General terms regarding process mining:

* Event: information related to an activity located on a date in time.
* Trace: sequence of events to carry out a process . Synonym: case or instance or execution.
* Play-in: I NFER gingiva from a model process automatically from raw event data .
* Throughput : number of units per unit of time.
* Achieved throughput : maximum number of units per unit of time
* Waiting time: waiting time for an activity to be carried out.
* Processing time: time it takes to carry out an activity without taking into account the delay of the process (waiting time).
* Cycle time ( Throughput time in Disk / Cellonis or as Sojurn time in ProM ) : total time it takes to carry out a process ( processing time + Waiting time) . Additionally, the articles mention the sum of the cycle times of two consecutive activities as a transition .
* Variant: traces that follow the same sequence of activities and therefore follow a pattern. Synonym: route (route).
* Process steps : stages of the process that include different activities ( Submission , Judgment by various responsible roles, Payment ) . These stages are defined by the organization of the contest.
* Fitness score: value of 0 .0 to 1.0 that represents how well a model fits the dat you the logs.
* Happy path : most common variant of the traces of a log.

General terminology about filtrates in process mining:

In process mining there are multiple types of filters, based on the techniques available in the different tools, we have identified the following filters in common :

* “ Variation filter ”: filtering of the most common variant. Equivalent to “ Variation Filter ” on Disk .
* "Time filter ": filtering traces over a period of time . Equivalent to Disk “ TimeFrame Filter ” .
* “ Activity filter ”: filtering traces that start in an activity X and end in an activity Y , or that do not contain certain activities , or that do contain certain activities . Equivalent to Disk “ Endpoints Filter ” or Cellonis “ Activity selection ” .
* " Process flow filter ": filtering traces that meet a certain pattern of activities. It is equivalent to “ F ollower filter ” in Disk or “ Process flow selection ” from Cellonis .
* " Attribute filter ": filtering traces depending on their attributes. It is equivalent to " Attribute filter " from Disco or " Attribute selection " from Cellonis .

* “ Rework filter ”: filtering of traces in which an activity occurs a number of times higher or lower than a limit. Equivalent to Cellonis " Rework selection " .
* " Performance filter " : filtering traces in the " cycle time" / " waiting time" / " processing time" between two activities is above or below a threshold. Equivalent to “ Throughput time selection ” in Cellonis or “ performance filter ” in Disco.

Terms relating to disk:

* " Variation Filter ": used to identify the most common variant. To do this, follow these steps:
* Definition of variants
* Calculation of the number of cases contemplated in each variant.
* Filtering of variants that do not contain a minimum number of cases.
* Filtering variants that exceed a maximum threshold of cases.
* Representation of a histogram with the remaining variants to decide which ones are included in the definition of the most common variant. For this, the frequencies of the cases of each variant are reviewed and a cut-off point is established from which all the variants will be considered to define the common variant.

* " Timeframe Filter ": filtering to select traces between certain dates.
* “ Endpoints Filter ” (synonym in Cellonis : “ Activity selection”): filtering to select cases that begin with an X activity and that end with a Z activity.
* " Attribute Filter " (synonym in Cellonis : " Attribute selection"): filtering cases depending on an attribute, the absence of an activity, or the presence of an activity.
* “ Follower Filter ” (synonym in Cellonis : “ Process flow selection”): used to filter cases in which one or more of the following conditions are met (in addition to attribute values ​​and time restrictions):
* “Eventually followed”: a pattern of activities occurs at some point in time, filtering even though they do not occur in a row (for example, if we want to filter by activities X and Y it does not matter if Z happens to half of the others) .
* "Directly followed": a pattern of activities or values ​​must occur directly subsequently (that is, X must occur just before Y, it is not allowed that they occur at different times).
* "Never eventually followed": activity X and activity Y can never occur in one case,
* "Never directly followed": activities X and Y can never occur in a row.

* “Max repetitions mode ”: Display mode, used to see where the most rework occurs. The values in the process map show the maximum number of times an activity was performed for a single case.
* "Performance filter ": filter of traces that exceed " cycle time", or cycle time of traces in which there is a start or end event . It can also be filtered by processing time or waiting time.

Terms regarding Celonis :

Types of filtering:

* " Attribute selection ": filtering traces by attributes.
* " Activity selection ": filtering of traces that begin or end in certain activities , or that contain certain activities .
* “ Process flow selection ”: filtering of traces in which one activity is or is not specifically followed by another activity . As in Disk, you can select that this pattern be followed directly or not (it is necessary that this pattern of activities be fulfilled immediately) , or that it be followed indirectly or not (it is necessary that there are two activities, but not immediately).

* “ Throughput time selection ” (Processing time selection ): filtering traces in which the “ cycle time” between two activities is higher or lower than a threshold.
* “ Rework selection ”: filtering of traces in which an activity occurs a number of times higher or lower than a limit.

Other methods:

* “ Conformance checker ”: method to compare an ideal process model with the existing one in a log to know to what degree the model fits with the data .

Terms regarding ProM:

* ActiTrac :

P Lugin of ProM , which is an algorithm of " clustering " grouping traces with similar behavior for the discovery process ( [https://ieeexplore.ieee.org/document/6507222](https://translate.google.com/translate?hl=es&prev=_t&sl=es&tl=en&u=https://ieeexplore.ieee.org/document/6507222) ). Follow the following steps in broad strokes:

* Selection of the traces that will form some initial groups depending on their frequency, using those that are more frequent.
* Grouping of the rest of the traces in the groups if it maintains the cohesion of the group (cohesion is measured by measuring the fitness of the process model that would be created). If they do not meet this condition, other groups are created until reaching a maximum number of groups.
* The remaining traces will be grouped with the group that worsens the least (measuring the fitness that the process model would have if it were added ).
* Finally, processes are obtained from the generated groups , with the largest group being the " happy path" for these data.

- Replay a log on a Petri Net ( [http://www.processmining.org/online/conformance\_checker](https://translate.google.com/translate?hl=es&prev=_t&sl=es&tl=en&u=http://www.processmining.org/online/conformance_checker) ) :

S and rerun l trace as the logs in a Petri net to check for missing tokens in the process or portions of the process occur delays (it can be understood token as the necessary condition for an activity is met). Finally , a process map can be obtained indicating where there are more delays or more tokens are missing.

- Process Comparator :

This plugin receives two groups of events and serves to compare their corresponding maps of p rocess.

- Inductive miner :

Algorithm to discover processes inductively separating activities from a lower level.

- Social network mining :

Algorithm to discover social networks in process mining.

More information in:

[http://www.processmining.org/\_media/publications/aalst2005g.pdf](https://translate.google.com/translate?hl=es&prev=_t&sl=es&tl=en&u=http://www.processmining.org/_media/publications/aalst2005g.pdf)

[https://www.futurelearn.com/info/courses/process-mining/0/steps/15652](https://translate.google.com/translate?hl=es&prev=_t&sl=es&tl=en&u=https://www.futurelearn.com/info/courses/process-mining/0/steps/15652)

- Doing similar task miner:

Plugin to detect the roles of the resources involved in a process.

Python terms:

Python libraries:

- Sberbank Bookstore: [https://pypi.org/project/sberbank/](https://translate.google.com/translate?hl=es&prev=_t&sl=es&tl=en&u=https://pypi.org/project/sberbank/)

Data processing algorithms:

- t- sne algorithm : unsupervised technique used to reduce the dimensionality of the data. M ore information on : [https://sitiobigdata.com/2018/08/27/algoritmo-t-sne-con-python/](https://translate.google.com/translate?hl=es&prev=_t&sl=es&tl=en&u=https://sitiobigdata.com/2018/08/27/algoritmo-t-sne-con-python/)

Statistical tests :

- ANOVA: is to compare the s deviations from the mean values of different groups to find differences s ignificativas. More information at: [https://www.cienciadedatos.net/documentos/19\_anova](https://translate.google.com/translate?hl=es&prev=_t&sl=es&tl=en&u=https://www.cienciadedatos.net/documentos/19_anova)