The first classifier selected was the support vector machine classifier (*Scikit-*

*Learn SVM Tutorial With Python (Support Vector Machines)*, n.d.):

○ Performs classification by constructing hyperplanes in multidimensional

space.

○ Based on the concept of decision planes that define decision boundaries.

○ Employs an iterative training algorithm, which is used to minimize an error

function and is implemented using a kernel.

For our second choice we selected logistic regression (Kanade, 2022):

○ Supervised machine learning algorithm that accomplishes binary

classification tasks by predicting the probability of an outcome, event, or

observation.

○ Analyzes the relationship between one or more independent variables

and classifies data into discrete classes.

The third classifier selected was the decision tree classifier (*1.10. Decision Trees*

*— Scikit-Learn 1.4.1 Documentation*, n.d.):

○ Supervised learning method used for classification and regression

○ Creates a model that predicts the value of a target variable by learning

simple decision rules inferred from the data features.

○ A tree can be seen as a piecewise constant approximation

As the fourth one we choose a random forest classifier (Donges, n.d.):

○ Supervised learning method where the “forest” it builds is an ensemble of

decision trees.

○ Usually trained with the “bagging method

○ The general idea of the bagging method which is a combination of

learning models increases the overall result.

○ Builds multiple decision trees and merges them to get a more accurate

and stable prediction.

Finally, the gradient boosting classifier was our fifth choice (*Gradient Boosting: A*

*Step-By-Step Guide*, n.d.):

○ The algorithm starts by building a decision stump and then assigning

equal weights to all the data points.

○ Increases the weights for all the points that are misclassified and lowers

the weight for those that are easy to classify or are correctly classified.

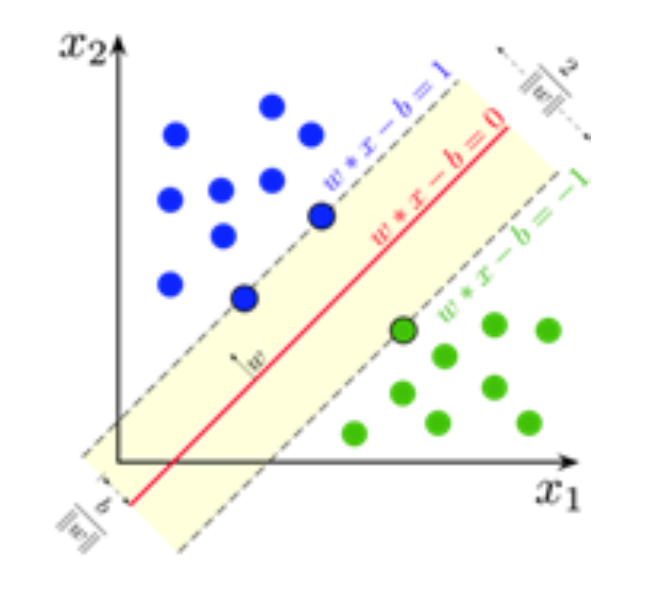
○ A new decision stump is made for these weighted data points.

The following figures illustrate each of these and are cited from the same references as

the above bullets.

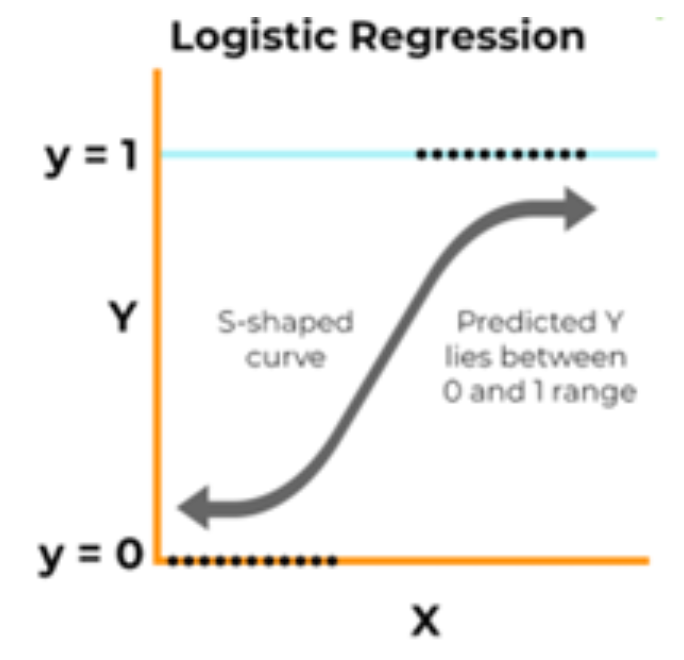
**Figure 6**

*Support vector (Scikit-Learn SVM Tutorial With Python (Support Vector Machines), n.d.).*



**Figure 7**

*Logic regression (Kanade, 2022)*

**

**Figure 8**

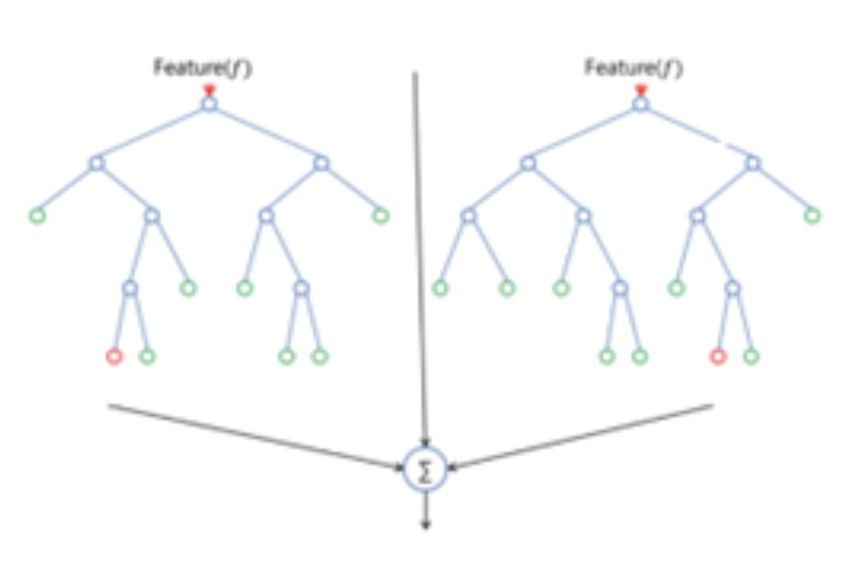
*Decision Tree, from (1.10. Decision Trees — Scikit-Learn 1.4.1 Documentation, n.d.).*

A graph with green and orange lines

Description automatically generated

**Figure 9**

*Random forest (Donges, n.d.).*

**

**Figure 10**

*Gradient boosting classifier* (*Gradient Boosting: A Step-By-Step Guide*, n.d.).

A diagram of a test

Description automatically generated

**References**

Donges, N. (n.d.). *What Is Random Forest? A Complete Guide*. Built In. Retrieved

February 21, 2024, from https://builtin.com/data-science/random-forest-algorithm

*Gradient Boosting: A Step-by-Step Guide*. (n.d.). Analytics Vidhya. Retrieved February

21, 2024, from https://www.analyticsvidhya.com/blog/2021/09/gradient-boosting-

algorithm-a-complete-guide-for-beginners/

Kanade, V. (2022, April 18). *Logistic Regression: Equation, Assumptions, Types, and*

*Best Practices*. Spiceworks. Retrieved February 21, 2024, from

https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-logistic-

regression/

*1.10. Decision Trees — scikit-learn 1.4.1 documentation*. (n.d.). Scikit-learn. Retrieved

February 21, 2024, from https://scikit-learn.org/stable/modules/tree.html

*Scikit-learn SVM Tutorial with Python (Support Vector Machines)*. (n.d.). DataCamp.

Retrieved February 21, 2024, from https://www.datacamp.com/tutorial/svm-

classification-scikit-learn-python