Contents

Document history

Version	Date	Author (Unit)	Description
v0.3	1/9 2014		First draft finished

1 Executive summary

2 Preliminary Models

2.1 Daimler Models

Introduction

The basic settings of the DAIMLER's use-case [][] are as follows. Let us suppose we are driving our car, which will be referred as the EGO vehicle, in a highway. This EGO vehicle is equipped with a video camera, radar and some on-board sensors. Using the data provided by these sensors, the problem consists in the early recognition a maneuver either of the EGO or another relevant car in the traffic scene. In total, the system is expected to recognized the following set of manoeuvres (a visual description of them is given below in Figure 1):

1. **Object-CutOut**: A vehicle that was driving in front of us is leaving the EGO lane.
2. **Object-CutIn**: A vehicle is moving to the lane where the EGO vehicle is placed.
3. **EGO-CutOut**: The EGO vehicle is leaving the lane where it was driving.
4. **EGO-CutIn**: The EGO vehicle is moving to a new lane already occupied for another vehicle.
5. **Object-Follow**: There is no lane change. The EGO is driving and there is some other vehicle in front.
6. **Lane-Follow**: There is no lane change. The EGO is driving and there is not any other vehicle in front.

The data used to address this problem do not contain raw data from the video, radar and on-board sensors. The manoeuvre recognition system directly works with the so-called "object data", which contains "high level" representations or features describing the "traffic scene" such as EGO's speed, distance between EGO and another vehicle in front, etc.

Figure 2 contains a visual description of the current data flow used to create this "object data". As can be seen in this figure, in a first step the raw data coming for the the video, radar and sensors is preprocessed. In a second step this preprocessed data is fused and the high-level or "object data" describing the traffic scene is obtained.

Using this "object data", Daimler has developed a probabilistic graphical model [] which is able to recognize an ongoing manoeuvre around 0.6 seconds before the manoeuvre really takes place. This probabilistic approach is based on modelling the problem in different layers as shown in Figure 3.

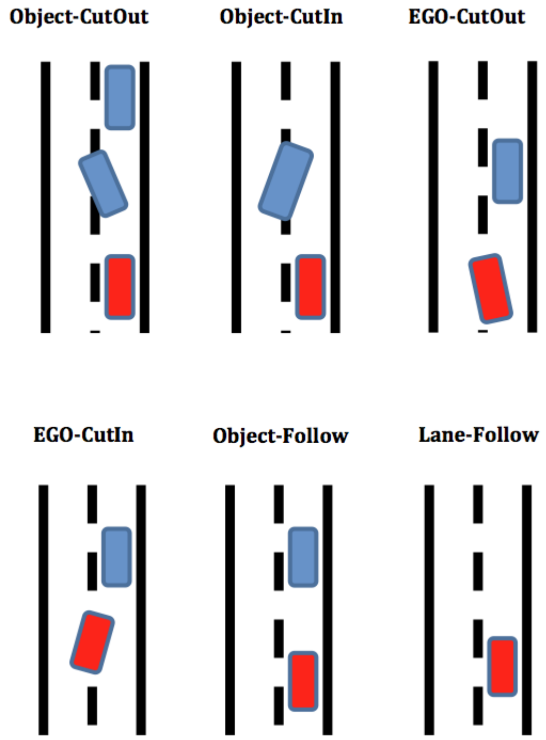

Figure 1: Different maneuvers which should be identified by the AMIDST system. Red blocks represents the EGO vehicle and blue blocks represents other vehicles in the scene. In the first four maneuvers, there is a lane change event or, under Daimler's terminology, a "Lane Marking Crossing" (LMC) event.

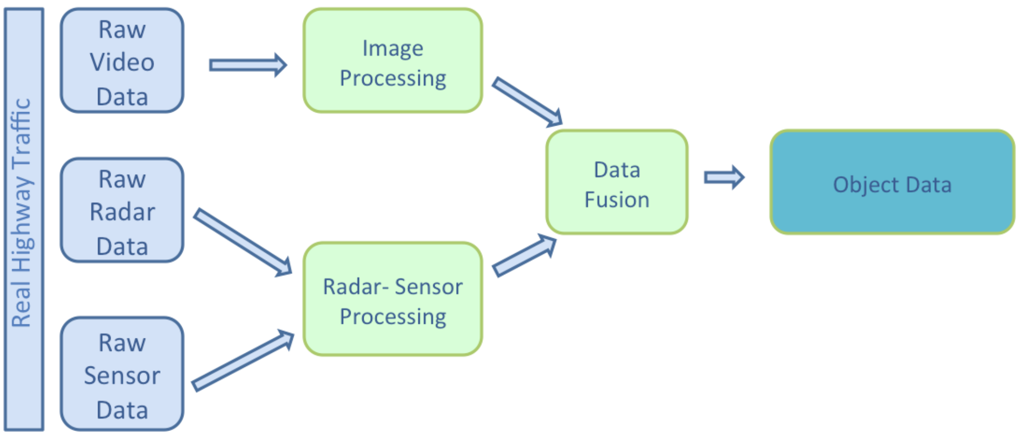

Figure 2: Daimler's Data Flow.

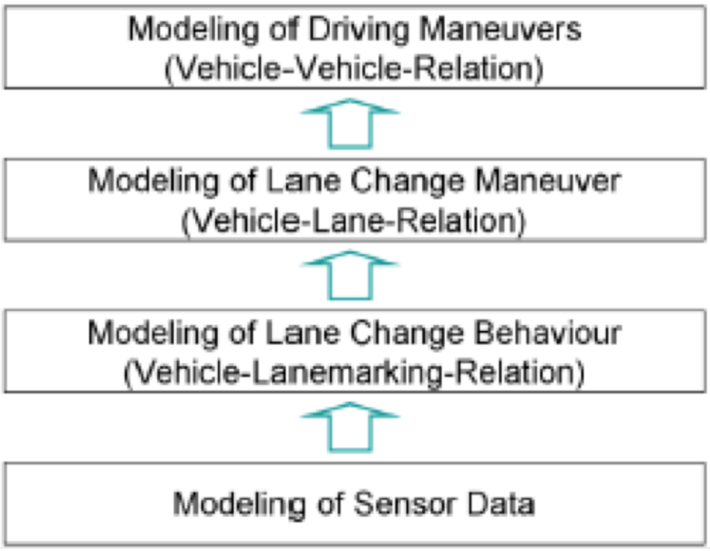

Figure 3: Hierarchical layers for the recognition of driving manoeuvres.

In a first step it is only modelled the sensor data. Using this layer, a new layer is created on top with the goal of detecting a lane change behaviour. The detection of a lane change behaviour allows the system to model the lane change manoeuvre in a higher layer. Finally, with this information, the system is able to identify the kind of driving manoeuvre which is taking place between a pair of vehicles.

The static-OOBN model

As commented above, this model will work with the so-called "object data". This data mainly consists on a set of measured and/or computed signals or situation-features denoted by S (e.g.. EGO speed, EGO lateral velocity, speed of a car in-front, etc., see [] for further details) describing the traffic scene. And the whole modelling is structured in hierarchical layers as detailed in Figure 3. This hierarchical modelling was previously implemented in [][] using an object-oriented Bayesian network (OOBN) [].

The general structure of this OOBN model consists of a number of abstraction levels (see Figure 4): all measured and/or computed signals S are handled with their uncertainties σ^2. These are represented as object classes at the lowest level (class S) of the OOBN. The real values μ of evidence signals are then used at the next level of hierarchy to evaluate the hypotheses (class H in Figure 4). The combined evaluation of several hypotheses results in the prediction of events, class E. In our case, the events are modelling traffic maneuvers of the own and neighbour vehicles.

As commented above, the observations characterizing a situation are acquired from sen-

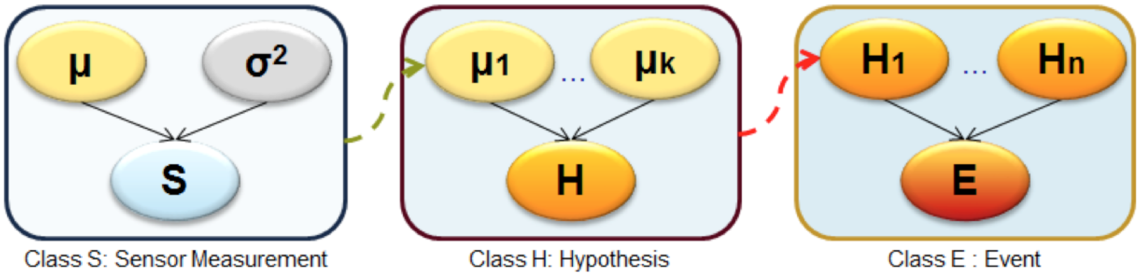

Figure 4: Static-OOBN model for the prediction of an event (maneuver).

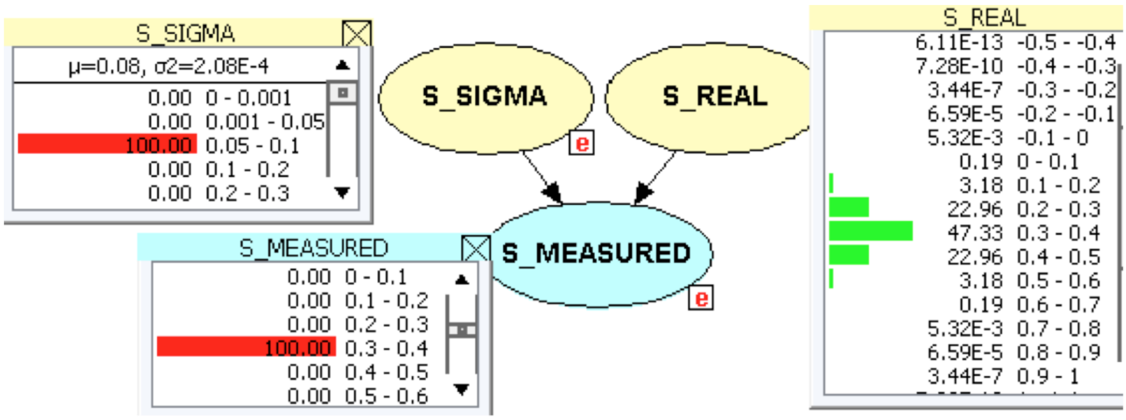

Figure 5: BN fragment for modeling of sensors uncertainties with a discrete "measurement" variable.

sors and computations (see Figure 2) and, in consequence, are *measured data*. If the measurement instrument is not functioning properly (due to sensor noise or fault), then the sensor-reading ($S\_MEASURED$) and the real variable (S_REAL) under measurement need not to be the same. This fact imposes the causal model structure as shown in Figure 5. The sensor-reading of any measured variable is conditionally dependent on random changes in two variables: real value under measurement (S_REAL) and sensor fault (S_SIGMA).

The situation features used for maneuver recognition are structured along three main dimensions: lateral evidence (LE), trajectory (TRAJ), and occupancy schedule grid (OCCGRID). They represent the three hypotheses (see Figure 4), which are modelled by the corresponding OOBN-fragments. For more details see [13], [14]. The hypothesis LE is shown in Figure 6. Its conditional probability distribution is represented by a sigmoid (logistic) function to expresses the growing probability for the lateral evidence on crossing the lane marking, when the vehicle is coming closer to the lane

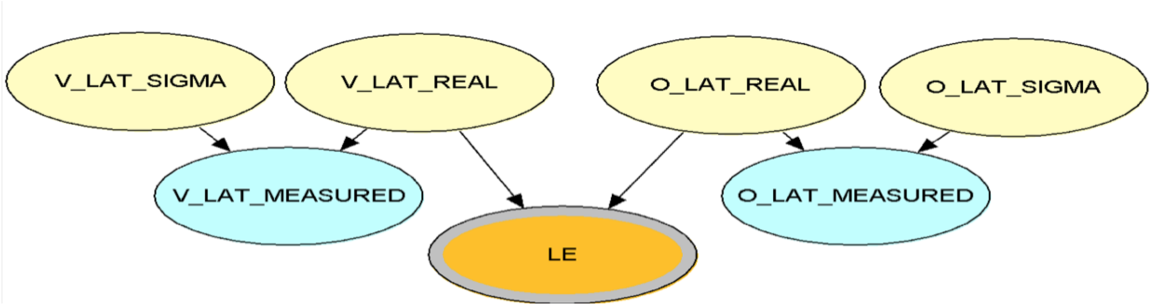

Figure 6: BN fragment for modeling of sensors uncertainties with a discrete "measurement" variable.

marking (modeled by $O\_LAT\_MEASURED$) by growing lateral velocity (modeled by $V\_LAT\_MEASURED$).

Figure 4 abstractly shows how these hypotheses are combined into events, which in our automotive scenario correspond to the different driving maneuvers: lane follow, lane change (cut-in, cut-out), expressed for ego and surrounding objects, see [12], [13].

The dynamic-OOBN model

The above described static OOBN is able to detect a maneuver 0.6s before execution. The goal is to extend the prediction horizon for manoeuvre recognition at least to 1-2 seconds (max. 4-5 seconds ahead) before the actual lane marking crossing, which is of advantage for the adaptive cruise control. Most precisely, and as indicated in the Use Case 8 on the Requirement Analysis, the area under the ROC curve (AUC) should be greater than 0.96 for 1 second and greater than 0.9 for 2 seconds.

Figure 7 shows the evolution on time for lateral velocity and offset in an object Follow (OF) → OBJ_Cutout and Lane Follow(LF) → OBJ_Cutin manoeuvres. The black line corresponds to the values massured from the point of view of the EGO car (left side) and the green line to the measures from the point of view of the right part of the OBJ car. The vertical bar indicates the moment in which the manoeuvre has been recognised by the static OOBN. By taking the temporal properties of the data into account on the model, we should be able to predict the manoeuvre earlier on time. Ideally, the manoeuvre should be detected as soon as the lateral offset and velocity for the OBJ start to increase.

Each manoeuvre can be considered as a process, developing in time, i.e., as data stream given by a time sequence of the transition from lane follow into lane change manoeuvre. The dynamic extension involves copies of the static OOBN for different number of time steps in the time window (e.g. see Fig. 8 where the two top nodes are temporal clone defining the share belief state between consecutive time steps creating a first order

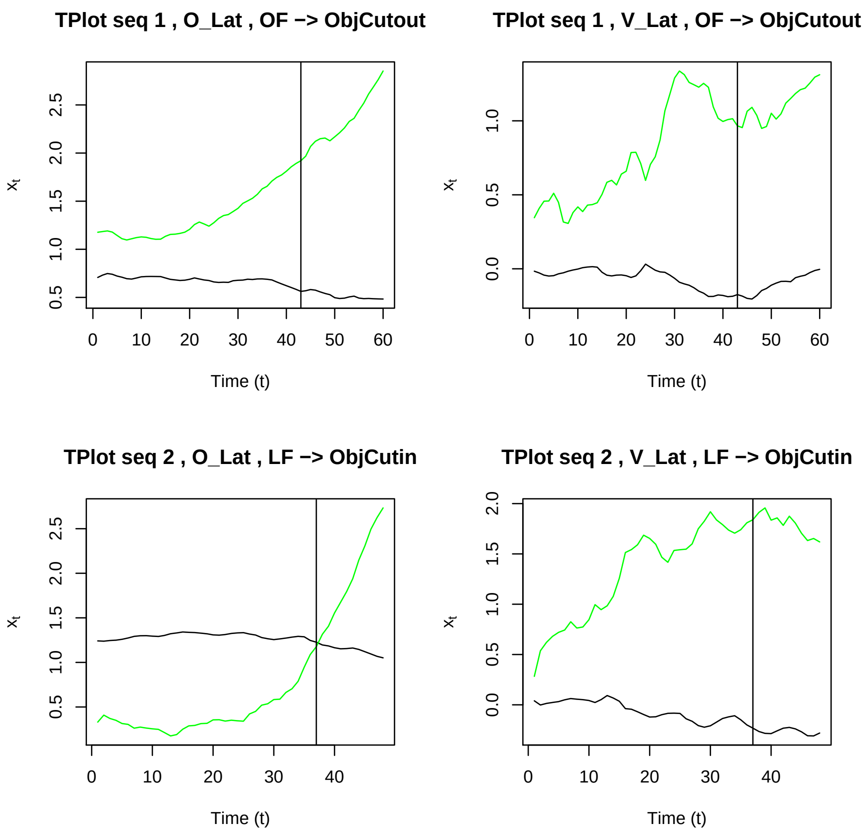

Figure 7: Daimler Time Plot

Markov process), if also the requirement on earlier prognostics of maneuver is to be satisfied.

A good starting point to model the dynamics of the data involves the variables that capture the lateral evidence for the different vehicles, given its relevance and simplicity. The dynamic BN (DBN) incorporates the trend of change for the real values, where their physics relations are represented as causal dependencies between the time steps dt, e.g. in Fig. 8 the transition function of O_LAT at time t, $O(t)$, is modeled as a Gaussian distribution. Its mean is affected by $O(t-1)$, and by V_LAT at time $t-1$, $v(t-1)$:

$$O(t) = O(t-1) + v(t-1)dt + N \tag{1}$$

where N denotes a white noise $N(0, \sigma^2)$ which is assumed to be small.

The shaded nodes represent the development of the real values of observations over several time steps in the time window. Thus, their trend estimation contributes to the

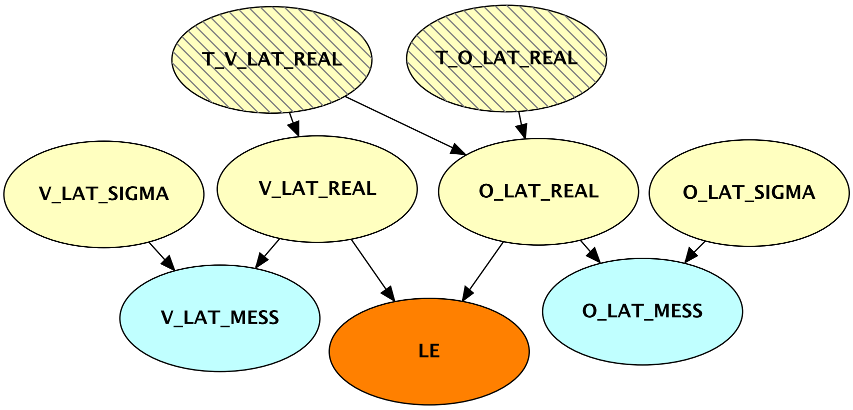

Figure 8: Daimler Temporal Model

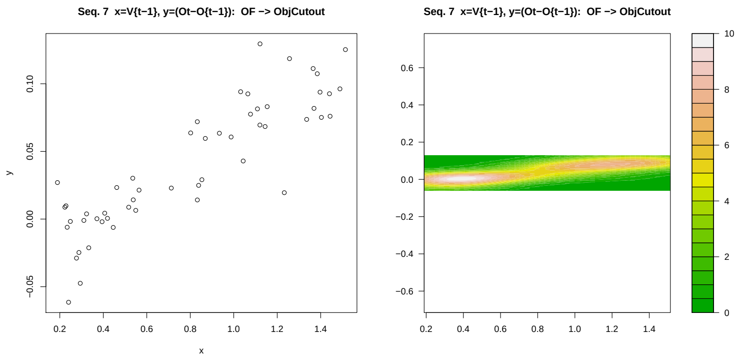

Figure 9: Time plot for $v(t-1)$ vs $O(t) - O(t-1)$. Linear correlation can be observed.

prediction of probability of transition from a lane follow to a lane change manoeuvre.

In order to corroborate the validity of this BN fragment, we have analysed the hypothesis $\Delta O = O(t) - O(t-1) = v(t-1)dt + N$ on our data. Figure 9 shows the plot and contour plots for v and ΔO, where we observe a linear correlation.

A DBN induces a number of constraints on the compilation of the network into a computational structure. One constraint relates to transferring the belief state from one time slice to the next where the belief state is the probability distribution over the variables shared by neighbouring time slices. In general, the belief state is transferred as a joint distribution. We have imposed limitations in our dynamic model so that the next state depends only on the current state, and not on the sequence of events that preceded it, i.e. first order Markov model. Although in principle this might seem as a strong limitation, we have observed this property to hold in our data. Fig. 10 displays, at the top, the sample correlograms for lateral velocity and offset, that is, the correlation of the

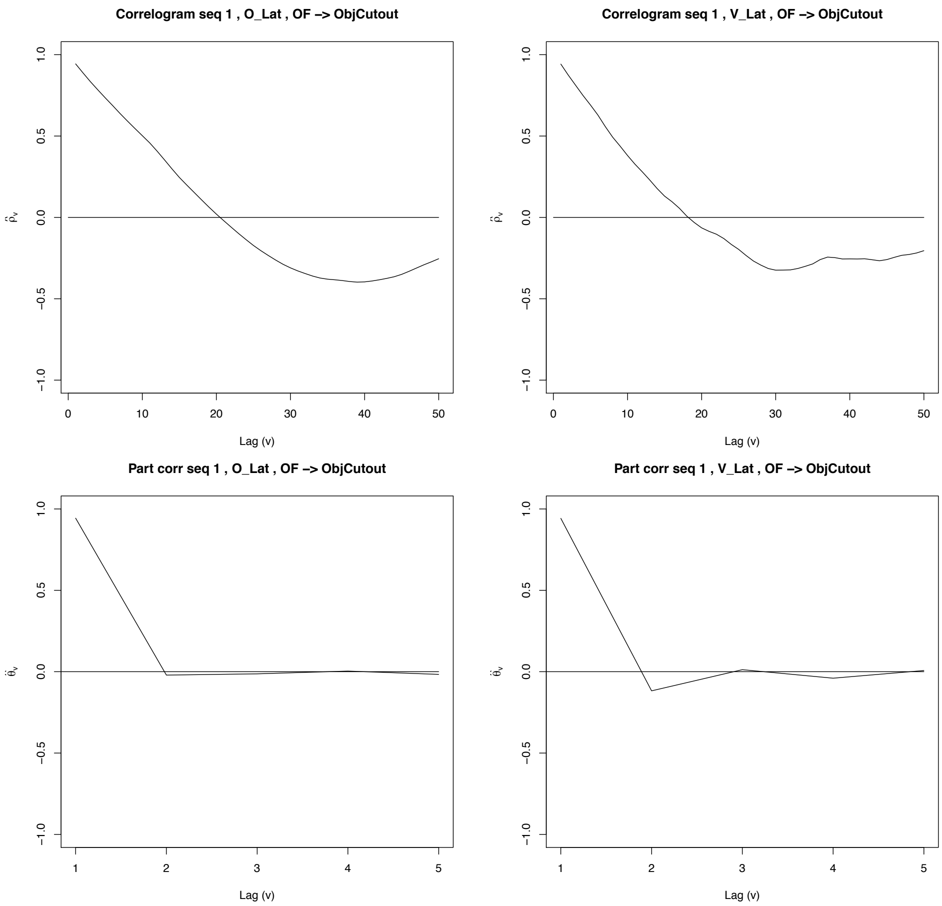

Figure 10: Correlograms and partial correlograms for Lateral velocity and offset.

data with lagged values of themselfves. The partial correlogram (bottom figures) is used to remove the common linear effect of the data in between samples. In our example, for both variables, the correolograms take some time to decay to zero, while the partial correlograms are large for lag one and then small of all other lags. This indicates that the correlation between non consecutive samples is due to the common relationship of these samples and the samples in between [**?**].

2.2 CajaMar Models

Introduction

There are two tasks to be solved for Cajamar's use case. The main one is the estimation of the *probability of default*, defined as the probability that a credit operation will end up to a default in two years. The other task is to obtain good customers profiles in terms of risk so that marketing campaigns can be specifically directed to low risk customers.

Predicting probability of default

The main task is predicting the probability that an operation will result in a default **2 years before** it actually happens. It is therefore a *supervised classification* problem, currently solved using *logistic regression* over *27 predictors*. These predictor variables (many of them manually built by CajaMar's experts) describes the financial behaviour of the customers in the last 180 days, which is a limit imposed by the Bank of Spain. The provided database is *imbalanced* -around 10% of defaulters-.

We will initially consider two possible approaches.

Static approach

The static scheme would consist of the following steps:

- Construct a single flat table, containing information on time windows of *180 days*.
- Build a BN classifier (NB or TAN). Presumably, a **TAN**.
- Update risk profiles using the classifier.

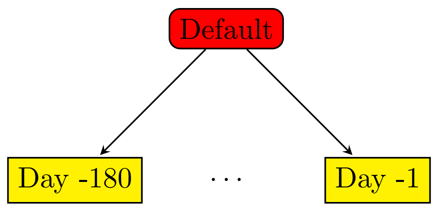

Figure 11: The static model. Each yellow box represents a set of variables measures during the same day. The variables within a box can be connected (according to a tree structure), globally conforming a TAN

Dynamic approach

Figure 12 represents the global idea of the temporal model. It can be compactly represented by a dynamic Bayesian network made of components as the one displayed in Figure 13. D_t represents the class variable at time slice t. Each feature variable at time t, denoted as X_t, is linked to the same variable at time t_1: X_{t-1} as well as to a *memory variable* $\bar{X}_t$ that represents the average value of X during the last 180 time slices (days). Finally, an indicator variable δ_{X_t} may be included if the variable is such that is observed many times at point 0. This is the case as, for instance, payments by credit card, where many of the days can be equal to zero for most of the customers.

- Construct *1 table* for each day
- Build a *dynamic* BN classifier (NB or TAN)
- Update risk profiles using the classifier

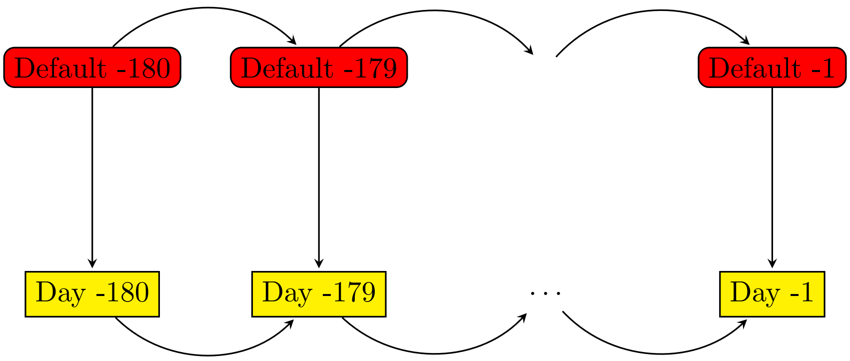

Figure 12: Global structure of the temporal model.

Model Structure

From a probabilistic modelling point of view, Caja-Mar faces two different problems [][]: the prediction of the risk of defaulting of a customer in the next two years; and the extraction of profiles of "desirable" prospective customers.

The risk prediction problem has been modelled as supervised dynamic prediction problem. We are given a data base with a set of variables or predictors (some of them manually built by CajaMar's experts) describing the financial behaviour of the customers and, also, whether the customer is considered as defaulter and non defaulter according to CajaMar standards (i.e. a binary class variable). The dynamic component of the problem needs to be considered because the behaviour of the customers evolves over time (e.g. the account balance is continuously changing from month to another, the level of incomes, etc.) as well as the labelling as defaulter or non-defaulter customer (e.g. one customer can be creditworthy and, but after some time, be in bankrupt for becoming unemployed). More specifically, the proposed model is expected to answer the following question: which is the probability that this customer will default in some of his/her loans in two years? And this prediction has to be made only using the customer's behavior in the last 180 days [1].

The graphical structure of the dynamic probabilistic graphical model devised for this problem is given in Figure **??**. The yellow square boxes "Day -180", ..., "Day-1" represents the temporal evolution of the predictor variables. The model only refers to 180

[1]This limit is imposed by the Bank of Spain.

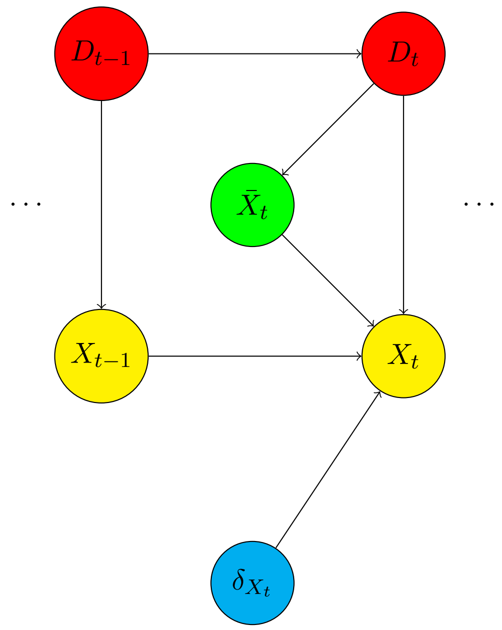

Figure 13: Basic component of the structure of the dynamic model.

days because this is the imposed limit of days when making predictions. Similarly, the class variable "default" is assumed to evolve over time but with the relevant different that the default class sequence refers to a point in the time **two years later** than the point in the time the daily predictor variables.

Data Analysis