<u>Assignment-04: Naïve Recursive Bayesian Classification</u>

Zain Abbas, CS-131, 07/25/25

Abstract

To accurately classify bird and airplane from the radar data, we constructed a model called Naïve Bayesian classification model. Using this model, we can correctly predict the classification of an unknown object and provide useful statistics to identify an anomaly in radar data. Initially, we will utilize likelihood and velocity data to train our model. The model will be tested by incorporating test data to classify unknown objects and provide corresponding statistics.

Introduction

A frequent problem at airports is the collision between airplanes and birds. Birds are small, fast-moving and often fly in unpredictable patterns. Usually, near airports and runaways the kinematics and flight paths of the airplane and birds tend to align to some extent, making it difficult for air traffic controllers and pilots to correctly identify birds from airplanes. As a result, a high-speed airplane gets damaged from a bird collision. One of the famous examples is the collision of US Airways flights with the flock of birds near a Hudson River in 2009. The pilot couldn't correctly identify a flock of birds from the radar data. As a result, the plane crashed in Hudson River.

To counter this problem, we designed a Naïve Recursive Bayesian classification model that would predict the class of an unknown object. We will use an initial dataset, and prior to update the posterior and classify unknown objects.

Methodology

I. <u>*Preprocessing the data*</u>

Initially, we preprocessed the data before feeding into classifier model. The length of likelihood dataset is (2, 400). For each class the likelihood data provides 400 values. However, the length of the velocity for each class of size 10 is 600. Therefore, we performed linear scaling to velocity data to obtain corresponding likelihood indexes.

```
# Define binning range
vmin  = np.nanmin(testing_data)
vmax = np.nanmax(testing_data)
num_bins = likelihood.shape[1]

# Compute likelihood indices directly using linear scaling
def velocity_to_index(v):
   # Scale velocity to bin index range [0 399]
   if np.isnan(v):
      return -1
   scaled = (v - vmin) / (vmax - vmin) * (num_bins - 1)
   index = int(np.clip(np.floor(scaled),0, num_bins - 1)) # ensure index is within
bounds
   return index
```

The likelihoods of two classes look like

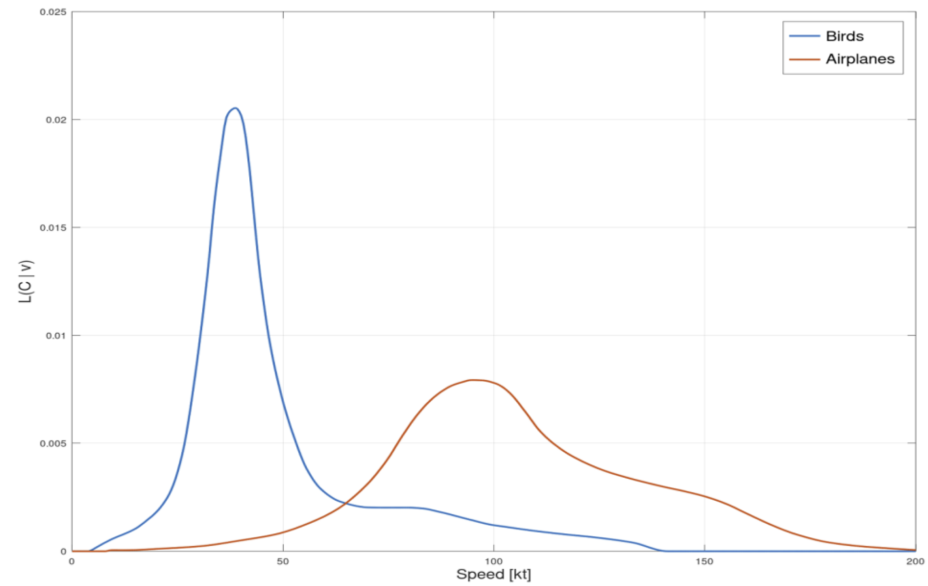

Figure 1: The above figure shows the likelihoods of bird and airplane as a function of velocity

II. *Designing Recursive Naïve Bayesian*

a. *Naïve Bayesian Classification:*
The classification model utilizes Bayes' theorem to calculate the probability of an object being a bird or airplane. The specific class is based on prior knowledge and current knowledge.

$$P(C \mid V) = \frac{P(\mathcal{V}|C).P(c)}{P(v)}$$

b. *Transition Probabilities:*
They reflect the likelihood of staying in the same class versus transitioning to another class over time. This adds temporary consistency to the classifier.

$$P(B_{t+1} \mid B_t)=0.9,\ \ P(A_{t+1} \mid A_t)=0.9$$
$$P(B_{t+1} \mid A_t)=0.1,\ \ \ P(A_{t+1} \mid B_t)=0.1$$

c. *Priors:* Initial priority is set to:

$$P(B) = 0.5, P(A) = 0.5$$

d. *Likelihoods:*
P (C | v) = probability of observing an unknown object given velocity

e. *Recursive Estimation:*
The posterior probability from the previous step becomes the prior for the next step, allowing the classifier to account for temporal dependencies.

f. *Normalization:*
At the end of the calculation, the probabilities are valid (sum to 1) after each update

g. *Handling Nans:*
When a value is missing (e.g., NaN), skip the step and retain the prior probabilities from the previous iteration.

III. *Feature selection:*

a. *Velocity:*
Initially, we used the velocity vector to test our classifier model. However, there were cases where models misclassified unknown objects. This entailed us to integrate second feature vector into our model to improve accuracy scores.

b. *Velocity and Acceleration:*
Using the velocity data we derived the second feature called acceleration.
The acceleration data contains the same size as the velocity dataset.
We used Naïve Bayesian multinomial model to integrate acceleration data into our model.

$$P(C|v, a) \propto P(C) \cdot \prod = P(wi|C)^{fi}$$

c. *Velocity, acceleration and total displacement:*
As a third feature, we incorporated total displacement derived from the velocity data. This addition made the model more robust, allowing it to effectively handle data anomalies and achieve higher classification accuracy.

IV. *Training the model:*
We used velocity and likelihood data to train our model. Using the test data we examined the accuracy of our model and derived second and third features from it.

V. *Test data:*
We tested the model using the velocity of 10 unknown objects. Each has a size of 600.

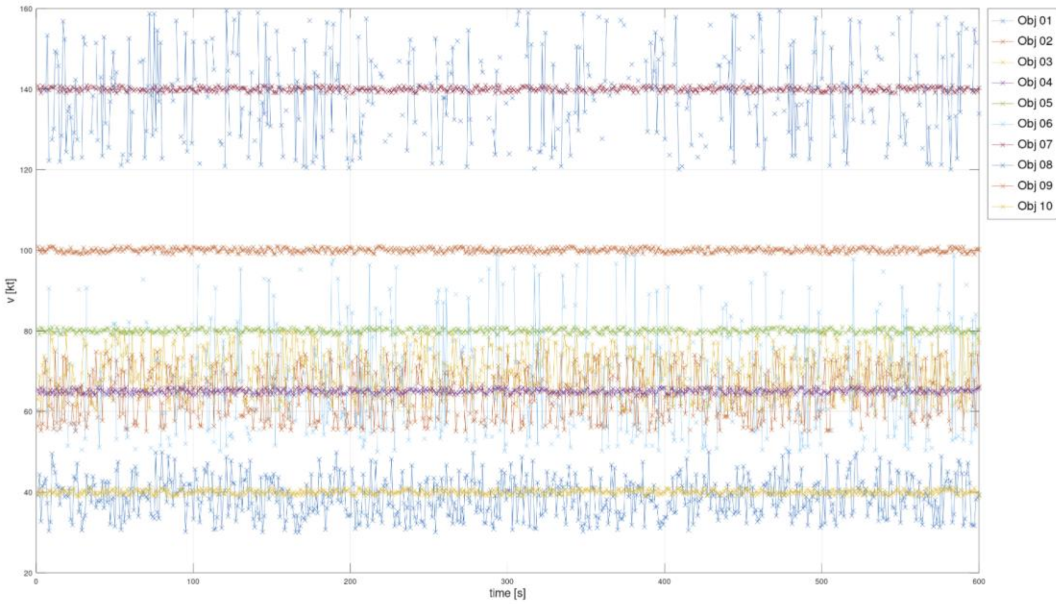

Figure 2: Velocity data for 10 unknown objects at 600 samples

Results

After training the model, we examined the performance of our model using velocity as feature vector.

I. *Feature vector: Velocity*

```
Track1: Final_class = b
Track2: Final_class = b
Track3: Final_class = b
Track4: Final_class = b
Track5: Final_class = a
Track6: Final_class = b
Track7: Final_class = a
Track8: Final_class = a
Track9: Final_class = a
Track10: Final_class = b
```

	precision	recall	f1-score	support
airplane	1.00	0.80	0.89	5
bird	0.83	1.00	0.91	5
accuracy			0.90	10
macro avg	0.92	0.90	0.90	10
weighted avg	0.92	0.90	0.90	10

Figure 3: The above shows the classification results using only velocity as a feature vector

Using the velocity as the only feature vector, we obtained accuracy of 90%. The precision for airplane is 1 whereas, for birds its 83%. It means that there are few instances where models misclassify airplane as birds. The Recall is 80% for airplane, meaning that it only identified 80% of actual airplane instances – some were missed (false negatives. For birds, the recall is 100%, which means that there are no false negatives. The F1 score for birds is about 91% which is slightly better than airplane (89%)

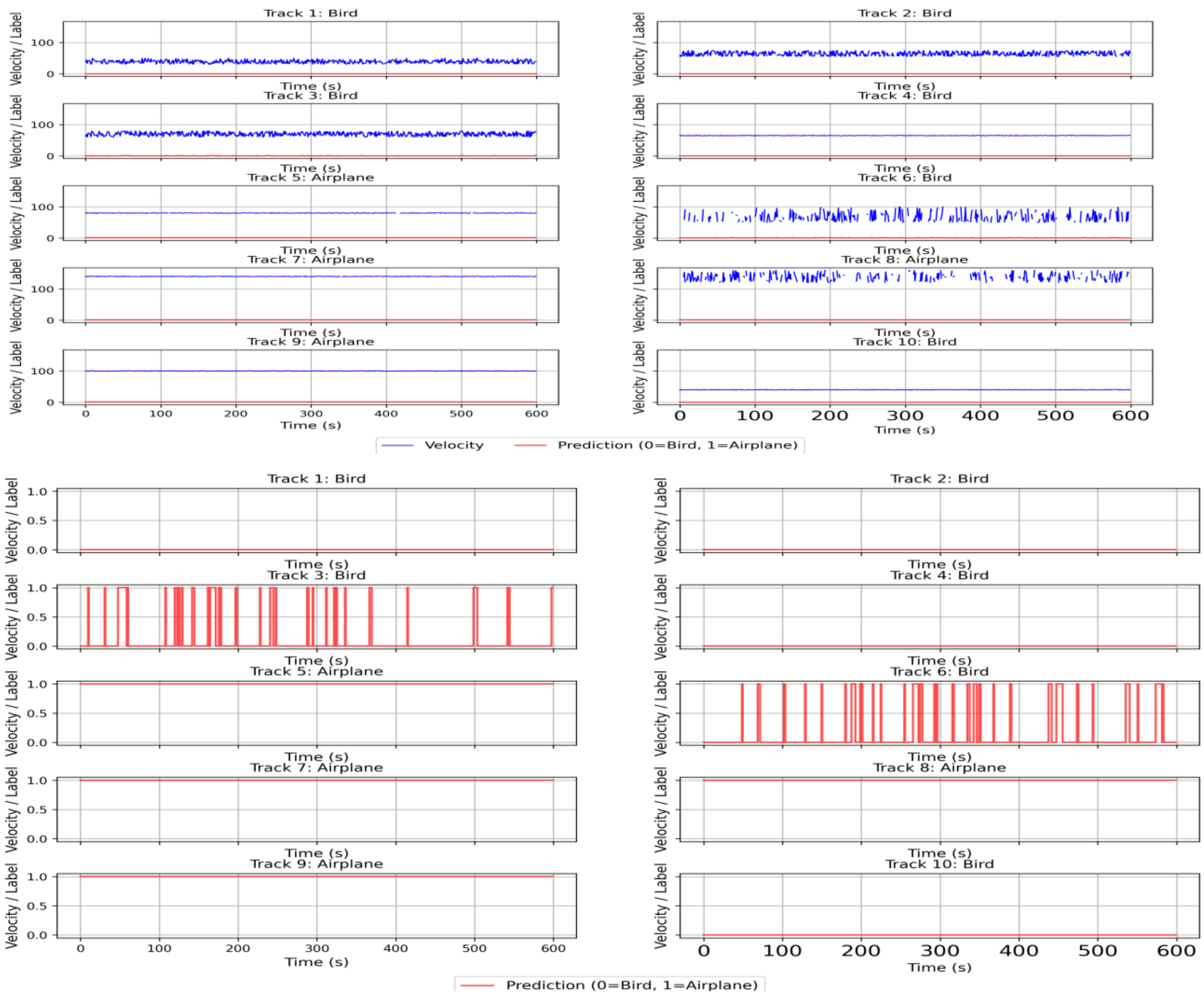

Figure 4: The above figure shows predicted values from the model.

II. *Feature vector: Velocity and Acceleration:*

To improve our model performance, we integrated acceleration into our NBC model and computed the performance results.

Figure 5 shows perfect classification results for 10 unknown objects.

The classification report shows perfect statistics for all accuracy parameters (See below)

```
Track1: Final_class = b
Track2: Final_class = b
Track3: Final_class = b
Track4: Final_class = a
Track5: Final_class = a
Track6: Final_class = b
Track7: Final_class = a
Track8: Final_class = a
Track9: Final_class = a
Track10: Final_class = b
```

	precision	recall	f1-score	support
airplane	1.00	1.00	1.00	5
bird	1.00	1.00	1.00	5
accuracy			1.00	10
macro avg	1.00	1.00	1.00	10
weighted avg	1.00	1.00	1.00	10

Figure 5: The above figure shows classification results using acceleration and velocity as features.

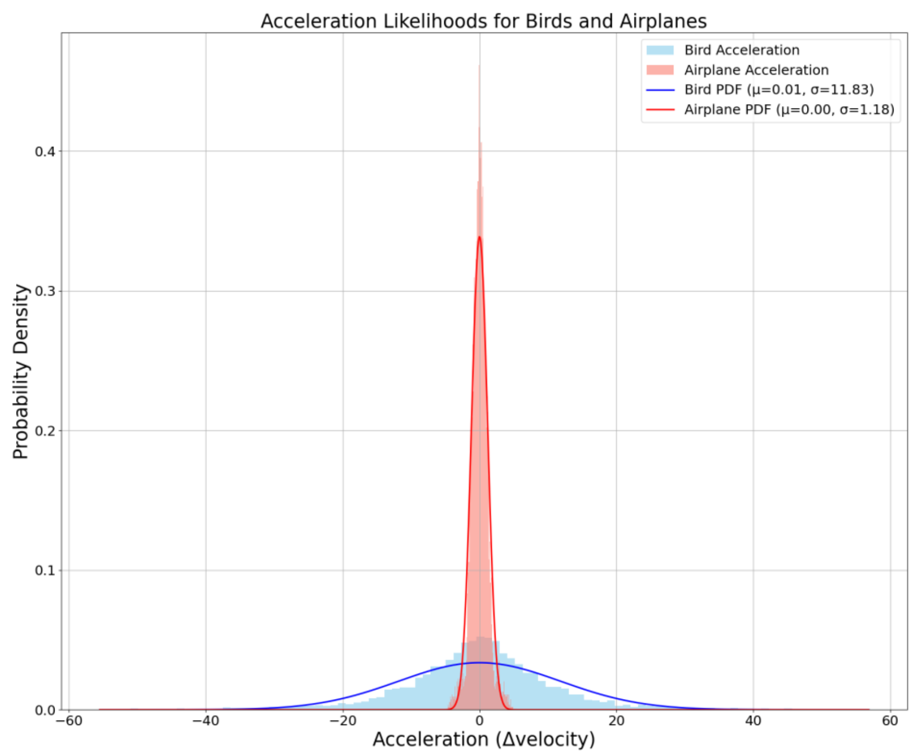

Figure 6: The above figure shows acceleration and likelihood for birds and airplane.

The histograms are symmetric at 0 where standard deviation for bird = 11.83 and airplane = 1.18. The points where two PDFs intersect are critical points because the model misclassifies the object.

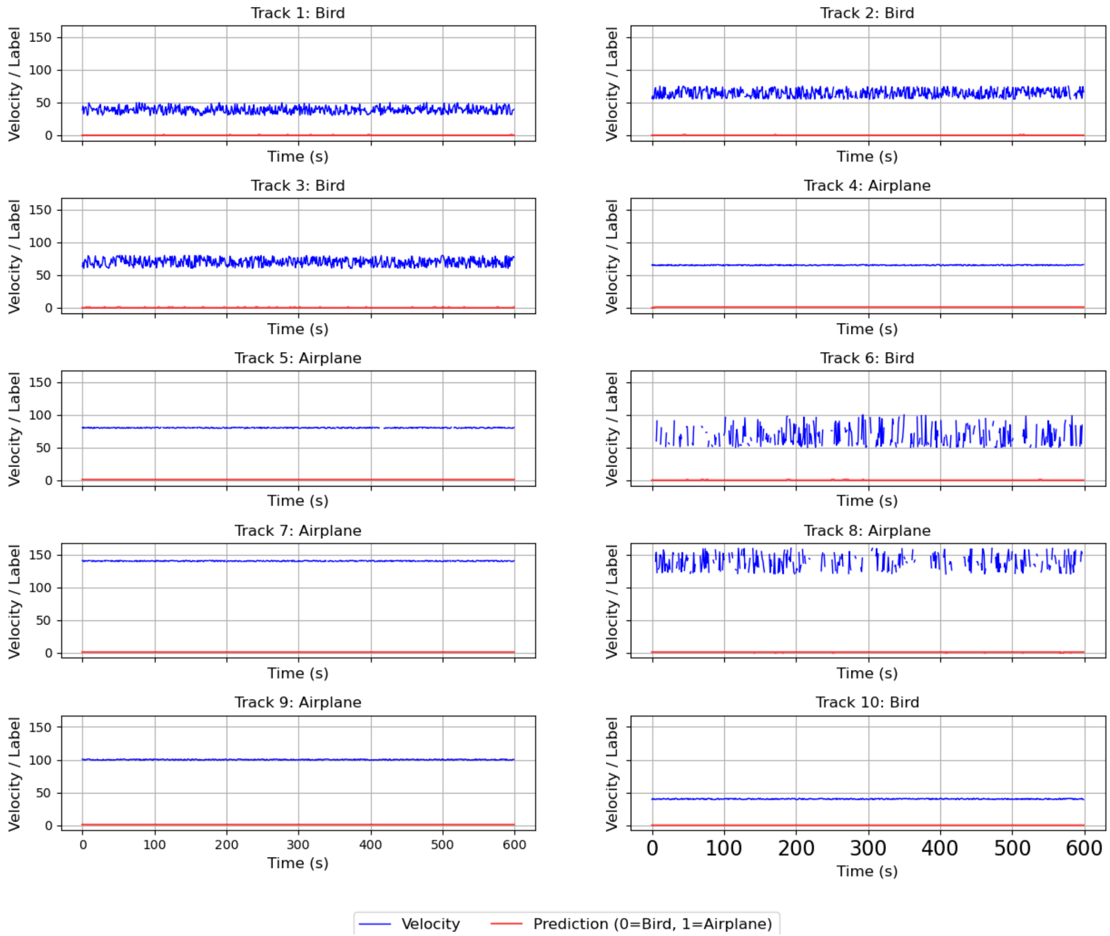

Figure 7: The above figure shows the velocity of unknown objects along with the classification in red

After integrating acceleration intos our model, we successfully predicted the objects. Figure 7 shows the velocity of each object (blue) along with the prediction. If it's a bird the prediction (red) stays at zero, but if it's an airplane it shows at 1.

Figure 8 below shows the predicted levels in detail. Using the acceleration and velocity vectors, the model successfully classifies bird and airplane objects for test data.

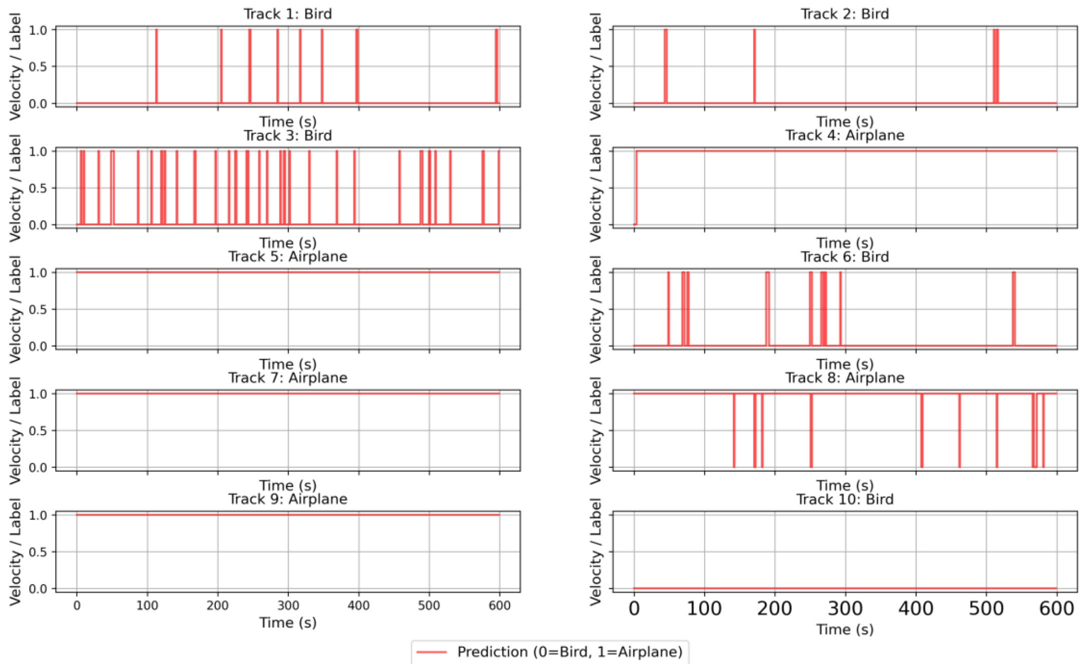

Figure 8: Predicted values for test data using velocity and acceleration as feature vectors.

III. *Feature vector: Acceleration, velocity, displacement*

Our third feature vector is total displacement. This feature is also derived from the velocity vector. It is described as:

$$d(t) = \sum_{0}^{t} v(t)\ .\ \triangle t$$

Where $\triangle t = 1s$

Using this feature, we derived the total displacement for each track. This allowed us to examine the classification of an object at total displacement intervals. The velocity, acceleration and displacement allowed us to create a robust model that could classify unknown objects with perfect accuracy. Figure 10 shows the classification result using displacement as third feature.

```
Track1: Final_class = b
Track2: Final_class = b
Track3: Final_class = b
Track4: Final_class = a
Track5: Final_class = a
Track6: Final_class = b
Track7: Final_class = a
Track8: Final_class = a
Track9: Final_class = a
Track10: Final_class = b
```

	precision	recall	f1-score	support
airplane	1.00	1.00	1.00	5
bird	1.00	1.00	1.00	5
accuracy			1.00	10
macro avg	1.00	1.00	1.00	10
weighted avg	1.00	1.00	1.00	10

Figure 9: The above figure shows the classification of unknown objects using displacement as third feature

Figure 10 below shows a likelihood plot for birds and airplanes at each displacement level. This gives an extra observing feature for analysts to classify objects as a function of distance. The pdf lines intersect at 30 km. This shows that models would possibly mislabel the object at this interval.

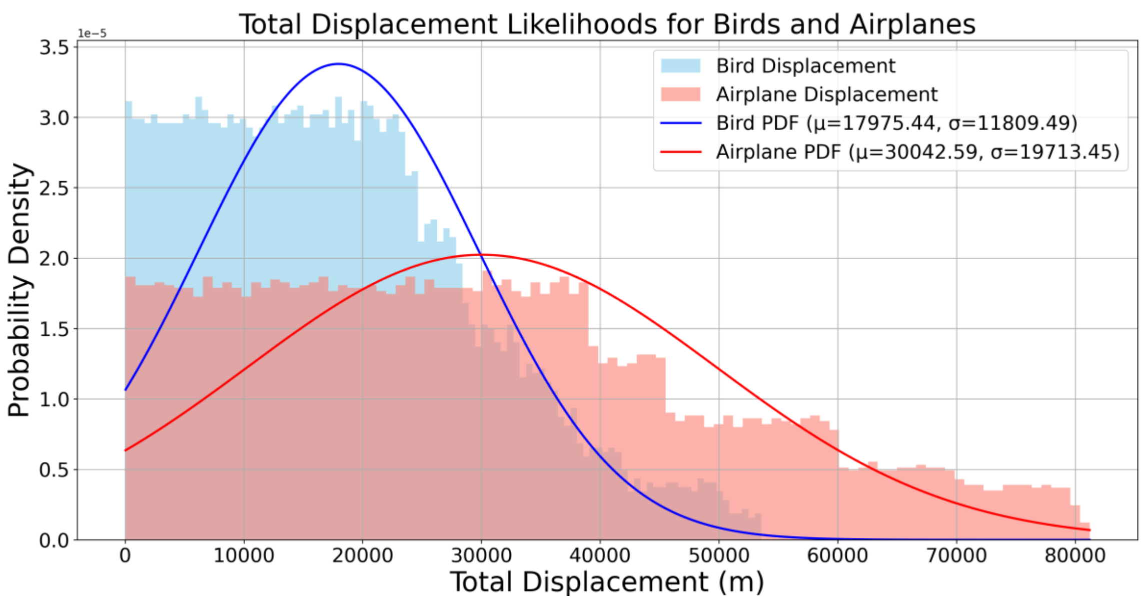

Figure 10: The above figure shows the probability density of total displacement for birds and airplane.

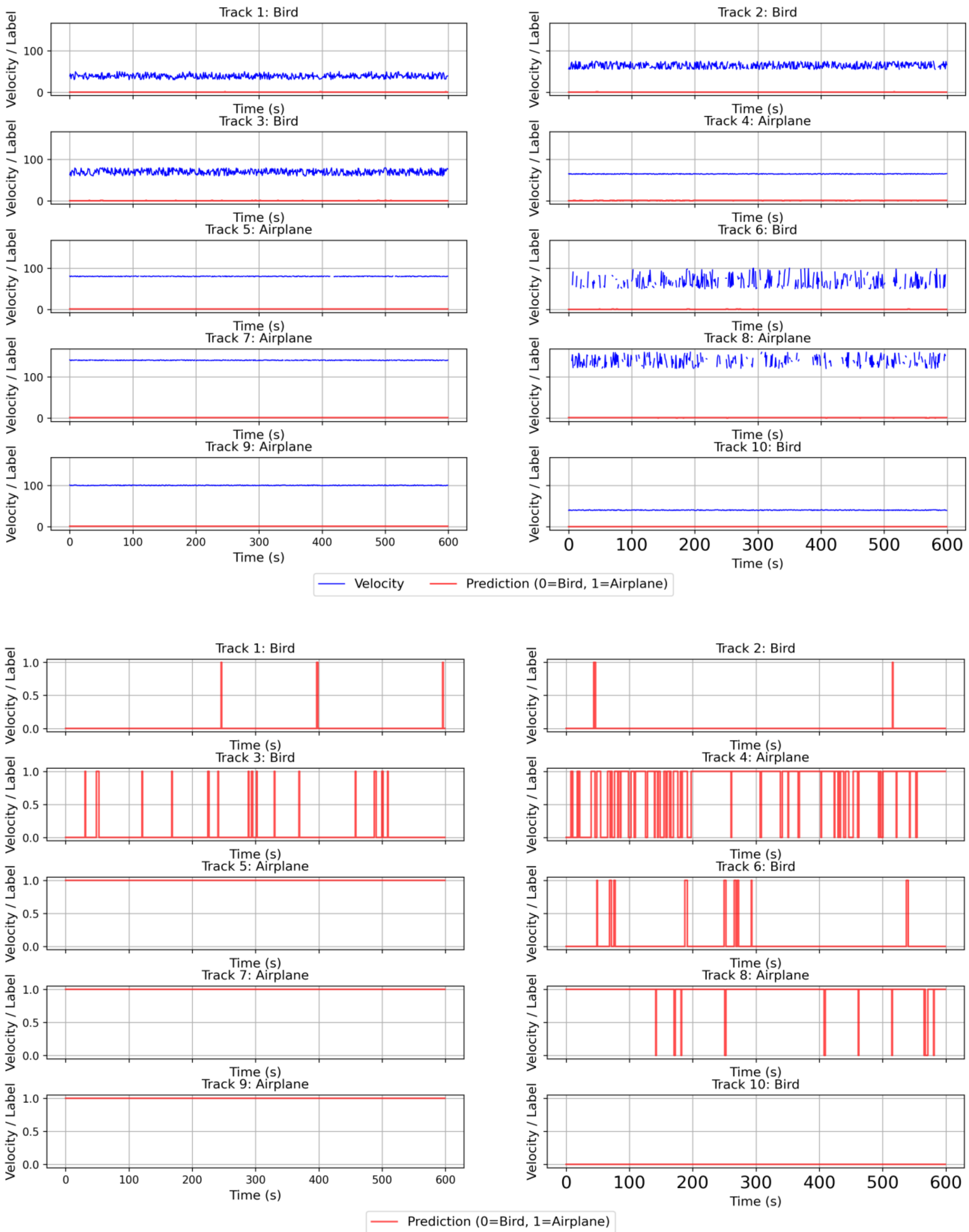

Figure 11: The above figure shows the classification of unknown objects using displacement as third feature

Conclusion

To successfully classify 10 unknown objects, we tested a recursive Naive Bayes model using velocity, acceleration, and total displacement as features. By incrementally adding these feature vectors, we significantly improved the model's ability to handle variations and anomalies in the data. The final model, which combined all three features, achieved a classification accuracy of 100%.

The addition of acceleration and displacement provided complementary information to velocity alone. Acceleration captured changes in motion dynamics, while displacement offered insight into the overall movement pattern over time. This richer representation allowed the model to better distinguish between birds and airplanes, especially in borderline or noisy cases. We also evaluated the performance of each feature combination using standard classification metrics such as accuracy, precision, and F1 score. These metrics consistently showed that incorporating more informative features led to more reliable and robust predictions. Overall, it was great assignment to classify objects using Naïve Bayesian model.