

CONHEÇA O MACHINE LEARNING

ALGORITMOS QUE APRENDEM A PARTIR DE DADOS, IMAGENS, E TEXTO

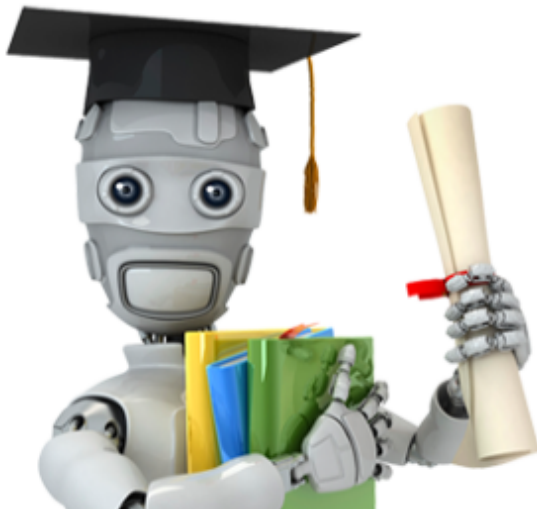
Júlio C. Batista

julio.batista@outlook.com

Agosto 14, 2017



O que é o *machine learning* [1]?



[1] Andrew Ng. *Machine Learning* - Coursera. <https://www.coursera.org/learn/machine-learning>. Accessed: 2017-08-10.

Definindo o "aprendizado"

- Dado um conjunto de dados \mathcal{D} e um modelo \mathcal{M}
- Definimos uma função objetivo (*loss/cost*) \mathcal{L}
- Minimizamos $\mathcal{L}(\mathcal{M}; \mathcal{D})$

- Um método comum de otimização é o *Stochastic Gradient Descent* (SGD) com *momentum*;

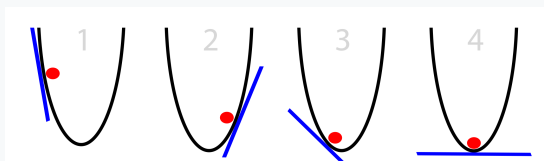
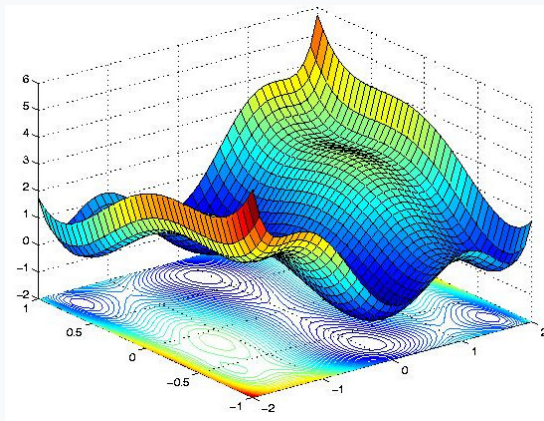


Figure: Exemplo do SGD [1].

[1] Andrew Trask. *A Neural Network in 13 lines of Python (Part 2 - Gradient Descent)*.
<https://iamtrask.github.io/2015/07/27/python-network-part2/>. Accessed: 2017-04-27.

Minimizando \mathcal{L} [1]



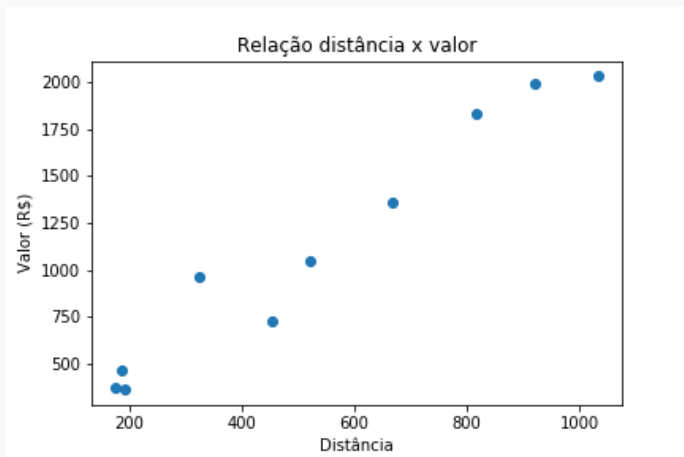
[1] Alexander Amini. *Projects*. <http://www.mit.edu/~amini/index.html>. Accessed: 2017-08-10.

Caso 1

Dada a tabela abaixo (\mathcal{D}), como podemos estimar o valor (\mathcal{M}) de uma viagem com base na distância?

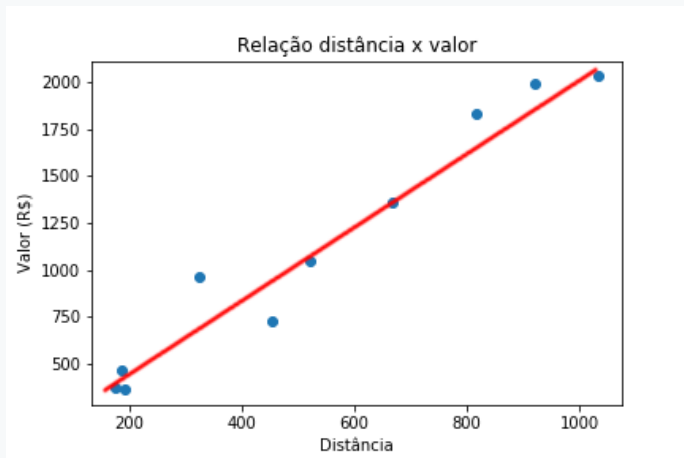
Distancia (km)	Custo (R\$)
176.13	370.40
187.05	464.42
192.10	364.23
325.31	962.69
453.10	727.68
521.72	1045.96
667.57	1356.53
816.79	1828.42
921.20	1993.64
1032.51	2029.67

Visualizando os dados



Regressão linear

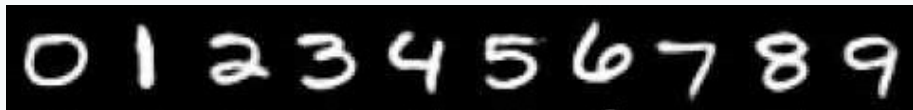
- $\mathcal{M}(x) = mx + b$
- $\mathcal{L}(\mathcal{M}; \mathcal{D}) = \frac{1}{|\mathcal{D}|} \|\mathcal{M}(\mathcal{D}_x) - \mathcal{D}_y\|_2^2$



- Definimos o modelo de regressão linear $\mathcal{M}(x) = mx + b$
- Definimos uma função objetivo $\mathcal{L}(\mathcal{M}; \mathcal{D}) = \frac{1}{|\mathcal{D}|} \|\mathcal{M}(\mathcal{D}_x) - \mathcal{D}_y\|_2^2$
- Encontramos os parâmetros m e b que minimizam o erro de \mathcal{M}

Caso 2

Dadas as imagens [1] abaixo, como podemos identificar o dígito na imagem?



[1] Juan Cazala. *mnist repository*. <https://github.com/cazala/mnist>. Accessed: 2017-08-10.

Representando imagens

- Imagens são matrizes onde os seus valores (*pixels*) variam de 0 (preto) a 255 (branco);



$$\mathbf{x}_{28 \times 28} = \begin{bmatrix} 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & 255 & 255 & \ddots & \vdots \\ \vdots & 255 & 255 & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix}$$

Computando características da imagem

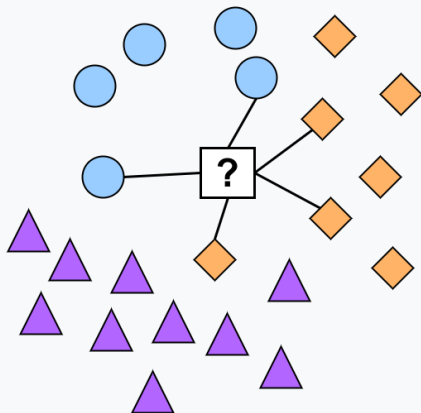
- Uma imagem I , geralmente, possui alta dimensionalidade
- Diminuimos a dimensionalidade calculando características através de descritores
- *Histogram of Oriented Gradients*(HOG) [1] é um descritor de imagens

$$\mathbf{x} = \text{hog}(I)$$

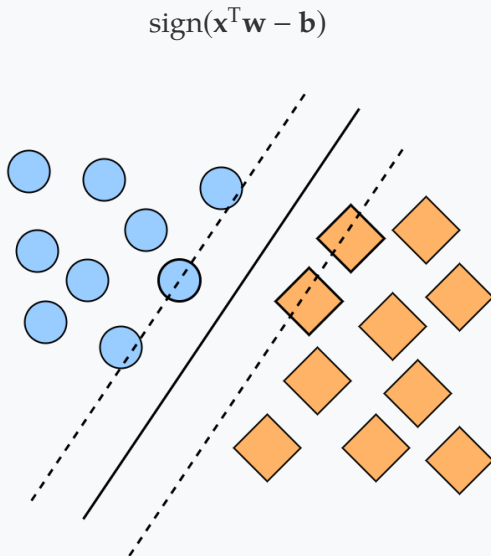
[1] Navneet Dalal and Bill Triggs. "Histograms of oriented gradients for human detection". In: *IEEE CVPR*. 2005.

K-Nearest Neighbors

$$\operatorname{argmin}_{\mathbf{p} \in \mathcal{D}} \|\mathbf{x} - \mathbf{p}\|_2^2$$



Support Vector Machines



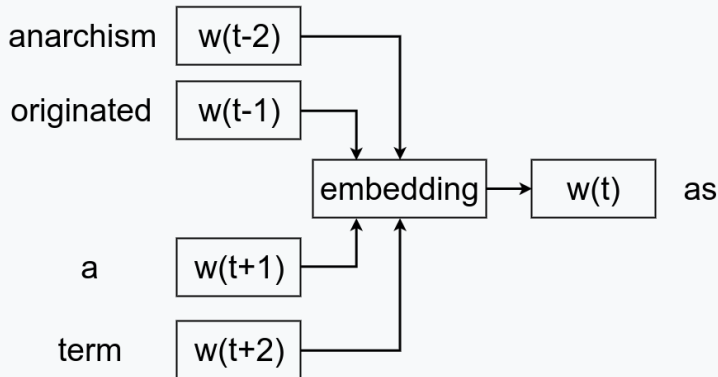
Caso 3: aprendendo o significado de palavras

Dado o texto [1] abaixo, como podemos aprender representações das palavras?

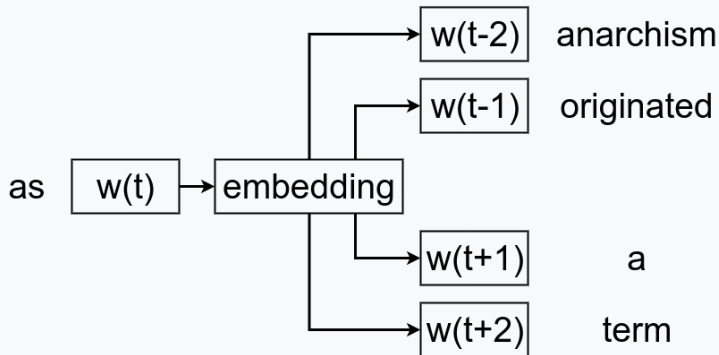
anarchism originated as a term of abuse first used against earl

[1] Matt Mahoney. *About the Test Data*. <http://mattmahoney.net/dc/textdata>. Accessed: 2017-08-10.

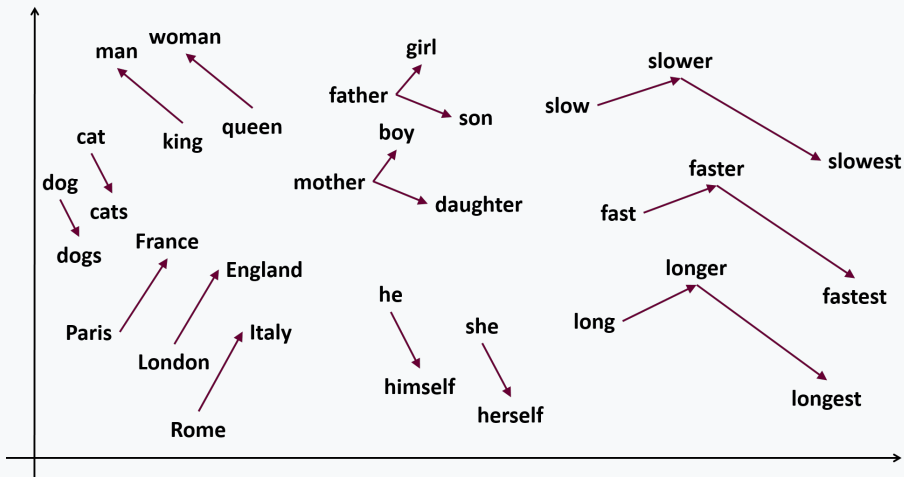
Word2Vec: Continuous Bag of Words (CBOW)



Word2Vec: Skip-gram



Visualizando os vetores [1]



[1] Samy Zafrany. NLP with gensim (word2vec). <http://www.samyzaf.com/ML/nlp/nlp.html>. Accessed: 2017-08-10.

- Depois que o modelo "aprendeu" as representações
- É possível efetuar operações matemáticas com os vetores
- **rei - homem = rainha**
- **cachorro + bebê = filhote**

PARTE 2

Deep learning e as redes neurais convolucionais

- Redes neurais foram muito utilizadas até os anos 2000;
- Houve um hiato relacionado as redes neurais com o uso de outros algoritmos (SVM, árvores de decisão);
- Em 2012, um resultado [1] trouxe muita atenção para as redes neurais convolucionais;

Model	Top-1	Top-5
Sparse coding	47.1%	28.2%
SIFT + FVs	45.7%	25.7%
CNN	37.5%	17.0%

Table: Porcentagem de erro por modelo

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Advances In Neural Information Processing Systems* (2012).

Exemplo de uma rede convolucional

- As camadas *conv* e *fc* constituem de: uma matriz de *weights* (\mathbf{W}) e um vetor de *bias* \mathbf{b}

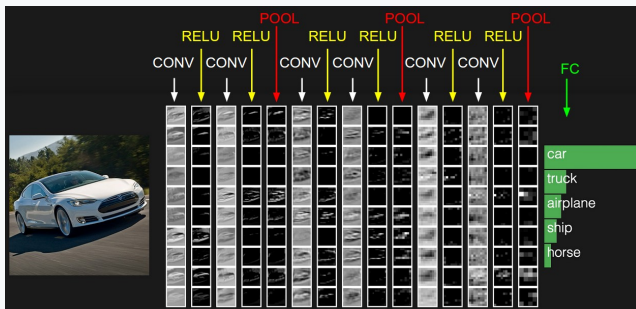


Figure: Exemplo de uma rede neural convolucional [1].

[1] CS231n. Convolutional Neural Networks (CNNs / ConvNets). <http://cs231n.github.io/convolutional-networks/>. Accessed: 2017-04-27.

Convoluções: o processo de filtrar uma imagem

- A rede "aprende as melhores" convoluções ajustando W e b durante o processo de aprendizado

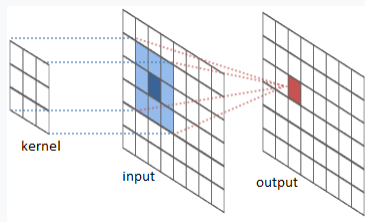


Figure: Processo de convolução [1].

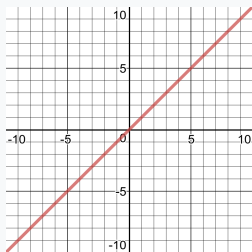
[1] Christopher Olah. *Understanding Convolutions*. <http://colah.github.io/posts/2014-07-Understanding-Convolutions/>. Accessed: 2017-04-27.

- *Bias* **b**: constante 0
- *Weights* **W**:
 - Normal (Gaussiano) com $\mu = 0$ e $\sigma = 0.05$
 - Xavier [1]
 - MSRA [2]

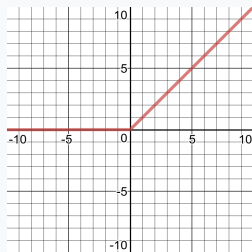
[1] Xavier Glorot and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks." In: *Aistats*. 2010.

[2] Kaiming He et al. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". In: *ICCV*. 2015.

- Basicamente: transforma uma reta em uma curva
- A mais comum atualmente é a *Rectified Linear Unit* (ReLU) [1]



(a) Linear: $y = \mathbf{Wx} + \mathbf{b}$

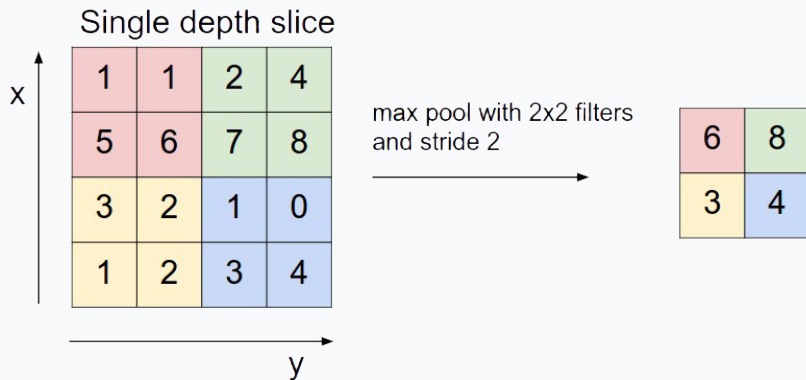


(b) ReLU: $y = \max(0, \mathbf{Wx} + \mathbf{b})$

[1] Vinod Nair and Geoffrey E Hinton. "Rectified linear units improve restricted boltzmann machines". In: *ICML*. 2010.

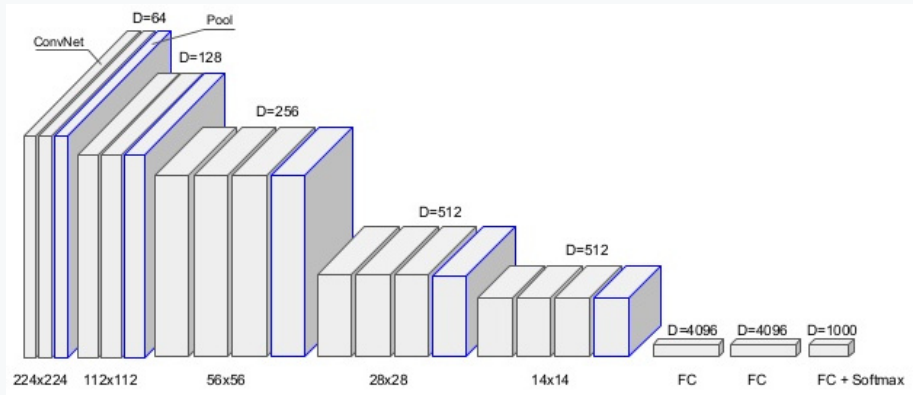
Pooling

- Reduz o tamanho (altura, largura) da representação;
- Tipos mais comuns: *max*, *average*;



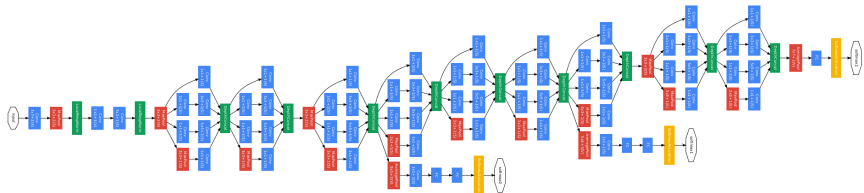
[1]

VGG16 [1]



[1] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556* (2014).

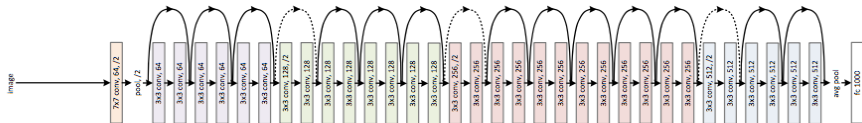
GoogLeNet [1]



[1] C. Szegedy et al. "Going deeper with convolutions". In: *IEEE CVPR*. 2015.

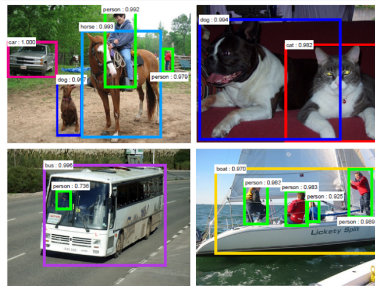
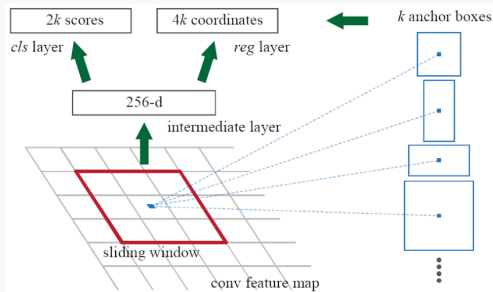
ResNet [1]

34-layer residual



[1] Kaiming He et al. "Deep residual learning for image recognition". In: *IEEE CVPR*. 2016.

Faster R-CNN [1]



[1] Shaoqing Ren et al. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". In: *IEEE TPAMI* (2016).

Para onde ir e outros frameworks

- Existem vários cursos gratuitos pela internet: [CS231n \(Stanford\)](#), [CS224n \(Stanford\)](#), [Deep Learning by Google \(Udacity\)](#), [Neural Networks for Machine Learning \(University of Toronto\)](#), [Deep Learning \(Oxford\)](#), [Neural Networks class \(Université de Sherbrooke\)](#)
- Livros: [Deep Learning Book](#), [Neural Networks and Deep Learning](#)
- Frameworks: [Keras](#), [Theano](#), [Torch](#), [pyTorch](#), [Caffe](#), [Caffe2](#), [Lasagne](#)

- Google Brain Residency Program
- Deep Mind
- Facebook AI Research
- Microsoft Research
- IBM
- Mestrado/Doutorado no IMAGO-UFPR Research Group;

CONHEÇA O MACHINE LEARNING

ALGORITMOS QUE APRENDEM A PARTIR DE DADOS, IMAGENS, E TEXTO

Júlio C. Batista

julio.batista@outlook.com

Agosto 14, 2017

