Generating subsets

Piotr Karamon

Contents

A subset of a n element set can be fully describe as a binary sequence of length n. This binary sequence indicates presence or absence of a element. Example:

Set A = {1,2,3,4}

Subset s = 0110

meaning s contains 2,3 and does not contain 1,4

Generating subsets comes down to generating those binary sequences of length n

1 Generating all subsets

```python
A = [1,2,3,4]
subset = [0]*len(A)

def subsets(n, p):
    if n == p:
        print('elements', [A[i] for i in range(len(subset)) if subset[i]
        ↪  == 1], subset)
        return
    subset[p] = 0
    subsets(n, p+1)
    subset[p] = 1
    subsets(n,p +1)

subsets(len(A), 0)
```

Results

```
elements [] [0, 0, 0, 0]
elements [4] [0, 0, 0, 1]
elements [3] [0, 0, 1, 0]
elements [3, 4] [0, 0, 1, 1]
elements [2] [0, 1, 0, 0]
elements [2, 4] [0, 1, 0, 1]
elements [2, 3] [0, 1, 1, 0]
elements [2, 3, 4] [0, 1, 1, 1]
elements [1] [1, 0, 0, 0]
elements [1, 4] [1, 0, 0, 1]
elements [1, 3] [1, 0, 1, 0]
elements [1, 3, 4] [1, 0, 1, 1]
elements [1, 2] [1, 1, 0, 0]
elements [1, 2, 4] [1, 1, 0, 1]
elements [1, 2, 3] [1, 1, 1, 0]
elements [1, 2, 3, 4] [1, 1, 1, 1]
```

2 Next subset

```python
A = [1,2,3,4]
subset = [0]*len(A)

def next_subset(subset):
    i = 0
    while i < len(subset) and subset[i] == 1:
        i+=1
    # looping behaviour
    if i==len(subset):
        for i in range(len(subset)):
            subset[i] = 0
        return subset
    subset[i] = 1
    for j in range(0, i):
        subset[j] = 0
    return subset


get_elements = lambda subset: [A[i] for i in range(len(subset)) if subset[i] == 1]
subset = [0,0,0,0]
for i in range(18):
    print(next_subset(subset), get_elements(subset))
```

Results

```
[1, 0, 0, 0] [1]
[0, 1, 0, 0] [2]
[1, 1, 0, 0] [1, 2]
[0, 0, 1, 0] [3]
[1, 0, 1, 0] [1, 3]
[0, 1, 1, 0] [2, 3]
[1, 1, 1, 0] [1, 2, 3]
[0, 0, 0, 1] [4]
[1, 0, 0, 1] [1, 4]
[0, 1, 0, 1] [2, 4]
[1, 1, 0, 1] [1, 2, 4]
[0, 0, 1, 1] [3, 4]
[1, 0, 1, 1] [1, 3, 4]
[0, 1, 1, 1] [2, 3, 4]
[1, 1, 1, 1] [1, 2, 3, 4]
[0, 0, 0, 0] []
[1, 0, 0, 0] [1]
[0, 1, 0, 0] [2]
```

3 Subsets with specified size

Say we have an n element set and we are only interested in k element subsets of it. Any set with n elements we can represent as $A = \{1, 2, 3, ..., n\}$.

Example:

Results

```
A=1,2,3,4,5
We want only subsets with size 3.
1 2 3
1 2 4
1 2 5
1 3 4
1 3 5
1 4 5
2 3 4
2 3 5
2 4 5
3 4 5
```

```python
def next_subset(n, subset):
    i = 0
    if subset[i] < n:
        subset[i]+=1
        return subset
    else:
        pass
    return subset

n, x = 5,[3,2,1]

print(x)
for i in range(10):
    x = next_subset(n,x)
    print(x)
```

Results

```
[3, 2, 1]
[4, 2, 1]
[5, 2, 1]
[5, 2, 1]
[5, 2, 1]
[5, 2, 1]
[5, 2, 1]
[5, 2, 1]
[5, 2, 1]
[5, 2, 1]
[5, 2, 1]
```