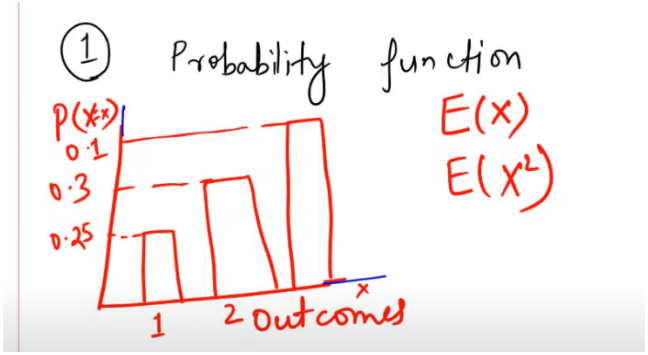
Session 16

Sunday, March 26, 2023 1:04 PM



$E(x^2) = 0.25 + 4(0.3) + 9(0.1)$

$= 0.25 + 1.2 + 0.9$

$= 0.25 + 2.1$

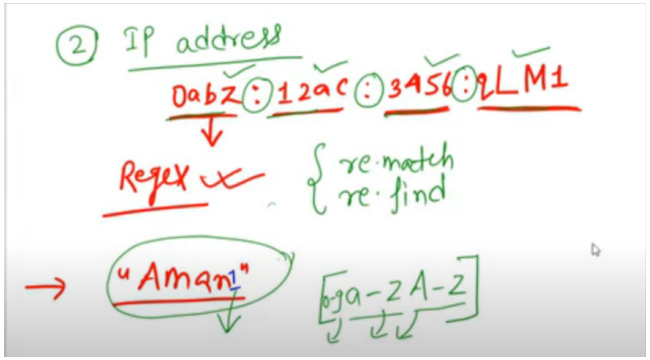
$E(x^2) = 2.35$

$E(x) = 0.25(1) + 2(0.3) + 3(0.1)$

$= 0.25 + 0.6 + 0.3$

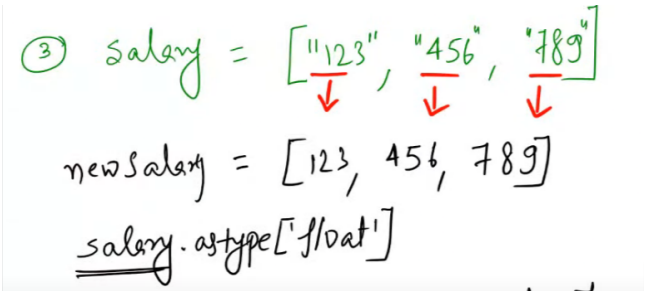
$= 0.25 + 0.9 = 1.15$

42.111.2

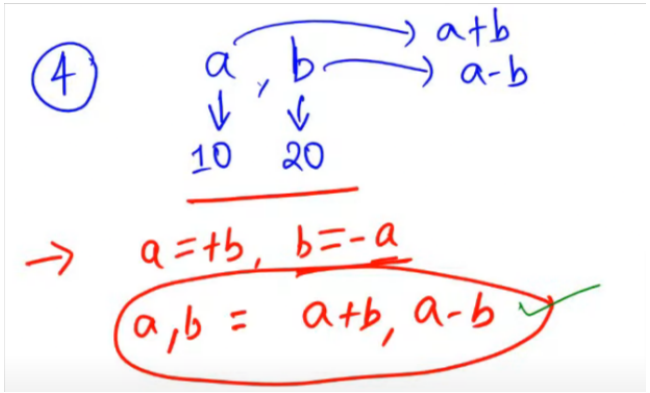


0

```
import re

def is_valid_ip(ip):
    pattern = '^((25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.){3}(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)$'
    return bool(re.match(pattern, ip))

# Example usage:
print(is_valid_ip('192.168.1.1'))  # True
print(is_valid_ip('10.0.0.1'))     # True
print(is_valid_ip('256.0.0.1'))    # False
print(is_valid_ip('abc.xyz.pqr'))  # False
```



[int(i) for i in salary]



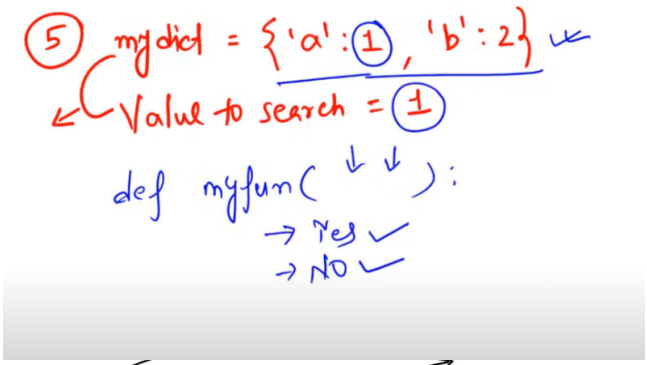

```
a=10
b=20
print(a,b)
```

```
10 20
```

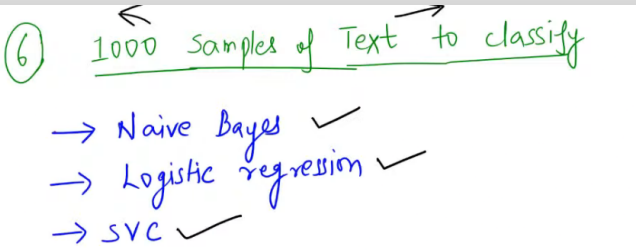
```
a,b=a+b,a-b
```

```
print(a,b)
```

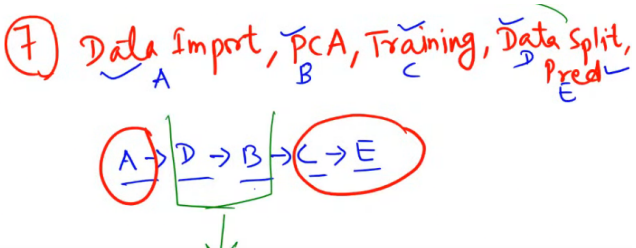
```
30 -10
```

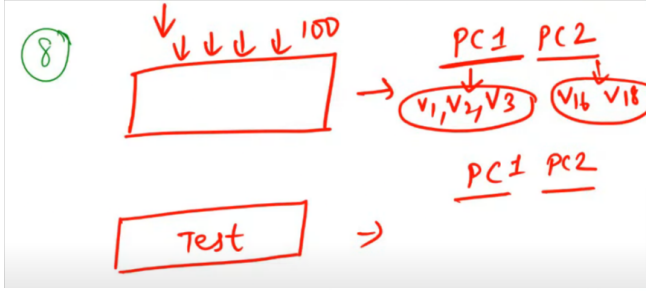


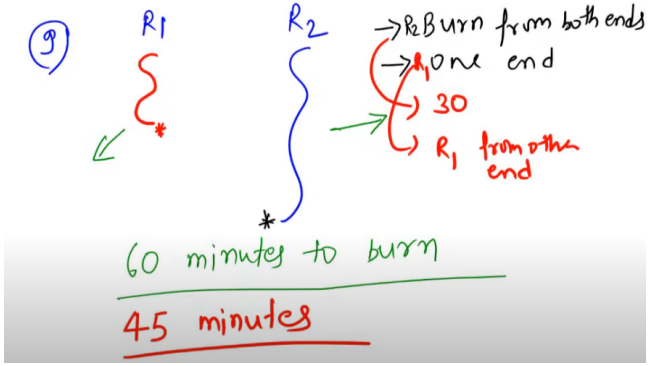

```
In [9]: def myfunc(dictionary, value):
            """
            This function takes a dictionary and a value as input,
            and returns True if the value is present in the dictionary,
            False otherwise.
            """
            if value in dictionary.values():
                return True
            else:
                return False

In [10]: myfunc(mydict,1)
Out[10]: True
```



However, given the three algorithms you mentioned, Naive Bayes is a popular choice for text classification tasks due to its simplicity, fast training time, and good performance on small to medium-sized datasets. Support Vector Machines (SVM) also perform well on text classification tasks, especially when combined with kernel methods. Logistic Regression is another widely used classification algorithm that works well when the relationship between features and target variables is linear.







burn one on both end and other on one end . after the first one completly burn then second one burn both end