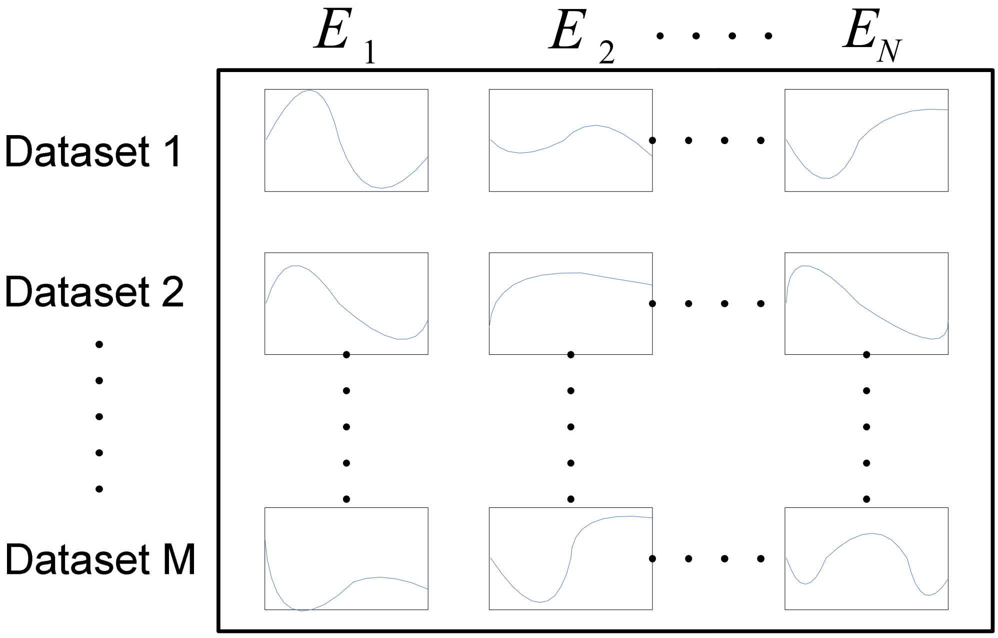
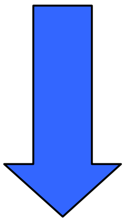
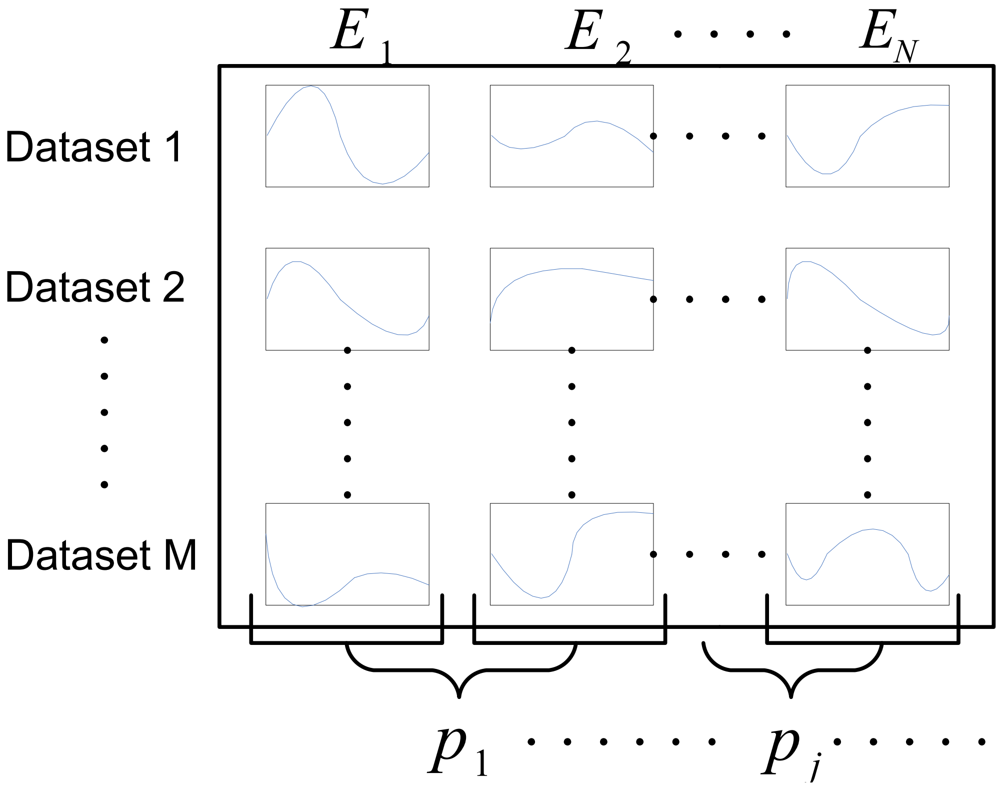
# Schematic Illustration of the Proposed Method

Supporting Material for Main Paper

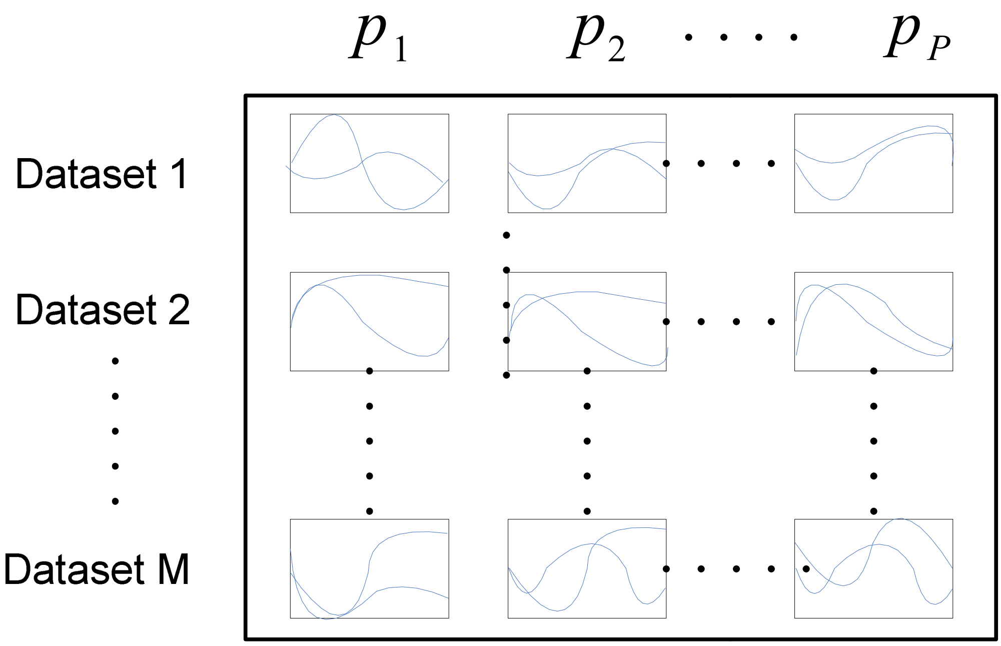


**Step 1.** Original data. Given $N$ elements, $\{E_1, E_2, ..., E_N\}$, we have $M$ datasets, each containing N time series profiles, each for one element. Within one dataset, the profiles have the same length.



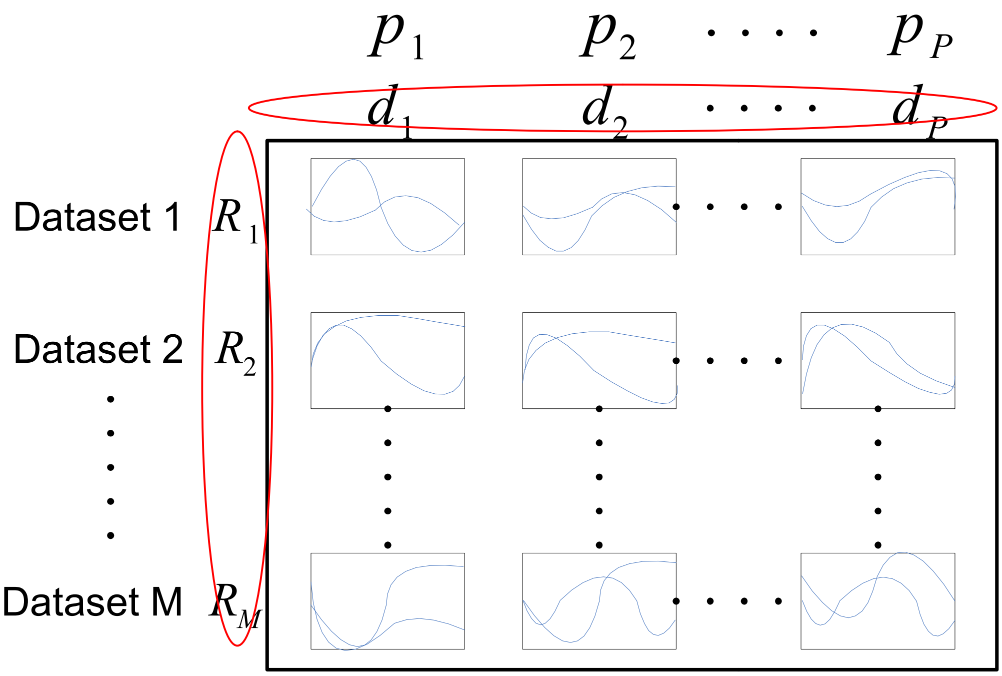


**Step 2.** Choosing a group of element pairs. We choose $P$ pairs of these elements. At least, a subset of this group of element pairs should display lagged correspondence in a subset of the $M$ datasets. Let $\mathcal{P} = \{< E_{a_j}, E_{b_j} >: j = 1,..., P\}$ denote the set of all selected $P$ pairs, where $E_t$ represents the $t^{th}$ element, and $a_j, b_j \in \{1,...., N\}$ . Let $p_j =< E_{a_j}, E_{b_j} >$ denote the $j^{th}$ pair.

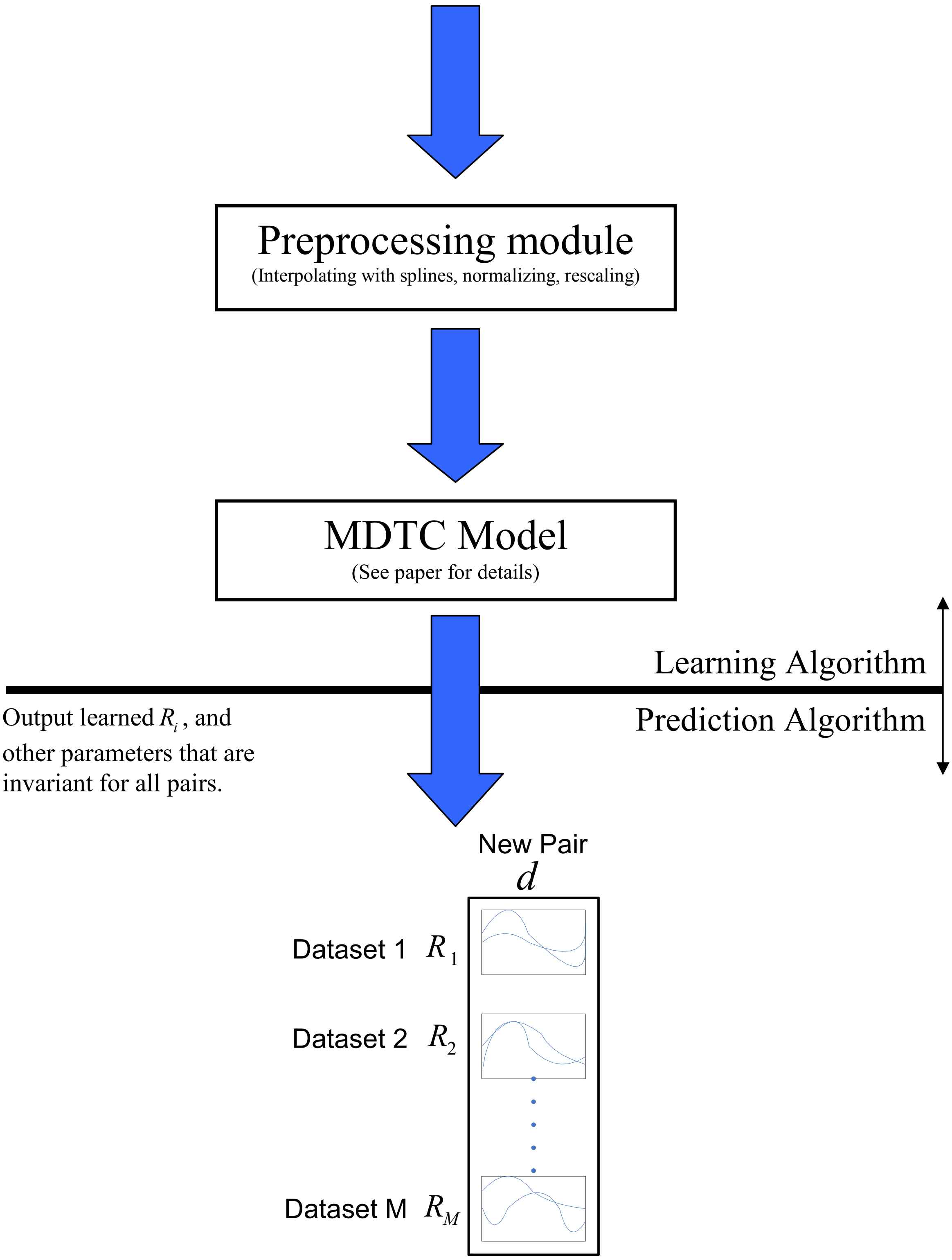


**Step 3.** The selected pairs under different conditions. Our goal is to assign a value to an indicator variable, $Z_{ij}$. This variable is set to 1 if there is a time lagged correspondence between element pair, $p_j =< E_{a_j}, E_{b_j} >$, in dataset $i$ and to 0 otherwise.

In the context of gene expression analysis, $p_j$ is a pair of TF and gene. We are interested to find whether these two genes have regulatory correspondence. If so, under which conditions their correspondence occurs.

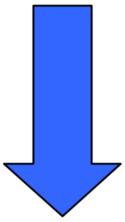


**Step 4**. Parameter association. We associate an unknown, linear, *time scale factor*, $R_i$, with each of the datasets. We also associate an unknown *canonical lag* $d_j$ with each of these pairs of elements, $p_j =< E_{a_j}, E_{b_j} >$. For such a pair, the expected lag in a given experiment will be the product of the experiment time scale factor and its canonical lag, $R_i \cdot d_j$.



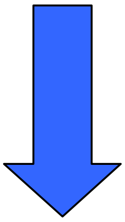
Output learned $R_i$, and other parameters that are invariant for all pairs.

**Step 1**. For a given new pair, the prediction algorithm runs the MDTC Model on a table with only one column, the column for this new pair. With learned $R_i$ fixed, this algorithm automatically learns $d$, the canonical lag for this new pair.



$$\begin{array}{l} \text{Dataset 1} \\ \text{Dataset 2} \\ \text{Dataset 3} \\ .. \\ .. \\ \text{Dataset M} \end{array} \begin{pmatrix} 0.1 \\ 0.8 \\ 0.2 \\ .. \\ .. \\ 0.6 \end{pmatrix}$$

**Step 2**. After convergence, the prediction algorithm assigns a confidence score vector to this new pair, representing the confidence that this pair has correspondence under each of the datasets.



$$\begin{array}{l} \text{Dataset 1} \\ \text{Dataset 2} \\ \text{Dataset 3} \\ .. \\ .. \\ \text{Dataset M} \end{array} \begin{pmatrix} 0 \\ 1 \\ 0 \\ .. \\ .. \\ 1 \end{pmatrix}$$

**Step 3**. We apply a threshold S (S = 0.5 in this case) to turn the confidence score into binary value. If a dataset has value 1, we predict that a correspondence occurs in this dataset, 0 otherwise.