Contents

[Background 1](#_Toc92148899)

[Distributions 2](#_Toc92148900)

[Discrete 2](#_Toc92148901)

[Continuous 2](#_Toc92148902)

[Concepts 2](#_Toc92148903)

[Model Evaluation 2](#_Toc92148904)

[Regression 2](#_Toc92148905)

[Classification 3](#_Toc92148906)

[Models 3](#_Toc92148907)

[Linear Regression 3](#_Toc92148908)

[Logistic Regression 3](#_Toc92148909)

[Decision Trees 3](#_Toc92148910)

[Random Forest 3](#_Toc92148911)

[Support Vector Machines 3](#_Toc92148912)

[Multiclass Prediction 4](#_Toc92148913)

[k-Nearest Neighbors 4](#_Toc92148914)

[Dimensionality Reduction 4](#_Toc92148915)

[Factor Analysis 4](#_Toc92148916)

[Natural Language Processing 4](#_Toc92148917)

[Neural Network 5](#_Toc92148918)

[Convolutional Neural Network 5](#_Toc92148919)

[Recurrent Neural Network 6](#_Toc92148920)

[Boosting 6](#_Toc92148921)

[Reinforcement Learning 6](#_Toc92148922)

[Anomaly Detection 7](#_Toc92148923)

[Time Series 8](#_Toc92148924)

[Naive Bayes 8](#_Toc92148925)

[Statistics 8](#_Toc92148926)

# Background

This glossary presents a concise list of topics for machine learning and deep learning. It is adapted from an existing cheat sheet (links below). A more detailed glossary is In the detailed glossary file

This document based on <https://github.com/aaronwangy/Data-Science-Cheatsheet> by Aaron Wang

Maintained by Ajit Jaokar. If you want to get updates to this Glossary and other content about our teaching, please [sign up for my newsletter on LinkedIn](https://www.linkedin.com/newsletters/artificial-intelligence-6793973274368856064/)

# Distributions

## Discrete

* **Binomial** -
* **Geometric**
* **Poisson**

## Continuous

* **Uniform**
* **Normal/Gaussian**
* **Central Limit Theorem**
* **Empirical Rule** - 68%, 95%, and 99.7% of values lie within one, two, and three standard deviations of the mean
* **Normal Approximation** - discrete distributions such as Binomial and Poisson can be approximated using z-scores when np, nq, and λ are greater than 10
* **Exponentia**l -
* **Gamma**

# Concepts

* **Prediction Error** = Bias + Variance + Irreducible Noise
* **Bias** - wrong assumptions when training → can’t capture underlying patterns → underfit
* **Variance** - sensitive to fluctuations when training→ can’t generalize on unseen data → overfit. The bias-variance tradeoff attempts to minimize these two sources of error, through methods such as: Cross validation to generalize to unseen data amd Dimension reduction and feature selection. In all cases, as variance decreases, bias increases.
* ML models can be divided into two types: – **Parametric** - uses a fixed number of parameters with respect to sample size and **Non-Parametric** - uses a flexible number of parameters and doesn’t make particular assumptions on the data
* Cross Validation - validates test error with a subset of training data, and selects parameters to maximize average performance **– k-fold -** divide data into k groups, and use one to validate **– leave-p-out** - use p samples to validate and the rest to trai
* **Supervised vs unsupervised**
* **Ensembles** Ensemble learning helps improve machine learning results by combining several models. This approach allows the production of better predictive performance compared to a single model. Basic idea is to learn a set of classifiers (experts) and to allow them to vote. Bagging and Boosting are two types of Ensemble Learning.

Bagging: It is a homogeneous weak learners’ model that learns from each other independently in parallel and combines them for determining the model average.

Boosting: It is also a homogeneous weak learners’ model but works differently from Bagging. In this model, learners learn sequentially and adaptively to improve model predictions of a learning algorithm. <https://www.geeksforgeeks.org/bagging-vs-boosting-in-machine-learning/>

# Model Evaluation

## Regression

* Mean Squared Error (MSE)
* Sum of Squared Error (SSE)
* Total Sum of Squares (SST)
* Adjusted R2

## Classification

* Predict Yes Predict No
* Precision
* Recall
* Sensitivity
* Specificity
* Working with imbalanced classes
* ROC curve
* Precision-Recall Curve

# Models

## Linear Regression

* Models linear relationships between a continuous response and explanatory variables
* Ordinary Least Squares OLS regression
* Assumptions of OLS regression
* Regularization: Add a penalty for large coefficients to the cost function, which reduces overfitting.
* LASSO (L1)
* Ridge (L2)

## Logistic Regression

* Predicts probability that y belongs to a binary class. Estimates parameters through maximum likelihood estimation (MLE) by fitting a logistic (sigmoid) function to the data.
* Assumptions of logistic regression

## Decision Trees

* Classification and Regression Tree
* Advantages and limitations of trees

## Random Forest

* Trains an ensemble of trees that vote for the final prediction

## Support Vector Machines

* Separates data between two classes by maximizing the margin between the hyperplane and the nearest data points of any class.

## Multiclass Prediction

## k-Nearest Neighbors

* Non-parametric method that calculates ^y using the average value or most common class of its k-nearest points.

## Dimensionality Reduction

* High-dimensional data can lead to the curse of dimensionality, which increases the risk of overfitting and decreases the value added.

## Factor Analysis

Describes data using a linear combination of k latent factors.

# Natural Language Processing

Transforms human language into machine-usable code

Processing Techniques

* Tokenization - splits text into individual words (tokens)
* Lemmatization - reduces words to its base form based on dictionary definition (am, are, is ! be)
* Stemming - reduces words to its base form without context (ended ! end)
* Stop words - removes common and irrelevant words (the, is)
* Markov Chain - stochastic and memoryless process that predicts future events based only on the current state
* n-gram - predicts the next term in a sequence of n terms based on Markov chains
* Bag-of-words - represents text using word frequencies, without context or order
* tf-idf - measures word importance for a document in a collection (corpus), by multiplying the term frequency (occurrences of a term in a document) with the inverse document frequency (penalizes common terms across a corpus)
* Cosine Similarity - measures similarity between vectors
* Word Embedding Maps words and phrases to numerical vectors
* word2vec - trains iteratively over local word context windows, places similar words close together, and embeds sub-relationships directly into vector
* GloVe - combines both global and local word co-occurrence data to learn word similarity
* BERT –
* Sentiment Analysis Extracts the attitudes and emotions from text
* Polarity - measures positive, negative, or neutral opinions
* Topic Modelling: Captures the underlying themes that appear in documents
* Latent Dirichlet Allocation (LDA) –

# Neural Network

Feeds inputs through different hidden layers and relies on

weights and nonlinear functions to reach an output

* Perceptron - the foundation of a neural network that multiplies inputs by weights, adds bias, and feeds the result z to an activation function
* Activation Function - defines a node's output: Sigmoid, ReLU, Tanh
* Softmax - given final layer outputs, provides class probabilities
* Loss Function
* Gradient Descent
* Backpropagation
* To prevent over\_tting, regularization can be applied by: Stopping training when validation performance drops; Dropout; Embedding weight penalties into the objective function; Batch Normalization - stabilizes learning by normalizing inputs to a layer
* Stochastic Gradient Descent

# Convolutional Neural Network

* Analyzes structural or visual data by extracting local features
* Convolutional Layers - iterate over windows of the image, applying weights, bias, and an activation function to create feature maps. Different weights lead to different features maps.
* Pooling - downsamples convolution layers to reduce dimensionality and maintain spatial invariance, allowing detection of features even if they have shifted slightly. Common techniques return the max or average value in the pooling window.
* The general CNN architecture is as follows: 1. Perform a series of convolution, ReLU, and pooling operations, extracting important features from the data 2. Feed output into a fully-connected layer for classification, object detection, or other structural analyses

# Recurrent Neural Network

* Predicts sequential data using a temporally connected system that captures both new inputs and previous outputs using hidden states
* RNNs can model various input-output scenarios, such as many-to-one, one-to-many, and many-to-many
* Long Short-Term Memory - learns long-term dependencies using gated cells and maintains a separate cell state from what is output.

# Boosting

* In machine learning, boosting is an ensemble meta-algorithm for primarily reducing bias, and also variance in supervised learning, and a family of machine learning algorithms that convert weak learners to strong ones. A weak learner is defined to be a classifier that is only slightly correlated with the true classification (it can label examples better than random guessing). In contrast, a strong learner is a classifier that is arbitrarily well-correlated with the true classification.(adapted from Wikipedia)
* AdaBoost
* Gradient
* XGBoost

# Reinforcement Learning

* Maximizes future rewards by learning through state-action pairs. That is, an agent performs actions in an environment, which updates the state and provides a reward.
* Multi-armed Bandit Problem - a gambler plays slot machines with unknown probability distributions and must decide the best strategy to maximize reward. This example is the exploration-exploitation tradeoff, as the best long-term strategy may involve short-term sacrifices.
* RL is divided into two types, with the former being more common: Model-free - learn through trial and error in the environment; Model-based - access to the underlying (approximate) state-reward distribution
* Q-Value Q(s; a) - captures the expected discounted total future reward given a state and action
* Policy - chooses the best actions for an agent at various states
* Deep RL algorithms
* Q-Learning
* Deep Q Network
* Policy Gradient Learning
* Actor-Critic Model

# Anomaly Detection

* Identifies unusual patterns that differ from the majority of the data. Assumes that anomalies are: Rare - the minority class that occurs rarely in the data; Different - have feature values that are very different from normal observations
* Anomaly detection techniques spans a wide range, including methods based on: Statistics - relies on various statistical methods to identify outliers, such as Z-tests, boxplots, interquartile ranges, and variance comparisons
* Density - useful when data is grouped around dense neighbourhoods, measured by distance. Methods include k-nearest neighbors, local outlier factor, and isolation forest.
* Isolation Forest - tree-based model that labels outliers based on an anomaly score
* Clusters - data points outside of clusters could potentially be marked as anomalies
* Autoencoders - unsupervised neural networks that compress data through an encoder and reconstruct it using a decoder. Can be used for anomaly detection
* Hidden Markov Model - Anomalies are observations that are unlikely to occur across states in an HMM. HMMs can be applied to many problems such as signal processing and part of speech tagging.

# Time Series

Extracts characteristics from time-sequenced data, which may exhibit the following characteristics:

* **Stationarity** - statistical properties such as mean, variance, and auto correlation are constant over time
* **Trend** - long-term rise or fall in values
* **Seasonality** - variations associated with specific calendar times, occurring at regular intervals less than a year
* **Cyclicality** - variations without a fixed time length, occurring in periods of greater or less than one year
* **Autocorrelation** - degree of linear similarity between current and lagged values
* **Exponential Smoothing** - uses an exponentially decreasing weight to observations over time, and takes a moving average.
* ARIMA –

# Naive Bayes

Classifies data using the label with the highest conditional probability. Naive because it

assumes variables are independent.

## Statistics

* p-value
* Type I Error
* Type II Error
* Confidence Level
* t-test
* Degrees of Freedom
* Goodness of fit
* ANOVA
* Conditional Probability
* Variance
* Covariance
* Correlation
* A/B testing Examines user experience through randomized tests with two variants.