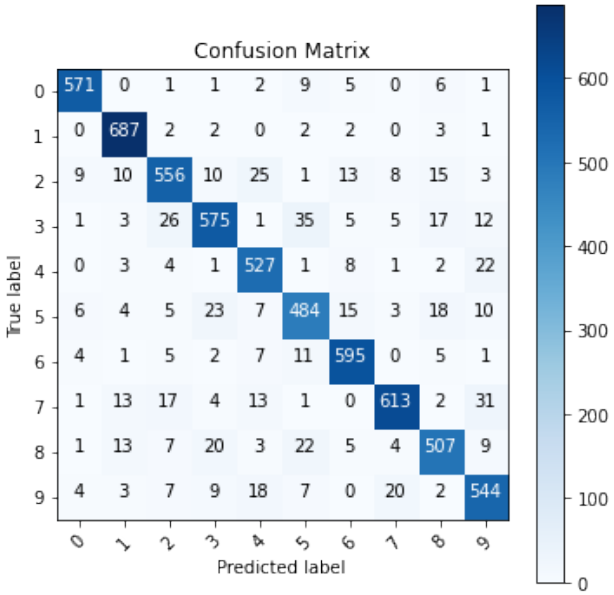

Figura 5: Matriz de confusão da predição do conjunto de testes da rede 1

IV. Resultados — Regressão

Os modelos todos foram treinados na base de dados apresentada na introdução: um dataset com $533,925$ instâncias contendo as questões assinaladas e o gabarito oficial para essas.

Um vetor com 45 entradas, uma para cada questão, contendo "1"para certa e "0"para errada, ali-

mentou um modelo criado em Python através da interface Keras do TensorFlow.

A. *Diferentes topologias*

Os modelos diferentes modelos treinados a seguir obtiveram erros médios absolutos nos conjuntos de validação como ilustra a tabela tal.

Modelo	Erro médio absoluto
1	6.35
2	5.79
3	3.73
4	0.97
5	0.77
6	0.91
7	0.75

Tabela VII: Erro médio absoluto de cada modelo.

1) Modelo 1: O modelo 1 treinou por 15 épocas, mas atingiu um plateau e não conseguiu reduzir o erro médio absoluto para a abaixo de 5.

Camada	Nº Perceptrons	Ativação
1ª camada	32	ReLu
2ª camada	1	Relu

Tabela VIII

2) Modelo 2: De modo a reduzir esse erro, a complexidade do modelo foi aumentada, quadruplicando o número de neurônios na primeira camada. O resultado foi muito similar ao do modelo 1.

Camada	Nº Perceptrons	Ativação
1ª camada	128	ReLu
2ª camada	1	ReLu

Tabela IX

3) Modelo 3: Para este modelo a complexidade foi aumentada novamente: 512 neurônios na primeira camada. O erro obtido foi o menor até então.

Camada	Nº Perceptrons	Ativação
1ª camada	512	ReLu
2ª camada	1	ReLu

Tabela X

4) Modelo 4: Dobrando o número de neurônios do modelo 3 obtemos um resultado muito similar.

Camada	Nº Perceptrons	Ativação
1ª camada	1024	ReLu
2ª camada	1	ReLu

Tabela XI

5) Modelo 5: Optando pela menor complexidade que oferece o melhor desempenho, a topologia do modelo 3 foi testada com uma camada a mais, como ilustra a tabela XII.

Camada	Nº Perceptrons	Ativação
1ª camada	512	ReLu
2ª camada	512	ReLu
3ª camada	1	ReLu

Tabela XII

6) Modelo 6: O modelo 6 avalia o desempenho do modelo 5 quando o parâmetro de momentum tem o seu valor alterado de 0 para 0.9.

Camada	Nº Perceptrons	Ativação
1ª camada	512	ReLu
2ª camada	512	ReLu
3ª camada	1	ReLu

Tabela XIII

7) Modelo 7: O modelo 7, analogamente, avalia o desempenho do modelo 6 quando a regularização do tipo L2 com valor de 0.01 é adicionada à rede.

Camada	Nº Perceptrons	Ativação
1ª camada	512	ReLu
2ª camada	512	ReLu e Regularização L2
3ª camada	1	ReLu

Tabela XIV

B. *Análise gráfica*

Comparando o modelo 1, o que obteve pior desempenho no conjunto de validação com o modelo 7, o que melhor performou, é possível analisar o quão eficiente é cada modelo na tarefa de prever a nota final de cada aluno com base em seu cartão de respostas.

A figura 6 ilustra dois histogramas: em roxo o modelo 7, e em vermelho o modelo 1. Cada histograma representa um vetor de diferenças entre os valores previstos pela rede e os valores reais das notas dos alunos avaliados.

Observando o gráfico, nota-se uma entropia menor no modelo 7, um pico perto do 0 — ele acerta muito mais do que o modelo 1.

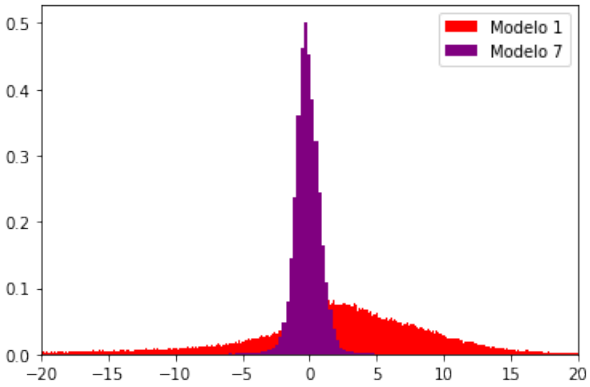

Figura 6: Histograma da diferença entre o resultado previsto pelo modelo e o resultado real obtido.

A figura 7 compara, em uma escala logarítmica, o erro médio absoluto de todas as redes treinadas — conjunto de validação. Observa-se que os modelos 6 e 7 oscilam com maior amplitude do que os demais modelos, isso se dá, pois esses detém de um valor maior de "momentum", 0.9, do que os demais, 0.

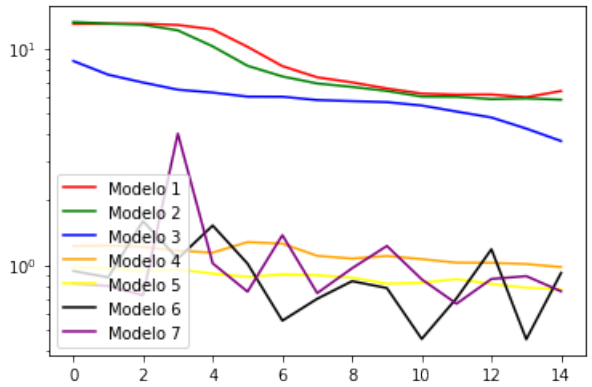

Figura 7: Erro médio absoluto dos resultados previsto pelos modelos de 1 a 7 em escala logarítmica em função da quantidade de épocas de treinamento.

A figura 8 ilustra, também em uma escala logarítmica, os ultimos 4 modelos: 4, 5, 6, e 7. As curvas passaram por um filtro de média móvel com janela 7. Observando-os é possível notar a similaridade da eficiência dos modelos. Nota-se que, segundo o gráfico, o modelo 6 possui a menor quantidade de erros, contrário ao que a tabela VII aponta.

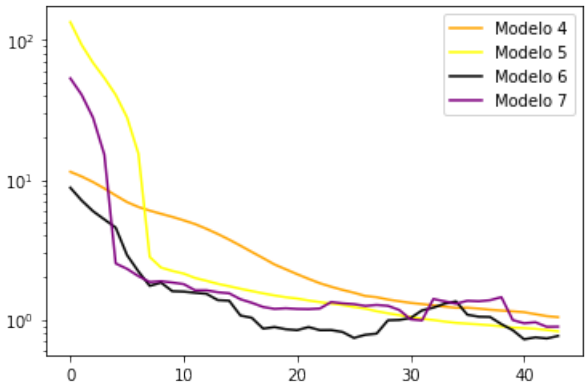

Figura 8: Erro médio absoluto dos resultados previstos pelos modelos de 4 a 7 em escala logarítmica em função da quantidade de épocas de treinamento com um filtro de média móvel de janela 7 aplicado.

A figura 9 ilustra o Erro médio absoluto de cada modelo, mas dessa vez no conjunto de teste. Nota-se que, de todos, o que possui menor valor é o modelo 5.

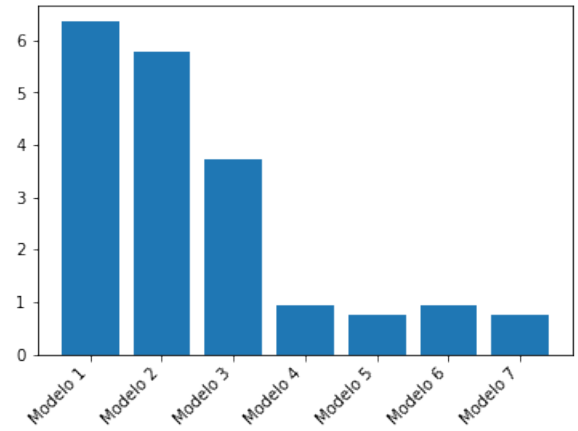

Figura 9: Erro médio absoluto de cada modelo no conjunto de testes.

V. Conclusão

O modelo treinado para classificação obteve bons resultados, classificando os dígitos corretamente cerca de 98% das vezes no conjunto de testes.

Analogamente, o modelo treinado para regressão também obteve resultados satisfatórios, prevendo as notas com base nos cartões de resposta com um desvio padrão de ≈ 3.0. Dessa forma, é possível com esse modelo prever com 99.7% de certeza que a nota de um candidato será igual à nota real ± 9.02 pontos.

A. Topologia

Em ambos os problemas, classificação e regressão, a topologia do modelo se mostrou, em quesitos de importância, crucial para o bom desenvolvimento do modelo. Uma rede com uma topologia pequena demais não consegue aprender padrões o suficiente para corretamente produzir a saída desejada.

Fica claro como aumentar o número de neurônios por camada, ou aumentar o número de camadas no modelo, influencia diretamente na capacidade de aprendizagem do modelo.

B. Hiperparâmetros

A alteração de hiperparâmetros, analogamente à da topologia, influencia, também, no resultado do modelo. Modelos de regressão, como o apresentado nesse relatório não possuem, por definição, um limite no valor final apresentado. Desta forma é possível que ocorra o fenômeno de explosão do gradiente, onde o resultado cresce de tal forma que o modelo fica incapacitado de continuar aprendendo.

A alteração de hiperparâmetros como a redução de η durante o treinamento evitam esse acontecimento, ilustrando, dessa forma, como por meio da alteração desses valores é possível melhor moldar o aprendizado da rede.

C. Treino, Validação, Teste

Os valores de acurácia, ou erro, obtidos nos conjuntos de treino, validação, e teste são vastamente diferentes. Por norma, idealmente o conjunto de treino possui acurácia de $\approx 100\%$, para regressão, erro de ≈ 0. A meta do treino é aproximar os valores do conjunto de validação dos valores ideais do conjunto de treino.

Porém, nem sempre isso ocorre da mesma forma para o conjunto de teste. O modelo treinado para regressão teve, como mais eficiente, a rede 7 no conjunto de validação, rede 5 no conjunto de testes, e a rede 6 quando considerada a curva de erros com um filtro de média móvel aplicada.

Conclui-se, portanto, a respeito dos diferentes conjuntos usados do "dataset", sendo tênue a classificação do melhor modelo, essa escolha deve ser feita com base em análises mais profundas, não considerando — apenas — os últimos valores produzidos pela rede em cada um de seus conjuntos.

REFERÊNCIAS

[1] Microdados enem, 2021.

[2] Digitos mnist, 2022.

[3] Simon S. Haykin. *Neural networks and learning machines*. Pearson Education, Upper Saddle River, NJ, third edition, 2009.