UNC
GREENSBORO

CS 405/605 Data Science

Dr. Qianqian Tong

About me

Contact information:

Email: q_tong@uncg.edu
Office: Petty 152
Office Hours: Thursday 3 pm – 4:30 pm @ Petty 152 or appointment by email

B.S. and M.S
in Mathematics

Ph.D. degree in Department of CS from University at Connecticut, UCONN in December 2022

Tenure-track Assistant Professor in Department of CS, UNCG

Research: Machine Learning; Stochastic Optimization; Deep Learning; Differential Privacy; Sparse Learning; Graph Convolutional Network (GCN) and Federated Learning.

Course Schedule

Week 13: April 3 topic: Random Forest
April 5 topic: Validation
Project stage IV and V released, and the ddl will be April 28.

Week 14: April 10 topic: PCA
April 12 topic: Clustering-Kmeans

Week 15: April 17 topic: Visualization
April 19 topic: Visualization
HW3 will be released, and the ddl will be April 28.

Week 16: April 24 Project Presentation (4 groups, each will have 15-20 min)
April 26 Project Presentation (4 groups, each will have 15-20 min)
All reports and homework must be submitted by April 28, and graded by the final week.

Why Data Science?

1. Demand: Data Scientists are in High Demand
Earlier this year, Coding Dojo named data science as the third most in-demand tech job in the U.S. coming in at No. 4 on the list is machine learning engineer, a mid- to senior-level data science job. With great earning potential, thousands of jobs available, and the utter importance of this career, data scientists are highly sought after, making for a great career choice.

2. High Salary: Data Science Careers Have High Earning Potential
There is a lot of money to be made in data science. With base salaries starting at about $100,000 and averaging out around $150K, a career in data science is not only lucrative but fulfilling, rewarding, and challenging.

3. Job Security: Data Science is a Fast-Growing Field
This year, LinkedIn released its annual U.S. emerging jobs report—and data scientists came out on top at 2nd place with 6.5x growth. While it didn't rank as the #1 fastest-growing field, it certainly cropped up pretty quickly.

Why Data Science?

4. Opportunity: Data Science Has a Range of Job Opportunities

According to the U.S. Bureau of Labor Statistics, studying data science gives you access to a diverse range of job opportunities. Some of these employment options include:

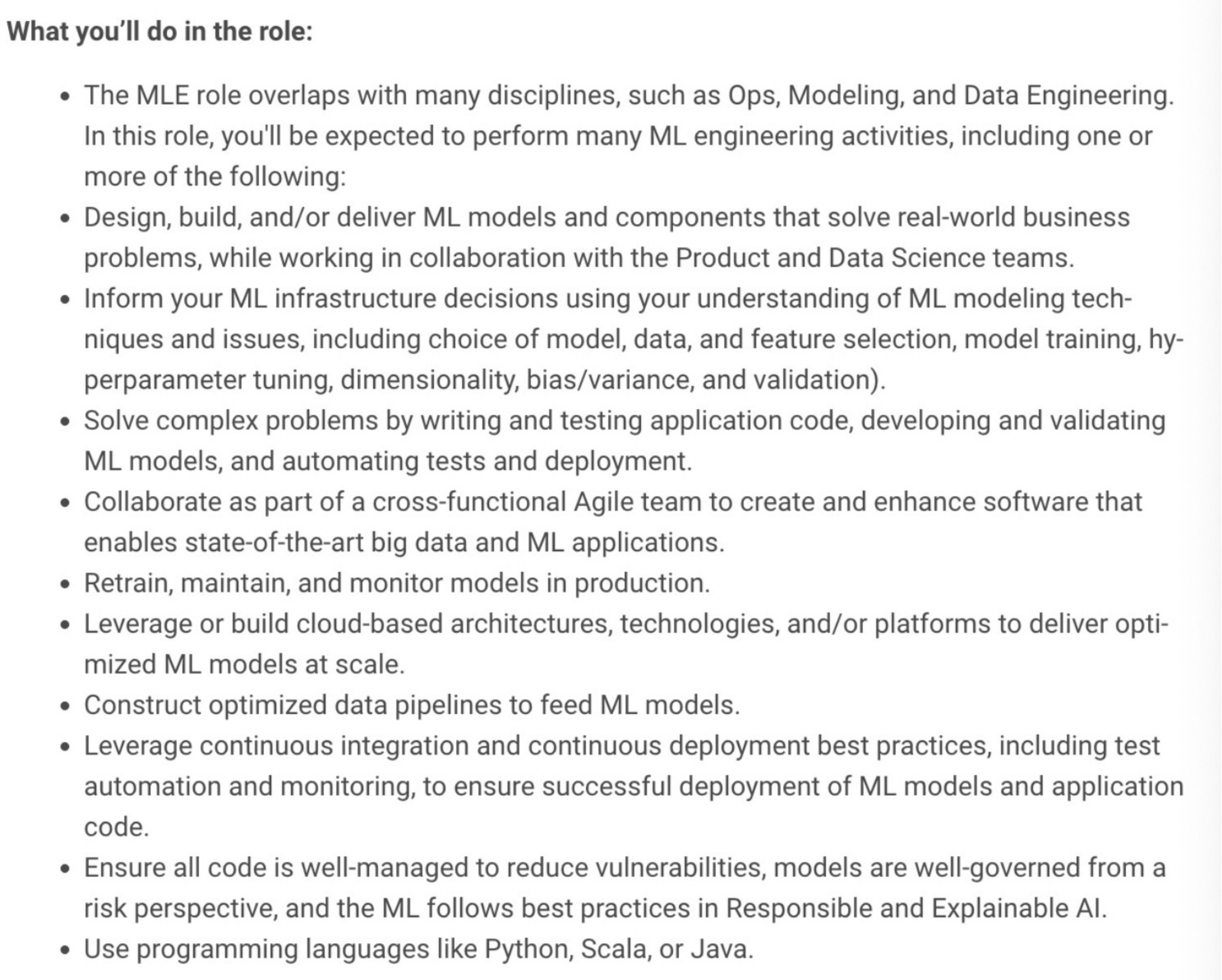
- Data Scientist
- Data Analyst
- Data Engineer
- Data Architect
- Business Analyst
- Software Engineer
- Machine Learning Engineer

5. Flexibility: Data Scientists are Needed in Various Sectors

Data scientists are in high demand across a range of industries and sectors. The need will only grow as businesses integrate more data environments into their operations.

Why Data Science?

Requirements

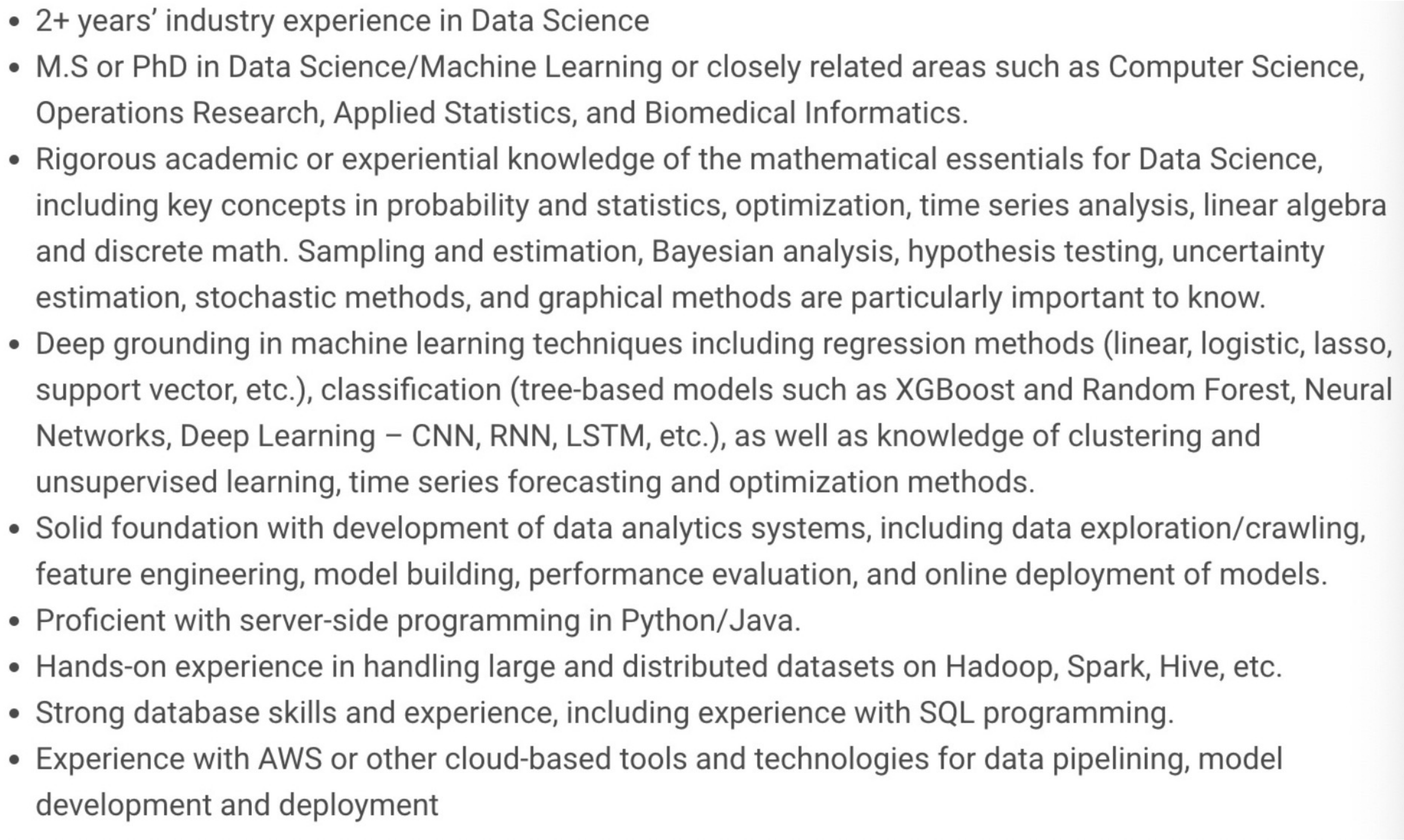


What you'll do in the role:

- The MLE role overlaps with many disciplines, such as Ops, Modeling, and Data Engineering. In this role, you'll be expected to perform many ML engineering activities, including one or more of the following:
- Design, build, and/or deliver ML models and components that solve real-world business problems, while working in collaboration with the Product and Data Science teams.
- Inform your ML infrastructure decisions using your understanding of ML modeling techniques and issues, including choice of model, data, and feature selection, model training, hyperparameter tuning, dimensionality, bias/variance, and validation).
- Solve complex problems by writing and testing application code, developing and validating ML models, and automating tests and deployment.
- Collaborate as part of a cross-functional Agile team to create and enhance software that enables state-of-the-art big data and ML applications.
- Retrain, maintain, and monitor models in production.
- Leverage or build cloud-based architectures, technologies, and/or platforms to deliver optimized ML models at scale.
- Construct optimized data pipelines to feed ML models.
- Leverage continuous integration and continuous deployment best practices, including test automation and monitoring, to ensure successful deployment of ML models and application code.
- Ensure all code is well-managed to reduce vulnerabilities, models are well-governed from a risk perspective, and the ML follows best practices in Responsible and Explainable AI.
- Use programming languages like Python, Scala, or Java.

Why Data Science?

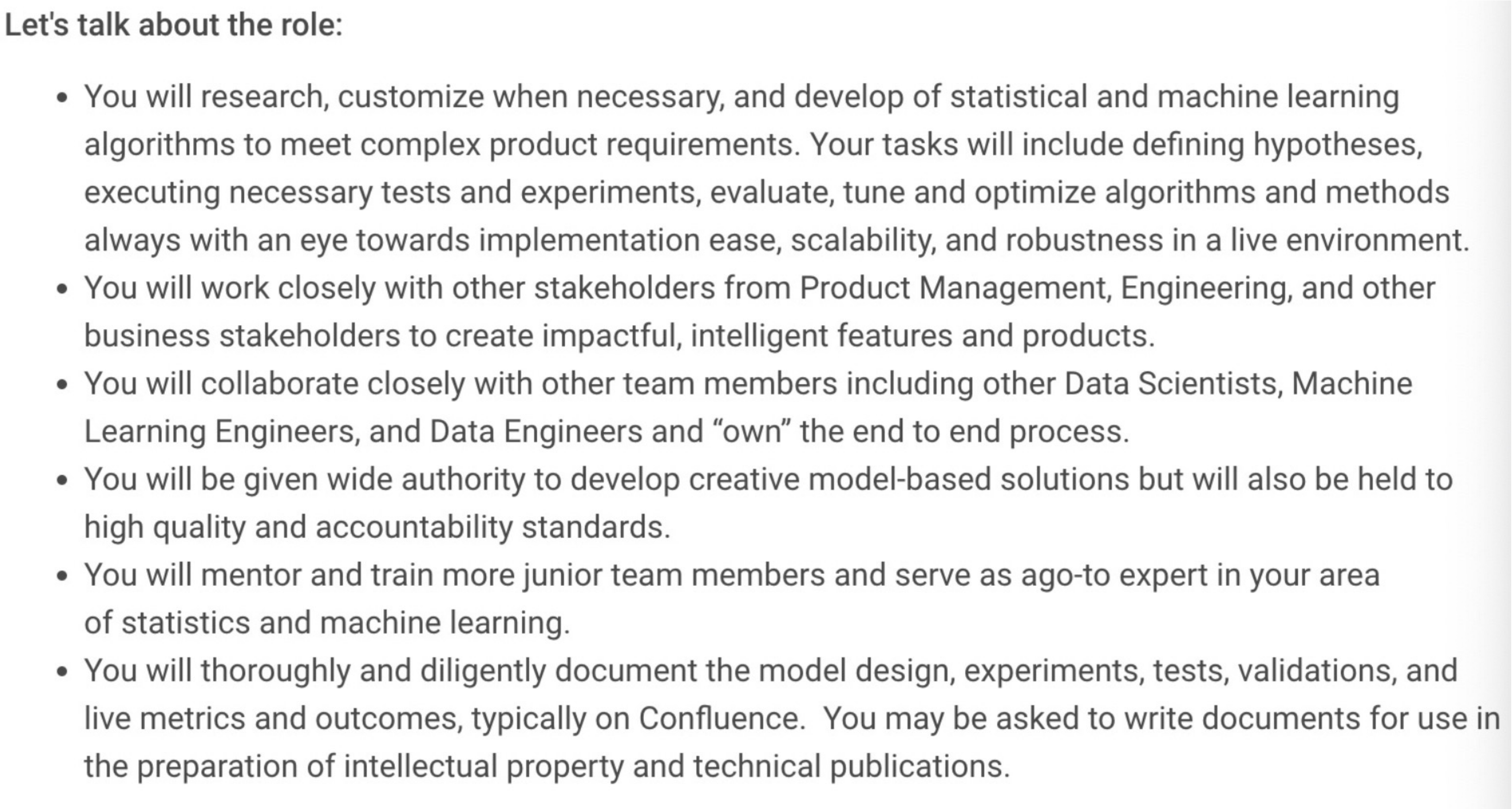
Requirements



- 2+ years' industry experience in Data Science
- M.S or PhD in Data Science/Machine Learning or closely related areas such as Computer Science, Operations Research, Applied Statistics, and Biomedical Informatics.
- Rigorous academic or experiential knowledge of the mathematical essentials for Data Science, including key concepts in probability and statistics, optimization, time series analysis, linear algebra and discrete math. Sampling and estimation, Bayesian analysis, hypothesis testing, uncertainty estimation, stochastic methods, and graphical methods are particularly important to know.
- Deep grounding in machine learning techniques including regression methods (linear, logistic, lasso, support vector, etc.), classification (tree-based models such as XGBoost and Random Forest, Neural Networks, Deep Learning – CNN, RNN, LSTM, etc.), as well as knowledge of clustering and unsupervised learning, time series forecasting and optimization methods.
- Solid foundation with development of data analytics systems, including data exploration/crawling, feature engineering, model building, performance evaluation, and online deployment of models.
- Proficient with server-side programming in Python/Java.
- Hands-on experience in handling large and distributed datasets on Hadoop, Spark, Hive, etc.
- Strong database skills and experience, including experience with SQL programming.
- Experience with AWS or other cloud-based tools and technologies for data pipelining, model development and deployment

Why Data Science?

Requirements



Let's talk about the role:

- You will research, customize when necessary, and develop of statistical and machine learning algorithms to meet complex product requirements. Your tasks will include defining hypotheses, executing necessary tests and experiments, evaluate, tune and optimize algorithms and methods always with an eye towards implementation ease, scalability, and robustness in a live environment.
- You will work closely with other stakeholders from Product Management, Engineering, and other business stakeholders to create impactful, intelligent features and products.
- You will collaborate closely with other team members including other Data Scientists, Machine Learning Engineers, and Data Engineers and "own" the end to end process.
- You will be given wide authority to develop creative model-based solutions but will also be held to high quality and accountability standards.
- You will mentor and train more junior team members and serve as ago-to expert in your area of statistics and machine learning.
- You will thoroughly and diligently document the model design, experiments, tests, validations, and live metrics and outcomes, typically on Confluence. You may be asked to write documents for use in the preparation of intellectual property and technical publications.

Course Schedule

Week 13: April 3 topic: Random Forest
April 5 topic: Validation
Project stage IV and V released, and the ddl will be April 28.

Week 14: April 10 topic: PCA
April 12 topic: Clustering-Kmeans

Week 15: April 17 topic: Visualization
April 19 topic: Visualization
HW3 will be released, and the ddl will be April 28.

Week 16: April 24 Project Presentation (4 groups, each will have 15-20 min)
April 26 Project Presentation (4 groups, each will have 15-20 min)
All reports and homework must be submitted by April 28, and graded by the final week.

Random Forest

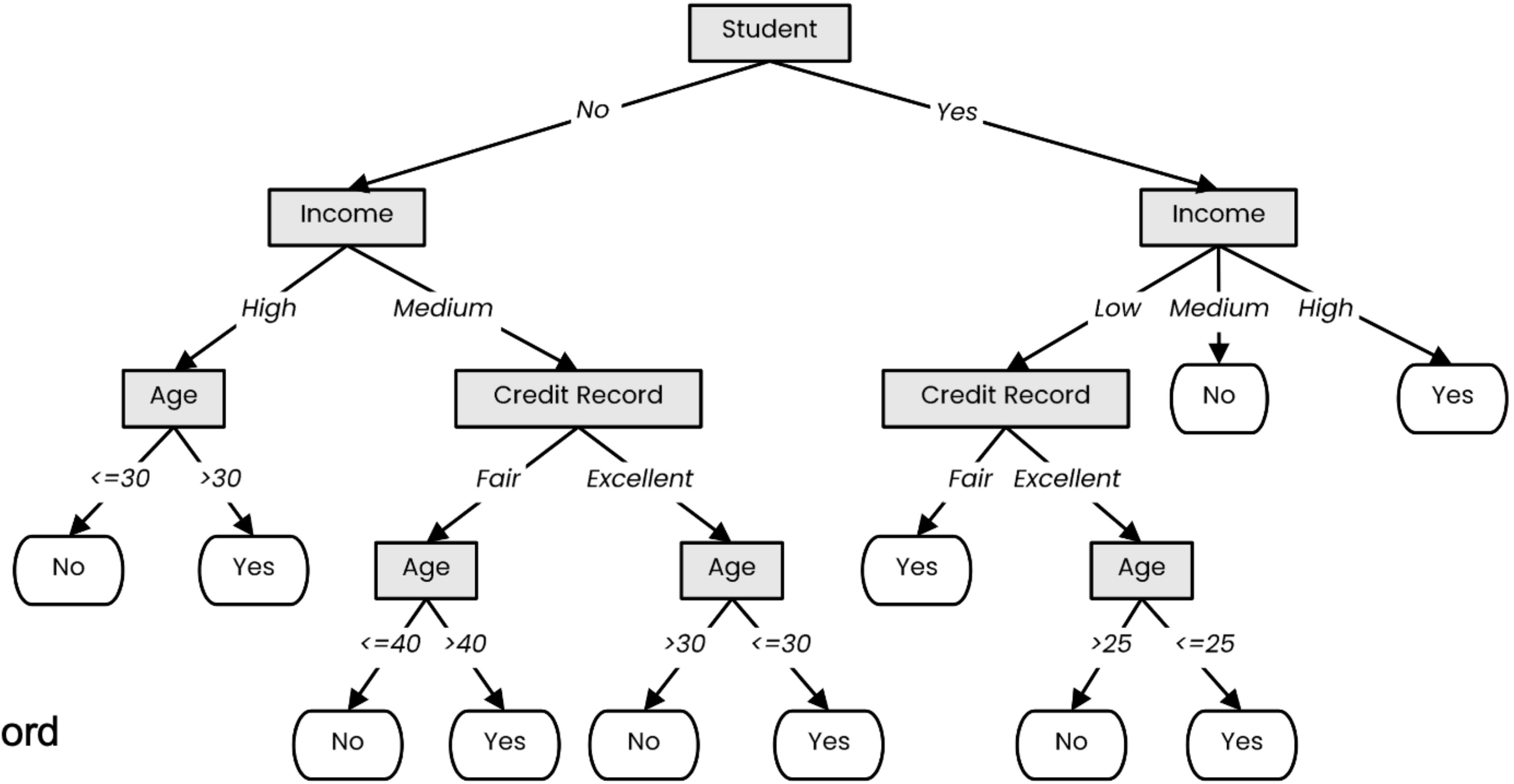
Recap Decision Tree

- A tree-like model that illustrates series of events leading to certain decisions
- Each node represents a test on an attribute and each branch is an outcome of that test

Who to loan?

- Not a student
- 45 years old
- Medium income
- Fair credit record

- Student
- 27 years old
- Low income
- Excellent credit record



Student
No
Yes
Income
High
Medium
Age
<=30
No
>30
Yes
Credit Record
Fair
Age
<=40
No
>40
Yes
Excellent
Age
>30
No
<=30
Yes
Income
Low
Credit Record
Fair
Yes
Excellent
Age
>25
No
<=25
Yes
Medium
No
High
Yes

Random Forest

Recap Decision Tree

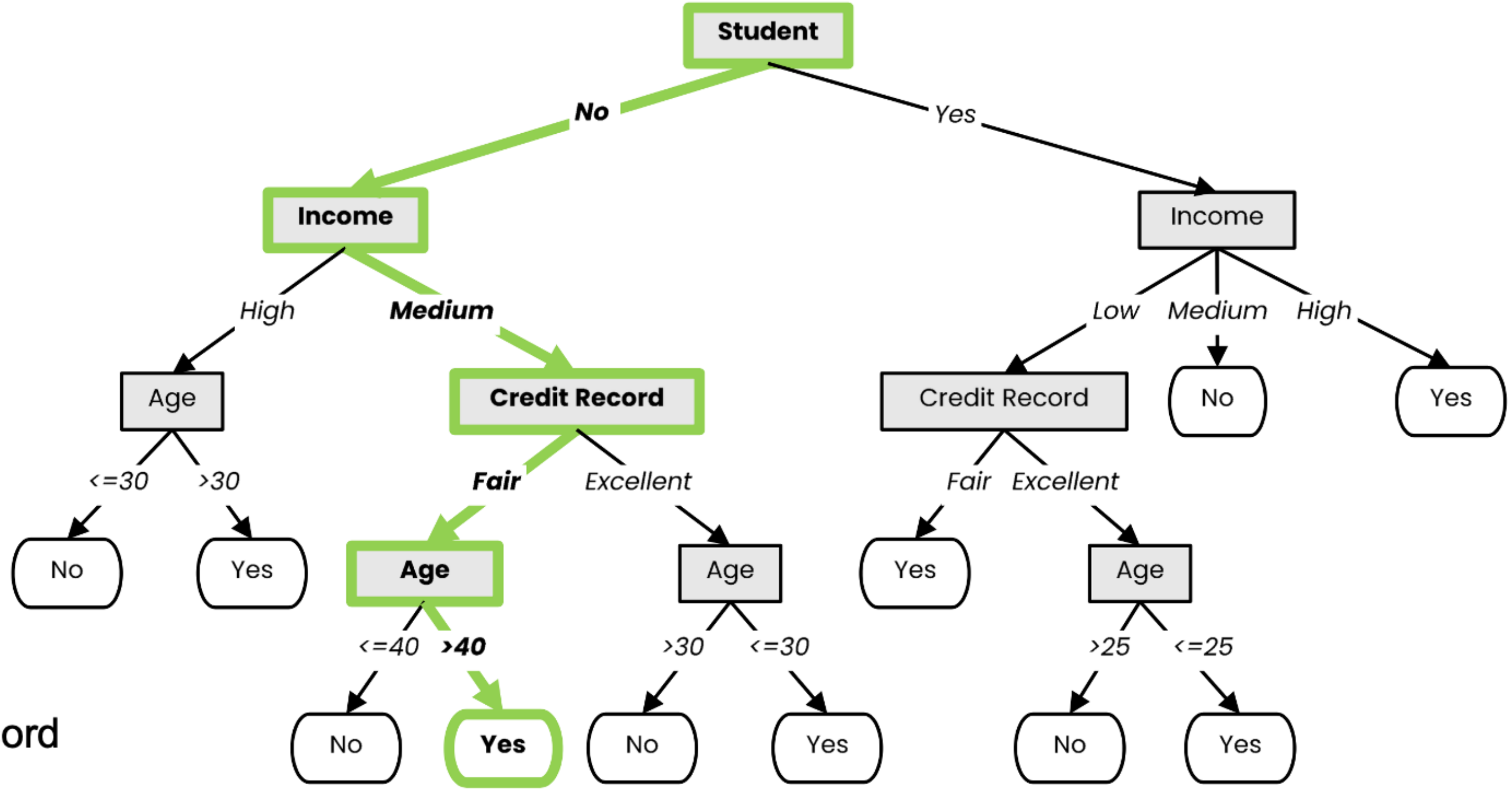
- A tree-like model that illustrates series of events leading to certain decisions
- Each node represents a test on an attribute and each branch is an outcome of that test

Who to loan?

- Not a student
- 45 years old
- Medium income
- Fair credit record

➢ Yes

- Student
- 27 years old
- Low income
- Excellent credit record

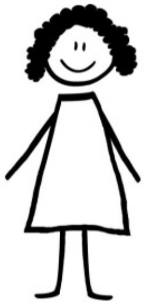


Random Forest

Recap Decision Tree

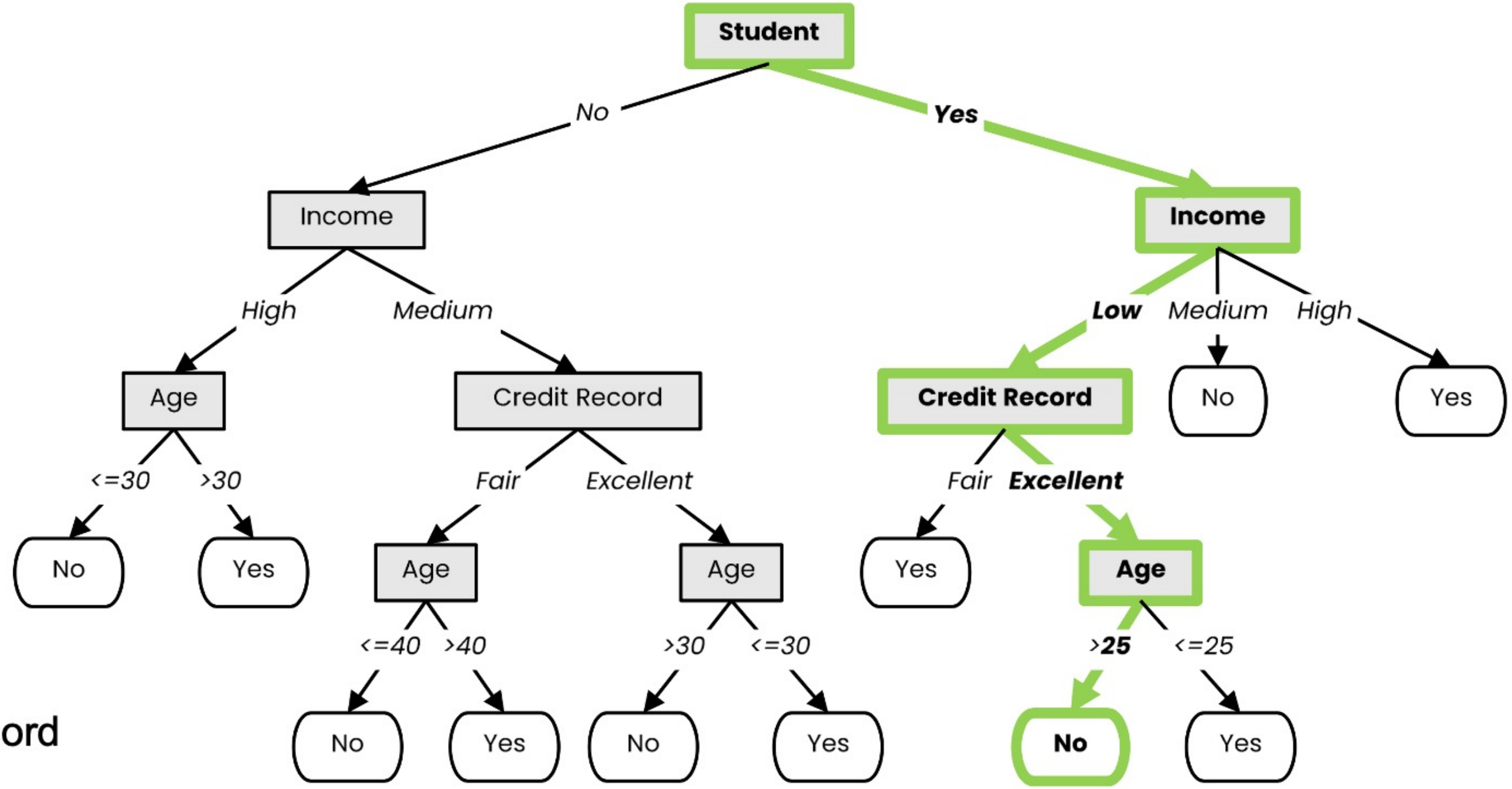
- A tree-like model that illustrates series of events leading to certain decisions
- Each node represents a test on an attribute and each branch is an outcome of that test

Who to loan?



- Not a student
- 45 years old
- Medium income
- Fair credit record
- ➢ Yes

- Student
- 27 years old
- Low income
- Excellent credit record
- ➢ No



Random Forest

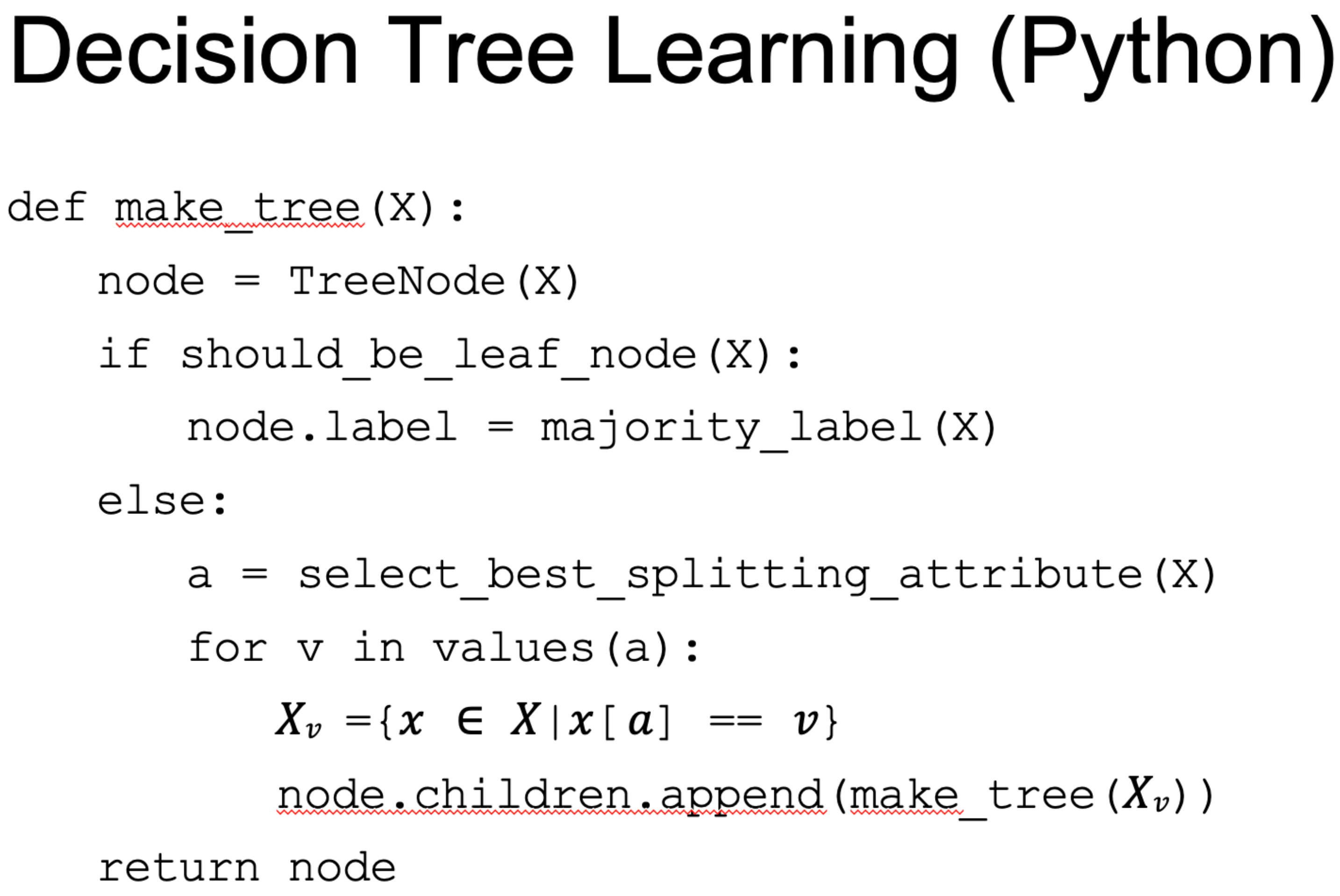
Recap Decision Tree

- We use labeled data to obtain a suitable decision tree for future predictions
  - ➢ We want a decision tree that works well on unseen data, while asking as few questions as possible

- Basic step: choose an attribute and, based on its values, split the data into smaller sets
  - ➢ Recursively repeat this step until we can surely decide the label

Random Forest

Recap Decision Tree



Decision Tree Learning (Python)

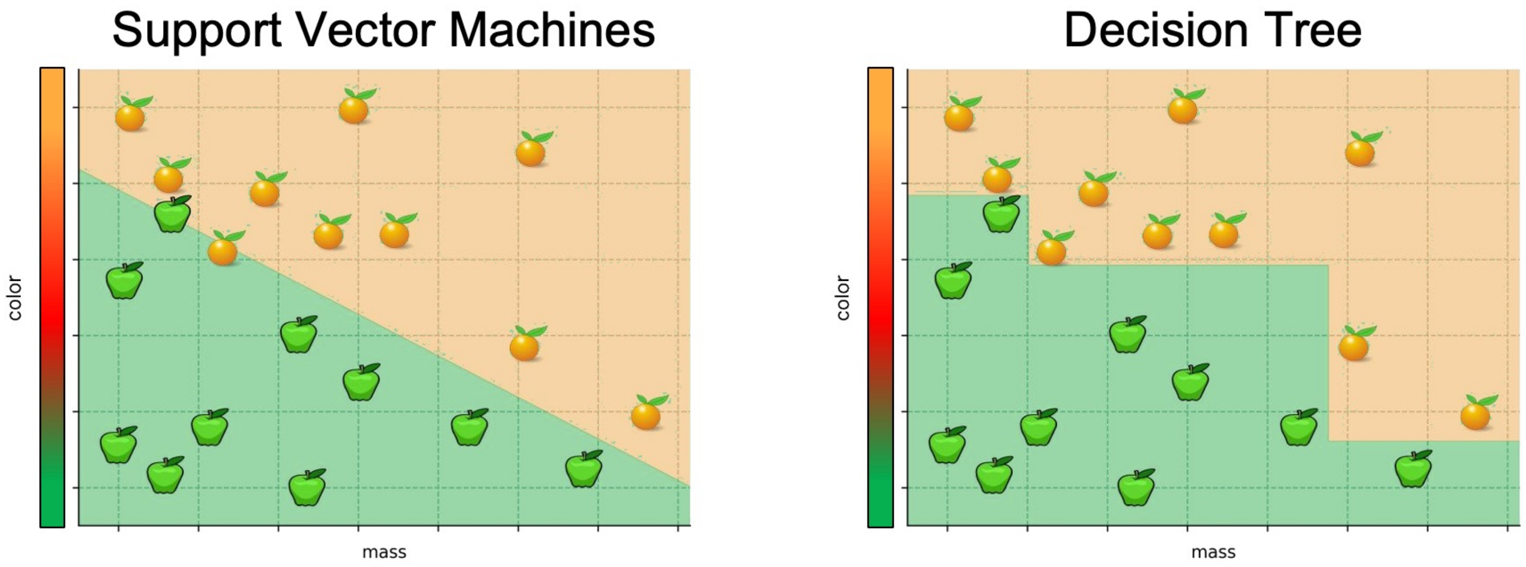
```
def make_tree(X):
    node = TreeNode(X)
    if should_be_leaf_node(X):
        node.label = majority_label(X)
    else:
        a = select_best_splitting_attribute(X)
        for v in values(a):
            $X_v = \{x \in X | x[a] == v\}$
            node.children.append(make_tree($X_v$))
    return node
```

Random Forest

Recap Decision Tree

Decision Boundaries

- Decision trees produce non-linear decision boundaries



Random Forest

Recap Decision Tree

- Decision trees represent a tool based on a tree-like graph of decisions and their possible outcomes
- Decision tree learning is a machine learning method that employs a decision tree as a predictive model
- While decision trees classify quickly, the time for building a tree may be higher than another type of classifier
- Decision trees suffer from a problem of errors propagating throughout a tree
  A very serious problem as the number of classes increases
- **Decision Trees have very high variance**

Random Forest

Random Forests

(Ensemble learning with decision trees)

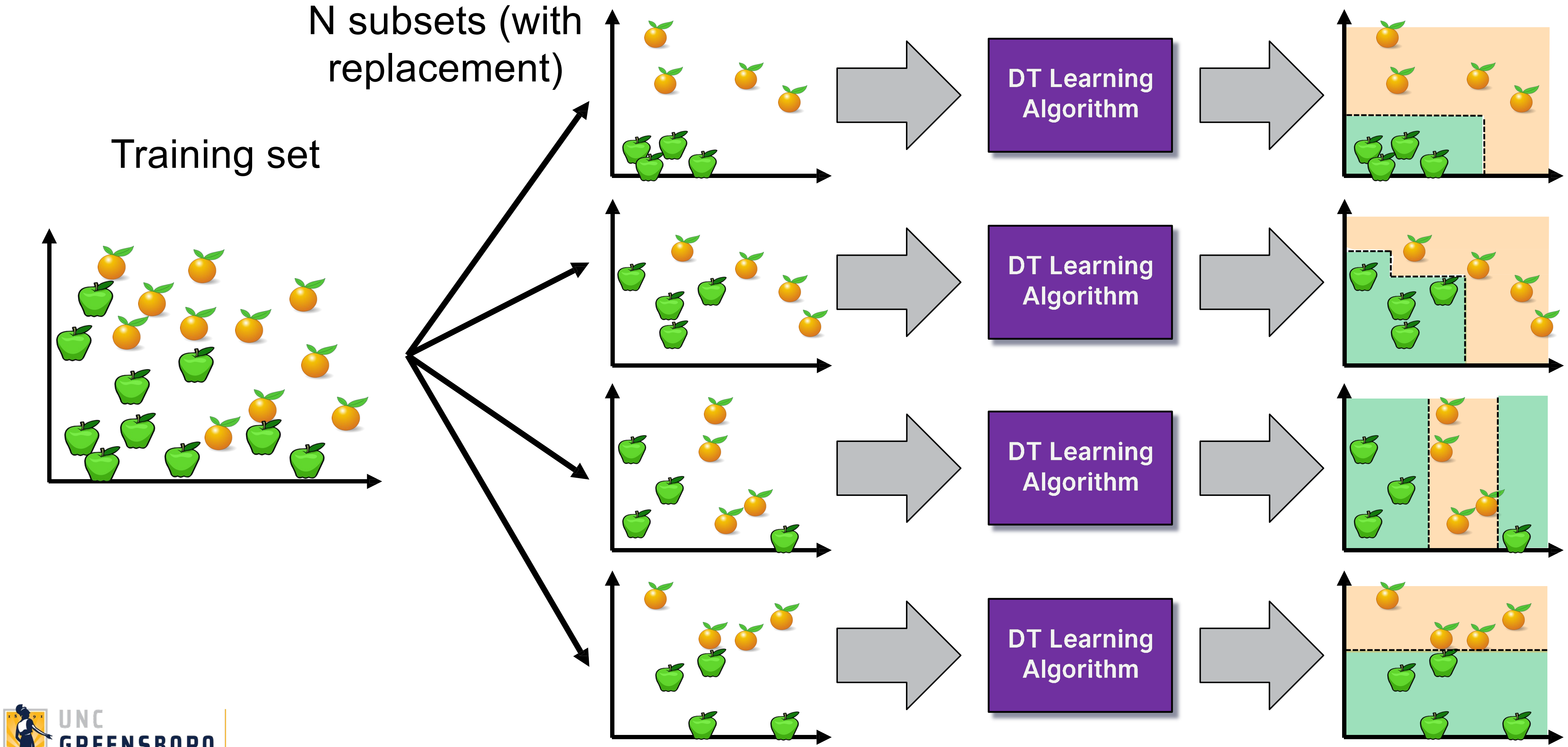
Random Forest

- Random Forests:
  - ➢ Instead of building a single decision tree and use it to make predictions, build many slightly different trees and combine their predictions
- We have a single data set, so how do we obtain slightly different trees?

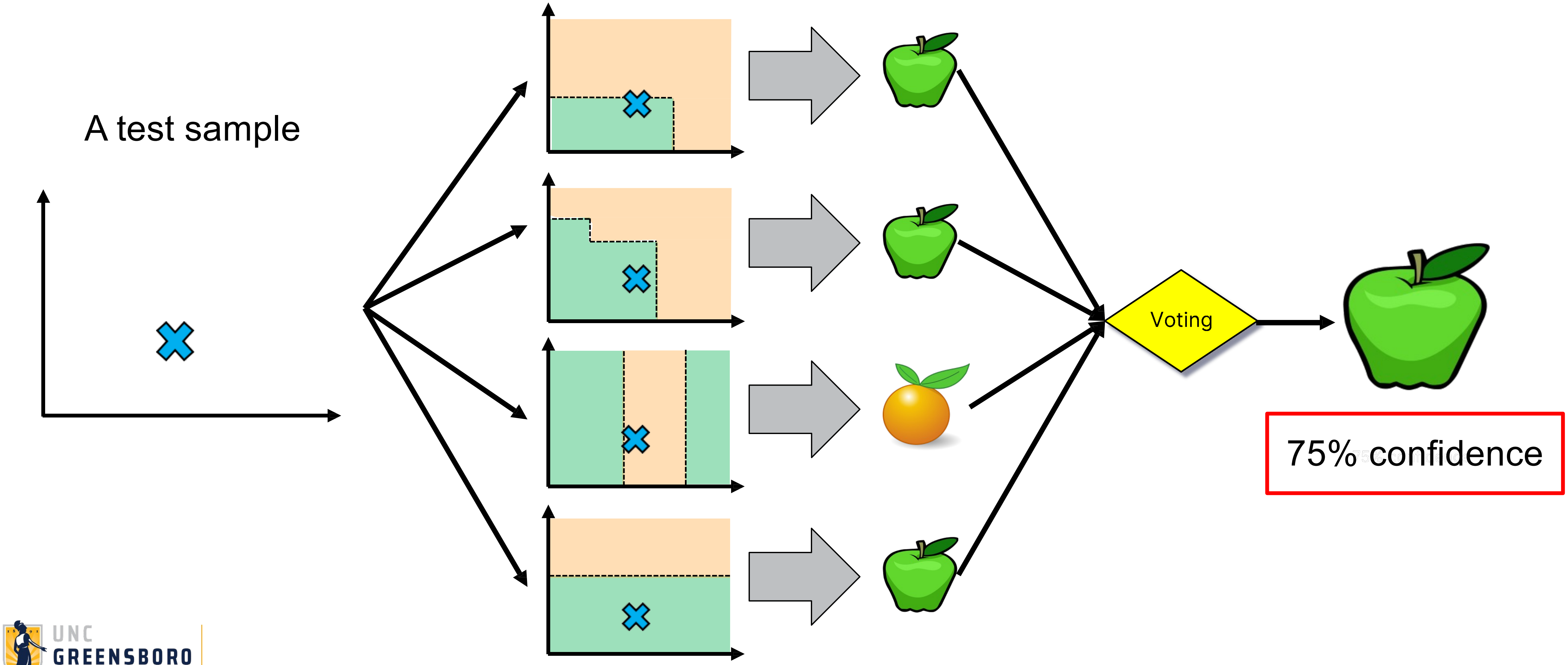
1. Bagging (**B**ootstrap **Agg**regat**ing**):
   - ➢ Take random subsets of data points from the training set to create N smaller data sets
   - ➢ Fit a decision tree on each subset

2. Random Subspace Method (also known as Feature Bagging):
   - ➢ Fit N different decision trees by constraining each one to operate on a random subset of features

Bagging at training time



Bagging at inference time



A test sample

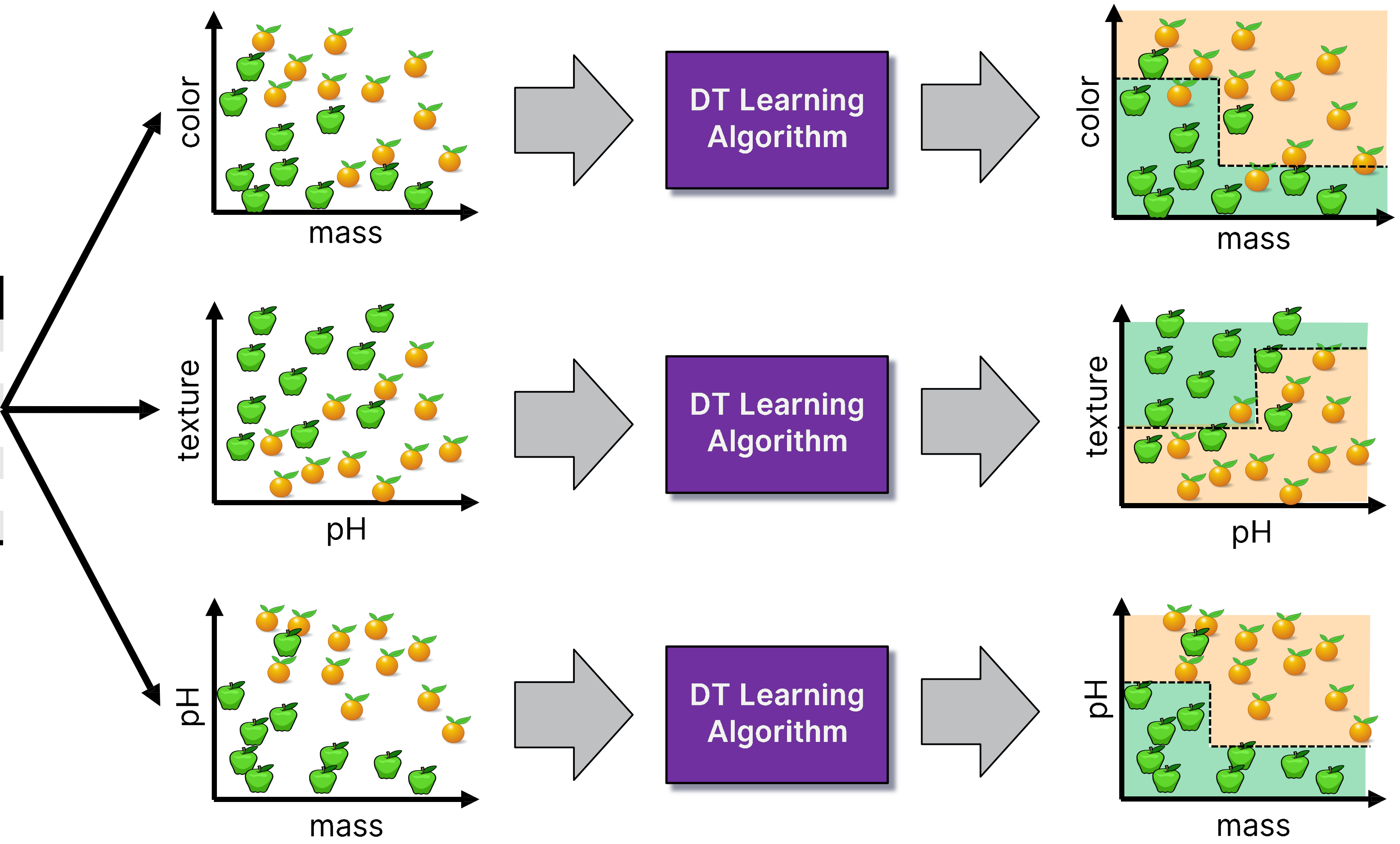
Voting

75% confidence

Random Subspace Method at training time

Training data

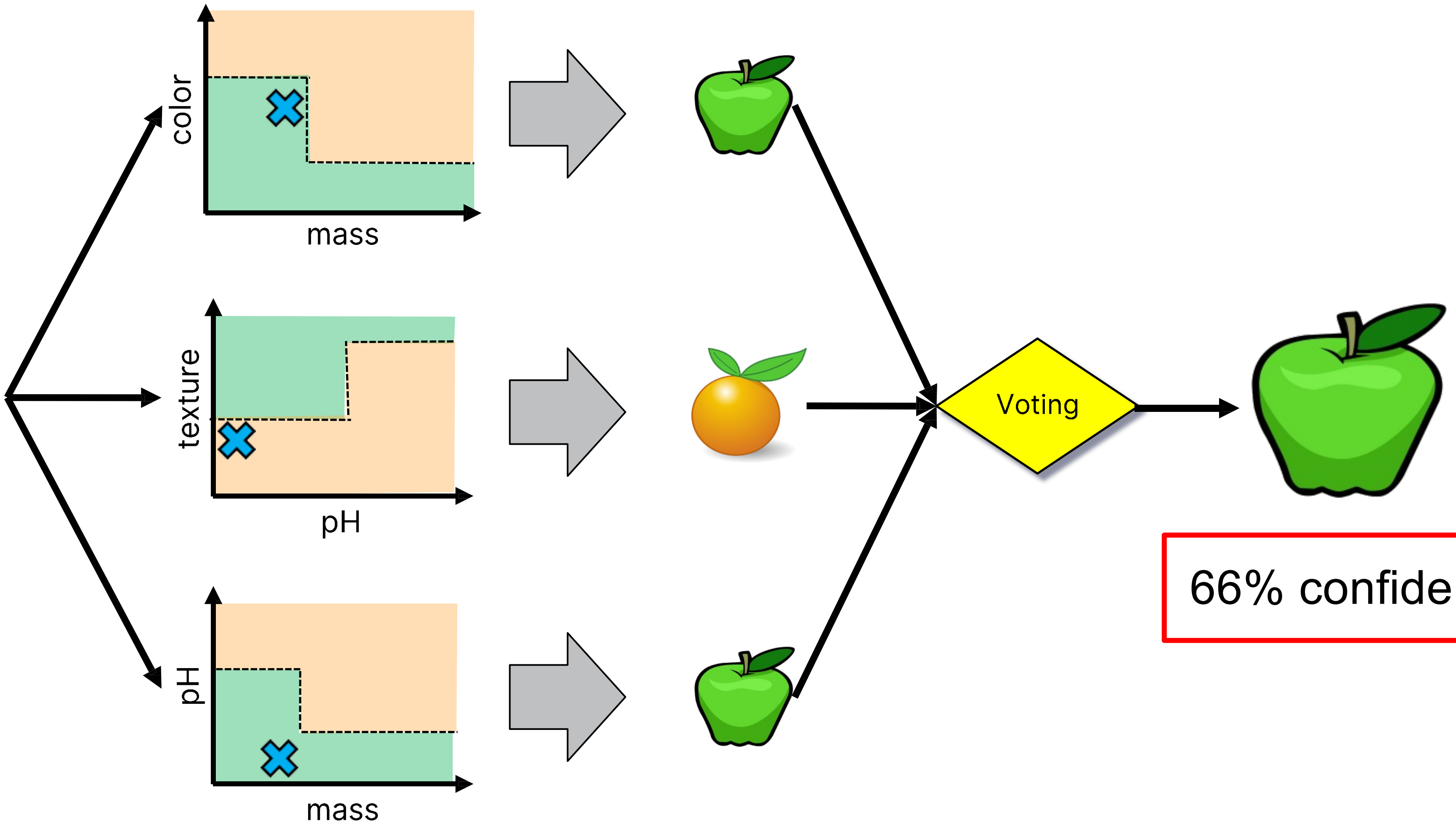
Mass (g)	Color	Texture	pH	Label
84	Green	Smooth	3.5	**Apple**
121	Orange	Rough	3.9	**Orange**
85	Red	Smooth	3.3	**Apple**
101	Orange	Smooth	3.7	**Orange**
111	Green	Rough	3.5	**Apple**
...				
117	Red	Rough	3.4	**Orange**



Random Subspace Method at inference time

A test sample

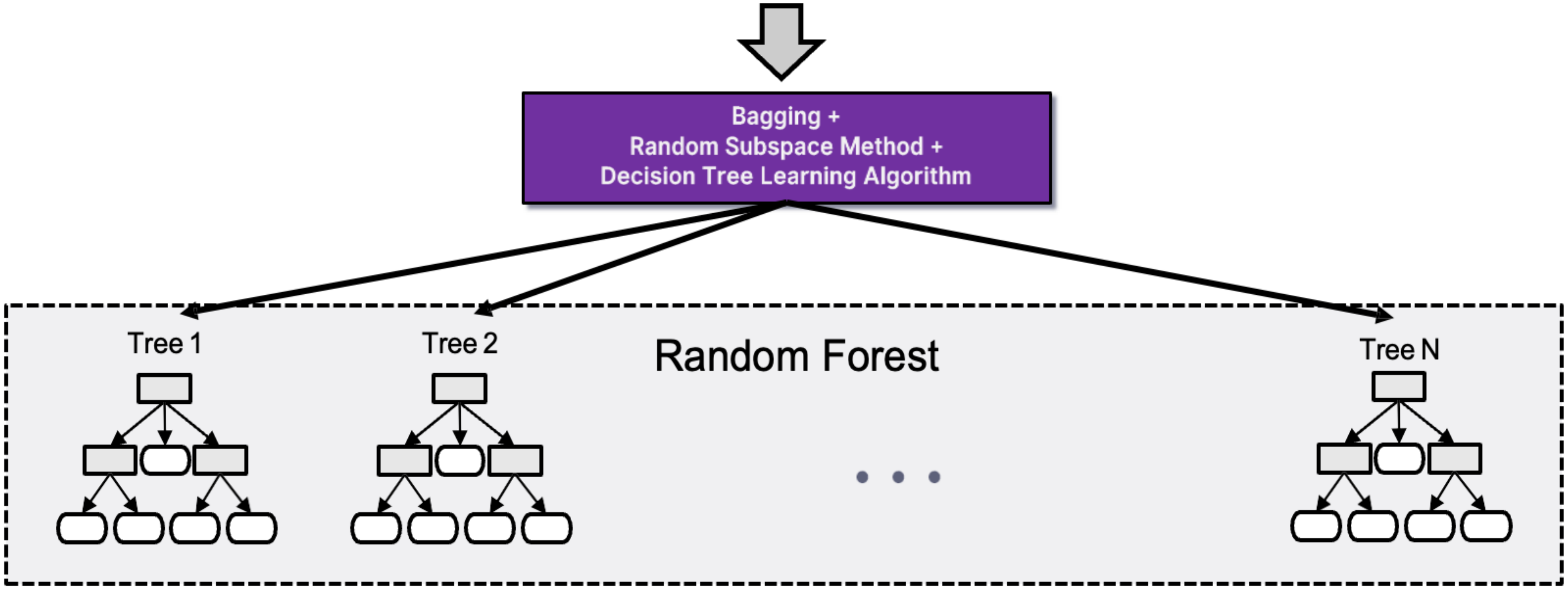
87	Red	Smooth	3.1



66% confidence

Random Forest

Mass (g)	Color	Texture	pH	Label
84	Green	Smooth	3.5	**Apple**
121	Orange	Rough	3.9	**Orange**
85	Red	Smooth	3.3	**Apple**
101	Orange	Smooth	3.7	**Orange**
111	Green	Rough	3.5	**Apple**
...				
117	Red	Rough	3.4	**Orange**



Bagging +
Random Subspace Method +
Decision Tree Learning Algorithm

Random Forest

History of Random Forests

- Introduction of the Random Subspace Method
  - “Random Decision Forests” [Ho, 1995] and “The Random Subspace Method for Constructing Decision Forests” [Ho, 1998]

- Combined the Random Subspace Method with Bagging. Introduce the term Random Forest (a trademark of Leo Breiman and Adele Cutler)
  - “Random Forests” [Breiman, 2001]

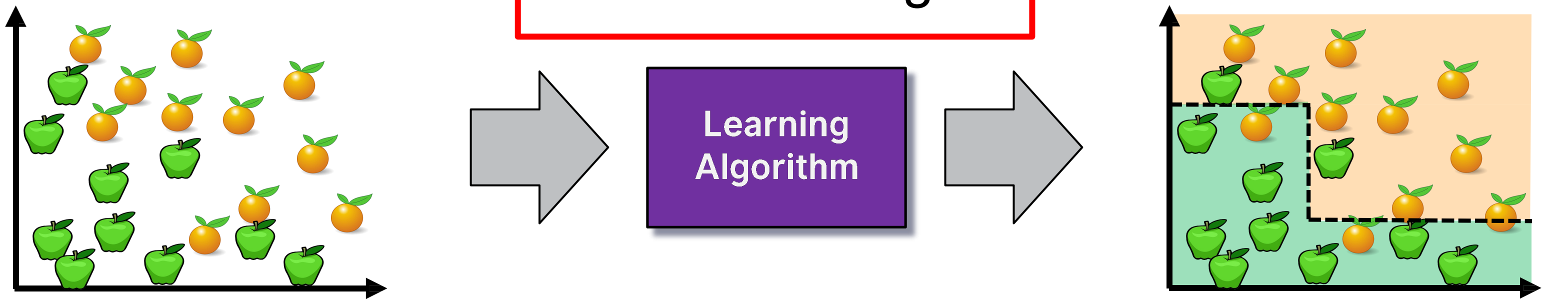
Random Forest

Ensemble Learning

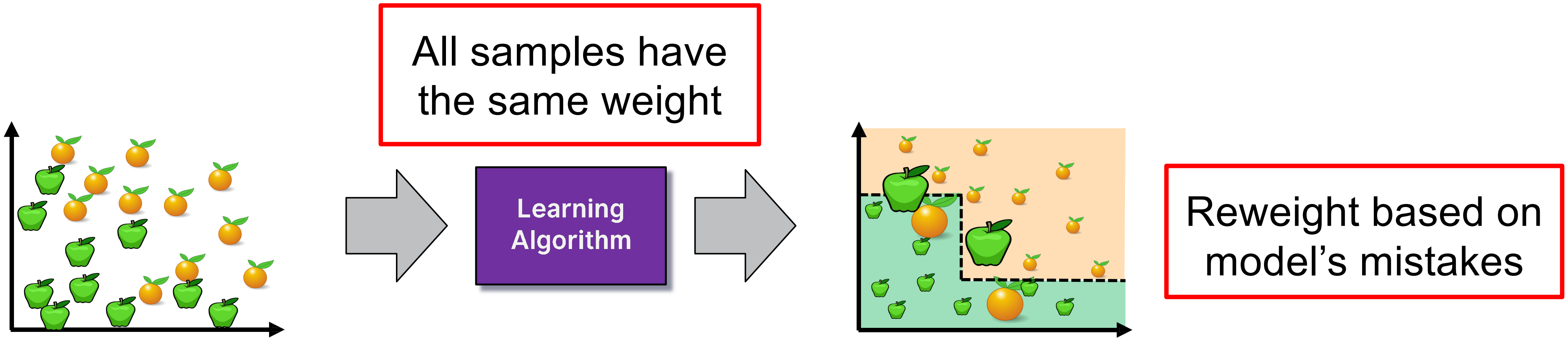
- Ensemble Learning:
  - ➢ Method that combines multiple learning algorithms to obtain performance improvements over its components
- **Random Forests** are one of the most common examples of ensemble learning
- Other commonly-used ensemble methods:
  - ➢ Bagging: multiple models on random subsets of data samples
  - ➢ Random Subspace Method: multiple models on random subsets of features
  - ➢ Boosting: train models iteratively, while making the current model focus on the mistakes of the previous ones by increasing the weight of misclassified samples

Boosting

All samples have the same weight

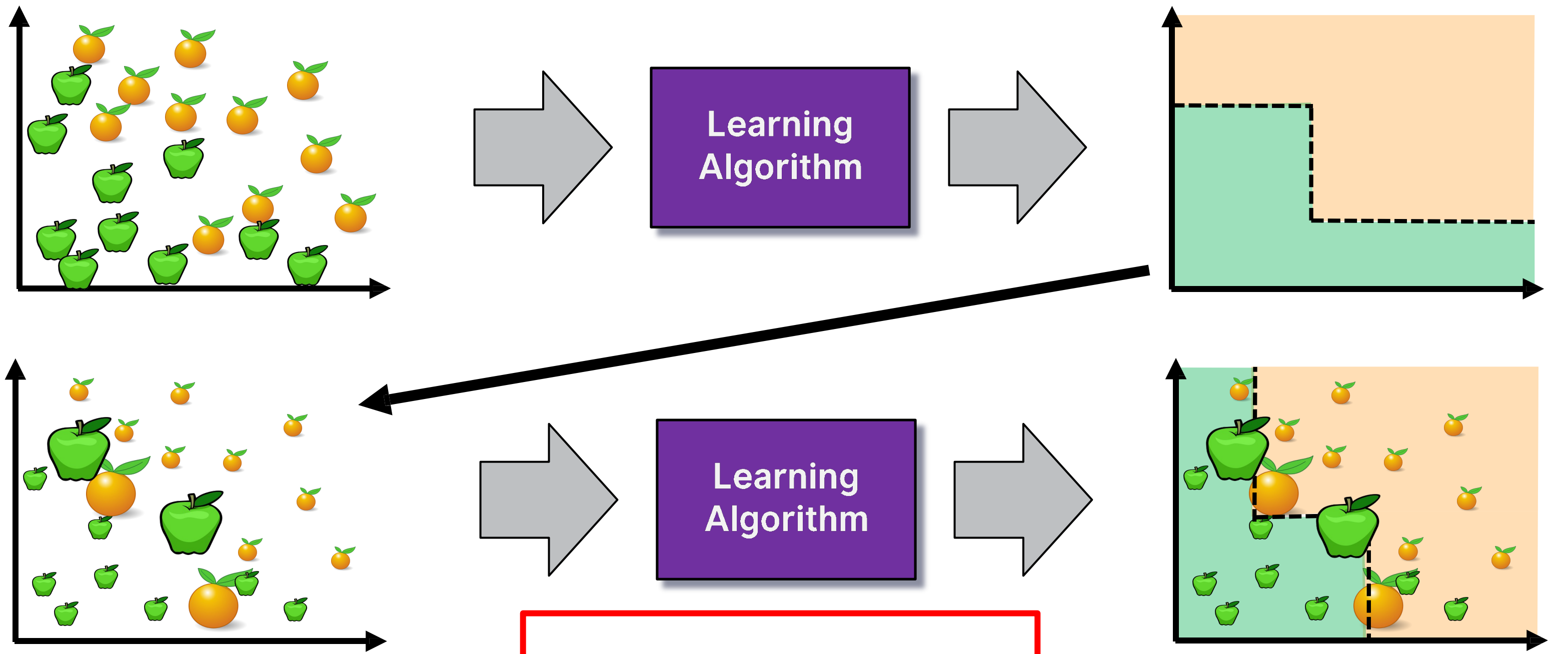


Boosting


All samples have the same weight

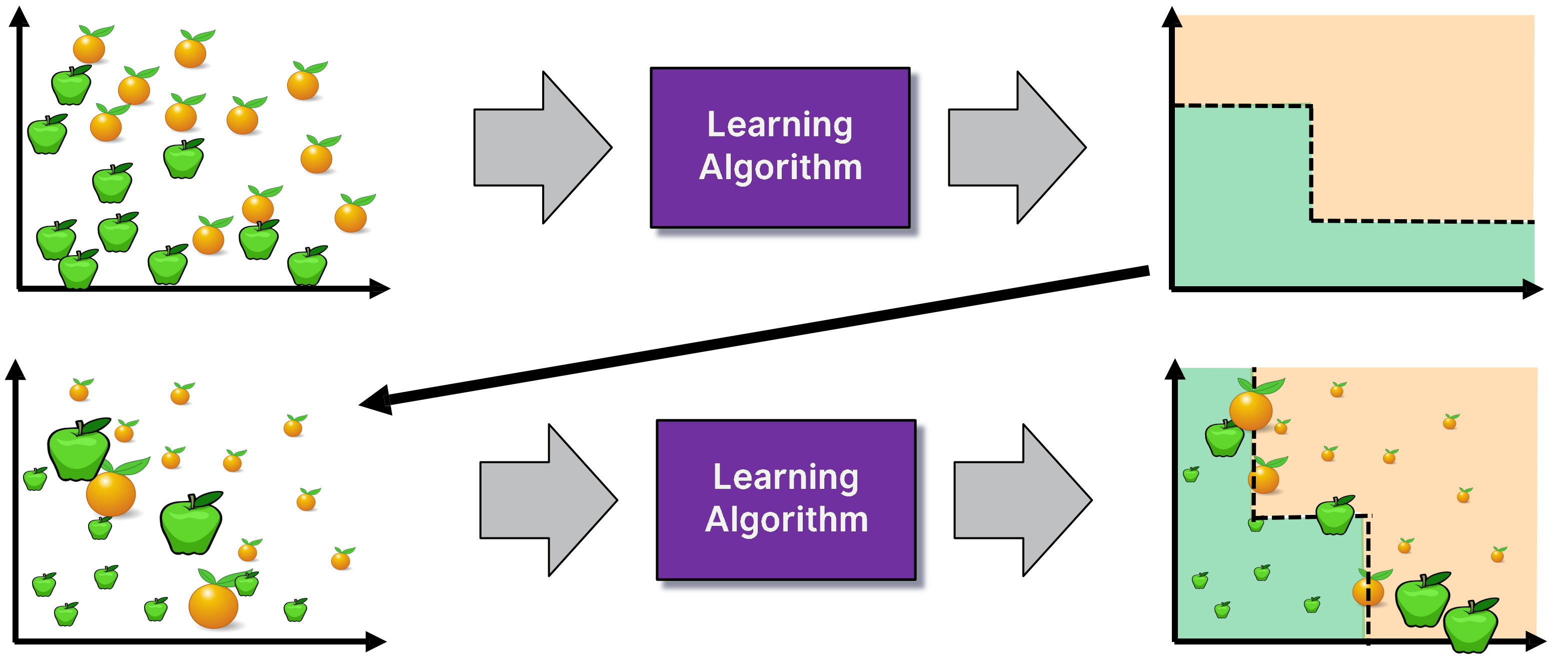
Reweight based on model’s mistakes

Boosting



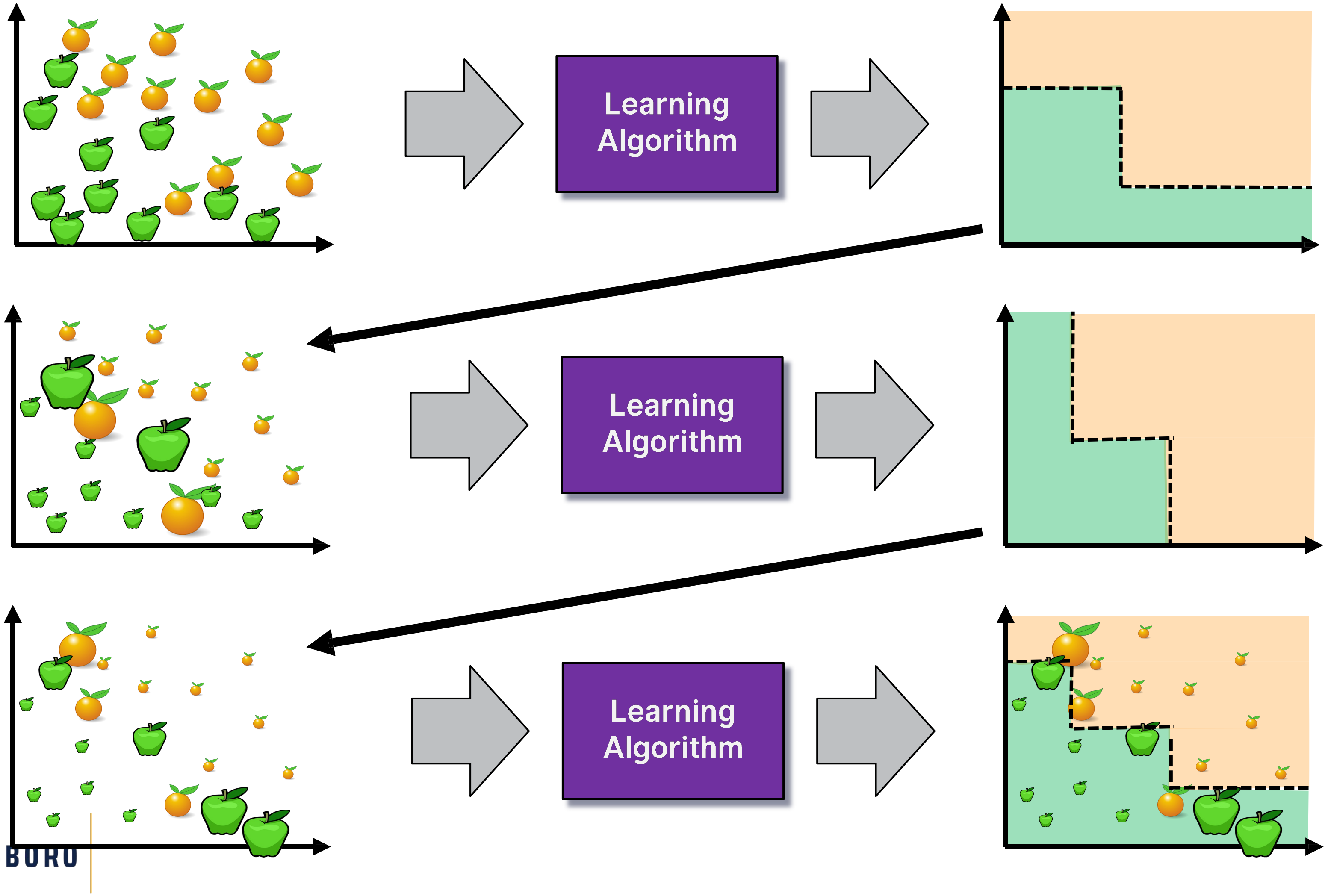
Next model sees weighted samples

Boosting

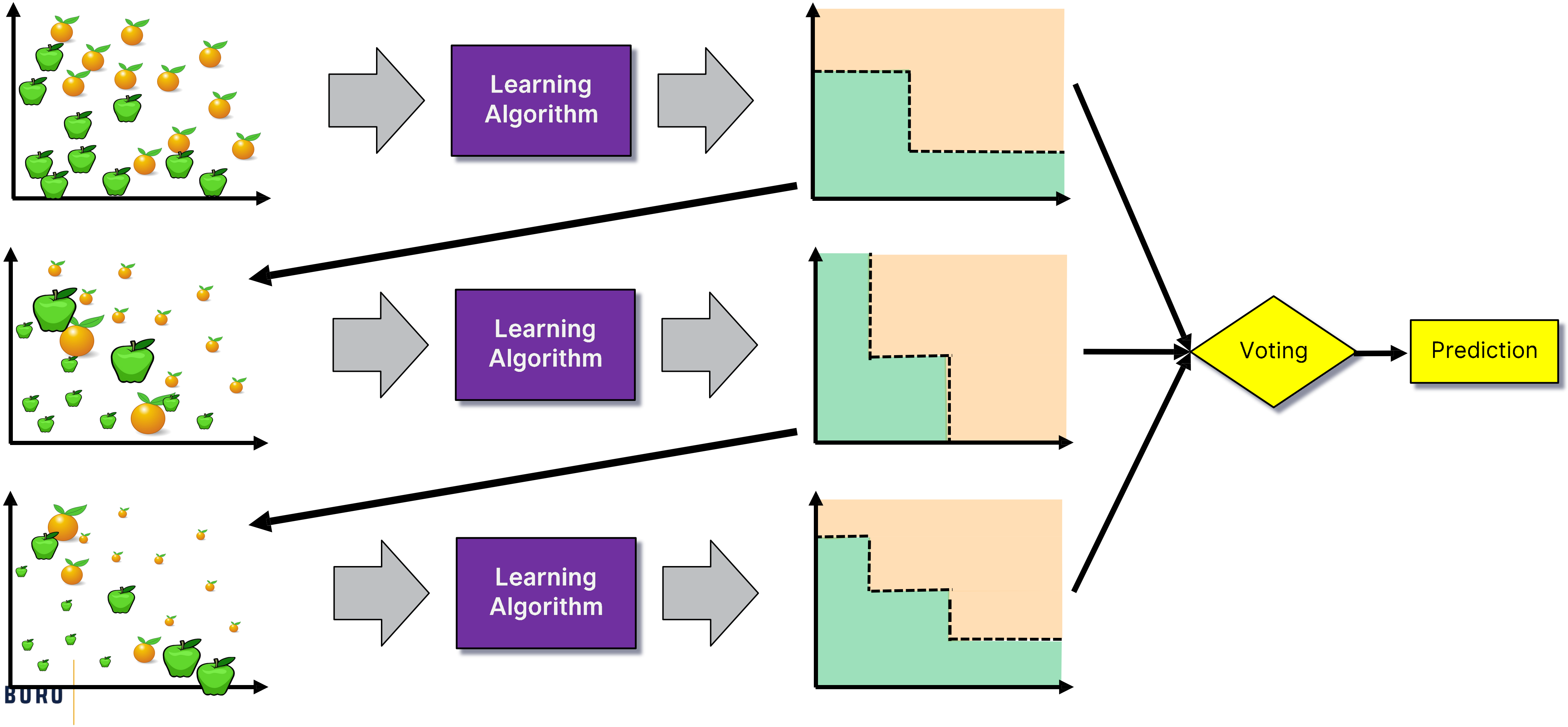


Reweight based on current model's mistakes

Boosting


Learning Algorithm
Learning Algorithm
Learning Algorithm

Boosting


Learning Algorithm
Learning Algorithm
Learning Algorithm
Voting
Prediction

Summary

- Ensemble Learning methods combine multiple learning algorithms to obtain performance improvements over its components
- Commonly-used ensemble methods:
  - Bagging (multiple models on random subsets of data samples)
  - Random Subspace Method (multiple models on random subsets of features)
  - Boosting (train models iteratively, while making the current model focus on the mistakes of the previous ones by increasing the weight of misclassified samples)
- **Random Forests** are an ensemble learning method that employ decision tree learning to build multiple trees through **bagging** and **random subspace method**.
  - They rectify the overfitting problem of decision trees!

Random Forest

Decision Trees and Random Forest (Python)

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier

clf = DecisionTreeClassifier(criterion = "entropy", min_samples_leaf = 3)
# Lots of parameters: criterion = "gini" / "entropy";
#                     max_depth;
#                     min_impurity_split;

clf.fit(X, y) # It can only handle numerical attributes!
# Categorical attributes need to be encoded, see LabelEncoder and OneHotEncoder

clf.predict([x]) # Predict class for x

clf.feature_importances_ # Importance of each feature
clf.tree_ # The underlying tree object

clf = RandomForestClassifier(n_estimators = 20) # Random Forest with 20 trees
```