ML ASSIGNMENT 3

NAME : Nishit Reddy Lingala

Id: 700747128

Git hub: https://github.com/Nishitreddy/machine_learning.git

Video link:
https://drive.google.com/file/d/1QnvRRXRKCM9n0byyHp2O29ieI6odifY9/view?usp=share_link

In [1]:
```
#Question 1.a
import numpy as np
integer_vec = np.random.randint(low=1, high=21, size=15)
arrTest = integer_vec.reshape((3, 5))
print("Array shape:", arrTest.shape)
arrTest[np.arange(arrTest.shape[0]), np.argmax(arrTest, axis=1)] = 0
print("Modified array:\n", arrTest)
```

```
Array shape: (3, 5)
Modified array:
 [[ 0 17  4 14  1]
 [15 16  0  4  6]
 [ 8 10  2  0  6]]
```

We use random.randint present in numpy to create the random values
Then we use the argmax, arrange functions

In [2]:
```
#Question 1.b
import numpy as np

# Define the square array
A = np.array([[3, -2], [1, 0]])

# Compute the eigenvalues and right eigenvectors
eigenvalues, eigenvectors = np.linalg.eig(A)

# Print the results
print("Eigenvalues:", eigenvalues)
print("Right eigenvectors:\n", eigenvectors)
```

```
Eigenvalues: [2. 1.]
Right eigenvectors:
 [[0.89442719 0.70710678]
 [0.4472136  0.70710678]]
```

We use lingasl.eig to compute the eigen value and right eigen vector

In [3]:
```
#Question 1.c
import numpy as np

# Define the array
input = np.array([[0, 1, 2], [3, 4, 5]])

# Compute the sum of the diagonal elements
diagonalSum = np.trace(input)

# Print the result
print("Sum of diagonal elements:", diagonalSum)
```

```
Sum of diagonal elements: 4
```

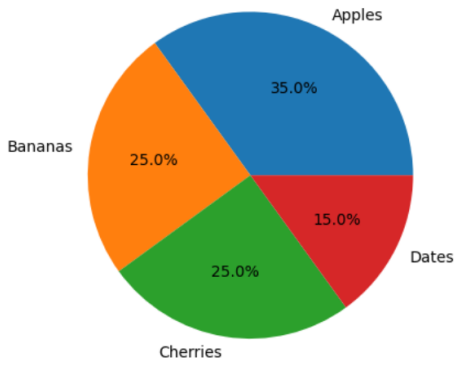
In this we use the trace function to find the sum of diagonal element

In [4]:
```
#Question 1.d
import numpy as np

# Create a NumPy array
arr = np.array([[1, 2], [3, 4], [5, 6]])

# Reshape the array to 2x3 shape without changing data
new_arr = arr.reshape(2, 3)

# Print the original and new array
print("Original array:")
print(arr)
print("\nNew array:")
print(new_arr)
```

```
Original array:
[[1 2]
 [3 4]
 [5 6]]

New array:
[[1 2 3]
 [4 5 6]]
```

Reshape function from the numpy module is used to reshape the array

In [5]:
```
#Question2
import matplotlib.pyplot as plt
import numpy as np

input = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]

plt.pie(input, labels = mylabels, autopct='%1.1f%%')
plt.show()
```



Here we use matploitlib.pyplot and use to pie function to generate the pie chart of given data.