Venomous Snake Classification Case Study Rubric
DS4002 – Spring 2025 – Cole Anderson
Due: 28 April 2025
Submission Format: Link to GitHub Repository uploaded to Canvas

Individual Assignment

Why am I doing this? This case study poses a challenge to utilize your data science and machine learning capabilities on a real-world issue: determining whether a snake is venomous or not using a CNN (Convolutional Neural Network) and images. As you complete this exercise, you will become more familiar with how to input image data sets, design CNN architectures, and assess model performance through important metrics like accuracy, recall, precision, and F1-score.

What am I going to do? You will work with a dataset of snake images organized by venomous and non-venomous classes. Using this data, you will preprocess the images, build a convolutional neural network (CNN) model using MobileNetV2, and train the model to classify the snakes. You will evaluate your model's performance using a confusion matrix, accuracy, precision, recall, and F1-score, aiming for at least 90% accuracy.

Final Deliverables Include:

- The image data organized by class and train/test folders.
- Well-documented code/scripts for preprocessing, training, and evaluating the model.
- A confusion matrix and final model performance metrics.
- A GitHub repository containing all project materials and instructions.

Spec Category	Spec Details
Formatting	One GitHub repository, submitted via Canvas link. Must contain: ● README.md ● LICENSE.md ● SCRIPTS folder ● DATA folder ● OUTPUT folder ● References

README.md	• The goal here is to explain your project clearly to new users, and new audiences, so that they can understand what is going on. • It's a brief intro that shows what you have created, produced, and how to reproduce it!
LICENSE.md	• MIT License (or equivalent)
DATA folder	• Organized train/test folders with venomous and non-venomous subfolders. • Can submit a link to another repo, or .zip file of images, containing the dataset.
OUTPUT folder	• Store EDA results as well as final model evaluation outputs. • Include confusion matrix, precision, recall, and F1-score summaries. • Use clear and distinctive filenames.
SCRIPTS folder	• Contains clean, well-commented Python notebooks. • Separate scripts for preprocessing and CNN modeling. • Comments should explain the logic and flow for new users to understand what is going on. Be detailed! It never hurts to tell too much.
References	• Provide a list of helpful and used resources. • Format them in IEEE style.