[24PAGE]

위에서 다양한 분류기들의 분류 성능을 평가한 결과, 정확도 면에서 성능이 가장 좋았던 분류기 5개를 선정하였습니다. 선정된 분류기는 KNN분류기, MLP분류기, SVC분류기, EXTRATREE분류기, RF분류기로, 저희는 해당 독립 모델들을 VotingClassifier에 넣어 보다 뛰어난 분류 성능을 구현하고자 하였습니다. 먼저 각 독립 모델별로 최적의 하이퍼 파라미터를 구하기 위해서 GridSearch, RandomizedSearch 등을 활용하였고, 그 결과 모든 독립 모델 으로부터 0.94이상의 분류 성능을 확인하였습니다. 학습 시간 면에서도 SVC 모델을 제외하면 모두 2min 안에 학습이 완료되면서 굉장히 빠른 시간 안에 학습이 완료됨을 알 수 있었습니다.

[25PAGE]

위에서 언급 드린 5개의 독립 모델을 활용하여 Voting 분류기을 만들었습니다. 학습 시간은 약 7분 여가 소요되었고, 하드 보팅 결과 0.955, 소프트 보팅 결과 0.9645의 정확도를 나타냈습니다. Soft voting 분류기를 사용했을 때 독립 모델을 썼을 때 보다 약 2%에 성능 향상을 보였는데, 이는 약 5.5% 오차를 3.5%로 낮춘 것이기 때문에 정확도가 이미 1로 거의 수렴한 상황에서 2%의 오차 개선은 유의미한 성능 개선이라고 볼 수 있겠습니다.

[26PAGE]

본 Summary 내용은 저희가 구현한 분류기의 주요 성능 지표 결과를 나타낸 것입니다. 보시면, 거의 모든 주요 성능에서 우수한 결과를 나타냄을 알 수 있습니다.

또한, 본 Learning curve에 경우 Soft Voting 분류기에서 교차 검증을 3으로 지정해서 나타낸 것으로 학습 데이터가 늘수록 Validation score가 증가함을 확인할 수 있습니다.

[27PAGE]

이렇게 해서 저희가 구현한 모델을 ‘my\_model.pkl’이라는 이름으로 저장하여 추후에 새로운 test data를 적용할 때 쉽게 불러올 수 있도록 하였고, 이것을 초기에 미리 정한 test data에 넣어 최종 정확도를 구한 결과, 0.9484의 높은 예측 성능을 보였습니다.

[28PAGE]

그럼 이것으로 저희가 구현한 MNIST 분류 모델에 대한 설명을 모두 마치도록 하겠고, 이어서 발표에 대한 Q&A 진행하도록 하겠습니다.