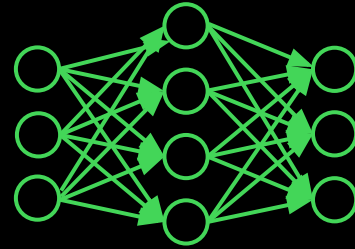


4 / 8



4
/ 8

Deep Learning Made Easy:

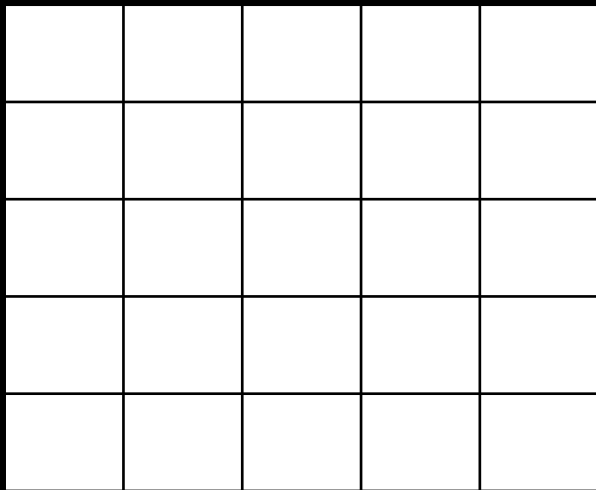
*Create a CNN Model for
Handwritten Digits*



CodeWithMe

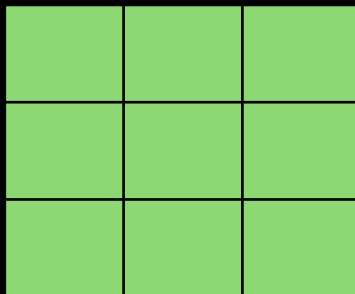


Conv2D:



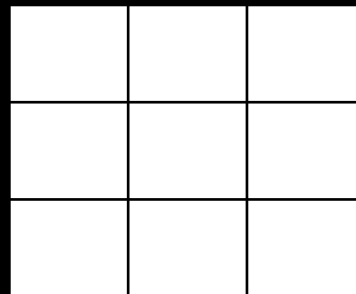
Input: image
5x5

X



Kernel:
3x3

=



Output: 3 x
3

Conv2D:

| | | | | |
|-------|-------|-------|--|--|
| i_1 | i_2 | i_3 | | |
| i_4 | i_5 | i_6 | | |
| i_7 | i_8 | i_9 | | |
| | | | | |
| | | | | |

Input: image
5x5

X

| | | |
|-------|-------|-------|
| w_1 | w_2 | w_3 |
| w_4 | w_5 | w_6 |
| w_7 | w_8 | w_9 |

Kernel:
3x3

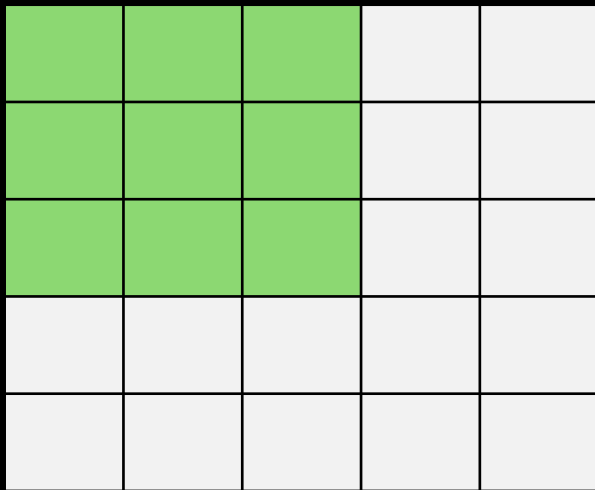
=

$$= i_1 w_1 + i_2 w_2 + \dots + i_9 w_9$$



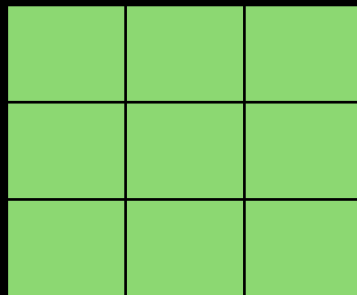
| | | |
|--|--|--|
| | | |
| | | |
| | | |

Conv2D:



Input: image
5x5

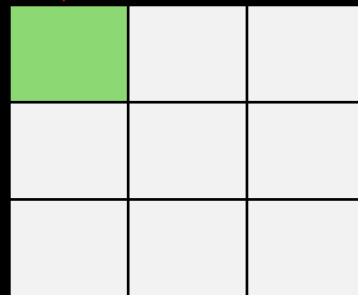
X



Kernel:
3x3

=

$$\sum_{j=1}^9 w_j i_j$$



Conv2D:

| | | | | |
|--|--|--|--|--|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Input: image
5x5

X

| | | |
|--|--|--|
| | | |
| | | |
| | | |

Kernel:
3x3

=

| | | |
|--|--|--|
| | | |
| | | |
| | | |

Conv2D:

| | | | | |
|--|--|--|--|--|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Input: image
5x5

X

| | | |
|--|--|--|
| | | |
| | | |
| | | |

Kernel:
3x3

=

| | | |
|--|--|--|
| | | |
| | | |
| | | |

Conv2D:

| | | | | |
|--|--|--|--|--|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Input: image
5x5

X

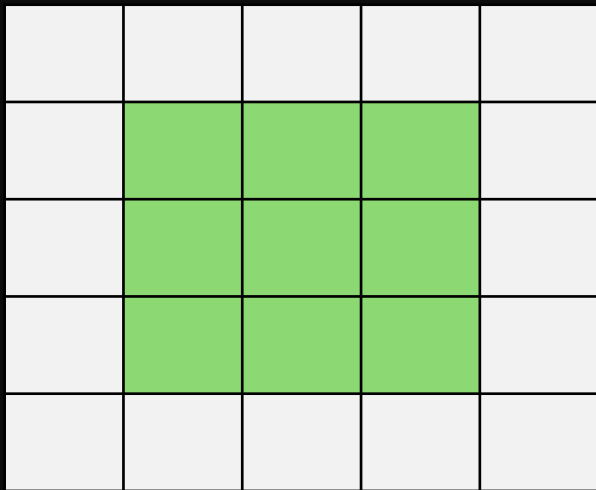
| | | |
|--|--|--|
| | | |
| | | |
| | | |

Kernel:
3x3

=

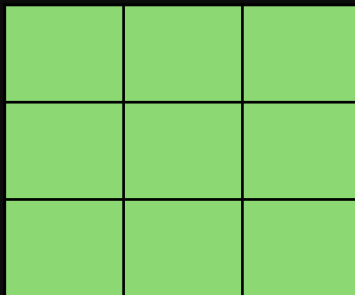
| | | |
|--|--|--|
| | | |
| | | |
| | | |

Conv2D:



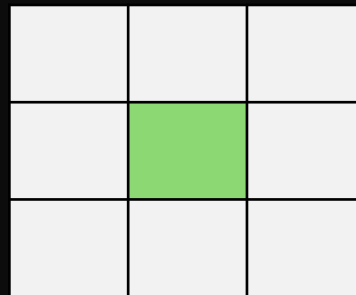
Input: image
5x5

X

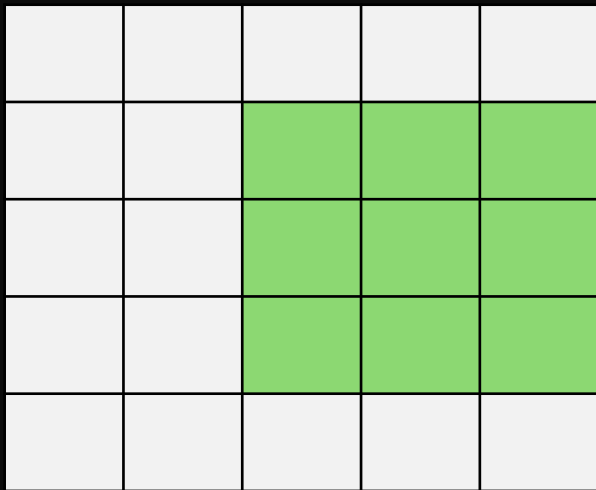


Kernel:
3x3

=

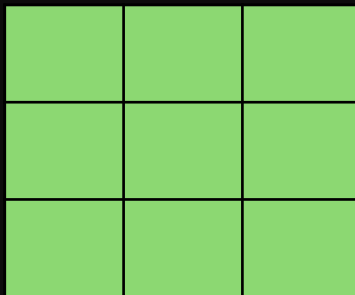


Conv2D:



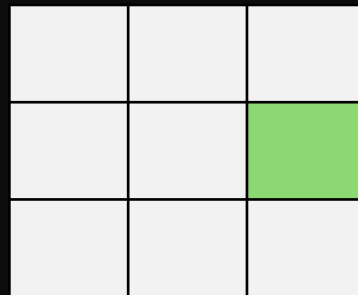
Input: image
5x5

X



Kernel:
3x3

=



Conv2D:

| | | | | |
|--|--|--|--|--|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Input: image
5x5

X

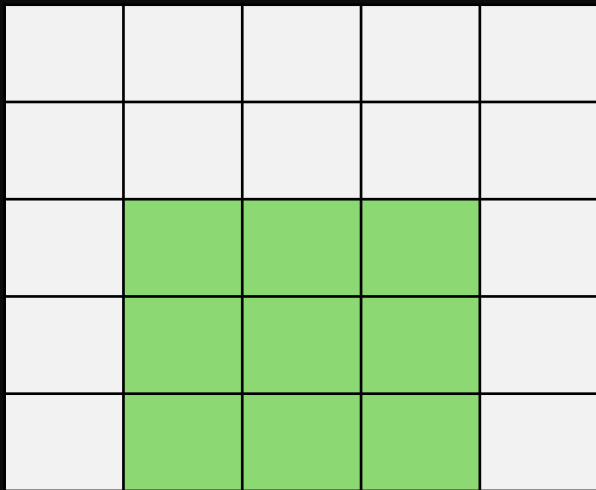
| | | |
|--|--|--|
| | | |
| | | |
| | | |

Kernel:
3x3

=

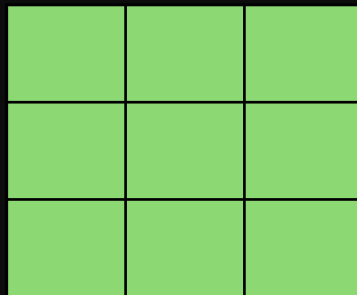
| | | |
|--|--|--|
| | | |
| | | |
| | | |

Conv2D:



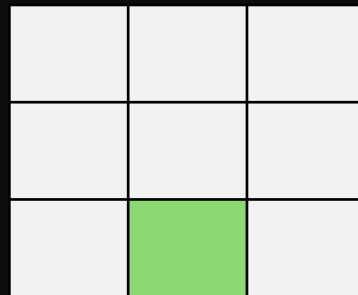
Input: image
5x5

X

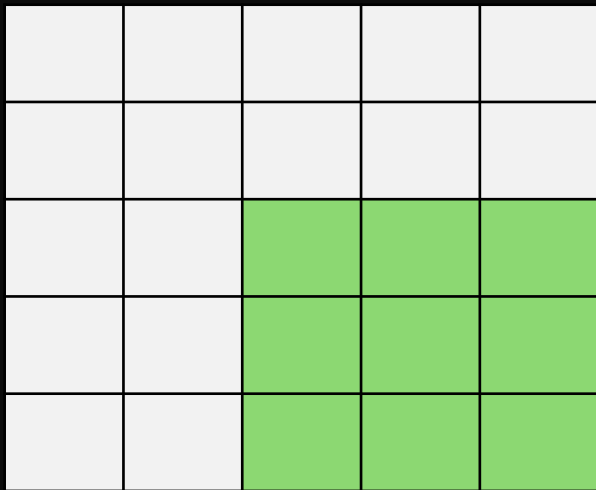


Kernel:
3x3

=

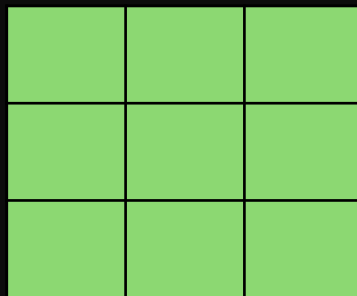


Conv2D:



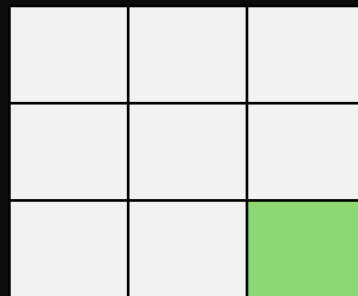
Input: image
5x5

X



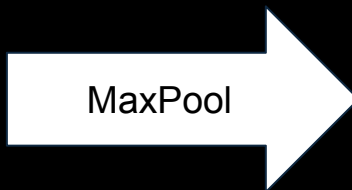
Kernel:
3x3

=



MaxPool2D:

| | | | |
|---|---|---|---|
| 4 | 1 | 1 | 8 |
| 1 | 1 | 0 | 2 |
| 7 | 3 | 2 | 2 |
| 3 | 8 | 4 | 1 |



| | |
|--|--|
| | |
| | |

MaxPool2D:

| | | | |
|---|---|---|---|
| 4 | 1 | 1 | 8 |
| 1 | 1 | 0 | 2 |
| 7 | 3 | 2 | 2 |
| 3 | 8 | 4 | 1 |

MaxPool

| | |
|--|--|
| | |
| | |

MaxPool2D:

| | | | |
|---|---|---|---|
| 4 | 1 | 1 | 8 |
| 1 | 1 | 0 | 2 |
| 7 | 3 | 2 | 2 |
| 3 | 8 | 4 | 1 |

MaxPool

| | |
|--|--|
| | |
| | |

What is $\text{Max}(4, 1, 1, 1)$?

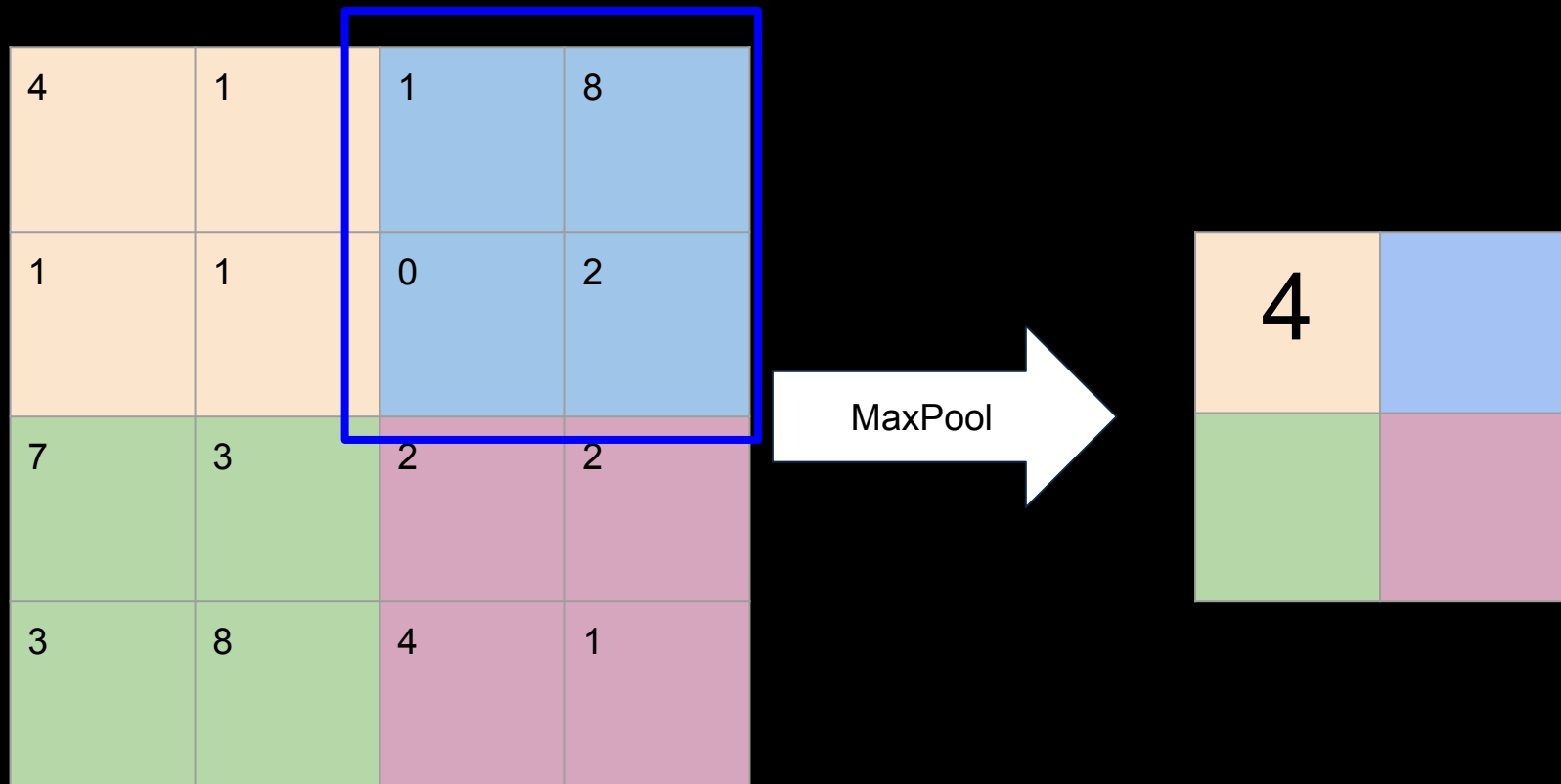
MaxPool2D:

| | | | |
|---|---|---|---|
| 4 | 1 | 1 | 8 |
| 1 | 1 | 0 | 2 |
| 7 | 3 | 2 | 2 |
| 3 | 8 | 4 | 1 |

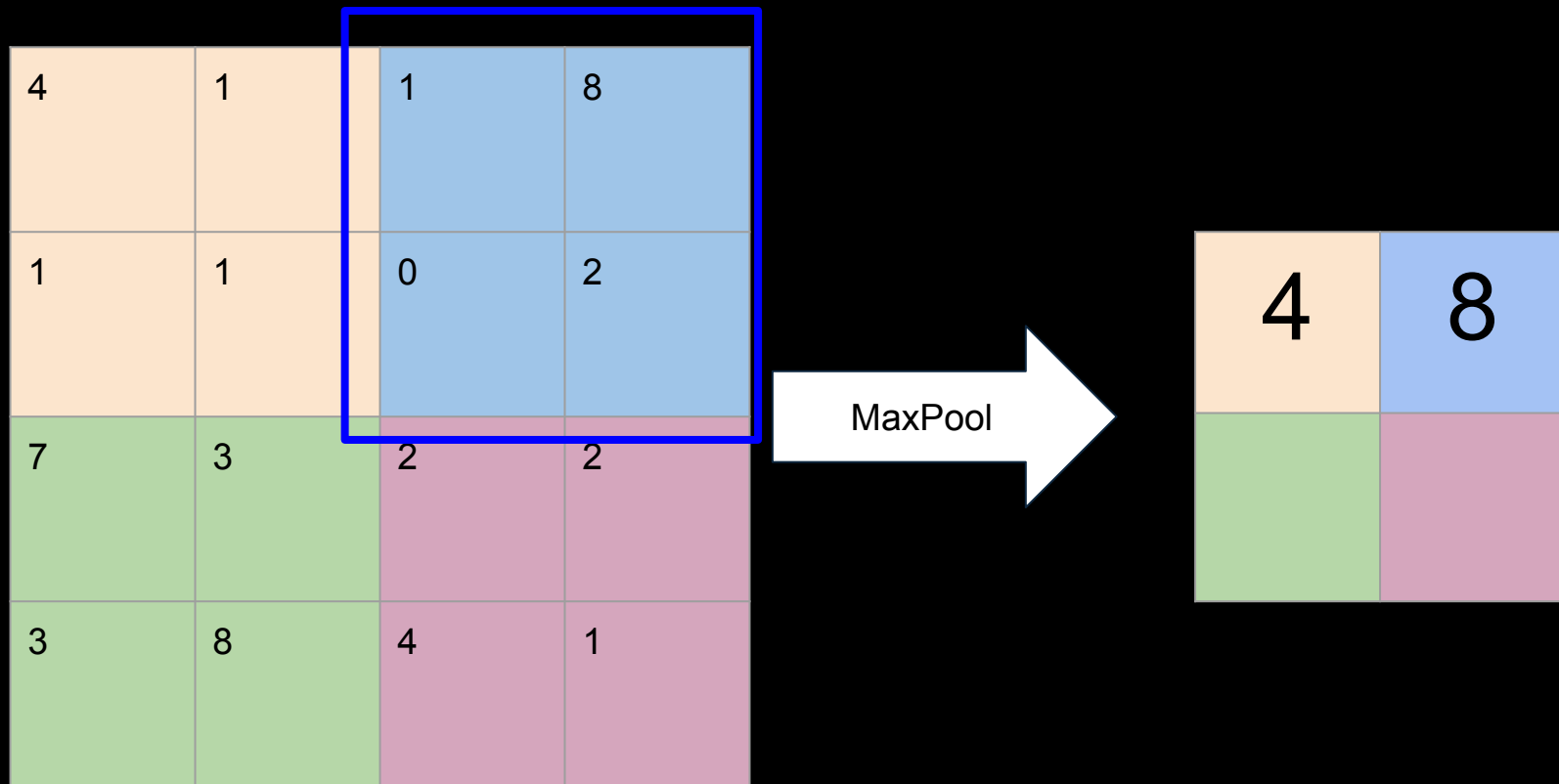
MaxPool

| | |
|---|--|
| 4 | |
| | |

MaxPool2D:

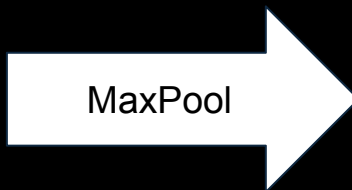


MaxPool2D:



MaxPool2D:

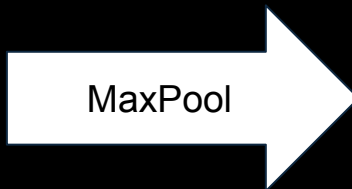
| | | | |
|---|---|---|---|
| 4 | 1 | 1 | 8 |
| 1 | 1 | 0 | 2 |
| 7 | 3 | 2 | 2 |
| 3 | 8 | 4 | 1 |



| | |
|---|--|
| 4 | |
| | |

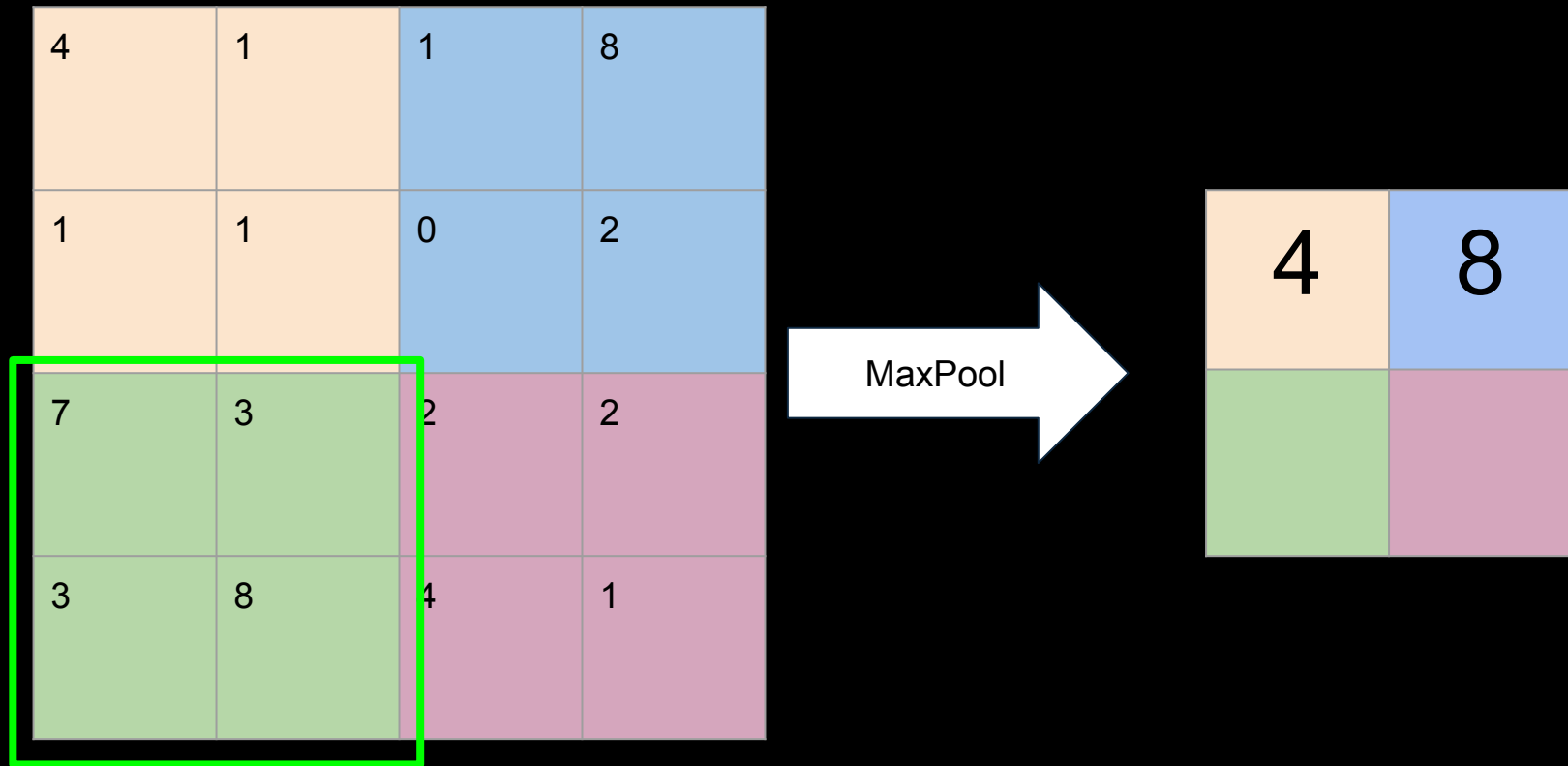
MaxPool2D:

| | | | |
|---|---|---|---|
| 4 | 1 | 1 | 8 |
| 1 | 1 | 0 | 2 |
| 7 | 3 | 2 | 2 |
| 3 | 8 | 4 | 1 |

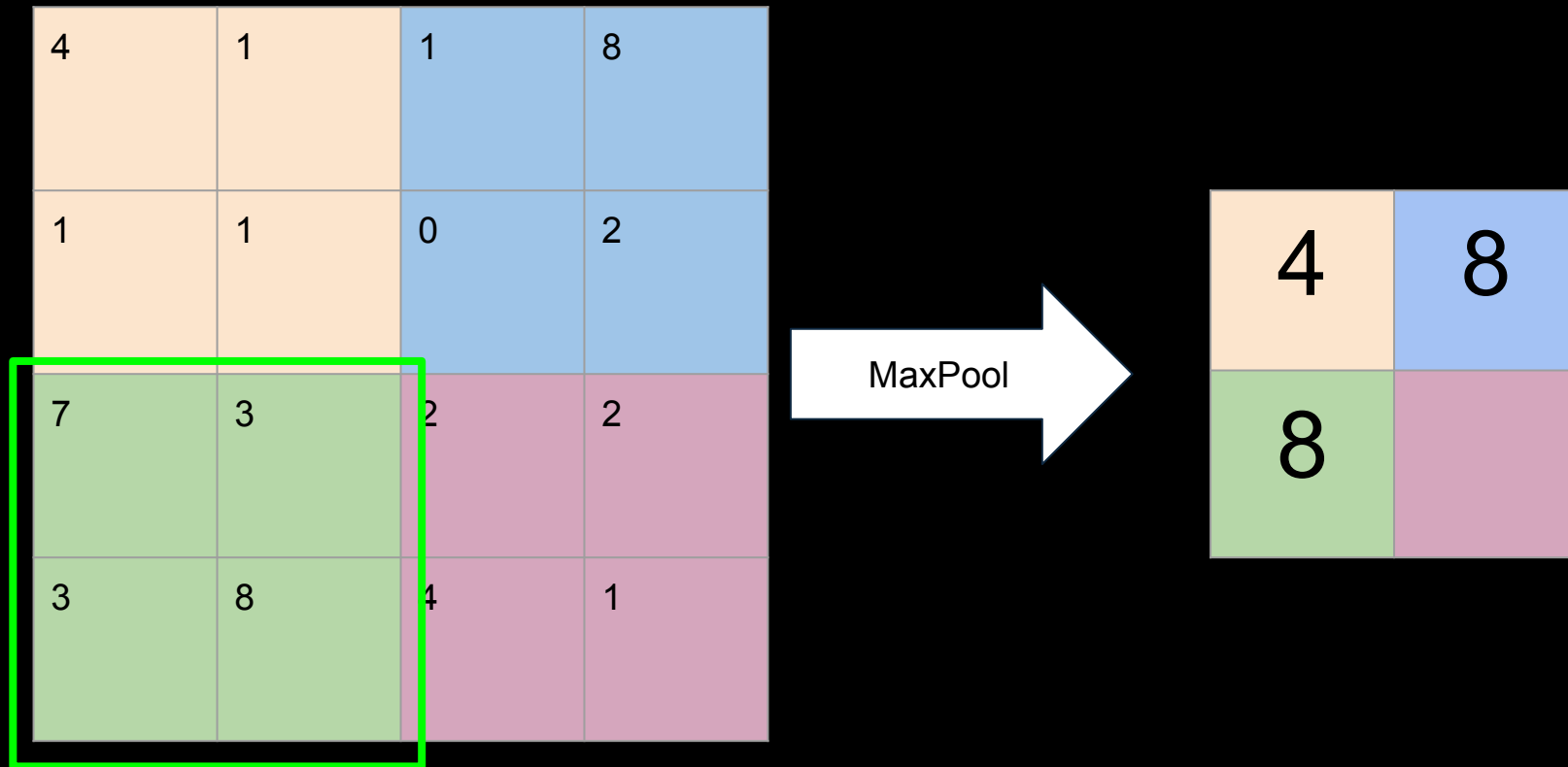


| | |
|---|---|
| 4 | 8 |
| | |

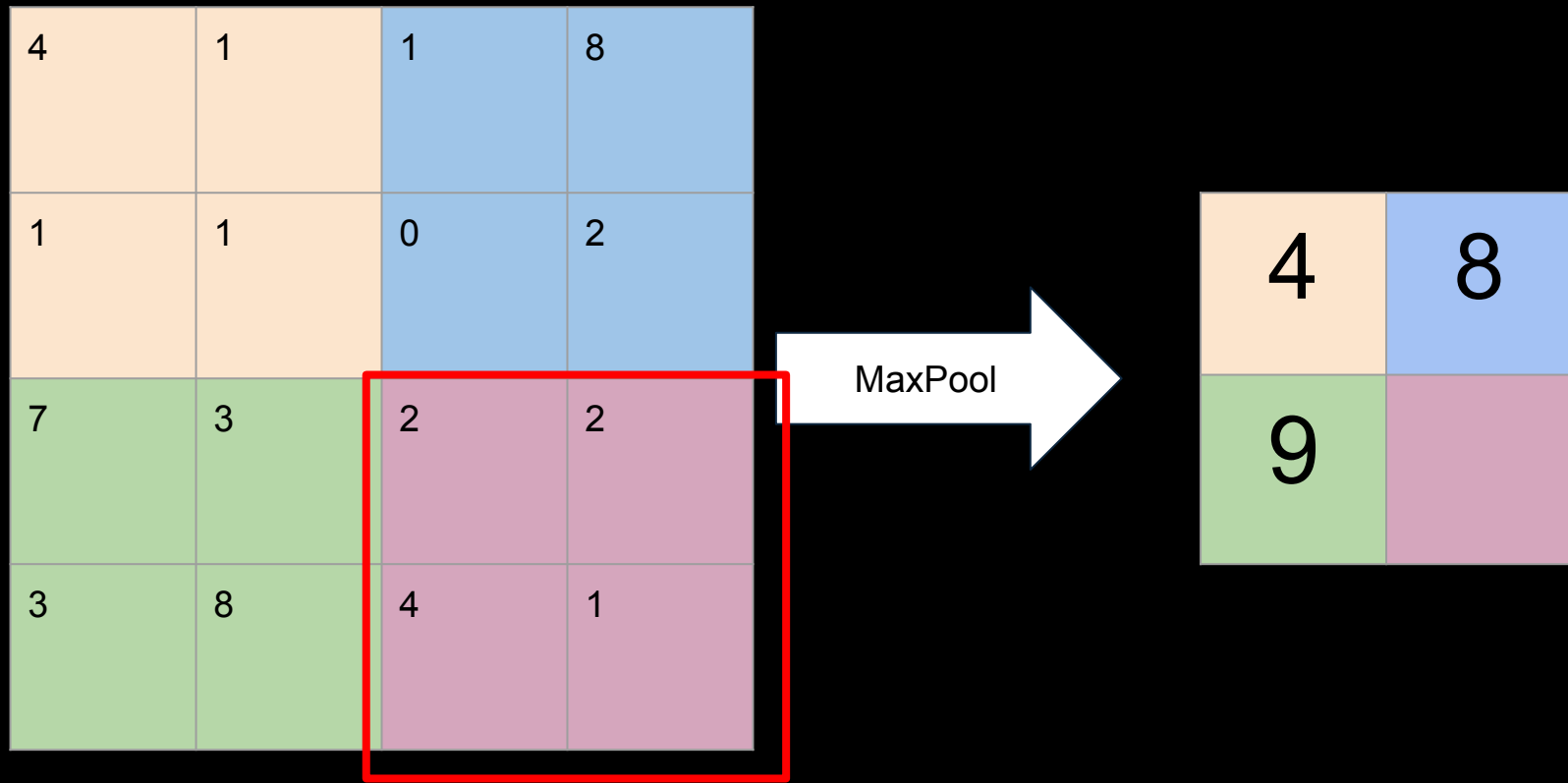
MaxPool2D:



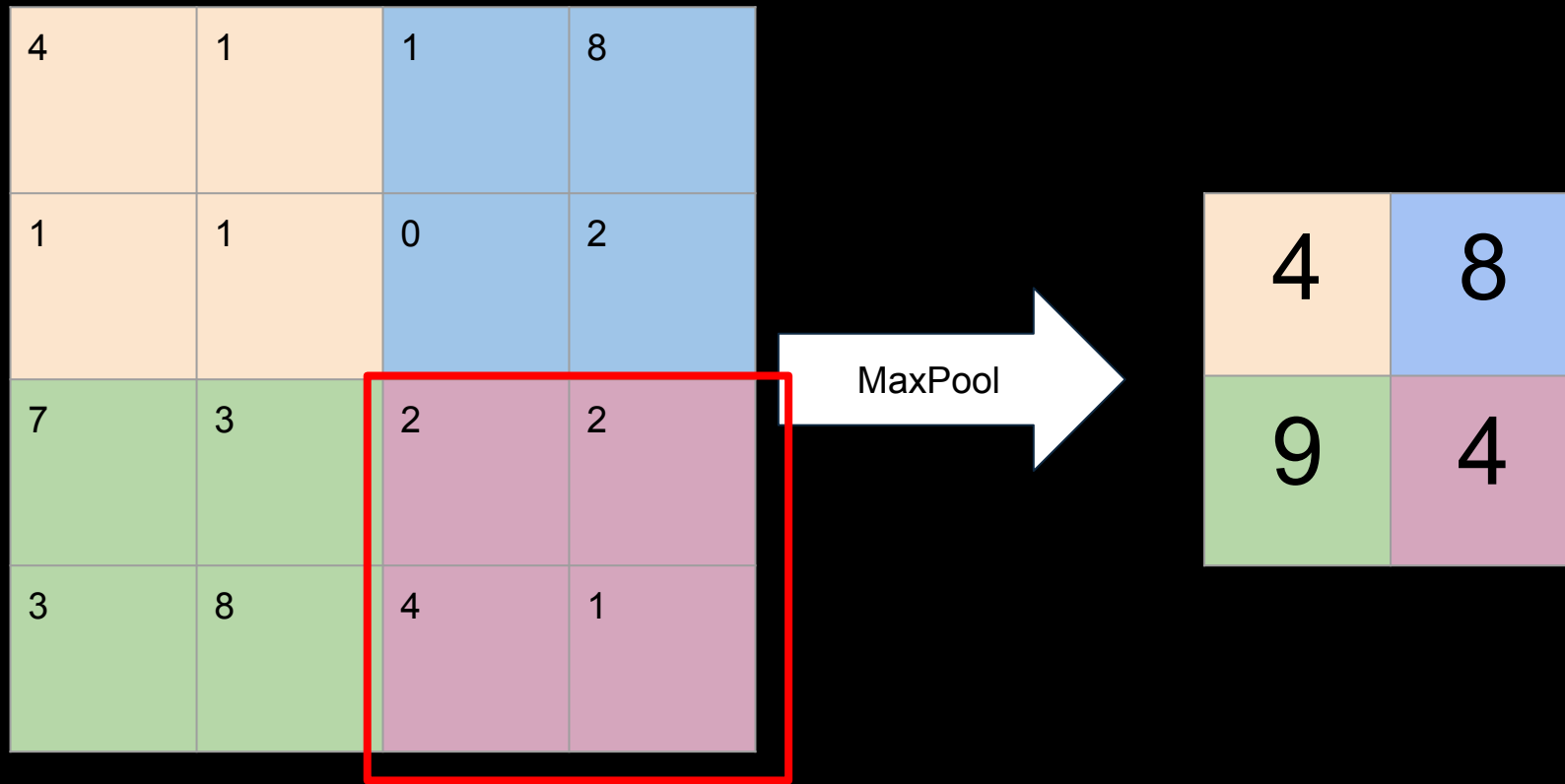
MaxPool2D:



MaxPool2D:

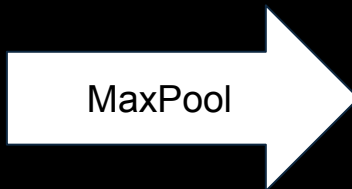


MaxPool2D:



MaxPool2D:

| | | | |
|---|---|---|---|
| 4 | 1 | 1 | 8 |
| 1 | 1 | 0 | 2 |
| 7 | 3 | 2 | 2 |
| 3 | 8 | 4 | 1 |



| | |
|---|---|
| 4 | 8 |
| 9 | 4 |

Batch Normalisation:

| | | | |
|---|---|---|---|
| 4 | 1 | 1 | 8 |
| 1 | 1 | 0 | 2 |
| 7 | 3 | 2 | 2 |
| 3 | 8 | 4 | 1 |