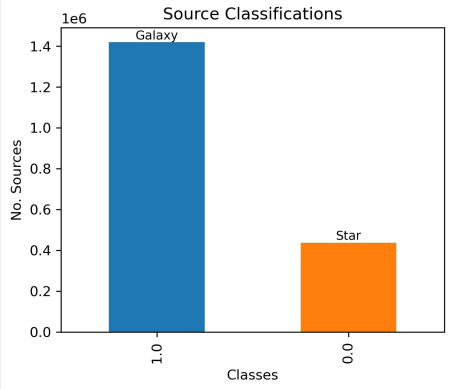
BROWN PHYSICS

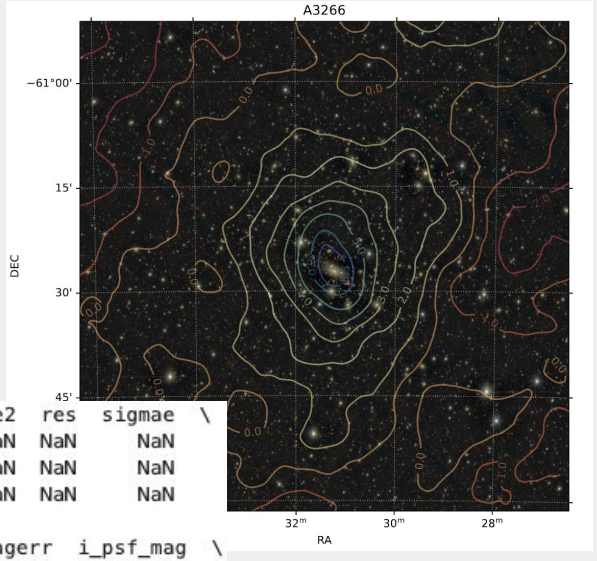
Star/Galaxy Classification in a Galaxy Cluster Field Using ML

Zacharias Escalante
Brown University Department of Physics
December 12, 2024
GitHub

Recap

- Catalogs of object magnitudes + shapes
- “Extendedness” column → Classification!!
- Identify stars and galaxies in our fields to better calibrate weak lensing measurements





	ra	dec	x	y	e1	e2	res	sigmae
0	68.914455	-62.906044	17394.772375	3890.481991	NaN	NaN	NaN	NaN
1	69.009523	-62.904732	16801.596400	3898.262765	NaN	NaN	NaN	NaN
2	68.977504	-62.904551	17001.230648	3904.279632	NaN	NaN	NaN	NaN

	rkron	extendedness	...	r_cmodel_mag	r_cmodel_magerr	i_psf_mag
0	2.177848	1.0	...	21.828693	0.031725	21.740775
1	6.150654	1.0	...	19.786428	0.011056	27.238589
2	3.431714	1.0	...	21.139035	0.020000	25.215983

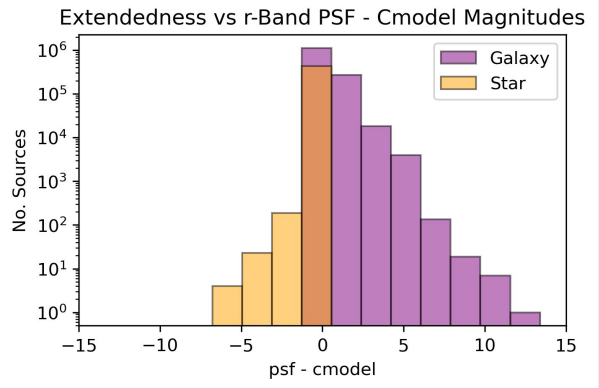
	i_psf_magerr	i_cmodel_mag	i_cmodel_magerr	z_psf_mag	z_psf_magerr
0	0.033023	21.712547	0.033202	21.778364	0.078916
1	1.082054	NaN	NaN	25.915949	0.343267
2	1.075873	NaN	NaN	NaN	-3.567432

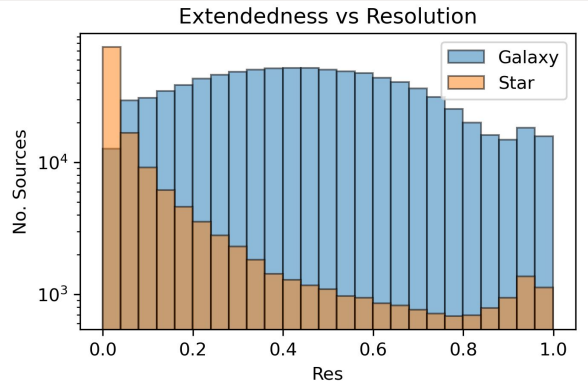
	z_cmodel_mag	z_cmodel_magerr
0	21.738405	0.078901
1	NaN	NaN
2	NaN	NaN

Target Variable

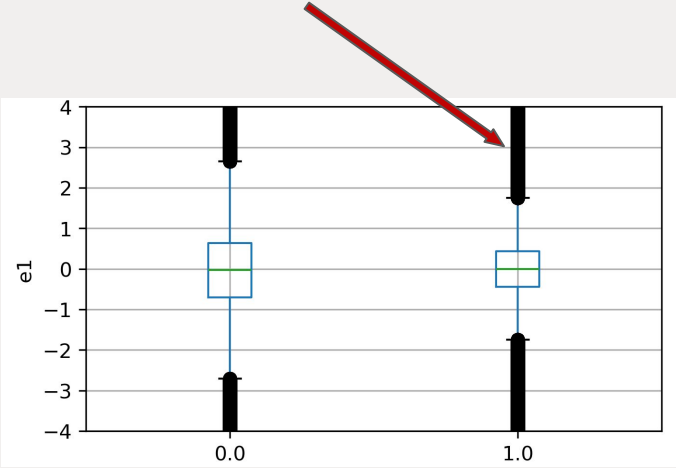
Recap

- ~1.8 Million rows, 41 features
- 5 engineered features added → 46 total





All outliers!



Splitting/Preprocessing

- ~72% with missing values
- IterativeImputer
- Stratified 60/20/20
- One categorical feature OHE
- MinMax → Ellipticities, coordinates, res, blendedness
- Standard → sigmae, rkron, all magnitudes

```
# loop through the different random states
for i in range(nr_states):
    print('random state '+str(i+1))

    X_train, X_other, y_train, y_other = train_test_split(X,y,train_size = 0.6,stratify=y,random_state=i)
    X_val, X_test, y_val, y_test = train_test_split(X_other,y_other,train_size = 0.5,stratify=y_other,random_state=i)

    # Set up preprocessor
    minmax_pipeline = Pipeline(steps=[
        ('imputer', IterativeImputer(max_iter=20, tol=1e-1,random_state=i)),
        ('scaler', MinMaxScaler())
    ])

    std_pipeline = Pipeline(steps=[
        ('imputer', IterativeImputer(max_iter=20, tol=1e-1,random_state=i)),
        ('scaler', StandardScaler())
    ])

    prep_iter = ColumnTransformer(
        transformers=[
            ('minmax', minmax_pipeline, minmax_ftrs),
            ('std', std_pipeline, std_ftrs)
        ],
        remainder='passthrough'
    )

    # preprocess the sets
    X_train_prep = prep_iter.fit_transform(X_train)
    X_val_prep = prep_iter.transform(X_val)
    X_test_prep = prep_iter.transform(X_test)
```

```
(array([0., 1.]), array([262635, 851976]))
(array([0., 1.]), array([ 87545, 283992]))
(array([0., 1.]), array([ 87545, 283992]))
```

```
preprocessed train size: (1114611, 46)
preprocessed validation size: (371537, 46)
preprocessed test size: (371537, 46)
```

Cross-Validation

- 5 random states
- Logistic Regression (Elastic Net)
- SVC
- KNeighborsClassifier
- Random Forest

```
'l1_ratio': np.linspace(0,1,3)
'C': 1/np.logspace(-5,5,5)
```

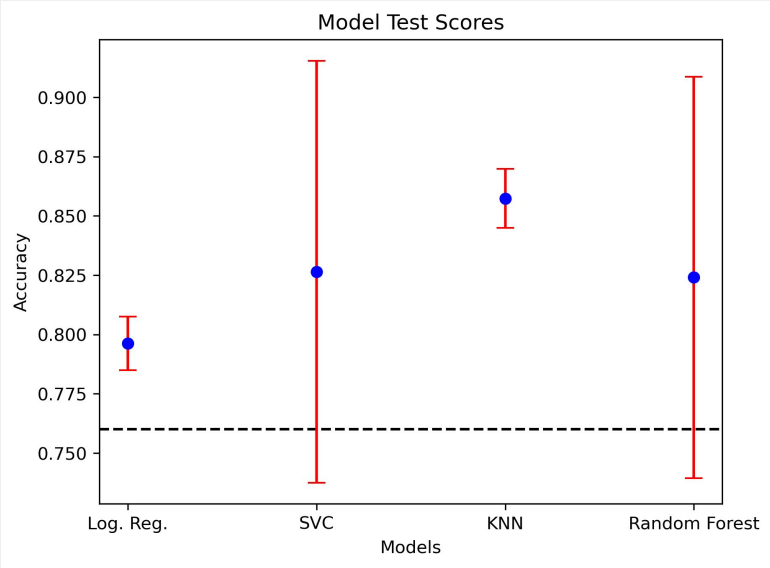
```
'gamma': [1e-1, 1e0, 1e1],
'C': [1e-2, 1e-1, 1e0, 1e1, 1e2]
'kernel': ['linear']
```

```
'n_neighbors': [10, 100, 1000],
'weights': ['uniform','distance']
```

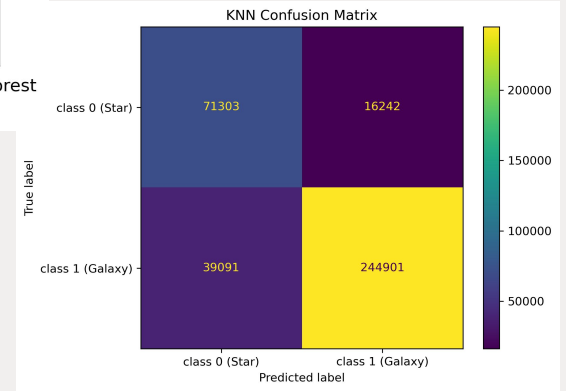
```
'max_depth': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
'n_estimators': [1, 3, 10, 30]
```

Results

- Baseline score: ~76%
- Best model: KNeighborsClassifier
- SVC/KNN/RF → 2%
  - ~85% - 99% Acc.

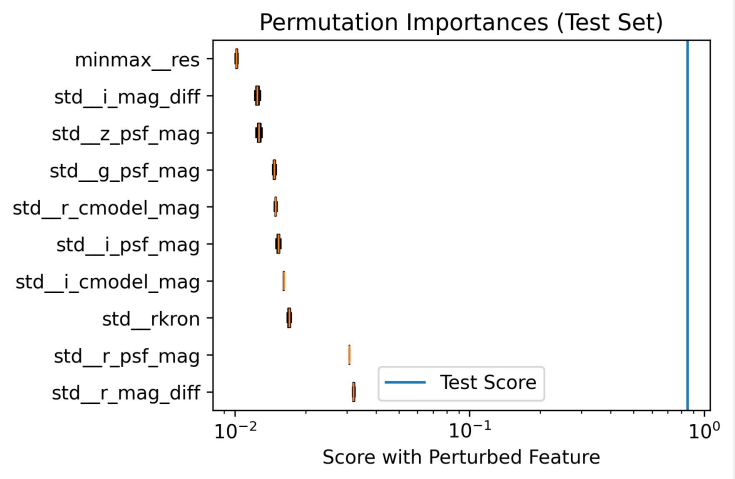


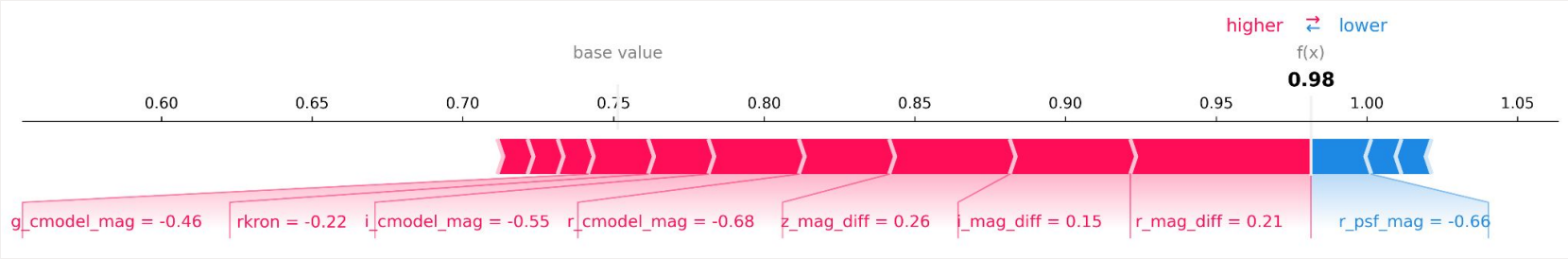
Accuracy: 0.8511
Precision: 0.9378
Recall: 0.8624



Results


Class 0 (Star) Shap Values
std__r_mag_diff
std__r_psf_mag
std__r_cmodel_mag
std__rkron
std__i_mag_diff
std__i_cmodel_mag
std__i_psf_mag
std__g_psf_mag
std__z_mag_diff
std__g_cmodel_mag
−0.2
−0.1
0.0
0.1
0.2
SHAP value (impact on model output)
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4


Permutation Importances (Test Set)
minmax__res
std__i_mag_diff
std__z_psf_mag
std__g_psf_mag
std__r_cmodel_mag
std__i_psf_mag
std__i_cmodel_mag
std__rkron
std__r_psf_mag
std__r_mag_diff
Test Score
10^{-2}
10^{-1}
10^{0}
Score with Perturbed Feature


higher ⇄ lower
base value
f(x)
0.98
0.60
0.65
0.70
0.75
0.80
0.85
0.90
0.95
1.00
1.05
g_cmodel_mag = -0.46
rkron = -0.22
i_cmodel_mag = -0.55
r_cmodel_mag = -0.68
z_mag_diff = 0.26
i_mag_diff = 0.15
r_mag_diff = 0.21
r_psf_mag = -0.66

Outlook/Conclusion

- Search wider hyperparameter range
- Use larger subset of data
- Data may not be i.i.d.?


Questions?