Predicting the penguin species

Oldřich Šmehlík, Artem Sorokin, Kryštof Dostál, Jan Jiran, Jakub Chládek

April 18, 2023

1 Introduction

In course 4IT439 Data-X – applied data analytics models in real world tasks taught at Faculty of Informatics and Statistics, Prague University of Economics and Business in summer semester 2023, the course instructors assigned us to write a term paper on predicting penguin species. This document is the result of our work.

2 Problem Definition

The target is to predict the species of penguin (multi-class classification), based on a popular, slightly modified (for the purpose of this exercise) Palmer Penguins dataset. The goal of this project is to create, optimize and evaluate several models to do so. The success of the models will be measured in prediction accuracy. Since the dataset is very small, calculation time is not a suitable measurement. The dataset does need some data preparation (such as treating NaN values, duplicates etc.). The preparation is part of this project, and will include replacing missing values, omitting unuseful attributes and other modifications.

3 Data Understanding

3.1 Dataset Introduction

The original dataset (without modifications) comes from a study (Gorman et al., 2014) and can also be publicly found at the following URL:

`https://gist.github.com/slopp/ce3b90b9168f2f921784de84fa445651`

The dataset provided for this project was however slightly changed for the purpose of learning to treat certain situations.

3.2 Dataset Content

Our dataset contains 363 rows and 8 columns:

- 1 column with target variable (`species`)
- 7 columns with possible input features (`island`, `bill_length_mm`, `bill_depth_mm`, `flipper_length_mm`, `body_mass_g`, `sex`, `year`)
- 5 columns with numerical values (`bill_length_mm`, `bill_depth_mm`, `flipper_length_mm`, `body_mass_g`, `year`), out of these 4 contains floating numbers (`bill_length_mm`, `bill_depth_mm`, `flipper_length_mm`, `body_mass_g`) and 1 contain integer number (`year`)
- 3 columns with categorical values (`species`, `island`, `sex`)

The representation of the target variable (`species`) is:

- `Adelie` – 160 occurences
- `Gentoo` – 124 occurences
- `Chinstrap` – 79 occurences

As we can see, the classes are imbalanced. To solve this, we have decided to use oversampling using SMOTE method to solve class imbalance in our case.

3.3 Missing values

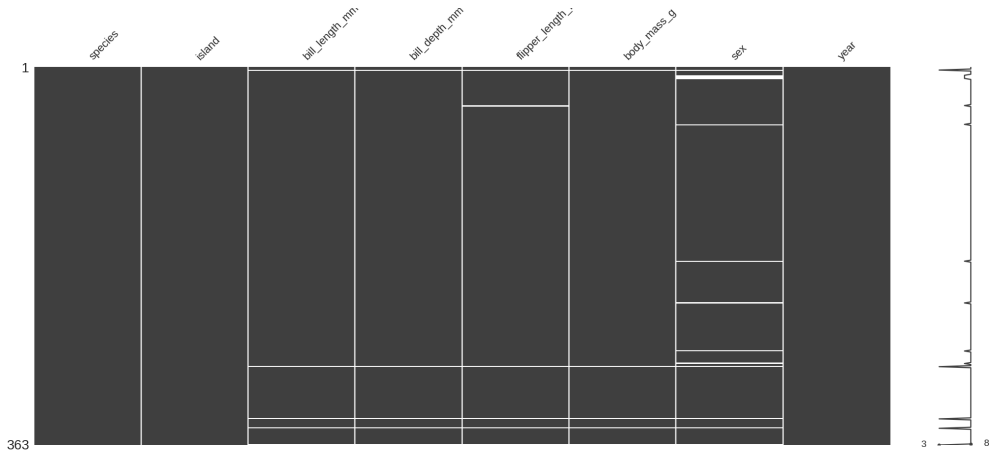

Figure 1: Placement of missing values in dataset

As we can see in Figure 1, our dataset contains a small amount of missing values in columns `bill_length_mm`, `bill_depth_mm`, `flipper_length_mm`, `body_mass_g`, `sex`. The gender (sex) of the penguin is missing quite often. This is probably due to the fact that telling the sex of the penguin is often difficult - sometimes even for the penguins. It is a popular theory that penguins have the highest percentage of same sex couples (up to 28% according to some studies), because most species mate for life and can "make a mistake", when picking their mate. One of the common ways to determine the penguin sex is to measure it's bill (beak) - male bills are often slightly larger.

3.4 Correlations

Correlation between input features is shown textually in Table 1 and visually in Figure 2. As we can see, the most correlated features are `body_mass_g` and `flipper_length_mm` with correlation value of 0.86.

	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	year
bill_length_mm	1.000000	-0.219291	0.639585	0.578546	0.050733
bill_depth_mm	-0.219291	1.000000	-0.586423	-0.479516	-0.090926
flipper_length_mm	0.639585	-0.586423	1.000000	0.869749	0.201139
body_mass_g	0.578546	-0.479516	0.869749	1.000000	0.082095
year	0.050733	-0.090926	0.201139	0.082095	1.000000

Table 1: Correlations

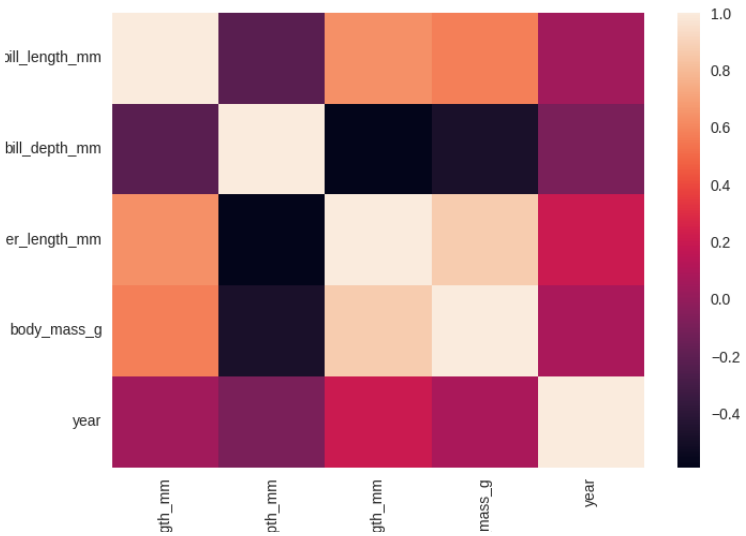

Figure 2: Correlations

4 Data Preparation

The `year` column was found not suitable for predicting, since it says little about the penguin, changes in the future and potentially degrades the quality of the predictions. Therefore it will be omitted, and only physical attributes of the penguin and the location where it was found (`island`) will be used.

Afterwards, 13 duplicates are removed, rows with more than 5 NaN values are also deleted (since they can hardly provide enough valuable information for accurate prediction) and datatypes are converted (`species`, `island`, `sex` to categorical).

For missing sex values, a Random Forest model is created to replace them. For the other missing values, a mean value of the other ones is used for replacement.

We split dataset into train (80% of the dataset) and test (20% of the dataset) parts using split stratified by our target variable. We do oversampling using SMOTE method on train part. We also standard scale numerical columns and one hot encode categorical columns.

5 Data Visualization

5.1 Island habitability

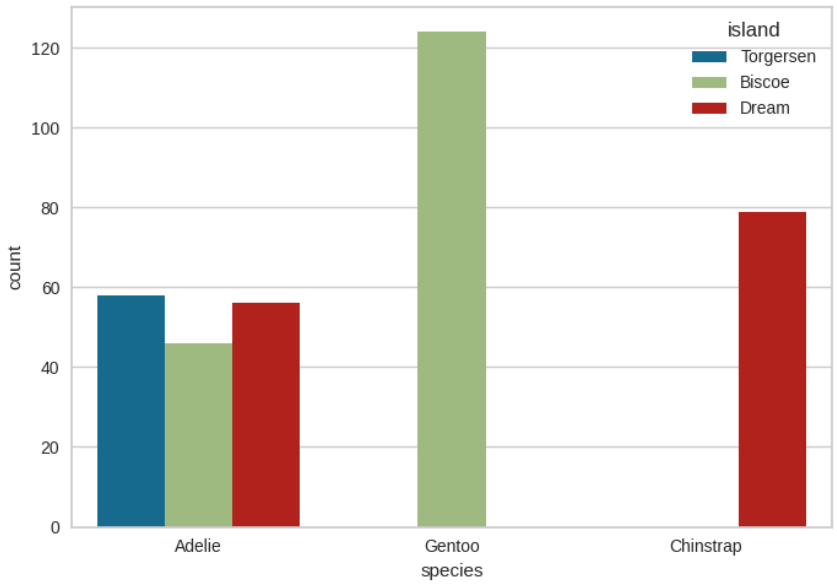

Figure 3: Island habitability

First, we used the countplot from seaborn to demonstrate the separation of penguins

on the three islands. This is shown on Figure 3, Adelie species inhabit all three islands, but Gentoo and Chinstrap are separated. Gentoo lives on Biscoe Island. Chinstrap lives on Dream Island. It follows that we need to find patterns in our data with which we can distinguish Adelie from the other species.

5.2 Outliers

The next visualization serves to identify outliers. Boxplots do an excellent job with this task. The boxplot edges correspond to 1 and 3 quartiles, the whisker tips represent the minimum and maximum values, and the points outside the graph are outliers. We can see from the Figure 4 that the outliers are present, but their number is minimal.

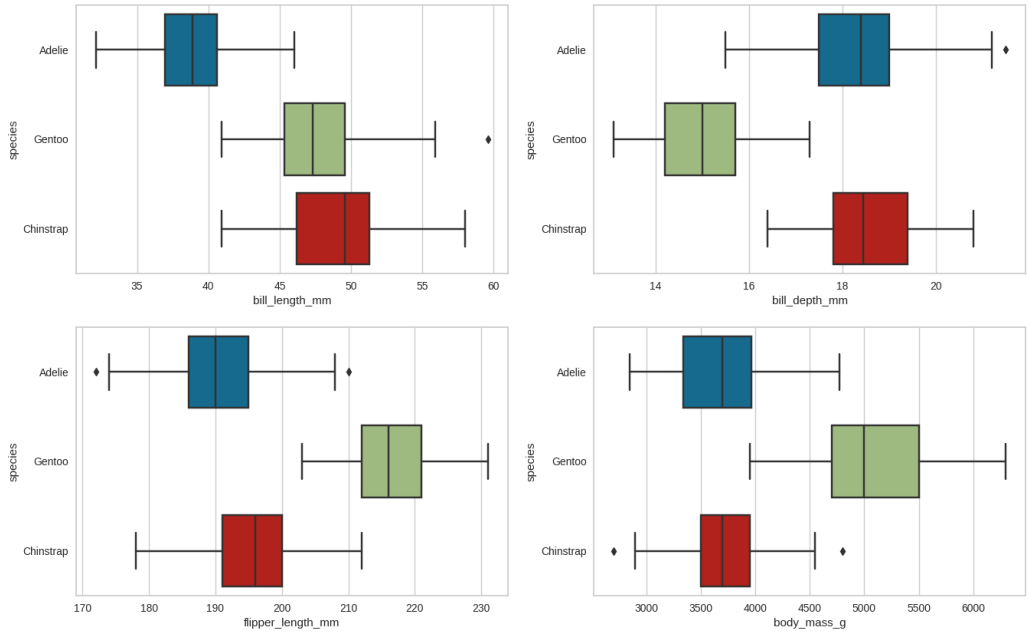

Figure 4: Boxplots

5.3 Pairplots

By using the sns.pairplot graph, which builds a scatter plot matrix, we can see how well the Gentoo species separates on the dot plot between the `bill_depth_mm` and `body_mass_g` features. This is shown on Figure 5.

But, unfortunately, the Adelie and Chinstrap separation is not so clear from this chart. On the diagonal of the graph are the histograms for each numeric attribute, and only by

the `bill_length_mm` attribute we can see the difference between the Adelie and Chinstrap histograms.

5.4 Adelie and Chinstrap in more detail

Let's look at Adelie and Chinstrap in more detail. We prepare filtered samples of the data by sex and species (Adelie is additionally filtered by Dream Island, because Chinstrap only lives on it). By plotting a histogram for male and another for female shown in Figure 6, we can see that the male sex is perfectly separated, but the female sex has little overlap.

For a more detailed analysis of the female gender of Adelie and Chinstrap, we constructed another sns.pairplot, which is shown in Figure 7. From this matrix of scatter diagrams we could not find features by which the two species could be separated without making a mistake. But the strongest separation was identified in the diagram x = `bill_length_mm` y = `flipper_length_mm`.

6 Modeling

We have decided to try few different models, mainly the ones which were presented in class, which means we have done a hyperparameter tuning on Linear Support Vector Classifier, Decision Tree, Random Forest, Bagging Classifier and Neural Network models. The values of hyperparameters tried are not that important, therefore we present only the best hyperparameters, which were found. These are shown in Table 2.

Model	**Hyperparameters**
Neural Network	(explained later)
Linear Support Vector Classifier	C = 0.5
Random Forest ,	criterion = gini max_depth = None min_samples_leaf = 1 min_samples_split = 5 n_estimators = 50
Bagging Classifier	max_features = 0.6, max_samples = 0.6
Decision Tree	criterion = gini, max_depth = 4

Table 2: The Best Hyperparameters Values Found

The measured accuracies of models with the best hyperparameters found are shown on Table 3. As we can see, the best accuracy had neural network. That was great also because the assignment explicitly mentions we must save model in h5 format, and we

Model	Accuracy
Neural Network	1.000000
Linear Support Vector Classifier	0.997260
Random Forest	0.989078
Bagging Classifier	0.989078
Decision Tree	0.978156

Table 3: Accuracies Of The Models With Best Hyperparameters Values

know an easy way how to do it in both Keras (which we use in our neural network model) and scikeras (which is a scikit-learn compatible wrapper for Keras), so we have decided to select neural network as the best model.

The architecture of our neural network model is as follows: The model consists of sequence of Input Layer with 9 input neurons, followed by a Dense Layer with 16 neurons and relu activation function, followed by a Dense Layer with 3 neurons and softmax activation function. The architecture is shown in Figure 8.

6.1 Model limitations and considerations

Our model is limited by following factors:

- The architecture is limited by number of input features, which places a restriction on the number of neurons in input layer, which must be same as number of input features. We do this by setting input_dim parameter of first Dense layer, however we probably could achieve the same result by using separate Input layer with required number of neurons.
- The architecture is limited by performed task, which is a multi class classification, which places the following restrictions on the network:
    1. the number of neurons in output layer must be the same as number of predicted classes
    2. we must use softmax activation function on the last layer, which returns a sequence of predicted class probability scores between 0 and 1 which summs up to 1, where the class with maximum probability score is the class which should be predicted

6.2 Ideas to improve the model

Model could be improved in such a way that using Dropout layer would prevent overfitting.

6.3 Choosing the values of hyperparameters

Values of hyperparameters (such as number of layers or number of neurons in layer) are hardcoded in our model making function and were chosen in such a way that we wanted to start with some small number of layers and neurons and if it would not work well, increase the number of layers or neurons. However, because chosen values worked well, we have not needed to use more complicated networks.

7 Evaluation

As we can see from confusion matrix shown at Figure 9, our model achieves 100% accuracy on the test set. That might look slightly suspicious, however because we assume perhaps the problem is not as hard as we thought, we have decided to present this as our result.

8 Development Environment

We use Python version 3.9.16 and we also directly use the following packages:

- `imblearn` (ver. 0.0) – for oversampling using SMOTE method
- `matplotlib` (ver. $3.7.1$) – for visualization
- `pandas` (ver. $1.5.3$)– for working with data in table form
- `scikeras` (ver. $0.10.0$) – contains scikit-learn compatible wrapper which we use
- `seaborn` (ver. $0.12.2$) – for visualization
- `scikit-learn` (ver. $1.2.2$) – for non-neural network models
- `tensorflow` (ver. $2.12.0$)– for neural network models using Keras API
- `yellowbrick` (ver. 1.5)– for visualization of class prediction error

The full list of used libraries in our environment, including their version, can be found on GitHub in the following link (since the list is unreasonably long for display here):

`https://github.com/osmehlik/vse_4it439/blob/master/requirements.txt`

References

GORMAN, Kristen B.; WILLIAMS, Tony D.; FRASER, William R., 2014. Ecological Sexual Dimorphism and Environmental Variability within a Community of Antarctic Penguins (Genus Pygoscelis). *PLOS ONE*. Vol. 9, no. 3, pp. 1–14. Available from DOI: 10.1371/journal.pone.0090081.

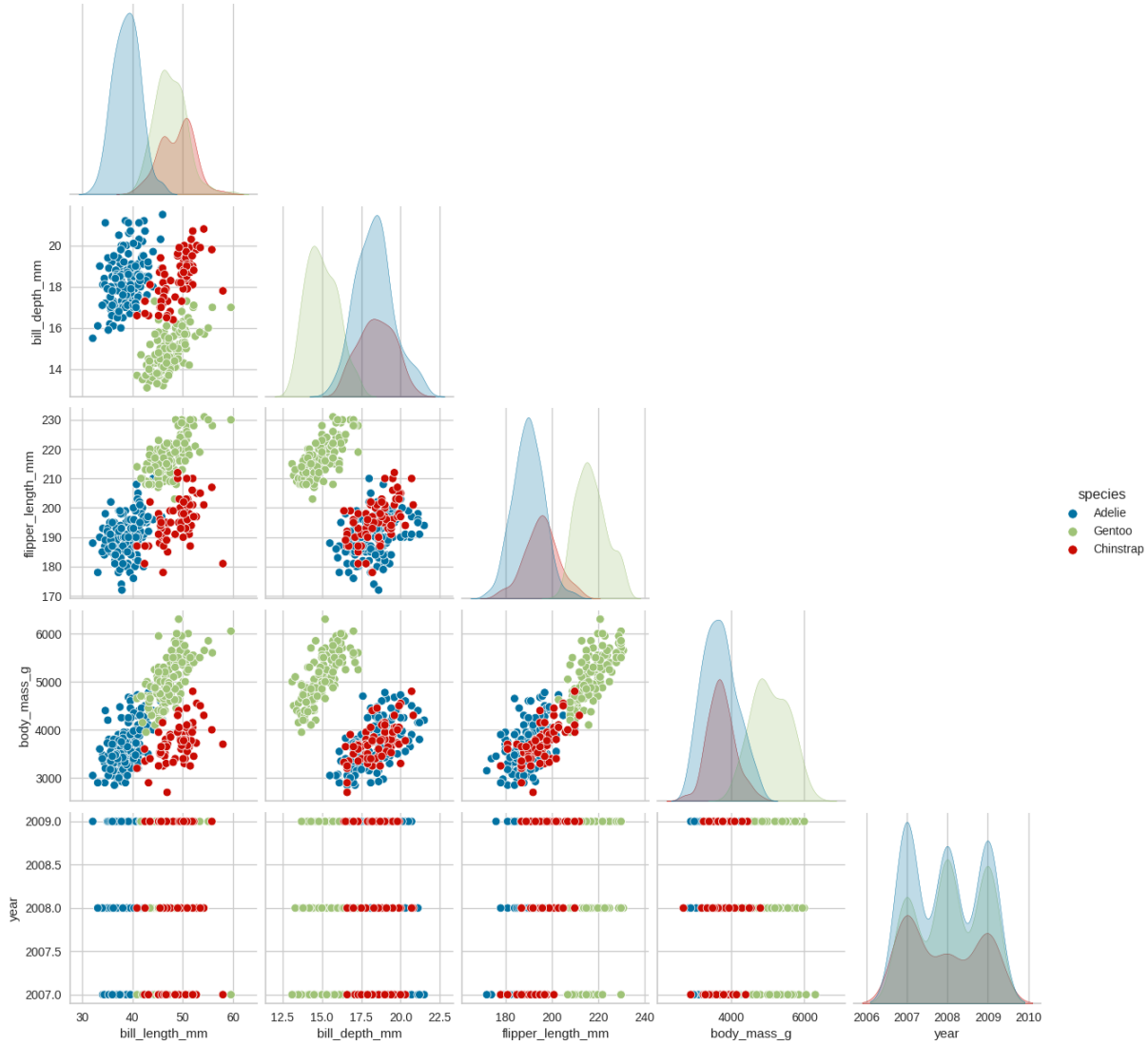

Figure 5: Pairplots

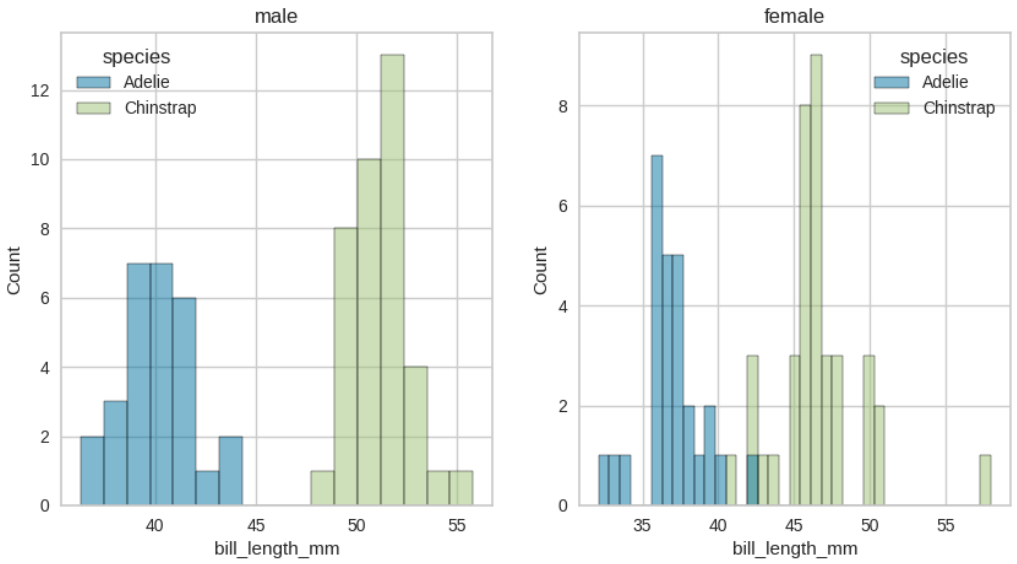

Figure 6: Adelie and Chinstrap Histograms

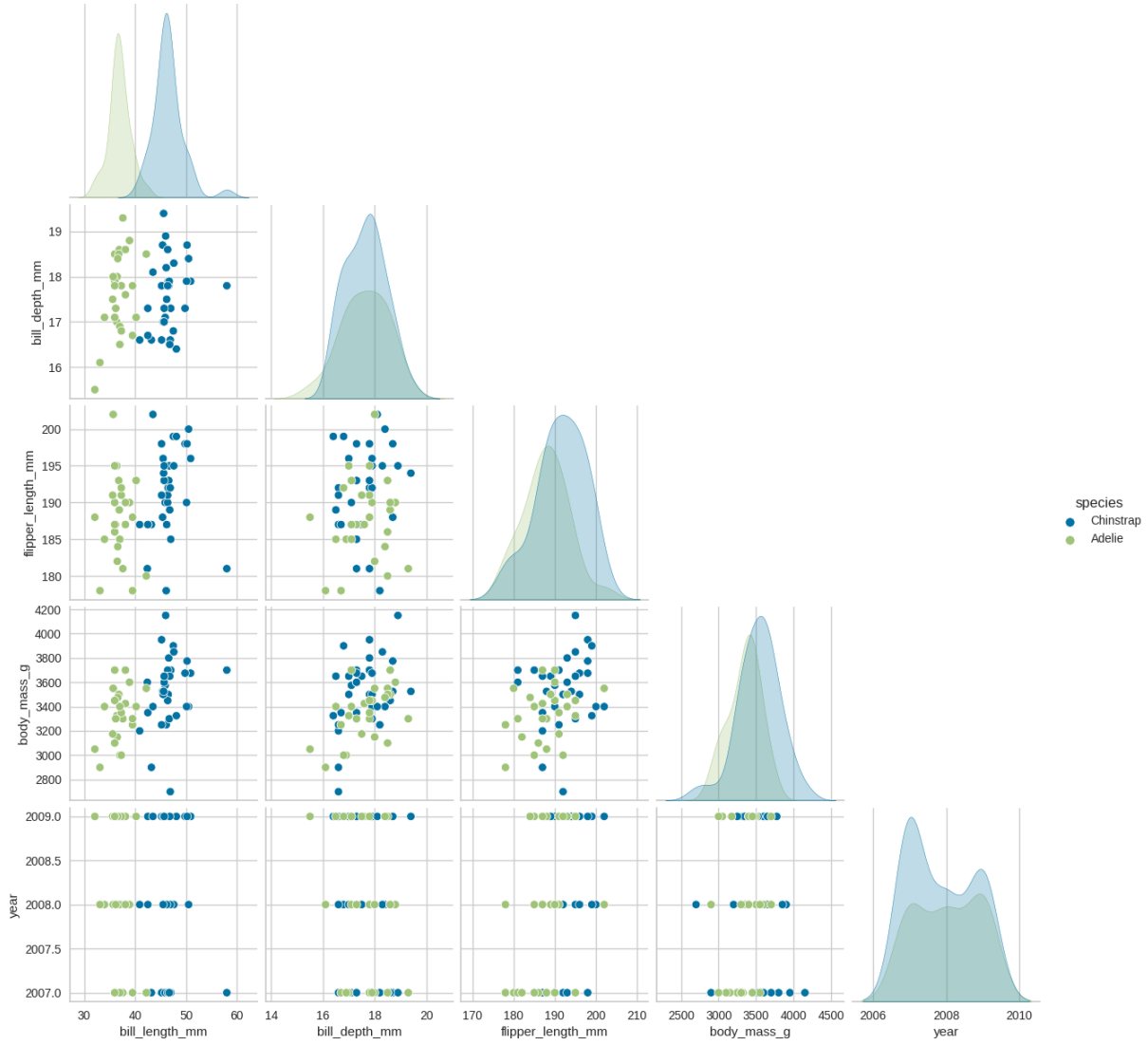

Figure 7: Adelie and Chinstrap Pairplot

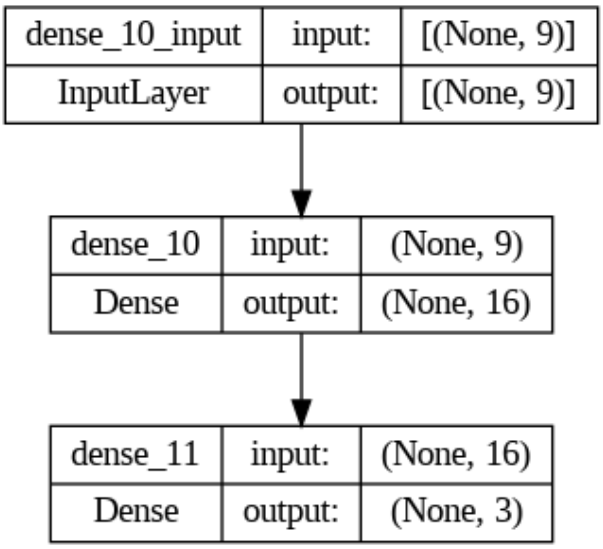

Figure 8: The Architecture Of Our Best Neural Network Model

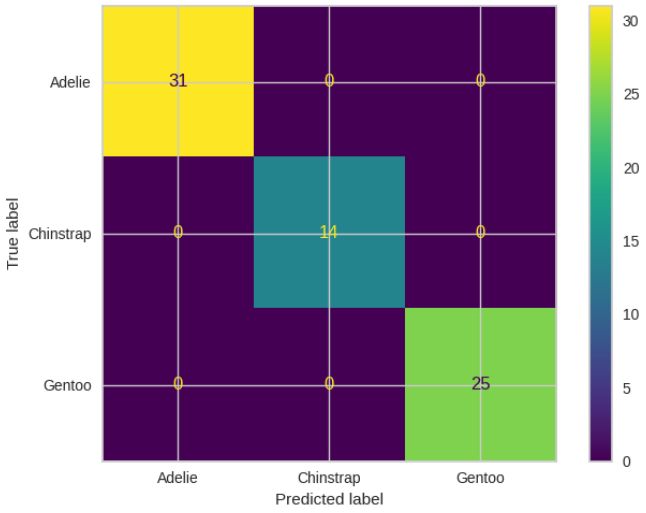

Figure 9: Confusion Matrix