

Sure, here's a README document for your repository named "Star-Classification-CNN":

Star-Classification-CNN

This repository contains a project that implements a Convolutional Neural Network (CNN) to classify exoplanet host stars using the Kepler dataset. The project involves data loading, model creation, training, evaluation, and performance tuning.

Table of Contents

- [Overview](#overview)
- [Dataset](#dataset)
- [Project Structure](#project-structure)
- [Usage](#usage)
- [Results](#results)
- [Conclusion](#conclusion)
- [Acknowledgements](#acknowledgements)

Overview

The objective of this project is to replace the fully connected neural network model with a convolutional neural network to classify exoplanet host stars. The tasks accomplished include:

1. Loading and splitting the data into input and output sets.
2. Plotting examples of the two classes of stars.
3. Creating a CNN model.
4. Training the model.
5. Evaluating the model on validation and test datasets.
6. Comparing predicted classes with observed classes.
7. Tuning hyperparameters to improve model performance.
8. Applying data augmentation to see its effect on performance.

Dataset

The dataset used in this project is the Kepler dataset, which contains information about stars and their classification as exoplanet host stars or not.

Project Structure

...

Star-Classification-CNN/

```
|— data/
|   |— kepler_train.csv
|   |— kepler_test.csv
|— notebooks/
```

```
|  |— star_classification_cnn.ipynb
|  |— README.md
|  |— requirements.txt
|  \
\  \
\  \
```

Usage

To run this project, follow these steps:

1. Clone this repository:

```
` `` bash
git clone https://github.com/yourusername/Star-Classification-CNN.git
` ``
```

2. Navigate to the project directory:

```
` `` bash
cd Star-Classification-CNN
` ``
```

3. Install the required dependencies:

```
` `` bash
pip install -r requirements.txt
` ``
```

4. Open the Jupyter notebook:

```
` `` bash
jupyter notebook notebooks/star_classification_cnn.ipynb
` ``
```

Results

The model's performance is evaluated using precision and recall metrics. The results show how well the CNN model classifies exoplanet host stars compared to a fully connected neural network. Hyperparameter tuning and data augmentation further help in improving the model's accuracy.

Conclusion

The CNN model provides a robust approach to classifying exoplanet host stars. Through various stages of model building, training, and evaluation, the project demonstrates the effectiveness of convolutional neural networks in astrophysical data classification tasks.

Acknowledgements

This project is based on the Kepler dataset and inspired by coursework on neural networks and deep learning. Special thanks to the instructors and peers for their guidance and support.
