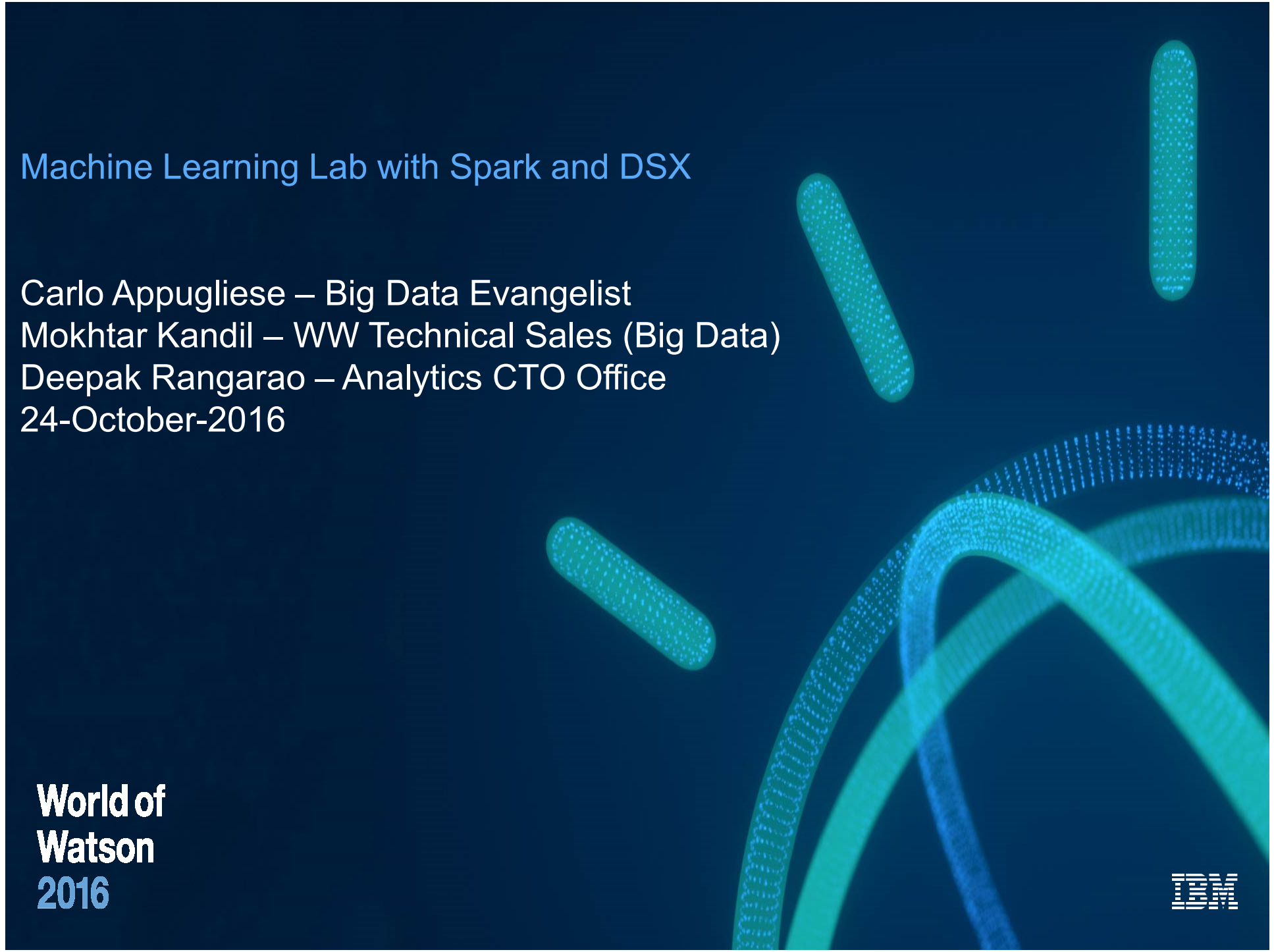
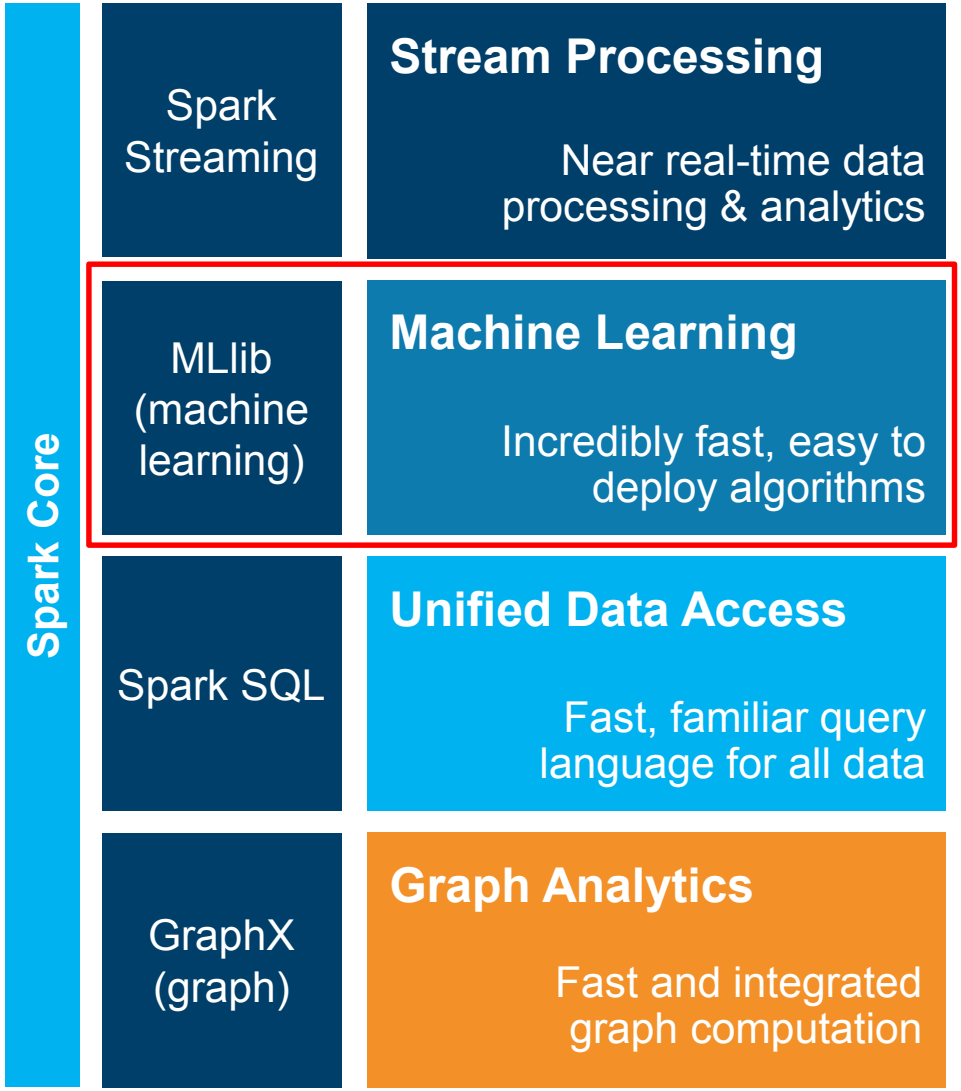
Machine Learning Lab with Spark and DSX

Carlo Appugliese – Big Data Evangelist
Mokhtar Kandil – WW Technical Sales (Big Data)
Deepak Rangarao – Analytics CTO Office
24-October-2016



Machine Learning Lab

Spark Capabilities



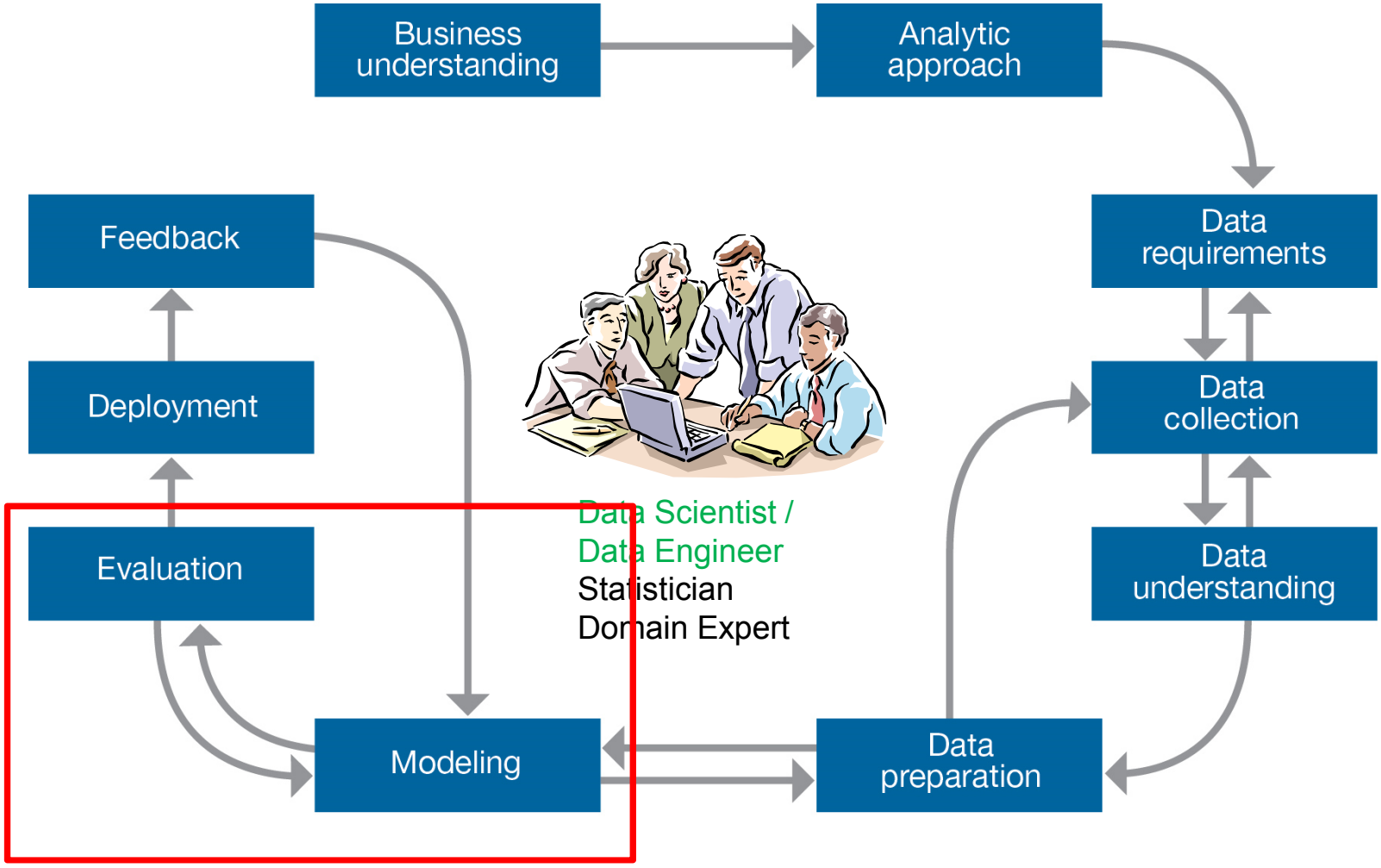
- **Micro-batch event processing** for near real-time analytics
- Process live streams of data (IoT, Twitter, Kafka)
- No multi-threading or parallel processing required

- **Predictive and prescriptive analytics**, and smart application design, from statistical and algorithmic models
- Algorithms are pre-built

- **Query your structured data sets** with SQL or other dataframe APIs
- Data mining, BI, and insight discovery
- Get results faster due to performance

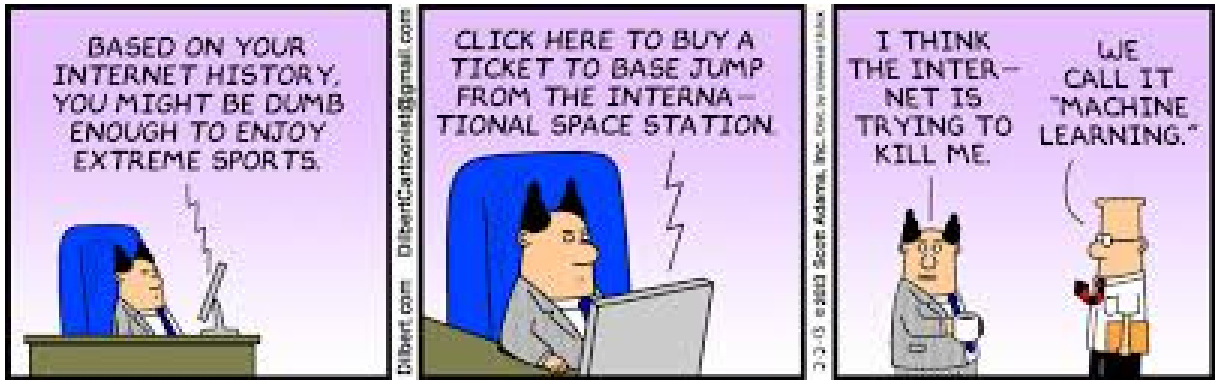
- **Represent data in a graph**
- Represent/analyze systems represented by nodes and interconnections between them
- Transportation, person to person relationships, etc.

Data Science Methodology



Machine Learning

- In 1959, Arthur Samuel defined machine learning as a "Field of study that gives computers the ability to learn without being explicitly programmed"
- Machine learning automates the development of analytical models that can learn and make predictions on data
- Machine learning allows computers to find hidden insights without being explicitly programmed where to look



Machine Learning – A more formal definition

Tom Mitchell of Carnegie Mellon University provides a widely quoted, more formal definition of machine learning

"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E"

Machine Learning vs Human Learning

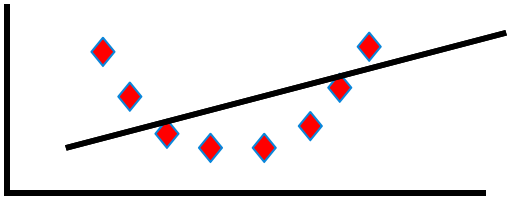
- **In many aspects, ML not fundamentally different from HL:**
  - Repeat the same task over and over again to gain experience.
  - Action of repeating the same task is referred to as “practice”
  - With practice and experience, we get better at learned tasks.

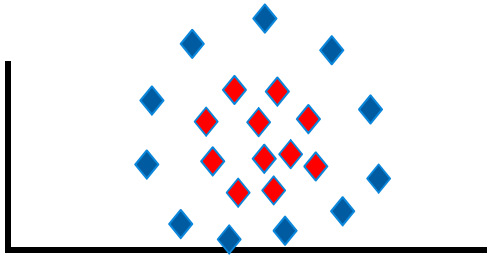
- **Examples:**
  - Learning how to play a music instrument
  - Learning how to play a sport (golf, tennis, etc…)
  - Practicing for a math exams doing exercises
  - A teacher or coach will measure performance to evaluate progress
  - Practice makes perfect

Learning challenges

- **Under fitting:**
  - Not knowing enough “basic” concepts, i.e. not being well-equipped enough to tackle learning at hand:
    - You can’t study calculus without knowing some algebra.
    - You can’t learn playing hockey without knowing how to skate.
    - You can’t learn polo without knowing how to ride.
  - This can lead to under fitting in Machine Learning: The chosen model is just not “sophisticated”, “rich”, enough to capture the concept.

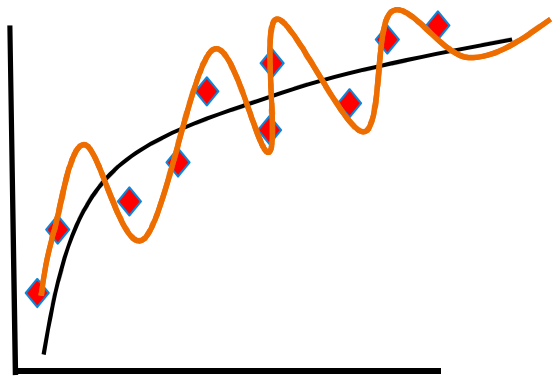
aX + BY + C



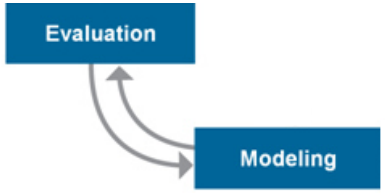


Learning challenges

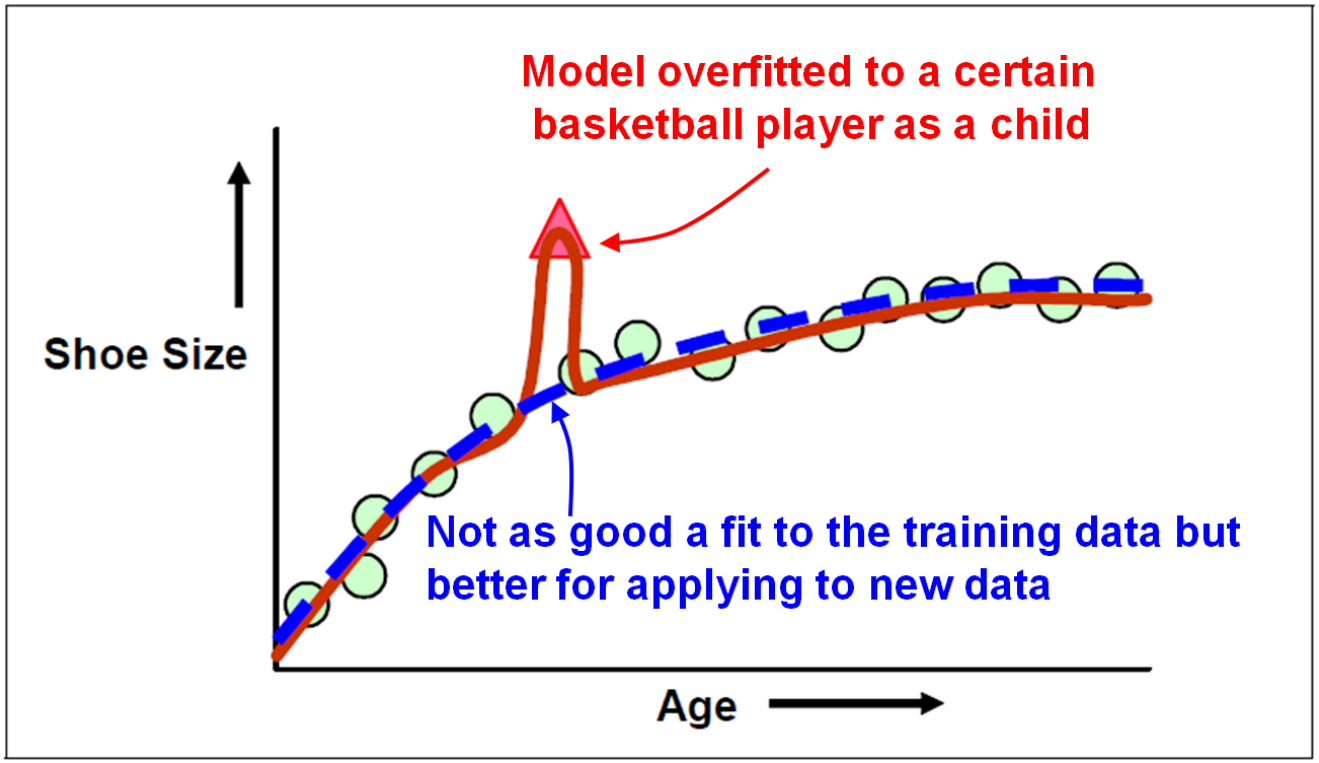
- **Over fitting:**
  - Hyper-sensitivity to minor fluctuations, ending up in modeling a lot of the unwanted noise in the data:
  - This can lead to over fitting in Machine Learning.



Model overfitting

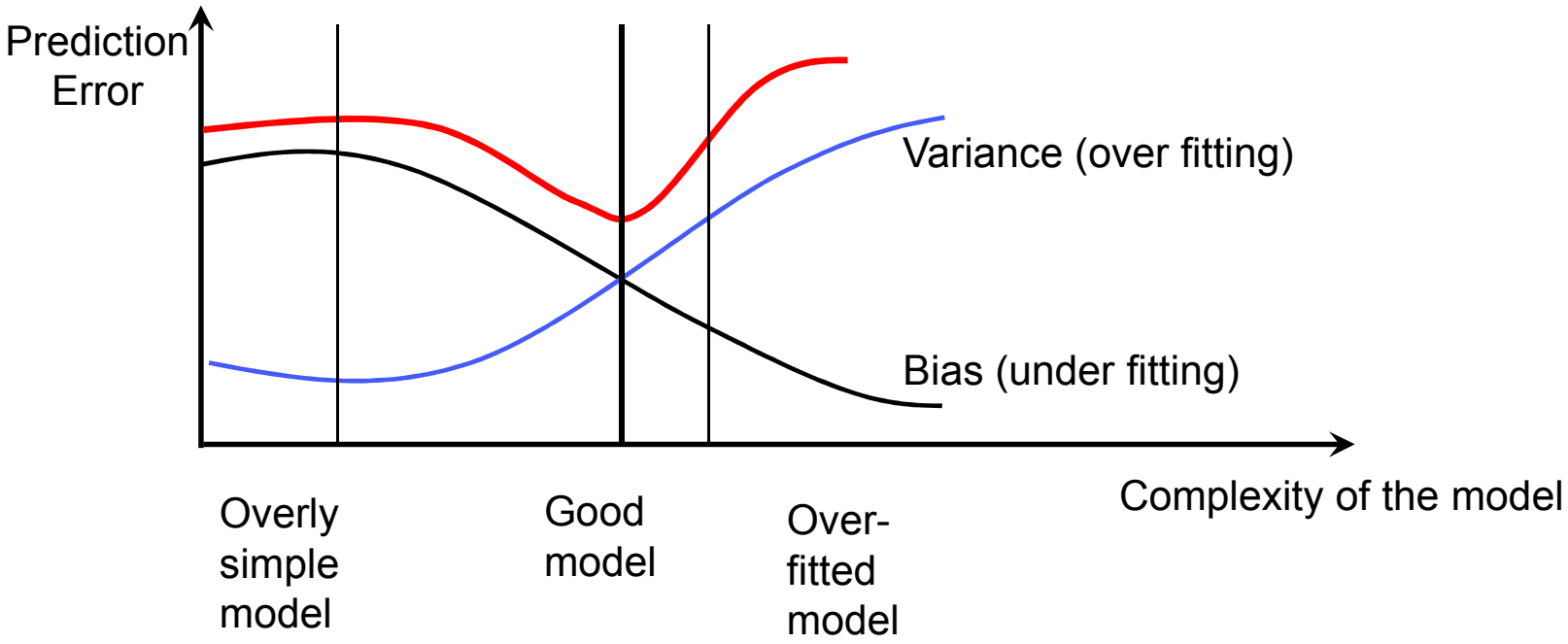


- When building a predictive model, there is a risk of overfitting the model to the training data.
- The model fits the training data very well, but it does not perform well when applied to new data.

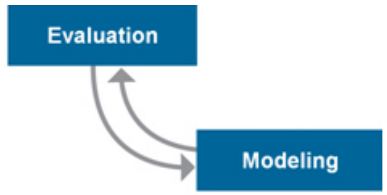


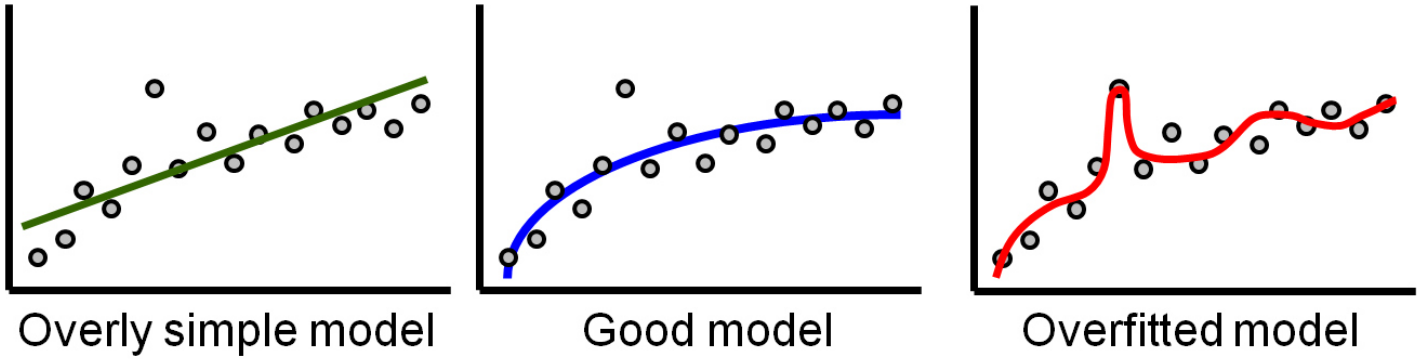
Learning challenges

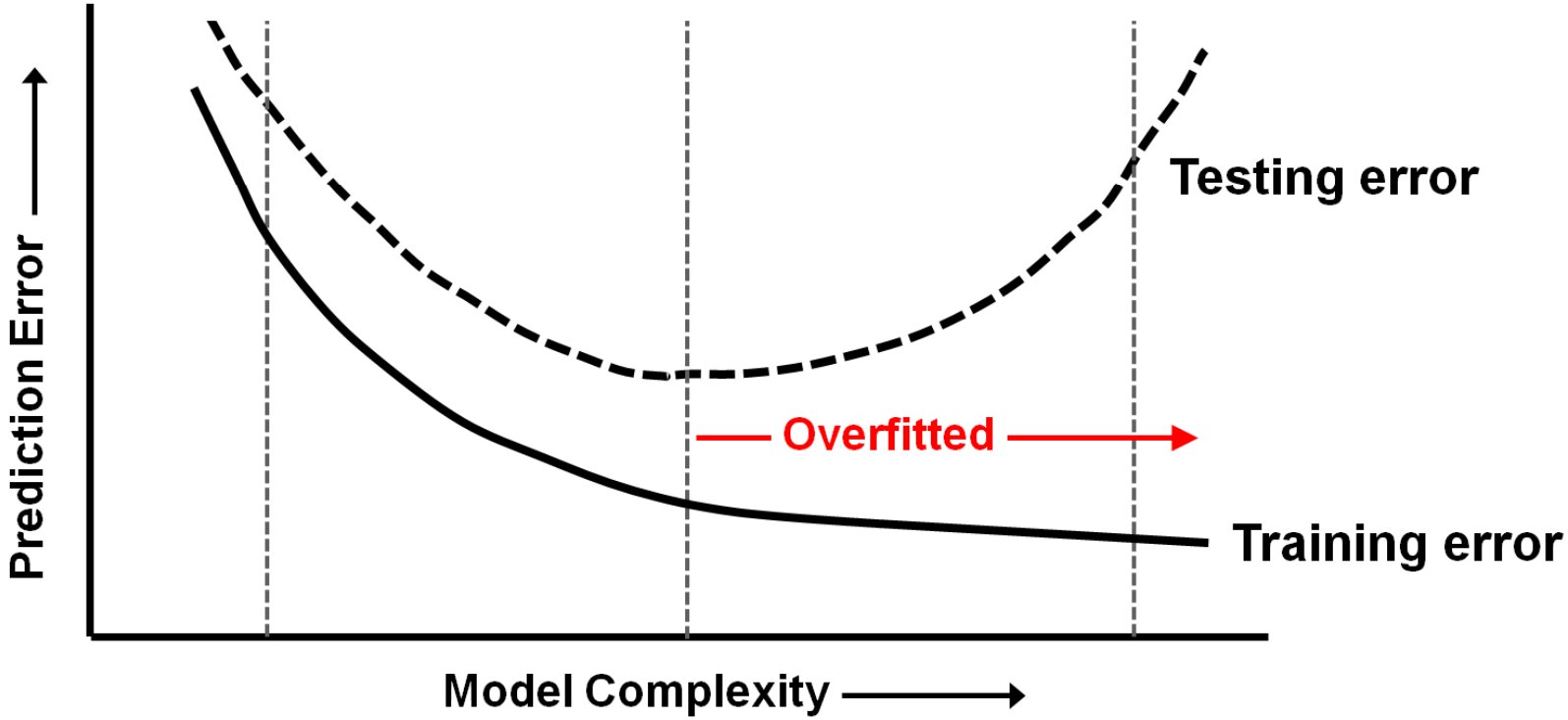
- **Compromise between bias and variance:**



When to stop training a model







Learning challenges

- **Diminishing returns:**
  - People can:
    - Have more or less talent
    - get bored or enthusiastic
  - Machines will not, however:
  - Making progress initially is usually more easy, but improving gets harder as we move along. We may need to try different learning methods, styles to keep going:
    - Machine learning algorithms have hyper-parameters which need to be tuned properly.
    - It may be necessary to use more than just one single method / algorithm to reach the goal.

Machine Learning Examples

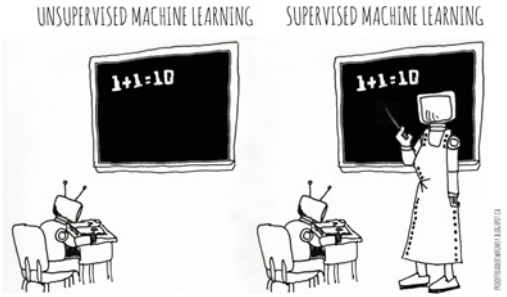
- Is this cancer ? (Medical diagnosis)
- Is this legitimate or fraud (spam) ?
- What is the market value of this house ?
- Which of these people are good friends with each other ?
- Will this engine fail (when) ?
- Will this person like this movie ?
- Who is this ?
- What did you say ? (Speech recognition)

Machine Learning solves problems that cannot be tackled by numerical means alone.

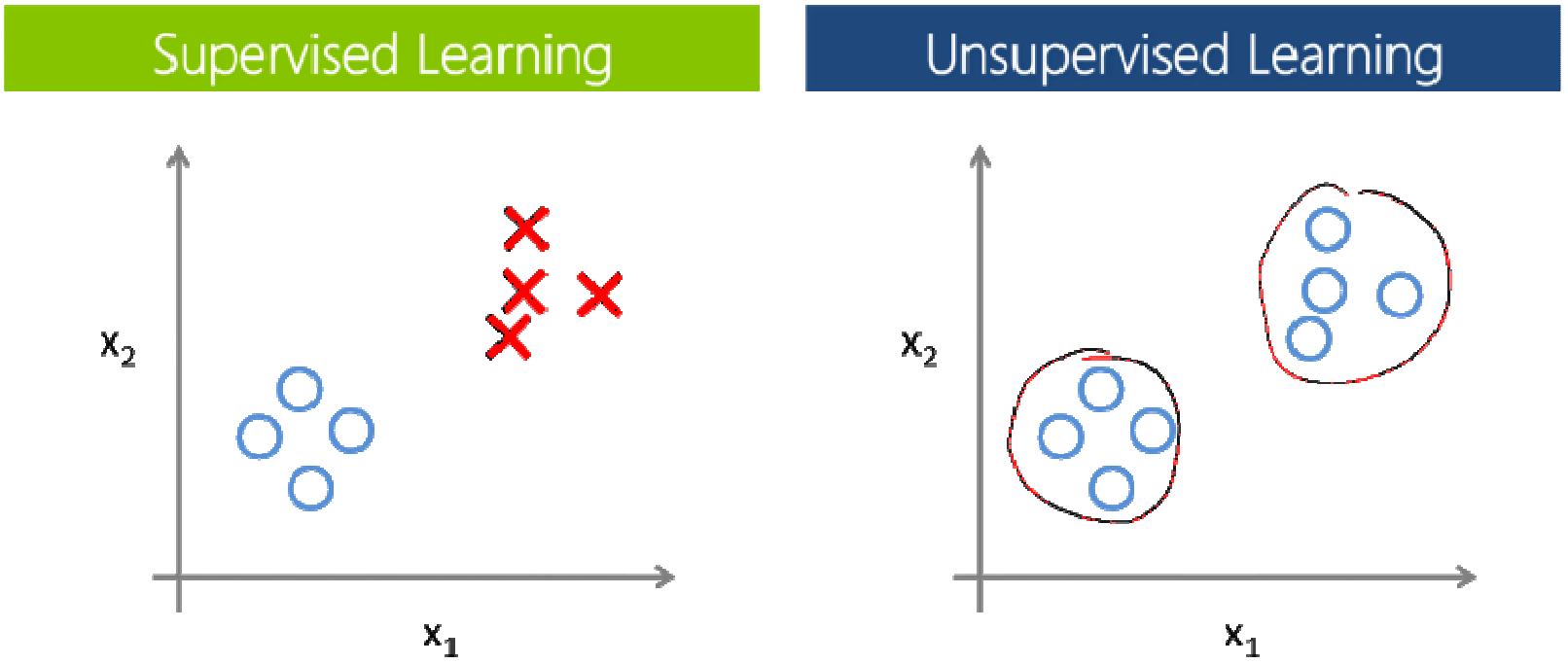
Categories of Machine Learning

- **Supervised learning**
  - The program is “trained” on a pre-defined set of “training examples”, which then facilitate its ability to reach an accurate conclusion when given new data
  - The algorithm is presented with example inputs and their desired outputs (correct results)
  - The goal is to learn a general rule that maps inputs to outputs

- **Unsupervised learning**
  - No labels are given to the learning algorithm, leaving it on its own to find structure (patterns and relationships) in its input
  - Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning)

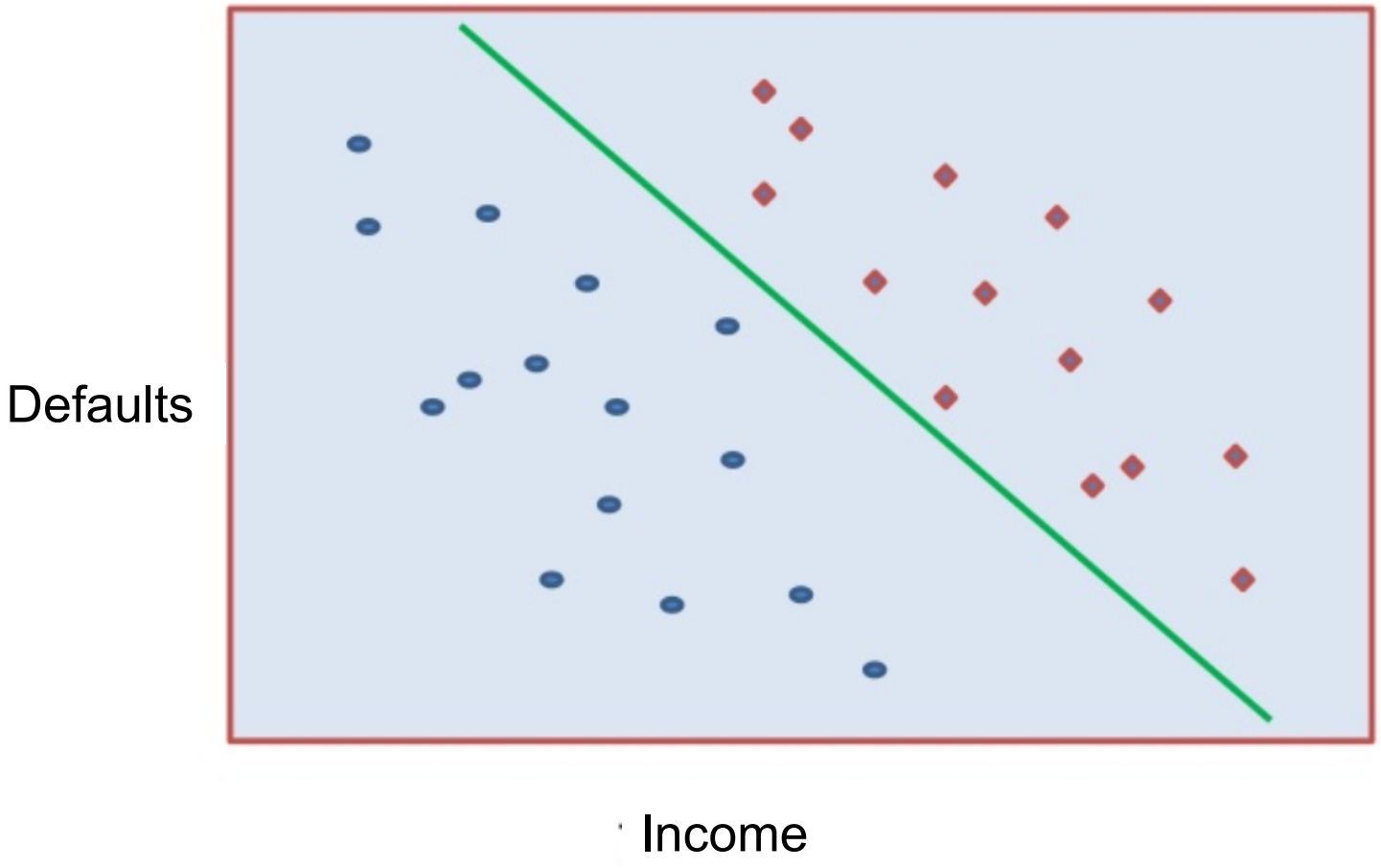


Supervised vs. Unsupervised Learning



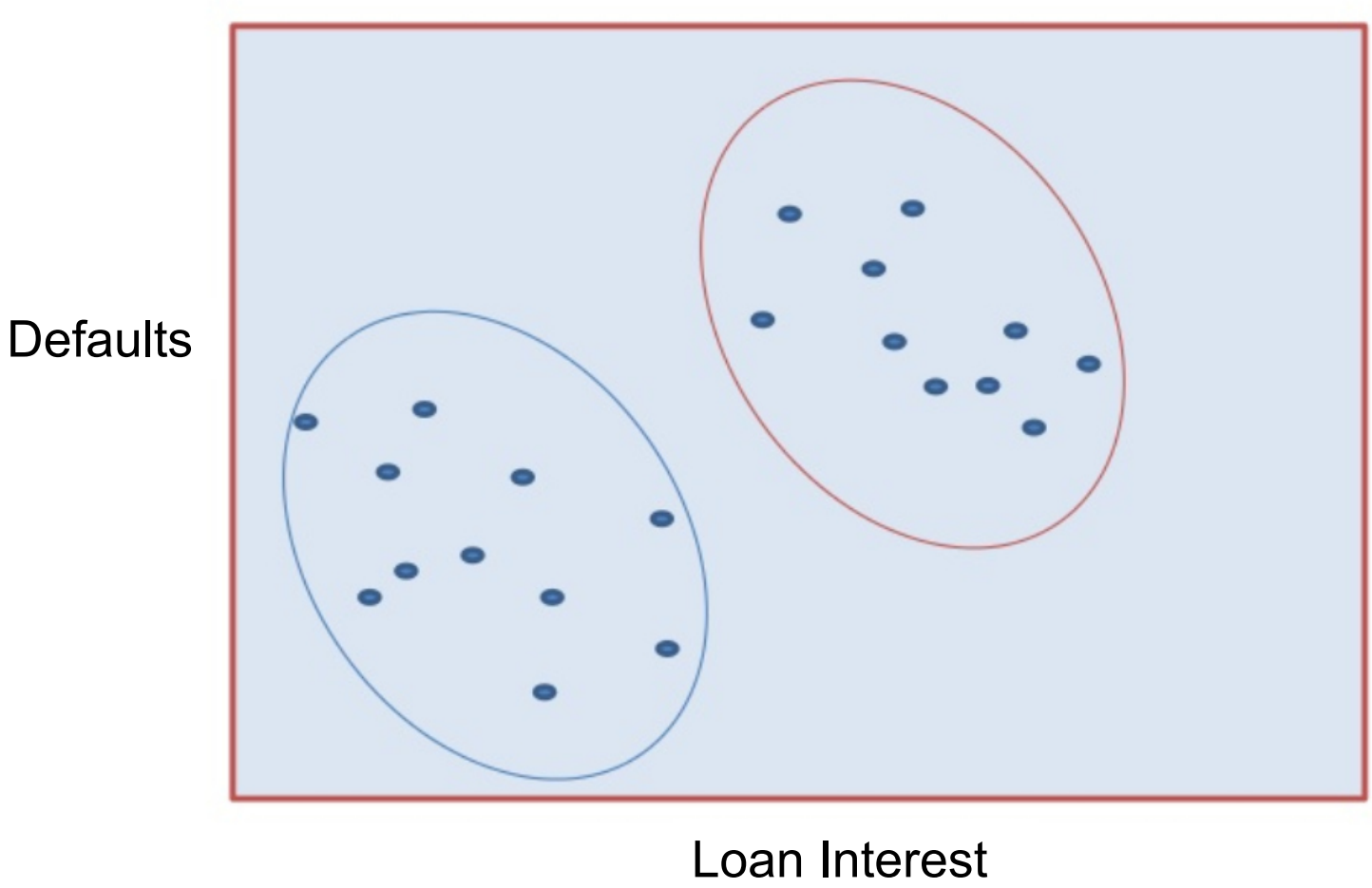
Example of Supervised Learning (Classification)

Goal is to make predictions



Example of Unsupervised Learning (Clustering)

Goal is to understand the structure of the data, not make predictions



Categories of Machine Learning

	Discrete Output	Continuous Output

Supervised Learning (require Ground-Truth)	• **Classification** (outcome is discrete)
	  • Binary Classification
	  • Detecting Fraud
	  • Predicting defaults on loans
	  • Discovering spam
	  • Predicting users who might churn
	
	  • Multi class Classification
	  • Classifying images, sounds
	  • Assigning categories to news articles, webpages, etc.…
		• **Regression**
		  - Predicting the price of a house
		  - Predicting loss amounts for loans

Unsupervised Learning (no Ground-Truth data required)	• Clustering
	  - Grouping discrete elements
	
	• Frequent Patterns and associations
	  - People who buy chips also buy beer
		• Clustering
		  - Grouping continuous variables
		
		• Dimensionality Reduction
		  - PCA
		  - SVD

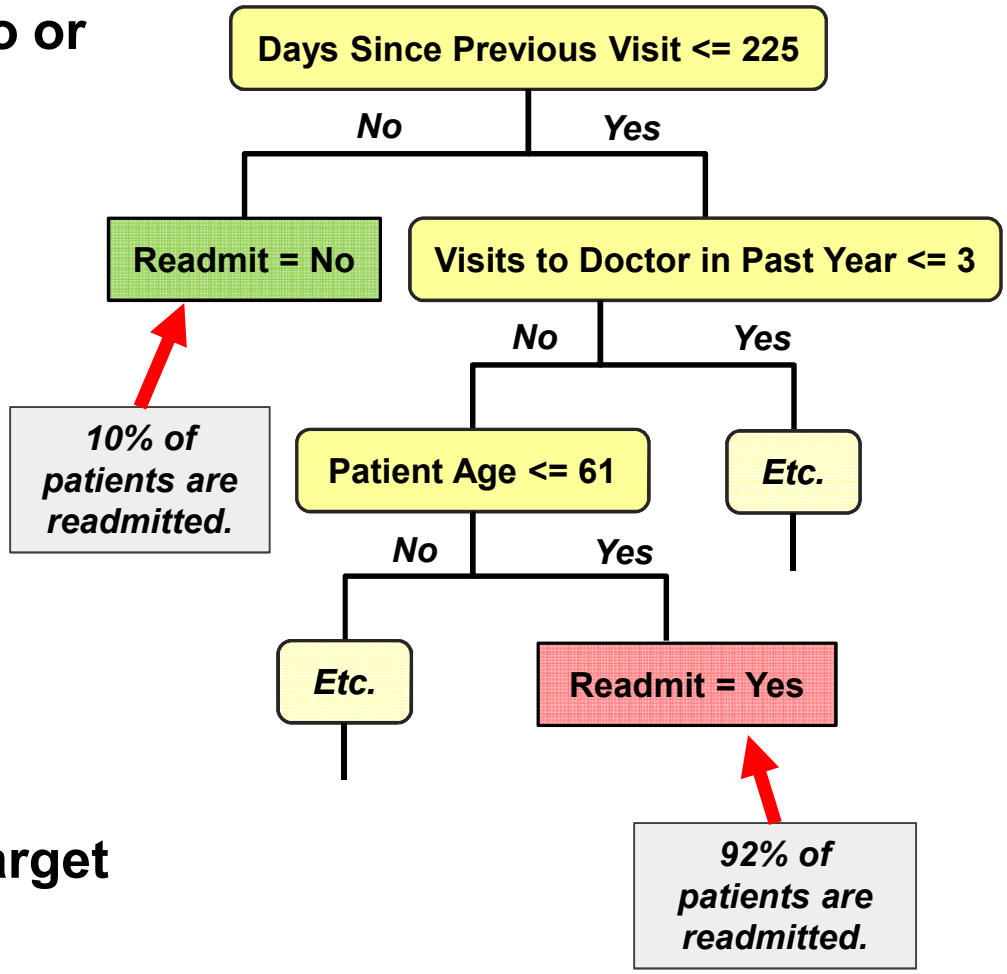
Categories of Machine Learning

	Discrete Output	Continuous Output
Supervised Learning (require Ground-Truth)	• **Classification** (outcome is discrete) • Binary Classification • Linear Models (Logistic Regression) • Decision Trees • Naïve Bayes • Multi class Classification • Decision Trees • Naïve Bayes • K-NN	• **Regression** - Linear - Ridge - Lasso • **Decision Trees** • Random Fo • G d Trees **Recommendation Engines** - Content Filtering - Collaborative Filtering
Unsupervised Learning (no Ground-Truth data required)	• Clustering - k-means • FP-Growth	• Clustering - k-means - Gaussian Mixture • Dimensionality Reduction - PCA - SVD

Classification – Decision tree

Modeling

- **Class variable (target) with two or more outcomes.**
- **Splits records in a tree-like series of nodes along mutually-exclusive paths.**
  - Algorithm decides which variable and threshold value to use at each split
  - New records are predicted (classified) based on the leaf assignment
  - Accurate
  - Explicit decision paths
- **Can also handle continuous target ("regression tree").**



Classification – Naïve Bayes

Modeling

- **Two or more outcomes.**
- **Assumes independence among explanatory variables, which is rarely true (thus "naïve").**
- **Despite its simplicity, often performs very well… widely used.**
- **Significant use cases:**
  - Text categorization (spam vs. legitimate, sports or politics, etc.) using word frequencies as the features
  - Medical diagnosis (*e.g.*, automatic screening)

Classification – Naïve Bayes

Modeling

Outlook	Temp	Humidity	Windy	Play golf
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Classification – Naïve Bayes

Modeling

Frequencies and probabilities for the weather data:

outlook			temperature			humidity			windy			play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2	3	hot	2	2	high	3	4	false	6	2	9	5
overcast	4	0	mild	4	2	normal	6	1	true	3	3		
rainy	3	2	cool	3	1								
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2/9	3/5	hot	2/9	2/5	high	3/9	4/5	false	6/9	2/5	9/14	5/14
overcast	4/9	0/5	mild	4/9	2/5	normal	6/9	1/5	true	3/9	3/5		
rainy	3/9	2/5	cool	3/9	1/5								

Classification – Naïve Bayes

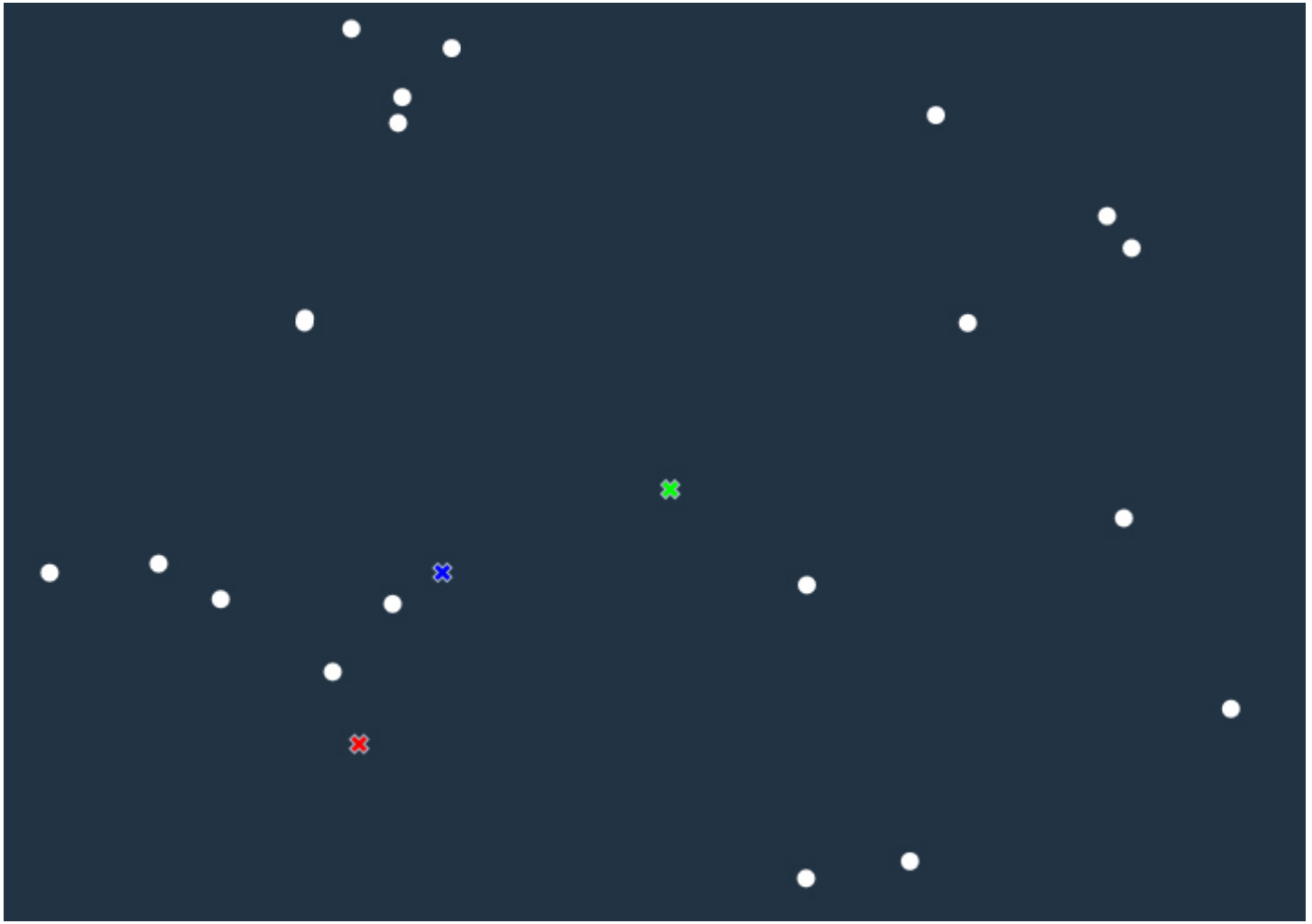
- L(yes) = 2/9 * 3/9 * 3/9 * 3/9 = 0.0082
- L(no) = 3/5 * 1/5 * 4/5 * 3/5 = 0.0577

- P(yes) = 0.0082 * 9/14 = 0.0053
- P(no) = 0.0577 * 5/14 = 0.0206

- The decision would be: NO.

Clustering – K-means method

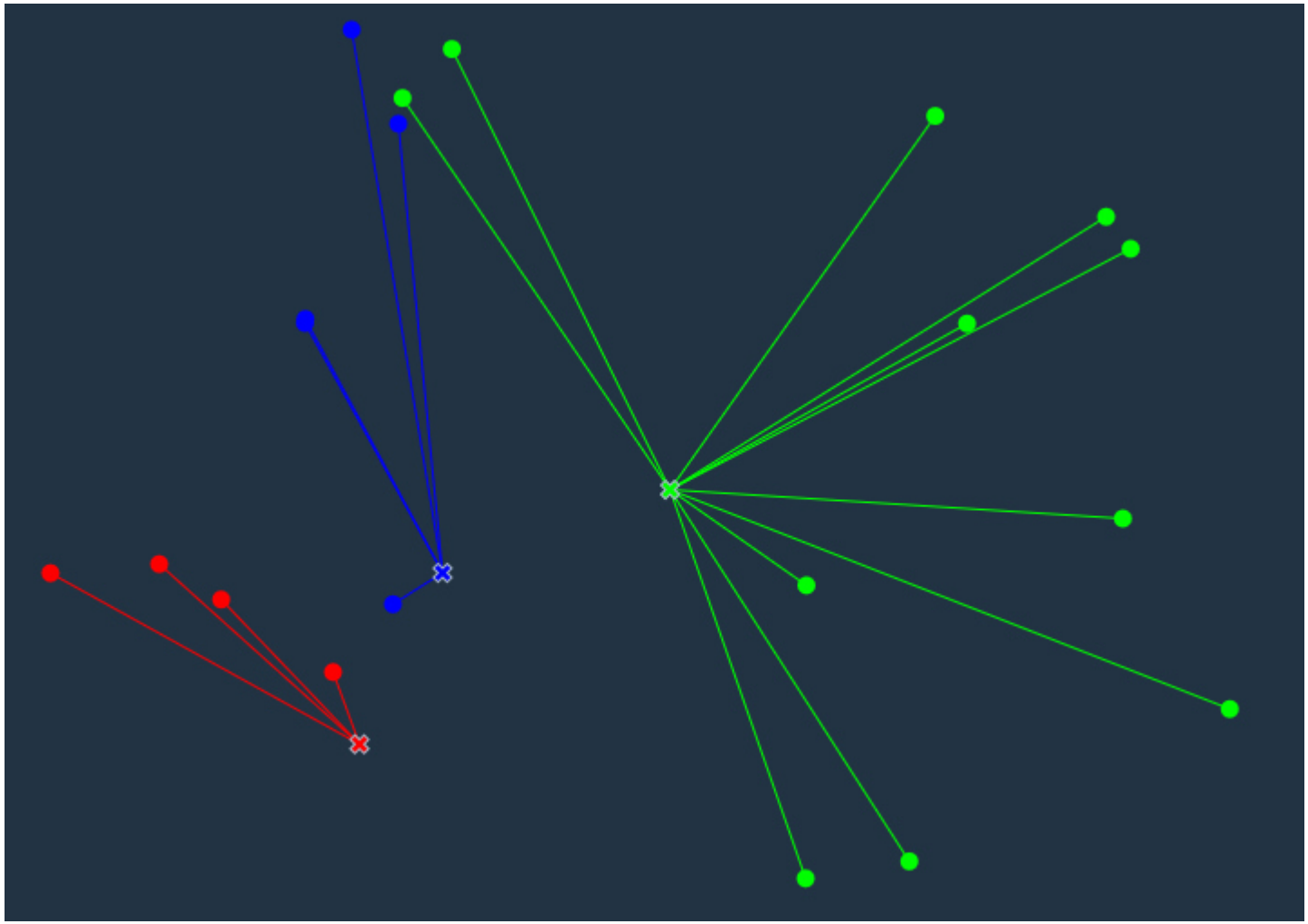
Modeling



Start with 20 data points and 3 clusters

Clustering – K-means method

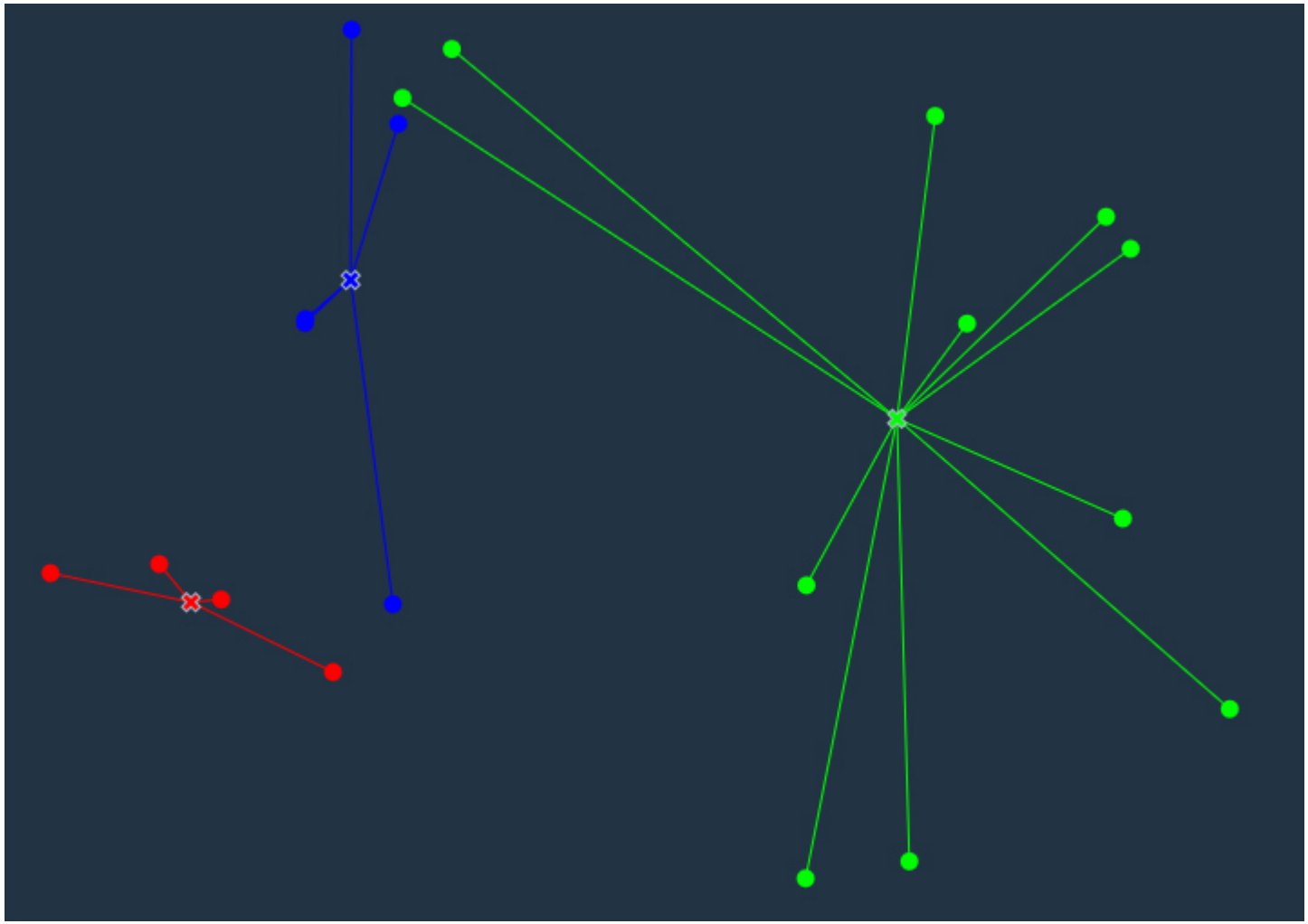
Modeling



Assign each data point to the nearest cluster

Clustering – K-means method

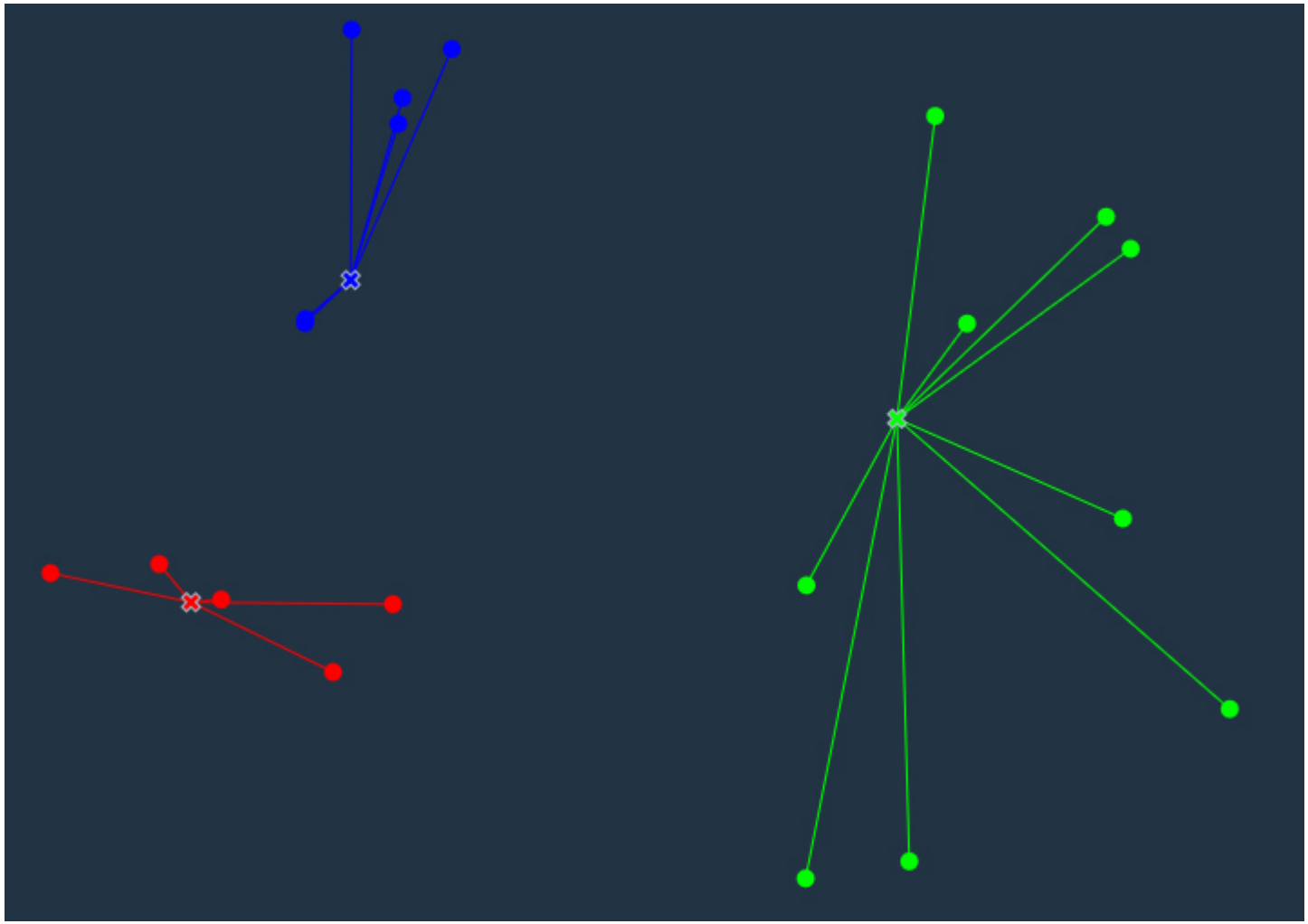
Modeling



Calculate centroids of new clusters

Clustering – K-means method

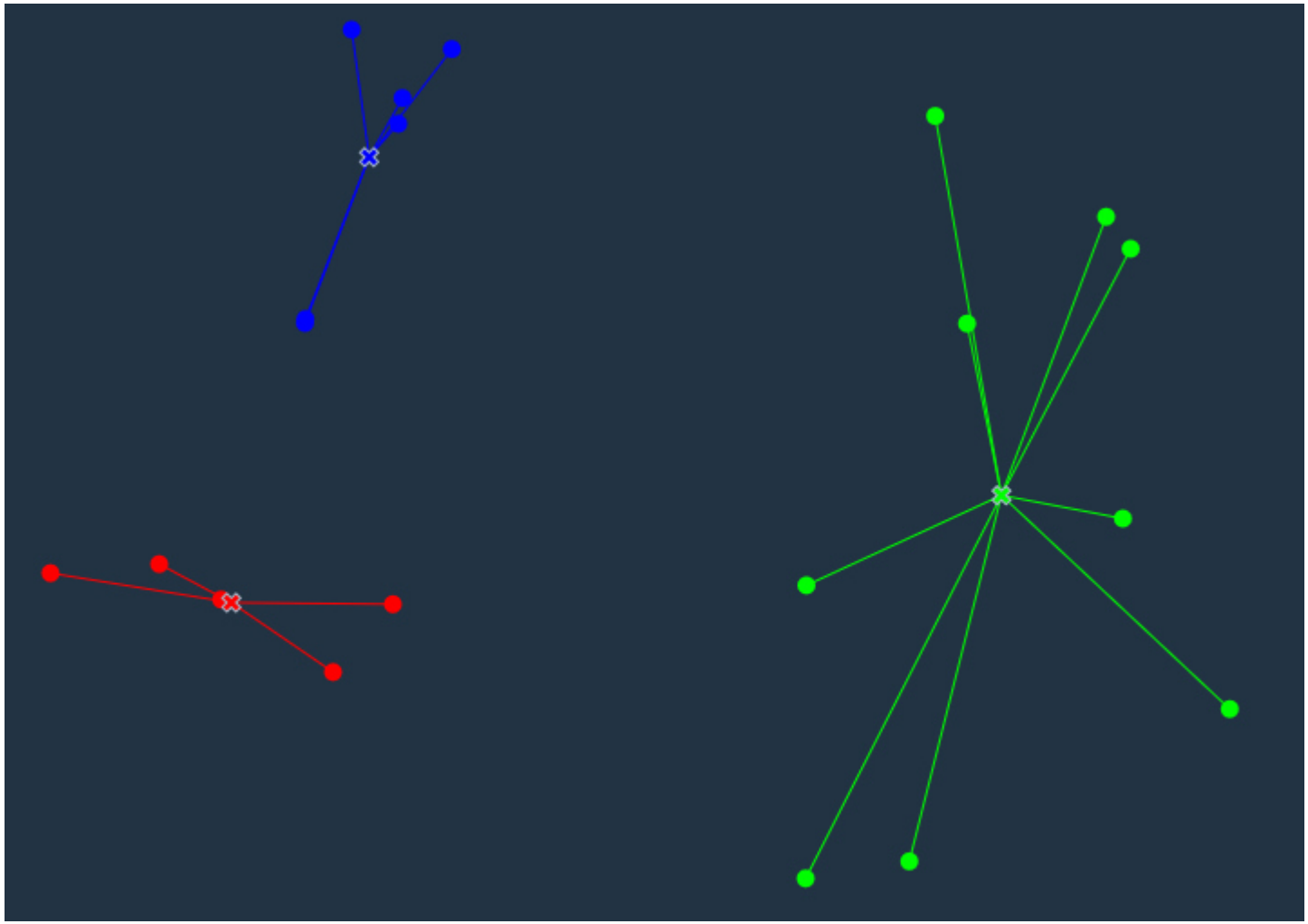
Modeling



Assign each data point to the nearest cluster

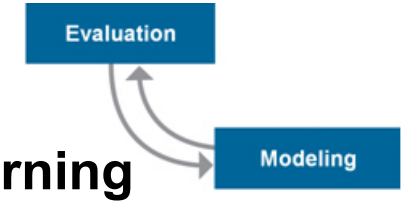
Clustering – K-means method

Modeling



Calculate centroids of new clusters…until convergence

Training, testing, & validation sets



- **During the model development process, supervised learning techniques employ training and testing sets and sometimes a validation set.**
  - Historical data with known outcome (*target*, *class*, *response*, or *dependent variable*)
  - Source data randomly split or sampled… mutually exclusive records
- **Why?**
  - Training set ➔ build the model (**iterative**)
  - Testing set ➔ tune the parameters & variables during model building (**iterative**)
    - Assess model quality during training process
    - Avoid overfitting the model to the training set
  - Validation set ➔ estimate accuracy or error rate of model (**once**)
    - Assess model's expected performance when applied to new data

Spark ML

- Spark ML is Spark's machine learning (ML) library
- Its goal is to make practical machine learning scalable and easy
- Consists of common learning algorithms and utilities, including
  - Classification
  - Regression
  - Clustering
  - Collaborative filtering
  - Dimensionality Reduction
- Lower-level optimization primitives
- Higher-level pipeline APIs

Spark ML

- Divides into two packages:
  - spark.mllib contains the original API built on top of RDDs
  - spark.ml provides higher-level API built on top of DataFrames for constructing ML pipelines

- Using spark.ml is recommended because with DataFrames the API is more versatile and flexible
  - spark.mllib will continue to be supported

Recommendation Systems

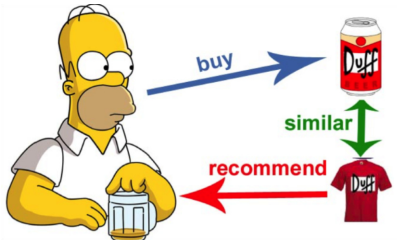
- **Recommendation systems seek to predict the rating (or preference) that a user would give to an item**
- **Recommendation systems attempt to improve customer experience through personalized recommendations based on prior user feedback**
- **Recommender systems have become extremely common in recent years, and are applied in a variety of applications**
  - movies, music, news, books, research articles, search queries, social tags, …
  - products in general
- **Collaborative filtering is a technique that is commonly used for recommender systems**
  - employs a form of wisdom of the crowd approach

Collaborative Filtering with Spark ML

- **Forms of Collaborative Filtering**
  - Explicit matrix factorization - preferences provided by users themselves are utilized
  - Implicit matrix factorization - only implicit feedback (e.g. views, clicks, purchases, likes, shares etc.) is utilized

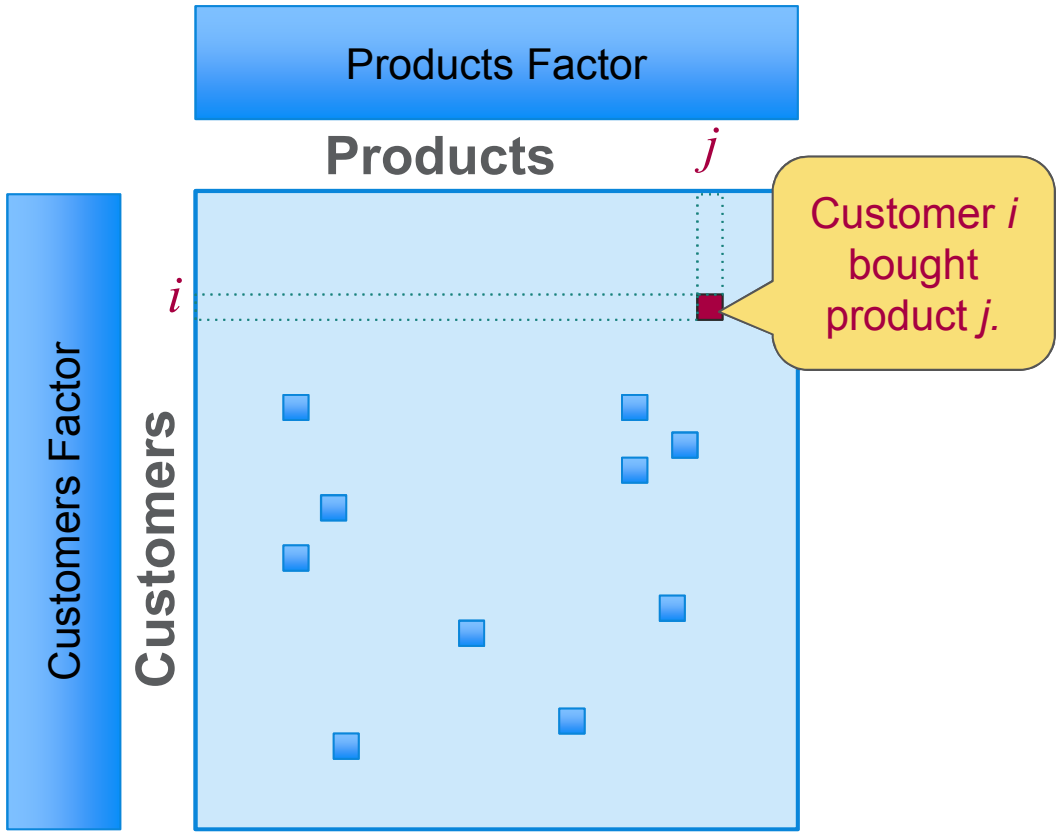
- **Spark ML supports an implementation of matrix factorization for collaborative filtering**
  - Matrix factorization models have consistently shown to perform extremely well for collaborative filtering

- **Collaborative filtering aims to fill in the missing entries of a user-item association matrix in which users and items are described by a small set of latent factors that can be used to predict missing entries**



Running Example:
Collaborative Filtering

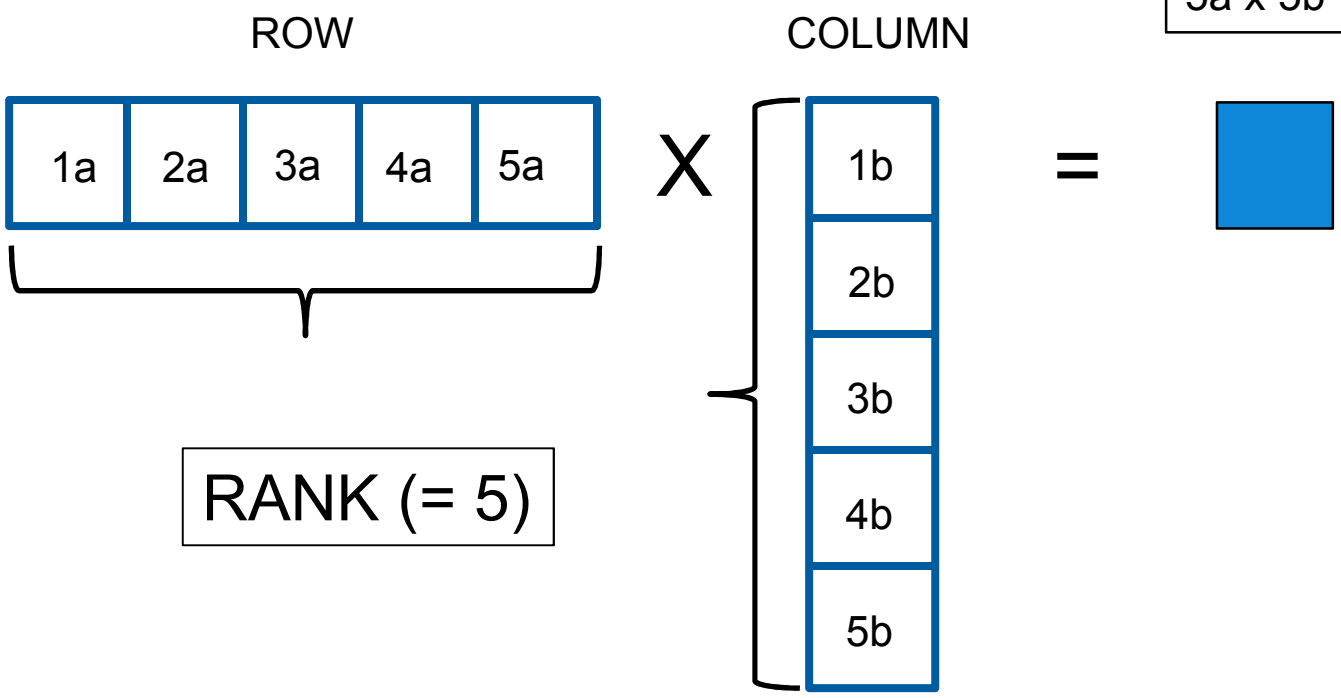
- **Problem:**
Recommend products to customers



Running Example:
Collaborative Filtering

DOT PRODUCT of ROW x COLUMN

1a x 1b +
2a x 2b +
3a x 3b +
4a x 4b +
5a x 5b

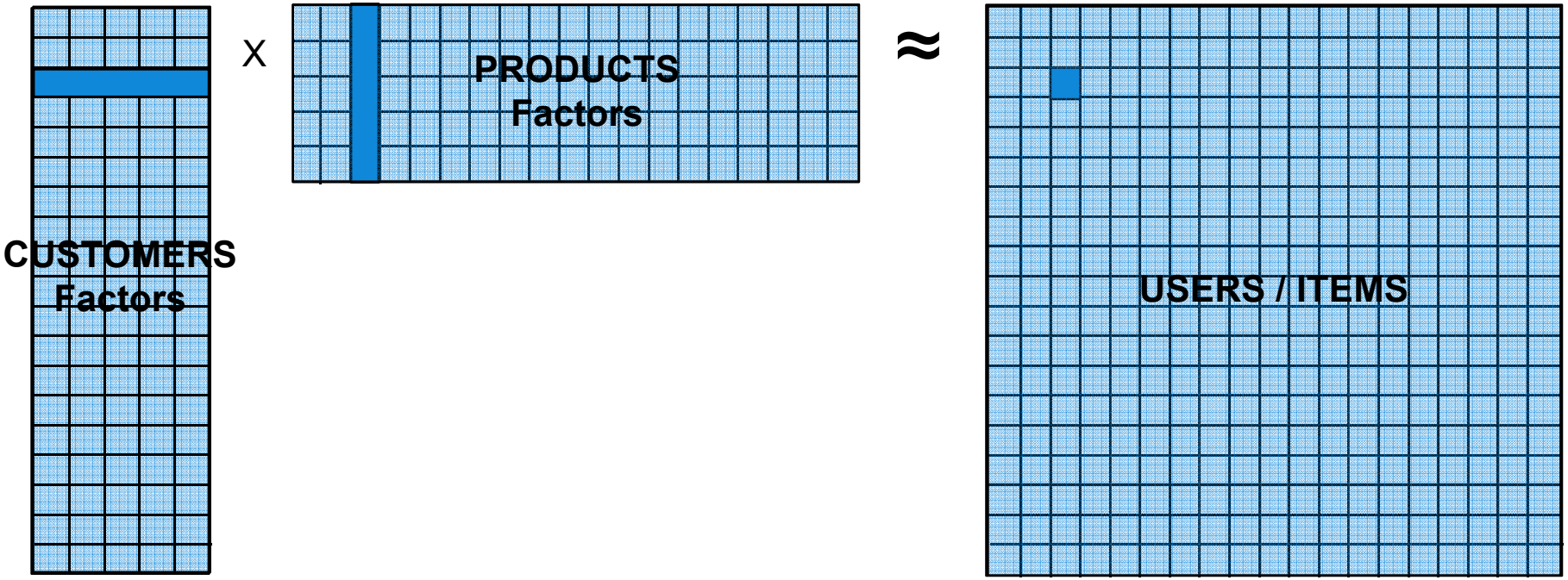


RANK (= 5)

Running Example:

Collaborative Filtering

Product of CUSTOMERS x PRODUCTS Factors

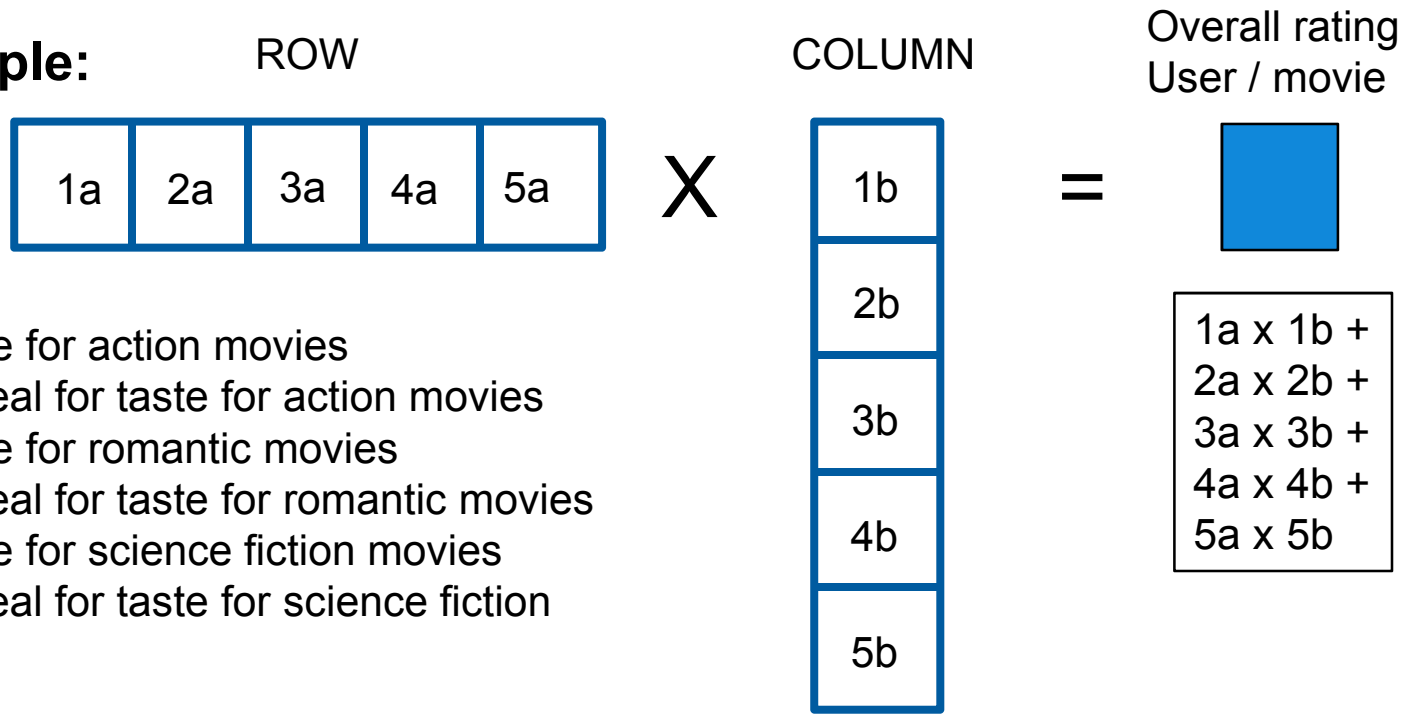


Collaborative Filtering

- **This approach is therefore building on the assumption that each user rating r_{ij} from the original matrix can be decomposed / explained as a combination of those k factors (the rank).**

- **Each user is represented by k "tastes"**
- **Each item is represented by k "appeals" (to those tastes)**

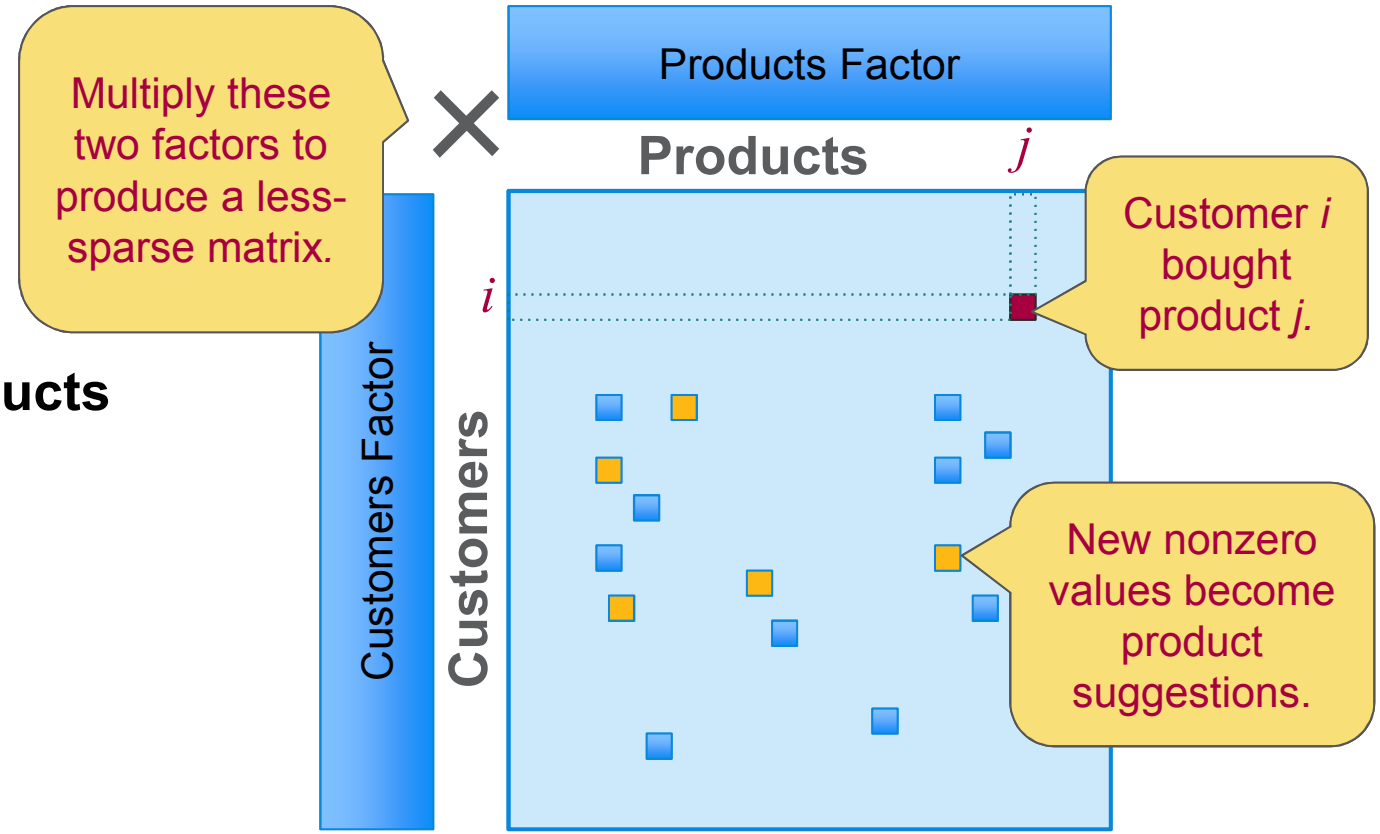
- **Example:**



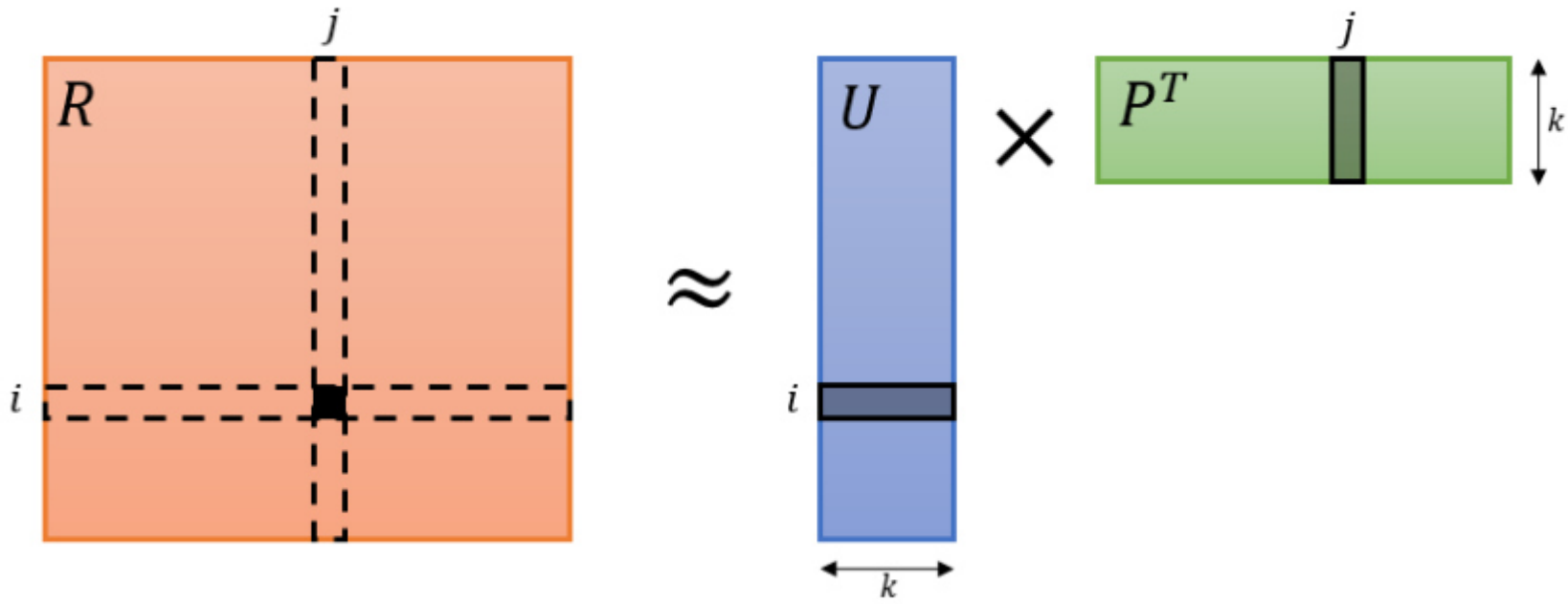
1a: Taste for action movies
1b: Appeal for taste for action movies
2a: Taste for romantic movies
2b: Appeal for taste for romantic movies
3a: Taste for science fiction movies
3b: Appeal for taste for science fiction
Etc…

Running Example:
Collaborative Filtering

- **Problem: Recommend products to customers**



Running Example:
Alternating Least Squares



- **Mathematically, we need to minimize the function: $\| R - U \times P^T \|^2$**

➔ $$\sum (r_{ij} - u_i \times p_j)^2$$

Running Example:
Alternating Least Squares

- $\sum (r_{ij} - u_i \times p_j)^2$

- **Somewhat difficult to solve because there are two unknowns: u_i and p_j**

- **The idea of ALS is to transform this into a simpler problem by giving values to p_j and finding u_i and then switching.**

- **Example:**
  - Find two factors of 532: X * Y = 532.
  - 1) Guess one factor at “4” (some small random value).
  - 2) Second factor is between 100 (4x100=400) and 2000 (4x200 = 800) : “150”
  - 3) Take 150 as fixed and now work on the other factor.
  - 4) 150 x 4 is too big (600) and 150 x 3 is too small (450): 3.5
  - 5) Take 3.5 as fixed and go back to estimating the first parameter
  - 6) Continue this “Alternating” approach of fixing one side and estimating the other side, until we get close enough to the original value

Lab Flow

1. Section 0

1. Download compressed CSV data and load into an RDD

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/10 8:26	2.55	17850	United Kingdom
536365	71053	WHITE METAL LANTERN	6	12/1/10 8:26	3.39	17850	United Kingdom
536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/10 8:26	2.75	17850	United Kingdom

2. Prepare the data

– Remove header
– Only keep rows that have
  - a purchase quantity greater than 0
  - a non blank customer ID
  - a non blank stock code after removing non-numeric characters

3. Create a dataframe

– Add a label column

4. Split the dataset

– 80% for training
– 20% for testing
– (Can add cross validation)

Lab 3 Flow (continued)

1. **Section 1: MLlib explicit feedback**
   1. Build test RDD
   2. Get predictions for the test RDD
   3. Join predictions results with original labels (customer ratings)
   4. Calculate Mean Squared Error (compare across models, tune hyper params)
   5. Evaluate recommendations for a customer
   6. Take a look under the hood at ALS internals

2. **Section 2: MLlib implicit feedback**
   1. Repeat similar steps from section 1

3. **Section 3: ML explicit feedback**
   1. Repeat similar steps from section 1
   2. Compare MLlib and ML interfaces

4. **Section 4: ML implicit feedback**
   1. Repeat similar steps from section 1
   2. Compare MLlib and ML interfaces

Surveys/uGifting

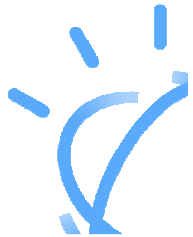
We’re all ears! How was your IBM World of Watson 2016 experience?

Let us know how we did! Complete your session surveys daily, as well as the overall conference survey, available on the IBM Events mobile app beginning Wednesday at 8:00am.

Each session survey earns you WoWBUCK$ bringing you closer to winning an Apple TV, sponsored by Cvent. 1000 points gets you into the drawing.*

After completing the overall conference survey, Clients and IBM Business Partners* will be provided a $20
e-voucher that can be applied toward the purchase of an item of your choice at the IBM Logo Store or the IBM Bookstore, or you can choose to donate those funds to charity.

Clients and IBM Business Partners, visit the “Redeem your gift” page in your IBM Events mobile app for full
details and restrictions.



*** Clients and IBM Business Partners only. Public sector employees are not eligible. Full rules at ibmevents.tumblr.com. Vouchers not valid on prior purchases.**