Applying Bayesian Networks to Reduce Fuel Consumption in Public Transportation

Applying Bayesian Networks to Reduce Fuel Consumption in Public Transportation

Federico Delussu, Faisal Imran, Rosa Meo and Michela Pellegrino
Dipartimento di Informatica, Università di Torino, Italy

***Abstract*—In this work we analyse data collected from sensors installed on some vehicles of the local public transportation system in a European city. Our analysis is conducted by means of generation and application of Bayesian networks to describe the dependence relationships between variables and to predict the target variable of fuel consumption. We experimented with different algorithms that explore the search space of the possible alternatives guided by heuristics. We compare them with the results obtained with the technology of High Performance Computing, that allowed us to do an exhaustive search and find the optimal solution from the viewpoint of the likelihood evaluation measure. We solve the model evaluation and selection problem by application of an alternative evaluation measure: Granger causality. In addition we compared the predictive ability of the target by the obtained networks. Finally, we conducted "what-if" analysis under the form of intervention and counterfactual analysis and show which decisions policy makers and the service owners should afford to reduce costs and pollution.**

I. Introduction

In [1], Judea Pearl effectively summarizes the multiple facets and benefits we can obtain analysing data by constructing Bayesian networks (BN): (1) we obtain models that graphically are able to describe immediately the relationships between the variables; (2) can be applied to do feature selection (because we can omit the variables that are not connected to a target), can be applied to prediction of a target (by regression by means of the variables connected in the network to a target or by a Bayesian probabilistic model) and finally can be used to conduct reasoning by "what-if" analysis and counterfactual analysis, with the goal of improving the target in the future intervening on the model.

We follow these suggestions by analysing the dataset collected by a set of sensors installed on-board on the buses of the public transportation system of a big city in Europe, with the goal of reducing fuel consumption, air pollution and costs of the service.

The collected variables from a period of 6 consecutive months are the following: time reference, vehicle identifier, travelled $Distance$, altitude ($Height$), air-conditioning on ($Aircond$), seconds with accelerator pedal pressed ($Traction$), velocity, mass of the bus and people. By intervention on some of the variables we can try to reduce fuel consumption but first we need to discover on which of the variables the target (fuel) depends upon. We try to solve this by discoverying BN on collected data.

II. Bayesian Network by Heuristic and Exhaustive Exploration of the Search Space of the Hypothesis

In a linear Gaussian BN [2] the distribution of a variable (node) conditioning on its parents is a normal random variable such that the conditioning effect of each parent is modeled by an additive linear term in the mean.

Some of the collected continuous variables violate the assumptions of Gaussianity, especially $Dist$, $Height$, $Aircond$ and $Traction$. To circumvent this issue we can convert the continuous variables into discrete ones, modeled as multinomial data and represented as contingency tables.
For the discretisation we employ the Information-Preserving Discretisation algorithm [3].
Learning the structure of a BN is a complex task because the number of possible DAGs grows super exponentially as the number of nodes increases. Only a small fraction of its elements can be investigated in a reasonable time. Therefore, most of the structure learning algorithms deliver a sub-optimal solution by applying an heuristics (like Bayesian Information Criterion (BIC) [4]) and a greedy algorithm like Hill-climbing.
In the current project we take advantage of an parallel computing to find the optimal structure by means of a brute force algorithm. Each possible DAG is assigned a goodness-of-fit "network score" (BIC). The brute force algorithm returns the BN having the maximum score since larger values means better fit.

We learn the first two causal models under the assumption of Gaussianity from the training dataset including $Fuelperkm$, $Dist$, $Height$, $Aircond$, $Traction$, $Mass$ and Vel. One is delivered by the brute force algorithm mentioned above and the other one is built through the Hill-Climbing algorithm initializing the network with an empty structure. Both of them evaluate the goodness of fit using the BIC score. Other three causal models are learnt from binarized data, including the same seven variables as above. The first one is obtained through the brute force algorithm with BIC score. The other ones are delivered by the Hill-Climbing algorithm using different scores, such as BIC and K2 [4].

In order to compare the quality of the different models, we evaluate their predictive performances. For each network we averaged the 10-folds Cross Validation error over the predictions corresponding to each node, as reported in the table below. In other words, each entry in the table is computed as $\frac{\sum_{j=1}^{7} PE_j^M}{7}$, where PE_j^M is the prediction error of variable X_j according the model M and M is one of the five networks defined above. For each feature, the prediction is based on its parents according the specified model M, then we can exploit each network structure for fitting both a Gaussian and a Multinomial model, whether it was learnt from continuous or discrete variables.

As regards the regression with continuous variables, we used the Mean Squared Error $MSE = \frac{\sum_{i=1}^{n}(X_i^{true} - X_i^{predicted})^2}{n}$ and the Root Mean Squared Error $RMSE = \sqrt{\frac{\sum_{i=1}^{n}(X_i^{true} - X_i^{predicted})^2}{n}}$ in order to measure the error for each fold, where n is the number of instances in the training dataset.

	MSE (cont)	RMSE (cont)	Misclassification Rate (discr)
disc_ bf_bic	7645.70	57.00	0.18
disc_ hc_bic	45992.80	104.80	0.19
disc_ hc_k2	39325.40	98.60	0.24
cont_ bf_bic	6224.10	52.70	0.14
cont_ hc_bic	46182.90	105.40	0.19

TABLE I: Each row name specifies the data used to learn the network structure (discrete or continuous variables), the structure learning algorithm (Brute Force or Hill-Climbing) and the network score (BIC or K2)). Each column name specifies the error measure and the type of data used for prediction (discrete or continuous).

As regards the classification with binary variables, we used the $misclassification\ rate = \frac{count\ of\ misclassifications}{n}$ in order to measure the error for each fold. As we can see, the two networks provided by the brute force algorithm account for the lowest predictive errors.

III. Model Evaluation by Granger Causality

We perform model evaluation of the Bayes Networks by means of the statistical concept of Granger Causality which applies to the time series domain. In fact the analysed data set D can be arranged as a set of multivariate time series; each time-series D_{vd} is defined as a time-ordered collection of records of vehicle v in the working day d.
Given two stationary time series A and B we perform a Granger Causality Test in order to assess whether A has a predictive power in forecasting B. We compare the two models:

$$(AR) \quad B_t = \beta_0 + \beta_1 B_{t-1} + ... + \beta_q B_{t-q} + \epsilon_t \tag{1}$$

$$(VAR) \quad B_t = \beta_0 + \beta_1 B_{t-1} + ... + \beta_q B_{t-q} + \alpha_1 A_{t-1} + ... + \alpha_q A_{t-q} + \epsilon_t \tag{2}$$

The first equation (AR) is a univariate autoregression model of order q over B lagged values (with coefficients β_i and a noise error term ϵ_t), the second equation (VAR) is a vector autoregression model of order q with the lagged values of A (with coefficients α_j). The Granger Causality test is an F-test on the null hypothesis: $H_0 := \{\alpha_j = 0; j = 1, .., q\}$ [5]. Each time series involved in the Granger Test must satisfy the stationarity condition, in our work we evaluate it with Augmented Dickey-Fuller test (ADF) [6] and apply iteratively first-differencing over the time-series until the stationarity condition is satisfied. Given the multivariate time series D_{vd} we conduct a series of Granger tests over all possible ordered pairs of variables in S and aggregate the results for all pairs into a matrix G_{vd}. We define the Granger matrix G as the average of the G_{vd} matrices computed over the set of multivariate time series. Thus the entry $G[i, j]$ is the success rate of the Granger test between S_i and S_j over the multivariate time series data set.
We perform a model evaluation over the different Bayes Networks computed so far. We compute a Granger matrix for each type of data set. Given a Bayes Network $\mathcal{B}$ we define its corresponding adjacency matrix as $A^{\mathcal{B}}$ we compute the euclidean distance of the Granger matrix G from $A^{\mathcal{B}}$ as $dist(G, A^{\mathcal{B}}) = \sqrt{\sum_{i,j=1}^{7}(G_{ij} - A_{ij}^{\mathcal{B}})^2}$.
The following table displays the distance $dist(G, A^{\mathcal{B}})$ for each considered BN:

TABLE II: $dist(G, A^{\mathcal{B}})$: each $\mathcal{B}$ is identified by: **Data** (columns), **Score** (rows), **Algorithm** (displayed within the entries; *BF* = Brute Force, *HC* = Hill Climbing)

$dist(G, A^{\mathcal{B}})$	**Data**	
Score	discrete	continuous
BIC	BF: 3.11	BF: 3.39
	HC: 3.30	HC: 3.43
K2	HC: 3.05	

$dist(A^{\mathcal{B}}, G)$ is used as a mean of comparison over the BNs, as distance decreases we have an higher compatibility between the causal relationships found by Granger tests and the edges of the BN. We observe that, considering the networks computed with the *BIC* score, the networks computed with the Brute Force algorithm have a lower distance with respect to those computed with the Hill Climbing algorithm: this result upholds the efficiency of the proposed *BF* algorithm.

IV. Application of the Discovered Knowledge

Let us consider the Gaussian BN learnt from the original continuous variables through the brute force algorithm. As explained in [7], since the model is linear, we have to regress the target Y on X and on all of the covariates satisfying the so called back-door criterion from X to Y. The coefficient of X is the desired causal effect.

TABLE III: Causal effect on Fuelperkm for Gaussian BN

	adjustment set	C.E.	structural parameters
Dist	{ Vel Traction }	0.081	0.096
Height	{ Vel Dist }	4.581	5.239
Vel	{}	16.42	12.271
Traction	{}	-0.516	-1.531
Aircond	{ Vel Traction }	0.352	0.361

Concerning the Gaussian BN, the table above shows that each parent of $Fuelperkm$ has a causal effect that is similar to the structural parameters of the linear regression

$Fuelperkm \sim Dist + Height + Vel + Traction + Aircond + const$) (coeffs in last column). The causal effect of $Traction$ on $Fuelperkm$ is negative: the reason is that traction time increases with distance ($corr(Dist, Traction) = 0.929$) and a long travel is likely to be covered mainly on straight roads with steady speed and reduced brakes usage. Moreover, Vel accounts for the strongest casual effect on $Fuelperkm$.

Conclusion: we can say that we were able to confirm the main causal relationships between fuel, traction and velocity, both in discretised and continuous data.

References

[1] J. Pearl, "The seven tools of causal inference, with reflections on machine learning," *Commun. ACM*, vol. 62, no. 3, p. 54âĂŞ60, Feb. 2019.

[2] D. Koller and N. Friedman, "Probabilistic graphical models - principles and techniques," 2009.

[3] A. J. Hartemink and D. K. Gifford, "Principled computational methods for the validation and discovery of genetic regulatory networks," Ph.D. dissertation, USA, 2001, aAI0803410.

[4] A. M. Carvalho, "Scoring functions for learning bayesian networks inesc-id tec. rep. 54/2009 apr 2009," 2009.

[5] C. W. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica: journal of the Econometric Society*, pp. 424–438, 1969.

[6] D. A. Dickey and W. A. Fuller, "Distribution of the estimators for autoregressive time series with a unit root," *Journal of the American statistical association*, vol. 74, no. 366a, pp. 427–431, 1979.

[7] B. Chen and J. Pearl, "Graphical tools for linear structural equation modeling," 2014.