DEEP LEARNING-BASED CARDIOVASCULAR RISK FACTOR RECOGNITION

Aim:
To study the incidence and prevalence of cardiovascular disease (CVD) and its risk factors using deep learning-based classifier.

Objectives:

- To download the CVD dataset
- To visualize the features of the CVD dataset
- To pre-process the obtained features
- To predict the presence or absence of cardiovascular disease using deep learning
- To analyze the effect of deep learning parameters on the prediction performance

Apparatus:

1. Laptop/Desktop
2. Python

Theory:
CVD is the primary cause of illness and deaths globally that contributes to enormous healthcare costs. The prevalence of CVD-related deaths increased from 12.1 million in 1990 to 18.6 million in 2019 and is estimated to reach 24 million by 2030. CVD refers to any disorder that can affect the heart and blood vessels, including coronary heart disease, cerebrovascular disease, peripheral arterial disease, rheumatic heart disease, congenital heart disease and deep vein thrombosis.

The most important behavioral risk factors of CVDs are unhealthy diet, physical inactivity, tobacco use and harmful use of alcohol. The effects of behavioral risk factors may show up in individuals as raised blood pressure, raised blood glucose, raised blood lipids, and overweight and obesity. These "intermediate risks factors" can be measured in primary care facilities and indicate an increased risk of heart attack, stroke, heart failure and other complications.

Prediction of CVD using traditional machine learning methods might not provide a better performance. In this experiment, the presence or absence of the CVD in individual is predicted using 11 handcrafted features and deep learning methods.

At a very basic level, deep learning is a machine learning technique. It teaches a computer to filter inputs through layers to learn how to predict and classify information. Observations can be in the form of images, text, or sound. The inspiration for deep learning is the way that the human brain filters information.

The Kaggle Cardiovascular Disease dataset consists of 70,000 records of patients' data in 11 features, such as age, gender, systolic blood pressure, diastolic blood pressure etc. The target class "cardio" equals to 1, when patient has cardiovascular disease, and it's 0, if patient is healthy. The task is to predict the presence or absence of CVD using the patient examination results and deep learning.

There are 3 types of input features:

1. Objective: factual information;
2. Examination: results of medical examination;
3. Subjective: information given by the patient.

Table 1. Kaggle cardiovascular disease dataset attributes description with some statistical calculation (Total Instances: 70,000)

Sl No	Variable Description	Type
1	Age (days) Min: 10798, Max: 23713, Mean: 19468.866, StdDev: 2467.252	Objective
2	Height (cm) Min: 55, Max: 250, Mean: 164.359, StdDev: 8.21	Objective
3	Weight (kg) Min: 10, Max: 200, Mean: 74.206, StdDev: 14.396	Objective
4	Gender-categorical code (f=female, m=male)	Objective
5	Systolic blood pressure Min: -150, Max: 16020, Mean: 128.817, StdDev: 154.011	Examination
6	Diastolic blood pressure Min: -70, Max: 11000, Mean: 96.63, StdDev: 188.473	Examination
7	Cholesterol- 1: normal, 2: above normal, 3: well above normal	Examination
8	Glucose 1: normal, 2: above normal, 3: well above normal	Examination
9	Smoke-binary (1=smoker, 0=non-smoker)	Subjective
10	Alcohol-binary (1=yes, 0=no)	Subjective
11	Active-binary (active=1, inactive=0)	Subjective
	Target- binary (1=Presence = 1, 0=absence of cardiovascular disease)	

All of the dataset values were collected at the moment of medical examination.

Methodology:

The typical deep learning process consists of understanding a problem. For our case its prediction of presence of CVD. The next step is the identification of the data. Our dataset is Kaggle Cardiovascular Disease dataset. The next method is selection of deep learning algorithm followed by training and testing the model. In the training process the loss is minimized while accuracy is increased. Data is divided into testing and training in the 80-20%.

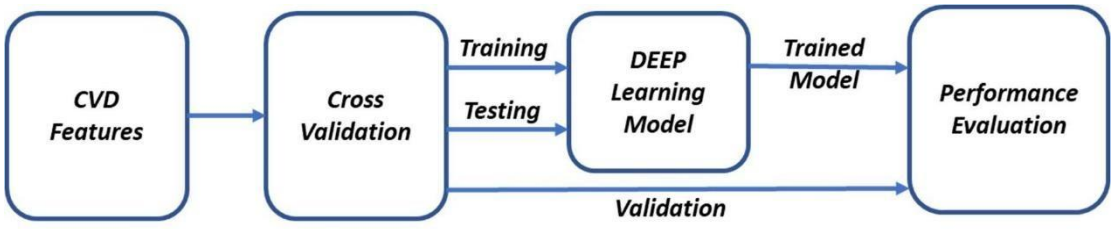

Figure 1. A typical deep learning process for classification

The methodology of the experiment is as follows:

1. Load the Kaggle Cardiovascular Disease dataset from the link: https://www.kaggle.com/datasets/sulianova/cardiovasculardiseasedataset/download?datasetVersionNumber=1
2. Load the dataset in the python.
3. Visualize the features and targets in the python using the script provided.
4. Divide the data into training and testing
5. Train the deep learning algorithm
6. Test the algorithm and report the performance
7. Vary the parameters of the deep learning network and hyperparameters of training
8. Report the performance in various parameters

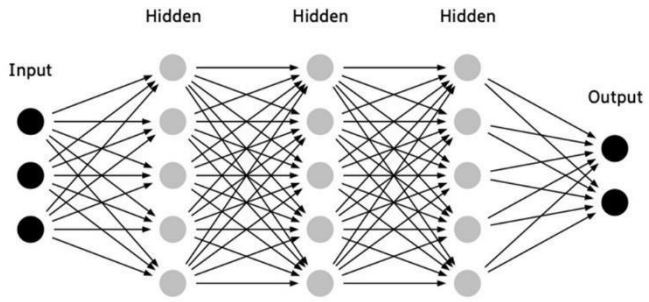

Figure 2. A deep neural network architecture

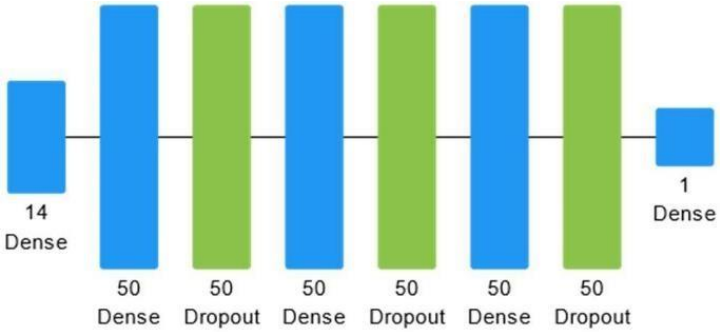

Figure 3. Neural Network Architecture used in this lab

Results:

Table 2. Mean training and test accuracy and loss (in %)

	Accuracy	Loss
Train	0.7339	0.5550
Test	0.7320	0.5425

- The training accuracy is more than testing accuracy
- The training loss is less than testing loss
- The training accuracy and training loss is similar to testing accuracy and testing loss until 200 epochs
- All the four quantities saturate and doesn't improve significantly

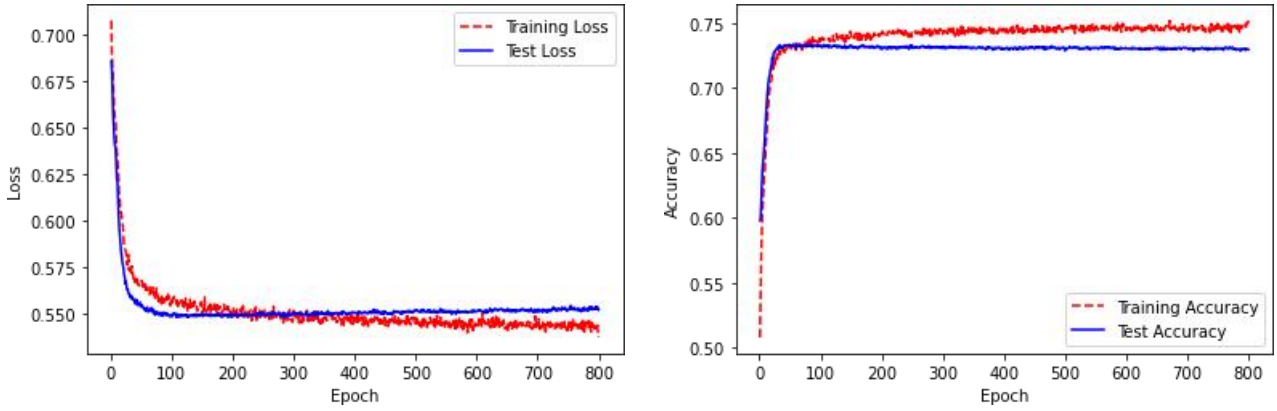

Figure 4. Training and Testing performance of the deep learning model

Table 3. Effect of the number of epochs on the model performance

Epoch	Accuracy	Precision	Recall	F1
100	0.7303	0.732	0.730	0.730
200	0.732	0.737	0.732	0.730
400	0.729	0.732	0.729	0.728
600	0.728	0.733	0.728	0.727
800	0.729	0.731	0.729	0.728

Table 4. Effect of batch size on the model performance

Batch Size	Accuracy	Precision	Recall	F1
256	0.732	0.736	0.732	0.731
512	0.732	0.732	0.732	0.732
1024	0.733	0.736	0.733	0.733
2048	0.730	0.732	0.730	0.730

- The prediction of cardiovascular risk factor was performed using deep learning method
- The parameters of the deep learning algorithm affect the performance of the classifier

CODE:

```
[Install required packages before]
neural_network = tf.keras.models.Sequential([
# Input Layer
tf.keras.layers.Dense(units=X_train.shape[1], activation='relu', input_shape=(11,)),
# Hidden Layer
tf.keras.layers.Dense(units=50, activation='relu'),
tf.keras.layers.Dropout(rate=0.5),
tf.keras.layers.Dense(units=50, activation='relu'),
tf.keras.layers.Dropout(0.5),
tf.keras.layers.Dense(units=50, activation='relu'),
tf.keras.layers.Dropout(0.5),
# Output Layer
tf.keras.layers.Dense(units=2, activation='softmax')
])neural_network.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=1e-3),
loss=tf.keras.losses.CategoricalCrossentropy(from_logits=True),
metrics = tf.keras.metrics.CategoricalAccuracy())
print(neural_network.summary())
BATCH_SIZE = 256
# ES = tf.keras.callbacks.EarlyStopping(monitor='val_loss', patience=5)
# MC = tf.keras.callbacks.ModelCheckpoint('/kaggle/working/model_weights.h5', monitor='val_loss',
save_best_only=True)
# RLR = tf.keras.callbacks.ReduceLROnPlateau(monitor='val_loss', factor=.2, patience=2)
# STEPS_PER_EPOCHS = round(X_train.shape[0] / BATCH_SIZE)
history = neural_network.fit(
X_train,
y_train,
epochs=10,
batch_size=BATCH_SIZE
)
prediction = neural_network.predict(X_test)
prediction = np.argmax(prediction,axis=1)
prediction
prediction.shape
X_test.shape
y_label = np.argmax(y_test)
y_label
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report,
ConfusionMatrixDisplay
confusion = confusion_matrix(y_test,prediction)
confusion
report = classification_report(y_test,prediction)
print(report)
```

Conclusions:

The cardiovascular disease risk factor can be estimated using the objective, examination and subjective features and deep learning algorithm. The parameters batch size and number of epochs affects the performance of the model and need to be carefully selected for accurate prediction of CVDs.