Glossary terms from week 4

Terms and definitions from Course 6, Week 4

%%time: A magic command that provides the runtime of the cell it's entered in to

AdaBoost: (Refer to **adaptive boosting**)

Adaptive boosting: A boosting methodology where each consecutive base learner assigns greater weight to the observations incorrectly predicted by the preceding learner

Bagging: A technique used by certain kinds of models that use ensembles of base learners to make predictions; refers to the combination of bootstrapping and aggregating

Base learner: Each individual model that comprises an ensemble

Black-box model: Any model whose predictions cannot be precisely explained

Boosting: A technique that builds an ensemble of weak learners sequentially, with each consecutive learner trying to correct the errors of the one that preceded it

Bootstrapping: Refers to sampling with replacement

Child node: A node that is pointed to from another node

Cross-validation: A process that uses different portions of the data to test and train a model on different iterations

Decision node: A node of the tree where decisions are made

Decision tree: A flowchart-like structure that uses branching paths to predict the outcomes of events, or the probability of certain outcomes

Ensemble learning: Refers to building multiple models and aggregating their predictions

Ensembling: (Refer to **ensemble learning**)

Extrapolation: A model's ability to predict new values that fall outside of the range of values in the training data

Gradient boosting: A boosting methodology where each base learner in the sequence is built to predict the residual errors of the model that preceded it

Gradient boosting machines (GBMs): Model ensembles that use gradient boosting

GridSearch: A tool to confirm that a model achieves its intended purpose by systematically checking every combination of hyperparameters to identify which set produces the best results, based on the selected metric

Hyperparameter tuning: Refers to changing parameters that directly affect how the model trains, before the learning process begins

Hyperparameters: Parameters that can be set by the modeler before the model is trained

Leaf node: The nodes where a final prediction is made

learning_rate: In XGBoost, a hyperparameter that specifies how much weight is given to each consecutive tree's prediction in the final ensemble

Magic commands: Commands that are built into IPython to simplify common tasks; always begin with either "%" or "%%"

Magics: (Refer to **magic commands**)

max_depth: In tree-based models, a hyperparameter that controls how deep each base learner tree will grow

max_features: In decision tree and random forest models, a hyperparameter that specifies the number of features that each tree randomly selects during training called "colsample_bytree" in XGBoost

min_samples_leaf: In decision tree and random forest models, a hyperparameter that defines the minimum number of samples for a leaf node called "min_child_weight" in XGBoost

min_samples_split: In decision tree and random forest models, a hyperparameter that defines the minimum number of samples that a node must have to split into more nodes

min_child_weight: In XGBoost models, a hyperparameter indicating that a tree will not split a node if it results in any child node with less weight than this value called "min_samples_leaf" in decision tree and random forest models

min_samples: In DBSCAN clustering models, a hyperparameter that specifies the number of samples in an ε-neighborhood for a point to be considered a core point (including itself)

Model selection: The process of determining which model should be the final product and put into production

Model validation: The set of processes and activities intended to verify that models are performing as expected

n_estimators: In random forest and XGBoost models, a hyperparameter that specifies the number of trees your model will build in its ensemble

Random forest: An ensemble of decision trees trained on bootstrapped data with randomly selected features

Root node: The first node of the tree, where the first decision is made

Shrinkage: (Refer to **learning_rate**)

Tree-based learning: A type of supervised machine learning that performs classification and regression tasks

Weak learner: A model that performs slightly better than randomly guessing

XGBoost (extreme gradient boosting): An optimized GBM package

Terms and definitions from previous weeks

A

Accuracy: The number of correct predictions divided by the total number of predictions

Affinity: The metric used to calculate the distance between points/clusters

Agglomerative clustering: A clustering methodology that works by first assigning every point to its own cluster, then progressively combining clusters based on intercluster distance

Average: The distance between each cluster's centroid and other clusters' centroids

B

Bayes' Theorem: An equation that can be used to calculate the probability of an outcome or class, given the values of predictor variables

C

Categorical variables: Variables that contain a finite number of groups or categories

Centroid: The center of a cluster determined by the mathematical mean of all the points in that cluster

Class imbalance: When a dataset has a predictor variable that contains more instances of one outcome than another

Collaborative filtering: A technique used by recommendation systems to make comparisons based on who else liked the content

Complete: The maximum pairwise distance between clusters

Content-based filtering: A technique used by recommendation systems to make comparisons based on attributes of content

Continuous variables: Variables that can take on an infinite and uncountable set of values

Customer churn: The business term that describes how many and at what rate customers stop using a product or service, or stop doing business with a company

D

DBSCAN: A clustering methodology that searches data space for continuous regions of high density; stands for "density-based spatial clustering of applications with noise"

Decision tree: A flowchart-like structure that uses branching paths to predict the outcomes of events, or the probability of certain outcomes

Discrete features: Features with a countable number of values between any two values

distance_threshold: A hyperparameter in agglomerative clustering models that determines the distance above which clusters will not be merged

Documentation: An in-depth guide that is written by the developers who created a package that features very specific information on various functions and features

Downsampling: The process of removing some observations from the majority class, making it so they make up a smaller percentage of the dataset than before

E

eps (Epsilon): In DBSCAN clustering models, a hyperparameter that determines the radius of a search area from any given point

F

F1-Score: The harmonic mean of precision and recall

Feature engineering: The process of using practical, statistical, and data science knowledge to select, transform, or extract characteristics, properties, and attributes from raw data

Feature extraction: A type of feature engineering that involves taking multiple features to create a new one that would improve the accuracy of the algorithm

Feature selection: A type of feature engineering that involves selecting the features in the data that contribute the most to predicting the response variable

Feature transformation: A type of feature engineering that involves modify existing features in a way that improves accuracy when training the model

I

Inertia: The sum of the squared distances between each observation and its nearest centroid

Integrated Development Environment (IDE): A piece of software that has an interface to write, run, and test a piece of code

K

K-means: An unsupervised partitioning algorithm used to organize unlabeled data into groups, or clusters

L

Linkage: The method used to determine which points/clusters to merge

M

Machine learning: The use and development of algorithms and statistical models to teach computer systems to analyze and discover patterns in data

min_samples: In DBSCAN clustering models, a hyperparameter that specifies the number of samples in an ε-neighborhood for a point to be considered a core point (including itself)

N

n_clusters: In K-means and agglomerative clustering models, a hyperparameter that specifies the number of clusters in the final model

Naive Bayes: A supervised classification technique that is based on Bayes's Theorem with an assumption of independence among predictors

P

Plan stage: The part of the PACE workflow process where a data professional first starts thinking about what the problem actually is and what needs to be done to find a solution

Popularity bias: The phenomenon of more popular items being recommended too frequently

Posterior probability: The probability of an event occurring after taking into consideration new information

Precision: The proportion of positive predictions that were correct to all positive predictions

R

Recall: The proportion of actual positives that were identified correctly to all actual positives

Recommendation systems: Unsupervised learning techniques that use unlabeled data to offer relevant suggestions to users

S

Silhouette analysis: The comparison of different models' silhouette scores

Silhouette score: The mean of the silhouette coefficients of all the observations in a model

Single: The minimum pairwise distance between clusters

Supervised machine learning: A category of machine learning that uses labeled datasets to train algorithms to classify or predict outcomes

Supervised model: A machine learning model that is used to make predictions about unseen events

U

Unsupervised model: A machine learning model that is used to discover the natural structure of the data, finding relationships within unlabeled data

Upsampling: The process of taking observations from the minority class and either adding copies of those observations to the dataset or generating new observations to add to the dataset

W

Ward: Merges two clusters whose merging will result in the lowest inertia

Z

"Zero Frequency" problem: Occurs when the dataset has no occurrences of a class label and some value of a predictor variable together

Mark as completed