FIRST BATCH OF TESTS

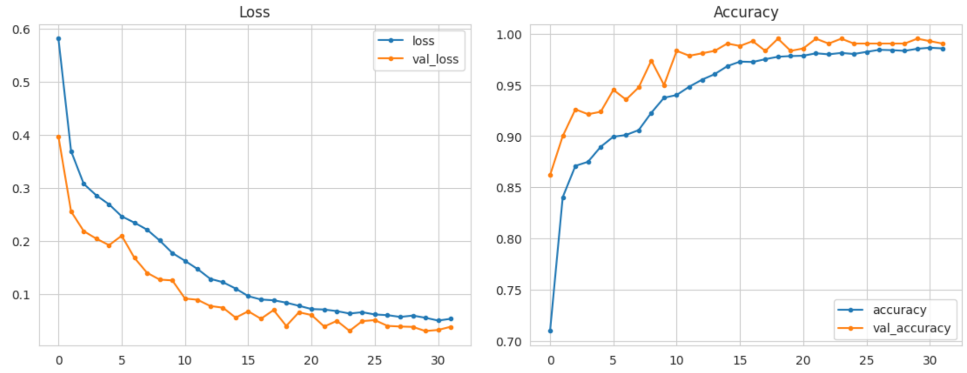
This first set of tests is purely explorative and focused mainly on setting the batch size and compare the rate of convergence of 3 optimizers.

TEST 0

INPUT

```
batch_size= 16
image_shape=(256,256,3)

 =============================================================
  model = Sequential()

  model.add(Conv2D(32, (3, 3), input_shape=image_shape, activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Conv2D(32, (3, 3), activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))
  model.add(Conv2D(64, (3, 3), activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))
  model.add(Flatten())
  model.add(Dense(64, activation='relu'))
  model.add(Dropout(.2))

  model.add(Dense(1, activation='sigmoid'))

  model.compile(loss='binary_crossentropy',
              optimizer='adagrad',
              metrics=['accuracy'])

=============================================================
no early stop
```

OUTPUT



```
=================================================================
Total params: 3,715,169
Trainable params: 3,715,169
Non-trainable params:0
=================================================================
Classification Report:
----------------------
                precision    recall  f1-score   support

      healthy       0.50      1.00      0.67       422
powdery_mildew      0.00      0.00      0.00       422

     accuracy                           0.50       844
    macro avg       0.25      0.50      0.33       844
 weighted avg       0.25      0.50      0.33       844
=================================================================
Model accuracy: 98.22%
Model Loss:  0.0639
=================================================================
Confusion Matrix:
[[422   0]
 [422 0]]
```

OBSERVATIONS

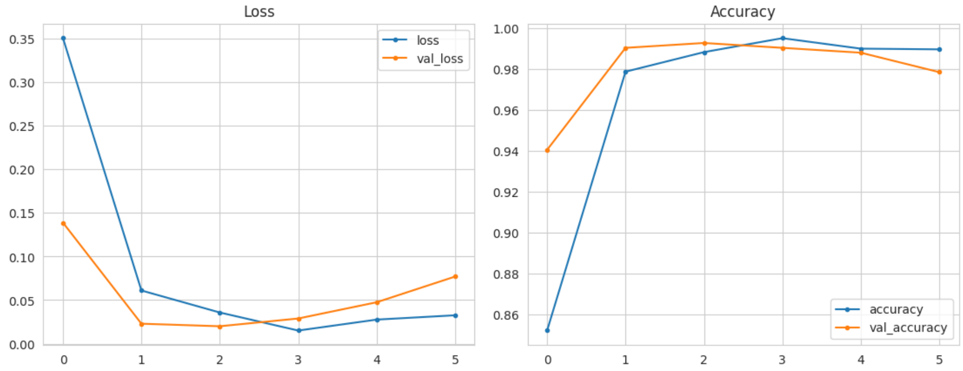
The model presents high recall but low precision. During the debugging phase it was found that `label_mode` of train and validation datasets was incoherent with the model and the activation function expecting binary imput labels.

TEST 1

INPUT

```
 batch_size= 20
 image_shape=(256,256,3)

=============================================================
  model = Sequential()

  model.add(Conv2D(filters=16, kernel_size=(3, 3),
          input_shape=image_shape, activation='relu', ))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Conv2D(filters=32, kernel_size=(3, 3),
          input_shape=image_shape, activation='relu', ))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Conv2D(filters=64, kernel_size=(3, 3),
          input_shape=image_shape, activation='relu', ))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Flatten())
  model.add(Dense(128, activation='relu'))

  model.add(Dropout(0.5))
  model.add(Dense(2, activation='softmax'))

  model.compile(loss='sparse_categorical_crossentropy',
             optimizer='adam',
             metrics=['accuracy'])
=============================================================
no early stop
```

OUTPUT



```
=================================================================
Total params: 7,396,770
Trainable params: 7,396,770
Non-trainable params:0
=================================================================
Classification Report:
----------------------
                precision    recall  f1-score   support

       healthy       0.50      1.00      0.67       422
powdery_mildew       0.00      0.00      0.00       422

      accuracy                           0.50       844
     macro avg       0.25      0.50      0.33       844
  weighted avg       0.25      0.50      0.33       844
=================================================================
Model accuracy: 98.70%
Model Loss:  0.0390
=================================================================
Confusion Matrix:
```

OBSERVATIONS

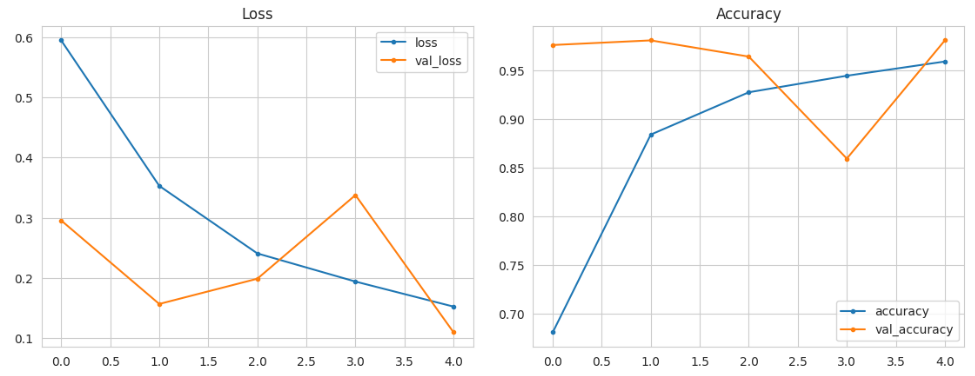
A much higher validation loss than training loss it’s a sign of **overfitting**. The model is learning patterns that accidentally happen to be true without learning the correlation between pattern and label. Infact, an infected leaf was wrongfully predicted as healthy.

TEST 2

INPUT

```
 batch_size= 64
 image_shape=(256,256,3)

============================================================
   model = Sequential()

   model.add(Conv2D(filters=16, kernel_size=(3, 3),
            input_shape=image_shape, activation='relu', ))
   model.add(MaxPooling2D(pool_size=(2, 2)))

   model.add(Conv2D(filters=32, kernel_size=(3, 3),
            input_shape=image_shape, activation='relu', ))
   model.add(MaxPooling2D(pool_size=(2, 2)))

   model.add(Conv2D(filters=64, kernel_size=(3, 3),
            input_shape=image_shape, activation='relu', ))
   model.add(MaxPooling2D(pool_size=(2, 2)))

   model.add(Flatten())
   model.add(Dense(16, activation='relu'))

   model.add(Dropout(0.5))
   model.add(Dense(1, activation='sigmoid'))

   model.compile(loss='binary_crossentropy',
              optimizer=RMSprop(learning_rate=0.0001),
              metrics=['accuracy'])
============================================================
early_stop = EarlyStopping(monitor='val_accuracy',mode='max',patience=3,verbose=1)
```

OUTPUT



```
=================================================================
Total params: 945,217
Trainable params: 945,217
Non-trainable params:0
=================================================================
Classification Report:
----------------------
               precision    recall  f1-score   support

       healthy      0.50      1.00      0.67       422
powdery_mildew      0.00      0.00      0.00       422

      accuracy                          0.50       844
     macro avg      0.25      0.50      0.33       844
  weighted avg      0.25      0.50      0.33       844
=================================================================
Model accuracy: 98.46%
Model Loss:  0.07884497195482254
=================================================================
Confusion Matrix:
```

OBSERVATIONS

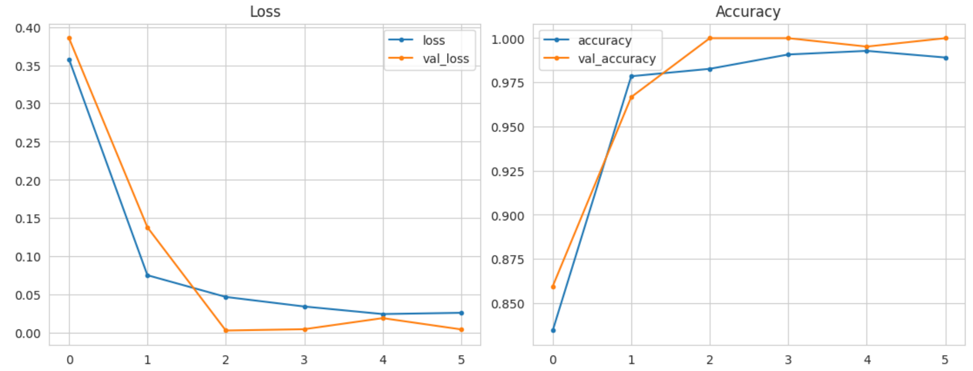
The model is **underfitting**. While it’s true that increasing batch size (compared to TEST 1) reduces validation loss, a wrong compensation of the learning rate resulted in validation loss << training loss.

TEST 3

INPUT

```
 batch_size= 20
 image_shape=(256,256,3)

============================================================
  model = Sequential()

  model.add(Conv2D(filters=32, kernel_size=(3, 3),
          input_shape=image_shape, activation='relu', ))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Conv2D(filters=64, kernel_size=(3, 3),
          input_shape=image_shape, activation='relu', ))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Conv2D(filters=64, kernel_size=(3, 3),
          input_shape=image_shape, activation='relu', ))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Flatten())
  model.add(Dense(128, activation='relu'))

  model.add(Dropout(0.5))
  model.add(Dense(1, activation='sigmoid'))

  model.compile(loss='binary_crossentropy',
             optimizer='adam',
             metrics=['accuracy'])
============================================================
early_stop = EarlyStopping(monitor='val_loss', patience=3)
```

OUTPUT



```
=================================================================
Total params: 7,429,377
Trainable params: 7,429,377
Non-trainable params:0
=================================================================
Classification Report:
----------------------
                precision    recall  f1-score   support

       healthy       0.50      1.00      0.67       422
powdery_mildew       0.00      0.00      0.00       422

      accuracy                           0.50       844
     macro avg       0.25      0.50      0.33       844
  weighted avg       0.25      0.50      0.33       844
=================================================================
Model accuracy: 99.64%
Model Loss:  0.0099
=================================================================
Confusion Matrix:
```

OBSERVATIONS

The stopped training after 6 epochs. The model is **underfitting**. Optimizer ‘adam’ converges faster than RMSProp and adagrad (see previous tests)

SECOND BATCH OF TESTS

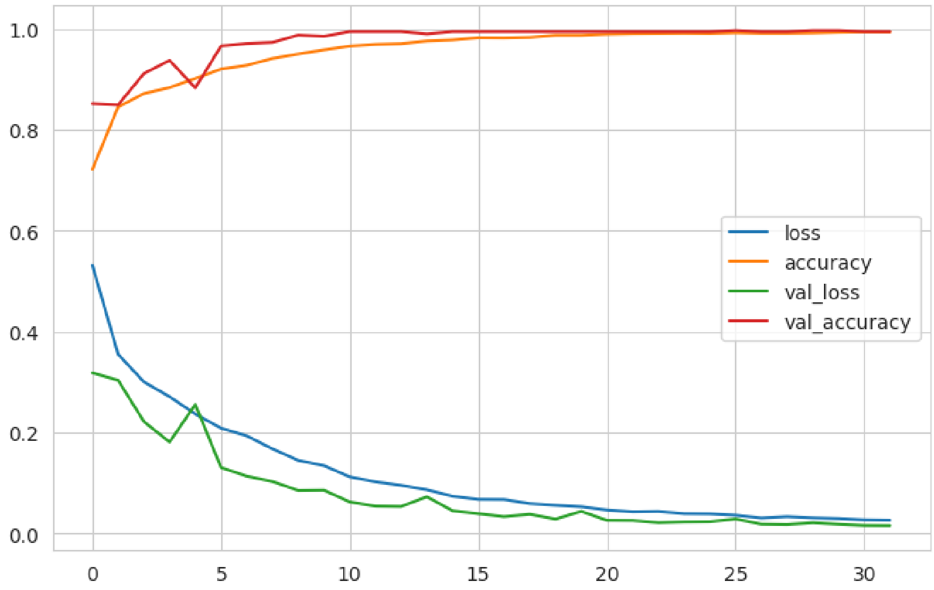
The model is built for a multiclass classification problem (See Hypothesis #2). Having set the model architecture, one parameter at a time was changed and the relative outcome evaluated. Among the tested hyperparameters: early stop parameters, optimizers and finally a hidden layer was removed.

TEST A – starting point

INPUT

```
batch_size= 16
image_shape=(256,256,3)

==============================================================
  model = Sequential()

  model.add(Conv2D(32, (3, 3), input_shape=image_shape, activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Conv2D(32, (3, 3), activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))
  model.add(Conv2D(64, (3, 3), activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))
  model.add(Flatten())
  model.add(Dense(64, activation='relu'))
  model.add(Dropout(.2))

  model.add(Dense(2, activation='softmax'))

  model.compile(loss='categorical_crossentropy',
            optimizer='adagrad',
            metrics=['accuracy'])

==============================================================
no early stop
```

OUTPUT



```
=================================================================
Total params: 3,715,234
Trainable params: 3,715,234
Non-trainable params: 0
=================================================================
Classification Report:
----------------------
                precision    recall  f1-score   support

      healthy       0.87      0.96      0.91       422
powdery_mildew       0.95      0.85      0.90       422

     accuracy                           0.91       844
    macro avg       0.91      0.91      0.90       844
 weighted avg       0.91      0.91      0.90       844
=================================================================
Model accuracy: 90.52%
Model Loss:  0.2601839005947113
=================================================================
Confusion Matrix:
[[405   17]
 [63  359]]
```

OBSERVATIONS

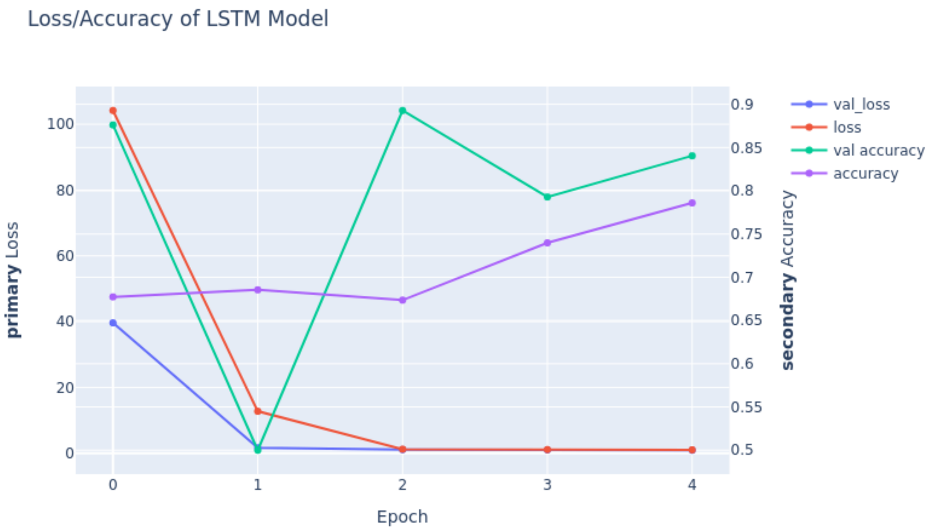
The model stopped training after 6 Epochs. The validation accuracy is lower than the training accuracy while the validation loss is increasing compared to training loss. Hence, the model is doing well on the training set but it’s not able to generalize. The model is **overfitting**.

TEST B – add early stop

INPUT

```
batch_size= 16
image_shape=(256,256,3)

============================================================
  model = Sequential()

  model.add(Conv2D(32, (3, 3), input_shape=image_shape, activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Conv2D(32, (3, 3), activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))
  model.add(Conv2D(64, (3, 3), activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))
  model.add(Flatten())
  model.add(Dense(64, activation='relu'))
  model.add(Dropout(.2))

  model.add(Dense(2, activation='softmax'))

  model.compile(loss='categorical_crossentropy',
            optimizer='adagrad',
            metrics=['accuracy'])

============================================================
early_stop = EarlyStopping(monitor='val_accuracy,mode=max,verbose=1, patience=2)
```

OUTPUT



```
=================================================================
Total params: 3,715,234
Trainable params: 3,715,234
Non-trainable params: 0

=================================================================
Classification Report:
----------------------
               precision    recall  f1-score   support
      healthy       0.82      0.92      0.87       422
powdery_mildew       0.91      0.80      0.85       422

     accuracy                           0.86       844
    macro avg       0.87      0.86      0.86       844
 weighted avg       0.87      0.86      0.86       844
=================================================================
Model accuracy: 86.26%
Model Loss:  1.1209170818328857

=================================================================
Confusion Matrix:
[[389  33]
 [83 339]]
```

OBSERVATIONS

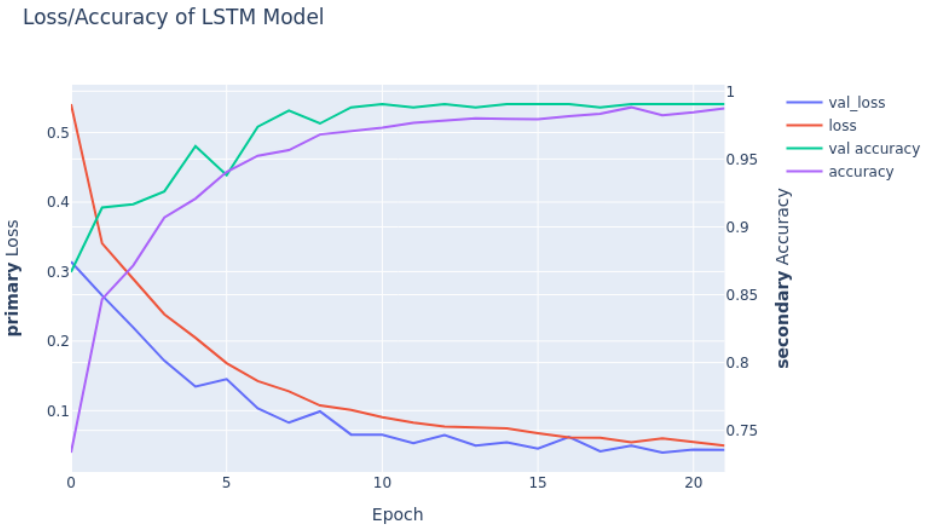
The model stopped training after 5 Epochs. The loss is high as confirmed by the confusion matrix which makes the model accuracy not satisfying despite the ‘high’ percentage. The model is likely **underfitting**.

TEST C – change early stop (monitor loss on validation set)

INPUT

```
batch_size= 16
image_shape=(256,256,3)

===========================================================
  model = Sequential()

  model.add(Conv2D(32, (3, 3), input_shape=image_shape, activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Conv2D(32, (3, 3), activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))
  model.add(Conv2D(64, (3, 3), activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))
  model.add(Flatten())
  model.add(Dense(64, activation='relu'))
  model.add(Dropout(.2))

  model.add(Dense(2, activation='softmax'))

  model.compile(loss='categorical_crossentropy',
          optimizer='adagrad',
          metrics=['accuracy'])

===========================================================
early_stop = EarlyStopping(monitor='val_loss',mode='min',verbose=1, patience=2)
```

OUTPUT



```
=================================================================
Total params: 3,715,234
Trainable params: 3,715,234
Non-trainable params: 0

=================================================================
Classification Report:
----------------------
                precision    recall  f1-score   support

       healthy       0.98      1.00      0.99       422
powdery_mildew       1.00      0.98      0.99       422

      accuracy                           0.99       844
     macro avg       0.99      0.99      0.99       844
  weighted avg       0.99      0.99      0.99       844
=================================================================
Model accuracy: 98.82%
Model Loss:  0.04888060688972473

=================================================================
Confusion Matrix:
[[422   0]
 [10 412]]
```

OBSERVATIONS

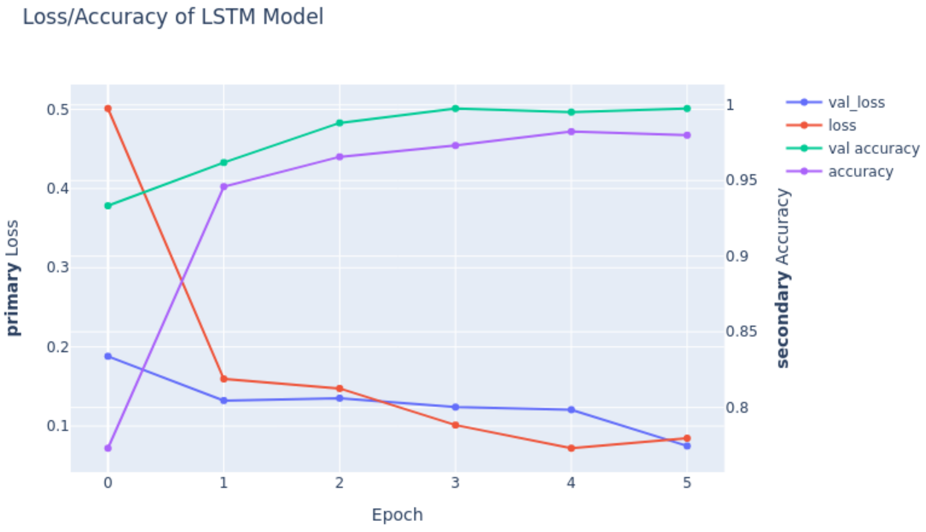
The model stopped training after 22 Epochs. Considering the number of epochs, the model accuracy and the confusion matrix (there's no wrong prediction over 422 infected leaves), the model tends to **overfit**.

TEST D – change optimizer (rmsprop)

INPUT

```
batch_size= 16
image_shape=(256,256,3)

 ==============================================================
   model = Sequential()

   model.add(Conv2D(32, (3, 3), input_shape=image_shape, activation='relu'))
   model.add(MaxPooling2D(pool_size=(2, 2)))

   model.add(Conv2D(32, (3, 3), activation='relu'))
   model.add(MaxPooling2D(pool_size=(2, 2)))
   model.add(Conv2D(64, (3, 3), activation='relu'))
   model.add(MaxPooling2D(pool_size=(2, 2)))
   model.add(Flatten())
   model.add(Dense(64, activation='relu'))
   model.add(Dropout(.2))

   model.add(Dense(2, activation='softmax'))

   model.compile(loss='categorical_crossentropy',
           optimizer='rmsprop',
           metrics=['accuracy'])

==============================================================
early_stop = EarlyStopping(monitor='val_accuracy,mode=max,verbose=1, patience=2)
```

OUTPUT



```
=================================================================
Total params: 3,715,234
Trainable params: 3,715,234
Non-trainable params: 0

=================================================================
Classification Report:
----------------------
                precision    recall  f1-score   support

       healthy       0.99      1.00      1.00       422
powdery_mildew       1.00      0.99      1.00       422

      accuracy                           1.00       844
     macro avg       1.00      1.00      1.00       844
  weighted avg       1.00      1.00      1.00       844
=================================================================
Model accuracy: 99.64%
Model Loss:  0.013791169971227646
=================================================================
Confusion Matrix:
[[422   0]
 [3 419]]
```

OBSERVATIONS

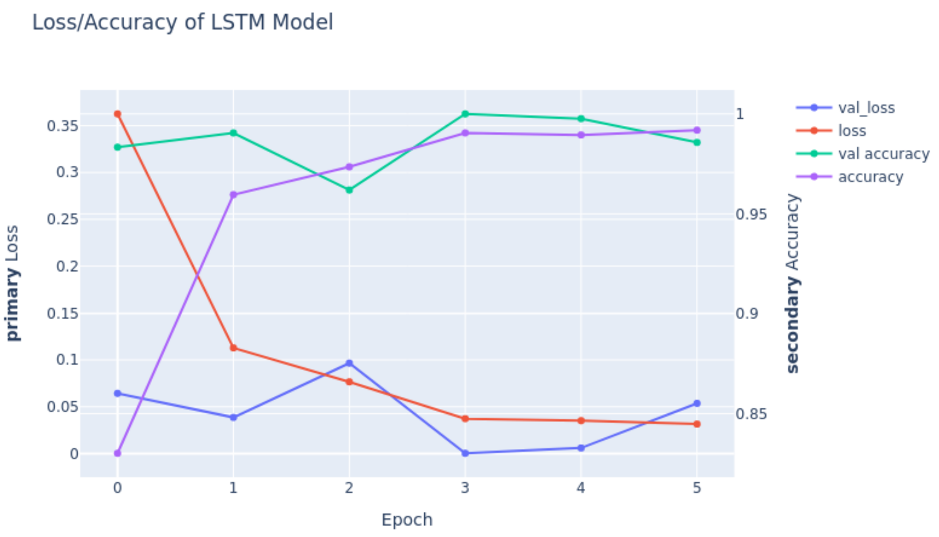
The model stopped training after 6 Epochs. Considering the unrealistic accuracy after just 6 Epochs, the loss on the train set even higher than the loss on the validation set, the accuracy which is not converging and that no healthy leaves were mistakenly predicted as infected, the model is **overfitting**.

TEST E – change optimizer (adam)

INPUT

```
 batch_size= 16
 image_shape=(256,256,3)

 ============================================================
   model = Sequential()

   model.add(Conv2D(32, (3, 3), input_shape=image_shape, activation='relu'))
   model.add(MaxPooling2D(pool_size=(2, 2)))

   model.add(Conv2D(32, (3, 3), activation='relu'))
   model.add(MaxPooling2D(pool_size=(2, 2)))
   model.add(Conv2D(64, (3, 3), activation='relu'))
   model.add(MaxPooling2D(pool_size=(2, 2)))
   model.add(Flatten())
   model.add(Dense(64, activation='relu'))
   model.add(Dropout(.2))

   model.add(Dense(2, activation='softmax'))

   model.compile(loss='categorical_crossentropy',
          optimizer='adam',
          metrics=['accuracy'])

============================================================
early_stop = EarlyStopping(monitor='val_accuracy',mode='max',verbose=1, patience=2)
```

OUTPUT



```
=================================================================
Total params: 3,715,234
Trainable params: 3,715,234
Non-trainable params: 0
=================================================================
Classification Report:
----------------------
                precision    recall  f1-score   support

       healthy       0.99      1.00      1.00       422
powdery_mildew       1.00      0.99      1.00       422

      accuracy                           1.00       844
     macro avg       1.00      1.00      1.00       844
  weighted avg       1.00      1.00      1.00       844
=================================================================
Model accuracy: 99.64%
Model Loss:  0.008717085234820843
=================================================================
Confusion Matrix:
[[422   0]
 [3 419]]
```

OBSERVATIONS

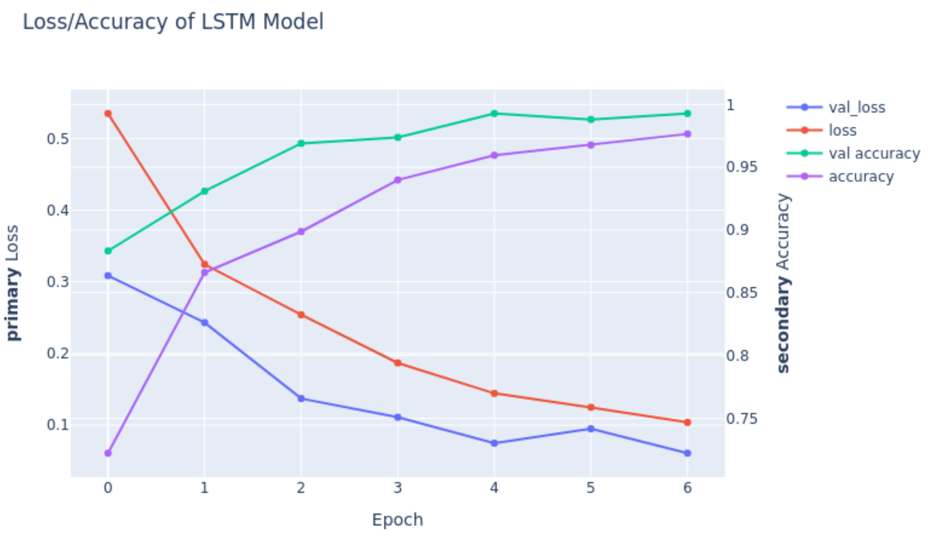
The model stopped training after 6 Epochs. The validation accuracy is lower than the training accuracy while the validation loss is increasing compared to training loss. Hence, the model is doing well on the training set but it’s not able to generalize. The model is **overfitting**.

TEST F – remove one Convolutional Layer

INPUT

```
 batch_size= 16
 image_shape=(256,256,3)

=============================================================
  model = Sequential()

  model.add(Conv2D(32, (3, 3), input_shape=image_shape, activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))

  model.add(Conv2D(32, (3, 3), activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))
  model.add(Conv2D(64, (3, 3), activation='relu'))
  model.add(MaxPooling2D(pool_size=(2, 2)))
  model.add(Flatten())
  model.add(Dense(64, activation='relu'))
  model.add(Dropout(.2))

  model.add(Dense(2, activation='softmax'))

  model.compile(loss='categorical_crossentropy',
            optimizer='adagrad',
            metrics=['accuracy'])

=============================================================
early_stop = EarlyStopping(monitor='val_accuracy',mode='max',verbose=1, patience=2)
```

OUTPUT



```
=================================================================
Total params: 15,764,610
Trainable params: 15,764,610
Non-trainable params: 0
=================================================================
Classification Report:
----------------------
                precision    recall  f1-score   support

       healthy       0.97      1.00      0.98       422
powdery_mildew       1.00      0.97      0.98       422

      accuracy                           0.98       844
     macro avg       0.98      0.98      0.98       844
  weighted avg       0.98      0.98      0.98       844
=================================================================
Model accuracy: 98.10%
Model Loss:  0.10587086528539658
=================================================================
Confusion Matrix:
[[420   2]
 [14 408]]
```

OBSERVATIONS

The model stopped training after 7 Epochs. The number of parameters increased reducing the model training speed, the accuracy seems unrealistic after just 7 Epochs (see recall and precision). The model tends to **overfitting**.