

Universidade de São Paulo Campus  
São Carlos  
ICMC  
Introdução às redes neurais

**Regressor de latitude, longitude em  
relação á dados de músicas**

Aluno: Jeffri Erwin Murrugarra Llerena  
Aluno: Jahir Gilbert Medina Garcia  
Professora: Roseli Aparecida Romero

Septiembre  
2018

## Conteúdo

<b>1</b>	<b>Resumo</b>	<b>1</b>
<b>2</b>	<b>Apresentação</b>	<b>1</b>
<b>3</b>	<b>Descrição de atividades</b>	<b>1</b>
<b>4</b>	<b>Análise dos Resultados</b>	<b>3</b>

## 1 Resumo

O objetivo deste projeto é fazer um regressor a partir de dados de musicas obter a latitude , longitude

## 2 Apresentação

O conjunto de dados foi construído a partir de uma coleção pessoal de 1059 faixas cobrindo 33 países / área. A música usada é tradicional, étnica ou "mundial" apenas, classificada pelos editores do produto em que aparece. Qualquer música ocidental não é incluída porque sua influência é global - o que procuramos são os aspectos da música que mais influenciam a localização. Assim, ser capaz de especificar um local com forte influência na música é central.

A localização geográfica da origem foi coletada manualmente das informações das notas da capa do CD e, quando essa informação era inadequada, pesquisamos outras fontes de informação. Os dados de localização são limitados em precisão ao país de origem.

O país de origem foi determinado pelo principal país / área de residência do artista ou dos artistas. Qualquer faixa que tenha origem ambígua não está incluída. Nós tomamos a posição da cidade capital de cada país (ou a província da área) por latitude e longitude como o ponto de origem absoluto.

## 3 Descrição de atividades

O objetivo é fazer um regressor de duas variables , para isso foi feito dois arquitecturas de uma red multicamada.

Primera Arquitectura:

- Camada de entrada: 68 neurônios
- Camada Oculta: 2 neurônios
- Camada de saída: 2 neurônios

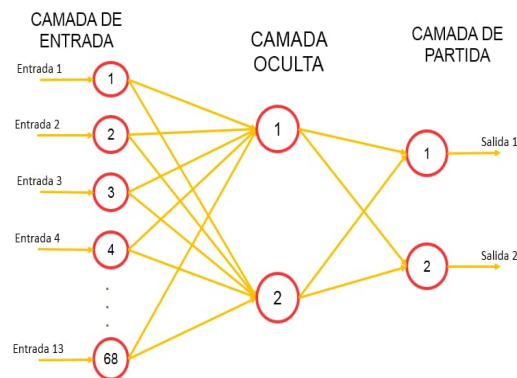


Figura 1: Arquitetura 68-2-2

Segunda Arquitetura:

- Camada de entrada: 68 neurônios
- Primeira Camada Oculta: 3 neurônios
- Segunda Camada Oculta: 2 neurônios
- Camada de saída: 2 neurônios

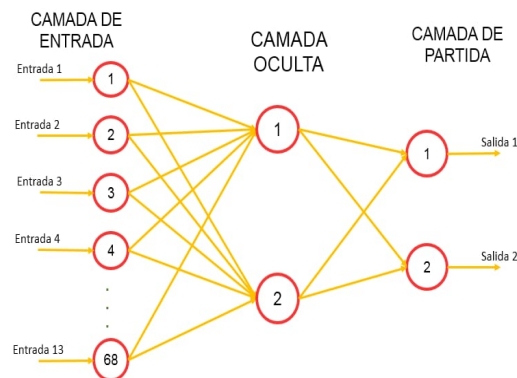


Figura 2: Arquitetura 68-3-2-2

Nas duas arquiteturas, o algoritmo de backpropagation foi usado para atualizar os pesos.

Proceso de treinamento e test:

- A rede foi construída com uma das arquiteturas explicadas

- Se particiono a data dos tracks, en um conjunto de treinamento, test
- A rede foi treinada con o algoritmo de backpropagation temdo em conta
  - Criterio de parada( Número de ciclos)
  - Parâmetro momentum
  - Parâmetro velocidade de aprendizaje
- Uma vez terminado o treinamento foi avaliado o erro cuadraático no conjunto de test e treinamento

## 4 Análise dos Resultados

Os resultados apresentados foram obtidos variando os parâmetros momentum( 0,0.25,0.55,0.8 ), variável de aprendizado( 0.1,0.4,0.7,0.9 ), tipo de arquitetura, tamanho da partição ( 0.65,0.7,0.75) e número de ciclos utilizados no treinamento( 40,70,100 ).

A metrica de avaliação para este caso é o erro cuadrático, que foi obtida para os conjuntos de treinamento e teste.

### Resultados na arquitectura 68-2-2

Número de iterações		
40	Acuracia(treinamento)	0.06681299
	Acuracia(test)	0.07401668
70	Acuracia(treinamento)	0.06457553
	Acuracia(test)	0.07479783
100	Acuracia(treinamento)	0.06160778
	Acuracia(test)	0.07149748

Tabela 1: Promedio de resultados de número de iterações

Tamanho de partição		
0.65	Acuracia(treinamento)	0.06429104
	Acuracia(test)	0.0727634
0.70	Acuracia(treinamento)	0.06429485
	Acuracia(test)	0.07396134
0.75	Acuracia(treinamento)	0.06441041
	Acuracia(test)	0.07358724

Tabela 2: Promedio de resultados de tamanho de partição

Parâmetro momentum		
0	Acuracia(treinamento)	0.06562328
	Acuracia(test)	0.07403919
0.25	Acuracia(treinamento)	0.06315983
	Acuracia(test)	0.07338172
0.55	Acuracia(treinamento)	0.0652189
	Acuracia(test)	0.07368107
0.8	Acuracia(treinamento)	0.06332638
	Acuracia(test)	0.07264733

Tabela 3: Promedio de resultados de parâmetro momentum

Parâmetro de aprendizado		
0.1	Acuracia(treinamento)	0.06290987
	Acuracia(test)	0.0694149
0.4	Acuracia(treinamento)	0.06420743
	Acuracia(test)	0.07422719
0.7	Acuracia(treinamento)	0.06413604
	Acuracia(test)	0.07365449
0.9	Acuracia(treinamento)	0.06607507
	Acuracia(test)	0.07645274

Tabela 4: Promedio de resultados de parâmetro de aprendizado

### Resultados na arquitectura 68-3-2-2

Número de iterações		
40	Acuracia(treinamento)	0.06889392
	Acuracia(test)	0.076432
70	Acuracia(treinamento)	0.06184592
	Acuracia(test)	0.07406044
100	Acuracia(treinamento)	0.06232792
	Acuracia(test)	0.07843052

Tabela 5: Promedio de resultados de número de iterações

Tamanho de partição		
0.65	Acuracia(treinamento)	0.06336332
	Acuracia(test)	0.075546
0.70	Acuracia(treinamento)	0.06529083
	Acuracia(test)	0.07766883
0.75	Acuracia(treinamento)	0.06441361
	Acuracia(test)	0.07570814

Tabela 6: Promedio de resultados de tamanho de partição

Parâmetro momentum		
0	Acuracia(treinamento)	0.06379943
	Acuracia(test)	0.07791362
0.25	Acuracia(treinamento)	0.06665237
	Acuracia(test)	0.07619666
0.55	Acuracia(treinamento)	0.06291061
	Acuracia(test)	0.07610739
0.8	Acuracia(treinamento)	0.06406128
	Acuracia(test)	0.07501294

Tabela 7: Promedio de resultados de parâmetro momentum

Parâmetro de aprendizado		
0.1	Acuracia(treinamento)	0.06505505
	Acuracia(test)	0.07349514
0.4	Acuracia(treinamento)	0.06134683
	Acuracia(test)	0.07573992
0.7	Acuracia(treinamento)	0.06484524
	Acuracia(test)	0.07690105
0.9	Acuracia(treinamento)	0.06617657
	Acuracia(test)	0.0790945

Tabela 8: Promedio de resultados de parâmetro de aprendizado