

Neural Network Models Report – Penguins Dataset

This report analyzes the performance of two classic neural network models — **Perceptron** and **Adaline** — on the Palmer Penguins dataset. Several combinations of features and species classes were tested to evaluate linear separability and model convergence behavior. Each combination was visualized and evaluated using manually computed confusion matrices to assess accuracy and boundary clarity.

1. Feature and Class Combinations Overview

Experiment	Features	Classes	Notes
1.1	CulmenDepth, CulmenLength	Adelie – Chinstrap	Moderate overlap, weak linear separability.
1.2	CulmenDepth, CulmenLength	Adelie – Gentoo	Clear separation; Gentoo larger in features.
1.3	CulmenDepth, CulmenLength	Chinstrap – Gentoo	Partially separable, Adaline performs better.
2.1	FlipperLength, OriginLocation	Adelie – Chinstrap	Categorical feature adds complexity.
2.2	FlipperLength, OriginLocation	Adelie – Gentoo	Gentoo separable by flipper size and location.
3	FlipperLength, BodyMass	Gentoo – Adelie	Strong linear separation, ideal for Perceptron.
4	CulmenLength, FlipperLength	Chinstrap – Adelie	Slight overlap, Adaline handles smoother boundary.

2. Model Comparison Summary

Perceptron:

- Works best when classes are **linearly separable**.
- Achieved high performance on **(FlipperLength, BodyMass) → (Gentoo, Adelie)**.
- Converged quickly with minimal misclassifications.

Adaline:

- Performs better when data has **minor overlaps**.
- Showed stable convergence in **(CulmenLength, FlipperLength) → (Chinstrap, Adelie)**.
- MSE-based updates smoothed classification boundaries.

3. Confusion Matrix Observations

Across experiments, confusion matrices indicated that: Adaline achieved slightly better balanced results when class distributions were uneven. Perceptron achieved near-perfect results only for well-separated feature combinations. For combinations involving **OriginLocation**, accuracy decreased due to non-linear separability.

4. Conclusions and Recommendations

- For linearly separable features (e.g., **FlipperLength, BodyMass**), use **Perceptron** for fast convergence and clarity.
- For overlapping features (e.g., **CulmenLength, FlipperLength**), **Adaline** performs more stably.
- Avoid using purely categorical combinations such as **OriginLocation** unless properly encoded.
- Feature scaling and encoding significantly impact both models' accuracy.

Best-performing configurations: Perceptron → (FlipperLength, BodyMass) vs (Gentoo, Adelie)
Adaline → (CulmenLength, FlipperLength) vs (Chinstrap, Adelie)