# Generative Perturbation Analysis for Probabilistic Black-Box Anomaly Attribution

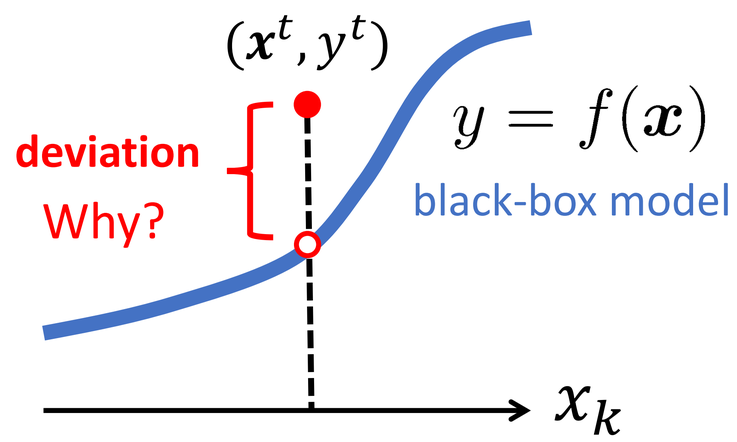
Tsuyoshi Idé (井手 剛), Naoki Abe (IBM Research, T. J. Watson Research Center)

**What's anomaly attribution ?**

**Given**:

- black-box regression function $y = f(\boldsymbol{x})$
- (set of) test sample(s) $(\boldsymbol{x}^t, y^t)$

**Explain**:
the deviation $f(\boldsymbol{x}^t) - y^t$ by computing the attribution score (responsibility score) for each of the input variables $\boldsymbol{x}$.



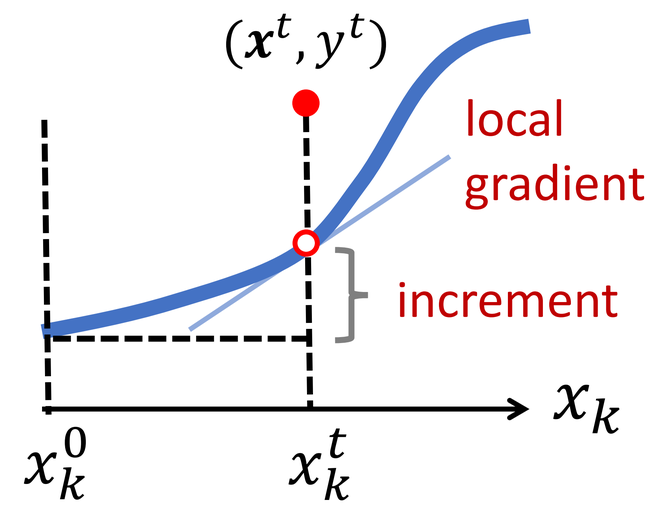

**What's wrong with existing methods?**

**Limitations of LIME, Shapley value (SV), and integrated gradient (IG) in anomaly attribution**:

- They explain $f(\boldsymbol{x}^t)$, NOT the deviation.
- Unable to compute score's uncertainty

**LIME, SV, and IG are deviation-agnostic!**

They compute either local gradient (LIME) or increment from a certain **reference point** $\boldsymbol{x}^0$ (Shapley values, IG), independently of the observed deviation.




**GPA allows providing the probability distribution of attribution scores in a deviation-sensitive fashion.**

**What is the new idea?**

Key question: Given $(\boldsymbol{x}^t, y^t)$ being anomalous, ***how much "work" would we need to bring it to the normalcy?***

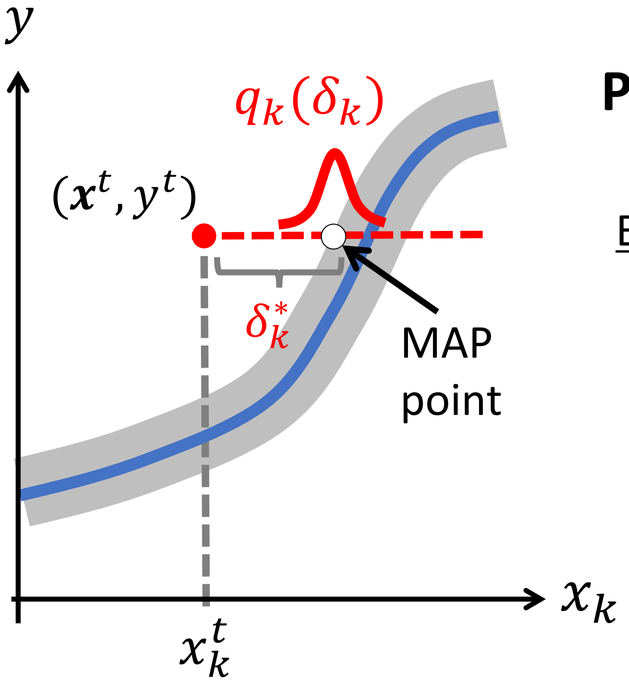
The amount of work assigned to each variable → attribution score.

We use the amount of shift as the "work".

Generative model for $y$ with the shift $\boldsymbol{\delta}$ as a "model parameter."

- observation model: $p(y^t \mid \boldsymbol{x}^t, \boldsymbol{\delta}, \lambda) = \mathcal{N}(y^t \mid f(\boldsymbol{x}^t + \boldsymbol{\delta}), \lambda^{-1})$
- priors: $p(\boldsymbol{\delta}) = \mathcal{N}(\boldsymbol{\delta} \mid \mathbf{0}, \eta \mathrm{I}),\ p(\lambda) = \mathrm{Gam}(\lambda \mid a_0, b_0)$

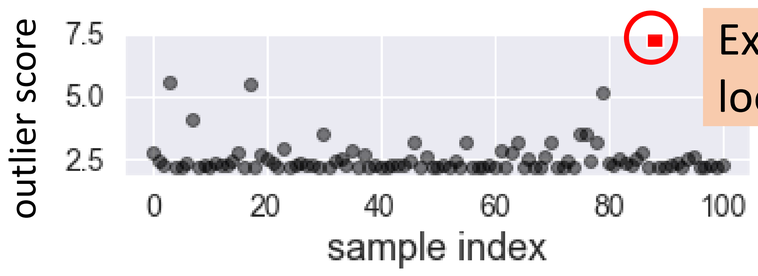
Deviation agnostic




**Posterior** = The distribution of attribution score.

Boston Housing example (bargain house hunting!)

- Looked into the sample of the highest outlier score.
- Computed attribution scores suggest unusually more rooms (RM) and fewer poor neighbors (LSTAT).




Explain why this looks anomalous.