Capsule Networks

Aurélien Géron, November 2017

https://youtu.be/pPN8d0E3900

NIPS 2017 Paper

Dynamic Routing Between Capsules

by Sara Sabour, Nicholas Frosst, Geoffrey E. Hinton

October 2017: https://arxiv.org/abs/1710.09829

Aurélien Géron, 2017

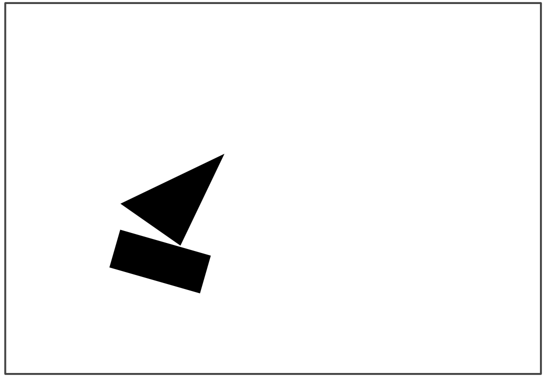
Computer Graphics

Rectangle

x=20
y=30
angle=16°

Triangle

x=24
y=25
angle=-65°



Instantiation parameters

Rendering

Image

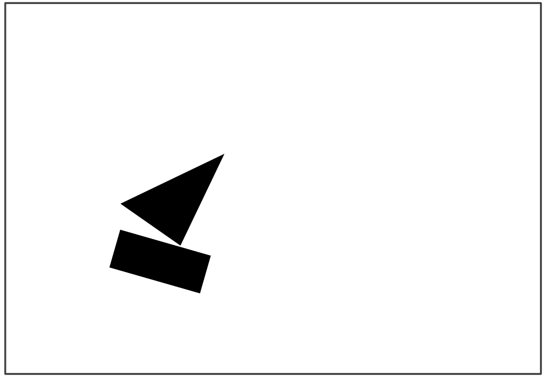
Inverse Graphics

Rectangle

x=20
y=30
angle=16°

Triangle

x=24
y=25
angle=-65°

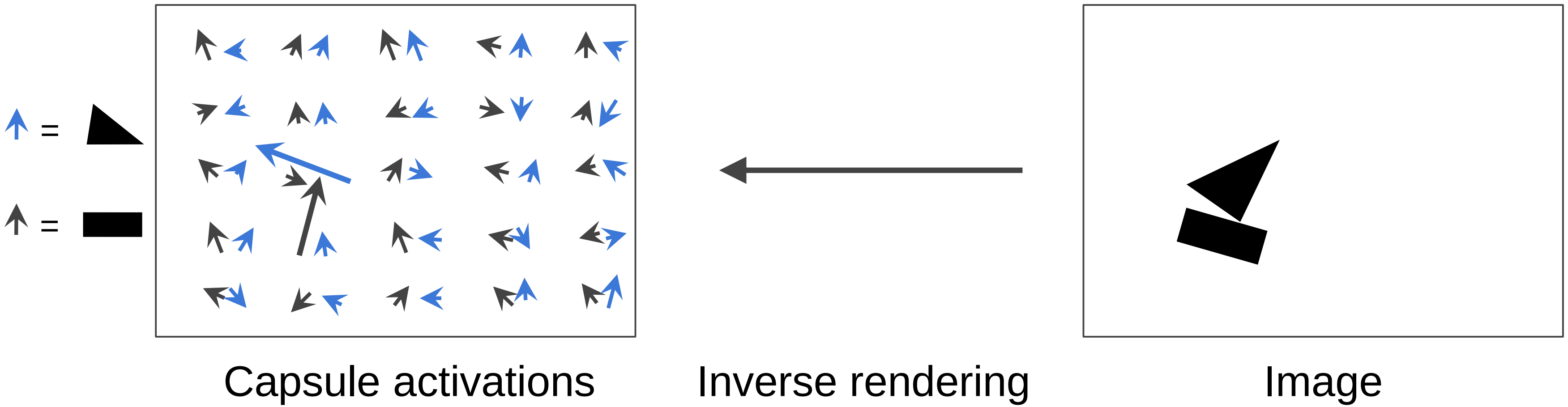


Instantiation parameters

Inverse rendering

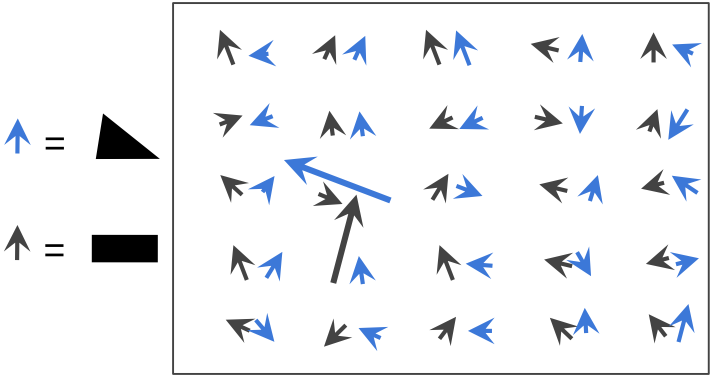
Image

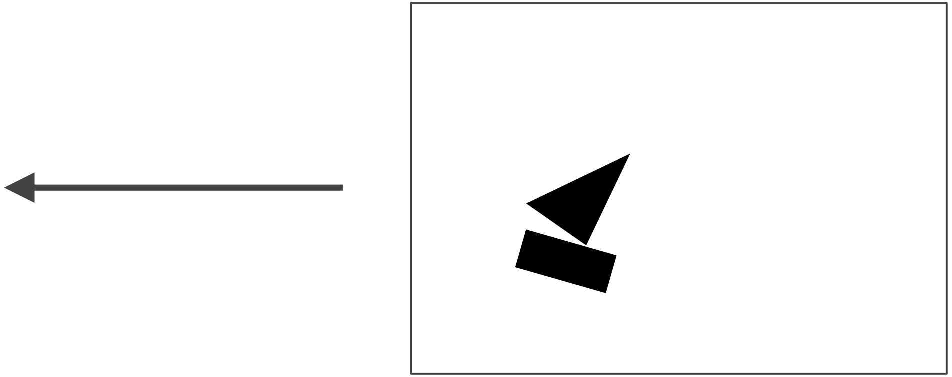
Capsules



Aurélien Géron, 2017

Capsules

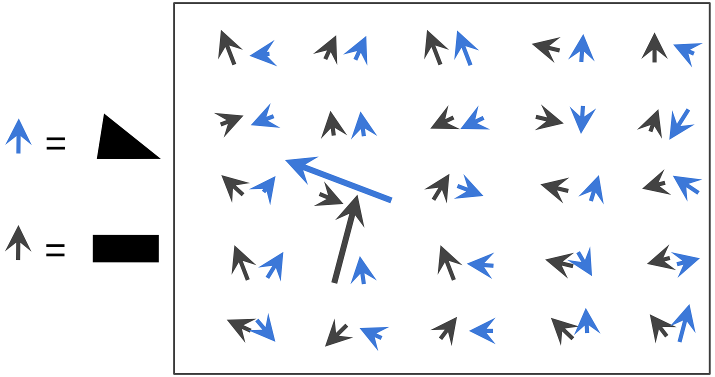




Activation vector:

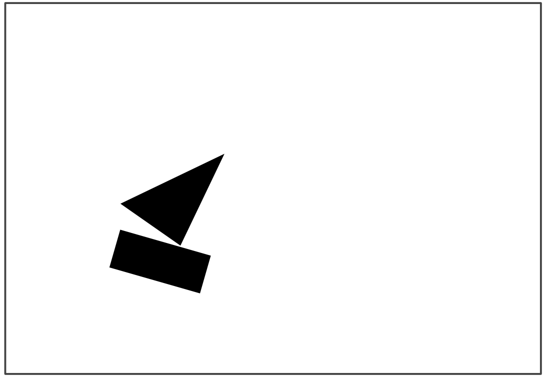
Length = estimated probability of presence
Orientation = object's estimated pose parameters

Capsules



Convolutional Layers

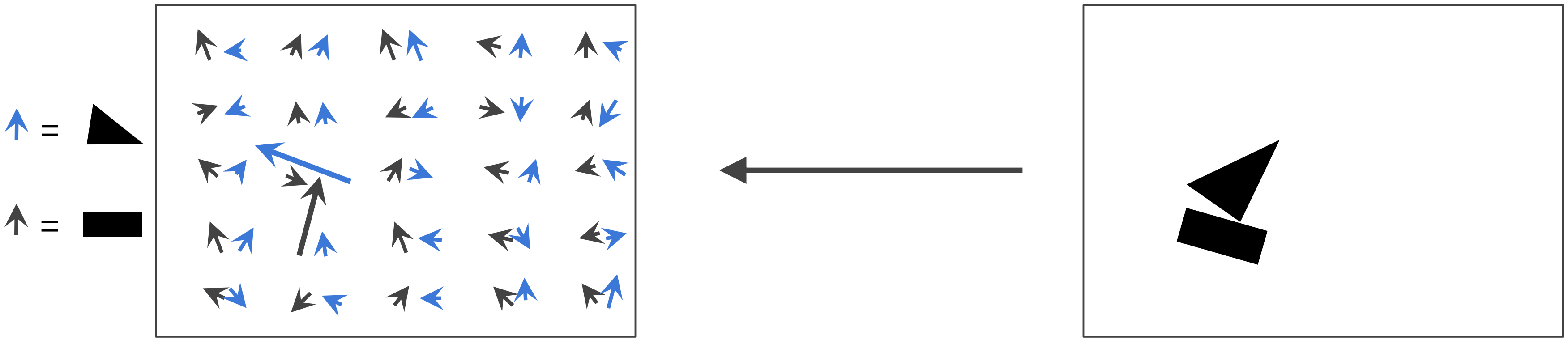
\+ Reshape
\+ Squash



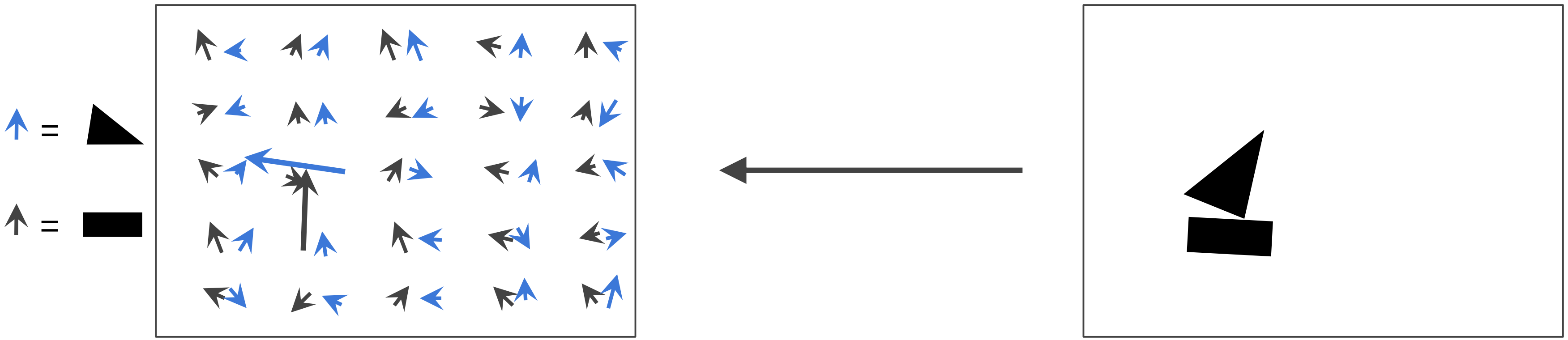
$$\text{Squash}(\mathbf{u}) = \frac{\|\mathbf{u}\|^2}{1 + \|\mathbf{u}\|^2} \frac{\mathbf{u}}{\|\mathbf{u}\|}$$

Aurélien Géron, 2017

Equivariance



Equivariance



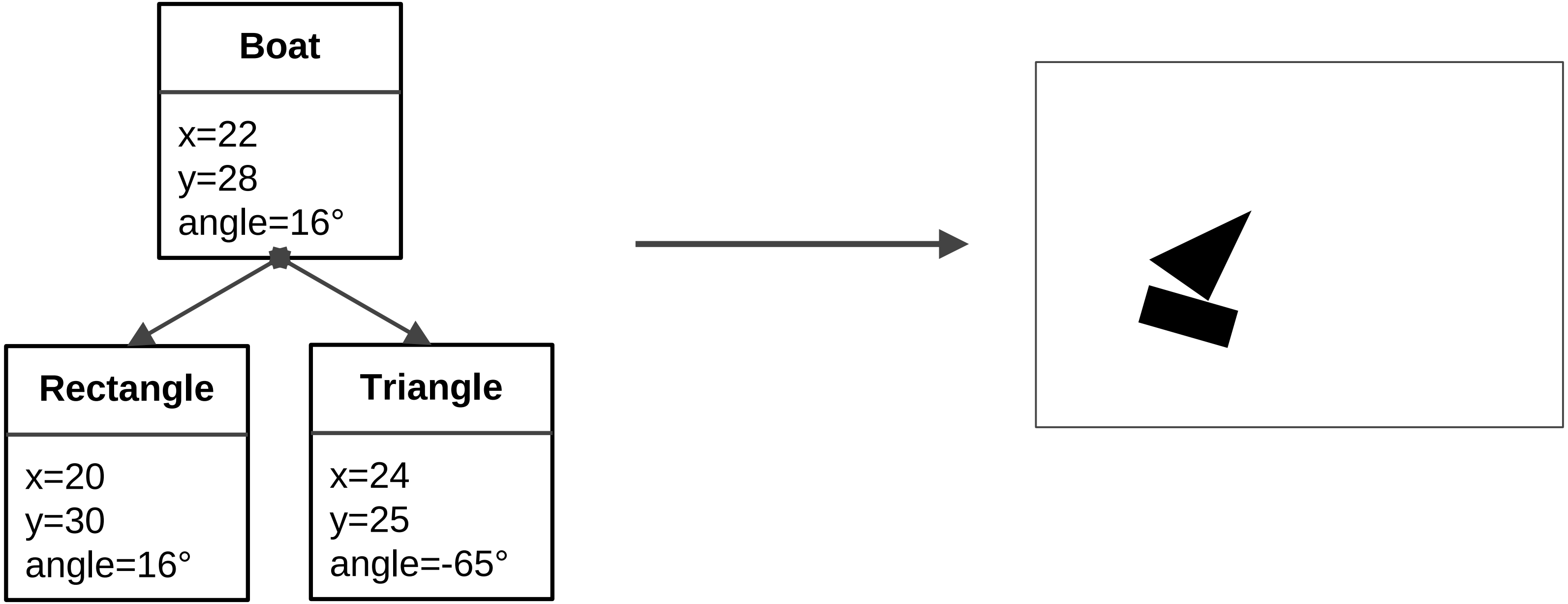
A hierarchy of parts

Boat

x=22
y=28
angle=16°

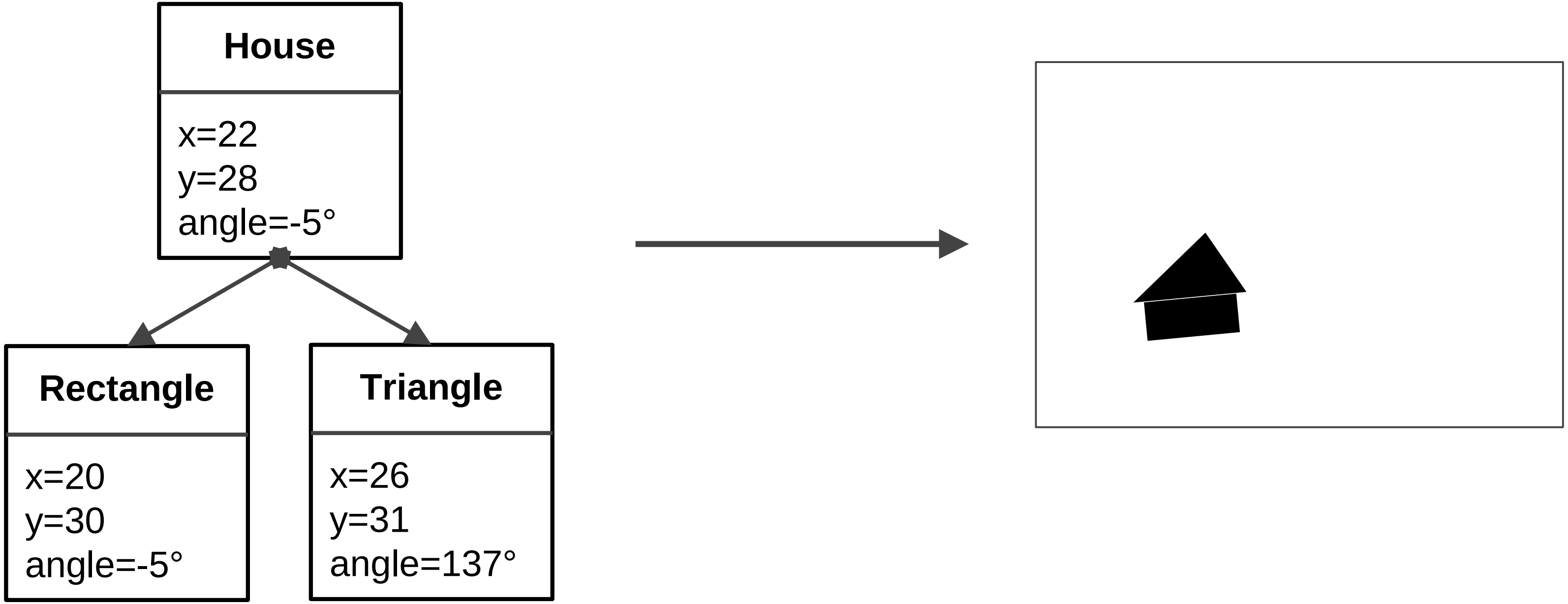
Aurélien Géron, 2017

A hierarchy of parts

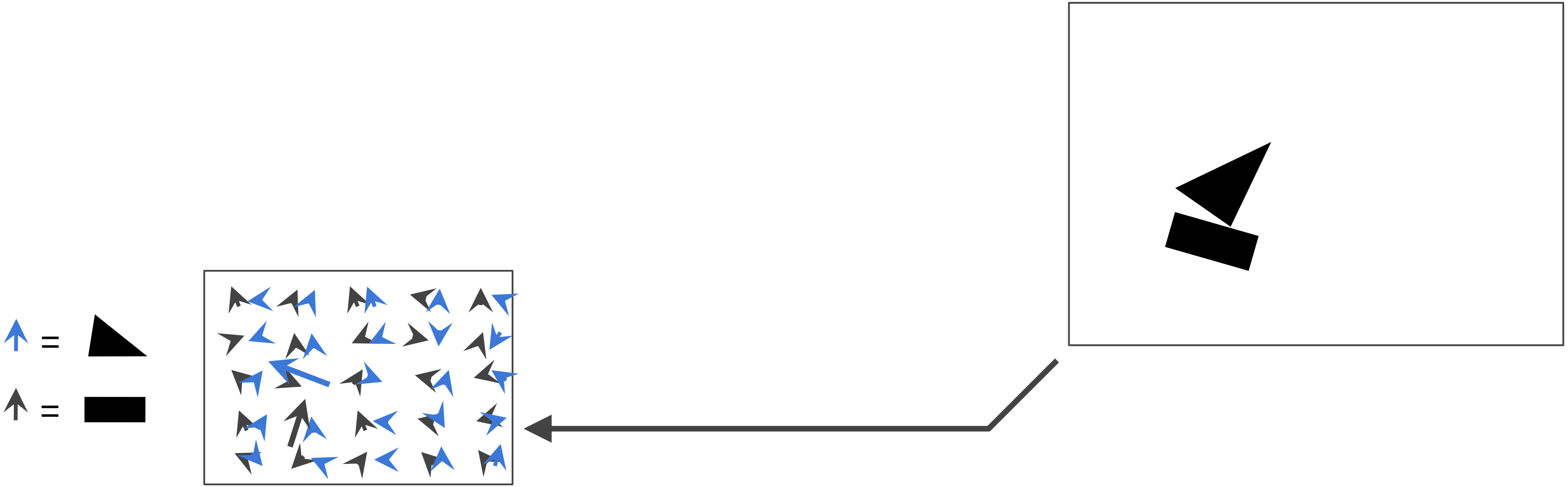


Aurélien Géron, 2017

A hierarchy of parts



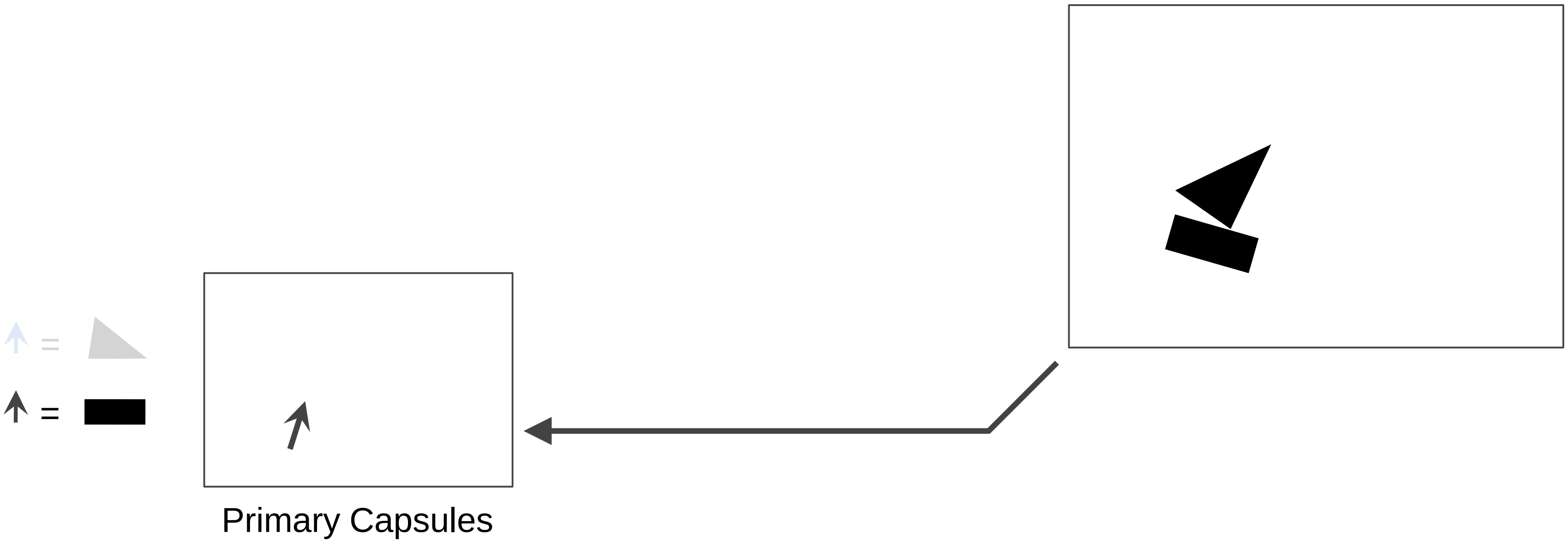
Primary Capsules



Primary Capsules

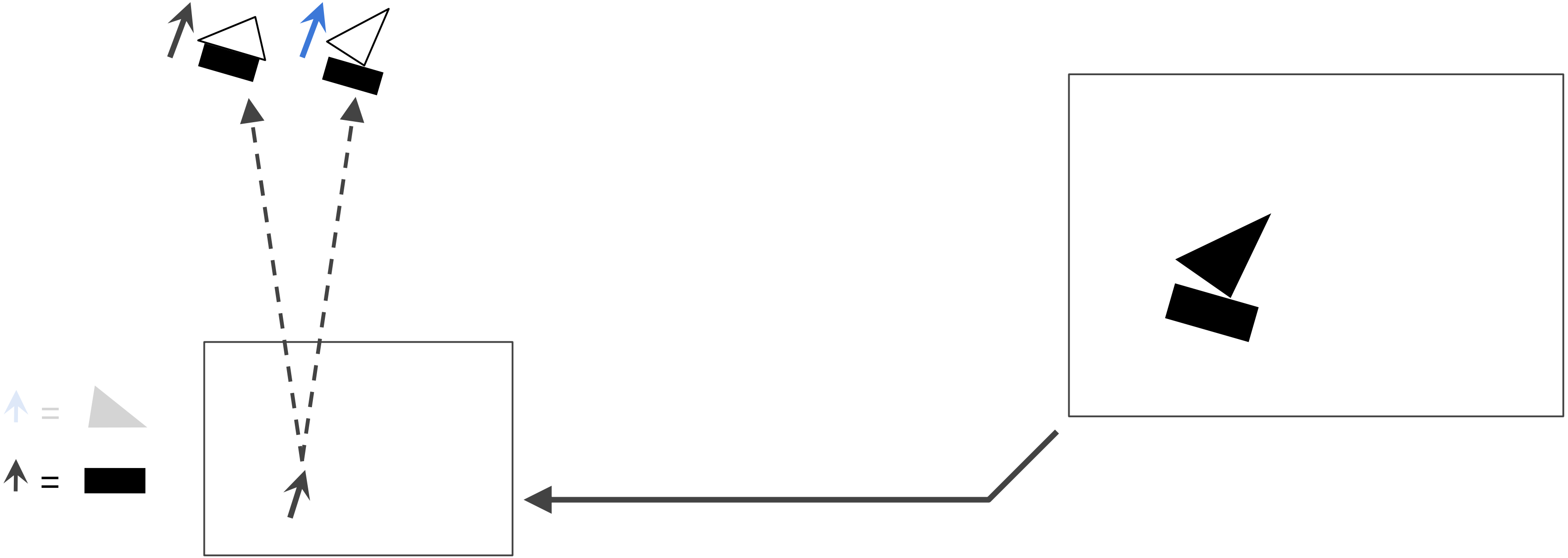
Aurélien Géron, 2017

Predict Next Layer's Output



Aurélien Géron, 2017

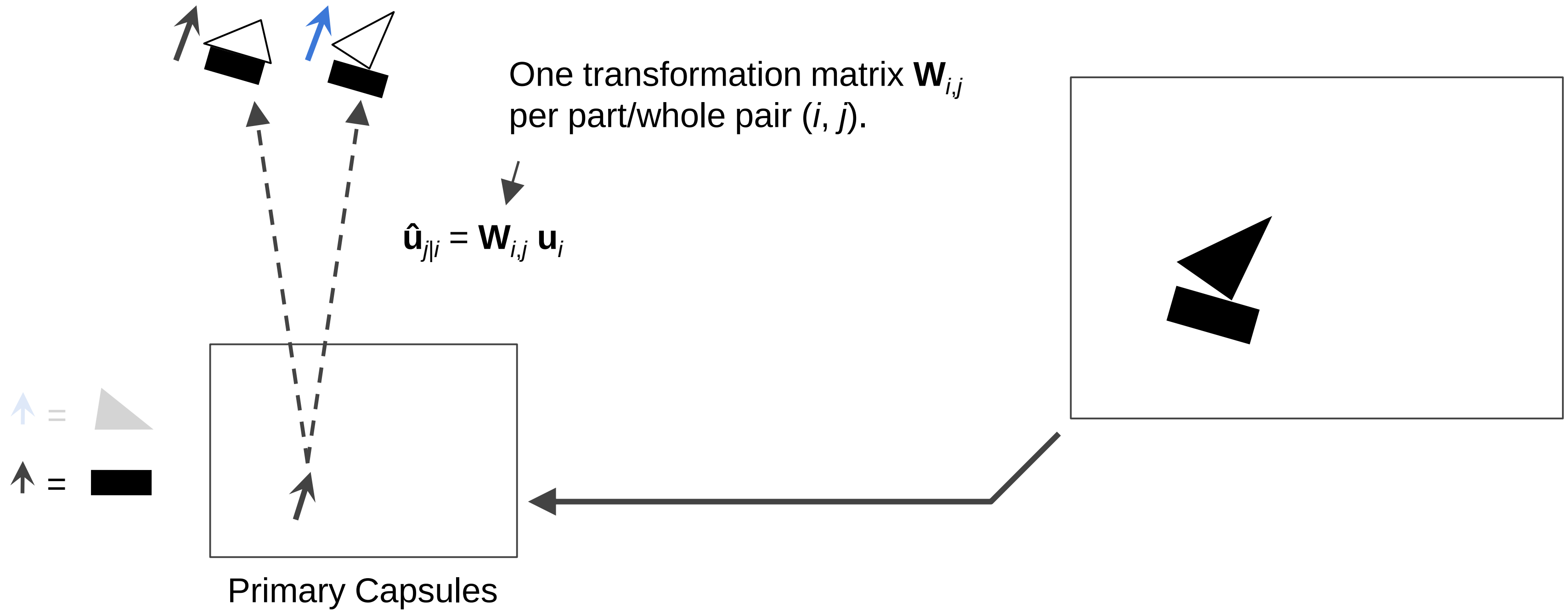
Predict Next Layer's Output



Primary Capsules

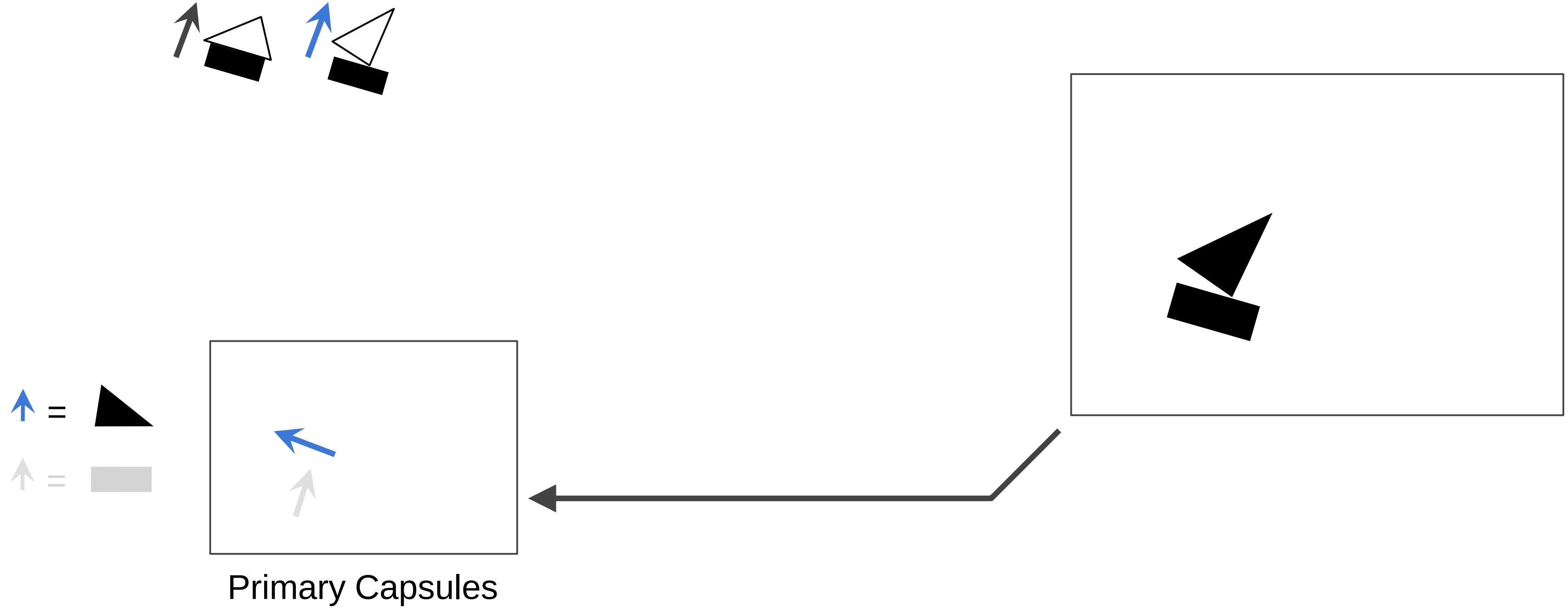
Aurélien Géron, 2017

Predict Next Layer's Output



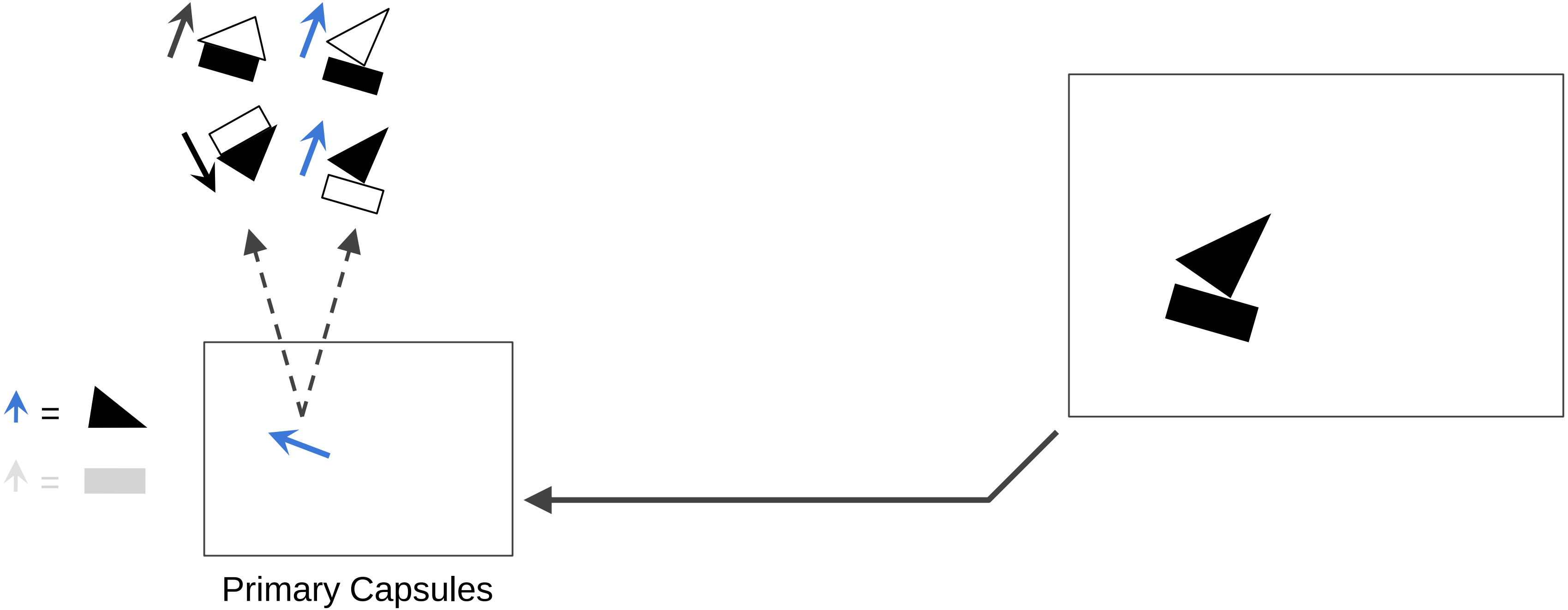
Aurélien Géron, 2017

Predict Next Layer's Output



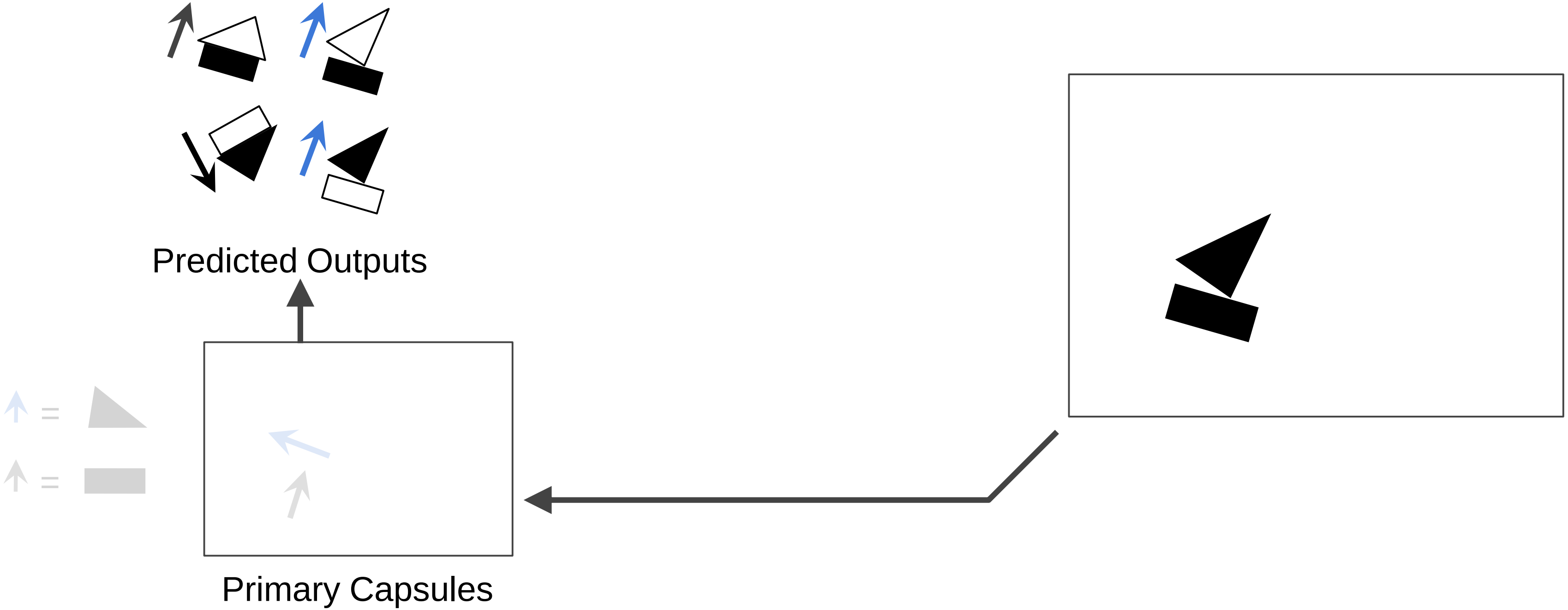
Aurélien Géron, 2017

Predict Next Layer’s Output



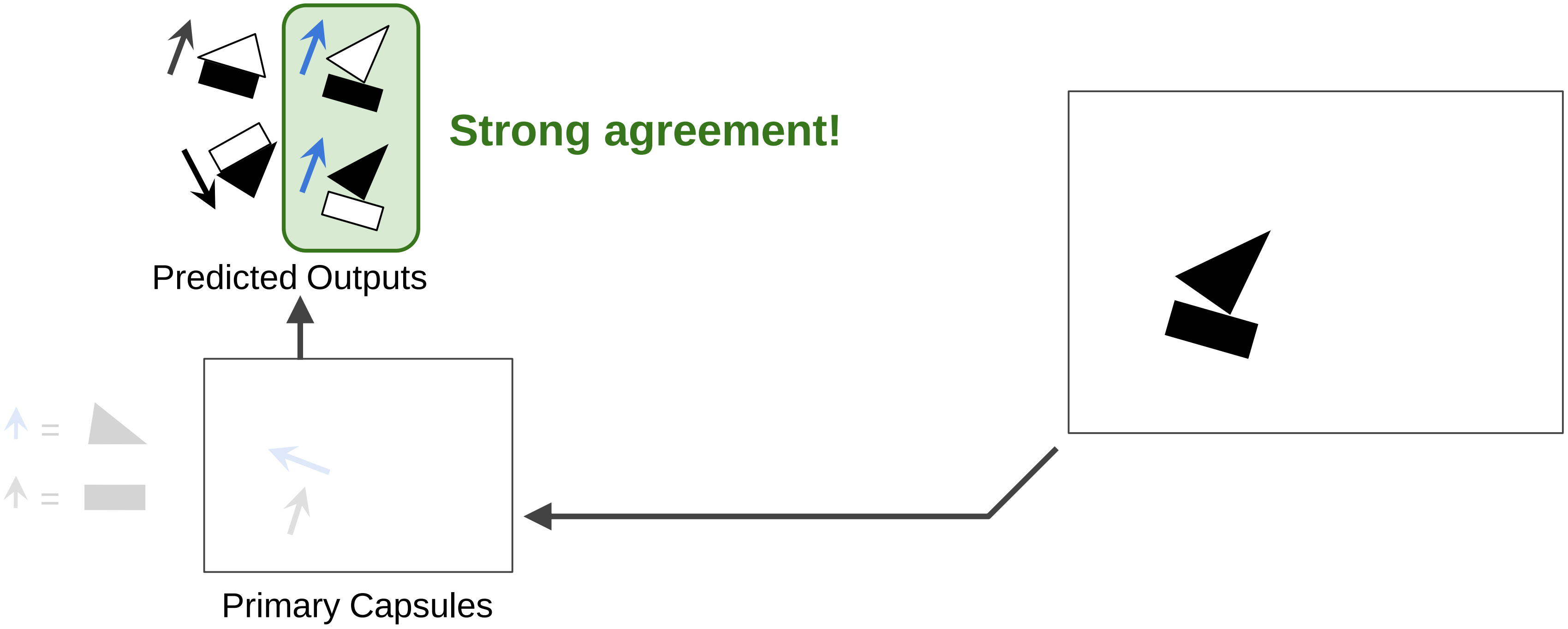
Aurélien Géron, 2017

Compute Next Layer's Output



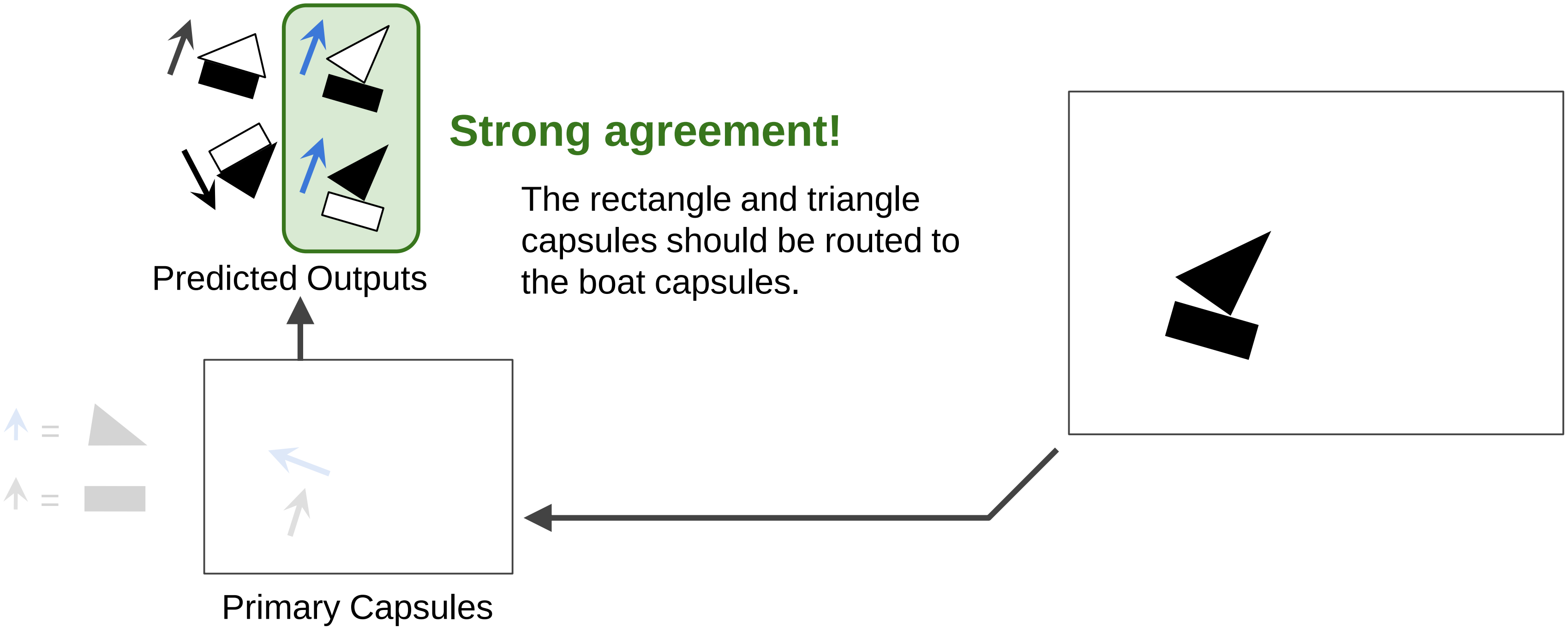
Aurélien Géron, 2017

Routing by Agreement



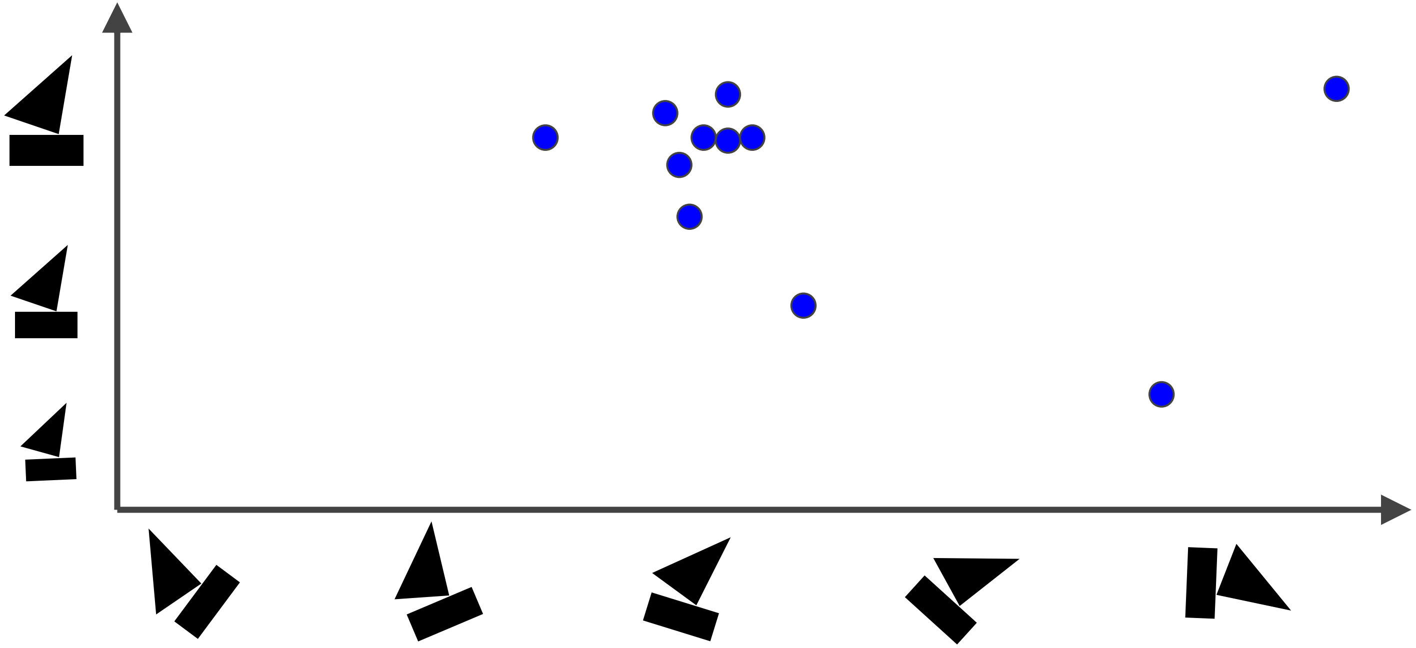
Aurélien Géron, 2017

Routing by Agreement



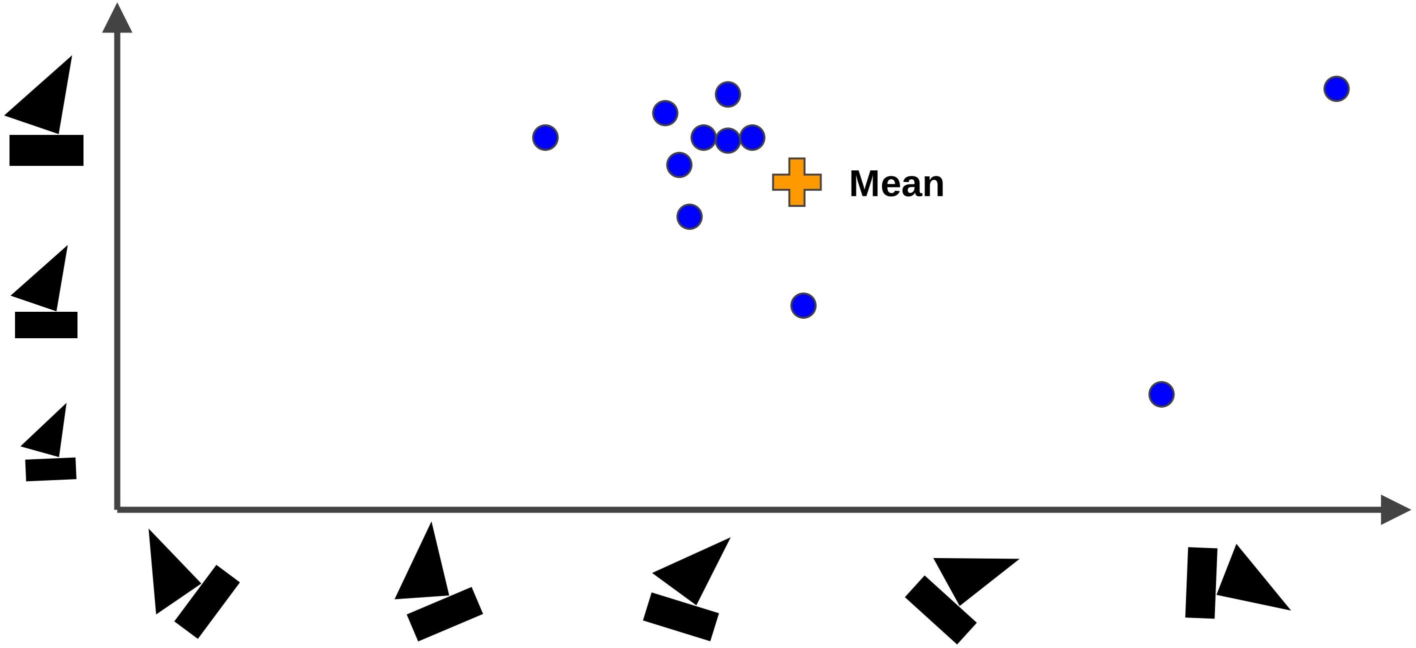
Aurélien Géron, 2017

Clusters of Agreement



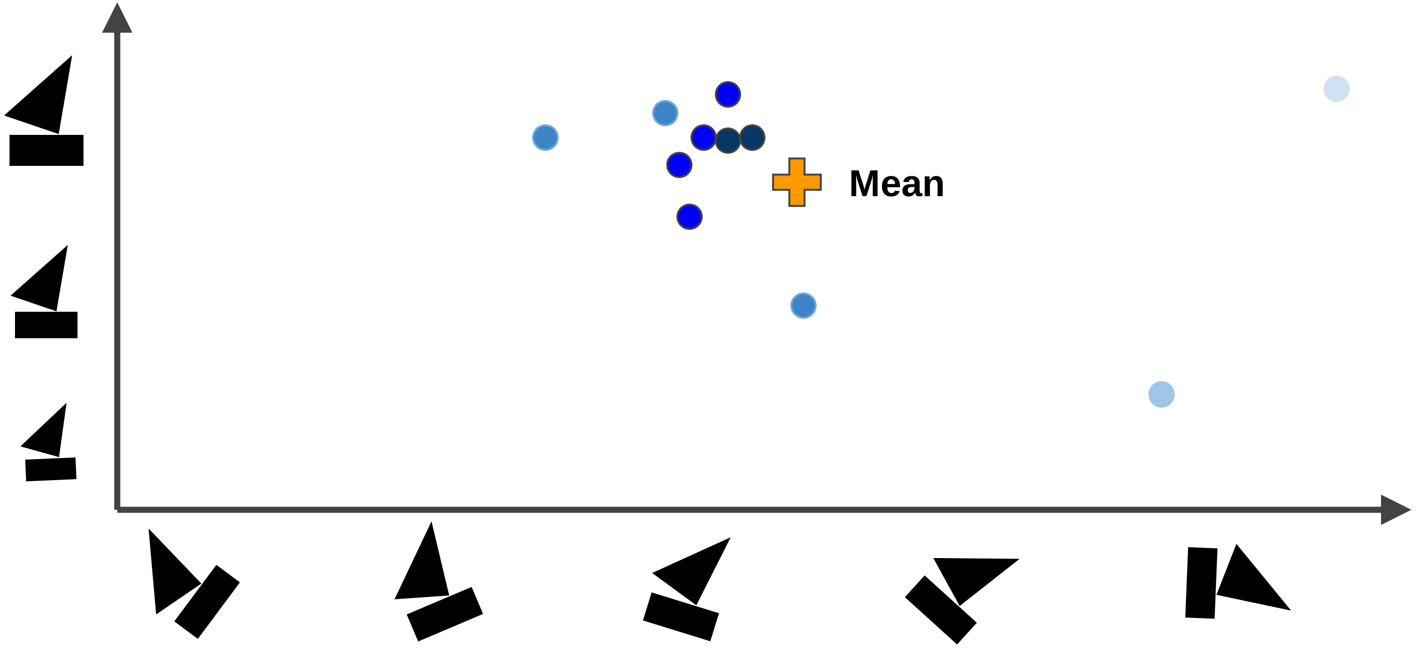
Aurélien Géron, 2017

Clusters of Agreement



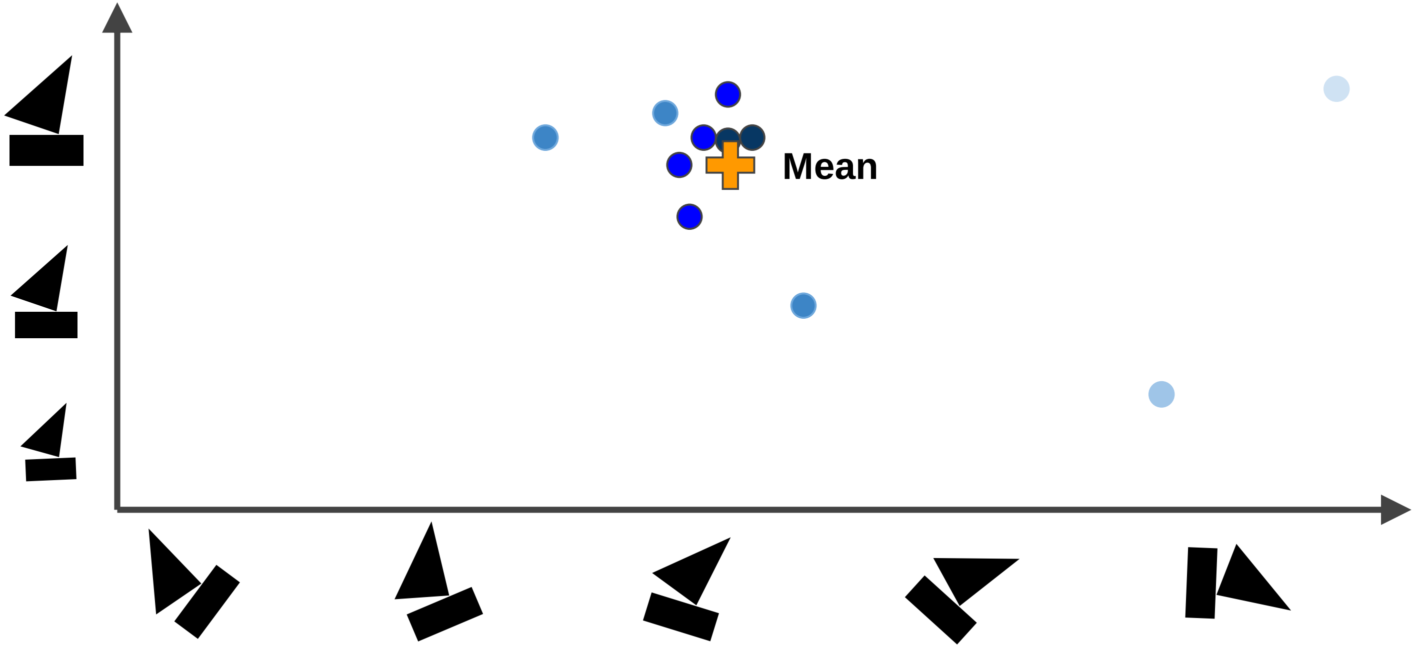
Aurélien Géron, 2017

Clusters of Agreement



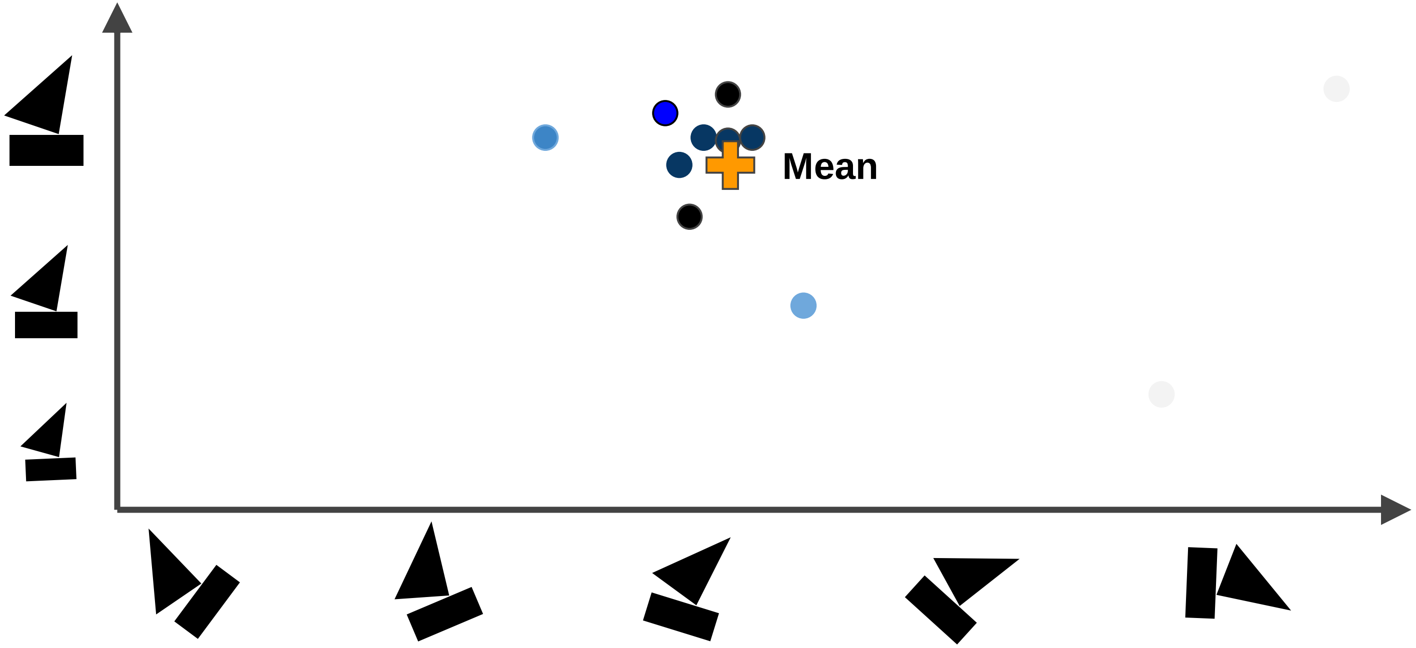
Aurélien Géron, 2017

Clusters of Agreement



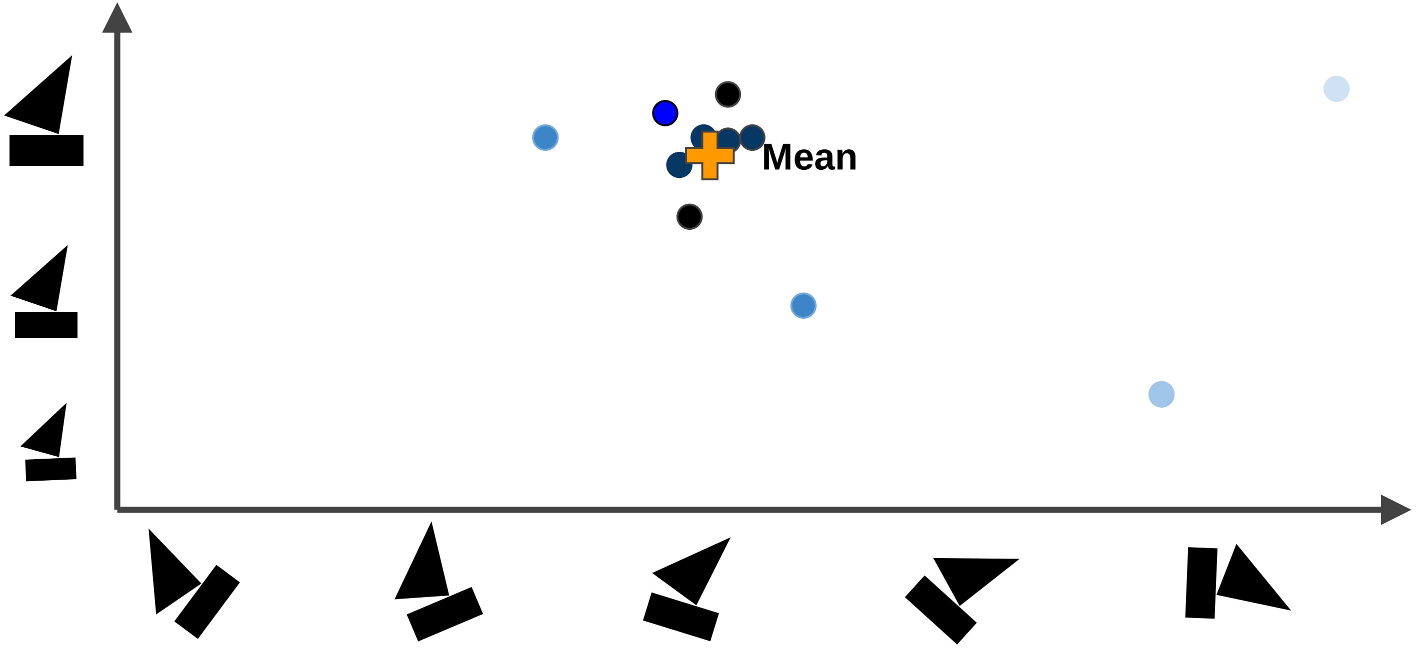
Aurélien Géron, 2017

Clusters of Agreement



Aurélien Géron, 2017

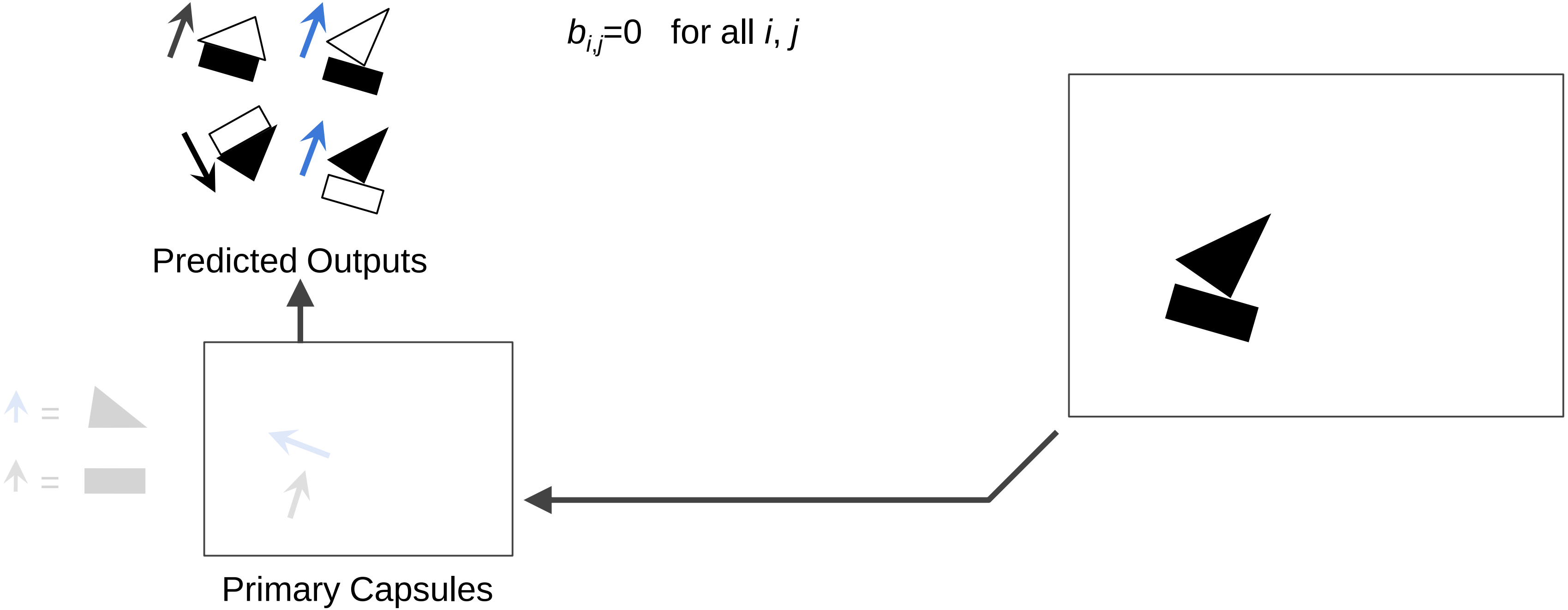
Clusters of Agreement



Aurélien Géron, 2017

Routing Weights

$b_{i,j}=0$ for all i, j

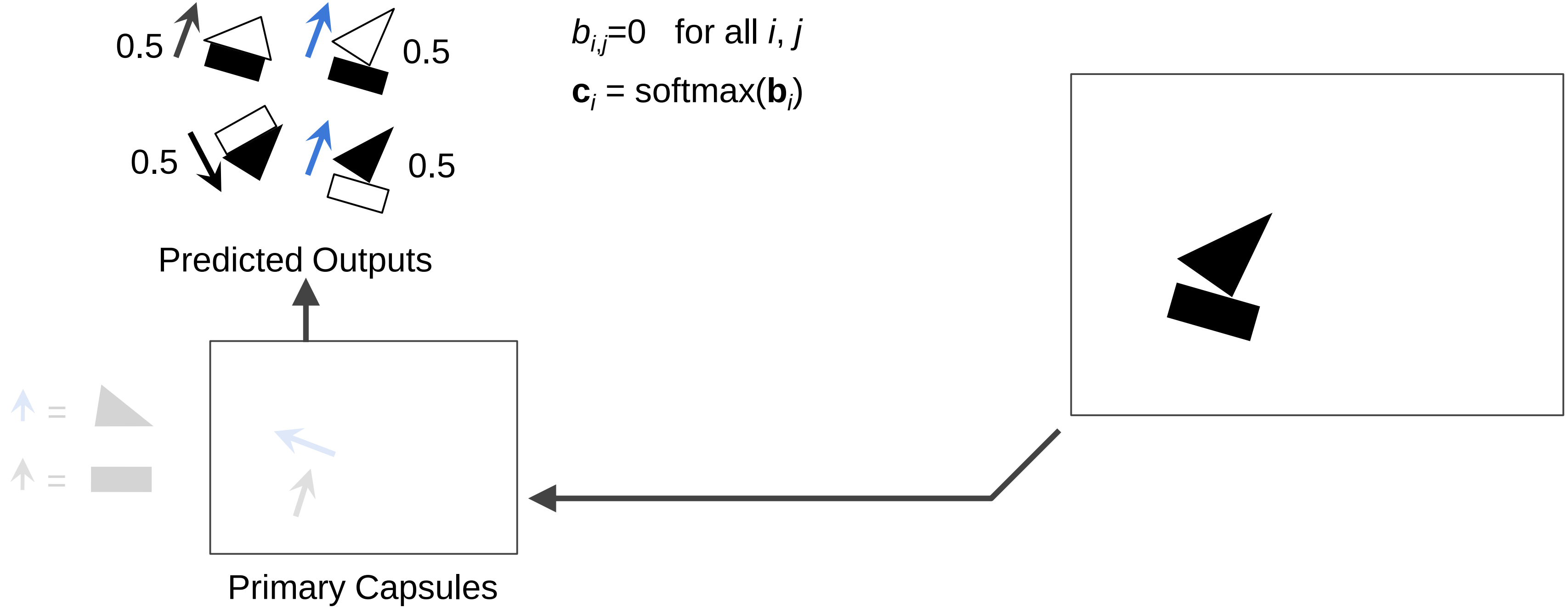


Aurélien Géron, 2017

Routing Weights

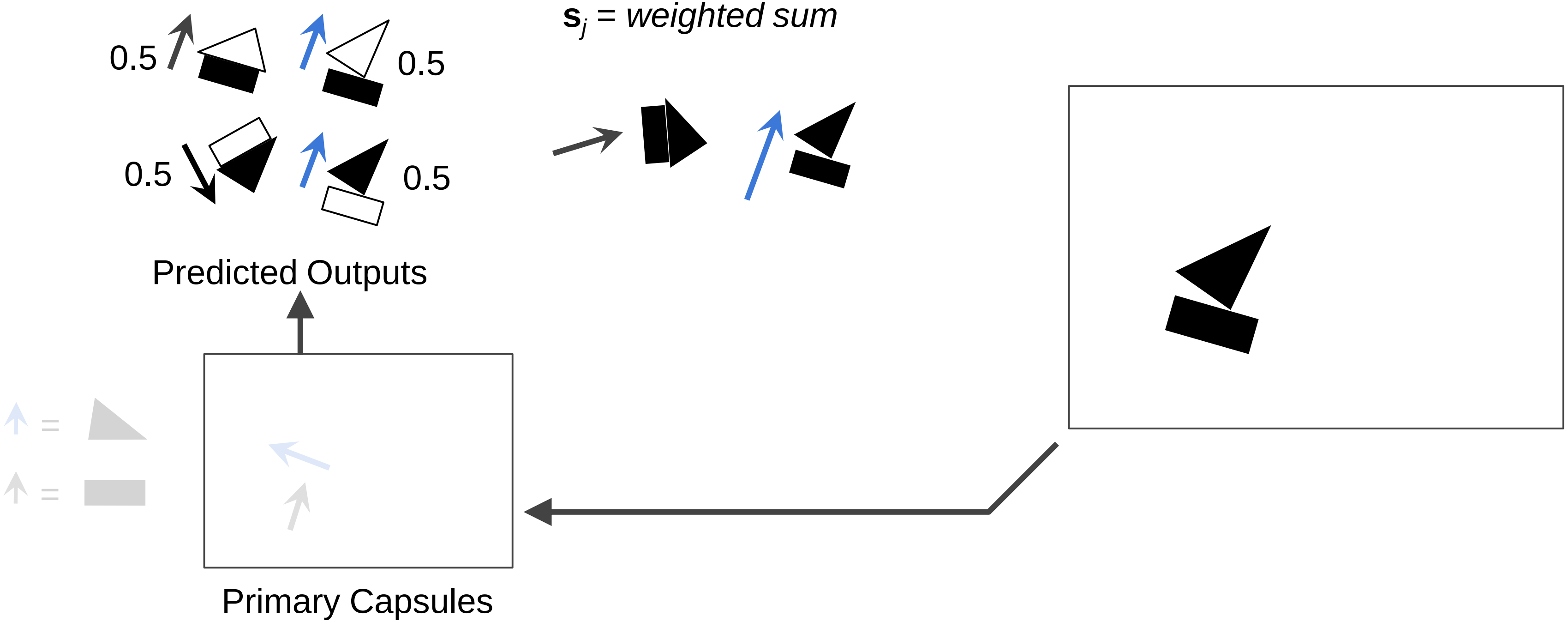
$b_{i,j}=0$ for all i, j

$\mathbf{c}_i = \text{softmax}(\mathbf{b}_i)$



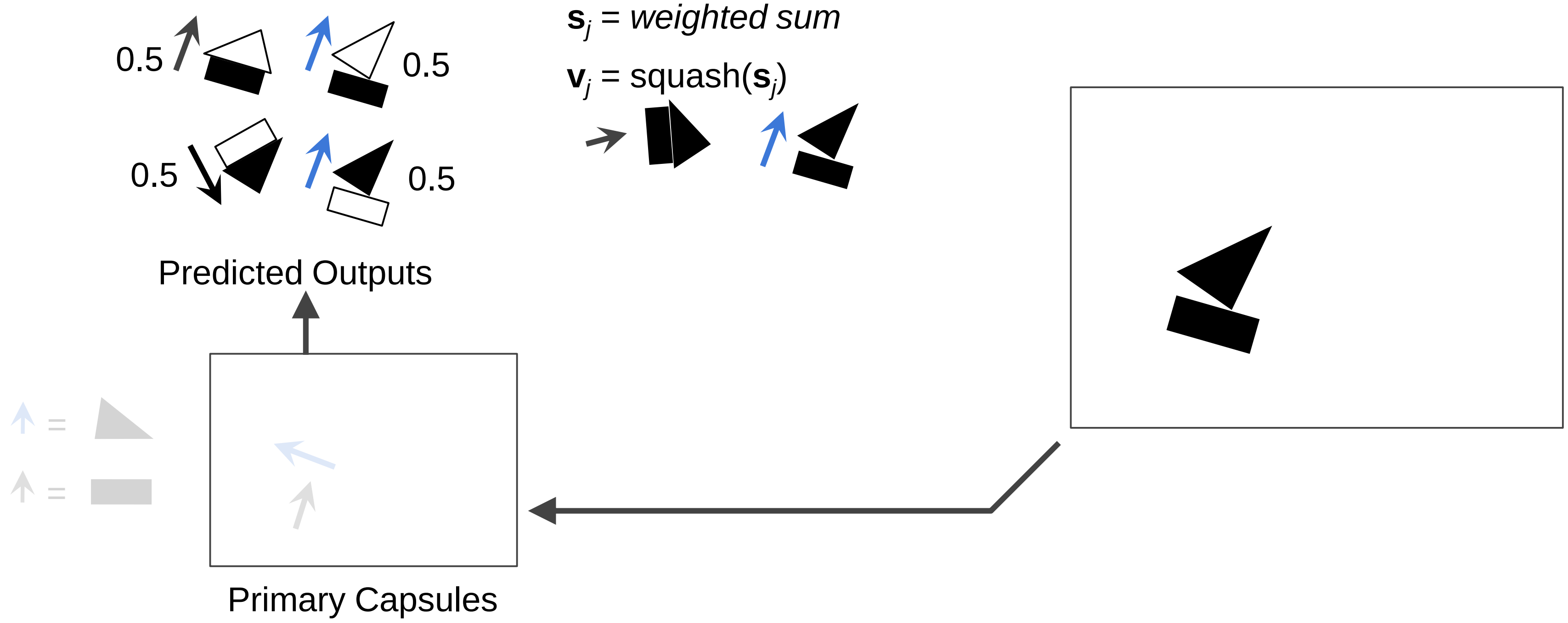
Aurélien Géron, 2017

Compute Next Layer's Output



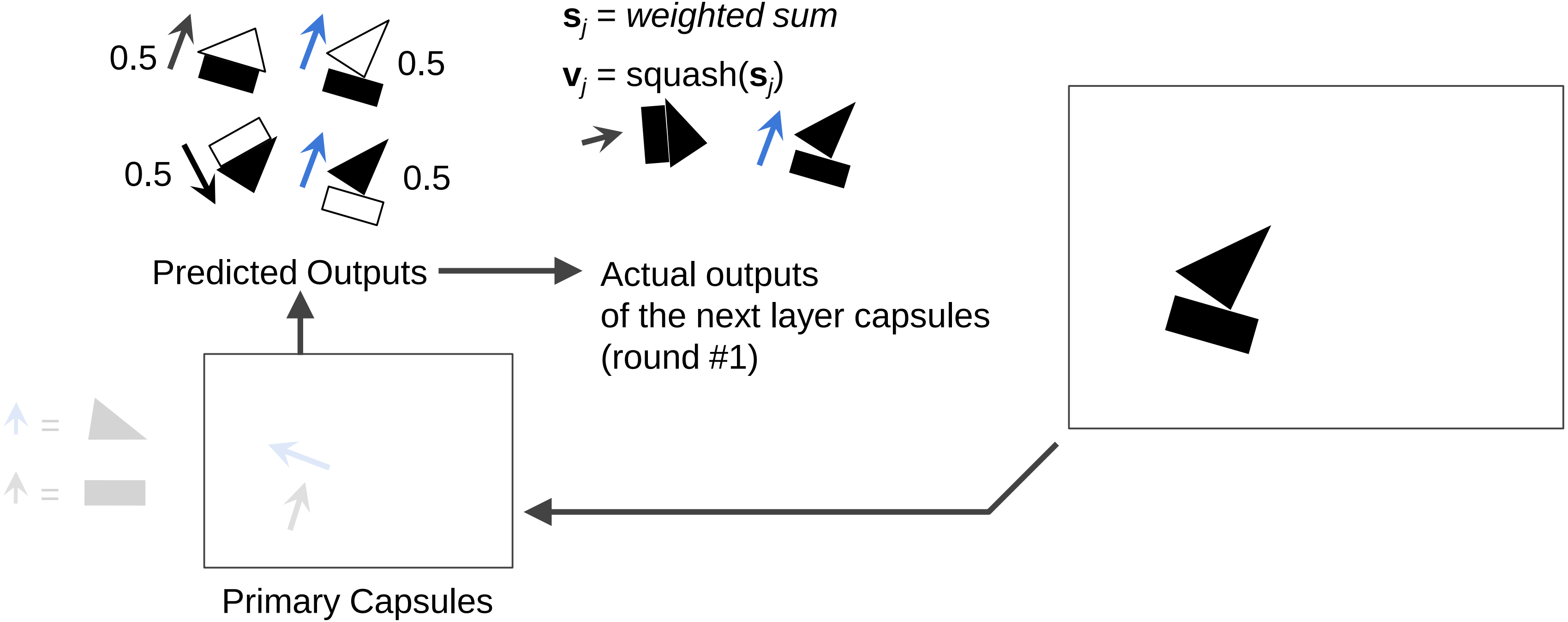
Aurélien Géron, 2017

Compute Next Layer's Output



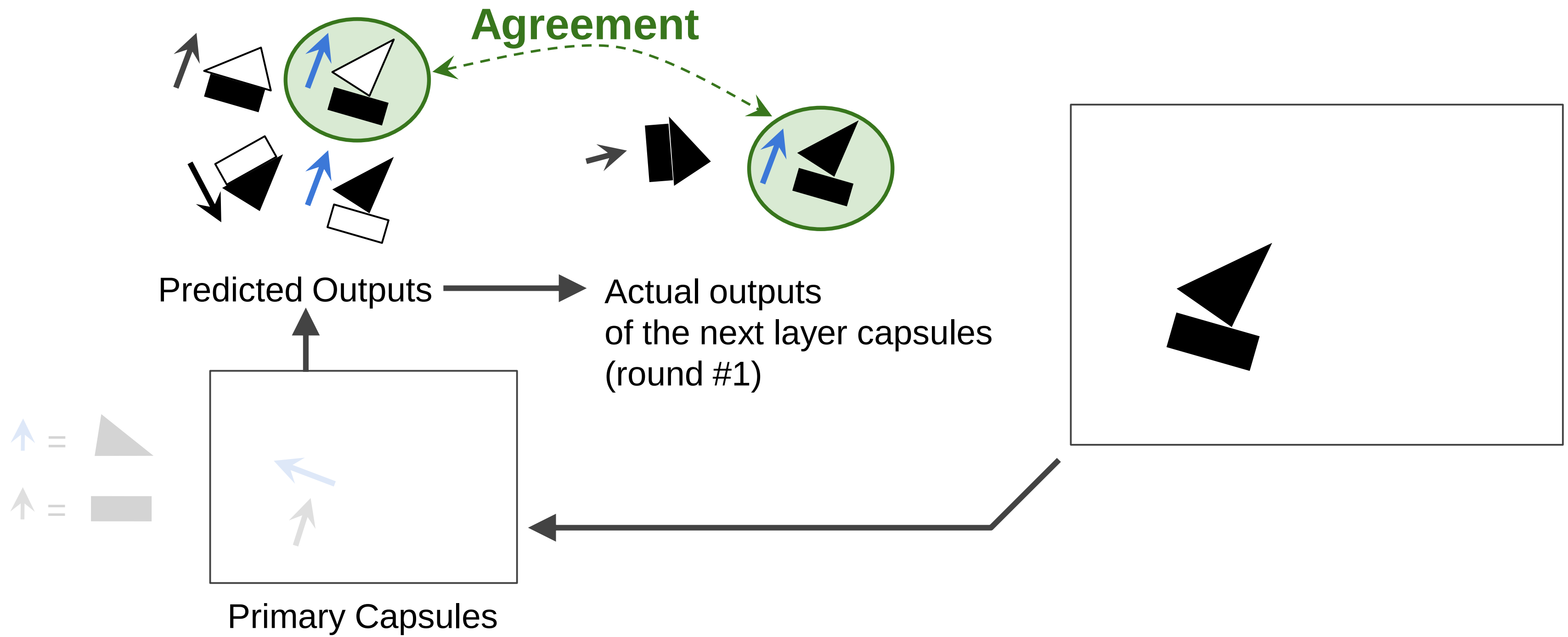
Aurélien Géron, 2017

Compute Next Layer's Output



Aurélien Géron, 2017

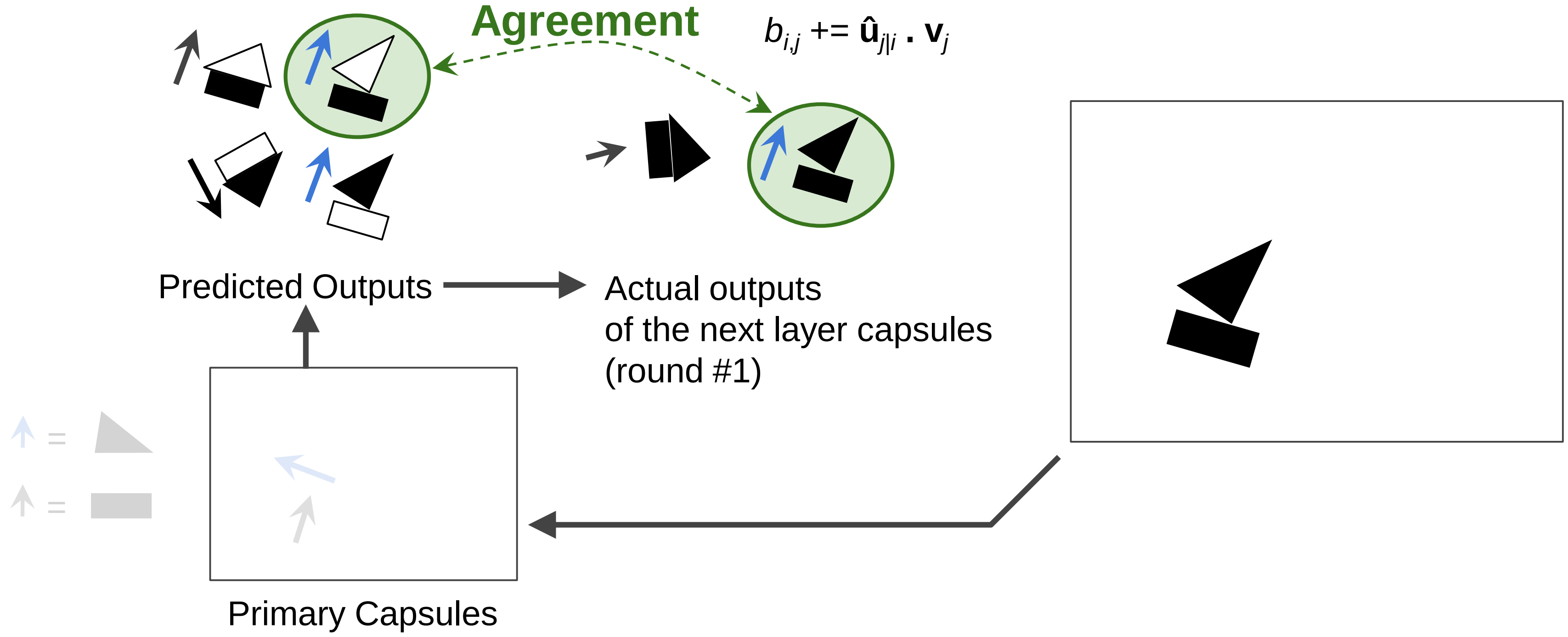
Update Routing Weights



Aurélien Géron, 2017

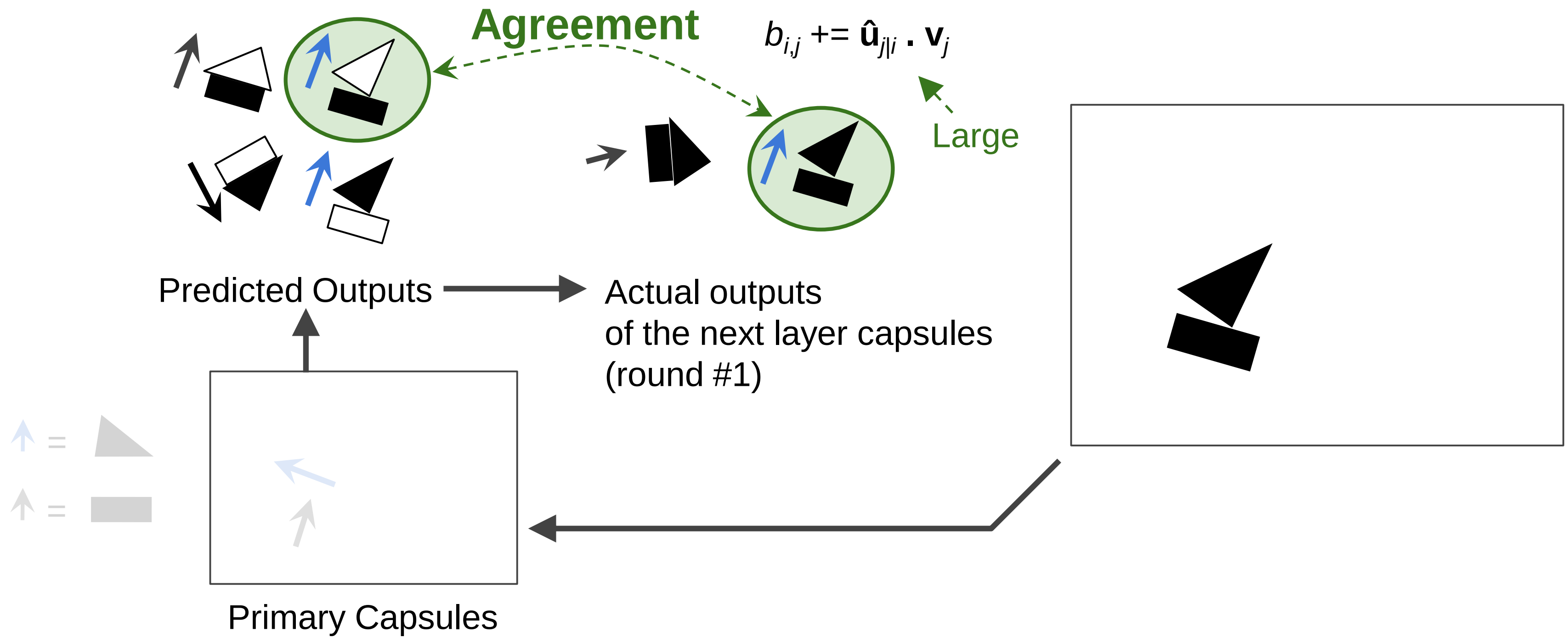
Update Routing Weights

$b_{i,j} \mathrel{+}= \hat{\mathbf{u}}_{j|i} \cdot \mathbf{v}_j$

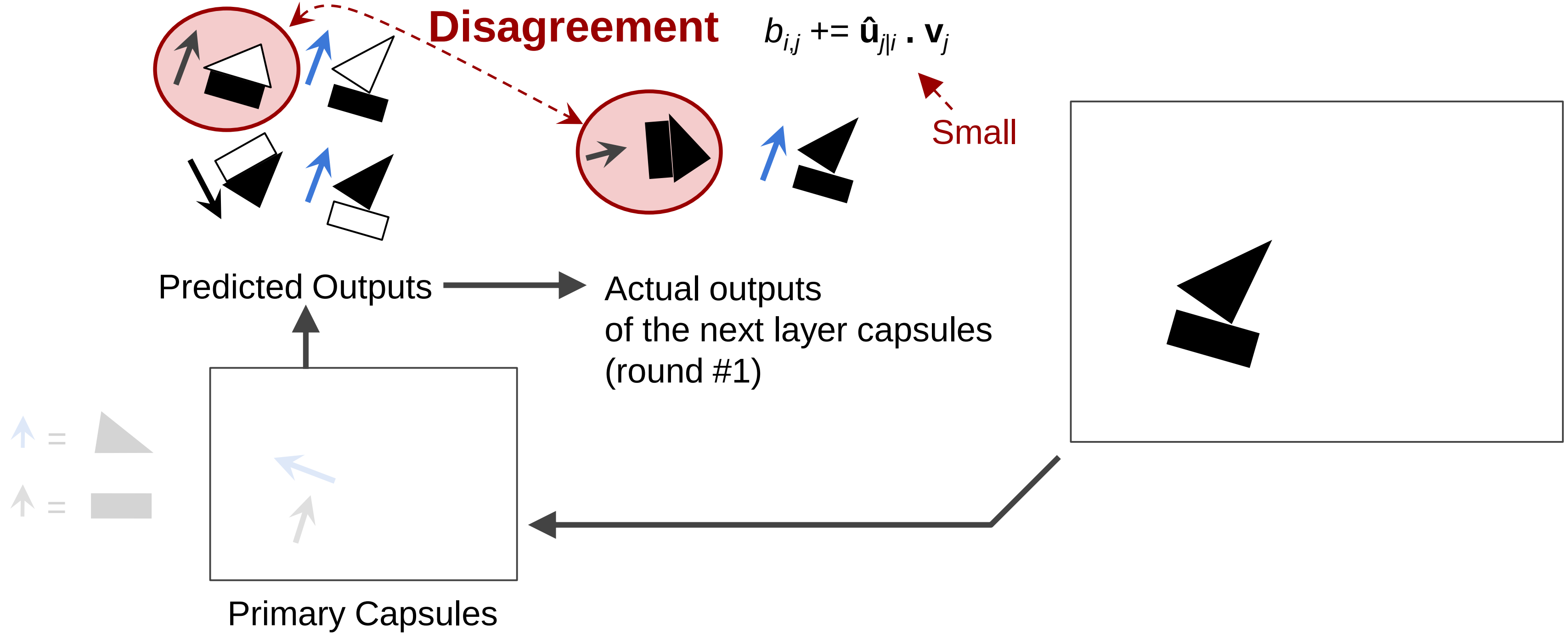


Aurélien Géron, 2017

Update Routing Weights

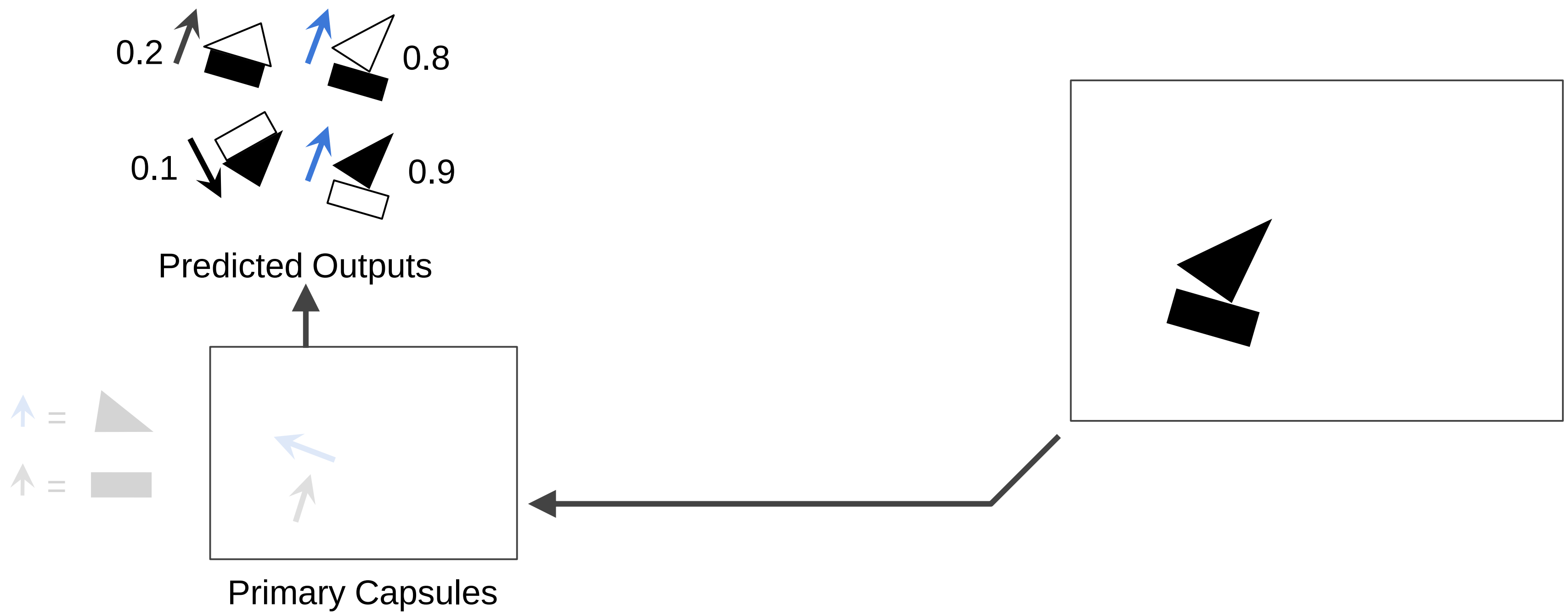


Update Routing Weights



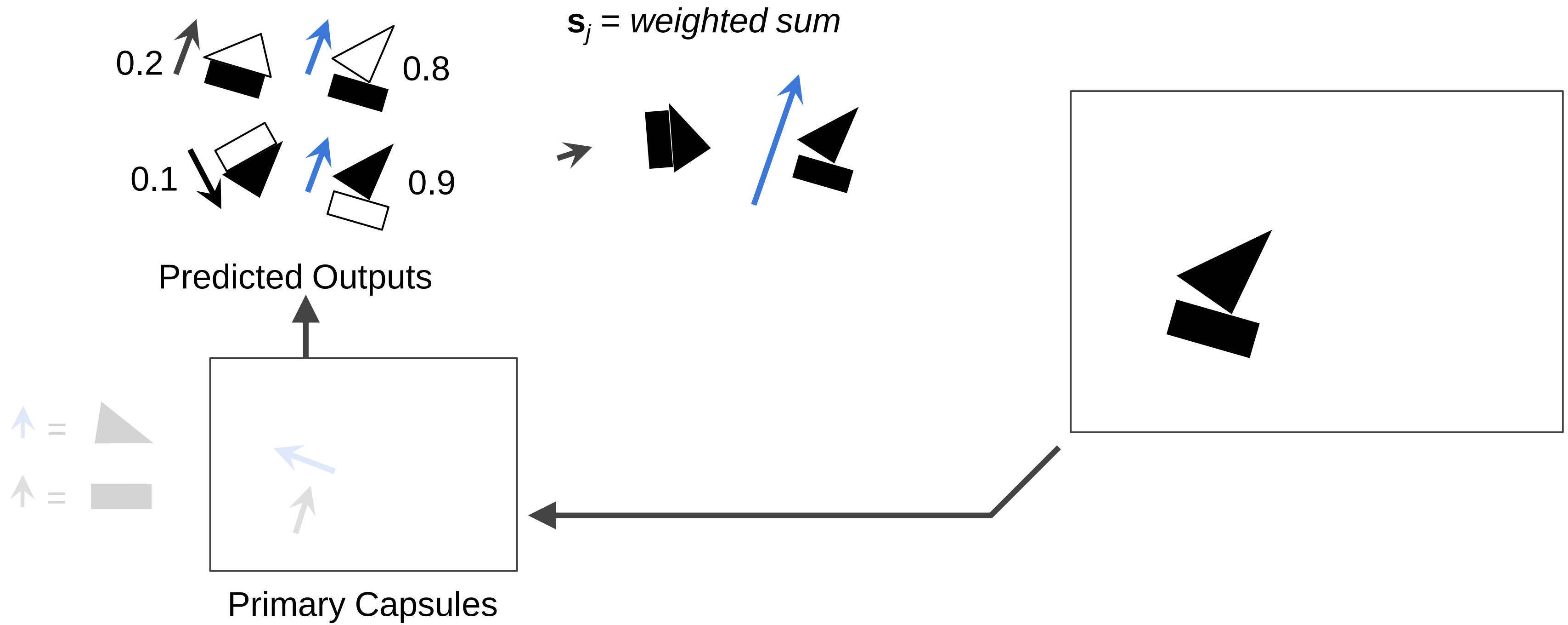
Aurélien Géron, 2017

Compute Next Layer's Output



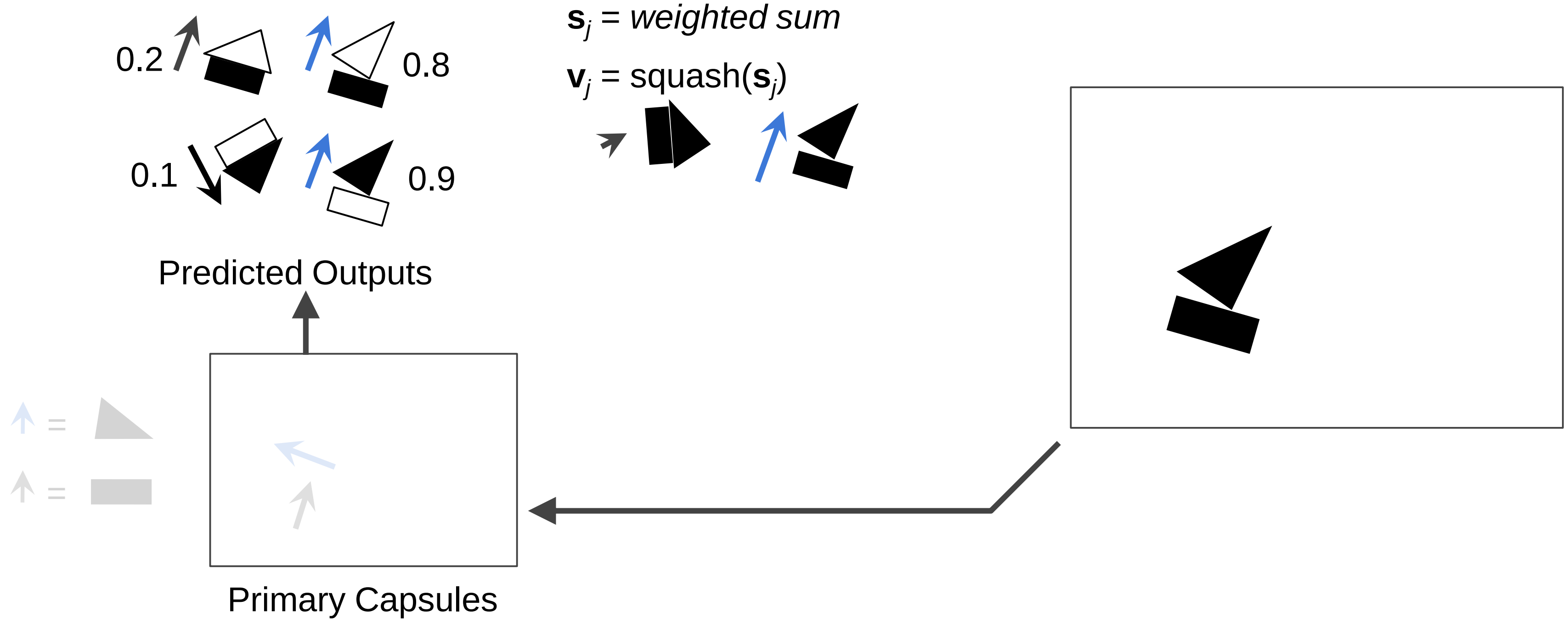
Aurélien Géron, 2017

Compute Next Layer's Output



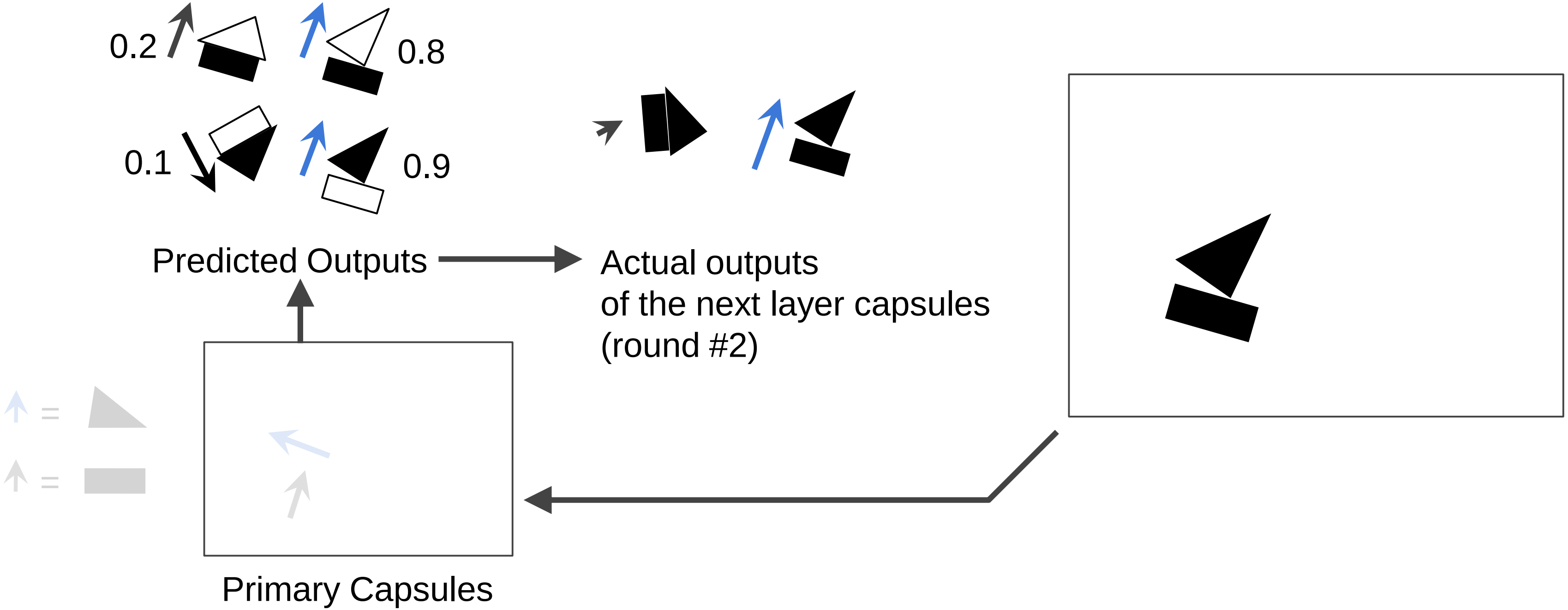
Aurélien Géron, 2017

Compute Next Layer's Output



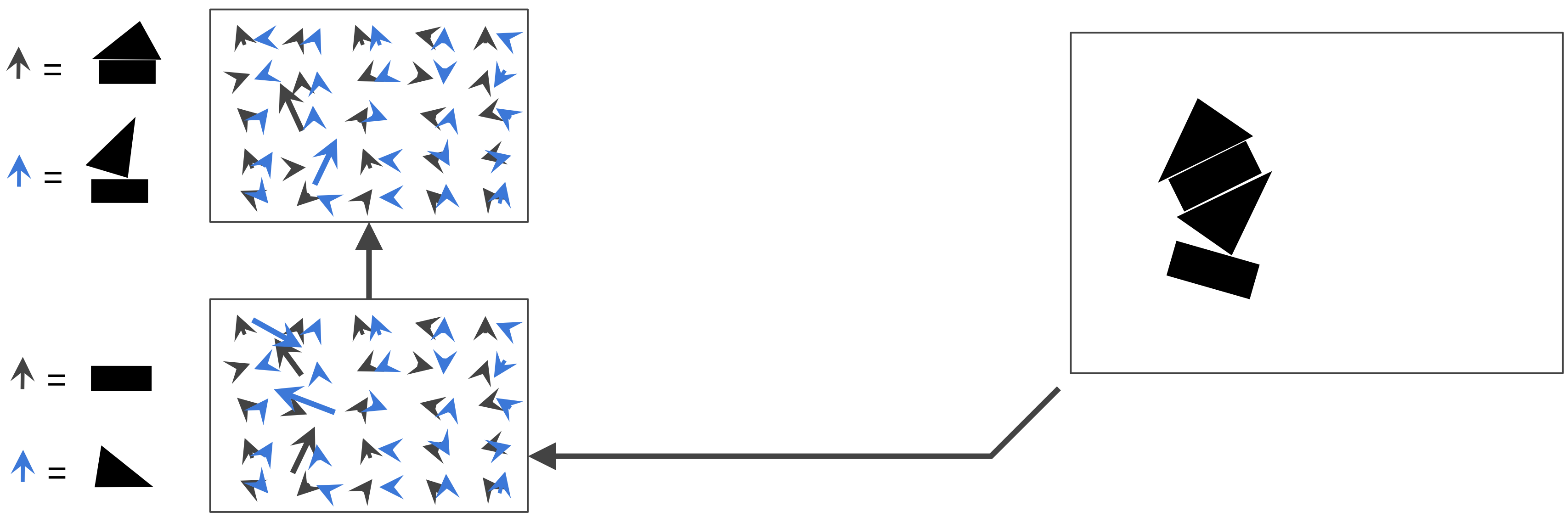
Aurélien Géron, 2017

Compute Next Layer's Output



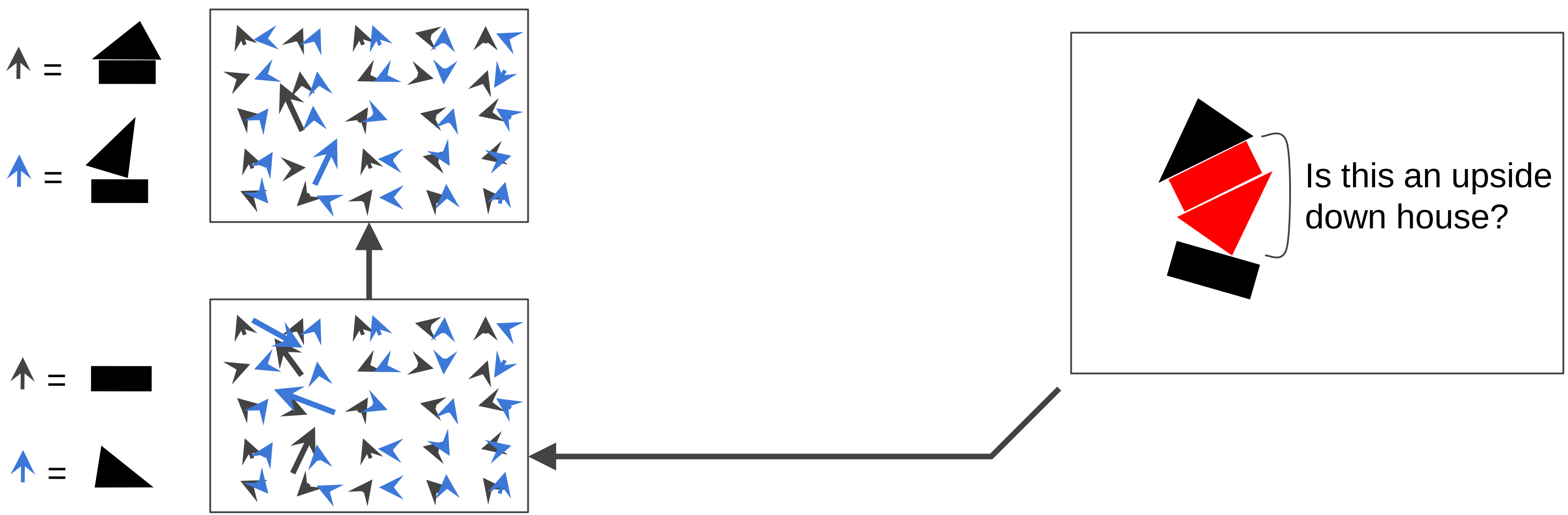
Aurélien Géron, 2017

Handling Crowded Scenes



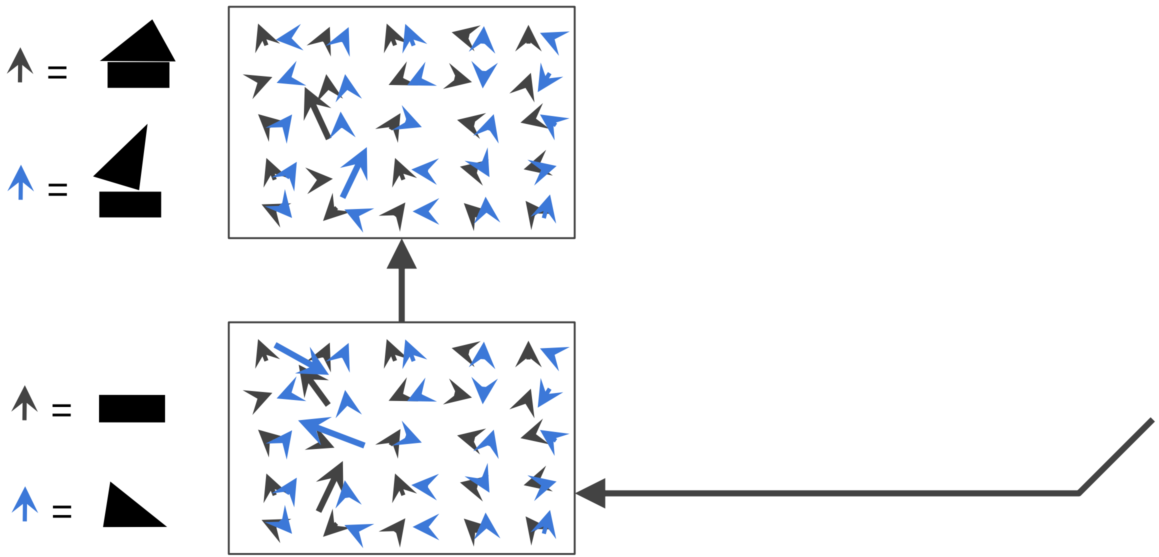
Aurélien Géron, 2017

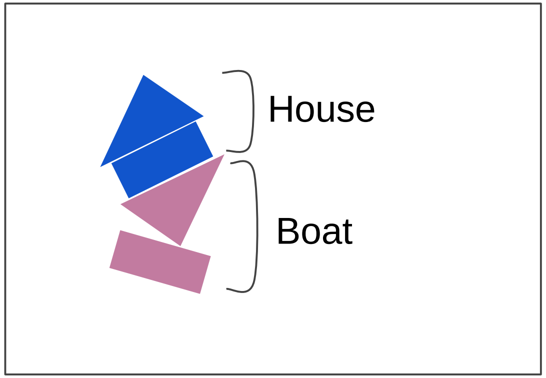
Handling Crowded Scenes



Aurélien Géron, 2017

Handling Crowded Scenes



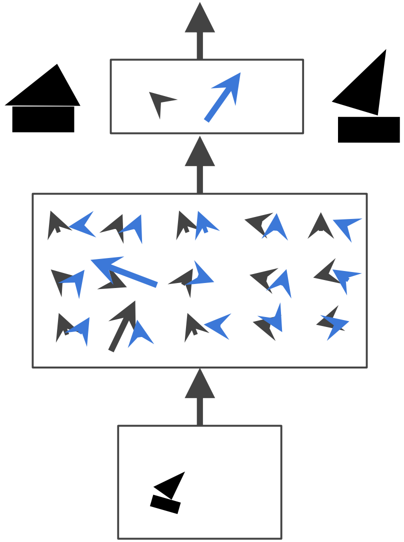


Thanks to routing by agreement, the ambiguity is quickly resolved (explaining away).

Aurélien Géron, 2017

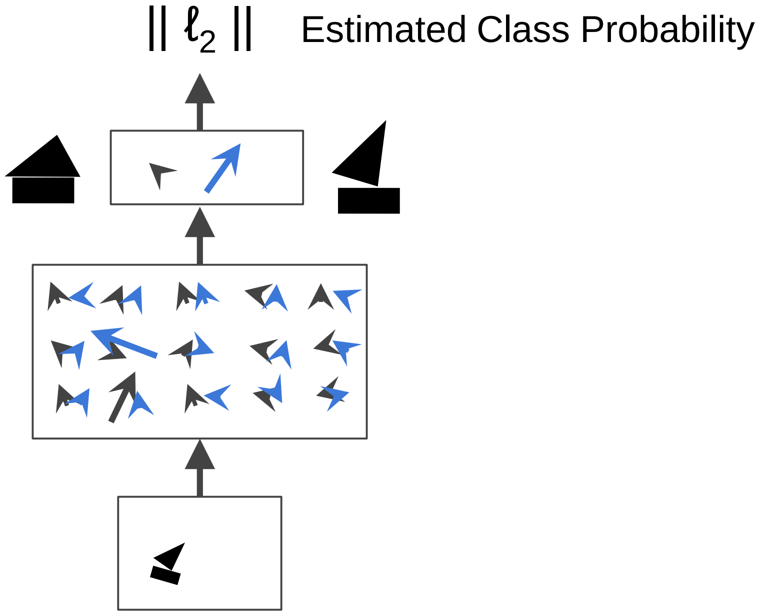
Classification
CapsNet

$|| \ell_2 ||$ Estimated Class Probability



Aurélien Géron, 2017

Training



To allow multiple classes, minimize margin loss:

$L_k = T_k \max(0, m^+ - ||\mathbf{v}_k||^2)$

$+ \lambda (1 - T_k) \max(0, ||\mathbf{v}_k||^2 - m^-)$

$T_k = 1$ iff class k is present

In the paper:

$m^- = 0.1$

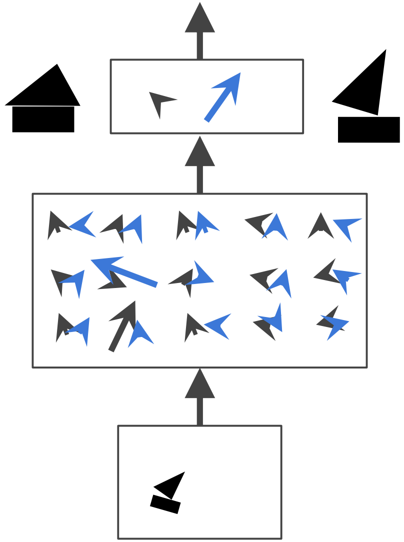
$m^+ = 0.9$

$\lambda = 0.5$

Aurélien Géron, 2017

Training

$\| \ell_2 \|$ Estimated Class Probability



Translated to English:

"If an object of class k is present, then $\|\mathbf{v}_k\|^2$ should be no less than 0.9. If not, then $\|\mathbf{v}_k\|^2$ should be no more than 0.1."

To allow multiple classes, minimize margin loss:

$$L_k = T_k \max(0, m^+ - \|\mathbf{v}_k\|^2)$$
$$+ \lambda\, (1 - T_k) \max(0, \|\mathbf{v}_k\|^2 - m^-)$$

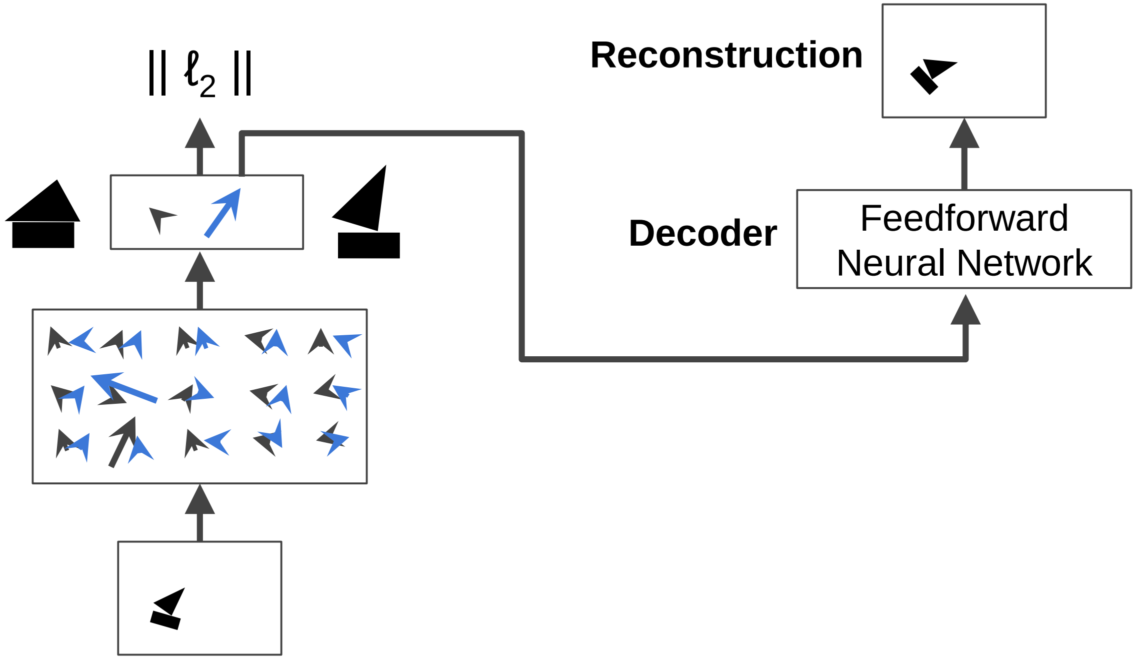
$T_k = 1$ iff class k is present

In the paper:

$m^- = 0.1$
$m^+ = 0.9$
$\lambda = 0.5$

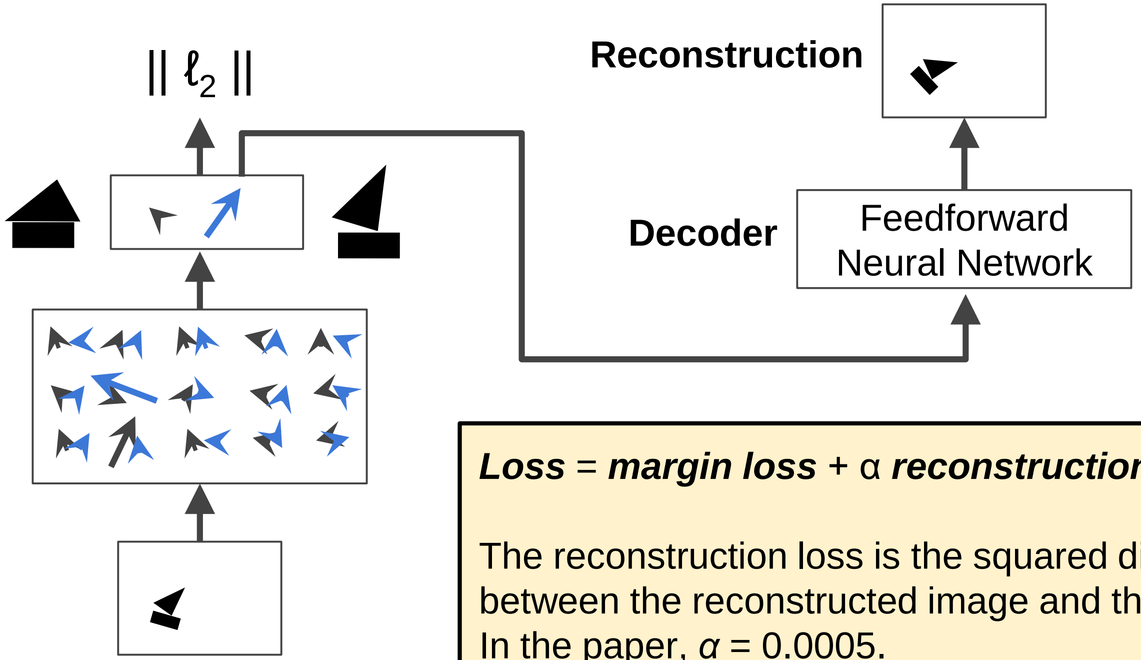
Aurélien Géron, 2017

Regularization by Reconstruction



Aurélien Géron, 2017

Regularization by Reconstruction

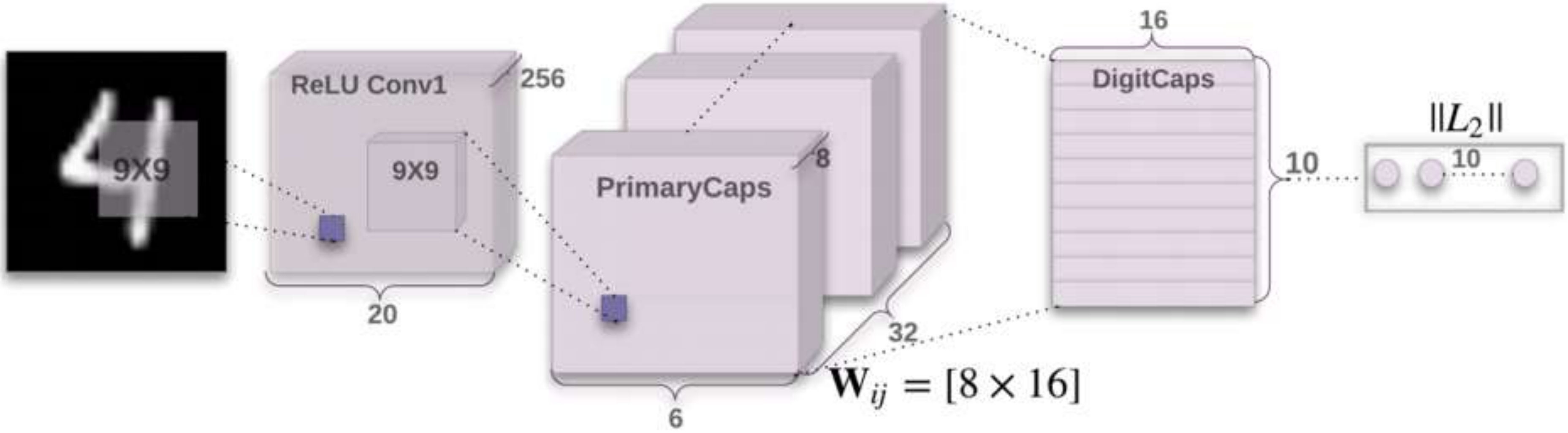


Loss = ***margin loss*** + α ***reconstruction loss***

The reconstruction loss is the squared difference between the reconstructed image and the input image. In the paper, α = 0.0005.

Aurélien Géron, 2017

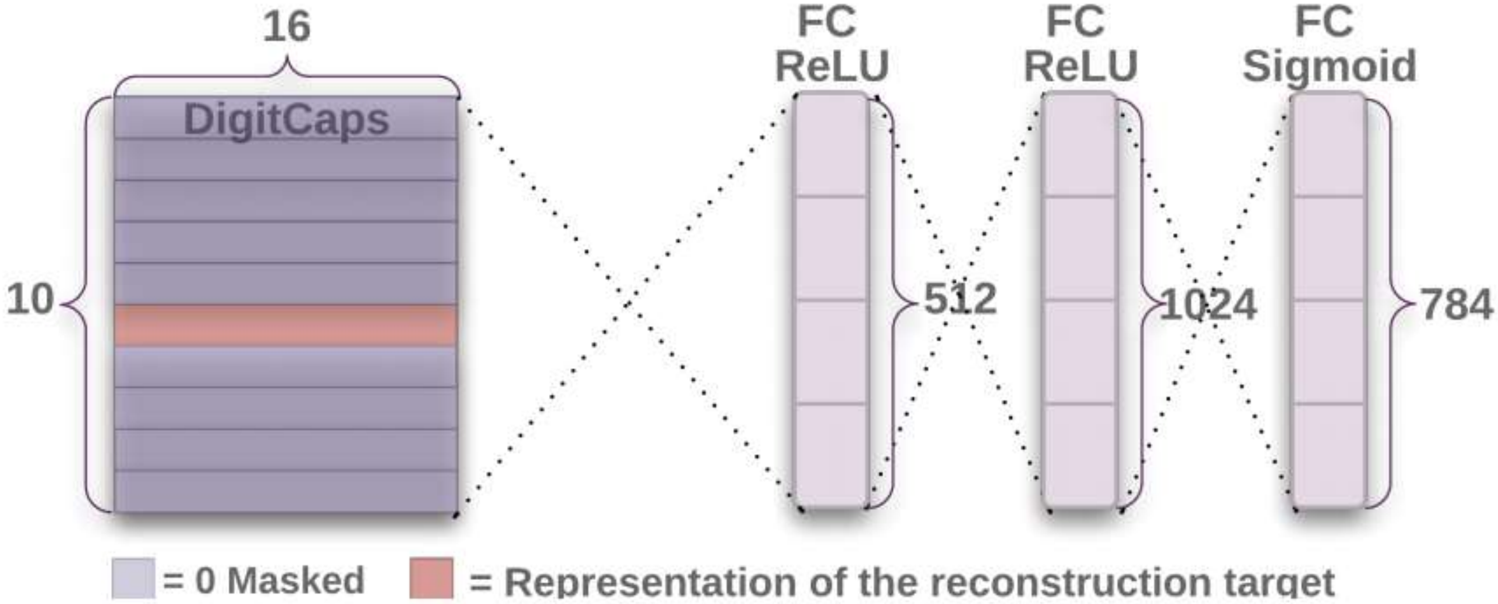
A CapsNet for MNIST



(Figure 1 from the paper)

Aurélien Géron, 2017

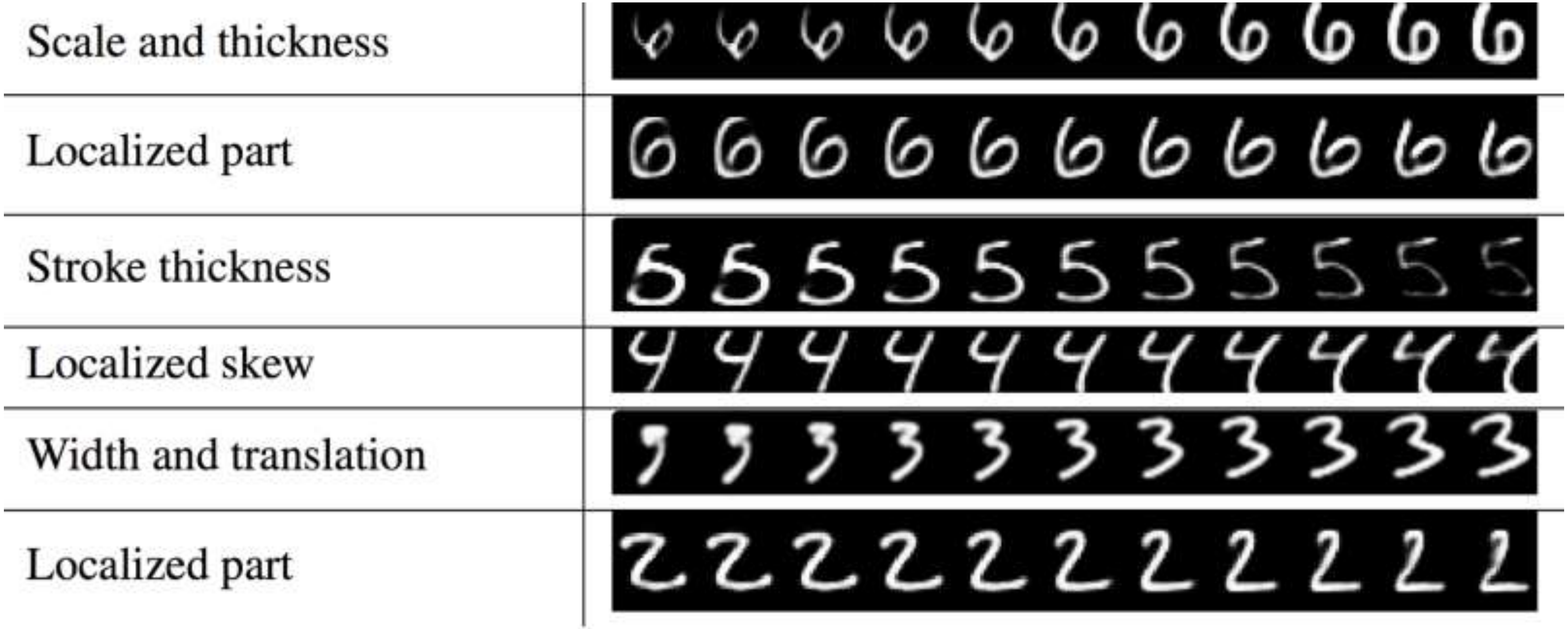
A CapsNet for MNIST – Decoder



(Figure 2 from the paper)

Aurélien Géron, 2017

Interpretable Activation Vectors



(Figure 4 from the paper)

Pros

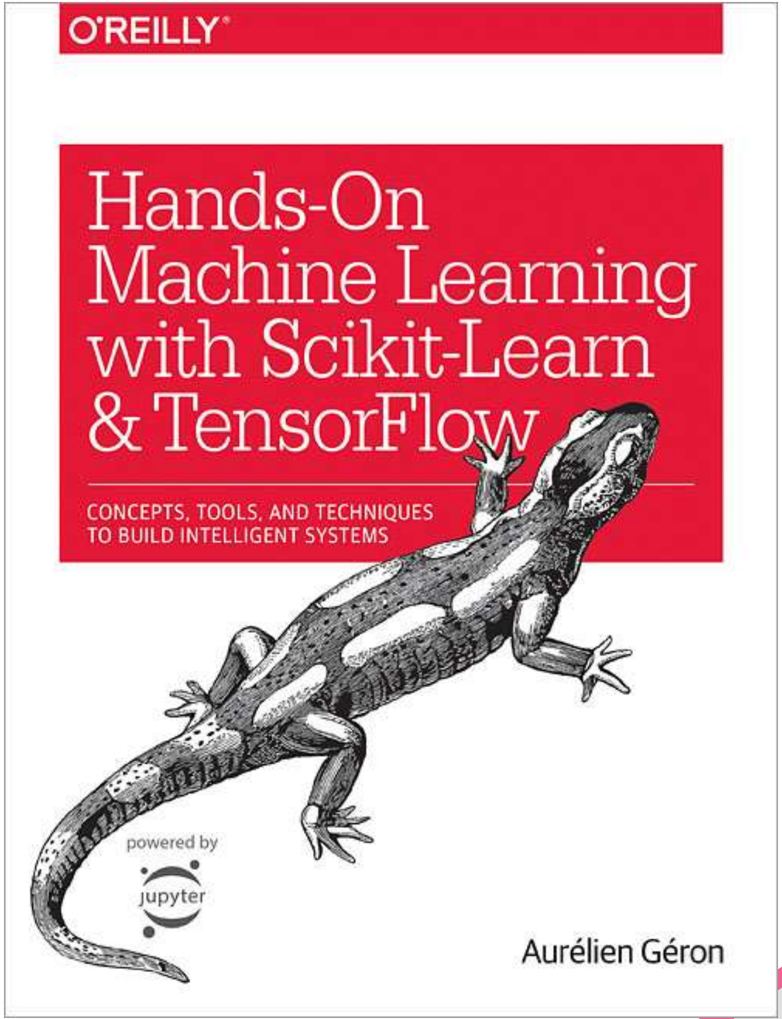
- Reaches high accuracy on MNIST, and promising on CIFAR10
- Requires less training data
- Position and pose information are preserved (equivariance)
- This is promising for image segmentation and object detection
- Routing by agreement is great for overlapping objects (explaining away)
- Capsule activations nicely map the hierarchy of parts
- Offers robustness to affine transformations
- Activation vectors are easier to interpret (rotation, thickness, skew…)
- It’s Hinton! ;-)

Cons

- Not state of the art on CIFAR10 (but it’s a good start)
- Not tested yet on larger images (e.g., ImageNet): will it work well?
- Slow training, due to the inner loop (in the routing by agreement algorithm)
- A CapsNet cannot see two very close identical objects
    - This is called “crowding”, and it has been observed as well in human vision

Implementations

- Keras w/ TensorFlow backend: https://github.com/XifengGuo/CapsNet-Keras
- TensorFlow: https://github.com/naturomics/CapsNet-Tensorflow
- PyTorch: https://github.com/gram-ai/capsule-networks



116 customer reviews

#1 Best Seller in Computer Vision & Pattern Recognition

Amazon: https://goo.gl/IoWYKD
Twitter: @aureliengeron
github.com/ageron