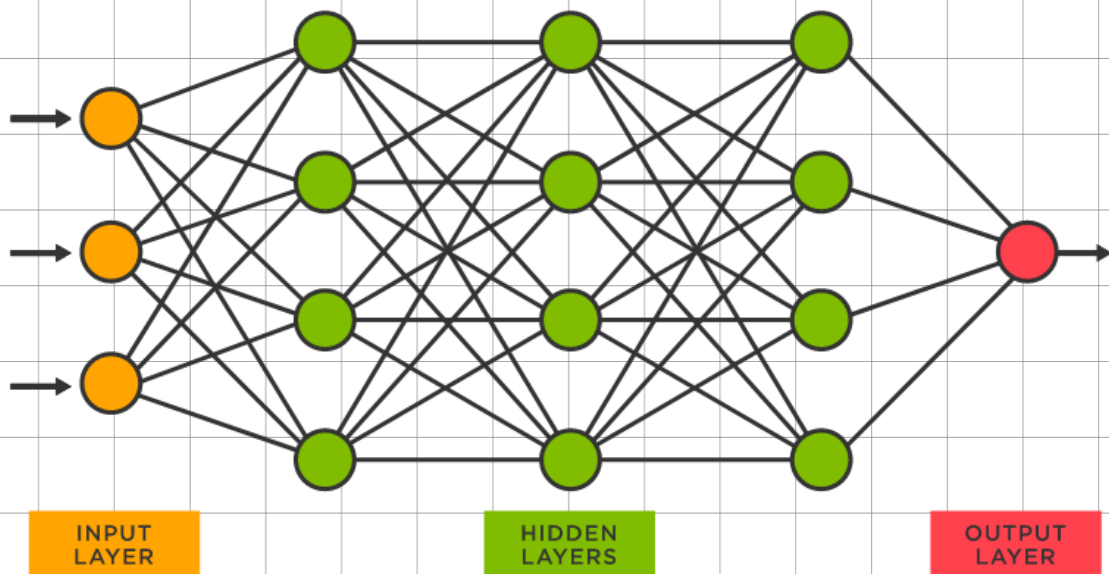


ภาพเรียนรู้ที่ 5 : Deep Learning

- 1 ตั้งคือ 1 layers
- Neural Network เชื่อมโยง layer ถัดไป input มีที่ node ที่ตามแต่ละ node จะเชื่อมโยง layer node ถัดไป
- ส่งผ่าน weight นำไปคูณกับ weight แล้ว sum รวมกัน
- มีการปรับปรุงเปลี่ยนแปลง จาก input -> output
- feature คูณกับ weight แล้วค่อยรวมกัน แล้วส่งต่อไปเรื่อย ๆ จนถึง output (เป็นตัวเลข เช่น 0.8 0.2 เป็นได้ตั้งแต่ 0-1 ถ้าน้อยกว่า 0.5 คำตอบคือ 0 ถ้ามากกว่า จะเป็น 1)
- สำหรับ 2 คลาส จะเพิ่ม Node



- เรียกว่า Fully connected layers ต่อทั้งหมดเลย / Dense
- ทำกับข้อมูลที่เป็นรูปภาพ
- ตัวอย่าง ขั้นตอนการเรียนรู้ ของ Neural Network
- การเรียนรู้ ต้องเอา features เอาไปคูณไปรวม จนออกมาเป็น output
- forward class คือ ค่าใน output node ไปด้านหน้า
- loss มีหลายอย่างเลย loss class -> จะเทียบกับค่า 0 หรือ 1 ว่าห่างไกลแค่ไหน
- backward class ทำในขากลับ นำความผิดพลาดมาปรับ weight ให้เข้าใกล้เลขที่ควรจะเป็นมากที่สุด (ใกล้กับค่าที่แท้จริงมากที่สุด)
- จบการเรียนรู้ 1 รอบ

เพิ่มเติม

- input เป็นรูปภาพ สิ่งที่เราเห็น แต่คอมพิวเตอร์จะเห็นเป็นตัวเลข
- ถ้าเป็นรูปภาพ แต่ละจุดในรูปจะมีความสำคัญ (features)
- convolutional จะสามารถเรียนรู้รูปภาพได้ แต่ Fully connected layers ไม่สามารถทำได้
- filter แต่ละอันจะเอาค่าของมันไปคูณกับค่าสี แล้วก็นำมา sum เป็นค่าสีสรุปตรงกลาง
- pooling คือการนำไปเฉลี่ย หรือ เบลอ จากทุก ๆ จุดมารวมเป็นค่าเดียว
- สิ่งที่ต้องทำ คือ ใช้ convolutions filter (เส้นเชื่อม) คูณค่าไปเรื่อย ๆ แปลว่า ภาพจะเล็กลงเรื่อย ๆ 3x3 คือ 9 จุด สรุปเป็น 1 จุด

เวลาทำหัด Convolutions

- ขนาดของ filter
- จำนวน filter

- สี่เหลี่ยมผืนผ้าที่เชื่อม คือจุดที่ตรงนั้น ไปเป็น 1 จุด
- output มี 10 node แปลว่ามี 10 class
- pooling คือการรวม layers ลดขนาดของข้อมูล เพื่อให้เหลือเฉพาะข้อมูลที่สำคัญๆ เท่านั้น ซึ่งมักจะนิยม นำมา ต่อกับชั้นคอนโวลูชัน โดยทั่วไปนิยมใช้การเลือกข้อมูลที่มีค่ามากที่สุด (Max Pooling) หรือ ค่าเฉลี่ย (Average Pooling) มาจากแต่ละช่องของเมตริกซ์ เพื่อสร้างเป็นเมตริกซ์ที่มีขนาดเล็ก

Tensorflow The Sequential model

url : https://www.tensorflow.org/guide/keras/sequential_model

- Sequential (ง่าย แต่สร้างได้แต่แบบไม่ซับซ้อน (จึงเป็นเส้นตรง))
- Functional (ยากขึ้นมากอีกนิด แต่สามารถปรับเส้นทาง ให้มีความซับซ้อนได้)
- Subclassing (ยากสุด แต่สามารถแก้ไขได้ตามที่เราต้องการ)

Sequential

- activation ช่วยปรับเลขไม่ให้เกิดการมั่วจนเกินไป
ปรับให้เหลือแค่ -1 -1 , relu จะทำให้เป็นแค่ 0-1
- Dense แปลว่าต่อกันหมด / Fully connected
- จะไม่ตรงเพราะว่า มันจะสร้าง node พิเศษมา 1 node เรียกว่า bias คือเป็นเลข 1 เสมอ
เพื่อให้เกิดการเรียนรู้ได้ดีมากยิ่งขึ้น มี weight ของมันเอง
- sigmoid คือ ดูจากค่า y กับ x activation function
- flatten คือการบีบรูป กลายเป็น เส้น 1 เส้น เป็น node
- learning less คือการ jump มาได้ไกลแค่ไหน แต่หากไปก็ไม่ได้ ต้องลงให้ต่ำสุดแปลว่า จะทำให้เกิดการเรียนรู้ที่ดี (ใช้ทำให้ train ได้เร็ว)

Functional

- skips connection คือข้าม layers มาเลย
- ResNet มี skips connection จะข้าม convolutional มา
- จะพบว่า accuracy ของ ResNet จะดีกว่า Functional ดีขึ้นเรื่อย ๆ

Data Training

- image propmentation ทำให้รูปมันหลากหลาย
- rescale ไม่สามารถใช้กับรูปภาพได้
- กราฟ validation มันจะค่อย ๆ ลดแล้วค่อย ๆ สูงขึ้น ตรงจุดต่ำสุด คือ overfit แปลว่ามันเริ่มรู้เต็มแล้ว
learning less/decay = learning less จะลดลงตาม epoch
- ควร set decay ด้วย
- cross-entropy loss prop. sum = 1 \rightarrow forward class
เอาคำตอบที่แท้จริงมาเทียบกับ \rightarrow ค่าความจริงมาคูณกับ log ของค่า p
ที่ผ่าน softmax มาแล้ว ถ้า ตอบถูกพอดีเลย ก็จะเป็น 0
- regression loss \rightarrow วัด mean square error / ตัว matrix ตัว loss ที่ต้องเปลี่ยน
- batch size คือ กำหนด weight ให้ปรับรูป ยิ่งเยอะยิ่งปรับสำหรับรูปภาพหลาย ๆ รูป