

Projeto Final

Reconhecedores e Regressores

PSI3571 - Práticas em Reconhecimento de
Padrões, Modelagem e Inteligência
Computacional - Grupo IV

Rodrigo Vidal Zobarán

André Ferrari Duarte do Páteo

Ibraim Rebouças

Escola Politécnica da Universidade de São Paulo

Resumo da Apresentação

1. Reconhecedor - Áudios Editados
2. Regressor - Corrente em Transistor
3. Reconhecedor - Defeito em Máquina

1. Reconhecedor de Áudios Editados

- a. Apresentação do Problema
- b. Limitações e Requisitos
- c. Coleta de dados
- d. Pré-processamento
- e. Estrutura da Rede
- f. Medida de Desempenho e Resultados
- g. possíveis melhoras e continuações

Apresentação do Problema

Programas de edição cada vez mais poderosos...

Importância de validar uma gravação:

Uso como provas

Jornalismo

Evitar desinformação - Fake news

Apresentação do Problema

Solução Atual:

Análise do espectro do ruído

Pró: Taxa de acerto alta

Con: Requer banco de dados enorme

Proposta: Pré-análise com rede neural

Limitações e Requisitos

Requisito: Evitar falsos negativos

Limitação: Restringir-se a um microfone

Requisito: Não demorar demais para treinar

Limitação: Restringir-se a um tipo de gravação

Coleta de dados

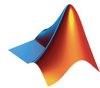
Áudios Verdadeiros:

- Várias gravações em um microfone só
- Uma pessoa falando, áudios de 10 segundos

Áudios Editados:

- Script do MatLab 
- Copia e cola pedaços de áudios verdadeiros

Pré-Processamento



1. Leitura dos áudios

$48000 \cdot 3 = 144000$ amostras/áudio

2. Picotar o áudio em bins

800 amostras por bin - 180 bins por áudio

3. Filtragem dos áudios

a. Filtro de Butterworth, 6 coefs, passa faixa

4. FFT em cada bin

5. 4 maiores módulos de freq. em cada bin

8 valores por bin - 1440 entradas por áudio



18	5	...	15
203	234	...	434
...

Estrutura da Rede

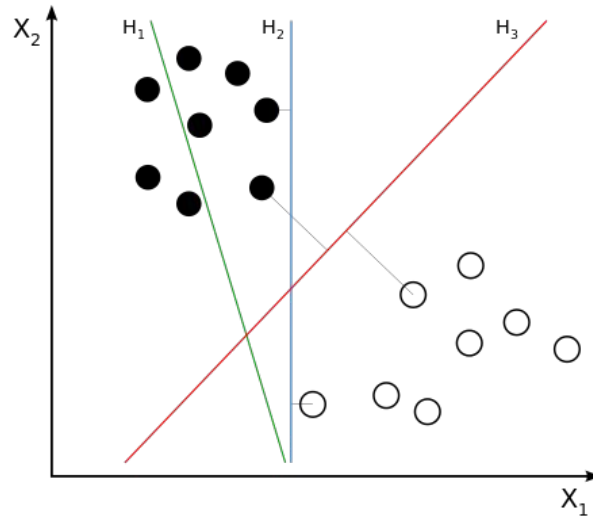
Implementação: Scikit Learn (Python)



Estrutura escolhida: **Linear SCV**

O github da microsoft tem algumas aplicações de SKLearn bem bacanas,
usei elas para a base do script

Estrutura da Rede - Linear SCV

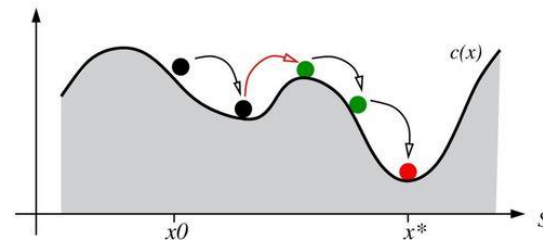


Estrutura da Rede

Outras otimizações:

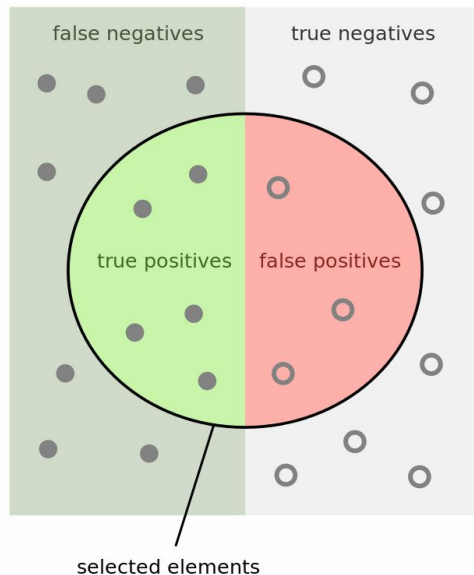
1. Têmpera Simulada - Evitar mínimos locais
2. Erro L2 - Punir pesos altos
3. SVC linear - Resultado desbalanceado

Favorece positivos!




Medida de Desempenho e Resultados

Importante distinguir falsos positivos, falsos negativos, verdadeiros positivos, verdadeiros negativos




Medida de Desempenho e Resultados

How many selected items are relevant?

$$\text{Precision} = \frac{\text{Relevant}}{\text{Selected}}$$


How many relevant items are selected?

$$\text{Recall} = \frac{\text{Relevant}}{\text{Total Relevant}}$$


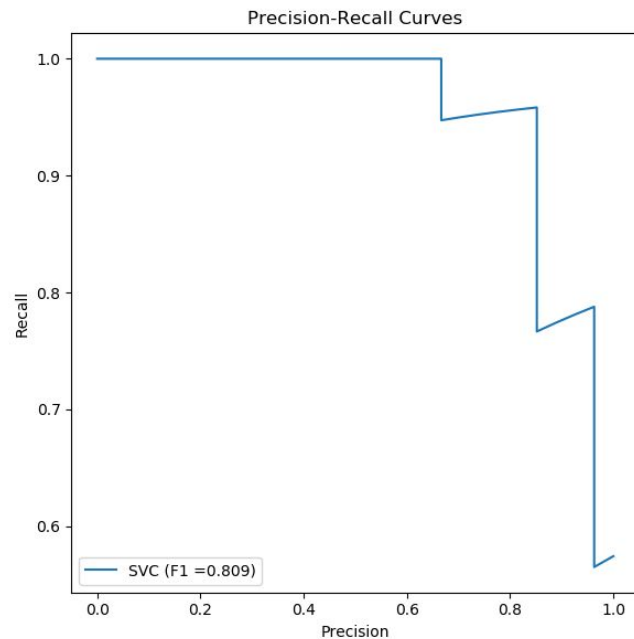
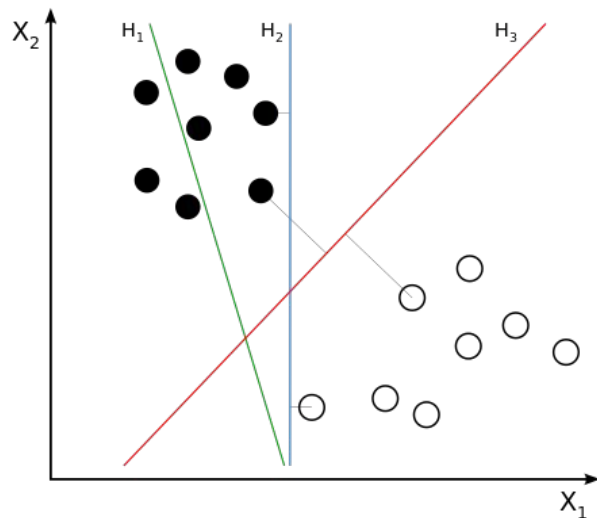
Medida de Desempenho e Resultados

Pontuação: Média Harmônica de *precision* e *recall*

$$F_1 = \frac{2}{\frac{1}{\text{recall}} + \frac{1}{\text{precision}}}$$

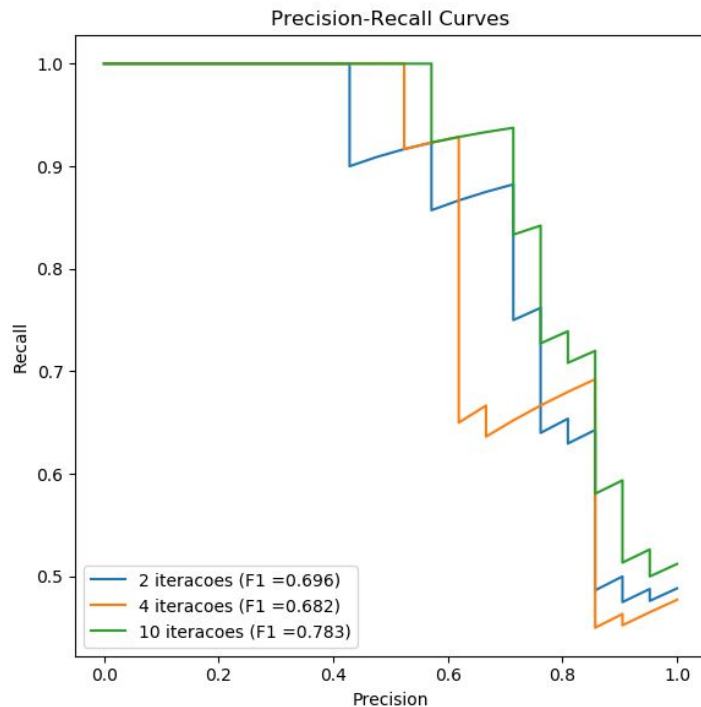
Medida de Desempenho e Resultados

Curva de P-R: relaxando requisitos para ser verdadeiro



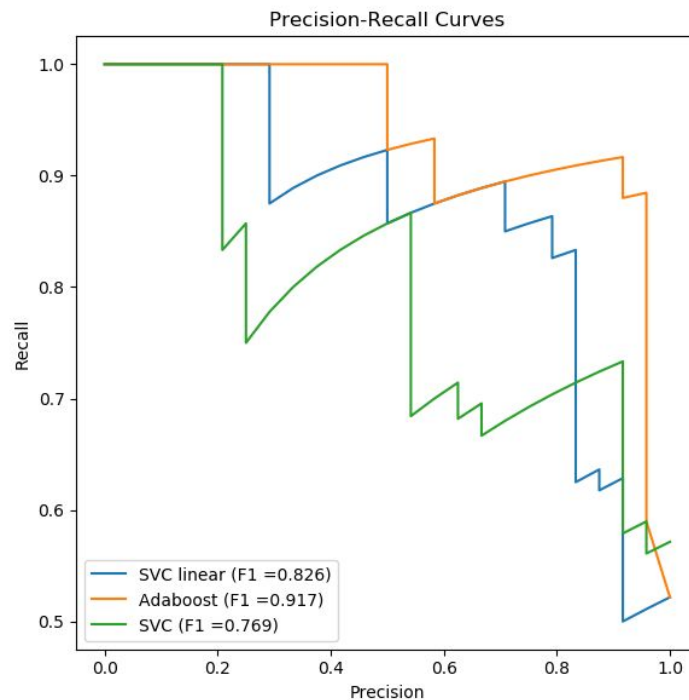
Medida de Desempenho e Resultados

- Usando quantidades diferentes de iterações para ver quão afetado é o resultado



Medida de Desempenho e Resultados

- Usando estruturas de redes neurais diferentes para testar como é afetado o resultado



F1 = 0.81

Bem razoável!

Possíveis Melhoras e Continuações

- Treinar a rede com mais de um microfone
- Usar outras técnicas de edição
- Adicionar mais ruído aos áudios

2 - Estimador de Corrente no Fototransistor

- a. Apresentação do Problema
- b. Desenvolvimento da Rede Neural
- c. Refinamento da Rede Neural
- d. Conclusões

Apresentação do Problema

Ferramentas atuais são caras por conterem muitas ferramentas além do desejado.

Interessante por ser um método mais rápido e simples do que realizar cálculos manualmente

Apresentação do problema

O problema envolve os seguintes dados de entrada:

- Tensão (V)
- Temperatura (°C)
- Intensidade Luminosidade (mW/cm^2)

Só há uma saída:

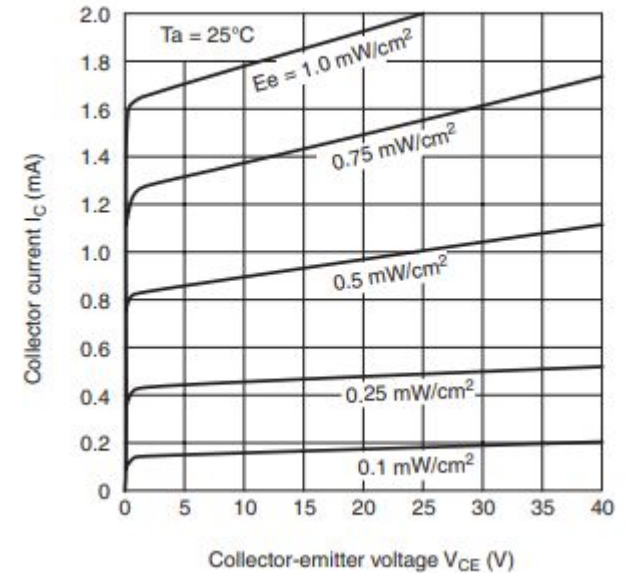
- Corrente (mA)

Coleta de dados

Equacionamento de curvas fornecidas em datasheet do transistor.

Geração de arquivos csv a partir do python

Fig. 7 Collector Current vs. Collector-Emitter Voltage



Desenvolvimento da rede

Versão inicial

- 309 pares empíricos: 259 de treino e 50 de teste
- Único nó para as três entradas

Erros obtidos:

Erro de Treino	Erro de Teste
0.01717	0.01956

Refinamento da rede

- Redução de complexidade
- Re-sorteio de pesos
- Aumento de pares empíricos
- Dados de validação
- Cross-Validation
- Aumento do número de nós

Redução de Complexidade

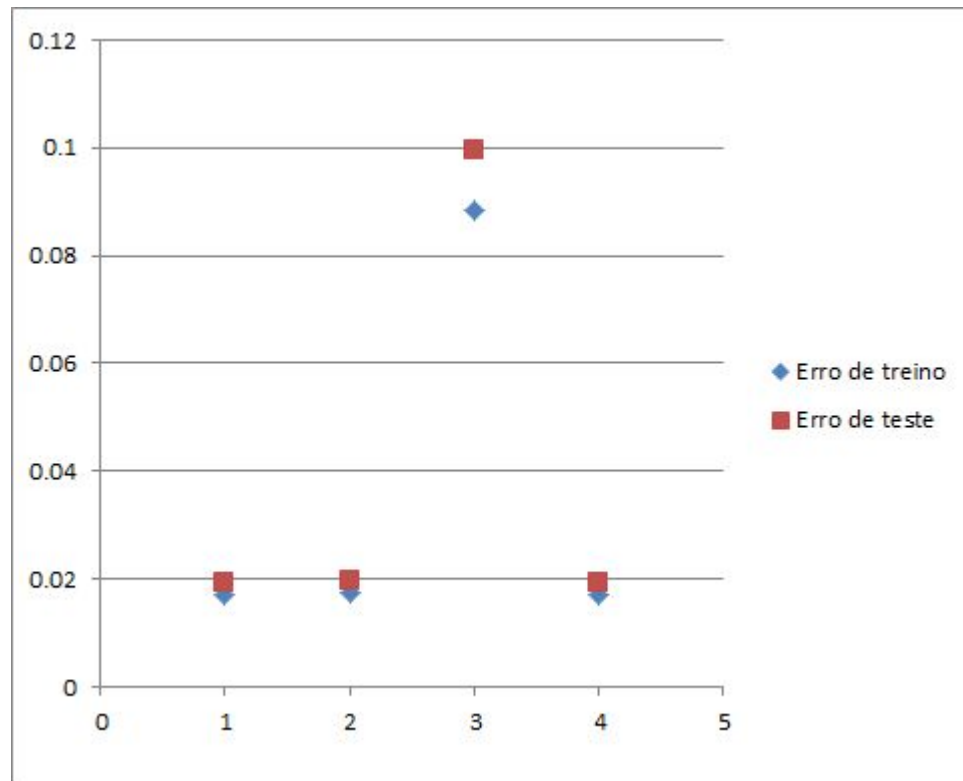
Redução e combinação do número de entradas: método de verificar se todas influenciam e não atrapalham na modelagem

Entradas combinadas duas a duas:

Tensão e Luminosidade
Tensão e Temperatura
Luminosidade e Temperatura

Redução de Complexidade

Entradas	Erro de Treino	Erro de Teste
Tensão, Luminosidade , Temperatura	0.01717	0.01956
Tensão e Luminosidade	0.017292	0.019756
Tensão e Temperatura	0.088405	0.099534
Luminosidade e Temperatura	0.017155	0.019369



Conclusão: Luminosidade tem maior impacto.

Re-sorteio de pesos

Tensão	Luminosidade	Temperatura	Viés	Erro de Treino	Erro de Teste	Tensão Final	Luminosidade Final	Temperatura Final	Viés Final
0.286294	0.457187	0.902463	0.28294	0.0132805	0.0133202	0.134232	3.12429	-12.1668	-12.0301
-0.172826	-0.311869	-0.674795	-0.00930815	0.0132897	0.0133333	0.145328	3.10735	-11.9115	-11.7623
-0.97571	-0.33799	-0.25169	0.860652	0.01332061	0.01337	0.151567	3.10121	-11.9672	-11.8831
-0.96771	-0.48039	0.382366	0.802545	0.01328633	0.013328	0.142	3.11707	-12.1739	-12.0218
-1.69088	3.63826	2.33299	0.317545	0.01330975	0.013357	0.149654	3.08661	-11.8942	-11.7491
0.743278	-4.59563	-3.72524	2.7044	0.01331038	0.01331	0.162071	3.10845	-13.082	-12.9613

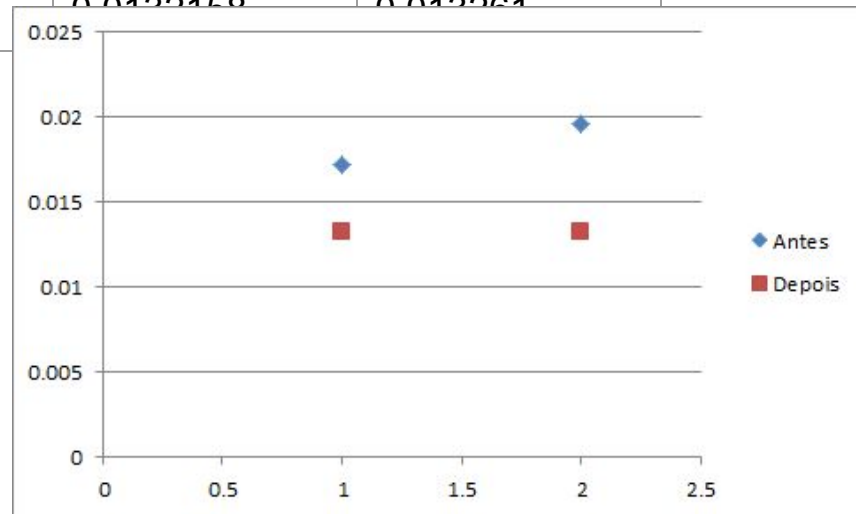
Convergência para um mesmo mínimo em todos os ensaios

Aumento de Pares Empíricos

Aumento de informação para aumento de eficiência

	Pares de Treino	Pares de Teste	Erro de Treino	Erro de Teste
Antes	259	259	0.01717	0.01956
Depois	1984	371	0.0122159	0.012264

Pequena melhora nos resultados



Dados de Validação e Cross-Validation

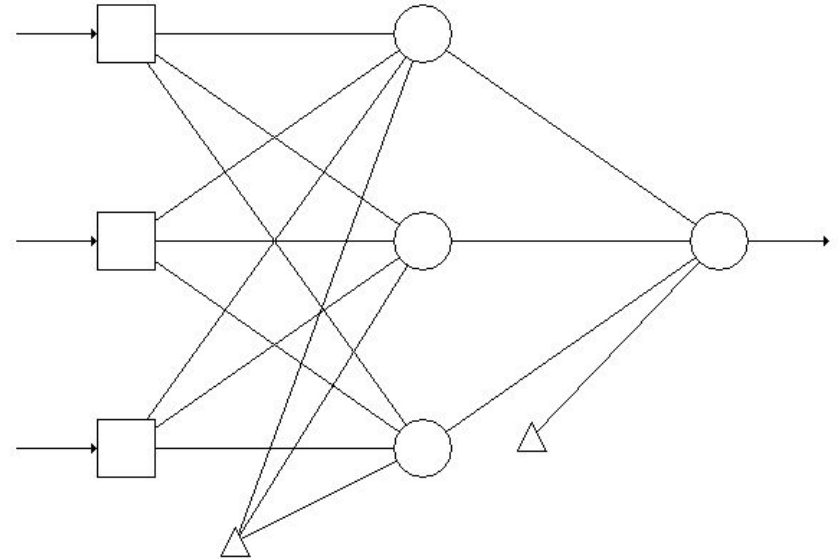
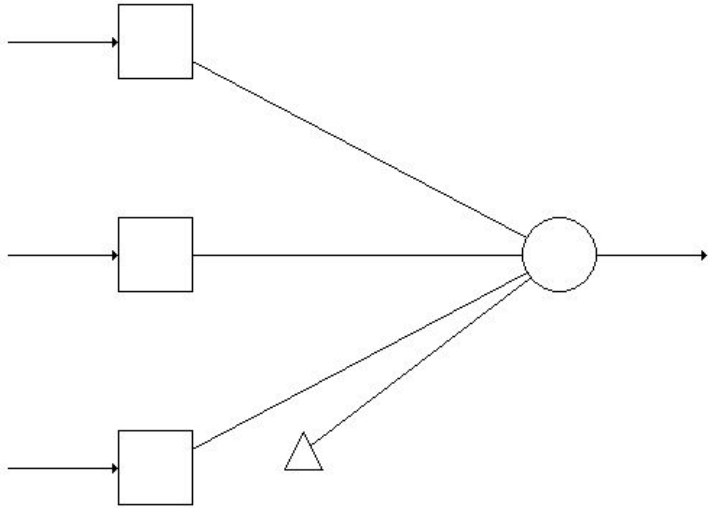
	Pares Empíricos	Erro de treino	Erro de teste
Validação	246	-	0.013434
Cross-Validation 1	1301	0.0133465	0.013273
Cross-Validation 2	1300	0.013264	0.0133402

Comparando com erros de treino e teste anteriores, os resultados são satisfatórios

Erro de Treino	0.0132158
Erro de Teste	0.013261

Aumento do número de nós da Rede

Mudança de uma rede de 1 nó para 3.



Aumento do número de nós da Rede

Melhora imediata:

	Erro de Treino	Erro de Teste
1 Nó e 309 dados	0.01717	0.01956
1 Nó e 2549 dados	0.0132158	0.013261
3 Nós e 2549 dados	0.0009888716	0.0009922179

Novos testes com base nos ensaios anteriores:

- Redução de complexidade para evitar sobre-aprendizado
- Re-sorteio de pesos para evitar mínimo local

Aumento do número de nós da Rede - Re-sorteio de pesos iniciais

Teste	Treino	Vies final	V Final	L Final	T final
0.00103659	0.00106891	-2.83899	-0.49454	2.87896	0.923761
		-6.75572	0.036397	-20.4757	15.385
		2.38397	0.157566	2.86824	-0.17515
		-4.75816	8.62442	-16.8903	4.90165
0.00290632	0.00304224	2.62102	-1.39482	-2.07438	-5.71399
		7.28714	0.112884	12.6981	-6.48711
		4.29341	-0.03405	-0.62817	3.8
		-0.32355	0.475702	10.1347	-16.4418

Mudança drástica nos resultados => mínimos locais.

Necessidade de se certificar que a rede não está em um mínimo local.

Três testes realizados. Resultados são melhores com pesos iniciais variando entre -1 e 1.

Aumento do número de nós da Rede - Redução do número de entradas

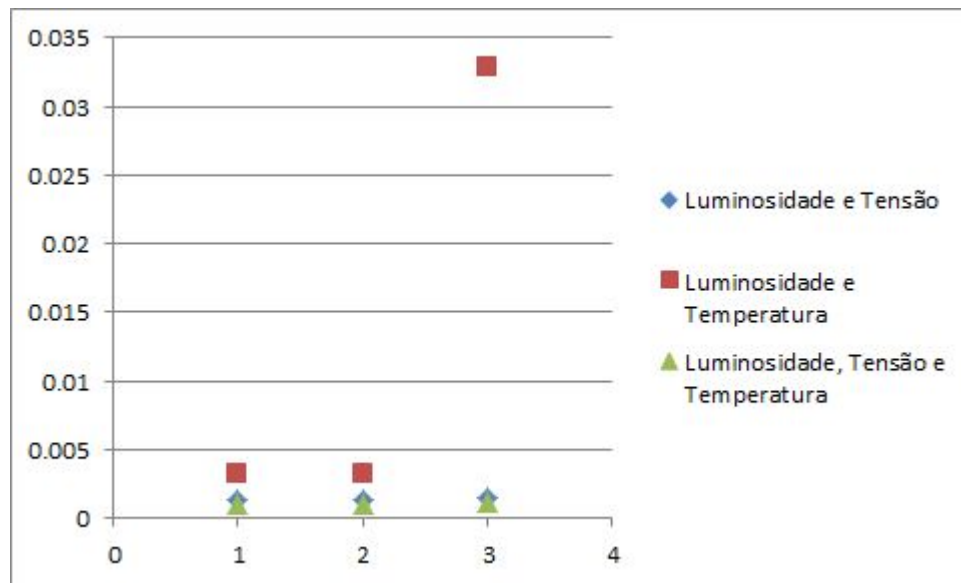
Com base nos ensaios feitos anteriormente, é pertinente testar novamente a influência da tensão e temperatura

	Erro de Treino	Erro de Teste	Erro de Validação
Luminosidade e Tensão	0.0012730978	0.0012934915	0.0014992802
Luminosidade e Temperatura	0.0032164363	0.0032150314	0.0032883681
Luminosidade, Tensão e Temperatura	0.0009888716	0.0009922179	0.0012103456

Aumento do número de nós da Rede - Redução do número de entradas

Luminosidade e temperatura apresentam pior desempenho.

Luminosidade, temperatura e tensão apresentam melhor desempenho em todos os ensaios



Conclusões

- Modelo eficiente e veloz
- Necessidade de prestar atenção nos pesos
- Temperatura é a entrada com menor influência e pode ser deixada de lado em caso de pouco tempo para fazer medições

3 - Identificador de defeitos em máquinas

- a. Apresentação do Problema
- b. Solução Proposta
- c. Parâmetros Envolvidas
- d. Pré Processamento
- e. Resultados
- f. Conclusão

Apresentação do Problema

Máquinas de grande porte possuem manutenção difícil e custosa, em geral para todo o processo do qual fazem parte.

A manutenção em geral é feita por pessoal autorizado pelo fabricante.

Atualmente a maioria das falhas é detectada quando seus efeitos já não podem ser facilmente revertidos.

→ Perda de dinheiro

Solução proposta

- Identificação do defeito por meio de análise espectral pré processada
- Exige a instalação de um ou mais microfones na máquina
- Processamento periódico (tempo necessário para a coleta de uma amostra significativa de áudio)
- Implementações possíveis
 - Computador comum
 - Sistema embarcado dedicado

Pontos positivos e negativos

→ Pontos positivos

- Redução e custos com manutenção preventiva
- Identificação mais rápida do problema
- Menos pessoas envolvidas no processo

→ Pontos negativos

- Necessita de dados de calibração para cada máquina (pode envolver a falha forçada de maquinário em ambiente de laboratório)

Parâmetros envolvidos

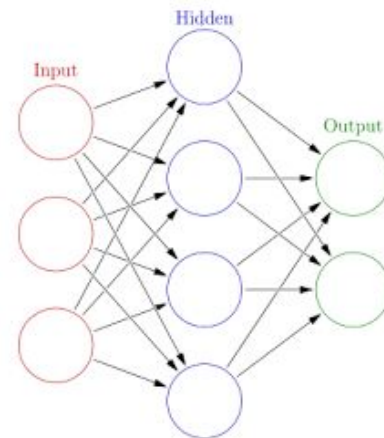
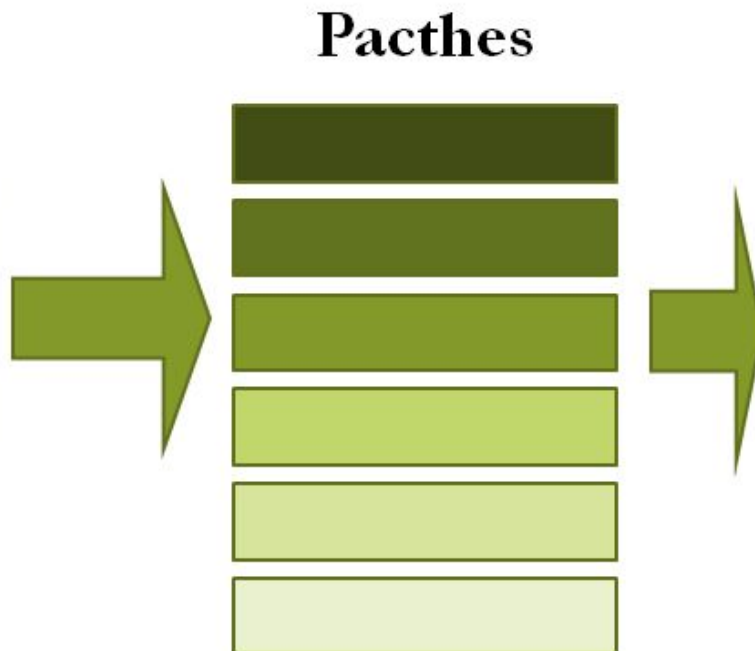
→ Parâmetro de entrada

- Amostra sonora bruta
 - Uma ou mais, dependendo do maquinário

→ Parâmetros de saída

- Classificação do defeito detectado
 - (NORMAL - DEFEITO1 - DEFEITO2)

Pré processamento



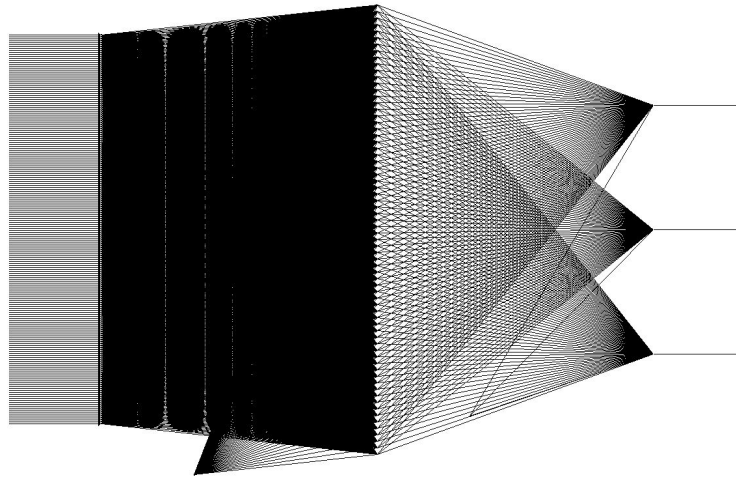
Pré processamento

- Feito em Matlab
- Para os testes finais:
 - Condições diversas simuladas automaticamente (adição de ruído, intermitência do defeito e permutações de amplitude)

Rede neural

→ Topologia otimizada

- 200-70-3 (obtenção empírica)

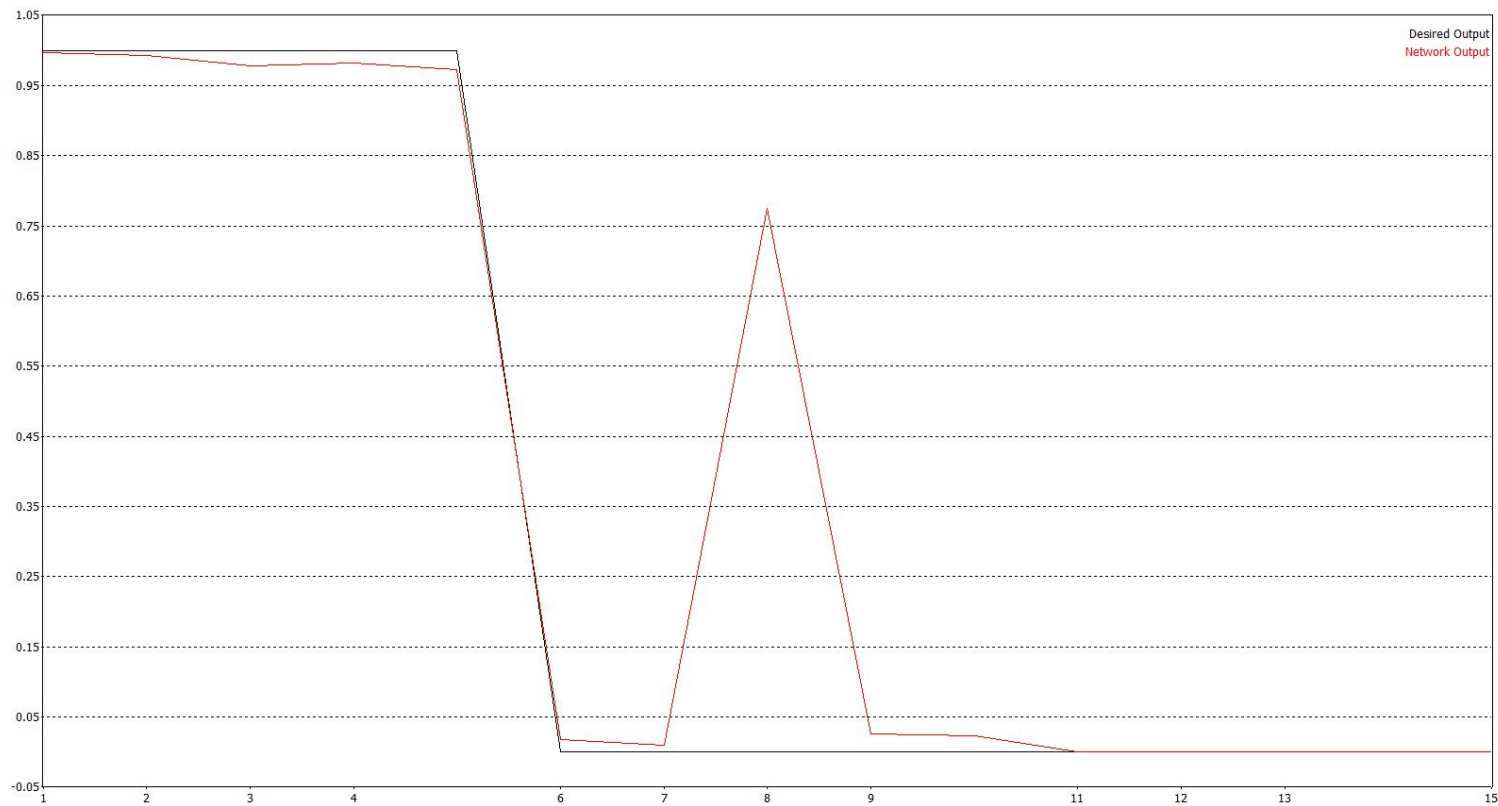


the input and the output layers

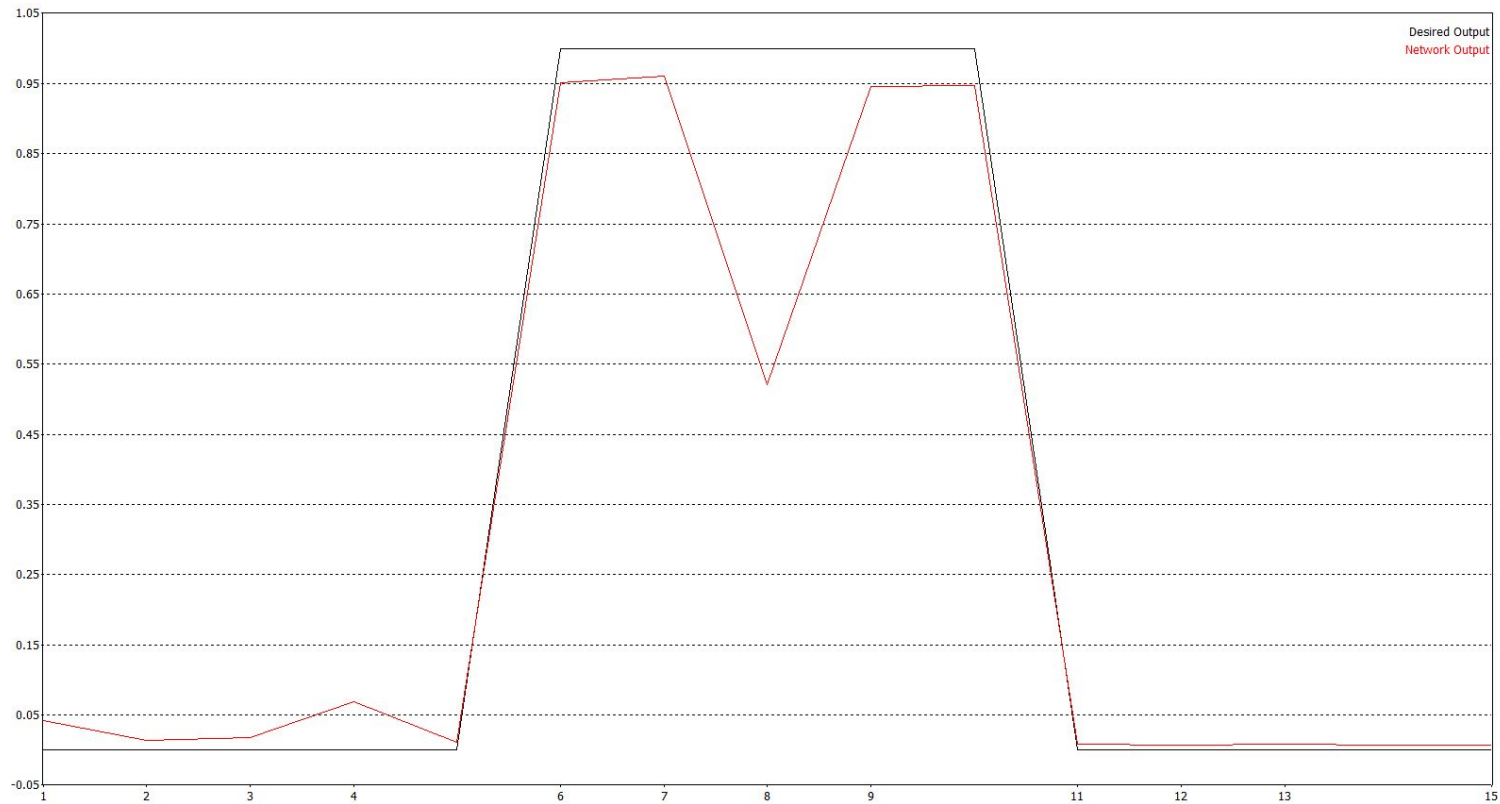
Resultados

- Quantidade de dados de treino
 - 105 vetores de entrada: 35 para cada tipo de amostra
 - (NORMAL - DEFEITO1 - DEFEITO2)
- Quantidade de dados de teste
 - 15 vetores de teste
- 200 patches para cada vetor

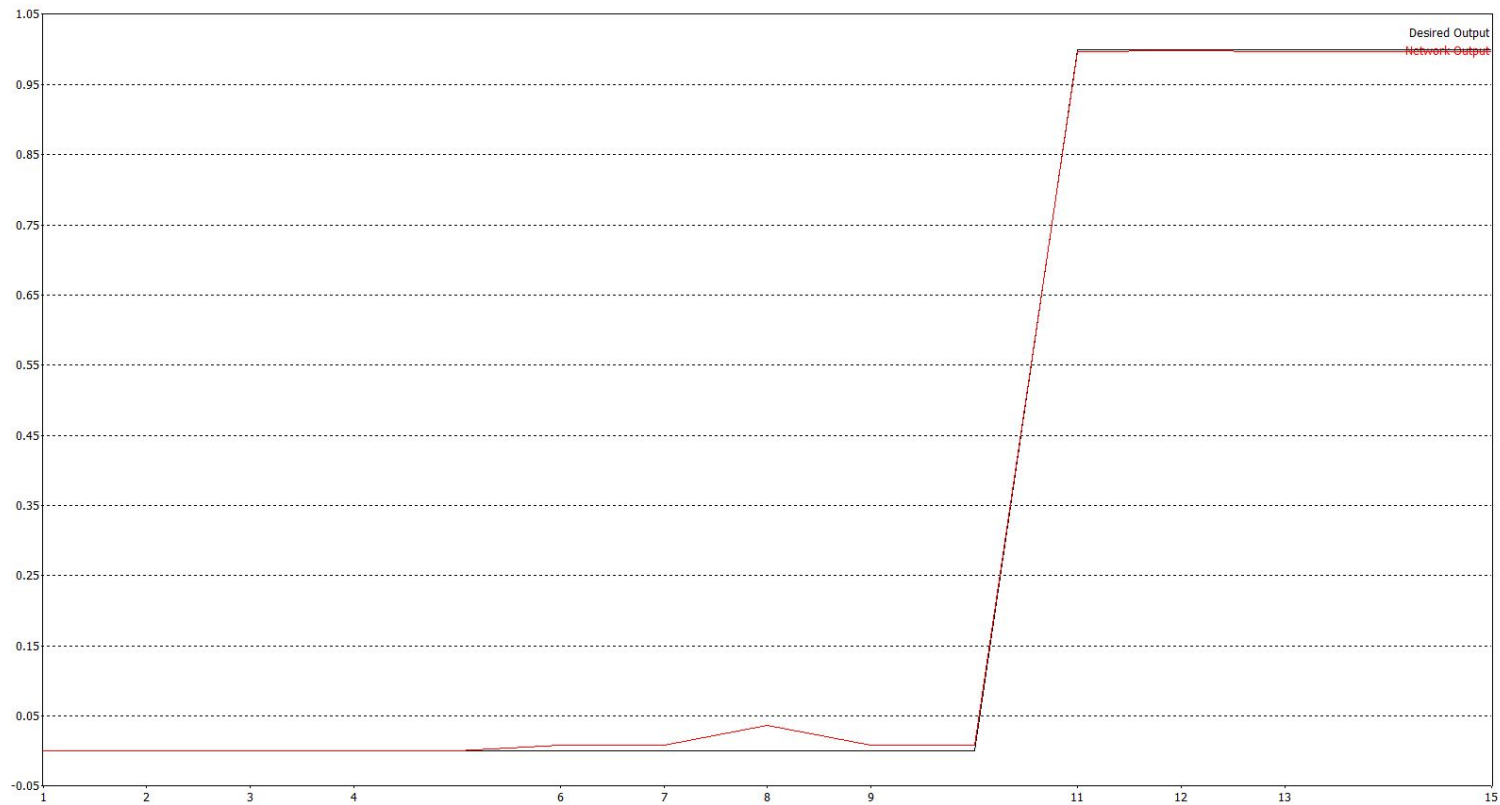
Resultados - Label “NORMAL”



Resultados - Label “DEFEITO1”



Resultados - Label “DEFEITO2”



Resultados - Análise de erros

→ Valor RMS

- $\text{RMS} = 0.068766$

→ **Outra medida**

→ Taxa de acertos

Valor esperado $\pm 15\%$
=
 1 ± 0.15

Taxa de acertos = 95,55%

Conclusão

- A análise original era complexa e teve que ser abandonada
- A alternativa mais simples (na mesma linha de raciocínio) mostrou-se satisfatória para os resultados intermediários e finais
- Pode ser que haja interferências não previstas em uma aplicação real (ruído das demais máquinas, pessoas falando, etc.) que podem exigir uma análise mais complexa

Bibliografia

Datasheet do transistor: http://www.sharp.co.jp/products/device/doc/opto/pt480_e.pdf