To process the data in subsequent processing steps, we have to discuss the way we encode the data. There are a multitude of ways to represent a log. We introduce four ways and the reason we choose the *hybrid-vector-representation*. Figure 1 shows schematically, how we can represent process data.

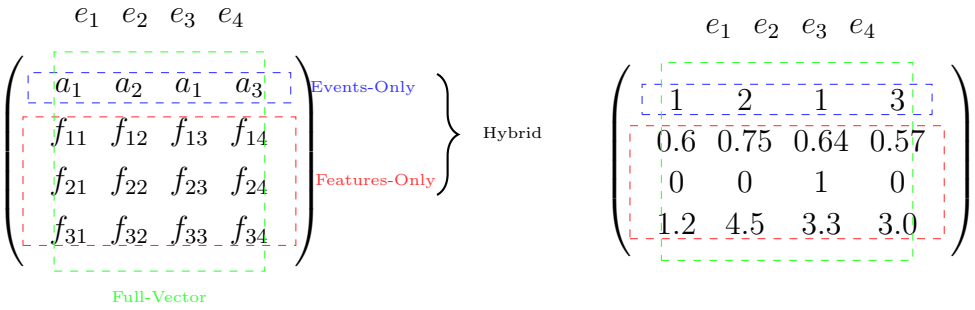


Figure 1: All four possible representations on an exemplary process instance.

First, we can choose to concentrate on *event-only-representation* and ignore feature attributes entirely. However, feature attributes hold significant amount of information. Especially in the context of using counterfactuals for explaining models as the path of a process instance might strongly depend on the event attributes. Similar holds for a *feature-only-representation*.

The first is a *single-vector-representation* with this representation we can simply concatenate each individual representation of every original column. This results in a matrix with dimensions (case-index, max-sequence-length, feature-attributes). The advantage of having one vector is the simplicity with which it can be constructed and used for many common frameworks. Here, the entire log can be represented as one large matrix. However, even though, it is simple to construct, it is quite complicated to reconstruct the former values. It is possible to do so by keeping a dictionary which holds the mapping between original state and transformed state. However, that requires every subsequent procedure to be aware of this mapping. Furthermore, we use methods, that treat events and their associated features (event attributes) separately. For instance, if we want to sample from a *Markov Model* with transition probabilities and emission probabilities, then it is much easier to first sample the event trajectory and then, the conditional feature attributes. Or, if we attempt to compute an edit distance between two sequences, it is easier to compute those, if we keep events and event attributes separate.

Therefore, we decide to keep the original sequence structure of events as a separate matrix and complementary to the remaining event attributes. If required, we turn the label encoded activities ad-hoc to one-hot encoded

vectors. Thus, this *hybrid-vector-representation* grants us greater flexibility. However, we now need to process two matrices. The first matrix has the dimensions (case-index, max-sequence-length) and the latter (case-index, max-sequence-length, feature-attributes).