Lecture Notes for
Machine Learning in Python

Professor Eric Larson

Introduction, Syllabus, Data Types

Class Logistics and Agenda

- Agenda:
  - Introductions
  - Syllabus
  - Overview of Machine Learning
  - Types of Data and Representation

- My approach to this course:
  - Programming
  - Math
  - **Applications** and **Analytics**

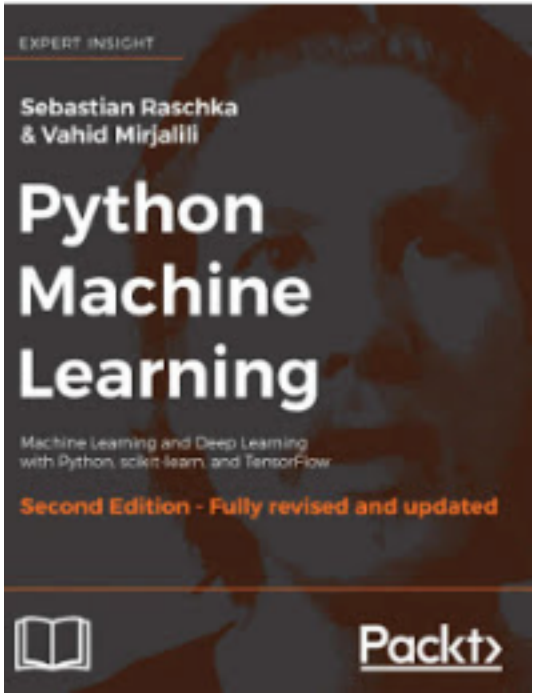
Introductions
& Course Syllabus

Introductions

- Me
    - Eric Larson
- You
    - Name, department, grad/ugrad
    - Something true or false

FAQ

- Text:
  - **Recommended**: Python Machine Learning, Raschka & Mirjalili, Second edition



- Use Canvas for posted course material
- Prerequisite:
  - Linear Algebra, Calculus
  - Basic statistics and probability
  - Python programming
- Version of python: 3.X
  - Install through Anaconda
  - Use conda environments
  - JupyterLab (notebooks)
- Most Used Libraries: Numpy, Pandas, Scikit-Learn, Matplotlib, Keras with Tensorflow

How will you grade participation?

- Participation will be graded in the course:
  - Distance students will answer these questions via canvas upload
  - must upload the questions throughout semester for full credit

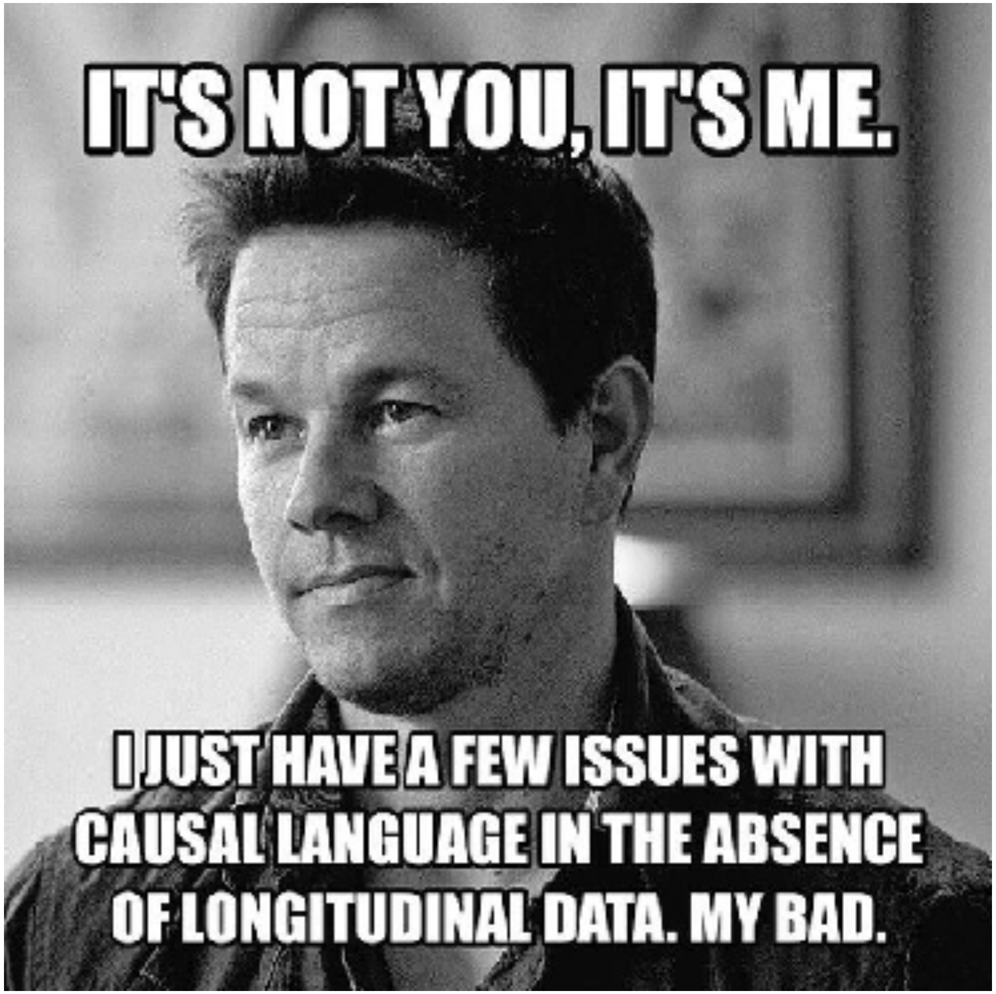
- Choose to respond to the question:
  - Do you think this will work?
    - A: **Yes** this is going to work
    - B: This is **not** going to work
    - C: I am **not even here**

Question ML1a

Canvas Syllabus

- Assignments
- Grading Rubrics
- Participation
- Course Schedule
- In-Class Assignments
- Difference between 5000 and 7000

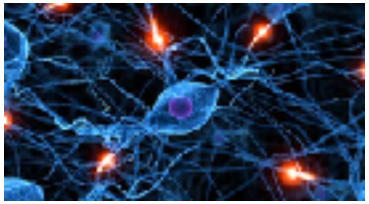
Machine Learning Overview



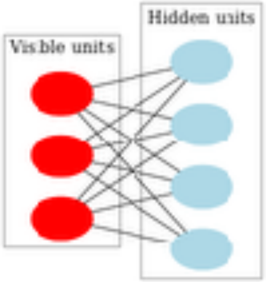
A History of Machine Learning

- Historically builds from disciplines statistics and computer science (algorithms)
- Its really just algorithms for optimizing weights 🤠

- **1952**: Arthur Samuel IBM creates checker program



- **1957**: Rosenblatt, Neural Network Perceptron
- **1967**: Nearest Neighbor Pattern Recognition
- **1970's**: AI Winter



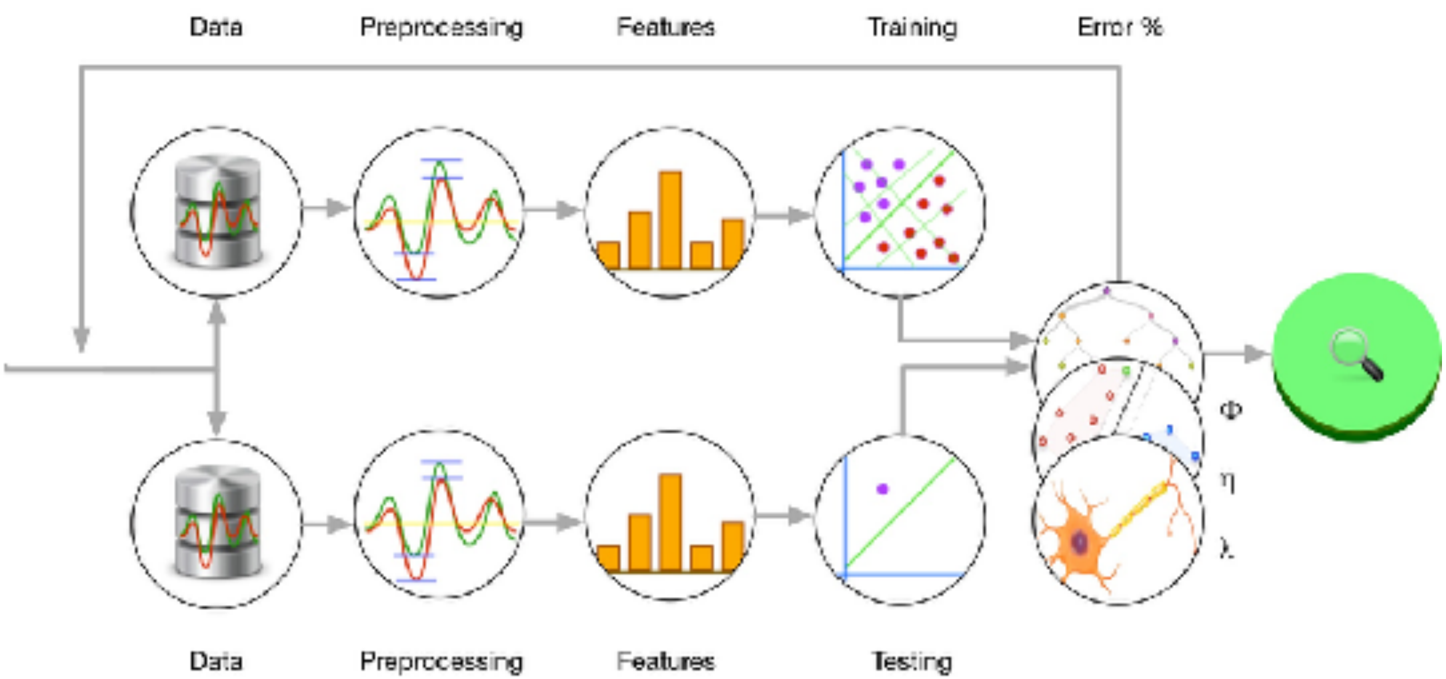
- **1990's**: Volley of "New" Machine learning Algorithms
- **2001**: Breiman's Random Forests
- ~**2004**: Modern Support Vector Machines with Kernels
- **2005**: Second AI Winter
- ~**2010**: Deep Learning Convolutional Networks
- **2015**: Deep Learning becomes buzz word, you hear about it and take this course

What is Machine Learning?

Machine learning is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. **Machine learning** focuses on the development of computer programs that can change when exposed to new data.

What is machine learning? - Definition from WhatIs.com
whatis.techtarget.com/**definition/machine-learning**

About this result • Feedback



Machine Learning is part of Data Mining

One Small Piece of Artificial Intelligence

Data Mining

ML

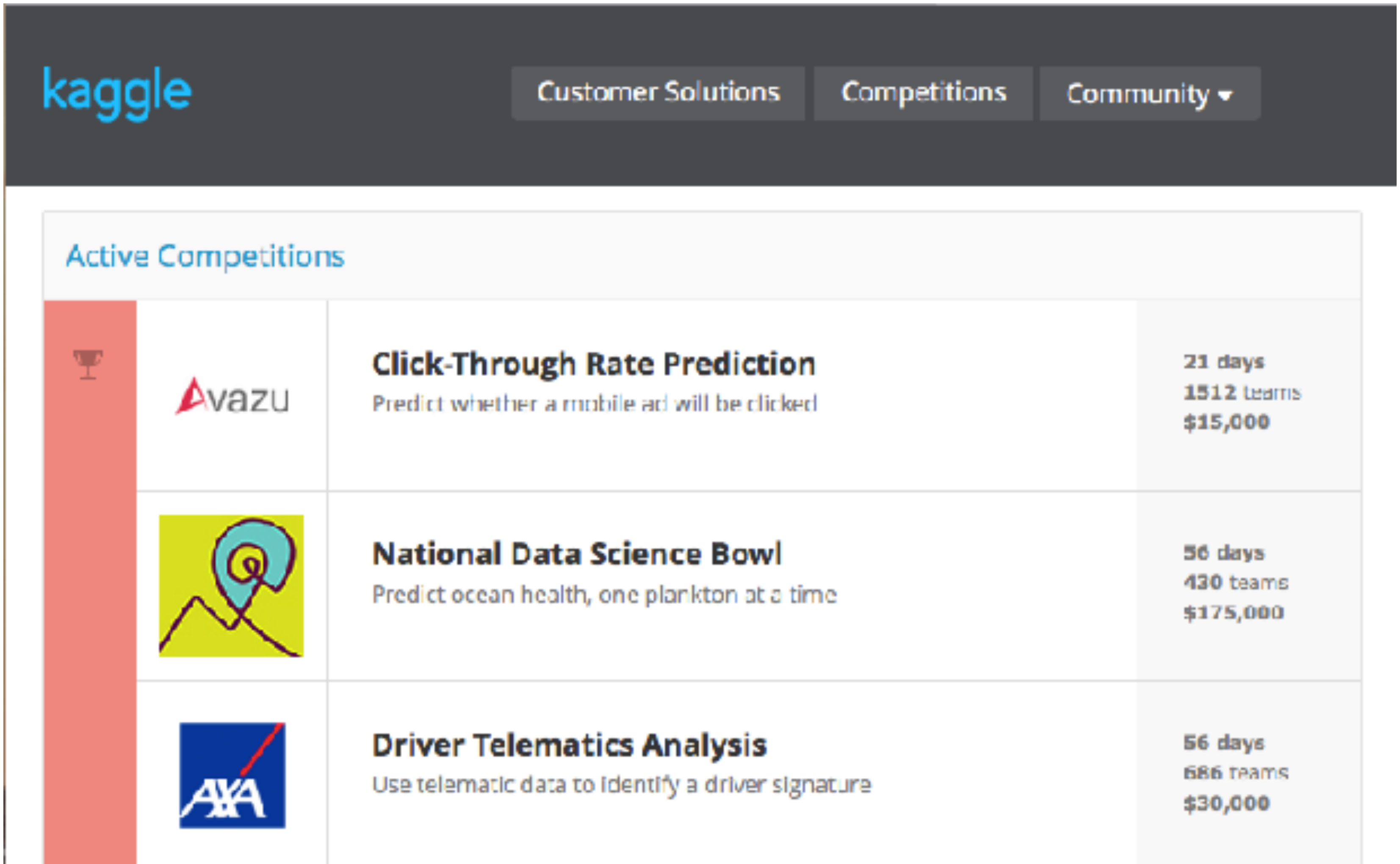
- Prediction Methods
  - Use some variables to predict unknown or future values of other variables
- Description Methods
  - Find human-interpretable patterns that describe the data.

ML

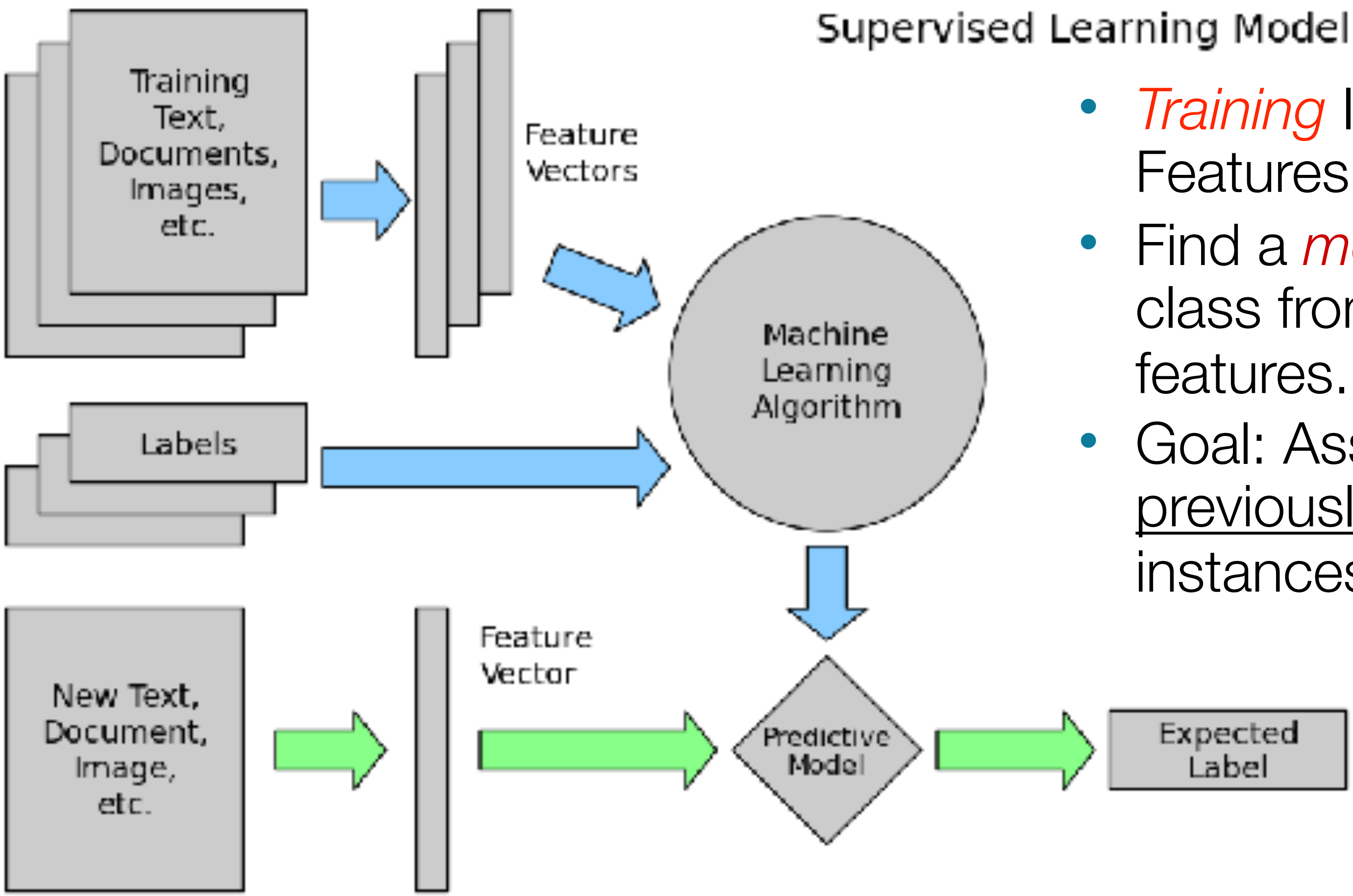
- Classification
- Regression
- Deviation Detection
- Clustering
- Association Rule Discovery
- Sequential Pattern Discovery

section 1, manipulated from Tan et al. Introduction to Data Mining

Problem Types in Machine Learning



Classification: Definition



- *Training* Instances: Features + Labels
- Find a *model* mapping class from values of features.
- Goal: Assign class to <u>previously unseen</u> instances

section 1, from Tan et al. Introduction to Data Mining and Scikit Learn Documentation

Classification: Definition

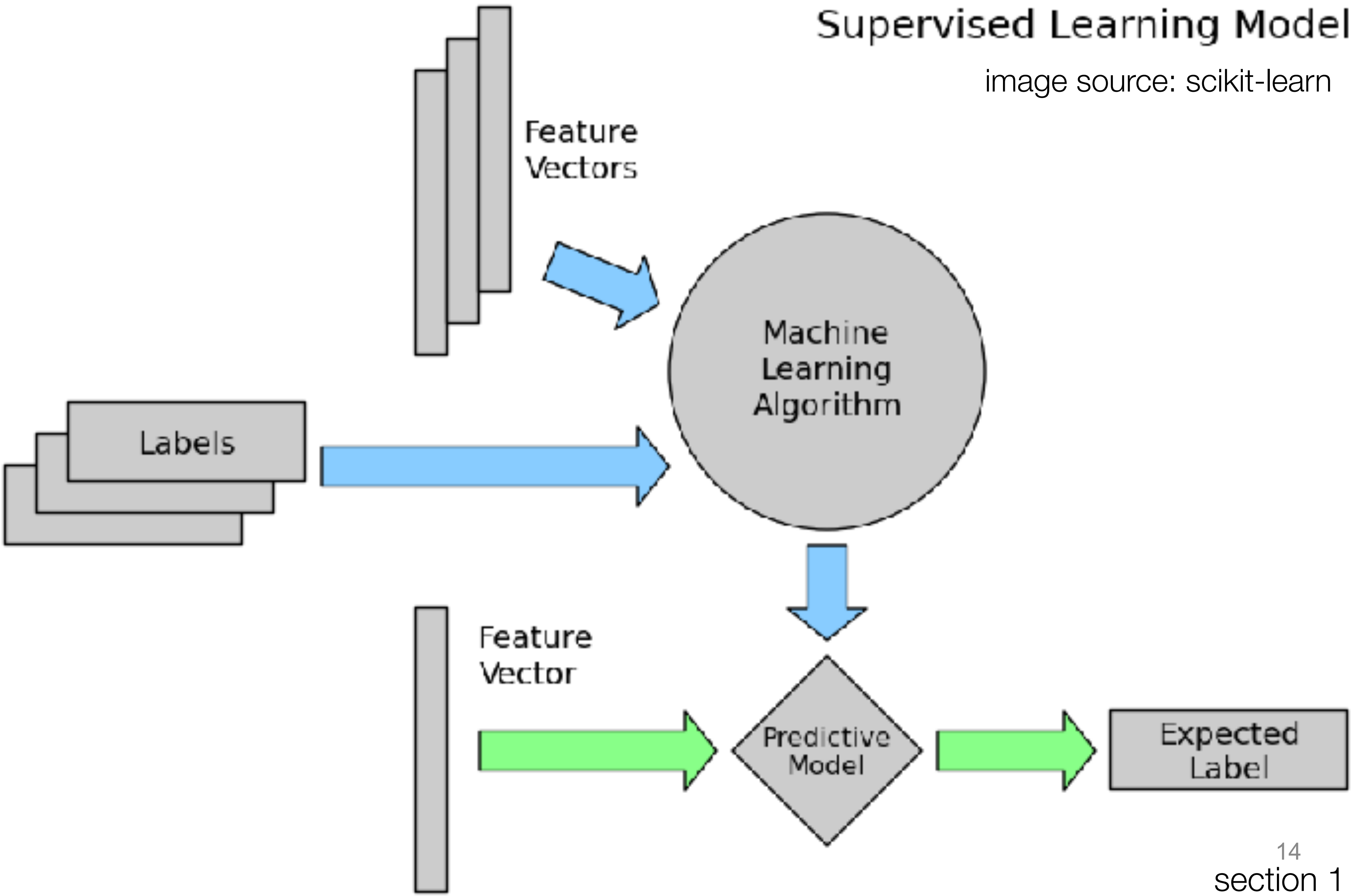


image source: scikit-learn

Classification: Malware

- Classify files as malware based on structure, size, and naming.
- Approach:
  - Use already classified malware files
  - Must translate name to set of features
  - ***{malware, not malware}*** decision forms the ***class attribute***
  - Collect various malware examples and a number of safe files, providing labels for each and a set of features

Training Set

TID	Name	Size	Class
1	erte.dll	916 b	not
2	fufu.bin	1M	yes
3	exe.exe	1G	not
4	ex.py	113 b	not

Unknown

TID	***Name***	***Size***
1	asdf.dll	11b

Classifying: Objects in Images

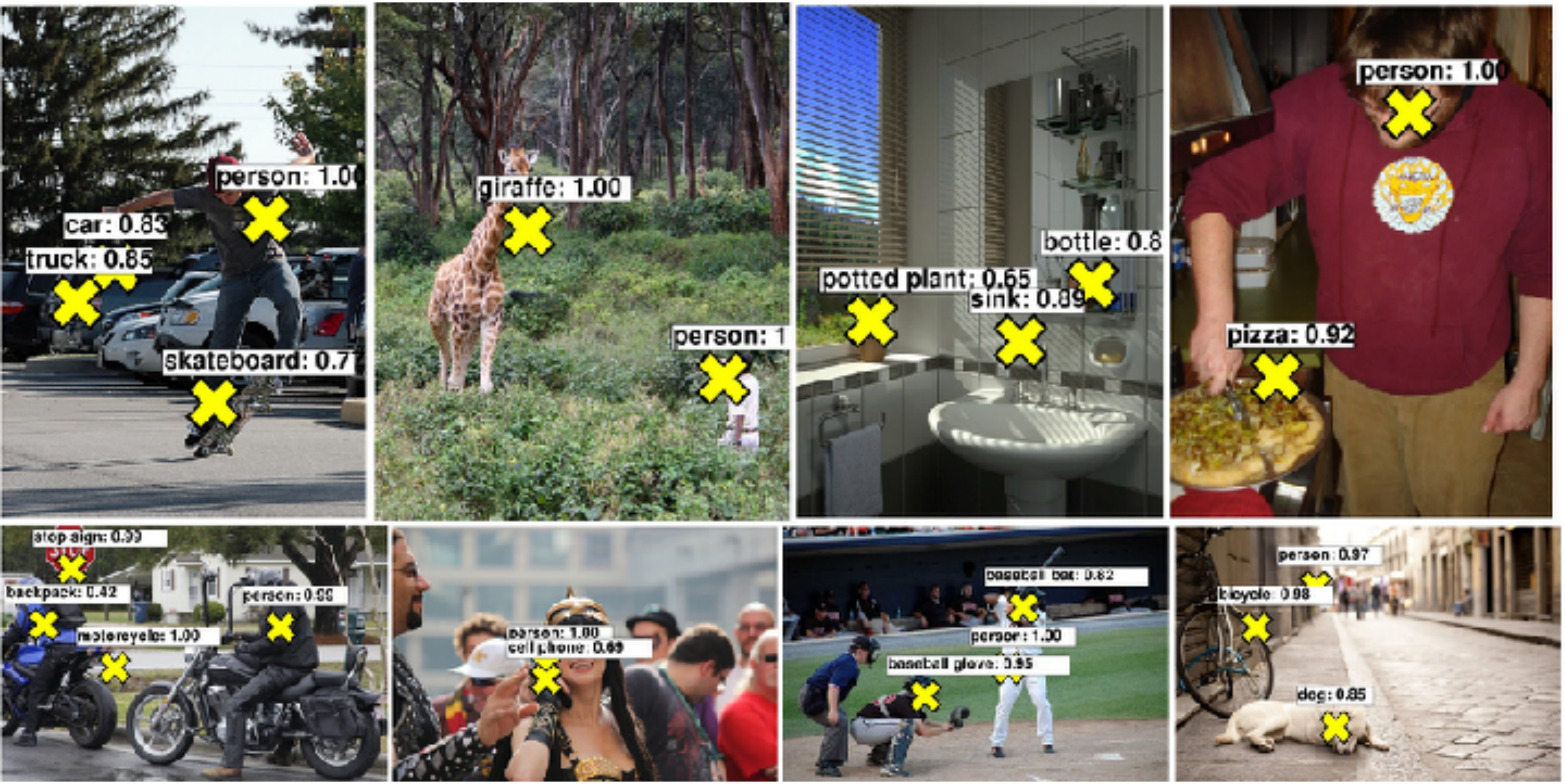


Image Net:
- 14 million images
- 200 Labeled Categories
- 1000 Location Labels

Attributes:
- Images

Regression

- Predict a value of a given *continuous valued* variable based on the values of other variables
- Examples:
  - Predicting sales amounts of new product based on advertising expenditure.
  - Predicting wind velocities as a function of temperature, humidity, air pressure, etc.
  - Predicting lung function as a function of gender, weight, height

Training Set

TI	Gend.	Weight	Asthma	LF
1	M	175lbs	N	85%
2	F	150lbs	N	87.3%
3	F	155lbs	Y	90%
4	M	225lbs	Y	65.2%

Unknown

TI	Gend.	Weight	Asthma
1	M	160lbs	N

Self Test

- **A. classification**
  B. regression
  C. not Machine Learning
    - Dividing up customers by potential profitability?
    - Extracting frequency of sound?

Types of Data and Categorization

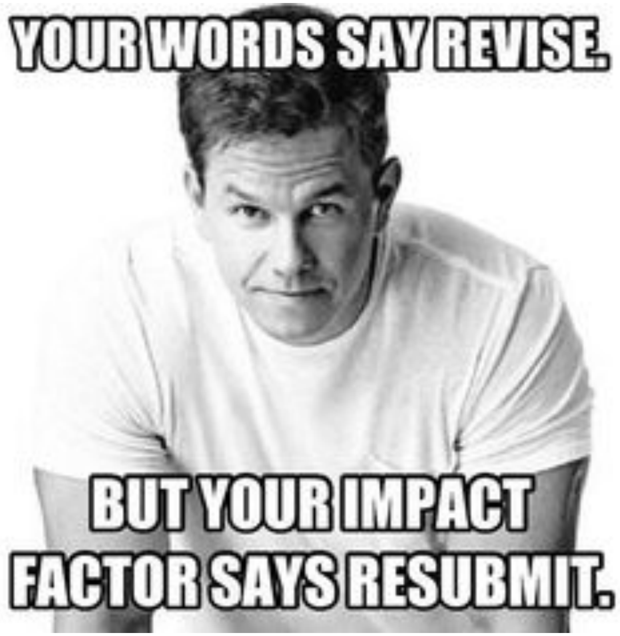


Table Data

- Collection of data **instances** and their **features**

- A **feature** is a property or characteristic of an object
  - Examples: eye color of a person, temperature, etc.

- A collection of features describe an **instance**

Attributes, variables, fields, characteristics, **Features**

Objects, records, points, samples, cases, entities, **Instances**

TID	*Pregnant*	*BMI*	*Age*	*Diabetes*
1	Y	33.6	41-50	positive
2	N	26.6	31-40	negative
3	Y	23.3	31-40	positive
4	N	28.1	21-30	negative
5	N	43.1	31-40	positive
6	Y	25.6	21-30	negative
7	Y	31.0	21-30	positive
8	Y	35.3	21-30	negative
9	N	30.5	51-60	positive
10	Y	37.6	51-60	positive

section 2, from Tan et al. Introduction to Data Mining

Types of Attributes

Nominal attribute: distinctness

- *e.g.*, ID numbers, eye color, zip codes
  - *Distinctness*: $= \neq$

Ordinal attribute: distinctness & order

- *e.g.*, rankings (e.g., taste of potato chips on a scale from 1-10), grades, height in {tall, medium, short}
  - *Order*: $< >$

Interval attribute: distinctness, order, & addition

- *e.g.*, calendar dates, temperatures in Celsius or Fahrenheit.
  - *Addition*: $+ -$

Ratio attribute: all properties

- *e.g.*, temperature in Kelvin, length, time, counts
  - *Multiplication*: $*$ $/$

section 2, from Tan et al. Introduction to Data Mining

Feature Type Representation

	Attribute	Representation Transformation	Comments
Discrete	**Nominal**	Any permutation of values one hot encoding	If all employee ID numbers were reassigned, would it make any difference?
	Ordinal	An order preserving change of values, i.e., new_value = f(old_value) where f is a monotonic function. integer	An attribute encompassing the notion of good, better best can be represented equally well by the values {1, 2, 3} or by { 0.5, 1, 10}.
Continuous	**Interval**	new_value =a * old_value + b where a and b are constants float	Thus, the Fahrenheit and Celsius temperature scales differ in terms of where their zero value is and the size of a unit (degree).
	Ratio	new_value = a * old_value float	Length can be measured in meters or feet.

Self Test

- Are these **A. interval or B. ratio**:
  - Height above sea level
    - interval or ratio depending on if sea level is considered arbitrary

- Are these **A. ordinal, B. nominal, or C. binary**?
  - military rank
    - ordinal
  - coat check number
    - nominal

Before Next Lecture

- Before next class:
  - install python on your laptop
  - install anaconda distribution of python

- Look at Python primer if you need review
  - I made ~4 hours of YouTube content…
  - https://www.youtube.com/playlist?list=PL7IPdRN5E0YKCnVl-fvx8jOOCWVeGTsrV

Demo

If time:
Jupyter Notebooks and Numpy

`01_Numpy and Pandas Intro.ipynb`

Lecture Notes for
Machine Learning in Python

Professor Eric Larson

Numpy, Pandas, Document Features

Class Logistics and Agenda

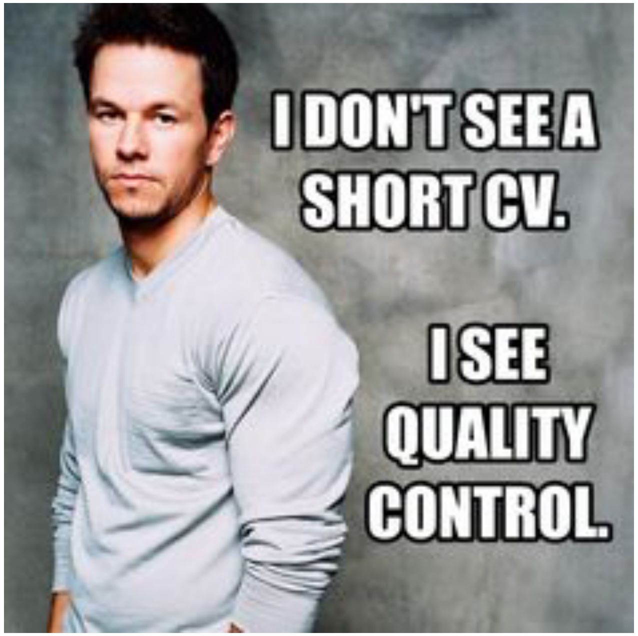
- Canvas? Anaconda Installs?
- Distance transfers?
- Agenda:
  - Numpy
  - Data Quality
  - Attributes Representation
    - documents
  - The Pandas eco-system
    - loading and manipulating attributes

Demo

“Finish” Jupyter Notebooks and Numpy

`01_Numpy and Pandas Intro.ipynb`

Data Quality



Review of Feature Data

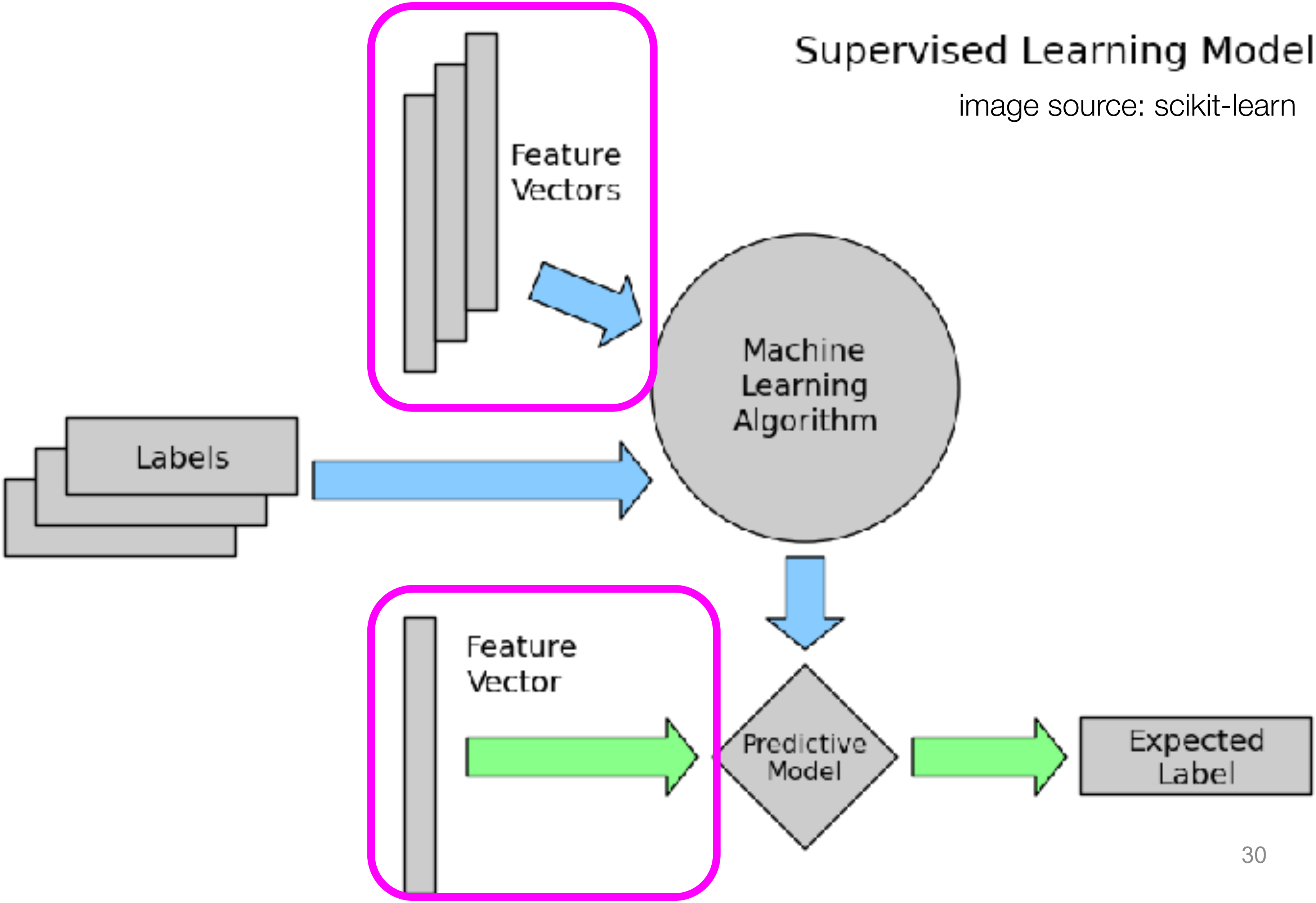


image source: scikit-learn

Data Quality Problems

- Noise and outliers
  - remove if you know its noise/outlier
- Missing values
  - replace or ignore
- Duplicate data
  - clean entries or merge

Missing Values

- Reasons for missing values
  - Information is **not collected**
    (*e.g.*, people decline to give their age and weight)
  - Features may **not be applicable** to all cases
    (*e.g.*, annual income for children)
  - **UCI ML Repository**: 90% of repositories have missing data

- Handling missing values
  - **Eliminate** Data Objects
  - **Impute** Missing Values How?
  - **Ignore** the Missing Value During Analysis
  - Replace with all possible values (talk about later)

Stats?
mean
median
mode

Imputation

- When is it probably fine to impute missing data:
  - (A) When there is not much missing data
  - (B) When the missing feature is mostly predictable from another feature
  - (C) When there is not much missing data for each subgroup of the data
  - (D) When it is the class you want to predict

self test ML1b.1

Split-Impute-Combine

TID	Pregnant	BMI	Age	Diabetes
1	Y	33.6	41-50	positive
2	N	26.6	31-40	negative
3	Y	23.3	?	positive
4	N	28.1	21-30	negative
5	N	43.1	31-40	positive
6	Y	25.6	21-30	negative
7	Y	31.0	21-30	positive
8	Y	35.3	?	negative
9	N	30.5	51-60	positive
10	Y	37.6	51-60	positive

split: pregnant
split: BMI > 32

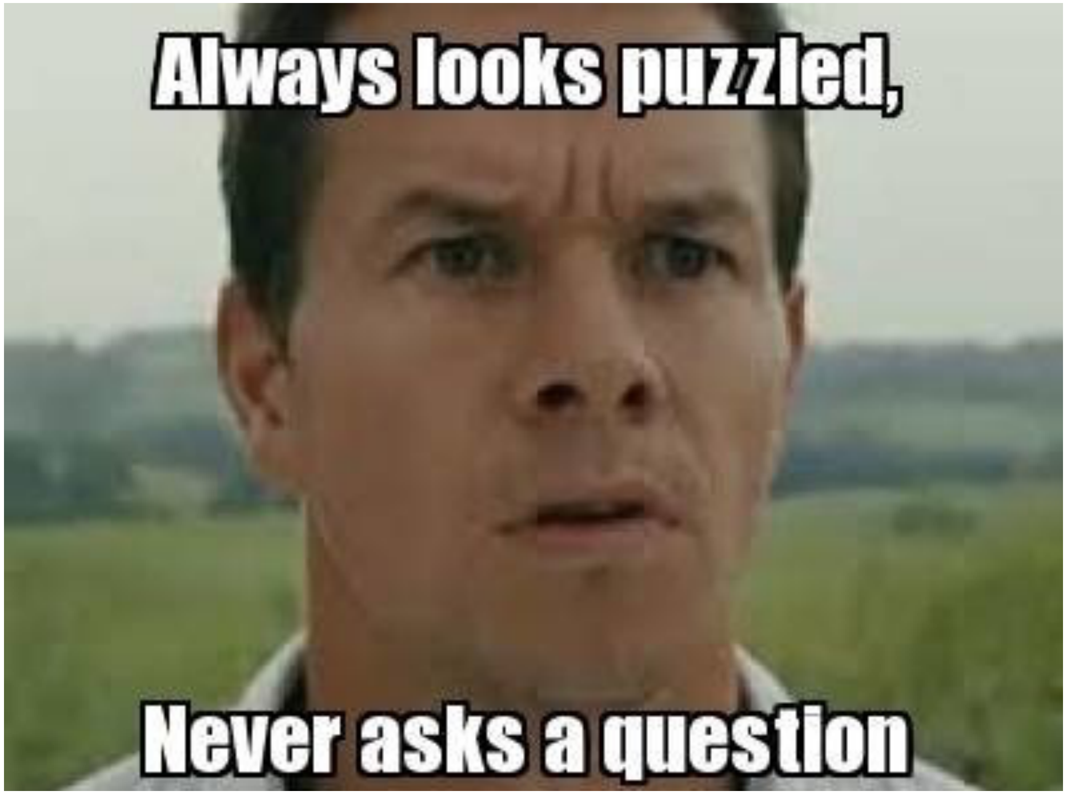
TID	Pregnant	BMI	Age	Diabetes
1	Y	>32	41-50	positive
8	Y	>32	?	negative
10	Y	>32	51-60	positive

Mode: none, can't impute

TID	Pregnant	BMI	Age	Diabetes
3	Y	<32	?	positive
6	Y	<32	21-30	negative
7	Y	<32	21-30	positive

Mode: 21-30

Data Representation



Feature Type Representation

	Attribute	Representation Transformation	Comments
Discrete	**Nominal**	Any permutation of values **one hot encoding**	If all employee ID numbers were reassigned, would it make any difference?
	Ordinal	An order preserving change of values, i.e., new_value = f(old_value) where f is a monotonic function. **integer**	An attribute encompassing the notion of good, better best can be represented equally well by the values {1, 2, 3} or by { 0.5, 1, 10}.
Continuous	**Interval**	new_value =a * old_value + b where a and b are constants **float**	Thus, the Fahrenheit and Celsius temperature scales differ in terms of where their zero value is and the size of a unit (degree).
	Ratio	new_value = a * old_value **float**	Length can be measured in meters or feet.

Data Tables as Variable Representations

Table

TID	Pregnant	BMI	Age	Eye Color	Diabetes
1	Y	33.6	41-50	brown	positive
2	N	26.6	31-40	hazel	negative
3	Y	23.3	31-40	blue	positive
4	N	28.1	21-30	brown	inconclusive
5	N	43.1	31-40	blue	positive
6	Y	25.6	21-30	hazel	negative

Internal Rep.

TID
1
2
3
4
5
6

Data Tables as Variable Representations

Table

TID	Pregnant	BMI	Age	Eye Color	Diabetes
1	Y	33.6	41-50	brown	positive
2	N	26.6	31-40	hazel	negative
3	Y	23.3	31-40	blue	positive
4	N	28.1	21-30	brown	inconclusive
5	N	43.1	31-40	blue	positive
6	Y	25.6	21-30	hazel	negative

Internal Rep.

TID	Binary	Float	Ordinal	Object	Diabetes
1	1	33.6	2	hash(0)	1
2	0	26.6	1	hash(1)	0
3	1	23.3	1	hash(2)	1
4	0	28.1	0	hash(0)	2
5	0	43.1	1	hash(2)	1
6	1	25.6	0	hash(1)	0

Bag of words model

TID	Pregnant	BMI	Chart Notes	Diabetes
1	Y	33.6	Complaints of fatigue wh…	positive
2	N	26.6	Sleeplessness and some…	negative
3	Y	23.3	First saw signs of rash o…	positive
4	N	28.1	Came in to see Dr. Steve…	inconclusive
5	N	43.1	First diagnosis for hospit…	positive
6	Y	25.6	N/A	negative

Vocabulary

Bag of Words

TID	Sleep	Fatigue	Weight	Rash	First	Sight
1	0	1	0	0	2	0
2	1	1	0	0	1	1
3	1	1	0	2	1	1

number of occurrences

Feature Hashing

what happens when we get more words?

TID	Slee	Fati	Wei	Ras	First	Sigh	Why	Fox	Bro	Lazy	Dog	Etc	Stev
1	0	1	0	0	2	0	0	0	0	1	0	2	0
2	1	1	0	0	1	1	0	0	4	0	1	3	0
3	1	1	0	2	1	1	1	0	1	0	0	1	0

or we could have a hashing function, $h(x) = y$

TID	h(x)=1	h(x)=2	h(x)=3	h(x)=4	h(x)=5	h(x)=6
1	0	1	0	1	2	0
2	1	1	4	0	2	1
3	2	1	1	2	1	1

multiple words mapped to one feature

(want to minimize collisions or make collisions meaningful)

Term-Frequency, Inverse-Document-Frequency

TID	Slee	Fati	Wei	Ras	First	Sigh	Why	Fox	Bro	Lazy	Dog	Etc	Stev
1	0	0.05	0	0	0.34	0	0	0	0	1	0	0.86	0
2	0.1	0.05	0	0	0.12	0.25	0	0	1.21	0	1	1.02	0
3	0.1	0.05	0	0.27	0.12	0.25	0.02	0	0.45	0	0	0.1	0

term frequency $\mathrm{tf}(t, d) = f_{td}, \; t \in T \text{ and } d \in D$

"num occurrences of t in doc d"/"words in d"

inverse document frequency: normalize occurrences

$$\mathrm{idf}(t, d) = \log \frac{|D|}{|n_t|}, \text{ where } n_t = d \in D \text{ with } t \in d$$

"total docs"/"num docs with t"

$$\text{tf-idf}(t, d) = \mathrm{tf}(t, d) \cdot \mathrm{idf}(t, d)$$

$$\text{tf-idf}(t, d) = \mathrm{tf}(t, d) \cdot (1 + \mathrm{idf}(t, d))$$ smoothed

TF-IDF

- The tf-idf value can never be greater than one.
  - (A) true
  - (B) false
  - (C) it depends on IDF normalization

term frequency $\mathrm{tf}(t,d) = f_{td},\ t \in T \text{ and } d \in D$

"num occurrences of t in doc d"/"words in d"

inverse document frequency: normalize occurrences

$$\mathrm{idf}(t,d) = \log\frac{|D|}{|n_t|},\ \text{where } n_t = d \in D \text{ with } t \in d$$

"total docs"/"num docs with t"

$$\text{tf-idf}(t,d) = \mathrm{tf}(t,d) \cdot \mathrm{idf}(t,d)$$

self test ML1b.2

Demo

Sklearn and Pandas

TF-IDF

DataFrames

Loading

Indexing

Imputing

`02_Document Feature Engineering.ipynb`

Other Tutorials:

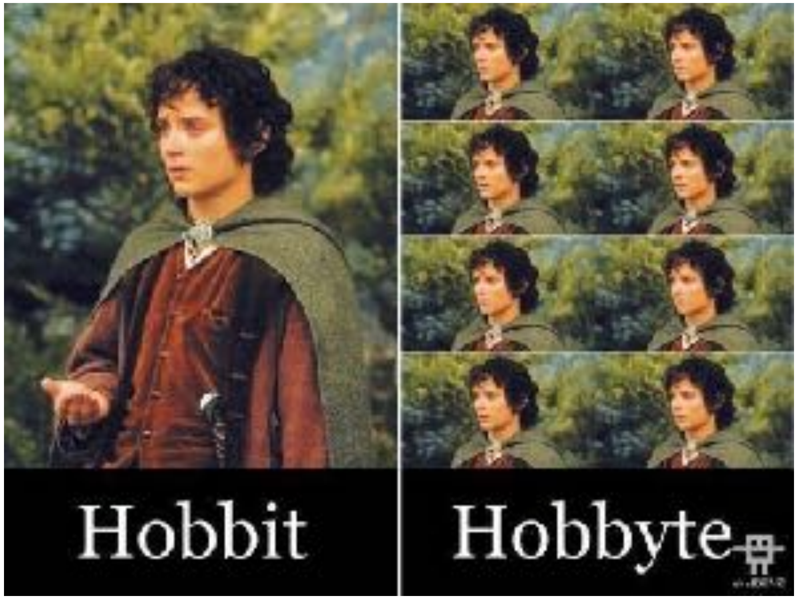
http://vimeo.com/59324550

http://pandas.pydata.org/pandas-docs/version/0.15.2/tutorials.html

For Next Lecture

- Before next class:
  - install seaborn
  - install plotly
  - look at pandas table data and look at additional tutorials

- Next Week: Data Visualization
- End of Next Week:
  - **Lab One Due, Table or Text Data**


Hobbit
Hobbyte


Dear Valued Customers:
Chicken Wings
& Cheesy Crust
Are Currently Out of Stock
Due to a Recent Cyberattack
Which Has Affected Imports
We Apologize
For The Inconvenience