



# orange 활용 데이터 분석 및 머신 러닝



# 8차시

## 비정형데이터의 분류와 군집화 1



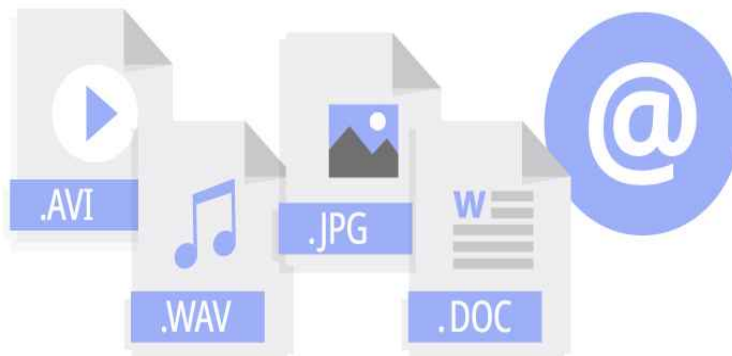
- 이미지 데이터를 활용한 분류모델 만들기
- 유사 이미지 찾기 - 이미지의 거리
- 이미지 데이터를 활용한 군집화 연습
- MNIST 데이터를 활용한 손글씨 인식

이미지 데이터를 활용한  
분류 문제 해결하기

# 정형 데이터와 비정형 데이터

ID	Name	AGE	SEX
01	KIM	32	M
02	LEE	26	F
03	PARK	72	F
04	CHOI	15	M

정형데이터



비정형데이터

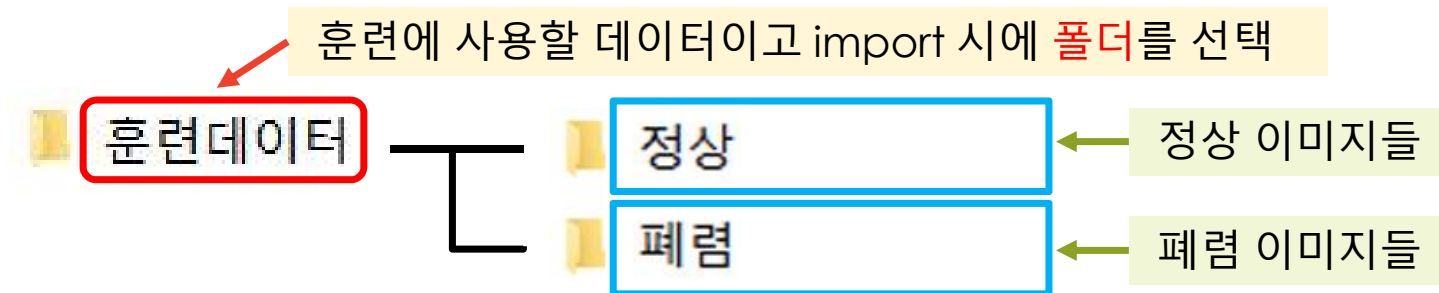


# 정형 데이터와 비정형 데이터

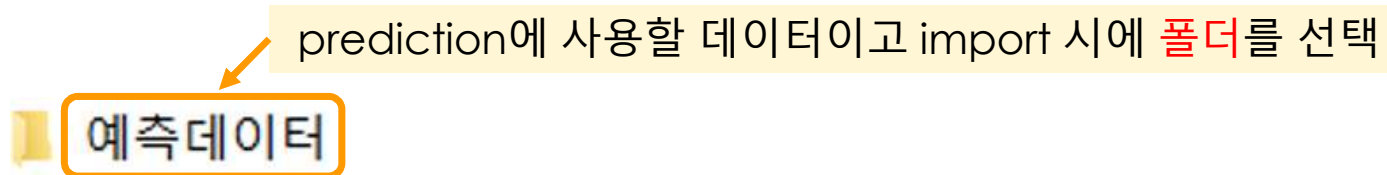


[출처] 빅 데이터: 산업 지각변동의 지원 (삼성경제연구소, 2012년 5월)

# 분류 작업 시 폴더 구성



이 이름이 바로 [레이블]  
폴더가 2개 있다는 것은 레이블이 2개이고,  
2개 중 하나로 분류한다는 의미



**군집화**의 경우, 하나의 폴더 안에 데이터를 모두 넣어두고  
import 시에 **폴더**를 선택함

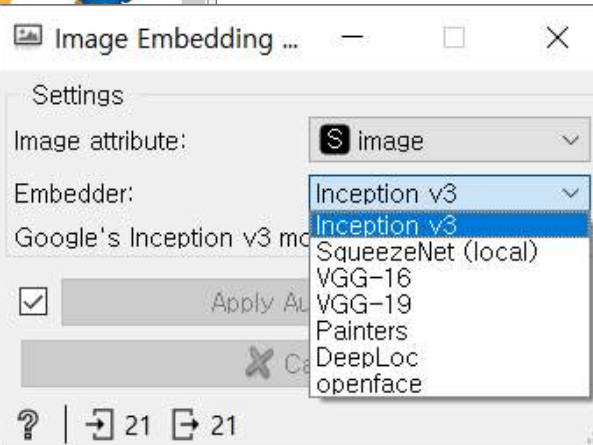
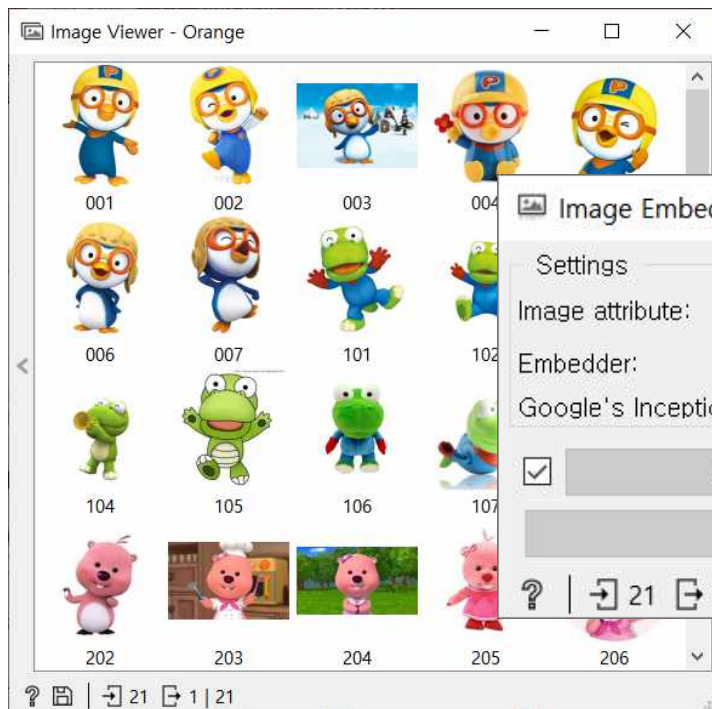
# 이미지 데이터 전처리



Import Images

Image Embedding

Data Table



← 원본 이미지

Data Table - Orange

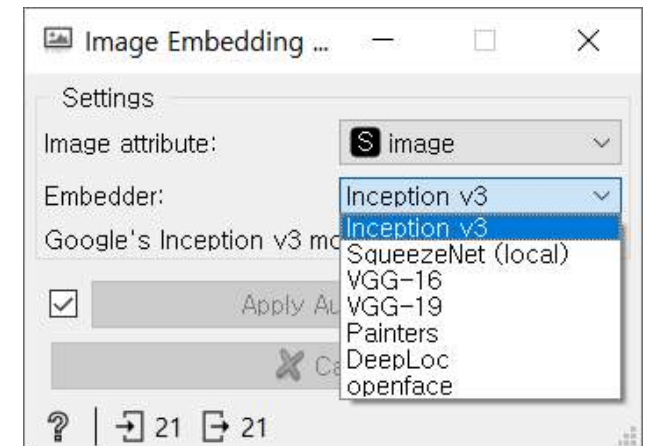
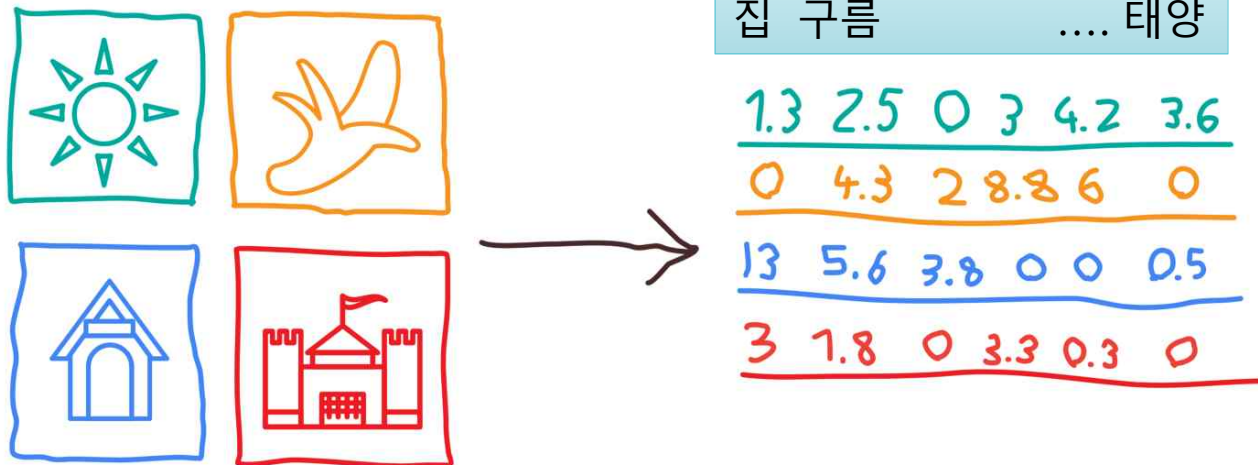
	n2043	n2044	n2045	n2046	n2047
1	1.00433	0.334118	0.754894	0.490184	0
2	0.0980886	0.146821	0.320586	0.395586	0.00522691
3	0.385838	0.508336	0.672387	0.20541	0.268841
4	0.336323	0.194329	1.06339	0.275789	0
5	0.119666	0.0154581	0.772878	0.895415	0
6	0.408689	0.530749	0.374655	0.248363	0
7	0.441067	0.263011	0.281285	0.730141	0.0231792
8	0.50626	0.314495	0.119101	0.051238	0
9	0.682405	0.0181672	0.931512	0.256288	0
10	0.439481	0.282428	0.93848	0.0459406	0.0250402
11	0.275164	0.0813116	0.620929	0.00449373	0
12	0.100278	0.071083	0.889523	1.11177	0.0428037
13	0.755859	0.029312	0.59449	0.0447758	0.0252815
14	1.05742	0.48215	0.564134	0.0149436	0.0202191
15	1.04125	0.0440536	0.423091	0.225932	0.114777
16	0.591044	0.211006	0.256806	0.0538434	0.0630299
17	0	0.436433	0.52646	0.226772	0.20744
18	0.246487	0.418108	0.0621264	0.474445	0.109877
19	0.850144	0.0487128	0.633016	0.217303	0
20	1.20983	0.00675485	0.620267	0.159912	0.198227
21	0.42726	0.650934	0.565784	0.148118	0.0417409

21 21 21

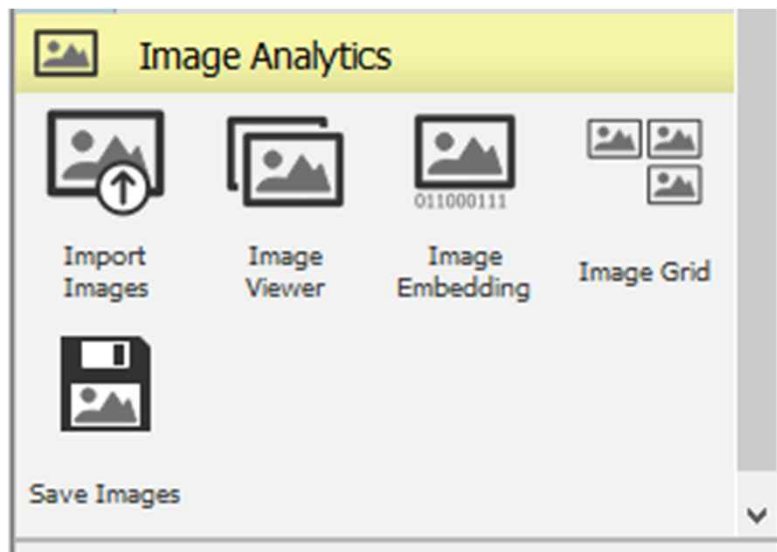


# 이미지 임베딩

- 임베더는 각각의 이미지를 숫자화 하는 과정으로 이미 학습된 CNN (합성곱 신경망 convolution neural network)을 활용하여 이미지의 특성을 숫자화 한다. (전이학습)
- 숫자들은 모두 이미지의 특성을 담고 있다.



# 추가 설치(Add-Ons)된 위젯 확인



**Import Images** : 이미지 가져오기

**Import Viewer** : 이미지 보기

**Image Embedding** : 이미지 Embedding

**Image Grid** : 이미지 Grid

**Save Images** : Image 이미지 저장

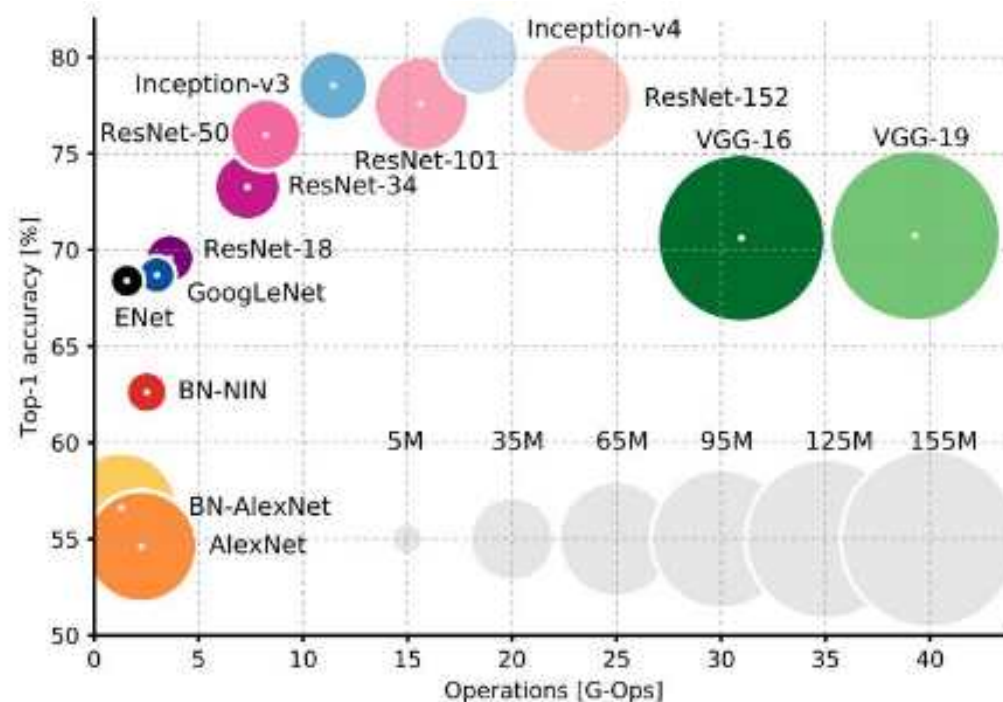
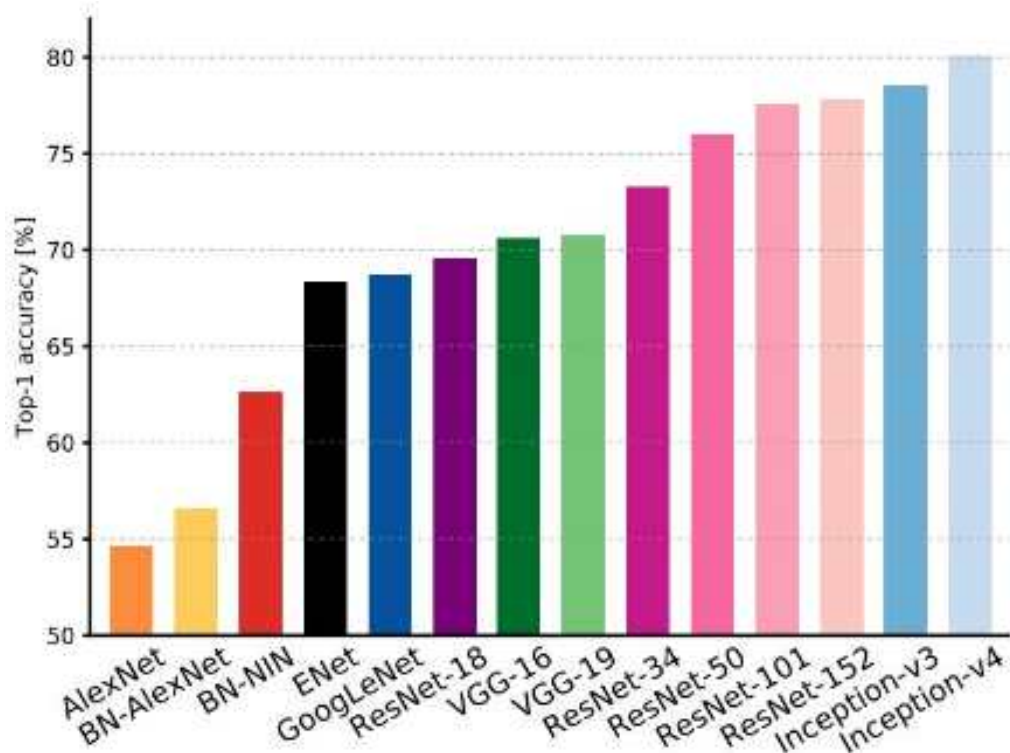
- **Image Embedder 종류**

1. **Inception V3**: 구글의 이미지 인식을 위한 심층 신경망으로 ImageNet에서 학습된 모델
2. **SqueezeNet(local)** : ImageNet에서 학습된 모델로 제한된 리소스 환경에서 활용 가능하도록 만든 모델. 인터넷이 연결되어 있지 않은 경우 사용
3. **VGG-16/VGG-19** : 옥스포드 대학의 비전 이미지 분류 모델. 레이어의 개수에 따라 VGG-16과 VGG-19로 나뉘어짐.
4. **Painters** : 1584명의 화가가 그린 79433개의 이미지를 통해 훈련된 모델로, 화가의 화풍 분류에 활용
5. **DeepLoc** : 효모 세포 이미지를 분석하도록 훈련된 모델
6. **Openface** : 얼굴 인식에 사용 가능한 모델

<https://www.image-net.org/index.php>

<https://www.kaggle.com/competitions/imagenet-object-localization-challenge/overview>

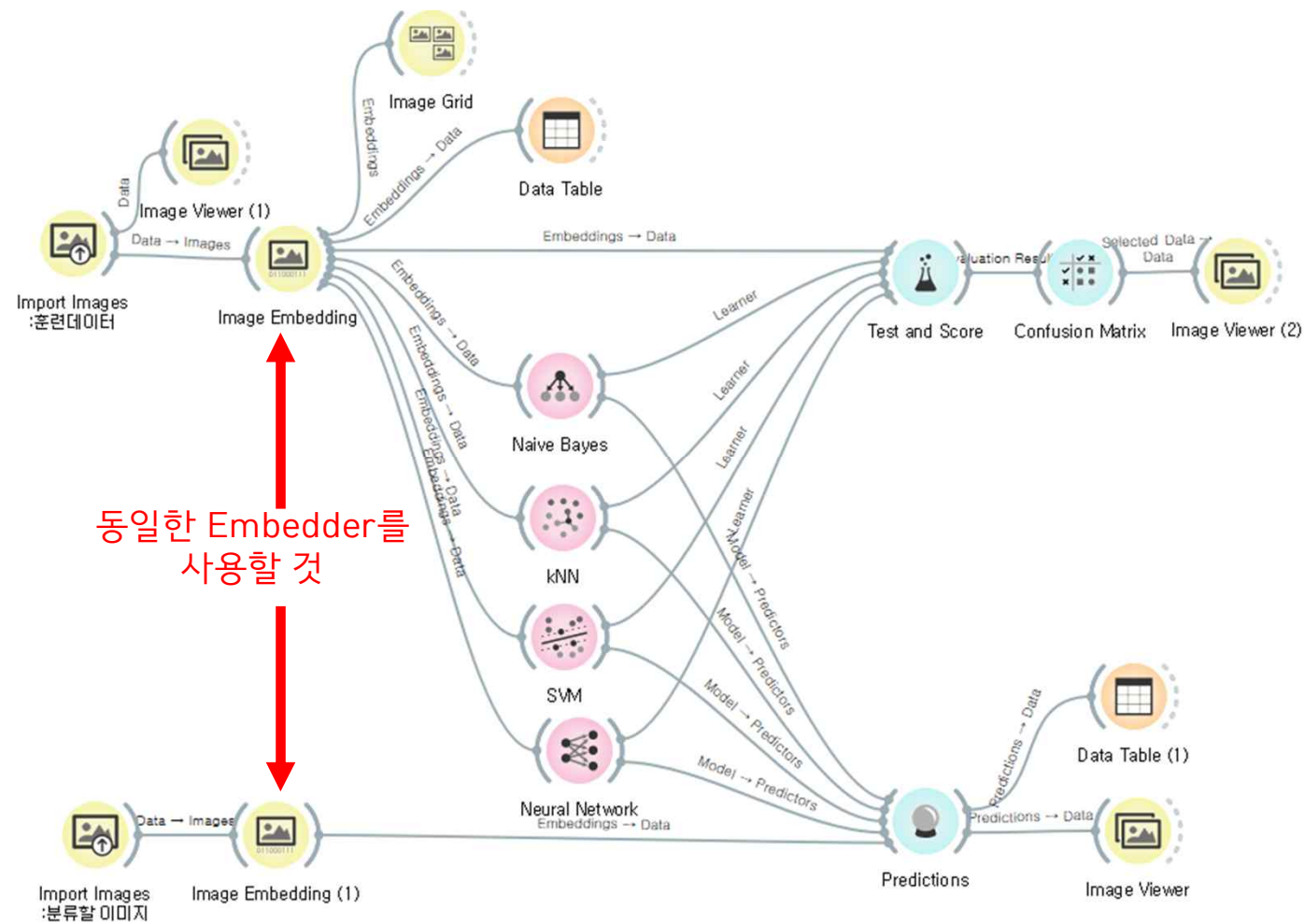
# 정확도-연산량 Trade-off



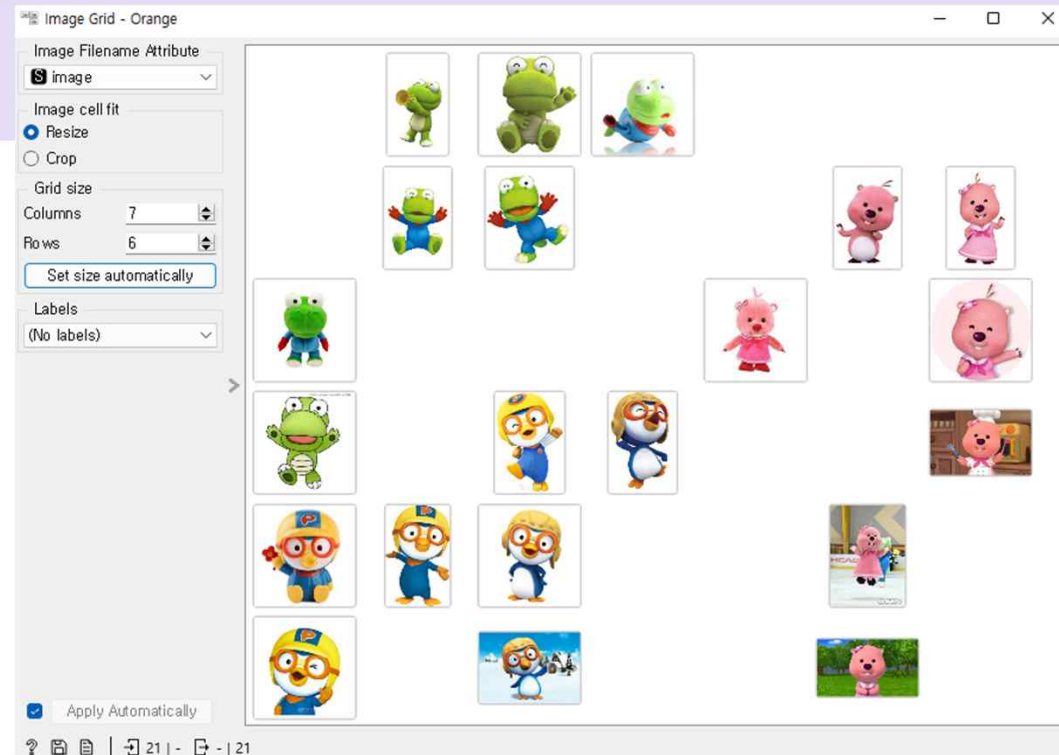
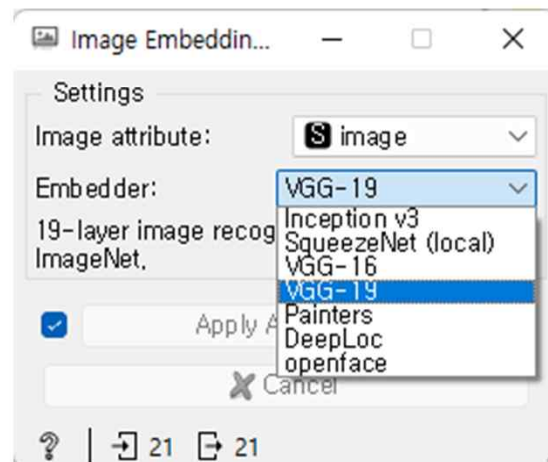
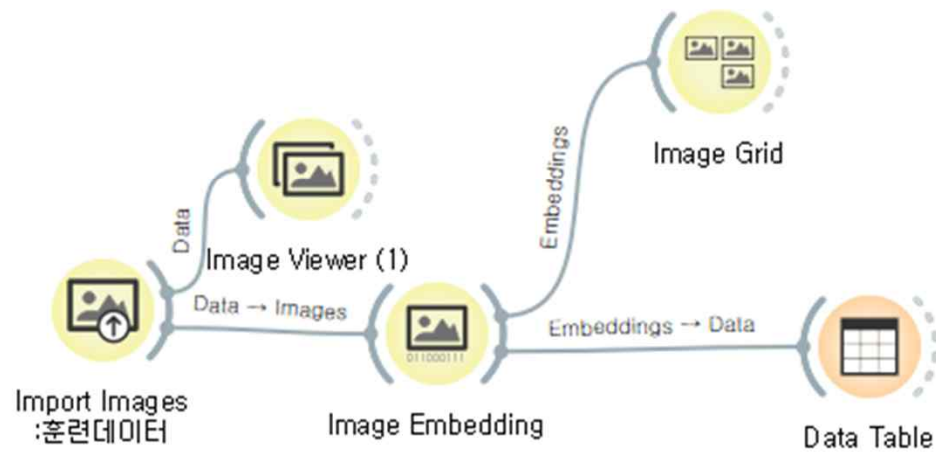
An Analysis of DNN Model for Practical Applications (2017)

출처: <https://medium.com/analytics-vidhya/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5>

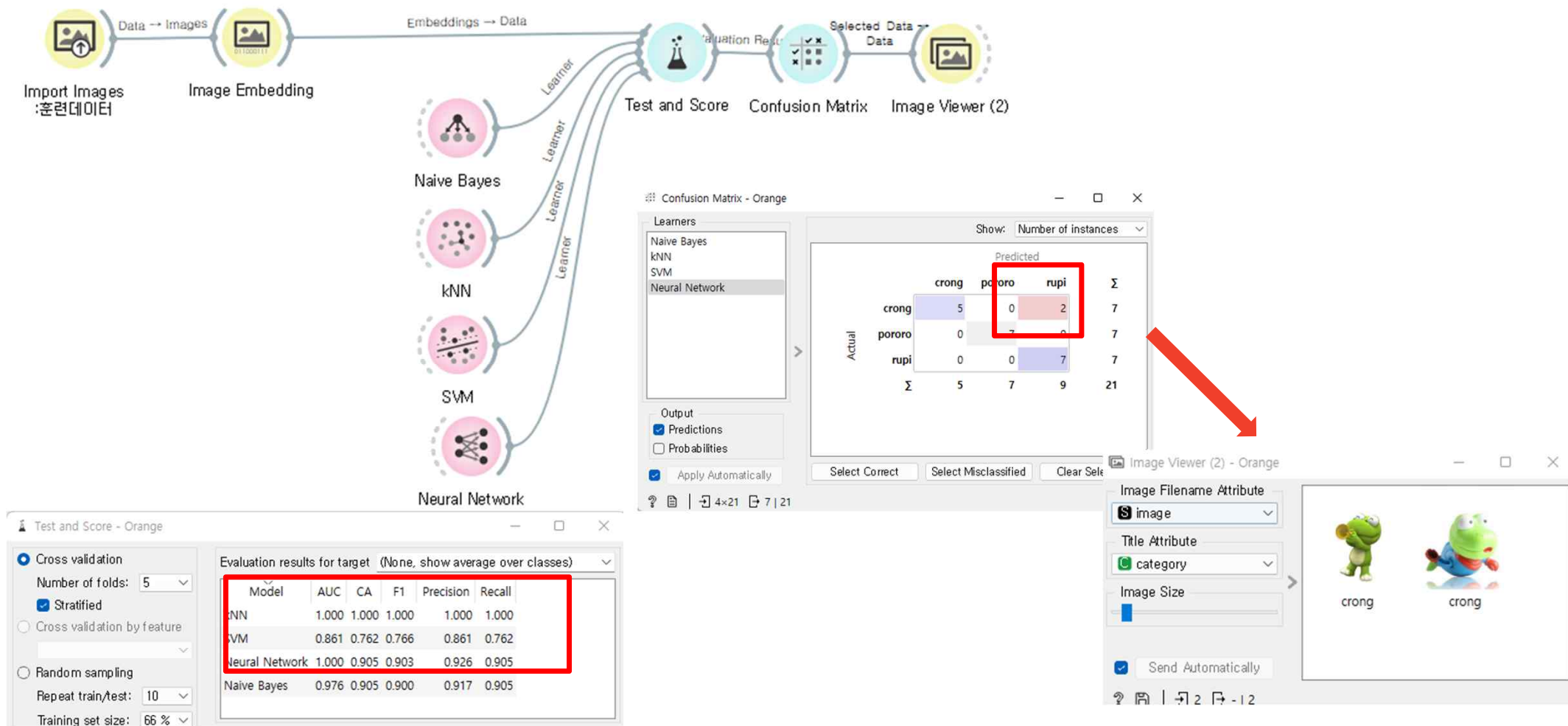
# 뽀로로와 그 친구들을 분류해봅시다.



# 이미지 탐색과 임베딩



# 성능평가와 혼동행렬 - 분류가 잘 되고 있나요?



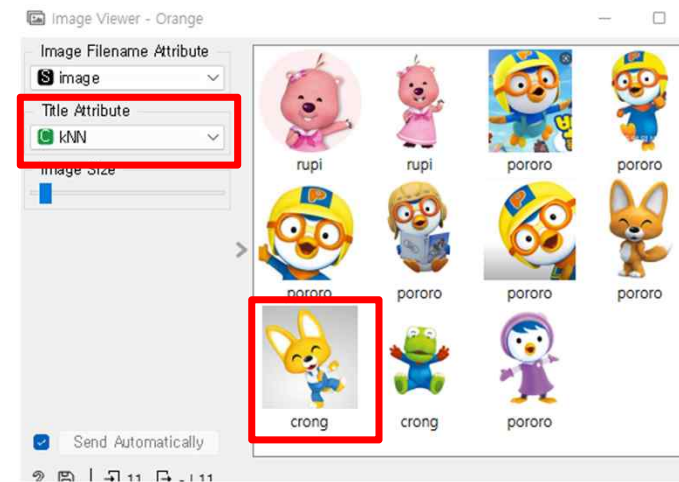
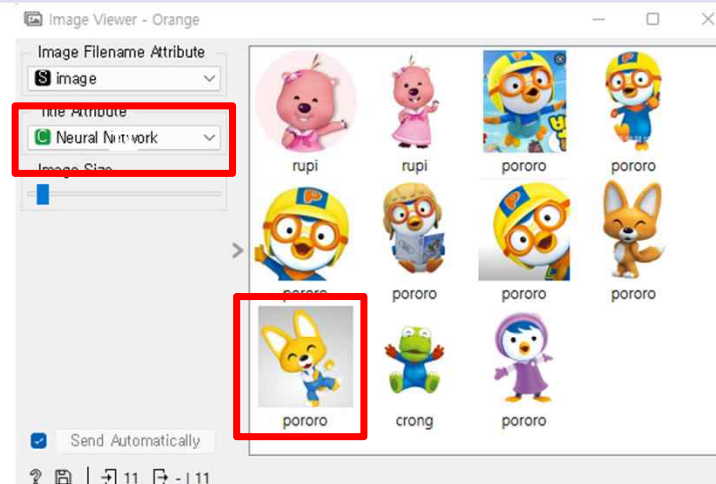
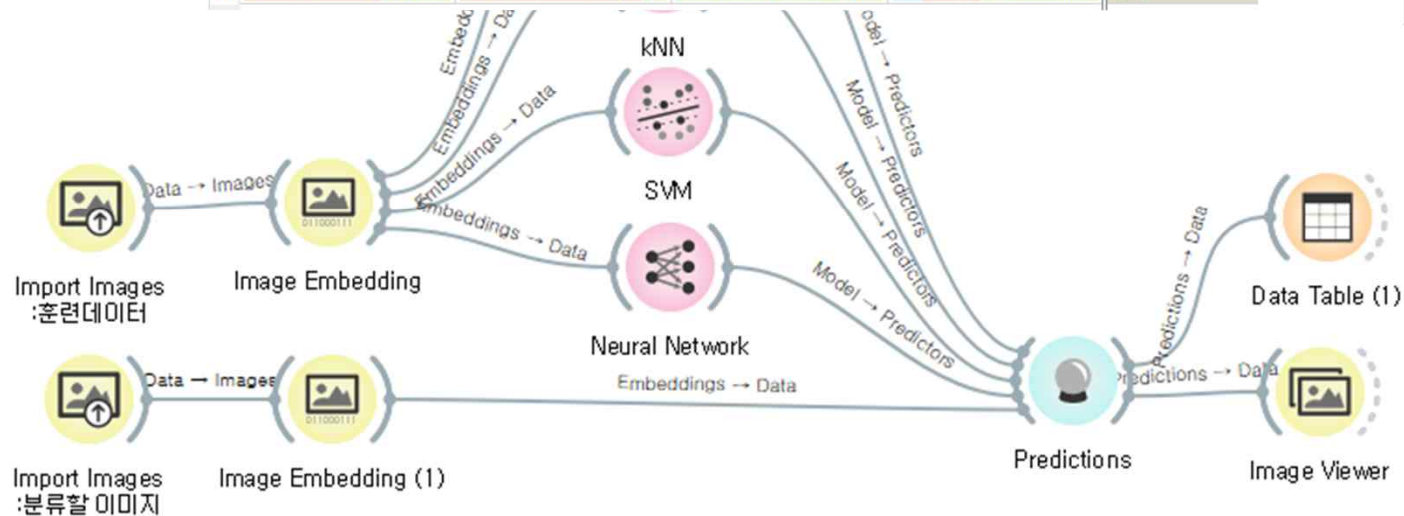


# 이미지 분류 예측

Predictions - Orange

Show probabilities for: Classes known to the model

	kNN	Neural Network	Naive Bayes	SVM	image name
1	0.00 : 0.00 : 1.00 → rupi	0.00 : 0.00 : 1.00 → rupi	0.00 : 0.00 : nan → rupi	0.01 : 0.02 : 0.97 → rupi	루피
2	0.00 : 0.00 : 1.00 → rupi	0.00 : 0.00 : 1.00 → rupi	0.00 : 0.00 : 1.00 → rupi	0.01 : 0.02 : 0.97 → rupi	루피2
3	0.00 : 1.00 : 0.00 → por...	0.00 : 1.00 : 0.00 → por...	0.00 : 1.00 : 0.00 → por...	0.11 : 0.67 : 0.22 → por...	뽀로로
4	0.00 : 1.00 : 0.00 → por...	0.00 : 1.00 : 0.00 → por...	0.00 : 1.00 : 0.00 → por...	0.04 : 0.89 : 0.07 → por...	뽀로로1
5	0.00 : 1.00 : 0.00 → por...	0.00 : 1.00 : 0.00 → por...	0.00 : 1.00 : 0.00 → por...	0.12 : 0.55 : 0.33 → por...	뽀로로2
6	0.00 : 1.00 : 0.00 → por...	0.00 : 1.00 : 0.00 → por...	0.00 : 1.00 : 0.00 → por...	0.15 : 0.15 : 0.71 → rupi	뽀로로3
7	0.00 : 1.00 : 0.00 → por...	0.00 : 1.00 : 0.00 → por...	0.00 : 1.00 : 0.00 → por...	0.06 : 0.88 : 0.06 → por...	뽀로로4
8	0.00 : 1.00 : 0.00 → por...	0.00 : 0.99 : 0.01 → por...	0.00 : 1.00 : 0.00 → por...	0.15 : 0.19 : 0.66 → rupi	에디
9	0.67 : 0.33 : 0.00 → crong	0.35 : 0.65 : 0.00 → por...	0.96 : 0.04 : 0.00 → crong	0.35 : 0.47 : 0.18 → por...	에디2
10	1.00 : 0.00 : 0.00 → crong	1.00 : 0.00 : 0.00 → crong	1.00 : 0.00 : 0.00 → crong	0.98 : 0.01 : 0.01 → crong	크롱
11	0.00 : 0.67 : 0.33 → por...	0.00 : 0.88 : 0.12 → por...	0.00 : 0.00 : 1.00 → rupi	0.14 : 0.29 : 0.57 → rupi	패티



- 이미지 유사성과 이미지 군집화
- 그림데이터의 분류 : 마네와 모네의 작품의 특성을 이용해 작가 예측하기





# orange 활용 데이터 분석 및 머신 러닝



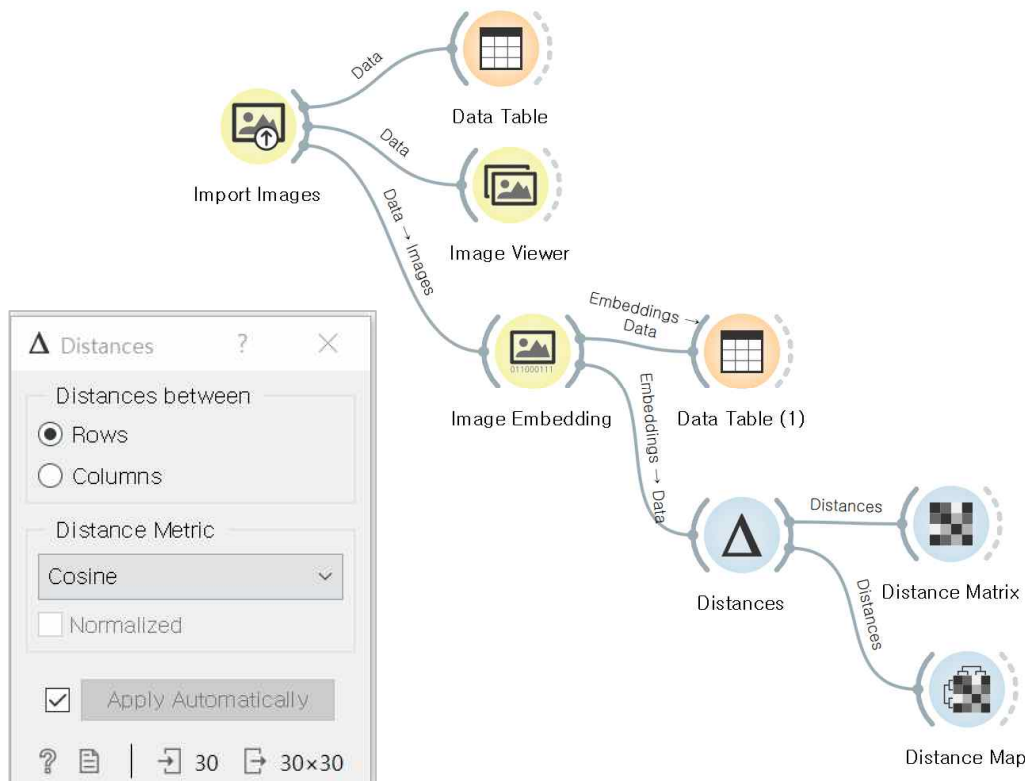
# 9차시

## 비정형데이터의 분류와 군집화 2

# 이미지 데이터의 군집화

# 이미지 거리

- Distances, Distance Matrix, Distance Map
- 두쌍으로 묶어서 거리를 구하여 행렬로 나타냄

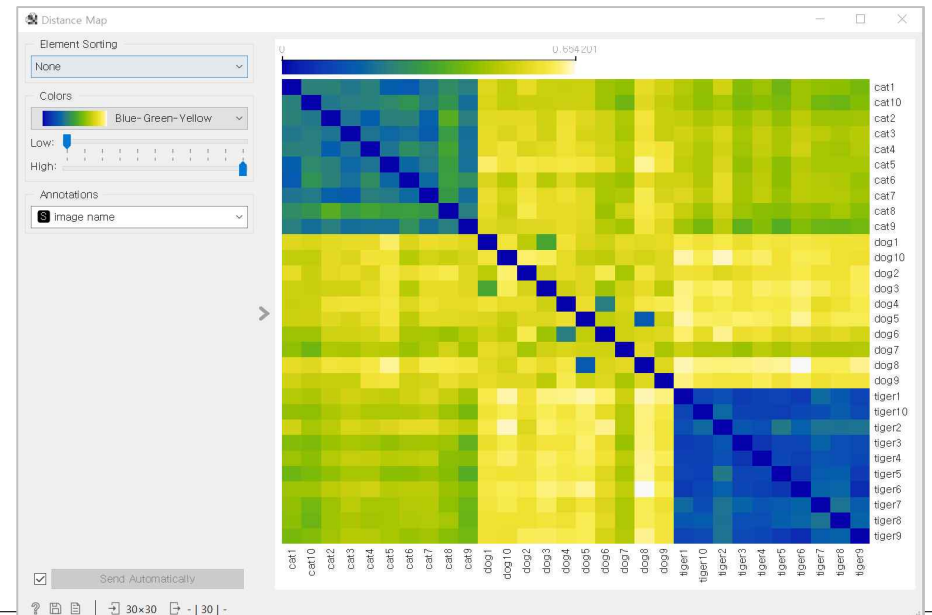


Distance Matrix

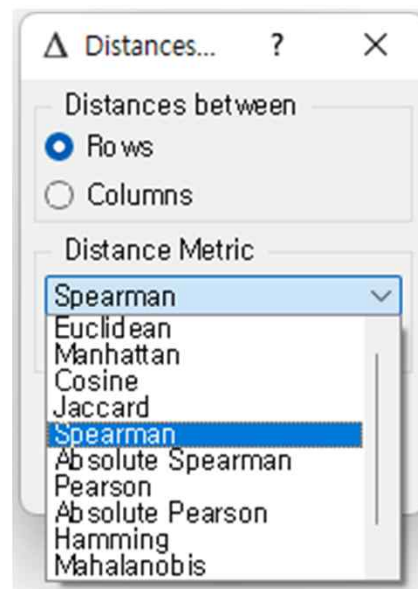
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
1		0.244	0.246	0.222	0.241	0.168	0.155	0.215	0.270	0.239	0.538	0.496	0.550	0.509	0.507	0.511	0.455	0.436	0.553	0.520	0.480	0.442	0.521	0.424	0.451	0.395	0.451
2	0.244		0.224	0.236	0.243	0.262	0.284	0.229	0.294	0.202	0.530	0.497	0.508	0.508	0.510	0.512	0.453	0.395	0.543	0.499	0.469	0.437	0.462	0.414	0.430	0.412	0.453
3	0.246	0.224		0.213	0.164	0.241	0.246	0.173	0.365	0.240	0.546	0.549	0.548	0.518	0.568	0.542	0.515	0.472	0.580	0.508	0.498	0.468	0.491	0.448	0.478	0.439	0.488
4	0.222	0.236	0.213		0.216	0.188	0.203	0.163	0.309	0.201	0.553	0.534	0.558	0.516	0.575	0.529	0.520	0.465	0.574	0.502	0.520	0.489	0.530	0.464	0.502	0.466	0.504
5	0.241	0.243	0.164	0.216		0.212	0.273	0.238	0.334	0.205	0.555	0.579	0.532	0.545	0.572	0.551	0.522	0.488	0.591	0.577	0.508	0.471	0.518	0.457	0.501	0.466	0.515
6	0.168	0.262	0.241	0.188	0.212		0.186	0.228	0.316	0.203	0.613	0.576	0.592	0.582	0.589	0.587	0.557	0.484	0.629	0.585	0.494	0.471	0.534	0.441	0.484	0.431	0.484
7	0.155	0.284	0.246	0.203	0.273	0.186		0.190	0.319	0.281	0.533	0.485	0.563	0.484	0.536	0.492	0.469	0.449	0.557	0.516	0.520	0.475	0.552	0.457	0.489	0.434	0.486
8	0.215	0.229	0.173	0.163	0.238	0.228	0.190		0.302	0.200	0.561	0.523	0.584	0.551	0.567	0.546	0.463	0.473	0.600	0.524	0.535	0.492	0.553	0.459	0.496	0.446	0.489
9	0.270	0.294	0.365	0.309	0.334	0.316	0.319	0.302		0.238	0.554	0.506	0.596	0.558	0.552	0.545	0.448	0.521	0.595	0.562	0.506	0.451	0.518	0.450	0.431	0.423	0.448
10	0.239	0.202	0.240	0.201	0.205	0.203	0.281	0.200	0.238		0.538	0.525	0.544	0.542	0.556	0.542	0.485	0.459	0.587	0.544	0.456	0.408	0.492	0.376	0.423	0.363	0.417
11	0.538	0.530	0.546	0.553	0.555	0.613	0.533	0.561	0.554	0.538		0.582	0.487	0.340	0.576	0.533	0.514	0.553	0.545	0.552	0.593	0.576	0.594	0.566	0.589	0.580	0.592
12	0.496	0.497	0.549	0.534	0.579	0.576	0.485	0.523	0.506	0.525	0.582		0.616	0.605	0.552	0.505	0.493	0.539	0.522	0.540	0.636	0.601	0.644	0.596	0.611	0.575	0.602
13	0.550	0.508	0.548	0.558	0.532	0.592	0.563	0.584	0.596	0.544	0.487	0.616		0.525	0.553	0.540	0.600	0.461	0.583	0.543	0.578	0.568	0.535	0.558	0.577	0.580	0.594
14	0.509	0.508	0.518	0.516	0.545	0.582	0.484	0.551	0.558	0.542	0.340	0.605	0.525		0.511	0.540	0.451	0.517	0.559	0.491	0.623	0.604	0.614	0.586	0.613	0.604	0.621
15	0.507	0.510	0.568	0.575	0.572	0.589	0.536	0.567	0.552	0.556	0.576	0.552	0.553	0.511		0.567	0.238	0.531	0.589	0.600	0.625	0.585	0.598	0.581	0.593	0.591	0.611
16	0.511	0.512	0.542	0.529	0.551	0.587	0.492	0.546	0.545	0.542	0.533	0.505	0.540	0.540	0.567		0.497	0.519	0.156	0.514	0.635	0.609	0.607	0.614	0.610	0.597	0.629
17	0.455	0.453	0.515	0.520	0.522	0.557	0.469	0.463	0.448	0.485	0.514	0.493	0.600	0.451	0.238	0.497		0.516	0.531	0.536	0.614	0.552	0.627	0.567	0.575	0.538	0.558
18	0.436	0.395	0.472	0.465	0.488	0.484	0.449	0.473	0.521	0.459	0.553	0.539	0.461	0.517	0.531	0.519	0.516		0.547	0.467	0.512	0.480	0.478	0.448	0.488	0.466	0.496
19	0.553	0.543	0.580	0.574	0.591	0.629	0.557	0.600	0.595	0.587	0.545	0.522	0.583	0.559	0.589	0.156	0.531	0.547		0.537	0.638	0.623	0.620	0.623	0.628	0.629	0.654
20	0.520	0.499	0.508	0.502	0.577	0.585	0.516	0.524	0.562	0.544	0.552	0.540	0.543	0.491	0.600	0.514	0.536	0.467	0.537		0.621	0.592	0.581	0.585	0.604	0.578	0.603
21	0.480	0.469	0.498	0.520	0.508	0.494	0.520	0.535	0.506	0.456	0.593	0.636	0.578	0.623	0.625	0.635	0.614	0.512	0.638	0.621		0.111	0.131	0.092	0.104	0.111	0.091
22	0.442	0.437	0.468	0.489	0.471	0.471	0.475	0.492	0.451	0.408	0.576	0.601	0.568	0.604	0.585	0.609	0.552	0.480	0.623	0.592	0.111		0.187	0.105	0.107	0.098	0.113
23	0.521	0.462	0.491	0.520	0.518	0.524	0.552	0.552	0.518	0.492	0.584	0.624	0.535	0.614	0.598	0.602	0.622	0.478	0.620	0.581	0.131	0.187		0.148	0.123	0.126	0.168

Labels: Enumerate

Send Automatically



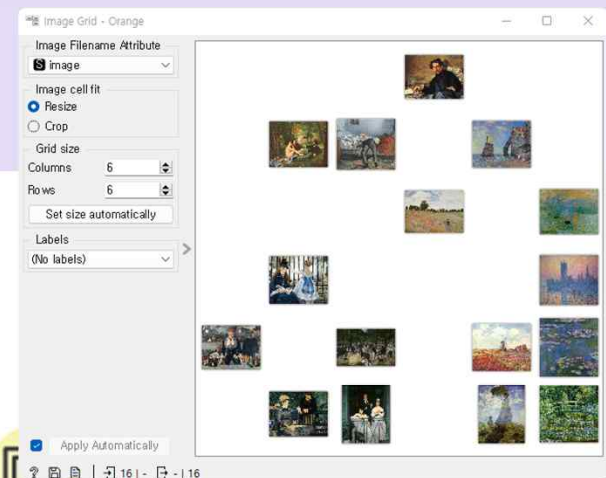
## 군집화의 결과와 분류의 결과 비교하기



# 누구의 작품일까요?

- 마네 모네의 작품의 화풍 분석을 통한 작가 예측하기

# 이미지 분류 워크플로를 만들어 봅시다.



Predictions - Orange

Show probabilities for Classes known to the model

	Naive Bayes	Neural Network	kNN	image name
1	<u>1.00 : 0.00 → Manet</u>	<u>0.09 : 0.91 → Monet</u>	<u>0.50 : 0.50 → Manet</u>	manet-Argenteuil
2	<u>1.00 : 0.00 → Manet</u>	<u>1.00 : 0.00 → Manet</u>	<u>0.75 : 0.25 → Manet</u>	manet-Cafe Concert
3	<u>1.00 : 0.00 → Manet</u>	<u>0.99 : 0.01 → Manet</u>	<u>0.25 : 0.75 → Monet</u>	monet-Camille Monet on a Bench
4	<u>0.00 : 1.00 → Monet</u>	<u>0.00 : 1.00 → Monet</u>	<u>0.00 : 1.00 → Monet</u>	monet-Woman with Umbrella



다음 시간에는 **비정형데이터 중**  
**텍스트** 데이터를 다루어 보도록 하  
겠습니다.