L3: ML-based estimation of occupancy patterns

Pablo Santana González

***Abstract*— For the problem of estimating occupancy levels in a room, a maximum likelihood estimator is used for identifying an ARX model for the CO_2 levels in a room. The likelihood function of the CO_2 level is derived as the likelihood of the noise present in the measurement for the parameter θ, from which the maximum likelihood estimator $\hat{\theta}_{ML}$ is computed by using numerical constrained optimization tools for a training dataset. Then, the ARX model is used to obtain a prediction of the occupancy for a testing dataset. The performance of the model is then evaluated and deemed satisfactory. The most significant learning outcomes were the derivation and understanding of the likelihood function, the computing of the maximum likelihood estimator for a particular set of data and its application to a real world problem, where the model for the desired predicted magnitude was not explicitly provided.**

I. Results

A. *Deriving the likelihood $L(\theta; Y)$*

We lay our ARX model for CO_2 content on an hypothetical room as shown in equation 1, where y stands for the CO_2 levels, u for the air-conditioning power, o for occupancy and e for the system noise, defined as Gaussian noise $N(0, \sigma^2)$, its variance σ^2 unknown. The identification parameter of this model is then the vector $\theta = [a, b_u, b_o, \sigma^2]$, for which a maximum likelihood (ML) estimator $\hat{\theta}$ is to be found. This will allow for the prediction of the occupancy from this model.

$$y(t) = ay(t-1) + b_u u(t-1) + b_o o(t-1) + e(t) \quad (1)$$

To lay out the maximum likelihood problem and, specifically, derive the ML estimator we define the following random variables (RVs) and their realizations:

- Y_1: RV representing to the real CO_2 levels, which can be assumed to follow the ARX model in eq. 1.
- E: RV modelling the Gaussian noise $E = N(0, \sigma^2)$ present in the RV Y_1.
- Y_2: noiseless RV derived from Y_1 so that $Y_1 = Y_2 + E$.
- $y_1(t)$, $y_2(t)$, $e(t)$: realizations of the RVs above. We can also name these as random vectors $\mathbf{y_1}$, $\mathbf{y_2}$ and $\mathbf{e}$.

The problem is now identifying the set of parameters θ that best fit the available data for the CO_2 levels. Thus it is desired to find the maximum likelihood estimator $\hat{\theta}$ that maximizes the likelihood function $L(\theta; Y)$, which is now to be derived.

The available measurement of CO_2 is a realization of the RV Y_1, for it is a real realization subjected to white noise. The likelihood function can be written as the probability that, for a given set of parameters θ, the RV equals a certain realization, as shown in equation 2.

$$L(\theta; Y_1) = P(\mathbf{Y_1} = \mathbf{y_1}; \theta) \quad (2)$$

This probability could be expressed using the probability density function of the RV, which sadly we do not have for Y_1. However, because we can construct the RV Y_2 from the expression $y_2(t) = ay_1(t-1) + b_u u(t-1) + b_o o(t-1)$ we could easily build a realization of it that matches our data, which would consequently lead to a realization of the RV E, $e = y_1 - y_2$. By the procedure shown in equation 3 we can reach a probabilty expression that can be expressed by an obtainable pdf.

$$\begin{aligned} P(\mathbf{Y_1} = \mathbf{y_1}; \theta) &= P(\mathbf{Y_2} + \mathbf{E} = \mathbf{y_1}; \theta) \\ &= P(\mathbf{E} = \mathbf{y_1} - \mathbf{Y_2}; \theta) \\ &= P(\mathbf{E} = \mathbf{e}; \theta) \end{aligned} \quad (3)$$

Since we know that E is a RV representing Gaussian noise, we can build its pdf from the theoretical expression of the normal distribution. To generalize this to the random vector form, we assume that the white noise realization is independent and identically distributed (i.i.d.). This leads to the pdf expression for a normal random vector shown in equation 4.

$$\begin{aligned} p(e) &= \frac{1}{\sqrt{2\pi\sigma^2}} * \exp\left[-\frac{1}{2}\left(\frac{e_i - \mu_e}{\sigma}\right)^2\right] \\ p(\mathbf{e}) &= \prod_{i=1}^{N} p(e_i; \theta) \\ p(\mathbf{e}) &= \frac{1}{(2\pi\sigma^2)^{N/2}} * \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{N}(e_i - \mu_e)^2\right] \end{aligned} \quad (4)$$

Combining equations 1-4 we can define the likelihood as $L(\theta; \mathbf{Y_1}) = L(\theta; \mathbf{E})$, thus arriving to the set of expressions shown in equation 5, which allow for the calculation of the ML estimator $\hat{\theta}$ from the available measurements of $y_1(t)$, $u(t)$ and $o(t)$.

$$\begin{aligned} &\theta = [a, b_u, b_o, \sigma^2] \\ &y_2(t) = \theta_1 y_1(t-1) + \theta_2 u(t-1) + \theta_3 o(t-1) \\ &e(t) = y_1(t) - y_2(t) \\ &L(\theta; \mathbf{E}) = \frac{1}{(2\pi\theta_4)^{N/2}} * \exp\left[-\frac{1}{2\theta_4}\sum_{i=1}^{N}(e_i - \mu_e)^2\right] \\ &\hat{\theta} = argmax_{\theta c \Theta} L(\theta; \mathbf{E}) \end{aligned} \quad (5)$$

B. Computation of the ML estimator $\hat{\theta}$

We can further simplify the likelihood expression by calculating the $argmin$ of the negative log-likelihood, so that $\hat{\theta} = argmin_{\theta c \Theta} - log\left[L(\theta; \mathbf{E})\right]$.

Note that the domain of the ML estimator $\hat{\theta}$ is restricted, because not all values of the parameters may return an stable model of the system even if the ML condition is satisfied. From the ARX model we can make a good guess on the θ restrictions:

- θ_1: The higher CO_2 levels in the immediate past the higher they will be in the present. Also, if ventilation is off and there is no occupancy it makes no sense that the CO_2 levels increase spontaneously. Must be positive and smaller/equal than 1.
- θ_2: The higher the ventilation the more clean air concentration thus lower the CO_2's. Must be negative.
- θ_3: The more people in the room the more CO_2 in the air due to people breathing it out. Must be positive.
- θ_4: Which is the noise variance, then it must be positive.

Thus, the problem ahead is the constrained minimization of the negative log-likelihood of θ. This can be done in $MATLAB$ through the function $fmincon()$, which solves for the result shown in the expression 6.

$$\hat{\theta}_{\mathbf{ML}} = [0.980, -0.250, 2.99, 16.7] \tag{6}$$

C. Making predictions on test data

Once $\hat{\theta}$ has been obtained, predictions can be made on the test dataset by reshaping the ARX model to the structure shown in equation 7.

$$o(t-1) = \frac{\theta_1 y(t-1) + \theta_2 u(t-1) - y(t)}{\theta_3} \tag{7}$$

Because the output does not really make sense right after this step, it will now undergo a series of non-linear operations:

- Bartlett smoothing: a triangular moving average (MA) to soften the spiky occupation.
- Rectified linear unit (relu): so that any negative value turns to zero. Positive numbers stay unmodified.
- Rounding: to convert the occupation into an integer.

The processed predicted occupancy, as well as the real occupancy and the error of the prediction over time, are shown in Figure 1.

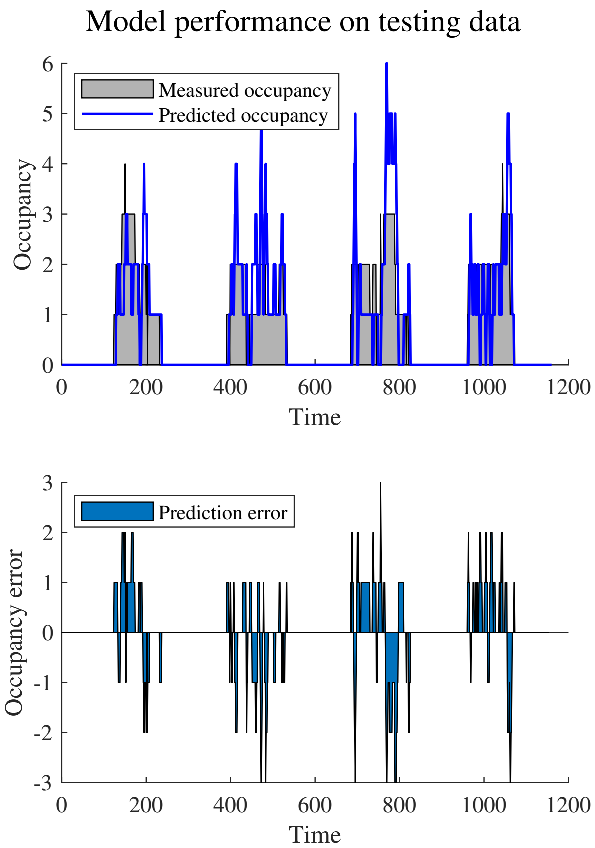

Fig. 1: Prediction on test data. Above, time series of measured occupancy and predicted. Below, time series of the prediction error.

II. Conclusions

The ML estimator $\hat{\theta}_{ML}$ was computed by using numerical methods to identify an ARX model for the CO_2 levels. Once this model was identified, it is then used as a tool for the estimation of occupancy. The computing of the likelihood function required the transformation of RVs to find an expression for a probability with both a pdf and representative data available.

The resulting model is able to predict the occupancy decently, however its performance on transients is not great: it can be seen that when the real occupancy changes rapidly the model under performs. On the other hand, slower regimes show much better performances: when the occupancy is held at zero the model is more accurate. The prediction errors are usually short in time and related to quick occupancy changes when the occupancy is different to zero, so it could be said that the prediction model looks unbiased. However, due to the non-linear operations performed on the model output, it is hard to accurately evaluate the predictor's properties.

For the real situation, just predicting when the room is empty is quite significant already, because it allows for energy savings when there is no people that could be in charge of the air-conditioning/light/etc. Also, accounting for people quickly entering and exiting the room not only is harder because their impact on the environment is smaller, but it could even be argued that there is really no need to account for their presence at all if the time they spend on the room is very small.

Overall, it is concluded that the model performs in an acceptable manner, however a better performance on quick transients would be desirable.

Acknowledgments: Thanks to Martin and Robert.