eMOTIONAL
Cities

Mapping the cities through the senses of those who make them

DELIVERABLE 7.6

Scenario Discovery library II

WP7 – Scenario Discovery

07.08.2023

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	eMOTIONAL Cities: mapping the cities through the senses of those who make them
Deliverable	D7.6 – Scenario Discovery Library II
Work package	WP7 – Scenario Discovery
Task	T7.5 – Scenario Discovery Library
Number of pages	
Dissemination level	Public
Leader	DTU
Contributors and peer-reviewers	FMUL, BR
Date	06/08/2024
File name	eMCities_WP7_D7.6
Version	V2
Authorship	Lorena Torres Lahoz (DTU), Carlos Lima Azevedo (DTU), Francisco Pereira (DTU)

Index

Executive Summary

This document is the deliverable “D7.6: Scenario Discovery Library II” 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 T7.5 “Scenario Discovery Library”, 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 Scenario Discovery methodology and its state-of-the-art and research gaps. It also describes the version of the Scenario Discovery methodology employed 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
AL	Active Learning
B.O.	Bayesian Optimization
CART	Classification and Regression Tree
CVT	Centroidal Voronoi Tessellation
E.I.	Expected improvement
G.P.	Gaussian Process
KPI	Key Performance Indicator
LHS	Latin Hypercube Sampling
PRIM	Patient Rule Induction Method
P.T.	Public transportation
Q.D.	Quality Diversity
RDM	Robust Decision Making
SD	Scenario Discovery

1. Scenario Discovery

1.1 Background

Uncertainty can be defined as limited knowledge about future, past or current events, and it can be quantified as any departure from the ideal of complete determinism [1]. [1]It is often quantified in terms of “probability” or “probabilistic measures.” For example, in the context of decision-making, by identifying plausible future conditions or scenarios, where probabilities are assigned to different outcomes based on their likelihood. Another example is when the input data is incomplete or noisy, leading to uncertainty in model predictions, where probabilistic measures are used to estimate the confidence or likelihood of various predicted outcomes.

In particular, we are interested in the **deep uncertainty** concept. Lempert et al. have defined it as “the condition in which analysts do not know or the parties to a decision cannot agree upon (1) the appropriate models to describe interactions among a system’s variables, (2) the probability distributions to represent uncertainty about key parameters in the models, and/or (3) how to value the desirability of alternative outcomes.” [2] In other words, deep uncertainty refers to situations with significant ambiguity and disagreement over the models to use, the probabilities of various outcomes, and even the criteria for evaluating these outcomes. This often occurs in complex systems where information is incomplete or too complex, making it difficult to make predictions or reach a consensus.

There are four (not mutually exclusive) ways of dealing with deep uncertainty when policymaking: [3]

- **Resistance**: Plan for the worst-case scenario. This approach can be costly and still cannot handle Black Swans [4], the highest level of uncertainty, where any past data cannot convincingly point to their existence.
- **Resilience**: Apply policies that can recover quickly given any change in the model conditions. It accepts a small quantity of pain initially but focuses on a quick recovery.
- **Static robustness**: Implement policies that will perform reasonably well in most of the possible future cases. A robust policy is good enough across various future case scenarios, contrary to an optimal policy with the best performance for a one-case scenario.
- **Adaptative robustness**: This approach combines a robust policy with a monitoring system. When some of the monitored values reach a defined threshold, the policy can be changed during the implementation phase to be more suitable, given the changes in the initial conditions.

This report will focus on static robustness within the Robust Decision-Making framework.

1.2 Robust Decision-Making Framework

Robust Decision-Making (RDM) is a quantitative decision-analytic method that uses available information to help decision-makers find more effective strategies to achieve their goals in the face of deep uncertainties. The robust Decision-Making objective is to find good policies across a wide variety of future case scenarios instead of finding a unique optimal policy for a one-case future scenario, like in a more traditional optimization framework. [5]

Therefore, a critical RDM step is identifying the combination of simulation model input parameters for which a candidate robust policy performs poorly compared to alternative strategies; this procedure is defined as Scenario Discovery.

Scenario Discovery (S.D.) aims to provide summarised, understandable, and valuable information for decision-makers about the key vulnerabilities of a given policy over the possible future states of the system. Moreover, within the eMotional cities framework, we can further expand these principles to explore other relevant policies, for example, searching for those that perform particularly well compared to other alternatives or seeking future case scenarios with more probability of occurrence given a known exogenous distribution.

In summary, S.D. answers the question: What are the most critical vulnerabilities of the strategy under consideration?

Vulnerable cases are those combinations of uncertainty input parameters whose outputs do not meet the performance requirements of the policy under consideration.

Therefore, SD aims to find descriptions of the combinations of a small number of input parameters of the simulation model that are most strongly correlated with certain classes of non-performing results. These combinations of descriptors define a subspace in the uncertainty input space that is often called a box or a scenario.

Scenario Discovery results are based on three metrics:

- **Coverage**: The percentage of vulnerable points captured in a box with respect to the total number of vulnerable cases in the whole exploration space.
- **Density:** fraction of vulnerable cases within the box relative to the total number of points inside the box.
- **Interpretability**: indicates how easy it is to understand a scenario, usually described by the number of constraints that define each box.

An ideal set of scenarios would combine high density, coverage and interpretability [6]. However, these measures are generally complementary; an increase in coverage will usually increase the number of captured vulnerable cases but decrease the box's density.

Generally, Scenario Discovery algorithms provide the user with multiple combinations of these three performing measures distributed in a Pareto-frontier, as the one provided in Figure 1, where some trade-off criteria or user domain knowledge interaction are applied to select the best-performing box of a given case study.

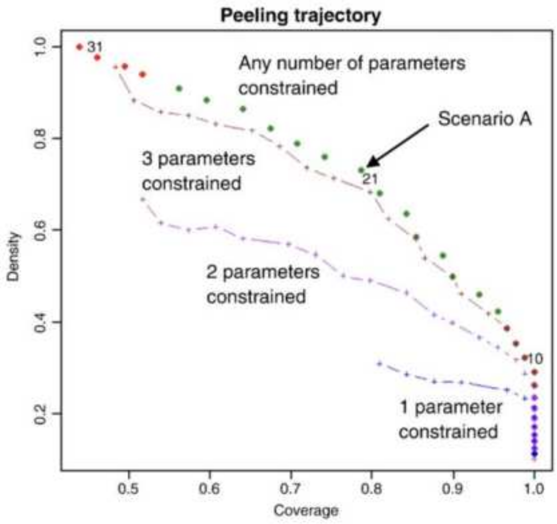

Figure 1. Coverage/density trade-off curves for scenarios using 1,2,3 or any number of restricted parameters [7]

In a nutshell, Scenario Discovery consists of a trade-off between optimization and exploration that highlights uncertain input conditions where the proposed policy will not satisfy a particular success/failure criterion, a crucial step in the design of robust policies. For example, we could introduce a policy to decrease the price of public transport to increase travel accessibility. However, the impacts of this policy can lose its relevance if the general income of the population also decreases.

The following section provides an overview of the main algorithm used for Scenario Discovery.

2. Algorithms for Scenario Discovery

To our knowledge, no existing algorithm performs a task identical to that required for scenario discovery [6]. However, the more similar algorithms that are most employed are classification and bump-hunting. The following subsections explain and compare them with other new and innovative proposed formulations.

2.1 Traditional Algorithms

2.1.1 Patient Rule Induction Method (PRIM)

PRIM [8] aims to find regions in the input space where the mean output value is relatively higher than the mean of the output over the entire space.

To accomplish this goal, PRIM performs what is defined as **peeling.** The algorithm starts with a box covering the whole input space, and a small sub-box in the extremes of the current box is removed at each algorithm step. The sub-box chosen for removal is the one that produces the highest output mean value in the remaining box. This procedure is repeated until the final box's **support** (the number of vulnerable points divided by the total number of points) arrives at a given threshold. Figure 2 shows a sequence of peeling operations for a given dataset, where the bold points are the vulnerable cases, represented as 1, and the rest are non-vulnerable cases set to 0.

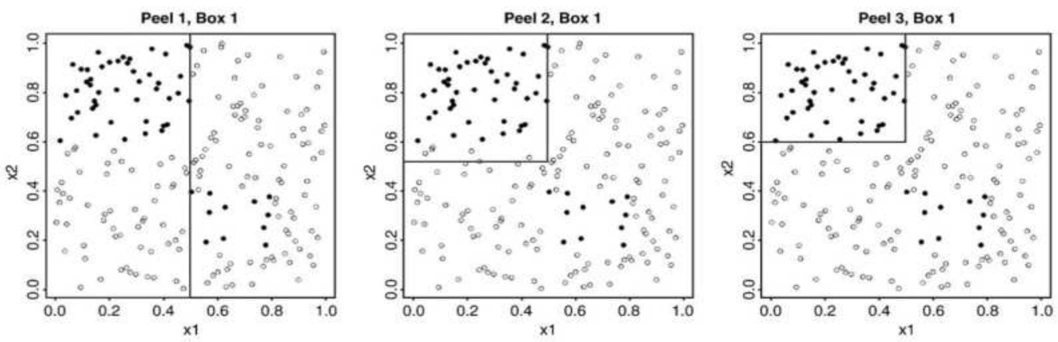

Figure 2. A sequence of peeling processes performed by the PRIM algorithm on a dataset [7]

PRIM has a hypermeter called **patient** that controls the percentage of data points removed at each iteration of the peeling process. Small values of the patient are preferred; however, they could also cause the algorithm to cut the end of the boxes inappropriately, which would otherwise extend the whole length of the parameter range.

The peeling process can be applied multiple times. A second box can be constructed using the initial points and removing the observations the first found box covers. This process is called covering and allows for a more intense exploration of the whole input space.

2.1.2 Classification and Regression Tree (CART)

CART's objective is to minimize misclassification errors and obtain regions of space with high purity. The output has the shape of a decision tree that determines the output class (vulnerable or not vulnerable cases) given combinations on the input dimensions [6].

The CART hyperparameter is called pruning. After dividing the whole space into a large and complicated tree, pruning combines the last spits given a selected criterion (usually the minimum number of points in the last split) to achieve a more interpretable and better predictive tree.

2.1.3 Comparison and Limitations

In conclusion, CART partitions the whole input space into disjoint regions in contrast with PRIM, whose boxes can overlap. CART generates better coverage/density performance than PRIM. However, we have generally found PRIM more practical for actual policy analysis applications because, in complex datasets, CART often fails to reach sufficient levels of coverage before requiring an infeasibly large number of boxes [6]. Finally, it is well noted that these algorithms can be enhanced by user interaction, which can choose the best-performing box at the end of the algorithm according to some selective criterion or domain knowledge. In CART, the user can choose between different prune trees and, in PRIM, between the more suitable combination of coverage/density/interpretability for its application.

Even though some efforts have been made to improve these traditional algorithms, like using PRIM after applying orthogonal rotations in the dataset [9], a clear improvement would employ adaptive sampling methods to generate more simulation model runs near the edges of scenarios to improve the confidence in the statistical significance of the results. New machine learning approaches based on active learning [10] may address sampling density challenges by iteratively requesting new simulation model runs that add cases to the dataset in the most needed regions to improve the performance of the scenario discovery algorithms.

2.2 Bayesian Optimization

Bayesian Optimization (B.O.) [11] is a powerful optimization technique that uses probabilistic models to efficiently search for the optimal solution in a complex, expensive input space. This approach has gained increasing attention in recent years due to its ability to handle black-box functions, where the objective function's analytical form is unknown, and only a limited number of evaluations are possible due to the high computational cost. In addition, it proves beneficial for functions that are either expensive or challenging to evaluate, particularly in cases where access to the first or second derivative, typically required by other optimization algorithms, is unavailable.

Bayesian optimization [12] involves the construction of a probabilistic model, which in our case would be a predefined metamodel, typically done using a Gaussian process (G.P.). A Gaussian Process is a collection of random variables, any finite number of which are consistent joint Gaussian distributions [13].

The two components of B.O. are the probabilistic model and an acquisition function. The acquisition function is the formula that selects the sample to evaluate next, thus representing the sampling criterion. One of the most used and straightforward acquisition functions is called Expected Improvement (E.I.).

In the simplest scenario, we assume that our observations do not have intrinsic noise. Therefore, when we evaluate our model in the *n* initial dataset, we can define the highest observed value as:

$$f_n^* = max_{m \leq n} f(x_m)$$

If we could make another evaluation in any input point x, we would like to choose the one where the difference $f(x) - f_n^*$ is the largest, which is the same as selecting $[f(x) - f_n^*]^+ = (f(x) - f_n^*, 0)$, where the + in the formula converts the value of the difference in a way that is always positive.

Therefore, at each iteration, n, we would choose the x that maximizes the acquisition function, defined as:

$$EI_n(x) = E_n[(f(x) - f_n^*)^+]$$

where $E_n[\cdot] = E_n[\cdot \, | x_{1:n}, y_{1:n}]$ is the expected value over the posterior distribution of the G.P. calculated over the previous evaluation of f at $x_1, \dots, x_n$. The posterior distribution is the function that results from incorporating the information on the previous points in our G.P., and it could be seen as the conditional probability of the points given the information on the already observed ones. The posterior distribution of a Gaussian distribution is also Gaussian, which makes this estimation easier to compute and use.

When we integrate over the posterior distribution, the acquisition function is higher for points with a high posterior mean value and for those with high posterior variance. This is intuitive since points with high mean estimation values will likely give us high predictive values. On the other hand, points with high uncertainty can also provide us with knowledge in zones that we have yet to explore and where the maximum value could also be located. This balance between high mean or high variance values is known as the *Exploration vs. exploitation trade-off.*

Figure 3 provides a visual interpretation of the E.I. value for different points in the input space.

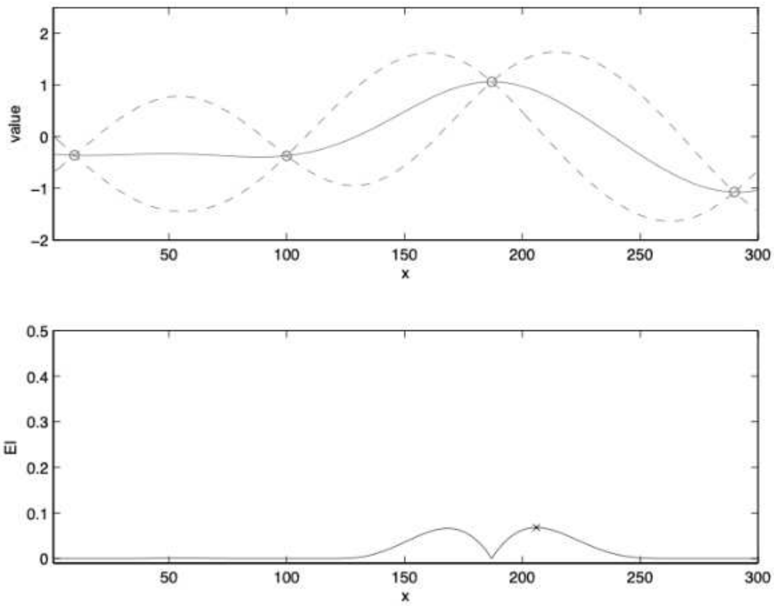

Figure 3. The upper panel represents the posterior distribution on a one-dimensional input space where the circles are the previously measured point, the solid line is the posterior mean, and the dashed lines are the confidence intervals. The lower panel represents the E.I. value where an x is marked for the point with the highest value that would be evaluated next [12]

Finally, this algorithm can also be implemented with some variations to account for situations when we assume that our simulation output points have some intrinsic noise values.

We performed E.I. for Scenario Discovery within the eMOTIONAL Cities framework and provided the corresponding code in the library.

2.3 Quality Diversity Algorithms

Traditional optimization algorithms have focused on providing a unique optimal value that maximizes the whole input space. However, inspired by natural evolution, where multiple species have appeared to adapt to different conditions in an optimized manner, Quality Diverse (Q.D.) algorithms [14] seek to explore the whole input space, looking for high-performance solutions in different areas and revealing the relationships between the different dimensions of interest.

Q.D. algorithms aim to illuminate the whole searchable space with diverse solutions of high quality, where the Q.D. score is represented by the sum of the fitness for all the solutions in the features space [15].

Q.D. algorithms are composed of a fitness function that measures the performance of each solution (how good or bad the results are for each point) and a space defined by some descriptors of interest; we call this space the feature space. The feature space is usually smaller than the search space, which is the space that contains all the input dimensions that define a solution and is usually very high-dimensional.

The process starts by discretizing the feature space in a grid that can be uniformly built or tuned according to the interest in the different dimensions. Then, we initialize the algorithms by filling some of the cells of this space, and we calculate the fitness for each solution with the fitness function.

Once the feature space is initialized, we start with the addition mechanism: we select a new point and add it to the feature space if the cell where it belongs is empty or if its fitness is higher than another point that is now in its corresponding cell.

To select the next point, one or two points that are already in the feature space are chosen to produce a new offspring via mutation or crossover.

- **Crossover:** swaps parts of the solution with another solution. Its primary role is to provide a mix of solutions.
- **Mutation:** change parts of one solution randomly, increasing the diversity of the solutions.

These mechanisms are performed in the search space of the solutions; therefore, if the relation between the search and the feature space is not linear, many points may be mapped into the same cell in the features space. This is when the addition mechanism comes into play to select the best-performing solution. The selection mechanism of the parents (the points used to produce the new offspring) is usually uniformly random distributed.

This process is repeated until the termination criterion is reached, usually set to the number of iterations performed.

Figure 4 represents a schematic explanation of the framework. Finally, the result of the algorithm will be a collection of points distributed in the feature space, defined by dimensions of interest, and whose performance values are high all over the space.

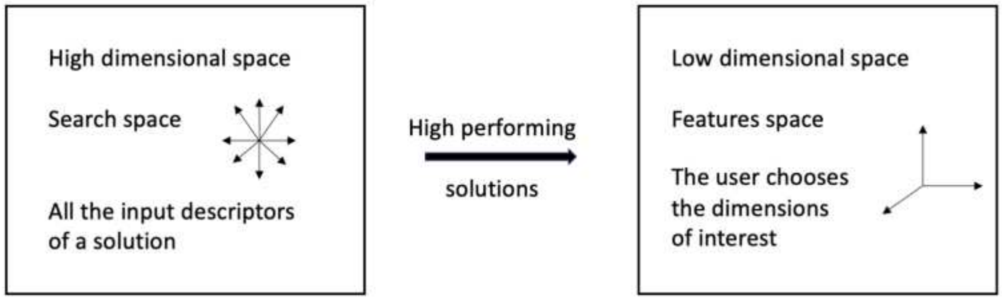

Figure 4. Framework of Quality Diversity Algorithms

In this report, we compute the CVT-MAPs Elites variant of these Quality Diversity algorithms. The concrete specification of CVT-MAPs Elites is that the grid of the features space is computed as a Centroidal Voronoi Tessellation (CVT), a space partition technique that is more efficient in high-dimensional spaces. Figure 5 explains and represents this efficiency. Moreover, in CVT-MAPs Elites, the selection mechanism of new points in the search space only employs crossover.

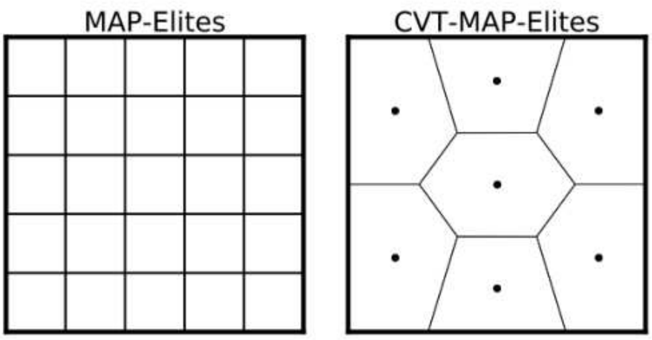

Figure 5. MAP-Elites divides the feature space according to some pre-specified number of discretizations per dimension. This means the number of niches grows exponentially with additional dimensions or discretizations. Thus, it cannot be used in high-dimensional feature spaces. Instead, the Centroidal Voronoi Tessellation partition the feature space into k homogeneous geometric regions, where k is the pre-specified number of niches [20].

Each region of the Centroidal Voronoi Tessellation partition is defined by its centroid. Therefore, each solution is assigned to the grid whose centroid is closer to the solution coordinates in the feature space. The archive is the collection of solutions that populate the feature space at each iteration.

3. Scenario Discovery in eMOTIONAL Cities

3.1 Framework

The Scenario Discovery framework used for the eMOTIONAL Cities framework can be summarised in Figure 6.

The first step is to define the S.D. problem based on the RAND's **XLRM framework**, which normally defines the limits and scope of Scenario Discovery.

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 [16].

The second step is to sample input points within the input uncertainty space. The most common approach is to use the Latin Hypercube Sampling (LHS) [17] technique since it has one-dimensional uniformity, where for each input variable, its range is divided into the same number of equally-spaced intervals as the target number of observations; in LHS there is exactly one observation per interval. This technique ensures that the whole input space is initially explored uniformly.

The third step is to run the simulation model or the metamodel simulator in the selected input points, given a selected policy lever, and calculate their performance measure for each point.

The fourth step, called Scenario Discovery, consists of running one of the previously defined algorithms or their combination to seek the regions in the input space that highlight the most vulnerable cases.

For the eMOTIONAL Cities framework, we introduce feedback from Step 4 back to Step 2. Based on the scenario discovery results, this feedback represents that we return to the sampling strategy and perform an adaptive sampling method that selects a point based on a criterion related to the already found PRIM boxes or the distribution of vulnerable cases.

This modification enables a more effective sampling that optimizes the knowledge of the already requested sampling points.

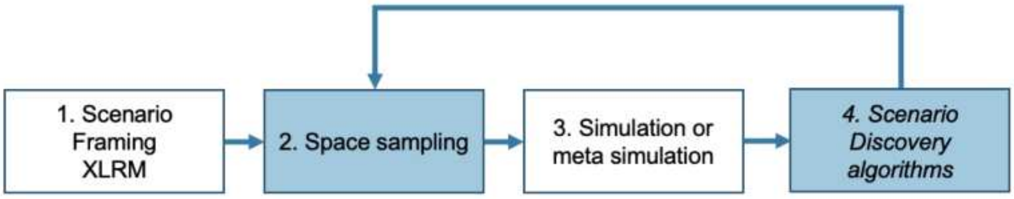

Figure 6. Scenario Discovery Framework for eMOTIONAl Cities

3.2 Performed Algorithms

In this second version of the Scenario Discovery library, the select algorithms are applied to the case study presented in the Metamodel Library II [21], where the whole Preday Pattern model from the SimMobility Mid-Term simulator [18] used in Case study 1 is here used as a showcase. Therefore, the **Relationships** (R) are defined by the model that returns the whole activity schedule for each agent in the simulation.

Gender equity is the focus of this case study. The **Exogenous uncertainties** (X), for this example, are defined as:

- **Female income:** a decrease/increase from 10% to 200% in women's income.
- **Beta_female_travel**: a decrease/increase from 10% to 200% in a coefficient that represents the willingness of women to go out of the house to perform any type of activity.
- **Uncertainty in the distribution of driver licences in women:** a coefficient that increases from the baseline case to 100% of women having a driving license.
- **Uncertainty in age distribution:** a coefficient that increases the age of a certain percentage of the agents from 0% to 100% of the population.

The output values are the total number of trips performed by men and women using public transportation. The **Measures** (M) of performance are selected as the future cases where the output value is lower than the 20 percentiles of their computed points. In other words, the cases where the number of public transport trips for men or women is the lowest within the predicted ones. By defining the performance measure in this way, we have a better intuition of the range of realistic output that we can aim for as policymakers.

As **Policy levers** (L), we considered including a public transport policy that makes public transport free for everyone. Since women tend to be more dependent on public transport, we will explore how the difference in gender mobility is affected by these changes.

Following the framework in Figure 6, we explain the code and performance of the algorithms presented in Section 2.

3.2.1 Algorithm 1

For this algorithm, we combine the PRIM algorithm with targeted sampling, where a new box is constructed at each iteration of the algorithm, and a new point is sampled from the borders or the inside of the corresponding PRIM box. The files supporting the execution of this algorithm for this second version of the case study are available at the repository address from Section 4, and are the following:

- SD_1.py:
  1. Load an initial 10 LHS input sample points and compute their corresponding simulation output value.
  2. For a given number of iterations:
     a. Fit a new R.F. with the initial points + new sampling points every third new iteration.
     b. Calculate the posterior in 40 LHS sample points of the input space.
     c. Perform PRIM in these 40 LHS computed points.
     d. Uniformly sample from the simulation model a point inside or on the border of the selected PRIM box.
     e. Compute the output of the selected point given the simulation model and add it to the initial dataset

  Repeat these steps until the stopping criteria (normally defined by the number of iterations) is reached.
  3. Fit a R.F. with the final dataset of simulation points.
  4. Calculate the value of the posterior in 30 LHS samples of the input space

- PRIM_plots_1_dim.ipynb and PRIM_plots_2_dim.ipynb: two notebooks that, given the previously computed points, plot the points distributions presented in for each output and find the final box by performing the PRIM algorithm over the final posterior distribution.

Figure 7 shows that half of the sampled points are indeed vulnerable cases. Thus, this algorithm performs a more SD-targeted sampling technique, allowing us to have a more accurate posterior distribution computed in 40 LHS.

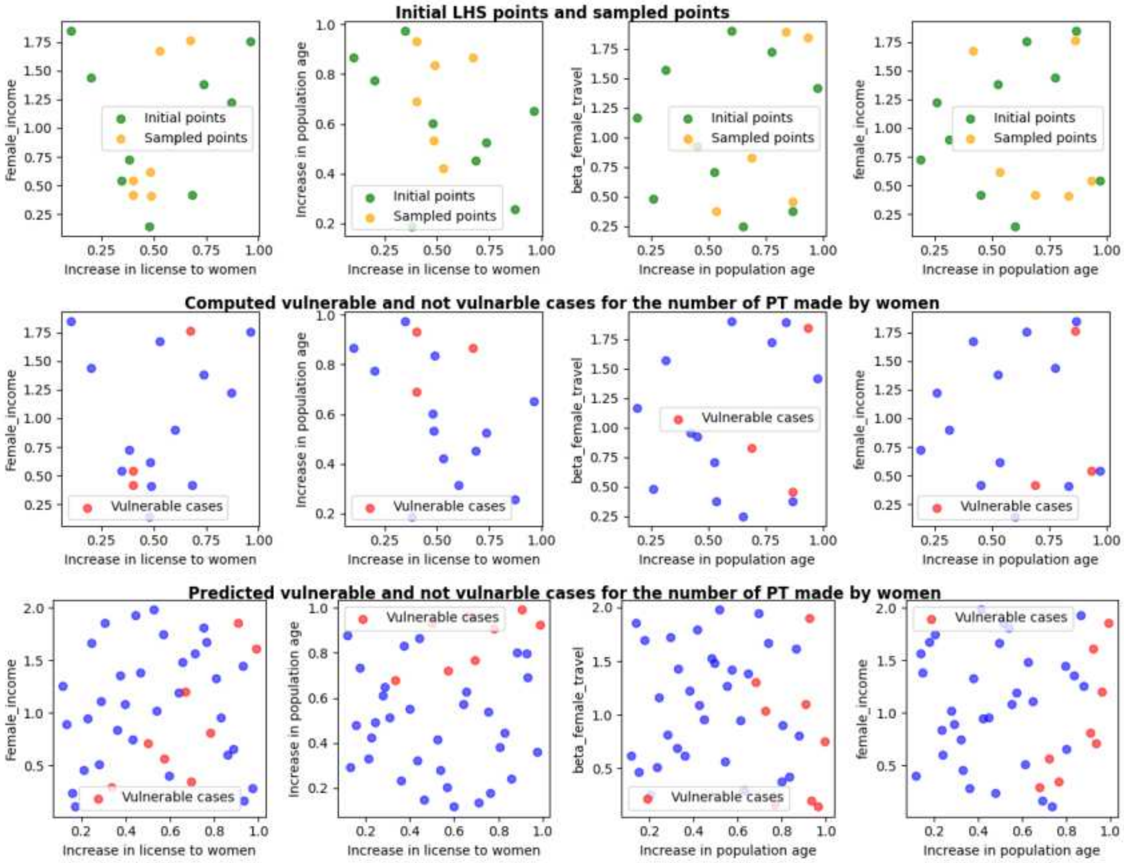

Figure 7. Figure 7. The upper plot shows the distribution of the initial (n_init = 10) and sample points (n_sampled = 6). The lower plots show the distribution of vulnerable cases (P.T. trips > percentile 20) when public transportation is free for everyone using algorithm 1

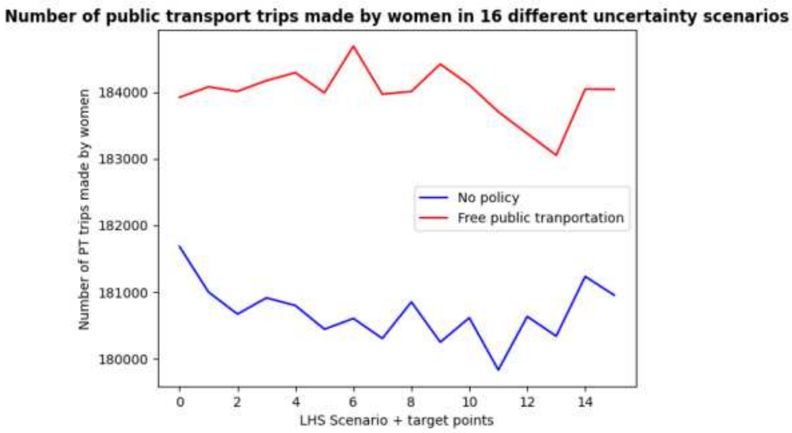

Figure 8. Number of P.T. trips made by women before and after policy for different uncertainty scenarios

In Figure 8, we can see how the free public transport policy causes an increase in the public transport trips made by women for different future scenarios. However, to have a better understanding of how this policy also increases the general mobility of women, in

Figure 9, we plot the total number of trips performed by women. We can see how, in most cases, the selected policy also increases women's mobility, thus achieving our goal of providing women with more travel possibilities.

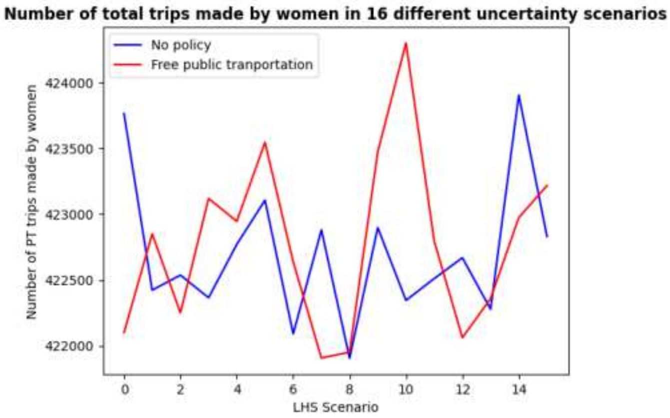

Figure 9. Number of total trips made by women before and after the policy for different uncertainty scenarios

Finally, figure 10 shows the final PRIM box provided by the algorithm when computed in the final 40 LHS points. Given the PRIM results, it seems that the bigger the increase in the age population (dimension 1), the lower the P.T. trips performed by women. On the other hand, an increase in the car license in women (dimension 0) from 30% to all women having a car license also reduces the P.T. trips they performed, as we would expect since they have access to more means of transportation.

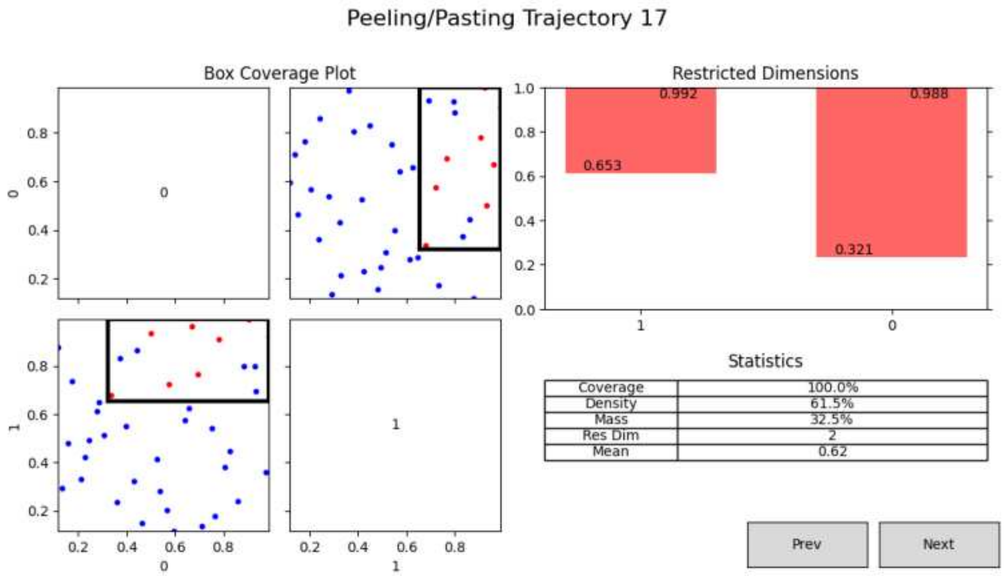

Figure 10. PRIM box found on the posterior distribution of women P.T. trips of 40 LHS samples using algorithm 1

Moreover, for this case study, we also investigate the trips performed by men to better understand gender differences and how the proposed policy interacts with them. In

Figure 11, we can see the total P.T. trips performed by men, while Figure 12 shows the total number of trips performed by men for the same future uncertain scenarios. Applying the free public transport policy results, as expected, in an increase in the P.T. trips made by men; however, it should be noted that this does not generally increase the total number of trips made by men in future scenarios. This could mean that men are less dependent on public transport to perform their daily activities.

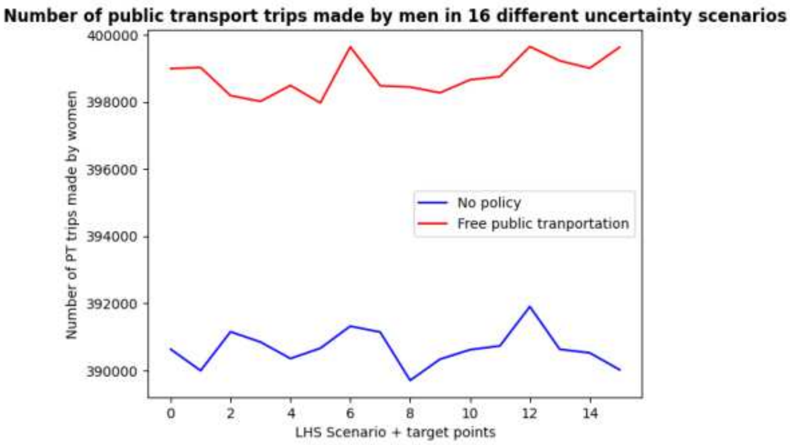

Figure 11. Number of P.T. trips made by men before and after policy for different uncertainty scenarios

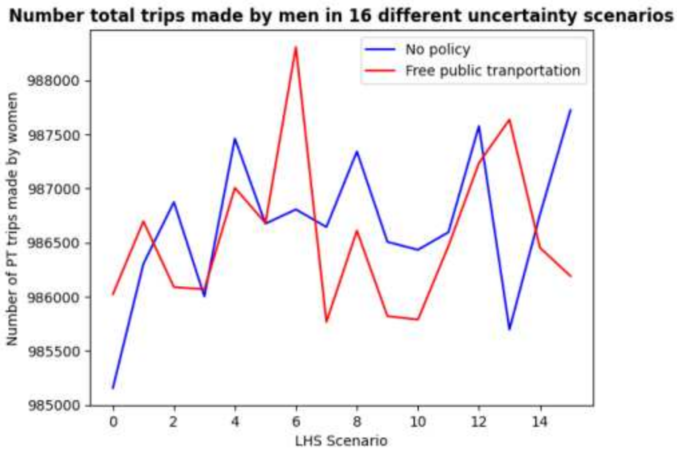

Figure 12. Number of total trips made by men before and after policy for different uncertainty scenarios

Finally, we also perform the PRIM algorithm for the vulnerable cases where the P.T. trips performed by men are lower than the 20 percentile. In Figure 13, we observed that the lower the income in women (dimension 3), the lower the number of P.T. trips performed by men, which makes sense since if the women have less access to the car in the household, it will be the men that will use it, causing a decrease in their P.T. trips. This is combined with values of the beta_female_travel (dimension 2) above and below the baseline, which indicates similar values of women's willingness to travel as the baseline. As in the women's case, an increase in the population age (dimension 1) will cause a decrease in the P.T. trips.

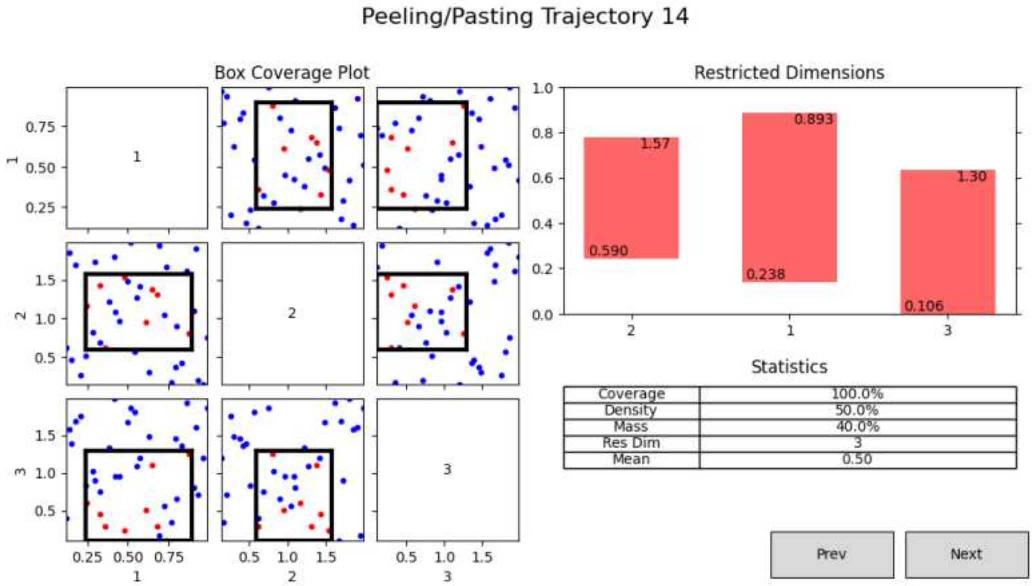

Figure 13. PRIM box found on the posterior distribution for the number of trips made by men on 40 LHS samples using algorithm 1

3.2.2 Algorithm 2

- ModelSD_AL_PRIM_borders.py: performs the same algorithm as before but samples new points only from the borders of the boxes at each iteration.
- PRIM_plots_1_dim_alg_2.ipynb and PRIM_plots_2_dim_alg_2.ipynb: two notebooks that, given the previously computed points, plot the points distributions presented in for each output and find the final box by performing the PRIM algorithm over the final posterior distribution.

Figure 14 shows that the distribution of the sampled points is mostly in the borders of the space, where the PRIM boxes tend to be allocated when vulnerable cases are more spread around the space. It is also interesting to highlight that for this algorithm, most of the sampled points are not vulnerable cases (x < percentile 20), which could indicate that algorithm 1 is more accurate than algorithm 2 in sampling vulnerable points when there are fewer sample points or when the structure of the vulnerable cases is not clear from the begging.

Figure 15 shows the final PRIM box found by the algorithm when computing the posterior in 40 LHS points. Similar to Figure 10, it shows that an increase in the number of car licenses held by women (dimension 0) and an increase in the population age (dimension 1) will decrease the number of P.T. trips performed by women. However, it also adds a wide range in the values of beta_female_travel, which could contribute to a decrease in the number of P.T. trips by women. In our opinion, this could be a misinterpretation caused by the sampled picked point that affects the correct performance of the metamodel.

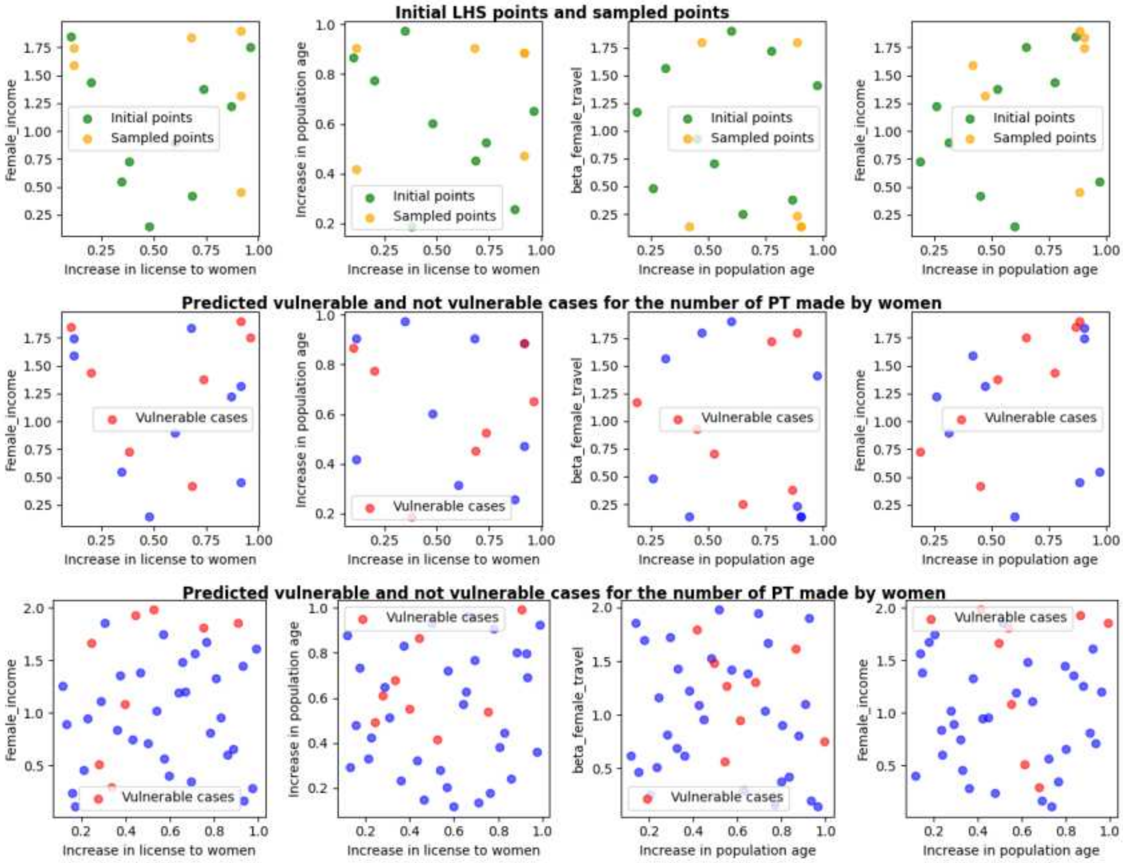

Figure 14. The upper plot shows the initial (n_init =10) and sample points (n_sampled =6). The lower plots show the distribution of vulnerable cases (P.T. trips > percentile 20) when public transportation is free for everyone using algorithm 2

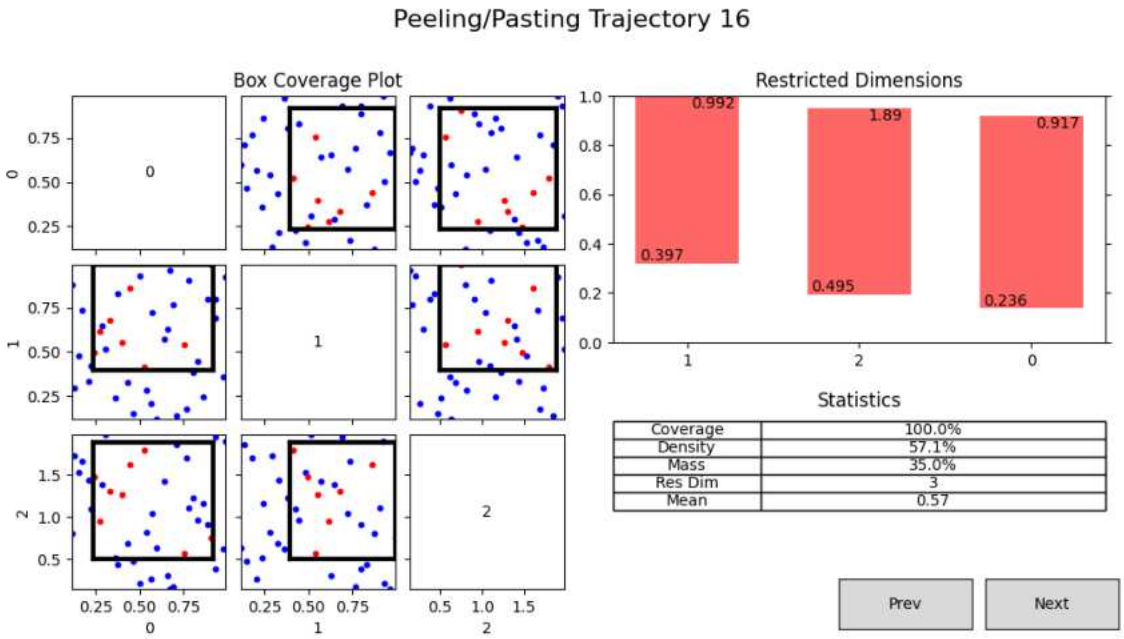

Figure 15. Final PRIM box found on the posterior distribution for the number of trips made by women using algorithm 2

We repeat the same process for the P.T. trips made by men. Figure 16 presents the distribution of the sampled points and the vulnerable cases. Here, we observed again that most of the sample points are also in the borders, and only one of the picked points is indeed a vulnerable case, which could mislead the distribution of the metamodeling results.

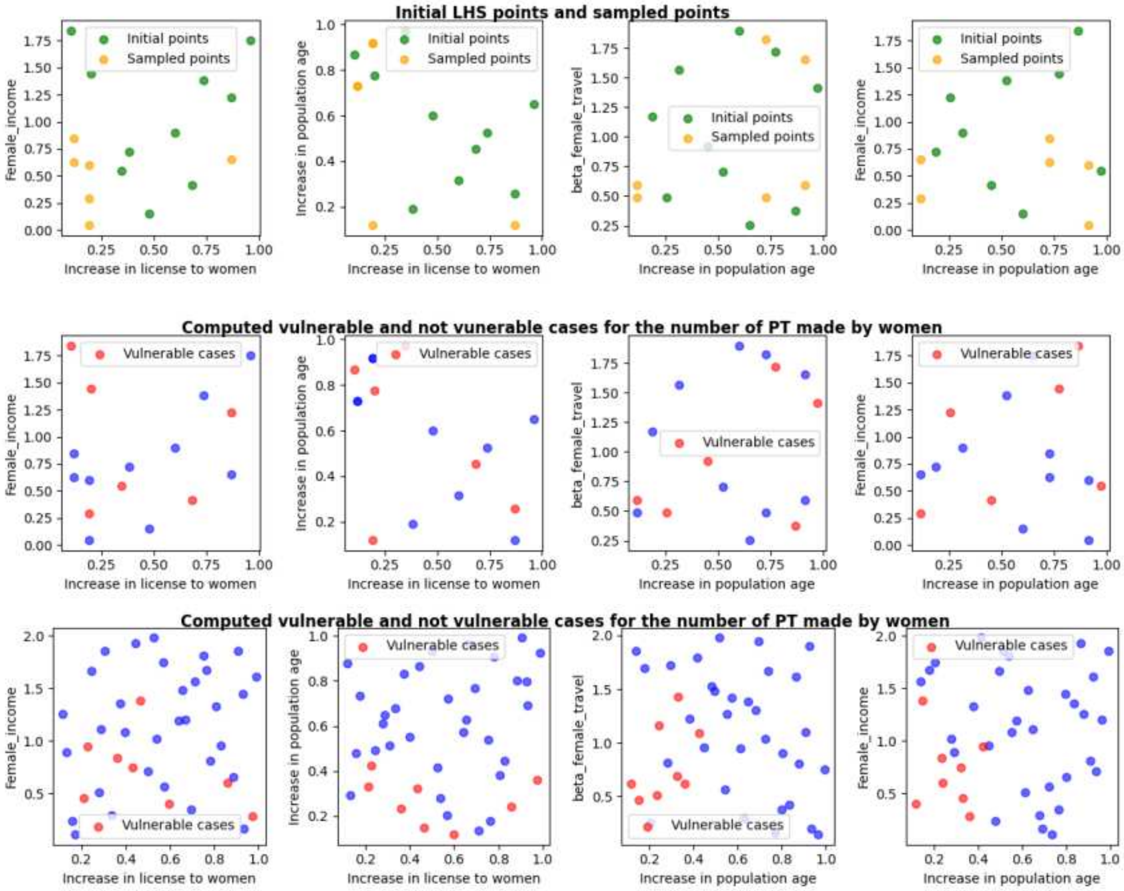

Figure 16. The upper plot shows the initial (n_init =10) and sample points (n_sampled =6). The lower plots show the distribution of vulnerable cases (P.T. trips > percentile 20) when public transportation is free for everyone using algorithm 2

Figure 17 shows the final PRIM box found by the algorithm for the cases where the P.T. trips by men are the lowest. While some moderate values of beta_female_travel (dimension 2) match with the previous box found by algorithm 1, this box differs from the previous one found by adding the percentage of women with car licenses (dimension 0); where more women with car licenses, result in less P.T. trips performed by men, and lower percentages of the older population (dimension 1) also result in less P.T. trips performed by men.

Further investigation of the results, by either sampling more points or changing the configuration of the metamodel, should be done to assess the real performance of the given algorithm.

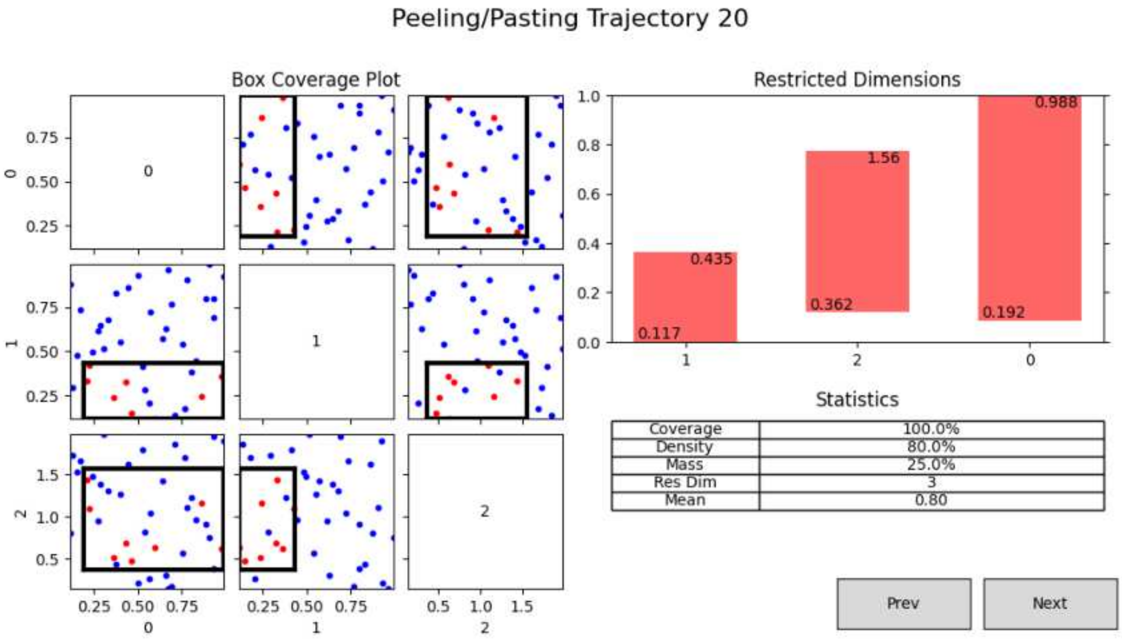

Figure 17. Final PRIM box found on the posterior distribution for the number of trips made by men using algorithm 2

3.2.3 Algorithm 3

This algorithm performs E.I. in a G.P. metamodel. Since we aim to find the regions of the input space where the public transport trips are lower than a given target, we change the simulation to return the number of trips in negative, which is the mathematical equivalent of minimizing instead of maximizing the output values.

- EI.py: a Python file that computes the E.I. algorithm with 10 random initial and 6 query samples.
- Model_BO.py: given the coordinates of the four uncertainty dimensions, it computes the number of P.T. trips in negative.
- EI.ipynb: a notebook that visualizes the sampled points and computes the final PRIM box.

This is an example of a strategy favouring more exploitation than exploration.

Figure 18 shows the distribution of the sampled and vulnerable cases employing algorithm 3. Here, only 2 out of the 6 sampled points are vulnerable cases. The lower plot shows the predicted value computed from the picked points using the same Random Forest metamodel as in the other algorithms.

In Figure 19, we presented the PRIM box found by the algorithm, which has a similar structure to the box in Algorithm 2. As the population ages and the number of car licenses issued to women increases, the number of P.T. trips made by men decreases. This trend is observed when values are close to the baseline for the beta_female_travel variable.

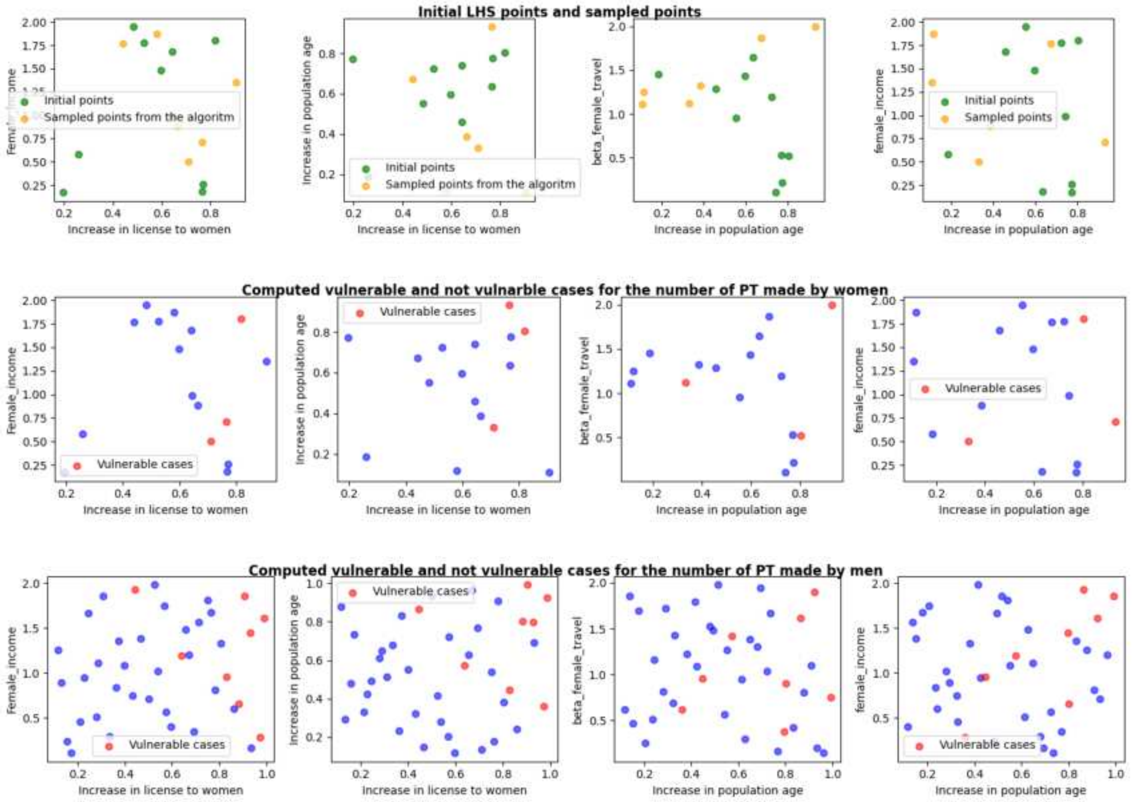

Figure 18. The upper plot shows the initial (n_init =10) and sample points (n_sampled =6) with algorithm 3. The lower plots show the distribution of vulnerable cases P.T.T trips > percentile 20) when public transportation is free for everyone using algorithm 3

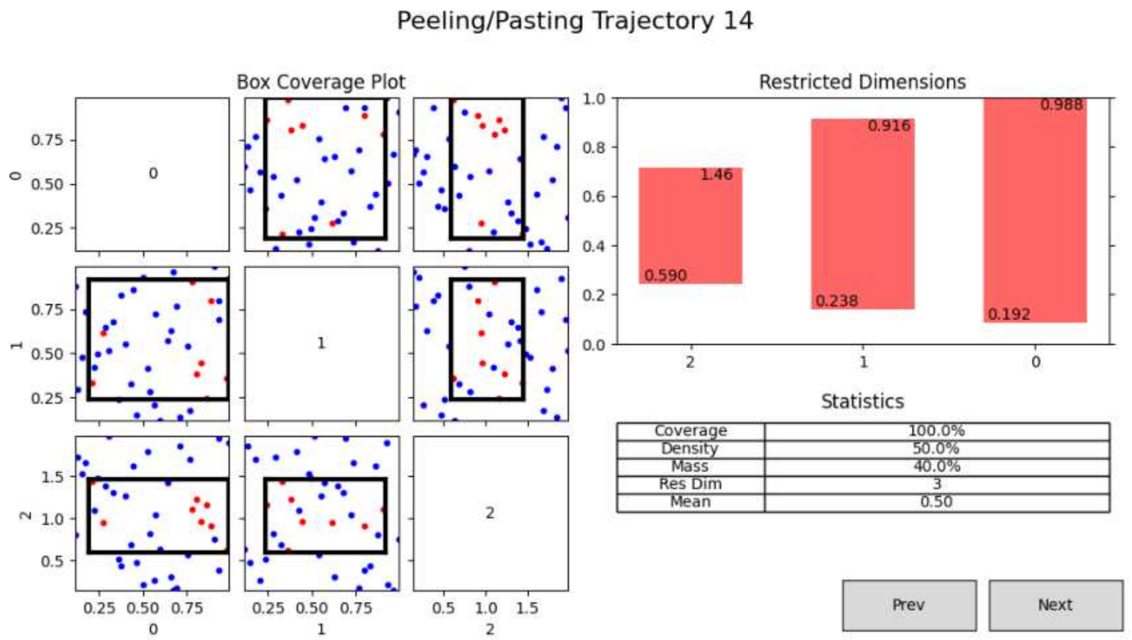

Figure 19. Final PRIM box found on the posterior distribution for the number of trips made by women using algorithm 3

The same process is repeated for the number of public transport trips made by men.

Figure 20 shows the distribution of the picked and vulnerable points, where we can point out that two picked cases are also vulnerable. Finally, in Figure 21, we can see the final PRIM box found by the third algorithm for the number of P.T. trips made by men. While it identifies the same dimensions as in algorithm 2, the structure of the box is a bit different. It shows that an increase in driving licenses for women results in a reduction in public transport trips, especially when combined with an ageing population and a wide range of values for the coefficient of female travel.

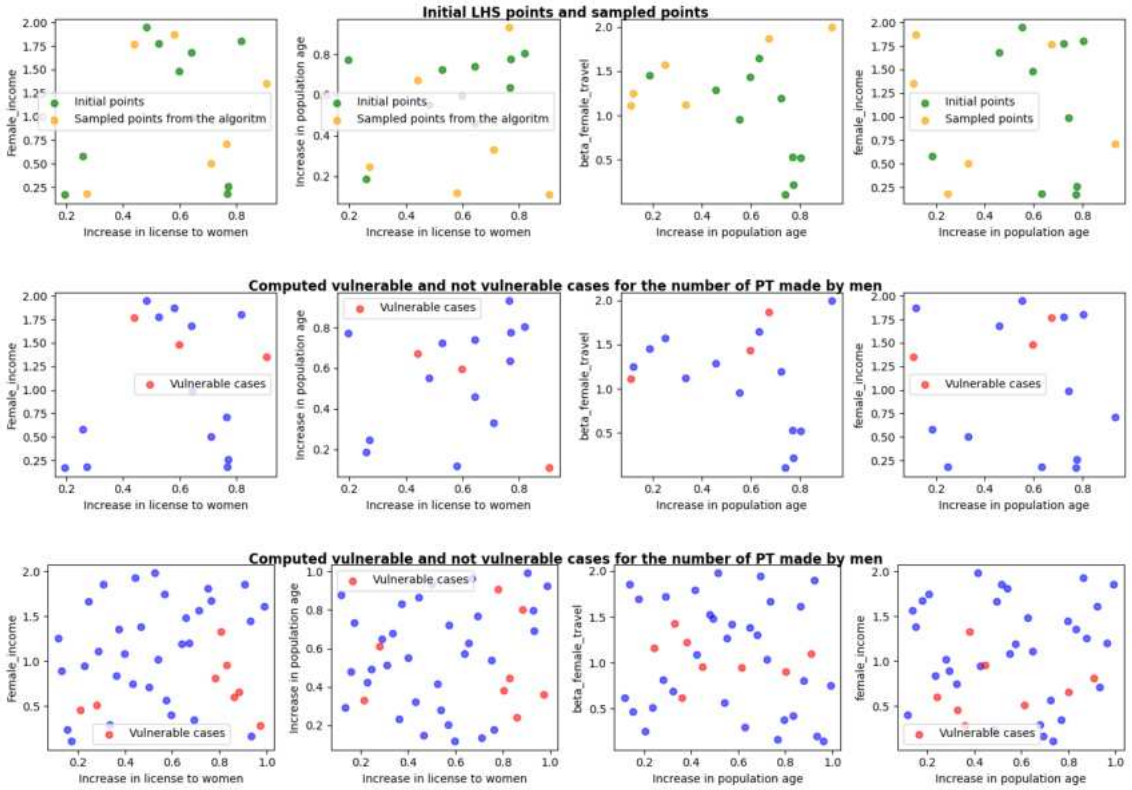

Figure 20. The upper plot shows the initial (n_init =10) and sample points (n_sampled =6) with algorithm 3. The lower plots show the distribution of vulnerable cases P.T.T trips > percentile 20) when public transportation is free for everyone using algorithm 3

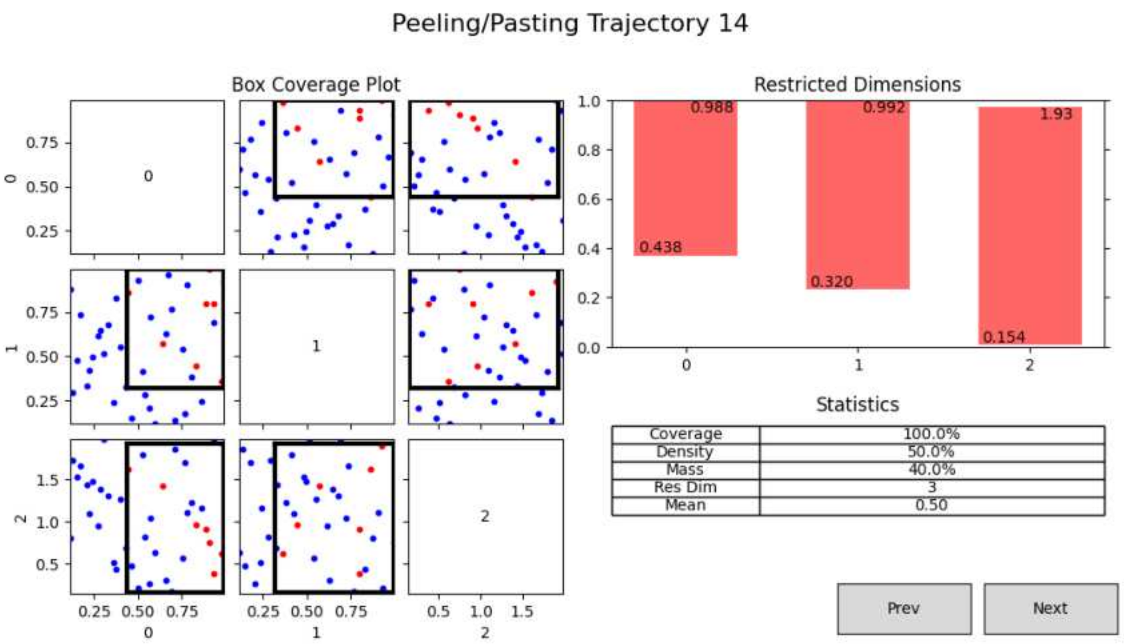

Figure 21. Final PRIM box found on the posterior distribution for the number of trips made by men using algorithm 3

3.3 Next Steps and the Three Case Studies

We presented the steps needed to implement the S.D. framework for the three case studies presented in deliverable D.7.1 and applied it to a Virtual city [22] for Case Study 1. [19] Algorithm 4 was not implemented in this deliverable due to time constraints.

For future case studies, stakeholders will be involved in selecting feasible policies and assessing the feasibility of the S.D. Various user interaction methods will be examined to ascertain a better balance between automation and human expertise, thereby ensuring the credibility of the outcomes.

Each solution will be constructed by a policy or a combination of them with the specification of the uncertainty factors range where the conditions for the success of the given policies would be no longer guaranteed, and, therefore, the policies should be changed or modified. This information could be used as a road map for policymakers to plan efficiently for uncertain futures.

This process will be repeated for each case study to find the best compromise solution targeted to their specific objective.

The final results will be summarised in the Showcase results report and Policy Recommendations Deliverable D.7.7.

4. Library

The library with the preliminary code for Scenario Discovery can be found at: https://github.com/emotional-cities/D.7.5.Scenario-Discovery-2

The specified requirements include the following packages:

- Pandas: version 1.5.3
- Numpy: version 1.24.2
- Matplotlib: version 3.7.1
- Pickleshare: version 0.7.5
- Sklearn: version 1.2.2
- Torch: version 2.0.0
- Gpytorch: version 1.9.1
- pip: version 23.2.1

The library can be installed via PIP or Anaconda; tutorials on the libraries' usage are widespread on the internet.

The library is firstly intended to be used by other partners of the eMotional Cities project. Throughout T7.6, "Showcasing Policy Discovery for Improved Urban Health", the library developers will support the other partners in the library use for scenario exploration through walkthrough sessions and technical meetings.

Furthermore, the open code is intended to reach other technical stakeholders interested in AI-supported decision-making. The prepared notebooks and readme files make sure there is practical guidance for the public on the use of the library.

5. References

[1] Walker, W.E., P. Harremoës, J. Rotmans, J.P. van der Sluijs, M.B.A. van Asselt, P. Janssen, M.P. Krayer von Krauss (2003). "Defining Uncertainty: A Conceptual Basis for Uncertainty Management in Model-Based Decision Support", *Integrated Assessment*, Vol. 4, No. 1, pp. 5-17.

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

[3] Walker, Warren E., Robert J. Lempert, and Jan H. Kwakkel (2013). "Deep Uncertainty", entry (pp. 395-402) in Gass, Saul I. and Michael C. Fu (eds.), Encyclopedia of Operations Research and Management Science., 3rd Edition, New York: Springer.

[4] Taleb, N.N. (2007). *The Black Swan: The Impact of the Highly Improbable*, Random House, New York.

[5] David G. Groves, Robert J. Lempert, A new analytic method for finding policy-relevant scenarios, Global Environmental Change, Volume 17, Issue 1, 2007, Pages 73-85, https://doi.org/10.1016/j.gloenvcha.2006.11.006.

[6] Lempert, Robert J., Benjamin P. Bryant, and Steven C. Bankes, Comparing Algorithms for Scenario Discovery. Santa Monica, CA: RAND Corporation, 2008. https://www.rand.org/pubs/working_papers/WR557.html.

[7] Bryant, B. P., & Lempert, R. J. (2010). Thinking inside the box: A participatory, computer-assisted approach to scenario discovery. Technological Forecasting and Social Change, 77(1), 34-49.

[8] Friedman, J.H., Fisher, N. I. (1999). Bump hunting in high-dimensional data. *Statistics and Computing* 9, 123–143. https://doi.org/10.1023/A:1008894516817

[9] S. Dalal, B. Han, R. Lempert, A. Jaycocks, A. Hackbarth, Improving scenario discovery using orthogonal rotations, Environmental Modelling & Software, Volume 48, 2013, Pages 49-64, ISSN 1364 8152,https://doi.org/10.1016/j.envsoft.2013.05.013.

[10] Kumar, P., Gupta, A. Active Learning Query Strategies for Classification, Regression, and Clustering: A Survey. *J. Comput. Sci. Technol.* **35**, 913–945 (2020). https://doi.org/10.1007/s11390-020-9487-4

[11] Peter I. Frazier. “A Tutorial on Bayesian Optimization”. In: (2018). doi: 10.48550/ARXIV. 1807.02811. url: https://arxiv.org/abs/1807.02811

[12] Frazier, Peter & Wang, Jialei. (2015). Bayesian Optimization for Materials Design. 10.1007/978-3-319-23871-5_3.

[13] C. E. Rasmussen and C. K. I. Williams. Gaussian Processes for Machine Learning. The MIT Press, 2006. ISBN: 026218253X. url: www.GaussianProcess.org/gpml.

[14] Pugh, J. K., Soros, L. B., & Stanley, K. O. (2016). Quality diversity: A new frontier for evolutionary computation. Frontiers in Robotics and A.I., 3, 40.

[15] Jean-Baptiste Mouret and Jeff Clune. “Illuminating search spaces by mapping elites”. In: (2015). arXiv: 1504.04909.

[16] Torres Lahoz, L., Lima Azevedo & Camara, F., (2023). *eMC_D72_Metamodeling_library_I_20231117.* eMOTIONAL Cities: mapping the cities through the senses of those who make them.

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

[18] 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, U.S.

[19] 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.

[20] Vassiliades, V., Chatzilygeroudis, K., Mouret, J-B. Using Centroidal Voronoi Tessellations to Scale Up the Multi-dimensional Archive of Phenotypic Elites Algorithm. https://doi.org/10.48550/arXiv.1610.05729

[21] Torres Lahoz, L., Lima Azevedo & Camara, F., (2023). *eMC_D72_Metamodeling_library_II.* eMOTIONAL Cities: mapping the cities through the senses of those who make them.

[22] Basu, R., Ponce-Lopez, R., Ferreira, F. (2021). A framework to generate virtual cities as sandboxes for land use-transport interaction model. The journal of transportation and land use. Vol. 14 No. 1. https://doi.org/10.5198/jtlu.2021.1791

eMOTIONAL Cities

Mapping the cities through the senses of those who make them