Python Interview Questions

Python Interview Questions

Question 1: What Is Python?

Python is an open-source interpreted language (like PHP and Ruby) with automatic memory management, exceptions, modules, objects, and threads.

The benefits of Python include its simplicity, portability, extensibility, and built-in data structures. As it's open-source, there's also a massive community backing it. Python is best suited for object-oriented programming. It's dynamically typed, so you won't have to state the types of variables when you declare them. Unlike C++, it doesn't have access to public or private specifiers.

Python's functions are first-class objects that make difficult tasks simple. While you can write code quickly, running it will be comparatively slower than other compiled programming languages.

Question 2: What is Python really? You can (and are encouraged) make comparisons to other technologies in your answer.

Here are a few key points:

Python is an interpreted language. That means that, unlike languages like *C* and its variants, Python does not need to be compiled before it is run.

Python is dynamically typed, this means that you don't need to state the types of variables when you declare them or anything like that. You can do things like `x=111` and then `x="I'm a string"` without error

Python is well suited to object-orientated programming in that it allows the definition of classes along with composition and inheritance. Python does not have access specifiers (like C++'s `public`, `private`), the justification for this point is given as "we are all adults here"

In Python, functions are first-class objects. This means that they can be assigned to variables, returned from other functions, and passed into functions. Classes are also first-class objects

Writing Python code is quick but running it is often slower than compiled languages. Fortunately, Python allows the inclusion of C based extensions so bottlenecks can be optimized away and often are. The `numpy` package is a good example of this, it's really quite quick because a lot of the number-crunching it does isn't actually done by Python

Python finds use in many spheres - web applications, automation, scientific modeling, big data applications, and many more. It's also often used as a "glue" code to get other languages and components to play nicely.

Python makes difficult things easy so programmers can focus on overriding algorithms and structures rather than nitty-gritty low-level details.

Question 3: What Native Data Structures Can You Name in Python?

Common native data structures in Python are as follows:

Dictionaries

Lists

Sets

Strings

Tuples

Question 4: Out of all datasets which Are Mutable, and Which Are Immutable?

Lists, dictionaries, and sets are mutable. This means that you can change their content without changing their identity.

Strings and tuples are immutable, as their contents can't be altered once they're created.

Question 5: What's the Difference Between a List and a Dictionary?

A list and a dictionary in Python are essentially different types of data structures. Lists are the most common data types that boast significant flexibility. Lists can be used to store a sequence of objects that are mutable (so they can be modified after they are created).

However, they have to be stored in a particular order that can be indexed into the list or iterated over it (and it can also take some time to apply iterations on list objects).

For example:

```
>>> a = [1,2,3]
>>> a[2]=4
>>> a
[1, 2, 4]
```

In Python, you can't use a list as a "key" for the dictionary (technically you can hash the list first via your own custom hash functions and use that as a key). A Python dictionary is fundamentally an unordered collection of key-value pairs. It's a perfect tool to work with an enormous amount of data since dictionaries are optimized for retrieving data (but you have to know the key to retrieve its value).
It can also be described as the implementation of a hashtable and as a key-value store. In this scenario, you can quickly look up anything by its key, but since it's unordered, it will demand that keys are hashes. When you work with Python, dictionaries are defined within curly braces {} where each item will be a pair in the form key:value.

Question 6: In a List and in a Dictionary, What Are the Typical Characteristics of Elements?

Elements in lists maintain their ordering unless they are explicitly commanded to be re-ordered. They can be of any data type, they can all be the same, or they can be mixed. Elements in lists are always accessed through numeric, zero-based indices.
In a dictionary, each entry will have a key and a value, but the order will not be guaranteed. Elements in the dictionary can be accessed by using their key.
Lists can be used whenever you have a collection of items in an order. A dictionary can be used whenever you have a set of unique keys that map to values.

Question 7: Is There a Way to Get a List of All the Keys in a Dictionary? If So, How Would You Do It?

If the interviewer follows up with this question, try to make your answer as specific as possible.
To obtain a list of all the keys in a dictionary, we have to use function keys():

```
mydict={'a':1,'b':2,'c':3,'e':5}
mydict.keys()
dict_keys(['a', 'b', 'c', 'e'])
```

Question 8: Can You Explain What a List or Dict Comprehension Is?

When you need to create a new list from other iterables, you have to use list comprehensions. As lists comprehensions return list results, they will be made up of brackets that contain the expressions that need to be executed for each element. Along with the loop, these can be iterated over each element.
Example of the basic syntax:

```
new_list = [expression for_loop_one_or_more conditions]
```

When you need to write for loops in Python, list comprehensions can make life a lot easier, as you can achieve this in a single line of code. However, comprehensions are not unique to lists. Dictionaries, which are common data structures used in data science, can also do comprehension.
If you have to do a practical test to demonstrate your knowledge and experience, it will be critical to remember that a Python list is defined with square brackets []. On the other hand, a dictionary will be represented by curly braces {}. Determining dict comprehension follows the same principle and is defined with a similar syntax, but it has to have a key:value pair in the expression.

Question 9: When Would You Use a List vs. a Tuple vs. a Set in Python?

A list is a common data type that is highly flexible. It can store a sequence of objects that are mutable, so it's ideal for projects that demand the storage of objects that can be changed later.
A tuple is similar to a list in Python, but the key difference between them is that tuples are immutable. They also use less space than lists and can only be used as a key in a dictionary. Tuples are a perfect choice when you want a list of constants.
Sets are a collection of unique elements that are used in Python. Sets are a good option when you want to avoid duplicate elements in your list. This means that whenever you have two lists with common elements between them, you can leverage sets to eliminate them.

Question 10: What's the Difference between a For Loop and a While Loop?

In Python, a loop iterates over popular data types (like dictionaries, lists, or strings) while the condition is true. This means that the program control will pass to the line immediately following the loop whenever the condition is false. In this scenario, it's not a question of preference, but a question of what your data structures are.

For Loop

In Python (and in almost any other programming language), For Loop is the most common type of loop. For Loop is often leveraged to iterate through the elements of an array.
For example:
For i=0, N_Elements (array) do...
For Loop can also be used to perform a fixed number of iterations and iterate by a given (positive or even negative) increment. It's important to note that by default, the increment will always be one.

While Loop

While Loop can be used in Python to perform an indefinite number of iterations as long as the condition remains true.
For example:
While (condition) do...
When using the While Loop, you have to explicitly specify a counter to keep track of how many times the loop was executed. However, While Loop can't define its own variable. Instead, it has to be previously defined and will continue to exist even after you exit the loop.
When compared to For Loop, While Loop is inefficient because it's much slower. This can be attributed to the fact that it checks the condition after each iteration. However, if you need to perform one or more conditional checks in a For Loop, you will want to consider using While Loop instead (as these checks won't be required).

Question 11: Write Output of the following given code

What does this code output:

```
def f(x,l=[]):
  for i in range(x):
    l.append(i*i)
  print(l)


f(2)
f(3,[3,2,1])
f(3)
```

Question 12: What Packages in the Standard Library, Useful for Data Science Work, Do You Know?

When Guido van Rossum created Python in the 1990s, it wasn't built for data science. Yet, today, Python is the leading language for machine learning, predictive analytics, statistics, and simple data analytics.
This is because Python is a free and open-source language that data professionals could easily use to develop tools that would help them complete data tasks more efficiently.
The following packages in the Python Standard Library are very handy for data science projects:
NumPy
NumPy (or Numerical Python) is one of the principal packages for data science applications. It's often used to process large multidimensional arrays, extensive collections of high-level mathematical functions, and matrices. Implementation methods also make it easy to conduct multiple operations with these objects.
There have been many improvements made over the last year that have resolved several bugs and compatibility issues.
NumPy is popular because it can be used as a highly efficient multi-dimensional container of generic data. It's also an excellent library as it makes data analysis simple by processing data faster while using a lot less code than lists.
Pandas
Pandas is a Python library that provides highly flexible and powerful tools and high-level data structures for analysis.
Pandas is an excellent tool for data analytics because it can translate highly complex operations with data into just one or two commands.
Pandas come with a variety of built-in methods for combining, filtering, and grouping data. It also boasts time-series functionality that is closely followed by remarkable speed indicators.
SciPy
SciPy is another outstanding library for scientific computing. It's based on NumPy and was created to extend its capabilities. Like NumPy, SciPy's data structure is also a multidimensional array that's implemented by NumPy.
The SciPy package contains powerful tools that help solve tasks related to integral calculus, linear algebra, probability theory, and much more.

Recently, this Python library went through some major build improvements in the form of continuous integration into multiple operating systems, methods, and new functions. Optimizers were also updated, and several new BLAS and LAPACK functions were wrapped.

Question 13: What does this stuff mean: *args, **kwargs? And why would we use it?

Use `*args` when we aren't sure how many arguments are going to be passed to a function, or if we want to pass a stored list or tuple of arguments to a function. `**kwargs` is used when we don't know how many keyword arguments will be passed to a function, or it can be used to pass the values of a dictionary as keyword arguments. The identifiers `args` and `kwargs` are a convention, you could also use `*bob` and `**billy` but that would not be wise. Here is a little illustration:

```
def f(*args,**kwargs): print(args, kwargs)
l = [1,2,3]
t = (4,5,6)
d = {'a':7,'b':8,'c':9}


f()
f(1,2,3)                    # (1, 2, 3) {}
f(1,2,3,"groovy")           # (1, 2, 3, 'groovy') {}
f(a=1,b=2,c=3)              # () {'a': 1, 'c': 3, 'b': 2}
f(a=1,b=2,c=3,zzz="hi")     # () {'a': 1, 'c': 3, 'b': 2, 'zzz': 'hi'}
f(1,2,3,a=1,b=2,c=3)        # (1, 2, 3) {'a': 1, 'c': 3, 'b': 2}
f(*l,**d)                   # (1, 2, 3) {'a': 7, 'c': 9, 'b': 8}
f(*t,**d)                   # (4, 5, 6) {'a': 7, 'c': 9, 'b': 8}
f(1,2,*t)                   # (1, 2, 4, 5, 6) {}
f(q="winning",**d)          # () {'a': 7, 'q': 'winning', 'c': 9, 'b': 8}
f(1,2,*t,q="winning",**d)   # (1, 2, 4, 5, 6) {'a': 7, 'q': 'winning', 'c': 9, 'b': 8}

def f2(arg1,arg2,*args,**kwargs): print(arg1,arg2, args, kwargs)
f2(1,2,3)                   # 1 2 (3,) {}
f2(1,2,3,"groovy")          # 1 2 (3, 'groovy') {}
f2(arg1=1,arg2=2,c=3)       # 1 2 () {'c': 3}
f2(arg1=1,arg2=2,c=3,zzz="hi")  # 1 2 () {'c': 3, 'zzz': 'hi'}
f2(1,2,3,a=1,b=2,c=3)       # 1 2 (3,) {'a': 1, 'c': 3, 'b': 2}


f2(*l,**d)                  # 1 2 (3,) {'a': 7, 'c': 9, 'b': 8}
f2(*t,**d)                  # 4 5 (6,) {'a': 7, 'c': 9, 'b': 8}
f2(1,2,*t)                  # 1 2 (4, 5, 6) {}
f2(1,1,q="winning",**d)     # 1 1 () {'a': 7, 'q': 'winning', 'c': 9, 'b': 8}
f2(1,2,*t,q="winning",**d)  # 1 2 (4, 5, 6) {'a': 7, 'q': 'winning', 'c': 9, 'b': 8}
```

Question 14: Consider the following code, what will it output?

```
class A(object):
    def go(self):
        print("go A go!")
    def stop(self):
        print("stop A stop!")
    def pause(self):
        raise Exception("Not Implemented")


class B(A):
    def go(self):
        super(B, self).go()
```

```
        print("go B go!")


class C(A):
    def go(self):
        super(C, self).go()
        print("go C go!")
    def stop(self):
        super(C, self).stop()
        print("stop C stop!")


class D(B,C):
    def go(self):
        super(D, self).go()
        print("go D go!")
    def stop(self):
 super(D, self).stop()
        print("stop D stop!")
    def pause(self):
        print("wait D wait!")


class E(B,C): pass


a = A()
b = B()
c = C()
d = D()
e = E()


# specify output from here onwards


a.go()
b.go()
c.go()
d.go()
e.go()


a.stop()
b.stop()
c.stop()
d.stop()
e.stop()


a.pause()
b.pause()
c.pause()
d.pause()
e.pause()
```

Answer

The output is specified in the comments in the segment below:

```
a.go()
# go A go!


b.go()
# go A go!
# go B go!


c.go()
# go A go!
# go C go!

d.go()
# go A go!
# go C go!
# go B go!
# go D go!


e.go()
# go A go!
# go C go!
# go B go!


a.stop()
# stop A stop!


b.stop()
# stop A stop!


c.stop()
# stop A stop!
# stop C stop!


d.stop()
# stop A stop!
# stop C stop!
# stop D stop!


e.stop()
# stop A stop!

a.pause()
# ... Exception: Not Implemented


b.pause()
# ... Exception: Not Implemented


c.pause()
# ... Exception: Not Implemented
```

```
d.pause()
# wait D wait!


e.pause()
# ...Exception: Not Implemented
```

Question 15: Consider the following code, what will it output?

```
class Node(object):
    def __init__(self,sName):
        self._lChildren = []
        self.sName = sName
    def __repr__(self):
        return "<Node '{}'>".format(self.sName)
    def append(self,*args,**kwargs):
        self._lChildren.append(*args,**kwargs)
    def print_all_1(self):
        print(self)
        for oChild in self._lChildren:
            oChild.print_all_1()
    def print_all_2(self):
        def gen(o):
            lAll = [o,]
            while lAll:
                oNext = lAll.pop(0)
                lAll.extend(oNext._lChildren)
                yield oNext
        for oNode in gen(self):
            print(oNode)


oRoot = Node("root")
oChild1 = Node("child1")
oChild2 = Node("child2")
oChild3 = Node("child3")
oChild4 = Node("child4")
oChild5 = Node("child5")
oChild6 = Node("child6")
oChild7 = Node("child7")
oChild8 = Node("child8")
oChild9 = Node("child9")
oChild10 = Node("child10")


oRoot.append(oChild1)
oRoot.append(oChild2)
oRoot.append(oChild3)
oChild1.append(oChild4)
oChild1.append(oChild5)
oChild2.append(oChild6)
oChild4.append(oChild7)
oChild3.append(oChild8)
oChild3.append(oChild9)
oChild6.append(oChild10)
```

```
# specify output from here onwards


oRoot.print_all_1()
oRoot.print_all_2()
```

Answer

`oRoot.print_all_1()` prints:

```
<Node 'root'>
<Node 'child1'>
<Node 'child4'>
<Node 'child7'>
<Node 'child5'>
<Node 'child2'>
<Node 'child6'>
<Node 'child10'>
<Node 'child3'>
<Node 'child8'>
<Node 'child9'>
```

`oRoot.print_all_2()` prints:

```
<Node 'root'>
<Node 'child1'>
<Node 'child2'>
<Node 'child3'>
<Node 'child4'>
<Node 'child5'>
<Node 'child6'>
<Node 'child8'>
<Node 'child9'>
<Node 'child7'>
<Node 'child10'>
```

Question 16: Describe Python's garbage collection mechanism in brief.

Answer

A lot can be said here. There are a few main points that you should mention:

Python maintains a count of the number of references to each object in memory. If a reference count goes to zero then the associated object is no longer live and the memory allocated to that object can be freed up for something else occasionally things called "reference cycles" happen. The garbage collector periodically looks for these and cleans them up. An example would be if you have two objects o1 and o2 such that `o1.x == o2` and `o2.x == o1`. If o1 and o2 are not referenced by anything else then they shouldn't be live. But each of them has a reference count of 1.

Certain heuristics are used to speed up garbage collection. For example, recently created objects are more likely to be dead. As objects are created, the garbage collector assigns them to generations. Each object gets one generation, and younger generations are dealt with first.

This explanation is CPython specific.

Question 16: Place the following functions below in order of their efficiency. They all take in a list of numbers between 0 and 1. The list can be quite long. An example input list would be `[random.random() for i in range(100000)]`. How would you prove that your answer is correct?

```
def f1(lIn):
    l1 = sorted(lIn)
    l2 = [i for i in l1 if i<0.5]
    return [i*i for i in l2]
```

```
def f2(lIn):
    l1 = [i for i in lIn if i<0.5]
    l2 = sorted(l1)
    return [i*i for i in l2]


def f3(lIn):
    l1 = [i*i for i in lIn]
    l2 = sorted(l1)
    return [i for i in l1 if i<(0.5*0.5)]
```

Most to least efficient: f2, f1, f3. To prove that this is the case, you would want to profile your code. Python has a lovely profiling package that should do the trick.

```
import cProfile
lIn = [random.random() for i in range(100000)]
cProfile.run('f1(lIn)')
cProfile.run('f2(lIn)')
cProfile.run('f3(lIn)')
```

For completion's sake, here is what the above profile outputs:

```
>>> cProfile.run('f1(lIn)')
         4 function calls in 0.045 seconds


   Ordered by: standard name
   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
        1    0.009    0.009    0.044    0.044 <stdin>:1(f1)
        1    0.001    0.001    0.045    0.045 <string>:1(<module>)
        1    0.000    0.000    0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}
        1    0.035    0.035    0.035    0.035 {sorted}


>>> cProfile.run('f2(lIn)')
         4 function calls in 0.024 seconds


   Ordered by: standard name
   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
        1    0.008    0.008    0.023    0.023 <stdin>:1(f2)
        1    0.001    0.001    0.024    0.024 <string>:1(<module>)
        1    0.000    0.000 0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}
        1    0.016    0.016    0.016    0.016 {sorted}


>>> cProfile.run('f3(lIn)')
         4 function calls in 0.055 seconds


 Ordered by: standard name
   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
        1    0.016    0.016    0.054    0.054 <stdin>:1(f3)
        1    0.001    0.001    0.055    0.055 <string>:1(<module>)
        1    0.000    0.000  0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}
        1    0.038    0.038    0.038    0.038 {sorted}
```

Question 17: In Python, How is Memory Managed?
In Python, memory is managed in private heap space. This means that all the objects and data structures will be located in a private heap. However, the programmer won't be allowed to access this heap. Instead, the Python interpreter will handle it. At the same time, the core API will enable access to some Python tools for the programmer to start coding. The memory manager will allocate the heap space for the Python objects while the inbuilt garbage collector will recycle all the memory that's not being used to boost available heap space.

Question 18: What is the purpose of the PYTHONSTARTUP environment variable?
PYTHONCASEOK – It is used in Windows to instruct Python to find the first case-insensitive match in an import statement. Set this variable to any value to activate it.

Question 19: Is python a case sensitive language?
Yes! Python is a case sensitive programming language.

Question 20: What are the supported data types in Python?
Python has five standard data types –
Numbers
String
List
Tuple
Dictionary

Question 21: What is the output of print str if str = 'Hello World!'?
It will print a complete string. The output would be Hello World!.

Question 22: What is the output of print str[0] if str = 'Hello World!'?
It will print the first character of the string. The output would be H.

Question 23: What is the output of print str[2:5] if str = 'Hello World!'?
It will print characters starting from 3rd to 5th. The output would be llo.

Question 24: What is the output of print str[2:] if str = 'Hello World!'?
It will print characters starting from 3rd character. The output would be llo World!.

Question 25: What is the output of print str * 2 if str = 'Hello World!'?
It will print the string two times. The output would be Hello World! Hello World!.

Question 26: What is the output of print str + "TEST" if str = 'Hello World!'?
It will print a concatenated string. The output would be Hello World!TEST.

Question 27: What is the output of print list if list = ['abcd', 786 , 2.23, 'john', 70.2]?
It will print complete list. Output would be ['abcd', 786, 2.23, 'john', 70.200000000000003].

Question 28: What is the output of print list[0] if list = ['abcd', 786 , 2.23, 'john', 70.2]?
It will print first element of the list. Output would be abcd.

Question 29: What is the output of print list[1:3] if list = ['abcd', 786 , 2.23, 'john', 70.2]?

It will print elements starting from 2nd till 3rd. Output would be [786, 2.23].

Question 30: What is the output of print list[2:] if list = ['abcd', 786 , 2.23, 'john', 70.2]?
It will print elements starting from 3rd element. Output would be [2.23, 'john', 70.200000000000003].

Question 31: What is the output of print tinylist * 2 if tinylist = [123, 'john']?
It will print list two times. Output would be [123, 'john', 123, 'john'].

Question 32: What is the output of print list1 + list2, if list1 = ['abcd', 786 , 2.23, 'john', 70.2] and ist2 = [123, 'john']?
It will print concatenated lists. Output would be ['abcd', 786, 2.23, 'john', 70.2, 123, 'john']

Question 33: What are tuples in Python?
A tuple is another sequence data type that is similar to the list. A tuple consists of a number of values separated by commas. Unlike lists, however, tuples are enclosed within parentheses.

Question 34: What is the difference between tuples and lists in Python?
The main differences between lists and tuples are – Lists are enclosed in brackets ([ ]) and their elements and size can be changed, while tuples are enclosed in parentheses (( )) and cannot be updated. Tuples can be thought of as read-only lists.

Question 35: What is the output of print tuple if tuple = ('abcd', 786 , 2.23, 'john', 70.2)?
It will print complete tuple. Output would be ('abcd', 786, 2.23, 'john', 70.200000000000003).

Question 36: What is the output of print tuple[0] if tuple = ('abcd', 786 , 2.23, 'john', 70.2)?
It will print the first element of the tuple. The output would be abcd.

Question 37: What is the output of print tuple[1:3] if tuple = ('abcd', 786 , 2.23, 'john', 70.2)?
It will print elements starting from 2nd till 3rd. The output would be (786, 2.23).

Question 38: What is the output of print tuple[2:] if tuple = ('abcd', 786 , 2.23, 'john', 70.2)?
It will print elements starting from the 3rd element. The output would be (2.23, 'john', 70.200000000000003).

Question 39: What is the output of print tinytuple * 2 if tinytuple = (123, 'john')?
It will print tuple two times. The output would be (123, 'john', 123, 'john').

Question 40: What is the output of print tuple + tinytuple if tuple = ('abcd', 786 , 2.23, 'john', 70.2) and tinytuple = (123, 'john')?
It will print concatenated tuples. Output would be ('abcd', 786, 2.23, 'john', 70.200000000000003, 123, 'john').

Question 41: What are Python's dictionaries?
Python's dictionaries are kind of hash table type. They work like associative arrays or hashes found in Perl and consist of key-value pairs. A dictionary key can be almost any Python type, but are usually numbers or strings. Values, on the other hand, can be any arbitrary Python object.

Question 42: How will you create a dictionary in python?

Dictionaries are enclosed by curly braces ({ }) and values can be assigned and accessed using square braces ([]).

```
dict = {}
dict['one'] = "This is one"
dict[2]     = "This is two"
tinydict = {'name': 'john','code':6734, 'dept': 'sales'}
```

Question 43: How will you get all the keys from the dictionary?

Using dictionary.keys() function, we can get all the keys from the dictionary object.

```
print dict.keys()   # Prints all the keys
```

Question 44: How will you get all the values from the dictionary?

Using dictionary.values() function, we can get all the values from the dictionary object.

```
print dict.values()   # Prints all the values
```

Question 45: How will you convert a string to an int in python?

int(x [,base]) - Converts x to an integer. base specifies the base if x is a string.

Question 46: How will you convert a string to a long in python?

long(x [,base]) - Converts x to a long integer. base specifies the base if x is a string.

Question 47: How will you convert a string to a float in python?

float(x) – Converts x to a floating-point number.

Question 48: How will you convert an object to a string in python?

str(x) – Converts object x to a string representation.

Question 49: How will you convert an object to a regular expression in python?

repr(x) – Converts object x to an expression string.

Question 50: How will you convert a string to an object in python?

eval(str) – Evaluates a string and returns an object.

Question 51: How will you convert a string to a tuple in python?

tuple(s) – Converts s to a tuple.

Question 52: How will you convert a string to a list in python?

list(s) – Converts s to a list.

Question 53: How will you convert a string to a set in python?

set(s) – Converts s to a set.

Question 54: How will you create a dictionary using tuples in python?

dict(d) – Creates a dictionary. d must be a sequence of (key, value) tuples.

Question 55: How will you convert a string to a frozen set in python?
frozenset(s) – Converts s to a frozen set.

Question 56: How will you convert an integer to a character in python?
chr(x) – Converts an integer to a character.

Question 57: How will you convert an integer to an Unicode character in python?
unichr(x) – Converts an integer to a Unicode character.

Question 58: How will you convert a single character to its integer value in python?
ord(x) – Converts a single character to its integer value.

Question 59: How will you convert an integer to a hexadecimal string in python?
hex(x) – Converts an integer to a hexadecimal string.

Question 60: How will you convert an integer to an octal string in python?
oct(x) – Converts an integer to an octal string.

Question 61: What is the purpose of ** operator?
** Exponent – Performs exponential (power) calculation on operators. a**b = 10 to the power 20 if a = 10 and b = 20.

Question 62: What is the purpose of // operator?
// Floor Division – The division of operands where the result is the quotient in which the digits after the decimal point are removed.

Question 63: What is the purpose of is the operator?
is – Evaluates to true if the variables on either side of the operator point to the same object and false otherwise. x is y, here is results in 1 if id(x) equals id(y).

Question 64: What is the purpose of not in the operator?
not in – Evaluates to true if it does not finds a variable in the specified sequence and false otherwise. x not in y, here not in results in a 1 if x is not a member of sequence y.

Question 65: What is the purpose break statement in python?
break statement – Terminates the loop statement and transfers execution to the statement immediately following the loop.

Question 66: What is the purpose of continue statement in python?
continue statement – Causes the loop to skip the remainder of its body and immediately retest its condition prior to reiterating.

Question 67: What is the purpose pass statement in python?

pass statement – The pass statement in Python is used when a statement is required syntactically but you do not want any command or code to execute.

Question 68: How can you pick a random item from a list or tuple?
choice(seq) – Returns a random item from a list, tuple, or string.

Question 69: How can you pick a random item from a range?
randrange ([start,] stop [,step]) – returns a randomly selected element from range(start, stop, step).

Question 70: How can you get a random number in python?
random() – returns a random float r, such that 0 is less than or equal to r and r is less than 1.

Question 71: How will you set the starting value in generating random numbers?
seed([x]) – Sets the integer starting value used in generating random numbers. Call this function before calling any other random module function. Returns None.

Question 72: How will you randomize the items of a list in place?
shuffle(lst) – Randomizes the items of a list in place. Returns None.

Question 73: How will you capitalize first letter of string?
capitalize() – Capitalizes first letter of string.

Question 74: How will you check in a string that all characters are alphanumeric?
isalnum() – Returns true if the string has at least 1 character and all characters are alphanumeric and false otherwise.

Question 75: How will you check in a string that all characters are digits?
isdigit() – Returns true if the string contains only digits and false otherwise.

Question 76: How will you check in a string that all characters are in lowercase?
islower() – Returns true if the string has at least 1 cased character and all cased characters are in lowercase and false otherwise.

Question 77: How will you check in a string that all characters are numerics?
isnumeric() – Returns true if a Unicode string contains only numeric characters and false otherwise.

Question 78: How will you check in a string that all characters are whitespaces?
isspace() – Returns true if the string contains only whitespace characters and false otherwise.

Question 79: How will you check in a string that it is properly titlecased?
istitle() – Returns true if the string is properly "titlecased" and false otherwise.

Question 80: How will you check in a string that all characters are in uppercase?
isupper() – Returns true if the string has at least one cased character and all cased characters are in uppercase and false otherwise.

Question 81: How will you merge elements in a sequence?
join(seq) – Merges (concatenates) the string representations of elements in sequence seq into a string, with separator string.

Question 82: How will you get the length of the string?
len(string) – Returns the length of the string.

Question 83: How will you get a space-padded string with the original string left-justified to a total of width columns?
ljust(width[, fillchar]) – Returns a space-padded string with the original string left-justified to a total of width columns.

Question 84: How will you convert a string to all lowercase?
lower() – Converts all uppercase letters in a string to lowercase.

Question 85: How will you remove all leading whitespace in a string?
lstrip() – Removes all leading whitespace in string.

Question 86: How will you get the max alphabetical character from the string?
max(str) – Returns the max alphabetical character from the string str.

Question 87: How will you get the min alphabetical character from the string?
min(str) – Returns the min alphabetical character from the string str.

Question 88: How will you replace all occurrences of an old substring in a string with new string?
replace(old, new [, max]) – Replaces all occurrences of old in a string with new or at most max occurrences if max gave.

Question 89: How will you remove all leading and trailing whitespace in a string?
strip([chars]) – Performs both lstrip() and rstrip() on string.

Question 90: How will you change the case for all letters in a string?
swapcase() – Inverts case for all letters in the string.

Question 91: How will you get title cased version of string?
title() – Returns "titlecased" version of the string, that is, all words begin with uppercase and the rest are lowercase.

Question 92: How will you convert a string to all uppercase?
upper() – Converts all lowercase letters in a string to uppercase.

Question 93: How will you check in a string that all characters are decimal?
isdecimal() – Returns true if a Unicode string contains only decimal characters and false otherwise.

Question 94: What is the difference between del() and remove() methods of list?

To remove a list element, you can use either the del statement if you know exactly which element(s) you are deleting or the remove() method if you do not know.

Question 95: What is the output of len([1, 2, 3])?
3.

Question 96: What is the output of [1, 2, 3] + [4, 5, 6]?
[1, 2, 3, 4, 5, 6]

Question 97: What is the output of ['Hi!'] * 4?
['Hi!', 'Hi!', 'Hi!', 'Hi!']

Question 98: What is the output of 3 in [1, 2, 3]?
True

Question 99: What is the output of x in [1, 2, 3]: print x?
1
2
3

Question 100: What is the output of L[2] if L = [1,2,3]?
3, Offsets start at zero.

Question 101: What is the output of L[-2] if L = [1,2,3]?
1, Negative: count from the right.

Question 102: What is the output of L[1:] if L = [1,2,3]?
2, 3, Slicing fetches sections.

Question 103: How will you compare two lists?
cmp(list1, list2) – Compares elements of both lists.

Question 104: How will you get the length of a list?
len(list) – Gives the total length of the list.

Question 105: How will you get the max valued item of a list?
max(list) – Returns item from the list with max value.

Question 106: How will you get the min valued item of a list?
min(list) – Returns item from the list with min value.

Question 107: How will you get the index of an object in a list?
list.index(obj) – Returns the lowest index in the list that obj appears.

Question 108: How will you insert an object at a given index in a list?
list.insert(index, obj) – Inserts object obj into list at offset index.

Question 109: How will you remove the last object from a list?
list.pop(obj=list[-1]) – Removes and returns last object or obj from list.

Question 110: How will you remove an object from a list?
list.remove(obj) – Removes object obj from list.

Question 111: How will you reverse a list?
list.reverse() – Reverses objects of the list in place.

Question 112: How will you sort a list?
list.sort([func]) – Sorts objects of list, use compare func if given.

Question 113: What is lambda function in python?
'lambda' is a keyword in python which creates an anonymous function. Lambda does not contain block of statements. It does not contain return statements.

Question 114: What we call a function that is an incomplete version of a function?
Stub.

Question 115: When a function is defined then the system stores parameters and local variables in an area of memory. What this memory is known as?
Stack.

Question 116: A canvas can have a foreground color? (Yes/No)
Yes.

Question 117: Is the Python platform independent?
No
There are some modules and functions in python that can only run on certain platforms.

Question 118: Do you think Python has a compiler?
Yes
Yes, it has a compiler that works automatically so we don't notice the compiler of python.

Question 119: What are the applications of Python?
Django (web framework of Python).
2. Micro Framework such as Flask and Bottle.
3. Plone and Django CMS for advanced content Management.

Question 120: What is the basic difference between Python version 2 and Python version 3?
Table below explains the difference between Python version 2 and Python version 3.

S.No	Section	Python Version2	Python Version3
1.	Print Function	The print command can be used without parentheses.	Python 3 needs parentheses to print any string. It will raise errors without parentheses.

2.	Unicode	ASCII str() types and separate Unicode() but there is no byte type code in Python 2.	Unicode (utf-8) and it has two-byte classes – Byte bytearray S.
3.	Exceptions	Python 2 accepts both new and old notations of syntax.	Python 3 raises a SyntaxError in turn when we don't enclose the exception argument in parentheses.
4.	Comparing Unorderable	It does not raise any error.	It raises 'TypeError' as a warning if we try to compare unorderable types.

Question 121: Which programming language is an implementation of Python programming language designed to run on Java Platform?

Jython

(Jython is the successor of Jpython.)

Question 122: Is there any double data type in Python?

No

Question 123: Is String in Python are immutable? (Yes/No)

Yes.

Question 124: Can True = False be possible in Python?

No.

Question 125: Which module of python is used to apply the methods related to OS.?

OS.

Question 126: When does a new block begin in python?

A block begins when the line is intended by 4 spaces.

Question 127: Write a function in python that detects whether the given two strings are anagrams or not.

```
def check(a,b):
  if(len(a)!=len(b)):
    return False
  else:
   if(sorted(list(a)) == sorted(list(b))):
      return True
  else:
    return False
```

Question 128: Name the python Library used for Machine learning.

Scikit-learn python Library used for Machine learning

Question 129: What does pass operation do?

Pass indicates that nothing is to be done i.e. it signifies a no operation.

Question 130: Name the tools that python uses to find bugs (if any).
Pylint and pychecker.

Question 131: Write a function to give the sum of all the numbers in the list?
Sample list – (100, 200, 300, 400, 0, 500)
Expected output – 1500

Question 132: Program for the sum of all the numbers in the list is –

```
def sum(numbers):
  total = 0
  for num in numbers:
    total+=num
  print("Sum of the numbers: ", total)
sum((100, 200, 300, 400, 0, 500))
```

We define a function 'sum' with numbers as a parameter. The in for loop we store the sum of all the values of the list.

Question 133: Write a program in Python to reverse a string without using the inbuilt function reverse string?
Program to reverse a string is given below –

```
def string_reverse(str1):

    rev_str = ''
    index = len(str1) #defining index as length of string.
    while(index>0):
    rev_str = rev_str + str1[index-1]
    index = index-1
    return(rev_str)


print(string_reverse('1tniop'))
```

First, we declare a variable to store the reversed string. Then using while loop and indexing of string (index is calculated by string length) we reverse the string. While loop starts when the index is greater than zero. The index is reduced to value 1 each time. When the index reaches zero we obtain the reverse of a string.

Question 134: Write a program to test whether the number is in the defined range or not?
Program is –

```
def test_range(num):
  if num in range(0, 101):
    print("%s is in range"%str(num))
  else:
    print("%s is not in range"%str(num))
```

Output –

```
test_range(101)
```

101 is not in the range
To test any number in a particular range we make use of the method 'if..in' and else condition.

Python Coding Challenges
Python Coding Challenges
Program 1: Predict the output of the following Python Programs.

```
class Acc:
   def __init__(self, id):
      self.id = id
      id = 555

acc = Acc(111)
print(acc.id)
```

Explanation: Instantiation of the class "Acc" automatically calls the method __init__ and passes the object as the self parameter. 111 is assigned to data attribute of the object called id.
The value "555" is not retained in the object as it is not assigned to a data attribute of the class/object. So, the output of the program is "111"

Program 2: Predict the output of the following Python Programs.

```
for i in  range(2):
   print (i)

for i in range(4,6):
   print(i)
```

Explanation: If only single argument is passed to the range method, Python considers this argument as the end of the range and the default start value of range is 0. So, it will print all the numbers starting from 0 and before the supplied argument.
For the second for loop the starting value is explicitly supplied as 4 and ending is 5.

Program 3: Predict the output of following Python Programs

```
values = [1, 2, 3, 4]
numbers = set(values)

def checknums(num):
   if num in numbers:
      return True
   else:
      return False

for i infilter(checknums, values):
   print(i)
```

Explanation: The function "filter" will return all items from list values that return True when passed to the function "check it". "check it" will check if the value is in the set. Since all the numbers in the set come from the values list, all of the original values in the list will return True.

Program 4: Predict the output of the following Python Programs

```
counter = {}

def addToCounter(country):
   if country in  counter:
counter[country] += 1
   else:
counter[country] = 1

addToCounter('China')
addToCounter('Japan')
addToCounter('china')

print len(counter)
```

Explanation: The task of "len" function is to return number of keys in a dictionary. Here 3 keys are added to the dictionary "country" using the "add to counter" function.
Please note carefully – The keys to a dictionary are case sensitive.

Program 5: Predict the output of following Python Programs

```
def dtFunction():
   "DataTrained is cool website for boosting up technical skills"
   return 1

print (dtFunction.__doc__[15:19] )
```

Explanation:
There is a docstring defined for this method, by putting a string on the first line after the start of the function definition. The docstring can be referenced using the __doc__ attribute of the function.
And hence it prints the indexed string.

Program 6: Predict the output of the following Python Programs

```
class A(object):
   val = 1

class B(A):
   pass

class C(A):
   pass

print(A.val, B.val, C.val)
B.val = 2
print(A.val, B.val, C.val)
A.val = 3
```

```
print(A.val, B.val, C.val)
```

Explanation:
In Python, class variables are internally handled as dictionaries. If a variable name is not found in the dictionary of the current class, the class hierarchy (i.e., its parent classes) are searched until the referenced variable name is found, if the variable is not found error is being thrown.
So, in the above program the first call to print() prints the initialized value i.e, 1.
In the second call since B. val is set to 2, the output is 1 2 1.
The last output 3 2 3 may be surprising. Instead of 3 3 3, here B.val reflects 2 instead of 3 since it is overridden earlier.

Program 7: Predict the output of the following Python Programs

```
check1 = ['Learn', 'Quiz', 'Practice', 'Contribute']
check2 = check1
check3 = check1[:]

check2[0] = 'Code'
check3[1] = 'Mcq'

count = 0
for c in (check1, check2, check3):
    if c[0] == 'Code':
        count += 1
    if c[1] == 'Mcq':
        count += 10

print(count)
```

Explanation:
When assigning check1 to check2, we create a second reference to the same list. Changes to check2 affects check1. When assigning the slice of all elements in check1 to check3, we are creating a full copy of check1 which can be modified independently (i.e, any change in check3 will not affect check1).
So, while checking check1 'Code' gets matched and count increases to 1, but Mcq doest gets matched since its available only in check3.
Now checking check2 here also 'Code' gets matched resulting in count value to 2.
Finally while checking check3 which is separate from both check1 and check2 here only Mcq gets matched and count becomes 12.

Program 8: Predict the output of the following Python Programs

```
def dataTr(x,l=[]):
    for i in range(x):
        l.append(i*i)
```

```
    print(l)

dataTr(2)
dataTr(3,[3,2,1])
dataTr(3)
```

Explanation:
The first function call should be fairly obvious, the loop appends 0 and then 1 to the empty list, l. l is a name for a variable that points to a list stored in memory. The second call starts off by creating a new list in a new block of memory. l then refers to this new list. It then appends 0, 1 and 4 to this new list. So that's great. The third function call is the weird one. It uses the original list stored in the original memory block. That is why it starts off with 0 and 1.

Program 9: Which of the options below could possibly be the output of the following program?

```
from random import randrange
L = list()
for x in range(5):
L.append(randrange(0, 100, 2)-10)

# Choose which of outputs below are valid for this code.
print(L)
```

a) [-8, 88, 8, 58, 0]
b) [-8, 81, 18, 46, 0]
c) [-7, 88, 8, 58, 0]
d) [-8, 88, 94, 58, 0]

Ans. (a)
Explanation: The for loop will result in appending 5 elements to list L. Range of the elements lies from [0, 98] – 10 = [-10, 88], which rules out option (d). The upper range is 98 because the step size is 2, thus option (c) and (b) are invalid. Also, note that each time you may not get the same output or the one in the options as the function is random.

Program 10: What is the output of the following program?

```
from math import *
a = 2.13
b = 3.7777
c = -3.12
print(int(a), floor(b), ceil(c), fabs(c))
```

a) 2 3 -4 3
b) 2 3 -3 3.12
c) 2 4 -3 3
d) 2 3 -4 3.12

Ans. (b)
Explanation: int() returns the integer value of a number, int(2.13) = 2. floor() returns the largest integer lesser or equal to the number, floor(3.777) = 3. ceil() returns smallest integer greater or equal to the number, ceil(-3.12) = -3. fabs() return the modulus of the number, thus fabs(-3.12) = 3.12.

Program 11: What is the output of the following program?

```
import re
p = re.compile('\d+')
print(p.findall("I met him once at 11 A.M. on 4th July 1886"), end = " ")
p = re.compile('\d')
print(p.findall("I went to him at 11 A.M."))
```

a) ['11', '4', '1886', '11']
b) ['1141886'] ['1', '1']
c) ['11', '4', '1886'] ['11']
d) ['11', '4', '1886'] ['1', '1']

Ans. (d)
Explanation: \d is equivalent to [0-9] and \d+ will match a group on [0-9], group of one or greater size. In first statement, group of digits are 11, 4, 1886. In the second statement, \d will treat each each digit as different entity, thus 1, 1.

Program 12: What is the output of the following program?

```
import re
print(re.sub('a', '**', 'DataTrainers', flags = re.IGNORECASE), end = " ")
print(re.sub('tr', '**', 'DataTrainers'))
```

a) D**t**Tr**iners Data**ainers
b) D*t*Tr*iners Data**ainers
c) D*t*Tr*iners Data*ainers
d) TypeError: 'str' object does not support item assignment
Ans. (a)
Explanation: In the first print statement, all 'a' will be replaced '**', and case is igonred. Case is not igonred in 2nd statement, thus 'ge' will be replaced but not 'Ge'.

Program 13: Which of the options below could possibly be the output of the following program?

```
import math
import random
```

```
L = [1, 2, 30000000000000]
for x in range(3):
    L[x] = math.sqrt(L[x])

# random.choices() is available on Python 3.6.1 only.
string = random.choices(["apple", "carrot", "pineapple"], L, k = 1)
print(string)
```

a) [‘pineapple’]
b) [‘apple’]
c) ‘pineapple’
d) both a and b

Ans. (d)
Explanation: Two modules math and random are used, L after the for loop will be [1.0, 1.4142135623730951, 5477225.575051662]. choices() has a choice as 1st parameter and their weights as the second parameter, k is the number valued needed from choice. The answer will come out to ‘pineapple’ almost always due to its the weight but ‘apple’ and ‘carrot’ may turn out to be the output at times.

Program 14: What is the output of the following program?

```
D = {1 : 1, 2 : '2', '1' : 1, '2' : 3}
D['1'] = 2
print(D[D[D[str(D[1])]]])
```

a) 2
b) 3
c) ‘2’
d) KeyError

Ans. (b)
Explanation: Simple key-value pair is used recursively, D[1] = 1, str(1) = ‘1’. So, D[str(D[1])] = D[‘1’] = 2, D[2] = ‘2’ and D[‘2’] = 3.

Program 15: What is the output of the following program?

```
D = dict()
for x in enumerate(range(2)):
    D[x[0]] = x[1]
    D[x[1]+7] = x[0]
print(D)
```

a) KeyError
b) {0: 1, 7: 0, 1: 1, 8: 0}

c) {0: 0, 7: 0, 1: 1, 8: 1}
d) {1: 1, 7: 2, 0: 1, 8: 1}

Ans. (c)
Explanation: enumerate() will return a tuple, the loop will have x = (0, 0), (1, 1). Thus D[0] = 0, D[1] = 1, D[0 + 7] = D[7] = 0 and D[1 + 7] = D[8] = 1.
Note: Dictionary is unordered, so the sequence of the key-value pair may differ in each output.

Program 16: Which of the options below could possibly be the output of the following program?

```
D = {1 : [1, 2, 3], 2: (4, 6, 8)}
D[1].append(4)
print(D[1], end = " ")
L  = list(D[2])
L.append(10)
D[2] = tuple(L)
print(D[2])
```

a) [1, 2, 3, 4] [4, 6, 8, 10]
b) [1, 2, 3] (4, 6, 8)
c) '[1, 2, 3, 4] TypeError: tuples are immutable
d) [1, 2, 3, 4] (4, 6, 8, 10)
Ans. (d)
Explanation: In the first part key-value indexing is used and 4 is appended into the list. As tuples are immutable, in the second part the tuple is converted into a list, valued 10 is added and then converted back to list.

Program 17: What is the output of the following program?

```
D = dict()
for i in range (3):
   for j in range(2):
      D[i] = j
print(D)
```

a) {0: 0, 1: 0, 2: 0}
b) {0: 1, 1: 1, 2: 1}
c) {0: 0, 1: 0, 2: 0, 0: 1, 1: 1, 2: 1}
d) TypeError: Immutable object

Ans. (b)
Explanation: 1st loop will give 3 values to i 0, 1 and 2. In the empty dictionary, valued are added and overwrited in j loop, for eg. D[0] = [0] becomes D[0] = 1, due to overwriting.

Program 18: What is the output of the following program?

```
data = [2, 3, 9]
temp = [[x for x in[data]] for x in range(3)]
print (temp)
```

a) [[[2, 3, 9]], [[2, 3, 9]], [[2, 3, 9]]]
b) [[2, 3, 9], [2, 3, 9], [2, 3, 9]]
c) [[[2, 3, 9]], [[2, 3, 9]]]
d) None of these

Ans. (a)
Explanation: [x for x in[data] returns a new list copying the values in the list data and the outer for statement prints the newly created list 3 times.

Program 19: What is the output of the following program?

```
data = [x for x in range(5)]
temp = [x for x in range(7) if x in data and x%2==0]
print(temp)
```

a) [0, 2, 4, 6]
b) [0, 2, 4]
c) [0, 1, 2, 3, 4, 5]
d) Runtime error

Ans. (b)
Explanation: The is statement checks whether the value lies in list data and if it does whether it's divisible by 2. It does so for x in (0, 7).

Program 19: What is the output of the following program?

```
temp = ['Data', 'for', 'Peoples']
arr = [i[0].upper() for i in temp]
print(arr)
```

a) ['D', 'F', 'P']
b) ['DATA']
c) ['DATA', 'FOR', 'PEOPLES']
d) Compilation error

Ans. (a)
Explanation: The variable i is used to iterating over each element in list temp. i[0] represent the character at 0th index of i and .upper() function is used to capitalize the character present at i[0].

Program 20: Whats is the output of the following program?

```
temp = 'Treat 22536 for 445 Geeks'
data = [x for x in (int(x) for x in temp if x.isdigit()) if x%2 == 0]
print(data)
```

a) [2, 2, 6, 4, 4]
b) Compilation error
c) Runtime error
d) ['2', '2', '5', '3', '6', '4', '4', '5']

Ans. (a)
Explanation: This is an example of nested list comprehension. The inner list created contains a list of integers in temp. The outer list only procures that x which are a multiple of 2.

Program 21: What is the output of the following program?

```
data = [x for x in (x for x in 'Geeks 22966 for Geeks' if x.isdigit()) if
(x in ([x for x in range(20)]))]
print(data)
```

a) [2, 2, 9, 6, 6]
b) []
c) Compilation error
d) Runtime error

Ans. (b)
Explanation: Since here x have not been converted to int, the condition in the if statement fails and therefore, the list remains empty.

What is the output of the following?
Program 22:

```
i = 1
while True:
    if i % 0O7 == 0:
        break
    print(i)
    i += 1
```

1. 1 2 3 4 5 6.
2. 1 2 3 4 5 6 7.
3. error.
4. None of these

Ans. (a)
Explanation: The loop will terminate when i will be equal to 7.

Ans. (c)
Explanation: 'Were' is at Index 15 in Line1, find() returns the index of substring if found in the string Line1. count() returns the total number of occurrences of the substring. Line4 is concatenated string from Line1, Line2 and Line3. This code works well with Python v2.x, as some string functions are deprecated in Python v3.x.

Program 25: What is the output of the following program?

```
line = "I'll come by then."
eline = ""
for i in line:
   eline += chr(ord(i)+3)
print(eline)
```

a) L*oo frph e| wkhq1
b) L*oo#frph#e|#wkhq1
c) l*oo@frph@e|$wkhq1
d) O*oo#Frph#E|#wKhq1
Ans. (b)
Explanation: This piece of code ciphers the plain text. Each character is moved to its 3rd next character by increasing the ASCII value. 'I' becomes 'L', thus option (c) and (d) are ruled out. ' ' has ASCII value of 32, thus it'll become 35('#'), thus option (a) is ruled out as, ' ' cannot remain to be ' ' in the ciphered text.

SQL Interview Questions
SQL Interview Questions
Question 1: What is the purpose of the group functions in SQL? Give some examples of group functions.
Group functions are necessary to get summary statistics of a data set. COUNT, MAX, MIN, AVG, SUM, and DISTINCT are all group functions.

Question 2: Tell me the difference between an inner join, left join/right join, and union.
"In a Venn diagram the inner join is when both tables have a match, a left join is when there is a match in the left table and the right table is null, a right join is the opposite of a left join, and a full join is all of the data combined."

Question 3: What does UNION do? What is the difference between UNION and UNION ALL?
"UNION removes duplicate records (where all columns in the results are the same), UNION ALL does not."

Question 4: What is the difference between SQL and MySQL or SQL Server?
"SQL stands for Structured Query Language. It's a standard language for accessing and manipulating databases. MySQL is a database management system, like SQL Server, Oracle, Informix, Postgres, etc."

Question 5: If a table contains duplicate rows, does a query result display the duplicate values by default? How can you eliminate duplicate rows from a query result?
Yes. One way you can eliminate duplicate rows with the DISTINCT clause.

Question 6: What are the different types of keys in a relational database?
There are a variety of keys in a relational database, including:
Alternate keys are candidate keys that exclude all primary keys.

Artificial keys are created by assigning a unique number to each occurrence or record when there aren't any compound or standalone keys.
Compound keys are made by combining multiple elements to develop a unique identifier for a construct when there isn't a single data element that uniquely identifies occurrences within a construct. Also known as a composite key or a concatenated key, compound keys consist of two or more attributes.
Foreign keys are groups of fields in a database record that point to a key field or a group of fields that create a key of another database record that's usually in a different table. Often, foreign keys in one table refer to primary keys in another. As the referenced data can be linked together quite quickly, it can be critical to database normalization.
Natural keys are data elements that are stored within constructs and utilized as primary keys.
Primary keys are values that can be used to identify unique rows in a table and the attributes associated with them. For example, these can take the form of a Social Security number that's related to a specific person. In a relational model of data, the primary key is the candidate key. It's also the primary method used to identify a tuple in each possible relation.
Super keys are defined in the relational model as a set of attributes of a relation variable. It holds that all relations assigned to that variable don't have any distinct tuples. They also don't have the same values for the attributes in the set. Super keys also are defined as a set of attributes of a relational variable upon which all of the functionality depends.

Question 7: What is the difference between SQL and MySQL or SQL Server?
SQL or Structured Query Language is a language; language that communicates with a relational database thus providing ways of manipulating and creating databases. MySQL and Microsoft's SQL Server both are relational database management systems that use SQL as their standard relational database language.

Question 8: What is the difference between SQL and PL/SQL?
PL/SQL is a dialect of SQL that adds procedural features of programming languages in SQL. It was developed by Oracle Corporation in the early 90's to enhance the capabilities of SQL.

Question 9: What are various DDL commands in SQL? Give brief description of their purposes.
Following are various DDL or Data Definition Language commands in SQL –
CREATE – it creates a new table, a view of a table, or other object in database.
ALTER – it modifies an existing database object, such as a table.
DROP – it deletes an entire table, a view of a table or other object in the database.

Question 10: What are various DML commands in SQL? Give brief description of their purposes.
Following are various DML or Data Manipulation Language commands in SQL –
SELECT – it retrieves certain records from one or more tables.
INSERT – it creates a record.
UPDATE – it modifies records.
DELETE – it deletes records.

Question 11: What are various DCL commands in SQL? Give a brief description of their purposes.
Following are various DCL or Data Control Language commands in SQL –
GRANT – it gives a privilege to the user.
REVOKE – it takes back privileges granted from the user.

Question 12: Can you sort a column using a column alias?
Yes. A column alias could be used in the ORDER BY clause.

Question 13: Is a NULL value the same as zero or a blank space? If not then what is the difference?
A NULL value is not the same as zero or a blank space. A NULL value is a value that is 'unavailable, unassigned, unknown or not applicable'. Whereas, zero is a number and blank space is a character.

Question 14: Say True or False. Give an explanation if False.
If a column value taking part in an arithmetic expression is NULL, then the result obtained would be NULLM.
True.

Question 15: If a table contains duplicate rows, does a query result display the duplicate values by default? How can you eliminate duplicate rows from a query result?
A query result displays all rows including the duplicate rows. To eliminate duplicate rows in the result, the DISTINCT keyword is used in the SELECT clause.

Question 16: What is the purpose of the condition operators BETWEEN and IN?
The BETWEEN operator displays rows based on a range of values. The IN condition operator checks for values contained in a specific set of values.

Question 17: How do you search for a value in a database table when you don't have the exact value to search for?
In such cases, the LIKE condition operator is used to select rows that match a character pattern. This is also called 'wildcard' search.

Question 18: What is the default ordering of data using the ORDER BY clause? How could it be changed?
The default sorting order is ascending. It can be changed using the DESC keyword, after the column name in the ORDER BY clause.

Question 19: What are the specific uses of SQL functions?
SQL functions have the following uses –
Performing calculations on data
Modifying individual data items
Manipulating the output
Formatting dates and numbers
Converting data types

Question 20: What are the case manipulation functions of SQL?
LOWER, UPPER, INITCAP

Question 21: Which function returns the remainder in a division operation?
The MOD function returns the remainder in a division operation.

Question 22: What is the purpose of the NVL function?
The NVL function converts a NULL value to an actual value.

Question 23: What is the difference between the NVL and the NVL2 functions?
The NVL(exp1, exp2) function converts the source expression (or value) exp1 to the target expression (or value) exp2, if exp1 contains NULL. The return value has the same data type as that of exp1.
The NVL2(exp1, exp2, exp3) function checks the first expression exp1, if it is not null then, the second expression exp2 is returned. If the first expression exp1 is null, then the third expression exp3 is returned.

Question 24: What is the use of the NULLIF function?
The NULLIF function compares two expressions. If they are equal, the function returns null. If they are not equal, the first expression is returned.

Question 25: Which expressions or functions allow you to implement conditional processing in a SQL statement?
There are two ways to implement conditional processing or IF-THEN-ELSE logic in a SQL statement.
Using CASE expression
Using the DECODE function

Question 26: You want to display a result query from joining two tables with 20 and 10 rows respectively. Erroneously you forget to write the WHERE clause. What would be the result?
The result would be the Cartesian product of two tables with 20 x 10 = 200 rows.

Question 27: What is the difference between cross joins and natural joins?
The cross join produces the cross product or Cartesian product of two tables. The natural join is based on all the columns having same name and data types in both the tables.

Question 28: What is the purpose of the group functions in SQL? Give some examples of group functions.
Group functions in SQL work on sets of rows and returns one result per group. Examples of group functions are AVG, COUNT, MAX, MIN, STDDEV, SUM, VARIANCE.

Question 29: Say True or False. Give explanation if False.
By default the group functions consider only distinct values in the set.
By default, group functions consider all values including the duplicate values.

Question 30: Say True or False. Give explanation if False.
The DISTINCT keyword allows a function consider only non-duplicate values.
True.

Question 31: Say True or False. Give explanation if False.
All group functions ignore null values.
True.

Question 32: Say True or False. Give explanation if False.
COUNT(*) returns the number of columns in a table.
False. COUNT(*) returns the number of rows in a table.

Question 33: What's wrong in the following query?
SELECT subject_code, count(name)
FROM students;
It doesn't have a GROUP BY clause. The subject_code should be in the GROUP BY clause.

```
SELECT subject_code, count(name)
FROM students
GROUP BY subject_code;
```

Question 34: What's wrong in the following query?

```
SELECT subject_code, AVG (marks)
FROM students
WHERE AVG(marks) > 75
GROUP BY subject_code;
```

The WHERE clause cannot be used to restrict groups. The HAVING clause should be used.

```
SELECT subject_code, AVG (marks)
FROM students
HAVING AVG(marks) > 75
GROUP BY subject_code;
```

Question 35: Say True or False. Give explanation if False.
Group functions cannot be nested.
False. Group functions can be nested to a depth of two.

Question 36: What do you understand by a subquery? When is it used?
A subquery is a SELECT statement embedded in a clause of another SELECT statement. It is used when the inner query, or the subquery returns a value that is used by the outer query. It is very useful in selecting some rows in a table with a condition that depends on some data which is contained in the same table.

Question 37: Say True or False. Give explanation if False.
A single row subquery returns only one row from the outer SELECT statement
False. A single row subquery returns only one row from the inner SELECT statement.

Question 38: Say True or False. Give explanation if False.
A multiple row subquery returns more than one row from the inner SELECT statement.
True.

Question 39: Say True or False. Give explanation if False.
Multiple column subqueries return more than one column from the inner SELECT statement.
True.

Question 40: What's wrong in the following query?

```
SELECT student_code, name
FROM students
WHERE marks =
(SELECT MAX(marks)
        FROM students
        GROUP BY subject_code);
```

Here a single row operator = is used with a multiple row subquery.

Question 41: What are the various multiple row comparison operators in SQL?
IN, ANY, ALL.

Question 42: What is the pupose of DML statements in SQL?
The DML statements are used to add new rows to a table, update or modify data in existing rows, or remove existing rows from a table.

Question 43: Which statement is used to add a new row in a database table?

The INSERT INTO statement.

Question 44: Say True or False. Give explanation if False.
While inserting new rows in a table you must list values in the default order of the columns.
True.

Question 45: How do you insert null values in a column while inserting data?
Null values can be inserted into a table by one of the following ways –
Implicitly by omitting the column from the column list.
Explicitly by specifying the NULL keyword in the VALUES clause.

Question 46: Say True or False. Give explanation if False.
INSERT statement does not allow copying rows from one table to another.
False. INSERT statement allows to add rows to a table copying rows from an existing table.

Question 47: How do you copy rows from one table to another?
The INSERT statement can be used to add rows to a table by copying from another table. In this case, a subquery is used in the place of the VALUES clause.

Question 48: What happens if you omit the WHERE clause in the UPDATE statement?
All the rows in the table are modified.

Question 49: Can you modify the rows in a table based on values from another table? Explain.
Yes. Use of subqueries in UPDATE statements allow you to update rows in a table based on values from another table.

Question 50: Say True or False. Give explanation if False.
The DELETE statement is used to delete a table from the database.
False. The DELETE statement is used for removing existing rows from a table.

Question 51: What happens if you omit the WHERE clause in a delete statement?
All the rows in the table are deleted.

Question 52: Can you remove rows from a table based on values from another table? Explain.
Yes, subqueries can be used to remove rows from a table based on values from another table.

Question 53: Say True or False. Give explanation if False.
Attempting to delete a record with a value attached to an integrity constraint, returns an error.
True.

Question 54: Say True or False. Give explanation if False.
You can use a subquery in an INSERT statement.
True.

Question 55: What is the purpose of the MERGE statement in SQL?
The MERGE statement allows conditional update or insertion of data into a database table. It performs an UPDATE if the rows exists, or an INSERT if the row does not exist.

Question 56: Say True or False. Give explanation if False.
A DDL statement or a DCL statement is automatically committed.
True.

Question 57: What is the difference between VARCHAR2 AND CHAR datatypes?
VARCHAR2 represents variable length character data, whereas CHAR represents fixed length character data.

Question 58: Say True or False. Give explanation if False.
A DROP TABLE statement can be rolled back.
False. A DROP TABLE statement cannot be rolled back.

Question 59: Which SQL statement is used to add, modify or drop columns in a database table?
The ALTER TABLE statement.

Question 60: What is a view? Why should you use a view?
A view is a logical snapshot based on a table or another view. It is used for –
Restricting access to data;
Making complex queries simple;
Ensuring data independency;
Providing different views of same data.

Question 61: Discuss the syntax and use of the COALESCE function?
The COALESCE function has the expression COALESCE(exp1, exp2, ... expn)
It returns the first non-null expression given in the parameter list.

Statistics Interview Questions

Statistics Interview Questions

Question 1: What is the Central Limit Theorem and why is it important?
"Suppose that we are interested in estimating the average height among all people. Collecting data for every person in the world is impossible. While we can't obtain a height measurement from everyone in the population, we can still sample some people. The question now becomes, what can we say about the average height of the entire population given a single sample. The Central Limit Theorem addresses this question exactly."

Question 2: What is sampling? How many sampling methods do you know?
"Data sampling is a statistical analysis technique used to select, manipulate and analyze a representative subset of data points to identify patterns and trends in the larger data set being examined."
There are two main types of Sampling techniques:
Probability Sampling
Non-Probability Sampling

Question 3: What is the difference between type I vs type II error?
"A type I error occurs when the null hypothesis is true, but is rejected. A type II error occurs when the null hypothesis is false, but erroneously fails to be rejected."

Question 4: What is linear regression?
A linear regression is a good tool for quick predictive analysis: for example, the price of a house depends on a myriad of factors, such as its size or its location. In order to see the relationship between these variables, we need to build a linear regression, which predicts the line of best fit between them and can help conclude whether or not these two factors have a positive or negative relationship.

Question 5: What are the assumptions required for linear regression?
There are four major assumptions:
1. There is a linear relationship between the dependent variables and the regressors, meaning the model you are creating actually fits the data,
2. The errors or residuals of the data are normally distributed and independent from each other,
3. There is minimal multicollinearity between explanatory variables, and
4. Homoscedasticity. This means the variance around the regression line is the same for all values of the predictor variable.

Question 6: What is a statistical interaction?
"Basically, an interaction is when the effect of one factor (input variable) on the dependent variable (output variable) differs among levels of another factor.".

Question 7: What is selection bias?
"Selection (or 'sampling') bias occurs in an 'active,' sense when the sample data that is gathered and prepared for modeling has characteristics that are not representative of the true, future population of cases the model will see. That is, active selection bias occurs when a subset of the data are systematically (i.e., non-randomly) excluded from analysis."

Question 8: What is an example of a data set with a non-Gaussian distribution?
"The Gaussian distribution is part of the Exponential family of distributions, but there are a lot more of them, with the same sort of ease of use, in many cases, and if the person doing the machine learning has a solid grounding in statistics, they can be utilized where appropriate."

Question 9: What is the Binomial Probability Formula?
"The binomial distribution consists of the probabilities of each of the possible numbers of successes on N trials for independent events that each have a probability of n (the Greek letter pi) of occurring.

The binomial distribution formula is:

$$b(x; n, P) = {}_nC_x * P^x * (1 - P)^{n-x}$$

Where:

b = binomial probability

x = total number of "successes" (pass or fail, heads or tails, etc.)

P = probability of success on an individual trial

n = number of trials

Question 10: What is statistical power?

Wikipedia defines Statistical power or sensitivity of a binary hypothesis test is the probability that

the test correctly rejects the null hypothesis (H0) when the alternative hypothesis (H1) is true.

To put in another way, Statistical power is the likelihood that a study will detect an effect when the effect is present. The higher the statistical power, the less likely you are to make a Type II error (concluding there is no effect when, in fact, there is).

Question 11: Explain what resampling methods are and why they are useful. Also explain their limitations.
Classical statistical parametric tests compare observed statistics to theoretical sampling distributions. Resampling a data-driven, not theory-driven methodology which is based upon repeated sampling within the same sample.

Resampling refers to methods for doing one of these
Estimating the precision of sample statistics (medians, variances, percentiles) by using subsets of available data (jackknifing) or drawing randomly with replacement from a set of data points (bootstrapping)
Exchanging labels on data points when performing significance tests (permutation tests, also called exact tests, randomization tests, or re-randomization tests)
Validating models by using random subsets (bootstrapping, cross validation)

Question 12: What is selection bias, why is it important and how can you avoid it?
Selection bias, in general, is a problematic situation in which error is introduced due to a non-random population sample. For example, if a given sample of 100 test cases was made up of a 60/20/15/5 split of 4 classes which actually occurred in relatively equal numbers in the population, then a given model may make the false assumption that probability could be the determining predictive factor. Avoiding non-random samples is the best way to deal with bias; however, when this is impractical, techniques such as resampling, boosting, and weighting are strategies which can be introduced to help deal with the situation.

Question 13: What is the difference between "long" and "wide" format data?
In the **wide-format**, a subject's repeated responses will be in a single row, and each response is in a separate column. In the **long-format**, each row is a one-time point per subject. You can recognize data in wide format by the fact that columns generally represent groups.

Question 14: What do you understand by the term Normal Distribution?
Data is usually distributed in different ways with a bias to the left or to the right or it can all be jumbled up. However, there are chances that data is distributed around a central value without any bias to the left or right and reaches normal distribution in the form of a bell-shaped curve.

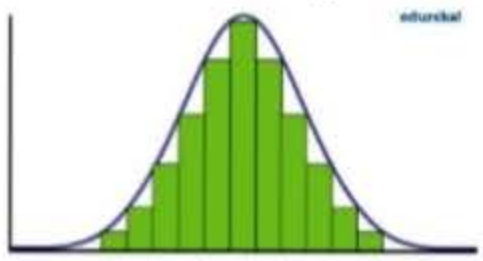


Figure:*Normal distribution in a bell curve*
The random variables are distributed in the form of a symmetrical, bell-shaped curve.
Properties of Normal Distribution are as follows;
Unimodal -one mode
Symmetrical -left and right halves are mirror images
Bell-shaped -maximum height (mode) at the mean
Mean, Mode, and Median are all located in the center
Asymptotic

Question 15: What is correlation and covariance in statistics?
Covariance and Correlation are two mathematical concepts; these two approaches are widely used in statistics. Both Correlation and Covariance establish the relationship and also measure the dependency between two random variables. Though the work is similar between these two in mathematical terms, they are different from each other.
Correlation: Correlation is considered or described as the best technique for measuring and also for estimating the quantitative relationship between two variables. Correlation measures how strongly two variables are related.
Covariance: In covariance two items vary together and it's a measure that indicates the extent to which two random variables change in cycle. It is a statistical term; it explains the systematic relation between a pair of random variables, wherein changes in one variable reciprocal by a corresponding change in another variable.

Question 16: What is the difference between Point Estimates and Confidence Interval?
Point Estimation gives us a particular value as an estimate of a population parameter. Method of Moments and Maximum Likelihood estimator methods are used to derive Point Estimators for population parameters.
A confidence interval gives us a range of values which is likely to contain the population parameter. The confidence interval is generally preferred, as it tells us how likely this interval is to contain the population parameter. This likeliness or probability is called Confidence Level or Confidence coefficient and represented by 1 — alpha, where alpha is the level of significance.

Question 17: What is the goal of A/B Testing?
It is a hypothesis testing for a randomized experiment with two variables A and B.
The goal of A/B Testing is to identify any changes to the web page to maximize or increase the outcome of interest. A/B testing is a fantastic method for figuring out the best online promotional and marketing strategies for your business. It can be used to test everything from website copy to sales emails to search ads
An example of this could be identifying the click-through rate for a banner ad.

Question 18: What is p-value?
When you perform a hypothesis test in statistics, a p-value can help you determine the strength of your results. p-value is a number between 0 and 1. Based on the value it will denote the strength of the results. The claim which is on trial is called the Null Hypothesis.
Low p-value (≤ 0.05) indicates strength against the null hypothesis which means we can reject the null Hypothesis. High p-value (≥ 0.05) indicates strength for the null hypothesis which means we can accept the null Hypothesis p-value of 0.05 indicates the Hypothesis could go either way.
To put it in another way,
High P values: your data are likely with a true null. Low P values: your data are unlikely with a true null.

Question 19: In any 15-minute interval, there is a 20% probability that you will see at least one shooting star. What is the probability that you see at least one shooting star in the period of an hour?
Probability of not seeing any shooting star in 15 minutes is
= 1 – P(Seeing one shooting star)
= 1 – 0.2 = 0.8
Probability of not seeing any shooting star in the period of one hour
= (0.8) ^ 4 = 0.4096
Probability of seeing at least one shooting star in the one hour
= 1 – P(Not seeing any star)
= 1 – 0.4096 = 0.5904

Question 20: How can you generate a random number between 1 – 7 with only a die?
Any die has six sides from 1-6. There is no way to get seven equal outcomes from a single rolling of a die. If we roll the die twice and consider the event of two rolls, we now have 36 different outcomes.
To get our 7 equal outcomes we have to reduce this 36 to a number divisible by 7. We can thus consider only 35 outcomes and exclude the other one.
A simple scenario can be to exclude the combination (6,6), i.e., to roll the die again if 6 appears twice.
All the remaining combinations from (1,1) till (6,5) can be divided into 7 parts of 5 each. This way all the seven sets of outcomes are equally likely.

Question 21: A certain couple tells you that they have two children, at least one of which is a girl. What is the probability that they have two girls?
In the case of two children, there are 4 equally likely possibilities
BB, BG, GB and GG;
where **B** = Boy and **G** = Girl and the first letter denotes the first child.
From the question, we can exclude the first case of BB. Thus from the remaining 3 possibilities of **BG**, **GB** & **BB**, we have to find the probability of the case with two girls.
Thus, P(Having two girls given one girl) = **1 / 3**

Question 22: A jar has 1000 coins, of which 999 are fair and 1 is double headed. Pick a coin at random, and toss it 10 times. Given that you see 10 heads, what is the probability that the next toss of that coin is also a head?
There are two ways of choosing the coin. One is to pick a fair coin and the other is to pick the one with two heads.
Probability of selecting fair coin = 999/1000 = **0.999**
Probability of selecting unfair coin = 1/1000 = **0.001**
Selecting 10 heads in a row = Selecting fair coin * Getting 10 heads + Selecting an unfair coin
P (A) = 0.999 * (1/2)^5 = 0.999 * (1/1024) = **0.000976**
P (B) = 0.001 * 1 = **0.001**
P(A / A + B) = 0.000976 / (0.000976 + 0.001) = **0.4939**
P(B / A + B) = 0.001 / 0.001976 = **0.5061**
Probability of selecting another head = P(A/A+B) * 0.5 + P(B/A+B) * 1 = 0.4939 * 0.5 + 0.5061 = **0.7531**

Question 23: What do you understand by statistical power of sensitivity and how do you calculate it?
Sensitivity is commonly used to validate the accuracy of a classifier (Logistic, SVM, Random Forest etc.).
Sensitivity is nothing but "Predicted True events/ Total events". True events here are the events which were true and model also predicted them as true.
Calculation of seasonality is pretty straightforward.
Seasonality = (**True Positives**) / (**Positives in Actual Dependent Variable**)

Question 24: Why Is Re-sampling Done?
Resampling is done in any of these cases:
Estimating the accuracy of sample statistics by using subsets of accessible data or drawing randomly with replacement from a set of data points
Substituting labels on data points when performing significance tests
Validating models by using random subsets (bootstrapping, cross-validation)

Question 25: What are the differences between over-fitting and under-fitting?

In statistics and machine learning, one of the most common tasks is to fit a *model* to a set of training data, so as to be able to make reliable predictions on general untrained data.

In **overfitting**, a statistical model describes random error or noise instead of the underlying relationship. Overfitting occurs when a model is excessively complex, such as having too many parameters relative to the number of observations. A model that has been overfitted, has poor predictive performance, as it overreacts to minor fluctuations in the training data.
Underfitting occurs when a statistical model or machine learning algorithm cannot capture the underlying trend of the data. Underfitting would occur, for example, when fitting a linear model to non-linear data. Such a model too would have poor predictive performance.

Question 26: How to combat Overfitting and Underfitting?
To combat overfitting and underfitting, you can resample the data to estimate the model accuracy (k-fold cross-validation) and by having a validation dataset to evaluate the model.

Question 27: What is regularisation? Why is it useful?
Regularisation is the process of adding tuning parameter to a model to induce smoothness in order to prevent overfitting. This is most often done by adding a constant multiple to an existing weight vector. This constant is often the L1(Lasso) or L2(ridge). The model predictions should then minimize the loss function calculated on the regularized training set.

Question 28: What Is the Law of Large Numbers?
It is a theorem that describes the result of performing the same experiment a large number of times. This theorem forms the basis of **frequency-style** thinking. It says that the sample means, the sample variance and the sample standard deviation converge to what they are trying to estimate.

Question 29: What Are Confounding Variables?
In statistics, a confounder is a variable that influences both the dependent variable and independent variable.
For example, if you are researching whether a lack of exercise leads to weight gain,
lack of exercise = independent variable
weight gain = dependent variable.
A confounding variable here would be any other variable that affects both of these variables, such as the **age of the subject**.

Question 30: What Are the Types of Biases That Can Occur During Sampling?
Selection bias
Under coverage bias
Survivorship bias

Question 31: What is Survivorship Bias?
It is the logical error of focusing aspects that support surviving some process and casually overlooking those that did not work because of their lack of prominence. This can lead to wrong conclusions in numerous different means.

Question 32: What is selection Bias?
Selection bias occurs when the sample obtained is not representative of the population intended to be analysed.

Question 33: Explain how a ROC curve works?

The **ROC** curve is a graphical representation of the contrast between true positive rates and false-positive rates at various thresholds. It is often used as a proxy for the trade-off between the sensitivity(true positive rate) and false-positive rate.

Question 34: What is TF/IDF vectorization?
TF–IDF is short for term frequency-inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining.
The TF–IDF value increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus, which helps to adjust for the fact that some words appear more frequently in general.

Question 35: Why we generally use Softmax non-linearity function as last operation in-network?
It is because it takes in a vector of real numbers and returns a probability distribution. Its definition is as follows. Let x be a vector of real numbers (positive, negative, whatever, there are no constraints).
Then the i'th component of Softmax(*x) is* —

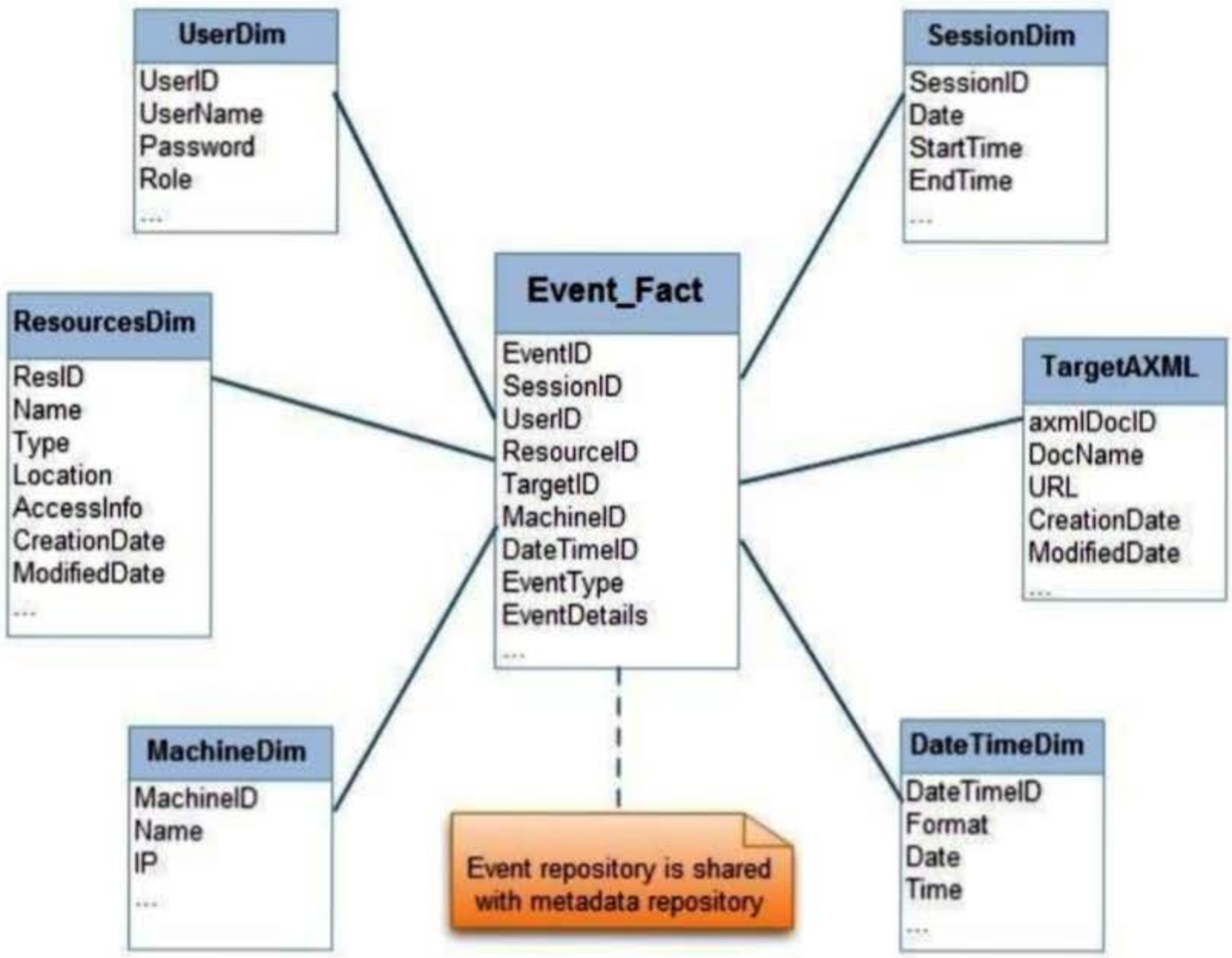
It should be clear that the output is a probability distribution: each element is non-negative and the sum over all components is 1.

Question 37: Python or R – Which one would you prefer for text analytics?
We will prefer Python because of the following reasons:
Python would be the best option because it has Pandas library that provides easy to use data structures and high-performance data analysis tools.
R is more suitable for machine learning than just text analysis.
Python performs faster for all types of text analytics.

Question 38: How does data cleaning plays a vital role in the analysis?
Data cleaning can help in the analysis because:
Cleaning data from multiple sources helps to transform it into a format that data analysts or data scientists can work with.
Data Cleaning helps to increase the accuracy of the model in machine learning.
It is a cumbersome process because as the number of data sources increases, the time taken to clean the data increases exponentially due to the number of sources and the volume of data generated by these sources.
It might take up to 80% of the time for just cleaning data making it a critical part of the analysis task.

Question 39: Differentiate between univariate, bivariate, and multivariate analysis.
Univariate *analyses* are descriptive statistical analysis techniques which can be differentiated based on the number of variables involved at a given point of time. For example, the pie charts of sales based on territory involve only one variable, and can the analysis can be referred to as univariate analysis.
The ***bivariate*** *analysis* attempts to understand the difference between two variables at a time as in a scatterplot. For example, analyzing the volume of sales and spending can be considered as an example of bivariate analysis.
The multivariate analysis deals with the study of more than two variables to understand the effect of variables on the responses.

Question 40: Explain Star Schema.
It is a traditional database schema with a central table. Satellite tables map IDs to physical names or descriptions and can be connected to the central fact table using the ID fields; these tables are known as lookup tables and are principally useful in real-time applications, as they save a lot of memory. Sometimes star schemas involve several layers of summarization to recover information faster.



Question 41: What is Cluster Sampling?
Cluster sampling is a technique used when it becomes difficult to study the target population spread across a wide area and simple random sampling cannot be applied. Cluster Sample is a probability sample where each sampling unit is a collection or cluster of elements.
For eg., A researcher wants to survey the academic performance of high school students in Japan. He can divide the entire population of Japan into different clusters (cities). Then the researcher selects a number of clusters depending on his research through simple or systematic random sampling.

Question 42: What is Systematic Sampling?
Systematic sampling is a statistical technique where elements are selected from an ordered sampling frame. In systematic sampling, the list is progressed in a circular manner so once you reach the end of the list, it is progressed from the top again. The best example of systematic sampling is equal probability method.

Question 43: What are Eigenvectors and Eigenvalues?
Eigenvectors are used for understanding linear transformations. In data analysis, we usually calculate the eigenvectors for a correlation or covariance matrix. Eigenvectors are the directions along which a particular linear transformation acts by flipping, compressing or stretching.
Eigenvalue can be referred to as the strength of the transformation in the direction of eigenvector or the factor by which the compression occurs.

Question 44: Can you cite some examples where a false positive is important than a false negative?

Let us first understand what false positives and false negatives are.

False Positives are the cases where you wrongly classified a non-event as an event a.k.a Type I error.

False Negatives are the cases where you wrongly classify events as non-events, a.k.a Type II error.

Example 1: In the medical field, assume you have to give chemotherapy to patients. Assume a patient comes to that hospital and he is tested positive for cancer, based on the lab prediction but he actually doesn't have cancer. This is a case of false positive. Here it is of utmost danger to start chemotherapy on this patient when he actually does not have cancer. In the absence of cancerous cell, chemotherapy will do certain damage to his normal healthy cells and might lead to severe diseases, even cancer.

Example 2: Let's say an e-commerce company decided to give $1000 Gift voucher to the customers whom they assume to purchase at least $10,000 worth of items. They send free voucher mail directly to 100 customers without any minimum purchase condition because they assume to make at least 20% profit on sold items above $10,000. Now the issue is if we send the $1000 gift vouchers to customers who have not actually purchased anything but are marked as having made $10,000 worth of purchase.

Question 45: Can you cite some examples where a false negative important than a false positive?

***Example 1*:** Assume there is an airport 'A' which has received high-security threats and based on certain characteristics they identify whether a particular passenger can be a threat or not. Due to a shortage of staff, they decide to scan passengers being predicted as risk positives by their predictive model. What will happen if a true threat customer is being flagged as non-threat by airport model?

***Example 2*:** What if Jury or judge decides to make a criminal go free?

***Example 3*:** What if you rejected to marry a very good person based on your predictive model and you happen to meet him/her after a few years and realize that you had a false negative?

Question 46: Can you cite some examples where both false positive and false negatives are equally important?

In the **Banking** industry giving loans is the primary source of making money but at the same time if your repayment rate is not good you will not make any profit, rather you will risk huge losses.

Banks don't want to lose good customers and at the same point in time, they don't want to acquire bad customers. In this scenario, both the false positives and false negatives become very important to measure.

Question 47: Can you explain the difference between a Validation Set and a Test Set?

A ***Validationset*** can be considered as a part of the training set as it is used for parameter selection and to avoid overfitting of the model being built.

On the other hand, a ***Test Set*** is used for testing or evaluating the performance of a trained machine learning model.

In simple terms, the differences can be summarized as; training set is to fit the parameters i.e. weights and test set is to assess the performance of the model i.e. evaluating the predictive power and generalization.

Question 48: Explain cross-validation.

Cross-validation is a model validation technique for evaluating how the outcomes of statistical analysis will **generalize** to an **independent dataset**. Mainly used in backgrounds where the

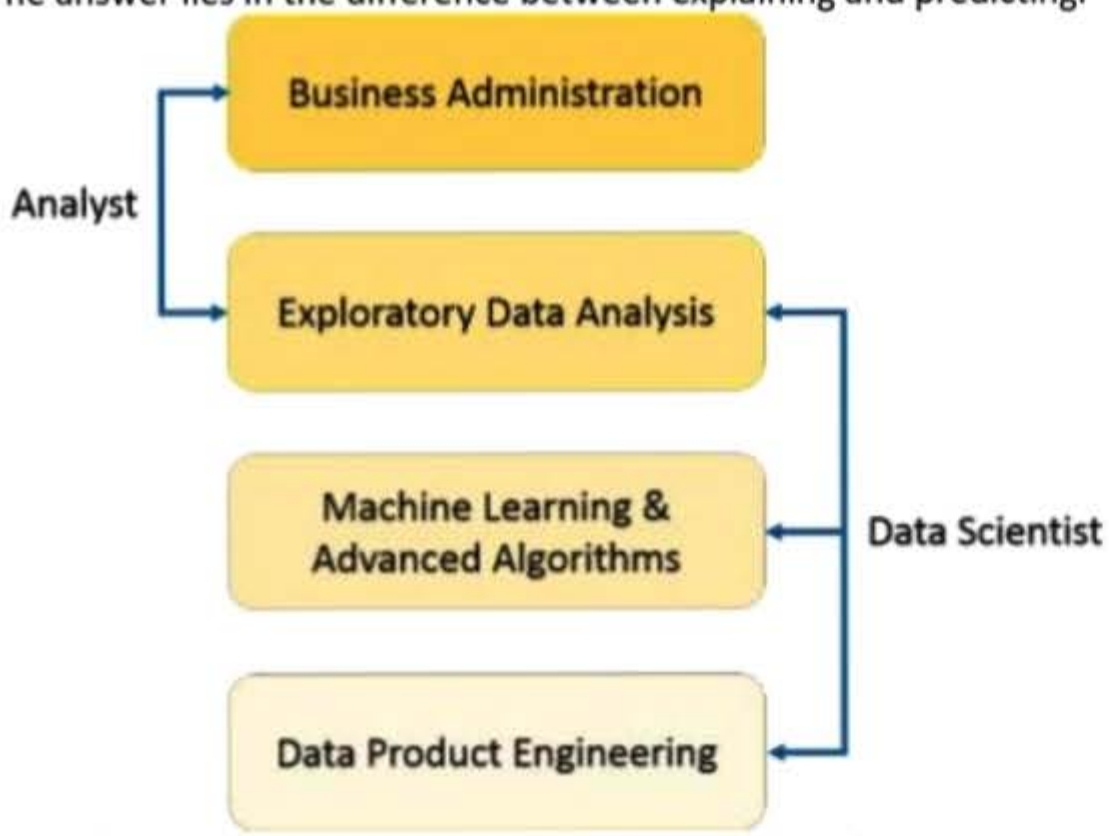
objective is forecast and one wants to estimate how accurately a model will accomplish in practice.
The goal of cross-validation is to term a data set to test the model in the training phase (i.e. validation data set) in order to limit problems like overfitting and get an insight on how the model will generalize to an independent data set.

Machine Learning Interview Questions

Machine Learning Interview Questions

Question 1: What is Data Science? List the differences between supervised and unsupervised learning.

Data Science is a blend of various tools, algorithms, and machine learning principles with the goal to discover hidden patterns from the raw data. How is this different from what statisticians have been doing for years?
The answer lies in the difference between explaining and predicting.



The differences between supervised and unsupervised learning are as follows;

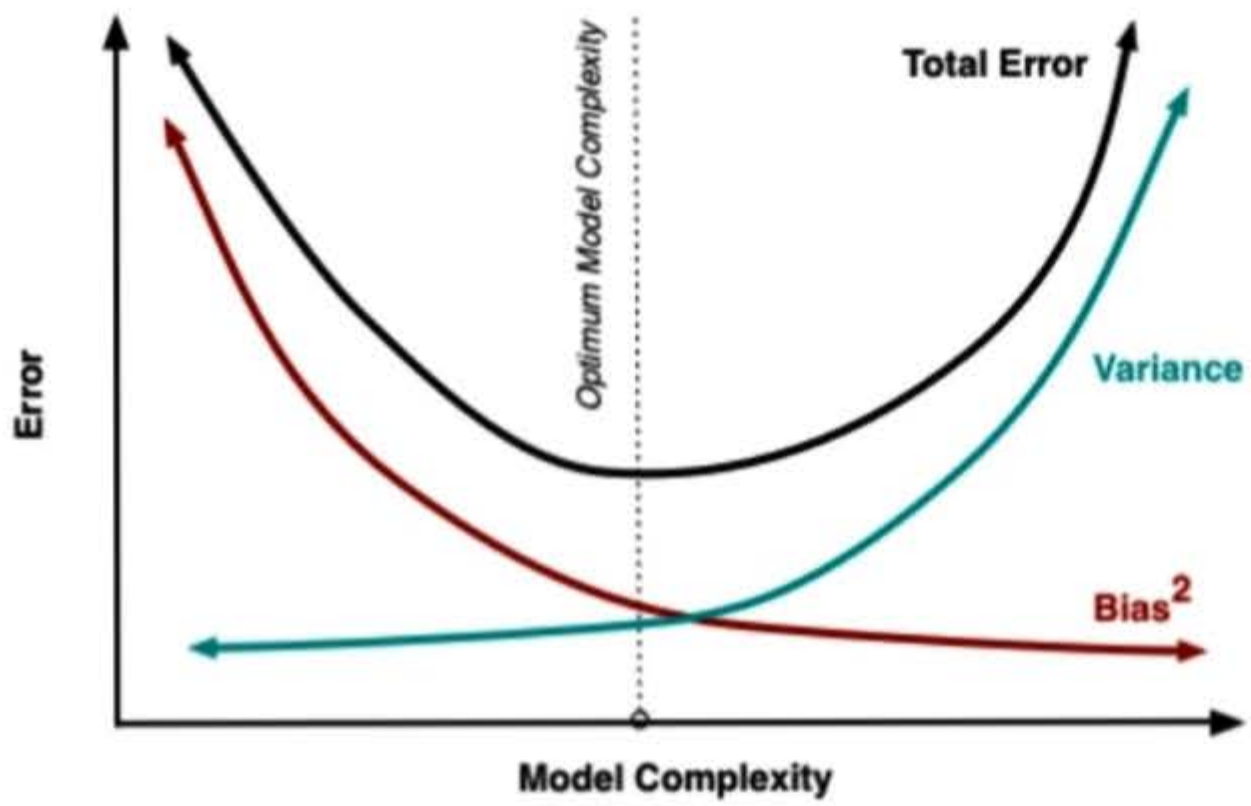
Supervised Learning	Unsupervised Learning
Input data is labelled.	Input data is unlabelled.
Uses a training data set.	Uses the input data set.
Used for prediction.	Used for analysis.
Enables classification and regression.	Enables Classification, Density Estimation, & Dimension Reduction

Question 2: What is Selection Bias?

Selection bias is a kind of error that occurs when the researcher decides who is going to be studied. It is usually associated with research where the selection of participants isn't random. It is sometimes referred to as the selection effect. It is the distortion of statistical analysis, resulting from the method of collecting samples. If the selection bias is not taken into account, then some conclusions of the study may not be accurate.
The types of selection bias include:
Sampling bias: It is a systematic error due to a non-random sample of a population causing some members of the population to be less likely to be included than others resulting in a biased sample.
Time interval: A trial may be terminated early at an extreme value (often for ethical reasons), but the extreme value is likely to be reached by the variable with the largest variance, even if all variables have a similar mean.
Data: When specific subsets of data are chosen to support a conclusion or rejection of bad data on arbitrary grounds, instead of according to previously stated or generally agreed criteria.
Attrition: Attrition bias is a kind of selection bias caused by attrition (loss of participants) discounting trial subjects/tests that did not run to completion.

Question 3: What is bias-variance trade-off?

Bias: Bias is an error introduced in your model due to oversimplification of the machine learning algorithm. It can lead to underfitting. When you train your model at that time model makes simplified assumptions to make the target function easier to understand.
Low bias machine learning algorithms — Decision Trees, k-NN and SVM High bias machine learning algorithms — Linear Regression, Logistic Regression
Variance: Variance is error introduced in your model due to complex machine learning algorithm, your model learns noise also from the training data set and performs badly on test data set. It can lead to high sensitivity and overfitting.
Normally, as you increase the complexity of your model, you will see a reduction in error due to lower bias in the model. However, this only happens until a particular point. As you continue to make your model more complex, you end up over-fitting your model and hence your model will start suffering from high variance.



Bias-Variance trade-off: The goal of any supervised machine learning algorithm is to have low bias and low variance to achieve good prediction performance.
The k-nearest neighbour algorithm has low bias and high variance, but the trade-off can be changed by increasing the value of k which increases the number of neighbours that contribute to the prediction and in turn increases the bias of the model.
The support vector machine algorithm has low bias and high variance, but the trade-off can be changed by increasing the C parameter that influences the number of violations of the margin allowed in the training data which increases the bias but decreases the variance.
There is no escaping the relationship between bias and variance in machine learning. Increasing the bias will decrease the variance. Increasing the variance will decrease bias.

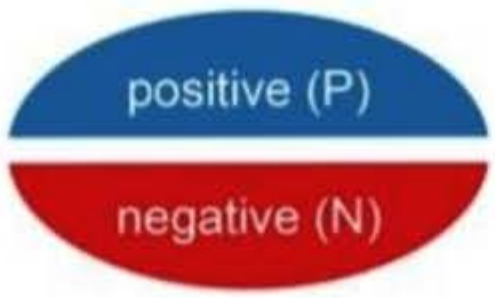
Question 4: What is a confusion matrix?

The confusion matrix is a 2X2 table that contains 4 outputs provided by the **binary classifier**. Various measures, such as error-rate, accuracy, specificity, sensitivity, precision and recall are derived from it. *Confusion Matrix*

		Predicted class: P	Predicted class: N
Actual Class	P	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

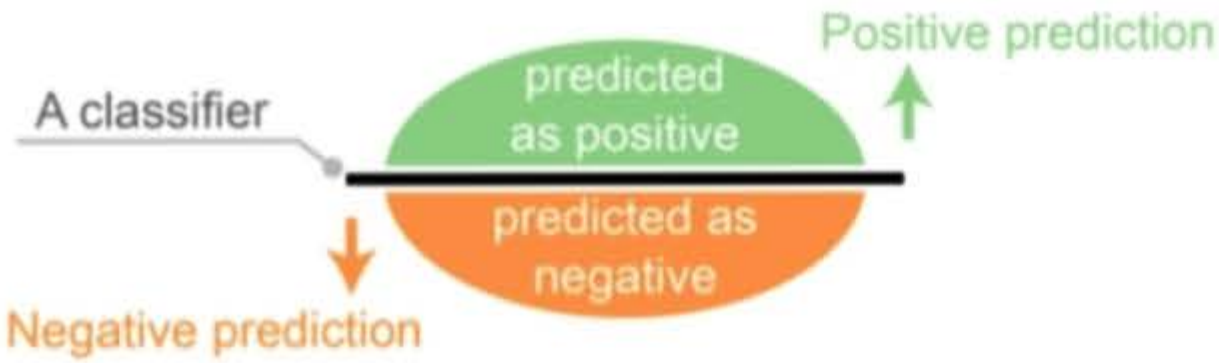
A data set used for performance evaluation is called a **test data set**. It should contain the correct labels and predicted labels.

Two actual classes or observed labels



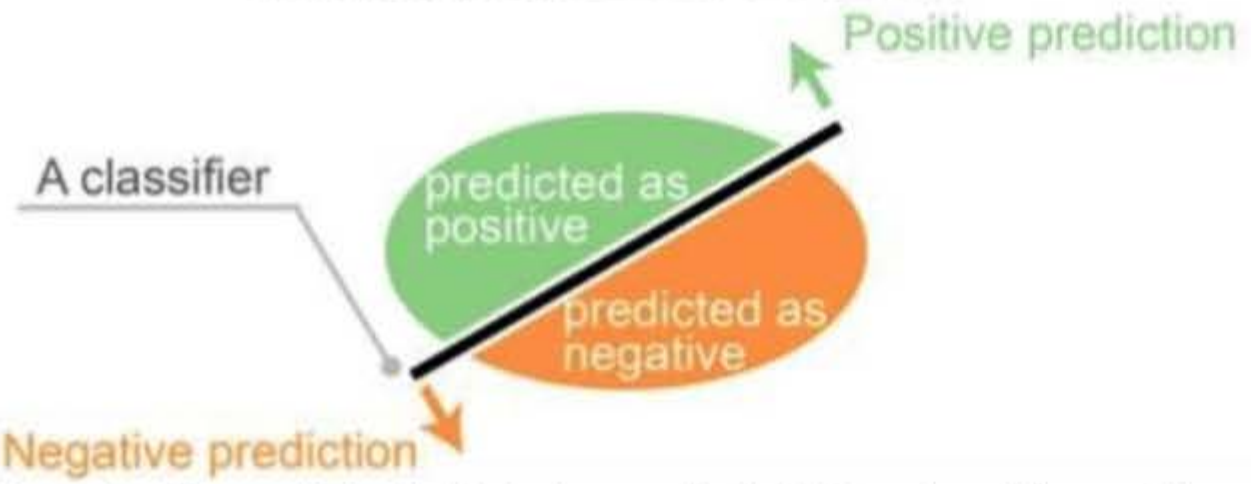
The predicted labels will exactly the same if the performance of a binary classifier is perfect.

Predicted classes of a perfect classifier



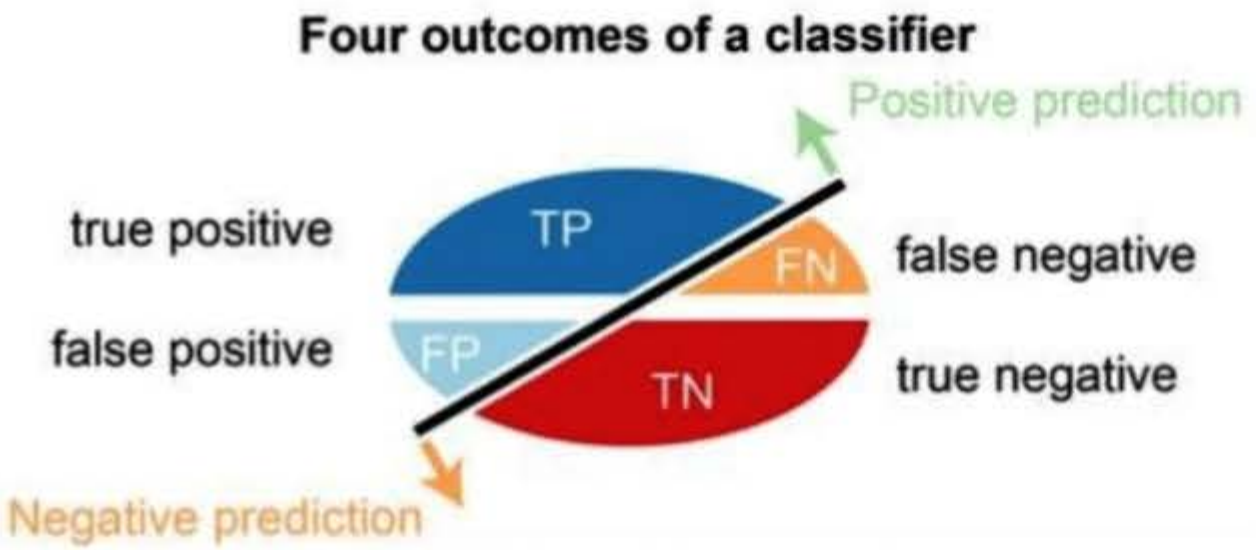
The predicted labels usually match with part of the observed labels in real-world scenarios.

Predicted classes of a classifier



A binary classifier predicts all data instances of a test data set as either positive or negative. This produces four outcomes-

True-positive(TP) — Correct positive prediction
False-positive(FP) — Incorrect positive prediction
True-negative(TN) — Correct negative prediction
False-negative(FN) — Incorrect negative prediction



Basic measures derived from the confusion matrix

Accuracy = (TP+TN)/(P+N)

Sensitivity(Recall or True positive rate) = TP/P

Specificity(True negative rate) = TN/N

Precision(Positive predicted value) = TP/(TP+FP)

F-Score(Harmonic mean of precision and recall) = (1+b)(PREC.REC)/(b^2PREC+REC) where b is commonly 0.5, 1, 2.

Question 5: What is Supervised Learning?

Supervised learning is the machine learning task of inferring a function from labeled training data. The training data consist of a set of training examples.

Algorithms: Support Vector Machines, Regression, Naive Bayes, Decision Trees, K-nearest Neighbor Algorithm and Neural Networks

E.g. If you built a fruit classifier, the labels will be "this is an orange, this is an apple and this is a banana", based on showing the classifier examples of apples, oranges and bananas.

Question 6: What is Unsupervised learning?

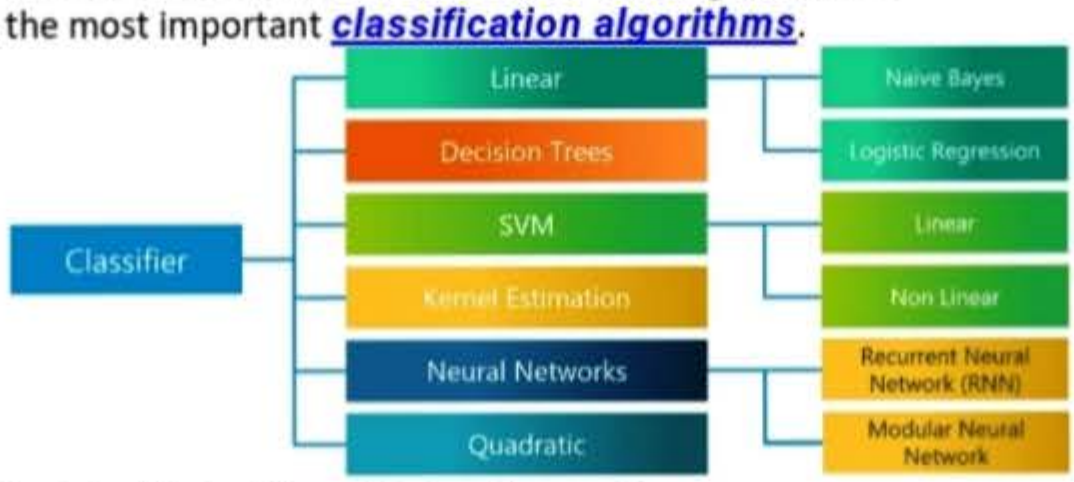
Unsupervised learning is a type of machine learning algorithm used to draw inferences from datasets consisting of input data without labelled responses.

Algorithms: Clustering, Anomaly Detection, Neural Networks and Latent Variable Models

E.g. In the same example, a fruit clustering will categorize as "fruits with soft skin and lots of dimples", "fruits with shiny hard skin" and "elongated yellow fruits".

Question 7: What are the various classification algorithms?

The diagram lists the most important ***classification algorithms***.



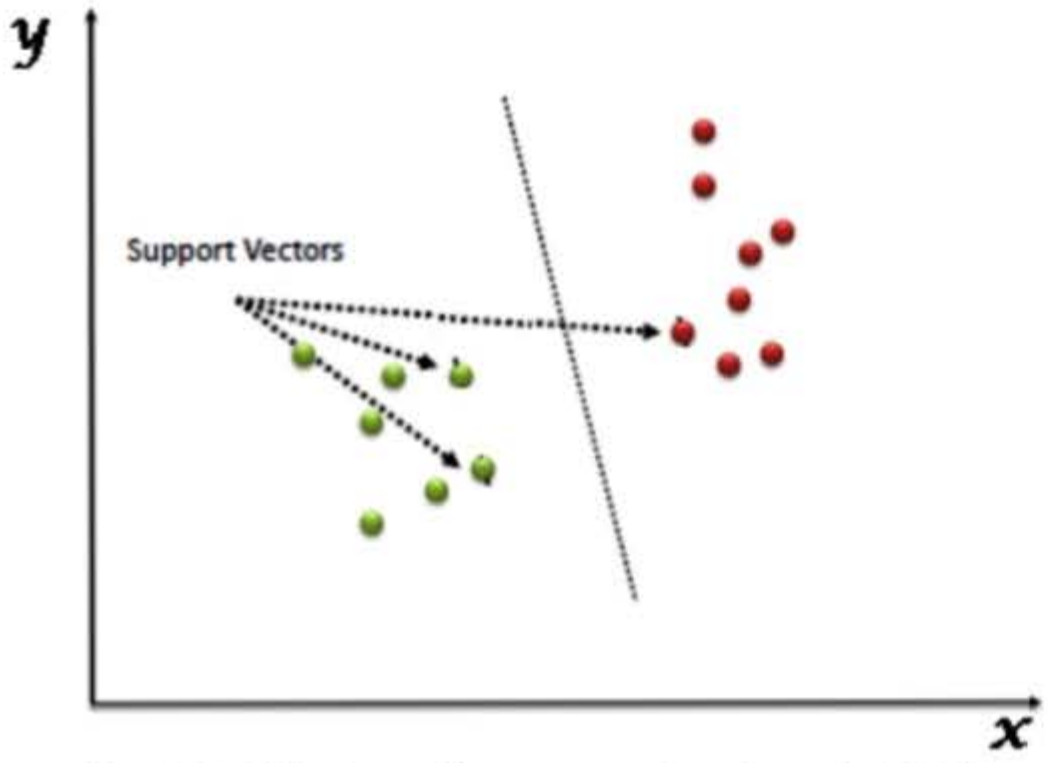
Question 8: What is 'Naive' in a Naive Bayes?

The ***Naive Bayes Algorithm*** is based on the Bayes Theorem. Bayes' theorem describes the probability of an event, based on prior knowledge of conditions that might be related to the event.

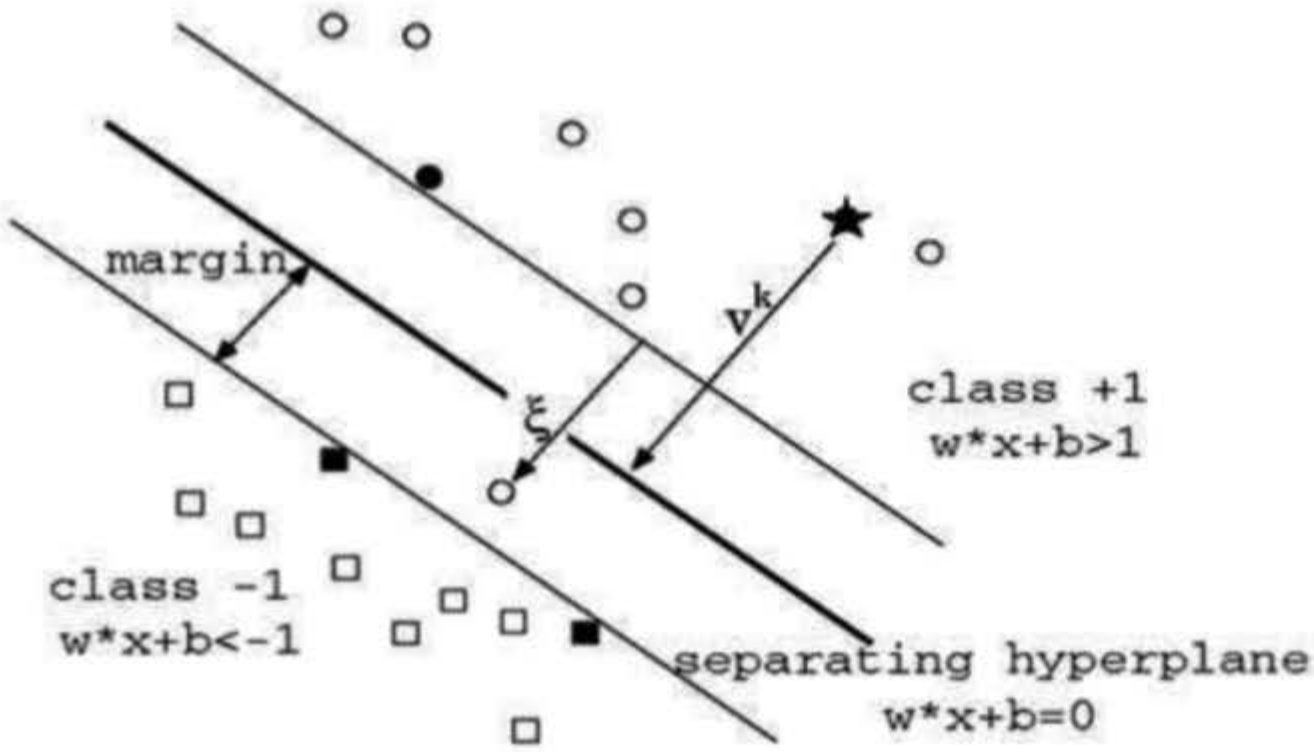
The Algorithm is 'naive' because it makes assumptions that may or may not turn out to be correct.

Question 9: Explain SVM algorithm in detail.

SVM stands for support vector machine, it is a supervised machine learning algorithm which can be used for both ***Regression*** **and Classification**. If you have n features in your training data set, SVM tries to plot it in n-dimensional space with the value of each feature being the value of a particular coordinate. SVM uses hyperplanes to separate out different classes based on the provided kernel function.



Question 10: What are the support vectors in SVM?



In the diagram, we see that the thinner lines mark the distance from the classifier to the closest data points called the support vectors (darkened data points). The distance between the two thin lines is called the margin.

Question 11: What is Machine Learning?

Machine Learning explores the study and construction of algorithms that can learn from and make predictions on data. Closely related to computational statistics. Used to devise complex models and algorithms that lend themselves to a prediction which in commercial use is known as predictive analytics. Given below, is an image representing the various domains Machine Learning lends itself to.

Question 12: What are the different kernels in SVM?

There are four types of kernels in SVM.
Linear Kernel
Polynomial kernel
Radial basis kernel
Sigmoid kernel

Question 13: Explain Decision Tree algorithm in detail.

A ***decision tree***is a supervised machine learning algorithm mainly used for **Regression and Classification**. It breaks down a data set into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision tree can handle both categorical and numerical data.

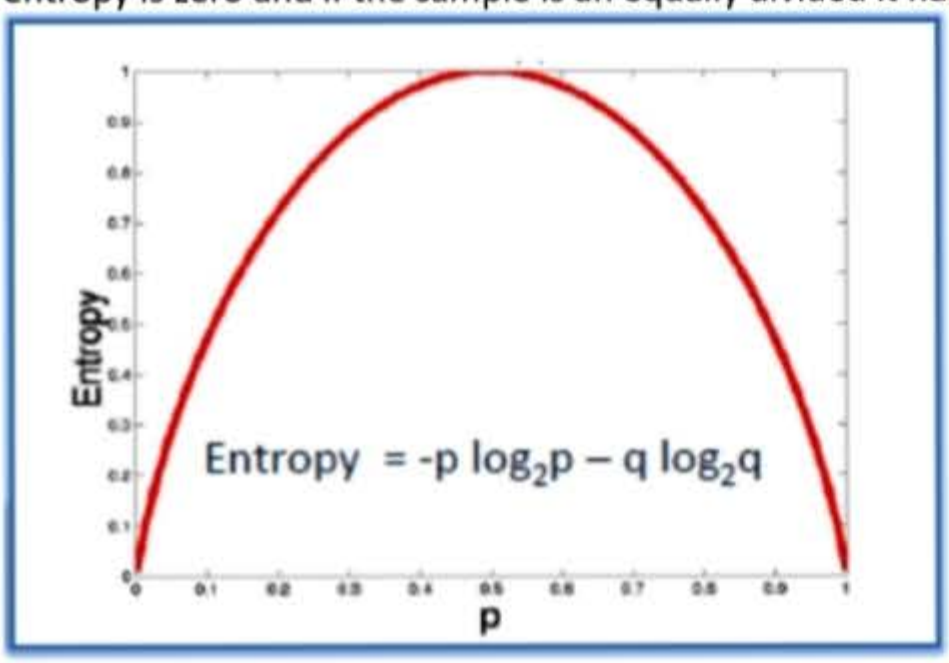
Predictors				Target
Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Question 14: What are Entropy and Information gain in Decision tree algorithm?

The core algorithm for building a decision tree is called **ID3**. **ID3** uses **Entropy** and **Information Gain*to construct a decision tree.***

Entropy

A decision tree is built top-down from a root node and involve partitioning of data into homogenious subsets. **ID3** uses enteropy to check the homogeneity of a sample. If the sample is completely homogenious then entropy is zero and if the sample is an equally divided it has entropy of one.



$$\text{Entropy} = -0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$$

Information Gain

The **Information Gain** is based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding attributes that return the highest information gain.

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1
Gain = 0.029			

		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1
Gain = 0.152			

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3
Gain = 0.048			

$$Gain(T,X) = Entropy(T) - Entropy(T,X)$$

G(PlayGolf, Outlook) = **E**(PlayGolf) – **E**(PlayGolf, Outlook)

= 0.940 – 0.693 = 0.247

Question 14: What is pruning in Decision Tree?

Pruning is a technique in machine learning and search algorithms that reduces the size of **decision trees** by removing sections of the **tree** that provide little power to classify instances. So, when we remove sub-nodes of a decision node, this process is called ***pruning***or opposite process of splitting.

Question 15: What is logistic regression? State an example when you have used logistic regression recently.

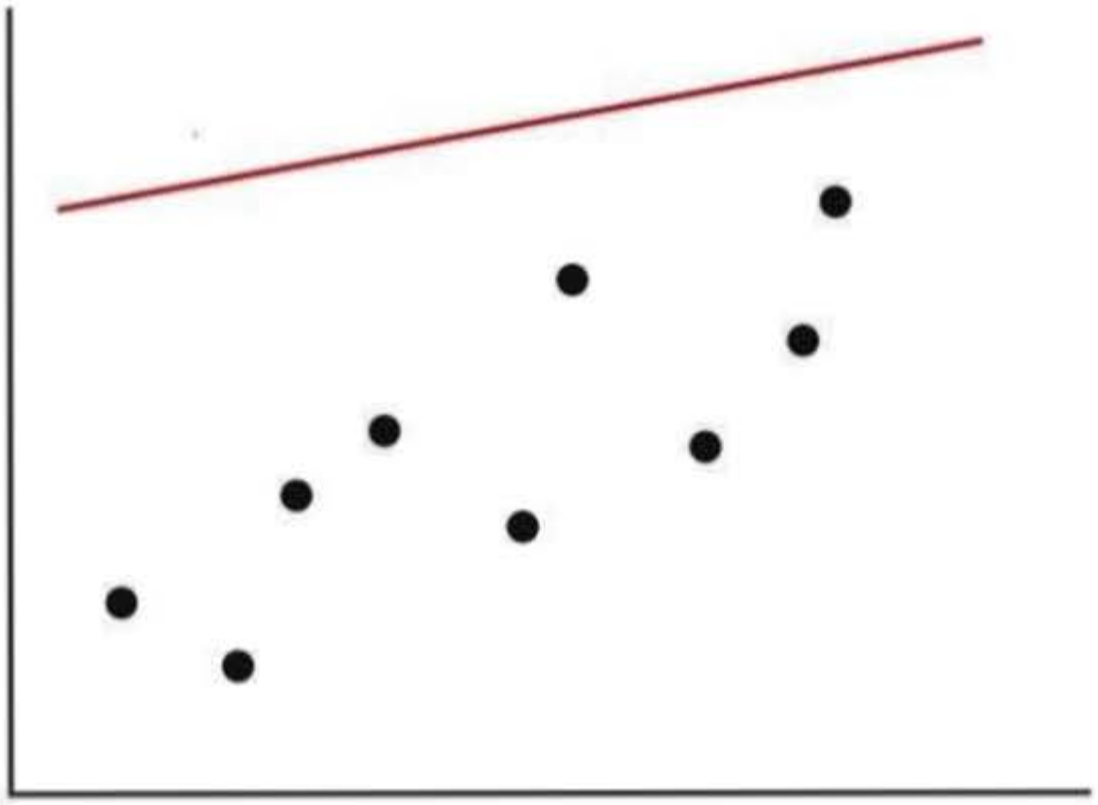
Logistic Regression often referred to as the logit model is a technique to predict the binary outcome from a linear combination of predictor variables.

For example, if you want to predict whether a particular political leader will win the election or not. In this case, the outcome of prediction is binary i.e. 0 or 1 (Win/Lose). The predictor variables here would be the amount of money spent for election campaigning of a particular candidate, the amount of time spent in campaigning, etc.

Question 16: What is Linear Regression?

Linear regression is a statistical technique where the score of a variable Y is predicted from the score of a second variable X. X is referred to as the predictor variable and Y as the criterion variable.

Machine Learning Algorithm
Regression - Alternating Least Squares



D =

The regression line is the one with the least value of D

Question 17: What Are the Drawbacks of the Linear Model?

Some drawbacks of the linear model are:

The assumption of linearity of the errors.

It can't be used for count outcomes or binary outcomes

There are overfitting problems that it can't solve

Question 18: What is the difference between Regression and classification ML techniques?

Both Regression and classification machine learning techniques come under **Supervised machine learning algorithms**. In Supervised machine learning algorithm, we have to train the model using labelled data set, While training we have to explicitly provide the correct labels and algorithm tries to learn the pattern from input to output. If our labels are discrete values then it will a classification problem, e.g A,B etc. but if our labels are continuous values then it will be a regression problem, e.g 1.23, 1.333 etc.

Question 19: What are Recommender Systems?

Recommender Systems are a subclass of information filtering systems that are meant to predict the preferences or ratings that a user would give to a product. Recommender systems are widely used in movies, news, research articles, products, social tags, music, etc.

Examples include movie recommenders in IMDB, Netflix & BookMyShow, product recommenders in e-commerce sites like Amazon, eBay & Flipkart, YouTube video recommendations and game recommendations in Xbox.

Question 20: What is Collaborative filtering?

The process of filtering used by most of the recommender systems to find patterns or information by collaborating viewpoints, various data sources and multiple agents.

Movie	Alice	Bob	Carol	Dave
Shutter Island	4	3	5	1
Fight Club	5	4	4	2
Dark Knight	5	3	4	?
21	4	3	?	5
Home Alone	4	4	5	5

Figure: *Predicting the rating of Dave for Dark Knight and Carol for 21 using Collaborative Filtering*

An example of collaborative filtering can be to predict the rating of a particular user based on his/her ratings for other movies and others' ratings for all movies. This concept is widely used in recommending movies in IMDB, Netflix & BookMyShow, product recommenders in e-commerce sites like Amazon, eBay & Flipkart, YouTube video recommendations and game recommendations in Xbox.

Question 21: How can outlier values be treated?

Outlier values can be identified by using univariate or any other graphical analysis method. If the number of outlier values is few then they can be assessed individually but for a large number of outliers, the values can be substituted with either the 99th or the 1st percentile values.

All extreme values are not outlier values. The most common ways to treat outlier values

To change the value and bring it within a range.

To just remove the value.

Question 22: What are the various steps involved in an analytics project?

The following are the various ***steps involved in an analytics project***:

Understand the Business problem

Explore the data and become familiar with it.

Prepare the data for modelling by detecting outliers, treating missing values, transforming variables, etc.

After data preparation, start running the model, analyze the result and tweak the approach. This is an iterative step until the best possible outcome is achieved.

Validate the model using a new data set.

Start implementing the model and track the result to analyze the performance of the model over the period of time.

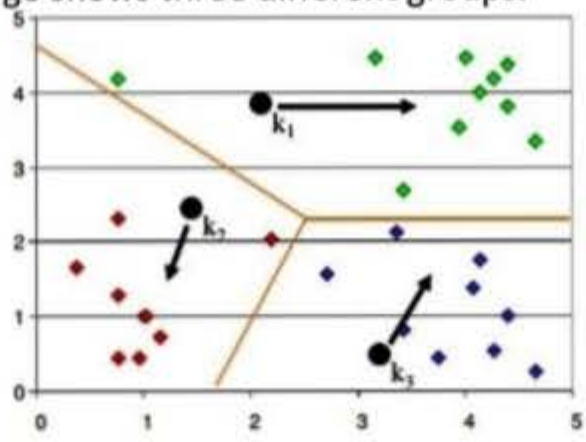
Question 23: During analysis, how do you treat missing values?

The extent of the missing values is identified after identifying the variables with missing values. If any patterns are identified the analyst has to concentrate on them as it could lead to interesting and meaningful business insights.

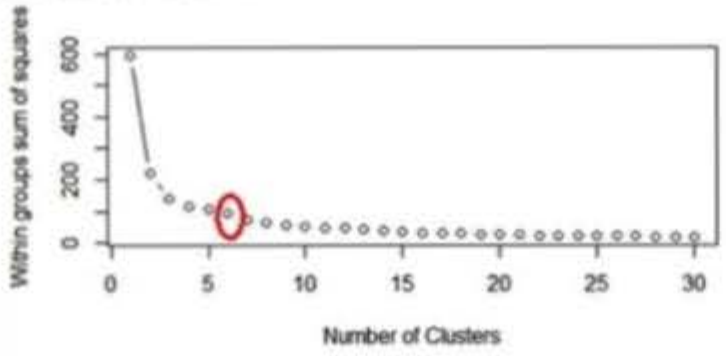
If there are no patterns identified, then the missing values can be substituted with mean or median values (imputation) or they can simply be ignored. Assigning a default value which can be mean, minimum or maximum value. Getting into the data is important.
If it is a categorical variable, the default value is assigned. The missing value is assigned a default value. If you have a distribution of data coming, for normal distribution give the mean value.
If 80% of the values for a variable are missing then you can answer that you would be dropping the variable instead of treating the missing values.

Question 24: How will you define the number of clusters in a clustering algorithm?

Though the Clustering Algorithm is not specified, this question is mostly in reference to ***K-Means clustering*** where "K" defines the number of clusters. The objective of clustering is to group similar entities in a way that the entities within a group are similar to each other but the groups are different from each other.
For example, the following image shows three different groups.



Within Sum of squares is generally used to explain the homogeneity within a cluster. If you plot WSS for a range of number of clusters, you will get the plot shown below.



The Graph is generally known as ***Elbow Curve***.
Red circled a point in above graph i.e. ***Number of Cluster =6***is the point after which you don't see any decrement in WSS.
This point is known as the **bending** point and taken as K in K – Means.
This is the widely used approach but few data scientists also use Hierarchical clustering first to create dendrograms and identify the distinct groups from there.

Question 25: What is Ensemble Learning?

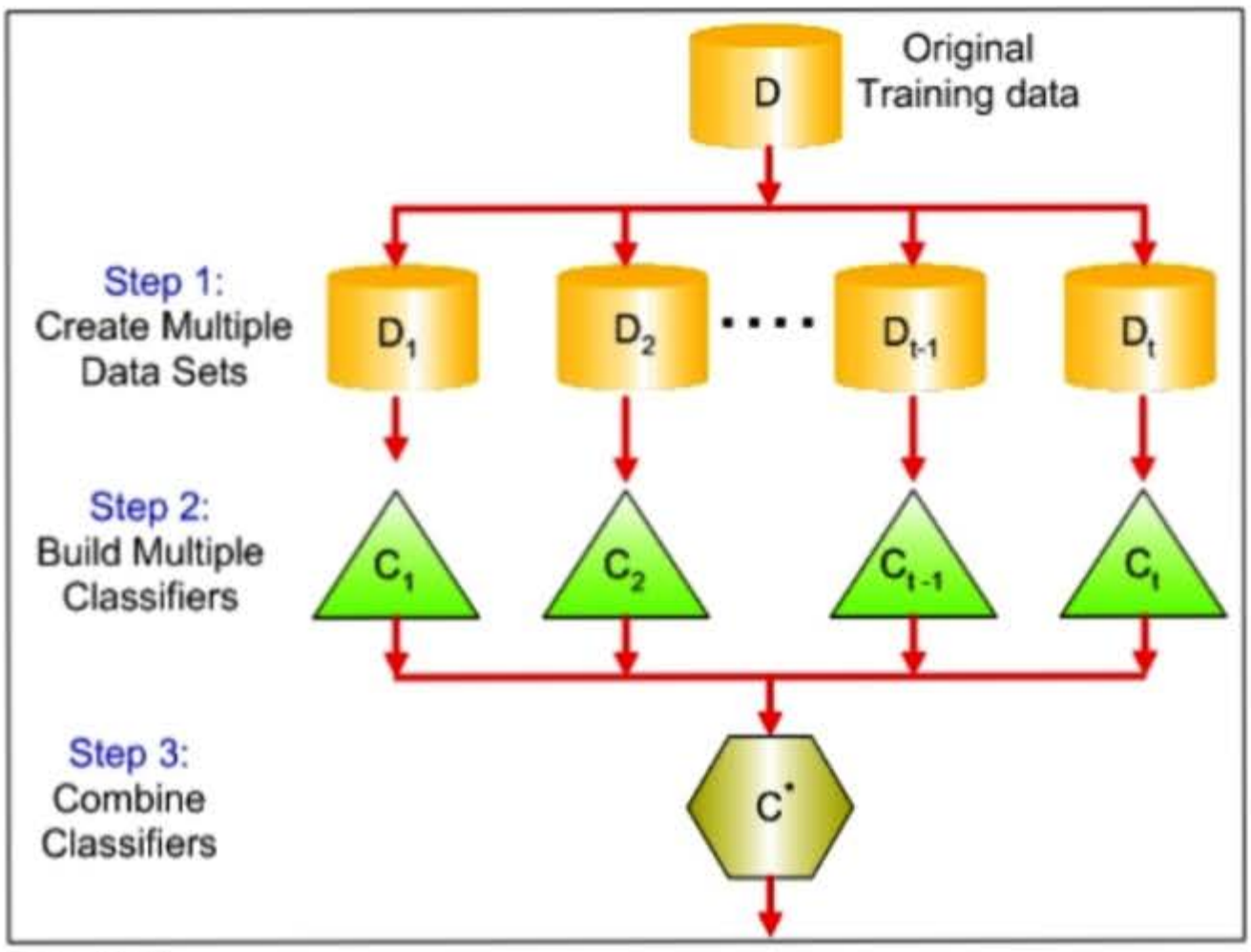
Ensemble Learning is basically combining a diverse set of learners(Individual models) together to improvise on the stability and predictive power of the model.

Question 26: Describe in brief any type of Ensemble Learning?

Ensemble learning has many types but two more popular ensemble learning techniques are mentioned below.

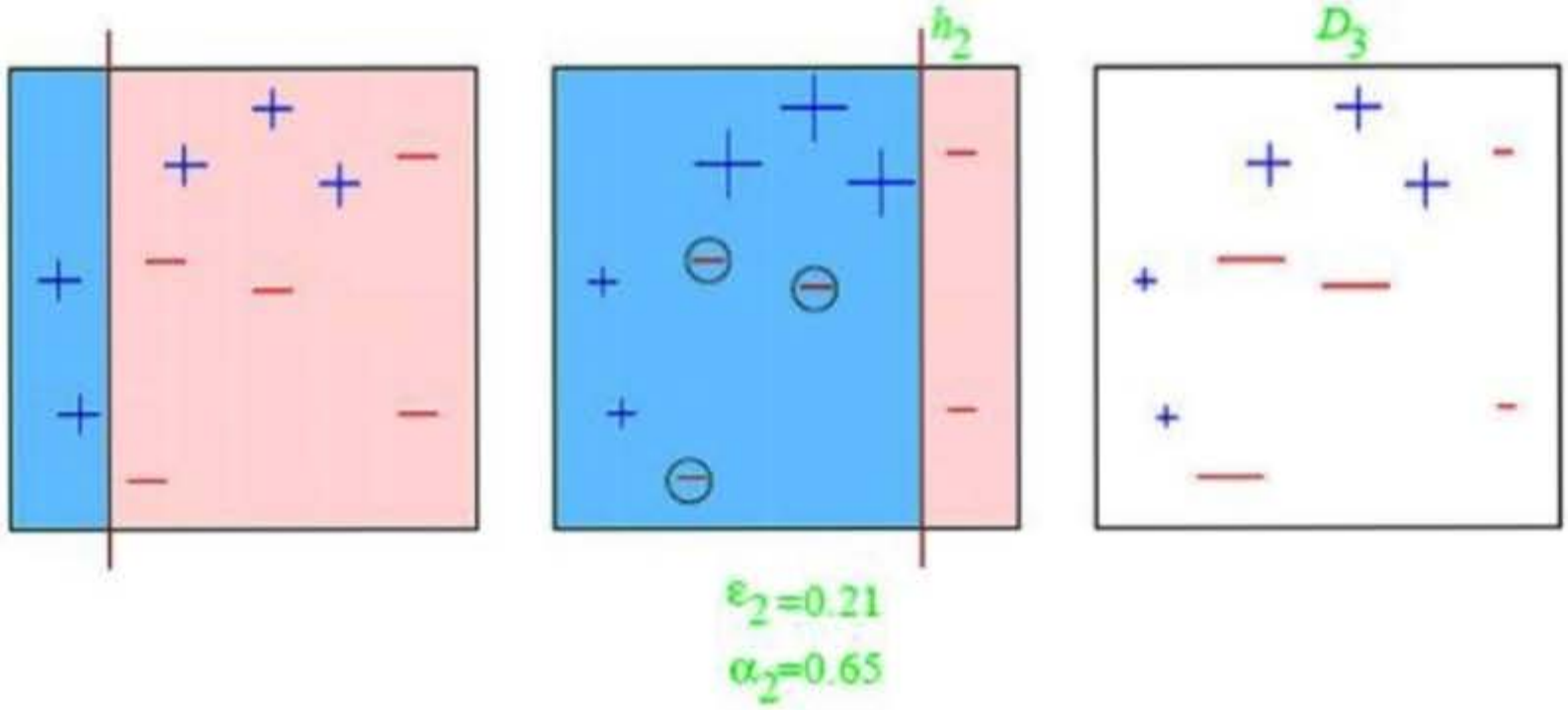
Bagging

Bagging tries to implement similar learners on small sample populations and then takes a mean of all the predictions. In generalised bagging, you can use different learners on different population. As you expect this helps us to reduce the variance error.



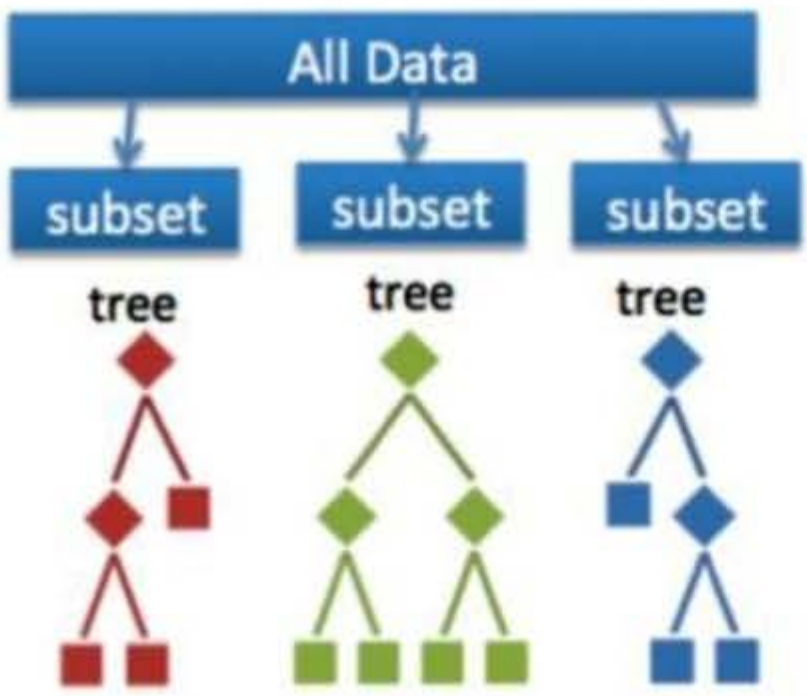
Boosting

Boosting is an iterative technique which adjusts the weight of an observation based on the last classification. If an observation was classified incorrectly, it tries to increase the weight of this observation and vice versa. Boosting in general decreases the bias error and builds strong predictive models. However, they may over fit on the training data.



Question 27: What is a Random Forest? How does it work?

Random forest is a versatile machine learning method capable of performing both regression and classification tasks. It is also used for dimensionality reduction, treats missing values, outlier values. It is a type of ensemble learning method, where a group of weak models combine to form a powerful model.



In Random Forest, we grow multiple trees as opposed to a single tree. To classify a new object based on attributes, each tree gives a classification. The forest chooses the classification having the most **votes**(Overall the trees in the forest) and in case of regression, it takes the average of outputs by different trees.

Question 28: How Do You Work Towards a Random Forest?

The underlying principle of this technique is that several weak learners combined to provide a keen learner. The steps involved are

Build several decision trees on bootstrapped training samples of data

On each tree, each time a split is considered, a random sample of mm predictors is chosen as split candidates, out of all pp predictors

Rule of thumb: At each split m=pVm=p

Predictions: At the majority rule

Question 29: How Regularly Must an Algorithm be Updated?

You will want to update an algorithm when:

You want the model to evolve as data streams through infrastructure

The underlying data source is changing

There is a case of non-stationarity

The algorithm underperforms/ results lack accuracy

Question 30: If you are having 4GB RAM in your machine and you want to train your model on 10GB data set. How would you go about this problem? Have you ever faced this kind of problem in your machine learning/data science experience so far?

First of all, you have to ask which ML model you want to train.

For Neural networks: Batch size with Numpy array will work.

Steps:

You can pass an index to Numpy array to get required data.

Use this data to pass to the Neural network.

Have a small batch size.

For SVM: Partial fit will work

Steps:

Divide one big data set in small size data sets.

Use a partial fit method of SVM, it requires a subset of the complete data set.

Repeat step 2 for other subsets.

Question 31: How is k-NN different from k-means clustering?

k-NN, or k-nearest neighbors is a classification algorithm, where the k is an integer describing the number of neighboring data points that influence the classification of a given observation. K-means is a clustering algorithm, where the k is an integer describing the number of clusters to be created from the given data.

Question 32: Explain the 80/20 rule, and tell me about its importance in model validation.
"People usually tend to start with a 80-20% split (80% training set – 20% test set) and split the training set once more into a 80-20% ratio to create the validation set."

Question 33: Explain what precision and recall are. How do they relate to the ROC curve?
Recall describes what percentage of true positives are described as positive by the model. Precision describes what percent of positive predictions were correct. The ROC curve shows the relationship between model recall and specificity–specificity being a measure of the percent of true negatives being described as negative by the model. Recall, precision, and the ROC are measures used to identify how useful a given classification model is.
Question 34: **You are given a train data set having 1000 columns and 1 million rows. The data set is based on a classification problem. Your manager has asked you to reduce the dimension of this data so that model computation time can be reduced. Your machine has memory constraints. What would you do? (You are free to make practical assumptions.)**
Answer: Processing a high dimensional data on a limited memory machine is a strenuous task, your interviewer would be fully aware of that. Following are the methods you can use to tackle such situation:
Since we have lower RAM, we should close all other applications in our machine, including the web browser, so that most of the memory can be put to use.
We can randomly sample the data set. This means, we can create a smaller data set, let's say, having 1000 variables and 300000 rows and do the computations.
To reduce dimensionality, we can separate the numerical and categorical variables and remove the correlated variables. For numerical variables, we'll use correlation. For categorical variables, we'll use chi-square test.
Also, we can use PCA and pick the components which can explain the maximum variance in the data set.
Using online learning algorithms like Vowpal Wabbit (available in Python) is a possible option.
Building a linear model using Stochastic Gradient Descent is also helpful.
We can also apply our business understanding to estimate which all predictors can impact the response variable. But, this is an intuitive approach, failing to identify useful predictors might result in significant loss of information.
Note: For point 4 & 5, make sure you read about online learning algorithms & Stochastic Gradient Descent. These are advanced methods.

Question 35: **Is rotation necessary in PCA? If yes, Why? What will happen if you don't rotate the components?**
Answer: Yes, rotation (orthogonal) is necessary because it maximizes the difference between variance captured by the component. This makes the components easier to interpret. Not to forget, that's the motive of doing PCA where, we aim to select fewer components (than features) which can explain the maximum variance in the data set. By doing rotation, the relative location of the components doesn't change, it only changes the actual coordinates of the points.
If we don't rotate the components, the effect of PCA will diminish and we'll have to select more number of components to explain variance in the data set.

Question 36: **You are given a data set. The data set has missing values which spread along 1 standard deviation from the median. What percentage of data would remain unaffected? Why?**
Answer: This question has enough hints for you to start thinking! Since, the data is spread across median, let's assume it's a normal distribution. We know, in a normal distribution, ~68% of the data lies in 1 standard deviation from mean (or mode, median), which leaves ~32% of the data unaffected. Therefore, ~32% of the data would remain unaffected by missing values.

Question 37: **You are given a data set on cancer detection. You've build a classification model and achieved an accuracy of 96%. Why shouldn't you be happy with your model performance? What can you do about it?**

Answer: If you have worked on enough data sets, you should deduce that cancer detection results in imbalanced data. In an imbalanced data set, accuracy should not be used as a measure of performance because 96% (as given) might only be predicting majority class correctly, but our class of interest is minority class (4%) which is the people who actually got diagnosed with cancer. Hence, in order to evaluate model performance, we should use Sensitivity (True Positive Rate), Specificity (True Negative Rate), F measure to determine class wise performance of the classifier. If the minority class performance is found to to be poor, we can undertake the following steps:

We can use undersampling, oversampling or SMOTE to make the data balanced.

We can alter the prediction threshold value by doing probability caliberation and finding a optimal threshold using AUC-ROC curve.

We can assign weight to classes such that the minority classes gets larger weight.

We can also use anomaly detection.

Question 38: **You are assigned a new project which involves helping a food delivery company save more money. The problem is, company's delivery team aren't able to deliver food on time. As a result, their customers get unhappy. And, to keep them happy, they end up delivering food for free. Which machine learning algorithm can save them?**

Answer: You might have started hopping through the list of ML algorithms in your mind. But, wait! Such questions are asked to test your machine learning fundamentals.

This is not a machine learning problem. This is a route optimization problem. A machine learning problem consist of three things:

There exist a pattern.

You cannot solve it mathematically (even by writing exponential equations).

You have data on it.

Always look for these three factors to decide if machine learning is a tool to solve a particular problem.

Question 39: **You came to know that your model is suffering from low bias and high variance. Which algorithm should you use to tackle it? Why?**

Answer: Low bias occurs when the model's predicted values are near to actual values. In other words, the model becomes flexible enough to mimic the training data distribution. While it sounds like great achievement, but not to forget, a flexible model has no generalization capabilities. It means, when this model is tested on an unseen data, it gives disappointing results.

In such situations, we can use bagging algorithm (like random forest) to tackle high variance problem. Bagging algorithms divides a data set into subsets made with repeated randomized sampling. Then, these samples are used to generate a set of models using a single learning algorithm. Later, the model predictions are combined using voting (classification) or averaging (regression).

Also, to combat high variance, we can:

Use regularization technique, where higher model coefficients get penalized, hence lowering model complexity.

Use top n features from variable importance chart. May be, with all the variable in the data set, the algorithm is having difficulty in finding the meaningful signal.

Question 40: **You are given a data set. The data set contains many variables, some of which are highly correlated and you know about it. Your manager has asked you to run PCA. Would you remove correlated variables first? Why?**

Answer: Chances are, you might be tempted to say No, but that would be incorrect. Discarding correlated variables have a substantial effect on PCA because, in presence of correlated variables, the variance explained by a particular component gets inflated.

For example: You have 3 variables in a data set, of which 2 are correlated. If you run PCA on this data set, the first principal component would exhibit twice the variance than it would exhibit with uncorrelated variables. Also, adding correlated variables lets PCA put more importance on those variable, which is misleading.

Question 41: **After spending several hours, you are now anxious to build a high accuracy model. As a result, you build 5 GBM models, thinking a boosting algorithm would do the magic. Unfortunately, neither of models could perform better than benchmark score. Finally, you decided to combine those models. Though, ensembled models are known to return high accuracy, but you are unfortunate. Where did you miss?**

Answer: As we know, ensemble learners are based on the idea of combining weak learners to create strong learners. But, these learners provide superior result when the combined models are uncorrelated. Since, we have used 5 GBM models and got no accuracy improvement, suggests that the models are correlated. The problem with correlated models is, all the models provide same information.

For example: If model 1 has classified User1122 as 1, there are high chances model 2 and model 3 would have done the same, even if its actual value is 0. Therefore, ensemble learners are built on the premise of combining weak uncorrelated models to obtain better predictions.

Question 42: **How is True Positive Rate and Recall related? Write the equation.**

Answer: True Positive Rate = Recall. Yes, they are equal having the formula (TP/TP + FN).

Question 43: **You have built a multiple regression model. Your model R^2 isn't as good as you wanted. For improvement, your remove the intercept term, your model R^2 becomes 0.8 from 0.3. Is it possible? How?**

Answer: Yes, it is possible. We need to understand the significance of intercept term in a regression model. The intercept term shows model prediction without any independent variable i.e. mean prediction. The formula of $R^2 = 1 - \sum(y - y')^2/\sum(y - ymean)^2$ where y' is predicted value.

When intercept term is present, R^2 value evaluates your model wrt. to the mean model. In absence of intercept term (ymean), the model can make no such evaluation, with large denominator, $\sum(y - y')^2/\sum(y)^2$ equation's value becomes smaller than actual, resulting in higher R^2.

Question 44: **When is Ridge regression favorable over Lasso regression?**

Answer: You can quote ISLR's authors Hastie, Tibshirani who asserted that, in presence of few variables with medium / large sized effect, use lasso regression. In presence of many variables with small / medium sized effect, use ridge regression.

Conceptually, we can say, lasso regression (L1) does both variable selection and parameter shrinkage, whereas Ridge regression only does parameter shrinkage and end up including all the coefficients in the model. In presence of correlated variables, ridge regression might be the preferred choice. Also, ridge regression works best in situations where the least square estimates have higher variance. Therefore, it depends on our model objective.

Question 45: **While working on a data set, how do you select important variables? Explain your methods.**

Answer: Following are the methods of variable selection you can use:

Remove the correlated variables prior to selecting important variables

Use linear regression and select variables based on p values

Use Forward Selection, Backward Selection, Stepwise Selection

Use Random Forest, Xgboost and plot variable importance chart

Use Lasso Regression

Measure information gain for the available set of features and select top n features accordingly.

Question 46: Explain the difference between L1 and L2 regularization methods.

"A regression model that uses L1 regularization technique is called Lasso Regression and model which uses L2 is called Ridge Regression. The key difference between these two is the penalty term."

Question 47: **Both being tree based algorithm, how is random forest different from Gradient boosting algorithm (GBM)?**

Answer: The fundamental difference is, random forest uses bagging technique to make predictions. GBM uses boosting techniques to make predictions.
In bagging technique, a data set is divided into n samples using randomized sampling. Then, using a single learning algorithm a model is build on all samples. Later, the resultant predictions are combined using voting or averaging. Bagging is done is parallel. In boosting, after the first round of predictions, the algorithm weighs misclassified predictions higher, such that they can be corrected in the succeeding round. This sequential process of giving higher weights to misclassified predictions continue until a stopping criterion is reached.
Random forest improves model accuracy by reducing variance (mainly). The trees grown are uncorrelated to maximize the decrease in variance. On the other hand, GBM improves accuracy my reducing both bias and variance in a model.

Question 48: **Running a binary classification tree algorithm is the easy part. Do you know how does a tree splitting takes place i.e. how does the tree decide which variable to split at the root node and succeeding nodes?**
Answer: A classification trees makes decision based on Gini Index and Node Entropy. In simple words, the tree algorithm find the best possible feature which can divide the data set into purest possible children nodes.
Gini index says, if we select two items from a population at random then they must be of same class and probability for this is 1 if population is pure. We can calculate Gini as following:
Calculate Gini for sub-nodes, using formula sum of square of probability for success and failure (p^2+q^2).
Calculate Gini for split using weighted Gini score of each node of that split
Entropy is the measure of impurity as given by (for binary class):

$$\text{Entropy} = -p \log_2 p - q \log_2 q$$

Here p and q is probability of success and failure respectively in that node. Entropy is zero when a node is homogeneous. It is maximum when a both the classes are present in a node at 50% – 50%. Lower entropy is desirable.

Question 49: **You've built a random forest model with 10000 trees. You got delighted after getting training error as 0.00. But, the validation error is 34.23. What is going on? Haven't you trained your model perfectly?**
Answer: The model has overfitted. Training error 0.00 means the classifier has mimiced the training data patterns to an extent, that they are not available in the unseen data. Hence, when this classifier was run on unseen sample, it couldn't find those patterns and returned prediction with higher error. In random forest, it happens when we use larger number of trees than necessary. Hence, to avoid these situation, we should tune number of trees using cross validation.

Question 50: **We know that one hot encoding increasing the dimensionality of a data set. But, label encoding doesn't. How ?**
Answer: Don't get baffled at this question. It's a simple question asking the difference between the two.
Using one hot encoding, the dimensionality (a.k.a features) in a data set get increased because it creates a new variable for each level present in categorical variables. For example: let's say we have a variable 'color'. The variable has 3 levels namely Red, Blue and Green. One hot encoding 'color' variable will generate three new variables as `Color.Red`, `Color.Blue` and `Color.Green` containing 0 and 1 value.
In label encoding, the levels of a categorical variables gets encoded as 0 and 1, so no new variable is created. Label encoding is majorly used for binary variables.

Question 51: **You are given a data set consisting of variables having more than 30% missing values? Let's say, out of 50 variables, 8 variables have missing values higher than 30%. How will you deal with them?**
Answer: We can deal with them in the following ways:
Assign a unique category to missing values, who knows the missing values might decipher some trend
We can remove them blatantly.
Or, we can sensibly check their distribution with the target variable, and if found any pattern we'll keep those missing values and assign them a new category while removing others.

Question 52: **'People who bought this, also bought...' recommendations seen on amazon is a result of which algorithm?**

Answer: The basic idea for this kind of recommendation engine comes from collaborative filtering.
Collaborative Filtering algorithm considers "User Behavior" for recommending items. They exploit behavior of other users and items in terms of transaction history, ratings, selection and purchase information. Other users behaviour and preferences over the items are used to recommend items to the new users. In this case, features of the items are not known.

Question 53: **What do you understand by Type I vs Type II error ?**

Answer: Type I error is committed when the null hypothesis is true and we reject it, also known as a 'False Positive'.
Type II error is committed when the null hypothesis is false and we accept it, also known as 'False Negative'.
In the context of confusion matrix, we can say Type I error occurs when we classify a value as positive (1) when it is actually negative (0). Type II error occurs when we classify a value as negative (0) when it is actually positive(1).

Question 54: **You are working on a classification problem. For validation purposes, you've randomly sampled the training data set into train and validation. You are confident that your model will work incredibly well on unseen data since your validation accuracy is high. However, you get shocked after getting poor test accuracy. What went wrong?**

Answer: In case of classification problem, we should always use stratified sampling instead of random sampling. A random sampling doesn't takes into consideration the proportion of target classes. On the contrary, stratified sampling helps to maintain the distribution of target variable in the resultant distributed samples also.

Question 55: **You have been asked to evaluate a regression model based on R^2, adjusted R^2 and tolerance. What will be your criteria?**

Answer: Tolerance (1 / VIF) is used as an indicator of multicollinearity. It is an indicator of percent of variance in a predictor which cannot be accounted by other predictors. Large values of tolerance is desirable.
We will consider adjusted R^2 as opposed to R^2 to evaluate model fit because R^2 increases irrespective of improvement in prediction accuracy as we add more variables. But, adjusted R^2 would only increase if an additional variable improves the accuracy of model, otherwise stays same. It is difficult to commit a general threshold value for adjusted R^2 because it varies between data sets. For example: a gene mutation data set might result in lower adjusted R^2 and still provide fairly good predictions, as compared to a stock market data where lower adjusted R^2 implies that model is not good.

Question 56: **I know that a linear regression model is generally evaluated using Adjusted R^2 or F value. How would you evaluate a logistic regression model?**

Answer: We can use the following methods:
Since logistic regression is used to predict probabilities, we can use AUC-ROC curve along with confusion matrix to determine its performance.
Also, the analogous metric of adjusted R^2 in logistic regression is AIC. AIC is the measure of fit which penalizes model for the number of model coefficients. Therefore, we always prefer model with minimum AIC value.
Null Deviance indicates the response predicted by a model with nothing but an intercept. Lower the value, better the model. Residual deviance indicates the response predicted by a model on adding independent variables. Lower the value, better the model.
Know more: Logistic Regression

Question 57: **Considering the long list of machine learning algorithm, given a data set, how do you decide which one to use?**

Answer: You should say, the choice of machine learning algorithm solely depends of the type of data. If you are given a data set which is exhibits linearity, then linear regression would be the best algorithm to use. If you given to work on images, audios, then neural network would help you to build a robust model.

If the data comprises of non linear interactions, then a boosting or bagging algorithm should be the choice. If the business requirement is to build a model which can be deployed, then we'll use regression or a decision tree model (easy to interpret and explain) instead of black box algorithms like SVM, GBM etc.
In short, there is no one master algorithm for all situations. We must be scrupulous enough to understand which algorithm to use.

Question 58: **Do you suggest that treating a categorical variable as continuous variable would result in a better predictive model?**
Answer: For better predictions, categorical variable can be considered as a continuous variable only when the variable is ordinal in nature.

Question 59: **When does regularization becomes necessary in Machine Learning?**
Answer: Regularization becomes necessary when the model begins to ovefit / underfit. This technique introduces a cost term for bringing in more features with the objective function. Hence, it tries to push the coefficients for many variables to zero and hence reduce cost term. This helps to reduce model complexity so that the model can become better at predicting (generalizing).

Question 60: **What do you understand by Bias Variance trade off?**
Answer: The error emerging from any model can be broken down into three components mathematically. Following are these component :

$$Err(x) = \left(E[\hat{f}\,(x)] - f(x)\right)^2 + E\left[\hat{f}\,(x) - E[\hat{f}\,(x)]\right]^2 + \sigma_e^2$$

$$Err(x) = \text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

Bias error is useful to quantify how much on an average are the predicted values different from the actual value. A high bias error means we have a under-performing model which keeps on missing important trends. **Variance** on the other side quantifies how are the prediction made on same observation different from each other. A high variance model will over-fit on your training population and perform badly on any observation beyond training.

Question 61: Explain what precision and recall are. How do they relate to the ROC curve?
Calculating precision and recall is actually quite easy. Imagine there are 100 positive cases among 10,000 cases. You want to predict which ones are positive, and you pick 200 to have a better chance of catching many of the 100 positive cases. You record the IDs of your predictions, and when you get the actual results you sum up how many times you were right or wrong. There are four ways of being right or wrong:
TN / True Negative: case was negative and predicted negative
TP / True Positive: case was positive and predicted positive
FN / False Negative: case was positive but predicted negative
FP / False Positive: case was negative but predicted positive
Makes sense so far? Now you count how many of the 10,000 cases fall in each bucket, say:

	Predicted Negative	**Predicted Positive**
Negative Cases	TN: 9,760	FP: 140
Positive Cases	FN: 40	TP: 60

Now, your boss asks you three questions:

What percent of your predictions were correct?
You answer: the "accuracy" was (9,760+60) out of 10,000 = 98.2%
What percent of the positive cases did you catch?
You answer: the "recall" was 60 out of 100 = 60%
What percent of positive predictions were correct?
You answer: the "precision" was 60 out of 200 = 30%

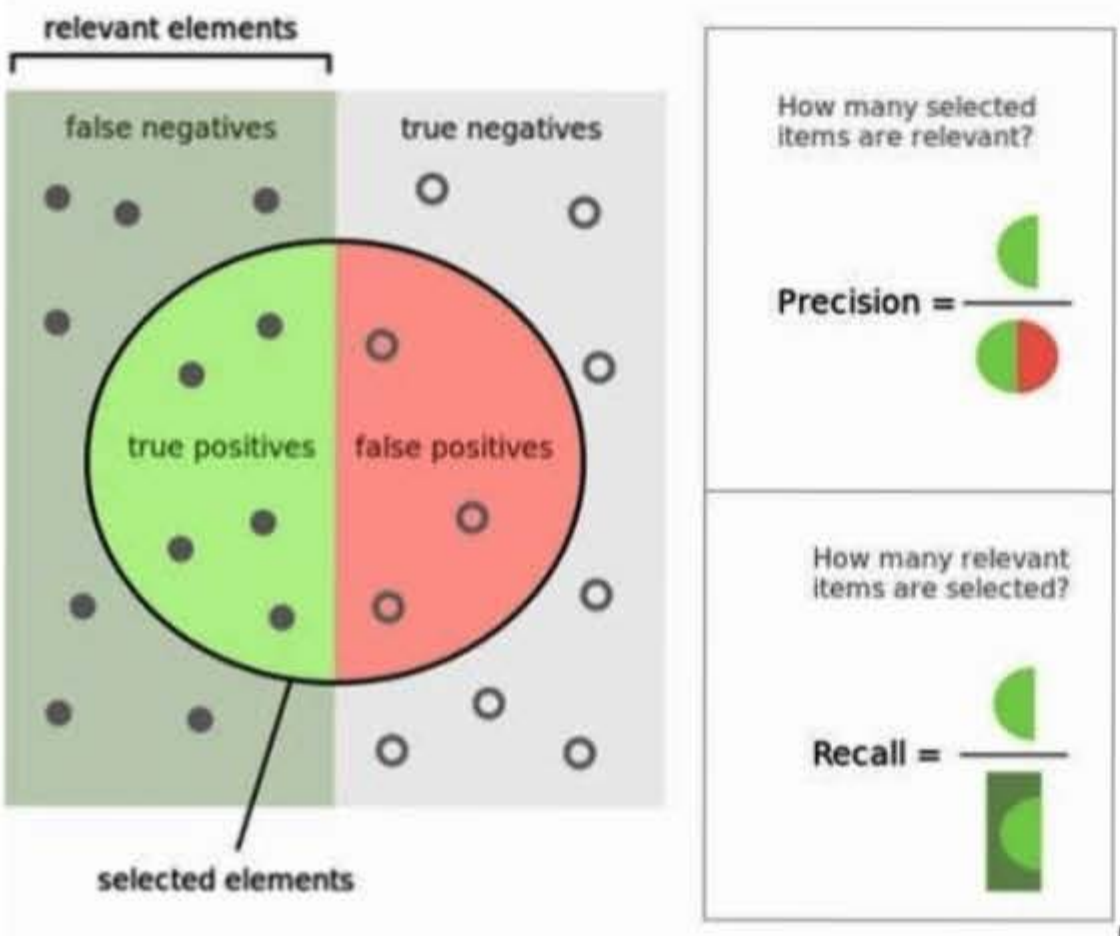


Fig 4: Precision and Recall.

ROC curve represents a relation between sensitivity (RECALL) and specificity(NOT PRECISION) and is commonly used to measure the performance of binary classifiers. However, when dealing with highly skewed datasets, Precision-Recall (PR) curves give a more representative picture of performance.

Question 62: How can you prove that one improvement you've brought to an algorithm is really an improvement over not doing anything?

Appealing insights that are confirmed without rigorous validation. One such scenario is the case that given the task of improving an algorithm to yield better results, you might come with several ideas with potential for improvement.

An obvious human urge is to announce these ideas ASAP and ask for their implementation. When asked for supporting data, often limited results are shared, which are very likely to be impacted by selection bias (known or unknown) or a misleading global minima (due to lack of appropriate variety in test data).

Data scientists do not let their human emotions overrun their logical reasoning. While the exact approach to prove that one improvement you've brought to an algorithm is really an improvement over not doing anything would depend on the actual case at hand, there are a few common guidelines:
Ensure that there is no selection bias in test data used for performance comparison
Ensure that the test data has sufficient variety in order to be symbolic of real-life data (helps avoid overfitting)
Ensure that "controlled experiment" principles are followed i.e. while comparing performance, the test environment (hardware, etc.) must be exactly the same while running original algorithm and new algorithm
Ensure that the results are repeatable with near similar results
Examine whether the results reflect local maxima/minima or global maxima/minima

One common way to achieve the above guidelines is through A/B testing, where both the versions of algorithm are kept running on similar environment for a considerably long time and real-life input data is randomly split between the two. This approach is particularly common in Web Analytics.

Question 63:Is it better to have too many false positives, or too many false negatives? Explain.

It depends on the question as well as on the domain for which we are trying to solve the question.

In medical testing, false negatives may provide a falsely reassuring message to patients and physicians that disease is absent, when it is actually present. This sometimes leads to inappropriate or inadequate treatment of both the patient and their disease. So, it is desired to have too many false positive.

For spam filtering, a false positive occurs when spam filtering or spam blocking techniques wrongly classify a legitimate email message as spam and, as a result, interferes with its delivery. While most anti-spam tactics can block or filter a high percentage of unwanted emails, doing so without creating significant false-positive results is a much more demanding task. So, we prefer too many false negatives over many false positives.

Question 64: In Python's standard library, what packages would you say are the most useful for data scientists?

Python wasn't built for data science. However, in recent years it has grown to become the go-to programming language for the following:
Machine learning
Predictive analytics
Simple data analytics
Statistics
For data science projects, the following packages in the Python standard library will make life easier and accelerate deliveries:
NumPy (to process large multidimensional arrays, extensive collections of high-level mathematical functions, and matrices)
Pandas (to leverage built-in methods for rapidly combining, filtering, and grouping data)
SciPy (to extend NumPy's capabilities and solve tasks related to integral calculus, linear algebra, and probability theory)

Question 65: Can you list some disadvantages related to linear models?

There are many disadvantages to using linear models, but the main ones are:
Errors in linearity assumptions
Lacks autocorrelation
It can't solve overfitting problems
You can't use it to calculate outcomes or binary outcomes

Question 66: What's a feature vector?

A feature vector is an n-dimensional vector that contains essential information that describes the characteristics of an object. For example, it can be an object's numerical features or a list of numbers taken from the output of a neural network layer.

In AI and data science, feature vectors can be used to represent numeric or symbolic characteristics of an object in mathematical terms for seamless analysis.
Let's break this down. A data set is usually organized into multiple examples where each example will have several features. However, a feature vector won't have the same feature for numerous examples. Instead, each example will correspond to one feature vector that will contain all the numerical values for that example object.
Feature vectors are often stacked into a design matrix. In this scenario, each row will be a feature vector for one example. Each column will feature all the examples that correspond to that particular feature. This means that it will be like a matrix, but with just one row and multiple columns (or a single column and multiple rows) like [1,2,3,5,6,3,2,0].

Question 67: What are the typical characteristics of elements in a list and a dictionary?

In lists, elements maintain their order unless they are explicitly commanded to re-order. These can be made up of any data type that can be all the same or mixed. However, elements in lists can only be accessed via numeric, zero-based indices.
In a dictionary, the order isn't guaranteed. However, each entry will be assigned a key and a value. As a result, elements within a dictionary can be accessed by using their individual key.
So whenever you have a set of unique keys, you have to use a dictionary. Whenever a collection of items are in order, you can use a list.
It's difficult to predict how an AI interview will unfold, so if they follow up by asking you how to get a list of all the keys in a dictionary, respond with the following:
To obtain a list of keys in a dictionary, you'll have to use the following function keys():

```
mydict={'a':1,'b':2,'c':3,'e':5}
mydict.keys()
dict_keys(['a', 'b', 'c', 'e'])
```

Question 68: What are the different algorithm techniques you can use in AI and ML?

Some algorithm techniques that can be leveraged are:
Learning to learn
Reinforcement learning (deep adversarial networks, q-learning, and temporal difference)
Semi-supervised learning
Supervised learning (decision trees, linear regression, naive bayes, nearest neighbor, neural networks, and support vector machines)
Transduction
Unsupervised learning (association rules and k-means clustering)

Question 69: How would you go about choosing an algorithm to solve a business problem?

First, you have to develop a "problem statement" that's based on the problem provided by the business. This step is essential because it'll help ensure that you fully understand the type of problem and the input and the output of the problem you want to solve.
The problem statement should be simple and no more than a single sentence. For example, let's consider enterprise spam that requires an algorithm to identify it.
The problem statement would be: **"Is the email fake/spam or not?"** In this scenario, the identification of whether it's fake/spam will be the output.
Once you have defined the problem statement, you have to identify the appropriate algorithm from the following:
Any classification algorithm
Any clustering algorithm
Any regression algorithm
Any recommendation algorithm
Which algorithm you use will depend on the specific problem you're trying to solve. In this scenario, you can move forward with a clustering algorithm and choose a k-means algorithm to achieve your goal of filtering spam from the email system.
While examples aren't always necessary when answering questions about artificial intelligence, sometimes it will help make it easier for you to get your point across.

Question 70: What steps would you take to evaluate the effectiveness of your ML model?

You have to first split the data set into training and test sets. You also have the option of using a cross-validation technique to further segment the data set into a composite of training and test sets within the data.
Then you have to implement a choice selection of the performance metrics like the following:
Confusion matrix
Accuracy
Precision
Recall or sensitivity
Specificity
F1 score
For the most part, you can use measures such as accuracy, confusion matrix, or F1 score. However, it'll be critical for you to demonstrate that you understand the nuances of how each model can be measured by choosing the right performance measure to match the problem.

Question 71: What would you do if data in a data set were missing or corrupted?

Whenever data is missing or corrupted, you either replace it with another value or drop those rows and columns altogether. In Pandas, both isNull() and dropNA() are handy tools to find missing or corrupted data and drop those values. You can also use the fillna() method to fill the invalid values in a placeholder—for example, "0."

Deep Learning Interview Questions

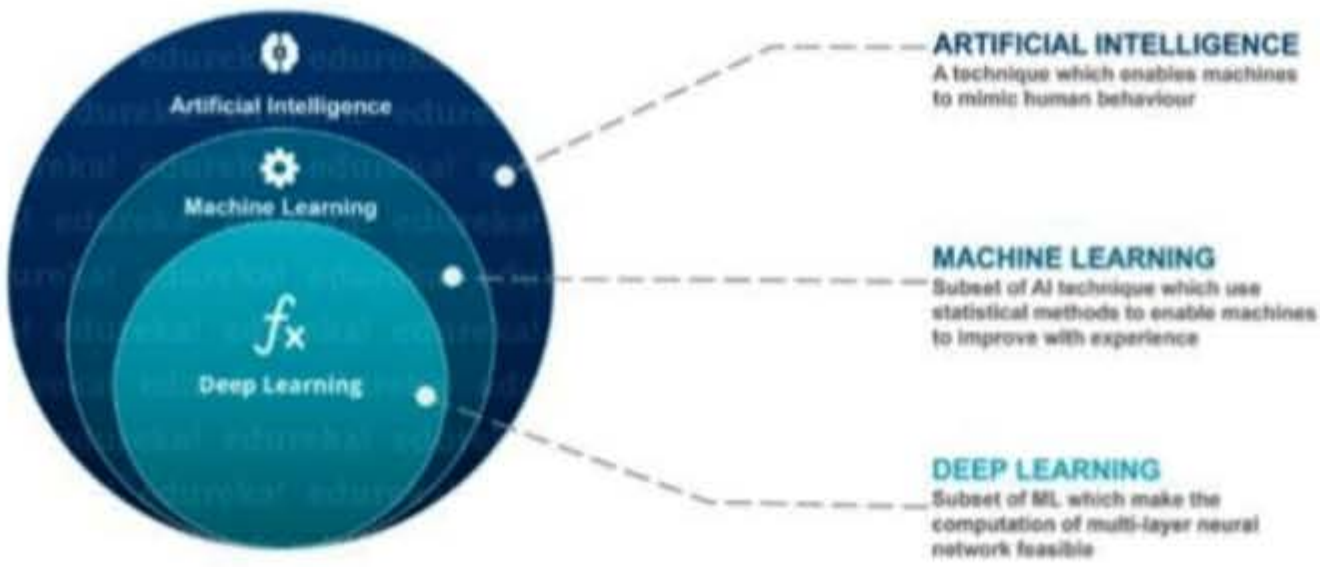
Deep Learning Interview Questions

Question 1: What do you mean by Deep Learning?

Deep Learning is nothing but a paradigm of machine learning which has shown incredible promise in recent years. This is because of the fact that Deep Learning shows a great analogy with the functioning of the human brain.

Question 2: What is the difference between machine learning and deep learning?

Machine learning is a field of computer science that gives computers the ability to learn without being explicitly programmed. Machine learning can be categorized in the following three categories.
Supervised machine learning,
Unsupervised machine learning,
Reinforcement learning



Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.

Question 3: What, in your opinion, is the reason for the popularity of Deep Learning in recent times?

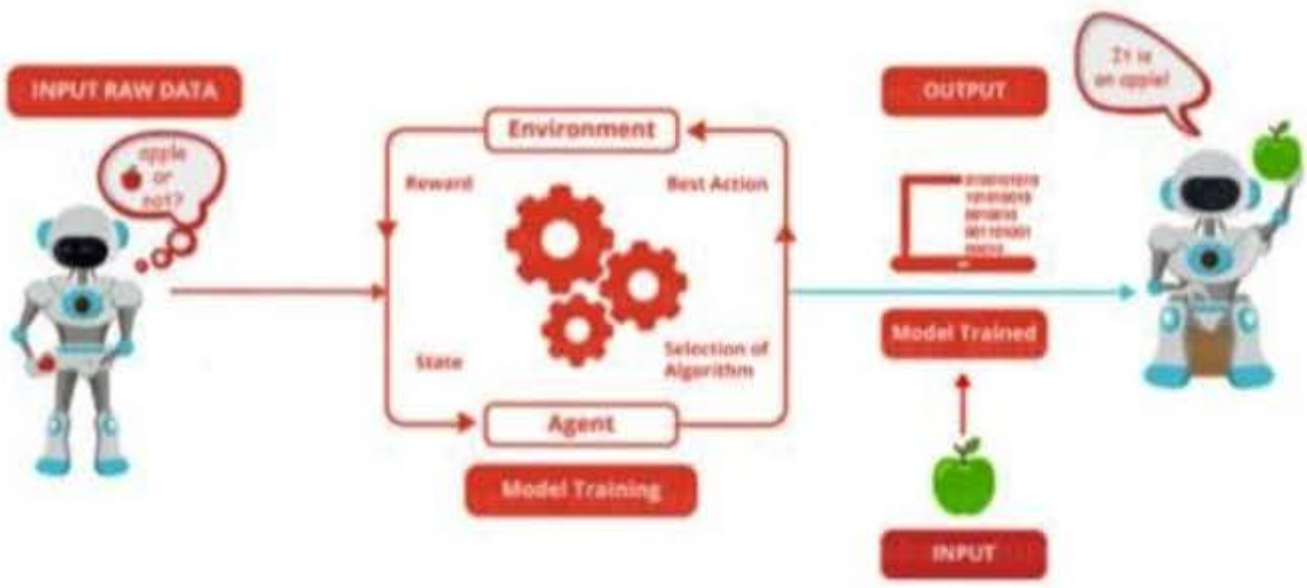
Now although Deep Learning has been around for many years, the major breakthroughs from these techniques came just in recent years. This is because of two main reasons:

The increase in the amount of data generated through various sources

The growth in hardware resources required to run these models

GPUs are multiple times faster and they help us build bigger and deeper deep learning models in comparatively less time than we required previously.

Question 4: What is reinforcement learning?



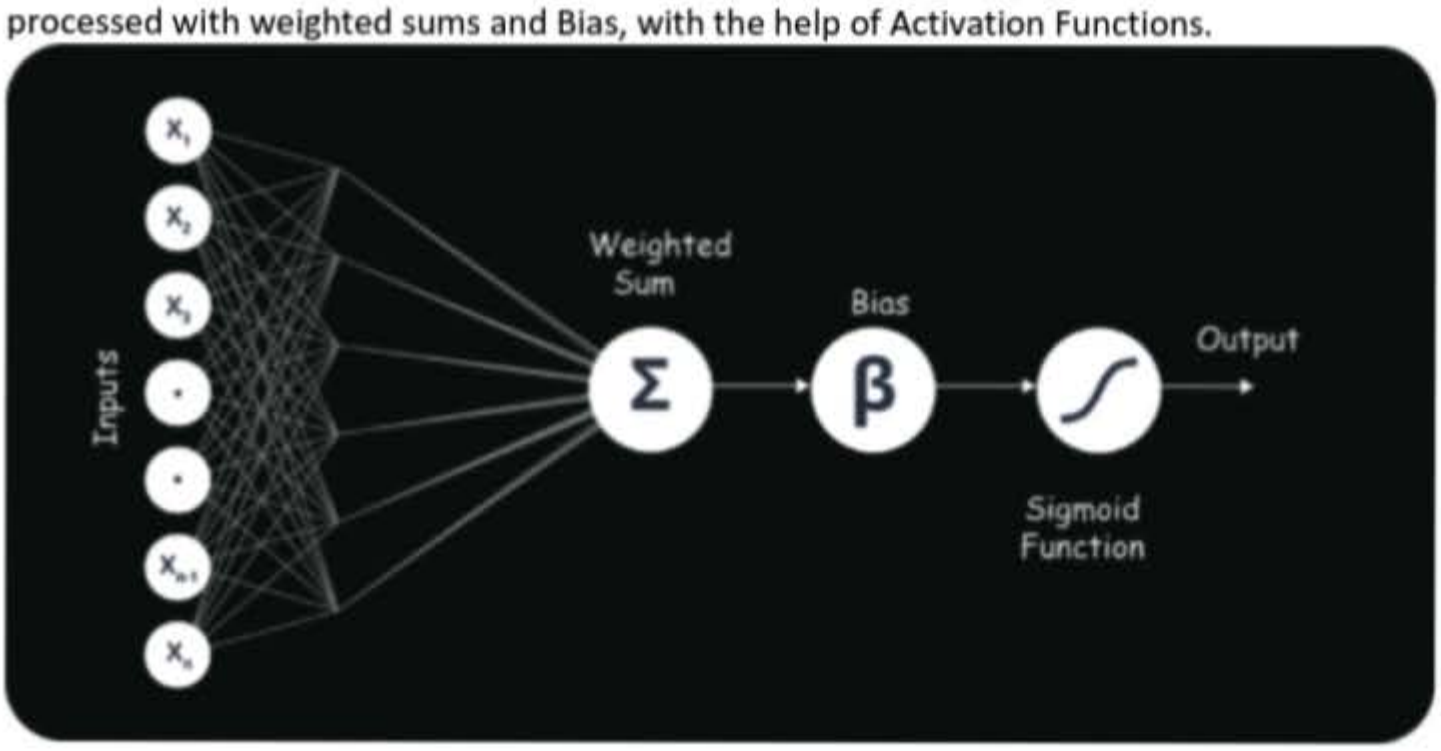
Reinforcement Learning is learning what to do and how to map situations to actions. The end result is to maximise the numerical reward signal. The learner is not told which action to take but instead must discover which action will yield the maximum reward. Reinforcement learning is inspired by the learning of human beings, it is based on the reward/penalty mechanism.

Question 5: What are Artificial Neural Networks?

Artificial Neural networks are a specific set of algorithms that have revolutionized machine learning. They are inspired by biological neural networks. **Neural Networks** can adapt to changing the input so the network generates the best possible result without needing to redesign the output criteria.

Question 6: Describe the structure of Artificial Neural Networks?

Artificial Neural Networks works on the same principle as a biological Neural Network. It consists of inputs that get processed with weighted sums and Bias, with the help of Activation Functions.



Question 7: How Are Weights Initialized in a Network?

There are two methods here: we can either initialize the weights to zero or assign them randomly.
Initializing all weights to 0: This makes your model similar to a linear model. All the neurons and every layer perform the same operation, giving the same output and making the deep net useless.
Initializing all weights randomly: Here, the weights are assigned randomly by initializing them very close to 0. It gives better accuracy to the model since every neuron performs different computations. This is the most commonly used method.

Question 8: What Is the Cost Function?

Also referred to as "loss" or "error," cost function is a measure to evaluate how good your model's performance is. It's used to compute the error of the output layer during backpropagation. We push that error backward through the neural network and use that during the different training functions.

Question 9: What Are Hyperparameters?

With neural networks, you're usually working with ***hyperparameters*** once the data is formatted correctly. A hyperparameter is a parameter whose value is set before the learning process begins. It determines how a network is trained and the structure of the network (such as the number of hidden units, the learning rate, epochs, etc.).

Question 10: What Will Happen If the Learning Rate Is Set inaccurately (Too Low or Too High)?

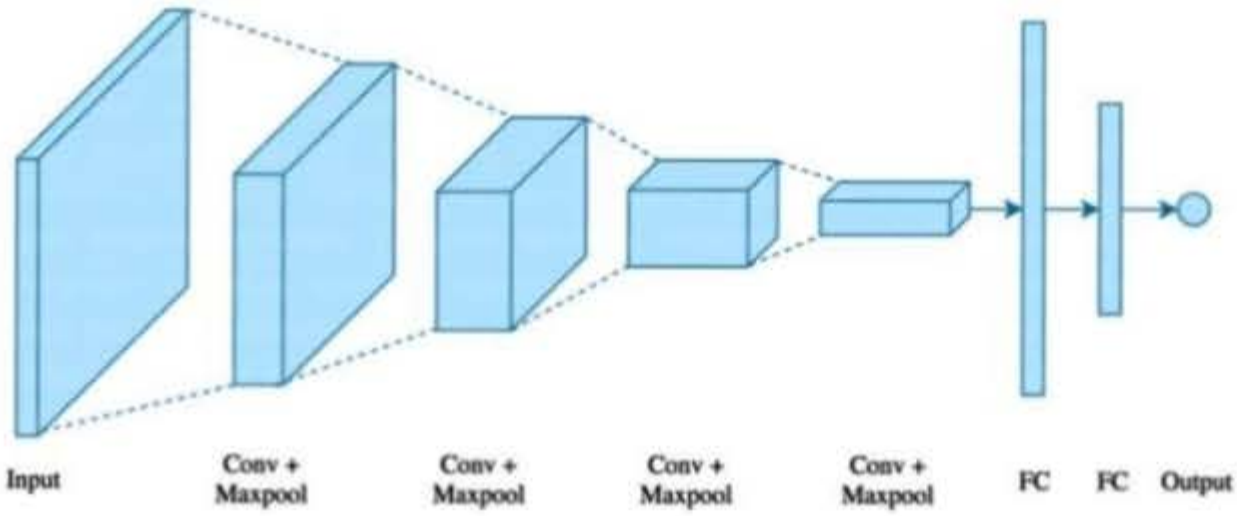
When your learning rate is too low, training of the model will progress very slowly as we are making minimal updates to the weights. It will take many updates before reaching the minimum point.
If the learning rate is set too high, this causes undesirable divergent behavior to the loss function due to drastic updates in weights. It may fail to converge (model can give a good output) or even diverge (data is too chaotic for the network to train).

Question 11: What Is the Difference Between Epoch, Batch, and Iteration in Deep Learning?

Epoch – Represents one iteration over the entire dataset (everything put into the training model).
Batch – Refers to when we cannot pass the entire dataset into the neural network at once, so we divide the dataset into several batches.
Iteration – if we have 10,000 images as data and a batch size of 200. then an epoch should run 50 iterations (10,000 divided by 50).

Question 12: What Are the Different Layers on CNN?

There are four layers in ***CNN***:
Convolutional Layer – the layer that performs a convolutional operation, creating several smaller picture windows to go over the data.
ReLU Layer – it brings non-linearity to the network and converts all the negative pixels to zero. The output is a rectified feature map.
Pooling Layer – pooling is a down-sampling operation that reduces the dimensionality of the feature map.
Fully Connected Layer – this layer recognizes and classifies the objects in the image.



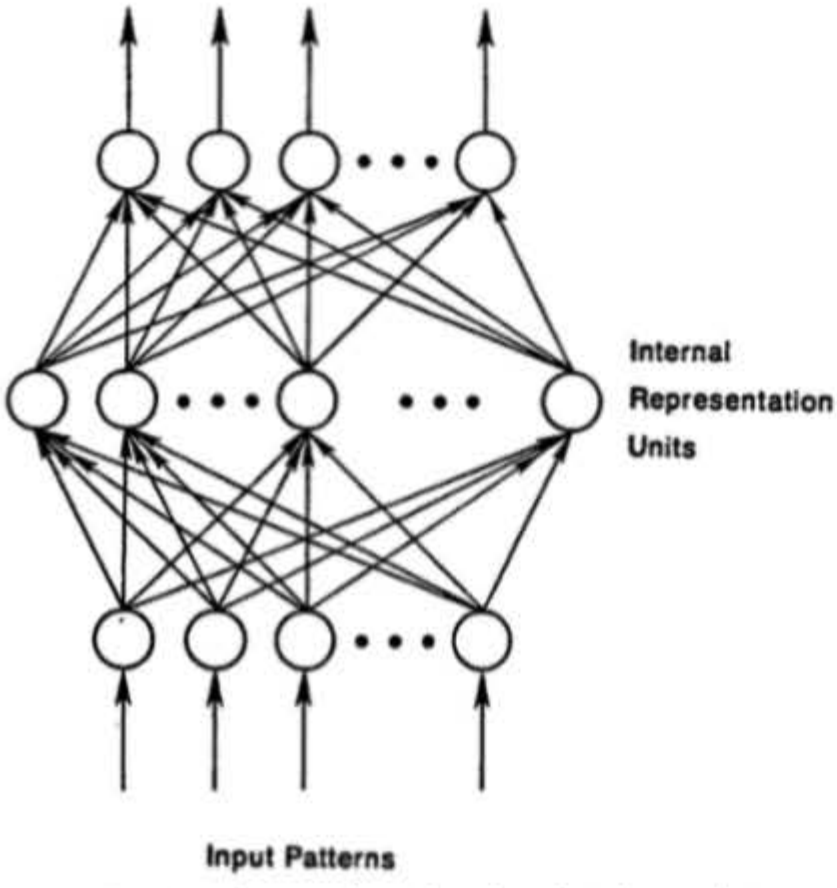
Q81. What Is Pooling on CNN, and How Does It Work?

Pooling is used to reduce the spatial dimensions of a CNN. It performs down-sampling operations to reduce the dimensionality and creates a pooled feature map by sliding a filter matrix over the input matrix.

Question 13: What are Recurrent Neural Networks(RNNs)?

RNNs are a type of artificial neural network designed to recognize the pattern from the sequence of data such as Time series, stock market, and government agencies, etc. To understand recurrent nets, first, you have to understand the basics of feedforward nets.

Both these networks RNN and feed-forward named after the way they channel information through a series of mathematical orations performed at the nodes of the network. One feeds information through straight(never touching the same node twice), while the other cycles it through a loop, and the latter are called recurrent.



Recurrent networks, on the other hand, take as their input, not just the current input example they see, but also what they have perceived previously in time.

The decision a recurrent neural network reached at time t-1 affects the decision that it will reach one moment later at time t. So recurrent networks have two sources of input, the present and the recent past, which combine to determine how they respond to new data, much as we do in life.

The error they generate will return via backpropagation and be used to adjust their weights until error can't go any lower. Remember, the purpose of recurrent nets is to accurately classify sequential input. We rely on the backpropagation of error and gradient descent to do so.

Question 14: How Does an LSTM Network Work?

Long-Short-Term Memory (LSTM) is a special kind of recurrent neural network capable of learning long-term dependencies, remembering information for long periods as its default behavior. There are three steps in an LSTM network:

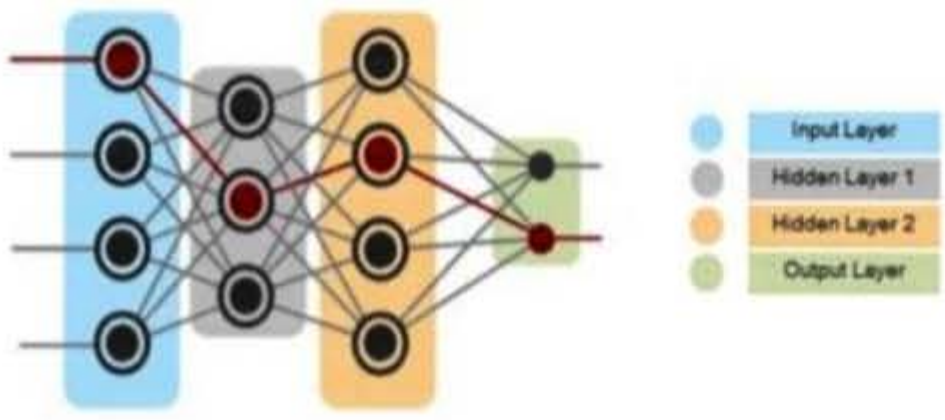
Step 1: The network decides what to forget and what to remember.

Step 2: It selectively updates cell state values.

Step 3: The network decides what part of the current state makes it to the output.

Question 15: What Is a Multi-layer Perceptron(MLP)?

As in ***Neural Networks, MLPs*** have an input layer, a hidden layer, and an output layer. It has the same structure as a single layer ***perceptron*** with one or more hidden layers. A single layer perceptron can classify only linear separable classes with binary output (0,1), but MLP can classify nonlinear classes.



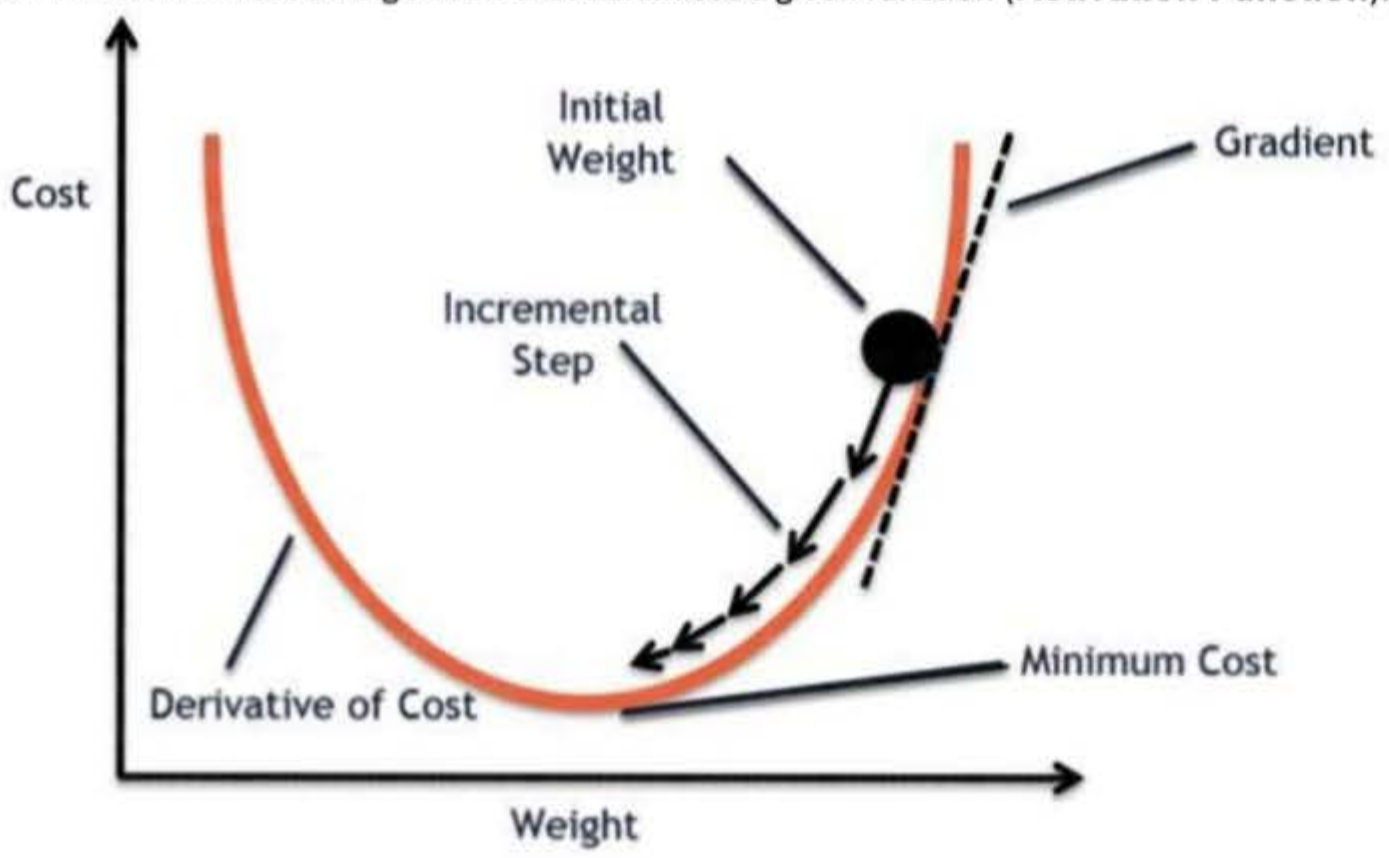
Except for the input layer, each node in the other layers uses a nonlinear activation function. This means the input layers, the data coming in, and the activation function is based upon all nodes and weights being added together, producing the output. MLP uses a supervised learning method called "backpropagation." In ***backpropagation***, the neural network calculates the error with the help of cost function. It propagates this error backward from where it came (adjusts the weights to train the model more accurately).

Question 16: Explain Gradient Descent.

To Understand Gradient Descent, Let's understand what is a ***Gradient*** first.

A **gradient** measures how much the output of a function changes if you change the inputs a little bit. It simply measures the change in all weights with regard to the change in error. You can also think of a gradient as the slope of a function.

Gradient Descent can be thought of climbing down to the bottom of a valley, instead of climbing up a hill. This is because it is a minimization algorithm that minimizes a given function (**Activation Function**).



Question 17: What is exploding gradients?

While training an RNN, if you see **exponentially growing (very large) error gradients** that accumulate and result in very large updates to neural network model weights during training, they're known as exploding gradients. At an extreme, the values of weights can become so large as to overflow and result in NaN values.
This has the effect of your model is unstable and unable to learn from your training data.

Question 18: What is vanishing gradients?

While training an RNN, your slope can become either too small; this makes the training difficult. When the slope is too small, the problem is known as a Vanishing Gradient. It leads to long training times, poor performance, and low accuracy.

Question 19: What is Back Propagation and Explain it's Working.

Backpropagation is a training algorithm used for multilayer neural networks. In this method, we move the error from an end of the network to all weights inside the network and thus allowing efficient computation of the gradient.
It has the following steps:
Forward Propagation of Training Data
Derivatives are computed using output and target
Back Propagate for computing derivative of error wrt output activation
Using previously calculated derivatives for output
Update the Weights

Question 20: What are the variants of Back Propagation?

Stochastic Gradient Descent: We use only a single training example for calculation of gradient and update parameters.
Batch Gradient Descent: We calculate the gradient for the whole dataset and perform the update at each iteration.
Mini-batch Gradient Descent: It's one of the most popular optimization algorithms. It's a variant of Stochastic Gradient Descent and here instead of a single training example, a mini-batch of samples is used.

Question 21: What are the different ***Deep Learning Frameworks***?

Pytorch
TensorFlow
Microsoft Cognitive Toolkit
Keras
Caffe
Chainer

Question 22: What is the role of the Activation Function?

The ***Activation function*** is used to introduce non-linearity into the neural network helping it to learn more complex function. Without which the neural network would be only able to learn linear function which is a linear combination of its input data. An activation function is a function in an artificial neuron that delivers an output based on inputs.

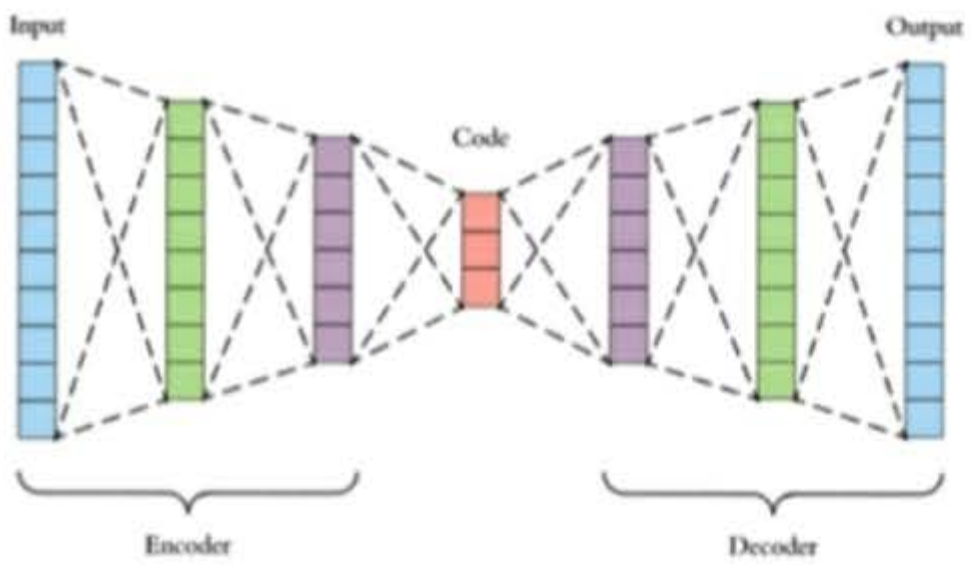
Question 23: Name a few ***Machine Learning libraries*** for various purposes.

Purpose	**Libraries**
Scientific Computation	Numpy
Tabular Data	Pandas
Data Modelling & Preprocessing	Scikit Learn

Time-Series Analysis	Statsmodels
Text processing	Regular Expressions, NLTK
Deep Learning	Tensorflow, Pytorch

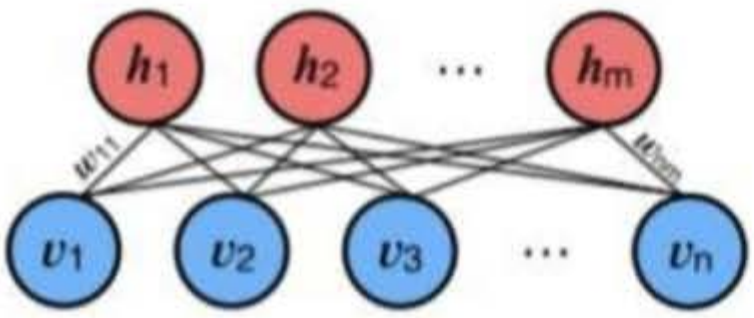
Question 24: What is an Auto-Encoder?

Auto-encoders are simple learning networks that aim to transform inputs into outputs with the minimum possible error. This means that we want the output to be as close to input as possible. We add a couple of layers between the input and the output, and the sizes of these layers are smaller than the input layer. The auto-encoder receives unlabelled input which is then encoded to reconstruct the input.



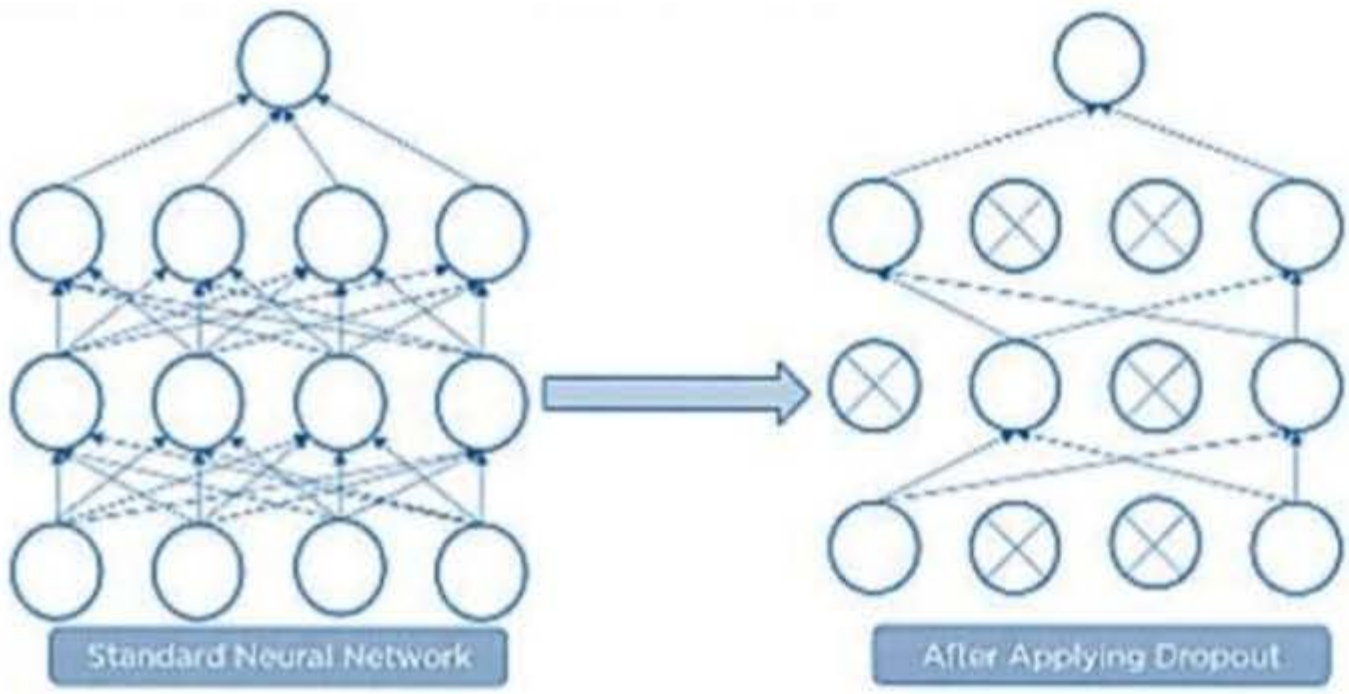
Q95. What is a Boltzmann Machine?

Boltzmann machines have a simple learning algorithm that allows them to discover interesting features that represent complex regularities in the training data. The Boltzmann machine is basically used to optimize the weights and the quantity for the given problem. The learning algorithm is very slow in networks with many layers of feature detectors. "**Restricted Boltzmann Machines**" algorithm has a single layer of feature detectors which makes it faster than the rest.



Q96. What Is Dropout and Batch Normalization?

Dropout is a technique of dropping out hidden and visible units of a network randomly to prevent overfitting of data (typically dropping 20 percent of the nodes). It doubles the number of iterations needed to converge the network.



Batch normalization is the technique to improve the performance and stability of neural networks by normalizing the inputs in every layer so that they have mean output activation of zero and standard deviation of one.

Question 25: What Is the Difference Between Batch Gradient Descent and Stochastic Gradient Descent?

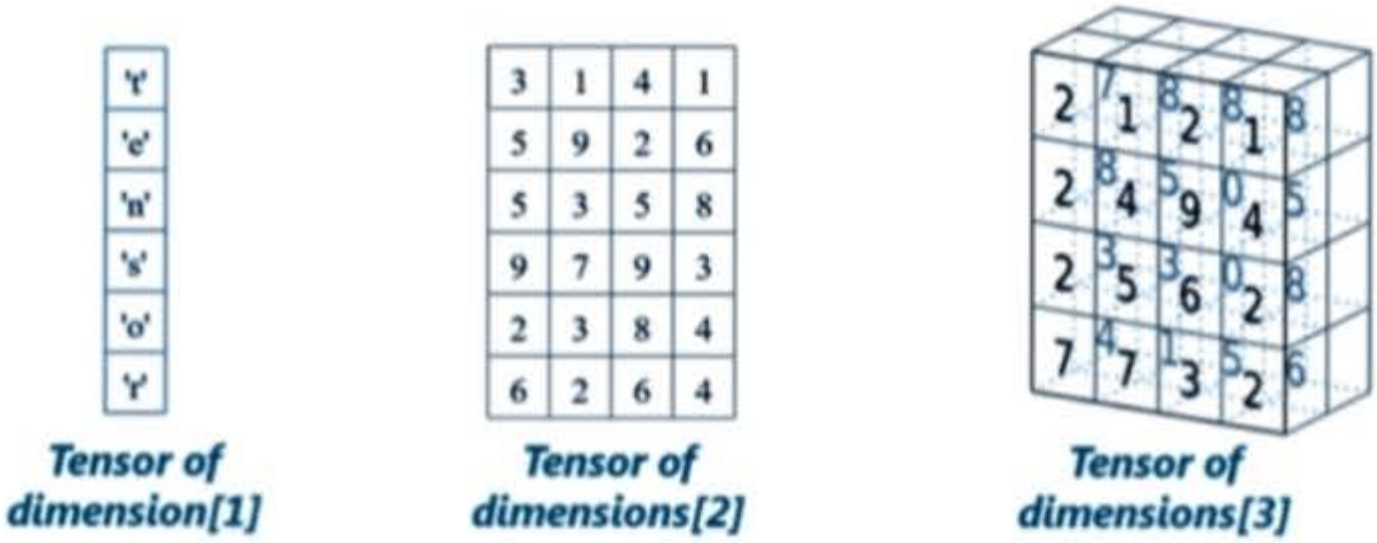
Batch Gradient Descent	**Stochastic Gradient Descent**
The batch gradient computes the gradient using the entire dataset.	The stochastic gradient computes the gradient using a single sample.
It takes time to converge because the volume of data is huge, and weights update slowly.	It converges much faster than the batch gradient because it updates weight more frequently.

Q98. Why Is Tensorflow the Most Preferred Library in Deep Learning?

Tensorflow provides both C++ and Python APIs, making it easier to work on and has a faster compilation time compared to other Deep Learning libraries like Keras and Torch. Tensorflow supports both CPU and GPU computing devices.

Question 26: What Do You Mean by Tensor in Tensorflow?

A tensor is a mathematical object represented as arrays of higher dimensions. These arrays of data with different dimensions and ranks fed as input to the neural network are called "***Tensors***."



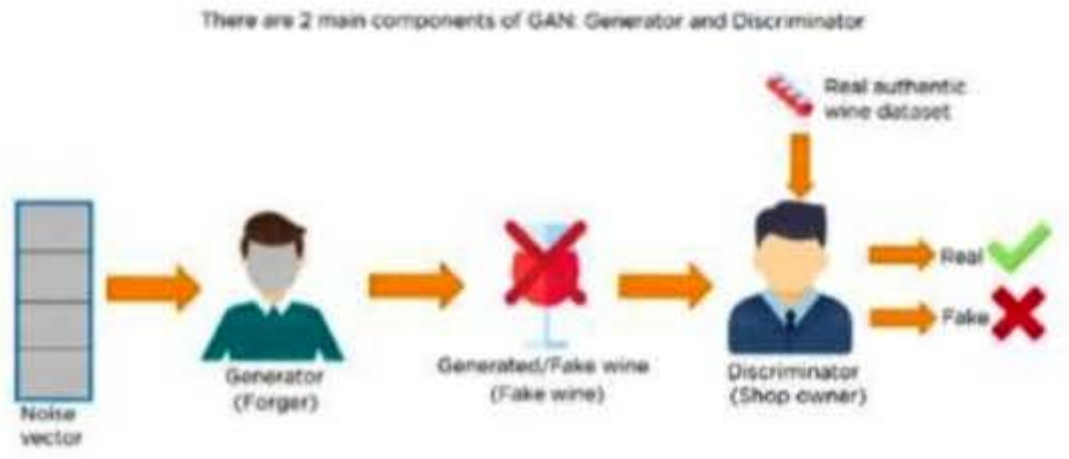
Tensor of dimension[1] ***Tensor of dimensions[2]*** ***Tensor of dimensions[3]***

Question 27: What is the Computational Graph?

Everything in a TensorFlow is based on creating a computational graph. It has a network of nodes where each node operates, Nodes represent mathematical operations, and edges represent tensors. Since data flows in the form of a graph, it is also called a "DataFlow Graph."

Question 28: What is a Generative Adversarial Network?

Suppose there is a wine shop purchasing wine from dealers, which they resell later. But some dealers sell fake wine. In this case, the shop owner should be able to distinguish between fake and authentic wine.
The forger will try different techniques to sell fake wine and make sure specific techniques go past the shop owner's check. The shop owner would probably get some feedback from wine experts that some of the wine is not original. The owner would have to improve how he determines whether a wine is fake or authentic.
The forger's goal is to create wines that are indistinguishable from the authentic ones while the shop owner intends to tell if the wine is real or not accurately
Let us understand this example with the help of an image.



There is a noise vector coming into the forger who is generating fake wine.
Here the forger acts as a Generator.
The shop owner acts as a Discriminator.
The Discriminator gets two inputs; one is the fake wine, while the other is the real authentic wine. The shop owner has to figure out whether it is real or fake.
So, there are two primary components of Generative Adversarial Network (GAN) named:
Generator
Discriminator
The generator is a CNN that keeps keys producing images and is closer in appearance to the real images while the discriminator tries to determine the difference between real and fake images The ultimate aim is to make the discriminator learn to identify real and fake images.
Apart from the very technical questions, your interviewer could even hit you up with a few simple ones to check your overall confidence, in the likes of the following.

Question 29: What are the important skills to have in Python with regard to data analysis?

The following are some of the important skills to possess which will come handy when performing data analysis using Python.
Good understanding of the built-in data types especially lists, dictionaries, tuples, and sets.
Mastery of N-dimensional NumPy Arrays.
Mastery of Pandas data frames.
Ability to perform element-wise vector and matrix operations on NumPy arrays.
Knowing that you should use the Anaconda distribution and the conda package manager.
Familiarity with Scikit-learn.
Ability to write efficient list comprehensions instead of traditional for loops.
Ability to write small, clean functions (important for any developer), preferably pure functions that don't alter objects.
Knowing how to profile the performance of a Python script and how to optimize bottlenecks.

Question 30: Do you have research experience in AI?

At present, a lot of work within the AI space is research-based. As a result, many organizations will be digging into your background to ascertain what kind of experience you have in this area. If you authored or co-authored research papers or have been supervised by industry leaders, make sure to share that information.
In fact, take it a step further and have a summary of your research experience along with your research papers ready to share with the interviewing panel.

However, if you don't have any formal research experience, have an explanation ready. For example, you can talk about how your AI journey started as a weekend hobby and grew into so much more within a space of two or three years.

Question 31: What's your favorite use case?

Just like research, you should be up to date on what's going on in the industry. As such, if you're asked about use cases, make sure that you have a few examples in mind that you can share. Whenever possible, bring up your personal experiences.

You can also share what's happening in the industry. For example, if you're interested in the use of AI in medical images, Health IT Analytics has some interesting use cases:

Detecting Fractures And Other Musculoskeletal Injuries
Aiding In The Diagnosis Neurological Diseases
Flagging Thoracic Complications And Conditions
Screening For Common Cancers

Question 32: How is Google training data for self-driving cars?

Google has been using reCAPTCHA to source labeled data on storefronts and traffic signs for many years now. The company also has been using training data collected by Sebastian Thrun, CEO of the Kitty Hawk Corporation, and the co-founder (and former CEO) of Udacity.

Such information, although it might not seem significant, will show a potential employer that you're interested in and excited about this field.

Question 33: Where do you usually source your data sets?

If you talk about AI projects that you've worked on in your free time, the interviewer will probably ask where you sourced your data sets. If you're genuinely passionate about the field, you would have worked on enough projects to know where you can find free data sets.

For example, here are some freely available public data sets that you should know about (without conducting a Google search):

CelebFaces (with 200,000 celebrity images along with 40 attribute annotations)
CIFAR (with 60,000 images that map 10 different classes)
YouTube-8M (with over 4,000 annotated entities taken from an enormous data set of YouTube videos)

Researchers have released hundreds of free resources like these along with the actual network architecture and weights used in their examples. So it will serve you well to explore some of these data sets and run some experiments before heading out for an AI interview.

Question 34: What is artificial intelligence?

AI can be described as an area of computer science that simulates human intelligence in machines. It's about smart algorithms making decisions based on the available data.

Whether it's Amazon's Alexa or a self-driving car, the goal is to mimic human intelligence at lightning speed (and with a reduced rate of error).

Question 35: What are the intelligent agents?

An intelligent agent is an autonomous entity that leverages sensors to understand a situation and make decisions. It can also use actuators to perform both simple and complex tasks.

In the beginning, it might not be so great at performing a task, but it will improve over time. The Roomba vacuum cleaner is an excellent example of this.

Question 36: What's the most popular programming language used in AI?

The open-source modular programming language Python leads the AI industry because of its simplicity and predictable coding behavior.
Its popularity can be attributed to open-source libraries like Matplotlib and NumPy, efficient frameworks such as Scikit-learn, and practical version libraries like Tensorflow and VTK.
There's a chance that the interviewer might keep the conversation going and ask you for more examples. If that happens, you can mention the following:
Java
Julia
Haskell
Lisp

Question 37: What are AI neural networks?

Neural networks in AI mathematically model how the human brain works. This approach enables the machine to think and learn as humans do. This is how smart technology today recognizes speech, objects, and more.

Question 38: What's the difference between strong AI and weak AI?

The difference between the two is just like the terms sound. Strong AI can successfully imitate human intelligence and is at the core of advanced robotics.
Weak AI can only predict specific characteristics that resemble human intelligence. Alexa and Siri are excellent examples of weak AI.

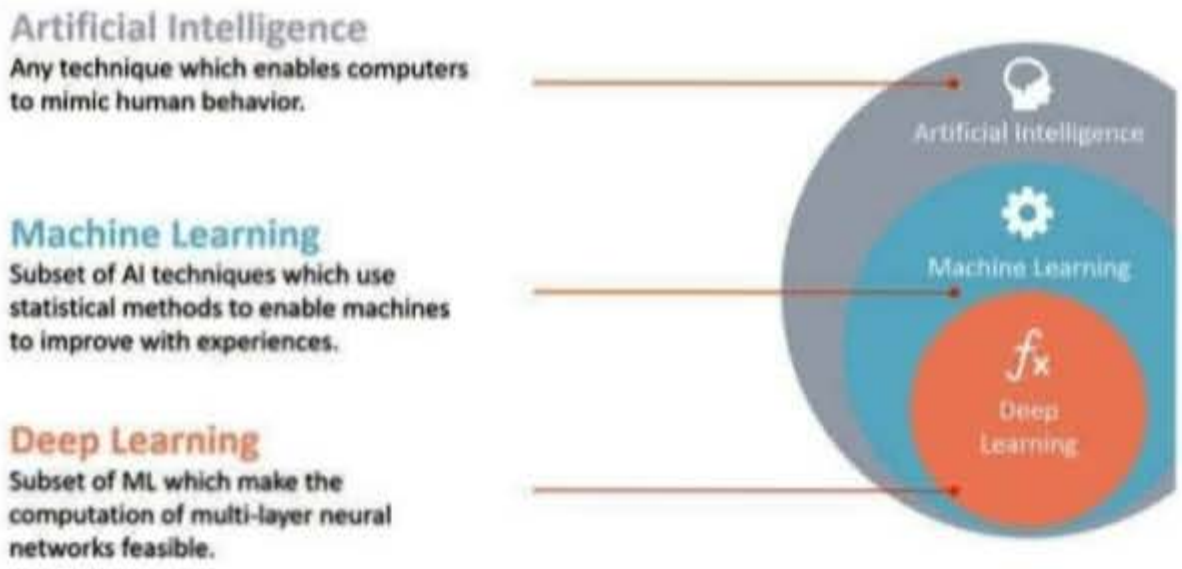
Strong AI

Can be applied widely
Extensive scope
Human-level intelligence
Processes data by using clustering and association

Weak AI

Can be great at performing some simple tasks
Uses both supervised and unsupervised learning
The scope can be minimal

Question 39: What's the difference between AI and ML?



AI and ML are closely related, but these terms aren't interchangeable. ML actually falls under the umbrella of AI. It demands that machines carry out tasks in the same way that humans do.
The current application of ML in AI is based around the idea that we should enable access to data so machines can observe and learn for themselves.

Question 40: How would you describe ML to a non-technical person?

ML is geared toward pattern recognition. A great example of this is your Facebook newsfeed and Netflix's recommendation engine.
In this scenario, ML algorithms observe patterns and learn from them. When you deploy an ML program, it will keep learning and improving with each attempt.
If the interviewer prods you to provide more real-world examples, you can list the following:
Amazon product recommendations
Fraud detection
Search ranking
Spam detection
Spell correction

Question 41: What are some examples of AI in use?

Some compelling examples of AI applications are:
Chatbots
Facial recognition
Image tagging
Natural language processing
Sales prediction
Self-driving cars
Sentiment analysis

Question 42: What's a Turing test?

The Turing test, named after Alan Turing, is a method of testing a machine's human-level intelligence. For example, in a human-versus-machine scenario, a judge will be tasked with identifying which terminal was occupied by a human and which was occupied by a computer-based on individual performance.
Whenever a computer can pass off as a human, it's deemed intelligent. The game has since evolved, but the premise remains the same.

Question 43: What's TensorFlow?

TensorFlow is an open-source framework dedicated to ML. It's a comprehensive and highly adaptable ecosystem of libraries, tools, and community resources that help developers build and deploy ML-powered applications. Both AlphaGo and Google Cloud Vision were built on the Tensorflow platform.

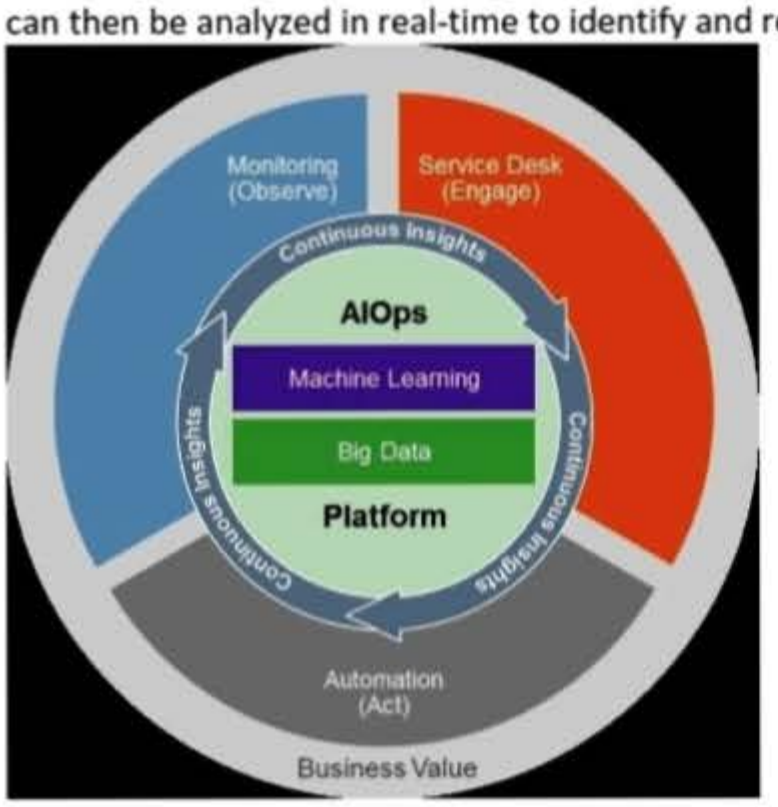
Question 44: Why is game theory important to AI?

Game theory, developed by American mathematician Josh Nash, is essential to AI because it plays an underlying role in how these smart algorithms improve over time.
At its most basic, AI is about algorithms that are deployed to find solutions to problems. Game theory is about players in opposition trying to achieve specific goals. As most aspects of life are about competition, game theory has many meaningful real-world applications.
These problems tend to be dynamic. Some game theory problems are natural candidates for AI algorithms. So, whenever game theory is applied, multiple AI agents that interact with each other will only care about utility to themselves.
Data scientists within this space should be aware of the following games:
Symmetric vs. asymmetric
Perfect vs. imperfect information
Cooperative vs. non-cooperative
Simultaneous vs. sequential
Zero-sum vs. non-zero-sum

Question 45: In your opinion, how will AI impact application development?

These types of questions help the interviewer ascertain your level of interest in the field. If you're naturally passionate about AI and everything related to it, you should have some knowledge about current industry trends.

So, if you have been actively following this space, you'll know all about AIOps. In the coming months, you can expect AI to be more involved in how we build applications. It has the potential to transform how we use and manage the infrastructure at a micro and macro level.
Some say that DevOps will be replaced by what they are calling AIOps because it allows developers to engage in accurate root cause analysis by combining big data, ML, and visualization.
AIOps can be described as a multilayered platform that can be used to automate and improve IT operations. In this scenario, developers can leverage analytics and ML to collect and process data from a variety of sources. This information can then be analyzed in real-time to identify and rectify problems.



Question 46: What would you say are common misconceptions about AI?

Many AI-related misconceptions are making the rounds in the age of "fake news." The most common ones are:
AI will replace humans
AI systems aren't safe
AI will lead to significant unemployment
While these types of stories are common, they're far from the truth. Even though some AI-based technology is able to complete some tasks—for example, analyzing zettabytes of data in less than a second—it still needs humans to gather the data and define the patterns for identification.
So we aren't near a reality where technology has the potential to replace us or our jobs.

Question 47: Can you name the properties of a good knowledge representation system?

From the perspective of systems theory, a good knowledge representation system will have the following:
Acquisition efficiency to acquire and incorporate new data
Inferential adequacy to derive knowledge representation structures like symbols when new knowledge is learned from old knowledge
Inferential efficiency to enable the addition of data into existing knowledge structures to help the inference process
Representation adequacy to represent all the knowledge required in a specific domain
Natural Language Processing Questions

Natural Language Processing Questions

Q1. What is Lemmatization?

Ans: Lemmatization is the process of arriving at a lemma of a word. What is a lemma, then? Lemma is the root from which a word is formed.

For example: If we lemmatize 'studies' and 'studying', we will end up with 'study' as its lemma. We got to this conclusion after the morphological analysis of both words. These were mapped in a dictionary which helped in arriving at the lemma

Q2 What is Stemming?

TFt,d is the number of occurrences of t in document d.
DFt is the number of documents containing the term t.
N is the total number of documents in the corpus
For More details, Visit the below link
https://www.onely.com/blog/what-is-tf-idf/

Q7: What is ngram in NLP. ?

Ans - An N-gram means a sequence of N words. So for example, "Medium blog" is a 2-gram (a bigram), "A Medium blog post" is a 4-gram, and "Write on Medium" is a 3-gram (trigram). Basically, an N-gram model predicts the occurrence of a word based on the occurrence of its N – 1 previous words.

Let us see a way to assign a probability to a word occurring next in a sequence of words. First of all, we need a very large sample of English sentences (called a corpus).
For the purpose of our example, we'll consider a very small sample of sentences, but in reality, a corpus will be extremely large. Say our corpus contains the following sentences:

He said thank you.
He said bye as he walked through the door.
He went to San Diego.
San Diego has nice weather.
It is raining in San Francisco.

Let's assume a bigram model. So we are going to find the probability of a word based only on its previous word. Let us see a way to assign a probability to a word occurring next in a sequence of words. First of all, we need a very large sample of English sentences (called a corpus).
For the purpose of our example, we'll consider a very small sample of sentences, but in reality, a corpus will be extremely large. Say our corpus contains the following sentences:
He said thank you.
He said bye as he walked through the door.
He went to San Diego.
San Diego has nice weather.
It is raining in San Francisco.
Let's assume a bigram model. So we are going to find the probability of a word based only on its previous word. In general, we can say that this probability is (the number of times the previous word 'wp' occurs before the word 'wn') / (the total number of times the previous word 'wp' occurs in the corpus) =
(Count (wp wn))/(Count (wp))
Let's work this out with an example.
To find the probability of the word "you" following the word "thank", we can write this as P (you | thank) which is a conditional probability.
This becomes equal to:

=(No. of times "Thank You" occurs) / (No. of times "Thank" occurs) = 1/1 = 1
We can say with certainty that whenever "Thank" occurs, it will be followed by "You" (This is because we have trained on a set of only five sentences,and "Thank" occurred only once in the context of "Thank You").
Refernce:- https://blog.xrds.acm.org/2017/10/introduction-n-grams-need/

Q 8 What is "perplexed" in NLP?

Ans- The word "perplexed" means "puzzled" or "confused", thus Perplexity in general means the inability to tackle something complicated and a problem that is not specified. Therefore, Perplexity in NLP is a way to determine the extent of uncertainty in predicting some text.

In NLP, perplexity is a way of evaluating language models. Perplexity can be high and low; Low perplexity is ethical because the inability to deal with any complicated problem is less while high perplexity is terrible because the failure to deal with a complicated is high.
Points to take care:
1. Average branching factor in predicting the next word .
2. Lower is better (lower perplexity -> higher probability)

N = number of words NP

$$Per = \sqrt[N]{\frac{1}{P(w_1 w_2 \ldots w_N)}}$$

Reference:- https://towardsdatascience.com/perplexity-intuition-and-derivation-105dd481c8f3

Q 9 What is Pragmatic Ambiguity in NLP?

Ans- Ambiguity, generally used in natural language processing, can be referred to as the ability to be understood in more than one way. In simple terms, we can say that ambiguity is the capability of being understood in more than one way. Natural language is very ambiguous. NLP has the following types of ambiguities.
1. Lexical Ambiguity
The ambiguity of a single word is called lexical ambiguity. For example, treating the word silver as a noun, an adjective, or a verb.
2. Syntactic Ambiguity

This kind of ambiguity occurs when a sentence is parsed in different ways. For example, the sentence "The man saw the girl with the telescope". It is ambiguous whether the man saw the girl carrying a telescope or he saw her through his telescope.

3. Semantic Ambiguity
This kind of ambiguity occurs when the meaning of the words themselves can be misinterpreted. In other words, semantic ambiguity happens when a sentence contains an ambiguous word or phrase. For example, the sentence "The car hit the pole while it was moving" is having semantic ambiguity because the interpretations can be "The car, while moving, hit the pole" and "The car hit the pole while the pole was moving".

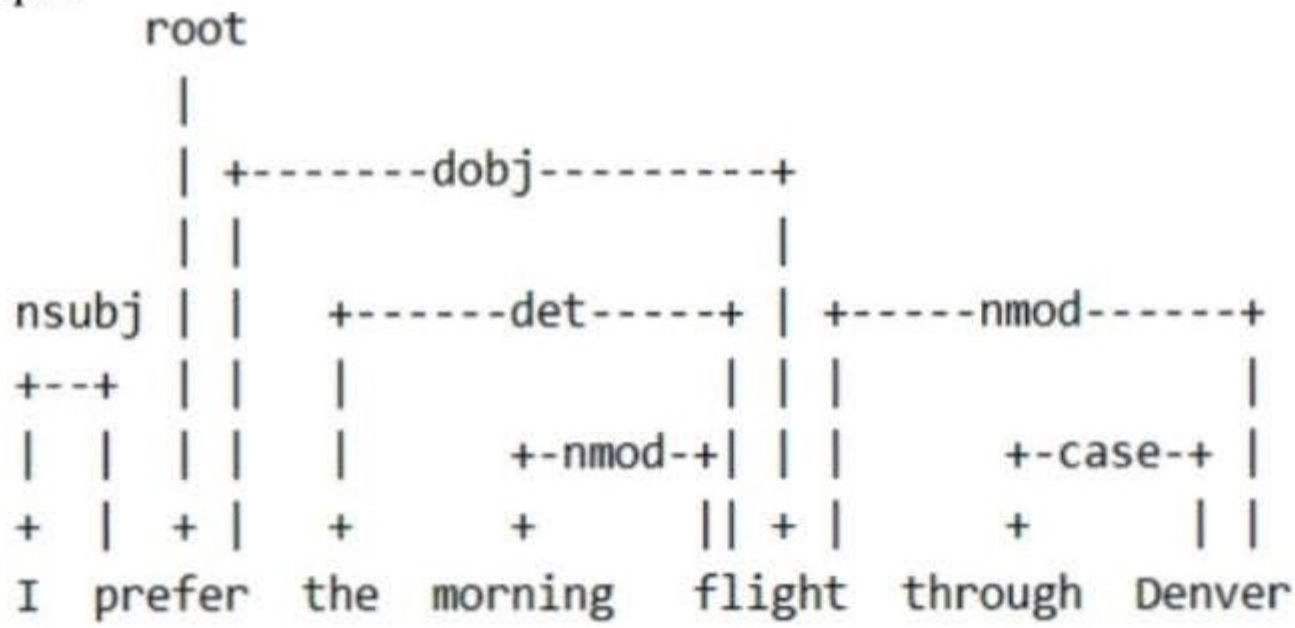
4. Anaphoric Ambiguity
This kind of ambiguity arises due to the use of anaphora entities in discourse. For example, the horse ran up the hill. It was very steep. It soon got tired. Here, the anaphoric reference of "it" in two situations cause ambiguity.

5.Pragmatic ambiguity

Such kind of ambiguity refers to the situation where the context of a phrase gives it multiple interpretations. In simple words, we can say that pragmatic ambiguity arises when the statement is not specific. For example, the sentence "I like you too" can have multiple interpretations as I like you (just like you like me), I like you (just like someone else does).

Q9. Explain Dependency Parsing.

Ans - Dependency parsing is the task of extracting a dependency parse of a sentence that represents its grammatical structure and defines the relationships between "head" words and words, which modify those heads.
Example:



Relations among the words are illustrated above the sentence with directed, labeled arcs from heads to dependents (+ indicates the dependent).
Refernce:-https://medium.com/@5hirish/dependency-parsing-in-nlp-d7ade014186

Q 10 What is Pragmatic Analysis?

Ans - Pragmatic Analysis is part of the process of extracting information from text. Specifically, it's the portion that focuses on taking a structured set of text and figuring out what the actual meaning was.

Why is this important? Because a lot of text's meaning does have to do with the context in which it was said/written. Ambiguity, and limiting ambiguity, are at the core of natural language processing, so needless to say, pragmatic analysis is actually quite crucial with respect to extracting meaning or information.

Q11. Explain Masked language modeling?

Ans- Masked language modeling is a fill-in-the-blank task, where a model uses the context words surrounding a [MASK] token to try to predict what the [MASK] word should be.
The model shown here is BERT, the first large transformer to be trained on this task. Enter text with one or more "[MASK]" tokens and the model will generate the most likely substitution for each.

Sentence:

The doctor ran to the emergency room to see [MASK] patient.

Mask 1 Predictions:

38.3% **his**
36.9% **the**
8.1% **another**
7.3% **a**
6.0% **her**

Q12 What is the difference between NLP and CI(Conversational Interfaces)?

Ans

Difference between NLP and CI(Conversational Interfaces)

Natural Language Processing	**Conversational Interfaces**
NLP is a kind of artificial intelligence technology that allows identifying, understanding and interpreting the request of users in the form of language.	CI is a user interface that mixes voice, chat and another natural language with images, videos or buttons.
NLP aims to make users understand a particular concept.	Conversational Interface provides only what the users need and not more than that.

Q13 Where can we use NLP? Give some real life examples?

Ans Natural Language Processing can be used for
Semantic Analysis
Automatic summarization
Text classification
Question Answering
Some real-life example of NLP is IOS Siri, the Google Assistant, Amazon echo

Q14. Define some of the NLP Terminologies?

Ans -NLP Terminology is based on the following factors:
Weights and Vectors: TF-IDF, length(TF-IDF, doc), Word Vectors, Google Word Vectors
Text Structure: Part-Of-Speech Tagging, Head of the sentences, Named Entity
Sentiment Analysis: Sentiment Dictionary, Sentiment Entities, Sentiment Features
Text Classification: Supervised Learning, Train Set, Dev(=Validation) Set, Test Set, Text Features, LDA.
Machine Reading: Entity Extraction, Entity Linking,dbpedia, FRED (lib) / Pikes

Q15. What is POS tagging?

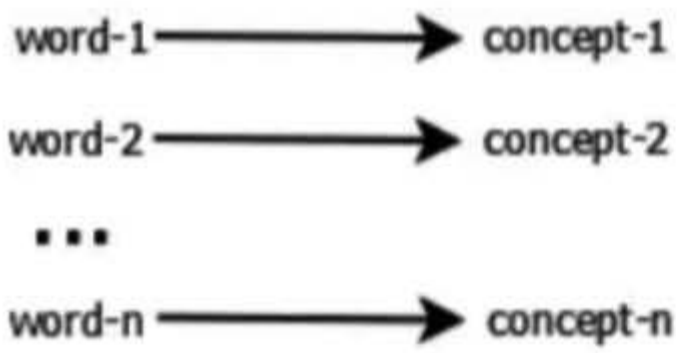
Ans. t is a process of converting a sentence to forms – list of words, list of tuples (where each tuple is having a form *(word, tag)*). The tag in case is a part-of-speech tag and signifies whether the word is a noun, adjective, verb, and so on.

Part of Speech	**Tag**
Noun	n
Verb	v
Adjective	a
Adverb	r

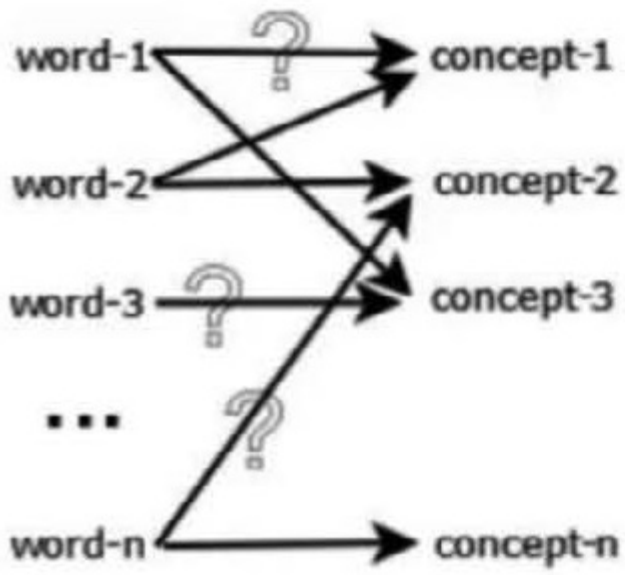
Part of Speech	Tag
Noun	n
Verb	v
Adjective	a
Adverb	r

Q16. What is tokenization?
Ans. Tokenization is a way of separating a piece of text into smaller units called tokens. Here, tokens can be either words, characters, or subwords. Hence, tokenization can be broadly classified into 3 types – word, character, and subword (n-gram characters) tokenization.
For example, consider the sentence: "Never give up".
The most common way of forming tokens is based on space. Assuming space as a delimiter, the tokenization of the sentence results in 3 tokens – Never-give-up. As each token is a word, it becomes an example of Word tokenization.
Similarly, tokens can be either characters or subwords. For example, let us consider "smarter":
Character tokens: s-m-a-r-t-e-r
Subword tokens: smart-er

Q17. What is Latent Semantic Indexing (LSI), where can you apply them?
Ans: Latent Semantic Analysis (LSA), also known as Latent Semantic Indexing (LSI) literally means analyzing documents to find the underlying meaning or concepts of those documents. If each word only meant one concept, and each concept was only described by one word, then LSA would be easy since there is a simple mapping from words to concepts.



Unfortunately, this problem is difficult because English has different words that mean the same thing (synonyms), words with multiple meanings, and all sorts of ambiguities that obscure the concepts to the point where even people can have a hard time understanding.



For example, the word bank when used together with mortgage, loans, and rates probably means a financial institution. However, the word bank when used together with lures, casting, and fish probably means a stream or riverbank

Q 18. What is the difference between Regular Grammars and regular expressions?

Ans- Both have equal expressive power, and moreover, both are used to represent regular languages. But regular grammars is a grammar means, it can be represented by {V, T, P, S},
where V = set of Variables,
T = set of terminals,
P = set of productions,
S = start symbol.
Also, regular grammars are either left-linear or right-linear.

But, regular expressions can be expressed only using alphabets, (,), *, +, union, intersection, difference,etc.

Regular grammars can be converted to regular expressions and vice-versa.

Q19. What are distance-based classifiers?

Ans - Distance-based classifier" is a pretty ambiguous term. I will assume for this answer that you are referring to a classifier basing it's decision on the distance calculated from the target instance to the training instances, for example the k-Nearest Neighbors algorithm.
In k-Nearest Neighbors, you determine the nearest k training instances to your target instance. Figuring out which k are the nearest involves calculating some sort of distance function. If all the input features are normalized to real values between 0 and 1, this is usually a rather simple Euclidean Distance.

Q 20. How to tokenize a sentence using the nltk package?

Ans Tokenization is a process used in NLP to split a sentence into tokens. Sentence tokenization refers to splitting a text or paragraph into sentences.
For tokenizing, we will import sent_tokenize from the nltk package:

```
from nltk.tokenize import sent_tokenize<>
```

We will use the below paragraph for sentence tokenization:

```
Para = “Hi Guys. Welcome to DataTrained. This is a blog on the NLP interview questions and answers.”
sent_tokenize(Para)
```

Output:

```
[ 'Hi Guys.' ,  'Welcome to DataTrained.',
a blog on the NLP interview questions and answers. ' ]
```

Tokenizing a word refers to splitting a sentence into words.
Now, to tokenize a word, we will import word_tokenize from the nltk package.

```
from nltk.tokenize import word_tokenize
Para = “Hi Guys. Welcome to DataTrained. This is a blog on the NLP interview Questions and answers.
word_tokenize(Para)
```

Output:

```
[ 'Hi' , 'Guys' , ' . ' , 'Welcome' , 'to' , 'DataTrained'
'This' , 'is' ,   'a', 'blog' , 'on' , 'the' , 'NLP' , 'interview' , 'questions' , 'and' , 'answers' , ' . ' ]
```