LECTURE 18

WINTER 2021
APPLIED MACHINE LEARNING
CIHANG XIE

SLIDE CREDIT:
NARGES NOROUZI

GROUP ACTIVITIES

- Week 9 is the last week that you need to turn in your files
- Most groups are doing extremely great

CLASS PARTICIPATION

IF you submitted all polling questions:

you are good 😀

ELSEIF #(zoom participations less than 1hr) < 3

you are good 😀

ELSE

you may lose participation credit

Come to my next Tuesday office hour

-- send me an email for booking this appointment

EXTRA CREDITS

Course Feedback

Student Experience of Teaching Surveys - Spring 2021 for CSE-144-01-64331-Applied ML

Medium	Online
Timing	Scheduled

- Start Date 2021-05-24 08:00
- End Date 2021-06-06 23:59

Response Rate

	Responded	Invited	% Rate
Students	20	102	19.61%

60+ responded --- everyone get 1pt

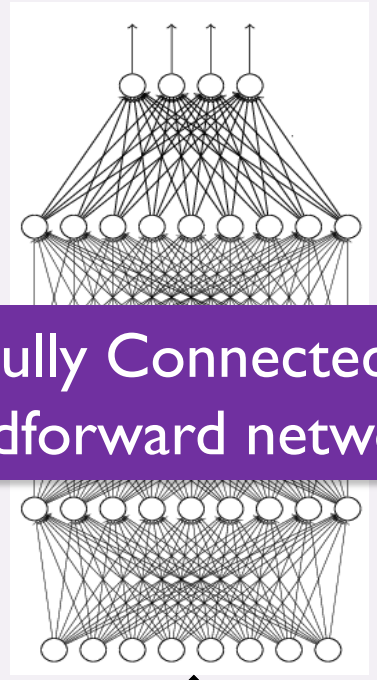
80+ responded --- everyone get 2pts

All responded --- everyone get 3pts

TODAY

- Convolutional Neural Network
  - TensorFlow implementation
- Transfer Learning
- Other Training Considerations
  - Batch normalization
  - Proper weight initialization

THE WHOLE CNN



cat dog ……

Fully Connected Feedforward network

Flatten

Convolution

Max Pooling

Convolution

Max Pooling

Can repeat many times

THE WHOLE CNN

Input

7	3	5	2
8	7	1	6
4	9	3	9
0	8	4	5

maxpool

Output

8	6
9	9

Property 1

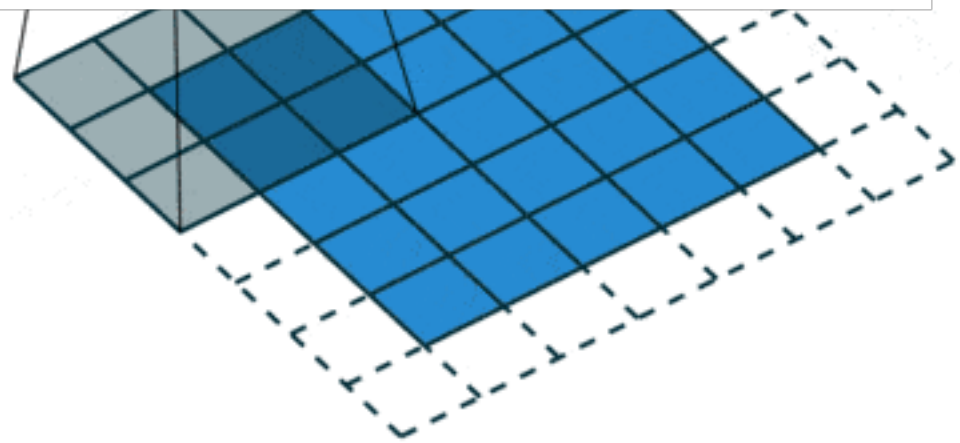
- Some pat
  smaller th

Property 2

- The same
  different r

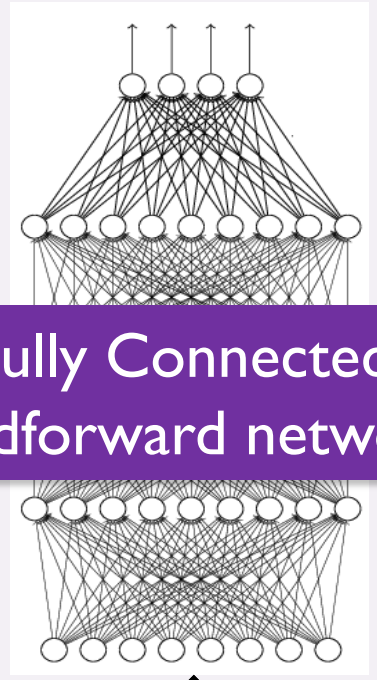
Property 3

- Subsampling the pixels will not change the object



Flatten

THE WHOLE CNN



cat dog ……

Fully Connected Feedforward network

Flatten

Convolution

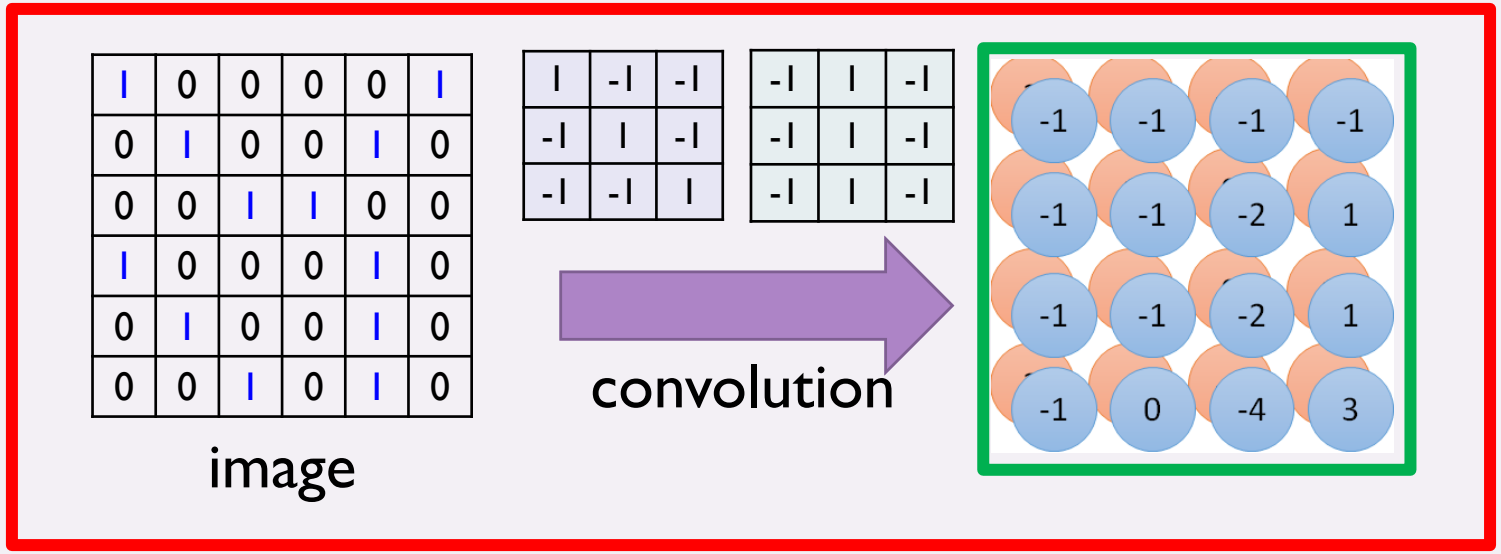
Max Pooling

Convolution

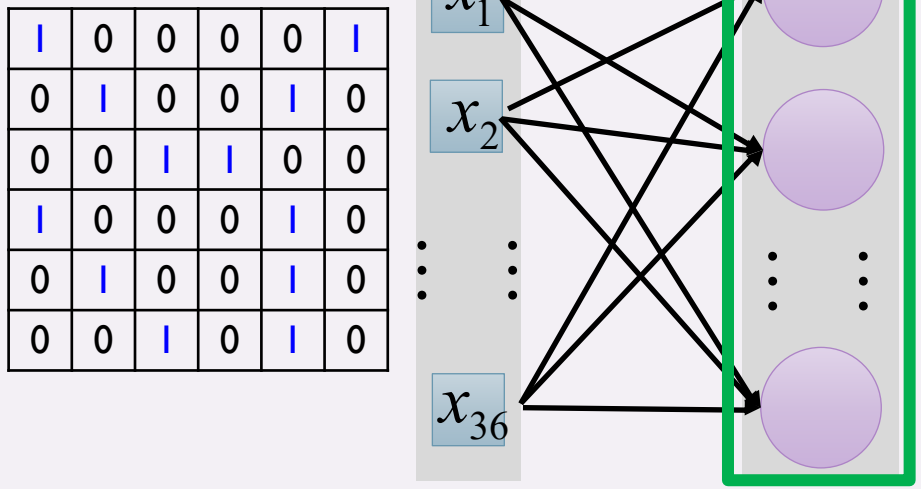
Max Pooling

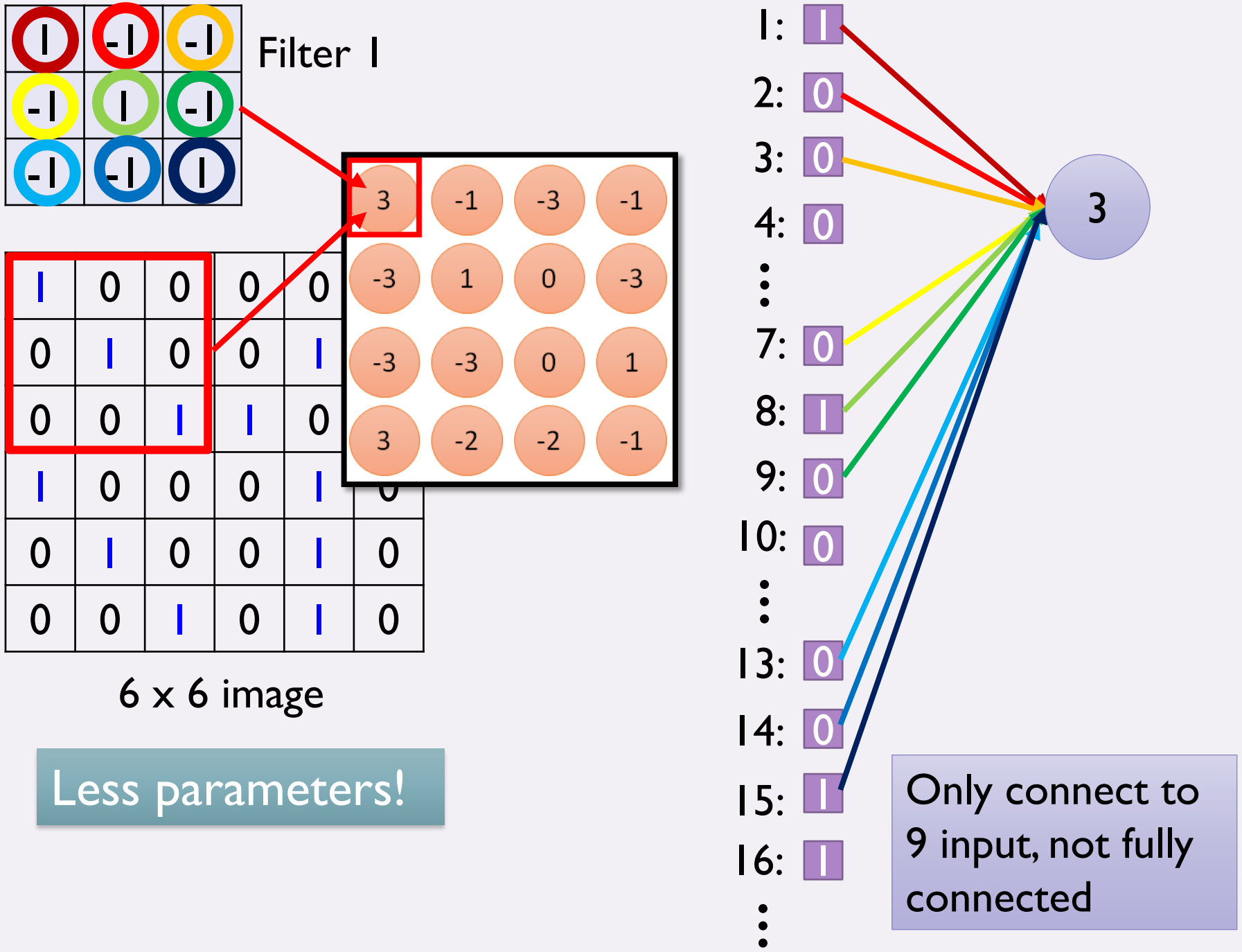
Can repeat many times

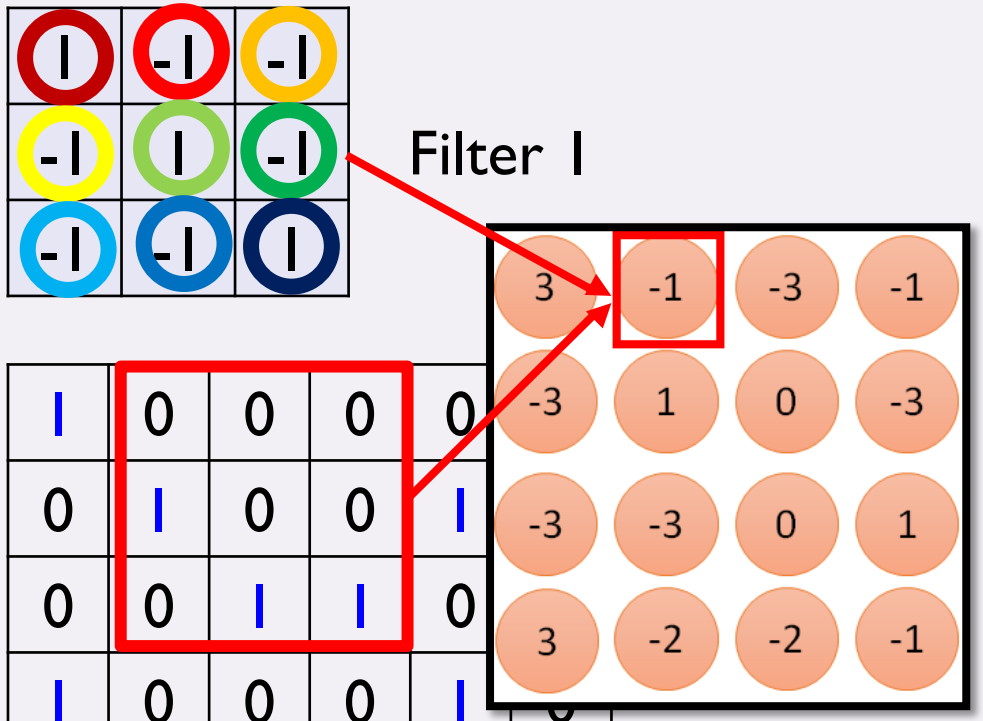
CONVOLUTION VS FULLY-CONNECTED



Fully-connected

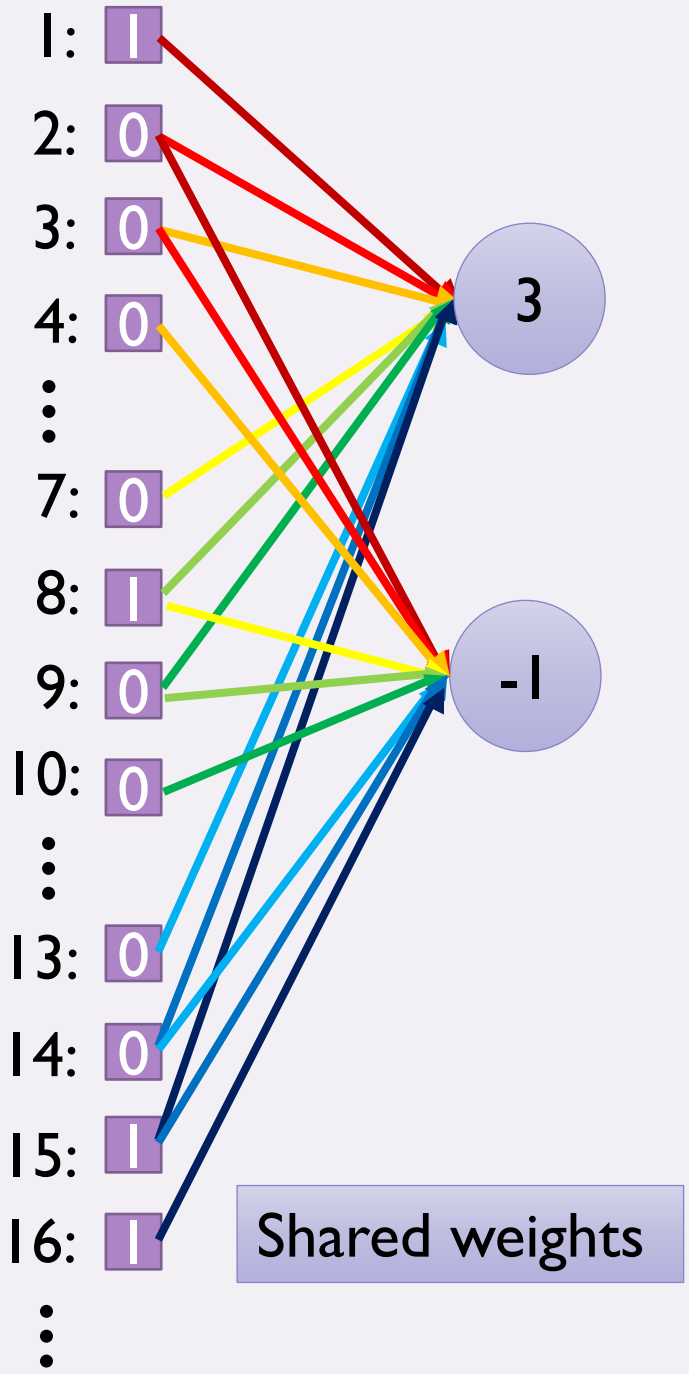



1 -1 -1
-1 1 -1
-1 -1 1
Filter 1
3 -1 -3 -1
-3 1 0 -3
-3 -3 0 1
3 -2 -2 -1
1 0 0 0 0
0 1 0 0 1
0 0 1 1 0
1 0 0 0 1 0
0 1 0 0 1 0
0 0 1 0 1 0
6 x 6 image
Less parameters!
1:
2:
3:
4:
7:
8:
9:
10:
13:
14:
15:
16:
3
Only connect to 9 input, not fully connected

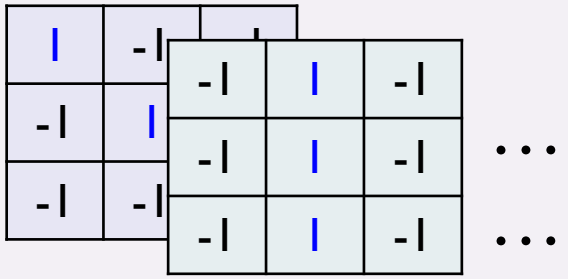


Less parameters!

Even less parameters!



```
model.add(Conv2D(filters = 25,
kernel_size = (3, 3),
padding = 'same',
activation = 'relu',
input_shape = (28, 28, 1)))
```

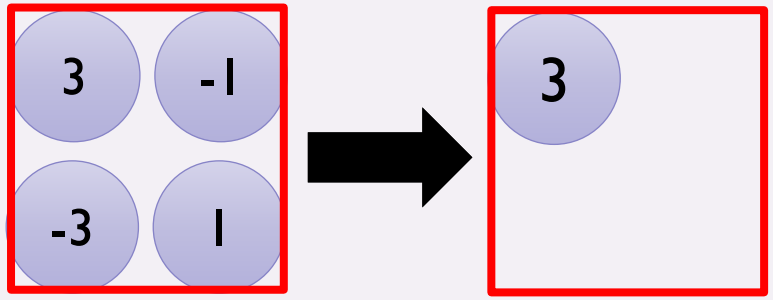


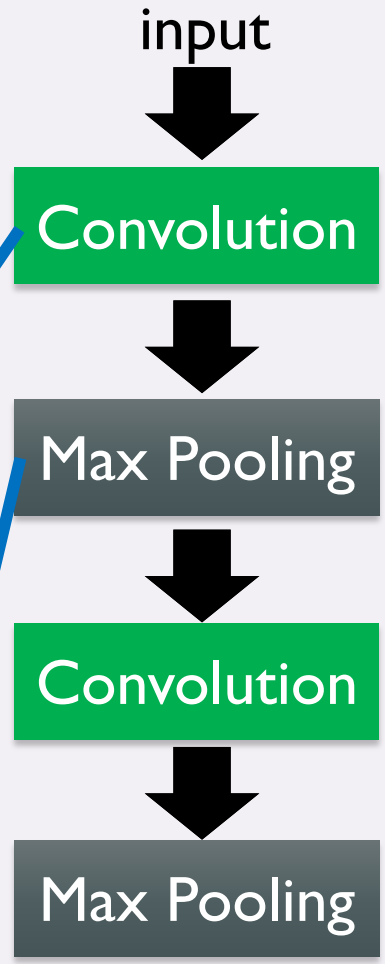
...

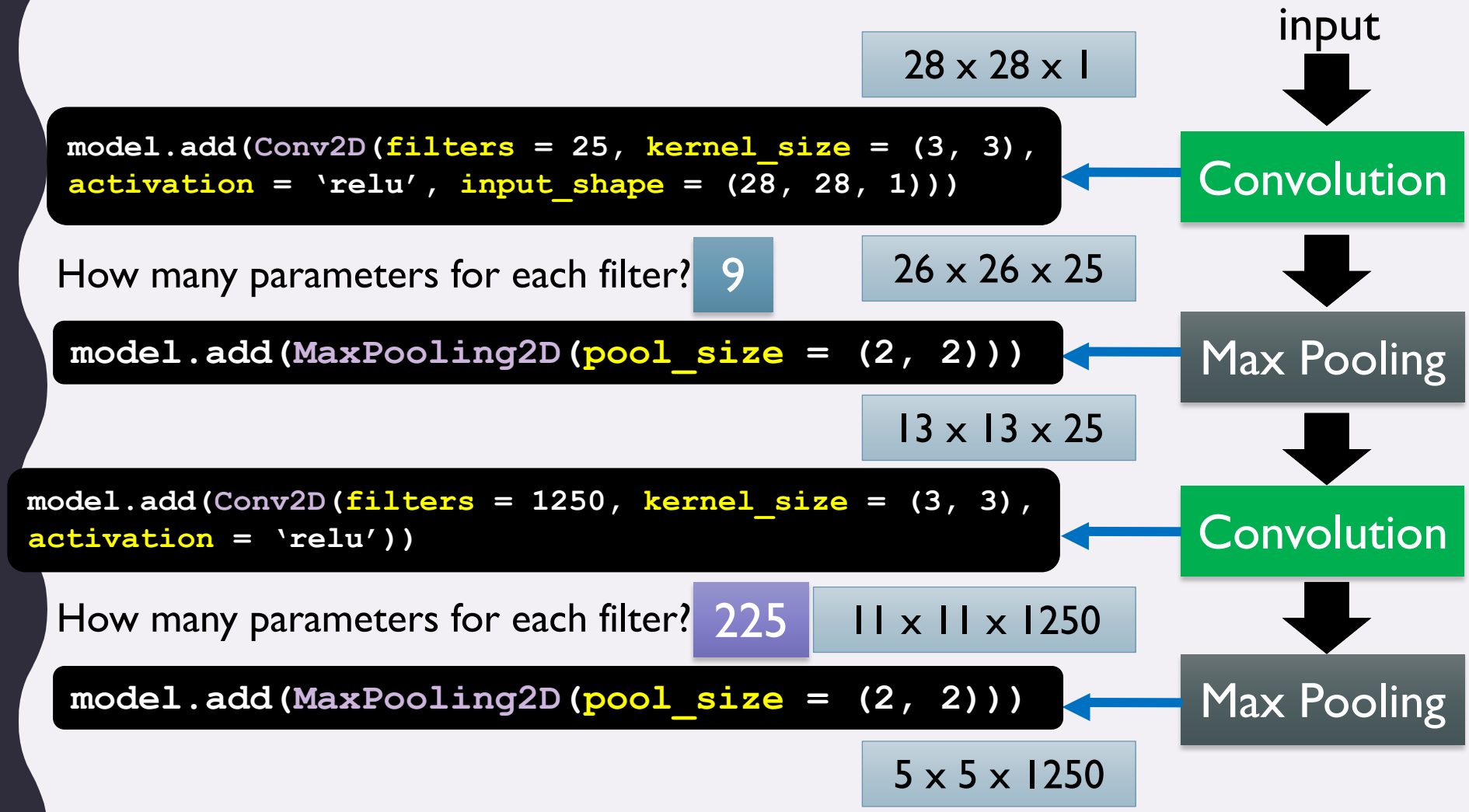
...

There are 25 3x3 filters.

```
model.add(MaxPooling2D(pool_size = (2, 2)))
```






input
28 x 28 x 1
model.add(Conv2D(filters = 25, kernel_size = (3, 3), activation = 'relu', input_shape = (28, 28, 1)))
Convolution
How many parameters for each filter?
9
26 x 26 x 25
model.add(MaxPooling2D(pool_size = (2, 2)))
Max Pooling
13 x 13 x 25
model.add(Conv2D(filters = 1250, kernel_size = (3, 3), activation = 'relu'))
Convolution
How many parameters for each filter?
225
11 x 11 x 1250
model.add(MaxPooling2D(pool_size = (2, 2)))
Max Pooling
5 x 5 x 1250

input

28 x 28 x 1

Convolution

26 x 26 x 25

Max Pooling

13 x 13 x 25

Convolution

11 x 11 x 1250

Max Pooling

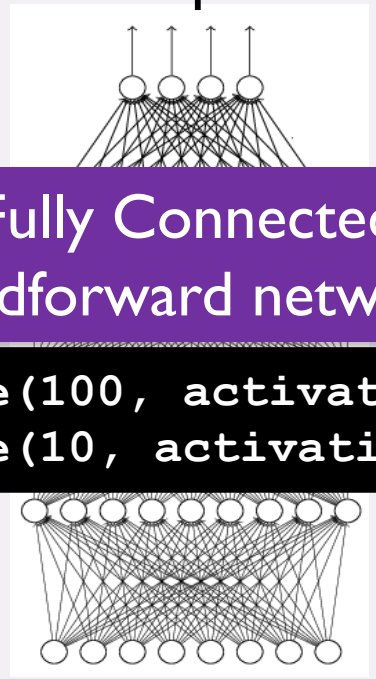
5 x 5 x 1250

```
model.add(Flatten())
```

Flatten

31250

output



Fully Connected Feedforward network

```
model.add(Dense(100, activation='relu'))
model.add(Dense(10, activation='softmax'))
```

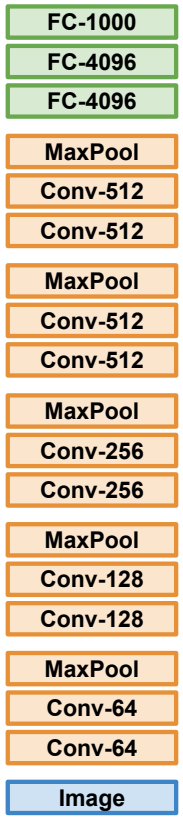
TRANSFER LEARNING

TRANSFER LEARNING

- A shortcut in training neural networks for recognition tasks
- The idea is to start with a fully trained image recognition neural network, off the shelf with trained weights.
- We can repurpose the trained network for our particular recognition task.
- What was learned by the neural network in it's early layers are useful features in recognizing various things in images.
- Keras even has pretrained models built in for this purpose:
  - Xception, VGG16, VGG19, ResNet, InceptionV3, InceptionResNetV2, MobileNet, DenseNet, NASNet

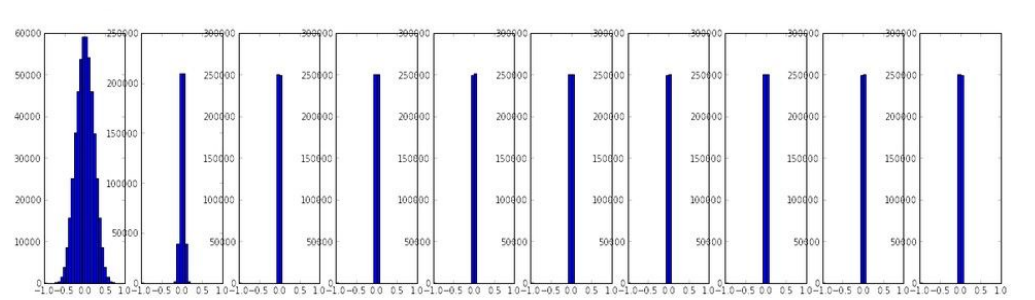
TRANSFER LEARNING WITH CNN

1. Train on Imagenet



OTHER TRAINING CONSIDERATIONS

PROPER WIGHT INITIALIZATION



- **Initialization too small**: Activations go to zero, gradients also zero → No learning

```
# Usage in a Keras layer:
initializer = tf.keras.initializers.HeNormal()
layer = tf.keras.layers.Dense(3, kernel_initializer=initializer)
```

Delving Deep into Rectifiers:
Surpassing Human-Level Performance on ImageNet Classification

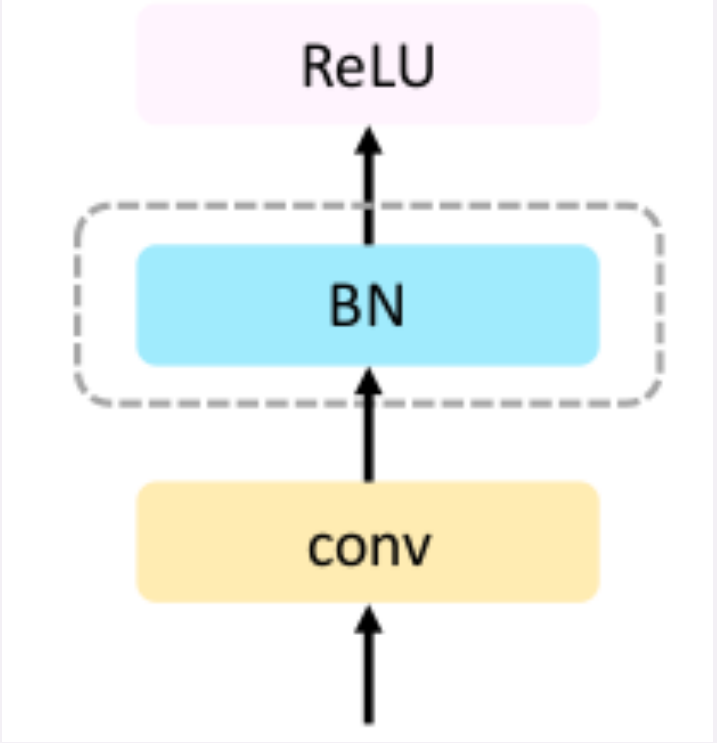
Kaiming He Xiangyu Zhang Shaoqing Ren Jian Sun

Microsoft Research

{kahe, v-xiangz, v-shren, jiansun}@microsoft.com

BATCH NORMALIZATION

- Distribution of each layer's input changes during training, as parameters of previous layers change → Internal covariate shift
- We try to fix this problem by normalization:
  - Input x: $N \times D$
  - Learnable Parameters γ, β: D
  - Intermediates μ, σ: D
  - $\hat{x}$: $N \times D$
  - Output y: $N \times D$



EXTRA CREDITS

Course Feedback

Student Experience of Teaching Surveys - Spring 2021 for CSE-144-01-64331-Applied ML

Medium	Online
Timing	Scheduled

- Start Date 2021-05-24 08:00
- End Date 2021-06-06 23:59

Response Rate

	Responded	Invited	% Rate
Students	2	102	1.96%

60+ responded --- everyone get 1pt
80+ responded --- everyone get 2pts
All responded --- everyone get 3pts

EXERCISE

HTTPS://BIT.LY/2RKUDRQ

QUESTIONS?