

קובץ ניסויים – מטלה 2

אציג את הפרמטרים כתמונה שלהם , ואתעד את התוצאה הכי טובה
עבור הפרמטרים האלה. תעודת הזחות שלי היא 318190816 ולכן
השכבות יהיו לפי ההנחיות:
16,08,19,18 - ואכתוב 08 כ - 8.

ניסוי 1 :

הפרמטרים :

```
layers = [
    tf.keras.layers.Flatten(input_shape=image_shape),

    tf.keras.layers.Dense(16),
    tf.keras.layers.Activation('relu'),

    tf.keras.layers.Dense(8),
    tf.keras.layers.Activation('relu'),

    tf.keras.layers.Dense(19),
    tf.keras.layers.Activation('relu'),

    tf.keras.layers.Dense(18),
    tf.keras.layers.Activation('relu'),

    tf.keras.layers.Dense(num_of_classes),
    tf.keras.layers.Softmax()
]
```

התוצאה הטובה ביותר:

The best Epoch was : 21/50

loss: 0.1548 - sparse_categorical_accuracy: 0.9591 - val_loss: 0.2783 - val_sparse_categorical_accuracy: 0.9470

ניסוי 2 :

הפרמטרים :

```
layers = [
    tf.keras.layers.Flatten(input_shape=image_shape),

    tf.keras.layers.Dense(16),
    tf.keras.layers.Activation('sigmoid'),

    tf.keras.layers.Dense(8),
    tf.keras.layers.Activation('sigmoid'),

    tf.keras.layers.Dense(19),
    tf.keras.layers.Activation('sigmoid'),

    tf.keras.layers.Dense(18),
    tf.keras.layers.Activation('sigmoid'),

    tf.keras.layers.Dense(num_of_classes),
    tf.keras.layers.Softmax()
]
```

התוצאה הטובה ביותר:

The best Epoch was : 49/50

loss: 0.1769 - sparse_categorical_accuracy: 0.9539 - val_loss: 0.2518 - val_sparse_categorical_accuracy: 0.9365

ניסוי 3 :
הפרמטרים :

```
layers = [  
    tf.keras.layers.Flatten(input_shape=image_shape),  
  
    tf.keras.layers.Dense(16, kernel_regularizer=tf.keras.regularizers.l2(0.00001)),  
    tf.keras.layers.Activation('sigmoid'),  
  
    tf.keras.layers.Dense(8, kernel_regularizer=tf.keras.regularizers.l2(0.0001)),  
    tf.keras.layers.Activation('sigmoid'),  
  
    tf.keras.layers.Dense(19, kernel_regularizer=tf.keras.regularizers.l2(0.001)),  
    tf.keras.layers.Activation('sigmoid'),  
  
    tf.keras.layers.Dense(18, kernel_regularizer=tf.keras.regularizers.l2(0.01)),  
    tf.keras.layers.Activation('sigmoid'),  
  
    tf.keras.layers.Dense(num_of_classes),  
    tf.keras.layers.Softmax()  
]
```

התוצאה הטובה ביותר:

The best Epoch was : 50/50

loss: 1.0874 - sparse_categorical_accuracy: 0.7059 - val_loss: 1.1649 - val_sparse_categorical_accuracy: 0.7225

הדגשה : כמובן שבכל ההרצות מכאן יש את השורה הראשונה :

```
tf.keras.layers.Flatten(input_shape=image_shape),
```

ואז את 4 השכבות כנדרש, ואז את שתי השורות :

```
tf.keras.layers.Dense(num_of_classes),
```

```
tf.keras.layers.Softmax()
```

לא אצלם אותם כדי לחסוך מקום ושיראה יותר מסודר, אבל בכל הרצה השורות קיימות כמובן.

ניסוי 4 :

הפרמטרים :

```
tf.keras.layers.Dense(16, kernel_regularizer=tf.keras.regularizers.l2(0.0001)),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(8, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(19, kernel_regularizer=tf.keras.regularizers.l2(0.01)),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(18, kernel_regularizer=tf.keras.regularizers.l2(0.1)),
tf.keras.layers.Activation('relu'),
```

התוצאה הטובה ביותר:

The best Epoch was : 34/50

loss: 0.2408 - sparse_categorical_accuracy: 0.9562 - val_loss: 0.2551 - val_sparse_categorical_accuracy: 0.9530

ניסוי 5 :

הפרמטרים :

```
tf.keras.layers.Dense(16),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(8),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(19),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(18),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
```

התוצאה הטובה ביותר:

The best Epoch was : 50/50

loss: 0.1454 - sparse_categorical_accuracy: 0.9558 - val_loss: 0.1309 - val_sparse_categorical_accuracy: 0.9659

```
tf.keras.layers.Dense(16),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('sigmoid'),
tf.keras.layers.Dropout(0.1),

tf.keras.layers.Dense(8),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('sigmoid'),
tf.keras.layers.Dropout(0.2),

tf.keras.layers.Dense(19),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('sigmoid'),
tf.keras.layers.Dropout(0.3),

tf.keras.layers.Dense(18),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('sigmoid'),
tf.keras.layers.Dropout(0.4),
```

ניסוי 6 :
הפרמטרים :

כאן נראה דוגמא
שבה התוצאות לא
מאוד טובות, אבל אין
over fitting.

התוצאה הטובה ביותר:

The best Epoch was : 50/50

loss: 1.0154 - sparse_categorical_accuracy: 0.6604 - val_loss: 0.4788 - val_sparse_categorical_accuracy: 0.8542

```
tf.keras.layers.Dense(16),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.0001),

tf.keras.layers.Dense(8),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.002),

tf.keras.layers.Dense(19),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.03),

tf.keras.layers.Dense(18),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.4),
```

ניסוי 7 :
הפרמטרים :

התוצאה הטובה ביותר:

The best Epoch was : 44/50

loss: 0.3895 - sparse_categorical_accuracy: 0.8937 - val_loss: 0.1693 - val_sparse_categorical_accuracy: 0.9574

```
tf.keras.layers.Dense(16),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.001),

tf.keras.layers.Dense(8),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.002),

tf.keras.layers.Dense(19),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.03),

tf.keras.layers.Dense(18),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.04),
```

ניסוי 8 :
הפרמטרים :

התוצאה הטובה ביותר:

The best Epoch was : 14/50
loss: 0.1828 - sparse_categorical_accuracy: 0.9498 - val_loss: 0.1787 - val_sparse_categorical_accuracy: 0.9539

ניסוי 9 :
הפרמטרים :

```
tf.keras.layers.Dense(16,kernel_regularizer=tf.keras.regularizers.l2(0.001)),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(8,kernel_regularizer=tf.keras.regularizers.l2(0.001)),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(19,kernel_regularizer=tf.keras.regularizers.l2(0.01)),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(18,kernel_regularizer=tf.keras.regularizers.l2(0.1)),
tf.keras.layers.Activation('relu'),
```

התוצאה הטובה ביותר:

The best Epoch was : 47/50
loss: 0.2928 - sparse_categorical_accuracy: 0.9476 - val_loss: 0.2974 - val_sparse_categorical_accuracy: 0.9474

ניסוי 10 :

הפרמטרים :

```
tf.keras.layers.Dense(16,kernel_regularizer=tf.keras.regularizers.l2(0.001)),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(8,kernel_regularizer=tf.keras.regularizers.l2(0.001)),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(19,kernel_regularizer=tf.keras.regularizers.l2(0.01)),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),

tf.keras.layers.Dense(18,kernel_regularizer=tf.keras.regularizers.l2(0.1)),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
```

התוצאה הטובה ביותר:

The best Epoch was : 21/50

loss: 0.2684 - sparse_categorical_accuracy: 0.9346 - val_loss: 0.1963 - val_sparse_categorical_accuracy: 0.9548

ניסוי 11 :

הפרמטרים :

```
tf.keras.layers.Dense(16),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('sigmoid'),

tf.keras.layers.Dense(8),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('sigmoid'),

tf.keras.layers.Dense(19),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('sigmoid'),

tf.keras.layers.Dense(18),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('sigmoid'),
```

התוצאה הטובה ביותר:

The best Epoch was : 36/50

loss: 0.1541 - sparse_categorical_accuracy: 0.9530 - val_loss: 0.1348 - val_sparse_categorical_accuracy: 0.9623

```
tf.keras.layers.Dense(16),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.001),

tf.keras.layers.Dense(8),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.02),

tf.keras.layers.Dense(19),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.3),

tf.keras.layers.Dense(18),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.4),
```

ניסוי 12 :

הפרמטרים :

התוצאה הטובה ביותר:

The best Epoch was : 50/50

loss: 0.6013 - sparse_categorical_accuracy: 0.8320 - val_loss: 0.2215 - val_sparse_categorical_accuracy: 0.9453

```
tf.keras.layers.Dense(16),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.0001),

tf.keras.layers.Dense(8),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('sigmoid'),
tf.keras.layers.Dropout(0.0002),

tf.keras.layers.Dense(19),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.0003),

tf.keras.layers.Dense(18),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('sigmoid'),
tf.keras.layers.Dropout(0.0004),
```

ניסוי 13 :

הפרמטרים :

התוצאה הטובה ביותר:

The best Epoch was : 41/50

loss: 0.1507 - sparse_categorical_accuracy: 0.9542 - val_loss: 0.1415 - val_sparse_categorical_accuracy: 0.9614

ניסוי 14 :

הפרמטרים :

```
tf.keras.layers.Dense(16, kernel_regularizer=tf.keras.regularizers.l2(0.0001)),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.0001),

tf.keras.layers.Dense(8, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.0002),

tf.keras.layers.Dense(19, kernel_regularizer=tf.keras.regularizers.l2(0.01)),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.0003),

tf.keras.layers.Dense(18, kernel_regularizer=tf.keras.regularizers.l2(0.1)),
tf.keras.layers.BatchNormalization(),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.0004),
```

התוצאה הטובה ביותר:

The best Epoch was : 22/50

loss: 0.2300 - sparse_categorical_accuracy: 0.9408 - val_loss: 0.1825 - val_sparse_categorical_accuracy: 0.9598

ניסוי 15 :

הפרמטרים :

```
tf.keras.layers.Dense(16, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.1),

tf.keras.layers.Dense(8, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.1),

tf.keras.layers.Dense(19, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.2),

tf.keras.layers.Dense(18, kernel_regularizer=tf.keras.regularizers.l2(0.1)),
tf.keras.layers.Activation('relu'),
tf.keras.layers.Dropout(0.2),
```

התוצאה הטובה ביותר:

The best Epoch was : 41/50

loss: 0.6390 - sparse_categorical_accuracy: 0.8481 - val_loss: 0.3786 - val_sparse_categorical_accuracy: 0.9302

התוצאה הטובה ביותר מכל הניסויים הייתה בניסוי 5 ולכן אשאיר את הפרמטרים שלה בקובץ המטלה.