UNIVERSIDADE FEDERAL DA PARAÍBA
Centro de Ciências Exatas e da Natureza
Departamento de Matemática
Mestrado Profissional em Matemática em Rede Nacional

Uso de Algoritmo genético no ajuste linear através de dados experimentais

por

Erinaldo Leite Siqueira Júnior

2015

UNIVERSIDADE FEDERAL DA PARAÍBA
Centro de Ciências Exatas e da Natureza
Departamento de Matemática
Mestrado Profissional em Matemática em Rede Nacional

Uso de Algoritmo genético no ajuste linear através de dados experimentais

†

por

Erinaldo Leite Siqueira Júnior

sob orientação de

Prof. Dr. Napoleon Caro Tuesta

Projeto de Trabalho de Conclusão de Curso do Curso de Pós-Graduação em Matemática em rede Nacional - PROFMAT - DM - CCEN - UFPB, como requisito parcial para obtenção do título de Mestre em Matemática.

Maio/2015

João Pessoa - PB

† Este trabalho contou com apoio financeiro da Capes.

UNIVERSIDADE FEDERAL DA PARAÍBA
Centro de Ciências Exatas e da Natureza
Departamento de Matemática
Mestrado Profissional em Matemática em Rede Nacional

Uso de Algoritmo genético no ajuste linear através de dados experimentais

Erinaldo Leite Siqueira Júnior

Trabalho de Conclusão de Curso do Curso de Pós-Graduação em Matemática em rede Nacional - PROFMAT - DM - CCEN - UFPB, como requisito parcial para obtenção do título de Mestre em Matemática.

Orientador:

Prof. Dr.Napoleon Caro Tuesta
Universidade Federal da Paraíba

Banca Examinadora:

Prof. Dr. Lizandro Sanchez Challapa
Universidade Federal da Paraíba

Prof. Dr. Kleber Napoleão N. O. Barros
Universidade Estadual de Campina Grande

Sumário

Lista de Figuras

Lista de Tabelas

Resumo

Neste trabalho abordaremos o problema de ajuste linear para dados experimentais através de um método de otimização bio-inspirado, isto é, que mimetiza conceitos biológicos na tentativa de buscar resultados ótimos ou sub-ótimos. O método utilizado é o algoritmo genético (AG), AG faz uso da teoria da evolução Darwiniana para buscar a melhor rota para o ponto de máximo desejado. Tradicionalmente, o ajuste linear é feito através do método de mínimos quadrados. Tal método é eficiente, porém é de difícil justificativa para as turmas pré-cálculo. Diante disso, a alternativa do AG vem como um procedimento exaustivo computacionalmente, entretanto de fácil justificativa para essas turmas. Assim, a proposta do trabalho é comparar os resultados de ajuste linear para alguns cenários de controle através dos dois métodos e certificar a qualidade dos ajustes obtidos pelo método aproximado. No final do trabalho constatou-se que os resultados encontrados sÃ£o sólidos o bastante para justificar o método alternativo e que a proposta da utilização desse processo de otimização tem potencial para despertar interesse em outras áreas da matemática.

Palavras-chave:Algoritmo Genético, Otimização, Ajuste Linear, Mínimos Quadrados, Computação natural.

Abstract

In this paper we discuss the problem of linear fitting to experimental data using a method bio-inspired of optimization, i.e., it imitates the biological concepts attempt to find optimal or suboptimal results. The method used is the genetic algorithm (GA), AG makes use of the theory of Darwinian evolution to find the best route for the desired maximum point. Traditionally, the linear fitting is made through the method of least squares. The method is efficient, but is difficult to justify the pre-calculus classes. Therefore, the alternative AG comes as a computationally exhaustive procedure, however easy justification for these classes. Thus, the purpose of this study is to compare the results of linear fitting for some control scenarios using this methods and certify the quality of the adjustments obtained by the approximate method. At the end of the work it was found that the results are solid enough to justify the alternative method and the proposed use of this optimization process has the potential to spark interest in other areas of mathematics.

Keywords: Genetic Algorithm Optimization, Adjust Linear Least Squares, Natural Computing.

1 Introdução e Motivação

A regressão linear é um método para estimar a variável Y (em geral a resposta para o experimento feito) em função das observações sob a variável X. Em geral, consiste de obter uma relação funcional entre as variáveis dependentes e independentes, e através disso, ajustar os valores prováveis para a variável Y dentro do intervalo observado [9].

Para obtermos a reta ajustada aos dados é possível apresentar o gráfico da dispersão dos pontos observados (x, y) em que x é a observação trabalhada e y é a variável resposta obtida, conforme figura 1.

É fácil perceber que qualquer reta ajustada ao diagrama não poderá conter todos os pontos do diagrama, isso se deve em sua grande maioria a erros de medição do experimento, ou casualidades inerentes ao experimento o que assegura que o evento estudado não é um fenômeno matemático. A partir disso, é necessário que seja utilizado metodologias que otimizem a reta ajustada com a finalidade de que as distâncias de qualquer ponto a reta seja mínima.

Uma alternativa para isso é a utilização do método de mínimos quadrados que consiste numa técnica de otimização matemática que procura encontrar o melhor ajuste para um conjunto de dados tentando minimizar a soma dos quadrados das diferenças entre o valor estimado e os dados observados (chamadas geralmente de resíduos) [31]:

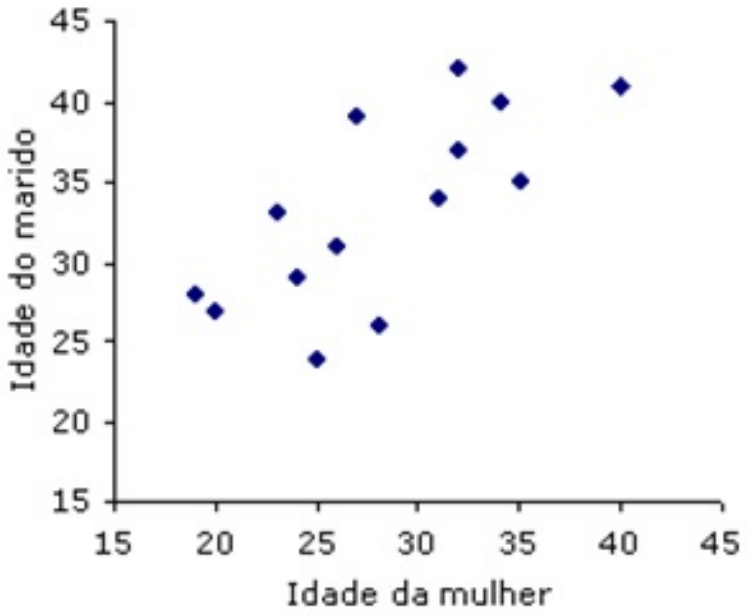


Figura 1: Diagrama de dispersão para idade de homens e mulheres [20].

Seja Y a variável resposta para o modelo definido por:

$$Y = \alpha + \beta \cdot X + \epsilon \tag{1.1}$$

em que:

α é o parâmetro do modelo chamado fator linear;

β é o parâmetro do modelo chamado fator de inclinação da variável X;

ϵ é o erro, ou variação de Y que não pode ser explicada pelo modelo.

Assim, ao estimar o modelo usando as observações teremos a representação:

$$y_i = \alpha + \beta \cdot x_i + \epsilon_i \tag{1.2}$$

sendo i uma das n observações da base de dados, $i \in 1, 2, \ldots, n$. Para o método de mínimos quadrados minimizamos o resíduo quadrado:

$$S(a,b) = \sum_{i=1}^{n} e_1^2 = \sum_{i=1}^{n} (y_i - bx_i - a)^2 \tag{1.3}$$

Para minimizar, calcula-se a derivada parcial com relação a a e b e igualando a zero:

$$\frac{\partial S(a,b)}{\partial a} = -2 \sum_{i=1}^{n} (y_i - bx_i - a) = 0 \tag{1.4}$$

$$\frac{\partial S(a,b)}{\partial b} = -2 \sum_{i=1}^{n} x_i (y_i - bx_i - a) = 0 \tag{1.5}$$

Distribuindo e dividindo a expressão (1.4) por 2n:

$$\frac{\frac{\partial S(a,b)}{\partial a}}{2n} = -\frac{\sum_{i=1}^{n} (y_i - bx_i - a)}{n} = 0 \tag{1.6}$$

$$-\overline{y} + b\overline{x} + a = 0 \tag{1.7}$$

$$a = \overline{y} - b\bar{x} \tag{1.8}$$

Com $\bar{y}$ e $\bar{x}$ sendo as médias amostrais de y e x respectivamente. Substituindo o resultado na expressão 1.5:

$$-2 \sum_{i=1}^{n} x_i (y_i - bx_i - a) = 0 \tag{1.9}$$

$$-2 \sum_{i=1}^{n} x_i (y_i - bx_i - \overline{y} + b\overline{x}) = -2 \sum_{i=1}^{n} x_i (y_i - \overline{y} + b\overline{x} - bx_i) = 0 \tag{1.10}$$

$$\sum_{i=1}^{n}[x_i(y_i-\overline{y})+bx_i(\overline{x}-x_i)]=\sum_{i=1}^{n}x_i(y_i-\overline{y})+b\sum_{i=1}^{n}x_i(\overline{x}-x_i)=0 \tag{1.11}$$

$$-b\sum_{i=1}^{n}x_i(x_i-\overline{x})=-\sum_{i=1}^{n}x_i(y_i-\overline{y}) \tag{1.12}$$

$$b=\frac{\sum_{i=1}^{n}x_i(y_i-\overline{y})}{\sum_{i=1}^{n}x_i(x_i-\overline{x})} \tag{1.13}$$

em que (1.8) e (1.13) são as estimações para α e β para o ajuste da reta.

O grande problema em se trabalhar com o método de mínimos quadrados nas escolas básicas no momento em que é ensinado tratamento da informação / estatística básica está no fato de apenas apresentar os estimadores a e b sem apresentar o desenvolvimento do raciocínio que remete a cálculo diferencial, em especial de várias variáveis, tema esse que não é abordado no currículo médio.

Uma alternativa a essa metodologia é recorrer a processos de otimização computacional para o ajuste dos parâmetros devido a facilidade de compreensão da ideia da recorrência por parte do alunado, mesmo que não seja possível atingir o nível de explicitar a programação envolvida. Um método particularmente interessante é o algoritmo genético que consiste num processo de busca que tem embasamento em evolução genética (tema trabalhado em sala de aula por outras disciplinas) facilitando a compreensão do método e a aceitação em sala [5].

O objetivo do trabalho é comparar os resultados obtidos através de bancos de dados entre o método de mínimos quadrados e o método de algoritmo genético evidenciando as potencialidades e fraquezas desse método alternativo.

2 Algoritmo Genético

2.1 Computação evolutiva

É possível definir a computação evolutiva como uma área de pesquisa da modelagem computacional que tem como principal ponto de inspiração o processo de evolução natural. Notadamente, é possível perceber como colônias de formigas, a própria evolução genética, enxame de abelhas entre outros apresentam formas características de eliminar as tentativas menos práticas ou danosas e repetir/melhorar na população os casos em que houve mais sucesso ou que ao menos não levaram a perdas ([24]; [1]).

Tal filosofia é usada como metodologia para resolução de problemas que podem ser transcritos como um dilema evolucionário, com base nisso e se apropriando da observação de como a natureza soluciona tais dilemas, a computação evolucionária surge como uma forma de resolução de problemas baseado em tentativa-e-erro (direcionado) ([14]).

Mas como definir a evolução natural e sua interpretação em computação evolutiva? Tome um cenário ambiental em que neste vive uma população que tem como objetivo sobreviver e se reproduzir. A adaptabilidade desses indivíduos será governada pelas características do cenário no qual a população está inserida, essa adaptabilidade indica o quanto eles conseguiram seu objetivo (sobreviver e se multiplicar). Sob o ponto de vista de resolução de problema por tentativa-e-erro, a população servirá como uma coleção de possíveis soluções para o problema, como são soluções, é possível medir o quanto essas possíveis soluções respondem ao problema

(e quão boa é a solução, e tal critério será utilizado para definir as chances dessa possível solução permanecer na coleção e servir de parâmetro para soluções futuras).

Evolução	Resolução do Problema
Ambiente	Problema
Indivíduo	Possível solução
Ajuste	Qualidade

Figura 2: Ilustração do comparativo entre computação evolutiva e a evolução natural em suas abordagens para solucionar seus problemas. [14]

2.2 Histórico computação evolutiva

A utilização dos princípios apresentados por Charles Darwin na tentativa de resolução de problemas de forma autônoma teve início na década de 40, bem antes da disseminação de computadores; David B. Fogel em seu livro *Evolutionary Computation: The Fossil Record* ([17]) apresenta os artigos que fundamentam a computação evolutiva. Em 1948, Turing propôs o processo de busca intitulado "Busca genética" ([38]) e em 1962, Bremermann ([8]) executou experimentos em otimização através de evolução e recombinação.

Variações da metodologia ocorreram na década de 60 em três locais distintos:
- Nos Estados Unidos, nos anos de 1965 e 1966, os primeiros artigos sobre programação evolutiva foram publicados por Fogel, Owens e Walsh. A programação evolucionária foi inventada por Lawrence J. Fogel (1928-2007) enquanto fazia parte da Fundação da ciência nacional em 1960, em sua variação, Fogel trabalhava em uma série de experimentos em que as máquinas de estados finitos representam organismos individuais em uma população de solucionadores de problemas. Estes modelos gráficos são usados para descrever o comportamento, é por isso que chamou sua abordagem de programação evolutiva ([18];[19]).

- Na Alemanha, Rechenberg e Schwefel na Universidade Técnica de Berlin apresentaram a estratégia evolutiva para resolução de problemas de otimização contínua de parâmetros de controle e são descritas de forma completa no trabalho *Evolutionstrategie: Optimierung Technisher Systeme nach Prinzipien des Biologischen Evolution* de 1973; a grande mudança apresentada nessa técnica é dada pela utilização de um operador mutação baseado em uma distribuição Normal ([34]).

- Novamente nos Estados Unidos, John Henry Holland denominou seu método por algoritmo genético, método este que é usado para achar soluções aproximadas em problemas de otimização e busca. Os algoritmos genéticos diferem dos algoritmos tradicionais de otimização em quatro aspectos ([13];[22];[23]):

- Baseiam-se em uma codificação do conjunto das soluções possíveis, e não nos parâmetros da otimização em sí;
- Os resultados são apresentados como uma população de soluções e não como uma solução única;
- Usam transições probabilísticas e não determinísticas;
- Não necessitam de nenhum conhecimento inerente ao problema, apenas de uma forma de avaliação dos resultados (particularmente, o aspecto mais interessante do método). Pois, uma vez que é retirada a responsabilidade de se conhecer a singularidade da resolução de um problema, o algoritmo está apto a utilizar o procedimento em problemas de natureza aleatória ou que se tenha pouco conhecimento acerca de sua dinâmica. ([21])

Por cerca de 15 anos essas áreas foram desenvolvidas separadamente, porém, na década de 90 passaram a ser vistos como representações diferentes de uma mesma tecnologia que ficou conhecida como computação evolutiva. Uma quarta corrente seguindo as idéias gerais surgiu, programação genética, defendida por Koza ([26]; [27]). Na programação genética, consiste numa técnica automática de

programação que propicia a evolução de programas de computadores para resolverem (ou aproximar a resolução) de problema.

A terminologia contemporânea denota todos os campos da computação evolutiva, os algoritmos contidos nessa terminologia são chamados de algoritmos evolutivos, e esta classificação considera a programação evolutiva, estratégia evolutiva, algoritmo genético e programação genetica como sub-áreas pertencentes a variantes deste.

Uma vertente mais recente é a computação de enxames, o algoritmo conhecido como Otimização por Enxame de Partículas (PSO - Particle Swarm Optimization). O algoritmo PSO foi proposto inicialmente por J. Kennedy e R. Eberhart (Kennedy, 1995) e é um técnica inspirada no comportamento social de bandos de pássaros. A busca por alimento e a interação entre os pássaros ao longo do vôo são modeladas como um mecanismo de otimização. No caso, a área sobrevoada é equivalente ao espaço de busca e encontrar o local com maior quantidade de comida corresponde a encontrar a solução ótima.

2.3 Contexto biológico

Pode-se dizer que a teoria da evolução tem participação fundamental na compreensão deste capítulo (e com isso, compreensão dos algoritmos genéticos), daí, é necessário conhecer algumas definições:

• Um *gene* é uma sequência de bases de DNA que codificam para uma característica, por exemplo, cor de cabelo ou da pele;

• Um *alelo* é o valor de uma característica. O gene da cor do cabelo pode ter um alelo preto ou um alelo vermelho em diferentes pessoas.

Com isso, é possível definir a evolução:

Evolução é a variação da frequência alélica na população ao longo do tempo ([37]).

Nessa definição, conceitua-se sobre a população. Ou seja, toda vez que nasce uma criança de cabelos pretos, então a frequência da população quanto ao gene cor de cabelo tem a frequência do alelo preto aumentada (analogamente, para o caso de morte de um homem de cabelos pretos), daí, é possível notar que a frequência alélica muda a todo tempo, já que nascem e morrem pessoas a todo instante. Enfim, nota-se que a evolução acontece a todo tempo (podemos definir o parâmetro t discreto sob esse caso já que o evento nascimento/morte pode ser discretizado até um passo de tempo em que nasce/morre um único indivíduo).

Como a evolução produz novos indivíduos? Há duas formas opostas que conduzem a evolução: variação e seleção. Variação é o processo que produz novos alelos e, mais lentamente, genes. Variação também pode mudar os genes que são e não são expressos em um dado indivíduo, o método de fazer isso é a reprodução sexual com a sua interação de genes dominantes e recessivos. Seleção é o processo em que alguns alelos sobrevivem e outros não. Em resumo, a variação aumenta a diversidade e a seleção a diminui.

O processo de variação é complexo e ocorre a nível molecular. Biólogos aprendem novos sistemas para gerar variação a nível molecular. Seleção biológica é melhor entendida do que a variação biológica. Por outro lado, a seleção natural - sobrevivência dos mais adaptados ao meio em que vivem - tem sido o principal tipo de seleção biológica ([12]; [32]).

Na computação evolutiva, as operações sobre a estrutura de dados (população) que geram a variação recebem os nomes de *mutação* e *cruzamento* e atuam por mudanças aleatórias nessa estrutura e misturando partes de estruturas diferentes, respectivamente. Já a seleção, pode ser interpretada como qualquer algoritmo que favoreça estruturas com maior adaptabilidade ao nosso problema. Há muitos métodos possíveis de seleção.

Mutações biológicas no DNA de algum organismo, sob um ponto de vista aleatório, são tipicamente neutras. Muitos trechos do DNA não codificam informação útil (conhecida). A parte do DNA que codifica informação útil utiliza

uma codificação robusta o que assegura que modificações de base simples não alteram a funcionalidade do gene. A rede de interação entre os genes é, por sí só, robusta com multiplas cópias de alguns genes e múltiplos genes que executam uma mesma tarefa específica ([30]).

Será usado esse referecial para abordar os componentes de um algoritmo evolutivo.

2.4 Componentes de um algoritmo evolutivo

Um algoritmo evolutivo apresenta componentes bem característicos que devem ser bem definidos na elaboração da codificação do problema que será otimizado, são estes:

- Representação - Definição dos indivíduos;
- Função de avaliação (Função de ajuste ou Função de fitness);
- População;
- Mecanismo de seleção dos reprodutores;
- Operadores de variação: mutação e recombinação;
- Mecanismo de seleção dos sobreviventes (Substituição).

nesse passo, será discutido cada um desses componentes na tentativa de identificar cada particularidade associada a eles.

2.4.1 Representação

Essa é a etapa em que se deve configurar a conexão entre o cenário do problema e o espaço de solução do problema. Os objetos que formam possíveis soluções no contexto original do problema são referenciados como *fenótipos*, enquanto que as

estruturas de dados, ou seja, indivíduos de um algoritmo evolutivo são chamados de *genótipos*.

Inicia-se definindo o mapeamento que relaciona os fenótipos (possíveis soluções) com os genótipos, de forma que cada genótipo corresponde a um fenótipo no cenário do problema. O espaço-genótipo pode ser completamente diferente do espaço-fenótipo, por exemplo, o problema em questão pode ser achar o ponto de máximo absoluto de uma função real de variáveis reais; como é uma função real, seu conjunto de fenótipo tem representação decimal e alguém pode codificar seus elementos genótipos em representação binária (completamente distinto do espaço-fenótipo), assim, toda a busca por soluções ótimas se dá no espaço genótipo, isto é, no espaço de representação binária.

Sempre que se encontra uma solução ótima (ou sub-ótima) no espaço-genótipo, é possível obter a solução associada no espaço-fenótipo através da decodificação da estrutura de dados ao final do processo de otimização.

A terminologia adotada para computação evolutiva denomina por: possível solução, fenótipo ou indivíduo os pontos do espaço de possíveis soluções, o próprio espaço de possíveis soluções é chamado de espaço-fenótipo. Já o genótipo, cromossomos e novamente indivíduo podem ser usados para pontos onde a busca evolucionária ocorre (espaço-genótipo).

Há também muitos termos relacionando os elementos de um indivíduo, cada componente do indivíduo é chamado de variável, locus, posição ou gene. Um objeto em tal lugar pode ser chamado de valor ou um alelo.

O trabalho de Nicol N. Schraudolph intitulado *Dynamic parameter encoding for genetic algorithms* apresenta uma interessante codificação dinâmica para evitar o dilema de sacrificar a precisão da representação numérica ou a eficiência na busca por uma solução ótima quando se trabalha com otimização de funções de valor real ([35]).

2.4.2 Função de avaliação (Função de fitness)

A função de avaliação tem o papel de apresentar os requisitos para a adaptação, ela é o parâmetro para a seleção dos genótipos pais que darão origem a novos representantes mais adaptados ou não. Através da função de avaliação é possível medir o quanto um genótipo está resolvendo o problema associado, sua medida de qualidade está associada ao espaço dos genótipos.

Em geral, a função de fitness é qualquer função que capta a qualidade nos genótipos, qualidades estas que o tornam mais próximos da solução ótima ou subótima para o problema. Porém, há casos em que a função de fitness pode vir a ser a própria função associada ao contexto do problema ou uma simples tranformação desta.

No artigo *The Effects of Fitness Functions on Genetic Programming-Based Ranking Discovery For Web Search* escrito por Weiguo Fan, é apresentado o estudo a respeito da função de fitness com relação à eficiência e à efetividade de algoritmos evolutivos. O autor conclui que a adequação da função de fitness é fundamental para a melhoria no desempenho ([16]).

2.4.3 População

A população contém as possíveis soluções (representação destas), nesses termos, a população é uma coleção de genótipos. Ela é um passo da evolução, seus indivíduos são estáticos, isto é, não sofrem adaptação nem se modificam dentro da população sob uma geração fixa; em geral, a quantidade de indivíduos contidos na população se mantém fixa durante todo o processo de busca evolutiva.

Os operadores de variação atuam a nível de indivíduos, isto é, uma vez que o indivíduo é selecionado tais operadores atuam. Já os operadores de seleção agem a nível de população, daí, levam em conta a característica de todos os genótipos envolvidos na população e decide sua ação em função dessa.

A medida de diversidade está associada ao número de diferentes soluções que estão contidas em um dado momento da população, assim, é possível perceber que como a população é mutável e nesse contexto altera seus indivíduos ao longo do tempo, a diversidade tende a ser modificada durante toda a busca. É importante compreender que embora seja possível ter um único valor de fitness, ao fim de um processo de busca, não significa que se tenha um único genótipo; também, um único fenótipo não significa que se tenha um único genótipo. Por outro lado, o contrário não é verdadeiro, se houver um único genótipo terá um único valor de fitness e também um único fenótipo associado.

Alander J. T. em seu trabalho *On optimal population size of genetic algorithms* apresenta um estudo sobre o tamanho ótimo da população como função da complexidade do problema, o autor conclui que em máquinas sequenciais para problemas de complexidade moderada, o tamanho ótimo da população é função do tamanho do vetor que representa um genótipo. Já em arquitetura paralela, o tamanho ótimo é maior que o correspondente caso em arquitetura sequencial e que o tamanho exato é sensível a detalhes na implementação ([2]). Já o trabalho de Arabas J., intitulado *GAVaPS-a genetic algorithm with varying population size* apresenta uma abordagem de população variável que aumenta ou diminui em função de algumas características da pesquisa, ele justifica que se o tamanho da população for muito pequeno a convergência pode ser muito rápida e levaria a um máximo/mínimo local; e que se a população for muito grande, haverá desperdício dos recursos computacionais e o tempo de espera por uma melhoria pode ser demasiado longo ([3]).

2.4.4 Mecanismo de seleção dos reprodutores

Os mecanismos adotados para selecionar os reprodutores são quaisquer algoritmos que beneficiam um genótipo proporcionalmente à sua qualidade na função de avaliação. Nesse sentido, os melhores genótipos terão maior chance de se tornarem os genótipos-pais (um genótipo será definido como genótipo-pai se o

mesmo for selecionado através de algum mecanismo de seleção dos reprodutores). A composição do mecanismo de seleção dos reprodutores e o mecanismo de seleção dos sobreviventes são responsáveis pela melhora na qualidade da população através da eliminação/inserção de indivíduos menos/mais qualificados.

Vale salientar que o algoritmo que terá o papel de selecionar os genótipos tem características estocásticas, isto é, a seleção do indivíduo mais qualificado não é determinístico, e sim probabilisticamente, de forma que, indivíduos mais qualificados terão maior probabilidade de serem selecionados. Probabilidade esta que é proporcional à sua qualidade (quão melhor for a qualidade, maior será a probabilidade de seleção associada a tal genótipo), embora todos os indivíduos tenham probabilidade positiva dando a possibilidade que o genótipo menos qualificado também possa ser selecionado. Essa condição garante que a seleção não sentencie que a função de avaliação tenda para um máximo/mínimo local, assegurando a diversidade dos alelos que se encontram espalhados por toda a população (basta imaginar o caso em que dado um momento da evolução, haja somente uma qualidade limitada de alelos comuns aos genótipos mais qualificados. Daí, se vier a selecionar deterministicamente os mais qualificados, está apenas combinando a mesma coleção de alelos, e assim, a cota superior da função de avaliação seria limitada a combinação dos alelos mais significativos desse subconjunto próprio dos alelos da população).

O problema do Caixeiro viajante - *Travelling Salesman Problem* ([4]) consiste de um problema de otimização em que se busca a melhor rota passando uma única vez por diversos pontos em uma topologia qualquer, é apresentado por Larrañaga em seu trabalho *Genetic Algorithms for the Travelling Salesman Problem: A Review of Representations and Operators* ([28]). Neste trabalho o autor apresenta os operadores de seleção, dentre outros operadores, em diversas representações (binária, caminho, adjacência, ordinal e matricial) verificando os desempenhos experimentais de tais operadores em função das diversas representações.

Já o trabalho de Goldberg e Deb, intitulado *A comparative analysis of selection*

schemes used in genetic algorithms ([21]) apresenta uma análise sobre os operadores de seleção mais utilizados em computação evolutiva, entre eles temos: reprodução proporcional, ranking, torneio e genitor (estado estacionário). Como resultado, são fornecidas soluções aproximadas e exatas para simulações, bem como tempo de convergência útil e taxa de crescimento; a principal contribuição é dada por sugestão de caminhos para investigação analítica mais detalhada das técnicas de seleção.

2.4.5 Operadores de variação: mutação e recombinação

Os operadores de variação atuam nos genótipos modificando-os a fim de obter melhoramentos na qualidade do indivíduo, isto é, recebem genótipos selecionados na população através de operadores de seleção e os modifica seguindo algum algoritmo. Basicamente há dois tipos de operadores: Mutação e recombinação ou cruzamento.

Mutação

A mutação tem a tarefa de modificar o genótipo selecionado atuando diretamente na modificação de um ou mais alelos através de alguma rotina estocástica que define o alelo que sofrerá a mutação (Essa condição garante que as modificações ocorram em diferentes segmentos e assegure genótipos distintos a cada aplicação da mutação, mesmo que seja utilizado o mesmo genótipo como entrada para o operador).

Sob o espaço dos fenótipos, isto é, o espaço de soluções, o ato de aplicar o operador mutação é na verdade tomar uma nova solução para o problema em estudo. Assim, é possível perceber que para um tempo t, suficientemente grande, a mutação possibilitará que através de computação evolutiva se obtenha o ótimo (máximo ou mínimo) global ([15]).

Recombinação

A recombinação (ou cruzamento) é a criação de genótipos. Assim, uma vez que através de um operador de seleção foram escolhidos os genótipos pais, a

recombinação atua selecionando um (ou mais de um) ponto de corte de forma aleatória em que será permutado os alelos de um genótipo pai para outro. Em outras palavras, o processo de recombinação transmite parte da informação (alelos) dos pais para os filhos. Tal atitude, gera genótipos filhos que compõem parte dos alelos de cada um de seus genótipos pais.

É possível perceber que a escolha aleatória do ponto de corte (a partir de que momento a informação será misturada) garante que mesmo que, por ventura, escolha-se em um outro momento os mesmos genótipos pais para obter informação, terá pouca chance de que os atuais genótipos filhos sejam semelhantes (no quesito informação genética) de outros genótipos filhos gerados anteriormente por esses selecionados.

O espaço dos fenótipos encara tal modificação como uma nova solução obtida através de uma função de outras duas soluções, que naquele momento, se mostravam as melhores soluções para o problema.

No trabalho *Adaptive probabilities of crossover and mutation in genetic algorithms* ([36]) foi apresentada uma abordagem sobre operadores de mutação e recombinação sob o ponto de vista adaptativo, isto é, a probabilidade de mutação e recombinação varia dependendo do valor da função de avaliação dos genótipos envolvidos. Essa tentativa visa a proteger as soluções de alto valor de fitness. Ainda nesse trabalho, também é possível definir o valor ótimo para a probabilidade de mutação e recombinação para cada caso, não sendo necessário arbitrar uma probabilidade para esses operadores de forma geral. O autor justifica que essa abordagem se mostra satisfatória em dois sentidos: Manter a diversidade da população e sustentar a capacidade de convergência do algoritmo.

2.4.6 Mecanismo de seleção dos sobreviventes (Substituição)

Uma vez que se tenha uma coleção de genótipos (população), e após a aplicação dos operadores de seleção - mutação - recombinação, terá também os genótipos filhos.

A quantidade de elementos na coleção deve ser constante para não prejudicar o desempenho computacional no processo de busca, assim, devemos ter um mecanismo que substituirá tantos quantos forem os genótipos filhos na população. Porém, isso ocorre no melhor caso em que todos os genótipos filhos tem valor em sua função de avaliação maior que o de seus genótipos pais, caso contrário, sua inserção não contribuiria em nada para a convergência da função de avaliação para seu ótimo global (de fato, como todos os genótipos filhos tem alelos ofertados por seus genótipos pais a menos dos alelos alterados pelo processo mutacional, e que o valor de suas funções de avaliação não superaram o valor apresentado pelos genótipos pais, podemos concluir que os alelos pertencentes ao genótipo filho se distanciaram ainda mais do genótipo ótimo).

Com relação à característica, o mecanismo de seleção dos sobreviventes se assemelha muito ao procedimento apresentado pelo operador de seleção que fará seu julgamento através do valor da função de fitness, embora sua concepção (diferente do operador de seleção) é dada de forma determinística. Ou seja, o critério de substituição será feito na tentativa de melhorar a população. Com isso, uma forma é classificar através do fitness e a partir disso identificar os de menor valor (coleção de genótipos + genótipos filhos) e daí, eliminar tantos genótipos quanto forem necessários para manter a coleção com a mesma cardinalidade da geração anterior.

Figura 3: Ilustração da característica monótona e não negativa da função de avaliação (Eiben, 2008).

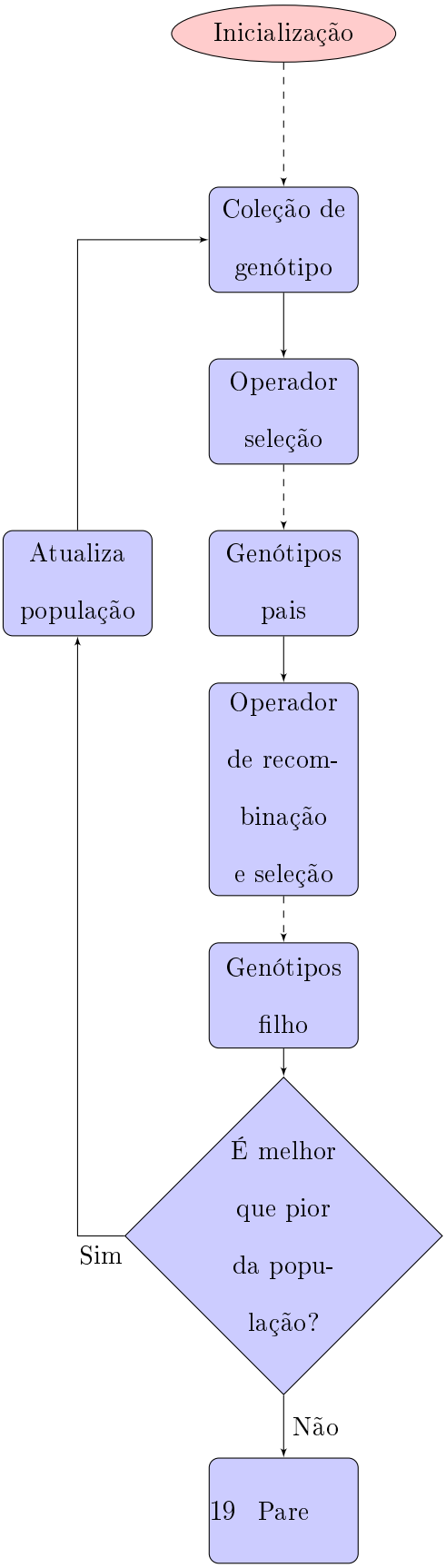
A cada passo teremos uma função de avaliação monótona e não negativo. Na Figura 3 apresentamos uma ilustração da curva da função de avaliação, também é possível perceber o ciclo que se forma entre selecionar - mutacionar - recombinar - substituir, fluxograma mostrado na página 19.

Por fim, uma vez o ciclo formado, é necessário que se defina como iniciar e finalizar o processo de busca. Geralmente o critério para iniciar a população é o preenchimento aleatório de todos os alelos. Assim, será obtida uma coleção de genótipo bem heterogênea.

Quanto à finalização, ao conhecer o valor da função de fitness ótimo, é possível estabelecer um critério de precisão (digamos $\epsilon > 0$), e com isso, obter um critério de parada para a busca tão logo se obtenha um valor para função de avaliação de algum candidato com distância menor que ϵ. Por outro lado, em um sentido mais realista, geralmente não é de conhecimento o ponto ótimo ou não há garantia de que se consiga chegar tão próximo do mesmo para satisfazer um critério de parada nesse contexto. Assim, cabe utilizar outra abordagem para finalizar um processo de busca:

- Exceder o tempo limite de CPU;
- Atingir uma quantidade de avaliações do fitness;
- Treinamento sem progresso, isto é, não ocorrer mudança por um tempo determinado;
- A diversidade de genótipos decaia até um limiar pré-estabelecido.

Basta que um desses critérios seja adotado para que se garanta a finalização do processo de busca.


Inicialização
Coleção de genótipo
Operador seleção
Genótipos pais
Operador de recombinação e seleção
Genótipos filho
É melhor que pior da população?
Sim
Atualiza população
Não
19 Pare

2.5 Codificação de algoritmo genético

Agora que são conhecidos os componentes necessários para trabalhar com computação evolutiva, a compreensão será direcionada para a forma de codificar um problema através da abordagem criada por Holland, isto é, algoritmo genético - AG. Não somente será abordada essa técnica, como será focado na abordagem adotada para modelagem do problema no próximo capítulo.

2.5.1 Representação do genótipo - AG

Uma representação em que o genótipo (indivíduo) é um string binário, isto é, um vetor (denomina-se vetor no sentido de informar que o posicionamento de cada bit tem importância) formado apenas por 0 ou 1.

Neste sentido, o vetor como um todo é um indivíduo e cada alelo é um agrupamento de bits, digamos 4 bits. Daí, é possível representar números que variam de 0 até 15 em cada alelo, neste exemplo.

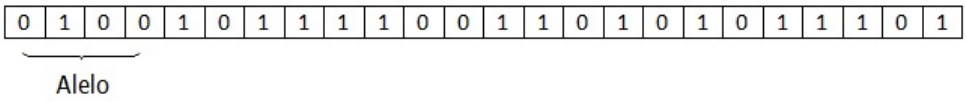


Figura 4: Ilustração de um genótipo em sua codificação binária. Note que, neste exemplo, a cada 4 bits formamos um alelo.

Na Figura 4 é ilustrado o que seria um genótipo na codificação binária para um vetor pertencente ao conjunto $\{0,1\}^{24}$. Daí, fazendo a decodificação nota-se que se trata do vetor fenótipo $\langle 4, 11, 12, 13, 5, 13\rangle$, isto é, um vetor do conjunto $\{0, 1, \cdots, 15\}^6$. Assim é possível perceber que o espaço fenótipo está univocamente correspondido com espaço genótipo tornando-o um isomorfismo do espaço que compõe nosso problema.

Outras formas de representação são possíveis como é o caso da representação por números inteiros ou representação de número real (ponto flutuante).

2.5.2 Operador de Mutação - AG

Na codificação binária o processo de mutação é realizado através de uma pequena probabilidade $p_m > 0$, assim, sob tal probabilidade o alelo selecionado tem seu valor alterado (de 0 para 1 ou de 1 para 0, também conhecido como *bit flip*). A quantidade de elementos que sofrem mutação geralmente é dado por $K \cdot p_m$, em que K é o comprimento do genótipo.

Por exemplo, suponha um genótipo formado por $K = 20$ bits, sendo cada alelo formado por 4 bits, se fixarmos a probabilidade de mutação de $p_m = 5\%$ haverá a possibilidade mutar 1 bit desse genótipo $1 = 0,05 \cdot 20 = p_m \cdot K$. Na Figura 5 é ilustrado o processo de mutação, os pontos que sofrerão a mudança também são escolhidos aleatoriamente dentre todos os pontos que compõem o indivíduo.

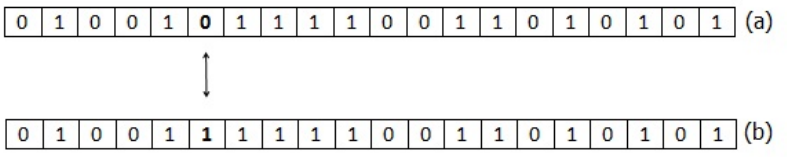


Figura 5: Ilustração do processo de mutação na codificação binária, através do processo bit flip. Em (a) o genótipo inicial e em (b) o mesmo genótipo após a mutação.

2.5.3 Operador de recombinação - AG

O operador de recombinação geralmente associado a uma probabilidade de ocorrência p_r que varia entre $[0,5; 1,0]$. Assim, se aplicado um teste o valor for menor ou igual a p_r, o operador de recombinação gera os fenótipos filhos através da

recombinação de seus alelos; caso contrário, os fenótipos filhos são gerados de forma assexuada (através de uma cópia direta dos seus pais).

A recombinação na representação binária pode ocorrer sob o ponto de corte diretamente sobre 1 ponto entre os bits (em que a partir do ponto de corte toda a informação do genótipo 1 é repassado para o genótipo 2 e vice-versa) ou através de n pontos através da permuta dos bits que se encontram entre dois pontos de corte distintos. A seleção do ponto de corte é dado aleatoriamente seguindo algum processo de seleção.

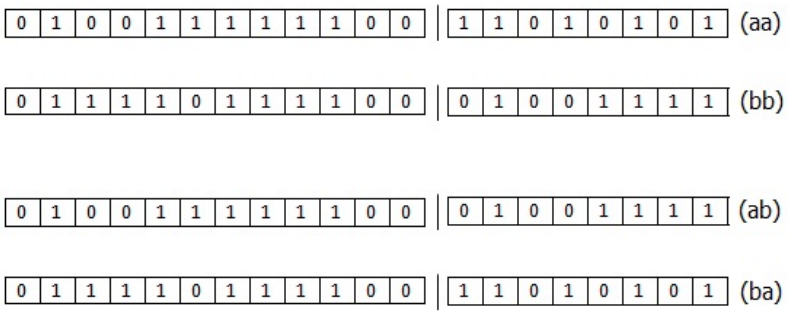


Figura 6: Ilustração do operador de recombinação atuando sobre dois genótipos pais através de uma recombinação de um único ponto.

Nas Figuras 6 e 7 ilustramos as duas situações, na primeira a atuação do operador de recombinação para 1 ponto e a atuação para um caso em que tomamos 3 pontos em um genótipo de comprimento $K = 20$.

2.5.4 Modelo de população

Há dois tipos de modelos populacionais, o modelo geracional e o de estado estacionário ([15]):

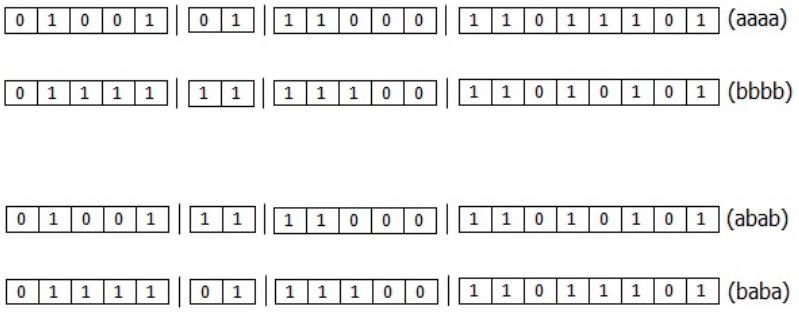


Figura 7: Ilustração do operador de recombinação atuando sobre dois genótipos pais através de uma recombinação de três pontos. Note que as partes 2 e 4 de cada fenótipo foram permutadas.

- Geracional - Neste modelo toda a coleção de genótipos é selecionada para receber os operadores de variação, com isso, são gerados genótipos filhos em quantidade igual à cardinalidade da coleção de genótipos pais. Após cada geração, toda a coleção é substituída pelos genótipos filhos gerados. Esse modelo apresenta uma situação indesejada quando a mutação modifica os alelos mais significativos e ao modificar toda população perde-se tal informação, daí, o indivíduo sofre uma perda em sua adaptabilidade nessa geração.

- Estado estacionário - Neste modelo não é a coleção que modifica a cada geração e sim parte dela. Assim uma quantidade q (menor que a cardinalidade da coleção) será substituída por q genótipos filhos; a porcentagem de elementos que serão selecionados é dada por q/K em que K é a cardinalidade da coleção. Para Whitney, em seu algoritmo Genitor, o valor de q é igual a 1, e com isso, o percentual de indivíduos selecionados é dado por $1/K$ ([39]).

2.5.5 Mecanismo de seleção dos reprodutores

Seleção proporcional ao fitness

Consiste de selecionar os genótipos pais através da proporção gerada pela comparação do valor função de avaliação com relação à soma dos valores da função

de fitness da coleção de genótipo. A expressão proporcional é dada por:

$$fp_i = \frac{f_i}{\sum_{i=1}^{K} f_i} \tag{2.1}$$

em que f_i é o valor da função de avaliação para o genótipo i.

O principal problema desse método é a convergência prematura, em que um genótipo obtenha um valor muito alto para sua função de avaliação e com isso passe a dominar o processo de seleção forçando toda a coleção a ser uma combinação de seus alelos. Daí, a busca evolutiva ficaria restrita a um ótimo local que dependa dos alelos desse genótipo ([23]). Uma tentativa de diminuir a possibilidade dessa convergência seria a condição de que o indivíduo com valor de fitness muito alto já houver sido selecionado para geração de um filho, este terá um fator associado que temporariamente diminua seu valor de fitness dando possibilidade aos demais genótipos de distribuir sua informação genética.

Seleção por ranking

Nesse método, é feito um mapeamento do valor da função de fitness através de uma função que defina sua probabilidade de escolha. Em geral, é adotada uma expressão linear ou exponencial. Para o caso em que a expressão é linear usa-se como parâmetro um valor t $(1,0 < t \leq 2,0)$, daí, uma possível expressão linear é dada por:

$$P_{ranking}(i) = \frac{2-t}{K} + \frac{2i(t-1)}{K(K-1)} \tag{2.2}$$

em que K é a cardinalidade da coleção de genótipos e o índice i se refere ao genótipo i.

Essa abordagem visa melhorar o critério de seleção nos momentos em que toda a coleção apresenta valor de fitness muito próximo, e sob o método de seleção anterior, viria a ser uma escolha praticamente aleatória sem evidenciar a qualidade que cada

genótipo associe (mesmo que seja muito sutil tal qualidade) ([6]).

2.5.5.1 Seleção pelo algoritmo roleta

Esse algoritmo visa obter os indivíduos através de um procedimento semelhante a uma roleta. Inicialmente é utilizado o ranking ou o fitness proporcional para definirmos a probabilidade P_i para cada indivíduo i da coleção contendo K elementos. Esse procedimento compartilha o mesmo problema da seleção proporcional que pode gerar convergência prematura.

Uma vez definida a probabilidade de cada indivíduo proporcional ao valor de sua função de avaliação, são tomados os pontos acumulados:

$$a_j = \sum_{i=1}^{j} P_i \tag{2.3}$$

assim, haverá a lista $(0, a_1, a_2, \ldots, a_K)$ que representam os pontos de acumulados das probabilidades dos indivíduos i. Note ainda que o intervalo $[a_{i-1}, a_i] = P_K$, isto é, cada partição dessa lista diz respeito ao intervalo proporcional ao valor de fitness para o indivíduo correspondente.

Sabendo que:

$$\sum_{i=1}^{K} P_i = 1 \tag{2.4}$$

basta que se selecione aleatoriamente um ponto entre o intervalo $[0, 1]$, e dado a partição em que esse ponto pertença, será selecionado o indivíduo correspondente.

Seleção por torneio

O procedimento por torneio tem características mais simplistas, ideal para grandes populações ou coleções que não ocorra acesso a todas as informações naquele

presente. Sua grande vantagem é não ter a necessidade de fazer uma avaliação com relação à toda a população, e sim a uma amostragem desta.

Inicialmente, define-se a quantidade de genótipos pais que serão selecionados, digamos k, daí, selecionam-se n elementos da população aleatoriamente (claramente $n < K$), e de posse desses genótipos será avaliado o de melhor valor de fitness. Chama-se esse indivíduo de g_1, e se repete o procedimento até que se obtenha todos os k genótipos desejados.

Notoriamente, quanto maior for a quantidade de elementos no torneio, melhor será a chance de se obter genótipos com alto valor de fitness e menor será a chance de que ocorra um torneio completamente preenchido de indivíduos com baixo valor de fitness ([7]).

2.5.6 Mecanismo de seleção dos sobreviventes (substituição)

Sobrevivência dos genótipos mais novos

Nesse contexto, não importa quão alto seja o valor de fitness obtido por um genótipo, ele só permanecerá na população uma quantidade fixa de gerações. Assim, os indivíduos mais velhos "morrem" ao completar seu ciclo de vida dando lugar para os mais novos.

Perceba que a lógica adotada não visa uma função do tipo serra, em que os valores de fitness atingem um patamar mais elevado e depois de uma quantidade fixa de gerações perdem esse efeito e voltam ao patamar anteriormente estabelecido; espera-se que os alelos significativos tiveram tempo suficiente para serem repassados para os demais genótipos mais novos, e a partir disso, assegurar que o ótimo local alcançado não seja perdido (geralmente).

É possível perceber que o modelo de população geracional, em que todos são substituídos, pode ser considerado a aplicação do caso trivial do mecanismo de seleção por sobrevivência, em que, o ciclo de vida de qualquer genótipo é de uma única geração.

Sobrevivência baseada no valor da função de fitness

O caso mais utilizado é sem dúvida o de *substituição do pior genótipo*. Daí, se tivermos k genótipos filho e K elementos na população, isto é $K+k$ elementos para que sejam selecionados K dentre estes; é feito um ranking com relação ao valor de fitness de todos os $K+k$ indivíduos, e por fim, selecionado os K primeiros elementos.

Esse tipo de estratégia tem o problema de convergir para ótimos locais, assim, são normalmente associados a grandes coleções de genótipos ou exigências de não haver mais de um genótipo com a mesma combinação de alelos.

Outra característica é o *elitismo*, nesse caso, os indivíduos de mais alto valor de fitness permanecem na população. Geralmente vem associado à estratégia de sobrevivência dos mais novos, garantindo assim que a condição de um indivíduo velho com alto valor de fitness permaneça na população nos casos em que nenhum dos indivíduos gerados melhoram ou estabilizam o valor da função de fitness.

2.6 Algoritmo genético utilizado no trabalho

Neste trabalho, utilizaremos uma variante melhorada do algoritmo genético, o algoritmo melhorado do Leung ([29]) utilizado para treinar os parâmetros da redes neurais. Resultados comparativos mostram a superioridade dessa variante com relação ao algoritmo genético tradicional.

3 Metodologia

3.1 Banco de dados

O banco de dados que será utilizado foi obtido através do desempenho semestral de uma turma de 40 alunos do curso de Licenciatura em Química do Instituto Federal de Pernambuco - IFPE, como foram realizadas duas avaliações, há 40 pontos cartesianos $(Nota_1, Nota_2)$, a dispersão das notas é apresentada na tabela 1:

Tabela 1: Desempenho semestral Turma de Estatística - Tabulação.

(7, 3.1)	(9, 4)	(1.5, 5.5)	(10, 8.5)	(9.5, 8.3)
(9.5, 6)	(8.5, 9)	(8.2, 7.6)	(8.3, 7)	(7.9, 6.1)
(8, 9)	(7.5, 8.5)	(3.1, 2.9)	(5.8, 5)	(1.8, 1)
(3, 5)	(7.1, 5.1)	(5.3, 7.9)	(6.2, 6.1)	(7.4, 6.9)
(5.9, 4)	(4.1, 6.1)	(5.5, 6.5)	(4.3, 2.6)	(4.2, 3.9)
(6.5, 7.9)	(5.1, 4.3)	(4.8, 5)	(0.5, 0.5)	(3.1, 1.6)
(1.1, 1)	(2, 2.3)	(2.7, 1.8)	(4, 5)	(2.3, 4.1)
(1, 1.8)	(3.7, 4.1)	(2.5, 2.9)	(1.5, 3.1)	(0.9, 2.5)

As estatísticas básicas das notas é apresentada na tabela 2.

Para medir o nível de correlação será usado o coeficiente de correlação de Pearson ou coeficiente de correlação produto-momento. Esse coeficiente mede o grau de associação linear (e a direção dessa correlação - se positiva ou negativa) entre duas variáveis métricas. Normalmente se representa por ρ. Quando o valor de $\rho = 1$

Tabela 2: Estatísticas descritivas para o desempenho da turma.

	Mínimo	1^o Quartil	Mediana	Média	3^o Quartil	Máximo
1^a Prova	0.500	2.650	4.950	5.008	7.425	10.000
2^a Prova	0.500	2.900	5.000	4.838	6.600	9.000

significa que há uma forte correlação positiva entre as variáveis, já se $\rho = -1$ há uma forte correlção negativa entre as variáveis e se $\rho = 0$, então não há dependência linear entre as duas variáveis. Entretanto, um $\rho = 0$ não significa que não haja dependência, somente garante-se que esta não seja linear. Tendo enfim $-1 \leq \rho \leq 1$.

Para calcular o coeficiente de Pearson utiliza-se a expressão [10]:

$$\rho = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3.1}$$

em que $x_1, x_2, \cdots, x_n$ são as notas da turma para a primeira prova, $y_1, y_2, \cdots, y_n$ são as notas da turma para a segunda prova. Ainda, $\bar{x}$ e $\bar{y}$ são as médias aritméticas para a primeira e segunda prova, respectivamente.

A interpretação do coeficiente de Pearson é dada por:

- $\rho = 1$ correlação perfeita;
- $\rho > 0.7$ positivo ou negativo indica uma forte correlação;
- $0.3 \leq \rho < 0.7$ positivo ou negativo indica uma correlação moderada;
- $0 < \rho < 0.3$ positivo ou negativo indica uma fraca correlação;
- $\rho = 0$ correlação inexistente.

Além disso, é possível discutir a interpretação geométrica para o coeficiente de Pearson. Basta tomar as séries $x = (x_1, x_2, \cdots, x_n)$ e $y = (y_1, y_2, \cdots, y_n)$ como sendo vetores em um espaço de dimensão n: $x =< x_1 - \bar{x}, x_2 - \bar{x}, \cdots, x_n - \bar{x} >$ e $y =< y_1 - \bar{x}, y_2 - \bar{x}, \cdots, y_n - \bar{x} >$.

Assim, o cosseno do ângulo α entre esses dois vetores é dado pela expressão (produto escalar normado):

$$cos(\alpha) = \frac{\sum_{i=1}^{n}(x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3.2}$$

portanto $cos(\alpha) = \rho$.

Dessa forma, o coeficiente de correlação é o cosseno do ângulo entre dois vetores.

- Se $\rho = 1$, o ângulo $\alpha = 0$, os dois vetores são colineares;
- Se $\rho = 0$, o ângulo $\alpha = 90^o$, os dois vetores são ortogonais;
- Se $\rho = -1$, o ângulo $\alpha = 180^o$, os dois vetores são colineares com sentidos opostos.

Por fim, $\alpha = arccos(\rho)$ [11].

Aplicando o coeficiente de Pearson ao banco de dados obtemos o valor $\rho = 0,77813$ indicando uma forte correlação positiva entre os notas da primeira e da segunda prova. A Figura 8 apresenta a dispersão das notas como pares ordenados, é possível perceber a tendência linear positiva apresentada pelo coeficiente de Pearson.

3.2 Análise de Regressão

A análise de regressão consiste na realização de uma análise estatística com o objetivo de verificar a existência de uma relação funcional entre uma variável dependente (segunda nota) com uma ou mais variáveis independentes (primeira prova). De outro modo, é a tentativa de explicar as variações ocorridas na variável dependente em função das variações apresentadas pela variável independente.

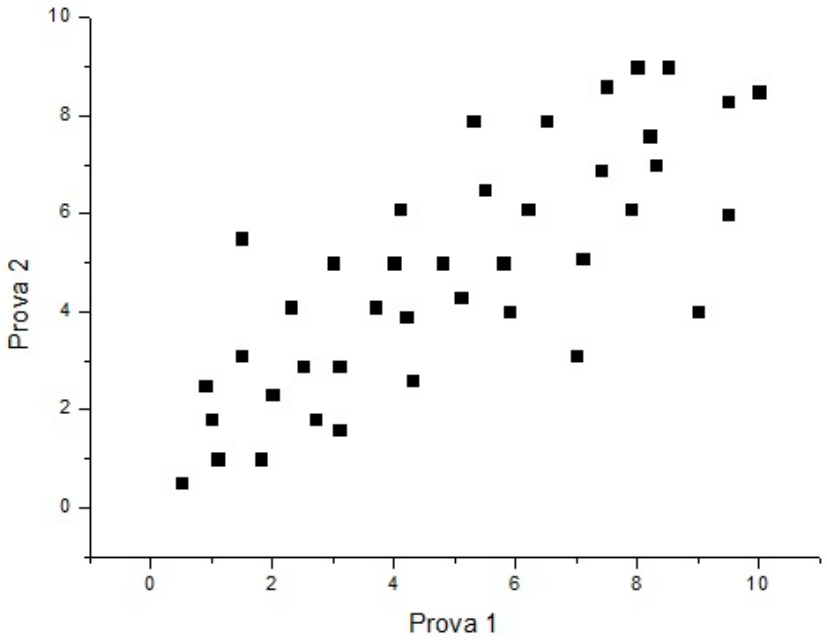


Figura 8: Dispersão das notas semestrais, Prova 1 × Prova 2.

Na tentativa de estabelecer a equação, é possível fazer o gráfico (chamado gráfico de dispersão, Figura 8) para verificar como se comportam os valores da variável dependente (Y) em função da variável independente (X).

O comportamento apresentado por Y em relação a X pode ter qualquer formato: linear, quadrático, cúbico, exponencial, logarítmico e etc. Para se estabelecer o modelo para explicar o fenômeno, deve-se verificar qual o tipo de curva e a equação do modelo matemático que mais se aproxime dos pontos representados no diagrama de dispersão.

Claramente, não será possível obter uma curva que se ajuste a todos os pontos, em sua maioria, os pontos estarão a uma certa distância da curva-modelo. Isso se deve ao fato de que o banco de dados em estudos não ser oriundo de uma função matemática, muito além disso, o banco de dados foi originado por um experimento estocástico que se encontra sujeito a erros de medição e outras características aleatórias. Dessa forma, a regressão tem como objetivo obter o modelo matemático

que melhor se ajusta ao banco de dados.

Um dos métodos para obtenção dos estimadores das constantes do modelo é Método de Mínimos Quadrados que visa obter a minimização dos resíduos ocasionados entre os valores observados no banco de dados e os valores estimados pelo modelo. Tal demonstração para o caso de regressão linear simples foi apresentado na Introdução e Motivação 1.

Diante disso, os estimadores obtidos para o modelo:

$$y = \alpha + \beta \cdot x + \epsilon \tag{3.3}$$

São apresentados na Tabela 3.

Tabela 3: Estimadores para o intercepto α e a inclinação β.

	Valor	Erro padrão
Intercepto α	1,45334	0,50642
Inclinação β	0,67632	0,08856
R^2 Ajustado	0,5951	N / A

assim, o modelo preliminar baseado em regressão linear simples para explicar a dependência linear entre a primeira e segunda prova é dado por:

$$y = 1,45334 + 0,67632 \cdot x + \epsilon \tag{3.4}$$

a interpretação livre sugere que se ele tirar zero na primeira prova, espera-se que tire 1,45 na segunda. O aumento de uma unidade na primeira prova implica em um aumento de 0,68 na segunda.

Isso posto, é possível determinar a nota crítica para prever a aprovação na disciplina. Assumindo a média de $6,0$, temos que seria necessário obter nota acima ou igual a $5,27333$ para assegurar a aprovação na disciplina através do modelo linear

proposto.

Ainda, é preciso notar que o valor do R^2 ajustado foi de $0,5951$. Essa estatística fornece informação sobre o ajuste do modelo para o banco de dados, assumindo a "porcentagem"da variação de Y que é explicada pela regressão, ou quanto da variação na variável dependente Y está sendo explicada pela variável X.

O R^2 é obtido pela expressão:

$$R^2 = \frac{SQReg}{SQTotal} = 1 - \frac{SQRes}{SQTotal} \tag{3.5}$$

em que SQRes é a soma dos quadrados dos resíduos - soma das diferença entre as amostras da variável dependente (segunda prova) e as amostras estimadas para a segunda prova a partir do modelo; SQTotal é a soma dos quadrados totais - somatório dos quadrados das amostra da variável indepentende (primeira prova) subtraído do quadrado do somatório das amostras da variável independente dividido pelo total de amostras.

Já o R^2 ajustado é a mesma proporção só que penalizando a inclusão de regressores (amostras) pouco explicativas. Sua expressão é dada por [10]:

$$R^2_{ajustado} = 1 - \frac{n-1}{n-(k+1)}(1-R^2) \tag{3.6}$$

em que $k+1$ representa o número de variáveis explicativas mais a constante, n é a quantidade de amostras e R^2 é o coeficiente determinado anteriormente.

Com isso, temos que o modelo foi capaz de explicar $59,51\%$ do banco de dados trabalhado.

3.3 Testando a normalidade dos dados e dos resíduos

A normalidade dos resíduos é uma suposição essencial para que os resultados do ajuste do modelo de regressão linear sejam confiáveis. Podemos verificar essa suposição por meio de gráfico de Papel de probabilidade e por meio de testes tais como Shapiro-Wilk, Anderson-Darling e Kolmogorov-Smirnov.

Neste trabalho será utilizado o teste de Shapiro-Wilk, e sua justificativa se deve ao fato de ser um teste bem ajustado para amostras menores do que 50.

O teste de hipóteses utilizado é dado por:

- H_0 : Os erros (desvios) da característica em estudo seguem a distribuição normal;
- H_1 : Os erros não seguem a distribuição normal.

ou seja, é necessário não rejeitar H_0 para assegurar normalidade para os dados.

O nível de significância adotado é o nível padrão de 5%. A estatística para aferir o teste é dado por:

$$W_c = \frac{g^2}{SQE} \tag{3.7}$$

em que:

$$g = \sum_{i=1}^{n/2} a_{i,n} \cdot (x_{n-i+1} - x_i) \tag{3.8}$$

e SQE:

$$SQE = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (x_i - \bar{x})^2 \tag{3.9}$$

os valores para $a_{i,n}$ são obtidos através da tabela para teste Shapiro-Wilk, a Figura 9 ilustra a tabela necessária.

A conclusão do teste é dada pela análise do valor de W_c, se o valor de W_c for menor que o valor crítico W_t obtido na tabela em função do tamanho da amostra (n) e significância (α), rejeita-se H_0 e conclui-se que a característica em estudo da população ou os erros não seguem a distribuição normal, caso contrário, aceita-se H_0 [10].

Utilizando o teste de Shapiro-Wilk em ambos os bancos de dados, prova 1 e prova 2, obtém-se o resultado apresentado na tabela 4. É possível perceber que os valores obtidos no teste foram $0,9539$ e $0,96429$, respectivamente, em ambos os casos foi constatado que não rejeita-se H_0 o que confirma a normalidade dos resíduos, isto é, a variância dos erros é constante.

Tabela 4: Quadro para o teste Shapiro-Wilk.

GL	Estatística W	Prob < W
40	0,9539	0,1033
40	0,96429	0,23414

3.3.1 Testes de Hipótese na regressão linear simples

É necessário que após feita a regressão e ajustado o modelo, verifique-se a adequabilidade por meio de testes de hipóteses ou intervalo de confiança.

Isso só é verdade se for pressuposto que os erros tenham distribuição normal. Para constatar esse fato, é possível usar o teste de aderência (qui-quadrado) ou recorrer ao teste de Shapiro-Wilk. Neste trabalho foi utilizado o segundo teste e sua descrição foi apresentada neste capítulo na seção de normalidade 3.3.

Teste de Shapiro-Wilk
Coeficientes

Anexo 3. Coeficientes α_{N-i+1} para o teste de normalidade W de SHAPIRO – WILK

(Para N = 2(1)50)

$i \setminus N$	2	3	4	5	6	7	8	9	10	
1	0.7071	0.7071	0.6872	0.6646	0.6431	0.6233	0.6052	0.5888	0.5739	
2		0.0000	0.1677	0.2413	0.2806	0.3031	0.3164	0.3244	0.3291	
3				0.0000	0.0875	0.1401	0.1743	0.1976	0.2141	
4						0.0000	0.0561	0.0947	0.1224	
5								0.0000	0.0399	

$i \setminus N$	11	12	13	14	15	16	17	18	19	20
1	0.5601	0.5475	0.5359	0.5251	0.5150	0.5056	0.4968	0.4886	0.4808	0.4734
2	0.3315	0.3325	0.3325	0.3318	0.3306	0.3290	0.3273	0.3253	0.3232	0.3211
3	0.2260	0.2347	0.2412	0.2460	0.2495	0.2521	0.2540	0.2553	0.2561	0.2565
4	0.1429	0.1586	0.1707	0.1802	0.1878	0.1939	0.1988	0.2027	0.2059	0.2085
5	0.0695	0.0922	0.1099	0.1240	0.1353	0.1447	0.1524	0.1587	0.1641	0.1686
6	0.0000	0.0303	0.0539	0.0727	0.0880	0.1005	0.1109	0.1197	0.1271	0.1334
7			0.0000	0.0240	0.0433	0.0593	0.0725	0.0837	0.0932	0.1013
8					0.0000	0.0196	0.0359	0.0496	0.0612	0.0711
9							0.0000	0.0163	0.0303	0.0422
10									0.0000	0.0140

Figura 9: Ilustração para a tabela do teste Shapiro-Wilk. [33]

Como há dois parâmetros no modelo trabalhado, pode-se realizar os seguintes testes:

- H_0: β=β^* contra H_1: $\beta \neq \beta^*$
- H_0: α= α^* contra H_1: $\alpha \neq \alpha^*$

Em cada um a estatística de teste e as conclusões seriam:

$$t_{calc} = \frac{\beta - \beta^*}{\sqrt{\hat{Var}(\hat{\beta})}} \tag{3.10}$$

em que

$$\hat{Var}(\hat{\beta}) = \frac{\hat{\sigma}^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{3.11}$$

e a regra de decisão para o primeiro teste é:

- Se $|t_{calc}| \geq t_{(\alpha/2,n-2)}$, então se rejeita a hipótese nula (H_0).

em que $t_{(a,b)}$ é o valor da distribuição t-Student para o nível de significância $2a$ e a quantidade de amostras $b + 2$.

Já para o segundo teste, temos:

$$t_{calc} = \frac{\alpha - \alpha^*}{\sqrt{\hat{Var}(\hat{\alpha})}} \tag{3.12}$$

em que

$$\hat{Var}(\hat{\beta}) = \hat{\sigma}^2 \left(\frac{1}{n} + \frac{\bar{X}^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \right) \tag{3.13}$$

e a regra de decisão para o segundo teste é a mesma:

- Se $|t_{calc}| \geq t_{(\alpha/2,n-2)}$, então se rejeita a hipótese nula (H_0).

Atentar para a estimativa da variância dos erros que é dada por:

$$\hat{\sigma}^2 = \frac{SQRes}{n-2} = \frac{\sum_{i=1}^{n} y_i^2 - \frac{\left(\sum_{i=1}^{n} y_i\right)^2}{n} - \hat{\beta} \cdot SPD_{xy}}{n-2} \tag{3.14}$$

com SPD_{xy} sendo:

$$\sum_{i=1}^{n}(x_i - \bar{x}) \cdot (y_i - \bar{y}) - \frac{(\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} y_i)}{n} \tag{3.15}$$

O caso especial, e o que geralmente se está interessado é:

- H_0: β=0 contra H_1: $\beta \neq 0$

esse teste de hipótese está relacionado com a significância da regressão, assim se não for rejeitado H_0 é o mesmo que concluir que não há relação linear entre X e Y. Por outro lado, se H_0 for rejeitado, indicaria que X é importante para explicar a variabilidade em Y.

3.3.2 ANOVA - Análise de Variância

De maneira alternativa, pode-se testar a significância da regressão pelo método de Análise de Variância (ANOVA).

O método em sí, consiste em fazer uma partição da variabilidade total da variável resposta Y em outros componentes de acordo com o modelo e o teste a ser feito. Pode-se verificar a identidade:

$$\sum \left(Y_i - \bar{Y}\right)^2 = \sum \left(\hat{Y}_i - \bar{Y}\right)^2 + \sum \left(Y_i - \hat{Y}\right)^2 \tag{3.16}$$

Ou simplesmente:

$$\text{SQTotal} = \text{SQRegressão} + \text{SQResíduo}$$

em que:

SQTotal = Variação total de Y;

SQRegressão = Variação em Y explicada pela regressão ajustada;

SQResíduo = Variação não explicada pela regressão.

A ANOVA é baseada no quadro 5:

A estatística F, é baseada na distribuição F de Snedecor, serve para testar a significância da regressão, isto é, testar:

- H_0: $\beta = 0$ contra H_1: $\beta \neq 0$

e a regra de decisão é dada por:

Tabela 5: Quadro para o cálculo da análise de variância.

FV	GL	SQ	QM	F
Regressão	1	SQRes	QMReg = SQReg	$\frac{QMReg}{QMRes}$
Resíduo	n-2	SQRes	QMRes=$\frac{SQRes}{n-2}$	
Total	n-2	SQTotal		

Se $F_{calc} \geq F_{(\alpha,1,n-2)}$, então rejeita-se H_0. Ao rejeitarmos H_0, assume-se que não há relação linear entre os parâmetros da regressão.

A equação obtida, apenas estabelece uma relação funcional, entre a variável dependente e a variável independente, para representar o fenômeno em estudo. Portanto, a simples obtenção da equação estimada não responde ao pesquisador se a variação da variável independente influencia significativamente na variação da variável dependente.

Para responder a essa pergunta, é necessário realizar o teste estatístico para as estimativas dos coeficientes da equação da regressão estimada. Um teste que pode ser utilizado para verificar tal fato é o teste F da análise de variância. Portanto, é necessário realizar uma análise de variância dos dados observados, em função do modelo proposto [10].

Para o banco de dados adotado nesse trabalho, a Tabela 6 apresenta os resultados da análise de variância na tentativa de assegurar a significância do modelo proposto. Nessa tabela é possível perceber que a ANOVA mostrou-se significante, isto é, o parâmetro β é diferente de zero e com isso há relação linear entre os parâmentros (nota da segunda prova tem relação linear com a nota da primeira prova).

Apresentamos na Figura 10 a dispersão dos pontos e a reta ajustada a partir da regressão linear, e na figura 11 apresentamos a plotagem da variável independe em função dos resíduos mostrando a tendência de normalidade apresentada pelos dados.

Tabela 6: Quadro para o cálculo da análise de variância.

FV	GL	SQ	QM	F	Prob > F
Regressão	1	139,47562	139,47562	58,31923	$3,44082 \times 10^{-9}$
Resíduo	38	90,88038	2,39159		
Total	38	230,356			

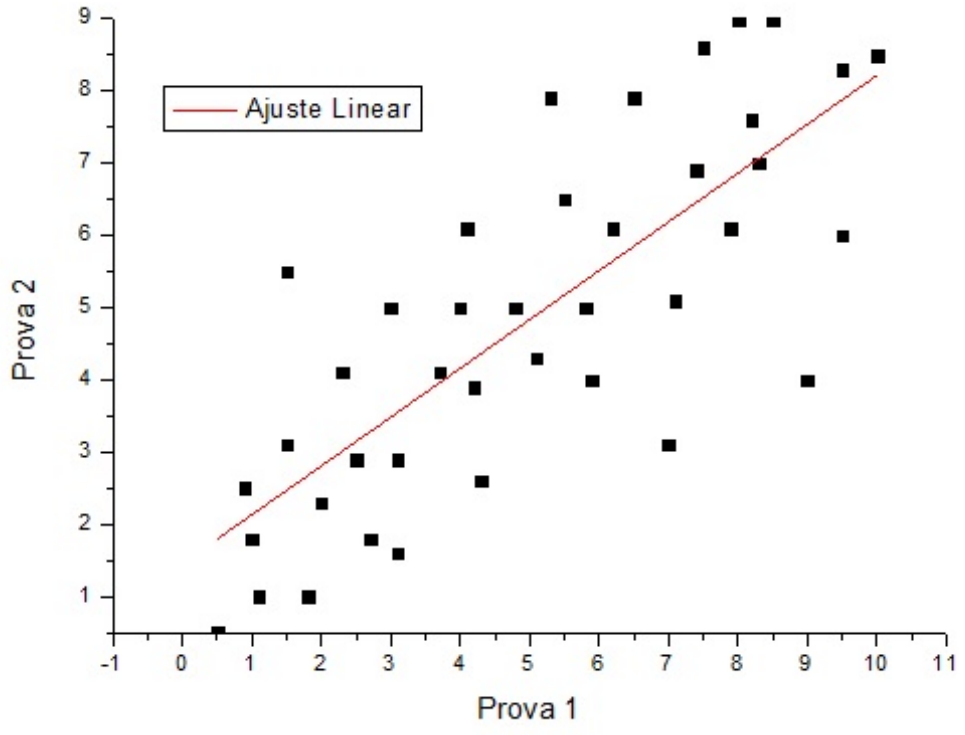


Figura 10: Dispersão das notas semestrais, Prova 1 × Prova 2 e o ajuste linear proposto.

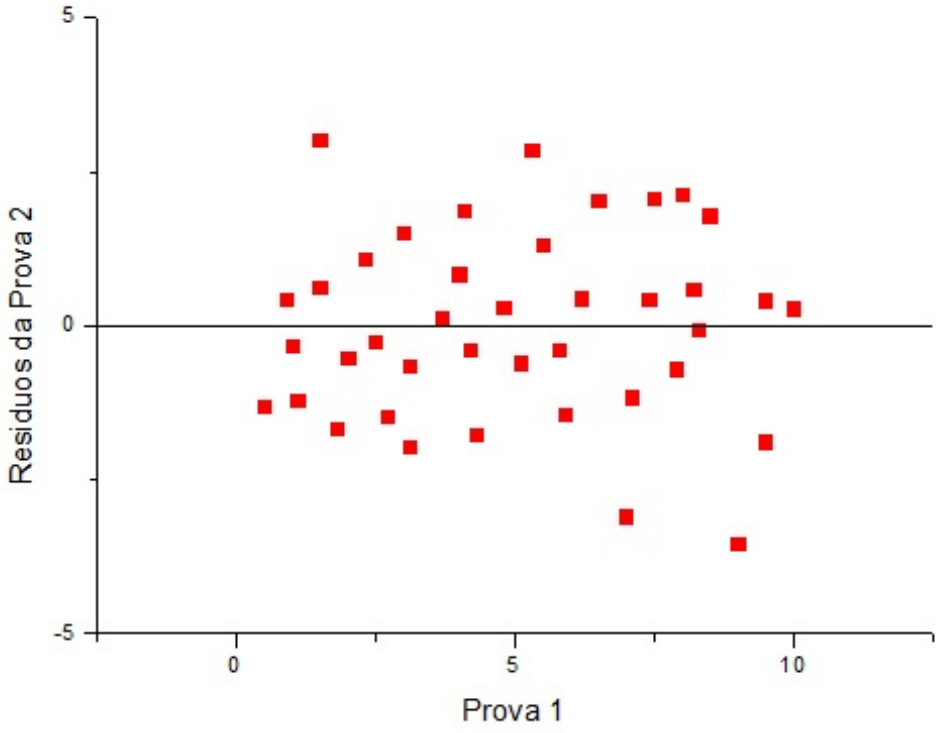


Figura 11: Dispersão dos resíduos apresentando a tendência de normalidade, confirmada mediante teste.

3.4 Modelando o algoritmo genético

População

Para inicialização do algoritmo será gerada uma população de 10 indivíduos (Coleção de genótipos). Os genótipos serão gerados aleatoriamente através de duas amostra de uma distribuição uniforme contínua $U_C(-3,3)$ e $U_C(0,5)$ (A primeira diz respeito ao intercepto e a segunda diz respeito a inclinação, como através de estudos anteriores foi possível constatar que se trata de uma inclinação positiva o intervalo foi restrito de 0 a 5, já para o intercepto a dispersão dos dados denotam um intercepto passando pelos números -3 a 3).

Este par ordenado $(U_C(-3,3), U_C(0,5))$ será calculado para cada um dos 10 indivíduos da população original. A partir desse par ordenado do indivíduo i é

obtida a equação da reta, r_i, e são usados os pontos da primeira prova na reta r_i para obter as notas ajustadas da segunda prova para r_i. Ficando com a sequência de pontos:

$$\left(y_1^i, y_2^i, y_3^i, \cdots, y_{40}^i\right) \tag{3.17}$$

em que $1 \leq i \leq 40$.

Sendo assim, a população inicial é formada por 10 pares ordenados aleatórios obtidos por distribuições uniformes discretas.

Neste trabalho será utilizado o algoritmo apresentado pelo Leung (Algoritmo 1).

O algoritmo 1 apresenta o algoritmo melhorado para um AG. Iniciando pela construção da população $P(\tau)$ em que τ é o número de gerações, sendo assim a população de origem da otimização é dada por $P(0)$.

A população original que foi obtida pelo mero preenchimento estocástico é então submetida a avaliação quanto ao seu valor de fitness, o fitness utilizado no trabalho foi dado por:

$$f_{fitness} = \frac{SQReg}{SQTotal} = \frac{\sum \left(Y_i - \bar{Y}\right)^2}{\sum \left(\hat{Y}_i - \bar{Y}\right)^2} = R^2 \tag{3.18}$$

é uma escolha plausível para a função de fitness, já que $0 \leq R^2 \leq 1$, e quanto maior foi o valor do R^2 maior será o ajustamento da curva aos dados.

Em seguida, o algoritmo entra em um laço condicional em que o processo se repete até a condição de finalização esteja satisfeita. Para esse trabalho a condição de finalização foi a de completar as 4000 evoluções sem mudança no melhor fitness da população.

Em sequência, é realizada a seleção de dois indivíduos, T_a e T_b, da população inicialmente avaliada através do procedimento de roleta ([14]), a probabilidade de

ocorrer seleção é 1 (Em toda geração haverá seleção), logo após dado início ao procedimento de cruzamento, seja T_i o indivíduo i (coleção de notas) da população:

$$C_1 = \frac{T_a + T_b}{2} = \left[\frac{t_{a1} + t_{b1}}{2}, \frac{t_{a2} + t_{b2}}{2}\right] \tag{3.19}$$

em que t_i é a média aritmética entre as taxas i dos indivíduos selecionados no passo anterior, $0 \leq i \leq 2$;

$$C_2 = t_{MAX} \cdot (1 - p) + max\left([t_{a1}, t_{a2}], [t_{b1}, t_{b2}]\right) \cdot p \tag{3.20}$$

em que t_{MAX} é a coleção dos valores de limiar máximo para cada parâmetro avaliado e p é o peso associado, que em questão foi utilizado o valor $p = 0,3$;

$$C_3 = t_{MIN} \cdot (1 - p) + min\left([t_{a1}, t_{a2}], [t_{b1}, t_{b2}]\right) \cdot p \tag{3.21}$$

em que t_{MIN} é a coleção dos valores de limiar mínimos para cada parâmetros avaliado;

$$C_4 = \frac{(t_{MAX} + t_{MIN})(1 - p) + ([t_{a1}, t_{a2}] + [t_{b1}, t_{b2}]) \cdot p}{2} \tag{3.22}$$

uma vez gerados os quatro candidatos a indivíduo de cruzamento, é feita a avaliação de cada um dos candidatos e é escolhido aquele que apresentar maior valor de fitness.

Após escolhermos o indivíduo filho gerado pelo cruzamento, T_c, é dado início ao processo de mutação:

$$T_{mi} = T_c + T_d = [t_{c1}, t_{c2}] + [b_1\Delta_1, b_2\Delta_2] \tag{3.23}$$

em que b_j, $1 \leq j \leq 2$, assume valor 0 ou 1. Δ_k, $1 \leq k \leq 2$ são números aleatoriamente gerados tais a soma do parâmetro t_k e o Δ_k podem assumir valores entre o limiar inferior e superior do parâmetro k. Assim, são geradas 3 mutações:

A primeira, com apenas um b_j assumindo valor 1 e todos os demais igual a 0 (a escolha de qual assumirá o valor 1 é dada através de uma amostra aleatória de uma distribuição uniforme discreta, $U_d(1, 2)$); a segunda, alguns dos b_j escolhidos para assumir valor 1 e os demais valor 0; a terceira, todos os b_j assumem valor 1. Por fim, cada uma das três mutações são avaliadas através da função de fitness.

Em seguida, uma amostra aleatória de uma distribuição uniforme contínua entre 0 e 1 é testada contra a probabilidade de aceitação (que para o trabalho foi definida como 10%), se a amostra for menor que a probabilidade de aceitação, então T_{mi}, $1 \leq i \leq 3$ com maior valor de fitness irá substituir o indivíduo com pior valor de fitness na população. Se a amostra for maior ou igual do que a probabilidade de aceitação, então cada T_{mi} é testado contra o pior indivíduo da população, substituindo-o se tiver maior valor de fitness.

O procedimento se repete a cada geração do algoritmo, a quantidade de gerações foi fixada para um limite de 4000 gerações.

begin
  $\tau \leftarrow 0$ // τ: número de iterações;
  Inicializa $\mathbf{P}(\tau)$ // $\mathbf{P}(\tau)$: população para iteração τ ;
  Avalia $f(\mathbf{P}(\tau))$ // $f(\mathbf{P}(\tau))$: função de fitness ;
  while *(Não atingir condição de parada)* **do**
    begin
      $\tau \leftarrow \tau + 1$;
      Seleciona 2 pais $\boldsymbol{p_1}$ e $\boldsymbol{p_2}$ de $\mathbf{P}(\tau - 1)$;
      Realiza a operação de *cruzamento* pelas expressões 3.18 a 3.21 ;
      Realiza a operação de *mutação* pela expressão 3.22 para gerar 3 filhos nos_1, nos_2 e nos_3 ;
      // reproduz um novo $\mathbf{P}(\tau)$
      if *(Número aleatório* $< p_a$ *// p_a: probabilidade de aceitação)* **then**
        O filho de maior valor da função de fitness entre nos_1, nos_2 e nos_3 substitui o indivíduo com o menor valor da função de fitness da população;
      else
        begin
          if *($f(nos_1)$ > menor valor de fitness na população* $\boldsymbol{P}(\tau - 1)$*)* **then**
            nos_1 substitui o indivíduo com o menor valor de fitness;
          end
          if *($f(nos_2)$ > menor valor de fitness na população* $\boldsymbol{P}(\tau - 1)$*)* **then**
            nos_2 substitui o indivíduo com o menor valor de fitness;
          end
          if *($f(nos_3)$ > menor valor de fitness na população* $\boldsymbol{P}(\tau - 1)$*)* **then**
            nos_3 substitui o indivíduo com o menor valor de fitness;
          end
        end
        Avalia $f(\mathbf{P}(\tau))$
      end
    end
  end
end

Algoritmo 1: Rotina do algoritmo genético melhorado (Leung, 2003).

4 Resultados

Após a realização da simulação, foi possível constatar melhora no valor do R^2 ajustado. A Tabela 7 apresente a população gerada ao final da simulação.

Tabela 7: População otimizada para o algoritmo genético.

Indivíduo	Intercepto	Inclinação	R^2 ajustado
1	0,55422	0,90334	0,56616
2	1,03927	0,75751	0,46018
3	2,9434	0,41097	0,02108
4	0,78138	0,88439	0,22405
5	-0,05289	0,90073	0,3316
6	0,94551	0,79774	0,50012
7	1,51314	0,70645	0,47945
8	1,24087	0,70255	0,58973
9	1,45849	0,67399	0,59015
10	1,45291	0,6758	0,59501

É notável que ao final da simulação o resultado foi satisfatório se comparado ao modelo gerado pela regressão linear. Houve uma sensível diferença quando confrontamos o melhor indivíduo da simulação e o resultado da regressão.

A Figura 12 apresenta a evolução da simulação. Note que o melhor indivíduo da população só se apresenta na geração de número 2797 e que permanece sem melhoramentos até o fim da simulação (O que poderia por sí só determinar o encerramento da simulação) e perceba também o comportamento quase assintótico da curva quando ignorado o fato dela ser uma curva definida por saltos.

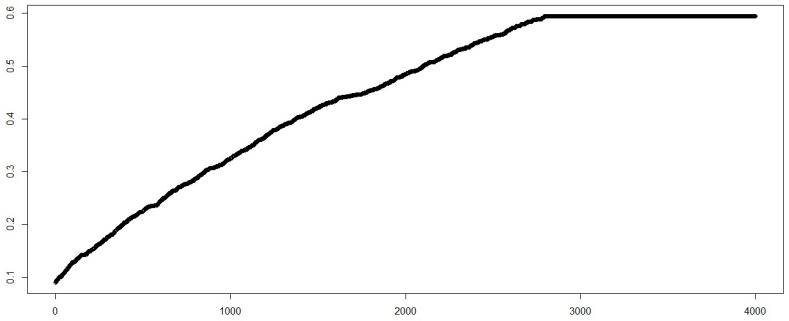


Figura 12: Gráfico da simulaçã: Gerações × Fitness

Isso serve para validar ainda mais o modelo apresentado pela regressão. As Figuras 13 e 14 apresentam, respectivamente, o ajuste linear apresentado pelo melhor indivíduo da população (indivíduo 10) e a dispersão dos resíduos.

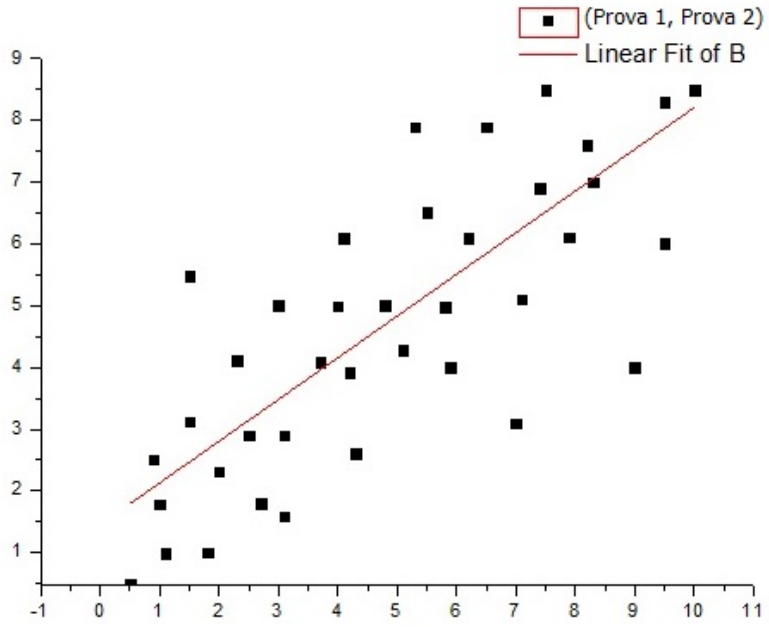


Figura 13: Ajuste linear proposto pelo indivíduo 10.

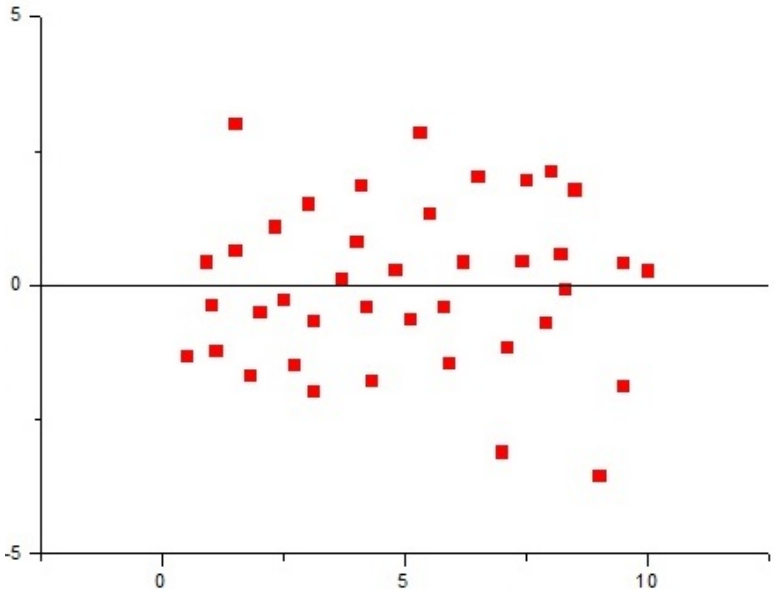


Figura 14: Dispersão dos resíduos indivíduo 10.

Pode-se notar a eficácia do método para o ajustamento de curvas. O propósito do trabalho não visa propor um método para substituir o tão bem estabelecido método de regressão linear, o principal propósito é apresentar a alternativa e confrontar os resultados quantificando-os.

A abordagem se mostra salutar como capítulo a parte de um curso de estatística e/ou cálculo numérico.

A interdisciplinaridade apresentada pela alternativa ainda levanta alguns tópicos interessantes para discussão como a integração acadêmica entre áreas como: Computação, Biologia, Estatística e Matemática. Isso posto, o enriquecimento cultural é enorme não somente para o alunado, como também para os professores envolvidos na atividade.

Referências

[1] AL SALAMI, N. M. A., Ant Colony Optimization Algorithm, **UbiCC Journal**, v. 4, n. 3, p. 823-826,2009.

[2] ALANDER, J. T., On optimal population size of genetic algorithms, In: CompEuro '92 - COMPUTER SYSTEMS AND SOFTWARE. The Hague, Netherlands, **Proceedings**, 1992, p. 65-70.

[3] ARABAS, J., MICHALEWICZ, Z., MULAWKA, J., GAVaPS-a genetic algorithm with varying population size, In: Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence. Orlando, FL, **Proceedings of the First IEEE Conference on Evolutionary Computation**, 1994, p. 73-78 v.1.

[4] APPLEGATE, D. L.; BIXLY, R. E.; CHVÁTAL, V.; COOK, W. J., **The Traveling Salesman Problem: A Computational Study**, Princeton University Press, 2006, 606 p.

[5] Algoritmo genético - Acessado em 16/08/2014: http://pt.wikipedia.org/wiki/Algoritmo_gen%C3%A9tico.

[6] BAKER, J. E., Reducing bias and inefficiency in the selection algorithm. In. GREFENSTETTE, J. J. Eds., **Proceedings of the 2nd International Conference on Genetic Algorithms and Their Applications**. Lawrence Erlbaum, Hillsdale, New Jersey, 1987, p. 14-21.

[7] BLICKLE, T., THIELE, L., A comparison of selection schemes used in genetic algorithms. **Tech. Rep. TIK Report 11**, December 1995, Computer Engineering and Communication Networks Lab, Swiss Federal Institute of Techmology, 1995.

[8] BREMERMANN, H.J. Optimization Through Evolution and Recombination, In M.C. Yovits et al, Editors, **Self-Organizing Systems**, Spartan Books, Washington, DC: 1962, p. 93-106.

[9] Capítulo 9 - Regressão Linear e correlação - Prof. Luiz Alexandre Peternelli, acessado em 16/08/2014: http://www.dpi.ufv.br/ peternelli/inf162.www.16032004/materiais/CAPITULO9.pdf.

[10] CASELLA, G., BERGER, R. L., **Statistical Inference**. Cengage Learning, 2nd edition, 2001, 660p.

[11] Coeficiente de Correlação Pearson - Acessado em 16/12/2014: http://pt.wikipedia.org/wiki/Coeficiente_de_correla%C3%A7%C3%A3o_de_Pearson

[12] DAWKINS, R. **O relojoeiro cego. A teoria da evolução contra o desígnio divino**. Editora Companhia das Letras. 1991, 488p.

[13] DE JONG, K. A., **An Analysis of the Behaviour of a Class of Genetic Adaptive Systems**, Tese (Doctor of Philosophy), University of Michigan, 1975.

[14] EIBEN, A. E., SMITH, J. E., **Introduction to Evolutionary Computing** (Natural Computing Series), Springer, 2008, 315 p.

[15] EIBEN, A. E., AARTS, E. H. L., VAN HEE, K. M., Global convergence of genetic algorithms: a Markov chain analysis, In: SCHWEFEL, H. -P., MÄNNER, R., Eds., **Proceedings of the 1st Conference on Parallel Problem Solving from Nature n. 496 in Lecture Notes in Computer Science**, Springer, Berlin, Heidelberg, New York, 1991.

[16] FAN, W., FOX, E. A., PATHAK, P., WU, H., The Effects of Fitness Functions on Genetic Programming-Based Ranking Discovery For Web Search, **Journal of the American Society for Information Science and Technology**, v. 55, n. 7, p. 628-636, 2004.

[17] FOGEL, D. B., **Evolutionary Computation: The Fossil Record**, IEEE press, Piscataway, NJ, 1998, 656 p.

[18] FOGEL, L. J., OWENS, A. J., WALSH, M. J., Artificial intelligence through a simulation of evolution. In: CALLAHAN, A., MAXFIELD, M., FOGEL, L.J., Eds., **Biophysics and Cybernetic Systems**. Spartan, Washington DC, p. 131-156, 1965.

[19] FOGEL, L. J., OWENS, A. J., WALSH, M. J., **Artificial Intelligence through Simulated Evolution**. Wiley, Chichester, UK, 1966.

[20] Gráfigo de dispersão, acessado em 16/08/2014: http://wikiciencias.casadasciencias.org/wiki/images/4/44/Img_Diagrama_ou_gr%C3%A1fico_de_dispers%C3%A3o_Grafico_1.png.

[21] GOLDBERG, D. E., DEB, K., A Comparative Analysis of Selection Schemes Used in Genetic Algorithms, **Foundations of Genetic Algorithms**, p 69-93. 1991.

[22] HOLLAND, J. H., Genetic algorithms and the optimal allocation of trials, **SIAM J. of Computing**, v. 2, p. 88-105, 1973.

[23] HOLLAND, J. H., **Adaption in Natural and Artificial Systems**, MIT Press, Cambridge, MA, 1992, 211 p.

[24] KARABOGA, D., BASTURK, B., A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, **Journal of Global Optimization**, v. 39, n. 3, p. 459-471, 2007.

[25] KENNEDY, J., EBERHART, R. Particle swarm optimization. In Proc. of the **IEEE Int. Conf. on Neural Networks**, Piscataway, NJ., 1995. p. 1942-1948.

[26] KOZA, J. R., **Genetic Programming**, MIT Press, Cambridge, MA, 1992, 840 p.

[27] KOZA, J. R., **Genetic Programming II**, MIT Press, Cambridge, MA, 1994, 768 p.

[28] LARRAÑAGA, P., KUIJPRES, C. M. H., MURGA, R. H., INZA, I., DIZDAREVIC, S., Genetic Algorithms for the Travelling Salesman Problem: A Review of Representations and Operators, Artificial Intelligence Review, v. 13, p. 129-170, 1999.

[29] LEUNG, F. H. F. **Tuning of the structure and parameters of a neural network using an improved genetic algorithm**, IEEE transactions on neural networks, V. 14, N. 1, pp. 79-88, 2003.

[30] LEWIN, B., **Genes VII**. Oxford University Press, New York, 2000, 990 p.

[31] Método mínimos quadrados - Acessado em 16/08/2014: http://pt.wikipedia.org/wiki/M%C3%A9todo_dos_m%C3%ADnimos_quadrados.

[32] RIDLEY, M. **Evolution**. 3rd edition, Blackwell Publishing, 2004, 751p.

[33] SHAPIRO., Tabela para o teste Shapiro-Wilk - Acessado em 16/12/2014: http://docentes.esa.ipcb.pt/estatistica/apontamentos/Testes_Ajustamento.pdf

[34] SCHWEFEL, H. -P., **Numerical Optimisation of Computer Models**, Wiley, New York, 1981, 398 p.

[35] SCHRAUDOLPH, N. N., BELEW, R. K., Dynamic Parameter Encoding for Genetic Algorithms, **Machine Learning**, v. 9, p. 9-21, 1992.

[36] SRINIVAS, M., PATNAIK, L. M., Adaptive probabilities of crossover and mutation in genetic algorithms, **IEEE Transactions on Systems, Man and Cybernetics**, v. 24, n. 4, p. 656-667, 1994.

[37] TELLES, M. P. C., DINIZ-FILHO, J. A. F., COELHO, A. S. G., CHAVES,L. J., Autocorrelação espacial das freqüências alélicas em subpopulações de cagaiteira (*Eugenia dysenterica DC., Myrtaceae*) no sudeste de Goiás, **Revista brasil. Bot.**, v. 24, n. 2, p. 145-154, 2001.

[38] TURING, A. M. (1948), Intelligent Machinery, In: Mechanical Intelligence, Collected Works, D. C. Ince ed., Amsterdam : North-Holland, 1992, p. 49. As to initiative, it is examplified by the research of algorithms for open arithmetical problems.

[39] WHITNEY, L. D., KAUTH, J., Genitor: A different genetic algorithm. In: **Proceedings of the Rocky Mountain Conference on Artificial Intelligence**. 1988, p. 118-130.