Projeto 1 - Fundamentos em Sistemas Inteligentes

Renan Godoi de Medeiros
Matrícula: 15/0146612

***Abstract*—O projeto possui o intuito de criar um sistema inteligente, que seja capaz de classificar dígitos manuscritos através dos algoritmos de classificação LDA e Knn.**

I. Introdução

Este artigo tem por objetivo descrever o projeto da disciplina de Fundamentos em Sistemas Inteligentes e demonstrar as capacidades e limitações dos algoritmos de classificação K-Nearst Neibor e Linear Discriminant Analysis no âmbito de Sistemas Inteligentes.

Os tópicos foram divididos em 6 partes, o tópico II explica de forma clara os problemas que acarretaram no desenvolvimento do projeto assim como as dificuldades encontradas no mesmo.O tópico III explica a razão pela qual tentei resolver os problemas citados em II.

Ainda no tópico IV eu descrevo os materiais utilizados para a realização do projeto, assim como os métodos que eu utilizei no desenvolvimento do mesmo, em seguida, no tópico V é apresentada de forma clara como o projeto foi desenvolvido e também como ele foi dividido dentro do código do software. No tópico VI descrevo os resultados encontrados após uma série de testes e pesquisas a respeito.E por fim em VII apresento as minhas conclusões a respeito do trabalho, assim como as experiências que este trabalho me proporcionou.

II. Problema

Os problemas que envolvem desenvolver um software que seja capaz de realizar uma boa classificação são vários, entretanto o maior problema encontrado foi a escolha da linguagem de programação e a metodologia para que fosse possível realizar a classificação dos manuscritos, pois apesar de existir bastante material e bibliotecas para serem utilizadas na implementação, a falta de informação sobre as mesmas prejudicou muito o andamento do trabalho.

Na parte de implementação, a problematização se deu na leitura dos dados, já que os arquivos foram compactados em um formato desconhecido mas parecido com binário, assim foi necessário ler os dados relacionados ao tamanho de cada item dentro do arquivo, e na aplicação do algoritmo LDA e Knn pois a linguagem utilizada para o experimento não possui uma boa otimização, fora as limitações da linguagem em si, tanto que não foi possível modularizar o código para uma melhor legibilidade para quem fosse utiliza-lo como base para outros trabalhos acadêmicos.

Também obtive problemas em relação ao tempo disponível para a entrega do trabalho, devido as outras tarefas e trabalhos que me foram designados a serem feitos no mesmo prazo deste trabalho, mesmo utilizando como base outras pesquisas e algoritmos produzidos e colocados em código aberto na internet, ainda foi difícil sintetizar o código devido a extrema dificuldade de manuseio da linguagem utilizada.

III. Por que?

O motivo pelo qual quis resolver os problemas que acarretaram no desenvolvimento do trabalho foi é claro o prazo de entrega pois era um trabalho extenso para se fazer em muito pouco tempo, e também o mais importante o aprendizado, já que as minhas pesquisas sobre a linguagem que utilizei até então eram desconhecidas para mim, até porque não tive muita experiência com essa linguagem de programação ao longo do meu curso.

A outra razão pela qual me motivou a resolver os problemas, é simplesmente por que eu gosto de resolver problemas, meu curso acadêmico é voltado para isso, para resolução de problemas de diversos tipos que envolvem soluções computacionais.

IV. Material e Métodos

Utilizei alguns artigos,livros e alguns fórums na internet para a construção do software, depois de muita pesquisa a respeito de qual linguagem seria a ideal, a linguagem adotada para a realização do projeto foi a linguagem R, uma linguagem que é comumente utilizada em computação estatística e gráfica para realização de trabalhos desse tipo, a escolha desta linguagem se deu pela facilidade aparente que ela proporcionava para este tipo de trabalho, devido às suas funções relacionadas tanto a LDA quanto a Knn que a própria linguagem proporciona para uso geral, e também a facilidade de criar gráficos e relações matemáticas dentro do programa.

Para facilitar o desenvolvimento eu utilizei a IDE do RStudio que é disponibilizada na internet para o uso em geral, e também utilizei o software Astah para modelagem do projeto, para facilitar a vizualização de como o projeto foi construido(apesar da linguagem R não possuir muito suporte a modularização como outras linguagens).Foi utilizado também a plataforma github para evitar eventuais problemas em relação ao computador utilizado no experimento.

V. Implementação do projeto

Primeiro para a criação do projeto, comecei pela modelagem do software, devido a dificuldade de escolher a linguagem de programação a ser utilizada neste projeto.

Foram criados no total 3 diagramas de modelagem, o primeiro diagrama que foi feito foi o diagrama geral do projeto, como é mostrado a baixo

Como é posssível perceber a main possui uma relação de dependência entre os módulos LeituraDados e ProcessamentoDados, enquanto o ProcessamentoDados dependeria apenas

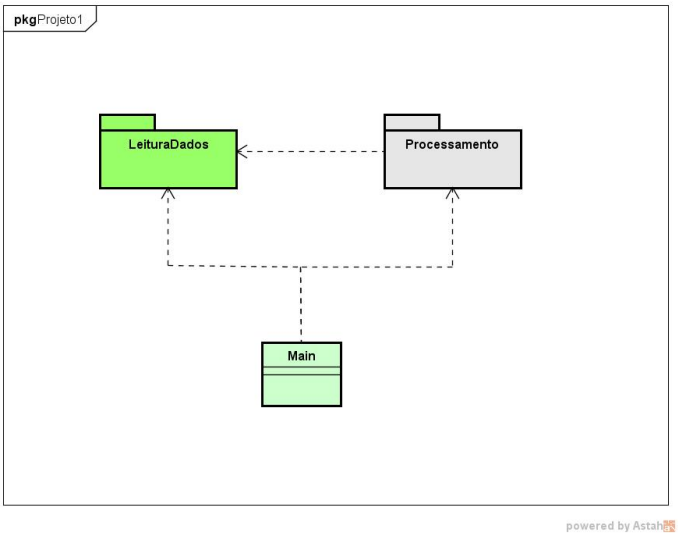

Fig. 1. Diagrama geral

de LeituraDados.Como o próprio nome sugere, LeituraDados é responsável pela leitura dos dados no banco de dados MNist disponibilizado na especificação do projeto, logicamente existiam diversas maneiras para fazer a leitura de dados do banco MNist em R, entretanto escolhi um algoritmo e adaptei de acordo com o modelo que havia criado.

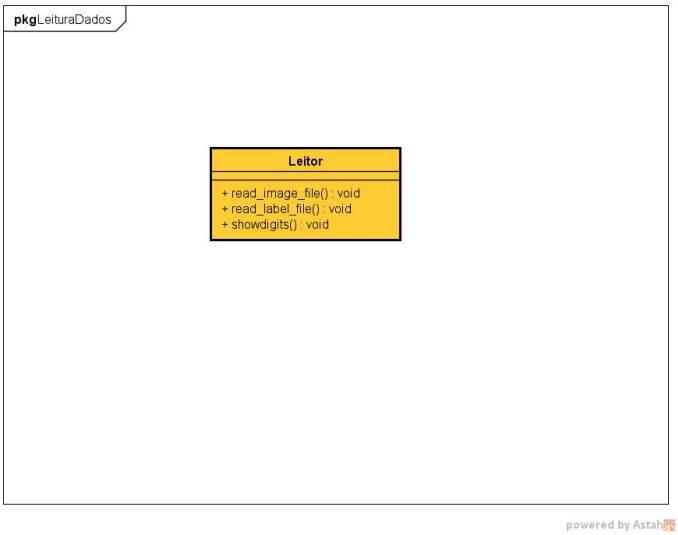

Fig. 2. Diagrama do módulo de Leitura

Na figura 2 é possível observar que este é um diagrama simples, pois este módulo faria apenas a leitura dos dados do banco de dados, entretanto para verificar se os dados eram lidos corretamente decidi criar uma função para tal, que fosse capaz de mostrar os números renderizados que o software aparentemente iria ler, e mostrá-los na tela assim como as imagens lidas do banco de dados.

Como pode ser observado o modulo Leitor é composto por 3 funções, a *read_image_file()* responsável pela leitura das imagens presentes no arquivo do banco de dados, a *read_label_file()* responsável pela leitura das etiquetas correspondentes aos números manuscritos das imagens, e também a *showDigits()* responsável por mostrar as imagens juntamente com as etiquetas lidas na tela.

Em seguida foi construido o módulo de processamento como pode ser visto na figura 3 que seria responsável por processar e produzir os gráficos e a matriz de confusão gerada após a realização dos processos dos algoritmos de classificação Knn e LDA, assim como seria possível produzir um gráfico para demostrar o funcionamento dos algoritmos.

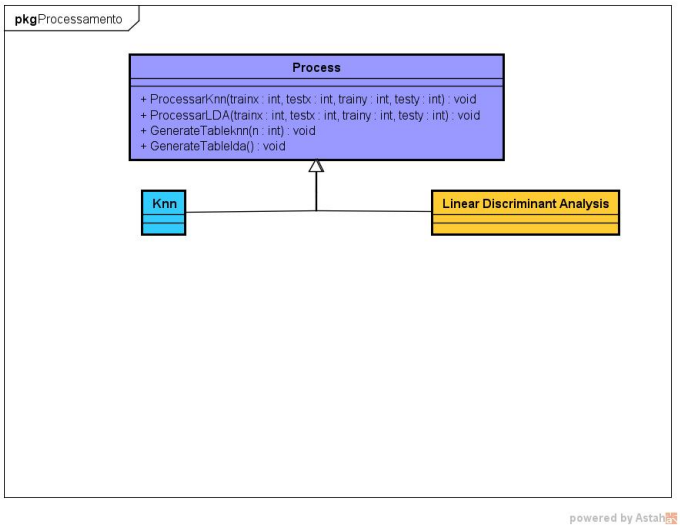

Fig. 3. Diagrama do módulo de Processo

Como é perceptível na figura acima, o processo herda os módulos Knn e LDA e é composta por 4 funções específicas, *ProcessarKnn(trainx, testx, trainy, testy)* que basicamente processa os dados utilizando o algoritmo Knn, recebe é claro dados de teste e os dados para o treino do algoritmo.Da mesma forma *ProcessarLDA(trainx, testx, trainy, testy)*, recebe os mesmos parâmetros, porém com a diferença que o algoritmo utilizado para realizar a classificação é o LDA, *GenerateTableKnn(n)* gera uma matriz de confusão mostrando estatísticamente os acertos e erros do algoritmo utilizado juntamente com as suas frequências, e *GenerateTablelda()* da mesma forma que a função *GenerateTableKnn(n)*, também gera uma matriz de confusão, entretanto a matriz corresponde ao processo do algoritmo LDA.

VI. Resultados

Os resultados não foram muito satisfatórios, apesar da grande quantidade de funções e facilidades que o R proporciona e que foram essenciais para este trabalho, não existe uma maneira simples de aplicá-las.

Mesmo na leitura dos arquivos foi um processo complicado, pois o R não possui facilidade nenhuma para acessar os arquivos do banco de dados, e infelizmente não existe uma maneira simples de acessar o diretório local em que o script está sendo executado, assim utilizei uma função do próprio R

para que o usuário busque os arquivos dos bancos de dados para que o software possa realizar as operações.Entretanto a leitura dos arquivos foi bem sucedida como é mostrado na figura

A. *Classificador Knn*

Para realizar o experimento primeiro comecei com o Knn que aparentava ser o menos trabalhoso, como a quantidade de dados era muito grande e isso afetava muito negativamente no desempenho do software para mostrar os resultados calculados, utilizei uma amostra de aproximadamente 100 imagens e labels, para que fosse possível verificar o comportamento do programa, ainda para o algoritmo Knn os valores de K testados para este experimento foram K=1,K=5 e K=20.

No final o experimento foi bem sucedido, visto que o algoritmo retornou um valor de aproximadamente 67% de acerto para k=1, sendo que a amostra utilizada não foi tão grande,entretanto para valores k $>$ 1, como k=5 ou k=20 o acerto diminuiu significativamente variando entre 1 e 12%

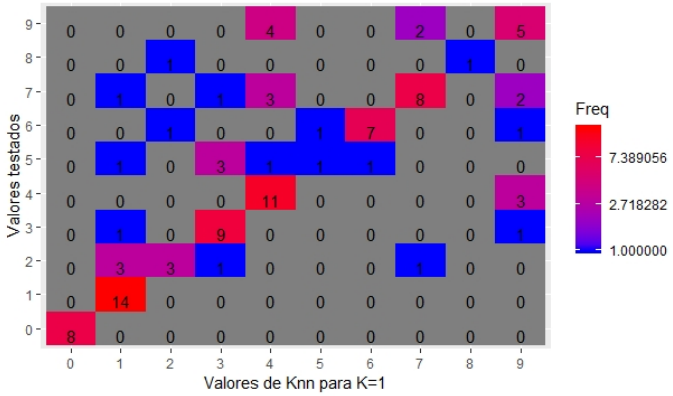

Fig. 4. Matriz de confusão para k=1

A matriz de confusão mostra de uma maneira mais clara qual o dígito que o algoritmo aparentemente errou, e como pode ser visto o algoritmo obteve mais erros nos dígitos 3,4 e 9, como é mostrado na legenda da figura 4 isso provavelmente se deve a grande semelhança entre seus manuscritos.

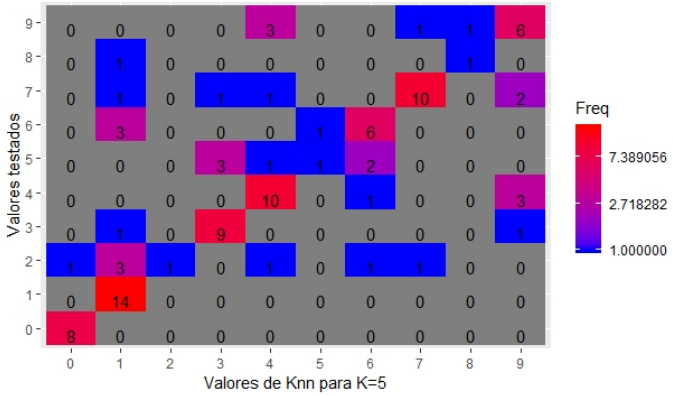

Fig. 5. Matriz de confusão para k=5

A partir deste cenário onde k=5 já é perceptível que a quantidade de erros que o algoritmo comete é maior em relação a k=1,entretanto os valores ainda permanecem com uma boa taxa de acerto, já que a frequência ainda está no limiar de acertos da matriz.

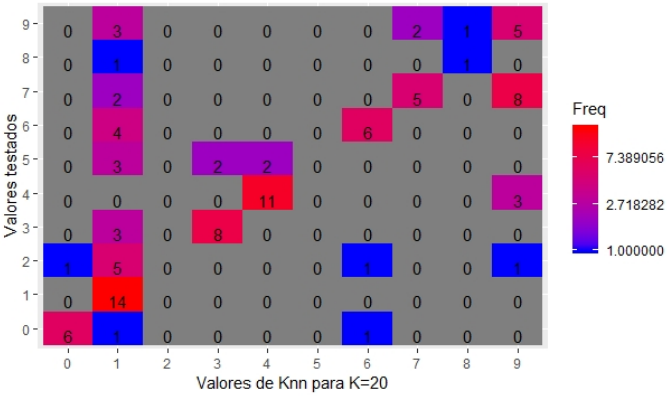

Fig. 6. Matriz de confusão para k=20

A partir deste ponto, para k=20, as taxas de acerto do algoritmo já começam a piorar um pouco em relação aos outros valores de k testados, percebe-se que para o valor 2 e 5, o algoritmo não conseguiu sequer acertar algum número, entretanto o maior problema está no 1, já que aparentemente o algoritmo previu erroneamente uma boa parte dos números pensando que era o número 1 por algum motivo desconhecido, até porque o número 1, comparado aos outros números é muito mais fácil de ser discernido. Portanto o melhor resultado realmente foi para k=1.

B. *Classificador LDA*

Infelizmente não consegui utilizar o classificador LDA, por diversos motivos, eu tentei várias vezes de diferentes formas possíveis, entretanto sempre ouvera um erro por mais que tentasse normalizar os valores dos dados de entrada(diferente do algoritmo Knn o LDA não normaliza os valores calculados por ele), por isso o desenvolvimento dessa parte só ficou no modelo mesmo.

VII. Conclusões

Neste trabalho abordei o assunto de sistemas inteligentes, com o objetivo de mostrar o funcionamento dos algoritmos de classificação LDA e Knn, analisando os resultados obtidos analisando a performance e se eram satisfatoriamente bons ou não.Também foi feita uma visão geral de como o projeto foi realizado,além dos softwares e IDE's que foram utilizados na sua construção.

Não consegui cumprir todos os objetivos do trabalho pois o curto prazo e as outras responsabilidades me afetaram bastante a desenvolver o projeto, mas oque mais dificultou foram as limitações e problemas da linguagem R.

Este trabalho foi muito importante para o meu conhecimento, aprendi várias coisas em relação a sistema inteligentes

além da linguagem R que até então não tinha muito conhecimento sobre.O projeto levou bastante tempo para ser construido, mas apesar das dificuldades foi interessante, a àrea de inteligência artificial é muito vasta principalmente quando pensamos em sistemas.

REFERENCES

[1] Exemplo do algoritmo Knn em R https://rstudio-pubs-static.s3.amazonaws.com/123438_3b9052ed40ec4cd2854b72d1aa154df9.html

[2] 3-2-1-0 Classifying Digits with R https://rpubs.com/zkajdan/235844

[3] R Classifying Handwritten Digits (MNIST) using Random Forests http://beyondvalence.blogspot.com/2014/01/r-classifying-handwritten-digits-mnist.html

[4] Gareth James, Daniela Witten, Trevor Hastle, Robert Tibshirani An Introduction to Statistical Learning with applications in R, 2014

[5] Github Project, Load Mnist data in R https://gist.github.com/brendano/39760