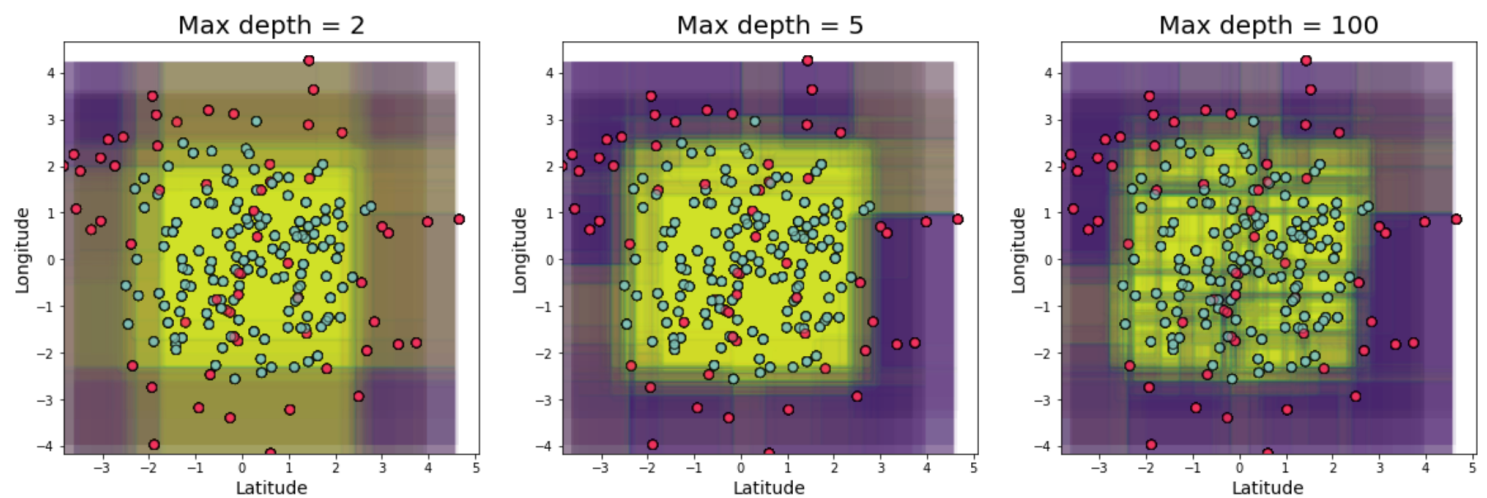


Exercise 3.2: Bagging Classification with Decision Boundary

□□□ Exercise: Bagging Classification with Decision Boundary

The goal of this exercise is to use **Bagging** (Bootstrap Aggregated) to solve a classification problem and visualize the influence on Bagging on trees with varying depths.

Your final plot will resemble the one below.



Instructions:

- Read the dataset `agriland.csv`.
- Assign the predictor and response variables as `x` and `y`.
- Split the data into train and test sets with `test_split=0.2` and `random_state=44`.
- Fit a single `DecisionTreeClassifier()` and find the accuracy of your prediction.
- Complete the helper function `prediction_by_bagging()` to find the average predictions for a given number of bootstraps.
- Perform `Bagging` using the helper function, and compute the new accuracy.
- Plot the accuracy as a function of the number of bootstraps.
- Use the helper code to plot the decision boundaries for varying `max_depth` along with `num_bootstraps`. Investigate the effect of increasing bootstraps on the variance.

Hints:

```
sklearn.tree.DecisionTreeClassifier()
```

A decision tree classifier.

```
DecisionTreeClassifier.fit()
```

Build a decision tree classifier from the training set (X, y).

```
DecisionTreeClassifier.predict()
```

Predict class or regression value for X.

```
train_test_split()
```

Split arrays or matrices into random train and test subsets.

```
np.random.choice
```

Generates a random sample from a given 1-D array.

```
plt.subplots()
```

Create a figure and a set of subplots.

```
ax.plot()
```

Plot y versus x as lines and/or markers

Note: This exercise is auto-graded and you can try multiple attempts.