Computational Tools for Data Science

Week 1 Lecture: Introduction

What is Data Mining/Science?

“Data Mining” originally used in statistics: Attempting to extract info not supported by the data.

1990’s: Data mining

2010: Big Data

Today: Data Science

What it means:
Use the most powerful hardware, the most powerful programming systems, and the most efficient algorithms to solve problems in science, commerce, healthcare, government, etc.

What is Data Mining/Science?

- **Given lots of data**
- **Discover patterns and models that are:**
  - **Valid:** hold on new data with some certainty
  - **Useful:** should be possible to act on the item
  - **Unexpected:** non-obvious to the system
  - **Understandable:** humans should be able to interpret the pattern

Data Mining Tasks

- **Descriptive Methods**
  - Find human-interpretable patterns that describe the data
    - **Example:** Clustering

- **Predictive Methods**
  - Use some variables to predict unknown
    or future values of other variables
    - **Example:** Recommender systems

Modeling

- **Create a model of your data that**
  - Provides a good description of your data
  - Allows you to make predictions about new data

- **Example:** Detecting phishing emails
  - The model could be weights on words
    - Phrases appearing unusally often in phishing emails receive positive weights
    - Negative weights for words that do not often appear in phishing emails
    - Sum the weights of all words in a given email to decide if it is a phishing attack
  - Easy to use, hard to find the weights

Statistical Modeling

- **Find an underlying probability distribution from which the data is drawn**

- **Example:**
  - Our data is a set of numbers (process that outputs a number)
  - By sampling we might guess it comes from a Gaussian distribution and approximate the mean and standard deviation
  - These parameters completely characterize the distribution and would be the model of our data

Machine Learning

- **Use data to train one of many types of algorithms used in ML**
- **The resulting parameters are the model of the data**
- **Best used when we do not fully understand what the data tells us about our problem**
  - **Example:** Netflix Challenge
- **Less effective when we better understand the data**
  - **Example:** Finding resumes online (WhizBang! Labs)
- **Often yield a model that we cannot fully explain**
  - Fine for detecting spam
  - Potentially bad for determining insurance costs

Approaches to Modeling

- **Build a (random) process that could have generated the data**

- **Summarization**
  - Summarizing the data succinctly and approximately

- **Feature Extraction**
  - Extracting the most prominent features of the data and ignoring the rest

Examples of Summarization

- **Google's PageRank algorithm**
  - The whole web is summarized by a single number for each page

- **Clustering**
  - Data viewed as points in multidimensional space
  - Create "clusters" of points that are close to each other
  - Clusters are summarized by, e.g., their centers and the average distance from the center to points in the cluster

Examples of Feature Extraction

- **Frequent Itemsets**
  - For data that consist of “baskets” or sets of items
  - Look for small sets of items that appear together in many baskets
  - These ”frequent itemsets” are the characterization of the data that we want
  - The “features” we are extracting are the strongest dependencies/connections among the items
  - **Example:** Actual market baskets

- **Similar Items**
  - Want to find pairs of sets that have a relatively large fraction of their elements in common
  - **Example:** sets of items customers have bought on Amazon
  - Look for ”similar” customers and recommend something many of them have bought

Some useful things to know

TF.IDF measure of word importance

Bonferroni’s Principle

Power Laws/Matthew Effect

TF.IDF Measure of word importance

- **Often want to categorize documents by topic**

- **A simple way is to just use the individual words in each document**
  - The topic(s) of a document will be identified by special words related to that topic
  - E.g. Articles about baseball would use “bat”, “pitch”, ”run”, etc. many times

- **Cannot come up with a list of words for *every* topic**
  - Want to reverse engineer the topics from the words in the documents

TF.IDF Measure of word importance

- **Problem: How do we decide which words in a document are significant?**

- **Most frequent words don't work**
  - "the", "and", "that", etc.
  - Often remove the several hundred most common words (stop words)

- **Want relatively rare words, but not all rare words**
  - E.g., "albeit", "notwithstanding", etc.

- **Want words that appear fairly often in a document, but do not appear in too many documents**

TF.IDF Measure of word importance

- N documents
- $f_{t,d}$ = # of occurrences of term/word t in document d
- $TF_{t,d} = f_{t,d} / \max_w f_{w,d}$ **(Term Frequency)**
  - So most frequent word in a document gets $TF = 1$
- n_t = # of documents term t appears in
- $IDF_t = log_2(\frac{N}{n_t})$ **(Inverse Document Frequency)**
  - Large if term t appears in few documents
- $TF.IDF$ score for term t in document $d = TF_{t,d} \cdot IDF_t$
- Terms with high $TF.IDF$ score are often the terms that best characterize the topic of a document

TF.IDF Example

- $N = 2^{20} = 1048576$ documents
- Term t appears in $2^{10} = 1024$ documents
- $IDF_t = log_2\left(\frac{2^{20}}{2^{10}}\right) = 10$
- Document d contains term t 20 times (i.e. $f_{t,d} = 20$) and this is the most frequent term in document d
  - $TF_{t,d} = 1$ and so $TF.IDF$ score is 10
- If instead $f_{t,d} = 1$ and $\max_w f_{w,d} = 20$
  - Then $TF_{t,d} = 1/20$ and the $TF.IDF$ score is $1/2$

Bonferroni’s Principle

- **Not all patterns are meaningful**

- **Certain patterns/events in your data that you are interested in might also occur randomly**

- **The principle:**
  - Calculate the expected number of the events you are looking for, assuming the data is random
  - If this number is significantly higher than the number of “real” or “meaningful” instances you expect to find, then you should expect that almost anything you find is bogus

Bonferroni's Principle: Example

- Suppose evil-doers preiodically gather at a hotel to plot
- $10^9 = 1$ billion people (that might be evil-doers)
- Everyone goes to a hotel 1 in $10^2 = 100$ days
- Hotels hold $10^2 = 100$ people, so there are $10^5 = 100{,}000$ hotels
- We examine hotel records for $10^3 = 1000$ days
- We look for people who, on two different days, go to the same hotel
- How many such pairs can we expect to find if there are no evil-doers?

10^9 people, 10^{-2}prob of hotel, 10^2people/hotel, 10^5 hotels, 10^3 days

- Prob any two people visit *a* hotel on given day: $(10^{-2})^2 = 10^{-4}$
- Same hotel: $\frac{10^{-4}}{10^5} = 10^{-9}$
- Prob visit same hotel on two different given days: $(10^{-9})^2 = 10^{-18}$
- # pairs of people: $\binom{10^9}{2} \approx 5\times10^{17}$
- # pairs of days: $\binom{10^3}{2} \approx 5\times10^5$
- Expected # of events that look like evil-doing:
  - (# pairs of people) x (# pairs of days) x (prob a pair of people visit same hotel on 2 given days)
  - $\approx (5\times10^{17})\times(5\times10^5)\times(10^{-18}) = 250{,}000$

Power Laws

- **Many phenomena relate two variables by a "power law":**
  - Linear relationship between the logarithms of the variables
  - E.g. $log_{10}y = 6 - 2\ log_{10}x$

- **General form:** $\log y = b + a \log x$
  - Thus $y = e^b e^{a\ logx}\ = e^b x^a = cx^a$

The Matthew Effect, i.e., The rich get richer

- Occurs when having a high value of some property causes that property to increase

- **Example**
  - Webpage has many links to it
  - This increases traffic to the page
  - More people decide to link to it

- Leads to power laws with $|a|>1$

Things that obey power laws

- **Node degrees in Web graph:** $a \approx 2.1$
- **Sales of products:** let y be the # of sales of x^{th} most popular book on Amazon
- **Sizes of website:** order sites by # of pages, y the # of pages at the x^{th} site
- **Zipf's Law:** $y =$ # times x^{th} most frequent word appears
  - $y = cx^{-1/2}$
  - Many other kinds of data obey this, e.g., populations of US states