PyCaret

AutoML 분야에서 가장 활성화 되어 있는 라이브러리.

> PyCarer외에 Microsoft AutoML, Auto-Sklearn 등이 있다.

패키지 설치

이 패키지가 스스로 모든 모델을 처리하는 것이 아니라 현재 시스템에 설치되어 있는 학습 모델들을 호출하는 것 뿐이다.

```
!pip install pycaret
```

```
!pip install shap
```

```
!pip install fairlearn
```

```
!pip install pycaret[analysis]
```

```
!pip install pycaret[models]
```

```
!pip install pycaret[tuner]
```

```
!pip install pycaret[mlops]
```

```
!pip install pycaret[parallel]
```

```
!pip install flask
```

패키지 참조

```
import warnings
warnings.filterwarnings('ignore')

from pycaret.classification import *
from pandas import read_excel
```

```
origin = read_excel("https://data.hossam.kr/G02/breast_cancer.xlsx")
origin.head()
```

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	m
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0

5 rows × 31 columns

데이터 전처리 자동화

그래도 전처리는 직접 수행하고 setup() 함수의 파라미터들은 이용하지 않는 것이 좋지 않을까?...

```
# 독립변수의 이름들만 추출
x_names = list(origin.drop('target', axis=1).columns)
x_names
```

```
['mean radius',
 'mean texture',
 'mean perimeter',
 'mean area',
 'mean smoothness',
 'mean compactness',
 'mean concavity',
 'mean concave points',
 'mean symmetry',
 'mean fractal dimension',
 'radius error',
 'texture error',
 'perimeter error',
 'area error',
 'smoothness error',
 'compactness error',
 'concavity error',
 'concave points error',
 'symmetry error',
 'fractal dimension error',
 'worst radius',
 'worst texture',
 'worst perimeter',
 'worst area',
 'worst smoothness',
 'worst compactness',
 'worst concavity',
 'worst concave points',
 'worst symmetry',
 'worst fractal dimension']
```

```
clf = setup(data=origin,
            target='target',
            #ignore_features=ignore_features,           # 분석/학습에 고려하지 않을
feature(컬럼) 제거
            #categorical_features=categorical_features, # 범주형 컬럼 지정
            # bin_numeric_features=['age', 'trestbps', 'chol','thalach',
'oldpeak'], # numeric 컬럼에 대하여 binning
            numeric_features=x_names,         # 수치형 컬럼 지정
            normalize=True,                            # 정규화 적용
            normalize_method='zscore',                 # 정규화 방식 지정
            imputation_type='iterative',               # 결측치를 lightgbm으로 예측하여
채움
            categorical_iterative_imputer='lightgbm',
            polynomial_features=True,                  # 다항식 생성
            session_id=12345,
            use_gpu=True
)
clf
```

```
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Info] This is the GPU trainer!!
[LightGBM] [Info] Total Bins 0
[LightGBM] [Info] Number of data points in the train set: 2, number of used features:
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf
[LightGBM] [Warning] Using sparse features with CUDA is currently not supported.
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Info] This is the GPU trainer!!
[LightGBM] [Info] Total Bins 0
[LightGBM] [Info] Number of data points in the train set: 2, number of used features:
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf
[LightGBM] [Warning] Using sparse features with CUDA is currently not supported.
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Info] This is the GPU trainer!!
[LightGBM] [Info] Total Bins 0
[LightGBM] [Info] Number of data points in the train set: 2, number of used features:
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf
[LightGBM] [Warning] Using sparse features with CUDA is currently not supported.
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Info] This is the GPU trainer!!
[LightGBM] [Info] Total Bins 0
[LightGBM] [Info] Number of data points in the train set: 2, number of used features:
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf
[LightGBM] [Warning] Using sparse features with CUDA is currently not supported.
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Info] This is the GPU trainer!!
```

```
[LightGBM] [Info] Total Bins 0
[LightGBM] [Info] Number of data points in the train set: 2, number of used features:
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Warning] Using sparse features with CUDA is currently not supported.
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Info] This is the GPU trainer!!
[LightGBM] [Info] Total Bins 0
[LightGBM] [Info] Number of data points in the train set: 2, number of used features:
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Warning] Using sparse features with CUDA is currently not supported.
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Info] This is the GPU trainer!!
[LightGBM] [Info] Total Bins 0
[LightGBM] [Info] Number of data points in the train set: 2, number of used features:
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Warning] Using sparse features with CUDA is currently not supported.
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Info] This is the GPU trainer!!
[LightGBM] [Info] Total Bins 0
[LightGBM] [Info] Number of data points in the train set: 2, number of used features:
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Warning] Using sparse features with CUDA is currently not supported.
[LightGBM] [Info] Number of positive: 1, number of negative: 1
```

	Description	Value
0	Session id	12345
1	Target	target
2	Target type	Binary
3	Original data shape	(569, 31)
4	Transformed data shape	(569, 496)
5	Transformed train set shape	(398, 496)
6	Transformed test set shape	(171, 496)
7	Numeric features	30
8	Preprocess	True
9	Imputation type	iterative
10	Iterative imputation iterations	5
11	Numeric iterative imputer	lightgbm
12	Categorical iterative imputer	lightgbm
13	Polynomial features	True
14	Polynomial degree	2
15	Normalize	True

	Description	Value
16	Normalize method	zscore
17	Fold Generator	StratifiedKFold
18	Fold Number	10
19	CPU Jobs	-1
20	Use GPU	True
21	Log Experiment	False
22	Experiment Name	clf-default-name
23	USI	c000

```
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Info] This is the GPU trainer!!
[LightGBM] [Info] Total Bins 0
[LightGBM] [Info] Number of data points in the train set: 2, number of used features:
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Warning] Using sparse features with CUDA is currently not supported.
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Info] Number of positive: 1, number of negative: 1
[LightGBM] [Info] This is the GPU trainer!!
[LightGBM] [Info] Total Bins 0
[LightGBM] [Info] Number of data points in the train set: 2, number of used features:
[LightGBM] [Warning] There are no meaningful features which satisfy the provided conf:
[LightGBM] [Warning] Using sparse features with CUDA is currently not supported.
[LightGBM] [Info] Number of positive: 1, number of negative: 1
```

```
<pycaret.classification.oop.ClassificationExperiment at 0x7a2c0d7c8640>
```

모델별 비교 후 상위 5개 선정

여기서는 정확도가 높은 순서로 정렬함

다소 시간이 걸린다. 꼭 GPU 사용하자.

```
top5_models = compare_models(sort='acc', n_select=5)
top5_models
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
lr	Logistic Regression	0.9724	0.9914	0.9840	0.9736	0.9783	0.9404	0.9424	0.4110
knn	K Neighbors Classifier	0.9675	0.9748	0.9840	0.9658	0.9745	0.9297	0.9311	0.1960

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
et	Extra Trees Classifier	0.9675	0.9912	0.9840	0.9658	0.9743	0.9300	0.9320	0.2970
catboost	CatBoost Classifier	0.9674	0.9920	0.9840	0.9658	0.9743	0.9298	0.9318	39.8400
ada	Ada Boost Classifier	0.9649	0.9907	0.9800	0.9659	0.9724	0.9244	0.9265	1.3740
lightgbm	Light Gradient Boosting Machine	0.9625	0.9923	0.9760	0.9653	0.9700	0.9199	0.9221	2.0330
rf	Random Forest Classifier	0.9624	0.9935	0.9720	0.9690	0.9699	0.9198	0.9217	0.4320
ridge	Ridge Classifier	0.9623	0.0000	0.9880	0.9557	0.9709	0.9174	0.9209	0.2770
svm	SVM - Linear Kernel	0.9622	0.0000	0.9720	0.9692	0.9702	0.9187	0.9200	0.1760
gbc	Gradient Boosting Classifier	0.9599	0.9925	0.9720	0.9655	0.9679	0.9145	0.9175	5.8670
xgboost	Extreme Gradient Boosting	0.9599	0.9885	0.9680	0.9688	0.9678	0.9147	0.9167	1.0700
nb	Naive Bayes	0.9322	0.9452	0.9520	0.9415	0.9464	0.8540	0.8552	0.1680
dt	Decision Tree Classifier	0.9322	0.9298	0.9400	0.9518	0.9452	0.8564	0.8587	0.2940
lda	Linear Discriminant Analysis	0.8013	0.7949	0.8560	0.8370	0.8440	0.5695	0.5761	0.2530
qda	Quadratic Discriminant Analysis	0.7612	0.7492	0.7960	0.8236	0.8072	0.4921	0.4971	0.1720
dummy	Dummy Classifier	0.6282	0.5000	1.0000	0.6282	0.7716	0.0000	0.0000	0.0930

```
Processing:   0%|          | 0/73 [00:00<?, ?it/s]
```

```
[LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                    intercept_scaling=1, l1_ratio=None, max_iter=1000,
                    multi_class='auto', n_jobs=None, penalty='l2',
                    random_state=12345, solver='lbfgs', tol=0.0001, verbose=0,
                    warm_start=False),
 KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                      metric_params=None, n_jobs=-1, n_neighbors=5, p=2,
                      weights='uniform'),
 ExtraTreesClassifier(bootstrap=False, ccp_alpha=0.0, class_weight=None,
                      criterion='gini', max_depth=None, max_features='sqrt',
                      max_leaf_nodes=None, max_samples=None,
                      min_impurity_decrease=0.0, min_samples_leaf=1,
                      min_samples_split=2, min_weight_fraction_leaf=0.0,
                      n_estimators=100, n_jobs=-1, oob_score=False,
                      random_state=12345, verbose=0, warm_start=False),
 <catboost.core.CatBoostClassifier at 0x7a2b93dabd90>,
 AdaBoostClassifier(algorithm='SAMME.R', base_estimator='deprecated',
                    estimator=None, learning_rate=1.0, n_estimators=50,
                    random_state=12345)]
```

```
WARNING: Runtime no longer has a reference to this dataframe, please re-run this cell
```

모델 최적화

상위 5개 모델에 대하여 Voting 알고리즘 적용

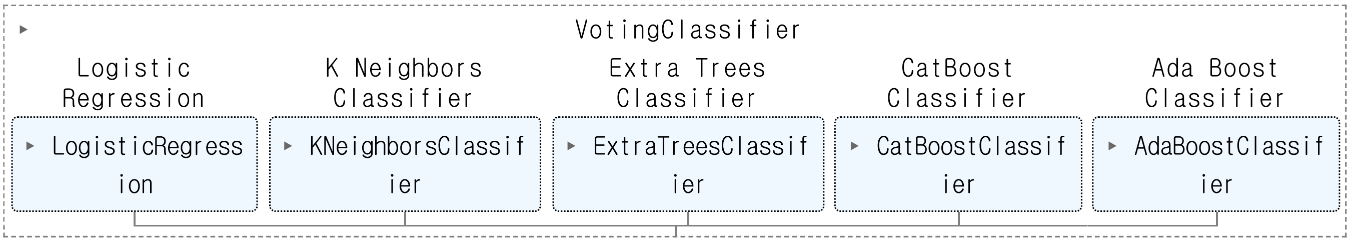
`ensemble_model()` 함수를 사용하면 앙상블 기법을 적용할 수 도 있다.

```
ens_models = ensemble_model(top5_models)
```

```
blended_models = blend_models(estimator_list=top5_models)
blended_models
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
Fold							
0	0.9750	1.0000	1.0000	0.9615	0.9804	0.9459	0.9473
1	0.9750	0.9787	1.0000	0.9615	0.9804	0.9459	0.9473
2	0.9500	1.0000	1.0000	0.9259	0.9615	0.8904	0.8958
3	0.9750	0.9920	1.0000	0.9615	0.9804	0.9459	0.9473
4	0.9250	0.9947	0.9200	0.9583	0.9388	0.8421	0.8433
5	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
6	0.9500	0.9333	1.0000	0.9259	0.9615	0.8904	0.8958
7	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
8	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
9	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Mean	0.9750	0.9899	0.9920	0.9695	0.9803	0.9461	0.9477
Std	0.0250	0.0199	0.0240	0.0280	0.0200	0.0535	0.0522

```
Processing:   0%|          | 0/6 [00:00<?, ?it/s]
```



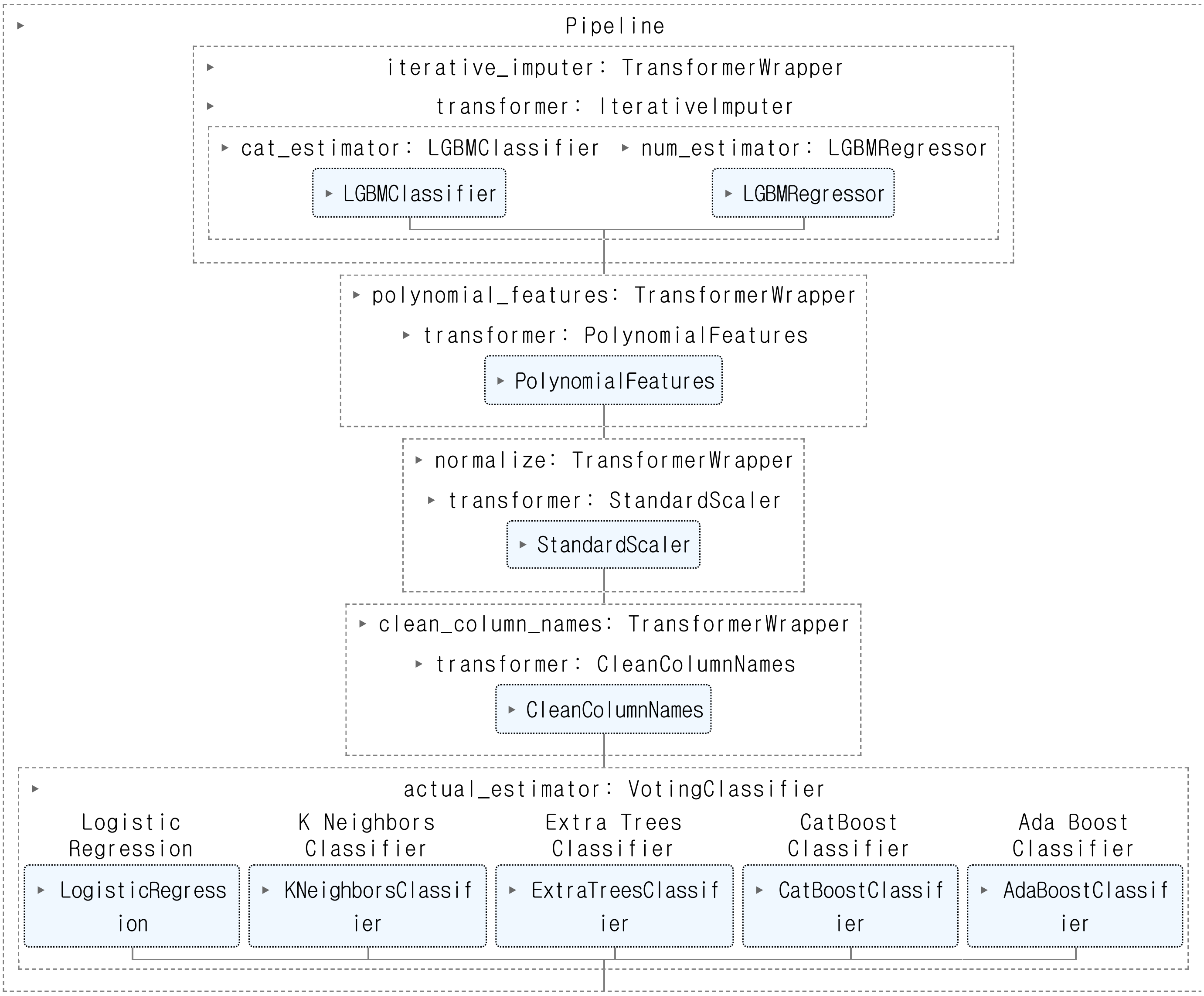
모델에 대한 하이퍼 파라미터 튜닝

> 엄청 오래 걸림. 각오할 것!!!

```
tuned_rf = tune_model(blended_models)
tuned_rf
```

최종 모델 정의

```
final_model = finalize_model(tuned_rf)
final_model
```


Pipeline
iterative_imputer: TransformerWrapper
transformer: IterativeImputer
cat_estimator: LGBMClassifier
num_estimator: LGBMRegressor
LGBMClassifier
LGBMRegressor
polynomial_features: TransformerWrapper
transformer: PolynomialFeatures
PolynomialFeatures
normalize: TransformerWrapper
transformer: StandardScaler
StandardScaler
clean_column_names: TransformerWrapper
transformer: CleanColumnNames
CleanColumnNames
actual_estimator: VotingClassifier
Logistic Regression
K Neighbors Classifier
Extra Trees Classifier
CatBoost Classifier
Ada Boost Classifier
LogisticRegression
KNeighborsClassifier
ExtraTreesClassifier
CatBoostClassifier
AdaBoostClassifier

결과 확인

예측치 산성

```
prediction = predict_model(final_model, data = origin)
prediction
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	Voting Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness
0	17.990000	10.380000	122.800003	1001.000000	0.11840	0.27760
1	20.570000	17.770000	132.899994	1326.000000	0.08474	0.07864
2	19.690001	21.250000	130.000000	1203.000000	0.10960	0.15990
3	11.420000	20.379999	77.580002	386.100006	0.14250	0.28390
4	20.290001	14.340000	135.100006	1297.000000	0.10030	0.13280
...	...	...	...	...	...	...
564	21.559999	22.389999	142.000000	1479.000000	0.11100	0.11590
565	20.129999	28.250000	131.199997	1261.000000	0.09780	0.10340
566	16.600000	28.080000	108.300003	858.099976	0.08455	0.10230
567	20.600000	29.330000	140.100006	1265.000000	0.11780	0.27700
568	7.760000	24.540001	47.919998	181.000000	0.05263	0.04362

569 rows × 33 columns

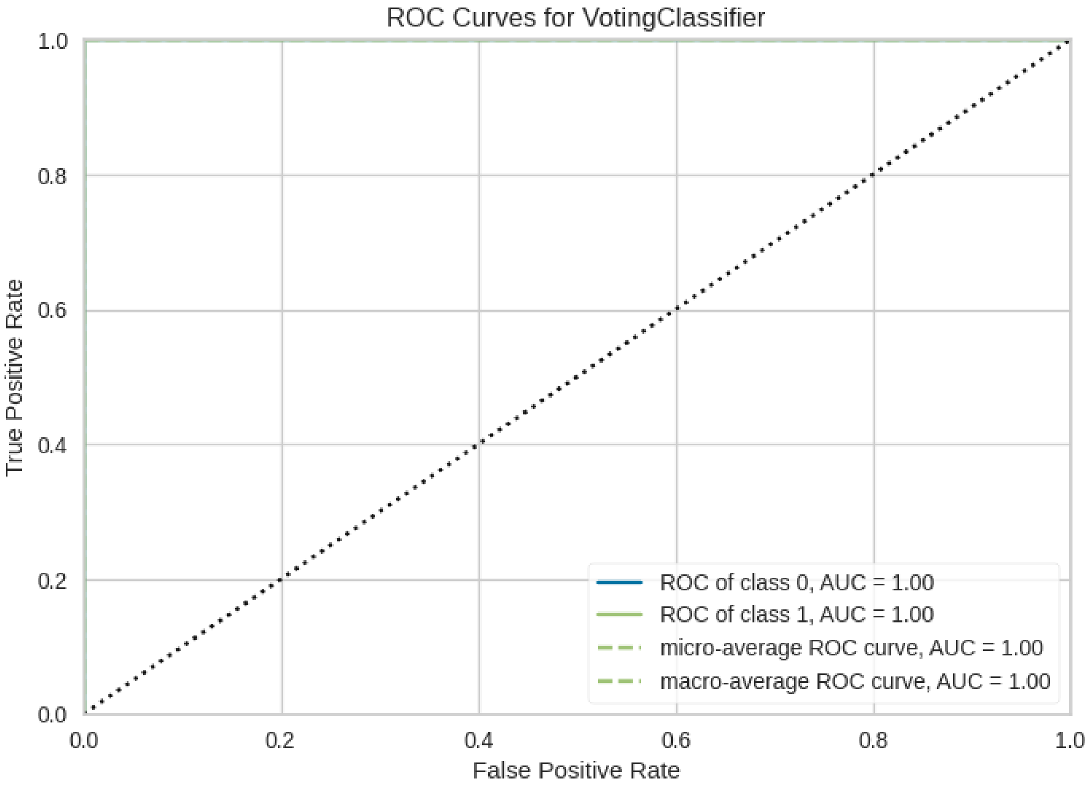
각 플롯 확인

```
evaluate_model(final_model)
```

```
interactive(children=(ToggleButtons(description='Plot Type:', icons=('',), options=((
```

학습한 모델에 대한 결과 확인

```
plot_model(final_model)
```



데이터셋에 각 feature들에 대해 measure를 확인

예를 들어 성별이 feature로 있을 때 check_fairness를 적용시키면, 성별 별로 얼마만큼의 데이터셋이 존재하고, 결과가 어떻게 되는 지 제공

여기서는 적합한 변수가 없어서 아무 값이나 지정함

```
check_fairness(final_model, sensitive_features=['mean radius'])
```

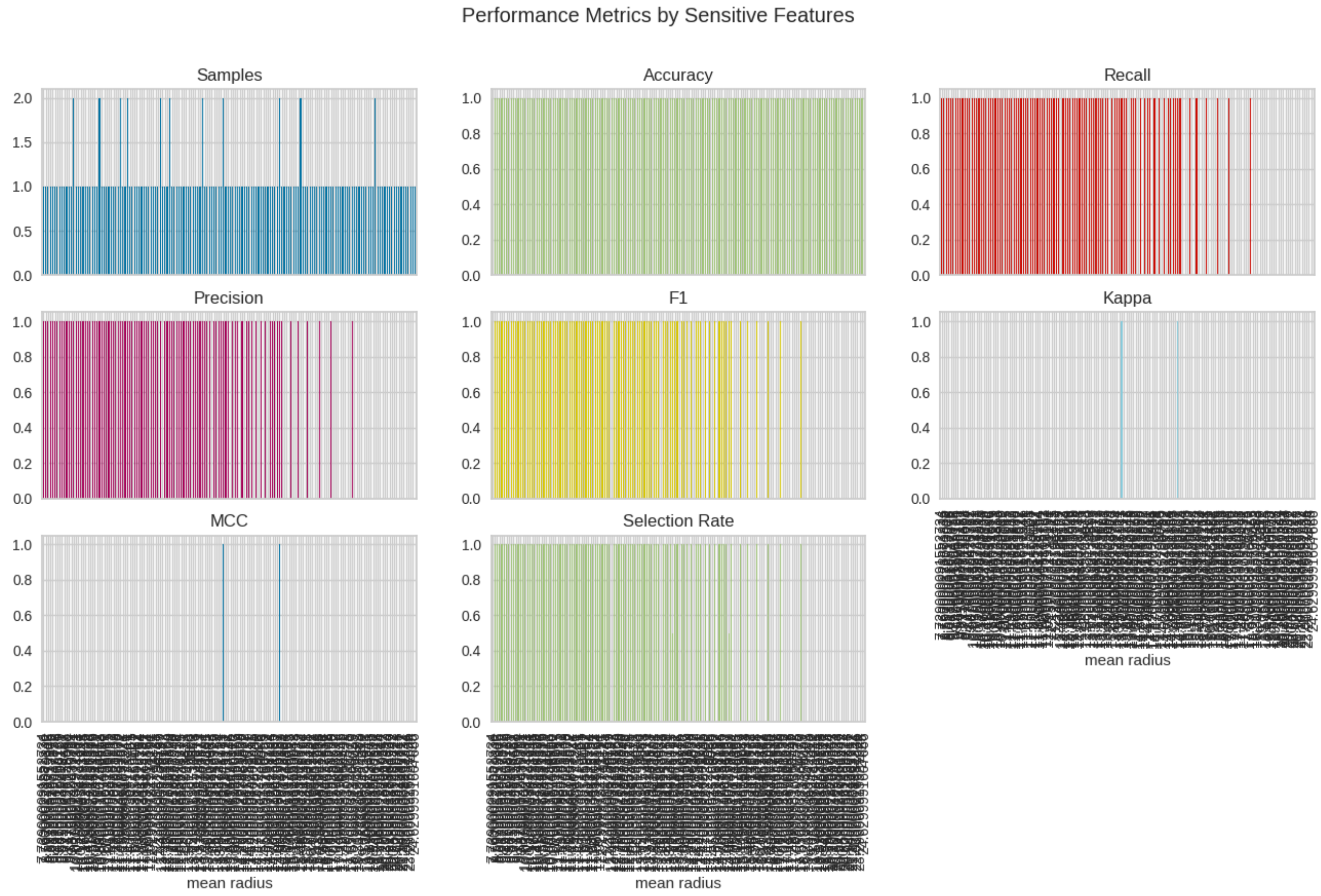
	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	Voting Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

```
WARNING:fairlearn.metrics._metric_frame:Found 160 subgroups. Evaluation may be slow
WARNING:fairlearn.metrics._metric_frame:Found 160 subgroups. Evaluation may be slow
```

	Samples	Accuracy	Recall	Precision	F1	Kappa	MCC	Selection Rate
mean radius								
7.729000	1	1.0	1.0	1.0	1.0	NaN	0.0	1.0
7.760000	1	1.0	1.0	1.0	1.0	NaN	0.0	1.0
8.597000	1	1.0	1.0	1.0	1.0	NaN	0.0	1.0
8.598000	1	1.0	1.0	1.0	1.0	NaN	0.0	1.0
8.618000	1	1.0	1.0	1.0	1.0	NaN	0.0	1.0
...	...	...	...	...	...	...	...	...
21.370001	1	1.0	0.0	0.0	0.0	NaN	0.0	0.0
22.010000	1	1.0	0.0	0.0	0.0	NaN	0.0	0.0

	Samples	Accuracy	Recall	Precision	F1	Kappa	MCC	Selection Rate
mean radius								
23.290001	1	1.0	0.0	0.0	0.0	NaN	0.0	0.0
24.250000	1	1.0	0.0	0.0	0.0	NaN	0.0	0.0
24.629999	1	1.0	0.0	0.0	0.0	NaN	0.0	0.0

160 rows × 8 columns



setup 이후 훈련된 모든 모델을 출력

```
get_leaderboard()
```

```
Processing:   0%|          | 0/18 [00:00<?, ?it/s]
```

	Model Name	Model	Accuracy	AU
Index				
0	Logistic Regression	(TransformerWrapper(exclude=None, include=None...	0.9724	0.99
1	K Neighbors Classifier	(TransformerWrapper(exclude=None, include=None...	0.9675	0.97
2	Naive Bayes	(TransformerWrapper(exclude=None, include=None...	0.9322	0.94
3	Decision Tree Classifier	(TransformerWrapper(exclude=None, include=None...	0.9322	0.92
4	SVM - Linear Kernel	(TransformerWrapper(exclude=None, include=None...	0.9622	0.00

	Model Name	Model	Accuracy	AU
Index				
5	Ridge Classifier	(TransformerWrapper(exclude=None, include=None...	0.9623	0.00
6	Random Forest Classifier	(TransformerWrapper(exclude=None, include=None...	0.9624	0.99
7	Quadratic Discriminant Analysis	(TransformerWrapper(exclude=None, include=None...	0.7612	0.74
8	Ada Boost Classifier	(TransformerWrapper(exclude=None, include=None...	0.9649	0.99
9	Gradient Boosting Classifier	(TransformerWrapper(exclude=None, include=None...	0.9599	0.99
10	Linear Discriminant Analysis	(TransformerWrapper(exclude=None, include=None...	0.8013	0.79
11	Extra Trees Classifier	(TransformerWrapper(exclude=None, include=None...	0.9675	0.99
12	Extreme Gradient Boosting	(TransformerWrapper(exclude=None, include=None...	0.9599	0.98
13	Light Gradient Boosting Machine	(TransformerWrapper(exclude=None, include=None...	0.9625	0.99
14	CatBoost Classifier	(TransformerWrapper(exclude=None, include=None...	0.9674	0.99
15	Dummy Classifier	(TransformerWrapper(exclude=None, include=None...	0.6282	0.50
16	Voting Classifier	(TransformerWrapper(exclude=None, include=None...	0.9750	0.98

간단한 dashboard 생성

> 웹 서버가 가동되어야 하므로 코랩에서는 실행되지 않는 듯 함

```
dashboard(final_model)
```