```
# Given a mean and a variance we can plot the normal distribution for
# our engineering measurements

# ---------- Import modules that we need ----------
import matplotlib.pyplot as plt
import numpy as np
import scipy.stats as stats
import math

# Some additional commands (at your choise) to set font globally
# as times new roman, italic, and bold for nicer plots
plt.rcParams["font.family"] = "Times New Roman"
plt.rcParams["font.style"] = 'italic'
plt.rcParams["font.weight"] = 'normal'
plt.rcParams["font.size"] = 16
plt.rcParams["mathtext.fontset"] = 'cm'

# ---------- Statistics Part ----------
mu = 5                  # mean value
variance = 1            # variance in measured data
sigma = math.sqrt(variance)# standard deviation of measured data

# Generate x-axis with 3 standard deviations to the left of the mean
# and 3 to the right of it.
x = np.linspace(mu - 3*sigma, mu + 3*sigma, 100)
shade = np.linspace(mu - 2*sigma, mu + 2*sigma, 100)

plt.plot(x, stats.norm.pdf(x, mu, sigma),'-b')
plt.axvline(x=mu+2*sigma, linestyle='--',c='orange')
plt.axvline(x=mu-2*sigma, linestyle='--',c='orange')
plt.fill_between(shade,stats.norm.pdf(shade, mu, sigma),alpha=0.45, color='teal')
```

```
plt.arrow(mu,0.1,2*sigma,0,head_width=0.01, head_length=0.15,
        length_includes_head=True,color= 'k',linewidth=1)
plt.arrow(mu,0.1,-2*sigma,0,head_width=0.01, head_length=0.15,
        length_includes_head=True,color= 'k',linewidth=1)

plt.text(mu-sigma,0.11,'95% Conf. Int.')
plt.text(mu-sigma,0.071,r'$\mathrm{\bar{x}_0 = 5, \sigma = 1}$')
##plt.axis([-6, 2, -3, 3])# set axis limits
ax = plt.gca()
plt.grid(True)
plt.grid(linestyle='dotted')
ax.set_position([0.15,0.17, 0.75, 0.75])

plt.xlabel('Measured Distance, [in]')
plt.ylabel('Probability Density')

plt.show()
```