# Os arquivos entregues, para fins de resolução do teste, são compostos por:

- os arquivos README.docx e README.md, ambos com os mesmos conteúdos e só diferindo em extensão para facilitar a análise do que foi desenvolvido. Estes arquivos explicam em linhas gerais tudo o que foi feito, neste teste. Estes arquivos também contém a importante sessão denominada de ‘# Instalação do ambiente virtual’, que contém todos os comandos necessários, para a instalação (em ambiente Windows 10), do zero, de um ambiente virtual que execute o arquivo sex\_predictor.py

- o arquivo contendo o modelo treinado, serializado, pela biblioteca pickle, que é chamado no arquivo .py para fazer previsões em outros datasets. Este arquivo chama-se finalized\_model1.sav.

- o arquivo requirements.txt, que contem as versões de todas as dependências do que é necessário para se instalar, no ambiente virtual, para que se possa fazer previsões com o modelo treinado, por meio do arquivo sex\_predictor.py

- o arquivo sex\_predictor.py que irá fazer previsões em novos datasets contidos em arquivos denominados newsample.csv. Para que o arquivo sex\_predictor.py funcione corretamente, é necessário que contenha as mesmas variáveis, no mesmo formato, do arquivo fornecido (denominado test\_data\_CANDIDATE(2).csv) por email inicialmente pelo time da Portal Telemedicina. Esse arquivo também executa diversas atividades antes de fazer as previsões e salvá-las em um arquivo. Para maiores informações sobre este arquivo, vide a sessão abaixo.

- o arquivo denominado newsample.csv, contando exatamente o mesmo conteúdo do arquivo test\_data\_CANDIDATE(2).csv, que foi utilizado para testar o código e simular novas amostras que virão para testar o modelo.

-o notebook [Alfredo's Solutions to Portal Telemedicina Challenge-1-FINAL.ipynb](http://localhost:8888/notebooks/Desafio_Portal_Telemedicina/Alfredo's%20Solutions%20to%20Portal%20Telemedicina%20Challenge-1-FINAL.ipynb) contendo todas as etapas de importação do dataset ‘test\_data\_CANDIDATE(2).csv’, análise exploratória dos dados, gráficos, data wrangling, resampling, treinamento dos classificadores preditivos, validação dos modelos e demais comentários pormenorizados sobre tudo o que diz respeito à construção/ estimação do modelo final. Este notebook também salva em arquivo o modelo final, serializado na extensão .sav para que possa ser consumido por outros serviços/ códigos!

- o arquivo originalmente fornecido, test\_data\_CANDIDATE(2).csv, utilizado no notebook, com o qual os classificadores foram treinados e validados.

- o arquivo newsample\_PREDICTIONS\_{Alfredo\_Ricardo\_de\_Faria\_Passos}.csv, que contém as previsões do modelo final, que é chamado no código do arquivo sex\_predictor.py. Essas previsões foram obtidas com o arquivo sex\_predictor.py fazendo-se previsões no arquivo newsample.csv que eu encaminho (que na verdade contém os mesmos dados do arquivo ‘test\_data\_CANDIDATE(2).csv’) e foi usado para testar o arquivo .py.

- o arquivo Teste.docx que contém as instruções dadas por email para o Desafio.

# Instalação do ambiente virtual

A criação de um ambiente virtual, com a instalação no mesmo de todos os pacotes necessários para se rodar o arquivo sex\_predictor.py (para que se faça previsões de sexo utilizando-se como input um arquivo denominado newsample.csv); assim como a criação de um arquivo requirements.txt com todas as bibliotecas/dependências para tanto, foi realizada em um sistema Windows 10.

Antes de se proceder aos comandos utilizados, deve-se ressaltar que utilizei a seguinte versão do Python/ Anaconda:

conda version : 4.6.2

conda-build version : 3.17.6

python version : 3.7.1.final.0

Portanto, ou a versão do Python ou do Anaconda correspondentes devem ser instalados, ANTES da execução dos comandos a seguir.

A seguir lista-se os comando realizados em prompt cmd, na ordem realizada para tanto.

1. pip install virtualenv
2. pip install virtualenvwrapper-win
3. utilzei o comando mkvirtualenv HelloWold para criar um virtualenv chamado HelloWold (no Windows, esse comando cria esse virtual env pip install dentro da pasta “C:\Users\<nome do suário>\Envs”; no meu caso, na pasta “C:\Users\Alfredo Ricardo\Envs\HelloWold”).
4. Ativação do ambiete virtual criado:

C:\Users\Alfredo Ricardo>cd Envs

C:\Users\Alfredo Ricardo\Envs>cd HelloWold

C:\Users\Alfredo Ricardo\Envs\HelloWold>cd Scripts

C:\Users\Alfredo Ricardo\Envs\HelloWold\Scripts>activate

1. Criação de um diretório para ser o diretório-raiz do ambiente virtual criado. Isso envolveu os seguintes comandos:

cd..

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold> mkdir dev

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold>cd dev

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold\dev>mkdir HelloWold

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold\dev>cd HelloWold

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold\dev\HelloWold>setprojectdir

1. Instalação das bibliotecas necessárias para se rodar o arquivo sex\_predictor.py nesse ambiente virtual:

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold\dev\HelloWold>pip install sklearn

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold\dev\HelloWold>pip install pandas

Ao se instalar essas duas bibliotecas, todas as outras dependências das mesmas são automaticamente instaladas.

1. Visualização das dependências ou salvamento das dependências em um arquivo .txt:

Para visualizar as dependências instaladas no virtual env, pode-se executar o comando pip freeze requirements.txt (no meu caso específico seria:

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold\dev\HelloWold>pip freeze requirements.txt

)

Para se salvar as dependências em um arquivo requirements.txt, no diretório-raiz do ambiente virtual, executei o comando:

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold\dev\HelloWold>pip freeze > requirements.txt

1. Após essas etapas é necessário se colocar no diretório raiz do ambiente virtual (que no caso deste exemplo foi: C:\Users\Alfredo Ricardo\Envs\HelloWold\dev\HelloWold) os seguintes arquivos:

- arquivo serializado com extensão .sav contendo o modelo treinado e serializado no notebook. No caso deste exercício, o arquivo a ser colocado lá chama-se finalized\_model1.sav.

- arquivo contendo os novos dados, chamado newsample.csv

- arquivo .py contendo as rotinas de leitura e data wrangling dos dados a serem imputados no modelo (dados contidos em um arquivo chamado newsample.csv). No caso aqui este arquivo chama-se sex\_predictor.py. Este arquivo também executa o método predict para fazer previsões, do modelo treinado nos dados deste arquivo .csv.

De forma mais detalhada o que este arquivo .py faz o seguinte:

1. Importa as bibliotecas-base necessárias (pandas, sklearn e pickle). A partir dessas bibliotecas-base, importa outros métodos.
2. Lê o objeto serializado .sav, onde o modelo treinado se encontra, utilizando para isso a biblioteca pickle.
3. Lê o arquivo newsample.csv como um pandas dataframe.
4. Realiza algumas atividades de data wrangling neste dataframe:

- na coluna sex, transforma os labels minúsculos em mipusculos

- elimina as colunas ‘cp’, ‘index’ e ‘slope’ (para mais detalhe do porquê dessas ações vide o notebook),

- Transforma o tipo dos valores da coluna ‘sex’ em tipo string.

- Aplica um método de pré-processamento do sklearn conhecido como LabelEncoder na ‘sex’.

- Muda o tipo da coluna ‘age’ para int.

- Elimina as demais linhas de dados do dataframe com missing values.

- define os conjuntos y-test e X-test a partir do dataframe anterior. O conjunto X\_test (que contém somente as features remanescentes do problema) é o que servirá de input para o modelo.

1. Aplica o método predict no conjunto X\_test e salva o resultado deste método em uma variável chamada prediction2.
2. Altera os valores da variável predcit2, substituindo os valores 1 pela letra ‘M’ e os valores 0 pela letra ‘F’.
3. Salva os valores deste dataframe (conjuntamente om o seu título, ‘sex’) no diretório raiz do virtual env.

Uma vez realizadas estas etapas, pode-se rodar no cmd do Windows o comando:

python sex\_predictor.py --newsample.csv

Este commando tem que ser realizado com o ambiente virtual ativo. NO caso do exemplo, tem-se:

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold\dev\HelloWold>python sex\_predictor.py --newsample.csv

Alternativamente pode-se rodar simplesmente:

(HELLOW~1) C:\Users\Alfredo Ricardo\Envs\HelloWold\dev\HelloWold>python sex\_predictor.py

Estes comandos gerarão no diret´´orio-raiz do ambiente virtual o arquivo:

newsample\_PREDICTIONS\_{Alfredo\_Ricardo\_de\_Faria\_Passos}.csv

Finalmente, o arquivo requirements.txt irá apresentar o seguinte conteúdo:

joblib==0.13.2

numpy==1.16.3

pandas==0.24.2

python-dateutil==2.8.0

pytz==2019.1

scikit-learn==0.21.0

scipy==1.2.1

six==1.12.0

sklearn==0.0

Sendo estas as dependências (além é claro da versão do Python que deve ser instaladas ANTES dos procedimentos descritos aqui).:

# Arquivos que devem estar no ambiente virtual, após a sua instalação e ativação, para que se possa fazer previsões com o arquivo sex\_predictor.py:

Após a instalação e ativação do ambiente virtual, para que se possa realizar previsões com o modelo treinado, é necessário que, no ambiente virtual estejam os seguintes arquivos:

- o arquivo contendo o modelo treinado, serializado, pela biblioteca pickle, que é chamado no arquivo .py para fazer previsões em outros datasets. Este arquivo chama-se finalized\_model1.sav.

- o arquivo sex\_predictor.py.

- o arquivo denominado newsample.csv, contando novos samples, com as mesmas colunas e formatos do arquivo fornecido inicialmente.