Random variable

A random variable is a variable whose value is unknown to the function i.e, the value is depends upon the outcome of experiment

For example, while throwing a dice, the variable value is depends upon the outcome.

Mostly random variables are used for regression analysis to determine statistical relationship between each other. There are 2 types of random variable:

1 — Continuous random variable

2 — Discrete random variable

Continuous random variable:- A variable which having the values between the range/interval and take infinite number of possible ways is called Continuous random variable . OR the variables whose values are obtained by measuring is called Continuous random variable. For e.g, A average height of 100 peoples, measurement of rainfall

Discrete Random Variable:-A variable which takes countable number of distinct values. OR the variables whose values are obtained by counting is called Discrete Random Variable. For e.g, number of students present in class

In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline

n=np.random.randint(2,10,40)
print(n)
```

```
[3 9 4 8 7 8 5 3 2 7 4 9 2 9 9 4 5 8 9 3 6 3 8 4 3 2 2 9 3 5 6 2 9 3 7 5 3
 8 2 2]
```

Probability Mass Function

Probability Mass Function is a frequency function which gives usprobability for discrete random variable.

In [4]:
```
#Convert list n to dataframe
df=pd.DataFrame(n)

#Count each variable how many times repeated
df=pd.DataFrame(df[0].value_counts())
df
```

Out[4]:

	0
3	8
9	7
2	7
8	5
5	4
4	4
7	3
6	2

In [5]:
```
length=len(n)
length
```

Out[5]:
```
40
```

In [9]:
```
#Give name for the column
df.columns=['Counts']
df
```

Out[9]:

	Counts
3	8
9	7
2	7
8	5
5	4
4	4
7	3
6	2

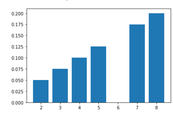
In [10]:
```
# To find probability mass function(PMF) take count value of each number and devide by entire length
df['Prob']=df['Counts']/length
df
```

Out[10]:

	Counts	Prob
3	8	0.200
9	7	0.175
2	7	0.175
8	5	0.125
5	4	0.100
4	4	0.100
7	3	0.075
6	2	0.050

In [11]:
```
# Plot PMF
plt.bar(df['Counts'], df['Prob'])
```

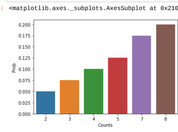
Out[11]:
```
<BarContainer object of 8 artists>
```



In [12]:
```
# Plot PMF using Seaborne
import seaborn as sns

sns.barplot(df['Counts'],df['Prob'])
```

Out[12]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2105a291100>
```



In [13]:
```
# Another example for PMF

data={'Candy':['Blue','Orange','Green','Purple'],
      'Total':[30000,10000,20000,12000]}

df=pd.DataFrame(data)
df
```

Out[13]:

	Candy	Total
0	Blue	30000
1	Orange	10000
2	Green	20000
3	Purple	12000

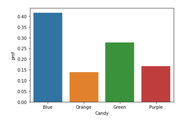
In [14]:
```
df["pmf"]=df["Total"]/df["Total"].sum()
df
```

Out[14]:

	Candy	Total	pmf
0	Blue	30000	0.416667
1	Orange	10000	0.138889
2	Green	20000	0.277778
3	Purple	12000	0.166667

In [15]:
```
# Plot it
sns.barplot(df['Candy'],df['pmf'])
```

Out[15]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2105a3187f0>
```



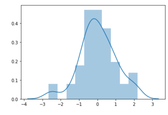
Probability Density Function

It is a statistical expression that defines probability distribution function for continous random variable. e.g. What is the probability of a person that weigh between 50 to 100 kg

$$f(x) = P(a <= x <= b) = integration\ of\ a\ to\ b\ f(x)dx >= 0$$

In [19]:
```
data=np.random.normal(size=100)
data=np.append(data, [1.2,1.2,1.2,1.2,1.2])
sns.distplot(data)
```

Out[19]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2105a499160>
```



In [20]:
```
#Another example
import scipy.stats as stats

mu=20
sigma=2
h=sorted(np.random.normal(mu,sigma, 100))
```

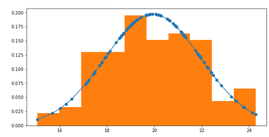
In [36]:
```
import scipy.stats as stats
plt.figure(figsize=(15,5))

fit=stats.norm.pdf(h, np.mean(h), np.std(h))

plt.plot(h,fit,'-o')

plt.hist(h,density=True)
```

Out[36]:
```
(array([0.02169604, 0.03254406, 0.13017625, 0.13017625, 0.19526437,
        0.15187229, 0.16272031, 0.15187229, 0.04339208, 0.06508812]),
 array([15.05803606, 15.97986321, 16.90169036, 17.82351751, 18.74534466,
        19.66717182, 20.58899897, 21.51082612, 22.43265327, 23.35448042,
        24.27630757]),
 <a list of 10 Patch objects>)
```



Cummulative distribution function(CDF)

Cummulative distribution function(CDF) is another function to describe the distribution of random variable. It can be defined for any type of random variable i.e. discrete as well as continous or even mixed CDF gives you cummulative probability associated with it.

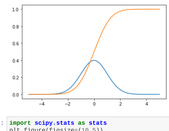
In [25]:
```
import scipy.stats as ss

x=np.linspace(-5,5,5000)
mu=0
sigma=1

y_pdf=ss.norm.pdf(x,mu,sigma)    #the normal pdf
y_cdf=ss.norm.cdf(x,mu,sigma)    #the normal cdf

plt.plot(x, y_pdf, label='pdf')
plt.plot(x, y_cdf, label='cdf')
```

Out[25]:
```
[<matplotlib.lines.Line2D at 0x2105a482520>]
```



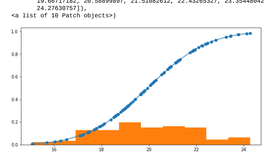
In [35]:
```
import scipy.stats as stats
plt.figure(figsize=(15,5))

fit=stats.norm.cdf(h, np.mean(h), np.std(h))

plt.plot(h,fit,'-o')

plt.hist(h,density=True)
```

Out[35]:
```
(array([0.02169604, 0.03254406, 0.13017625, 0.13017625, 0.19526437,
        0.15187229, 0.16272031, 0.15187229, 0.04339208, 0.06508812]),
 array([15.05803606, 15.97986321, 16.90169036, 17.82351751, 18.74534466,
        19.66717182, 20.58899897, 21.51082612, 22.43265327, 23.35448042,
        24.27630757]),
 <a list of 10 Patch objects>)
```



In [ ]:
```

```