# Heatmap anomaly detection

| | |
|---|---|
| **Project:** | **Data Science Capstone Criteo & Columbia University**<br>**ENGI E4800 Spring 2024** |
| **Criteo Supervisors:** | **Jan Benzing (j.benzing@criteo.com)**<br>**Matt Merenich (m.merenich@criteo.com)** |

## PROBLEM:

Criteo (NASDAQ: CRTO) is a worldwide leader in data-driven advertising. Every day, we serve around 4 billion ads, and we have around 20+ millions of clicks. We have various algorithms that help us serve the most appropriate ads that provide a good user experience (UX) to the final user. However, we noticed that some clicks have odd patterns, so-called misclicks. Practice showed that those misclicks are due to slow phones, heavy banners, and misleading websites that spoil UX. Heatmaps can usually spot this. In the case of slow banners, heat maps look completely broken.

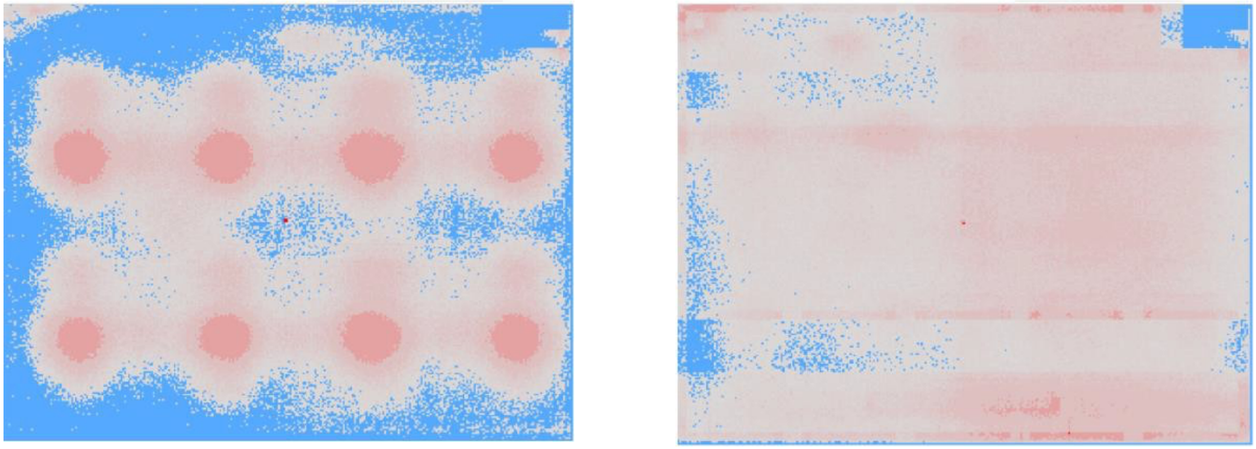

*Figure 1: Good (left) and "broken" (right) heatmap*

Criteo is working on spotting such misclicks and slow banner issues. For that, we've developed a p-misclick metric (the probability of a click being a misclick) and a heatmap generation tool (the user can observe the click distribution on a given scope). We want to automatize this approach. Your objective is to build a clusterization/anomaly detection algorithm (i.e., fully unsupervised learning) to predict whether the given heatmap is broken. Thus, we could spot the "hot points" with issues and address them.

**You could use various tools:**

- for data manipulation (Python, Excel)
- for modelling (Python)
- for visualization (Python/Excel/Tableau)

**The project could follow the list of steps:**

## I. For a specific scope with a fixed ad size:

1. **Data exploration and problem definition**: Study examples of good and broken heatmaps to clearly understand the meaning of an anomaly, in this context. Study outside literature to get inspired with models used in similar tasks.

    a. Datasets given are all of ad formats of one (1) size: pixel 300x250 and contains three (3) products

2.

    a. **Feature Extraction:** Design *simpler* descriptors of the heatmap (e.g. measure of the spread of clicks, distance between each click etc) to act as a baseline for clusterization. Then move on to learning models (e.g. PCA), with the objective of outperforming the baseline.

    b. **OR as an another approach**, you can directly go to next steps using raw info (which is a matrix of the nb of clicks per pixel)

3. **Pixels Aggregation** & **Estimation of distribution:** aggregate heatmap data to buckets of pixels instead of raw pixels & smooth click distribution (e. g. KDE).

4. **Models definition**: Design models to work on your transformed dataset, with multiple approach possible (clustering / anomaly detection algorithm, etc).

5. **Model performance:** Compare performance between the initial baseline and more elaborate models. Work on the explainability of your models (why a heatmap is considered as broken by your model, etc...).

6. **Bonus 1:** Once identified suspicious heatmaps scope, use this information to classify new heatmap (e.g. KNN).

7. **Bonus 2**: Once identified suspicious heatmaps scope, use other data features like FPS, Closing rate, CTR, Landed clicks etc to see common patterns.

## II. See if can be generalized for all ad sizes (separate datasets)

## PROJECT TOOLS

We provide:

- a dataset with 1 row per heatmap with metrics like: displays, clicks, landed clicks, avg last second framerate, sov ttc, etc.
- a dataset with pixel coordinates for each click by domain and grid_id (heatmap creation)