Makine Öğrenmesi

"Machine Learning and Data Analytics"

Dr. Cahit Karakuş

- **Hesaplamalı düşünme:** hesaplama yoluyla sorunlara yaklaşmanın yeni bir yolu
  - Soyutlama, ayrıştırma, modülerlik,…
- **Veri bilimi:** veri açısından karmaşık probleleri çözmek için disiplinler arası bir yaklaşım
  - Makine öğrenimi, büyük ölçekli bilgi işlem, anlamsal meta veriler, iş akışları,…

Machine Learning and Data Analytics

I. Machine learning and data analysis tasks

II. Classification
  - Classification tasks
  - Building a classifier
  - Evaluating a classifier

III. Pattern learning and clustering
  - Pattern detection
  - Pattern learning and pattern discovery
  - Clustering
    - K-means clustering

IV. Causal discovery
  - Correlation
  - Causation
  - Causal models
    - Bayesian networks
    - Markov networks

V. Simulation and modeling

VI. Practical use of machine learning and data analysis

Machine Learning and Data Analytics

I. Makine öğrenimi ve veri analizi görevleri
II. Sınıflandırma
  - sınıflandırma görevleri
  - Bir sınıflandırıcı oluşturma
  - Bir sınıflandırıcıyı değerlendirme
III. Örüntü öğrenme ve kümeleme
  I. Desen algılama
  II. Örüntü öğrenme ve desen keşfi
  III. Kümeleme: K-kümeleme

IV. Causal discovery
  - Korelasyon
  - Nedensellik
  - nedensel modeller
    - Bayesian networks
    - Markov networks
V. Simulation and modeling
VI. Practical use of machine learning and data analysis

Different Data Analysis Tasks

- Her görev türü, ihtiyaç duydukları veri türleri ve ürettikleri çıktı türleri ile karakterize edilir.
- Her görev türü farklı algoritmalar kullanır

- **Classification**
  - Yeni bir örnek için bir kategori (yani bir sınıf) atanır.
- **Clustering**
  - Bir dizi örnekle kümeler (yani gruplar) oluşturulur.
- **Pattern detection**
  - Zamansal veya uzamsal verilerdeki düzenlilikleri (yani kalıpları) tanımlanır.
- **Simulation**
  - Toplanan gözlemlere benzer veriler üretebilen matematiksel formülleri tanımlanır.

Learning Approaches

Supervised Learning

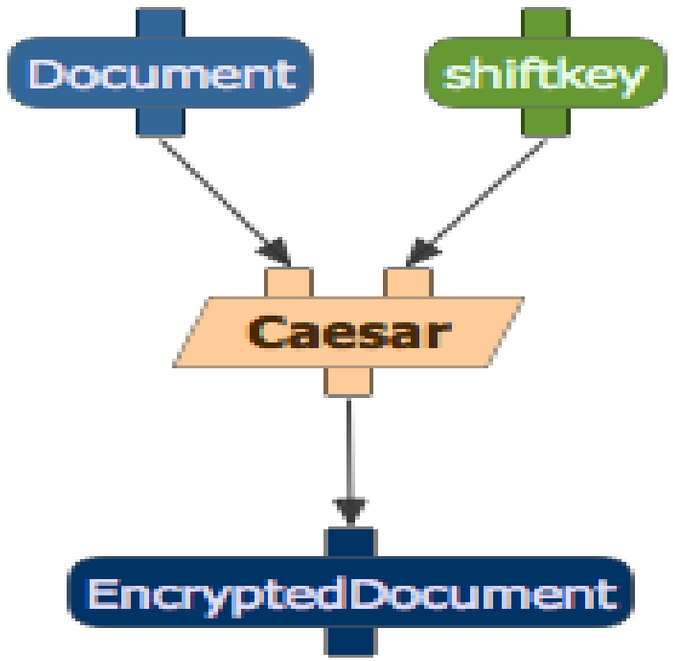
- Eğitim verileri, öğrenme sistemine yardımcı olacak bilgilerle açıklanır

Unsupervised Learning

- Eğitim verilerine, öğrenme sistemine yardımcı olmak için herhangi bir ek bilgi eklenmez.

Programlara “Kara Kutular” Olarak Bakın

- Yazılımı kullanmak için karmaşık matematik ve programlamayı anlamanız gerekmez
- Bu nedenle yazılıma genellikle “kara kutu” diyoruz.
- Doğru kullanabilmek için sadece girdileri ve çıktıları ve programın işlevini anlamanız gerekir.
- Fonksiyon Olarak Programlar: Girişler, Çıkışlar ve Parametreler
- Fonksiyonların Bileşimi Olarak İş Akışı

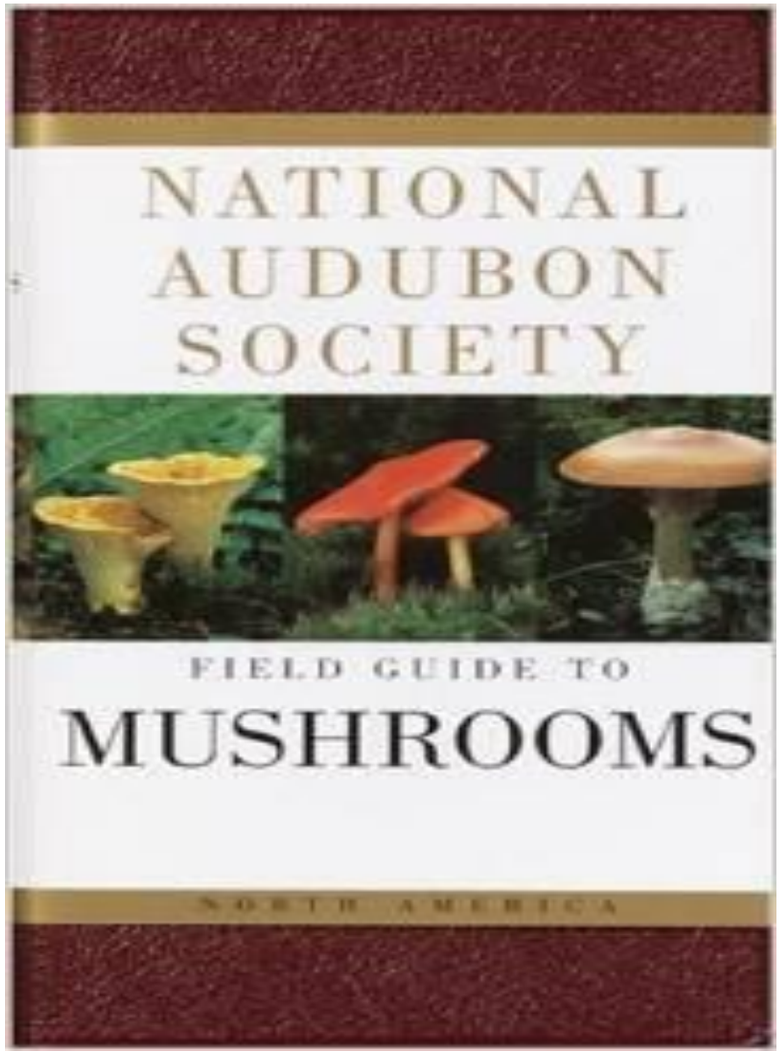


Classification

Topics

1. Classification tasks
2. Building a classifier
3. Evaluating a classifier

Classifying Mushrooms



- Hangi mantarlar yenilebilir, yani zehirli değil?
- Soldaki kitap, yenilebilir, zehirli veya bilinmeyen olarak tanımlanan birçok mantar türünü listeler.
- Kitapta listelenmeyen yeni bir tür mantar verildiğinde yenilebilir mi?

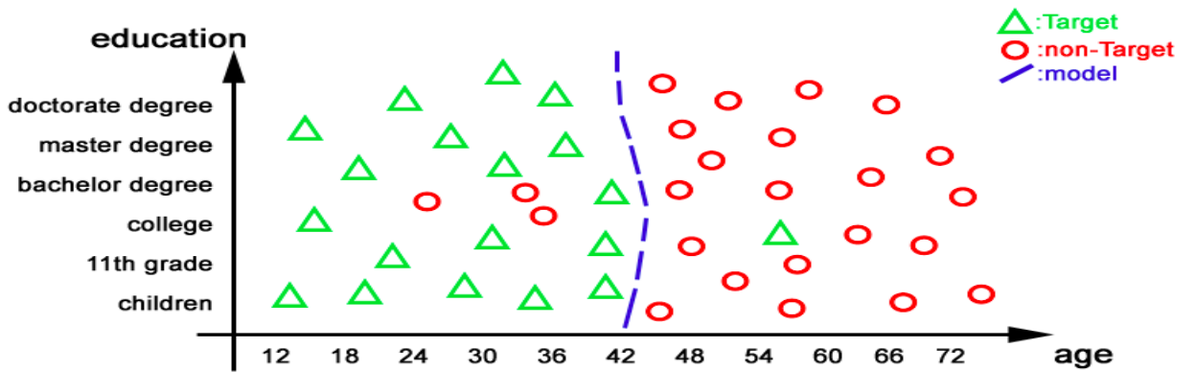
Classifying Iris Plants

- İris çiçeklerinin farklı sepal ve petal şekilleri vardır:
  - iris setosaİris
  - Versicolor
  - Iris Virginica
- Her türden çok sayıda örnek gösterildiğini varsayalım.
- Yeni bir iris çiçeği verildiğinde, türü nedir?

1. CLASSIFICATION TASKS

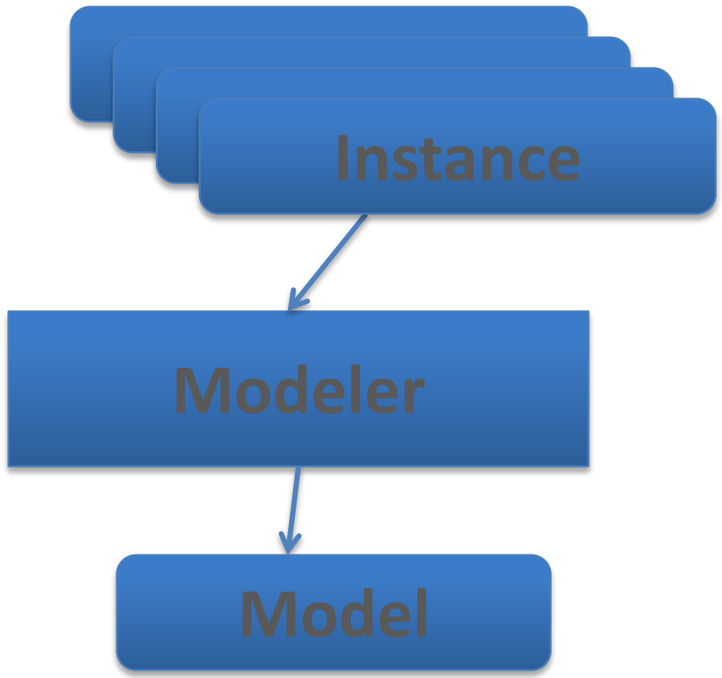
Classification Tasks

- Verilen:
  - Bir dizi sınıf
  - Her sınıfın örnekleri (örnekler)
- Üretmek: Yeni bir örnek verildiğinde sınıfını belirleyeceği bir yöntem (diğer adıyla model)
- Örnekler, bir dizi özellik veya nitelik ve bunların değerleri olarak tanımlanır.
- Örneğin ait olduğu sınıfa “etiket” adı da verilir.
- Giriş, "etiketli örnekler" kümesidir

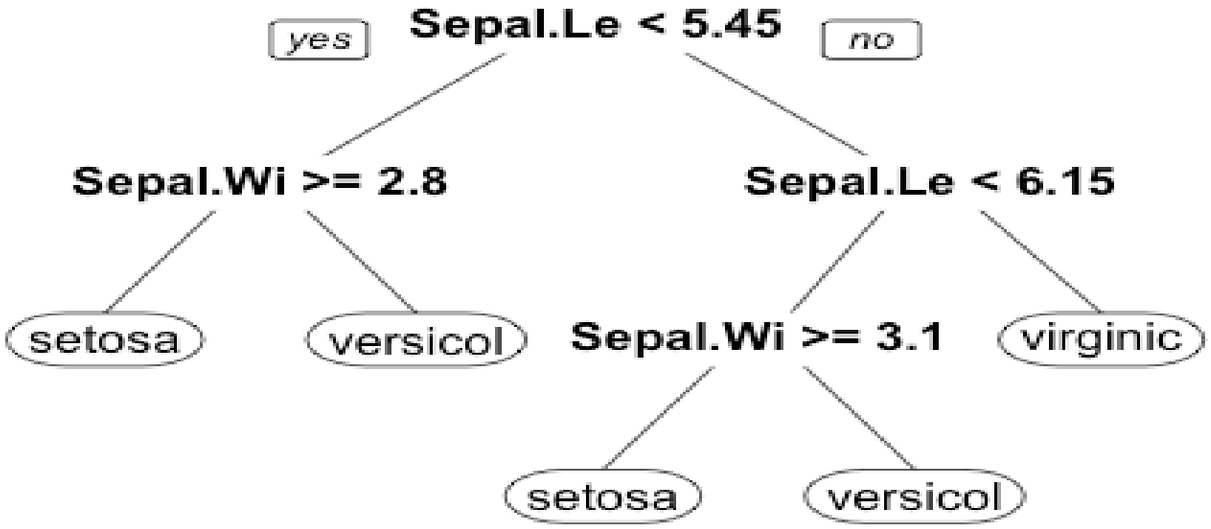


Classification Tasks

- Verilen: Bir dizi etiketli örnek
- Generate: Yeni bir örnek verildiğinde sınıfını varsayacağı bir yöntem (diğer adıyla model)



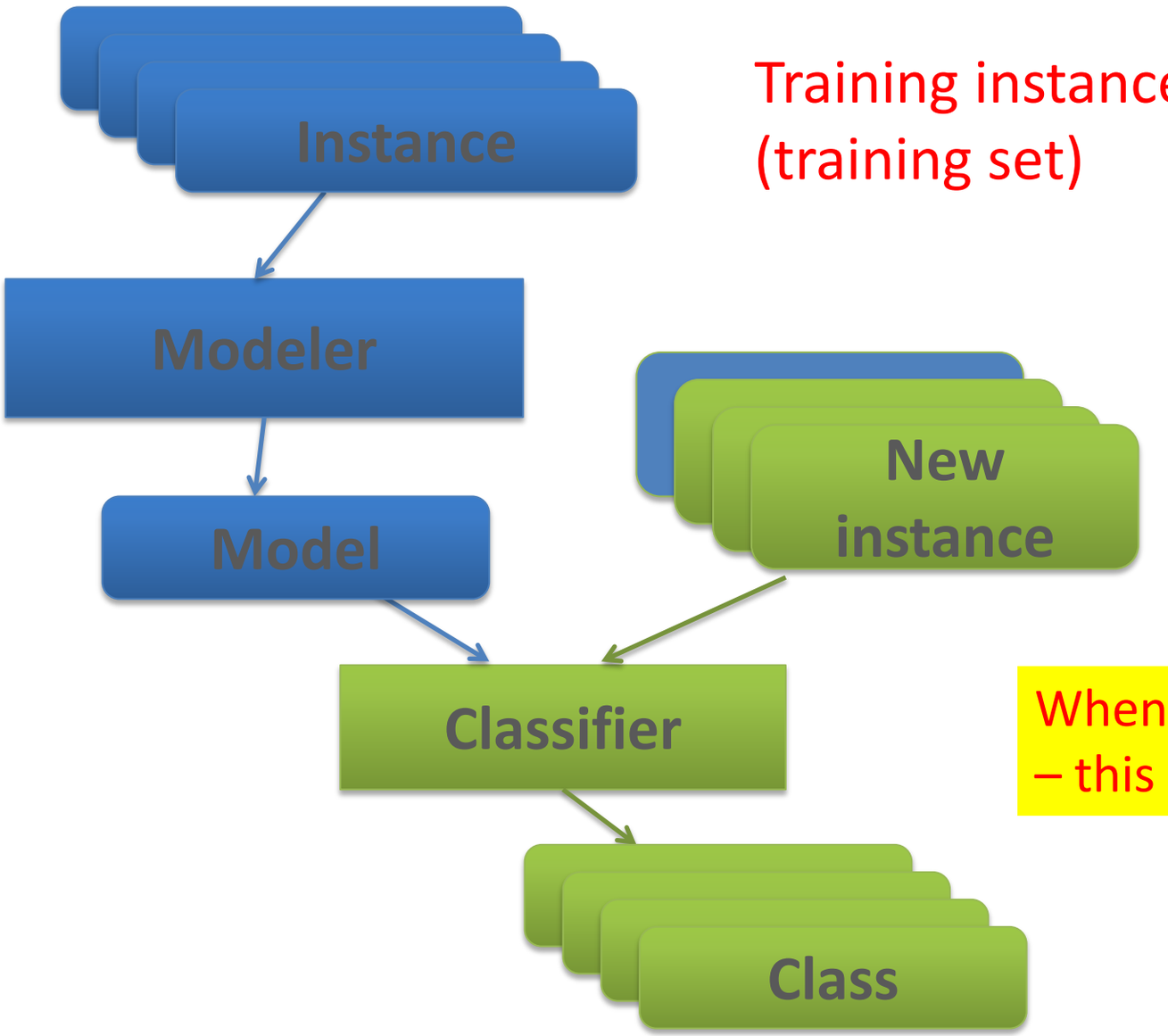
Using a Decision Tree



- Kök düğümdeki tüm örneklerin kümesiyle başlayın
- Kümeyi en iyi bölen özniteliği seçin ve alt düğümler oluşturun
- Örneğin, alt kümelere daha eşit şekilde
- Bir düğüm aynı sınıfta tüm örneklere sahip olduğunda, onu bir yaprak düğüm yapın
- Tüm düğümler ayrılana kadar yineleyin

- Düğümler: nitelik tabanlı kararlar
- Dallar: niteliklerin alternatif değerleri
- Yapraklar: her yaprak bir sınıftırYeni bir örnek verildiğinde, özniteliklerine göre ağaçta bir yol alın
- Bir yaprağa ulaşıldığında, örneğe atanan sınıf budur.

Bulaşma



Training instances
(training set)

Test instances
(test set)

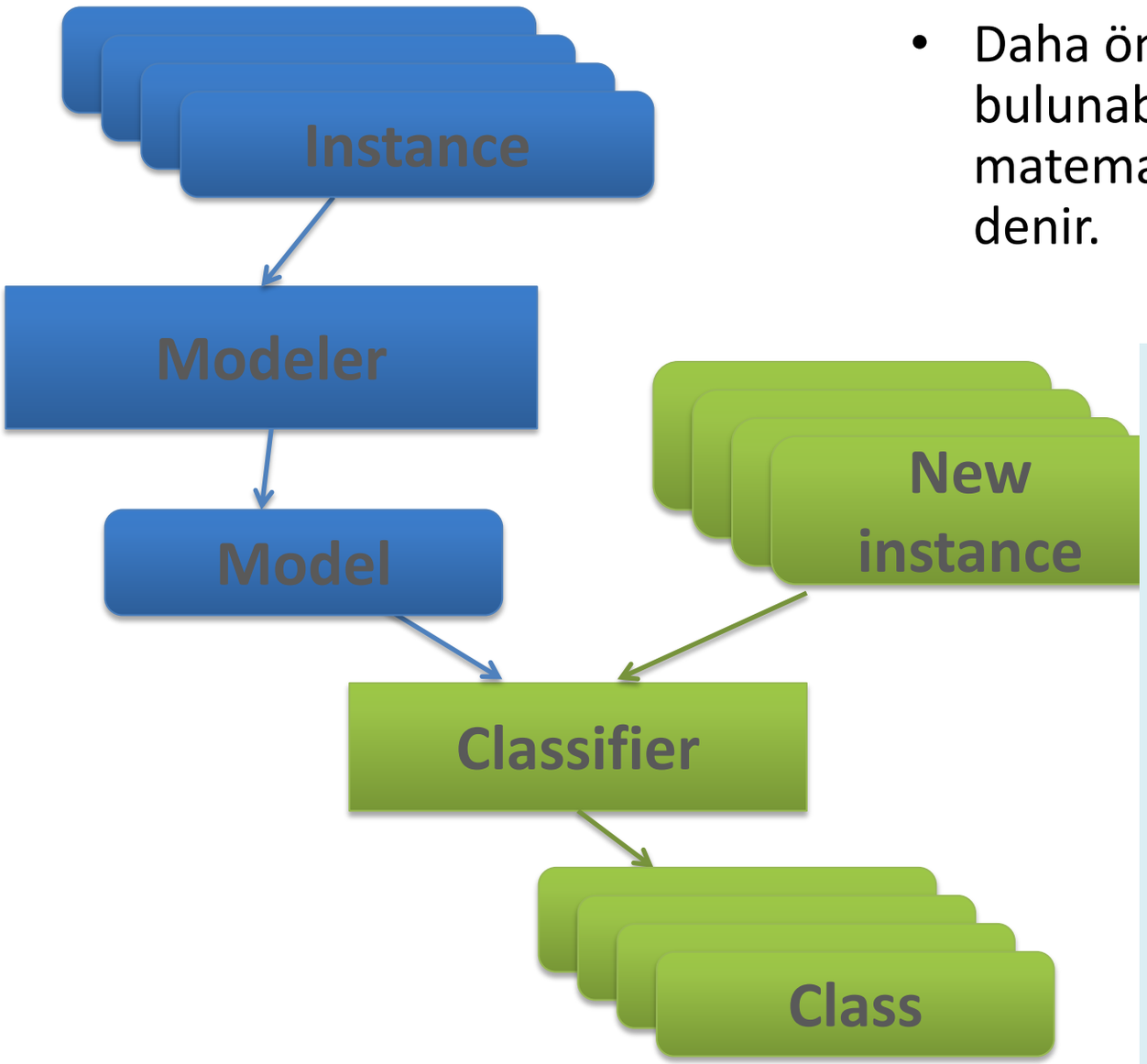
When training and test sets overlap
– this should NEVER happen

About Classification Tasks

- Sınıflar ayrık olmalıdır, yani her örnek yalnızca bir sınıfa aittir
- Yalnızca iki sınıf varsa, sınıflandırma görevleri “ikili”dir.
- Sınıflandırma yöntemi nadiren mükemmel olacaktır, yeni örneklerin sınıflandırılmasında hatalar yapacaktır.

2. BUILDING A CLASSIFIER

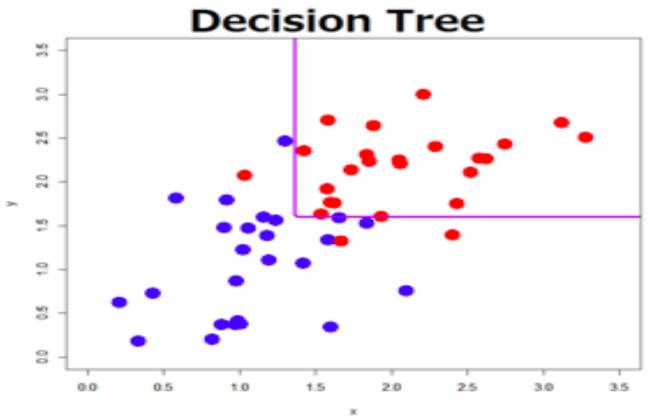
What is a Modeler?

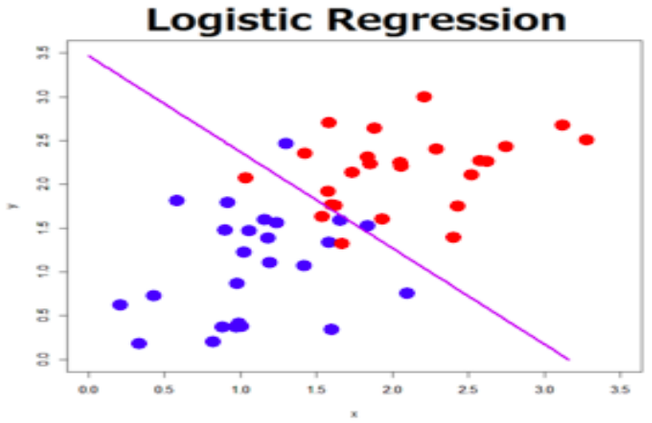


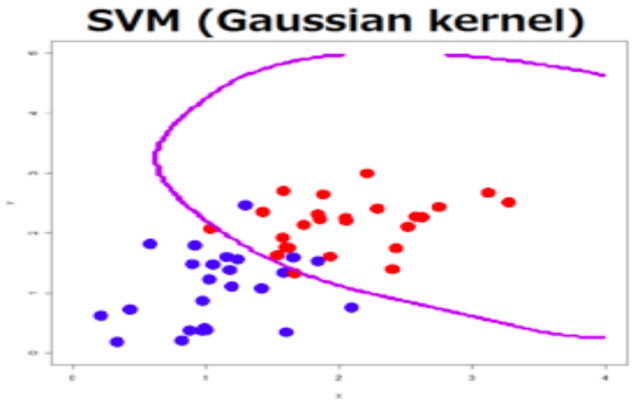
- Daha önce görmediği örnekler hakkında tahminlerde bulunabilmesi için örneklerden genellemeye yönelik matematiksel/algoritmik bir yaklaşım çıktısına model denir.

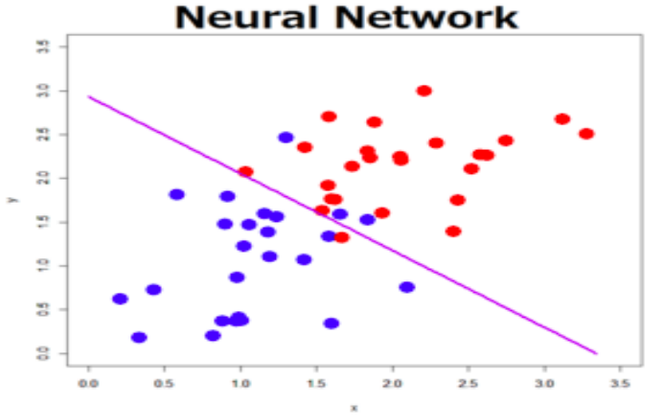
Types of Modelers/Models:

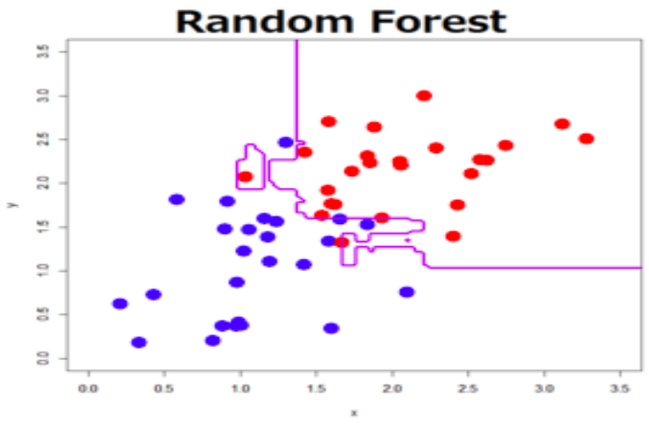
- Logistic regression
- Naïve Bayes classifiers
- Support vector machines (SVMs)
- Decision trees
- Random forests
- Kernel methods
- Genetic algorithms
- Neural networks

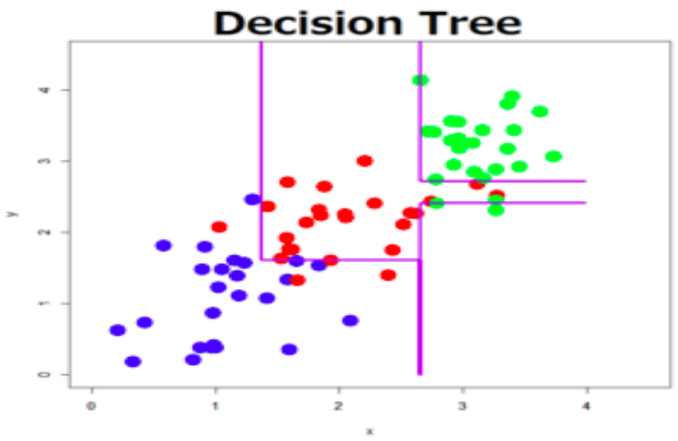

Decision Tree

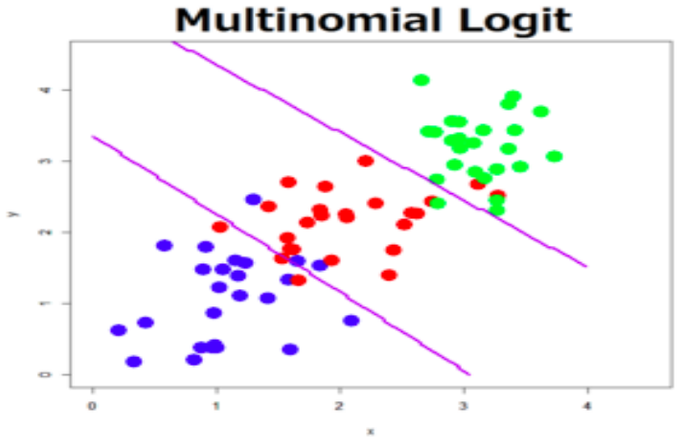

Logistic Regression

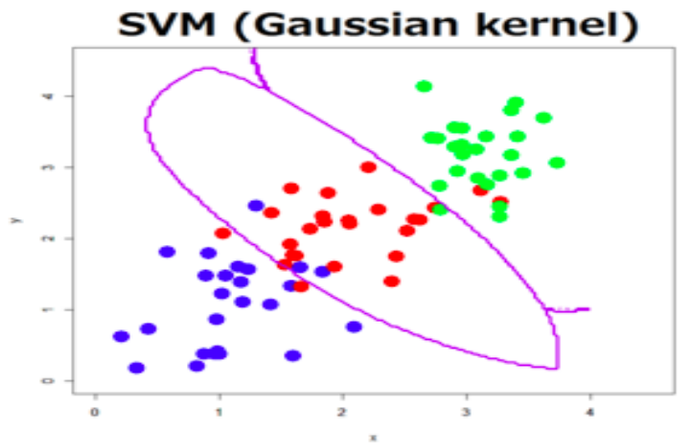

SVM (Gaussian kernel)

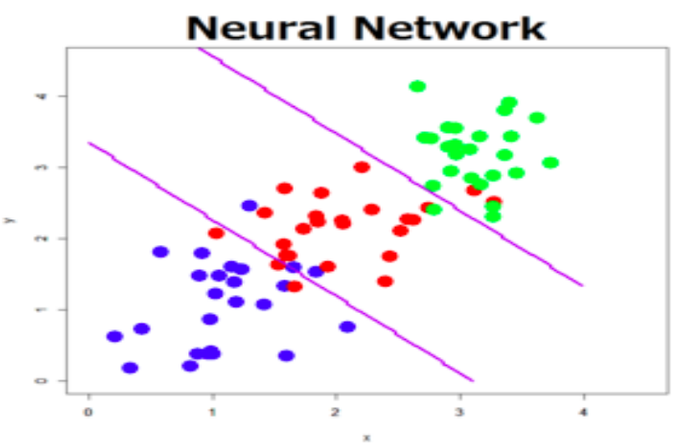

Neural Network

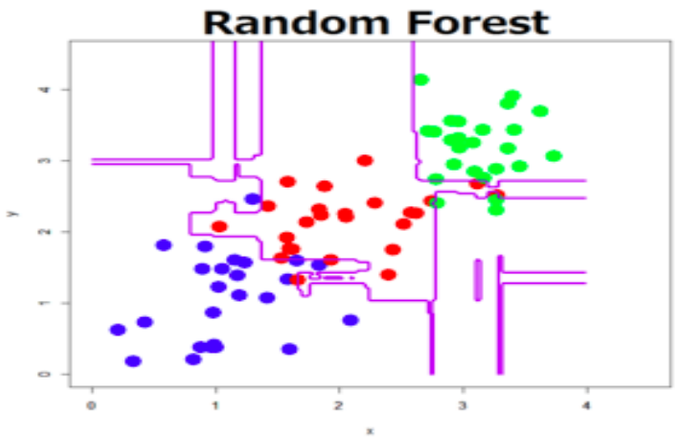

Random Forest


Decision Tree
x
y

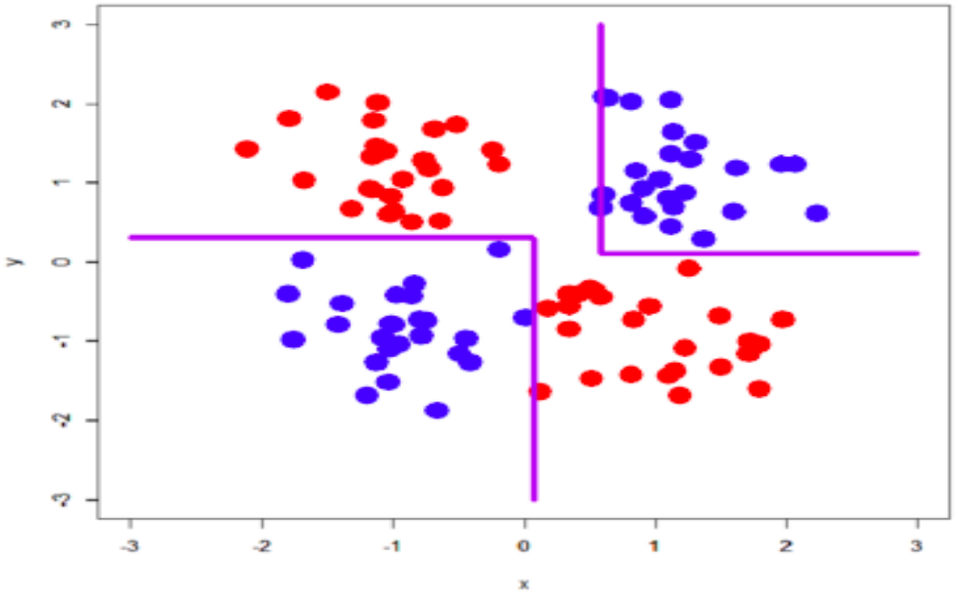

Multinomial Logit
x
y


SVM (Gaussian kernel)
x
y

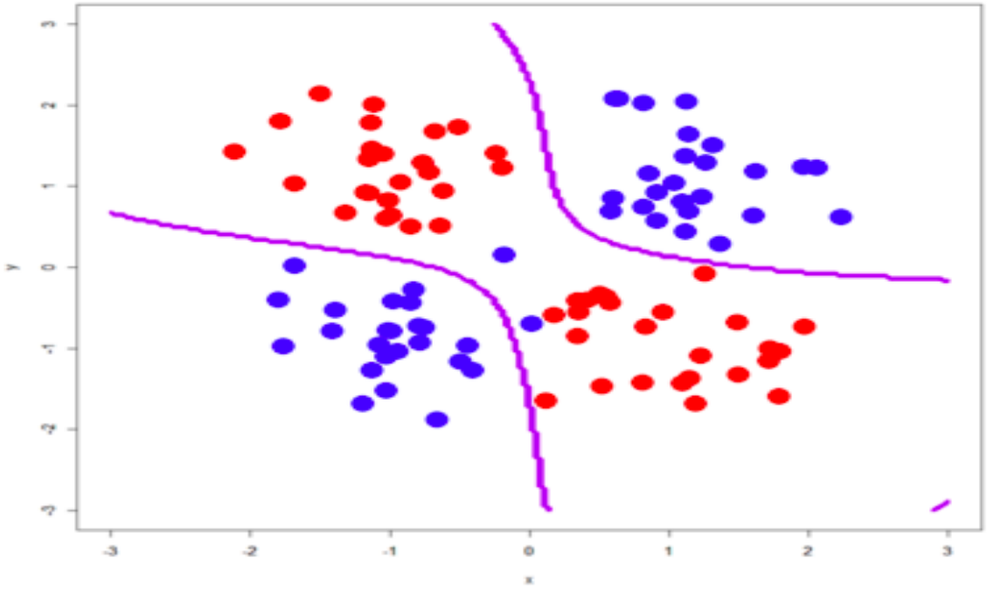

Neural Network
x
y


Random Forest
x
y

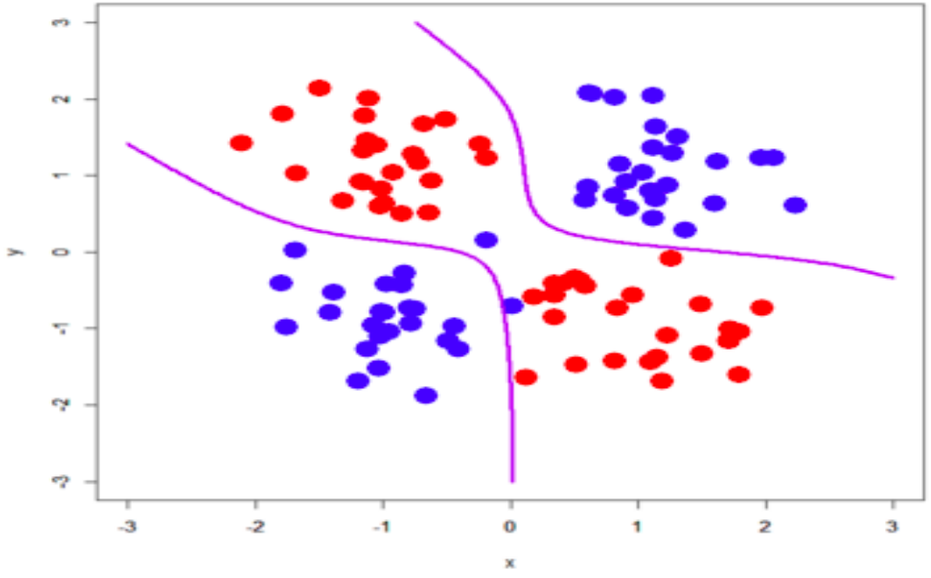
Decision Tree



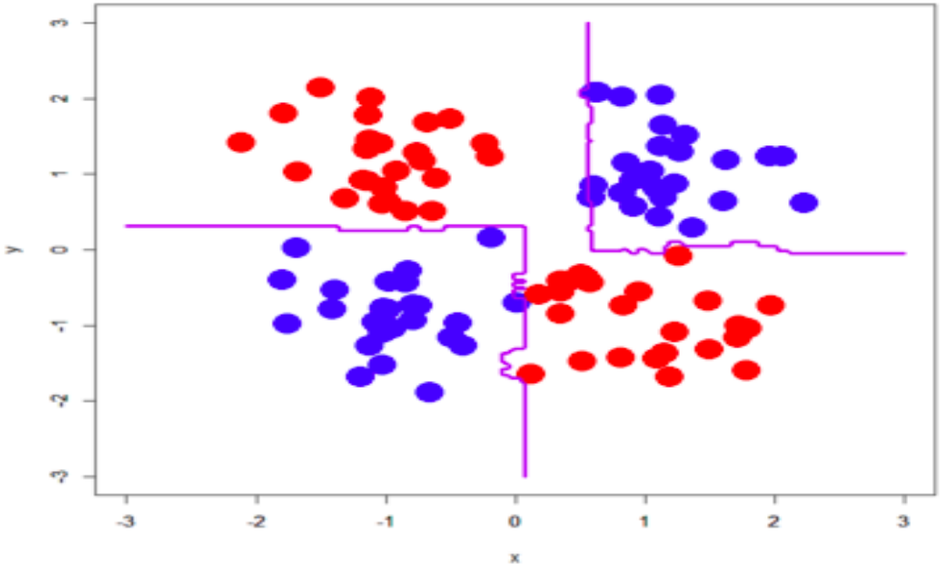
SVM (Gaussian kernel)

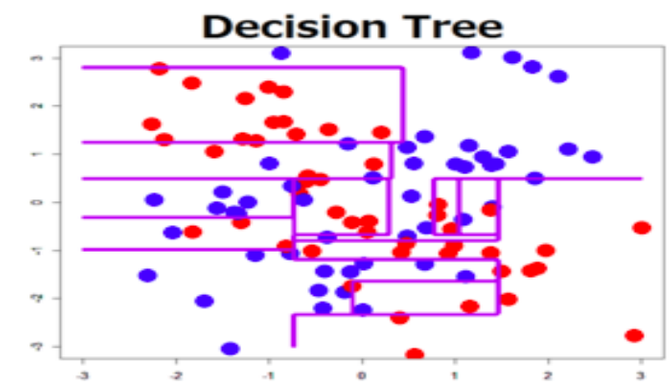


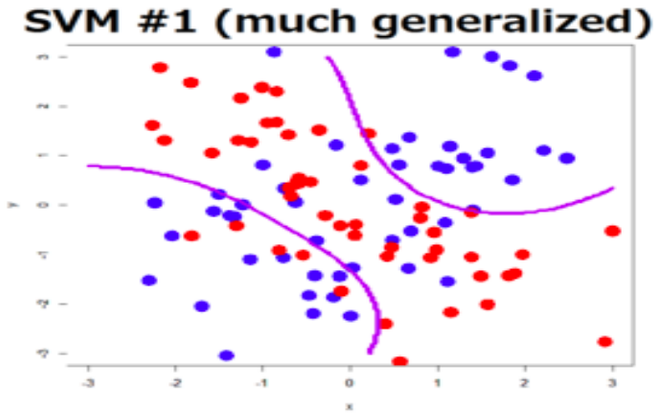
Neural Network

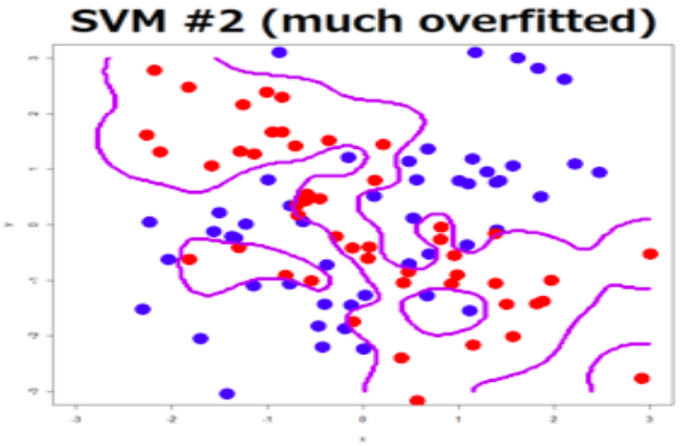


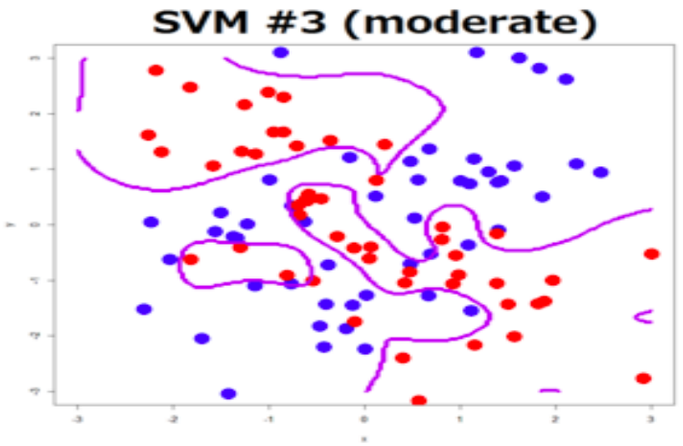
Random Forest

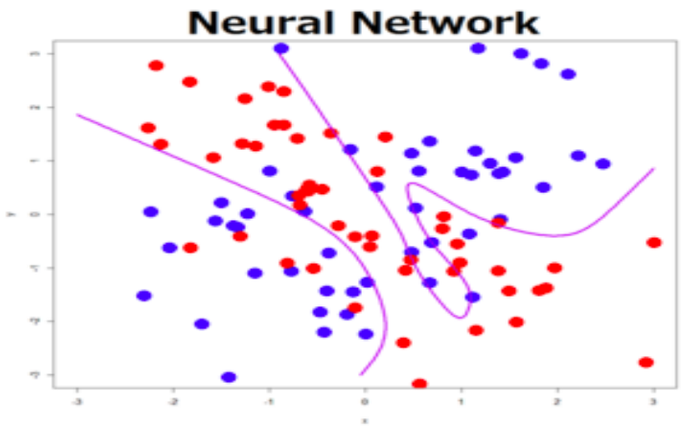


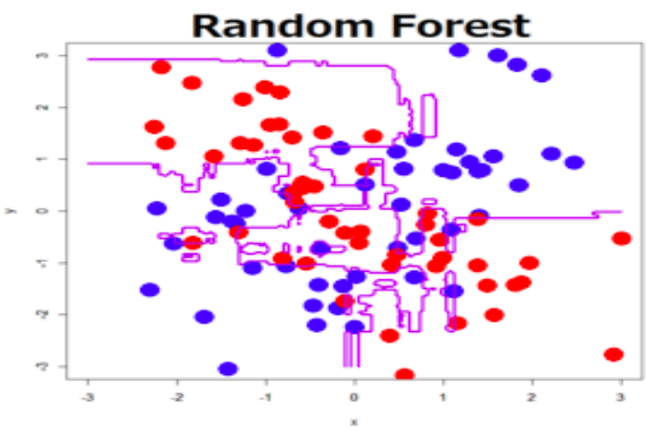

Decision Tree

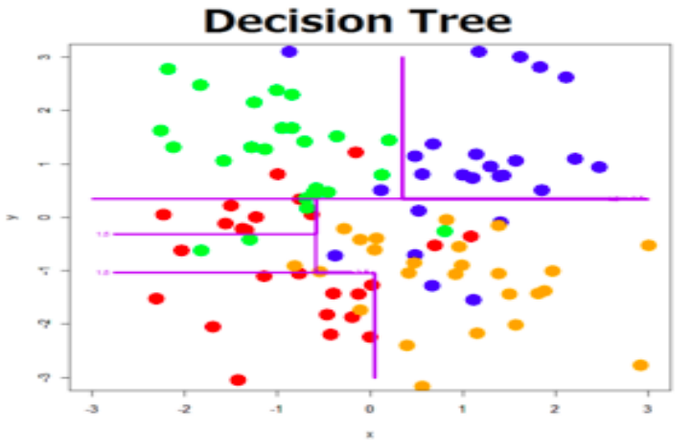

SVM #1 (much generalized)

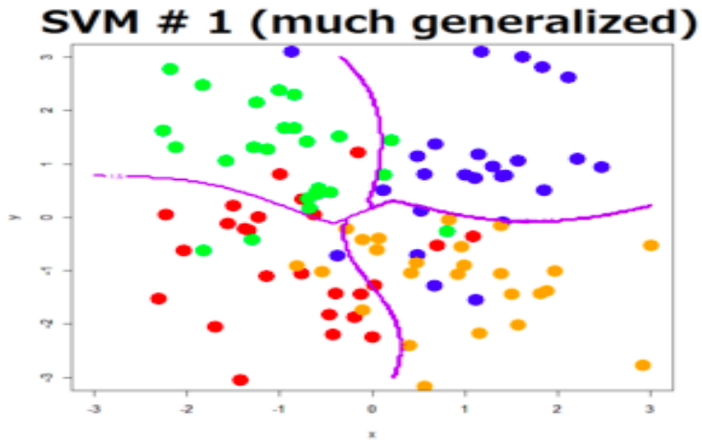

SVM #2 (much overfitted)

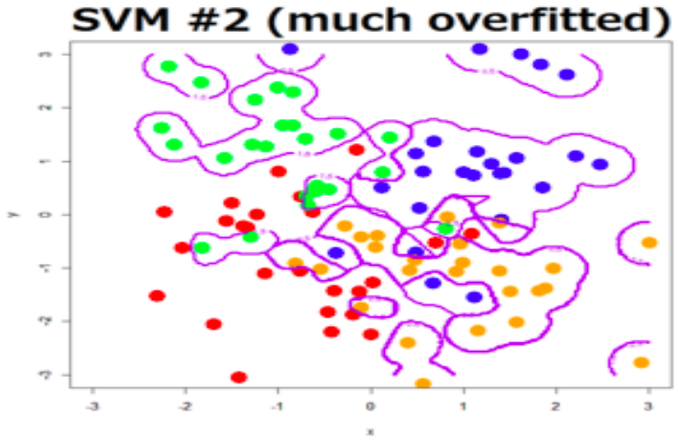

SVM #3 (moderate)

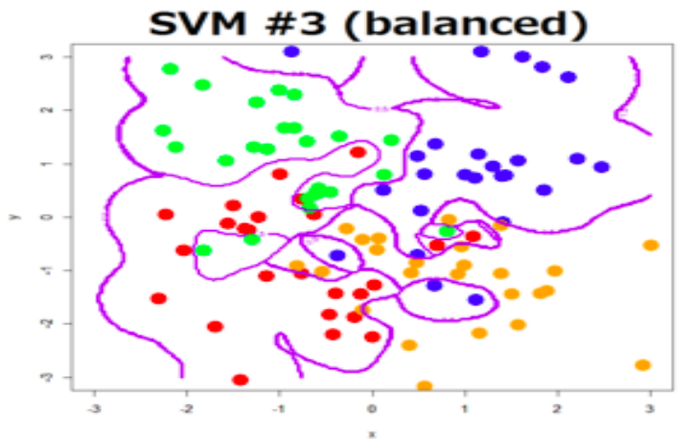

Neural Network

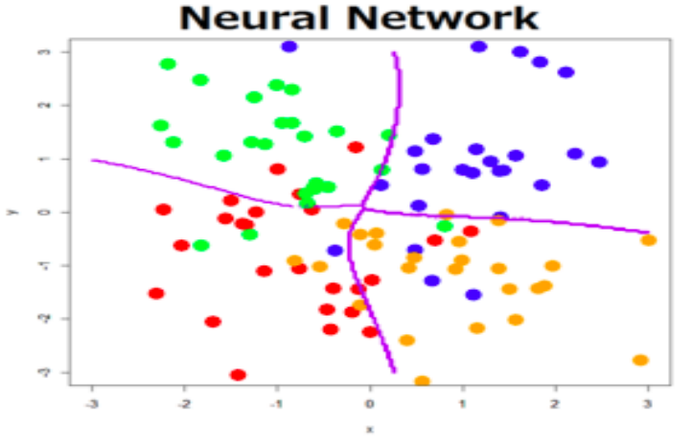

Random Forest

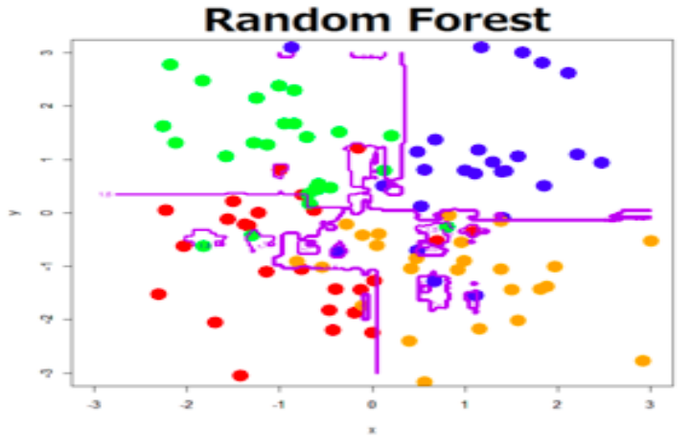

Decision Tree


SVM # 1 (much generalized)


SVM #2 (much overfitted)


SVM #3 (balanced)


Neural Network

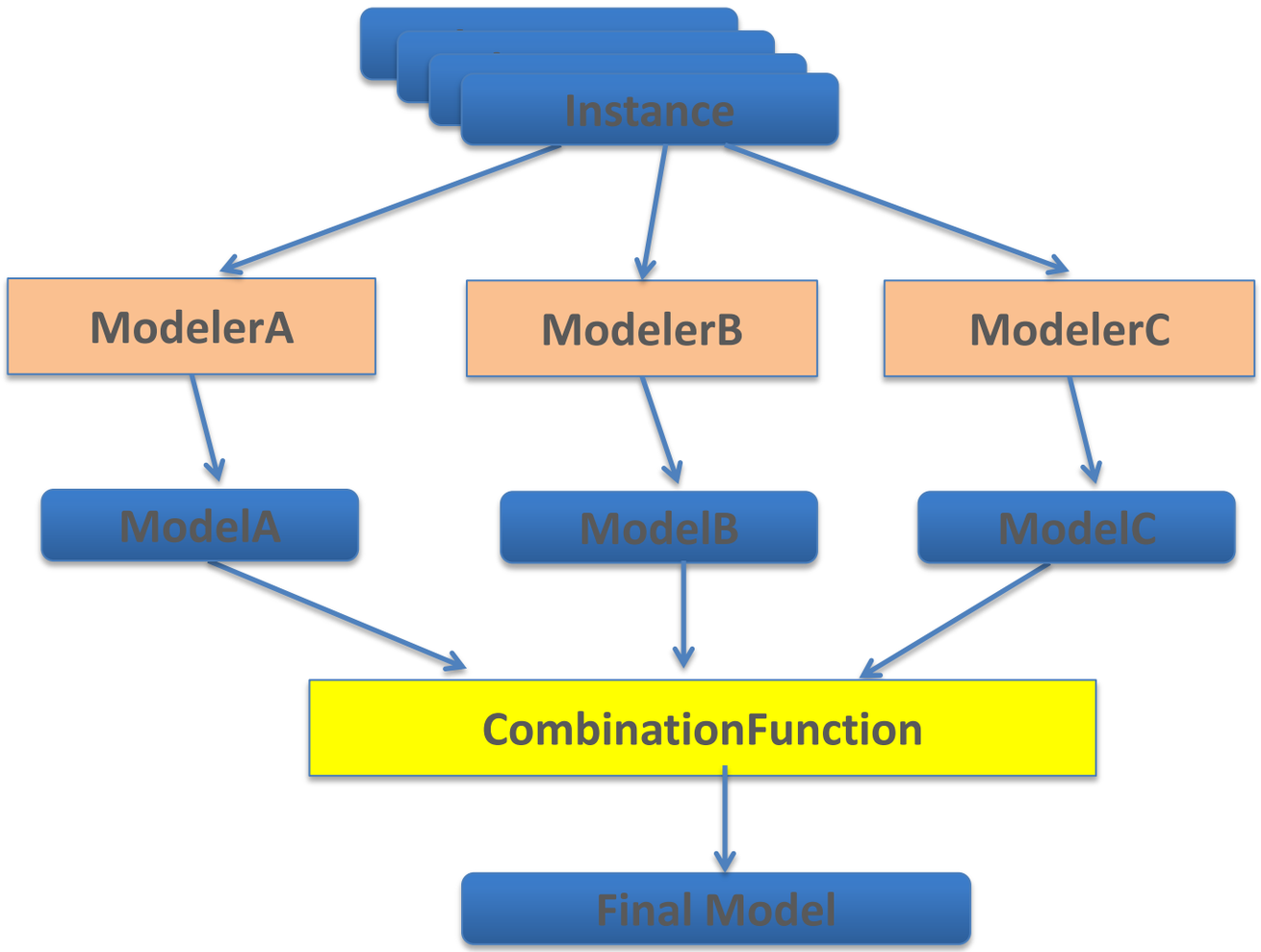

Random Forest

What Modeler to Choose?

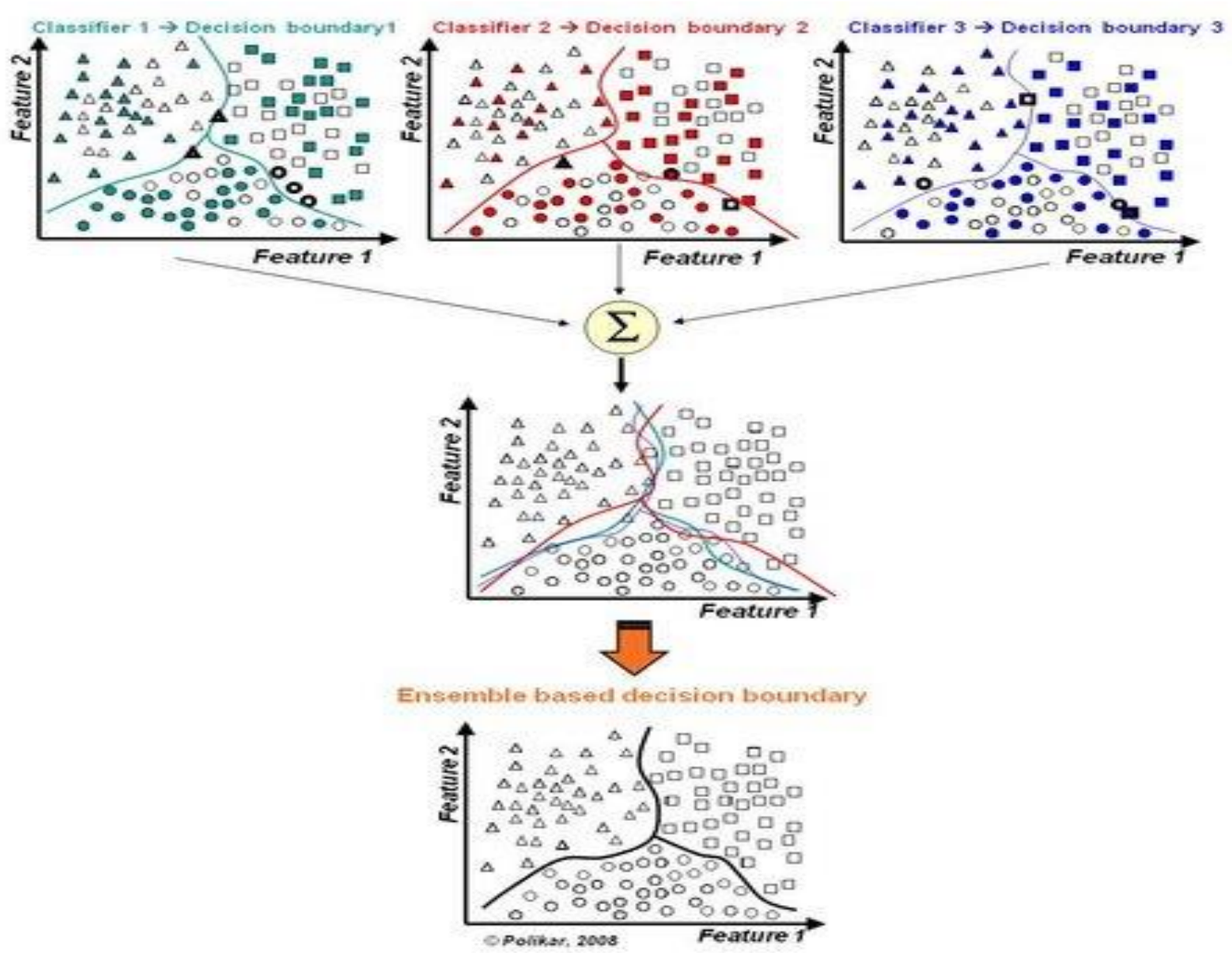
- Logistic regression
- Naïve Bayes classifiers
- Support vector machines (SVMs)
- Decision trees
- Random forests
- Kernel methods
- Genetic algorithms (GAs)
- Neural networks: perceptrons

- Veri bilimcileri, farklı parametrelere sahip farklı modelleyiciler dener ve eldeki veriler için hangisinin en iyi sonucu verdiğini bulmak için doğruluğu kontrol eder.

Topluluklar (Ensembles)



- Bir topluluk yöntemi, aynı görevi yapan birkaç algoritma kullanır ve sonuçlarını birleştirir.“
  - Topluluk öğrenimi”
  - Bir kombinasyon işlevi sonuçları birleştirir
- Çoğunluk oyu: her algoritma bir oy alır
  - Ağırlıklı oylama: her algoritmanın oyu bir ağırlığa sahiptir
  - Diğer karmaşık kombinasyon fonksiyonları


Classifier 1 → Decision boundary1
Classifier 2 → Decision boundary 2
Classifier 3 → Decision boundary 3
Feature 2
Feature 1
Σ
Ensemble based decision boundary
© Polikar, 2008

3. EVALUATING A CLASSIFIER
Bir Sınıflandırıcıyı Değerlendirmek

Classification Accuracy

- Doğruluk: doğru sınıflandırmaların yüzdesi

$$\text{Accuracy} = \frac{\text{Doğru sınıflandırılmış toplam test örneği}}{\text{Toplam test örneği sayısı}}$$

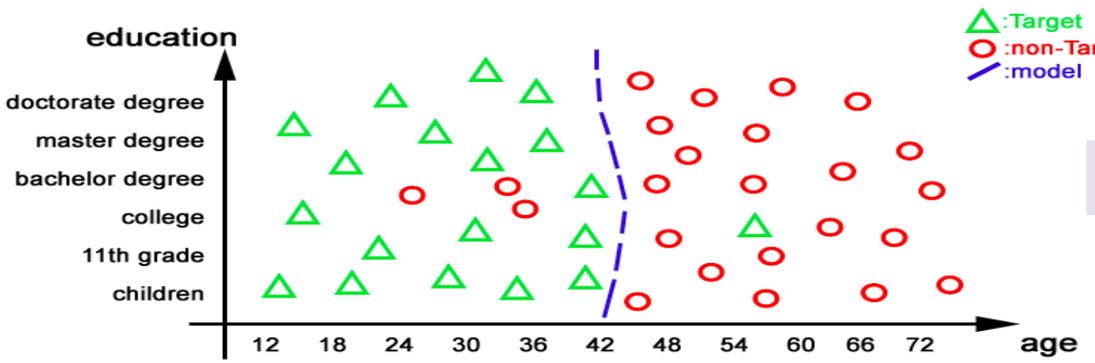
Evaluating a Classifier: What Affects the Performance

- Görevin karmaşıklığı
  - Çok miktarda özellik (yüksek boyutluluk)
  - Özellik(ler) çok az kez görünüyor (seyrek veri)
- Karmaşık bir sınıflandırma görevi için birkaç örnek
- Örnekler için eksik özellik değerleri
- Örnekler için öznitelik değerlerindeki hatalar
- Eğitim örneklerinin etiketlerindeki hatalar
- Sınıflarda örneklerin eşit olmayan kullanılabilirliği

Aşırı uyum gösterme

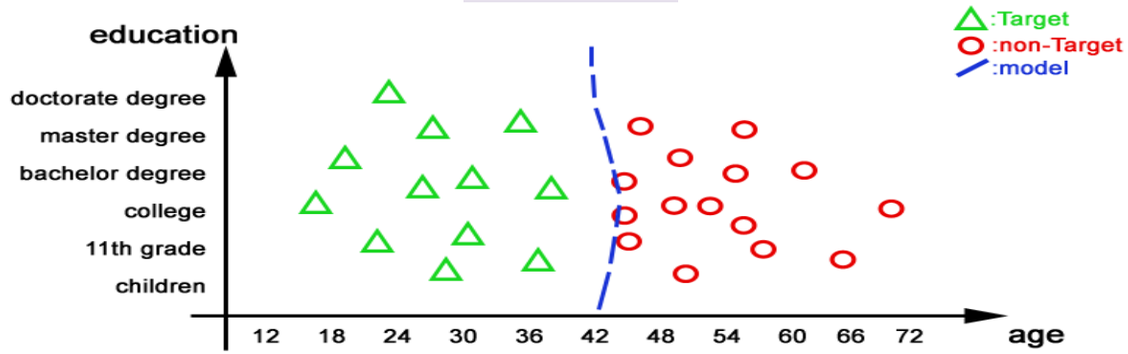
- Bir model, verilerle çok doğru uyulu olduğunda eğitim verilerine fazla uyar ve yeni test verileriyle çok iyi sonuç vermeyebilir.

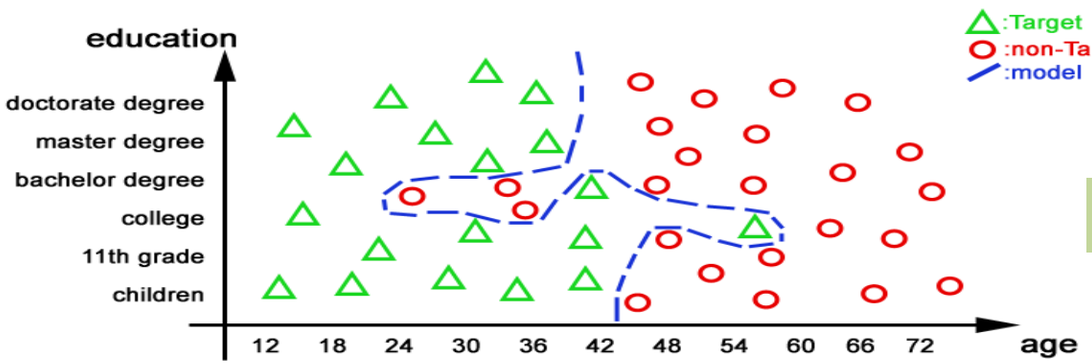
Training Data



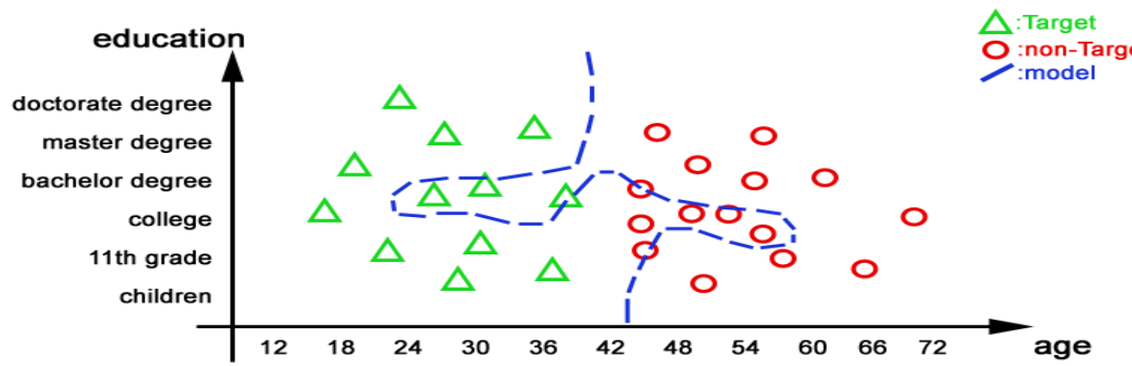
Model 1

Test Data





Model 2



Tümevarım

- Tümevarım, geçmişte görülen örnekler hakkında genel kurallar çıkarmayı gerektirir.
  - Tümdengelimle zıtlık: geçmişte gördüklerimizin mantıklı bir sonucu olan şeyleri çıkarsama
- Sınıflandırıcılar tümevarım kullanır: hedef sınıflar hakkında genel kurallar oluştururlar.
  - Kurallar, yeni veriler hakkında tahminler yapmak için kullanılır
  - Bu tahminler yanlış olabilir

When Facing a Classification Task

- Hangi özellikleri seçmeli
  - Farklı özellikleri tanımlama denenir
  - Bazı problemler için yüzlerce, belki de binlerce özellik mümkün olabilir.
  - Bazen özellikler doğrudan gözlemlenemez (yani “gizli” değişkenler vardır)

- Hangi sınıfları seçilmeli
  - Kabul edilebilir / zehirli mi?
  - Kabul edilebilir / zehirli / bilinmiyor mu?
- Kaç etiketli örnek
  - çok çalışma gerektirebilir
- Hangi modelleyici seçilir
  - Farklı olanları denemek daha iyi

Summary of Major Concepts

- Instances, features, values
- Classes, disjoint classes
- Labels, binary tasks
- Learning
  - Decision trees
  - Modeler
  - Ensembles, combination function
    - Majority vote, weighted vote
- Induction

- Training and test sets
- Evaluation
  - Accuracy, confusion matrix, precision & recall
  - N-fold cross validation
  - Overfitting
- About the data
  - High dimensionality
  - Sparse data
  - Continuous/discrete values
  - Latent variables

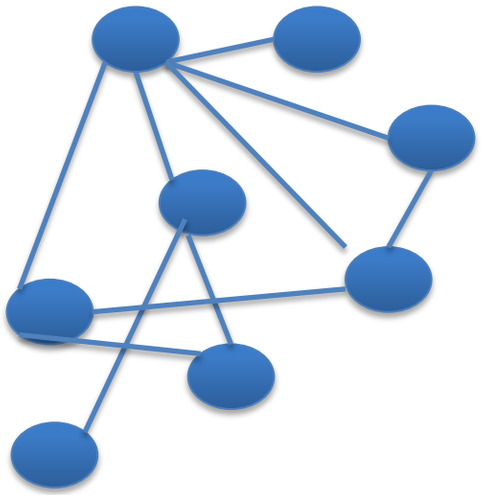
PART III:
Pattern Learning and Clustering

Pattern Learning and Clustering

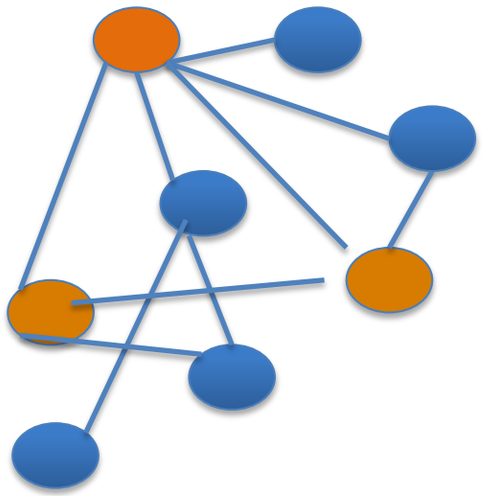
1. Desen algılama
2. Örüntü öğrenme ve desen keşfi
3. Kümeleme

1. PATTERN DETECTION

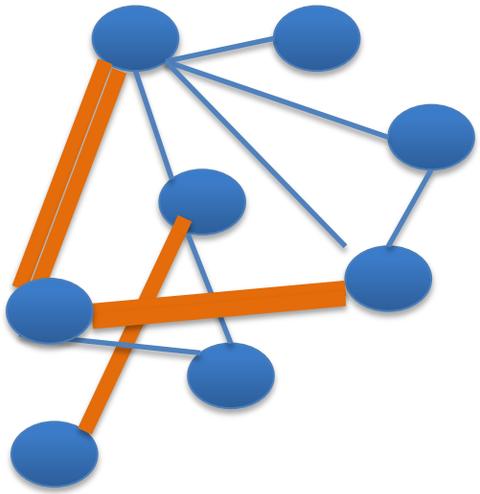
Network Patterns



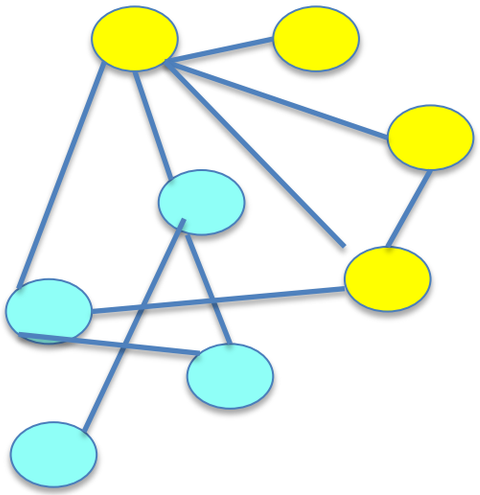
Central entities



Strength of ties

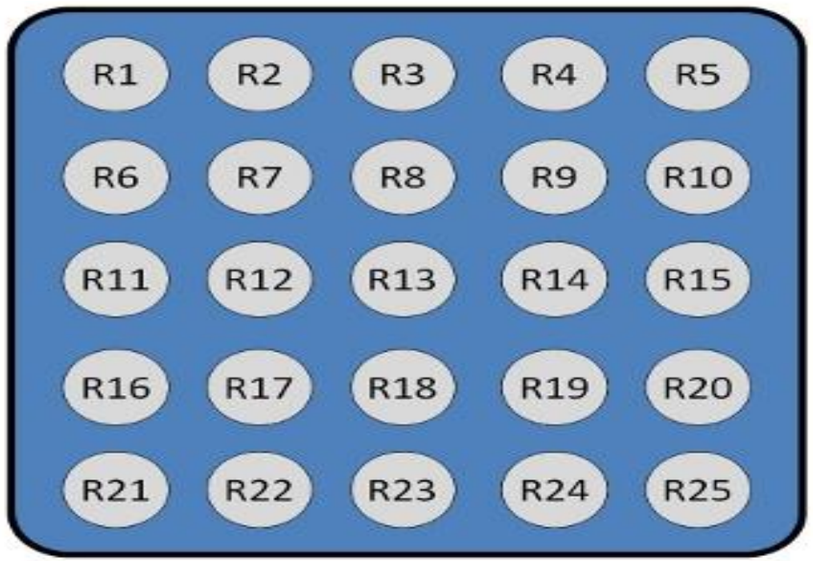


Subgroups

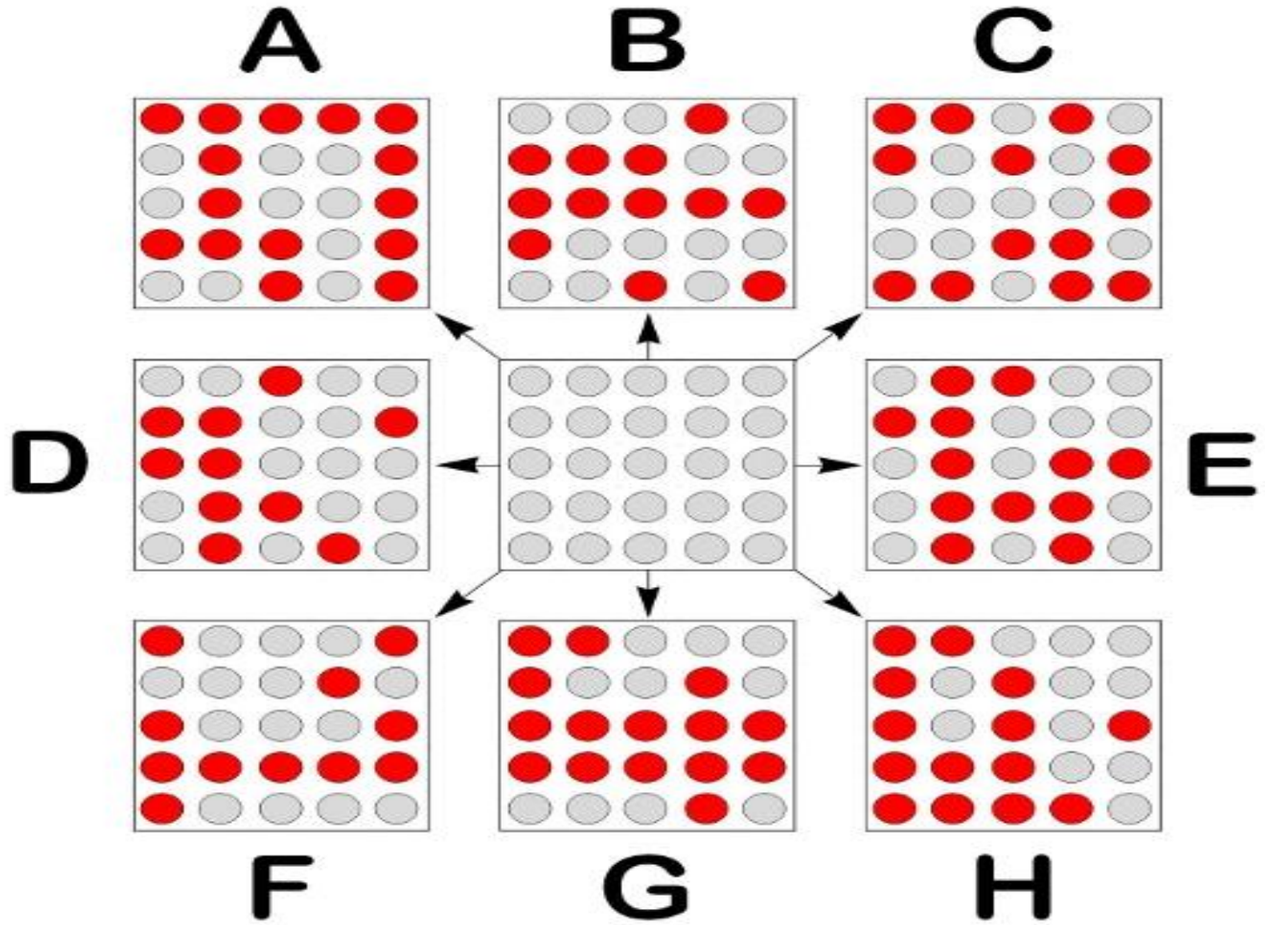


Patterns of activity over time

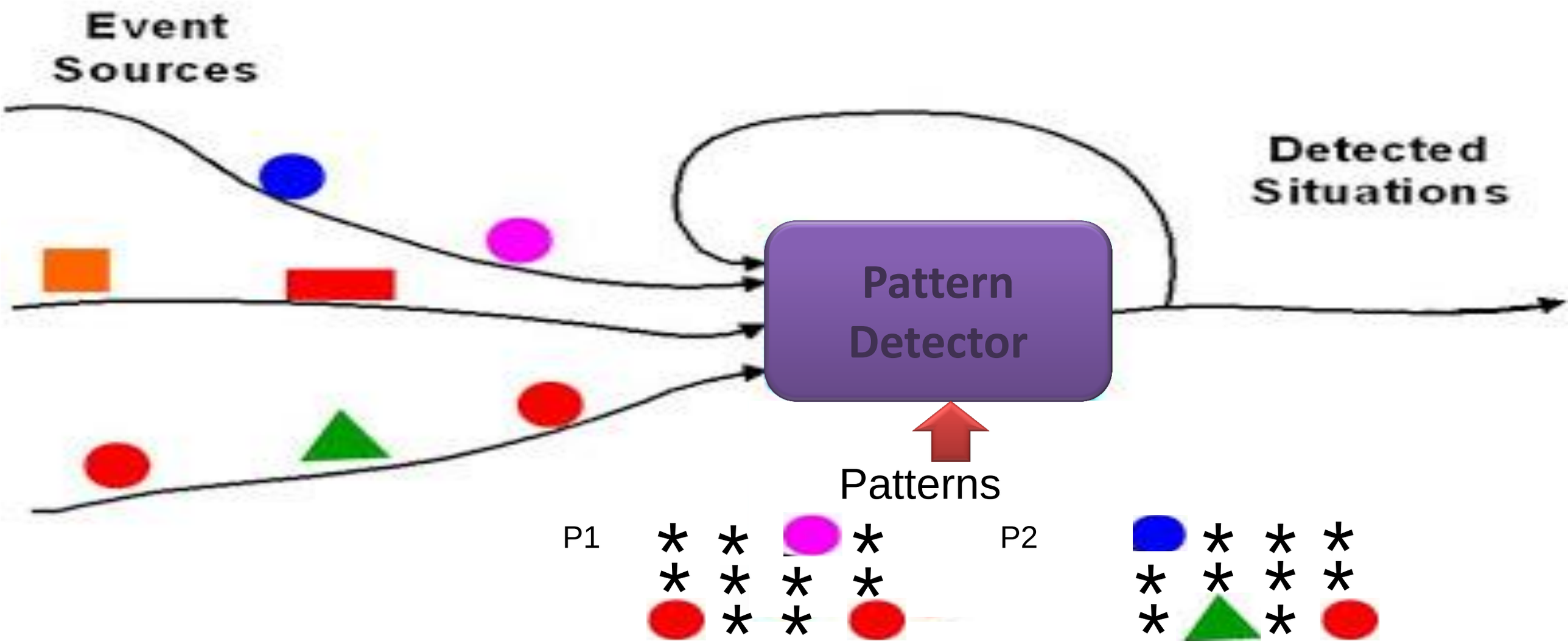
Spatial Patterns



Patterns



Temporal Patterns



Detecting Patterns in a Text String

- abababababab
- abcabcabcabc
- abccccccccabcccabccccccccccccabcabccc

A Pattern Language

- abababababab
  - (ab)*
- abcabcabcabc
  - (abc)*
- abccccccccabcccabccccccccccccabcabccc
  - ((ab)(c)*)*

Detecting Patterns in Streaming Data

- (ab)*x*
  - Abababthsrthwababyertueyrtyertheabsgd
- abcabcabcabc
  - abcabcrgkskhgsnrhnabcabcabcabcrjgjsrn

Concept Drift

- Over time, the data source changes and the concepts that were learned in the past have now changed

2. PATTERN LEARNING AND PATTERN DISCOVERY

Pattern Detection vs Pattern Learning

Pattern Detection

- Inputs:
  - Data
  - A set of patterns
- Output:
  - Matches of the patterns to the data

Pattern Learning

- Inputs:
  - Data annotated with a set of patterns
- Output:
  - A set of patterns that appear in the data with some frequency

Pattern Detection vs Pattern Learning

Pattern Learning

- Inputs:
  - Data annotated with a set of patterns
- Output:
  - A set of patterns that appear in the data with some frequency

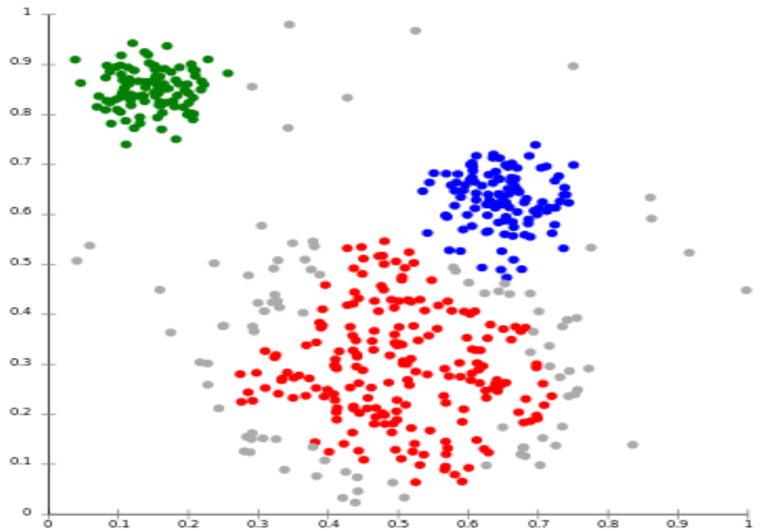
Pattern Discovery

- Inputs:
  - Data
- Output:
  - A set of patterns that appear in the data with some frequency

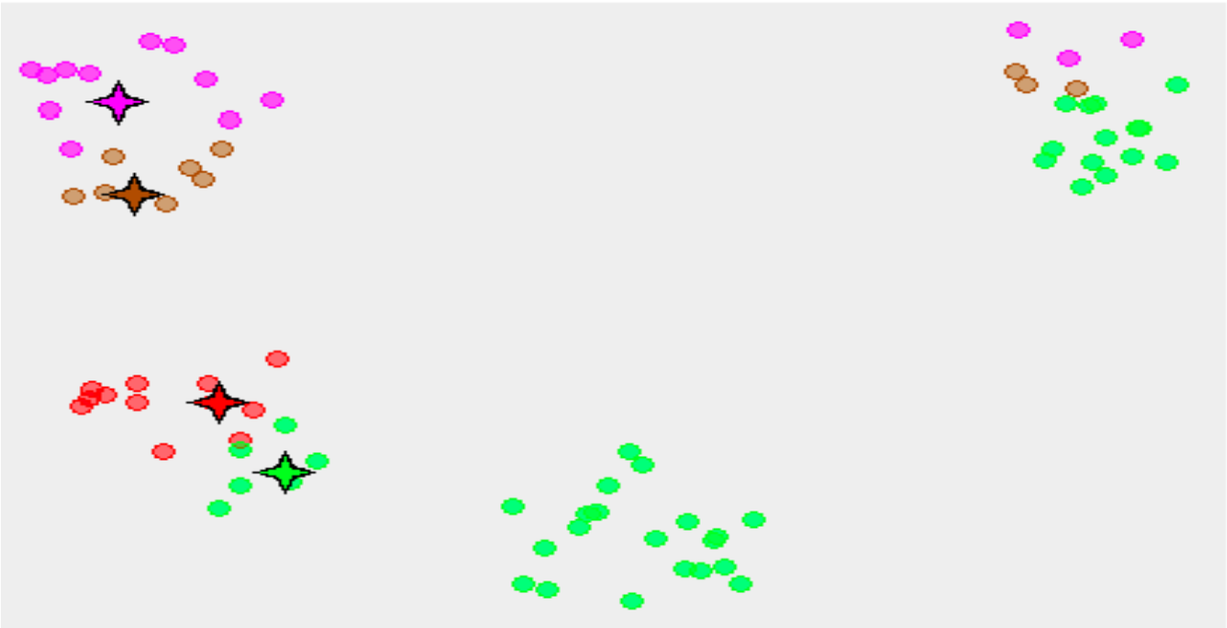
3. CLUSTERING

Clustering

- Find patterns based on features of instances
- Given:
  - A set of instances (datapoints), with feature values
    - Feature vectors
  - A target number of clusters (k)
- Find:
  - The “best” assignment of instances (datapoints) to clusters
    - “Best”: satisfies some optimization criteria
    - “clusters” represent similar instances

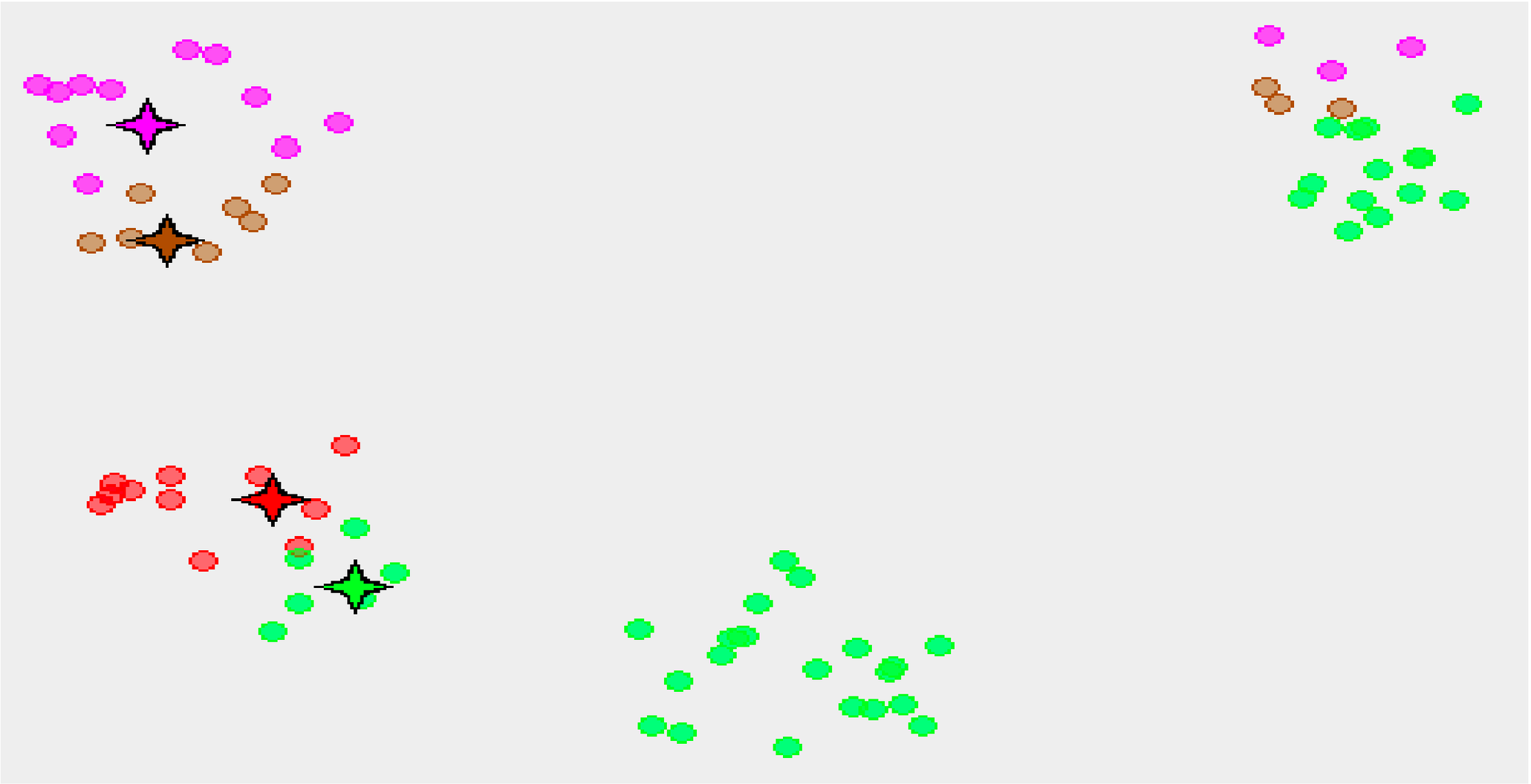


K-Means Clustering Algorithm

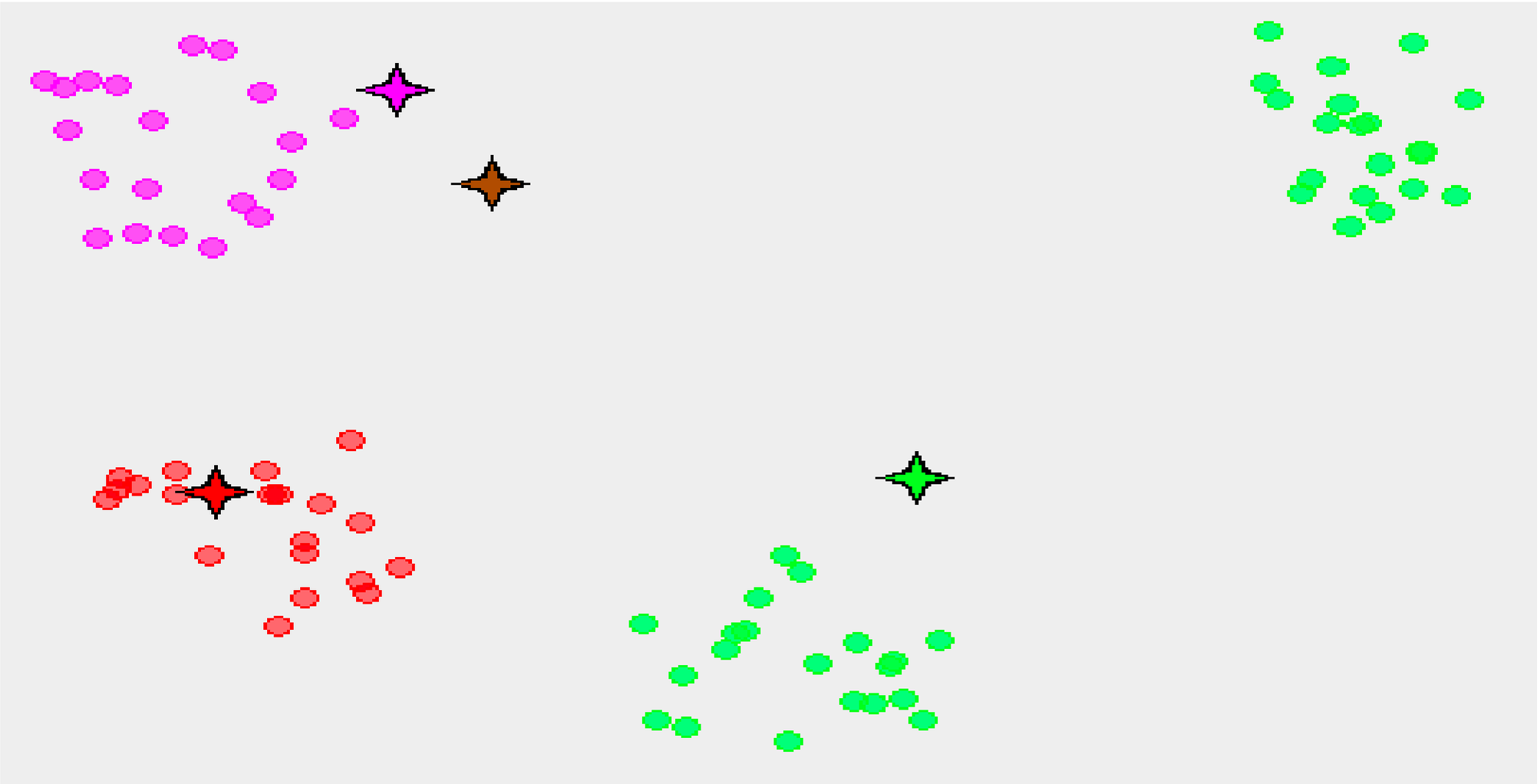


- User specifies a target number of clusters (k)
- Place randomly k cluster centers
- For each datapoint, attach it to the nearest cluster center
- For each center, find the centroid of all the datapoints attached to it
- Turn the centroids into cluster centers
- Repeat until the sum of all the datapoint distances to the cluster centers is minimized

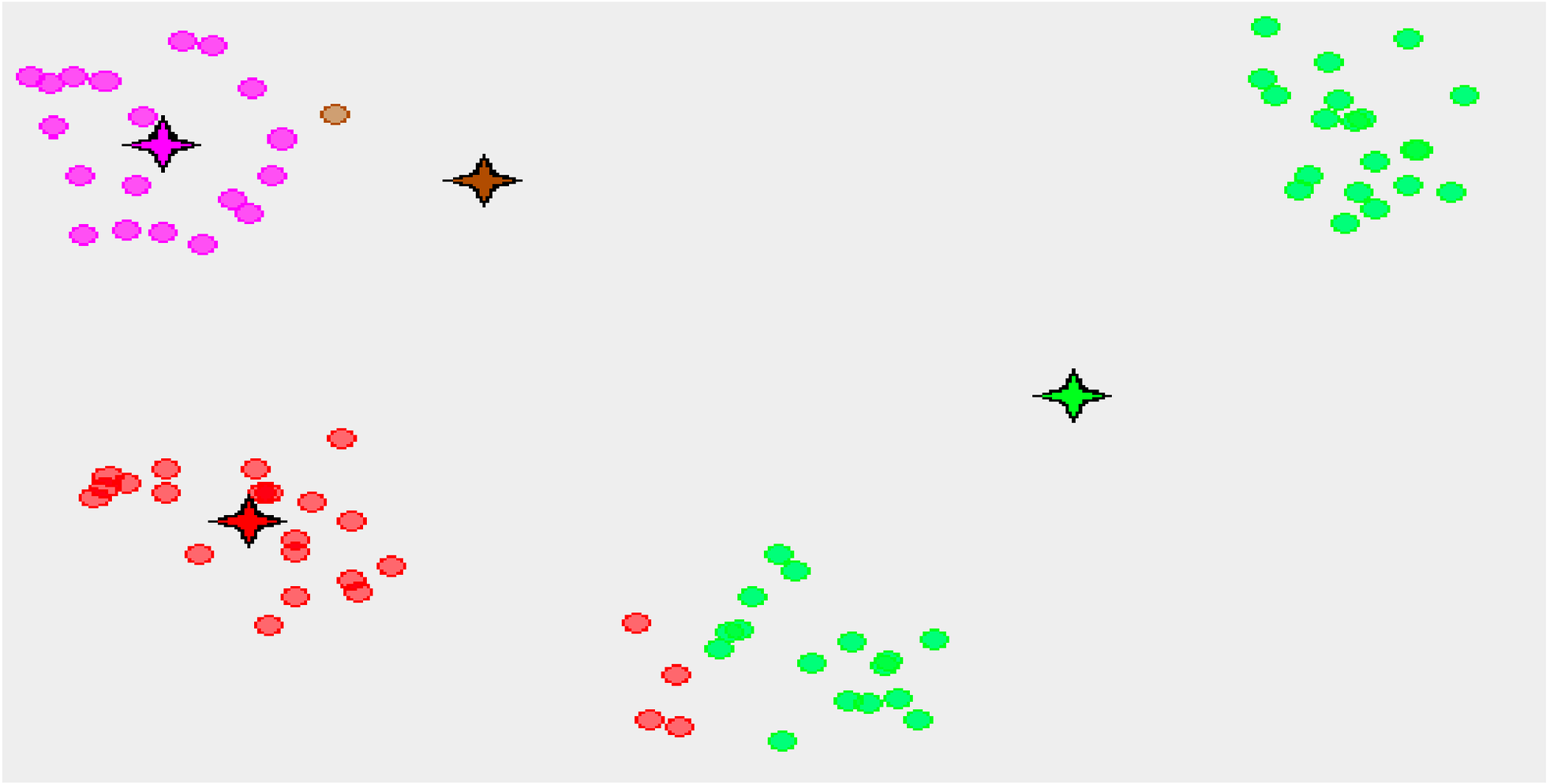
K-Means Clustering (1)



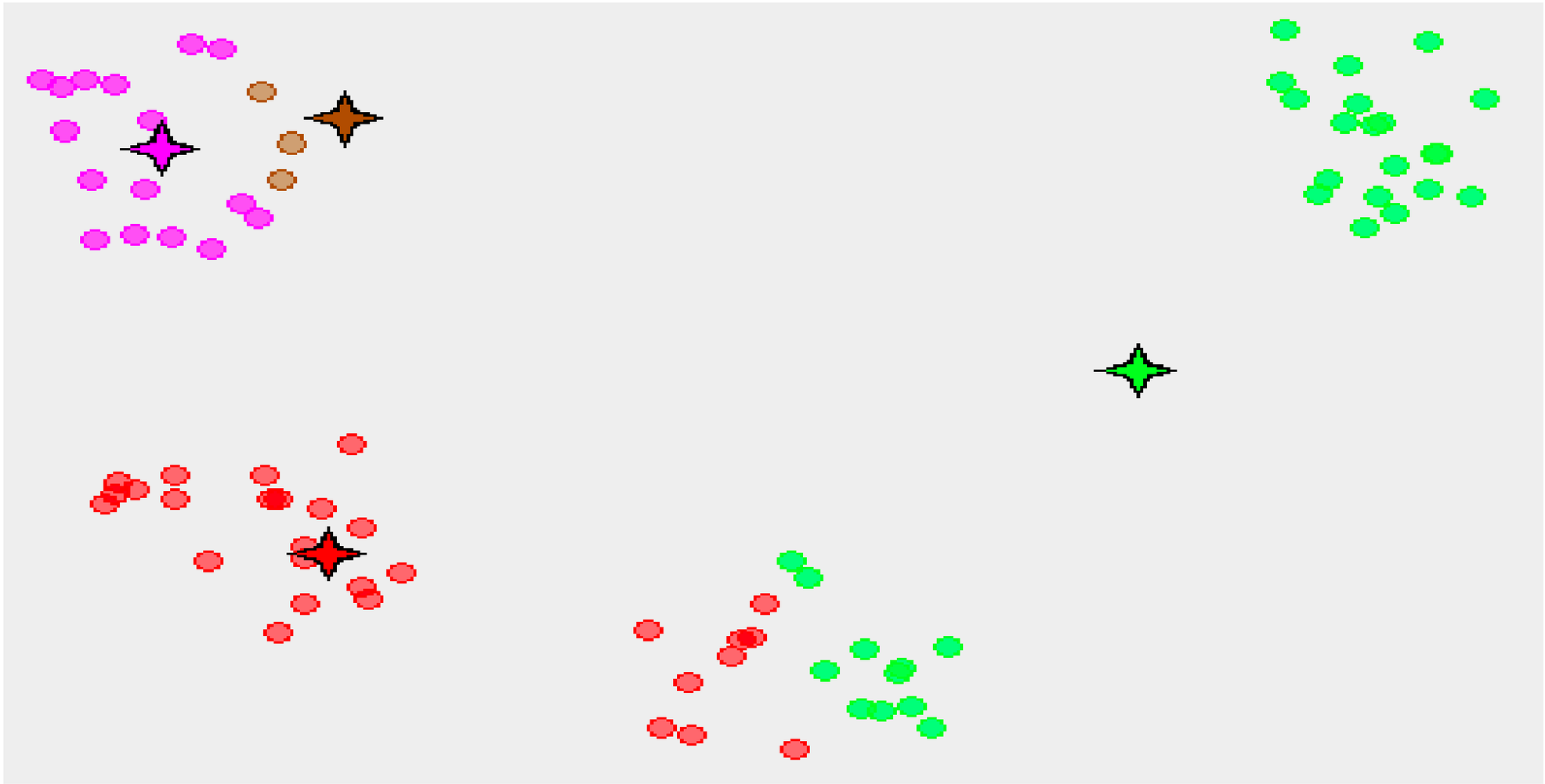
K-Means Clustering (2)



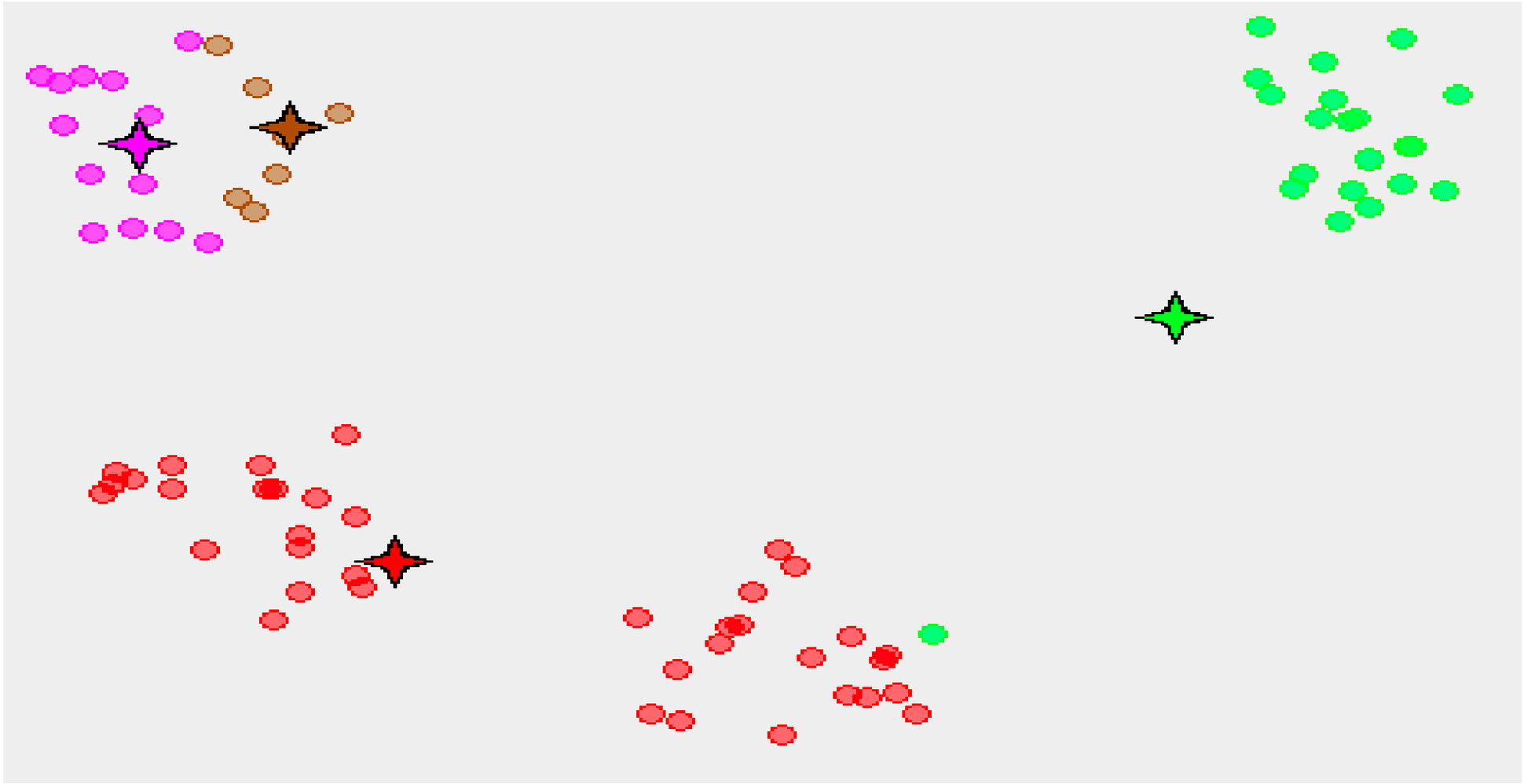
K-Means Clustering (3)



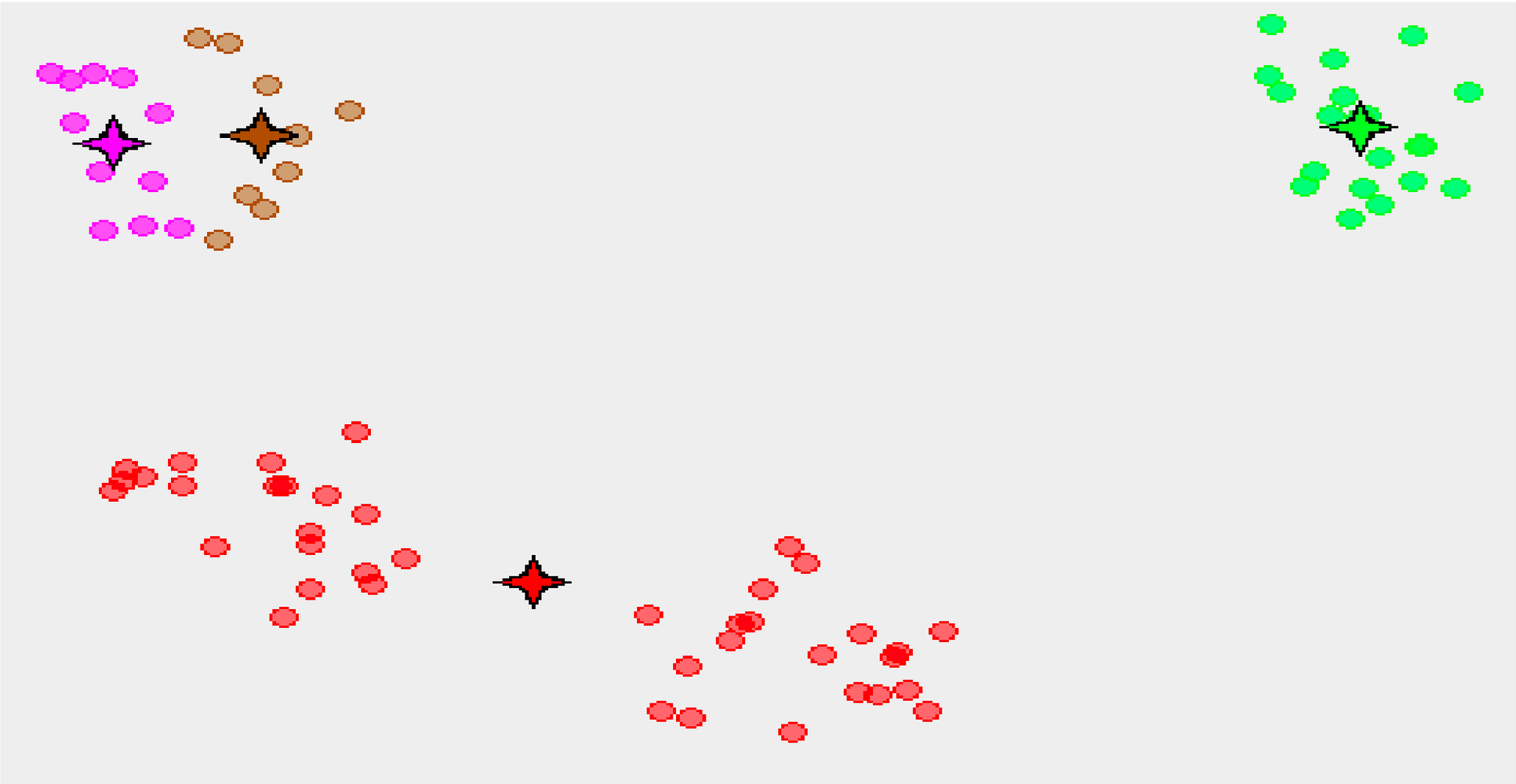
K-Means Clustering (4)



K-Means Clustering (5)

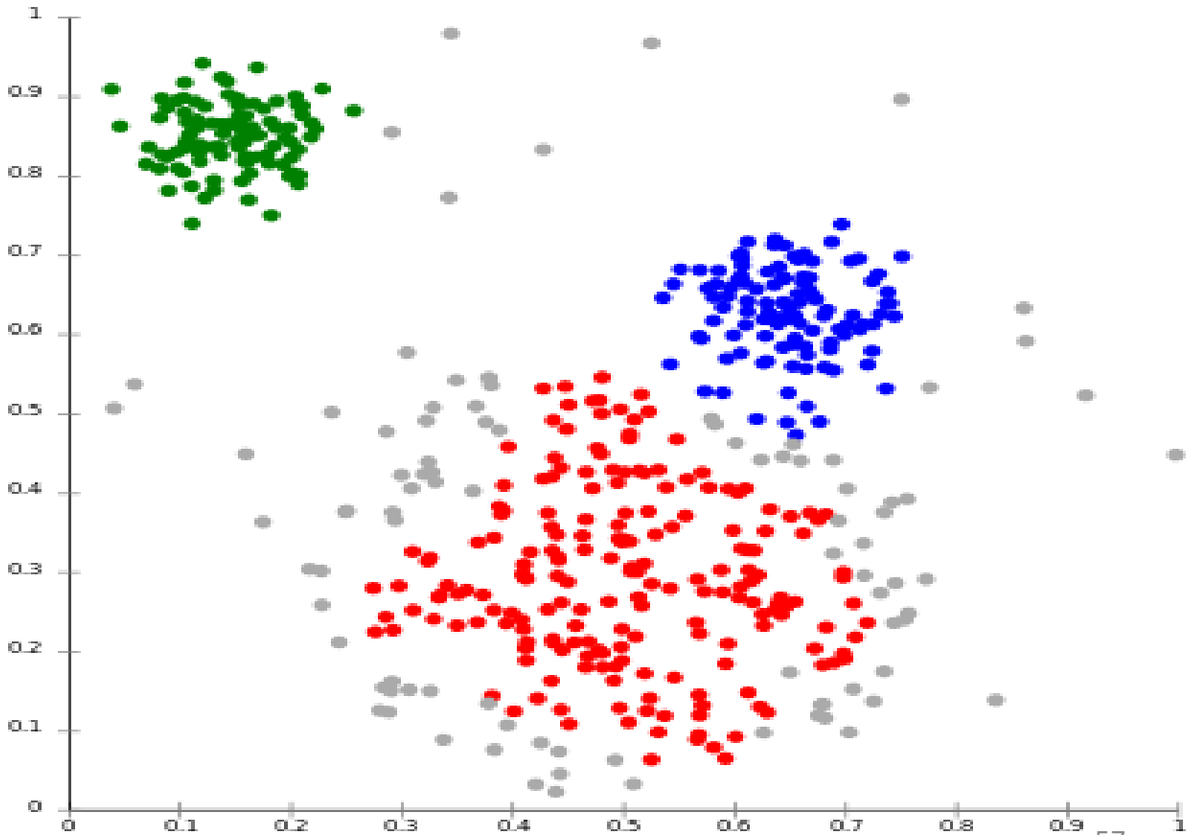


K-Means Clustering (6)



Clustering Methods

- K-Means clustering
  - Centroid-based
- Hierarchical clustering
  - Attach datapoints to root points
- Density-based methods
  - Clusters contain a minimal number of datapoints
- …



Part III: Pattern Learning and Clustering

Summary of Topics Covered

1. Pattern detection
2. Pattern learning
3. Pattern discovery
4. Clustering

Summary of Major Concepts

- Supervised learning, unsupervised learning, semi-supervised learning
- Patterns
  - Pattern language
- Streaming data
- Concept drift
- Pattern detection, pattern learning, pattern discovery

- Clustering
  - Feature vectors
- Algorithms:
  - K-means: cluster centers, centroids

PART IV:
Causal Discovery

Today's Topics

1. Correlation and causation
2. Causal models
   - Bayesian networks
   - Markov networks

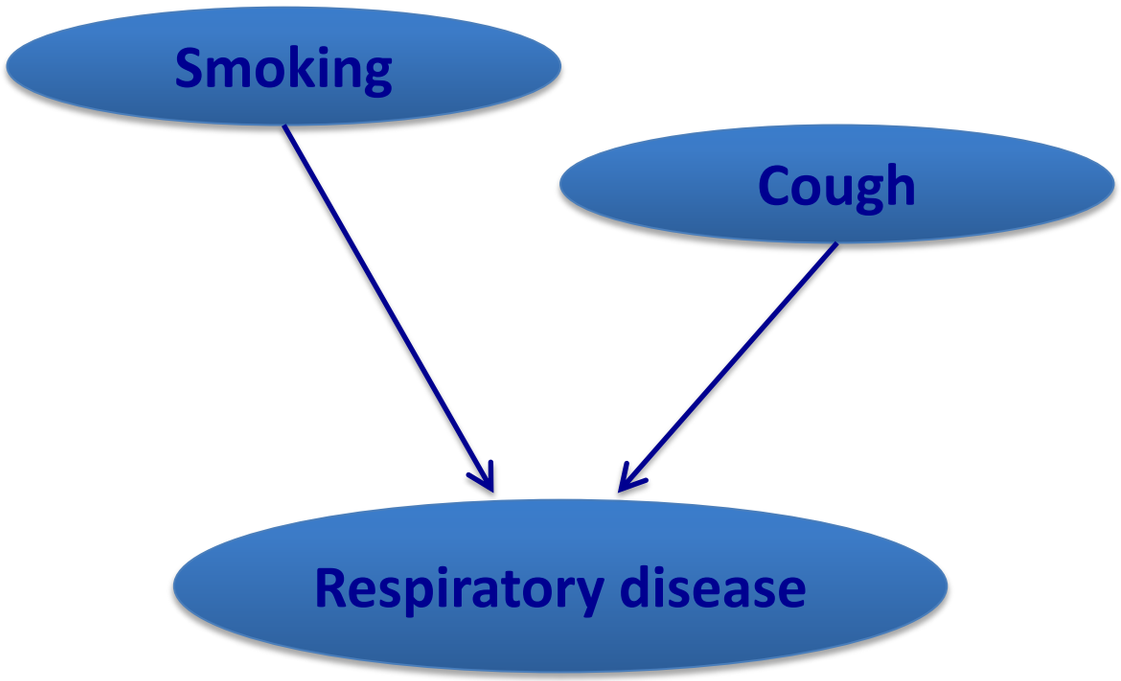
1. CORRELATION AND CAUSATION

Correlation

- Two variables are correlated (associated) when their values are not independent
  - Probabilistically speaking

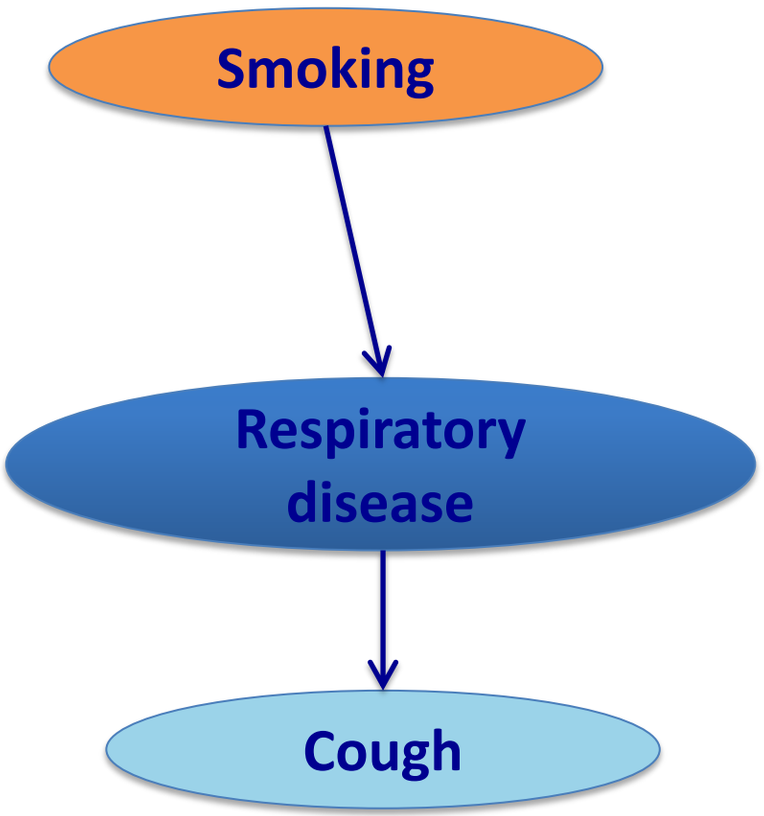
- Examples:
  - When people buy chips they are very likely to buy beer
  - When people have yellow fingers, they are very likely to smoke

Predictive Variables



- Some variables are predictive variables because they are correlated with other target independent variables
  - Smoking and coughing are predictive variables for respiratory disease
- BUT: Do predictive variables indicate the causes?

Cause and Effect

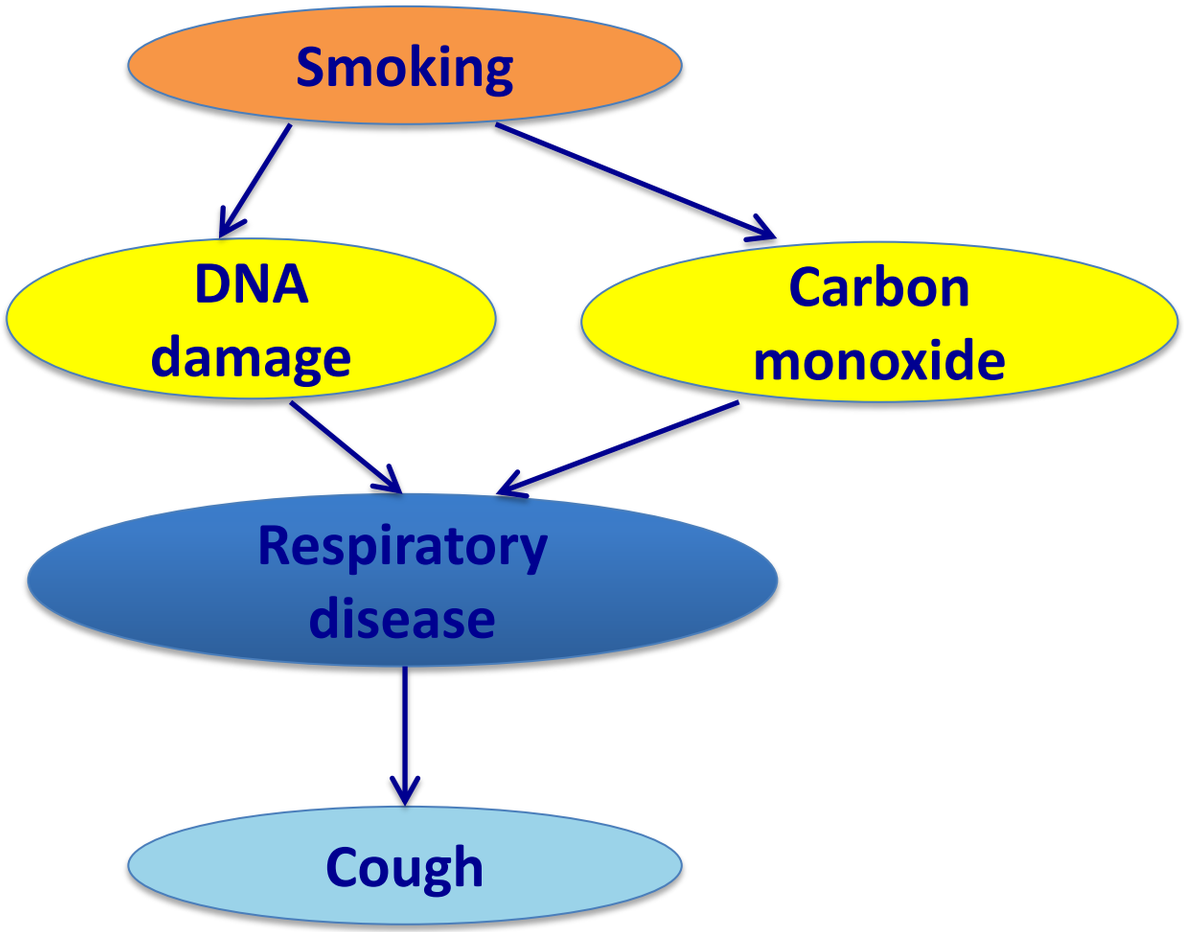


Cause

Effect

- A variable v1 is a cause for variable v2 if changing v1 changes v2
  - Smoking is a cause for respiratory disease
- A variable v3 is an effect of variable v2 if changing v3 does not change v1
  - Cough is an effect of respiratory disease

Latent Variables



- Latent variables are variables that cannot be directly observed, only inferred through a model
  - Eg DNA damage
  - Eg Carbon monoxide inhalation
- Latent variables can be hard to identify, even harder to learn automatically from data

Correlation vs Causation

Correlation

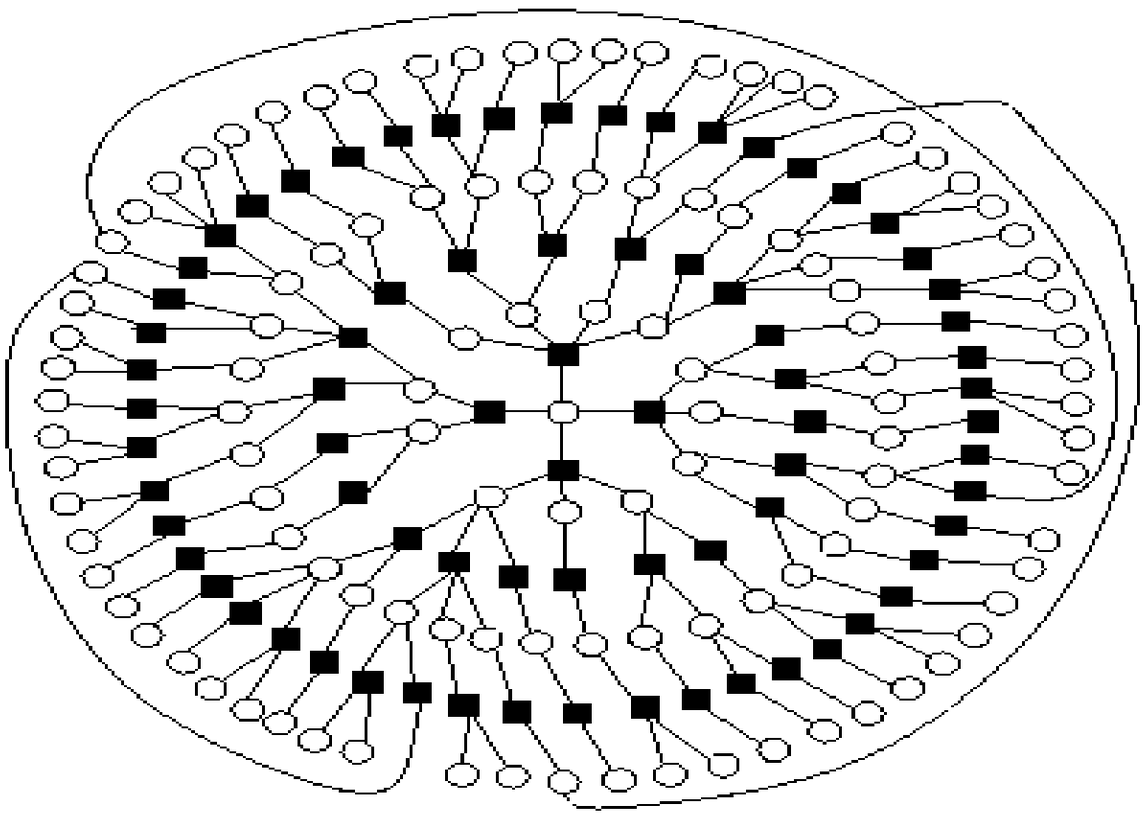
- Knowledge of v1 provides information for v2
  - Eg: yellow fingers, cough, smoking, lung cancer
- Can use any data collected (ie, by simple observation) and do statistical analysis

Causation

- Requires being able to collect specific data that helps show causality (ie, do experiments)
  - Randomized controlled trial
    - Select 1000 people, split evenly
      - 500 (control)
        - » Eg forced to smoke
      - 500 (treatment)
        - » Eg forced not to smoke
    - Collect data
    - Association persists only when causal relation

2. CAUSAL MODELS

(Probabilistic) Graphical Model

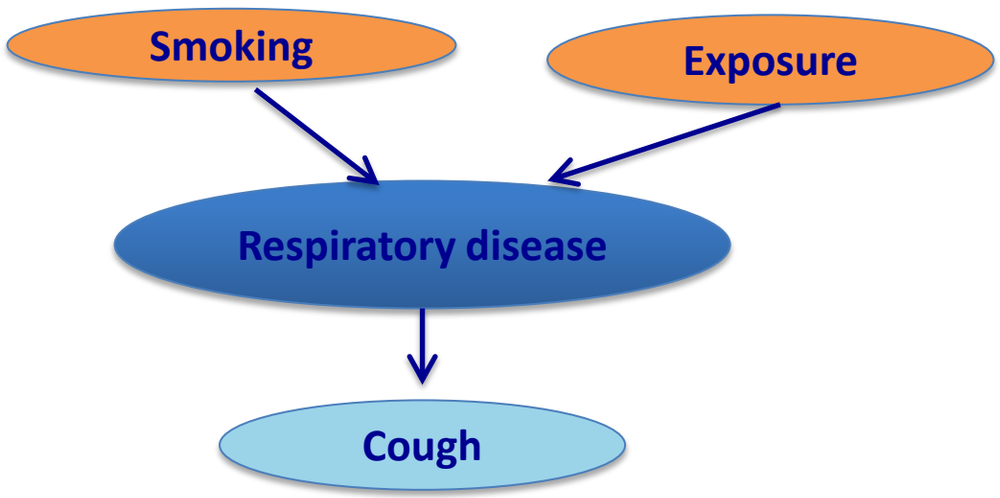


- Graph that captures dependencies among variables
  - Nodes are variables
  - Links indicate dependencies
  - Probabilities that represent how the dependencies work

Graphical Models

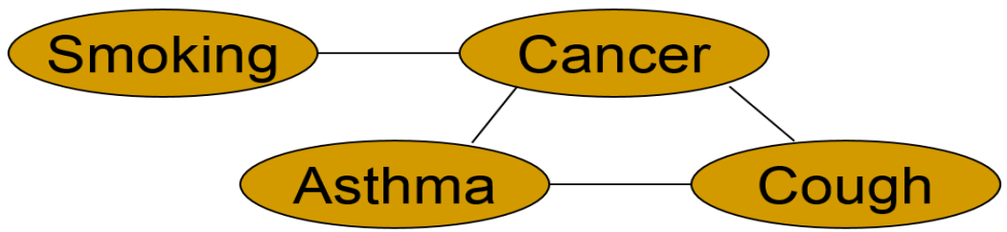
Bayesian Networks

- Graph links have a direction
- Cycles not allowed



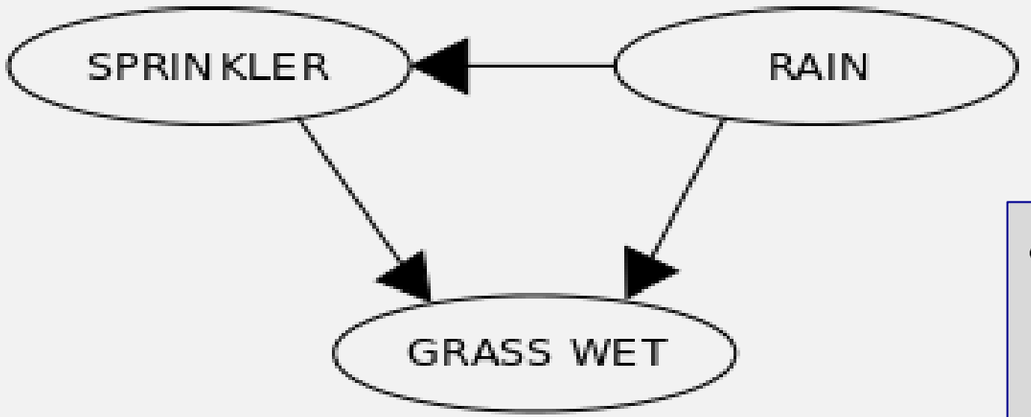
Markov Networks

- Graph links do not have direction
- Cycles are allowed



Bayesian Networks

RAIN	SPRINKLER T	SPRINKLER F
F	0.4	0.6
T	0.01	0.99



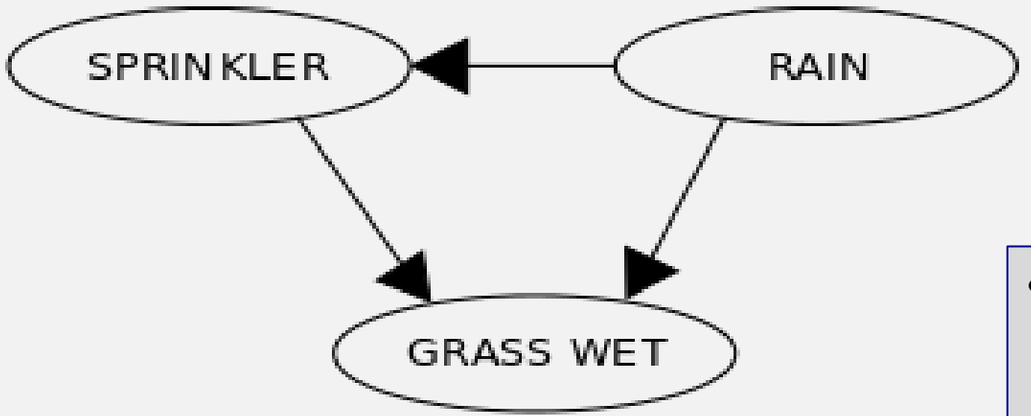
RAIN T	RAIN F
0.2	0.8

SPRINKLER	RAIN	GRASS WET T	GRASS WET F
F	F	0.0	1.0
F	T	0.8	0.2
T	F	0.9	0.1
T	T	0.99	0.01

- A Bayesian network is a graph
  - Directed edges show how variables influence others
    - No cycles allowed
  - Conditional probability distribution (tables or functions) show the probability of the value of a variable given the values of its parent variables
  - A variable is only dependent on its parent variables, not on its

Bayesian Inference

	SPRINKLER	
RAIN	T	F
F	0.4	0.6
T	0.01	0.99

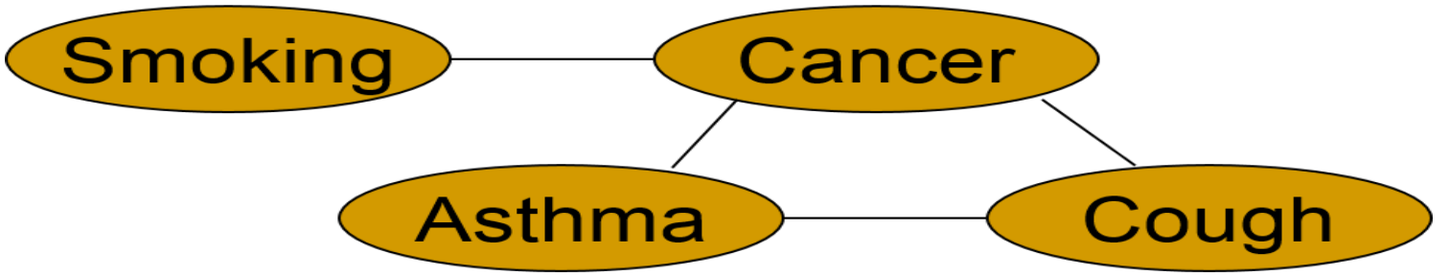


	RAIN	
T	F	
0.2	0.8	

		GRASS WET	
SPRINKLER	RAIN	T	F
F	F	0.0	1.0
F	T	0.8	0.2
T	F	0.9	0.1
T	T	0.99	0.01

- Bayesian inference is used to reason over a Bayesian network to determine the probabilities of some variables given some observed variables
  - Eg: Given that the grass is wet, what is the probability that it is raining?

Markov Networks



- A Markov network is an undirected graphical model that includes a *potential function* for each clique of interconnected nodes

Smoking	Cancer	Φ(S,C)
False	False	4.5
False	True	4.5
True	False	2.7
True	True	4.5

Causal Models

- A causal model is a Bayesian network where all the relationships among variables are causal
- Causal models represent how independent variables have an effect on dependent variables
- Causal reasoning uses the probabilities in the causal model to make inferences about the value of variables given the values of others
  - Eg: Given that the grass is wet, what is the probability that it rained?

Learning Causal Models

Parameter Learning

- Learning the parameters (probabilities) of the model

Structure Learning

- Learning the structure of the model
  - Usually more challenging

Summary of Topics Covered

1. Correlation and causation
2. Causal models
   - Bayesian networks
   - Markov networks

Summary of Major Concepts

- Predictive variables
- Cause and effect
- Latent variables
- Correlation vs causation
- Randomized Control Trials

- Probabilistic graphical models
- Bayesian networks
- Markov networks
- Causal models
- Parameter learning
- Structure learning

PART V:
Simulation and Modeling

Simulation

- Simulation is an approach to data analysis that uses a mathematical or formal model of a phenomenon to run different scenarios to make predictions
  - Eg By simulating people in a city and where they drive every day, we can analyze scenarios where there is a flu epidemic and predict people's behavior changes
- Simulation models can be improved to make predictions that correspond to the observed data
- From a Workflow Sketch to a Computational Workflow

Example: Landscape Evolution

1. Trees tip over, creating a downslope flux of regolith proportional to slope

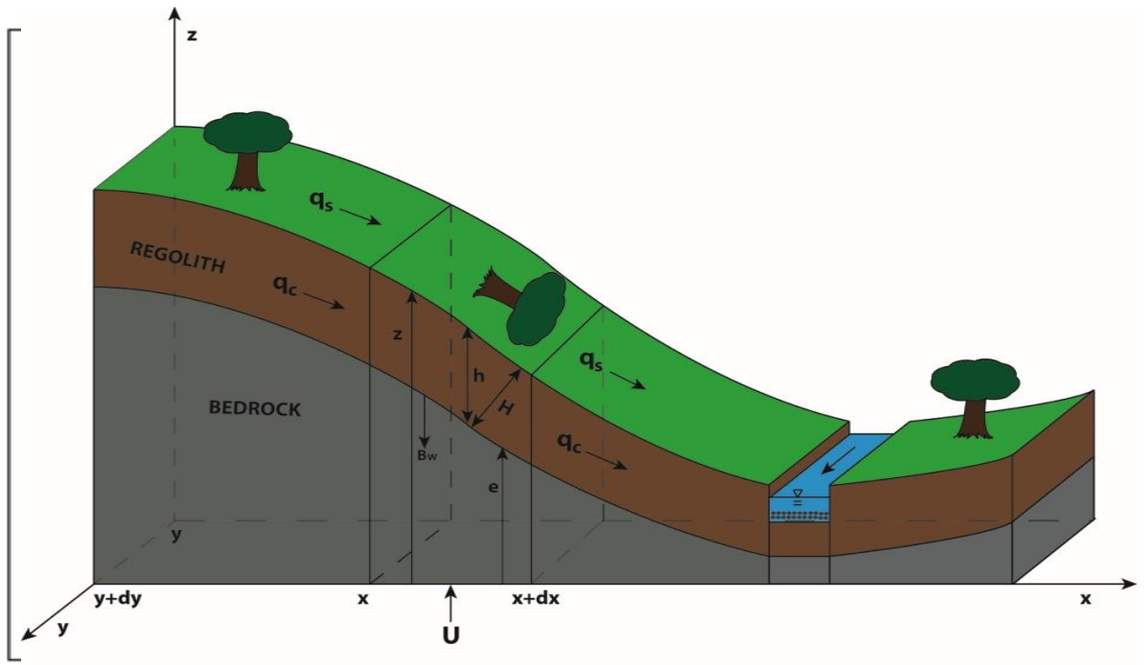
2. Proportionality constant is a function of tree type

3. Tree type is a function of moisture availability

4. Moisture availability depends upon regolith thickness

5. Regolith thickness depends upon downslope flux of regolith

6. downslope flux of regolith depends upon trees tipping over

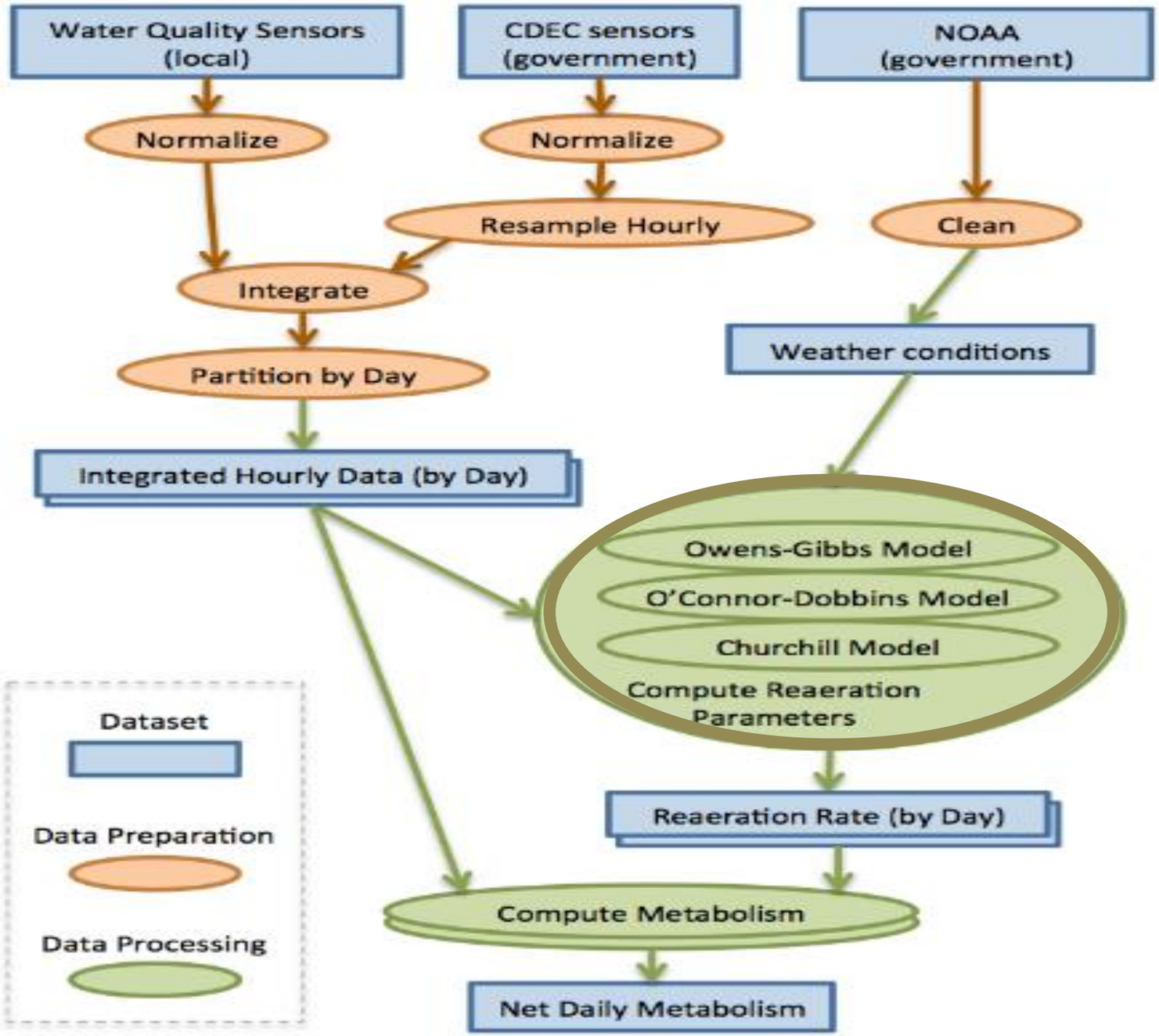


Workflow Sketch

Data preparation

Feature extraction

Models of how water mixes with air ("reaeration") and what chemical reactions occur ("metabolism")



PART VI:
Practical Use of Machine Learning and Data Analysis

RECAP:
Different Data Analysis Tasks

- **Classification**
  - Assign a label (ie, a class) for a new instance given many labeled instances
- **Clustering**
  - Form clusters (ie, groups) with a set of instances
- **Pattern learning/detection**
  - Learn patterns (i.e., regularities) in data

- **Causal modeling**
  - Learn causal (probabilistic) dependencies among variables
- **Simulation modeling**
  - Define mathematical formulas that can generate data that is close to observations collected

RECAP:
Different Data Analysis Tasks

- **Classification**
- **Clustering**
- **Pattern learning**
- **Causal modeling**
- **Simulation modeling**
- **…**

- Each type of task is characterized by the kinds of data they require and the kinds of output they generate
- Each type of task uses different algorithms

When Facing a Learning Task

- Supervised, unsupervised, or semi-supervised: cost of labels
- Setting up the learning task
  - Classification: What classes to choose
  - Clustering: How many target clusters
  - Causality: What observables
- What data is available
  - Collecting data
  - Buying data

- What features to choose
  - Try defining different features
  - For some problems, hundreds and maybe thousands of features may be possible
  - Sometimes the features are not directly observable (ie, there are "*latent*" variables)
- What learning method
  - Better to try different ones
- Scalability: processing time

Recent Trends: Neural Networks and “Deep Learning”