Intro to Gitmo Science Cheat Sheet

Data Visualization

Measurements

- nominal/categorical - no semantic meaning, like names
- ordinal - ranked order without significance
- interval - ranked order with significance, such as temperature
- ratio - absolute zero that is meaningful, such as weight

Distances

- Euclidean - distance bw two points
- Weighted euclidean - divide both variables by standard deviation
- Covariance - measure of how x and y vary together
- Mahalanobis - used for distance b/w data rows, accounts for scaling of coordinate axes, reduces to euclidean with identity matrix
- Mahalanobis takes the distance between two points, P and C and normalizes them over the standard deviation of the feature
  sqrt($\sum_{i=1}^{d}(\frac{P_i - C_i}{\sigma_i})^2$)

Transformation

logit - transform $p = [0, 1]$ into a real line

Data Wrangling & Preprocessing

Tasks

- Cleaning, removing commas, different values for same thing, ascii vs utf-8, etc
- google refine & data wrangler do this
- Integration with multiple data sets
- Reduction & Compression - reduce dimensions
- Transformation via normalization - smoothing, attribute construction
- Noisy data can be handled via binning, regression, clustering, etc

Filling in Missing Data

- global constant
- attribute mean
- most probable value, inference based like bayesian/decision tree

Types of Sampling

- simple random sampling - equal probability for all items
- sampling w/o replacement - selected object removed
- sampling w/ replacement - selected object not removed
- stratified sample - partition data, draw samples from each partition proportionately

Finding Similar Items

Applications

- Find news articles that are the same story
- Remove duplicate web pages on search results

This is difficult because comparing all documents to each other is computationally infeasible (both because of processing time and storage issues)

Shingling

- Used when order of items matters as well as sets
- Size k must be picked such that probability of any shingle being in a document is low - 5 for small documents, 10 for big documents

Min-hashing

- motivation - make smaller signature matrix by hashing columns
- used for jaccardian ($\frac{A \cap B}{A \cup B}$) similarity
- use independent hash functions to generate signature matrix with h rows (where h = number of hash functions) and d columns (where d = number of documents)

Locality-sensitive hashing

- motivation - find documents in signature matrix with jaccardian similarity about threshold S
- hash doc columns into many buckets - unique documents in the same bucket are candidate pairs and band parts in same assumed to be the same
- divide signature matrix into b bands with r rows per band
- hash each band into one of k buckets and balance M hash functions with bands and rows to balance false positives with false negative

Data Streams

Examples of streams: sensor, image data, internet traffic
Stream queries can either be standing (ex: whenever temperature exceeds 100 degrees) or retained (ex: maximum temperature ever)
Random Sample over stream – To take a sample of 1/n elements, a good approach is to hash user ids into one of n buckets, only keeping from one.
Fixed size sampling (Resevoir Sampling) – Store all the first s elements of the stream to S Suppose we have seen n-1 elements, and now the nth element arrives $n > s$. With probability s/n, keep the nth element, else discard it
If we picked the nth element, then it replaces one of the s elements in the sample S, picked uniformly at random
Flajolet-Martin - Distinct elements = 2^r r being tail 0s
Bloom Filter: Probabilty of picking 1 = $(1 - e^{-km/n})^k$
k = num. of h(x); m = 'darts'; n = 'buckets'

Decision Trees

$H(X) = -\sum_{i=1}^{n} P(X = i) \log_2 P(X = i)$
$IG(A, Y) = Hs(Y) - Hs(Y|A)$
Avoid overfitting - Stop growing when data split is not sig OR prune full tree

Components

- Decision nodes - squares
- Chance nodes - circles
- End nodes - triangles

Instances are fed to the tree with as attribute-value pairs
Decision trees are robust against errors, both in classification and input values
Some input values can even be missing!

Information Gain

- Synonym for Kullback-Leibler divergence
- Definition - IG is the reduction in entropy achieved by learning the state of a particular random variable
- This can be used to generate a preferred sequence of attributes with which we can most rapidly narrow down a state - such a sequence is known as a decision tree.

Naive Bayes

$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$
Prior Probability of GREEN: number of GREEN objects / total number of objects
Having formulated our prior probability, we draw a circle around X which encompasses a number (to be chosen a priori) of points irrespective of their class labels. From this we estimate likelihood.
Likelihood of X given GREEN = Number of GREEN in the vicinity of X / Total number of GREEN cases
Posterior probability of X being GREEN = Prior prob * Likelihood

Logistic/Logit Regression

$F(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_0 X)}}$ Classification Algorithm
Takes output from linear regression and makes it positive or negative
Gradient Descent: Need to choose α
Needs many iterations and better for large item sets ie. n^6
Normal Equation: No need to choose α
Don't need to iterate. (On^3). Item sets $n < 10^5$

Linear Regression

Regression Algorithm
See logistic regression.

Support Vector Machines

- Supervised learning model that analyzes and recognizes patterns for classification and regression purposes
- Binary linear classifier: seperates N-dimensional space with N-1 dimensional hyperplane
- Use kernel function to map non-linearly seperable data to higher dimensional space
- Hyperplane is picked to maximize margin between itself and 'support vectors'

Ensemble Methods

These create strong classifiers from weak classifiers.

Boosting

- Can a set of weak learners create a single strong learner?
- Reduces bias primarily, also reduces variance
- Most popular algorithm is AdaBoost, in which all different properties are weighted

Bagging/Bootstrap Aggregating

- Generates new training sets by sampling from training set uniformly and with replacement
- Used in classification and regression
- Reduces variance, helps avoid overfitting
- Random Forests algorithm uses Bagging to construct a collection of decision trees with controlled variance

MLE vs. MAP

- This is used in the Bayes Method for classification
- MLE stands for Maximum Likelihood Estimate - it's the value of the parameter that maximizes likelihood of a certain outcome
- This is the $P(B \mid A)$ part of the equation
- MAP = Maximum A Posteriori - a maximized posterior distribution based on an estimate
- Example - influencing a skewed distribution of voters to align more with what you think SHOULD be the distribution

Regularization

Process of introducing additional information (usually anomalies) to prevent overfitting
Example: least-squares method

- L1 - Produces sparse models, performs feature selection, not differentiable
- L2 - Also known as 'weight decay', generally performs way better than L1

Bias vs. Variance

The Difference

- Bias - error from erroneous assumptions in the learning algorithm
  - High bias causes an algorithm to miss relevant relations (underfitting)
- Variance - error from fluctuations/anomalies in the training set
  - High variance causes noise to have too much of an input (overfitting)

The Problem

In a perfect world, models would capture regularities accurately and generalize well to unseen data. In reality, accomplishing both is almost impossible.

Possible Solutions

- Generalized linear models can be regularized to increase their bias.
- In k-nearest neighbor models, a high value of k leads to high bias and low variance.
- In Instance-based learning, regularization can be achieved by varying the mixture of prototypes and exemplars.
- In decision trees, the depth of the tree determines the variance - decision trees can therefore be pruned to control variance.
- Mixture models and ensemble learning are also methods which help to control the bias/variance disparity.

Evaluation Techniques for Supervised Learning

Supervised learning = data is labeled
Keep in mind – performance of a model may be based on other factors such as Class Distribution, Cost of Miscalculation, Size of Teaching and Test Sets

Confusion Matrix

For [TN,FP,FN,TP] = [a,b,c,d]:

- accuracy = $\frac{a+d}{a+b+c+d}$
- recall/true positive = $\frac{d}{c+d}$
- false positive = $\frac{b}{a+b}$
- precision = $\frac{d}{b+d}$
- F1 score = $2 \cdot \frac{precision \cdot recall}{precision+recall}$

As a metric, accuracy has some problems. For instance the 'spam filter' which filters no emails but has an accuracy of 0.99 because only 1% of emails are spam.
One way to improve upon this would be to use a cost matrix $[C(Y|Y), C(N|Y), C(Y|N), C(N|N)]$ and a cost function $cost = \frac{aC(Y|Y) + dC(N|N)}{aC(Y|Y) + bC(N|Y) + cC(Y|N) + dC(N|N)}$ where C(i,j) is the cost of classifing j as i — cost is proportional to acc if $C(Y|N) = C(N|Y) \wedge C(Y|Y) = C(N|N)$

k-means Clustering

centroid: mean of points in a cluster

- Pick k using elbow method
- Initialize k centroids
- Cluster instances to centroid by finding lowest distance to a centroid
- Recompute centroids
- Repeat 3 and 4 until centroids stabilize or some max iteration reached

Hierarchical Clustering

unsupervised. Distance based clustering. Does not create a specific number of clusters. Number of clusters depends on where you "cut".

Types

- Agglomerative: preferred. bottom up. Each observation starts as an individual cluster, and pairs of clusters are merged, based on distances/similarity measures, as one moves up the hierarchy until one (or k) clusters left

  #### Algorithm

  - each point is a cluster; compute proximity matrix
  - merge closest clusters
  - recompute proximity matrix

- Divisive: top-down. All points are in one large cluster. break into smaller clusters till each point is its own cluster
- uses a minimum spanning tree to construct heirachy of clusters. repeat till singleton clusters

space complexity is n^2 to store proximity matrix
time complexity is n^3 usually, n^2 at best. poor for large data sets

Spectral Clustering

connectivity over compactness

Algorithm

- given: (n x n) Similarity Matrix W and k
- Build the similarity graph: K-Nearest Neighbor graph
- Make a Degree matrix D: # outgoing edges for each node
- Make an Affinity matrix A: weighted adjacency matrix
- Laplacian Matrix L:
  - unnormalized Laplacian: L = D - A
  - normalized Laplacian: $L = D^{-1/2}AD^{-1/2}$
- find the first k eigenvectors of L
- combine k eigenvectors into a matrix V
- do k-means on V to create clusters

Advantages

- superior: looks for connections between data
- useful in hard non-convex clustering
- data represented in low-dimensional space is easier to cluster

Similarity function and choosing K

- similarity function for matrix: kernel function
- Gaussian similarity function: $w_{i,j} = e^{\frac{||x_i - x_j||^2}{2\sigma^2}}$
- choose k that maximizes the distance between consecutive eigenvalues

Evaluation Techniques for Unsupervised Learning

Unsupervised learning = data is not tagged - algorithm clusters data into different groups

- purity of a cluster: $\frac{|largestcluster|}{\#clusters}$
- External: matching clusters to externally provided labels. Make confusion matrix
- Internal
  - correlation: measure pearson correlation of proximity matrix and incidence matrix. higher correlation, better clustering
  - sum of squared error for all clusters in clustering, smaller SSE, better clustering
  - silhouette coefficient: ratio of average distance to elements in the same cluster (a) with average distance to elements in other clusters(b). $s = 1 - \frac{a}{b}$ if s closer to 1, better clustering. $a \geq b$ is rare, means bad clustering, noisy data and outliers
- Relative: comparing the results of different clustering heuristics

Dimensionality Reduction

Goal

Reduce number of dimensions in matrix via compression
Example: many pictures of a face from different angles can be used to reconstruct a model of the original face

- Discover hidden correlations
- Remove redundant/noisy features
- Easier storage/visualization/processing of data

Rank of a matrix indicates its dimension - the theory of eigenvectors provides the basis for dimensionality reduction.