



UNIVERSIDADE DE SÃO PAULO  
INSTITUTO DE CIÊNCIAS MATEMÁTICAS E DE  
COMPUTAÇÃO  
DEPARTAMENTO DE CIÊNCIAS DE COMPUTAÇÃO



JOSÉ MARIA CLEMENTINO JUNIOR      11357281

## ***Projeto 3– Programação Dinâmica***

## **Apresentação**

Analisar séries temporais, medir a similaridade não é uma tarefa trivial, deste modo vem sendo desenvolvidas técnicas capazes de lidar com dados discretos que sofrem variações por meio do tempo e que possuem diferentes características dos dados mais simples. Para isso, o algoritmo [Dynamic Time Warping \(DTW\)](#) foi proposto, ele é capaz de encontrar e comparar o alinhamento entre duas séries temporais, sendo assim, o DTW é fortemente utilizado para realizar a classificação de sequências que se encontram em diferentes frequências.

Portanto o objetivo deste trabalho é desenvolver um algoritmo capaz de realizar a classificação entre as séries temporais, por meio do uso da distância de DTW, à partir dos dados encontrados por acelerômetros, mais especificamente extraídos de movimentos realizados por um controle de videogame Wii.

## **Execução:**

Para executar o código, após a instalação do NodeJS, basta executar o comando “**node main.js < teste.txt**” no prompt de comando. De modo que é responsável por executar a classificação dos pontos presentes no arquivo “teste.txt” com os pontos contidos no arquivo “treino.txt”.

**Implementação:** A implementação divi-se nos seguintes arquivos:

- **./main.js** : Arquivo principal que lê o arquivo “treino.txt” e faz a comparação do vizinho mais próximo com o arquivo lido do stdin. Após encontrar o vizinho mais próximo para cada entrada do arquivo de teste o programa salva um relatório em um arquivo de texto com um resumo do processamento.
- **./node\_modules/dtw.js**: Este arquivo define a classe “dtw”. Na qual é responsável pela comparação das sequências temporais. A implementação baseia-se no algoritmo disponível na [Wikipédia](#), no qual foi adaptado para atender as especificações proposta pelo trabalho.

**./node\_modules/utls.js**: Neste arquivo contém a função “*saveToFile*” responsável por guardar os arquivos gerados em um arquivo chamado “\_Result.csv”.

## Análise dos Resultados e Conclusões

Com o intuito de obter os resultados, por meio da acurácia deve-se classificar cada uma das séries temporais presentes no arquivo “treino”. Em seguida após obter os resultados, fez-se necessário realizar a comparação com o tipo de movimento utilizado presente no arquivo “teste”, deste modo para os movimentos distintos encontrados, fica evidente que aconteceu um erro na classificação da série analisada. A taxa de acurácia obtida foi de **84.4791%**. A matriz de confusão é apresentada na Figura 1, na qual as linhas referem-se ao tipo de movimento de teste apresentando a quantidade de cada tipo de movimento ao qual foram classificados os possíveis movimentos(coluna).

Deste modo é possível observar quais movimentos foram classificados corretamente e erroneamente, os movimentos que mais foram classificados incorretamente são os seguintes:

- ❖ **Movimento 6** classificado como movimento **11** e **1**;
- ❖ **Movimento 9** classificado como movimento **1** e **11**;
- ❖ **Movimento 11** classificado como movimento **6**;
- ❖ **Movimento 12** classificado como movimento **7**;
- ❖ Entre outros resultados que podem ser observados na Figura 1.

	Movimentos Preditos											
Movimentos	1	2	3	4	5	6	7	8	9	10	11	12
1	72	0	0	0	0	0	0	0	2	0	0	0
2	0	78	0	0	0	0	0	0	0	0	0	0
3	0	0	76	1	0	0	0	1	0	0	0	0
4	0	0	0	70	0	1	0	4	0	4	0	0
5	0	0	0	0	86	0	0	0	0	0	0	0
6	10	0	0	0	0	56	1	0	0	0	16	0
7	0	6	0	0	0	0	63	0	0	3	0	2
8	0	0	0	1	0	2	0	76	0	3	0	0
9	20	0	0	0	0	0	0	0	60	0	10	0
10	0	0	0	0	0	0	9	1	0	69	0	1
11	1	0	0	2	0	18	0	4	0	0	53	0
12	0	7	0	0	0	0	16	0	0	3	0	52

Figura 1: Matriz de Confusão

Também é possível afirmar que os movimentos que tiveram as maiores taxas de acertos foram os **Movimentos 5, 2, 3 e 10**.

De modo geral, o algoritmo proposto acertou **811** movimentos e errou **149**, com uma taxa de acerto de **84.4791%** e uma taxa de erro de **15,5209%** em um total de

**960**(100%) movimentos. O tempo médio de execução encontrado foi de 8,568 segundos. O arquivo de resultado pode ser encontrado no seguinte [repositório](#) juntamente com a implementação. Na imagem a seguir são apresentados os resultados aproximados e também é apresentado o tempo médio para realizar a classificação de um movimento 9,3 (ms), os resultados de tempo são apresentados na Figura 2 estão na medida milissegundos.

Resultados			
Nº Acertos	Taxa Acertos	Nº Erros	Taxa Erros
811	84.50%	149	15.50%
Tempo Médio para Classificar um Ponto (ms)			
	9.3		
Tempo Total de Execução(ms)			
	8973		

Figura 2: Resultados

Referências: [https://en.wikipedia.org/wiki/Dynamic\\_time\\_warping](https://en.wikipedia.org/wiki/Dynamic_time_warping)