Comparação de Modelos de Representação Latente

José Geraldo Fernandes
Escola de Engenharia
Universidade Federal de Minas Gerais
Belo Horizonte, Brasil
josegeraldof@ufmg.br

I. Introdução

Aprendizado de representação é uma propriedade muito importante que justifica alguma parte do grande sucesso de modelos profundos [1]. Em uma rede neural a última camada é apenas um discriminador linear, cabe, portanto, às camadas anteriores aprender uma representação dos dados que modele o problema de classificação linearmente. Em redes convolucionais, há uma semelhança em o quão fundo está a camada e a complexidade de *features* que o mesmo captura.

Uma vantagem muito grande de trabalhar com espaços lineares é todo o arcabouço formal da teoria de regressão linear [2]. Não só o problema de aprendizado é mais simples como há diversas propriedades interessantes e úteis quando o problema está matematicamente bem formulado.

Outro aspecto interessante do aprendizado de representação é carregar informação *a priori* para direcionar a projeção. Em redes neurais convencionais, o interesse é no desempenho do classificador de forma direta, todavia outro direcionamento é plenamente possível como informação de vizinhança [3].

Esse trabalho explora métodos de aprendizado de representação não supervisionada e compara com as técnicas convencionais. Também propõe um método heurístico de seleção de amostras para treinamento.

II. Revisão Bibliográfica

A. Autoencoder

Autoencoders são apenas uma rede neural convencional replicando a entrada na saída, isto é, o objetivo da rede é apenas reconstruir os dados e encontrar uma representação latente [4]. Pode-se considerar uma rede dividida em dois módulos: *encoder* $\boldsymbol{H} = f(\boldsymbol{X})$, que faz a projeção latente; e, o *decoder* $\boldsymbol{X} = g(\boldsymbol{H})$ que tenta reconstruir a saída a partir dessa representação.

É especialmente útil para encontrar uma representação latente compacta de dados muito extensos ou complexos, como imagens e sons, de outra forma intratáveis. Há trabalhos que mostram alto ganho de desempenho em comparação com outras técnicas mais clássicas de redução de dimensionalidade [5].

Apesar disso, o sentido contrário também é possível, principalmente quando há uma direção *a priori* que guie o treinamento da projeção, como dispersibilidade [6] por exemplo.

De qualquer forma, espera-se que o novo espaço $\boldsymbol{H}$ traga propriedades que facilitem em algum aspecto o problema de aprendizado.

B. Grafo de Gabriel

O Grafo de Gabriel é uma construção a partir de um conjunto de pontos $\mathcal{S}$, que define os vértices do grafo, para outro conjunto de arestas $\mathcal{E}$ tal que dois pontos $\boldsymbol{x_i}, \boldsymbol{x_j}$ são adjacentes, definem uma aresta, se não há outro ponto de $\mathcal{S}$ dentro da hiperesfera definida com a distância entre os dois pontos $\boldsymbol{x_i}, \boldsymbol{x_j}$ como diâmetro, como na Equação 1.

$$\begin{gathered}(\boldsymbol{x_i}, \boldsymbol{x_j}) \in \boldsymbol{\mathcal{E}} \leftrightarrow \\ \delta^2(\boldsymbol{x_i}, \boldsymbol{x_j}) \leq [\delta^2(\boldsymbol{x_i}, \boldsymbol{x_k}) + \delta^2(\boldsymbol{x_j}, \boldsymbol{x_k})] \ \forall \boldsymbol{x_k} \in \mathcal{S}\end{gathered} \tag{1}$$

Onde δ é a métrica de distância na construção e comumente definida como a distância euclidiana.

A construção desse grafo é uma técnica da Geometria Computacional e emprestada para os problemas de aprendizado como uma forma de expressar as características de vizinhança da estrutura do conjunto de dados.

1) Métrica de Qualidade: Baseado na diferença de classes entre o vértice e as amostras conectadas por suas arestas, define-se uma grandeza Q que representa a qualidade dessa amostra como na Equação 2, onde V representa o número de arestas e V_{eq} o número dessas para amostras de classe coincidente.

$$Q(\boldsymbol{x}_i) = \frac{V_{eq}(\boldsymbol{x}_i)}{V(\boldsymbol{x}_i)} \tag{2}$$

Amostras na região de fronteira e sobreposição terão sua qualidade afetada quão maior for a mistura. Também, amostras com qualidade perfeita $Q(\boldsymbol{x}_i) = 1$ estão em regiões predominantes e longe da fronteira.

A partir do princípio que as amostras mais importantes para o treinamento são as localizadas na região de fronteira, uma estratégia de seleção é filtrar amostras com essa métrica de qualidade alta.

III. Metodologia

A. Base de Dados

Para avaliação dos índices de qualidade dos agrupamentos seleciona-se um conjunto de 15 *datasets* padrão do repositório UCI [7] em todos os testes. Esses são separados em 10 partições determinadas para validação cruzada *k-fold*.

Para confiabilidade do resultado, as bases de dados selecionadas são amplamente utilizadas em aplicações semelhantes. Como pré-processamento, fez-se uma normalização de todos

os atributos e conversão dos categóricos em númericos, necessário para o algoritmo CLAS. Todos são de problemas de classificação binária. Segue a descrição da seleção: *Statlog Australian Credit Approval* (australian); *Banknote Authentication* (banknote); *Breast Cancer Wisconsin* (breastcancer); *Breast Cancer Hess Probes* (breastHess); *Liver Disorders* (bupa); *Climate Model Simulation Crashes* (climate); *Pima Indian Diabetes* (diabetes); *Fertility* (fertility); *Statlog German Credit Data* (german); *Gene Expression* (golub); *Haberman's Survival* (haberman); *Statlog Heart Dicease* (heart); *Indian Liver Patient* (ILPD); *Parkinsons* (parkinsons); *Connectionist Bench Sonar, Mines vs. Rocks* (sonar).

A Tabela I mostra as principais características dessa seleção, η representa a proporção entre as classes. Note a alta diversidade dos problemas para atestar a generalização do método.

Tabela I
CARACTERÍSTICAS DAS BASES DE DADOS SELECIONADAS.

Dataset	**Amostras**	**Atributos**	η
australian	690	14	0.44
banknote	1372	4	0.44
breastcancer	683	6	0.65
breastHess	133	30	0.74
bupa	345	6	0.42
climate	540	18	0.91
diabetes	768	8	0.65
fertility	100	9	0.12
german	1000	24	0.70
golub	72	50	0.65
haberman	306	3	0.74
heart	270	13	0.56
ILPD	579	10	0.72
parkinsons	195	22	0.75
sonar	208	60	0.47

B. Arquitetura

Neste trabalho comparou-se duas arquiteturas parecidas, *multi-layer perceptron* (MLP), e *autoencoder* (AE), ambos com uma camada escondida e mesmo tamanho, comumente maior que a dimensão dos dados. Na primeira abordagem, um espaço é calculado a partir da métrica de custo supervisionado. No segundo, no entanto, o espaço é aprendido sem informação rotulada e a classificação é feita por um regressor logístico no espaço latente.

A esperança dessa projeção é representar os dados em um espaço $\boldsymbol{H}$ mais tratável onde o problema é linearmente separável. Dessa forma, conclui-se o treinamento com uma regressão logística padrão como na Equação 3, onde λ é um parâmetro de regularização L2.

$$\boldsymbol{W} = (\boldsymbol{H}^T\boldsymbol{H} + \lambda \boldsymbol{I}_p)^{-1}\boldsymbol{H}^T\boldsymbol{y} \tag{3}$$

A síntese do modelo é direta, faz-se a projeção de novos dados a partir da rede aprendida, e calcula-se a regressão logística com os parâmetros $\boldsymbol{W}$, como na Equação 4.

$$\hat{\boldsymbol{y}} = S(\boldsymbol{H}\boldsymbol{W}) \tag{4}$$

Uma outra modificação é no sentido de direcionar a projeção a partir da função de custo. Enquanto um AE tradicional tenta reconstruir a entrada com um código latente, pode-se modificar a função de custo no sentido de forçar essa representação a uma propriedade de interesse.

Uma métrica conveniente para tentar ortogonalizar, o interesse final é linearizar, o espaço é o *crosstalk*, calculado no espaço latente como na Equação 5.

$$C(\boldsymbol{H}) = \sum_i \sum_j \boldsymbol{h}_i \boldsymbol{h}_j \tag{5}$$

C. Seleção de Parâmetros

Nesse tipo de modelo, combinação de projeção e regressão logística, há apenas dois principais parâmetros de ajuste: o tamanho da projeção p; e, o peso da regularização λ.

Uma das formas de fazer o ajuste é a partir de validação cruzada com conjuntos separados de teste. Contudo, esse método pode ser custoso computacionalmente além de reduzir o tamanho efetivo de dados no treinamento.

Nesse sentido, e em virtude de aproveitar a teoria fundamentada de regressão logística, utiliza-se uma equação fechada para o erro de validação cruzada *leave-one-out* (LOO) do regressor nos dados da projeção, como na Equação 6.

$$\begin{aligned} \boldsymbol{A} &= (\boldsymbol{H}^T\boldsymbol{H} + \lambda \boldsymbol{I}_p)^{-1} \\ \boldsymbol{P} &= \boldsymbol{I}_N - \boldsymbol{H}\boldsymbol{A}\boldsymbol{H}^T \\ \boldsymbol{\sigma}_i &= \operatorname{diag}(\boldsymbol{P})_i^{-1}(\boldsymbol{P}\boldsymbol{y})_i \\ \text{LOO} &= N^{-1}\boldsymbol{\sigma}^T\boldsymbol{\sigma} \end{aligned} \tag{6}$$

Espera-se haver uma correspondência entre otimizar essa função objetivo LOO, menos custosa, e a generalização, desempenho em novos dados.

A otimização multiobjetivo foi feita de forma simples e em duas etapas. Primeiro encontrou-se p ótimo em varredura com $\lambda = 0$, e, em seguida, variou-se o parâmetro de regularização também em varredura.

D. Classificação

O problema de classificação segue o padronizado. Para cada *dataset* separou-se 10 *folds* fixos, no conjunto de treinamento treinou-se a rede de projeção e o regressor com os parâmetros p, λ ótimos, no conjunto de teste avaliou-se o desempenho.

E. Amostragem

Para tratar o problema de classes desbalanceadas, há alguns exemplos graves como exposto na Tabela I, utilizou-se uma heurística de amostragem.

Como método proposto, utilizou-se o filtro de seleção a partir do Grafo de Gabriel retirando amostras com qualidade total $Q(\boldsymbol{x}_i) = 1$, *downsample*. Essas amostras estão longe da região de fronteira então é razoável assumir que têm pouca importância no problema de aprendizado.

IV. Resultados

A. Seleção de Parâmetros

Para assegurar a esperança de correspondência entre a função objetivo LOO e o desempenho em um conjunto de teste, avaliou-se simultaneamente ambos atributos. Na validação cruzada, a métrica foi o erro quadrático médio (MSE).

O comportamento observado é variado em função do *dataset*, as Figuras 1 e 2 mostram um experimento de varredura em p onde as curvas parecem próximas. Contudo, há exemplos, como nas Figuras 3 e 4, onde, apesar de alguns traços serem bem convergentes, a curva LOO diverge quando MSE parece ainda decrescer.

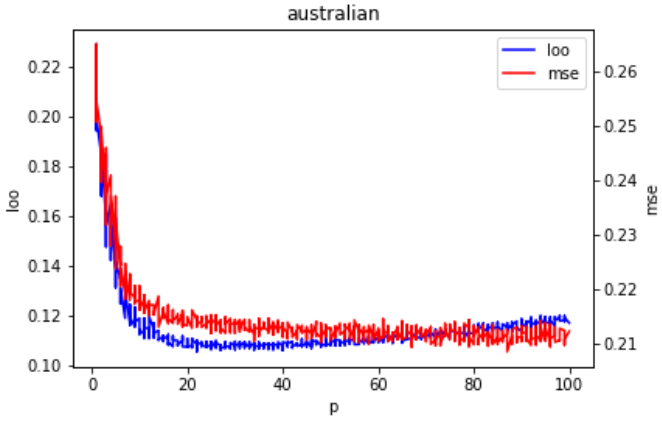

Figura 1. Curvas do custo LOO, no conjunto de treinamento, e MSE, no conjunto de teste; *dataset* australian e avaliando p na ELM.

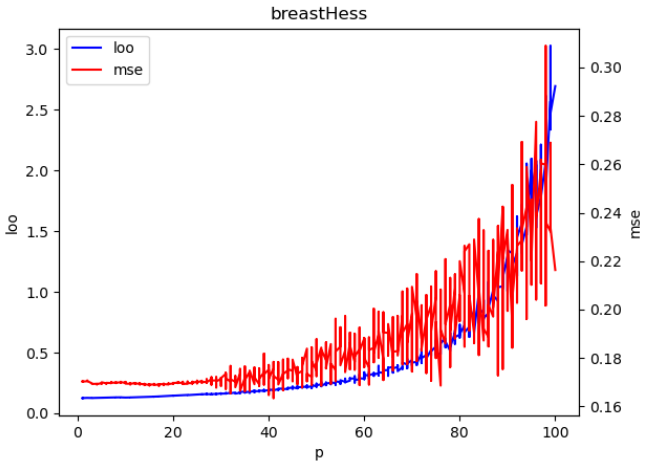

Figura 2. Curvas do custo LOO, no conjunto de treinamento, e MSE, no conjunto de teste; *dataset* breastHess e avaliando p no AE.

Na varredura de λ esse fenômeno também acontece. As Figuras 5 e 6 mostram curvas similares das funções de custo, enquanto as Figuras 7 e 8 mostram um comportamento

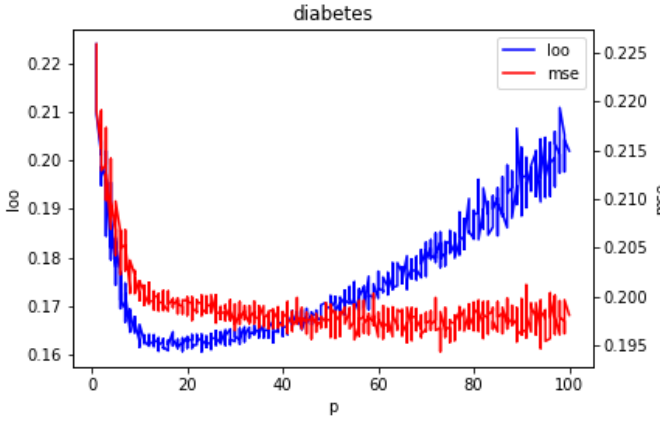

Figura 3. Curvas do custo LOO, no conjunto de treinamento, e MSE, no conjunto de teste; *dataset* diabetes e avaliando p na ELM.

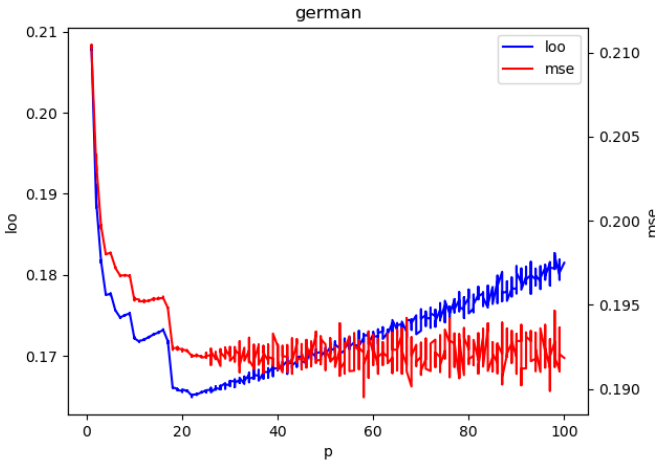

Figura 4. Curvas do custo LOO, no conjunto de treinamento, e MSE, no conjunto de teste; *dataset* german e avaliando p no AE.

divergente ainda mais agressivo, apesar da escala de variação ser menor.

Os parâmetros foram selecionados por inspeção visual do mínimo da curva LOO e são expostos na Tabela II.

B. Classificação

A Tabela III mostra os resultados obtidos com a área sob a curva ROC (AUC) média calculado no conjunto de teste. As duas redes neurais foram testadas, MLP, AE e AE com função de custo modificada (AE-C), além de um regressor logístico (RG) nos dados sem a projeção como comparação.

C. Amostragem

A Tabela IV mostra o número médio de amostras e de proporção entre as classes η no conjunto de treinamento após o *downsample*. Enquanto a tabela V mostra os resultados obtidos com essa heurística.

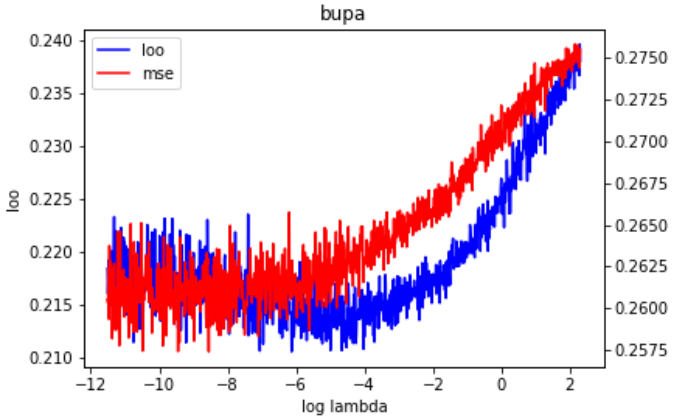

Figura 5. Curvas do custo LOO, no conjunto de treinamento, e MSE, no conjunto de teste; *dataset* bupa e avaliando λ na ELM.

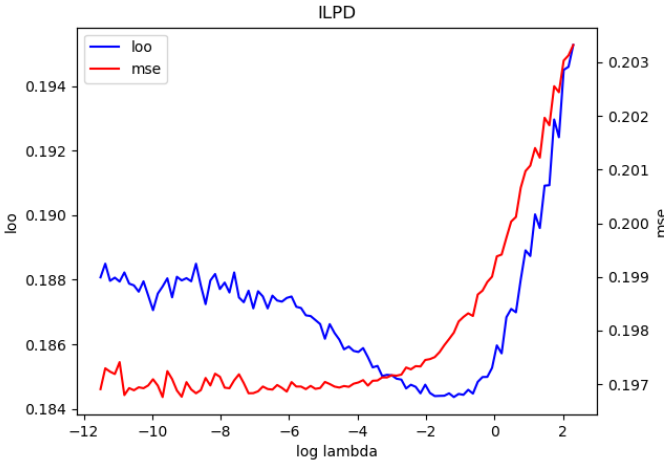

Figura 8. Curvas do custo LOO, no conjunto de treinamento, e MSE, no conjunto de teste; *dataset* ILPD e avaliando p no AE.

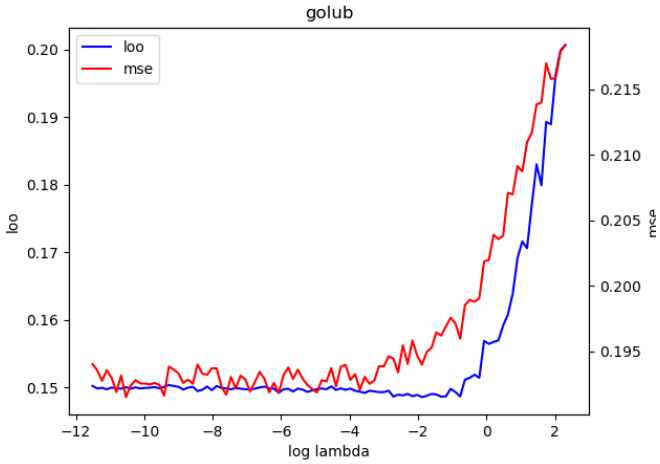

Figura 6. Curvas do custo LOO, no conjunto de treinamento, e MSE, no conjunto de teste; *dataset* golub e avaliando p no AE.

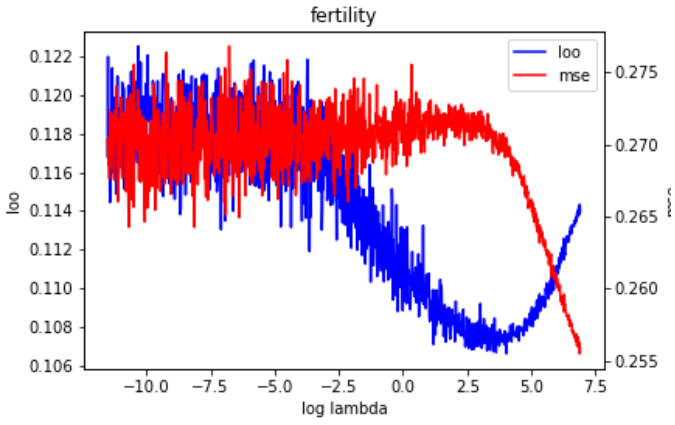

Figura 7. Curvas do custo LOO, no conjunto de treinamento, e MSE, no conjunto de teste; *dataset* fertility e avaliando p na ELM.

Tabela II
PARÂMETROS SELECIONADOS.

Dataset	p	$\log \lambda$
australian	20	-12
banknote	100	-12
breastcancer	40	-8
breastHess	5	-1
bupa	5	-1
climate	20	-1
diabetes	10	-1
fertility	5	2
german	20	-1
golub	5	-2
haberman	5	-2
heart	10	-1
ILPD	10	-1
parkinsons	10	-1
sonar	10	-1

Tabela III
RESULTADOS DE DESEMPENHO.

Dataset	**MLP**	**AE**	**AE-C**	**RG**
australian	0.93	0.92	0.93	0.86
banknote	1.00	1.00	1.00	0.97
breastcancer	0.99	1.00	1.00	0.97
breastHess	0.87	0.89	0.88	0.76
bupa	0.70	0.69	0.70	0.58
climate	0.96	0.91	0.96	0.77
diabetes	0.83	0.83	0.83	0.72
fertility	0.67	0.67	0.66	0.50
german	0.79	0.76	0.80	0.69
golub	0.71	0.69	0.60	0.77
haberman	0.69	0.70	0.72	0.50
heart	0.91	0.90	0.91	0.83
ILPD	0.75	0.72	0.72	0.52
parkinsons	0.89	0.87	0.88	0.76
sonar	0.88	0.84	0.81	0.79

Tabela IV
CARACTERÍSTICAS DO CONJUNTO DE TREINAMENTO NA TÉCNICA DE *downsampling*.

Dataset	**Amostras**	η
australian	427.8	0.61
banknote	586	0.94
breastcancer	243.1	0.12
breastHess	76.6	0.62
bupa	299.6	0.42
climate	476.6	0.91
diabetes	660.7	0.67
fertility	40.5	0.09
german	795.1	0.67
golub	52.6	0.58
haberman	169.2	0.71
heart	206	0.62
ILPD	362.9	0.65
parkinsons	83.9	0.49
sonar	178.8	0.44

Tabela V
RESULTADOS DE DESEMPENHO COM A HEURÍSTICA.

Dataset	**MLP**	**AE**	**AE-C**	**RG**
australian	0.93	0.92	0.93	0.86
banknote	1.00	1.00	1.00	0.97
breastcancer	0.99	1.00	1.00	0.97
breastHess	0.87	0.89	0.88	0.76
bupa	0.70	0.69	0.70	0.58
climate	0.96	0.91	0.95	0.77
diabetes	0.83	0.83	0.83	0.72
fertility	0.67	0.67	0.66	0.50
german	0.79	0.76	0.80	0.69
golub	0.71	0.69	0.63	0.77
haberman	0.69	0.70	0.71	0.50
heart	0.91	0.90	0.91	0.83
ILPD	0.75	0.72	0.72	0.52
parkinsons	0.89	0.87	0.88	0.76
sonar	0.88	0.84	0.81	0.79

V. DISCUSSÕES

Os resultados de desempenho mostram que mesmo uma representaçao não supervisionada foi suficientemente tão boa no problema de aprendizado quanto a convencional, MLP.

Sobre a modificação na função de custo, não houve uma diferença tão grande no desempenho destoante o maior custo computacional. Além disso, a função de custo combinada ainda representa um desafio maior de otimização multi-objetivo não trivial, mostrando problemas como balanceamento de componentes ou formas de combinação não linear ainda mais complexas.

Sobre as técnicas de amostragem houve um aumento razoável em desempenho inclusive do método proposto. Note, contudo, que esse não tratou o problema de desbalanceamento em si, parâmetro η, mesmo conseguindo um ganho de desempenho.

VI. CONCLUSÕES

Este trabalho apresentou uma comparação entre modelos de aprendizado de representação supervisionada e não-supervisionada. Os resultados sugerem que, mesmo sem informação de rótulo, o problema de aprendizado foi suficientemente tratado. Isso inspira o estudo de novas formas de direcionamento do espaço latente com o intuito de conseguir propriedades de interesse.

Além disso, foi estudado o uso de uma métrica para estimar o desempenho de generalização durante o treinamento. Experimentos mostram uma convergência do risco real com a métrica mas carece uma demonstração mais fundamentada. O processo de seleção de parâmetros também foi muito simples. Técnicas mais avançadas de otimização não linear ou até uma varredura dupla simultânea certamente indicaram um resultado melhor.

Finalmente, o método de *downsampling* não parece recomendado para o problema de desbalanceamento mas se mostrou promissor na direção de seleção de amostras. Note que não só o desempenho aumentou mas também o número de amostras foi reduzido, o que acelerou o treinamento.

AGRADECIMENTO

Este trabalho foi possível pela disponibilização das bases de dados pelo repositório *UCI Machine Learning* [7].

REFERÊNCIAS

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.
[2] G. Casella and R. L. Berger, *Statistical inference*. Cengage Learning, 2021.
[3] R. Salakhutdinov and G. Hinton, "Learning a nonlinear embedding by preserving class neighbourhood structure," in *Artificial Intelligence and Statistics*, pp. 412–419, PMLR, 2007.
[4] M. A. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE journal*, vol. 37, no. 2, pp. 233–243, 1991.
[5] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.
[6] A. Ng *et al.*, "Sparse autoencoder," *CS294A Lecture notes*, vol. 72, no. 2011, pp. 1–19, 2011.
[7] D. Dua and C. Graff, "UCI machine learning repository," 2017.