Building Probabilistic Causal Models using Collective Intelligence

21 Sept 2021

Olav Laudy, Alexander Denev, Allen Ginsberg

Olav Laudy is Chief Data Scientist at Causality Link

olav.laudy@causalitylink.com

9217 S 1300 E, Sandy, UT 84094

(801) 601-1053

Alexander Denev

alexander_denev@hotmail.com

Allen Ginsberg is head of NLP at Causality Link

allen.ginsberg@causalitylink.com

9217 S 1300 E, Sandy, UT 84094

(801) 601-1053

Building Probabilistic Causal Models using Collective Intelligence

25 Oct 2021

Abstract

The purpose of this paper is to show a novel approach to automatically generating Probabilistic Causal Models (Bayesian Networks (BN)) by applying Natural Language Processing (NLP) techniques to a corpus of millions of digitally published news articles in which views by different authors are expressed on the future states of economic and financial variables, and geopolitical events. The BNs that the authors will show how to derive will represent the wisdom-of-the-crowds forward-looking point-in-time views on various variables of interest and their dependencies. These BNs are likely to be of interest to asset managers and to economists who want to gain a better understanding of the current drivers of an economy based upon a rigorous probabilistic methodology. Additionally, in an asset allocation context, the BNs the authors derive can be fed to an optimization engine to construct a forward-looking optimal portfolio given the constraints of the asset manager (e.g., budget, short constraints etc.). The authors demonstrate various automatically derived BNs in a financial context.

THREE KEY TAKEAWAYS:

1- The authors show a novel approach to automatically generating Probabilistic Causal Models (Bayesian Networks (BN)) by applying Natural Language Processing (NLP) techniques to a corpus of millions of digitally published news articles.
2- They not only derive the DAG from the text, but also automatically derive the conditional probability; the latter is the main contribution of the paper
3- The BNs are likely to be of interest to asset managers and to economists who want to gain a better understanding of the current drivers of an economy based upon a rigorous probabilistic methodology

Keywords: NLP, causal graphs, Bayesian Networks, macro-finance, expert elicitation, knowledge mining.

Bayesian Networks[1] (BNs) have proven to be a very useful tool in practice in many domains[2] because 1) they represent causal knowledge about how different variables interact and 2) they are deeply rooted in probability theory and hence well-suited to express uncertainty. In other words, BNs, while still firmly grounded in the statistical realm, allow for the incorporation of knowledge of "why" things happen i.e., what causes what. For example, we know that a crash in stock markets almost surely will cause a spike in implied volatilities, but the opposite is not necessarily true. Hence, BNs incorporate causation as a primary concept and correlation as a derived one. To use Pearl's (Pearl (2009)) terminology, casting a treatment of a problem in terms of causation will make our judgements about (conditional) (in)dependence between variables 'robust'; will make them well suited to represent and respond to changes in the external environment; will allow us to work with conceptual tools which are more 'stable' than probabilities; will permit extrapolation to situations or combination of events that have not occurred in history.

In Rebonato (2010), BNs are applied to building stress scenarios based on causal reasoning when historical data about stress events (e.g., political events, unusual monetary policy etc.) is insufficient or absent. The BNs are built based on domain knowledge of which variables to include (e.g., are there any omitted variables?) and domain knowledge about the structure of the BN i.e., how the modelled variables interact with each other and their conditional independence properties. The parameters (probabilities) of the BN are derived from expert judgement, cross-sectional market data and other forward-looking data sources, and historical data when deemed relevant. This is of significance, for example, to asset managers when they aim to obtain an optimized portfolio in which forward-looking views are expressed, especially for events that have not been observed in the past.

As Denev et al (2016) point out, building a BN where more than two variables are at play suffers from three sources of uncertainty, namely 1) uncertainty with respect to which variables to include in the BN 2) uncertainty with respect to the skeleton of the BN on which two experts can have very different opinions, and 3) what data to use to infer the parameters of the BN[3]. Resolving the three sources of uncertainty can sometimes be arduous even for just a few variables. A combination of sensitivity analysis, trial-and-error process and domain expertise are needed[4]. Moreover, usually many experts with different backgrounds (e.g., one for each variable of interest) are required to come up with a credible model, usually through brainstorming sessions. This can a very time-consuming process. As a result, the model could even be obsolete before it is completed since economies could evolve rapidly, especially under stress. This is why the uptake of BNs in financial and economic modelling has been limited.

In this paper, we will show an automatic way to partially, possibly sometimes fully, resolve some of the 3 uncertainties and reduce the BN build time to a matter of seconds. We will leverage a platform where millions of news articles by different authors are stored daily. The philosophy behind the platform is to try to mine knowledge from the broad expanse of human

[1] For an introduction to Bayesian Networks please see Koller (2009), Pearl (2009).
[2] For example, medical diagnosis, reliability engineering, software fault prediction and linguistics. See Koller (2009) for numerous examples. For examples in the financial context, see Denev (2015).

[3] For example, if historical data is used, what is the source of this data (many macroeconomic sources could have slightly different information) and what is the time span used to derive the distribution of the modelled variables e.g., last year of data, two years etc.
[4] The latter is usually considered to be very subjective and prone to human bias

intelligence easily accessible in the world today. Humans are very good at sensing, understanding, and describing causal relations. However, a single human brain is not always capable of ingesting the full picture. Sometimes getting the right direction of causal influence between two variables requires a large number of observations and differing perspectives. By using an aggregating platform, however, such knowledge is more likely to be obtained by grouping together in 'the collective brain' many points of view concerning the causal structure of the domain of interest at a given point in time.

We must say that there exists a body of literature (Sanchez et al, 2004; Moghimifar et al, 2020) that deals with automatically extracting BNs skeletons from text. However, we have not found literature that also showed how to automatically populate BNs with probabilities. We consider the latter one of the main contributions of this paper.

The paper is structured as follows. In Section 'Data' we will describe the platform from which we will extract the data. How to transform the causal statements from the platform into a Directed Acyclic Graph (DAG) – the skeleton of a BN – will be shown in Section 'Deriving the DAG'. We will show how to populate the DAG with probabilities in Section 'Populating the DAG' together with an optimization procedure in case the extracted probabilities are not bona-fide and they do not yield a well-defined Joint Probability Table (JPT). We will discuss some practical examples in Section 'Results'. We will conclude in Section 'Conclusions and further work'.

DATA

Getting a computer to understand natural language in the same way a human speaker does has been the "holy grail" of Artificial Intelligence from its very inception. Over the years, researchers have scaled back the ambitious goal of achieving "understanding" in ways that have now generated the subfield of "Natural Language Processing[5]" or NLP. NLP essentially relies on statistical linguistic data (which is easily gathered these days) to perform many tasks of interest in many domains, e.g., construct language models which can be used to, for example, predict the next word in an input word sequence. An important NLP task is "information extraction" which involves ways of generating structured information, i.e., machine-readable data, from ordinary natural language text[6]. Siphoning out the kind of data needed to generate formal representations of causal knowledge contained in news articles or reports is an example of this kind of task.

For the purposes of this paper, we employ data from a proprietary platform that automatically analyses news articles to formalize causal links expressed in natural language about the forces acting in the geopolitical, financial, and economic world[7]. The platform ingests about 50,000 texts per day in 24 languages produced by news services such as the Boston Globe, the Washington Post or Agence France Press. Over the past 6 years, this has resulted in a collection of around 95M texts. These texts are passed through a proprietary

[5] To get a point of view on why natural language *understanding* is so difficult yet seemingly feasible according to many researchers see (Ginsberg 2006).
[6] The success of programs such as IBM's Watson and its descendant applications (see Beller,et. Al, 2016) is largely dependent on various forms of this capability.
[7] https://causalitylink.com/

Natural Language Processing (NLP) set of modules that aggregates the resulting structured data in an AWS-powered data lake accessible via SQL queries.

Four types of data types are extracted by the NLP modules and made available in the data lake: indicators, trends, events, and causal links.

Indicators represent all numeric data mentioned in financial texts e.g., the USA GDP for a certain quarter, the sales of Tesla Model 3 in China for a specific month, the EBITDA of Ford for a specific quarter. With the goal to be able to represent a universal data model in a single data structure, each indicator is composed of a Key Performance Indicator (KPI) (such as GDP or EBITDA, or ESG measures), and a set of descriptors characterizing the potential country, industry, company, and product mentioned in the context of this KPI. The set of KPIs and descriptors are part of a large ontology (explicit formal specifications of the terms in the domain and relations among them) that has been specifically developed for the application[8]. The ontology also provides a set of coherence checks between KPIs and descriptors that are an essential part of the NLP system. For example, GDP only applies to countries, while EBITDA only applies to companies. Impossible indicators such as "the GDP of Exxon-Mobil" or "the EBITDA of Finland" are avoided through these checks.

The NLP system also recognizes when statements mention indicators in the past or present tense as opposed to the future tense, which is essential to enable detailed comparisons of the future vs the known. And when possible, it extracts the span of the indicator, which is the duration of the measure of that indicator. Duration is expressed in days. The value 0 indicates instant measures such as stock-price; 7 is used for weeks for week-to-week comparisons; 28 to 31 for months, used for example for car sales; 90 to 92 for quarters, which is the most common span for financial measurements of public companies; 365 to 366 for years, etc.

Trends represent the evolution of indicators over a specific period, which is the span of the trend. The span of the trend is not to be confused with the span of the underlying indicator. For example, one can compare quarterly numbers (Q1 for example) year-over-year, where the span of the indicator is 92 and the span of the trend is 365. A trend is associated with a date and an offset. The date refers to the time when the article that contained the trend sentence was published. The offset refers to the time that was spoken about in the context of the trend. For example, a reference made to the last quarter will be associated with an offset of -90. Statements made about the future such as '…is expected to close the year up by 10%' is associated with a positive offset representing the number of days from the publish date till year end.

Events[9] usually represent non-recurring activities, such as specific mergers and acquisitions, natural disasters such as hurricanes and earthquakes, man-made disasters such as wars and industrial events, etc. Events are more complex than indicators in that they can relate

[8] For some early work on the use of ontologies in an information retrieval and automatic indexing application see (Ginsberg 1993). The blackboard architecture at the core of the NLP system leverages the ontology to enable the recognition of potentially trillions of indicators that can be represented as combinations of over 1,700 KPIs complemented as necessary by one of all the countries in the world, one of 40,000 public companies, one of 1,600 industries, and one of a rapidly expanding number of products from all these companies.

[9] We will not be using Events for this paper

multiple entities: an acquisition will involve two companies, a war at least two countries, etc. Events are recognized through the equivalent of a KPI which what is called the baseEvent, and several descriptors that complement the context of the baseEvent. The system currently recognizes over 700 different baseEvents. Events have a start date and an end date, which can sometimes be in the distant future.

Causal links represent causal statements made by authors linking a trend, an indicator or an event to another trend or another indicator. Statements such as "the increased strength of the US dollar led to a decline of the price of commodities" or "every 5% increase in sales of the Ford F150 translates into a 10% increase of Ford's profits" are recognized by the system and transformed into causal links between indicators with a direction, a strength and a positive or negative correlation sign.

The automatic extraction of causal statements from natural language texts is a nascent field, and the system has an inbuilt bias for precision vs coverage, that is, it only extracts causal statements from well-defined forms of statements such as "A is due to B" or "A has led to B" rather than by detecting proximity of A and B in a sentence. The complexity of the causal forms that are extracted from texts is, however, growing over time. At this point, "A and B led to C" is not considered as a valid form. First, such statements are very rare in text. Second, such expressions are less precise than the two-variable form. The form "A leads to B and C" is safely transformed into two causal links "A leads to B" and "A leads to C". One can view "a X% growth of A leads to a Y% growth of B" as equivalent to "Y/X is the elasticity of B vs A".

The above-mentioned structure of the indicators has in part been defined to enable a natural generalization of causal links using the taxonomy tree of the descriptors as specified by the ontology. For example, it is easy to aggregate individual causal statements between the "sales of the different models of Tesla cars in China" and "Tesla Profit" into a single causal pair (generalizing along the model descriptor) between the sales of Model Y and Model 3 of Tesla in China and its worldwide profits. Or the sales of Tesla worldwide (this time generalizing along the geography of the sales of Tesla) and its profits.

We note that perhaps the most straightforward way to derive a relation between the variables from text is to count their co-occurrence in a sentence. However, this co-occurrence does not imply causation. For example, a sentence "we think that inflation and gold will rise this year" does not indicate a causal relationship. The NLP used in the platform requires an explicit human expression of causality to take the co-occurrence into account. The sentence "higher inflation tends to pressure gold's price higher" is a valid example for the (causal) relation of inflation impacting gold. The sentence "higher inflation does not tend to pressure gold's price higher" is not counted and finally, a sentence such as "a higher gold price tends to push inflation higher" is counted towards the causal relationship of gold impacting inflation (and is not counted for the reverse relationship).

DERIVING THE DAG.

The individual causal statements are stored in the platform as the causal pair (source –> target), with explicit causal directionality from source to target. This explicit causal directionality implies that the causal pair ('A' -> 'B') is stored separately from the causal pair ('B' -> 'A').

An aggregation of the individual causal statements into a graph structure is required for the causal links to be used in a BN. A BN is expressed via a Directed Acyclic Graph (DAG). The nodes of this graph represent the economic indicators, such as 'USA GDP', and the links represent the causal relations found between the nodes from the news articles. For a DAG to be valid, it is required that there be no loops in the network structure.

The collection of causal statements we extract from the news articles does not necessarily adhere to the acyclic structure, i.e., (different) authors can express both the causal pair ('A'–>'B') as well as ('B'–>'A'). For example, in the sentence "Due to the demand of the popular F-150 truck, Ford had a much larger output in that segment this quarter", shows the (time-stamped) causal pair ('Ford F150 demand'->'Ford pickup-trucks production'), or shorter: ('Ford demand'->'Ford production'). Another author writes how "the demand will be driven by new EV facility that Ford opened", which translates to (Ford production->Ford demand). Furthermore, since the NLP captures statements about causal pairs, nothing prevents a loop in a larger structure of nodes. For example, we can have causal pairs ('A'->'B'), ('B'->'C') to start with. The addition of a causal pair like 'C'->'A' would generate a loop structure and would thus invalidate the DAG. An example would be ('USA GDP' -> 'USA interest rates'), ('USA interest rates' -> 'DXY) and ('DXY -> 'USA GDP').

It is natural to aggregate the individual causal statements to (driver, target) pair and use the count of the pair as a weight. Furthermore, this weight can be subject to a maximum time span and a decaying importance of older causal statements to adapt the BN to more current reasoning.

In what follows, we describe a step-by-step method to derive a DAG structure from the individual causal pairs.

First, low frequency causal statements are discarded. Such statements may either arise from an NLP error or single contrarian opinion. For this paper, a threshold value of 10 is chosen. That is: for a causal pair to be taken into consideration, there need to be 10 sentences across the whole corpus validly asserting that causal pair.

A collection of core nodes is selected as primary focus. These could be the factors on which a portfolio depends (e.g., rates, inflation, equity indices etc.). For each core node, the top n drivers by frequency are collected. The distinct nodes resulting from this query form the collection of nodes of the DAG.

Next, all available causal pairs between the nodes are collected. That is: if in the node query above 'A' was a core node and 'B' was found to be a driver of 'A', we collect both the causal pairs ('A'->'B') and ('B'->'A') along with their weights (given that such pair exist in the data). An iterative approach adds all the links to the graph by decreasing order of weight. Every addition is tested to result in a DAG. If the result is no longer a DAG, that causal pair is not considered further in the automatic analysis. However, the decision can be overridden by an expert using a Bayesian graph editor thereby ensuring the graph is a DAG by removing other elements based on economic reasoning.

This heuristic results in choosing the higher frequency link as the 'winning' link for the pair ('A'->'B') and ('B'->'A'), unless cancelled as earlier addition to the graph. For larger graphs, this procedure cancels the weakest link of a clique, as this is the lowest frequency pair that is added to activate the non-DAG alarm.

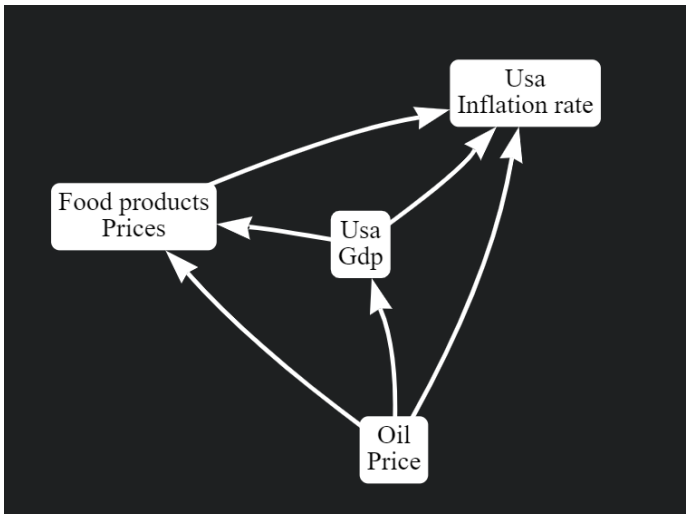
It may be instructive to understand what the drivers are through a concrete example. Table 1 shows the top 3 drivers of the USA inflation rate together with their frequencies as extracted from the platform.

Table 1 Top 3 drivers for the USA inflation rate

driver	target	count
Oil prices	USA inflation rate	1,922
Food products prices	USA inflation rate	1,091
USA GDP	USA inflation rate	663

In our approach to generate a DAG, we subsequently collect all available causal pairs. We find 12 causal pairs (which is also the maximum number of pairs for 4 nodes. This is due to the nature to the tight inter-relation of macro-economic concepts). With the iterative approach explained above applied, we end up with a DAG with 6 links. The graph is displayed in Figure 1. 'Oil price' drives 'USA inflation rate' directly, via 'Food products prices' and via 'USA GDP', with 'USA GDP' being a driver for 'Food products prices' as well.

Figure 1: The DAG resulting from the top 3 drivers of USA inflation rate



The next issue is how to automatically populate the DAG with probabilities. We discuss this in the next section.

POPULATING THE DAG

So far, we have derived the structure of the graph from the text, but not yet populated the BN with probabilities. In what follows, we will limit ourselves to Boolean BNs only with two states of the variables (Down, Up). This is not a limitation of BNs. In fact, BNs also allows incorporation of continuous nodes. We leave this for future research.

A BN is supplied with two types of probabilities: probabilities for root nodes (marginals) and conditional probabilities for any node with incoming links. The latter can be of any order, single (one parent), double (two parents), etc.

We describe the population of the probabilities for the following cases:

- Root marginals
- Single conditional probabilities
- Of greater order

Probabilities of the root nodes

The root nodes of a BN represent the independent inputs to the network. Probabilities of those root nodes are derived from forward-looking trends statements from the texts. For example, the sentence "Tesla's CEO Elon Musk said that the company's vehicle deliveries in 2020 would be up 30% to 40% from last year" is captured as a forward looking (up) trend on 'Tesla revenue'. As explained in Section 'Data', every extracted trend is associated with an offset, representing the time horizon of the prediction. In the following, the root node probabilities are based on predictions with any time horizon. The use of forward-looking statements with constraints on the time horizon is straightforward. Currently, all forward looking statements made in the last 90 days are selected and weighted based on an exponential decay with a half-life of 14 to give a larger weight to the more recent forward-looking trend statements. Note that the window and half-life parameters of this selection are subjective, and the current values are chosen based on a trade-off between stability and recency[10]. Subsequently, the trends are aggregated into a probability by computing the proportion of uptrends out of all forward-looking trends. This probability is also referred to as future Positive Trend Probability, or PTP_f. Table 2 shows such numbers at the time of writing: both 'Oil price' and 'USA inflation rate' will trend up according to the aggregated statements.

Table 2 Probabilities for the possible root nodes as of March 4 2021

Cause	**P(Cause) or PTP_f**
Oil price	.73
USA inflation rate	.80

Bivariate marginal and single conditional probabilities

Let us start with an example of a network with two nodes only. Let those be the price of oil ('Oil price') and the inflation rate in the USA ('USA inflation rate'). For the pairs ('Oil price' -> 'USA inflation rate') and ('USA inflation rate' -> 'Oil price'), we observe 1922 and 581 cases, respectively. In the procedure to generate a DAG, we would cancel the addition of ('USA inflation rate' -> 'Oil price') to the BN, as this would no longer make it a DAG. We present both cases to substantiate our logic.

In Tables 3 and 4, we see the collection of causal expressions in text broken out into how the authors express the direction of trends in both cause and target[11]. An example sentence of Table 3 where either trend direction is discussed as up-trend is: "We saw an uptick in inflation rate mainly due to the increased oil prices". In Table 4, the reverse relation is

[10] Sensitivity analyses can be performed to assess the robustness with respect to these parameters.
[11] We will refer to the quantities extracted from the platform as "observed"

discussed, where ‘USA inflation rate’ is the cause (or driver) of the ‘Oil price’. An example of a statement where both the trend of the cause and the target are down is: ‘’The inflation rate being down this quarter even further supressed the oil prices.” From the direct impact of ‘Oil price’ on the price of any oil derived product (i.e., plastic) and transportation costs, it may be economically more justified that oil drives inflation in a more direct way than the reverse relation. We find it thus acceptable and necessary in the automatic generation of this DAG to only consider the most frequently mentioned link and point out the ability for this be to overridden by an expert.

Table 3 The jointly observed counts and (probabilities) of trends and causal reasoning for the pair (‘Oil price’- > ‘USA inflation rate’)

		Target: USA inflation rate		
		Down	**Up**	
Cause: Oil price	**Down**	421 (*0.22*)	182 (*0.09*)	
	Up	264 (*0.14*)	1,055 (*0.55*)	**Total**
				1,922 (*1*)

Table 4 The jointly observed counts and (probabilities) of trends and causal reasoning for the pair (‘USA inflation rate’ -> ‘Oil price’)

		Target: Oil price		
		Down	**Up**	
Cause: USA inflation rate	**Down**	139 (*0.24*)	60 (*0.10*)	
	Up	39 (*0.07*)	343 (*0.59*)	**Total**
				581 (*1*)

We refer the interested reader to Appendix A, Table 16 which provides three example sentences for every combination of up and down for the driver-target pairs displayed in Table 1. The sentences have not been cherry-picked – and thus may be incorrectly classified due to NLP misinterpretation. This ‘honest’ view may give the reader an appreciation for the difficulty of the NLP problem and forewarn that the resulting BNs may be affected due to such NLP errors as not all sentences are curated by hand to be included or not in the estimation procedure.

It may be instructive to see both tables from the conditional probability perspective of the BN. In the simplest two node BN, we have that the joint probability of cause and effect factorizes as follows:

$$P(\text{cause},\text{effect}) = P(\text{cause})P(\text{effect}|\text{cause}) \quad (1)$$

The term P(effect|cause) which is calculated from Table 3 is shown in Table 5 as P(‘USA inflation rate’|’Oil price’). With oil prices up, it is more than twice as likely that the inflation rate is up compared to when the oil price is down. This confirms our intuition. i.e., it would be odd to suggest the reverse, namely, that oil prices being up lead to a low inflation rate but nevertheless there are authors who describe that relation, for example as ‘inflation slowdown despite the spike in gasoline prices’. Even for the reversed causal relation, shown in

Table 6, the conditional probabilities make sense: an inflation rate being up, will drive the oil prices up (oil is a commodity and thus will become more expensive relative to money).

Table 5 Conditional probability table for the pair ('Oil price' -> 'USA inflation rate')

		Target: USA inflation rate		
		Down	**Up**	
Cause: Oil price	**Down**	0.70	0.30	**1**
	Up	0.20	0.80	**1**

Table 6 Conditional probability table for the pair ('USA inflation rate' -> 'Oil price')

		Target: Oil price		
		Down	**Up**	
Cause: USA inflation rate	**Down**	0.70	0.30	**1**
	Up	0.10	0.90	**1**

At this point, we point out the following relation for between odds ratio and the probabilities in bivariate tables:

$$\text{Odds ratio (T,C)} = \frac{\mathrm{P}(\mathrm{C}=\mathrm{up},\mathrm{T}=\mathrm{up})}{\mathrm{P}(\mathrm{C}=\mathrm{up},\mathrm{T}=\mathrm{down})} \left[\frac{\mathrm{P}(\mathrm{C}=\mathrm{down},\mathrm{T}=\mathrm{up})}{\mathrm{P}(\mathrm{C}=\mathrm{down},\mathrm{T}=\mathrm{down})}\right]^{-1} \quad (2)$$

The odds ratio can be thought of as a single measure summary for how much leverage the cause variable has over the target variable. Odds ratios larger than one indicate a positive relation between the cause and the target while odds ratios smaller than one indicate a negative relationship. We will later use the odds ratios to encode the strength of the marginal (cause,effect) pair in the visualizations. Using the probabilities in Table 3, the odds ratio for the relation 'Oil price' to 'USA inflation rate' takes the value 9.3, indicating a strong positive relation between the 'Oil price' and 'USA inflation rate'.

At this point we have all the elements to calculate the Joint Probability Table (JPT) for the two node model P('Oil price')P('USA inflation rate'|'Oil price'). The forward-looking probability of the Oil price was given in Table 2 and along with the conditional table displayed in Table 5, this yield the JPT presented in Table 7.

Table 7 JPT for the two-node example ('Oil price' -> 'USA inflation rate')

		Target: USA inflation rate		
		Down	**Up**	
Cause: Oil price	**Down**	0.19	0.08	
	Up	0.15	0.58	**Total**
				1

The procedure described above implicitly derives the conditional probability table from the observed bivariate probability table. This step can be called the 'learning phase' as it learns the causal relationship between variables. The observed bivariate probability table is factored into a root node probability and a conditional table. The 'scoring' phase of the algorithm takes

place when an expert supplies new values for the root node (e.g., the forward-looking probabilities (PTP_f)), which then yields the JPT based on the relations (the conditional probability table) from the learning phase. This is useful when marginalized to the effect variable. The JPT in Table 7 marginalized to P('USA inflation rate'=up) equals 0.67 which shows us the future of the world as extracted from text. The USA inflation rate, of course, depends on more factors than Oil price alone, which we explore in Section 'Results'.

Multiple causes targets

In this section, we will explain why in our case an optimization procedure is required to obtain a well behaved JPT for a BN. A BN factorizes according to the chain formula:

$$P(x_1, x_2, \dots, x_V) = \prod_{i=1}^{V} P(x_i|\text{parents}(x_i)) \quad (3)$$

A BN with one target having three independent causes hence factorizes as:

$$P(T, C_1, C_2, C_3) = P(C_1)P(C_2)P(C_3)P(T|C_1, C_2, C_3) \quad (4)$$

The data provided by the platform yields $P(T,C_1)$, $P(T,C_2)$, and $P(T,C_3)$. The probabilities of the root nodes $P(C_1)$, $P(C_2)$ and $P(C_3)$ are implied by the observed bivariate tables. However, the required $P(T|C_1,C_2,C_3)$ isn't readily available. Let us treat the problem in more detail.

Typically, a BN is supplied with conditional probabilities given by experts. It is doubtful how an expert would be able to pinpoint the higher order conditional probabilities easily. In fact, the conditional probability table of $P(T|C_1,C_2,C_3)$ needs 8 probabilities. At least in the financial texts, sentences that discuss two causes and a target are virtually absent (an example of such sentence would be: "the increase in oil price drives inflation higher, especially in an environment where the interest rates are declining"). While the situation with two causes and a target could be expressed in human language, adding more causes quickly becomes too complex. The question that arises is then - if experts are not able to express such relations in language, are they valid to use to supply the conditional probabilities to fully specify a BN? How to derive conditional probabilities of any order given other supplied information and obtain a bona fide JPT (i.e., that sums up to 1 and all the joint probabilities are in [0,1]) is a problem already treated in Rebonato et al. (2014)[12].

In our specific example, the key is to reconstruct $P(T|C_1,C_2,C_3)$ from the three bivariate probabilities tables $P(T,C_1)$, $P(T,C_2)$, and $P(T,C_3)$. This is not a straightforward task as, in general, there is no exact analytical solution. In fact, the parameters supplied provide fewer parameters than required. For example, the BN in Equation 4 requires 3 marginal and 2^3 elements of trebly conditioned probability, adding up to 11 parameters. The supplied information with the bivariate (source,target) tables counts are 3 parameters for each link C_x-

[12] Other approaches to simplifying complex conditional probability tables exists such as e.g. noisy-or (see Koller (2009), Chapter 5). However, what we will present here is more appropriate and consistent with the information we extract from the platform.

>T (i.e., P(T=Up,C_x=Up), P(T=Up,C_x=Down) P(T=Down,C_x=Up)[13]), adding up to a total of 9 parameters. In other words, we supply less information than the JPT formula requires. By adding more causes, the difference in supplied and needed parameters can get increasingly larger. This means that there will be multiple solutions for P(T|C_1,C_2,C_3). We can find those through an optimization procedure where we will choose the most conservative BN from the set of multiple solutions according to criteria that we will describe.

A problem can arise when a solution does not exist. This could happen if the supplied probabilities, although bona fide probabilities i.e. in the interval [0,1], do not yield a valid P(T|C_1,C_2,C_3). In that case we need to find the closest possible bivariate probabilities that yield a bona fide P(T|C_1,C_2,C_3).

We will show how to solve this problem through an optimization procedure in the next section.

Deriving a well-defined JPT

The JPT of our BN with V vertices V$\epsilon\{x_1,x_2,\ldots x_V\}$ can be represented as a table with 2^V rows and V columns containing all combinations of 'up' and 'down' along with a column containing the probability. We refer to the JPT without the probability column as configuration matrix, where 'up' is replaced by 1 and 'down' is replaced by 0. This table is shown in Appendix B (Table 17).

The edges E of the BN is the collection of causal links. For example, for the BN represented by Equation 3, these are: E$\epsilon\{(C_1,T), (C_2,T), (C_3,T)\}$, with every edge associated with its own set of 4 probabilities $[p_{down,down}, p_{down,up}, p_{up,down}, p_{up,up}]_{(Cx,T)}$ representing the probabilities of the associated causal sentences, such that every $\sum_{i,j} p_{ij} = 1$ for every edge. Note, in the computation of the probabilities from the observed cell counts, the Haldane correction (Haldane, 1956) is applied (where the value of 0.5 is added to every cell) to deal with potential zero counts. The collection of these probabilities for all edges E can be presented in a linear form as a list of probabilities p with length S. We will refer to the vector p as observed edge probabilities.

In the previous section, we discussed how a single set of observed edge probabilities p can give rise to multiple solutions for a BN. To solve this issue and find a unique solution we start by first fitting a log-linear model (Agresti, 1980) to the data[14]. The parameters of a log-linear model describe the main effects and interaction effects in a multidimensional contingency table. For example, a three-way contingency table with variables A, B and C can be decomposed into three main effects (A, B, C), three two-way interaction effects (AB, AC, BC) and one three-way interaction effect (ABC). It is common to restrict the parameters of the log-linear model such that a lower rank JPT that describes the data more parsimoniously is obtained. For the three-way contingency table mentioned above, for example, one could restrict all interaction effects to be zero. Estimating the log-linear model under such restriction yields estimated cell probabilities where P(A,B,C)= P(A)P(B)P(C) holds i.e., the cell probabilities are

[13] The closure relation holds: P(T=Down|C_x=Down) = 1- P(T=Up,C_x=Up) + P(T=Up,C_x=Down) + P(T=Down,C_x=Up)

[14] For a more in-dept treatment of log-linear models we refer the reader to Vermunt, 2005.

the product of single dimensional marginals. This is the most restrictive model that still is interesting.

In our case, we only observe bivariate tables for the edges in the BN. It therefore makes sense to 'reconstruct' the full JPT, yet with only interaction terms allowed that correspond to the observed bivariate tables. This ensures the resulting JPT will not contain higher order associations beyond what can be derived from the observed edges. We note that the sufficient statistics of this log-linear model are exactly the observed bivariate tables. We will refer to this JPT as JPT^{LL} to indicate that this is a valid JPT, although derived through a different parameterization than the typical one of a BN. Subsequently, we use JPT^{LL} to estimate JPT^{BN}, which will now have a single solution, which is the most parsimonious solution that can give rise to observed marginals under the parametrization of a BN.

The JPT^{LL} for the variables in Equation 4, by allowing the main effects and only second order interaction terms corresponding to the observed edge probabilities, is given by:

$$JPT^{LL} = P(C_1, C_2, C_3, T) = \frac{e^{\mu+\lambda_i^{C_1}+\lambda_j^{C_2}+\lambda_k^{C_3}+\lambda_l^{T}+\lambda_{i,l}^{TC_1}+\lambda_{j,l}^{TC_2}+\lambda_{k,l}^{TC_3}}}{\sum_{i,j,k,l} e^{\mu+\lambda_i^{C_1}+\lambda_j^{C_2}+\lambda_k^{C_3}+\lambda_l^{T}+\lambda_{i,l}^{TC_1}+\lambda_{j,l}^{TC_2}+\lambda_{k,l}^{TC_3}}} \quad (5)$$

with the constraints that the terms λ sum to zero over any subscript such as $\sum_{i=1}^{2} \lambda_i^{C_1} = 0$ and $\sum_{i=1}^{2} \lambda_{i,l}^{TC_1} = \sum_{l=1}^{2} \lambda_{i,l}^{TC_1} = 0$ etc. We refer the interested reader to Appendix B for more background information regarding the log-linear models and their parameter restrictions. While the above equation gives insight in which interaction terms are fitted, its matrix notation variant is more convenient:

$$JPT^{LL} = \frac{e^{\mathrm{X}\theta}}{\sum e^{\mathrm{X}\theta}} \quad (6)$$

where the exponentiation is taken elementwise. The model matrix X contains known constants. For example, the term $P(C_1=up, C_2=down, C_3=up, T=up)$ in the JPT^{LL} is associated with the parameters $\mu+\lambda_{up}^{C_1} + \lambda_{down}^{C_2} + \lambda_{up}^{C_3} + \lambda_{up}^{T} + \lambda_{up,up}^{TC_1} + \lambda_{down,up}^{TC_2} + \lambda_{up,up}^{TC_3}$ which equals $\mu+\lambda_{up}^{C_1} - \lambda_{up}^{C_2} + \lambda_{up}^{C_3} + \lambda_{up}^{T} + \lambda_{up,up}^{TC_1} - \lambda_{up,up}^{TC_2} + \lambda_{up,up}^{TC_3}$ and hence the corresponding row in model matrix X equals [1,1,-1,1,1,1,-1,1].

The JPT^{LL} can be marginalized such that the result corresponds to the observed edge probabilities p. For example, for the relation (C_1, T) in Equation 4 holds: $P(C_1, T) = \sum_{C_2,C_3} P(C_1, C_2, C_3, T)$. The weights of this linear combination of the JPT^{LL} can be collected in a matrix M taking the value 1 where the element of the JPT^{LL} is contained in the bivariate marginal term is being calculated from the JPT and zero otherwise.

We aim to find parameters θ in a minimization procedure such that the following holds:

$$p \cong M \frac{e^{\mathrm{X}\theta}}{\sum e^{\mathrm{X}\theta}} \quad (7)$$

At this point, one may ask why we do not simply minimize the likelihood of the log-linear model. The answer lies in the possible (and likely) inconsistency of the observed edge probabilities. For the example, the three observed probability tables $P(C_x,T)$ each contain an

implied marginal T, and across the three tables, those marginals do not need to be consistent. As such, the minimization procedure we will show aims to find a single log-linear parameter for T (and any interaction term parameters that contains T) such that the estimated marginal T is on average the nearest to the three observed versions and yields a well-defined JPT^{LL}.

Once we obtain the JPT^{LL}, we follow Rebonato et al. (2014) and choose parameters for the BN such that we minimize the distance between the JPT^{LL} and the JPT^{BN}. The right-hand side of Equation 3 shows that the BN contains V independent probability terms, which we denote as the parameter vector γ. In the case of Equation 4 (one target and three causes) this means 11 independent terms – one for each root node and eight for the trebly conditioned probability table.

Recall that every term in a BN has its complement. In our binary case, P(A), or more explicitly expressed as P(A=1) has a complementing term P(A=0) which equals 1-P(A=1). For conditional terms, this also holds: P(A=1|B=0,C=1) has complementing term P(A=0|B=0,C=1) which equals 1- P(A=1|B=0,C=1). We construct a vector γ' which contains all the instantiations[15] of the independent terms of γ, including their complements.

We can express the relation between the JPT^{BN} and the parameters γ of the BN via a model matrix A such that when matrix A multiplies the vector $\log(\gamma')$, it yields a vector whose components are the logarithm of the probability associated with each of the states represented by the configuration matrix. Every column of the matrix A corresponds to a component of the BN and is set to one where the configuration matrix equals the state described by the component and zero otherwise. For example, the column of A corresponding to term P(T=1|C_1=1, C_2=0,C_3=0) is set to one for the states in the configuration matrix where C_1=1,C_2=0,C_3=0,T=1 and zero otherwise. Through this notation, we can express the joint probability vector as:

$$JPT^{BN} = e^{A\ln(\gamma')} \tag{8}$$

Note that both the exponentiation and the logarithms are taken elementwise.

We now have all components for the optimization. The loss function that jointly optimizes the parameters of the log-linear model and the BN[16] has the following form:

$$L(\theta,\gamma) = \sum_{s=1}^{S}\left(p_s - \left[M\frac{e^{X\theta}}{\sum e^{X\theta}}\right]_s\right)^2 + \sum_{r=1}^{2^V}\left(\left[\frac{e^{X\theta}}{\sum e^{X\theta}}\right]_r - \left[e^{A\ln(\gamma')}\right]_r\right)^2, \tag{9}$$

where the first term in the optimization procedure yields a JPT^{LL} with no more dependency than can be found in the observed edge probabilities, while the second term seeks to find the parameters for a BN that are as close as possible to the JPT^{LL}.

In summary, the minimization procedure ensures the obtained BN has its parameters based on a JPT^{LL} that contains no more information than can be available from the data acquisition procedure and is well defined. We will apply this procedure in the next section.

[15] An instantiation of a term, say, P(T|C1,C2) means setting the values in this term to a set of values e.g. P(T=Up|C1=Up,C2=Down).

[16] In practice, this optimization is done in seconds to minutes using the Scipy 'minimize' library for networks up to ~15 nodes (a JPT with roughly 32K rows) on an AWS ml.t2.large instance (with 2 vCPU and 8 GiB RAM).

RESULTS

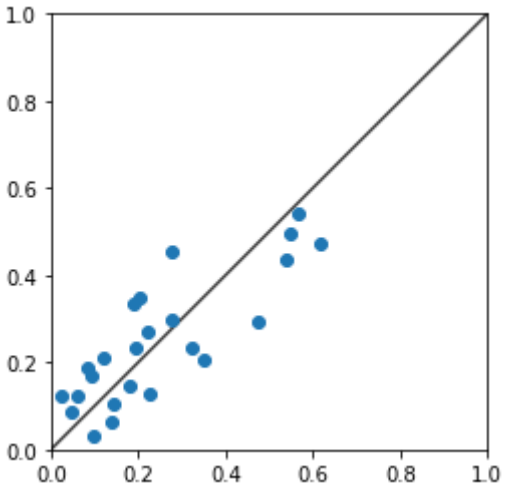
Let us go back to the BN displayed in Figure 1 and refer to it as 'Three driver USA inflation rate model'. Table 8 displays the 24 observed edge probabilities that the model is supplied with in order of decreasing odds ratio. Note that the odds ratios are not supplied to the BN but are just presented to provide insight in the link strength and are computed per Formula 2. According to the experts, the relation for 'Food products prices' and 'USA inflation rate' is the strongest relation among the causal pairs mentioned for this small model. The bivariate probability table for 'Oil price' and 'USA inflation rate' was also presented in Table 3.

Table 8: Observed edge probabilities and odds ratios for the Three driver USA inflation rate model

Driver	Target	Down, down	Down, up	Up, down	Up, up	Odds ratio
Food products prices	USA inflation rate	0.28	0.06	0.10	0.57	27.2
USA GSP	Food products prices	0.47	0.03	0.22	0.28	23.1
Oil price	Food products prices	0.19	0.14	0.05	0.62	17.1
Oil price	USA inflation rate	0.22	0.09	0.14	0.55	9.3
Oil price	USA GDP	0.35	0.32	0.12	0.20	1.8
USA GDP	USA Inflation rate	0.08	0.20	0.18	0.54	1.3

In Figure 2, we present the scatter plot of the 24 observed edge probabilities versus their fitted counterparts. The observed edge probabilities are adjusted as little as possible, yet in a way that they yield a valid BN. As the causal expressions are collected from thousands of authors and ten-thousands of different articles, our impression is that the supplied probabilities are surprisingly coherent.

Figure 2: Observed edge probabilities (x-axis) vs. the fitted probabilities (y-axis) for the three driver USA inflation ratio model in Figure 1.



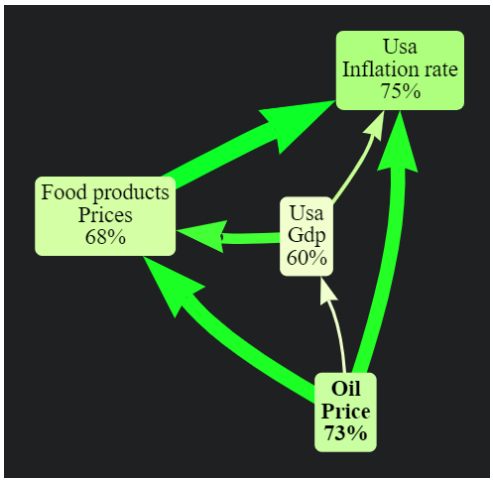
The most basic result of a BN are the marginal probabilities obtained by summing the rows of the JPT according to the marginalization formula for each node v: $P(v_i) = \sum_{v_1,\ldots,v_{|V|} \neq v_i} P(v_1, \ldots, v_i, \ldots, v_{|V|})$.

In Figure 3, the marginal probabilities of the JPT are displayed as percentages in the nodes of the BN. Additionally, the marginal probabilities are used to color the nodes. Probabilities smaller than 0.5 are increasingly red; around the value 0.5 the node color is white and for larger probabilities the node color is increasingly green.

'Oil price' is the single root node and is supplied with the forward-looking probability displayed in Table 2. The resulting 'USA inflation rate' is now at 75% compared to the 67% in the two-node model presented in Section 'Bivariate marginal and single conditional probabilities'.

Apart from the single variable marginals probabilities, it is useful to inspect the bivariate marginal probabilities from the fitted JPT. The fitted odds ratios determine the thickness and color of the edges in Figure 3. The stronger the relation, the thicker the edge. The color shows if a relation is negative (increasingly red when stronger), positive (increasingly green when stronger) and white if an edge the fitted odds ratio is near 1. This representation in the graph facilitates a quick understanding of the relationships, something that is known to be complex for the traditional visualization of BNs.

Figure 3 The graph of the Three driver USA inflation rate model



With the understanding of how to read the resulting graphs, we now turn to more complex examples. For this demonstration we choose three core nodes: 'Gold price', 'Oil price' and 'USA inflation rate'.

The link between 'Oil price' and 'USA inflation rate' was discussed in Section 'Bivariate marginal and single conditional probabilities'. Upfront the relation of either indicator to 'Gold price' is not immediately known. Would 'Oil price' drive 'Gold price' (because its production requires oil)? Or maybe 'USA inflation rate' drives 'Gold price' (because gold is used as a protection against inflation) and hence the effect of 'Oil price' on 'Gold-price' is mediated by 'USA inflation rate'? We will let the platform find the prevailing narrative at the time of writing.

For the three core nodes, we will build two models 1) every node will be allowed to have a single driver 2) every node will be allowed to have two drivers. While larger models are possible, inspection of (too) large graphs does not benefit the analysis. We will, however, present the numerical results behind such larger graphs.

One driver per code node

The data query that retrieves the top single driver per core node results in 5 unique nodes and 20 possible edges. The final DAG contains 9 edges and thus 9*4=36 probabilities to be fitted. Figure 3 shows the observed probabilities and the fitted probabilities under the BN when using the procedure described in Section 'Deriving a well-defined JPT'. When the fitted probabilities are transformed into odds ratios, we observe from inspecting the colors of the links in Figure 4 that every relation in the network conforms our intuition.

Figure 3 Observed edge probabilities (x-axis) vs fitted probabilities (y-axis) for the 1 driver per core node model.

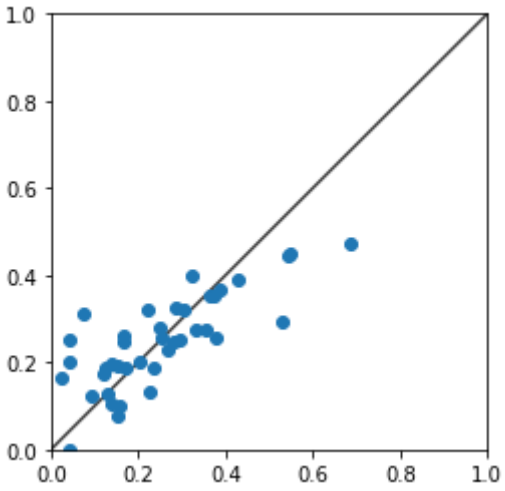
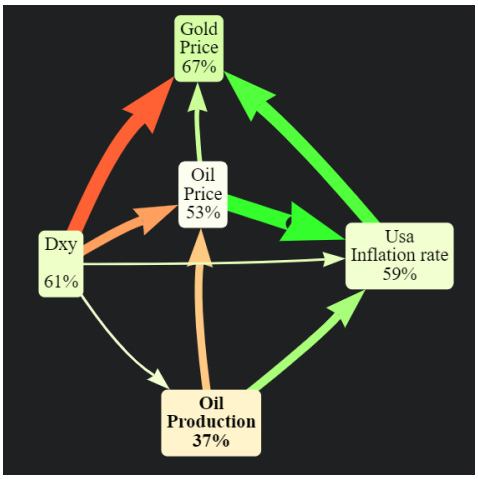


Figure 4 The graph of the single driver per core node model.



Earlier, we wondered how ‘Gold price’ would be related to the other two core nodes. Both ‘Oil price’ and ‘USA inflation rate’ are drivers for ‘Gold price’, ‘Oil price’ being the weaker driver. In addition, we observe how ‘DXY’ (the US dollar against a basket of currencies) has a strongly negative impact on ‘Gold price’. When ‘DXY’ goes up, ‘Gold price’ goes down. This is in line with intuition. In fact, when the value of the dollar increases relative to other currencies around the world, the price of gold tends to fall in USD terms. It is because gold becomes more expensive in other currencies. The negative relation between ‘DXY’ and ‘Oil price’ shows that a strong dollar makes the oil price cheaper. Earlier we discussed the relation between ‘Oil price’ and ‘USA inflation rate’. In Figure 4, we observe how both factors have a common driver: ‘Oil production’. Higher ‘Oil production’ drivers ‘Oil price’ down and ‘USA inflation rate’ up.

Table 9 displays the JPT marginalized to the three core nodes. The future scenario where both ‘Gold price’ and ‘Oil price’ are down and ‘USA inflation rate’ is up is the most unlikely scenario, while the most likely scenario is that all three variables are up.

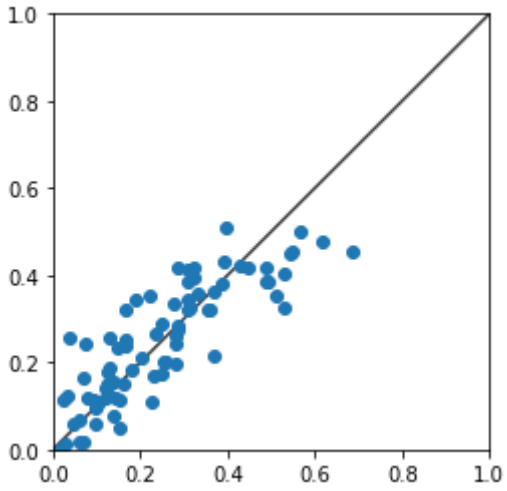
Table 9 The JPT for the single driver per core node model

Gold price	Oil price	USA inflation rate	Probability
down	down	down	0.20
down	down	up	0.00
down	up	down	0.04
down	up	up	0.09
up	down	down	0.12
up	down	up	0.15
up	up	down	0.05
up	up	up	0.35

Two drivers per core node

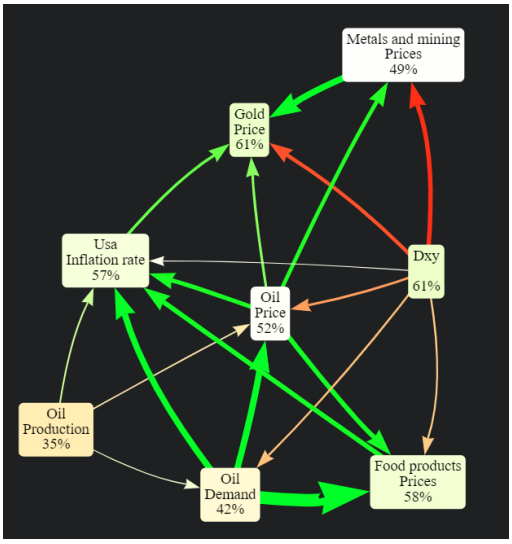
The data query that retrieves the top two drivers per core node results in 8 unique nodes and 56 possible edges. The final DAG contains 20 edges and thus 20*4=80 probabilities to be fitted. Figure 5 shows the observed probabilities are reasonably close to the fitted probabilities.

Figure 5 Observed edge probabilities (x-axis) vs fitted probabilities (y-axis) for the two drivers per core node model



In Figure 6, the graph of the BN is shown. While the experts indicated collectively a link between 'DXY' and 'Oil production' with an observed odds ratio of 1.6, the model fitted an odds ratio of 1.03, practically indicating no direct relation. For purposes of readability this link is not displayed in the graph. The direct link between 'DXY' and 'USA inflation rate' also a weak one with an odds ratio of 1.15. The influence of 'DXY' on 'USA inflation rate' happens indirectly via 'Oil price' and 'Food product prices'. If the dollar gets weaker, either factor will increase. In turn, both 'Food product prices' and 'Oil prices' push 'USA inflation rate' higher.

Figure 6 The graph of the two drivers per core node model



Reading DAGs with more and more drivers becomes increasingly difficult but before 'using' the BN built in this way every link should be inspected against economic common sense. Indeed, we acknowledge that economic considerations and review are needed to curate the graph to a fully acceptable BN that can be used in practice. Nevertheless, this paper demonstrates how a large part of the graph creation and population is possible in a fast way through recent advances in NLP. A network of the size of the one in Figure 6 could require many days if relying only on experts' input.

Another question is if adding more links helps to better pin down the probability distribution of the core nodes under consideration. We can check this by performing some sensitivity analyses. In Table 10 we present the progression of the marginal probabilities of the nodes for a series of "sub-models" starting from 0 links to the maximum number of links. Note that so far, we have always presented the models with the maximum number of links.

Table 10 The evolution of the marginal probabilities when adding more links to the two drivers per core node model

Node	0	5	10	15	18	19	20
DXY	0.61	0.61	0.61	0.61	0.61	0.61	0.61
Food products prices	0.67	0.67	0.70	0.62	0.58	0.58	0.58
Gold price	0.58	0.46	0.60	0.58	0.61	0.61	0.61
Metals and mining prices	0.69	0.69	0.69	0.48	0.51	0.50	0.49
Oil demand	0.75	0.45	0.45	0.49	0.43	0.42	0.42
Oil price	0.73	0.49	0.50	0.51	0.52	0.52	0.52
Oil production	0.55	0.55	0.55	0.55	0.42	0.41	0.35
USA inflation rate	0.80	0.62	0.63	0.61	0.64	0.64	0.57

We observe a stabilization of the results when adding more and more links to the model which is reassuring. We present the JPT for the core nodes: 'Gold price', 'Oil price' and 'USA inflation rate' in Table 11 as we increase the number of links.

Table 11 The evolution of the JPT when adding more links to the two driver per core node model

Gold price	Oil price	USA inflation rate	0	5	10	15	18	19	20
down	**down**	**down**	0.02	0.17	0.12	0.21	0.20	0.22	0.26
down	**down**	**up**	0.09	0.10	0.09	0.05	0.05	0.04	0.01
down	**up**	**down**	0.06	0.03	0.01	0.02	0.01	0.01	0.01
down	**up**	**up**	0.24	0.23	0.17	0.14	0.13	0.12	0.11
up	**down**	**down**	0.03	0.15	0.19	0.12	0.12	0.09	0.10
up	**down**	**up**	0.13	0.09	0.10	0.10	0.11	0.13	0.11
up	**up**	**down**	0.08	0.02	0.04	0.03	0.03	0.04	0.06
up	**up**	**up**	0.34	0.20	0.27	0.32	0.35	0.35	0.34

We notice that also the JPT is pretty much stable. This is another type of sensitivity analysis which shows the robustness of our procedure. A scenario where all three core nodes are up, is estimated to be the most likely scenario with probability of 34% in the model with 20 links. The second most probable scenario is all the core nodes down with probability of 26%. The fact that the most likely scenarios are either everything up or everything down confirms our intuition of the close relation between the three indicators. Two scenarios (1%) are the least probable: the scenario where 'Gold price' =down, 'Oil price' = down and 'USA inflation rate' = up and the scenario where 'Gold price' =down, 'Oil price' = up and 'USA inflation rate' = down. Compare this with Table . We see that the 'all up' state has very close

probability to the one in Table , while the ‘all down’ state has slightly increased from 20% to 26% because of the introduction of an additional second driver.

Next, we investigate the stability of the JPT under (non-biased or random) NLP errors and article inclusion changes. We address this by performing a parametric bootstrap on the two drivers per core node model as follows: for the root nodes, we sample 1,000 times from a binomial distribution Bin(N,π) with N set to the (rounded) weight corresponding to the root node at hand and π set to the observed PTP_f of the root node. For the observed edge probabilities, we sample 1,000 times from a multinomial distribution Multi(N, $\boldsymbol{\pi}$), with N set to the (rounded) weight associated with the observed edge probabilities and $\boldsymbol{\pi}$ set to the vector of observed edge probabilities $\boldsymbol{\pi}$ = [down_down, down_up, up_down, up_up] for the edge at hand. The samples for both the root nodes as well as the observed edge probabilities are subsequently transformed back into the probability domain. We thus obtain 1,000 new dataset that are perturbed versions of the observed dataset where the perturbation is governed by the sample fluctuations of the given distributions. Subsequently, we fit 1,000 BNs and collect the 5^{th}, 50^{th} and 95^{th} percentile of the distribution of the marginal probabilities as well as the same percentiles from the distribution of the JPTs for the core nodes: ‘Gold price’, ‘Oil price’ and ‘USA inflation rate’.

Table 12 displays the results of the bootstrapped data for the marginal probabilities. The column ‘Est’ displays the estimates based on the original dataset for reference. The median probabilities of the replicated data are nearly identical to the estimates based on the original dataset. The small spread between the 5^{th} and 95^{th} percentiles demonstrate that one would not draw different conclusions if the data collection had turned out slightly different due to normal sample fluctuations.

Table 12 Percentiles of the distribution of marginal probabilities for the bootstrapped data of the two driver per core model

Node	Est	5%	50%	95%
DXY	0.61	0.58	0.61	0.63
Food products prices	0.58	0.54	0.58	0.62
Gold price	0.61	0.57	0.61	0.64
Metals and mining prices	0.49	0.45	0.50	0.56
Oil demand	0.42	0.37	0.42	0.46
Oil price	0.52	0.49	0.52	0.55
Oil production	0.35	0.30	0.36	0.43
USA inflation rate	0.57	0.53	0.57	0.60

Table 13 displays the results of the bootstrapped data for JPTs for the core nodes: ‘Gold price’, ‘Oil price’ and ‘USA inflation rate’. The column ‘Est’ displays the estimates based on the original dataset for reference. The median probabilities of the replicated data show only slight differences compared to the estimates based on the original dataset. The small spread between the 5^{th} and 95^{th} percentiles demonstrate that the conclusions with respect to the JPT would not change if the data collection had turned out slightly different due to normal sample fluctuations.

Table 13 Percentiles of the distribution of the JPT of the core nodes for the bootstrapped data for the 2 drivers per core model

Gold price	Oil price	USA inflation rate	Est	5%	50%	95%
down	**down**	**down**	0.26	0.20	0.24	0.27
down	**down**	**up**	0.01	0.00	0.02	0.06
down	**up**	**down**	0.01	0.01	0.03	0.05
down	**up**	**up**	0.11	0.07	0.10	0.13
up	**down**	**down**	0.10	0.07	0.10	0.14
up	**down**	**up**	0.11	0.07	0.11	0.15
up	**up**	**down**	0.06	0.03	0.06	0.09
up	**up**	**up**	0.34	0.30	0.34	0.37

Finally, we present Tables 14 and 15 to demonstrate the effect of adding more drivers per core node. The resulting graphs quickly become complex to read and we will not show them. While such larger networks also can be used in practice, we note that both the marginal probabilities as well as the probabilities of the JPT of the core nodes are sufficiently similar to the two-drivers-per-core-node model, which gives confidence the smaller and more interpretable networks could yield robust results. For clarity: the model 'PTP_f implied' is BN with the three non-connected core nodes. Their values are supplied by the forward-looking probabilities and no fit is needed. The resulting JPT for this model is simply the multiplication of the root node probabilities P('Gold price', 'Oil price', 'USA inflation rate')=P('Gold price')P('Oil price')P('USA inflation rate').

Table 14 Marginal effects for the core nodes for models with increasing number of drivers per core node

Node	PTP_f	1 driver	2 drivers	3 drivers	4 drivers	5 drivers
Gold price	0.59	0.76	0.61	0.57	0.55	0.55
Oil price	0.73	0.53	0.52	0.48	0.54	0.49
USA inflation rate	0.80	0.59	0.57	0.60	0.58	0.51

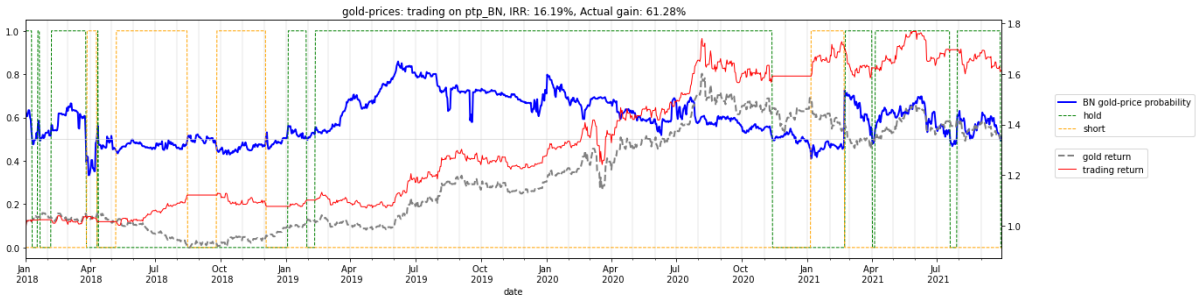
Table 15 The JPT of the core nodes for models with increasing number of drivers per core node

Gold price	Oil price	USA inflation rate	PTP_f implied	1 driver	2 drivers	3 drivers	4 drivers	5 drivers
down	**down**	**down**	0.02	0.20	0.26	0.25	0.22	0.26
down	**down**	**up**	0.09	0.00	0.01	0.05	0.04	0.06
down	**up**	**down**	0.06	0.04	0.01	0.04	0.06	0.05
down	**up**	**up**	0.24	0.09	0.11	0.09	0.13	0.09
up	**down**	**down**	0.03	0.12	0.10	0.09	0.07	0.11

up	**down**	**up**	0.13	0.15	0.11	0.12	0.12	0.08
up	**up**	**down**	0.08	0.05	0.06	0.03	0.06	0.07
up	**up**	**up**	0.34	0.35	0.34	0.33	0.30	0.28

Finally, we will show a real-world test to measure the success of a fit network. For this purpose, we built a series of BNs centred around the price of gold. More specifically, for the date range Jan 1, 2018 – Oct 14, 2021, we query the data source for the top 4 drivers of 'Gold price' for every day in the period mentioned. Note that this allows for the model to contain different drivers each day. During the 3 years, we observe 6 different drivers (DXY, stock market sentiment, cryptocurrency prices, gold risk, USA inflation rate and USA interest rates). We subsequently fit the BN for every day after which we derive the marginal probability of 'Gold price'. To back-test the performance of this probability, we deploy the following simple trading rules: if the probability>0.55, we buy gold. If the probability drops below 0.5, we exit the position. If the probability drops below 0.45, we short gold and once the probability increases over 0.5, we exit the short position. Trading costs are set to 5 bps. Note that the trade is executed the (business) day after the probability is computed. In this way we guarantee the information in the BN is truly forward looking. Figure 7 displays the marginal probability for 'Gold price' in blue along with the return for gold (Fred: GOLDPMGBD228NLBM, Gold Fixing Price 3:00 P.M. (London time) in London Bullion Market, based in U.S. Dollars). For most of the period, the BN suggests holding gold. The holding periods are indicated with the green boxes, while the yellow boxes indicate the periods in which a short position is taken. During the 3.5-year period, the trading strategy shown in red yields a 61% gain, outperforming a long-only strategy on gold with about 11%.

Figure 7 Back-test of a BN centred around gold price.



CONCLUSIONS AND FURTHER WORK

In this paper, we demonstrated a way to automatically convert causal reasoning expressed by thousands of experts into BNs. To the best of our knowledge this is the first paper to show how the trends expressed in causal reasoning by a crowd of authors can be cast as a BN, and hence yield a JPT which expresses different future states of the world and their probability. We consider this to be the main scientific contribution of the paper. The results show that reasonable looking networks result from this procedure, not only in the graph structure, but also in the automatically derived relations between the nodes.

There are a lot of open lines of research starting from the findings of this paper. In particular, a different way to construct and ensure a coherent DAG can be explored; the NLP

engine can be extended to consider higher order (e.g. double, triple etc.) probabilities; the extension to continuous variables of the BN could yield an answer to the question of how much a variable can be up or down – in other words infer the magnitude of a move in a variable not only the direction. Finally, models that include both directed and undirected links like chain graphs can be used to account for some cyclical relations that might arise in the real world. We leave this for future research.

REFERENCES

Agresti A., 1990. Categorical Data Analysis, New York, John Wiley

Beller, Charles; Katz, Graham; Ginsberg, Allen; et.al (2016). Watson Discovery Advisor: Question-answering in an industrial setting. Proceedings of The 2016 Conference of the North American Chapter of the Association for Computational Linguistics. San Diego, California. pp. 1-7.

Corkill, Daniel D.; Gallagher, Kevin Q.; Johnson, Philip M. (1987). Achieving flexibility, efficiency, and generality in blackboard architectures. Proceedings of the National Conference on Artificial Intelligence. Seattle, Washington. pp. 18–23.

Denev, A., (2015). Probabilistic graphical models: a new way of thinking in financial modelling, Risk Books.

Denev, A. and Mutnikas, Y., (2016). A formalized, integrated and visual approach to stress testing. Risk Management, 18(4), pp.189-216.

Friendly, M. & Meyer, D. (2015). Visualizing Categorical Data with R.

Ginsberg, A., (1993). A Unified Approach to Automatic Indexing and Information Retrieval. IEEE EXPERT, Volume 8 No. 5, pp. 46-56.

Ginsberg, A., May (2006). The Big Schema of Things: Two Philosophical Visions of The Relationship Between Language and Reality and Their Implications for The Semantic Web. Identity, Reference, and The Web, workshop, WWW2006, Edinburgh Scotland.

Haldane JBS. (1956). The estimation and significance of the logarithm of a ratio of frequencies. In: Annals of Human Genetics 20 (4), pages 306-311.

Koller, D. & Friedman, N. (2012). Probabilistic graphical models principles and techniques. MIT Press.

Moghimifar, F. & Rahimi, A. & Baktashmotlagh, M. & Li, X. (2020). Learning Causal Bayesian Networks from Text. Proceedings of the 18th Annual Workshop of the Australasian Language Technology Association.

Murphy, K. (2012). Machine Learning: A Probabilistic Perspective. MIT Press.

Pearl, J. (2009). Causality: models, reasoning, and interference. Cambridge University Press.

Rebonato, R., (2010). Coherent Stress Testing. Chichester: Wiley.

Rebonato, R. & Denev, A. (2014). Portfolio management under stress: a Bayesian-net approach to coherent asset allocation. Cambridge University Press.

Sanchez Graillet, Olivia & Poesio, Massimo. (2004). Acquiring bayesian networks from text.

Vermunt, J. K. (2005). Log-linear modelling. In B. Everit, & D. C. Howell (Eds.), Encyclopedia of Behavioral Statistics (pp. 1082-1093). John Wiley & Sons Inc.

APPENDIX A

Table 16 Example sentences for the three drivers of 'USA inflation rate' for every combination of the driver and target trend

Driver	Target	Driver trend	Target trend	No.	Sentence
Oil prices	USA inflation rate	Down	Down	1	With inflation easing on the back of lower oil prices and falling commodity prices has allowed RBI to lower interest rates, and further interest rate cuts cannot be ruled out, which would further boost the economic climate and consumer sentiment and create a positive demand environment for the auto sector.
				2	Wholesale inflation fell 0.2 % in December, driven by lower gasoline prices.
				3	U.S. inflation dropped to 1.6 % in January due to lower fuel prices, hitting a 19-month low.
		Down	Up	1	U.S. consumer prices rose modestly last month, weighed down by falling gasoline prices.
				2	US consumer prices edge up just 0.1 % in January, weighed down by falling gasoline prices.
				3	Data showed U.S. consumer prices were unchanged in November, in line with expectations, due to a sharp decline in gasoline price but underlying inflation pressures remained firm amid rising rents and healthcare costs.
		Up	Down	1	In June, we saw an inflation slowdown despite the spike in gasoline prices.
				2	Additionally, higher crude prices tend to adversely affect inflation.
				3	Furthermore, it was observed that despite higher oil prices, inflation in the external environment of the Polish economy was moderate, and in the euro area it was low.
		Up	Up	1	Inflation has been on an uptrend recently thanks mainly to higher oil prices.
				2	Surging gasoline prices caused U.S. consumer prices to rise last month at the fastest pace since March.

				3	A few days ago, we discussed how soaring oil prices have been a stagflationary double whammy to emerging markets, which have been hit not only by a surging dollar, resulting in a collapse in local currencies and spiking import costs, but a spike in local currency oil and gasoline prices resulting in a surge in inflation and a slowdown in the economy as local infrastructure grinds to a halt.
Food products prices	USA inflation rate	Down	Down	1	The softer inflation figures were attributed to a decline in food prices.
				2	Lower prices of food and non-alcoholic beverages drove inflation down last month.
				3	Official figures show that the recent fall in food prices has pushed down inflation.
		Down	Up	1	In his view, the progress of the comparative base of pork prices - which declined from July 2020 onwards - will boost food inflation, while the tightening of the labour market - with less contracting capacity due to improved employment figures - will do the same with the basic.
				2	Food inflation increased to 0.4 percent, the lowest since March 2018 with a significant drop in fresh food prices.
				3	The IMF report, which acknowledged the efforts of the Central Bank of Nigeria (CBN) to rein in inflation, however, maintained that despite an expected easing of food prices, inflation is projected to remain in double-digits and above the CBN's target range.
		Up	Down	1	Food inflation dipped on account of smaller increases in the prices of non-cooked food.
				2	On the other hand, food inflation dipped to 1.6 per cent last month, on account of smaller increases in the prices of non-cooked food and restaurant meals.
				3	He said the SBP has accepted that inflation was due to an increase in food prices and that these supply side pressures were likely to be temporary, with an expectation of decline in average inflation within the range of seven to nine percent.
		Up	Up	1	If food prices are rising, genuine "main street" inflation is on the rise.
				2	Consumer inflation has risen, mostly driven by a surge in pork prices.
				3	The CBRT also saw little risk in runaway inflation accelerating from here, despite prices - especially food - still soaring.
USA GDP	USA inflation rate	Down	Down	1	There are signs that the U.S. economy is stumbling, and that low inflation is more stubborn than the fed previously thought, both of which argue for lower rates.
				2	Inflation should fall by no less than 5 points, partly because of the cooling of the economy as by restrictions

					on access to foreign currency," said economist Gabriel Monzón.
				3	Working relentlessly to bring prices under control, Volcker raised the Fed's benchmark interest rate from 11% to a record 20% by late 1980 to try to slow the economy's growth and thereby shrink inflation.
		Down	Up	1	0.5 per cent in February 2020 (year-on-year), while food inflation stood at 1.0 per cent, mainly due to reported price declines for fresh vegetables and fruits.
				2	Economists expect the broad deflationary trend to give way to higher inflation at least next year partly because of extraordinary measures by the federal reserve to cushion the economy’s fall, including programs to help keep companies afloat.
				3	Although economic activity is slowing down, rising oil prices may put pressure on inflation, something that would force the federal reserve to raise rates sooner than expected.
		Up	Down	1	Bangko Sentral ng Pilipinas (BSP) governor Amando Tetangco said he did not foresee the inflation target being breached over the policy horizon despite strong GDP growth.
				2	What is notable, although nothing strange here, is that inflation and above all the inflation in 2020 would have passed 30% despite a collapse of the economy calculated at 11% and despite, also, the freezing of gas and electricity tariffs: 0.6% in the year, reports the chapter of the index that represents them.
				3	The moves were an attempt to boost economic activity and, in turn, lift still-sluggish inflation towards the bank's just-below-two-percent goal.
		Up	Up	1	The prospect of additional stimulus and ongoing vaccinations has raised concerns that as Americans eventually release pent-up demand for airline tickets, hotel rooms, new clothes and other goods and services, the economy might accelerate, and inflation could surge above 2%.
				2	Typically, if we are headed toward the end of an economic expansion, resources become scarce in an overheated economy leading to surging inflation so that the fed has to swoop in to cool things off by raising rates and pulling liquidity out of the financial system.
				3	Data on Tuesday showed U.S. consumer prices rose marginally in May as gasoline price increases slowed and the underlying trend continued to suggest moderate inflation in the economy.

APPENDIX B

In this appendix we provide background information on the fitting procedure employed in the paper and already described in paragraph 'Deriving a well-defined JPT'. Let *u* represent **up** and *d* represent **down** states. Regardless of the DAG structure, the model can be represented via the following exhaustive table:

Table 17: Joint probability table

$(x_1,\ldots x_n)$	**probability**
(d,…,d,d)	*p(d,…,d,d)*
(d,…,d,u)	*p(d,…,d,u)*
⋮	⋮
(u,…,u,u)	*p(u,…,u,u)*

This is called the **joint probability table (JPT)**. For the eight vertex DAG shown in Section 'Two drivers per core node model', the JPT contains 256 elements. In terms of simple parameter counting, this isn't necessarily daunting because there are many thousands of text statements related to the economic variables in this DAG. However, any individual text statement about these variables will only pertain to one or two of them, e.g., "oil processing tends to rise as oil demand rises" or "metal and mining production cost tend to fall as the US dollar rises". This relative coarse information doesn't immediately lend itself to the problem of estimating the joint probability table, as already explained in the body of the paper.

The log-linear model

In view of the data structure and characteristics above, we propose the following parametric model for the entries of the JPT:

$$p(x_1,\cdots,x_n)=\frac{\exp\left(\mu+\sum_{i=1}^{n}\lambda_{x_i}^{(i)}+\sum_{jk}\lambda_{x_jx_k}^{(jk)}\right)}{\sum_{y_1}\ldots\sum_{y_n}\exp\left(\mu+\sum_{i=1}^{n}\lambda_{y_i}^{(i)}+\sum_{jk}\lambda_{y_jy_k}^{(jk)}\right)}, \tag{10}$$

where the sums $\sum_{jk}(\ldots)$ take place over all arcs $j\to k$ in the DAG. This is an example of a **log-linear model**, which finds use in applications ranging from statistical physics to spell checking (Murphy, §19.3). It also generalizes the basic **saturated model** for two-way interactions (Friendly & Meyer, 2015, Eq. 8.2). By design, the **interaction strength** $\lambda_{xy}^{(jk)}$ captures co-movement between the binary random variables *x*, *y* at the respective ends of the arc $j\to k$.

Strictly speaking, μ is a redundant parameter; it factors out of $p(x_1,\cdots,x_n)$. In fact, there's considerable additional redundancy in the parameters of this log-linear model. Consider a typical interaction strength $\lambda_{xy}=\lambda_{xy}^{(jk)}$. Define

$$\lambda_{x+}:=\sum_{y\epsilon\{u,d\}}\lambda_{xy}$$

and define λ_{+y} and λ_{++} analogously. If

$$\alpha_x := \tfrac{1}{2}\lambda_{x+} - \tfrac{1}{4}\lambda_{++} \text{ and } \beta_y := \tfrac{1}{2}\lambda_{+y} - \tfrac{1}{4}\lambda_{++},$$

then

$$\lambda_{xy} = \tfrac{1}{4}\lambda_{++} + \alpha_x + \beta_y + \tilde{\lambda}_{xy}$$

where $\tilde{\lambda}_{x+} = \tilde{\lambda}_{+y} = 0$ for all *x*, *y*. Applying this formula to every interaction strength in (10), we can absorb the constants $\frac{1}{4}\lambda_{++}$ into μ and the first order terms α and β into the $\lambda_x^{(i)}$ terms, leaving us with an equality of the form

$$\mu + \sum_{i=1}^{n} \lambda_{x_i}^{(i)} + \sum_{jk} \lambda_{x_j x_k}^{(jk)} = \tilde{\mu} + \sum_{i=1}^{n} \tilde{\lambda}_{x_i}^{(i)} + \sum_{jk} \tilde{\lambda}_{x_j x_k}^{(jk)}, \qquad (11)$$

where

$$\tilde{\lambda}_{x+}^{(jk)} = \tilde{\lambda}_{+y}^{(jk)} = 0$$

For all *x*, *y* and for all arcs $j \to k$. Taking this one step farther, we can also absorb the means $\frac{1}{2}\tilde{\lambda}_{+}^{(i)}$ of the first order terms $\frac{1}{2}\tilde{\lambda}_{x}^{(i)}$ into $\tilde{\mu}$. So doing, we may also assume that $\tilde{\lambda}_{+}^{(i)} = 0$ for all vertices *i*. Accordingly, in view of the equality (11), we simply impose the constraints

$$\lambda_{+}^{(i)} = \lambda_{x+}^{(jk)} = \lambda_{+y}^{(jk)} = 0$$

on the original model parameters. These restrictions coincide with Friendly & Meyer (2015) Eq 8.3. According to Friendly & Meyer (p 349, para. 2), these constrains arise because not every parameter is separable estimable. They call them "typical ANOVA sum-to-zero restrictions". Subject to these restrictions, there is evidently one degree of freedom for every **main effect** $\lambda^{(i)}$. There's also a single degree of freedom for every 2x2 array λ_{xy}. These take the form $\begin{pmatrix} +\theta & -\theta \\ -\theta & +\theta \end{pmatrix}$ where $\theta \epsilon \mathbb{R}$. Thus, if the DAG has *a* arcs, model (10) has *n* + *a* degrees of freedom, one for each of the main effects and one for each of the *a* interactions. As noted earlier, μ factors out of the distribution so it is not included in the parameter count.

Using matrix-vector notation, the JPT corresponding to (10) can be expressed as (scaled) elementwise exponential

$$\mathbf{JPT}^{LL}(\boldsymbol{\Theta}) := \frac{\boldsymbol{e}^{\boldsymbol{X\Theta}}}{\mathbf{1} \cdot \boldsymbol{e}^{\boldsymbol{X\Theta}}}$$

Here, $\boldsymbol{\Theta}$ is an *(n+a)*-dimensional parameter vector and **X**, the **model matrix**, has entries 0 or ±1. It has 2^n rows, the length of the JPT vector. The super script 'LL' highlights the loglinear nature of the model.

Fitting the model to observed marginal probabilities

Using the data provided by the upstream NLP analysis, we form empirical estimates $\hat{p}(x_{j_\ell}, x_{k_\ell})$ of the bivariate marginal probabilities

$$p(x_{j_\ell}, x_{k_\ell}) = p(+\cdots+, x_{j_\ell}, +\cdots+, x_{k_\ell}, +\cdots+)$$

corresponding to all arcs $j_\ell \rightarrow k_\ell, 1 \leq \ell \leq a$, in the DAG. Here we apply the "+" convention introduced earlier. Note that the nature of the data is such that forming empirical estimates of higher order marginals is infeasible. Stacking these marginals into a *4a*-dimensional vector **P**, the previous equation can be written as

$$\mathbf{P} = \mathbf{M} \cdot \mathbf{JPT},$$

where **M** is a *4a x* 2^n matrix of zeros and ones. Accordingly, in the light of the expression for the $\mathbf{JPT}^{\mathbf{LL}}$, it is natural to see a parameter vector $\boldsymbol{\Theta}$ that minimizes the distance between the empirical marginal vector

$$\widehat{\boldsymbol{P}} = \begin{pmatrix} \vdots \\ \hat{p}(x_{j_\ell} = u, x_{k_\ell} = u) \\ \hat{p}(x_{j_\ell} = u, x_{k_\ell} = d) \\ \hat{p}(x_{j_\ell} = d, x_{k_\ell} = u) \\ \hat{p}(x_{j_\ell} = d, x_{k_\ell} = d) \\ \vdots \end{pmatrix}$$

And corresponding model marginal vector

$$\mathbf{P}(\boldsymbol{\Theta}) := \mathbf{M}\frac{\boldsymbol{e}^{\boldsymbol{X\Theta}}}{\mathbf{1} \cdot \boldsymbol{e}^{\boldsymbol{X\Theta}}}$$

While we take this natural distance minimization criterion into account, we do not apply it fully. This is because obtaining a best fir log-linear model is not the primary objective. The log-linear model does not fully account for the underlying DAG structure. Evidently univariate marginals can be accommodated along with bivariate marginals; we would simply expand the empirical marginal vector to include the relevant univariate estimates and expand the matrix **M** to match. An interesting property of the log–linear model, testifying to its relative parsimony, is that when fitting it to all univariate marginals and the bivariate marginals corresponding to DAG arcs, the number of model parameters *n + a* is precisely the same as the number of fitted marginal distributions.

The Bayesian network

The log-linear model cannot account for the orientation of the arcs in the underlying DAG. It would be entirely indifferent to the reversal of every arc in the graph. However, by design, the BN certainly accounts for the orientation of the arcs. Accordingly, the goal is to fit a BN to the data. As explained in the next section, it would be very difficult to fit a BN directly to the empirical marginal vector DP. However, this is not the case for the log–linear model. It serves as an intermediary to facilitate the overall fitting process.

A BN is parameterized by the conditional probabilities entering the expression $p(\mathbf{x}) = \prod_{v=1}^{n} p\left(x_v \middle| \boldsymbol{x}_{pa(v)}\right)$. Thus, if $x_i \epsilon \{u, d\}$ for all *I*, the BN parameters are the conditional probabilities

$$p\left(x_v = u \middle| \mathbf{x}_{pa(v)}\right), p\left(x_v = d \middle| \mathbf{x}_{pa(v)}\right), \quad v\epsilon \{1, \dots, n\}$$

In general Byesian networks with binary outcomes $\{u, d\}$, there is some number $2P$ of conditional probabilities, *P* of which take the form $p\left(x_v = u \middle| \mathbf{x}_{pa(v)}\right)$ while the other *P* takes the form $p\left(x_v = d \middle| \mathbf{x}_{pa(v)}\right)$. We denote the 'up' probabilities by the *P*-dimensional vector $\boldsymbol{\gamma}$ and the full set of *2P* probabilities by the vector $\boldsymbol{\gamma}'$. Evidently $\boldsymbol{\gamma}'$ is determined by $\boldsymbol{\gamma}$ $\boldsymbol{s}$ince conditional probabilities sum to one; i.e.

$$\gamma'_k + \gamma'_{P+k} = 1, \quad 1 \leq k \leq P$$

Furthermore, the equation $p(\mathbf{x}) = \prod_{v=1}^{n} p\left(x_v \middle| \mathbf{x}_{pa(v)}\right)$ directly translates into the following expression for the BN's JPT vector:

$$\mathbf{JPT}^{BN}(\gamma) = e^{\mathbf{A}\, \boldsymbol{ln}\, \gamma'}$$

Here the logarithm and the exponential are evaluated elementwise, and **A** is a 2^n *x 2P* matrix of zeros and ones determined by the DAG topology.

We choose $\boldsymbol{\Theta}$ **and** $\boldsymbol{\gamma}$ simultaneously by minimizing the objective

$$L(\boldsymbol{\Theta}, \boldsymbol{\gamma}) \;=\; \left\|\widehat{\mathbf{P}} \;-\; \mathbf{P}(\boldsymbol{\Theta})\right\|^2 \;+\; \|\mathbf{JPT}^{LL}(\boldsymbol{\Theta}) - \mathbf{JPT}^{BN}(\boldsymbol{\gamma})\|^2,$$

where $\|\dots\|$ denotes the appropriate Euclidean norms. We subsequently study the BNs corresponding to the fitted value $\boldsymbol{\gamma}^*$for several different DAGs.

Motivation

In principle, instead of using the two-stage procedure detailed above, we might have minimized the simpler objective

$$L(\boldsymbol{\gamma}) \;=\; \left\|\widehat{\mathbf{P}} \;-\; \mathbf{M} \cdot \mathbf{JPT}^{BN}(\gamma)\right\|^2,$$

with respect to $\boldsymbol{\gamma}$. However, according to Rebonato & Denev (2014), there are typically many possible parameter vectors $\boldsymbol{\gamma}$ corresponding to a given set of bivariate marginal probabilities, which renders this minimization problem ill-posed. While Rebonato & Denev suggest a regularization method based on the addition of an entropy-based penalty term, it involves an arbitrary weighting factor. Unfortunately, by experimenting with relatively simple DAG, we find that the penalized optimizer is highly dependent on the choice of this weighting factor. It would be infeasible to retune this factor for every DAG encountered in practice.

We introduce the log-linear model as an intermediary to deal with the ill-posed conditions noted above. By making a full JPT vector $\mathbf{JPT}^{LL}(\boldsymbol{\Theta})$ available for comparison to $\mathbf{JPT}^{BN}(\boldsymbol{\gamma})$, these difficulties seem to vanish. Even in cases where the regularization method

described above proves problematic, choosing γ to minimize the Euclidean distance between the two JPT vectors is a numerically well-behaved problem that gives sensible results.

We argue that such an intermediary model must satisfy several important design constraints:

1) The model must be **conservative**, in that is has just enough parameters to fit the observed marginals and is as "impartial" as possible.
2) The model must be reasonably easy to fit to the data, and
3) It lends itself to the problem of fitting a BN.

Regarding Criterion 1), the log-linear model has precisely the number of parameters required to capture the main and pairwise interaction effects, which are the only effects discernible form the data made available by the upstream NLP analysis. A key measure of the log-linear model's impartiality is the fact the its model matrix **X** is **orthogona**l. Regarding 2) there are several proven procedures for fitting log-linear models; see for example chapter 8 of Friendly & Meyer (2015). Minimizing the objective $L(\boldsymbol{\Theta}, \boldsymbol{\gamma})$ is a practical compromise that more readily accounts for 3) the end goal of fitting a BN. As we put it in Section 'Deriving a well-defined JPT', "the minimization procedure ensures the obtained BN has its parameters based on a log–linear JPT that contains no more information than can be available from the data acquisition procedure and is well defined."