Machine Learning Review

Typically, machine learning engineers don't like it when people claim that machine learning is just fancy statistics. Some of them will fight you over this claim. However, if you pick up a statistics book from, say, 1948, all of the machine learning math is in there, far before the invention of the computer.

Fancy Statistics

Machine Learning: Inspired by structure of the human brain and is particularly effective in feature detection.

Deep Learning: Model extracts features and classifies as a single step instead of two separate steps like Machine Learning.

Supervised Learning = Like learning from a teacher: Model trained from a pre-defined dataset before it starts making decisions when given new data.

Unsupervised Learning = Like self-teaching: Model learns through observation and finds structures in data. Given a dataset, the model is left alone to automatically find patterns and relationships in that data by creating clusters.

Reinforcement Learning: Model learns with hit and trial method. Based on reward or penalty for every action it performs

Classification Algorithm	Regression Algorithm
Predicts belonging to a category or set Class labels Ex: cheap vs. affordable, long vs. short	Predicts a value from a set Continuous values Ex: Size, area, location

Bias = wrong assumptions when training the model (underfitting): Error that causes one sampling group to be selected more than other groups included in the experiment. Model not properly learning the data.

Recall: Ratio of a number of events you can correctly recall to a number of all correct events. What if some of your answers are wrong?

Precision: The ratio of a number of events you can correctly recall to a number of all events you can recall.

© 2023 – Jessica Chipera, MBA, Doctoral Candidate Creative Commons / Not for Profit

Overfitting: When a model includes too much that occurred by chance. Basically, it is like the algorithm memorizes the data that was used to build it.

Minimize overfitting by:

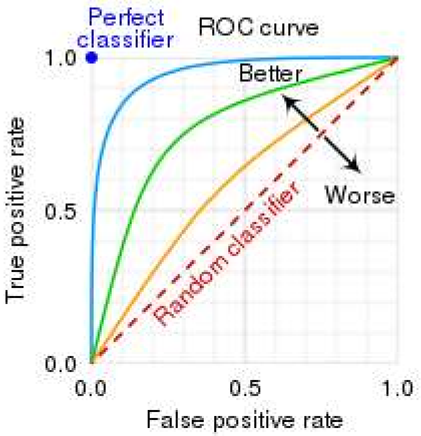
- Using a separate set of data to train the model and then use a test set to determine the predictive capabilities of the model. Usually, 80% of the model should be in the training set and 20% should be in the test set.
- You could also just choose a simpler model or collect more data.

>> import numpy as np	Numpy does scientific computing
>> import pandas as pd	Pandas is used for use of dataframes
>> import matplotlib.pyplot as plt	for data visualization
>> import seaborn as sns	for data visualization
>> import math	for math functions
>> import scipy.stats as stats	for statistics
>> import random	for creating random variables

ROC Curve = Receiver Operating Characteristic Curve: Measures the predictability of the various relationships present in the data.

In this case, a straight line is bad.

```
>> from sklearn.dataset import load_iris
>> from sklearn.model_selection import train_test_split
>> from sklearn.linear_model import LogisticRegression as lr
>> from sklearn.preprocessing import LabelBinarizer as lb
>> from sklearn.metrics import RocCurveDisplay as roc
```



CONFUSION MATRIX AKA Error Matrix: A table that summarizes the performance of a classification algorithm.

```
>> from sklearn import metrics
>> from sklearn.metrics.confustion_matrix (y_true, y_pred, *, labels = None, sample_weight = None,
   normalize = None)
```

	PREDICTED NO	**PREDICTED YES**	Total
ACTUAL NO	True Negative	False Positive	
ACTUAL YES	False Negative	True Positive	
Total			

© 2023 – Jessica Chipera, MBA, Doctoral Candidate Creative Commons / Not for Profit

True Positive = A fire alarm goes off and there's a fire
False Positive = A fire alarm goes off and there's no fire
False Negative = No alarm, but there's a fire
True Negative = No alarm and there's no fire

Regression Algorithms

Regression = Infer or predict a variable based on one or more variables

Independent variables = Predictors

Dependent variables = Criteria

Simple Regression	Multiple Regression	Logistic Regression
- One independent variable - Infer one dependent variable	- Multiple independent variables - Infer one dependent variable	- Yes or no answers only - Categorical variables
Independent variable = metric, ordinal, or nominal	Independent variable = metric, ordinal, or nominal	Independent variable = metric, ordinal, or nominal
Dependent variable = Metric	Dependent variable = Metric	Dependent variable = Ordinal or nominal

Linear Regression

Linear Regression = A type of Simple Regression: A straight-line relationship between dependent and independent variables.

- Data can be transformed into a linear relationship with scalars or logarithm functions
- Regression error (epsilon) must be normal distribution
- Must not have multicollinearity or instability of regression coefficients
- No heteroskedacity

This is a straight line, so remember that we will be using this familiar formula:

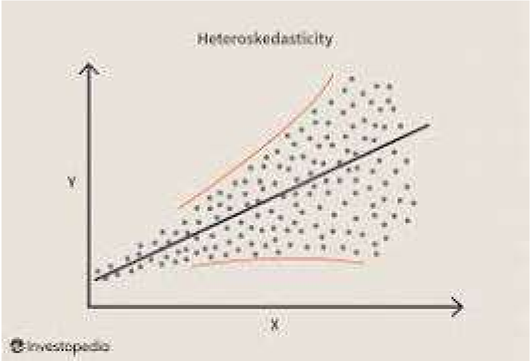
$$\hat{y} = b \cdot x + a$$

```
>> from sklearn.linear_model import LinearRegression
```

Multicollinearity: When two or more predictors correlate strongly with each other. In other words, the effect of individual variables cannot be clearly separated.

© 2023 – Jessica Chipera, MBA, Doctoral Candidate Creative Commons / Not for Profit

Heteroskedacity: When the standard deviations of a predicted variable, monitored over different values of an independent variable, are non-constant



Quadratic Regression

Gradient Descent: Control for the influence of a third (or multiple third-party) variable(s) in your regression algorithm:

Has the same properties as a simple linear regression model as listed above.

$$\hat{y} = b_1 \cdot x_1 + b_2 \cdot x_2 + \cdots + b_k \cdot x_k + a$$

R = Multiple correlation coefficient: Measures the relationship or correlation between the dependent variable and the independent variables

R^2 = Coefficient of Determination: Indicates how much of the variance of the dependent variable can be explained by the independent variables.

Classification Algorithms

Decision Trees

Can be supervised or unsupervised learning

```
>> from sklearn import tree
```

Random Forest: Trains a group of trees that vote for the final prediction. Reduces overfitting problem with decision trees by attempting to decorrelate the trees.

1) Random sunset of predictors is allowed for each split in the tree.
2) Usually, $\sqrt{p}$ subsets are allowed, where p = the number of predictors.
3) Typically, used in combination with bagging, especially when “p” is a huge number and a lot of the predictors are correlated with one another (multicollinearity).

```
>> from sklearn.ensemble import RandomForestClassifier
>> from sklearn.datasets import make_classification
```

Bootstrapping: Resampling our original sample with replacement many times (like thousands of times) with the same sample size.

© 2023 – Jessica Chipera, MBA, Doctoral Candidate Creative Commons / Not for Profit

GINI IMPURITY VS. ENTROPY

Gini Impurity	Entropy
• The probability of a random variable being classified correctly if you randomly pick a label according to the distribution of the branch. • Less computationally expensive than entropy	• Measurement of lack of information. • Measures messiness of your data. • Information gain is calculated by making a split, which is the difference in entropies. Information gain increases as you approach the leaf node.

BAGGING VS. BOOSTING

Bagging (Bootstrap Aggregation) = Bootstrapping + Predictive Model	**Boosting**
Reduces variance **Reduces overfitting**	**Reduces variance** **Can increase overfitting**
1) Obtain "B" random bootstrap resamples of training sample 2) For each resample, grow a large (low bias, high variance) tree 3) Average / aggregate predictions from all of the trees a. For regression models, take the mean of B predictions For classification, take the majority vote of the B predictions	1) Obtain random training datasets by random sampling 2) Adds new models where previous models failed 3) Weights the data to favor the most unbiased cases Average / aggregate predictions from all of the trees and either takes the mean or majority vote

Out of Bag = like when datapoints fall outside a defined probability universe: Each tree is trained on about 63% of the data, so the out-of-bag 37% can estimate prediction error. Then either average or majority vote it.

K-Nearest Neighbors

K-Nearest Neighbor: A supervised non-parametric calculation of $\hat{y}$ using the average value of its k-nearest points.

```
>> from sklearn.neighbors import NearestNeighbors
```

Minkowski Distance:

$$(\sum |a_i - b_i|^p)^{\frac{1}{p}}$$

$$Where\ p = 1\ if\ Manhattan\ distance; \quad p = 2\ if\ Euclidean\ distance$$

Hamming Distance: The count of the difference between two vectors, sometimes used to compare categorical variables.

Clustering Algorithms

Unsupervised, nonparametric methods that group similar datapoints together based on distance.

K-Means Clustering

K-Means Clustering: Randomly places “k” centroids across normalized data, and observes to the nearest centroid. Then it recalculates centroids as the mean of assignments and repeats until convergence. It uses the median or medoid (which is an actual data point) to be more robust to noise and outliers.

1) Choose the first center randomly
2) Compute the distance between points and the nearest center
3) Choose a new center using a weighted probability distribution proportional to distance
4) Repeat until “k” centers are chosen

```
>> from sklearn.cluster import KMeans
```

Dimension Reduction Algorithms

Principal Component Analysis

```
>> from sklearn.preprocessing import StandardScaler as ss
>> from sklearn.decomposition import PCA
```

t-Stochastic Neighbor Embedding

```
>> from sklearn.preprocessing import StandardScaler as ss
>> from sklearn.manifold import TSNE
```

© 2023 – Jessica Chipera, MBA, Doctoral Candidate Creative Commons / Not for Profit