

7. Pemprosesan Isyarat dan Pencaman Pola

7.0 Isyarat dan Hingar

7.1 Pemprosesan Isyarat

7.2 Pencaman Pola

7.3 Pengklasifikasian dan Penggugusan

- Tugas 13

7.0 Isyarat dan Hingar

7.0.0 Asas

Isyarat ialah sebarang bentuk data, tenaga, atau mesej yang membawa maklumat. Sesuatu ukuran memberi isyarat. Ia boleh wujud dalam pelbagai bentuk seperti arus elektrik, gelombang bunyi, atau isyarat optik.

Sesuatu isyarat itu, $S(t)$ biasanya diselubungi hingar $N(t)$, sesuatu yang mencemarkan ketulenan isyarat.

Hingar boleh jadi boleh diramal atau tak boleh diramal. Hingar mengurangkan maklumat, dan membawa kepada ketidakpastian dalam pengetahuan tentang isyarat.

Suatu sistem linear ialah di mana kita boleh tangani isyarat dan hingar secara berasingan dan sebagai superposisi,

$$m(t) = s(t) + n(t).$$

Nisbah isyarat ke hingar, ialah nisbah kuasa isyarat kepada kuasa hingar,

$$SNR = \frac{P_s}{P_n}$$

Bagi bentuk sinusoid,

$$SNR = \left(\frac{A_s}{A_n} \right)^2,$$

di mana A adalah punca min kuasadua amplitud.

Hingar diklasifikasi sebagai

- rawak, misalnya pergerakan molekul gas pada sesuatu nilai suhu
- bersistem, misalnya ralat yang terhasil daripada radas rosak, peleraian terhad, kesilapan operasi, dan sebagainya.

Hingar rawak membawa kepada ralat statistik, dan hingar bersistem kepada ralat sistematik.

Bagi hingar rawak $n(t)$, analisis statistik memberitahu, min

$$\langle n(t) \rangle = \int n p_n dn = (1/T) \int_{-T/2}^{T/2} n(t) dt \quad (T \rightarrow \infty)$$

di mana kebarangkalian menemui $n(t)$ di antara n dan $n+dn$ ialah $p_n dn$. Varians

$$\sigma^2 = \langle n^2 \rangle - \langle n \rangle^2 = \int (n - \langle n \rangle)^2 p_n dn = (1/T) \int_{-T/2}^{T/2} (n(t) - \langle n \rangle)^2 dt \quad (T \rightarrow \infty)$$

dan spektrum kuasa

$$P_n(f) = (1/T) |N(f)|^2 \quad (T \rightarrow \infty)$$

dengan $N(f) = FT[n(t)]$, iaitu transformasi Fourier $n(t)$. Hingar rawak bersumber daripada hingar fizik seperti sifat keazarahan jirim (iaitu ia tak selanjar), hingar terma (tenaga rawak Boltzmann $kT/2$ diambil oleh darjah kebebasan), dan daripada prinsip ketidakpastian (misalnya lebar garisan spektrum).

7.0.1 Hingar tembakan

Hingar ini dikaitkan dengan sifat keazarahan jirim. Ia timbul dari sumber yang besar, yang mengandungi banyak subsistem, setiapnya dengan kebarangkalian (bebas) tertentu sesuatu peristiwa

akan berlaku, misalnya pancaran spontan foton oleh atom, reputan radioaktif, dan pancaran termion electron oleh logam pada suhu tinggi.

7.1 Pemrosesan Isyarat

7.1.0 Am

Pemrosesan isyarat dilakukan atas data untuk menonjolkan isyarat sambil membunag hingar.

Apabila data berbentuk imej maka kita ada pemrosesan imej (lihat 4.2).

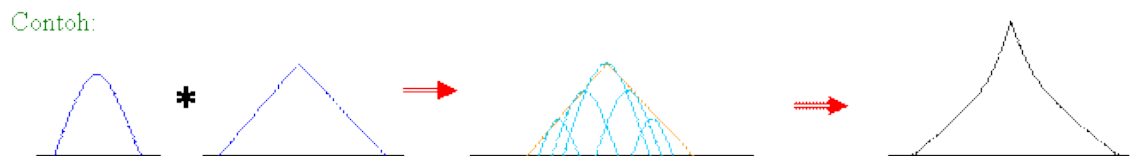
Berikut beberapa kaedah berguna dalam pemporsesan isyarat.

7.1.1 Konvolusi

Operasi konvolusi di antara 2 fungsi g_1 dan g_2 diberi,

$$g_1(t) * g_2(t) \equiv \int_{-\infty}^{\infty} g_1(t') g_2(t - t') dt'$$

Ia juga dirujuk sebagai lipatan g_2 ke dalam g_1 , atau sebaliknya. Sebagai contoh, proses konvolusi suatu fungsi kuadratik negatif dengan suatu fungsi tigasegi boleh dilihat seperti berikut:



Konvolusi mempunyai ciri-ciri berikut -

- kalis tukar tertib
$$g_1(t) * g_2(t) \equiv g_2(t) * g_1(t)$$
- kalis sekutuan
$$g_1(t) * [g_2(t) * g_3(t)] \equiv [g_1(t) * g_2(t)] * g_3(t)$$
- kalis agihan
$$g_1(t) * [g_2(t) + g_3(t)] \equiv g_1(t) * g_2(t) + g_1(t) * g_3(t)$$

Teorem konvolusi menyatakan,

$$g_1(t) * g_2(t) \equiv \text{FT}[G_1(f).G_2(f)] \equiv \text{FT}[\text{FT}[g_1(t)].\text{FT}[g_2(t)]] ,$$

di mana FT mewakili transformasi Fourier, yang bermakna

$$\text{FT}[g_1(t) * g_2(t)] \equiv G_1(f).G_2(f)$$

ataupun,

$$G_1(f) * G_2(f) \equiv \text{FT}[g_1(t).g_2(t)].$$

Dalam ukuran atau komunikasi, konvolusi timbul akibat modifikasi ke atas isyarat akibat, misalnya daripada sambutan pengesan, pengurangan atmosfera, penyampelan, dan sebagainya. Jadi, untuk dapatkan isyarat tulen asal, bolehlah di'nyahlipat' pemodifikasian ini (bila diketahui fungsi pemodifikasian) untuk memperolehi isyarat asal. Ini dilakukan dengan nyahkonvolusi menerusi

teorem konvolusi. Jika diukur $m(t) \equiv s(t)*n(t)$, maka

$$s(t) = \text{FT}[\text{FT}[m(t)]/\text{FT}[n(t)]].$$

Kiraan kamiran dalam konvolusi dan dalam transformasi Fourier bolehlah gunakan pengiraan komputer menerusi kamiran berangka. Perhatikan bahawa $n(t)$ atau $\text{FT}[n(t)] \equiv N(f)$ perlu diketahui awalnya.

7.1.2 Korelasi dan Autokorelasi

Korelasi mengukur keserupaan bentuk.

Korelasi di antara dua fungsi $m_1(t)$ dan $m_2(t)$ diberikan:

$$\phi_{12}(\tau) = \frac{1}{T} \int_{-T/2}^{T/2} m_1(t) m_2(t + \tau) dt \quad (T \rightarrow \infty)$$

Ia mengukur keserupaan fungsi-fungsi ini, dengan beza tempoh τ .

Autokorelasi adalah korelasi di antara fungsi yang sama. Autokorelasi bagi $m_1(t)$:

$$\phi_{11}(\tau) = \frac{1}{T} \int_{-T/2}^{T/2} m_1(t) m_1(t + \tau) dt \quad (T \rightarrow \infty)$$

terhadap beza tempoh τ .

Korelasi berguna bila diketahui bentuk isyarat, dan apa yang dikehendaki adalah mengesan dan/atau mengukur kekuatan isyarat di dalam hingar. Dengan mengukur/mengira korelasi di antara isyarat (+hingar) yg diukur dgn bentuk isyarat yg diketahui, wujudnya isyarat, bila tibanya, dan kekuatannya dapat dikesan. Kewujudan isyarat ditandakan dengan nilai korelasi tinggi pada sesuatu kedudukan, jika diandaikan hingar rawak dan korelasinya dipuratakan kepada 0. Kedudukan berkenaan memnyatakan masa tiba isyarat, dan nilai korelasi memberikan saiz isyarat.

Autokorelasi pula berguna walau tidak diketahui bentuk isyarat, akan tetapi diketahui bahawa isyarat dicari bersifat berkala, dan hingar bersifat rawak. Apabila dilihat autokorelasi isyarat (+hingar) yg diukur – hingar rawak dipuratakan ke 0 – autokorelasi punyai maksimum pada frekuensi sepadan dengan kala isyarat, kerana dengan beza tempoh berkenaanlah ukuran paling serupa dengan dirinya.

Korelasi dan autokorelasi boleh dijalankan secara pengiraan komputer menerusi kamiran berangka.

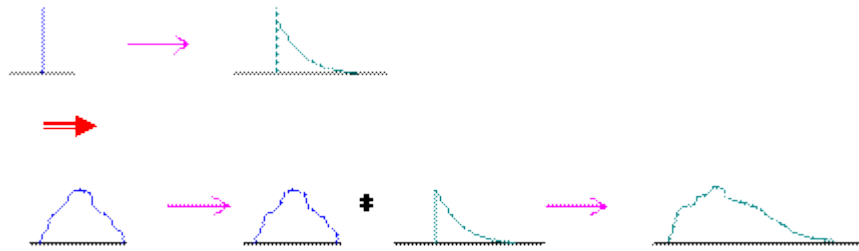
7.1.3 Turas

Menerusi tindakan suatu turas, suatu masukan $V(f)$ memberi keluaran $V_o(f)$.

$$G(f) \equiv V_o(f)/V(f)$$

ialah fungsi hantaran atau sambutan frekuensi (yang bernilai kompleks) bagi turas berkenaan.

Dalam domain masa, apabila suatu impuls masukan $\delta(0)$ memberi keluaran $K(t)$, fungsi $K(t)$ dikatakan sambutan impuls. Maka keluaran ialah konvolusi sambutan impuls dan fungsi masukan, Suatu contoh grafik diberikan di bawah.

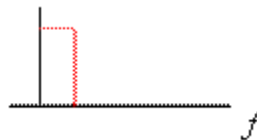


Daripada teorem konvolusi,

$$K(t) \equiv \text{FT}[G(f)].$$

Turas daripada $G(f)$ sebenarnya adalah kesan sambutan impuls $K(t)$.

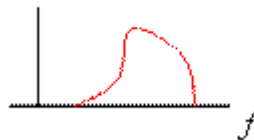
Dalam kes diketahui isyarat adalah berfrekuensi rendah, hingar frekuensi tinggi boleh dihapuskan menggunakan turas laluan rendah. Ini berbentuk seperti berikut:



Turas laluan tinggi pula boleh digunakan jika isyarat frekuensi tinggi, untuk menghapuskan hingar frekuensi rendah, seperti berikut:



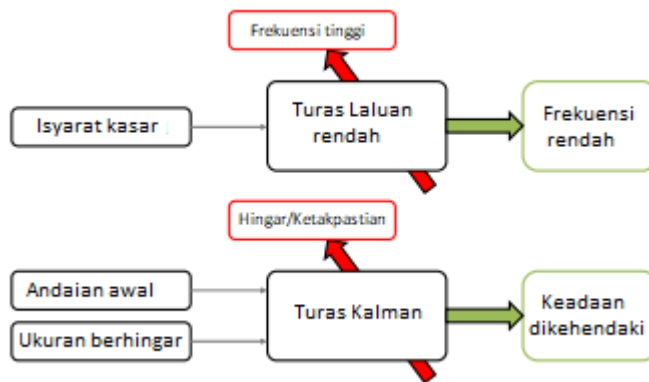
Turas sepadan ialah turas khas dengan sambutan frekuensi tertentu. Kesan sambutan impuls $K(t)$ adalah mengekan konvolusi ke atas isyarat. Kesan ini dihapuskan menerusi nyahkonvolusi dengan $\text{FT}[1/G(f)]$.



Proses turas boleh dilakukan menggunakan pengiraan komputer.

7.1.4 Turas Kalman

Turas Kalman ialah sejenis turas komputeran yang suai, yang membuat anggaran ralat semasa pengiraan berjalan. Ia mengambil maklumat yang diketahui ada sedikit ralat, ketidakpastian atau hingar, menangani bahagian yang berguna, dan mengurangkan ketidakpastian atau hingar berkenaan. Fungsinya sebagai turas dibandingkan dengan suatu turas laluan rendah di bawah ini.



Turas Kalman bertindak ke atas sistem linear yang diperihalkan seperti berikut:

$$\mathbf{x}_k = \mathbf{F}_{k-1} \mathbf{x}_{k-1} + \mathbf{G}_{k-1} \mathbf{u}_{k-1} + \mathbf{w}_{k-1}$$

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k$$

dengan pembolehubah dan dimensinya:

x	Vektor keadaan	$n_x \times 1$
y	Vektor keluaran	$n_y \times 1$
u	Vektor masukan	$n_u \times 1$
w	Vektor hingar proses	$n_x \times 1$
v	Vektor hingar ukuran	$n_y \times 1$
F	Matriks sistem - keadaan	$n_x \times n_x$
G	Matriks sistem - masukan	$n_x \times n_u$
H	Matriks pemerhatian	$n_y \times n_x$

Vektor **y** mewakili apa yang diukur, sementara **x** mewakili apa yang dikehendaki.

Kaedah peramal-pembetulan digunakan, dengan algoritma berikut.

Algoritma: mula dgn anggaran awal $\hat{\mathbf{x}}_0$ dan matriks kovarian ralat keadaan awal $\mathbf{P}_0$
ulang:

teka keadaan seterusnya

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_{k-1} \hat{\mathbf{x}}_{k-1} + \mathbf{G}_{k-1} \mathbf{u}_{k-1}$$

ramal matriks kovarian ralat keadaan

$$\mathbf{P}_{k|k-1} = \mathbf{F}_{k-1} \mathbf{P}_{k-1} \mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1}$$

kira matriks gandaan Kalman

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

kemaskini keadaan

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1})$$

kemaskini kekovarianan ralat keadaan

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1}$$

7.2 Pencaman Pola

7.2.0 Am

Proses pencaman pola adalah untuk mengklasifikasikan data input ke dalam objek, kelas, atau kategori berdasarkan corak, ciri, atau keteraturan yang diiktiraf dalam data. Data ini pelbagai jenis, termasuk data pengesan, penderia, dan sebagainya, yang mungkin berbentuk imej, video, teks, audio dan lain-lain.

Kita bincangkan beberapa kaedah di sini.

7.2.1 Kaedah statistik

Ini menggunakan data statistik sejarah. Ukuran statistik tertentu digunakan untuk mengecam corak daripada contoh-contoh yang positif dan negatif yang diketahui. Fasa pembelajaran membabitkan proses pemerhatian terdahulu untuk membentuk model, menentukan statistik yang akan digunakan, dan had-had pengecaman positif dan negatif.

Kaedah ini boleh digunakan untuk mengecam pola-pola yang digesa.

7.2.2 Kaedah sintaksan

Daalam kes pengecaman pola kompleks, boleh digunakan pendekatan hierarkian. Ia berdasarkan hubungan antara primitif. Hubungan-hubungan ini yang dikesan.

Contoh kaedah ini ialah pengecaman hubungan huruf-huruf dalam perkataan yang sah.

7.2.3 Kaedah neuronan

Senibina rangkaian neuronan berlapis (lihat Bab 8) digunakan untuk belajar mengenal pola-pola masukan. Ia kemudiannya digunakan untuk pengklasifikasian pola.

7.2.4 Padanan templat

Ini merupakan kaedah lansung. Pola/corak/imej cerapan dibanding dengan templat rujukan untuk penerimaan/tolakan. Bandingan boleh dibuat dari segi bentuk atau parameter tertentu.

Suatu contoh ialah dalam pemprosesan imej digitan. Bentuk bayang objek boleh digunakan.

7.2.5 Pendekatan kaburan

Pendekatan ini adalah berdasarkan ciri-ciri serupa dalam pola. Pola-pola dikotakkan berdasarkan persamaan terhadap ciri-ciri tertentu, dan ciri atau pola yang serupa diperihalkan oleh sesuatu pemerihal secara kabur.

Pola-pola ini kemudiannya diklasifikasikan ke dalam ruang ciri yang dikesan, secara kabur atau anggaran.

7.2.6 Kaedah hibrid

Ini ialah gabungan beberapa kaedah.

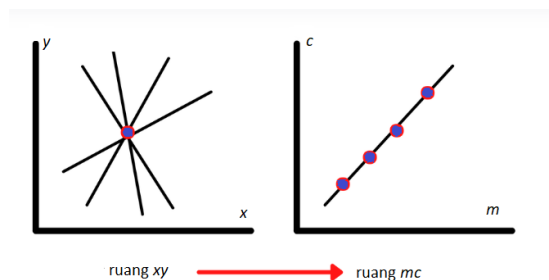
7.2.7 Beberapa contoh

Mari kita lihat 2 contoh, satu statistikan dan satu sintaksan. titik, kaedah Gauss-Siedel, dan pengenduran berlebihan berturutan.

7.2.7.1 Transformasi Hough

Kita ingin mencari garis lurus daripada taburan titik-titik. Setiap titik boleh merupakan sebahagian daripada garis lurus dengan parameter tertentu. Garis lurus punyai 2 parameter.

Kita transformasi setiap titik dari (x,y) ke ruang (m,c) . Titik (m,c) yang punyai banyak masukan merupakan parameter garis lurus yang punyai banyak kemungkinan titik di atasnya. Cari puncak dalam taburan (m,c) , atau lakukan pengambangan untuk memilih titik (m,c) yang punyai bilangan minimum tertentu sumbangan titik.

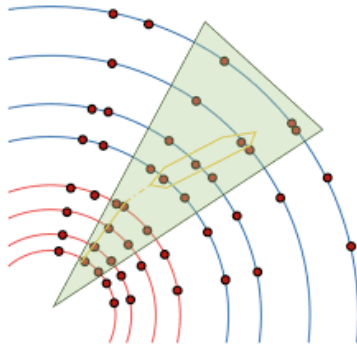


Kaedah ini merupakan kaedah sejagat kerana setiap masukan tidak mempengaruhi masukan yang lain.

7.2.7.2 Kaedah grafan mencari bulatan

Ini suatu kaedah tempatan kerana bulatan dibina daripada segmen-segmen satu-persatu. Segmen atau tepi boleh merupakan pasangan titik-titik dalam lampiran berjiranan.

Bina bulatan daripada tepi-tepi berjiranan yang saling padan. Padanan itu suatu pertimbangan sintaksan. Guna gelintaran (lihat bab 8) untuk pembinaan ini. Tepi-tepi ditambah lapisan demi lapisan, dan tambahan diteruskan bila ada padanan. Jika tiada padanan, calon tepi yang lain di lapisan sebelumnya yang hampir padan dicubai. Jika tiada lagi padanan, gelintaran berhenti.



Kaedah seperti ini sering digunakan untuk mengecam jejak-jejak zarah bercas dalam pengaruh medan magnet.

7.3 Pengklasifikasian dan Penggugusan

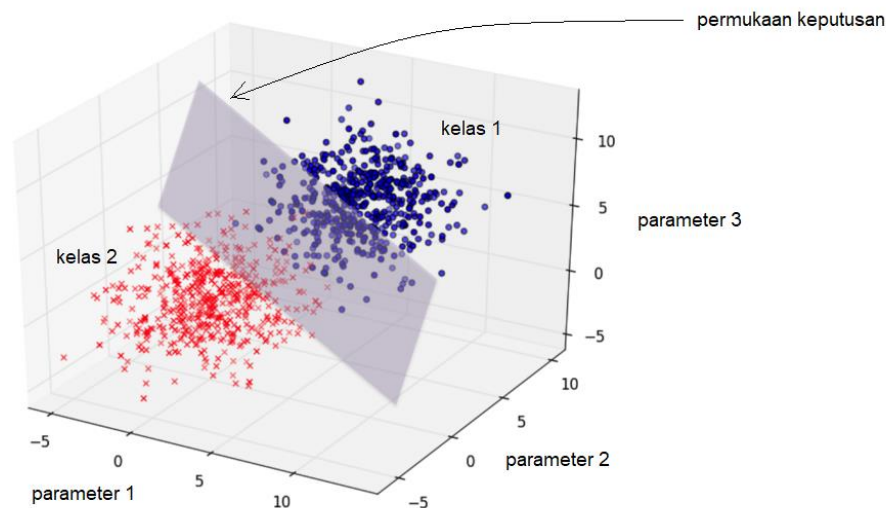
7.3.0 Am

Pengklasifikasian dan penggugusan serupa tetapi berbeza. Diberi data, pengklasifikasian ialah proses mengagihkan data kepada kategori tertentu yang telah ditetapkan, sementara penggugusan ialah proses mengelompokkan data ke dalam kumpulan berdasarkan kesamaan tanpa diberitahu pada awalnya apakah Kumpulan-kumpulan itu. Pengklasifikasian ialah untuk mengagihkan data kepada kelas sedia ada dan penggugusan ialah untuk menemui struktur semulajadi dalam data.

7.3.1 Pengklasifikasian

Pengklasifikasian ialah proses mengagihkan data kepada kategori tertentu yang telah ditetapkan. Data dikelaskan menurut persamaan. Contoh kelas atau kategori termasuk jenis e-mel - samada "spam" atau bukan, jenis penyakit seseorang pesakit, jenis perangai manusia, jenis imej seperti penggunaan tanah, lokasi yang disukai oleh beruang, lokasi potensi tanah runtuh., dan lain-lain. Jadi, di antara kegunaan pengklasifikasian adalah pengesanan penipuan dan pengesanan wajah.

Setiap titik data mempunyai nilai-nilai tertentu untuk suatu set parameter-parameter. Penentuan kategori atau kelas menggunakan gabungan nilai-nilai ini. Kita perlu [cari sempadan keputusan untuk memisahkan data kepada kelas tertentu](#).



Perhatikan bahawa titik-titik data dalam kelas tertentu dicirikan oleh nilai-nilai tertentu parameter tertentu. Ada kelas yang jelas. Ada juga titik-titik data dengan nilai-nilai parameter tertentu dan diketahui dalam kelas mana titik-titik itu berada ("data terlabel") dan diperlukan diterbitkan apakah ciri dan hubungan di antara parameter yang merujuk kepada label-label berkenaan. Ada kaedah untuk "mempelajari" ciri-ciri ini yang dikaitkan dengan sesuatu kelas itu.

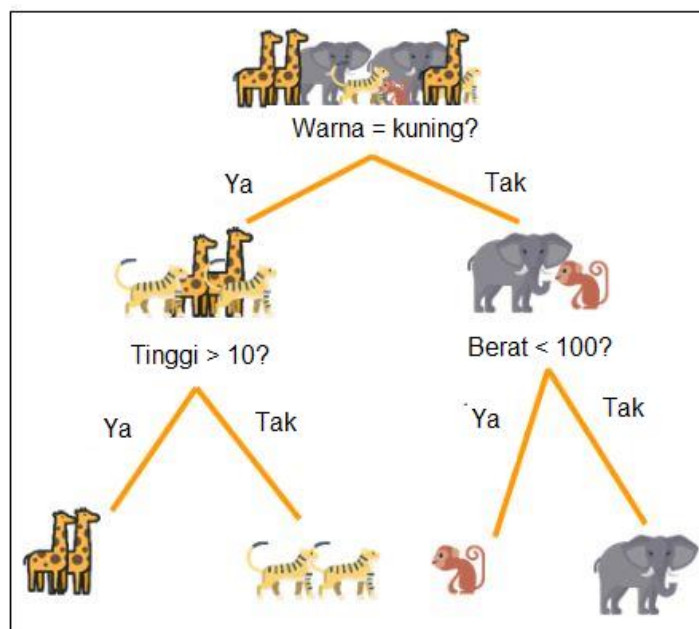
Ada beberapa kaedah untuk pengklasifikasian samada secara terus, dan dengan pembelajaran

terpandu (diberikan label-label berkenaan) yang ingin dibincangkan di sini: Pepohon Keputusan, K-Jiran Terdekat (KNN), Mesin Vektor Sokongan (SVM), Analisis Komponen Utama (PCA) dan Rangkaian Neuronan. Sementara yang lain merupakan kaedah pengklasifian mutlak, Analisis Komponen Utama sebenarnya ialah pemprosesan sebelum pengklasifikasian, diguna untuk mencari gabungan parameter yang paling dapat membezakan data. Cuma kaedahnya punyai keserupaan dengan kaedah-kaedah lain itu.

7.3.1.1 Pepohon Keputusan

Di sini, kita meletakkan kelas kepada sesuatu data atau item dengan mempertimbangkan nilai-nilai parameternya dalam jujukan tertentu.

Suatu contoh, dalam pengklasifikasian haiwan, diberikan di bawah.



Dalam contoh ini, ciri-ciri anggota kelas diketahui awal. Pengklasifikasian dibuat menurut ciri-ciri ini. Secara terus. Sempadan keputusan diberikan oleh syarat-syarat dalam membuat keputusan (misalnya, tinggi = 10).

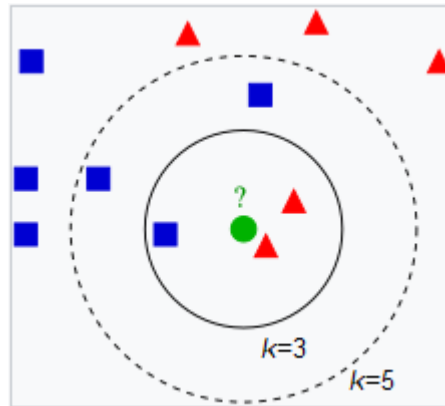
7.3.1.2 K-Jiran Terdekat

Ada masalah pengklasifikasian di mana kelas sesuatu data tidak begitu diketahui, tetapi perlu ditaakulkan daripada nilai-nilai parameter titik-titik data menurut kelas yang diketahui untuk titik berkenaan. Ada sekumpulan data yang telah dilabel kelasnya. Sistem perlu mempelajari label-label data baharu dengan membandingkan nilai-nilai parameter dengan data yang sudah terlabel. Ini adalah pembelajaran terarah atau pebelajaran berguru, di mana sistem dilatih menggunakan data yang mempunyai label kelas (data berlabel). Ramalan kelas atau kategori diperolehi bagi data baru.

Suatu kaedah pengklasifikasian dengan memadankan data baharu kepada kelas data berlabel sedia ada ialah kaedah k-jiran terdekat. Suatu nilai k diset, yang digunakan untuk menentukan bilangan jiran

dalam ruang parameter yang digunakan untuk pertimbangan. Ini bermakna, ada suatu ukuran ‘jarak’ dalam ruang parameter-parameter yang berbeza skala, yang digunakan untuk menentukan titik-titik data yang mana yang menjadi di antara k ‘jiran’ terdekat. Label data baharu ini ditentukan oleh label data yang mana yang terbanyak di antara k jiran terdekat tersebut.

Suatu contoh diberikan secara berskema di bawah.

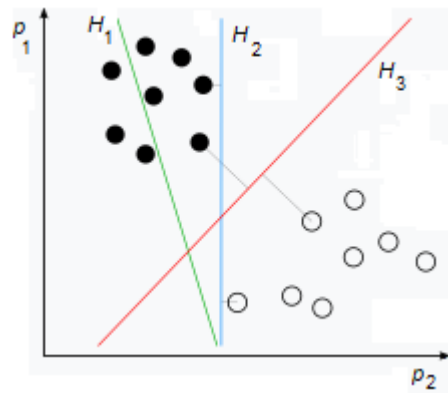


Ada dua kelas data: empatsegi biru dan tigasegi merah. Suatu data baharu (bulatan hijau) dipertimbangkan. Rajah di atas menunjukkan secara skema titik-titik data ini dalam ruang parameter-parameter yang digunakan. Bagi 3-jiran terdekat (bulatan garisan penuh), data baharu itu dikelaskan bersama tigasegi merah. Bagi 5-jiran terdekat (bulatan garisan berpecah-pecah, data baharu itu dikelaskan bersama empatsegi biru.

7.3.1.3 Mesin Vektor Sokongan

Mesin Vektor Sokongan berfungsi dengan mencari hipersatah sempadan yang terbaik untuk memisahkan data ke dalam kelas-kelas yang berbeza. Sempadan ini dalam ruang parameter-parameter. Bagi data yang dikaitkan dengan satu parameter, ia mencari aras-aras terbaik untuk memisahkan di antara kelas-kelas. Bagi dua parameter, garis-garis pemisah dicari; bagi tiga parameter, satah-satah pemisah, dan seterusnya, hipersatah secara am. Ia sangat berkesan dalam keadaan di mana terdapat banyak parameter cirian dalam data dan juga apabila data tidak terpisah secara linear.

Kaedah menggunakan konsep vektor sokongan, yang merupakan titik-titik data terdekat kepada hipersatah yang membentuk sempadan antara kelas-kelas. Vektor sokongan ini adalah penting kerana ia menentukan kedudukan hipersatah. Jika titik-titik ini diubah, kedudukan hipersatah juga berubah.



Bagi data dwiparameter di atas, hipersatah (garis) H_1 tidak memisahkan di antara kelas bulatan hitam dan bulatan putih, sementara H_2 memisahkan, namun dengan margin kecil, dan H_3 membuat pemisahan dengan margin maksimum.

Dalam kes linear, untuk mencari sempadan antara 2 kelas, cari margin – cari 2 hipersatah selari yang pisahkan 2 kelas ini dengan jarak di antara hipersatah paling besar. Hipersatah margin maksimum ialah hipersatah di pertengahan hipersatah-hipersatah ini.

Persamaan hipersatah diberikan oleh,

$$\mathbf{w} \cdot \mathbf{x} - b = 0,$$

di mana $\mathbf{w}$ adalah vektor serenjang kepada hipersatah, yang mengandungi titik-titik $\mathbf{x}$. b adalah pemalar yang menentukan offset terhadap asalan. Semak bahawa persamaan garis lurus diperolehi dalam 2 dimensi dan persamaan satah dalam 3 dimensi.

Katakan 2 hipersatah selari di berikan oleh persamaan-persamaan

$$\mathbf{w} \cdot \mathbf{x} - (b-1) = 0$$

$$\mathbf{w} \cdot \mathbf{x} - (b+1) = 0$$

tanpa menjejaskan keadaan am, kerana $\mathbf{w}$ tidak ternormal. Untuk titik-titik $\mathbf{x}$ daripada suatu kelas,

$$\mathbf{w} \cdot \mathbf{x} - b \geq 1$$

dan untuk yang satu lagi,

$$\mathbf{w} \cdot \mathbf{x} - b \leq -1.$$

Pengklasifikasi ditentukan $\{\mathbf{w}, b\}$ memeta $\mathbf{x}_i$ kepada $\text{sgn}(y_i) \equiv \text{sgn}(\mathbf{w} \cdot \mathbf{x}_i - b)$ di mana $\text{sgn}(\cdot)$ ialah fungsi tanda.

Jarak di antara 2 hipersatah ini ialah $2/\|\mathbf{w}\|$, jadi kita mahu maksimumkan jarak ini terhadap norma (atau magnitud) $\mathbf{w}$. Kita perlu minimumkan $\frac{1}{2} \|\mathbf{w}\|^2$ terhadap tertakluk kepada hukum pemetaan yang boleh ditulis

$$y_i (\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1$$

untuk semua i dalam data.

Jadi masalah mencari pemisah menjadi masalah pengoptimuman. Kekangan kepada peminimuman boleh dimasukkan dalam fungsi kos pengoptimuman seperti kaedah pekali Lagrange. Fungsi kerugian engsel,

$$\mu_i \equiv \max(0, 1 - y_i (\mathbf{w} \cdot \mathbf{x}_i - b))$$

mempunyai nilai 0 bila kekangan dipuaskan, dan nilai yang berkadaran dengan jarak $\mathbf{x}_i$ dari margin bila i tidak memuaskannya. Kita mahu minimumkan

$$\|\mathbf{w}\|^2 + C \sum_{i=1}^n \mu_i.$$

terhadap $\{\mathbf{w}, b\}$. Sebutan kedua sifar bila kekangan dipuaskan. C menentukan berapa kuat kita mahu kekangan dipuaskan, berbanding peminuman $\|\mathbf{w}\|^2$.

Peminimuman boleh dilakukan misalnya menggunakan kaedah Newton-Ralphson, yang telah dibincang sebelum ini. Kaedah ini juga dipanggil penurunan cerun.

Selain itu, SVM juga boleh digunakan untuk pengklasifikasian bukan linear dengan menggunakan teknik kernel, yang menggantikan hasildarab titik dengan kernel. Ini seperti mentransformasikan data yang seterusnya memberikan klasifikasi linear dalam ruang tertransformasi itu. Contoh kernel ialah fungsi polinomial,

$$k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + r)^d,$$

di mana $\phi(\mathbf{x})$ adalah hasil transformasi, dan fungsi asas jejarian,

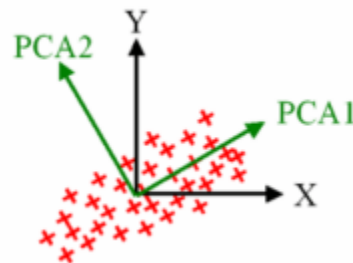
$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$$

dengan $\gamma > 0$.

7.3.1.4 Analisis Komponen Utama

Analisis komponen utama adalah kaedah untuk mengurangkan pemerihalan data, kepada ciri-ciri pentingnya, yang dipanggil komponen utama. Komponen utama adalah beberapa kombinasi linear pembolehubah pemerihalan asal yang secara optimum menjelaskan varians semua pembolehubah. Ia mencari transformasi yang menyerenjangkan pemerihalan data. Ini memudahkan pemisahan kelas.

Dalam contoh di bawah, suatu transformasi dilakukan daripada pembolehubah asal X dan Y kepada komponen-komponen utama PCA1 dan PCA2.



Bilangan komponen utama mungkin kurang daripada bilangan pembolehubah asal, dalam kes mana ia disebut pengurangan dimensi.

Kita mencari transformasi linear berortogon pada ruang hasildarab dalaman nyata yang mengubah data ke sistem koordinat baru supaya varians terbesar oleh sesuatu unjuran skalar data terletak pada komponen utama pertama, varians terbesar kedua pada komponen utama kedua, dan seterusnya.

Katakan transformasi pertama ini diberikan oleh $\mathbf{w}_1$ yang bertindak ke atas titik-titik data $\mathbf{x}_i$, memberikan unjuran ke atas komponen pertama

$$t_{1,i} = \mathbf{w}_1 \cdot \mathbf{x}_i.$$

Maka kita mahu maksimumkan

$$\{\sum_i (t_{1,i})^2\} \equiv \{\sum_i (\mathbf{w}_1 \cdot \mathbf{x}_i)^2\}$$

terhadap $\mathbf{w}_1$, tertakluk kepada ketetapan magnitudnya, $\|\mathbf{w}_1\|=1$. Ini serupa dengan memaksimumkan

$$\left\{ \sum_i \frac{(\mathbf{w}_1 \cdot \mathbf{x}_i)^2}{\|\mathbf{w}_1\|^2} \right\}.$$

Untuk komponen utama kedua, tolak komponen utama pertama daripada $\mathbf{x}_i$, dan lakukan pemaksimuman varians terhadap transformasi $\mathbf{w}_2$ secara serupa. Secara amnya, komponen utama ke- k diberikan oleh pemaksimuman

$$\left\{ \sum_i \frac{(\mathbf{w}_k \cdot \mathbf{x}_{i,k})^2}{|\mathbf{w}_k|^2} \right\}$$

terhadap $\mathbf{w}_k$ di mana

$$\mathbf{x}_{i,k} = \mathbf{x}_i - \sum_{s=1}^{k-1} \mathbf{w}_s \cdot \mathbf{x}_{i,s}$$

dengan $\mathbf{x}_{i,1} = \mathbf{x}_i$.

Peminimuman boleh dilakukan menggunakan kaedah penurunan cern dan sebagainya.

7.3.1.5 Rangkaian Neuronan

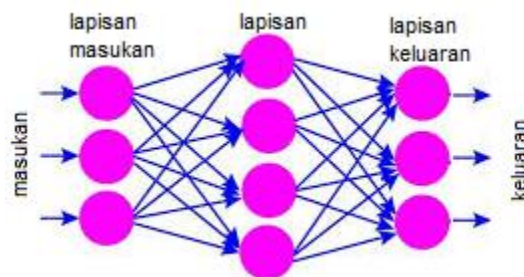
Model otak termasuk rangkaian neuronan berlapisan. Ini dibincangkan dalam 8 tetapi diterangkan sedikit di sini.

Rangkaian neuronan merupakan neuron-neuron V_i yang keadaannya diberikan oleh suatu fungsi terhadap jumlah keaktifan neuron lain diberatkan oleh kekuatan sambungan sinaps di antaranya, berbanding suatu nilai ambang,

$$V_i = f(\sum_j T_{ij} V_j - U_i) \equiv f(h_i)$$

Kekuatan sinaps T_{ij} boleh menguja (bernilai positif) atau merencat (bernilai negatif), dan ambang U_i biasanya dianggap 0 tanpa menghadkan model.

Senibina rangkaian neuronan berlapisan suap ke hadapan bermakna $T_{ij} = 0$ kecuali $i \in L_l$ dan $j \in L_{l-1}$ di mana L_l ialah lapisan ke- l , seperti digambarkan di bawah. Rangkaian sebegini menghasilkan pemetaan neuron-neuron di lapisan masukan, $\{V_i, i \in L_0\}$ kepada yang berada di lapisan terakhir iaitu lapisan keluaran, $\{V_i, i \in L_{lmaks}\}$. Bentuk pemetaan ditentukan oleh kekuatan sinaps yang bukan sifar (dan ambang).



Rangkaian neuron dengan 3 lapisan

Jadi rangkaian neuronan berlapisan boleh lakukan pengklasifikasian menerusi pemetaan, daripada ciri atau parameter data sebagai masukan, kepada kelas sebagai keluaran.

Ada kaedah untuk menyesuaikan kekuatan sinaps untuk memberikan pemetaan yang dikehendaki. Sistem ‘mempelajari’ pemetaan ini diberi set contoh-contoh masukan-keluaran (pengajaran terselia/oleh guru). Suatu masukan (yakni data) daripada set ini menghasilkan keluaran tertentu, yang pada mulanya tak semestinya keluaran (yakni kelas) yang betul. Bagi pasangan nilai neuron masukan-keluaran $\{\xi_k, \xi_i; k \in L_0, i \in L_{kel}\}$, ada ralat yang kita definisikan

$$\varepsilon = \frac{1}{2} \sum_{i \in L_{\text{kel}}} (V_i - \xi_i)^2$$

di mana adalah nilai yang didapati dengan nilai sekarang, dan ξ_i adalah nilai yang dikehendaki seperti yang diberikan oleh pasangan contoh. Untuk ralat keseleruhan, ia perlu dihasiltambah terhadap semua contoh masukan-keluaran yang diberi. Kita ingin meminumkan ralat ini terhadap pemilihan nilai sinaps T_{ij} . Kaedah Newton-Ralphson atau penurunan cerun memberikan suaian T_{ij} berikut:

$$\Delta T_{ij} = -\alpha \frac{\partial \varepsilon}{\partial T_{ij}}$$

dengan α sebagai kadar pembelajaran. Hukum rantai memberikan,

$$\Delta T_{ij} = -\alpha \sum_c (V_i - \xi_i) \frac{dV_i}{dh_i} V_j$$

(sekarang hasiltambah terhadap semua contoh masukan-keluaran c dibuat). Kita tetapkan rangkaian yang punyai 3 lapisan, dengan indeks i di lapisan keluaran, indeks j di lapisan pertengahan atau lapisan tersorok, dan k di masukan. Jika f dipilih fungsi logistik,

$$f(x) = g(x) \equiv \frac{\exp(\beta x)}{1 + \exp(\beta x)},$$

yang berbentuk sigmoid dengan kecuraman β , dengan $g' = g(1 - g)$, maka

$$\frac{dV_i}{dh_i} = \frac{d}{dh_i} g(h_i) = g(h_i)(1 - g(h_i)) = V_i(1 - V_i).$$

Pula, menggunakan hukum rantai,

$$\begin{aligned} \Delta T_{jk} &= -\alpha \frac{\partial \varepsilon}{\partial T_{jk}} = -\alpha \sum_c \sum_i (V_i - \xi_i) \frac{dV_i}{dh_i} \frac{dV_j}{dh_j} V_k \\ &= -\alpha \sum_c \sum_i (V_i - \xi_i) V_i(1 - V_i) V_j(1 - V_j) V_k \end{aligned}$$

Penyesuaian mungkin memerlukan banyak pusingan pembelajaran menggunakan set contoh diberi, yang akhirnya memberikan pengklasifikasian yang betul untuk set ini ('pengahafalan') dan juga bagi masukan baharu ('penyeluruhan'). Walaupun masukan baharu ini belum ditemui, lapisan tersorok dijangkakan mengorganisasi untuk memberikan keluaran yang betul.

7.3.2 Penggugusan

Penggugusan meneroka kumpulan yang semulajadi, dalam set data. Data dimasukkan ke dalam Kumpulan-kumpulan berdasarkan kesamaan. Gugusan/kumpulan/kelas data tidak dipradefinisikan. Matlamat penggugusan ialah mengenal pasti kumpulan semulajadi dalam data.

Penggugusan dikatakan suatu pembelajaran yang tidak diselia (atau, tanpa guru). Pola diperoleh daripada data tanpa label atau kelas yang telah ditetapkan.

Algoritma yang baik memastikan kesamaan intra-cluster yang tinggi dan kesamaan antara kluster yang rendah. Apabila setiap titik data input samada milik kluster sepenuhnya atau tidak, ia disebut penggugusan keras. Penggugusan lembut adalah apabila kebarangkalian atau kemungkinan titik data berada di dalam sesuatu gugusan diberikan.

Contoh penggugusan termasuk mengguguskan pelanggan berdasarkan corak pembelian, dalam analisis pelanggan (pengelompokan pelanggan), analisis genetik, pengenalan komuniti dalam rangkaian sosial, dan pengsegmenan imej iaitu mengenal pasti kawasan yang serupa dalam gambar.

Kaedah-kaedah utama bagi penggugusan termasuk kaedah pengotakan, hierarki, berasaskan

ketumpatan, berasaskan grid, berasaskan model, berasaskan kekangan, dan analisis pesisir.

7.3.2.1 Kaedah pengotakan

Tiada andaian dibuat tentang taburan data. Dalam model sentroid, ukuran kesamaan diterbitkan daripada kedekatan titik data ke pusat sentroid atau pusat gugusan dalam ruang parameter. Bermula dengan beberapa gugusan awal, beransur-ansur pindahkan titik data sesuai ke gugusan yang berlainan sehingga pengotakan optimum dicapai.

Bilangan gugusan akhir dimahukan perlu dinyatakan bagi pengotakan optimum, jadi pengetahuan terdahulu ini tentang set data perlu ada. Algoritma bersifat berlelar, dan ia menemukan optimum tempatan.

Kelebihan kaedah ini ialah ia mudah dan boleh skala (ia punyai kekompleksan linear: $O(n)$). Juga, ia berkesan untuk set data dengan gugusan padat dan terpisah (terutama gugusan berbentuk hipersferaan).

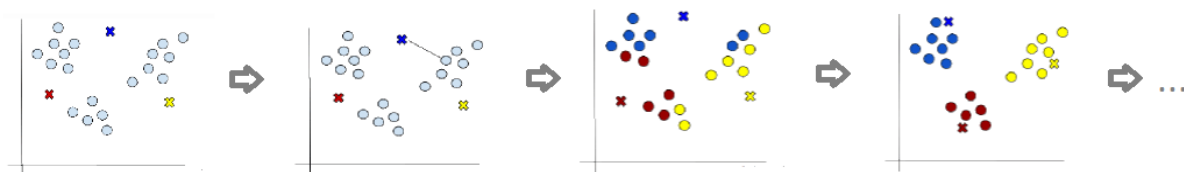
Kekurangannya ialah perlu ditentukan bilangan gugusan dikehendaki terlebih dahulu. Juga, ia tak begitu berkesan untuk data berdimensi tinggi. Ia tidak begitu lasak dan terkesan oleh hingar dan nilai pesisir. Hasil juga mungkin berlainan untuk pilihan awal yang berlainan.

Contoh kaedah ini ialah algoritma K -min, seperti berikut:

Pilih K (bilangan gugusan dikehendaki)
Pilih K titik sebagai pusat K gugusan awal
Ulang sehingga tiada perubahan:
 Kaitkan setiap titik data kepada gugusan yang pusatnya paling hampir
 Daripada ahl-ahli gugusan, kira semula pusat setiap gugusan

Suatu cara untuk memilih K ialah seperti berikut. Kira hasil tambah jarak titik ke pusat gugusan, dan plot lawan K . Pilih nilai K yang mana selepasnya nilai hasil tambah jarak ini tidak lagi turun mendadak.

Perjalanan algoritma ini ditunjukkan secara berskema di bawah:



7.3.2.2 Hierarki

Bagi kaedah ini juga, tiada andaian dibuat tentang pola taburan data.

Model kesambungan dipakai, berdasarkan tanggapan bahawa titik data lebih dekat dalam ruang data mempamerkan lebih banyak persamaan antara satu sama lain daripada titik data yang terletak lebih

jauh.

Ada dua pendekatan:

1. mula dengan mengklasifikasikan setiap titik data sebagai gugusan berasingan dan kemudian menggabungkan gugusan-gugusan yang mempunyai jarak berkurangan. Penggabungan diulang dengan meningkatkan jarak gabungan, sehingga keadaan munasabah.
2. semua titik data diletakkan dalam satu gugusan dan kemudian dibahagikan apabila mempunyai jarak besar. Pembelahan diulang dengan mengurangkan jarak pemisahan, sehingga keadaan munasabah.

Dengan ini, hierarki gugusan-gugusan dibina.

Kelebihan kaedah ini ialah

- Sangat mudah untuk mentafsir
- Tak perlu tentukan gugusan terlebih dahulu
- Hasil bolehulang
- Kaedah pemvisualan yang bagus seperti dendogram

sementara kekurangannya pula ialah

- Kurang berskala (ia punyai kekompleksan $O(n^2)$)
- Prestasi merosot dalam kes dimensi tinggi
- Setelah belahan atau gabungan kelompok dibuat, susah mahu dibetulkan

Contoh kaedah ini ialah algoritma penggugusan hierarki, yang diberikan seperti berikut:

Definisikan ukuran jarak ketaksamaan $d_{kk'} \equiv d(C_k, C_{k'})$ di antara gugusan C_k dan $C_{k'}$ sbg fungsi terhadap jejak antara titik2 dlm gugusan

Mula: setiap titik sbg gugusan awal (gugusan2 setitik)

Ulang sehingga 1 gugusan tinggal:

Cari minimum $d_{kk'}$ terhadap $C_k, C_{k'}$

Gabung 2 gugusan berkenaan

Ukuran jarak ketaksamaan antara titik data adalah misalnya,

$$d(x, y) = (x - y)^2$$

atau

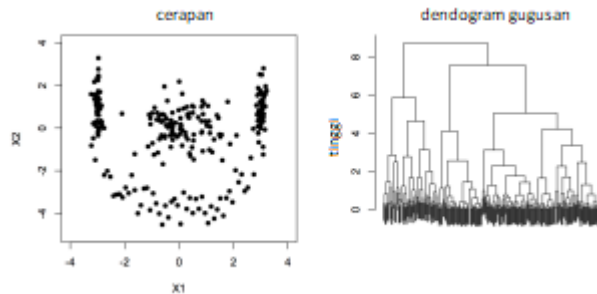
$$d(x, y) = |x - y|.$$

Ukuran jarak ketaksamaan antara gugusan bolehlah dipilih daripada misalnya,

- kawan-kpd-kawan – jarak terdekat antara titik terkandung: $d(C_k, C_{k'}) = \min_{x \in C_k, y \in C_{k'}} d(x, y)$
- jarak maksimum: $d(C_k, C_{k'}) = \max_{x \in C_k, y \in C_{k'}} d(x, y)$
- jarak Euklidan: $d(C_k, C_{k'}) = \sqrt{\sum_{x \in C_k, y \in C_{k'}} d(x, y)}$ dgn $d(x, y) = (x - y)^2$
 - Jarak Eulidan kuasdua
 - Jarak Manhattan – jarak Euklidan kuasdua tapi dgn $d(x, y) = |x - y|$
 - Jarak purata antara $d(x, y)$: $x \in C_k, y \in C_{k'}$
- Ward: $d(C_k, C_{k'}) = \frac{2(|C_k| \cdot |C_{k'}|)}{|C_k| + |C_{k'}|} \|\bar{x} - \bar{y}\|^2 \sqrt{\sum_{x \in C_k, y \in C_{k'}} d(x, y)}$
- Mahalanobis: menggunakan $d(x, y) = \sqrt{(\mathbf{x} - \mathbf{y})^T \mathbf{S}^{-1} (\mathbf{x} - \mathbf{y})}$, $\mathbf{S}$ = matriks kovarians

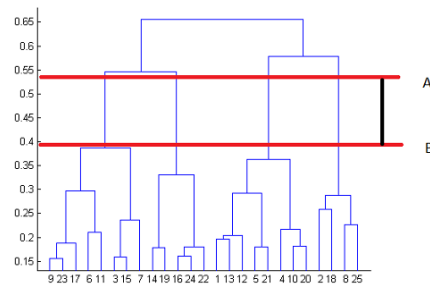
Matriks ketaksamaan **D** yang mengandungi ketaksamaan antara titik-titik data boleh digunakan untuk memerihalkan ketaksamaan di dalam gugusan dan di antara gugusan:

Kaedah ini berguna untuk pemvisualan data. Hasil penggugusan boleh ditunjukkan dalam bentuk dendogram, seperti di bawah.



Dalam dendogram, skala tegak mencerminkan jarak antara gugusan.

Bahkan, aras penggugusan boleh dipilih daripada dendogram. Cari ufuk yang memberikan gugusan malar untuk ketinggian terbesar, seperti contoh ini (AB terbesar):



7.3.2.3 Berasaskan Ketumpatan

Tiada andaian dibuat tentang pola taburan data.

Kaedah ini adalah berdasarkan ide yang dalam gugusan lebih tumpat daripada sekelilingnya.

Mula dengan satu titik data. Masukkan titik berjiran ke dalam gugusan yang sama, berdasarkan ketumpatan. (Kejiranan perlu didefinisikan).

Kelebihan kaedah ini:

- Ia bisa menemui gugusan berbagai bentuk dan saiz
- Ia baik dalam menangani hingar dan pesisih
- Tak perlu ditentukan bilangan gugusan terlebih dahulu

sementara kekurangannya:

- Jika gugusan berlainan ketumpatan, ia mungkin gagal
- Keluaran bergantung kuat kepada parameter masukan yang diset awal, termasuk dalam penentuan kejiranan

7.3.2.4 Berasaskan Grid

Kaedah ini ada persamaan dengan kaedah berasaskan ketumpatan.

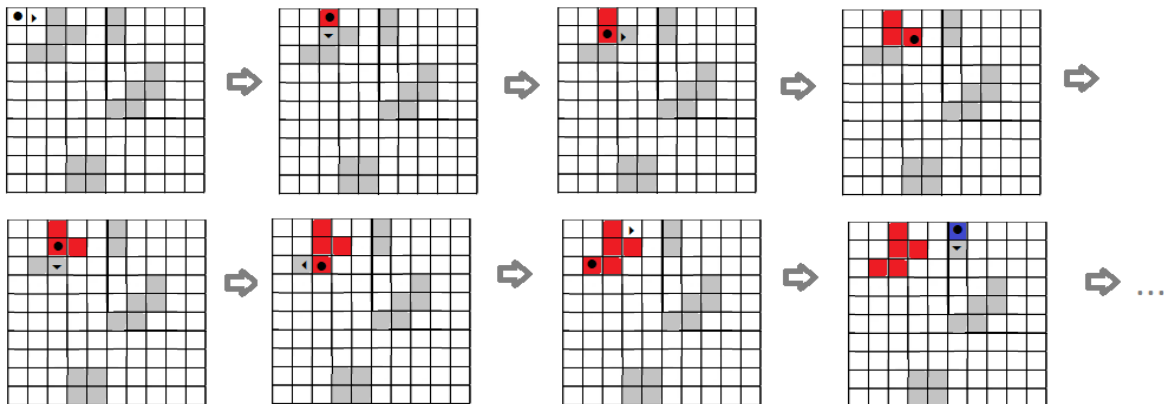
Ruang data dibahagikan kepada sel-sel dalam grid. Ketumpatan sel-sel dikira, dan diambangkan untuk memberi sel tumpat dan sel tak tumpat. Daripada sel tumpat berjiranan, gugusan dibentuk.

Kelebihan kaedah ini ialah dalam kelajuannya, kerana tiada pengiraan jarak dibuat, dan ia boleh tangani set data besar. Kekurangannya adalah sempadan gugusan dibentuk bersifat tegar, yang hanya mengandungi garisan mengufuk atau menegak.

Contoh kaedah ini ialah algoritma imbas-tekek:

Utk data dedua dalam grid – kelompokkan jiran-jiran bersentuhan
Cari kelompok sel2 ‘pasang’:
Imbas:
 pilih sel yang pasang, yg belum ditantadakan kelompok
 Tanda sel ini dgn no. kelompok baru
Imbas sel berjiranan:
 Pilih yg pasang
 Tanda sel ini dgn no. kelompok tadi
 Lakukan ‘imbas sel berjiranan’ yg serupa bagi sel ini

Suatu misalan diskemakan berikut.



7.3.2.5 Berasaskan Model

Ini merupakan penggugusan statistikan berparameter. Ia berdasarkan tanggapan bahawa sesuatu gugusan itu mempunyai taburan tertentu (misalnya, Gaussian). Parameter gugusan didapatkan.

Kelemahan kaedah ini ialah ia mengandaikan tiada korelasi, dan tidak sesuai untuk jenis data tertentu.

Suatu contoh ialah algoritma pemaksimuman jangkaan:

Pilih k pusat gugusan dan parameter lain gugusan secara rawak
Gugusan-gugusan dibaiki secara berlelar menerusi
Langkah jangkaan – peruntukkan gugusan kepada setiap titik menurut
parameter gugusan semasa
(misalnya, menggunakan kebarangkalian Bayesian)
Langkah pemaksimuman – optimumkan parameter gugusan (pusat, saiz, ...)
menurut keahlian titik semasa
(misalnya, maksimumkan kebarangkalian)

7.3.2.6 Berasaskan Kekangan

Ini menggunakan maklumat tambahan atau kekangan untuk membimbing proses pembentukan kluster.

Langkah yang dijalankan:

- (1) modelkan kekangan: ungkapan matematik yang mendefinisikan tatarajah gugusan yang dibolehkan
- (2) bentukkan gugusan: pengoptimuman suatu fungsi objektif yang mengimbangi kekangan dengan kehomogenan dalam setiap gugusan.

Lelarkan.

7.3.2.7 Analisis Pesisir

Titik-titik pesisir dikenalpasti dan diambilkira untuk memberikan penggugusan yang lebih baik.

Suatu contoh ialah DBSCAN (Penggugusan ruangan berasaskan kepadatan bagi aplikasi dengan hingar) yang melakukan penggugusan berasaskan kepadatan, dengan mengambilkira pesisir. Tiga jenis titik dikenalpasti – titik teras, yang berada dalam kepadatan tertentu, titik sempadan, yang berhampiran dengan titik teras tetapi tidak cukup kepadatan, dan titik hingar atau pesisir, yang bukan teras atau sempadan. Sesuatu gugusan terdiri daripada titik-titik teras yang disambung kepadatan.

Tugas 13

1. Gunakan autokorelasi untuk menyiasat samada isyarat berkala wujud atau tidak dalam ukuran berikut:

Masa	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0	1.1
Ukuran	0.1	0.7	0.8	-0.2	0.1	0.1	0.8	-0.5	0.2	0.8	0.9	0.1

1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0	2.1	2.2	2.3	2.4
0.2	0.8	1.0	-0.3	0.4	0.4	0.8	0.0	0.1	0.2	0.8	-0.3	-0.4

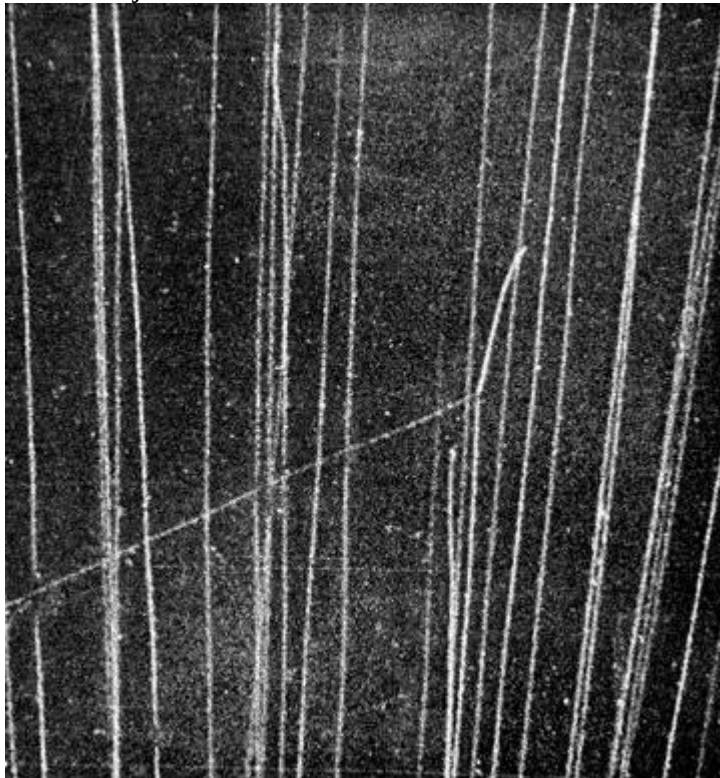
2. Gunakan korelasi untuk menentukan bila tibanya dan saiz isyarat berikut:

Masa	0.0	0.1	0.2	0.3
Ukuran	0.0	0.6	1.2	-0.3

dalam ukuran berikut:

Masa	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0	1.1
Ukuran	0.0	-0.1	-0.3	0.1	-0.3	0.7	0.6	-0.1	-0.1	0.2	-0.1	-0.3

3. Jejak zarah diimejkan:



Gunakan transformasi Hough (diskretkan ruang), dan penggugusan, untuk mengesan berapa jejak lurus yang ada, bersama parameternya. Plotkan.

Berhati-hati memilih parameter transformasi dan penggugusan supaya jawapan yang munasabah diperolehi.
 Imej ini mungkin perlu dipraproses (e.g. terbalikkan hitam↔putih, dsb).
 Terangkan apa dan bagaimana anda membuat pemprosesan dan tunjukkan hasil-hasilnya.

4. Kita gunakan turas Kalman linear untuk menentukan kedudukan suatu objek yang jatuh bebas tanpa rintangan udara berdasarkan maklumat tak pasti berkenaan kedudukan asal objek dan ukuran-ukuran keadaan dibuat oleh suatu penilik julat laser.
 Puaskan hati/otak anda bahawa parameter relevan adalah seperti berikut:

Terma	Definisi
Vektor keadaan (pada langkah masa k)	$\mathbf{x}_k = \begin{bmatrix} h_k \\ \dot{h}_k \end{bmatrix}$ h_k ketinggian
Vektor keluaran	$\mathbf{y}_k = h_k + \mathbf{v}_k$ $\mathbf{v}_k$ vektor hingar ukuran
Vektor masukan	$\mathbf{u}_{k-1} = g$ g pecutan graviti
Matriks keadaan sistem	$\mathbf{F}_{k-1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$
Matriks masukan	$\mathbf{G}_{k-1} = \begin{bmatrix} -\frac{1}{2}(\Delta t)^2 \\ -\Delta t \end{bmatrix}$
Matriks pemerhatian	$\mathbf{H}_k = [1 \ 0]$
Matriks kovarian hingar proses	$\mathbf{Q}_{k-1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$
Matriks kovarian hingar ukuran	$\mathbf{R}_k = 4$
Vektor keadaan awal sebenar	$\mathbf{x}_0 = \begin{bmatrix} 100 \\ 0 \end{bmatrix}$
Vektor keadaan awal diandaikan	$\hat{\mathbf{x}}_0 = \begin{bmatrix} 105 \\ 0 \end{bmatrix}$
Matriks kovarian ralat keadaan awal diandaikan	$\mathbf{P}_0 = \begin{bmatrix} 10 & 0 \\ 0 & 0.01 \end{bmatrix}$
Tokokan masa	$\Delta t = 0.001$

Lakukan penurasan Kalman untuk 1000 langkah masa. Jana hingar ukuran secara rawak
 Keadaan sebenar boleh didapati menggunakan kiraan analisis.
 Plotkan bersama ukuran yang dibuat, hasil daripada anggaran Kalman, dan nilai sebenar, untuk ketinggian terhadap masa.