Covid-19 Image Classifier

Daniel Jaso
May 4th, 2020
HS 611
Professor Hasan

What is the topic?

The topic corresponds to how multinomial regression is applied with real life scenarios; from going through a common data set with regards to this type of regression used for classification that have more than two classes we are trying to classify. This is used with a common data set, which is a MNIST data set that corresponds to image classifications of images from zero to nine digits, in other words, having ten different classes. Moving from this commonly used data set to apply multinomial regression but with pneumonia images that have become publicly available on Kaggle where this data set contains images of patients with normal lungs (healthy), Bacterial, Viral, and Covid-19 pneumonia. In other words, four different classes while MNIST data set contains nine different labels.

Why did you choose this topic?

I chose this topic because right now the data science community has launched this campaign with multiple data sets that are related to Covid-19 that goes from different domains of knowledge in NLP to the domain of computer vision that was the motivation for this project. Computer vision in healthcare has become an area that continues to grow more in order to be able to complement what a radiologist may describe from an X-Ray scan and with the help of advanced pipelines that are within tensorflow and keras libraries to be able to interpret the presence of a disease or none.

The idea behind this project was to make use of Tensorflow and Keras libraries in order to use multinomial logistic regression, while in parallel this same step was done without the usage of Keras in order to go through the same process of classifying multiple classes.

This process starts first by using a neural network to apply a multinomial regression with gradient descent without Keras approach working for MNIST data set. Once this step was done then I made use of Keras over the same data set and compared the results, which were pretty much similar.

```
(base) ML-ITS-701744:covid-19_image_classifier student1$ python -W ignore MNIST_python_version/MNIST_keras/MNIST_keras.py
WARNING:tensorflow:From /Users/student1/opt/anaconda3/lib/python3.7/site-packages/tensorflow/python/ops/resource_variable
 will be removed in a future version.
Instructions for updating:
Colocations handled automatically by placer.
WARNING:tensorflow:From /Users/student1/opt/anaconda3/lib/python3.7/site-packages/tensorflow/python/ops/math_ops.py:3066:
n a future version.
Instructions for updating:
Use tf.cast instead.
2020-05-04 20:41:36.227064: I tensorflow/core/platform/cpu_feature_guard.cc:141] Your CPU supports instructions that this
Epoch 1/5
60000/60000 [==============================] - 2s 40us/sample - loss: 0.9552 - acc: 0.7534
Epoch 2/5
60000/60000 [==============================] - 2s 34us/sample - loss: 0.4492 - acc: 0.8785
Epoch 3/5
60000/60000 [==============================] - 2s 33us/sample - loss: 0.3769 - acc: 0.8938
Epoch 4/5
60000/60000 [==============================] - 2s 33us/sample - loss: 0.3436 - acc: 0.9024
Epoch 5/5
60000/60000 [==============================] - 2s 31us/sample - loss: 0.3232 - acc: 0.9078
10000/10000 [==============================] - 0s 37us/sample - loss: 0.3002 - acc: 0.9132
Accuracy: 91.32 %
```

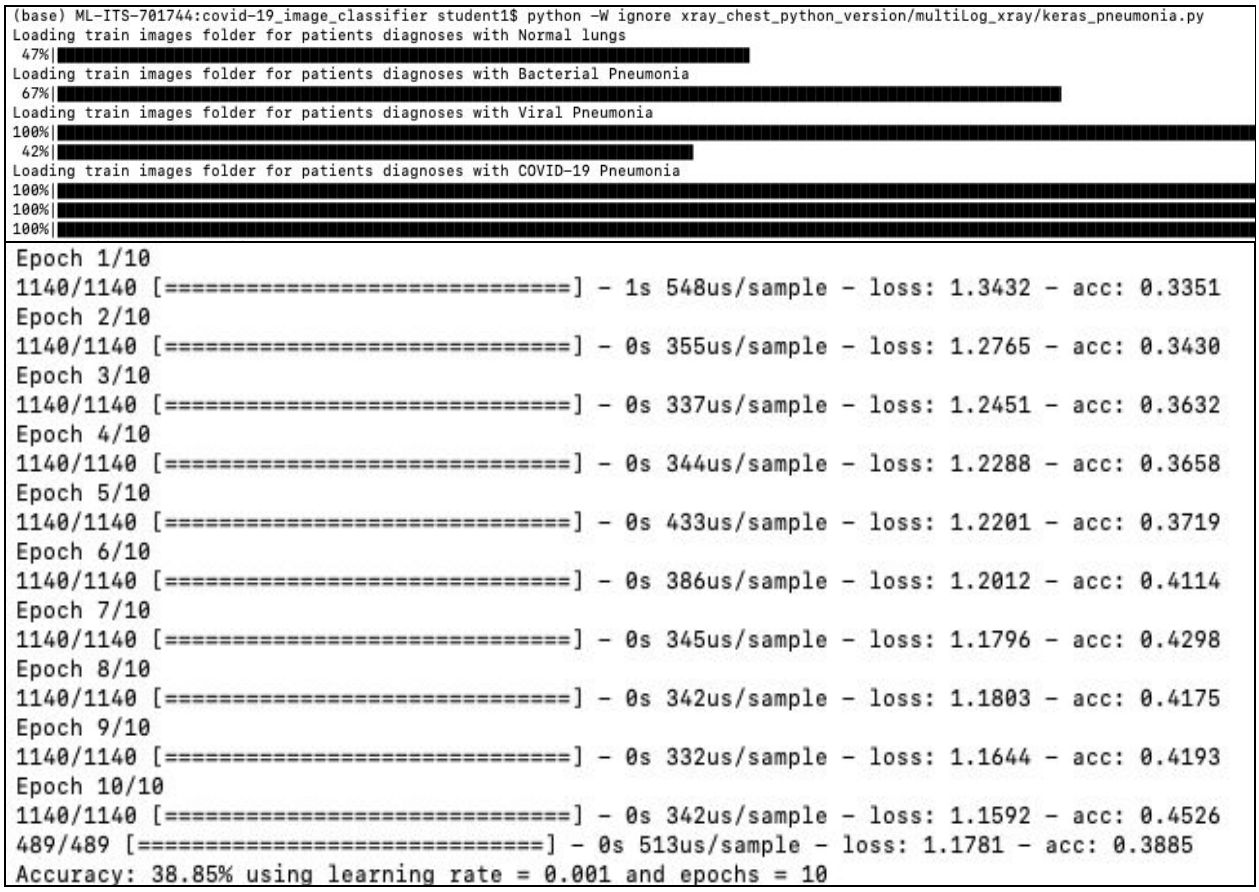
MNIST Results using Keras

```
(base) ML-ITS-701744:covid-19_image_classifier student1$ python -W ignore MNIST_python_version/MNIST_Multinomial_version/MNIST_MultiLogReg.py
Epochs: 1 Cost: 138155.10557972503
Epochs: 2 Cost: 22828.272705605294
Epochs: 3 Cost: 20335.76753462633
Epochs: 4 Cost: 19228.91972555498
Epochs: 5 Cost: 18471.437311636397
Time to compute Stochastic Gradient Descent: 2.85 minutes  for 5 epochs
Accuracy: 91.82 %
```

MNIST Results without using Keras

The following step was using a neural network with Keras approach for images of patients with normal lungs (healthy), Bacterial, Viral, and Covid-19 pneumonia. Once this step was done, then I moved forward and worked using a neural network to apply a multinomial regression with gradient descent without Keras. Problems encountered with this data set was that I was facing an imbalance dataset with 1351 healthy, bacterial pneumonia 974, Viral pneumonia

617, and Covid-19 129 patients. Therefore, in order to improve this I accounted for making use of oversampling in order to double up the smallest labels representation which was Covid-19 and lowering down the labels that had too many samples in order to balance the data. Although this is improve the accuracy, due to still low number of samples of Covid-19, the accuracy for this classifying these four classes did not went higher than 48% and I made used of different parameters (parameter tuning between the number of epochs and learning rate) in order to see what was the best one.



Pneumonia Images with Four Classes Results using Keras (without oversampling)

```
Results with oversampling

Epoch 1/5
1400/1400 [==============================] - 1s 546us/sample - loss: 1.4058 - acc: 0.2950
Epoch 2/5
1400/1400 [==============================] - 0s 338us/sample - loss: 1.3269 - acc: 0.3643
Epoch 3/5
1400/1400 [==============================] - 0s 357us/sample - loss: 1.2533 - acc: 0.3950
Epoch 4/5
1400/1400 [==============================] - 0s 335us/sample - loss: 1.1954 - acc: 0.4150
Epoch 5/5
1400/1400 [==============================] - 0s 331us/sample - loss: 1.1844 - acc: 0.4200
600/600 [==============================] - 0s 454us/sample - loss: 1.0968 - acc: 0.4550
Accuracy: 45.5% using learning rate = 0.01 and epochs = 5
Epoch 1/10
1400/1400 [==============================] - 1s 538us/sample - loss: 1.4132 - acc: 0.2543
Epoch 2/10
1400/1400 [==============================] - 0s 343us/sample - loss: 1.3863 - acc: 0.2471
Epoch 3/10
1400/1400 [==============================] - 0s 335us/sample - loss: 1.3844 - acc: 0.2636
Epoch 4/10
1400/1400 [==============================] - 1s 395us/sample - loss: 1.3863 - acc: 0.2436
Epoch 5/10
1400/1400 [==============================] - 0s 337us/sample - loss: 1.3862 - acc: 0.2507
Epoch 6/10
1400/1400 [==============================] - 0s 327us/sample - loss: 1.3863 - acc: 0.2500
Epoch 7/10
1400/1400 [==============================] - 0s 337us/sample - loss: 1.3863 - acc: 0.2314
Epoch 8/10
1400/1400 [==============================] - 0s 332us/sample - loss: 1.3865 - acc: 0.2600
Epoch 9/10
1400/1400 [==============================] - 0s 333us/sample - loss: 1.3863 - acc: 0.2571
Epoch 10/10
1400/1400 [==============================] - 0s 332us/sample - loss: 1.3857 - acc: 0.2486
600/600 [==============================] - 0s 582us/sample - loss: 1.3851 - acc: 0.2650
Accuracy: 26.5% using learning rate = 0.01 and epochs = 10
Epoch 1/5
1400/1400 [==============================] - 1s 553us/sample - loss: 1.3656 - acc: 0.3307
Epoch 2/5
1400/1400 [==============================] - 0s 335us/sample - loss: 1.2379 - acc: 0.4093
Epoch 3/5
1400/1400 [==============================] - 0s 333us/sample - loss: 1.1768 - acc: 0.4557
Epoch 4/5
1400/1400 [==============================] - 0s 351us/sample - loss: 1.1355 - acc: 0.4836
Epoch 5/5
1400/1400 [==============================] - 0s 340us/sample - loss: 1.1182 - acc: 0.4886
600/600 [==============================] - 0s 503us/sample - loss: 1.0931 - acc: 0.4783
Accuracy: 47.83% using learning rate = 0.001 and epochs = 5
```

Pneumonia images with Four Classes Results using Keras (with oversampling)

```
(base) ML-ITS-701744:covid-19_image_classifier student1$ python -W ignore xray_chest_python_version/multiLog_xray/multiLog_pneumonia.py
Loading train images folder for patients diagnoses with Normal lungs
 54%|
Loading train images folder for patients diagnoses with Bacterial Pneumonia
 76%|
Loading train images folder for patients diagnoses with Viral Pneumonia
100%|
 42%|
Loading train images folder for patients diagnoses with COVID-19 Pneumonia
100%|
100%|
100%|
Results without oversampling
Epochs: 1 Cost: 1580.3755716766336
Epochs: 2 Cost: 63596.656889814156
Epochs: 3 Cost: 28704.265510470083
Epochs: 4 Cost: 26859.496351467005
Epochs: 5 Cost: 36165.72452860325
Time to compute Stochastic Gradient Descent: 0.19 minutes  for 5 epochs
Accuracy: 38.65 % using learning rate = 0.01 and epochs = 5
Epochs: 1 Cost: 1580.3755716766336
Epochs: 2 Cost: 40261.82225111082
Epochs: 3 Cost: 34775.41977845469
Epochs: 4 Cost: 23589.612107742312
Epochs: 5 Cost: 51728.09317532561
Epochs: 6 Cost: 33971.19290964868
Epochs: 7 Cost: 43190.06185208326
Epochs: 8 Cost: 43547.03893207698
Epochs: 9 Cost: 57559.062320446974
Epochs: 10 Cost: 34804.8361840814
Time to compute Stochastic Gradient Descent: 0.36 minutes  for 10 epochs
Accuracy: 38.45 % using learning rate = 0.01 and epochs = 10
Epochs: 1 Cost: 1580.3755716766336
Epochs: 2 Cost: 5486.677985203057
Epochs: 3 Cost: 2691.050170397243
Epochs: 4 Cost: 7264.500913823852
Epochs: 5 Cost: 2927.854576540848
Time to compute Stochastic Gradient Descent: 0.2 minutes  for 5 epochs
Accuracy: 37.22 % using learning rate = 0.001 and epochs = 5
Epochs: 1 Cost: 1580.3755716766336
Epochs: 2 Cost: 2489.570824802508
Epochs: 3 Cost: 3499.9043229005742
Epochs: 4 Cost: 2055.8932565125633
Epochs: 5 Cost: 2039.819867269546
Epochs: 6 Cost: 3589.599903354804
Epochs: 7 Cost: 2802.675500075392
Epochs: 8 Cost: 4907.348346638018
Epochs: 9 Cost: 2425.666219860332
Epochs: 10 Cost: 2269.8964097753387
Time to compute Stochastic Gradient Descent: 0.37 minutes  for 10 epochs
Accuracy: 39.67 % using learning rate = 0.001 and epochs = 10
```

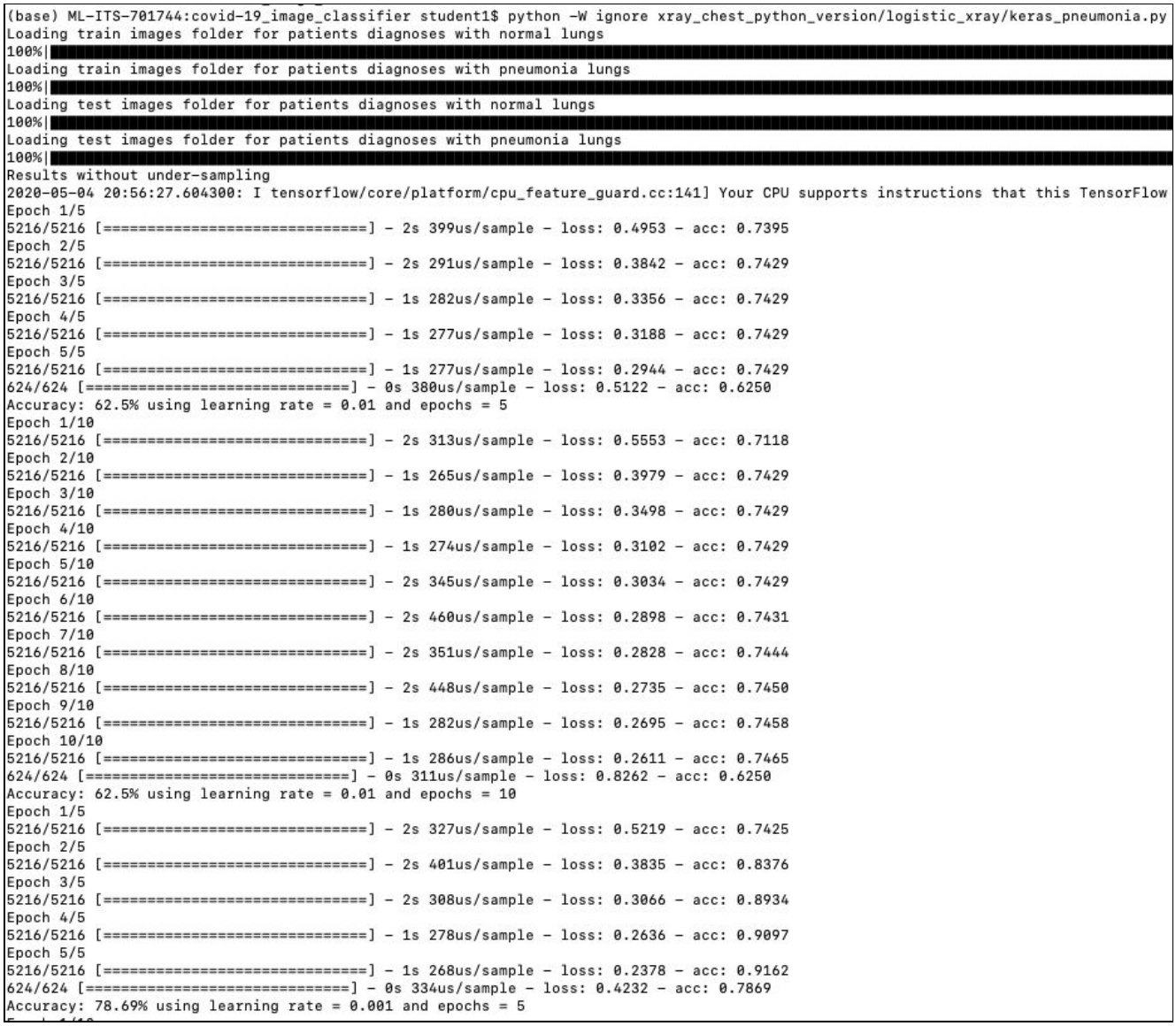
Pneumonia images with Four Classes Results without using Keras (without oversampling)

```
Results with oversampling

Epochs: 1 Cost: 1940.8121055677766
Epochs: 2 Cost: 35638.369489085846
Epochs: 3 Cost: 21723.029706240846
Epochs: 4 Cost: 34147.52034712337
Epochs: 5 Cost: 24985.964747224283
Time to compute Stochastic Gradient Descent: 0.13 minutes  for 5 epochs
Accuracy: 47.83 % using learning rate = 0.01 and epochs = 5
```

Pneumonia images with Four Classes Results without using Keras (without oversampling)

The final step was to move from a multinomial scenario to a logistic scenario in order to only classify patients that are healthy or patients who have pneumonia (without considering if it was bacterial, viral or Covid-19 pneumonia). Same approach was used just changing some of the parameters in order to change from a softmax activation function, which was used on the multinomial approach, to a sigmoid activation function that is used in the context of classifying zero or one. Moving to this context and by playing with the parameters tuning and using over here undersampling in order to lower the labels of the majority of which in this case the majority were the healthy with 1341 and pneumonia patients with 3875 that is one to three ratio. Therefore, this pneumonia label undersample to balance this dataset and it significantly improves the model up to approximately 82% .



Pneumonia Images with Two Classes Results using Keras (without undersampling)

```
Results with under-sampling

Epoch 1/5
2682/2682 [==============================] - 1s 374us/sample - loss: 0.6866 - acc: 0.5440
Epoch 2/5
2682/2682 [==============================] - 1s 265us/sample - loss: 0.6364 - acc: 0.6752
Epoch 3/5
2682/2682 [==============================] - 1s 270us/sample - loss: 0.5645 - acc: 0.8113
Epoch 4/5
2682/2682 [==============================] - 1s 269us/sample - loss: 0.5118 - acc: 0.7987
Epoch 5/5
2682/2682 [==============================] - 1s 277us/sample - loss: 0.4432 - acc: 0.8102
624/624 [==============================] - 0s 384us/sample - loss: 0.6838 - acc: 0.6667
Accuracy: 66.67% using learning rate = 0.01 and epochs = 5
Epoch 1/10
2682/2682 [==============================] - 1s 358us/sample - loss: 0.7740 - acc: 0.5093
Epoch 2/10
2682/2682 [==============================] - 1s 268us/sample - loss: 0.6932 - acc: 0.5000
Epoch 3/10
2682/2682 [==============================] - 1s 264us/sample - loss: 0.6930 - acc: 0.5000
Epoch 4/10
2682/2682 [==============================] - 1s 278us/sample - loss: 0.6928 - acc: 0.5000
Epoch 5/10
2682/2682 [==============================] - 1s 261us/sample - loss: 0.6924 - acc: 0.5000
Epoch 6/10
2682/2682 [==============================] - 1s 266us/sample - loss: 0.6904 - acc: 0.5000
Epoch 7/10
2682/2682 [==============================] - 1s 269us/sample - loss: 0.6679 - acc: 0.5537
Epoch 8/10
2682/2682 [==============================] - 1s 263us/sample - loss: 0.5638 - acc: 0.7528
Epoch 9/10
2682/2682 [==============================] - 1s 271us/sample - loss: 0.4880 - acc: 0.7774
Epoch 10/10
2682/2682 [==============================] - 1s 269us/sample - loss: 0.4410 - acc: 0.8009
624/624 [==============================] - 0s 366us/sample - loss: 0.4466 - acc: 0.7724
Accuracy: 77.24% using learning rate = 0.01 and epochs = 10
Epoch 1/5
2682/2682 [==============================] - 1s 362us/sample - loss: 0.6132 - acc: 0.6745
Epoch 2/5
2682/2682 [==============================] - 1s 264us/sample - loss: 0.4972 - acc: 0.7752
Epoch 3/5
2682/2682 [==============================] - 1s 266us/sample - loss: 0.4373 - acc: 0.8069
Epoch 4/5
2682/2682 [==============================] - 1s 260us/sample - loss: 0.3335 - acc: 0.8870
Epoch 5/5
2682/2682 [==============================] - 1s 272us/sample - loss: 0.3038 - acc: 0.8926
624/624 [==============================] - 0s 396us/sample - loss: 0.4056 - acc: 0.8237
Accuracy: 82.37% using learning rate = 0.001 and epochs = 5
```

Pneumonia Images with Two Classes Results using Keras (with undersampling)

```
(base) ML-ITS-701744:covid-19_image_classifier student1$ python -W ignore xray_chest_python_version/logistic_xray/logistic_pneumonia.py
Loading train images folder for patients diagnoses with normal lungs
100%|
Loading train images folder for patients diagnoses with pneumonia lungs
100%|
Loading test images folder for patients diagnoses with normal lungs
100%|
Loading test images folder for patients diagnoses with pneumonia lungs
100%|
Results without undersampling
Epochs: 1 Cost: -780.7137570392252
Epochs: 2 Cost: 1548.014660693251
Epochs: 3 Cost: 3118.934629588163
Epochs: 4 Cost: 6551.323632567101
Epochs: 5 Cost: 602.915556467806
Time to compute Stochastic Gradient Descent: 0.15 minutes  for 5 epochs
Accuracy: 73.56 %
Epochs: 1 Cost: 63333.56735806977
Epochs: 2 Cost: 12703.45037714774
Epochs: 3 Cost: 9319.046253243034
Epochs: 4 Cost: 1019.9551056319848
Epochs: 5 Cost: 1245.133980761927
Epochs: 6 Cost: 12606.608337873275
Epochs: 7 Cost: 7262.5898790834735
Epochs: 8 Cost: 37.0717976902304
Epochs: 9 Cost: 769.6337651712965
Epochs: 10 Cost: 30.803713955076198
Time to compute Stochastic Gradient Descent: 0.36 minutes  for 10 epochs
Accuracy: 78.69 %
Epochs: 1 Cost: -929.5103674653541
Epochs: 2 Cost: 2164.7137882517823
Epochs: 3 Cost: 2972.1102597424583
Epochs: 4 Cost: 2796.645537342753
Epochs: 5 Cost: 1164.273615876184
Time to compute Stochastic Gradient Descent: 0.14 minutes  for 5 epochs
Accuracy: 70.67 %
Epochs: 1 Cost: 264139.23965643055
Epochs: 2 Cost: 254.8095407581349
Epochs: 3 Cost: 2293.9159150140927
Epochs: 4 Cost: 2029.5375806513161
Epochs: 5 Cost: 1248.1055757750305
Epochs: 6 Cost: 1523.1649998412631
Epochs: 7 Cost: 4836.331496098818
Epochs: 8 Cost: 3590.713557090624
Epochs: 9 Cost: 1453.7701043374445
Epochs: 10 Cost: 1451.1703595530398
Time to compute Stochastic Gradient Descent: 0.29 minutes  for 10 epochs
Accuracy: 75.32 %

Results with undersampling

Epochs: 1 Cost: 6994.780528953908
Epochs: 2 Cost: 4474.634919458892
Epochs: 3 Cost: 396.0636512509447
Epochs: 4 Cost: 4014.105969656939
Epochs: 5 Cost: 97.99290374357888
Time to compute Stochastic Gradient Descent: 0.06 minutes  for 5 epochs
Accuracy: 81.57 %
```

Pneumonia Images with Two Classes Results without using Keras (without and with undersampling)

In summary, although in the beginning this was intended to go over a multinomial regression scenario, the results previously discussed did not go well and with the multinomial regression over the pneumonia images with four different labels, which is primarily because there is still not too much data publicly available. Therefore, this was the motivation to move to a simple scenario that has only two labels and perform a better classification on whether a patient has pneumonia or not that is one of the severe symptoms people experience.

References:

Cohen, J. P. (2020, March). ieee8023/covid-chestxray-dataset. Retrieved March 21, 2020, from https://github.com/ieee8023/covid-chestxray-dataset

Covid-19. (2020, March). Retrieved March 21, 2020, from https://www.newscientist.com/term/covid-19/

Kottasová, I. (2020, March 20). Data from China shows the majority of people with Covid-19 only suffer mild symptoms, then recover. Retrieved March 21, 2020, from https://www.cnn.com/2020/03/20/health/covid-19-recovery-rates-intl/index.html

Mooney, P. (2018, March 24). Chest X-Ray Images (Pneumonia). Retrieved March 19, 2020, from https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia

Naming the coronavirus disease (COVID-19) and the virus that causes it. (2020, March 19). Retrieved March 21, 2020, from https://www.cdc.gov/coronavirus/2019-ncov/faq.html

Narkhede, S. (2019, May 26). Understanding AUC - ROC Curve. Retrieved March 21, 2020, from https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

Ngbolin. (2017, July 5). MNIST Dataset: Digit Recognizer. Retrieved March 19, 2020, from https://www.kaggle.com/ngbolin/mnist-dataset-digit-recognizer

O'Connor, D. (2019). *Computing the gradient efficiently in multiclass logistic regression*. Retrieved from https://drive.google.com/file/d/1-YbJxErZWa4sXNiqNAy1M1HJ2x5Bppaa/view?usp=sharing

Weatherspoon, D. (2017, June). Chest X-Ray. Retrieved March 21, 2020, from https://www.healthline.com/health/chest-x-ray#preparation

Xu, A. Y. (2020, March 21). Detecting COVID-19 induced Pneumonia from Chest X-rays with Transfer Learning: An implementation... Retrieved March 19, 2020, from https://towardsdatascience.com/detecting-covid-19-induced-pneumonia-from-chest-x-rays-with-transfer-learning-an-implementation-311484e6afc1

Yu, G. (2020, March 20). How smoking, vaping and drug use might increase risks from Covid-19. Retrieved March 20, 2020, from https://www.cnn.com/2020/03/20/health/coronavirus-vaping-drugs/index.html?utm_medium=social&utm_term=link&utm_source=fbCNNi&utm_content=2020-03-20T17:30:33