deepchecks.

Detecting Drift in Structured and Unstructured Data

About Me

- Nadav Barak
- ~5 Years in Research
  - Mathematics
  - Data Science
- ML Researcher @ Deepchecks

About deepchecks.

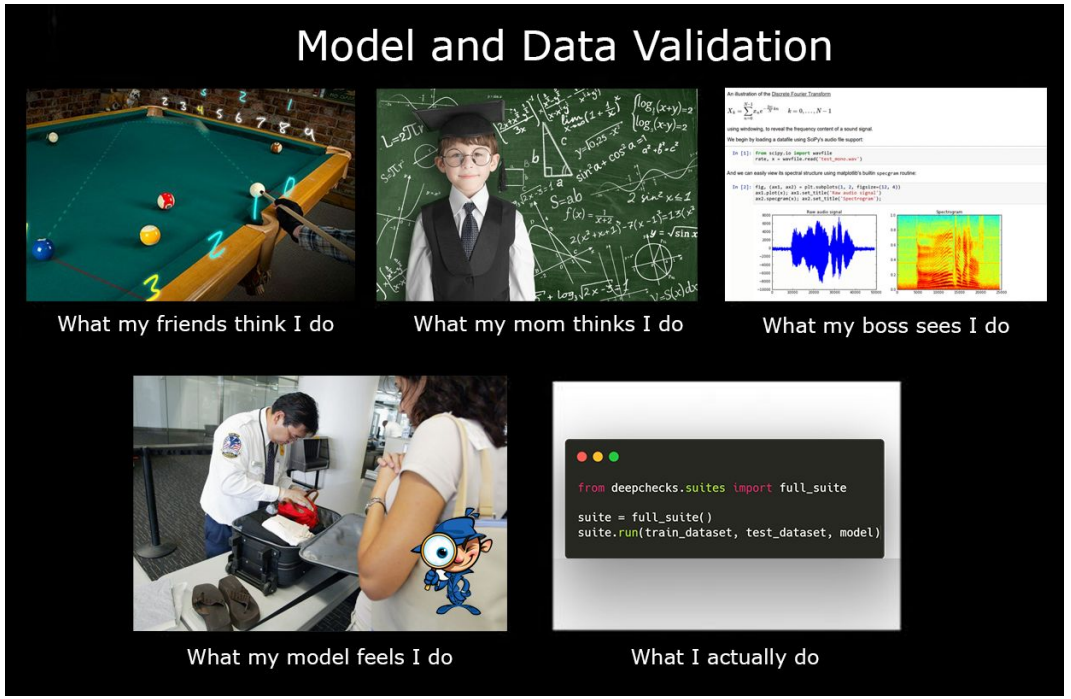
- Founded 3 years ago
- Backed by Grove Ventures & Hetz Ventures
- Continuous Validation for ML Systems

OPEN SOURCE PACKAGE
Test Suites for Offline Validation

DEEPCHECKS PRO
Production Monitoring

Agenda

- What should I test?
- What is drift, why is it important?
- Live code example - Drift detection in structured data
- Drift in unstructured data
- Drift mitigation

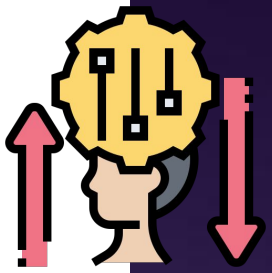


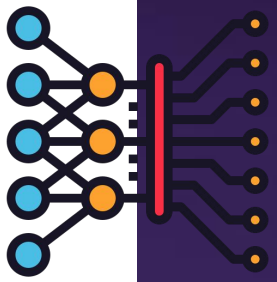
Testing software

Changes are clear

Can track coverage

Silent failures are **rare**





Testing ML

Data, Model and pipeline can all change

No clear “coverage”

Silent failures are **common**

Why testing – What Can Go Wrong?

Data Integrity

Dirty Data

Duplicates

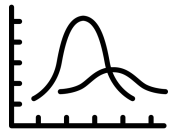
Labeling inconsistency

Fairness & Bias

Fairness Constraints

Indirect Protected attributes

Data Distribution



Drift

Anomalies

Methodological Flaws

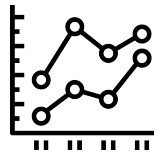
Leakage

Sampling Manner

Overfit

Shortcut learning

Model Performance



Degradation

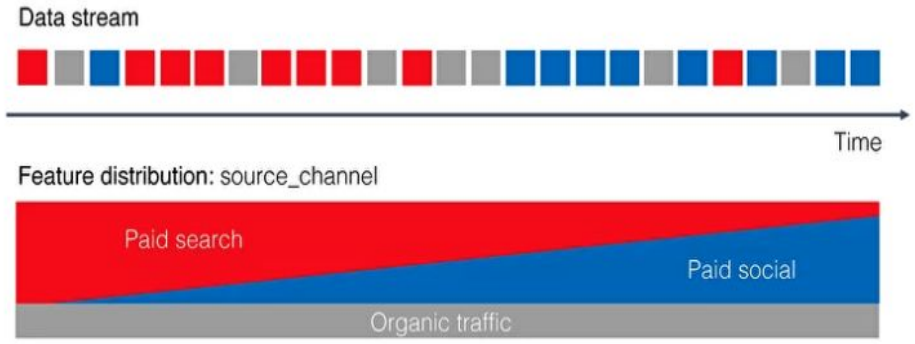
Segment Performance

Calibration

What is Drift?

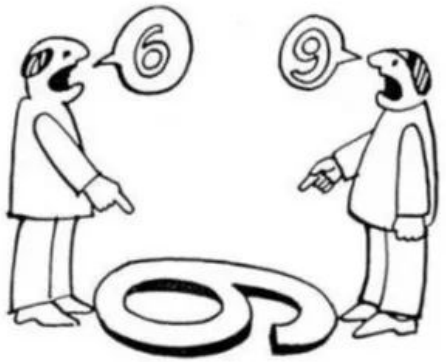
- Drift is a situation where the data, labels or their joint distribution changes over time.
- Drift can be categorized into two broad categories:

Data Drift



$$P_{t1}(X) \neq P_{t2}(X)$$

Concept Drift



$$P_{t1}(Y|X) \neq P_{t2}(Y|X)$$

Why is Drift Important?

- **Expectability**. If your model was trained and tested on different data from the production data, how can you guarantee its behavior?

- **Performance**. Drifted production data commonly mean deterioration in performance.
  - Data Drift: when weak model segments become more prominent.
  - Concept Drift: The model's previously accurate predictions become false.

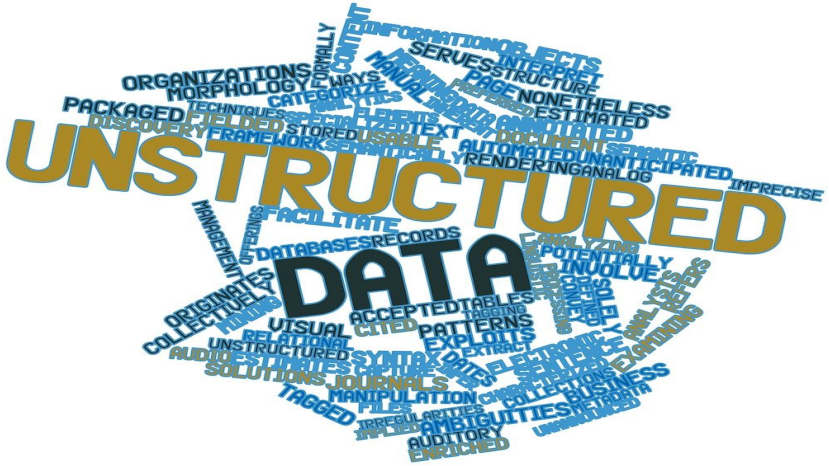
Detecting Drift Code Example
Structured Data

https://github.com/Nadav-Barak/MAFAT_Challenge_Talk

Drift in Unstructured Data

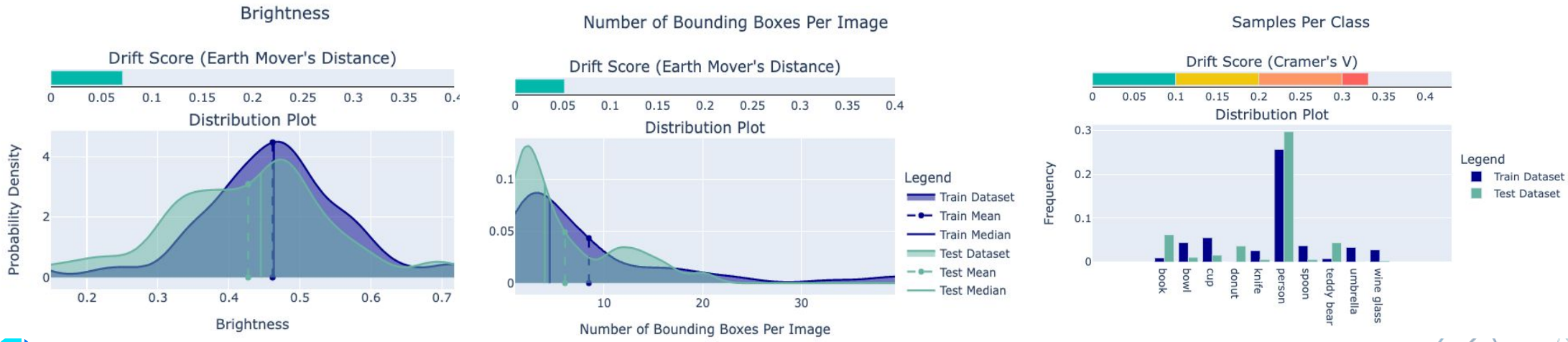
- **Main challenges:** Values at a specific dimension (ex. pixel) are not meaningful enough to ‘explain’ drift. Evan a combination of dimensions in most cases will not represent the semantic essence of the drift.

- **Secondary challenge:**. Usually the size of a single sample is large. This causes a problem for statistical computations that store the data in memory.



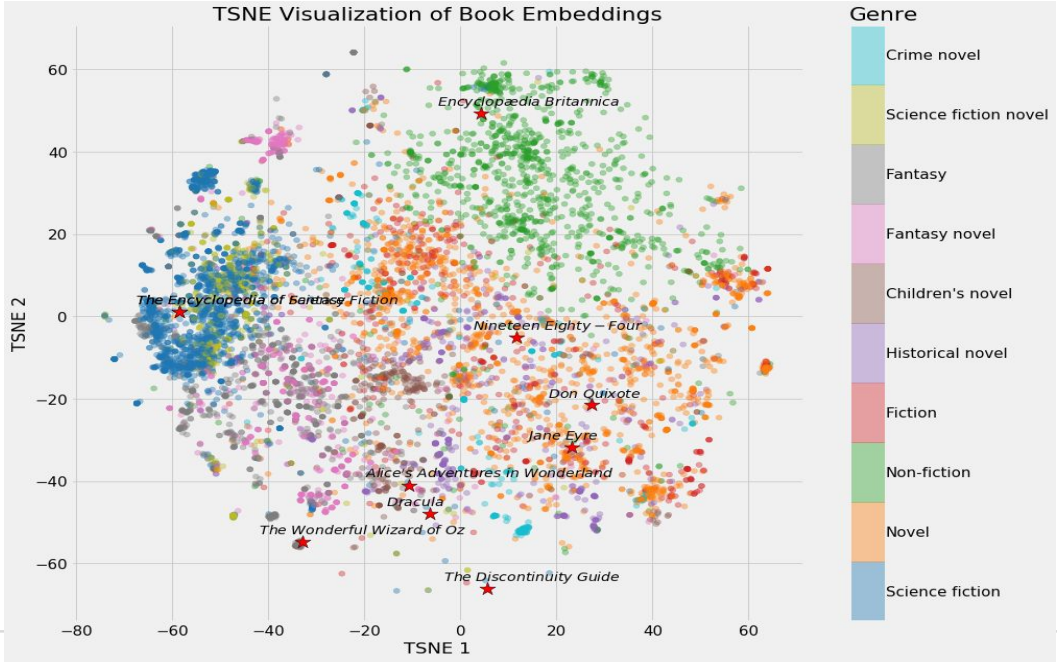
Properties

- Properties are one-dimension values that are extracted from either the samples, labels or predictions. For example, in object detection, an image property can be brightness, and a label property can be avg bounding box area.
- Quality properties should be explainable, domain relevant and have the ability to ‘explain’ different types of drift.



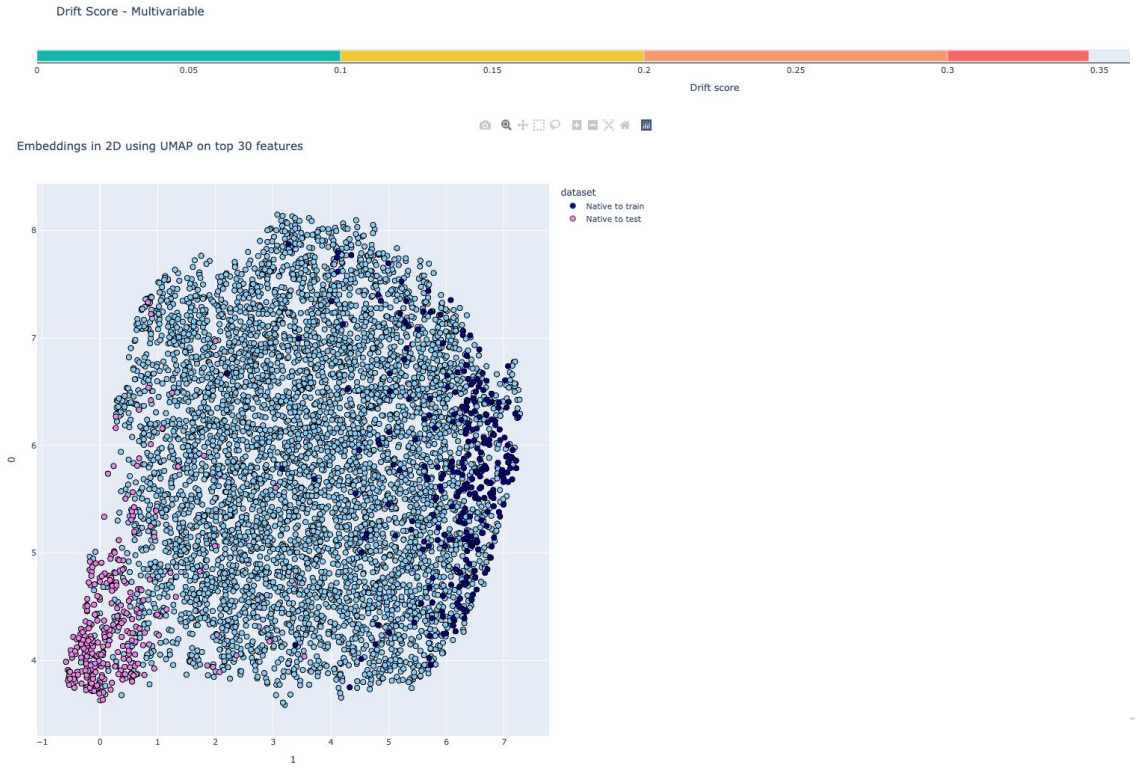
Embeddings

- Embeddings are a low-dimensional continuous **vector representation** of a sample. Quality embedding capture the essence of a sample such that semantically similar samples will have similar embedding vectors.
- In the context of neural networks, usually one of the final model layers are used as the embedding.



Embeddings

- In contrast to properties, the single dimensions in the embedding space usually don't have a specific meaning and lack the power to explain drift on their own.
- Domain classifier to the rescue!



Drift Detection MAFAT Challenge

- Alternatives for applying drift methods in object detection tasks:
  - Use embeddings / properties calculation functions to create a dataset containing the relevant information for each sample, then run the **Deepchecks' tabular** drift detection checks.
  - Run **Deepchecks' vision** designated checks. The main benefits of this option is the access to data integrity and model evaluation checks and the use of build-in properties.

Drift Mitigation

- **Reduce the amount of drift.** This can be achieved by:
  - Removing features / samples from the train set.
  - Adding synthetic examples to train set (ex. by using data augmentations, SMOTE).
- **Informed modelling.** Select a model and a training procedure that takes into account the drift in the setting. One example is to up-weight train samples in regions that are more prominent in the test set.

Both methods require access to the drifted production data and finding both the cause for the drift and relevant train samples / regions for mitigation.

deepchecks.

Thank you for listening, and may the tests be with you

https://github.com/deepchecks/deepchecks

👇 Feedback, ideas & feature requests are greatly appreciated! 🙏
and if you like what we're doing – give us a ⭐ on GitHub