모델링 – 랜덤포레스트 Script

P-SAT 구동현

1. (정의) 랜덤포레스트란 bagging(bootstrap+aggregating) 방법론의 일종으로, 실제 데이터 분석에 있어서 training data로부터 여러 개의 bag을 형성(bootstrap)해 tree를 병렬적으로 나열한 후 이를 모아(aggregate) 다수결 내지 평균 등의 방법으로 예측치를 구해주는 모형입니다. 랜포는 앙상블(ensemble) 방법의 대표적인 예입니다. (앙상블: weak model을 모두 모아 예측에 활용하여 하나의 strong model을 만드는 방법)
2. (예측 방식) 예측하고자 하는 변수가 범주형이면 다수결 방법에 따르게 되고, 연속형이면 회귀 트리를 주로 사용하여 모든 트리의 예측값의 평균을 구해줍니다.
3. (장점) 여타 배깅 방법론들과 랜덤포레스트의 결정적인 차이점은 parameter인 mtry에 있습니다. mtry를 정해주면 랜포모델은 자동으로 node별로 분류에 활용할 독립변수를 전체 독립변수들 중에서 랜덤으로 뽑아줍니다. 여기서 mtry라 함은 곧 각 트리들이 node들로 분류를 함에 있어서 분류의 기준이 되는 독립변수들을 트리가 계속해서 랜덤으로 정해주며 트리 간의 상관성을 낮춰줄 때, 그 독립변수의 '개수'를 의미하는 모수입니다. mtry를 달리 하면서 랜덤포레스트를 돌려주면 예측의 정확성이 mtry별로 조금씩 달라지게 되고, 이때 최적의 예측력을 가진 mtry를 최종 모수로 확정하면 되는 것입니다. (여타 모수들도 이와 같은 원리로 튜닝을 해주면 됩니다.) 랜덤포레스트는 배깅 방법을 따르므로 여러 트리들을 모아 평균을 내게 되어 분산이 적어지게 되고, 이에 따라 좋은 예측력을 가지게 됩니다.
4. (단점) 배깅도 결국은 트리들을 형성하는 예측모델이므로, 트리들이 training data에 과적합(overfit)된다면 training data에서는 좋은 성능을 보여줄지 몰라도 이를 test data로까지 일반화시킬 수는 없게 됩니다(트레이닝/테스트 MSE 비교 그래프 참조). 이때 maxdepth 등의 모수를 튜닝해주어 트리의 깊이를 적절히 낮춘다면 이러한 단점을 보완해줄 수 있습니다.
5. (OOB) K-fold 교차검증과 더불어 랜덤포레스트의 성능을 올려줄 수 있는 방법으로 Out Of Bag 방식이 있습니다. 트리의 개수가 많지 않을 때는 오히려 K-fold만을 쓰는 것보다 OOB를 써주는 것이 더 좋은 성능을 낼 정도라고 알려져 있습니다. OOB라고 불리는 이유는, 각 트리가 생성될 때 트리형성에 기여하지 않은 obs들이 해당 bag에 들어가지 않았기 때문입니다. 이렇게 예측을 하게 되면 k-fold와 같이 굳이 train/test split을 거치지 않아도 됩니다.
6. (Drugstore 데이터에 적용) 랜덤포레스트 모델을 만든 후, test data를 돌려 RMSPE 값을 구해줍니다. 이때 RMSPE의 도출식의 특성상 yi(실제 값)이 0인 경우는 구할 수가 없게 되므로, 0을 0.001로 살짝 바꿔주어 일단 값이 나오게 해주는 방법이 있습니다. 그러나 drugstore 데이터의 특성상 trainset에서 Sales가 0인 obs들은 모두 open 변수가 0인 관측치들이므로, 만약 이를 포함하여 랜포모델을 형성할 경우 test data를 넣고 돌리면 Sales 변수가 범주형이 아닌 연속형이라서 open을 하지 않은 obs임에도 불구하고 0을 넘는 값이 나오게 됩니다. 영업을 하지 않은 날의 약국이 그 날에 매출을 올린다는 것은 말이 되지 않는다고 보아도 과언이 아닙니다. 따라서 Open이 0인 test data는 그냥 sales도 0으로 예측하면 되지 않을까 싶습니다.

결론) 이렇게 돌린 랜포 모델의 RMSPE를 구하고, 이를 XGBOOST나 SARIMA의 RMSPE 값과 비교해 최적의 모델을 탐색합니다.