# eMOTIONAL Cities

Mapping the cities through the senses of those who make them

DELIVERABLE 7.2

# Metamodeling library I

## WP7 – Scenario Discovery

15.11.2023

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement nº945307. This document reflects only the author's view and the Commission is not responsible for any use that may be made of the information it contains.

| | |
|---|---|
| Project Title | *e*MOTIONAL Cities: mapping the cities through the senses of those who make them |
| Deliverable | D7.2 – Metamodeling Library I |
| Work package | WP7 – Scenario Discovery |
| Task | T7.4 – Metamodeling Library |
| Number of pages | |
| Dissemination level | Public |
| Leader | DTU |
| Contributors and peer-reviewers | FMUL, and NGR |
| Date | 07/12/2023 |
| File name | eMCities_WP7_D7.2 |
| Version | V3 |
| Authorship | Lorena Torres Lahoz (DTU), Carlos Lima Azeveo (DTU), Francisco Pereira (DTU), |

# Index

# Index of Figures

# Executive Summary

This document is the deliverable "D7.2: Metamodeling Library" that belongs to the "WP7: Scenario Discovery" of the European project "eMOTIONAL Cities: Mapping the cities through the senses of those who make them" (Project Number 945307; Project Acronym eMOTIONAL Cities).

This report relies mainly on activities from T7.4 "Metamodeling Library", T7.1 "Methodological Foundations", and T7.2 "Specification of Case Studies with cost-effectiveness Analysis" within WP7 but also from the modelling exercises ongoing in WP5 and WP6. This document presents the general metamodel methodology, their state-of-the-art and research gaps. It also describes the preliminary version of the metamodeling methodology for this project. The provided version herein should be revisited as the project evolves and the results are iteratively obtained.

This library will then be progressively enriched and refined based on the results of the case studies, leading to a revised, final version that will be delivered at the end of the project.

## Terminology and Acronyms

| Term | Acronyms |
| --- | --- |
| **GP** | Gaussian Process |
| **KPI** | Key Performance Indicator |
| **RF** | Random Forest |
| **AL** | Active Learning |
| **LHS** | Latin Hypercube Sampling |

# 1. Simulation Metamodels

## 1.1 Background

Simulation models tend to be complex and time-consuming. In this document, we address these issues from the perspective of the eMOTIONAL Cities project, by providing a metamodeling library that can approximate the simulation models provided in (Lima Azevedo, Miranda , Camara Pereira, Torres Lahoz, & Ancora, 2023), namely, the Daily Activities & Mobility – Stress model for Copenhagen, the Walking – Stress model for London and the Brain Process – Emotional Response model for Lisbon.

Even though the increase in computational power over the last decades has allowed for faster computations, simulation models have generally become more time-consuming particularly in terms of running times, given their improvement in realism and details.

Long simulation computational times can cause bottleneck problems when we need to run our simulation model multiple times, like in the Case of Scenario Discovery.

Simulation metamodels (Friedman, 2012) have existed since the early ’70s (Barton, 1998). A simulation metamodel is just a model of a model. One purpose of metamodels is to serve as surrogates, in other words, approximations of a more complex and time-consuming model that can be faster and more effectively computed. This document may interchange the terms’ simulation metamodel’ and ‘metamodel.’

If a simulation model approximates the real-world system, then a metamodel can be regarded as an approximation of the simulation model itself, as depicted in Figure 1.

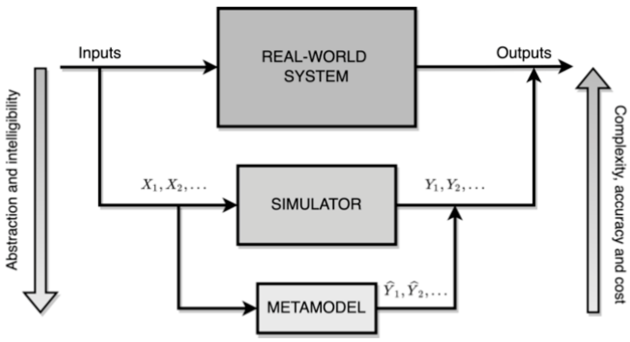


**Figure 1. Relationship between the real-world system under study, the simulation and the metamodel (Antunes, Camara, & Riss, 2022)**

The four primary purposes of simulation metamodeling are problem entity understanding, simulation output prediction, optimisation, and verification/validation. [5].

Metamodels can serve as analytics tools to explore and better understand the relationship or causality between some inputs and outputs of interest since when the simulation model is too complex, these relationships cannot be formulated in a closed-form expression. Metamodels can help understand the underlying input/output dependency by capturing the ‘emergent behaviour’ that appears during a complex simulation model’s inner interactions and dynamics.

Our project focuses more on the simulation output prediction aspect since we aim to compute the metamodel for many different starting points to explore the input space of parameters for the simulator, as well as considering uncertainty and stochasticity across this input space. Therefore, we aim to fit a metamodel for accurately predicting the simulation model’s output to replace it for the simulation model when multiple runs should be implemented. For example, when randomness in the model is such that, for the same input configuration, one potentially gets very different outputs (i.e. the output is stochastic), or when the output space is non-linear and therefore one cannot optimize without evaluating multiple points.

Traditionally, the metamodeling methodology comprises three steps: the definition of the experimental design, the metamodel specification, and the learning and fitting of the metamodel.

The experiment design requires the definition of the input/output variables whose relationships we would like to explore, the required KPIs and the assumptions and limitations of the proposed formulation.

Due to the short computational time requirement, metamodels are usually characterised by simple and functional formulations. For the metamodel specification, Gaussian processes or Decision Trees are the more commonly used metamodels, given their straightforward underlying formulation and their faster computing time compared to more complex models.

Lastly, fitting a metamodel can be seen as a trade-off between computational speed and reasonable accuracy loss, which should be defined according to the modelling goals. Even though the training requires unavoidable computational effort, for both sampling the initial data from the simulator and training the metamodel, this effort is rewarded when the metamodel computation time is much lower than the simulation model that it mimics.

Training a metamodel is finding the value of the parameters that define the relationship between inputs/outputs within its formulation, providing a minor loss between the simulation outputs and the metamodel computed ones.

Data should be divided between training and prediction sets for training a metamodel. The first set comprises labelled input points generated by independent simulation runs. On the other hand, the prediction set comprises input data without labels whose output results from the metamodel computation, called the output prediction samples.

Figure 2 highlights the relationships between the problem entity, the simulation model and the simulation metamodel. Even though they all serve different purposes, clear validation criteria, such as calibration, must be established between the different entities to ensure their validity.

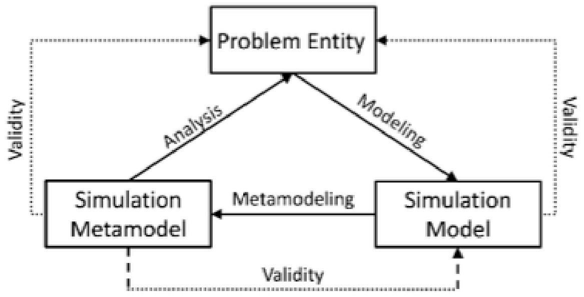


**Figure 2. Relationship between problem entity, simulation model, and simulation metamodel (Antunes F. , 2020)**

On the other hand, one of the drawbacks of simulation metamodels, mentioned in (Wang & Shan, 2007), is that the required computational cost and complexity of the metamodels may not be worth it when the number of input parameters is too large. This problem is known as the 'curse of dimensionality' (Köppen, 2000).

To counter this issue, and since it is not feasible to consider all the inputs and output variables if we want to simplify the simulation model, effort must be made to select input and output variables relevant to the problem we want to tackle. Figure 3 represents the input space of the metamodel as a subspace of the input space of the simulation model.

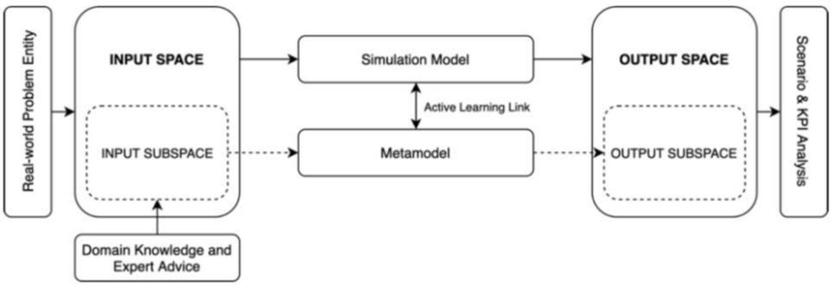


**Figure 3. Input-Output Subspaces used for metamodeling. (Antunes F. , 2021)**

Domain knowledge and expert advice should be employed for variable selection to ensure that the variables more relevant to the study KPIs are selected. Different subsets of variables should be considered within the iteration process.

## 1.2. Gaussian processes

One of the most employed metamodel formulations is the Gaussian Processes model (Williams, 2006).

Gaussian Processes (GP) belong to the family of non-parametric kernel methods. This means that they do not assume any particular function form (e.g., a linear model expects a specific y=wTx form) and that any prediction (e.g., for an input vector x) is entirely determined by the neighborhood relationships (e.g. all vectors in the dataset closer to x will contribute more to the prediction). This neighborhood is what we call a “kernel”. Thus, intuitively, one can see a Gaussian Processes as a type of *nearest-neighbour* algorithm, the key difference to the simplest K-nearest neighbour (KNN) approaches that GPs consider all pre-existing datapoints at once, by applying a complex mathematical process. Below, we give a summarized version of the mathematical approach. To further understand in detail, we recommend reading well-established literature (e.g. (Williams, 2006)).

Mathematically, a Gaussian Process (GP) is a collection of random variables, any finite number of which have joint Gaussian distributions. In other words, it is a multivariate Gaussian distribution with the same dimensions as the number of datapoints (vectors) in the space. Each of those vectors of the input space $(x_1, x_2, \dots x_n)$ is defined as a Gaussian distribution with a covariance matrix $\Sigma$ that connects its value to the other points in the space.

$$f = (f(x_1), \dots, f(x_n)) \qquad\qquad f \sim N(f|\mu(x), \Sigma)$$

As a consequence, in practice a GP is a distribution of random functions *f* that is itself multivariate Gaussian distributed. This function f is exactly what we want to use to make predictions (a new data point $x^*$ is therefore $y=f(x^*)$). A mean function $\mu(x)$ and a positive definite covariance matrix $\Sigma$ (also called *kernel*) fully specify the Gaussian process. One feature that makes GP an attractive model for metamodeling is that it automatically estimates the prediction **uncertainty** for each point $f(x)$ in the output space since it is equal to the estimated variance of its defined Gaussian distribution.

Finally, GP is also a good model for prediction since the estimation of a new point in the space is given by the posterior distribution of the previously trained points, which, in this case, is a closed-form expression that is also Gaussian and, therefore, easy to compute and use for predictions.

## 1.3. Random Forest

An alternative model in the metamodeling literature, is the Random Forest (RF), particularly for Agent-Based and Activity-Based models. (Agriesti, Kuzmanovski, Hollmen, Roncoli, & Nahmias-Biran, 2023)

A decision tree is a function that maps each input point to a discrete or continuous output class given its characteristics, where at each node a decision is made based on one of the input features to split the data into subsets. An RF is an ensemble (collection) of various regression trees, each trained in a subspace of the input space. The prediction of the ensemble space is given by the average of all the decision trees.

One of the benefits of RF is that we only need to decide a small number of hyperparameters (# decision trees to train; minimum number of samples in a terminating node; # data points in each sub-space partition) in order to configure it.

The main drawback is that RF does not quantify as well the prediction uncertainty, like in the case of GPs. However, one can simply “sample” from all individual decision trees (an RF with N decision trees will generate N different results), and use the result to approximate an output distribution.

# 2. The use of Metamodels

## 2.1 Initial Sample

Training a metamodel requires an initial sample of input and output values collected from the simulation model we aim to mimic. The most common sample technique to initialise our training dataset is the Latin Hypercube Sampling or LHS, (Lin & Tang, 2022), which has one-dimensional uniformity, meaning that, for each input variable, its range is divided into the same number of equally-spaced intervals as the number of observations, which means there is exactly one observation in each interval. This technique helps us explore the whole space of input uncertainty; however, it comes with a high increase in computational time if the number of samples is large.

## 2.2 Sampling strategy

Active learning emerges as a solution when labelled data from the simulation is difficult or expensive to obtain. Its objective is to maximise the knowledge of our labelled data point before another labelled point is requested to have as few training samples as possible while reaching the target model prediction accuracy.

Both metamodels and active learning are mathematical techniques that aim to reduce the computational costs of running multiple simulations. While metamodeling allows us faster simulation computations, active learning ensures that the computing time of training the metamodel is not so long as to lose the benefits of employing the metamodel itself.

In other words, by selecting the more informative data point in every iteration, we avoid running the simulation model as often as without the active learning strategy, saving computational resources.

A general active learning approach can be defined by its five principal components. *L* refers to the labelled training set, and *U* is the unlabelled one, where we assume that the size of *L* is much smaller than *U*. *M* is the machine learning model, in our case, the metamodel. *Q* is the query function that decides the data point which will be asked to be labelled next by the oracle, *O.*

The global process that combines active learning and metamodelling could be divided into the following four essential steps (Antunes, Riis, & Camara, 2022):

1. Training: In this phase, only a small initial training set is used to estimate the parameters of the metamodel.

2. Prediction: During this second step, we use the fitted metamodel to predict the simulation outputs in different unlabelled points of the input domain.

3. Request: Based on the query criteria, which generally relates to the variance of the estimated outputs computed by the metamodel (how sure are we about their estimation value), we select an unlabelled data point whose simulation value will be requested by the oracle.

4. Response: The new labelled data point is added to the training sample, and the first process is repeated.

These steps are repeated iteratively until a predefined stopping criteria is satisfied.

## 2.3 Policy Specific objectives

As mentioned before, metamodels serve a variety of objectives (Riis, 2023):

**Calibration:** Metamodels can increase the accuracy of simulation models by finding the parameters that better match the real-world data since quicker adjustments can be made to compare different input parameters to fit the observed outputs better, (Arora et al., (2021)

**Optimisation:** Metamodels can help to efficiently find the input parameters that optimize (maximize/minimize) an objective function (Osorio & Bierlaire, 2013).

**Prediction:** This is the essential use of the metamodels when we mimic the behaviour of the simulation model to enable fast prediction of the outputs, thus avoiding the need to run the heavier simulation model. This is significant when multiple predictions are needed. (Chris et al., (2021)

**Scenario Discovery:** This objective combines prediction and optimisation, where we aim to find regions in the input space where the simulation model performs according to some given criteria. E.g., regions where, given a specific policy, the simulation model will provide very low quality of outputs. Scenario Discovery is the primary purpose for using metamodels within the eMOTIONAL Cities project. (Antunes, Pereira, & Ribeiro, 2020)

**Sensitivity analysis:** Lastly, metamodels can also be helpful for sensitivity analysis, in other words, quantifying the influence of input variables on the model outputs. Sensitivity analysis helps find the input parameters that are more influential to our output values, for example, which parameters have a stronger impact on a specific result (Cuiffo & Lima Azevedo, 2014), i.e. by how much one needs to change an input to achieve a specific result.

The following deliverable will explore more effective ways to address the Scenario Discovery goal with more suitable and targeted sample techniques for the metamodeling framework.

# 3. Documentation for simulation models

Well-documented simulation models constitute a crucial component within the metamodeling framework (Antunes F. , 2021). The documentation for the simulation models should include the following:

- Type of variable, i.e., continuous vs. discrete.

- Possible range of values (theoretically).
- Accepted range of values (in practice).
- Impossible combination of values (both theoretically and in practice).
- Default range of values (e.g., the “average” standard value).
- Diagram of causal dependency between input and output variables and within the input and output dimensions.
- Brief description and interpretation within the entire simulation environment.

This information would be needed for the final version of the metamodelling framework for the three case studies mentioned in (Lima Azevedo, Miranda , Camara Pereira, Torres Lahoz, & Ancora, 2023).

DTU will define the format and content to be provided for such documentation in collaboration with all WP7 partners. The documentation will be prepared by each of the teams working in the modelling exercises for all three case studies, ongoing within WP5 and WP6 activities. The responsible for Case-Study 1 is DTU; the responsible for modelling Models 1 in both case-study 2 and 3 is the FMUL-U.Lisbon; the responsible for modelling Models 2 in both case-study 2 and 3 is IGOT in collaboration with DTU. The documentation and formulation of the models, along with the data sets needed for scenario discovery, will be provided to WP6 (namely CLIMA and DTU) to build the baseline scenarios for scenario discovery. WP6 will develop the code accordingly with T6.3

# 4. Metamodeling in eMOTIONAL Cities

## 4.1 Framework

The main objective for constructing metamodels within the eMOTIONAL Cities project is to apply Scenario Discovery in our simulation models. To accomplish this, we created a combined framework that can be visualised in Figure 4.

The first part of the framework represents the metamodeling. We chose a GP for the metamodel specification and trained it in a small LHS sample. Once the first model parameters are estimated, we use the metamodeling to predict the outputs of unlabelled points in the input space of uncertainty.

Then, instead of analysing the predictive output variance of the metamodeling prediction, which would be the case in the basic AL strategy, we focus on the values of these predictions that have more relevance given our criteria within the SD framework. Different criteria will be tested, applied and presented in the Scenario Discovery Deliverable D.7.3.

After analysing the predictions, the algorithm checks whether the stopping criteria have been met. The stopping criteria could be, for example, the number of sampled points so far.

If the stopping criterion is unmet, we will query the next point based on the selected query criterion. The selected point will be passed to the simulation model, and the output will be stored in the labelled dataset.

Given the increase in our training dataset, we will train our metamodel again and repeat the previously explained process until the stopping criterion is met.

Then, our metamodel will be ready for policy analysis. This could mean further exploration of the uncertain input space, trying different policies, and summarising the discovery insights for the stakeholders. We called this process the use of the meta simulation.

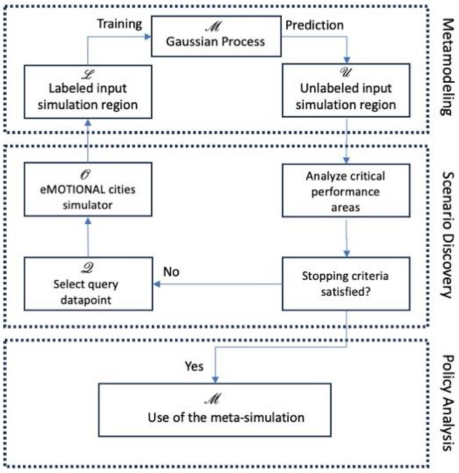


**Figure 4. Preliminary High-level Metamodeling approach.**

## 4.2 Case studies

The metamodeling framework presented above will be applied in the three eMOTIONAL Cities case studies.

When working with Scenario Discovery, we follow RAND's XLRM framework, which defines the limits and scope of Scenario Discovery (Lempert, 2003).

In the XLRM framework, X stands for **Exogenous uncertainties** (X), which are all the variables we have no control over; however, they play an essential part in the possible outcomes of our actions. **Policy levers** (L) are the actions policymakers can apply to modify the current environment. **The Relationships** (R) are potential ways the future could evolve based on the policymakers' choices of levers and the manifestation of the uncertainties. Given some initial conditions, the relationships could be seen as the results of our simulation. Finally, **Measures** (M) are the performance standards that policymakers and other interested stakeholders use to rank the desirability of various scenarios.

When combining this framework with the metamodeling one, we identify that the **input of the metamodel** corresponds with the exogenous uncertainties (X) and the policy levers (L) determined for each simulation since it is something that the modeller can tune.

On the other hand, the **output of the metamodel** should provide the measures (M) needed to identify each simulation run as desirable or not. Finally, the metamodel formulation represents the relationships (R) of the simulation model.

### 4.2.1 Case study 1. Daily activities, mobility, and emotions: a large-scale simulation of Copenhagen

For this case study, we rely on SimMobility (Adnan et al., 2016), an agent-based simulation framework that models the behaviour and operation of an urban system. We focus on the mid-term (MT) that provides the activities, travel plans and actions of the simulation agent during the day and match the predicted daily activities with a daily stress profile for each agent of the simulation.

Table 1 summarises the XLRM framework used for this case study.

**Table 1. Summary of the XLRM for Case Study 1**

| | |
|---|---|
| **Uncertainties (X)**<br>Population attributes<br>Behavioural preferences | **Policy Levers (L)**<br>Urban features<br>Transportation performance |
| **Relationships (R)**<br>Activity-based model<br>Daily emotional profile model | **Measures of Performance (M)**<br>Accessibility<br>Arousal indicators<br>Valence Indicators |

The preliminary metamodel library has been applied to a simplified version of this simulation model formulation, where we only employ the Day Pattern model. Here, the agents decide on the activities to perform during the day. These activities can either be the primary activities of tours, or activities performed at intermediate stops. After selecting their activities, agents then decide on the exact number of tours performed for each activity type (e.g., work, education, shopping, recreation, personal, or escort). (Lima Azevedo, Miranda , Camara Pereira, Torres Lahoz, & Ancora, 2023)

Our goal for the simplified version was to quantify the number of individuals in the sample who perform leisure trips since there is evidence in the literature relating leisure activities to better mental health and well-being. (De Vos, Schwanen, & Witlox, 2017)

In this example, we treat some of the estimated values of the preday pattern as uncertainties, and we seek to understand better the relationship that links the increase of leisure trips with some input variables that we could modify as policymakers, namely the coefficients:

- beta_female_travel: coefficient that represents the increase or decrease in the willingness to travel if the agent is a woman.
- beta_TRANS_travel: coefficient that represents the increase or decrease in the willingness to travel if the agent has a public transport monthly pass.
- beta_student_travel: coefficient that represents the increase or decrease in the willingness to travel if the agent is a student.

The Python files contained in the preliminary version of the metamodel library are described hereafter:

- **LHS.py**: performs LHS sampling in an input space of three variables.
- **Model.py:** compute the number of leisure trips given an x value with the changes in some beta values (input space of uncertainty). This could be seen as the simulation for this simplified example.
- **ModelSD.py**: this file combines the two previous ones. The user can set the number of simulations runs, and it directly saves the LHS input values and the corresponding output in two different files.
- **GaussianProcess.ipynb:** explains and constructs a Gaussian Process metamodel for the simulation model with 400 labelled points. After, it computes the prediction in another 400 unlabeled input points.

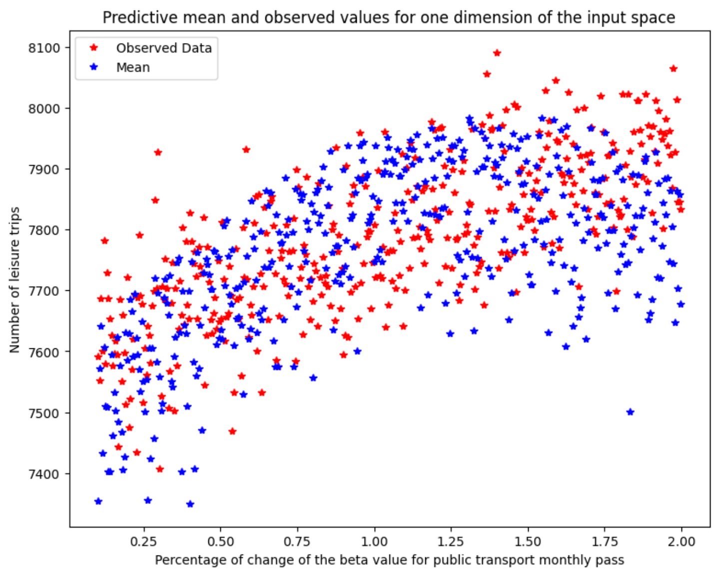


**Figure 5. Observed and predictive values for the number of leisure trips given a change in the beta for public transport monthly pass**

The results of this notebook are summarised in Figure 5. In red, the observed points with their corresponding outputs are computed by the simulation model. These points are used to train a GP metamodel whose mean values are represented as blue. As we can see, the metamodel learns the underlying structure that relates to the number of leisure trips with the input values. To quantify this, we compute the correlation coefficient between the predictive mean values and the true value of the simulation in these 400 test points. Note that this value will not be available in the real case, but it gives us an understanding of how well this metamodel fits the data in this proof-of-concept model. The correlation coefficient is 0.63. It is worth noticing that the model is less accurate in the extremes of the input space where the uncertainty of the GP metamodel is higher since there are no close observed values in all the input space directions.

- **GP_AL.ipynb:** explains and constructs a Gaussian Process metamodel for the simulation model using Active learning. It computes the prediction in 400 unlabeled input points using only 200 labelled points and 100 query points in estimation.

  Figure 6 represents the results of this notebook. The observed initial data points used to train the metamodel are represented in yellow. After the metamodel is trained, the point with the highest entropy, i.e. the point with the most uncertainty in its prediction, is query. The total number of query points is set to 100.

  The final metamodel is trained in the final 300 labelled points, and its mean values for 400 unlabeled LHS points are represented in blue. We observed that most of the query points belong to the borders of the input space, where, as we mentioned, the uncertainty is higher. The metamodel seems to learn the underlying structure that relates inputs with the output; even though it tends to underestimate the number of leisure trips. The correlation coefficient correlation coefficient between the predictive mean values and the true value of the simulation in these 400 test points is 0.82, higher than in the estimation without AL. Other targeted sampling techniques can be applied to improve even more the estimation, where not only the more uncertain sample is selected. We will explore more sampling techniques targeted for Scenario Discovery in the Scenario Discovery Deliverable D.7.3.

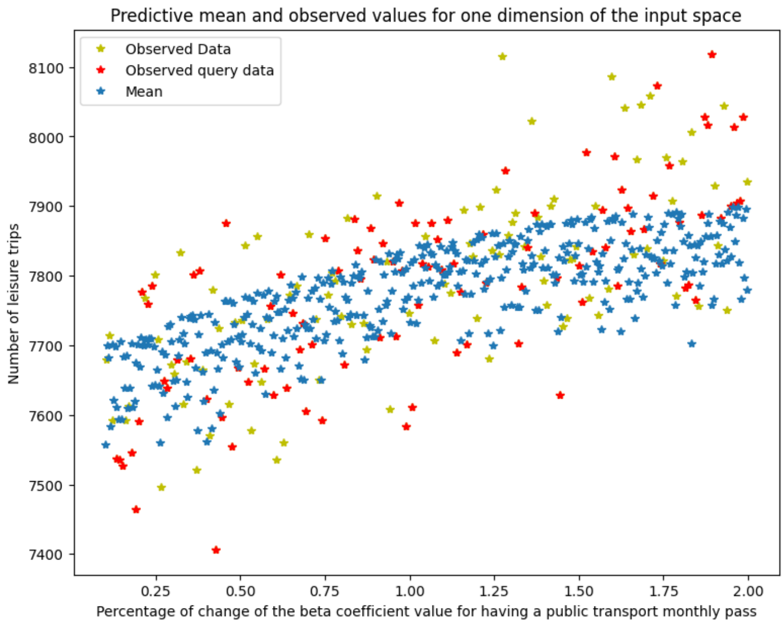


**Figure 6. Observed and predictive and query values for the number of leisure trips given a change in the beta for public transport monthly pass**

### 4.2.2 Case study 2. Walking, Streets and Emotions: A Neighbourhood Study in Lisbon

Case Study 2 relies on two model components. The first component models the individual emotional response from the exposure to different urban scenarios given individual characteristics, mental well-being and personality traits to predict emotional and cognitive experience. The second is an urban feature generative model that can create consistent levels of naturalness and complexity for different neighbourhoods in a scenario discovery setting. The combination of these two models within WP6 will allow the exploration of the emotions and cognitive experience in different surroundings within this Case Study 2.

Table 2 presents the summary of variables that will be used in the metamodeling formulation of this case study. (Lima Azevedo, Miranda , Camara Pereira, Torres Lahoz, & Ancora, 2023)

**Table 2. Summary of the XLRM for Case Study 2**

| Uncertainties (X) | Policy Levers (L) |
|---|---|
| Urban context | Street infrastructure |
| h/mGLM parameters | Building features |
| Individual characteristics | Greenery |

| Relationships (R) | Measures of Performance (M) |
|---|---|
| Emotional Response<br>Urban Feature Generation | Arousal indicators<br>Valence Indicators<br>Subjective Crowding<br>Subjective Naturalness |

### 4.2.3 Case Study 3: Walking, Streets, Microclimate and Emotions: A Path Study in Lisbon or London

Case-Study 3 is an extension of Case-Study 2, where the external environment is modelled with additional descriptors, and the emotional transitions are modelled explicitly (dynamics). For this Case-study 3, the model from Case Study 2 is also re-estimated using the data collected in WP5's outdoor navigation experiment.

Table **3** presents the summary of variables that will be used in the metamodeling formulation of this case study. (Lima Azevedo, Miranda , Camara Pereira, Torres Lahoz, & Ancora, 2023)

**Table 3. Summary of the XLRM for Case Study 3**

| Uncertainties (X) | Policy Levers (L) |
|---|---|
| Urban context<br>h/mGLM parameters<br>Individual characteristics<br>*Thermal comfort indicators and their sensitivities* | Street infrastructure<br>Building features<br>Greenery |
| **Relationships (R)** | **Measures of Performance (M)** |
| *Emotional Response v2*<br>Urban Feature Generation | Arousal indicators<br>Valence Indicators<br>Subjective Crowding<br>Subjective Naturalness |

# 5. Library

The library with the preliminary code for the metamodels can be found in the project's dedicated repository at:

https://github.com/emotional-cities/D7.2-Metamodelling-

# 6. References


Williams, C. K. (2006). *Gaussian Processes for Machine Learning. .* Cambridge, MA: MIT Press.

Antunes, F. (2021). *D3.2 Metamodels Requirements Specification.* NOSTROMO Grant: 892517 Call: H2020-SESAR-2019-2.

Antunes, F. (2020). *D3.1 Preliminary Metamodeling Methodology.* NOSTROMO, Grant: 892517, Call: H2020-SESAR-2019-2.

Antunes, F., Riis, C., & Camara, F. (2022). *D3.4 Final Metamodelling Methodology.* NOSTROMO, Grant: 892517 Call: H2020-SESAR-2019-2, Topic: SESAR-ER4-26-2019.

Lima Azevedo, C., Miranda , B., Camara, F., Torres Lahoz, L., & Ancora, L. (2023). *D7.1 - Preliminary Specification of Case Studies.* eMOTIONAL Cities: mapping the cities through the senses of those who make them .

Agriesti, S., Kuzmanovski, V., Hollmen, J., Roncoli, C., & Nahmias-Biran, B.-h. (2023). *A Bayesian Optimisation approach for calibrating large-scale activity-based transport models.* arXiv:2302.03480v1.

Wang, G. G., & Shan, S. (2007). *Review of metamodeling techniques in support of enginee ring design optimisation.*

Köppen, M. (2000). *The curse of dimensionality. In: 5th Online World Conference on Soft Computing in Industrial Applications p. 4-8. .*

Friedman, L. W. (2012). *The simulation metamodel.* Springer Science & Business Media.

Barton, R. R. (1998). *Simulation metamodels. In Simulation Conference Proceedings,.* Winter, Volume 1, pages 167–174. IEEE.

Lin, C., & Tang, B. (2022). *Latin Hypercubes and Space-filling Designs.* arXiv:2203.06334v1 [stat. ME].

Arora, N., Chen, Y.-f., Ganapathy, S., Li, Y., Lin, Z., Osorio, C., . . . Tsog- suren, I. (2021). *An Efficient Simulation-Based Travel Demand Calibration Algorithm for Large-Scale Metropolitan Traffic Models," pp. 1–14.* arXiv: 2109.11392.

Osorio, C., & Bierlaire, M. (2013). *"A Simulation-Based Optimisation Framework for Urban Trans- portation Problems,".* Operations Research, vol. 61, no. 6, pp. 1333–1345, issn: 0030-364X. doi: 10.1287/opre.2013.1226. [Online].

Riis, C., Antunes, F., Gurtner, G., Pereira, F., Delgado, L., & Azevedo, C. (s.f.). *Active learn- ing metamodels for atm simulation modeling,.* Proceedings of the 11th SESAR Innovation Days, vol. 2021, 2021.

Riis, C. (2023). *Bayesian Machine Learning for Simulation Metamodeling .* Phd Thesis. DTU Management.

Antunes, F., Pereira, F., & Ribeiro, B. (2020). *Directional Grid-Based Search for Simulation Meta-modeling Using Active Learning,”.* Intelligent Transport Systems. From Research and Devel- opment to the Market Uptake, pp. 32–46.

Cuiffo, B., & Lima Azevedo, C. (2014). *“A Sensitivity-Analysis-Based Approach for the Calibration of Traffic Simulation Models,”.* IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 3, pp. 1298–1309, issn: 1524-9050. doi: 10.1109/TITS.2.

De Vos, J., Schwanen, T., & Witlox, F. (2017). *The road to happiness: from obtained mood during leisure trips and activities to satisfaction with life.* 2017 World Symposium on Transport and Land Use Research (WSTLUR); July 3-6, 2017; Brisbane, Australia.

Lempert, R. J. (2003). *Shaping the next one hundred years: new methods for quantitative, long-term policy analysis. .*

Adnan, M., Pereira, F., Lima Azevedo, C., Basak, K., Lovric, M., Raveau, S., Ben-Akiva , M. (2016). *SimMobility: A Multi-Scale Integrated Agent-based Simulation Platform.* 95th Annual Meeting of the Transportation Research Board. Washington DC, US.

Antunes, F., Camara, F., & Riss, C. (2022). *D3.4 Final Metamodelling Methodology.* NOSTROMO Grant: 892517 Call: H2020-SESAR-2019-2.