CLASSIFICAÇÃO DE IMAGENS COM PROCESSAMENTO REDUCIONAL DE DADOS EM MAPAS AUTO-ORGANIZÁVEIS

Fernando Ferreira Cunha (IC) e Leandro Augusto da Silva (Orientador)

Apoio: PIBIC Mackpesquisa

RESUMO

O objetivo deste artigo é apresentar uma abordagem de redução da dimensão de imagens de conjuntos de dados (*Datasets*) com a preocupação de diminuir o percentual de perda do significado de cada amostra que decorre com a condensação de informações. Os resultados experimentais levam em consideração outras duas técnicas de representação de imagens com redução: *Thumbnail* e Histograma. A avaliação entre as técnicas apresentadas decorre da comparação entre a acurácia e o tempo despendido por três algoritmos de classificação, que são: k - Nearest Neighbours (kNN), Single Layer Neural Network (SLNN), Convolutional Neural Network (CNN). Como o processo de aprendizado de máquina é um constante ajuste em cima de uma base de dados de treinamento, quanto maior for a informação armazenada nessa base mais tempo decorre o processo de aprendizagem e maior é o poder computacional requerido. Nesse sentido que se insere a proposta do trabalho de representação de imagens utilizando Mapas Auto-Organizáveis (SOM - *Self Organizing Maps*), em que procura-se extrair características individualmente de cada dado utilizando filtros que calculam a distância correspondente entre porções de cada amostra e a respectiva área proporcional de um mapa de representação topológica de baixa dimensão de toda a base de dados (mapa de Kohonen), favorecendo dessa forma a diminuição do tamanho armazenado pelo *Dataset* e redução do tempo despendido para o treinamento e para a classificação de novos dados.

PALAVRAS-CHAVE: Mapas Auto-Organizáveis, redução de dados, classificação, acurácia, tempo de treinamento, extração de características, representação de dados, imagens.

ABSTRACT

The purpose of this article is to present a data reduction size approach of datasets with the aim of reducing the loss of meaning for each sample that results from the condensation of information, the results of the study take into account two other techniques of data representation with reduction: Thumbnail and Histogram. The evaluation between the techniques presented is based on the comparison between the accuracy and time spent by three classification algorithms, which are: k-Nearest Neighbors (kNN), Single Layer Neural Network (SLNN), Convolutional Neural Network (CNN). As the machine learning process is a constant adjustment over a training database, the longer the information stored on database the more time elapses the learning process and the greater the computational power required, through this representation technique using Self Organizing Maps (SOM) seeks to extract characteristics individually from each sample using filters that calculate the corresponding distance between portions of each sample and the respective proportional area of the topological map representation of smaller dimension taken the whole database (Kohonen map), thus favoring the reduction of the size stored by the dataset and the time spent to train a model and classificate new data.

KEYWORDS: Self-Organizing Maps, data reduction, classification, accuracy, training time, feature extraction, data representation, images.

1. INTRODUÇÃO

As técnicas de redução de dados são abordagens encarregadas de diminuir a quantidade de informação a fim de reduzir o armazenamento em memória e o tempo de execução do treinamento e da classificação de dados por algoritmos de predição (Albalate, 2007). Tradicionalmente, o conceito de redução de dados recebeu diversos nomes: edição, condensação, filtragem, desbaste, entre outros, dependendo do objetivo a ser realizado com a tarefa de redução de dados. Na literatura destacam-se duas possibilidades, dependendo do objeto a ser tratado, o primeiro cenário visa a redução da quantidade de instâncias de cada amostragem (Chen & Jozwik, 1996; Sánchez, 2004; Kohonen, 1995; Chang, 1974), enquanto que o segundo cenário visa selecionar um subconjunto de recursos entre os disponíveis no *dataset* (Dasarathy,1994; Aha et al.,1991; Hart, 1968; Toussaint et al.,1985; Tomek, 1976).

O processo de aprendizado de máquina consiste em três passos: construir uma base de dados para treinamento (*Training Set*), treinar o modelo de classificação e apresentar uma base de dados não classificada para realizar a predição. Quando utilizado algoritmos que baseiam suas regras de classificação na proximidade das amostragens, como por exemplo o algoritmo *k-Nearest Neighbours* (k-NN), a principal preocupação a se levar em conta é o tamanho das amostragens. Quanto maior o número de elementos que compõem as amostras, maior será o número de cálculos de distância realizados entre a amostragem de teste e as amostragens de treino, consequentemente maior o tempo despendido para cada processo de classificação (Albalate, 2007).

O processo de classificação refere-se à técnicas que classificam ou rotulam uma nova amostra utilizando uma função discriminante apreendida a partir de um conjunto de instâncias de uma base de treinamento. Atualmente, em muitos domínios de bancos de dados multimídia, o tamanho dos conjuntos de dados é tão grande que os requisitos para sistemas que possam armazená-los e processá-los em tempo real são custosos. Sob essas condições, classificar, compreender ou compactar as informações disponíveis pode se tornar uma tarefa muito problemática. Esse problema é especialmente dramático no caso de utilizar alguns algoritmos de aprendizado baseados em distâncias, como o *k-Nearest Neighbours* (kNN). A técnica leva em consideração a distância entre os vizinhos mais próximos de uma amostra a ser testada, enquanto que para realizar os cálculos de proximidade é necessário em tempo de execução manter os exemplos de treinamento na memória principal (Cunningham e Delany, 2007).

1.1. Objetivos

O objetivo geral do trabalho é representar por completo o *training set* estabelecido da maneira mais eficaz possível, no sentido de manter a precisão de classificação. Assim, procuramos resultados em que as necessidades de memória e tempo sejam reduzidas, enquanto a precisão da classificação original é preservada o máximo possível.

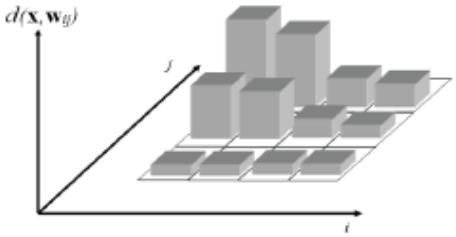
De maneira complementar, os objetivos específicos do projeto são assim definidos:

- Estudar o algoritmo de SOM como uma abordagem de redução de dimensionalidade;
- Estudar algoritmos de classificação para medição no quesito preservação das informações das bases de dados com foco na acurácia;
- Elaborar uma abordagem para extração de características de imagens com fins de redução da dimensionalidade utilizando o SOM;
- Avaliar a preservação dos dados através da acurácia de algoritmos de classificação de dados;
- Avaliar o tempo despendido pelos algoritmos de classificação em relação ao tamanho final das amostragens;
- Comparar os resultados com métodos clássicos da literatura (Thumbnail e Histograma).

2. REFERENCIAL TEÓRICO

No trabalho de Silva et al. (2013) apresentou-se uma proposta para representar imagens utilizando mapa de Kohonen, trazendo consigo uma redução na dimensão original das imagens. Em resumo o artigo propõe os seguintes passos para representar imagens com SOM: Primeiramente selecionar uma amostra e converter a matriz numérica da imagem original em um vetor, o próximo passo é calcular a distância entre o vetor da imagem e os vetores de peso (*codebook*) do mapa de kohonen gerado, como resultado uma série de valores é produzida e tomada como representação da imagem selecionada. A redução ocorre pela escolha da escala da malha do mapa gerado, uma vez que o tamanho final será reflexo da quantidade de neurônios do mapa. A Figura 1 demonstra graficamente a representação do resultado desse processo para uma imagem, onde *i* e *j* representam o índice do neurônio na grade do mapa e no eixo Y a distância entre o elemento *i* da imagem com o peso da ligação entre o neurônio e a entrada *Wij*.

Figura 1: Representação gráfica do funcionamento do Self-Organizing Map 2D



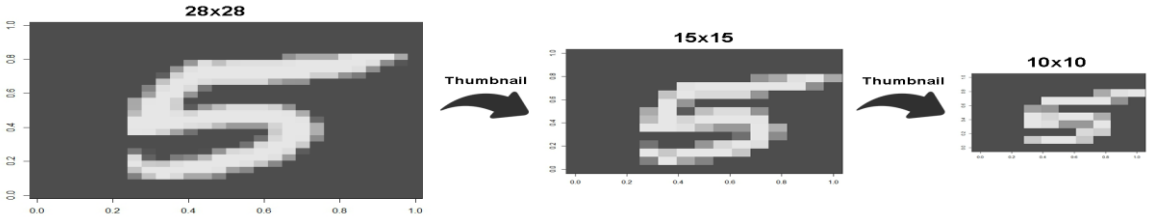
Fonte: Silva et al. (2013)

2.1. Métodos para representação de Imagens

2.1.1. Thumbnail

Thumbnail é uma técnica para redução da dimensionalidade de dados, seu algoritmo possui um baixo grau de complexidade, ocasionando em uma alta velocidade do processamento dos dados, porém seus resultados não são dos melhores quando comparados com outros métodos mais tradicionais (Silva et al., 2011). A proposta de tal algoritmo é a remoção de informações dos dados através da retirada de valores de forma intercalada nas linha e colunas, por conta de se tratar de uma técnica que envolve a perda por eliminação de informações, quanto maior for a redução mais drástico é a perda do significado de tal amostra. Na Figura 2 há a representação dos resultados da aplicação da técnica para uma amostra de imagem MNIST dígitos manuscritos.

Figura 2: Representação de imagem MNIST com técnica Thumbnail



Fonte: Autor

2.1.2. Histograma

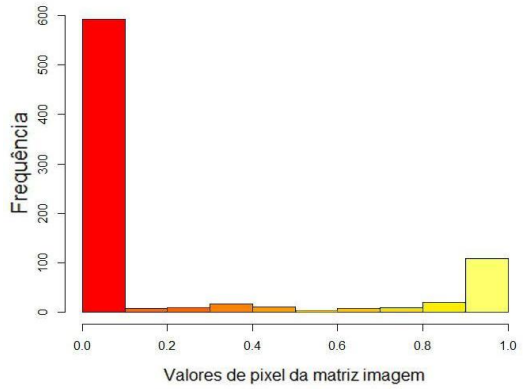
O histograma é uma técnica conhecida como distribuição de frequências, é a representação gráfica em colunas ou em barras de um conjunto de dados previamente tabulado e dividido em classes uniformes ou não uniformes. A base de cada retângulo representa uma classe. A altura de cada retângulo representa a quantidade ou a frequência absoluta com que o valor da classe ocorre no conjunto de dados para o caso de classes uniformes ou a densidade de frequência para classes não uniformes.

Em termos matemáticos, um histograma é uma função *M(i)* que conta o número de observações de cada um dos intervalos de classe. Um gráfico é apenas uma forma de representar um histograma. Então, se *n* for o número total de observações e se *k* for o número total de intervalos de classe, o histograma *M(i)* satisfaz a seguinte condição:

$$n = \sum_{i=1}^{k} m_i \quad (1)$$

Para as amostragens utilizadas, tratando-se de matrizes numéricas, a representação utilizada baseia-se no eixo X os intervalos (*breaks*) de valores de cada elemento das amostras, a quantidade de intervalos é pré-definida e sugere o tamanho final do dado e no eixo Y a frequência de ocorrências de cada valor em seus respectivos intervalos, como pode ser observado em um exemplo gráfico na Figura 3, onde a quantidade de intervalos varia de 0.1 em 0.1, totalizando 10 *breaks*.

Figura 3: Representação de amostragem MNIST por histograma



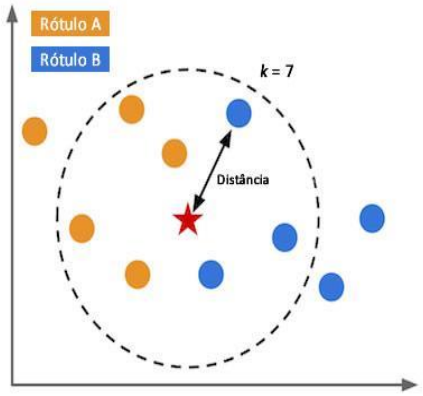
Fonte: Autor

2.2. Algoritmos de classificação de dados

2.2.1. k-NN

O algoritmo de k-NN em seu processo de treinamento consiste em armazenar e arranjar os vetores de recursos das amostragens de treinamento em um espaço de recurso multidimensional, cada um com um rótulo de classe. Na fase de classificação, utiliza-se uma constante definida pelo usuário *k* e um vetor não rotulado (ponto de consulta ou teste), que é classificado a partir da atribuição do rótulo mais frequente entre as k amostras de treinamento mais próximas a esse ponto de consulta. Uma métrica de distância comumente usada para variáveis contínuas é a distância euclidiana. Para variáveis discretas, como para classificação de texto, outra métrica pode ser usada, como a métrica de sobreposição (ou distância de Hamming).

Figura 4: Exemplo do processo classificação de k-NN com dois rótulos e k = 7

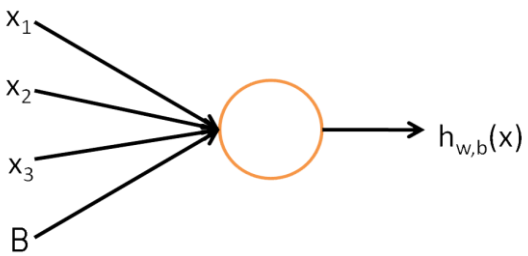


Fonte: Fukunaga e Narendra (1975)

2.2.2. Multi Layer Neural Network - MLNN

Considerando o problema de aprendizado supervisionado em que tenhamos acesso a exemplos de treinamento rotulados (x*(i)*, y*(i)*). As redes neurais fornecem uma maneira de definir uma forma complexa e não linear de hipóteses, com parâmetros *W*, *b* que pode ser ajustado aos dados obtidos. Para descrever as redes neurais é importante entender a principal e mais simples unidade funcional, que compreende um único "neurônio". A Figura 5 apresenta um diagrama representativo de um neurônio:

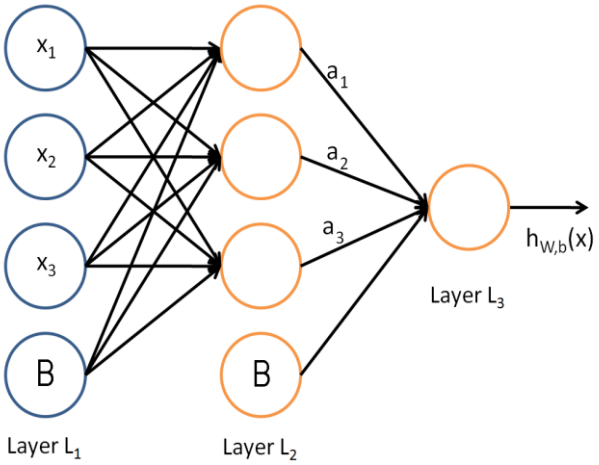
Figura 5: Diagrama de uma unidade funcional neural



Fonte: UFLDL (2018)

Uma rede neural é a conexão de *n* neurônios simples, de modo que a saída de um neurônio possa ser a entrada de outros. Cada neurônio realiza a multiplicação de seus pesos de conexão com as respectivas entradas e soma-se ao valor do bias (B), o resultado dessas operações são aplicadas em uma função denominada função ativadora, que devolverá o resultado que será passado a frente.

Figura 6: Diagrama de um modelo de rede neural simples



Fonte: UFLDL (2018)

Existem diversas construções e modelos de redes neurais artificiais, que variam em tamanho, profundidade, funções de ativação (sigmóide, tangente hiperbólica, linear retificado, etc), funções para cálculo de custo (erro), algoritmos de otimização (gradiente descendente, Adam, RMS, etc), toda essa diversidade implica diretamente no desempenho de cada modelo para cada *dataset* específico. Para fins de comparação, os modelos de redes neurais serão os mesmos para todos os *datasets* finais, será utilizado um modelo com duas camadas profundas, a primeira com 256 unidades neurais e a segunda com 128 e função de ativação retificadora (ReLU), a cada final possuirá 10 neurônios e função ativadora *softmax*. O algoritmo de otimização utilizado é o Adam, e o cálculo de erro é realizado com entropia cruzada categórica.

2.2.3. Convolutional Neural Network - CNN

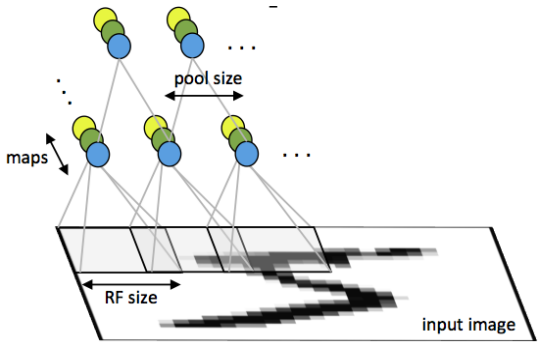
Uma rede neural convolucional (CNN) é constituída por uma ou mais camadas convolucionais (geralmente com uma etapa de subamostragem), seguida por uma ou mais camadas densamente conectadas, como em uma rede neural de multicamadas padrão. A arquitetura de uma CNN é projetada para aproveitar a estrutura bidimensional de uma imagem de entrada (ou outra matriz de entrada 2D, como por exemplo um sinal de fala). Isso é conseguido com conexões locais e pesos vinculados seguidos por alguma forma de agrupamento que resulta em recursos invariantes de tradução. Outro benefício das CNNs é que elas são mais fáceis de treinar e têm muitos parâmetros a menos do que redes totalmente conectadas com o mesmo número de unidades ocultas.

A entrada para uma camada convolucional é uma imagem em que m é a altura e largura da imagem e r é o número de canais, e. uma imagem RGB tem r = 3. A camada

convolucional terá k filtros (ou núcleos) de tamanho *n* x *n* x *q* onde n é menor que a dimensão da imagem e q pode ser o mesmo que o número de canais r ou menor e pode variar para cada núcleo. O tamanho dos filtros dá origem à estrutura conectada localmente, cada uma delas convolvida com a imagem para produzir *k* mapas de características de tamanho *m-n* + 1. Cada mapa é então subamostrado tipicamente com pool médio ou máximo sobre *p* x *p* regiões contíguas onde p varia entre 2 para imagens pequenas (por exemplo, MNIST) e geralmente não mais do que 5 para entradas maiores. Antes ou depois da camada de subamostragem, uma polarização aditiva e não-linearidade sigmoidal é aplicada a cada mapa de características. A Figura 7 ilustra uma camada completa em uma CNN consistindo de subcamadas convolucionais e de subamostragem (UFLDL Stanford, 2018).

A rede utilizada para a comparação dos datasets é composta sequencialmente por duas camada de convolução 2d com tamanho de *kernel* 3x3 uma camada de *max pooling* com *pool* de tamanho 2x2, uma camada de *dropout* com 25% de probabilidade, mais duas camadas de convolução 2d com tamanho de *kernel* 2x2 uma camada de *max pooling* com *pool* de tamanho 2x2, uma camada de *dropout* com 25% de probabilidade, uma camada densa com 512 neurônios, mais um *dropout* com probabilidade de 50% e por fim uma camada densa com 10 neurônios. As funções ativadoras entre camadas internas são do tipo ReLU e a função ativadora final é a *softmax*. O *dropout* mencionado anteriormente é um procedimento adotado apenas durante a fase de treinamento da rede, onde cada conexão tem a probabilidade estipulada de ter seu *output* anulado, isso deve-se ao fato de melhorar o ajuste da rede, fazendo com que não haja uma dependência de abstração em um único neurônio, mas que outros sejam forçados a aprender para chegar ao resultado desejado.

Figura 7: Primeira camada de uma rede neural convolucional com pooling.



Fonte: UFDL (2018)

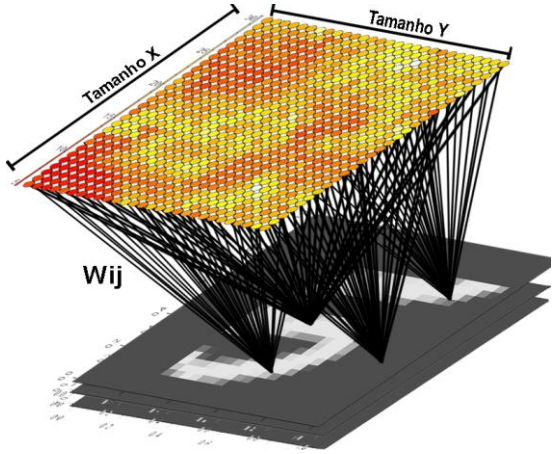
3. MAPAS AUTO ORGANIZÁVEIS - KOHONEN

Um mapa auto-organizável (SOM) está entre os algoritmos de rede neural artificial (ANN) que é treinada utilizando aprendizagem não supervisionada para produzir uma

representação de baixa dimensionalidade (tipicamente bidimensional), discretizando o espaço de entrada das amostras de treinamento, portanto, um método para reduzir a dimensionalidade. Mapas auto-organizados diferem de outras redes neurais artificiais à medida que aplicam aprendizado competitivo em oposição à aprendizagem de correção de erros, como a exemplo o *backpropagation* com gradiente descendente, os algoritmos dos mapas utilizam uma função de vizinhança para preservar as propriedades topológicas do espaço de entrada, isso torna os mapas de Kohonen úteis para a criação de visualizações de baixa dimensão de dados de alta dimensão.

A parte visível de um mapa auto-organizável consiste em componentes chamados nós ou neurônios, a dimensão do mapa é pré-definida, geralmente como uma região bidimensional finita onde os nós são organizados em uma grade hexagonal ou retangular, cada nó está associado a um vetor peso (*Wij*), que representa a conexão com as porções de entrada de cada amostra, representado na Figura 8, sendo assim, possuindo a mesma dimensão de cada vetor de entrada. Enquanto os nós no espaço do mapa permanecem fixos, o treinamento consiste em mover vetores de peso em direção aos dados de entrada, sem prejudicar a topologia induzida do espaço do mapa. Assim, o mapa de auto-organização descreve um mapeamento de um espaço de entrada de maior dimensão para um espaço de mapa de menor dimensão.

Figura 8: Mapa auto-organizável e suas conexões de peso com amostras de entrada



Fonte: Autor

O SOM pode ser considerado uma generalização não linear da análise de componentes principais (PCA) (Yin; Alexander; Kégl; Wunsch; Zinovyev, 2008). Foi demonstrado, usando tanto dados geofísicos artificiais quanto reais, que o SOM tem muitas vantagens sobre os métodos convencionais de extração de características, como Funções Ortogonais Empíricas (EOF) ou PCA (Liu & Weisberg, 2005).

3.1. Preparação dos mapas auto-organizáveis

Os estudos iniciais conduzidos, os primeiros mapas a serem utilizados possuíam uma malha cuja dimensão no eixo X era de 1 unidade e no eixo y de 28 unidades, totalizando 28 nós ou neurônios compostos no mapa. Ao avanço da pesquisa pode-se notar que com o aumento da dimensionalidade da malha do mapa de kohonen, os resultados finais de acurácia com a redução dos dados melhoraram, tomando então por final para cada *dataset* um mapa de malha com dimensões no eixo X de 28 unidades e no eixo Y de 28 unidades, totalizando 784 nós ou neurônios, a dimensão escolhida teve por motivo a abstração do tamanho original das amostras de imagens utilizadas no dataset e não houve aumento pelo processo já ser custoso, tendo em mente que quanto mais nós ou neurônios compuserem o mapa, maior o número de cálculos realizados para o treinamento da rede e consequentemente maior o tempo despendido. Os treinamentos dos mapas foram realizados com cada um dos *datasets* utilizados com seus respectivos valores normalizados.

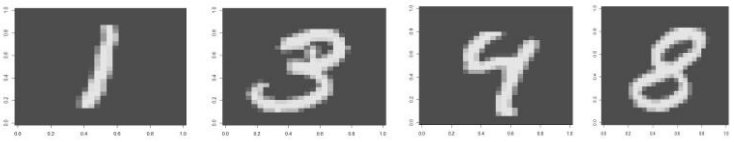
4. METODOLOGIA

4.1. Datasets utilizados

4.1.1. MNIST Dígitos Manuscritos

Os dados de MNIST dígitos manuscritos é um conjunto amplamente conhecido em trabalhos de visão computacional. Consiste em imagens de dígitos manuscritos, como pode-se observar na Figura 9.

Figura 9. Imagens do *dataset* MNIST Dígitos Manuscritos



Fonte: Autor

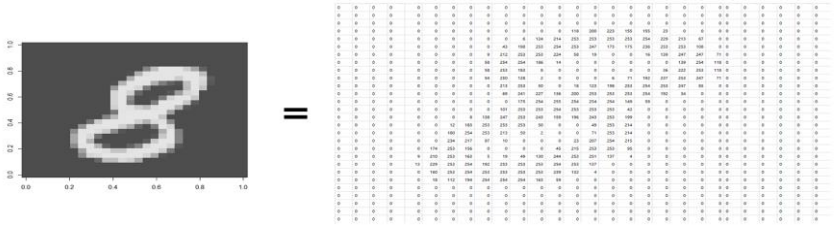
Os dados MNIST são divididos em três conjuntos: 55.000 amostras de dados para o treinamento, 10.000 amostras de dados para o teste e 5.000 amostras de dados para a validação durante o treinamento, utiliza-se a validação nos casos dos algoritmos de redes neurais, já para algoritmos como k-NN, totalizam-se 60.000 amostras de treino.

Como mencionado anteriormente, cada amostra de dados do MNIST é composta por uma imagem de um dígito manuscrito que corresponde ao *x* do *dataset*, e um rótulo

correspondente ao *y*. Tanto o conjunto de treinamento quanto o conjunto de testes contêm imagens e seus respectivos rótulos.

Cada imagem possui dimensão de 28 pixels por 28 pixels, que pode ser interpretada como uma grande matriz numérica, como ilustra a Figura 10

Figura 10. Ilustração da matriz numérica de uma amostra do *dataset* MNIST



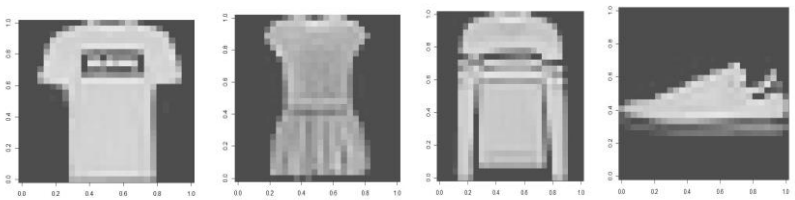
Fonte: Autor

4.1.2. MNIST Artigos de Moda

MNIST artigos de moda é um conjunto de dados de visão computacional mais complicado, devido às formas que compõem as imagens terem certa proximidade e suas diferenças serem mais sutis. É composto por imagens de peças de vestuários, como pode-se observar na Figura 11.

Os dados MNIST também são divididos em três conjuntos: 55.000 amostras de dados para o treinamento, 10.000 amostras de dados para o teste e 5.000 amostras de dados para a validação. Cada amostra de dados do MNIST é composta por uma imagem de uma peça de vestuário que corresponde ao *x* do *dataset*, e um rótulo correspondente ao *y*. As imagens são classificadas nas seguintes categorias: camisa, calça, pulôver, vestido, casaco, sandália, camiseta, sapato, bolsa e bota.

Figura 11. Imagens do *dataset* MNIST Artigos de Moda



Fonte: Autor

4.2. Preparação das amostragens

Para a utilização nos algoritmos de *Multi Layer Neural Network* (MLNN) e *k-Nearest Neighbours* (kNN) o *dataset* de treinamento e validação MNIST passou por uma

reestruturação de sua matriz de dimensão inicial 28x28 para um vetor de tamanho 784, mantendo sempre a consistência entre as imagens de forma que o processo de vetorização seja o mesmo para todas as amostras. A partir dessa perspectiva, as imagens MNIST passam a representar pontos em um espaço vetorial. Já para o algoritmo de *Convolutional Neural Network* (CNN) manteremos a estrutura 2D das amostras tendo em vista que este algoritmo explora tal estrutura para seu aprendizado e predição. No caso do *dataset* de teste, inicialmente possuímos um valor numérico correspondente ao número representado na imagem, com valores entre 0 e 9, para o algoritmo de k-NN manteremos sua estrutura, para os algoritmos envoltos em redes neurais artificiais é necessário fazer uma conversão desse único valor numérico para um vetor binário cujo tamanho é igual a quantidade de classes a serem representadas onde cada posição demarcada pelo valor 1 ditará a qual classe tal amostra pertence.

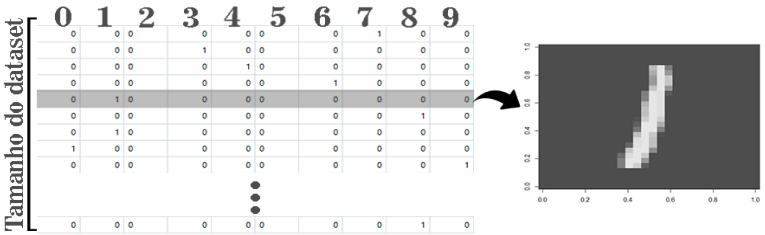
Os resultados são: dois *datasets* representados por uma matriz de dimensões 60.000/10.000 por 784, onde a primeira dimensão da estrutura é o índice da representação de uma amostra de imagem e a segunda dimensão refere-se ao índice de cada pixel composto em cada imagem, representado na Figura 12:

Figura 12: Representação visual do *dataset x* de imagens MNIST

Fonte: Autor

E dois *datasets* representados por uma matriz de dimensões 60.000/10.000 por 10, onde a primeira dimensão da estrutura é o índice do rótulo de sua respectiva imagem e a segunda dimensão representando os índices de cada classes assumida pela imagem, como mostra a Figura 13:

Figura 13: Representação visual do *dataset* y de rótulos MNIST



Fonte: Autor

4.3. Abordagem Proposta

A técnica de redução de imagens proposta consiste em calcular a distância entre porções de dimensões proporcionais aos valores de pixel da amostra de entrada em relação aos valores de peso do mapa auto-organizável (*codebook*) treinado a partir do *dataset* ao qual tais amostragens pertencem.

O primeiro aspecto a ser avaliado é a dimensão das amostragens de entrada, para a abordagem em questão será levado em consideração uma matriz de menor dimensão denominada de filtro que será utilizada para a captação das sub-matrizes das imagens. Nesse quesito é necessário que as dimensões X e Y do filtro sejam múltiplos do tamanho da amostra de entrada para que ao percorrer os valores em ambos os sentidos, não ocorra um erro de acesso a valores inexistentes. Como serão processadas imagens com dimensões de 28x28, os tamanhos de filtros selecionados foram: 2x2, 4x4, 7x7, 14x14. O segundo aspecto é calcular os intervalos que serão obtidos do *codebook*, a matriz resultante dos pesos dos vetores entre os nós do mapa auto-organizável e a matriz de entrada das amostras, terá dimensão no eixo Y correspondente à quantidade de nós compostos pelo mapa, e dimensão no eixo X correspondente à quantidade de elementos compostos pelas amostras de entrada, nos datasets utilizados será correspondente à quantidade de pixels das imagens. Tendo os tamanhos dos filtros citados acima, os intervalos utilizados para esses datasets serão o quociente entre a quantidade de pixels das imagens pelo quociente entre a dimensão do eixo correspondente ao intervalo pelo tamanho do eixo do filtro correspondente ao eixo da imagem, como os eixos são iguais em ambas as dimensões, os intervalos no eixo X e Y também resultam no mesmo tamanho, sendo para os filtros 2x2, 4x4, 7x7 e 14x14 o tamanho dos intervalos respectivamente: 56x56, 112x112, 196x196, 392x392.

Uma vez estipulado os filtros e intervalos é possível iniciar a abordagem, os cálculos são realizados amostra por amostra. Dado a matriz numérica da imagem, segmenta-se em sub-matrizes de dimensões iguais ao filtro de forma a não se sobreporem, igualmente para a matriz numérica do *codebook* do mapa segmenta-se em sub-matrizes não sobrepostas de dimensões iguais aos respectivos intervalos dos eixos. As sub-matrizes de cada porção da matriz imagem e da matriz *codebook* são convertidas em vetores e concatenadas por linha em uma nova matriz, de forma a obter na primeira linha os valores da sub-matriz da imagem e na segunda linha os valores da sub-matriz do *codebook*, note que como os valores do vetor da porção da matriz imagem são menores que os do vetor da porção da matriz codebook então são repetidas as informações do menor vetor até que se atinja o mesmo número de elementos em ambos os vetores durante a concatenação. Com essa nova matriz gerada é aplicado o cálculo de distância euclidiana a fim de se obter um único valor final representando cada porção da imagem original, a distância euclidiana entre os indivíduos *a* (vetor numérico

provindo da submatriz da imagem) e *b* (vetor numérico provindo da submatriz do *codebook*) é dado por:

$$d_{ab} = \left[\sum_{j=1}^{p} \left(X_{aj} - X_{bj}\right)^2\right]^{1/2}$$

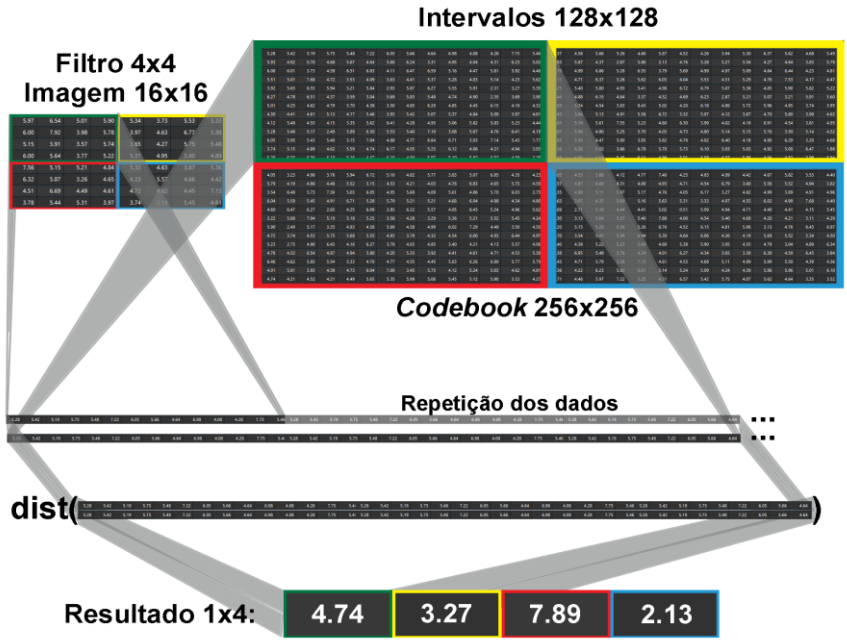
$p = 1,2,\cdots,\mathrm{n}$
$X_{aj} =$ valor do elemento j para o vetor a
$X_{bj} =$ valor do elemento j para o vetor b (2)

Esses valores resultantes são concatenados em um vetor final que representará a amostragem de entrada com a dimensão reduzida. A Figura 14 a seguir é uma representação ilustrativa do procedimento adotado em uma das porções utilizadas durante o processo proposto, a imagem de menor matriz numérica com rótulos Filtro 4x4 e Imagem 16x16 representa uma imagem de entrada escolhida do *dataset* cuja dimensão inicial é de 16x16, o filtro proposto para reduzir o tamanho da amostra tem dimensão de 4x4, assim como explicado anteriormente, por consequência de escolha do filtro com essa dimensão os intervalos gerados na matriz de pesos do mapa SOM treinado serão de 128x128, representado na imagem da maior matriz numérica cujas dimensões são 256x256, após selecionado as primeiras porções para o cálculo, indicado pelo retângulo verde, ocorre a vetorização dessas sub-matrizes extraídas da imagem original e do *codebook*, seguido da concatenação por linha dos dois vetores gerados em uma nova matriz e por fim o cálculo de distância aplicado para gerar o valor numérico representativo da porção da imagem selecionada, os valores presentes na imagem não representam necessariamente a realidade dos *datasets* e resultados finais.

Após o processamento de todas as amostras e a formação do *dataset* reduzido resultante, aplica-se uma função de normalização mínimo e máximo para redimensionar linearmente os valores de dados *x,* de cada uma das imagens com posição *i* no *dataset,* tendo um valor mínimo e um valor máximo observados em um novo intervalo arbitrário entre 0 e 1, cuja fórmula é representada a seguir:

$$z_i = \frac{x_i - \min(x)}{\max(x) - \min(x)} \quad (3)$$

Figura 14: Ilustração do processo de representação com redução de dimensão



Fonte: Autor

4.4. Método para comparação de resultados

Os resultados do processamento dos bancos de dados citados serão utilizados em algoritmos de classificação, como redes neurais artificiais e k-NN, com metodologia de validação cruzada com comparação entre as técnicas de Thumbnail e Histograma de forma a medir o desempenho envolvendo tempo e acurácia com segmentação por dimensão resultante. Para efeito de comparação no quesito acurácia, todos os parâmetros dos algoritmos de classificação foram mantidos os mesmos, para que a visão de preservação criada esteja de acordo com os resultados de amostragens originais e de seus semelhantes. Os resultados de tempo apresentam valores cujos processos foram concorrentes aos do sistema operacional, portanto serão apresentadas porcentagens de redução baseados nos tempos observados.

5. RESULTADO E DISCUSSÃO

Os resultados produzidos foram implementados na plataforma RStudio, utilizando-se dos seguintes pacotes: kohonen, keras, tensorflow, class e imager.

5.1. Resultados de Acurácia

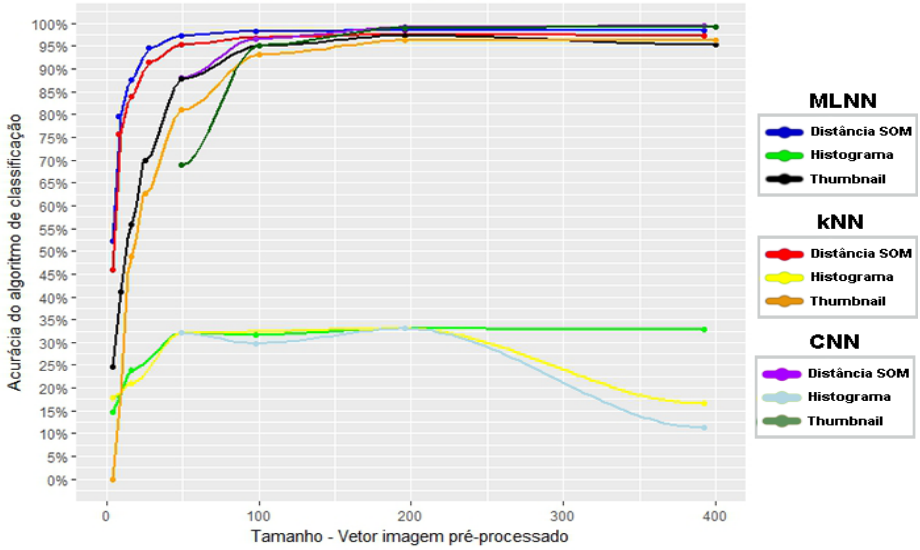
Os resultados de acurácia do *dataset* MNIST dígitos manuscritos estão descritos na tabela 1 separados pelos respectivos algoritmos de classificação utilizados.

Tabela 1. Resultados de acurácia MNIST Dígitos Manuscritos

MNIST DÍGITOS MANUSCRITOS											
KNN		Sem redução: 96,88%		MLNN		Sem redução: 98,32%		CNN		Sem redução: 99,53%	
Filtro	Tamanho	Acurácia	Perda	Filtro	Tamanho	Acurácia	Perda	Filtro	Tamanho	Acurácia	Perda
1x2	392	97,25%	-0,37%	1x2	392	98,35%	-0,03%	1x2	392	99,40%	0,13%
2x2	196	97,56%	-0,68%	2x2	196	98,40%	-0,08%	2x2	196	99,09%	0,44%
2x4	98	96,87%	0,01%	2x4	98	98,23%	0,09%	2x4	98	96,60%	2,93%
4x4	49	95,35%	1,53%	4x4	49	97,33%	0,99%	4x4	49	88,04%	11,49%
4x7	28	91,54%	5,34%	4x7	28	94,59%	3,73%				
7x7	16	83,90%	12,98%	7x7	16	87,50%	10,82%				
7x14	8	75,77%	21,11%	7x14	8	79,65%	18,67%				
14x14	4	45,87%	51,01%	14x14	4	52,12%	46,20%				

No algoritmo de k-NN com uma redução de até 93,75% do tamanho original das imagens (49 elementos) as perdas de acurácia foram inferiores à 1,53%, para o algoritmo de MLNN para uma redução de até 93,75%, as perdas de acurácias foram inferiores à 0,99%. Por fim, no algoritmo de CNN para uma redução de até 87,50% do tamanho original das imagens (98 elementos) as perdas de acurácia foram inferiores à 2,93%. Para efeito de comparação entre as técnicas de redução abordadas, foi construído gráficos de dispersão com os resultados de cada técnica em relação a um algoritmo de classificação, demonstrado na Figura 15. Os resultados apontam que em ambos os casos a abordagem de redução com SOM apresentou melhores resultados de acurácia que se tornam mais evidentes à medida em que os dados se tornam cada vez mais condensados.

Figura 15: Gráfico de dispersão com resultados de acurácia - MNIST dígitos manuscritos



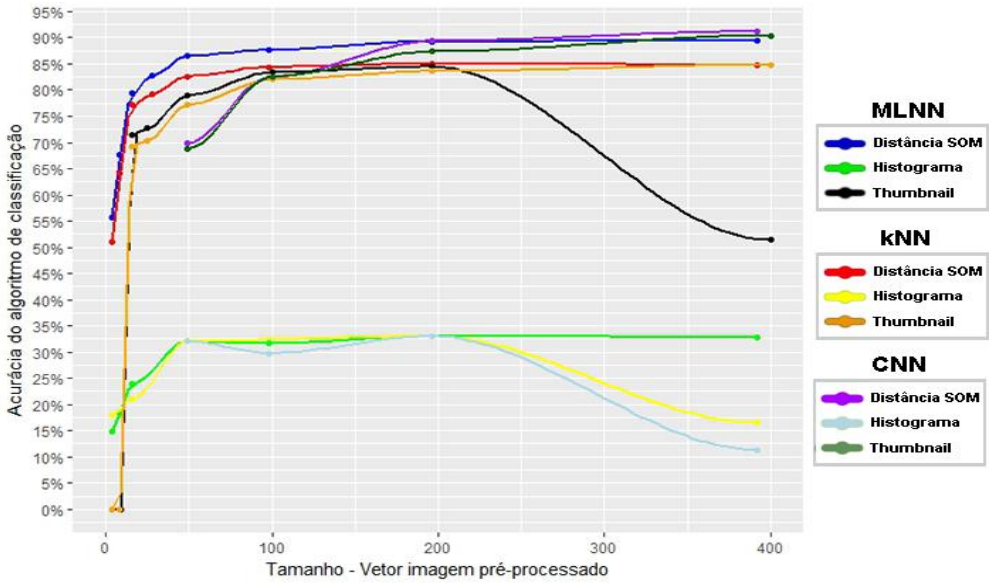
Os resultados de acurácia do *dataset* MNIST artigos de vestuário estão descritos na tabela 2 separados pelos respectivos algoritmos de classificação utilizados.

Tabela 2. Resultados de acurácia MNIST artigos de vestuário

MNIST ARTIGOS DE MODA											
KNN		Sem redução: 85,3%		MLNN		Sem redução: 89,75%		CNN		Sem redução: 93,19%	
Filtro	Tamanho	Acurácia	Perda	Filtro	Tamanho	Acurácia	Perda	Filtro	Tamanho	Acurácia	Perda
1x2	392	84,79%	0,51%	1x2	392	89,42%	0,33%	1x2	392	91,18%	2,01%
2x2	196	85,06%	0,24%	2x2	196	89,27%	0,48%	2x2	196	89,35%	3,84%
2x4	98	84,41%	0,89%	2x4	98	87,66%	2,09%	2x4	98	82,39%	10,80%
4x4	49	82,65%	2,65%	4x4	49	86,52%	3,23%	4x4	49	69,87%	23,32%
4x7	28	79,21%	6,09%	4x7	28	82,76%	6,99%				
7x7	16	77,27%	8,03%	7x7	16	79,52%	10,23%				
7x14	8	64,04%	21,26%	7x14	8	67,64%	22,11%				
14x14	4	51,11%	34,19%	14x14	4	55,65%	34,10%				

No algoritmo de k-NN com uma redução de até 93,75% do tamanho original das imagens (49 elementos) as perdas de acurácia foram inferiores à 2,65%, para o algoritmo de MLNN para uma redução de até 93,75%, as perdas de acurácias foram inferiores à 3,23%. Por fim, no algoritmo de CNN para uma redução de até 75,00% do tamanho original das imagens (196 elementos) as perdas de acurácia foram inferiores à 3,84%. A comparação entre as duas outras técnicas também foi utilizada para o *dataset* artigos de vestuário através de um gráfico de dispersão com linhas de tendência referidos na Figura 16.

Figura 16. Gráfico de dispersão com resultados de acurácia - MNIST artigos de vestuário



5.2. Resultados de Tempo

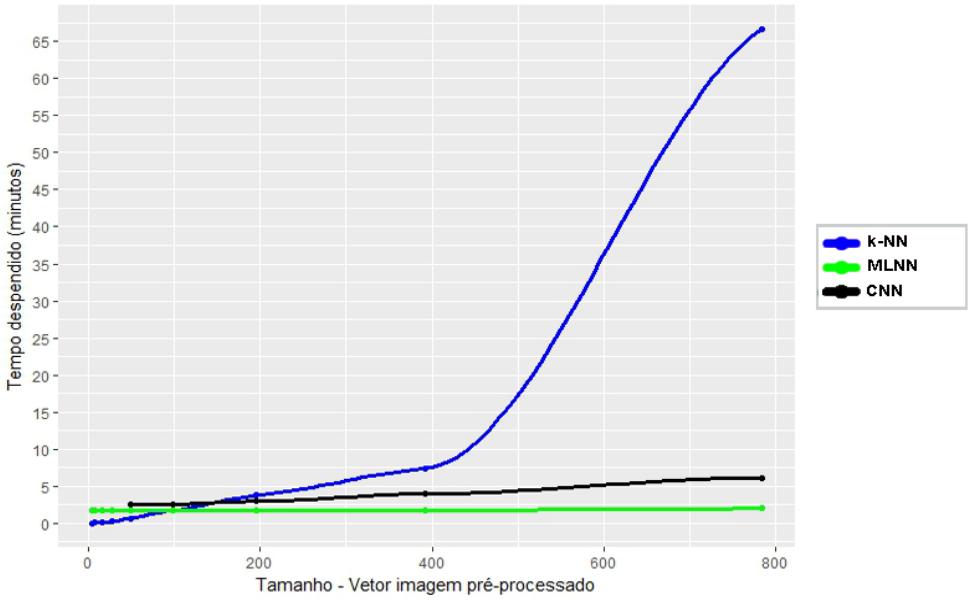
Os resultados de tempo e velocidade ganha com a redução do tempo despendido, estão descritos na Tabela 3 separados pelos respectivos algoritmos de classificação utilizados:

Tabela 3. Resultados de tempo despendido e ganho com a redução da dimensionalidade dos dados

TEMPO MÉDIO DESPENDIDO											
KNN		Sem redução: 1.11 h		MLNN		Sem redução: 2 min		CNN		Sem redução: 6.15 min	
Filtro	Tamanho	Tempo	Redução	Filtro	Tamanho	Tempo	Redução	Filtro	Tamanho	Tempo	Redução
1x2	392	7.41 min	88,87%	1x2	392	1.78 min	11,00%	1x2	392	3.98 min	35,28%
2x2	196	3.77 min	94,34%	2x2	196	1.78 min	11,00%	2x2	196	2.99 min	51,38%
2x4	98	1.77 min	97,34%	2x4	98	1.78 min	11,00%	2x4	98	2.6 min	57,72%
4x4	49	40 seg	98,99%	4x4	49	1.78 min	11,00%	4x4	49	2.6 min	57,72%
4x7	28	13.74 seg	99,65%	4x7	28	1.78 min	11,00%				
7x7	16	7.10 seg	99,82%	7x7	16	1.78 min	11,00%				
7x14	8	4.12 seg	99,89%	7x14	8	1.78 min	11,00%				
14x14	4	2.43 seg	99,94%	14x14	4	1.72 min	14,00%				

Pode-se notar que a partir de uma redução de 50% do tamanho original das amostras totalizando 392 elementos por imagem, apresentou-se um salto na velocidade de processamentos das amostragens para o algoritmo de k-NN, reduzindo em até 99,94% o tempo despendido para a predição de todo o *dataset*. A Figura 17 representa um gráfico de dispersão com linha de tendência da redução do tempo despendido pelos algoritmos de classificação conforme a redução do tamanho de cada amostragem.

Figura 17. Gráfico com resultados do tempo despendido pelos algoritmos de classificação



Fonte: Autor

6. CONSIDERAÇÕES FINAIS

Pode-se concluir com os resultados demonstrados anteriormente, que a técnica estudada apresentou ótimos resultados para o objetivo proposto, reduzir a dimensão ou tamanho dos dados de multimídia para processamento e classificação, sugerindo um viés para abordagens que visam o mesmo objetivo, trabalhando com a distância entre as imagens

e um mapa dimensional que traduza e extrair características de todo a base de dados a ser trabalhada.

Com reduções de até 75% do tamanho original dos dados, pode-se obter resultados com ganhos ou perdas de acurácia ínfimos e um ganho na velocidade de processamento e classificação de até 94,34% além do ganho no armazenamento de toda a informação.

Os próximos passos consistem em trabalhar com imagens que representem cenários ou objetos compostos e outros tipos de dados, avaliando o desempenho da abordagem e seu comportamento em relação às acurácias com novas reduções.

7. REFERÊNCIAS

LOZANO ALBALATE, M.T. Data Reduction Techniques in Classification Processes: Castellón, 2007.

CUNNINGHAM, P. & DELANY, S.J. k-Nearest Neighbour Classifiers. Technical Report UCD-CSI-2007-4 March 27, 2007.

YIN, H.; Learning Nonlinear Principal Manifolds by Self-Organising Maps, in Gorban, ALEXANDER N.; KÉGL, B.; WUNSCH, D.C.; & ZINOVYEV, A.; Principal Manifolds for Data Visualization and Dimension Reduction, Lecture Notes in Computer Science and Engineering (LNCSE), vol. 58, Berlin, Germany: Springer, 2008.

LIU, Y. & WEISBERG, R.H. Patterns of Ocean Current Variability on the West Florida Shelf Using the Self-Organizing Map. Journal of Geophysical Research, 2005.

SILVA L.A., DEL-MORAL-HERNANDEZ E., MORENO R.A., FURUIE S.S. Combining Wavelets Transform and Hu moments with Self-Organizing Maps for Medical Image Categorization. Journal of Electronic Imaging 1, 1–20, 2011.

SILVA L.A., PAZZINATO B., COELHO O.B., Image Representation Using the Self-Organizing Map, College of Computing and Informatics, Mackenzie Presbyterian University, 2013.

UFDL (UNSUPERVISED FEATURE LEARNING AND DEEP LEARNING) Stanford. Disponível em: <http://ufldl.stanford.edu/tutorial/> Acesso em: 20 de julho de 2018.

DASARATHY B.V., Minimal Consistent Subset (MCS) Identification for Optimal Nearest Neighbor Decision Systems Design, IEEE Trans. on Systems, Man and Cybernetics 24,1994.

AHA ET AL. D.W., D. KIBLER, M. K. ALBERT, Instance-based Learning Algorithms, Machine Learning, 1991.

HART P.E., The Condensed Nearest Neighbor Rule, IEEE Trans. on Information Theory 14 no. 5, 1968.

TOUSSAINT ET AL., G.T. TOUSSAINT AND B.K. BHATTACHARYA AND R.S. POULSEN, The Application of Voronoi Diagrams to Nonparametric Decision Rules, Computer Science and Statistics: The Interface L. Billard, Elsevier Science, North Holland, Amsterdam, 1985.

TOMEK I., Two Modifications of CNN, IEEE Trans. on Systems, Man and Cybernetics 6, 1976.

CHEN C.H., J´OZWIK A., A Sample Set Condensation Algorithm for the Class Sensitive Artificial Neural Network, Pattern Recognition Letters, 1996.

SÁNCHEZ J.S., High Training Set Size Reduction by Space Partitioning and Prototype Abstraction, Pattern Recognition 37 no. 7, 2004.

KOHONEN T., Self-Organizing Maps: Springer-Verlag, 1995.

CHANG C.L., Finding Prototypes for Nearest Neighbor Classifiers, IEEE Trans. on Computers, 1974.

FUKUNAGA, K.; NARENDRA, P. M. A branch and bound algorithm for computing k-nearest neighbors. IEEE Transactions on Computers, v. 100, n. 7, p. 750–753, 1975.

Contatos: fernandofc16@gmail.com e leandroaugusto.silva@mackenzie.br