EEP 596: Adv Intro ML || Lecture 2

Dr. Karthik Mohan

Univ. of Washington, Seattle

Jan 6, 2022

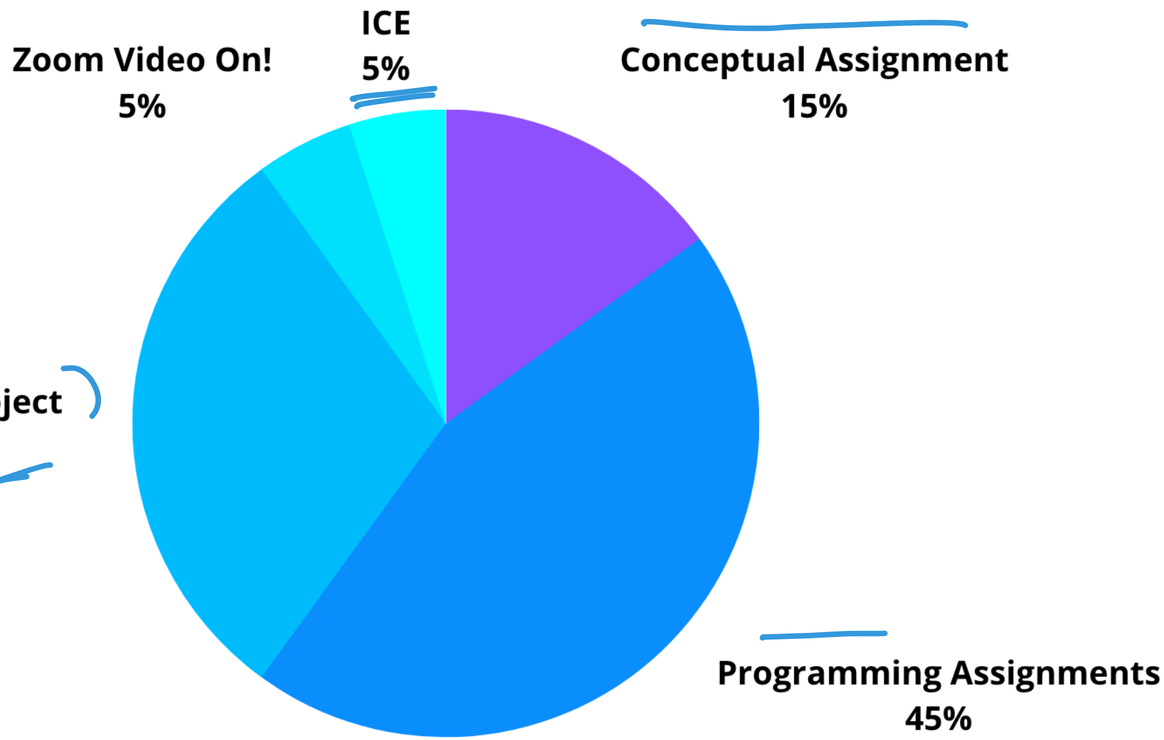
Logistics

- **Lecture** Tuesday Lecture: Expectation that you join in person. Thursday Lecture: Zoom (zoom attendance will be taken).
- **Assignment** Programming Assignment 1 to be assigned - Due next Thursday, January 12th, midnight
- **Office Hours** Karthik: 6 - 6:30 pm on Thursday, Ayush - TBD
- **Calendly slots** Feel free to pick calendly slots for 1:1 15 min syncs as needed (recommended)
- **Course Webpage** https://bytesizeml.github.io/ml2023/

Weekly Schedule

	Day	Timings	Class type
Lecture 1 (In-person)	T	4 pm - 6 pm	In-person
Lecture 2	Th	4 pm - 6 pm	Zoom
Office Hours Karthik	Th	6 - 6:30 pm	Zoom
Office Hours Ayush	TBD	TBD	Zoom
Quiz Section Ayush	TBD	TBD	Zoom
Grading hours	TBD	TBD	Zoom

Assessments Breakdown



Lecture Structure

Format for each lecture (ICE)

- Sprinkle in a few In-class exercises MCQ for conceptual understanding
- Where required - Will set extra context on applications/background - This may be slow for some but super useful for rest of class - Let's adjust and adapt!
- Break at 1 hour mark
- Break-outs in between/end of class for peer discussion + networking
- Anything else ?

Class goes at the average pace!

Quick pointers

- We will cater the lecture to discuss fundamentals and go at a pace comfortable with the average of the class
- If the class/topic is going too fast for you - There maybe brushing up of background (e.g. linear algebra/calculus/programming) that you may have to do in your own time!
- If a topic is going slow - Opportunity to dive deeper into the topic through additional reading of papers or programming
- Be sure to brush up/catch up on your python and linear algebra to gear up for upcoming lectures and assignments

Lectures and Programming Assignments (Tentatively)

Week	Lecture Material	Assignment
1	Linear Regression	Housing Price Prediction
2	Classification	Spam classification (Kaggle)
3	Classification	Flower/Leaf classification
4	Clustering	MNIST digits clustering
5	Anomaly Detection	Crypto Prediction (Kaggle + P)
6	Data Visualization	Crypto Prediction (Kaggle + P)
7	Deep Learning	Visualizing 1000 images
8	Deep Learning (DL)	ECG Arrythmia Detection
9	DL in NLP	TwitterSentiment Analysis (Kaggle + P)
10	DLs in Vision	TwitterSentiment Analysis (Kaggle + P)

Coding pointers

- Assignments assume python as the main language (e.g. for hints and modules, etc)
- Coding environment set-up will be one of the problems on HW 1
- Prototyping can be done on notebooks and submitted as such for smaller assignments.
- For mini-projects and kaggle assignments - Please keep your code modular and organized.

Coding Environment

- Pointers below if you want to get set up on Google Colab for both prototyping, running machine-intensive ML experiments and working with code through IDEs
- Prototype Coding work in Notebooks recommended on Google Colab
- For terminal access on Google Colab, sign up for pro
- `pip3 install colabcode` on termainal
- ColabCode enables you to have a VSCode IDE port into Google Colab - So you can work on the IDE from your laptop but run experiments on Google Colab!

Maximizing Your Learning of Machine Learning!

- Ask questions during lectures - Clarify things as they happen!

Maximizing Your Learning of Machine Learning!

- Ask questions during lectures - Clarify things as they happen!
- Make use of office hours and quiz section!

Maximizing Your Learning of Machine Learning!

- Ask questions during lectures - Clarify things as they happen!
- Make use of office hours and quiz section!
- Collaborative learning - Discord is a great place to brainstorm concepts outside class and unblock yourself.

Maximizing Your Learning of Machine Learning!

- Ask questions during lectures - Clarify things as they happen!
- Make use of office hours and quiz section!
- Collaborative learning - Discord is a great place to brainstorm concepts outside class and unblock yourself.
- 30% of your learning happens in class and office hours - The remaining 70% happen when you work on the assignments. (You ofcourse need the 30 to get to the 70 :D)

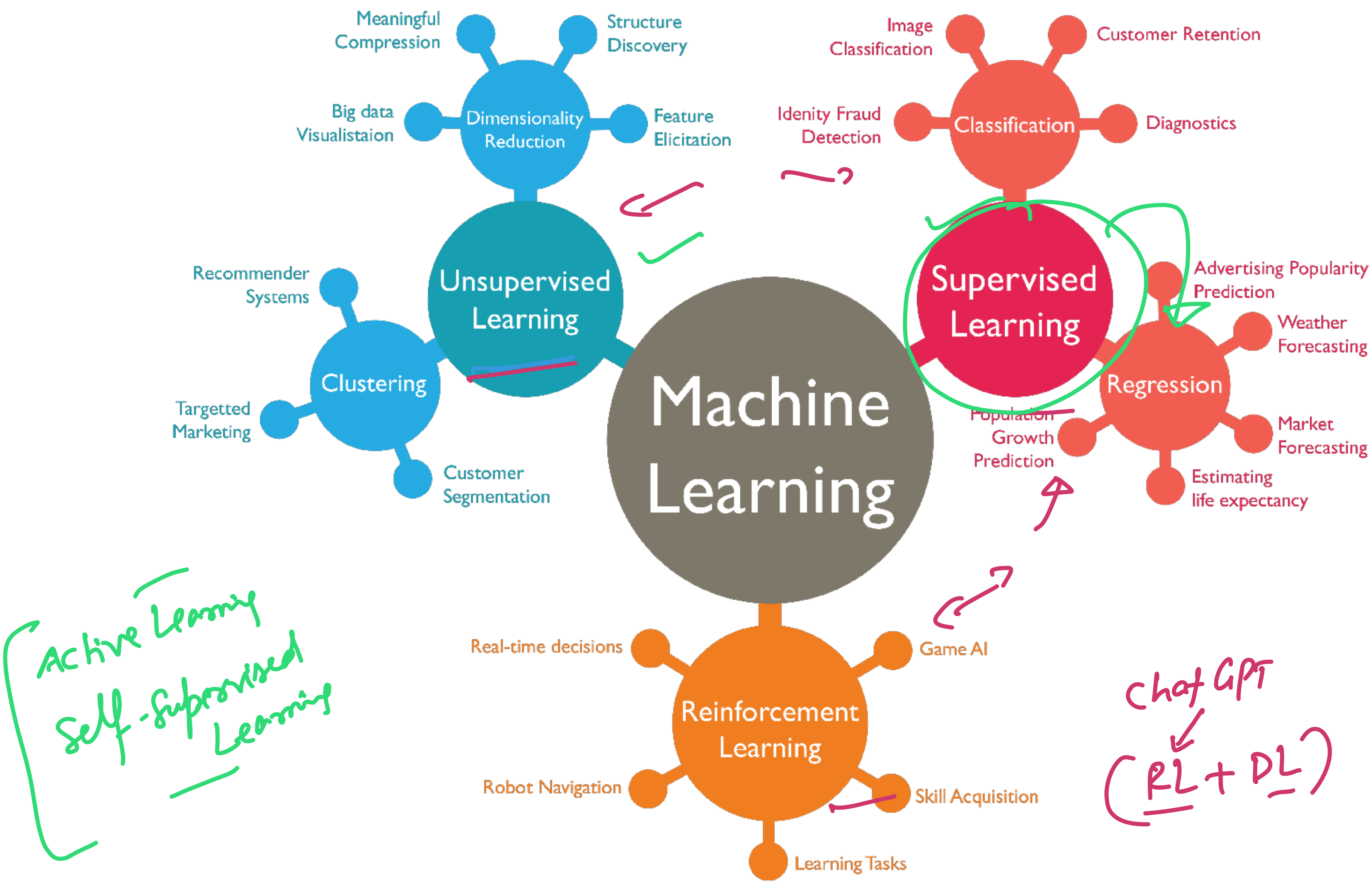
Maximizing Your Learning of Machine Learning!

- Ask questions during lectures - Clarify things as they happen!
- Make use of office hours and quiz section!
- Collaborative learning - Discord is a great place to brainstorm concepts outside class and unblock yourself.
- 30% of your learning happens in class and office hours - The remaining 70% happen when you work on the assignments. (You ofcourse need the 30 to get to the 70 :D)
- What you put in is what you get out!

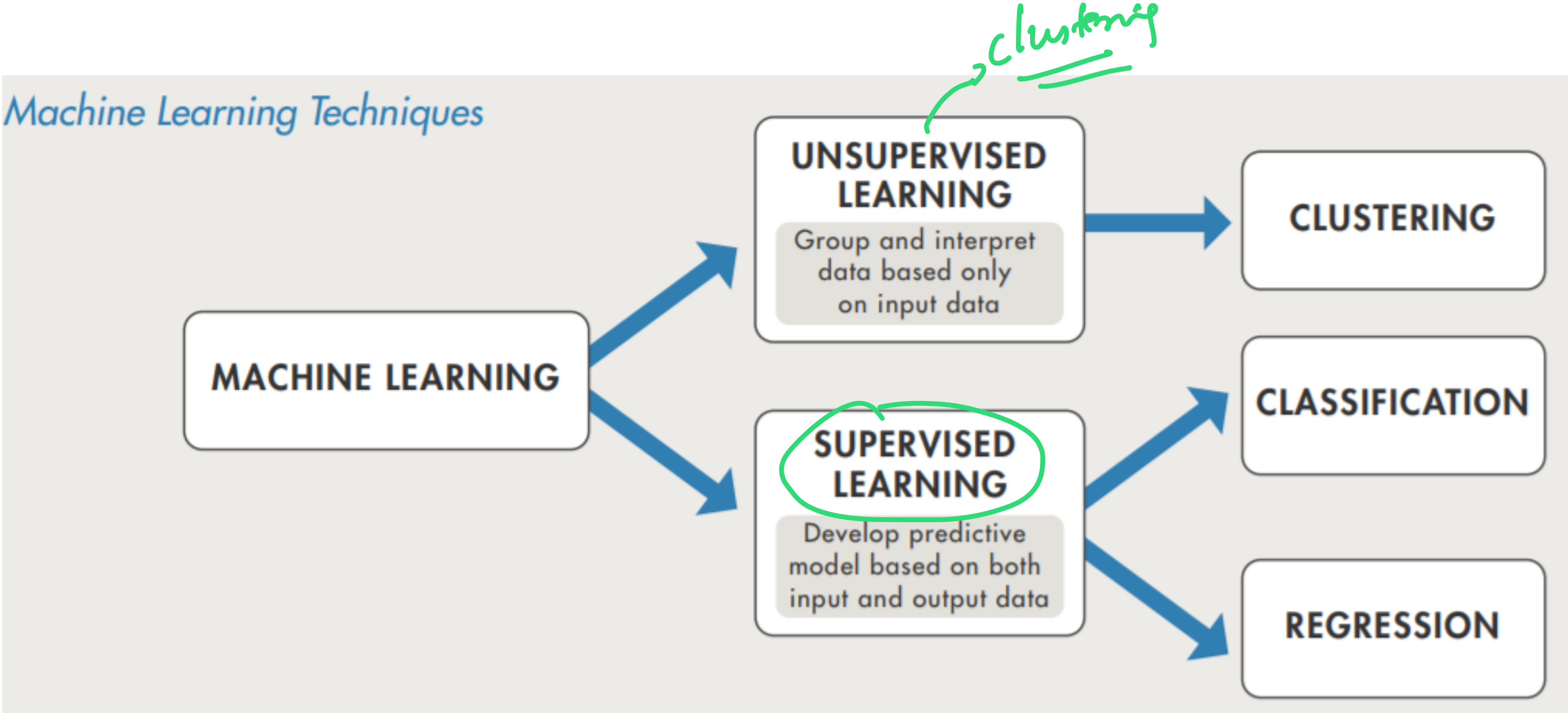
Maximizing Your Learning of Machine Learning!

- Ask questions during lectures - Clarify things as they happen!
- Make use of office hours and quiz section!
- Collaborative learning - Discord is a great place to brainstorm concepts outside class and unblock yourself.
- 30% of your learning happens in class and office hours - The remaining 70% happen when you work on the assignments. (You ofcourse need the 30 to get to the 70 :D)
- What you put in is what you get out!
- Excitement + Smart work + Inquisitiveness = Maximized learning!

What is Machine Learning?



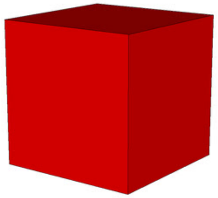
Supervised vs Unsupervised Learning



Supervised Learning

Objects — Label

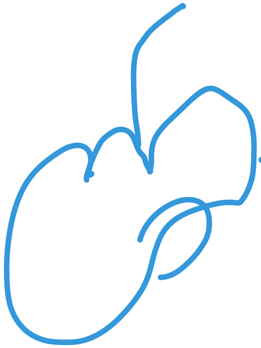
1  "Apple"

2  "Cube"

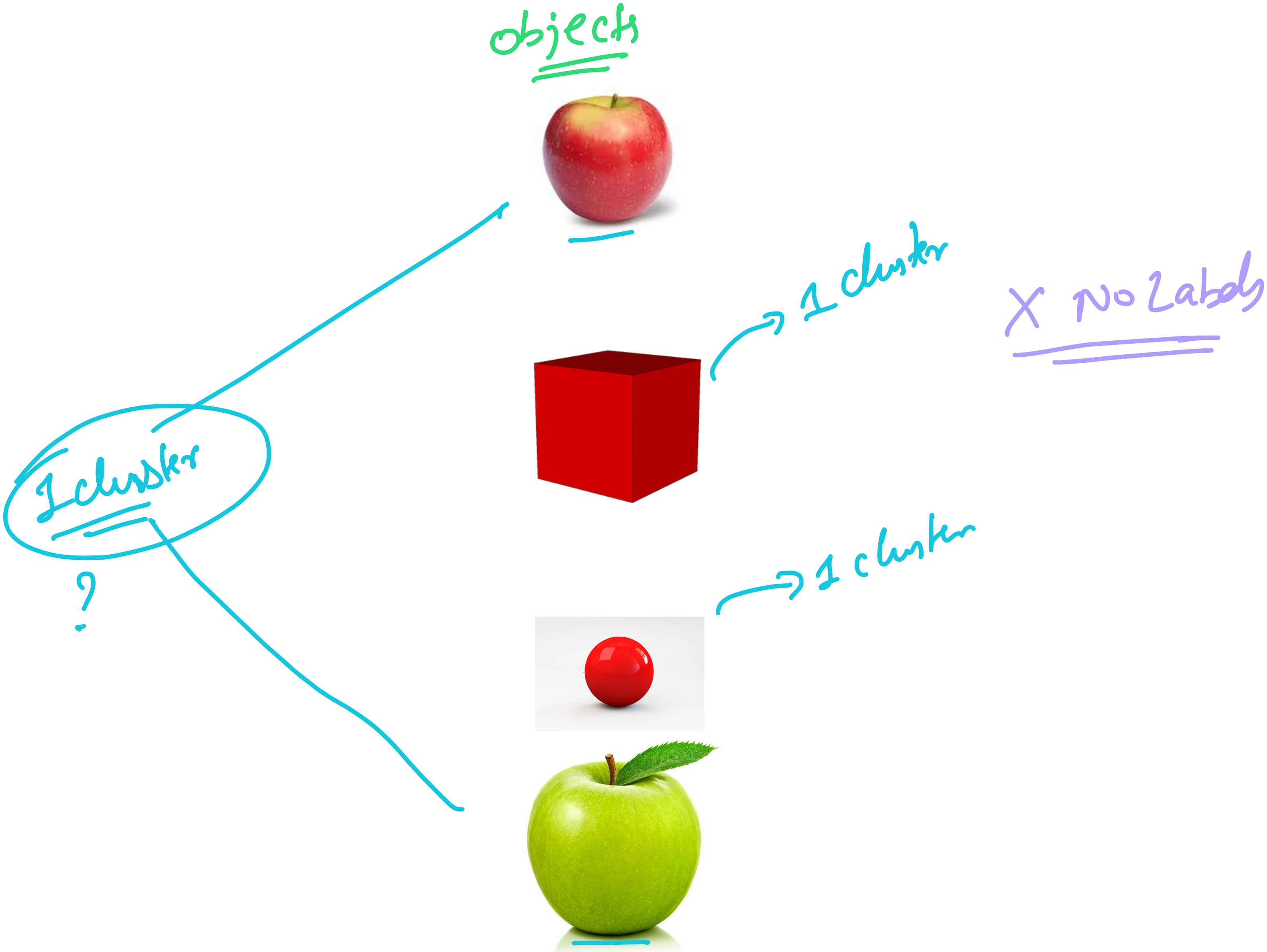
3  "Ball"

4  "Apple"

→ Training + Predictions

 → Apple

Un-Supervised Learning

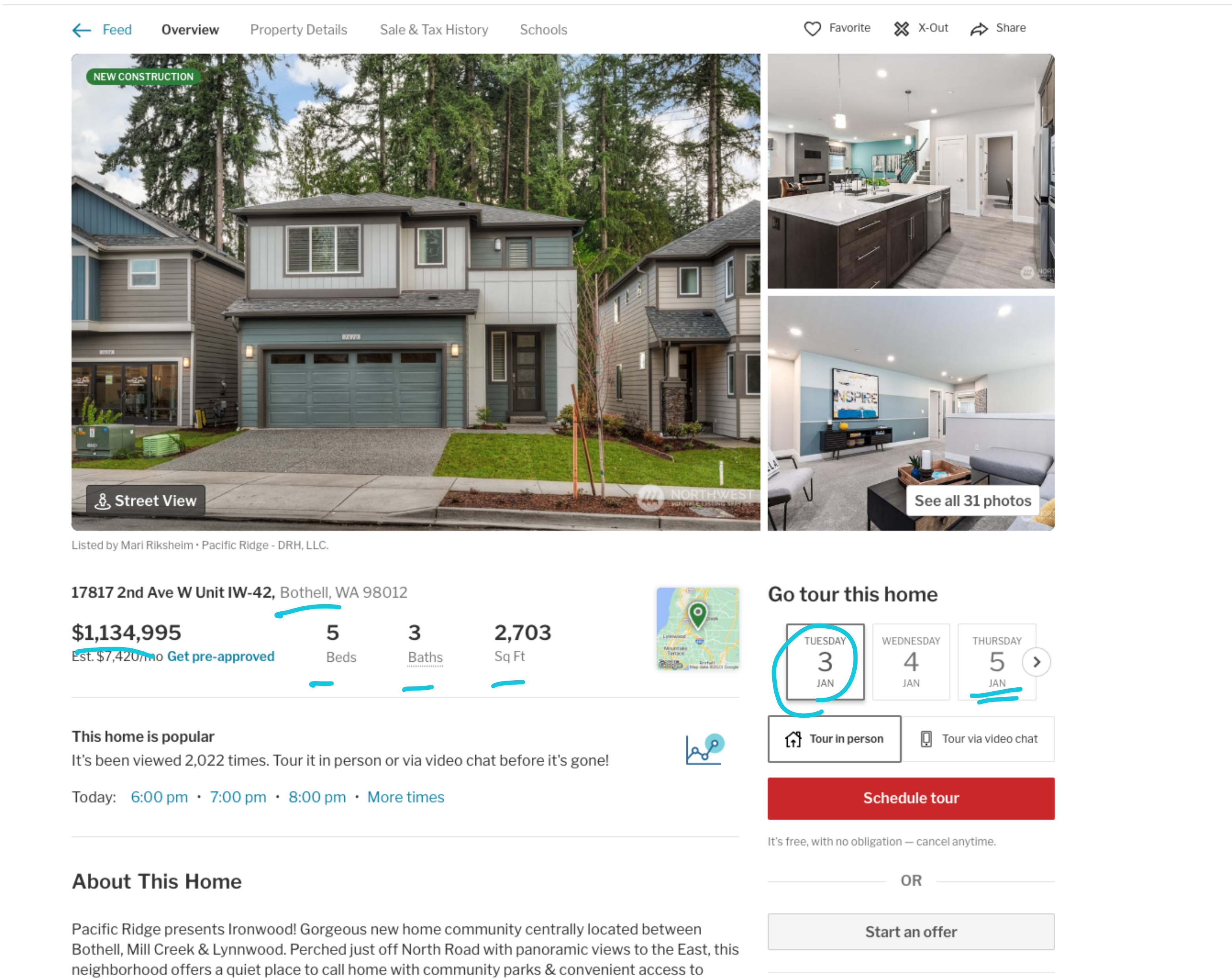


Our first ML method: Linear Regression

Supervised Learning Problem

Application: Housing Prices

Redfin


Feed
Overview
Property Details
Sale & Tax History
Schools
Favorite
X-Out
Share
NEW CONSTRUCTION
Street View
See all 31 photos
Listed by Mari Riksheim • Pacific Ridge - DRH, LLC.
17817 2nd Ave W Unit IW-42, Bothell, WA 98012
$1,134,995
Est. $7,420/mo Get pre-approved
5
Beds
3
Baths
2,703
Sq Ft
Go tour this home
TUESDAY
3
JAN
WEDNESDAY
4
JAN
THURSDAY
5
JAN
Tour in person
Tour via video chat
This home is popular
It's been viewed 2,022 times. Tour it in person or via video chat before it's gone!
Today: 6:00 pm • 7:00 pm • 8:00 pm • More times
Schedule tour
It's free, with no obligation — cancel anytime.
OR
Start an offer
About This Home
Pacific Ridge presents Ironwood! Gorgeous new home community centrally located between Bothell, Mill Creek & Lynnwood. Perched just off North Road with panoramic views to the East, this neighborhood offers a quiet place to call home with community parks & convenient access to

Redfin Estimate

This home is popular
It's been viewed 2,022 times. Tour it in person or via video chat before it's gone!

Today: 6:00 pm • 7:00 pm • 8:00 pm • More times

About This Home

Pacific Ridge presents Ironwood! Gorgeous new home community centrally located between Bothell, Mill Creek & Lynnwood. Perched just off North Road with panoramic views to the East, this neighborhood offers a quiet place to call home with community parks & convenient access to

Continue reading

Listed by Mari Riksheim • Pacific Ridge - DRH, LLC
Listed by Melissa Cogswell • Pacific Ridge - DRH, LLC
Redfin checked: 3 minutes ago (Jan 3, 2023 at 2:57pm) • Source: NWMLS #2024145

Home Facts

Status	Active	Time on Redfin	5 days
Property Type	Residential, Residential	HOA Dues	$88/month
Year Built	2023	Style	Contemporary
Community	Lynnwood	Lot Size	6,252 Sq. Ft.
MLS#	2024145		

Price Insights

List Price	$1,134,995	Est. Mo. Payment	$7,420
Redfin Estimate	$1,136,063	Price/Sq.Ft.	$420

Go tour this home

TUESDAY 3 JAN | WEDNESDAY 4 JAN | THURSDAY 5 JAN

Tour in person | Tour via video chat

Schedule tour

It's free, with no obligation — cancel anytime.

OR

Start an offer

Ask a question | (425) 584-3263

Zillow:
("Zestimate")

Zoom Breakout #1

Zillow Estimate/RedFin Estimate

If you are on the market to buy a house, you would perhaps be looking at "Zestimates" or "RedFin Estimates" to filter out houses in your budget range. Discuss in your group, what are the factors that influence the price of a home and what are the factors (also called features in ML) that may have been used to construct these estimates. Once you have a set of factors identified, how do you combine them to produce the final house price estimate?

Typical Housing Data, Seattle

Index	SqFt	#Rooms	# Bathrooms	Location	Selling Price
1	2500	4	3	Bothell	1 MM
2	2000	3	2	Bellevue	950k
3	3000	4	3	Sammamish	1.3 MM
4	3000	4	3	Issaquah High	1.6 MM
5	...	...	...	...	...

Typical Housing Data, Seattle

Index	SqFt	#Rooms	# Bathrooms	Location	**Selling Price**
1	2500	4	3	Bothell	1 MM
2	2000	3	2	Bellevue	950k
3	3000	4	3	Sammamish	1.3 MM
4	3000	4	3	Issaquah High	1.6 MM
5	...	...	...	...	...

Other attributes for housing price prediction

Other attributes that matter?

- Crime / School Districts
- Avg. SP in the neighborhood
- View (Issaquah Highlands)

Categorical vs Numerical Attributes

Categorical

Attributes that fall into a clear set of categories. Example: zipcode of a place

Categorical vs Numerical Attributes

Categorical

Attributes that fall into a clear set of categories. Example: zipcode of a place

Numerical

Attributes that fall in a numeric range. Example: weight or height of a person

Categorical vs Numerical Attributes

Categorical

Attributes that fall into a clear set of categories. Example: zipcode of a place

Numerical

Attributes that fall in a numeric range. Example: weight or height of a person

Modeling Choice

Sometimes, whether an attribute is categorical or numerical is a modeling choice!

Categorical vs Numerical Attributes

Categorical or Numerical??

Index	SqFt	#Rooms	# Bathrooms	Location	Selling Price
1	2500	4	3	Bothell	1 MM
2	2000	3	2	Bellevue	950k
3	3000	4	3	Sammamish	1.3 MM
4	3000	4	3	Issaquah High	1.6 MM
5	...	...	...	...	...

Numerical

"Treat" as categorical attribute

Categorical

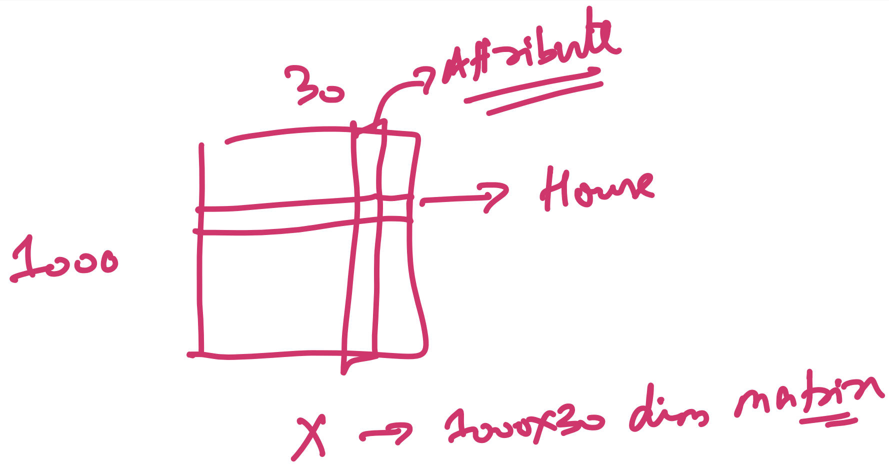
numeric output

Boolean Attribute: 0 / 1 } Still categorical!

Matrices and Vectors

Data matrix X

Let's say in our Housing database, we have 1000 houses and 30 attributes. If we wanted to represent this as a data matrix, X, what would be the dimensions of such a matrix ?



Matrices and Vectors

Data matrix X

Let's say in our Housing database, we have 1000 houses and 30 attributes. If we wanted to represent this as a data matrix, X, what would be the dimensions of such a matrix ?

Price vector y

For the same example as before, we take the housing prices of all the homes and put them into a price vector y. What would be the dimension of this vector y ?

y : 1000×1

$X \in \mathbb{R}^{1000 \times 30}$

$y \in \mathbb{R}^{1000 \times 1}$

X and y in housing data

Index	SqFt	#Rooms	# Bathrooms	Location	**Selling Price**
1	2500	4	3	Bothell	1 MM
2	2000	3	2	Bellevue	950k
3	3000	4	3	Sammamish	1.3 MM
4	3000	4	3	Issaquah High	1.6 MM
5	...	...	...	...	...

Linear Model

Linear Model

In Linear models, we assume that the target y is a linear combination of the attributes or features x. This is a 'modeling assumption'. The combination is represented by a weight vector w.

x – features
y – target
w – weight vector

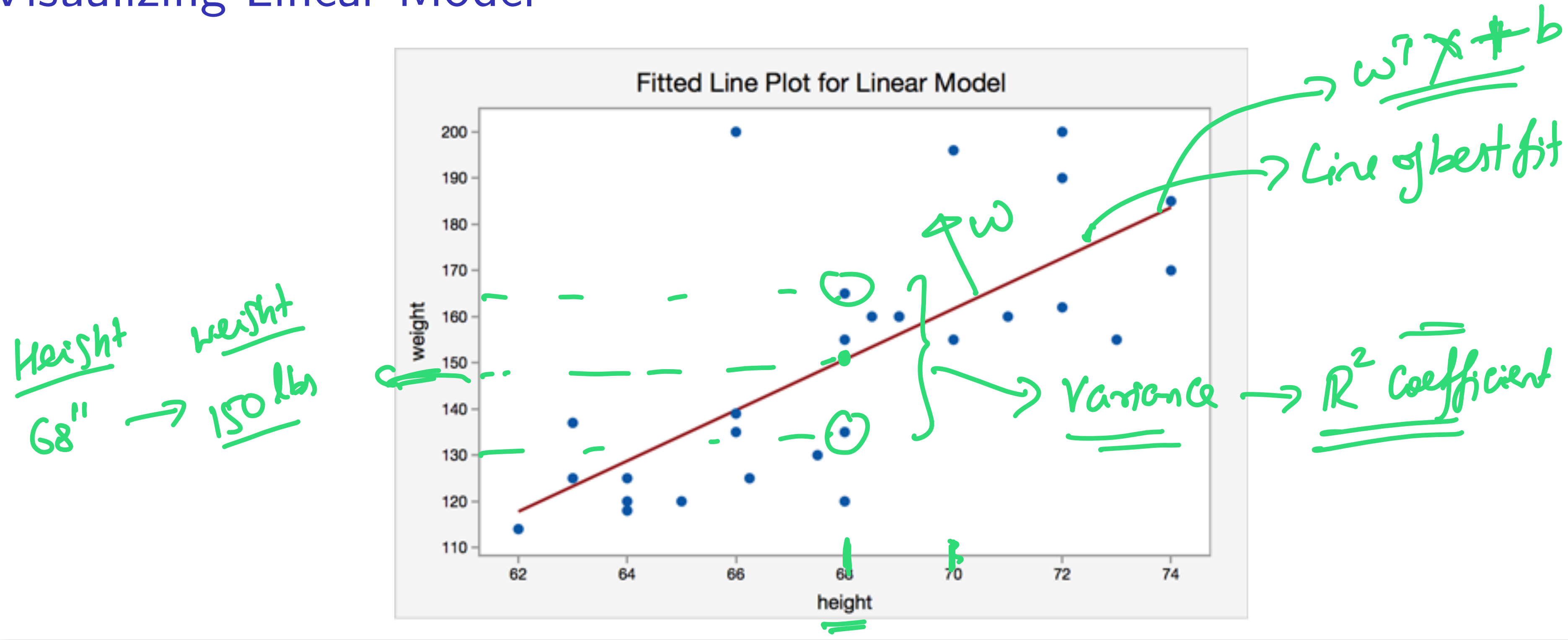
Decision Trees
– Non-linear model

Linear Model

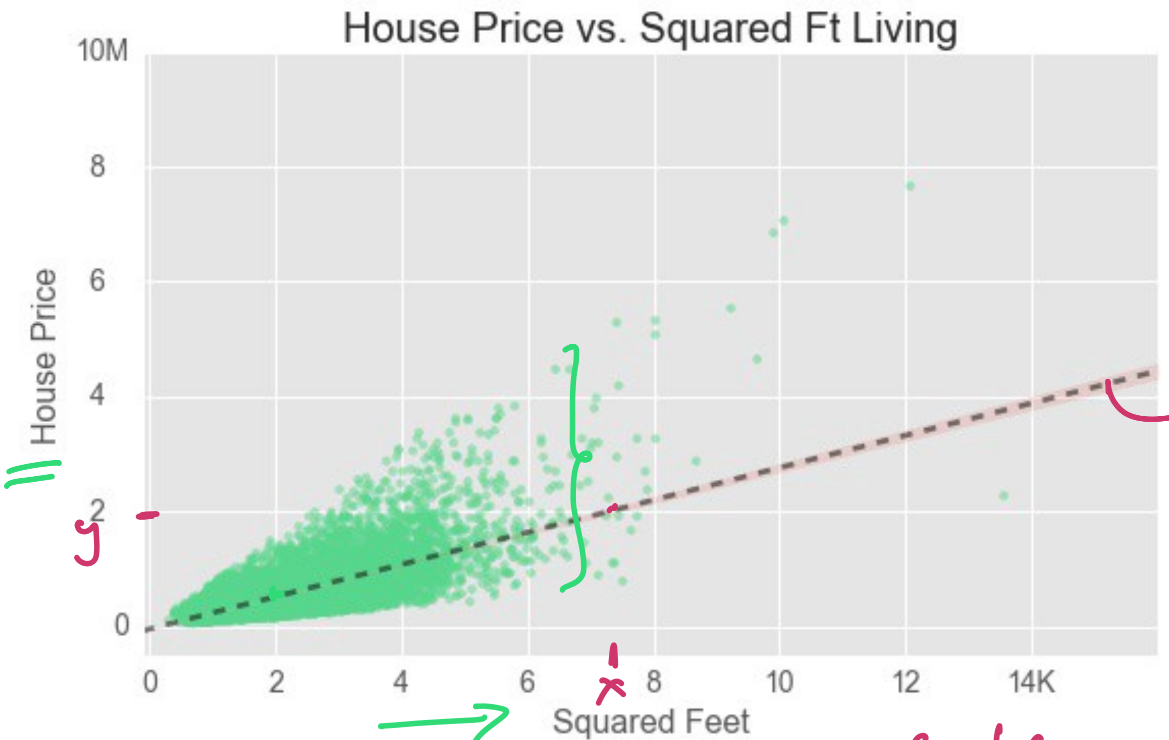
Linear Model

In Linear models, we assume that the target, y is a linear combination of the attributes or features x. This is a 'modeling assumption'. The combination is represented by a weight vector w.

Visualizing Linear Model



Linear Model for Housing Prices Application



Linear Model
- Good place to start
- Good Baseline

y — x — $w^T x$ — 1 factor

Linear Model

In Linear models, we assume that the target, y is a linear combination of the attributes or features x. The combination is represented by a weight vector w.

Linear Model

In Linear models, we assume that the target, y is a linear combination of the attributes or features x. The combination is represented by a weight vector w.

In the housing price example

$$y = w_0 + w_1 \times x_1 + w_2 \times x_2 + w_3 \times x_3 + w_4 \times x_4$$

Bias term

Linear Model

In Linear models, we assume that the target, y is a linear combination of the attributes or features x. The combination is represented by a weight vector w.

In the housing price example

$$y = w_0 + w_1 \times x_1 + w_2 \times x_2 + w_3 \times x_3 + w_4 \times x_4$$

In the housing price example

$$1MM = w_0 + w_1 \times 2500 + w_2 \times 4 + w_3 \times 3 + w_4 \times \text{ Bothell}$$

SP sqft #bed #bathroom

Linear Model

In Linear models, we assume that the target, y is a linear combination of the attributes or features x. The combination is represented by a weight vector w.

In the housing price example

$$y = w_0 + w_1 \times x_1 + w_2 \times x_2 + w_3 \times x_3 + w_4 \times x_4$$

In the housing price example

$$1MM = w_0 + w_1 \times 2500 + w_2 \times 4 + w_3 \times 3 + w_4 \times \text{Bothell}$$

There's one problem though!

How do we multiply a 'location' by a weight ?

Dealing with categorical attributes

One approach: Create new dummy attributes!

$x_{Bothell}, x_{Bellevue}, x_{Sammamish}, x_{IssaquahHigh}$ - One dummy variable for each location that takes a value 1 if its the true location and 0 otherwise.

One-hot encoding

$x_{Location} = [\,0 \quad 1 \quad 0 \quad 0\,]$

Bothell, Bellevue, Sammamish, Issaquah

w

$w_{Bothell}$, $w_{Bellevue}$, w_{Samm}, $w_{Issaquah}$

Dealing with categorical attributes

One approach: Create new dummy attributes!

$x_{Bothell}, x_{Bellevue}, x_{Sammamish}, x_{IssaquahHigh}$ - One dummy variable for each location that takes a value 1 if its the true location and 0 otherwise.

ICE #1 (2 mins): How many attribues do we have now?

Let's say our data consisted of the following attributes: Square Footage, # Rooms, # Bathrooms, Location. After applying "pre-processing" to the data of introducing dummy attributes, how many total attributes do we have now ? Answer poll pollev.com/karthikmohan088

a) 10
b) 12
c) 16
d) 15

Room:- 1 & 7
Bath:- 1 & 4
Location:- 4

Context
Rooms, Bathrooms, Location
→ categorical var.

Modifying the Data Matrix

Where we started: X

Index	x_1	x_2	x_3	x_4	y
1	2500	4	3	Bothell	1 MM

Modifying the Data Matrix

Where we started: X

Index	x_1	x_2	x_3	x_4	y
1	2500	4	3	Bothell	1 MM

After pre-processing for categorical attributes: New X

Index	x_1	x_2	x_3	x_4	x_5	x_6	
1							
2							
⋮							

- - - - x_{15} x_{16}

Raw Data → Pre-Processed Data → "Features"

Modifying the Data Matrix

Where we started: X

Index	x_1	x_2	x_3	x_4	y
1	2500	4	3	Bothell	1 MM

After pre-processing for categorical attributes: New X

Index	x_1	x_2	x_3	x_4	x_5	x_6	
1							
2							
⋮							

Does vector y change?

$X_{1000 \times 4} \rightarrow X^{new}_{1000 \times 16}$

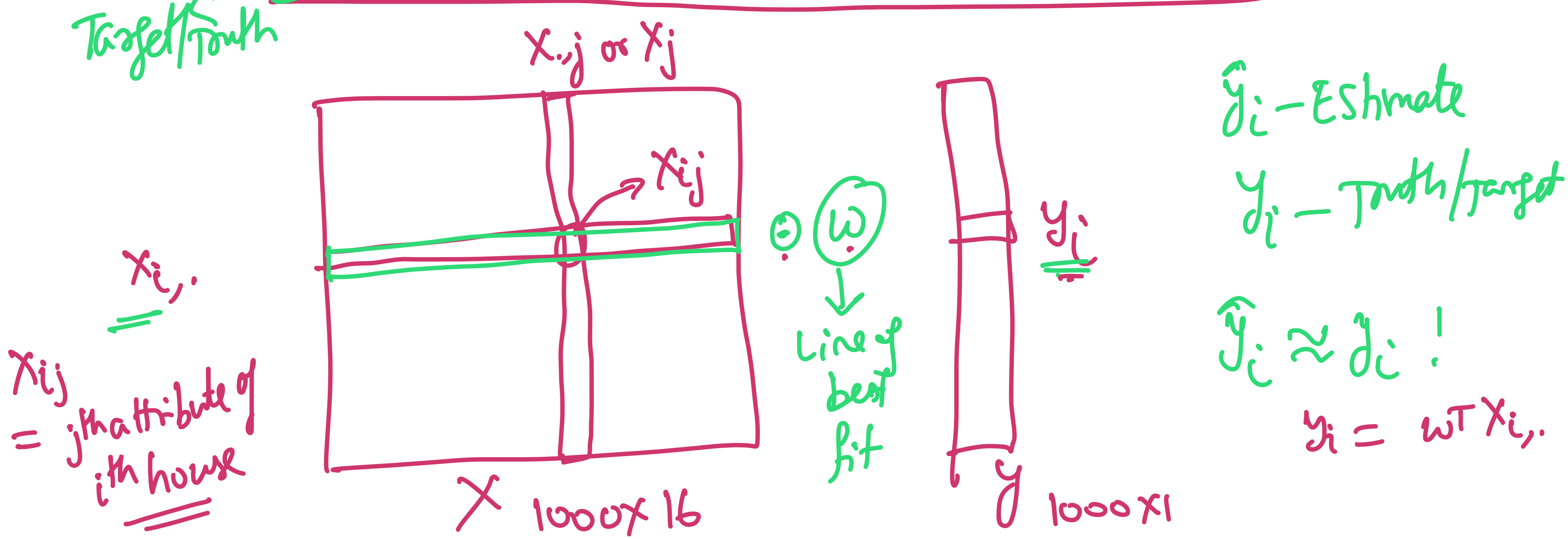
$y_{1000 \times 1} \rightarrow y^{new}_{1000 \times 1}$

Back to the Linear Model

A formula for the house price

Let y_i be the price of the i_{th} home. Let X_{ij} denote the j_{th} attribute of the i_{th} home. Then

$$y_i \sim w_0 + w_1 \times X_{i1} + w_2 \times X_{i2} + w_3 \times X_{i3} + \ldots$$



Back to the Linear Model

A formula for the house price

Let y_i be the price of the i_{th} home. Let X_{ij} denote the j_{th} attribute of the i_{th} home. Then

$$y_i \sim w_0 + w_1 \times X_{i1} + w_2 \times X_{i2} + w_3 \times X_{i3} + \ldots$$

A succinct expression for the i_{th} house

$$\hat{y}_i = w^T X_{i,.} = w \cdot X_{i,.}$$

$$\underbrace{[\cdot \ \cdot]}_{w^T} \underbrace{\begin{bmatrix} \vdots \end{bmatrix}}_{X_{i,.}} = w^T X_{i,.} = \hat{y}_i$$

Back to the Linear Model

A formula for the house price

Let y_i be the price of the i_{th} home. Let X_{ij} denote the j_{th} attribute of the i_{th} home. Then

$$y_i \sim w_0 + w_1 \times X_{i1} + w_2 \times X_{i2} + w_3 \times X_{i3} + \ldots$$

A succinct expression for the i_{th} house

$$y_i = w^T X_{i,\cdot} = w \cdot X_{i,\cdot}$$

ICE #2 (2 mins): Succinct expression for y in terms of X and w?

Linear Regression: Putting it all Together

Definition

Find the best weights/parameters/coefficients w such that $X_{i,\cdot}^T w$ is as close to y_i as possible! $\forall i$

Linear Regression: Putting it all Together

Definition

Find the best weights/parameters/coefficients w such that $X_{i,\cdot}^T w$ is as close to y_i as possible!

Mathematically

Minimize the following expression:

$$\min_w \|\underbrace{Xw}_{\hat{y}} - y\|_2^2$$

$$\|z\|_2^2 = z_1^2 + z_2^2 + \dots + z_N^2$$

N → # houses in your data

$\hat{y} = y$

In practice $\hat{y} \neq y$

$\hat{y} \approx y$

$$\frac{1}{N}\|\hat{y} - y\|_2^2 = \frac{1}{N}\left[(\hat{y}_1 - y_1)^2 + (\hat{y}_2 - y_2)^2 + (\hat{y}_3 - y_3)^2 + \dots + (\hat{y}_N - y_N)^2\right] \rightarrow \text{MSE}$$

Linear Regression: Putting it all Together

Definition

Find the best weights/parameters/coefficients w such that $X_{i,\cdot}^T w$ is as close to y_i as possible!

Mathematically

Minimize the following expression:

$$\min_w \|Xw - y\|_2^2$$

Estimate or "learned" parameter

Represented usually by $\hat{w}$ and $\hat{y}$ is the "predicted" house price for all the homes.

Linear Regression: Putting it all Together

Definition

Find the best weights/parameters/coefficients w such that $X_{i,\cdot}^T w$ is as close to y_i as possible!

Mathematically

Minimize the following expression:

$$\min_{w} \|Xw - y\|_2^2$$

Estimate or "learned" parameter

Represented usually by $\hat{w}$ and $\hat{y}$ is the "predicted" house price for all the homes.

ICE #3 (1 min)

What's the succinct expression for $\hat{y}$?

Line of best fit

Best fit

$\hat{w}$ defines the line of best fit. $h(x) = \hat{w}^T x$ gives us the line and in higher dimensions, it's called a "hyperplane".

Line of best fit

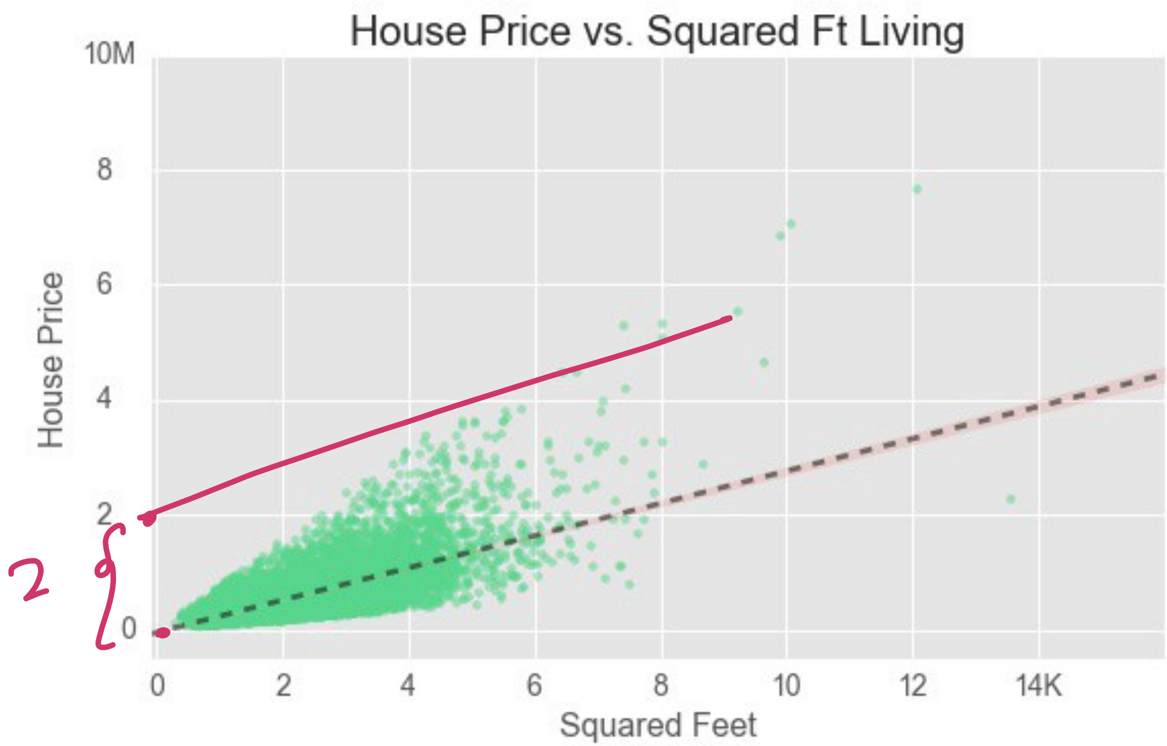
Best fit

$\hat{w}$ defines the line of best fit. $h(x) = \hat{w}^T x$ gives us the line and in higher dimensions, it's called a "hyperplane".

Housing price example

Line of best fit

Housing price example

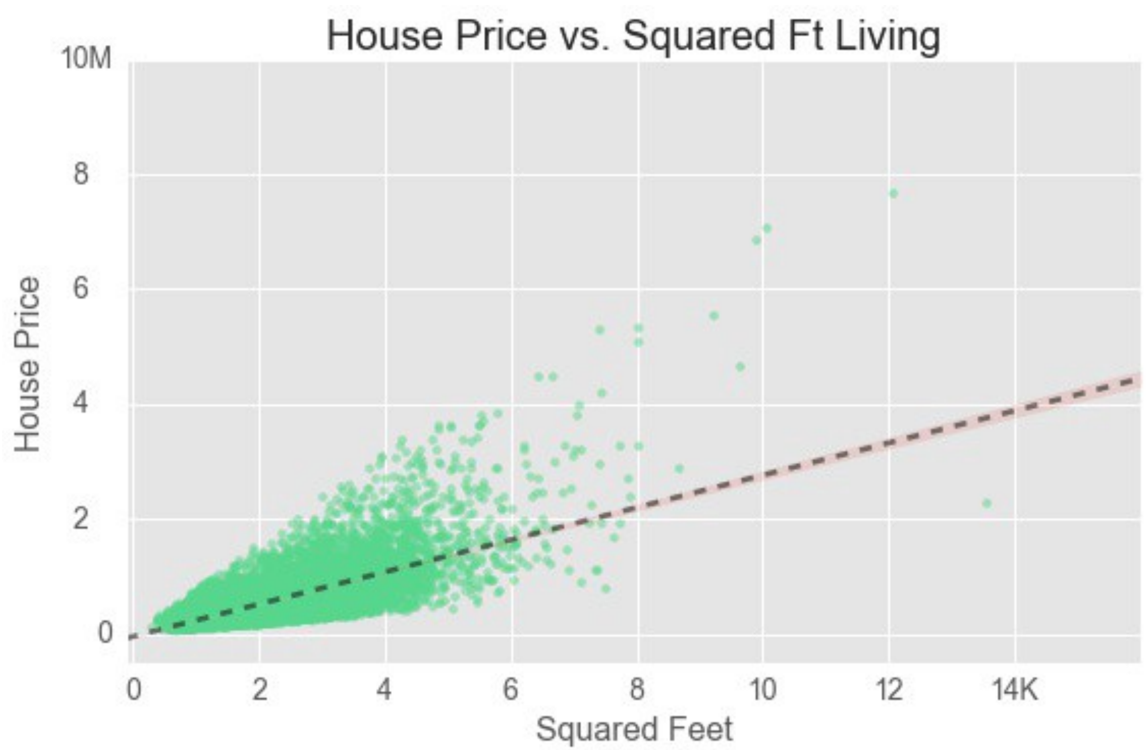


ICE #4 (1 min)

What would you say is the value of the bias, w_0 for the line in the visual above?

Hyperplane in 3 dimensions

Housing price example



3 dim hyperplane

$$\hat{y} = w_0 + w_1 \times x_1 + w_2 \times x_2$$

x_1 could be square footage and x_2 could be number of bedrooms.

Linear Regression

Closed form!

There is actually a closed form expression for Linear Regression!

$$\min_{w} \|Xw - y\|_2^2$$

$\hat{w} = (X^T X)^{-1} X^T y$! (Q: How do we arrive at this?)

Linear Regression

Closed form!

There is actually a closed form expression for Linear Regression!

$$\min_w \|Xw - y\|_2^2$$

$\hat{w} = (X^TX)^{-1}X^Ty$! (Q: How do we arrive at this?)

In practice!

In practice, a linear regression library might revert to doing "gradient descent" on the learning objective. Why do that?

Housing price Example

Pre-processing of data

One is taking care of categorical variables such as location with dummy attributes (also called 'bag of words' model). Anything else we may need to do on the data to get good predictions?

Training the Linear Regression Model

x_1	x_2	x_3	x_4	x_5	x_6	x_7	y

Can we use of all of data for training?

- Why not use all data for training ?

The phenomenon of Overfitting

Overfitting

Overfitting is when your mdoel performs great on training data but doesn't match up on test data. To account for overfitting, we also have a validation data set.

Understanding over-fitting better

When do we expect over-fitting?

When the number of attributes in our model exceeds the size of the data set.

In terms of data matrix X

$\#$ rows $<<$ $\#$ columns

Data Splits for Machine Learning

- **Training** Learning parameters/weights, i.e. w for Linear Regression is called Training.

Data Splits for Machine Learning

- **Training** Learning parameters/weights, i.e. w for Linear Regression is called Training.
- We don't use all data for training - Some portion of the data is kept for validation and testing.

Data Splits for Machine Learning

- **Training** Learning parameters/weights, i.e. w for Linear Regression is called Training.
- We don't use all data for training - Some portion of the data is kept for validation and testing.
- **Data Splits:** Usually, 80% of data is kept for training, 10% for validation and 10% for training. The splits are chosen randomly.

Data Splits for Machine Learning

- **Training** Learning parameters/weights, i.e. w for Linear Regression is called Training.
- We don't use all data for training - Some portion of the data is kept for validation and testing.
- **Data Splits:** Usually, 80% of data is kept for training, 10% for validation and 10% for training. The splits are chosen randomly.
- Why not use all data for training ?

Data Splits for Machine Learning

- **Training** Learning parameters/weights, i.e. w for Linear Regression is called Training.
- We don't use all data for training - Some portion of the data is kept for validation and testing.
- **Data Splits:** Usually, 80% of data is kept for training, 10% for validation and 10% for training. The splits are chosen randomly.
- Why not use all data for training ?
- Why not just have **train** and **test** data? What's the point of validation data set?

Example: 70 : 10 : 20 Train-Val-Test data split

Choose 70% train data at random

x_1	x_2	x_3	x_4	x_5	x_6	x_7	y

Example: 70 : 10 : 20 Train-Val-Test data split

Add 20% test data at random

x_1	x_2	x_3	x_4	x_5	x_6	x_7	y

Example: 70 : 10 : 20 Train-Val-Test data split

Remainder becomes validation data

x_1	x_2	x_3	x_4	x_5	x_6	x_7	y

Coding Pointers

- **Pandas** library in Python is good for data pre-processing before training your Linear Regression model

Coding Pointers

- **Pandas** library in Python is good for data pre-processing before training your Linear Regression model
- **Dummy attributes** for categorical variables can also be added in through `pandas.get_dummies()` method.

Coding Pointers

- **Pandas** library in Python is good for data pre-processing before training your Linear Regression model
- **Dummy attributes** for categorical variables can also be added in through `pandas.get_dummies()` method.
- Use **Scikit-learn** for implementing Linear Regression

Coding Pointers

- **Pandas** library in Python is good for data pre-processing before training your Linear Regression model
- **Dummy attributes** for categorical variables can also be added in through `pandas.get_dummies()` method.
- Use **Scikit-learn** for implementing Linear Regression
- Should now be ready to tackle both the conceptual and programming Assignment 1!

Summary so far

- Linear Regression finds a line of best fit through the data.
- R^2 measure determines the goodness of fit.
- Usually multiple good attributes are needed for a good prediction and a good fit.
- Data pre-processing. Categorical attributes are handled through creation of dummy attributes and in addition normalizing of the attributes brings all attributes on the same scale for regression.
- We have a closed form/analytical solution for Linear Regression, but for large data sets, gradient descent algorithm (iterative) gets used for scalability reasons.
- We don't use all of a data set for training. A portion of data is kept for validation and testing. This is to prevent over-fitting and also for fair evaluation purposes.
- The data set split is usually $80-10-10$ or $70-10-20$ (train-val-test).

Summary so far

- Over-fitting happens when we have fewer data points as compared to the number of attributes or features.
- Over-fitting can be taken care off by increasing data-set size, decreasing number of attributes or through regularization strategies

Questions/Thoughts?