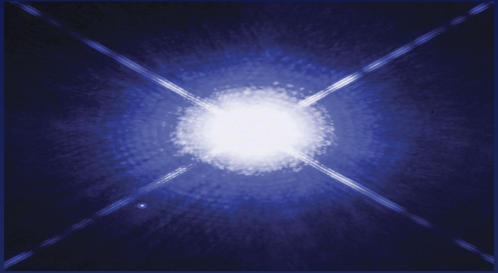
# AAI 595 Final Project Report- Machine Learning Approaches for Celestial Object Classification

Chris Muro, Rocco Gannon, Marc DiGeronimo

# Introduction

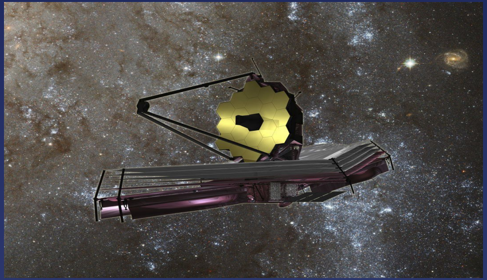
Dataset- Stellar Classification Dataset - SDSS17



Star

Galaxy

Quasar



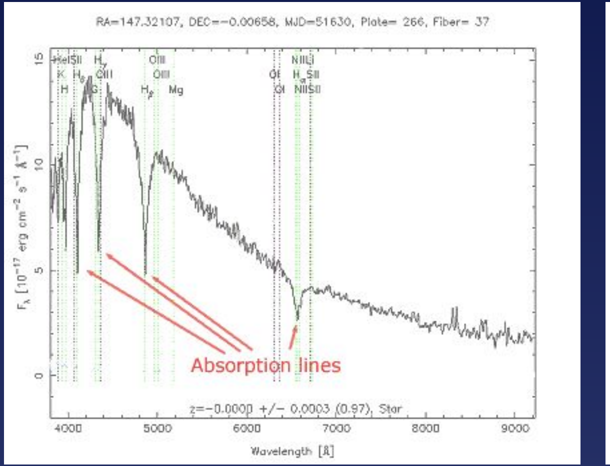
James Webb Space Telescope

Vera C. Rubin Observatory

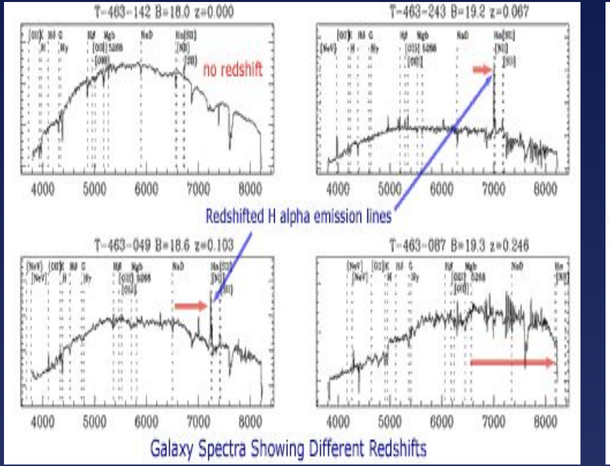
Sloan Memorial Telescope

# Related Work

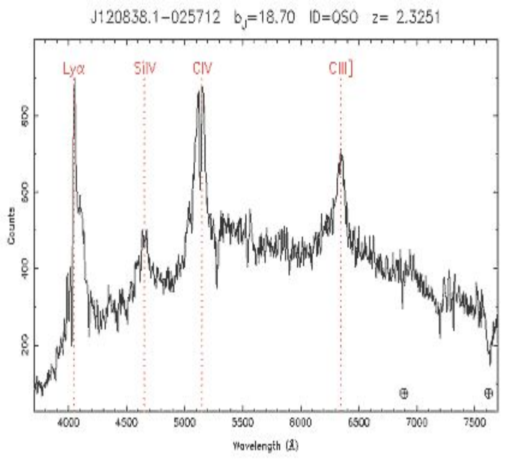
Australia Telescope National Facility Spectrum Graphs



Star Spectrum Showing Absorption Lines



Galaxy Spectra Showing different Redshifts



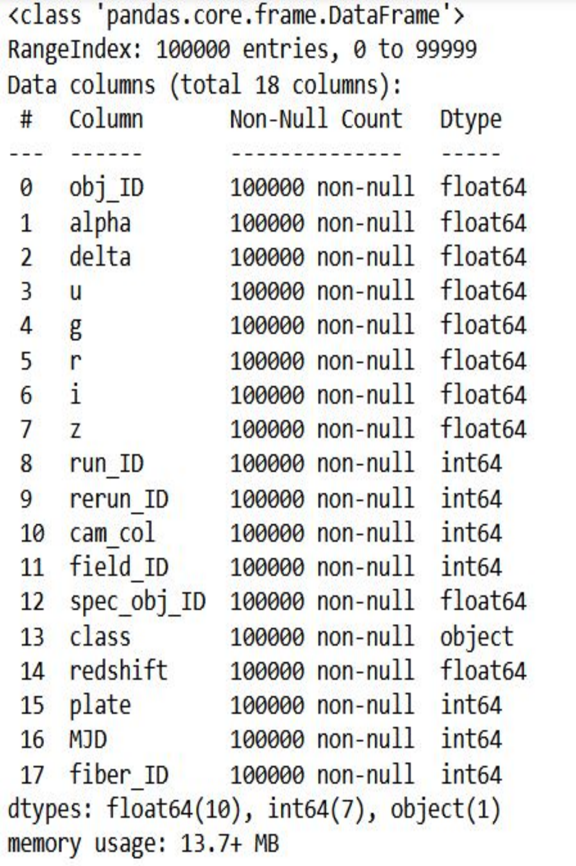
Quasar Spectrum Showing Emission Lines

# Related Work Continued

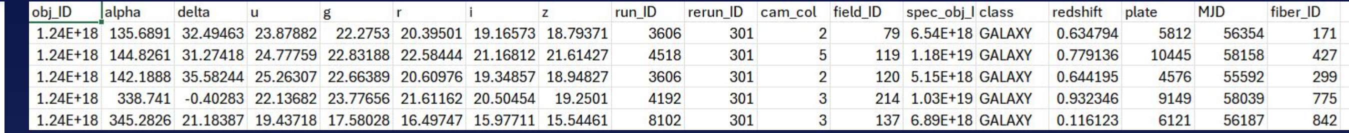
Hubble Extreme Deep Field

Diffraction Spikes
In a Star

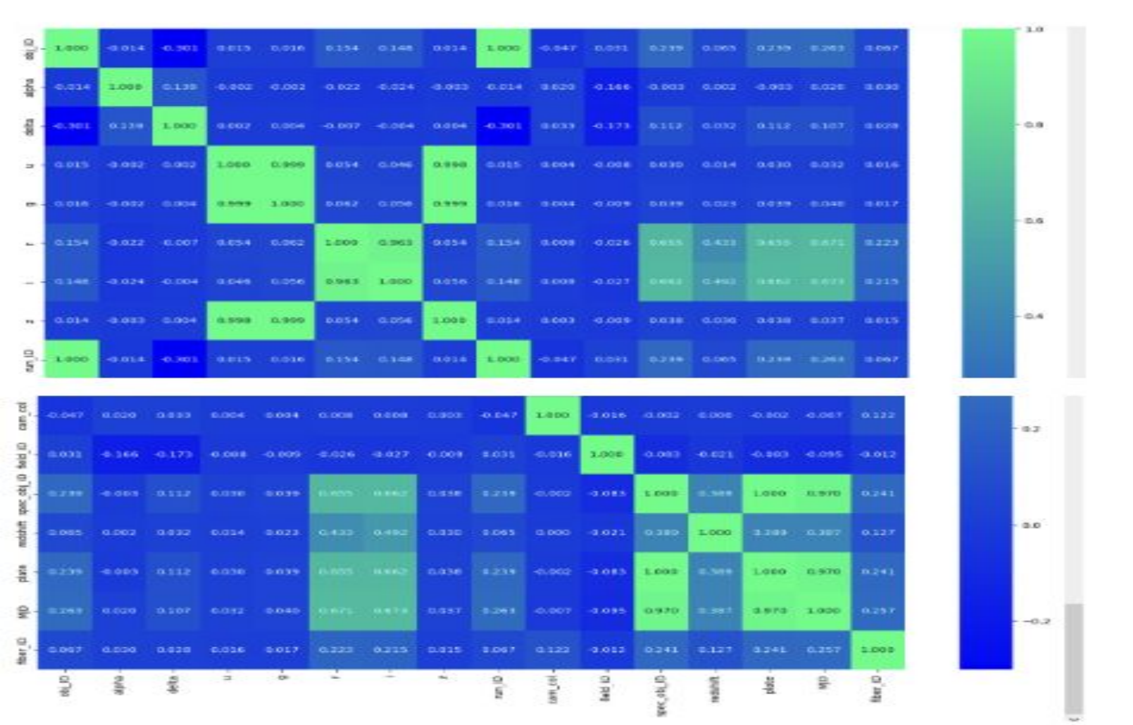
# Description of Dataset



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100000 entries, 0 to 99999
Data columns (total 18 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   obj_ID       100000 non-null  float64
 1   alpha        100000 non-null  float64
 2   delta        100000 non-null  float64
 3   u            100000 non-null  float64
 4   g            100000 non-null  float64
 5   r            100000 non-null  float64
 6   i            100000 non-null  float64
 7   z            100000 non-null  float64
 8   run_ID       100000 non-null  int64
 9   rerun_ID     100000 non-null  int64
 10  cam_col      100000 non-null  int64
 11  field_ID     100000 non-null  int64
 12  spec_obj_ID  100000 non-null  float64
 13  class        100000 non-null  object
 14  redshift     100000 non-null  float64
 15  plate        100000 non-null  int64
 16  MJD          100000 non-null  int64
 17  fiber_ID     100000 non-null  int64
dtypes: float64(10), int64(7), object(1)
memory usage: 13.7+ MB
```



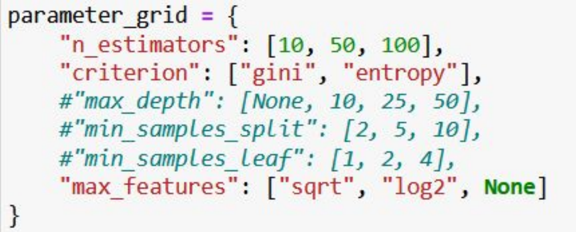
| obj_ID | alpha | delta | u | g | r | i | z | run_ID | rerun_ID | cam_col | field_ID | spec_obj_I | class | redshift | plate | MJD | fiber_ID |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1.24E+18 | 135.6891 | 32.49463 | 23.87882 | 22.2753 | 20.39501 | 19.16573 | 18.79371 | 3606 | 301 | 2 | 79 | 6.54E+18 | GALAXY | 0.634794 | 5812 | 56354 | 171 |
| 1.24E+18 | 144.8261 | 31.27418 | 24.77759 | 22.83188 | 22.58444 | 21.16812 | 21.61427 | 4518 | 301 | 5 | 119 | 1.18E+19 | GALAXY | 0.779136 | 10445 | 58158 | 427 |
| 1.24E+18 | 142.1888 | 35.58244 | 25.26307 | 22.66389 | 20.60976 | 19.34857 | 18.94827 | 3606 | 301 | 2 | 120 | 5.15E+18 | GALAXY | 0.644195 | 4576 | 55592 | 299 |
| 1.24E+18 | 338.741 | -0.40283 | 22.13682 | 23.77656 | 21.61162 | 20.50454 | 19.2501 | 4192 | 301 | 3 | 214 | 1.03E+19 | GALAXY | 0.932346 | 9149 | 58039 | 775 |
| 1.24E+18 | 345.2826 | 21.18387 | 19.43718 | 17.58028 | 16.49747 | 15.97711 | 15.54461 | 8102 | 301 | 3 | 137 | 6.89E+18 | GALAXY | 0.116123 | 6121 | 56187 | 842 |



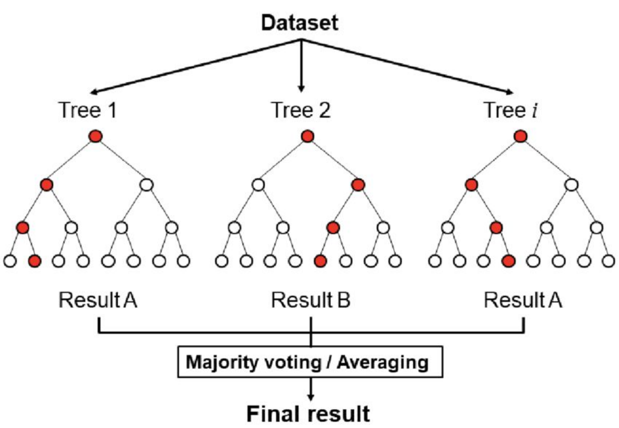
# Random Forest

Reason: Decision trees are well suited for classification problems, designed to perform inherent feature selection, ensemble method.

Parameters Tested:

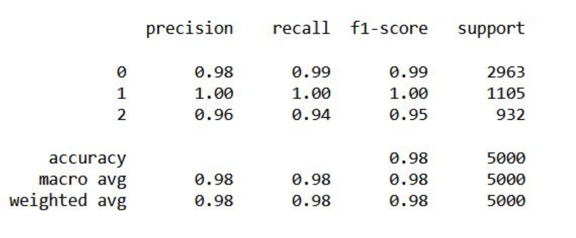


```
parameter_grid = {
    "n_estimators": [10, 50, 100],
    "criterion": ["gini", "entropy"],
    #"max_depth": [None, 10, 25, 50],
    #"min_samples_split": [2, 5, 10],
    #"min_samples_leaf": [1, 2, 4],
    "max_features": ["sqrt", "log2", None]
}
```



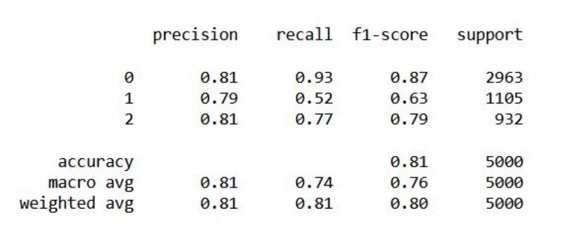
Best Parameters:  n_estimators = 100, criterion = entropy, max_features = None

# Random Forest Results



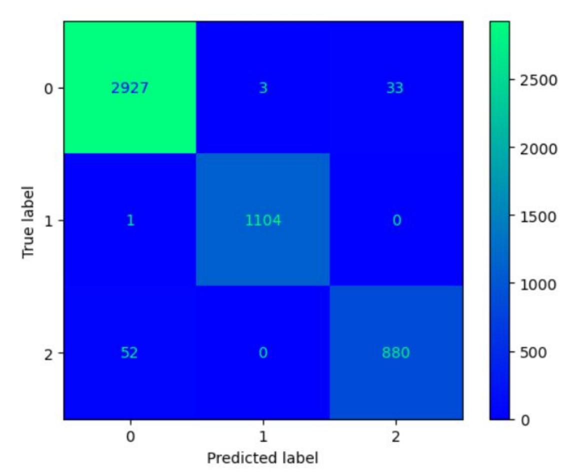
```
              precision    recall  f1-score   support

           0       0.98      0.99      0.99      2963
           1       1.00      1.00      1.00      1105
           2       0.96      0.94      0.95       932

    accuracy                           0.98      5000
   macro avg       0.98      0.98      0.98      5000
weighted avg       0.98      0.98      0.98      5000
```

Classification Report (Best Parameters)

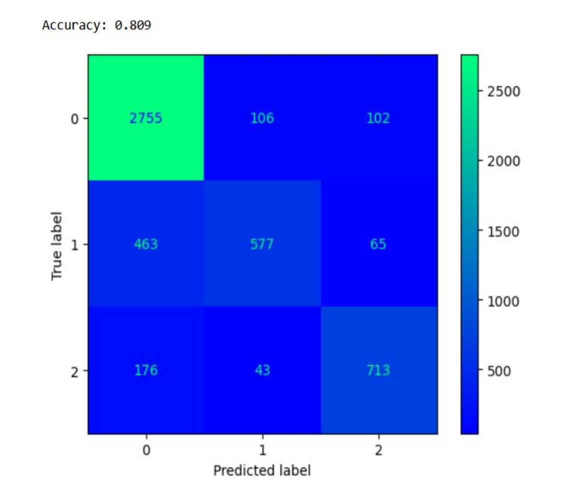


```
              precision    recall  f1-score   support

           0       0.81      0.93      0.87      2963
           1       0.79      0.52      0.63      1105
           2       0.81      0.77      0.79       932

    accuracy                           0.81      5000
   macro avg       0.81      0.74      0.76      5000
weighted avg       0.81      0.81      0.80      5000
```

Classification Report (PCA)



Confusion Matrix (Best Parameters)



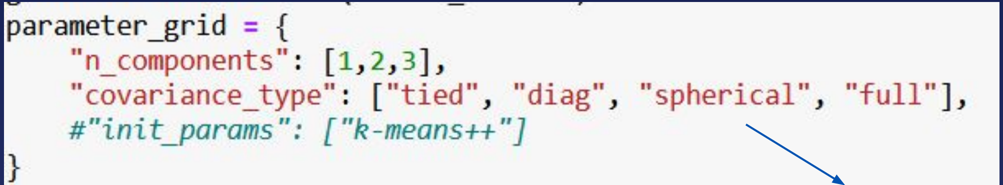
```
Accuracy: 0.809
```

Confusion Matrix (PCA)

# Gaussian Mixture Model

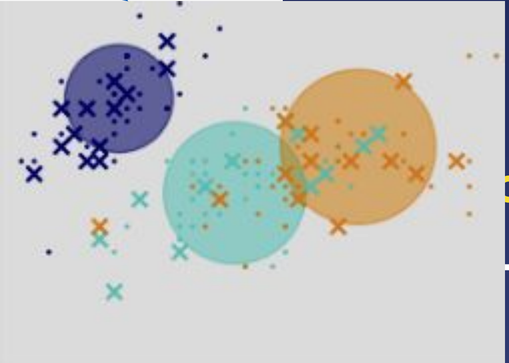
Reason: Unsupervised learning method, large amount of samples may be represented by gaussian distribution.

Parameters Tested:

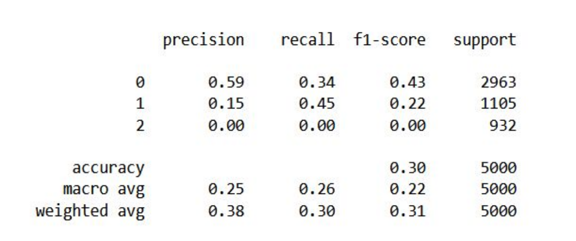


```
parameter_grid = {
    "n_components": [1,2,3],
    "covariance_type": ["tied", "diag", "spherical", "full"],
    #"init_params": ["k-means++"]
}
```

Best Parameters: n_components = 3, covariance_type = spherical

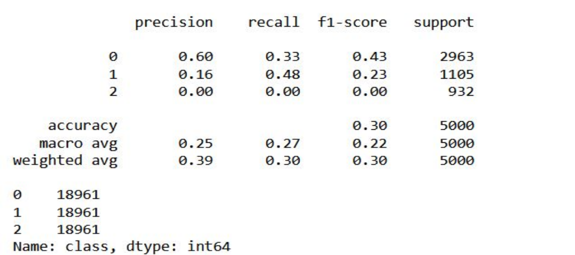


# Gaussian Mixture Model Results



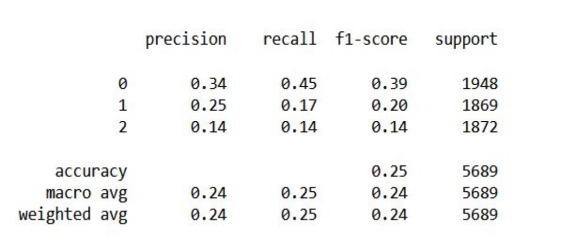
```
              precision    recall  f1-score   support

           0       0.59      0.34      0.43      2963
           1       0.15      0.45      0.22      1105
           2       0.00      0.00      0.00       932

    accuracy                           0.30      5000
   macro avg       0.25      0.26      0.22      5000
weighted avg       0.38      0.30      0.31      5000
```

Classification Report (Best Parameters)



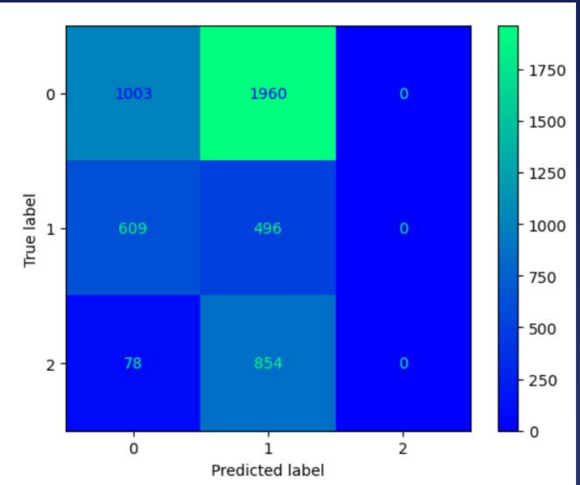
```
              precision    recall  f1-score   support

           0       0.60      0.33      0.43      2963
           1       0.16      0.48      0.23      1105
           2       0.00      0.00      0.00       932

    accuracy                           0.30      5000
   macro avg       0.25      0.27      0.22      5000
weighted avg       0.39      0.30      0.30      5000

0    18961
1    18961
2    18961
Name: class, dtype: int64
```

Classification Report (PCA)

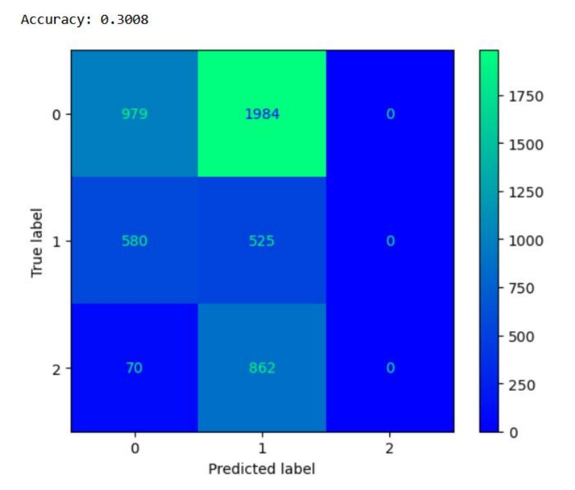


```
              precision    recall  f1-score   support

           0       0.34      0.45      0.39      1948
           1       0.25      0.17      0.20      1869
           2       0.14      0.14      0.14      1872

    accuracy                           0.25      5689
   macro avg       0.24      0.25      0.24      5689
weighted avg       0.24      0.25      0.24      5689
```

Classification Report (Balanced Data)

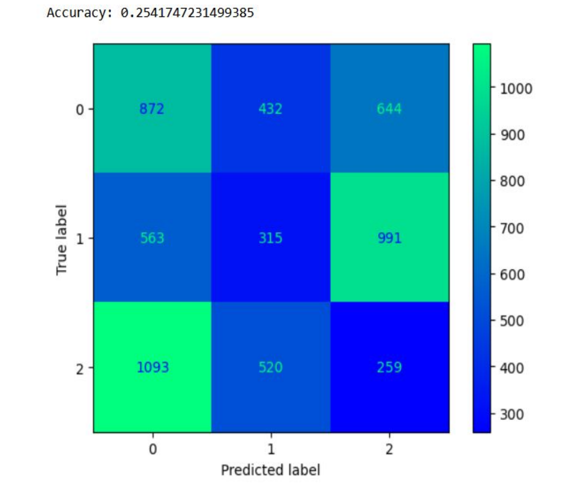


Confusion Matrix (Best Parameters)



```
Accuracy: 0.3008
```

Confusion Matrix (PCA)

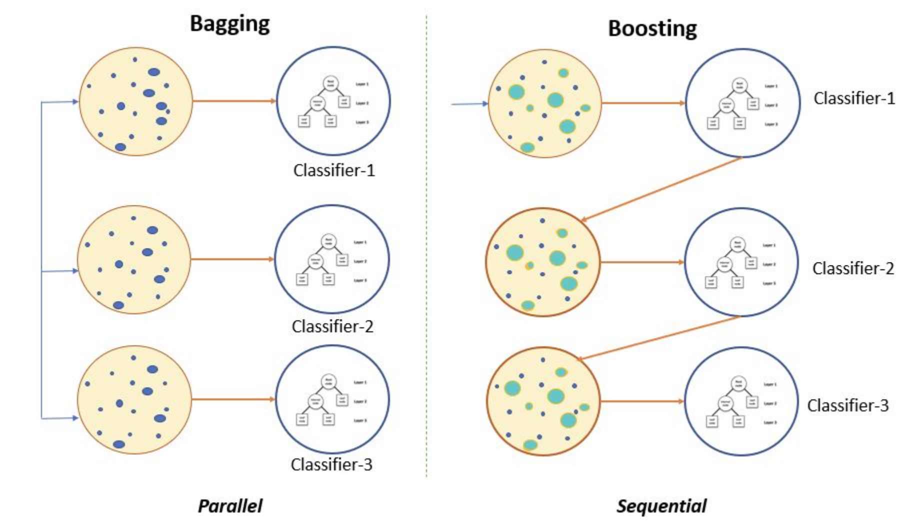


```
Accuracy: 0.2541747231499385
```

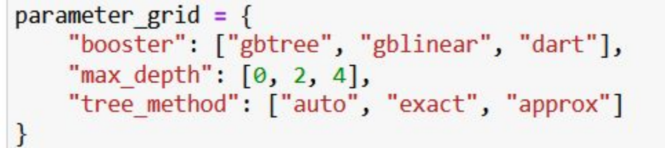
Confusion Matrix (Balanced Data)

# XGBoost Classifier

Reason: Ensemble method, Builds trees sequentially by using boosting rather than bagging, prioritizes accuracy metric.



Parameters Tested:

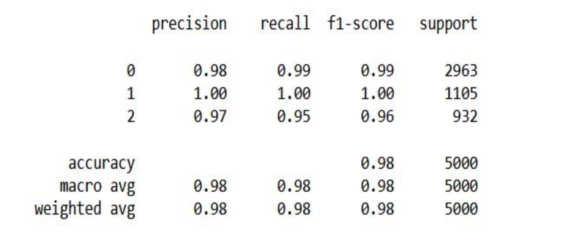


```
parameter_grid = {
    "booster": ["gbtree", "gblinear", "dart"],
    "max_depth": [0, 2, 4],
    "tree_method": ["auto", "exact", "approx"]
}
```

Best Parameters:

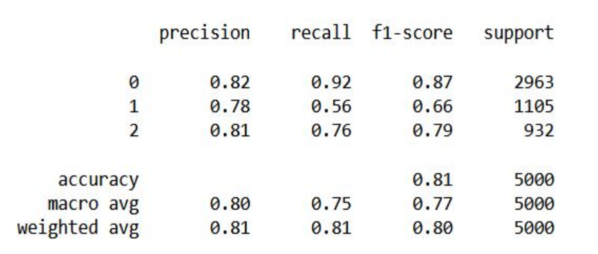
booster = gbtree
max_depth = 0
tree_method = approx

# XGBoost Classifier Results



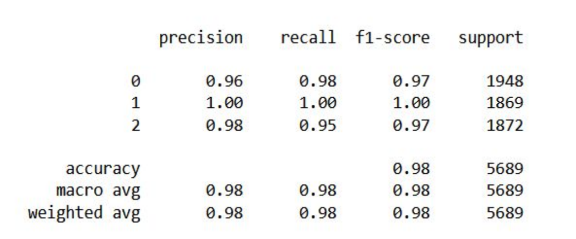
```
              precision    recall  f1-score   support

           0       0.98      0.99      0.99      2963
           1       1.00      1.00      1.00      1105
           2       0.97      0.95      0.96       932

    accuracy                           0.98      5000
   macro avg       0.98      0.98      0.98      5000
weighted avg       0.98      0.98      0.98      5000
```

Classification Report (Best Parameters)



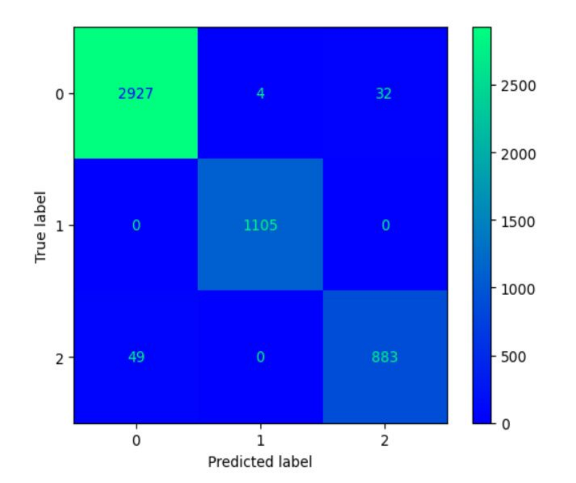
```
              precision    recall  f1-score   support

           0       0.82      0.92      0.87      2963
           1       0.78      0.56      0.66      1105
           2       0.81      0.76      0.79       932

    accuracy                           0.81      5000
   macro avg       0.80      0.75      0.77      5000
weighted avg       0.81      0.81      0.80      5000
```

Classification Report (PCA)

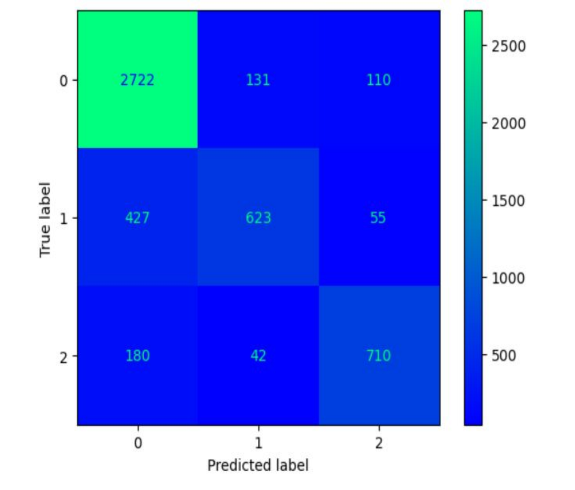


```
              precision    recall  f1-score   support

           0       0.96      0.98      0.97      1948
           1       1.00      1.00      1.00      1869
           2       0.98      0.95      0.97      1872

    accuracy                           0.98      5689
   macro avg       0.98      0.98      0.98      5689
weighted avg       0.98      0.98      0.98      5689
```

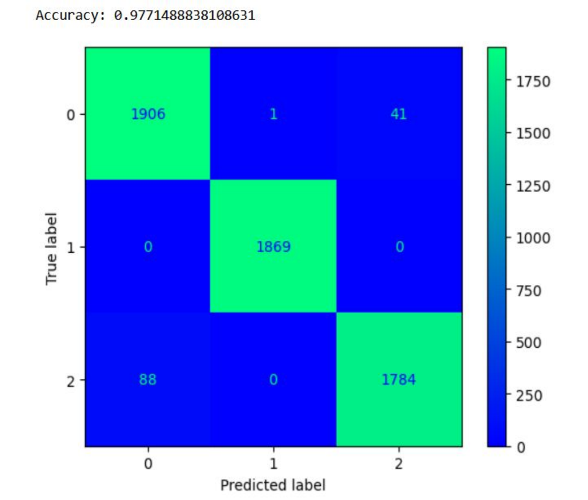
Classification Report (Balanced Data)



Confusion Matrix (Best Parameters)



Confusion Matrix (PCA)



```
Accuracy: 0.9771488838108631
```

Confusion Matrix (Balanced Data)

# References

[1] Kaggle “Stellar Classification Dataset - SDSS17.” Available: https://www.kaggle.com/datasets/fedesoriano/stellar-classification-dataset-sdss17

[2] Australia Telescope National Facility. "Spectra of Stars, Galaxies, and Quasars." Available: https://www.atnf.csiro.au/outreach/education/senior/astrophysics/spectra_astro_types.html.

[3] Bertin, E. & Arnouts, S. "SExtractor: Software for Source Extraction." Available: http://www.astromatic.net/software/sextractor.

[4] The Pan-STARRS Project. "How to Separate Stars and Galaxies." Available: https://outerspace.stsci.edu/display/PANSTARRS/How+to+separate+stars+and+galaxies.

[5] Chen, Tianqi & Guestrin, Carlos. “XGBoost: A Scalable Tree Boosting System” Available: https://arxiv.org/abs/1603.02754

[6] Rashmi, Korlakai Vinayak  & Gilad-Bachrach, Ran. “DART: Dropouts meet Multiple Additive Regression Trees”
Available: https://proceedings.mlr.press/v38/korlakaivinayak15.pdf