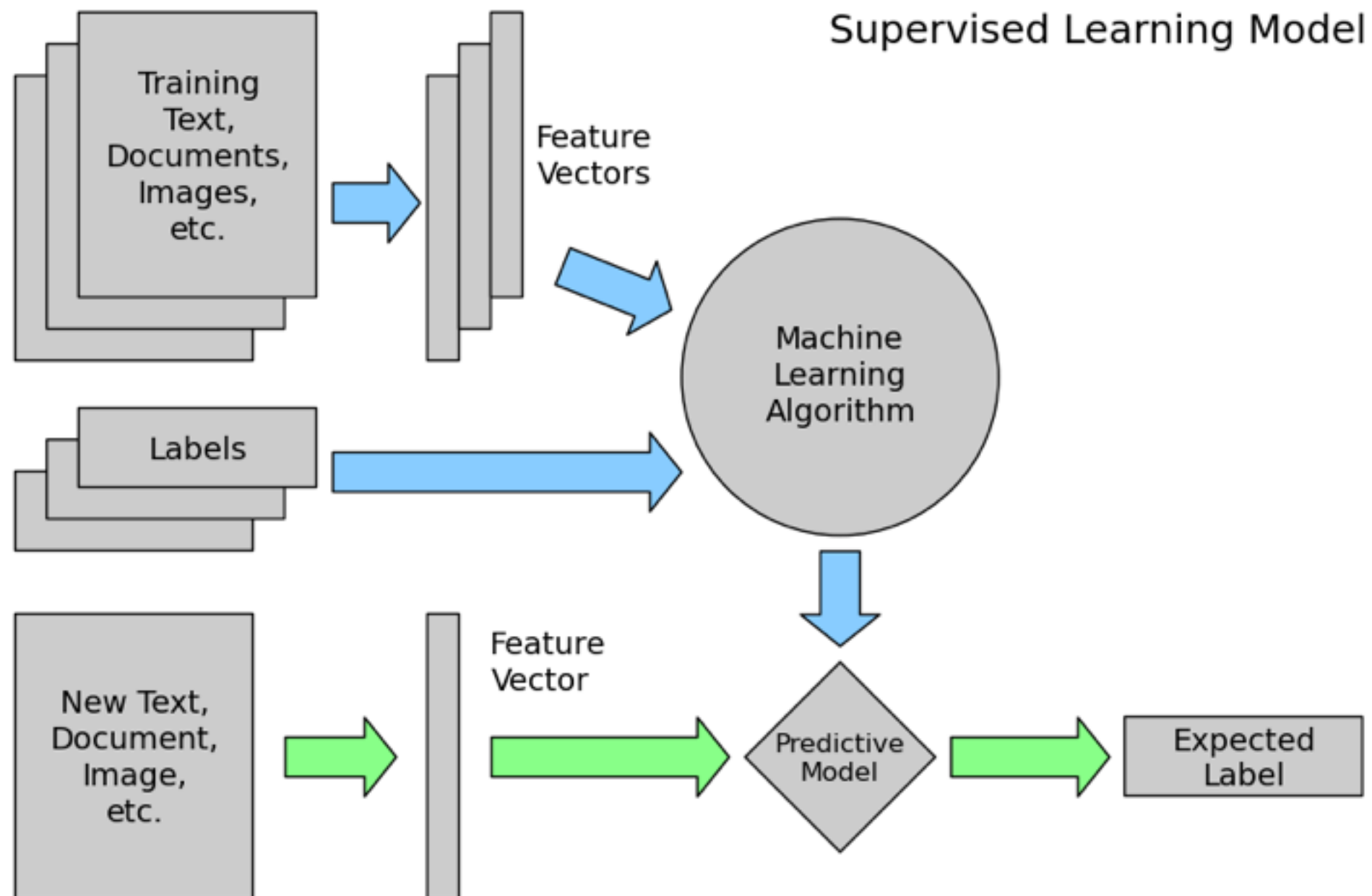# Intro to prediction problems

Prediction problem statement

- Data set $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ..., (\mathbf{x}_n, y_n)\}$
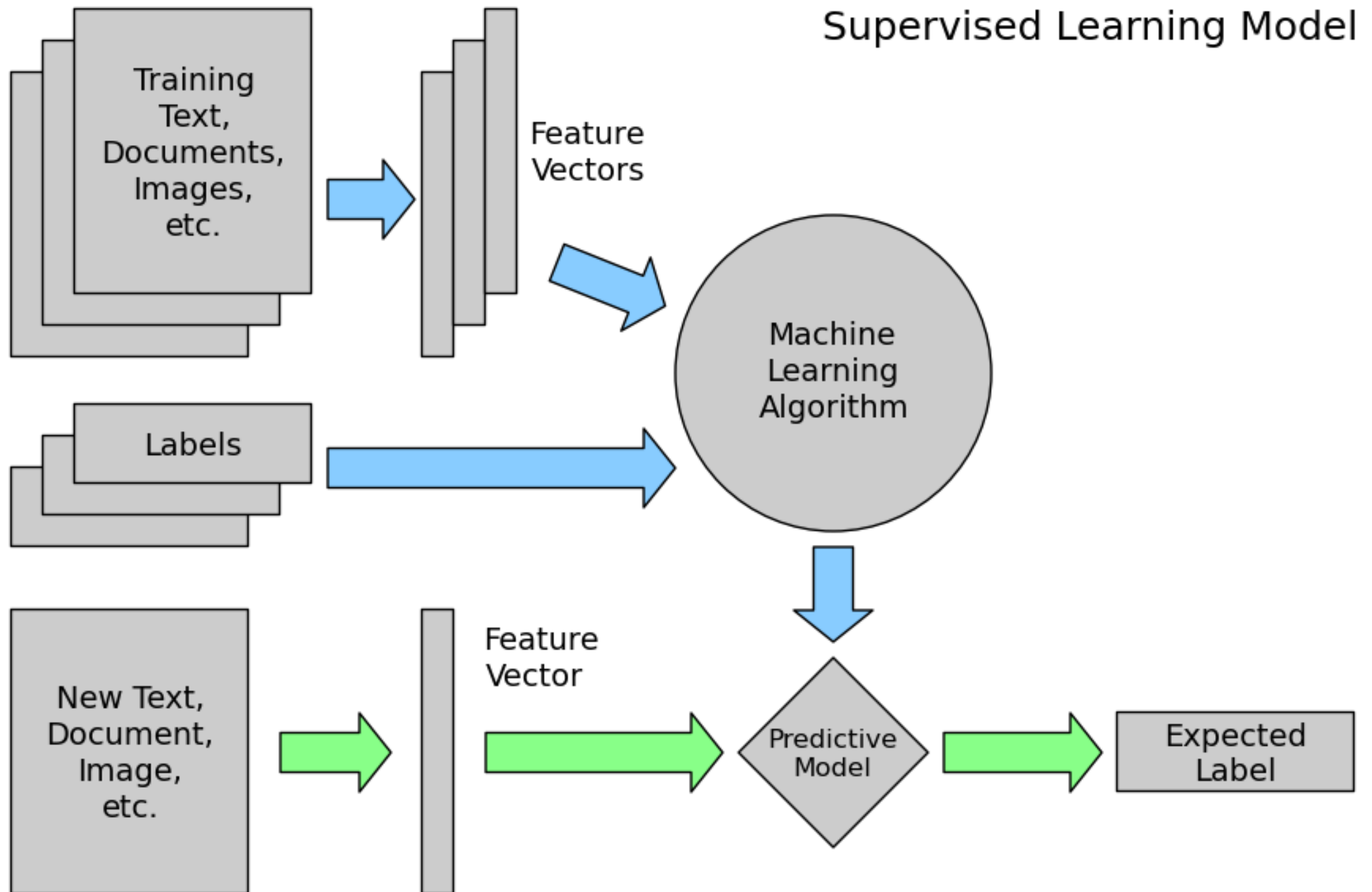
  $\mathbf{x}_i \in \mathcal{X}$ is the i-th object and $y_i \in \mathcal{Y}$ is the corresponding target designation

  We usually assume that $\mathcal{X} = \mathbb{R}^d$, in which case $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ik})$

- Each xi a data point or sample

- Each dimension of xi is called a feature or attribute

- Underlying assumption: features easy/easier to obtain and targets are difficult to observe or collect

Supervised learning

Supervised Learning Model

Training Text, Documents, Images, etc. → Feature Vectors → Machine Learning Algorithm

Labels → Machine Learning Algorithm

New Text, Document, Image, etc. → Feature Vector → Predictive Model → Expected Label

3

*from http://www.astroml.org/sklearn_tutorial/general_concepts.html

Types of predictions

- The target could be anything; convenient to separate into different types (even though can have related approaches)

- Generally two main types discussed

  - classification, e.g. $\mathcal{Y} = \{\text{sports, medicine, travel}, \dots\}$.

  - regression, e.g. $\mathcal{Y} = \mathbb{R}$.

- Structured output often a type of classification problem

- Can have structured output for regression as well

Example: binary classification

	wt [kg]	ht [m]	T [°C]	sbp [mmHg]	dbp [mmHg]	y
$\mathbf{x}_1$	91	1.85	36.6	121	75	-1
$\mathbf{x}_2$	75	1.80	37.4	128	85	$+1$
$\mathbf{x}_3$	54	1.56	36.6	110	62	-1

Table 3.1: An example of a binary classification problem: prediction of a disease state for a patient. Here, features indicate weight (wt), height (ht), temperature (T), systolic blood pressure (sbp), and diastolic blood pressure (dbp). The class labels indicate presence of a particular disease, e.g. diabetes. This data set contains one positive data point ($\mathbf{x}_2$) and two negative data points ($\mathbf{x}_1$, $\mathbf{x}_3$). The class label shows a disease state, i.e. $y_i = +1$ indicates the presence while $y_i = -1$ indicates absence of disease.

Example: regression

	size [sqft]	age [yr]	dist [mi]	inc [$]	dens [ppl/mi^2]	y
$\mathbf{x}_1$	1250	5	2.85	56,650	12.5	2.35
$\mathbf{x}_2$	3200	9	8.21	245,800	3.1	3.95
$\mathbf{x}_3$	825	12	0.34	61,050	112.5	5.10

Table 3.2: An example of a regression problem: prediction of the price of a house in a particular region. Here, features indicate the size of the house (size) in square feet, the age of the house (age) in years, the distance from the city center (dist) in miles, the average income in a one square mile radius (inc), and the population density in the same area (dens). The target indicates the price a house is sold at, e.g. in hundreds of thousands of dollars.

Other examples

- In general, as with much of machine learning, there are no strict rules: the way you specify a problem can be an art

    - learning can be strongly impacted by specification

- E.g: multi-label classification problem $\mathcal{Y} = \{\text{sports}, \ \text{medicine}, \ \text{travel}, \ldots\}$.

    - could specify as high-dimensional multi-class classification

    $$\mathcal{Y} = \mathcal{P}(\{\text{sports}, \ \text{medicine}, \ \text{travel}, \ldots\})$$

    - could specify as a regression problem

    - could try to learn hierarchical tree structure

- A key part of ML is not only knowing algorithms, but also how to formalize (prediction) problems

Thought exercise: specifying prediction problems

- Imagine someone has given you a database of samples

  - e.g., Netflix data of people's movie rankings

- What do you need to consider before starting to learn a predictor? (without knowing much about algorithms yet)

  - How do I know if my predictions are successful? What is my measure?

  - How many samples are there? Is it a large database?

  - Is efficiency important?

  - What simple algorithms can I try first?

  - Is the data useful? Could it be significantly improved with different and/or more data collection?

  - How should I represent my prediction problem?
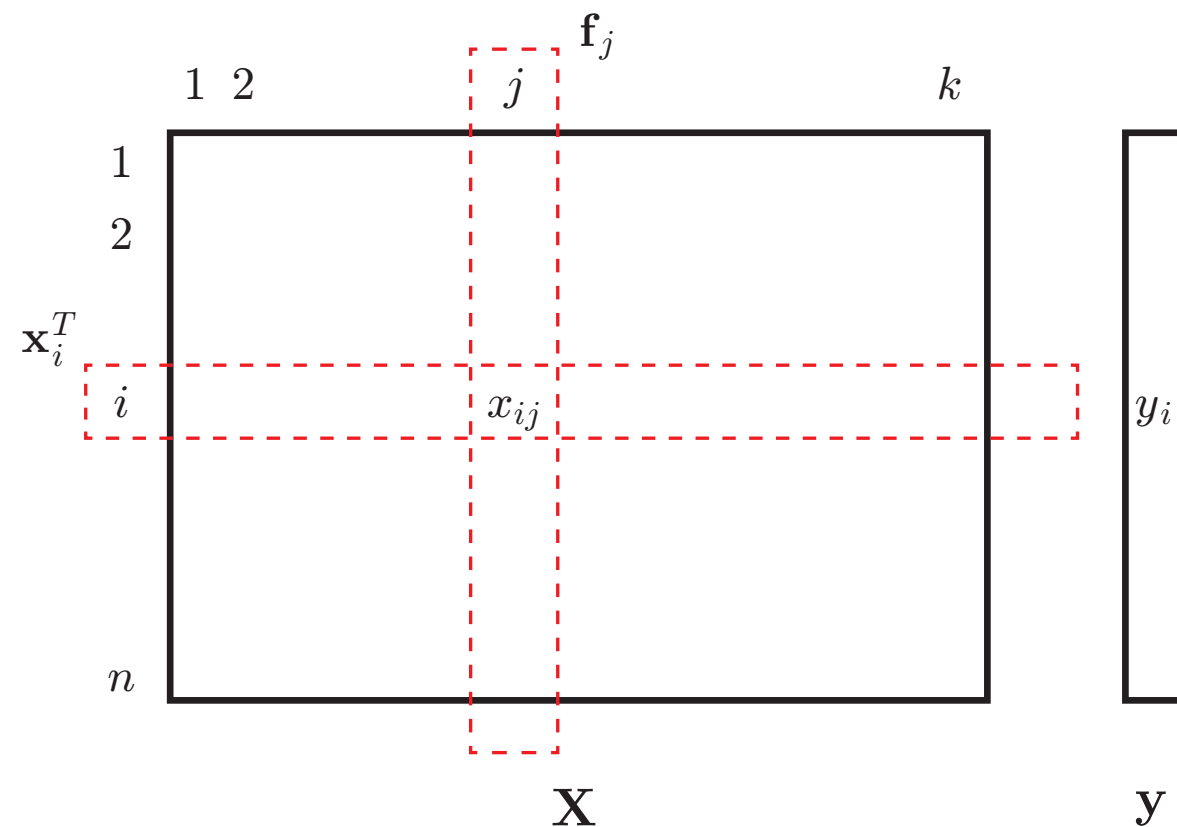
Vector and matrix notation

Figure 3.1: Data set representation and notation. $\mathbf{x}$ *is an n-by-k matrix representing features and data points, whereas y is an n-by-1 vector of targets.*

Optimal prediction

- We want to learn a prediction function

- We will need to define cost/error of a prediction

  - Cost function $\quad c : \mathcal{Y} \times \mathcal{Y} \to [0, \infty)$

  - For true target y, get cost $\quad c(\hat{y}, y)$

- We will see that modeling p(y | x) is useful for this task

- Want to find predictor that minimizes the expected cost

  - could choose other metrics, such as minimize number of costs that are very large or minimize the maximum cost

Whiteboard

- Expected cost

- Next topic: linear regression