The Gaussian Distribution

Contents


```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
sns.set(rc={"figure.dpi":100, "savefig.dpi":300})
sns.set_context("notebook")
sns.set_style("ticks")
```

Objectives

- To practice with the Gaussian distribution.

The Normal distribution

The normal (or Gaussian) distribution is a ubiquitous one. It appears over and over again. There are two explanations as to why it appears so often:

- It is the distribution of maximum uncertainty that matches a known mean and a known variance variance.
- It is the distribution that arises when you add a lot of random variables together.

We will learn about both these in the next lectures.

We write:

$$X|\mu,\sigma \sim N(\mu,\sigma),$$

and we read "X conditioned on μ and σ follows a normal distribution with mean μ and variance σ^2.

When $\mu = 0$ and $\sigma^2 = 1$, we say that we have a *standard normal* distribution. Let

$$Z \sim N(0,1).$$

The PDF of the standard normal is:

$$\phi(z) := N(z|0,1) = \frac{1}{\sqrt{2\pi}}\exp\left\{-\frac{z^2}{2}\right\}.$$

The CDF of the standard normal is:

$$\Phi(z) := p(Z \leq z) = \int_{-\infty}^{z} \phi(z')dz',$$

is not analytically available. However, there are codes that can compute it.

Here is how you can get the PDF of the standard normal. First, let's make a standard normal random variable in `scipy.stats`:

```
import scipy.stats as st
Z = st.norm()
```

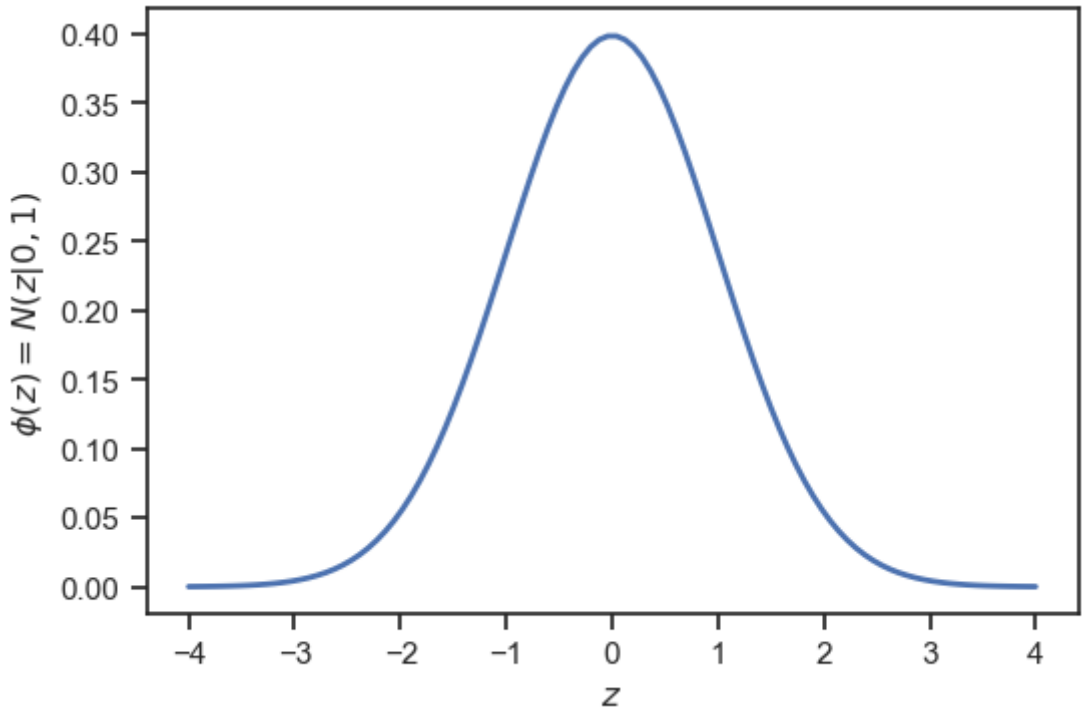
You can evaluate it anywhere you want:

```
print(f"phi(0.5) = {Z.pdf(0.5):.2f}")
```

```
phi(0.5) = 0.35
```

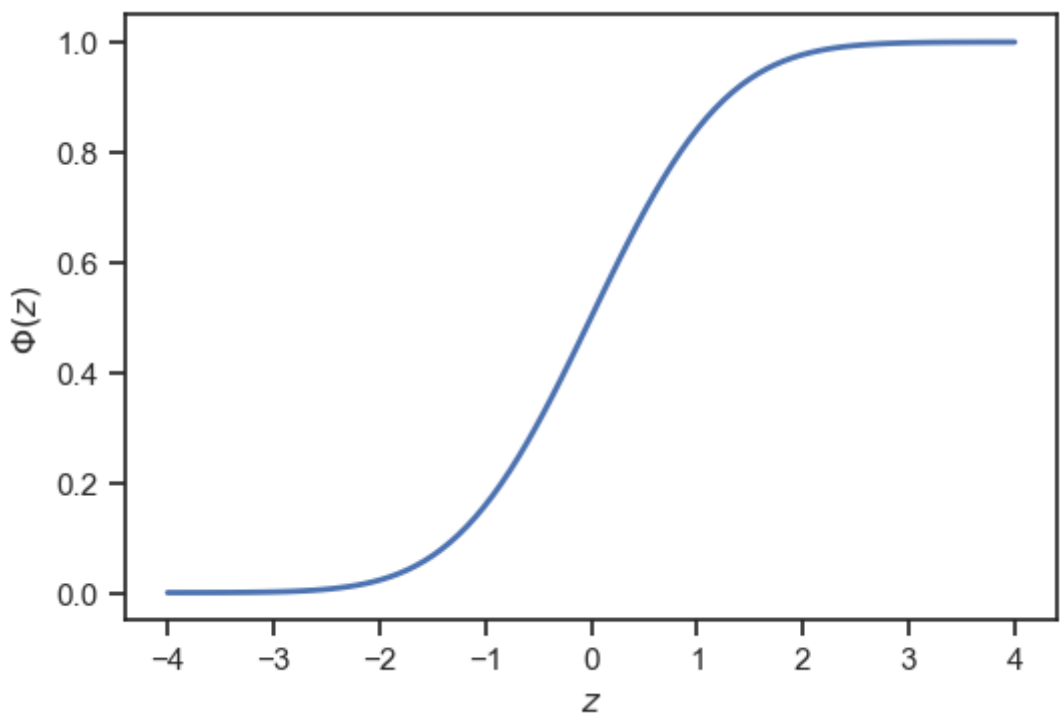
Let's plot the PDF:

```
fig, ax = plt.subplots()
zs = np.linspace(-4.0, 4.0, 100)
ax.plot(zs, Z.pdf(zs), lw=2)
ax.set_xlabel("$z$")
ax.set_ylabel("$\phi(z) = N(z|0,1)$");
```



And here is the CDF:

```
fig, ax = plt.subplots()
ax.plot(zs, Z.cdf(zs), lw=2)
ax.set_xlabel("$z$")
ax.set_ylabel("$\Phi(z)$");
```



Here is the expectation:

```
print(f"E[Z] = {Z.expect():.2f}")
```

```
E[Z] = 0.00
```

And the variance:

```
print(f"V[Z] = {Z.var():.2f}")
```

```
V[Z] = 1.00
```

Here is the probability that Z is between two numbers:

```
a = 1.0
b = 3.0
prob_Z_in_ab = Z.cdf(b) - Z.cdf(a)
print(f"p({a:.2f} <= Z <= {b:.2f}) = {prob_Z_in_ab:.2f}")
```

```
p(1.00 <= Z <= 3.00) = 0.16
```

And here is how you can sample:

```
Z.rvs(100)
```

```
array([-3.58667233,  0.82942319, -0.25653277,  0.43583335,  0.68975566,
        0.51848385, -1.34177656, -1.32390597, -1.15299102, -0.7508944 ,
       -0.32263962, -0.21784019,  1.12613805,  1.04215185, -0.33395233,
       -1.54168367, -0.28127672,  1.56696645,  1.1350127 , -2.37963206,
       -0.61948889,  1.00213673, -0.41036464,  1.13026477, -0.59939966,
       -0.93288013,  0.82198916, -0.89842021, -1.25612781, -0.9230532 ,
        0.47120924, -0.3730327 ,  0.92510947, -0.91029047,  1.78859637,
       -0.79326493, -0.97721941, -0.49003939,  1.41059884, -1.05501536,
       -1.53876273,  0.61504445,  0.23264693, -0.5678752 , -0.50096853,
        0.03584837, -1.64530236, -0.95579757, -0.86432902,  2.19674933,
       -0.35008941, -0.73963126, -0.9032805 , -2.58490305, -0.6500442 ,
       -0.07777188,  0.56949711, -1.30210539,  0.14189044,  0.35669112,
        0.0701197 , -0.28244914, -0.79224212, -0.89442166, -1.32821148,
       -0.49202892, -0.15305202, -0.15727341,  0.59379838,  0.71020491,
       -0.50937386,  0.13364911, -1.11032862,  0.0419184 ,  0.49395323,
       -0.48614475,  0.68118233,  0.12655437, -0.53081968,  0.23165793,
        0.71781573, -0.34803331, -1.05500499,  0.26206415, -0.87148726,
       -0.65704631, -0.83408412, -1.25437996,  0.64506716, -0.27488982,
        1.38919422,  1.36548361, -0.61566789,  1.02332063,  0.27090872,
        1.77524751,  0.96148267, -1.52663014,  1.76161908, -0.73263512])
```

And, of course, you can also sample using the functionality of numpy:

```
np.random.randn(100)
```

```
array([-0.23469999,  0.27755578, -1.14542298,  0.61999159,  0.66278615,
       -0.80405238,  1.32698119, -0.55223417, -0.73110381,  0.88875229,
        0.73530515,  0.40675527, -1.90818471,  1.60683929,  0.41899729,
       -1.55848729, -0.34124616, -0.42223755, -0.08155453,  0.24604101,
       -0.51869289, -1.47315225, -0.12449093, -0.91382721,  0.07605132,
        0.18137048,  1.39347593,  0.9795214 ,  0.49578326,  1.47234159,
        0.07898487, -0.83142454,  0.57563444, -0.12894866, -0.9923975 ,
        0.03931736,  0.14816031,  0.03575066,  2.47800784,  0.88038494,
       -0.16582722, -0.81716577,  0.41017561,  0.8516808 ,  0.04927732,
       -0.69102354,  1.95397465,  1.02581568,  0.1200221 ,  0.59705658,
       -2.02627913, -0.19967328, -0.84019389,  0.93886259,  1.13559056,
        0.23267641, -0.02044625, -0.86366042,  1.15554285, -0.34527991,
        0.60110257,  1.05405921, -1.3887897 , -0.37117395,  1.7365931 ,
       -0.49366772,  1.55443796,  1.05094731, -1.38744444,  1.19202857,
        0.32396964,  0.64951179, -0.08338823, -0.3585551 ,  0.65066711,
        0.20636664,  1.14717404, -0.63035333,  2.21709185, -0.70943613,
       -0.77693046, -0.22835926,  0.06231472,  0.99618518, -0.5220756 ,
        0.06119245,  1.10130623,  0.91477383,  0.08696716,  1.00477995,
        1.59614416,  0.97068325,  0.07890009,  0.09593324,  0.62347618,
       -0.23755432,  1.17106149,  2.43469512,  0.11457891, -0.29778902])
```

Quantiles of the normal

There are a few more interesting things to know about the standard normal. For, example how can you find a value z_q such that the probability of Z being less that z_q is q%. Mathematically, you wish to find this:

$$\Phi(z_q) = \frac{q}{100}.$$

The point z_q is called the q% quantile. To find it, you need to do this:

$$z_q = \Phi^{-1}\left(\frac{q}{100}\right).$$

For example, z_{50} is called the median (and it coincides with the expectation here). Another set of interesting quantiles is are $z_{2.5}$ and $z_{97.5}$. Why? Because the probability between them is 95%. Here it is:

$$p(z_{2.5} \leq Z \leq z_{97.5}) = \Phi(z_{97.5}) - \Phi(z_{2.5}) = \frac{97.5}{100} - \frac{2.5}{100} = \frac{95}{100}.$$

Let's find these quantiles and visualize them using the functionality of `scipy.stats`. We will use the percent point function (ppf) which the inverse of the CDF:

```
z_025 = Z.ppf(0.025)
z_500 = Z.ppf(0.5)
z_975 = Z.ppf(0.975)
print(f"2.5% quantile of Z = {z_025:.2f}")
print(f"50% quantile of Z = {z_500:.2f}")
print(f"97.5% quantile of Z = {z_975:.2f}")
```

```
2.5% quantile of Z = -1.96
50% quantile of Z = 0.00
97.5% quantile of Z = 1.96
```

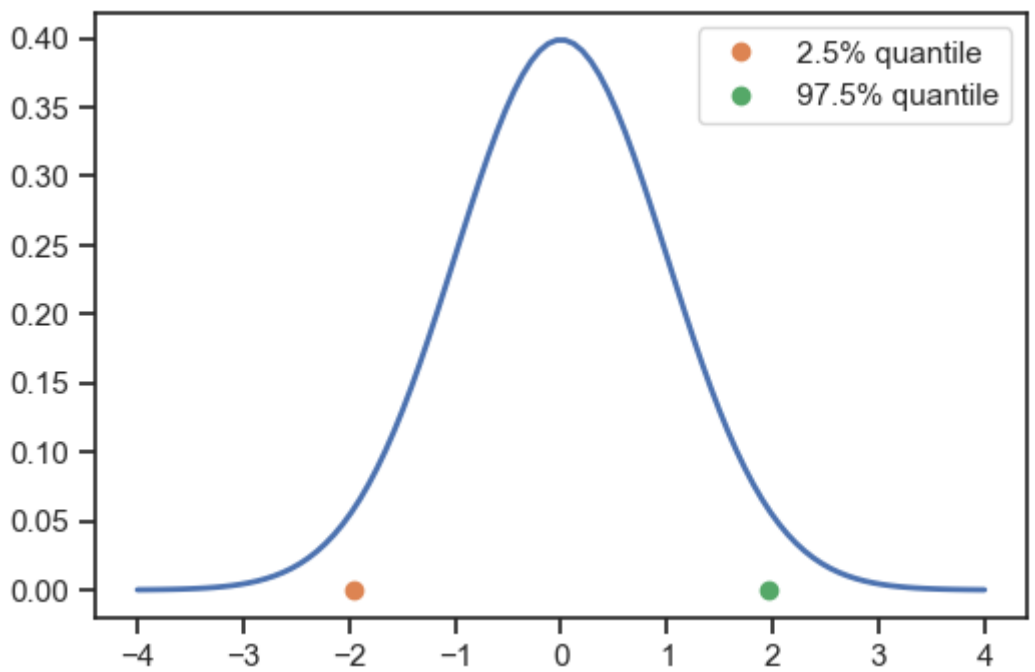
Here is how much probability there is between the two extreme quantiles:

```
print(f"p({z_025:.2f} <= Z <= {z_975:.2f}) = {Z.cdf(z_975) - Z.cdf(z_025):.2f}")
```

```
p(-1.96 <= Z <= 1.96) = 0.95
```

Let's also visualize the quantiles on top of the PDF:

```
fig, ax = plt.subplots()
ax.plot(zs, Z.pdf(zs), lw=2)
ax.plot(z_025, [0.0], "o", label="2.5% quantile")
ax.plot(z_975, [0.0], "o", label="97.5% quantile")
plt.legend(loc="best");
```



Question

- Modify the code above so that you find and vizualize $z_{0.001}$ and $z_{99.999}$.
- What is the difference between $z_{99.999}$ and $z_{0.001}$?
- What is the probability that Z is between $z_{99.999}$ and $z_{0.001}$?

Getting any normal from the standard normal

Using the standard normal, we can express any normal. It is easy to show that:

$$X = \mu + \sigma Z,$$

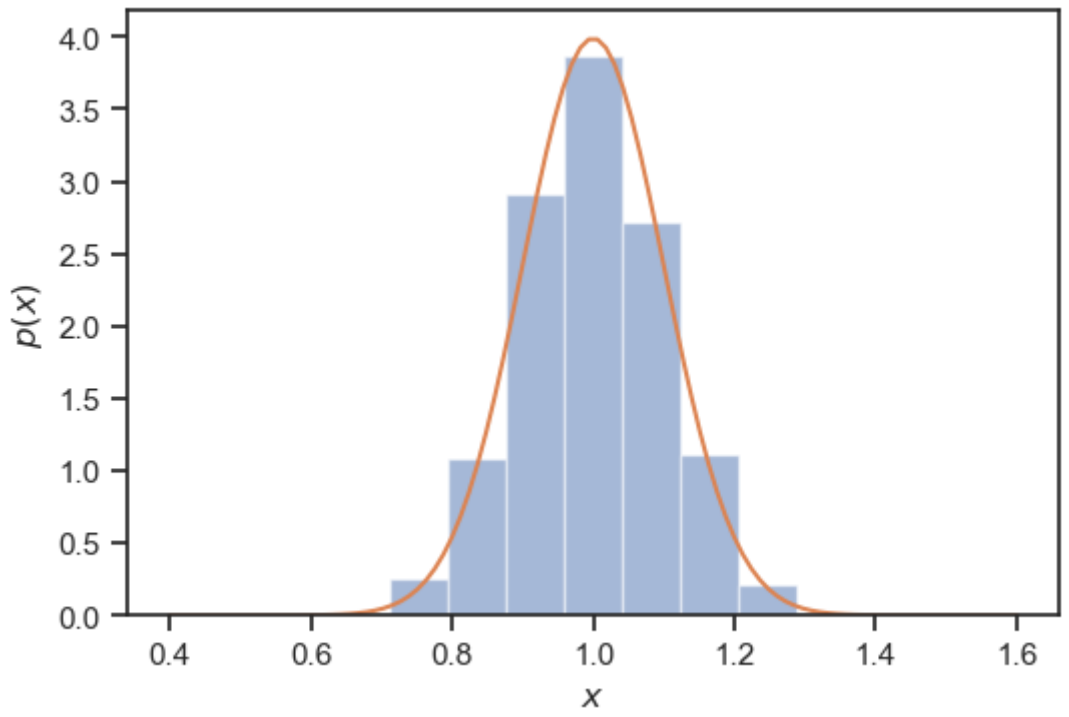
follows a $N(\mu, \sigma^2)$ if Z follows $N(0, 1)$. Note that σ is called the **standard deviation** of X (the standard deviation of a random variable is just the square root of the variance). You must remember this! It is extremely useful and it will appear again and again. For example, using this relationship you can sample from any normal using samples from the standard normal.

Let's take some samples exploiting this relationship and then compare the histogram to the true PDF.

```
mu = 1.0
sigma = 0.1
X = st.norm(mu, sigma)
xs = np.linspace(mu - 6.0 * sigma, mu + 6.0 * sigma, 100)
x_samples = mu + sigma * Z.rvs(size=10000)
```

And here is their histogram compared to the PDF of $N(\mu, \sigma^2)$:

```
fig, ax = plt.subplots()
ax.hist(x_samples, density=True, alpha=0.5)
ax.plot(xs, X.pdf(xs))
ax.set_xlabel("$x$")
ax.set_ylabel("$p(x)$");
```



How can you find the quantiles of this normal? Well, you can simply use the functionality of `scipy.stats`. As an example, let's find $x_{2.5}$:

```
x_025 = X.ppf(0.025)
print(f"2.5% quantile of N({mu:.2f}, {sigma:.2f}^2) = {x_025:1.2f}")
```

```
2.5% quantile of N(1.00, 0.10^2) = 0.80
```

But we can also find this quantile by exploiting the connection between X and Z. The definition of a quantile of X is:

$$p(X \leq x_q) = \frac{q}{100}.$$

But, since $X = \mu + \sigma Z$, this is equivalent to:

Begin solving for the q-quantile of Z: $$p(\mu + \sigma Z \leq x_q) = \frac{q}{100},$$

which becomes:

$$p(\sigma Z \leq x_q - \mu) = \frac{q}{100},$$

and then:

$$p\left(Z \leq \frac{x_q - \mu}{\sigma}\right) = \frac{q}{100}.$$

This is just:

$$\Phi\left(\frac{x_q - \mu}{\sigma}\right) = \frac{q}{100},$$

which tells us that $\frac{x_q - \mu}{\sigma}$ is the q-quantile of Z, i.e.,

$$z_q = \frac{x_q - \mu}{\sigma}.$$

Solving for x_q, we get:

$$x_q = \mu + \sigma z_q.$$

Where X is some non-standard normal and Z is the standard normal

Let's do a sanity check:

```
z_025 = Z.ppf(0.025)
print(f"mu + sigma * z_025 = {mu + sigma * z_025:.2f}")
```

```
mu + sigma * z_025 = 0.80
```

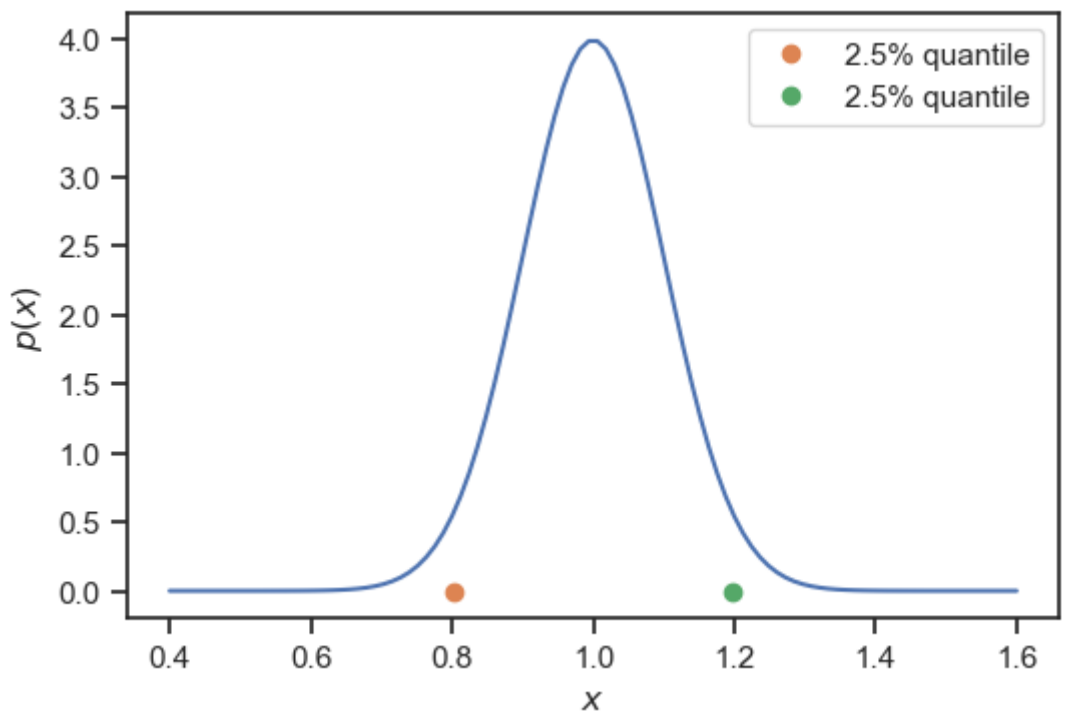
which is the same as what we found before. So, let's find also the 97.5% quantile:

```
z_975 = Z.ppf(0.975)
x_975 = mu + sigma * z_975
print(f"97.5% quantile of N({mu:.2f}, {sigma:.2f}^2) = {x_975:1.2f}")
```

```
97.5% quantile of N(1.00, 0.10^2) = 1.20
```

Let's visualize the quantiles like we did before:

```
fig, ax = plt.subplots()
ax.plot(xs, X.pdf(xs))
ax.plot(x_025, 0, "o", label="2.5% quantile")
ax.plot(x_975, 0, "o", label="2.5% quantile")
ax.set_xlabel("$x$")
ax.set_ylabel("$p(x)$")
plt.legend(loc="best");
```



Now, let's find the distance between $x_{2.5}$ and $x_{97.5}$ in terms of the standard deviation σ. We have:

$$x_{97.5} - x_{2.5} = \mu + \sigma z_{97.5} - \mu - \sigma z_{2.5} = \sigma(z_{97.5} - z_{2.5}).$$

This is:

```
print(f"x_975 - x_025 ~= sigma * {z_975 - z_025:.2f}")
```

```
x_975 - x_025 ~= sigma * 3.92
```

Okay. So we see that 95% of the probability is contained within a 3.92σ interval. This interval is centered at the median (which here happends to be the same as the mode and the expectation of the probability density). The value 3.92 is a little bit awkward, so we are going to round up

to 4σ intervals. That is slightly more than 95% of the probability, but it's simpler to remember. So, remember:

$$p(\mu - 2\sigma < X < \mu + 2\sigma) \approx 0.95,$$

for a normal random variable $N(\mu, \sigma^2)$.

Given that approximately 95% of the probability lies within mu +/- 2*sigma; true value is mu +/- 1.96*sigma

Questions

- Write code that finds exactly how much probability there is between $\mu - 2\sigma$ and $\mu + 2\sigma$, i.e., find $p(\mu - 2\sigma < X < \mu + 2\sigma)$.
- Modify the code you just written, to find how much probability there is in $\mu - 3\sigma$ and $\mu + 3\sigma$, i.e., find $p(\mu - 3\sigma < X < \mu + 3\sigma)$. This is six-sigmas interval about the mean. Have you ever heard of the six-sigma process improvement technique?

By Ilias Bilionis (ibilion[at]purdue.edu)