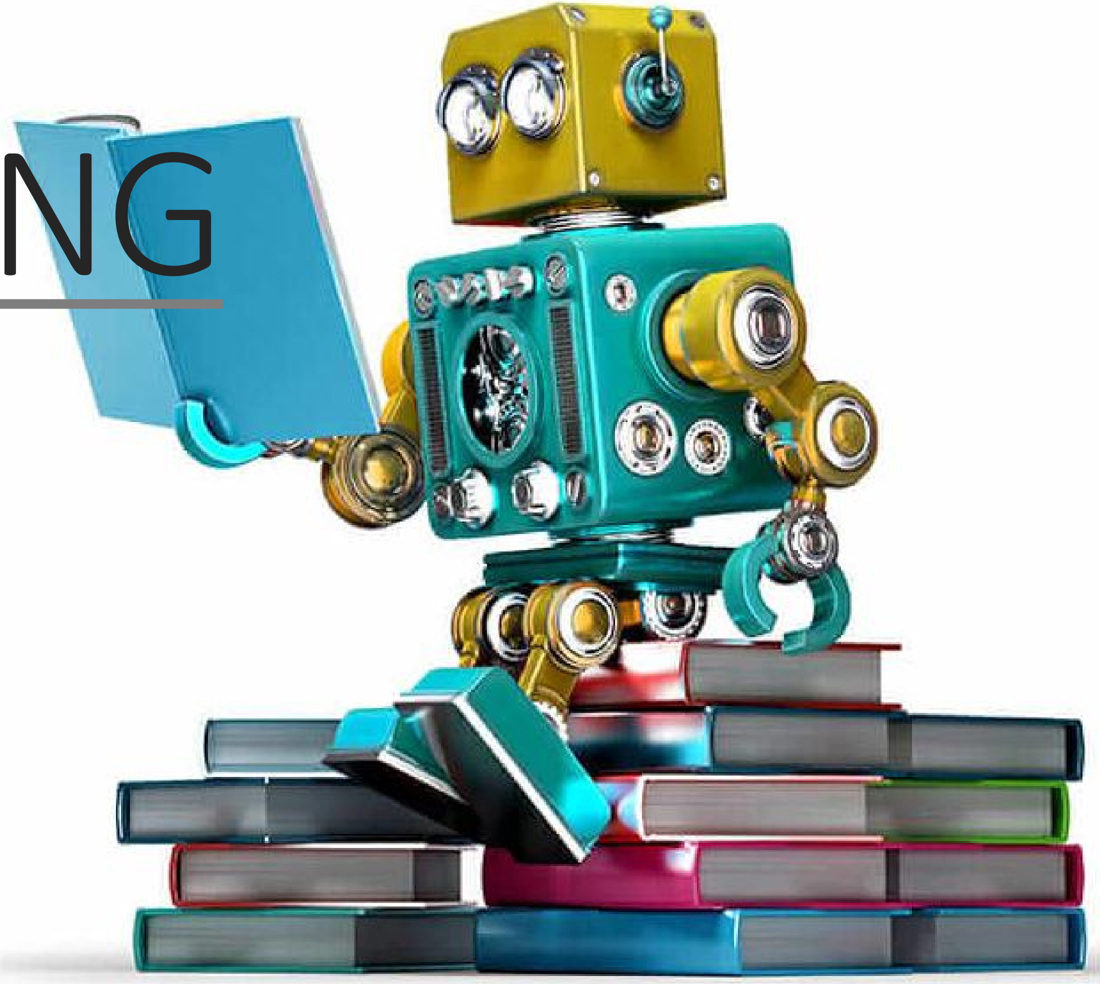
MACHINE LEARNING

LAB2 Lab Preliminary



贾艳红 Jana
Email:jiayh@mail.sustech.edu.cn

OBJECTIVES

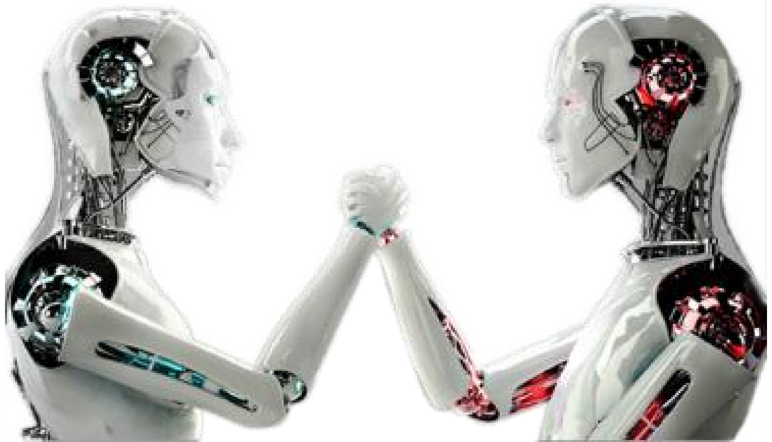
01 Understanding and Preprocessing Data

02 Evaluating Machine Learning Algorithms

03 Lab Task

PART ONE

Understanding and Preprocessing Data



Machine learning-Outline

- Raw Data and Feature Representation:
  - ✓ Concepts, instances, attributes
- Pills of Statistics
  - ✓ Sampling, mean, variance, standard deviation, normalization, standardization, etc.
- Data Visualization
  - ✓ how to read a histogram, scatter plot, etc

What is data?

- Data is a collection of facts, such as numbers, words, measurements, observations or even just descriptions of things.

- Data can be qualitative or quantitative

  – Qualitative data is descriptive information (it describes something)

  – Quantitative data is numeric information

Concepts, Instances, and Attributes

- Concepts: kinds of things that can be learned
- Instances: the individual, independent examples of a concept
- Attributes: measuring aspects of an instance

	age	workclass	education_level	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	native-country	income
0	39	State-gov	Bachelors	13.0	Never-married	Adm-clerical	Not-in-family	White	Male	2174.0	0.0	40.0	United-States	<=50K
1	50	Self-emp-not-inc	Bachelors	13.0	Married-civ-spouse	Exec-managerial	Husband	White	Male	0.0	0.0	13.0	United-States	<=50K
2	38	Private	HS-grad	9.0	Divorced	Handlers-cleaners	Not-in-family	White	Male	0.0	0.0	40.0	United-States	<=50K
3	53	Private	11th	7.0	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0.0	0.0	40.0	United-States	<=50K
4	28	Private	Bachelors	13.0	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0.0	0.0	40.0	Cuba	<=50K

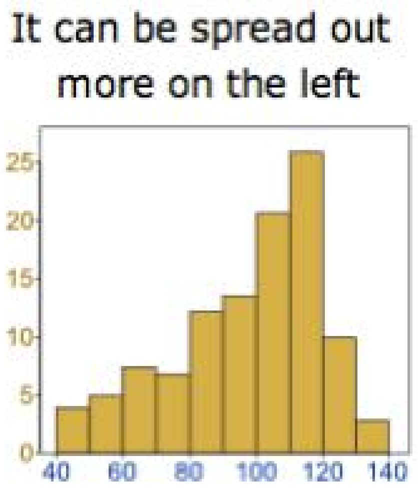
Loading Data

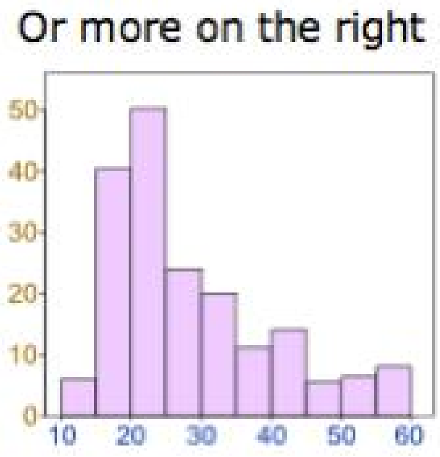
```
[1]: # Import libraries necessary for this project
import numpy as np
import pandas as pd
from time import time
from IPython.display import display # Allows the use of display() for DataFrames

# Import supplementary visualization code visuals.py
import visuals as vs

# Pretty display for notebooks
%matplotlib inline

# Load the Census dataset
data = pd.read_csv("census.csv")

# Success - Display the first record
display(data.head(n=1))
```

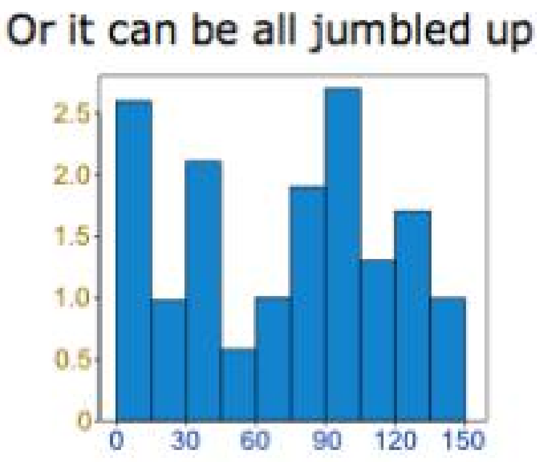
	age	workclass	education_level	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	native-country	income
0	39	State-gov	Bachelors	13.0	Never-married	Adm-clerical	Not-in-family	White	Male	2174.0	0.0	40.0	United-States	<=50K

Distribution

Data can be distributed in different ways

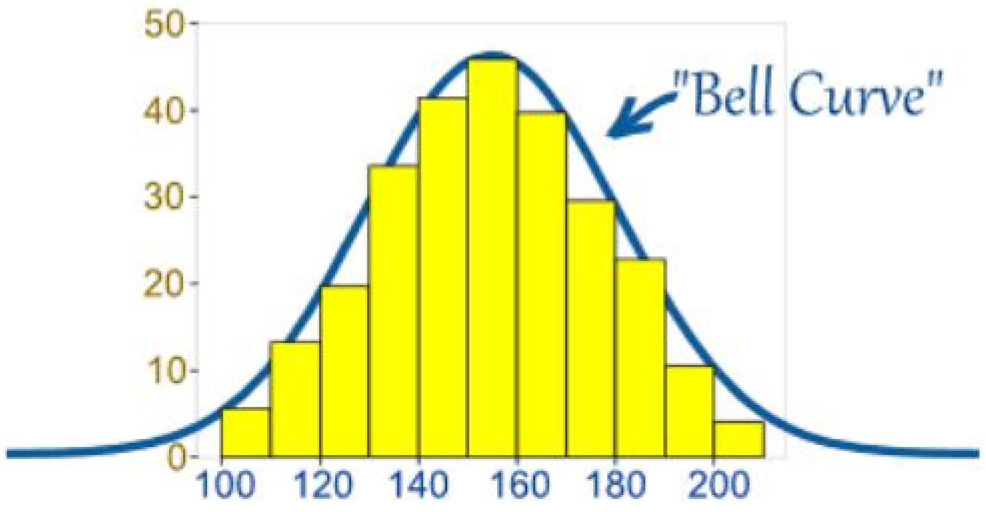






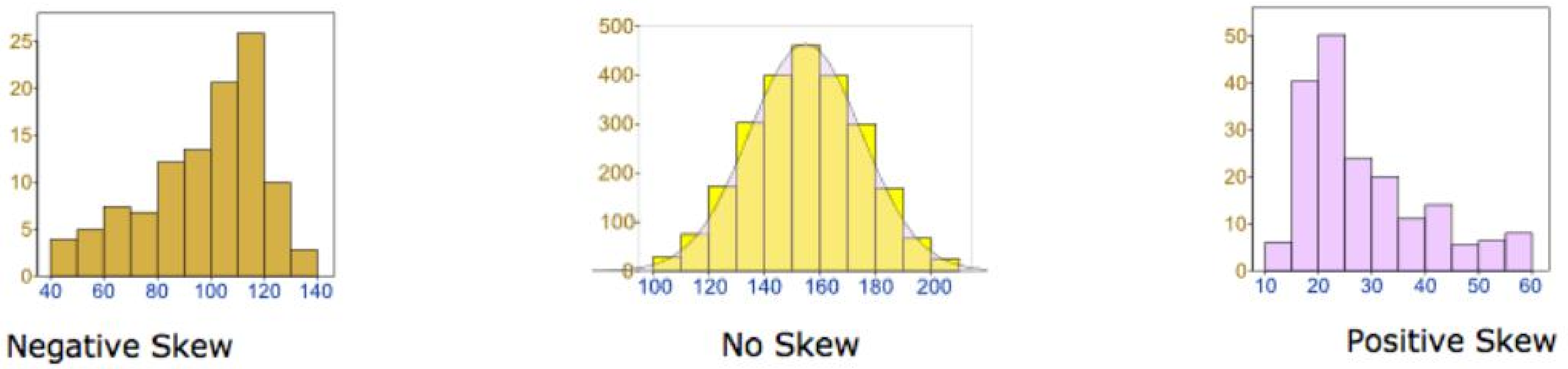
Normal Distribution

A normal distribution is an arrangement of a data set in which most values cluster in the middle of the range and the rest taper off symmetrically toward either extreme.



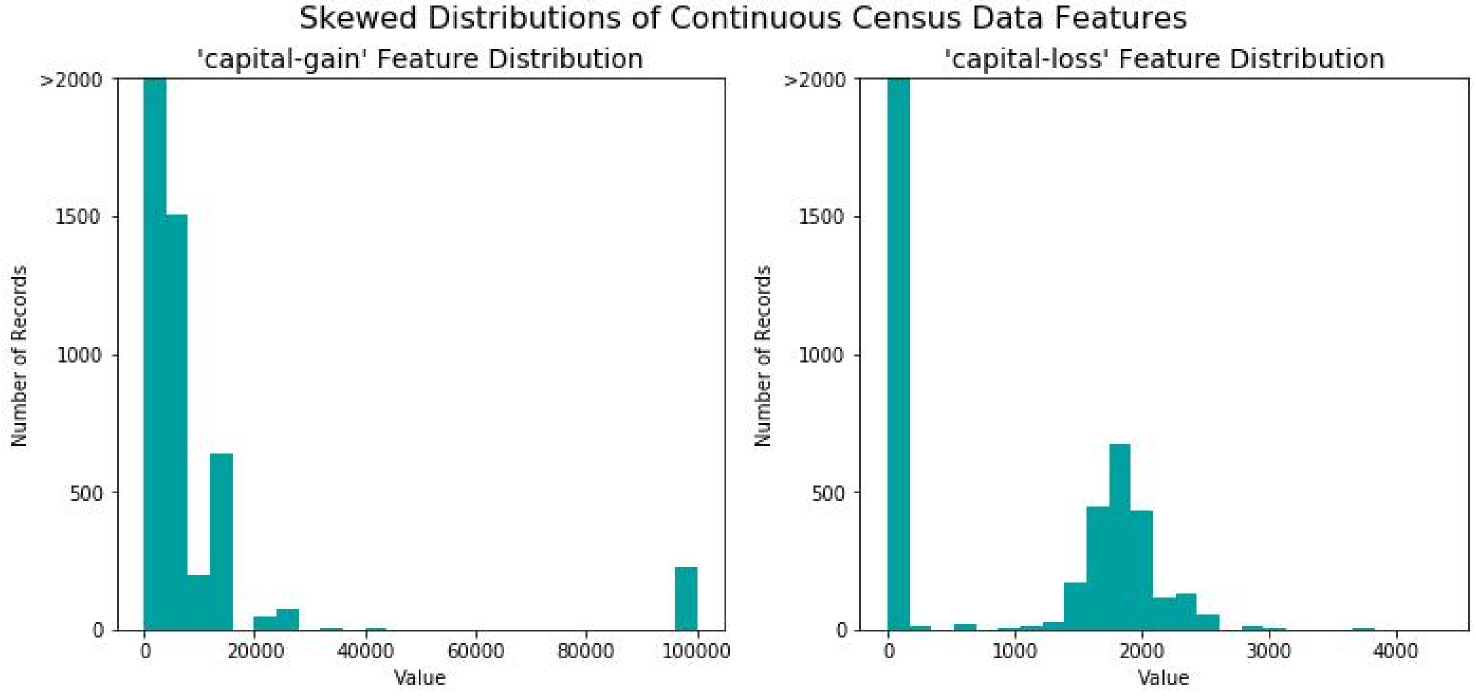
Skewness

When data is “skewed”, it shows long tail on one side or the other:



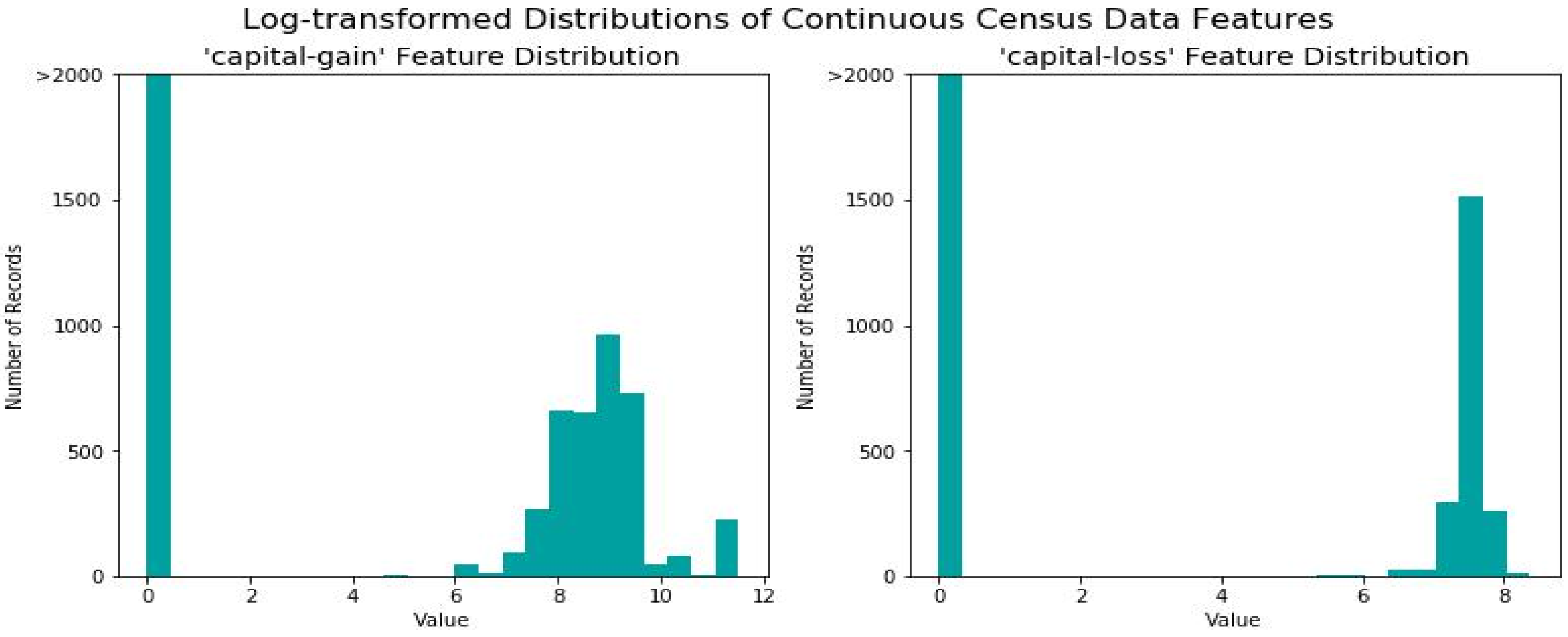
Skewed Distributions

```
# Split the data into features and target label
income_raw = data['income']
features_raw = data.drop('income', axis = 1)

# Visualize skewed continuous features of original data
vs.distribution(data)
```



Log-Transformed Distributions

8]:
```
# Log-transform the skewed features
skewed = ['capital-gain', 'capital-loss']
features_log_transformed = pd.DataFrame(data = features_raw)
features_log_transformed[skewed] = features_raw[skewed].apply(lambda x: np.log(x + 1))

# Visualize the new log distributions
vs.distribution(features_log_transformed, transformed = True)
```



Normalization

To normalize data means to fit the data within unity, so all the data will take on a value between 0 and 1.

Ex: $$X_{i,\,0\text{ to }1} = \frac{X_i - X_{Min}}{X_{Max} - X_{Min}}$$

Look at column “age” “education-num”

	age	workclass	education_level	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	native-country
0	0.301370	State-gov	Bachelors	0.800000	Never-married	Adm-clerical	Not-in-family	White	Male	0.667492	0.0	0.397959	United-States
1	0.452055	Self-emp-not-inc	Bachelors	0.800000	Married-civ-spouse	Exec-managerial	Husband	White	Male	0.000000	0.0	0.122449	United-States
2	0.287671	Private	HS-grad	0.533333	Divorced	Handlers-cleaners	Not-in-family	White	Male	0.000000	0.0	0.397959	United-States
3	0.493151	Private	11th	0.400000	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0.000000	0.0	0.397959	United-States
4	0.150685	Private	Bachelors	0.800000	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0.000000	0.0	0.397959	Cuba

Normalization

```
# Import sklearn.preprocessing.StandardScaler
from sklearn.preprocessing import MinMaxScaler

# Initialize a scaler, then apply it to the features
scaler = MinMaxScaler() # default=(0, 1)
numerical = ['age', 'education-num', 'capital-gain', 'capital-loss', 'hours-per-week']

features_log_minmax_transform = pd.DataFrame(data = features_log_transformed)
features_log_minmax_transform[numerical] = scaler.fit_transform(features_log_transformed[numerical])

# Show an example of a record with scaling applied
display(features_log_minmax_transform.head(n = 5))
```

	age	workclass	education_level	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	native-country
0	0.301370	State-gov	Bachelors	0.800000	Never-married	Adm-clerical	Not-in-family	White	Male	0.667492	0.0	0.397959	United-States
1	0.452055	Self-emp-not-inc	Bachelors	0.800000	Married-civ-spouse	Exec-managerial	Husband	White	Male	0.000000	0.0	0.122449	United-States
2	0.287671	Private	HS-grad	0.533333	Divorced	Handlers-cleaners	Not-in-family	White	Male	0.000000	0.0	0.397959	United-States
3	0.493151	Private	11th	0.400000	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0.000000	0.0	0.397959	United-States
4	0.150685	Private	Bachelors	0.800000	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0.000000	0.0	0.397959	Cuba

Feature Representation

Binary data is a special type of categorical data. Binary data takes only two values.

pandas.get_dummies(data, prefix=None, prefix_sep='_', dummy_na=False, columns=None, sparse=False, drop_first=False, dtype=None)[source]
Convert categorical variable into dummy/indicator variables

workclass_ Local-gov	workclass_ Private	workclass_ Self-emp-inc	workclass_ Self-emp-not-inc	...	native-country_ Portugal	native-country_ Puerto-Rico	native-country_ Scotland	native-country_ South	native-country_ Taiwan	native-country_ Thailand	native-country_ Trinadad&Tobago	native-country_ United-States	native-country_ Vietnam	native-country_ Yugoslavia
0	0	0	0	...	0	0	0	0	0	0	0	1	0	0
0	0	0	1	...	0	0	0	0	0	0	0	1	0	0
0	1	0	0	...	0	0	0	0	0	0	0	1	0	0
0	1	0	0	...	0	0	0	0	0	0	0	1	0	0
0	1	0	0	...	0	0	0	0	0	0	0	0	0	0

Feature selection

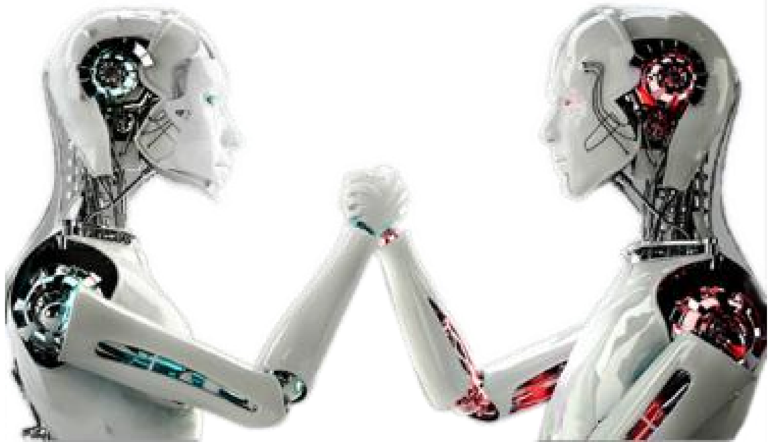
- Feature Selection
  - ✓ Achieves the reduction of the data set by removing irrelevant or redundant features (or dimensions).

  https://scikit-learn.org/stable/modules/classes.html#module-sklearn.feature_selection

- Instance Selection
  - ✓ Consists of choosing a subset of the total available data to achieve the original purpose of the DM application as if the whole data had been used.

PART TWO

Evaluating Machine Learning Algorithms



Evaluation

- Is accuracy an adequate measure of predictive performance?
- accuracy may not be useful measure in cases where there is a large class skew
  - ✓ Is 98% accuracy good if 97% of the instances are negative?
- there are differential misclassification costs – say,getting a positive wrong costs more than getting a negative wrong
  - ✓ Consider a medical domain in which a false positive results in an extraneous test but a false negative results in a failure to treat a disease
- we are most interested in a subset of high-confidence predictions

Evaluation-accuracy metrics

		actual class: positive	actual class: negative
predicted class	positive	true positives (TP)	false positives (FP)
	negative	false negatives (FN)	true negatives (TN)

$$\text{recall (TP rate)} = \frac{\text{TP}}{\text{actual pos}} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{precision} = \frac{\text{TP}}{\text{predicted pos}} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Evaluation-accuracy metrics

		actual class: positive	actual class: negative
predicted class	positive	true positives (TP)	false positives (FP)
	negative	false negatives (FN)	true negatives (TN)

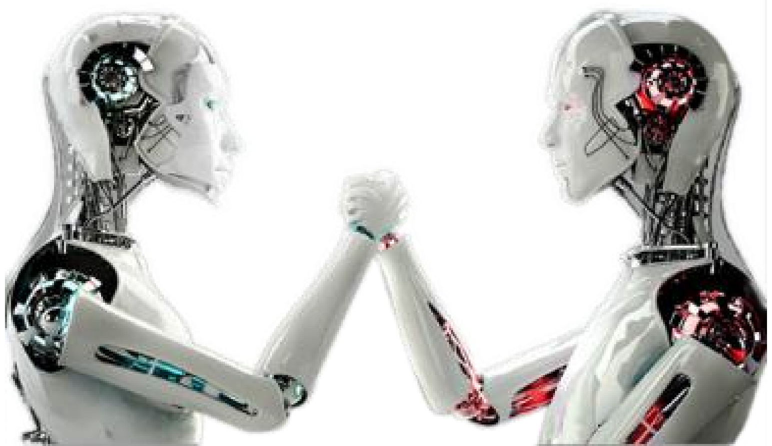
$$\text{recall (TP rate)} = \frac{\text{TP}}{\text{actual pos}} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{precision} = \frac{\text{TP}}{\text{predicted pos}} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\mathrm{F}_{\beta} = (1 + \beta^{2}) \bullet \frac{precision \bullet recall}{(\beta^{2} \bullet precsion) + recall}$$

PART THREE

Lab Task



Lab Task

1. Complete the exercises and questions in the lab02_preliminary.pdf
2. Submit your result file with an extension ".ipynb" to BB.

Thanks

贾艳红 Jana
Email:jiayh@mail.sustech.edu.cn