# Introduction to Machine Learning in Python

Pens, Sabtu 16 Maret 2019

# Introduction Python, Anaconda, & Jupyter Notebook


Python adalah Bahasa pemograman tingkat tinggi yang telah berkembang sejak tahun 1991. Saat ini sudah banyak digunakan untuk pengembangan '*machine learning*'


Jupyter Notebook adalah aplikasi open source web yang dapat digunakan untuk membuat dan berbagi dokumen yang berisi *live code*, persamaan, visualisasi, dan teks.

# Introduction Python, Anaconda, & Jupyter Notebook

Why use python?

- Readibility, error yang terjadi disaat melakukan koding muda dipahami, sehingga mudah diatasi.
- General Purpose, umumnya digunakan untuk pemrograman '*machine learning*'

```cpp
#include "lib/system.hpp"
#include "lib/microphone.hpp"

#include "core.hpp"
#include "sessions/session.hpp"
#include "windows/main.hpp"
#include "wizards/quickstart/wizard.hpp"

using namespace SpeechControl;
using namespace SpeechControl::Wizards;

using SpeechControl::Core;

Core* Core::s_inst = 0;

/// @todo Generate default settings.
/// @todo Invoke first-run wizard.
/// @todo Pass command-line arguments to Qt5 *after* QApplication gets them.
Core::Core(int argc,char** argv) : QObject(new QApplication(argc,argv)){
    s_inst = this;

    // start application.
    QApplication* l_app = qobject_cast<QApplication*>(QApplication::instance());
    l_app->setApplicationName("SpeechControl");
    l_app->setOrganizationDomain("thesii.org");
    l_app->setOrganizationName("Synthetic Intellect Institute");
    l_app->setApplicationVersion("0.0b");

    System::start(&argc,&argv);
    Session::init();

    QDir l_dir;
    l_dir.mkdir(QDir::homePath() + "/.speechcontrol/contents");

    // build settings
    m_settings = new QSettings(QSettings::UserScope,"Synthetic Intellect Institute","SpeechControl",this);

    // check for microphones
    if (Microphone::allMicrophones().empty()){
        QErrorMessage* l_msg = new QErrorMessage;
        l_msg->setWindowTitle("No Microphones Found");
        l_msg->showMessage(tr("No microphones were found on your system. Please ensure that you have one installed and detectable by ") +
                           tr("the audio system and make sure that <b>gstreamer-plugins-good</b> is installed on your system."),
                             "noMicrophonesFoundOnStart");
    }
}

Core::~Core () {
    m_settings->sync();
}

void Core::start() {
    Windows::Main* l_mw = new Windows::Main;

    if (!s_inst->m_settings->contains("User/Name")){
        if (QMessageBox::question(l_mw,tr("First Run"),
                                  tr("This seems to be the first time you've run SpeechControl on this system. "
                                     "A wizard allowing you to start SpeechControl will appear."),QMessageBox::Yes,QMessageBox::No) == QMessageBox::Yes){
            QuickStart* l_win = new QuickStart(l_mw);
            l_win->exec();
        }
    }

    l_mw->show();
}

void Core::stop() { }

QVariant Core::getConfig(const QString &p_pth, QVariant p_vrt) const {
    return m_settings->value(p_pth,p_vrt);
}
```

# Introduction Python, Anaconda, & Jupyter Notebook

**Number**

| • **Type** | • **Format** | • **Description** |
|---|---|---|
| • int | • a = 10 | • Signed Integer |
| • float | • a = 45.67 | • (.) Floating point real values |
| • complex | • a = 3.14J | • (J) Contains integer in the range 0 to 255. |

Example

- a = 5
- print(a, "is of type", type(a))
- a = 2.0
- print(a, "is of type", type(a))
- a = 1+2j
- print(a, "is complex number?", isinstance(1+2j,complex))

# Introduction Python, Anaconda, & Jupyter Notebook

## String

Create string variables by enclosing characters in quotes. Python uses single quotes ' double quotes " and triple quotes """ to denote literal strings. Only the triple quoted strings """ also will automatically continue across the end of line statement.

```
Var1 = ‘helloworld’
Var2 = ‘workshoppens’

print var1[2]
print var2[1:4]
```

# Introduction Python, Anaconda, & Jupyter Notebook

## Arithmetic Operators

| Operator | Meaning | Example |
|---|---|---|
| + | Add two operands or unary plus | x+y |
| - | Substract right operand from the left or unary minus | x-y |
| * | Multiply two operands | x*y |
| / | Divide left operand by the right one (always result into float) | x / y |
| % | Modulus - remainder of the division of left operand by the right | x % y (remainder of x/y) |
| // | Floor division - division that results into whole number adjusted to the left in the number line | x // y |
| ** | Exponent - left operand raised to the power of right | x**y (x to the power y) |

# Introduction Python, Anaconda, & Jupyter Notebook

## Comparison Operators

| Operator | Meaning | Example |
|---|---|---|
| > | Greater that - True if left operand is greater than the right | x > y |
| < | Less that - True if left operand is less than the right | x < y |
| == | Equal to - True if both operands are equal | x == y |
| != | Not equal to - True if operands are not equal | x != y |
| >= | Greater than or equal to - True if left operand is greater than or equal to the right | x >= y |
| <= | Less than or equal to - True if left operand is less than or equal to the right | x <= y |

# Introduction Python, Anaconda, & Jupyter Notebook

## Logical operators

| Operator | Meaning | Example |
|---|---|---|
| and | True if both the operands are true | x and y |
| or | True if either of the operands is true | x or y |
| not | True if operand is false (complements the operand) | not x |

# Introduction Python, Anaconda, & Jupyter Notebook

Identity operators

| • Operator | • Meaning | • Example |
|---|---|---|
| • is | • True if the operands are identical (refer to the same object) | • x is True |
| • is not | • True if the operands are not identical (do not refer to the same object) | • x is not True |

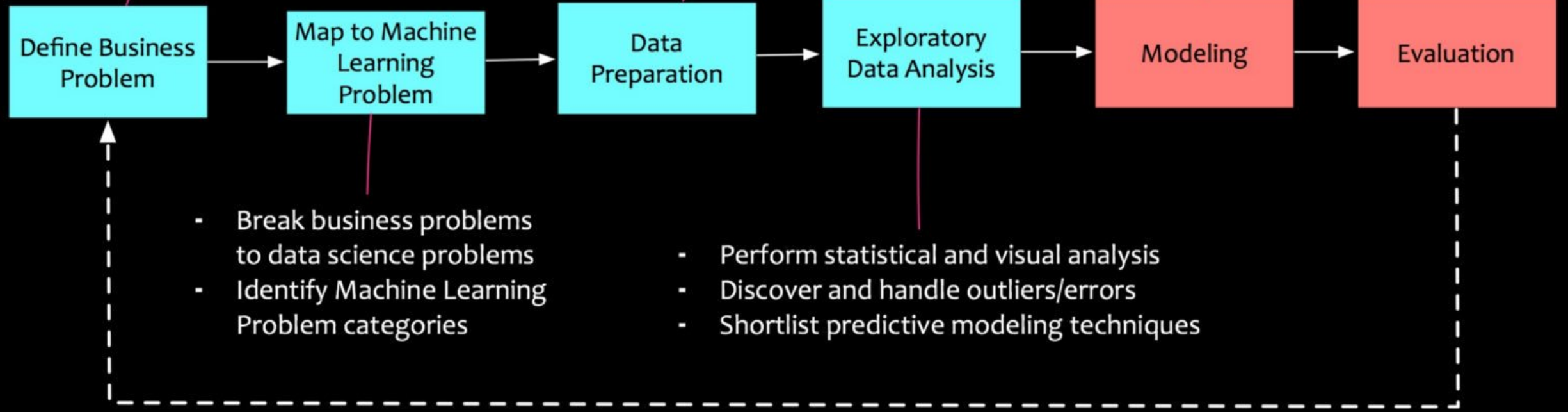
Membership operators

| Operator | Meaning | Example |
|---|---|---|
| in | True if value/variable is found in the sequence | 5 in x |
| not in | True if value/variable is not found in the sequence | 5 not in x |

- Clearly defined business problem
- Set success criteria
- Define clear data science objectives

- Understand data points and constraints
- Formulate data analytics strategy
- Perform required transformation

- Experiment with multiple models
- Choose the most optimal model
- Create a feedback loop




- Break business problems to data science problems
- Identify Machine Learning Problem categories

- Perform statistical and visual analysis
- Discover and handle outliers/errors
- Shortlist predictive modeling techniques

# Apa itu *'machine learning'*?

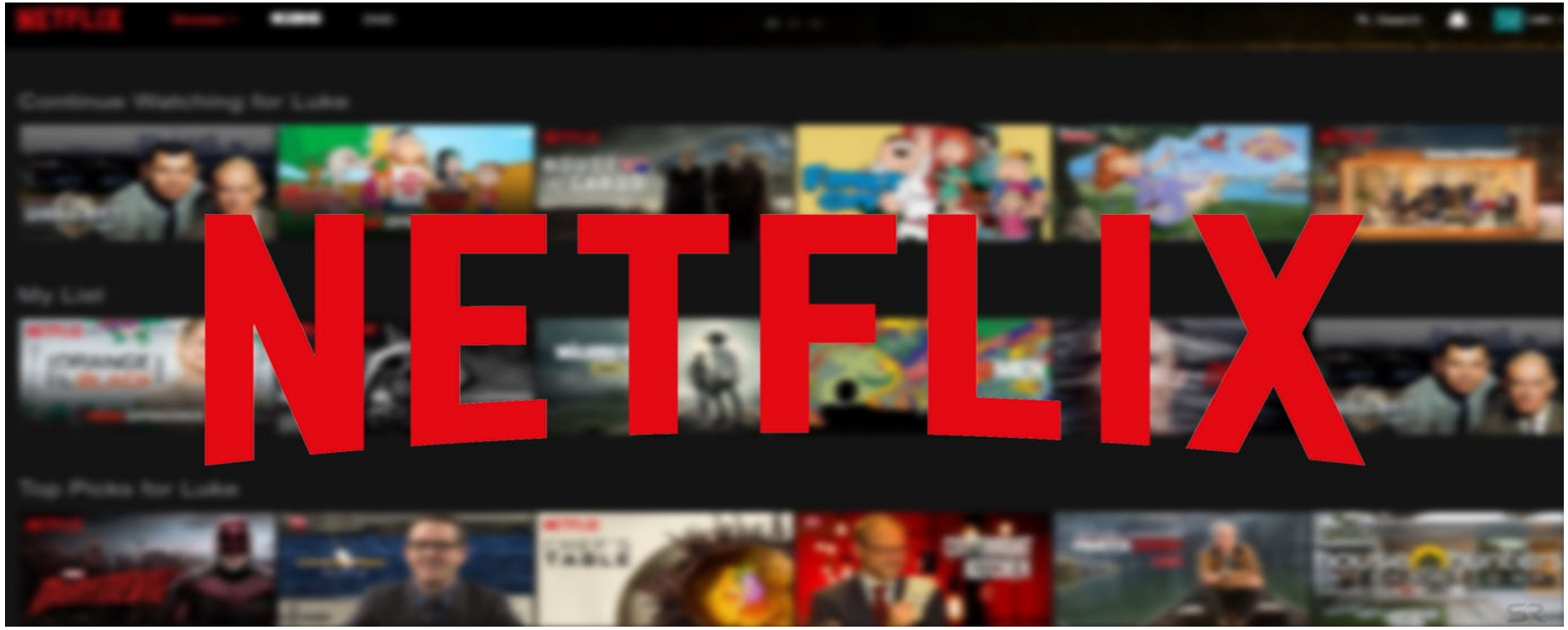
Field of study that gives "*computers the ability to learn being explicitly programmed*"

Bidang Study yang memberikan "kemampuan pada computer untuk belajar diprogram secara eksplisit"

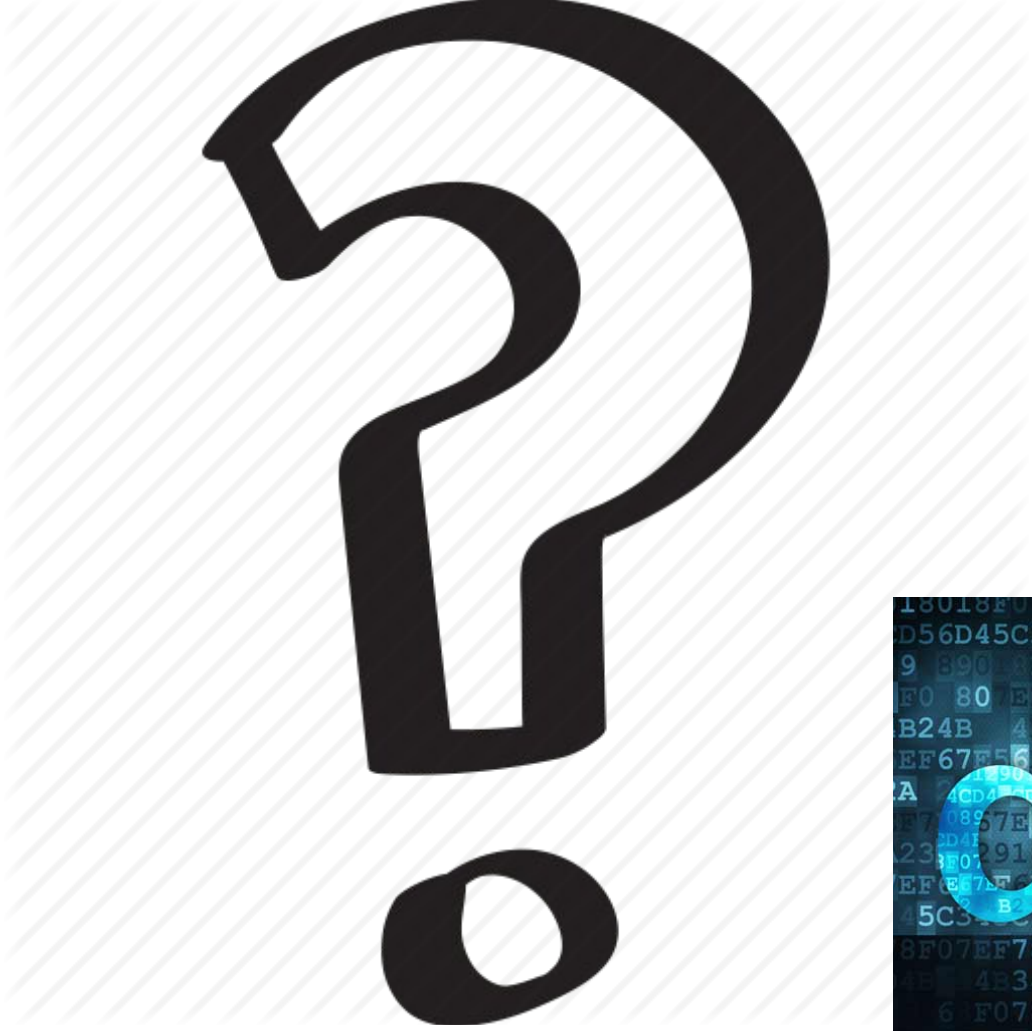
--Arthur Samuel, 1959

# Beberapa contoh didunia nyata

# Rekomendasi film dari Netflix

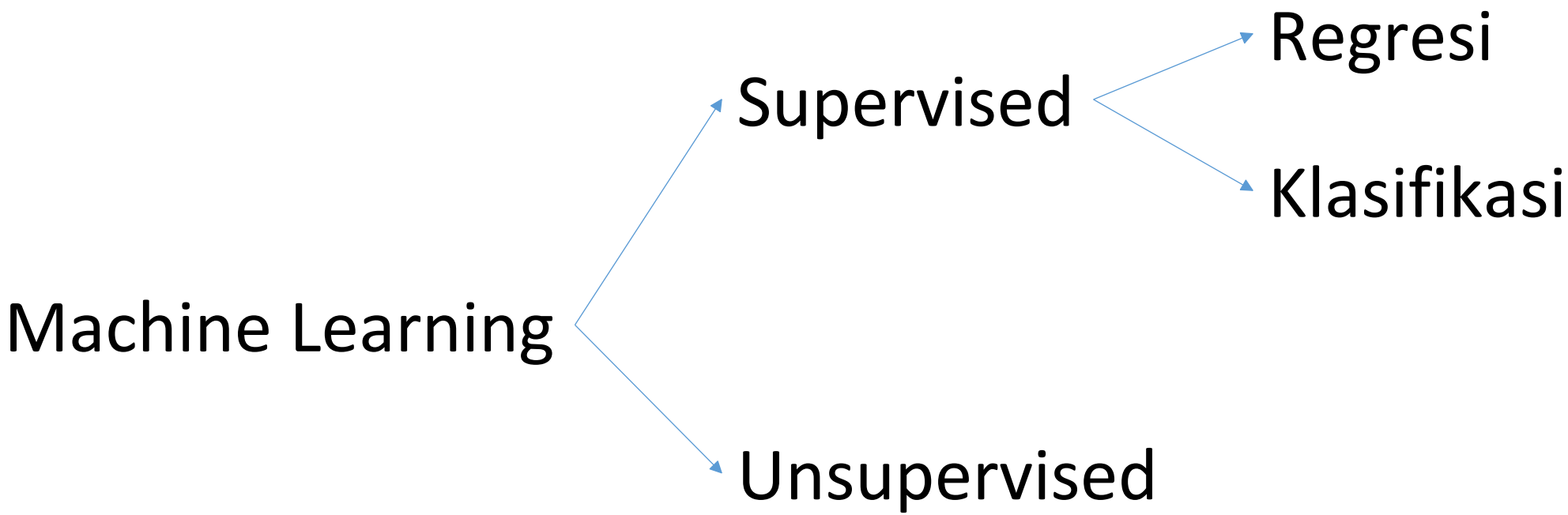


# Rekomendasi Produk dari Lazada

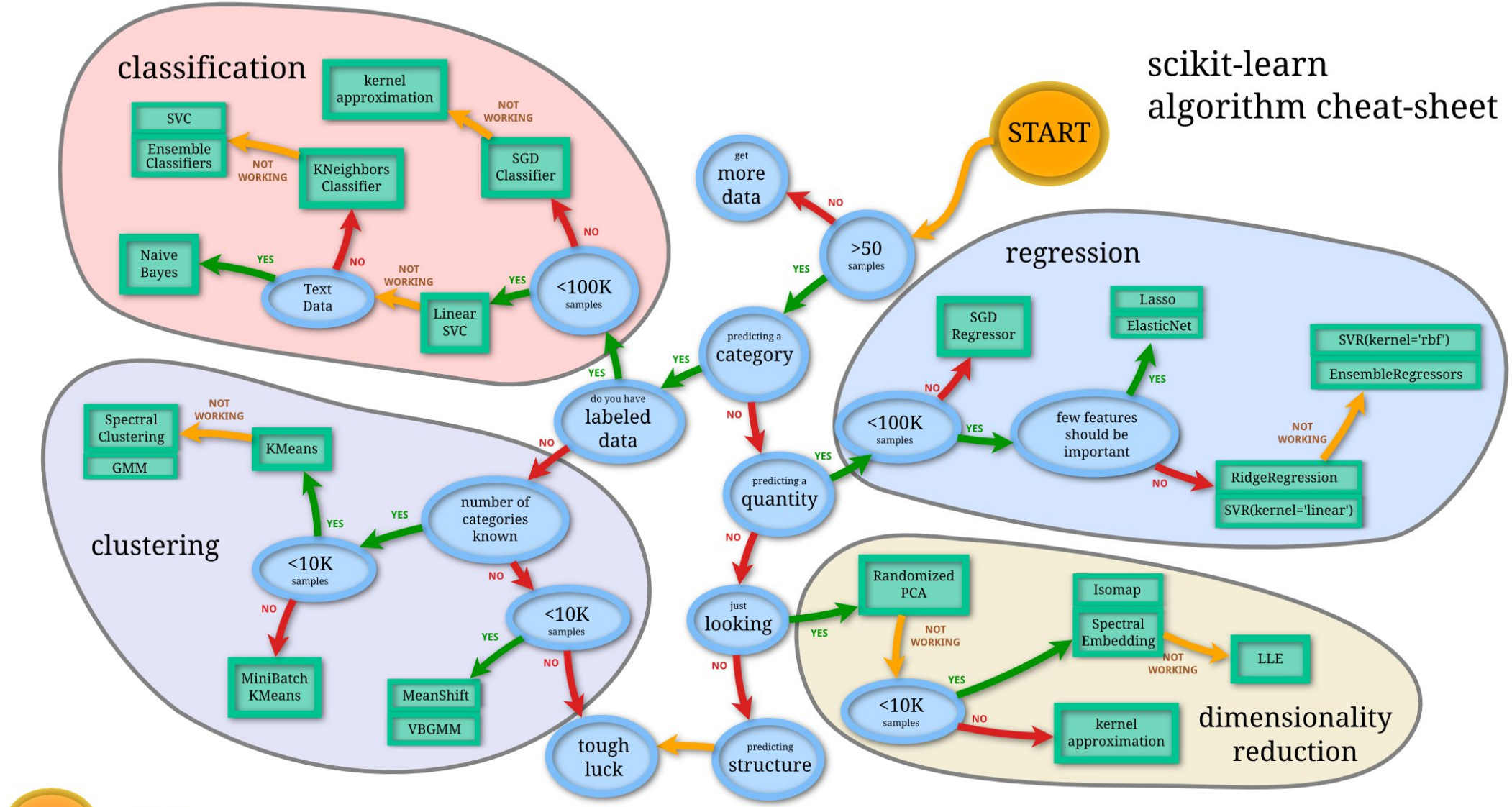


Bagaimana cara melakukannya?
-Mengumpulkan data(?)


+


Machine Learning
Supervised
Regresi
Klasifikasi
Unsupervised

# scikit-learn algorithm cheat-sheet

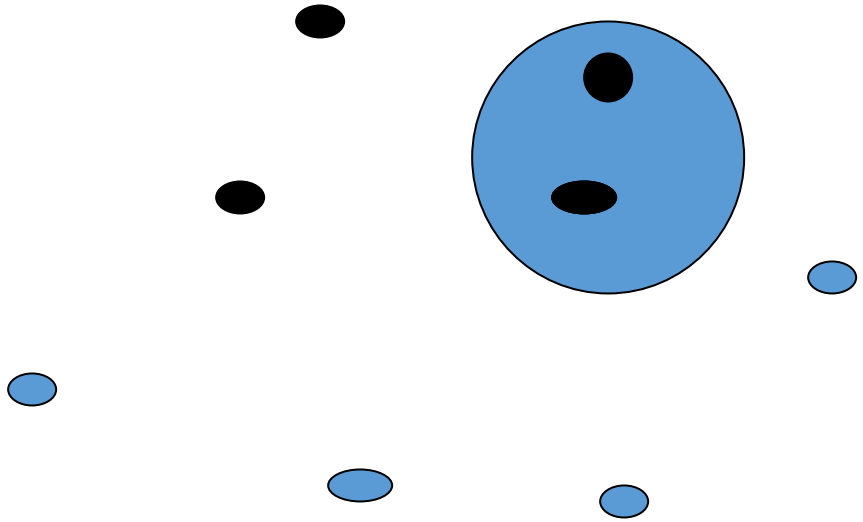


# K Nearest Neighbors (KNN)

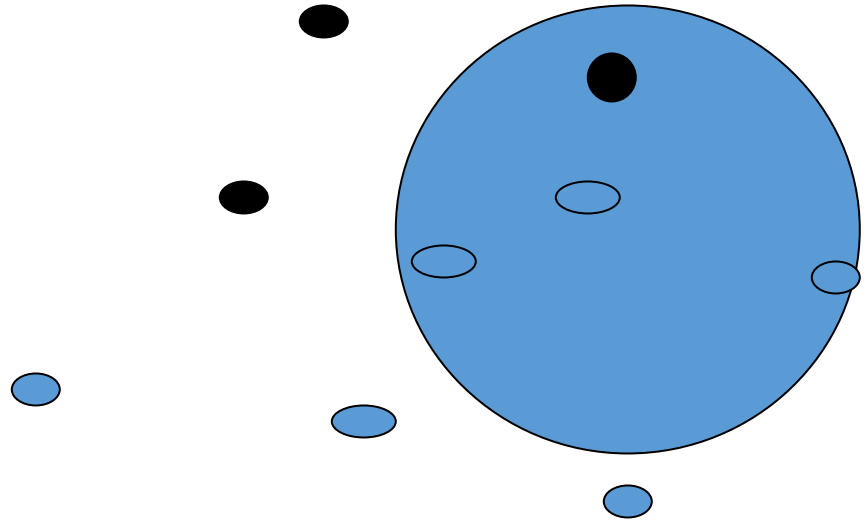
- KNN adalah algoritma supervised learning, yang berarti menggunakan label kelas dari data train selama fase ‘learning’nya. Algoritma knn merupakan mesin learning berbasis instance, di mana titik data baru diklasifikasikan berdasarkan contoh yang disimpan dan berlabel (titik data). KNN dapat digunakan untuk klasifikasi dan regresi.

KNN memerlukan beberapa hal

- Features Space (Data Train)
- Matriks Jarak
- Nilai K



1 Nearest Neighbor



3 Nearest Neighbors

# Algrotima KNN Secara Ringkas

- Pertama, buka data dan tentukan nilai k
- Selanjutnya, hitung jarak antara data yang ingin diklasifikasikan dengan data yang telah tersimpan
- Urutkan nilai jarak (ascending atau descending), kemudian tentukan jumlah tetangga terdekatnya
- Label k-tetangga terdekat dikumpulkan, dan suara terbanyak untuk mengklasifikasikan titik data baru.
- Selesai

## Distance functions

Euclidean

$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

Manhattan

$$\sum_{i=1}^{k}|x_i - y_i|$$

Minkowski

$$\left(\sum_{i=1}^{k}\left(|x_i - y_i|\right)^q\right)^{1/q}$$

# Contoh

| • Nama | • Daya Tahan | • Kekuatan | • Kelas |
|---|---|---|---|
| • A26 | • 7 | • 7 | • Premium |
| • B78 | • 7 | • 4 | • Premium |
| • C32 | • 3 | • 4 | • Standart |
| • D55 | • 1 | • 4 | • Standart |

Test data -> daya tahan = 3, kekuatan = 7, kelas = ?

| • Nama | • Daya Tahan | • Kekuatan | • Kelas | • Distance |
|---|---|---|---|---|
| • A26 | • 7 | • 7 | • Premium | • $Sqrt((7-3)^2+(7-7)^2)=4$ |
| • B78 | • 7 | • 4 | • Premium | • 5 |
| • C32 | • 3 | • 4 | • Standart | • 3 |
| • D55 | • 1 | • 4 | • Standart | • 3.6 |

| • Nama | • Daya Tahan | • Kekuatan | • Kelas | • Distance | • Rank |
|---|---|---|---|---|---|
| • A26 | • 7 | • 7 | • Premium | • 4 | • 3 |
| • B78 | • 7 | • 4 | • Premium | • 5 | • 4 |
| • C32 | • 3 | • 4 | • Standart | • 3 | • 1 |
| • D55 | • 1 | • 4 | • Standart | • 3.6 | • 2 |

Kemudian gunakan nilai K yang telah ditentukan guna mendapatkan kelas pada data test.

# Live Coding