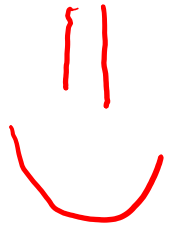
Warm-up as you log in



What are these linear shapes called for 1-D, 2-D, 3-D, M-D input?

$\boldsymbol{x} \in \mathbb{R}$ $\quad$ $\boldsymbol{x} \in \mathbb{R}^2$ $\quad$ $\boldsymbol{x} \in \mathbb{R}^3$ $\quad$ $\boldsymbol{x} \in \mathbb{R}^M$

$$y = \boldsymbol{w}^T\boldsymbol{x} + b$$

$$\boldsymbol{w}^T\boldsymbol{x} + b = 0$$

$$\boldsymbol{w}^T\boldsymbol{x} + b \geq 0$$

Announcements

Assignments

- HW2
  - Due Mon, 9/21, 11:59 pm
- HW3
  - Out tomorrow, due Mon, 9/28, 11:59 pm
  - Written, but in Gradescope

Midterm 1

- Mon, 10/5
- In lecture; Gradescope exam (like HW1 written); proctored via Zoom
- Content up to and including linear regression and optimization
- Stay tuned to Piazza for details and a few forms to fill out

Plan

Last time

- Model selection
  - Parameters, Hyperparameters
  - Train, Test, and Validation sets

Today

- A few more things on model selection
- Regression
- Linear regression
- Optimization for linear regression

Introduction to Machine Learning

Linear Regression and Optimization

Instructor: Pat Virtue

Model Selection

- Two *very* similar definitions:
  - *Def*: **model selection** is the process by which we choose the "best" model from among a set of candidates
  - *Def*: **hyperparameter optimization** is the process by which we choose the "best" hyperparameters from among a set of candidates **(could be called a special case of model selection)**
- **Both** assume access to a function capable of measuring the quality of a model
- **Both** are typically done "outside" the main training algorithm --- typically training is treated as a black box

Slide credit: CMU MLD Matt Gormley

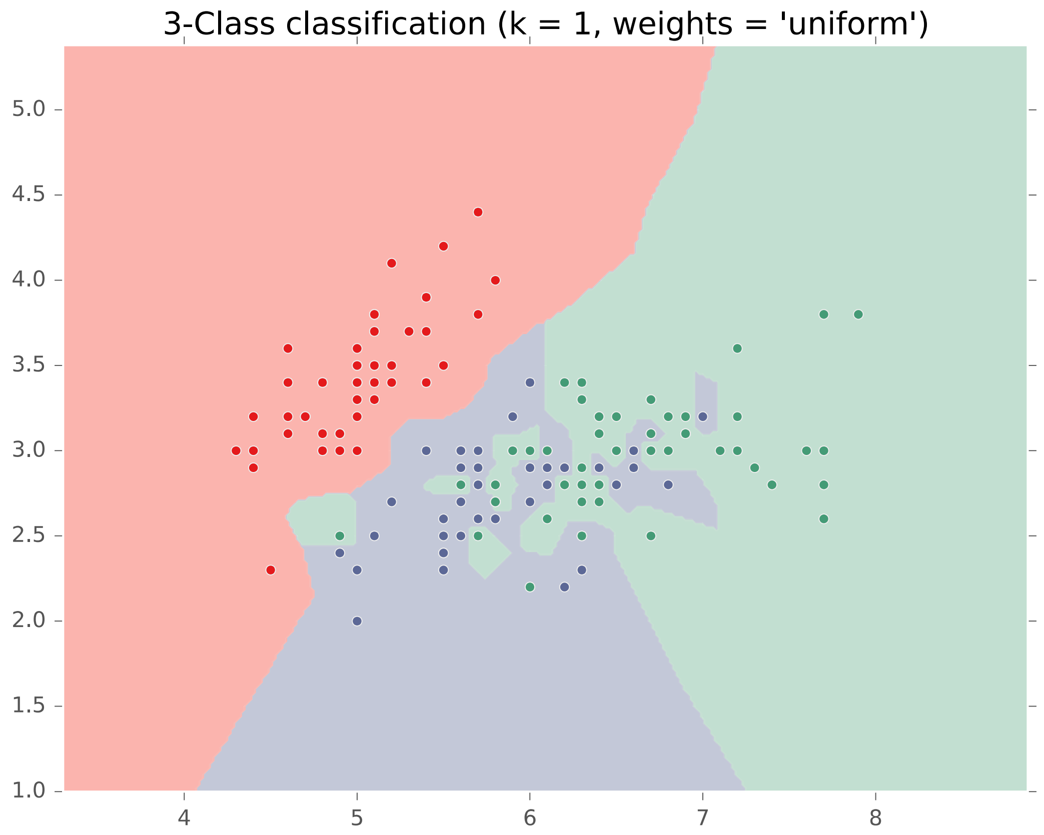
Experimental Design

	Input	Output	Notes
Training	• training dataset • hyperparameters	• best model parameters	We pick the best model parameters by learning on the training dataset for a fixed set of hyperparameters
Hyperparameter Optimization	• training dataset • validation dataset	• best hyperparameters	We pick the best hyperparameters by learning on the training data and evaluating error on the validation error
Testing	• test dataset • hypothesis (i.e. fixed model parameters)	• test error	We evaluate a hypothesis corresponding to a decision rule with fixed model parameters on a test dataset to obtain test error

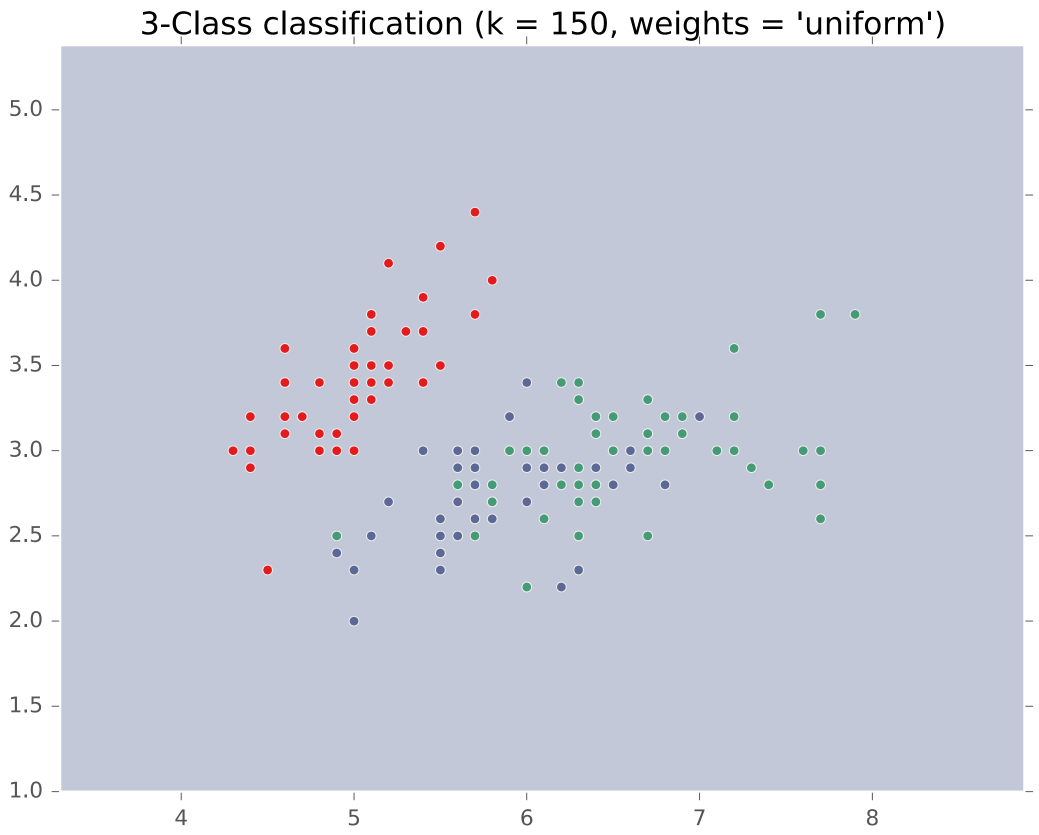
Slide credit: CMU MLD Matt Gormley

Special Cases of k-NN

k=1: Nearest Neighbor



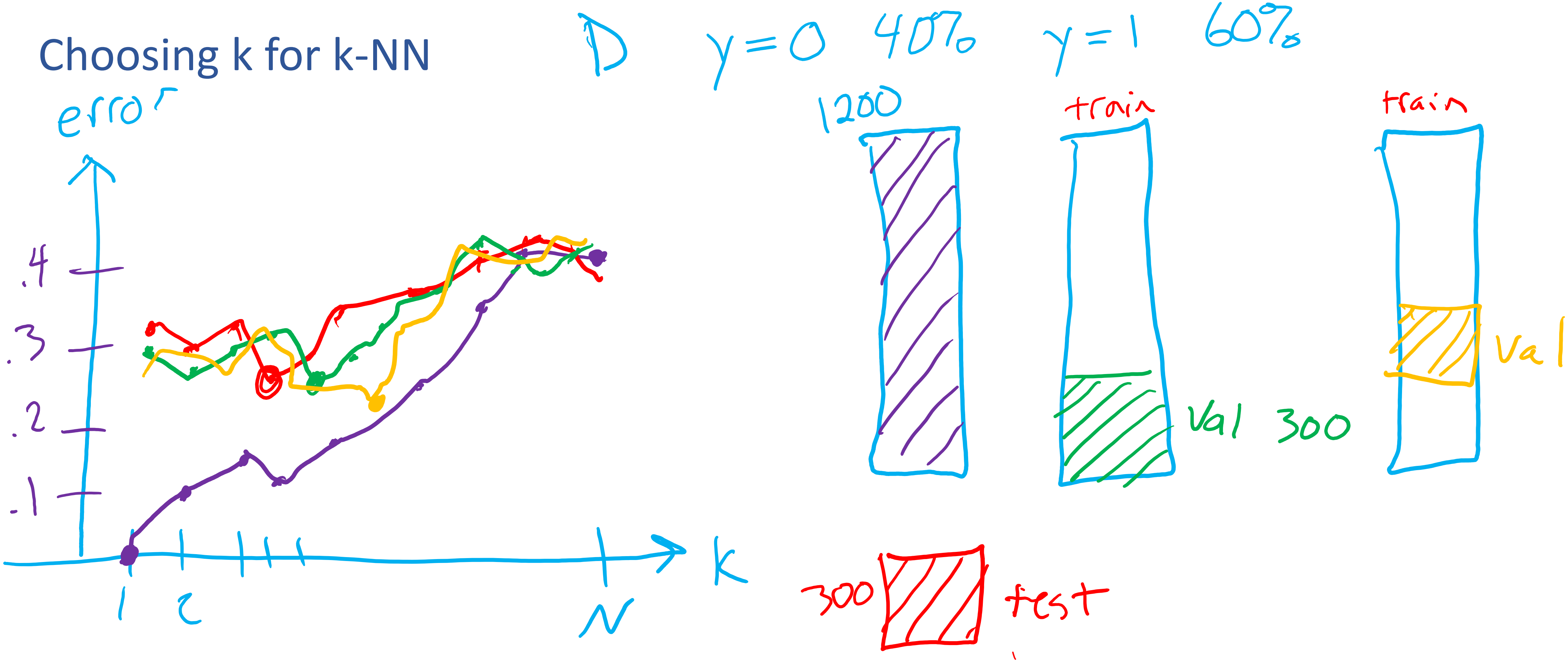
k=N: Majority Vote



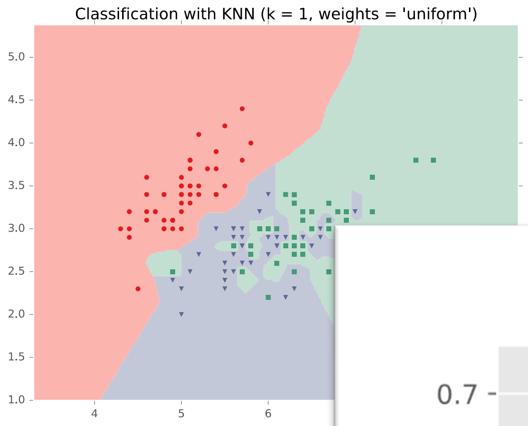
Slide credit: CMU MLD Matt Gormley

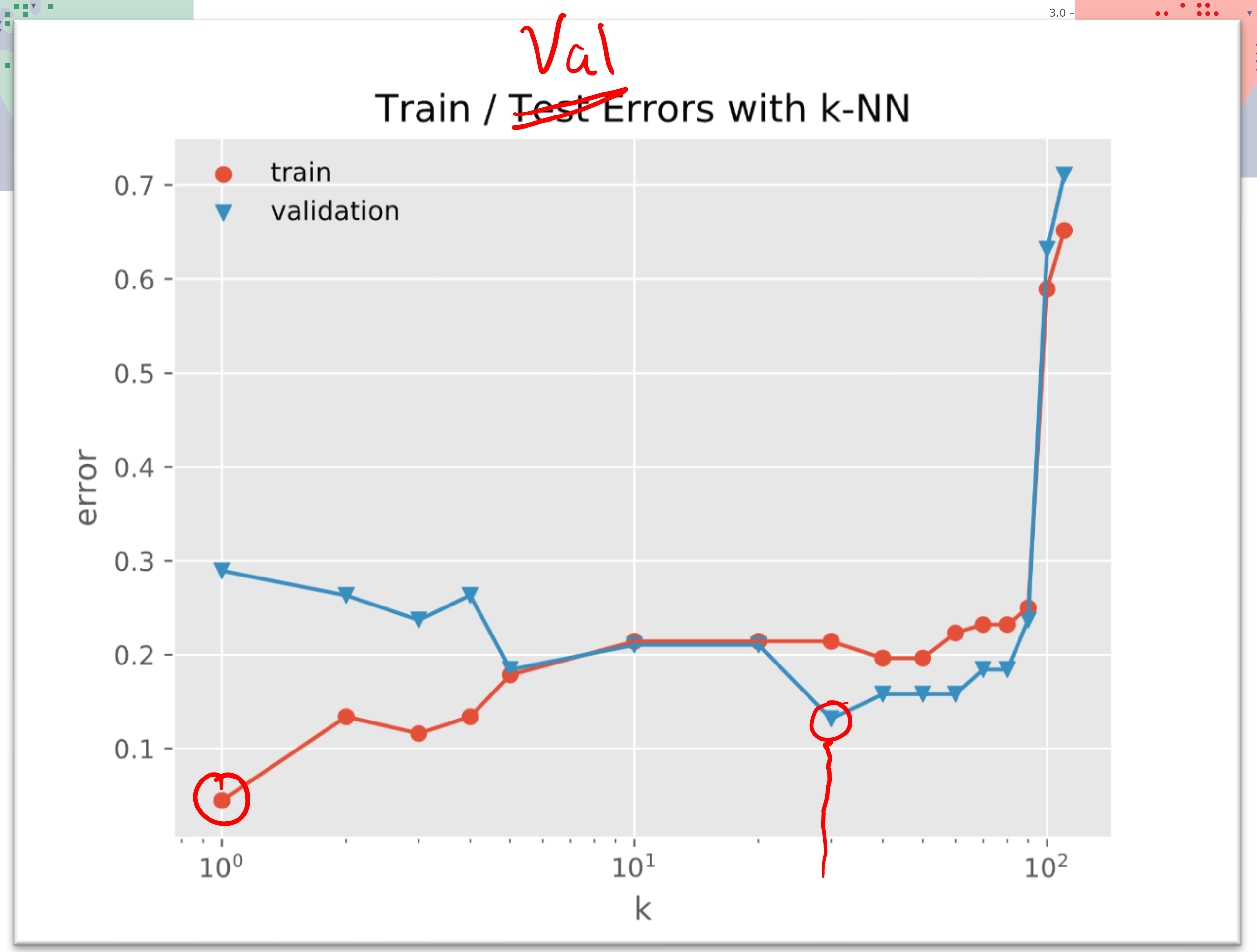
Example of Hyperparameter Optimization

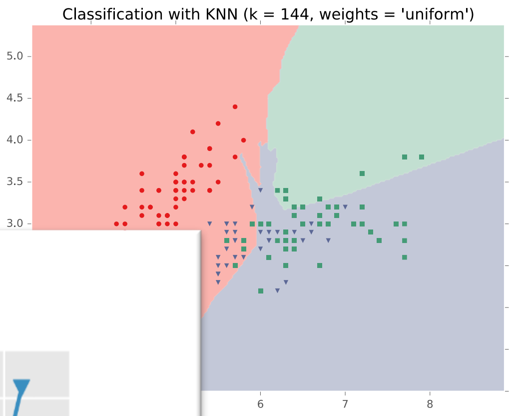
Choosing k for k-NN



k-NN: Choosing k



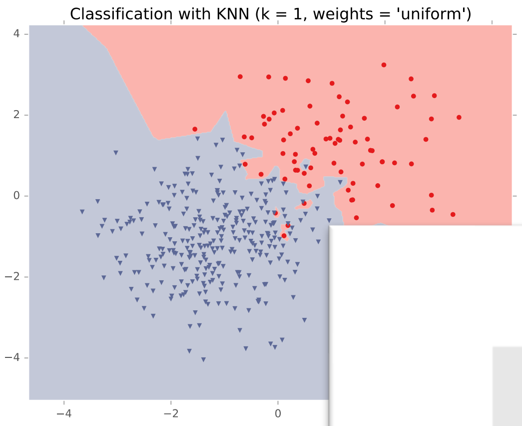


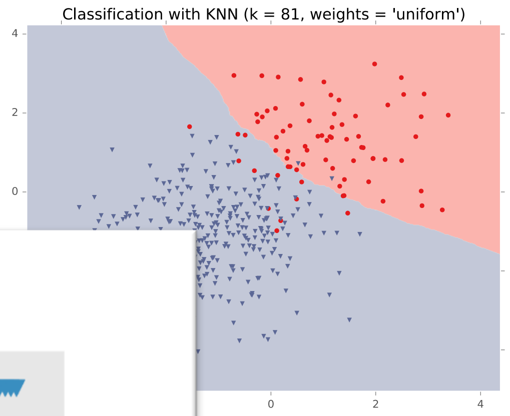


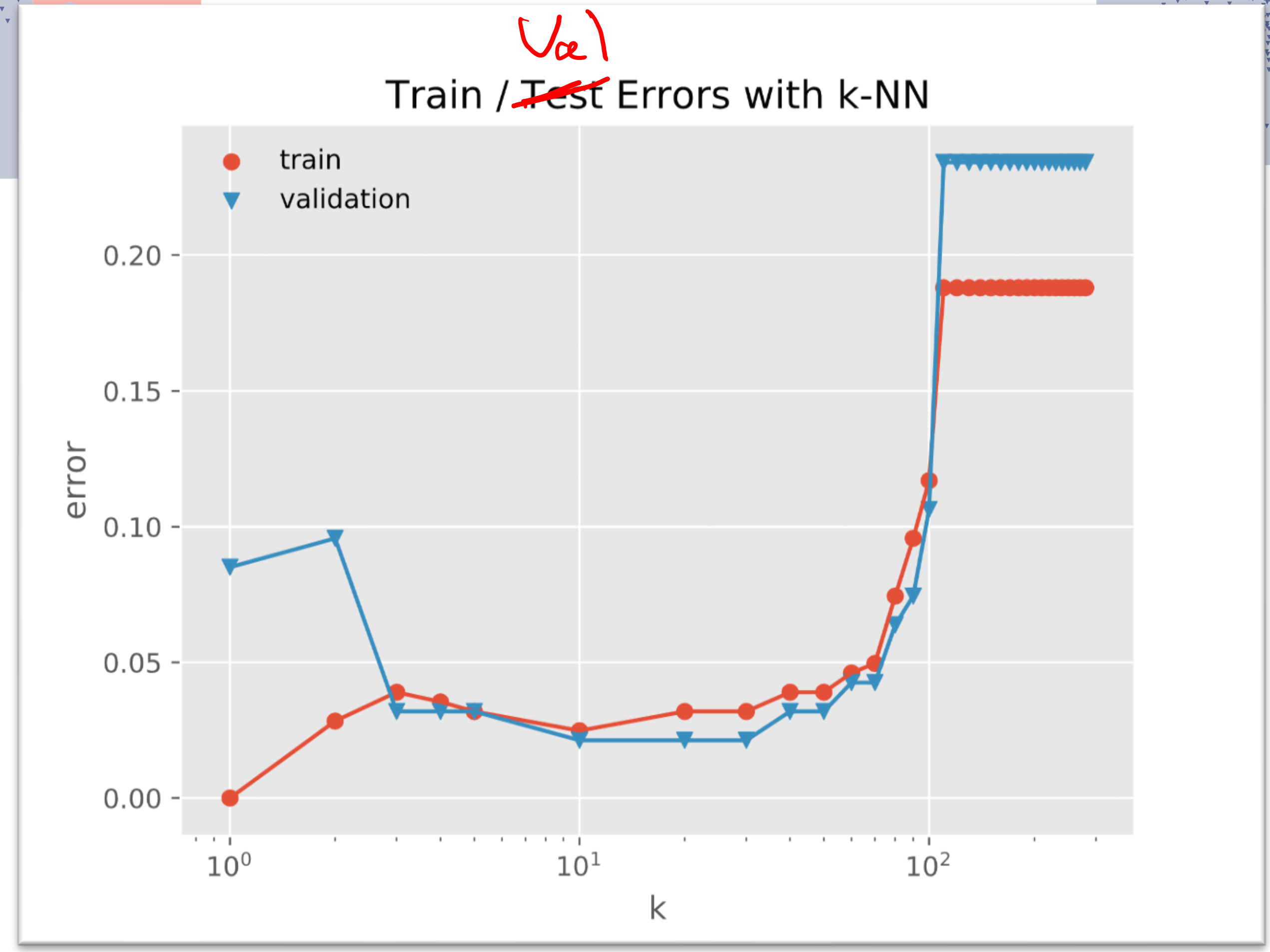
Fisher Iris Data: varying the value of k

Slide credit: CMU MLD Matt Gormley

k-NN: Choosing k







Gaussian Data: varying the value of k

Slide credit: CMU MLD Matt Gormley

Validation

Why do we need validation?

- Choose hyperparameters
- Choose technique
- Help make any choices beyond our parameters

But now, we have another choice to make!

- How do we split training and validation?

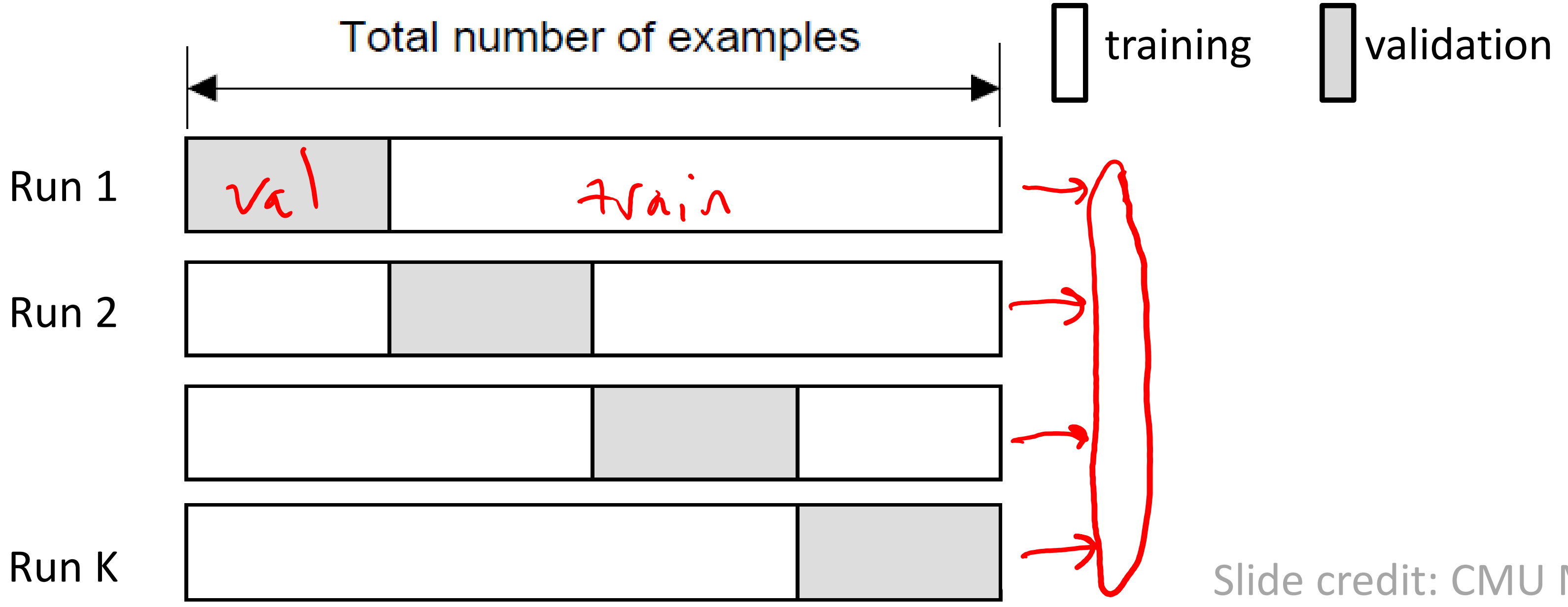
Trade-offs

- More held-out data, better meaning behind validation numbers
- More held-out data, less data to train on!

Cross-validation

K-fold cross-validation

Create K-fold partition of the dataset.
Do K runs: train using K-1 partitions and calculate validation error on remaining partition (rotating validation partition on each run).
Report average validation error

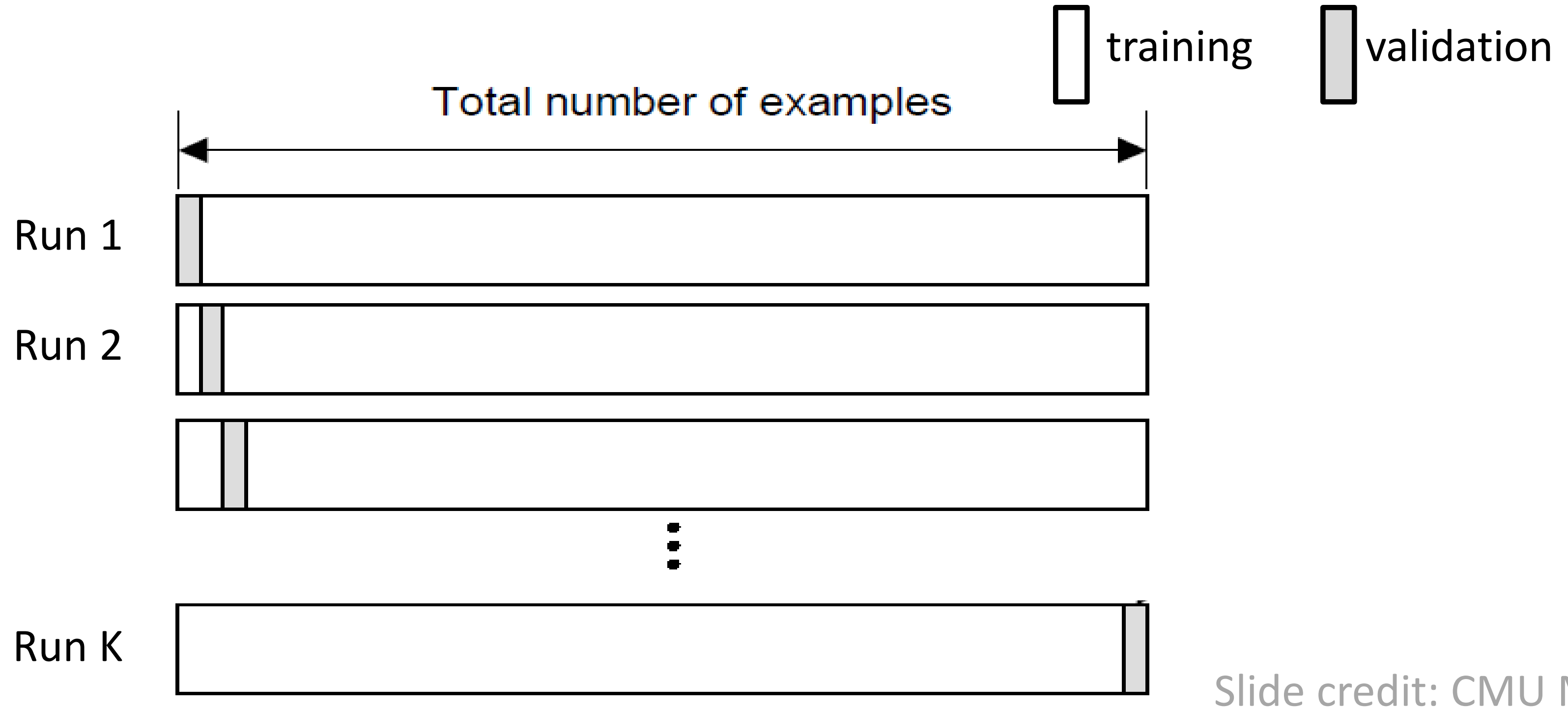


Slide credit: CMU MLD Aarti Singh

Cross-validation

Leave-one-out (LOO) cross-validation

Special case of K-fold with K=N partitions
Equivalently, train on N-1 samples and validate on only one sample per run for N runs



Slide credit: CMU MLD Aarti Singh

Cross-validation

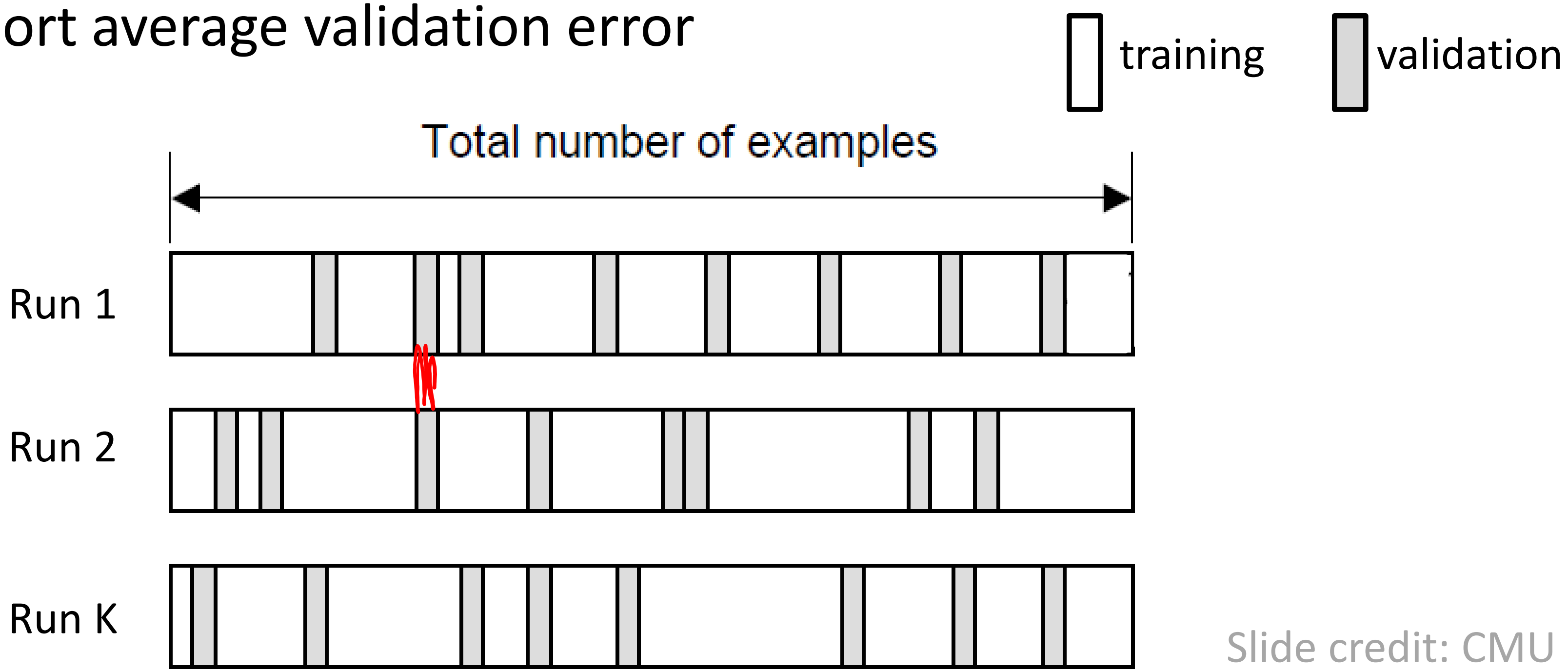
Random subsampling

Randomly subsample a fixed fraction αN ($0 < \alpha < 1$) of the dataset for validation.

Compute validation error with remaining data as training data.

Repeat K times

Report average validation error



Slide credit: CMU MLD Aarti Singh

Practical Issues in Cross-validation

How to decide the values for K and α ?

- Large K
  + Validation error can approximate test error well
  - Observed validation error will be unstable (few validation pts)
  - The computational time will be very large as well (many experiments)

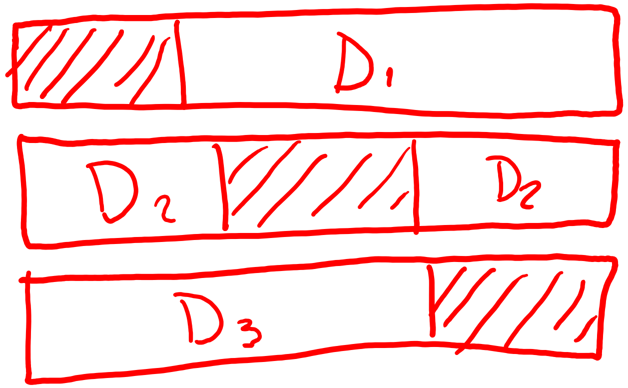
- Small K
  + The # experiments and, therefore, computation time are reduced
  + Observed validation error will be stable (many validation pts)
  - Validation error cannot approximate test error well

Common choice: K = 10, α = 0.1 ☺

Piazza Poll 1

0, 0.01, 0.02 ...

Say you are choosing amongst 10 discrete values of a decision tree *mutual information threshold*, and you want to do K=10-fold cross-validation.



How many times do I have to train my model?

A. 0
B. 1
C. 10 30%
D. 20
E. 100 50% → 60%
F. 10^{10}

for thresh in ______
for D_k in $D_1 \ldots D_K$
train(D_k, thresh)
cv error
best thresh

Piazza Poll 1

Say you are choosing amongst 10 discrete values of a decision tree *mutual information threshold*, and you want to do K=10-fold cross-validation.

How many times do I have to train my model?

A. 0
B. 1
C. 10
D. 20
E. 100
F. 10^{10}

Model Selection

WARNING (again):

– This section is only scratching the surface!
– Lots of methods for hyperparameter optimization: (to talk about later)
  - Grid search
  - Random search
  - Bayesian optimization
  - Graduate-student descent
  - …

Main Takeaway:

– Model selection / hyperparameter optimization is just another form of learning

Model Selection Learning Objectives

You should be able to…

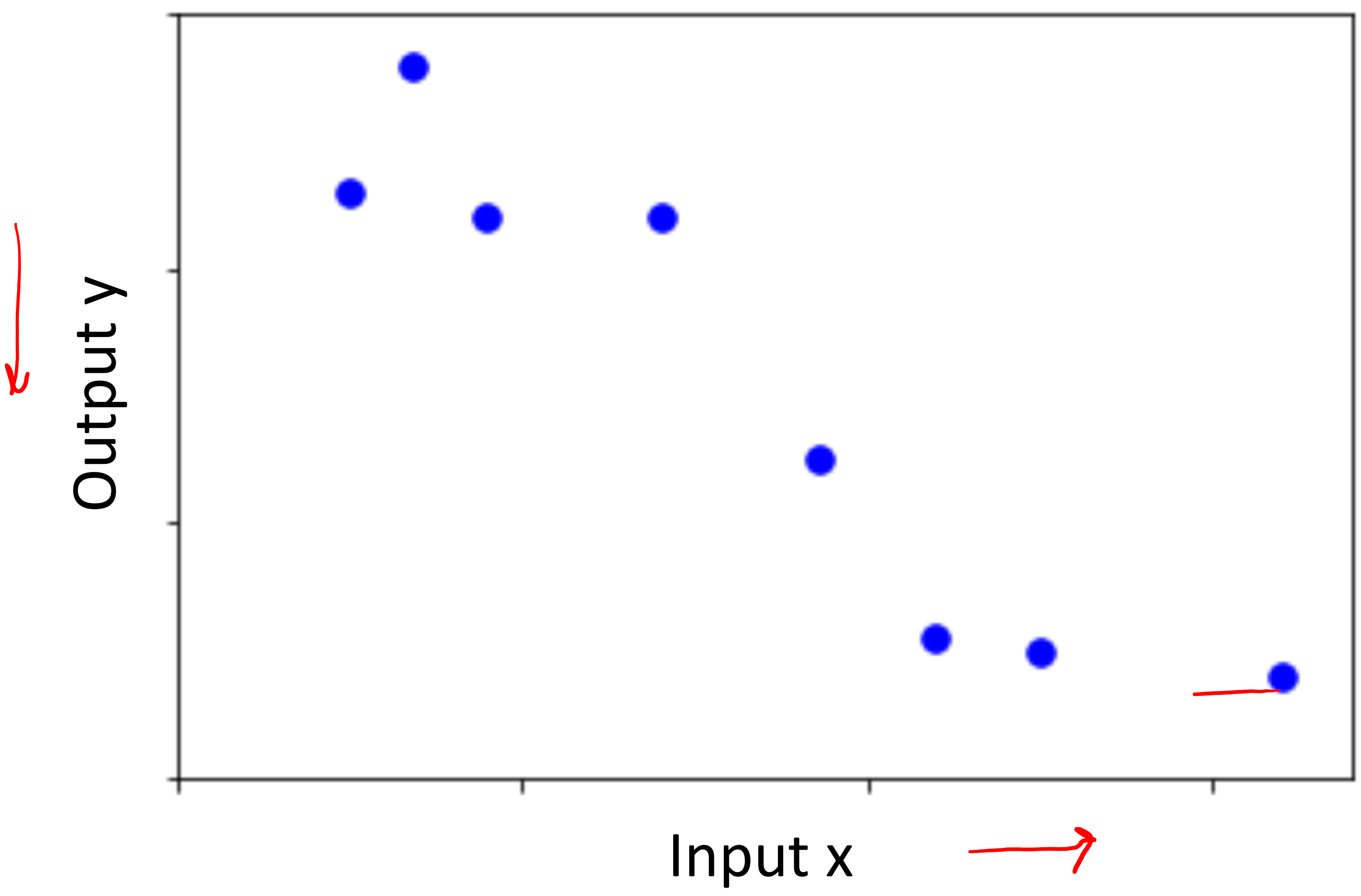
- Plan an experiment that uses training, validation, and test datasets to predict the performance of a classifier on unseen data (without cheating)
- Explain the difference between (1) training error, (2) validation error, (3) cross-validation error, (4) test error, and (5) true error
- For a given learning technique, identify the model, learning algorithm, parameters, and hyperparamters

LINEAR REGRESSION AND OPTIMIZATION

Breakout Room

In your breakout room

Come up with a story for this data



Lecture 2: Problem Formulation

Experience $\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$ $x \in \mathcal{X}$ $y \in \mathcal{Y}$ } Train Data / Test Data

$y^{(i)}$: output, class label, class, label

$x^{(i)}$: input, features, attributes

Regression: $y \in \mathbb{R}$

Classification: y discrete (and not ordered) set

Hypothesis

function: $\hat{y} = h(x)$ $h: \mathcal{X} \to \mathcal{Y}$

Hypothesis space $h \in \mathcal{H}$ $\hat{y} = \underline{m}x + \underline{b}$

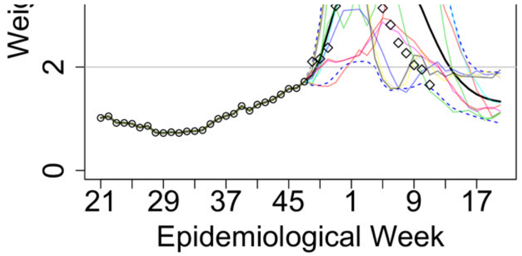
Performance measure

Error

Loss

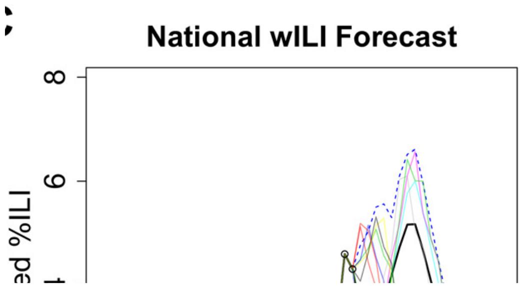
Objective function

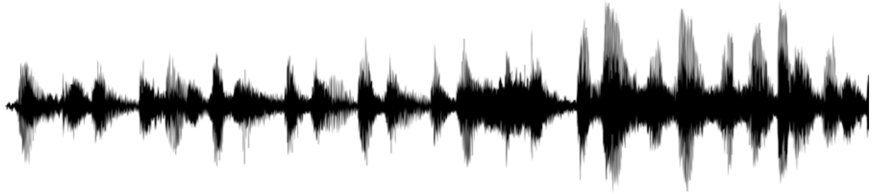
Regression



Goal:

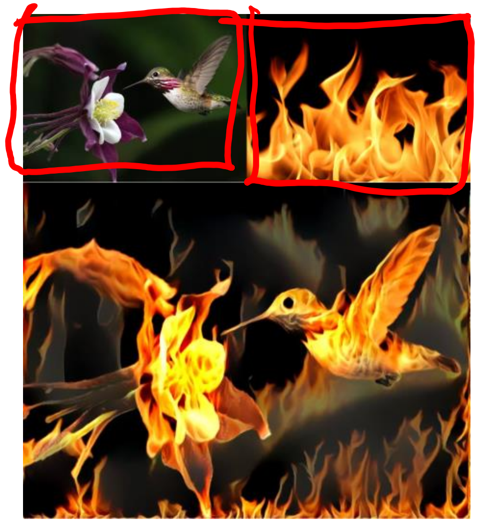
- Given a training dataset of pairs (x,y) where
  - y is a continuous, rather than a label
- Learn a function (aka. curve or line) $\hat{y} = h(x)$ that best fits the training data



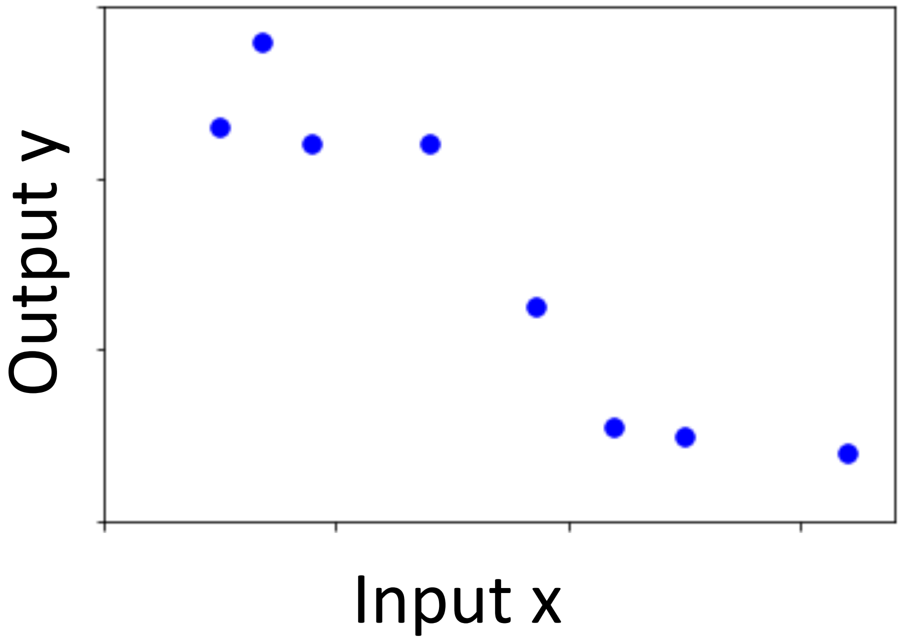


Example Applications:

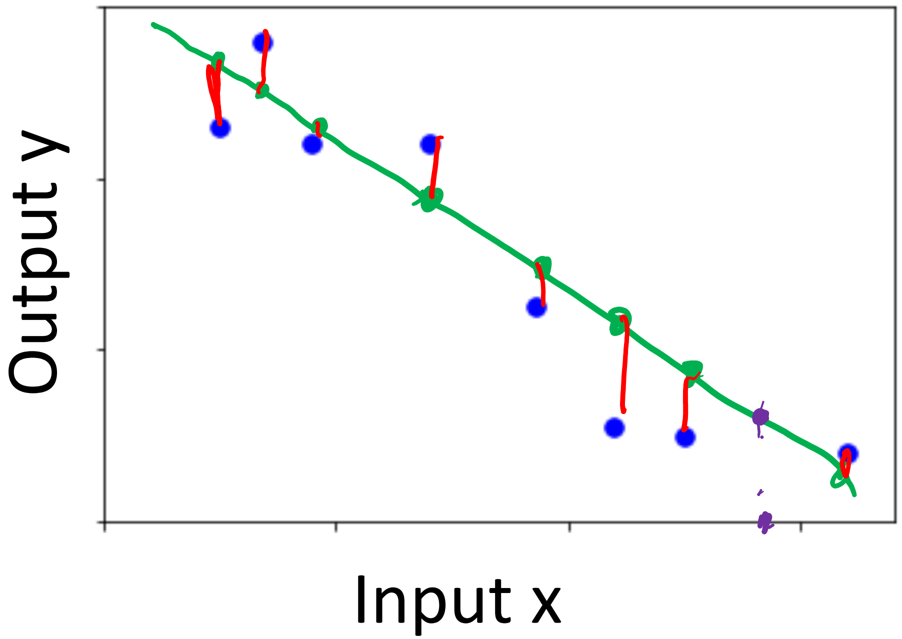
- Stock price prediction
- Forecasting epidemics
- Speech synthesis
- Generation of images (e.g. *Deep Fake*)



Lecture 2: Problem Formulation



Lecture 2: Problem Formulation



Regression

$x \in \mathbb{R}$

$y \in \mathbb{R}$

$\hat{y} = h(x) = mx + b$

$\hat{y}^{(i)} = h(x^{(i)})$

$3x + 2$

$4x - 5$

$-32x + 7$

Error $y - \hat{y}$

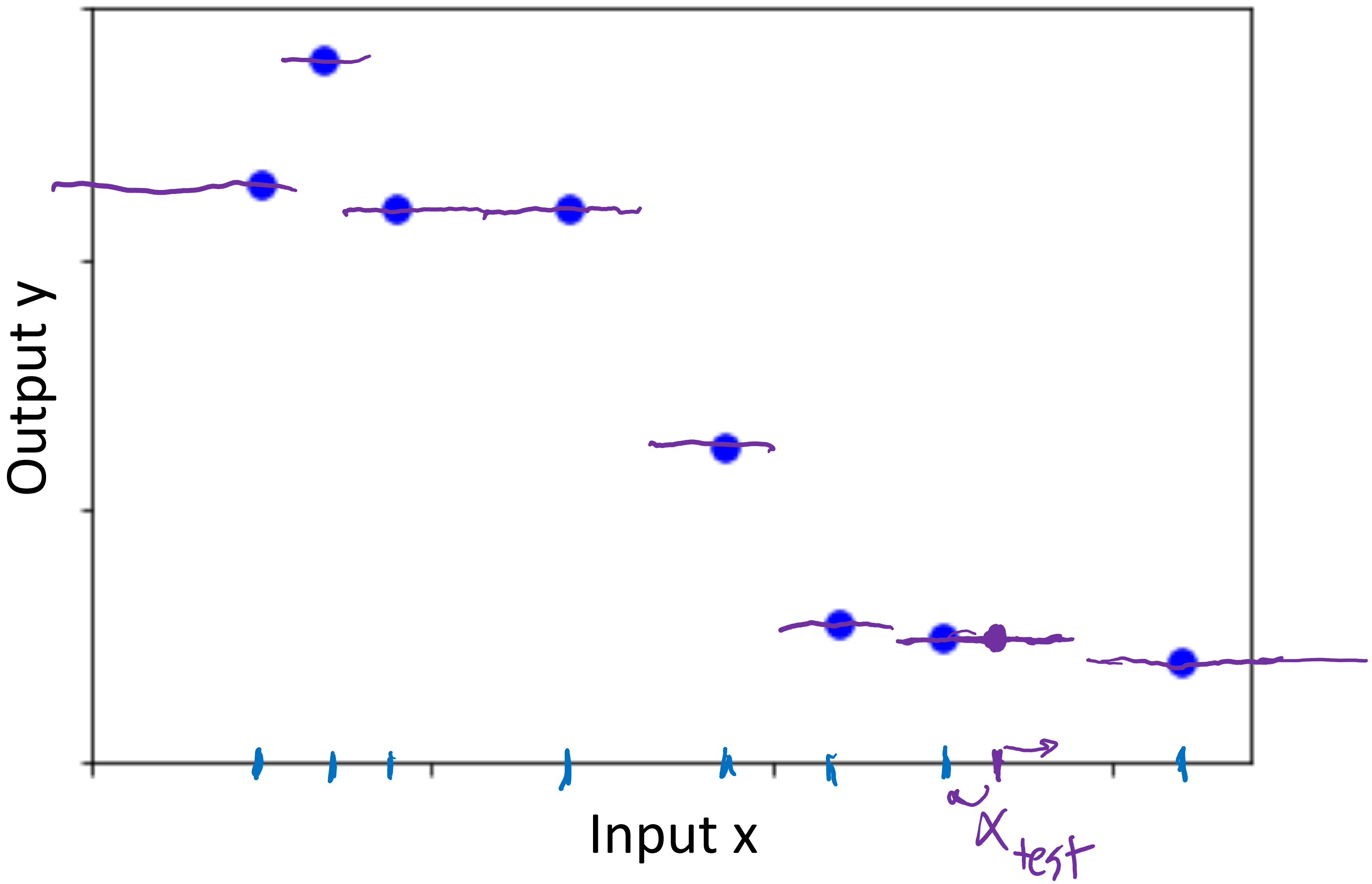
$\sum_{i=1}^{N} (y^{(i)} - \hat{y}^{(i)}) \rightarrow \sum |y^{(i)} - \hat{y}^{(i)}| \rightarrow \sum (y^{(i)} - \hat{y}^{(i)})^2$

sum sq error

$\frac{1}{N} \sum (y^{i} - \hat{y}^{i})^2$

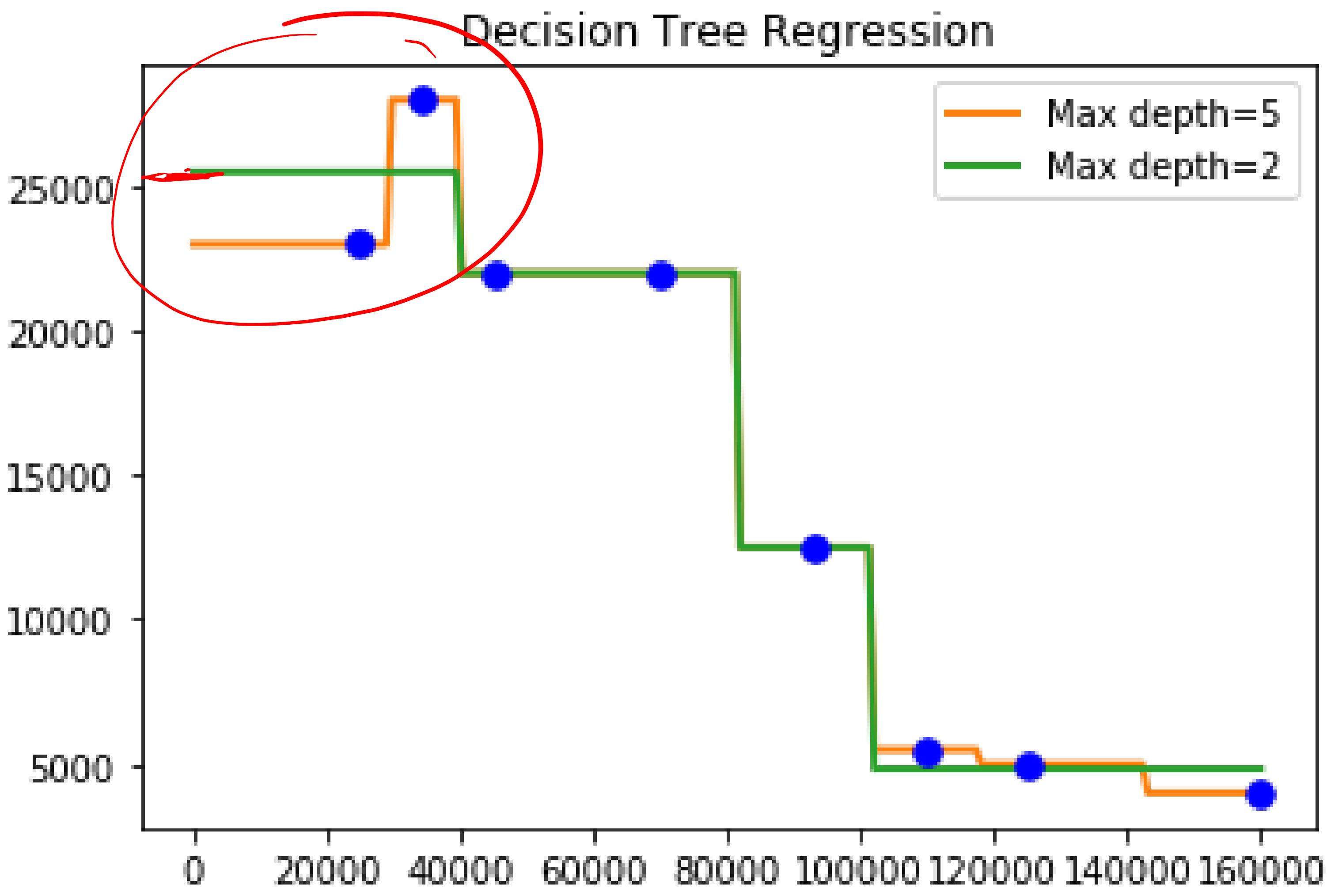
mean sq error

Regression: Nearest Neighbor



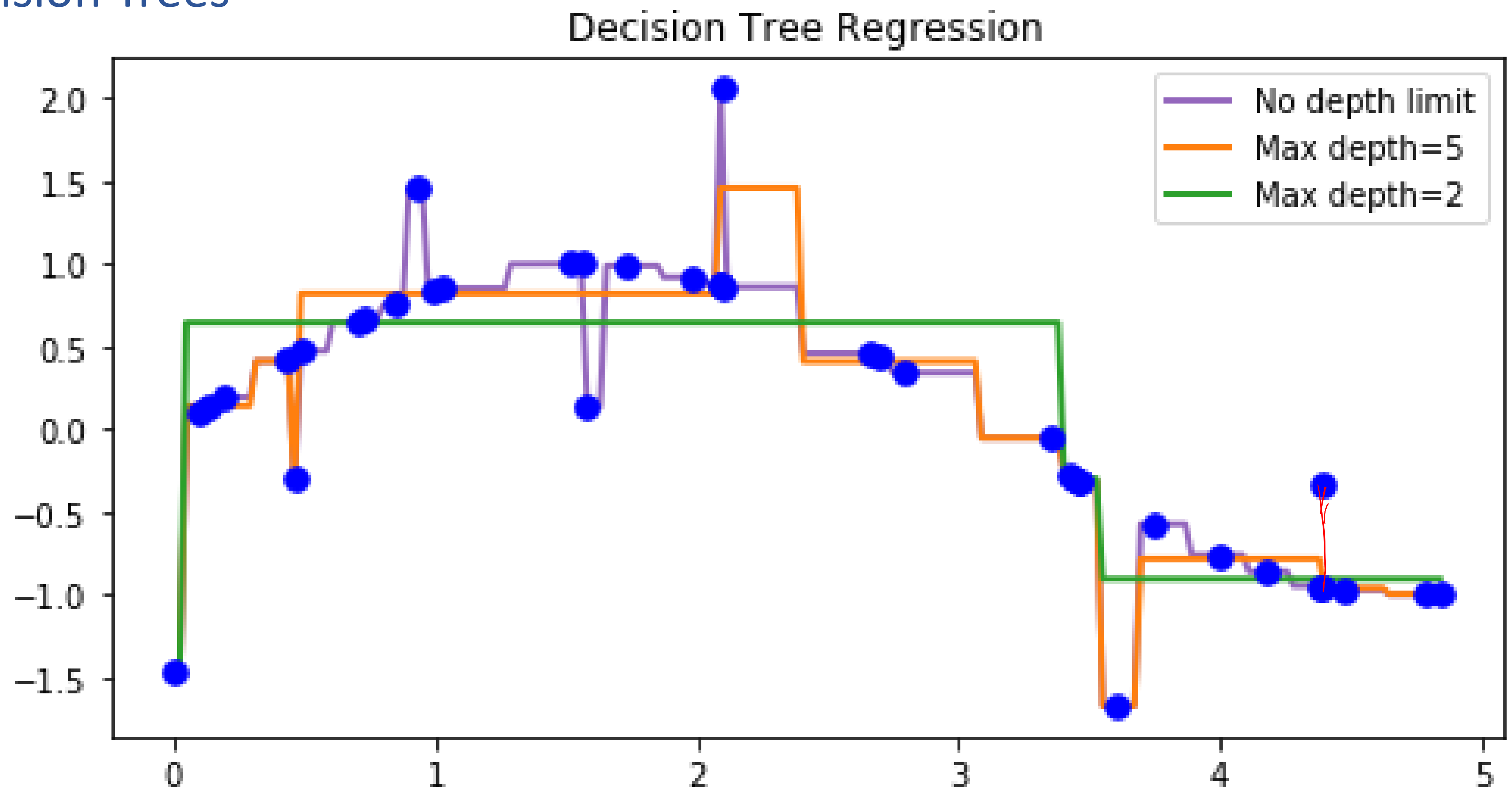
Regression

Decision Trees



Nonparametric Regression

Decision Trees



Linear Regression

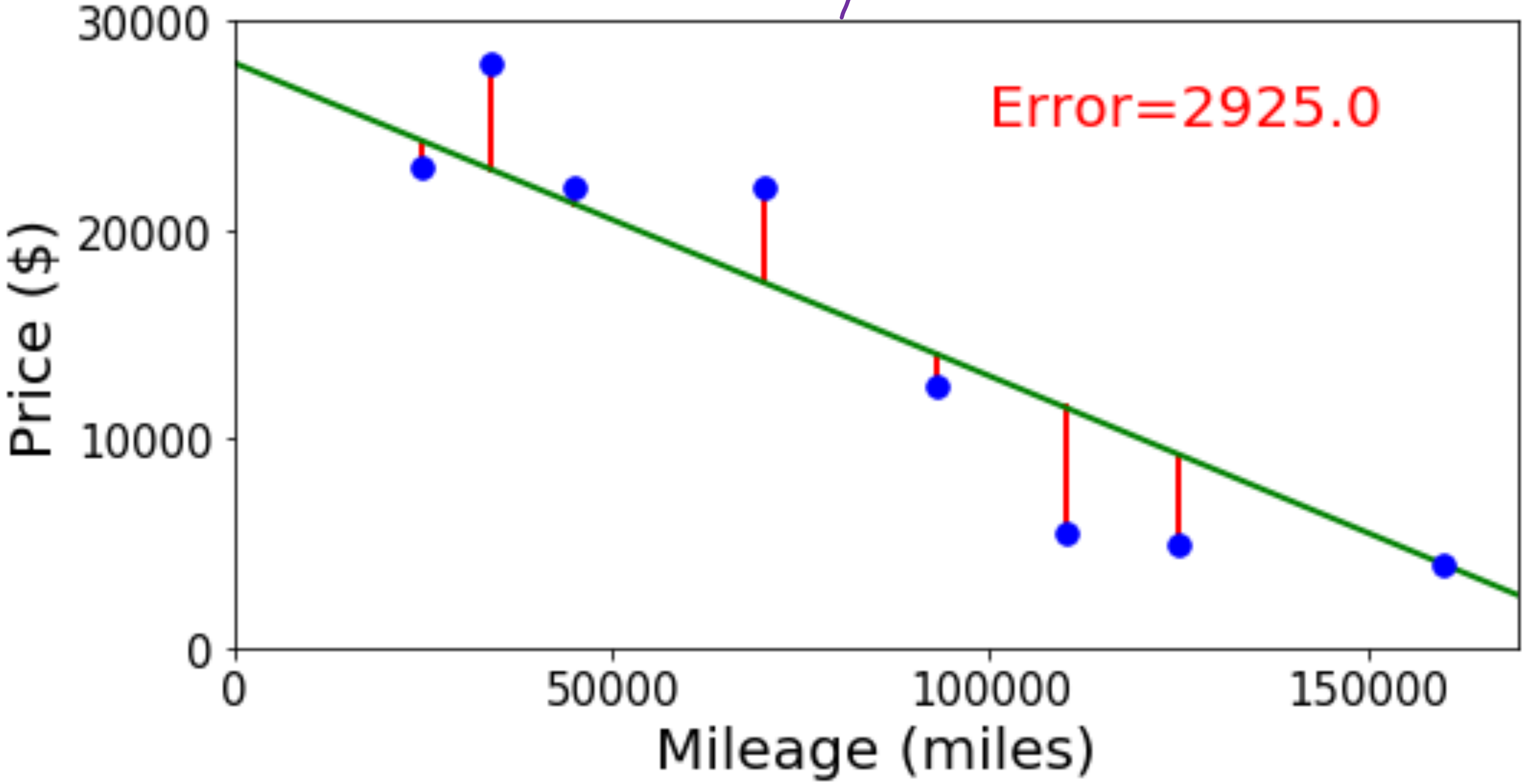
Selling my car

$y = mx + b$

$y = wx + b$

$y = w_1 x + w_0$

$y = \theta_1 x + \theta_0$



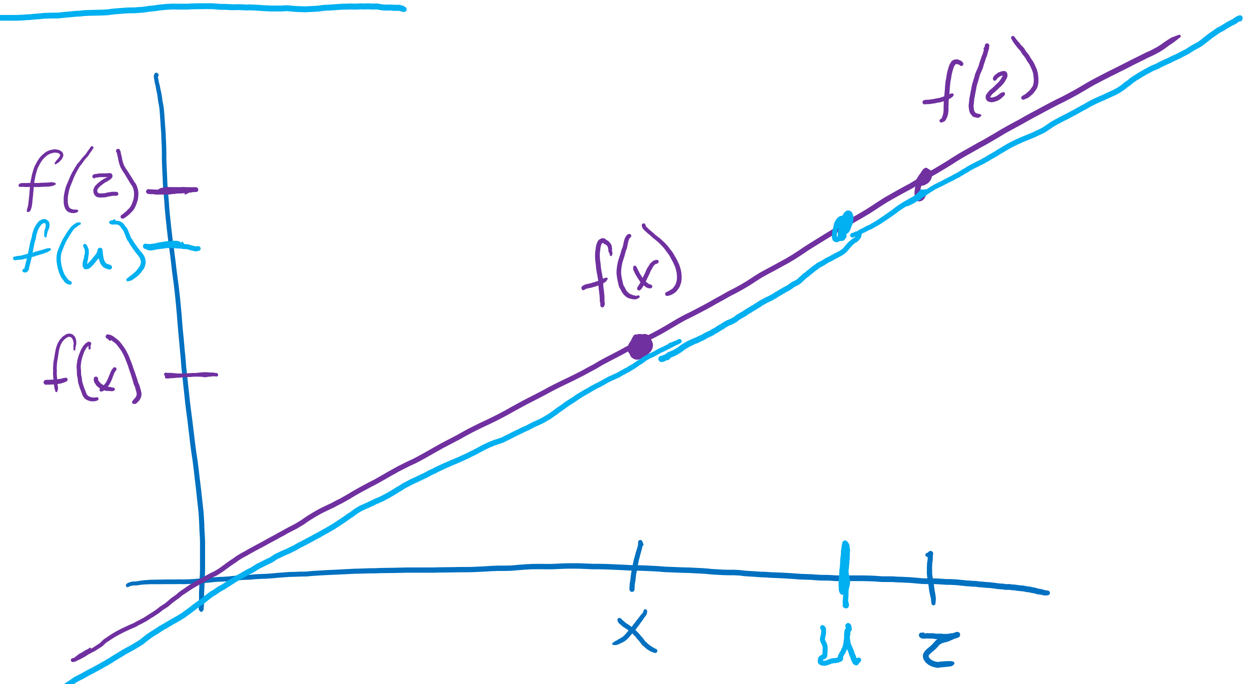
Linear Function

Linear function

If $f(\boldsymbol{x})$ is linear, then:

- $f(\boldsymbol{x}+\boldsymbol{z}) = f(\boldsymbol{x}) + f(\boldsymbol{z})$
- $f(\alpha\boldsymbol{x}) = \alpha f(\boldsymbol{x}) \quad \forall\alpha$
- $f(\alpha\boldsymbol{x}+(1-\alpha)\boldsymbol{z}) = \alpha f(\boldsymbol{x}) + (1-\alpha)f(\boldsymbol{z}) \quad \forall\alpha$

$\alpha = 0.25$



Linear in Higher Dimensions

1-D $y = wx + b$

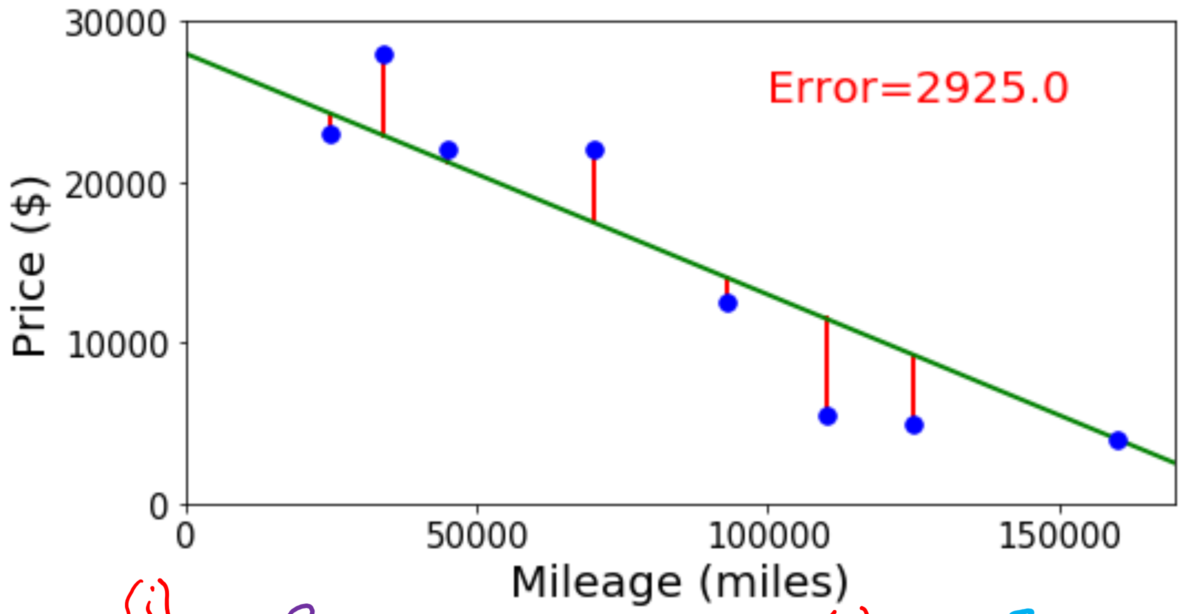
2-D $y = w_1 x_1 + w_2 x_2 + b$

What are these linear shapes called for 1-D, 2-D, 3-D, M-D input?

	$\boldsymbol{x} \in \mathbb{R}$	$\boldsymbol{x} \in \mathbb{R}^2$	$\boldsymbol{x} \in \mathbb{R}^3$	$\boldsymbol{x} \in \mathbb{R}^M$
→ $y = \boldsymbol{w}^T \boldsymbol{x} + b$	line	plane	hyperplane	hyperplane
$\boldsymbol{w}^T \boldsymbol{x} + b = 0$	point	line	plane	hyperplane
$\boldsymbol{w}^T \boldsymbol{x} + b \geq 0$	halfline	halfplane	halfspace	halfspace

Linear Regression

Linear algebra formulation



$$y = \vec{w}^T x_{orig} + b$$

$$y^{(i)} = \vec{w}^T x_{orig}^{(i)} + b$$

$$y^{(i)} = \vec{\theta}^T \vec{x} = \vec{x}^T \vec{\theta}$$

$$\begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(N)} \end{bmatrix} = \begin{bmatrix} — & x^{(1)T} & — \\ & \vdots & \\ — & x^{(N)T} & — \end{bmatrix} \begin{bmatrix} b \\ w_1 \\ \vdots \\ w_M \end{bmatrix}$$

$$\vec{y} = X \vec{\theta}$$

← Design Matrix X

$$\vec{x}_{orig}^{(i)} \in \mathbb{R}^2 \longrightarrow \vec{x}^{(i)} \in \mathbb{R}^3$$

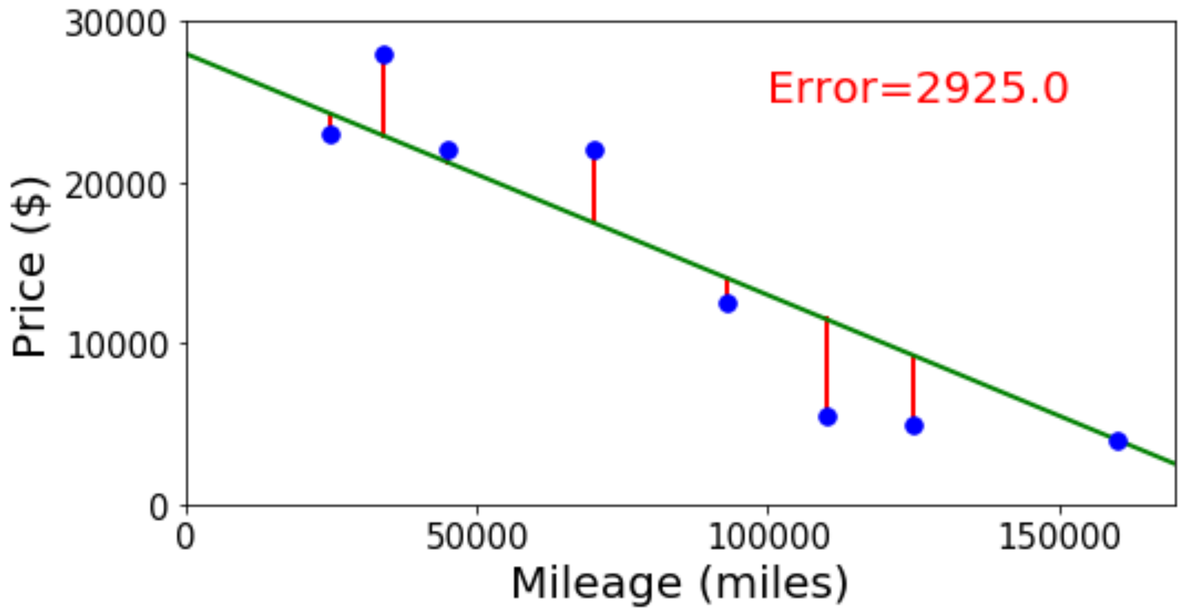
$$\vec{x}_{orig}^{(i)} = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \end{bmatrix} \longrightarrow \vec{x}^{(i)} = \begin{bmatrix} 1 \\ x_1^{(i)} \\ x_2^{(i)} \end{bmatrix}$$

$$\vec{w} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \longrightarrow \vec{\theta} = \begin{bmatrix} b \\ w_1 \\ w_2 \end{bmatrix}$$

b

Linear Regression

Error and objectives



$$J(w, b) = \frac{1}{N} \sum_{i=1}^{N} \left(y^{(i)} - \hat{y}^{(i)}\right)^2$$

$$\hat{y}^{(i)} = w x^{(i)} + b$$

$$J(w_1, w_2, b) = \frac{1}{N} \sum \left(y^{(i)} - \hat{y}^{(i)}\right)^2$$

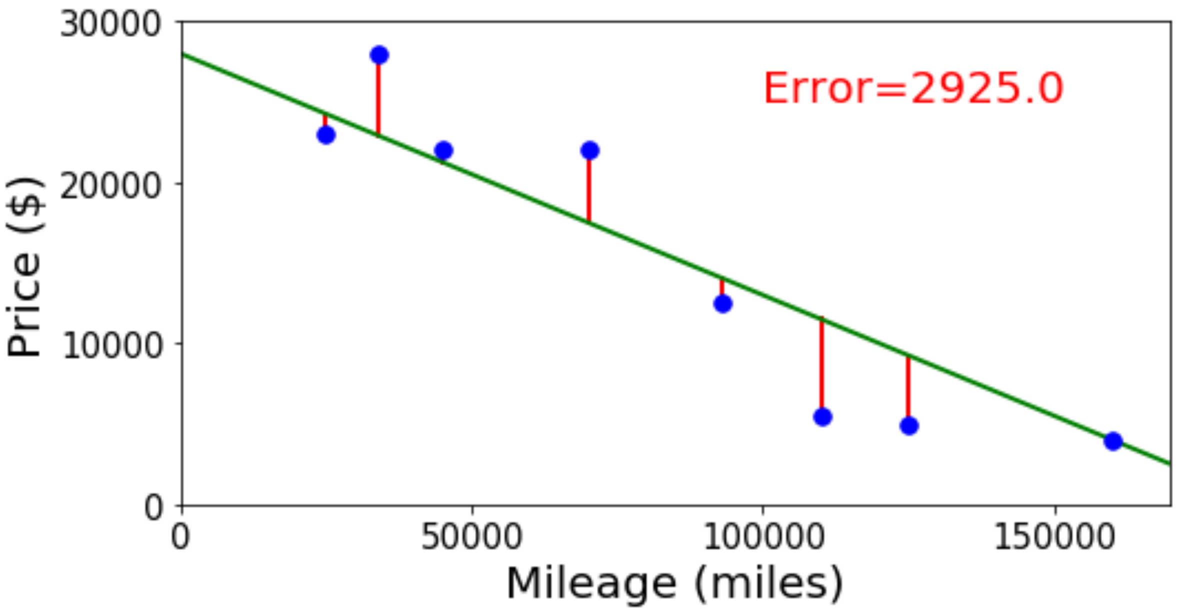
$$\hat{y}^{(i)} = w_1 x_1^{(i)} + w_2 x_2^{(i)} + b$$

$$J(w_1, \ldots, w_m, b) =$$

$$\hat{y}^{(i)} = \sum_{j=1}^{m} w_j x_j^{(i)} + b$$

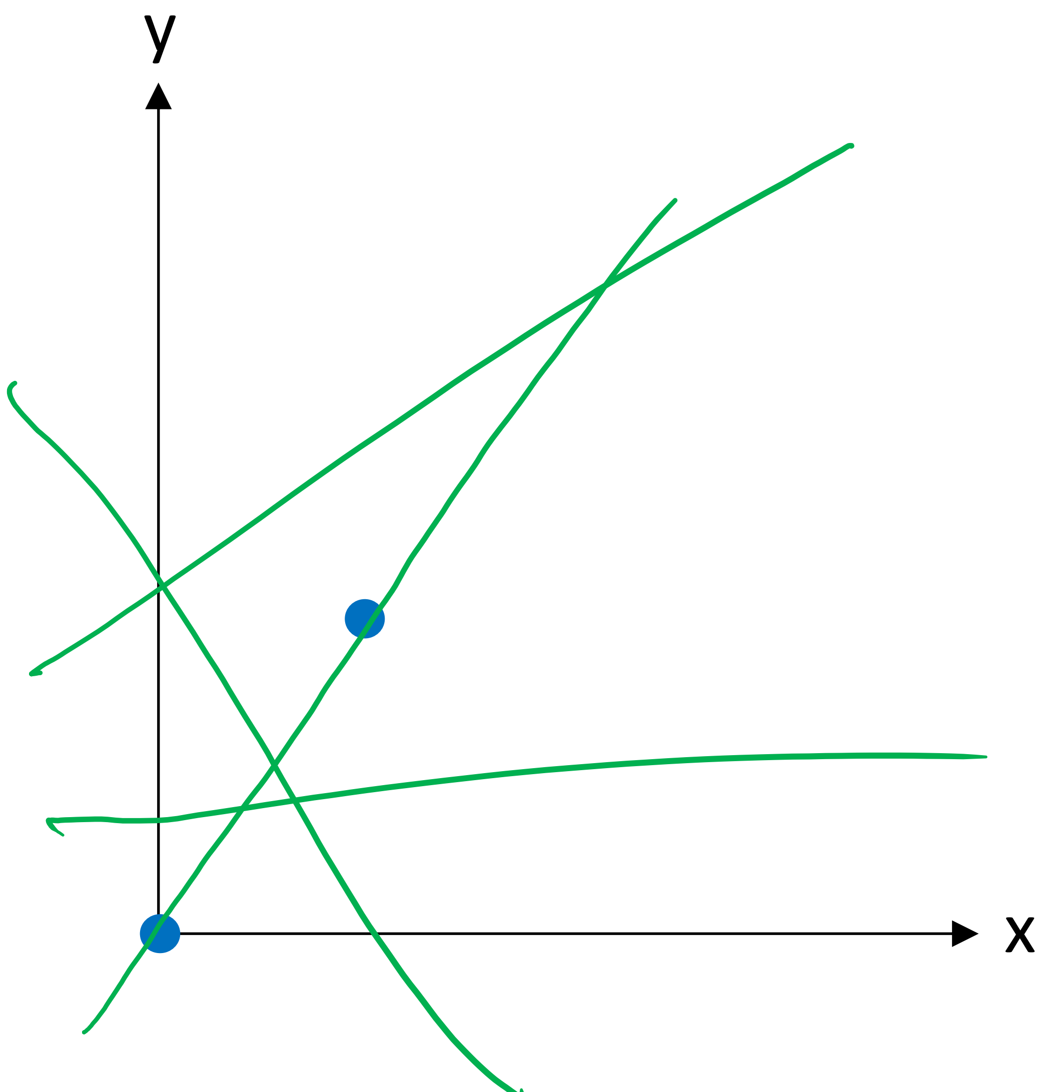
Linear Regression

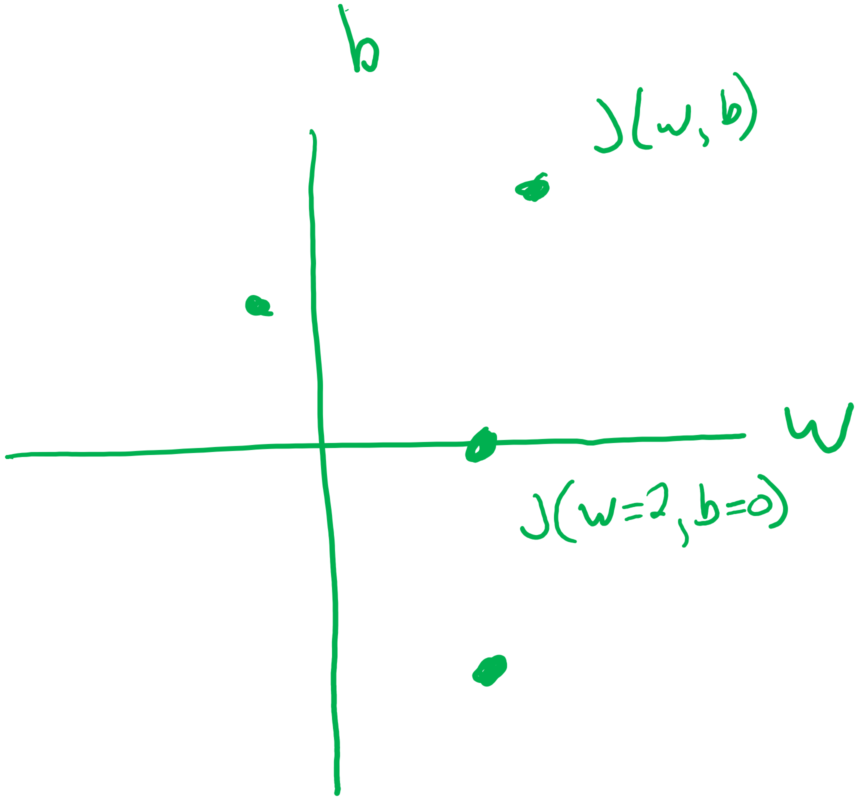
Linear algebra formulation



Linear Regression

Optimizing the objective





Piazza Poll 2

For fixed data and fixed slope, w, what shape do we get by plotting MSE objective vs intercept, b?

A. Line 30% → 15%

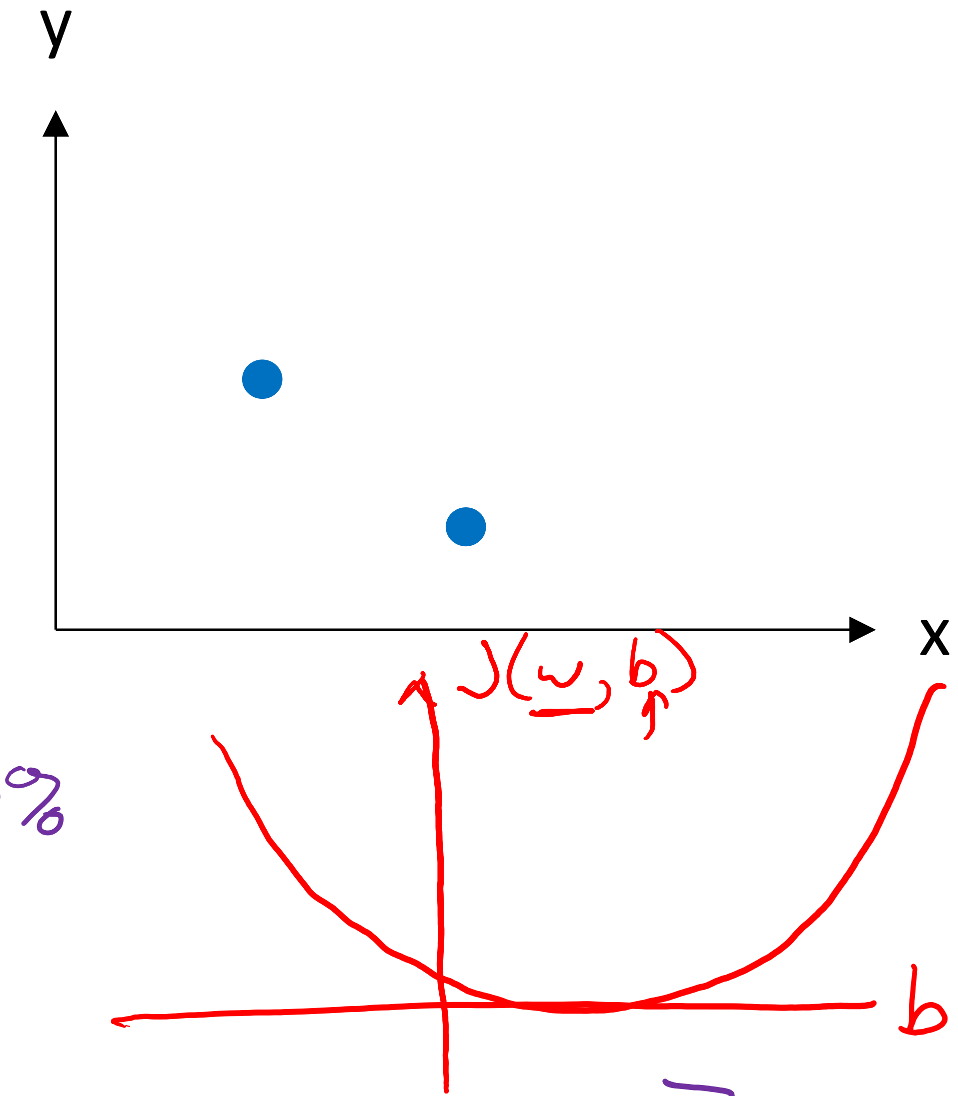
B. Plane

C. Half-plane

D. Convex Parabola (U-shape) 30% → 60%

E. Concave parabola (up-side-down U)

F. None of the above



$$J(w,b) = \frac{1}{2}\left[\left(y^{(1)} - (wx^{(1)} + b)\right)^2 + \left(y^{(2)} - (wx^{(2)} + b)\right)^2\right]$$