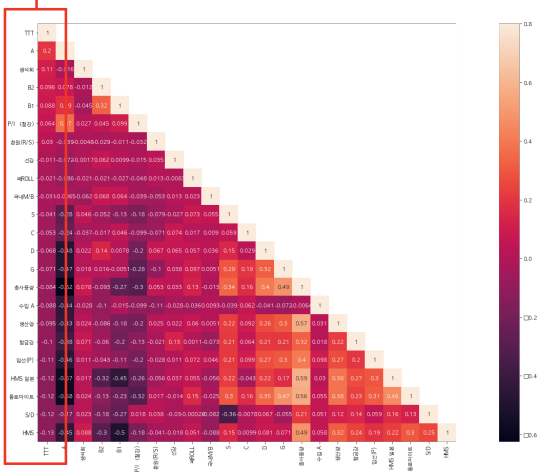
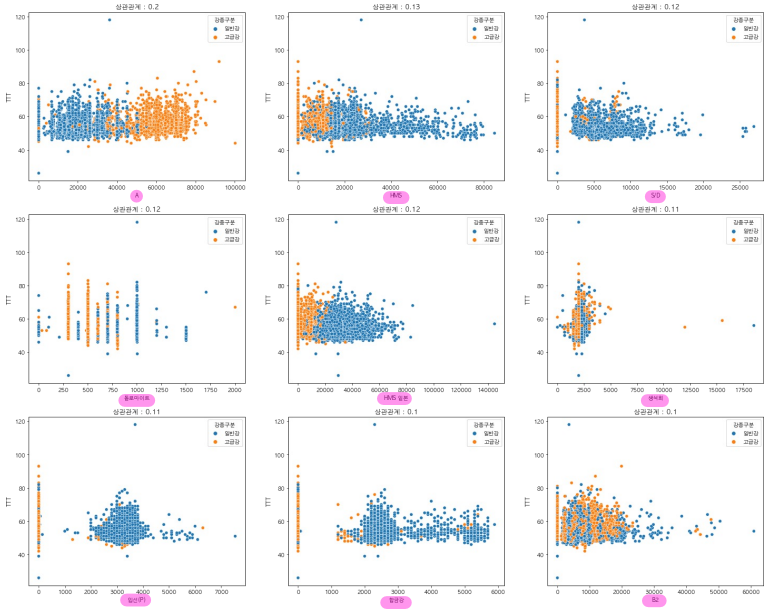
투입되는 스크랩 및 부원료 종류/양 으로 TTT 예측 모델링

- 2019~2021년까지 축적된 총 히트(row) 수: 8013
- 총 칼럼수: 35

TTT와 스크랩/부원료의 상관관계





상관관계가 높은 상위 9개 변수와 TTT간 산점도

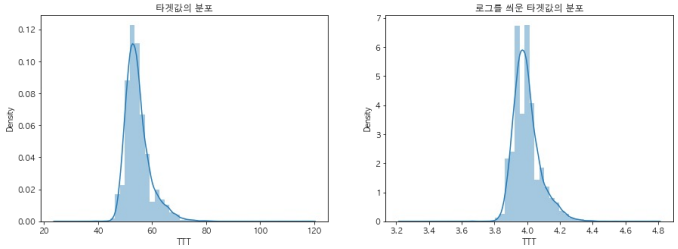
- 상위 9개 변수와 TTT의 산점도입니다.
- 일반강/고급강 여부에 따라 특징적인 패턴이 있는지 확인해봤으나, 유의미한 결과를 도출하진 못했습니다.
- 결론적으로 스크랩의 종류나 그 양이 TTT에 큰 영향을 주는 것은 아니라고 유추해 볼 수 있습니다.
- 상관관계의 계수값이 작다고 하여 그 가치가 중요하지 않다고 단정 지을 수는 없으므로, 한번 예측 모델링을 시도해 봤습니다.

데이터 전처리 및 모델링 과정 요약

1. 타겟값인 TTT의 분포가 한쪽으로 치우치진 않았습니다. (118 혹은 26 처럼 이상치인 듯한 값이 보이긴 함)
2. 모델링 시에는 90분 이상, 30분 이하의 TTT는 제거했습니다. (3개 히트 제거됨) 또한, 각 스크랩 양에 대한 값들은 모두 0~1로 정규화하였습니다.
3. 데이터는 총 8013개이고, Train셋으로 7013개, Test셋으로 1000개를 랜덤하게 나누었습니다.
4. 총 9개의 기본 회귀모델로 모델의 성능을 확인해봤습니다. 모델 성능을 확인할 때는 각 모델별로 10번의 교차검증으로 RMSE를 비교했습니다.
5. GBM, Extra Trees, SVR, Random Forest 순으로 모델 성능이 높게 나오는 것을 확인했고, 이 네 개의 모델에 대해서는 그리드 서치로 하이퍼파라미터 튜닝했습니다.
6. 마지막으로 Voting으로 네 가지 모델을 앙상블하여 Test셋을 예측했습니다.

- [추가] 변수 중요도가 낮은 변수들을 제거하고도 학습해봤지만, 결과가 크게 달라지진 않았습니다.

1,2 타겟값의 분포 및 이상치 제거

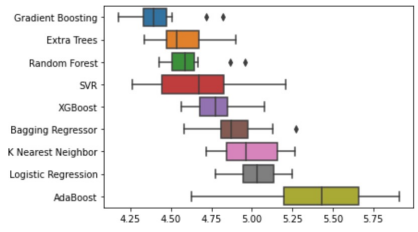



```
count    8016.000000
mean       54.707710
std         4.788244
min        26.000000
25%        52.000000
50%        54.000000
75%        57.000000
max       118.000000
Name: TTT, dtype: float64
```

- **min : 26, max : 118** 등의 값은 제거했습니다.

---

4 베이스 모델(하이퍼파라미터 튜닝X)로 학습 및 검증

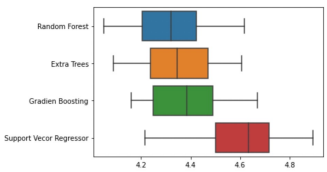


	0	1	2	3	4	5	6	7	8	9	mean
Gradient Boosting	4.384647	4.337039	4.719236	4.253721	4.174771	4.328888	4.423323	4.506181	4.392635	4.819201	4.438015
Extra Trees	4.730431	4.529724	4.901412	4.536943	4.336578	4.449924	4.480292	4.612290	4.465985	4.858901	4.593593
Random Forest	4.509322	4.667597	4.959881	4.563841	4.505363	4.428491	4.589386	4.601118	4.500592	4.868674	4.622244
SVR	4.550168	4.264076	4.734619	4.316953	4.334552	4.633751	4.790551	4.922410	4.860028	5.207789	4.670327
XGBoost	4.688998	4.778741	4.953368	4.868554	4.684965	4.566116	4.644842	4.827518	4.668778	5.075609	4.778063
Bagging Regressor	4.949541	5.006293	5.269698	4.827230	4.620579	4.582205	4.870709	4.800826	4.818474	5.127351	4.891404
K Nearest Neighbor	4.804388	4.939111	5.135175	4.717661	4.717286	4.880663	4.960871	5.263726	5.183393	5.205485	4.984563
Logistic Regression	5.246540	5.157615	5.199751	5.107582	5.021054	5.029625	4.974467	4.920212	4.861246	4.773130	5.031187
AdaBoost	5.425530	5.910596	5.893628	5.677497	5.053151	5.116087	5.644516	5.279816	4.628404	5.538027	5.430266

- 각 모델별 10번의 교차검증으로 모델의 rmse 성능을 확인했습니다. 평균적으로 가장 높은 성능을 보이는 모델은 **Gradient Boosting, Extra Trees, Random Forest, SVR(Support Vector Regression)** 입니다.

---

5 상위 4개 모델에 대해 그리드 서치를 통한 하이퍼파라미터 튜닝

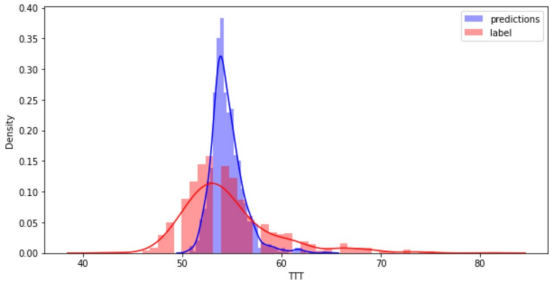


	0	1	2	3	4	5	6	7	8	9	mean
Random Forest	4.261755	4.450069	4.358167	4.616892	4.510952	4.215655	4.132102	4.192084	4.050673	4.396246	4.321784
Extra Trees	4.330979	4.483722	4.361573	4.605827	4.501952	4.231589	4.126614	4.242896	4.087615	4.457844	4.346080
Gradien Boosting	4.286704	4.525215	4.452259	4.669499	4.590250	4.259277	4.181171	4.239288	4.160488	4.444644	4.384180
Support Vecor Regressor	4.632634	4.736170	4.665835	4.891695	4.716198	4.500621	4.216114	4.499874	4.318016	4.718664	4.593745

```
gbr = GradientBoostingRegressor(learning_rate= 0.01, n_estimators=1000)
rf = RandomForestRegressor(max_features=0.2, min_samples_leaf=5, min_samples_split=6, n_estimators=200, random_st
ext = ExtraTreesRegressor(max_features=0.8, min_samples_leaf=10, min_samples_split=10, n_estimators=75, random_st
svr = SVR(C = 50, gamma=0.1)
```

- 그리드 서치로 찾은 하이퍼 파라미터(코드 이미지)로, Train셋에 대해 각각 10번의 교차검증을 진행했습니다. 모델의 성능은 위 테이블 및 왼쪽 Boxplot을 통해 확인하실 수 있습니다.
- 단일 모델은 새로운 데이터를 예측할 때, robust하지 못할 수 있습니다. 더 좋은 일반화 성능을 위해 여러 모델을 앙상블하여 최종적으로 Test셋을 예측해봤습니다.

6 모델앙상블을통한Test셋 예측



- 최종앙상블결과: RMSE : 4.182(Test셋 기준)
- 대부분50~60 사이로모델이값을 예측했습니다.

예측 결과예시(일부 발췌)

label	predict
49.0	53.96
50.0	53.69
60.0	53.9
50.0	54.65
55.0	55.98
56.0	54.33
50.0	54.13
48.0	53.79
56.0	51.89
44.0	58.63
57.0	53.71
56.0	53.46
54.0	55.6
56.0	56.5
51.0	54.82
56.0	52.46
50.0	53.58
59.0	59.13
58.0	54.82
53.0	53.77
73.0	58.44
61.0	53.93
66.0	55.64
54.0	53.9
54.0	55.35
47.0	52.02
75.0	55.91
67.0	55.61

56.0	54.99
54.0	53.63
53.0	50.72
52.0	55.36
52.0	55.75
54.0	54.6
51.0	53.65
54.0	53.95
55.0	55.72
53.0	54.01
51.0	54.62
52.0	53.8
49.0	52.65
49.0	52.77
54.0	54.45
50.0	53.47
53.0	51.81
53.0	54.08
50.0	54.26
50.0	53.42
53.0	56.91
63.0	53.72
52.0	52.22
55.0	54.71
49.0	54.03
52.0	53.73
49.0	52.43
55.0	56.99
60.0	55.56
50.0	52.66
48.0	51.42

결론

- **스크랩의 비율값을 새로운 변수로 추가하는 등의 feature engineering도 해봤지만, RMSE 4.xxx에서 더 나아지진 않았습니다.**
- **지금의 모델은 현장에서 크게 매력적이거나 도움이 될 것 같진 않습니다. 석사님 말씀대로 스크랩이 TTT에 크게 영향을 주지 않는 것 같습니다. 지난 목요일 미팅 에서 제안하신 방법으로 분석 및 모델링을 진행해보겠습니다.**
- **스크랩으로 TTT를 예측하는 모델링 결과에 대한 석사님의 피드백 부탁드립니다.**