

# Часть 5:

# Сверточные нейронные сети



Романов Михаил, Игорь Слинько  
Mail.ru и МФТИ

# Зачем нам это все

- Сигмоидные полносвязанные нейронные сети полны. При помощи них можно восстановить любую зависимость с любой точностью

# Зачем нам это все

- Сигмоидные полносвязанные нейронные сети полны. При помощи них можно восстановить любую зависимость с любой точностью
- Но у нас ограниченные вычислительные возможности

# Зачем нам это все

- Сигмоидные полносвязанные нейронные сети полны. При помощи них можно восстановить любую зависимость с любой точностью
- Но у нас ограниченные вычислительные возможности
- Нужно использовать дополнительную информацию о структуре данных

# Зачем нам это все

- Сигмоидные полносвязанные нейронные сети полны. При помощи них можно восстановить любую зависимость с любой точностью
- Но у нас ограниченные вычислительные возможности
- Нужно использовать дополнительную информацию о структуре данных
- Поэтому мы используем несколько другой подход для работы со структурированными данными

# Зачем нам это все

- Сигмоидные полносвязанные нейронные сети полны. При помощи них можно восстановить любую зависимость с любой точностью
- Но у нас ограниченные вычислительные возможности
- Нужно использовать дополнительную информацию о структуре данных
- Поэтому мы используем несколько другой подход для работы со структурированными данными
- Используем соседство

# Свертка

5	2	3	1	-2
3	1	1	0	4
6	2	1	0	-1
4	7	2	1	0
3	6	2	2	1

# Свертка

0	0	0	0	0	0	0
0	5	2	3	1	-2	0
0	3	1	1	0	4	0
0	6	2	1	0	-1	0
0	4	7	2	1	0	0
0	3	6	2	2	1	0
0	0	0	0	0	0	0

1	0	-1
0	1	0
-1	0	1

6				

Padding=(1,1)



# Свертка

0	0	0	0	0	0	0
0	5	2	3	1	-2	0
0	3	1	1	0	4	0
0	6	2	1	0	-1	0
0	4	7	2	1	0	0
0	3	6	2	2	1	0
0	0	0	0	0	0	0

1	0	-1
0	1	0
-1	0	1

6				

# Свертка

0	0	0	0	0	0	0
0	5	2	3	1	-2	0
0	3	1	1	0	4	0
0	6	2	1	0	-1	0
0	4	7	2	1	0	0
0	3	6	2	2	1	0
0	0	0	0	0	0	0

1	0	-1
0	1	0
-1	0	1

6	0			

# Свертка

0	0	0	0	0	0	0
0	5	2	3	1	-2	0
0	3	1	1	0	4	0
0	6	2	1	0	-1	0
0	4	7	2	1	0	0
0	3	6	2	2	1	0
0	0	0	0	0	0	0

1	0	-1
0	1	0
-1	0	1

6	0	2		

# Свертка

0	0	0	0	0	0	0
0	5	2	3	1	-2	0
0	3	1	1	0	4	0
0	6	2	1	0	-1	0
0	4	7	2	1	0	0
0	3	6	2	2	1	0
0	0	0	0	0	0	0

1	0	-1
0	1	0
-1	0	1

6	0	2	4	-2

# Свертка

0	0	0	0	0	0	0
0	5	2	3	1	-2	0
0	3	1	1	0	4	0
0	6	2	1	0	-1	0
0	4	7	2	1	0	0
0	3	6	2	2	1	0
0	0	0	0	0	0	0

1	0	-1
0	1	0
-1	0	1

6	0	2	4	-2
3	-2	0	3	5

# Свертка

0	0	0	0	0	0	0
0	5	2	3	1	-2	0
0	3	1	1	0	4	0
0	6	2	1	0	-1	0
0	4	7	2	1	0	0
0	3	6	2	2	1	0
0	0	0	0	0	0	0

1	0	-1
0	1	0
-1	0	1

6	0	2	4	-2
3	-2	0	3	5
12	1	-4	-5	-2
8	11	0	2	-2
-4	8	8	4	2

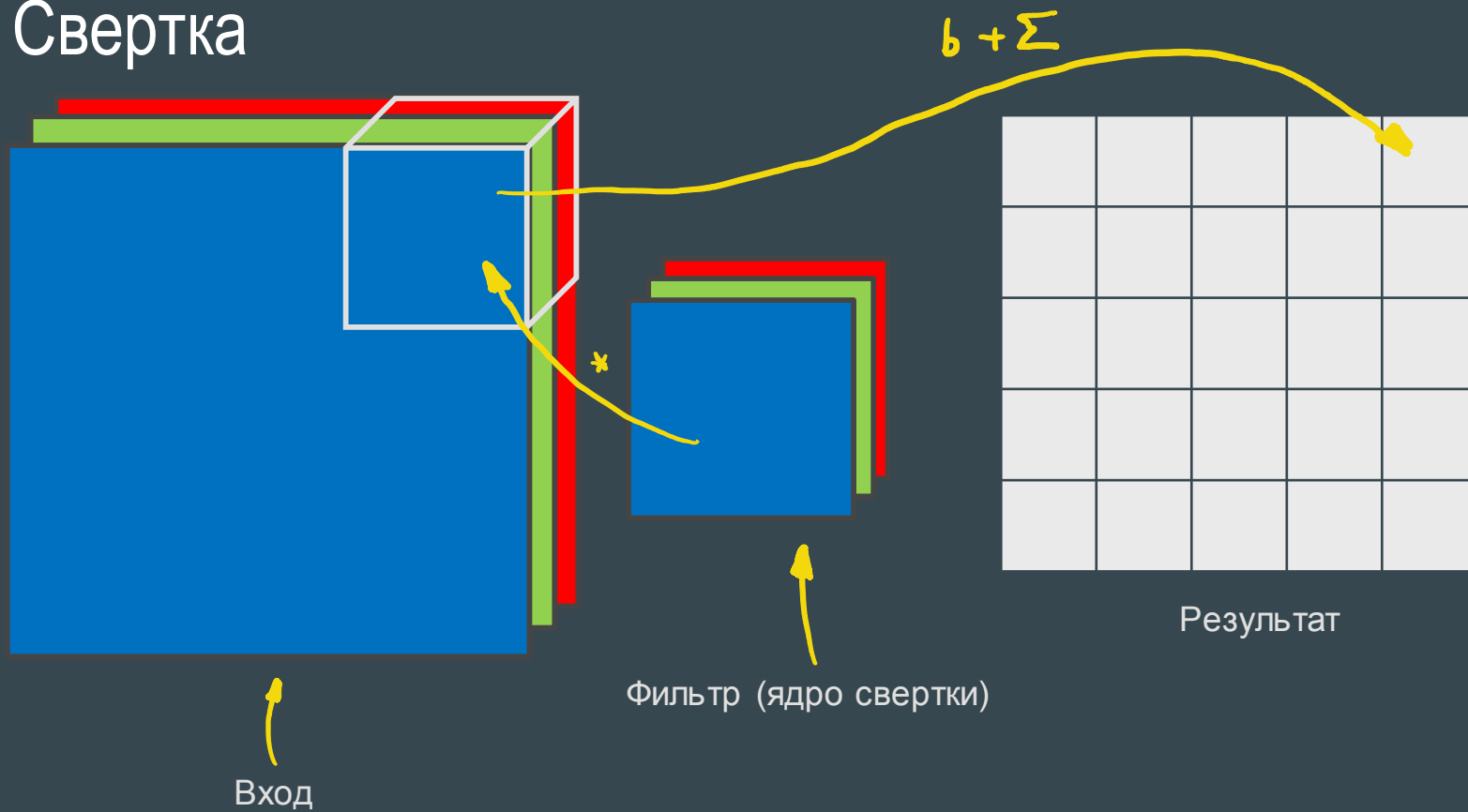
# Свертка

0	0	0	0	0	0	0
0	5	2	3	1	-2	0
0	3	1	1	0	4	0
0	6	2	1	0	-1	0
0	4	7	2	1	0	0
0	3	6	2	2	1	0
0	0	0	0	0	0	0

1	0	-1
0	1	0
-1	0	1

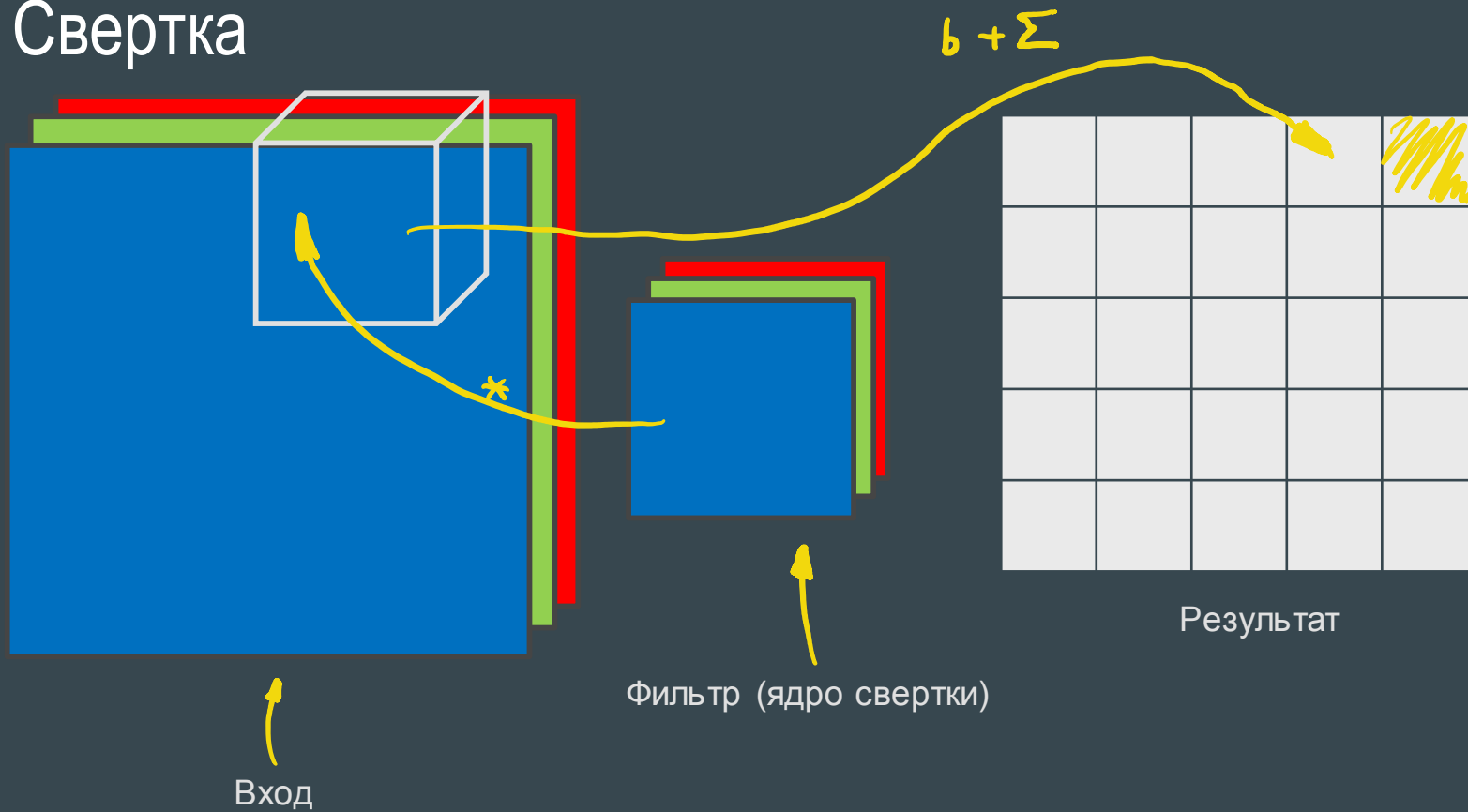
6	2	-2
12	-4	-2
-4	8	2

# Свертка

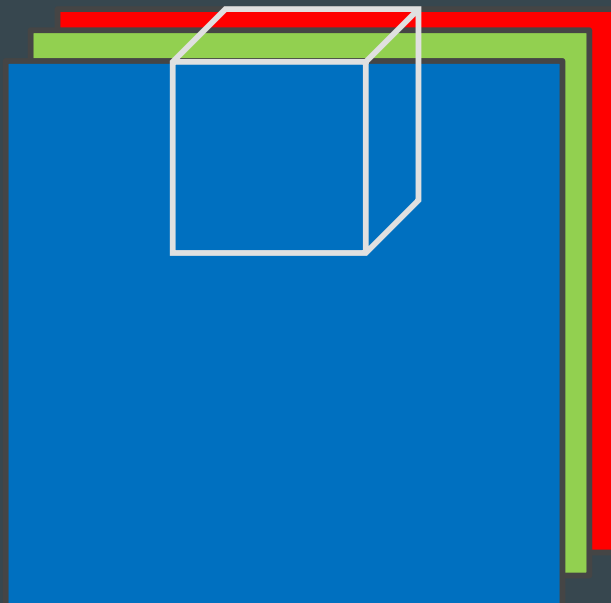




# Свертка



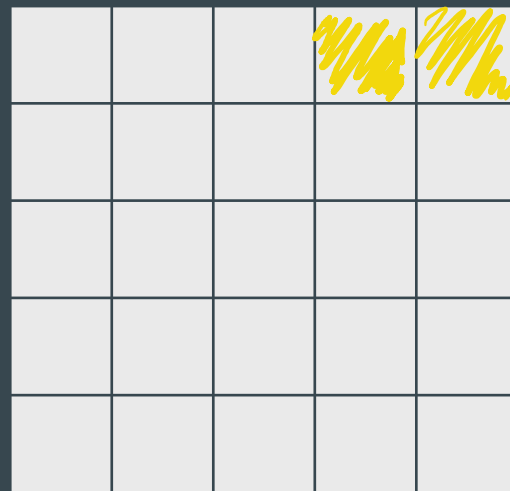
# Свертка



Вход

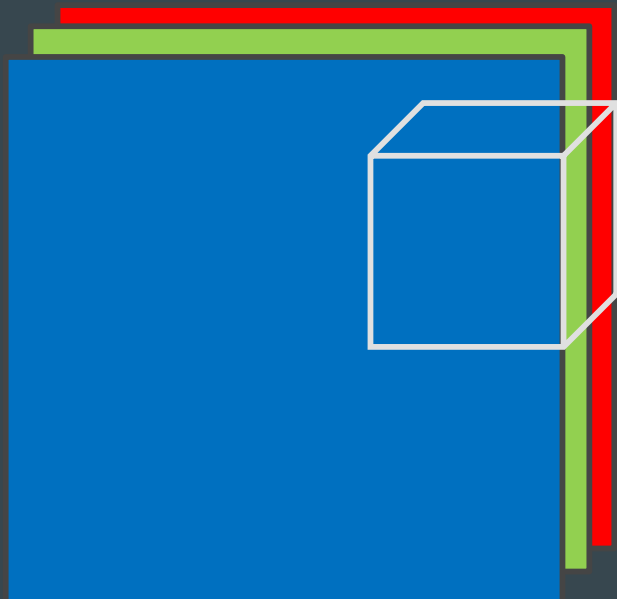


Фильтр (ядро свертки)



Результат

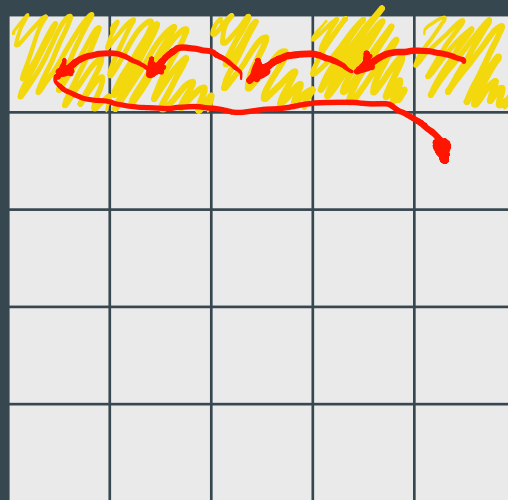
# Свертка



Вход

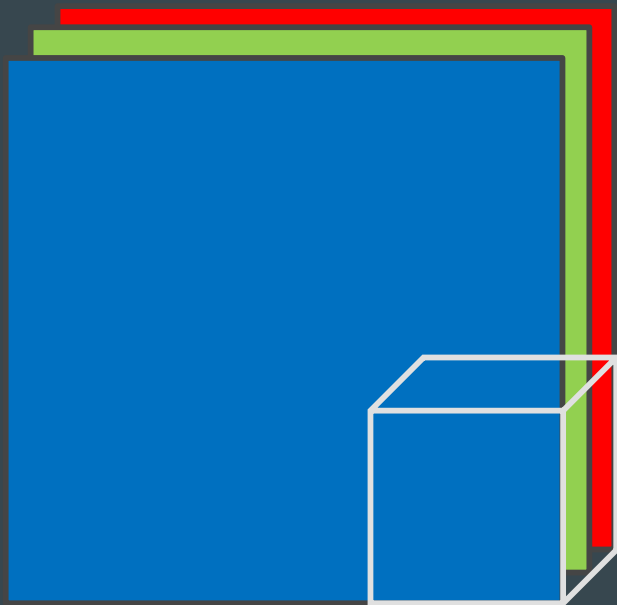


Фильтр (ядро свертки)



Результат

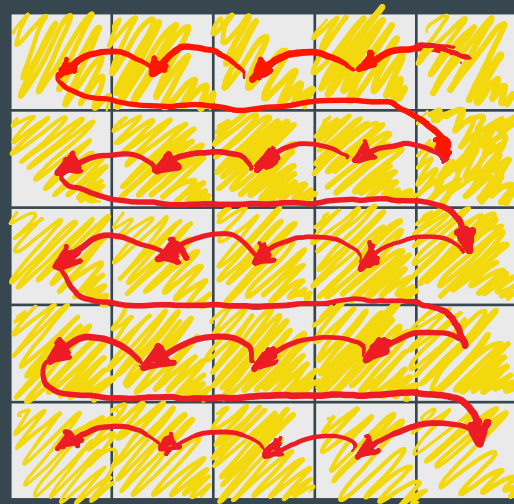
# Свертка



Вход

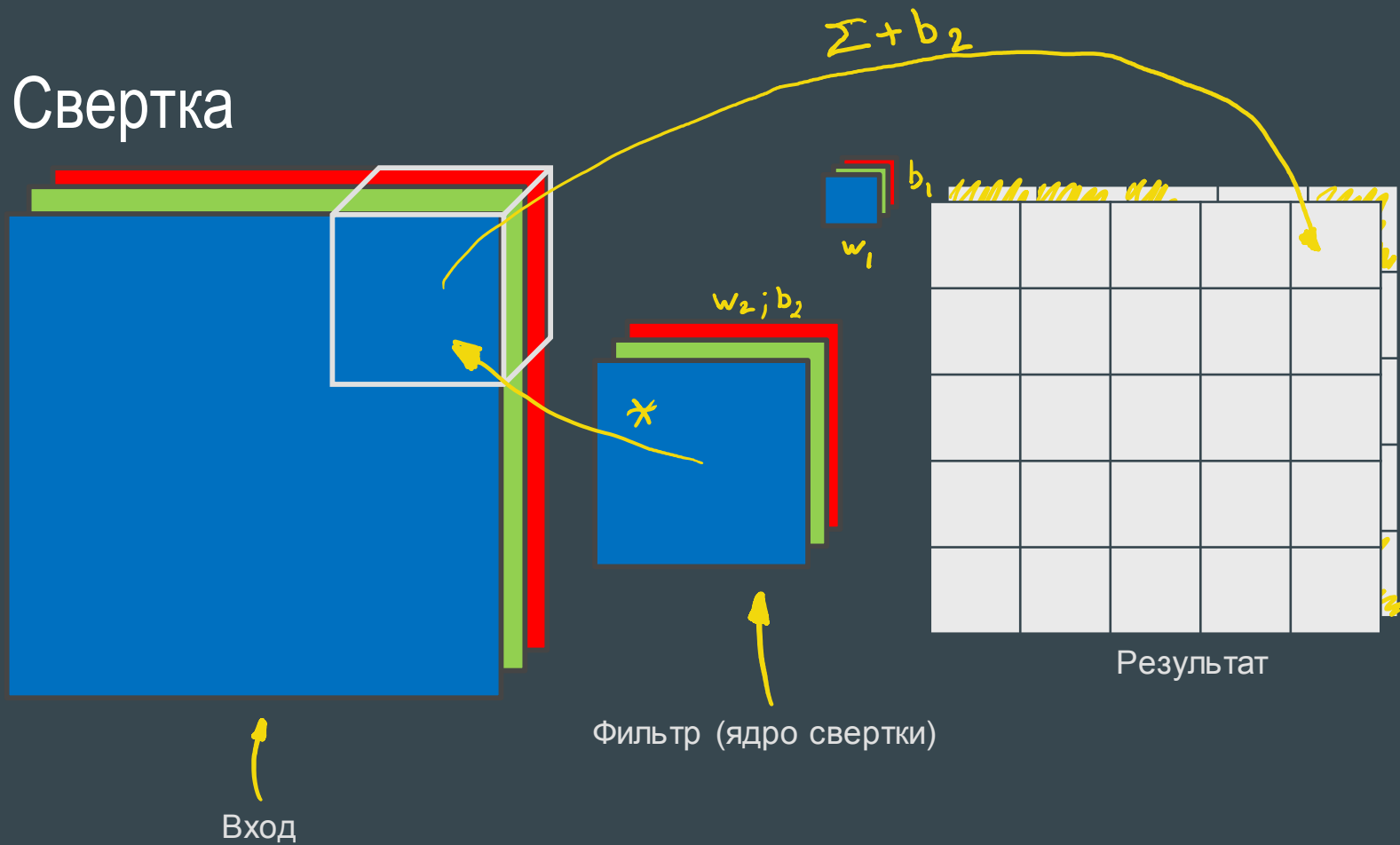


Фильтр (ядро свертки)

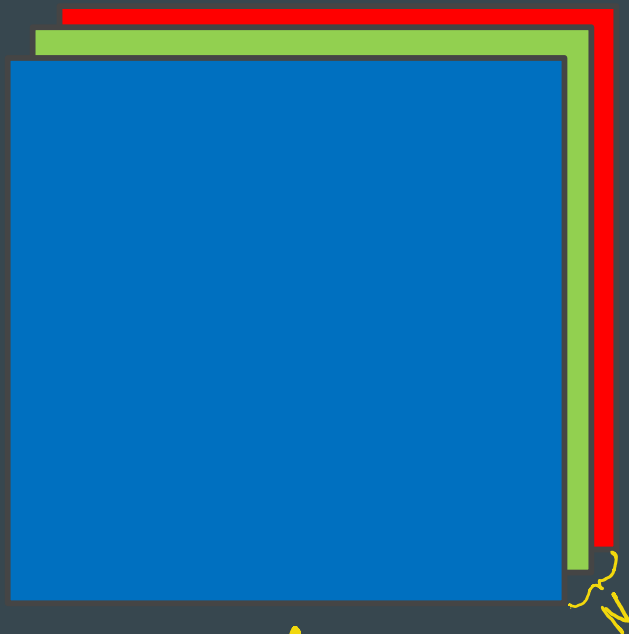


Результат

# Свертка

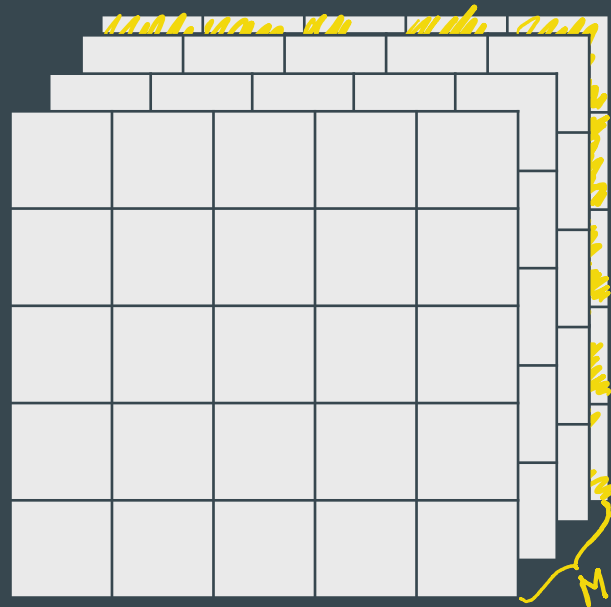


# Свертка



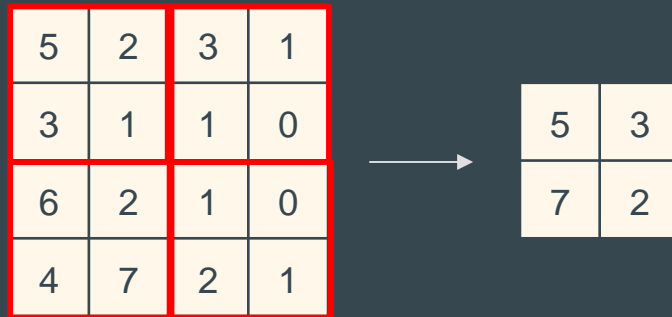
Вход

$Conv(N, M)$

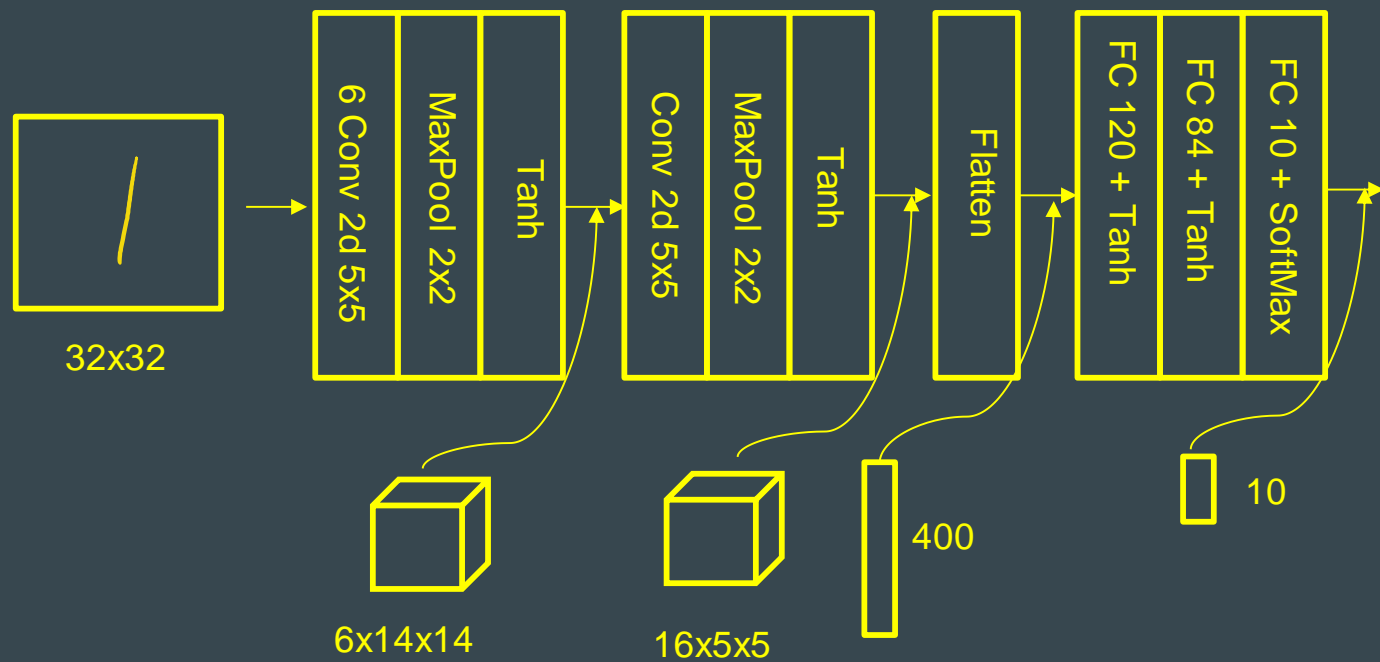


# Пулинг

MaxPool(2,2, stride=(2,2))



# LeNet





# Итоги

- Свертка
- Пулинг
- Сверточная сеть LeNet