Abordagem Semissupervisionada usando *Deep Learning* Aplicada à Rotulação e Classificação de Dados

Bruno Vicente Alves de Lima[1]
Adrião Duarte Doria Neto[1]
Lúcia Emilia Soares Silva[2]
Vinicius Ponte Machado[2]

[1]Departamento de Automação e Computação (DCA)
Universidade Federal do Rio Grande do Norte (UFRN)
Natal – RN

[2]Departamento de Computação (DC)
Universidade Federal do Piauí (UFPI)
Teresina – PI

Abstract. *Large-scale data generation has brought the need for the development of intelligent techniques capable of analyzing this data automatically. In this sense, this paper proposes a semisupervisioned classification model capable of labeling unlabeled data from a few labeled examples. For this, a deep neural network was trained with labeled and unlabeled examples, simutaneally. The experiments performed show that the model is efficient in labeling data and predicting new examples.*

1. Introdução

O acúmulo de dados provenientes da informatização e principalmente do advento da Internet possibilita a prática da análise de dados, gerando conhecimento aplicável às mais diversas áreas, como por exemplo economia, indústria, saúde e educação.

Neste sentido, diversas técnicas e ferramentas computacionais têm sindo desenvolvidas a fim de auxiliar o processo de análise de dados. Dentre elas destacam-se aquelas baseadas na Aprendizagem de Máquina (AM), uma subárea da Inteligência Artificial (IA) que pode ser descrita como o desenvolvimento de técnicas computacionais que permitam a construção de sistemas capazes de adquirir conhecimento de forma automática [Mitchell 1997].

Apesar da existência de outros viés, a AM é primordialmente subdividida em supervisionada e não-supervisionada, em que, na primeira, o conjunto de dados é formado por elementos rotulados, tornando possível a tarefa de classificação, enquanto na segunda, dados não-rotulados são utilizados para extração de padrões a partir da similaridade entre eles.

Na aprendizagem supervisionada, a tarefa de rotular elementos a fim de induzir um classificador para generalizar o problema pode ser um trabalho dispendioso em tempo e custo [Amini and Gallinari 2003] [Basu et al. 2002]. Por outro lado, a análise dos padrões encontrados pelos algoritmos na aprendizagem não-supervisionada pode ser uma tarefa complexa para o ser humano. Nesse contexto, outra vertente da AM, chamada de aprendizagem semissupervisionada, têm sido estudada. Essa nova metodologia

estabelece um intermédio entre a supervisionada e a não-supervisionada, utilizando dados rotulados e não-rotulados para realizar o treinamento [Faceli et al. 2011].

Outras técnicas de AM em destaque nos últimos anos são as técnicas de *Deep Learning*, que utilizam redes neurais artificiais profundas, com muitas camadas intermediárias entre a camada de entrada e a de saída [LeCun et al. 2015]. O diferencial tecnológico dessa abordagem está nos excelentes resultados obtidos em determinadas tarefas, que superam até mesmo o desempenho de especialistas, como por exemplo, o reconhecimento de localidades e características semânticas em imagens, a vitória em jogos de estratégia e a superação de seres humanos em testes psicométricos de compreensão verbal [Goodfellow et al. 2016].

Neste trabalho propõe-se o treinamento de uma técnica de *Deep Learning*, a *Deep Belief Network*, para uma abordagem semissupervisionada a fim de rotular os elementos não-rotulados de uma base de dados.

2. Aprendizagem de Máquina

A Aprendizagem de Máquina lida com a construção de softwares que possam "aprender" com a experiência, ou seja, a própria máquina irá encontrar, após a aprendizagem, uma hipótese que melhor define o problema em questão, melhorando seu desempenho com o tempo de execução.

Os algoritmos de AM têm provado ser de grande valor prático para uma variedade de domínios, podendo ser aplicados em problemas de mineração de dados, reconhecimento de padrões e em domínios onde o programa precisa adaptar-se dinamicamente às mudanças [Russell and Norvig 2004].

2.1. Aprendizado Supervisionado

No aprendizado supervisionado têm-se um conjunto de exemplos $E = \{E_i\}|_{i=1}^{n}$ em que cada amostra $E_i \in E$ possui um rótulo associado, determinando a classe à qual a amostra pertence, de modo que E_i pode ser descrito como $E_i = (\vec{x_i}, y_i)$, em que $\vec{x_i}$ é o vetor de valores que representam os atributos da amostra e y_i é o valor da classe para a amostra.

No aprendizado supervisionado o objetivo é induzir um mapeamento geral dos vetores $\vec{x}$ para os valores y. Desta forma, o sistema de aprendizado gera um modelo $y = f(\vec{x})$, sendo f uma função desconhecida que permite predizer futuros valores y para amostras não conhecidas.

2.2. Aprendizado Não-supervisionado

No aprendizado não-supervisionado têm-se um conjunto de exemplos E, no qual cada amostra consiste em um vetor $\vec{x}$, sem a informação sobre a classe y. O objetivo é construir um modelo capaz de encontrar padrões nas amostras, formando grupos de elementos com base em suas características similares.

Desta forma, para um conjunto de dados $E = \{E_i\}|_{i=1}^{n}$, formado por vetores $\vec{x}_1, ..., \vec{x}_n$, deve-se encontrar uma similaridade entre os dados de modo que os grupos sejam formados pelos elementos mais similares possível, definindo um conjunto $c = \{c_i\}|_{i=1}^{n}$ que represente as classes das amostras.

3. Aprendizado Semissupervisionado

O aprendizado semissupervisionado é um meio termo entre os aprendizados supervisionado e não-supervisionado. Neste método, dois tipos de dados são utilizados: dados não-rotulados e rotulados. Frequentemente, estes são utilizados para determinar o rótulo daqueles [Zhu 2005].

No contexto do o aprendizado semissupervisionado, tem-se um conjunto de dados rotulados $L = (x_i^l, y_i^l)|_{i=1}^n$ e um conjunto de dados não-rotulados $U = x_i^u|_{i=1}^m$, em que $m >> n$. O objetivo é encontrar o conjunto de dados $L' = (x_i^l, y_i^l)|_{i=1}^m$, sendo este o conjunto rotulado obtido a partir do conjunto U. Ao final, têm-se que todo elemento x_i^l e x_i^u possui um rótulo y_i associado, produzindo assim um conjunto totalmente rotulado.

4. *Deep Learning*

Deep Learning é a subárea de aprendizado de máquina que estuda como solucionar problemas intuitivos. Este tipo de solução permite que computadores aprendam a partir de experiências anteriores e compreendam o mundo em termos de uma hierarquia de conceitos, no qual os conceitos mais complexos são definidos e compreendidos em termos de sua relação com conceitos mais simples e já conhecidos [Goodfellow et al. 2016].

As Redes de Crença Profunda (*Deep Belief Network* - DBN), utilizadas nesse trabalho, são uma classe de redes profundas construídas a partir de Máquinas de Bolztmann Restritas (RBM) [Goodfellow et al. 2016].

As RBMs são modelos gráficos probabilísticos não direcionados contendo uma camada de variáveis observáveis e uma única camada de variáveis latentes conectadas simetricamente, cuja função é a detecção de características. A rede atribui uma probabilidade para cada par de neurônios-vetores visíveis e ocultos de acordo com a distribuição da Equação 1.

$$P(v, h; \theta) = \frac{1}{Z(\theta)} e^{-E(v,h,\theta)} \tag{1}$$

Na equação, $Z(\theta) = \sum_v \sum_h exp(-E(v, h; \theta))$ e a energia do Sistema é dada por $E(v, h) = -a^T h - b^T v - v^T wh$, em que a representa o vetor de bias, h a camada oculta, v a camada visível e w o conjunto de pesos. Como mostra a Figura 1, pode-se construir uma *Deep Belief Network*, empilhando várias RBM's.

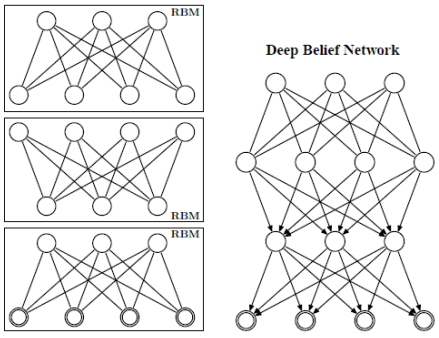


Figura 1. Representação visual da *Deep Belief Network*.

5. Abordagem Proposta

5.1. Formulação do Problema

No problema, tem-se um conjunto de dados rotulados $L = (\vec{x_i^l}, y_i^l)|_{i=1}^n$, com $y = \{y_i | y_i \in \mathbf{N}\}$ representando as classes e um conjunto de dados não-rotulados $U = \{x_i^u\}|_{i=1}^m$, em que $m >> n$. O objetivo é encontrar uma função não-linear f capaz de generalizar o conjunto $L \cup U$, para predizer $y = \{y_i | y_i \in \mathbf{N}\}$ tanto para o conjunto U, como para qualquer entrada desconhecida não-rotulada. A função f pode ser expressada pela Equação 2 para $x_i \in L$ ou $x_i \in U$.

$$f(x_i) = y_i \tag{2}$$

6. Descrição do Modelo

O modelo proposto consiste no treinamento de uma rede *Deep Learning* (DBN) utilizando dados rotulados e não-rotulados, ou seja, os dados de $L = (x_i^l, y_i^l)|_{i=1}^n$ são usados para determinar rótulos para o conjunto $U = \{x_i^u\}|_{i=1}^m$. De acordo com a Figura 2, esse o processo é realizado em quatro etapas: **Agrupamento**, **Treinamento** e **Classificação** e **Validação**.

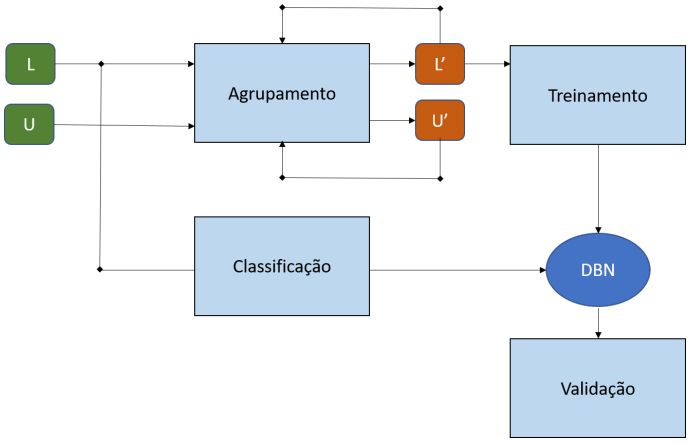


Figura 2. Representação gráfica do Modelo Proposto.

6.1. Agrupamento

Na etapa de Agrupamento, utiliza-se o conjunto de dados rotulados L para determinar a rotulação inicial dos elementos pertencentes ao conjunto U, não-rotulados. Para esta etapa leva-se em consideração o conjunto $P = \{\vec{x_i^p}\}|_{i=1}^k$, com $P \subset L$, como os k vizinhos rotulados mais próximos de uma amostra $\vec{x} = \{x_i\}_{i=1}^q | \vec{x} \in U$ e q a quantidade de característica da amostra.

Seja y o rótulo de um elemento $\vec{x} \in U$, define-se o valor de y como o rótulo dos vizinhos rotulados mais próximo ($\vec{x^p}$) e $S = \{d_i\}|_{i=1}^k$ o conjunto das distâncias entre $\vec{x}$ e todos os elementos de P, o rótulo deve ser definido se, e somente se:

1. $\forall d_i(\vec{x}, \vec{x^p}) \in S | d_i(\vec{x}, \vec{x^p}) \leq T$, sendo T um valor *threshold*;

2. c, se, e somente se, c for a classe da maioria absoluta dos k vizinhos rotulados mais próximos.

Sendo $d_i(\vec{x}, \vec{x^p})$ a distância Euclidiana [Deza and Deza 2009] entre $\vec{x}$ e $\vec{x^p}$, definida na Equação 3, e os valores de k e T definidos como parâmetros do modelo.

$$d(\vec{x}, \vec{x^p}) = \sqrt{\sum_{i=1}^{q}(x_i - x_i^p)^2} \tag{3}$$

O rótulo de uma amostra será indefinido caso as restrições não sejam satisfeitas, gerando ao fim da Etapa de Agrupamento dois outros conjuntos L' e U', representando os elementos rotulados e os elementos com rótulos indefinidos, respectivamente, vide a Figura 2. A cada iteração do Agrupamento, os elementos do conjunto L' são adicionado à L, a fim de colaborar com a definição do rótulo dos elementos que ficaram indefinidos.

Exemplificando o processo de Agrupamento, na Figura 3 têm-se um elemento não rotulado, representado por X, e elementos de três classes definidas por: Vermelha, Azul e Verde. Assumindo $k = 5$, selecionou-se k vizinhos rotulados mais próximos, sendo estes os elementos rotulados dentro do círculo.

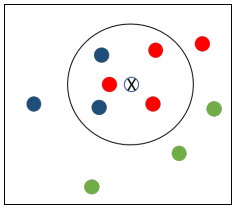
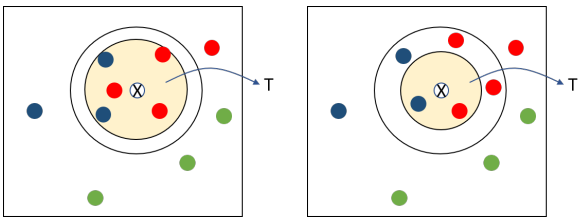


Figura 3. Seleção dos k vizinhos rotulados mais próximos.

Considerando o raio do círculo interno ao redor do elemento X como o limite de distância T, na Figura 4(a), têm-se que todos os k vizinhos rotulados mais próximos estão à uma distância menor ou igual à T, atendendo a primeira condição do modelo. De acordo com a segunda condição, a classe representada pela maioria absoluta dos elementos rotulados, neste caso a classe ”Vermelho”, deve atribuída ao elemento X.

Outro exemplo, representado pela Figura 4(b), mostra uma situação em que apenas 2 dos k elementos vizinhos rotulados mais próximos atendem à restrição do T, impossibilitando a definição do rótulo.



(a) Todos os k vizinhos rotulados mais proximos atendem a condição do T.

(b) Nem todos os k vizinhos mais próximos atendem a condição do T.

Figura 4. Representação gráfica da análise do parâmetros T.

6.2. Etapas de Treinamento e Classificação

A Etapa de Agrupamento tem como objetivo, além definir o rótulo do conjunto U, selecionar os dados de treinamento para uma rede neural capaz de generalizar o problema, uma vez que apenas o conjunto L não é suficiente por ter uma quantidade muito menor de dados que o conjunto U. Desta forma, o conjunto L', gerado pelo agrupamento é submetido no treinamento da rede neural. Neste caso, uma *Deep Belif Network*.

Esse subprocesso é definido como Etapa de Treinamento. O objetivo é treinar a DBN para gerar uma função f que seja capaz de generalizar os dados, considerando os dados rotulados e não-rotulados simultaneamente. A DBN treinada é então utilizada para predição do conjunto L na Etapa de Classificação.

6.3. Etapa de Validação

Na Etapa de Validação, a eficiência da DBN na classificação do conjunto rotulado L é analisada. Supõe-se que, se a rede é capaz de classificar corretamente os elementos de L, então esta generalizou o problema, determinando assim a condição de parada do modelo.

Pode-se afirmar que a eficiência é satisfatória quando a DBN consegue aprender utilizando dados rotulados pelo próprio modelo na Etapa de Agrupamento, ou seja, o treinamento encontrou uma função f capaz de generalizar os dados de um determinado problema.

6.3.1. Pós-Validação

Como discutido anteriormente, a Etapa de Agrupamento produz dois conjuntos de dados: o conjunto L' de dados rotulados, e o conjunto U' de dados que não obtiveram uma classe definida. Isso ocorre devido duas condições: a parada do modelo devido a convergência da rede e generalização do problema, ou, a inexistência de elementos do conjunto L que satisfaçam a condição de T para o elemento do conjunto U.

Para garantir uma rotulação completa da base, o conjunto U' é submetido à classificação pela DBN treinada. Considerando que a rede convergiu, esta pode indicar a classe dos elementos que não obtiveram classe definida na Etapa de Agrupamento.

6.3.2. Formalização do Método

O modelo apresentado pode ser formalizado pelo Algoritmo 1. As entradas são representadas pelos conjuntos $L = (\vec{x_i^l}, y_i^l)|_{i=1}^n$ e $U = \{x_i^u\}|_{i=1}^m$ de dados rotulados e não-rotulados, respectivamente; a quantidade k de vizinhos rotulados mais próximos que serão analisados, e o valor de *threshold* T utilizado para determinar a distância máxima entre os exemplos rotulados e o elemento a ser classificado. Como saída, produz-se o conjunto rotulado $L' = (\vec{x_i^l}, y_i^l)|_{i=1}^m$, os rótulos de cada amostra $x_i \in U$ e a função generalizadora f representada pela DBN treinada.

7. Metodologia dos Experimentos

Para validar o modelo proposto, um experimento inicial foi realizado utilizando um conjunto de bases de dados encontradas na literatura.

Algoritmo 1: Abordagem Proposta

Entrada: L, U, k, T
Saída: $L \cup L'$ totalmente rotulados, f

enquanto Eficácia DBN insatisfatória **faça**
  para Cada $\vec{x} \in U$ **faça**
    Tentar Definir Classe de $\vec{x}$
    se Classe Definida **então**
      adiciona em x em L'
    senão
      adiciona em x em U'
    fim se
  fim para
  Treinar a DBN com L'
  Classificar o conjunto L' com DBN
  Verificar a eficácia da DBN
fim enquanto
Classificar U' restante;

7.1. Base de Dados Utilizadas

Para conduzir os experimentos, foram utilizadas três bases de dados: Sementes, Câncer [Bennett and Mangasarian 1992] e Dermatologia. Todas elas podem ser encontradas no repositório UCI [Bache and Lichman 2013].

A base Sementes, apresentada por [Kulczycki and Charytanowicz 2011], é formada por informações referentes à identificação de três tipos de sementes de trigo. A base possui a seguinte divisão: 70 elementos do tipo *Kama*, 70 elementos do tipo *Rosa* e 70 elementos do tipo *Canadian*. Cada um dos 210 elementos é descrito por 7 características geométricas que formam o conjunto de atributos: área, perímetro, densidade, comprimento da semente, largura da semente, coeficiente de assimetria e comprimento do sulco da semente.

A base Câncer descreve a recorrência ou não de câncer de mama em pacientes. A base é formada por 699 instâncias divididas em duas classes: 241 elementos com câncer e 458 elementos sem câncer. Cada amostra é formada por 9 atributos: idade, menopausa, tamanho do tumor, invasão dos nodos, node-caps, grau de maligno, mama direita ou esquerda e quadrante do tumor.

A base Dermatologia refere-se à identificação de diferentes infecções dermatológicas através de 34 atributos que descrevem as características de cada infecção. A base contém um total de 366 elementos divididos em 6 classes, que representam as diferentes doenças, representadas da seguinte maneira: 112 elementos de *soríase*, 61 elementos de *dermatite*, 72 elementos de *líquen plano*, 49 elementos de textitpitiríase rósea, 52 elementos de *dermatite crônica* e 20 elementos de *pitiríase rubra pilar*.

7.2. Experimentos

Uma vez que as bases de dados descritas na Sub-seção 7.1 são totalmente rotuladas, para avaliar de forma semissupervisionada, dividiu-se cada base em 10 partes iguais conside-

rando a cada execução uma das partes como o conjunto rotulado (L) e o restante como o conjunto não-rotulado (U). Então, o método proposto foi executado 10 vezes para cada base de dados.

Para avaliar os resultados, utilizou-se as métricas de avaliação Acurácia, *F-score*, e *Recall* [Sokolova and Lapalme 2009] utilizando como parâmetros as classes informadas pelas bases de dados e as predições fornecidas na saída da rede neural para o conjunto de dados não-rotulados (U).

Além disso, avaliou-se a eficácia da DBN no momento da convergência, ou seja, no momento em que a DBN conseguiu generalizar o problema em cada execução, mensurando a capacidade da função f encontrada de predizer futuros dados não-rotulados. Para isso, utilizou-se as mesmas métricas descritas anteriormente: Acurácia, *F-score* e *Recall*. O método proposto foi implementado utilizando a linguagem Python, com auxílio das bibliotecas *numpy*, *sckit-learn* [Pedregosa et al. 2011] e *pandas*.

Como dito anteriormente, os parâmetros k e T são definidos como entrada do algoritmo, neste caso utilizou-se $k = 7$ e $T = 2,5$. A arquitetura da DBN compôs-se um total de 3 RBM's empilhadas, cada uma com duas camadas, sendo uma visível e uma oculta, contendo 100 neurônios cada. A taxa de aprendizado foi definida como 0.1 e a função *Relu* utilizada como ativação dos neurônios.

8. Resultados

Nesta sessão serão apresentados os resultados obtidos com a execução do método proposto para as bases de dados descritas.

A Tabela 1 apresenta os resultados da rotulação das bases de dados. Para todas as bases utilizadas, a acurácia mostra-se satisfatória, com valores entre 0,93 e 0,95 paras as bases Câncer e Sementes, respectivamente.

Considerando ainda a Tabela 1, sobre a métrica *Recall*, nota-se que o método proposto também obteve desempenho satisfatório considerando cada classe separadamente. A taxa *F-score*, reflete o desempenho da Acurácia e do *Recall*.

Bases	**Acurácia**	**Recall**	**F-Score**
Sementes	0,95	0,95	0,95
Dermatologia	0,94	0,93	0,94
Câncer	0,93	0,93	0,92

Tabela 1. Resultados da Rotulação por base de dados.

Na Tabela 2 são apresentados os resultados da DBN no momento da convergência da rede. Como pode-se notar, na Tabela 2, em todos os testes, para todas as bases de dados, a DBN apresentou Acurácia satisfatória, entre 0,93 e 0,96 para as bases Câncer e Sementes, respectivamente. As taxas *Recall* e *F-Score* refletem ainda o acerto balanceado da classificação para as classes.

Analisando as Tabelas 1 e 2, nota-se que para todas as bases de dados o algoritmo conseguiu predizer corretamente o rótulo dos elementos não-rotulados e que a DBN é capaz generalizar o problema e predizer a classe para possíveis elementos futuros.

Bases	Acurácia	Recall	F-Score
Sementes	0,96	0,95	0,96
Dermatologia	0,98	0,93	0,94
Câncer	0,93	0,94	0,94

Tabela 2. Acurácia da DBN no instante em que convergiu.

9. Conclusão e Trabalhos Futuros

Neste trabalho foi apresentado um método *Deep Learning* Semissupervisionado, capaz de gerar rótulos para dados não-rotulados utilizando o conhecimento adquirido a partir de poucos dados rotulados. O modelo pode ser aplicado à problemas em que $m >> n$, sendo n e m a quantidade de dados rotulados e não-rotulados, respectivamente.

Os experimentos realizados utilizaram três base de dados encontradas na literatura: Sementes, Dermatologia e Câncer; e foram avaliados por três métricas: Acurácia, *Recall* e *F-Score*, obtendo restultados satisfatórios para todas as métricas em todas os problemas abordados.

Com isso, conclui-se que o método proposto pode ser aplicado no problema do Aprendizado Semissupervisionado e que, além predizer os rótulos de uma base a partir de poucas amostras rotuladas, apresenta um modelo *Deep Learning* capaz de generalizar os dados, permitindo sua utilização como classificador.

Como trabalhos futuros, pretende-se melhorar a Etapa de Agrupamento, aplicando outra técnica *Deep Learning* para tal tarefa, bem como verificar a eficiência do método em bases de dados de maior dimensão, verificando o comportamento da abordagem proposta. Além disso, pretende-se estabelecer estudos para determinar o melhor valor de k e T para cada base de dados utilizada.

Referências

Amini, M.-R. and Gallinari, P. (2003). Semi-supervised learning with explicit misclassification modeling. In *International Joint Conference on Artificial Intelligence*, IJCAI'03, pages 555–560, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Bache, K. and Lichman, M. (2013). (uci) machine learning repository.

Basu, S., Banerjee, A., and Mooney, R. (2002). Semi-supervised clustering by seeding. *International Conference on Machine Learning*, pages 19–26.

Bennett, K. P. and Mangasarian, O. L. (1992). Robust linear programming discrimination of two linearly inseparable sets. *Optimization methods and software*, 1(1):23–34.

Deza, M. M. and Deza, E. (2009). *Encyclopedia of Distances*. Springer, Berlin, Heidelberg.

Faceli, K., Lorena, A. C., Gama, J., and de Carvalho, A. C. P. L. F. (2011). *Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina*. Rio de Janeiro.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Kulczycki, P. and Charytanowicz, M. (2011). A complete gradient clustering algorithm. *International Conference on Artificial Intelligence and Computational Intelligence*, pages 497–504.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.

Mitchell, T. M. (1997). Machine learning.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Russell, S. and Norvig, P. (2004). *Artificial Intelligence: A Modern Approach*.

Sokolova, M. and Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427 – 437.

Zhu, X. (2005). Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison.