



EXTREME GRADIENT BOOSTING WITH XGBOOST

**Welcome to the
course!**

Sergey Fogelson

VP of Analytics, Viacom



Before we get to XGBoost...

- Need to understand the basics of
 - Supervised classification
 - Decision trees
 - Boosting



Supervised learning

- Relies on labeled data
- Have some understanding of past behavior

Supervised learning example

- Does a specific image contain a person's face?



- Training data: vectors of pixel values
- Labels: 1 or 0



Supervised learning: Classification

- Outcome can be binary or multi-class



Binary classification example

- Will a person purchase the insurance package given some quote?



Multi-class classification example

- Classifying the species of a given bird

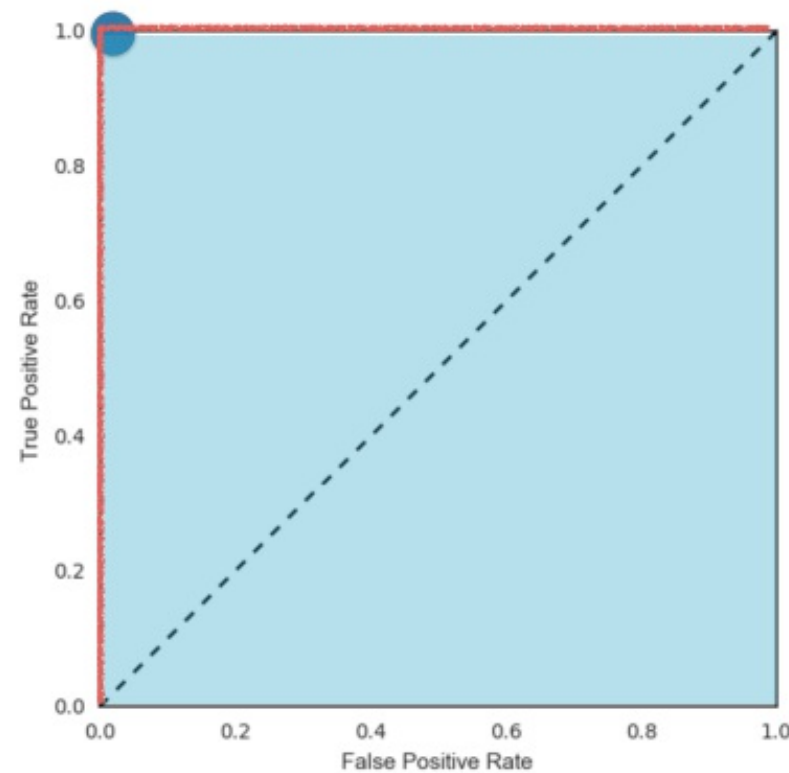




AUC: Metric for binary classification models

Area under the ROC curve (AUC)

- Larger area under the ROC curve = better model





Accuracy score and confusion matrix

- Confusion matrix

	Predicted: Spam Email	Predicted: Real Email
Actual: Spam Email	True Positive	False Negative
Actual: Real Email	False Positive	True Negative

- Accuracy: $\frac{tp + tn}{tp + tn + fp + fn}$



Supervised learning with scikit-learn

PAID COURSE

Supervised Learning with scikit-learn

[Start Course For Free](#)[▶ Play Intro Video](#)

4 hours

|

17 Videos

|

54 Exercises

|

18,961 Participants

|

4,300 XP



Other supervised learning considerations


- Features can be either numeric or categorical
- Numeric features should be scaled (Z-scored)
- Categorical features should be encoded (one-hot)



Ranking

- Predicting an ordering on a set of choices



gradient boos 

gradient boosting
gradient boosting regression
gradient boosting sklearn
gradient boosting explained
gradient boosting classifier
gradient boosting vs random forest
gradient boosting algorithm
gradient boosting tutorial
gradient boosting vs adaboost
gradient boosting decision tree

[Report inappropriate predictions](#)



Recommendation

- Recommending an item to a user
- Based on consumption history and profile
- Example: Netflix



EXTREME GRADIENT BOOSTING WITH XGBOOST

Let's get to work!



EXTREME GRADIENT BOOSTING WITH XGBOOST

Introducing XGBoost

Sergey Fogelson

VP of Analytics, Viacom



What is XGBoost?

- Optimized gradient-boosting machine learning library
- Originally written in C++
- Has APIs in several languages:
 - **Python**
 - R
 - Scala
 - Julia
 - Java



What makes XGBoost so popular?

- Speed and performance
- Core algorithm is parallelizable
- Consistently outperforms single-algorithm methods
- State-of-the-art performance in many ML tasks

Using XGBoost: A Quick Example

```
In [1]: import xgboost as xgb

In [2]: import pandas as pd

In [3]: import numpy as np

In [4]: from sklearn.model_selection import train_test_split

In [5]: class_data = pd.read_csv("classification_data.csv")

In [6]: X, y = class_data.iloc[:, :-1], class_data.iloc[:, -1]

In [7]: X_train, X_test, y_train, y_test= train_test_split(X, y,
    test_size=0.2, random_state=123)

In [8]: xg_cl = xgb.XGBClassifier(objective='binary:logistic',
    n_estimators=10, seed=123)

In [9]: xg_cl.fit(X_train, y_train)

In [10]: preds = xg_cl.predict(X_test)

In [11]: accuracy = float(np.sum(preds==y_test))/y_test.shape[0]

In [12]: print("accuracy: %f" % (accuracy))
accuracy: 0.78333
```



EXTREME GRADIENT BOOSTING WITH XGBOOST

**Let's begin using
XGBoost!**



EXTREME GRADIENT BOOSTING WITH XGBOOST

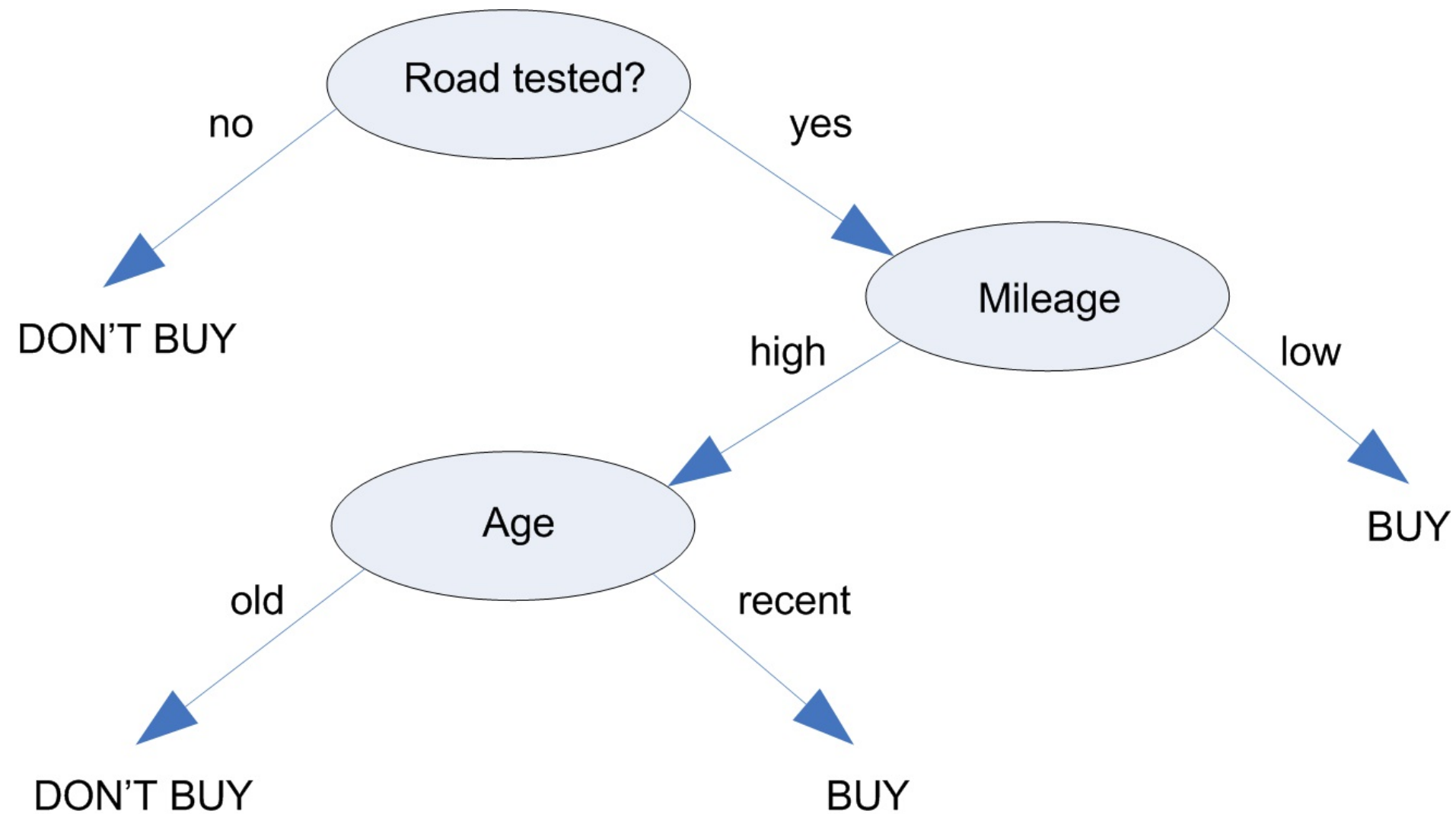
What is a decision tree?

Sergey Fogelson

VP of Analytics, Viacom



Visualizing a decision tree





Decision trees as base learners

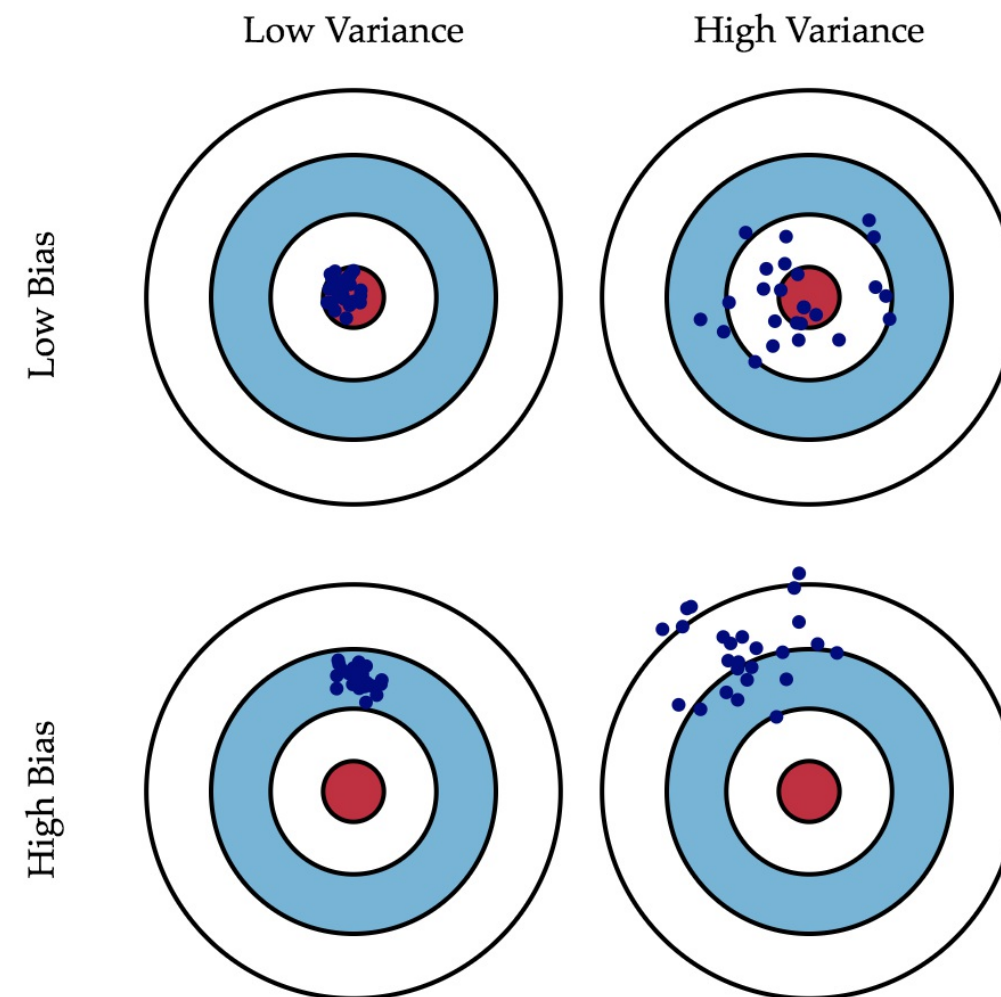
- Base learner - Individual learning algorithm in an ensemble algorithm
- Composed of a series of binary questions
- Predictions happen at the "leaves" of the tree



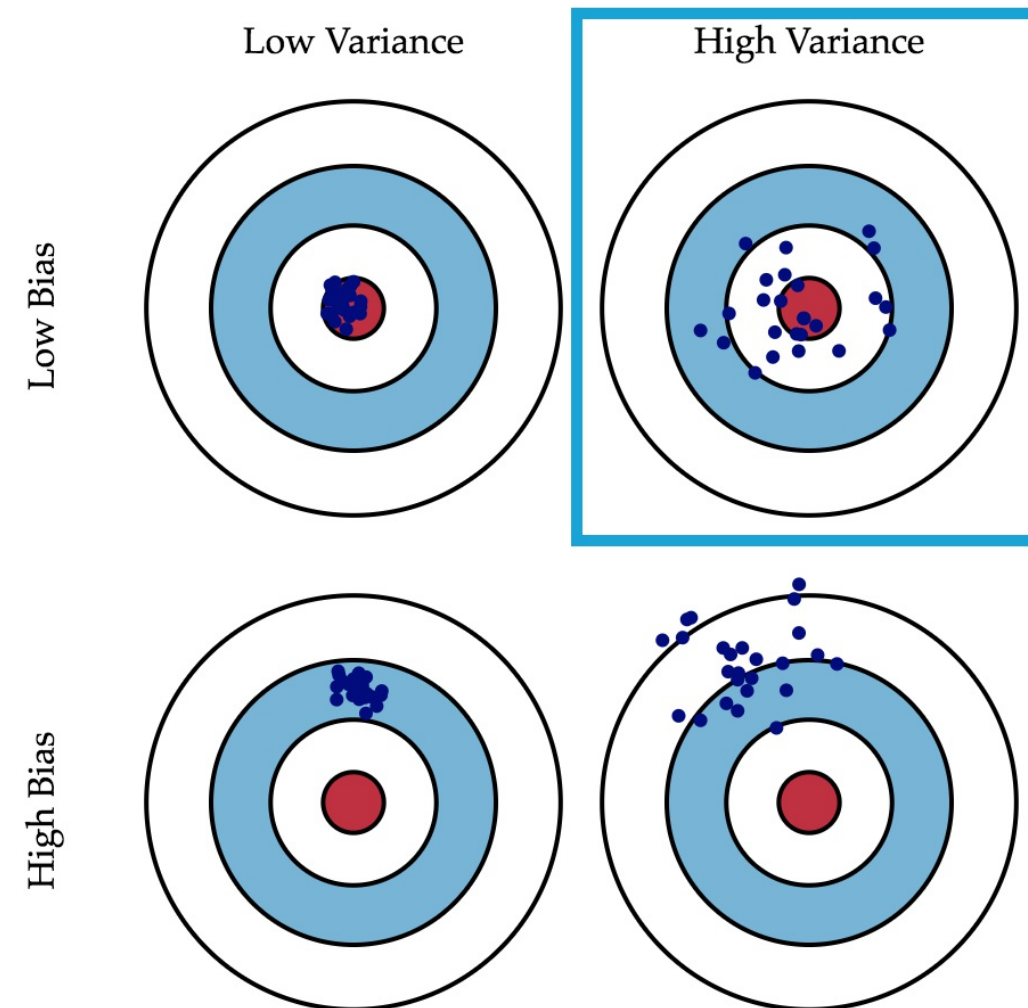
Decision trees and CART

- Constructed iteratively (one decision at a time)
 - Until a stopping criterion is met

Individual decision trees tend to overfit



Individual decision trees tend to overfit





CART: Classification and Regression Trees

- Each leaf **always** contains a real-valued score
- Can later be converted into categories



EXTREME GRADIENT BOOSTING WITH XGBOOST

**Let's work with some
decision trees!**



EXTREME GRADIENT BOOSTING WITH XGBOOST

What is Boosting?

Sergey Fogelson

VP of Analytics, Viacom



Boosting overview

- Not a specific machine learning algorithm
- Concept that can be applied to a set of machine learning models
 - "Meta-algorithm"
- Ensemble meta-algorithm used to convert many weak learners into a strong learner



Weak learners and strong learners

- Weak learner: ML algorithm that is slightly better than chance
 - Example: Decision tree whose predictions are slightly better than 50%
- Boosting converts a collection of weak learners into a strong learner
- Strong learner: Any algorithm that can be tuned to achieve good performance

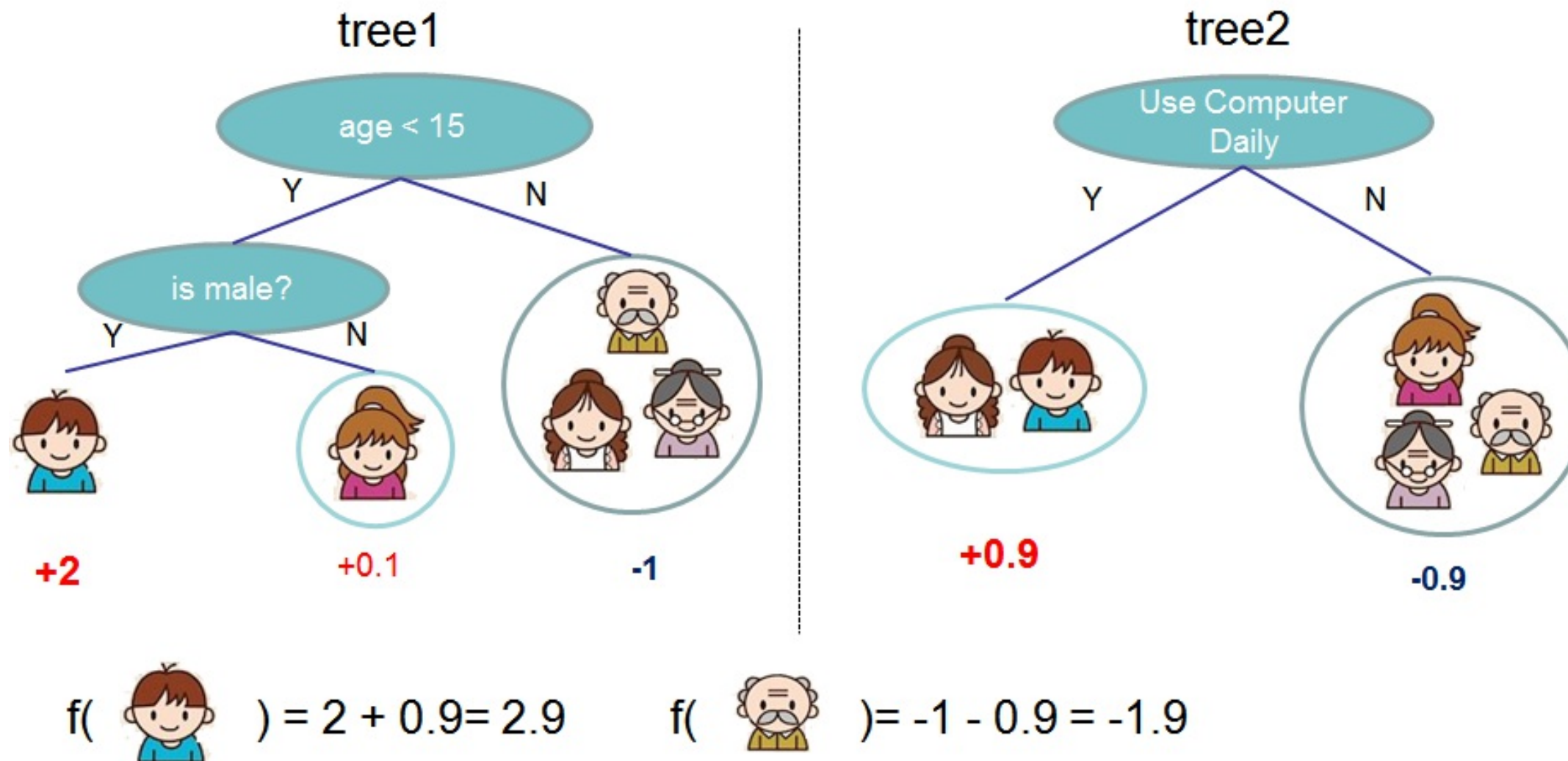


How boosting is accomplished

- Iteratively learning a set of weak models on subsets of the data
- Weighing each weak prediction according to each weak learner's performance
- Combine the weighted predictions to obtain a single weighted prediction
- ... that is much better than the individual predictions themselves!



Boosting example





Model evaluation through cross-validation

- Cross-validation: Robust method for estimating the performance of a model on unseen data
- Generates many non-overlapping train/test splits on training data
- Reports the average test set performance across all data splits

Cross-validation in XGBoost example

```
In [1]: import xgboost as xgb

In [2]: import pandas as pd

In [3]: class_data = pd.read_csv("classification_data.csv")

In [4]: churn_dmatrix = xgb.DMatrix(data=churn_data.iloc[:, :-1],
    label=churn_data.month_5_still_here)

In [5]: params={"objective": "binary:logistic", "max_depth": 4}

In [6]: cv_results = xgb.cv(dtrain=churn_dmatrix, params=params, nfold=4,
    num_boost_round=10, metrics="error", as_pandas=True)

In [7]: print("Accuracy: %f" % ((1 - cv_results["test-error-mean"]).iloc[-1]))
Accuracy: 0.88315
```



EXTREME GRADIENT BOOSTING WITH XGBOOST

Let's practice!



EXTREME GRADIENT BOOSTING WITH XGBOOST

When should I use XGBoost?

Sergey Fogelson

VP of Analytics, Viacom



When to use XGBoost

- You have a large number of training samples
 - Greater than 1000 training samples and less 100 features
 - The number of features $<$ number of training samples
- You have a mixture of categorical and numeric features
 - Or just numeric features



When to NOT use XGBoost

- Image recognition
- Computer vision
- Natural language processing and understanding problems
- When the number of training samples is significantly smaller than the number of features



EXTREME GRADIENT BOOSTING WITH XGBOOST

Let's practice!