Q1. What is a probability distribution, exactly? If the values are meant to be random, how can you predict them at all?

Probability distribution yields the possible outcomes for any random event. It is also defined based on the underlying sample space as a set of possible outcomes of any random experiment. These settings could be a set of real numbers or a set of vectors or a set of any entities.

In probability theory and statistics, a probability distribution is the mathematical function that gives the probabilities of occurrence of different possible outcomes for an experiment. It is a mathematical description of a random phenomenon in terms of its sample space and the probabilities of events (subsets of the sample space).

For instance, if X is used to denote the outcome of a coin toss ("the experiment"), then the probability distribution of X would take the value 0.5 (1 in 2 or 1/2) for X = heads, and 0.5 for X = tails (assuming that the coin is fair). Examples of random phenomena include the weather conditions at some future date, the height of a randomly selected person, the fraction of male students in a school, the results of a survey to be conducted, etc

Q2. Is there a distinction between true random numbers and pseudo-random numbers, if there is one? Why are the latter considered “good enough”?

Pseudorandom Number Generation. Software-generated random numbers only are pseudorandom. They are not truly random because the computer uses an algorithm based on a distribution, and are not secure because they rely on deterministic, predictable algorithms.

Using pseudo-random numbers is perfectly acceptable in this case because there's no quantitative advantage in the degree of randomness.” Similarly, a CD player in “random” mode is probably really playing in pseudo-random mode, with a pattern that is discernible if you listen carefully enough.

Q3. What are the two main factors that influence the behaviour of a "normal" probability distribution?

The two main parameters of a (normal) distribution are the mean and standard deviation.

A normal distribution is symmetric from the peak of the curve, where the mean is. This means that most of the observed data is clustered near the mean, while the data become less frequent when farther away from the mean. The resultant graph appears as bell-shaped where the mean, median, and mode are of the same values and appear at the peak of the curve.

The graph is a perfect symmetry, such that, if you fold it at the middle, you will get two equal halves since one-half of the observable data points fall on each side of the graph.

The parameters determine the shape and probabilities of the distribution. The shape of the distribution changes as the parameter values change.

1. Mean

The mean is used by researchers as a measure of central tendency. It can be used to describe the distribution of variables measured as ratios or intervals. In a normal distribution graph, the mean defines the location of the peak, and most of the data points are clustered around the mean. Any changes made to the value of the mean move the curve either to the left or right along the X-axis.

2. Standard Deviation

The standard deviation measures the dispersion of the data points relative to the mean. It determines how far away from the mean the data points are positioned and represents the distance between the mean and the observations.

On the graph, the standard deviation determines the width of the curve, and it tightens or expands the width of the distribution along the x-axis. Typically, a small standard deviation relative to the mean produces a steep curve, while a large standard deviation relative to the mean produces a flatter curve.

Q4. Provide a real-life example of a normal distribution.

The height of people is an example of normal distribution. Most of the people in a specific population are of average height. The number of people taller and shorter than the average height people is almost equal, and a very small number of people are either extremely tall or extremely short.

Q5. In the short term, how can you expect a probability distribution to behave? What do you think will happen as the number of trials grows?

In the short term the probability distribution will be more discreet. However, as the number of trials grow it will turn out to be one of the distribution types: Normal distribution, chi-square distribution, binomial distribution, poisson distribution, and uniform distribution and so on.

Q6. What kind of object can be shuffled by using random.shuffle?

The shuffle() method takes a sequence, like a list, and reorganize the order of the items. Note: This method changes the original list, it does not return a new list.

Q7. Describe the math package's general categories of functions.

**Function Name Description**

ceil(x) Returns the smallest integral value greater than the number

copysign(x, y) Returns the number with the value of ‘x’ but with the sign of ‘y’

fabs(x) Returns the absolute value of the number

factorial(x) Returns the factorial of the number

floor(x) Returns the greatest integral value smaller than the number

gcd(x, y) Compute the greatest common divisor of 2 numbers

fmod(x, y) Returns the remainder when x is divided by y

frexp(x) Returns the mantissa and exponent of x as the pair (m, e)

fsum(iterable) Returns the precise floating-point value of sum of elements in an iterable

isfinite(x) Check whether the value is neither infinity not Nan

isinf(x) Check whether the value is infinity or not

isnan(x) Returns true if the number is “nan” else returns false

ldexp(x, i) Returns x \* (2\*\*i)

modf(x) Returns the fractional and integer parts of x

trunc(x) Returns the truncated integer value of x

exp(x) Returns the value of e raised to the power x(e\*\*x)

expm1(x) Returns the value of e raised to the power a (x-1)

log(x[, b]) Returns the logarithmic value of a with base b

log1p(x) Returns the natural logarithmic value of 1+x

log2(x) Computes value of log a with base 2

log10(x) Computes value of log a with base 10

pow(x, y) Compute value of x raised to the power y (x\*\*y)

sqrt(x) Returns the square root of the number

acos(x) Returns the arc cosine of value passed as argument

asin(x) Returns the arc sine of value passed as argument

atan(x) Returns the arc tangent of value passed as argument

atan2(y, x) Returns atan(y / x)

cos(x) Returns the cosine of value passed as argument

hypot(x, y) Returns the hypotenuse of the values passed in arguments

sin(x) Returns the sine of value passed as argument

tan(x) Returns the tangent of the value passed as argument

degrees(x) Convert argument value from radians to degrees

radians(x) Convert argument value from degrees to radians

acosh(x) Returns the inverse hyperbolic cosine of value passed as argument

asinh(x) Returns the inverse hyperbolic sine of value passed as argument

atanh(x) Returns the inverse hyperbolic tangent of value passed as argument

cosh(x) Returns the hyperbolic cosine of value passed as argument

sinh(x) Returns the hyperbolic sine of value passed as argument

tanh(x) Returns the hyperbolic tangent of value passed as argument

erf(x) Returns the error function at x

erfc(x) Returns the complementary error function at x

gamma(x) Return the gamma function of the argument

lgamma(x) Return the natural log of the absolute value of the gamma function

Q8. What is the relationship between exponentiation and logarithms?

An exponential function has the form ax, where a is a constant; examples are 2x, 10x, ex. The logarithmic functions are the inverses of the exponential functions, that is, functions that "undo'' the exponential functions,

Q9. What are the three logarithmic functions that Python supports?

The logarithmic functions of Python help the users to find the log of numbers in a much easier and efficient manner.

The following are the variants of the basic log function in Python:

log2(x)

log(x, Base)

log10(x)

log1p(x)

log2(x) - log base 2

The math.log2(x) function is used to calculate the logarithmic value of a numeric expression of base 2.

Syntax:

math.log2(numeric expression)

log(n, Base) - log base n

The math.log(x,Base) function calculates the logarithmic value of x i.e. numeric expression for a particular (desired) base value.

Syntax:

math.log(numeric\_expression,base\_value)

log10(x) - log base 10

The math.log10(x) function calculates the logarithmic value of the numeric expression to the base 10.

Syntax:

math.log10(numeric\_expression)

log1p(x)

The math.log1p(x) function calculates the log(1+x) of a particular input value i.e. x

Note: math.log1p(1+x) is equivalent to math.log(x)

Syntax:

math.log1p(numeric\_expression)