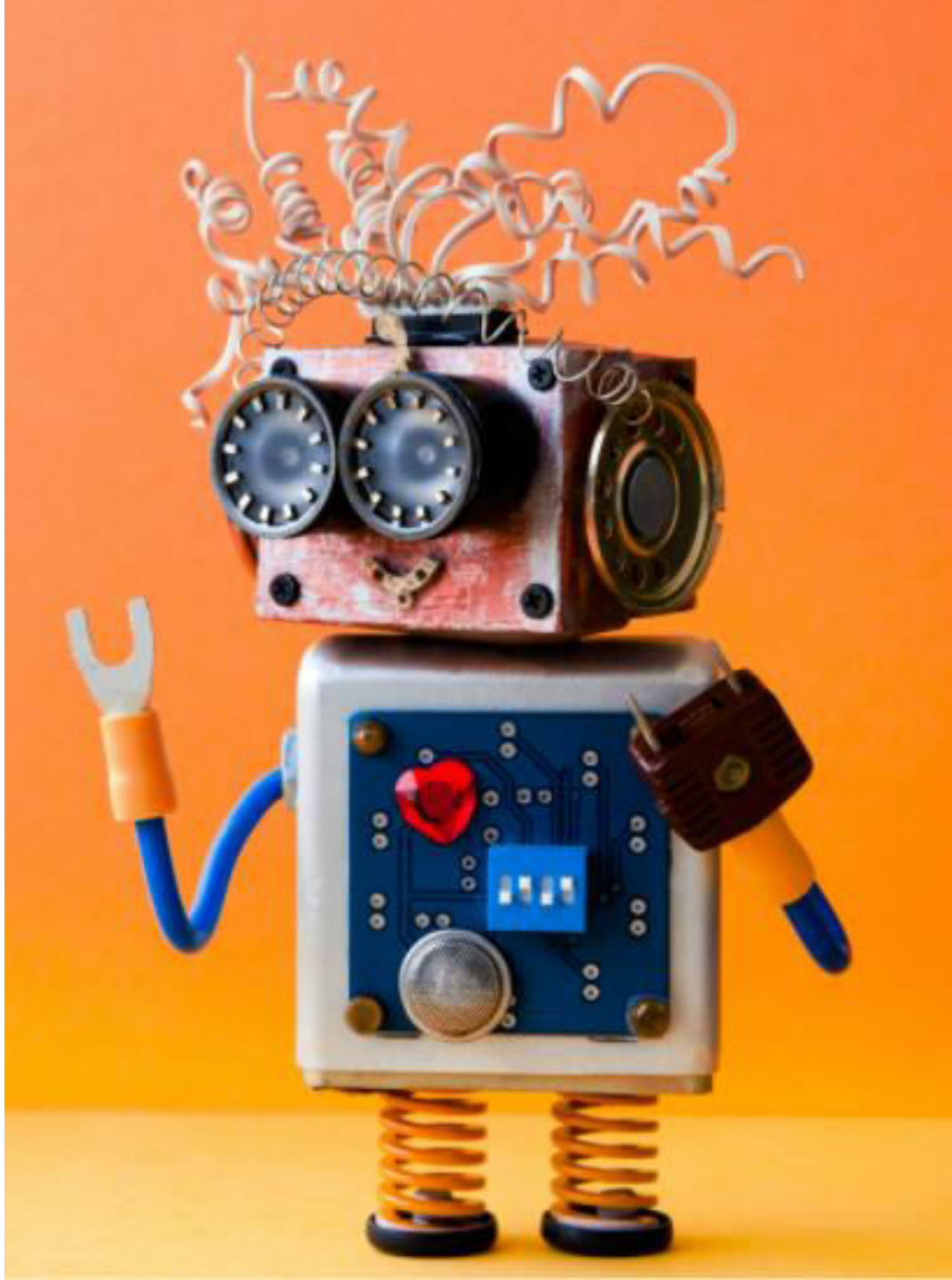
PROJECT OVERVIEW

EASY

ADVANCED



PROJECT OVERVIEW

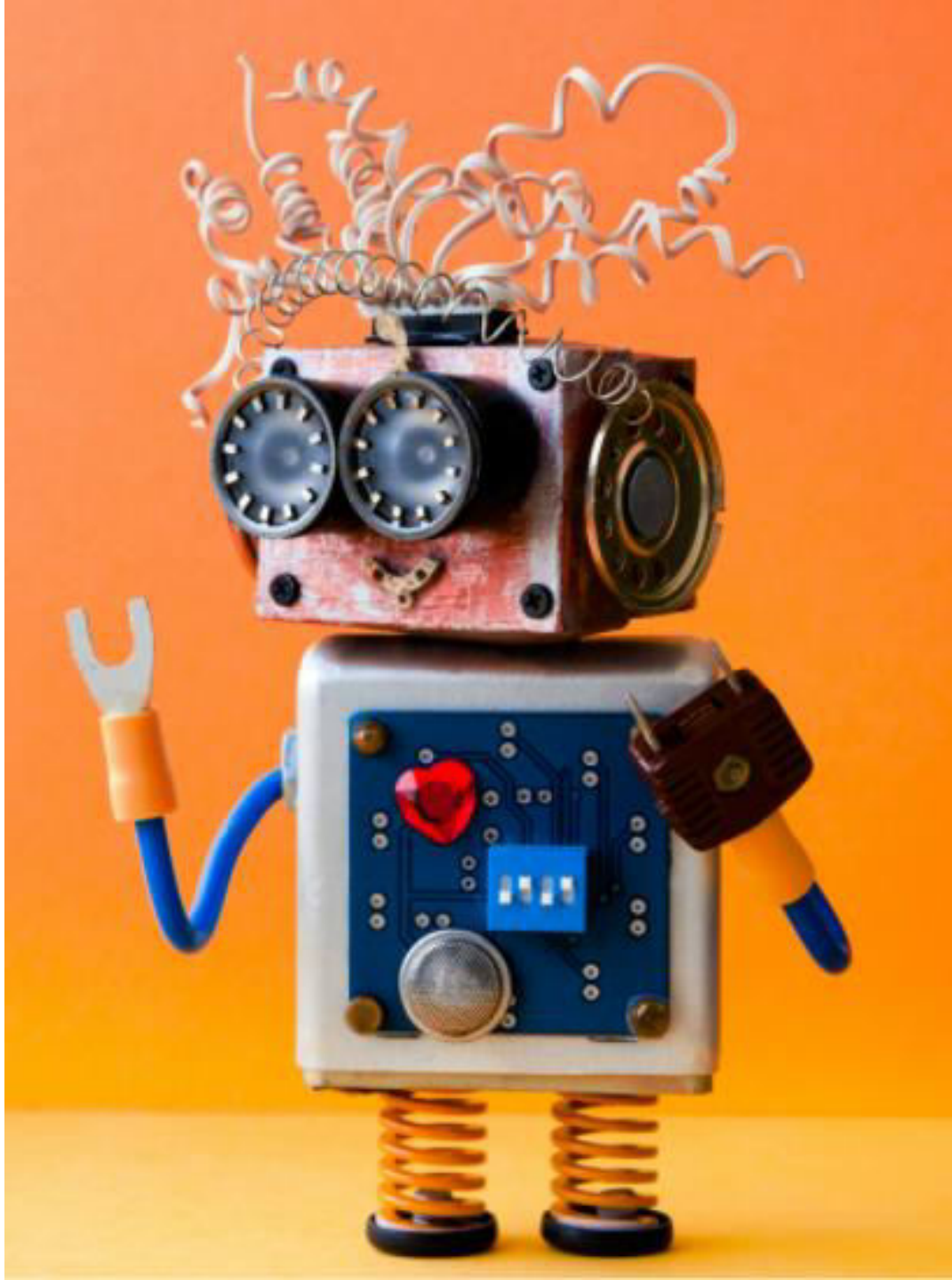
- Kyphosis is an abnormally excessive convex curvature of the spine. The kyphosis data frame has 81 rows and 4 columns representing data on children who have had corrective spinal surgery. Dataset contains 3 inputs and 1 output
- **INPUTS:**
  - Age: in months
  - Number: the number of vertebrae involved
  - Start: the number of the first (topmost) vertebra operated on.

- **OUTPUTS:**
  - Kyphosis: a factor with levels “absent” or “present” indicating if a kyphosis (a type of deformation) was present after the operation.

- Link to dataset: https://www.kaggle.com/abbasit/kyphosis-dataset
- Source: John M. Chambers and Trevor J. Hastie eds. (1992) Statistical Models in S, Wadsworth and Brooks/Cole, Pacific Grove, CA.
- Photo Credit: https://commons.wikimedia.org/wiki/File:Kyphosis.png

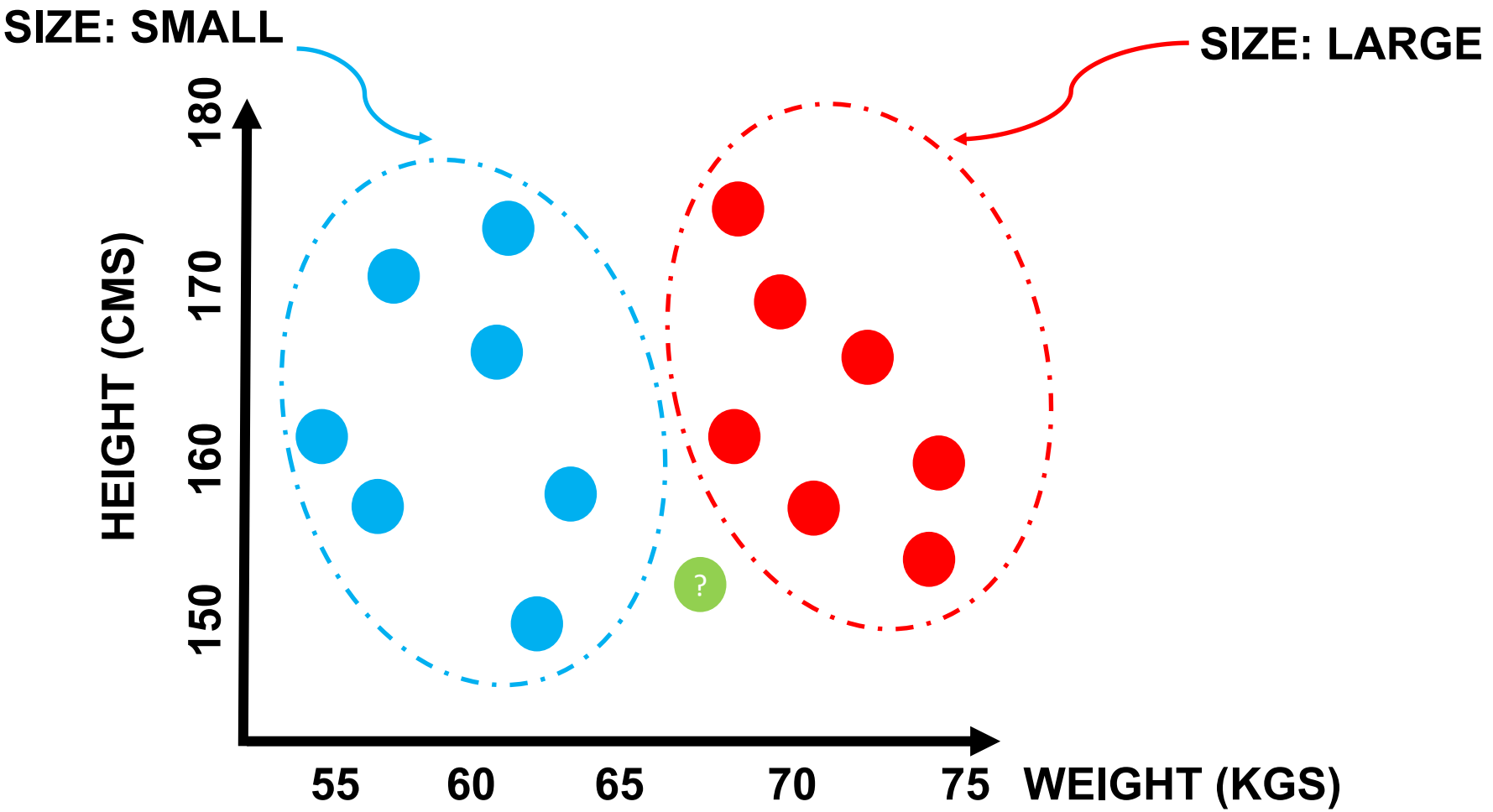
K NEAREST NEIGHBORS (KNN) ALGORITHM

EASY ADVANCED

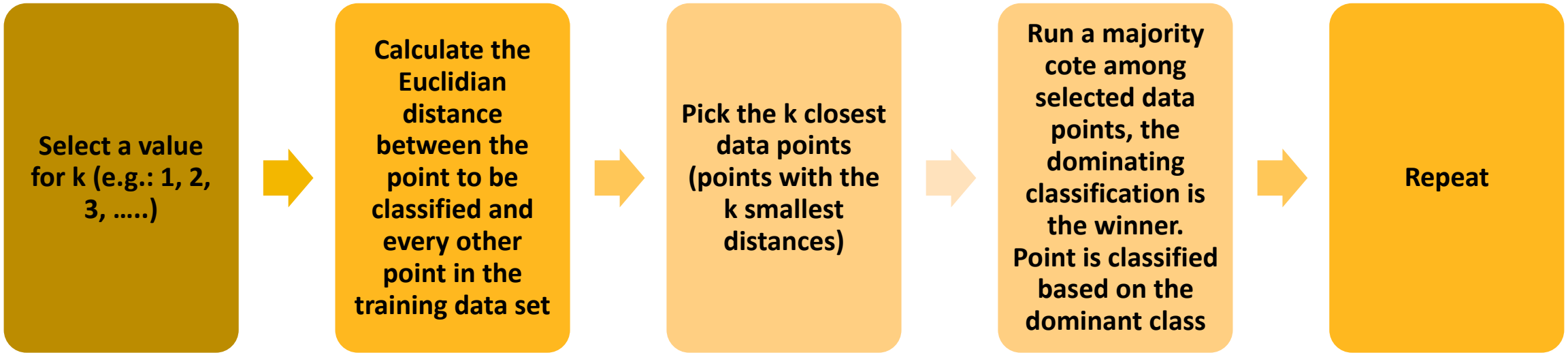


K NEAREST NEIGHBORS (KNN): INTUITION

- K-Nearest Neighbors (KNN) algorithm is a classification algorithm
- KNN works by finding the most similar data points in the training data, and attempt to make an educated guess based on their classifications

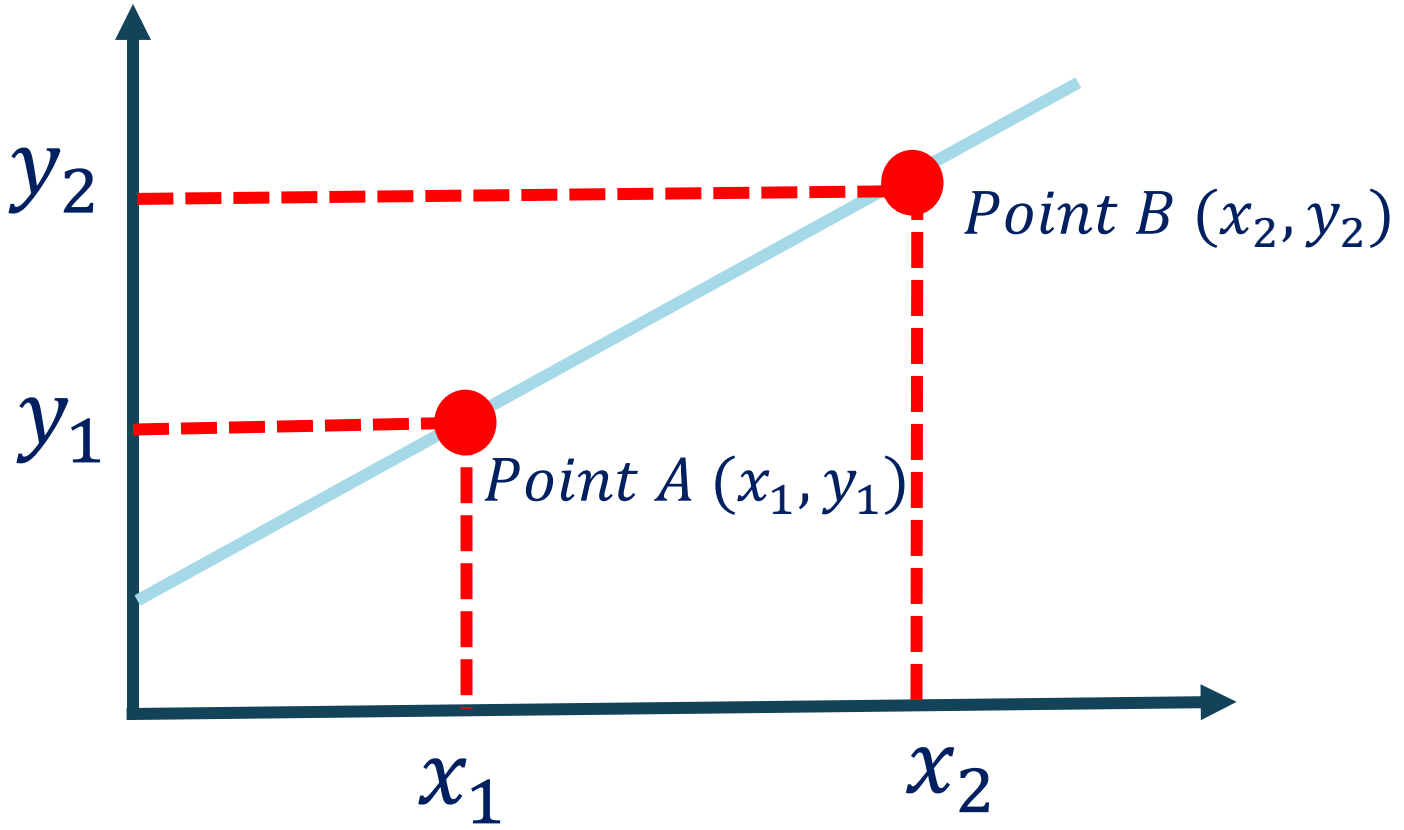


K NEARSET NEIGHBORS (KNN): ALGORITHM STEP



EUCLIDEAN DISTANCE: INTUITION

$$\boldsymbol{Euclidean\ Distance} = \sqrt{(\boldsymbol{x_2} - \boldsymbol{x_1})^2 + (\boldsymbol{y_2} - \boldsymbol{y_1})^2}$$



K NEARSET NEIGHBORS (KNN): EXAMPLE

- KNN will look for the 5 data points that are closest to the new customer data point
- The algorithm will determine which category (class) are these 5 points in
- Since 4 points had class “SMALL” and 1 had “LARGE”, then new customer shall be assigned small size

Height	Weight	T-Shirt Size	Euclidian Dist	Vote
158	58	S	4.242640687	
158	59	S	3.605551275	
158	63	S	3.605551275	
160	59	S	2.236067977	3
160	60	S	1.414213562	1
163	60	S	2.236067977	3
163	61	S	2	2
160	64	L	3.16227766	5
163	64	L	4	
165	61	L	4.123105626	
165	62	L	5.656858249	

New Customer Information:

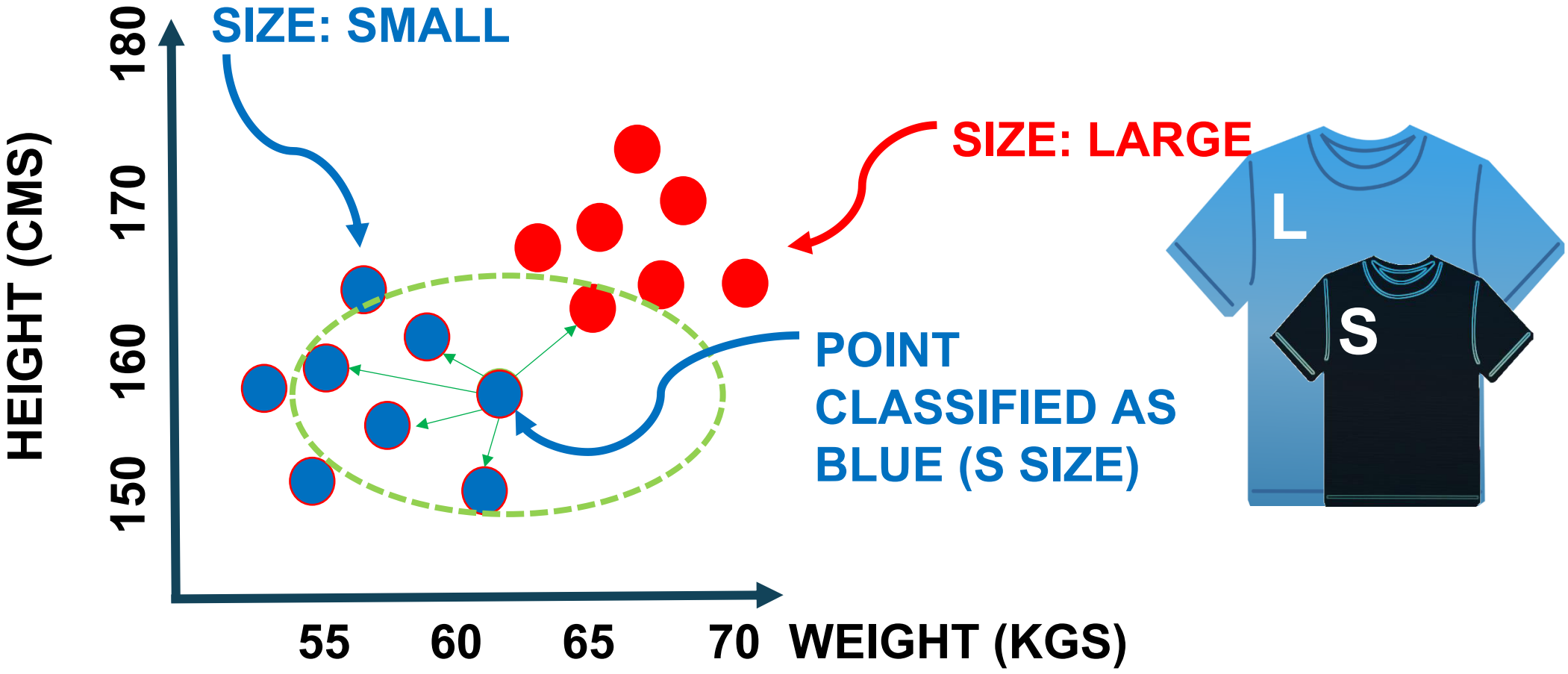
Height: 161
Weight: 61

Assume, k= 5

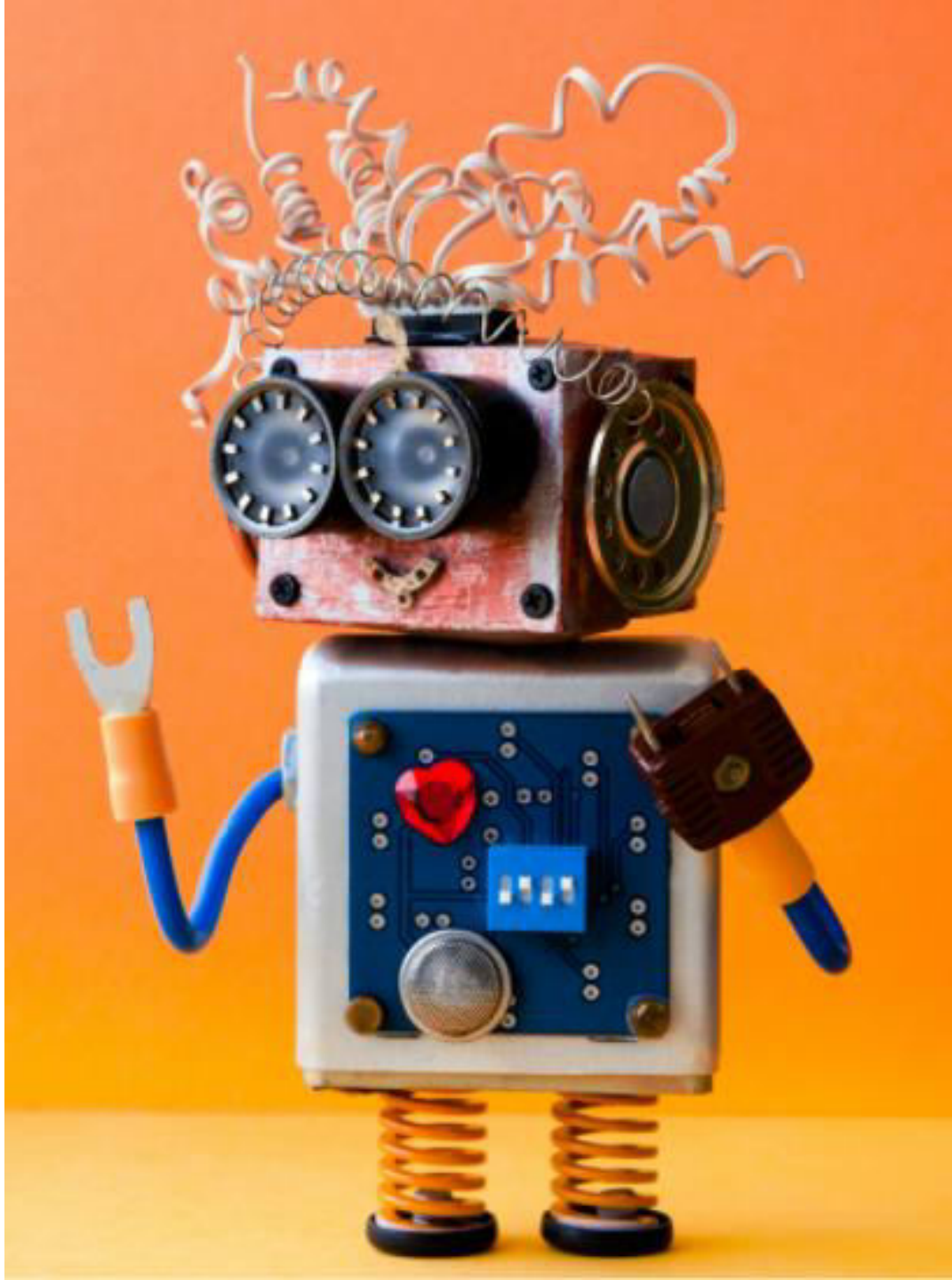
Example Source: https://www.listendata.com/2017/12/k-nearest-neighbor-step-by-step-tutorial.html

K NEAREST NEIGHBORS (KNN): EXAMPLE

- Let's understand this example visually!



K NEAREST NEIGHBORS (KNN) ALGORITHM IN SAGEMAKER



K NEAREST NEIGHBORS (KNN) IN SAGEMAKER

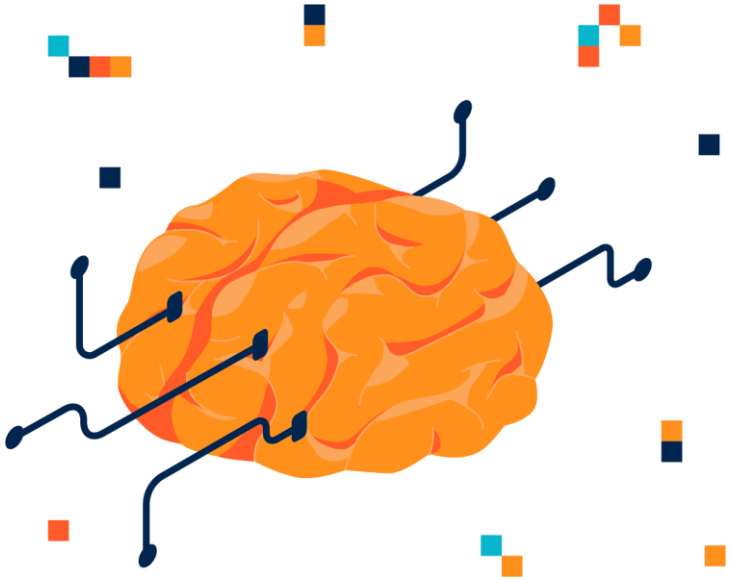
- KNN in SageMaker could be used to perform simple classification or regression
  - **Classification:** algorithm finds the K-closest points to a given sample point and return the most frequent label
  - **Regression:** algorithm finds K-closest points to a given sample point and return the average value.
- KNN is a lazy algorithm, it does not try to generalize the model for the entire training dataset, but it rely on neighbouring data points.
- Training with the KNN algorithm has three steps:
  - Sampling
  - Dimension reduction
  - Index building
- Sampling is used to minimize the size of dataset to optimize memory.
- Dimensionality reduction is performed to:
  - Decrease the feature dimension of the data to reduce the footprint of the k-NN model in memory and inference latency and avoids the “curse of dimensionality”

K NEAREST NEIGHBORS (KNN): HYPERPARAMETERS

- Full set of hyperparameters: https://docs.aws.amazon.com/sagemaker/latest/dg/kNN_hyperparameters.html
- K: The number of nearest neighbors
- Sample_size: The number of data points to be sampled from the training data set.
- feature_dim: The number of features in the input data.
- predictor_type: classification or regression
- dimension_reduction_target: The target dimension to reduce to.

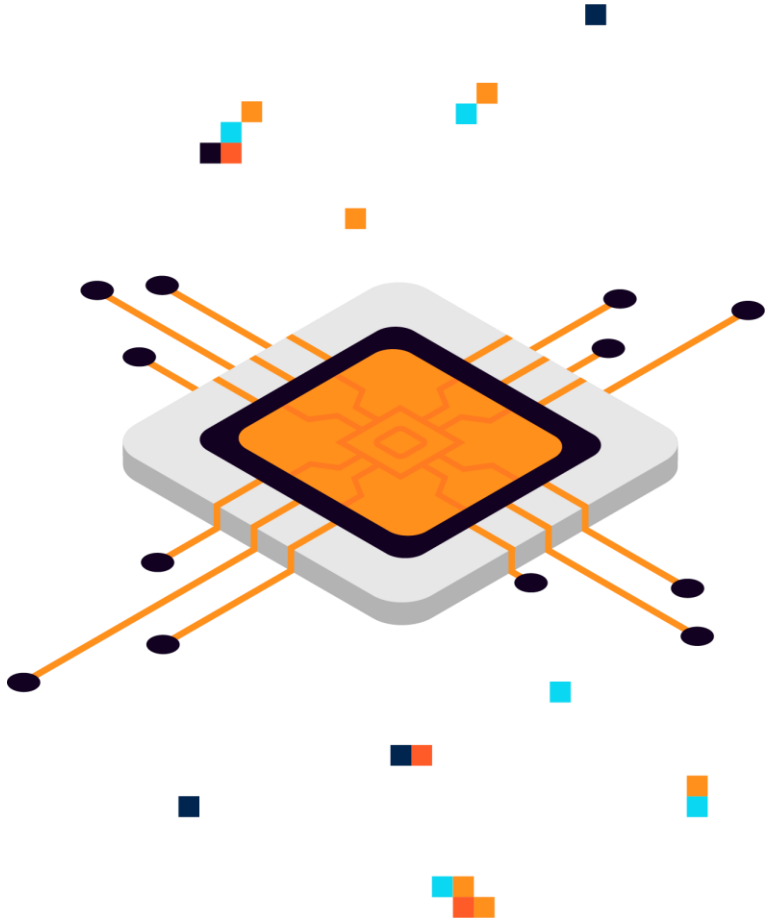
K NEAREST NEIGHBORS (KNN): INPUT/OUTPUT

- KNN supports two channels:
  - Train channel contains training data
  - Test channel to provide test scores such as accuracy for classifier or MSE for regressor
- SageMaker KNN algorithm supports recordIO-protobuf or CSV formats
- KNN can be used in both File or pipe mode

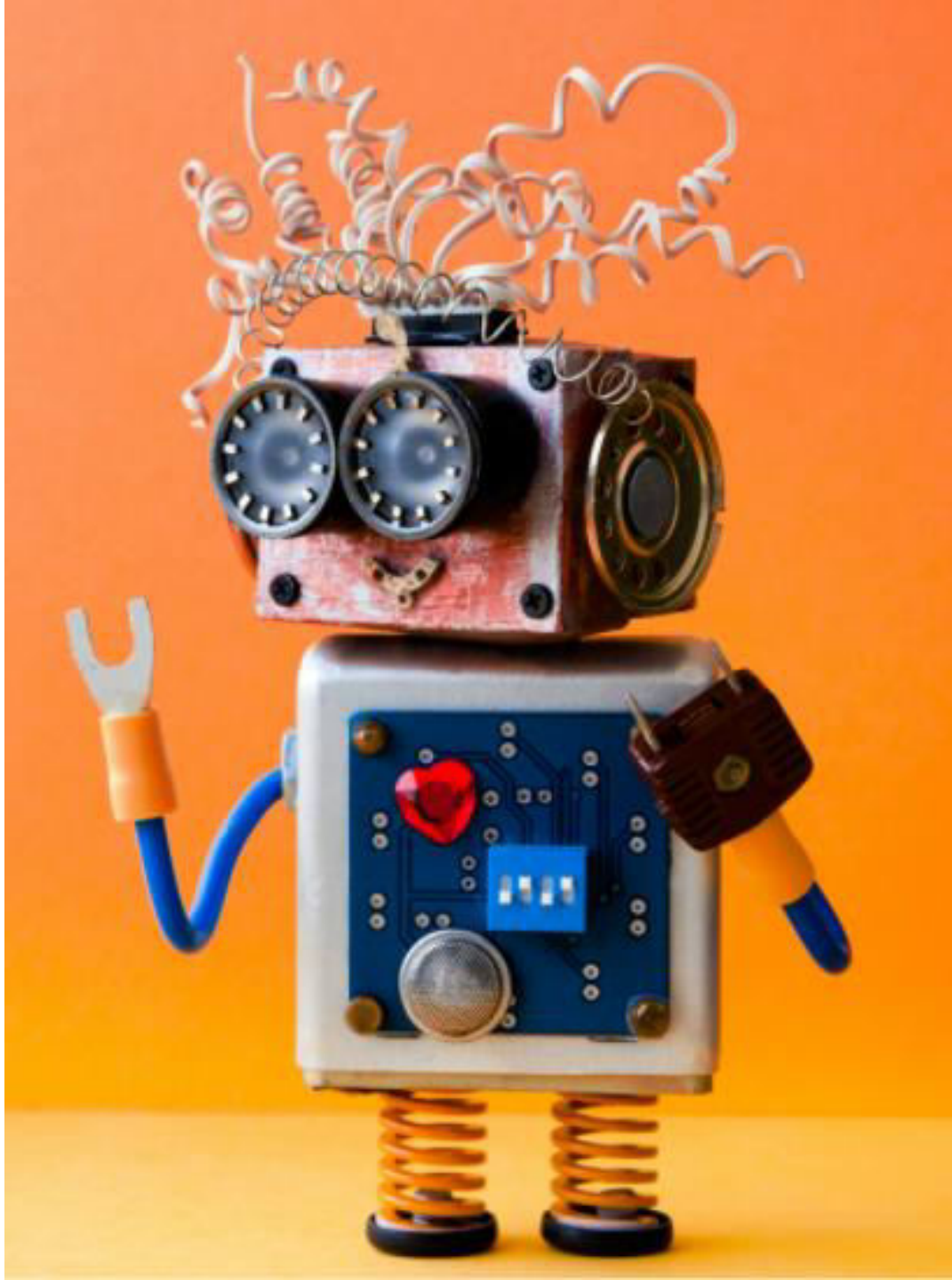


K NEAREST NEIGHBORS (KNN): INSTANCE TYPES

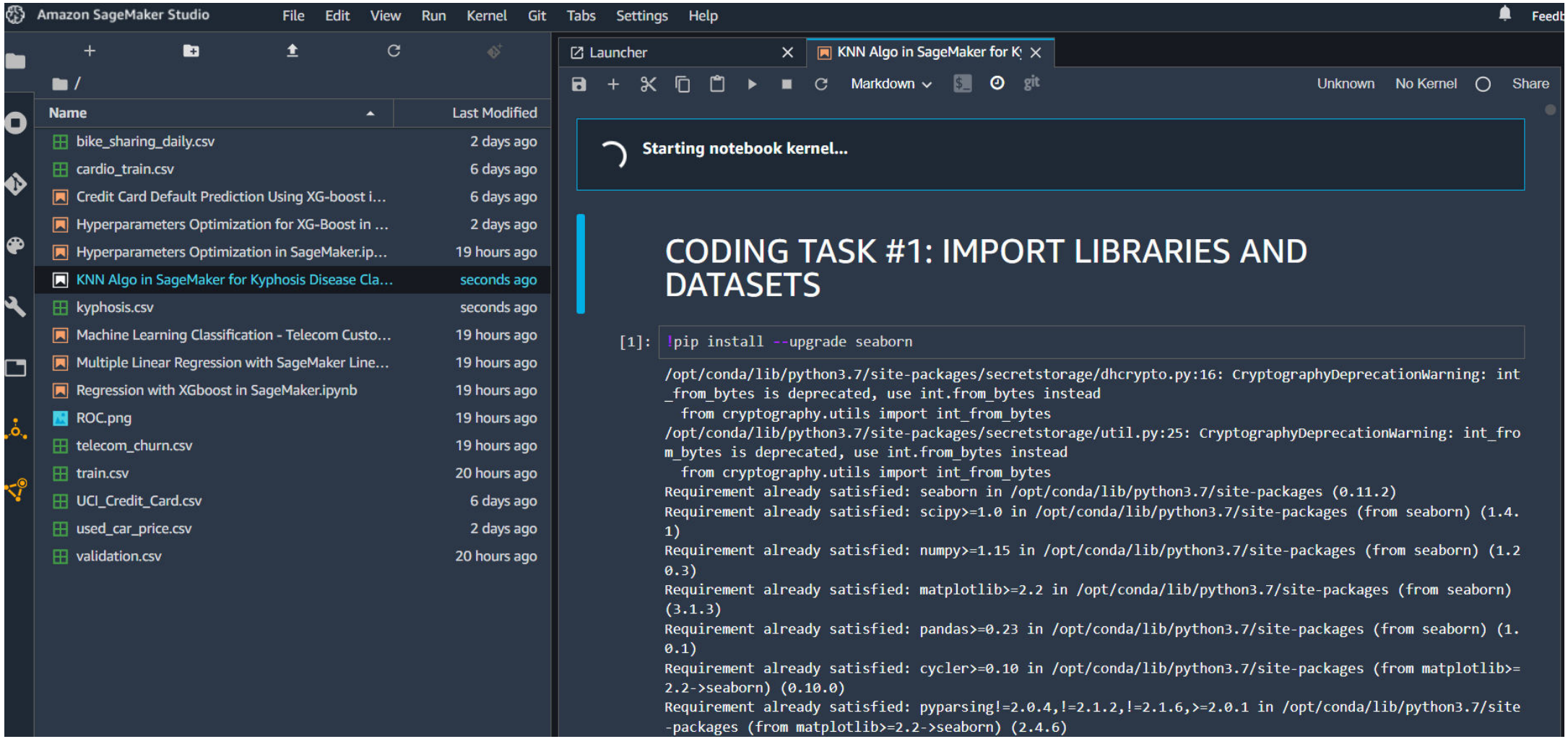
- For KNN Training:
  - CPU such as Ml.m5.2xlarge
  - GPU such as Ml.p2.xlarge
- For Inference:
  - GPU for higher throughput on large batches
  - CPU generally provides lower latency



CODE DEMO

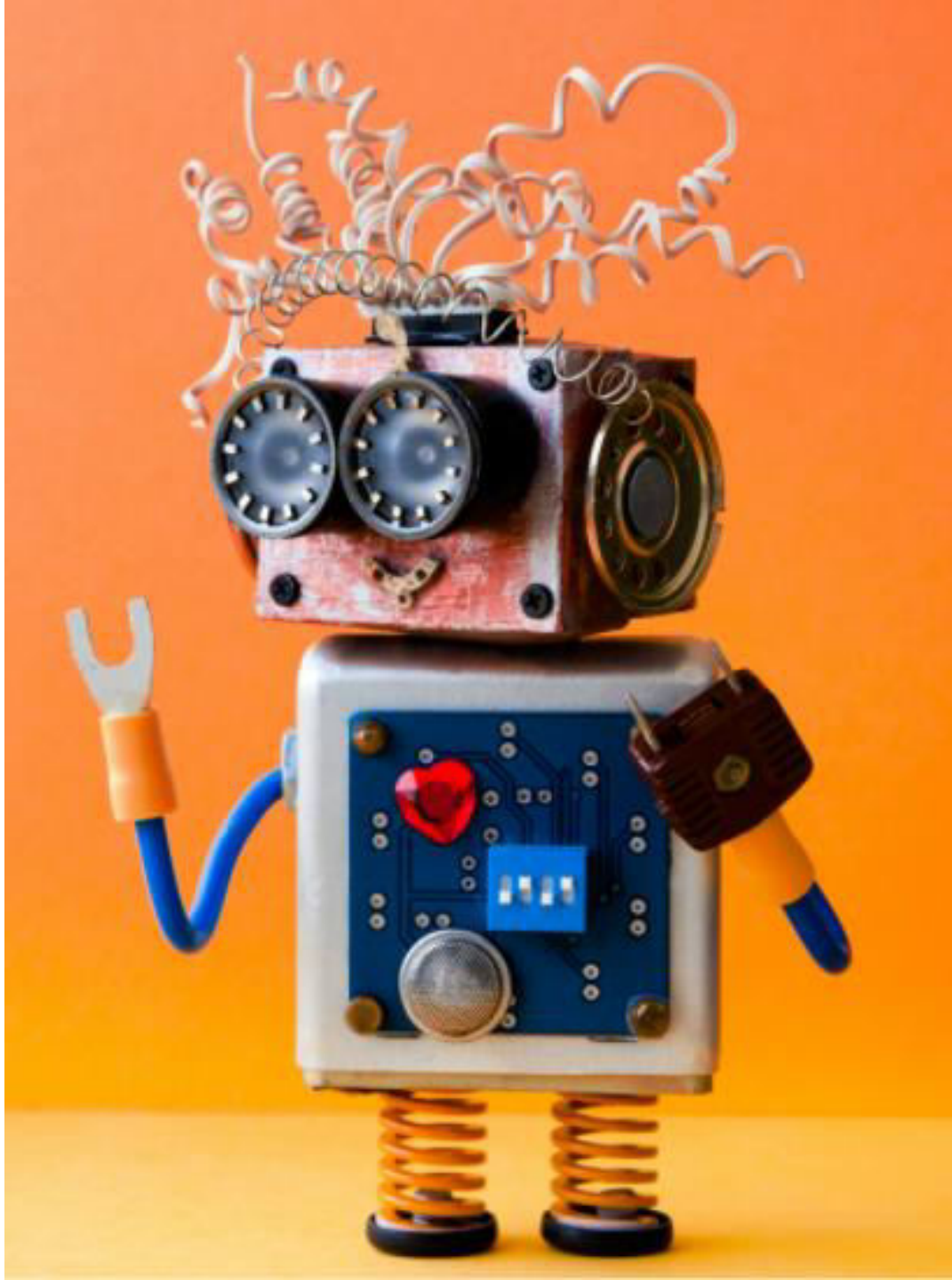


CODE DEMO


Amazon SageMaker Studio
File
Edit
View
Run
Kernel
Git
Tabs
Settings
Help
Launcher
KNN Algo in SageMaker for K
Markdown
Unknown
No Kernel
Share
Name
Last Modified
bike_sharing_daily.csv 2 days ago
cardio_train.csv 6 days ago
Credit Card Default Prediction Using XG-boost i... 6 days ago
Hyperparameters Optimization for XG-Boost in ... 2 days ago
Hyperparameters Optimization in SageMaker.ip... 19 hours ago
KNN Algo in SageMaker for Kyphosis Disease Cla... seconds ago
kyphosis.csv seconds ago
Machine Learning Classification - Telecom Custo... 19 hours ago
Multiple Linear Regression with SageMaker Line... 19 hours ago
Regression with XGboost in SageMaker.ipynb 19 hours ago
ROC.png 19 hours ago
telecom_churn.csv 19 hours ago
train.csv 20 hours ago
UCI_Credit_Card.csv 6 days ago
used_car_price.csv 2 days ago
validation.csv 20 hours ago
Starting notebook kernel...
CODING TASK #1: IMPORT LIBRARIES AND DATASETS
[1]: !pip install --upgrade seaborn
/opt/conda/lib/python3.7/site-packages/secretstorage/dhcrypto.py:16: CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_bytes instead
from cryptography.utils import int_from_bytes
/opt/conda/lib/python3.7/site-packages/secretstorage/util.py:25: CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_bytes instead
from cryptography.utils import int_from_bytes
Requirement already satisfied: seaborn in /opt/conda/lib/python3.7/site-packages (0.11.2)
Requirement already satisfied: scipy>=1.0 in /opt/conda/lib/python3.7/site-packages (from seaborn) (1.4.1)
Requirement already satisfied: numpy>=1.15 in /opt/conda/lib/python3.7/site-packages (from seaborn) (1.20.3)
Requirement already satisfied: matplotlib>=2.2 in /opt/conda/lib/python3.7/site-packages (from seaborn) (3.1.3)
Requirement already satisfied: pandas>=0.23 in /opt/conda/lib/python3.7/site-packages (from seaborn) (1.0.1)
Requirement already satisfied: cycler>=0.10 in /opt/conda/lib/python3.7/site-packages (from matplotlib>=2.2->seaborn) (0.10.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /opt/conda/lib/python3.7/site-packages (from matplotlib>=2.2->seaborn) (2.4.6)

FINAL END-OF-DAY CAPSTONE PROJECT

EASY ADVANCED



PROJECT TASKS

Using the same dataset, perform the following tasks:

1. Train a random forest classifier model in SKLearn and assess its performance
2. Plot the confusion matrix
3. Print the classification Report
4. Train a decision tree classifier model in SKLearn and assess its performance
5. Plot the confusion matrix
6. Print the classification Report
7. Calculate Feature Importance
8. Train an XG-Boost Algorithm in SageMaker
9. Comment on the results
10. Delete the endpoint