BAB 2
LANDASAN TEORI

2.1 Bencana Alam

Menurut Asian Disaster Reduction Center “bencana alam adalah gangguan serius pada masyarakat yang bisa menyebabkan kerugian secara meluas dan dirasakan oleh semua lapisan masyarakat dan lingkungan, dan untuk mengatasinya dibutuhkan kemampuan yang melebihi kemampuan manusia”. Bencana alam menjadi salah satu peristiwa yang tidak dapat diprediksi kapan terjadinya dan tentunya sangat tidak diharapkan oleh semua makhluk hidup.

Bencana alam banyak ragamnya, tanah longsor, banjir, tsunami, gempa bumi, dan gunung meletus merupakan beberapa contoh dari bencana alam. Setiap bencana alam juga memiliki faktor-faktor sendiri yang dapat memicu terjadinya sebuah bencana alam. Bencana alam dapat terjadi kapanpun dan dimanapun, terkadang bencana alam dapat terjadi tanpa kita duga sebelumnya.

2.2 Banjir

Banjir merupakan keadaan dimana suatu wilayah berada pada posisi yang tergenang oleh air dalam volume yang begitu banyak dan begitu besar dan tidak dapat ditampung lagi oleh sungai, bendungan serta tidak dapat diserap lagi oleh tanah dan pohon yang berfungsi sebagai daerah resapan.

Pada dasarnya banjir tidak hanya terjadi didaerah yang rendah, banjir juga dapat terjadi ditempat yang tinggi. Karena banjir bisa terjadi dimana saja jika tempat itu tidak memiliki daerah resapan yang cukup. Terutama ketika musim penghujan tiba, hujan yang turun terus menerus menyebabkan air yang turun semakin banyak sehingga memenuhi sungai dan saluran-saluran air yang telah penuh. Dan jika tidak ada tempat lain lagi untuk menampung air hujan yang turun, maka disitu akan terjadi bencana banjir.

Banyak sekali faktor yang menyebabkan banjir, terutama curah hujan, kondisi topografi, jenis penutup tanah, jenis tanah, dan lain-lain. Penyebab tersebut

merupakan parameter-parameter yang dapat menyebabkan banjir. Faktor kemiringan lereng, ketinggian, dan geologi sangat menentukan daerah rentan banjir. Selanjutnya kondisi fisik ditambah dengan perilaku manusia akan menentukan tingkat kerawanannya [4].

2.3 Kecerdasan Buatan

Kecerdasan buatan merupakan proses di mana peralatan mekanik dapat melaksanakan kejadian-kejadian dengan menggunakan pemikiran atau kecerdasan seperti manusia[9].

Kecerdasan buatan (AI) merupakan sebuah studi tentang bagaimana membuat computer melakukan hal-hal yang pada saat ini dapat dilakukan lebih baik oleh manusia. Berikut beberapa pengertian kecerdasan buatan dari berbagai sudut pandang antara lain :

1. Sudut pandang kecerdasan

   Kecerdasan buatan akan membuat mesin menjadi ‘cerdas’.

2. Sudut pandang penelitian

   Kecerdasan buatan adalah suatu studi bagaimana membuat komputer dapat melakukan sesuatu sebaik yang dikerjakan oleh manusia. Domain yang sering dibahas para peneliti meliputi :

   a. *Mundane task*
      1. Persepsi (*vision and speech).*
      2. Bahasa alami (*understanding, generation and translation)*.
      3. Pemikiran yang bersifat *common sense* dan robot kontorl.

   b. *Formal task*
      1. Permainan atau games.
      2. Matematika (geometri, logika, kalkulus integral, pembuktian).

   c. *Expert task*
      1. Analisis *financial*
      2. Analisis *medical*
      3. Analisis ilmu pengetahuan
      4. Rekayasa (desain, pencarian kegagalan, perencanaan manufaktur).

3. Sudut pandang bisnis

   Kecerdasan buatan adalah kumpulan peralatan yang sangat *powerfull* dan metodologis dalam menyelesaikan masalah-masalah bisnis.

4. Sudut pandang pemograman

   Kecerdasan buatan meliputi studi tentang pemograman simbolik, penyelesaian masalah dan pencarian.

Dari pengertian-pengertian kecerdasan buatan tersebut, Winston dan Prendergast (1984) menjabarkan tujuan dari kecerdasan buatan adalah :

1. Membuat mesin menjadi lebih pintar (tujuan utama)
2. Memahami apa itu kecerdasan (tujuan ilmiah)
3. Membuat mesin lebih bermanfaat (tujuan entrepreneurial)

Kecerdasan buatan sendiri merupakan salah satu bagian ilmu komputer yang membuat agar komputer dapat melakukan pekerjaan seperti dan sebaik yang dilakukan manusia. Pada awal diciptakannya, komputer hanya difungsikan sebagai alat hitung saja. Namun seiring dengan perkembangan jaman, maka peran komputer semakin mendominasi kehidupan manusia. Komputer tidak lagi digunakan sebagai alat hitung, namun komputer diharapkan dapat melakukan segala sesuatu yang bisa dikerjakan oleh manusia.

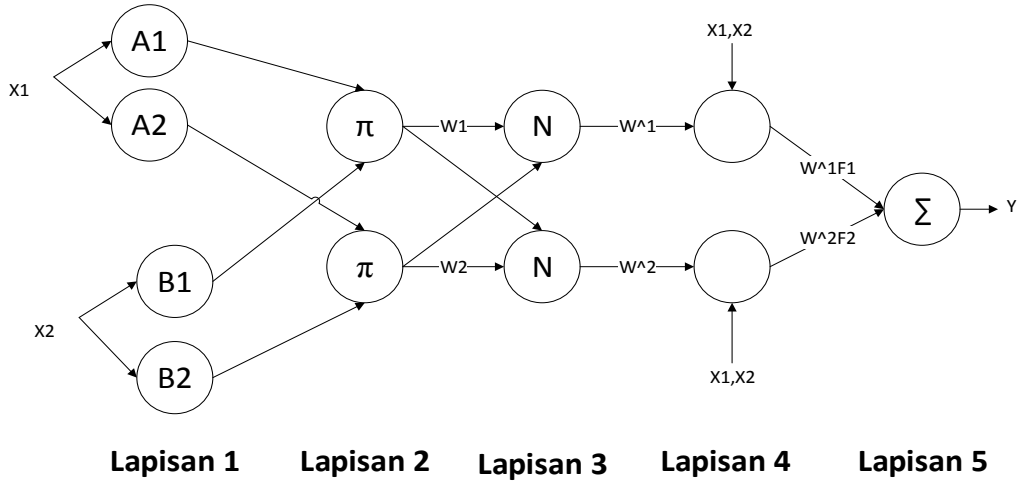
2.4 Adaptive Neuro Fuzzy Inference *System*

Adaptive Neuro Fuzzy Inference System (ANFIS) adalah arsitetktur yang secara fungsional sama dengan *fuzzy rule base* model Sugeno. Arsitektur ANFIS juga sama dengan jaringan syaraf dengan fungsi radial dengan sedikit batasan tertentu. Bisa dikatakan bahwa ANFIS adalah suatu metode yang mana dalam melakukan penyetelan aturan digunakan algoritma pembelajaran terhadap sekumpulan data. Pada ANFIS juga memungkinkan aturan-aturan untuk beradaptasi.

2.4.1 Langkah-langkah ANFIS

System inferensi fuzzy model Tagaki-Sugeno-Kang (TSK) direpresentasikan sebagai jaringan syaraf *feed forward* lima lapis khusus pada

struktur ANFIS yang digambarkan dalam arsitektur jaringan syaraf tiruan seperti betikut :



Gambar 2.1 Struktur ANFIS

Pada gambar 2.1 terlihat *system* inferensi fuzzy *system* yang digambarkan pada arsiterktur jaringan syaraf tiruan terdiri atas lima lapisan dengan fungsi yang berbeda- beda. Berikut merupakan penjelasan dari tiap lapisan yang ada dalam struktur ANFIS :

1. Lapisan 1

Fungsi keanggotaan yang dipakai adalah jenis *generalized bell* (*gbell*) yang dirumuskan sebagai berikut :

$$gbell(x, a, b, c) = \frac{1}{1 + \left|\frac{x - c}{a}\right|^{2b}} \qquad \ldots (2.1)$$

Koefisien b: koefisien adalah angka positif, dan biasanya bernilai 1. Jumlah tersebut terkait dengan bagian atas kurva dan menandakan bahwa kurva berbentuk lonceng dan penyebab naik ke bagian datar. Koefisien a menyatakan lebar dari titik pusat hingga batas keanggotaan dan koefisien c menyatakan titik pusat *membership function*. Nilai a,b,c dan jumlah himpunan masukan fuzzy (x) ditentukan pada awal proses dan akan berubah melalui proses pembelajaran dinamakan parameter premis yang adaptif.

2. Lapisan 2

Semua simpul pada lapisan ini adalah nonadaptif (parameter tetap). Fungsi simpul ini adalah mengalikan setiap sinyal masukan yang datang. Fungsi simpul :

$$O_{2.i} = w_i = \mu_{A_i}(x).\mu_{B_i}(y), untuk\ i = 1,2 \quad \ldots(2.2)$$

Keterangan :

x	=	Nilai masukan
y	=	Nilai masukan
w_i	=	Bobot ke-i
μ_i	=	Nilai keanggotaan kelas-i
$O2_{,i}$	=	Nilai keluaran lapisan 2 ke-i

Tiap keluaran simpul menyatakan derajat pengaktifan (*firing strength*) tiap aturan fuzzy. Fungsi ini dapat diperluas apabila bagian premis memiliki lebih dari dua himpunan fuzzy. Banyaknya simpul pada lapisan ini menunjukkan banyaknya aturan yang dibentuk. Fungsi perkalian yang digunakan adalah interpretasi kata hubung *and* dengan menggunakan operator *t-norm*.

3. Lapisan 3

Lapisan Setiap simpul pada lapisan ini adalah simpul nonadaptif yang menampilkan fungsi derajat pengaktifan ternomalisasi (*normalized firing strength*) yaitu rasio keluaran simpul ke-i pada lapisan sebelumnya terhadap seluruh keluaran lapisan sebelumnya, dengan bentuk fungsi simpul:

$$O_{3.i} = \overline{w}_i = \frac{w_i}{w_1 + w_2}, untuk\ i = 1,2 \quad \ldots(2.3)$$

Keterangan :

w_i	=	Bobot ke-i
$\overline{w}_i$	=	derajat pengaktifan ternomalisasi ke-i
$O_{3,i}$	=	Nilai keluaran lapisan 3 ke-i

Apabila dibentuk lebih dari dua aturan, fungsi dapat diperluas dengan membagi w_i dengan jumlah total w untuk semua aturan.

4. Lapisan 4

Setiap simpul pada lapisan ini adalah simpul adaptif dengan fungsi simpul :

$$O_{4.i} = \overline{w}_i f_i = \overline{w}_i(p_i x + q_i y + r_i) \quad \ldots(2.4)$$

dengan $\overline{w}_i$ adalah derajat perngaktifan ternormalisasi dari lapisan 3 dan parameter p, q, r menyatakan parameter konsekuen yang adaptif.

Keterangan :

$\bar{w}_i$ = derajat pengaktifan ternomalisasi ke-I

f_i = sinyal kontrol ke –i

$O_{4,i}$ = Nilai keluaran lapisan 4 ke-i

5. Lapisan 5

Pada lapisan ini hanya ada satu simpul tetap yang fungsinya untuk menjumlahkan semua masukan. Fungsi simpul :

$$O_{5.i} = \sum_i \bar{w}_i f_i \qquad \text{... (2.5)}$$

Keterangan :

$\bar{w}_i$ = derajat pengaktifan ternomalisasi ke-I

f_i = sinyal kontrol ke –i

$O_{5,i}$ = Nilai keluaran lapisan 5 ke-i

Jaringan adaptif dengan lima lapisan tersebut ekivalen dengan system inferensi fuzzy TSK.

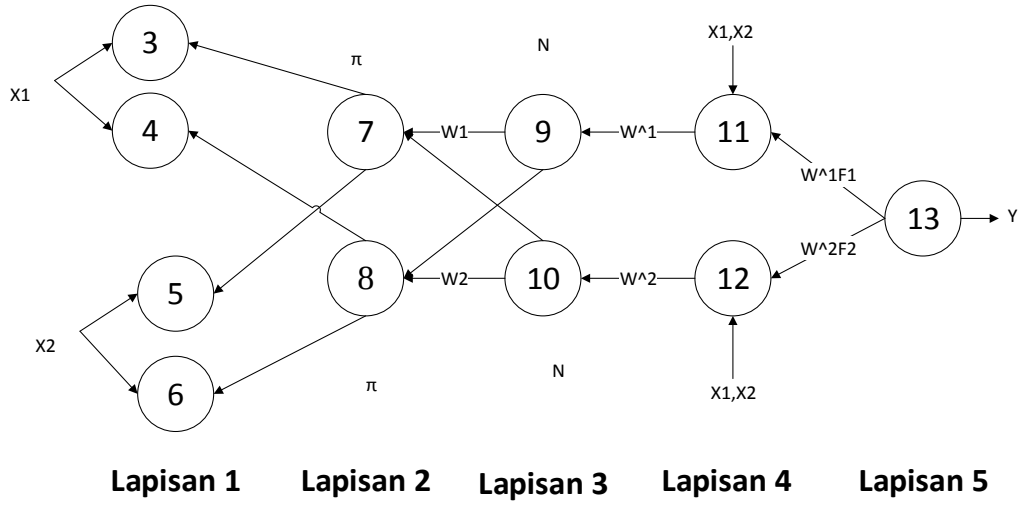
2.4.2 Proses Belajar ANFIS

Proses belajar yang ada digunakan untuk memperbaharui parameter premis dan parameter konsukuen yang ada pada jaringan ANFIS. Pembelajaran *hybrid* yang dilakukan terdiri atas dua bagian yaitu arah maju (*forward pass*) dan arah mundur (*backward pass*).

Pada arah maju, parameter premis dibuat tetap. Dengan menggunakan metode *Recursive Least Square Estimator* (RLSE), parameter konsekuen diperbaiki berdasarkan pasangan data masukan – keluaran. Metode RLSE dapat diterapkan karena parameter konsekuen yang diperbaiki adalah parameter linier. Metode RLSE akan mempercepat proses belajar *hybrid*. Kemudian setelah parameter konsekuen didapatkan, data masukan dilewatkan jaringan adaptif kembali dan hasil keluaran jaringan adaptif ini dibandingkan dengan keluaran yang sebenarnya.

Pada arah mundur, parameter konsekuen dibuat tetap. Kesalahan yang terjadi antara keluaran jaringan adaptif dan keluaran sebenarnya dipropagasikan balik dengan menggunakan gradient descent untuk memperbaiki parameter premis. Pembelajaran ini dikenal sebagai algoritma *Backpropagation-error*.

Berikut arsitektur jaringan pada saat menghitung *Backpropagation-error* :



Gambar 2.2 Arsitektur Jaringan *Backpropagation-error*

1. *Error* pada lapisan ke-5

Apabila kita memiliki jaringan seperti pada Gambar 2.2, yang hanya memiliki 1 simpul pada lapisan *output* (simpul 13), maka propagasi *error* yang menuju pada lapisan ke-5 dapat dirumuskan sebagai berikut :

$$\varepsilon_{13} = \frac{\partial E_p}{\partial X_{13}} = -2\left(y_p - y_p^*\right) \quad \text{... (2.6)}$$

Dengan y_p adalah target *output* data pelatihan ke-p, dan y_p^* adalah output jaringan pada data pelatihan ke-p

2. *Error* pada lapisan ke-4

Dengan melihat gambar 2.2, propagasi *error* yang menuju pada lapisan 4, yaitu simpul 11 dan 12, sehingga dapat dirumuskan sebagai berikut :

$$\varepsilon_{11} = \left(\frac{\partial E_p}{\partial X_{13}}\right)\left(\frac{\partial f_{13}}{\partial X_{11}}\right) = \varepsilon_{13}\left(\frac{\partial f_{13}}{\partial X_{11}}\right) \quad \text{... (2.7)}$$

$$\varepsilon_{12} = \left(\frac{\partial E_p}{\partial X_{13}}\right)\left(\frac{\partial f_{13}}{\partial X_{12}}\right) = \varepsilon_{13}\left(\frac{\partial f_{13}}{\partial X_{12}}\right) \quad \text{... (2.8)}$$

3. *Error* pada lapisan ke-3

Dengan melihat gambar 2.2, propagasi error yang menuju pada lapisan 3, yaitu simpul 9 dan simpul 10, sehingga dapat dirumuskan sebagai berikut :

$$\varepsilon_9 = (\frac{\partial E_p}{\partial X_{13}})(\frac{\partial f_{13}}{\partial X_{11}})(\frac{\partial f_{11}}{\partial X_9}) = \varepsilon_{11}(\frac{\partial f_{11}}{\partial X_9}) \quad \text{... (2.9)}$$

$$\varepsilon_{10} = (\frac{\partial E_p}{\partial X_{13}})(\frac{\partial f_{13}}{\partial X_{12}})(\frac{\partial f_{12}}{\partial X_{10}}) = \varepsilon_{12}(\frac{\partial f_{12}}{\partial X_{10}}) \quad \text{... (2.10)}$$

4. *Error* pada lapisan ke-2

Dengan melihat gambar 2.2, propagasi *error* yang menuju pada lapisan ke-2 yaitu simpul 7 dan simpul 8, sehingga dapat dirumuskan sebagai berikut :

$$\varepsilon_7 = \left(\frac{\partial E_p}{\partial X_{13}}\right)\left(\frac{\partial f_{13}}{\partial X_{11}}\right)\left(\frac{\partial f_{11}}{\partial X_9}\right)\left(\frac{\partial f_9}{\partial X_7}\right) + \left(\frac{\partial E_p}{\partial X_{13}}\right)\left(\frac{\partial f_{13}}{\partial X_{12}}\right)\left(\frac{\partial f_{12}}{\partial X_{10}}\right)\left(\frac{\partial f_{10}}{\partial X_7}\right) \quad \text{... (2.11)}$$

$$\varepsilon_8 = \left(\frac{\partial E_p}{\partial X_{13}}\right)\left(\frac{\partial f_{13}}{\partial X_{12}}\right)\left(\frac{\partial f_{12}}{\partial X_{10}}\right)\left(\frac{\partial f_{10}}{\partial X_8}\right) + \left(\frac{\partial E_p}{\partial X_{13}}\right)\left(\frac{\partial f_{13}}{\partial X_{11}}\right)\left(\frac{\partial f_{11}}{\partial X_9}\right)\left(\frac{\partial f_9}{\partial X_8}\right) \quad \text{... (2.12)}$$

5. *Error* pada lapisan ke-1

Dengan melihat gambar 2.2, propagasi *eror* yang menuju pada lapisan 1, yaitu simpul 3, 4, 5, dan 6 sehingga dapat dirumuskan sebagai berikut :

$$\varepsilon_3 = \varepsilon_7(\left(\frac{\partial f_7}{\partial X_3}\right)) \quad \text{... (2.13)}$$

$$\varepsilon_4 = \varepsilon_8\left(\left(\frac{\partial f_8}{\partial X_4}\right)\right) \quad \text{... (2.14)}$$

$$\varepsilon_5 = \varepsilon_7\left(\left(\frac{\partial f_7}{\partial X_5}\right)\right) \quad \text{... (2.15)}$$

$$\varepsilon_6 = \varepsilon_8\left(\left(\frac{\partial f_8}{\partial X_6}\right)\right) \quad \text{... (2.16)}$$

2.5 *Entity-relationship Diagram* (ERD)

ERD adalah diagram grafikal keseluruan struktur logikadari sebuah basis data. ERD tidak menggambarkan aliran data atau proses data, tetapi ERD lebih menggambarkan data pada data *store.*

ERD ini berfungsi untuk menggambarkan relasi dari dua tabel yang dapat digolongkan dalam tiga macam bentuk relasi, yaitu relasi satu ke satu, satu kebanyak dan relasi banyak ke banyak.

Model ERD yang berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta yang ditinijau.

2.6 Diagram Konteks

Diagram Konteks adalah sebuah diagram yang mencakup masukan-maskan dasar, *system* umum dan keluaran. Diagram ini merupakan tingkatan tertinggi dalam diagram aliran data dan hanya memuat satu proses. Menunujukan *system* secara keseluruhan, diagram konteks tidak memuat penyimpanan dan penggambaran aliran data yang sederhana. Semua entitas luar yang ditunjukan pada diagram konteks berikut aliran data-aliran data utama yang menuju dan dari *system*.

2.7 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah representasi grafik dari sebuah *system*. DFD menggambarkan komponen-komponen sebuah *system*, aliran-aliran data dari komponen-komponen yang ada mulai dari aliran asal, tujuan, dan penyimpanan dari data tersebut.

DFD digunakan untuk dua hal utama, yaitu untuk membuat dokumentasi dari *system* yang ada atau untuk menyusun dokumentasi untuk *system* yang baru. DFD merupakan alat bantu dari pengembangan sebuah bantu dari pengembangan sebuah *system* yang dibangun secara terstruktur atau procedural.

2.8 Kamus Data

Kamus data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu *system* informasi. Kamus data dapat mendefinisikan data yang mengalir pada *system* dengan lengkap. Kamus data juga dapat digunakan pada tahap analisa dan perancangan *system*.

Dengan adanya kamus data, didapat definisi-definisi dari bentuk-bentuk yang kurang dimengerti dalam isi DFD seperti aliran data, file, proses, dan elemen-elemen data.

2.9 Flowchart

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urut- urutan prosedur dari suatu program. *Flowchart* menolong *analyst* dan *programmer* untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian. *Flowchart* biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut. [5]

Flowchart adalah bentuk gambar/diagram yang mempunyai aliran satu atau dua arah secara *sekuensial*. *Flowchart* digunakan untuk merepresentasikan maupun mendesain program. Oleh karena itu *flowchart* harus bisa merepresentasikan komponen-komponen dalam bahasa pemrograman. [5]

2.10 MySQL

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis. Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Kehandalan suatu sistem basisdata (DBMS) dapat diketahui dari cara kerja pengoptimasi-nya dalam melakukan proses perintah-perintah SQL yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. Sebagai peladen basis data, MySQL mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional. Pada modus operasi non-transaksional, MySQL dapat dikatakan unggul dalam hal unjuk kerja dibandingkan perangkat lunak peladen basisdata kompetitor lainnya. Namun demikian pada modus non-transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modus non-transaksional hanya cocok untuk jenis aplikasi yang tidak

membutuhkan reliabilitas data seperti aplikasi blogging berbasis web (wordpress), CMS, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modus basisdata transaksional, hanya saja sebagai konsekuensinya unjuk kerja MySQL pada modus transaksional tidak secepat unjuk kerja pada modus non-transaksional. Adapun beberapa keistimewaan dari MySQL sehingga membuat MySQL dipilih oleh kebanyakan orang :

1. Portabilitas

   MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. Perangkat lunak sumber terbuka (open source)

   MySQL didistribusikan sebagai open source sehingga dapat digunakan secara gratis.
3. Multi-user

   MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. Performance tuning

   MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. Ragam tipe data

   MySQL memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
6. Perintah dan Fungsi

   MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (*query*).
7. Keamanan

   MySQL memiliki beberapa lapisan keamanan seperti password yang terenkripsi.
8. Skalabilitas dan Pembatasan

MySQL mampu menangani basis data dalam skala besar, dengan jumlah record lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

9. Konektivitas

   MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix soket (UNIX), atau named pipes (NT).

10. Lokalisasi

    MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.

11. Antar Muka

    MySQL memiliki antar muka (interface) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).

12. Klien dan Peralatan

    MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk online.

13. Struktur table

    MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya semacam PostgreSQL ataupun Oracle.

2.11 Microsoft Visual Studio

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi console, aplikasi Windows, ataupun aplikasi Web. Visual Studio mencakup kompiler, SDK, Integrated Development Environment (IDE), dan dokumentasi (umumnya berupa MSDN Library). Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual

Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe.

Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam native code (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun managed code (dalam bentuk Microsoft Intermediate Language di atas .NET Framework). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi Silverlight, aplikasi Windows Mobile (yang berjalan di atas .NET Compact Framework).

Visual Studio kini telah menginjak versi Visual Studio 9.0.21022.08, atau dikenal dengan sebutan Microsoft Visual Studio 2008 yang diluncurkan pada 19 November 2007, yang ditujukan untuk platform Microsoft .NET Framework 3.5. Versi sebelumnya, Visual Studio 2005 ditujukan untuk platform .NET Framework 2.0 dan 3.0. Visual Studio 2003 ditujukan untuk .NET Framework 1.1, dan Visual Studio 2002 ditujukan untuk .NET Framework 1.0. Versi-versi tersebut di atas kini dikenal dengan sebutan Visual Studio .NET, karena memang membutuhkan Microsoft .NET Framework. Sementara itu, sebelum muncul Visual Studio .NET, terdapat Microsoft Visual Studio 6.0 (VS1998).

2.12 Pengujian *Matrix Confusion*

Matriks Confusion adalah sebuah matriks yang memberikan gambaran mengenai tingkat kesalahan atau *eror rate* dan kualitas prediksi sebuah model. Ukuran dimensi *matriks confusion* adalah L x L yang dimana L merupakan jumlah nilai yang ingin dibandingkan [6]. *Matriks Confusion* berisi informasi tentang klasifikasi actual dan diperkirakan yang dilakukan oleh *system* klasifikasi.

Tabel 2.1 *Matriks Confusion*

		Prediksi	
		Negative	*Positive*
Aktual	*Negative*	a	b
	Positive	c	d

Setiap kolom pada *matriks confusion* mewakili contoh di kelas yang dilakukan oleh *system*, sedangkan setiap baris mewakili contoh di kelas yang sudah ditentukan. Setelah mengetahui nilai a, b, c, dan d maka selanjutnya melakukan perhitungan akurasi, perhitungan akurasi menggunakan rumus berikut :

$$Akurasi = \frac{a + d}{a + b + c + d} \quad \text{... (2.17)}$$