Data Science in Business:
Data Structures & Analysis

Dr. Peter Molnar

Sarah Zeis

Carly Wieting

Georgia State University | J. Mack Robinson College of Business

Robinson

Course Overview

Class 1: Introduction to Machine Learning and Set-Up Python

Class 2: Data Exploration

Class 3: Machine Learning Models (Decision Tree and KNN)

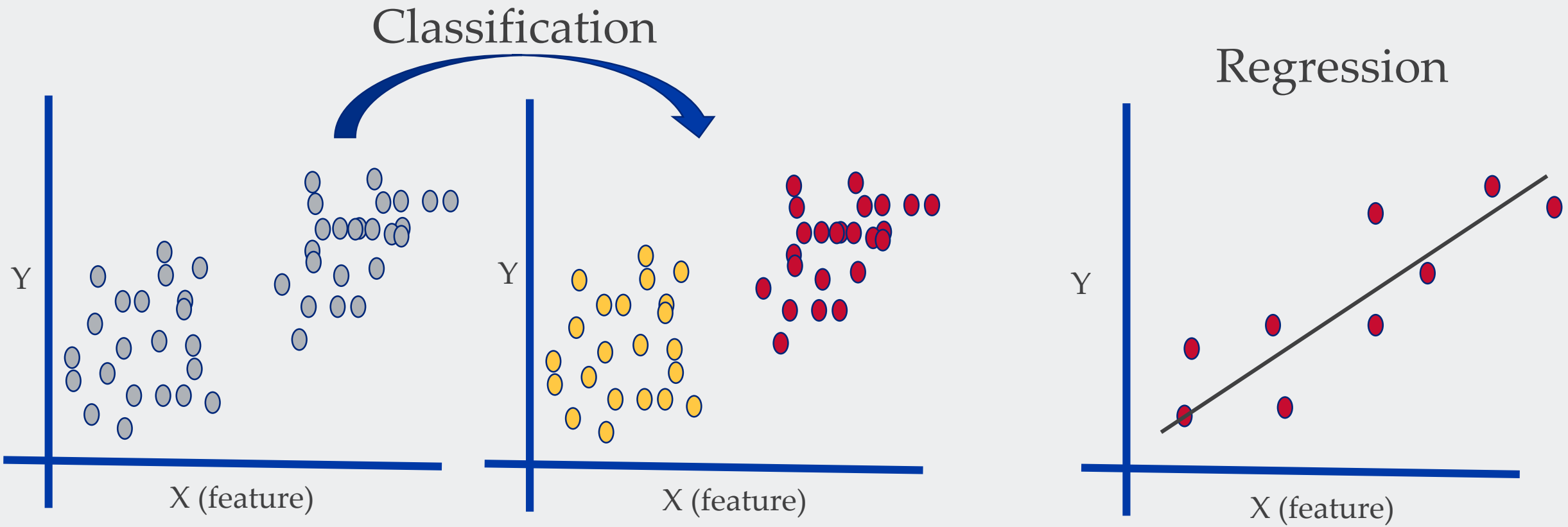
Class 4: Analyze Celebrity Tweets

Class 5: Forecasting with Facebook Prophet

Goals and Takeaways

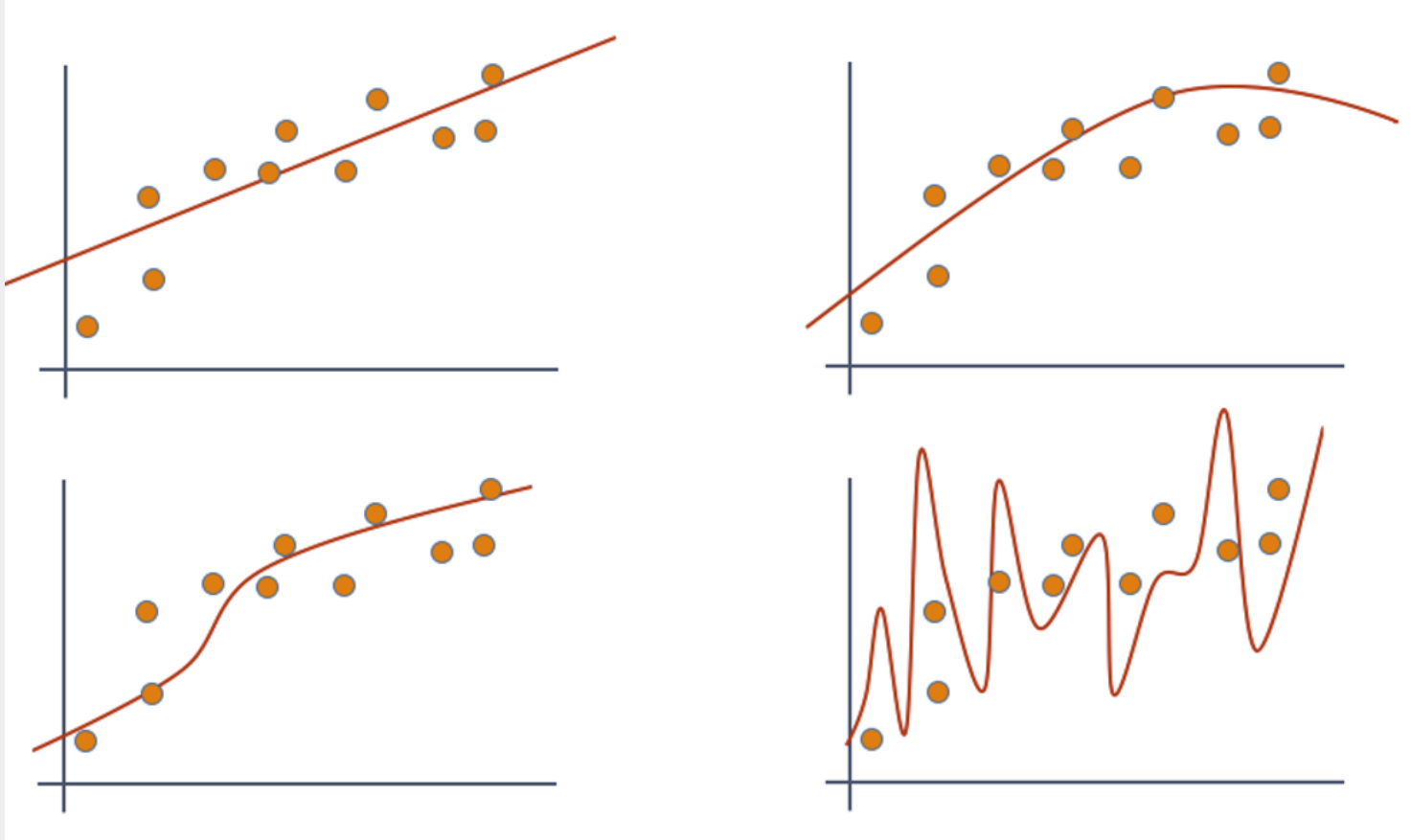
- Introduce train/test datasets and understand their importance
- Why feature selection is important and its challenges
- Deploy KNN, decision tree, and XGBoost machine learning algorithms in python

Review - Classification and Regression



Which is the best model?

Which are overfitted, underfitted, or just right?



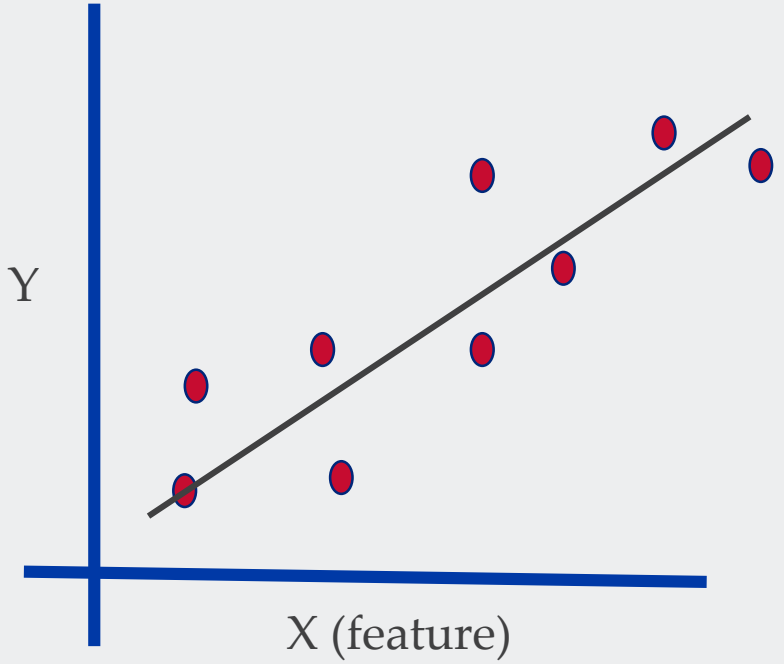
Test and Train Data Sets

In every supervised machine learning algorithm, you divide the dataset into two parts: train and test sets

You build your model based on the training set, and then use the test set to see how good is your model's predictive performance by measuring an error

Typically the ratio for train:test is roughly 3:1

$$L(\theta) = \sum_i (y_i - \hat{y}_i)^2$$



The black line is the model that you build based on the training set.

The red points are data points from the test set. You can measure the error by calculating the distance from the point to the line and squaring it.
(It's important to square the result otherwise the summation will be zero.)

Feature Selection

Some features are useless and you do not want to incorporate them into your model.

There are 2^(# of features) number of possible features to choose from.

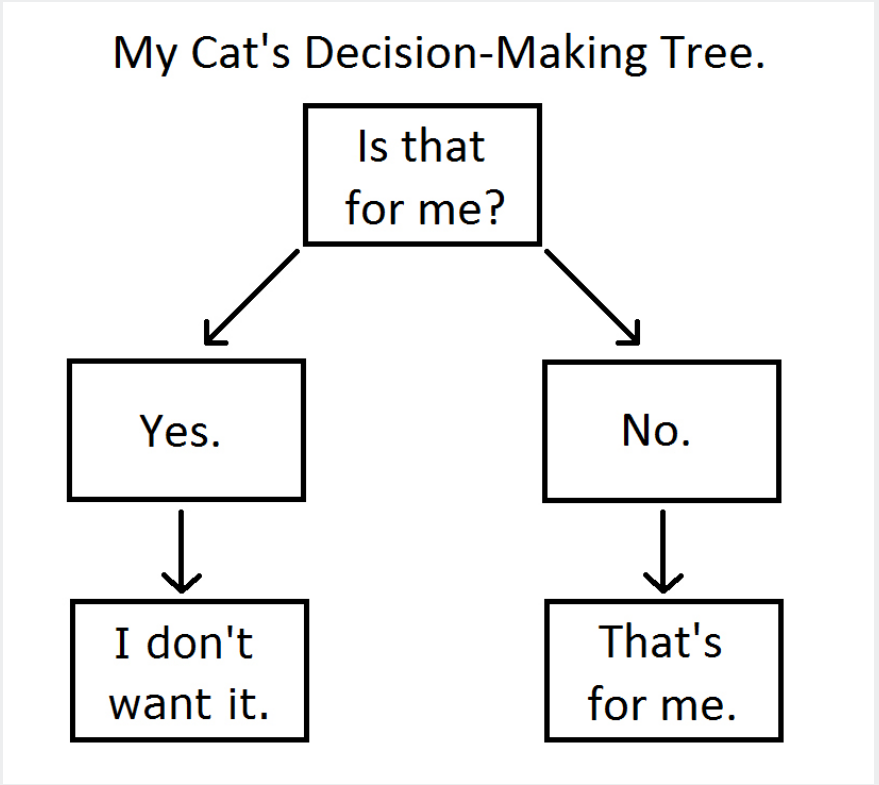
Forward selection starts with one feature, then two. If the second feature is useless (low p-value), then it removes it and adds another feature. If the next feature is useful, then you keep it. This process continues until all features are checked.

Backward selection is similar to forward, but in the opposite direction. Start with all features then remove them one by one.

Id	OverallQu	MSSubCla	MSZoning	LotFronta	LotArea	Street	Alley	LotShape	La
1	7	60	RL	65	8450	Pave	NA	Reg	Lv
2	6	20	RL	80	9600	Pave	NA	Reg	Lv
3	7	60	RL	68	11250	Pave	NA	IR1	Lv
4	7	70	RL	60	9550	Pave	NA	IR1	Lv
5	8	60	RL	84	14260	Pave	NA	IR1	Lv
6	5	50	RL	85	14115	Pave	NA	IR1	Lv
7	8	20	RL	75	10084	Pave	NA	Reg	Lv
8	7	60	RL	NA	10382	Pave	NA	IR1	Lv
9	7	50	RM	51	6120	Pave	NA	Reg	Lv
10	5	190	RL	50	7420	Pave	NA	Reg	Lv

Housing dataset has 55 features…
surely not all improve the model


KNN BE LIKE
"Show me your friends, and I'll tell you who you are."


My Cat's Decision-Making Tree.
Is that for me?
Yes.
No.
I don't want it.
That's for me.

K-Nearest Neighbor (KNN)

It uses both classification and regression!

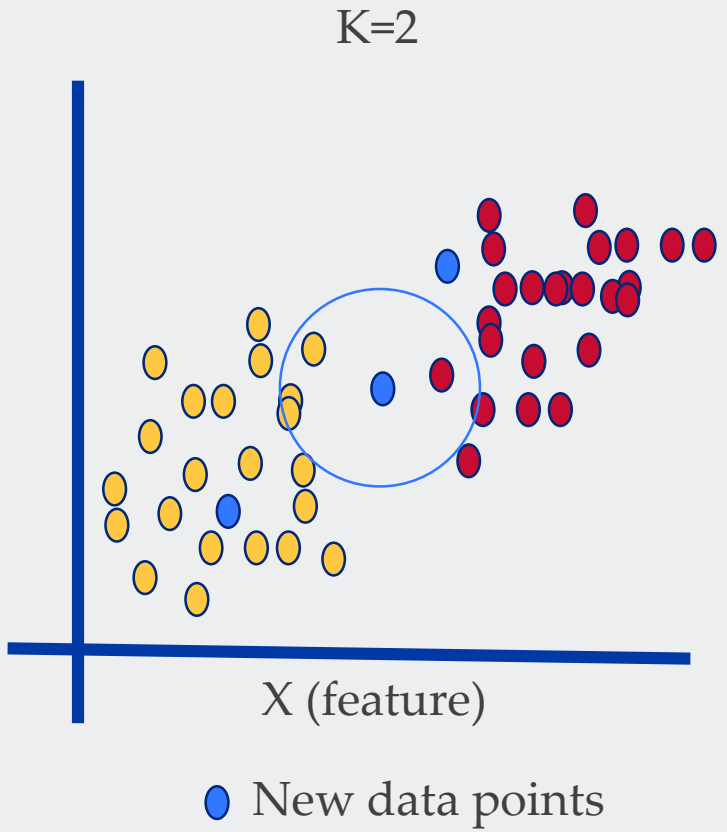
**Notice two quantitative and one categorical variables

KNN assumes that things near each other are similar.

<u>Process:</u>

- Start with structured dataset with several features and a label
  - The labels define which are red and which are yellow.
- The distance between a new data point and its nearest neighbors is calculated or there is a vote.
- Finally, the data point is given a value.

Examples: Using age and loan money to predict credit card default, using length and width to predict flower type

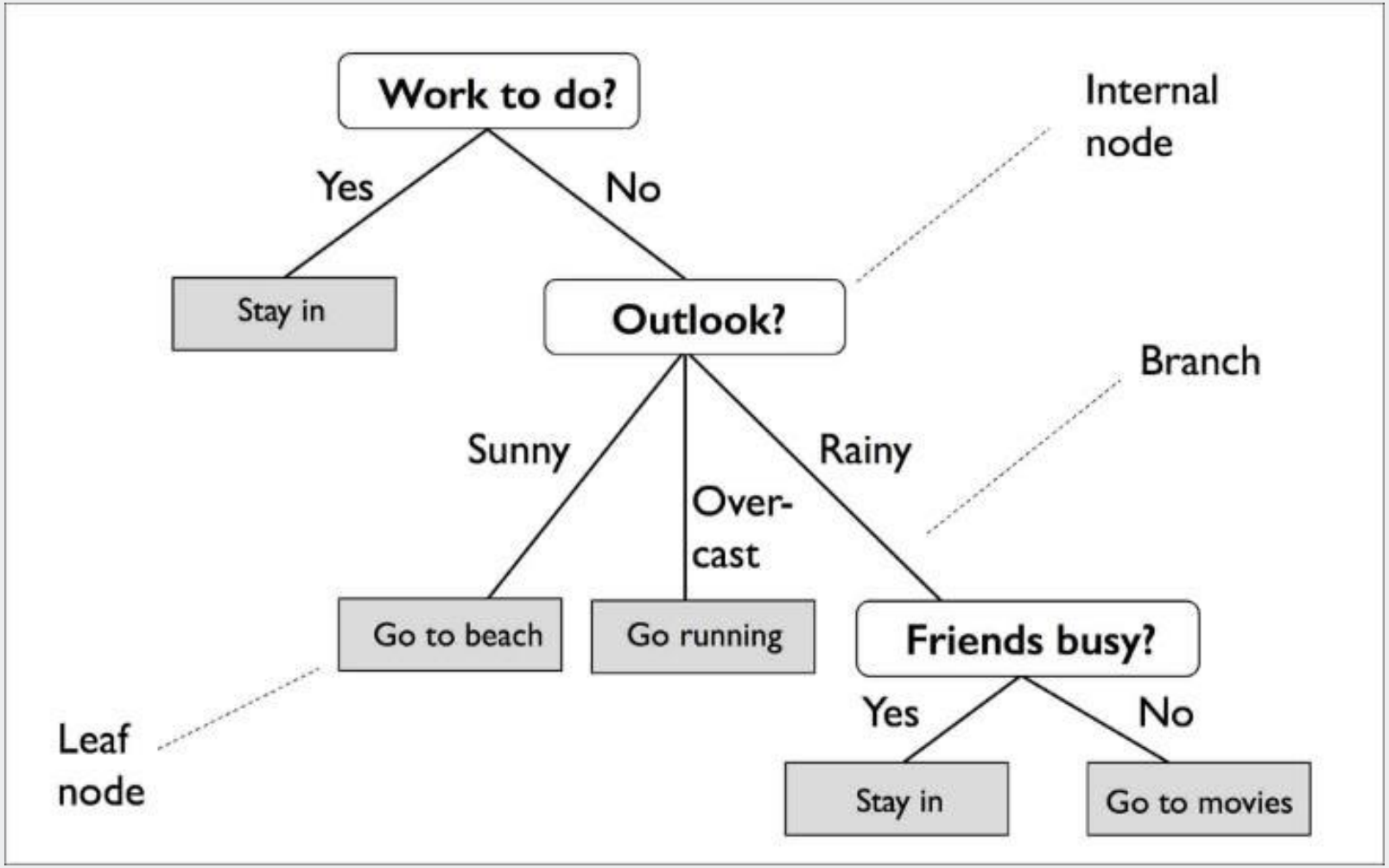


Decision Tree

Each data observation is run through the tree, and the output (leaf nodes) is analyzed.

Categorical and numerical values can be used as inputs.

It's the same as the decision tree that businesses use to choose strategy.



Extreme Gradient Boosting (XGBoost)

Boosted gradient descent decision trees

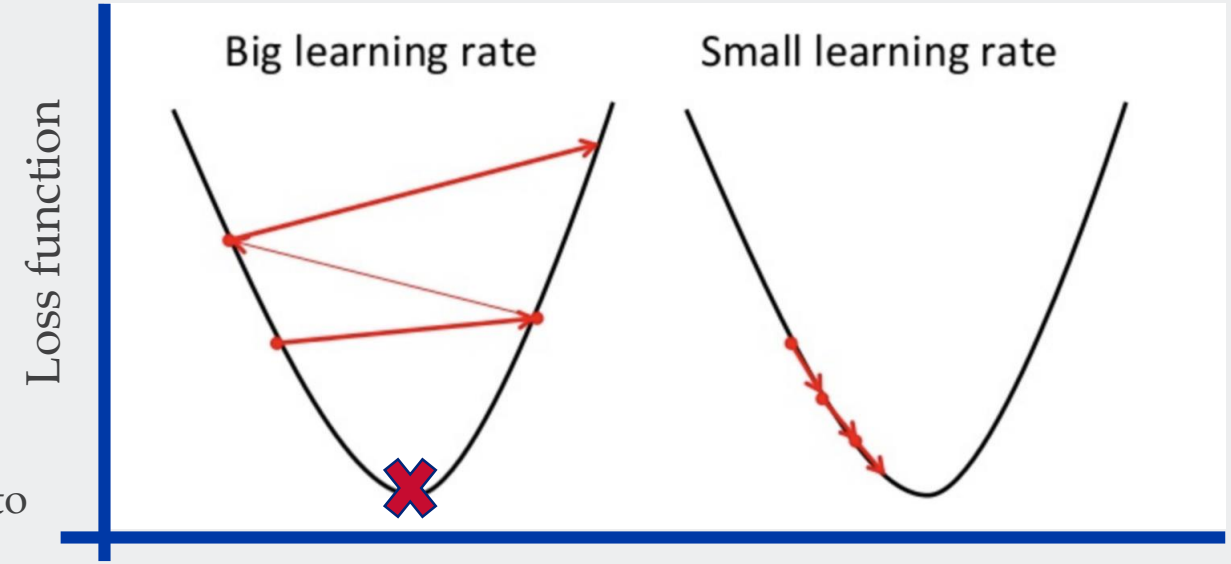
Supervised learning

High computational power and model performance

Performs regression and classification problems

Boosting is an ensemble technique where new models are added to correct the errors made by existing models until no further improvements can be made.

Gradient boosting is an approach where new models are created that predict the residuals or errors of prior models and then added together to make the final prediction. It is called gradient boosting because it uses a gradient descent algorithm to minimize the loss when adding new models.



Gradient descent minimizes the cost function to give the best performance

https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/