{

"nbformat": 4,

"nbformat\_minor": 0,

"metadata": {

"colab": {

"name": "DEVIKA BIJU.ipynb",

"provenance": []

},

"kernelspec": {

"name": "python3",

"display\_name": "Python 3"

},

"language\_info": {

"name": "python"

}

},

"cells": [

{

"cell\_type": "code",

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "BAt\_C5YqjXFa",

"outputId": "e7c1d801-95eb-47c2-ac8f-cfe20618edc7"

},

"source": [

"from keras.datasets import mnist\n",

"\n",

"data = mnist.load\_data()"

],

"execution\_count": 1,

"outputs": [

{

"output\_type": "stream",

"text": [

"Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz\n",

"11493376/11490434 [==============================] - 0s 0us/step\n"

],

"name": "stdout"

}

]

},

{

"cell\_type": "code",

"metadata": {

"id": "UjTwy46ajqvp"

},

"source": [

"((X\_train, y\_train), (X\_test, y\_test)) = data"

],

"execution\_count": 4,

"outputs": []

},

{

"cell\_type": "code",

"metadata": {

"id": "RT\_b6UOdj8\_Z"

},

"source": [

"X\_train = X\_train.reshape((X\_train.shape[0], 28\*28)).astype('float32')\n",

"X\_test = X\_test.reshape((X\_test.shape[0], 28\*28)).astype('float32')"

],

"execution\_count": 6,

"outputs": []

},

{

"cell\_type": "code",

"metadata": {

"id": "aMzEgM8SlgZ-"

},

"source": [

"X\_train = X\_train / 255\n",

"X\_test = X\_test / 255"

],

"execution\_count": 8,

"outputs": []

},

{

"cell\_type": "code",

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "qS\_cee3vlxEf",

"outputId": "88ed18bc-f225-470a-d38b-45f2f71ab98b"

},

"source": [

"from keras.utils import np\_utils\n",

"\n",

"print(y\_test.shape)"

],

"execution\_count": 11,

"outputs": [

{

"output\_type": "stream",

"text": [

"(10000,)\n"

],

"name": "stdout"

}

]

},

{

"cell\_type": "code",

"metadata": {

"id": "BATRA9tMmLC2"

},

"source": [

"X\_train = X\_train / 255\n",

"X\_test = X\_test / 255"

],

"execution\_count": 12,

"outputs": []

},

{

"cell\_type": "code",

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "VGnrFeVImmLf",

"outputId": "811106f2-a5dc-438a-978a-f0aa17f31bd7"

},

"source": [

"from keras.utils import np\_utils\n",

"\n",

"print(y\_test.shape)\n",

"\n",

"y\_train = np\_utils.to\_categorical(y\_train)\n",

"y\_test = np\_utils.to\_categorical(y\_test)\n",

"\n",

"num\_classes = y\_test.shape[1]\n",

"print(y\_test.shape)"

],

"execution\_count": 14,

"outputs": [

{

"output\_type": "stream",

"text": [

"(10000,)\n",

"(10000, 10)\n"

],

"name": "stdout"

}

]

},

{

"cell\_type": "code",

"metadata": {

"id": "lrSeiPsKnqjP"

},

"source": [

"from keras.models import Sequential\n",

"\n",

"from keras.layers import Dense"

],

"execution\_count": 16,

"outputs": []

},

{

"cell\_type": "code",

"metadata": {

"id": "EIoFaoFZpJAo"

},

"source": [

"from keras.models import Sequential\n",

"from keras.layers import Dense"

],

"execution\_count": 17,

"outputs": []

},

{

"cell\_type": "code",

"metadata": {

"id": "yCcyUB9upz\_Q"

},

"source": [

"model = Sequential()\n",

"model.add(Dense(32, input\_dim = 28\*28, activation='relu'))\n",

"model.add(Dense(64, activation='relu'))\n",

"model.add(Dense(10, activation='softmax'))"

],

"execution\_count": 19,

"outputs": []

},

{

"cell\_type": "code",

"metadata": {

"id": "fRTQrn7lrB7I"

},

"source": [

"model.compile(loss='categorical\_crossentropy', optimizer='adam', metrics=['accuracy'])"

],

"execution\_count": 20,

"outputs": []

},

{

"cell\_type": "code",

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "BjibVWadrfl6",

"outputId": "abbd5570-e52c-4127-dd04-3836402e7f18"

},

"source": [

"model.summary()"

],

"execution\_count": 21,

"outputs": [

{

"output\_type": "stream",

"text": [

"Model: \"sequential\_1\"\n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n",

"Layer (type) Output Shape Param # \n",

"=================================================================\n",

"dense\_3 (Dense) (None, 32) 25120 \n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n",

"dense\_4 (Dense) (None, 64) 2112 \n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n",

"dense\_5 (Dense) (None, 10) 650 \n",

"=================================================================\n",

"Total params: 27,882\n",

"Trainable params: 27,882\n",

"Non-trainable params: 0\n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n"

],

"name": "stdout"

}

]

},

{

"cell\_type": "code",

"metadata": {

"id": "ELmfOd7AtQhs"

},

"source": [

"model.add(Dense(10, activation='softmax'))"

],

"execution\_count": 24,

"outputs": []

},

{

"cell\_type": "code",

"metadata": {

"id": "BHo0Rm-2vUDp"

},

"source": [

"model.compile(loss='categorical\_crossentropy', optimizer='adam', metrics=['accuracy'])"

],

"execution\_count": 26,

"outputs": []

},

{

"cell\_type": "code",

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "xO7ZG3Lmv0tY",

"outputId": "23e1feed-a7fa-4d00-dccb-362d75bcb468"

},

"source": [

"model.summary()"

],

"execution\_count": 27,

"outputs": [

{

"output\_type": "stream",

"text": [

"Model: \"sequential\_1\"\n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n",

"Layer (type) Output Shape Param # \n",

"=================================================================\n",

"dense\_3 (Dense) (None, 32) 25120 \n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n",

"dense\_4 (Dense) (None, 64) 2112 \n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n",

"dense\_5 (Dense) (None, 10) 650 \n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n",

"dense\_8 (Dense) (None, 10) 110 \n",

"=================================================================\n",

"Total params: 27,992\n",

"Trainable params: 27,992\n",

"Non-trainable params: 0\n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n"

],

"name": "stdout"

}

]

},

{

"cell\_type": "code",

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "GduJMfe1wTEP",

"outputId": "3b3193e2-655e-4bac-b3a5-8df989b1dc65"

},

"source": [

"model.fit(X\_train, y\_train, epochs=10, batch\_size=100)"

],

"execution\_count": 28,

"outputs": [

{

"output\_type": "stream",

"text": [

"Epoch 1/10\n",

"600/600 [==============================] - 2s 2ms/step - loss: 2.3033 - accuracy: 0.1026\n",

"Epoch 2/10\n",

"600/600 [==============================] - 1s 2ms/step - loss: 2.3015 - accuracy: 0.1127\n",

"Epoch 3/10\n",

"600/600 [==============================] - 1s 2ms/step - loss: 2.3015 - accuracy: 0.1132\n",

"Epoch 4/10\n",

"600/600 [==============================] - 1s 2ms/step - loss: 2.3015 - accuracy: 0.1135\n",

"Epoch 5/10\n",

"600/600 [==============================] - 2s 3ms/step - loss: 2.3009 - accuracy: 0.1121\n",

"Epoch 6/10\n",

"600/600 [==============================] - 1s 2ms/step - loss: 2.3011 - accuracy: 0.1141\n",

"Epoch 7/10\n",

"600/600 [==============================] - 1s 2ms/step - loss: 2.3015 - accuracy: 0.1127\n",

"Epoch 8/10\n",

"600/600 [==============================] - 1s 2ms/step - loss: 2.3014 - accuracy: 0.1115\n",

"Epoch 9/10\n",

"600/600 [==============================] - 1s 2ms/step - loss: 2.3017 - accuracy: 0.1111\n",

"Epoch 10/10\n",

"600/600 [==============================] - 1s 2ms/step - loss: 2.3010 - accuracy: 0.1121\n"

],

"name": "stdout"

},

{

"output\_type": "execute\_result",

"data": {

"text/plain": [

"<tensorflow.python.keras.callbacks.History at 0x7f26de34c910>"

]

},

"metadata": {

"tags": []

},

"execution\_count": 28

}

]

},

{

"cell\_type": "code",

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "KX1QdJDYwylt",

"outputId": "b51afeb7-4e51-48ff-a828-46b51766f74d"

},

"source": [

"scores = model.evaluate(X\_test, y\_test)\n",

"print(scores)"

],

"execution\_count": 29,

"outputs": [

{

"output\_type": "stream",

"text": [

"313/313 [==============================] - 0s 1ms/step - loss: 2.3011 - accuracy: 0.1135\n",

"[2.3010971546173096, 0.11349999904632568]\n"

],

"name": "stdout"

}

]

},

{

"cell\_type": "code",

"metadata": {

"id": "OIw9uDTBxWMZ"

},

"source": [

""

],

"execution\_count": null,

"outputs": []

}

]

}