

Data X

Introduction to Data-X
Applied Data Science with Venture Applications
Data, Signals, and Systems

Ikhlaq Sidhu
Chief Scientist & Faculty Director
Sutardja Center for Entrepreneurship & Technology
IEOR Emerging Area Professor Award, UC Berkeley

Welcome to Applied Data Science with Venture Applications

Sutardja Center for Entrepreneurship & Technology
Industrial Engineering End Operations Research Department

Teaching Team

Course History

Prerequisite: Students should have:

- a working knowledge of Python
- completed a fundamental probability / statistics course
- basic understanding Linear Algebra.

Slowly Converting: Flip Model and More In-class Project Discussion



Course Philosophy

Data-X



Make the Tools



Use State-of-the-Art
Open Source Tools



Architect
the System



Sell, market, and
pitch the product

Most CS / Math

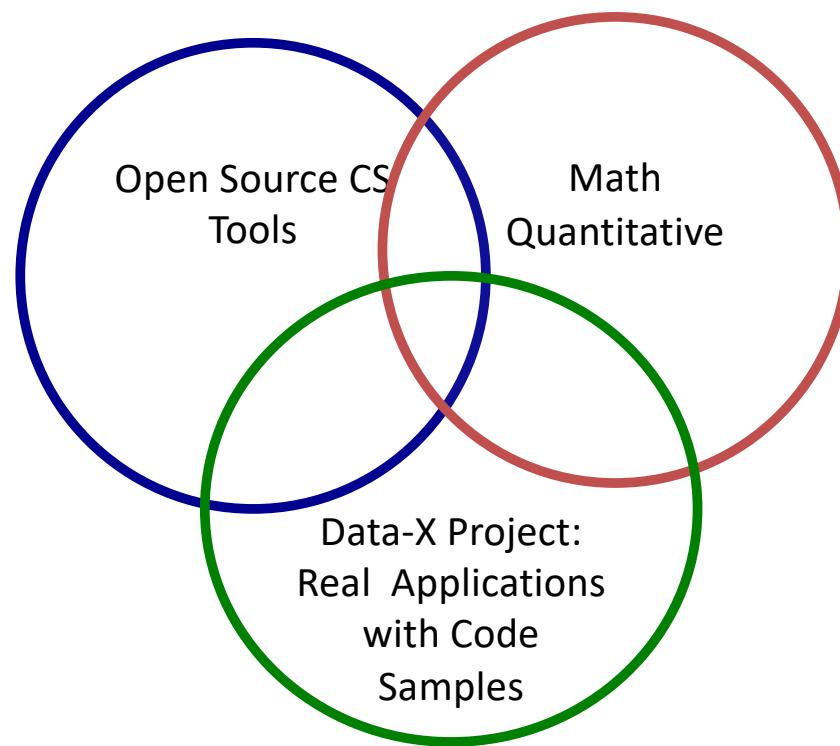
Data-X

Business Topics



What is in this course

- Course Materials
- Applied Project
- Holistic Perspective:
Industry, Social
Applications,
Customer Driven



Holistic Perspective: Industry, Social Applications, Customer Driven

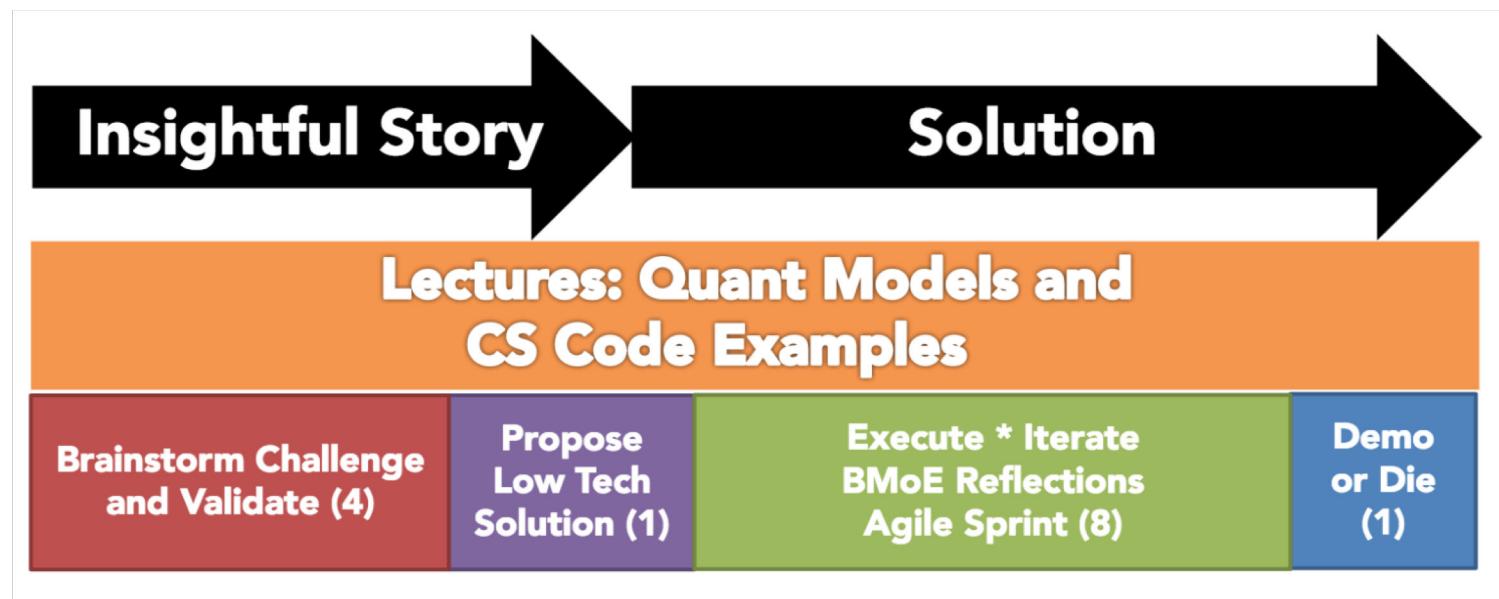


What is taught in the class?

- The ML stack most commonly used in creating ML/AI/Data applications
- Application and systems viewpoint of data and ML
- Implementation, architecture, and relevant processes to build real systems
- Connection with relevant mathematical, statistical foundations (optimization, entropy, correlation, LTI, prediction, classification)
- Practical insight into advanced techniques and tools: (eg. CNNs, NLP, scraping, recurrent networks, etc.)
- System modeling for data applications
- Application talks: Recommender systems, Blockchain, Spark etc.



Course Overview



Open-ended, real-world project: Typically 5 students, with available advisor network

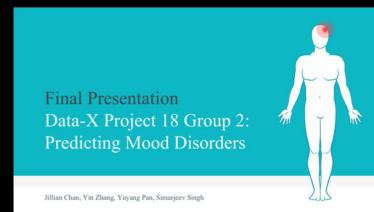
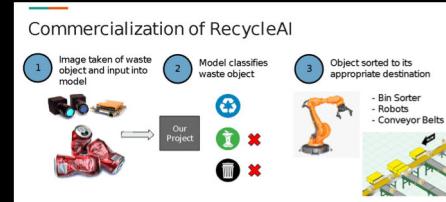


Data-X Project Examples

- Detection of fake news
- Prediction of long-term energy prices
- Automatic recycling through image recognition
- AI for crime detection, traffic guidance, medical diagnostics, etc.
- A version of Zillow that is recalculated with the effects of AirBnB income
- Signal processing and pattern analysis to improve earthquake warning systems
- Early Autism Detection
- Secure Health Records stored on a Blockchain

find many, many more at:

Data-X www.data-x.blog/projects

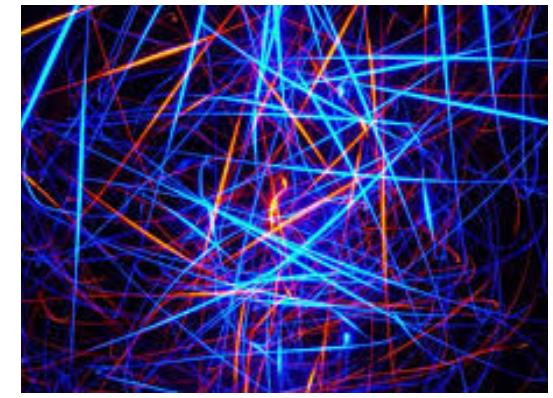


Project Types



Business or Consumer
Use Case

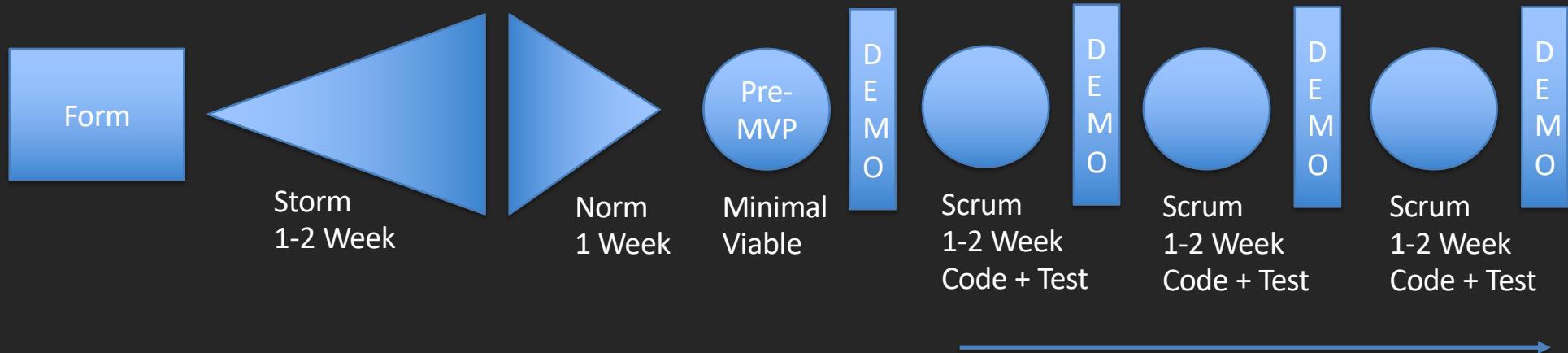
Social Impact
(or improve part of a data pipeline
or work towards a research result)



Its Just Cool



Implementation Behaviors and Process



1. Form, Storm, Norm
2. Minimum Viable
3. Key skeleton components
4. Hypothesis → Test → Record
5. Agile Model for Feature Increments (for a changing objective)
6. Agile Analytics

4 or more iterations



Most Resources Are Available at data-x.blog

1. Go to Data-X.blog
 - Syllabus
 - Instructions for SW Install
 - Link to GitHub with Cookbook Code Samples and Slides
2. Download Instructions to Install Python 3.x Anaconda Environment. For now you only need Anaconda, don't worry about other packages that are not already included.
3. Be able to create your own Jupyter notebook
4. Self-Review Python references as needed. See Ref CS01 and as needed BIDS Python Bootcamp.

The screenshot shows the Data-X at Berkeley website. The header features the text "DATA-X AT BERKELEY" and "A Framework for Digital Transformation". Below the header is a navigation bar with links: Home, Resources, Syllabus, Posts, Labs, Advisors, and Contact. The main content area is titled "SYLLABUS" and includes a link to "Edit". Below this, course information is provided: "Applied Data Science with Venture Applications" and "IEOR 135/290-002". It also lists the "Instructor: Ikhlaq Sidhu" and "Department of Industrial Engineering & Operations Research". A note states "3 Units, Lecture and Lab".

We will also be adding more video lectures to the Data-X On-line Tab



Project Ideation

- Past Projects Concepts:
 - See the Advisor's Tab of data-x.blog
- Past Projects:
 - See the archive on the Posts page and on the Labs page of Data-x.blog
- Combine ideas or extend previous work
- You can also choose to build part of a system,
 - i.e., just the part that automatically collects data by web scraping, or
 - just the part that makes a decision based on data already available

The screenshot shows a website interface with a navigation bar at the top. The navigation bar includes links for Home, Resources, Syllabus, Posts, Labs, Advisors, and Contact. Below the navigation bar, there is a section titled "BLOCKCHAIN ADVISING: BLOCKCHAIN AT BERKELEY". Under this section, there is a heading "Project Concept Links:" followed by a list of 18 items, each with a link. Below this, there is a heading "Extended Mentor Network:" followed by a list of 1 item.

Project Concept Links:

- New Venture Success ([link](#))
- Concept: Blockchain based social currency to regulate social platform such as Twitter ([link](#))
- Concept: Personal Genome Hacking ([link](#))
- Concept: Holy Grail of Venture Capital ([link](#))
- AI Music Software development student cooperation opportunity ([link](#))
- Concept: Predicting future outcomes based on historical records ([link](#))
- Concept: US Power Plant project ([link](#))
- Concept: Multi-disciplinary data analysis of common psychological conditions ([link](#))
- Concept: Visualizing investment opportunities in touristic regions (Open Data for Greece 1.0) ([link](#))
- Concept: The University Bot ([link](#))
- Concept: Materials Recycling using Machine Learning
- Concept: Insights from Personal Photos
- Fuzzy Joins – A Modeling Discussion for Probabilistic Joins in Data Tables
- Concept: Faculty Research Matching with NLP and ML
- Concept: Inferred Information via Probabilistic Joins

Extended Mentor Network:

- Amir Najian, Geospatial Data Scientist at RMS – Geospatial Machine Learning and uncertainty modeling in Geocoder systems



Homework For Week 1

- HW Part 1: For Your Project – By Next week
- Come up with 3 ideas for class projects in 1-3 sentences.
- A systems or application you will build
- **Communicate:** WHO the project is for, WHAT will it do, WHY this is needed/valuable.
- ---
- Homework Part II
- Python-based review notebook (Breakout and Homework, BKHW). To sent by email.



Data X

Introduction Data, Signals, and Systems

Ikhlaq Sidhu
Faculty Director and Chief Scientist
Sutardja Center for Entrepreneurship & Technology
IEOR Emerging Area Professor Award, UC Berkeley

An Overview of Data and AI Applications

Data X

Basic Concept of Working with Data



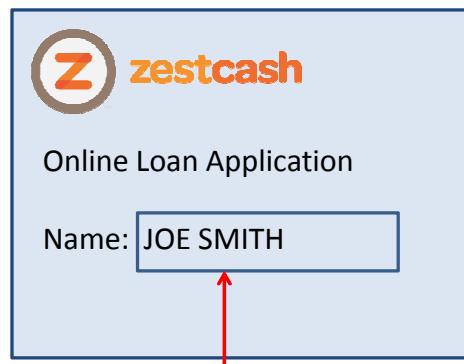
- Data Wrangling
- In Production



Example: Data and Information is a competitive advantage

Real-life Example: ZestCash

- “All data is credit data”



The data says: greater credit risk!

The data says: lesser credit risk!

Reference: Shomit Ghose



- Service provider of Gambling and Casinos
- Entry Card
- Pain points
- Intervention



Harrah's Casino: Knowing your customer

PLAY & WIN ▶

Reference: Supercrunchers

Why: More Simply

Customer
Insight/
Engagement

Operations:
Reliable &
Predictable

Security &
Fraud



Financial Firms

Network Security



Data X

Implementation: SW Tools / Stack

Data X

The Most Common Open Source Tools: AI/ML Stack

Start with Python as an interface
Jupyter Notebooks for prototyping

- Python: The interface
- NumPy, SciPy: Working with Arrays
- Pandas: Working in Tables, SQL to Pandas
- Sklearn: ML
- Matplotlib: Visualizing Data
- TensorFlow, Keras: Neural Networks
- SQL to Pandas
- NLP / NLTK: Natural Language
- Spark: For large data sets (GB, TB+)



<https://www.youtube.com/watch?v=Q0jGAZAdZqM>

<https://conda.io/docs/user-guide/install/download.html>



Where Does Data Come From?

Data X

Where Does Data Come From?

Real-life Example: ZestCash

- "All data is credit data"



The data says: greater credit risk!

Public datasets on AWS

To enable more innovation, AWS hosts a selection of datasets that anyone can access for free. Data in our public datasets is available for rapid access to our flexible and low-cost computing resources.

Life Sciences
1000 Genomes Project

Earth Science
NASA Earth Exchange (NASA NEX)

Internet Science
Common Crawl Corpus

amazon
aws



Your Own Web Site

Public Data Sets
Stock market, etc.

IOT/Sensors

Other Web Sites



Web Scraping



```
1  from bs4 import BeautifulSoup
2  import requests
3  page_link ='https://www.website_to_crawl.com'
4  # fetch the content from url
5  page_response = requests.get(page_link, timeout=5)
6  # parse html
7  page_content = BeautifulSoup(page_response.content, "html.parser")
8
9  # extract all html elements where price is stored
10 prices = page_content.find_all(class_='main_price')
11 # prices has a form:
12 #<div class="main_price">Price: $66.68</div>,
13 # <div class="main_price">Price: $56.68</div>
14
15 # you can also access the main_price class by specifying the tag of the class
16 prices = page_content.find_all('div', attrs={'class':'main_price'})
```

<https://github.com/ikhlaqsidhu/data-x>

https://github.com/ikhlaqsidhu/data-x/tree/master/03-tools-webscraping-crawling_api_af0

Data X

Formatting Data

Data X

An ML High Level Framework

In Real Life

- Objects
- Events / Experiments
- People / Customers
- Products
- Stocks
- ...

Features, but also loss of information

The diagram illustrates the machine learning process. It starts with a box labeled "In Real Life" containing a list of various entities. An arrow points from this box to a central table labeled "Features, but also loss of information". This table contains a header row and several data rows for individuals labeled "Person 1", "Person 2", "Person 3", and "Person N". The columns represent various features like Sex, Age, Marital Status, Occupation, Job Time, Checking, Savings, and Good/Bad Mark. Arrows point from the table to two boxes: "In Sample" (containing the first few rows) and "Out of Sample" (containing the last few rows). Finally, arrows point from these boxes to a list of outcomes: "Some data has observed results" (leading to "Characteristics", "Patterns", and "Models"), and "Out of Sample" (leading to "Predictions", "Similarities", "Differences", and "Distance").

Sex	Age	Marital ...	Occupation	Job Time	Checking	Savings	GoodBad Mark
female	27.17	Married	Semi-professional	0	No	Yes	Good
male	25.92	Married	Blue Collar	0.375	No	Yes	Good
male	23.08	Married	Blue Collar	1	No	Yes	Good
male	39.58	Married	Semi-professional	0	No	Yes	Good
male	30.58	Single	Blue Collar	0.125	No	No	Good
male	17.25	Married	Blue Collar	0.04	No	No	Good
female	17.67	Single	Semi-professional	0	No	No	Bad
male	16.5	Married	Blue Collar	0.165	No	No	Good
female	27.33	Married	Semi-professional	0	No	No	Good
male	31.25	Married	Semi-professional	0	No	Yes	Good
male	20	Married	Blue Collar	0.5	No	No	Bad
male	39.5	Married	Blue Collar	1.5	No	No	Good
male	36.5	Married	Blue Collar	3.5	No	No	Good
male	52.42	Married	Blue Collar	3.75	No	No	Good

Copyright © Plug&Score

In Sample

Out of Sample

Some data has observed results

- Characteristics
- Patterns
- Models

- Predictions
- Similarities
- Differences
- Distance

Data X

An ML High Level Framework

In Real Life

- Objects
- Events / Experiments
- People / Customers
- Products
- Stocks
- ...

Features, but also loss of information

The diagram illustrates the flow of data from 'In Real Life' through a 'Table' to 'In Sample' and 'Out of Sample' datasets, and finally to a mathematical matrix representation.

In Real Life: A box containing a list of objects, events, people, products, stocks, etc.

Features, but also loss of information: An arrow points from the 'In Real Life' box to a table representing a dataset.

Table: A grid of data for multiple individuals (Person 1, Person 2, Person 3, ..., Person N). The columns represent various features like Sex, Age, Marital Status, Occupation, Job Time, Checking, Savings, and GoodBadMark. The table includes a copyright notice: "Copyright © Plug&Score".

In Sample: An arrow points from the table to the left side of the table, indicating the portion used for training or analysis.

Out of Sample: An arrow points from the table to the right side, indicating the portion used for testing or validation.

Some data has observed results: An arrow points from the table to the right, leading to a list of characteristics and patterns.

Characteristics, **Patterns**, **Models**: A list of concepts related to the observed data.

Predictions, **Similarities**, **Differences**, **Distance**: A list of concepts related to the data's applications.

Math: Matrix X , with N rows – each person m columns, each feature (age, salary, ...)

$$X = \begin{bmatrix} -2 & 4 & 7 & 31 \\ 6 & 9 & 12 & 6 \\ 12 & 11 & 0 & 1 \\ 9 & 10 & 2 & 3 \end{bmatrix}$$

CS: Table

Math: Matrix X , with N rows – each person m columns, each feature (age, salary, ...)

$X =$

$$\begin{bmatrix} -2 & 4 & 7 & 31 \\ 6 & 9 & 12 & 6 \\ 12 & 11 & 0 & 1 \\ 9 & 10 & 2 & 3 \end{bmatrix}$$



Data X

A Fundamental Idea: From Table to Score

$X =$

Cust	F1	F2	F3
A	4	2	2
B	4.5	1.5	3
C	3	3	5
D	1	2	2
E	3	1.5	5
F	3.5	3.5	1
..

$F(X)$

Cust	Credit Score
A	552
B	381
C	760
D	330
E	452
F	678
..	..



A Fundamental Idea: From Table to Score

X =

Cust	F1	F2	F3
A	4	2	2
B	4.5	1.5	3
C	3	3	5
D	1	2	2
E	3	1.5	5
F	3.5	3.5	1
..

$F(X)$

Cust	Credit Score
A	552
B	381
C	760
D	330
E	452
F	678
..	..

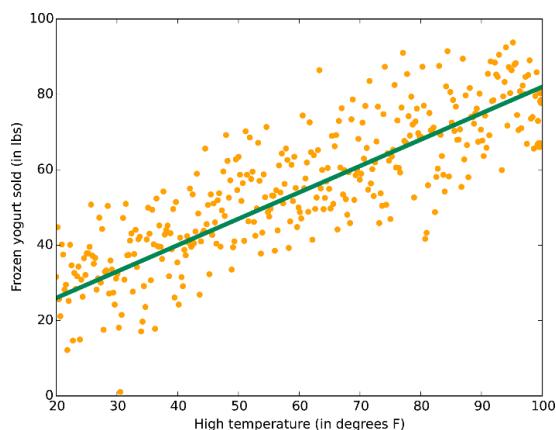
#Setting up for Supervised learning
First clean: use mapping + buckets

X = matrix of data – e.g 1000 rows
Y = In sample responses

Typically we want to split in to training data and test data

```
X_train = X[0:500]  
Y_train = Y[0:500]  
X_test = X[501:1000]  
Y_test = Y[501:1000]
```

Linear Regression Illustration



```
#Setting Linear Regression in sklearn  
from sklearn import linear_model  
  
model= linear_model.LinearRegression()  
model.fit(X_train, Y_train)  
  
Y_pred_train = model.predict(X_train)  
Y_pred_test = model.predict(X_test)  
  
# Compare Y_pred_test with Y_test for  
error.
```

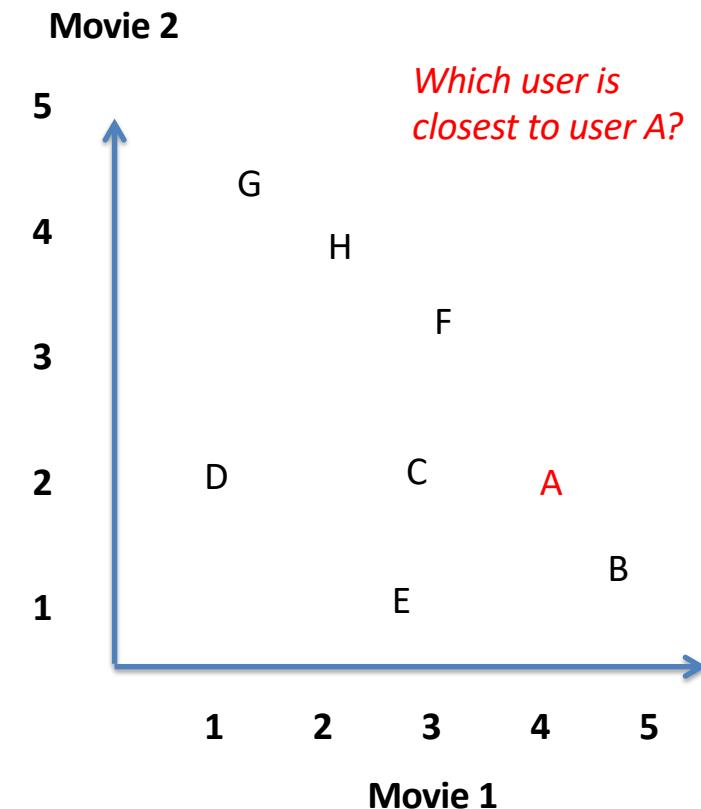
Illustration Source: <https://docs.microsoft.com/en-us/azure/machine-learning/machine-learning-algorithm-choice>



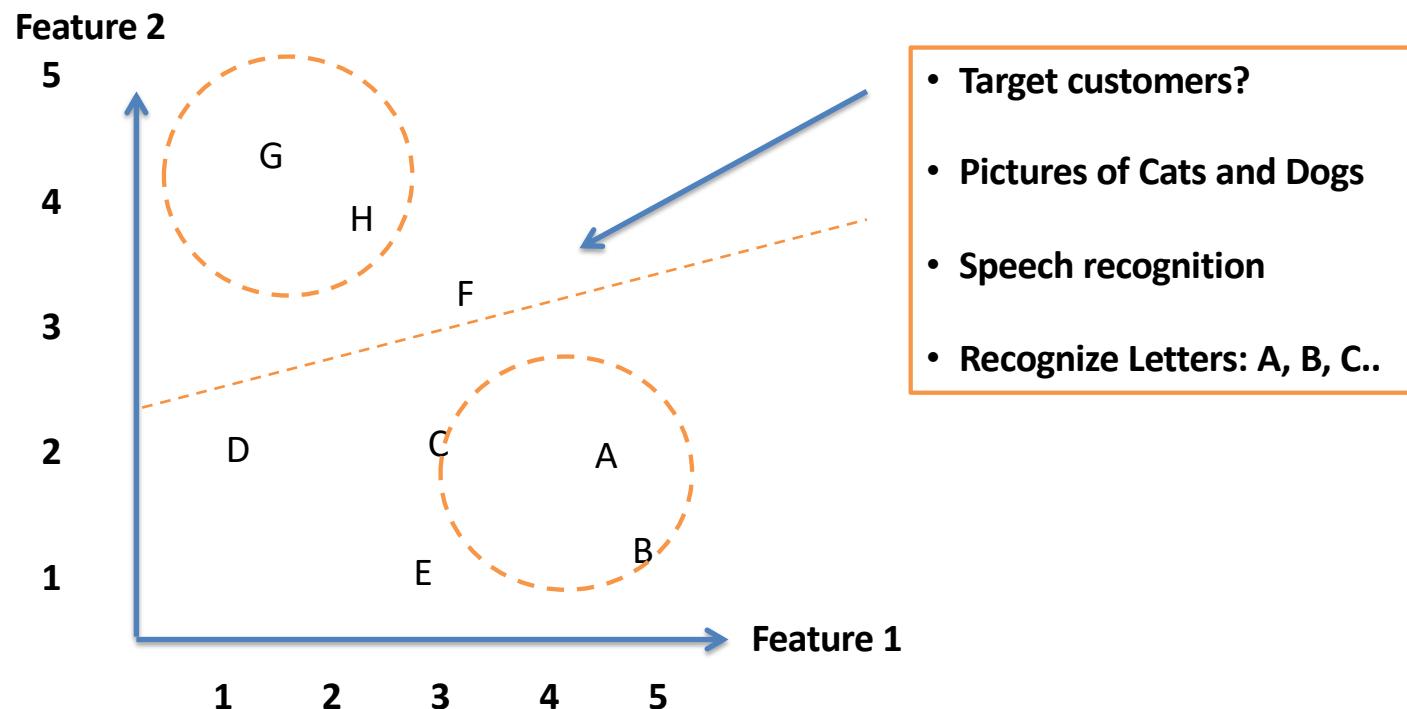
A Fundamental Idea: From Table to N- Dimensional Space

$X =$

Element	F1	F2	F3
A	4	2	2
B	4.5	1.5	3
C	3	3	5
D	1	2	2
E	3	1.5	5
F	3.5	3.5	1
..

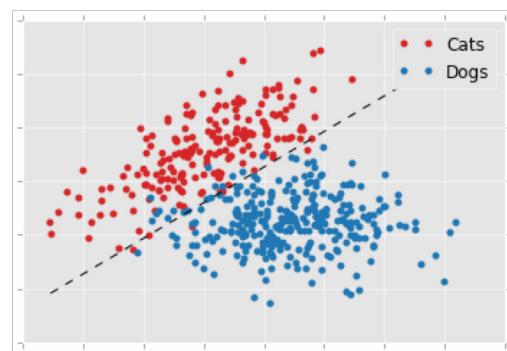


Clustering to Classification

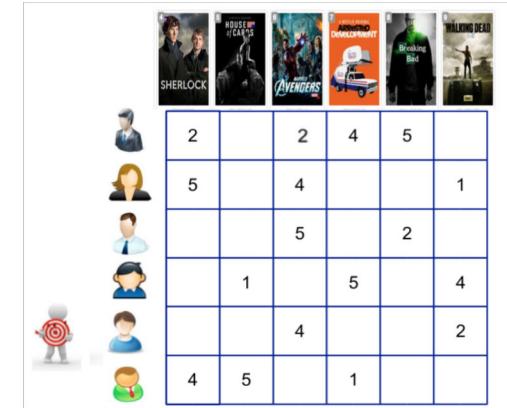


Traditionally 2 Tasks: Classification & Predictive Scoring

Extracted Data
often in
Table
Format



Classification:
Cats and Dogs, Speech Recognition
Movie Recommendation



The most famous
application has been
recommendation:
“which other user is
most like you”



Scoring:

Credit 1 0 Movie Rating 8
Heat 1 0 My Inquiry 0

Data X

We have now switched
to Neural Networks as
Function Approximators

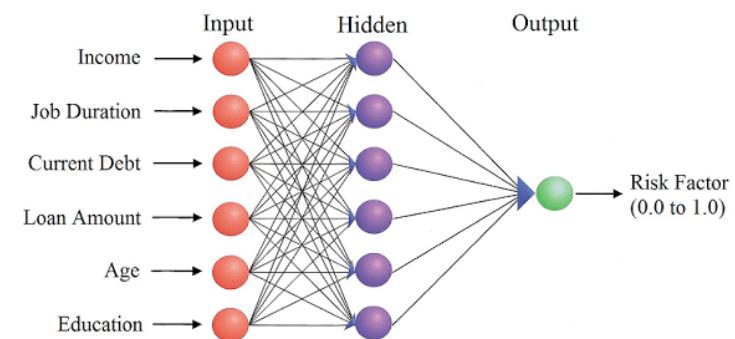
X →

ML Algorithms Guess
this function F(x)

Y →

"Non-deep" feedforward
neural network

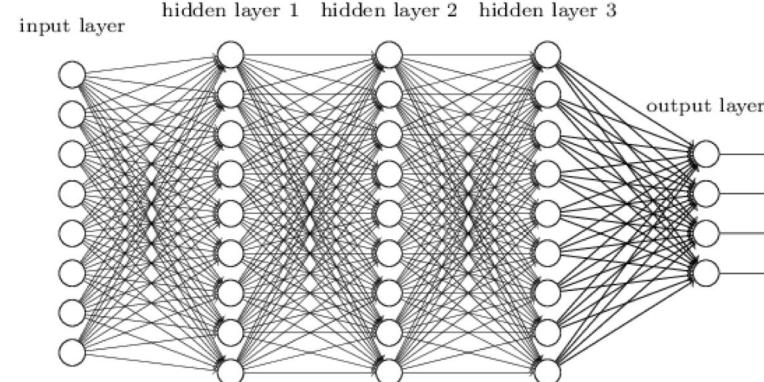
X



Y

Deep neural network

X



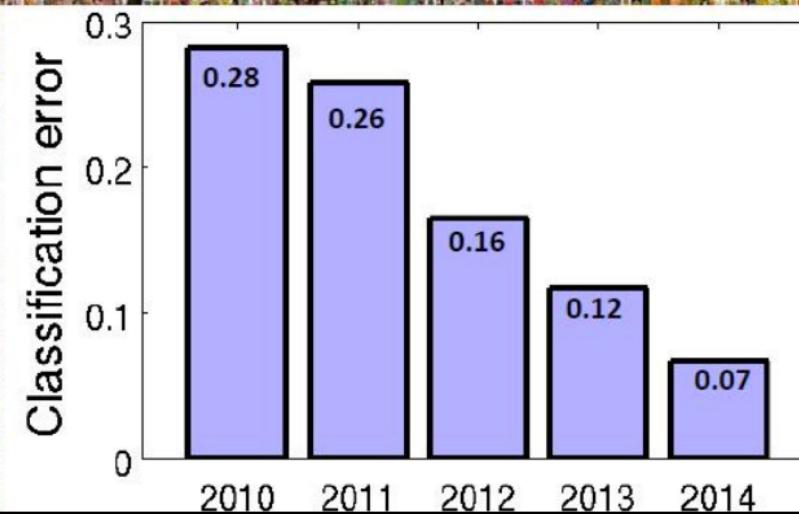
Y



IMAGENET Large Scale Visual Recognition Challenge

Street drum

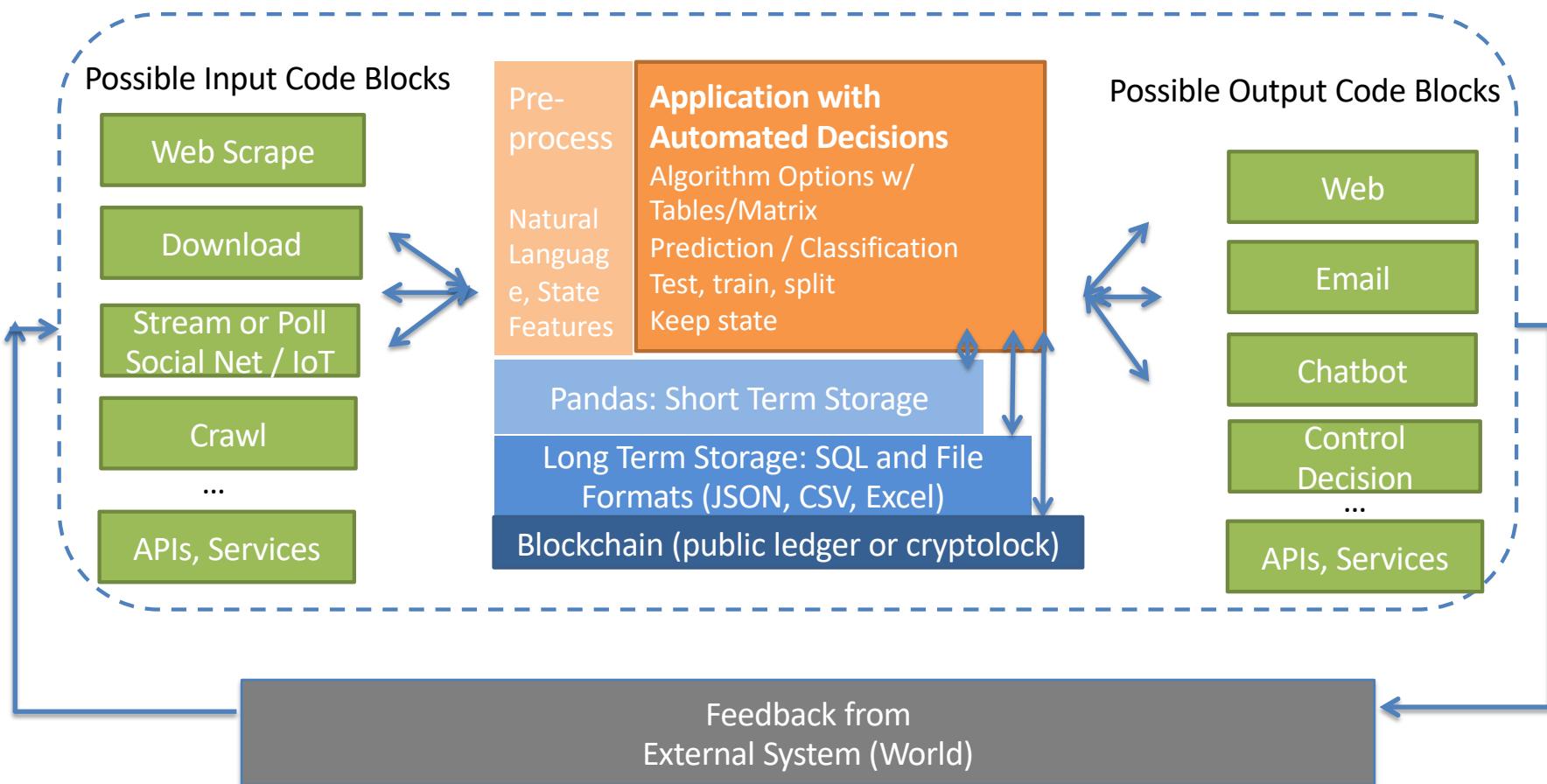
The Image Classification Challenge:
1,000 object classes
1,431,167 images



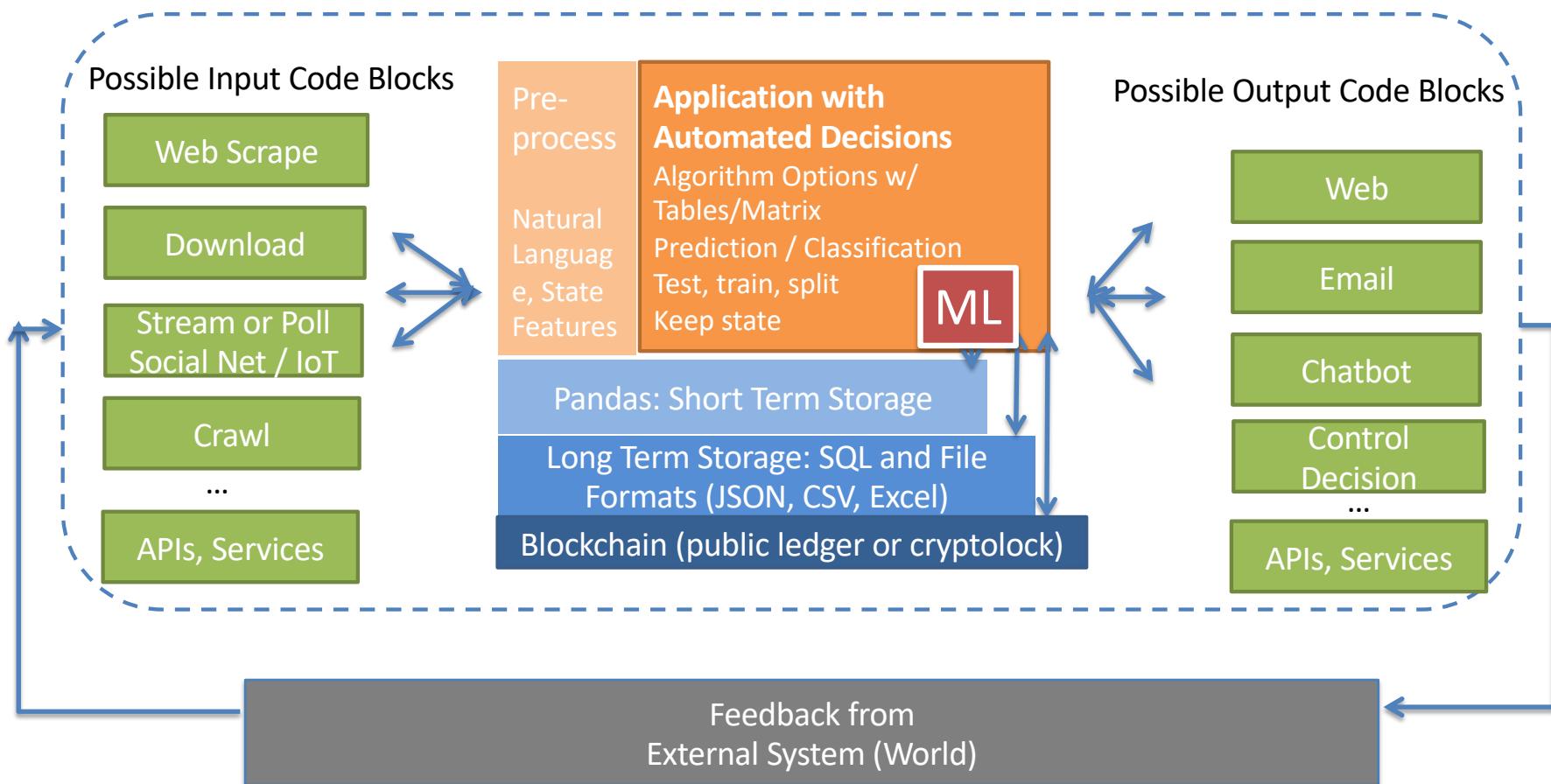
Neural net results are close to human results

Russakovsky et al. arXiv, 2014

The Data-X System View



The Data-X System View: It's more than ML, it's also systems and models



Reminder: Homework For Week 1

- HW Part 1: For Your Project – By Next week
- Come up with 3 ideas for class projects in 1-3 sentences.
- A systems or application you will build
- **Communicate:** WHO the project is for, WHAT will it do, WHY this is needed/valuable.
- ---
- Homework Part II
- Python-based review notebook (Breakout and Homework, BKHW). To sent by email.



End of Section

Data X