Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Estudo sobre Clusterização de Dados Biomédicos com Técnicas de Algoritmos Genéticos

Ícaro Marcelino Miranda

Monografia apresentada como requisito parcial
para conclusão da disciplina Estudos Em Inteligência Artificial

Orientador
Prof. Dr. Marcelo Ladeira

Coorientador
Prof. Dr. Claus de Castro Aranha

Brasília
2015

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Estudo sobre Clusterização de Dados Biomédicos com Técnicas de Algoritmos Genéticos

Ícaro Marcelino Miranda

Monografia apresentada como requisito parcial
para conclusão de disciplina Estudos Em Inteligência Artificial

Prof. Dr. Marcelo Ladeira (Orientador)
CIC/UnB

Prof. Dr. Claus de Castro Aranha
Department of Computer Sciences/University of Tsukuba

Prof. Dr. Guilherme Novaes Ramos
CIC/UnB

Brasília, 15 de dezembro de 2015

Dedicatória

Dedico esse trabalho à minha família, que me dá todo o apoio possível. À minha mãe, que despertou meu interesse pelas aplicações na área médica. Ao meu irmão, que sempre me ajuda quando necessário. Que o conhecimento aqui adquirido, possa ser útil, de alguma forma, para a melhoria da sua qualidade de vida.

Agradecimentos

Agradeço à minha mãe por prover todo o suporte e apoio necessário, apesar das dificuldades, para os meus estudos.

Agradeço aos professores Marcelo Ladeira e Claus Aranha, que guiaram o desenvolvimento desse projeto, pela oportunidade de participar dessa iniciativa.

E agradeço à equipe, Yuri Lavinas e Gabriel Ferreira, pelas boas discussões e recomendações.

Resumo

Esse projeto aborda o desenvolvimento de um modelo que possa auxiliar a prevenção e tratamento de doenças causadas por microorganismos. Através de técnicas de *Clustering* otimizadas com algorítmos genéticos, espera-se encontrar um modelo eficaz sobre as bases de dados disponíveis. Com os dados do Ministério da Saúde sobre resistência de microorganismos à antibióticos em diversas pessoas - diferentes idades, gênero e região que habita - serão extraídas informações implícitas nos dados que possam resolver à problemática inicial. Esta primeira fase trata do aprendizado dos conceitos fundamentais para realização do projeto, levantando as principais técnicas, e o desenvolvimento de um primeiro modelo que poderá ser utilizado sobre os dados.

Palavras-chave: Agorítmos genéticos, Medicina preventiva, Clusterização, Dados biomédicos, Microorganismos, Antibióticos

Abstract

This project deals with the development of a model that can help prevent and treat diseases caused by micro-organisms. Using Clustering techniques optimized with genetic algorithms, it is expected to find an effective model about the available databases. With the data from Brazilian Ministry of Health about micro-organisms' resistance to antibiotics in different people - different ages, gender and region that inhabits - is expected to find implicit information that can solve the initial problem. To be able to develop the project the basic concepts were studied, gathering the key techniques and the development of a first model that will be used on the data.

Keywords: Genetics algorithms, Preventive medicine, Clustering, Biomedic data, Micro-organisms, Antibiotics

Sumário

Lista de Tabelas

Capítulo 1

Introdução

Nesse capítulo é elaborada uma descrição geral do projeto, definindo-o, apresentando seus objetivos e algumas considerações sobre a proveniência dos dados que serão utilizados. Esse projeto será desenvolvido em parceria com o Prof. Dr. Claus Aranha da Universidade de Tsukuba, Japão.

1.1 Definição do Problema

Medicina preventiva é uma área da saúde que objetiva prevenir a ocorrência de doenças, assim evitando gastos com tratamentos e melhorando a qualidade de vida do paciente. É de suma importância, principalmente em regiões menos afortunadas que ainda possuem problemas de infraestrutura e consequentemente saneamento básico deficiente, aumentando a incidência de doenças, principalmente infectocontagiosas.

Ainda hoje é difícil identificar muitas doenças, mesmo com sintomas iniciais. Já que um único sintoma é comum a muitos problemas. Algumas vezes, nem o próprio paciente consegue notar todas as anormalidades que estão ocorrendo no seu corpo. Os exames de rotina são extremamente importantes nesse aspecto. Porém, pequenas perturbações nas informações coletadas podem passar despercebidas.

Cada vez mais se têm novas informações sobre problemas de saúde, sejam sobre suas causas ou efeitos. Essa massa de informação se torna mais densa, logo, mais difícil de ser interpretada. Isso se torna um incentivo para uma análise superficial ou parcial, que pode esconder informações de muita relevância.

No tratamento de doenças causadas por microorganismos são usados antibióticos - compostos naturais ou sintéticos capazes de inibir o crescimento ou causar a morte de fungos ou bactérias [8]. Porém seu uso constante pode trazer consequências piores que os efeitos dos microorganismos.

A falta de conhecimento do medicamento correto para determinada situação, a falta de informação sobre as características do paciente são os principais fatores envolvidos na ocorrência das reações indesejadas pelo medicamento [14].

Ferramentas computacionais já se mostraram um auxílio poderoso em diversos tratamentos médicos, por sua precisão e confiabilidade. De modo que, utilizando-as, é possível gerar resultados potencialmente mais precisos e rápidos que técnicas anteriores.

Através de mineração de dados (do inglês, data mining - DM), é possível agrupar dados seguindo suas características, diferenciando-os e, ainda, mensurando sua confiabilidade. A proposta para esse projeto é a utilização de algumas técnicas de inteligência artificial e mineração de dados (serão definidas no capítulo 2) para, a partir da base de dados com informações sobre os microorganismos presentes nos pacientes, obter padrões sobre a efetividade de antibióticos e possíveis tendências para a ocorrência de um problema decorrente do microorganismo.

Como dito, existem muitas informações, exigindo um esforço computacional muito grande para seu processamento. Por isso, serão utilizados algoritmos genéticos para o aprimoramento de técnicas tradicionais de mineração de dados e o desenvolvimento de novas, de modo que sejam obtidas soluções mais rápidas e não piores que as anteriores.

1.2 Sobre os dados que serão utilizados

A base de dados consiste em informações sobre culturas de micro-organismos encontradas em amostras coletadas de pacientes de diferentes faixas etárias de todo Brasil. Também apresenta características específicas dos micro-organismos e os antibióticos utilizados para o tratamento de cada um. Os dados provenientes do Ministério da Saúde são de domínio público. São 863834 amostras com 16 atributos associados (Será melhor descrito no capítulo 4) sobre os microorganismos e os pacientes.

Vale ressaltar que os dados não possuem a identificação dos pacientes, não sendo necessário obter termo de consentimento livre e esclarecido ou submissão à comissão de ética.

1.3 Objetivos

Esse projeto visa à obtenção de modelos relevantes quanto à eficiência de antibióticos em microorganismos dado suas informações em pacientes de diferentes caracteríscas, sobre as bases de dados disponíveis. A eficiência e eficácia das técnicas utilizadas são de suma importância para o desenvolvimento, já que são esperados modelos e resultados mais rápidos, robustos e confiáveis.

Essa parte inicial do desenvolvimento consiste em um aprendizado dos conceitos fundamentais que serão as bases desse projeto e as discussões sobre as futuras aplicações.

Capítulo 2

Fundamentos Teóricos

Aqui são apresentados os conceitos fundadmentais aprendidos que tornaram possivel a realização da parte inicial do projeto.

2.1 Algorítmos Genéticos

Algorítmos Genéticos (GAs) são algorítmos de busca e otimização, que procuram obter a melhor solução para um determinado problema. Foram inventados para imitar processos observados na evolução natural [3] e seus mecanismos [7].

2.1.1 Analogias com a Genética e Seleção Natural

Um GA faz uso dos mecanismos que ocorrem na evolução para atingir seus objetivos. De maneira que o objeto de análise é uma população de indivíduos, onde cada um possui suas próprias características. Essas características são representadas por um cromossomo, que é um grupamento de genes. Com o passar do tempo, a população evolui através da reprodução para a geracão de novos indivíduos e possíveis mutações nos mesmos.

No contexto computacional, o indivíduo representa uma possível solução para um determinado problema. Portanto, a população é um conjunto de soluções. Através da sua evolução, os indivíduos considerados menos aptos tendem a desaparecer e os mais aptos sobrevivam e continuem gerando outros tão bons ou melhores.

Naturalmente, outros mecanismos também são implementados. Os indivíduos filhos herdam características dos seus genitores. Devido à selecção natural, onde as melhores características são mantidas, as gerações mais recentes se tornam melhores ou iguais à anteriores.

O ponto crucial na modelagem de um GA é definir como um indivíduo é avaliado, ou seja, mensurar a qualidade das suas características. Essa medida é chamada de *fitness*.

A execução do algorítmo começa com a inicialização de uma população finita, onde cada indivíduo tem suas características iniciadas, com valores pseudo-aleatórios - em geral - e válidos (dentro do contexto do problema). Feito isso, todos são avaliados pela função *fitness*. Nesse ponto, são selecionados indivíduos para o cruzamento (*crossover*), onde também pode ocorrer mutação. Alguns elementos da população são eliminados para dar lugar aos novos gerados. Novamente, todos são avaliados pela função *fitness*. Enquanto for requerido, a população é evoluída. Por fim, a última geração terá melhores indivíduos que as anteriores.

2.2 Codificação

Para a implementação, um cromossomo é representado por uma *string* - esturutra homogênea unidimensional -, onde cada posição é um gene. Esta pode ser binária, inteira ou real. A escolha da codificação vai depender dos tipos dos dados disponíveis.

Qualquer informação pode ser representada facilmente de maneira binária pelo computador, o que torna a representação por strings binárias tão atrativa. Porém *strings* de inteiros e reais são amplamente utilizadas [5].

2.3 Operadores genéticos

Os operadores genéticos são os mecanismos necessários para promover a evolução da população representada.

2.3.1 Seleção

Para uma nova geração, devem ser selecionados os indivíduos que para a operação de crossover. É possível usar uma abordagem psuedo-aleatória, como também, uma mais elitista, realizando o cruzamento entre os elementos que possuem melhor fitness, e consequentemente, uma boa configuração de genes - chamado de esquema.

2.3.2 Crossover

Todas as gerações na execução de um GA provém da parental por meio de cruzamentos entre esses primeiros indivíduos. Utilizando *strings*, um filho é gerado pela junção de um trecho do cromossomo pai e um trecho do cromossomo mãe (*one point crossover*). Naturalmente, esse processo pode ser feito utilizando mais fragmentos de cada genitor. Dessa maneira, o mecanismo de herança também é implementado.

2.3.3 Mutação

Um *crossover* gera um cromossomo filho com características que estavam presentes nos cromossomos pais. Logo, por mais que uma sequência inédita seja gerada, os genes em si, não serão. Ou seja, dada uma população já inicializada, somente por *crossover* - que gere filhos formados por fragmentos dos pais -, é impossível o surgimento de um gene que não esteja presente na população. De maneira que haverá pouca diversidade, levando a uma rápida homogeneidade.

A mutação consiste em pequenas perturbações nos genes de um cromossomo. Uma inversão de *bits* em uma string binária, um acréscimo ou decréscimo em um inteiro ou uma perturbação gaussiana numa representação real. Alterações bruscas causariam uma descaracterização total do cromossomo, podendo haver perda de informação.

Essas alterações geram novos cromossomos ligeiramente diferentes, que podem aumentar a precisão dos resultados e testam a robustez das respostas encontradas. Particularmente, a representação real possui maior precisão que as outras devido ao ponto flutuante.

2.4 A função Fitness

O que mais diferencia um algorítmo genético de outro é a maneira que um indivíduo é classificado, já que, depende do contexto da aplicação. A função *fitness* deve mostrar o quão bom é um indivíduo dentro do problema.

2.5 Evolução

A partir dos cruzamentos da geração parental, as seguintes são geradas. Nesse processo ocorre *crossover* e mutação para a produção dos cromossomos filhos. Então, algumas abordagens para a criação da nova geração podem ser aplicadas.

Pode ocorrer uma substituição geracional, onde a parental é completamente substituída.

Numa perspectiva elitista, um certo número de melhores indivíduos da geração passada é mantido, para que boas sequências de genes não sejam perdidas.

É possível também a não permissão de indivíduos repetidos, para a permanência da diversidade na população.

2.6 Mineração de Dados

Mineração de dados é uma área que objetiva encontrar padrões, correlações ou anomalias em grandes bancos de dados [15]. É uma parte do processo de descoberta de conhecimento em banco de dados para os dados de baixo nível em informação de alto nível. A mineração tem por objetivo extrair padrões e modelos dos dados [4].

2.6.1 CRISP-DM

CRISP-DM (do inglês, CRoss Industry Standard Process for Data Mining) é um modelo de processo que identifica as diferentes fases na implantação de um projeto de mineração de dados. Essa metodologia independe do domínio de aplicação, podendo ser aplicada nesse contexto específico pela generalidade proposta pelo modelo [11]. É uma metodologia bastante difundida, e define as atividades do processo de maneira hierárquica [2].

A fase inicial - entendimento de negócio - trata do levantando dos objetivos do projeto a partir da definição do problema abordado numa perspectiva de negócio. Em seguida, na fase de entendimento de dados, os dados são descritos, identificando pontos importantes para análise. Na fase de preparação, os dados são tratados, eliminando atributos indesejados, e selecionando os que serão analisados. Na modelagem, modelos são aplicados para calibração de parâmetros. Então, são avaliados e o modelo final é construído, desde que esteja dentro dos objetivos de negócio. Por último, a informação adquirida é organizada para apresentação para o portador final pronto para ser utilizado.

2.7 Clusterização

Clusterização (*Clustering*) é uma importante técnica de classificação não-supervisionada, onde, os dados similares pertencem a um mesmo grupo (*cluster*) [12], logo dados com características diferentes devem pertencer a *clusters* diferentes.

As técnicas de *clustering* podem ser classificadas quanto à variação do número de *clusters*. Clusterização com número de clusters não-variável exige que seja esse valor k seja dado, para que os dados sejam agrupados em k grupos. Já os não-variáveis, determinam o melhor número de *clusters* com base nas similaridades dos próprios dados [2].

2.7.1 K-means

Das técnicas de *clustering*, o algorítmo *K-means* (K-médias) é um dos mais conhecidos e usados [12]. Dado um número k de *clusters*, são eleitos k centróides, que são pontos que representam o centro de um *cluster* - em geral, escolhidos de maneira pseudo-aleatória do

banco de dados. Então, para cada ponto da amostra é calculada a distância euclidiana para os três centróides, de modo que o ponto é atribuído ao centróide mais próximo.

A distância euclidiana entre dois pontos x e y de $\mathrm{R^n}$ é dada por:

$$d = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2 + ... + (x_n - y_n)^2} \tag{2.1}$$

Depois, os centróides são reposicionados, calculando a distância média para cada ponto do seu *cluster*. Ao término dessa etapa, novamente é verificada a distância euclidiana, atribuindo os pontos aos *clusters* dos centróides mais próximos e, em seguida, reposicionando os centróides com base no novo grupo. Esse processo é repetido até que os centróides não se movam mais, ou até que essa movimentação seja pequena o suficiente.

Para se ter maior confiabilidade do resultado, não basta utilizar o algorítmo uma única vez, seria necessário utilizar essa técnica para todos os arranjos de k centróides. Tendo como resultado, os centróides resultantes de cada arranjo, para se ter o valor médio e métricas para confiabilidade.

2.8 Rigor quanto aos alogrítmos

Para aumentar a confiabilidade dos resultados na implementação dos algorítmos, técnicas de validação cruzada (cross validation) foram utilizadas para a experimentação do modelo. Também foi criado o gerador de números pseudo-aleatórios *Linear Feedback Shift Register* (LFSR) para maior confiabilidade nos operadores e na inicialização dos cromossomos.

2.8.1 Linear Feedback Shift Register

O algorítmo LFSR é um registrador de *shift* de *bits* utilizado em aplicações para geração de números pseudo-aleatórios. Dada uma sequência inicial de *bits*, é realizada a operação lógica XOR - chamada ”ou exclusivo” - entre algumas posições escolhidas da sequência. Depois, cada *bit* avança uma posição para dar lugar ao novo gerado pelas operações.

2.8.2 Cross-Validadtion

cross-validation é uma técnica estatística que permite mensurar a qualidade de um modelo a partir de uma base dados, permite avaliar a robustez do modelo para dados que não se encontram na base [9].

Para aferir a capacidade de generalização do modelo, a base de dados é divida em dados de treino e dados de teste. Esses grupos devem ser representativos da população e mutuamente exclusivos. Isso garante que a modelagem seja exclusiva daquele conjunto de dados, evitando redundâncias (*overfitting*).

Porém, uma simples divisão não garante bons subconjuntos de dados, já que pode depender das particularidades de cada divisão. Existem meios mais efetivos de divisão que minimizam esse problema. O método *K-fold* separa a base de dados em k subconjuntos, dos quais k - *1* são usados para treino e 1 para teste. Cada subconjunto de teste é avaliado pelo modelo, e aplicado no subconjunto de teste. O resutaldo final pode ser obtido como a média entre as avaliações.

Um caso particular do *K-fold*, conhecido como *Leave one out*, utiliza k com valor igual ao total de dados, ou seja, as k - *1* parametrizações são testadas em um único ponto.

Capítulo 3

Ferramentas Utilizadas

Nesse capítulo são apresentadas as ferramentas que serão utilizadas para o desenvolvimento do projeto e a descrição dos primeiros algorítmos implementados.

3.1 Linguagem Python

Os algorítmos foram implementados na linguagem *Python* 3.4.3. Foi escolhida pelo suporte disponível pelo *framework* DEAP para manipulação de algorítmos genéticos, aumentando a rapidez no desenvolvimento do projeto, já que contém os operadores básicos dos GAs. É uma linguagem interpretada, multi-paradigma com alta legibilidade, sendo muito utilizada para o ensino.

3.1.1 Distributed Evolutionary Algorithms in Python (DEAP)

O *framework* Distributed Evolutionary Algorithms in *Python* (DEAP) proporciona uma prototipação rápida de GAs, com estruturas de dados transparentes e algorítmos explícitos [6].

Seus módulos disponibilizam um grande aparato de ferramentas para a construção de um algorítmo genético. Fazendo com que o programador se preocupe mais com a modelagem do problema.

3.1.2 scikit-learn

Scikit-learn é uma biblioteca de código aberto para programação em linguagem *Python* [13]. Faz uso de outras bibliotecas matemáticas e científicas (por exemplo *NumPy* e *SciPy*). Dá suporte para clusterização, classificação e regressão, o que a torna interessante para aplicação nesse projeto.

Capítulo 4

Dados Utilizados

Para testes nos algorítmos, foram selecionados dois bancos de dados conehcidos mais simples e menores que os dados sobre microorganismos.

4.1 Iris

O banco de dados Iris [10] consiste em informações sobre três espécies diferentes de flor de íris. Contém 150 amostras cada uma com medidas do comprimento e largura médias de suas sépalas e comprimento e largura médias de suas pétalas, em centímetros. Foi o primeiro banco utilizado por ser muito pequeno de modo a facilitar a visualização do funcionamento dos algorítmos.

4.2 Dados sobre microorganismos

A base proveniente do Ministério da Saúde é composta por 863834 dados, com 16 atributos cada uma. Foram recolhidas de homens e mulheres, de todo o Brasil entre 0 e 100 anos de idade. O material biológico foi colhido entre 2008 e 2015 e têm grande variedade.

Os microorganismos são descritos por sua classe, família, gênero e espécie. É citada sua característica fenotípica e morfo-tintorial - padrão de coloração apresentado em sua identificação -, o antibiótico utilizado e se apresentou ou não resistência.

Capítulo 5

Metodologia Científica

Dada a dificuldade de se trabalhar com grandes bancos de dados (este caso, por exemplo), é importante que o máximo de informações relevantes possam ser extraídas com confiabilidade dentro de um espaço de tempo plausível. O objetivo dessa fase inicial é criar um primeiro modelo híbrido de *clustering* com algorítmos genéticos que atenda a esses princípios.

No primeiro momento, foram estudados os conceitos e algorítmos fundamentais de GAs. Em seguida foi estudado e implementado o algorítmo clássico *K-means*[1] seguindo a descrição do algorítmo clássico encontrado na literatura [1], entendendo seu funcionamento e como seria possível melhorá-lo usando GAs. A última etapa consiste em efetivamente elaborar o algorítmo híbrido, que é descrito na próxima seção.

Espera-se encontrar, comparando o algorítmo clássico e o híbrido, resultados mais consistentes do segundo se ambos forem executados apenas uma vez. Apesar da execução mais rápida do primeiro, para resultados confiáveis são necessárias diversas execuções.

Para comprovar a hipótese, será utilizado o banco de dados iris, para se ter resultados mais rápidos.

[1] `https://github.com/IcaroMarcelino/Testes_Deap`

Capítulo 6

Hibridização

Na elaboração das primeiras estratégias para a solução do problema levantado para esse projeto, foi proposta a hibridização do algorítmo *K-means* com algorítmos genéticos. Uma vez que os GAs conseguem otimizar uma solução inicial válida que não é necessariamente boa. E com o avançar das gerações o resultado tende ao seu ótimo [12].

O cromossomo (possível solução do problema) é um conjunto de k centróides reais de dimensão n. Para sua representação, foi escolhida uma representação por *string* real de tamanho nk, onde as primeiras n posições correspondem às coordenadas no primeiro centróide, as n seguintes ao segundo e assim sucessivamente.

Os indivíduos são inicializados de maneira aleatória. Cada gene do cromossomo é um real pseudo-aleatório entre o valor menor valor encontrado na base de dados e o maior deles acrescido da média entre os dois.

O *crossover* fragmenta os genitores em três partes cada, gerando dois filhos formados pela combinação dessas partes, de modo que suas posições são mantidas dentro dos cromossomos gerados. Nessa proposta, não ocorre mutação durante essa fase. O cruzamento ocorre entre dois indivíduos vizinhos, começando pelo primeiro da população. Porém esse processo só ocorre dentro de uma determinada probabilidade pré-definida. Um mesmo indivíduo não participa dessa operação duas vezes numa mesma geração.

Com a prole formada, todos os indivíduos têm chance de sofrer mutação. Os selecionados sofrem pequenas perturbações (dentro de uma distribuição gaussiana) em seus genes, com o objetivo de aumentar a diversidade populacional. Esta substitui completamente a geração anterior.

O *fitness* dos indivíduos é dado pela soma das distâncias euclidianas dos pontos para seus respectivos *clusters*. Logo, um bom conjunto de centróides minimiza esse valor, ou seja, um melhor cromossomo é aquele que possui o menor fitness.

Para a formação da nova geração, são mantidos os dois melhores cromossomos da geração anterior. Essa abordagem elitista garante que o melhor indivíduo da próxima

geração será não pior que na anterior.

A cada geração, os indivíduos são avaliados, sofrem *crossover* e mutação. Isso ocorre até que o número de gerações desejado seja atingido, ou até que um determinado grau de homogeneidade seja atingido - o ponto onde a população perde sua diversidade.

Antes da execução devem ser definidos o número total de gerações, a probabilidade de um indivíduo sofrer mutação, probabilidade de sucesso de uma operação de *crossover* e o tamanho da população.

Capítulo 7

Análise dos Resultados

Para a verificação do modelo, os dados foram divididos em treino e teste (70% para treino e 30% para teste). O algorítmo foi parametrizado com 75% de chance de *crossover* e 10% de chance de mutação, para 100, 200 e 500 gerações de 100 indivíduos e 100 gerações de 250 indivíduos.

Para aumentar a diversidade, à cada geração, a chance de *crossover* decresce em 0.1% - até atingir 60% - e a chance de mutação aumenta na mesma medida - até atingir 30%.

Nesse experimento, foram feitas três execuções consecutivas de cada algorítmo e, na última geração, foi selecionado o conjunto de centróides que possuía o menor *fitness*.

Nas três execuções do *K-means* clássico, as coordenadas tiveram os resultados mais discrepantes. Como dito anteriormente, sendo necessário avaliar os arranjos possíveis de centróides para se ter o valor médio. Já nas execuções do algorítmo híbrido[1] as discrepâncias diminuem com o aumento do número de gerações.

Enquanto houver tempo, o número de gerações pode ser aumentado para resultados mais precisos e menos discrepantes - entre duas execuções diferentes. Quando o número de gerações tende à infinito, a resposta tende ao seu melhor resultado [12]. Nesse contexto, a métrica de tempo deve ser o comparativo do que seria necessário para realizar o procedimento com *K-means* com o que é viável para decidir o melhor número de gerações. O aumento do número de indivíduos introduz mais diversidade de genes, porém, aumenta o tempo de execução.

As tabelas a seguir mostram os resultados obtidos com o algorítmo clássico e híbrido em três execuções independentes e consecutivas.

[1] `https://github.com/IcaroMarcelino/Testes_SciKitLearn`

Tabela 7.1: Execução do algorítmo K-means sobre o banco de dados Iris.

Execução	**Centróide 1**	**Centróide 2**	**Centróide 3**
1	(7.70, 2.80, 6.50, 2.30)	(6.45, 3.25, 5.05, 1.90)	(6.00, 2.90, 4.90, 1.50)
2	(6.30, 3.30, 5.35, 2.30)	(6.15, 3.05, 5.30, 1.80)	(5.05, 3.00, 2.15, 0.70)
3	(6.95, 3.00, 6.15, 2.30)	(5.70, 2.70, 4.45, 1.45)	(5.35, 3.85, 1.45, 0.30)

Tabela 7.2: Execução do algorítmo K-means Híbrido sobre o banco de dados Iris - 100 gerações, 100 indivíduos.

Execução	**Centróide 1**	**Centróide 2**	**Centróide 3**
1	(6.79, 3.04, 5.52, 2.14)	(5.76, 2.68, 4.43, 1.31)	(5.14, 3.58, 1.60, 0.28)
2	(6.59, 3.10, 5.38, 1.98)	(5.77, 2.69, 4.32, 1.48)	(5.26, 3.40, 1.44, 0.35)
3	(6.64, 2.92, 5.11, 1.65)	(5.76, 2.60, 3.64, 1.23)	(4.88, 3.52, 1.39, 0.28)

Tabela 7.3: Execução do algorítmo K-means Híbrido sobre o banco de dados Iris - 200 gerações, 100 indivíduos.

Execução	**Centróide 1**	**Centróide 2**	**Centróide 3**
1	(6.57, 3.08, 5.54, 1.96)	(5.63, 2.67, 4.09, 1.43)	(4.97, 3.38, 1.44, 0.16)
2	(6.55, 3.01, 5.20, 1.75)	(5.85, 2.58, 3.97, 0.71)	(4.98, 3.50, 1.37, 0.16)
3	(6.48, 2.99, 5.17, 1.89)	(5.62, 2.57, 4.12, 1.18)	(4.94, 3.34, 1.36, 0.27)

Tabela 7.4: Execução do algorítmo K-means Híbrido sobre o banco de dados Iris - 500 gerações, 100 indivíduos.

Execução	**Centróide 1**	**Centróide 2**	**Centróide 3**
1	(6.75, 3.16, 5.65, 2.07)	(5.98, 2.79, 4.35, 1.58)	(4.93, 3.34, 1.49, 0.23)
2	(6.76, 3.07, 5.58, 2.06)	(5.85, 2.74, 4.25, 1.29)	(5.03, 3.33, 1.40, 0.18)
3	(6.67, 3.07, 5.58, 2.07)	(5.73, 2.92, 4.28, 1.34)	(5.07, 3.34, 1.54, 0.24)

Tabela 7.5: Execução do algorítmo K-means Híbrido sobre o banco de dados Iris - 100 gerações, 250 indivíduos.

Execução	**Centróide 1**	**Centróide 2**	**Centróide 3**
1	(6.74, 3.14, 5.64, 1.92)	(5.73, 2.68, 4.13, 1.30)	(4.98, 3.31, 1.48, 0.18)
2	(6.76, 3.07, 5.58, 2.06)	(5.85, 2.74, 4.25, 1.29)	(5.03, 3.33, 1.40, 0.18)
3	(6.77, 3.11, 5.48, 1.87)	(5.73, 2.82, 4.23, 1.29)	(4.82, 3.60, 1.60, 0.29)

Capítulo 8

Clusterização de Dados Biomédicos com Técnicas de Algoritmos Genéticos

Esse projeto apresentou a parte inicial de um estudo novo sobre aplicação de algorítmos genéticos e técnicas de *clustering* para dados biomédicos. Essa etapa de aprendizado pela falta de familiaridade com a área foi responsável por estabelecer conceitos sólidos que serão fundamentais para as próximas etapas.

Os algorítmos genéticos se mostraram uma ferramenta poderosa para os problemas de otimização e busca heurística, e podem melhorar o desempenho das técnicas de *clustering*. O custo computacional aumenta com a complexidade e tamanho das bases de dados. Portanto, os estudos na hibridização terão continuidade para o aprimoramento dessas técnicas e para atender aos objetivos estabelecidos para o projeto como um todo.

8.1 Trabalhos Futuros

Para a próxima etapa, as técnicas aprendidas serão utilizadas no desenvolvimento de novos modelos a partir da base dados. Será utilizada a abordagem CRISP-DM para a mineração de dados de maneira mais sistemática. As funções de aleatoriedade utilizadas até então serão substituídas pelo algorítmo LFSR implementado. E, também, técnicas mais rigorosas *cross validation* validarão os próximos modelos.

Os próximos algorítmos serão usados na própria base dados, para que já possam ser extraídas informações para que sejam comparadas futuramente.

Será necessário um estudo do domínio de aplicação, para o entendimento mínimo necessário para a interpretação correta das informações extraídas da base de dados.

É necessário que eficiência e eficácia dos algorítmos sejam atingidas, assim como resultados confiáveis e modelos robustos.

Com a finalização do projeto, apresentar os resultados no Congresso Anual de Iniciação Científica e submeter artigos à congressos internacionais.

O mais desejado é que as técnicas e resultados obtidos possam ser úteis à população, encontrando novos meios de prevenção e tratamento para as doenças causadas pelos microorganismos estudados. E que a continuação desse projeto possa ser incentivada.

8.2 Cronograma para o próximo semestre

Planejamento das atividades por mês:

• Meses 1, 2 e 3: Estudo do domínio de aplicação, proposição, construção e avaliação de técnicas baseados em algoritmos genéticos para a identificação de *clusters* na base de dados.

• Meses 3 e 4 - Gerar documentação sobre o desenvolvimento da pesquisa, a ser defendida perante uma banca de pesquisadores em Inteligência Artificial dos Departamentos de Ciência da Computação da UnB e da Universidade de Tsukuba, Japão.

• Meses 5 e 6 - Preparar o relatório final, resumo e pôster do projeto visando à participação no Congresso Anual de Iniciação Científica e submeter artigos à congressos internacionais.

Referências

[1] Charu C Aggarwal e Chandan K Reddy. *Data clustering: algorithms and applications*. CRC Press, 2013. 12

[2] Pete Chapman, Julian Clinton, Randy Kerber, Thomas Khabaza, Thomas Reinartz, Colin Shearer, e Rudiger Wirth. Crisp-dm 1.0 step-by-step data mining guide. Technical report, The CRISP-DM consortium, 2000. 7

[3] Lawrence Davis. *Handbook of Genetic Algorithms*. Van Nostrand Reeinhold, 115 Fifth Avenue, New York 10003, 1st edition, 1991. 4

[4] Maria Madalena Dias. Parâmetros na escolha de técnicas e ferramentas de mineração de dados. *Acta Scientiarum. Technology*, 24:1715–1725, 2008. 7

[5] Alex A. Freitas Eduardo R. Hruschka, Ricardo J. G. B. Campello e André C. P. L. F. de Carvalho. A survey of evolutionary algorithms for clustering. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, Vol. 39, Issue 2:133–155, 2009. 5

[6] Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, e Christian Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012. 10

[7] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1989. 4

[8] Denise Oliveira Guimarães, Luciano da Silva Momesso, Mônica Tallarico Pupo, et al. Antibióticos: importância terapêutica e perspectivas para a descoberta e desenvolvimento de novos agentes. *Quim. Nova*, 33(3):667–679, 2010. 1

[9] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145, 1995. 8

[10] M. Lichman. UCI machine learning repository, 2013. 11

[11] Emerson Lopes Machado e Marcelo Ladeira. Um estudo de limpeza em base de dados desbalanceada e com sobreposição de classes. In *XXVII Congresso da Sociedade Brasileira de Computação*, pages 330–340, 2007. 7

[12] Ujjwal Maulik e Sanghamitra Bandyopadhyay. Genetic algorithm-based clustering technique. *Pattern Recognition*, Vol. 33, Issue 9:1455–1465, September 2000. 7, 13, 15

[13] 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. Pedregosa, F.; Varoquaux e E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. 10

[14] Eliane RibeiroIII. Eventos adversos a antibióticos em pacientes internados em um hospital universitário. *Rev Saúde Pública*, 41(6):1042–8, 2007. 2

[15] Malik Magdon-Ismail Yaser S. Abu-Mostafa e Hsuan-Tien Lin. *Learning From Data - A Short Course*. AMLBook, 2012. 7