

RELATÓRIO DE TRABALHO
SCC-0270 - INTRODUÇÃO A REDES NEURAIS
PROJETO 1

Classificação e Regressão
usando backpropagation e momentum

Alunos:

Gil Barbosa Reis N° 8532248

Leonardo Sampaio F. Ribeiro N° 8532300

Professora:

Dra. Roseli Aparecida Francelin Romero

INSTITUTO DE CIÊNCIAS MATEMÁTICAS E DE COMPUTAÇÃO
UNIVERSIDADE DE SÃO PAULO

São Carlos, SP
24 de setembro de 2016

Sumário

1	Introdução	2
2	Arquitetura	2
3	Experimentos	2
3.1	Como reproduzir	2
3.2	Variação de Parâmetros	3
4	Resultados	3
5	Considerações Finais	7

Lista de Tabelas

1	Resultados do treinamento para a rede da regressão	4
2	Resultados do treinamento para a rede da Classificação	5
3	Rede de regressão com melhor aprendizado (menor <i>loss</i>)	6
4	Redes de regressão com melhor generalização dos padrões (menor <i>validation loss</i>)	6
5	Rede de classificação com melhor aprendizado (menor <i>loss</i>)	6
6	Rede de classificação com melhor generalização dos padrões (menor <i>validation loss</i>)	6

Lista de Figuras

1	Arquitetura da rede usada para Classificação. Note o uso da ativação Softmax para esta tarefa.	2
2	Arquitetura da rede usada para Regressão.	3

1 Introdução

Este trabalho busca colocar em prática os conceitos de Redes Neurais aprendidos durante a disciplina; em especial observamos o potencial do método para as tarefas de classificação e regressão. Utilizamos um algoritmo de *backpropagation* muito utilizado na literatura, SGD(*Stochastic Gradient Descent*) e funções de ativação bem estudadas e conhecidas (*Softmax*, *Sigmoid* e *relu - rectifier linear unit*). As próximas sessões apresentam a arquitetura da rede estudada e os resultados de experimentos com diferentes hiperparâmetros.

2 Arquitetura

Ambas as redes seguem a mesma arquitetura interna, divergindo apenas nas camadas de entrada e saída. A Figura 1 tem um diagrama da arquitetura utilizada para Classificação e a Figura 2 da rede usada para Regressão. Internamente elas são compostas de duas camadas escondidas de 5 neurônios ativados por ReLU (*Rectified Linear Unit*), função de computação simples mas muito utilizada por ser eficiente e usualmente melhor que a sigmoidal para camadas internas.

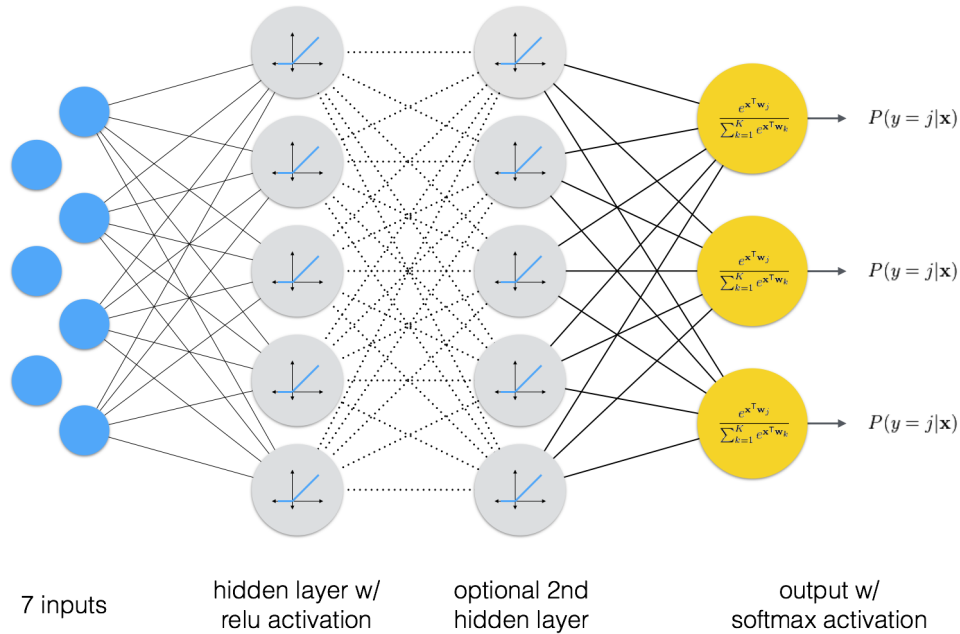


Figura 1: Arquitetura da rede usada para Classificação. Note o uso da ativação Softmax para esta tarefa.

3 Experimentos

3.1 Como reproduzir

O grupo utilizou a biblioteca Keras, criada como uma API minimalista para os frameworks Theano e TensorFlow. Escolhemos o Theanos como framework por ser mais estável quando usado com o Keras. As dependências principais da biblioteca são numpy, scipy, pyyaml e h5py. Para instalação completa da biblioteca, recomendamos que seja criado um *virtual environment* e sejam executados os comandos:

```
pip install numpy
pip install scipy
pip install pyyaml
```

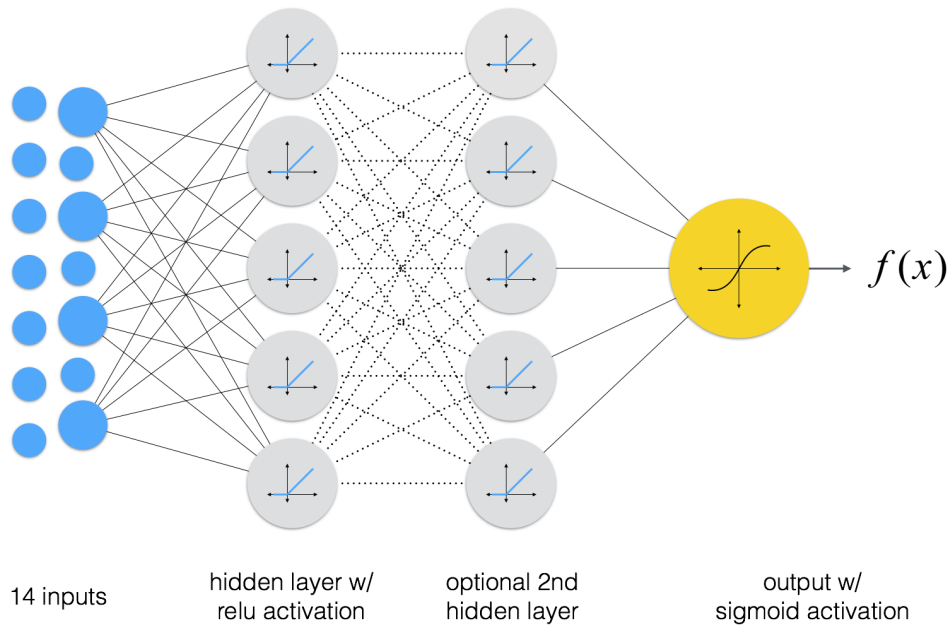


Figura 2: Arquitetura da rede usada para Regressão.

```
pip install h5py
pip install Theano
```

```
pip install keras
```

A execução dos experimentos também é simples, cd até a pasta do projeto e:

```
python main.py
```

3.2 Variação de Parâmetros

Para identificar o efeito de diferentes hiperparâmetros no resultado do treinamento, variámos as seguintes configurações da rede, uma de cada vez:

- Número de camadas intermediárias
- Taxa de Aprendizado
- Momentum
- Número de Ciclos de Treinamento
- Proporção dos dados usados para teste e validação

4 Resultados

As tabelas 1 e 2 mostram os resultados do treinamento para os vários parâmetros diferentes das redes de regressão e classificação, respectivamente.

Number of hidden layers	Epochs	Validation split	Learning rate	Momentum	Loss	Validation loss
1	1000	0.1	1e-09	0.1	31.2278177595	32.6808510638
1	1000	0.1	1e-09	0.9	31.3487684429	32.6808524842
1	1000	0.1	1e-07	0.1	41.635392364	43.8936169401
1	1000	0.1	1e-07	0.9	31.2278177458	32.6808510638
1	1000	0.1	1e-05	0.1	41.6408457585	43.8936170213
1	1000	0.1	1e-05	0.9	31.2398081535	32.6808510638
1	1000	0.3	1e-09	0.1	42.3385155863	40.7785714286
1	1000	0.3	1e-09	0.9	36.97654349	35.5745120146
1	1000	0.3	1e-07	0.1	42.3333333333	40.7785714286
1	1000	0.3	1e-07	0.9	42.348765431	40.7785714286
1	1000	0.3	1e-05	0.1	31.774691358	30.45
1	1000	0.3	1e-05	0.9	31.774691358	30.45
1	10000	0.1	1e-09	0.1	31.4434248611	32.9622370892
1	10000	0.1	1e-09	0.9	31.2278177527	32.6808510638
1	10000	0.1	1e-07	0.1	41.6474809361	43.8936167473
1	10000	0.1	1e-07	0.9	31.2398132565	32.6808517233
1	10000	0.1	1e-05	0.1	41.6474820144	43.8936170213
1	10000	0.1	1e-05	0.9	31.2278181998	32.6808512211
1	10000	0.3	1e-09	0.1	31.7793914237	30.4578448601
1	10000	0.3	1e-09	0.9	31.774831869	30.45
1	10000	0.3	1e-07	0.1	42.3487649394	40.7785350042
1	10000	0.3	1e-07	0.9	42.3487654321	40.7785714286
1	10000	0.3	1e-05	0.1	31.7901234568	30.45
1	10000	0.3	1e-05	0.9	42.3333339746	40.7785714286
2	1000	0.1	1e-09	0.1	32.7935319771	35.186290675
2	1000	0.1	1e-09	0.9	31.2384601353	32.6969261169
2	1000	0.1	1e-07	0.1	31.2282320997	32.680964338
2	1000	0.1	1e-07	0.9	31.2278177458	32.6808510638
2	1000	0.1	1e-05	0.1	41.6474820144	43.8936170213
2	1000	0.1	1e-05	0.9	31.2278177572	32.6808510638
2	1000	0.3	1e-09	0.1	31.7746921866	30.45
2	1000	0.3	1e-09	0.9	42.3487654321	40.7785714286
2	1000	0.3	1e-07	0.1	42.3487632333	40.7785714286
2	1000	0.3	1e-07	0.9	42.3487654321	40.7785714286
2	1000	0.3	1e-05	0.1	42.3487654321	40.7785714286
2	1000	0.3	1e-05	0.9	31.8219460342	30.481032343
2	10000	0.1	1e-09	0.1	41.6474820144	43.8936170213
2	10000	0.1	1e-09	0.9	31.2278177458	32.6808510638
2	10000	0.1	1e-07	0.1	31.227857345	32.6808510638
2	10000	0.1	1e-07	0.9	31.2278205348	32.6808510638
2	10000	0.1	1e-05	0.1	41.6474820144	43.8936170213
2	10000	0.1	1e-05	0.9	31.2278177458	32.6808510638
2	10000	0.3	1e-09	0.1	31.8456498494	30.4929107241
2	10000	0.3	1e-09	0.9	36.613690315	35.1571229935
2	10000	0.3	1e-07	0.1	31.7816558243	30.4500000817
2	10000	0.3	1e-07	0.9	31.7901253465	30.4500005535
2	10000	0.3	1e-05	0.1	31.7746921792	30.45
2	10000	0.3	1e-05	0.9	42.3487654321	40.7785714286

Tabela 1: Resultados do treinamento para a rede da regressão

Number of hidden layers	Epochs	Validation split	Learning rate	Momentum	Loss	Validation loss
1	1000	0.1	1e-09	0.1	4.65042301886	0.16448897762
1	1000	0.1	1e-09	0.9	2.13459493531	0.205786863431
1	1000	0.1	1e-07	0.1	1.99240061191	4.5895781517
1	1000	0.1	1e-07	0.9	1.09014299622	1.08798020794
1	1000	0.1	1e-05	0.1	0.743302127908	1.03291565747
1	1000	0.1	1e-05	0.9	0.157948497697	0.539643923042
1	1000	0.3	1e-09	0.1	8.4382381423	16.0375367876
1	1000	0.3	1e-09	0.9	8.31945196747	13.2031213821
1	1000	0.3	1e-07	0.1	3.69076581451	0.984492667138
1	1000	0.3	1e-07	0.9	0.787470547318	3.27344236298
1	1000	0.3	1e-05	0.1	0.408875179407	2.12082453569
1	1000	0.3	1e-05	0.9	0.204828913225	0.869746240713
1	10000	0.1	1e-09	0.1	5.76763293514	16.1180953979
1	10000	0.1	1e-09	0.9	1.79299218046	2.30914832864
1	10000	0.1	1e-07	0.1	1.04424314398	1.58854038942
1	10000	0.1	1e-07	0.9	0.74354789276	0.780896206697
1	10000	0.1	1e-05	0.1	0.493798189437	0.782380918662
1	10000	0.1	1e-05	0.9	0.0572737164354	0.842883232842
1	10000	0.3	1e-09	0.1	3.96831095604	0.0982260471062
1	10000	0.3	1e-09	0.9	1.8709994638	9.8700873057
1	10000	0.3	1e-07	0.1	1.07846577151	1.14968633652
1	10000	0.3	1e-07	0.9	0.728004142541	1.70076556953
1	10000	0.3	1e-05	0.1	0.277923048111	2.69823680984
1	10000	0.3	1e-05	0.9	0.0719952543135	0.588487792511
2	1000	0.1	1e-09	0.1	1.54825145322	0.888648282914
2	1000	0.1	1e-09	0.9	1.41656816526	1.43296200321
2	1000	0.1	1e-07	0.1	1.02185559635	1.92271499407
2	1000	0.1	1e-07	0.9	1.08248799913	1.24054038525
2	1000	0.1	1e-05	0.1	1.08946295579	1.21365270728
2	1000	0.1	1e-05	0.9	0.199744720261	0.411574090232
2	1000	0.3	1e-09	0.1	8.05014169984	7.55047801941
2	1000	0.3	1e-09	0.9	8.07612658616	0.00644011501341
2	1000	0.3	1e-07	0.1	1.50554338016	1.6712057704
2	1000	0.3	1e-07	0.9	1.07607070199	1.1033100325
2	1000	0.3	1e-05	0.1	0.549565103336	2.38215727749
2	1000	0.3	1e-05	0.9	0.238774631367	1.75236418323
2	10000	0.1	1e-09	0.1	1.90749146793	4.37840983981
2	10000	0.1	1e-09	0.9	1.79156062669	4.57608423914
2	10000	0.1	1e-07	0.1	0.767303552735	0.802334860677
2	10000	0.1	1e-07	0.9	10.1484304799	16.1180953979
2	10000	0.1	1e-05	0.1	1.08572677519	1.34920394421
2	10000	0.1	1e-05	0.9	1.08575358498	1.37702664875
2	10000	0.3	1e-09	0.1	7.80227544029	1.12003275516
2	10000	0.3	1e-09	0.9	2.75036978306	1.4633467292
2	10000	0.3	1e-07	0.1	1.07956205096	1.14478886127
2	10000	0.3	1e-07	0.9	0.640413055006	2.77662606466
2	10000	0.3	1e-05	0.1	0.285357107067	2.05820527909
2	10000	0.3	1e-05	0.9	0.0886417903964	0.722294584153

Tabela 2: Resultados do treinamento para a rede da Classificação

A partir dos valores encontrados de erro no treinamento e na validação, podemos dizer quais parâmetros foram melhores para cada problema:

- Regressão: Neste caso podemos notar que, infelizmente, os casos com melhor generalização na verdade tiveram o problema de "Gradiente Desaparecido" (*vanishing gradient*), onde a rede leva a função objetivo até próximo de um ponto onde o gradiente zera e a *backpropagation* deixa de ter efeito nos pesos, estagnando a perda. Uma opção para resolver esse problema seria diminuir mais a taxa de aprendizado. Não fizemos este teste por falta de tempo (treinar com uma taxa de aprendizado mais baixa usaria mais ciclos para treino).

Number of hidden layers	Epochs	Validation split	Learning rate	Momentum	Loss	Validation loss
1	10000	0.1	1e-09	0.9	31.2278177527	32.6808510638

Tabela 3: Rede de regressão com melhor aprendizado (menor *loss*)

Number of hidden layers	Epochs	Validation split	Learning rate	Momentum	Loss	Validation loss
1	1000	0.3	1e-05	0.1	31.77469136	30.45
1	1000	0.3	1e-05	0.9	31.77469136	30.45
2	10000	0.3	1e-05	0.1	31.77469218	30.45
2	1000	0.3	1e-09	0.1	31.77469219	30.45
1	10000	0.3	1e-09	0.9	31.77483187	30.45
1	10000	0.3	1e-05	0.1	31.79012346	30.45

Tabela 4: Redes de regressão com melhor generalização dos padrões (menor *validation loss*)

- Classificação:

Number of hidden layers	Epochs	Validation split	Learning rate	Momentum	Loss	Validation loss
1	10000	0.1	1e-05	0.9	0.0572737164354	0.842883232842

Tabela 5: Rede de classificação com melhor aprendizado (menor *loss*)

Number of hidden layers	Epochs	Validation split	Learning rate	Momentum	Loss	Validation loss
2	1000	0.3	1e-09	0.9	8.07612658616	0.00644011501341

Tabela 6: Rede de classificação com melhor generalização dos padrões (menor *validation loss*)

5 Considerações Finais

Com esse experimento pudemos perceber como a mudança da topologia e dos parâmetros das redes neurais construídas causa grandes impactos nos resultados; concluímos que é importante testar as várias possibilidades antes de se aplicar uma rede neural na prática.