Intro to ML

March 12, 2019
Data Science CSCI 1951A
Brown University
Instructor: Ellie Pavlick
HTAs: Wennie Zhang, Maulik Dang, Gurnaaz Kaur

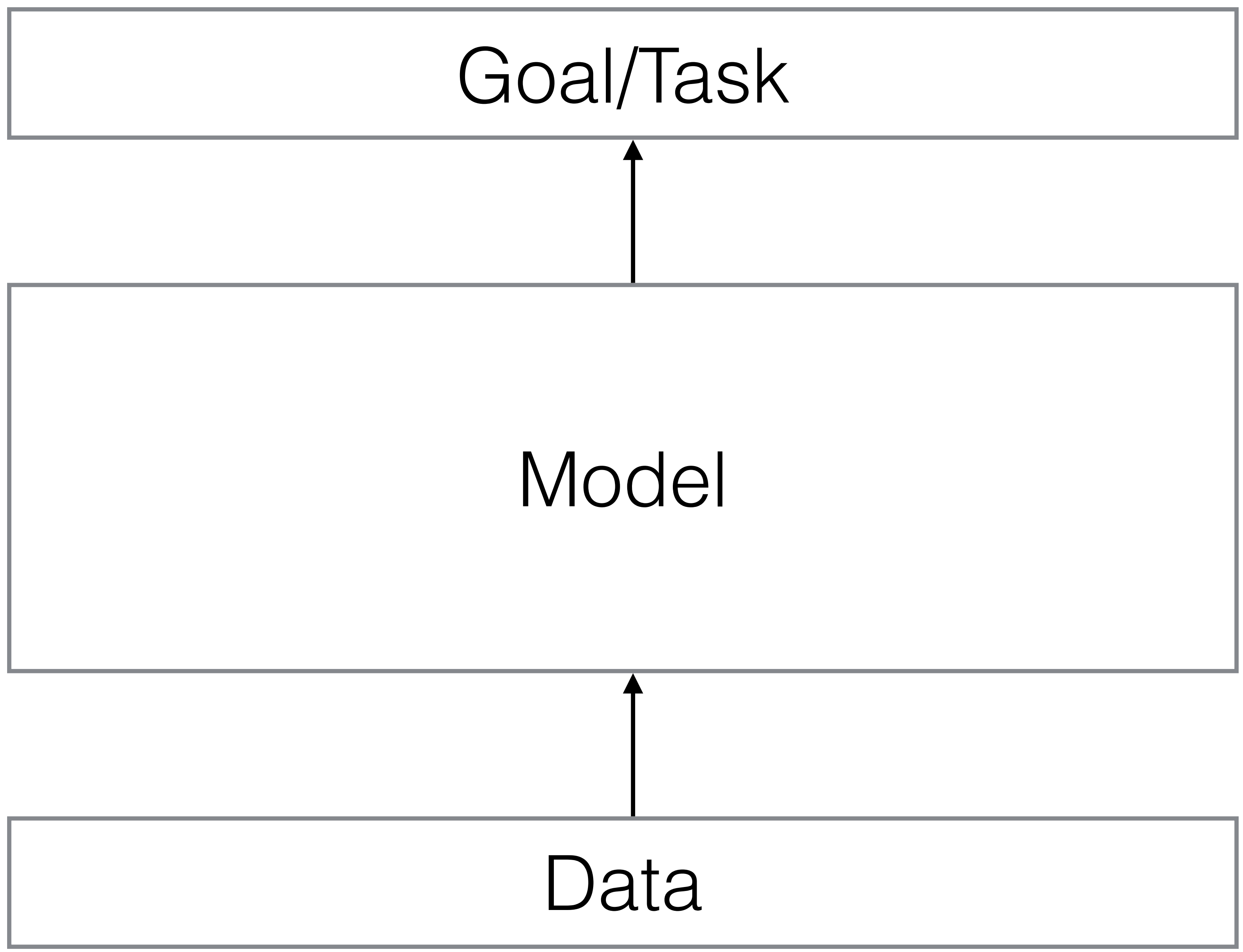
Announcements

- Extensions….try try try try to use late days! That is what they are for!
- Summer Research plug: https://www.clsp.jhu.edu/workshops/18-workshop/undergrads/
- Questions from the audience?

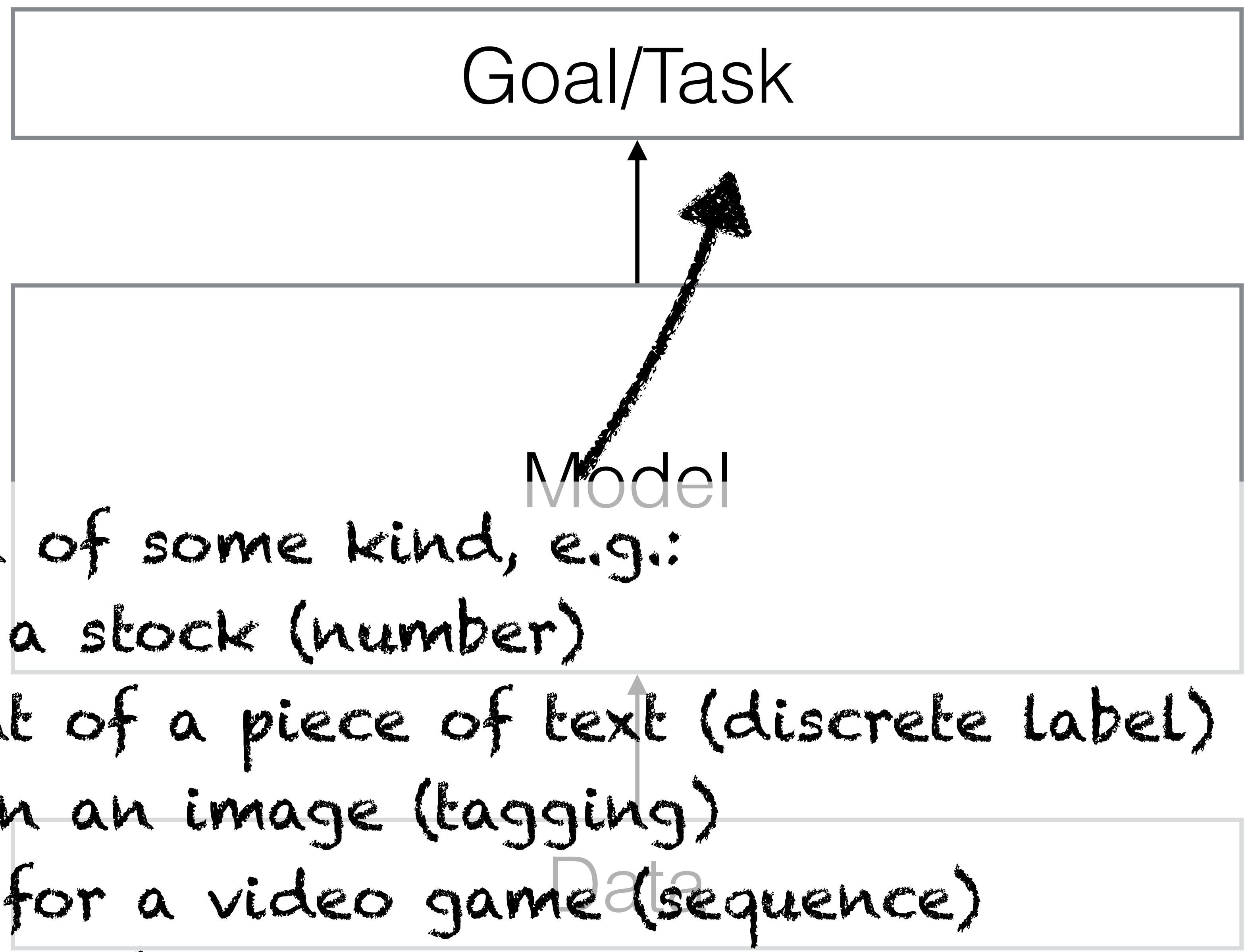
Today

- ML “preliminaries”—terminology, basic building blocks, conceptual background
- Linear Regression with Stochastic Gradient Descent

Oversimplified ML



Oversimplified ML



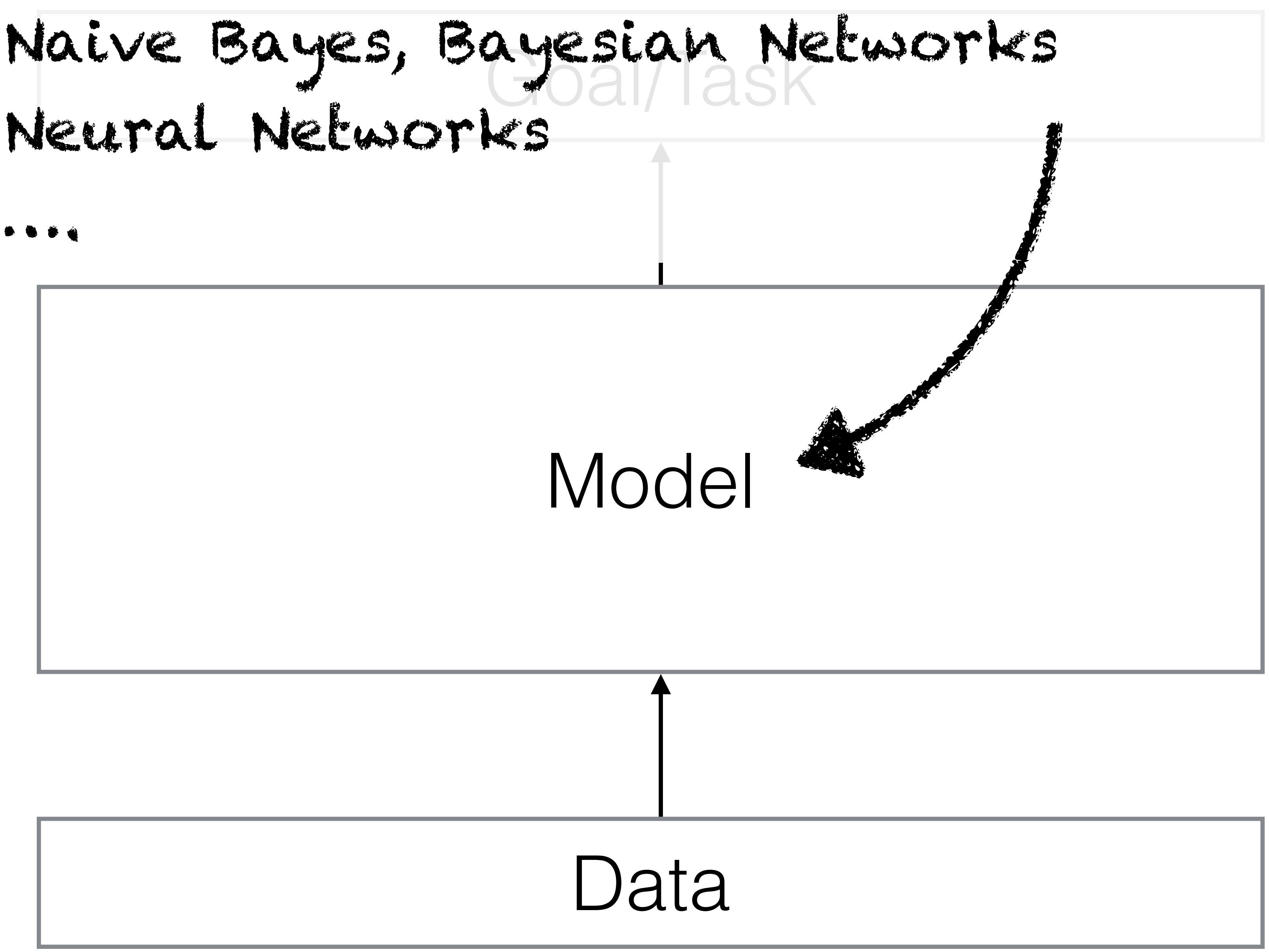
Prediction of some kind, e.g.:

- price of a stock (number)
- sentiment of a piece of text (discrete label)
- objects in an image (tagging)
- strategy for a video game (sequence)
- parse tree of a sentence (tree structure)

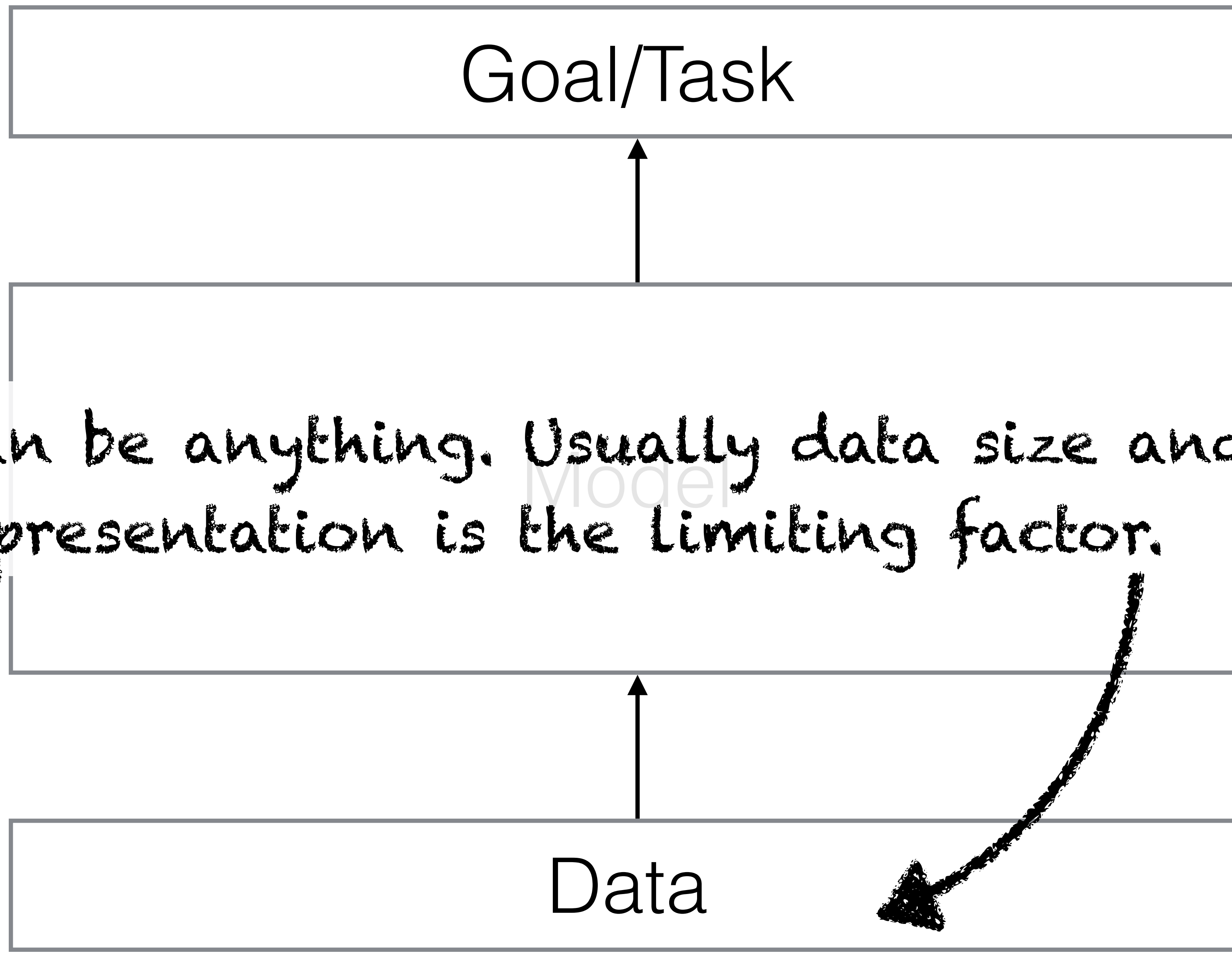
Oversimplified ML

Decisions about how the problem is structured AND how to estimate parameters

- linear/logistic regression
- SVMs
- Naive Bayes, Bayesian Networks
- Neural Networks
- ....

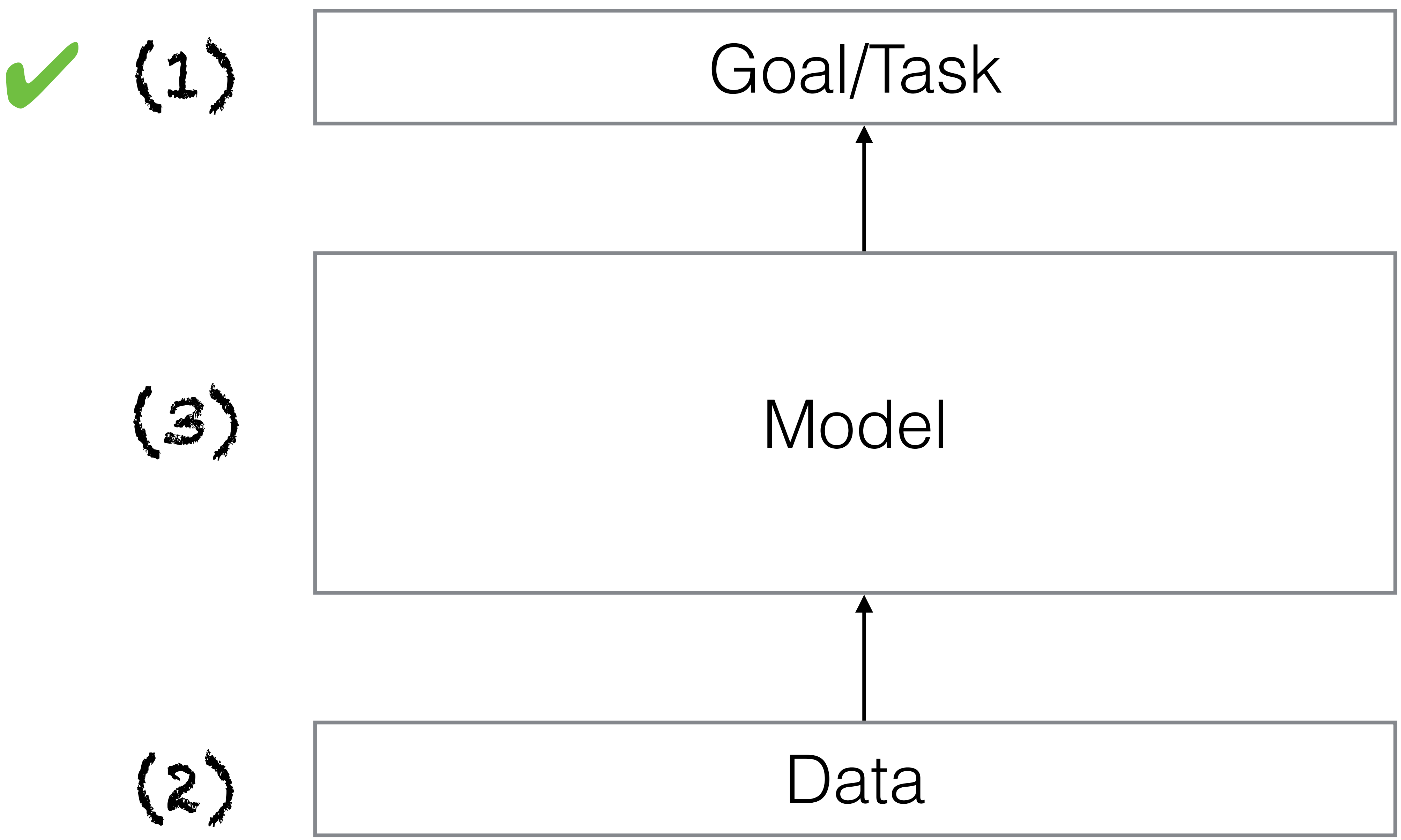


Oversimplified ML

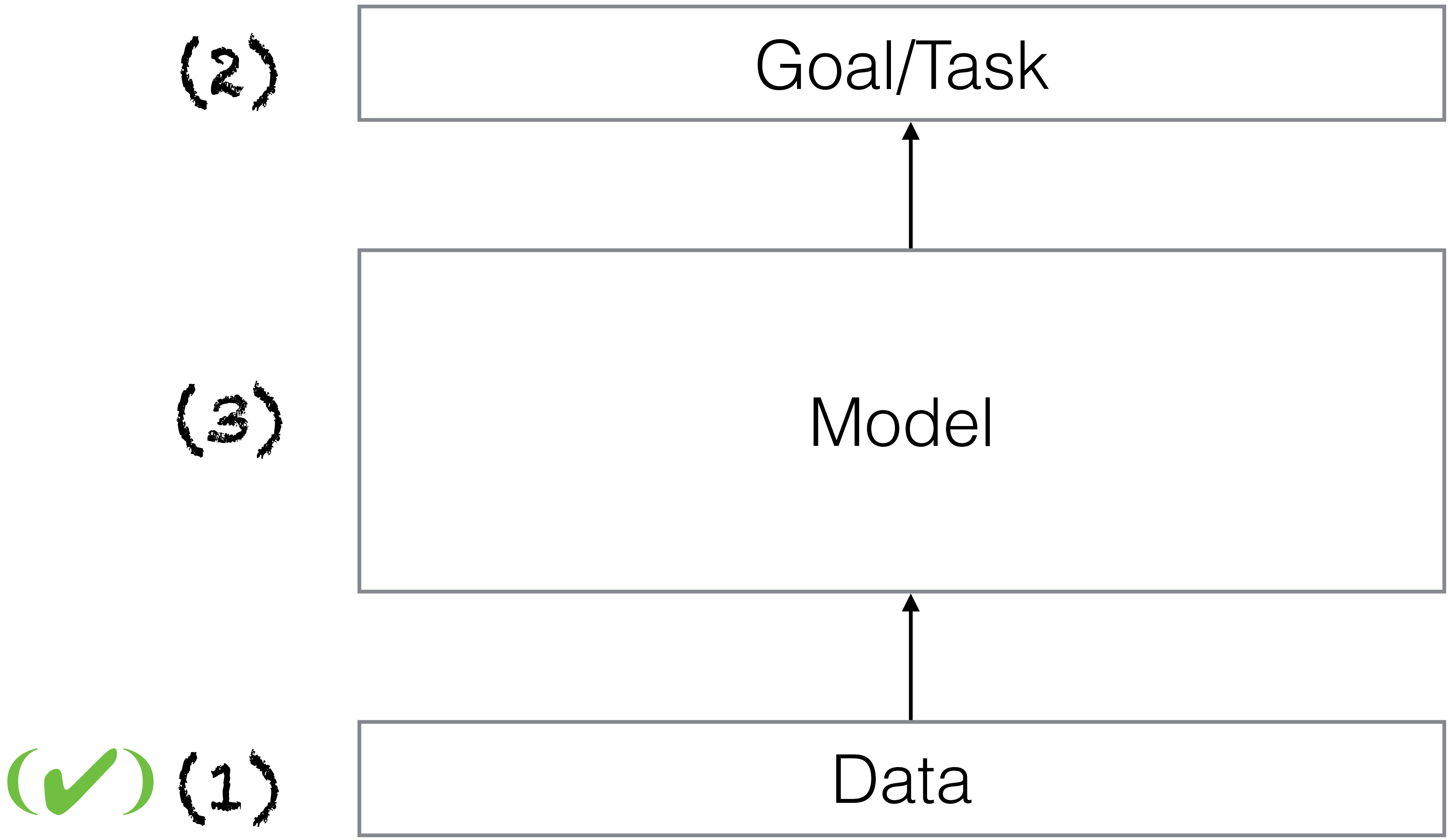


Can be anything. Usually data size and/or representation is the limiting factor.

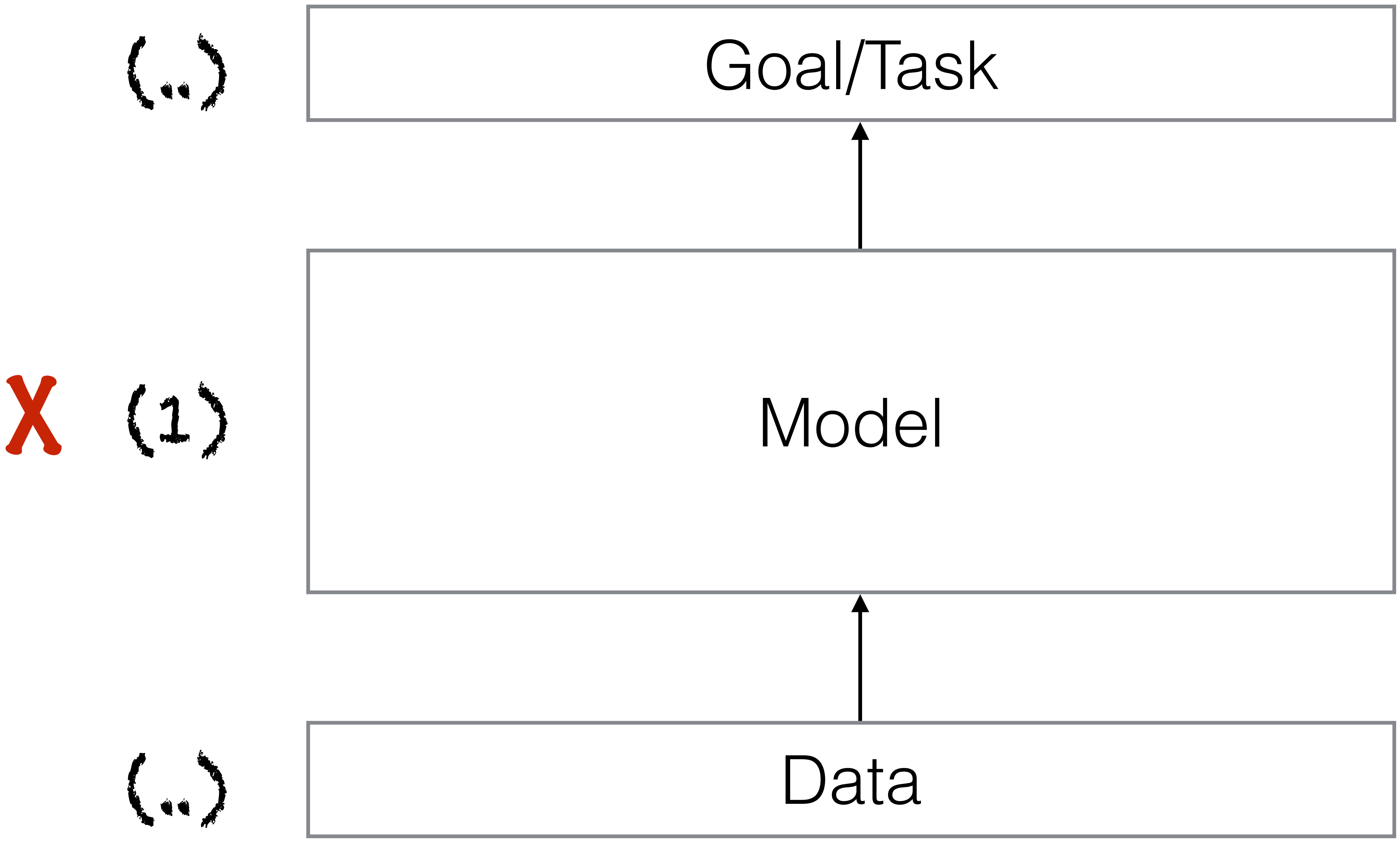
Oversimplified ML



Oversimplified ML



Oversimplified ML



Defining an ML problem

- What is “machine learnable”?

Defining an ML problem

- What is “machine learnable”?
- ~~Like…basically everything, right?~~

Defining an ML problem

- What is “machine learnable”?
- ~~Like…basically everything, right?~~ WRONG!!

Defining an ML problem

- What is “machine learnable”?
- ~~Like…basically everything, right?~~ WRONG!! (kind of)

Defining an ML problem

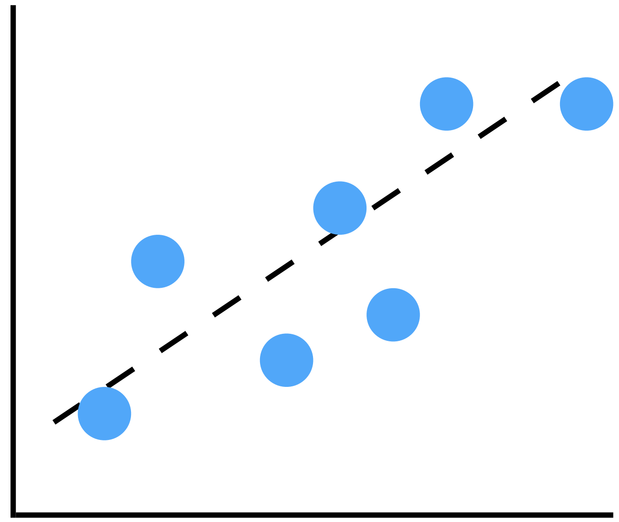
- What is “machine learnable”?
- ~~Like…basically everything, right?~~ WRONG!! (kind of)
- Input features need to be concrete and representable. Definition of “success” needs to be quantifiable (and, right now, usually differentiable).

Defining an ML problem

ML = Function Approximation

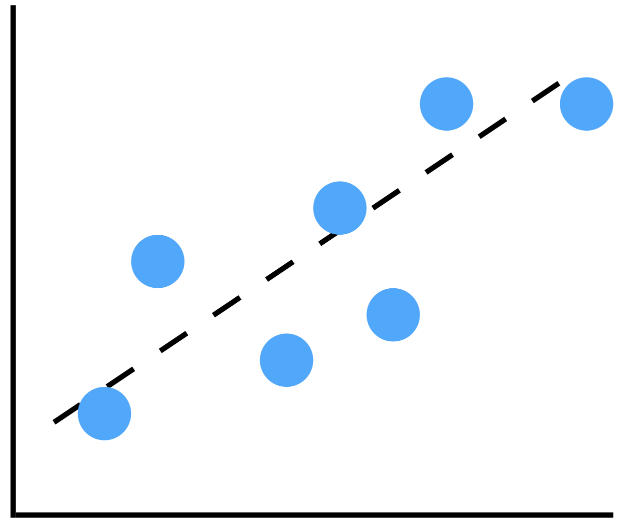
Defining an ML problem

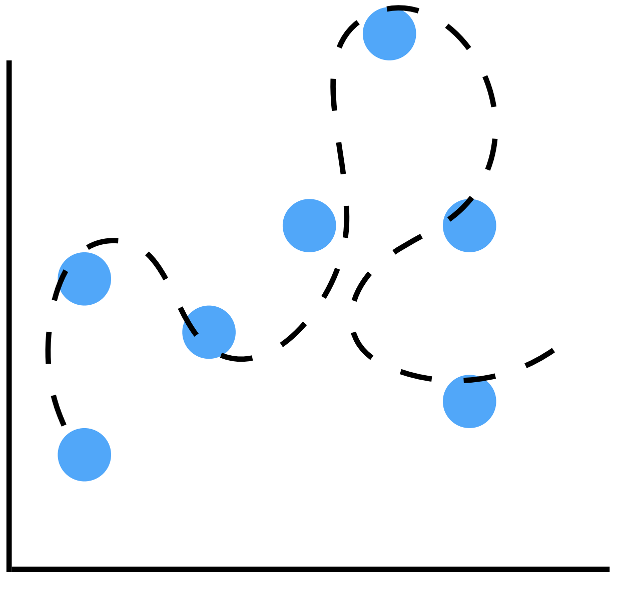
ML = Function Approximation



Defining an ML problem

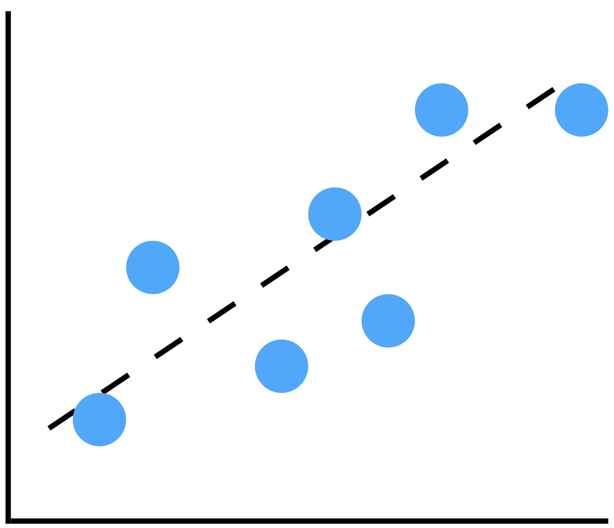
ML = Function Approximation

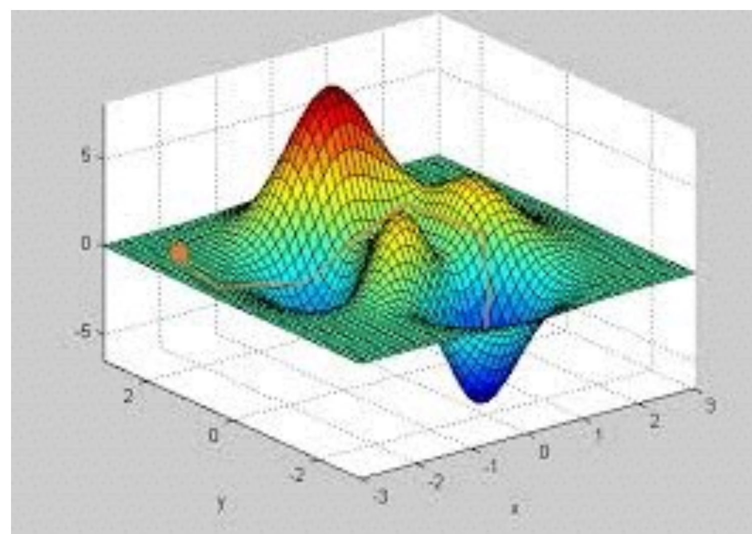


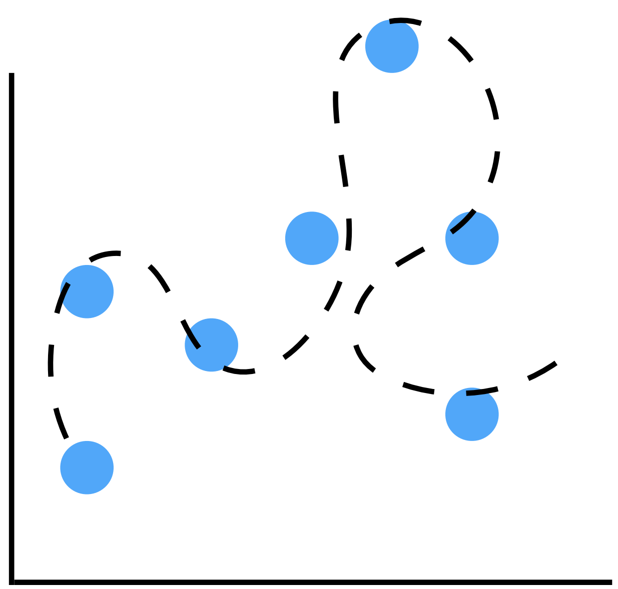


Defining an ML problem

ML = Function Approximation

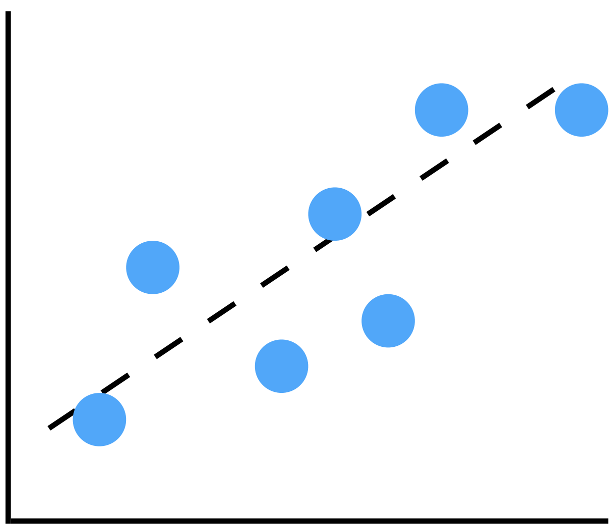


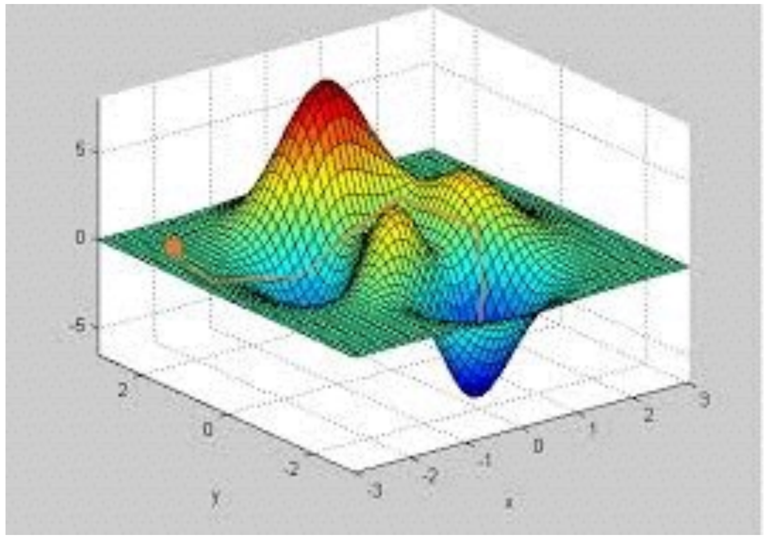


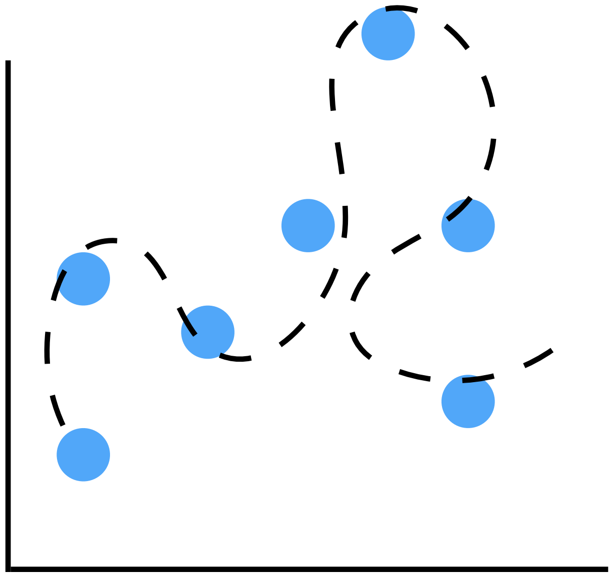


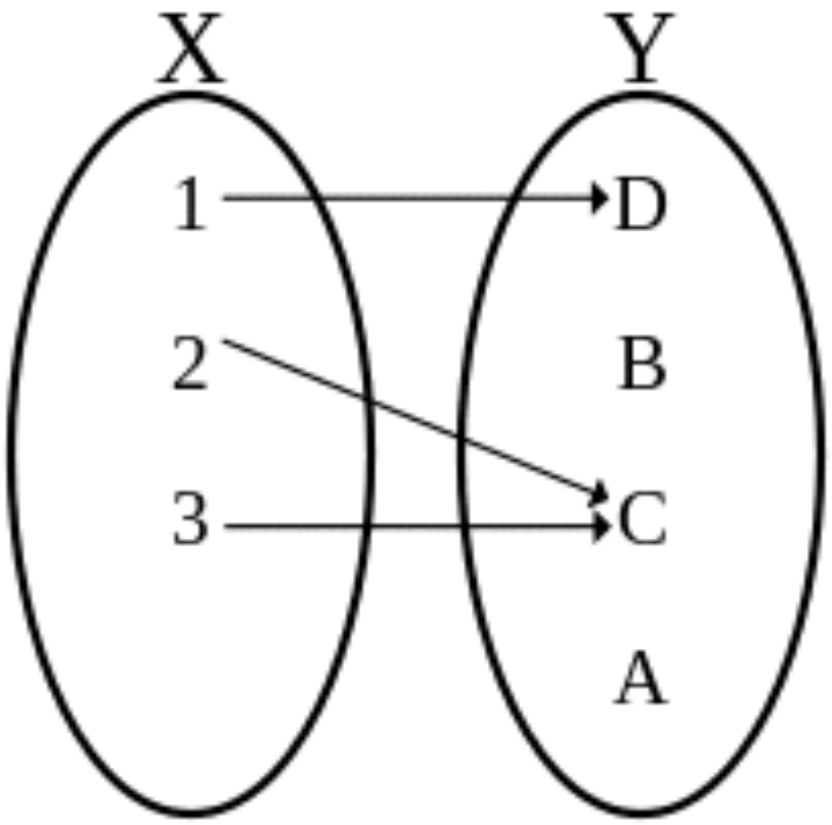
Defining an ML problem

ML = Function Approximation



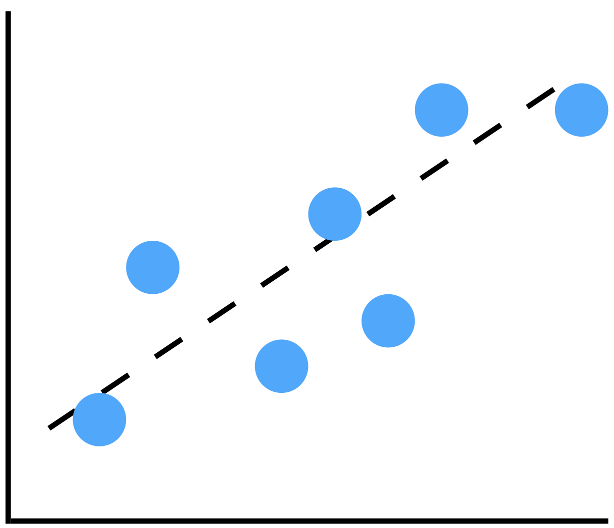


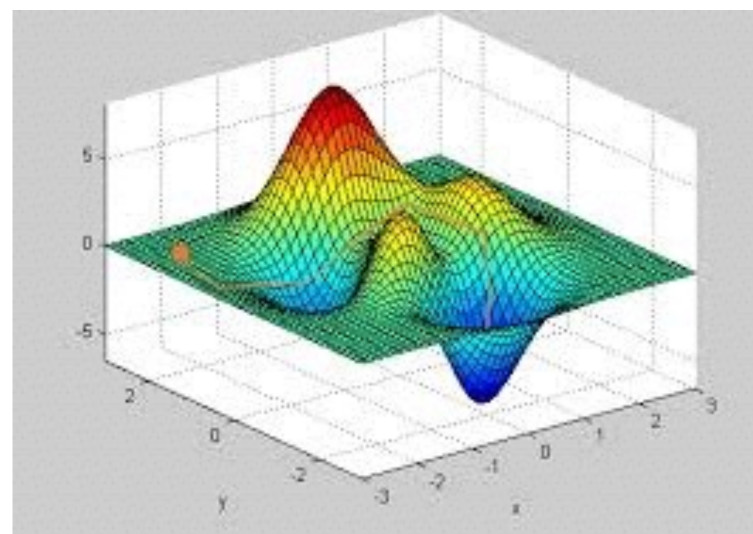




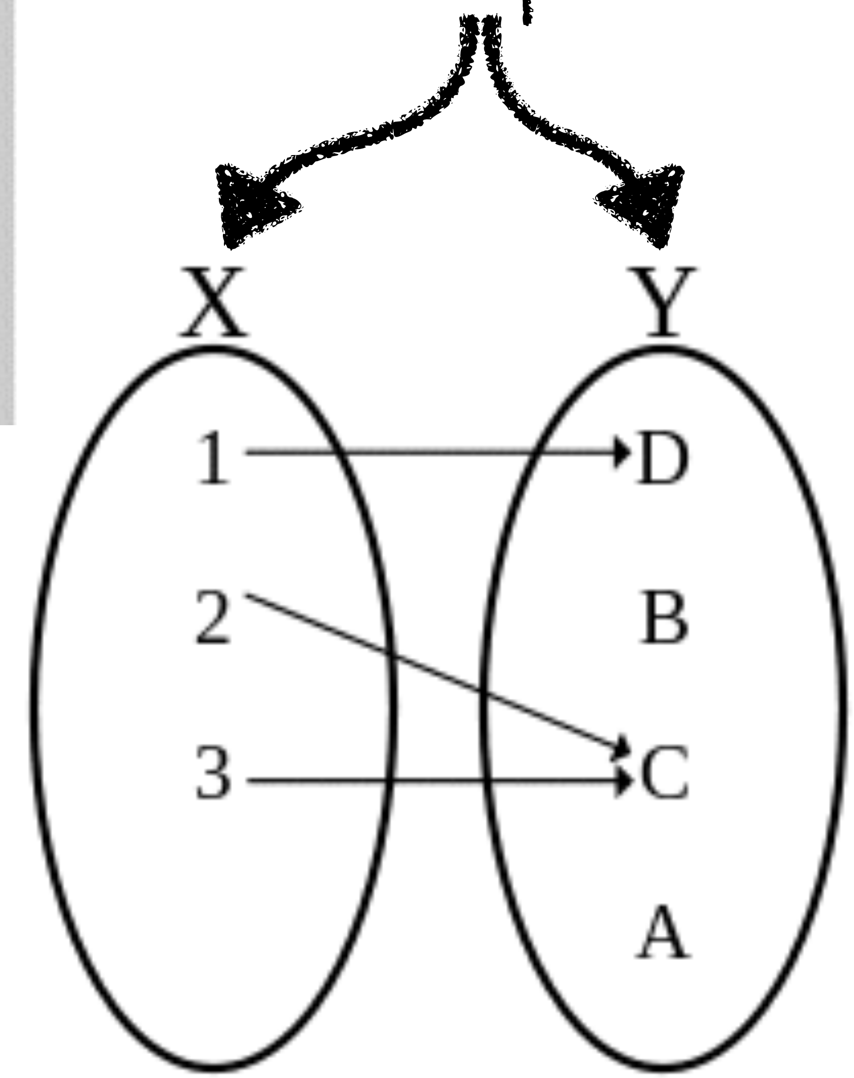
Defining an ML problem

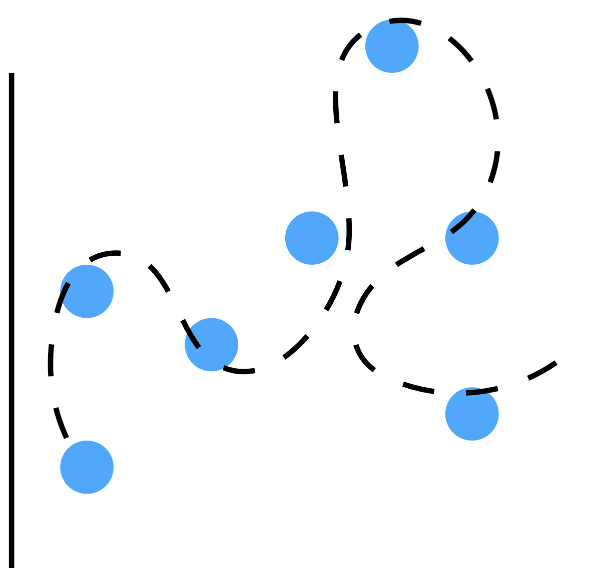
ML = Function Approximation





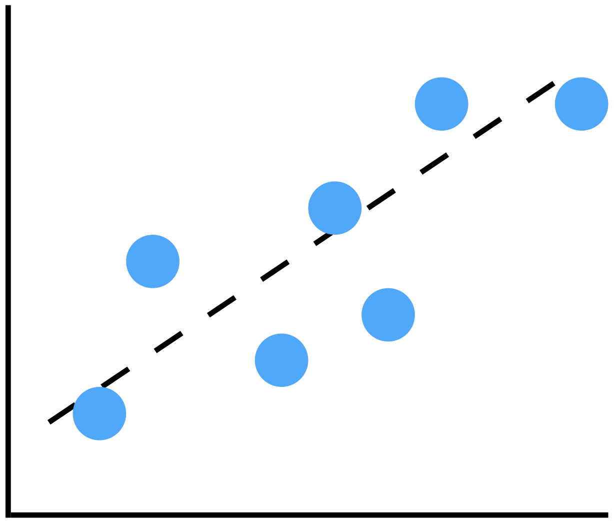
You define inputs and outputs.

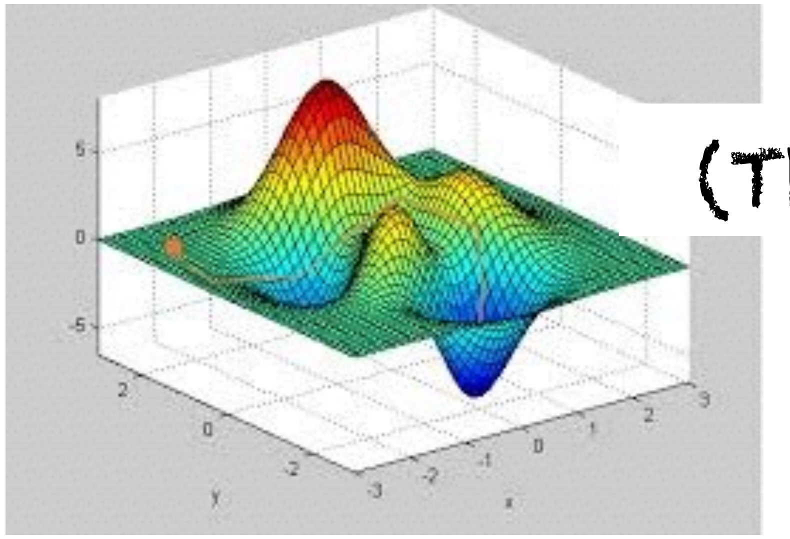


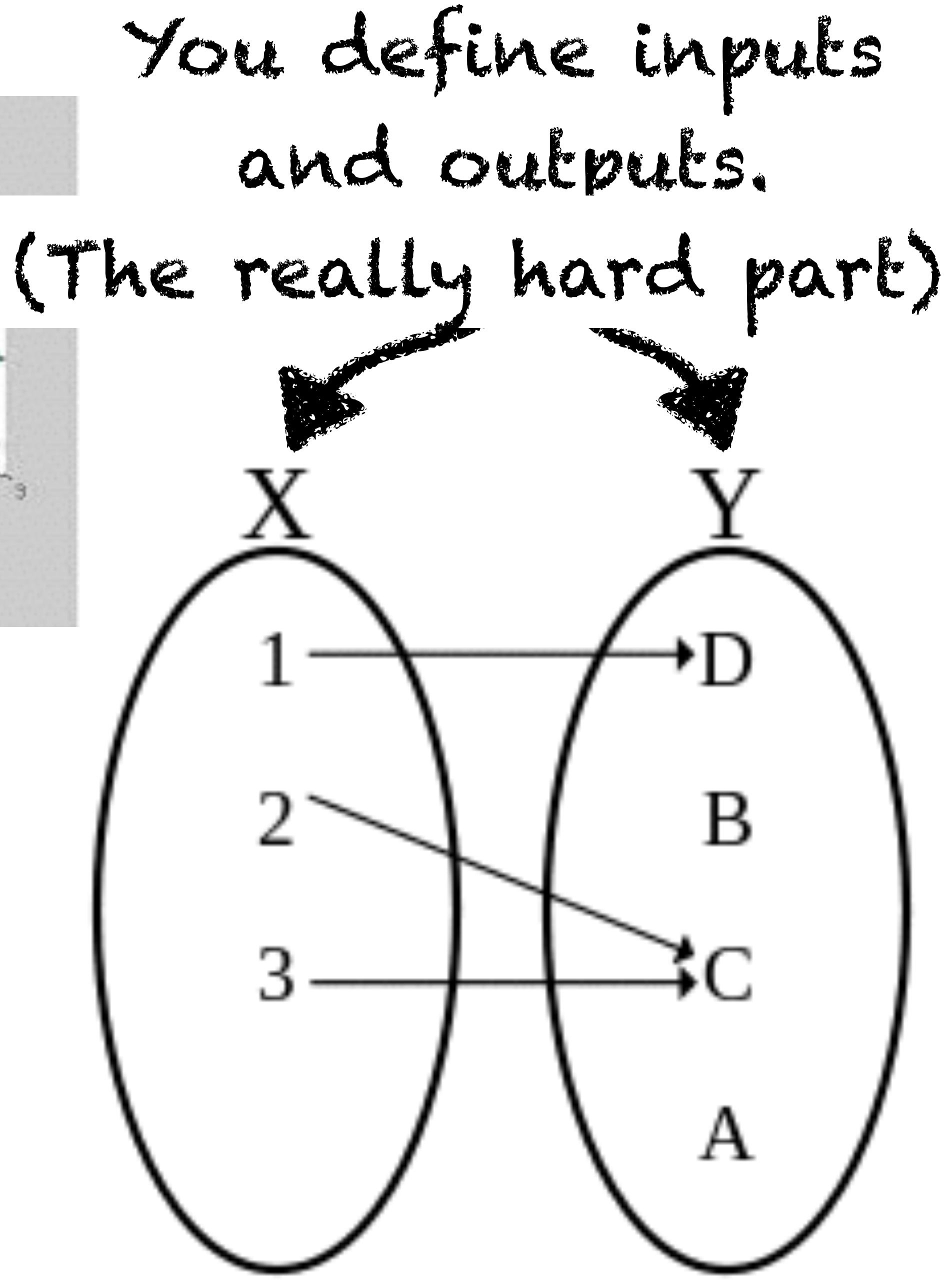


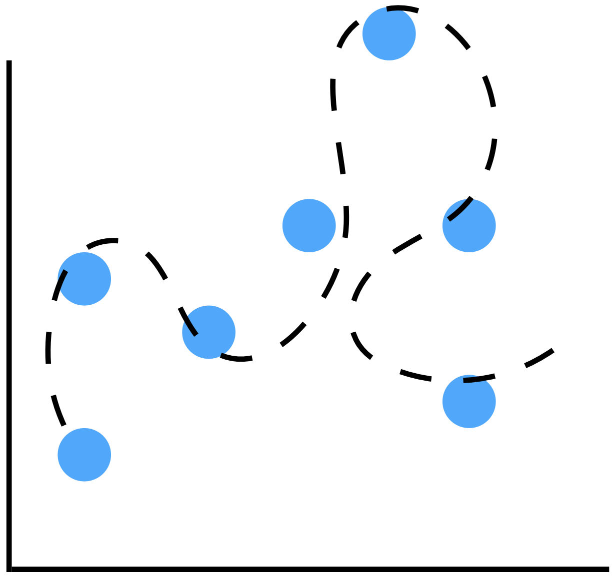
Defining an ML problem

ML = Function Approximation





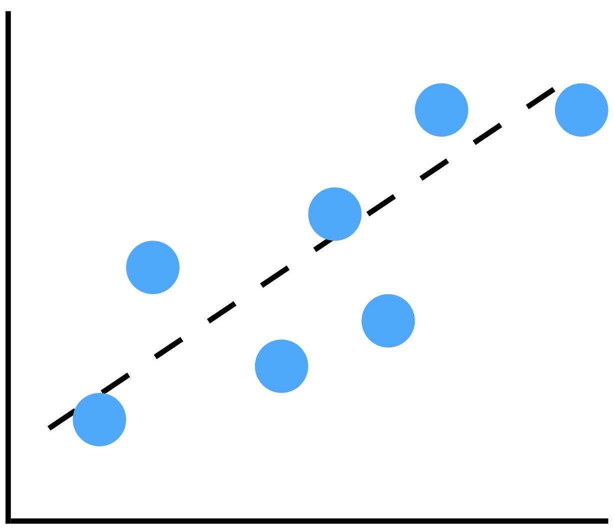


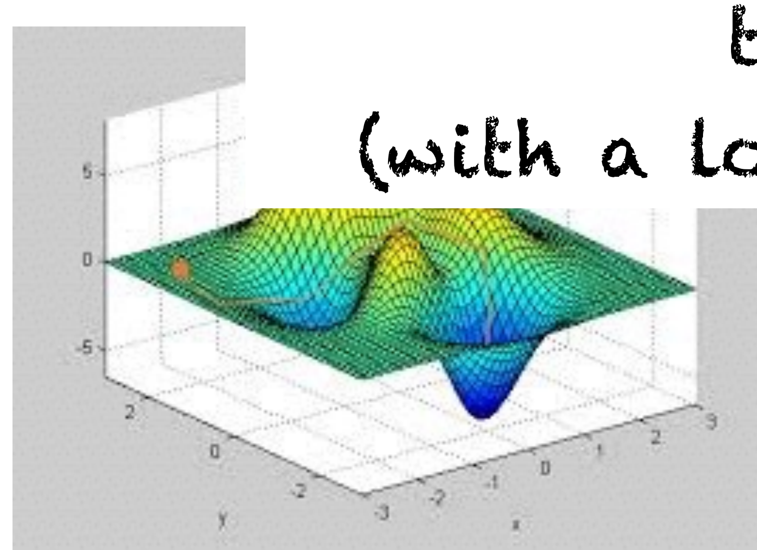


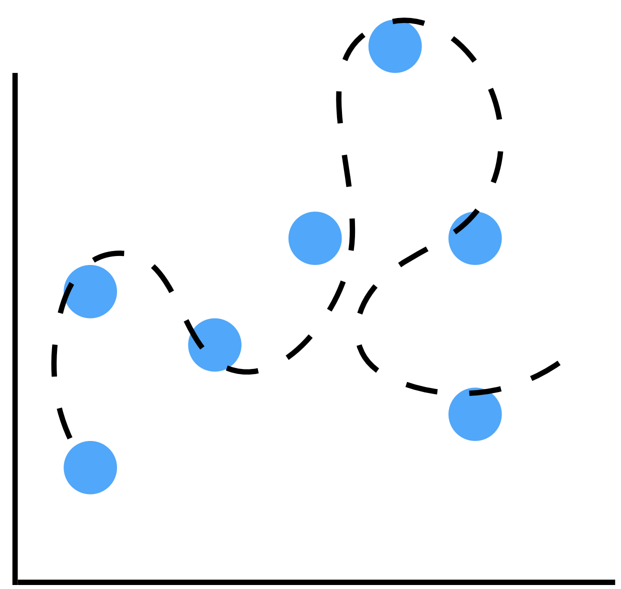
Defining an ML problem

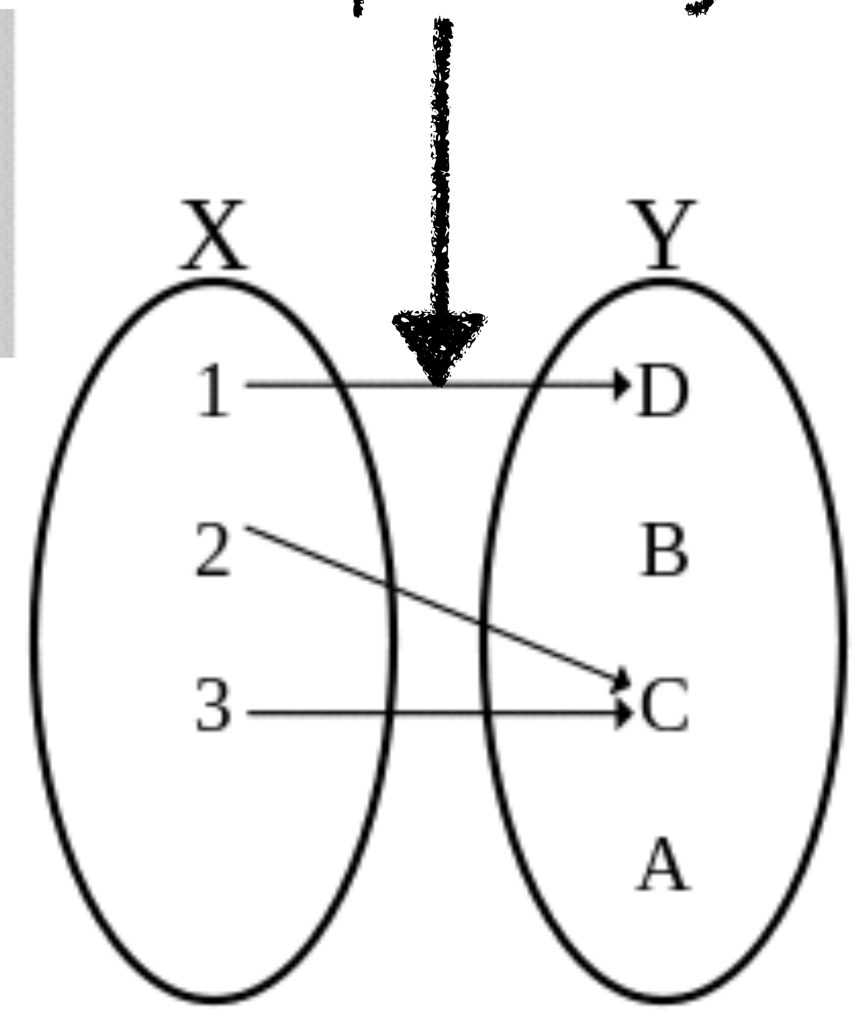
ML = Functic

The machine will (ideally) learn
the function
(with a lot of help from you)



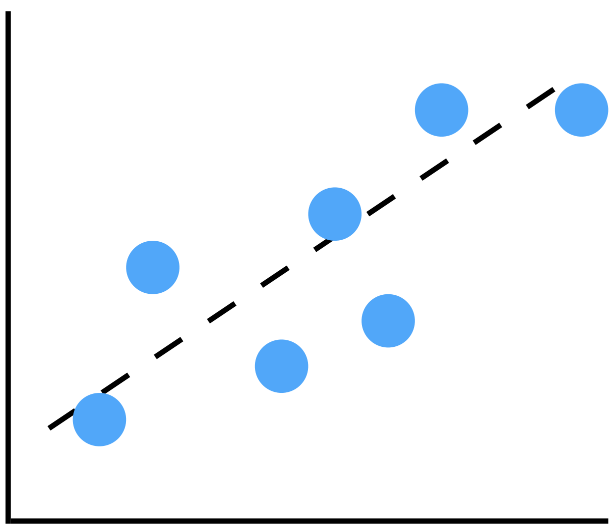


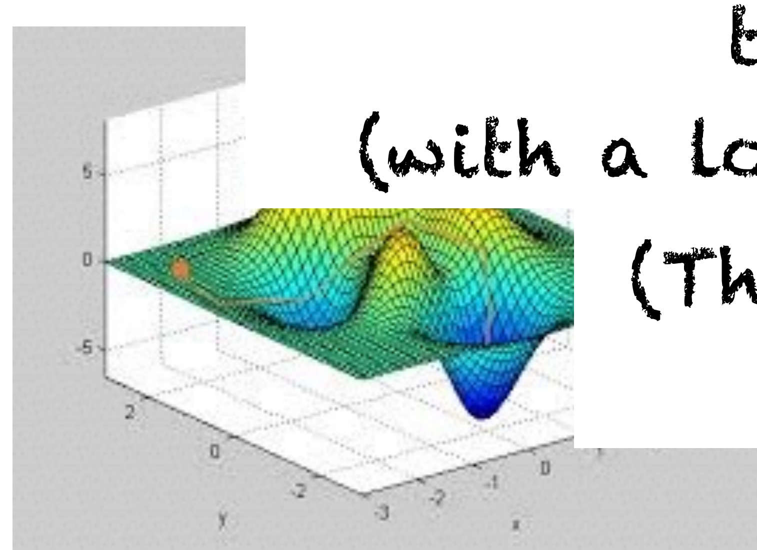




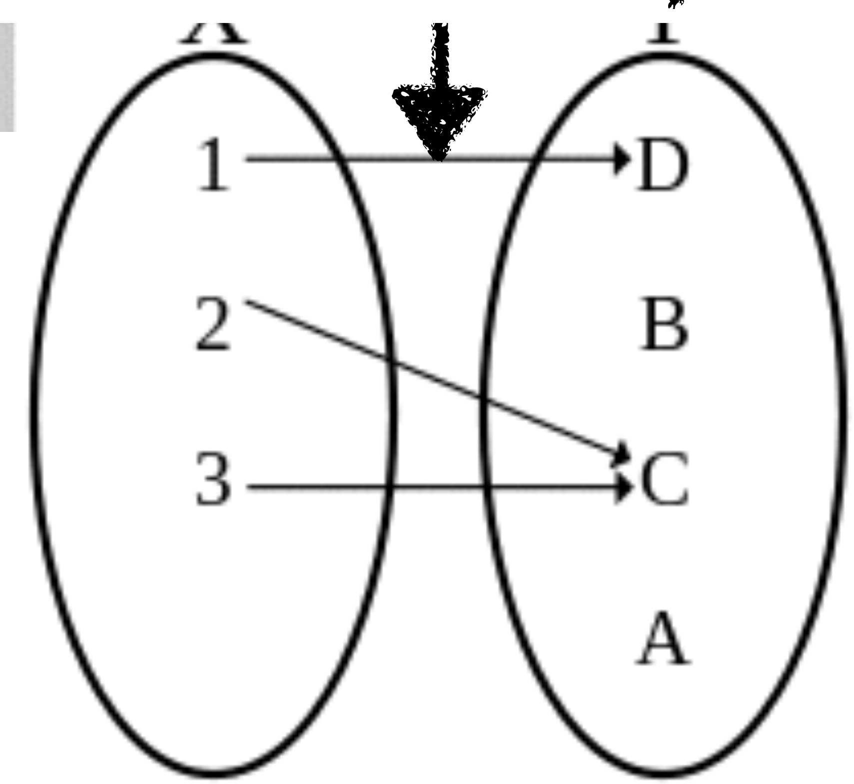
Defining an ML problem

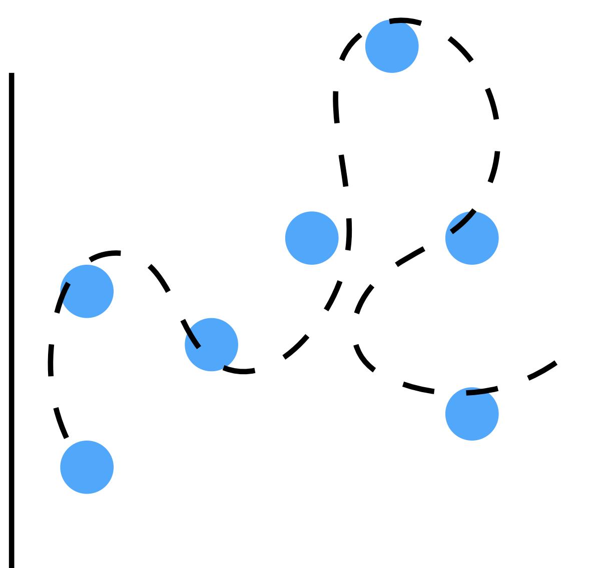
ML = Functi



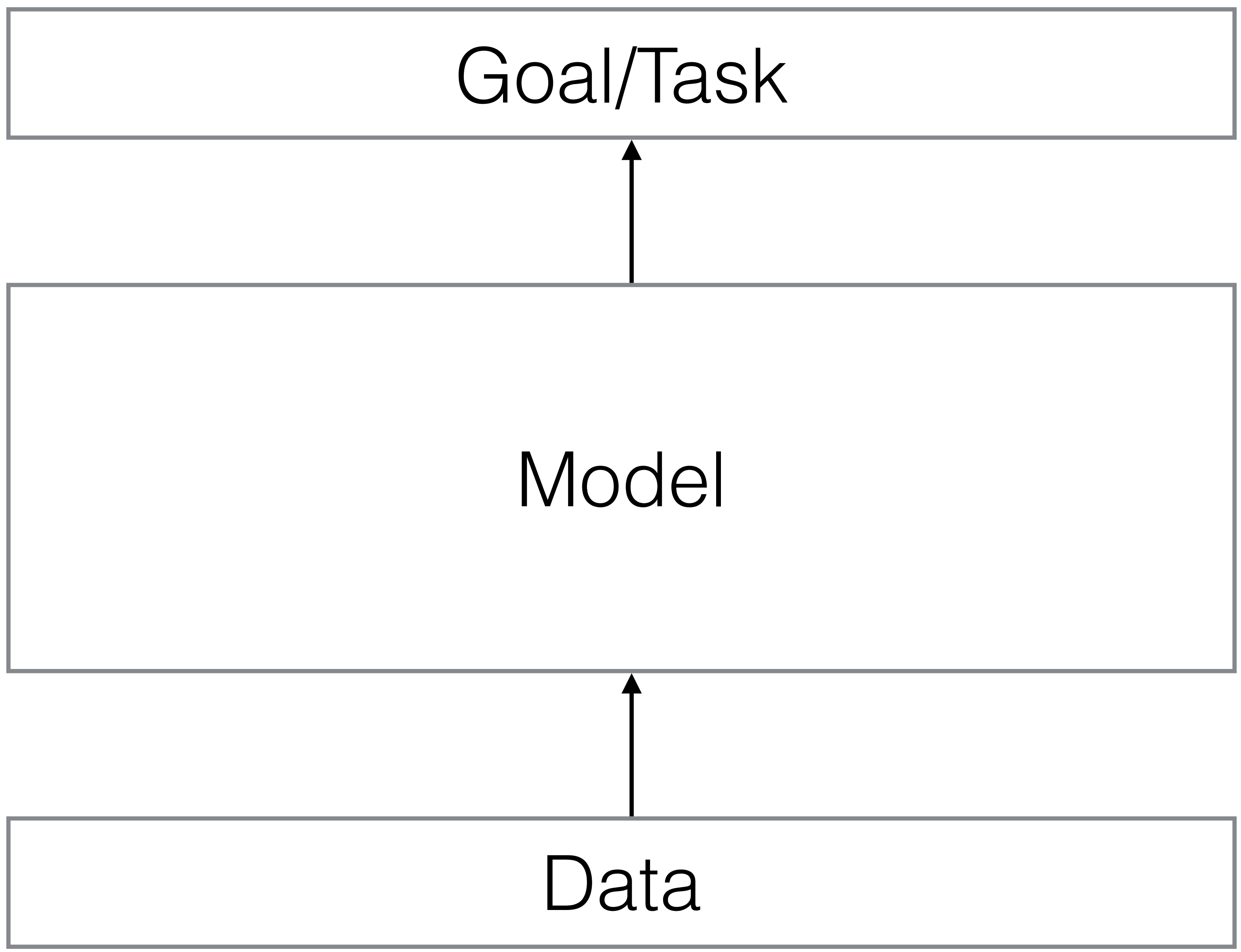


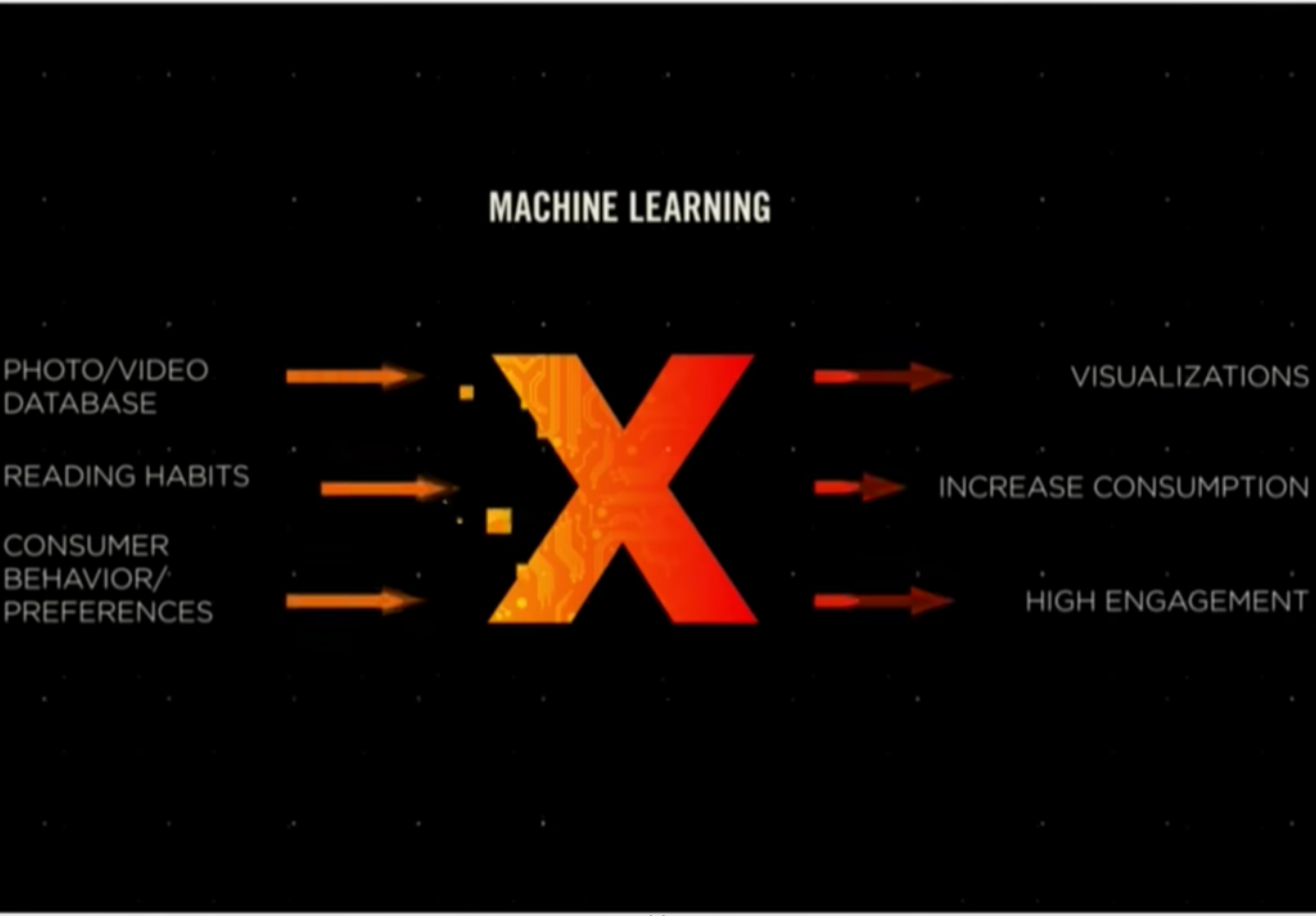
The machine will (ideally) learn
the function
(with a lot of help from you)
(The part that gets the
most attention.)

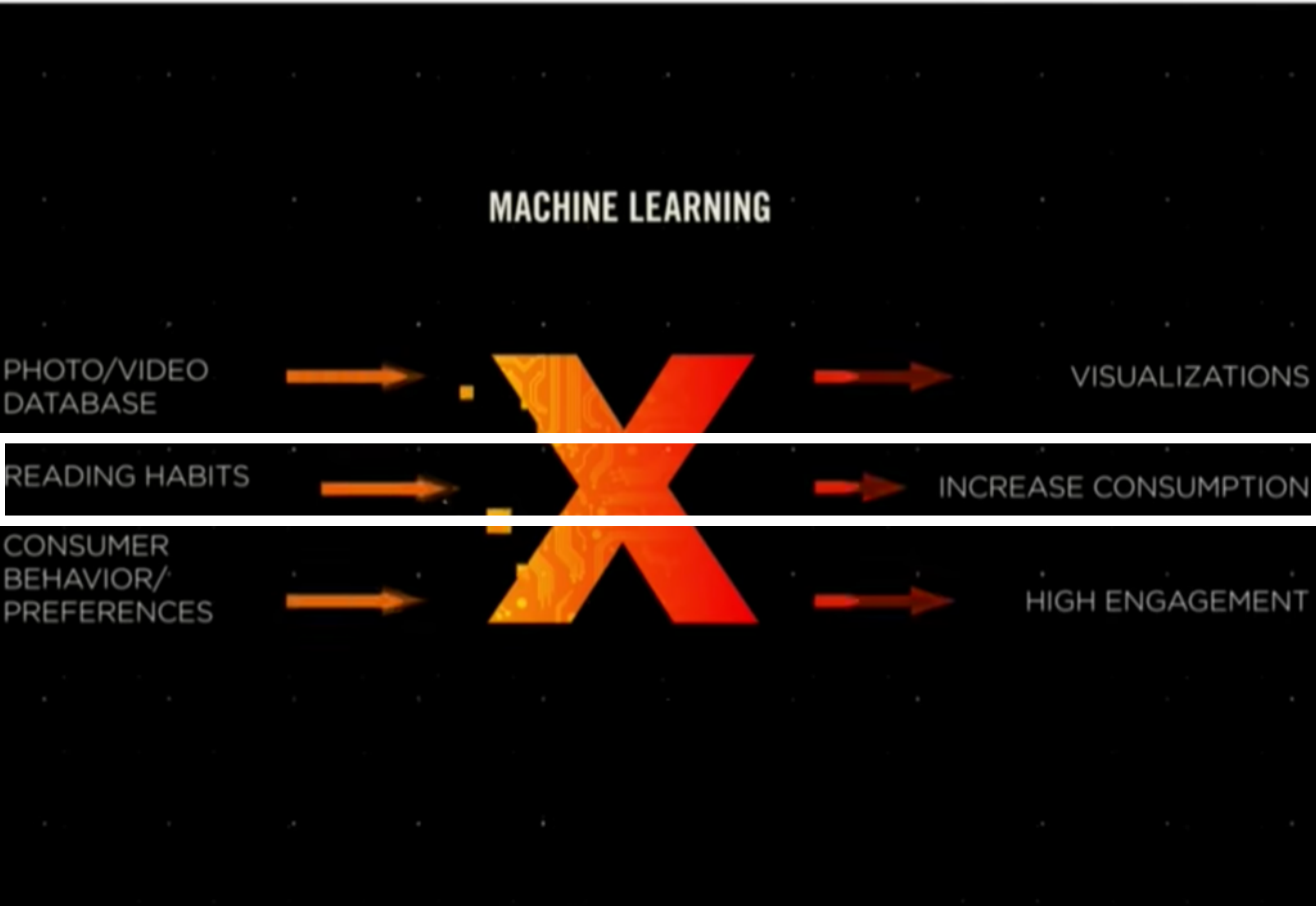




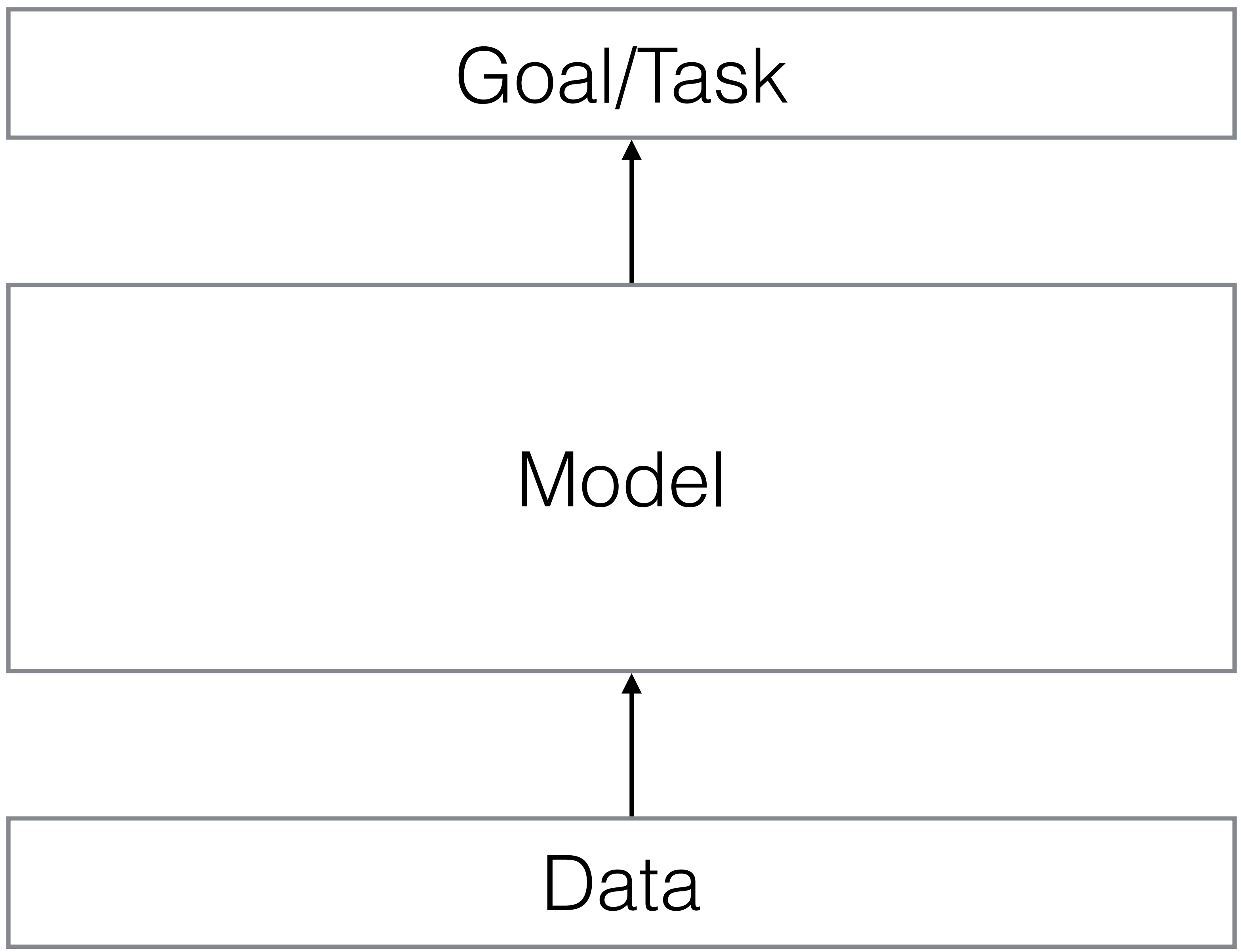
Defining an ML problem




MACHINE LEARNING
PHOTO/VIDEO DATABASE
READING HABITS
CONSUMER BEHAVIOR/ PREFERENCES
X
VISUALIZATIONS
INCREASE CONSUMPTION
HIGH ENGAGEMENT


MACHINE LEARNING
PHOTO/VIDEO DATABASE
READING HABITS
CONSUMER BEHAVIOR/ PREFERENCES
X
VISUALIZATIONS
INCREASE CONSUMPTION
HIGH ENGAGEMENT

Defining an ML problem



Defining an ML problem

Task = Increase Consumption

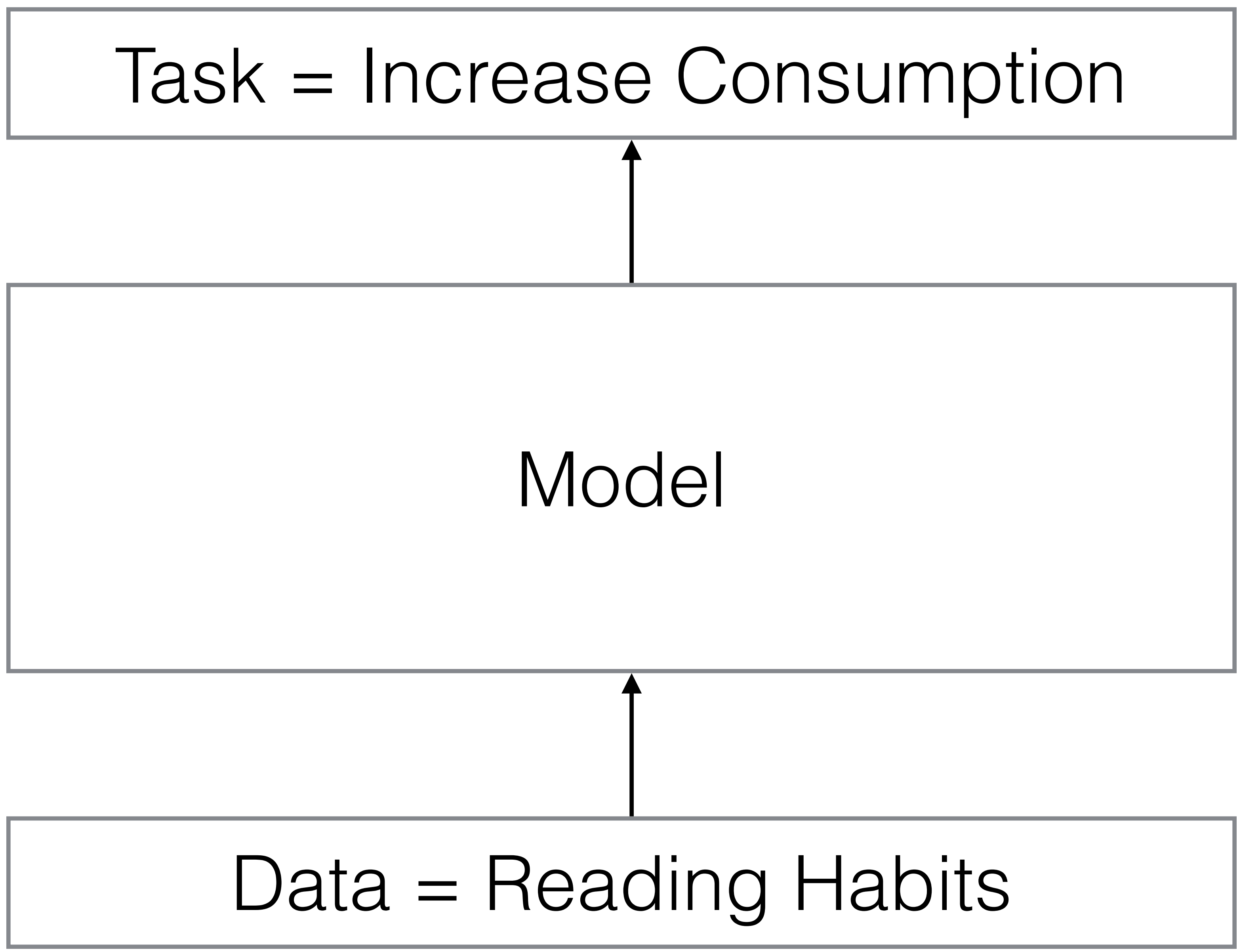
↑

Model

↑

Data

Defining an ML problem



Defining an ML problem

Task = Increase Consumption

Model

Data = Reading Habits

Defining an ML problem

Objective/Loss Function = ???

~~Task = Increase Consumption~~

Model

Data = Reading Habits

Defining an ML problem

Objective/Loss Function = ???

~~Task = Increase Consumption~~

Model

~~Data = Reading Habits~~

Features = ???

Defining an ML problem

Objective/Loss Function = ???

~~Task = Increase Consumption~~

Model

~~Data = Reading Habits~~

Features = ???

Prediction Target

- Goal = Increase consumption of “content” NOS for your ~~clickbait farm~~ pulitzer-prize worthy publication

Prediction Target

- Goal = Increase consumption of “content” NOS for your ~~clickbait farm~~ pulitzer-prize worthy publication
- Objective function….ideas?

Discussion Question!

Prediction Target

- Goal = Increase consumption of “content” NOS for your ~~clickbait farm~~ pulitzer-prize worthy publication

- Objective function….ideas?

> - Time spent on site (avg. per user/total)
> - Number of users
> - Number of articles read (need to define “read”)
> - Number of articles clicked on
> - Time per article
> - Articles shared…

Prediction Target

- Goal = Increase consumption of “content” NOS for your ~~clickbait farm~~ pulitzer-prize worthy publication

- Objective function….ideas?

- Time spent on site (avg. per user/total)
- Number of users
- Number of articles read (need to define “read”)
- **Number of articles clicked on**
- Time per article
- Articles shared…

Defining an ML problem

Objective/Loss Function = ???

~~Task = Increase Consumption~~

Model

~~Data = Reading Habits~~

Features = ???

Defining an ML problem

Objective/Loss Function = total number of clicks

~~Task = Increase Consumption~~

Model

~~Data = Reading Habits~~

Features = ???

Defining an ML problem

Objective/Loss Function = total number of clicks

~~Task = Increase Consumption~~

↑

Model

↑

~~Data = Reading Habits~~

Features = ???

Features

- Data = Reading habits collected via ~~unauthorized ever-present cookies and remote control of webcam~~ user-consented GDPR-compliant data usage agreements

Features

- Data = Reading habits collected via ~~unauthorized ever-present cookies and remote control of webcam~~ user-consented GDPR-compliant data usage agreements
- Features….ideas?

Features

- Data = Reading habits collected via ~~unauthorized ever-present cookies and remote control of webcam~~ user-consented GDPR-compliant data usage agreements

- Features….ideas?

Discussion Question!

Features

- Data = Reading habits collected via ~~unauthorized ever-present cookies and remote control of webcam~~ user-consented GDPR-compliant data usage agreements

- Features

- Article topic
- Recency (minutes since release)
- Words in title/snippet
- Presence of photo
- Reading level
- Fonts/layouts
- User location
- Topics of articles the user has read previously
- Number of likes

…

Features

- Data = Reading habits collected via ~~unauthorized ever-present cookies and remote control of webcam~~ user-consented GDPR-compliant data usage agreements

- Features

- Article topic
- **Recency** (minutes since release)
- **Words in title**/snippet
- **Presence of photo**
- **Reading level**
- Fonts/layouts
- User location
- Topics of articles the user has read previously
- Number of likes

…

Features

- Recency: Float
- Words in title: String
- Presence of photo: Boolean
- Reading level: Integer

Features

Clicks	Recency	Reading Level	Photo	Title
10	1.3	11	1	“New Tax Guidelines”
1000	1.7	3	1	“This 600lb baby…”
1000000	2.4	2	1	“18 reasons you should *never* look at this cat unless you…”
1	5.9	19	0	“The Brothers Karamazov: a neo-post-globalist perspective”

Features

y

Clicks	Recency	Reading Level	Photo	Title
10	1.3	11	1	“New Tax Guidelines”
1000	1.7	3	1	“This 600lb baby…”
1000000	2.4	2	1	“18 reasons you should *never* look at this cat unless you…”
1	5.9	19	0	“The Brothers Karamazov: a neo-post-globalist perspective”

Features

X

Clicks	Recency	Reading Level	Photo	Title
10	1.3	11	1	"New Tax Guidelines"
1000	1.7	3	1	"This 600lb baby…"
1000000	2.4	2	1	"18 reasons you should *never* look at this cat unless you…"
1	5.9	19	0	"The Brothers Karamazov: a neo-post-globalist perspective"

Features

numeric features – defined for (nearly) every row

Clicks	Recency	Reading Level	Photo	Title
10	1.3	11	1	“New Tax Guidelines”
1000	1.7	3	1	“This 600lb baby…”
1000000	2.4	2	1	“18 reasons you should *never* look at this cat unless you…”
1	5.9	19	0	“The Brothers Karamazov: a neo-post-globalist perspective”

Features

boolean features – 0 or 1 ("dummy" variables)

Clicks	Recency	Reading Level	Photo	Title
10	1.3	11	1	"New Tax Guidelines"
1000	1.7	3	1	"This 600lb baby…"
1000000	2.4	2	1	"18 reasons you should *never* look at this cat unless you…"
1	5.9	19	0	"The Brothers Karamazov: a neo-post-globalist perspective"

Features

strings = boolean features — 0 or 1 ("dummy" variables)

Clicks	Recency	Reading Level	Photo	Title
10	1.3	11	1	"New Tax Guidelines"
1000	1.7	3	1	"This 600lb baby…"
1000000	2.4	2	1	"18 reasons you should *never* look at this cat unless you…"
1	5.9	19	0	"The Brothers Karamazov: a neo-post-globalist perspective"

Features

strings = boolean features – 0 or 1 ("dummy" variables)

Clicks	Recency	Reading Level	Photo	Title: "new"	Title: "tax"	Title: "this"	Title: "…"	…
10	1.3	11	1	1	0	0	0	…
1000	1.7	3	1	0	0	1	1	…
1000000	2.4	2	1	0	0	1	1	…
1	5.9	19	0	0	0	0	0	…

Features

"sparse features" – 0 for most rows

Clicks	Recency	Reading Level	Photo	Title: "new"	Title: "tax"	Title: "this"	Title: "…"	…
10	1.3	11	1	1	0	0	0	…
1000	1.7	3	1	0	0	1	1	…
1000000	2.4	2	1	0	0	1	1	…
1	5.9	19	0	0	0	0	0	…

Clicker Question!

Clicker Question!

For the problem set up, how many features will there be? I.e. how many columns in our X matrix, (not including Y)?

```
Y: happiness
X1: day of week ("monday", "tuesday", … "sunday")
X2: bank account balance (real value)
X3: breakfast (yes,no)
X4: whether you have found your inner peace
(yes,no)
X5: words from last week's worth of tweets
(assuming tweets are at most 15 words long and
there are 100K words in the English vocabulary)
```

(a) 112,000

(b) 5

(c) 27

(d) 110,000

Clicker Question!

For the problem set up, how many features will there be? I.e. how many columns in our X matrix, (not including Y)?

```
Y: happiness
X1: day of week ("monday", "tuesday", … "sunday") 7
X2: bank account balance (real value) 1
X3: breakfast (yes,no) 1
X4: whether you have found your inner peace 1
(yes,no)
X5: words from last week's worth of tweets 100,000
(assuming tweets are at most 15 words long and
there are 100K words in the English vocabulary)
```

(a) 100,012

(b) 5

(c) 27

(d) 100,010

Defining an ML problem

Objective/Loss Function = total number of clicks

~~Task = Increase Consumption~~

Model

~~Data = Reading Habits~~

Features = ???

Defining an ML problem

Objective/Loss Function = total number of clicks

~~Task = Increase Consumption~~

Model

~~Data = Reading Habits~~

Features = {Recency:float, ReadingLevel:Int, Photo:Bool, Title_New:Bool, Title_Tax:Bool, ...}

Defining an ML problem

Objective/Loss Function = total number of clicks

~~Task = Increase Consumption~~

Model

~~Data = Reading Habits~~

Features = {Recency:float, ReadingLevel:Int, Photo:Bool, Title_New:Bool, Title_Tax:Bool, ...}

Model

#1

- Make assumptions about the problem domain.

Model

#1

- Make assumptions about the problem domain.
- How is the data generated?

Model

#1

- Make assumptions about the problem domain.
- How is the data generated?
- How is the decision-making procedure structured?

Model

#1

- Make assumptions about the problem domain.
- How is the data generated?
- How is the decision-making procedure structured?
- What types of dependencies exist?

Model

#1

- Make assumptions about the problem domain.
- How is the data generated?
- How is the decision-making procedure structured?
- What types of dependencies exist?
- Trending buzzword: “inductive biases”

Model

#1

- Make assumptions about the problem domain.
- How is the data generated?
- How is the decision-making procedure structured?
- What types of dependencies exist?
- Trending buzzword: “inductive biases”

#2

- How to train the model?

Model

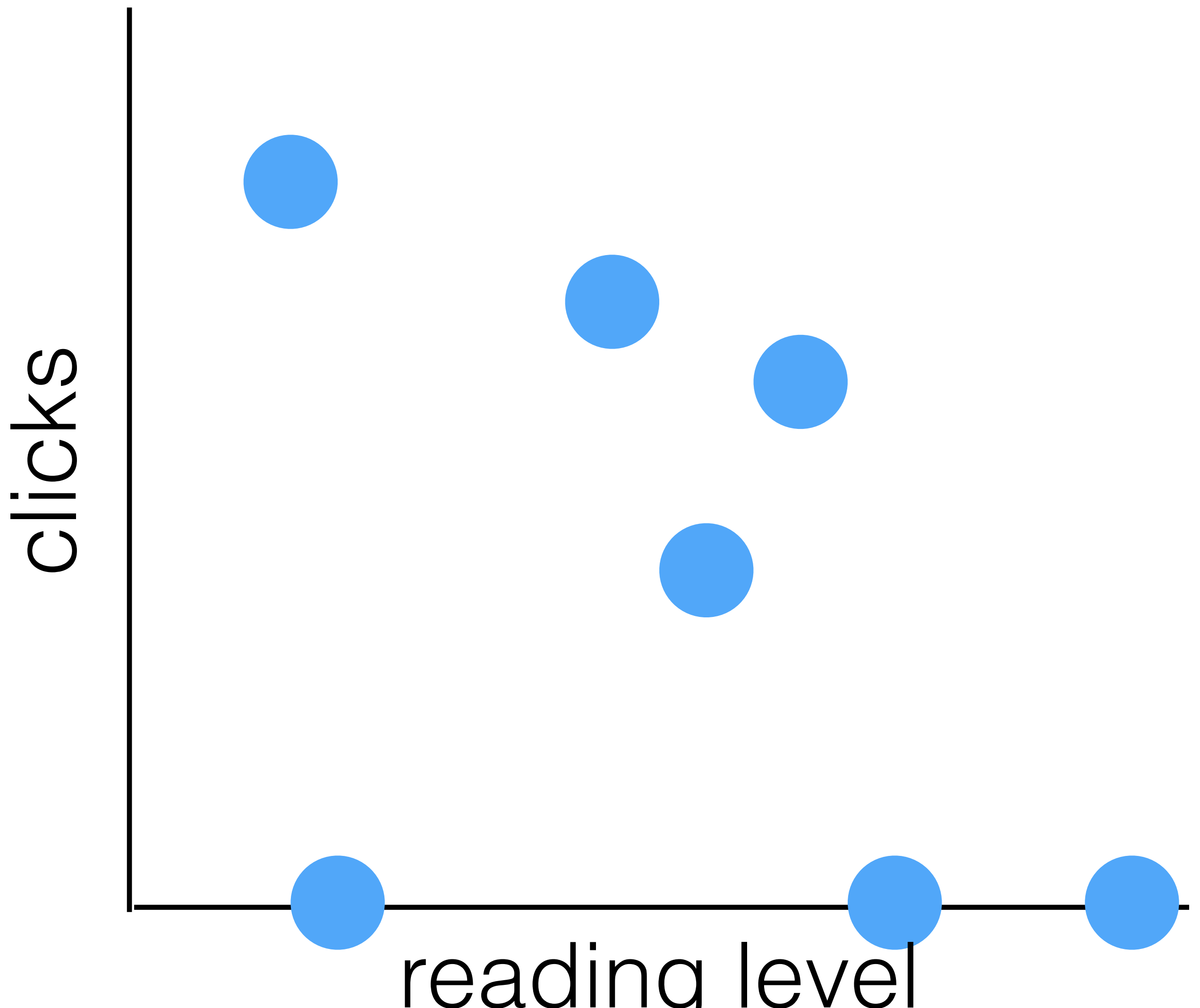
#1

- Make assumptions about the problem domain.
- How is the data generated?
- How is the decision-making procedure structured?
- What types of dependencies exist?
- Trending buzzword: “inductive biases”

#2

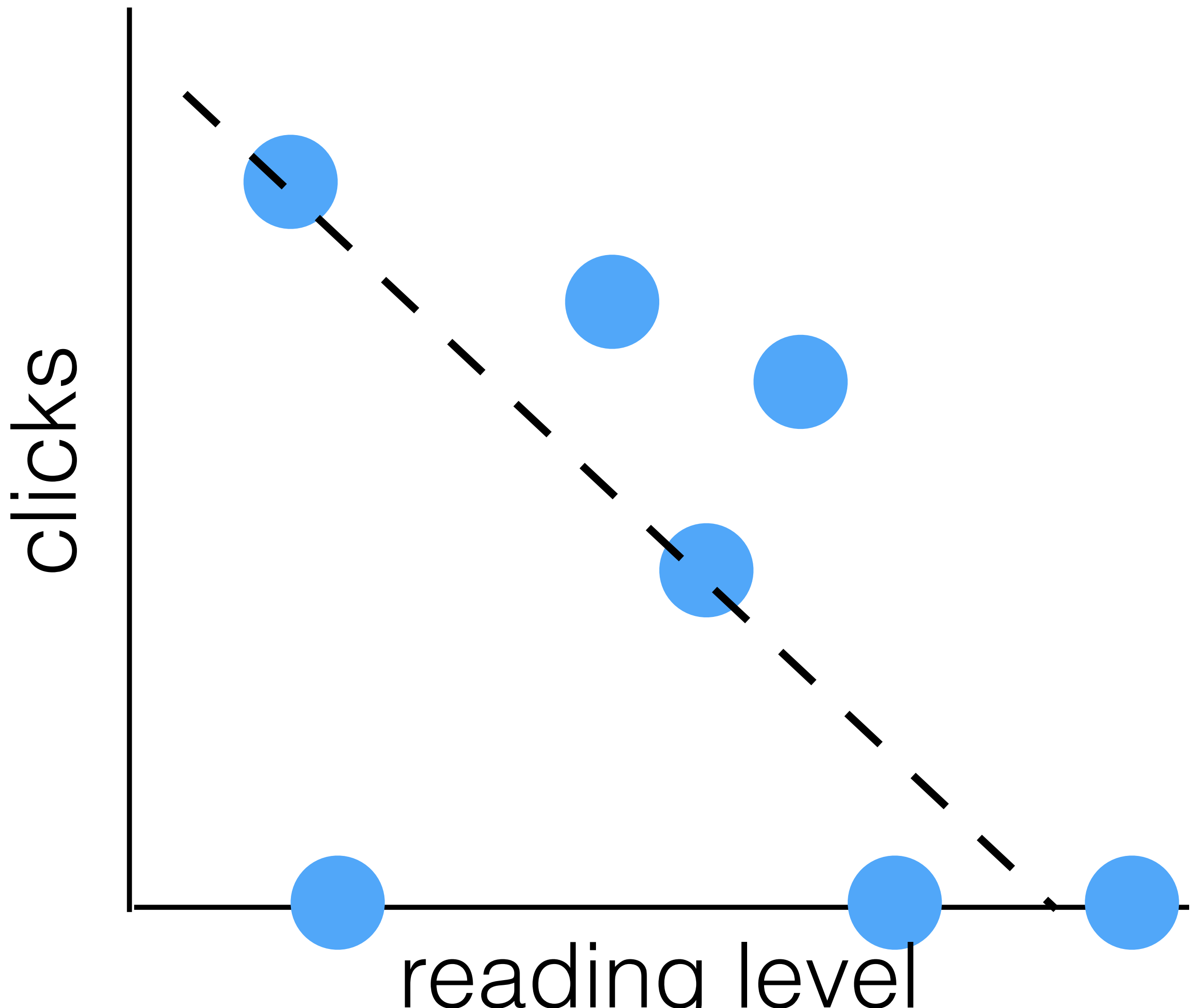
- How to train the model?

Model



Model

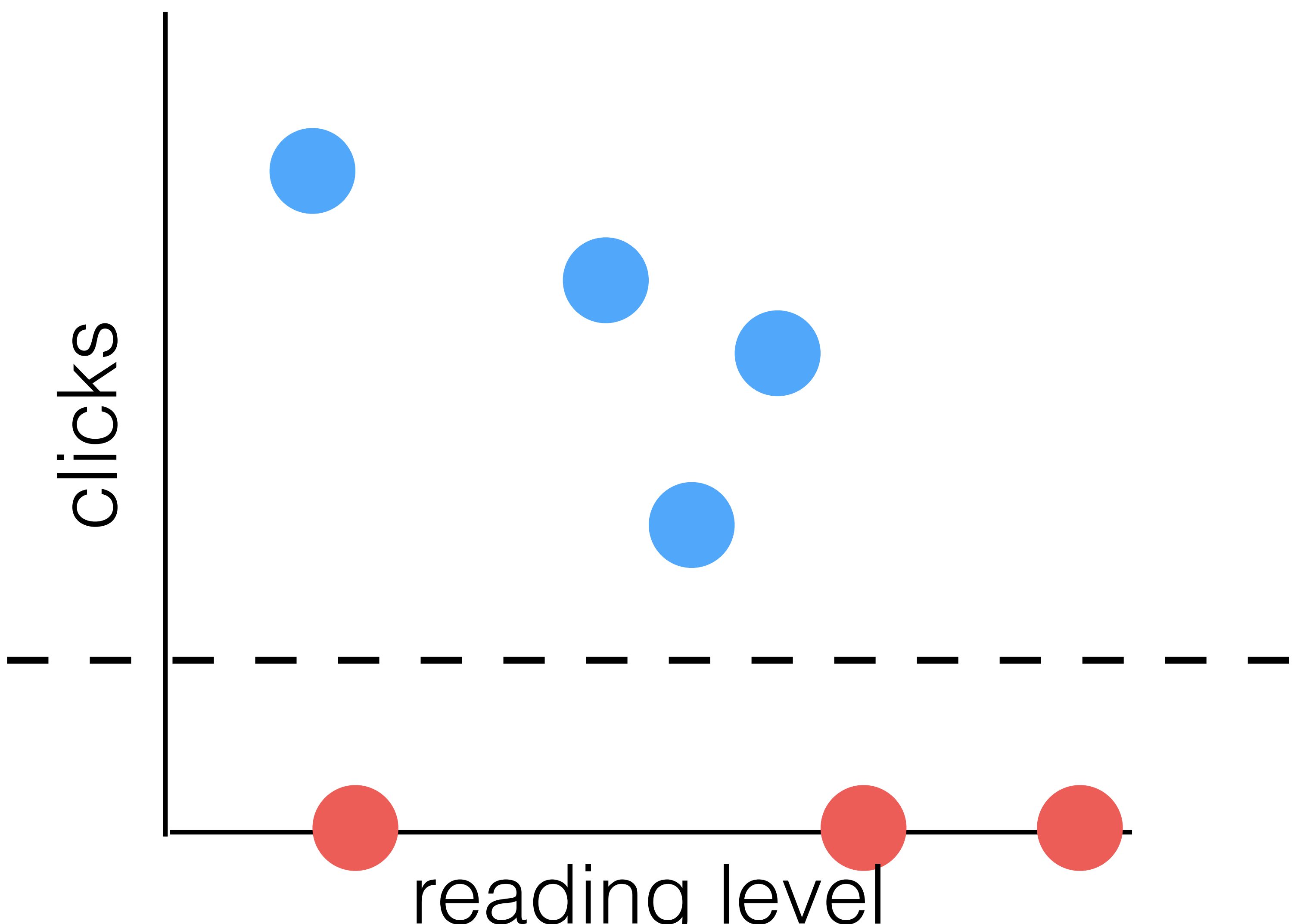
Regression: continuous (infinite) output
f(reading level) = # of clicks



Model

Classification: discrete (finite) output

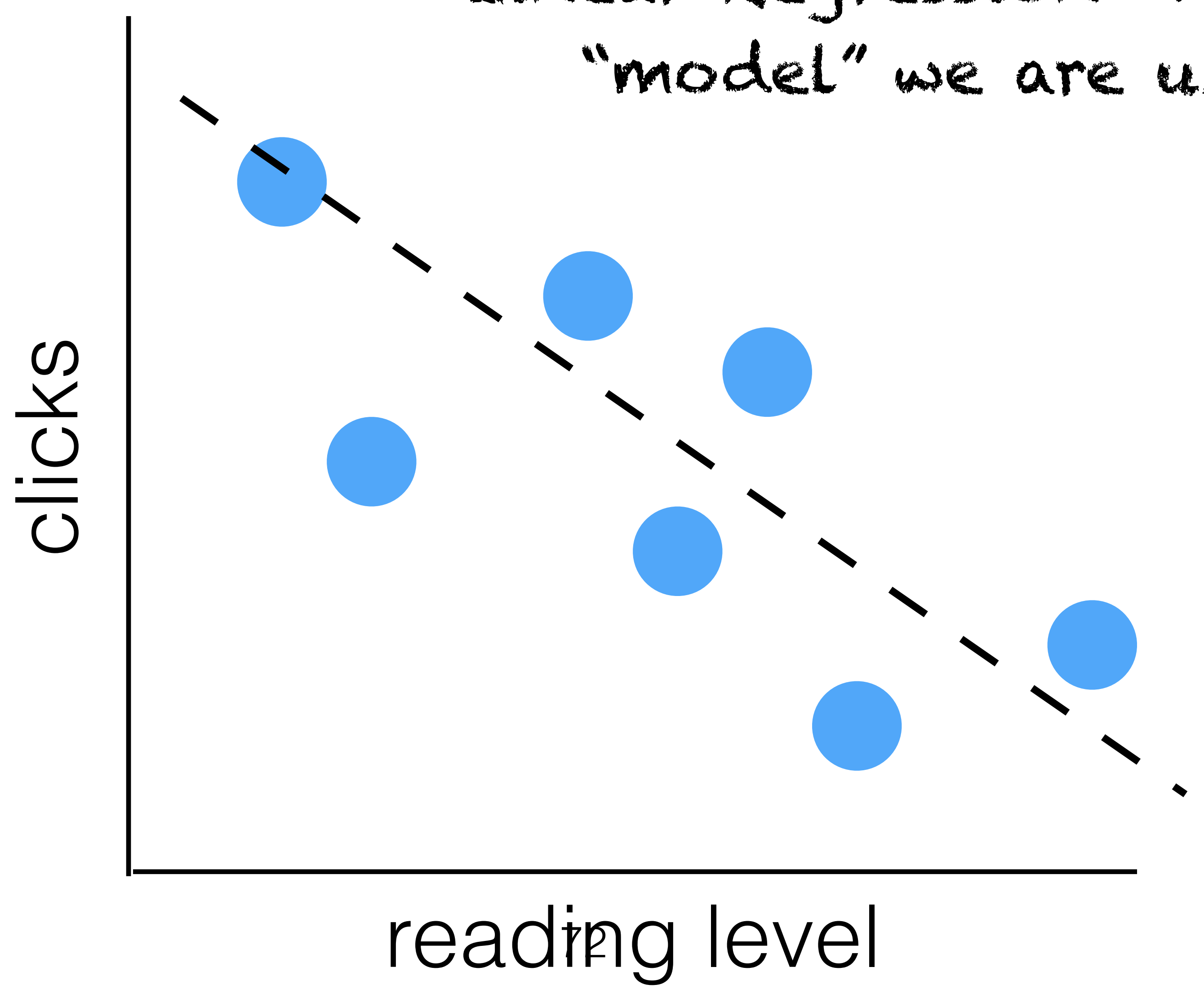
f(reading level) = {clicked, not clicked}



Model

```
clicks = m(reading_level) + b
```

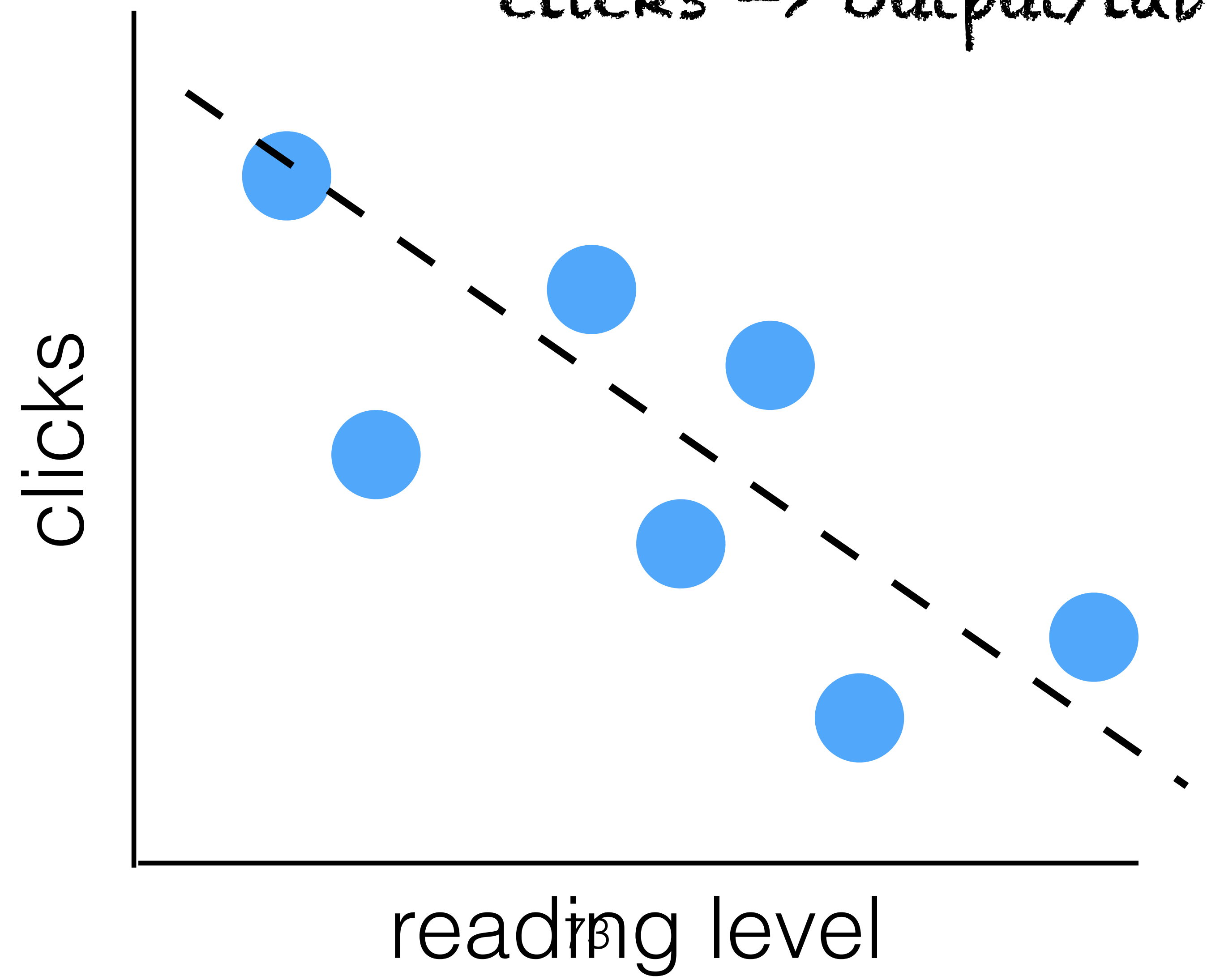
Linear Regression —> The specific "model" we are using here.



Model

`clicks = m(reading_level) + b`

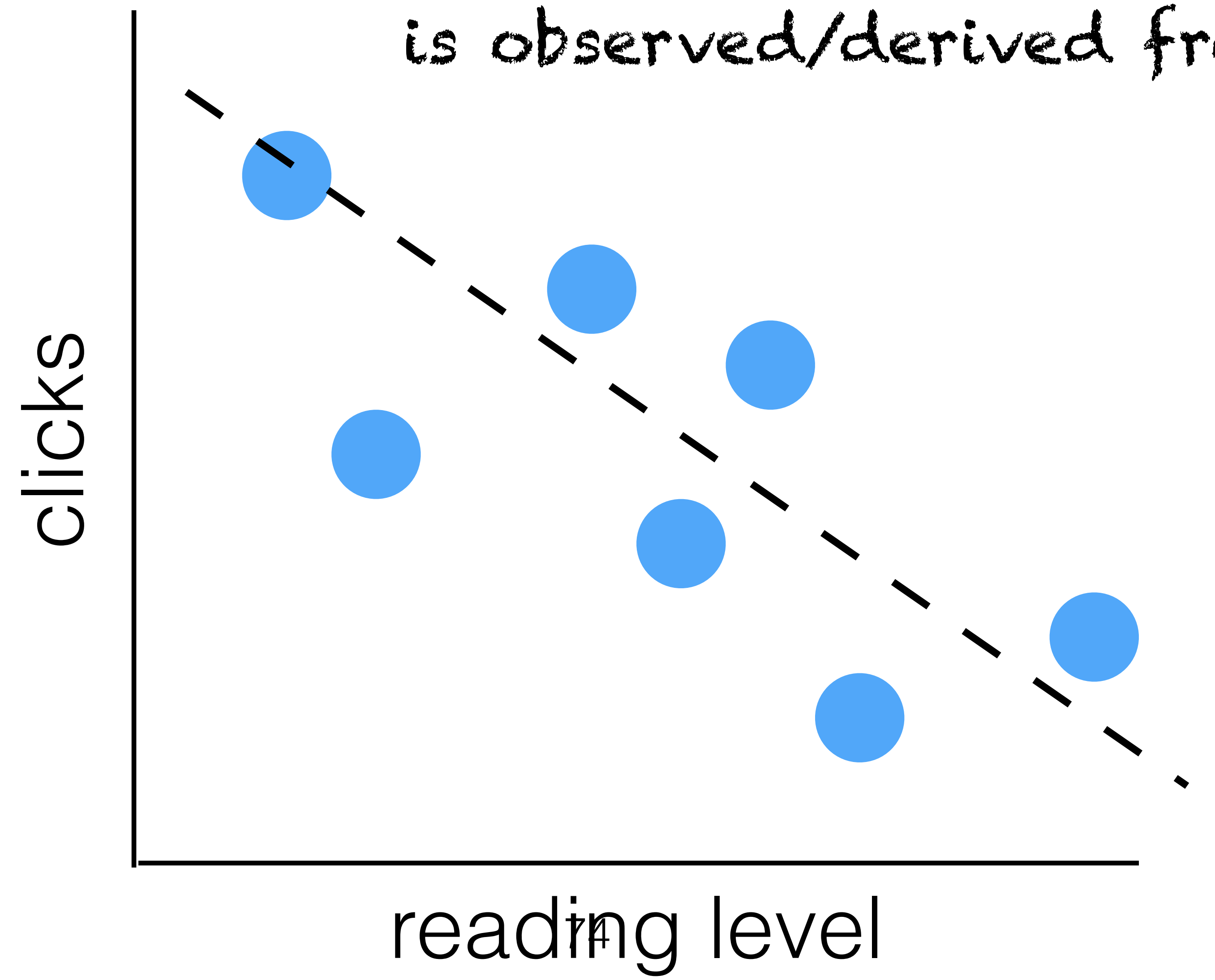
clicks → output/labels/target



Model

```
clicks = m(reading_level) + b
```

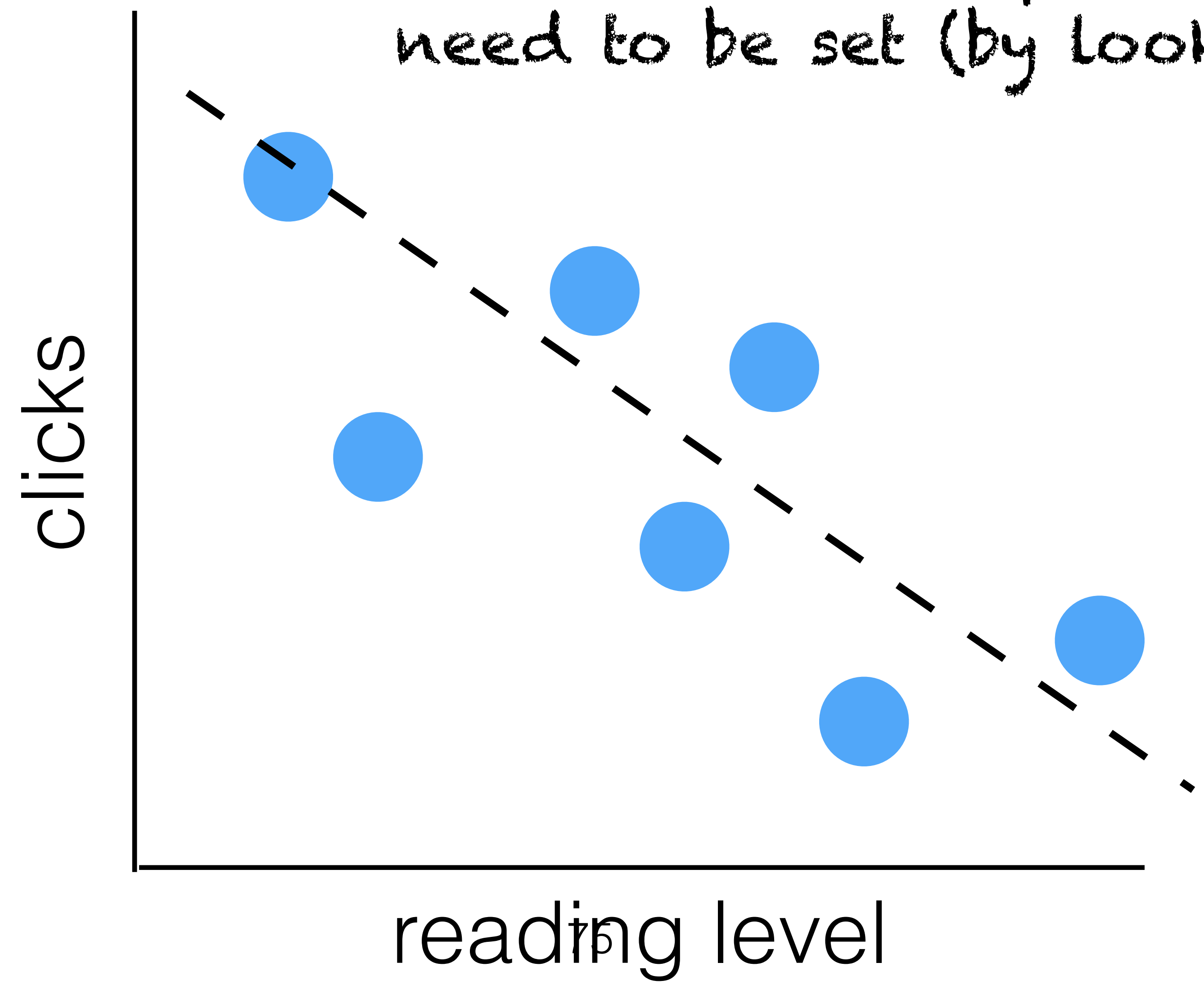
reading level —> The "feature" which is observed/derived from the data



Model

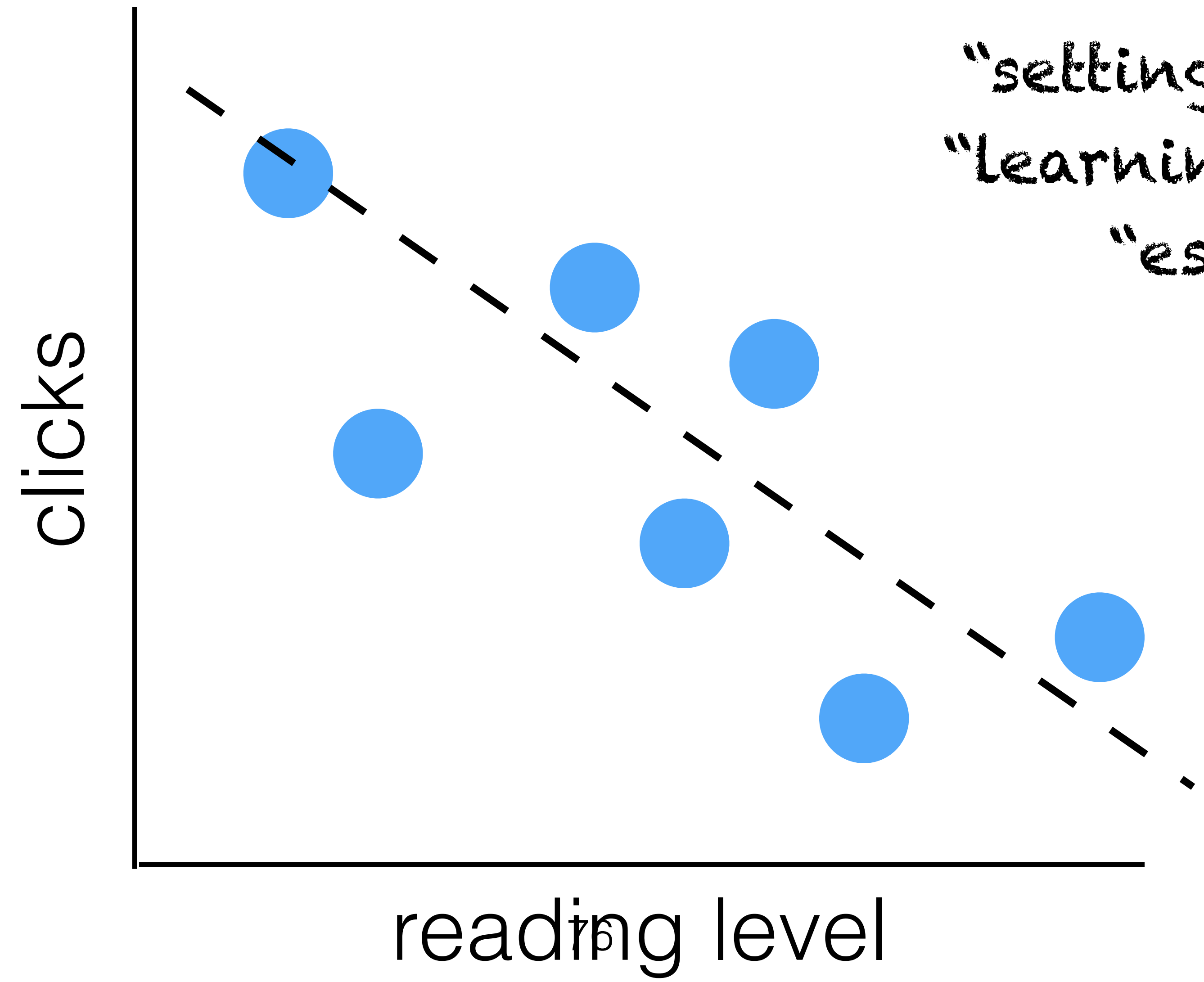
```
clicks = m(reading_level) + b
```

m and b -> The "parameters" which need to be set (by looking at data)



Model

```
clicks = m(reading_level) + b
m = cov(rl, c)/var(rl)
```

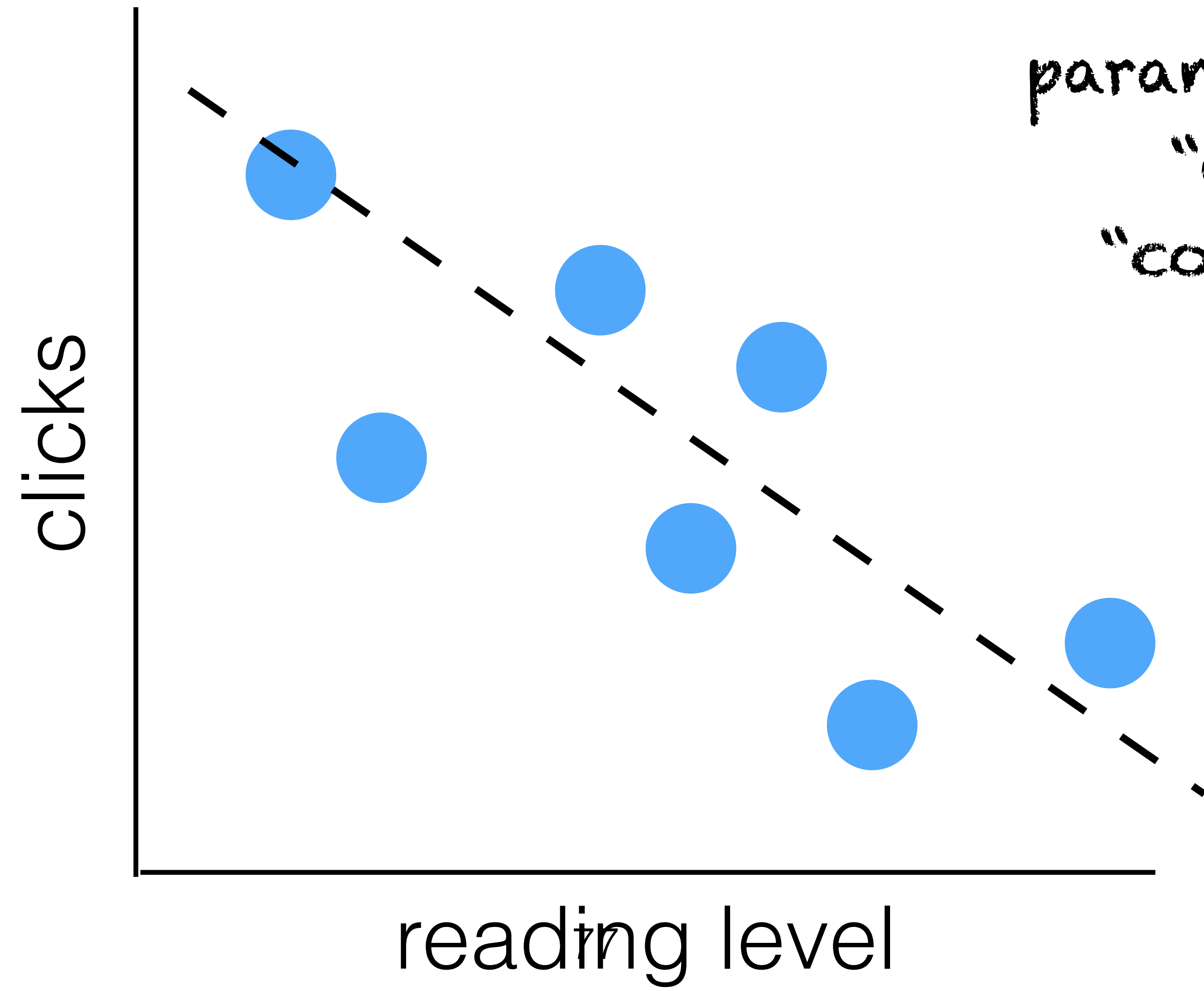


"setting parameters", "learning", "training", "estimation"

Model

`clicks = m(reading_level) + b`

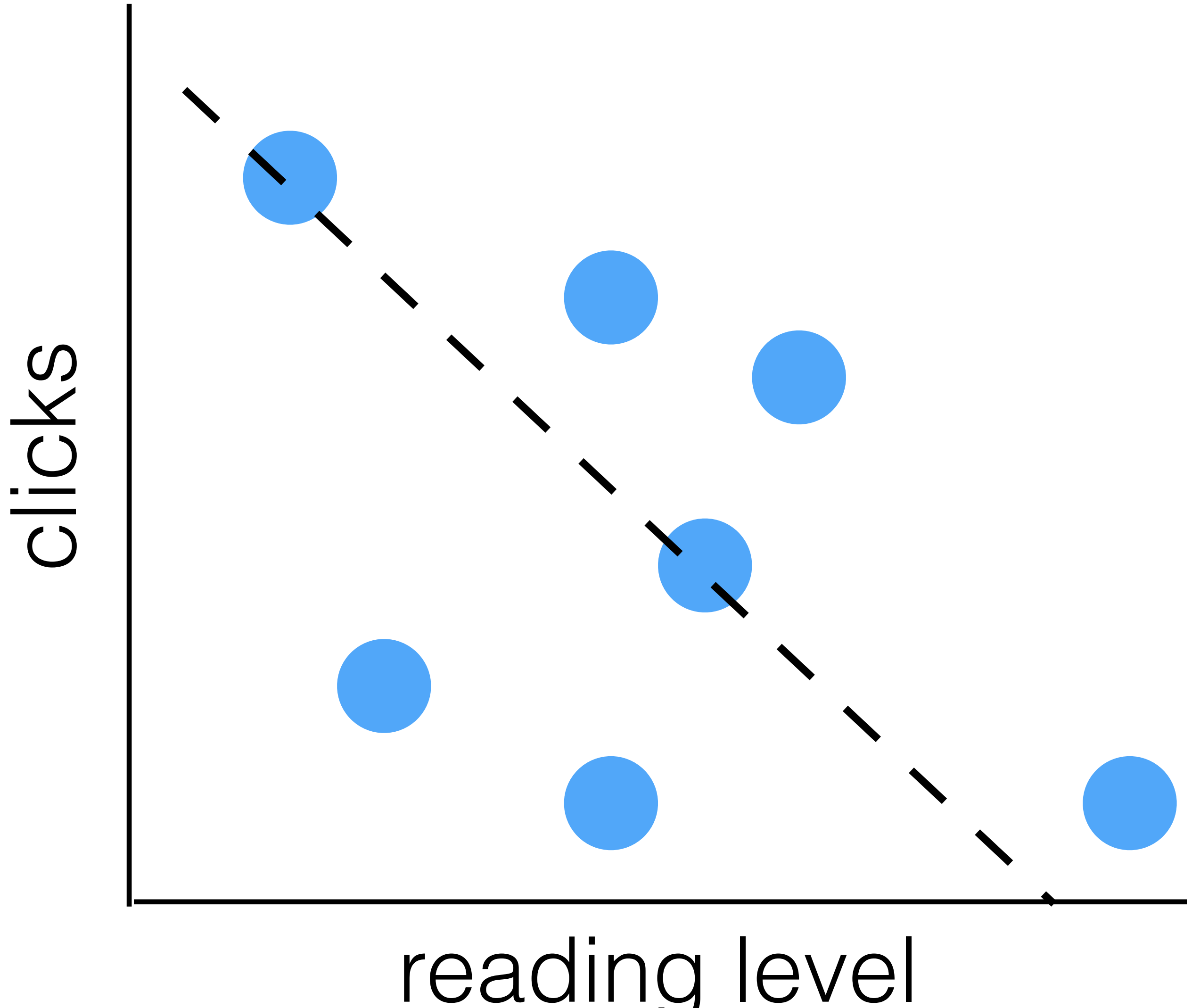
`m = -2.4`



parameter values, "weights", "coefficients"

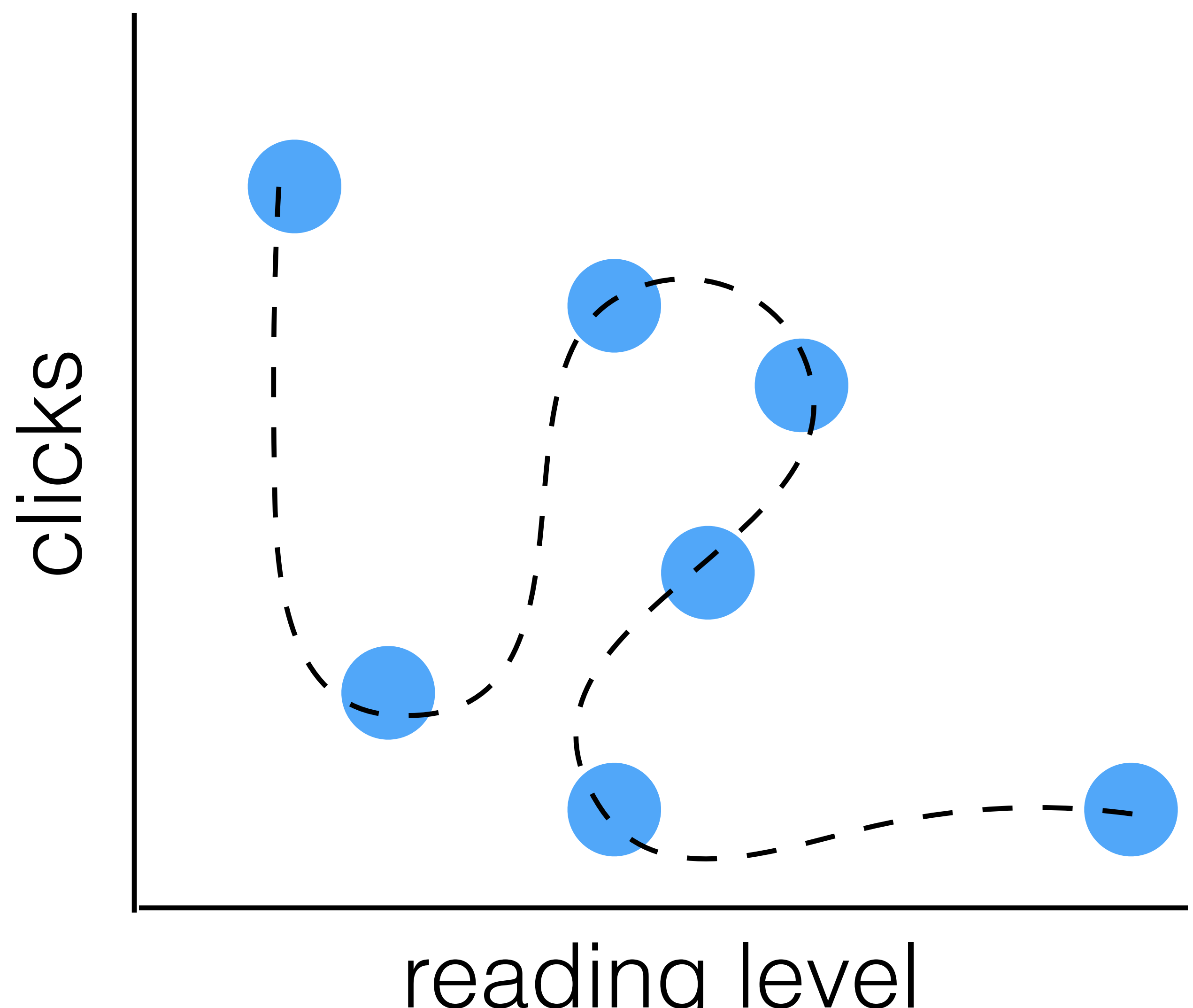
Model

Lots of ways to build in assumptions about problem domain



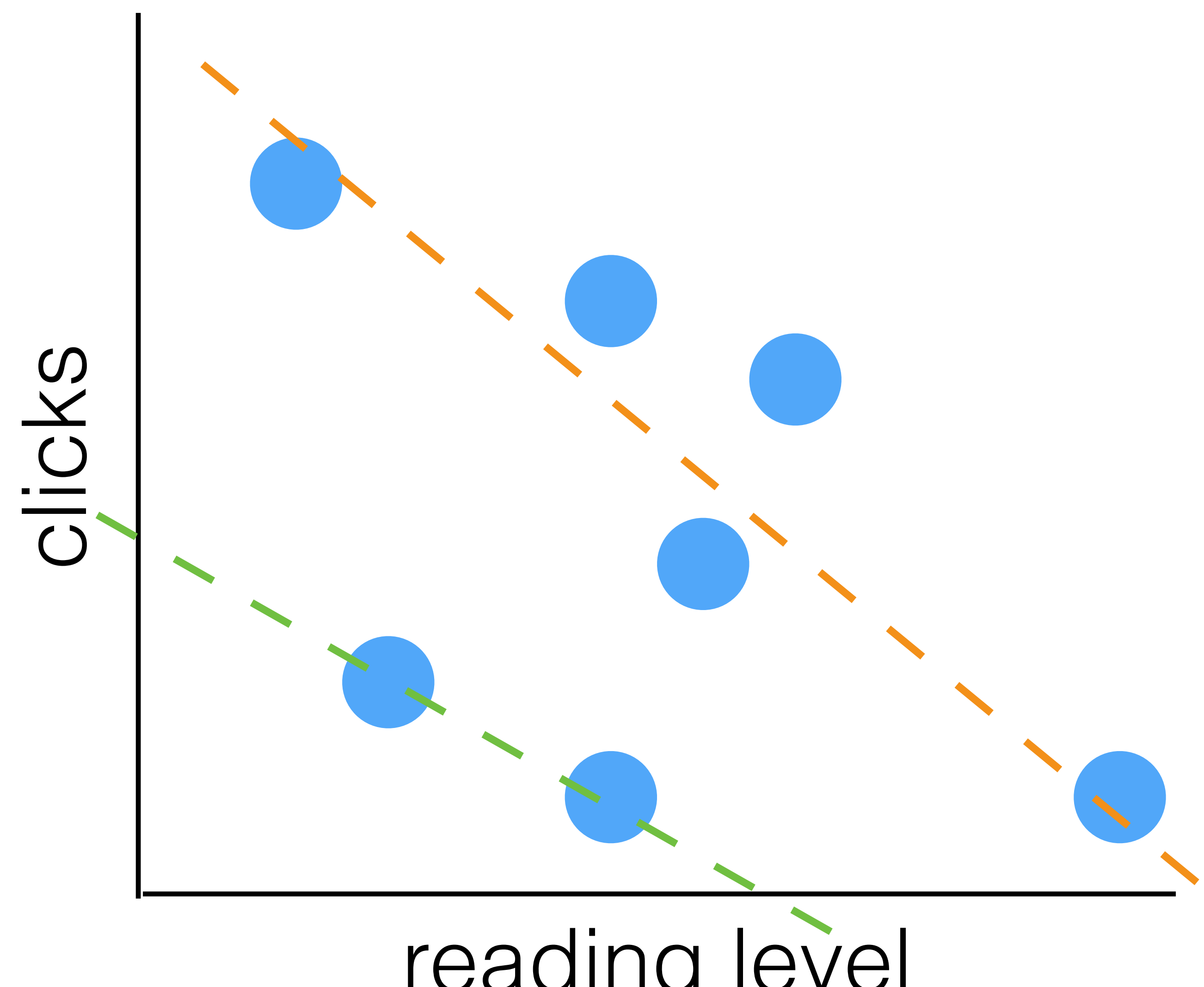
Model

Lots of ways to build in assumptions about problem domain



Model

Lots of ways to build in assumptions about problem domain



P( ) = p

P( ) = 1 - p

Model

#1

- Make assumptions about the problem domain.
- How is the data generated?
- How is the decision-making procedure structured?
- What types of dependencies exist?
- Trending buzzword: “inductive biases”

#2

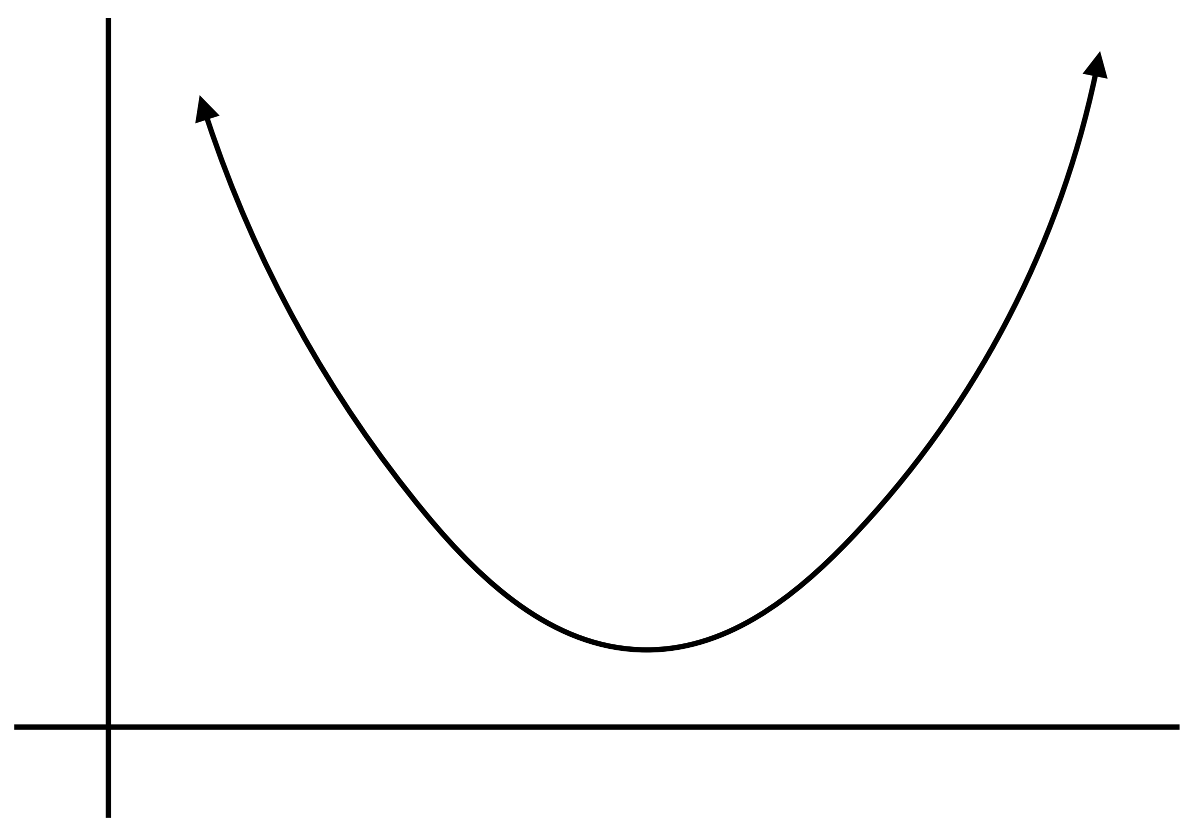
- How to train the model?

Training with Gradient Descent

minimize $$\sum_{i=1}^{n}(Y_i - \hat{Y})^2$$

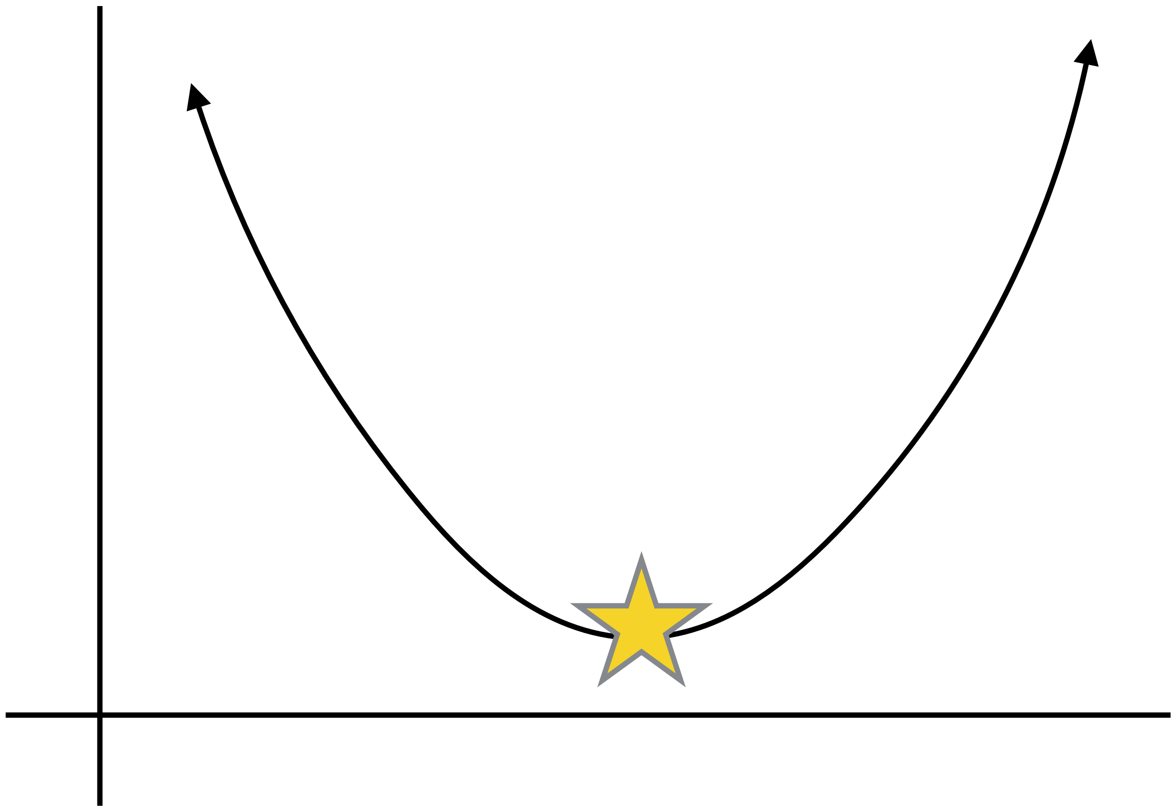
Training with Gradient Descent

minimize $$\sum_{i=1}^{n}(Y_i - \hat{Y})^2$$



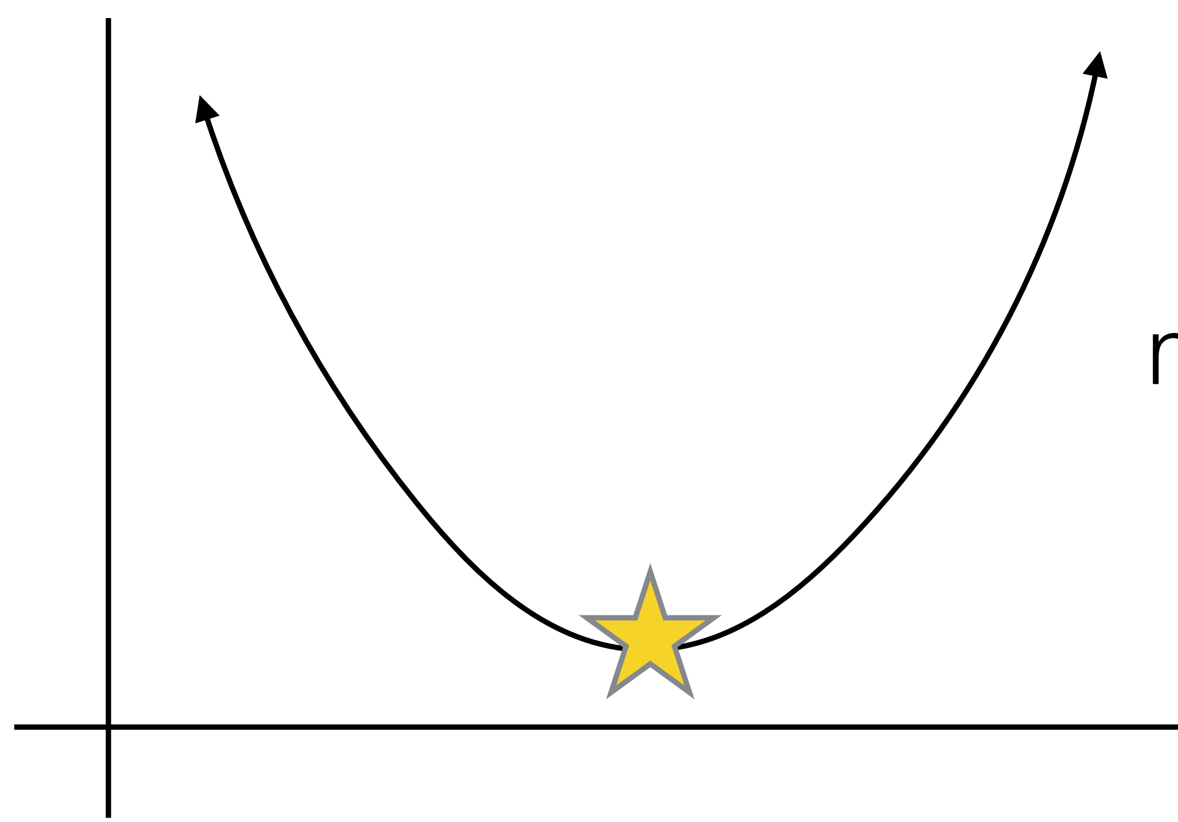
Training with Gradient Descent

minimize $$\sum_{i=1}^{n}(Y_i - \hat{Y})^2$$



Training with Gradient Descent

minimize $$\sum_{i=1}^{n}(Y_i - \hat{Y})^2$$

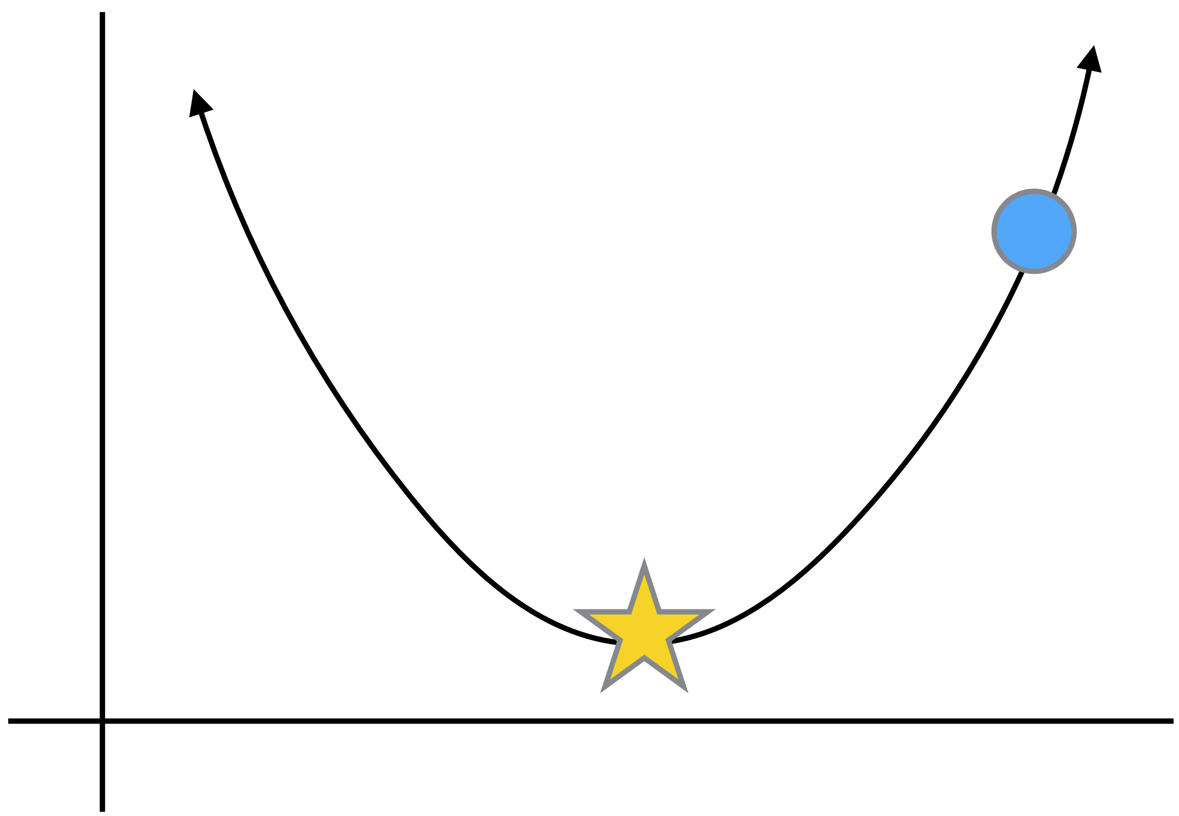


b = $\bar{Y} - m\bar{X}$

m = $\dfrac{Cov(X,Y)}{Var(X)}$

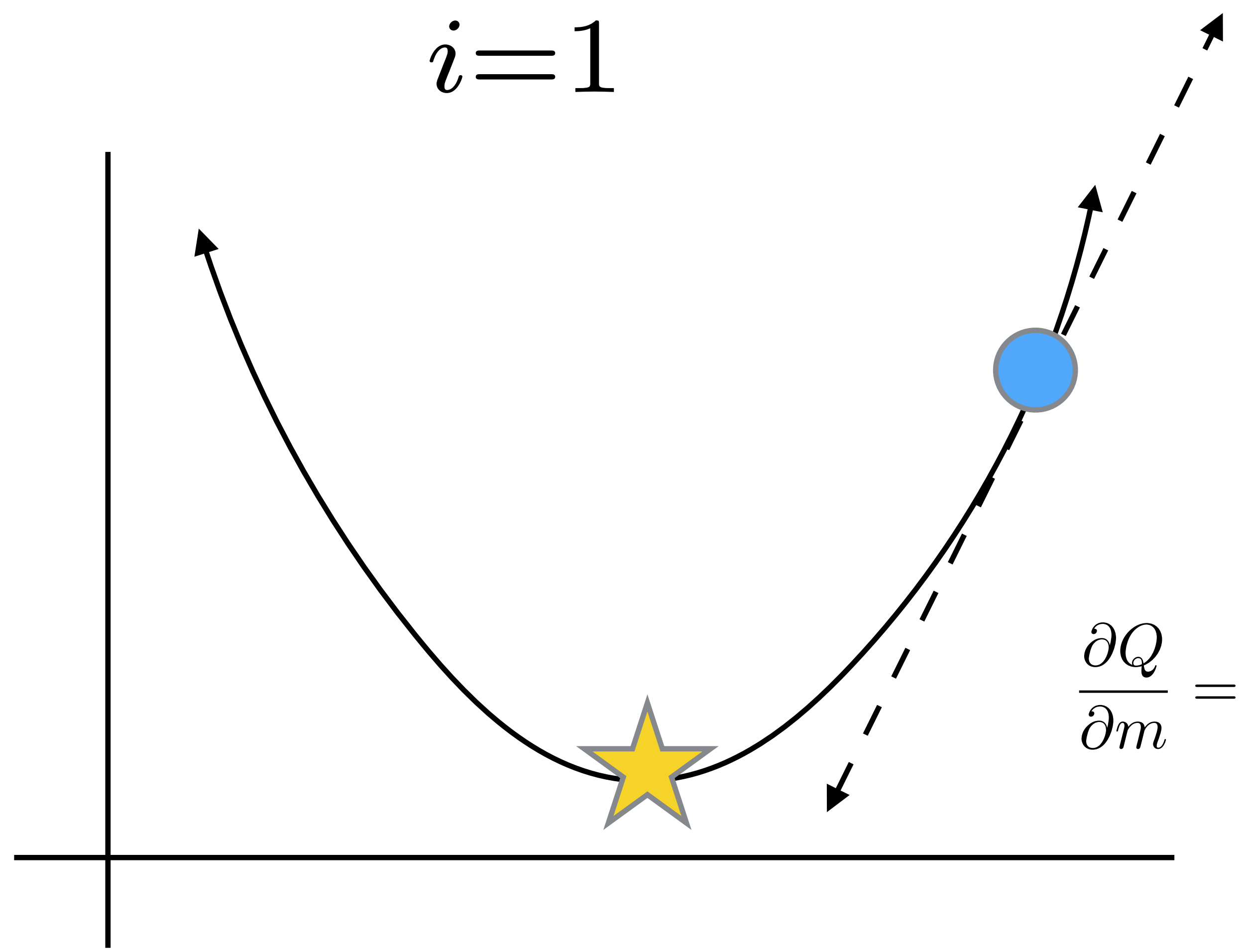
Training with Gradient Descent

minimize $\sum_{i=1}^{n}(Y_i - \hat{Y})^2$



Training with Gradient Descent

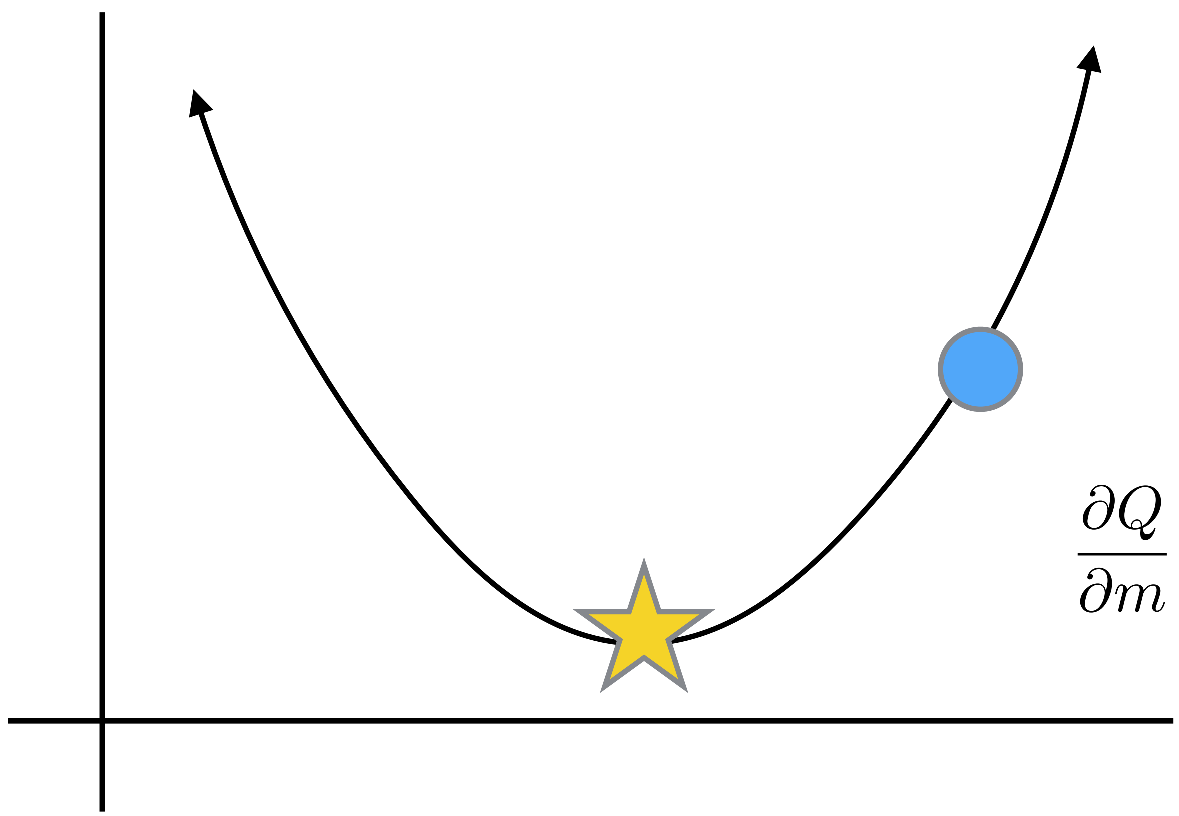
minimize $$\sum_{i=1}^{n}(Y_i - \hat{Y})^2$$



$$\frac{\partial Q}{\partial m} = \sum_{i=1}^{n} -2X_i(Y_i - b - mX_i)$$

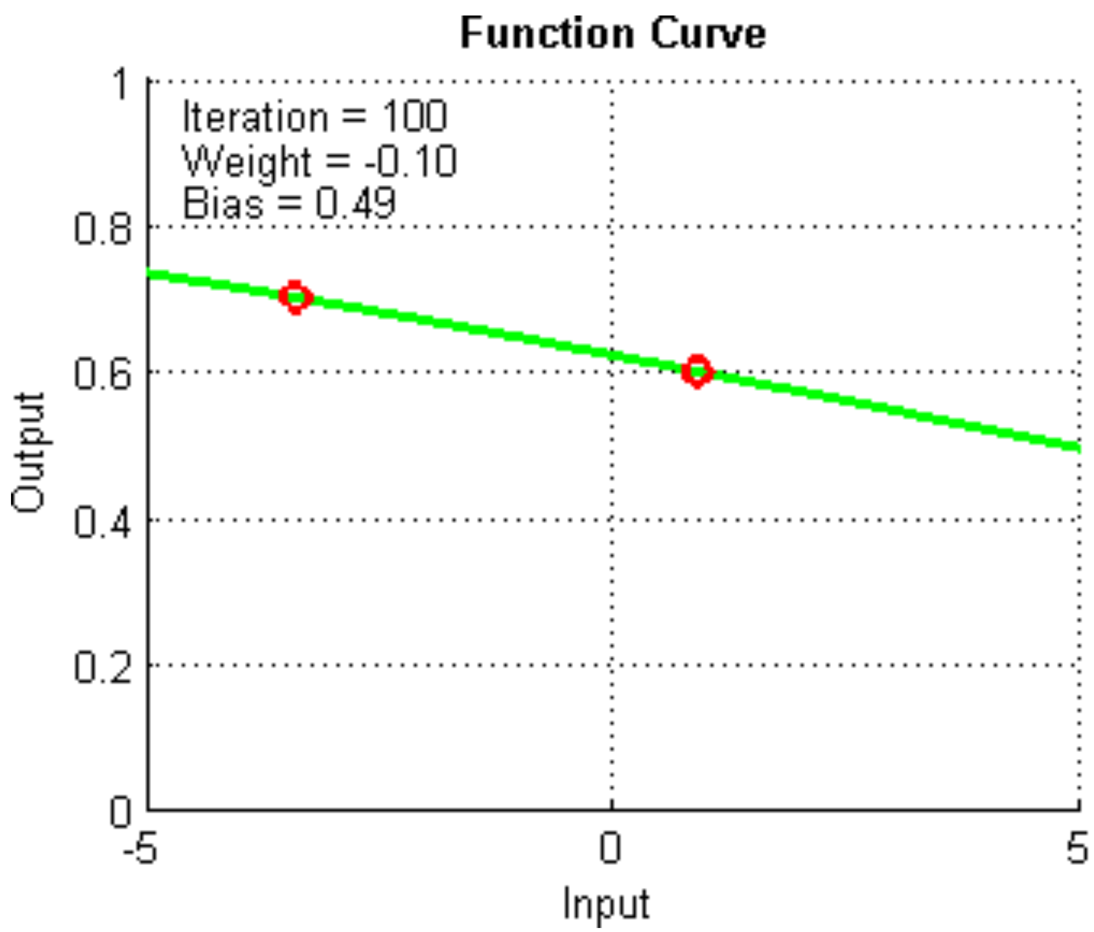
Training with Gradient Descent

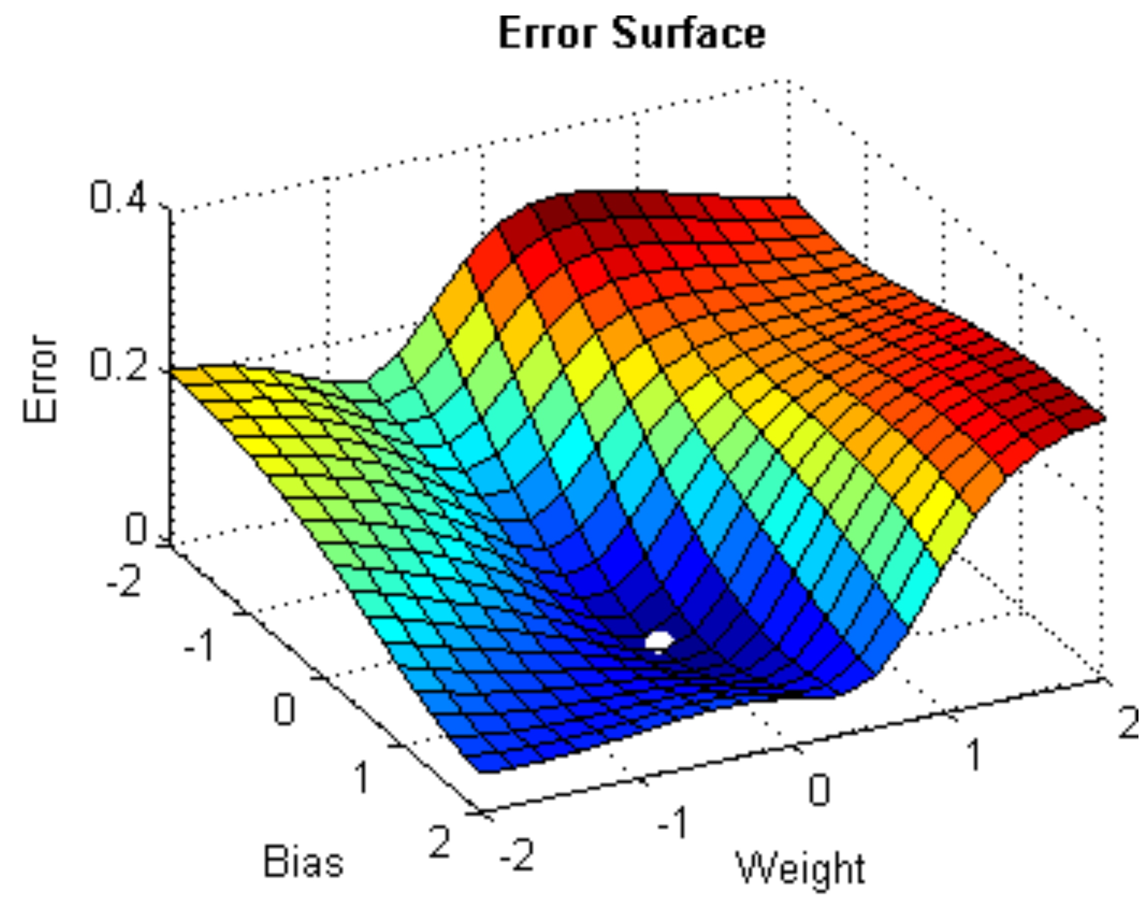
minimize $$\sum_{i=1}^{n}(Y_i - \hat{Y})^2$$



$$\frac{\partial Q}{\partial m} = \sum_{i=1}^{n} -2X_i(Y_i - b - mX_i)$$

Training with Gradient Descent





https://independentseminarblog.com/2018/01/12/moving-below-the-surface-3-gradient-descent-william/

Training with Gradient Descent

Helpful equations for following along in the jupyter notebook

$$Q = \sum_{i=1}^{n} (Y_i - (mX_i + b))^2$$

$$\frac{\partial Q}{\partial b} = \sum_{i=1}^{n} -2(Y_i - mX_i - b) = 0$$

$$\frac{\partial Q}{\partial m} = \sum_{i=1}^{n} -2X_i(Y_i - b - mX_i) = 0$$

$$m = \frac{Cov(X, Y)}{Var(X)} \qquad b = \bar{Y} - m\bar{X}$$

https://independentseminarblog.com/2018/01/12/moving-below-the-surface-3-gradient-descent-william/

👋