UNIVERSIDADE FEDERAL DO MARANHÃO

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIAS - CCET

COORDENAÇÃO DO CURSO DE ENGENHARIA DA COMPUTAÇÃO

IMPLEMENTAÇÃO E ANÁLISE ALGORÍTMO DE REGRESSÃO LINEAR

Prof. Dr. THALES LEVI AZEVEDO VALENTE

Discente: CAIO REIS BATISTA

São Luís-MA

2025

**Introdução**

Este documento apresenta a análise dos experimentos realizados com o algoritmo de regressão linear, focando especificamente no impacto da taxa de aprendizado (α) e da inicialização dos parâmetros (θ) no comportamento do algoritmo de descida do gradiente. A taxa de aprendizado (α) é um hiperparâmetro crucial no algoritmo de descida do gradiente, influenciando diretamente a velocidade e a qualidade da convergência. Nos experimentos, foram testados três valores distintos: α = 0.001, 0.01 e 0.1.

**Metodologia**

Os experimentos foram conduzidos utilizando um conjunto de dados que relaciona a população de cidades com seus respectivos lucros. Foram utilizados e aplicados o algoritmo de regressão linear utilizando Python para gerar e plotar gráficos e imagens para a análise.

**Resultados e Discussões**

1. Variação da Taxa de Aprendizado (**α**)

Foram testados três valores diferentes (0.001, 0.01 e 0.1)

1.1 Comportamento com **α** muito pequeno (0.001)**:**

* A convergência é mais lenta, necessitando de mais interações para atingir o mínimo.
* O algoritmo é mais estável e tem menor risco de divergir.
* Pode ser computacionalmente mais custoso devido ao maior número de iterações necessárias.

1.2 Comportamento com **α** muito grande(0.1)

* A convergência é mais rápida inicialmente
* Pode apresentar oscilações ao redor do mínimo
* Risco de divergência se o valor for muito alto
* Pode “pular” o mínimo global, resultando em uma solução sub-ótima

1.3 Valor intermediário (0.01)

* Oferece um bom equilíbrio entre velocidade e estabilidade
* Convergência consistente sem oscilações significativas
* Recomendado como ponto de partida para este problema específico

Quando α é muito pequeno (0.001), o algoritmo leva muito tempo para convergir, pois os passos na direção do gradiente são muito pequenos. Por outro lado, quando α é muito grande (0.1), o algoritmo pode oscilar excessivamente ou até mesmo divergir em alguns casos, pois os passos são muito grandes e podem ultrapassar o mínimo da função de custo.

**Visualização da Convergência**

Gráfico

O conteúdo gerado por IA pode estar incorreto.

Figura 1: Gráfico mostrando a convergência da função de custo para diferentes taxas de aprendizado (α). Note como α=0.001 converge muito lentamente, α=0.01 oferece convergência equilibrada, e α=0.1 diverge rapidamente, levando o custo a explodir.

2. Análise da Inicialização dos Pesos

A escolha dos valores iniciais dos pesos (θ) pode impactar significativamente o processo de otimização. Foram testadas três inicializações fixas ([0,0], [5,5], [-5,5]) e três inicializações aleatórias.3. Impacto da Inicialização:

3.1. Inicialização próxima ao ótimo ([0,0]):

   - Convergência mais rápida

   - Menor número de iterações necessárias

   - Trajetória mais direta ao mínimo global

3.2. Inicialização distante ([5,5] ou [-5,5]):

   - Requer mais iterações para convergir

   - Trajetória mais longa até o mínimo global

   - Maior custo computacional

3.3. Inicializações aleatórias:

   - Demonstram a robustez do algoritmo

   - Diferentes trajetórias, mas convergência ao mesmo ponto

   - Útil para evitar mínimos locais em problemas mais complexos

Gráfico

O conteúdo gerado por IA pode estar incorreto.

Figura 2: Contorno da função de custo mostrando as trajetórias do gradiente para diferentes inicializações. As linhas coloridas mostram o caminho percorrido pelo algoritmo a partir de cada ponto inicial (marcado com círculo) até a convergência. Note como todas as trajetórias eventualmente convergem para o mesmo ponto mínimo, independente do ponto de partida.

 4. Relação com Fine-tuning em Redes Neurais4.1. Convergência: Uma boa inicialização pode levar a uma convergência mais rápida e estável.

4.2. Mínimos Locais: Em problemas mais complexos, a inicialização pode determinar em qual mínimo local o algoritmo irá convergir.

4.3. Eficiência Computacional: Inicializações adequadas podem reduzir significativamente o tempo de treinamento.

A análise da inicialização dos pesos tem uma relação direta com o conceito de fine-tuning em redes neurais:

1. Transferência de Conhecimento:

   - Em fine-tuning, iniciamos com pesos pré-treinados em vez de aleatórios

   - Similar a iniciar mais próximo do ótimo em nossa regressão linear

   - Reduz o tempo de treinamento e melhora a convergência

2. Estabilidade:

   - Inicialização adequada evita problemas de vanishing/exploding gradients

   - Permite ajustes mais suaves e controlados dos pesos

   - Fundamental para redes profundas e complexas

3. Generalização:

   - Inicialização apropriada pode levar a soluções que generalizam melhor

   - Evita convergência para mínimos locais sub-ótimos

   - Importante tanto em regressão linear quanto em redes neurais complexas

 Conclusão

Os experimentos demonstram que tanto a taxa de aprendizado quanto a inicialização dos parâmetros são cruciais para o desempenho do algoritmo de descida do gradiente. Uma taxa de aprendizado adequada (α = 0.01) oferece o melhor equilíbrio entre velocidade e estabilidade de convergência. Quanto à inicialização dos parâmetros, o algoritmo mostrou-se robusto o suficiente para convergir a partir de diferentes pontos iniciais, embora algumas inicializações possam levar a convergência mais rápida. Uma recomendação prática é começar com = 0.01 e ajustar conforme necessário e usar inicialização aleatória para problemas mais completos além de monitorar a convergência através das curvas de custo.