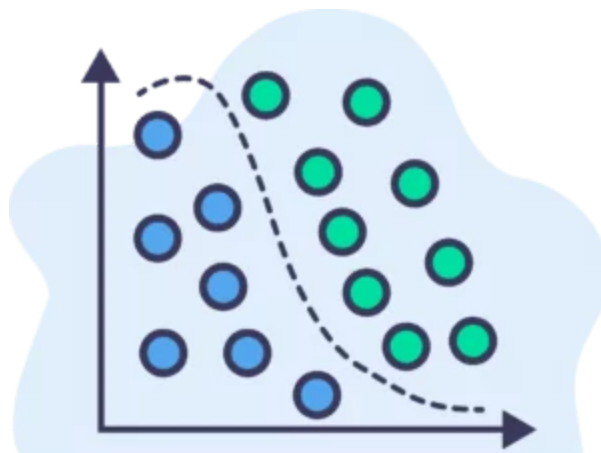


As machine learning models become more and more popular solutions for automation and prediction tasks, many tech companies and data scientists have adopted the following working paradigm: the data scientist is tasked with a specific problem to solve, they receive a snapshot of the available relevant data, and they work on training a model to solve it. Once the model is tested, it moves to production. Eventually, the model's performance begins to degrade, and more often than not, this is due to concept drift.

In the literature, concept drift is a situation in which the statistical properties of a target variable (what the model is trying to predict) changes over time in unforeseen ways.



Visually, we can say that a concept is a way to separate between the blue and green dots in the plot above. The black line represents a concept that separates the blue and green dots.

## Types of Drifts in Machine Learning

For the following definitions let's denote the following parameters:

$X$ - Model's input population.

$\hat{y}$  – Model's prediction.

$Y$ - True label population.

**Concept drift** – a change in the distribution of  $p(Y|X)$ , meaning that there was a change in the relationship between the input of the model and the true label.

**Prediction drift** – a change in the distribution of the predicted label –  $p(\hat{y}|X)$ , meaning that there was a change in the relationship between the input of the model and the model's prediction.

**Label drift** – a change in the probability of a label  $p(Y)$ .

**Feature drift** – a change in the probability of  $p(X)$ , meaning there was a change in the distribution of the model's input.

• • •

## Concept Drift can Appear in Different Ways

**Sudden drift** – A new concept occurs in a short period of time (e.g. at the beginning of COVID-19 in March 2020, stock prices suddenly changed).


**Gradual drift** – A new concept gradually replaces an old one over an extended period of time (e.g. you witness fewer new oil companies and more new tech companies).

**Incremental drift** – an old concept incrementally changes to a new concept over some period of time (e.g. a stock price gradually and steadily increases)

**Recurring concepts** – an old concept may reoccur after some time (e.g. changes in food delivery volumes on the weekends vs weekdays).

Note: in gradual drift, the two concepts go back and forth until the new concept finally stabilizes; in an incremental drift, the concept gradually changes to the new one.

These different concept drifts are illustrated in the figure below.

 Concept drift in machine learning 101

• • •

## Two Types of Concept Drift

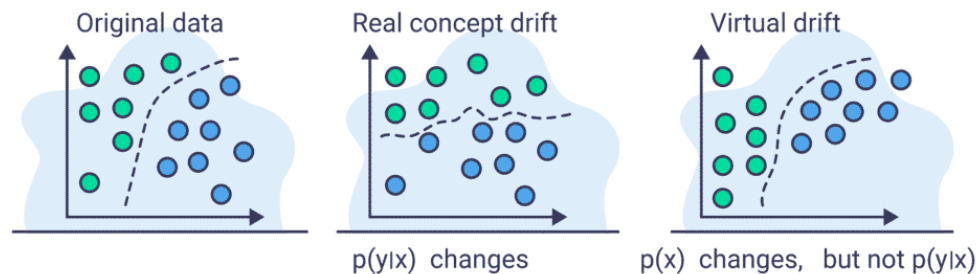
In order to better understand the effects of concept drift, we need to distinguish between two types of concept drift:

- **Virtual drift** – when  $p(X)$  changes but  $p(Y|X)$  does not change. Meaning that there was a change in the features' underlying

distribution, but the model's performance hasn't changed.

- Real drift – there was a change in  $p(Y|X)$ , meaning the performance of the model changed.

Virtual drift vs real drift is illustrated in the following figure.



## Concept Drift in Real Life

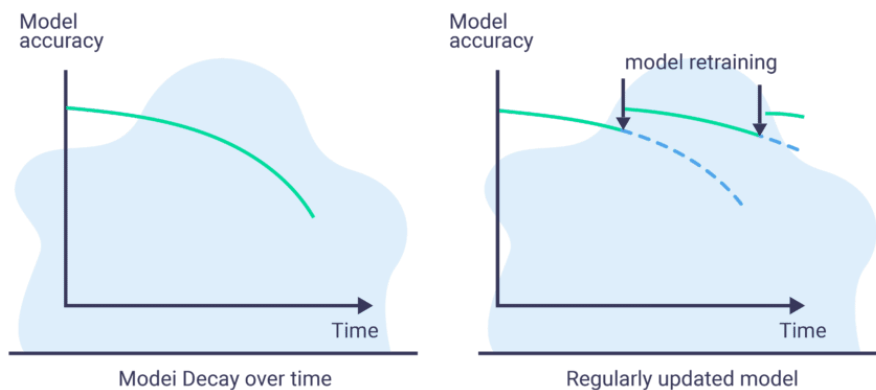
To understand how concept drift manifests itself in real life, we should look at our evaluation metrics. When searching for concept drift we view our data as a stream and proceed to examine how the evaluation metrics fluctuate over time.

Once a model is in its production environment, we can monitor its performance and be on the lookout for concept drift by looking at its performance on the data stream. By dividing the data stream into time frames (e.g. hour, day, week), we can understand what is changing in our data.

If the input distribution (i.e.  $p(X)$ ) changes but the true labels  $p(Y|X)$  don't, we can classify it as a virtual drift, and conclude that we have a change in the input data that hasn't affected the performance yet. If there is an actual change in the labels, i.e.  $p(Y|X)$  changed, that means that a real drift occurred during the time frame and we are detecting its effects.

For example, we have a trained model and we measure its performance using accuracy as the metric. We would like to view its accuracy over time: if the model's performance degrades over time that means there is **model decay**.

Model decay (i.e. model drift, staleness) is when the degradation of a model's performance over time and varies from use case to use case. Some models can perform well for years while other models' shelf life is less than a day.



It has been shown that regularly retraining the model before the point of significant degradation has occurred helps alleviate the issue to some extent, but in many cases, it is simply not enough and in others may even make matters worse.

- Sometimes it's hard to identify the point of the models' significant degradation.
- We don't always know the timeframe after which we need to retrain: a week may be too late, but hourly may cost too much to maintain.
- In many cases, the time window for retraining is in flux and so this parameter itself needs to be monitored and optimized.
- In many cases, retraining simply isn't enough. A significant change in the concept may require an entirely new type of model.
- In some cases, the change originates from flaws in other parts of the pipeline such as schema changes, dropped fields, application bugs, and more. In such cases, retraining a model with flawed data may have no effect or possibly even exacerbate the problem.

The best way to avoid concept drift ruining your models is by staying on top of it with the help of monitoring tools.

**Start Detecting Concept Drift in Your Models with aporia**

aporia has over 50 different types of customizable monitors to help you detect drift, bias, and data integrity issues. [Book a demo to see aporia in action and begin detecting drift in your machine learning models.](#)