CSX415
Data Science Principals and Practice

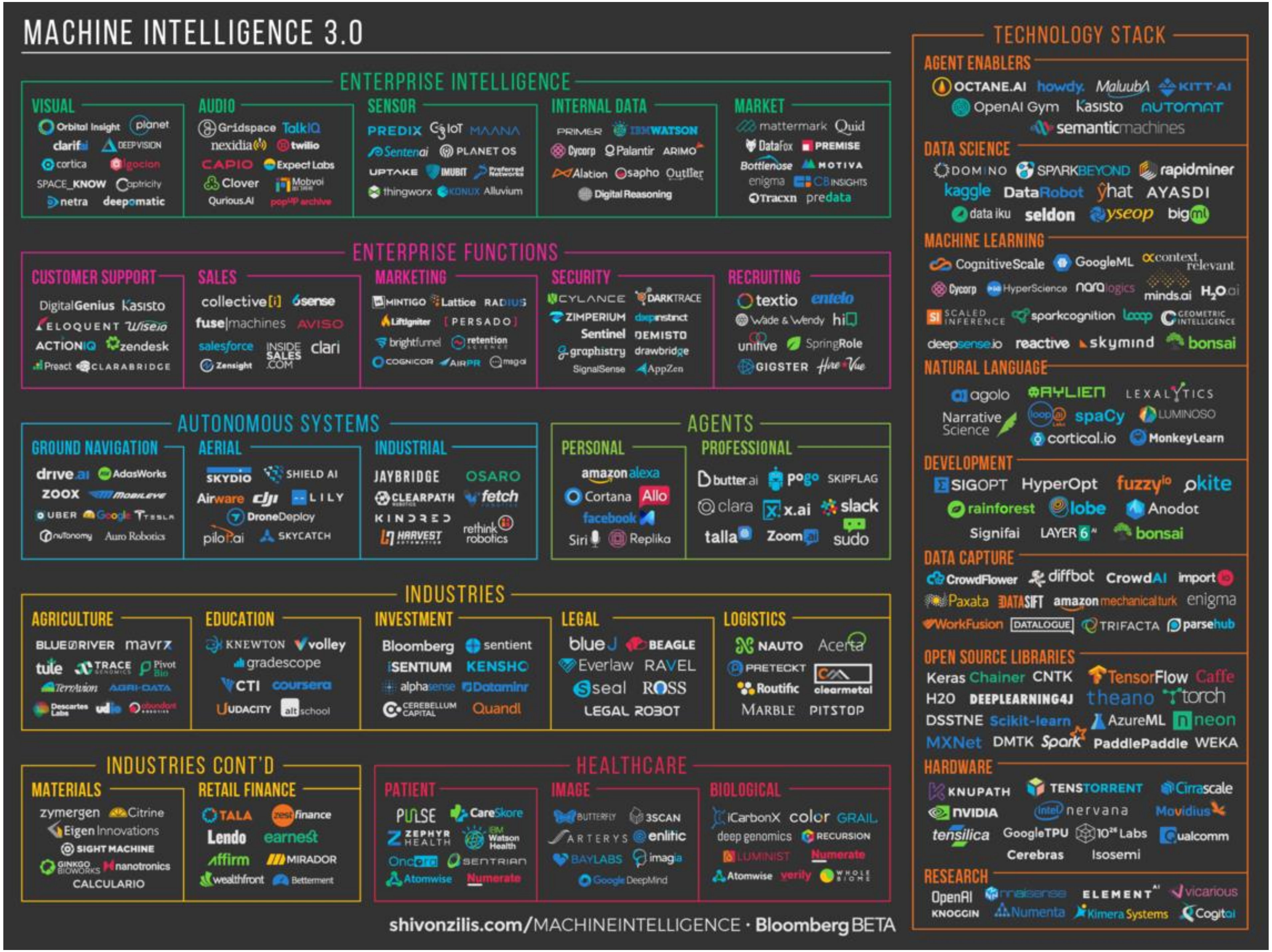
Christopher Brown
U.C. Berkeley / Decision Patterns LLC
Spring 2018

Berkeley
UNIVERSITY OF CALIFORNIA

What is Machine Learning?

Today’s Question:

What three things do all Machine Learning Algorithms have in common?



MACHINE INTELLIGENCE 3.0
ENTERPRISE INTELLIGENCE
VISUAL
Orbital Insight
planet
clarifai
DEEPVISION
cortica
gocion
SPACE_KNOW
Captricity
netra
deepomatic
AUDIO
Gridspace
TalkIQ
nexidia
twilio
CAPIO
Expect Labs
Clover
Mobvoi
Qurious.AI
popUP archive
SENSOR
PREDIX
C3 IoT
MAANA
Sentenai
PLANET OS
UPTAKE
IMUBIT
Preferred Networks
thingworx
KONUX
Alluvium
INTERNAL DATA
PRIMER
IBM WATSON
Cycorp
Palantir
ARIMO
Alation
sapho
Outlier
Digital Reasoning
MARKET
mattermark
Quid
Datafox
PREMISE
Bottlenose
MOTIVA
enigma
CB INSIGHTS
Tracxn
predata
ENTERPRISE FUNCTIONS
CUSTOMER SUPPORT
DigitalGenius
Kasisto
ELOQUENT
Wise.io
ACTIONIQ
zendesk
Preact
CLARABRIDGE
SALES
collective[i]
6sense
fuse machines
AVISO
salesforce
INSIDE SALES .COM
clari
Zensight
MARKETING
MINTIGO
Lattice
RADIUS
Liftigniter
PERSADO
brightfunnel
retention science
COGNICOR
AIRPR
msg.ai
SECURITY
CYLANCE
DARKTRACE
ZIMPERIUM
deepinstinct
Sentinel
DEMISTO
graphistry
drawbridge
SignalSense
AppZen
RECRUITING
textio
entelo
Wade & Wendy
hi
unitive
SpringRole
GIGSTER
Hire Vue
AUTONOMOUS SYSTEMS
GROUND NAVIGATION
drive.ai
AdasWorks
ZOOX
MOBILEYE
UBER
Google
TESLA
nuTonomy
Auro Robotics
AERIAL
SKYDIO
SHIELD AI
Airware
dji
LILY
DroneDeploy
pilot.ai
SKYCATCH
INDUSTRIAL
JAYBRIDGE
OSARO
CLEARPATH
fetch
KINDRED
HARVEST
rethink robotics
AGENTS
PERSONAL
amazon alexa
Cortana
Allo
facebook
Siri
Replika
PROFESSIONAL
butter.ai
pogo
SKIPFLAG
clara
x.ai
slack
talla
Zoom.ai
sudo
INDUSTRIES
AGRICULTURE
BLUE RIVER
mavrx
tule
TRACE
Pivot Bio
TerrAvion
AGRI-DATA
Descartes Labs
udio
abundant
EDUCATION
KNEWTON
volley
gradescope
CTI
coursera
UDACITY
altschool
INVESTMENT
Bloomberg
sentient
iSENTIUM
KENSHO
alphasense
Dataminr
CEREBELLUM CAPITAL
Quandl
LEGAL
blue J
BEAGLE
Everlaw
RAVEL
seal
ROSS
LEGAL ROBOT
LOGISTICS
NAUTO
Acerta
PRETECKT
clearmetal
Routific
MARBLE
PITSTOP
INDUSTRIES CONT'D
MATERIALS
zymergen
Citrine
Eigen Innovations
SIGHT MACHINE
GINKGO BIOWORKS
nanotronics
CALCULARIO
RETAIL FINANCE
TALA
zest finance
Lendo
earnest
Affirm
MIRADOR
wealthfront
Betterment
HEALTHCARE
PATIENT
PULSE
CareSkore
ZEPHYR HEALTH
IBM Watson Health
Oncora
SENTRIAN
Atomwise
Numerate
IMAGE
BUTTERFLY
3SCAN
ARTERYS
enlitic
BAYLABS
imagia
Google DeepMind
BIOLOGICAL
iCarbonX
color
GRAIL
deep genomics
RECURSION
LUMINIST
Numerate
Atomwise
verily
WHOLE BIOME
shivonzilis.com/MACHINEINTELLIGENCE · Bloomberg BETA
TECHNOLOGY STACK
AGENT ENABLERS
OCTANE.AI
howdy.
Maluuba
KITT.AI
OpenAI Gym
Kasisto
AUTOMAT
semanticmachines
DATA SCIENCE
DOMINO
SPARKBEYOND
rapidminer
kaggle
DataRobot
yhat
AYASDI
data iku
seldon
yseop
bigml
MACHINE LEARNING
CognitiveScale
GoogleML
context relevant
Cycorp
HyperScience
nara logics
minds.ai
H2O.ai
SCALED INFERENCE
sparkcognition
loop
GEOMETRIC INTELLIGENCE
deepsense.io
reactive
skymind
bonsai
NATURAL LANGUAGE
agolo
AYLIEN
LEXALYTICS
Narrative Science
loop ai Labs
spaCy
LUMINOSO
cortical.io
MonkeyLearn
DEVELOPMENT
SIGOPT
HyperOpt
fuzzy.io
kite
rainforest
lobe
Anodot
Signifai
LAYER 6
bonsai
DATA CAPTURE
CrowdFlower
diffbot
CrowdAI
import.io
Paxata
DATASIFT
amazon mechanical turk
enigma
WorkFusion
DATALOGUE
TRIFACTA
parsehub
OPEN SOURCE LIBRARIES
Keras
Chainer
CNTK
TensorFlow
Caffe
H2O
DEEPLEARNING4J
theano
torch
DSSTNE
Scikit-learn
AzureML
neon
MXNet
DMTK
Spark
PaddlePaddle
WEKA
HARDWARE
KNUPATH
TENSTORRENT
Cirrascale
NVIDIA
intel nervana
Movidius
tensilica
GoogleTPU
10^{26} Labs
Qualcomm
Cerebras
Isosemi
RESEARCH
OpenAI
maluuba
ELEMENT AI
vicarious
KNOGGIN
Numenta
Kimera Systems
Cogitai

What is **machine learning?**

A formal ***process*** for building a ***model***

Other names for ML:

Machine Learning
Artificial Intelligence
Statistical Learning
Pattern Recognition
Data Mining
Predictive Analytics
Knowledge Discovery
Predictive Modeling
Model Induction
…

What is a **model?**

a ***function (f)***
… that *estimates* a ***response*** $(\hat{\boldsymbol{y}})$
… associated with (a set of) known ***predictors*** $(\boldsymbol{x})$

What is a function?

$$\hat{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n)$$

$$\hat{\boldsymbol{y}} = \boldsymbol{f}(\vec{\boldsymbol{x}})$$

$$\vec{\boldsymbol{x}} \xrightarrow{f} \hat{\boldsymbol{y}}$$

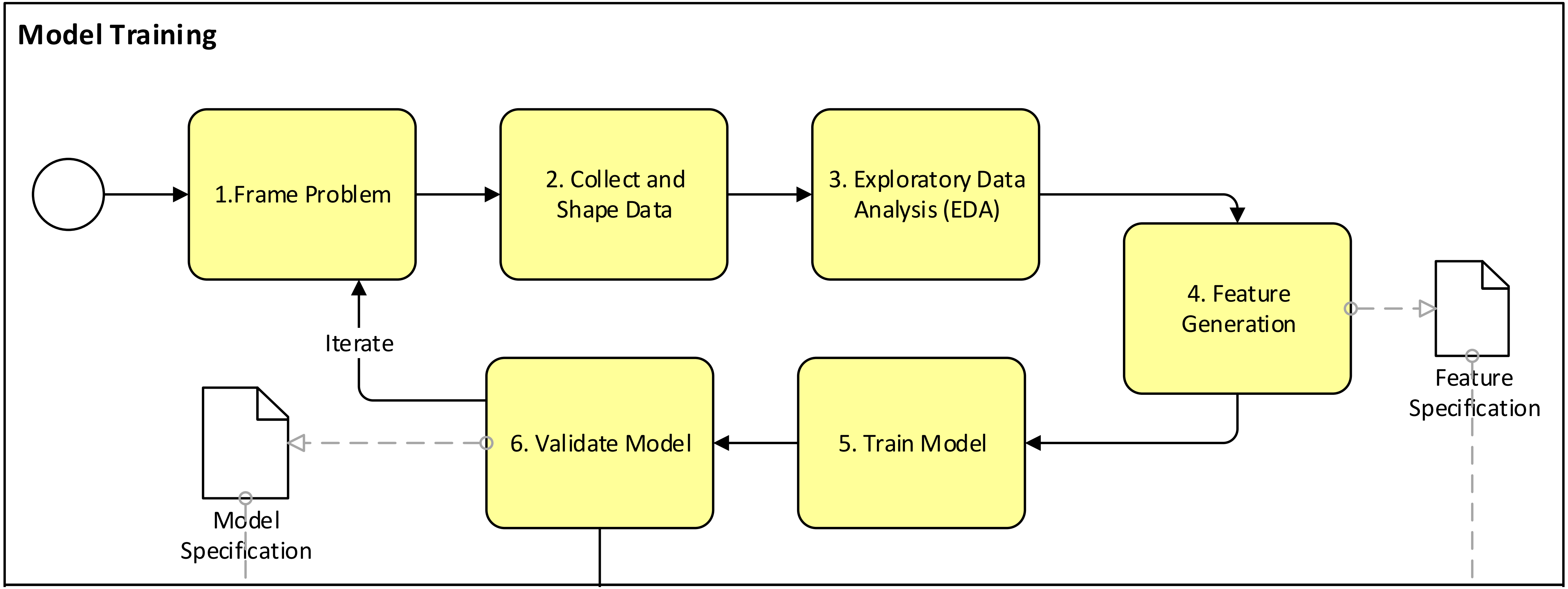
"maps"

Independent variables, covariates predictors, attribute, descriptor, **feature**
…

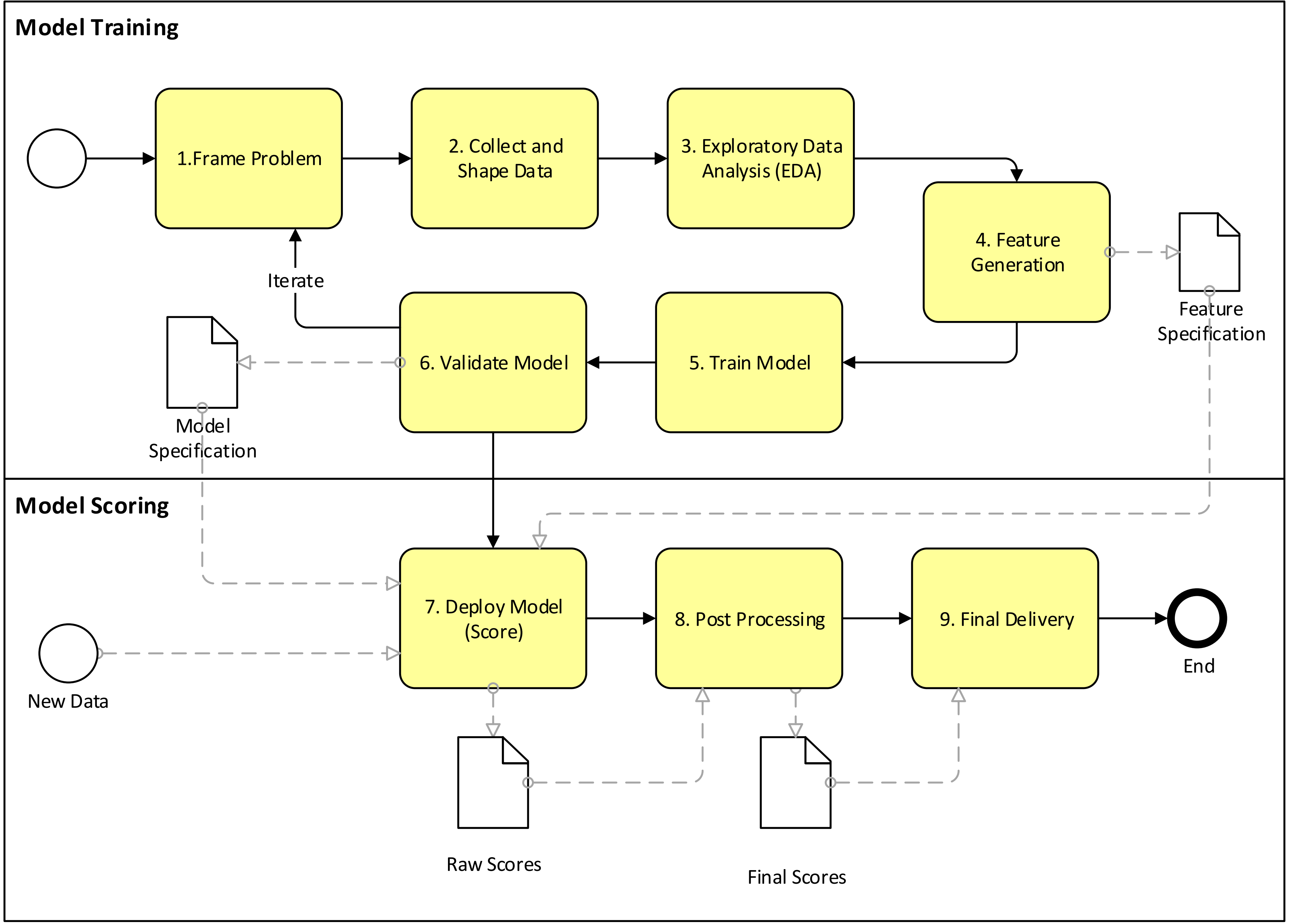
Prediction,
Forecast,
Estimate,
…

How do we find $\boldsymbol{f}$?

Model Training: Finding f



How do we use $\boldsymbol{f}$?


Model Training
1.Frame Problem
2. Collect and Shape Data
3. Exploratory Data Analysis (EDA)
4. Feature Generation
Feature Specification
5. Train Model
6. Validate Model
Iterate
Model Specification
Model Scoring
7. Deploy Model (Score)
8. Post Processing
9. Final Delivery
End
New Data
Raw Scores
Final Scores

There are **two major** ways to distinguish ML problems ... both are determined by $\boldsymbol{y}$

i.e. by what we are trying to *learn*

1. Based on availability of y

Are there previous/historical observations to learn from?

Yes → **SUPERVISED LEARNING**

No → **UNSUPERVISED LEARNING**

Not Necessarily Binary …

There are *special (edge?)* cases

SPECIAL CASE 1

Only some $\boldsymbol{y}$'s are known

- and/or -

$\boldsymbol{y}$'s are not directly known inferable

SEMI-SUPERVISED Learning

SPECIAL CASE 2

$\boldsymbol{y}$ ‘s change during training/scoring

-and/or-

$\boldsymbol{y}$ ‘s become available during training/scoring

ADAPTIVE REINFORCEMENT Learning

*** Less commonly, more frequently “adversary learning”**

2. Based on the type of y

What values can y assume ?

Continuous → **Regression**

(predict an count or amount)

Categorical* → **Classification**

(predict a class or category)

*Binary classification is an important special case

Not Necessarily Binary …

There are *special (edge?)* cases

SPECIAL CASE 1

ORDINAL RESPONSE

Use Either Regression or Classification

SPECIAL CASE 2

Date

Use Either Regression or Classification

-or-

Special Techniques

(forecast|survival)

$$y$$

Known

Dependent variable,

Target (variable),

Outcome, Response,

Class (classification)

Data Uses

Dependent variable, Target (variable), Outcome, **Response**, **Class (classification)**

Independent variables, covariates predictors, attribute, descriptor, **feature**
…

Unit of observation, Cases, Instance, Data Point, Sample

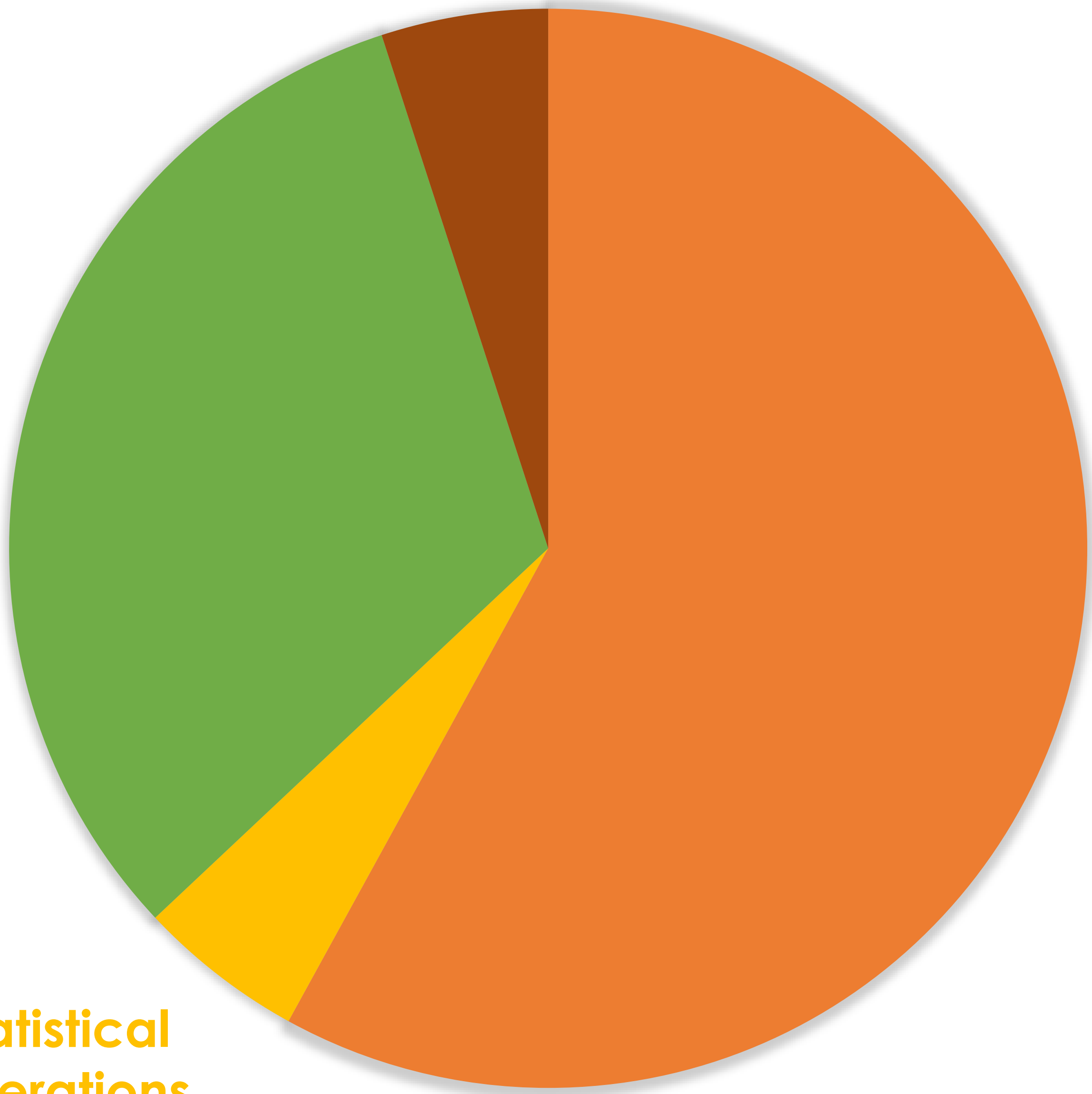
Y	X_1	X_2	X_3	… X_n

```
data, data, data, data, data, data, data, data
data, data, data, data, data, data, data, data
data, data, data, data, data, data, data, data
data, data, data, data, data, data, data, data
data, data, data, data, data, data, data, data
data, data, data, data, data, data, data, data
data, data, data, data, data, data, data, data
data, data, data, data, data, data, data, data
data, data, data, data, data, data, data, data
```

A major limitation of ML is:

(nearly) every ML algorithm expects data in a tabular form.

Task Breakdown



Now what about … f?

How do we find f ?

Well what properties should f have?

Desirable Properties of $\boldsymbol{f}$?

- Takes a one or more inputs
- Yields a single output value for each input
- Should be easy* to evaluate
- Outputs, $\hat{\boldsymbol{y}}$, should be “close to” observed values, $\mathbf{y}$:

$$\hat{\boldsymbol{y}} \sim \boldsymbol{y}$$

* Computational cheap/efficient

What do we mean by “Close to”?

qualitative measure of “close to”?

…

$$\boldsymbol{f}(\widehat{\boldsymbol{y}}\,,\boldsymbol{y})$$

$$\mathcal{L}(\widehat{\boldsymbol{y}}\,,\boldsymbol{y})$$

How do we calculate $\mathcal{L}(\widehat{\boldsymbol{y}}\,,\boldsymbol{y})$?

Depends on whether we are doing regression or classification

qualitative measure of "close to"?

…

Depends on whether we are doing regression or classification

Regression

…

$$\mathcal{L}(\widehat{\boldsymbol{y}}, \boldsymbol{y}) = \boldsymbol{y} - \widehat{\boldsymbol{y}}$$

$$(\boldsymbol{y} - \widehat{\boldsymbol{y}}) = 0$$

That's just one observation

…

We need to evaluate

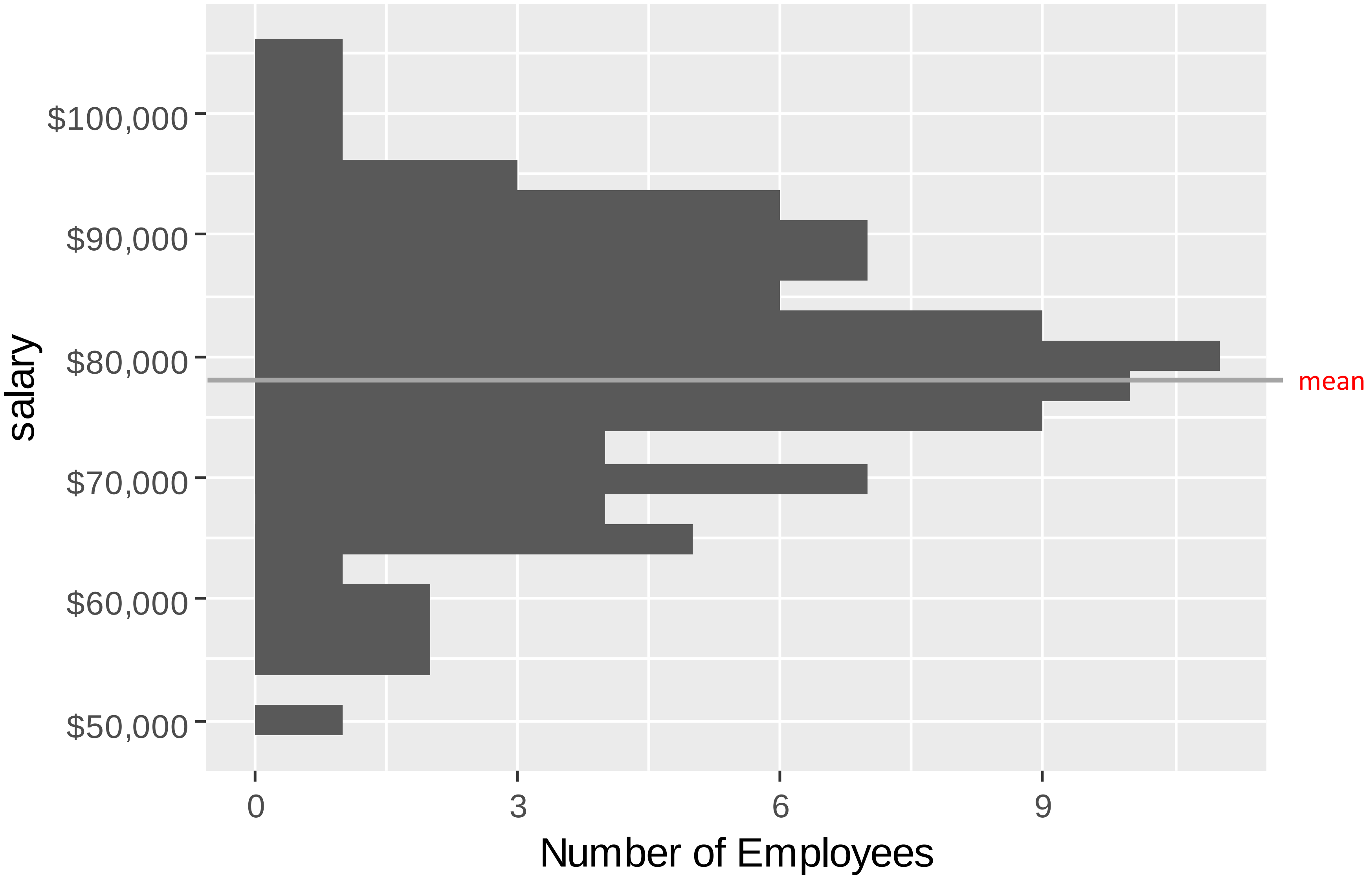
$$\mathcal{L}(\widehat{\boldsymbol{y}}\,,\boldsymbol{y}) = \boldsymbol{y} - \widehat{\boldsymbol{y}}$$

for all pairs

And arrive at a single value,
we need:

$$\boldsymbol{L}\big(\mathcal{L}(\widehat{\boldsymbol{y}}\,,\boldsymbol{y})\big) = (\boldsymbol{L}\ \boldsymbol{o}\ \mathcal{L})(\widehat{\boldsymbol{y}}\,,\boldsymbol{y})$$

Example



salary
$100,000
$90,000
$80,000
$70,000
$60,000
$50,000
0
3
6
9
Number of Employees
mean

Our Model

“Naïve” model

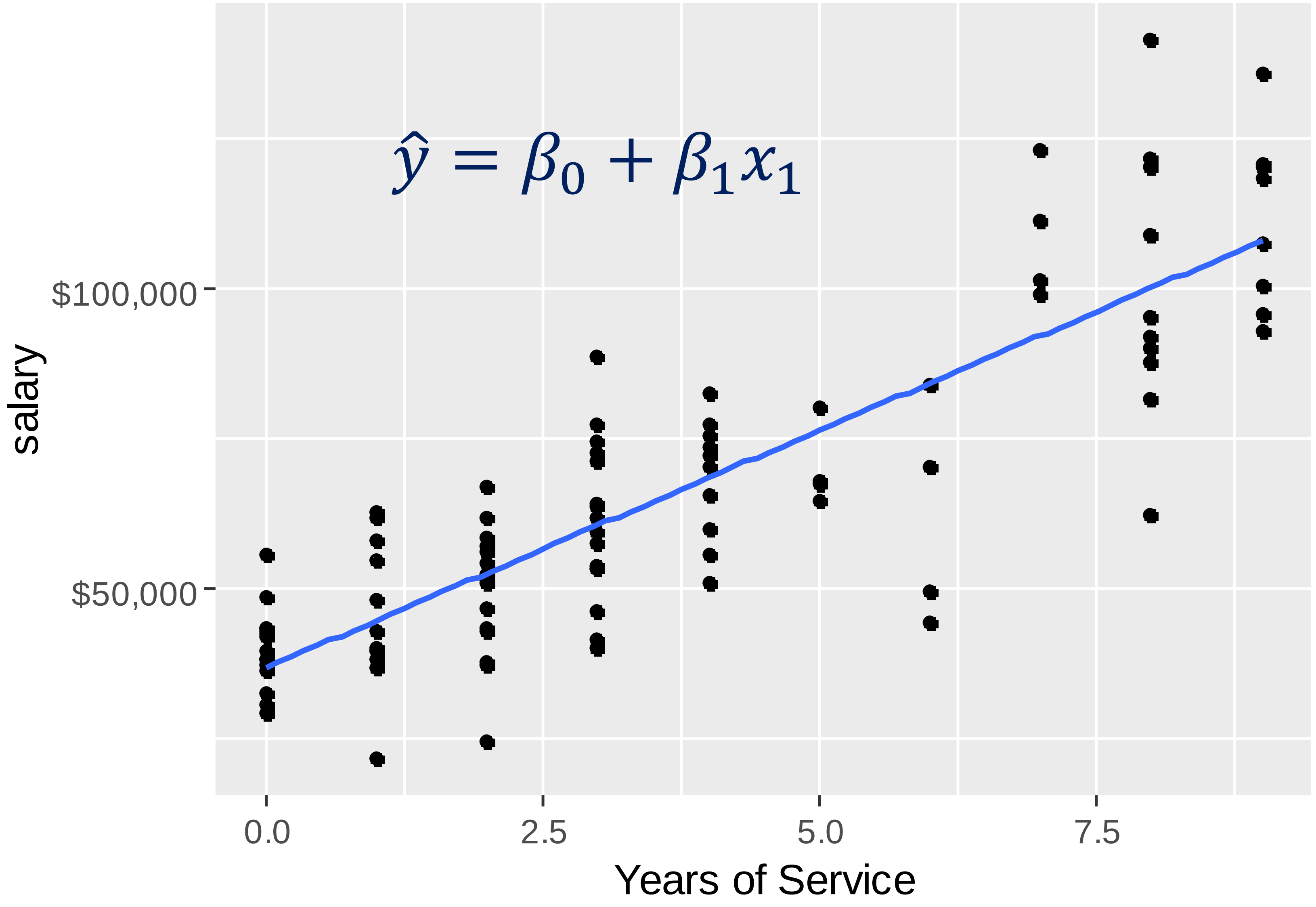
$$\hat{y} = mean(y)$$

Linear functions (of one variable)

$$\hat{y} = b + mx$$

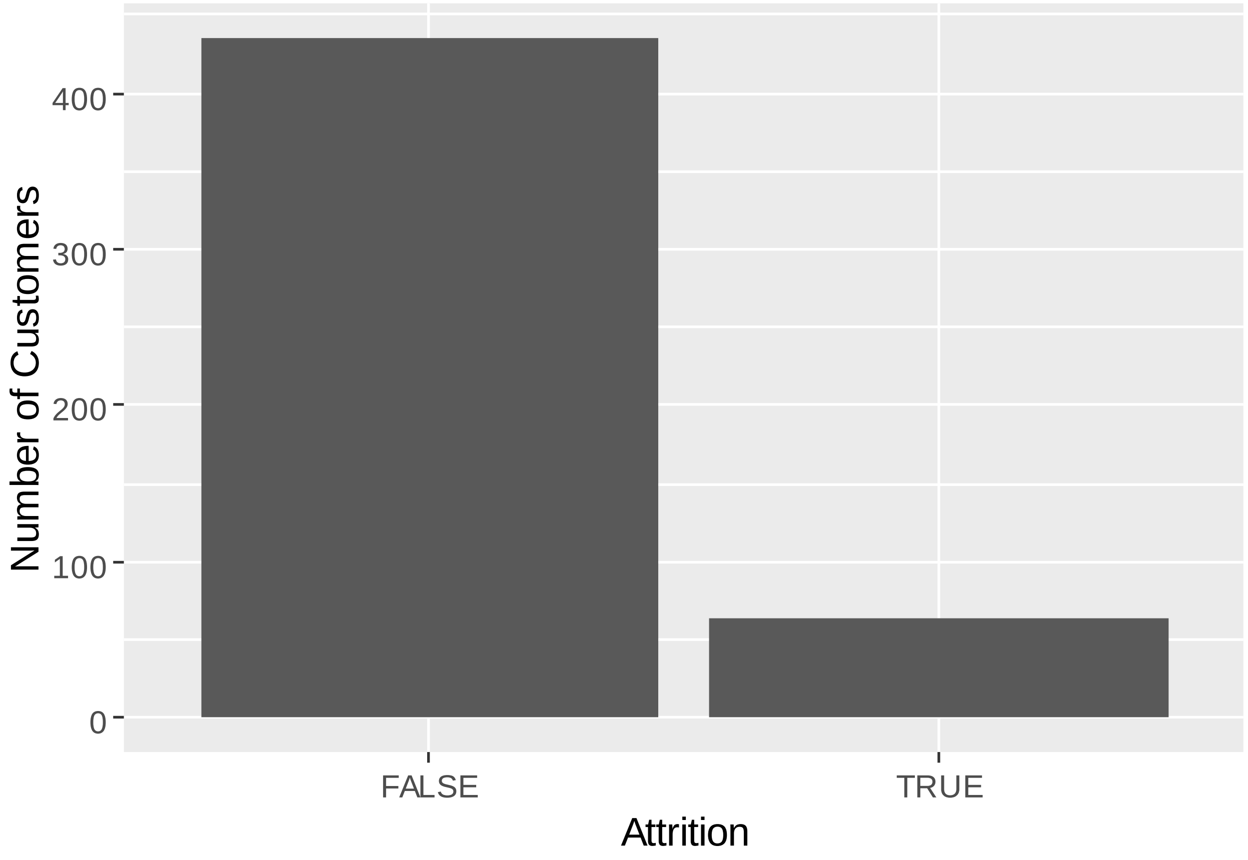
$$\hat{y} = \beta_0 + \beta_1 x_1$$

Example



$\hat{y} = \beta_0 + \beta_1 x_1$
$100,000
$50,000
salary
0.0
2.5
5.0
7.5
Years of Service

Example



Classification

…

$$\mathcal{L}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \left\{\begin{matrix} \mathbf{0} \mid \boldsymbol{y} = \hat{\boldsymbol{y}} \\ \mathbf{1} \mid \boldsymbol{y} \neq \hat{\boldsymbol{y}} \end{matrix}\right\}$$

$$\boldsymbol{L}\left(\mathcal{L}(\hat{\boldsymbol{y}}, \boldsymbol{y})\right) = (\boldsymbol{L}\ \boldsymbol{o}\ \mathcal{L})(\hat{\boldsymbol{y}}, \boldsymbol{y})$$

What functions $\boldsymbol{f}$ can be used?

$$\infty$$

Search / Optimization

Find the **parameters** (β) that minimize that minimize the loss function …

SOLVE:

$$\hat{y} = \beta_0 + \beta_1 x_1$$

$$argmin_{\beta}\, L(\boldsymbol{y}, \widehat{\boldsymbol{y}})$$

$$argmin_{\beta} \sum (\boldsymbol{y} - \widehat{\boldsymbol{y}})^2$$ (SSE)

Solution Methods

- Direct Solution (special case)
- Numerical optimization; recursive goal seeking

3 Requirement for ML Algorithm

- A method for evaluating how well the algorithm performs (**ERRORS**)
- A restricted class of functions (**MODEL**)
- A process for proceeding through the restricted class of functions to identify the functions (**SEARCH/OPTIMIZATION**)

How to understand Algorithms

1. Errors

2. Model

3. Search Optimization

* Strengths / Limitations

Theory

Frame problems to make the suitable for solution via machine learning

Distinguish fundamental aspects of machine learning algorithms →**know** what algorithms are appropriate for which problems

Measures/evaluate model performance

Know how to **improve** a model **and** determine when the model is good enough

Execution

is more than building/training models:

Deploying machine learning models to operations

Generating high quality, graphical and textual results regarding model behavior

Collaborating in a group using tools for collaborative/social programming