자료분석특론 기말과제

202STG01 고유정

머신러닝을 이용한 최적 모형 찾기

목차

I. Mushroom data

1) 자료설명

출처 : https://www.kaggle.com/uciml/mushroom-classification

자료설명 :

1987년 4월 27일에 UCI에서 사용한 자료다. 자료는 Agaricus and Lepiota Family Mushroom에 속해 있는 gilled mushroom의 23가지 종류를 변수로 뒀다. 데이터는 8124개의 gilled mushroom의 특성을 나타내고 있으며 모든 변수는 23개로 버섯의 특징들을 각각 나열한 범주형 변수다. 'class'를 y로 두고 나머지 변수들을 x로 두어 식용 가능한 버섯과 독성을 띄는 버섯을 구분하는 중요 특성/요인을 분류하고 최적 예측 모형을 알아보려 한다.

변수설명 :

-classes: edible=e, poisonous=p

-cap-shape: bell=b,conical=c,convex=x,flat=f, knobbed=k,sunken=s

-cap-surface: fibrous=f,grooves=g,scaly=y,smooth=s

-cap-color: brown=n,buff=b,cinnamon=c,gray=g,green=r,pink=p,purple=u,red=e,white=w,yellow=y

-bruises: bruises=t,no=f

-odor: almond=a,anise=l,creosote=c,fishy=y,foul=f,musty=m,none=n,pungent=p,spicy=s

-gill-attachment: attached=a,descending=d,free=f,notched=n

-gill-spacing: close=c,crowded=w,distant=d

-gill-size: broad=b,narrow=n

-gill-color:

black=k,brown=n,buff=b,chocolate=h,gray=g,
green=r,orange=o,pink=p,purple=u,red=e,white=w,yellow=y

-stalk-shape: enlarging=e,tapering=t

-stalk-root:
bulbous=b,club=c,cup=u,equal=e,rhizomorphs=z,rooted=r,missing=?

-stalk-surface-above-ring: fibrous=f,scaly=y,silky=k,smooth=s

-stalk-surface-below-ring: fibrous=f,scaly=y,silky=k,smooth=s

-stalk-color-above-ring:
brown=n,buff=b,cinnamon=c,gray=g,orange=o,pink=p,red=e,white=w,yello
w=y

-stalk-color-below-ring:
brown=n,buff=b,cinnamon=c,gray=g,orange=o,pink=p,red=e,white=w,yello
w=y

-veil-type: partial=p,universal=u

-veil-color: brown=n,orange=o,white=w,yellow=y

-ring-number: none=n,one=o,two=t

-ring-type:
cobwebby=c,evanescent=e,flaring=f,large=l,none=n,pendant=p,sheathing=
s,zone=z

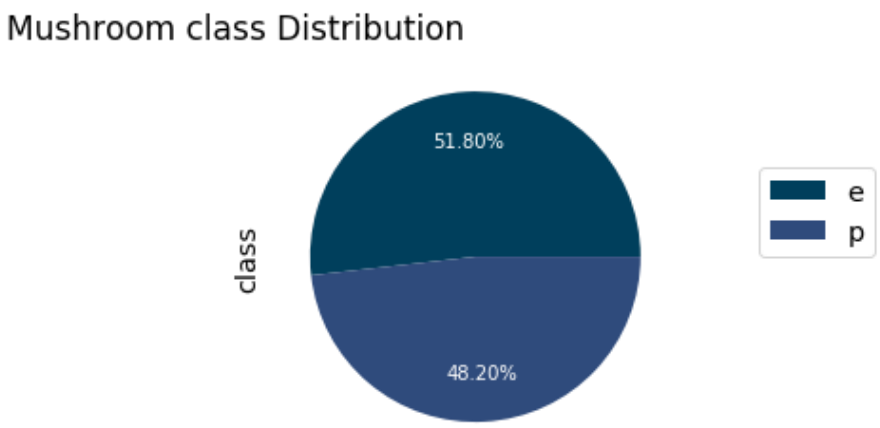
-spore-print-color:
black=k,brown=n,buff=b,chocolate=h,green=r,orange=o,purple=u,white=w
,yellow=y

-population:
abundant=a,clustered=c,numerous=n,scattered=s,several=v,solitary=y

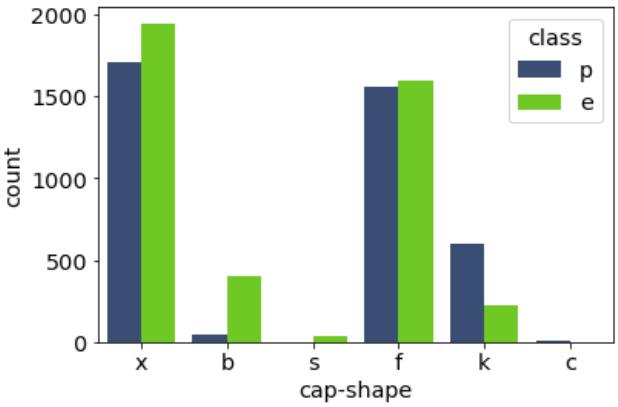
-habitat:

grasses=g,leaves=l,meadows=m,paths=p,urban=u,waste=w,woods=d

2) EDA

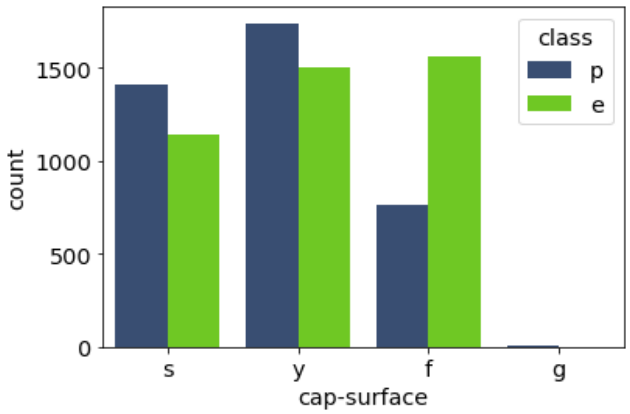


Class는 p,e 두개로 분류되며 p : 독성, e : 식용가능 을 나타낸다. 식용 가능한 버섯이 조금 더 비율이 높다.

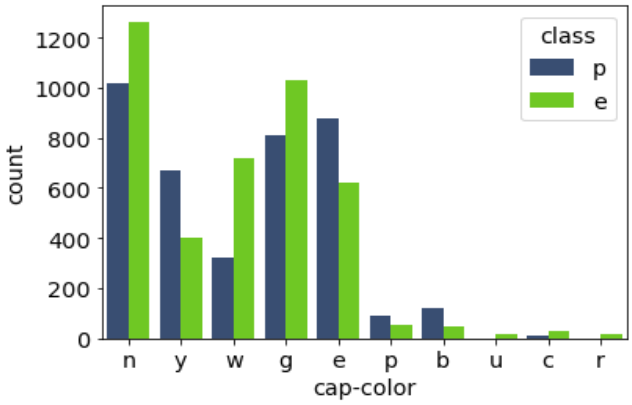


b=bell, c=conical, x=convex, f=flat, k=knobbed, s=sunken

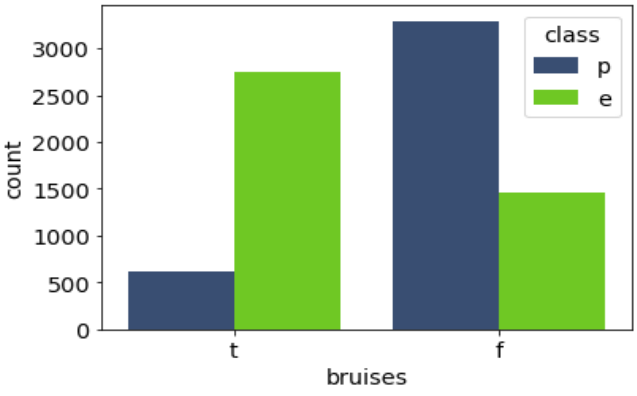
cap-shape에 따른 class의 분포를 살펴보았다. 독성과 식용버섯을 분류하는데 큰 영향을 끼치지 않는 것을 알 수 있다.



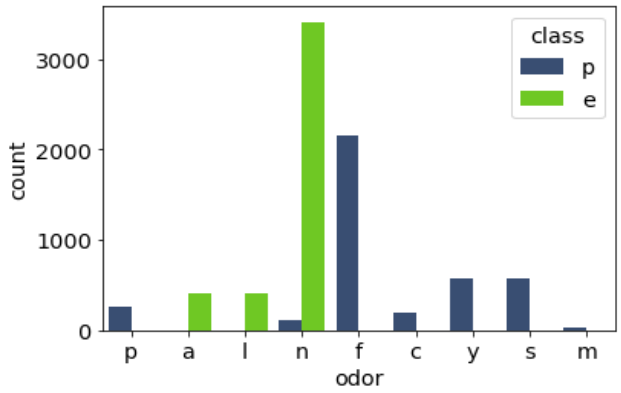
cap-surface에 따른 class의 분포를 살펴보았다. 독성과 식용버섯을 분류하는데 큰 영향을 끼치지 않는 것을 알 수 있다.



cap-color에 따른 class의 분포를 살펴보았다. 독성과 식용버섯을 분류하는데 큰 영향을 끼치지 않는 것을 알 수 있다.

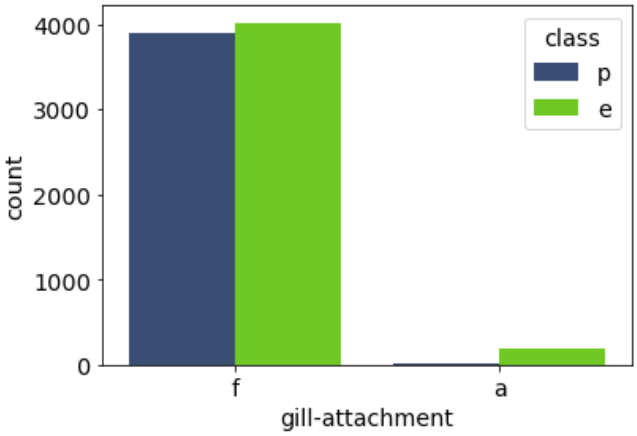


bruises에 따른 class의 분포를 살펴보았다. bruises=t, no bruises=f 를 나타낸다. 멍이 있는 버섯은 식용인 경우가 훨씬 많고, 멍이 없는 버섯은 독성을 띈 경우가 확연히 많다.



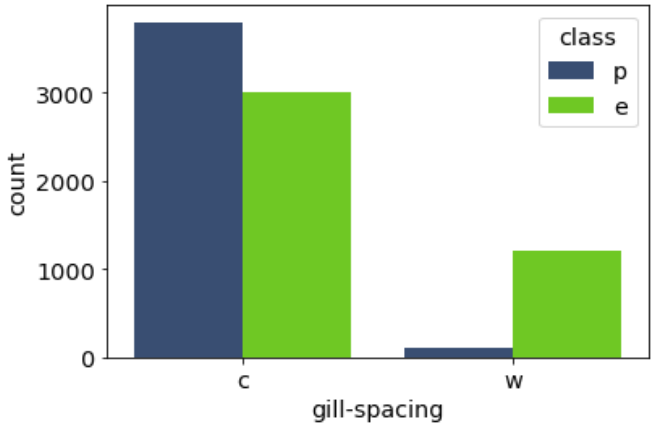
odor(향)은 극명한 분류를 보여준다.

향이 almond=a, anise=l, none=n 인 버섯은 식용인 경우가 대다수이며, 나머지 향을 띄는 버섯은 독성을 띈다.

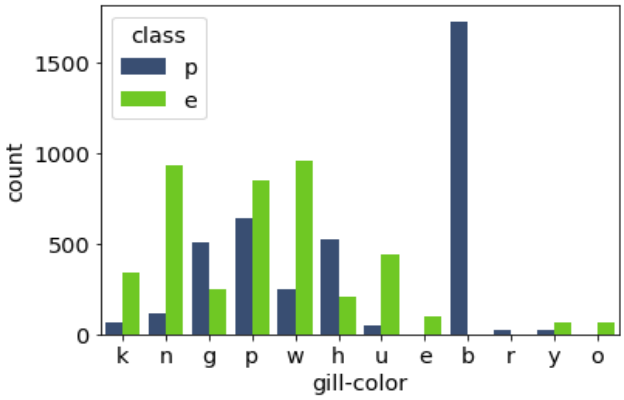


a : attached, f : free

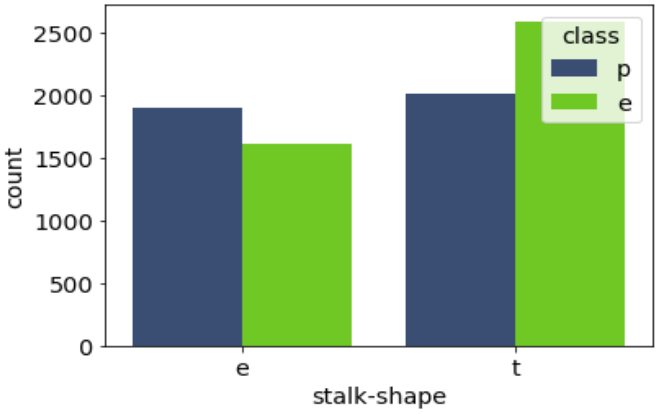
gill-attachment가 f인 경우가 훨씬 많기 때문에 class 분류에 영향을 준다고 보기 어렵다.



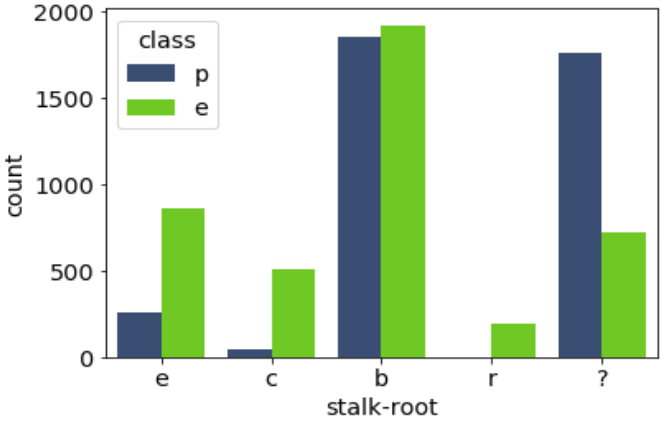
gill-spacing이 c인 경우 class와 큰 상관관계는 덜하지만 w의 경우 대부분 식용이 가능하다는 점을 알 수 있다.



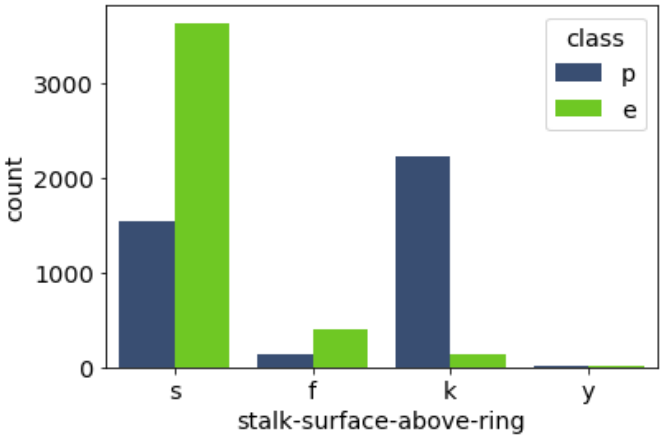
버섯의 경우 gill-color가 buff(b)일 때 무조건 독성을 띄며 그 수가 많다. Gill-color가 n,w,u인 경우 식용 버섯인 경우가 대다수다.



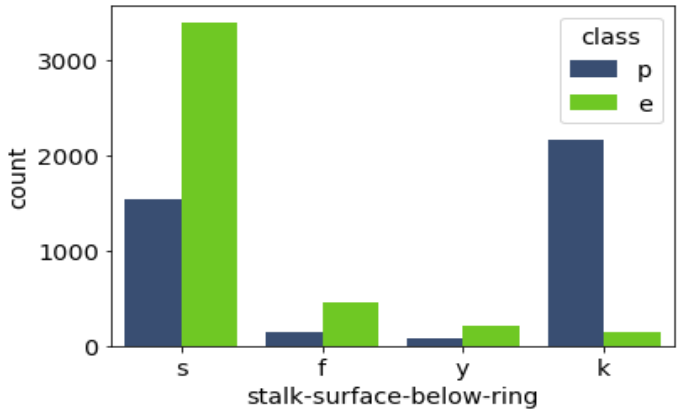
e, t 둘다 균등한 양상을 보이기에 class 분류에 큰 영향이 없다는 점을 확인할 수 있다.



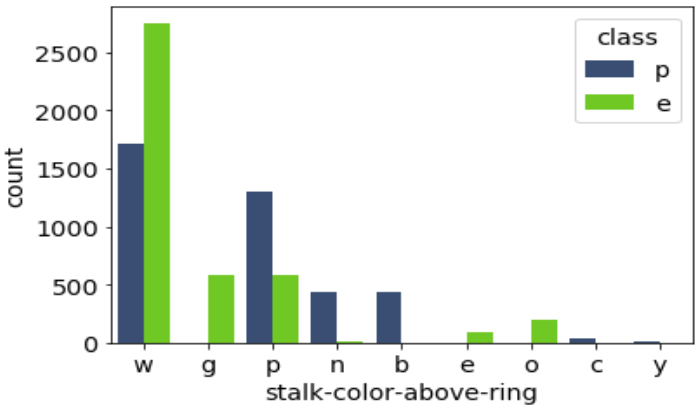
stalk-root가 r 일때는 식용, e, c일 때는 주로 식용 버섯이며 ?인 경우 대부분 독성 버섯이다. 그러나 class 분류에 큰 영향을 끼치는 요인은 아니다.



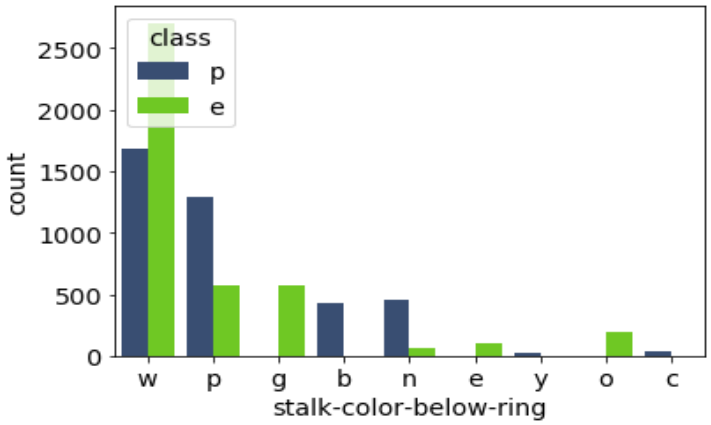
stalk-surface-above-ring이 s일 때는 주로 식용이며 k일때는 압도적으로 독성버섯인 경우가 많다.



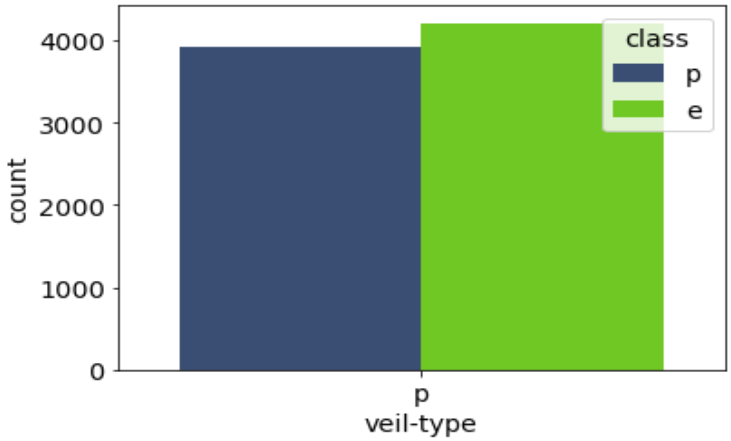
stalk-surface-below-ring이 s일때는 주로 식용이며 k일때는 압도적으로 독성버섯이 많아 위에 stalk-surfac-above-ring 변수와 양상이 비슷하다.

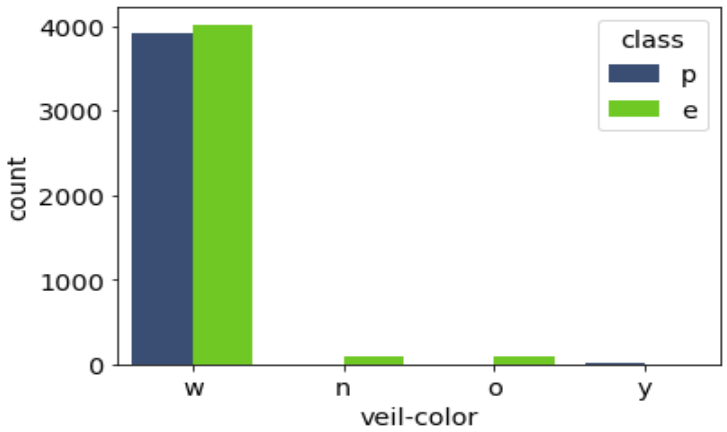


stalk-color-above-ring은 그래프에서 보여지다시피 class 분류에 큰 영향을 끼치지 않는다.

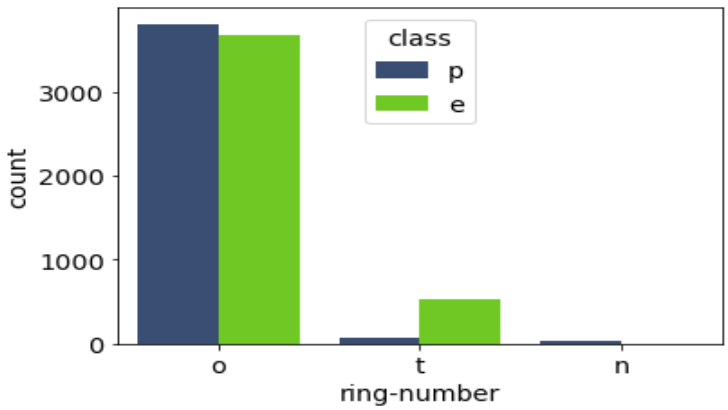


stalk-color-below-ring이 g 일때는 식용, b일때는 독성을 띈다.

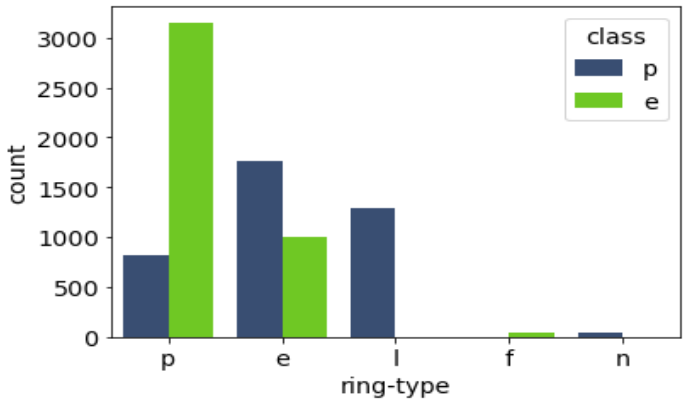




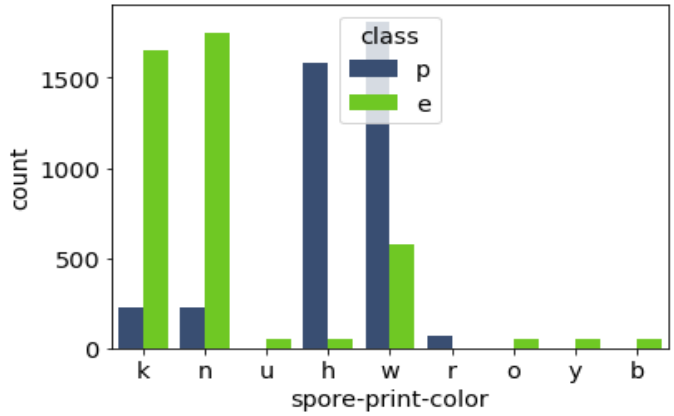
Veil-color가 대부분 w이지만 n, o 인 경우 식용, y인 경우 독성으로 분류된다.



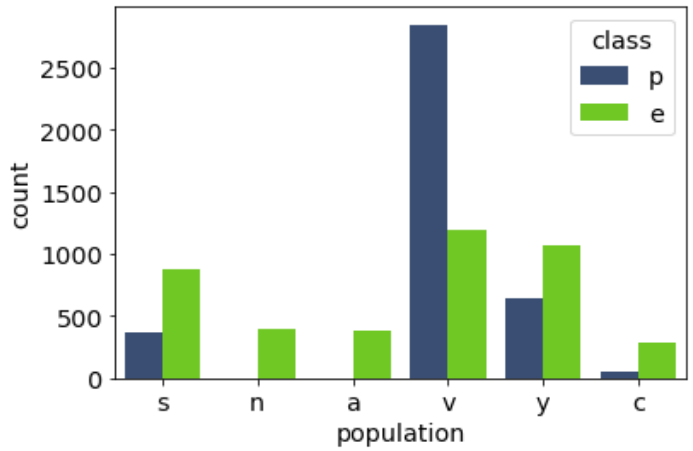
ring-number가 t인 경우 대부분 식용버섯이다.



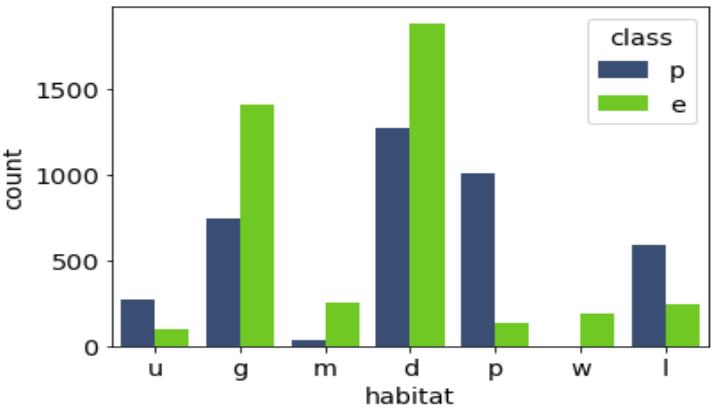
ring-type 이 p인 경우 압도적으로 식용 버섯인 경우가 많으며, e와 l인 경우 독성인 경우가 많다.



spore-pring-color가 k, n일 때 주로 식용 버섯이며, h, w일때는 주로 독성을 띈 버섯으로 분류된다.



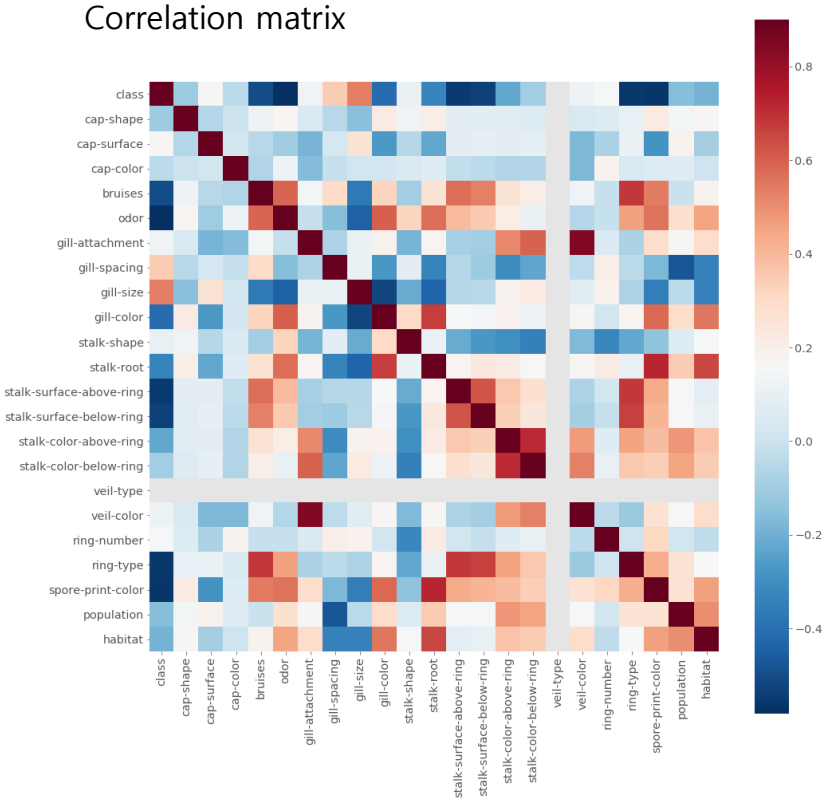
population이 v인 경우를 제외하고 대부분 식용으로 분류된다.



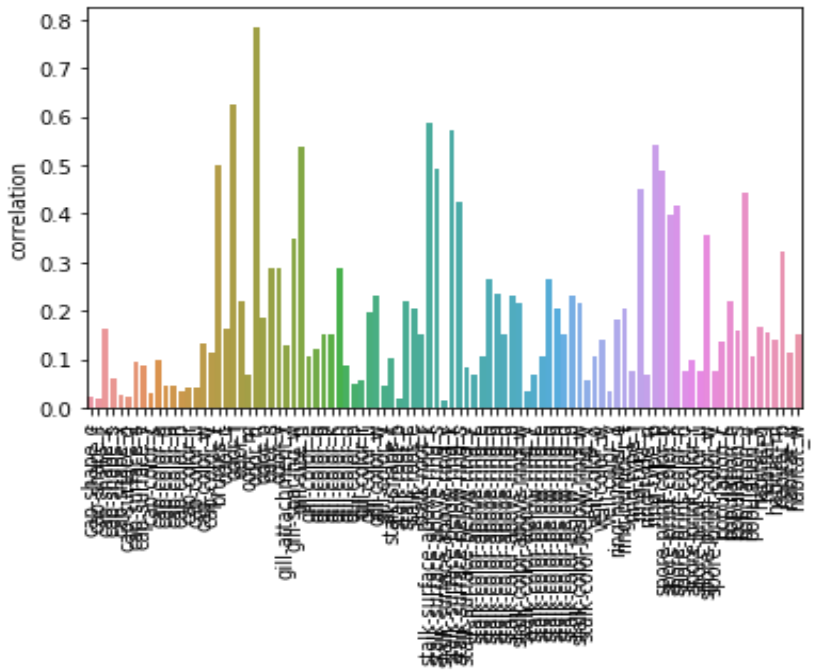
grasses=g,leaves=l,meadows=m,paths=p,urban=u,waste=w,woods=d

다른 서식지는 독성과 식용버섯이 고르게 분포하지만 w인 경우 전부 식용버섯이다.

Correlation matrix



Feature selection through investigation correlation importance



Class에 대한 변수들간의 상관관계를 그려보았다. 그리고 correlation

importance를 찾아 점수가 높은 feature를 선택했다. Correlation이 0.4보다 큰 feature를 상위변수로 지정하여 추출한 결과, 아래의 13개의 feature가 선정되었다.

feature	correlation
odor_n	0.785557
odor_f	0.623842
stalk-surface-above-ring_k	0.587658
stalk-surface-below-ring_k	0.573524
ring-type_p	0.540469
gill-size_n	0.540024
bruises_t	0.501530
stalk-surface-above-ring_s	0.491314
spore-print-color_h	0.490229
ring-type_l	0.451619
population_v	0.443722
stalk-surface-below-ring_s	0.425444
spore-print-color_n	0.416645

- **독성버섯의 결정적인 요인**

  Odor_f : foul

  Stalk-surface-above-ring_k : silky

  Stalk-surfce-below-ring_k : silky

  gill-size_n : narrow

  spore-print-color_h : chocolate

  ring-type_l : large

  population_v : several

- **식용가능한 버섯의 결정적 요인**

  Odor_n : none 무향

  ring-type_p : pendant

  bruise_t : 멍이 없는 경우

  stalk-surface-above-ring_s: smooth

  stalk-surface-below-ring_s : smooth

  spore-print-color_n : brown

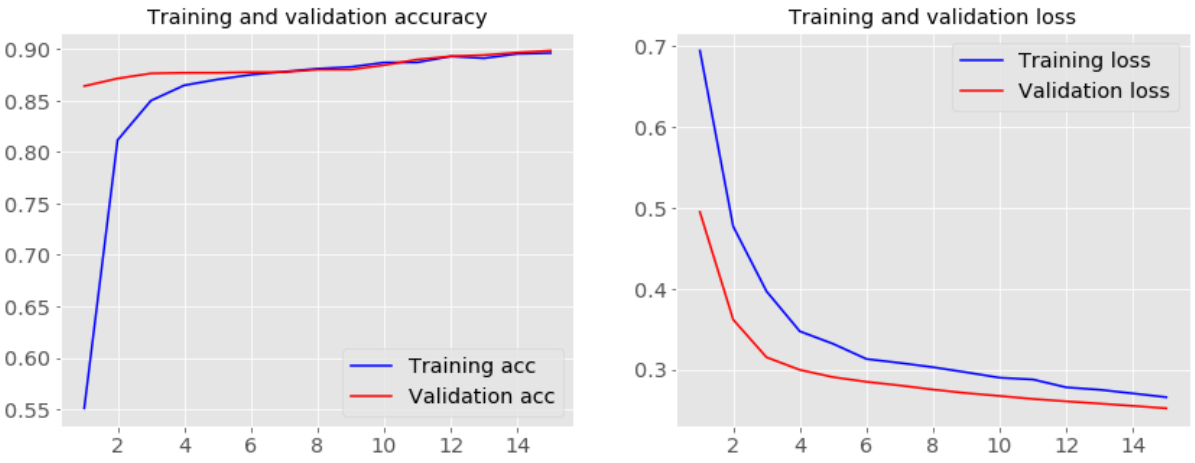
3) 모형탐색

데이터를 불러와서 결측치와 shape을 확인한 결과 결측치는 없었다. Class 데이터는 y로, 나머지 변수 데이터는 x로 지정했다. Class에서 p와 e를 0과 1로 숫자를 부여했다. 또한 x 데이터의 경우 전부 범주형이기에 원핫인코딩을 한뒤 standard scaling을 했고 pca를 사용하여 데이터 전처리를 하였다.

➔ 총 7가지의 모형을 fitting 시켜 accuracy score를 비교해보았다. 과적합을 방지하기위하여 GirdSearchCV를 이용하여 하이퍼파라미터를 튜닝하여 모델을 적합시켰다.

Model	Accuracy
Decision Tree	0.971
Random Forest	0.974
Support Vector Machine	0.978
Logistic regression	0.879
KNN	0.985
XGboost	0.985
CNN	0.896

CNN Training and Validation graph



4) 최적 모형 선택

위에 accuracy를 따져본 결과 점수가 제일 높은 SVM 모델을 최적모형으로 선택했다.

II. Garbage image data

1) 자료설명

출처: https://www.kaggle.com/asdasdasasdas/garbage-classification

자료설명: 쓰레기 분류 이미지 데이터로 6가지 분리수거 기준으로 나뉘어져 있으며 모든 쓰레기들은 image 파일로 저장되어 있다.

분류 설명: 1. cardboard 상자, 두꺼운 종이

2. glass 유리

3. metal 금속

4. paper 종이

5. plastic 플라스틱

6. trash 일반쓰레기

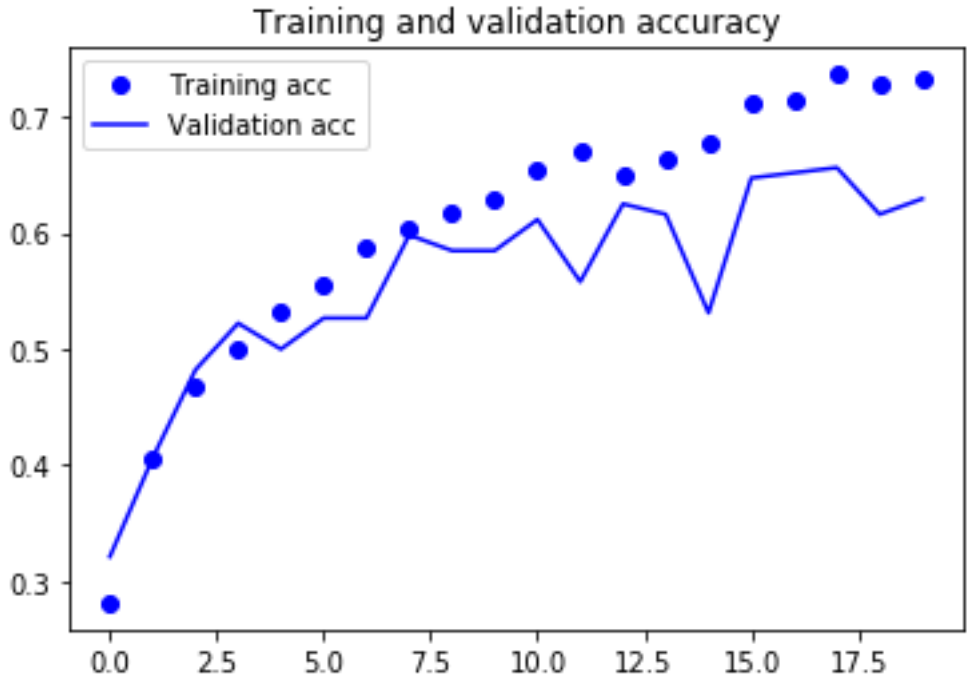
2) 모형 탐색과정 $ 최적 모형선택

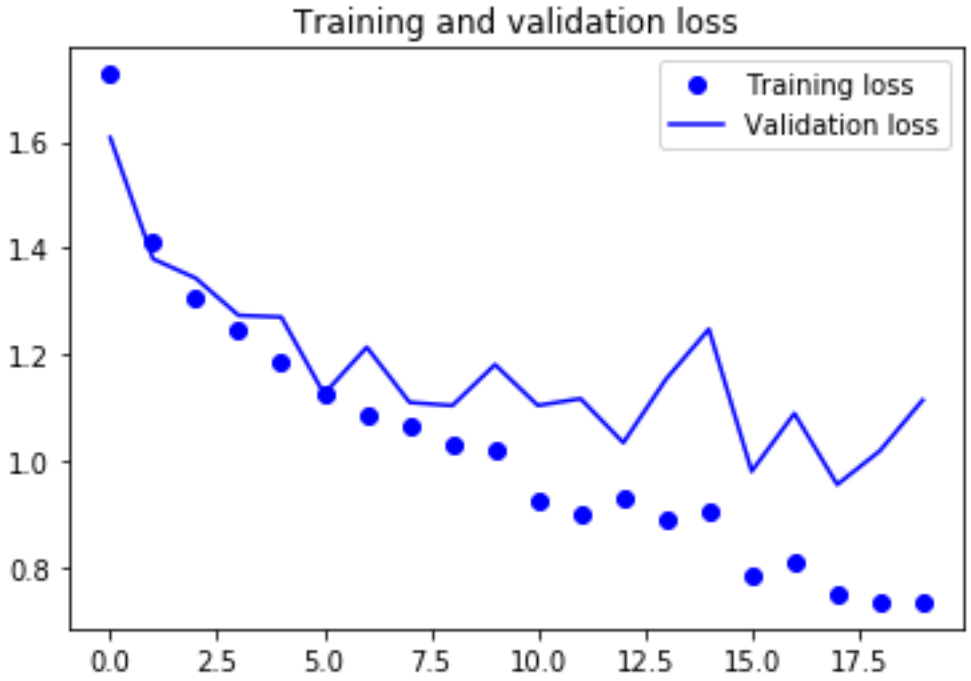
ImageDataGenerator를 이용하여 데이터를 전처리했다. Test, Train data를 나눠주고 Sequential을 이용하여 소형 합성망을 구축했다. 그리고 compile을 이용하여 optimizer='adam'를 써서 훈력을 위한 모델을 구성했다.

```
history=model.fit_generator(train_generator,epochs=20,steps_per_epoch=2276//32,validation_data=test_generator,validation_steps=251//32,callbacks=callbacks_list)
```

[CNN 모델 적합]

모델을 적합시키기 위해 구한 batch는 32며 보강이나 미세조정 없이 CNN 모델을 훈련하여 'garbage_1.h5'로 저장했다. 이때 accuracy는 0.73이다.





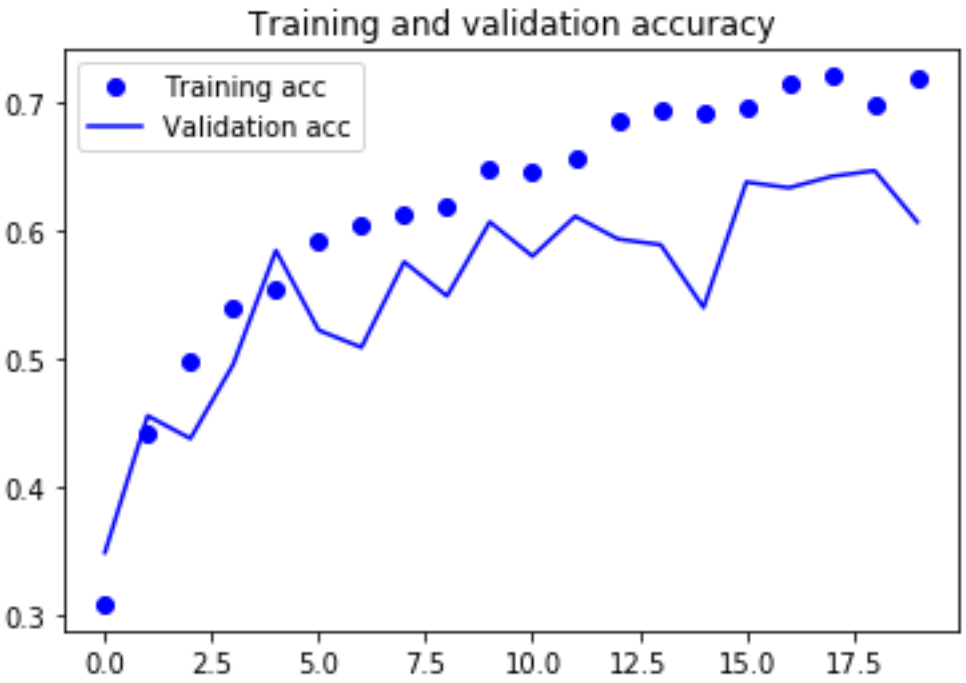
보강을 하지 않아도 위에 훈련중 손실과 정확도 그래프를 봤을 때 충분히 잘 적합되었으며 적합한 CNN모델이라고 할 수 있다.

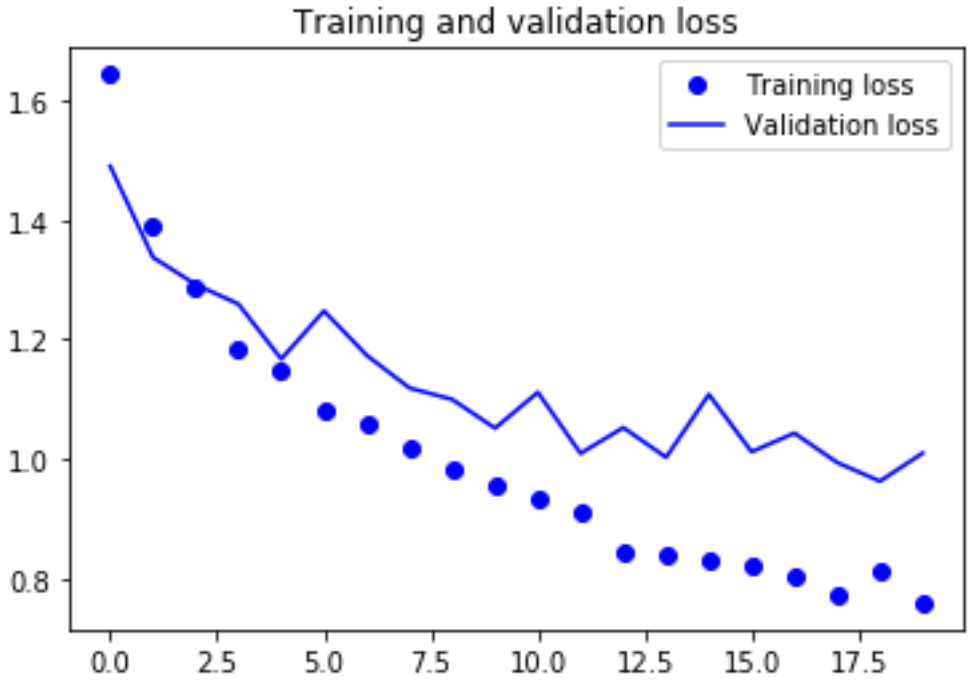
[데이터 보강]

```
datagen = ImageDataGenerator(
        rotation_range=40,
        width_shift_range=0.2,
        height_shift_range=0.2,
        shear_range=0.2,
        zoom_range=0.2,
        horizontal_flip=True,
       fill_mode='nearest')
```

을 추가하고 dropout을 이용하여 데이터를 조정하고 다시 돌려보았다.

새로 돌린 모델은 "garbage_2.h5"로 저장했다. Accuracy score는 0.75로 위에 보강없이 돌린 일반모형과 상당히 비슷하다.

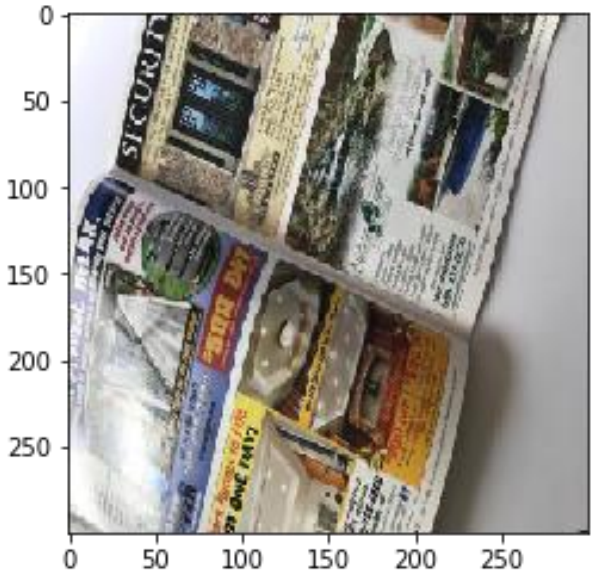




그러나 보강후에 적합시킨 CNN 모형이 조금 더 accuracy가 높으므로 최적모형으로 선택했다.

3) 합성망이 학습한 내용 시각화

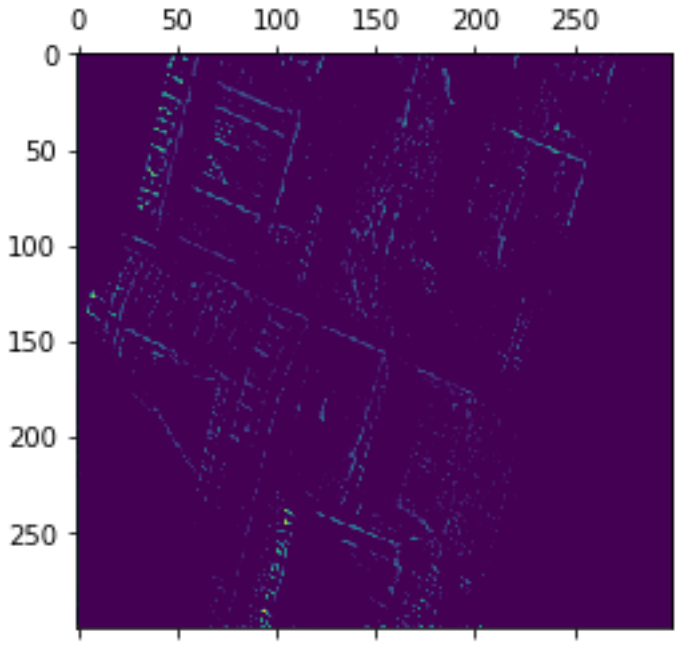
1. 예시로 test data에 해당하는 하나의 이미지 'paper380.jpg'를 입력했다.



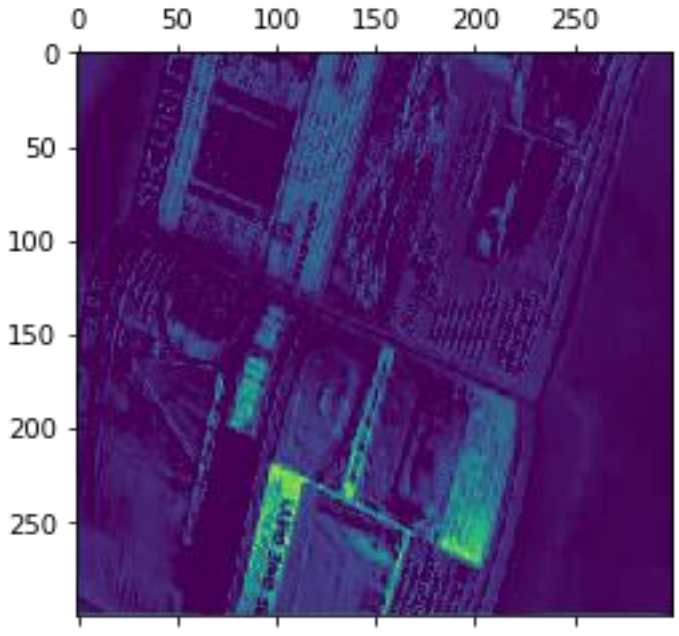
2. 상위 8개 계층의 출력을 추출하고 5개 배열의 리스트를 돌려줬다. 따라서 계층마다 한 개의 배열이 활성화 된다.

➔ 32채널, 300 x 300인 특징지도

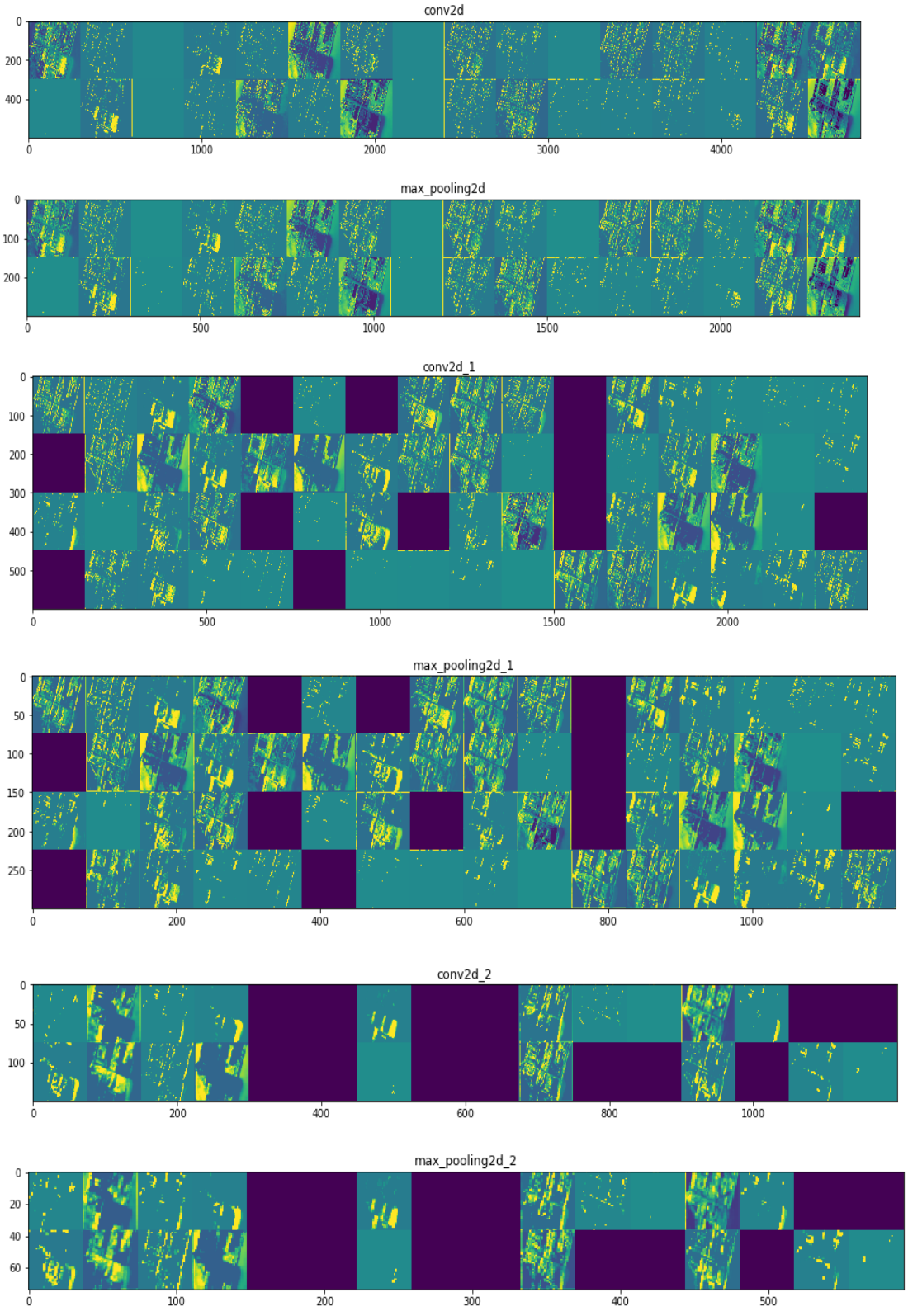
➔ 두번째 채널을 그려보았다.



➔ 15번째 채널 그리기

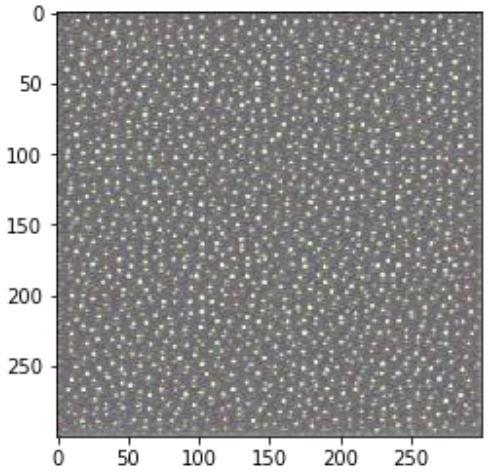


3. zip함수를 이용하여 특성 맵을 그리고 활성화 채널을 위한 그리드 크기를 구했다. 각 활성화를 하나의 큰 그리드에 채워보았다.



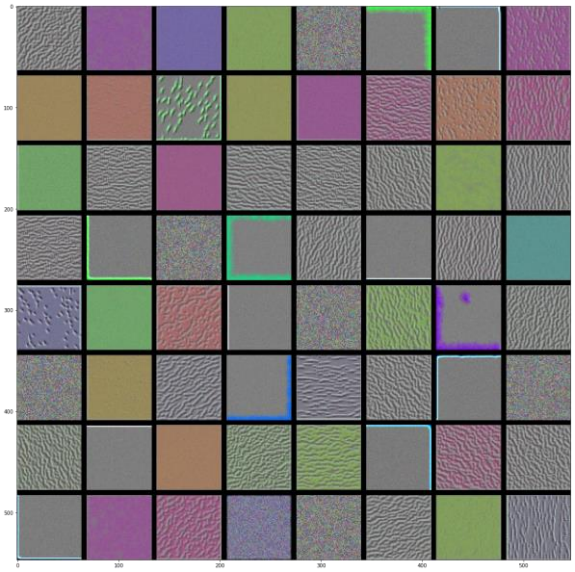
4. 필터 시각화

손실에 대한 입력 이미지의 그래디언트를 계산하고 그래디언트 정규화를 실시했다. 그리고 입력 이미지에 대한 손실과 그래디언트를 반환했다. 잡음이 석인 회색 이미지에 경사 상승법을 40단계 실행하여 필터를 아래와 같이 시각화했다.

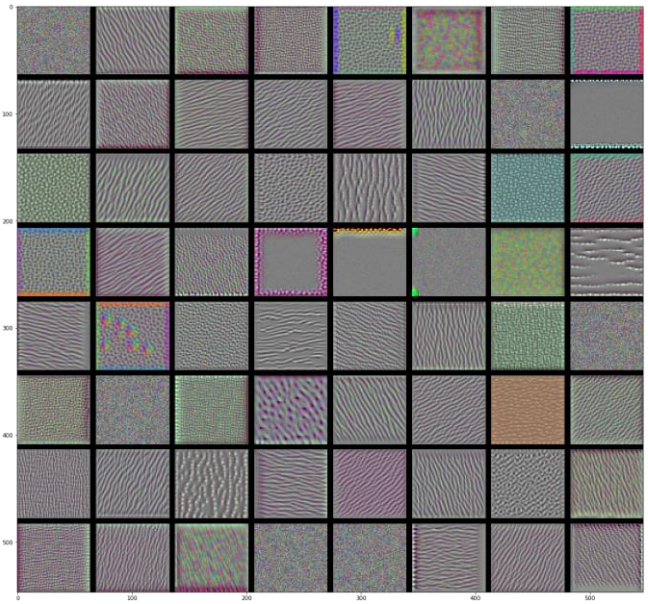


5. 계층내의 모든 필터 응답 패턴의 격자망 생성

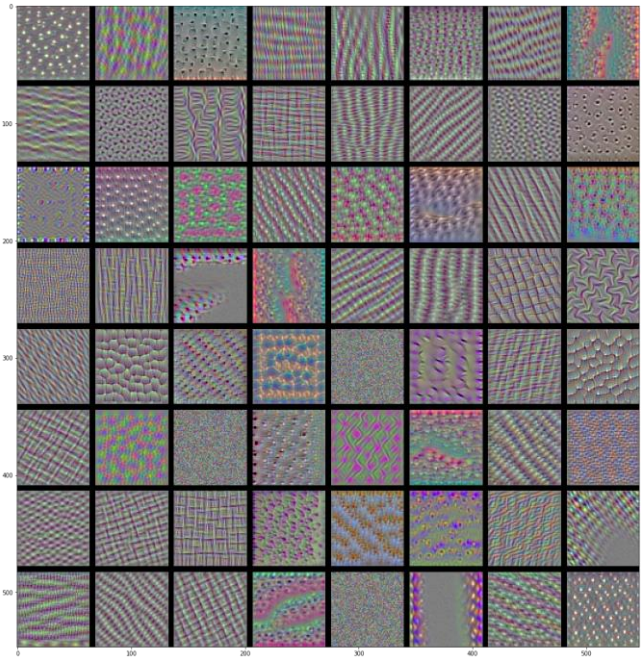
block1_conv1 계층의 필터패턴



block2_conv1 계층의 필터패턴



block3_conv1 계층의 필터패턴



block4_conv1 계층의 필터패턴