DYNAMIC SUBSET SENSITIVITY ANALYSIS FOR DESIGN EXPLORATION

Laura Hinkle
Gregory Pavlak
Nathan Brown

Department of Architectural Engineering
Pennsylvania State University
Engineering Unit A, 104 The Pennsylvania State University, University Park, PA 16802, USA
{luh183, gxp93, ncb5048}@psu.edu

Leland Curtis

Independent Researcher

lelandcurtis88@gmail.com

ABSTRACT

This paper presents a method for dynamically assessing parametric variable importance and likely influence on performance objectives as a large, precomputed design space is filtered down to explore more specific problems. Custom parametric models coupled with performance simulation can support early design, but they can be inflexible and are not always created in practice due to time and other constraints. Large parametric datasets of previously simulated design subproblems could thus make performance-based modeling more accessible, but they can have too much information and fail to focus on supporting design decisions for specific variables and ranges. Using a parametric daylight room model as an example, we first train a linear model tree. As variable bounds are filtered and adjusted by a designer, remaining coefficients are interpolated to provide an adjusted variable importance for the new domain.

Keywords: sensitivity analysis, design variable importance, reusable parametric models

1 INTRODUCTION

With the integration of simulation software into visual programming environments, parametric modeling and design space exploration are increasingly used in early building design. Despite their limitations, parametric models can explore a range of options within an initial design concept, highlighting performance trends or tradeoffs when the concept is still flexible. Many design approaches have been proposed at this stage, including design catalogues (Brown and Mueller 2017), interactive and automated exploration (Ferrara et al. 2014), and machine learning for visualization (Wortmann 2017; Danhaive and Mueller 2021). However, generating a custom parametric model is time-consuming, and there can be considerable uncertainty in the design variables, simulation settings, and metrics, even if the problem is simplified. Thus, a practical barrier is a potential mismatch between what information designers want out of a parametric model and when a model is able to supply it. Design practice moves fast, and tools can get left behind if they do not provide salient information of useful accuracy at the right time. Even with newly available tools, there remains a need for fast feedback and guidance on the key early design decisions that impact performance (Bambardekar and Poerschke 2009).

Researchers have worked to create design models for repeated use. Ideally, these large models contain many variables and configurations, but could be filtered down to provide useful feedback for a given problem. For these models, machine learning can predict performance for a new design based on existing simulation data. Yet while a general model is easier to reuse, it likely gives lower resolution feedback for early geometric, material, or other design decisions. As such, many designers might prefer to simply use a general parametric model to discover which variables tend to "matter" in terms of performance—how likely they are to improve metrics, and where good settings tend to be—before modifying the design outside a restrictive parametric framework. This information must respond to customization of a large dataset to the specific design project, which may only include a subset of variable settings.

The task of determining variable importance and providing information about performance response is a type of sensitivity analysis. In architecture and engineering, a dynamically adjustable sensitivity analysis may be a form of feedback that, while lower resolution than accurate performance prediction, may match the speed of design and make parametric simulation more widespread in practice. Sensitivity analysis helps identify and prioritize significant model inputs, enabling designers to focus on key decisions before the design approach consolidates around a sub-optimal, or even counter-productive, strategy. It can also point designers in the right direction for specific model settings. Yet a traditional design space sensitivity analysis might still require building a custom parametric model for a problem, running an analysis of variable importance, adjusting the model as design criteria emerges, and re-running the analysis.

In response, this paper presents a new approach called dynamic subset sensitivity analysis for common early design problems. Examples of recurring design problems that could be modeled and reused for many projects, as long as variables can be restricted or eliminated, include a daylit room, shoebox energy calculation, or structural framing layout. In this approach, a general design space is first established through parametric simulation. Next, machine learning is used to train a linear model tree with a structure that enables repeated filtering of variable ranges and rapid recalculation of variable importance and sensitivity. Finally, visualizations of each calculation give immediate information on the relationship between design drivers and performance metrics as bounds are adjusted. This process is explained and tested on a room daylight model. The ultimate goal is to develop an interactive tool that tracks with the speed of design and while providing multi-resolution sensitivity information. The desired outcome is greater participation by architects and consultants in performance-driven tools, leading to better performing buildings.

2 LITERATURE REVIEW

This section explains relevant background literature in machine learning and sensitivity analysis (SA) for building design, construction, and operation. SA can help with decisions by explaining how the uncertainty in the response can be allocated to uncertainties in the variables. It has been used most often to support model calibration (Yang et al. 2016) or decision-making in design and operation (Brembilla, Hopfe, and Mardaljevic 2018; Samuelson et al. 2016; Brown and Mueller 2019), often attempting to find the most influential variables. This paper focuses on decision-making. If a design variable is highly influential to the performance response, it is considered a critical decision and may require multi-disciplinary input. On the other hand, if the variable is less influential to the response, it is more flexible and a lower priority.

Several SA approaches have been applied to building performance problems with the goal of computing variable ranking, degree of importance, or interactions (Nguyen and Reiter 2015). These include one-at-a-time (OAT), screening, regression-based, and variance-based approaches. While OAT methods are suitable for other fields, they are not recommended for building performance analysis due to the possible nonlinearity of building systems (Pang et al. 2020). Meanwhile, screening methods such as global OAT and Morris method capture nonlinear effects and have been used to address a broad range of problems, ranging from improving building life cycle assessment (Pannier, Schalbart, and Peuportier 2018) to thermal comfort (De Wit and Augenbroe 2002). Regression-based methods are straightforward, using standardized regression coefficients as variable importance. Some energy model researchers have demonstrated regression-based variable selection procedures such as stepwise regression (Arababadi et al. 2015). The main drawback of linear regression is the linearity condition, which may not be satisfied depending on the building system. Finally, variance-based approaches tend to achieve higher accuracy at a computational cost.

These methods compute influence over the entire variable domain. As the design space, or variables' domain, is refined during early design, the initial SA may no longer be accurate. Some researchers approached this issue by retraining the underlying regression model on the restricted variable domain (Gao, Mae, and Taniguchi 2020). However, depending on how the domain is restricted, predictions are not consistently accurate; there may be few remaining training points or the data behaves differently in the specified region. Most importantly, retraining may cause a lag which is disruptive to the decision-making process. Another approach is to leverage information from the original model and filter based on design criteria. Regional sensitivity analysis (RSA) identifies regions in the design space corresponding to particular values of the response. This method has been demonstrated on building energy models (Østergård, Jensen, and Maagaard 2017b; 2017a). However, the analysis was based on an idealized model rather than detailed BPS software and required additional evaluations in limited subsets. Nevertheless, filtering is a valuable design space exploration technique, especially as reusable parametric models emerge as a new research area.

With many existing methods, it is still difficult to quickly and accurately update variable importance as the design is refined. Implications on the design process are also unclear. We thus propose dynamic subset sensitivity analysis for common design problems, as well as potential visualizations and inferences that attempt to provide conceptual designers with useful information throughout the process.

3 METHODS

The main process demonstrated in this paper involves leveraging variable sensitivity in regions determined by a regression tree to quickly return variable information without full model retraining. The overall research procedure is shown in Figure 1. First, a common building design problem was selected and parameterized. Data was generated using a visual programming environment and then used to train a linear model tree. Two additional models were trained, multiple linear regression and traditional regression trees, to evaluate the accuracy of the linear model tree. Next, the variable domains were binned, and a weighted average method was employed to indicate where in the variable range certain variables tended to have a large influence on the response. Finally, variable sensitivity was updated through a new subset sensitivity analysis method that involves interpolating linear model tree leaves as the design space is filtered. Several design scenarios are used to demonstrate the new dynamic subset sensitivity analysis method.

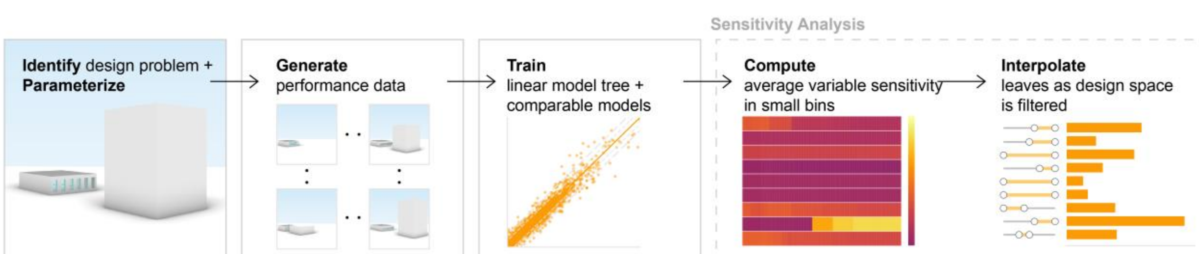


Figure 1: Overall methodology

This paper envisions a future dynamic interface shown in Figure 2. This interface would allow a designer to explore the initial sensitivity analysis across all variable ranges, dynamically adjust the bounds to match their problem, and even select individual variables to view the constituent leaf models that form the basis of the sensitivity visualizations. As an example, the mock-up in Figure 2 shows an initial sensitivity analysis, dynamic subset sensitivity analysis, and detailed sensitivity analysis in terms of the variable panel width. The dynamic subset sensitivity analysis shows updated feature importance based on the adjusted bounds and a selected available design. Further details about each component are provided in the results.

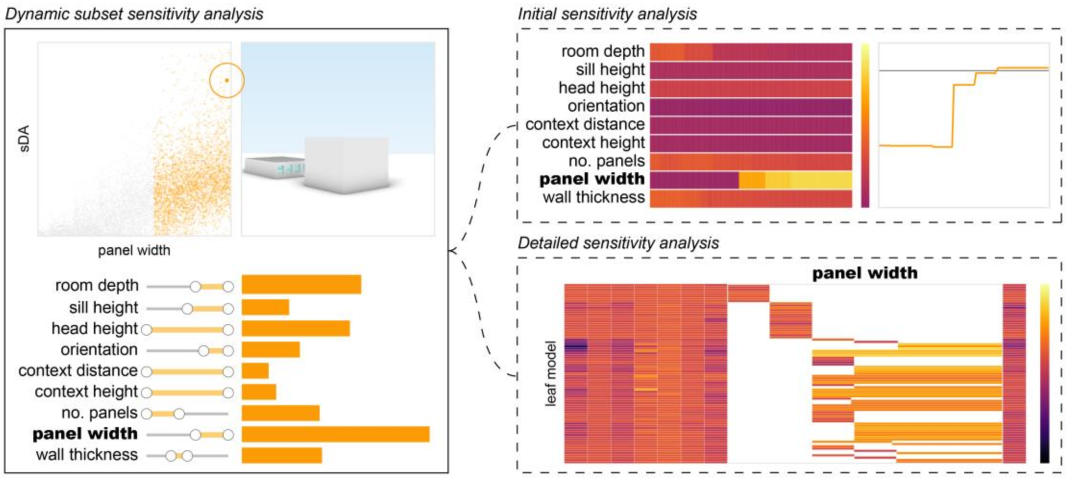
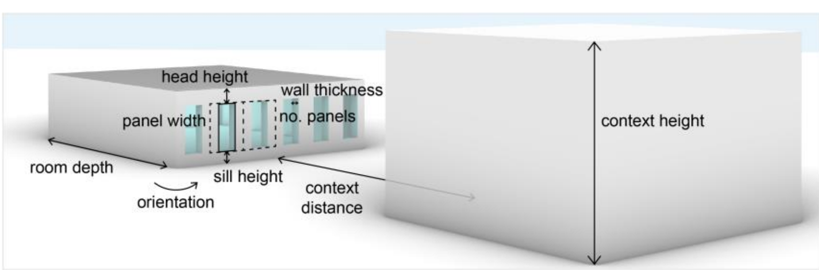


Figure 2: Potential tool interface based on the method for dynamic subset sensitivity analysis

3.1 Problem Selection and Parametrization

A single room model was selected to demonstrate the method due to its repeated use in practice. The ubiquity of room models in the United States stems from the simulations required to obtain LEED v4 Daylight credits. Significant efforts are made to model and simulate daylight in qualifying spaces. The daylight room model was modeled parametrically in Grasshopper. The surface properties were assigned according to LM-83 guidelines, and a typical double-pane low-e window was selected with 61% visible transmittance. The automated shade fabric has 7.2% visible transmittance and 6.6% permeability. The room width and height were held constant at 9 m and 3 m, respectively. The design space consisted of nine variables: room depth, sill height, head height, orientation, context distance, context height, number of panels, panel width, and wall thickness (Figure 3). The upper and lower bounds were chosen to provide enough flexibility for repeated use within modern construction standards. The dataset is intuitive and fairly linear, although additional repeated early design problems could be explored in the future.



Variable	Minimum	Maximum
room depth (m)	6.00	15.00
sill height (m)	0.10	1.10
head height (m)	0.10	1.10
orientation (deg from south)	0.00	360.00
context distance (m)	3.00	15.00
context height (m)	0.00	15.00
number of panels	1	20
panel width (m)	0.10	0.90
wall thickness (m)	0.20	1.00

Figure 3: Common design space: daylight room on building facade

3.2 Data Generation

To create the general dataset, spatial daylight autonomy (sDA) at 300 lux was simulated using ClimateStudio in Grasshopper. The sensor spacing was 1 m and the workplane was positioned 0.762 m above floor finish. Within the path-tracing settings, the number of rays emitted for each sensor at each pass was 500. The Radiance parameters considered up to 6 ambient bounds before discarding a ray. The design space was sampled at N =12,500 using Latin Hypercube sampling. Simulations were conducted in

Pittsburgh, PA, and Phoenix, AZ, to yield two individual datasets. The Pittsburgh sky is often overcast, while Phoenix is typically clear, and they are located at different latitudes. In the results section, variable sensitivities are compared through different potential design paths for both cities. Before training the linear model tree, the datasets were split 80/20 for training and testing. All variables were normalized from 0-1.

3.3 Leveraging Linear Model Trees

The next step is to create regression trees for the datasets. They are built through recursive binary splitting, where predictor X_j is split at cutpoint s such that splitting the predictor space into the regions $\{X \mid X_j < s\}$ and $\{X \mid X_j \geq s\}$ leads to the greatest reduction in the residual sum of squares (RSS) in equation (1). Splitting stops based on some threshold and each terminal node, or leaf, contains a model that applies in the j-th region only. For traditional regression trees, the estimated response $\hat{y}_{R_j}$ is the mean response for the training observations in the j-th region. However, this is often an over-simplification of the true relationships. To address this issue, linear model trees use a linear model to estimate the response. By the end of the training process, each leaf node contains its own linear model.

$$RSS = \sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2 , \tag{1}$$

where the outer summation accounts for each variable and the inner summation accounts for all points in the specified region. While previous studies have achieved high accuracy with nonparametric models, it is often not possible to make inferences and inform the building design process. It was hypothesized that linear model trees achieve sufficient accuracy for early design while allowing for dynamic interpretations about variable sensitivity because of how they are constructed.

The hyperparameters for a linear model tree are the maximum depth and minimum number of samples per leaf, which have to be tuned for a given dataset. For this application, the maximum depth was set to 8 and minimum number of samples per leaf was set to 30. At this depth, the model achieved sufficient accuracy, and enforcing at least 30 points per leaf ensured the model was valid. Once the linear model tree was built, the leaves were used to compute average sensitivity in small bins.

3.3.1 Compute Average Variable Sensitivity

The next step is to determine how coefficients of individual leaves should be combined to indicate variable importance. To get a sense of sensitivity over the entire variables' domain, the average linear model coefficient was computed in small bins. The domain of each variable X_j is partitioned into 100 bins of equal length. The m-th bin is denoted by $b_m \coloneqq \left[\frac{m-1}{100}, \frac{m}{100}\right)$, for $1 \leq m \leq 100$. The k-th leaf is denoted by ℓ_k and the number of samples in ℓ_k is n_k. Then, the domain of each variable X_j is constrained by $c_{j,k} \leq X_j \leq d_{j,k}$ in leaf ℓ_k. Let $\theta_{j,k}$ be the original coefficient of X_j in ℓ_k. Then the weighted coefficient restricted to bin $b_{j,m}$ is shown by $\hat{\theta}_{j,k,m}$ and is given by the equation (2).

$$\hat{\theta}_{j,k,m} = \theta_{j,k} * \frac{n_k}{n(b_{j,m})}, \tag{2}$$

where $n(b_{j,m})$ is the number of samples that lie in $b_{j,m}$ for X_j over all leaves. Finally, the weighted coefficient for variable X_j in b_m is given by equation (3).

$$\hat{\theta}_{j,m} = \sum_k \hat{\theta}_{j,k,m} \tag{3}$$

The result is a local sensitivity analysis over the entire domain that can be used to understand changes in the response. Next, the model leaves are used to update variable importance for user-defined intervals.

3.3.2 Interpolate Leaves as Design Space is Filtered

While many data models can return importance metrics, they are often established through training, requiring retraining if the variables and their corresponding bounds are modified. By precomputing linear models in regions determined by the regression tree, the model coefficients can be interpolated to quickly return variable information without full model retraining. If the user-defined intervals correspond exactly to a pre-defined region, variable sensitivity is provided by that model. Otherwise, the model coefficients must be interpolated based on the "agreement" between the user-defined intervals and the variable domains in the leaves. The agreement of the user restricted intervals with the constraints of ℓ_k is given by equation (4).

$$\widetilde{w}_k = \left(\sum_{j=1}^{J} w_{k,j}{}^{\frac{1}{p}}\right)^{p}, \tag{4}$$

where $w_{k,j}$ is the amount of "agreement" of X_j in ℓ_k and $p > 1$ is a hyperparameter. Let $[a_j, b_j]$ be the user-defined interval on X_j. Then, the amount of agreement $w_{k,j}$ is defined as equation (5).

$$w_{k,j} = \frac{\min\{\mathrm{d}_{j,k}, b_j\} - \max\{\mathrm{c}_{j,k}, a_j\}}{b_j - a_j} \tag{5}$$

Without loss of generality, assume $\widetilde{w}_1, \widetilde{w}_2, \ldots, \widetilde{w}_t$ are the top t agreements. The total weight w_k is a function of top t agreements normalized by their sum as equation (6).

$$w_k = \frac{\widetilde{w}_k}{\sum_{k=1}^{t} \widetilde{w}_k} \tag{6}$$

Finally, variable importance was computed using the equation (7).

$$\widehat{\boldsymbol{\theta}} = \sum_{k=1}^{t} w_k \cdot abs(\boldsymbol{\theta}_k), \tag{7}$$

where $\boldsymbol{\theta}_k$ is the linear model coefficients at ℓ_k, and $abs(\cdot)$ is element-wise absolute value of a vector. Note that p and t are hyperparameters that can be tuned based on the dataset. For this dataset, p and t were set to 3 and 10, respectively. For higher values of p, the contrast between the top t agreements becomes sharper. As t approaches the total number of leaves, the impact of individual leaves gets lost due to normalization. On the other hand, if $t = 1$, only one leaf is used, which might not be an accurate model of the user-defined region. Once the intervals are specified, individual predictions are made with the linear model tree itself. Single designs only fall into one leaf, since the regions do not overlap. The prediction is made by the linear model in the appropriate leaf. Once this model has been established, a metric for overall variable importance and visualizations of how performance changes with variable setting modifications can both be returned to a designer without the added time of model retraining. The results section first presents the dataset itself before showing these potential visualizations for the designer.

4 RESULTS AND DISCUSSION

4.1 Model Comparison

Before performing computations with leaf model coefficients, the linear model tree fit was assessed to ensure sufficient demonstration of the method. Additionally, a multiple linear regression model and traditional regression tree were trained to evaluate performance. Figure 4 shows the simulated response versus the predicted response for each model in Pittsburgh and Phoenix. The response for this dataset is sDA, which is a percentage, so the values range from 0 to 1. Although the multiple linear regression model and traditional regression tree make accurate predictions for low sDA values, the overall fit is poor. It is clear that implementing a linear model tree captures nonlinearity in the data thus improving the accuracy of the model. An important assumption of linear regression is that the variables are independent. For both the Pittsburgh and Phoenix datasets, all Pearson correlation coefficients were less than 0.02. This indicates that the variables are indeed independent.

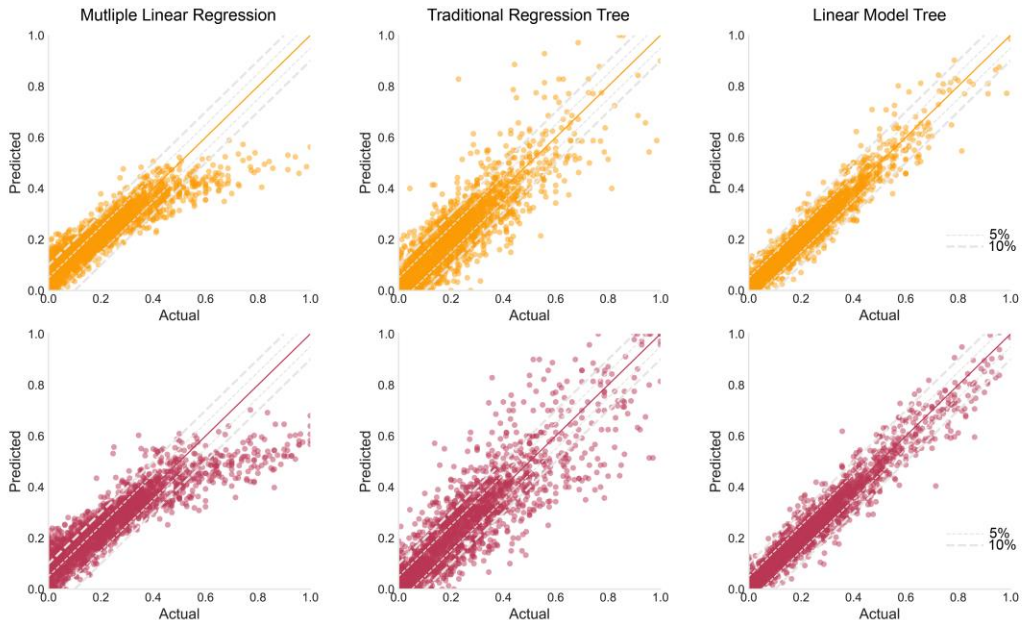


Figure 4: Interpretable model comparison, Pittsburgh (upper) and Phoenix (lower)

4.2 Coefficients Over Variable Domain

After building these models, the average coefficients for each variable were plotted over their domains. Such visualizations could be directly useful for designers, but they contain similar information to later figures. While many variables maintain a constant coefficient over their domain, some show noteworthy changes. In Figure 5, the gray line is the coefficient from the multiple linear regression model, and the orange line is the result from the procedure described in 3.3.1. On average among the leaf models, panel width does not significantly affect sDA until it reaches ~0.50 m. This means that designers can freely choose within 0.10-0.50 m without affecting sDA. The Phoenix graph for panel width shows the coefficient increases more gradually. Similarly, it is not likely changing panel width within this range will greatly affect sDA. The other variable showing substantial changes over its domain is room depth. For both Pittsburgh and Phoenix, room depth greatly influences sDA until it reaches approximately 8.7 m. This information is potentially useful while designing floorplans; for deep rooms, other variables must be adjusted to reach sDA targets.

Figure 6 presents the same data in terms of coefficient magnitude. This visualization shows at which point the variables' relationship with the response changes. Many variables' magnitudes are constant, and in terms of design, these variables hold the same importance throughout. This dataset was selected for its interpretability to demonstrate the method, however, these visualizations may reveal less intuitive changes for highly non-linear data. Since these results demonstrated similar trends for Pittsburgh and Phoenix, the following sections present results in terms of the Pittsburgh dataset.

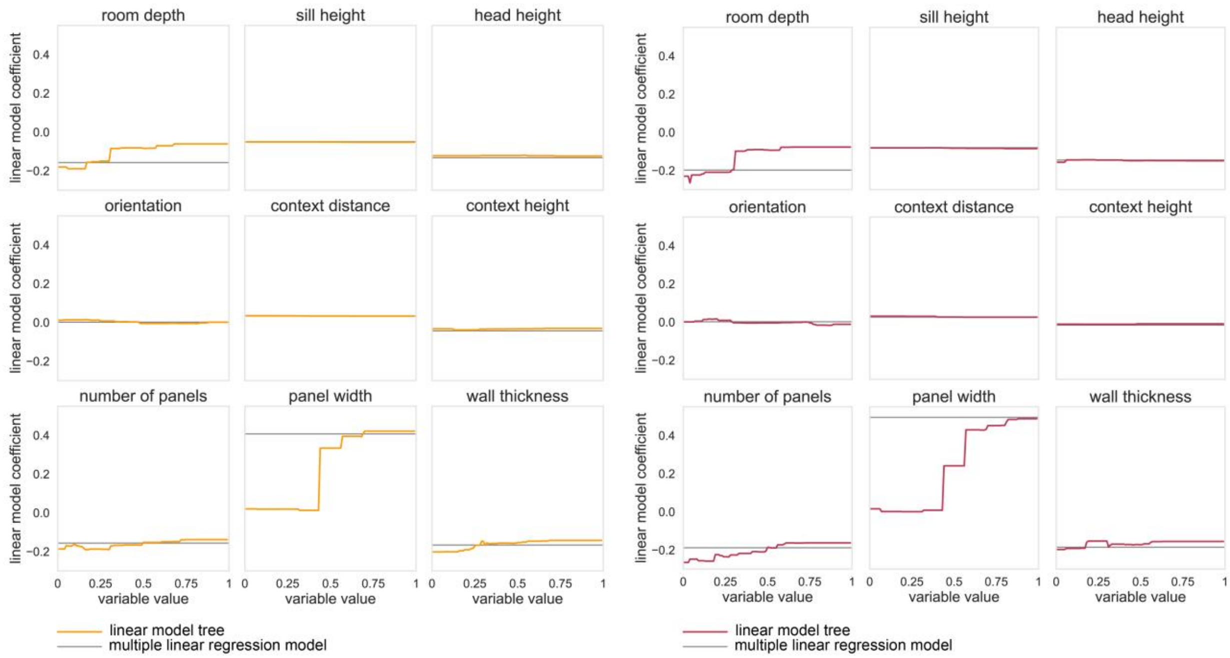


Figure 5: Coefficients on average over entire domain, Pittsburgh (left) and Phoenix (right)

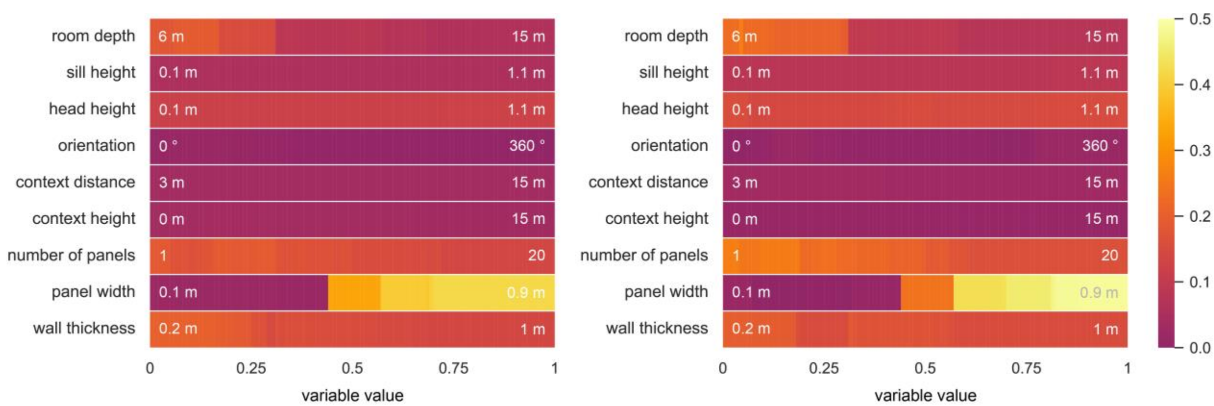


Figure 6: Magnitude of coefficients on average over entire domain, Pittsburgh (left) and Phoenix (right)

4.3 Dynamic Subset Sensitivity Analysis

Data-driven parametric design often involves setting variable domains, generating data, and fitting a prediction model. As the design is refined, variable domains are narrowed until one value is ultimately selected. Previously, the prediction model needed to be re-trained on the subset of data to provide accurate variable importance and support decisions. We achieve subset sensitivity analysis by precomputing linear regression models in regions determined by the tree and then interpolating between regions to estimate the variable importance in the subset. Four examples are shown in Figure 7, which includes a slider for each variable, the user-defined intervals, and variable importance, presenting a potential visualization for a design tool. The images on the right represent three potential designs in the subset: variable lower bounds on the left, variable middle values in the middle, and variable upper bounds on the right. It is important to note that a series of visualizations presented to the designer should show both (1) *which* variables deserve attention (by virtue of producing a large effect on performance, regardless of direction) and (2) *how* such variables tend to affect performance along their domains (where the variable makes the performance trend up or down). There is some loss of precision due to the averaging in the simpler graphics, but the intention is rapid feedback for designers that can be explored in more detail if desired.

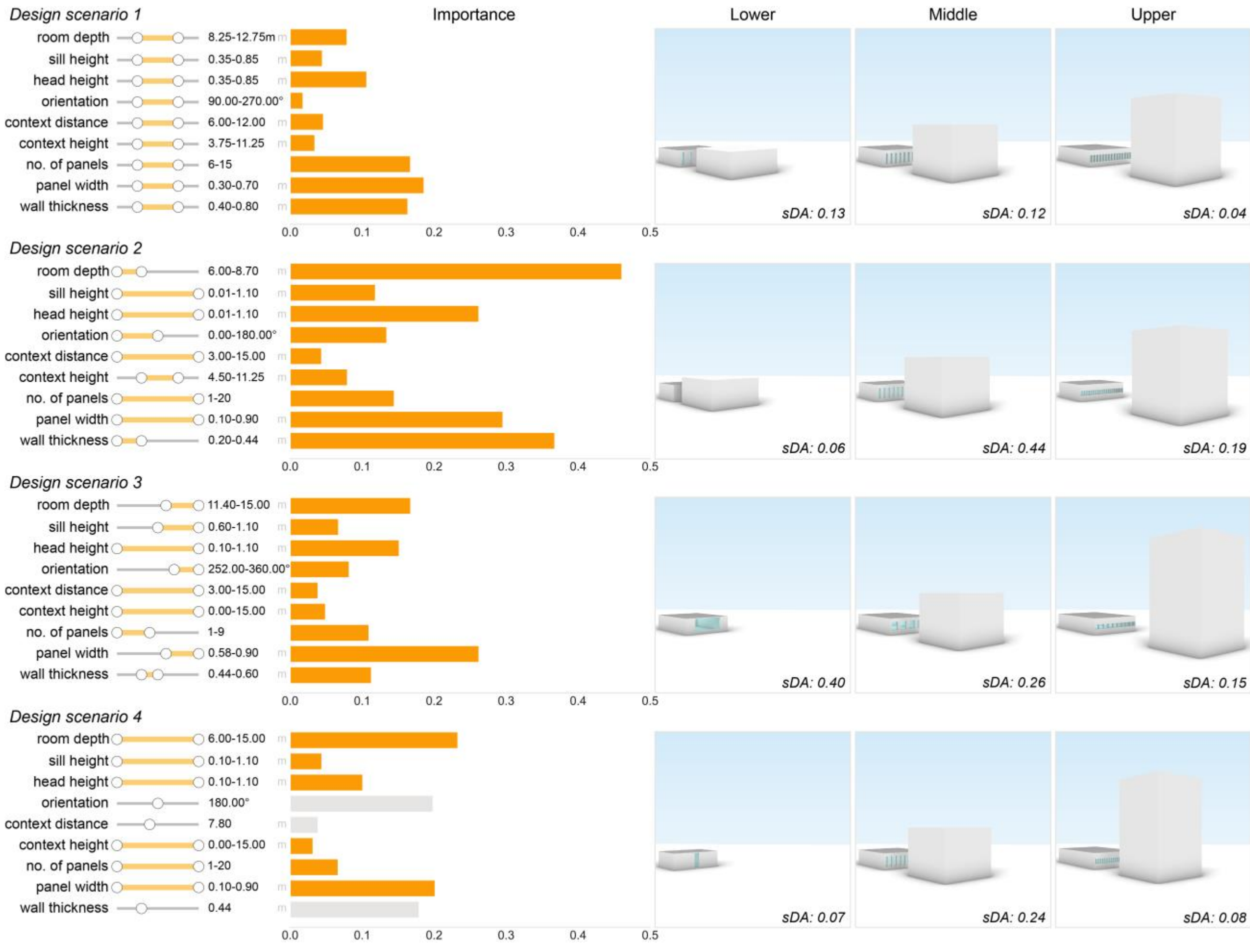


Figure 7: Filtering the Pittsburgh dataset

Design scenario 1 represents typical values, eliminating the extremes. The importances are similar to those of the full dataset. In design scenario 2, room depth is shallow, the context building is between 4.50 and 11.25 m tall, and the exterior wall is between 0.20 and 0.44 m thick. Because lower values of room depth and wall thickness significantly affect sDA, their importances increase. Additionally, due to the inclusion of both south and north orientations in this scenario, orientation became more important than context distance and context height. For design scenario 3, room depth is deeper, sill height is higher, the room faces between northwest and south, there are fewer panels, but higher panel widths, and the wall thickness is heavily restricted. With the number of panels limited, panel width is most important. Head height also became the third most important variable. Lastly, in design scenario 4, three values are already known and set to specific values. With these variables eliminated, room depth and panel width are most important. Although this dataset is fairly linear, these examples of dynamic subset sensitivity analysis begin to show how refining the variable domain affects variable importance and ultimately design decisions.

While the images above can alert the designer to overall importance and how it changes as variables are restricted, the dynamic subset sensitivity analysis visualizations above do not capture the sign of the coefficient or slope. In some cases, it may be useful to understand the relationships between the variables and the response on a more granular level. For a closer look, Figure 8 shows heatmaps where each row is a leaf model and the color of the cell represents the coefficient. The heatmap on the left shows all 144 leaf models that make up the linear model tree. The heatmap on the right shows the remaining available leaf models when the dataset is filtered based on design scenario 1. 77 leaf models intersect the user-defined intervals. In comparing the two graphs, it is possible to see how areas of significant variable influence change as the bounds are restricted. For example, many leaves with a high negative coefficient for room depth are excluded after restricting the bounds. Also, several leaf models with high positive coefficients for panel width are also excluded when extreme values are taken out of the domains.

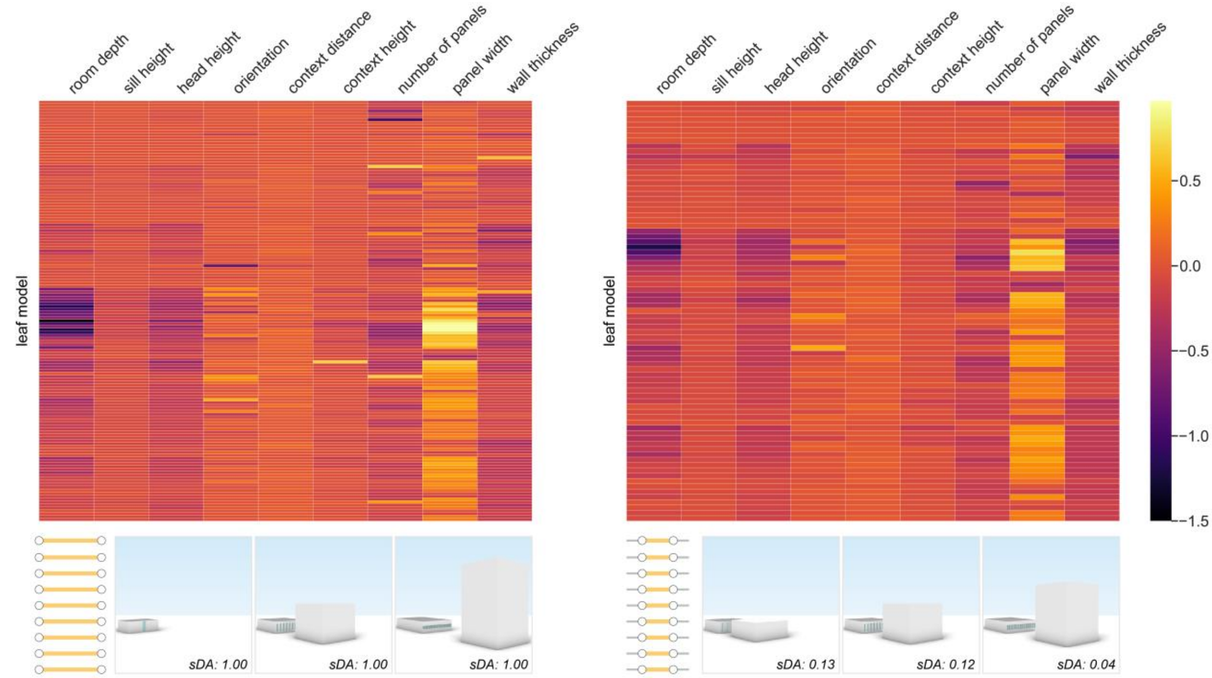


Figure 8: Filtering the Pittsburgh dataset based on design scenario 1

If this visualization were incorporated into a tool, the user could expand one variable to display the domain for each leaf model to help communicate relationships among variables, which can get lost in the simpler graphics. Figure 9 provides two examples, a and b, to demonstrate how this visualization tool could work. In example ***a***, panel width's domain spans from ~0.65-0.90 and it has a moderately strong positive linear relationship with sDA. Under the assumption that panel width is 0.65-0.90, room depth has a highly negative linear relationship with sDA. However, in example ***b***, the domain of panel width is expanded to ~0.45-0.90, and in this case, the negative linear relationship between room depth and sDA is weaker, but orientation has a strongly positive linear relationship to sDA. These examples show how a specific relationship with the response can be achieved by adjusting the domain of multiple variables.

5 CONCLUSION AND FUTURE WORK

In this work, we introduced a new way of thinking about sensitivity analysis dynamically in early design and presented methods to support it. Subset sensitivity analysis for common early design problems allows designers to discover which variables tend to matter for performance before modifying the design outside a restrictive parametric framework. While traditional sensitivity analysis requires building a custom parametric model, running variable importance analysis, adjusting the model as design criteria emerges, and re-running the analysis, dynamic subset sensitivity analysis provides designers variable sensitivity feedback in real-time as design criteria emerges. However, if the common design problem data is highly nonlinear, the linear model tree model may not be able to build regions with high enough accuracy. In future work, other building performance datasets will be tested to demonstrate robustness. Dynamic subset sensitivity analysis tracks with design to make the greatest impact without generating a custom parametric model for each project.

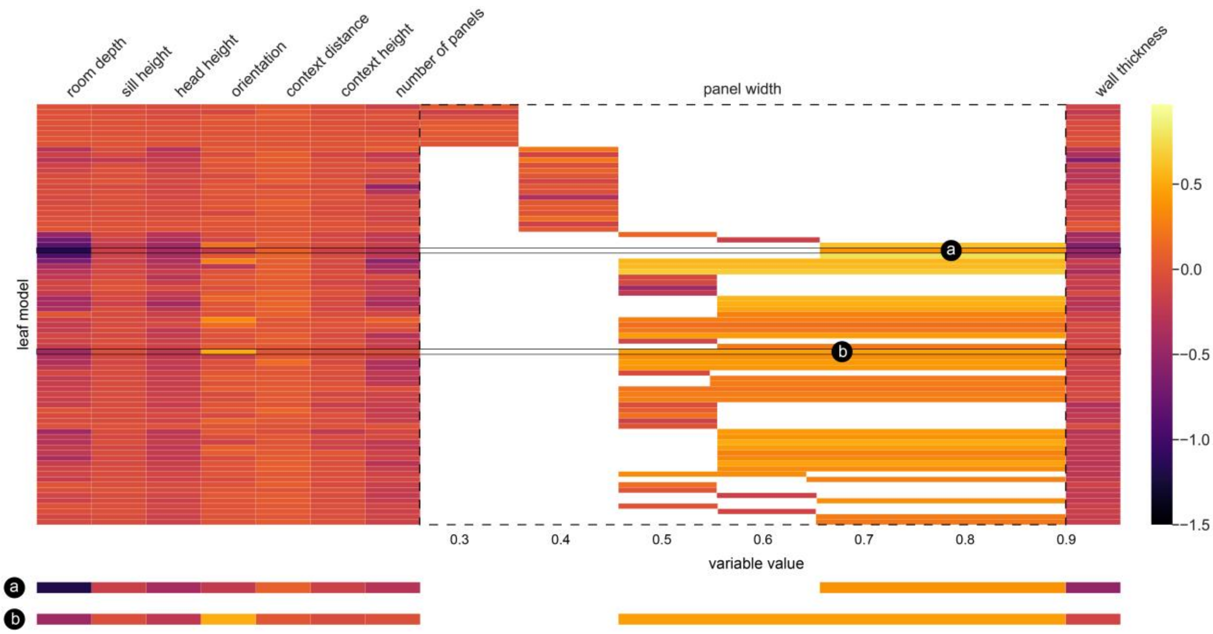


Figure 9: Expanding one variable to understand domain in leaves

ACKNOWLEDGMENTS

The authors would like to acknowledge the financial support of the Graduate Assistance in Areas of National Need (GAANN) Fellowship in completing this study.

REFERENCES

Arababadi, R., H. Naganathan, K. Parrish, and W.K. Chong. 2015. “Determining the Feasibility of Statistical Techniques to Identify the Most Important Input Parameters of Building Energy Models.” *Procedia Eng.* vol. 118 (January), pp. 1258–65.

Bambardekar, S., and U. Poerschke. 2009. “The Architect as Performer of Energy Simulation in the Early Design Stage.” In *Eleventh International IBPSA Conference*, Glasgow, Scotland.

Brembilla, E., C. J. Hopfe, and J. Mardaljevic. 2018. “Influence of Input Reflectance Values on Climate-Based Daylight Metrics Using Sensitivity Analysis.” *J. Build. Perform. Simul.* vol. 11 (3), pp. 333–49.

Brown, N. C, and C. T Mueller. 2019. “Design Variable Analysis and Generation for Performance-Based Parametric Modeling in Architecture.” *Int. J. Archit. Comput.* vol. 17 (1), pp. 36–52.

Brown, N., and C. Mueller. 2017. “Designing with Data: Moving beyond the Design Space Catalog.” In *37th Annual Conference of the ACADIA, ACADIA 2017*.

Danhaive, R., and C. T. Mueller. 2021. “Design Subspace Learning: Structural Design Space Exploration Using Performance-Conditioned Generative Modeling.” *Autom. Constr.* vol. 127, art. 103664.

Ferrara, M., E. Fabrizio, J. Virgone, and M. Filippi. 2014. “A Simulation-Based Optimization Method for

Cost-Optimal Analysis of Nearly Zero Energy Buildings." *Energy Build.* vol. 84, pp. 442–457.

Gao, Y., M. Mae, and K. Taniguchi. 2020. "A New Design Exploring Framework Based on Sensitivity Analysis and Gaussian Process Regression in the Early Design Stage." *J. Asian Archit. Build. Eng.*, pp. 1–14.

Nguyen, A. T., and S. Reiter. 2015. "A Performance Comparison of Sensitivity Analysis Methods for Building Energy Models." *Build. Simul.* vol. 8 (6), pp. 651–64.

Østergård, T., R. L. Jensen, and S. E. Maagaard. 2017a. "Early Building Design: Informed Decision-Making by Exploring Multidimensional Design Space Using Sensitivity Analysis." *Energy Build.* vol. 142 (May), pp. 8–22.

Østergård, T., R. L. Jensen, and S.E. Maagaard. 2017b. "Interactive Building Design Space Exploration Using Regionalized Sensitivity Analysis." In *Building Simulation Conference Proceedings*, vol. 4, pp. 1997–2006. International Building Performance Simulation Association.

Pang, Z., Z. O'Neill, Y. Li, and F. Niu. 2020. "The Role of Sensitivity Analysis in the Building Performance Analysis: A Critical Review." *Energy Build*. Elsevier.

Pannier, M. L., P. Schalbart, and B. Peuportier. 2018. "Comprehensive Assessment of Sensitivity Analysis Methods for the Identification of Influential Factors in Building Life Cycle Assessment." *J. Clean. Prod.* vol. 199 (October), pp. 466–80.

Samuelson, H., S. Claussnitzer, A. Goyal, Y. Chen, and A. Romo-Castillo. 2016. "Parametric Energy Simulation in Early Design: High-Rise Residential Buildings in Urban Contexts." *Build. Environ.* vol. 101 (May), pp. 19–31.

Wit, S. D., and G. Augenbroe. 2002. "Analysis of Uncertainty in Building Design Evaluations and Its Implications." *Energy Build.* vol. 34 (9), pp. 951–958.

Wortmann, T. 2017. "Surveying Design Spaces with Performance Maps: A Multivariate Visualization Method for Parametric Design and Architectural Design Optimization." *Int. J. Archit. Comput.* vol. 15 (1), pp. 38–53.

Yang, T., Y. Pan, J. Mao, Y. Wang, and Z. Huang. 2016. "An Automated Optimization Method for Calibrating Building Energy Simulation Models with Measured Data: Orientation and a Case Study." *Appl. Energy* vol. 179 (October), pp. 1220–1231.

AUTHOR BIOGRAPHIES

LAURA HINKLE is a PhD student in the Architectural Engineering department at Penn State University. She holds a Bachelor and Master of Architectural Engineering from Penn State University. Her research interests include building envelopes and applications of data science to building design.

GREGORY PAVLAK is an Assistant Professor of Architectural Engineering at Penn State University. He holds a PhD in Architectural Engineering from University of Colorado Boulder. His research interests include smart building systems, building-to-grid integration, and distributed and renewable energy.

LELAND CURTIS is an independent researcher with 10 years of experience in AEC, most recently at CRB. He holds a Bachelor and Master of Architectural Engineering from Penn State University. His research interests include integrated building design and practical applications of machine learning.

NATHAN BROWN is an Assistant Professor of Architectural Engineering at Penn State University. He holds a PhD in Building Technology from the Massachusetts Institute of Technology. His research interests include conceptual building design, structural and multi-objective optimization, and computational design.