914. X of a Kind in a Deck of Cards

In a deck of cards, each card has an integer written on it.

Return `true` if and only if you can choose `X >= 2` such that it is possible to split the entire deck into 1 or more groups of cards, where:

- Each group has exactly `X` cards.
- All the cards in each group have the same integer.

Example 1:

```
Input: deck = [1,2,3,4,4,3,2,1]
Output: true
Explanation: Possible partition [1,1],[2,2],[3,3],[4,4].
```

Example 2:

```
Input: deck = [1,1,1,2,2,2,3,3]
Output: false
Explanation: No possible partition.
```

Example 3:

```
Input: deck = [1]
Output: false
Explanation: No possible partition.
```

Example 4:

```
Input: deck = [1,1]
Output: true
Explanation: Possible partition [1,1].
```

Example 5:

```
Input: deck = [1,1,2,2,2,2]
Output: true
Explanation: Possible partition [1,1],[2,2],[2,2].
```

Constraints:

- `1 <= deck.length <= 10<sup>4</sup>`
- `0 <= deck[i] < 10<sup>4</sup>`

-
  ```
  import math
  class Solution:
      def hasGroupsSizeX(self, deck: List[int]) -> bool:
          if len(deck)==1:
              return False
          if len(deck)==2:
              if deck[0]==deck[1]:
                  return True
              else:
                  return False
          freq = collections.Counter(deck)
          ans = 0
          for key in freq.keys():
              ans = gcd(ans,freq[key])
          if ans>=2:
              return True
          else:
              return False
  ```