	<pre>import numpy as np import pandas as pd import matplotlib.pyplot as plt</pre>
In [2]	<pre>import seaborn as sns from statsmodels.distributions.empirical_distribution import ECDF from scipy.stats import norm  ! gdown 1f7ca1XrMunwDr-4r_0-sDZJm5kEh3bIZ</pre>
	Downloading From: https://drive.google.com/uc?id=1f7ca1XrMunwDr-4r_0-sDZJm5kEh3bIZ To: /Users/nikhilsanghi/Downloads/01_dsml-course-main-live/batches/1_Dec_Beg_Mon_Dec_Beg_Tue/Prob_Stats/07_Gaussian_Distribution_2/weight-height.csv 100%  428k/428k [00:00<00:00, 1.21MB/s]  df=pd.read_csv("weight-height.csv") df
Out[3]	
	<ul> <li>Male 74.110105 212.740856</li> <li>Male 71.730978 220.042470</li> <li>Male 69.881796 206.349801</li> </ul>
	<ul> <li></li> <li>9995 Female 66.172652 136.777454</li> <li>9996 Female 67.067155 170.867906</li> </ul>
	9997       Female       63.867992       128.475319         9998       Female       69.034243       163.852461         9999       Female       61.944246       113.649103
In [4]	10000 rows × 3 columns  : sns.histplot(df["Height"]) plt.show()
	400 -
	300 - 200 -
	100 - 100 -
In [5]	<pre>sns.displot(df["Height"], kde=True) plt.show()</pre>
	400 -
	300 - 100 - 200 -
In [6]	0 55 60 65 70 75 80 Height  sns.kdeplot(df["Height"])
	plt.show() 0.08
	0.06 - 
	0.00
In [7]	: count 10000.000000
	mean 66.367560 std 3.847528 min 54.263133 25% 63.505620 50% 66.318070
In [8]	75% 69.174262 max 78.998742 Name: Height, dtype: float64  c=ECDF(df["Height"])
	<pre>plt.plot(e.x,e.y)  : [<matplotlib.lines.line2d 0x7fc5f0baa130="" at="">]  10 </matplotlib.lines.line2d></pre>
	0.8 - 0.6 -
	0.4 - 0.2 -
In [ ]	0.0
In [ ]	
In [10]	<pre>sns.kdeplot(df["Height"]) plt.show()</pre>
	0.06 - \frac{1}{2}
	0.02
in [11]	0.00 55 60 65 70 75 80 Height  sns.kdeplot(df["Height"]*2.54) plt.show()
	0.035 -
	0.025 - 
	0.005 - 0.000 130 140 150 160 170 180 190 200 Height
n [16] ut[16]	<pre>z_score=(df["Height"]-df["Height"].mean())/df["Height"].std() z_score</pre>
	1 0.627505 2 2.012343 3 1.393991 4 0.913375
	9995 -0.050658 9996 0.181830 9997 -0.649655 9998 0.693090 9999 -1.149651
n [17]	plt.show()
	0.35 - 0.30 - 0.25 -
	0.05 0.00 -3 -2 -1 0 1 2 3 4 Height
in [18] Out[18]	1 0.627505
	2
	9997 -0.649655 9998 0.693090 9999 -1.149651 Name: Height, Length: 10000, dtype: float64
in [19]	<pre>sns.kdeplot(z_score_cm) plt.show()</pre>
	0.25 - \$\frac{1}{2} 0.20 - \frac{1}{2} 0.15 - \fra
	0.10
n [20]	-3 -2 -1 0 1 2 3 4  Height  desc=df["Height"].describe() desc
ıt[20]	mean 66.367560 std 3.847528 min 54.263133 25% 63.505620
[21]	50% 66.318070 75% 69.174262 max 78.998742 Name: Height, dtype: float64  cd_mean=desc["mean"]
ıt[21]	<pre>d_mean : 66.36755975482106 : d_std=desc["std"]</pre>
	<pre>d_std : 3.8475281207732324 : d_median=desc["50%"]</pre>
t[23]	<pre>d_median=desc["50%"] d_median  : 66.31807008178464  : norm.cdf?</pre>
[25]	<pre>norm.cdf? : norm.cdf(0) : 0.5</pre>
t[26]	<pre>norm.cdf(x=d_mean, loc=d_mean, scale=d_std)  norm.cdf(x=d_median, loc=d_mean, scale=d_std)  norm.cdf(x=d_median, loc=d_mean, scale=d_std)</pre>
t[27]	<pre>norm.cdf(x=d_median, loc=d_mean, scale=d_std)  0.4948686590493638  norm.cdf(1)-norm.cdf(-1)</pre>
[29]	: 0.6826894921370859 : norm.cdf(2)-norm.cdf(-2) : 0.9544997361036416
[30]	: 0.9544997361036416 : norm.cdf(3)-norm.cdf(-3) : 0.9973002039367398
t[31]	: norm.ppf(0.5) : 0.0
	norm.ppf(0.8413)  0.9998150936147446  norm.ppf(0.5)
[33] t[33]	: norm.cdf(1) : 0.8413447460685429
t[34]	norm.ppf(0.8413447460685429)  1.0  norm.ppf(0.45)
[36]	: -0.12566134685507402 : 40+((norm.ppf(0.45))*6)
[39]	39.246031918869555  norm.ppf(q=0.45,loc=40,scale=6)  39.246031918869555
[40] t[40]	norm.ppf(q=0.86,loc=40,scale=6) 46.481916044889736
	<pre>norm.cdf(x=2.3,loc=3,scale=0.5)  0.08075665923377101  norm.cdf(x=834,loc=800,scale=40)-norm.cdf(x=778,loc=800,scale=40)</pre>
t[42]	: 0.5111777700889613 : norm.ppf(.85)
	: 1.0364333894937898 : norm.ppf(.12) : -1.1749867920660904
[44]	: -1.1749867920660904 : norm.cdf(x=53,loc=50,scale=2)
t[43] [44] t[44] [45]	0.9331927987311419
it[43] i [44] it[44] it[45] it[45] it[47]	<ul> <li>0.9331927987311419</li> <li>norm.cdf(1.5)</li> <li>0.9331927987311419</li> </ul>
ut[43] n [44] ut[44] n [45] ut[45] n [47] ut[47] n [ ] n [48]	<ul> <li>0.9331927987311419</li> <li>norm.cdf(1.5)</li> <li>0.9331927987311419</li> </ul>
it[43] it[44] it[44] it[44] it[45] it[45] it[47] it[47] it[47] it[47] it[47]	: 0.9331927987311419 : norm.cdf(1.5) : 0.9331927987311419 : norm.cdf(x=72.5,loc=65,scale=2.5)-norm.cdf(x=60,loc=65,scale=2.5) : 0.9758999700201907
ut[43] n [44] ut[44] n [45] ut[45] n [47] ut[47] n [48] n [48] n [49] ut[49] n [50]	: 0.9331927987311419 : norm.cdf(1.5) : 0.9331927987311419 : norm.cdf(x=72.5, loc=65, scale=2.5)-norm.cdf(x=60, loc=65, scale=2.5) : 0.9758999700201907 : norm.cdf(x=54, loc=50, scale=2)-norm.cdf(x=46, loc=50, scale=2) : 0.9544997361036416 : norm.ppf(q=0.69, loc=50, scale=2)
ut[43]  n [44]  ut[44]  n [45]  ut[45]  n [47]  ut[47]  in [48]  ut[48]  n [49]  ut[49]  in [50]  ut[50]  n [51]	: 0.9331927987311419 : norm.cdf(1.5) : 0.9331927987311419 : norm.cdf(x=72.5,1oc=65,scale=2.5)-norm.cdf(x=60,1oc=65,scale=2.5) : 0.9758999700201907 : norm.cdf(x=54,1oc=50,scale=2)-norm.cdf(x=46,1oc=50,scale=2) : 0.9544997361036416 : norm.pf(q=8.69,1oc=59,scale=2) : 50.991700694694906
ut[43]  n [44]  ut[44]  n [45]  ut[45]  n [47]  ut[47]  in [48]  ut[48]  n [49]  ut[49]  ut[50]  n [51]  ut[51]  ut[52]	: 0.0351027087311410  : norm.cdf(1.5)  : 0.0351027087311410  : norm.cdf(x=72.5,loc=85,scale=2.5)-norm.cdf(x=88,loc=85,scale=2.5)  : norm.cdf(x=72.5,loc=85,scale=2.5)-norm.cdf(x=88,loc=85,scale=2.5)  : norm.cdf(x=54,loc=58,scale=2)-norm.cdf(x=88,loc=59,scale=2)  : norm.cdf(x=54,loc=58,scale=2)-norm.cdf(x=68,loc=59,scale=2)  : norm.ppf(q=8.69,loc=58,scale=2)  : norm.ppf(q=8.69,loc=58,scale=2)  : norm.ppf(q=8.69,loc=58,scale=2)  : onrm.ppf(0=8.99,loc=58,scale=2)
ut[43]  n [44]  ut[44]  n [45]  ut[45]  n [47]  ut[47]  In [48]  ut[48]  n [49]  ut[49]  ut[50]  n [51]  ut[51]  n [52]  ut[52]  ut[53]	: 0.9331927987311419 : murm.cuff(1.5) : 0.9331927987311419 : murm.cuff(x=72.5, loc=65, scale=2.5)-norm.coff(x=68, loc=65, scale=2.5) : 0.9758999780281987 : marm.cuff(x=54, loc=56, scale=2)-norm.cdf(x=46, loc=56, scale=2) : 0.9544997381036416 : narm.pff(q=0.69, loc=56, scale=2) : 9.991708094094096 : murm.ppf(q=0.69, loc=56, scale=2) : 9.495853473474532 : 58+(0.495693473474532*2) : 58+(0.495693473474532*2) : 58-(0.495693473474532*2) : 58-(0.495693473474532*2) : 58-(0.495693473474532*2) : 58-(0.495693473474532*2) : 58-(0.495693473474532*2) : 58-(0.495693473474532*2) : 58-(0.495693473474532*2) : 58-(0.495693473474532*2) : 58-(0.495693473474532*2) : 58-(0.495693473474532*2) : 58-(0.495693473474532*2)
ut[43] n [44] ut[44] n [45] ut[45] n [47] ut[47] In [48] ut[48] n [49] ut[49] In [50] ut[50] n [51] ut[51] ut[52] ut[52] n [53] ut[53] n [54] ut[54] n [55]	: 0.9331927887311419  : marm.cdf(1.5)  : 0.9331927887311419  : marm.cdf(x=72.5,1m=85, stal=e2.5) marm.cdf(x=60,1m=85, scal=e2.5)  0.974897881030416  : marm.cdf(x=72.5,1m=85, stal=e2.5) marm.cdf(x=45,1m=85, scal=e2.5)  : marm.cdf(x=51,1m=85, scal=e2.5) marm.cdf(x=45,1m=85, scal=e2.5)  : marm.cdf(x=51,1m=85,1m=85, scal=e2.5) marm.cdf(x=45,1m=85, scal=e2.5)  : marm.cdf(x=51,1m=85,1m=8
ut [43]  n [44]  ut [44]  n [45]  n [47]  ut [47]  In [48]  ut [48]  n [49]  ut [49]  ut [50]  n [51]  ut [51]  n [52]  ut [52]  n [53]  ut [53]  ut [54]  n [55]  n [56]	: 0.0331927067311410 : norm.cdf(x=50, hu=50, scalve2.b)-norm.ruf(x=60, hu=50, muh=2.8) : 0.0331927067311410 : norm.cdf(x=72.b, hu=50, scalve2.b)-norm.ruf(x=60, hu=50, muh=2.8) : 0.033192706739397 : norm.coff(x=50, hu=50, scalve2)-norm.cdf(x=60, hu=50, scalve2) : norm.coff(x=50, hu=50, scalve2)-norm.cdf(x=60, hu=50, scalve2) : norm.puf(x=6.9, hu=50, scalve2) : norm.puf(x=6.9, hu=50, scalve2) : norm.puf(x=6.9, hu=50, scalve2) : norm.puf(x=6.9, hu=50, hu=50, scalve2) : norm.puf(x=6.9, hu=50, hu=50, hu=50, scalve2) : norm.puf(x=6.9, hu=50,