The Mathematics of Supervized Learning

What Maths is Needed for ML (& why)

1. Linear Algebra → Datasets & transformations
   Eg. Data Compression
2. Calculus
   ↳ Optimizations → Rates of Change & Areas (...)
   Eg. Using change in loss as guide to optimize estimate
3. Statistics
   ↳ Probability → Populations & Samples
   Description & Inference
   Eg. How well does the sample (i.e. my dataset) represent the population?

Mathematics & Method of Supervised Learning

Maths

- Review SL
- Heart of SL – Optimization
- How to do Optimization well.

1pm

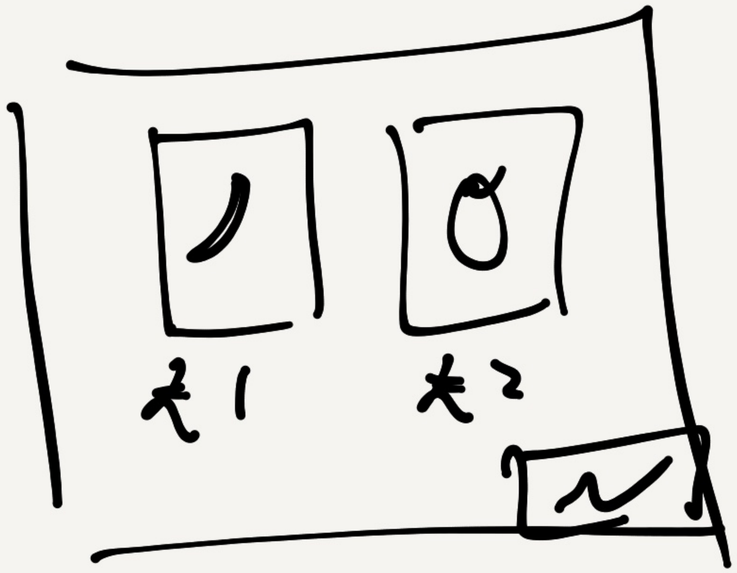
Applications

- How do I choose a ML Model / SL Model?

Review of S.L.

Why ML?

Retail Website



Customer Behavior

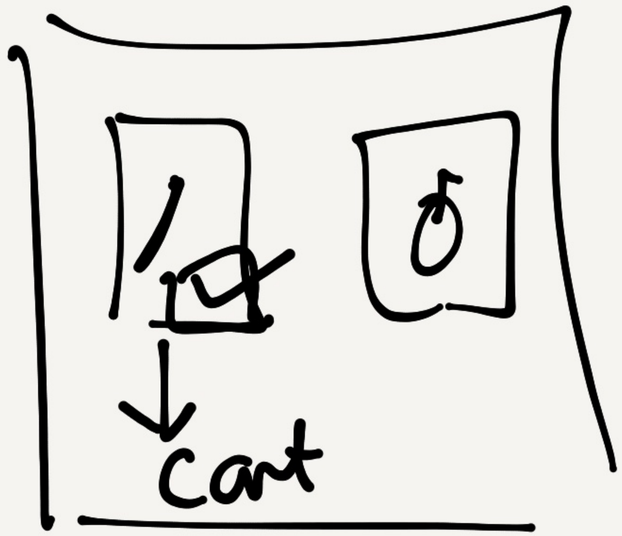
→ orders.

"Improve" the website.

analyze over history

long-term predict trends.

with ML. (#1)



observe customer behavior <u>live</u>

& predict / "Recommend" an associated product.

estimate

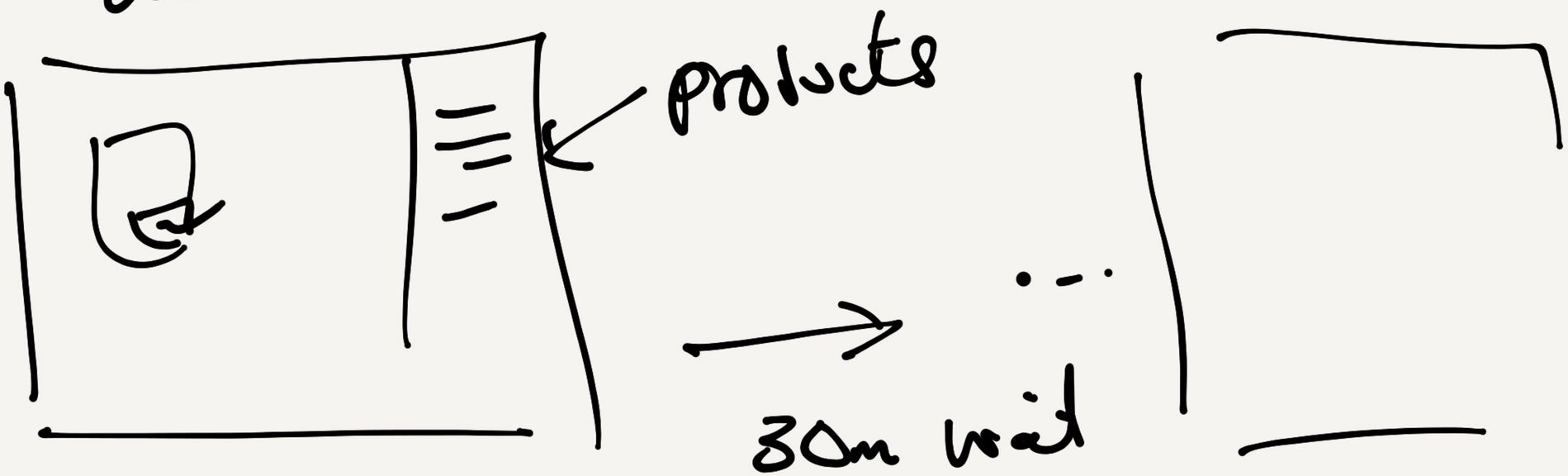
$$\hat{y} = \hat{f}(x_{\text{product-in-cart}})$$

↑

liked product

With ML #2

- CoA abandonment



- "intervene" → sends an email
  → Discount / voucher
  → Chatbot Notification

$$\hat{y} = \hat{f}(x_{\text{time since last interact}})$$

← decision to intervene.

Supervised learning

Goal: $\hat{y} = \hat{f}(\vec{x}; \vec{w})$ s.t. $\underbrace{\hat{f} \text{ "is close" to } f}_{\text{what does this mean?}}$

where:

f is the true relationship

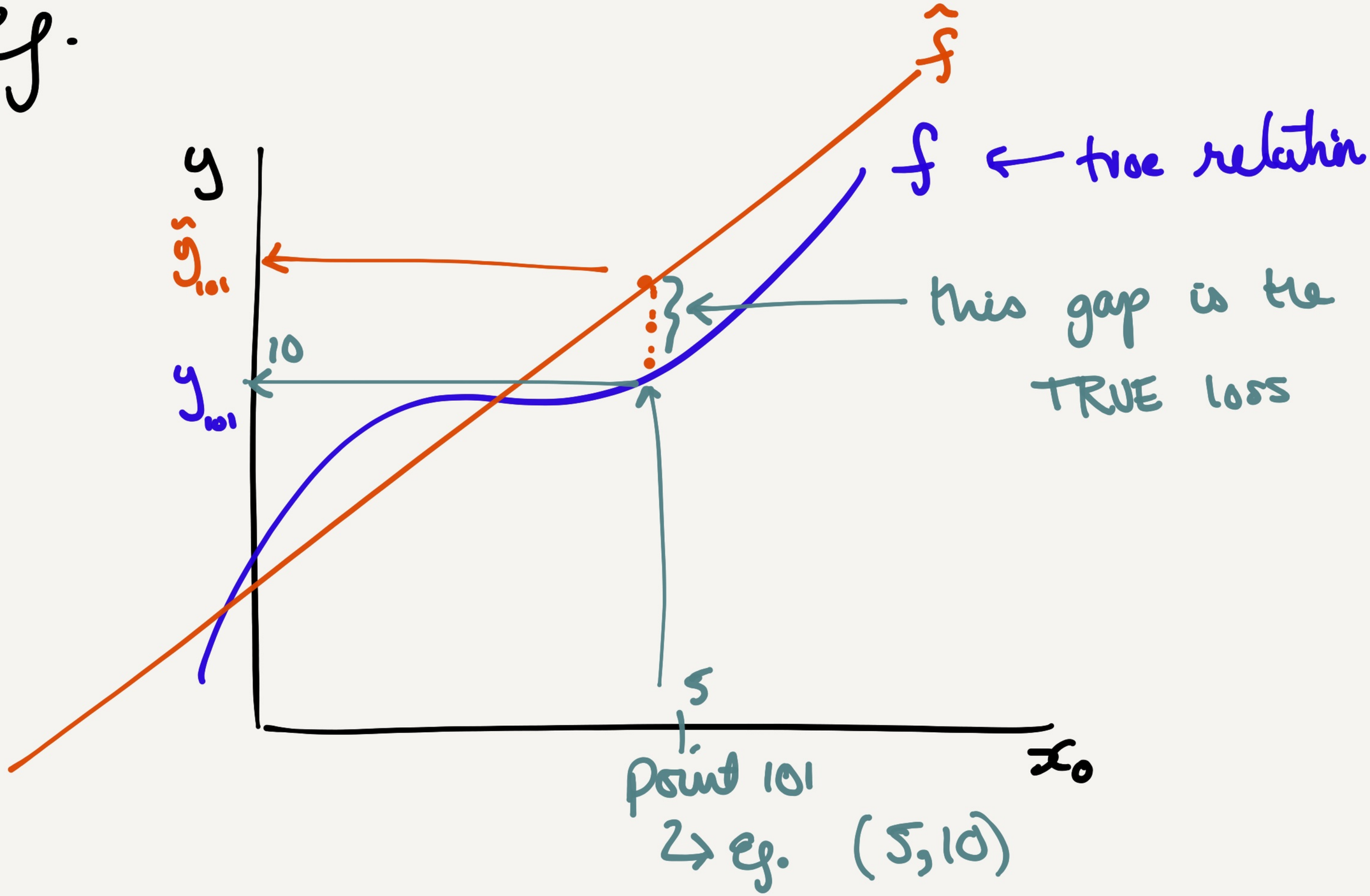
$\hat{f}$ is the estimated "

$\vec{x}$ is a vector of features $x_1 \dots x_n$

y is the true target

$\hat{y}$ is estimated target.

Eg.


$\hat{f}$
f ← true relation
y
$\hat{y}_{101}$
10
y_{101}
this gap is the TRUE loss
5
x_0
point 101
↳ eg. (5,10)

The meaning of $\hat{f}$ "is close to" f.

$y^i = f$: the truth

$\hat{y}^i = \hat{f}$: estimate

if loss <u>decreases</u>

$f \xrightarrow{\text{close}} \hat{f}$

Closeness → given by "Loss"

$$l(\hat{y}, y) = |\hat{y} - y|$$

<u>Total Loss</u>

$$L = \sum_{\text{all points}} l(\hat{y}^i, y^i)$$ ← <u>loss for a point</u>

Eg.

i	x_0	y	$\hat{y}$	l
0	2	5	4	1
1	3	7	6	1
2	4	9	8	1

$\hat{f}(x) = 2x$

$f(x) = 2x + 1$

$L = 3$

"Measure of closeness"

Problem: we don't know f!

ie. what the truth is!

We have: a dataset, $\mathcal{D} = \{(\vec{x}^i, y^i) \dots\}$

observed targets

feature vector

i = index

Aside:

$\rightarrow \underline{\vec{x}^2}$ means $\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$ for point #2

x_2 means All points, feature #2

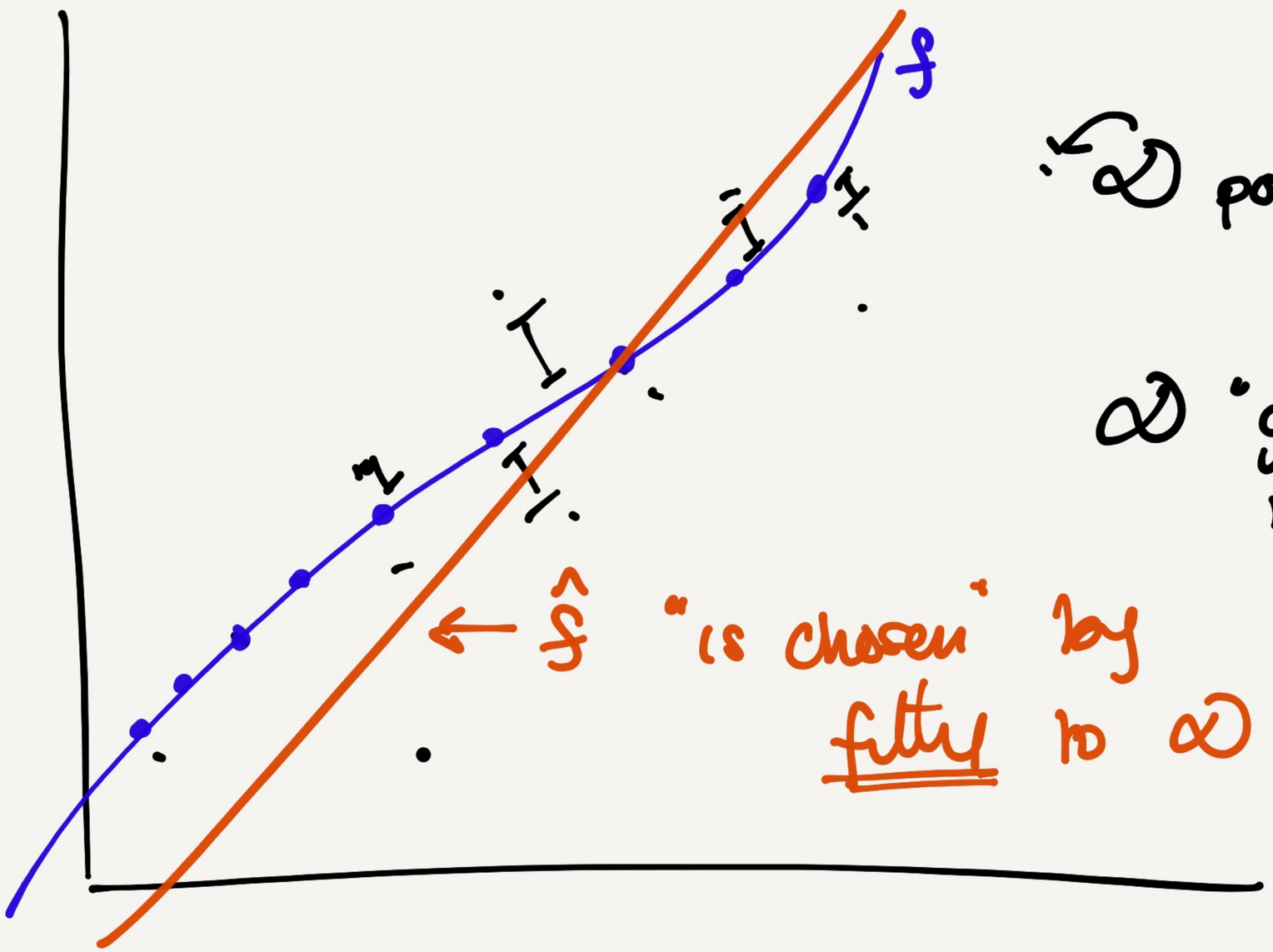
... We hope that $\mathcal{D}$ 'Represents' f

ie. $\boxed{\mathcal{D} \sim \text{like } f}$

Empirical Loss Minimization

from Dataset (vs. the TRUTH)

(i.e. supervised learning)



$\mathcal{D}$ points

$\mathcal{D}$ "good approx" to f

$\hat{f}$ "is chosen" by fitting to $\mathcal{D}$

$$\mathcal{D} = f + \underbrace{\mathcal{N}(0, \text{error})}_{\text{no pattern to the errors.}}$$

assumption of Linear Regression

Loss Minimization

We choose $\hat{f}$ by minimizing the Empirical loss

← Dataset

ie.

$$\hat{y} = \hat{f}(\vec{x}; \vec{w}) \quad \text{st.}$$

$$\boxed{\min \sum_{\mathcal{D}} \ell(\hat{y}^i, y^i)}$$

this is the goal.

How?

goal: $\min \sum_{\mathcal{D}} \ell(\hat{y}^i, y)$

ℓ? $\rightarrow$ $\ell(\hat{f}(x), y) = |\hat{f}(x) - y|$

$\hat{f}$? eg. <u>linear</u> $ax+b$

$$\underbrace{\ell(ax+b, y)}_{\text{Rephrase}} = |ax+b-y|$$

$\mathcal{D} = (x^i, y^i)$

$$\ell(a, b; \mathcal{D}) = |a x^i + b - y^i|$$

variables: a, b

fixed: x^i, y^i

Excel Demo

Summary

We find the best $\hat{f}$ ....

by varying parameters in loss

we get minimal total loss

These parameters are the "best"

& ∴ the solution model

$$f(x; \underbrace{a, b}_{\text{"Core of Model"}}) = ax + b$$

Kernel

we get a particular "optimal" (a, b)

$$f(x, a^*, b^*) = \underline{a}^* x + \underline{b}^*$$

Python Demo

Next Question:

How do machines optimize?

(w/o Excel & graphs....)