TRAINING, DEVELOPMENT, TESTING SET

Big Data
STF1724

Rini Nuraini Sukmana, M.T
0020087901 - 08882024236
rini.nuraini@usbypkp.ac.id

TRAINING, DEVELOPMENT, TESTING SET

- *Training* adalah proses membangun model
- *Testing* adalah proses menguji kinerja model pembelajaran.
- *Dataset* adalah kumpulan data (sampel dalam statistik).
- Sampel ini adalah data yang kita gunakan untuk membuat model maupun mengevaluasi model *machine learning*.
- Umumnya, *dataset* dibagi menjadi tiga jenis yang tidak beririsan (satu sampel pada himpunan tertentu tidak muncul pada himpunan lainnya):

DATA SET

- ***Training set*** adalah himpunan data yang digunakan untuk melatih atau membangun model.
- ***Development set*** atau ***validation set*** adalah himpunan data yang digunakan untuk mengoptimisasi saat melatih model.
- Model dilatih menggunakan *training set* dan pada umumnya kinerja **saat latihan** diuji dengan *development set*. Hal ini berguna untuk generalisasi (agar model mampu mengenali pola secara generik).
- ***Testing set*** adalah himpunan data yang digunakan untuk menguji model setelah **proses latihan selesai**.

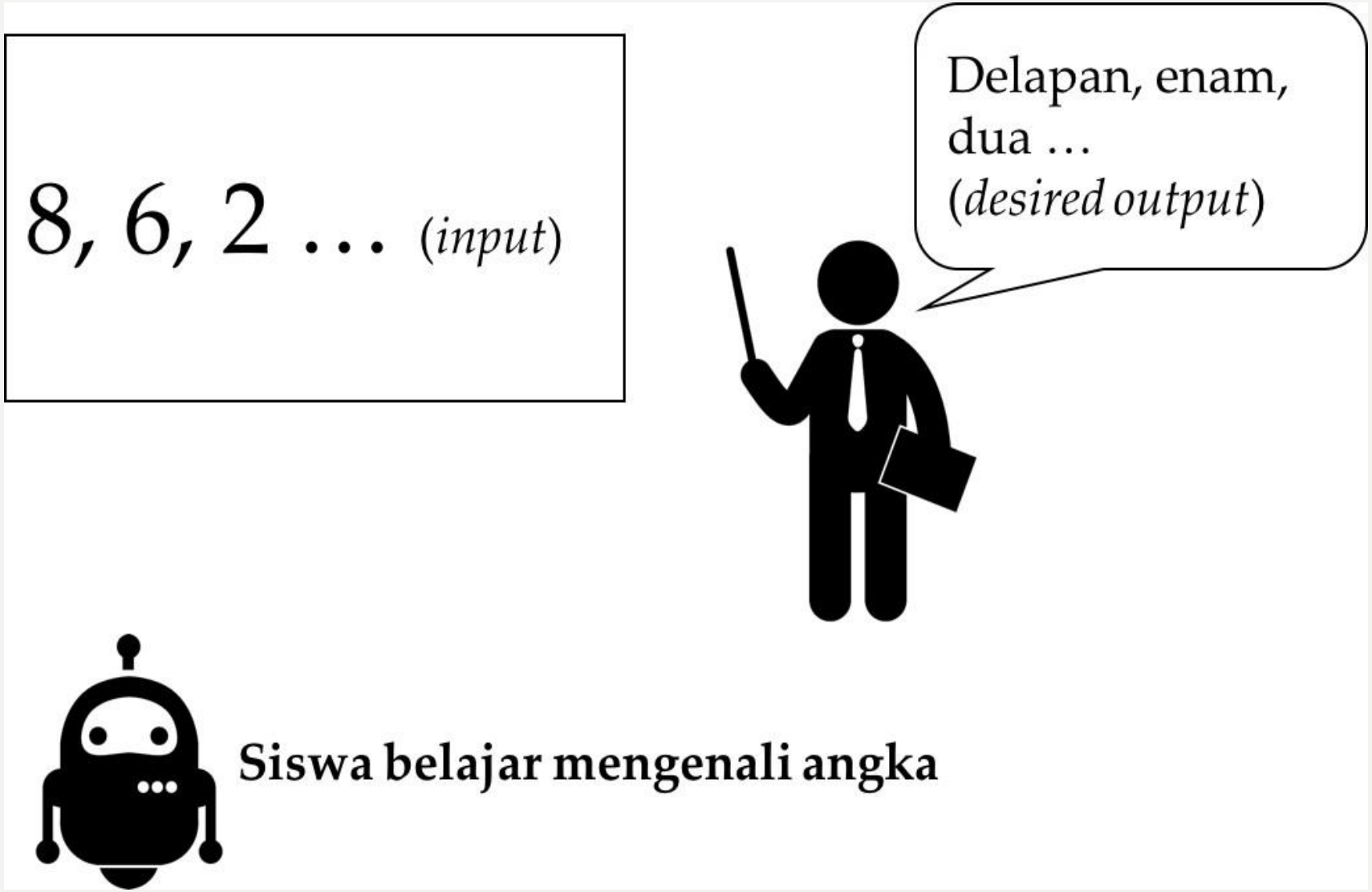
- Satu sampel pada himpunan data kita sebut sebagai ***data point*** atau instans (***instance***) yang merepresentasikan suatu kejadian statistik (*event*).
- Perlu diingat, *training*, *development*, dan *testing data* diambil (*sampled*) dari distribusi yang sama dan memiliki karakteristik yang sama (*independently and identically distributed*).
- Distribusi pada masing-masing dataset ini juga sebaiknya seimbang (*balanced*) dan memuat seluruh kasus.
- Misal, sebuah dataset *binary classification* sebaiknya memuat 50% kasus positif dan 50% kasus negatif.

- Pada umumnya, rasio pembagian *dataset* adalah (80% : 10% : 10%) atau (90% : 5% : 5%) (*training* :*development* :*testing*).
- *Development set* pada umumnya bisa tidak digunakan apabila *dataset* berukuran kecil (hanya dibagi menjadi *training* dan *testing set* saja).
- Dalam kasus ini, pembagian *dataset* menjadi *training* dan *testing* set pada umumnya memiliki rasio (90% : 10%), (80% : 20%), (70% : 30%), atau (50% : 50%).
- Pada kasus ini, kinerja saat *training* diuji menggunakan *training set* (dikenal sebagai ***closed testing***).

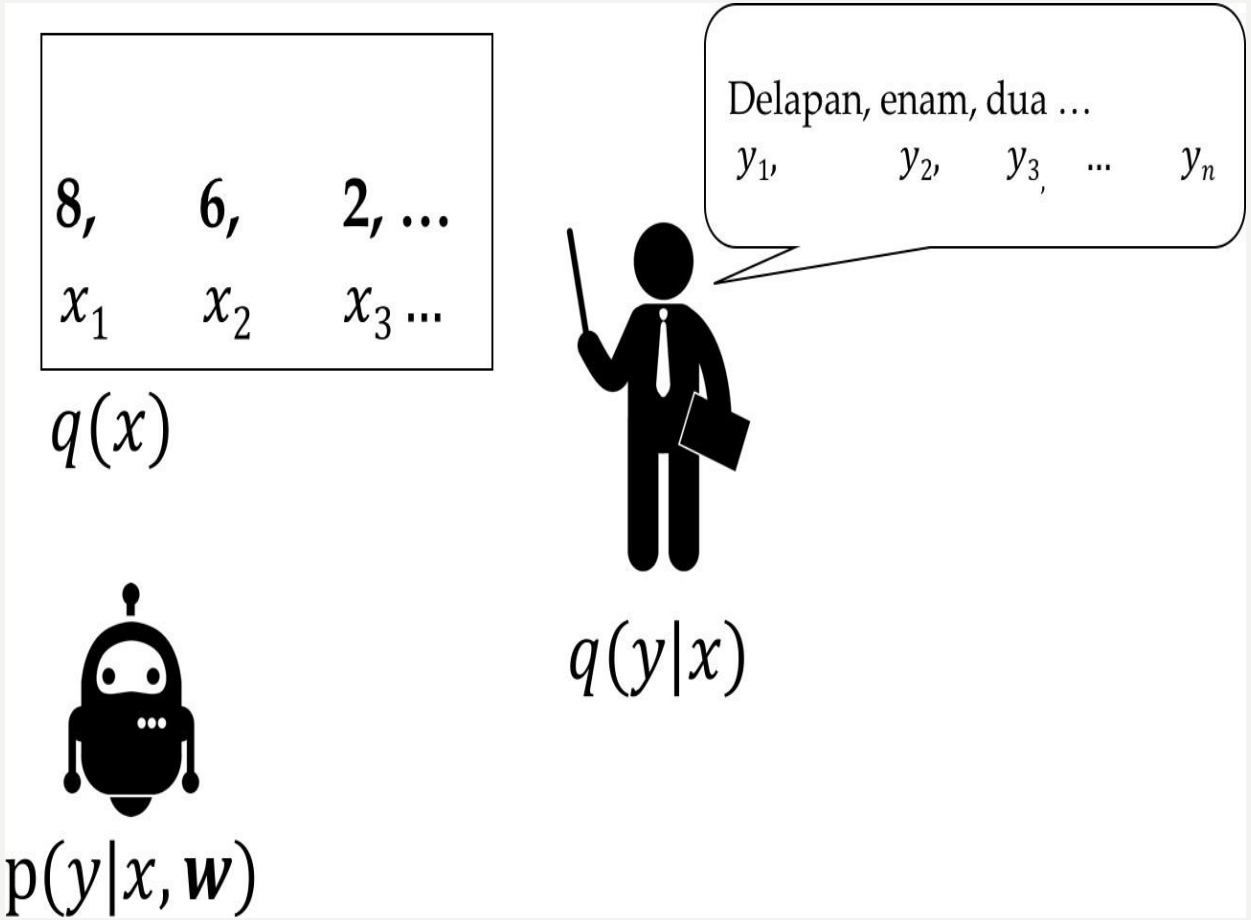
Supervised Learning

- *Supervised learning* adalah pembelajaran terarah/terawasi.
- Artinya, pada pembelajaran ini, ada guru yang mengajar (mengarahkan) dan siswa yang diajar.
- Kita di sini berperan sebagai guru, kemudian mesin berperan sebagai siswa.
- Seorang guru menuliskan angka di papan "8, 6, 2" sebagai contoh untuk siswanya, kemudian gurunya memberikan cara membaca yang benar untuk masing-masing angka.
- Contoh angka melambangkan ***input***, kemudian cara membaca melambangkan ***desired output***.
- Pasangan ***input–desired output*** ini disebut sebagai *instance* (untuk kasus *supervised learning*).
- Pembelajaran metode ini disebut *supervised* karena ada yang memberikan contoh jawaban (*desired output*).

SUPERVISED LEARNING

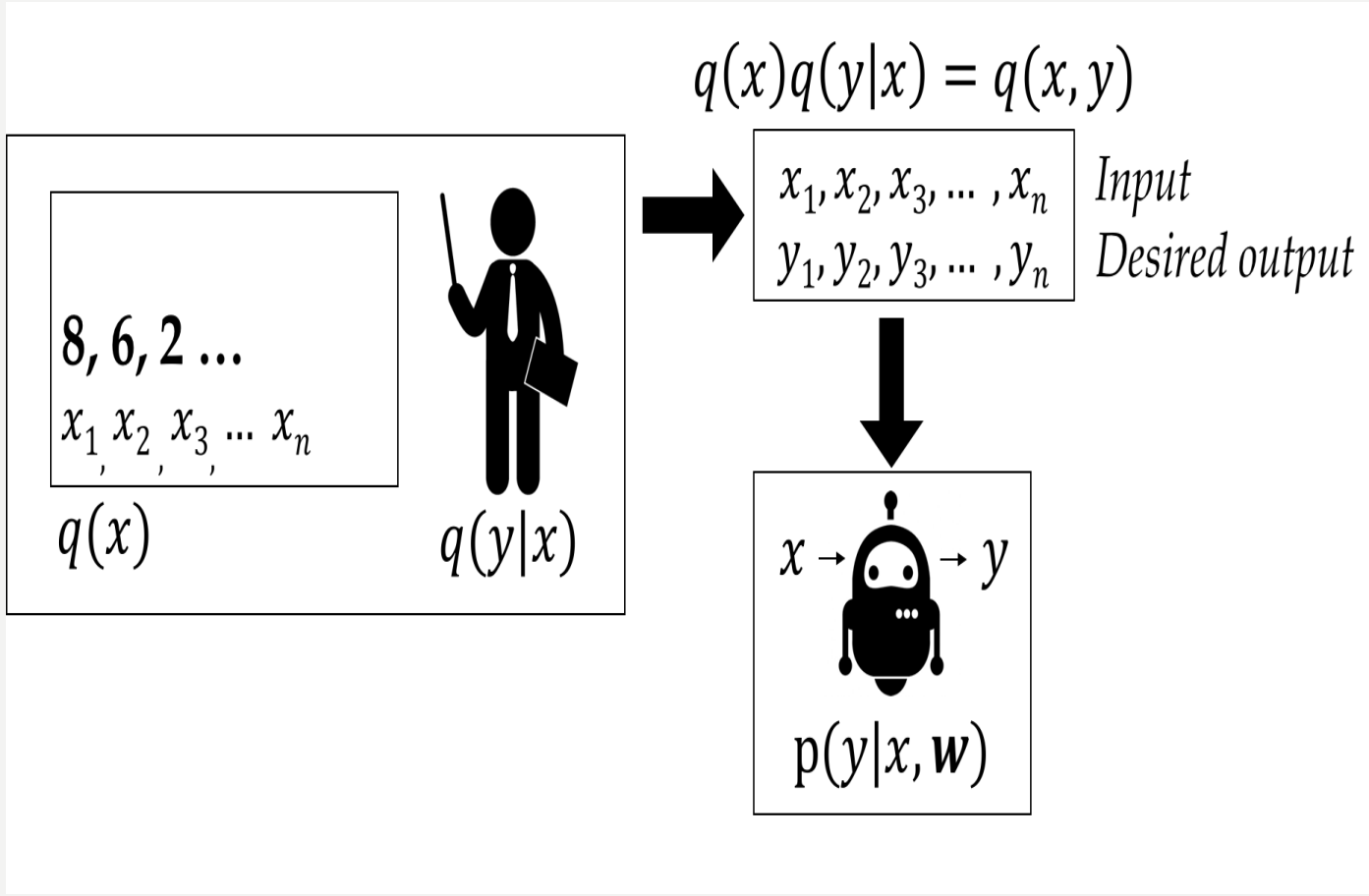


Supervised learning - mathematical explanation



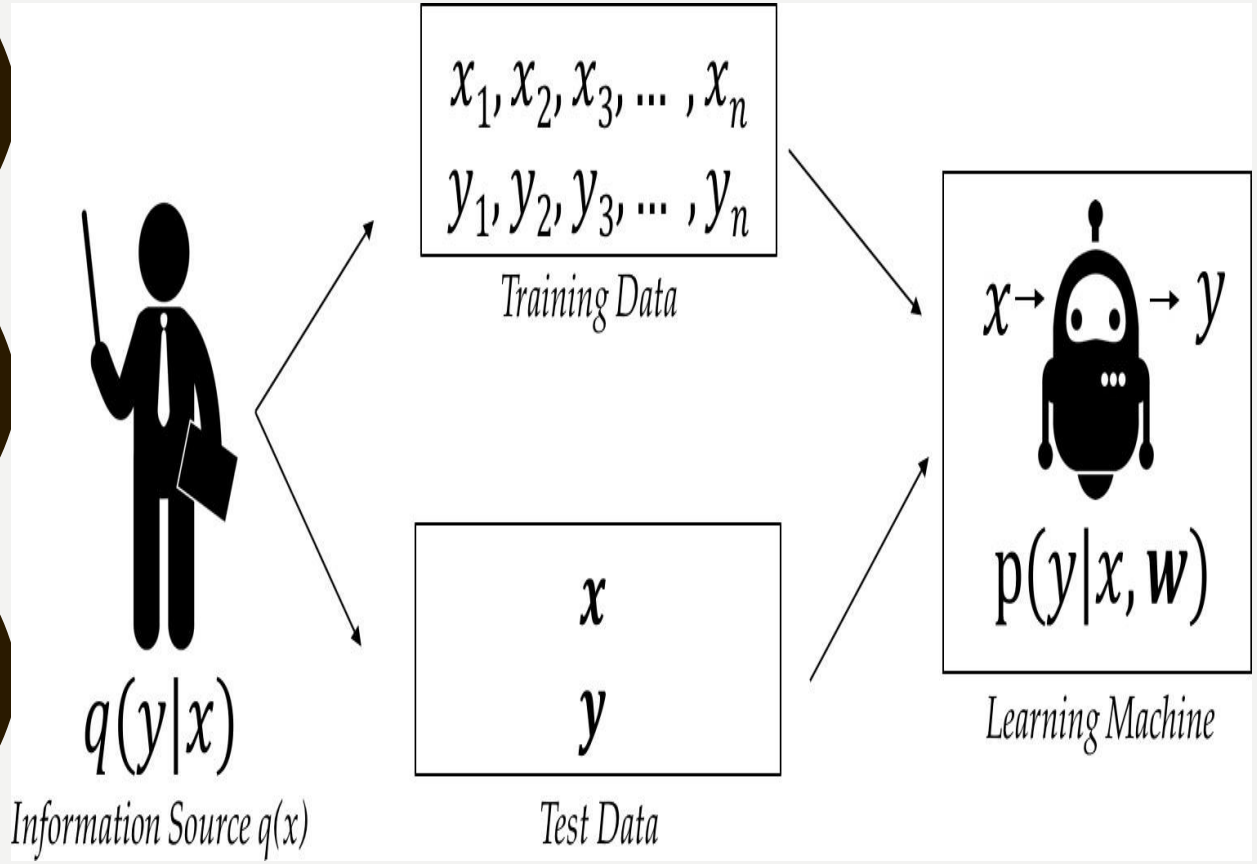
- *x* adalah kejadian (*event* – *random variable*)
- untuk *event* tertentu dapat dinotasikan sebagai $\{x_1, x_2, x_3, \cdots, x_N$.
- *x* dapat berupa vektor, teks, gambar, dan lain sebagainya
- *x* yang merepresentasikan *event*, *data point*, atau *input*.

Supervised learning - mathematical explanation 2



- Seorang guru sudah mempunyai jawaban yang benar untuk masing-masing contoh dengan suatu fungsi distribusi probabilitas kondisional (***conditional probability density function***) *q*(*y x*)
- *q*(*y x*) baca: *function q for y given x*, melambangkan hasil yang benar/diharapkan untuk suatu event.

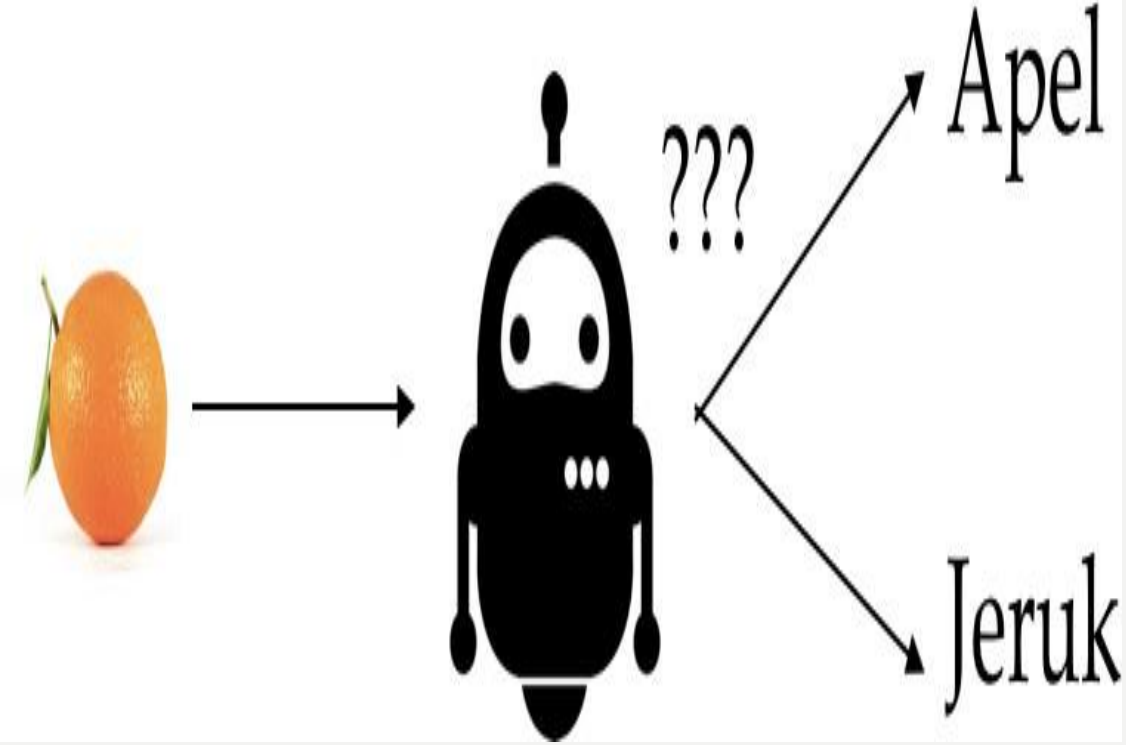
Supervised learning framework (3)



- Siswa (mesin) **mempelajari tiap pasang pasangan *input-desired output (training data)*** dengan mengoptimalkan *conditional probability density function* $p(y \mid x, \mathbf{w})$,
- dimana y adalah target (*output*), x adalah input dan vektor **w** adalah ***learning parameters***.
- Proses belajar ini, yaitu mengoptimalkan **w** disebut sebagai training.
- Proses *training* bertujuan untuk mengaproksimasi $q(y \mid x)$ melalui $p(y \mid x, \mathbf{w})$.

- Perhatikan Gambar 3 model memiliki panah ke *training data* dan *test data*,
- Artinya model hasil *training* sangat bergantung pada **data dan guru**.
- Model yang dihasilkan *training* (hasil pembelajaran kemampuan siswa) untuk data yang sama bisa berbeda untuk guru yang berbeda.
- Tujuan *supervised learning*, secara umum untuk melakukan klasifikasi (*classification*).
- Misalkan mengklasifikasikan gambar buah (apa nama buah pada gambar),
- Apabila hanya ada dua kategori, disebut ***binary classification***.
- Sedangkan bila terdapat lebih dari dua kategori, disebut ***multi-class classification.***

- Contoh *multiclass classification* adalah mengklasifikasikan gambar buah ke dalam himpunan kelas: *apel*, *mangga* atau *sirsak*.



ILUSTRASI *MULTI-LABEL* DAN *MULTI-CLASS CLASSIFICATION*

Instans	Apel	Mangga	Sirsak
Gambar-1	1	0	0
Gambar-2	0	1	0
Gambar-3	0	0	1
…			

Multi-class Classification

Instans	Agama	Politik	Hiburan
Berita-1	1	1	0
Berita-2	0	1	1
Berita-3	1	0	1
…			

Multi-label Classification

- Ada tipe klasifikasi lain disebut ***multi-label classification*** yaitu ketika kita ingin mengklasifikasikan suatu instans ke dalam suatu himpunan kelas.
- Perbedaan *multi-class* dan *multi-label classification* agak *tricky*.
- Pada *multi-class classification*, suatu instans hanya bisa berkorespondensi dengan satu kelas.
- Sedangkan pada *multi-label classification*, satu instans dapat berkorespondensi dengan lebih dari satu kelas.

- Misalnya, suatu berita dapat masuk ke kategori *agama* dan *politik* pada waktu bersamaan.
- Artinya, label pada *multi-class classification* bersifat *mutually exclusive*, sedangkan label tidak bersifat *mutually exclusive* pada *multi-label classification*.
- Perhatikan ilustrasi di atas, dimana setiap baris merepresentasikan kelas yang berkorespondensi dengan setiap instans, nilai "1" melambangkan TRUE dan nilai "0" melambangkan FALSE.
- *Multi-label classification* dapat didekomposisi menjadi beberapa *Binary classification*, yaitu mengklasifikasikan apakah instans dapat di*assign* ke suatu kelas atau tidak.

$p(y \mid x, \mathbf{w})$ (1.1)

$$p(y \mid x, \mathbf{w}) \qquad (1.1)$$

- Pemahaman *supervised learning* adalah mengingat persamaan 1.1. Ada tiga hal penting pada *supervised learning* yaitu *input*, *desired output*, dan *learning parameters*.
- Perlu ditekankan *learning parameters* berjumlah lebih dari satu, dan sering direpresentasikan dengan vektor (*bold*) atau matriks.
- Berdasarkan model yang dibuat, kita dapat melakukan klasifikasi (misal simbol yang ditulis di papan adalah angka berapa).
- Secara konseptual, klasifikasi didefinisikan sebagai persamaan 1.2 yaitu memilih label (kelas/kategori *y*) paling optimal dari sekumpulan label *C*, diberikan (*given*) suatu instans data tertentu

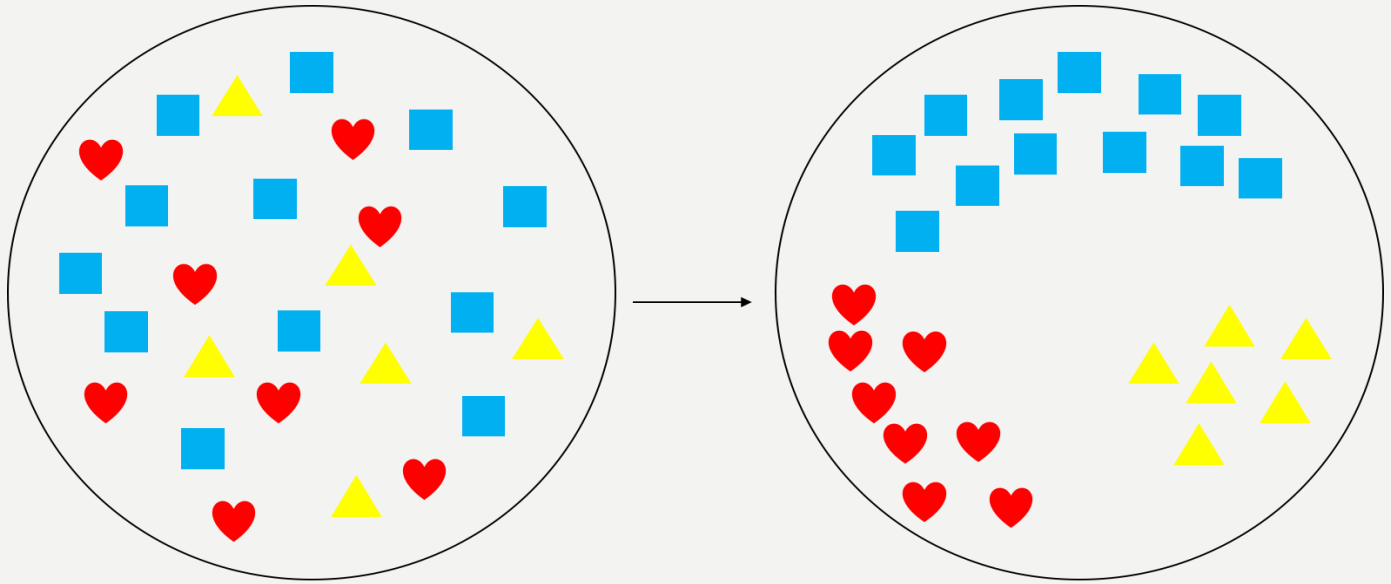
$$\hat{y}_i = \arg\max_{y_i \in C} p(y_i \mid x_i, \mathbf{w}) \qquad (1.2)$$

SEMI-SUPERVISED LEARNING

- *Semi-supervised learning* mirip dengan *supervised learning*, bedanya pada proses pelabelan data.
- Pada *supervised learning*, ada "guru" yang harus membuat "kunci jawaban" *input-output*.
- Sedangkan pada *semi-supervised learning* tidak ada "kunci jawaban" eksplisit yang harus dibuat guru.
- Kunci jawaban ini dapat diperoleh secara otomatis (misal dari hasil *clustering*).
- Pada kategori pembelajaran ini, umumnya kita hanya memiliki sedikit data.
- Kita kemudian menciptakan data tambahan baik menggunakan *supervised* ataupun *unsupervised learning*, kemudian membuat model belajar dari data tambahan tersebut.

UNSUPERVISED LEARNING

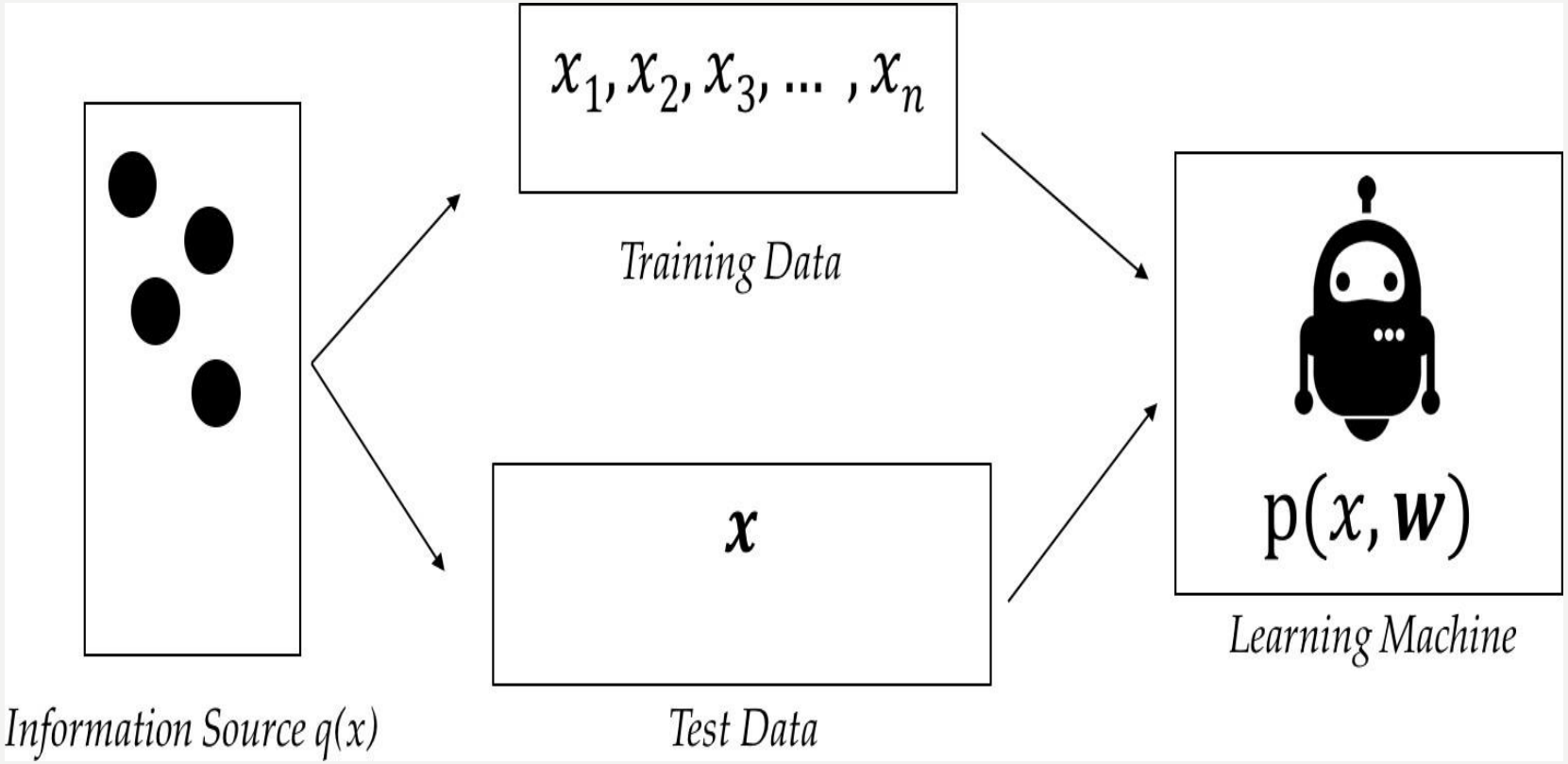
- Jika pada *supervised learning* ada guru yang mengajar, maka pada *unsupervised learning* tidak ada guru yang mengajar.
- Contoh permasalahan unsupervised learning adalah *clustering*.
- Contoh kita ingin mengelompokkan kue-kue yang sama.



Sebelum *clustering*

Hasil *clustering*

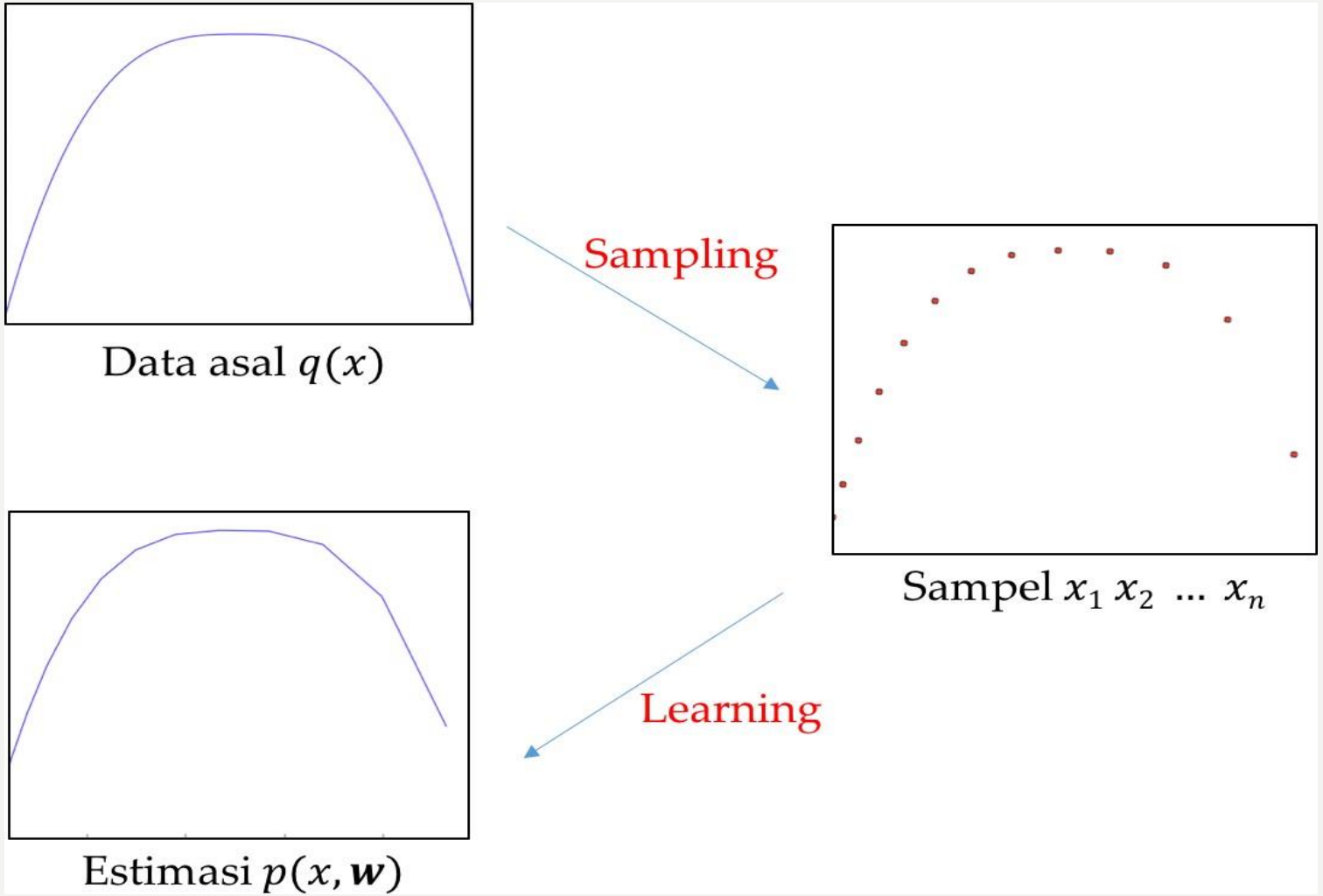
- Yang dilakukan adalah membuat kelompok-kelompok berdasarkan karakteristik kue, misal kelompok kue biru, kelompok kue kuning, atau kelompok kue merah.
- Contoh algoritma *unsupervised learning* sederhana adalah *K-means*.



Unsupervised learning framework

- Berbeda dengan supervised learning yang memiliki *desired output*,
- Pada *unsupervised learning* tidak ada *desired output* (jelas, tidak ada gurunya, tidak ada yang memberi contoh).
- Kita ingin mencari tahu distribusi asli data $q(x)$, berdasarkan beberapa sampel data.
- *Learning* dilakukan dengan mengoptimalkan $p(x \mid \mathbf{w})$ yang mengoptimasi parameter **w**.
- Perbedaan antara estimasi dan fungsi asli disebut sebagai ***generalization loss*** (atau ***loss*** saja).
- Kunci pemahaman *unsupervised learning* adalah mengingat persamaan 1.3, yaitu ada *input* dan parameter.
- $p(x \mid \mathbf{w})$ (1.3)

GENERALIZATION ERROR OF UNSUPERVISED LEARNING



- *Unsupervised learning* = *clustering* !
- *Clustering* adalah salah satu bentuk *unsupervised learning* ; yaitu salah satu hasil inferensi persamaan 1.3.
- *Unsupervised learning* adalah mencari sifat-sifat (*properties*) data.
- Kita ingin aproksimasi $p(x \mid \mathbf{w})$ semirip mungkin dengan $q(x)$, dimana $q(x)$ adalah distribusi data yang asli.
- Dataset disampel dari distribusi $q(x)$, kemudian kita ingin mencari tahu $q(x)$ tersebut.

PROSES BELAJAR

- *supervised* maupun *unsupervised learning*, kita ingin mengestimasi sesuatu dengan teknik *machine learning*.
- Kinerja *learning machine* berubah-ubah sesuai dengan parameter **w** (parameter pembelajaran).
- Kinerja *learning machine* diukur oleh fungsi tujuan (*utility function*/*performance measure*), yaitu mengoptimalkan nilai fungsi tertentu; misalnya meminimalkan nilai *error*, atau meminimalkan *loss*.
- Secara intuitif, *learning machine* mirip seperti saat manusia belajar.
- Kita awalnya membuat banyak kesalahan, tetapi kita mengetahui/diberi tahu mana yang benar.

- Untuk itu kita menyesuaikan diri secara perlahan agar menjadi benar (iteratif).
- Inilah yang juga dilakukan *learning machine*, yaitu mengubah-ubah parameter **w** untuk mengoptimalkan suatu fungsi tujuan.
- Akan tetapi, *machine learning* membutuhkan sangat banyak data. Sementara, manusia dapat belajar dengan contoh yang sedikit.

- Secara bahasa lebih matematis, diberi contoh *supervised learning*.
- Kita mempunyai distribusi klasifikasi asli $q(y \mid x)$.
- Dari distribusi tersebut, kita diberikan beberapa sampel pasangan *input-output* $\{z_1, z_2, z_3, \ldots, z_n\}$; $z_i = (x_i, y_i)$.
- Kita membuat *learning machine* $p(y| x, \mathbf{w})$.
- Awalnya diberi (x_1, y_1), *learning machine* mengestimasi fungsi asli dengan mengoptimalkan parameter **w** sesuai dengan data yang ada.
- Seiring berjalannya waktu, ia diberikan data observasi lainnya, sehingga *learning machine* menyesuaikan dirinya (konvergen) terhadap observasi yang baru (x_2, y_2), (x_3, y_3),
- Semakin lama, kita jadi makin percaya bahwa *learning machine* semakin optimal (mampu mem-prediksi fungsi aslinya).
- Apabila kita diberikan data sejumlah tak hingga, kita harap aproksimasi kita sama persis dengan distribusi aslinya