Neural Networks: Unrolling Parameters

Unrolling our parameters Θ from matrices into vectors will be necessary when using more advanced optimization algorithms other than gradient descent. In Python, there are two methods that you can use to flatten a matrix (multi-dimensional array): *flatten()* and *ravel()*

Create a Matrix

Input:

```python
import numpy as np

matrix = np.array([[  1,  2,  3,  4,  5 ],
                   [  6,  7,  8,  9, 10 ],
                   [ 11, 12, 13, 14, 15 ],
                   [ 16, 17, 18, 19, 20 ],
                   [ 21, 22, 23, 24, 25]])

matrix
```

Output:

```
array([[ 1,  2,  3,  4,  5],
       [ 6,  7,  8,  9, 10],
       [11, 12, 13, 14, 15],
       [16, 17, 18, 19, 20],
       [21, 22, 23, 24, 25]])
```

Unroll Into a Vector

```python
vector = matrix.flatten()
vector
```

Output:

```
array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17,
       18, 19, 20, 21, 22, 23, 24, 25])
```

Reshape Back to Matrix

Input:

```python
vector.reshape(5, 5)
```

Output:

```
array([[ 1,  2,  3,  4,  5],
       [ 6,  7,  8,  9, 10],
       [11, 12, 13, 14, 15],
       [16, 17, 18, 19, 20],
       [21, 22, 23, 24, 25]])
```

https://www.python-course.eu/numpy_changing_dimensions.php

```
array([[ 1,  2,  3,  4,  5],
       [ 6,  7,  8,  9, 10],
       [11, 12, 13, 14, 15],
       [16, 17, 18, 19, 20],
       [21, 22, 23, 24, 25]])
```