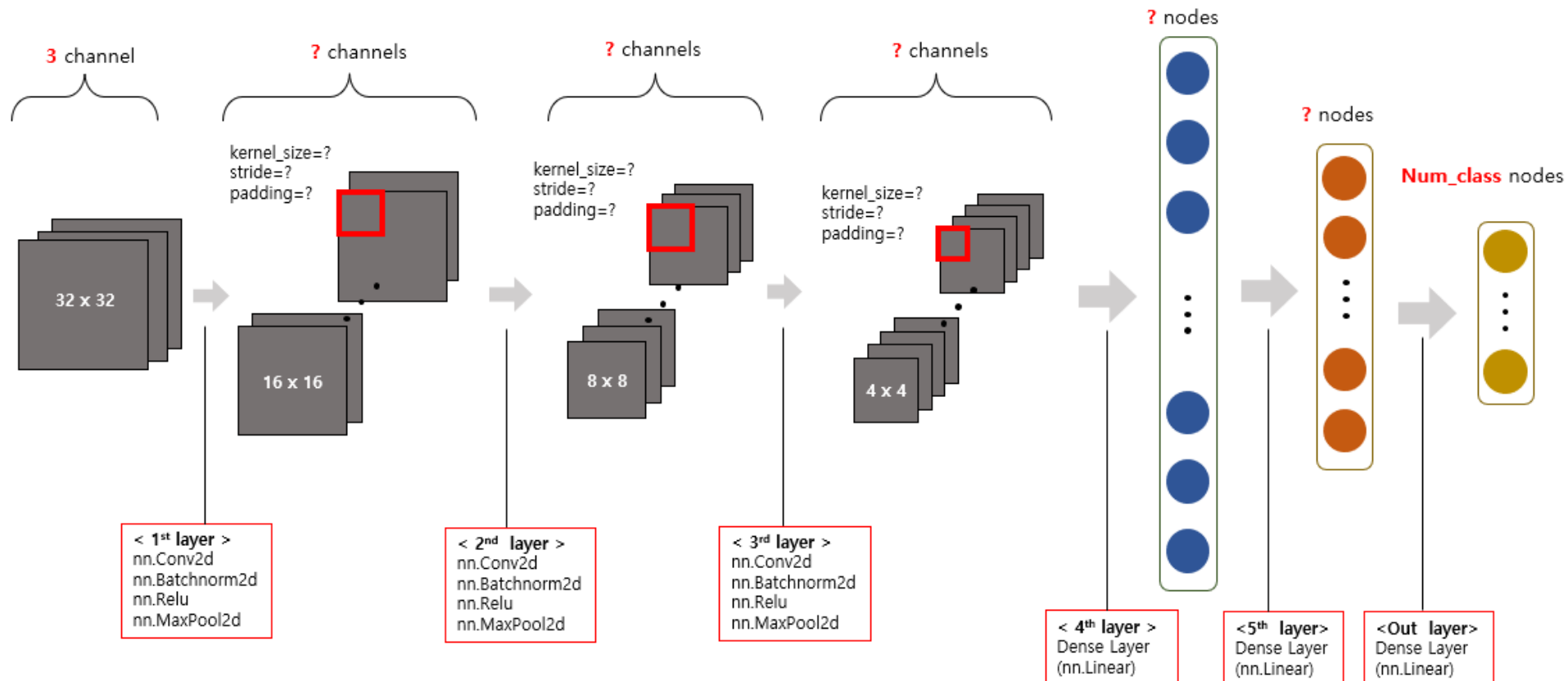


# HW2 : CIFAR10 Classifier with CNN

- CNN 을 통한 CIFAR10 Dataset Classifier Model 만들기
  - Model Structure : 아래 조건을 만족하는 5-layer Model 을 Pytorch 로 작성
  - ? 부분은 예러가 나지 않는 선에서 자유롭게 설정
  - Dense layer의 activation function은 자유



# HW2 : CIFAR10 Classifier with CNN

- 제한조건

- Layer의 개수는 앞 장의 그림과 동일하게 설정  
(3개의 Convolutional layer와 3개의 Dense layer)
- Max-pooling layer의 kernel size 는 2 x 2로 구성
- Optimizer : Adam Optimizer 사용
- Test dataset 에 대한 accuracy 가 65% 이상이 되도록 Training
- Epoch, Learning rate, Batch size, weight initialization 등의 hyperparameter 는 각자 tuning  
→ 보고서에 자세히 작성바람



# HW2 : CIFAR10 Classifier with CNN

- 제출 기한
  - 11/18 (금) 자정
  - 사이버 캠퍼스 과제란을 통해 제출 (Late의 경우, 하루 당 총점 10점에서 2점씩 감점)
- 제출 형태 : HW2\_학번.zip [압축 필수]
  1. Train python file : **train.py** (colab으로 작성한 경우 train.ipynb도 함께 제출)
    - Google Colab으로 작성시 '파일' -> '.py 다운로드'로 변환하여 제출  
+ ipynb파일도 함께 제출
  2. Saved Model : **model.pth**
    - 학습된 Model을 .pth 형식으로 저장
  3. Test Python file : **test.py**
    - 2번의 model.pth를 통해 학습된 model의 parameter를 불러와 train dataset 에 대한 성능 확인
    - Model 저장, 불러오기 [\[참고링크\]](https://pytorch.org/tutorials/beginner/saving_loading_models.html) : [https://pytorch.org/tutorials/beginner/saving\\_loading\\_models.html](https://pytorch.org/tutorials/beginner/saving_loading_models.html)
  4. Report : HW2\_학번\_이름.pdf
    - Learning Curve (Error 감소 그래프) 포함
    - 각 Layer를 통과한 시점에서의 output의 shape(size)를 포함
    - Accuracy를 올리기 위한 본인의 방법 설명

