Analysing Google Search Trends Data with Dynamic Bayesian Networks

Marco Scutari
scutari@bnlearn.com

Dalle Molle Institute for
Artificial Intelligence (IDSIA)

January 20, 2023

→ Google Trends and Infodemiology

Dynamic Bayesian Networks

Inferring Skin-Brain-Skin Connections

Conclusions

The Problem: Building Causal Models

Studying comorbidities for several diseases at an epidemiological level is hard because:

- they must all be monitored simultaneously;
- they must be monitored over large populations for the results to generalise;
- they must be monitored for long enough periods of time to build a longitudinal data set that can capture their evolution and interactions.

For these reasons, most comorbidity studies:

- only include 2–3 diseases, and
- are cross-sectional, or contain at most 2–3 time points.

This makes it very difficult to build Bayesian networks that can be used as causal models and that can describe complex feedback loops.

The Data: Google Trends

During the COVID 19 pandemic, Google has made available a large data set with (among other things) the search queries containing keywords identifying ~400 health conditions:

Google's COVID-19 Open-Data
https://github.com/GoogleCloudPlatform/covid-19-open-data

The data:

- span different countries (US, UK, Ireland, Australia, Singapore), sometimes drilling down into individual states and counties (US, Australia);
- span three years (2020, 2021 and 2022) with daily records;
- are normalised and aggregated by search keyword using the best of Google's NLP models;
- are normalised consistently into search frequencies by geographical area.

The Solution: Infodemiology

If we assume that:

- there is a non-negligible association between the frequency of online searches for specific diseases and the actual incidence of those diseases in physicians' diagnoses;
- restricting ourselves to searches performed on Google is not a significant limitation because of the prevalence of its use.

We can approach the study of comorbidities using infodemiology (short for "information epidemiology"): scanning Internet-generated data for user-contributed health-related content, with the ultimate goal of improving public health.

Here the Internet-generated data are Google's COVID-19 Open-Data, and our aim is to build a causal network linking skin diseases and mental illnesses using dynamic Bayesian networks.

- ✓ Google Trends and Infodemiology
- → Dynamic Bayesian Networks
- Inferring Skin-Brain-Skin Connections
- Conclusions

A Graph and a Probability Distribution

A Bayesian network (BN) [5] is defined by:

- a network structure, a directed acyclic graph $\mathcal{G}$ in which each node corresponds to a random variable X_i;
- a global probability distribution $\mathbf{X}$ with parameters Θ, which can be factorised into smaller local probability distributions according to the arcs present in $\mathcal{G}$.

The main role of the network structure is to express the conditional independence relationships among the variables in the model through graphical separation, thus specifying the factorisation of the global distribution:

$$\mathrm{P}(\mathbf{X}) = \prod_{i=1}^{N} \mathrm{P}(X_i \mid \Pi_{X_i}; \Theta_{X_i}) \quad \text{where} \quad \Pi_{X_i} = \{\text{parents of } X_i \text{ in } \mathcal{G}\}\,.$$

Bayesian Network Structure Learning

Learning a BN $\mathcal{B} = (\mathcal{G}, \Theta)$ from a data set $\mathcal{D}$ involves two steps:

$$\underbrace{\mathrm{P}(\mathcal{B} \mid \mathcal{D}) = \mathrm{P}(\mathcal{G}, \Theta \mid \mathcal{D})}_{\text{learning}} = \underbrace{\mathrm{P}(\mathcal{G} \mid \mathcal{D})}_{\text{structure learning}} \cdot \underbrace{\mathrm{P}(\Theta \mid \mathcal{G}, \mathcal{D})}_{\text{parameter learning}} .$$

Structure learning consists in finding the DAG with the best

$$\mathrm{P}(\mathcal{G} \mid \mathcal{D}) \propto \underbrace{\mathrm{P}(\mathcal{G})}_{\text{graph prior}} \cdot \underbrace{\mathrm{P}(\mathcal{D} \mid \mathcal{G})}_{\text{marginal likelihood}} = \mathrm{P}(\mathcal{G}) \int \mathrm{P}(\mathcal{D} \mid \mathcal{G}, \Theta)\, \mathrm{P}(\Theta \mid \mathcal{G})\, d\Theta$$

which is known as score-based learning [2]. The alternative, constraint-based learning, uses tests following Pearl's work on causality [6]:

$$\underbrace{X_i \perp\!\!\!\perp_P X_j \mid \mathbf{S}_{X_i, X_j}}_{\text{conditional independence}} \Longrightarrow \underbrace{X_i \perp\!\!\!\perp_G X_j \mid \mathbf{S}_{X_i, X_j}}_{\text{graphical separation}} .$$

Parameter learning consists in estimating the parameter sets $\Theta_{X_i} \mid \Pi_{X_i}$.

Dynamic Bayesian Networks

Dynamic BNs (DBNs) [3] combine classic BNs and Markov processes to model dynamic data in which each individual is measured repeatedly over time, such as longitudinal or panel data.

Assume we have one set $\mathbf{X}^{(t)}$ of random variables for each of $t = 1, \ldots, T$ time points. We can model it as a DBN with a Markov process of the form

$$\mathrm{P}\left(\mathbf{X}^{(0)}, \ldots, \mathbf{X}^{(T)}\right) = \mathrm{P}\left(\mathbf{X}^{(0)}\right) \prod_{t=1}^{T} \mathrm{P}\left(\mathbf{X}^{(t)} \mid \mathbf{X}^{(t-1)}\right).$$

where $\mathrm{P}(\mathbf{X}^{(0)})$ gives the initial state of the process and $\mathrm{P}(\mathbf{X}^{(t)} \mid \mathbf{X}^{(t-1)})$ defines the transition between times $t-1$ and t. When modelling $\mathbf{X}^{(t)}$, the nodes in $\mathbf{X}^{(t-1)}$ only appear in the conditioning; we take them to be essentially fixed and to have no free parameters, so we leave them as root nodes.

Dynamic Bayesian Networks

We can model this transition with a 2-time BN (2TBN) defined over $(\mathbf{X}^{(t-1)}, \mathbf{X}^{(t)})$, in which we naturally assume that any arc between a node in $t-1$ and a node in t must necessarily be directed towards the node in t following the arrow of time.

We may also want to assume that there are no arcs connecting two nodes in the same t or, in other words, no instantaneous dependencies.

We can then write the decomposition into local distributions

$$\mathrm{P}\left(\mathbf{X}^{(t)} \mid \mathbf{X}^{(t-1)}\right) = \prod_{i=1}^{N} \mathrm{P}\left(X_i^{(t)} \mid \Pi_{X_i^{(t)}}\right),$$

and we usually assume that the parameters associated with the local distributions do not change over time to make the process time-homogeneous.

Dynamic Bayesian Networks as Causal Models

- Granger Causality: $X_2 \to X_1$ if the predictions of the value of a variable X_1 based on its own past values and on the past values of X_2 are better than the predictions of X_1 based only on its own past values [1].
- Pearl Causality: a more general theory that builds on BNs to describe the semantics of causal reasoning including confounding, identifiability, interventions, counterfactuals, causal queries, etc. [4].

DBNs are structured like an auto-regressive time series, so Granger causality applies. They are also a specific instance of BNs, so Pearl's causality applies as well.

- ✓ Google Trends and Infodemiology
- ✓ Dynamic Bayesian Networks
- → Inferring Skin-Brain-Skin Connections
- Conclusions

The Network

The DBN includes the following skin diseases and mental illnesses:

- obesity (“OBE”)
- acne (“ACNE”)
- alcoholism (“ALC”)
- anxiety (“ANX”)
- asthma (“ASTH”)
- attention deficit hyperactivity disorder (“ADHD”)
- burn (“BURN”)
- depression (“DEP”)
- dermatitis (“DER”)
- erectile dysfunction (“ED’)
- sleep disorder (“SLD”)
- scar (“SCAR”).

We use search queries aggregated weekly from users in the US between 2020-03-02 and 2022-01-24 (100 weeks), which are the least impacted by missing data and are collected by state and county.

The overall sample size is 287900, making this a big-data problem.

The Network

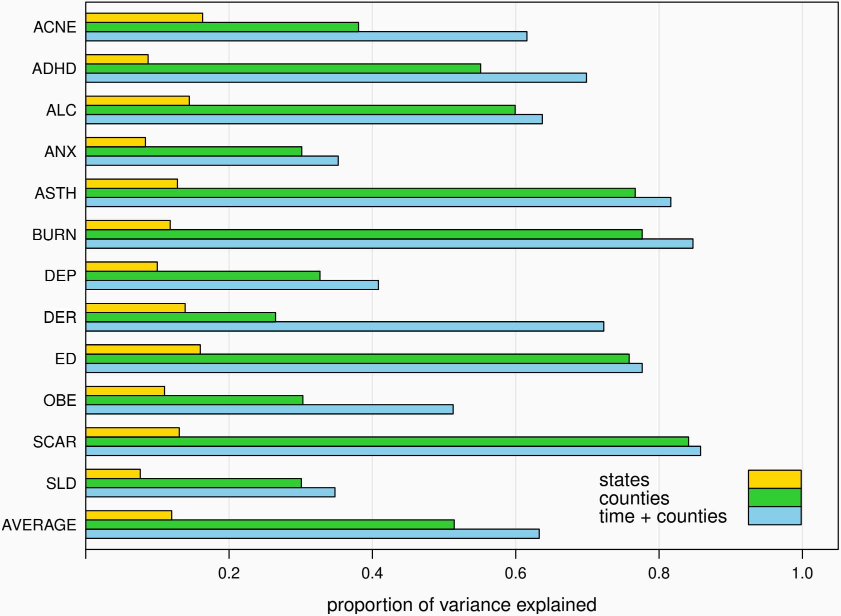
All variables are continuous, since they are normalised search query frequencies. Therefore, we model them with a Gaussian BN so that each local distribution $\mathrm{P}(X_i \mid \Pi_{X_i})$ takes the form

$$X_i^{(t)} = \mu_i^{(t)} + \Pi_{X_i^{(t)}} \boldsymbol{\beta}_i^{(t)} + \boldsymbol{\varepsilon}_i^{(t)} \qquad\qquad \boldsymbol{\varepsilon}_i^{(t)} \sim N(0, \sigma^2_{X_i^{(t)}}).$$

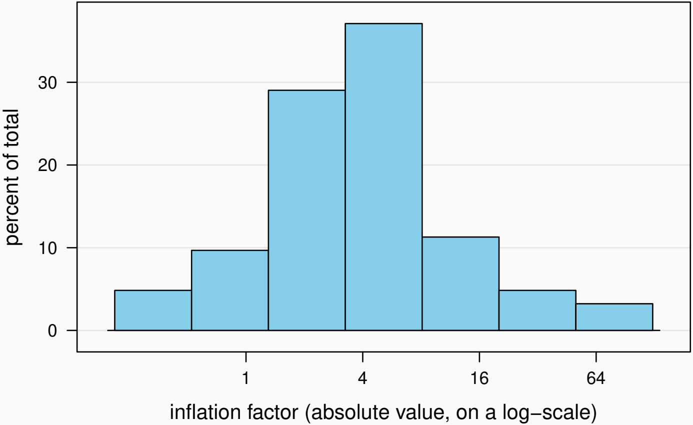
In practice, each $X_i^{(t)}$ represents a skin disease or mental condition in $\mathbf{X}^{(t)}$ and will depend on the $X_i^{(t-1)}$ representing the same skin disease or mental condition in $\mathbf{X}^{(t-1)}$ among other $\Pi_{X_i^{(t)}}$.

PROS: this accounts (in part) for different baseline frequencies in different states because $X_i^{(t)}$ and $X_i^{(t-1)}$ are paired variables and have the same baseline frequency for each observation.
CONS: frequencies are compositional data that are positive and bound to sum up to 1, a fact which we are disregarding in modelling them with a Gaussian distribution.

Temporal and Spatial Dependence Structure of the Data



We Cannot Just Use a Static Bayesian Network!



- Decorrelating the data to remove the space-time dependence produces a network that differs in 71% of the arcs from the original.
- 63% of the regression coefficients are inflated by a factor of 2 or more.
- The sign is different 29% of the coefficients.

Steps to Learn the Dynamic Bayesian Network

1. Rearrange the data to learn the 2TBN: keep the date and the location, create the $(t-1, t)$ pairs for all the 12 conditions.

```
reshaped = parLapply(cl, levels(data$county),
             function(each.county, data, symptoms) {

  county.data = subset(data, county == each.county)
  available.t0 = county.data$date[-length(county.data$date)]
  available.t1 = county.data$date[-1]

  t0 = subset(county.data, date %in% available.t0)[, symptoms]
  t1 = subset(county.data, date %in% available.t1)[, symptoms]
  names(t0) = paste0(symptoms, "_0")
  names(t1) = paste0(symptoms, "_1")

  return(cbind(county = each.county, date = available.t0, t0, t1))

}, data = data, symptoms = symptoms)

dyndata = do.call(rbind, reshaped)
dyndata$county = as.factor(dyndata$county)
```

Steps to Learn the Dynamic Bayesian Network

2. Set up the blacklist that prevents instantaneous arcs and arcs that go backwards in time from being learned, then use it with hill-climbing and BIC as a score to learn a Gaussian BN. The penalty for BIC is the standard one for the moment, more on that later...

```
t0.vars = grep("_0", names(dyndata), value = TRUE)
t1.vars = grep("_1", names(dyndata), value = TRUE)

learn.dbn = function(data, penalty = 1) {

  bl = rbind(tiers2blacklist(list(t0.vars, t1.vars)),
             set2blacklist(t0.vars), set2blacklist(t1.vars))
  hc(data, blacklist = bl, score = "bic-g", k = penalty * log(nrow(data) / 2))

}#LEARN.DBN
```

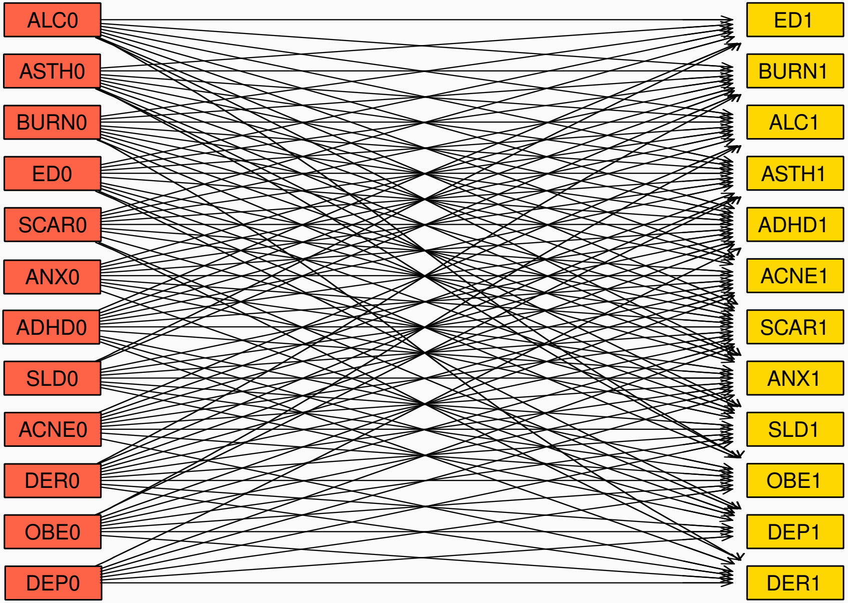
(Now this where I would normally conclude the talk with “Then we use model averaging to remove the noise in the learning process and we obtain an averaged network which is our final model”, but...)

Steps to Learn the Dynamic Bayesian Network

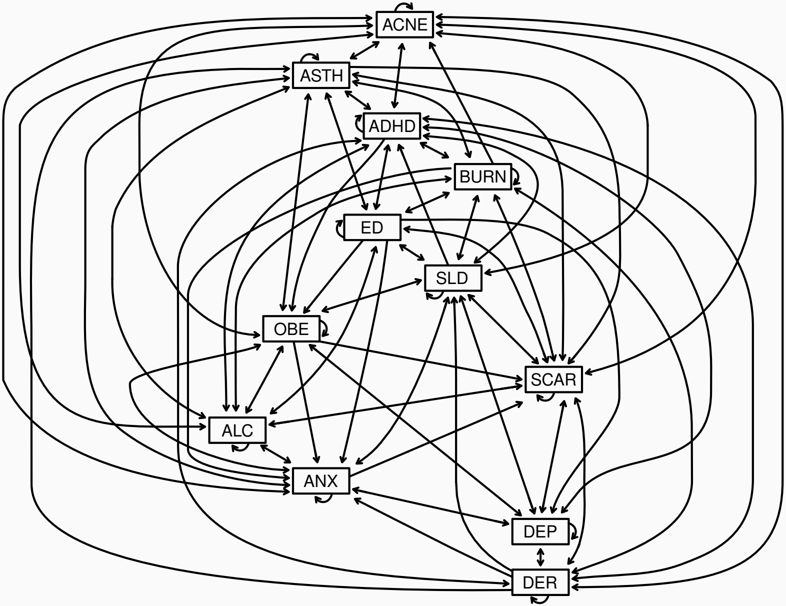
3. We are dealing with big-data: the amount of noise is small compared to the information in the data. Hence the averaged network is not any sparser than the individual networks being averaged.

```
bagging = function(i, data, counties, penalty = 1) {

  keep = sample(counties, 0.75 * nlevels(counties))
  boot.sample = data[counties %in% keep, ]
  boot.sample = boot.sample[, sample(ncol(data), ncol(data))]

  learn.dbn(boot.sample, penalty = penalty)

}#BAGGING
averaging = function(data, penalty = 1) {

  dags = parLapply(cl, seq(500), bagging, data = data[, c(t0.vars, t1.vars)],
                   counties = data$county, penalty = penalty)
  strength = custom.strength(dags, nodes = c(t0.vars, t1.vars))

  averaged.network(strength)

}#AVERAGING

avg.dag = averaging(dyndata, penalty = 1)
```

The 2-Time Bayesian Network



The Dynamic Bayesian Network

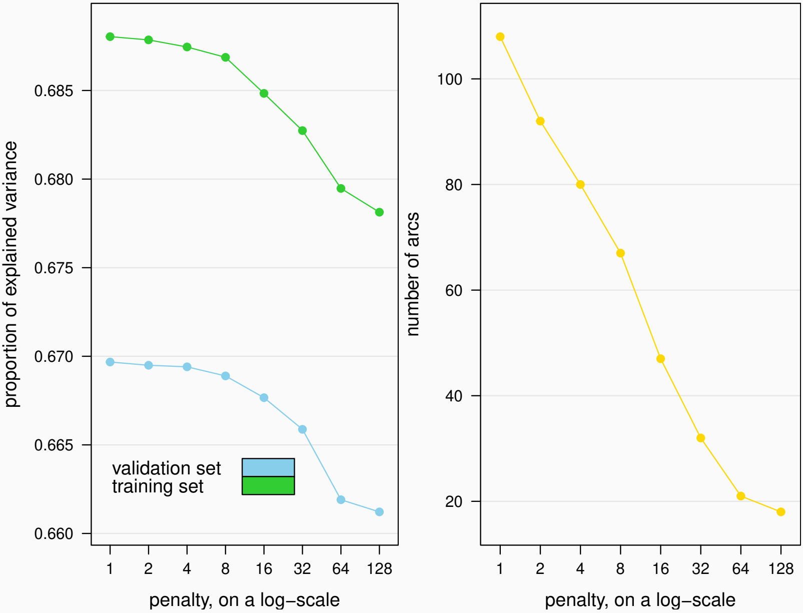


What Went Wrong?

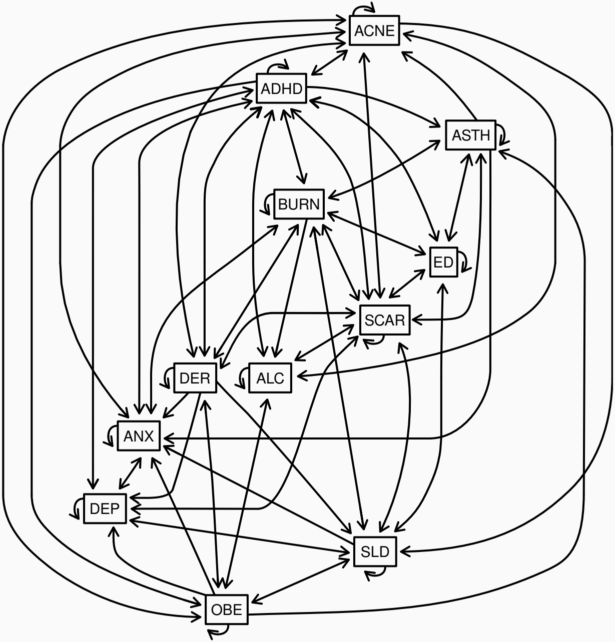
- The sample size is so large that even the tiniest effects are statistically significant, which results in dense networks where most nodes are connected to most other nodes.
- For the same reason, the amount noise in the data is relatively small and all the networks learned during model averaging include the same arcs. Therefore, model averaging has no arcs with low confidence to remove and the averaged network is also dense.
- Each condition is regressed against itself at the previous time point: a different baseline for the county then appears in both sides of the equation, and accounts for some of the spatial dependence between observations. Not all of it, however: we would need a spatial correlation matrix between counties for that.

Things we can do: increase the penalty of BIC to make it drop the arcs with the smallest effects, reduce the size of the bootstrap samples and permute their columns to make learning noisier.

Making the Dynamic Bayesian Network Sparser



The Final Dynamic Bayesian Network



- ✓ Google Trends and Infodemiology
- ✓ Dynamic Bayesian Networks
- ✓ Inferring Skin-Brain-Skin Connections
- → Conclusions

Epidemiological Insights

DBNs provide a causal model of multiple conditions with longitudinal measurements, allowing feedback loops to reproduce the natural cycle of human health.

- The DBN confirms the interplay between skin diseases and mental illnesses, including well-known clinical relationships, and puts them into a larger context.
- The large number of feedback loops supports the existence of vicious circles in which diseases exacerbate each other until treated appropriately, even though the DBN does not show the starting point of these circles.
- The DBN highlights key mediators, like sleep disorders, that establish a bridge between the skin and the brain.
- Not controlling for important comorbidities like obesity may lead to spurious conclusions, hiding the true relationships.

Conclusions

- Dynamic BNs allow us to differentiate between feedback loops and unidirectional causal effects in a rigorous way through Pearl and Granger causality frameworks.
- Sparsity in a BN learned from big data is about interpretability more than denoising: there are many causal effects that are statistically significant but are too small to be relevant in practical applications. We want to remove them to make the BN more readable.
- Completely modelling spatial dependence in big data is challenging because the spatial correlation matrix scales quadratically in the sample size.
- Infodemiology can give valuable insights when the structure of the data is taken into account.

Thanks!

Any questions?

References I

- C. W. J. Granger.
  Some Recent Development in a Concept of Causality.
  Journal of Econometrics, 39(1–2):199–211, 1988.

- D. Heckerman and D. Geiger.
  Learning Bayesian Networks: a Unification for Discrete and Gaussian Domains.
  In *UAI*, pages 274–284, 1995.

- K. Murphy.
  Dynamic Bayesian Networks: Representation, Inference and Learning.
  PhD thesis, UC Berkeley, Computer Science Division, 2002.

- J. Pearl and D. Mackenzie.
  The Book of Why: the New Science of Cause and Effect.
  Basic Books, 2018.

- M. Scutari and J.-B. Denis.
  Bayesian Networks with Examples in R.
  Chapman & Hall, 2nd edition, 2021.

- T. S. Verma and J. Pearl.
  Equivalence and Synthesis of Causal Models.
  In *UAI*, pages 255–268, 1990.