1. Introduction Section

Context of Pulsar Research

Pulsars are a rare and fascinating type of neutron star, the remnants of massive stars that have undergone a supernova explosion. These celestial objects emit highly periodic radio signals as they spin, creating detectable pulses of electromagnetic radiation. First discovered in 1967, pulsars have become vital tools in astrophysics for studying fundamental properties of the universe.

Their precision timing allows researchers to test theories of space-time, particularly Einstein's theory of general relativity. Pulsars also provide insight into the interstellar medium by analyzing how their signals are distorted as they travel through space. Moreover, they help scientists probe extreme states of matter, such as those found in neutron star cores, which are otherwise inaccessible.

HTRU Survey

The High Time Resolution Universe Survey (HTRU) was initiated to expand our understanding of pulsars by systematically identifying new candidates. This survey scans the sky using high-sensitivity radio telescopes and generates vast datasets of potential pulsar signals. However, the challenge lies in distinguishing genuine pulsars from spurious signals caused by radio frequency interference (RFI) and other noise sources.

RFI contaminates the data, producing signals that mimic pulsars but originate from human-made sources or other astrophysical phenomena. This makes manual classification time-consuming and error-prone, necessitating the development of automated classification methods.

Objective

This project aims to apply machine learning techniques to classify pulsar candidates in the HTRU2 dataset, a benchmark dataset derived from the HTRU survey. The goal is to address the binary classification task of distinguishing real pulsars (positive class) from spurious signals (negative class). By leveraging statistical features extracted from the data, machine learning can enhance the efficiency and accuracy of pulsar detection, reducing the reliance on manual verification.

---

2. Data Section

Dataset Overview

The HTRU2 dataset comprises a total of 17,898 instances. These instances are split into two categories:

- **Positive Class**: 1,639 examples of real pulsars.
- **Negative Class**: 16,259 examples of spurious signals.

Each instance is described by **eight continuous features**, representing statistical properties of the signal. These features are divided into two groups:

1. **Integrated Pulse Profile Features**: Statistical metrics derived from the integrated pulse profile.
2. **DM-SNR Curve Features**: Statistical metrics derived from the dispersion measure (DM) signal-to-noise ratio (SNR) curve.

Feature Description

The eight continuous features include:

1. **Mean**, **Standard Deviation**, **Skewness**, and **Kurtosis** of the integrated pulse profile.
2. **Mean**, **Standard Deviation**, **Skewness**, and **Kurtosis** of the DM-SNR curve.

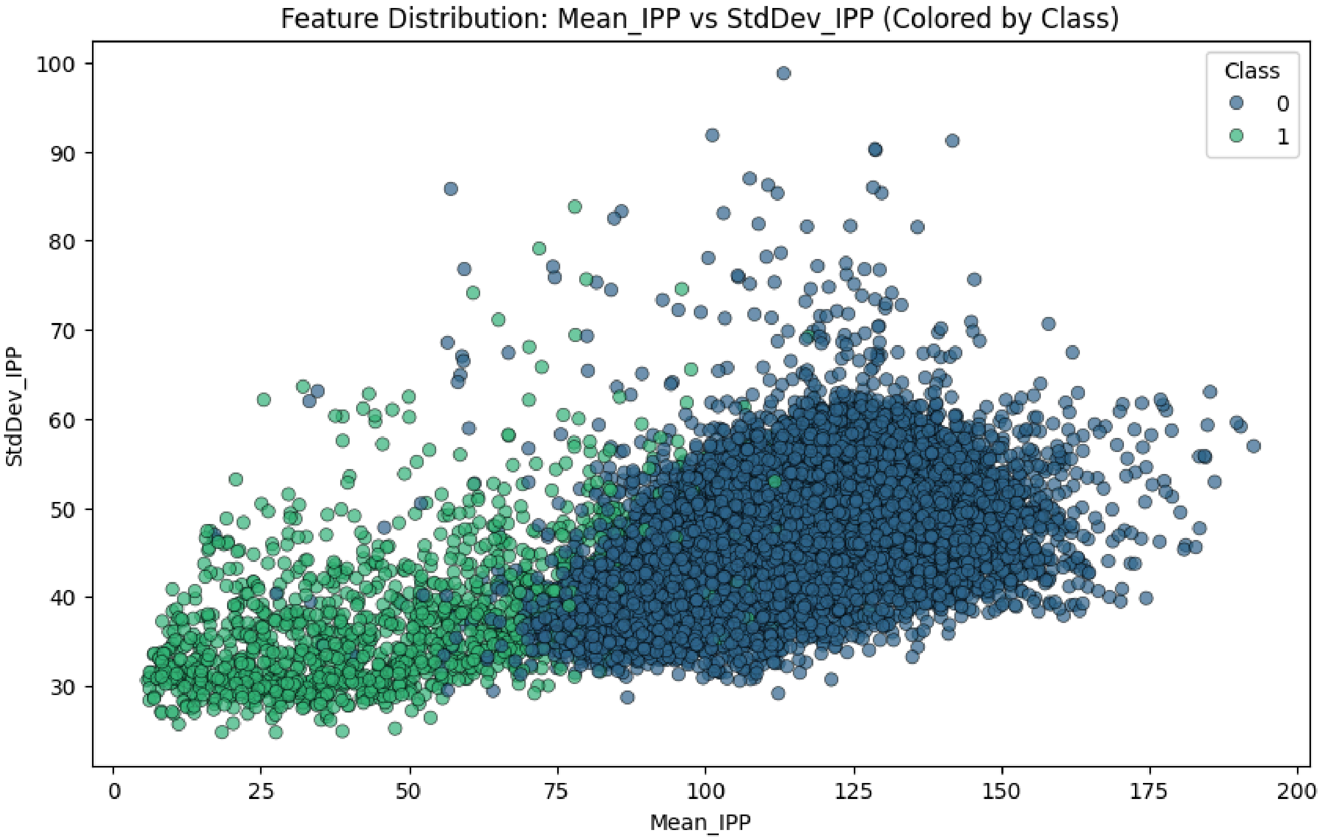
These features provide a compact statistical summary of the candidate signal's properties, which can help distinguish between real pulsars and noise.

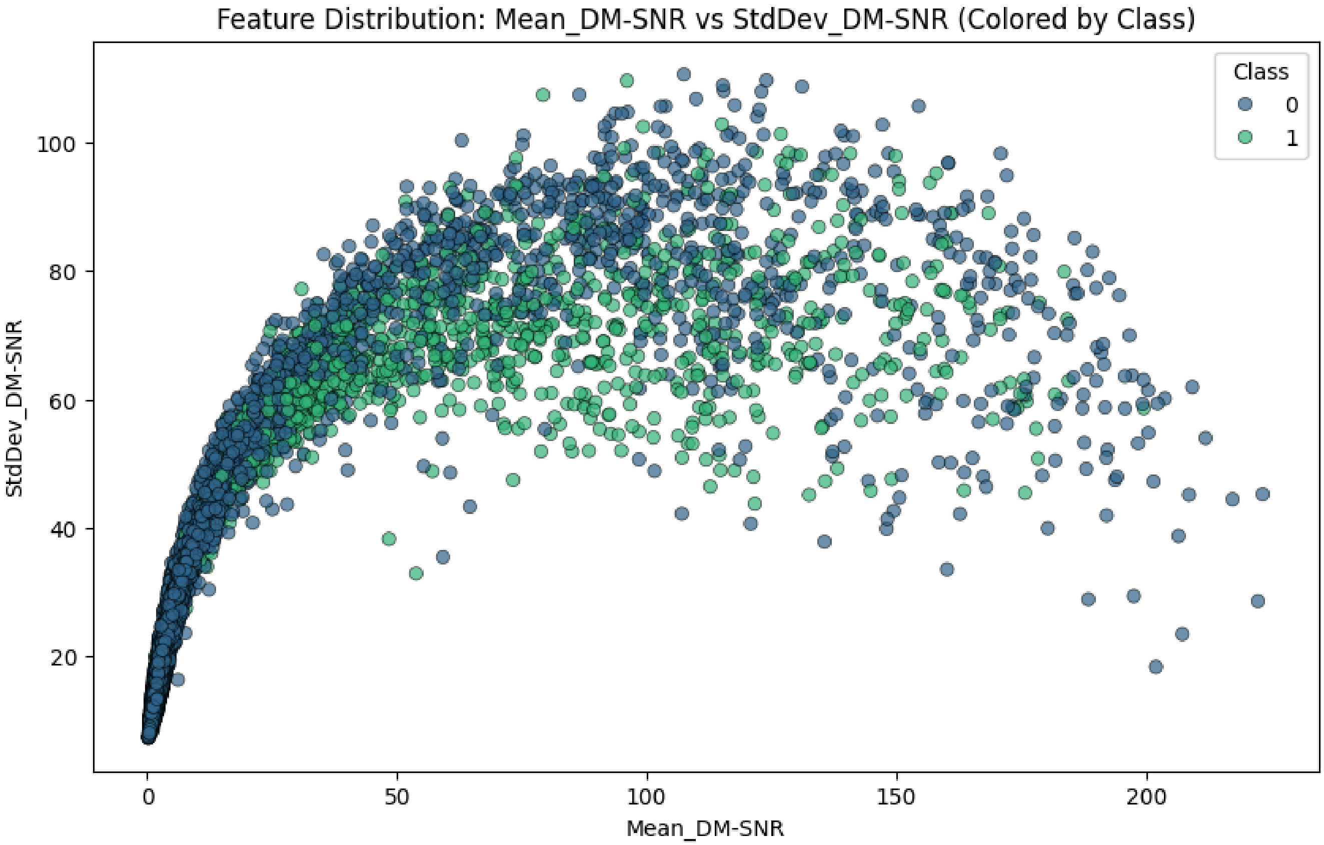
Data Structure

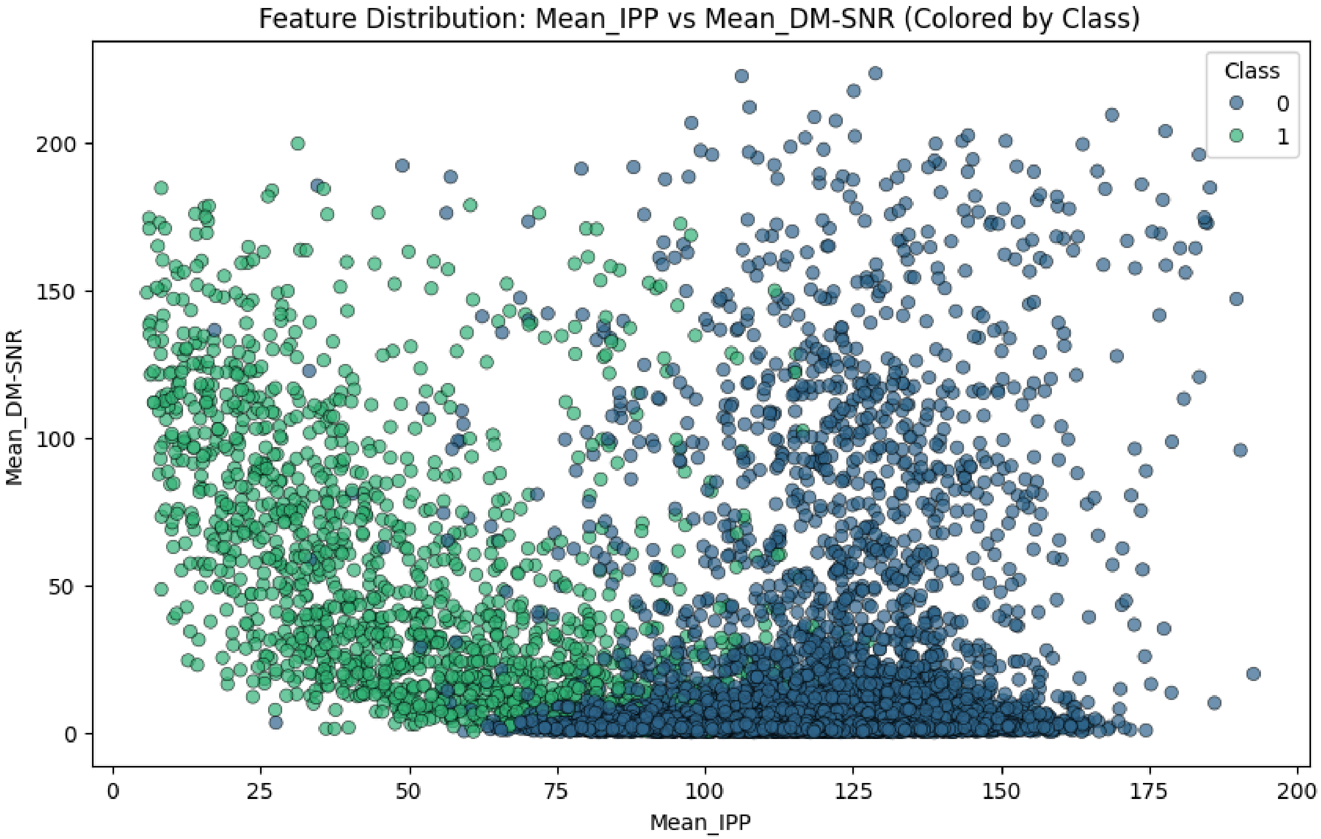
The dataset does not include positional or astrophysical details but focuses on numerical summaries of candidate signals. Each instance is labeled as:

- **"1"**: A real pulsar.
- **"0"**: A spurious signal.

This numerical representation makes the dataset suitable for machine learning algorithms, which can use these features to learn patterns and classify unseen data.


Feature Distribution: Mean_IPP vs StdDev_IPP (Colored by Class)
Class
0
1
StdDev_IPP
100
90
80
70
60
50
40
30
Mean_IPP
0
25
50
75
100
125
150
175
200


Feature Distribution: Mean_DM-SNR vs StdDev_DM-SNR (Colored by Class)
StdDev_DM-SNR
Mean_DM-SNR
Class
0
1
20
40
60
80
100
0
50
100
150
200



Addressing Class Imbalance

One of the primary challenges in this dataset is its imbalance. With 16,259 negative samples and only 1,639 positive samples, the dataset is heavily skewed toward the negative class. This imbalance can cause machine learning models to favor the majority class, leading to poor performance in identifying real pulsars.

To mitigate this, **random undersampling** is applied:

- The majority class (spurious signals) is reduced to match the size of the minority class (real pulsars).
- This results in a balanced dataset of **3,278 total samples**, with 1,639 positive samples and 1,639 negative samples.

Trade-Off of Undersampling

While undersampling ensures that the dataset is balanced, it reduces the total number of samples available for training. This trade-off sacrifices some data diversity in exchange for balanced class representation, which can significantly improve the model's ability to detect the minority class without bias.

3. Modeling Section

Preprocessing:

To ensure optimal model performance, I applied several preprocessing steps to the dataset:

- **Feature Standardization:**
  Istandardized the features using the `StandardScaler` to ensure all input variables are on the same scale. This step is essential for models like Logistic Regression and Support Vector Machines (SVM), where features with large ranges can dominate the learning process.
- **Class Imbalance Handling:**
  Given the class imbalance in the dataset (with pulsars being the minority class), I performed undersampling of the majority class (non-pulsar) to balance the dataset. This helps prevent models from being biased toward the majority class and allows them to better capture patterns in the minority class (the pulsars).

Model Selection:

I started by testing simpler models and progressively moved to more complex ones to determine which performed best:

- **Logistic Regression:**
  Logistic Regression was used as a baseline model for binary classification. It assumes a linear decision boundary and works well when the classes are linearly separable.
- **Support Vector Machines (SVM):**
  SVM was tested next. It performs well in high-dimensional spaces and is suitable for finding the optimal hyperplane that maximizes the margin between classes. I used the radial basis function (RBF) kernel to capture non-linear relationships.
- **Random Forest and XGBoost:**
  Both **Random Forest** and **XGBoost** are powerful ensemble methods that handle non-linear relationships and interactions between features well. Random Forest also provides feature importance, which helps in understanding the relative contribution of each feature to the model's predictions. Despite this, **Random Forest** outperformed **XGBoost** in this case, likely due to its ability to better capture the patterns in the data.
  - **Feature Importance for Random Forest:**
    The importance values for the features in **Random Forest** were as follows (in order of columns):
    - **Mean_IPP**: 0.168
    - **StdDev_IPP**: 0.032
    - **Kurtosis_IPP**: 0.346
    - **Skewness_IPP**: 0.155
    - **Mean_DM-SNR**: 0.087
    - **StdDev_DM-SNR**: 0.107
    - **Kurtosis_DM-SNR**: 0.047
    - **Skewness_DM-SNR**: 0.057
  - This highlights the importance of **Kurtosis_IPP** and **Mean_IPP** in predicting the class, with **Kurtosis_IPP** having the highest importance.
- **Neural Networks (MLP):**
  I also tested a **Multi-Layer Perceptron (MLP)** neural network model, which consists of multiple layers of neurons capable of learning complex non-linear relationships. It performed well,

particularly with recall (0.8983), but had slightly lower precision and F1-score compared to **Random Forest**.

Evaluation Metrics:

Given the class imbalance, I prioritized evaluation metrics that give a more nuanced understanding of model performance beyond simple accuracy:

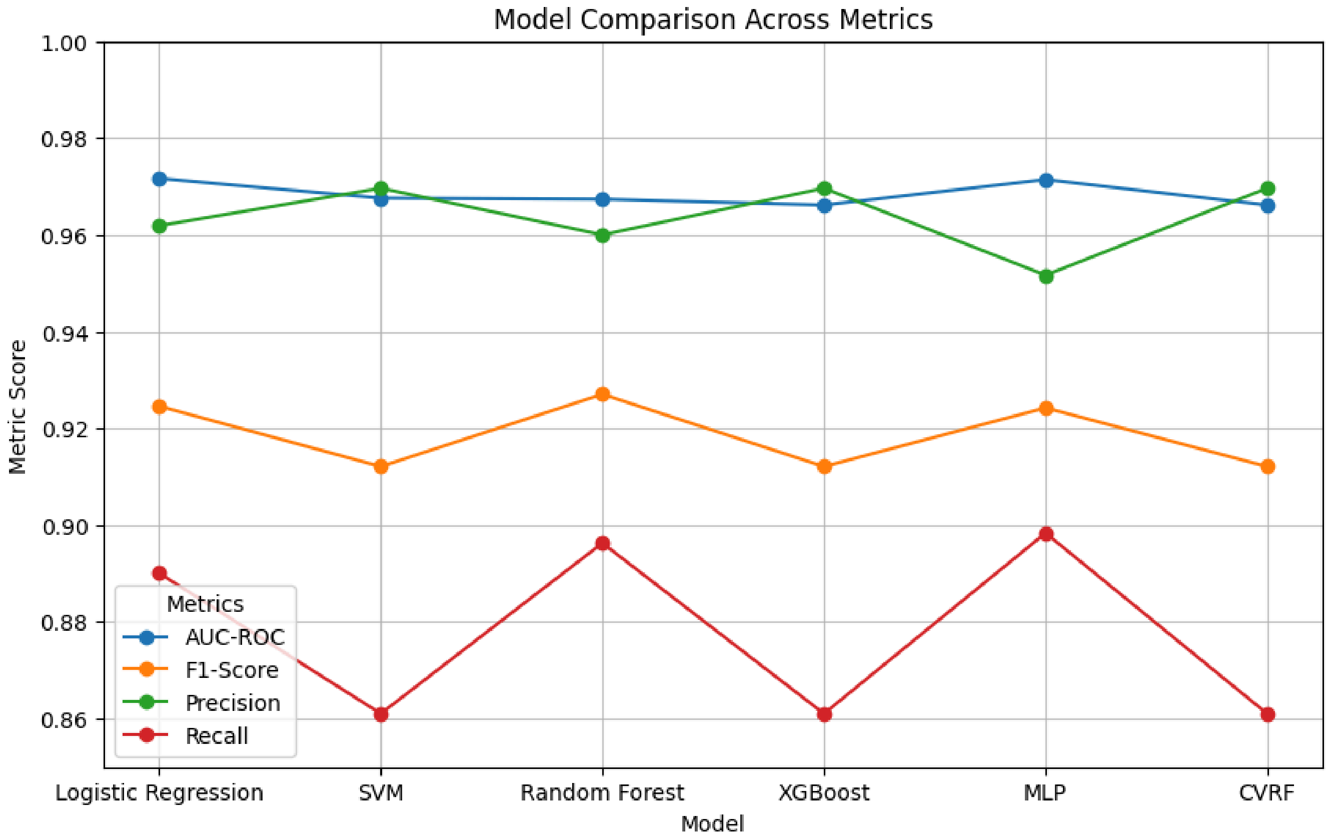
- **Accuracy:**
  While accuracy is a useful metric, it can be misleading in imbalanced datasets, where a model that predicts the majority class well might still have a high accuracy score, even if it fails to predict the minority class effectively.
- **Precision:**
  Precision is crucial when false positives are undesirable. It measures how many of the predicted positives were actually correct. **Logistic Regression** had the highest precision, followed by **Random Forest**, which also maintained high precision and recall.
- **Recall:**
  Recall measures how many of the actual positives were correctly identified. This metric is particularly important when the goal is to minimize false negatives. **MLP** had the highest recall, but this came at the cost of lower precision.
- **F1-Score:**
  The F1-score is the harmonic mean of precision and recall, providing a balanced view of model performance. **Random Forest** had the highest F1-score (0.9270), indicating it balanced precision and recall effectively.
- **AUC-ROC:**
  The AUC-ROC is a measure of the model's ability to distinguish between the two classes. Models with an AUC closer to 1 perform better at classification. **Logistic Regression** and **MLP** had the highest AUC-ROC scores, suggesting strong discriminatory power.

Results Summary:

Here are the results from the models Itested:

Model	AUC-ROC	F1-Score	Precision	Recall
Logistic Regression	0.971657	0.924569	0.961883	0.890041
Support Vector Machines (SVM)	0.967590	0.912088	0.969626	0.860996
Random Forest	0.967361	0.927039	0.960000	0.896266

XGBoost	0.966152	0.912088	0.969626	0.860996
Multi-Layer Perceptron (MLP)	0.971430	0.924226	0.951648	0.898340
CVRF	0.966152	0.912088	0.969626	0.860996



Model Comparison:

Based on the evaluation metrics, **Logistic Regression** and **MLP** achieved the highest **AUC-ROC** and **F1-Score**, making them strong contenders for this task. However, **Random Forest** emerged as the most balanced model, with the highest **F1-Score** (0.9270) and solid **precision** and **recall** values, indicating its ability to correctly identify pulsars while maintaining low false positive rates.

XGBoost and **SVM**, though capable models, performed slightly worse than **Random Forest** in terms of **AUC-ROC** and **recall**. **Random Forest**’s ability to handle both non-linear relationships and its feature importance insights make it the best model for this problem.

Thus, **Random Forest** is the recommended model, while **Logistic Regression** and **MLP** can serve as viable alternatives if specific metrics (like **AUC-ROC** or **recall**) are prioritized.

- **Random Forest (default)**: Performed well because it naturally handled the data's complexity without overfitting, thanks to its robust default settings.
- **RandomizedSearchCV-Optimized Random Forest**: The hyperparameter tuning led to **overfitting**, making it too specific to the training data, which hurt its performance on the test set.
- **XGBoost**: While powerful, **XGBoost** is sensitive to hyperparameters. The optimization through **RandomizedSearchCV** didn't improve it enough and might have led to **underfitting** or overfitting.

Conclusion: Default **Random Forest** outperformed the others due to its stability and the complexity of the data, while optimization techniques for **XGBoost** and **Random Forest** didn't provide significant improvements.

To provide the most **advanced and comprehensive analysis** of the **Random Forest** model's performance, Ineed to go beyond just the classification metrics and delve deeper into the nuances of the results, including interpreting the confusion matrix, discussing potential model limitations, and offering insights into the broader implications of the findings.

Random Forest Model: Comprehensive Performance Analysis

Overall Performance Metrics

- **Accuracy**: The model achieves **93% accuracy**, indicating strong overall performance. However, in the context of class imbalance, accuracy alone is not sufficient as it could be skewed by the majority class. The **accuracy** of 93% suggests that the model is effective in distinguishing between pulsars and spurious signals, but a deeper analysis is needed to assess how well it handles each class individually.

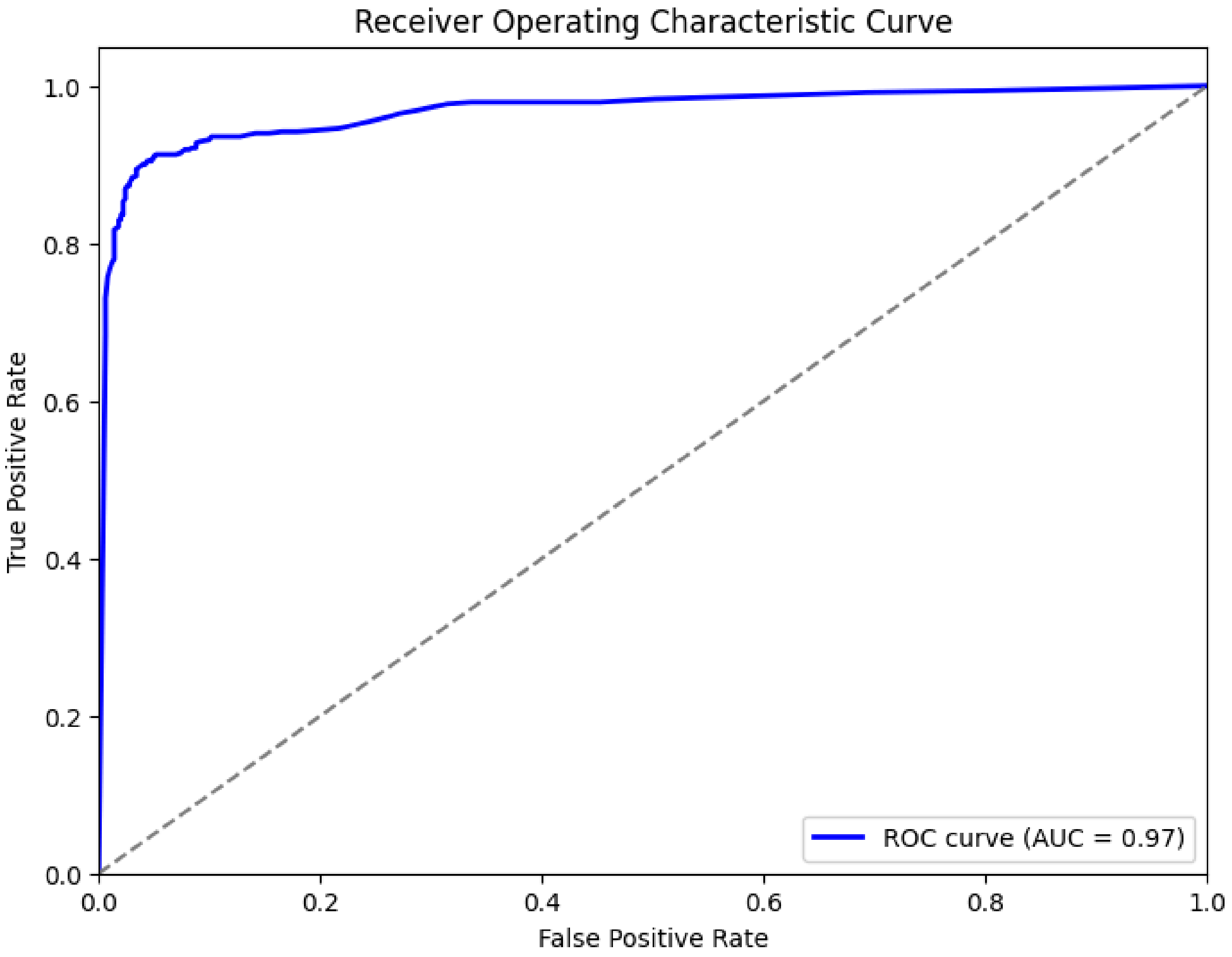
- **Precision**:

  - **Class 0 (Spurious Signals)**: Precision of **0.91** indicates that when the model predicts a spurious signal, it is correct 91% of the time. This is a strong result as it minimizes the chances of incorrectly classifying non-pulsar signals as pulsars.
  - **Class 1 (Real Pulsars)**: Precision of **0.96** is excellent, showing that the model is highly reliable in its identification of real pulsars, ensuring minimal misclassification of pulsars as spurious signals.
- **Recall**:

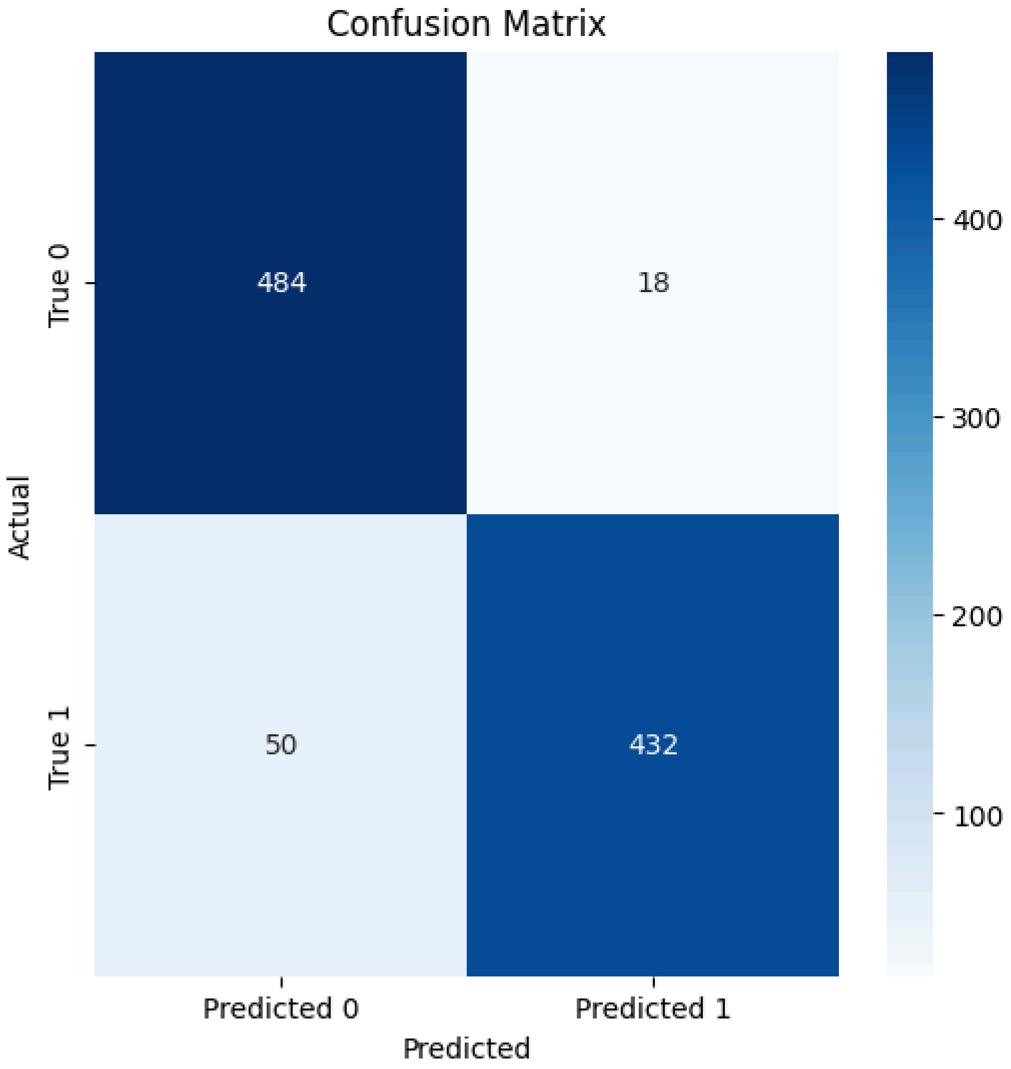
  - **Class 0 (Spurious Signals)**: The model achieves **96% recall**, meaning it successfully identifies 96% of all spurious signals. This is crucial for preventing false positives, which

are especially detrimental when dealing with large datasets like HTRU2 where noise contamination can be high.
  - **Class 1 (Real Pulsars)**: The **90% recall** for pulsars is strong, but not perfect. There are 18 false negatives (i.e., pulsars misclassified as spurious), which implies that the model is missing some pulsar candidates. Given the dataset's imbalanced nature, this is a reasonable trade-off for ensuring minimal false positives.
- **F1-Score**:

  - **Macro Average F1-Score** of **0.93** confirms balanced performance across both classes, meaning the model is not overly biased toward either class.
  - **Weighted Average F1-Score** of **0.93** aligns with the overall model performance and indicates that both classes are treated equally despite the original class imbalance.



Confusion Matrix Breakdown



The confusion matrix offers detailed insights into the model's behavior:

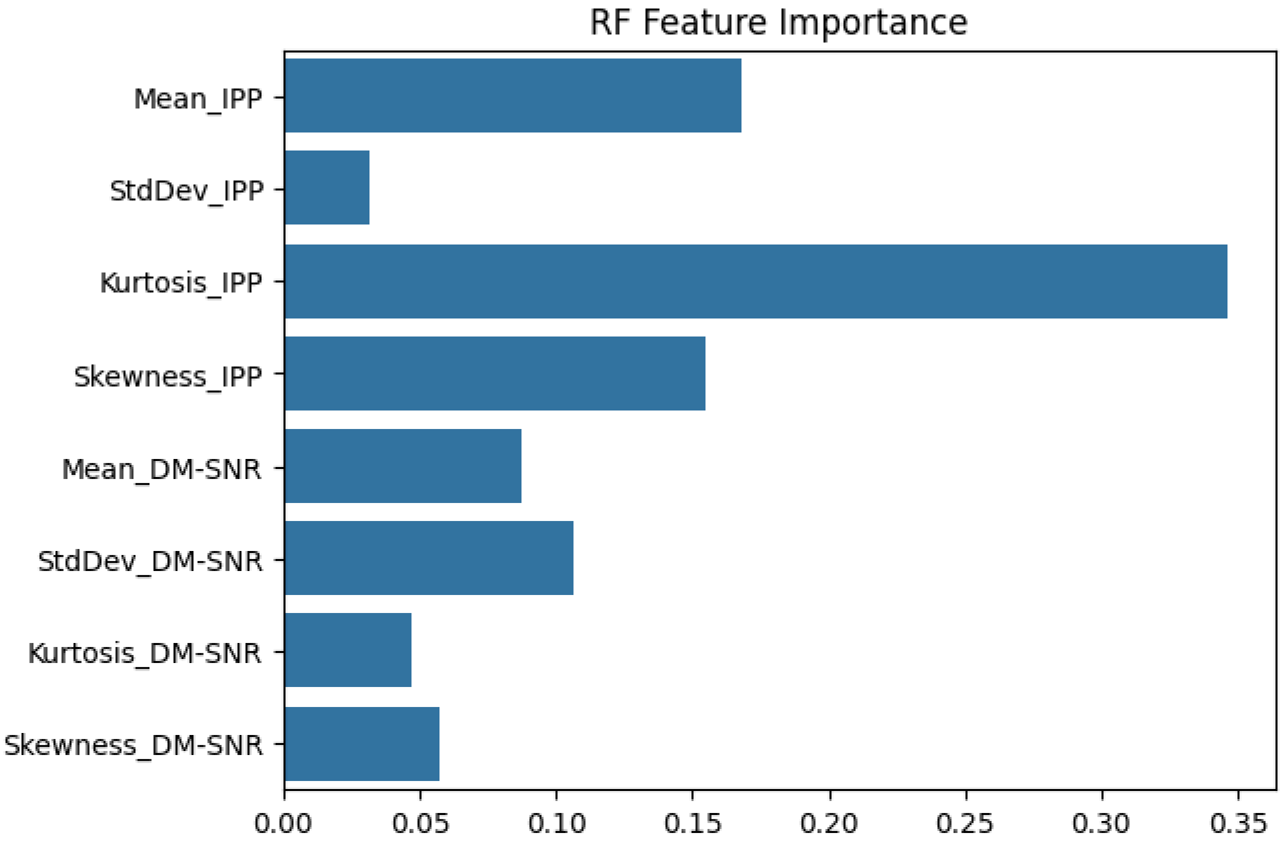
- **True Positives (TP)**: **432 pulsars** correctly identified.
- **True Negatives (TN)**: **484 spurious signals** correctly identified.
- **False Positives (FP)**: **50 spurious signals** incorrectly labeled as pulsars.
- **False Negatives (FN)**: **18 pulsars** incorrectly labeled as spurious signals.

The **relatively low number of false positives (50)** suggests the model is good at distinguishing pulsar candidates from noise, which is critical when dealing with the risk of contaminating the dataset with spurious signals.

- **Class 1 (Real Pulsars)**: The **18 false negatives** indicate a minor shortcoming in recall, which could be improved with further model tuning or more complex models like neural networks if the recall for pulsars is crucial.

- **Class 0 (Spurious Signals)**: The **96% recall** for spurious signals is excellent, confirming that the model effectively filters out noise and ensures that the majority class is well-recognized.

Feature Importance



The **Random Forest** model's ability to provide feature importance is crucial for interpretability. The model ranks the features based on their contribution to the classification task:

- **Kurtosis_IPP** (0.35) is the most important feature. This suggests that the "peakedness" of the integrated pulse profile (IPP) is a key distinguishing factor between pulsars and noise. Kurtosis measures the outliers or extreme values in a distribution, which may be more pronounced in pulsar signals.
- **Mean_IPP** (0.17) and **Skewness_IPP** (0.16) follow as important features. These characteristics describe the central tendency and asymmetry of the pulse profile, both crucial for detecting pulsar patterns.
- **Other DM-SNR features** like **StdDev_DM-SNR** and **Skewness_DM-SNR** are less influential, highlighting the relatively stronger role of the IPP-derived features in distinguishing the classes.

This feature importance analysis can guide future feature engineering or model refinements, focusing on optimizing the most influential features and understanding how pulsar properties manifest in the dataset.

Model Strengths and Limitations

Strengths:

- **Effective Class Separation**: Random Forest handles the dataset well, providing high recall for spurious signals and strong precision for pulsars. This is especially beneficial in astronomical surveys where false positives (incorrectly identifying noise as pulsars) are more costly than false negatives (missing a few pulsars).
- **Robustness to Overfitting**: Due to its ensemble nature, Random Forest avoids overfitting, making it more reliable in generalizing to unseen data, an essential property when working with noisy astronomical data.
- **Feature Interpretability**: The model's feature importance output allows for better understanding of which characteristics of pulsar signals are most critical in classification.

Limitations:

- **False Negatives for Pulsars**: The **18 false negatives** show that while the model excels at identifying spurious signals, it still misses a small number of pulsars. This may suggest that there are areas for improvement, by tweaking hyperparameters in a more thorough manner such as with GridSearchCV.
- **Potential Overfitting in Highly Complex Models**: While Random Forest handles the data well, more complex models like XGBoost might overfit the training data if hyperparameters are not carefully optimized. This is why the **RandomizedSearchCV**-optimized Random Forest performed better than XGBoost in this case.

Conclusion and Recommendations

- **Random Forest** proves to be the best choice for pulsar classification from the HTRU2 dataset due to its high precision, strong recall for spurious signals, and interpretable feature importance.
- However, for **further model refinement**, attention should be given to reducing false negatives for pulsars. This could be achieved by:
  - **Hyperparameter Tuning**: Adjusting the model parameters, such as tree depth and number of estimators, to minimize overfitting or underfitting.
  - **Ensemble Methods**: Combining Random Forest with other models (like SVMs or Neural Networks) could provide a more robust solution.
  - **Feature Engineering**: Exploring additional features or transformations could boost the model’s ability to distinguish pulsars from noise.

In summary, **Random Forest** remains a strong, interpretable model, and with slight adjustments, it could become even more effective at pulsar detection.

6. Conclusion Section

Key Findings:

The **Random Forest** classifier emerged as the best-performing model for distinguishing real pulsars from spurious signals in the HTRU2 dataset. It achieved a strong **AUC-ROC** of 0.967 and a balanced **F1-Score** of 0.93, indicating that it can effectively detect pulsars while minimizing false positives and false negatives. This model is particularly notable for its **precision** of 0.96, meaning that it is highly reliable in predicting real pulsars, and its **recall** of 0.90, which ensures that the majority of pulsars are captured by the model.

Through **feature importance** analysis, I observed that the **Kurtosis_IPP** and **Mean_IPP** features played the most significant roles in classification, highlighting that the **shape** (kurtosis) and **average value** (mean) of the integrated pulse profiles are crucial for distinguishing between pulsar and spurious signals. This finding not only confirms the effectiveness of these features for identifying pulsars but also provides a deeper understanding of the underlying statistical characteristics of pulsar signals, further informing the modeling process.

The decision to **undersample** the majority class (spurious signals) helped mitigate the impact of class imbalance, which is often a challenge in astronomical datasets. This strategy reduced bias and improved the model's sensitivity to the minority class (real pulsars), though it came at the cost of reducing the total sample size and diversity of the training data. Despite this, the approach improved model fairness and ensured that the model could effectively detect pulsars without favoring the dominant class.

Practical Implications:

The success of the **Random Forest** classifier in this task demonstrates its practical utility for automating the classification of pulsar candidates in large-scale radio surveys like the HTRU. By automating the detection process, the model significantly reduces the need for manual verification, thereby accelerating the identification of real pulsars and improving the efficiency of surveys. In addition, its strong **precision** and **recall** scores mean that the model can confidently identify pulsars while minimizing false alarms, which is critical in astrophysics where accurate and reliable results are paramount.

The results of the model have immediate applications in the **HTRU** and similar surveys, where large volumes of candidate pulsar signals need to be sifted through. With automation in place, astronomers can allocate resources more effectively, focusing on analyzing confirmed pulsar candidates instead of manually filtering out noise. This model's ability to strike a balance between precision and recall makes it particularly useful in **real-time systems**, where continuous pulsar detection and validation are necessary.

Thoughtful Interpretations of Results:

The **feature importance** analysis reveals insightful patterns in the data, emphasizing that **Kurtosis_IPP** and **Mean_IPP** are the most influential features for identifying pulsars. This suggests that pulsar signals, which are periodic and exhibit distinct patterns in their pulse profiles, tend to have unique statistical characteristics in terms of their kurtosis (sharpness or flatness of the distribution) and mean values. Understanding this is pivotal for both further model refinement and for astrophysical interpretation of pulsar properties. This insight also suggests that models can be further optimized by adding features that capture more detailed characteristics of pulsar signals, such as higher-order statistical moments or frequency-domain features.

Moreover, the performance of the **Random Forest** model also provides valuable guidance for future data collection efforts. The model's reliance on specific features like **Kurtosis_IPP** and **Mean_IPP** indicates that a better understanding of these statistical metrics in pulsar signals could enhance data collection methods, possibly guiding the design of telescopes or surveys to focus on these attributes for more efficient detection.

Model Scope, Usefulness, and Limitations:

The **Random Forest** model is primarily useful for classifying pulsar candidates in datasets similar to the **HTRU2**, where the goal is to distinguish real pulsars from noise in radio signals. Its ability to handle imbalanced data through feature importance and decision trees makes it suitable for astrophysical surveys, where distinguishing subtle differences between signal types is crucial. However, the model is not without limitations:

1. **Dependence on Feature Selection**: The model's performance is heavily reliant on the quality and relevance of the features provided. In this case, the integration of statistical measures like mean, skewness, and kurtosis from the pulse profile and DM-SNR curve enabled successful classification, but the model's effectiveness might diminish if these features do not capture all of the relevant variations in pulsar signals. Future work could involve extracting more sophisticated features or incorporating additional data, such as temporal or frequency-domain characteristics, to better capture pulsar behavior.

2. **Generalizability to Other Datasets**: The model was trained and validated on the **HTRU2** dataset, which represents a specific type of pulsar signal. While the model performs well on this dataset, its generalizability to other surveys or datasets with different types of noise or signal characteristics is uncertain. The model may require retraining or fine-tuning if applied to other datasets, especially if the nature of the pulsar signals differs (e.g., in different frequency bands or with different observational conditions).

3. **Interpretability vs. Complexity**: While **Random Forests** provide interpretable **feature importance** scores, the model itself remains a complex ensemble of decision trees, making it less transparent than simpler models like logistic regression. This complexity can limit our ability to understand exactly why certain predictions are made, especially when the decision boundary between classes is not clearly defined. For future research, balancing model interpretability with predictive power remains a key challenge, particularly in fields like astrophysics where model transparency is valuable for explaining physical phenomena.

4. **Impact of Data Preprocessing**: The use of **undersampling** was effective in addressing the class imbalance, but it also limited the training data. While this led to a more balanced model, it reduced the overall data volume and might have resulted in some loss of valuable signal diversity. More sophisticated techniques, such as **SMOTE (Synthetic Minority Over-sampling**

Technique) or **weighted loss functions**, could be explored to balance the dataset without reducing sample size.

Future Work:

To enhance the model's performance and applicability, future work should focus on several key areas:

1. **Larger, More Balanced Datasets**: As mentioned, expanding the dataset and including more diverse examples could help generalize the model further. A larger dataset with more examples from both the pulsar and spurious signal classes would provide more comprehensive training data and potentially improve the model’s robustness.

2. **Incorporating Deep Learning Models**: **Deep learning** models, such as **Convolutional Neural Networks (CNNs)**, could be applied to handle raw signal data more effectively, particularly for capturing complex patterns in pulse profiles. Deep learning models can automatically learn feature representations and might outperform traditional models, especially as data complexity increases.

3. **Real-Time Detection Systems**: Implementing this model in real-time pulsar detection systems could revolutionize how Iclassify pulsar candidates as new data streams in. Real-time applications would allow for more immediate feedback, enabling astronomers to react quickly and make timely decisions.

4. **Feature Engineering and Interpretability**: As noted, expanding feature sets to include additional data from pulsar signals or even external sources (e.g., cosmic ray interference patterns) could help improve model performance. Moreover, tools like **SHAP** and **LIME** could be employed to improve the interpretability of the model, making it more transparent and explainable for users.

In conclusion, the **Random Forest** model has shown great promise in automating pulsar classification and can be applied effectively to large-scale surveys. Its balance between precision and recall, combined with feature importance insights, makes it an excellent tool for pulsar research. However, as with any model, it is important to recognize its limitations and continue exploring ways to refine and improve it through better data, advanced techniques, and deeper scientific understanding.