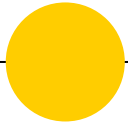


Yoga Health and Machine Learning



Projeto Final Iron Hack
Robson Silva da Silva / 7 feb 2020

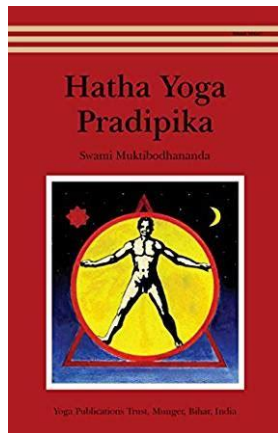


Nas tradições milenares do Yoga

Corpo e Mente é uma coisa só.

O **corpo** é um aspecto grosseiro da **mente**.

A **mente** é um aspecto sutil do **corpo**.



Pela prática correta de
Pranayamas, qualquer **doença**
pode ser erradicada.

Porém pela prática incorreta
qualquer **doença** pode surgir.

“

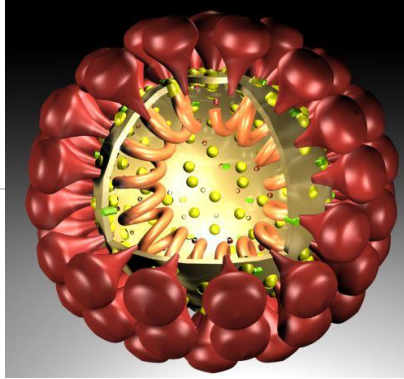
Rsi Swatmarama



Pneumonia no Mundo

- “De acordo com o relatório Save the Children, divulgado na semana passada - 2017 - , duas crianças morrem a cada minuto devido à pneumonia.”
- “Em 2015, foram 920 mil óbitos, a maioria em países pobres do sul da Ásia e da África Subsaariana, daí a alcunha de "a doença da pobreza", conforme a ONG.”

Coronavirus



Pneumonia

Microsoft no Programa “AI for Health” liberou 40 milhões de Dólares para Pesquisa em em saúde e Inteligência Artificial. (29/jan/20)

John Kahan - Chief Data Analytics Office

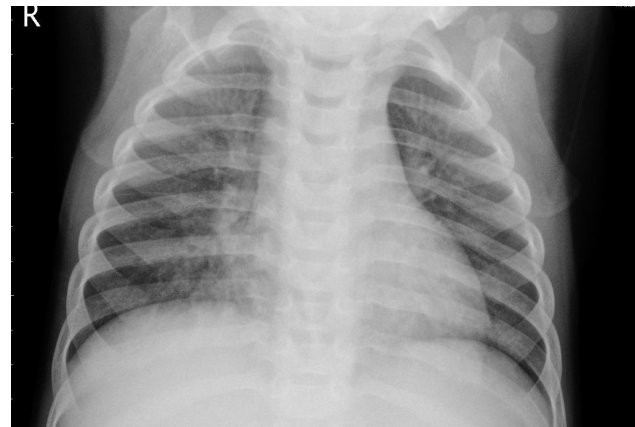
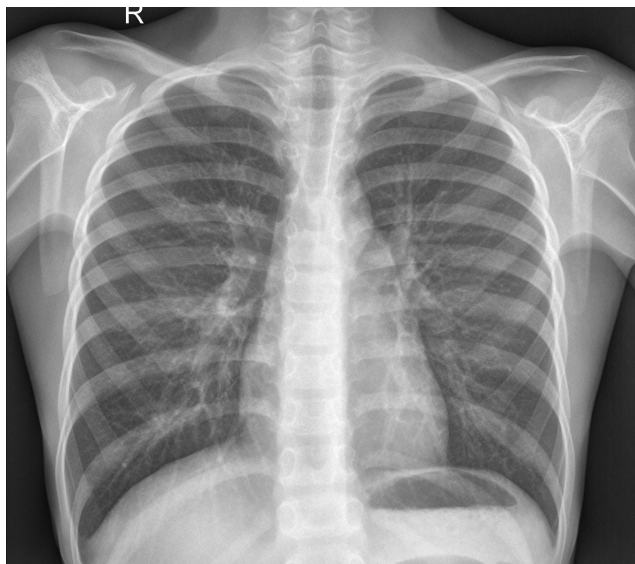


- *Quest for discovery.* Accelerating medical research to advance **prevention**, **diagnoses** and **treatment** of diseases;
- *Global health insights.* Increasing our shared understanding of mortality and longevity to protect against global health crises;
- *Health equity.* Reducing health inequity and improving access to care for underserved populations

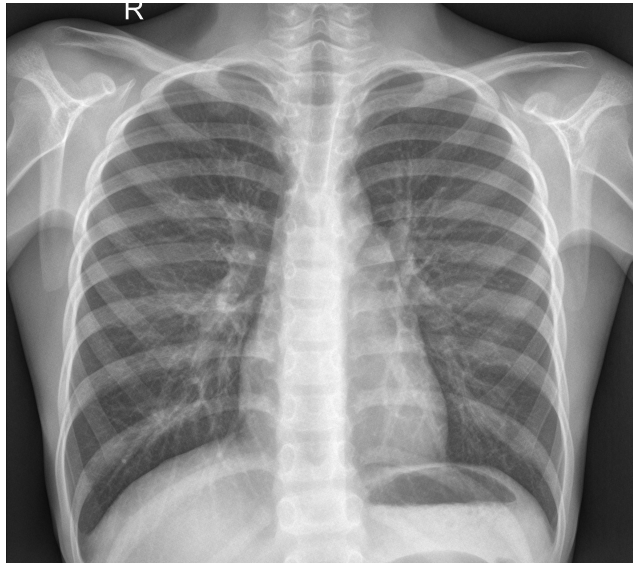
Fonte: <https://blogs.microsoft.com/on-the-issues/2020/01/29/ai-for-health-child-mortality/?linkId=81414123>

Identificando Pneumonia em Imagens de Raio-X

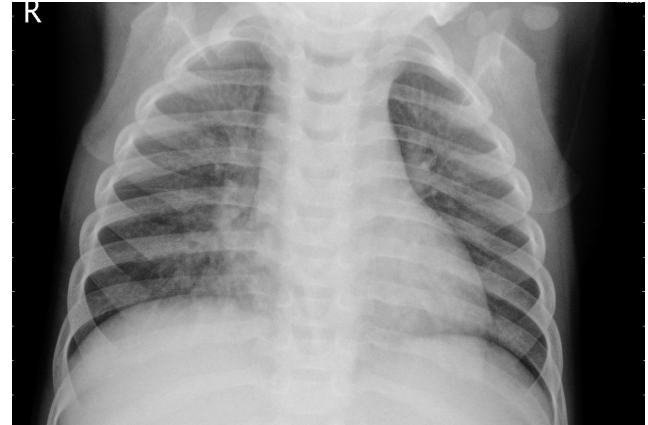
Usando Convolução com Pytorch,
XGBoost e LightGBM
com Principal Component Analysis (PCA)



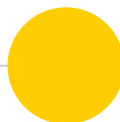
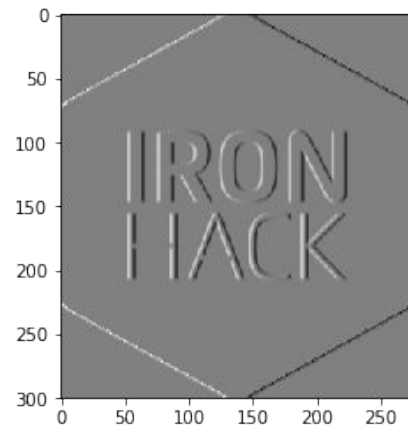
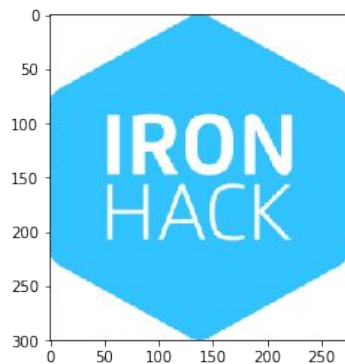
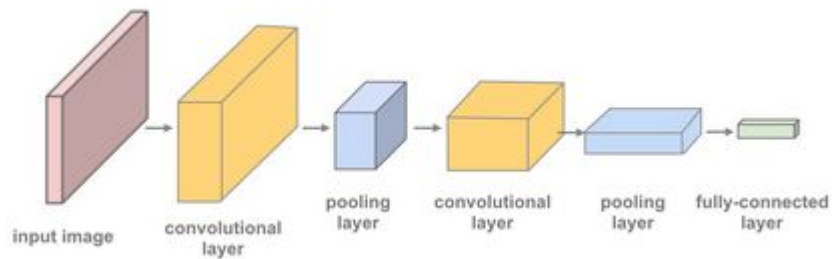
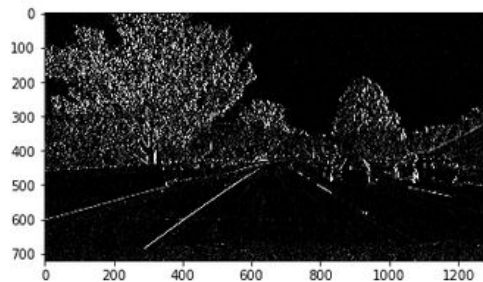
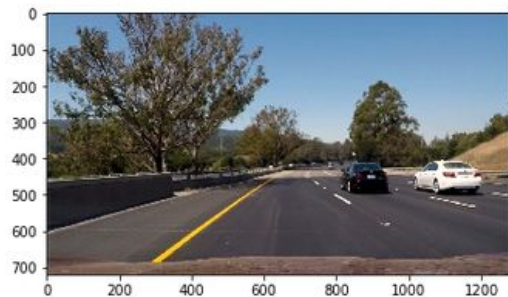
Normal



Pneumonia

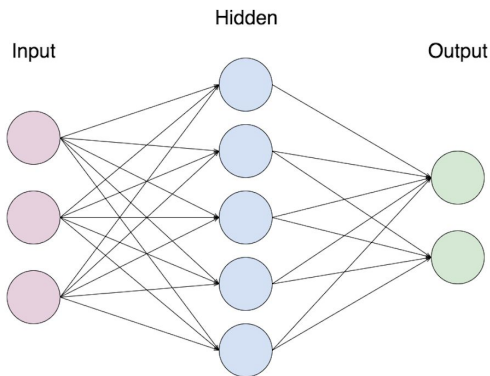


Convolução de Imagens



Nosso modelo em Pytorch

Input Vector com **115200** colunas.



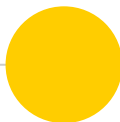
Hidden Layers = (256,512,512,64)

```
input_vector = 128*30*30
```

```
# define the CNN architecture
class Rob(nn.Module):
    def __init__(self):
        super(Rob, self).__init__()
        # convolutional layer
        self.conv1 = nn.Conv2d(3, 16, 3, padding=1)
        self.conv2 = nn.Conv2d(16, 32, 3, padding=1)
        self.conv3 = nn.Conv2d(32, 128, 3, padding=1)

        # MLP layers
        self.fc1 = nn.Linear(input_vector, 256)
        self.fc2 = nn.Linear(256, 512)
        self.fc3 = nn.Linear(512, 512)
        self.fc4 = nn.Linear(512, 64)
        self.fc5 = nn.Linear(64, 2)


        self.pool = nn.MaxPool2d(2, 2)
        self.dropout = nn.Dropout(0.5)
```

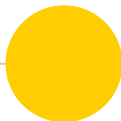


Nvidia GeForce 930MX

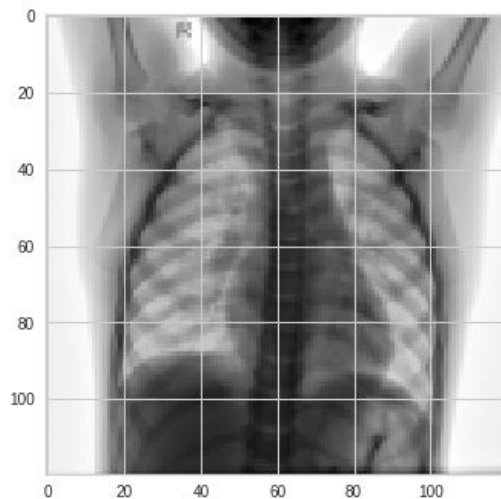


XGBoost e LightGBM

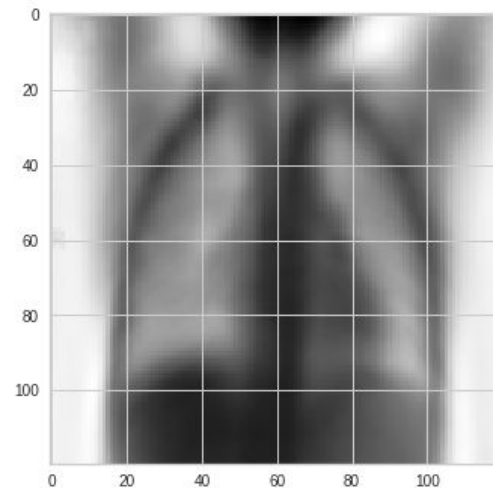
- Transformamos as imagens para 120x120 via biblioteca do Pytorch.
 - Aplicamos uma PCA reduzindo de $120 \times 120 \times 3$ (= **43.200**) para **36** colunas mantendo **82%** da informação.
- 



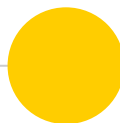
Efeito do PCA na Imagem



Antes do PCA





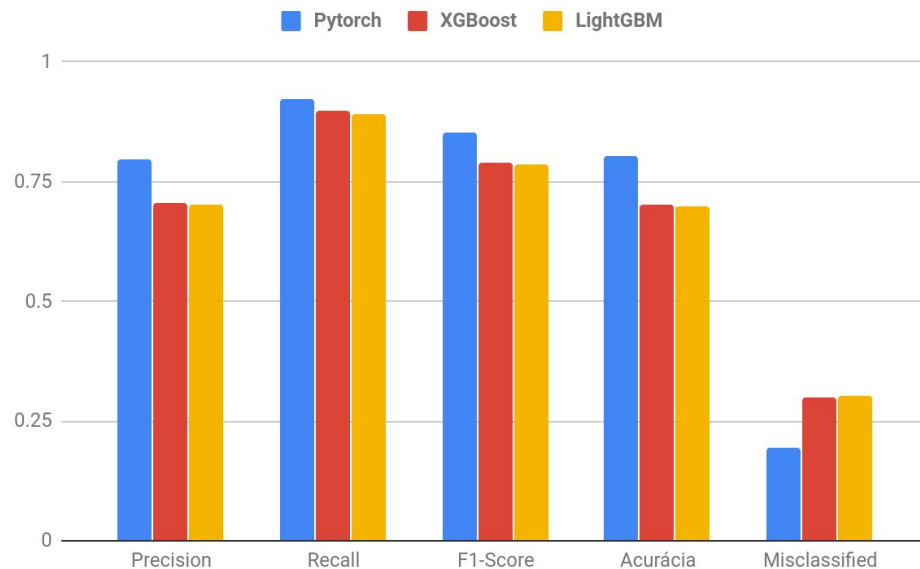
Inverse Transform





RESULTADOS

	 PyTorch	<i>XGBoost</i>	LightGBM 
Precision	0.796	0.704	0.703
Recall	0.922	0.897	0.892
F1-Score	0.854	0.788	0.786
Acurácia	0.805	0.702	0.698
Misclassified	0.195	0.298	0.302



Deployment





Muito obrigado!

Todo material estará disponível no meu GitHub!

robsonsilvadasilva@gmail.com

[github.com/](https://github.com/robsonsilvadasilva)[robsonsilvadasilva](https://github.com/robsonsilvadasilva)

[linkedin.com/in/](https://linkedin.com/in/robsonsilvadasilva)[robsonsilvadasilva](https://linkedin.com/in/robsonsilvadasilva)