January 2025

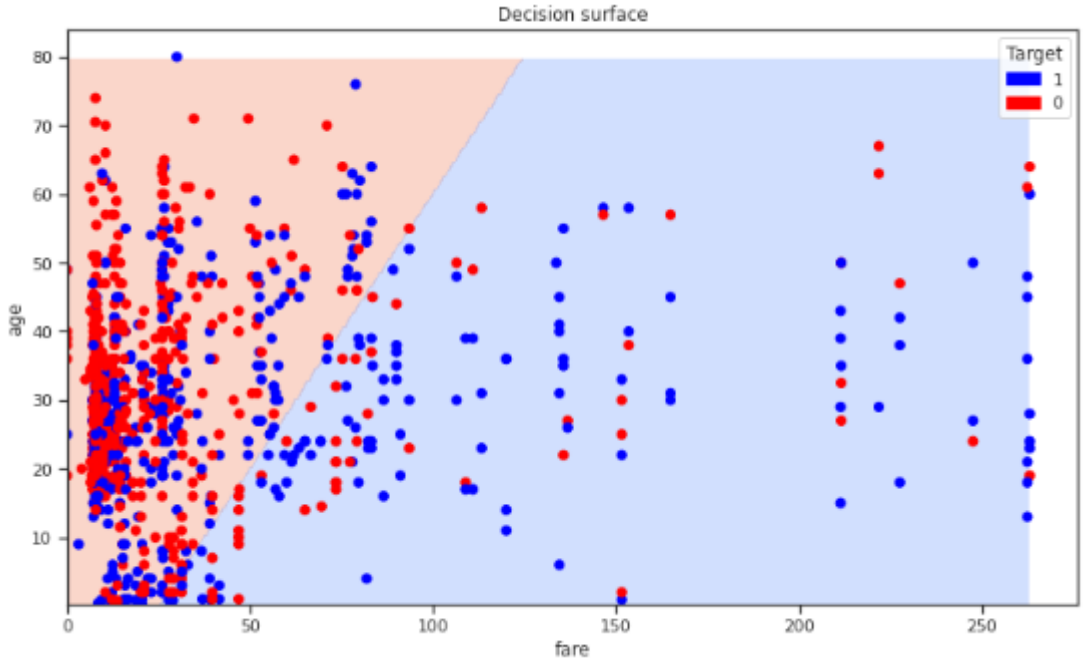
Module 3 – Fitting, Overfitting, Generalization

Data Science For Business

Quiz time!

Quiz discussion!

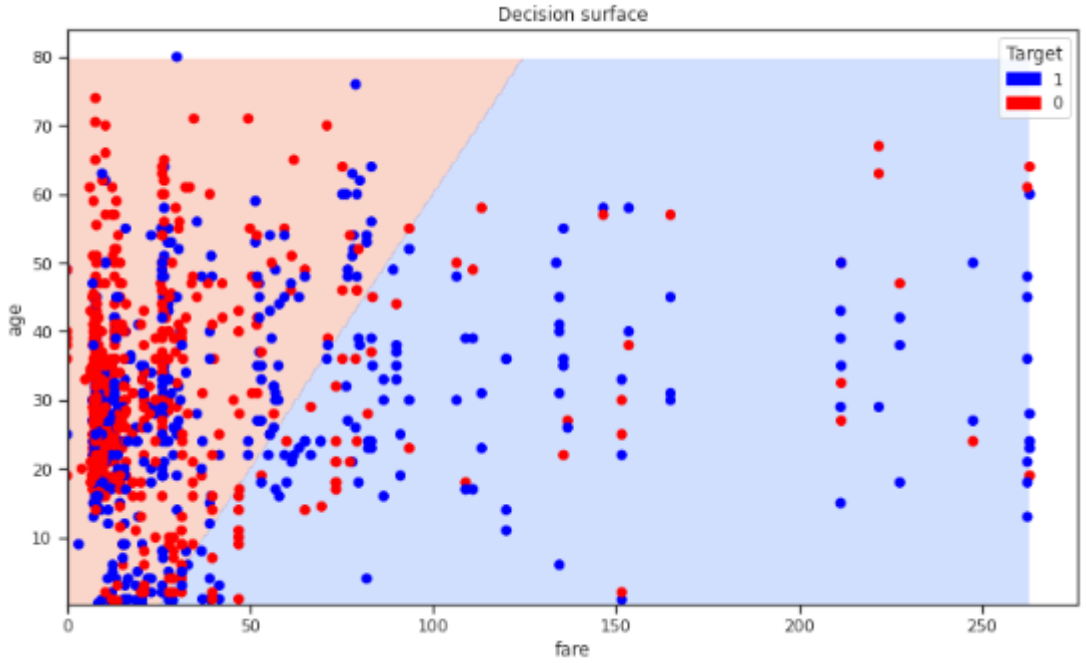
Q1



- 1
- 0

age = 70, fare = 200

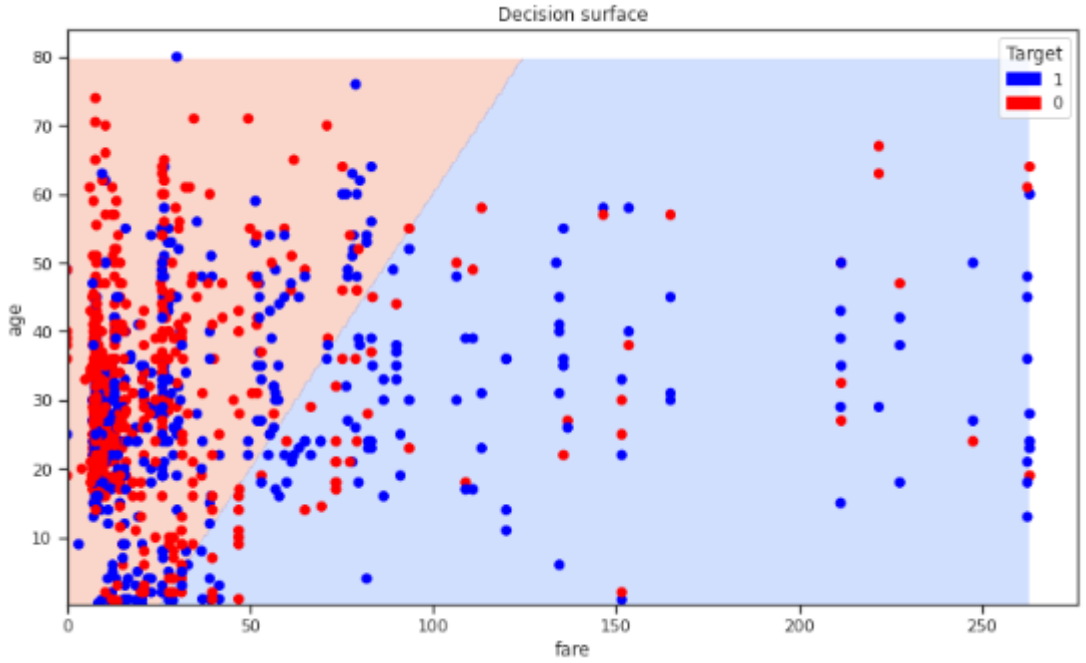
Q1



age = 70, fare = 200

- 1
- 0

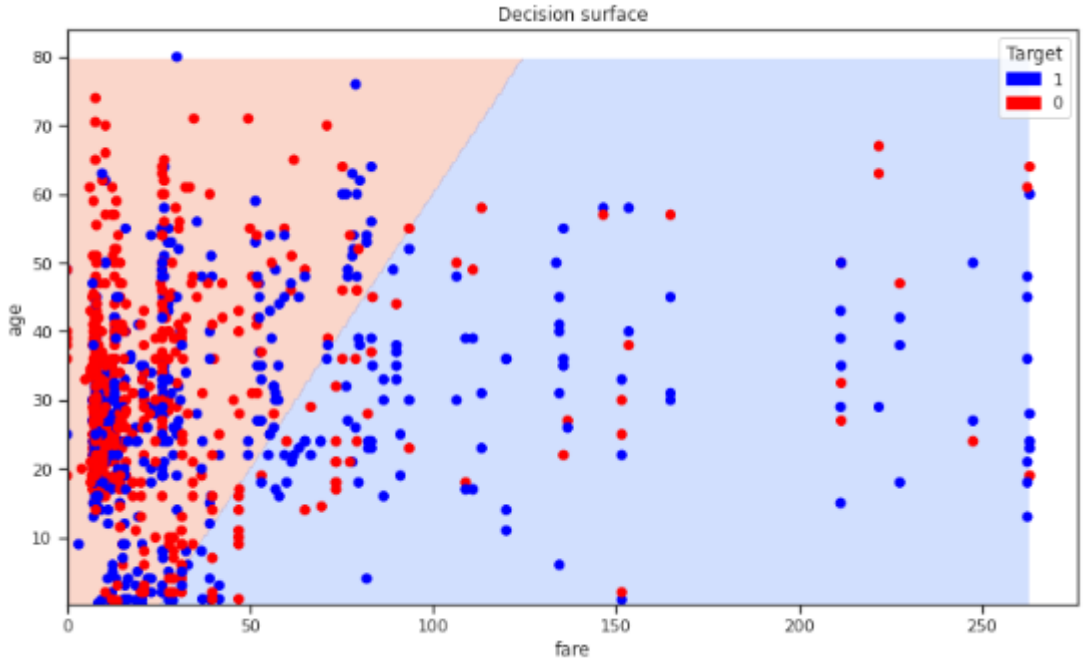
Q2



- 1
- 0

age = 30, fare = 50

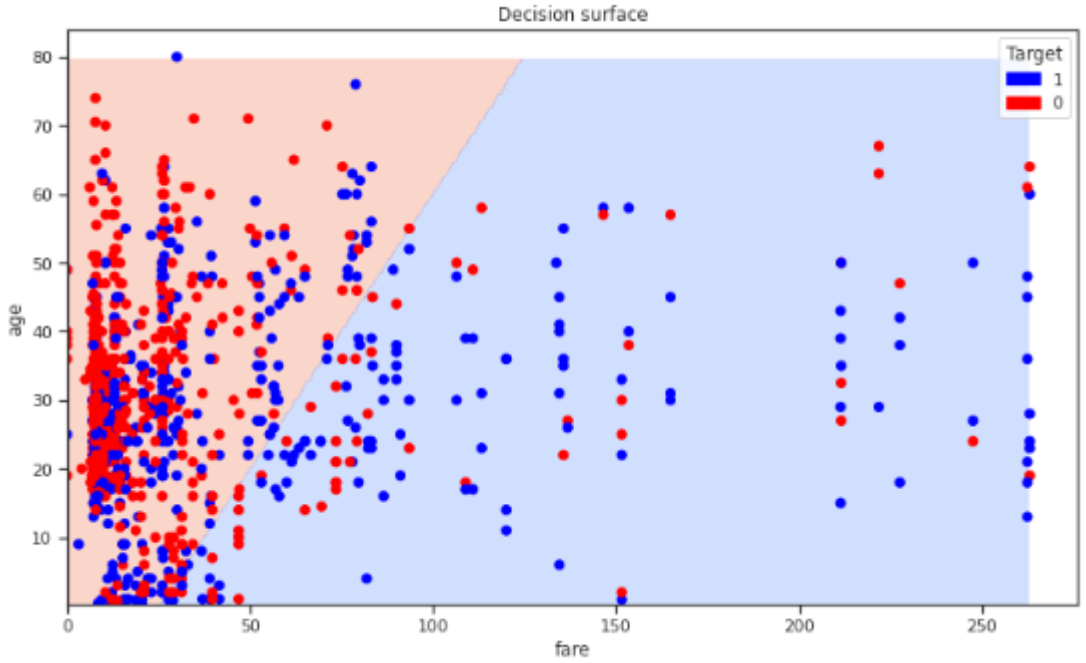
Q2



age = 30, fare = 50

- 1
- 0

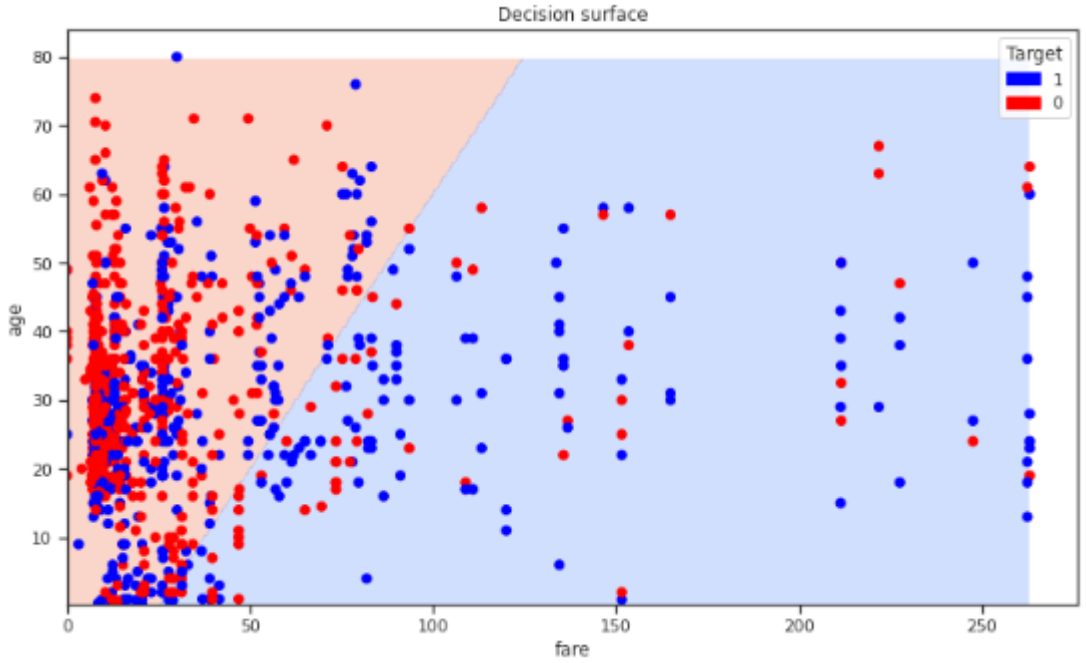
Q3



- 1
- 0

age = 10, fare = 0

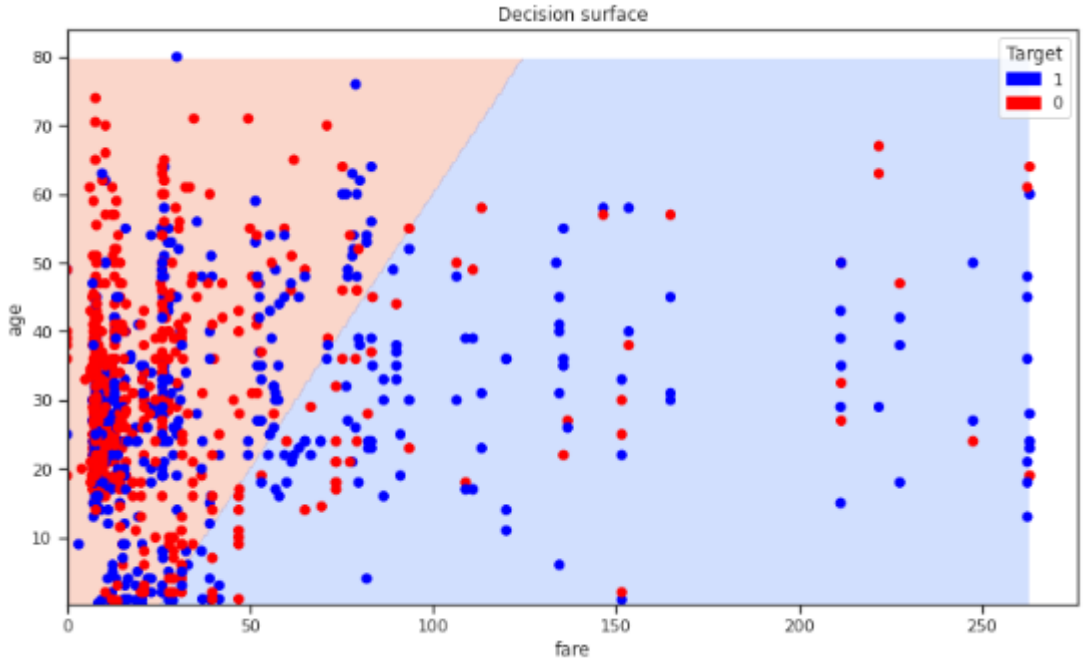
Q3



- 1
- 0

age = 10, fare = 0

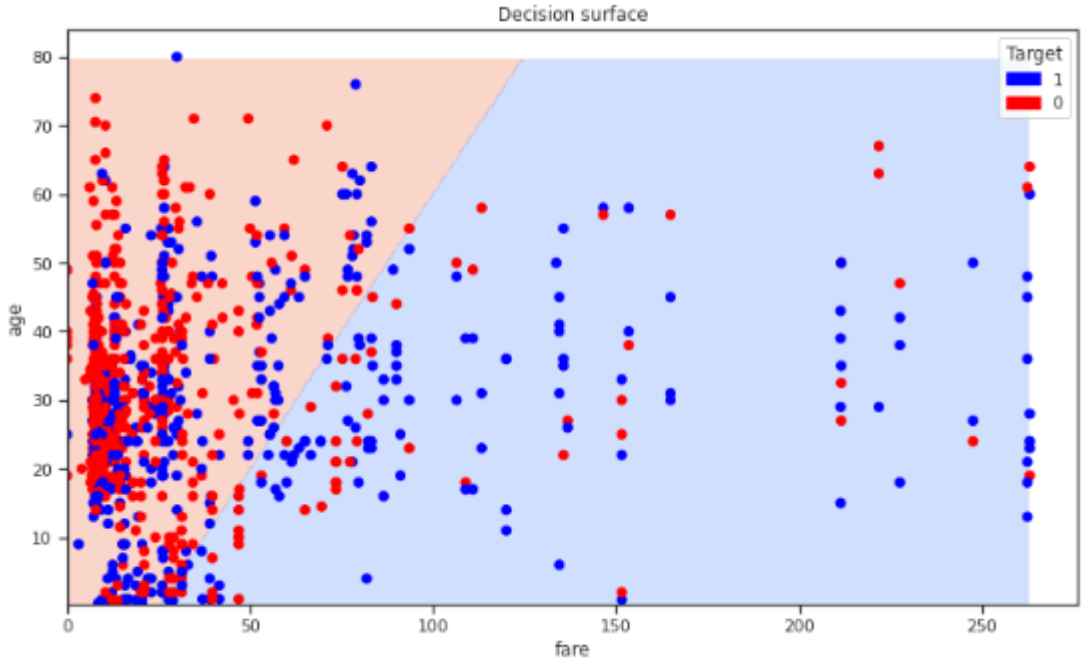
Q4



age = 5, fare = 50

- 1
- 0

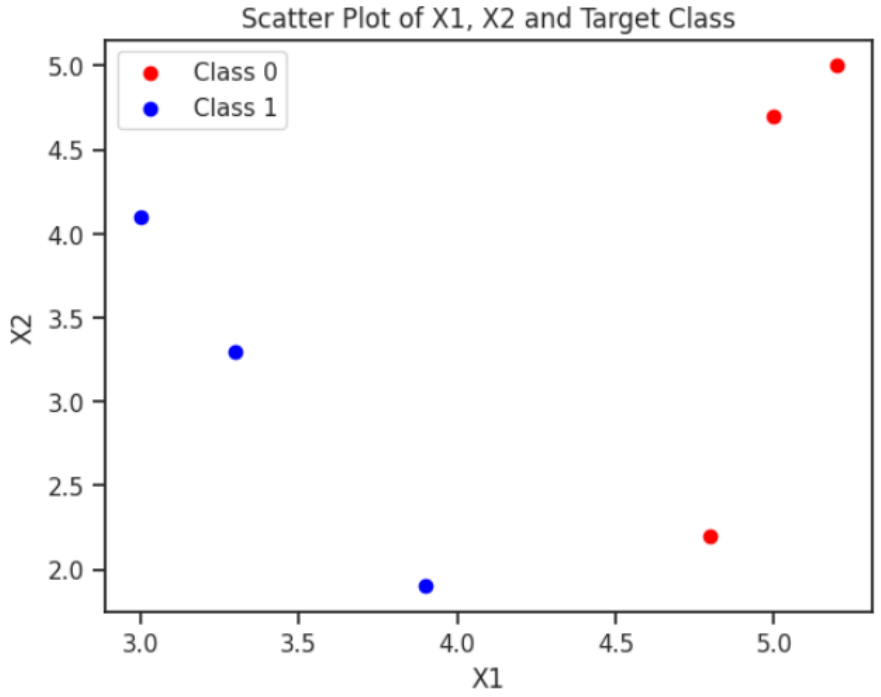
Q4



age = 5, fare = 50

- 1
- 0

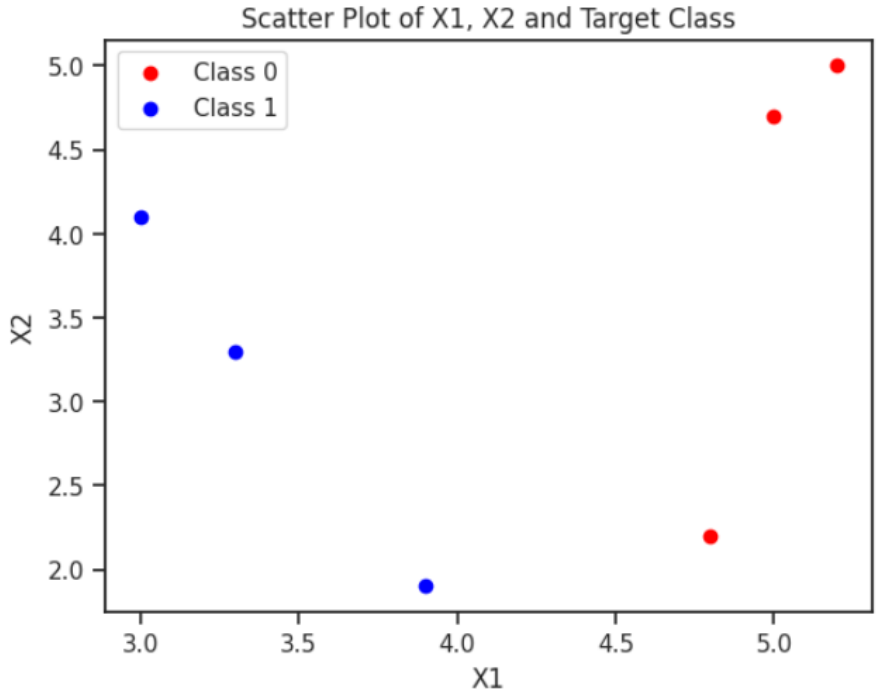
Q5



What is the current entropy without any splits? Please use 1 significant digit.

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

Q5



What is the current entropy without any splits? Please use 1 significant digit.

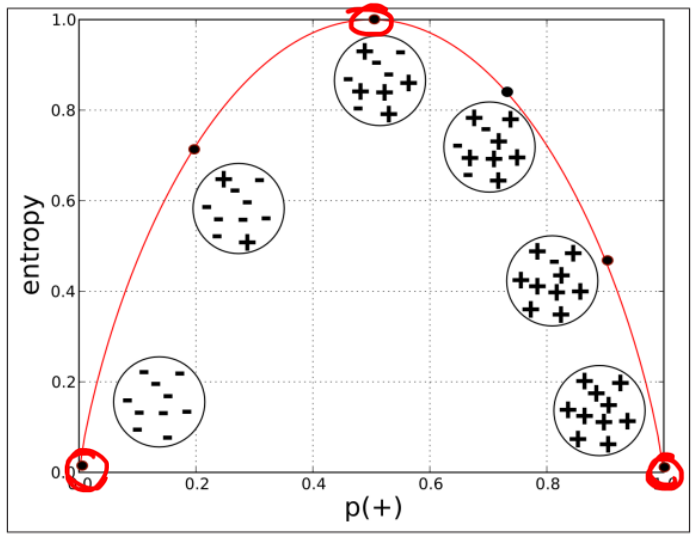
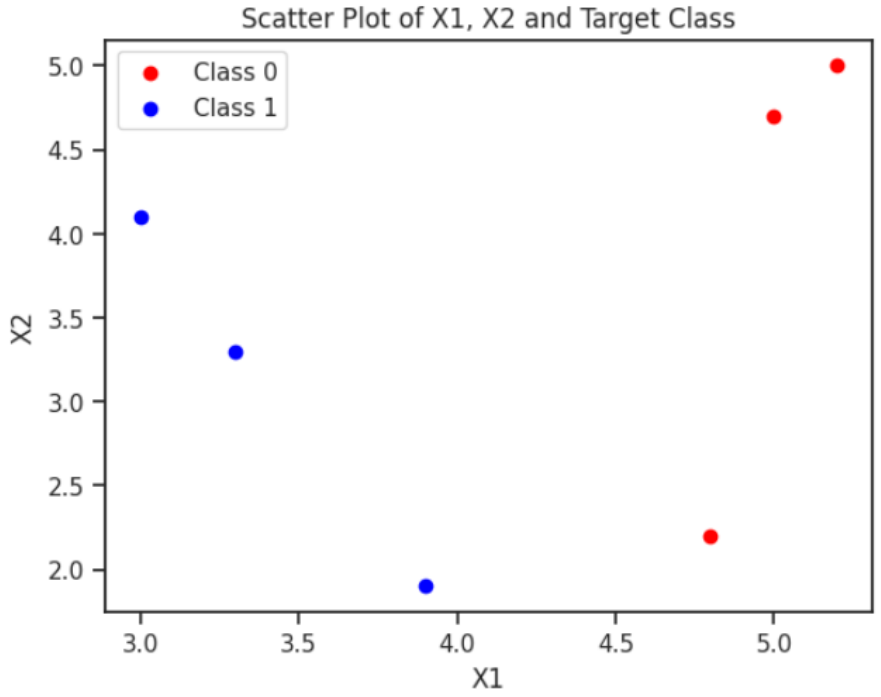


Figure 3-3. Entropy of a two-class set as a function of p(+).

$$\begin{aligned} entropy(S) &= -[0.7 \times \log_2(0.7) + 0.3 \times \log_2(0.3)] \\ &\approx -[0.7 \times -0.51 + 0.3 \times -1.74] \\ &\approx 0.88 \end{aligned}$$

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

Q5

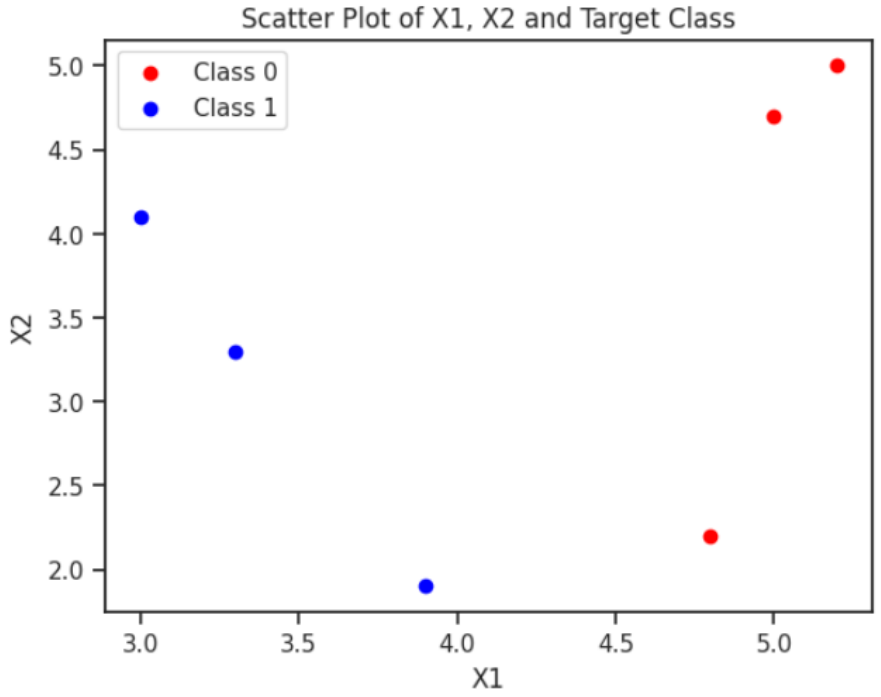


What is the current entropy without any splits? Please use 1 significant digit.

1.0 (totally mixed up)

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

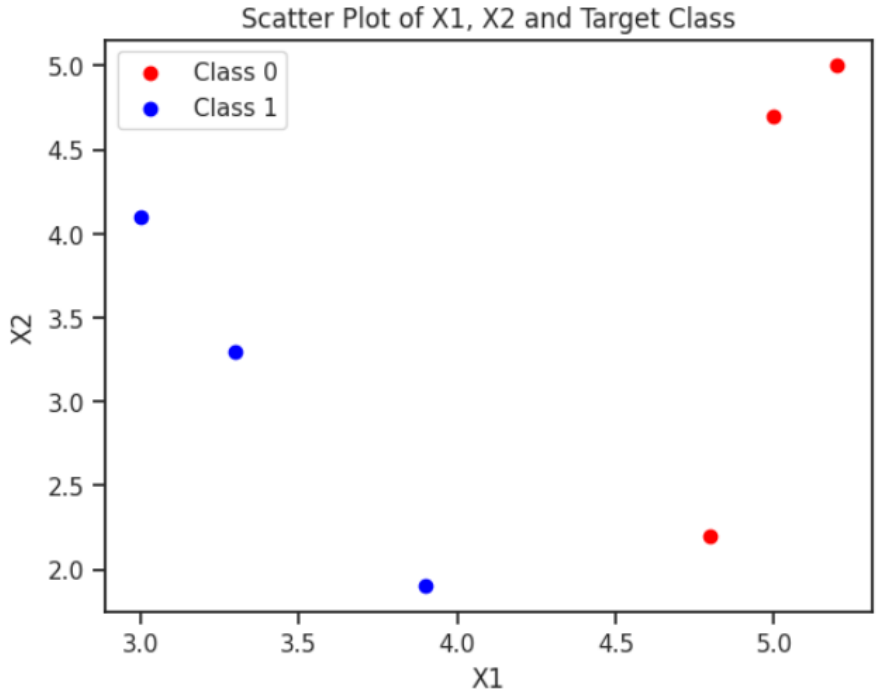
Q6



Which feature (X1 or X2) should we split on to maximize information gain (i.e. create the most predictive split)?

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

Q6

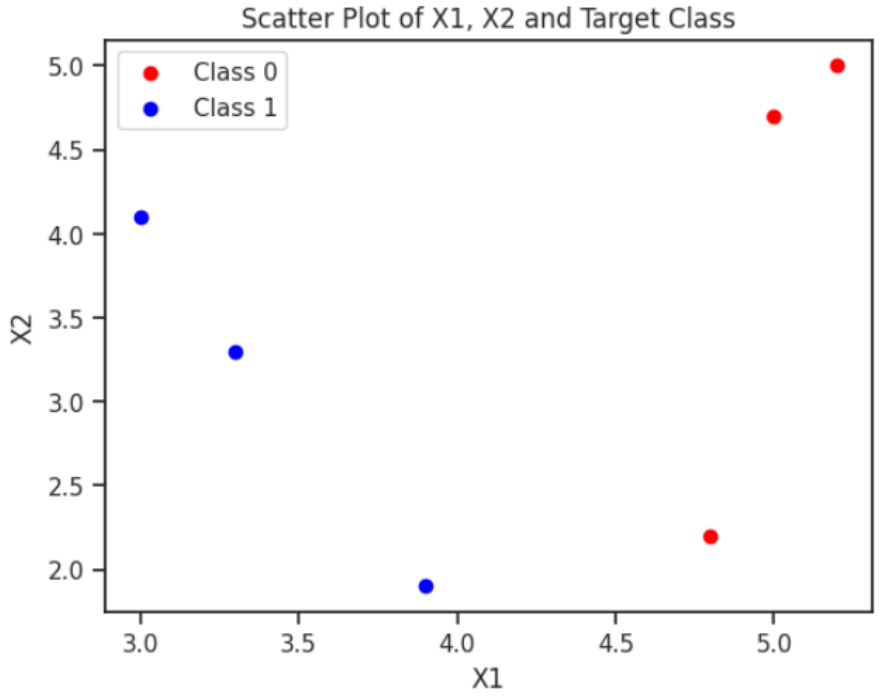


Which feature (X1 or X2) should we split on to maximize information gain (i.e. create the most predictive split)?

X1

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

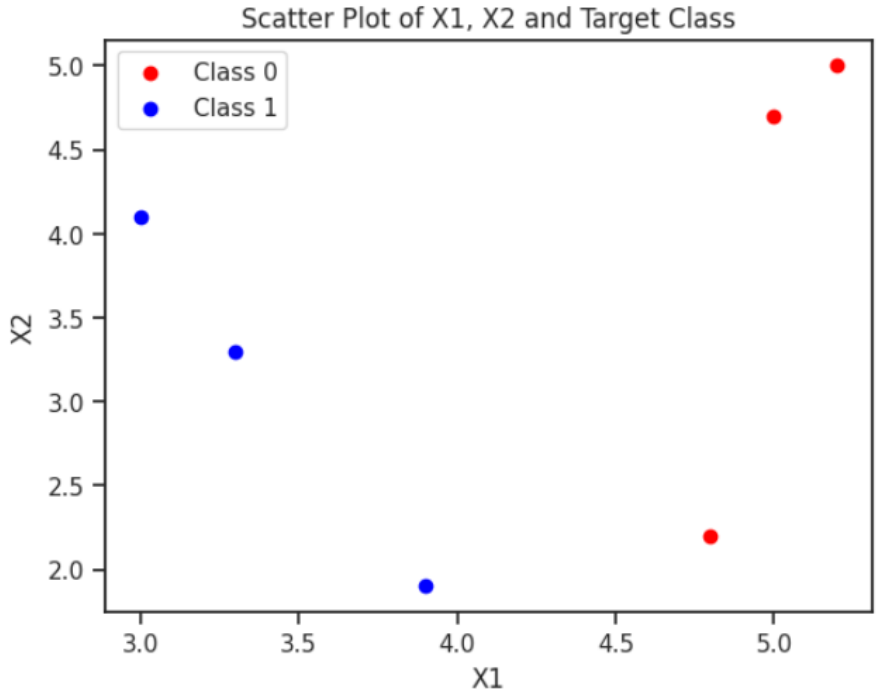
Q7



What threshold should be picked for this feature to maximize information gain?

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

Q7

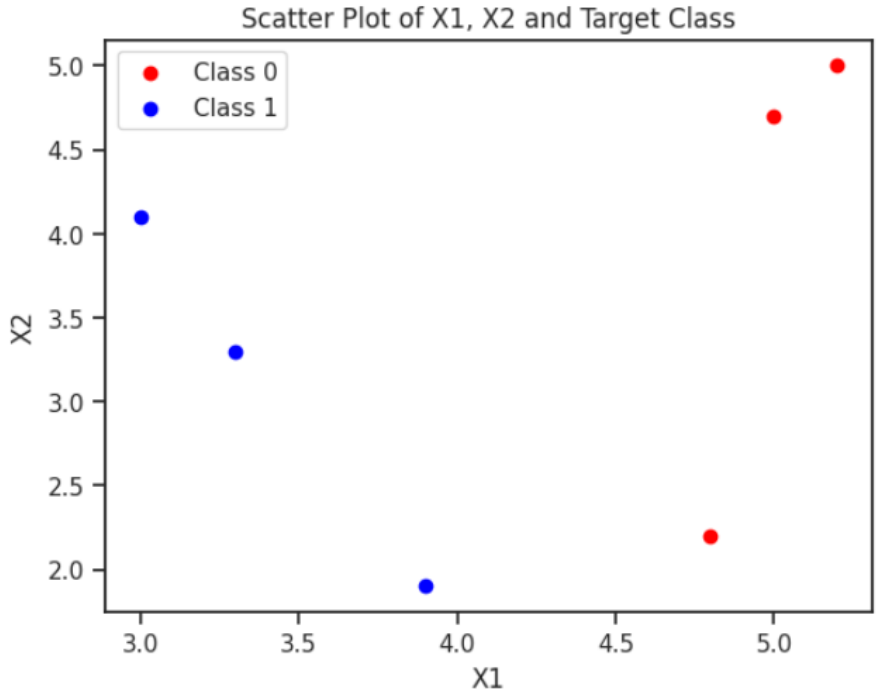


What threshold should be picked for this feature to maximize information gain?

~4

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

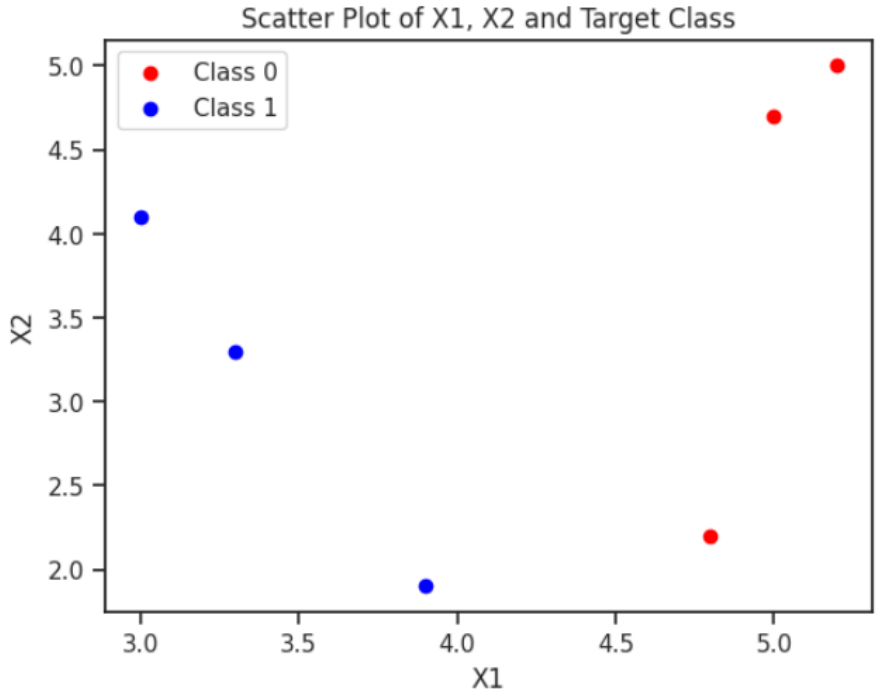
Q8



What is the entropy of each leaf node (each sub-group) after this split?

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

Q8

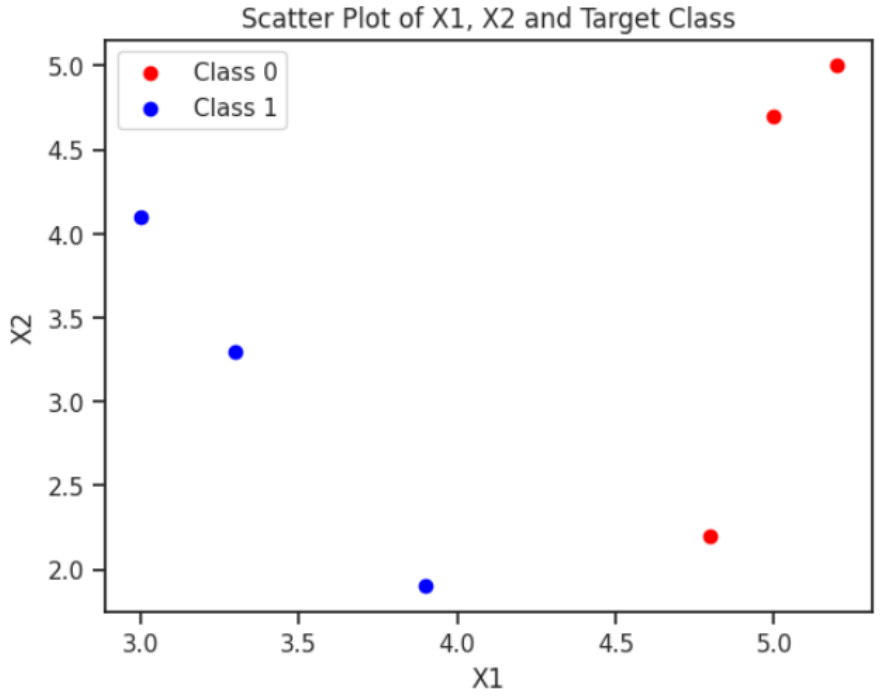


What is the entropy of each leaf node (each sub-group) after this split?

0 for X1<4.0 and 0 for X1 >= 4.0

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

Q9

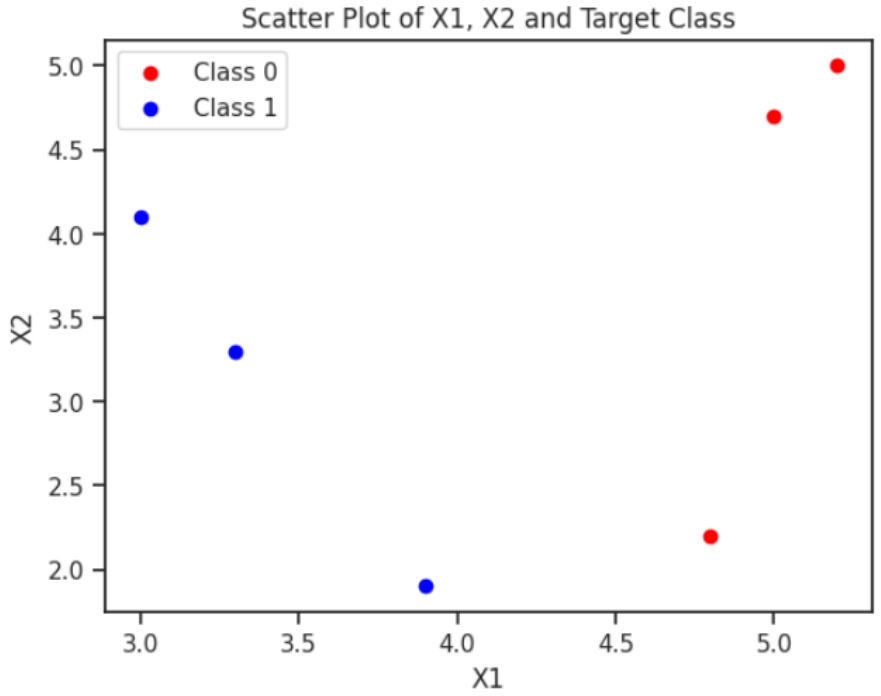


What is the entropy of each leaf node (each sub-group) after this split?

0 for X1<4.0 and 0 for X1 >= 4.0

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

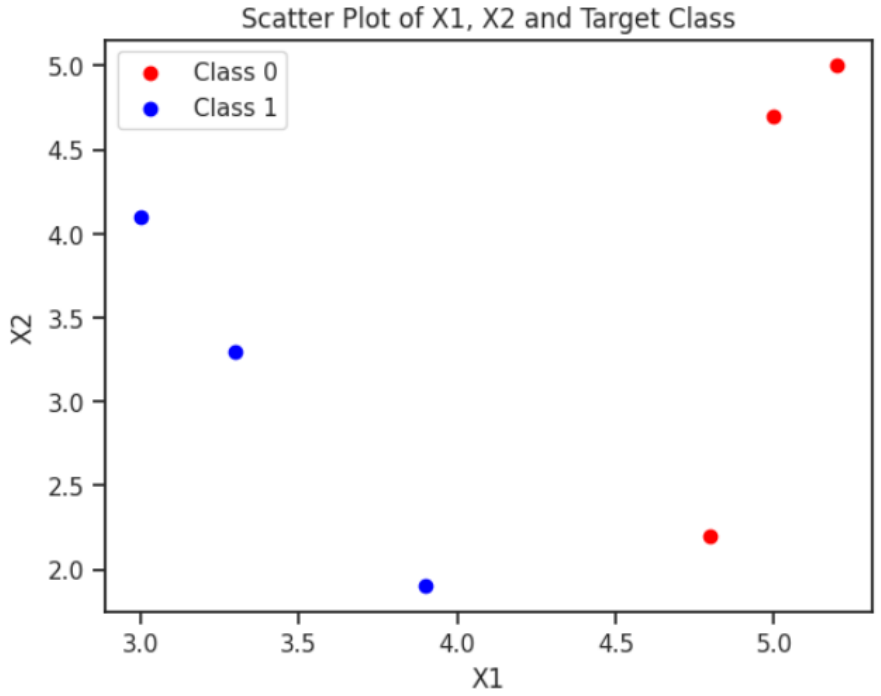
Q9



What is the information gain of this split?

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

Q9



What is the information gain of this split?

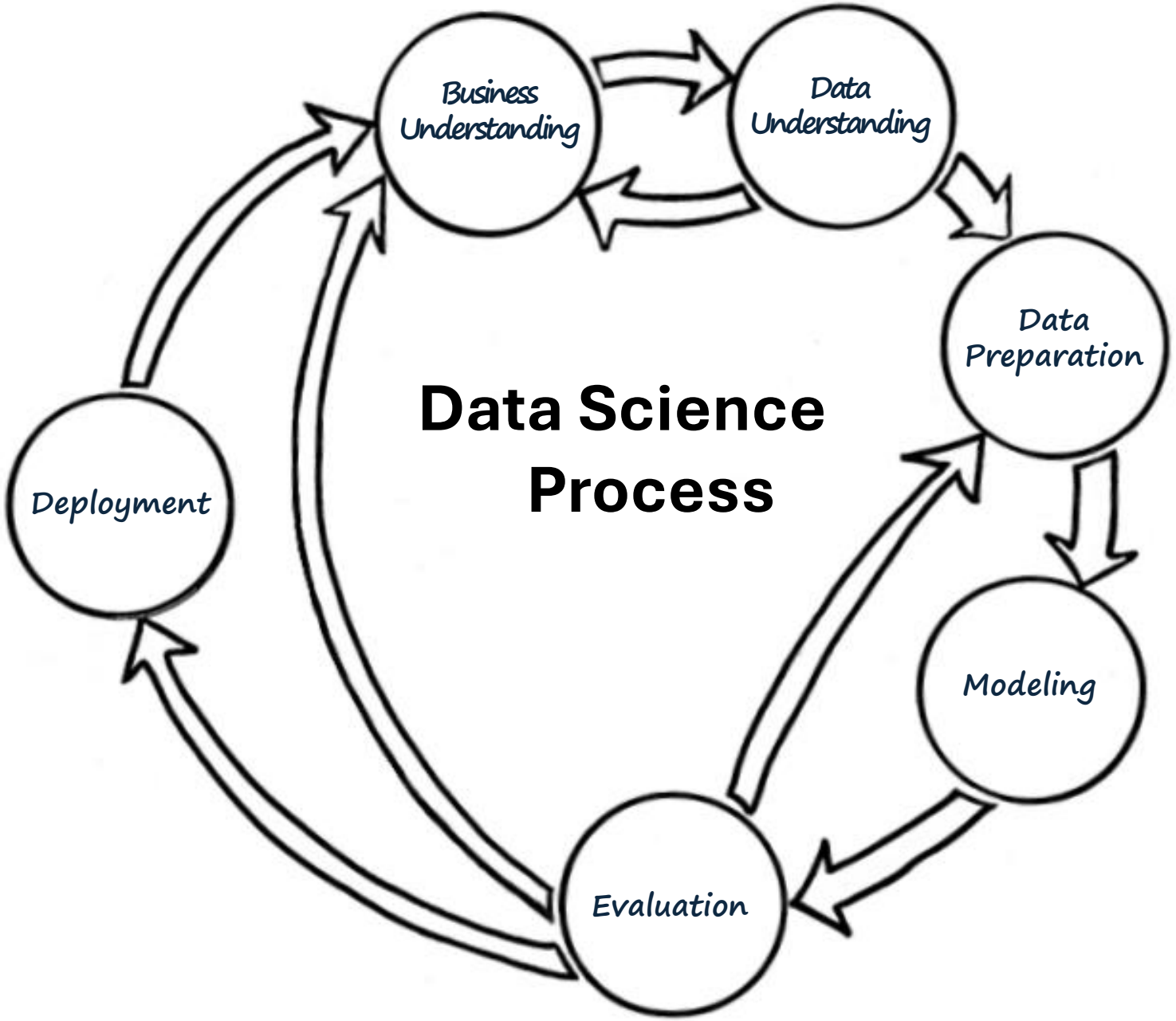
1.0

We are now going to create a **SINGLE SPLIT (depth = 1)** decision tree

Agenda

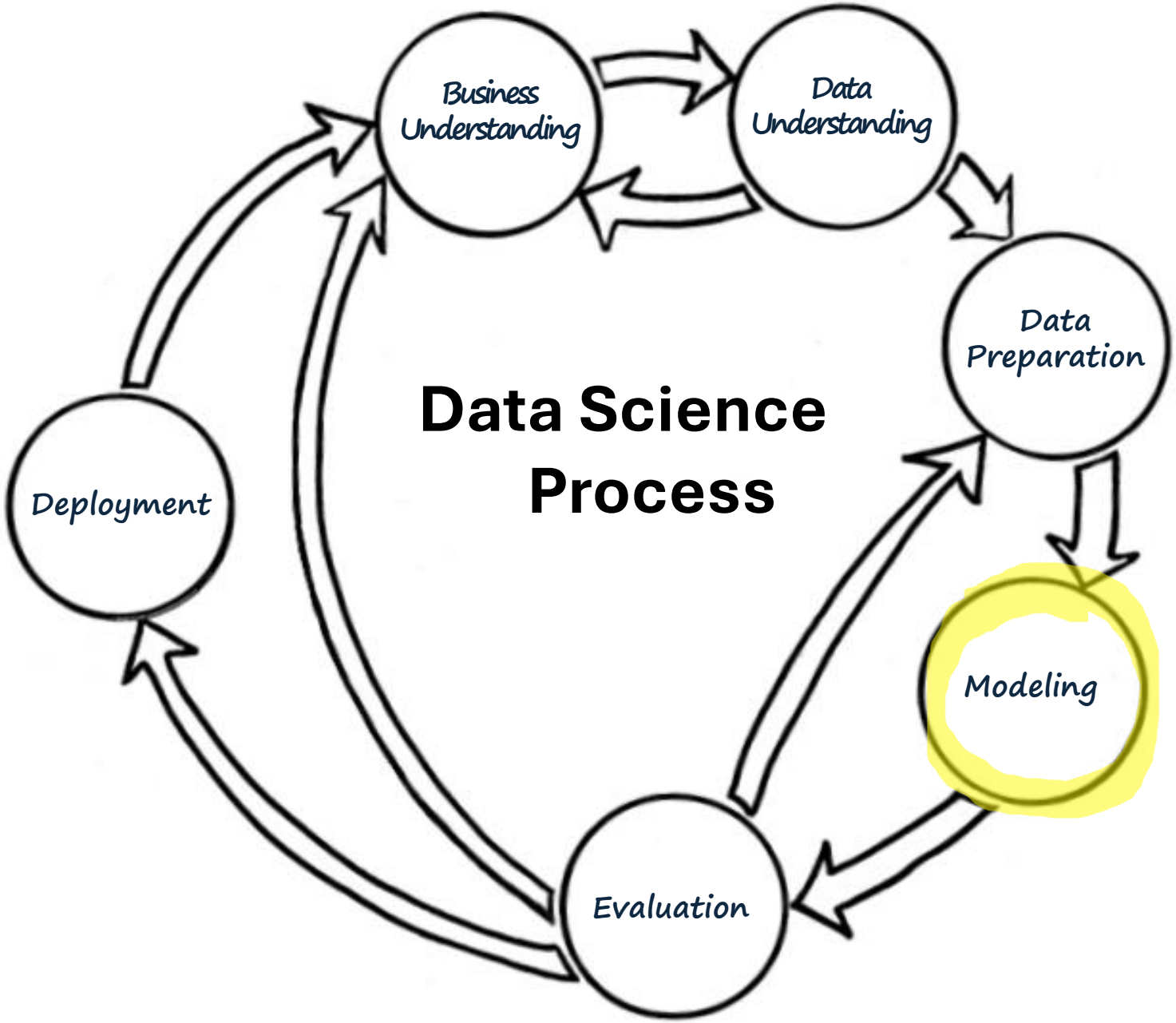
- **Week 1**
  - ~~**Module 1 (Thursday):** Intro to data science + Python for DS~~
  - ~~**Module 2 (Friday):** Intro to supervised learning~~
- **Week 2**
  - **Module 3 (Monday):** Fitting models, generalization
  - **Module 4 (Tuesday):** Regularization
  - **Module 5 (Wednesday):** Evaluation (ROC, cost visualization)
  - **Module 6 (Thursday):** Modeling text data
- **Week 3**
  - **Module 7 (Monday):** Neural networks, GenAI
  - **Module 8 (Tuesday):** Guest lecture(s)
  - **Module 9 (Wednesday):** Causal inference, AB testing, wrap up
  - **Final Exam (Thursday)**

Where we are


Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment
Data Science Process

Where we are

Last class



Where we are

Last class

Goal: **Build a predictive model** to predict whether a given person will survive the titanic

Where we are

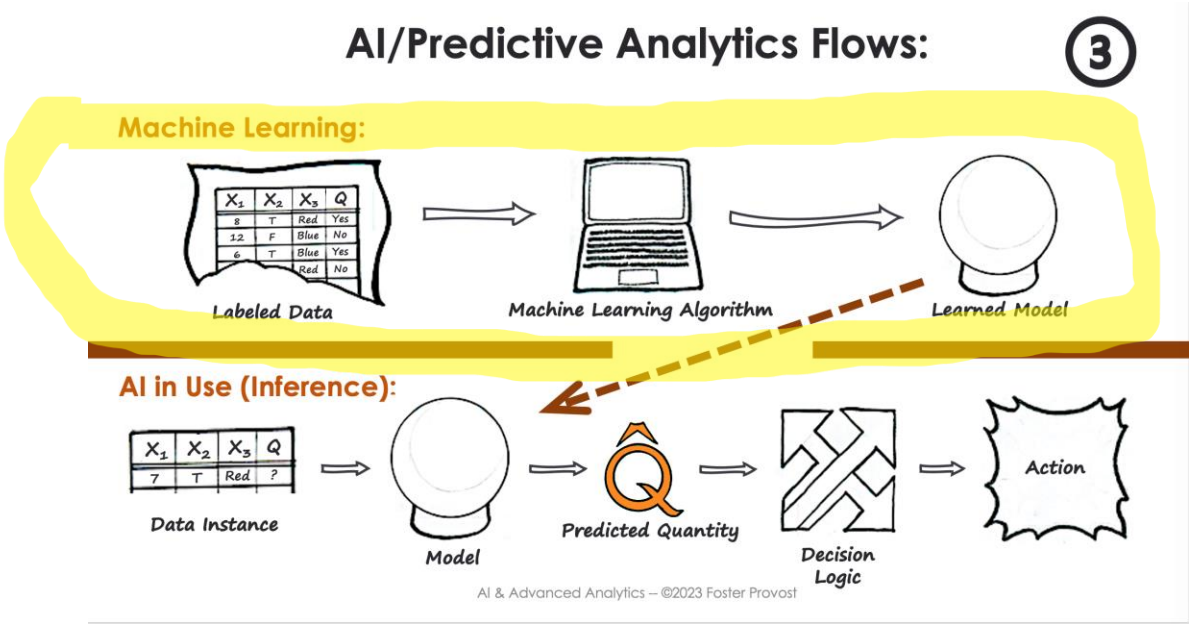
Last class

Goal: **Build a predictive model** to predict whether a given person will survive the titanic

Solution: **Decision Trees**

Where we are

Last class



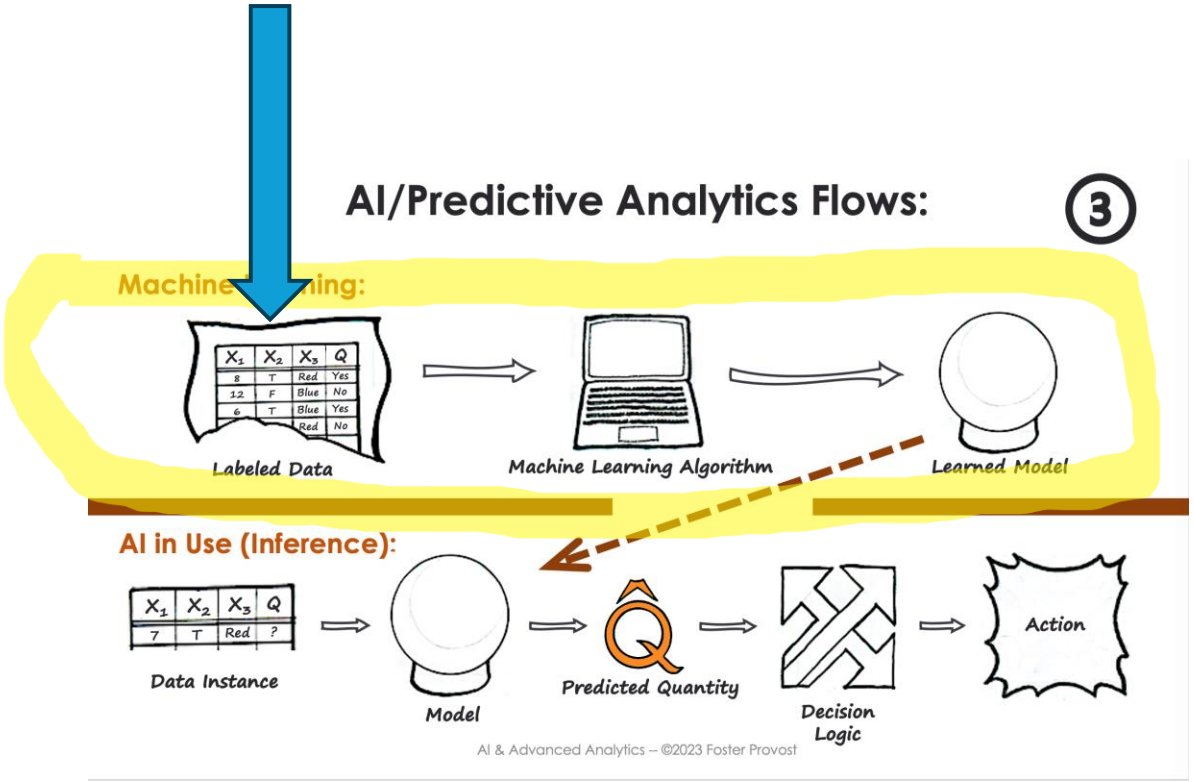
Where we are

Last class

Titanic:

Features (inputs) = attributes about passengers (class, fare, sex, etc.)

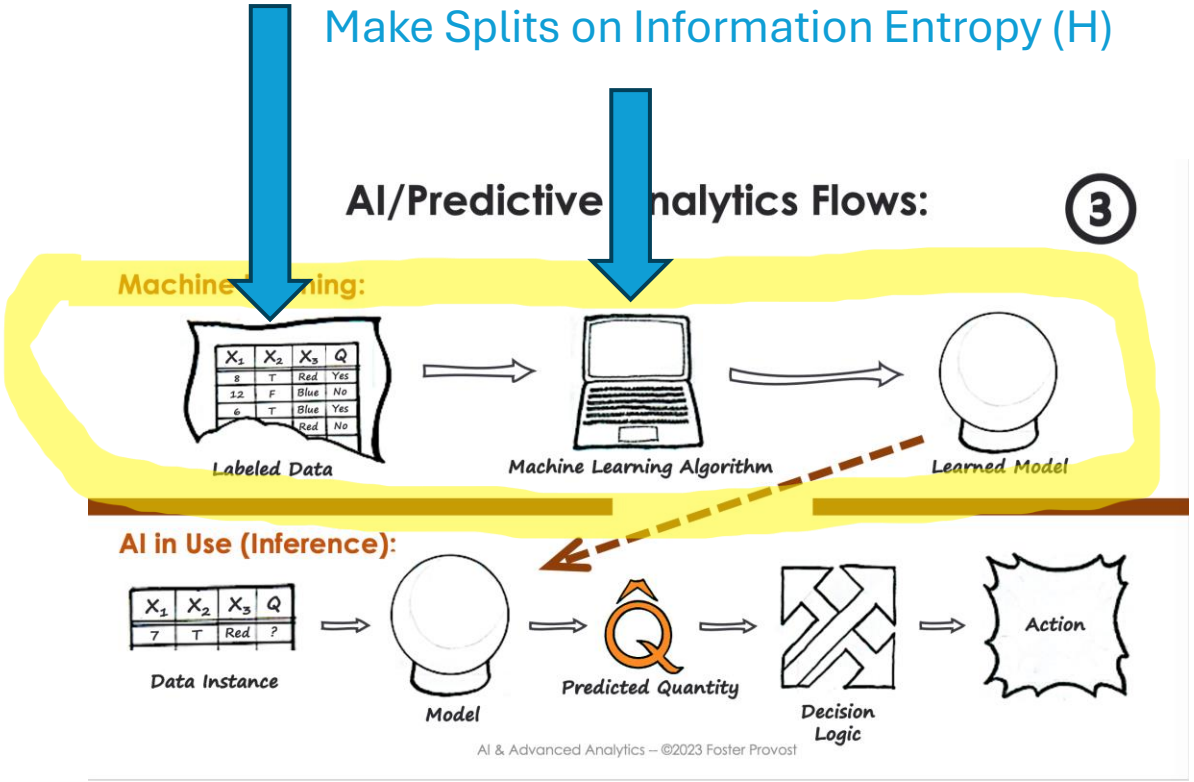
Target (quantity to predict) = Survive (yes = 1/ no = 0)



Where we are
Last class

Titanic Survival Data

Make Splits on Information Entropy (H)



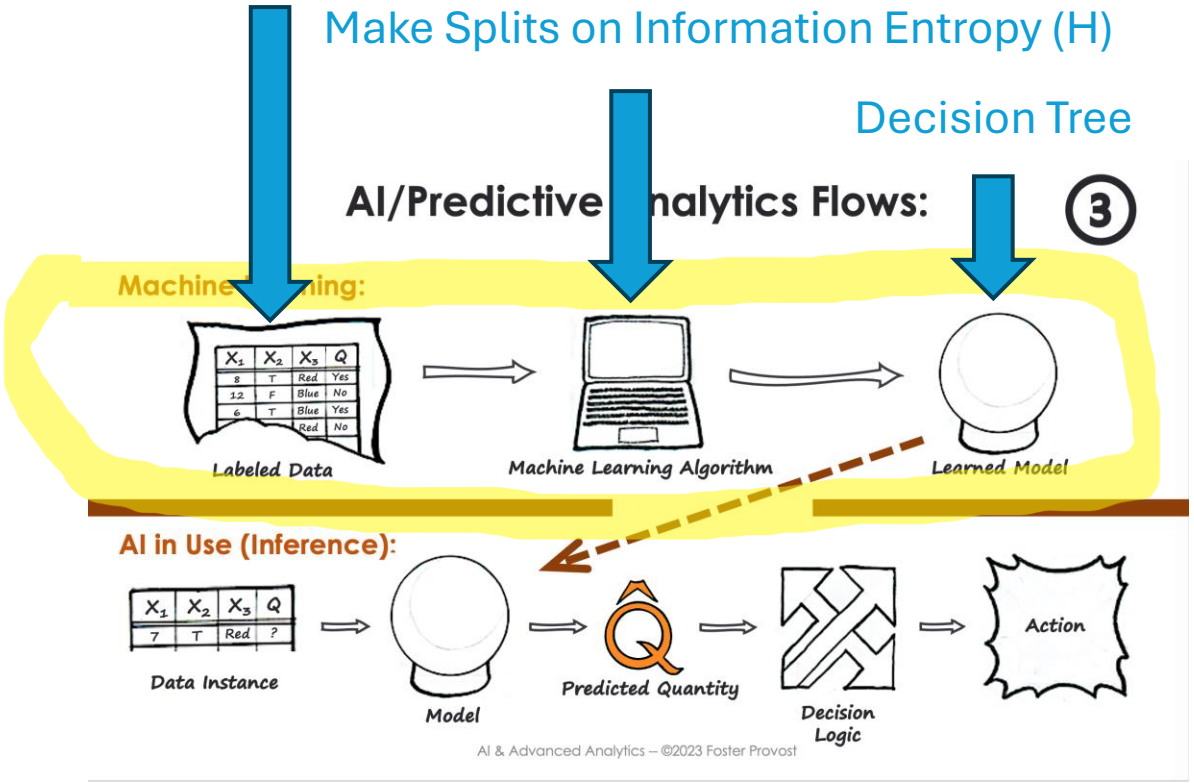
Where we are

Last class

Titanic Survival Data

Make Splits on Information Entropy (H)

Decision Tree



Where we are

Last class

In doing this, we find the **subsets of feature values (e.g. female = 1 & class = 1.0)** that correlate with high probability of survival (or not survival)

Where we are

Last class

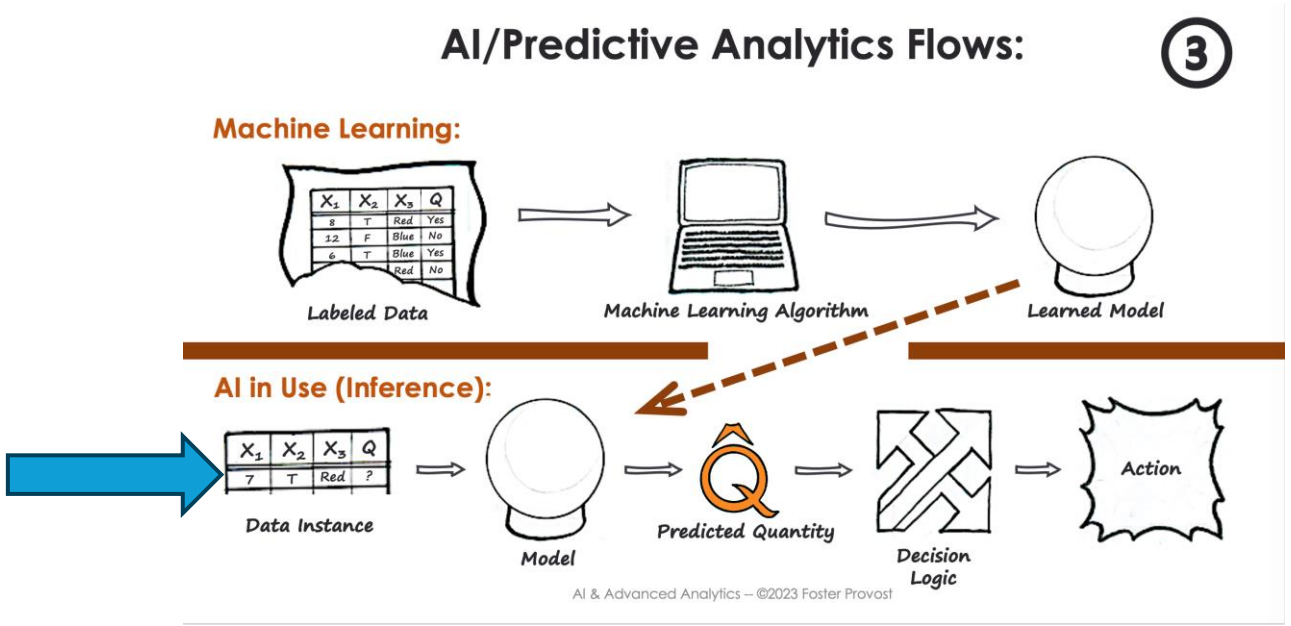
We can then use this **learned model** to predict whether someone will survive **if we don’t have this information**

Where we are

Last class

We can then use this **learned model** to predict whether someone will survive **if we don't have this information**

New person (instance) with **feature values** but **no target value**

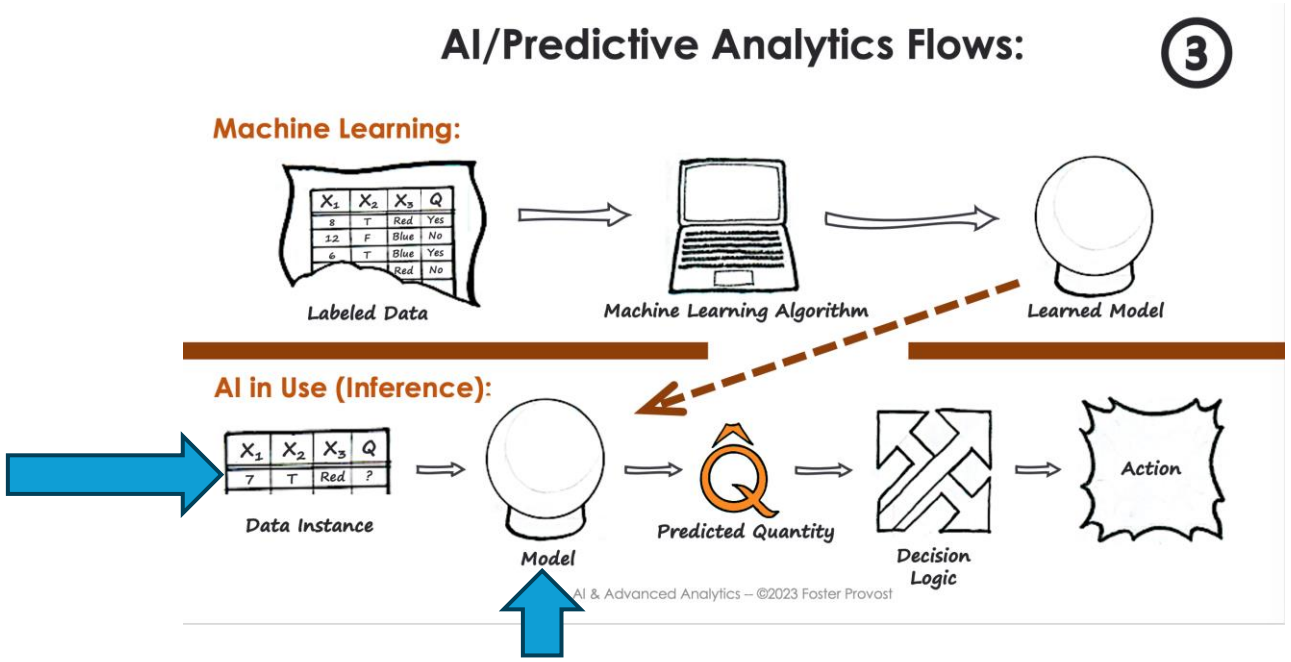


Where we are

Last class

We can then use this **learned model** to predict whether someone will survive **if we don't have this information**

New person (instance) with **feature values** but **no target value**



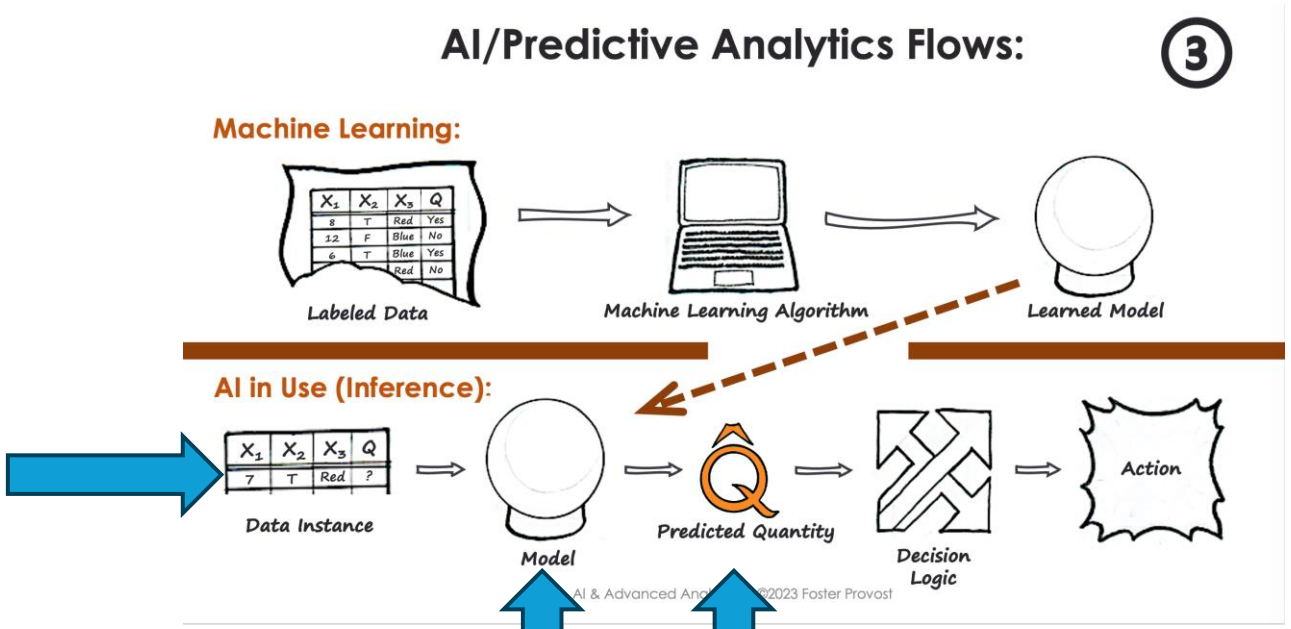
Decision Tree

Where we are

Last class

We can then use this **learned model** to predict whether someone will survive **if we don't have this information**

New person (instance) with **feature values** but **no target value**



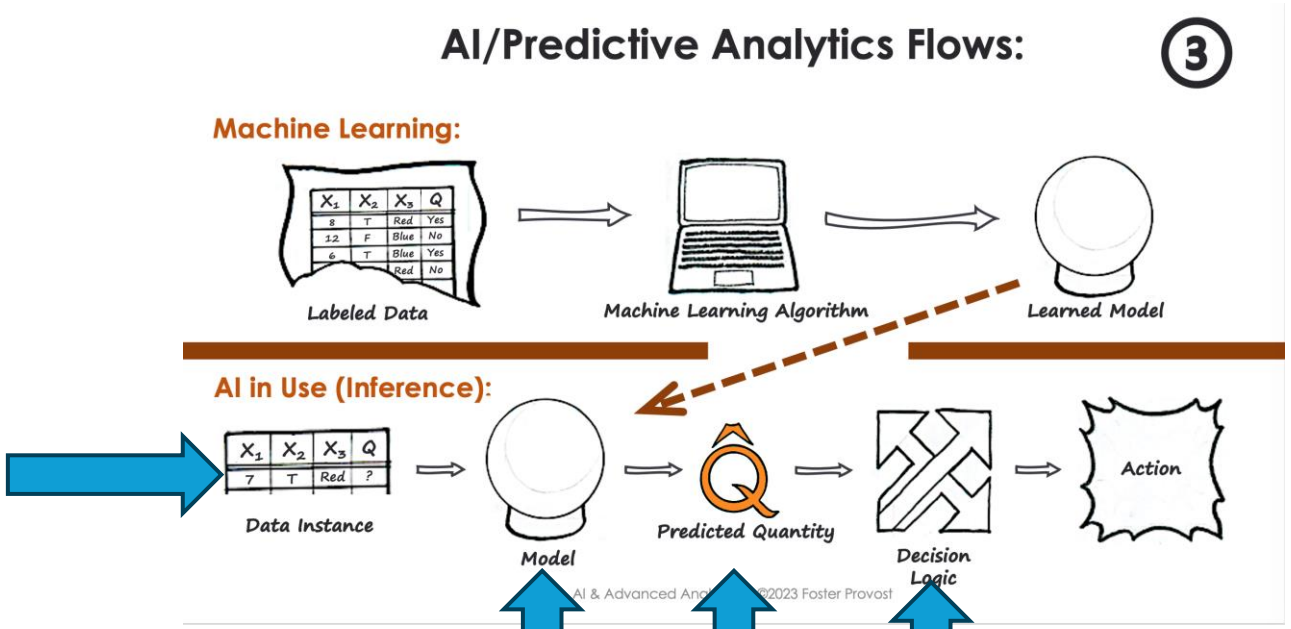
Decision Tree

Survive = {0,1} or P(Leaf of instance that survived)

Where we are

Last class

We can then use this **learned model** to predict whether someone will survive **if we don't have this information**



New person (instance) with **feature values** but **no target value**

Decision Tree

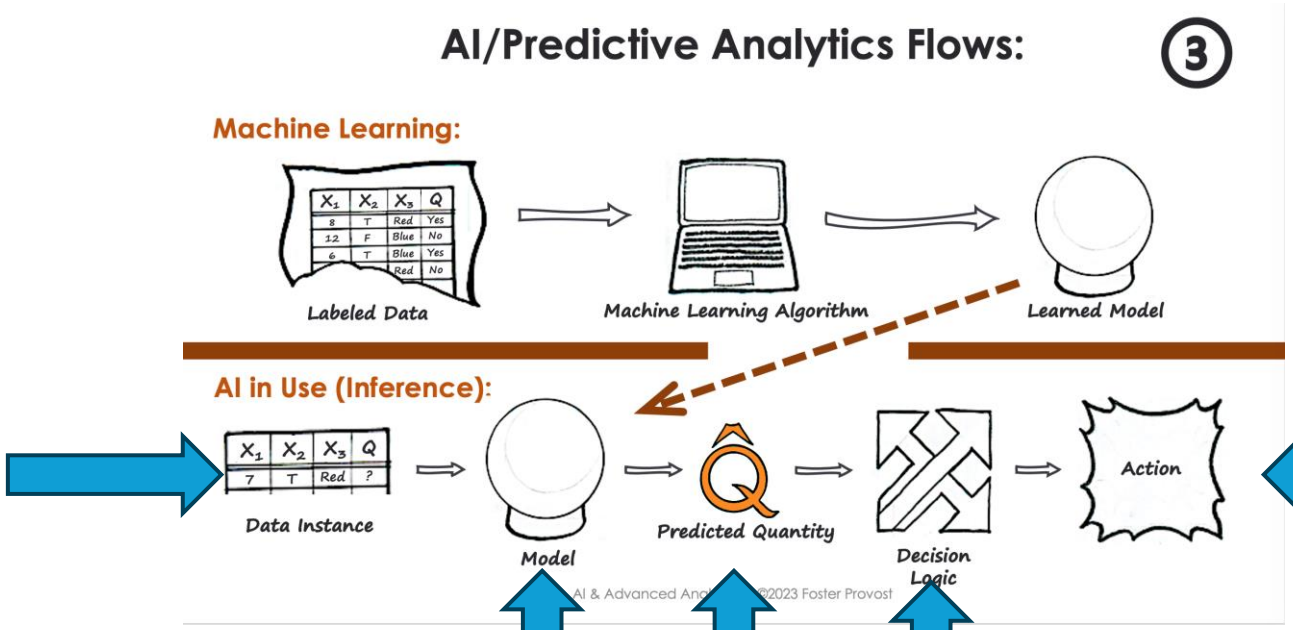
If P(survive)<.5 → Predict WILL NOT survive

Survive = {0,1} or P(Leaf of instance that survived)

Where we are

Last class

We can then use this **learned model** to predict whether someone will survive **if we don't have this information**



New person (instance) with **feature values** but **no target value**

Add extra life vests to rooms of people not predicted to survive?

Decision Tree

If P(survive)<.5 → Predict WILL NOT survive

Survive = {0,1} or P(Leaf of instance that survived)

Where we are

Last class

Goal: **Build a predictive model** to predict whether a given person will survive the titanic

Where we are

Last class

Goal: **Build a predictive model** to predict whether a given person will survive the titanic

Solution: **Decision Trees**

Where we are

Last class

Goal: **Build a predictive model** to predict whether a given person will survive the titanic **(Classification)**

Solution: **Decision Trees**

Where we are

Last class

Types of Tasks and Models

Where we are

Last class

Types of Tasks and Models

Supervised Learning

Classification / Probability Estimation

- Decision tree

(?)

Where we are

Last class

Types of Tasks and Models

We have some **target value** in training data **but not in test data**

→ Supervised Learning

Classification / Probability Estimation

- Decision tree

(?)

Where we are

Last class

Types of Tasks and Models

We want to predict some **class** for an instance (e.g. survive/ churn/ purchase)

→

Supervised Learning

Classification / Probability Estimation

- Decision tree

(?)

Where we are

Last class

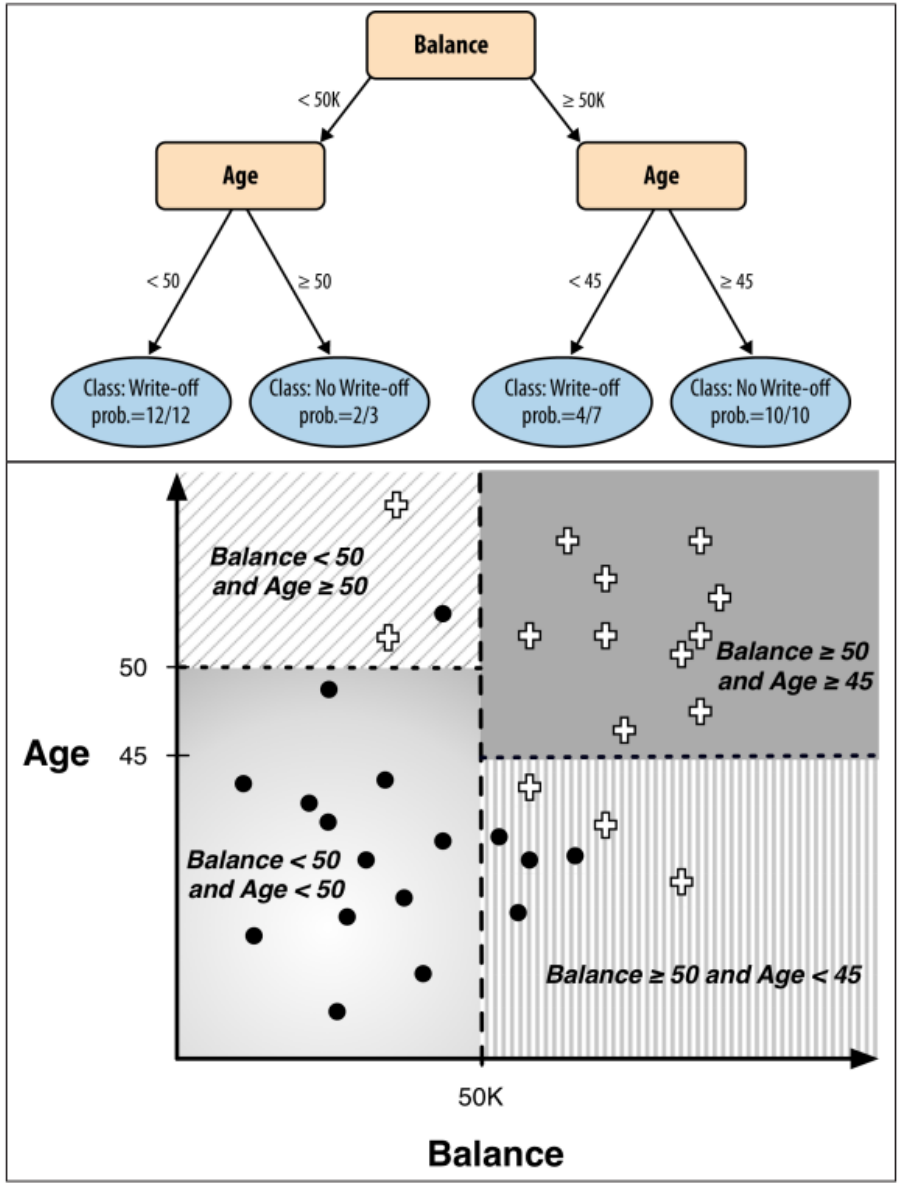
Types o

Supervised Learning

Classification / Probability Estimation

- Decision tree

Split on class entropy to **maximize information gain**/ minimize uncertainty



Where we are

Last class

Types of Tasks and Models

Supervised Learning

Classification / Probabili
Estimation

- Decision tree
- Linear separator (briefly)

Find the line that minimizes some **loss function** (e.g. sum of misclassified points)

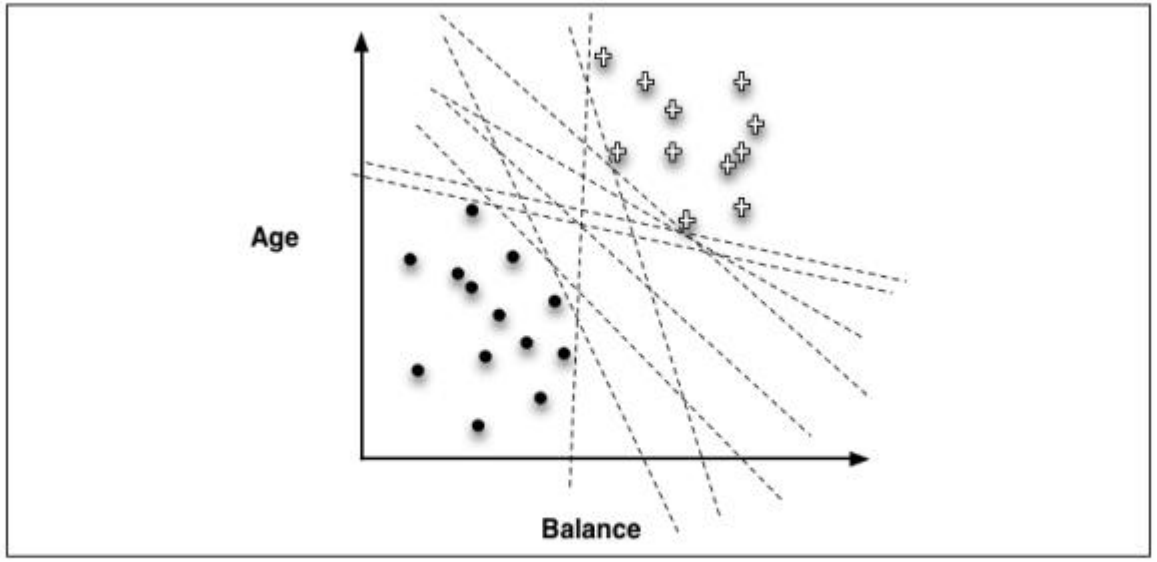


Figure 4-5. Many different possible linear boundaries can separate the two groups of points of Figure 4-4.

Where we are

Today

Types of Tasks and Models

Supervised Learning

Classification / Probability Estimation

- Decision tree
- Linear separator (briefly)

Regression

- Linear/polynomial regression

Find the line/polynomial the minimizes some **loss function** →

(?)

Where we are

Today

Types of Tasks and Models

Supervised Learning

(?)

Classification / Probability Estimation

- Decision tree
- Linear separator (briefly)

Regression

- Linear/polynomial regression
- Regression tree

Assigns predicted value to all items in a leaf node

Where we are

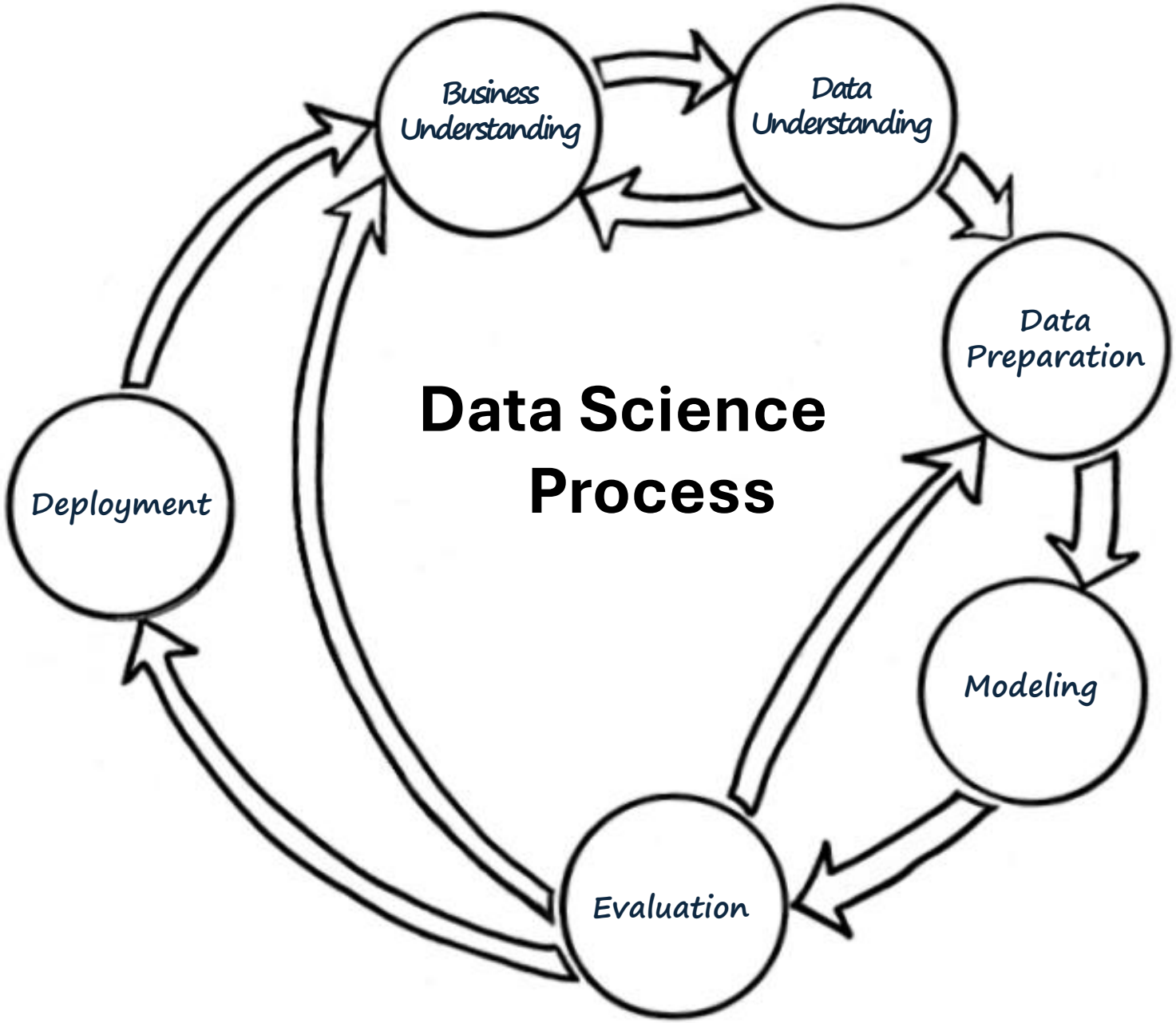
Today

Specifically, we're going to use regression (mostly) to illustrate the ideas of **fitting and generalization**

Regression

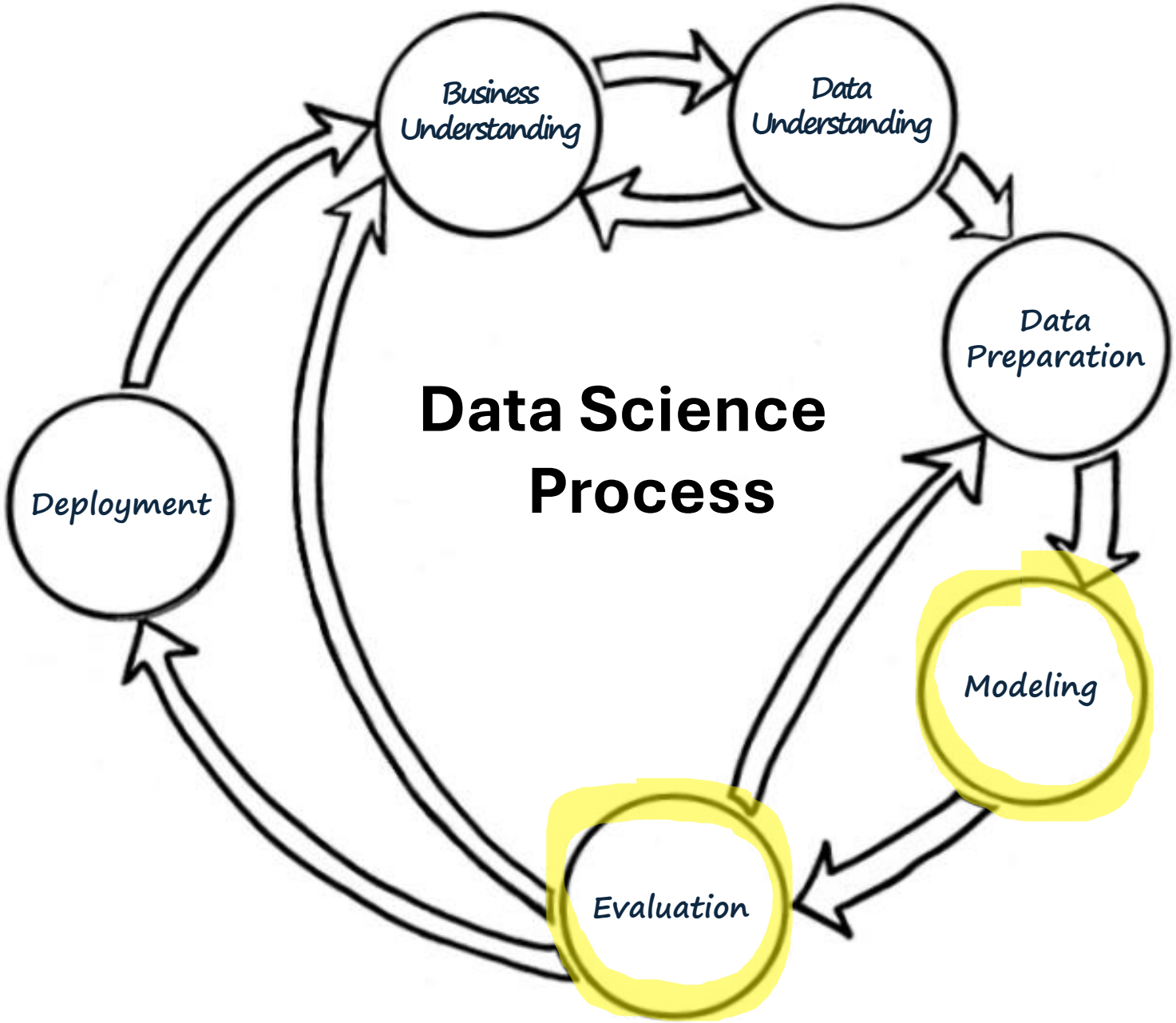
- Linear/polynomial regression
- Regression tree

Where we are
Today



Where we are

Today



Driving Question

Today

What do we want our model to work well on?

Driving Question

Today

What do we want our model to work well on?

Labeled Training Data
(Has Target Value)

OR

Unlabeled Data At Inference Time
(No Target Value)

Driving Question

Today

What do we want our model to work well on?

Labeled Training Data
(Has Target Value)

OR

Unlabeled Data At Inference Time
(No Target Value)

Driving Question Today

We'll talk about how we precisely define **"work well"** in a few classes

What do we want our model to work well on?

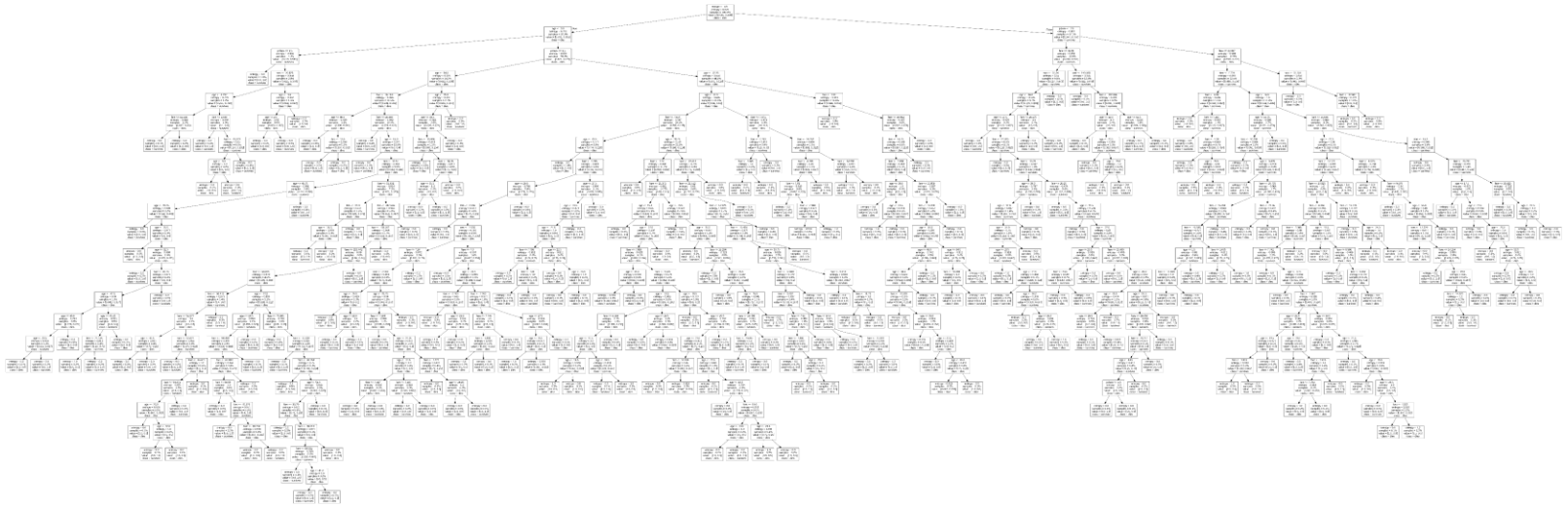
Labeled Training Data
(Has Target Value)

OR

Unlabeled Data At Inference Time (No Target Value)

BIG Decision Trees

- Consider a tree model. How do we assess how good it is?
- Easiest measure – **accuracy** – how many are correct?
- If we apply accuracy to trees built on the training set …
  - **the biggest tree will always be best!**
  - In fact, you can often build a tree with **100% accuracy on the training set**
- But, **our goal is to generalize to data we have not seen yet**



Notebook time!

Building Your Toolbox

Types of Tasks and Models

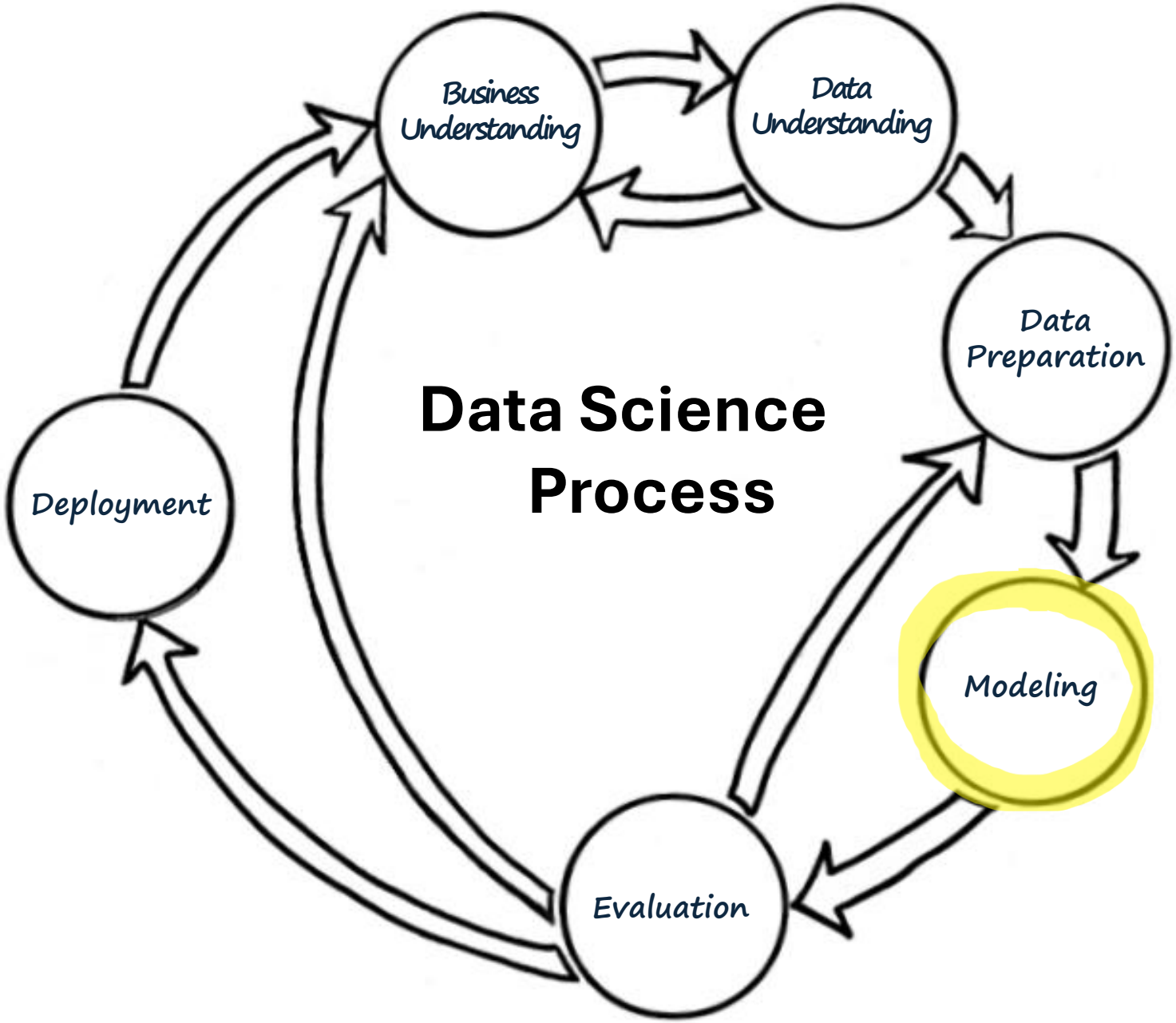
Supervised Learning

Classification / Probability Estimation

- Decision tree
- Linear separator/ logistic regression

Regression

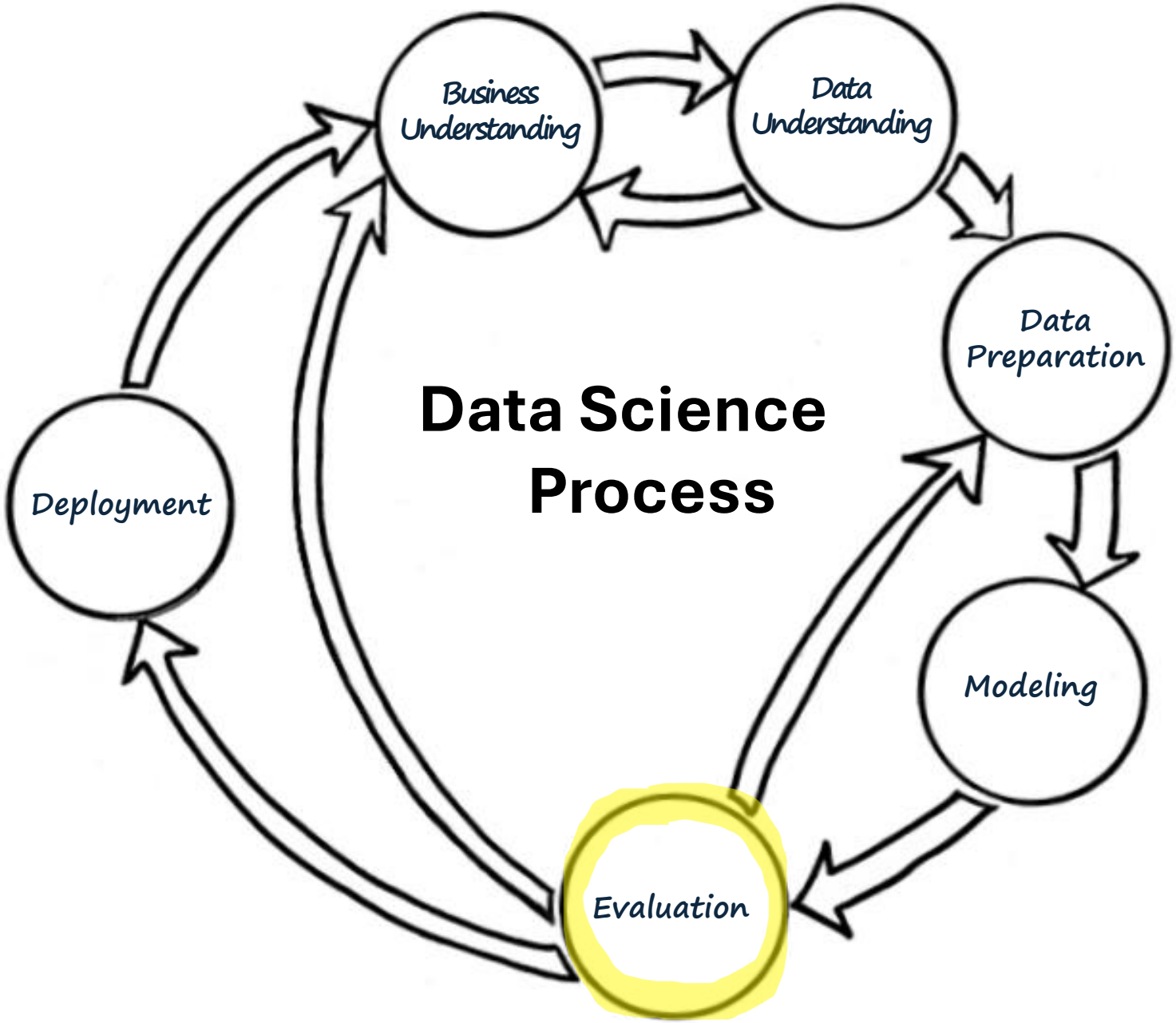
- Linear/polynomial regression
- Regression tree



Building Your Toolbox

Evaluation Techniques/Considerations

- **Overfitting** (the training data) → **worse generalization** (on unseen data)
- **K-fold cross-validation** is a way to evaluate generalization
- **Fitting curves** help us evaluate generalization (performance metric across model complexity)



Assignment time!