Inteligência Computacional - 2019.1
Relatório do Primeiro Trabalho

Juliana Franco Ibiapina[1]

[1] Universidade Federal do Ceará (UFC) – Sobral, CE – Brasil

Os três problemas dados no trabalho foram: achar o máximo de uma função pelo algoritmo *Hill-Climbing*, implementar uma função baseada em Lógica *Fuzzy* para resolver uma situação previamente explicada e implementar um algoritmo que determine os modelos de regressão polinomial de graus 2, 3, 4 e 5 para os dados de um aerogerador. Para isso, foi escolhida a linguagem de programação Python, pela familiaridade da estudande com a mesma e pela sua facilidade em trata matrizes através da biblioteca Numpy e geração de gráficos pela biblioteca Matplotlib.

No primeiro problema foi dada a função $f(x, y) = |x \sin(y\pi/4) + y \sin(x\pi/4)|$ para que se ache o máximo da mesma, onde $x, y \in [0, 20]$. O programa inicia em um ponto inicial $p[x, y]$, onde x e y são valores aleatórios pertencentes ao intervalo especificado. Depois, os vizinhos desse ponto são calculados conforme foi exemplificado na definição do problema. O valor da função para o ponto e seus vizinhos é calculado, e então o algoritmo escolhe o ponto cujo valor da função seja máximo, se esse valor for o do ponto dado, então o programa para e o máximo local já foi encontrado. Mas, se o ponto cujo valor máximo é em um dos vizinhos, esse vizinho passa a ser o ponto atual e o altoritmo repete todos os passos anteriormente explicados até que o próprio ponto seja o máximo em relação a seus vizinhos. Essa foi a forma como o algotimo *Hill-Climbing* foi implementado e sua limitação consiste em achar apenas um máximo local da função.

A segunda questão consiste em implementar uma função baseada em lógica *fuzzy* que simule o sistema de freio de um carro, cujos valores de entrada são pressão no pedal, velocidade da roda e velocidade do carro, e valor de saída a pressão a ser aplicada no freio. Todas as variáveis de entrada possuem valores pertencentes a um intervalo de 0 a 100. A base para o algoritmo é a Inferência de Mamdani, portanto foram definidos três conjuntos nebulosos para cada uma das variáveis de entrada bem como suas funções de pertinência. Os conjuntos são: alto, médio e baixo para a variável pressão no freio, devagar, médio e rápido para cada uma das outras duas variáveis.

Depois de calculados os valores de petinência das variáveis de entrada, as regras 1, 2, 3 e 4, definidas no problema, foram aplicadas resultando em quatro valores. As regras 1 e 2 geram valores para ”aperte o freio”e as regras 3 e 4 produzem valores para ”libere o freio”. Essas duas ações foram definidas matematicamente como duas funções, da forma como foi mostrado nos slides. Cada valor para ”aperte o freio”é somado, bem como cada um referente a ”libere o freio”, resultando em dois valores. O passo seguinte é usas esses valores para limitar as funções aperte o freio e libere o freio. Isso foi implementado gerando amostras dessas funções, que variam em uma uma unidade. O passo de limitar as funções aos dois valores gerados pelas regras foi realizado comparando cada amostra ao valor limite referente a sua função, que quando é ultrapassado recebe no lugar o própio valor limitante. Depois disso ser aplicado a cada função a união entre as duas é feita pegando-se o máximo entre elas. O passo seguinte é calcular o centro de gravidade, como é especificado nos slides. O resultado final é a pressão que deve ser aplicada no freio, a qual é a saída da função.

No terceito problema, os dados de um aerogerador devem ser usados para gerar os modelos de regressão polinomial de graus 2, 3, 4 e 5. O programa também criou o modelo de regressão linear. Para realizar as operações matriciais foi usada a biblioteca Numpy. O primeiro passo é criar as matrizes X para os graus 2, 3, 4 e 5. Depois disso, a matriz dos coeficientes $\hat{\beta}$ para cada um dos graus desejados de polinômio, a qual é definida como:

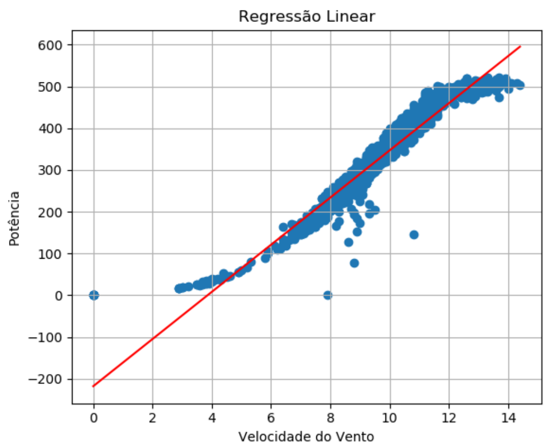
$$\hat{\beta} = (X^T X)^{-1} X^T y$$

Dessa forma o modelo de predição para cada grau pôde ser calculado como:

$$\hat{y} = X\hat{\beta}$$

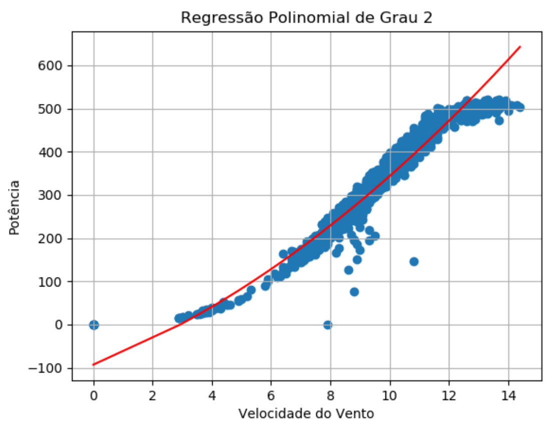
Os gráficos referentes a cada modelo são mostrados a seguir.

Figura 01



Fonte: Elaborado pelo autor.

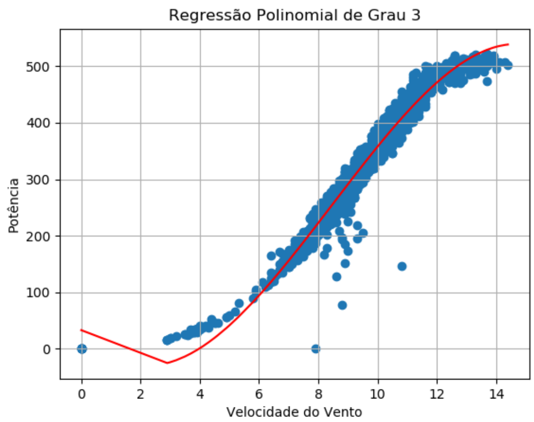
Figura 02



Fonte: Elaborado pelo autor.

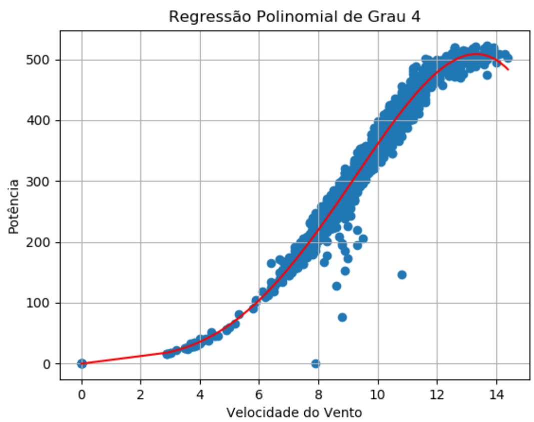
Analisando os modelos pode-se perceber que a medida que o grau da regressão polinomial aumenta o modelo se adequa melhor aos dados, porém perce-se que essa melhoria possui um limite. Isso pode ser obervado na regressão polinomial de grau 4 e 5,

Figura 03



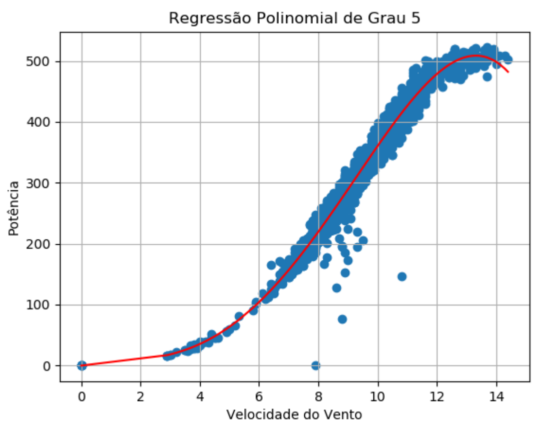
Fonte: Elaborado pelo autor.

Figura 04



Fonte: Elaborado pelo autor.

Figura 05



Fonte: Elaborado pelo autor.

onde a melhoria entre os dois modelos é muito pequena. Isso pode ser melhor observado através do cálculo do coeficiente de determinação e do coeficiente de determinação ajustado, descritos na Tabela 1 a seguir.

Tabela 1. Comparação entre os coeficientes

Grau da regressão polinomial	R^2	R^2_{aj}
2	0.943424	0.943374
3	0.969023	0.968982
4	0.973724	0.973677
5	0.973726	0.973667