STATISTICAL LEARNING & ASSOCIATION RULES MINING

SETTING THE STAGE

“Data science does not replace statistical modeling and data analysis; it augments them.”

(P. Boily)

“Data is not information, information is not knowledge,
knowledge is not understanding, understanding is not wisdom.”

(attributed to Cliff Stoll in Keeler's *Nothing to Hide: Privacy in the 21st Century*, 2006)

WHAT IS DATA SCIENCE? (REPRISE)

Data Science (DS) is the collection of processes by which we extract useful and **actionable insights** from data.

(paraphrased from T. Kwartler)

DS is the **working intersection** of statistics, engineering, computer science, domain expertise, and “hacking.” It involves two main thrusts: **analytics** (counting things) and **inventing new techniques** to draw insights from data.

(paraphrased from H. Mason)

LEARNING IN GENERAL

Beyond "just taking a quick look," humans learn through:

- answering questions
- testing hypotheses
- creating concepts
- making predictions
- creating categories and classifying objects
- grouping objects

The central Data Science/Machine Learning problem is:

can we design algorithms that can learn?

TYPES OF LEARNING

Supervised Learning (learning with a teacher)

- classification, regression, rankings, recommendations
- uses **labeled training data** (student gives an answer to each test question based on what they learned from worked-out examples)
- performance is evaluated using **testing data** (teacher provides the correct answers)

Unsupervised Learning (grouping similar exercises together as a study aid)

- clustering, association rules discovery, link profiling, anomaly detection
- uses **unlabeled** observations (teacher is not involved)
- accuracy **cannot** be evaluated (students might not end up with the same groupings)

TYPES OF LEARNING

Semi-Supervised Learning (teacher providing worked-out examples **and** a list of unsolved problems)

Reinforcement Learning (embarking on a Ph.D. with an advisor)

In **supervised learning**, there's a target against which to train the model. In **unsupervised learning**, we don't know what the target is, or even if there is one.

The distinction is **crucial**. Make sure you understand it.

[Jensen, A.B., et al. [2014], Temporal disease trajectories condensed from population-wide registry data covering 6.2 million patients, *Nature Communications*]

CASE STUDY: DANISH MEDICAL STUDY

The *Danish National Patient Registry* contains **68 million** health observations on **6.2 million** patients over a 15 year time span (Jan '96 – Nov '10).

Objectives:

- finding connections between different diagnoses
- determining how a diagnosis at some point in time might allow for the prediction of another diagnosis at a later point in time

METHODOLOGY

1. Compute **strength of correlation** for pairs of diagnoses over a 5 year interval on a representative subset of the data
2. Test diagnoses pairs for **directionality** (one diagnosis repeatedly occurring before the other)
3. Determine reasonable diagnosis trajectories (**thoroughfares**) by combining smaller frequent trajectories with overlapping diagnoses
4. Validate the trajectories by comparison with **non-Danish** data
5. Cluster the thoroughfares to identify central medical conditions (**key diagnoses**) around which disease progression is organized

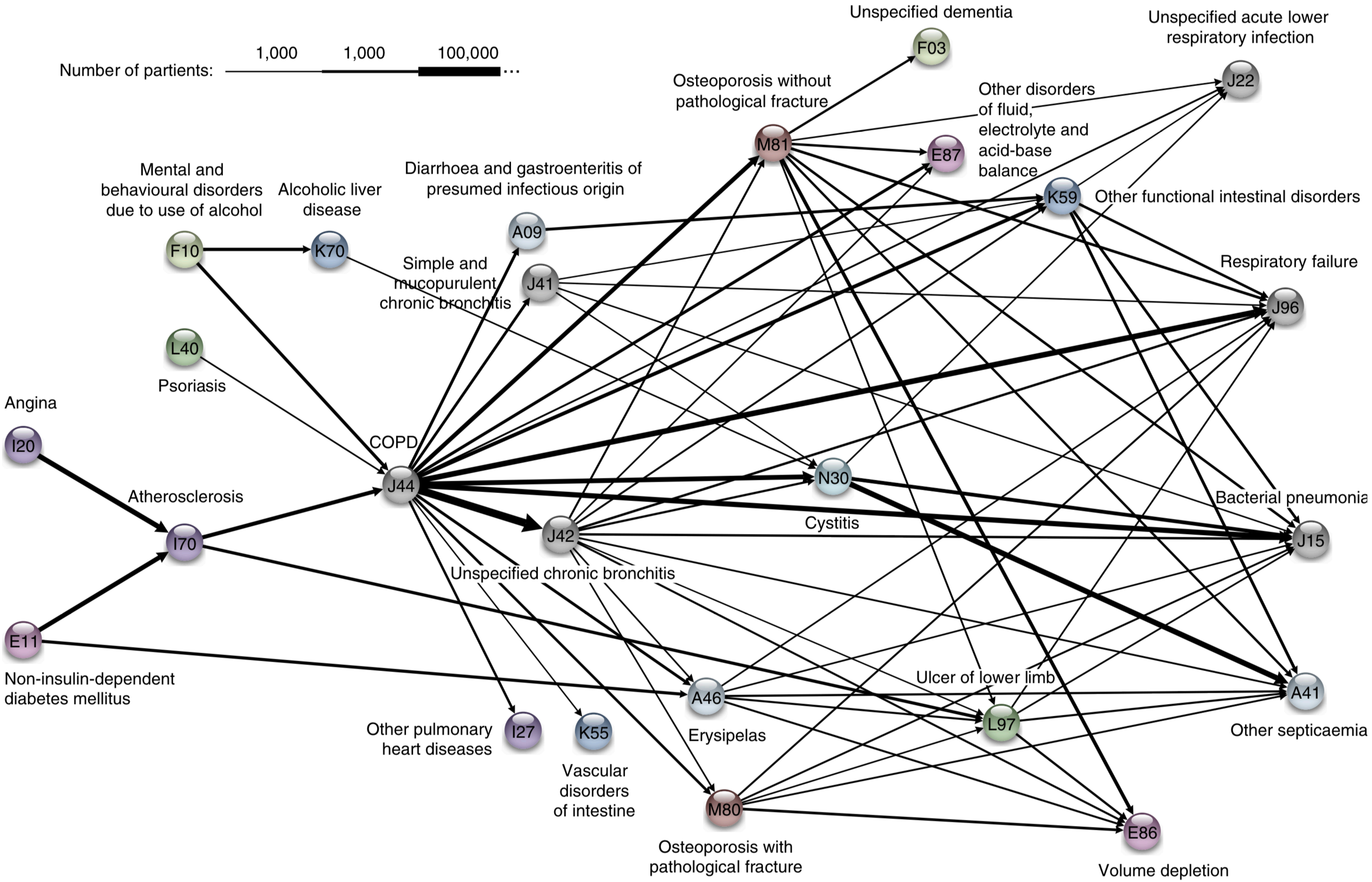
RESULTS

Data was reduced to 1,171 thoroughfares on the course of

- diabetes
- chronic obstructive pulmonary disease (COPD)
- cancer
- arthritis
- cardiovascular disease.

The data analysis showed, for example:

- diagnoses of anemia followed later by the discovery of colon cancer
- gout was identified as a step toward cardiovascular disease.
- COPD is **under-diagnosed** and **under-treated**.


Number of partients:
1,000
1,000
100,000
Unspecified dementia
F03
Unspecified acute lower respiratory infection
J22
Osteoporosis without pathological fracture
M81
Other disorders of fluid, electrolyte and acid-base balance
E87
Mental and behavioural disorders due to use of alcohol
F10
Alcoholic liver disease
K70
Diarrhoea and gastroenteritis of presumed infectious origin
A09
K59
Other functional intestinal disorders
Simple and mucopurulent chronic bronchitis
J41
Respiratory failure
J96
L40
Psoriasis
Angina
I20
COPD
J44
N30
Atherosclerosis
I70
Bacterial pneumonia
J15
Cystitis
J42
Unspecified chronic bronchitis
E11
Non-insulin-dependent diabetes mellitus
Ulcer of lower limb
A46
L97
A41
Other septicaemia
Erysipelas
Other pulmonary heart diseases
I27
K55
Vascular disorders of intestine
M80
Osteoporosis with pathological fracture
E86
Volume depletion

ASSOCIATION RULES BASICS

Association Rule Discovery is a type of unsupervised learning that finds connections among attributes (and combinations of attributes).

Example: we might analyze a dataset on the physical activities and purchasing habits of North Americans and discover that

- *runners who are also triathletes* (the **premise**) tend to *drive Subarus, drink microbrews, and use smartphones* (the **conclusion**), or
- individuals who have purchased home gym equipment are unlikely to be using it 1 year later (to name some fictitious possibilities)

ORIGINAL APPLICATION

Supermarkets record the contents of shopping carts at check-outs to determine items which are frequently purchased together.

Examples:

- bread and milk are often purchased together, but that's not so interesting given how often they are purchased individually
- hot dogs and mustard are also often purchased as a pair, but more rarely purchased individually

A supermarket could then have a sale on hot dogs while raising the price on condiments.

OTHER APPLICATIONS

Related Concepts

- looking for pairs (triplets, etc) of words that represent a joint concept
- {Ottawa, Senators}, {Michelle, Obama}, {veni, vidi, vici}, etc.

Plagiarism

- looking for sentences that appear in various documents
- looking for documents that share sentences

Biomarkers

- diseases that are frequently associated with a set of biomarkers

CAUSATION AND CORRELATION

Association rules can automate hypothesis discovery, but one must remain **correlation-savvy** (which is less prevalent among data scientists than one would hope…).

If attributes A and B are shown to be correlated, there are (at least) 5 possibilities:

- A and B are correlated **entirely by chance** in this particular dataset
- A is a relabeling of B
- A causes B
- B causes A
- combinations of other attributes $C_1, \ldots, C_n$ (known or not) cause A & B

[E. Siegel, Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die]

CAUSATION AND CORRELATION

Insight	Organization
Pop-Tarts before a hurricane	Walmart
Higher crime, more Uber rides	Uber
Typing with proper capitalization indicates creditworthiness	A financial services startup company
Users of the Chrome and Firefox browsers make better employees	A human resources professional services firm, over employee data from Xerox and other firms
Men who skip breakfast get more coronary heart disease	Harvard University medical researchers
More engaged employees have fewer accidents	Shell
Smart people like curly fries	Researchers at the University of Cambridge and Microsoft Research
Female-named hurricanes are more deadly	University researchers
Higher status, less polite	Researchers examining Wikipedia behavior

DEFINITIONS

premise conclusion

A rule $X \to Y$ is a statement of the form “if X then Y” built from any logical combinations of a dataset attributes.

A rule **need not be true for all observations** in the dataset (i.e. rules are not necessarily 100% accurate).

In fact, sometimes the “best” rules could be those which are only accurate 10% of the time, as opposed to rules for which the accuracy is only 5% of the time, say.

As always, **it depends on the context**.

DEFINITIONS

To determine a rule’s strength, we compute rule metrics:

- **Support** (coverage) measures the frequency at which a rule occurs in a dataset. A low coverage value indicates that the rule rarely occurs (whether it is true or not).
- **Confidence** (accuracy) measures the reliability of the rule: how often does the conclusion occur in the data given that the premises have occurred. Rules with high confidence are “truer”.
- **Interest** measures the difference between its confidence and the relative frequency of its conclusion. Rules with high absolute interest are... well, more interesting.
- **Lift** measures the increase in the frequency of the conclusion due to the premises. In a rule with a high lift (> 1), the conclusion occurs more frequently than it would if it was independent of the premises.

FORMULAS

If N is the number of observations in the dataset:

- $\text{Support}(X \to Y) = \frac{\text{Freq}(X \cap Y)}{N} \in [0,1]$ ⇐ Proportion of instances where the premise and the conclusion occur together
- $\text{Confidence}(X \to Y) = P(Y|X) = \frac{\text{Freq}(X \cap Y)}{\text{Freq}(X)} \in [0,1]$ ⇐ Proportion of instances where the conclusion occurs when the premise occurs
- $\text{Interest}(X \to Y) = \text{Confidence}(X \to Y) - \frac{\text{Freq}(Y)}{N} \in [-1,1]$
- $\text{Lift}(X \to Y) = \frac{N^2 \cdot \text{Support}\,(X \to Y)}{\text{Freq}(X) \cdot \text{Freq}(Y)} \in (0, N^2]$

... ?!?

A SIMPLE EXAMPLE

Hypothetical music dataset containing data for $N = 15{,}356$ music lovers.

Candidate Rule (RM): "If an individual is born before 1976 (X), then they own a copy of at least one Beatles album, in some format (Y)".

Let's assume that

- $\text{Freq}(X) = 3888$ individuals were born before 1976
- $\text{Freq(Y)} = 9092$ individuals have a copy of at least one Beatles album
- $\text{Freq}(X \cap Y) = 2720$ individuals were born before 1976 and have a copy of at least one Beatles album

A SIMPLE EXAMPLE

$$1.2 \approx \frac{0.70}{0.56}$$

The 4 metrics are:

- $\text{Support}(RM) = \frac{2720}{15{,}356} \approx\ 18\%$ (RM occurs in 18% of the observations)
- $\text{Confidence}(RM) = \frac{2720}{3888} \approx 70\%$ (RM is true in 70% when born prior to 1976)
- $\text{Interest}(RM) = \frac{2720}{3888} - \frac{9092}{15356} \approx 0.11$ (RM is not very interesting)
- $\text{Lift}(RM) = \frac{15{,}356^2 \cdot 0.18}{3888 \cdot 9092} \approx 1.2$ (weak correlation between being born prior to 1976 and owning a copy of a Beatles' album)

Interpretation of the Lift: 70% of those born before 1976 own a copy, whereas 56% of those born after 1976 own a copy.

BRUTE FORCE ALGORITHM

1. Generate item sets (of size 1, 2, 3, 4, etc.)
   - e.g. {purchasing = Typical, membership = False, coupon = Yes}
2. Create rules from each item set
   - e.g. **IF** (purchasing = Typical AND membership = False) **THEN** coupon = Yes
3. Calculate the support, confidence, interest, lift for each rule
4. Retain only the rules with “high enough” coverage, accuracy, interest, and/or lift (or other metrics)
5. These rules are considered to be **true** for the dataset – they are **new knowledge derived from the data**

GENERATING RULES

An **item set** (or instances) is a list of attributes and values.

A set of **rules** can be created by adding ‘**IF … THEN**’ to each of the instances. As an example, from the instance set

{membership = True, age = Youth, purchasing = Typical}

we can create the rules

- **IF** (membership = True AND age = Youth) **THEN** purchasing = Typical
- **IF** membership = True **THEN** (age = Youth AND purchasing = Typical)
- **IF** ø **THEN** (membership = True AND age = Youth AND purchasing = Typical)
- etc.

Parentheses are usually dropped

NUMBER OF RULES

Consider an item set C with n members.

In a rule derived from C, each of the n members shows up either in the **premise** or in the **conclusion**, so there are 2^n such rules.

The rule where each member is part of the premise (and the conclusion is empty) is not allowed, thus $2^n - 1$ rules can be derived from C.

The # of rules increases exponentially when the # of features increases linearly.

That's not good.

VALIDATION

The brute force algorithm works relatively well for **small datasets** (small number of features).

For **Big(ger) Data**, it can be costly to generate rules in that fashion (especially when the number of attributes increases). How do we generate **promising** candidate rules, in general?

How **reliable** are association rules? What is the likelihood that they occur by **chance**? How **relevant** are they? Can they be generalized **outside** the dataset, or to **new** data?

NOTES

Since frequent rules correspond to instances that occur repeatedly in the dataset, algorithms that generate item sets often try to **maximize coverage**.

When **rare events** are more meaningful (such as detection of a rare disease), we need algorithms that can generate rare item sets. **This is not a trivial problem**.

A reminder, in spite of Tufte's rejoinder: **correlation is not causation**.

OTHER ALGORITHMS

Continuous vs. **Categorical**: continuous data has to be binned into categorical data in order for association rules to be meaningful. There's more than one way to do that.

Item sets are sometimes called **market baskets**.

Other algorithms:

AIS, SETM, Apriori, AprioriTid, AprioriHybrid, Eclat, PCY, Multistage, Multihash, etc.

APRIORI ALGORITM

Developed initially for transaction data

- every reasonable dataset can be transformed into a transaction dataset using dummy variables

Finds **frequent item sets** from which to build candidate rules

- instead of building rules from all possible item sets

Starts by identifying frequent individual items in the database and extends them to larger and larger item sets, assuming these are still found **frequently enough** in the dataset

- **bottom-up** approach, uses the downward closure property of support

APRIORI ALGORITM

Prunes candidates which have **infrequent sub-patterns**

- requires a support threshold
- that threshold has to be set sufficiently high to minimize the number of frequent item sets

If a 1-item set is not frequent, any 2-item set containing it is also infrequent, for instance.

The algorithm terminates when no further successful extensions are found.

STRENGTHS AND LIMITATIONS

Easy to implement, easily parallelized.

Apriori is **slow** and it requires frequent data set scans.

- possible solutions: **sampling** and **partitioning**

Not ideal at finding rules for **infrequent** or **rare** item sets.

Other algorithms have since displaced it (historical value):

- **Max-Miner** tries to identify frequent item sets without enumerating them; performs jumps in space instead of using bottom-up approach
- **Eclat** is faster and uses depth-first search, but requires extensive memory storage