π Estimation

π is a widely used constant that enthusiasts around the world challenged to get to the most precise value. In Machine Learning, especially Monte Carlo simulation, this exercise demonstrates the enormous power of random numbers.

The experiment design is straight-forward: the area ratio can be estimated using the probability of randomly selected points that fall inside of a circle inscribed in a square.

- The area of a circle is πr^2, or just π, and the area of unit square is 1.
- Only use a quarter of the circle and the box for simplicity:
  - points inside of the circle follow $\sqrt{a^2 + b^2} <= 1$

Therefore, probability of points inside of the circle = circle area / area outside of circle.

$$p = \frac{\frac{\pi}{4}}{1 - \frac{\pi}{4}}$$

Solve for π,

$$\pi = \frac{4p}{(1 + p)}$$

In [45]:
```
import numpy as np
counter_a = 0.
counter_b = 0.
ratio = 0.
iteration = 1000000
for a, b in zip(np.random.uniform(0,1,iteration),np.random.uniform(0,1,itera
    if np.sqrt(a**2 + b**2) <= 1.:
        counter_a = counter_a + 1
    elif np.sqrt(a**2 + b**2) > 1.:
        counter_b = counter_b + 1
ratio = counter_a/counter_b
print(f"Estimated pi = {4.*ratio/(1.+ratio):.3f} (pi ~= 3.142)")
```

```
Estimated pi = 3.140 (pi ~= 3.142)
```

Amazing! One would start believing "...order comes out of chaos..."

In [ ]:
```
```