



# 당뇨망막증 증상의 분석과 예측

(통계학과 김서영 / 소프트웨어학과 정민서)  
Department of Statistics, Sookmyung Women's University.

## Introduction

당뇨는 성인 인구의 8.8%를 차지하고 있는 흔한 질병이고 당뇨가 있는 경우 당뇨 망막 병증을 1년에 한번씩 검사를 받아야 하기 때문에 이를 검사하는 것은 매우 중요하다. 따라서 이를 모델을 통해 직접 구현해 보면 좋은 경험이 될 것이라 생각하여 본 주제로 선정했다.

이미지 딥러닝을 이용하여 자동화 분류모델을 구축하고, 그 모델을 training 시켜서 환자의 당뇨성망막병 진단을 자동화하는 것이 본 프로젝트의 목적이다.

환자의 망막 이미지와 질병 level 정도에는 유의미한 상관관계가 있다. 이러한 상관관계에 기반하여 자동화 분류 모델을 구축하면, 새로운 환자의 망막 이미지가 데이터로 들어왔을 때, 구축한 자동화 분류기를 이용해 환자가 병의 level을 추측할 수 있을 것이다.

더 나아가 질병의 증세의 정도를 판단하여 환자와 의료 기관 양 측의 시간과 인력, 비용을 절감하는 효과를 기대한다.

## Data

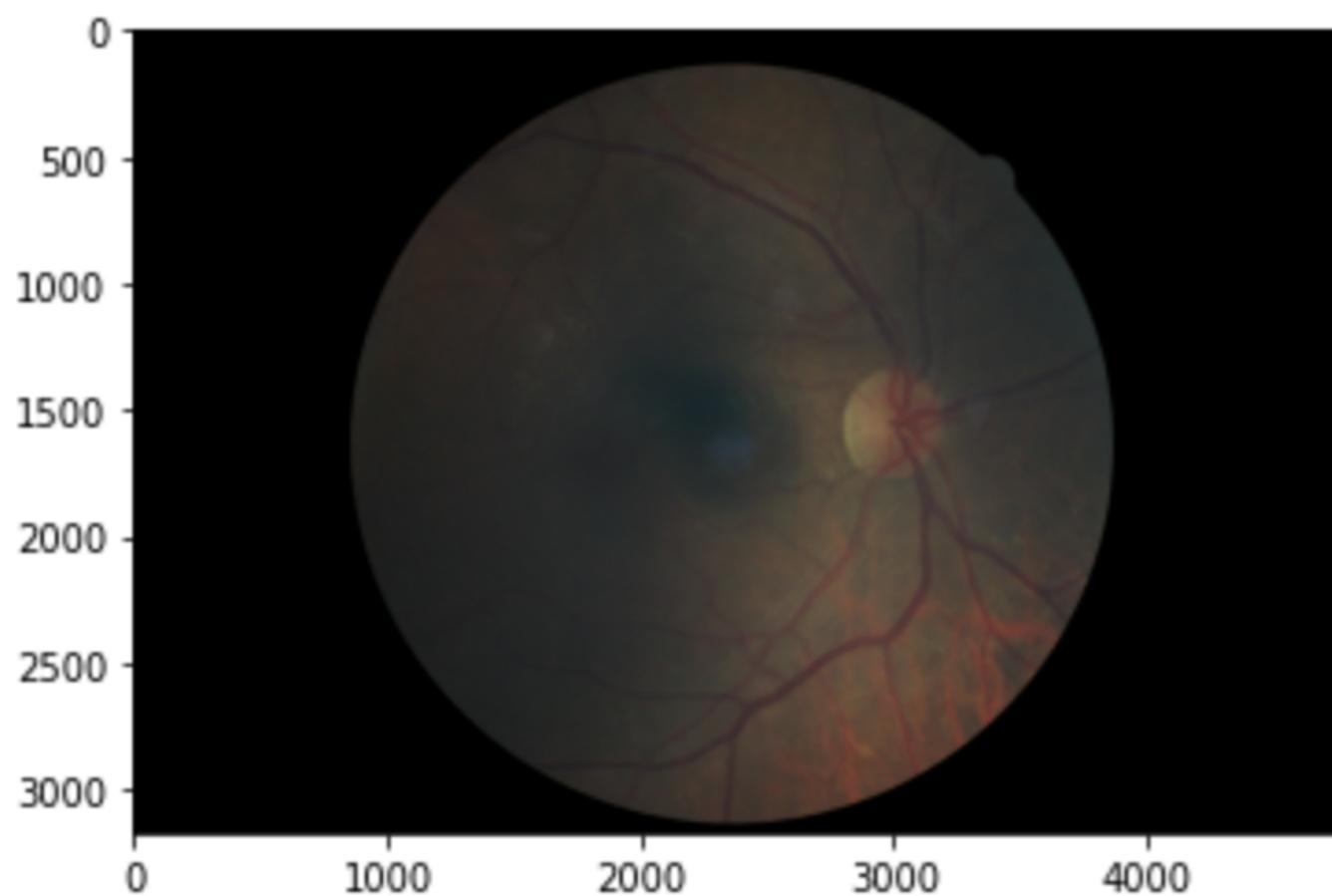
본 프로젝트는 캐글의 <https://www.kaggle.com/c/diabetic-retinopathy-detection/data>에 나타난 데이터셋을 참조하고 있다.

데이터는 당뇨성망막병의 증상이 있는 환자와 없는 환자의 고화질 컬러 망막 사진과, 환자들의 정보를 담은 csv 파일로 구성되어 있다.

환자의 정보를 담은 trainLabels.csv 파일에서는 환자들을 id에 따라 넘버링하고, 안구의 left, right를 구분한 파일 이름으로 저장되어 있으며, 병증이 있는 환자들은 증상의 정도가 1~4 클래스, 증상이 없는 경우 0번 클래스로 분류되어 있다.

### 1. 망막 이미지 데이터

```
[ ] #이미지 잘 보이는지 확인
img=mpimg.imread(path)
imgplot=plt.imshow(img)
plt.show()
```



```
[ ] #train 이미지 데이터수 확인
f_list = os.listdir('/content/drive/My Drive/통계분석실습/train')
print(len(f_list))
```

8408

8408개의 이미지 데이터가 존재한다.

8408개의 병 level(0~4)은 아래 사진과 같다.

	image	PatientId	path	exists	eye	level_cat
level	0	6149	6149	6149	6149	6149
1	588	588	588	588	588	588
2	1283	1283	1283	1283	1283	1283
3	221	221	221	221	221	221
4	166	166	166	166	166	166

level=0의 개수가 6149로, 다른 레벨들 보다 압도적으로 높은 수치를 보이고 있다. 이러한 class imbalance의 문제를 data processing 단계에서 해결하도록 한다.

## Data Preprocessing

### 1. 이미지 resize

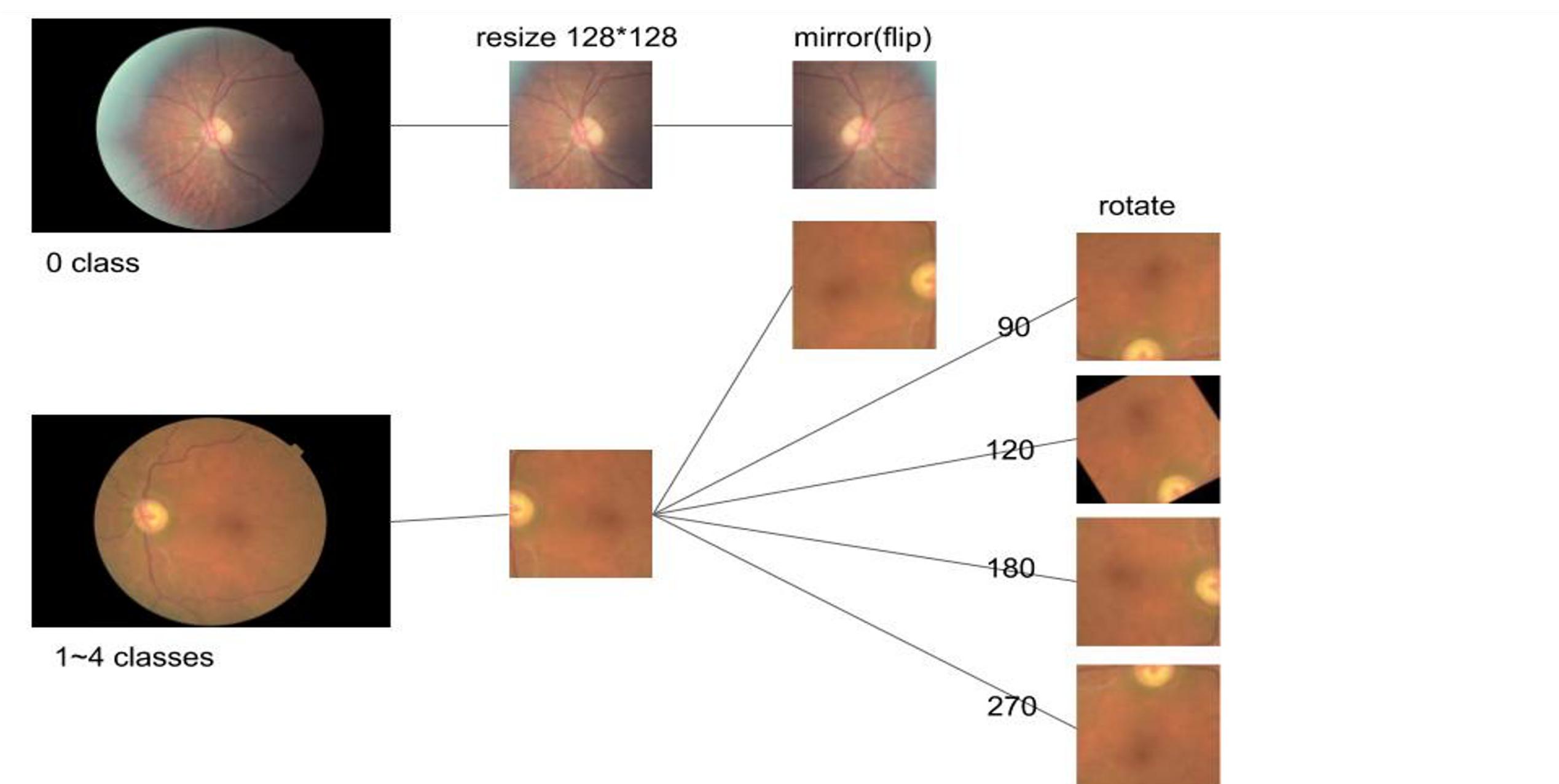
이미지 데이터가 매우 크기 때문에 계산 비용 절감을 위해 128x128 사이즈로 축소시킵니다.

### 2. 이미지 crop과 blur check

이미지 중 color space가 존재하지 않아서 train으로 적절치 못한 이미지를 제거합니다. 또한 이미지 데이터의 blur 정도를 계산하여 역시 적절치 못한 이미지를 drop시킵니다.

### 3. 이미지 rotate / mirror

이미지를 좌우반전시키고, 클래스 1~4에 속하는 이미지들은 추가적으로 90, 120, 180, 270도로 회전시켜 이미지를 강화시킵니다.



## Model

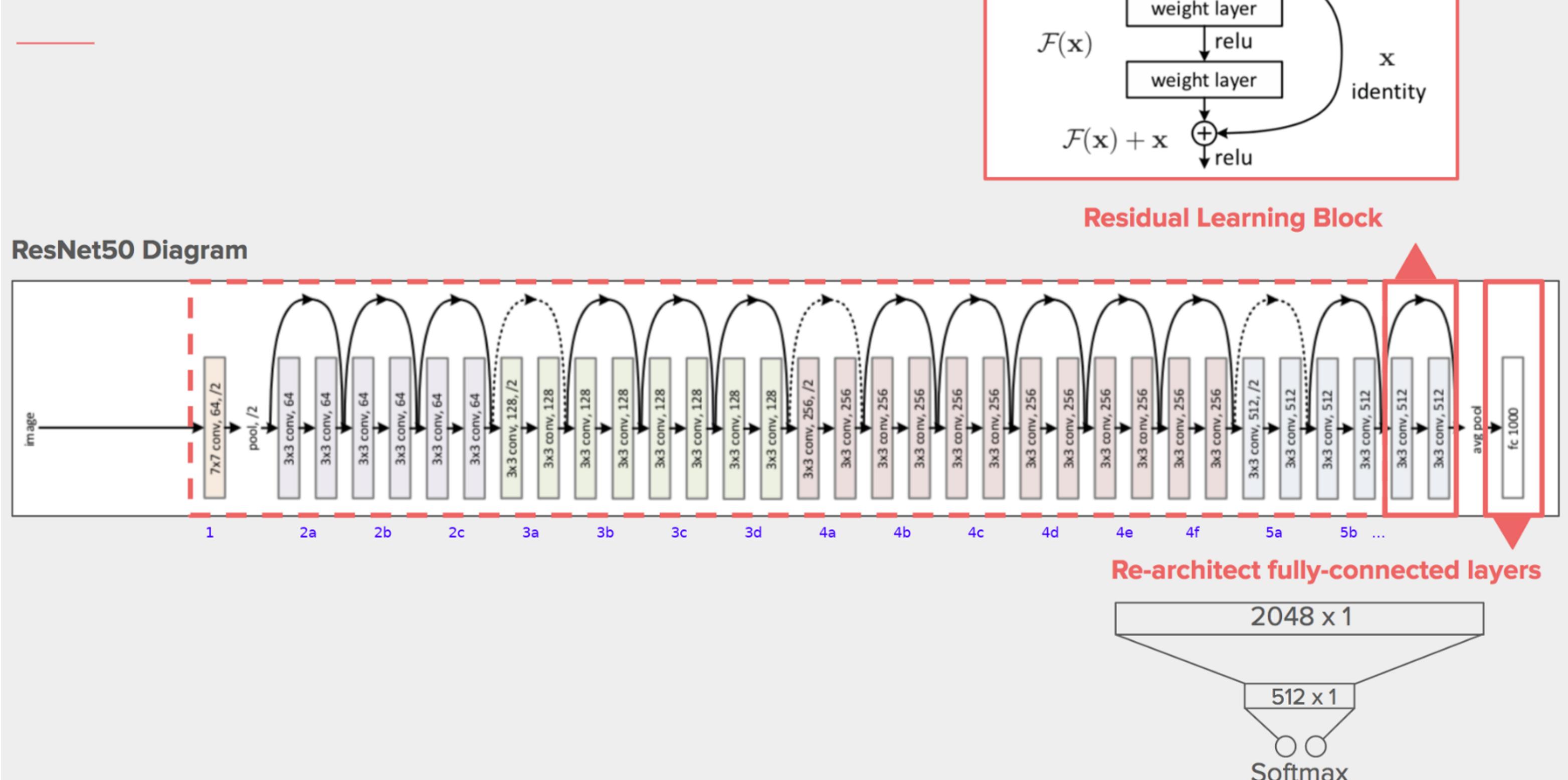
### 1. CNN

	input	filter	kernel	stride	padding	output
FC	Conv (3, 128, 128)		32	5	1	2 (32, 128, 128)
	Batchnorm					
	ReLU					
3x3 conv, 128	Conv (32, 128, 128)	18	3	1	1	1 (18, 128, 128)
	Batchnorm					
	ReLU					
3x3 conv, 128	Flatten (294912.)					(294912.)
	Linear (294912.)					(5.)
input						

	input	filter	kernel	stride	padding	output
FC	Conv (3, 128, 128)	64	12	4	2	(64, 31, 31)
	BatchNorm	.	.	.	.	(64, 31, 31)
	Maxpool	(64, 31, 31)	.	3	2	(64, 15, 15)
3x3 conv, 256	Conv (64, 15, 15)	192	5	1	2	(192, 15, 15)
	BatchNorm	(192, 15, 15)	.	.	.	(192, 15, 15)
	Maxpool	(192, 15, 15)	.	3	2	(192, 7, 7)
3x3 conv, 256	Conv (192, 7, 7)	384	3	1	1	(384, 7, 7)
	BatchNorm	(384, 7, 7)	.	.	.	(384, 7, 7)
3x3 conv, 384	Conv (384, 7, 7)	256	3	1	1	(256, 7, 7)
	BatchNorm	(256, 7, 7)	.	.	.	(256, 7, 7)
5x5 conv, 192	Conv (256, 7, 7)	256	3	1	1	(256, 7, 7)
	BatchNorm	(256, 7, 7)	.	.	.	(256, 7, 7)
Maxpool	Maxpool (256, 7, 7)	.	3	2	2	(256, 3, 3)
12x12 conv, 64	Flatten (2304.)					(2304.)
	Linear (2304.)					(5.)
Input						

### 2. ResNet

#### Retrain ResNet50



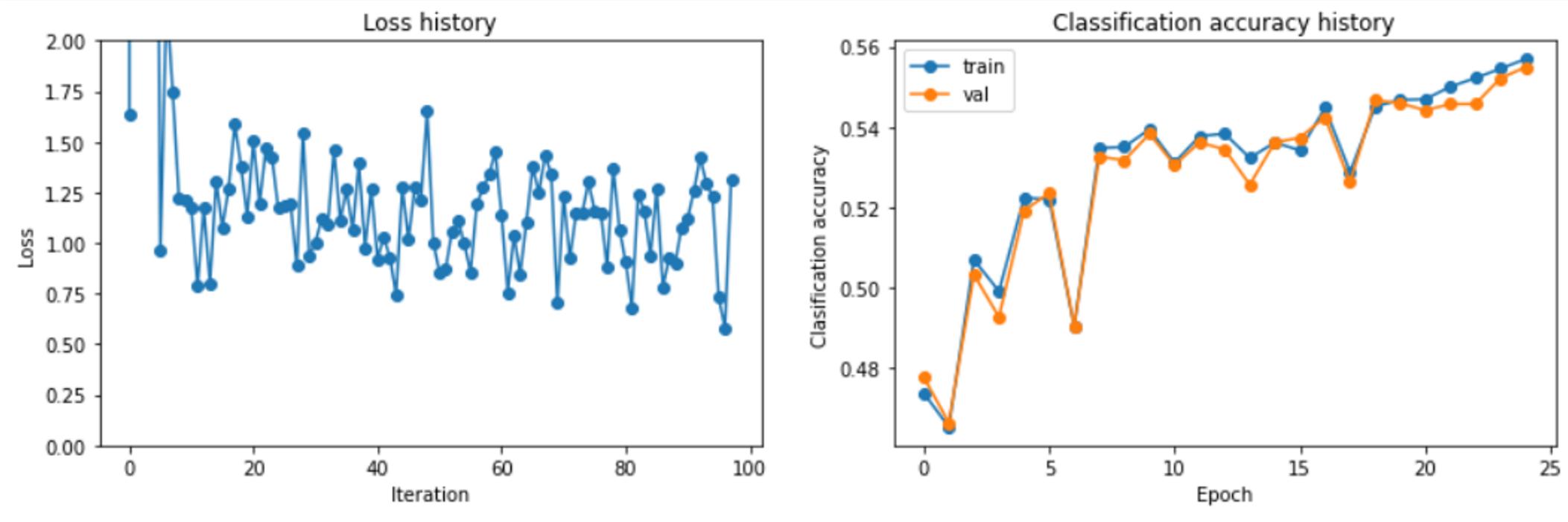


# 당뇨망막증 증상의 분석과 예측

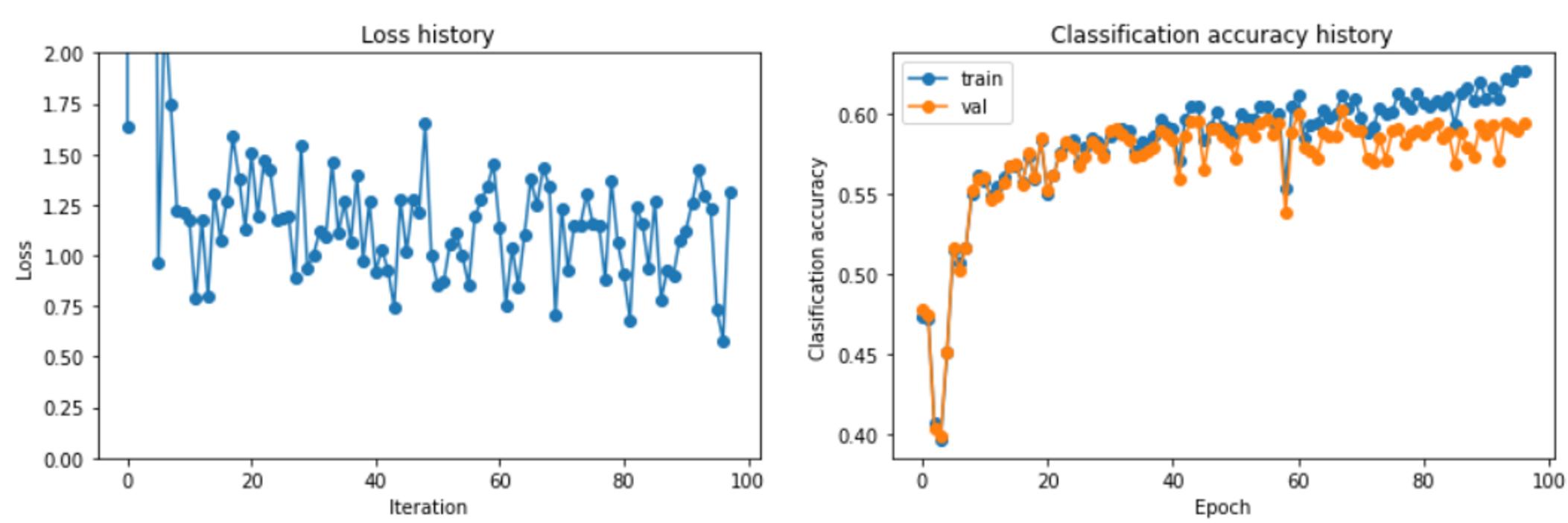
(통계학과 김서영 / 소프트웨어학과 정민서)  
Department of Statistics, Sookmyung Women's University.

## Results

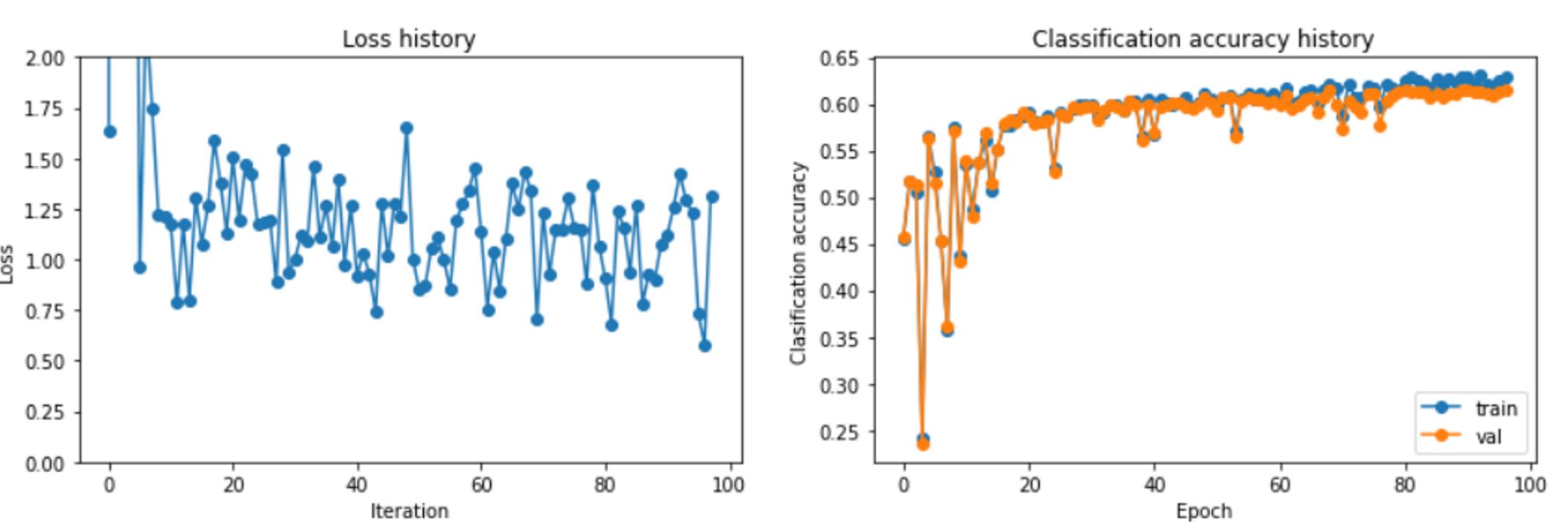
### 1. CNN



batch size를 64, learning rate를  $10^{-2}$ 로 설정한 2 layer CNN모델을 사용했을 때 test set에서의 정확도가 56.71%가 나오기 때문에, 같은 모델에서 batch size와 learning rate를 조절하며 성능을 개선시키고자 했다.



batch size = 16, learning rate=2e-3으로 설정했을 때 test set에서의 성능이 약 60%로 향상되는 것을 확인할 수 있었으며, 해당 hyper-parameter를 더 깊은 두 번째 모델에 적용해 더 높은 성능을 얻고자 했다.

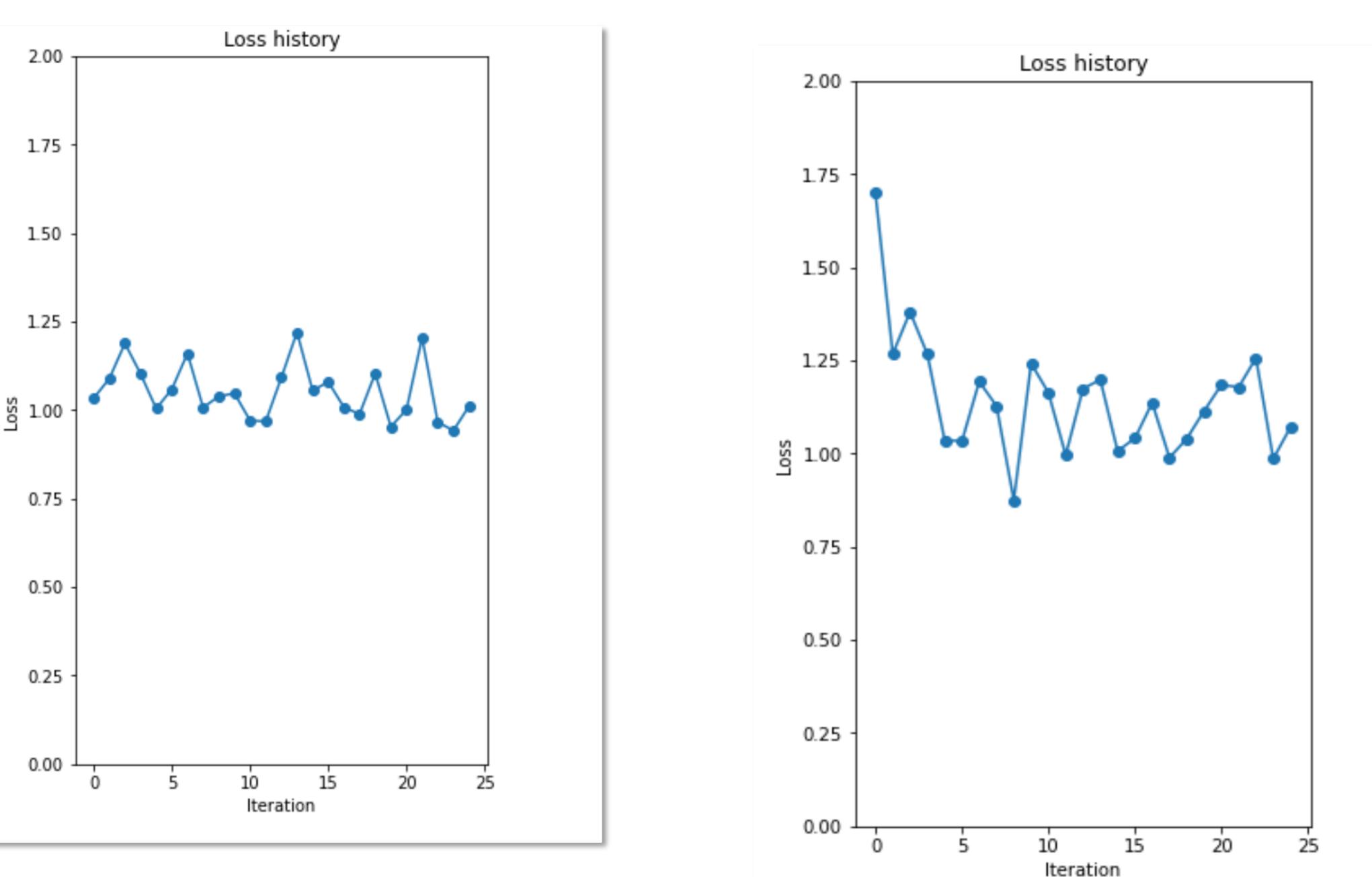


더 깊은 두 번째 모델에 적용시킨 결과 test set에서의 정확도 61.6%로 레이어의 깊이와 복잡도가 유의미한 변화를 이끌어내지 못한다는 한계를 발견했다.

### 1. ResNet50 & ResNet101

모델의 성능을 향상시키기위하여 batch size/ optimizer/ learning rate 등을 조절했다.

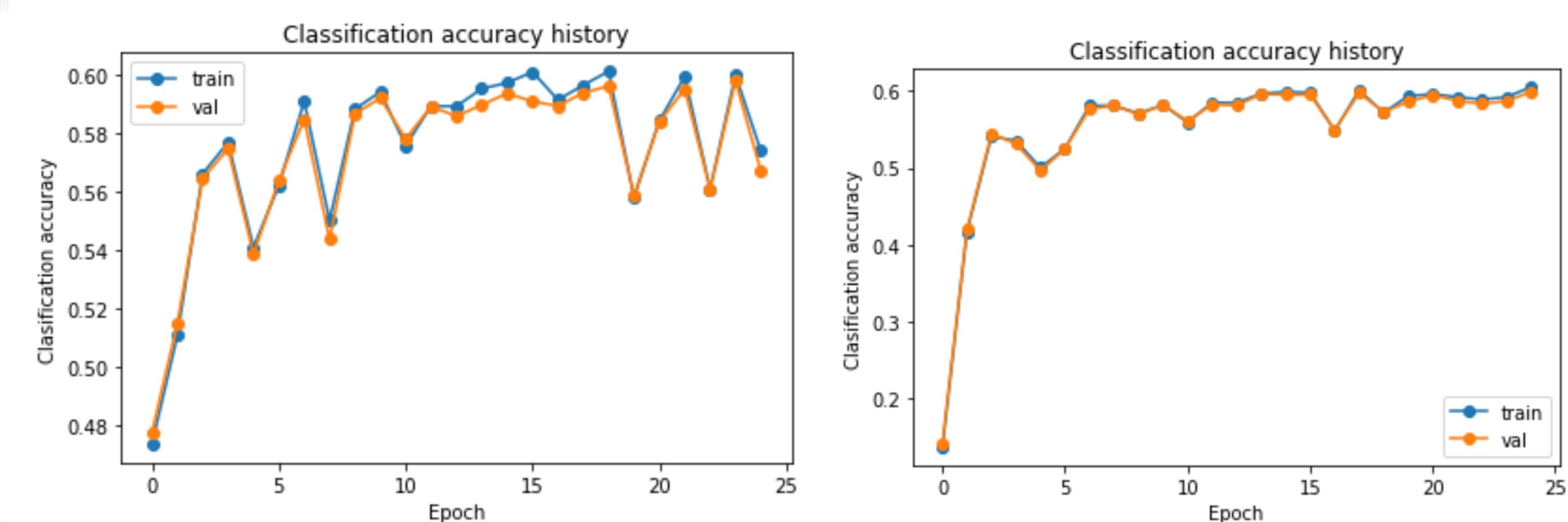
아래의 loss 값들의 추이를 보면서 loss 값을 줄이려고 했지만 iteration 마다 loss값의 편차가 크지 않음을 확인했다.



ResNet50 과 ResNet101에서는 test set의 정확도가 각각 60.11%과 59.7%의 결과가 나왔다. 즉 ResNet101의 성능이 더 우수할것이라고 기대했지만 비슷한 수준으로 확인되었다.

## Discussion

구현한 모델 세개 CNN, ResNet50, ResNet100 모두 train데이터와 validation데이터의 accuracy가 60%정도로 상당히 비슷한 값이 나타났다.  
<그라프>



train데이터에서의 accuracy가 60% 언저리 밖에 못미친 한계는 고해상도의 망막 데이터를 128x128 픽셀로 downsize하면서 유의미한 혹은 trivial한 feature들을 무시함으로써 발생하였다고 생각해 볼수 있다. 의료데이터에서는 사소한 값이 큰 차이를 발생시키면서, 병의 원인에 큰 요인으로 작용할 수 있기 때문이다.

model의 train 정확도와 validation의 정확도가 상당히 비슷한 이유로는 이미지 자체의 noise 처리 한계라고 볼 수 있다. 이미지들을 밝기와 이미지 내 망막의 위치가 데이터마다 달라서 우리는 이러한 noise를 처리하는데 어려움을 겪었다.

또한, class imbalance의 문제를 해결하기 위하여 1~4 level의 사진을 mirror/rotate 시킴으로써 데이터의 양을 증가 시켰고 이 복사된 이미지들을 랜덤으로 train/validation/test set에 분배하였다. 결국 train/validation/test의 이미지 feature들이 비슷해졌기 때문에, 역시 accuracy도 비슷하게 나오지 않았나 싶다.

## Future

- 모델의 최고 성능을 약 60% 까지밖에 끌어올리지 못했기 때문에, 이를 개선하기 위해 밝기를 일정하게 조절하는 전처리 과정을 수행한다.
- pixel size를 128\*128로 downsize한 데이터로 train을 시킨 후 다시 사이즈를 두배로 키운 데이터로 retrain시켜 작은 크기의 데이터에서 얻지 못한 정보를 추가적으로 얻는다.
- class imbalance를 줄이기 위한 방법으로 class가 1~4인 데이터들을 하나의 class로 묶어 binary classification, 즉 환자가 당뇨성망막병증의 유무만을 확인 할 수 있는 모델을 구축한다.
- 새로운 데이터에서의 분류 정확도를 개선하기 위해 Inception-v3나 GoogLeNet과 같은 다른 모델도 구축해본다.