CS 7324/5324

Machine Learning in Python

Professor Ginger Alford

Introduction, Syllabus, Data Types

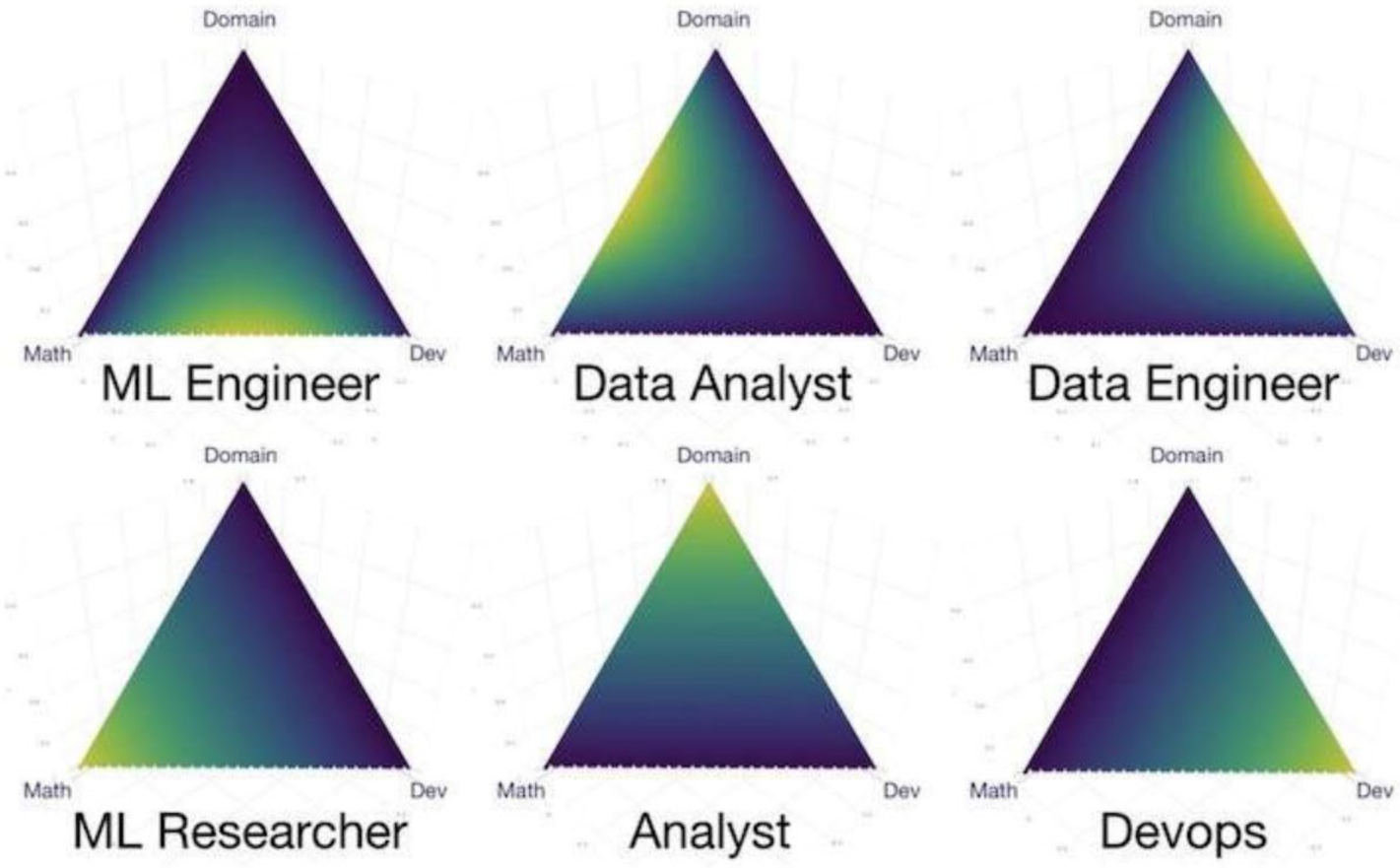
Agenda

- Agenda:
  - Machine Learning Overview
    Who, what, why, when and how
  - Course Approach and Introductions
  - Syllabus and Course Overview

  - Getting Started

- **To Do** for Next Time
  - Install support for Python 3 and Jupyter notebooks
  - Read Chapter 1 of recommended text (PML)

Machine Learning: Who?

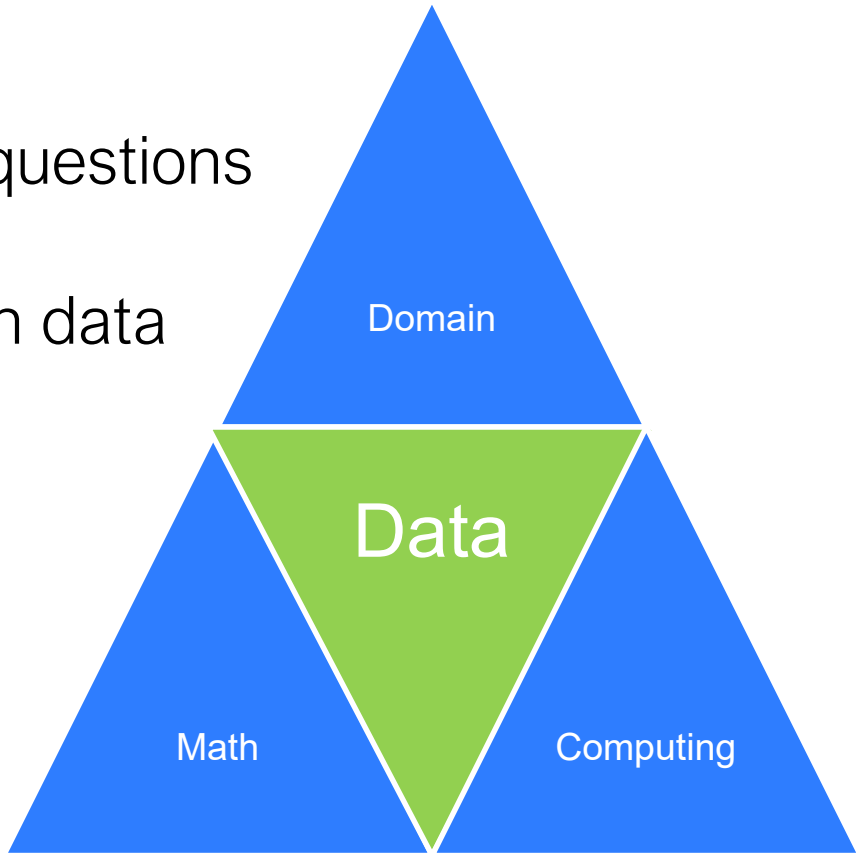


Machine Learning: What?

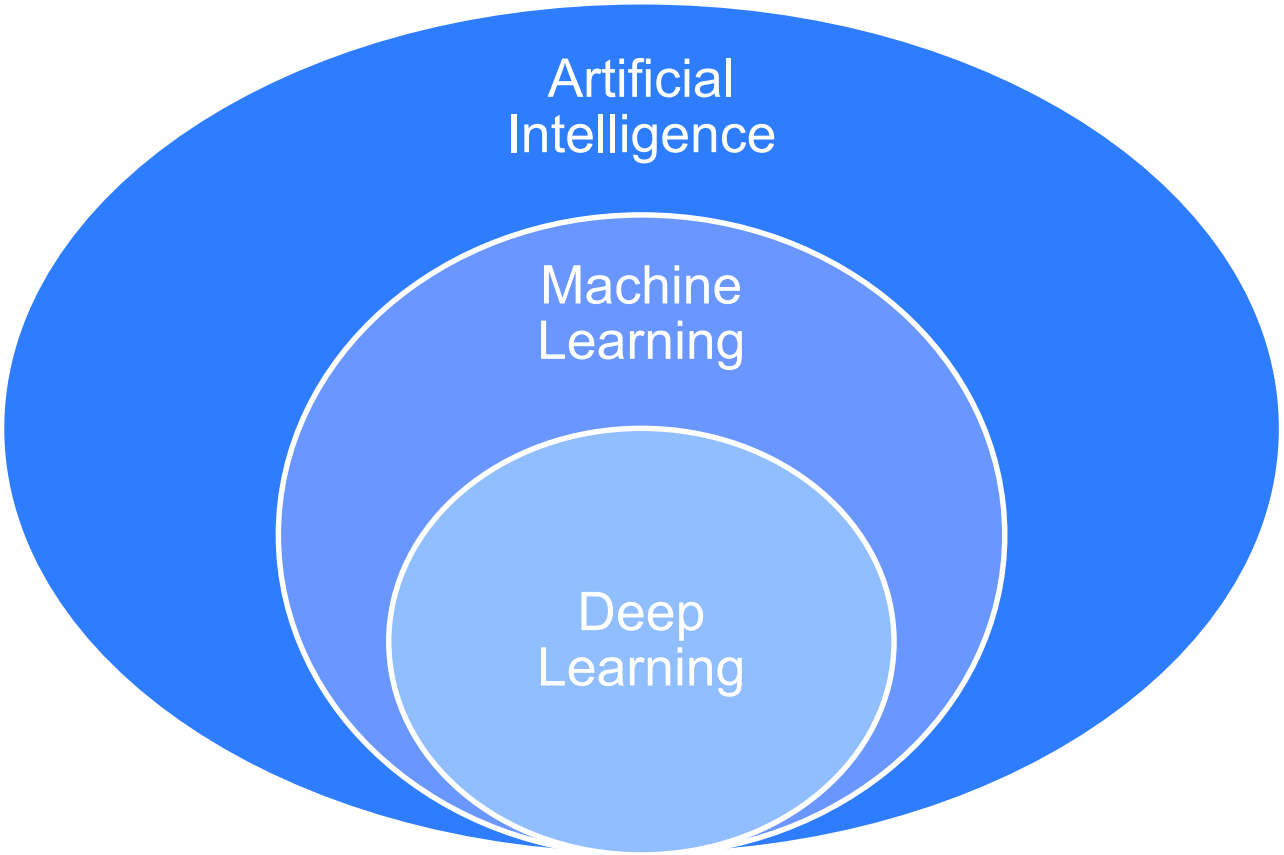
Address domain questions

Finding patterns in data

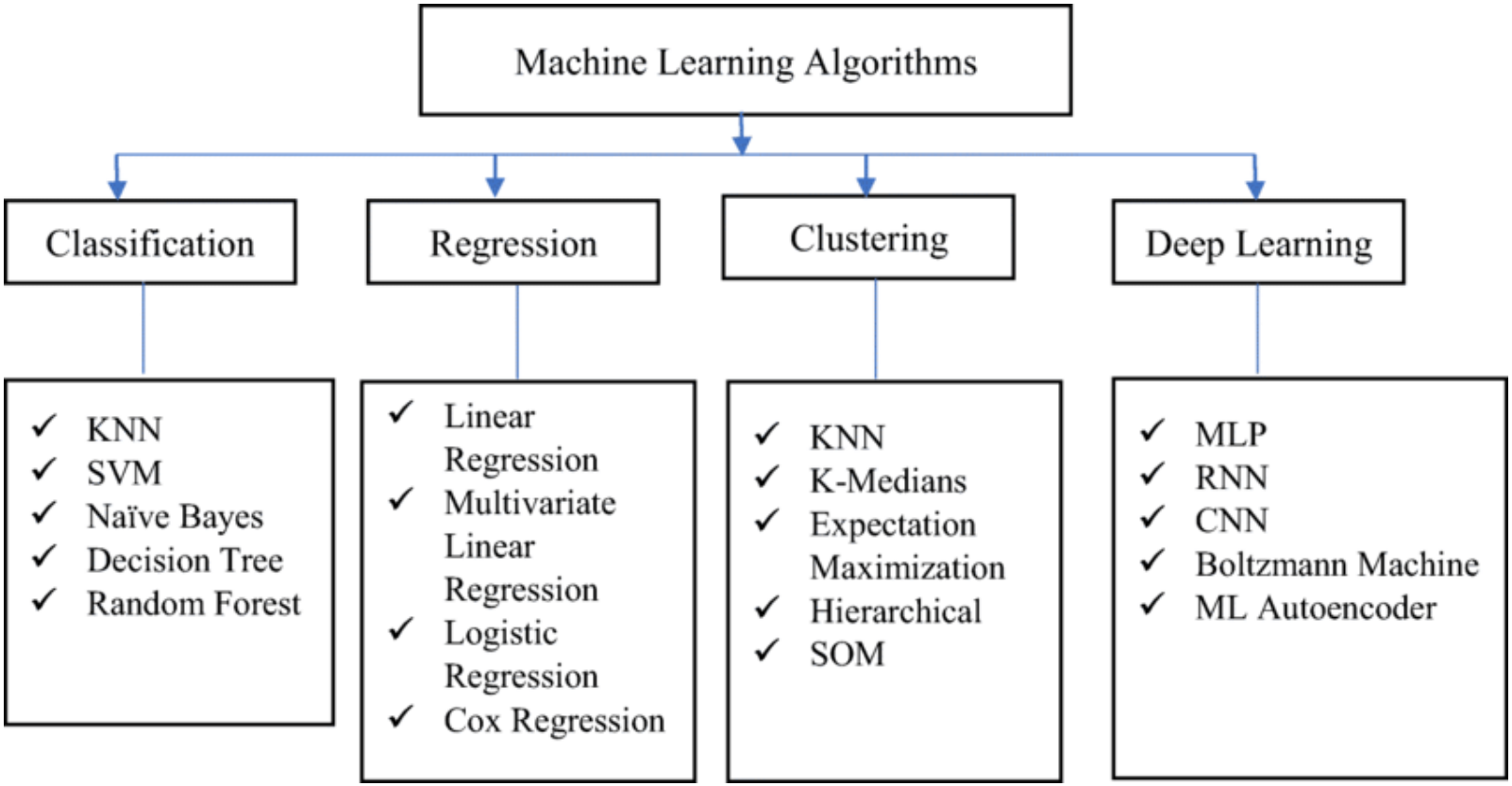
Make predictions



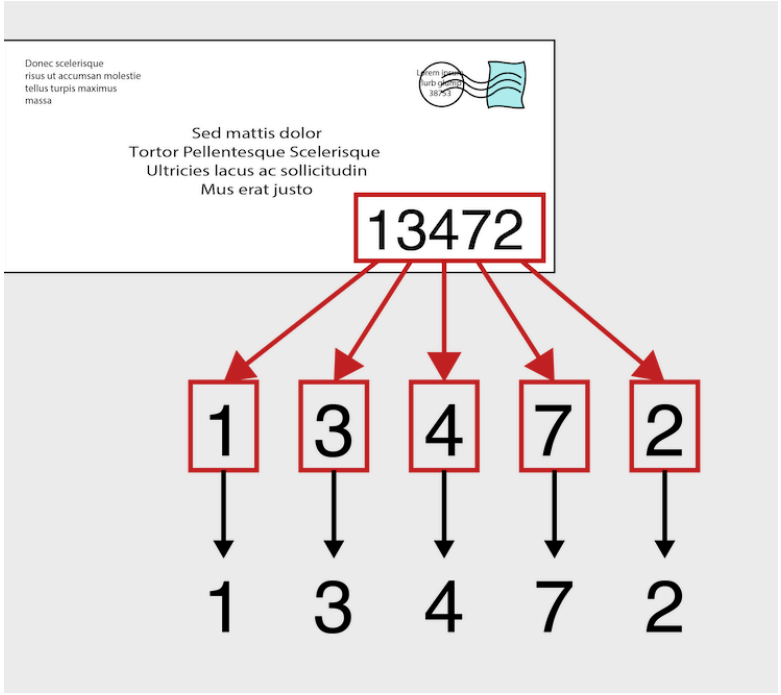
Machine Learning: What?



Machine Learning Approaches

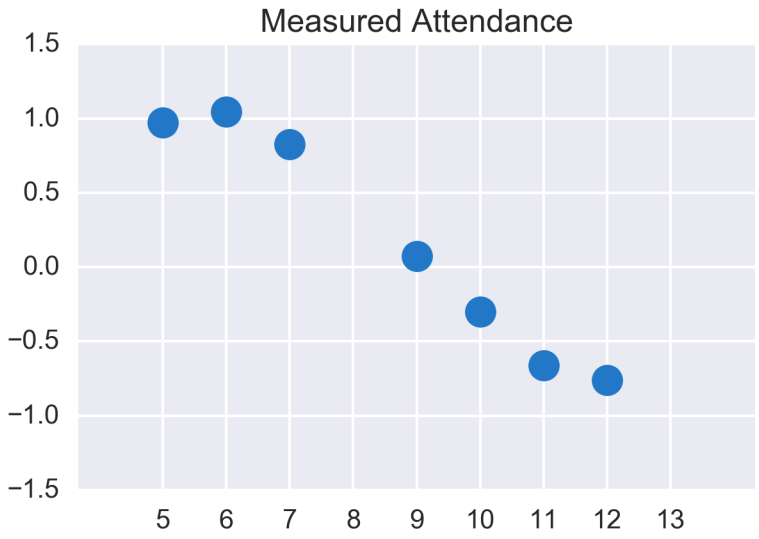


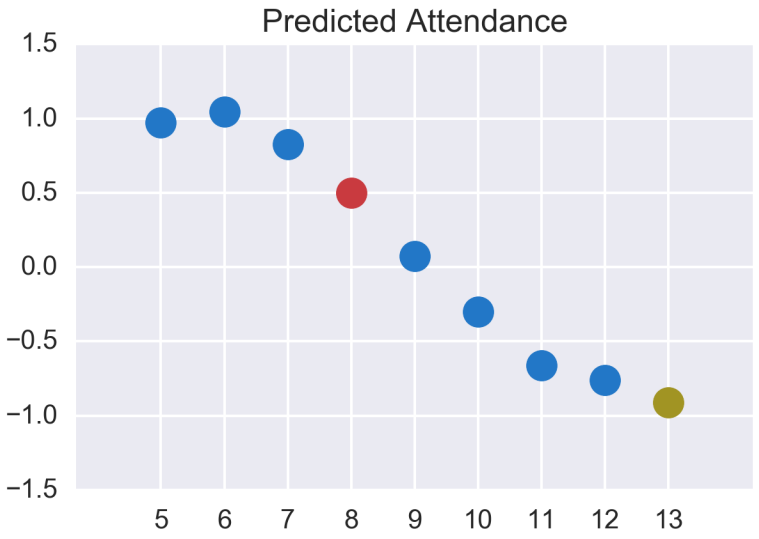
Machine Learning: Why?



Classification (Supervised Learning)

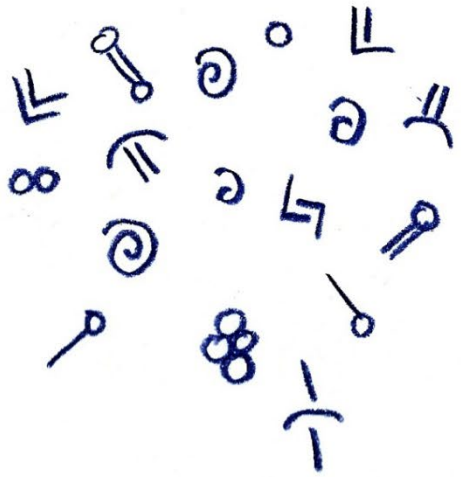
Machine Learning: Why?



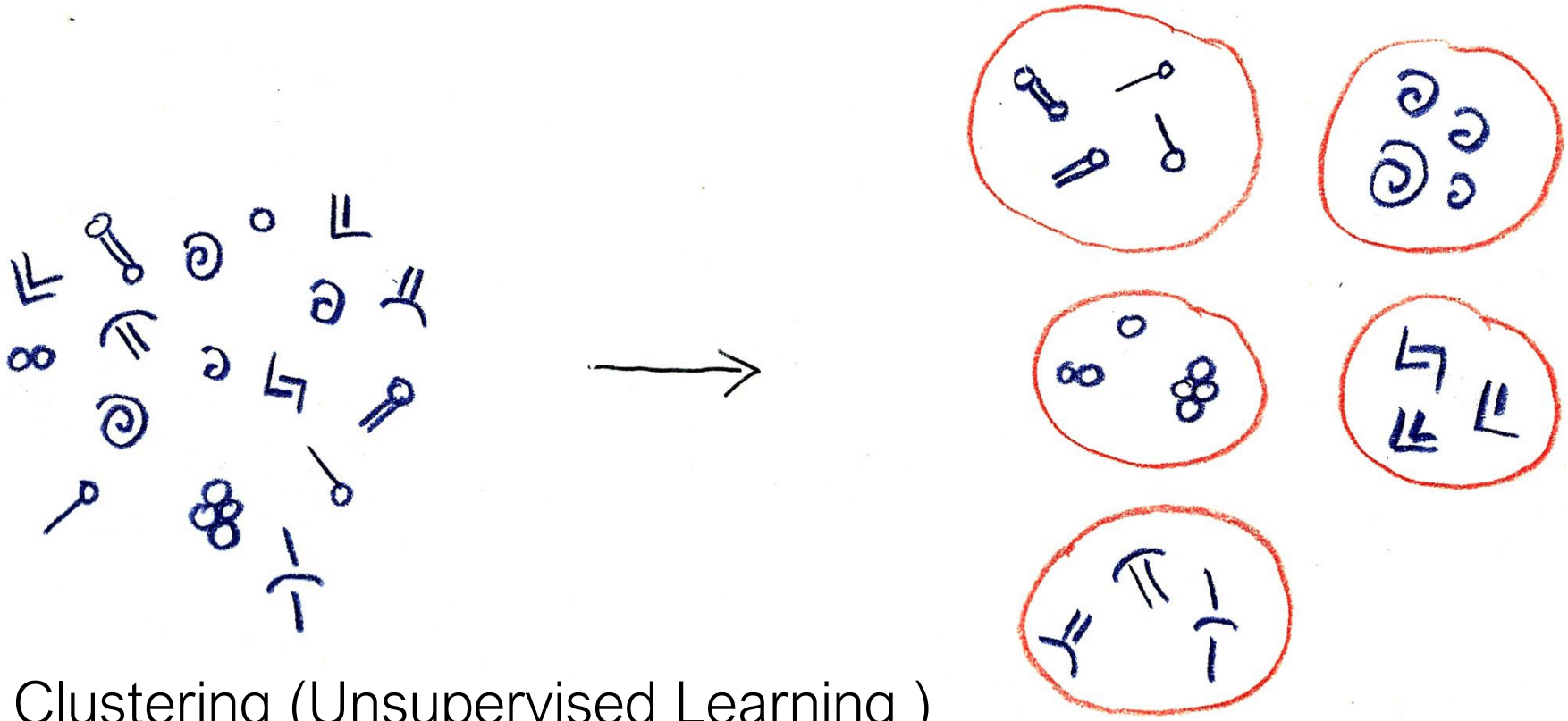


Regression (Supervised Learning)

Machine Learning: Why?



Machine Learning: Why?

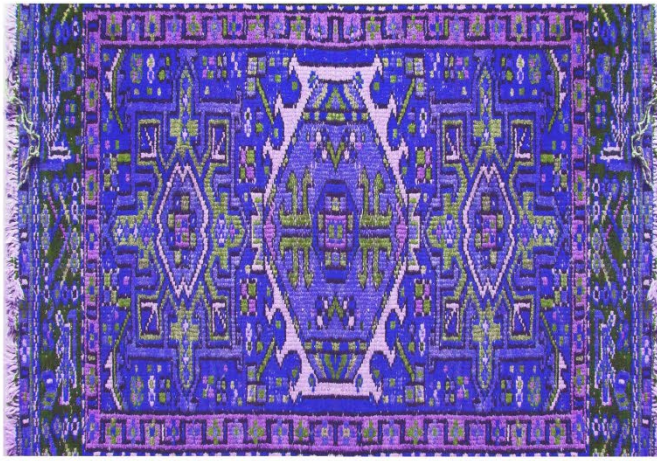


Clustering (Unsupervised Learning)

Machine Learning: Why?

Denoising (Deep Learning)

Machine Learning: Why?



Generate new samples (Reinforcement Learning)

Machine Learning: When?

- 1950's – Early concepts
  - **1952**: Arthur Samuel IBM creates checker program
  - **1957**: Rosenblatt, Neural Network Perceptron
- 1970's – First AI Winter
- 1990s – "New" Algorithms
- 2005's – Second AI Winter
- 2010's – GPU-acceleration Enabled Deep Learning
- 2020's – Data-driven applications to all domains

Machine Learning: How?

- Lots of Data
- Lots of Math
- Lots of Computing
- Lots of Energy Consumption
- Lots of Understanding, Reasoning and Intuition

…which brings us back to the course!

Class Logistics and Agenda

- Week 1 Agenda:
  - What is Machine Learning?
  - Syllabus and Course Overview
  - Introductions / Attendance
  - Types of Data
  - Numpy/Pandas Demo

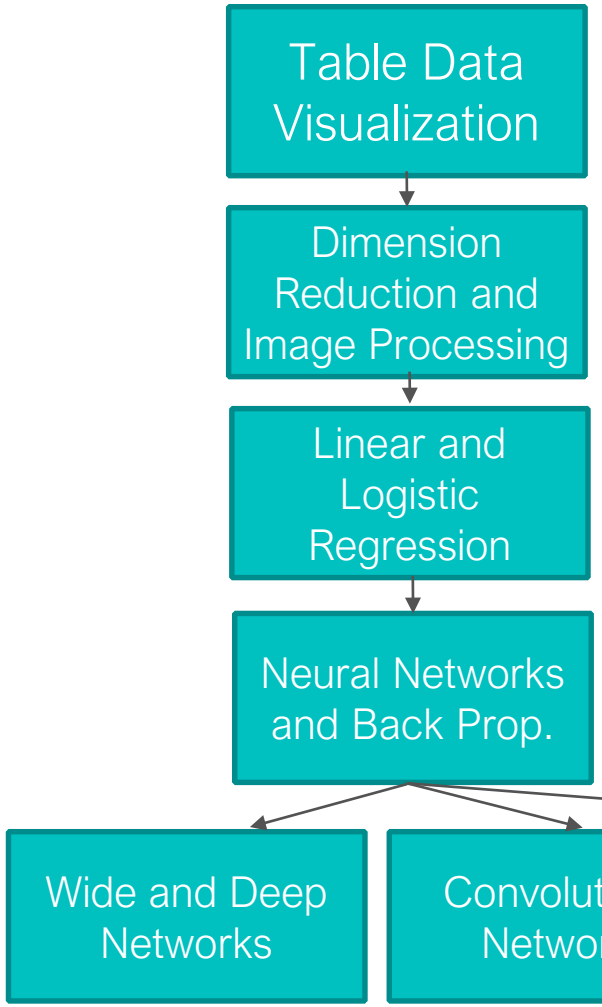
Class Logistics and Agenda

- Approach to this course:
  - Programming(Python, Object-Oriented Concepts)
  - Math(Linear Algebra, Calculus, Statistics)
  - **PROJECT BASED**
    - **Applications** and **Analytics**
  - **AI-tools**
    - Allowed with citations and clear disclosure (See syllabus)
    - Full credit provided only upon demonstrated understanding and disciplined use (e.g. validate correctness, edit output for relevance to prompt, etc.)

Expectations and Tools

- Use Canvas for posted course material
- Attend regularly
- Prerequisite knowledge:
  - Linear algebra, calculus (multivariate)
  - Basic statistics and probability
  - Basic python programming with object-oriented concepts
- Setup and manage your own environment for using jupyter notebooks (on your own machine; on jupyter colab; on HPC)
- Version of python: 3.X
  - JupyterLab (or notebook)
- Most Used Libraries:
  - Numpy, Pandas,
  - Scikit-Learn,
  - Matplotlib, Seaborn,
  - Tensorflow

Class Overview, by topic



Numpy, Pandas, Matplotlib, Seaborn
Overviews with some in-depth discussion

Scikit-learn, Scikit Image,
Intuition only, Some mathematics

Numpy, Recreate API for Scikit-learn
Detailed mathematics for simple optimization
intuition for advanced optimization

Numpy
Detailed mathematics for NN operations

Tensorflow, Keras
Intuition, Detailed implement.

Canvas Syllabus

- Lab Assignments
- Flipped Assignments
- Participation

- Course Schedule
- Difference between 5324 and 7324
  - Merged on Canvas – main tile CS 5324
- Differences between sections
  - Some slides modifications (posted on Home page)
  - Participation measured differently
- Office Hours (In person and by appointment virtually)
- Course Policies
- Canvas Technical Issues
  - Syllabus (Simple Syllabus)
  - Joining groups for assignments
- Zoom Study Room

Class Overview, by assignment

- **Lab One**: Visualize data and extract some features
- **Lab Two**: Analyze Images, Use dimensionality Reduction
- **Lab Three**: Program Logistic Regression in style of Sci-kit Learn
- **Lab Four**: Program NN Back propagation from Scratch, implement Adaptive Gradient Techniques
  - Use given dataset for this lab
- **Lab Five**: Wide and Deep networks
- **Lab Six**: Classify Images with Convolutional Networks
- **Lab Seven**: Classify Text with Recurrent Neural Networks

All Assignments posted on Canvas, with Rubric

Everything is a team assignment except quizzes and participation activities

Flipped Assignments (Live Coding)

- **Three "Live Coding" Days**
- **Required videos to watch in preparation**
- **Can not be rescheduled**
- **Open resources / open peer**

All Assignments posted on Canvas, with Rubric
Everything is a team assignment except quizzes and participation activities

Desired Student Outcomes

Thinkers with Skills

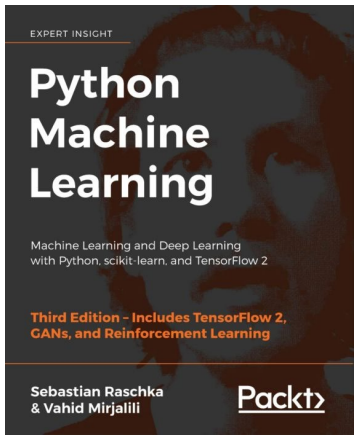
Professionals with Integrity
- chatGPT as tool

Teamwork Approach Encouraged
- Balance skillsets
- Coordinate and communicate
- Shared application domain interests

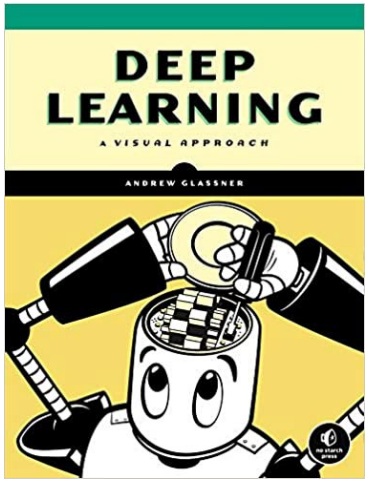
Brief Introductions / attendance

Resources and Course Materials Credits

- Text:
  - **Recommended**: Python Machine Learning, Raschka & Mirjalili, Third Edition

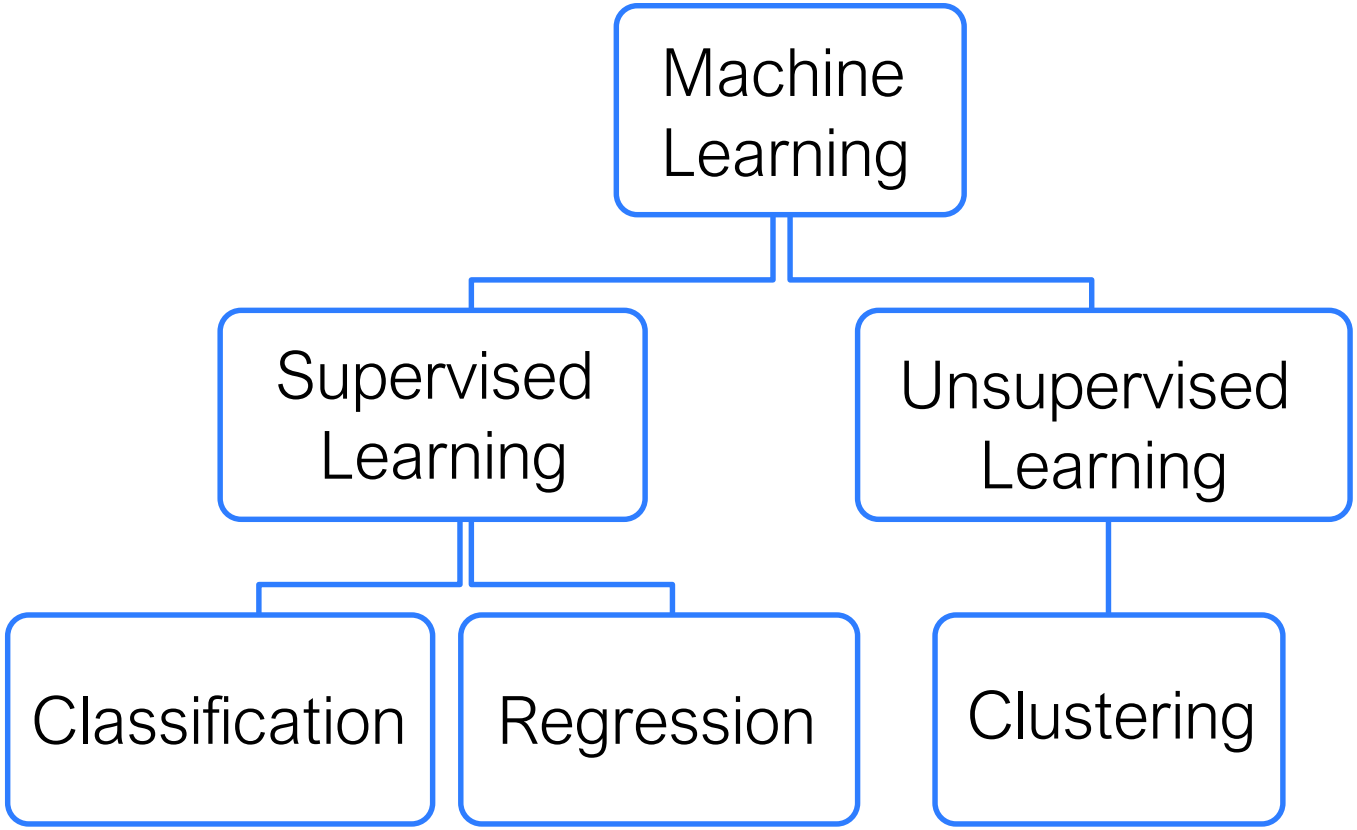


- Dr. Eric Larson's course materials on github
  - notebooks, slides, videos
  - Linked on course Canvas page

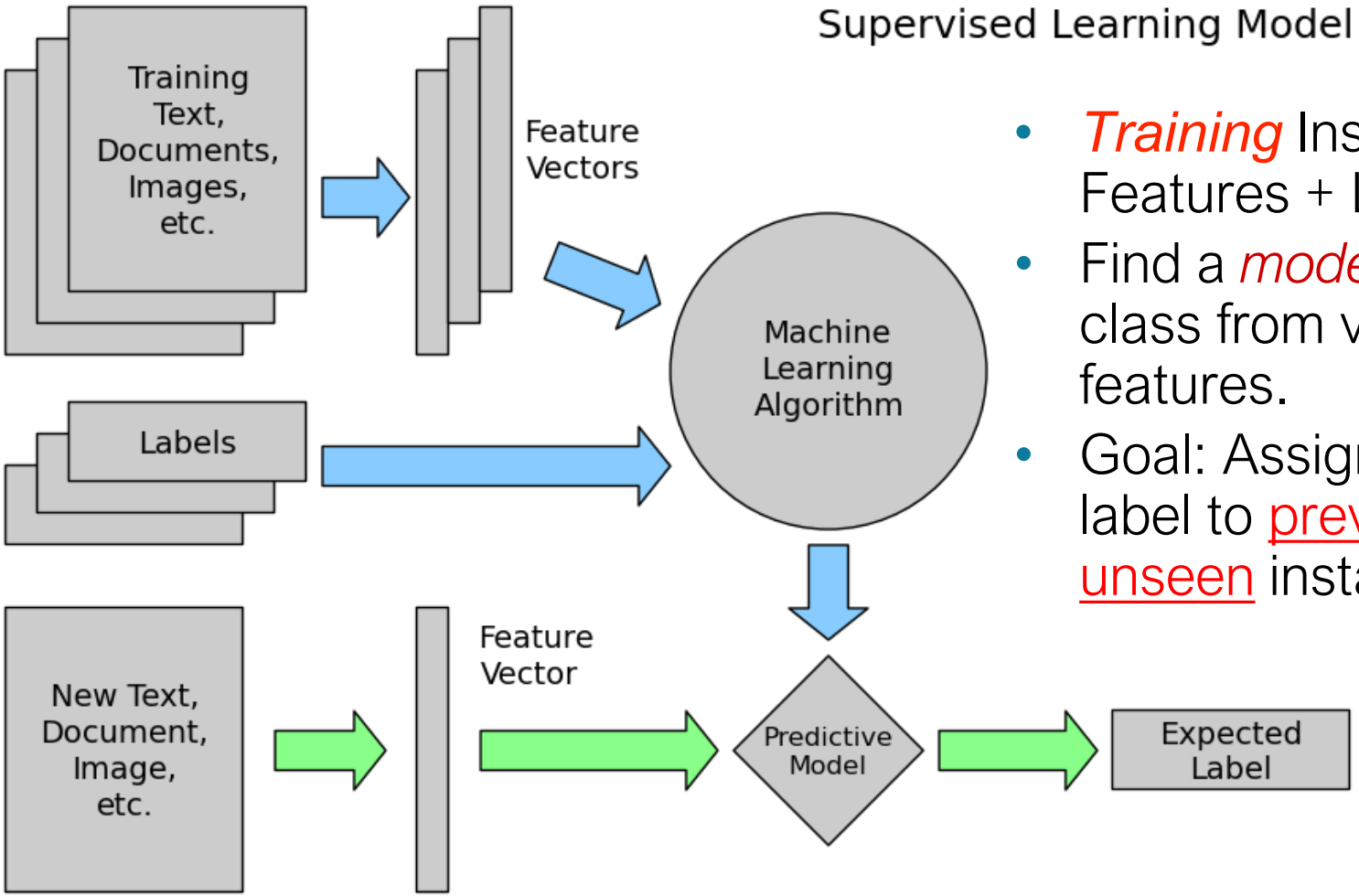


- Deep Learning: A Visual Approach
  - Andrew Glassner

High-Level Framework



Supervised Learning: Classification and Regression



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

Classification Task Overview

START WITH: Labeled data (many, many samples)

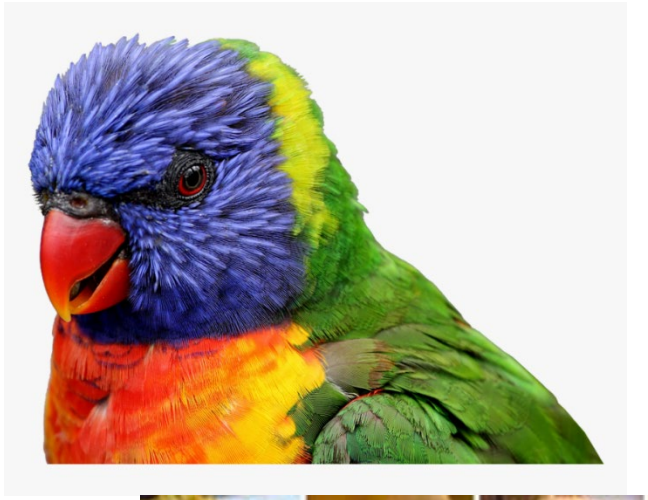
GOAL: Find a model that correctly predicts the label on *new data.* *(we say we TRAIN a model so that it LEARNS)*

ANALYSIS:

- How to represent the data?
- What can be learned from the data we train with?
- How to determine when our model is good enough?

Example: Muffin or Chihuahua?

Deploy: How do you think it will do?



Classification: Discrete

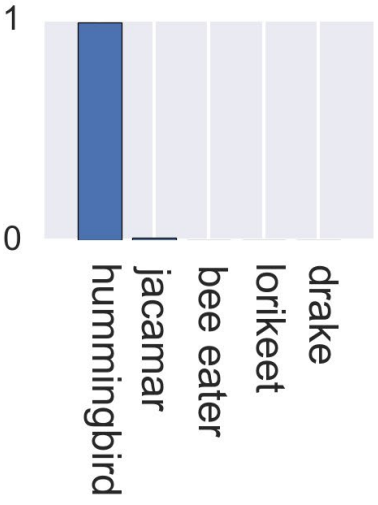
Parrot or guacamole?

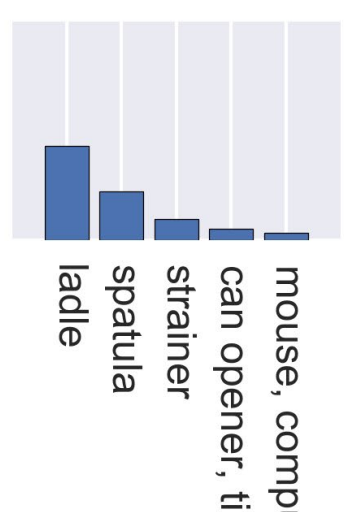
Classification: Discrete

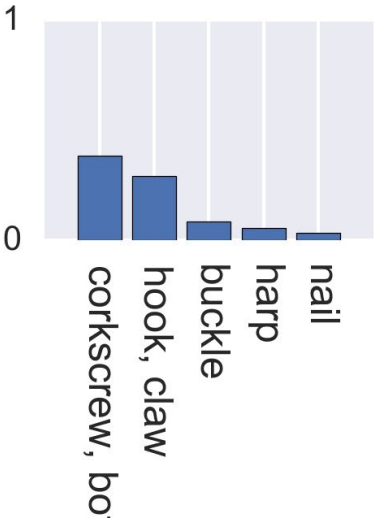
How many categories?

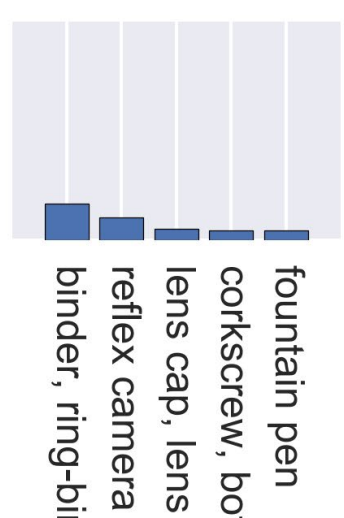
What is your confidence that this new image is in each category?

Towards object classification using image data









Example Classifying: Objects in Images

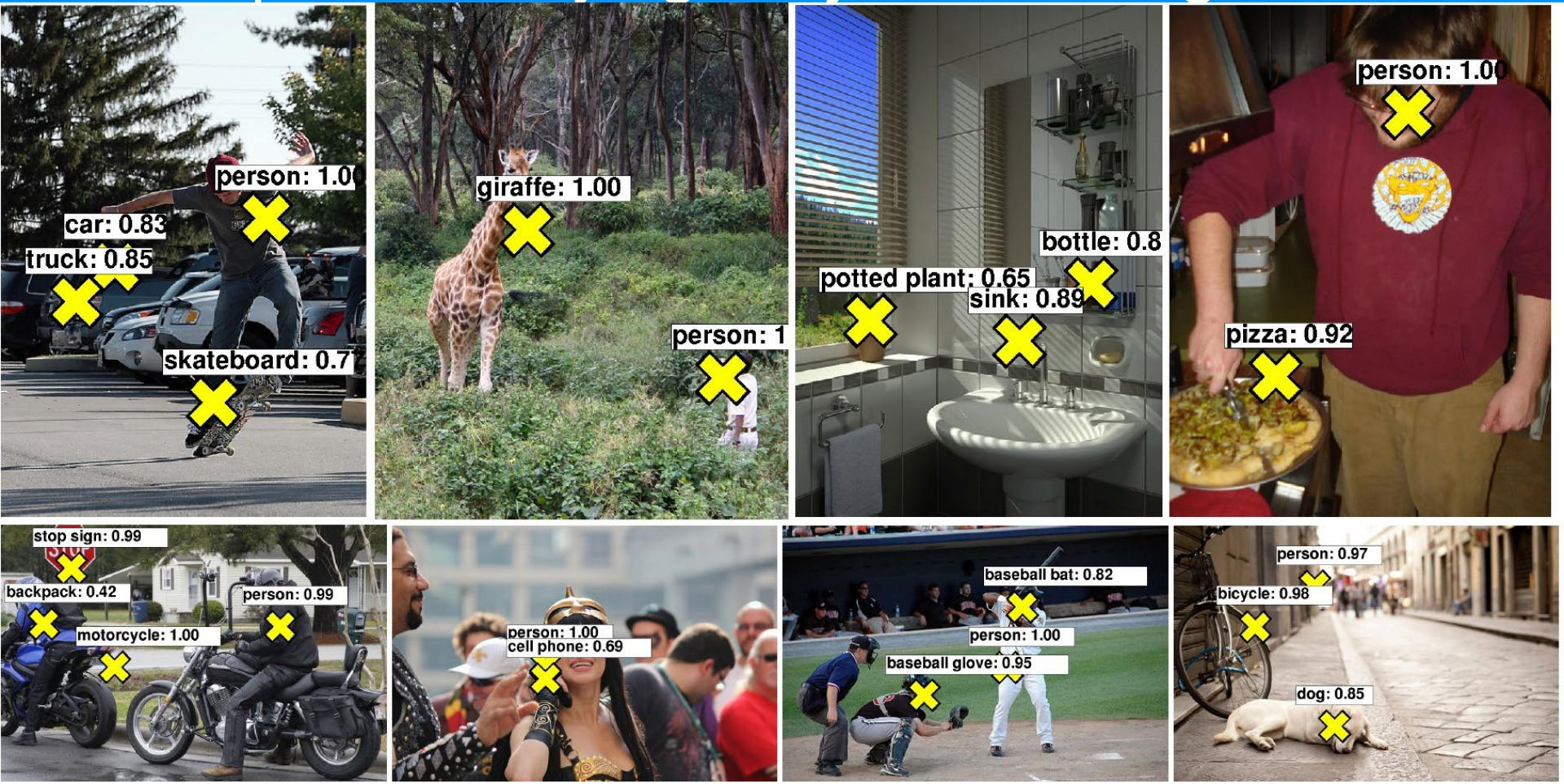


Image Net:

- 14 million images
- 200 Labeled Categories
- 1000 Location Labels

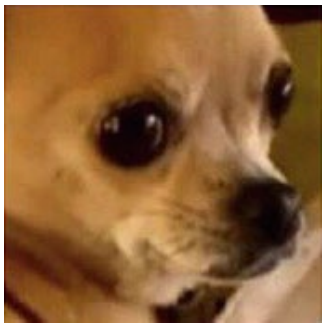
Data Features (Attributes):

- Images

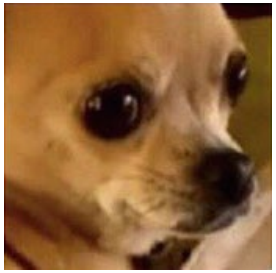
Classification: Discrete

Where are biggest differences?

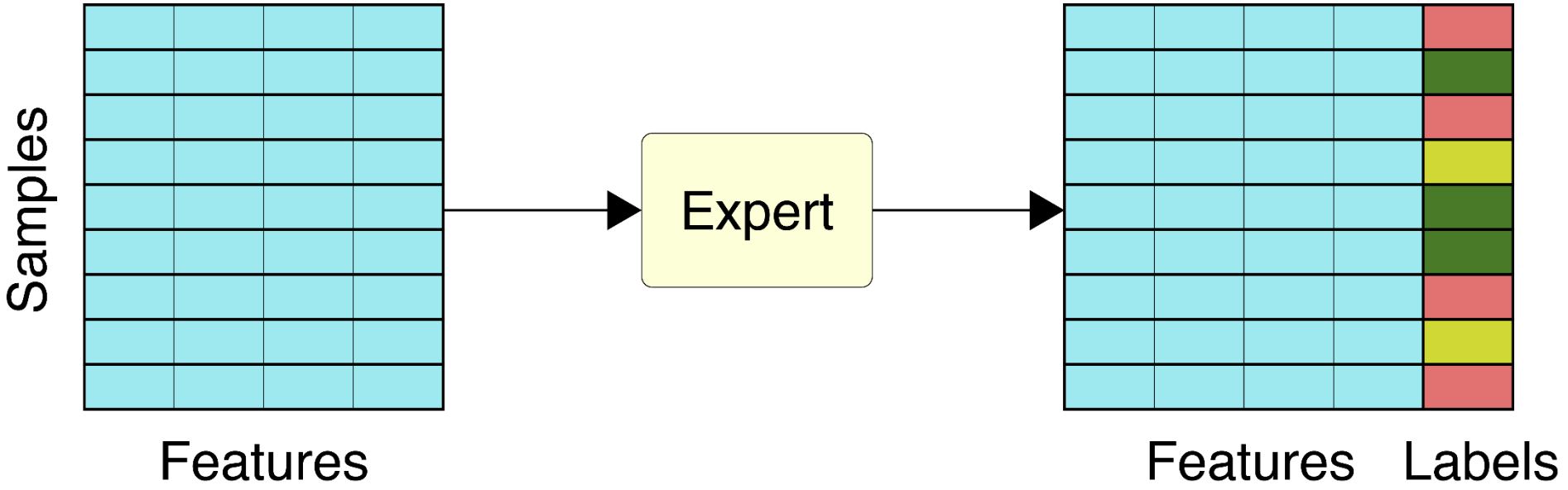
What is your confidence that this is in each of one category?



Towards defining a features space

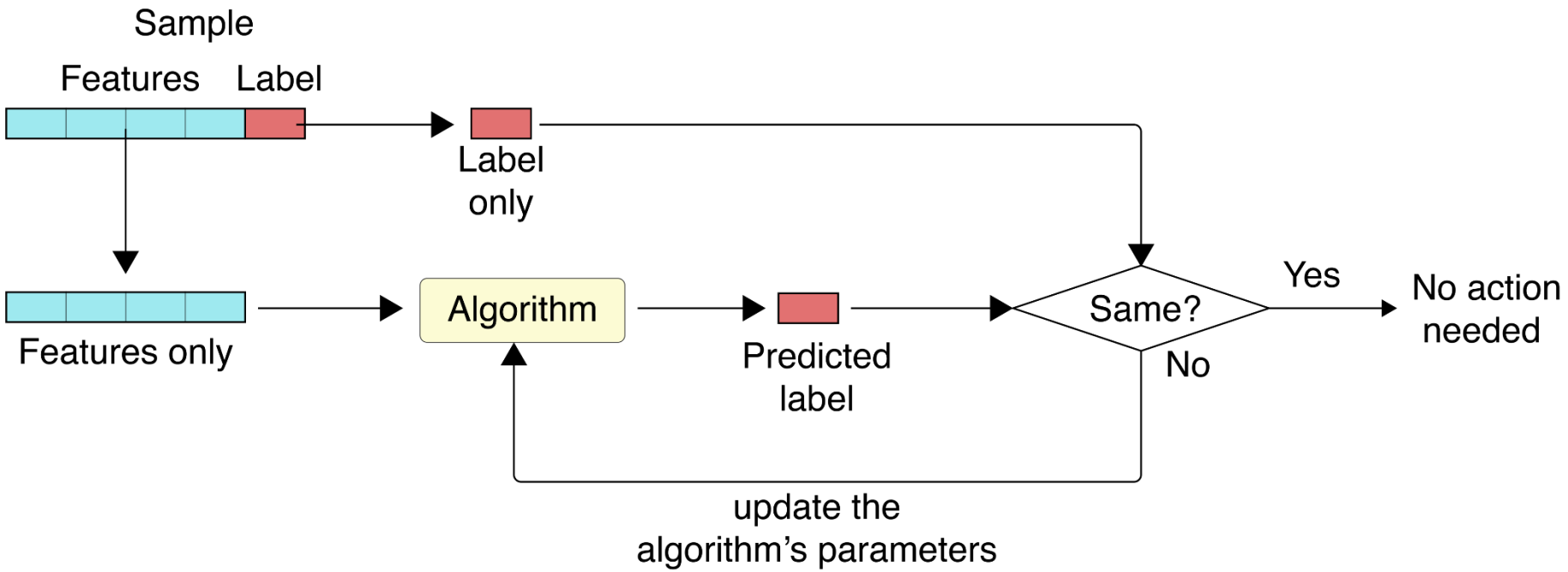


Define the features; Label the data

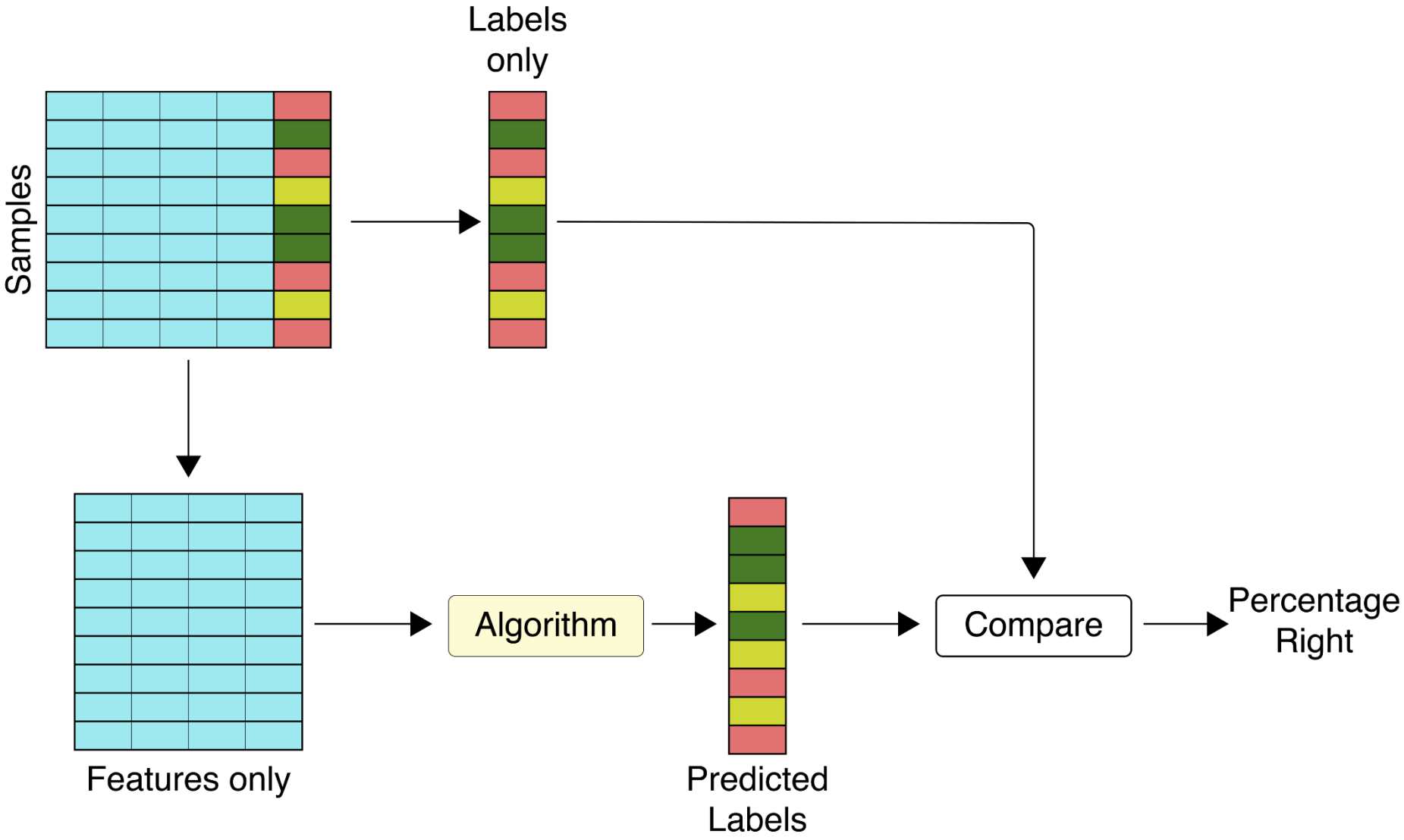


Process of training for a single sample* of data

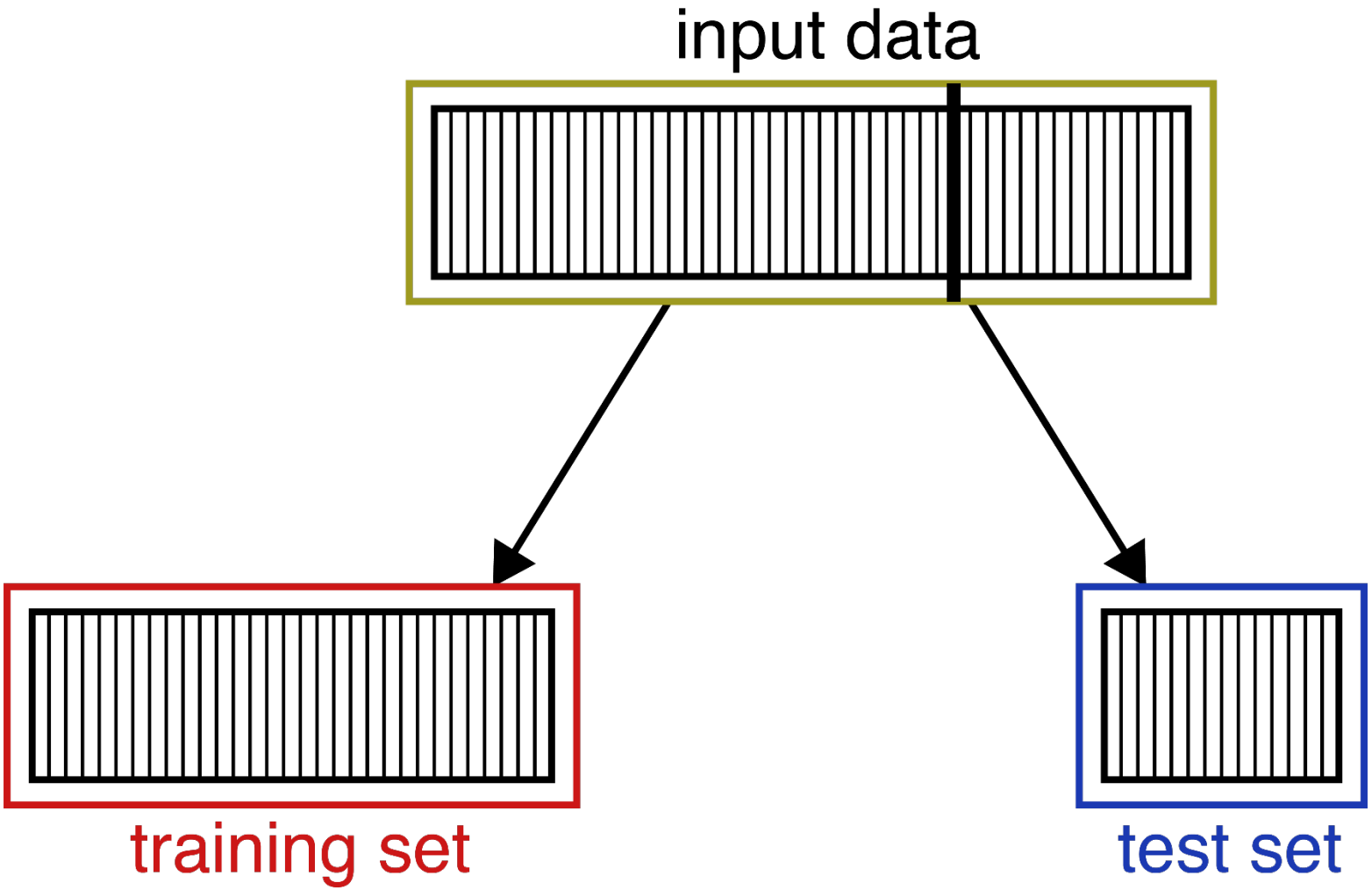
*Illustration only – single sample insufficient



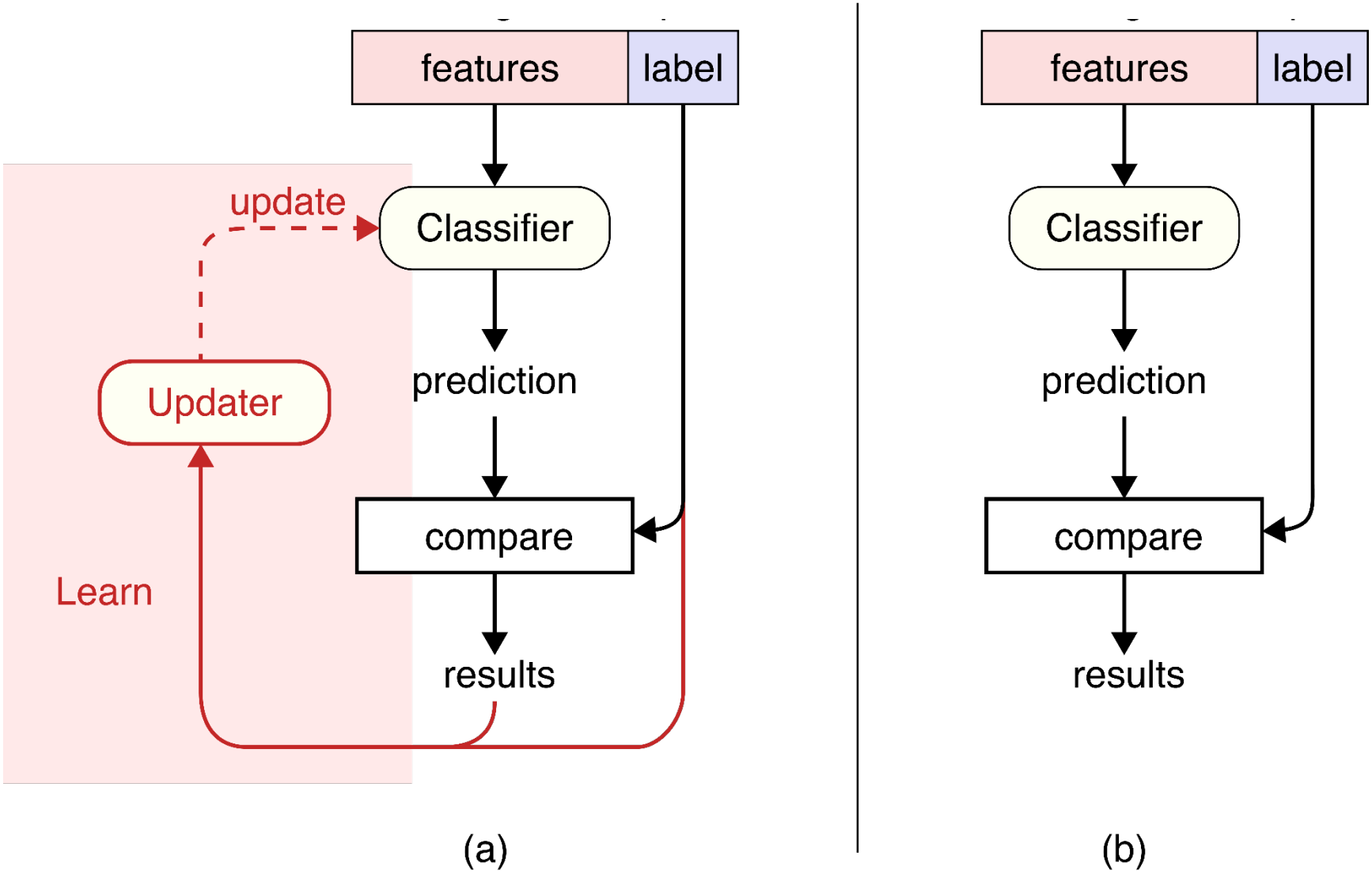
Train on many samples



Data set split to support train, test



Train, Test – must be separated!



Get Prepared …

- Prepare to use jupyter notebooks
  - install python on your laptop
  - refamiliarize yourself with Python basics
  - get ready to use Jupyter Notebooks
  - Install packages
    - NumPy, SciPy, Pandas, Matplotlib,seaborn
    - Tensorflow, jupyter

Let's discuss!

Book resource page has package list

- Guidance from Textbook here:
- https://github.com/rasbt/python-machine-learning-book-3rd-edition/blob/master/ch01/README.md

Resources…

- Canvas course resources
- TA support

Class Logistics and Agenda

- Week 1 Agenda:
  - What is Machine Learning?
  - Syllabus and Course Overview
  - Introductions / attendance
  - Types of Data
  - Numpy/Pandas Demo

Data

Features

Representation

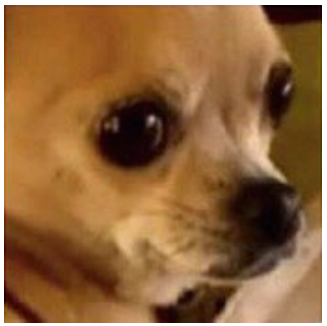
Quality

Redundancy

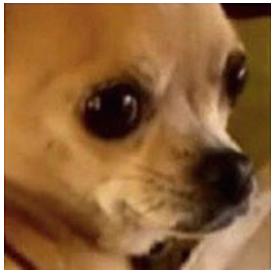
Classification: Discrete

Where are biggest differences?

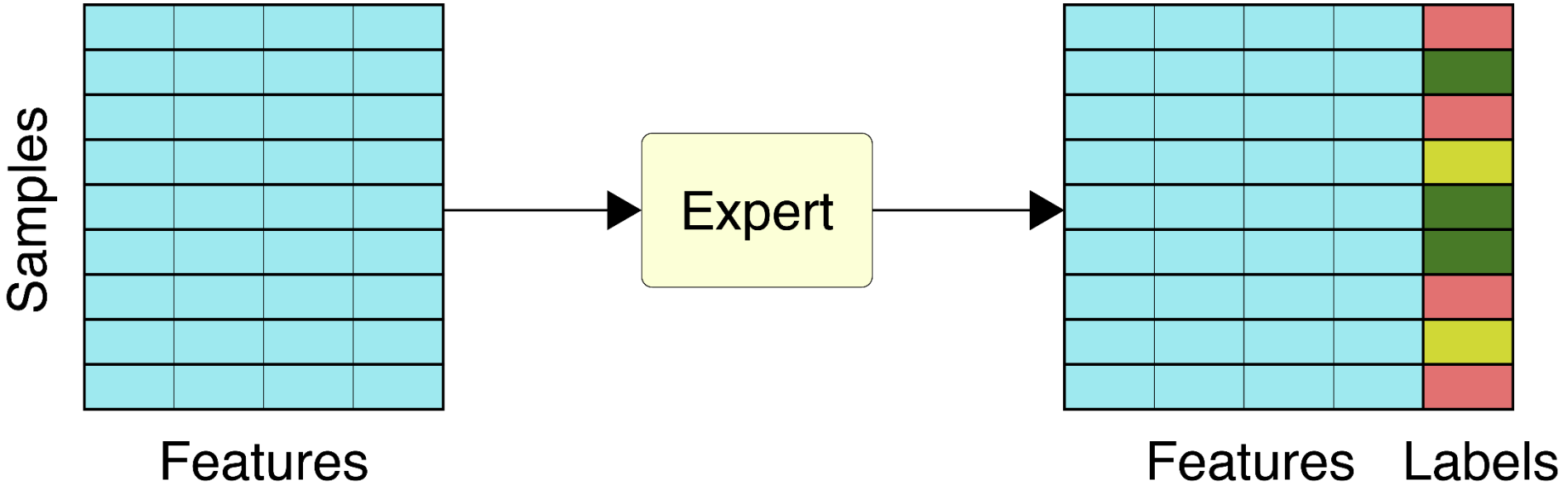
What is your confidence that this is in each of one category?



Towards defining a features space

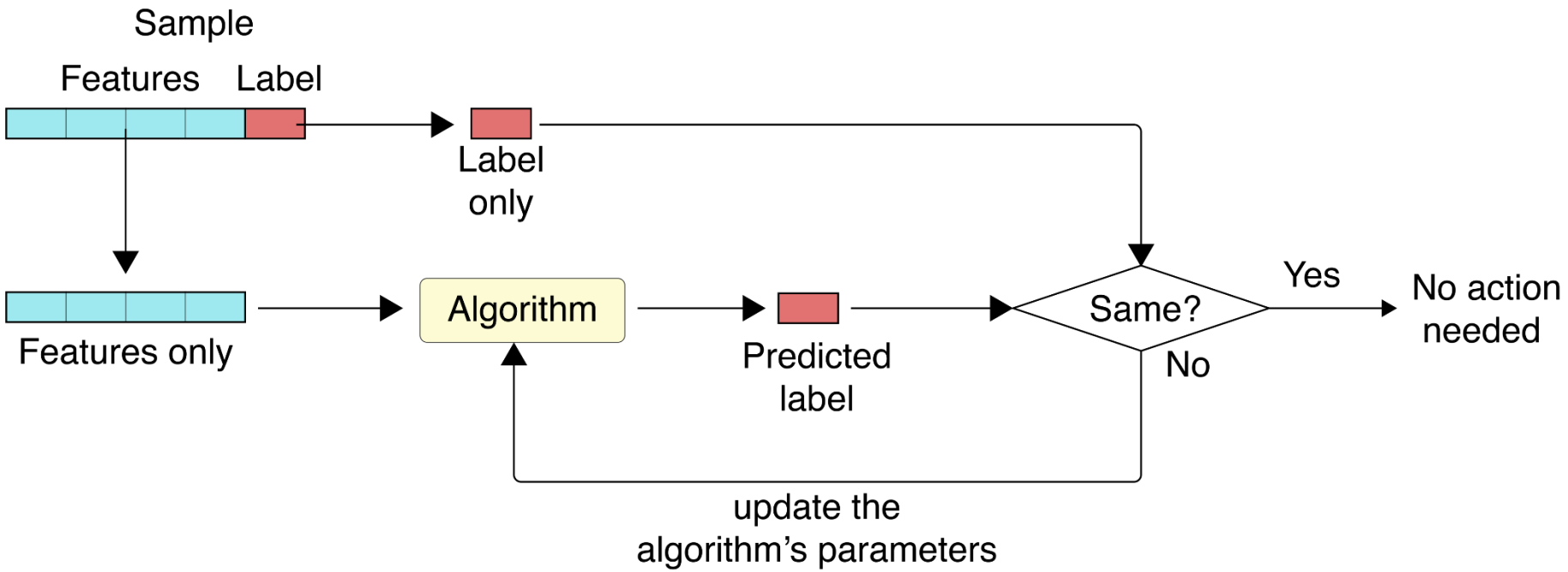


Define the features; Label the data

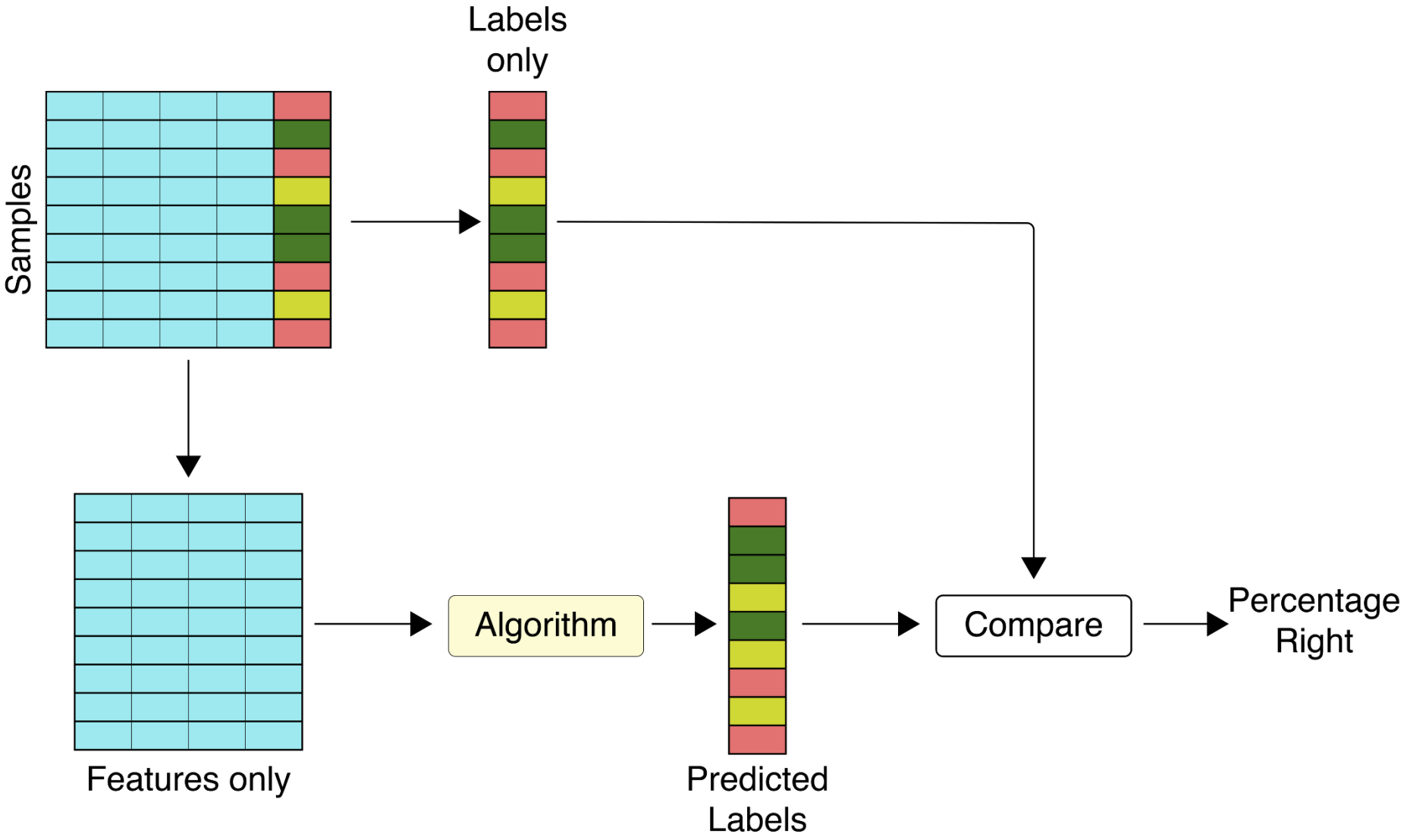


Process of training for a single sample* of data

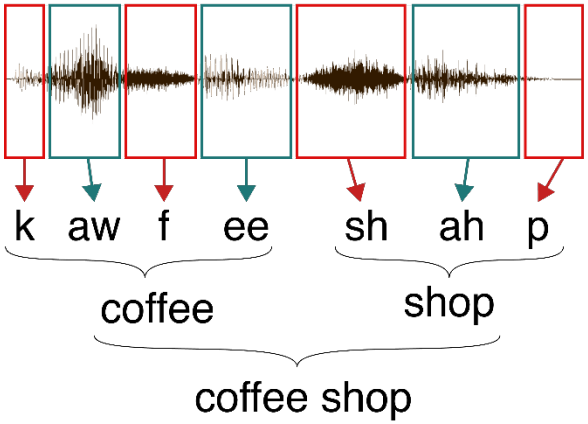
*Illustration only – single sample insufficient

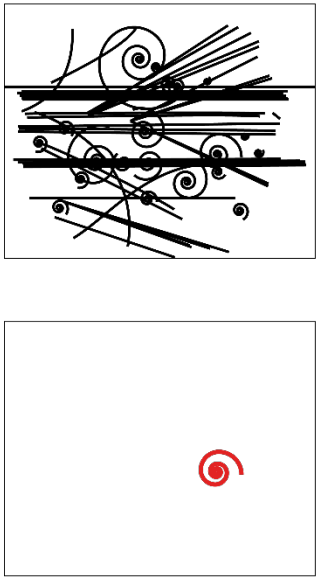


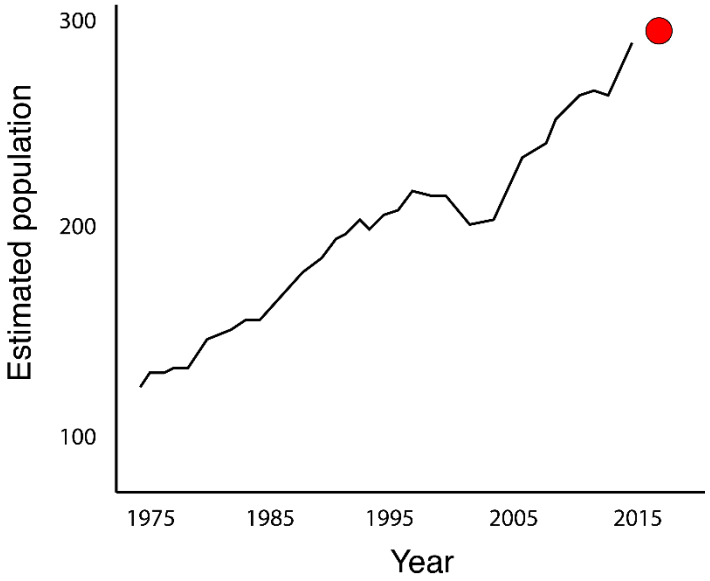
Train on many samples

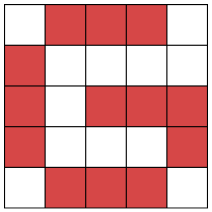


What data features represent our domain?









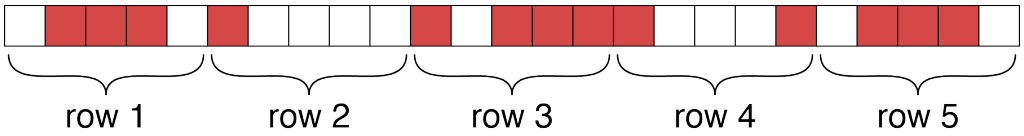


Table Data

- **Table Data**: Collection of data

  instances and their **features**

  - **Python:** Pandas Dataframe
  - **R:** Data.frame
  - **Matlab:** Table Class
  - **C++**: Trick Question

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

Objects, records, rows, 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

Feature Type Representation

	Attribute	Representation Transformation	Comments
Discrete	**Nominal**	Permutation of values only. one hot encoding or hash function	If all **employee ID** numbers were reassigned, would it make any difference?
Discrete	**Ordinal**	Order must be preserved `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 =f(old_value) + b` *f* is monotonic through origin float	Thus, the **Fahrenheit** and **Celsius** temperature scales differ in terms of where their zero value is and the size of a unit (degree).
Continuous	**Ratio**	`new_value = f(old_value)` *f* is monotonic through origin float	**Length** can be measured in meters or feet, but **zero is zero**

from Tan et al. Introduction to Data Mining

Concept Check

- Are these
- **A. ordinal,**
- **B. nominal, or**
- **C. binary**?

  - T-shirt sizes: Small, Medium, Large

  - T-shirt colors : RED, BLUE, WHITE, BLACK, GREEN

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

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

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

Demo:
Jupyter Notebooks

01_Numpy and Pandas Intro.ipynb

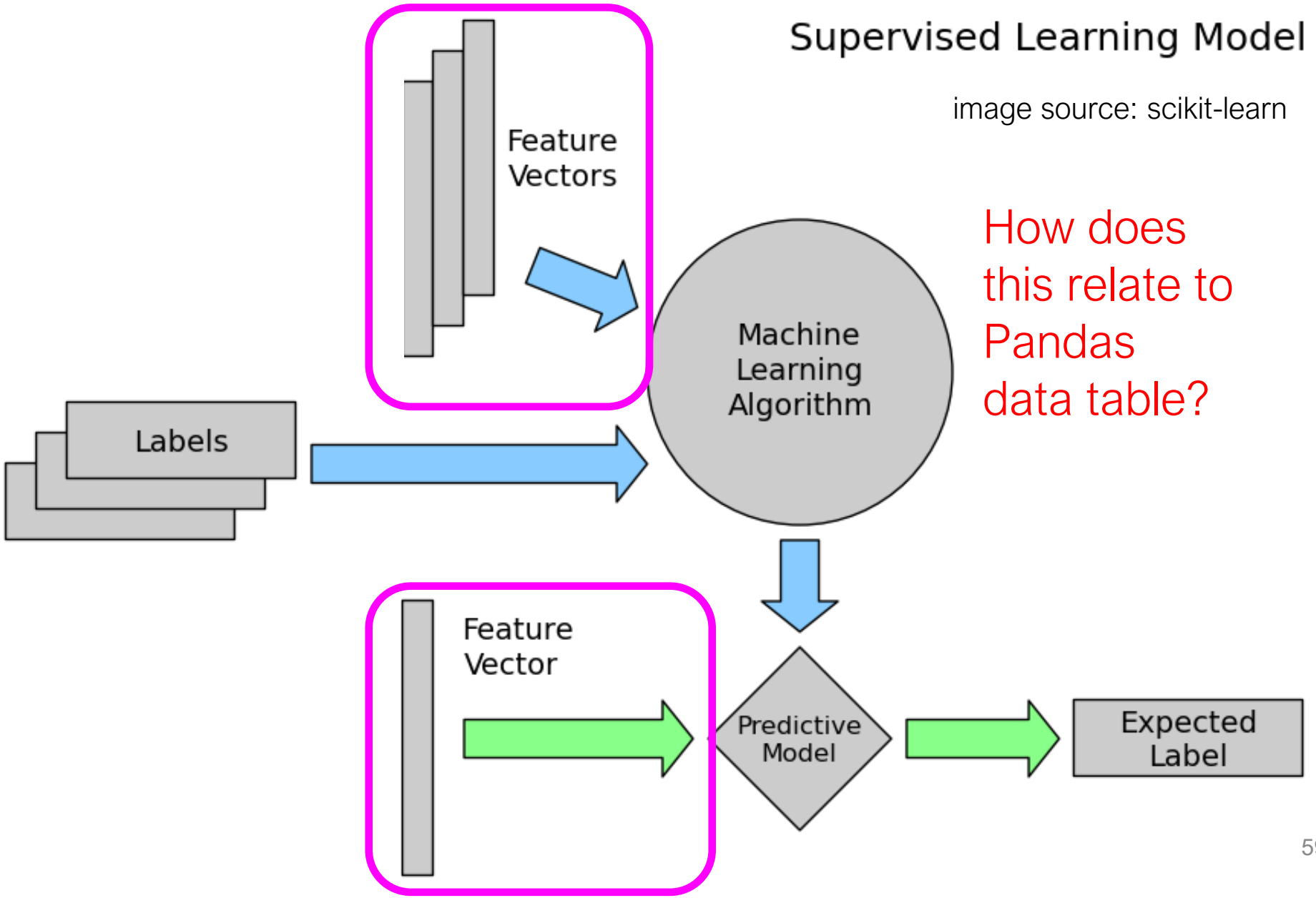
Demo

Warmup Assignment:

Jupyter Notebook Assignment

Visualization piece (Matlab or seaborn)

Review of Feature Data



Data Quality

Data Quality Problems

- Missing
  - Easy to find, NaNs
- Duplicated
  - Easy to find, hard to verify
- Noise or Outlier
  - Hard to define
  - Hard to catch

Information is not collected (e.g., people decline to give their age and weight)

Features **not applicable** (e.g., annual income for children)

UCI ML Repository: 90% of repositories have missing data

TID	Hair Color	Height	Age	Hired
1	Brown	5'2"	999	no
2	Hazel	1.5m	12	yes
3	Bl	5	23	no
4	Brown	5'2"	23	no

Handling Issues with Data Quality

- **Eliminate** Instance

TID	Hair Color	Height	Age	Hired
1	Brown	5'2"	999	no
2	Hazel	5'0"	12	yes
3	Bl	5'0"	23	no
4	Brown	5'2"	23	no

- **Eliminate** Feature

TID	Hair Color	Height	Age	Hired
1	Brown	5'2"	999	no
2	Hazel	5'0"	12	yes
3	Bl	5'0"	23	no
4	Brown	5'2"	23	no

Handling Issues with Data Quality

- **Eliminate** Instance or Feature
- **Ignore** the Missing Value During Analysis Replace with all possible values
- **Impute** Missing Values

How?

Is it a good idea?

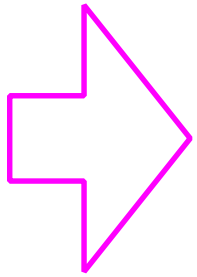
Stats?
mean
median
mode

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

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

K-Nearest Neighbors Imputation

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	?	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

For K=3, find 3 closest neighbors

TID	Pregnant	BMI	Age	Diabetes	Distance
3	Y	23.3	?	positive	0
6	Y	25.6	21-30	negative	(0 + 2.3 + 1)/3
2	N	26.6	31-40	negative	(1 + 3.3 + 1)/3
4	?	28.1	21-30	negative	(4.8 + 1)/2

Imputed Age: 21-30

How to calculate distance?

- Difference for valid features only
- May need to normalize ranges
- Or weight neighbors differently
- Or have min # of valid features
- Euclidean, city-block, etc.

K-Nearest Neighbors Imputation

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	?	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

TID	Pregnant	BMI	Age	Diabetes
1	1	33.6	2	1
2	0	26.6	1	0
3	1	23.3	?	1
4	?	28.1	0	0
5	0	43.1	1	1
6	1	25.6	0	0
7	1	31.0	0	1
8	1	35.3	?	0
9	0	30.5	3	1
10	1	37.6	3	1

Define a distance function across all features.
Use internal representation.

K-Nearest Neighbors Imputation

TID	Pregnant	BMI	Age	Diabetes
1	1	33.6	2	1
2	0	26.6	1	0
3	1	23.3	?	1
4	?	28.1	0	0
5	0	43.1	1	1
6	1	25.6	0	0
7	1	31.0	0	1
8	1	35.3	?	0
9	0	30.5	3	1
10	1	37.6	3	1

For K=3, evaluate distance among all and find 3 closest neighbors

TID	Pregnant	BMI	Age	Diabetes	Distance
3	Y	23.3	?	positive	0
6	Y	25.6	21-30	negative	(1-1)+(25.6-23.3) – (1-0) (0 + 2.3 + 1)/3 = 3.3/3
2	N	26.6	31-40	negative	(1 + 3.3 + 1)/3 = 5.3/3
4	?	28.1	21-30	negative	(4.8 + 1)/3 = 5.8/3

K-Nearest Neighbors Imputation

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	?	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

For K=3, find 3 closest neighbors

TID	Pregnant	BMI	Age	Diabetes	Distance
3	Y	23.3	?	positive	0
6	Y	25.6	21-30	negative	(0 + 2.3 + 1)/3
2	N	26.6	31-40	negative	(1 + 3.3 + 1)/3
4	?	28.1	21-30	negative	(4.8 + 1)/2

Imputed Age: 21-30

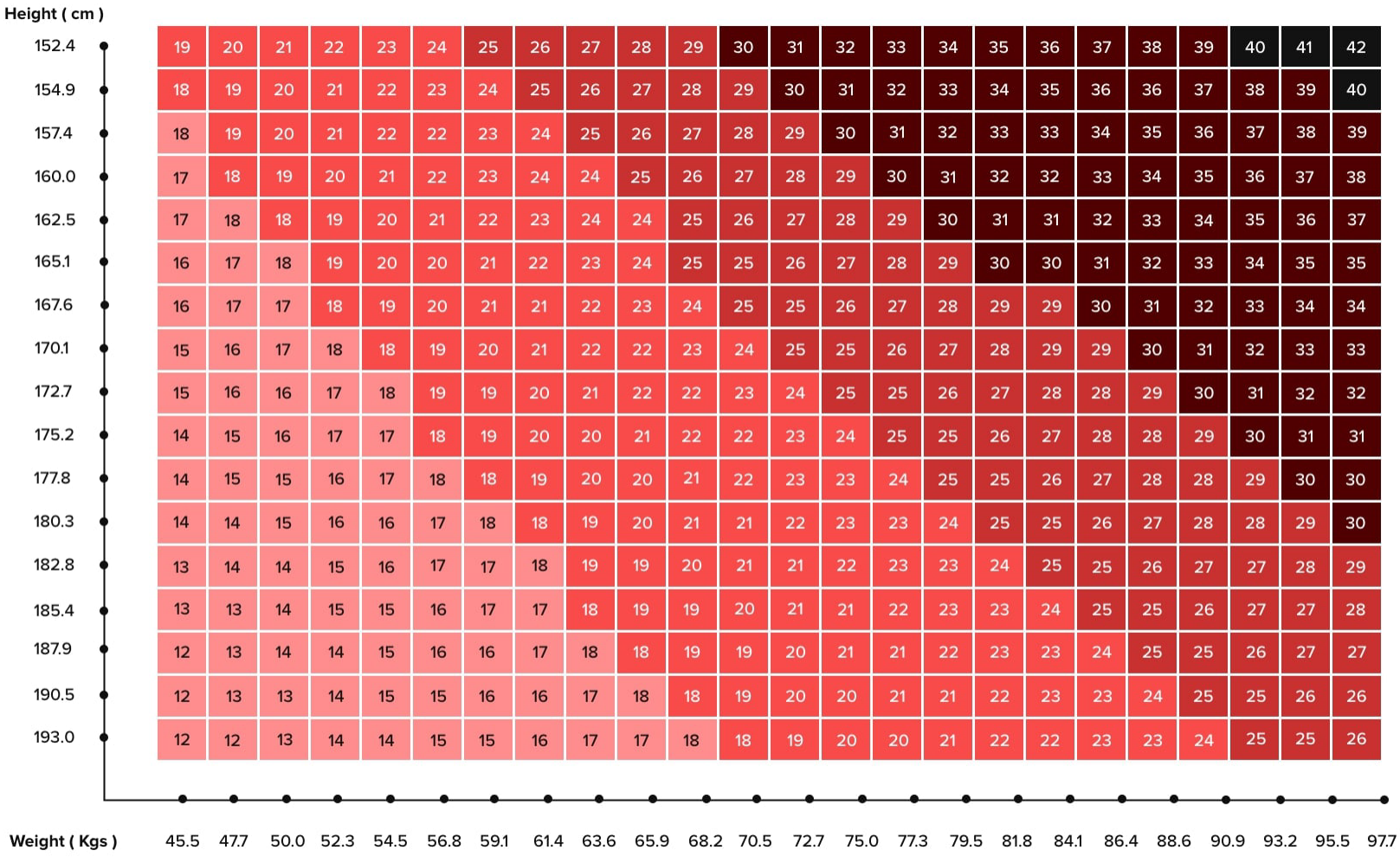
How to calculate distance?

- Difference for valid features only
- May need to normalize ranges
- Or weight neighbors differently
- Or have min # of valid features
- Euclidean, city-block, etc.

Data Redundancy

Table

TID	Pregnant	BMI	Height	Weight	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


Height (cm)
152.4 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
154.9 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 36 37 38 39 40
157.4 18 19 20 21 22 22 23 24 25 26 27 28 29 30 31 32 33 33 34 35 36 37 38 39
160.0 17 18 19 20 21 22 23 24 24 25 26 27 28 29 30 31 32 32 33 34 35 36 37 38
162.5 17 18 18 19 20 21 22 23 24 24 25 26 27 28 29 30 31 31 32 33 34 35 36 37
165.1 16 17 18 19 20 20 21 22 23 24 25 25 26 27 28 29 30 30 31 32 33 34 35 35
167.6 16 17 17 18 19 20 21 21 22 23 24 25 25 26 27 28 29 29 30 31 32 33 34 34
170.1 15 16 17 18 18 19 20 21 22 22 23 24 25 25 26 27 28 29 29 30 31 32 33 33
172.7 15 16 16 17 18 19 19 20 21 22 22 23 24 25 25 26 27 28 28 29 30 31 32 32
175.2 14 15 16 17 17 18 19 20 20 21 22 22 23 24 25 25 26 27 28 28 29 30 31 31
177.8 14 15 15 16 17 18 18 19 20 20 21 22 23 23 24 25 25 26 27 28 28 29 30 30
180.3 14 14 15 16 16 17 18 18 19 20 21 21 22 23 23 24 25 25 26 27 28 28 29 30
182.8 13 14 14 15 16 17 17 18 19 19 20 21 21 22 23 23 24 25 25 26 27 27 28 29
185.4 13 13 14 15 15 16 17 17 18 19 19 20 21 21 22 23 23 24 25 25 26 27 27 28
187.9 12 13 14 14 15 16 16 17 18 18 19 19 20 21 21 22 23 23 24 25 25 26 27 27
190.5 12 13 13 14 15 15 16 16 17 18 18 19 20 20 21 21 22 23 23 24 25 25 26 26
193.0 12 12 13 14 14 15 15 16 17 17 18 18 19 20 20 21 22 22 23 23 24 25 25 26
Weight (Kgs)
45.5 47.7 50.0 52.3 54.5 56.8 59.1 61.4 63.6 65.9 68.2 70.5 72.7 75.0 77.3 79.5 81.8 84.1 86.4 88.6 90.9 93.2 95.5 97.7