Learning Systems (DT8008)

Introduction to Machine Learning

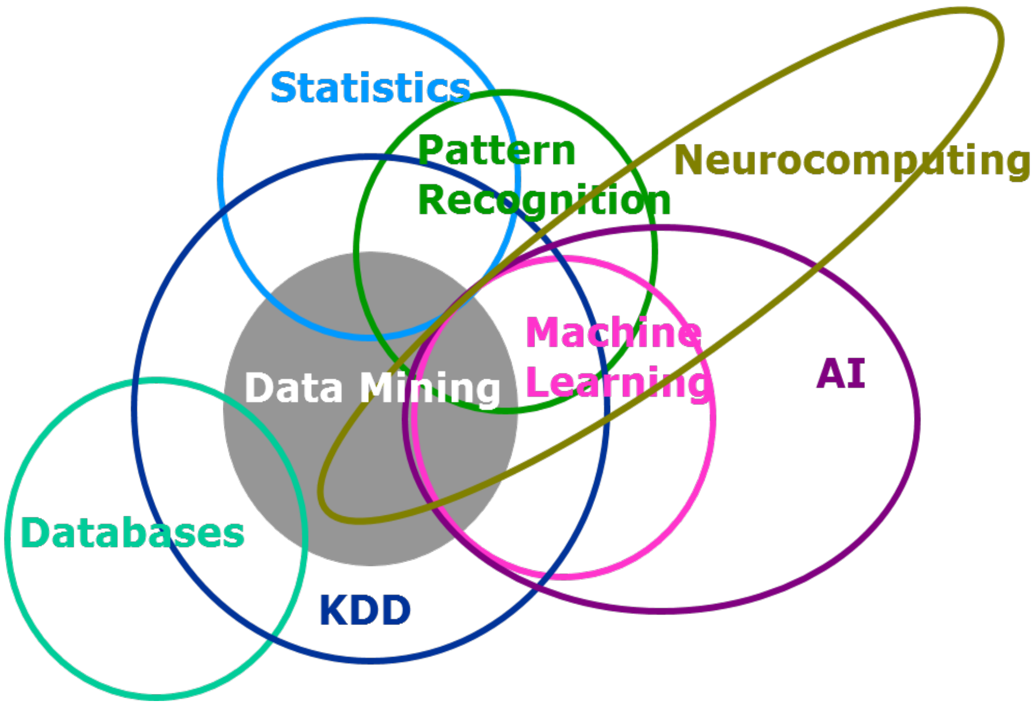
Dr. Mohamed-Rafik Bouguelia
mohamed-rafik.bouguelia@hh.se

Halmstad University

What is Machine Learning ?

What is Machine Learning (ML) ?

- **Machine Learning**: Field of study that gives computers the ability to learn without being explicitly programmed with rules.



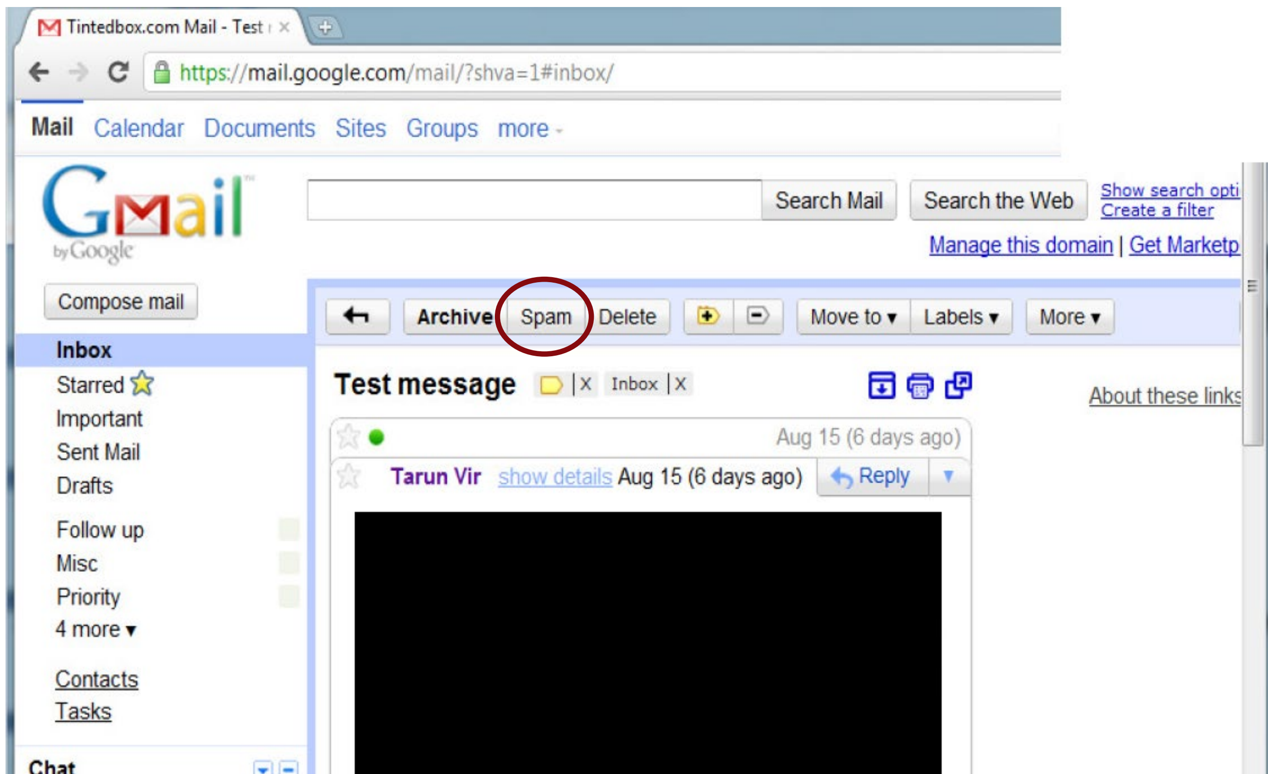
What is Machine Learning (ML) ?

Definition:

- A computer program is said to *learn* from **experience** ***E*** with respect to some **task** ***T*** and **performance measure** ***P*** if its performance on *T*, as measured by *P*, improves with experience *E*.

Example:

Suppose your email program watches which emails you do or do not mark as spam, and based on that learns how to better filter spam. What is the **task T** in this setting ?



What is Machine Learning (ML) ?

Definition:

- A computer program is said to *learn* from **experience** ***E*** with respect to some **task** *T* and **performance measure** ***P*** if its performance on *T*, as measured by *P*, improves with experience *E.*

Example:

Suppose your email program watches which emails you do or do not mark as spam, and based on that learns how to better filter spam. What is the **task T** in this setting ?

1. Classifying emails as spam or not spam.
2. Watching you label emails as spam or not spam.
3. The number (or fraction) of emails correctly classified as spam/not spam.
4. None of the above—this is not a machine learning problem.

What is Machine Learning (ML) ?

Definition:

- A computer program is said to *learn* from **experience** ***E*** with respect to some **task** ***T*** and **performance measure** ***P*** if its performance on *T*, as measured by *P*, improves with experience *E.*

Example:

Suppose your email program watches which emails you do or do not mark as spam, and based on that learns how to better filter spam. What is the **task T** in this setting ?

T 1. Classifying emails as spam or not spam.

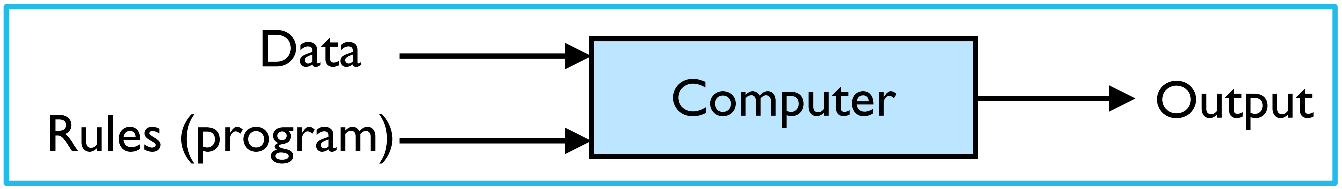
E 2. Watching you label emails as spam or not spam.

P 3. The number (or fraction) of emails correctly classified as spam/not spam.

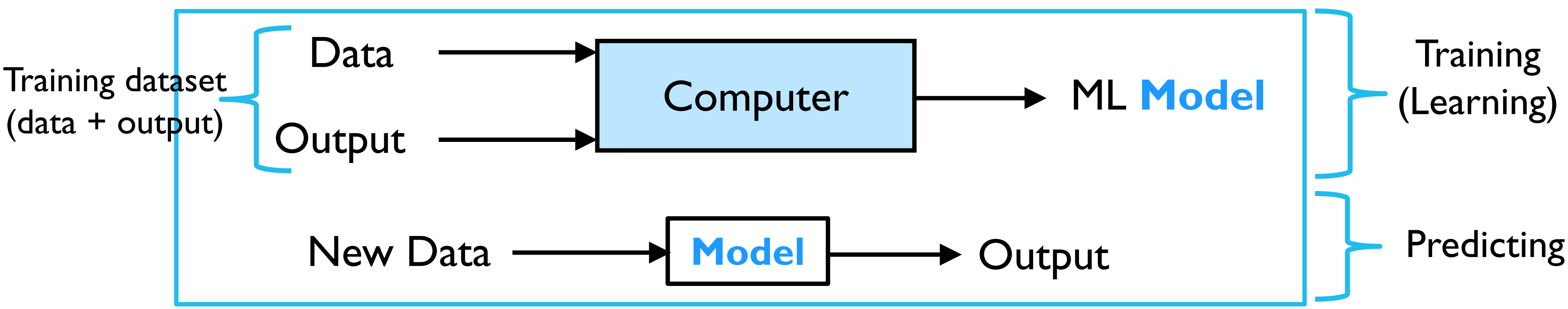
4. None of the above—this is not a machine learning problem.

What is Machine Learning (ML) ?

- **Usual programming**

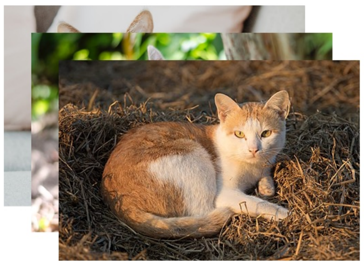


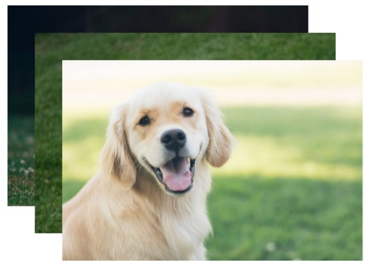
- **(Supervised) Machine learning**



- Machine learning algorithms build a **model** from the **training data**, then uses this model to make **predictions** or decisions without being explicitly programmed to perform the task.

What is Machine Learning (ML) ?


Cat


Dog


Lion

Example:
The data consist of images …

The output consists of labels (Cat, Dog or Lion)

- **(Supervised) Machine learning**

Data → Computer → ML **Model**
Output → Computer

Training (Learning)

New Data → **Model** → Output

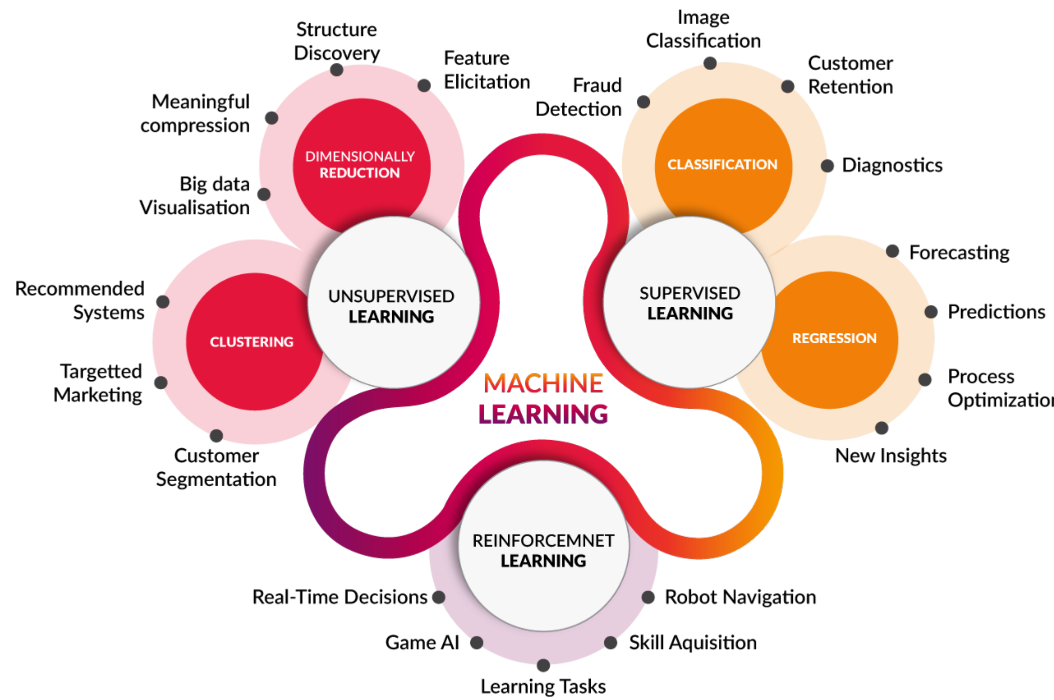
Predicting

Dog (predicted label)
Cat 5%, Dog 65%, lion 30%

What is Machine Learning (ML) ?

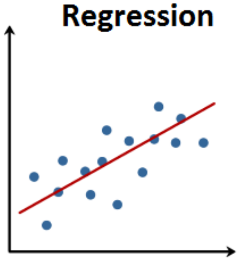
Machine learning types:

- Supervised learning
- Unsupervised learning
- Others
  - Reinforcement learning
  - Semi-supervised learning
  - Active learning
  - etc.



Supervised ML:
Training data includes desired outputs.

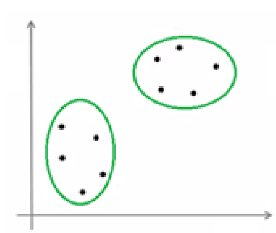
Regression



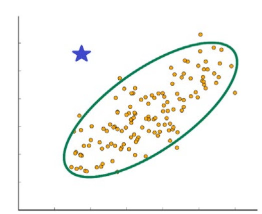
Classification

Unsupervised ML:
Training data does not include outputs.

Clustering



Anomaly detection



Question

- You want to do some task …
  - e.g. predicting if an email is a spam or not.

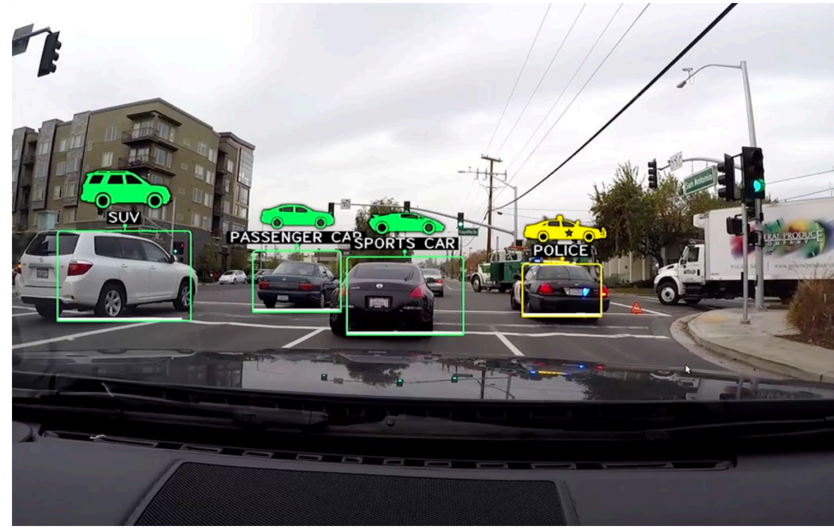
- Why would you need machine learning?

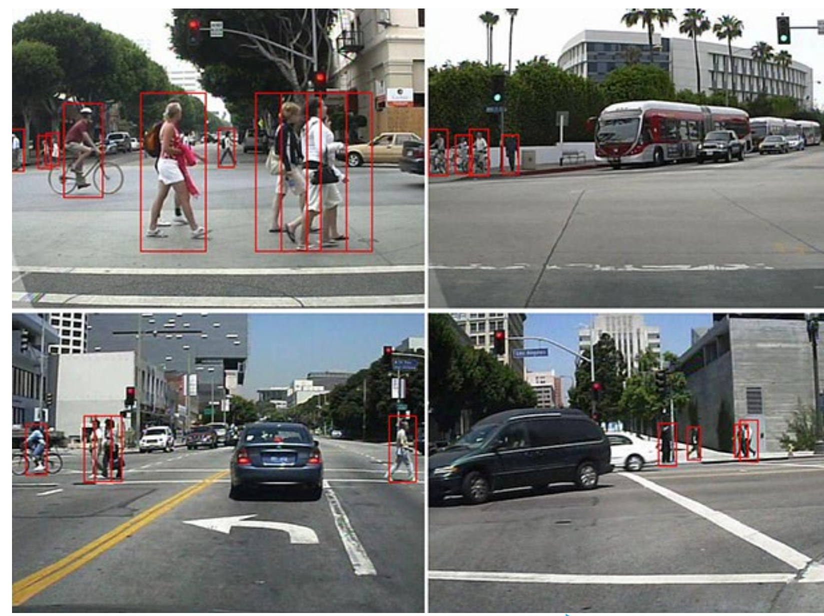
- Why don't you just explicitly program/write rules to perform the task (without ML) ?
  - e.g. if the email is from an unknown sender and contains keywords such as
    - "*send x usd*", "*only for you*", "*invest now*", "your computer is compromised" …
  - then it's a spam

Example (self-driving car)

Consider the following problem:

- You have a camera on your car that periodically captures images of the road and send them to your app.
- You want your app to recognize what is present on each image (pedestrians, bikes, other cars, etc …)





Question:

- Why do we need machine learning for this? Why don't we just explicitly program/write rules that allows us to recognize what the image contains?

Example (self-driving car)

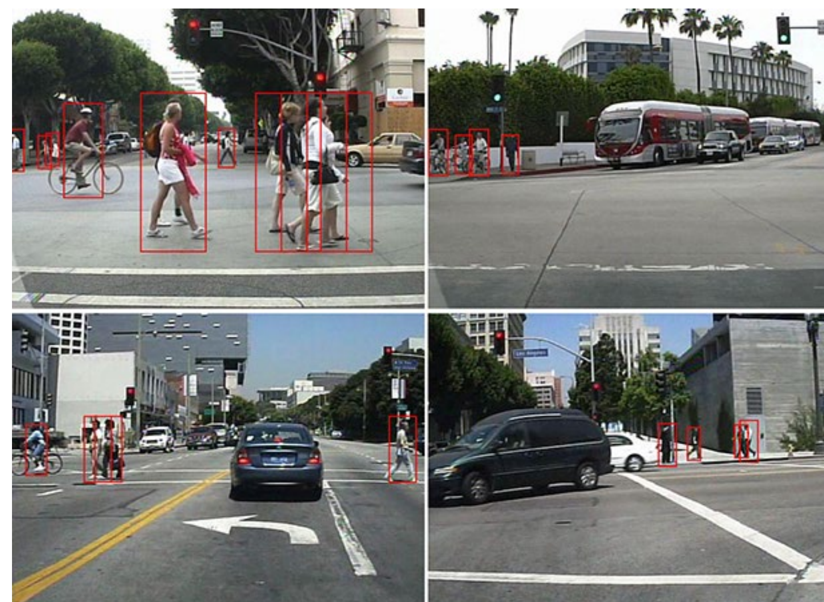
Consider the following problem:

- You have a camera on your car that periodically captures images of the road and send them to your app.
- You want your app to recognize what is present on each image (pedestr… bikes, other cars, etc …)

Question:

- Why d… machine learning for thi… don't we just explicitly prog…/write rules that allows us to recognize what the image contains?

Hard without Machine Learning



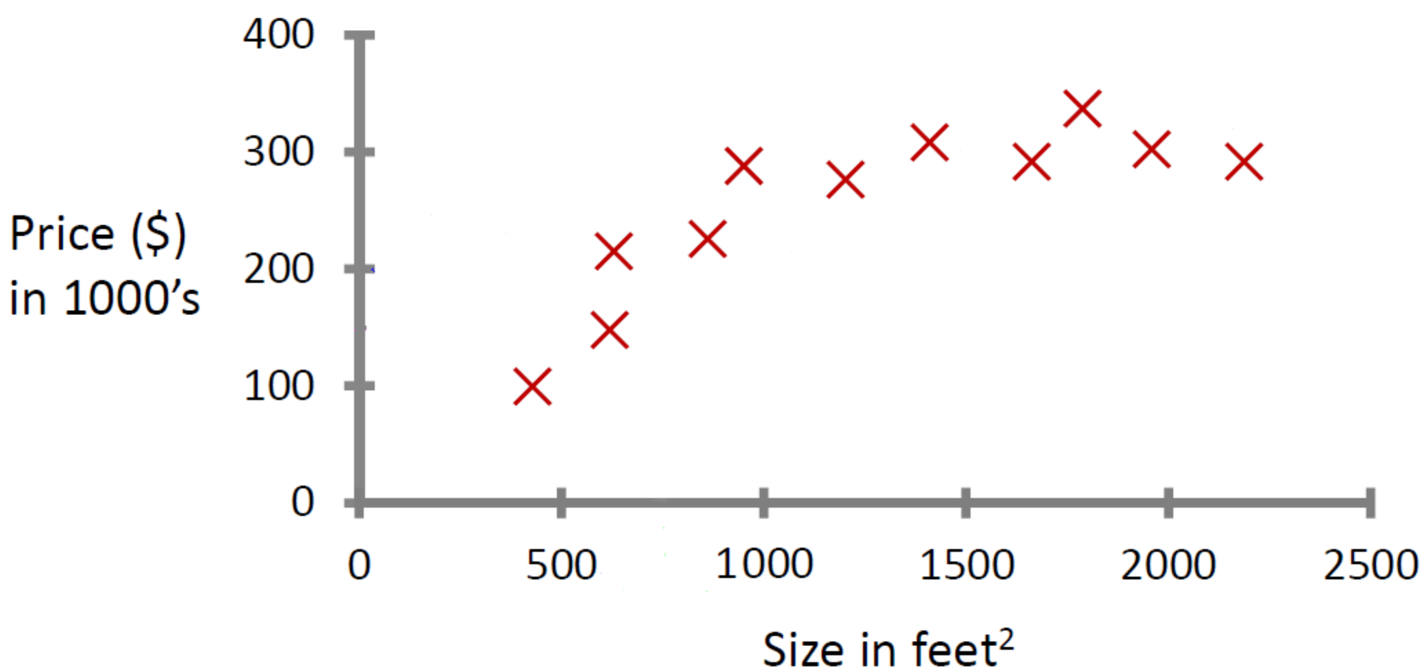
HALMSTAD UNIVERSITY

Introduction to Supervised Machine Learning

Regression problems

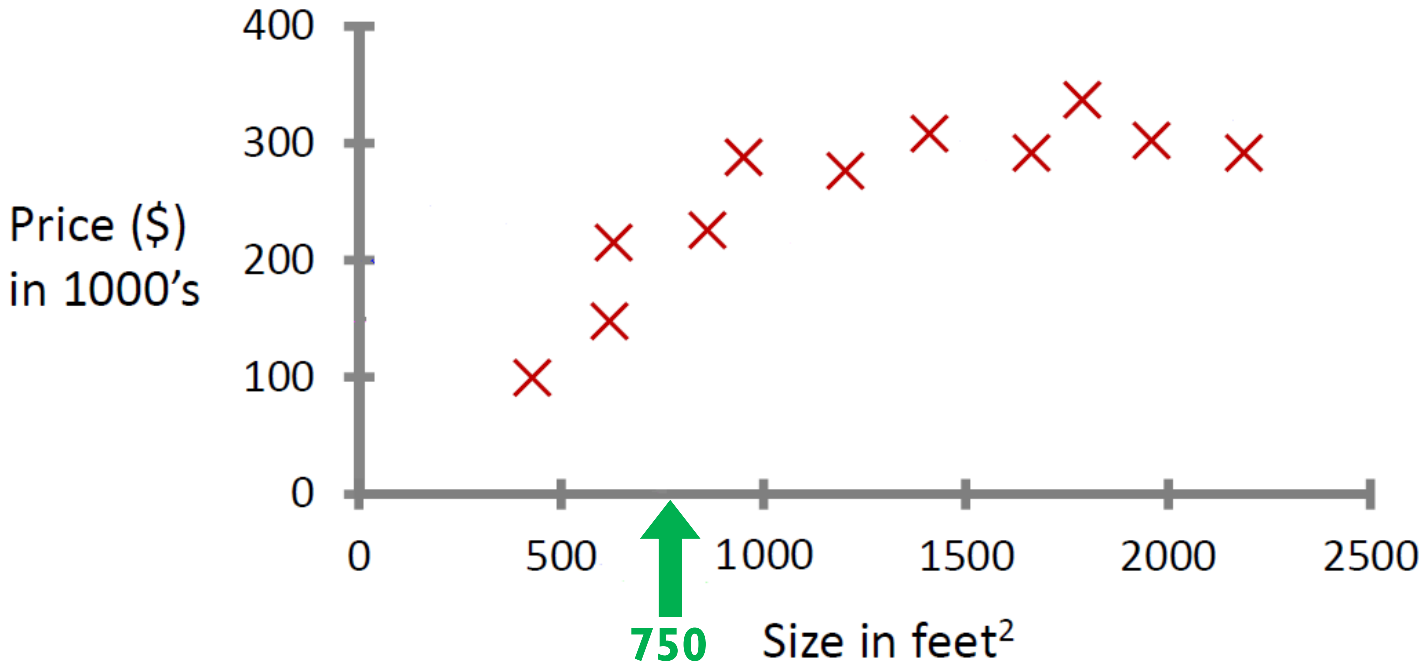
Introduction to Supervised Learning

- Housing price prediction (regression)



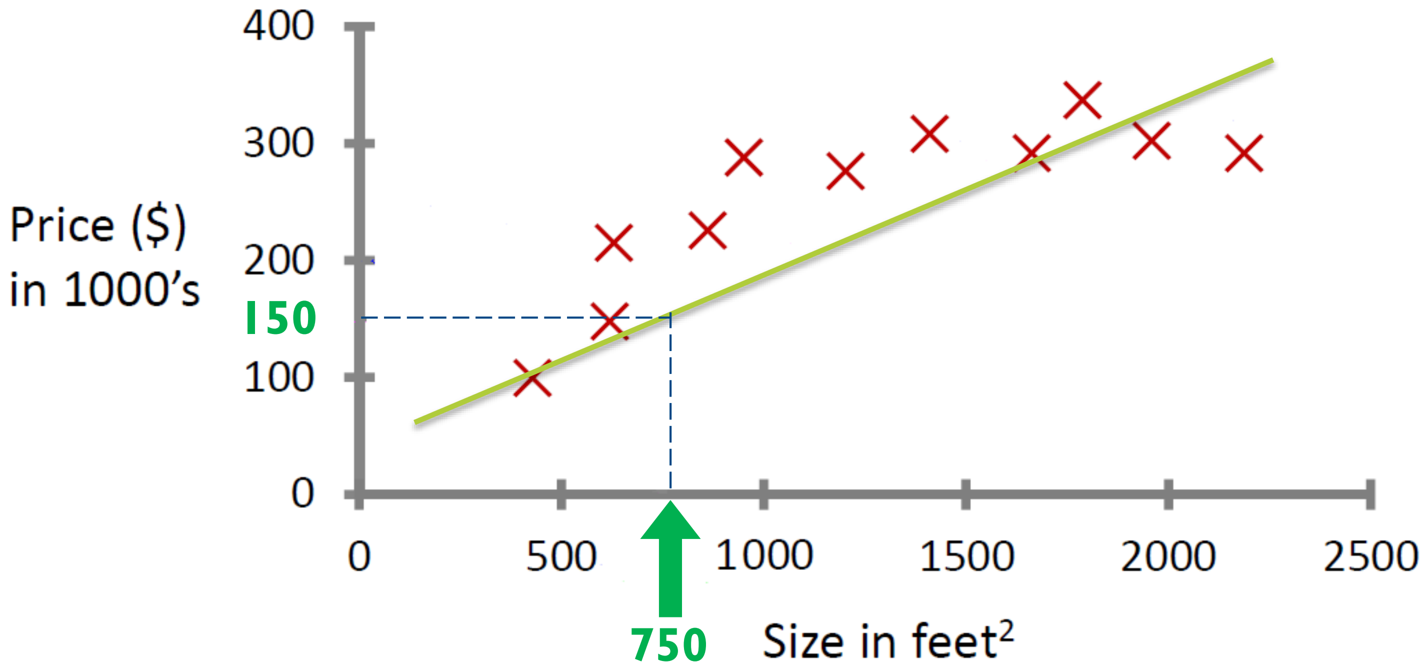
Introduction to Supervised Learning

- Housing price prediction (regression)
  - Suppose that you want to sell a house of **size 750 feet²** and want to know how much you can get for this house, i.e. **predict its price**.
  - How can a learning algorithm help you?



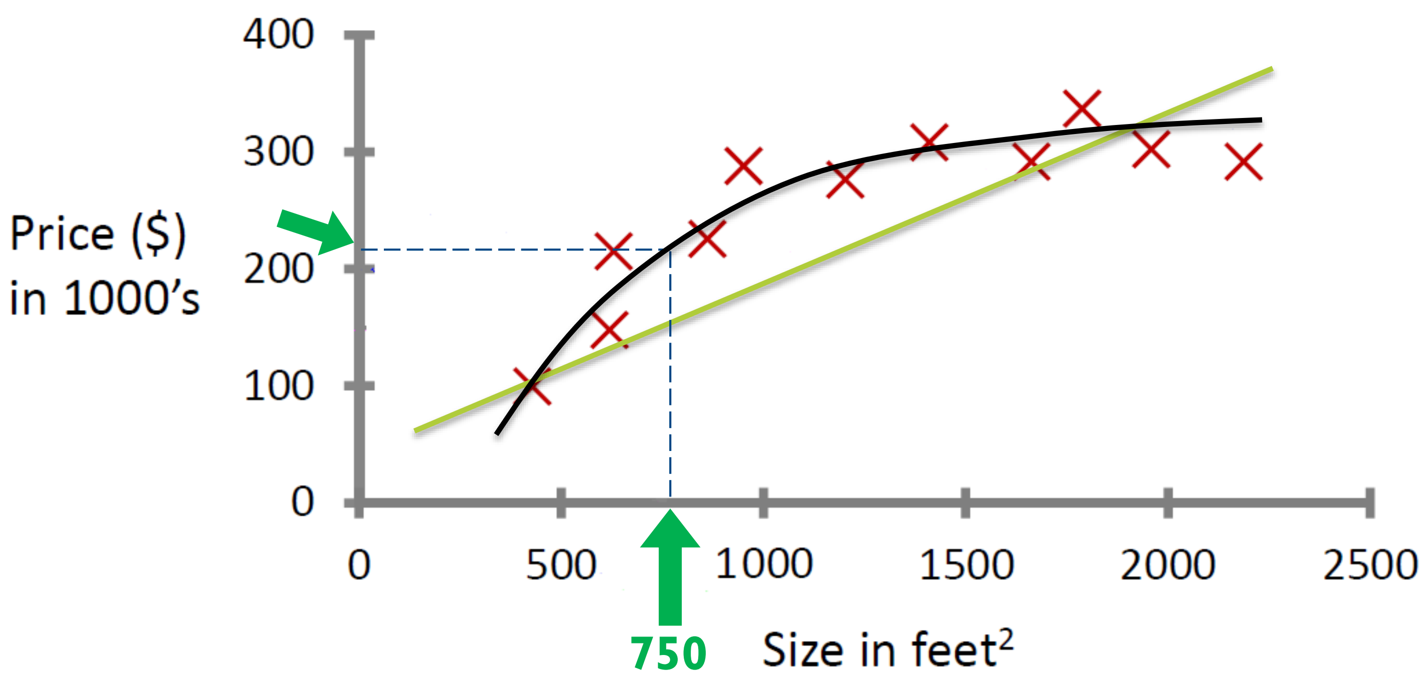
Introduction to Supervised Learning

- Housing price prediction (regression)
  - you can fit a straight line to the data, and predict the price of the house.



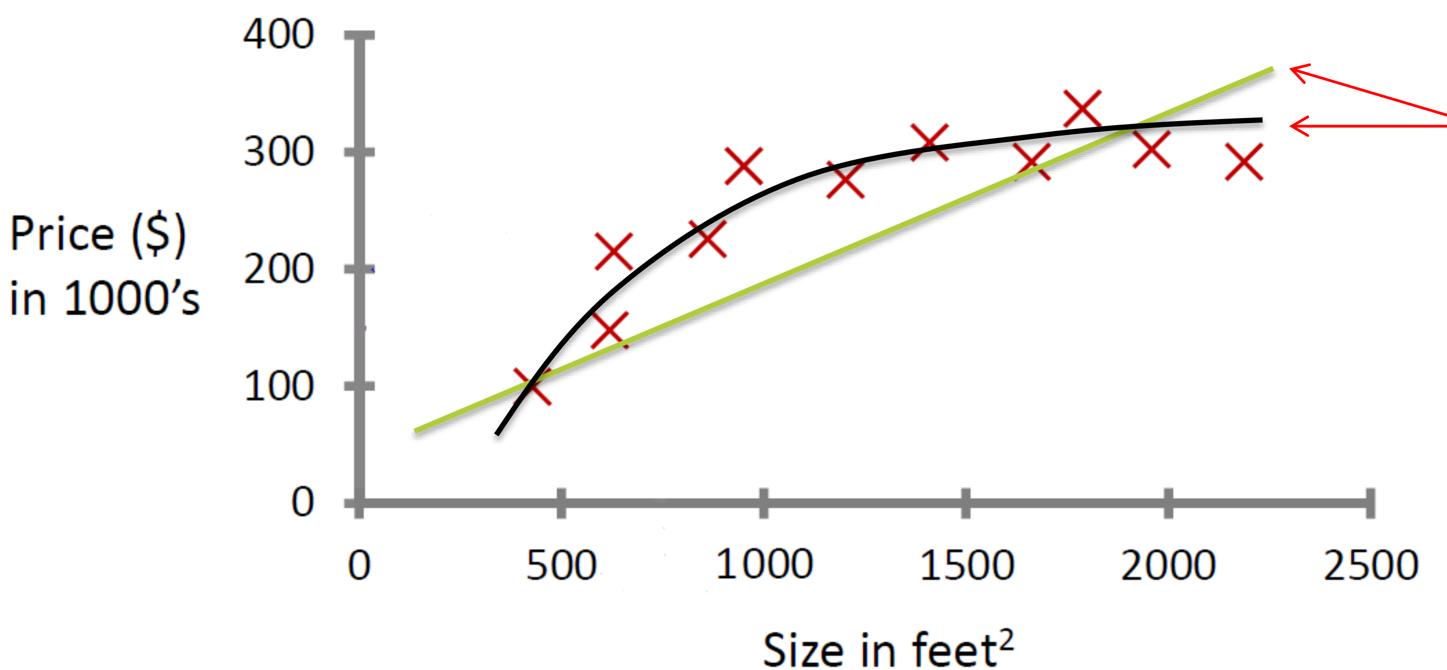
Introduction to Supervised Learning

- Housing price prediction (regression)
  - you can fit a straight line to the data, and predict the price of the house.
  - or maybe its better to fit a quadratic function (2^{nd} order polynomial).



Introduction to Supervised Learning

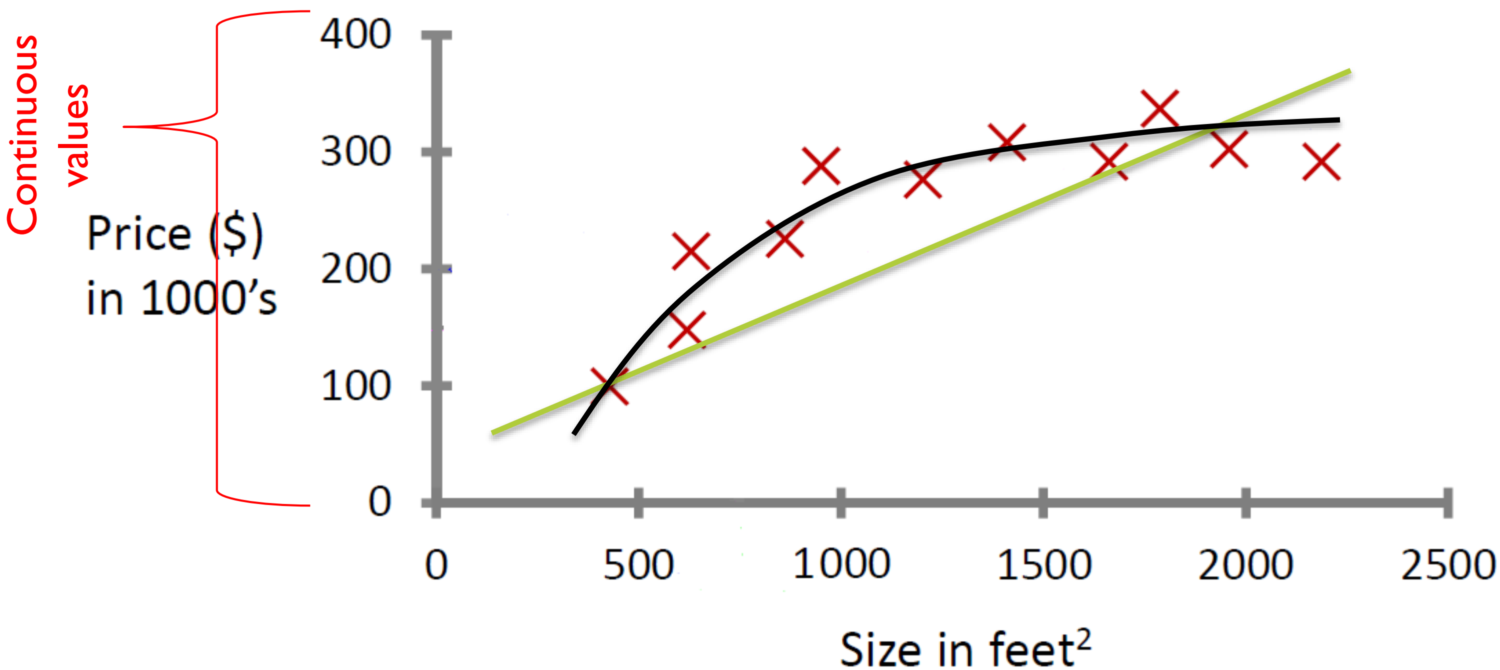
- Housing price prediction (regression)
  - you can fit a straight line to the data, and predict the price of the house.
  - or maybe its better to fit a quadratic function (2^{nd} order polynomial).



How to decide which model to choose for this dataset ?
We will see this later in the course when we talk about **model selection** …

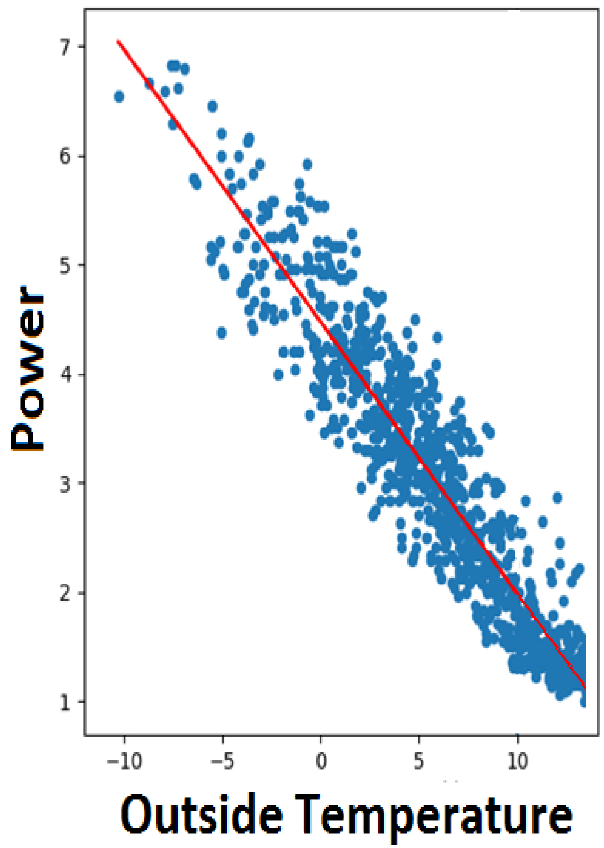
Introduction to Supervised Learning

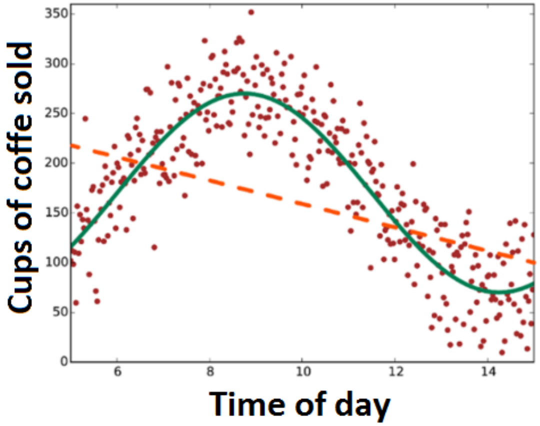
- Housing price prediction (regression)
  - This is an example of a **supervised learning** algorithm:
    - The right answers (here, the prices) are given in the training dataset.
  - More specifically, this example was a **regression problem**:
    - Predicting a continuous valued output (price).

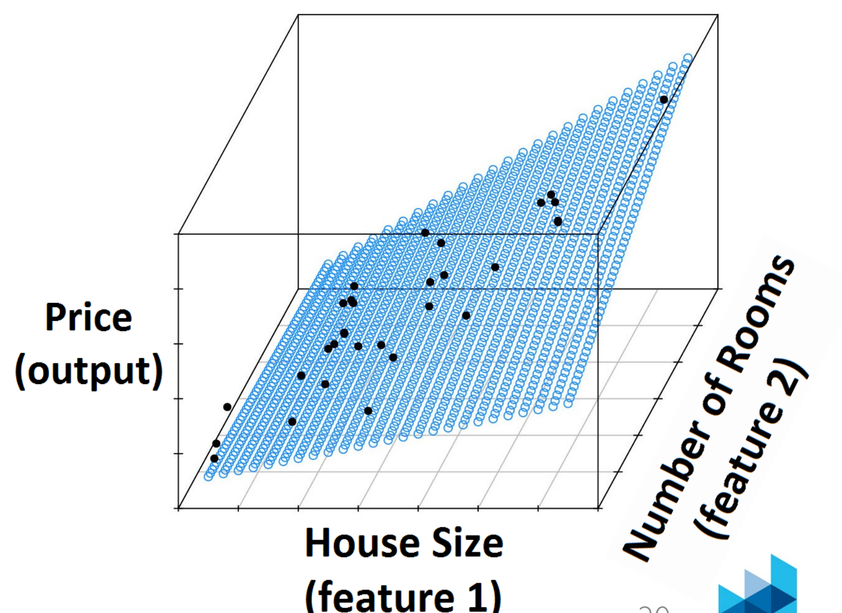


Introduction to Supervised Learning

Other regression examples





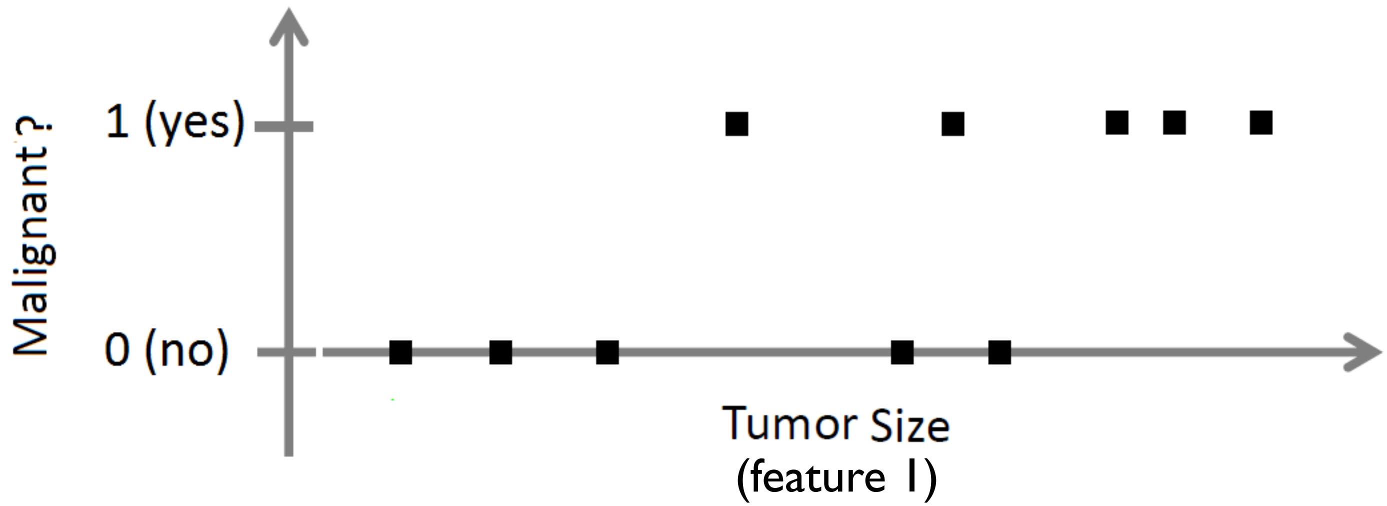


Introduction to Supervised Machine Learning

Classification problems

Introduction to Supervised Learning

- Breast cancer malignant/benign (classification)

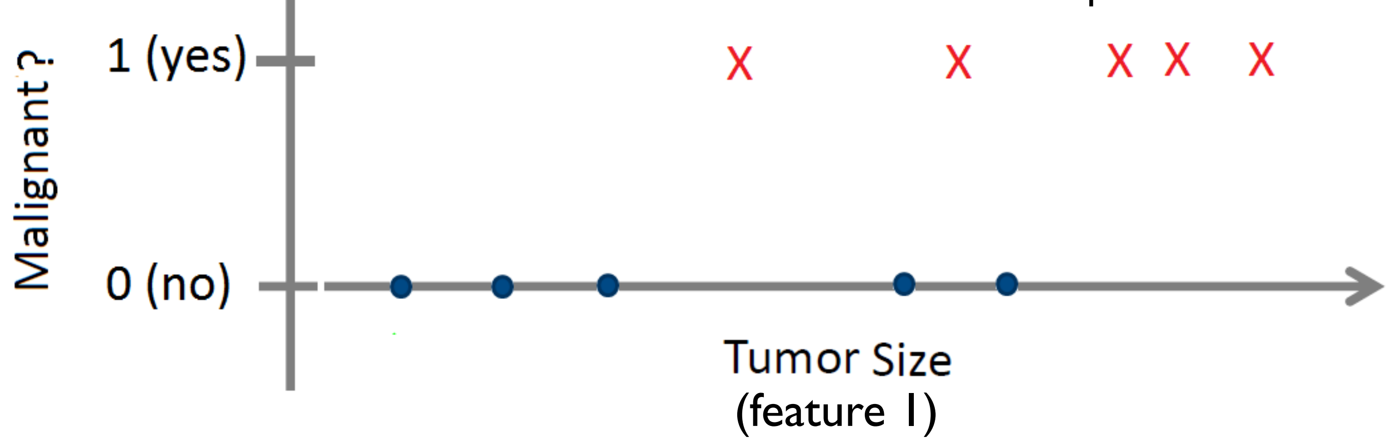


We only have discrete output values (in this example: 1 or 0)

Introduction to Supervised Learning

- Breast cancer malignant/benign (classification)

We don't need the second axis if we simply visualize the classes with different colors or shapes.

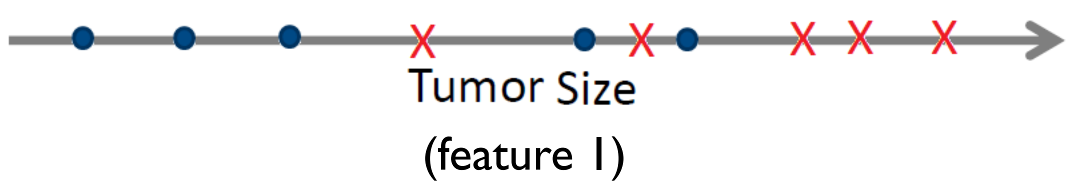


Two axis

Malignant ?

X = 1 (yes)

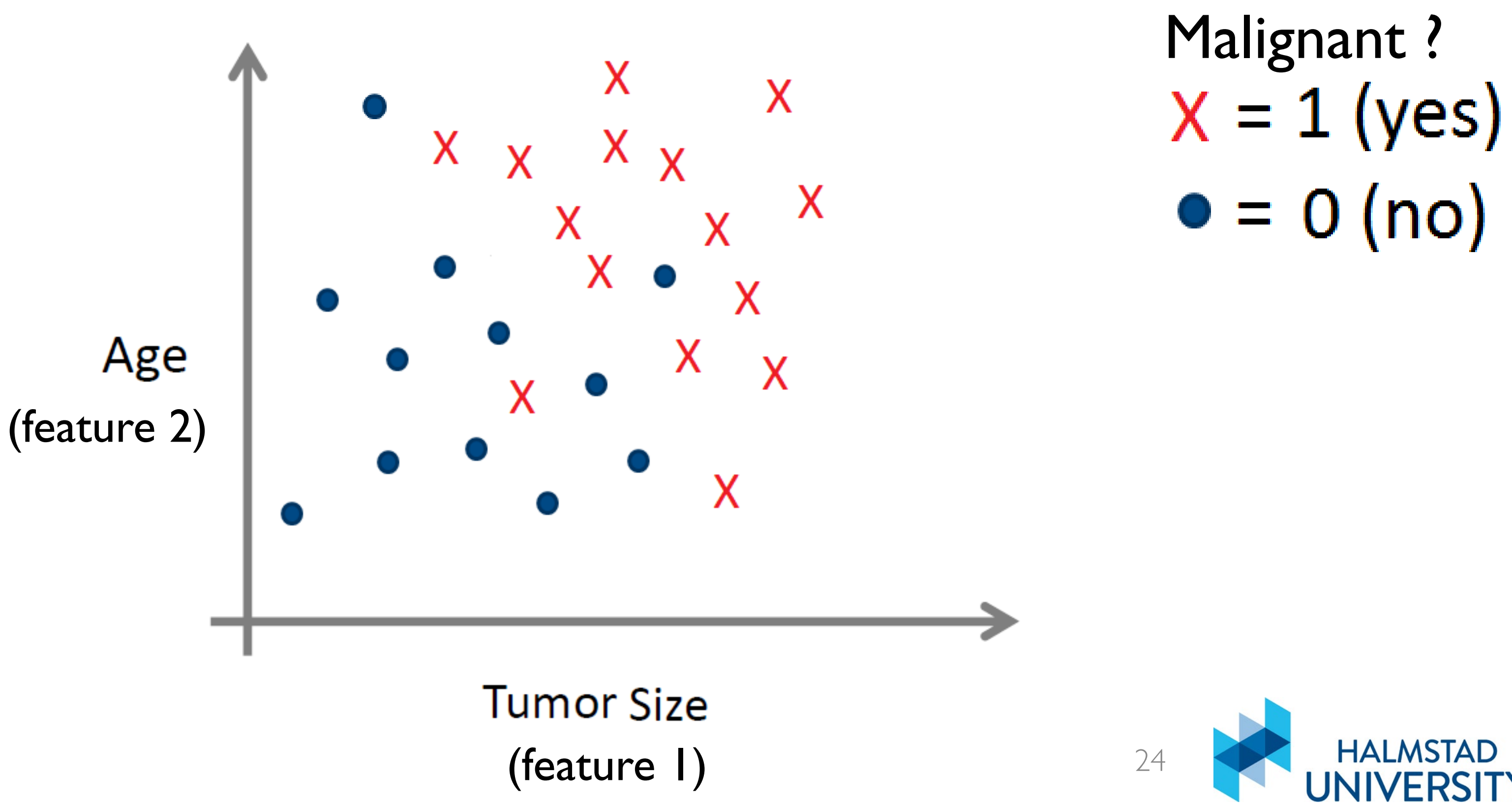
● = 0 (no)



Only one axis

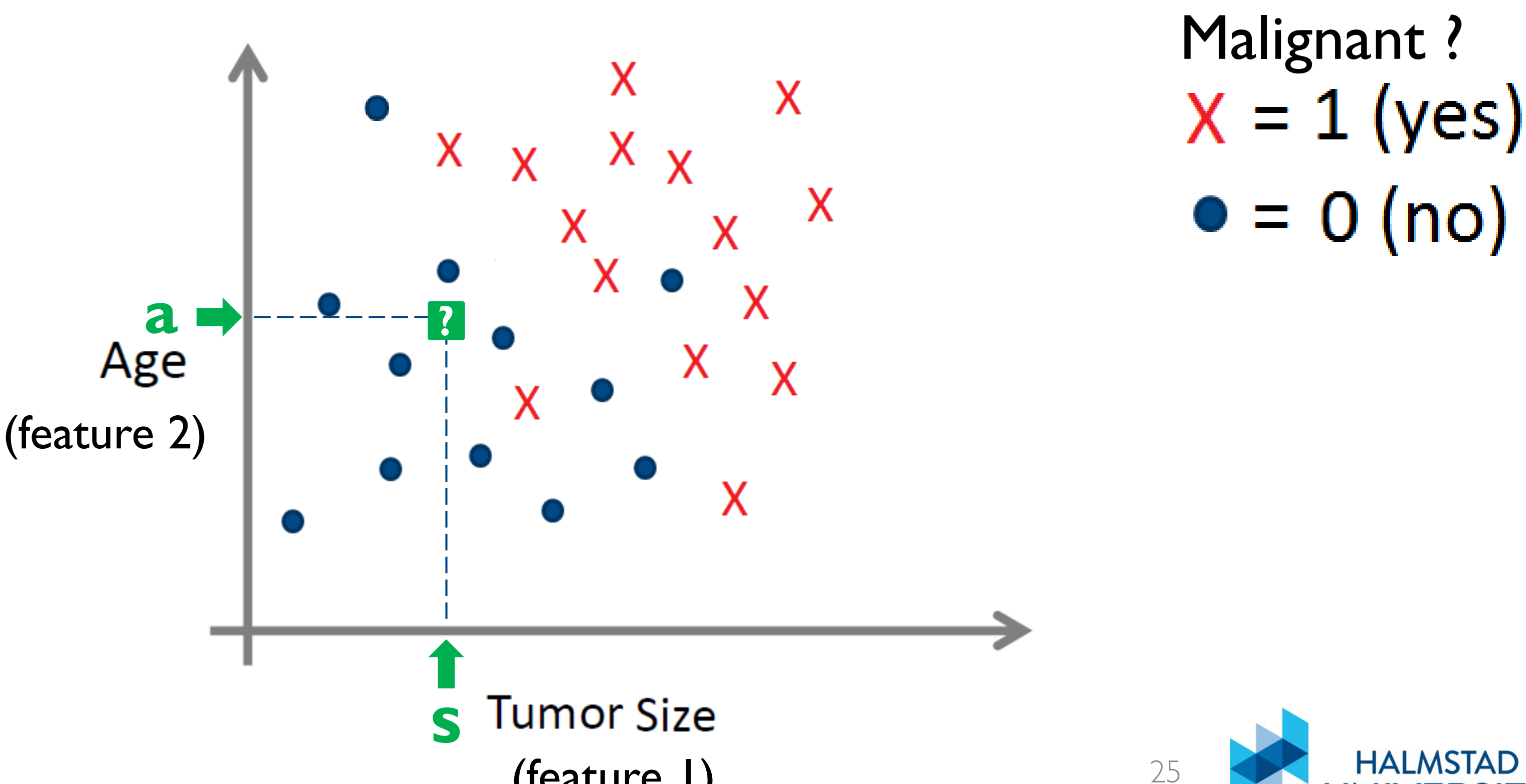
Introduction to Supervised Learning

- Breast cancer malignant/benign (classification)
  - The patients data can be characterized by more than one feature
    - e.g. Tumor size and Age …



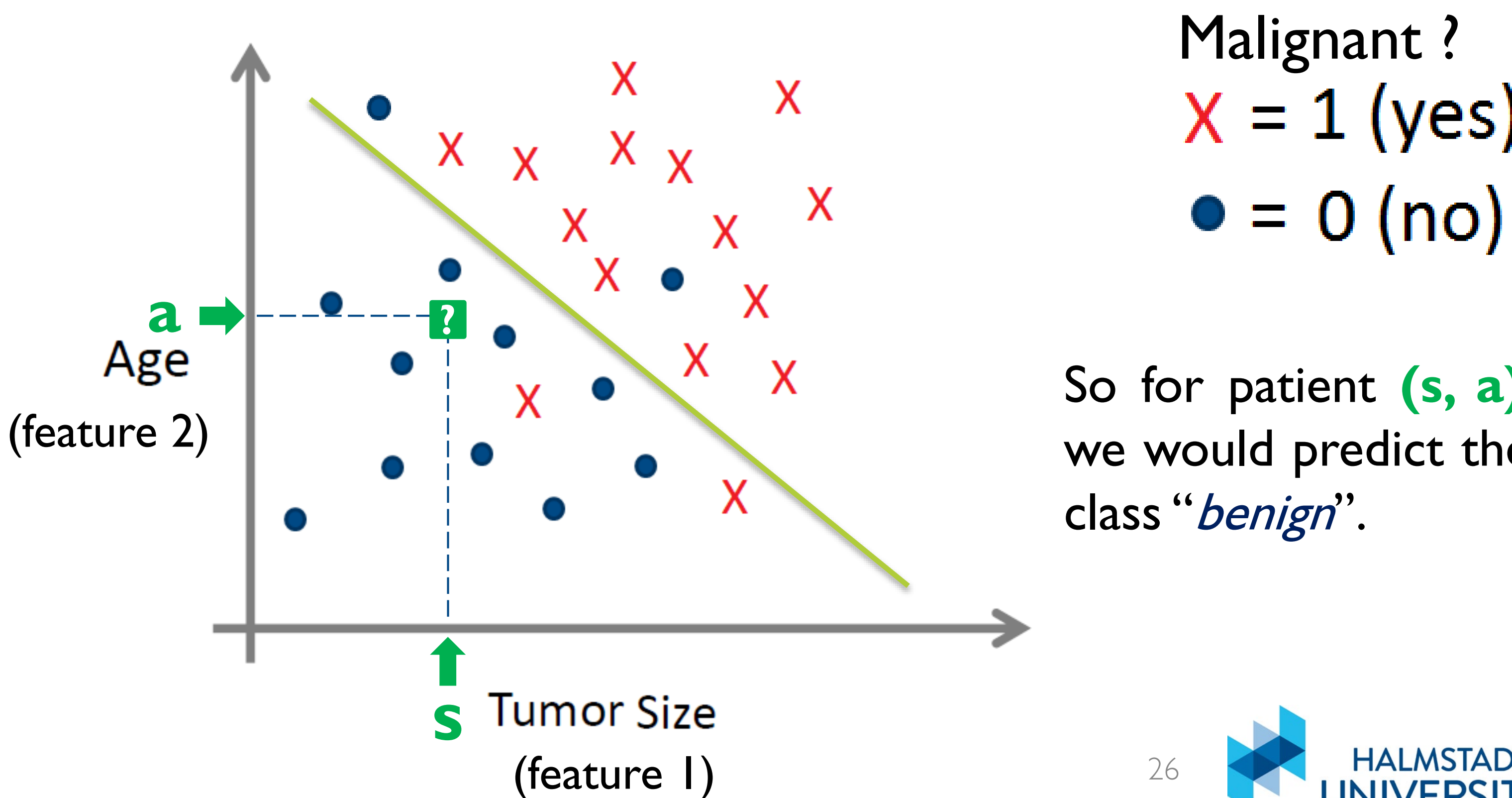
Introduction to Supervised Learning

- Breast cancer malignant/benign (classification)
  - Suppose that you get a new patient who has some tumor size **s** and age **a**, and you want to predict if it is malignant or benign. How can a learning algorithm help you?



Introduction to Supervised Learning

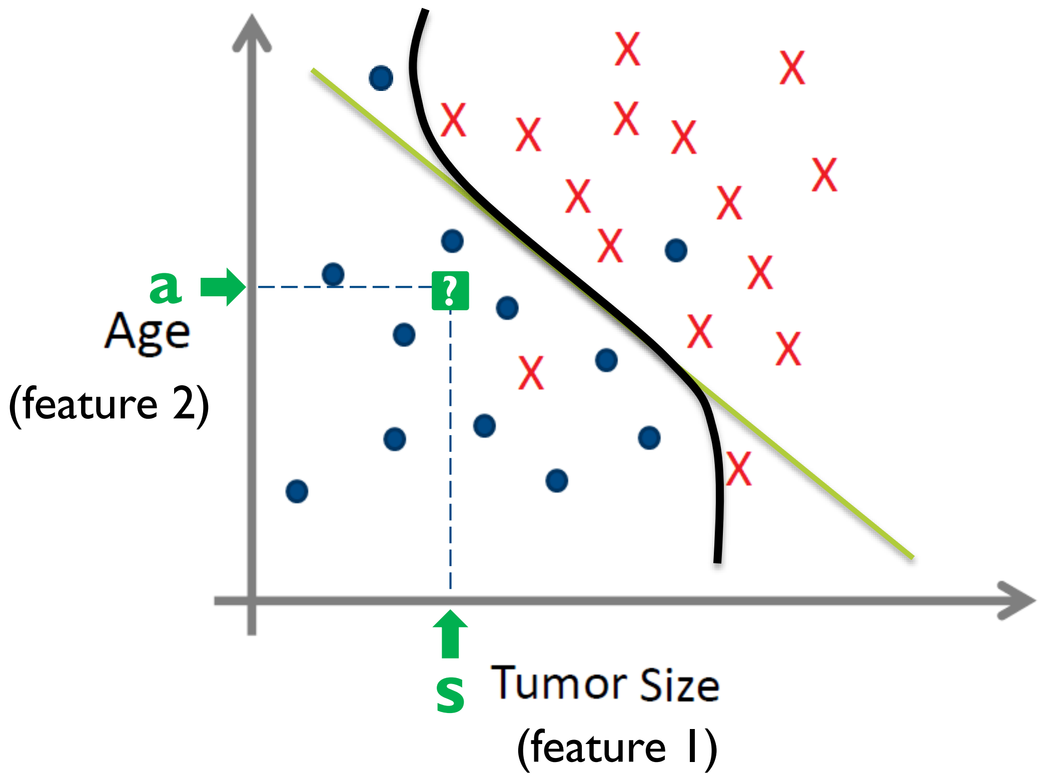
- Breast cancer malignant/benign (classification)
  - you can fit a linear model to the training data, then predict the class of the new patient.



So for patient **(s, a)**, we would predict the class "*benign*".

Introduction to Supervised Learning

- Breast cancer malignant/benign (classification)
  - you can fit a linear model to the training data, then predict the class of the new patient
  - or you can fit a non-linear model to the training data …



Malignant ?

X = 1 (yes)

● = 0 (no)

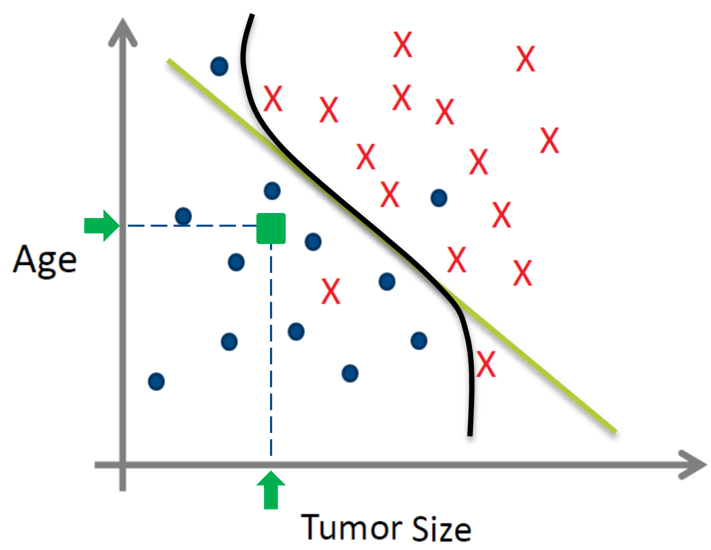
So for patient **(s, a)**, we would predict the class "*benign*".

Introduction to Supervised Learning

- Breast cancer malignant/benign (classification)

❑ Again, this is an example of a **supervised learning** algorithm:
  - The right answers (here, the classes malignant / benign) are given with the training dataset.
  - i.e. for each patient (data-point) in the training dataset, we know if he is has a malignant or benign cancer.

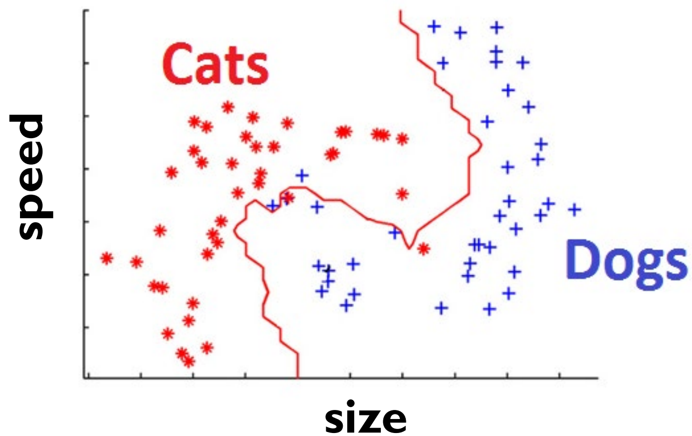
❑ However, this example was a **classification problem**:
  - Predicting a discrete valued output (malignant / benign).

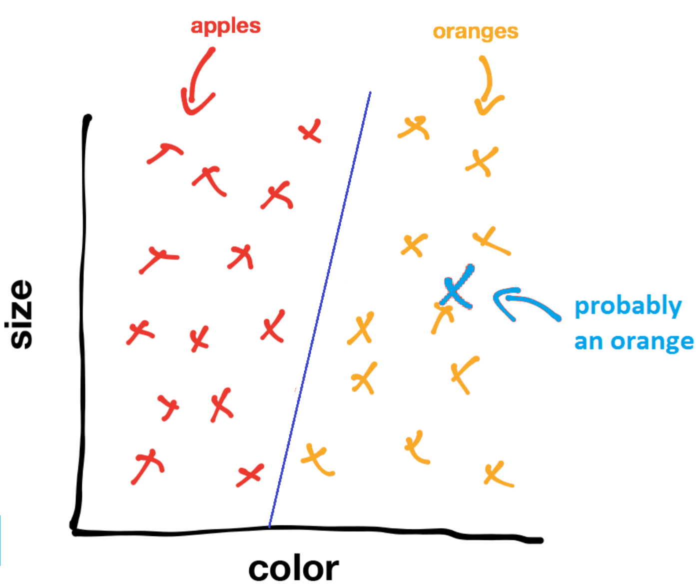


Note: In this example we had two features (age, size), but we will see ML algorithms that can easily deal with a much larger number of features …

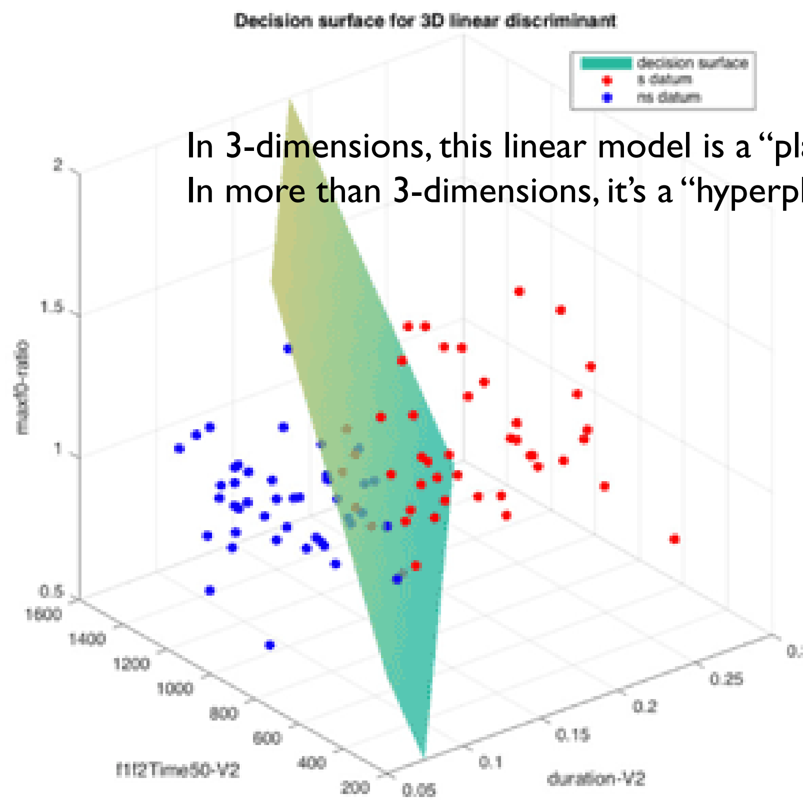
Introduction to Supervised Learning

Other classification examples





- Classification is about learning decision boundaries, and predicting the "class" of new data-point.



In 3-dimensions, this linear model is a "plan".
In more than 3-dimensions, it's a "hyperplan".

HALMSTAD UNIVERSITY

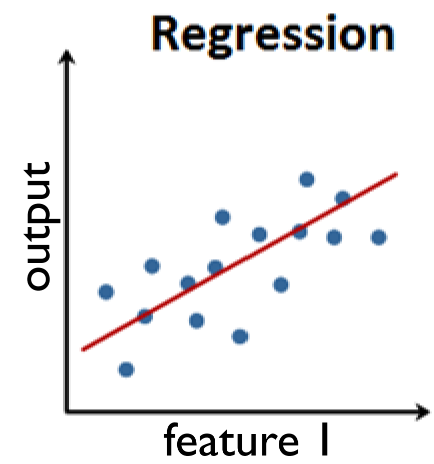
Introduction to Supervised Machine Learning

Difference between Regression and Classification

Difference between Regression and Classification

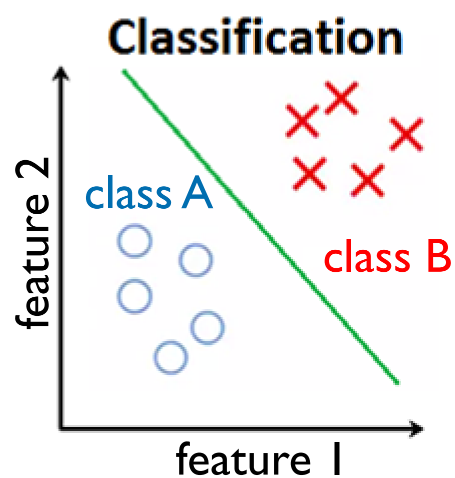
Regression:

- The output (i.e. target variable) is continuous. It consist of real values.
  - predicting the price of houses (in SEK)
  - predicting the power consumption (in kW)
  - predicting how much healthy is the patient (e.g. ∈ [0, 1])
  - etc.



Classification:

- The output (i.e. target variable) is discrete. It consists of classes (or categories).
  - predicting if an image contains a cat or a dog
  - predicting customer categories
  - good/bad, healthy/sick, red/green/blue, A/B/C/D, 0/1/2 …
  - etc.



Difference between Regression and Classification

- You're running a company, and you want to develop machine learning algorithms to address each of two following problems:

- **Problem 1**:
  - You have a large inventory of identical items. You want to predict how many of these items will sell over the next 3 months.

- **Problem 2**:
  - You'd like software to examine individual customer accounts, and for each account decide if it has been hacked/compromised.

- Should you treat these as *classification* or as *regression* problems?
  - Treat both as *classification* problems.
  - Treat *problem 1* as a *classification* problem, *problem 2* as a *regression* problem.
  - Treat *problem 1* as a *regression* problem, *problem 2* as a *classification* problem.
  - Treat both as *regression* problems.

Difference between Regression and Classification

- You're running a company, and you want to develop learning algorithms to address each of two following problems.

- **Problem 1**: The output is the number of items. Time is a feature here.
  - You have a large inventory of identical items. You want to predict how many of these items will sell over the next 3 months.

- **Problem 2**: The output consists of two classes: hacked / not hacked
  - You'd like software to examine individual customer accounts, and for each account decide if it has been hacked/compromised.

- Should you treat these as *classification* or as *regression* problems?
  - ✗ Treat both as *classification* problems.
  - ✗ Treat *problem 1* as a *classification* problem, *problem 2* as a *regression* problem.
  - ✓ Treat *problem 1* as a *regression* problem, *problem 2* as a *classification* problem.
  - ✗ Treat both as *regression* problems.

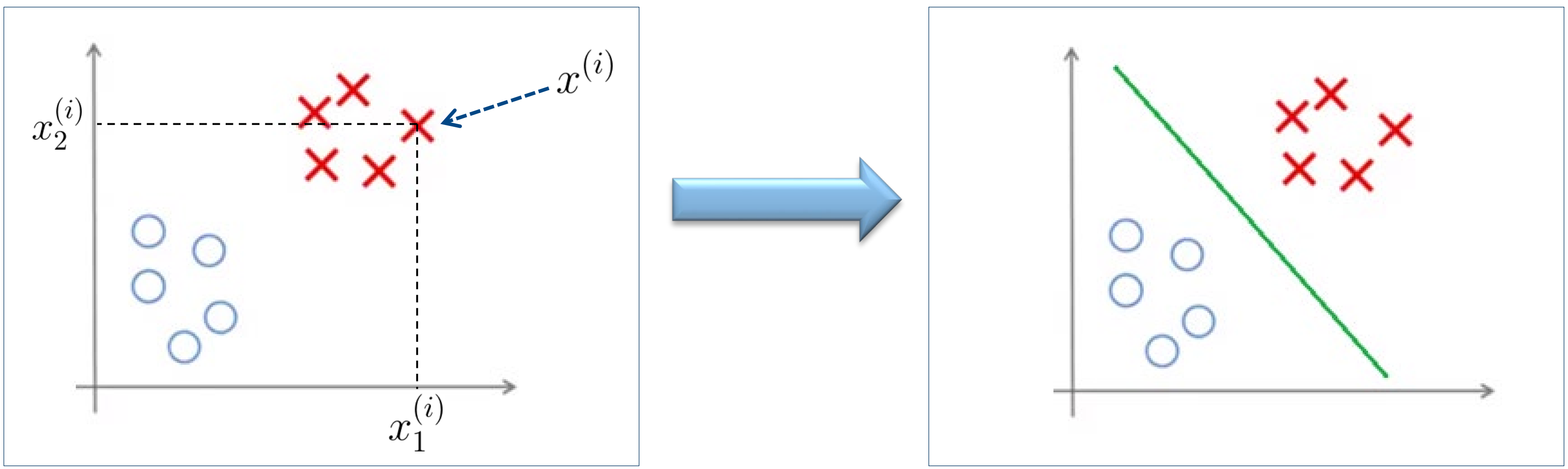
HALMSTAD UNIVERSITY

Introduction to Unsupervised Learning

Introduction to Unsupervised Learning

In **supervised** learning (e.g. classification) we have a **labeled** training dataset:

$$\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(n)}, y^{(n)})\}$$

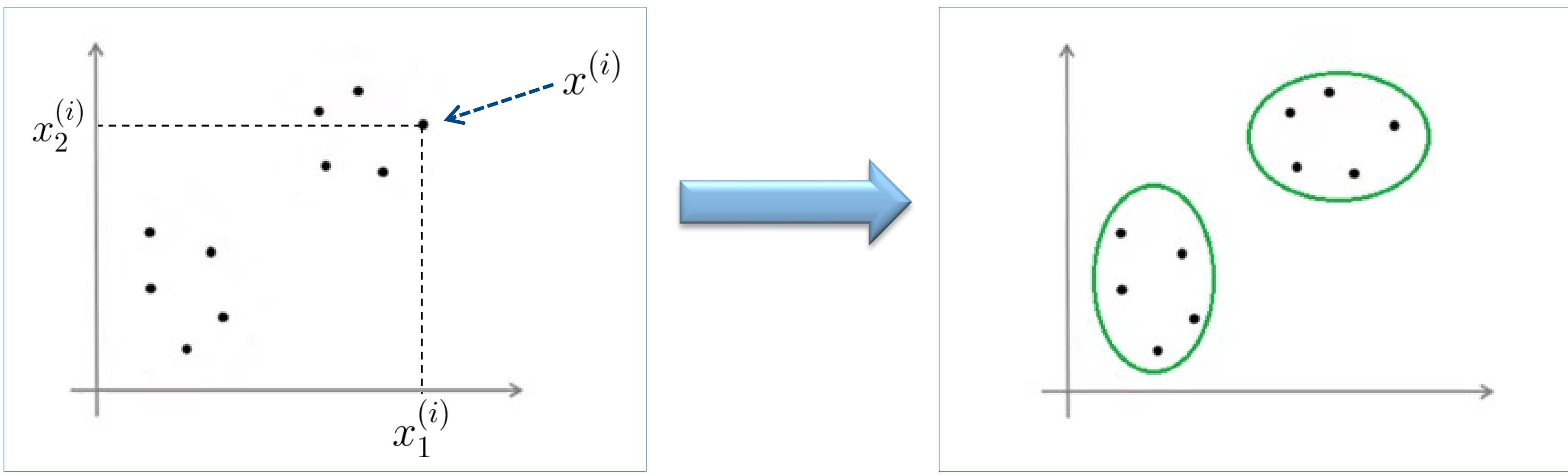


So, for each data-point $x^{(i)} \in R^2$, we have the corresponding class-label $y^{(i)} \in \{ \times, \circ \}$

Introduction to Unsupervised Learning

In **unsupervised** learning (e.g. clustering) we have an **unlabeled** training dataset:

$$\{x^{(1)}, x^{(2)}, \ldots, x^{(n)}\}$$

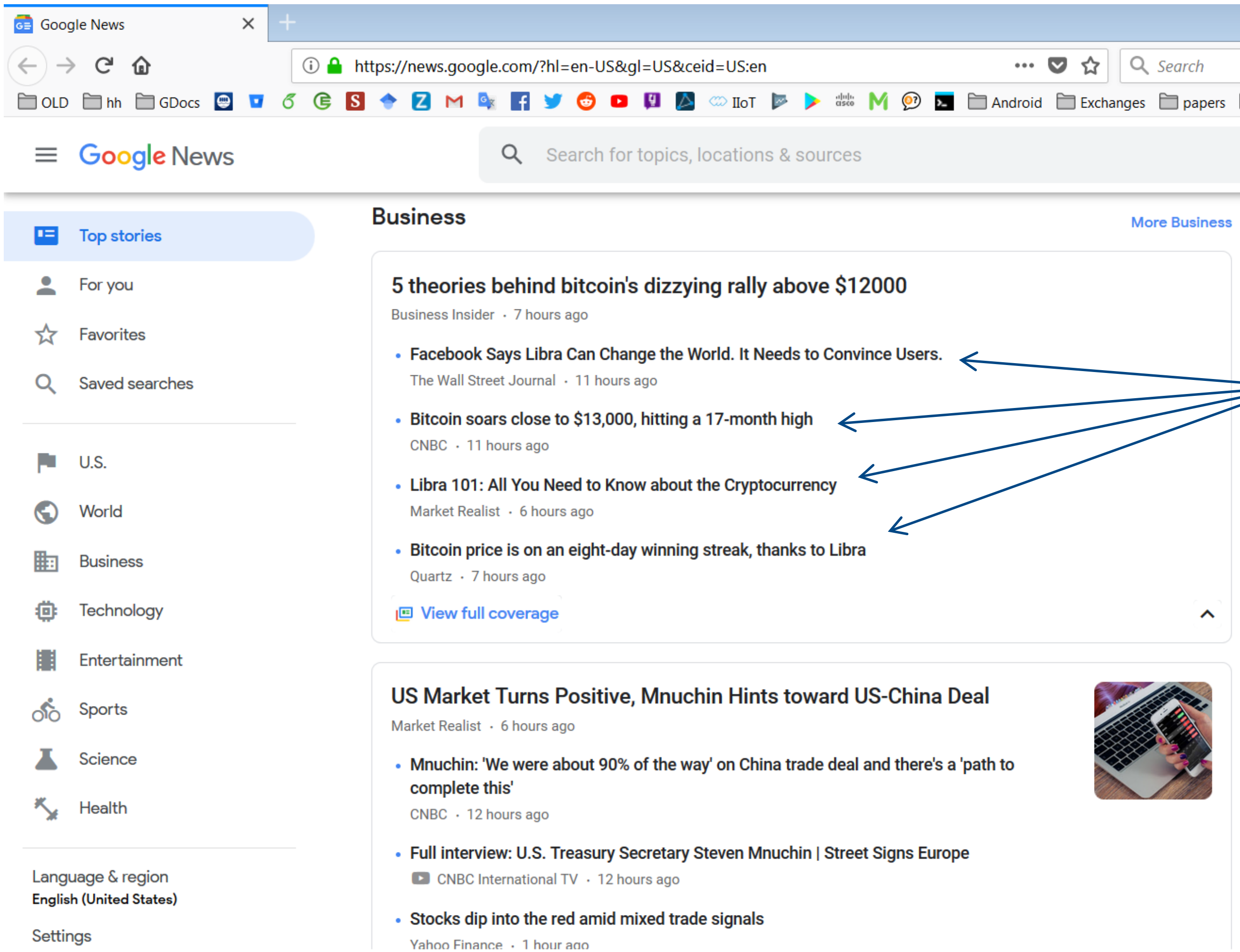


We only have data-points $x^{(i)} \in R^2$

In clustering, we want to explore the data to find some intrinsic groups (clusters) in it. The clusters are not known beforehand.

Introduction to Unsupervised Learning

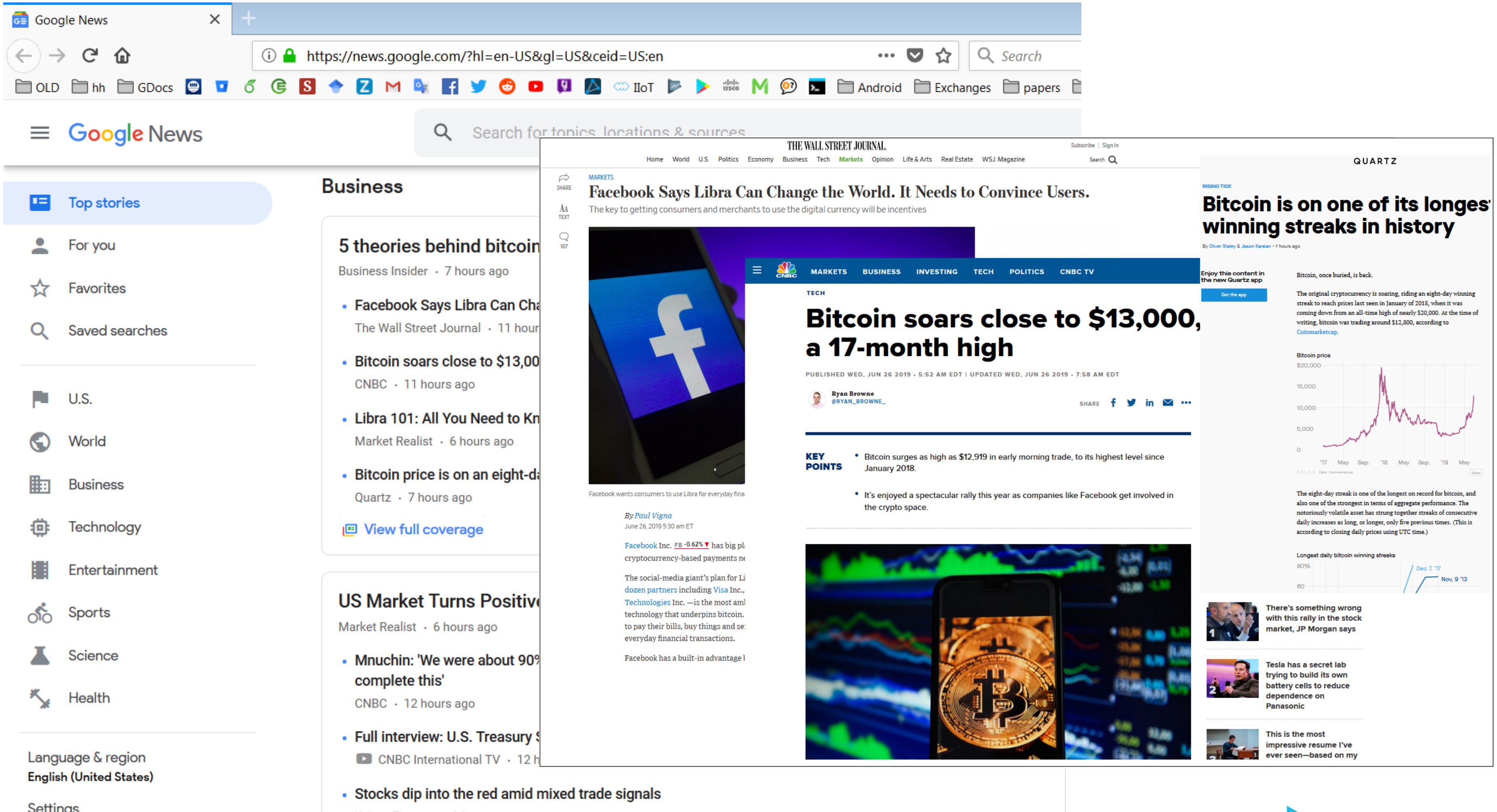
Some applications of clustering



Automatically grouping together the stories (news articles on the Web) that talk about the same topic.

Introduction to Unsupervised Learning

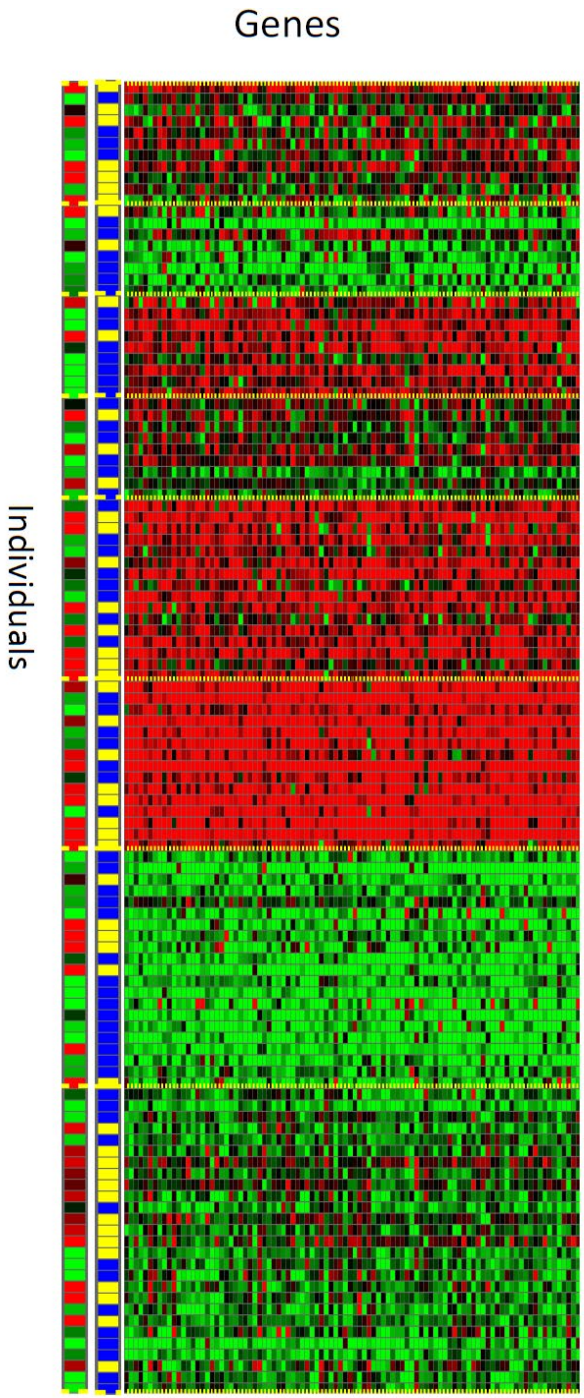
Some applications of clustering



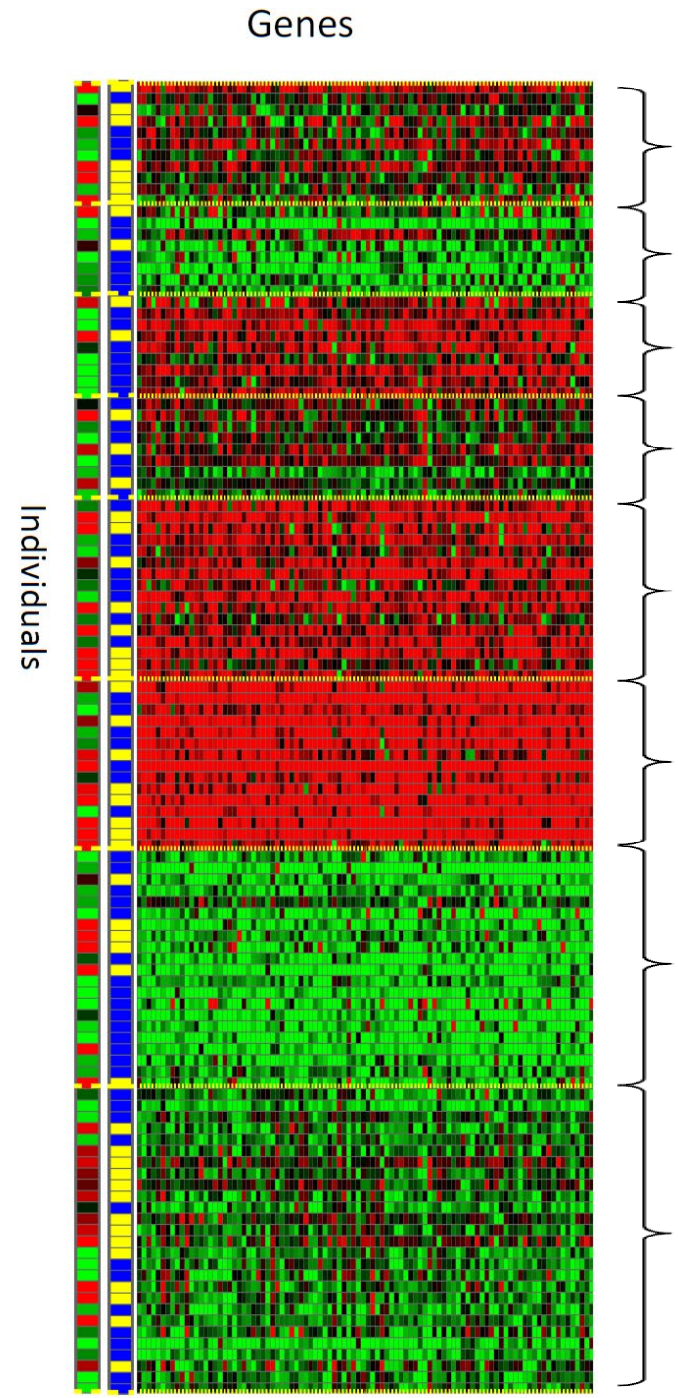
Introduction to Unsupervised Learning
Some applications of clustering

- DNS Microarray data.

- Colors here corresponds to how much individuals do or do not have a certain gene.



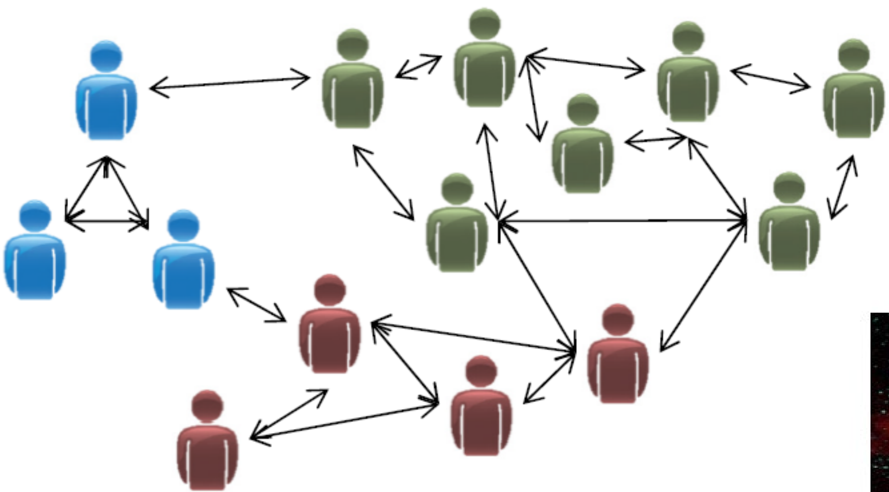
Introduction to Unsupervised Learning
Some applications of clustering



- DNS Microarray data.
- Colors here corresponds to how much individuals do or do not have a certain gene.
- Run a clustering algorithm to group individuals into different groups/types of people.

Introduction to Unsupervised Learning

Some applications of clustering



Social network analysis

Astronomical data analysis

Introduction to Unsupervised Learning

Of the following examples, which would you address using an <u>unsupervised</u> learning algorithm? (Check all that apply.)

- [ ] Given email labeled as spam/not spam, learn a spam filter.
- [ ] Given a set of news articles found on the web, group them into set of articles about the same story.
- [ ] Given a database of customer data, automatically discover market segments and group customers into different market segments.
- [ ] Given a dataset of patients diagnosed as either having diabetes or not, learn to classify new patients as having diabetes or not.

Introduction to Unsupervised Learning

Of the following examples, which would you address using an <u>unsupervised</u> learning algorithm? (Check all that apply.)

- ☐ Given email labeled as spam/not spam, learn a spam filter.
- ✓ Given a set of news articles found on the web, group them into set of articles about the same story.
- ✓ Given a database of customer data, automatically discover market segments and group customers into different market segments.
- ☐ Given a dataset of patients diagnosed as either having diabetes or not, learn to classify new patients as having diabetes or not.

Course contents and format

Course contents and format

- Week 4 (Basics)
  - **Lecture 1.1** Introduction to machine learning (this lecture).
  - **Lecture 1.2** Basics, prerequisite, and review of important notions.
  - **Lab 1**: Hands-on Python for ML
- Week 5 (Regression)
  - **Lecture 2.1** Linear Regression.
  - **Lecture 2.2** Nonlinear Regression (KNN and Kernel regression)
  - **Lab 2**: Implementing linear regression (with/without gradient descent) + Kernel regression.
- Week 6 (Classification)
  - **Lecture 3.1** Classification using Logistic Regression.
  - **Lecture 3.2** Nonlinear Classification (Polynomial features, KNN, DTrees, …)
  - **Lab 3**: Implementing logistic regression + KNN.
- Week 7 (Generalization)
  - **Lecture 4.1** Overfitting and Regularization.
  - **Lecture 4.2** Ensemble Methods (Random Forest).
  - **Lab 4**: Re-implementing LinReg and LogisticReg with Regularization + Implementing Random Forest.
- Week 8 (SVM)
  - **Lecture 5** Support Vector Machines.
  - **Lab 5**: Using SVM (Linear + with Kernel Trick) for Spam Classification.
- Week 9 (ANN)
  - **Lecture 6.1** Artificial Neural Networks (ANN).
  - **Lecture 6.2** Artificial Neural Networks (ANN) – Continuation.
  - **Lab 6**: Implementing a simple ANN.
- Week 10 (Unsupervised Learning)
  - **Lecture 7.1** Dimensionality Reduction (using Principal Components Analysis)
  - **Lecture 7.2** Clustering
  - **Lab 7**: Implementing PCA + K-means clustering.
- Week 11 (Presentations)
  - Seminars where your present your projects …

Course contents and format

1. **Written examination** (3 credits)
   - Mainly based on the contents of lectures.
   - and some content related to the Labs.

2. **Practical Projects and Labs** (3 credits)
   - The weekly Labs (jupyter notebooks).
     - The Labs are to be done individually.
   - Written **report** about the final project (to submit before week 11).
     - The final project can be done in a group of one or two students (maximum).

3. **Seminars** (1.5 credits)
   - Oral presentation of the final project (on week 11)
     - The presentation (in a group of one or two students) should take 20 to 25 minutes max.
     - The slides should show the project results achieved so far as well as a state-of-the-art section which refers to research articles/papers related to your project (use Google Scholar to find relevant papers).

Course contents and format

- The report should be about 7 to 10 pages including figures and tables). It can be structured as follows:
  1. **Introduction**
     - Brief presentation of problem, 1 page.
  2. **State-of-the-art**
     - Brief description of research papers doing work related to your project, 1 page.
  3. **Methodology**
     - Brief listing of methods used, 1 page.
  4. **Data**
     - Presentation of your dataset with important observations, 1-2 pages.
  5. **Results and their interpretation** (3-5 pages).
  6. **Discussion**
     - Conclusions about your results and comparison to other researchers' results, 1 page.

Course contents and format

- Regarding Labs:
  - There is a Lab to do on each week (total of 7 Labs).
  - You have to start working on each Lab soon after the corresponding lecture (i.e. before the Lab session) and prepare questions for the Lab assistant who will help you during the Lab session.
  - The deadlines to submit each Lab are on Blackboard.
  - Submit your Lab solutions as jupyter notebooks to the Lab assistants: Reza reza.khoshkangini@hh.se and Yuantao yuantao.fan@hh.se and add Rafik (as cc) mohamed-rafik.bouguelia@hh.se

- Regarding Projects:
  - You have to submit your written report before week 11 to mohamed-rafik.bouguelia@hh.se