Minimal sampling for stochastic transmission economic assessment

Erich Trieschman [1]

[1]Department of Statistics

Introduction

Objective: To minimize the number of samples needed for generating uncertainty bounds for transmission economic assessments

- Traditional transmission planning models use deterministic load and resource availability forecasts, neglecting uncertainty in assessing economic benefits.
- Transmission Economic Assessment Methodology (TEAM) offers a solution to address uncertainties by stress-testing transmission alternatives under future conditions in a stochastic production cost simulation. However, the simulations are computationally expensive.
- We develop methods to select minimal samples of load, wind, and solar forecasts for use in the TEAM framework; our methods leverage experiment design and active learning techniques.

Background: Production cost simulation and stochastic profiles

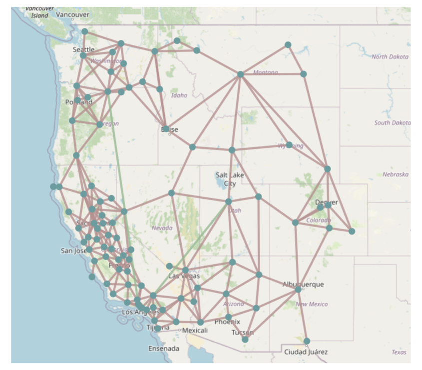


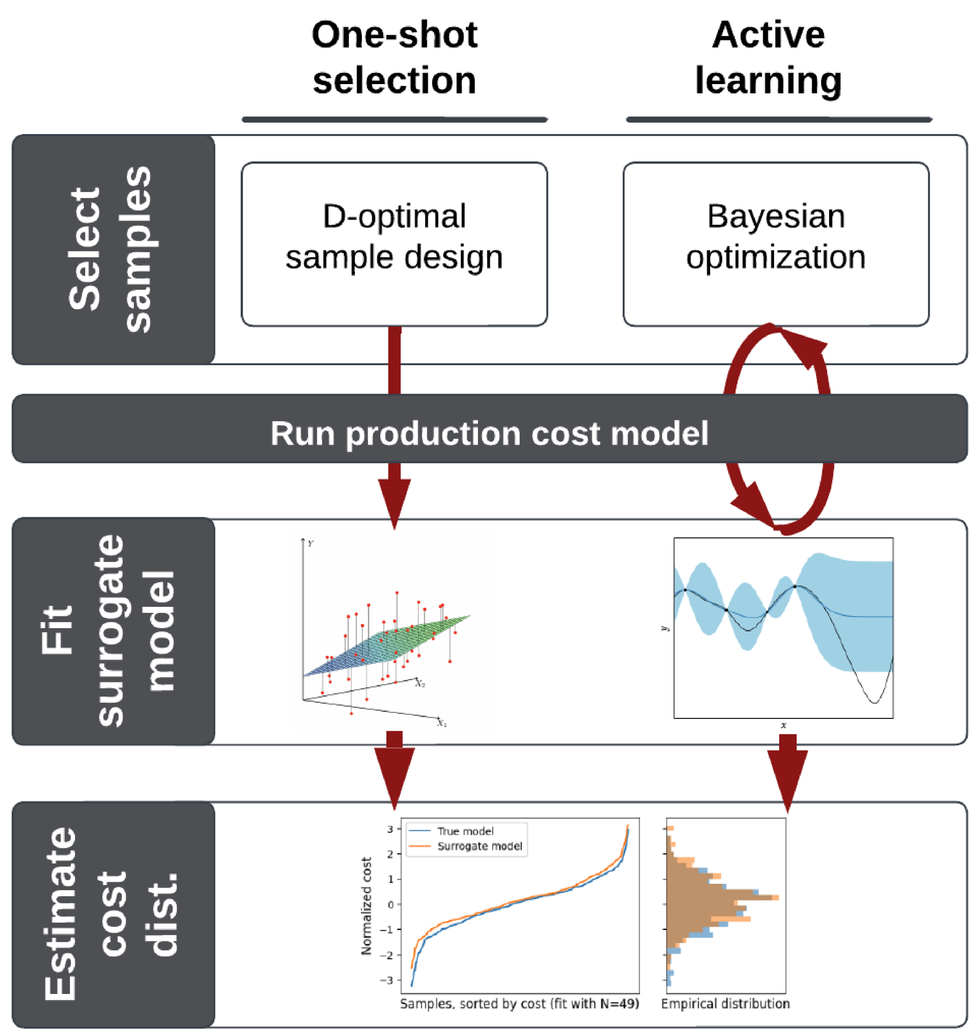
Figure 1. pypsa-usa 96-node WECC network

- **Scenario:** We evaluate our methods under a single transmission expansion scenario for an HVDC VSC cable connecting Humboldt to PG&E Bay in a synthetic 96-node network of the Western Electricity Coordinating Council (WECC).

$$\text{Benefits} = [\tilde{\lambda}_{n,t} - \lambda_{n,t}] \cdot P_{\text{load},n,t}$$

- **Cost model:** To estimate economic costs we use a nodal production cost simulation (PCS) model built on the Breakthrough Energy synthetic WECC network model. The WECC model is built within the Python for Power System Analysis (pypsa) toolkit.
- **Stochastic profiles:** We generate stochastic load, wind, and solar profiles using the mean-reversion stochastic process method developed by the CAISO. Profiles are encoded using PCA to aid in sample selection; Only the first 10 principal components are used.

Methods



Methods: Baselines

- **Full sample:** Empirical distribution of the full sample of production cost model runs ($N = 500$)
- **Random sample:** Empirical distribution generated by a surrogate model fit with randomly selected samples
- **Bootstrapping:** Bootstrapping used to generate confidence intervals around performance metrics

Methods: One-shot sample selection with D-optimal experiment design

- **Surrogate model:** Linear relationship between forecasts and costs
- **Objective:** Select samples that maximize forecast covariance matrix (equivalent to minimizing standard errors of $\hat{\beta}$). Use $m \in \{0,1\}^n$ instead of the canonical formulation, $m \in \mathbb{Z}^n$, because the cost model is deterministic.
- **Convex relaxations:** Scalarize with D-optimal design formulation as described in Boyd et. al. and use L1-norm heuristics to maximize solution sparsity.

$$\begin{aligned} \min_{\lambda}. \quad & -\log\det\left(\sum_{j=1}^{p} \lambda_j v_j v_j^T\right) \\ s.t. \quad & 0 \preceq \lambda \preceq 1, \quad \|\lambda\|_1 \leq \alpha M \end{aligned}$$

For p distinct forecasts, distinct scenario v_j, m_j selecting scenario v_j, target scenarios M, and hyperparameter α. This approach requires discretizing the dimensions of our input space x_i into scenarios v_j.

Methods: Active learning sample selection with Bayesian optimization

- **Surrogate model:** Gaussian Process (GP) model

$$f^{(n+1)} \mid \mathbf{x}^{(1:n+1)} \sim \mathcal{N}\left(\mathbf{k}(x^{(n+1)})^T \mathbf{K}^{-1} \mathbf{f}^{(1:n)}, k(x^{(n+1)}, x^{(n+1)}) - \mathbf{k}(x^{(n+1)})^T \mathbf{K}^{-1} \mathbf{k}(x^{(n+1)})\right)$$

$$k(x^{(i)}, x^{(j)}) := \exp\left(-\frac{1}{2l^2}\|x^{(i)} - x^{(j)}\|_2^2\right) \quad \mathbf{k}(x^{(n+1)}) := \left[\ldots, k(x^{(n+1)}, x^{(i)}), \ldots\right] \quad \mathbf{K}_{ij} := k(x^{(i)}, x^{(j)})$$

- **Objective:** Select the next sample point to maximize the entropy of the surrogate model fit to all previous samples. Maximum entropy search (MES) aims to reduce overall surrogate model uncertainty

$$\operatorname{argmax}_x \ \frac{1}{2}\log(2\pi k(x^{(n+1)}, x^{(n+1)}) - \mathbf{k}(x^{(n+1)})^T \mathbf{K}^{-1} \mathbf{k}(x^{(n+1)})) + \frac{1}{2} = \ \operatorname{argmin}_x \mathbf{k}(x)^T \mathbf{K}^{-1} \mathbf{k}(x)$$

- **Convex relaxations:** Non-convex objective; Sequential convex programming (SCP) used to iteratively solve for local maxima for each sample, $x^{(n+1)}$

$$x^{(n+1)^{(k+1)}} = \operatorname{argmin}_x \hat{f}(x) \ \text{ s.t } \ x \in \mathcal{T}^{(k)}$$

- **Function approximations:** Both second-order Taylor approximation and particle method approximation methods used to approximate the objective
  - **Second-order Taylor approximation:**

$$\hat{f}(x) = f(x^{(k)}) + \nabla f(x^{(k)})^T (x - x^{(k)}) + (x - x^{(k)})^T P (x - x^{(k)}) \text{ for } P = \left(\nabla^2 f(x^{(k)})\right)_+$$

  - **Particle method approximation:**

$$\begin{aligned} &\hat{f}(x) = (x - x^{(k)})^T P (x - x^{(k)}) + q^T (x - x^{(k)}) + r \text{ for} \\ &P, q, r = \operatorname{argmin}_{P \succeq 0, q, r} \sum_{i=1} \left((z_i - x^{(k)})^T P (x_i - x^{(k)}) + q^T (z_i - x^{(k)}) + r - f(z_i)\right)^2, \text{ for } z_i \sim \text{Unif}\left[\mathcal{T}^{(k)}\right] \end{aligned}$$

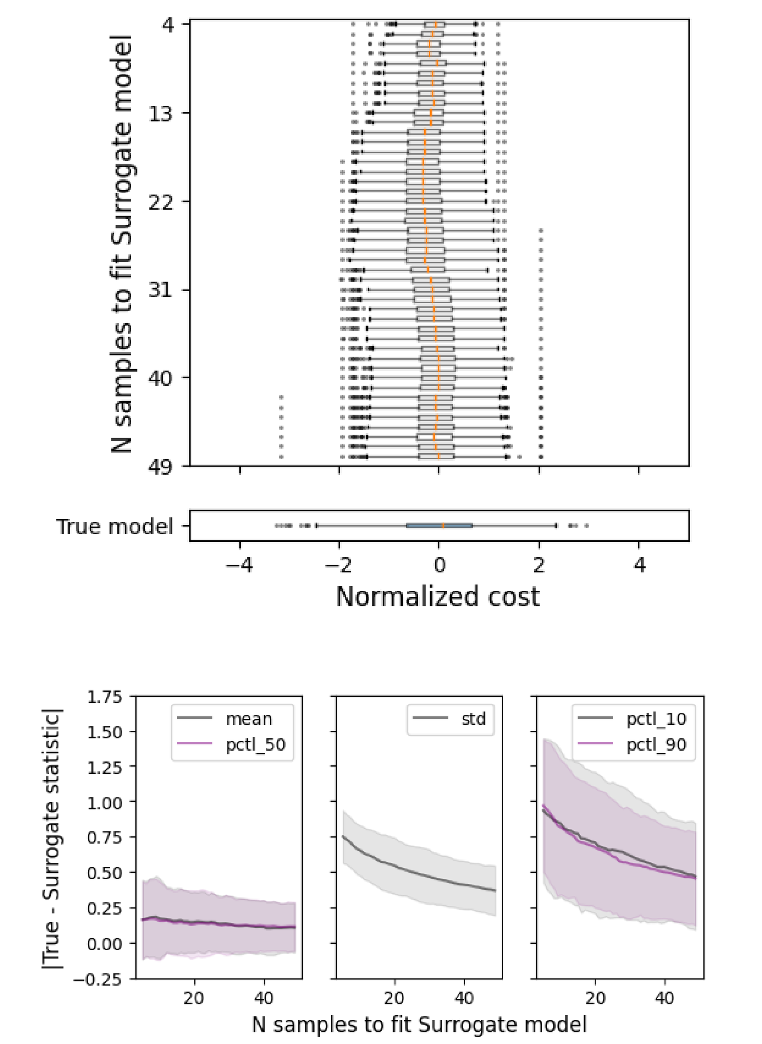
Results

Table 1. Absolute difference between true distribution and surrogate distribution statistic (samples=50, bootstraps=100)

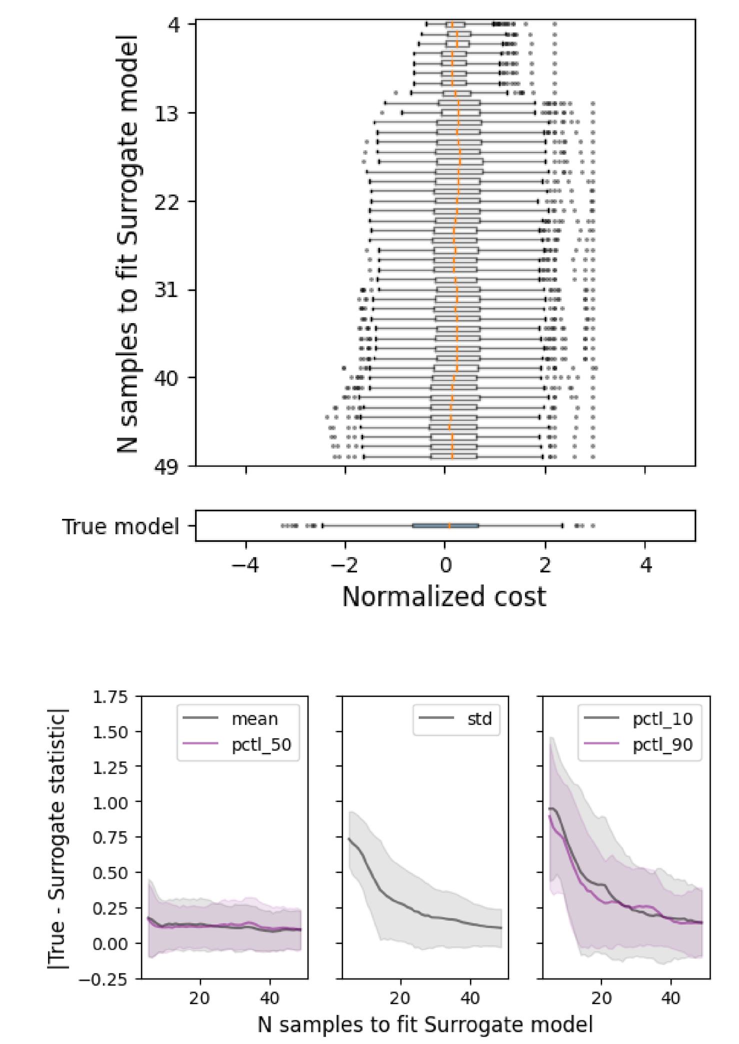
	One-shot		Active learning		
	Baseline	D-optimal	Baseline	Taylor	Particle
Min	0.73±0.37	1.49	0.64 ± 0.53	0.46 ± 0.41	0.41 ± 0.39
5th pctl	0.62±0.23	0.85	0.68 ± 0.23	0.24 ± 0.16	0.23 ± 0.16
10th pctl	0.56±0.20	0.53	0.47 ± 0.19	0.14 ± 0.12	0.17 ± 0.14
25th pctl	0.42±0.17	0.26	0.28 ± 0.14	0.11 ± 0.10	0.14 ± 0.11
50th pctl	0.16±0.12	0.11	0.11 ± 0.09	0.09 ± 0.07	0.14 ± 0.10
75th pctl	0.30±0.19	0.34	0.26 ± 0.13	0.11 ± 0.09	0.12 ± 0.09
90th pctl	0.77±0.27	0.66	0.45 ± 0.17	0.14 ± 0.13	0.16 ± 0.12
95th pctl	0.89±0.30	0.75	0.47 ± 0.20	0.18 ± 0.17	0.18 ± 0.15
Max	2.41±0.69	1.35	0.76 ± 0.46	0.40 ± 0.45	0.53 ± 0.37
Mean	0.14±0.10	0.04	0.11 ± 0.09	0.09 ± 0.07	0.11 ± 0.09
Std	0.50±0.13	0.50	0.37 ± 0.09	0.10 ± 0.07	0.10 ± 0.07

Results

Active learning baseline



Active learning Taylor approx.



Active learning samples selected with Taylor approximations in SCP. See final report appendix for the performance of the particle method and one-shot sample selection

Discussion

- Optimal sample selection methods can be used generate cost distributions at 5-10% of the computational costs required to run the full production cost model for all samples
- Optimal sample selection methods outperform random sample selection methods, especially for estimating cost distribution percentiles and standard deviations.
- Active learning sample selection runtime is negligible compared to the production cost model runs (30min); one-shot sample selection runtimes are on the order of minutes.
- **Future work:** Apply sampling method to general generation and transmission capacity expansion modeling (instead of a single-scenario assessment); improved hyperparameter tuning

Contributions and acknowledgements

This work is conducted by Erich Trieschman as part of a larger project in collaboration with Kamran Tehranchi. Kamran is responsible for running the production cost model and generating stochastic profile data. Code and results are available on Github.

References

[1] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[2] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Nov. 2005. ISBN: 9780262256834. DOI: 10.7551/mitpress/3206.001.0001. URL: https://doi.org/10.7551/mitpress/3206.001.0001.

[3] Mohamed Labib et al. Awad. "Using market simulations for economic assessment of transmission upgrades". en. In: (2010). Ed. by Xiao-Ping Zhang, pp. 241–270. DOI: 10.1002/9780470608555.ch7.