UNIVERSIDADE CATÓLICA DE PELOTAS
Mestrado em Engenharia Eletrônica e Computação

Intelligent System for Identification of Wheelchair User's Posture using Machine Learning Techniques

Disciplina: Machine Learning

Pedro Tavares

Prof. Mateus Grellert

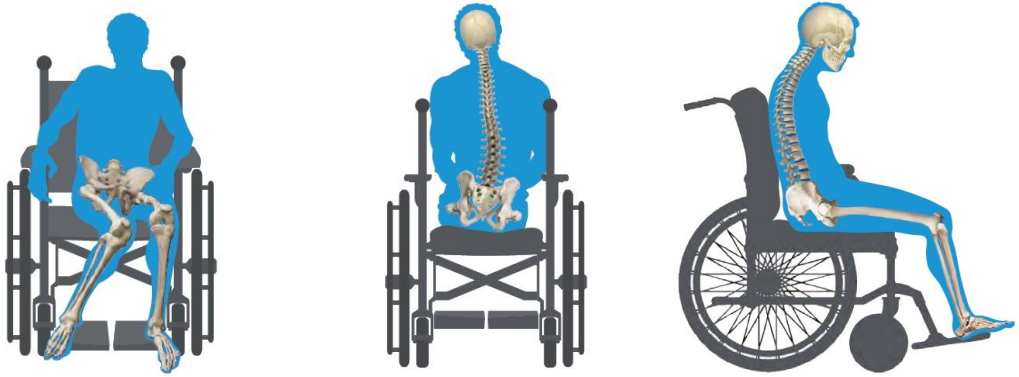
Pelotas, 18 de outubro de 2019

Introdução

- Usuários de cadeiras de rodas podem possuir limitações físicas ou mentais que dificultam suas atividades cotidianas;
- Em todo o mundo aproximadamente 15% das pessoas possuem algum tipo de problema de mobilidade;
- Estudos apontam que o mercado de cadeiras manuais cresceu de U$ 1.8 bilhões em 2011 para U$ 2.9 bilhões em 2018;
- Quanto mais aumenta o número de usuários de cadeiras de rodas, aumentam também o número de ocorrências de problemas de saúde relacionados ao *seating;*

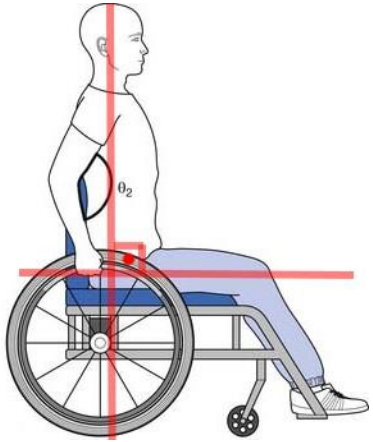
Introdução

- Má postura em cadeira de rodas pode causar: dor crônica, esclerose, cifose, problemas respiratórios e de pele, rigidez muscular, fadiga, dores, entre outros;
- Em alguns casos, o usuário não possui sensação de dor e permanece em má postura até que sua saúde esteja seriamente comprometida;
- Postura correta acarreta em múltiplos benefícios: redução de dores, diminuição de ocorrência de úlceras, entre outros.



Introdução

- Vários estudos demonstram que a posição de 90º entre tronco e coxas é a melhor para tarefas do dia-a-dia.
- Não há no mercado nenhum sistema que classifique posturas em tempo-real para usuários de cadeiras de rodas;

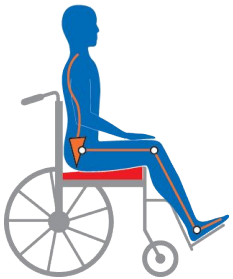


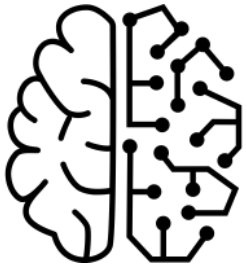
UNIVERSIDADE CATÓLICA DE PELOTAS

Objetivo

- Apresentar um sistema inteligente capaz de detectar e classificar a postura de um usuário de cadeira de rodas, com o objetivo de corrigir posturas incorretas em tempo-real evitando, assim, lesões de maior gravidade.

- Para a construção desse sistema uma rede de sensores foi utilizada para capturar os dados e treinar um dispositivo embarcado que, através do algoritmo k-NN (k-Nearest Neighbors), consegue classificar más posturas em tempo hábil e sugerir correções ao cadeirante.

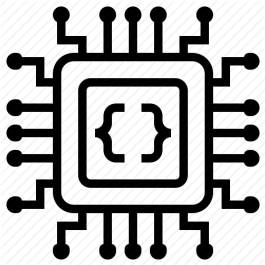




UNIVERSIDADE CATÓLICA DE PELOTAS

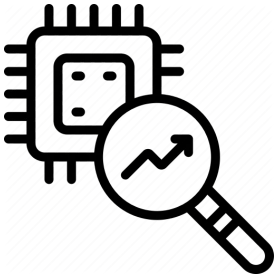
Etapas do Projeto

1. Desenvolvimento do Hardware

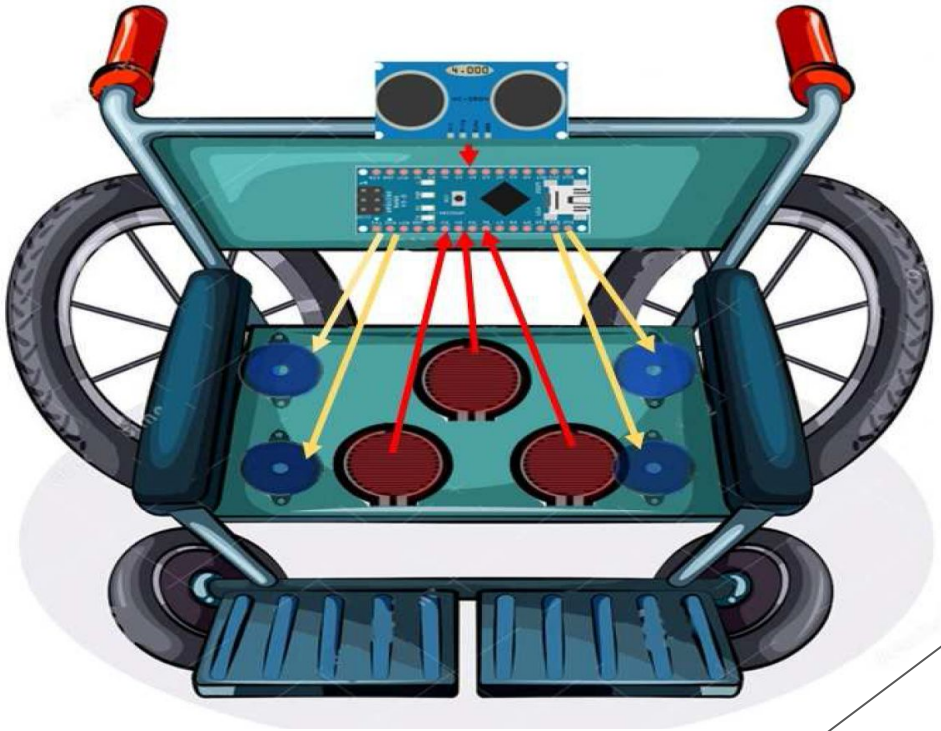


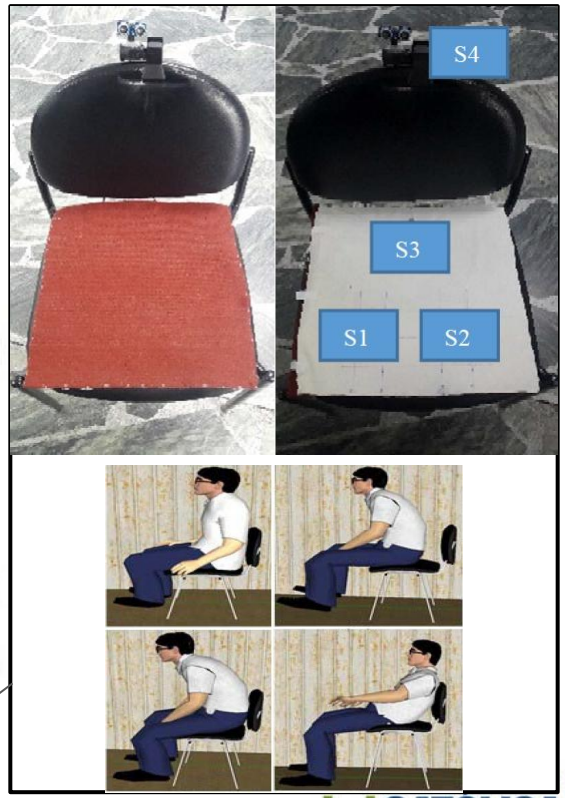
2. Configuração da Base de Dados

3. Análise de Dados



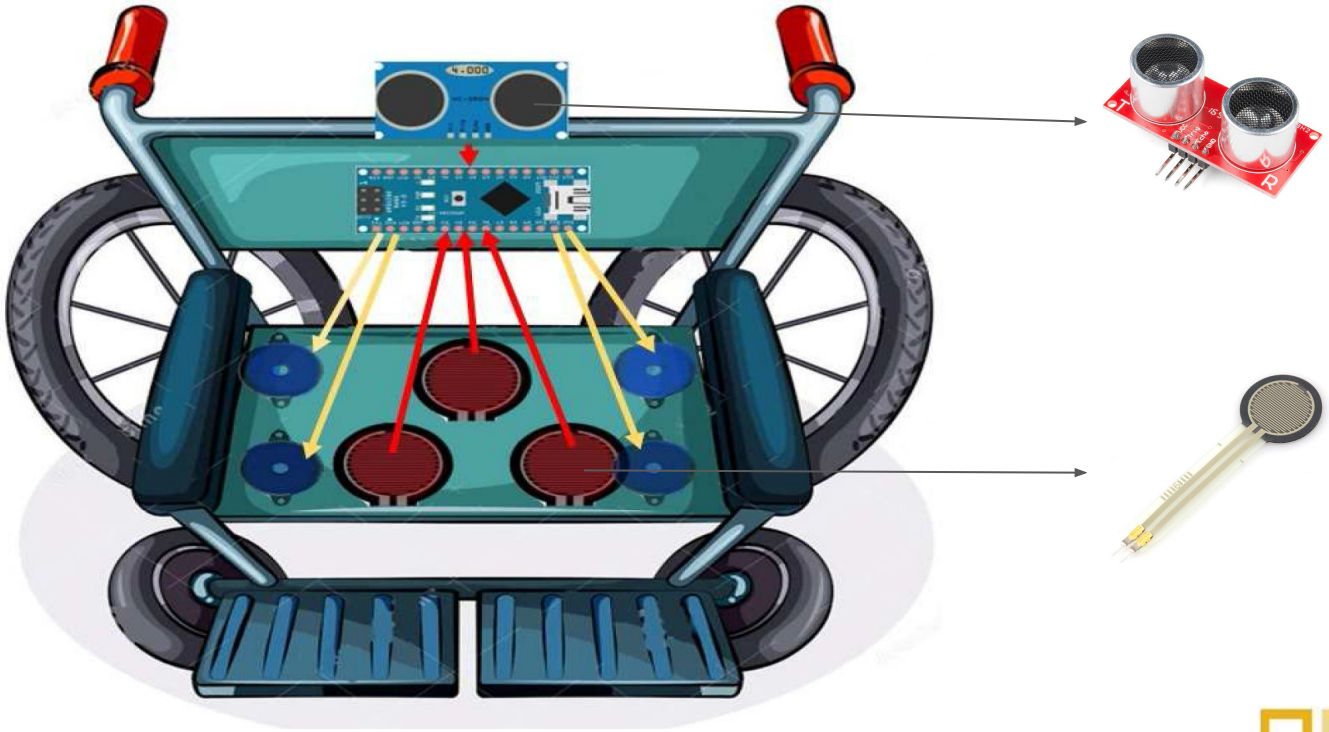
1. Projeto do Hardware do Sistema



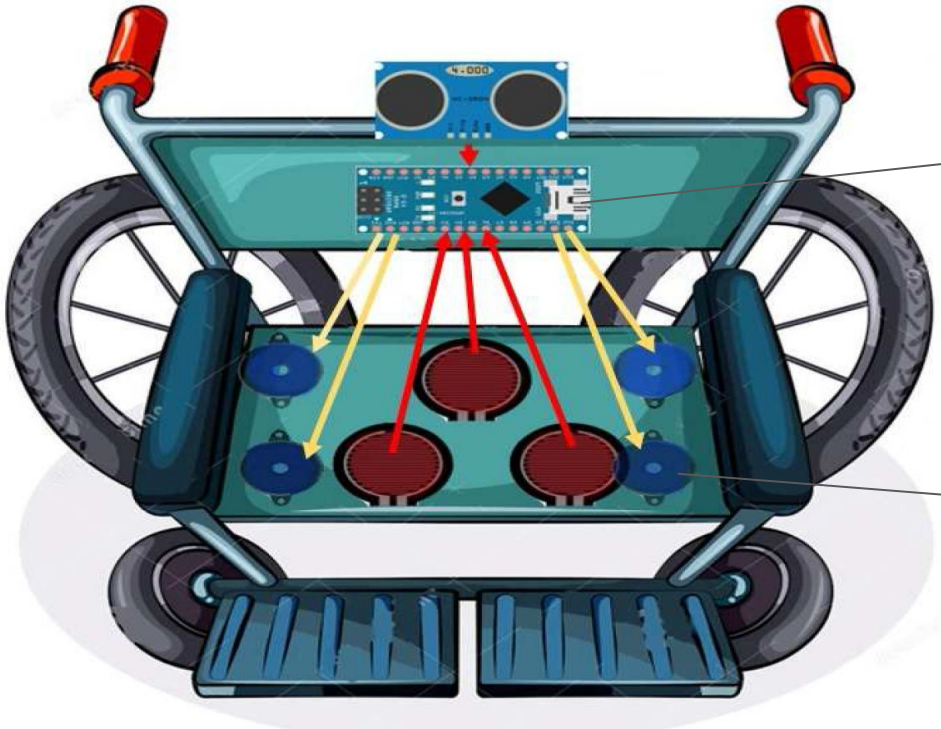


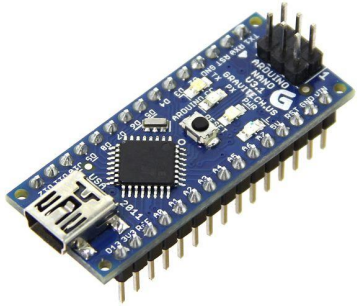
Trabalho relacionado: “Human-sitting-pose detection using data classification and dimensionality reduction,” in 2016 IEEE Ecuador Technical Chapters Meeting (ETCM), Oct 2016, pp. 1–5.

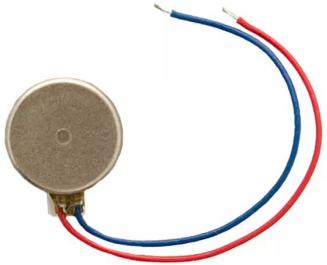
1. Projeto do Hardware do Sistema



1. Projeto do Hardware do Sistema

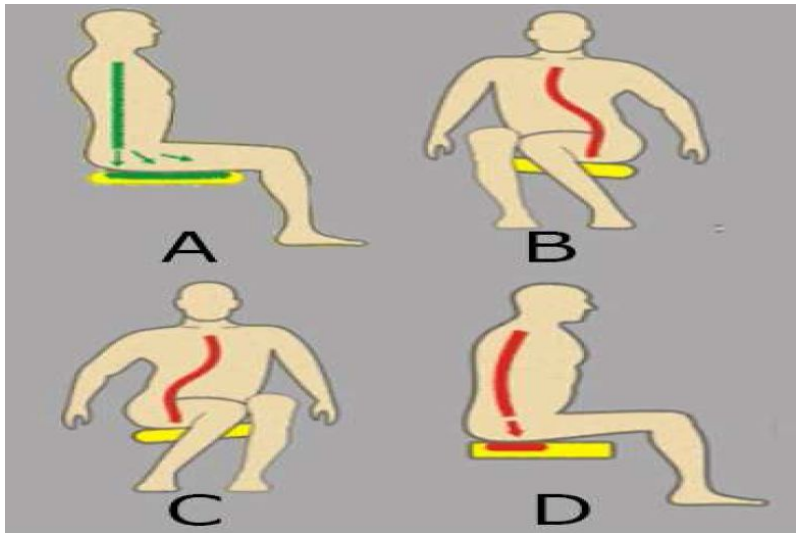






2. Base de Dados

Para resolver o problema da classificação, foi utilizada uma taxonomia típica recomendada por médicos e fisioterapeutas acerca das posturas sentadas que mais afetam a coluna vertebral.



A - **Postura correta** (sem danos)
B - **Maior pressão no lado direito** (problemas respiratórios e musculares, pressão no fígado, estômago e rim direito)
C - **Maior pressão no lado esquerdo** (problemas respiratórios e musculares, pressão no baço e rim esquerdo)
D - **Maior pressão para frente** (problemas nos joelhos, dor nas costas e estresse no abdômen)

2. Base de Dados

Método de aquisição de dados:

- 5 indivíduos;
- Cada um sentava nas 4 posições (A, B, C e D) por 2 minutos;
- Após 2 minutos em uma posição “ruim” os motores de vibração próximos aos sensores de maior pressão eram ativados;
- Após usuário corrigir a postura, caso o sistema identificasse que o mesmo estava na posição A, os motores eram desligados;

- Cada um dos indivíduos repetiram 25 vezes cada posição;
- Como o número de sensores = 4:

Total de Amostras = 2000 pts (5 x 4 x 25 x 4)

UNIVERSIDADE CATÓLICA DE PELOTAS

3. Análise de Dados

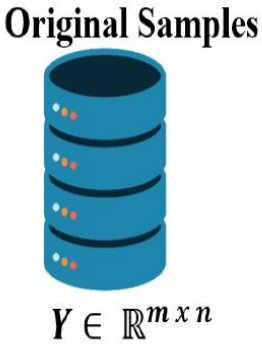
- Dados armazenados em uma matriz $\boldsymbol{Y} \in R^{m \times n}$, onde:

  m = número de amostras = *500*

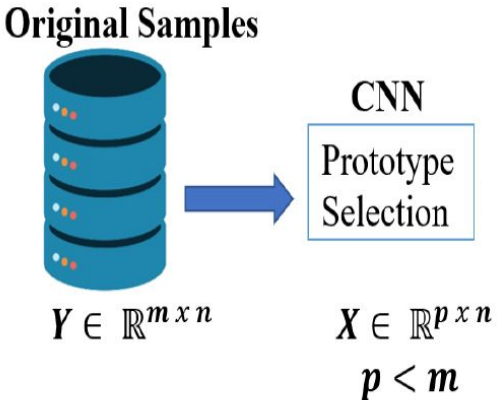
  n = número de sensores = *4*

- O framework de análise de dados proposto segue uma estratégia de reduzir progressivamente o número de amostras (*dimension reduction*) através de *prototype selection* e *data balance* e também o número de atributos para, finalmente, aplicar um algoritmo de classificação para determinar a postura do cadeirante;

3. Análise de Dados

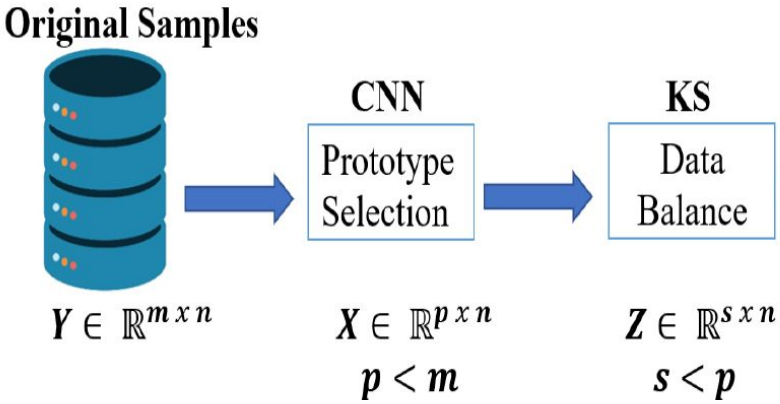


3. Análise de Dados

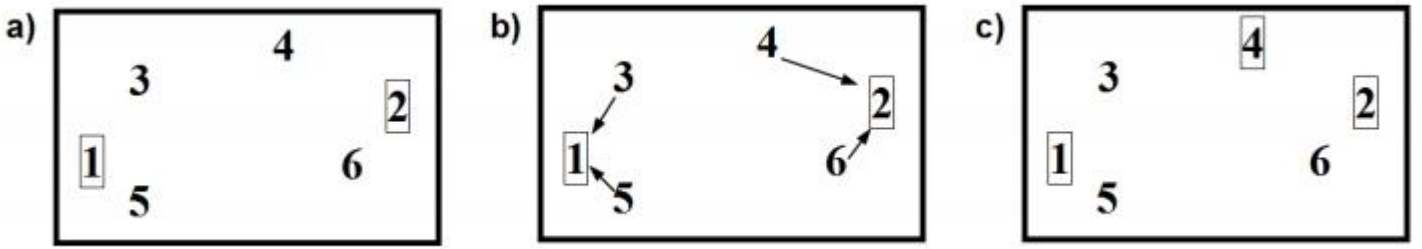


- Para reduzir o número de amostras e eliminar os ruídos visando diminuir os custos computacionais (hw limitado) utilizou-se o algoritmo ***CNN (Condensed Nearest Neighbour).*** Esse algoritmo seleciona um subset de pontos chamados de *prototypes* que apresentam o mesmo resultado (ou muito próximo) de se usar o data set original quando aplicado o classificador ***k*-NN.**
- Então, removendo os *outliers* e os *absorbed points*, ficamos com um set de *prototypes:* $X \in R^{p \times n}$, onde $p < n$

3. Análise de Dados

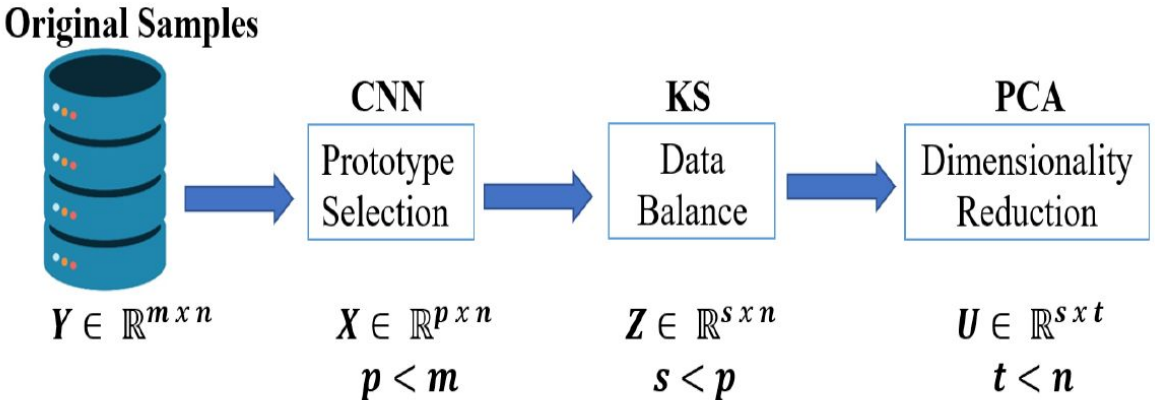


- K-S: seleção de amostras através do cálculo das distâncias euclidianas entre os vetores



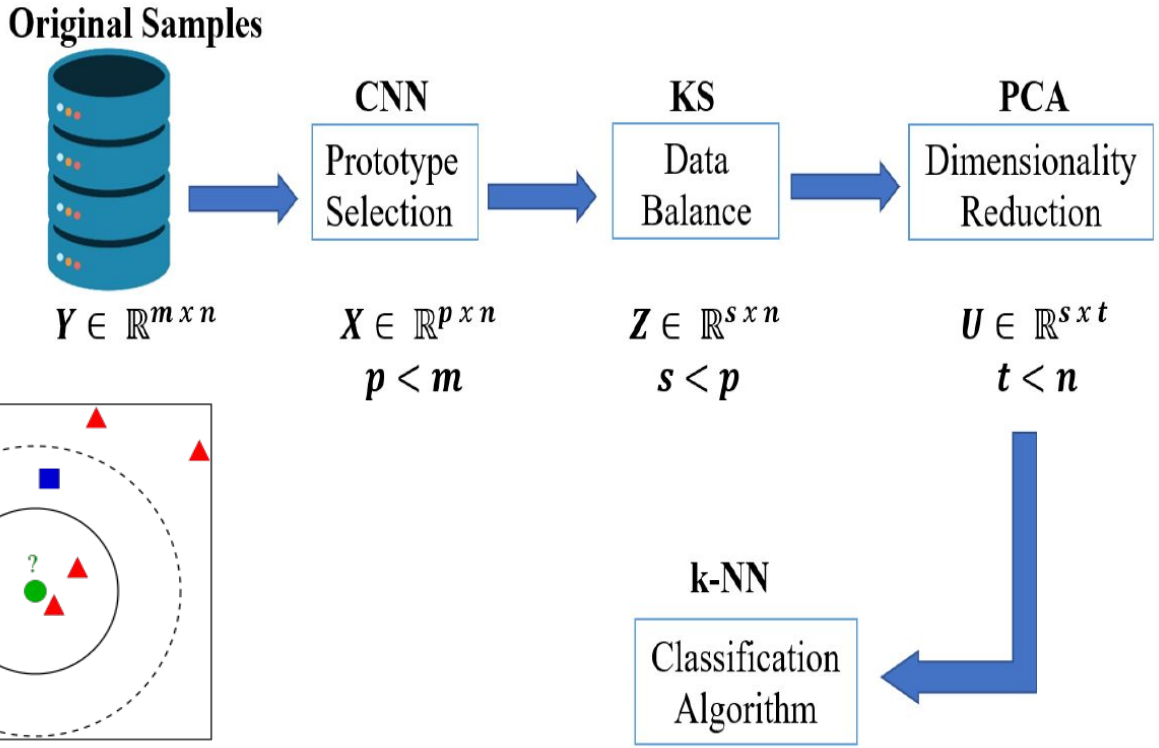
- Então: $Z \in R^{s\,x\,n}$, onde $s < p$

3. Análise de Dados

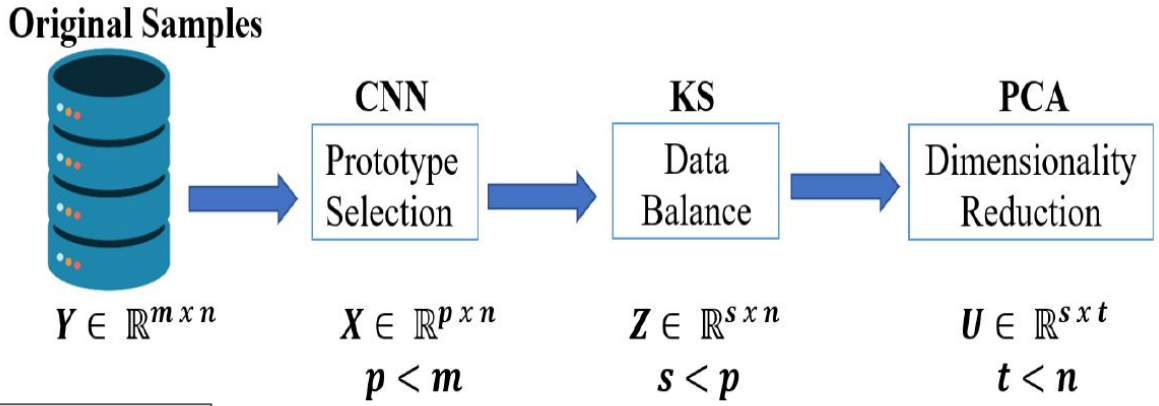


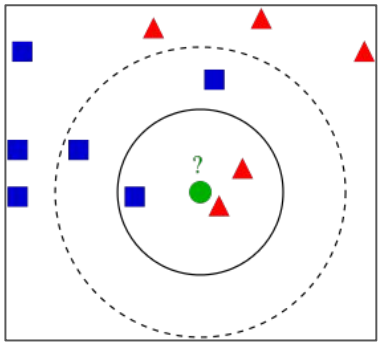
- Com o número de amostras já reduzido aplicou-se o algoritmo de aprendizado não-supervisionado ***PCA (Principal Component Analysis)*** para reduzir a dimensionalidade dos dados.
- Após o ***PCA*** gerou-se a matriz: $\boldsymbol{U} \in \mathbb{R}^{s\,x\,t}$, onde $t < n$

3. Análise de Dados



3. Análise de Dados

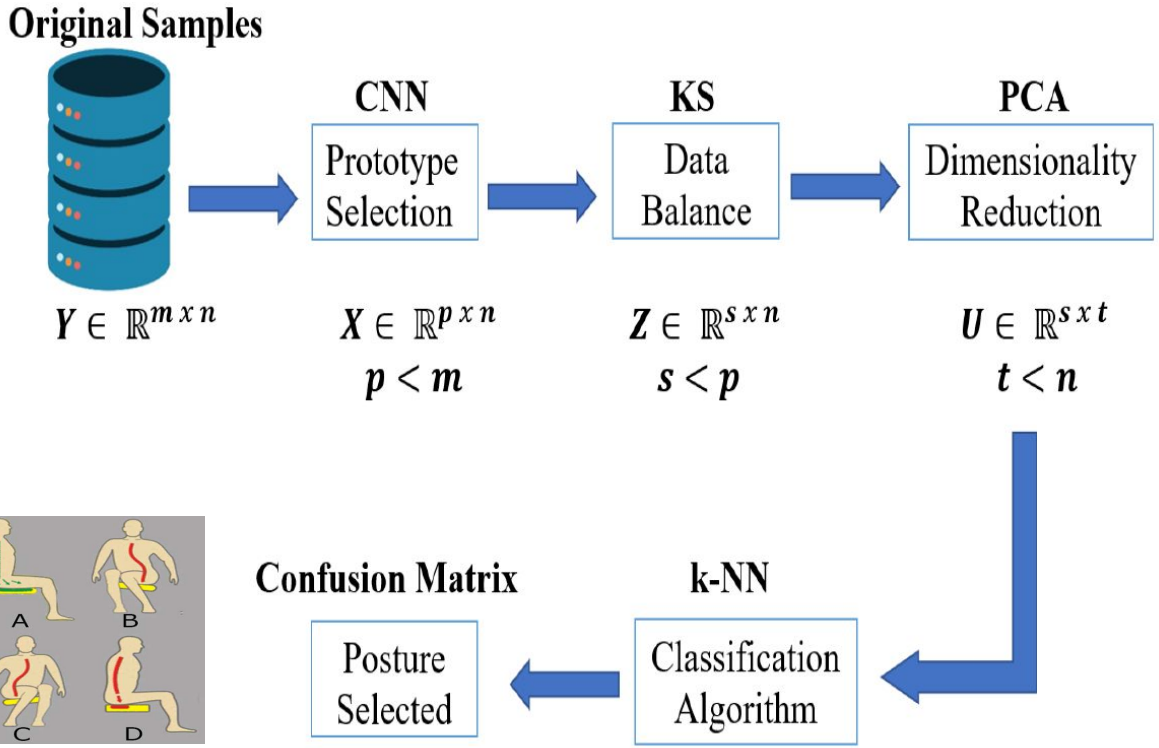




- simples de implementar;

- bom desempenho em tempo-real;

3. Análise de Dados



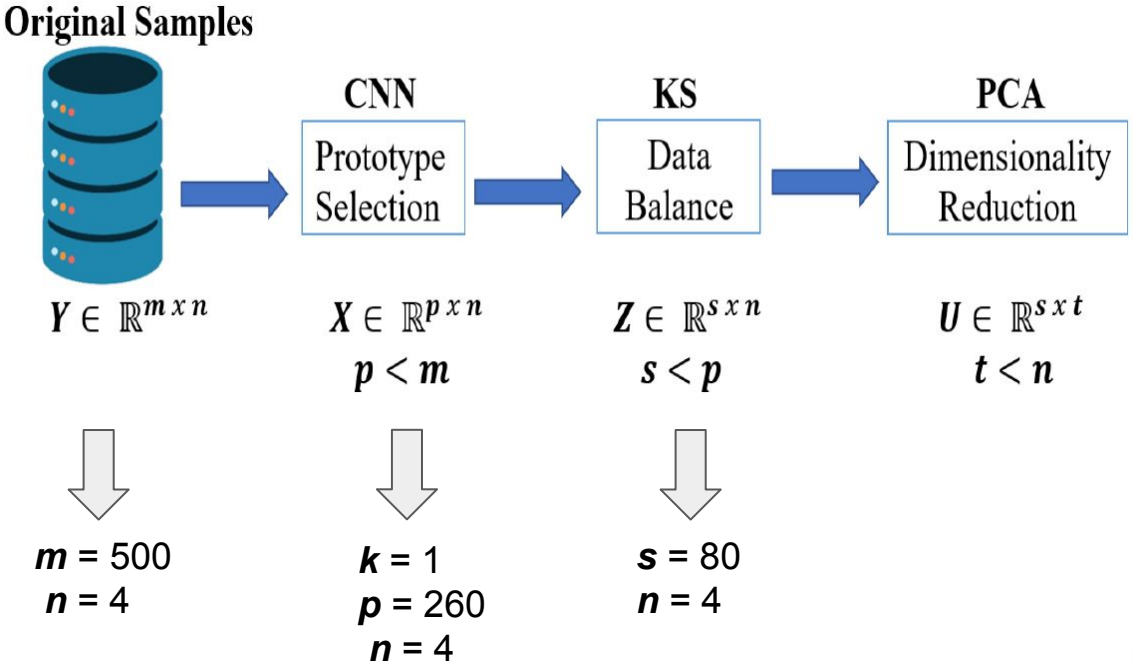
Resultados

- ***Redução dos dados de treinamento***

O sistema proposto reduz significativamente o número de dados da matriz de treinamento (88%), mantendo a mesma performance de classificação. Através dessa redução o microcontrolador consegue executar em tempos ótimos (menos de 1 minuto) e sem sobrecarregar os recursos de hardware;

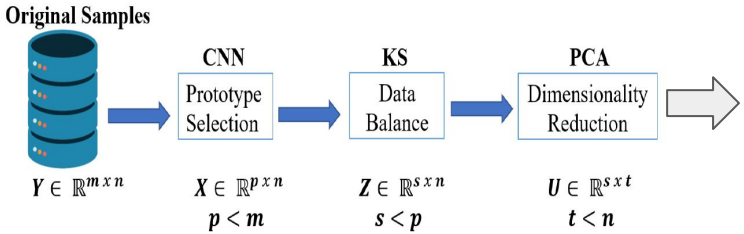
Resultados

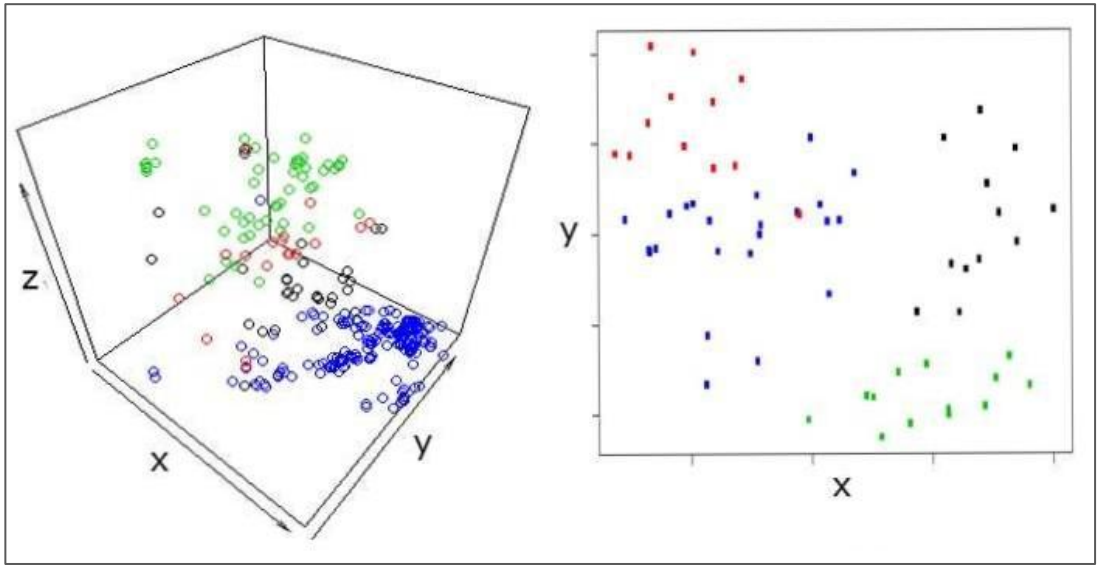
- ***Redução dos dados de treinamento***



Resultados

- ***Redução dos dados de treinamento***





Resultados

- ***Redução dos dados de treinamento***

Component indicators	PC1	PC2	PC3	PC4
Standard deviation	223.66	171.77	112.89	11.24
Proportion of variance	0.5414	0.3193	0.1379	0.0014
Cumulative proportion	0.5414	0.8607	0.9986	1.00

$$\mathbf{x} = \sum_{i=0}^{\mathbf{s}} (0.6125 * i) + (0.5228 * i) - (0.592 * i) + (0.019 * i)$$

$$\mathbf{y} = \sum_{i=0}^{\mathbf{s}} (0.7724 * i) - (0.554 * i) - (0.309 * i) + (0.008 * i)$$

Resultados

- ***Performance do algoritmo de classificação***

O algoritmo k-NN foi testado utilizando todas as matrizes de treinamento geradas durante o processo (***Y***, ***X*** , ***Z*** e ***U***). A precisão de acerto das classificações após uma validação cruzada de 10 iterações foi:

Training Set representation	Pos. A (mean ± std)	Pos. B (mean ± std)	Pos. C (mean ± std)	Pos. D (mean ± std)
matrix $\mathbf{Y}_{(500\times4)}$	85.45 ± 4.25	82.28 ± 4.8	78 ± 5.68	63.5 ± 8.2
matrix $\mathbf{X}_{(260\times4)}$	88.45 ± 3.28	80.21 ± 4.12	78 ± 4.32	65.5 ± 5.11
matrix $\mathbf{Z}_{(80\times4)}$	83.28 ± 2.45	80.21 ± 3.75	78 ± 3.77	66.5 ± 4.89
matrix $\mathbf{U}_{(80\times2)}$	81.58 ± 1.89	77.50 ± 2.14	78 ± 2.55	67.14 ± 3.04

Méd. Geral 76,06%

Resultados

- ***Performance do algoritmo de classificação***

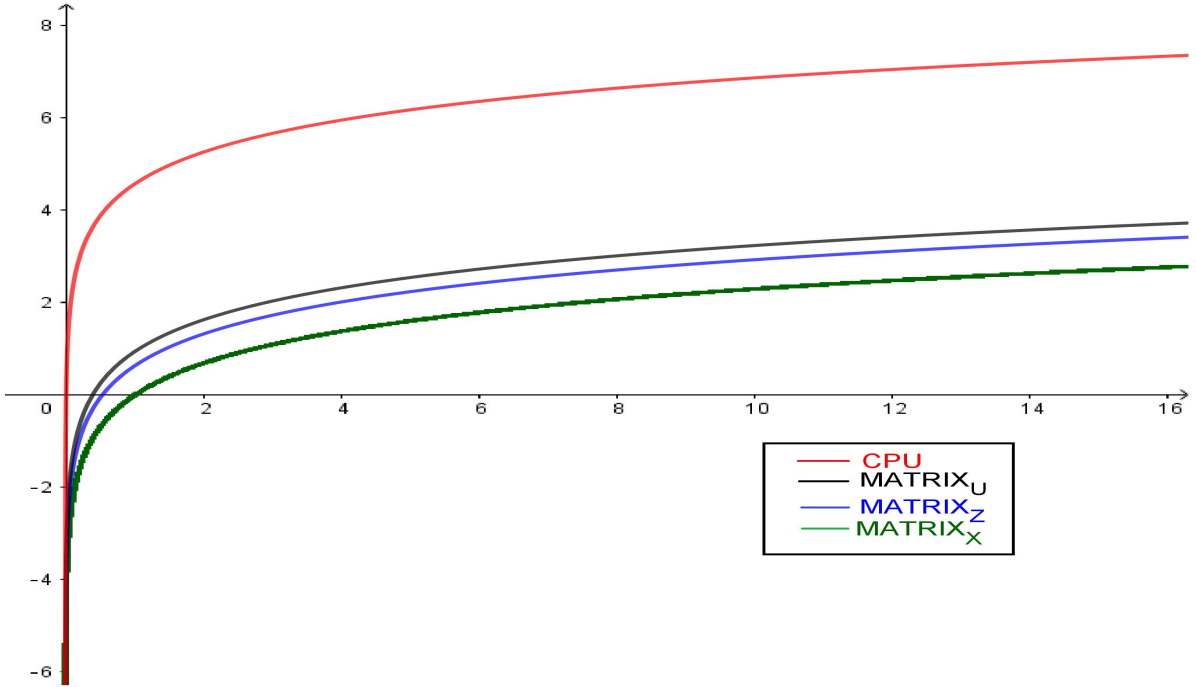
Classificação em tempo-real: **Méd. Geral 75,2%**

Training Set representation	**Used %** Memory	**Exec. time** Simulation	**Exec. Time** Real	**Current**
matrix $\mathbf{Y}_{(500\times5)}$	NN	NN	NN	NN
matrix $\mathbf{X}_{(260\times5)}$	95	1.71s	3.5s	359mA
matrix $\mathbf{Z}_{(80\times5)}$	65	1.25s	2.2s	267mA
matrix $\mathbf{U}_{(80\times2)}$	42	1.1s	1.5s	202mA

UNIVERSIDADE CATÓLICA DE PELOTAS

Resultados

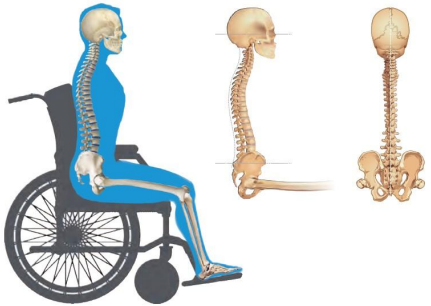
- ***Performance do algoritmo de classificação***



Conclusão

- Neste artigo foi proposto um sistema embarcado rodando um algoritmo de classificação k-NN em memória RAM que, juntamente com fases de redução do *dataset* de treinamento, é capaz de detectar posturas incorretas de um cadeirante com uma precisão de mais de 75%;

- Além disso, o sistema proposto reduz a quantidade de dados de treinamento em 88% (utilizando *CNN*, *KS* e *PCA*); Essa representação reduzida permite que nosso sistema embarcado armazene e classifique a postura do cadeirante em tempo-real;



UNIVERSIDADE CATÓLICA DE PELOTAS

Conclusão

- A performance do sistema foi satisfatória dadas as limitações de hardware: ele é capaz de detectar más posturas em aproximadamente 1 segundo, podendo gerar feedback imediato para o cadeirante corrigir-se e sentar adequadamente;

- Resultados experimentais demonstraram que os usuários de cadeiras de rodas podem melhorar sua qualidade de vida através de soluções embarcadas de baixo custo dotadas de *Machine Learning*;

Trabalhos Futuros

- Utilizar formatos de dados de arquivos mais adequados para sistemas inteligentes de tempo real (ex. *csv*);
- Incorporar um sensor extra ao sistema para tentar melhorar ainda mais a performance do sistema, especialmente para classificação da posição D;

Referências

- Rosero, Paul & Peluffo, Diego & Batista, Vivian & Serrano, Jorge & Rosero, Edwin. (2018). Intelligent System for Identification of Wheelchair User's Posture using Machine Learning Techniques. IEEE Sensors Journal. PP. 1-1. 10.1109/JSEN.2018.2885323.

- S. Nunez-Godoy, V. Alvear-Puertas, S. Realpe-Godoy, E. Pujota-Cuascota, H. Farinango-Endara, I. Navarrete-Insuasti, F. Vaca-Chapi, P. Rosero-Montalvo, and D. H. Peluffo, "Human-sitting-pose detection using data classification and dimensionality reduction," in 2016 IEEE Ecuador Technical Chapters Meeting (ETCM), Oct 2016, pp. 1–5.

- Y. R. Huang and X. F. Ouyang, "Sitting posture detection and recognition using force sensor," in 2012 5th International Conference on BioMedical Engineering and Informatics, Oct 2012, pp. 1117–1121.

- P. Rosero-Montalvo, D. H. Peluffo-Ordez, A. Umaquinga, A. Anaya, J. Serrano, E. Rosero, C. Vsquez, and L. Suarz, "Prototype reduction algorithms comparison in nearest neighbor classification for sensor data: Empirical study," in 2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM), Oct 2017, pp. 1–5.

UNIVERSIDADE CATÓLICA DE PELOTAS