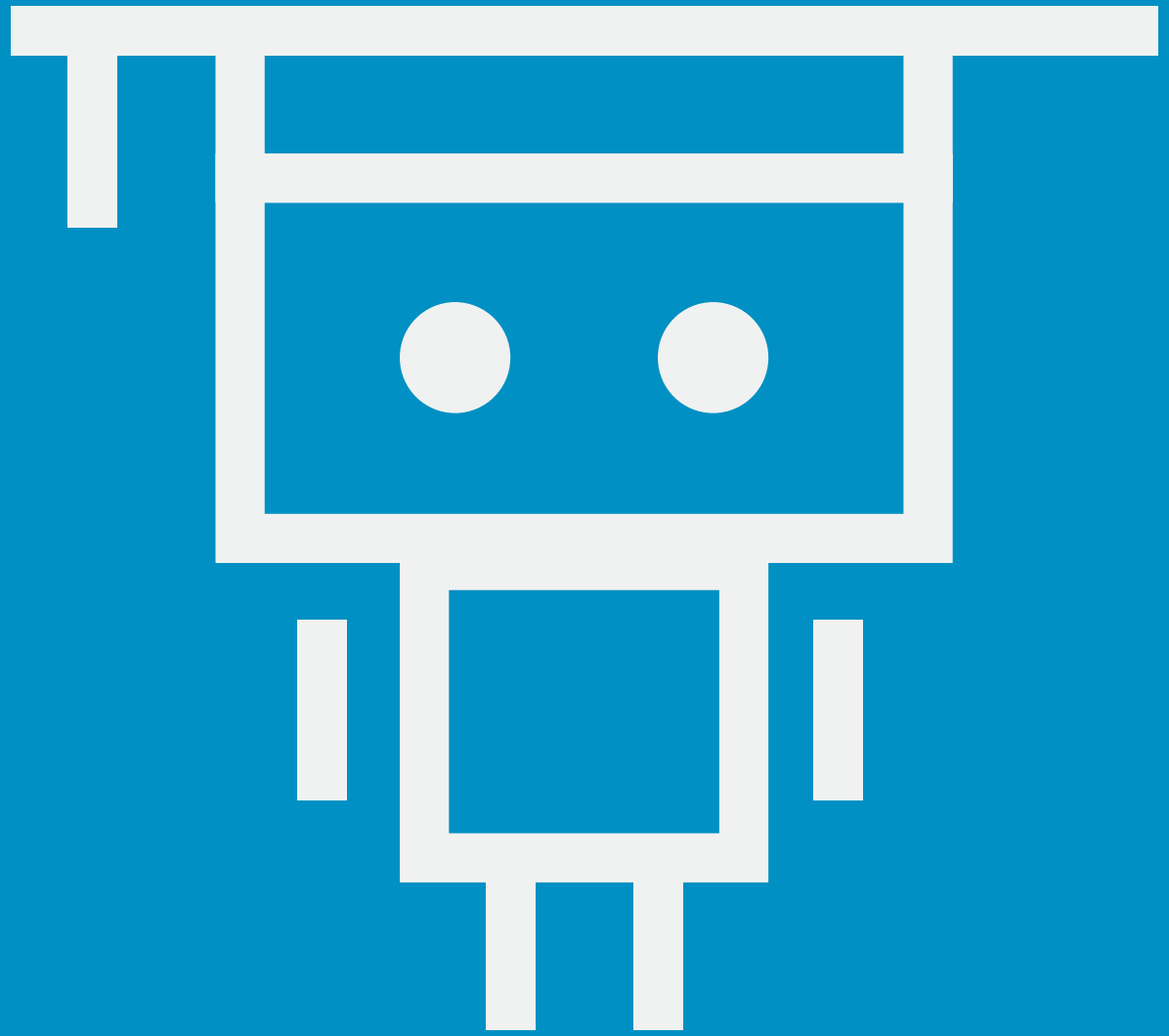


Course Description

- Covers techniques for managing data throughout an end-to-end ML process
- Learn statistical analysis and visualization techniques, including methods for detecting and remedying overfitting
- Gain familiarity with tools in standard ML and data processing libraries

Class Deliverables

Final Project: Apply and experiment with concepts and techniques from course on a real-world Amazon dataset. Completion of this course is based on your Leaderboard submission.

Submission page: https://leaderboard.corp.amazon.com/tasks/478

Submission is open until the next day after day 3 (day 4) at 5:00 PM (PST)

After the competition,

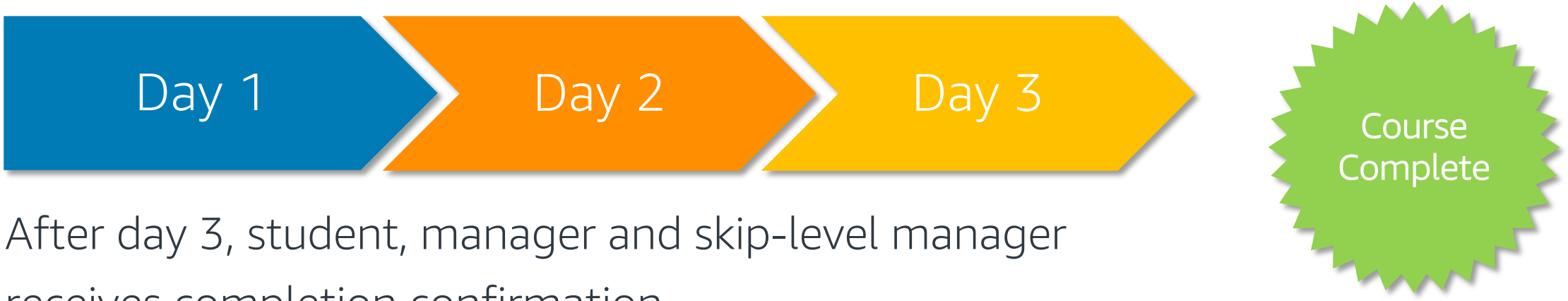
- All competitors will get *ML Accelerator Grad phone tool icon!* 

- The top 3 submissions will get a *MLA Champion phone tool icon!* 

Class Deliverables

You will get a score after each submission. You can improve your score by making multiple submissions (no upper limit on number of submissions).

- **Expectations:** Submit a model to Leaderboard after *each* Day
- **Requirements:** Submit 1+ model. At least one submission is **REQUIRED**
- **Non-completion:** Submit 0 models to the Class Leaderboard



After day 3, student, manager and skip-level manager receives completion confirmation

Topics for today

- Introduction to ML
- Evaluation Metrics for ML problems
- Tools and Libraries
- Exploratory Data Analysis
- Model Development
- Final Project

Topics for today

- Introduction to ML
- Evaluation Metrics for ML problems
- Tools and Libraries
- Exploratory Data Analysis
- Model Development
- Final Project

Sample Problem – Food Delivery Prediction

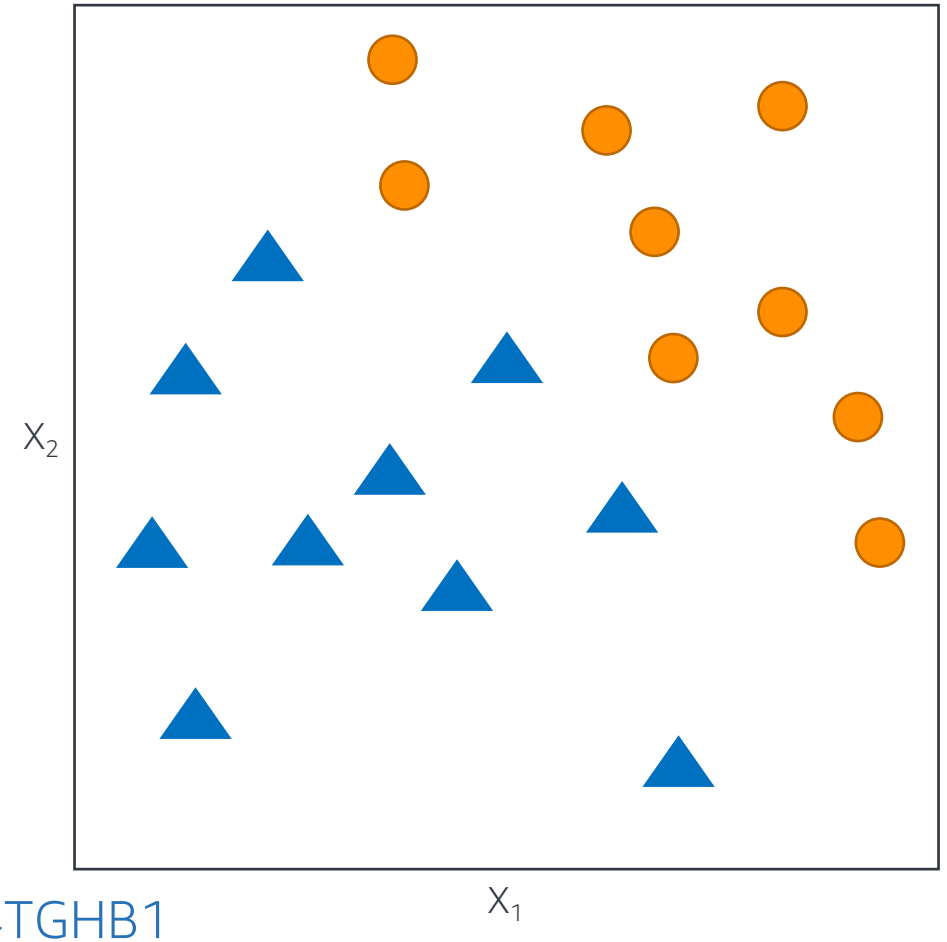
- John loves to order his food online for home and work. His office is in an urban area an his house is away from urban area.
- John wants to predict whether his order will be on time or late beforehand.
- He logged his previous 45

orders like this:

IsBadWeather?	IsRush-hour?	Mile distance from restaurant	IsUrbanAddress?	Late
0	1	5	1	0
1	0	7	0	1
0	1	2	1	0
1	1	4.2	1	1
0	0	7.8	0	0
..	..	..	..	..

Simple ML Model: K Nearest Neighbors (KNN)

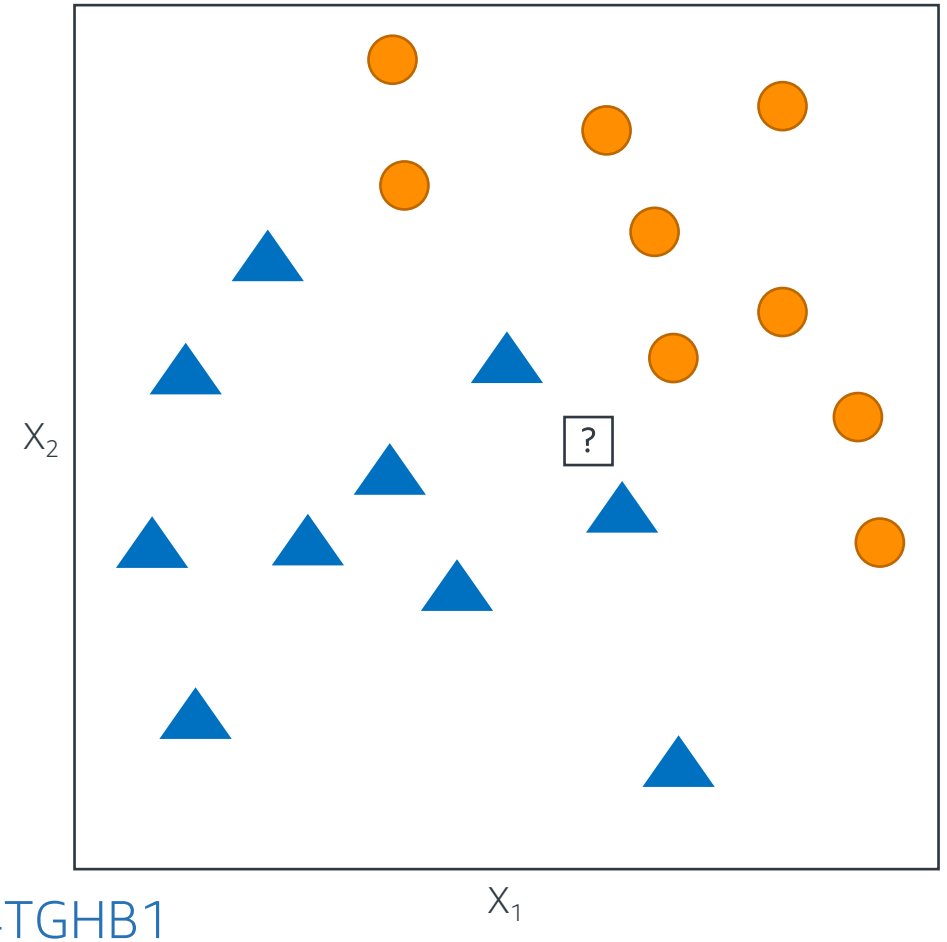
- KNN Algorithm classifies a record by comparing it to its nearest neighbors.
- K is the number of nearest neighbors we will consider.
- In this context: Physically closer ≈ Similar Record
- Follow this notebook:



https://eider.corp.amazon.com/sazaracs/notebook/NB8JDU4TGHB1

Simple ML Model: K Nearest Neighbors (KNN)

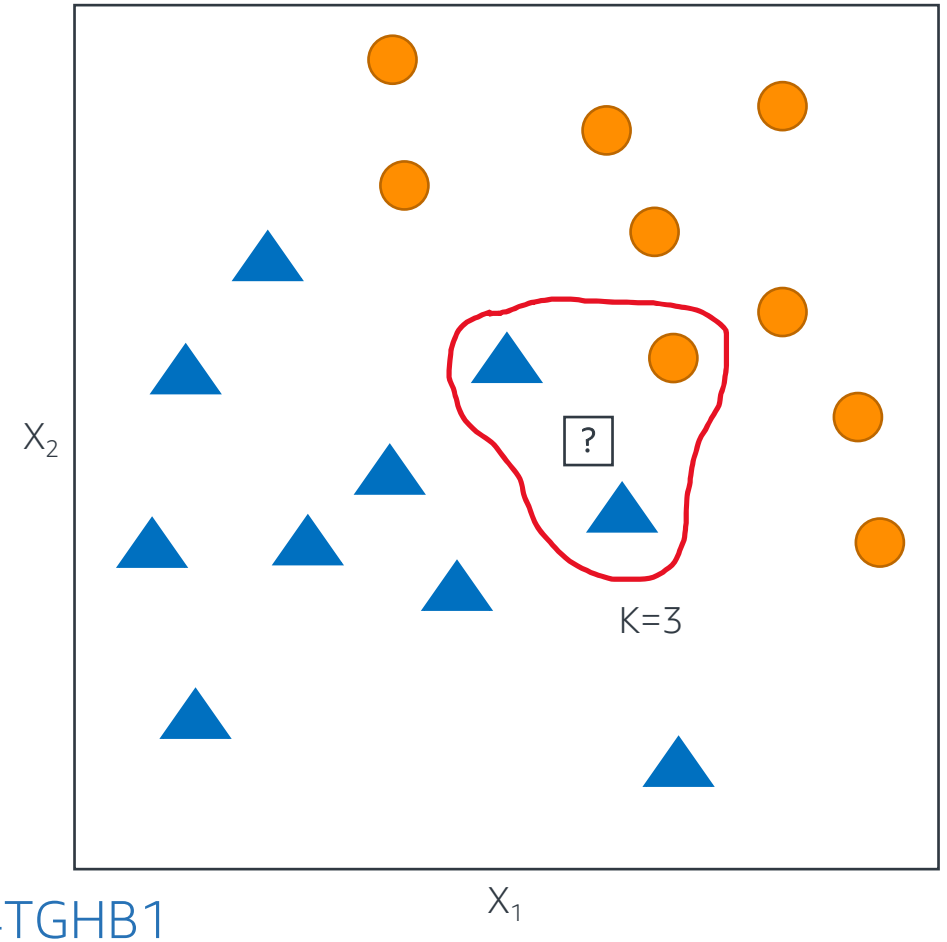
- KNN Algorithm classifies a record by comparing it to its nearest neighbors.
- K is the number of nearest neighbors we will consider.
- In this context: Physically closer ≈ Similar Record
- Follow this notebook:



https://eider.corp.amazon.com/sazaracs/notebook/NB8JDU4TGHB1

Simple ML Model: K Nearest Neighbors (KNN)

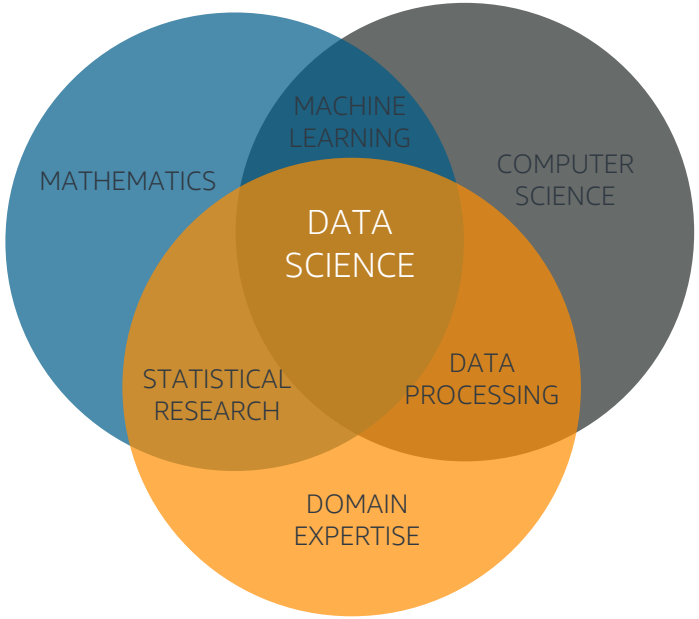
- KNN Algorithm classifies a record by comparing it to its nearest neighbors.
- K is the number of nearest neighbors we will consider.
- In this context: Physically closer ≈ Similar Record
- Follow this notebook:



https://eider.corp.amazon.com/sazaracs/notebook/NB8JDU4TGHB1

What is Data Science?

Wikipedia describes Data Science as processes and systems to extract knowledge or insights from data, either structured or unstructured.



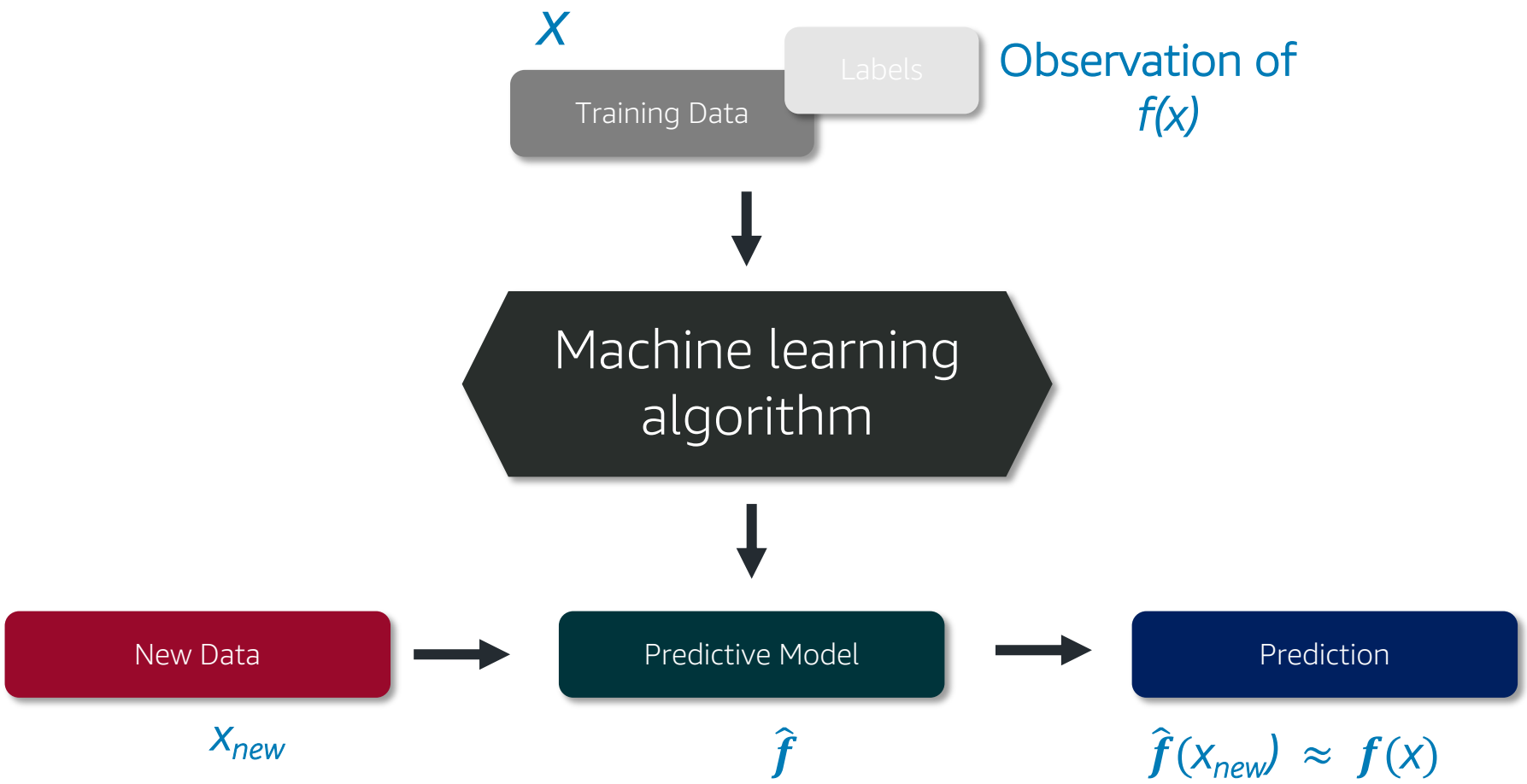
What is Machine Learning

Main idea: Learning = estimating underlying function f mapping data attributes to some target value

Training set: A set of labeled examples $(x, f(x))$ where x is the input variables and the label $f(x)$ is the observed target truth

Goal: Given a training set, find approximation $\hat{f}$ of f that best generalizes, i.e., predicts labels for new examples

Machine Learning Lifecycle



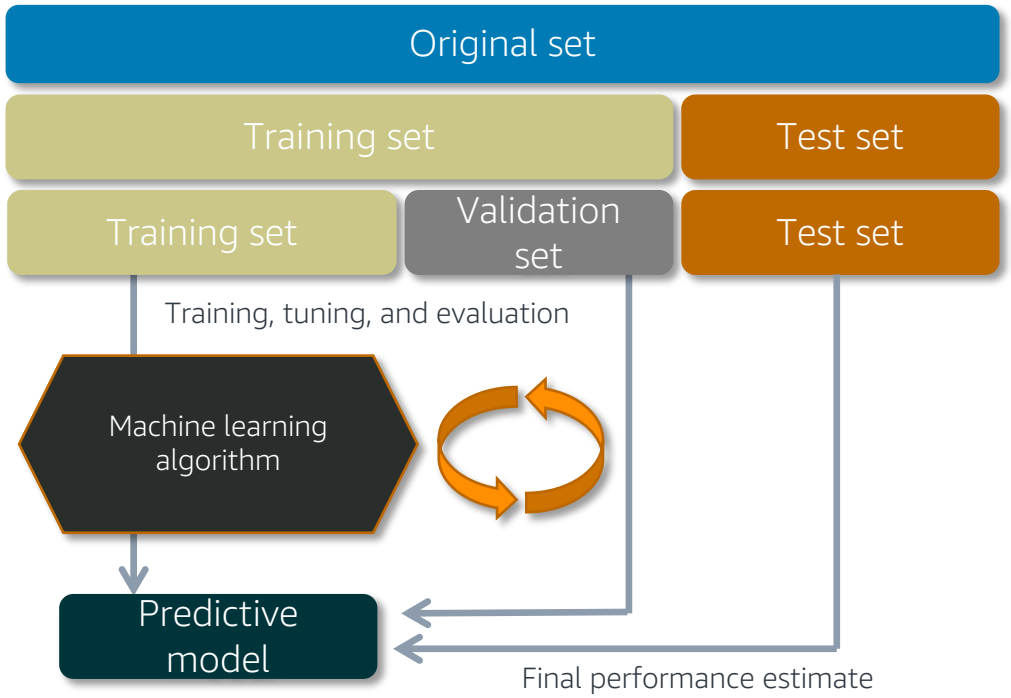
Why Machine Learning?

Some programs are difficult or impossible to write explicitly

- We may not know how (task too complex), e.g., face recognition
- Too much relevant data, e.g., stock market prediction
- Relevant information may only become available dynamically, e.g., a recommendation system

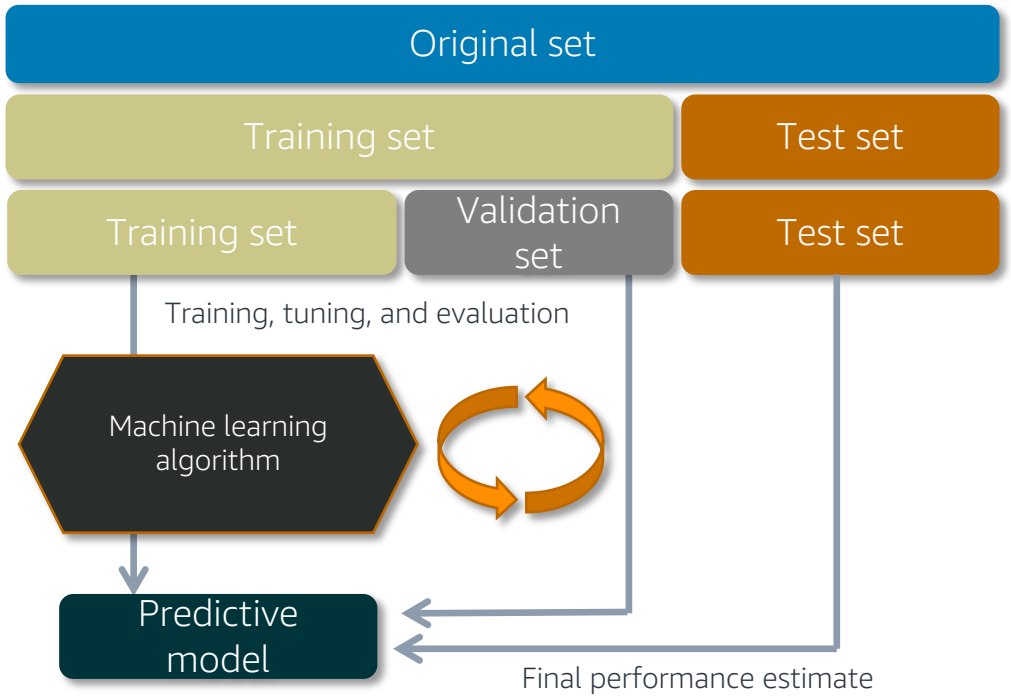
Humans handle this by improving based on experience (data)

Training - Validation - Test datasets



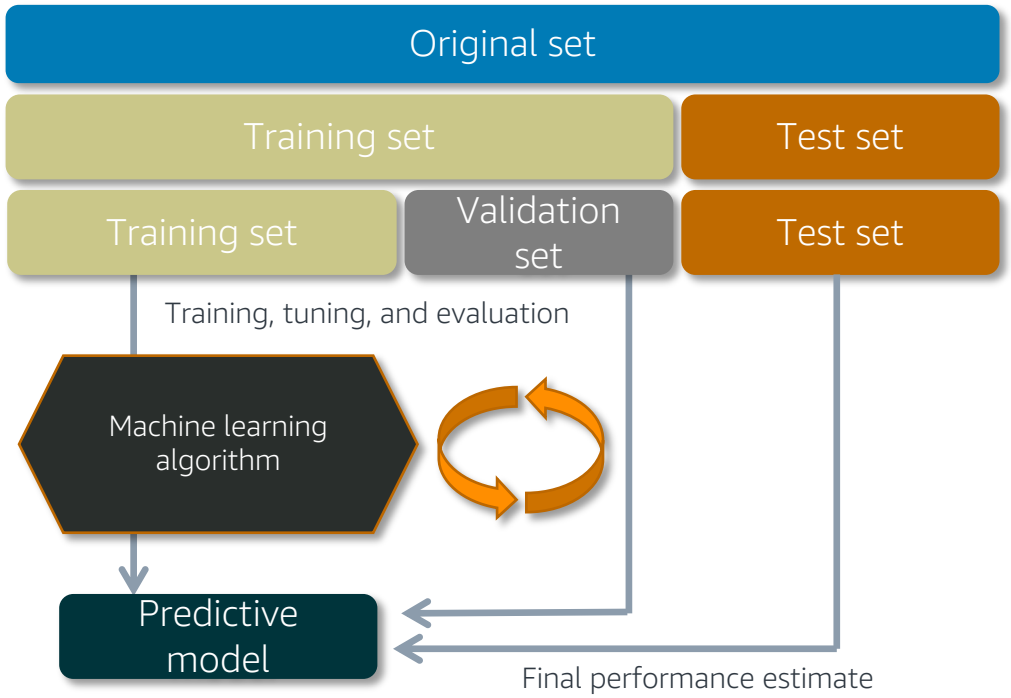
Why do we divide the data into these three sub-datasets?
How do we make sure our model generalizes well?

Training - Validation - Test datasets



Training set: Used to learn the relationship and train the ML model.
Validation set: Used to for unbiased evaluation of the ML model.
Test set: Independent final evaluation of the ML model.

Training - Validation - Test datasets



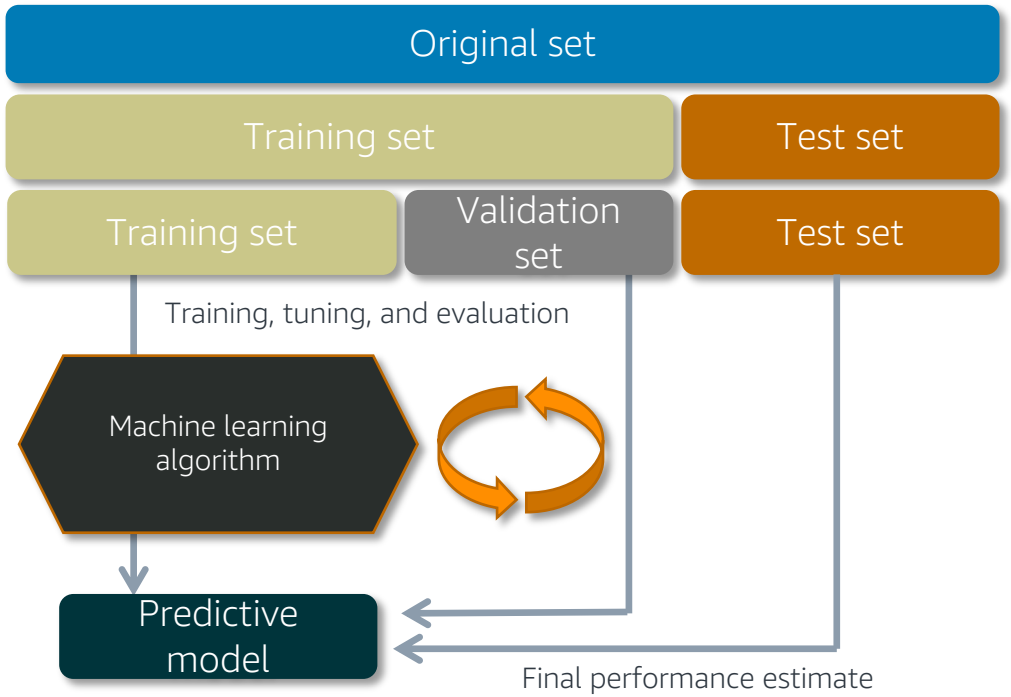
id	uid	title	text	isPositive
0	24124	This is the first time...	Stay away! This seller does ...	0
1	13244	Great	I simply fell in love with this ...	1
2	1244	One star	Terbbile customer service ...	0
3	5515	Needed help	It wasn't easy to mount, but..	0
4	55919	Love it!	This seller is very nice. He ...	1
5	9124	The best	I have always been a fan of..	1
6	3215	Disappointing	It couldn't last more than ...	0
7	98512	Great flavor	I found it much tastier than..	1
8	75310	Great Deal	This products usually sells ...	1
9	28192	Came broken	I don't know how long it ..	0
10	2810	Good product	I was hesitant to buy this pro..	1

Training

Validation

Test

Training - Validation - Test datasets



id	uid	title	text	isPositive
0	24124	This is the first time...	Stay away! This seller does ...	0
1	13244	Great	I simply fell in love with this ...	1
2	1244	One star	Terbbile customer service ...	0
3	5515	Needed help	It wasn't easy to mount, but..	0
4	55919	Love it!	This seller is very nice. He ...	1
5	9124	The best	I have always been a fan of..	1
6	3215	Disappointing	It couldn't last more than ...	0
7	98512	Great flavor	I found it much tastier than..	1
8	75310	Great Deal	This products usually sells ...	1
9	28192	Came broken	I don't know how long it ..	0
10	2810	Good product	I was hesitant to buy this pro..	1

Training

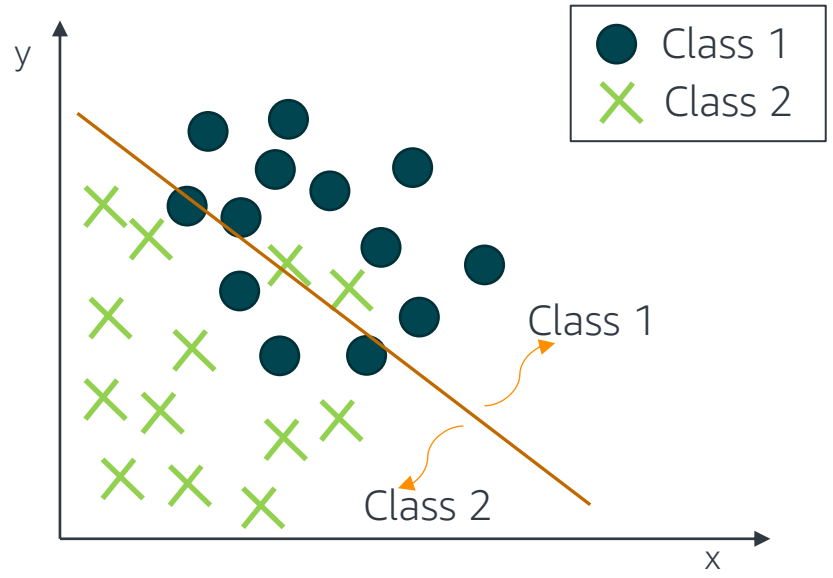
Validation

Test

It is usually a good idea to shuffle our rows before the split to make sure we don't have any bias in the dataset.

Model Evaluation: Underfitting

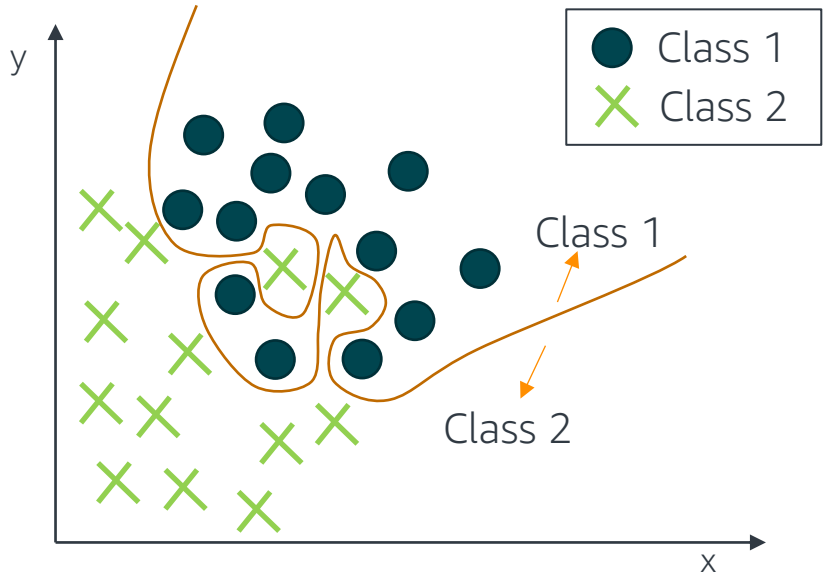
Underfitting: The model is not good enough to describe the relationship between the input data (x) and output (y).



- The model is too simple to capture the input/output relationship.
- It will have poor training and test performance.

Model Evaluation: Overfitting

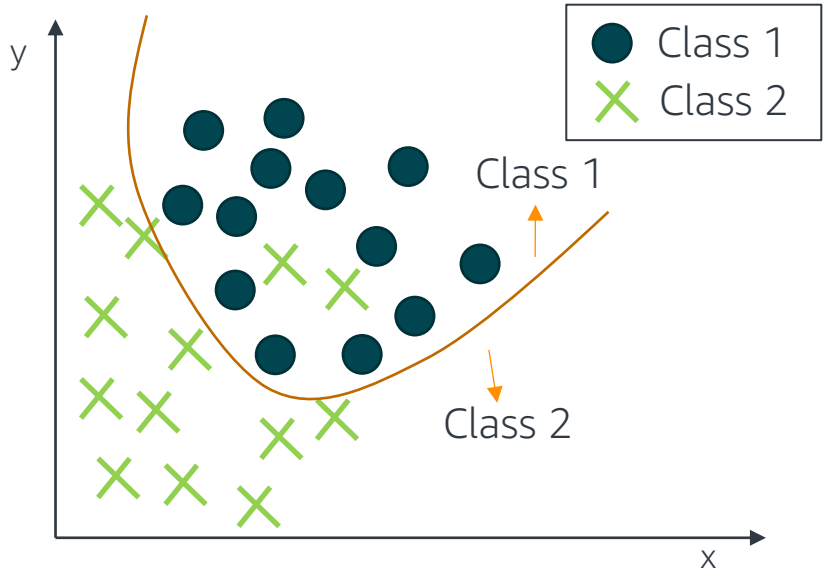
Overfitting: Model memorizes or imitates training data and doesn't generalize well with new "unseen" data (test data).



- Having too complex models for simple problems can cause overfitting.
- The model picks up the noise instead of the underlying relationship.
- We will see good scores in training data but poor in test data.

Model Evaluation: Appropriate Fitting

Appropriate fitting: It captures the general relationship between the input data (x) and output (y).



Overfitting example

- Let's use our initial food delivery example one more time
- We will fit a K Nearest Neighbors model and analyze overfitting.
- Follow this notebook:

  https://eider.corp.amazon.com/sazaracs/notebook/NBHY5MMQQROJ

Supervised vs Unsupervised learning

Supervised: Data is provided with the correct labels (outputs). A machine learning model learns looking at these examples.

Two types based on target:

- **Classification:** Range of target outcome is categorical
  - Example: Classification of if customer will click an ad or not (binary)
  - Example: Product type classification: Electronics, Cleaning, Furniture (multi-class)
- **Regression:** Range of target outcome is numeric
  - Example: Forecasting product demand

Supervised vs Unsupervised learning

- **Unsupervised learning:** Correct output not available for training examples; must find patterns in data (e.g., using clustering)
  - Example: Grouping customers according to what books and movies they like
- **Reinforcement learning:** Not told what action is correct, but given some reward or penalty after each action in a sequence
  - Example: Learning how to play soccer

Topics for today

- Introduction to ML
- Evaluation Metrics for ML problems
- Tools and Libraries
- Exploratory Data Analysis
- Model Development
- Final Project

Evaluating Classification Models

Assume a binary classification problem.

	Predicted: YES ($\hat{y} = 1$)	Predicted: NO ($\hat{y} = 0$)
Actual: YES ($y = 1$)	TP	FN
Actual: NO ($y = 0$)	FP	TN

True Positive: Predicted YES when the actual is YES
False Positive: Predicted YES when the actual is NO
False Negative: Predicted NO when the actual is YES
True Negative: Predicted NO when the actual is NO

Evaluating Classification Models

Assume a binary classification problem.

	Predicted: YES ($\hat{y} = 1$)	Predicted: NO ($\hat{y} = 0$)
Actual: YES ($y = 1$)	TP	FN
Actual: NO ($y = 0$)	FP	TN

Accuracy: The percent (ratio) of cases classified correctly.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

Precision: Accuracy of a predicted positive outcome.

$$Precision = \frac{TP}{TP + FP}$$

Evaluating Classification Models

Assume a binary classification problem.

	Predicted: YES ($\hat{y} = 1$)	Predicted: NO ($\hat{y} = 0$)
Actual: YES ($y = 1$)	TP	FN
Actual: NO ($y = 0$)	FP	TN

Recall (sensitivity): Measures the strength of the model to predict a positive (1) outcome.

$$Recall = \frac{TP}{TP + FN}$$

Specificity: Measures a model's ability to predict a negative (0) outcome.

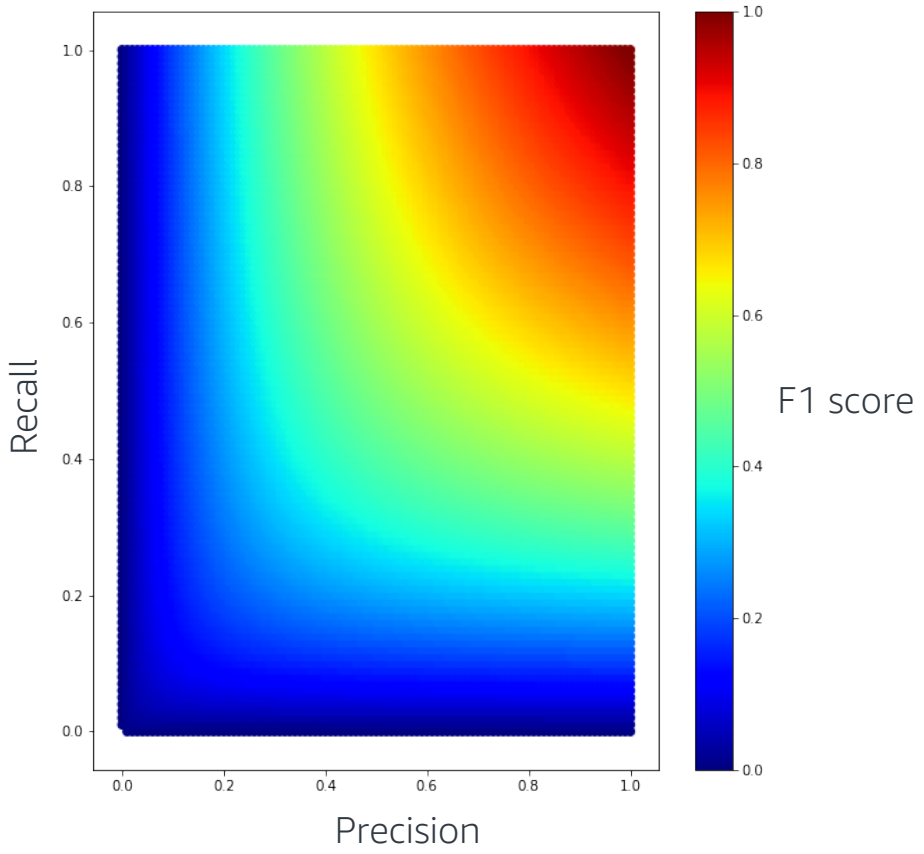
$$Specificity = \frac{TN}{TN + FP}$$

Evaluating Classification Models

F1 Score: It is a combined metric. Harmonic mean of precision and recall.

$$\text{f1\_score} = \frac{2 * (precision * recall)}{(precision + recall)}$$

$0\ (bad) \leq$ f1_score ≤ 1 (good)



Evaluating Regression Models

Metrics	Equations
Mean Squared Error (MSE)	$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
Root Mean Square Error (RMSE)	$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}$
Absolute Mean Error (AME)	$AME = \frac{1}{n}\sum_{i=1}^{n}\lvert y_i - \hat{y}_i \rvert$
R Squared (R^2)	$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$

$\boldsymbol{y_i}$: Data value
$\hat{\boldsymbol{y}}_{\boldsymbol{i}}$: Predicted values
$\bar{\boldsymbol{y}}$: Mean value
n: Number of records

Topics for today

- Introduction to ML
- Evaluation Metrics for ML problems
- Tools and Libraries
- Exploratory Data Analysis
- Model Development
- Final Project

Numpy

- Multi-dimensional arrays of items with the same type.

Examples:

```
>>> a = np.array([1, 2, 1])
>>> print(a)
[1 2 1]
>>> a.shape
(3,)
>>> a.dtype
dtype('int64')
```

```
>>> a = np.array([[1, 0.6, 0], [0, 1, 2.5]])
>>> print(a)
[[1. 0.6 0. ]
[0. 1. 2.5]]
>>> a.shape
(2, 3)
>>> a.dtype
dtype('float64')
```

Pandas

- Fast and efficient data analysis tools.
- **Data frames** are used to store and manipulate tabular data.
- For example: “people.csv”

Name	Age	Gender	Rating
Steve	32	Male	3.45
Lia	28	Female	4.6
Vin	45	Male	3.9
Katie	38	Female	2.78

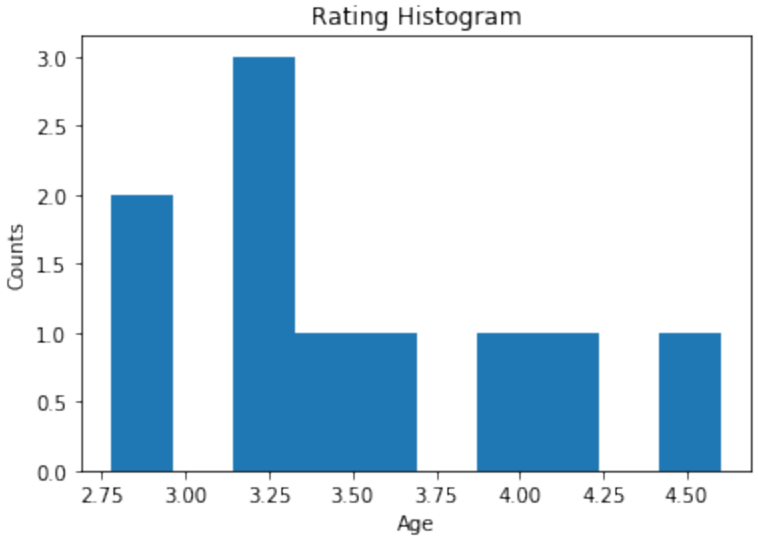
```
import pandas as pd

df = pd.read_csv("people.csv")
```

Matplotlib/seaborn

- Matplotlib and Seaborn are Python plotting libraries.
- Example: Let's use the people dataset (people.csv) with more people.

HISTOGRAM:

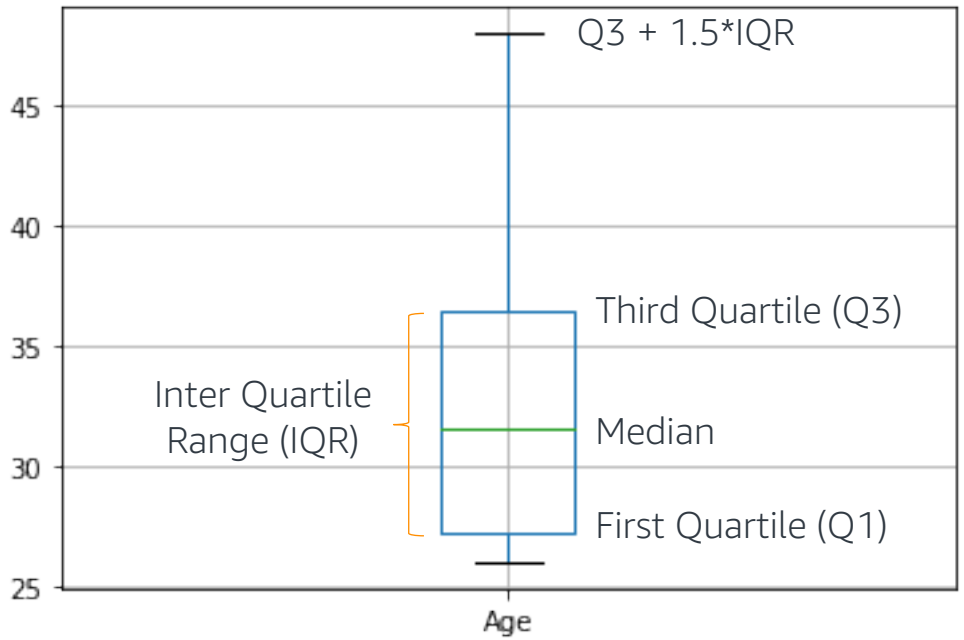
```
import matplotlib.pyplot as plt
plt.hist(df["Rating"])
plt.title('Rating Histogram')
plt.ylabel('Counts')
plt.xlabel('Age')
plt.show()
```



Matplotlib/seaborn

BOX PLOT:

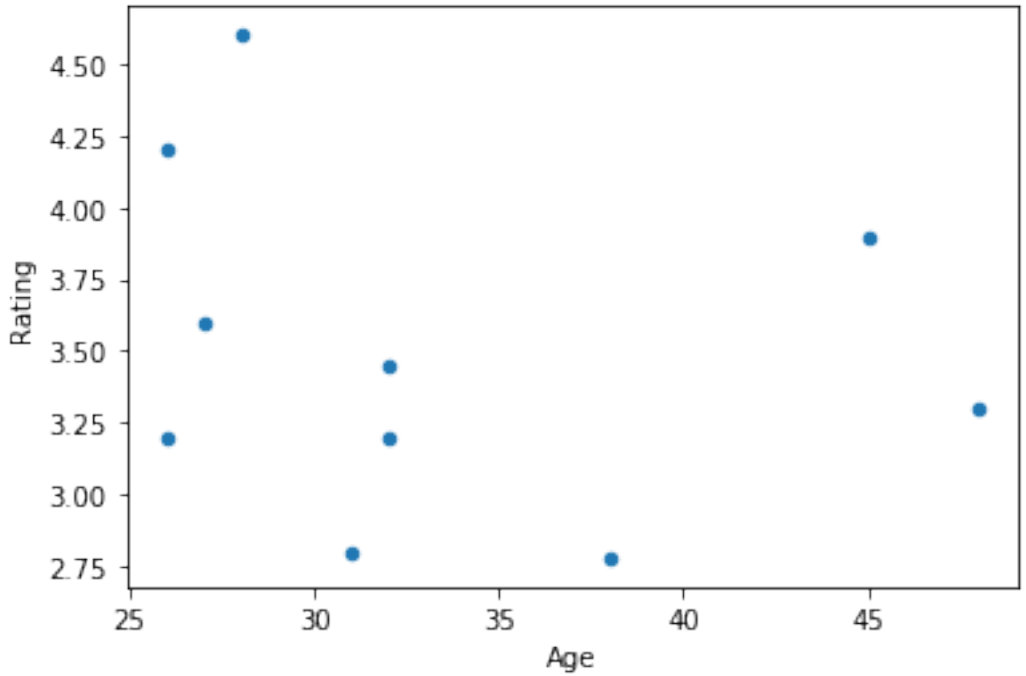
```
import matplotlib.pyplot as plt
df.boxplot(['Age'])
plt.show()
```



Matplotlib/seaborn

SCATTER PLOT:

```
import matplotlib.pyplot as plt
df.plot.scatter(x='Age', y='Rating')
plt.show()
```



Sklearn

- Open source Machine Learning library
- Provides a great selection of machine learning algorithms and data processing methods.
- https://github.com/scikit-learn/scikit-learn

Exploratory Data Analysis

- Introduction to ML
- Evaluation Metrics for ML problems
- Tools and Libraries
- Exploratory Data Analysis
- Model Development
- Final Project

Exploratory Data Analysis

- Exploratory Data Analysis (EDA) is an approach to analyze a dataset and capture main characteristics of it.
- We usually use visual methods such as plots and histograms of data points.
- Example Problem:
  - **Problem:** We will work on an Amazon dataset. Our problem is about predicting whether an ASIN has electrical plug or not (classification)
  - **Motivation:** In some marketplaces, ASINs having electrical plugs need to be accompanied by a compliance form.

Data Schema

Column	Description	Feature type	Data Type
ASIN	Product ASIN	Text	string
target_label	Binary field with values in {0,1}. A value of 1 show ASIN has a plug, otherwise 0	Numerical	integer
ASIN_STATIC_ITEM_NAME	Title of the ASIN	Text	string
ASIN_STATIC_PRODUCT_DESCRIPTION	Description of the ASIN	Text	string
ASIN_STATIC_ITEM_CLASSIFICATION	Item classification of whether it is a standalone or bundle parent item etc.	Categorical	string
ASIN_STATIC_GL_PRODUCT_GROUP_TYPE	GL product group information for the ASIN	Categorical	string
ASIN_STATIC_ITEM_PACKAGE_WEIGHT	Weight of the ASIN	Numerical	float
ASIN_STATIC_LIST_PRICE	Price information for the ASIN	Numerical	float
ASIN_STATIC_BATTERIES_INCLUDED	Information whether batteries are included along with the product	Binary	bool
ASIN_STATIC_BATTERIES_REQUIRED	Information whether batteries are required for using the product	Binary	bool

Descriptive Statistics

- Overall statistics
  - Number of instances (i.e. number of rows)
  - Number of attributes (i.e. number of columns)
- Attribute statistics (univariate or single variable)
  - Statistics for numeric attributes (mean, variance, etc.) -- df.describe()
  - Statistics for categorical attributes (histograms, mode, most/least frequent values, percentage, number of unique values)
    - Histogram of values: E.g., df[<attribute>].value_counts() or seaborn's distplot()
  - Target statistics
    - Class distribution: E.g., df[<target>].value_counts() or np.bincount(y)
- Multivariate statistics (more than one variable)
  - Correlation, Contingency Tables

EDA – Hands-on

- Let's read this Amazon dataset and analyze it
- This will also be a good practice for your final project

- Follow this notebook:

  https://eider.corp.amazon.com/sazaracs/notebook/NB7BVDSSGXU2

Correlations

- **Correlations:** How strongly pairs of attributes are related.
- **Scatterplot matrices** visualize attribute-target and attribute-attribute pairwise relationships
- **Correlation matrices** measure the linear dependence between features; can be visualized with heat-maps

- Mean: $\boldsymbol{\mu_x = \frac{1}{n}\sum_{i=1}^{n} x^{(i)}}$
- Variance: $\boldsymbol{\sigma_x^2 = \frac{1}{n}\sum_{i=1}^{n}\left(x^{(i)} - \mu_x\right)^2}$

- Covariance between (x, y):

$$\sigma_{xy} = \frac{1}{n}\sum_{i}^{n}\left(x^{(i)} - \mu_x\right)\left(y^{(i)} - \mu_y\right)$$

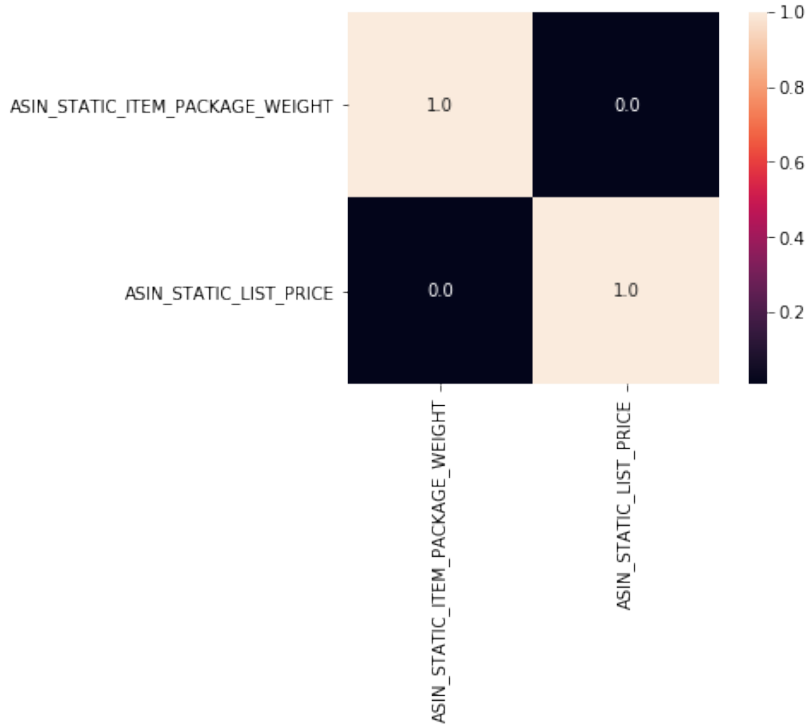
- Correlation between (x, y):

$$r = \frac{\sigma_{xy}}{\sigma_x \sigma_y}$$

which is always between -1 and 1

Correlation Matrix Heatmap

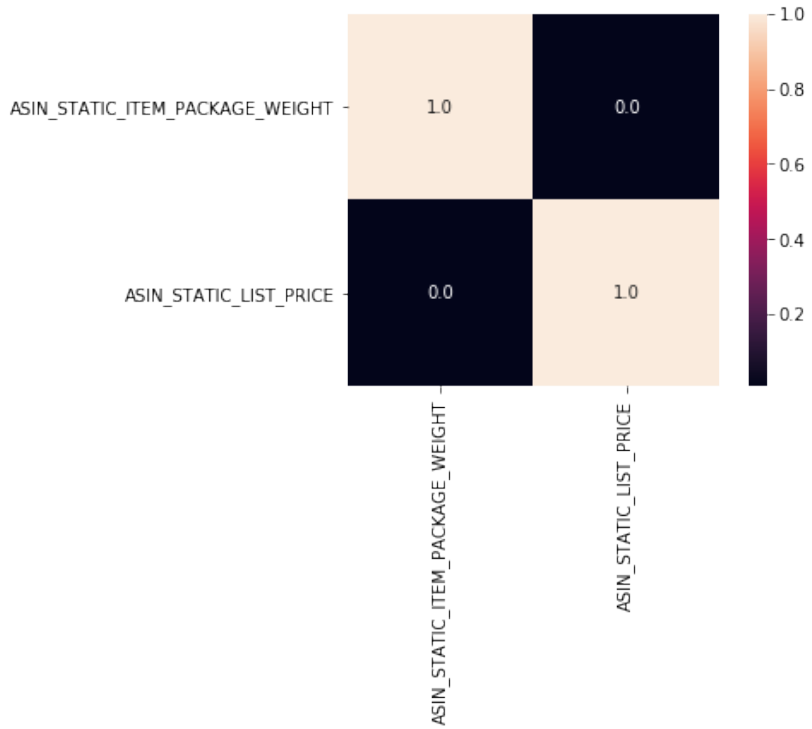
```
cols = ['ASIN_STATIC_ITEM_PACKAGE_WEIGHT', 'ASIN_STATIC_LIST_PRICE']
cm = np.corrcoef(df[cols].values.T)
ax = sns.heatmap(cm, cbar=True, annot=True, square=True, fmt='.1f', yticklabels=cols, xticklabels=cols)
plt.show()
```



- Correlation matrix is usually easier to read than scatter plot matrixes.
- Correlation values are between -1 and 1.
- +1 means perfect positive correlation and -1 shows perfect negative correlation
- 0 means there is no relationship between the variables

Correlation Matrix Heatmap

```
cols = ['ASIN_STATIC_ITEM_PACKAGE_WEIGHT', 'ASIN_STATIC_LIST_PRICE']
cm = np.corrcoef(df[cols].values.T)
ax = sns.heatmap(cm, cbar=True, annot=True, square=True, fmt='.1f', yticklabels=cols, xticklabels=cols)
plt.show()
```



Multi-collinearity

- Highly correlated (pos. or negative) attributes usually degrade performance of linear ML models such as linear and logistic regression models.
- With the regression models, we should select one of the correlated pairs and discard the other.
- Decision Trees are immune to this problem.

Topics for today

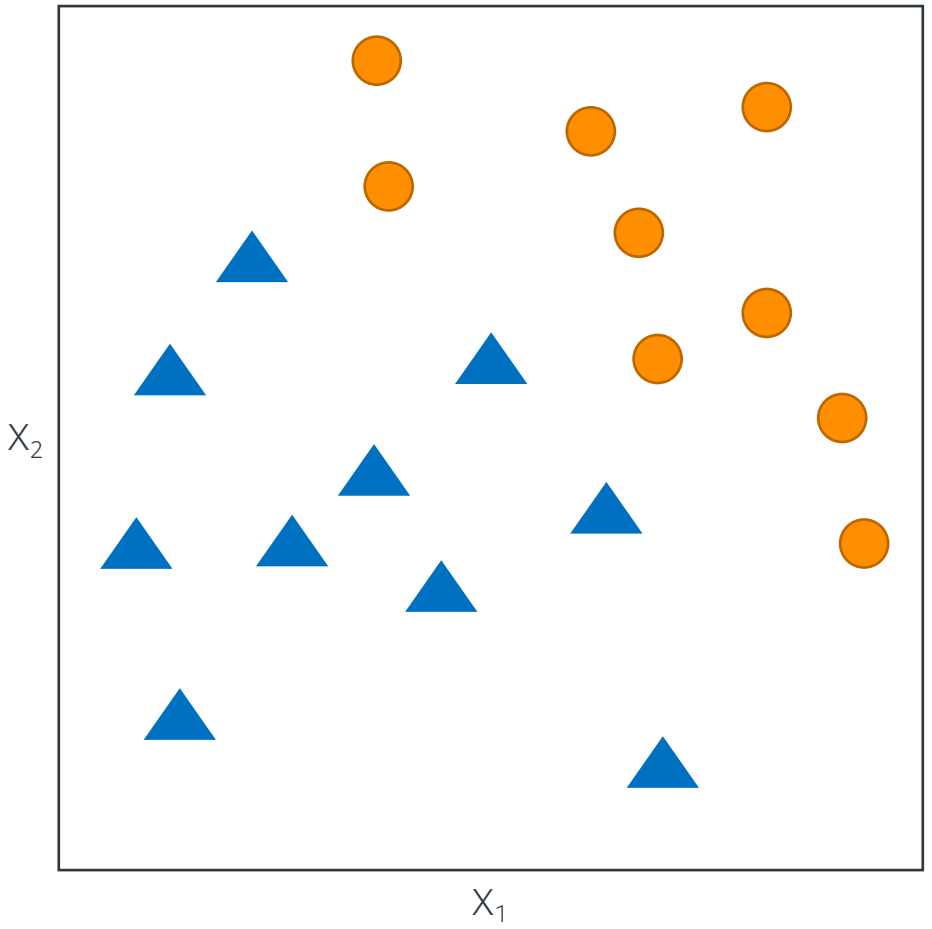
- Introduction to ML
- Evaluation Metrics for ML problems
- Tools and Libraries
- Exploratory Data Analysis
- Model Development
- Final Project

K Nearest Neighbors (KNN)

- K Nearest Neighbors (KNN) model classifies new data points based on the similar other records in a dataset.
- Algorithm:
  - Find “K” similar records
  - For classification (class prediction): Take the majority class of those K records
  - For regression (numerical value prediction): Take the average value of those K records
- Assumptions:
  - Similarity can be measured by the distance between features of data points.
  - Points that are close to each other in space are also considered similar to each other.

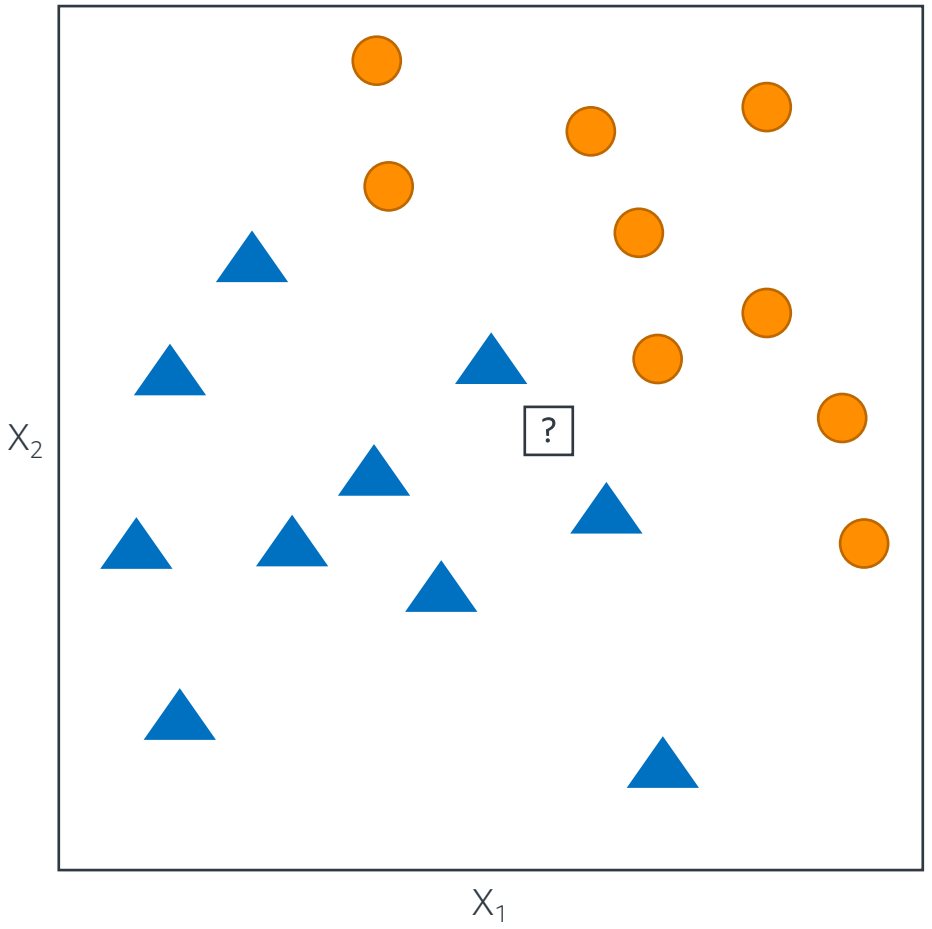
K Nearest Neighbors (KNN)

- Assume a dataset:
  - Two classes: ● and ▲
  - Two features X_1 and X_2
  - K=3



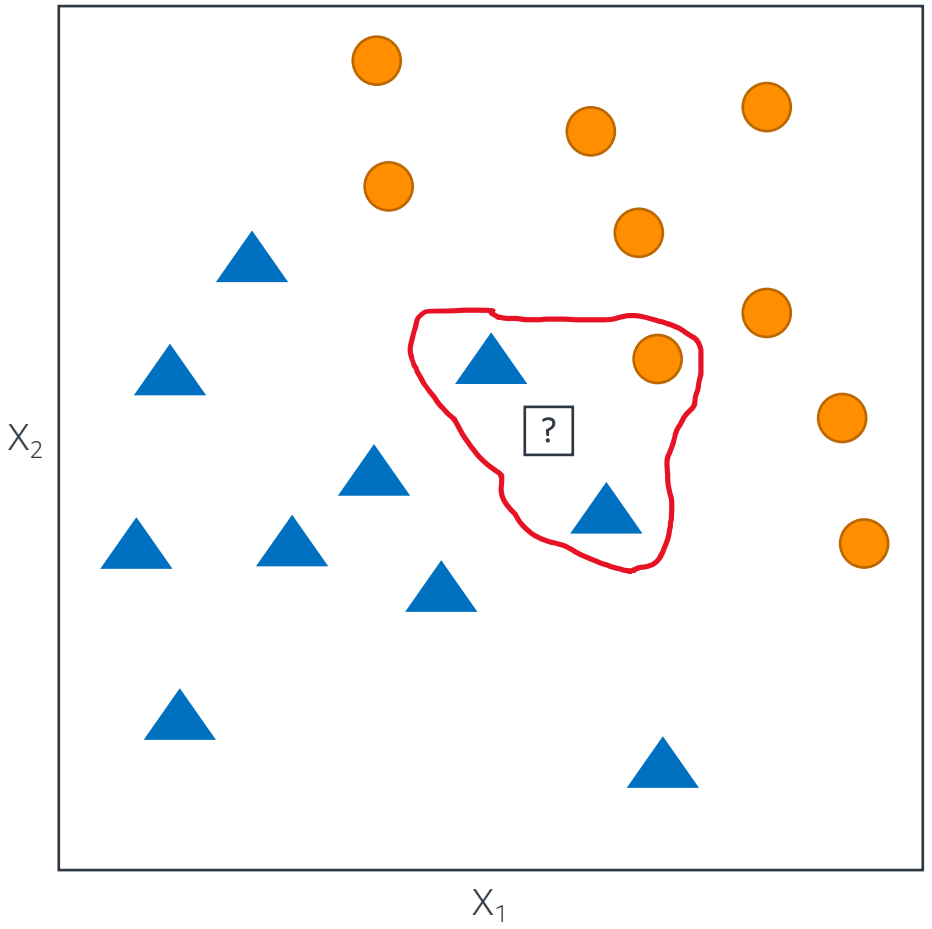
K Nearest Neighbors (KNN)

- Assume a dataset:
  - Two classes: ● and ▲
  - Two features X_1 and X_2
  - K=3
- What class does ? belong?



K Nearest Neighbors (KNN)

- Assume a dataset:
  - Two classes: ● and ▲
  - Two features X_1 and X_2
  - K=3
- What class does [?] belong?
  - Look at the closest K points
  - Pick the majority class: ▲



K Nearest Neighbors (KNN)

- **Scaling:** We should scale our features to values between 0-1. Otherwise, the model can rely on the features with large spreads.
- **K value:** We need to try and select an appropriate K number. We can use a validation set or apply cross-validation for this.
- **Curse of dimensionality:** It suffers from high dimensional data (too many features). Spaces between points can get very large and our **closer points ≈ similar records** assumption may not hold very well.
- We will apply this method to our final project today.

Putting it all together

- In this exercise we will work with our internal Amazon dataset.
- We will do the following tasks:
  - Exploratory Data Analysis
  - Splitting dataset into training and test sets.
  - Fit a K Nearest Neighbors Classifier.
  - Check the performance metrics on test set.
- Follow this notebook:

  https://eider.corp.amazon.com/sazaracs/notebook/NB78VKF3786W

Final Project

Product substitute: Given products (A, B), predict whether B is a substitute for A

- We say that B is a "substitute" for A if a customer would buy B in place of A -- say, if A were out of stock.
- **The goal** of this project is to predict a substitute relationship between pairs of products.
- **Link:** https://leaderboard.corp.amazon.com/tasks/478

Final Project

Business rationale:

Predicting substitutes is of critical importance to Amazon, for several reasons:

- When products go out of stock, we want to provide a suitable replacement;
- When products are overpriced (due to greedy 3P sellers), we want to provide a reasonable alternative;

Substitutable products are "similar," so a substitute model can enable other ML models that require product-to-product similarity.

Final Project – Dataset Files

1. **asin_product.csv:** Data file with 113 columns for Amazon products.

Region Id	MarketPlace	ASIN	Binding Code	binding_desc	brand_code	case_pack_q	classificatior	classificatior	color_map	country_of_c	cpsia_cautio	creation_dat	currency_coc	custom
1	1	153427507	hardcover	Hardcover			base_produc	Base Product				15-Dec-08	USD	
1	1	267648340	hardcover	Hardcover	FOS3T		base_produc	Base Product				16-Sep-16	USD	
1	1	545496470	hardcover	Hardcover	KLUTZ	6	base_produc	Base Product	Black		choking_haz	19-Aug-12	USD	
1	1	679858040	paperback	Paperback			base_produc	Base Product				29-Nov-96	USD	
1	1	078694742X	toy	Toy	WZDCS	12	base_produc	Base Product			choking_haz	3-Mar-08	USD	
1	1	1059998254	consumer_el	Electronics			base_produc	Base Product	black			12-May-14	USD	

This file contains important product information that we need to build our ML models.

Final Project – Dataset Files

2. **dataset_metadata.csv:** Details of the columns in the asin_product.csv file.

Column Name	Data Type	Description
REGION_ID	NUMBER(2,0)	DW specific locale identifier. Referenced in domain
MARKETPLACE_ID	NUMBER	Unique identifier for a marketplace. A replacement
ASIN	CHAR(10)	Amazon Standard Item Number sometimes also kn
BINDING	VARCHAR2(96)	This former books term is used across all product li
BINDING_DESCRIPTION	VARCHAR2(100)	Text description of the above binding column. descr
BRAND_CODE	VARCHAR2(5)	Publisher code for an ASIN. Source column for Bran
CASE_PACK_QUANTITY	NUMBER(38,14)	The number of items packed together and sold. For
CLASSIFICATION_CODE	VARCHAR2(18)	The code value for the classification the ASIN falls u
CLASSIFICATION_DESCR	VARCHAR2(100)	Description of the classification the ASIN falls unde
COLOR_MAP	VARCHAR2(100)	The base color with which an item is associated. Th
COUNTRY_OF_ORIGIN	VARCHAR2(2)	The country in which the product was published or r
CPSIA_CAUTIONARY_ST	VARCHAR2(100)	Refer: https://w.amazon.com/index.php/CPSIA/Det
CREATION_DATE	DATE	Source audit column with the date and time the rec
CURRENCY_CODE	VARCHAR2(15)	Identifies the currency used in the metrics. Example

Final Project – Dataset Files

3. **training.csv**: Our training data with product pairs given by their ASINs. “1” substitute, “0” not substitute

ID	key_asin	cand_asin	label
42595	B01L7CFUWC	B01CU4SOQ0	1
35775	B01KDAKKTM	B01CGQE5YC	0
37152	B013FA0UVA	B06W9HY6MV	1
4340	B008KPZLEC	B01M5AMNA1	0
37349	B0196BJHXY	B00XCHMLI2	1
38826	B000GZEK00	B01H0SJU9Q	0
33081	B01MD0VMF2	B00WW5FAVA	1
14964	B000I1R0JU	B003LPZT02	1
40912	B001LQWBX6	B001DNZ0H6	1
15339	B01IRGLTUS	B000HM9R14	0

Final Project – Dataset Files

4. public_test_features.csv: Our test data. Similar to the training.csv except we will use this data to test our model and predict the labels for each row.

ID	key_asin	cand_asin	label
39236	B01C5TFLSE	B06XDMZ5MY	
1353	B003YJ8TVQ	B01G6R24CM	
39280	B0063X7BT6	B01BO2NOD2	
1665	B01DJH637O	B017SCJACQ	
14925	B003U8ESI4	B00HGDMGM4	
31626	B015P0UH66	B00JSK2ZNS	
45515	B00L8QBMS4	B001ROTRLG	
41672	B00YCZ6IKA	B01HF24PH4	
29905	B00NQ17ZUS	B000E1FYQA	
18576	B01LPR16GI	B00TGKAPYQ	
18182	B00D7ANRVY	B00JXED39O	

Final Project Walkthrough – Day 1

- See the provided project walkthrough below to get started with this project.
- Follow the given hints and directions there.
- Complete the following notebook and submit your results

https://eider.corp.amazon.com/sazaracs/notebook/NBJ7LORBIF3G

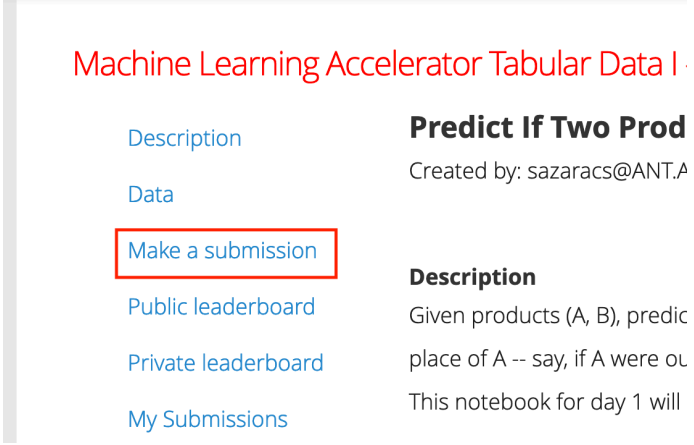
How do I submit to Leaderboard?

How do I submit to Leaderboard?

- Your submission should be a csv file with the first row: "**ID, label**". Your IDs will be from the **public_test_features.csv** file.

ID	label
39236	0
1353	0
39280	1
1665	1
14925	0

- You submit your model's output (a .csv file) to Leaderboard via the 'Make a Submission' tab.



Public and Private Leaderboards

Your submission **automatically** goes to two leaderboards: **Public** and **Private**.

- The **public Leaderboard** is a subset of the test dataset that gives you live feedback on the performance of your model on unseen data. We keep the highest score of all your submissions.
- The **private Leaderboard** is another subset of your test dataset. These results are **hidden until the end of the competition**. This leaderboard will be used to determine the winners of the competition.

Notebooks for day 1

- Our first ML model on a food delivery problem: https://eider.corp.amazon.com/sazaracs/notebook/NB8JDU4TGHB1
- Overfitting with the K Nearest Neighbors model: https://eider.corp.amazon.com/sazaracs/notebook/NBHY5MMQQROJ
- Exploratory Data Analysis of an Amazon dataset: Electrical plug prediction https://eider.corp.amazon.com/sazaracs/notebook/NB7BVDSSGXU2
- Full cycle model development on the Electrical plug prediction: https://eider.corp.amazon.com/sazaracs/notebook/NB78VKF3786W
- Final project walkthrough for day 1: https://eider.corp.amazon.com/sazaracs/notebook/NBJ7LORBIF3G

Tomorrow

Tomorrow we will work on these topics:

- **Data Imputation:** Fixing missing values
- **Feature Scaling:** Dealing with ranges of numerical variables
- **Data Labeling:** Demo with the **AWS Ground Truth** tool.
- **Feature Engineering:** Handling text and categorical data
- **Trees, Ensemble Learning and Boosting:** More advanced ML models
- **Hyperparameter tuning:** Tune your ML model for best performance