nature portfolio

https://doi.org/10.1038/s41586-022-05639-4

Supplementary information

Organic reaction mechanism classification using machine learning

In the format provided by the authors and unedited

Supporting Information

Organic reaction mechanism classification using machine learning

Jordi Burés*, Igor Larrosa*

The University of Manchester, Department of Chemistry
Oxford Road, Manchester, M13 9PL, UK
E-mail: jordi.bures@manchester.ac.uk
E-mail: igor.larrosa@manchester.ac.uk

Table of Contents

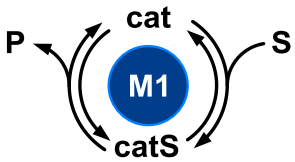
1. Materials and methods

Construction of the neural network, training and other analyses were carried out using Tensorflow 2.1[1], Python 3.7, NumPy 1.19.2[2], SciPy 1.6.1[3], Matplotlib 3.3.4[4], pandas 1.2.3[5], scikit-learn 0.24.1[6], and PyCM[7].

2. Definition of mechanisms

Each mechanism is defined by the reaction network and ordinary differential equations shown. We have applied chemical criteria to reduce the overlap between kinetic space of mechanisms leading to more chemical meaning mechanisms. The criteria applied are precisely detailed for each mechanism.

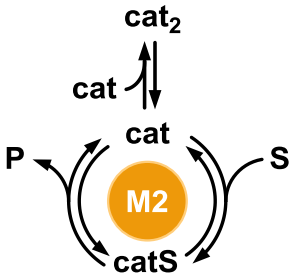
M1



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2)[catS]$$

Mechanism M1 is defined by the reaction network shown, with any set of kinetic constants.

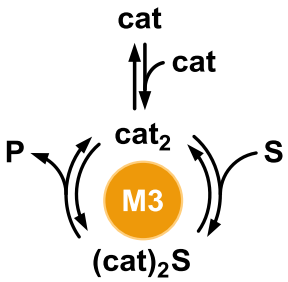
M2



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] + 2\,k_3[cat_2] - (k_1[S] + k_{-2}[P] + 2\,k_{-3}[cat])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2)[catS]$$
$$\frac{d[cat_2]}{dt} = k_{-3}[cat]^2 - k_3[cat_2]$$

Mechanism M2 is defined by the reaction network shown, with any set of kinetic constants such that the reaction run with the standard concentration of S and 5 mol% of catalyst has a significant amount of catalyst (> 10% of the total amount) as cat_2 during the first half of the reaction (between 20 and 50% of the maximum conversion possible).

M3



$$\frac{d[S]}{dt} = k_{-1}[(cat)_2S] - k_1[S][cat_2]$$

$$\frac{d[P]}{dt} = k_2[(cat)_2S] - k_{-2}[cat_2][P]$$

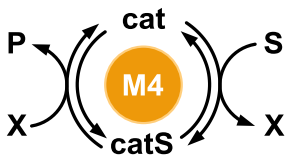
$$\frac{d[cat_2]}{dt} = (k_{-1} + k_2)[(cat)_2S] + k_3[cat]^2 - (k_1[S] + k_{-2}[P] + k_{-3})[cat_2]$$

$$\frac{d[(cat)_2S]}{dt} = (k_1[S] + k_{-2}[P])[cat_2] - (k_{-1} + k_2)[(cat)_2S]$$

$$\frac{d[cat]}{dt} = 2k_{-3}[cat_2] - 2k_3[cat]^2$$

Mechanism M3 is defined by the reaction network shown, with any set of kinetic constants such that the reaction run with the standard concentration of S and 5 mol% of catalyst has a significant amount of catalyst (> 5% of the total amount) as cat at the beginning of the reaction (< 10% of the maximum conversion possible).

M4



$$\frac{d[S]}{dt} = k_{-1}[X][catS] - k_1[S][cat]$$

$$\frac{d[P]}{dt} = k_2[X][catS] - k_{-2}[cat][P]$$

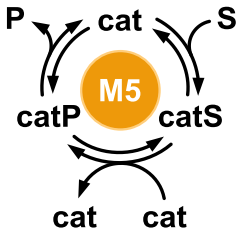
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[X][catS] - (k_1[S] + k_{-2}[P])[cat]$$

$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2)[X][catS]$$

$$\frac{d[X]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2)[X][catS]$$

Mechanism M4 is defined by the reaction network shown, with any set of kinetic constants.

M5



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$

$$\frac{d[P]}{dt} = k_3[catP] - k_{-3}[cat][P]$$

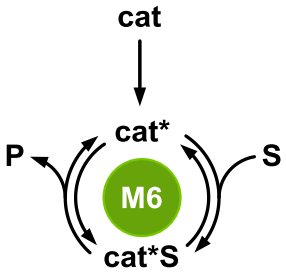
$$\frac{d[cat]}{dt} = k_{-1}[catS] + k_3[catP] - (k_1[S] + k_{-3}[P])[cat]$$

$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[catP])[cat] - (k_{-1} + k_2[cat])[catS]$$

$$\frac{d[catP]}{dt} = k_2[catS][cat] + k_{-3}[P][cat] - (k_3 + k_{-2}[cat])[catP]$$

Mechanism M5 is defined by the reaction network shown, with any set of kinetic constants.

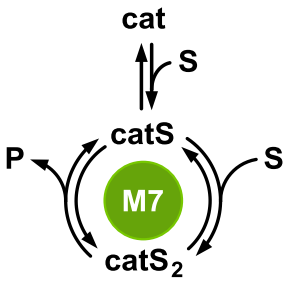
M6



$$\frac{d[S]}{dt} = k_{-1}[cat^* S] - k_1[S][cat^*]$$
$$\frac{d[P]}{dt} = k_2[cat^* S] - k_{-2}[cat^*][P]$$
$$\frac{d[cat]}{dt} = -k_3[cat]$$
$$\frac{d[cat^*]}{dt} = k_3[cat] + (k_{-1} + k_2)[cat^* S] - (k_1[S] + k_{-2}[P])[cat^*]$$
$$\frac{d[cat^* S]}{dt} = (k_1[S] + k_{-2}[P])[cat^*] - (k_{-1} + k_2)[cat^* S]$$

Mechanism M6 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction run with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 10 and 90% of active catalyst (cat* + cat*S) at 20% of the maximum conversion.

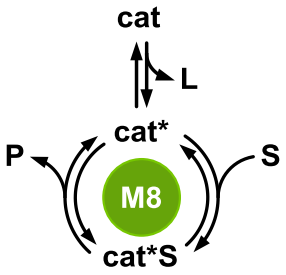
M7



$$\frac{d[S]}{dt} = k_{-1}[catS_2] - k_1[S][catS] + k_{-3}[catS] - k_3[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS_2] - k_{-2}[catS][P]$$
$$\frac{d[cat]}{dt} = k_{-3}[catS] - k_3[S][cat]$$
$$\frac{d[catS]}{dt} = k_3[S][cat] - k_{-3}[catS] + (k_{-1} + k_2)[catS_2] - (k_1[S] + k_{-2}[P])[catS]$$
$$\frac{d[catS_2]}{dt} = (k_1[S] + k_{-2}[P])[catS] - (k_{-1} + k_2)[catS_2]$$

Mechanism M7 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction run with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 10 and 80% of active catalyst (catS + $catS_2$) at 20% of the maximum conversion.

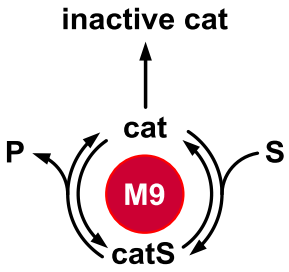
M8



$$\frac{d[S]}{dt} = k_{-1}[cat^* S] - k_1[S][cat^*]$$
$$\frac{d[P]}{dt} = k_2[cat^* S] - k_{-2}[cat^*][P]$$
$$\frac{d[cat]}{dt} = k_{-3}[L][cat^*] - k_3[cat]$$
$$\frac{d[cat^*]}{dt} = k_3[cat] - k_{-3}[L][cat^*] + (k_{-1} + k_2)[cat^* S] - (k_1[S] + k_{-2}[P])[cat^*]$$
$$\frac{d[cat^* S]}{dt} = (k_1[S] + k_{-2}[P])[cat^*] - (k_{-1} + k_2)[cat^* S]$$
$$\frac{d[L]}{dt} = k_3[cat] - k_{-3}[L][cat^*]$$

Mechanism M8 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 10 and 90% of active catalyst (cat* + cat*S) at 50% of the maximum conversion.

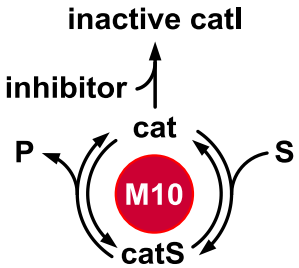
M9



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P] + k_{-3})[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2)[catS]$$
$$\frac{d[\text{inactive cat}]}{dt} = k_{-3}[cat]$$

Mechanism M9 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 50 and 90% of active catalyst (cat + catS) at 50% of the maximum conversion.

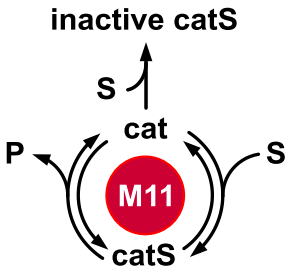
M10



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P] + k_{-3}[inhibitor])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2)[catS]$$
$$\frac{d[inhibitor]}{dt} = -k_{-3}[inhibitor][cat]$$
$$\frac{d[\text{inactive catI}]}{dt} = k_{-3}[inhibitor][cat]$$

Mechanism M10 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has more than 10% of the inactive catalyst (inactive catI) at 50% of the maximum conversion.

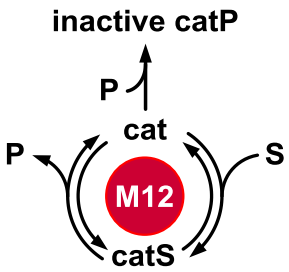
M11



$$\frac{d[S]}{dt} = k_{-1}[catS] - (k_1 + k_{-3})[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P] + k_{-3}[S])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2)[catS]$$
$$\frac{d[\text{inactive catS}]}{dt} = k_{-3}[S][cat]$$

Mechanism M11 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 50 and 90% of active catalyst (cat + catS) at 50% of the maximum conversion.

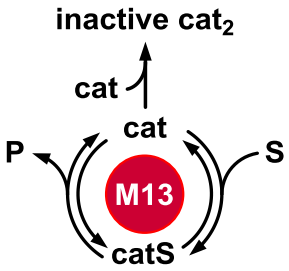
M12



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - (k_{-2} + k_{-3})[P][cat]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P] + k_{-3}[P])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2)[catS]$$
$$\frac{d[\text{inactive catP}]}{dt} = k_{-3}[P][cat]$$

Mechanism M12 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 50 and 90% of active catalyst (cat + catS) at 50% of the maximum conversion.

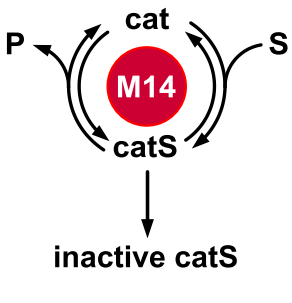
M13



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P] + 2\,k_{-3}[cat])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2)[catS]$$
$$\frac{d[\text{inactive cat}_2]}{dt} = k_{-3}[cat]^2$$

Mechanism M13 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 50 and 80% of active catalyst (cat + catS) at 50% of the maximum conversion.

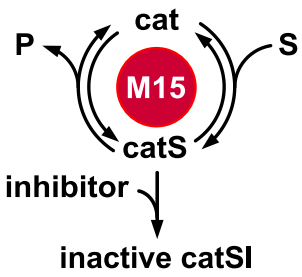
M14



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2 + k_{-3})[catS]$$
$$\frac{d[\text{inactive catS}]}{dt} = k_{-3}[catS]$$

Mechanism M14 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 50 and 90% of active catalyst (cat + catS) at 50% of the maximum conversion.

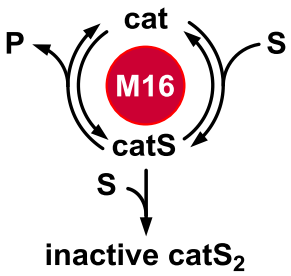
M15



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2 + k_{-3}[inhibitor])[catS]$$
$$\frac{d[inhibitor]}{dt} = -k_{-3}[inhibitor][catS]$$
$$\frac{d[inactive\ catSI]}{dt} = k_{-3}[inhibitor][catS]$$

Mechanism M15 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has more than 10% of inactive catalyst (inactive catSI) at 50% of the maximum conversion.

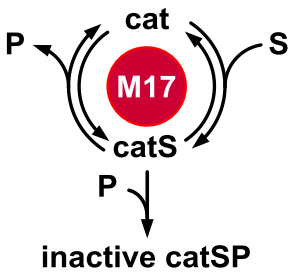
M16



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat] - k_{-3}[S][catS]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2 + k_{-3}[S])[catS]$$
$$\frac{d[inactive\ catS_2]}{dt} = k_{-3}[S][catS]$$

Mechanism M16 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 50 and 90% of active catalyst (cat + catS) at 50% of the maximum conversion.

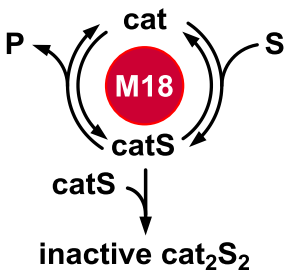
M17



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P] - k_{-3}[P][catS]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2 + k_{-3}[P])[catS]$$
$$\frac{d[inactive\ catSP]}{dt} = k_{-3}[P][catS]$$

Mechanism M17 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 50 and 90% of active catalyst (cat + catS) at 50% of the maximum conversion.

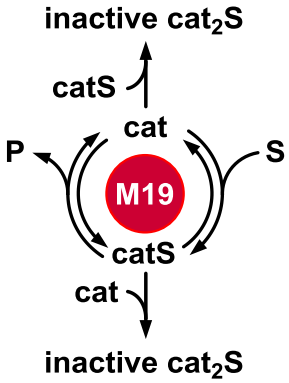
M18



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2 + k_{-3}[catS])[catS]$$
$$\frac{d[\text{inactive cat}_2\,S_2]}{dt} = k_{-3}[catS]^2$$

Mechanism M18 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 50 and 80% of active catalyst (cat + catS) at 50% of the maximum conversion.

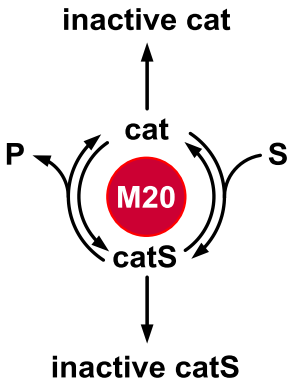
M19



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P] + k_{-3}[catS])[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2 + k_{-3}[cat])[catS]$$
$$\frac{d[\text{inactive cat}_2\,S]}{dt} = k_{-3}[cat][catS]$$

Mechanism M19 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 50 and 80% of active catalyst (cat + catS) at 50% of the maximum conversion.

M20



$$\frac{d[S]}{dt} = k_{-1}[catS] - k_1[S][cat]$$
$$\frac{d[P]}{dt} = k_2[catS] - k_{-2}[cat][P]$$
$$\frac{d[cat]}{dt} = (k_{-1} + k_2)[catS] - (k_1[S] + k_{-2}[P] + k_{-3})[cat]$$
$$\frac{d[catS]}{dt} = (k_1[S] + k_{-2}[P])[cat] - (k_{-1} + k_2 + k_{-4})[catS]$$
$$\frac{d[\text{inactive cat}]}{dt} = k_{-3}[cat]$$
$$\frac{d[\text{inactive catS}]}{dt} = k_{-4}[catS]$$

Mechanism M20 is defined by the reaction network shown, with any set of kinetic constants such that at least one reaction with the standard concentration of S and a catalyst loading between 3 and 7 mol% of catalyst has between 50 and 90% of active catalyst (cat + catS) and there is a significant amount of each inactive catalyst (> 5% of inactive cat and > 5% of inactive catS) at 50% of the maximum conversion.

3. Data generation

To create the kinetic data, we solved the set of ordinary differential equations that describe the kinetic behavior of each mechanism as function of kinetic constants (k_1, … k_q) and concentration of chemical species (Fig. S1a left), using solve_ivp from the scipy Python package, based on a LSODA solver. For each mechanism, we established a pool of unique sets of kinetic constants ($\mathbf{s^x}$), with a total of 108,208 unique kinetic sets across all mechanisms (Fig. S1a left). All kinetic constants were picked randomly in the range 10^5-10^{-5} A.U. and rounded to three significant figures. The chosen kinetic constants must fulfill the conditions defined for their corresponding mechanism as set in section 2 and allow for reversible and irreversible reactions with maximum yields between 50 and 100%.

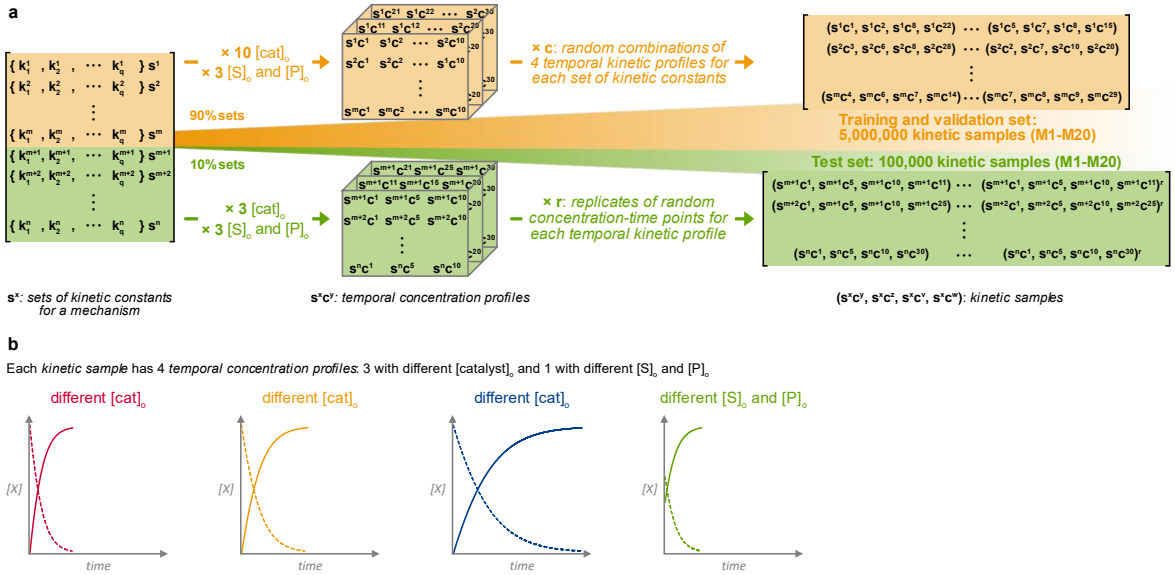


Fig. S1. Schematic of process for generation of training, validation, and test data.

The pool of kinetic constants for each mechanism was split, with 90% used for training and validation and the remaining 10% to create the test set. This splitting of unique sets of kinetic constants ensures the absence of overlap between training and test sets. For each set of kinetic constants ($\mathbf{s^x}$), we generated 30 temporal concentration profiles ($\mathbf{s^xc^y}$) varying the initial catalyst concentration ($[\mathbf{cat}]_0$) randomly across the range 1-10 mol %, and with three different initial concentrations of substrate and product, resulting in a total pool of approximately 7M labelled kinetic profiles (Fig. S1a, middle). From this 7M pool of individual kinetic profiles, we generated 5M kinetic samples containing each four kinetic profiles for the same set of kinetic constants ($\mathbf{s^xc^y}$, $\mathbf{s^xc^z}$, $\mathbf{s^xc^v}$, $\mathbf{s^xc^w}$).

Each kinetic sample of the training and validation set contains 4 temporal concentration profiles, three of identical $[\mathbf{S}]_0$ but different $[\mathbf{cat}]_0$, and the fourth with a smaller $[\mathbf{S}]_0$ with the difference as added **P** (Fig. S1a right and Fig. S1b). Specifically, the $[S]_0$ is always 1 for the first three temporal concentration profiles, and in the last one $[\mathbf{S}]_0$ is a random value between 0.4 and 0.8, with a an additional $[\mathbf{P}]_0$ of 1 – $[\mathbf{S}]_0$ (to make it a 'same excess with product added' type of experiment[8]). Despite we always use $[\mathbf{S}]_0$ = 1 for training, the trained model is applicable to any value of $[\mathbf{S}]_0$, since one only needs to normalize by dividing all concentrations in the temporal profiles by $[\mathbf{S}]_0$, as shown in the application to experimental data example (Fig 4c).

For the 5,000,000 samples in the training and validation sets, the $[\mathbf{cat}]_0$ for the first three profiles are three random values between 0.01 and 0.10 (corresponding to a range of 1-10 mol % catalyst loading). This leads to a robust training able to deal with samples with $[cat]_0$ close to each other as well as far apart from each other. The $[\mathbf{cat}]_0$ for the fourth profile is the same as one of the first three (chosen randomly). Furthermore, only 20 concentration-time points were randomly selected from each kinetic profile, ensuring a breadth of possible temporal concentration profiles including examples reaching different conversions.

The 100,000 kinetic samples of the test set (5,000 per mechanism) were created similarly, but in this case the three $[\mathbf{cat}]_0$ were randomly chosen in the ranges 1-2 mol %, 4.5-5.5 mol % and 9-10 mol %, respectively. In addition, the concentration time points were chosen ensuring a good distribution across the whole reaction profile. Both of these choices mirror a typical sampling strategy when collecting kinetic data experimentally. Various tests sets were created with varying amounts of concentration time points per kinetic sample, in the range 2-20 time points. In addition, test sets containing varying amounts of Gaussian error in concentration were created, with the gauss function in the random Python library, applying error with standard deviations ranging from 0.5-5%.

Formatted training, validation and test sets are provided in the TrainValTest.zip file (SI Section 7.5.).

3.1. Evaluation of similarity between test set and training set

In order to assess how similar the test samples are to the samples in the training set we measured the Euclidean distance of each test sample to each sample in the training set for the corresponding mechanism and recorded the closest match. The Euclidean distance was defined as the square root of the sum of the squared difference between each concentration-time point in the samples, with each pair of temporal concentration profiles across both samples normalized in time and concentrations to ensure similar weighting. In Fig. S2 we plot the distribution of closest matches for each mechanism.

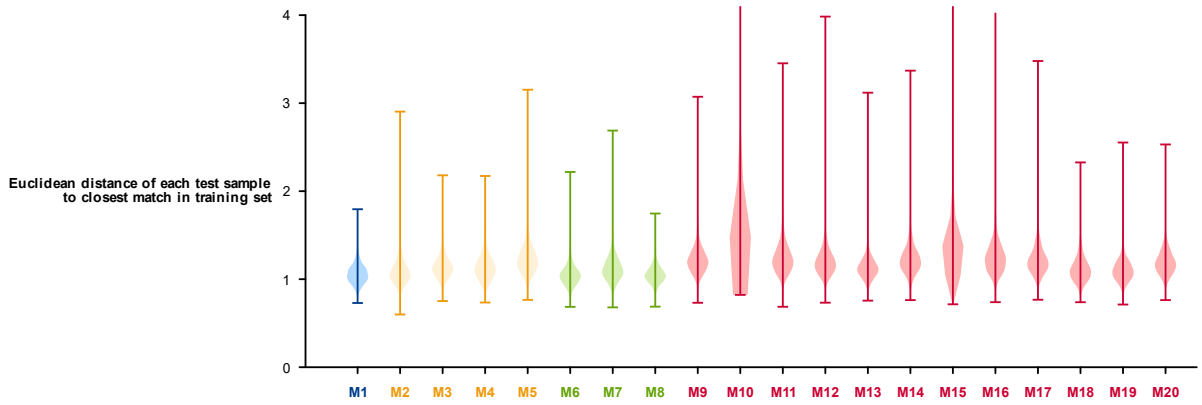


Fig. S2. Violin plot of the distribution of Euclidean distances of the test samples to their closest match in the training set.

As shown in Fig. S2, the Euclidean distance is non-zero, which demonstrate that the test set does not contain identical samples to the ones in the training set. The distribution of distances varies in each mechanism, due to the randomness of the generation process and the intrinsic kinetic variability of each mechanism.

3.2. Training and validation set input formatting

The training and validation sets have two inputs: input_1 contains the four $[\mathbf{cat}]_0$ for each profile, with shape (number of samples, 4); input_2 includes three columns (time, $[\mathbf{S}]_t$ and $[\mathbf{P}]_t$) for each of the four temporal concentration profiles, with shape (number of samples, 21, 12). Each sample's kinetic profiles contain 20 time points plus the time zero point.

Each sample has been normalized with min-max normalization scaling concentrations and times to the range 0 to 1, to guarantee that the trained model performs optimally for reactions of any duration and any initial concentration of substrate.

The training set contains 4.95M samples with shapes (4950000, 4) and (4950000, 21, 12), for input_1 and input_2, respectively.

The validation set's input_1 and input_2 have shapes (50000, 4) and (50000, 21, 12), respectively.

Each sample is composed of four temporal concentration profiles generated from the same set of kinetic constants, such that: profiles 1-3 have $[\mathbf{S}]_0 = 1$, $[\mathbf{P}]_0 = 0$ and 3 different $[\mathbf{cat}]_0$ between 1 and 10% ordered from lower to higher $[\mathbf{cat}]_0$, plus a fourth profile with $[\mathbf{S}]_0 < 1$, $[\mathbf{P}]_0 > 0$, and a $[\mathbf{cat}]_0$ matching one from the previous three profiles. For all samples $[\mathbf{S}]_0 + [\mathbf{P}]_0 = 1$.

The label for each sample is the corresponding mechanism used to generate the kinetics of that sample, encoded as a one-hot vectors as follows:

M1: (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M2: (0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M3: (0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M4: (0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M5: (0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M6: (0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M7: (0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M8: (0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M9: (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M10: (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M11: (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0)
M12: (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0)
M13: (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0)
M14: (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
M15: (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0)
M16: (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0)
M17: (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0)
M18: (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0)
M19: (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0)
M20: (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1)

The training and validation set can be downloaded from https://doi.org/10.48420/16965292 (SI Section 7.5.).

3.3. Test set input formatting

The test set has two inputs: input_1 contains the four $[\mathbf{cat}]_0$ for each profile, with shape (number of samples, 4); input_2 includes three columns (time, $[\mathbf{S}]_t$ and $[\mathbf{P}]_t$) for each of the four temporal concentration profiles, with shape (number of samples, 7, 12). Each sample's kinetic profiles contain 6 time points plus the time zero point.

Each sample has been normalized with min-max normalization scaling concentrations and times to the range 0 to 1, to guarantee that the trained model performs optimally for reactions of any duration and any initial concentration of substrate.

The test set contains 100k samples (5000 per label) with input shapes (100000, 4) and (100000, 7, 12), for input_1 and input_2, respectively.

All kinetic constants used to generate the samples in the test set are different from those used to generate the training and validation sets and produce unique kinetic profiles. Thus, no samples in the test set have been previously seen by the model during training.

Further variants of the test set were created with 20, 15, and 10, 9, … 1 time points, and by introducing random Gaussian error with standard deviations of 1, 2, … 5% to all the points of the temporal concentration profiles.

The test set can be downloaded from https://doi.org/10.48420/16965292 (SI Section 7.5).

4. Main deep learning model

4.1. Model description

The model requires two inputs: input_1 contains the four $[\mathrm{cat}]_0$ for each profile, with shape (number of samples, 4); input_2 includes time, $[\mathbf{S}]_t$ and $[\mathbf{P}]_t$ for each of the four temporal concentration profiles, with shape (number of samples, number of time points + 1, 12).

Input_1 is fed to a 200 neuron FCC layer with ReLu activation; Input_2 is fed to a 200 neuron bidirectional LSTM, followed by a second LSTM. The output of the FCC layer and recurrent layers is then concatenated and fed to another 200 neuron FCC layer with ReLu activation, followed by a 20 neuron softmax layer, that outputs predicted probabilities for each mechanism.

Output is on the form of a one-hot vector encoded for the different mechanisms M1-20.

Model summary:

Layer (type)	Output Shape	Number of parameters	Connected to
input_1 (InputLayer)	[(None, 4)]	0	
input_2 (InputLayer)	[(None, None, 12)]	0	
LSTM1 (LSTM)	(None, None, 200)	170400	input_2[0][0]
DENSE1 (Dense)	(None, 200)	1000	input_1[0][0]
LSTM2 (LSTM)	(None, 200)	320800	lstm[0][0]
Concatenation (Concatenate)	(None, 400)	0	dense[0][0] lstm_1[0][0]
DENSE2 (Dense)	(None, 200)	80200	concatenate[0][0]
SOFTMAX (Dense)	(None, 20)	4020	dense_1[0][0]

Total params: 576,420

Trainable params: 576,420

Non-trainable params: 0

Supplementary Table 1. Model Summary

A schematic representation of the model architecture is shown in Fig. S3.

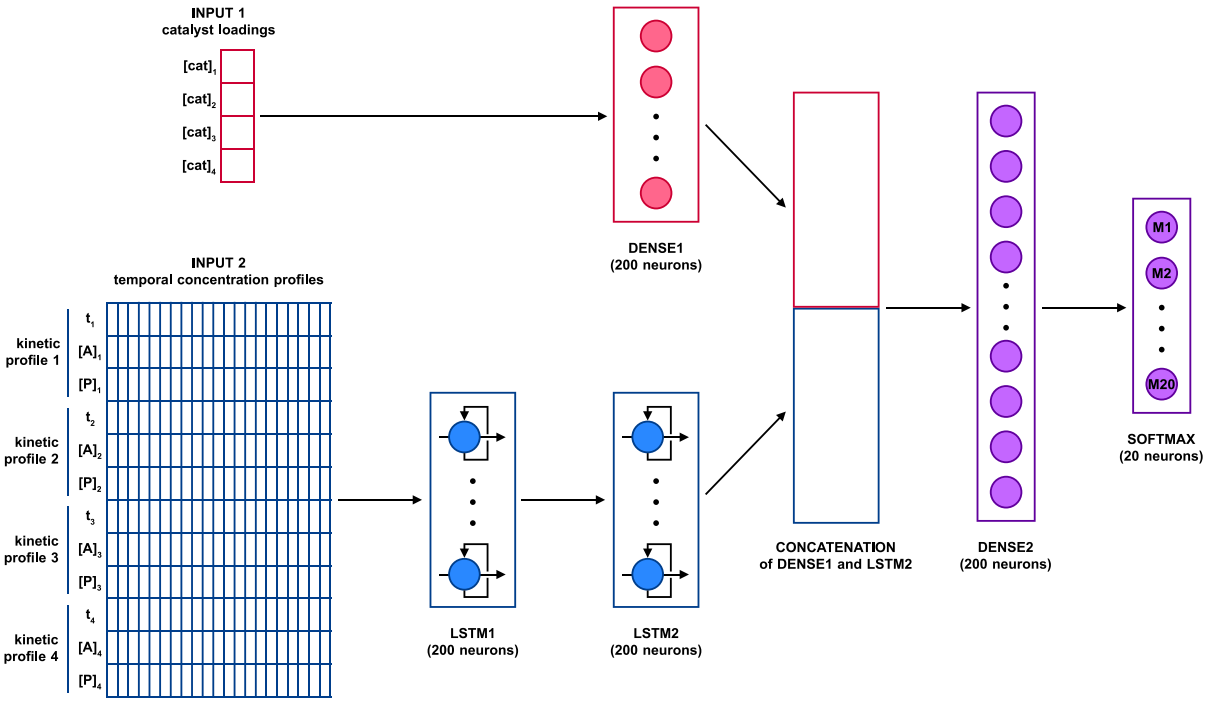


Fig. S3. Model architecture.

4.2. Model training and hyperparameters

The model was constructed and trained with Tensorflow 2.1.0 using Adam optimizer with a learning rate of 0.0001.

The model was trained with the training set and the validation set evaluated after every epoch. Data augmentation on the training set was carried out on-the-fly as follows:

- on each epoch, the sample order was randomized;
- the training set was divided into batches of 2048 samples;
- each batch was cut to a number of time points chosen randomly from the list: [3,4,5,6,7,8,9,10,15,20];
- Each batch was subjected to random error following a Gaussian distribution on all concentration values with a standard deviation of either 0, 0.5, 1, or 2%. Any negative values were clipped to 0.

The same data augmentation was applied to the validation set, with batches of 25 samples, instead of 2048.

Categorical cross-entropy (loss) and categorical accuracy for both training and validation sets were monitored during training (Fig S1). Training was stopped when categorical accuracy for the validation set did not increase for 300 consecutive epochs, and the weights providing maximum validation categorical accuracy were chosen for the model. The final model was trained for a total of 1827 epochs.

Evaluation of the standard test set (6 time points, 100,000 samples) on this model, reveals a categorical accuracy of 0.927, top-3 accuracy of 1.0 and CEN of 0.053.

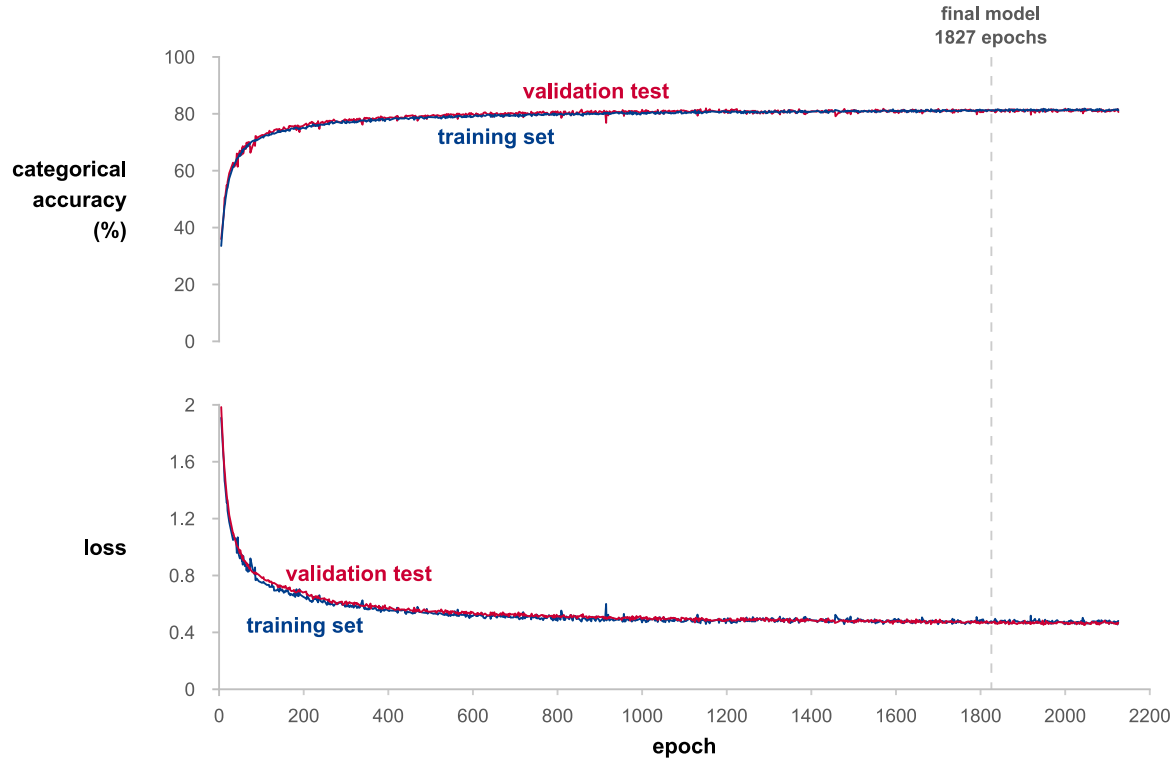


Fig. S4. Evolution of categorical cross-entropy (loss) and categorical accuracy during training.

4.3. Training with scrambled labels

Training the model with a training set where the labels have been scrambled results in no learning, with accuracy in both the training and validations sets remaining at the random level of 0.050 ± 0.001 across 217 epochs (yellow line in Figure S5). This is not unexpected since our training set is very large (5M samples), which makes overfitting nearly impossible with a model as small as ours (800 neurons).

Evaluation of the (non-scrambled) standard test set (6 time points, 100,000 samples) on this model, also reveals a categorical accuracy of 0.04, top-3 accuracy of 0.12 and CEN of 0.62.

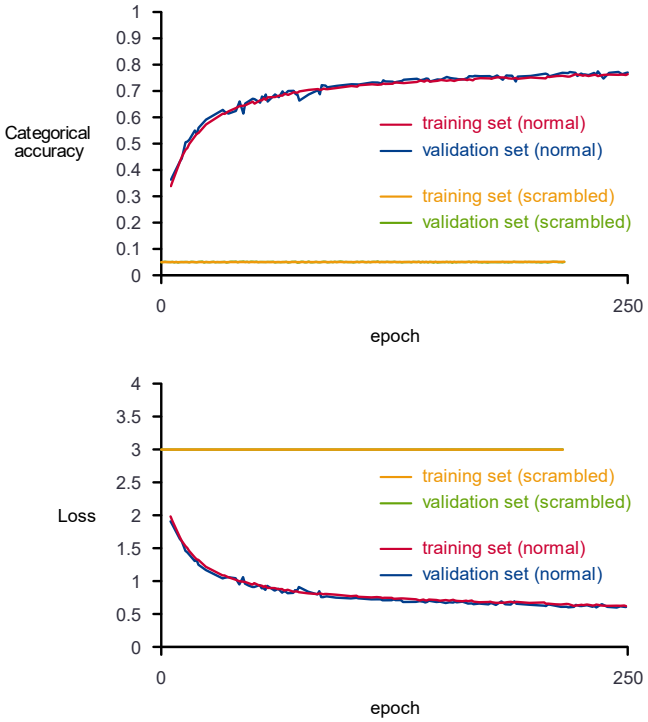


Fig. S5. Evolution of categorical cross-entropy (loss) and categorical accuracy during training with

4.4. Model cross-validation

We carried out *k*-fold cross-validation (*k* = 10) using the 5,000,000 samples in the original (20 time point per run) training set as follows:

1) The 5M samples were divided into 10 groups of 500,000 samples. Importantly, the sample separation was carried out in a way that the samples in each of these 10 groups have kinetic constants different from the samples in all other groups, so that there is no repetition between groups.
2) For each cross-validation iteration we picked one group, which we split equally into a test and a validation set (250,000 samples each), with the remaining 9 groups combined into a training set (4,500,000 samples). The model was then trained with the training set for 2,000 epochs. The trained model performance was then evaluated with the test set. This process was repeated 10 times, each time using a different group for validation and testing.

The categorical accuracy of each test set for each iteration is reported in the Supplementary Table 2.

Model	1	2	3	4	5	6	7	8	9	10
Categorical accuracy	0.920	0.919	0.921	0.923	0.924	0.921	0.919	0.917	0.916	0.920

Supplementary Table 2. Categorical accuracy of each cross-validation iteration

It is clear that the categorical accuracies from each iteration are spread across an extremely narrow range with an average categorical accuracy of 0.920 ± 0.005 (95% confidence of the standard deviation). This is a result of the large training set, containing a wide variety of representations for each mechanism, leading to a highly convergent training of the model.

4.5. Analysis of grouping performance

The model performs with a 99.996% accuracy when allowing for grouping mechanisms up to a threshold of 0.99 cumulative predicted probability.

We analysed the distribution of the predictions for the 6,493 samples where the model groups 3 or more mechanisms. This analysis revealed that in 65% of those, the 'correct' mechanism was the one proposed with the highest probability, in 30% of the cases the second highest probability proposal was the 'correct' one, and in 5% of the cases the third highest probability proposal was the 'correct' one (Fig. S6, left). Mechanisms grouped together but in 4th or higher positions were never the correct answer.

An analysis of the distribution of probability for the correct answers (Fig. S6, right) shows a bimodal distribution for those 'correct' in the first position, with those samples around 0.5 probability corresponding mainly to mechanisms M11 and M14 (as one would expect from the individual probability plots shown in Fig 3b in the manuscript).

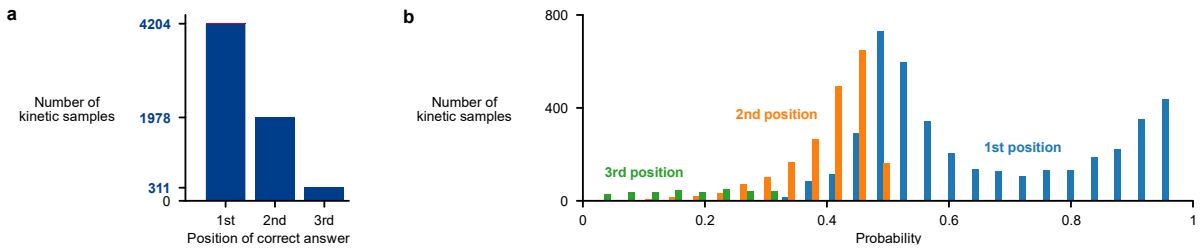


Fig. S6. (a) Position of correct answer in all test predictions where the model groups 3 or more mechanisms together. (b) Probability of predicted correct answer by position.

4.6. Effect of outliers in test set

We took the test sets with 20 and 6 time points and we randomly introduced one outlier time point with a fixed absolute error to each sample. Evaluation of this new test sets showed that the model can cope well with outliers of up to 25% absolute error in the 20 time point set, with accuracies still above 96% (orange line in Fig. S7). Instead, in the test set with only 6 time points, accuracy falls below 90% for outliers with absolute errors above 20% (red line in Fig. S7).

Remarkably, when the same process is carried out with a 20 time point and 6 time point test sets that already contain error with a standard error of 1% the effect of outliers is somewhat reduced, leading to higher accuracies in both test sets (blue and green lines in Fig. S7). This can be explained by the model recognizing the presence of error in the samples and grouping more mechanisms together as a result.

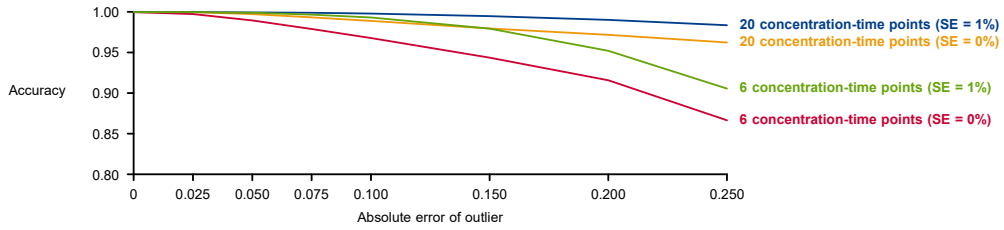


Fig. S7. Accuracy of model on test sets containing one outlier with absolute error in the range 0-0.25 (equivalent to 0-25% absolute error on concentration values).

4.7. Effect of error in the grouping of mechanisms by the machine learning model

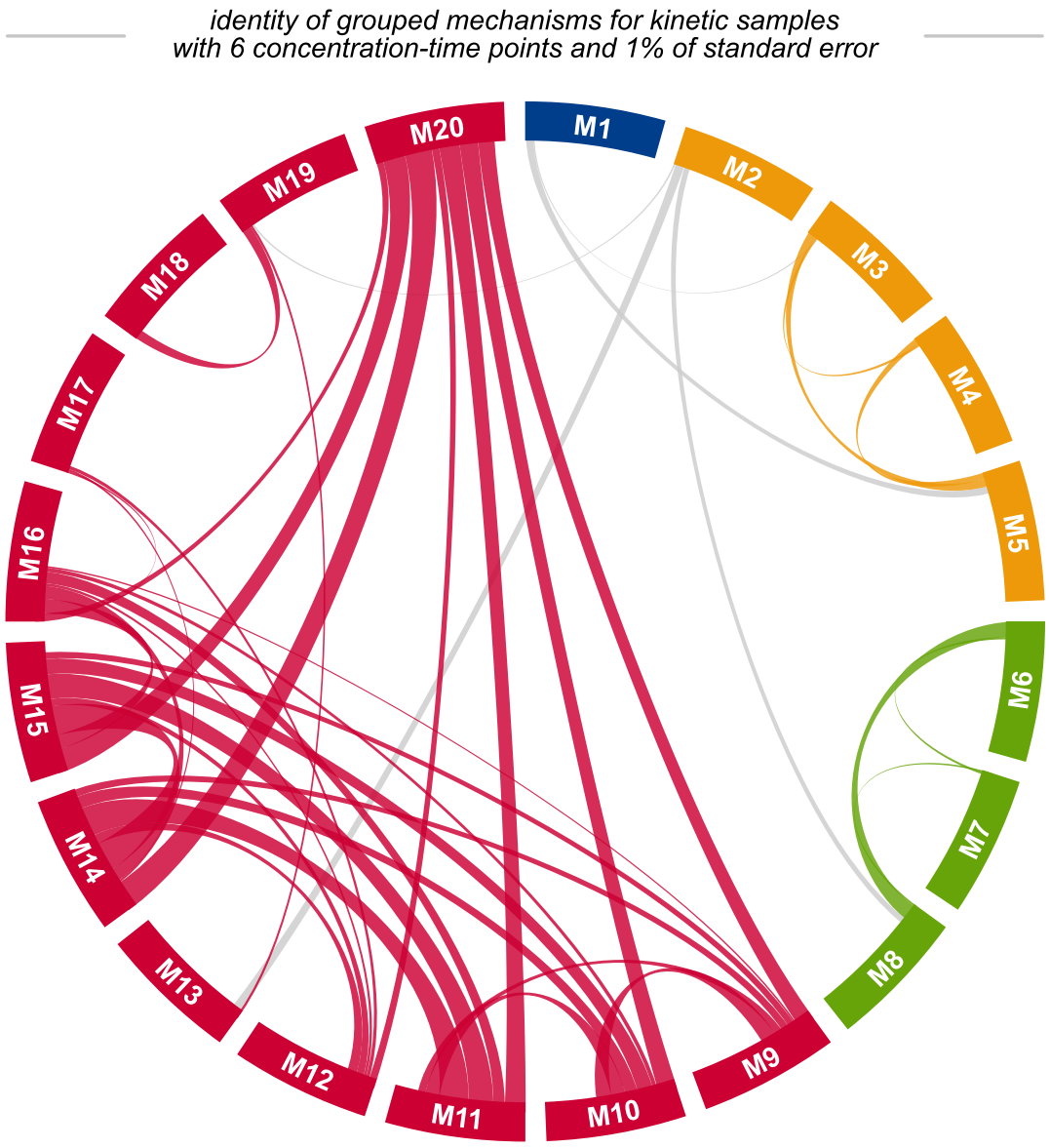


Fig. S8. Circos plots showing the mechanisms grouped together in the model's predictions when analyzing kinetic data with 1% standard error.

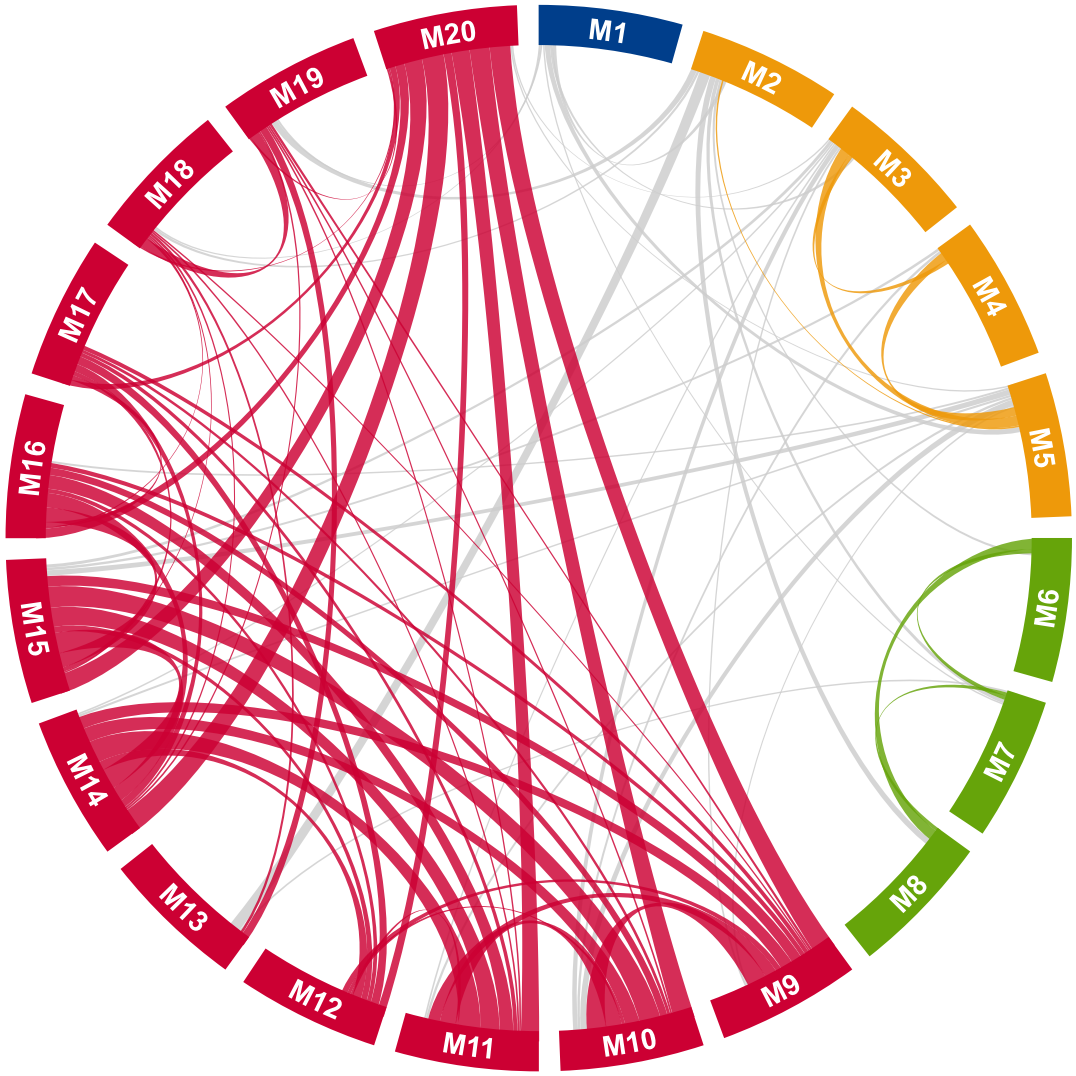


Fig. S9. Circos plots showing the mechanisms grouped together in the model's predictions when analyzing kinetic data with 5% standard error.

5. Training with other model architectures

5.1. Similarity searches

A similarity search approach faces numerous problems. Our training and test sets are built with $[\mathbf{cat}]_0$ that are randomly chosen between 1-10 mol%, as well as a random $[\mathbf{S}]_0$ for the same excess experiment. Consequently, for a particular test sample, it is highly unlikely that there will exist a training sample for the correct mechanism fulfilling all the following conditions: 1) that all $[\mathbf{cat}]_0$ for the 4 kinetic profiles are very similar between test and train samples, 2) that the kinetic constants are similar enough to have kinetic profiles with the same shape and similar (normalized) times to completion for all 4 kinetic profiles between test and train samples, 3) that the $[\mathbf{S}]_0$ and the $[\mathbf{P}]_0$ in the same excess experiment are very similar between the test and train samples.

A deep learning model, instead, can be trained to learn and extract the most important features of the kinetic profiles and is able to extrapolate (or interpolate) for missing data (e.g. exact $[\mathbf{cat}]_0$ or $[\mathbf{S}]_0$ values).

As an example, we ran a k-Nearest Neighbour with Euclidean Distance based similarity search algorithm (from the scikit-learn Python library) using the full 5,000,000 sample training set. For evaluation we used our test set made of 20 time points per sample and 100,000 total samples. We optimized k with a small fraction of the test set (1,000 samples), finding that optimum accuracy was achieved with k = 30. We then evaluated the model on the full test set, showing a categorical accuracy of just 35% and CEN of 0.59. The resulting confusion matrix is shown in Fig. S10. Random choice would deliver a 5% accuracy, whereas our deep learning model has a 93% accuracy and CEN of 0.053 on the same test set.

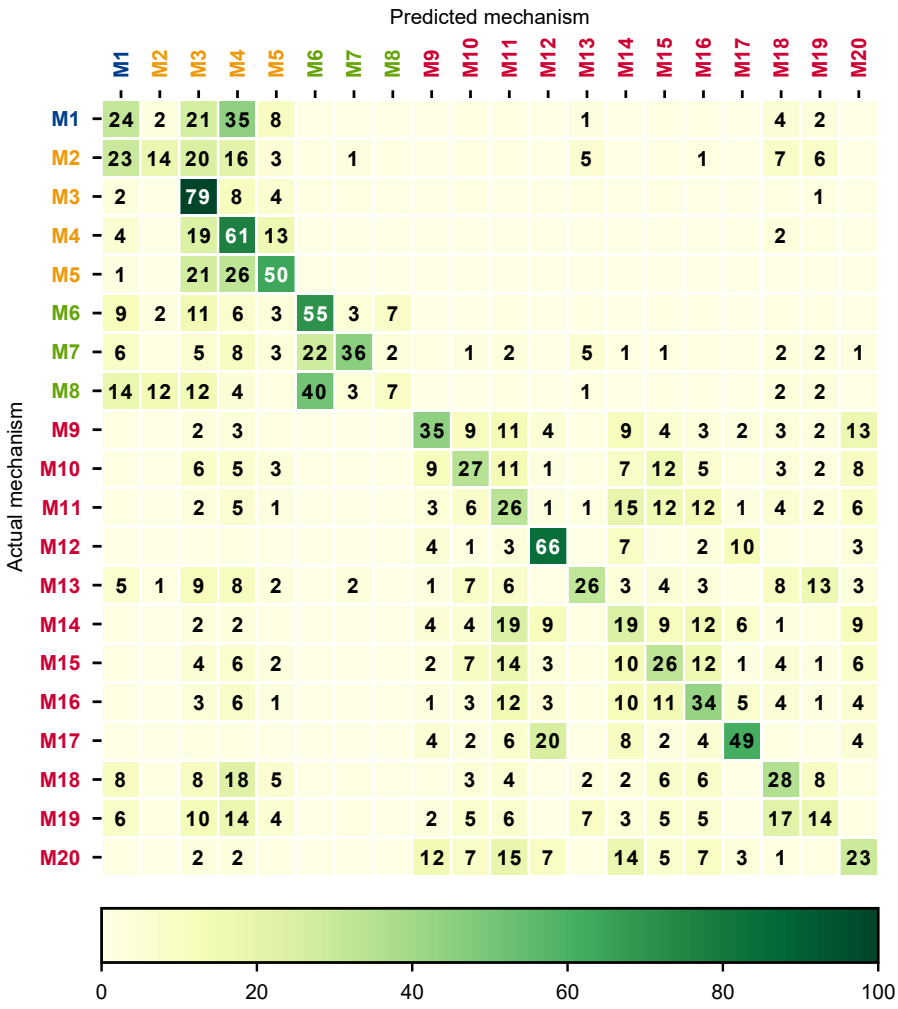


Fig. S10. Confusion matrix of top predicted mechanism versus actual mechanism (in percentage) with a 30-NN similarity search type model.

5.2. SVM

We applied SVM through the linearSVC implementation in the scikit-learn Python library to our dataset. Optimization of the C hyperparameter led to a best value of 7. The evaluation of this model on the test set made of 20 time points per sample (100,000 samples) resulted in a categorical accuracy of 48% and a CEN of 0.42. The resulting confusion matrix is shown in Fig. S11.

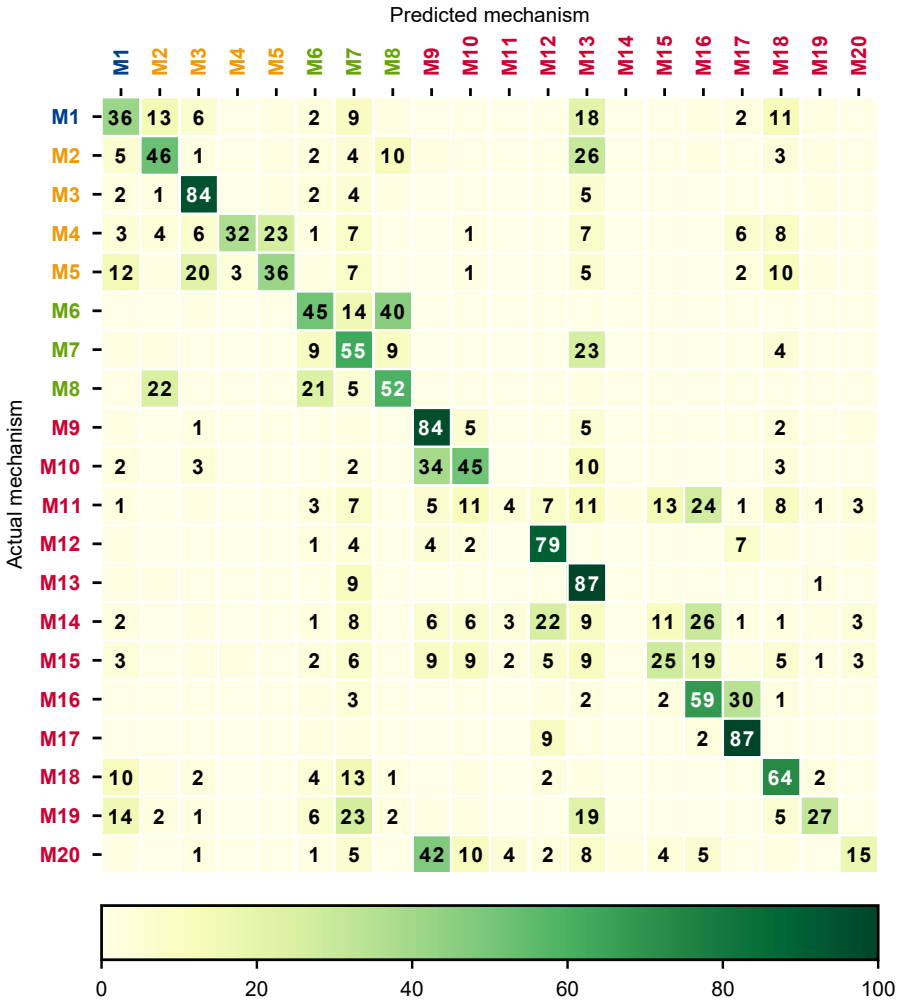


Fig. S11. Confusion matrix of top predicted mechanism versus actual mechanism (in percentage) with a SVM model.

5.3. Random Forest

A Random Forest model with 100 trees was trained (using RandomForestClassifier from scikit-learn Python library). The evaluation of this model on the test set made of 20 time points per sample (100,000 samples) resulted in a categorical accuracy of 54% and a CEN of 0.40. The resulting confusion matrix is shown in Fig. S12.

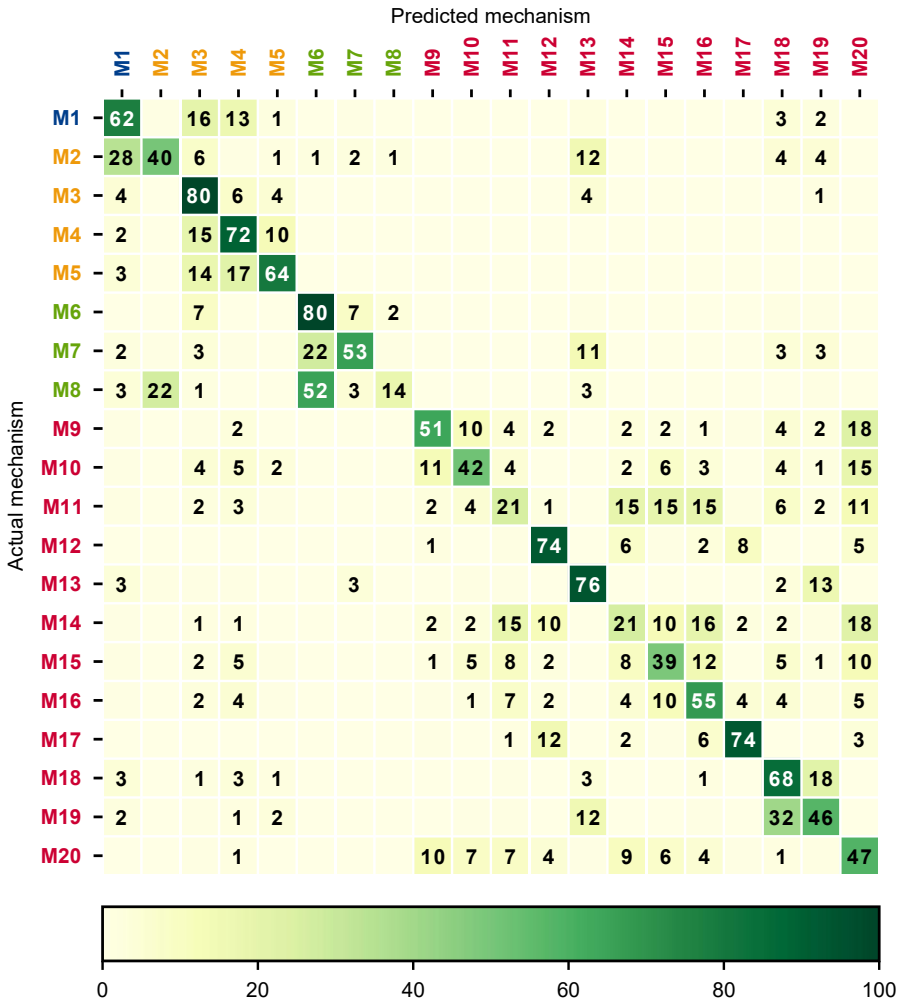


Fig. S12. Confusion matrix of top predicted mechanism versus actual mechanism (in percentage) with a Random Forest model.

6. Analysis of experimental kinetic data assisted by AI-models

6.1. General considerations

We are thankful to the authors of the case studies for sharing their numerical data with us.

In each case, of all the kinetic data available we selected the kinetic profiles with concentration of species within the range of the trained model.

We have slightly modified the original AI-model described in the main text to accommodate for the variety of experimental data available in the literature. The exact description of the different versions of AI-models is located in section S 6.2.

In this manuscript, we are focusing our attention on the kinetic data of experiments with different catalyst loading and same excess experiments (when available). The objective of the analysis of the case studies is to showcase the potential of the AI-models to identify kinetic features that are often outside the capabilities of classic kinetic analyses. For the exhaustive mechanistic study of each case, we refer the reader to the excellent original publications that often contain extra experiments such as KIE experiments, spectroscopic characterization of catalytic intermediates, stochiometric reactions involving catalytic intermediates and DFT calculations.

The AI-models we have used in the analysis of the case studies only consider the 20 mechanisms specified in the manuscript. These 20 mechanisms have less elementary steps than the mechanisms proposed by the authors or what might be reasonable to assume for some of the transformations. However, we consider them to be a good pool of mechanisms for this study because several elementary steps are often kinetically irrelevant and because it contains a variety of universal mechanistic features (such as catalyst dimerization, catalyst activation and catalyst deactivation) that the AI-models are able to identify.

6.2. Training of models with reduced input data

The main model of this study has been trained providing 4 kinetic runs per sample, containing both substrate and product concentration measurements, and including three runs at different catalyst concentration and a same-excess experiment with product added.[8-10]

However, in some cases it is not possible or convenient to measure the concentration of a substrate or product, or to carry out a same-excess experiment.[8-10] Our AI-based tool for kinetic analysis can still be applied, provided it is trained with the same data composition. In these cases, overall categorical accuracy for single-mechanism prediction is reduced, since the samples contain less meaningful

information. Importantly, when allowing for mechanism grouping, it generally still performs to similar overall accuracies as the main model by grouping more mechanisms together.

In the next subsections we describe a variety of models trained using a reduced input that will be used in the case studies.

6.2.1. Model training with reduced input data: only [P] and no same-excess experiment

This model was trained as described in section 4.2, but with using a training set with the following differences to the standard training set:

1) Kinetic runs were recorded using 0.1 to 1 mol% $[\mathbf{cat}]_0$, instead of 1 to 10 mol%.
2) Kinetic samples only contain $[\mathbf{P}]_t$ and not $[\mathbf{S}]_t$.
3) Each kinetic sample contains only the three different $[\mathbf{cat}]_0$ kinetics, and not the fourth same-excess run
4) Kinetic samples had 200 time points, instead of 20.

During training, data augmentation was carried out as described in section 4.2, except for the number of time points which were cut to the following values: [5, 6, 7, 8, 9, 10, 15, 20, 50, 100, 150, 200].

The model was trained for a total of 1384 epochs.

This model can be retrieved from the provided files as: M1_20_model_P_noXS.h5

Test set evaluation

This model was evaluated with a test set made of 20 time points per sample (100,000 samples) resulting in a categorical accuracy of 77.9% and a CEN of 0.20. The resulting confusion matrix is shown in Fig. S13.

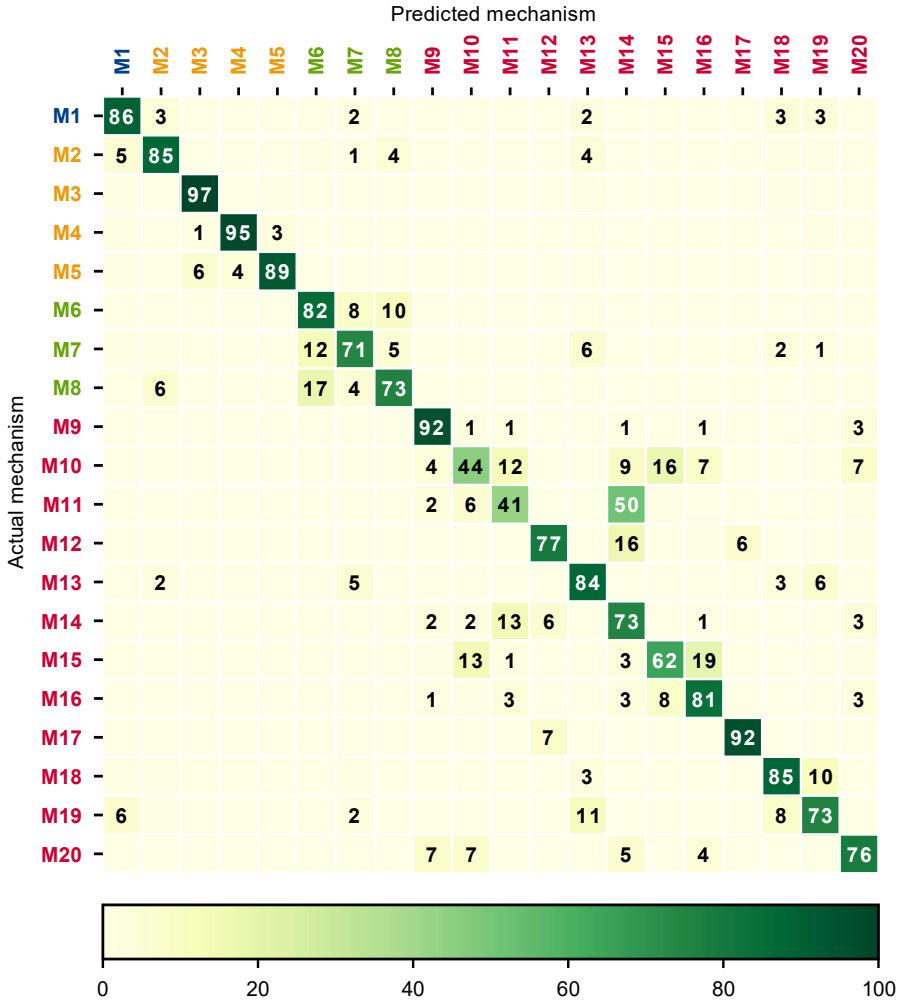


Fig. S13. Confusion matrix of top predicted mechanism versus actual mechanism (in percentage) with reduced model using only [P] and no same-excess experiment.

When allowing the model to group answers together precision increases to 99.8%, with the grouping distribution shown in Fig. S14.

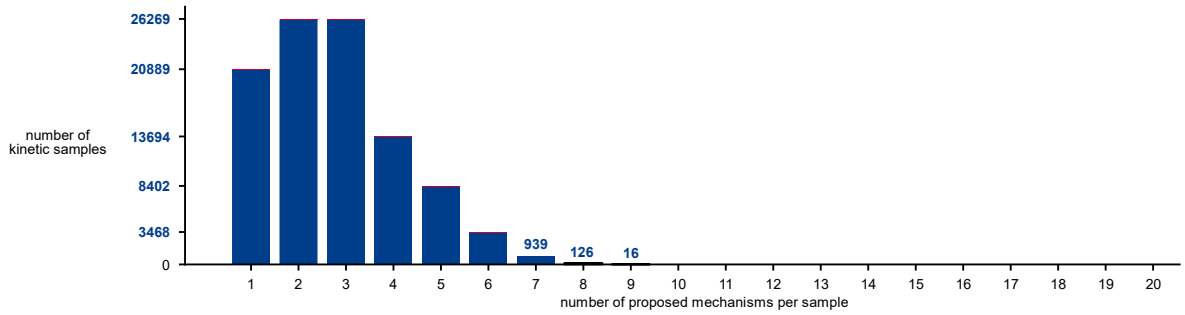


Fig. S14. Grouping distribution when using the reduced model using only [P] and no same-excess experiment.

6.2.2. Model training with reduced input data: only [S] and no same-excess experiment (0.1-5 mol%)

This model was trained as described in section 4.2, but with using a training set with the following differences to the standard training set:

1) Kinetic runs were recorded using 0.1 to 5 mol% $[\mathbf{cat}]_0$, instead of 1 to 10 mol%.
2) Kinetic samples only contain $[\mathbf{S}]_t$ and not $[\mathbf{P}]_t$.
3) Each kinetic sample contains only the three different $[\mathbf{cat}]_0$ kinetics, and not the fourth same excess run

During training, data augmentation was carried out as described in section 4.2.

The model was trained for a total of 1447 epochs.

This model can be retrieved from the provided files as: M1_20_model_S_noXS_01to5.h5

Test set evaluation

This model was evaluated with a test set made of 20 time points per sample (100,000 samples) resulting in a categorical accuracy of 80.7% and a CEN of 0.16. The resulting confusion matrix is shown in Fig. S15.

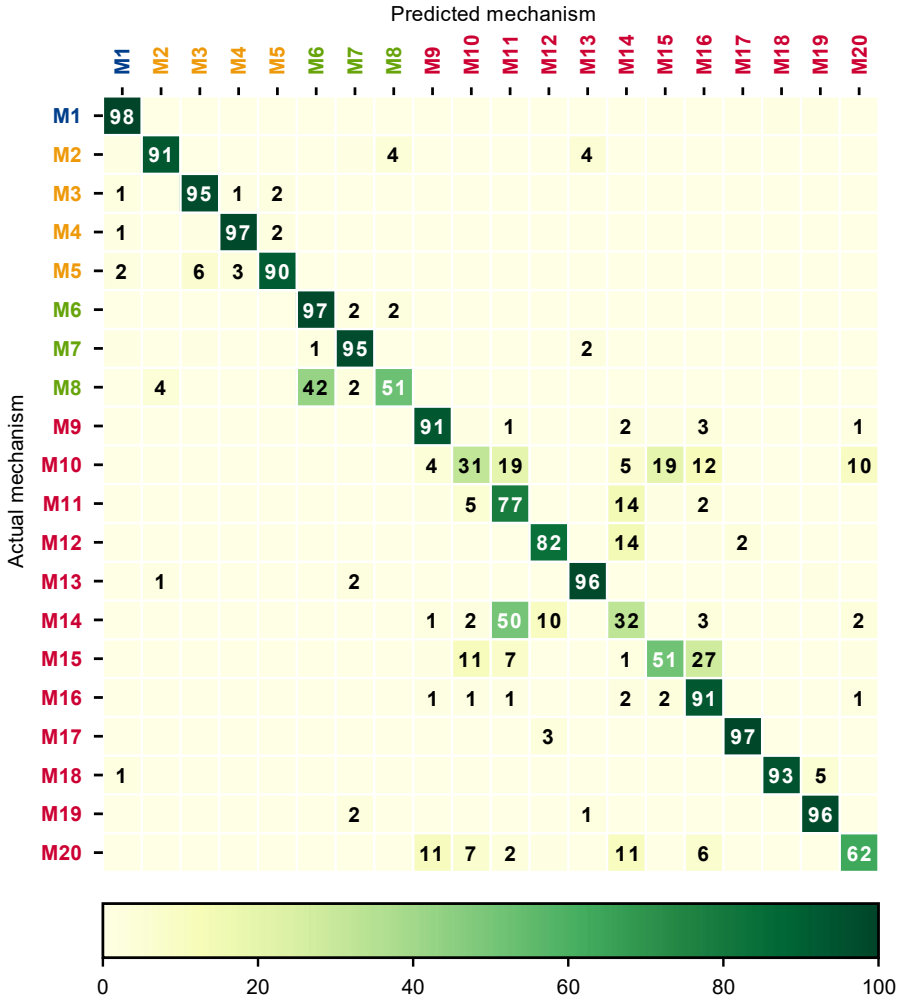


Fig. S15. Confusion matrix of top predicted mechanism versus actual mechanism (in percentage) with reduced model using only [S] and no same-excess experiment (0.1-5 mol%).

When allowing the model to group answers together precision increases to 99.9%, with the grouping distribution shown in Fig. S16.

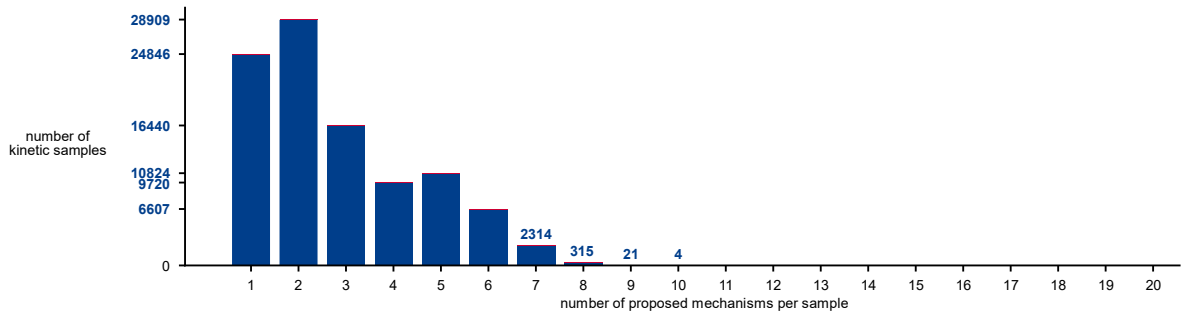


Fig. S16. Grouping distribution when using the reduced model using only [S] and no same-excess experiment (0.1-5 mol%).

6.2.3. Model training with reduced input data: only [S] and no same-excess experiment

This model was trained as described in section 4.2, but with using a training set with the following differences to the standard training set:

1) Kinetic samples only contain $[\mathbf{S}]_t$ and not $[\mathbf{P}]_t$.
2) Each kinetic sample contains only the three different $[\mathbf{cat}]_0$ kinetics, and not the fourth same excess run

During training, data augmentation was carried out as described in section 4.2.

The model was trained for a total of 1447 epochs.

This model can be retrieved from the provided files as: M1_20_model_S_noXS.h5

Test set evaluation

This model was evaluated with a test set made of 20 time points per sample (100,000 samples) resulting in a categorical accuracy of 83.9% and a CEN of 0.14. The resulting confusion matrix is shown in Fig. S17.

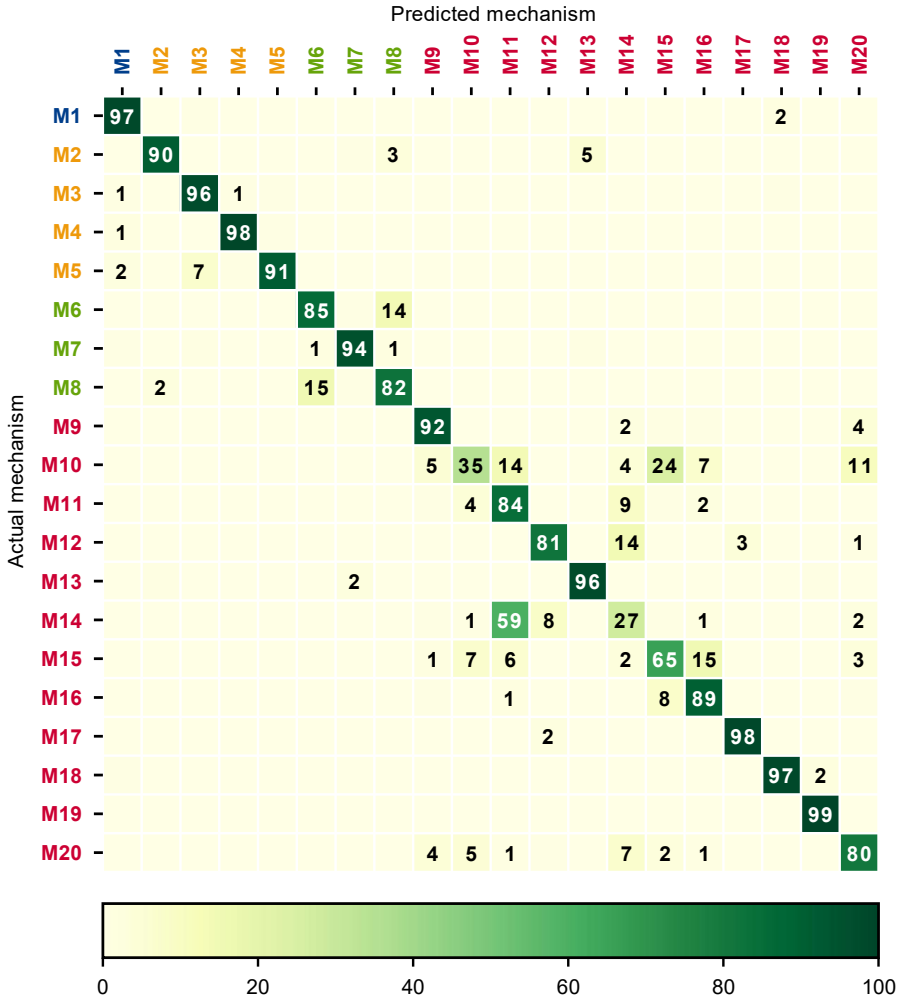


Fig. S17. Confusion matrix of top predicted mechanism versus actual mechanism (in percentage) with reduced model using only [S] and no same-excess experiment.

When allowing the model to group answers together precision increases to 99.9%, with the grouping distribution shown in Fig. S18.

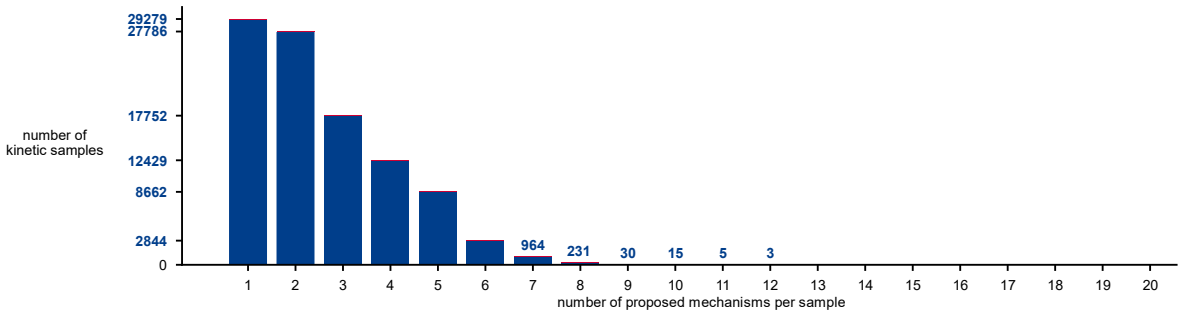


Fig. S18. Grouping distribution when using the reduced model using only [S] and no same-excess experiment.

6.2.4. Model training with reduced input data: only [P] and a no-product added same-excess experiment (0.1-5 mol%)

This model was trained as described in section 4.2, but with using a training set with the following differences to the standard training set:

1) Kinetic runs were recorded using 0.1 to 5 mol% $[\mathbf{cat}]_0$, instead of 1 to 10 mol%.
2) Kinetic samples contain accurate $[\mathbf{P}]_t$ whereas $[\mathbf{S}]_t$ is calculated as $[\mathbf{S}]_0 - [\mathbf{P}]_t$.
3) The fourth kinetic sample is a same excess run without product added. That is, a kinetic run with a reduced $[\mathbf{S}]_0$.

During training, data augmentation was carried out as described in section 4.2.

The model was trained for a total of 980 epochs.

This model can be retrieved from the provided files as: M1_20_model_P_noPXS_01to5.h5

Test set evaluation

This model was evaluated with a test set made of 20 time points per sample (100,000 samples) resulting in a categorical accuracy of 82.8% and a CEN of 0.15. The resulting confusion matrix is shown in Fig. S19.

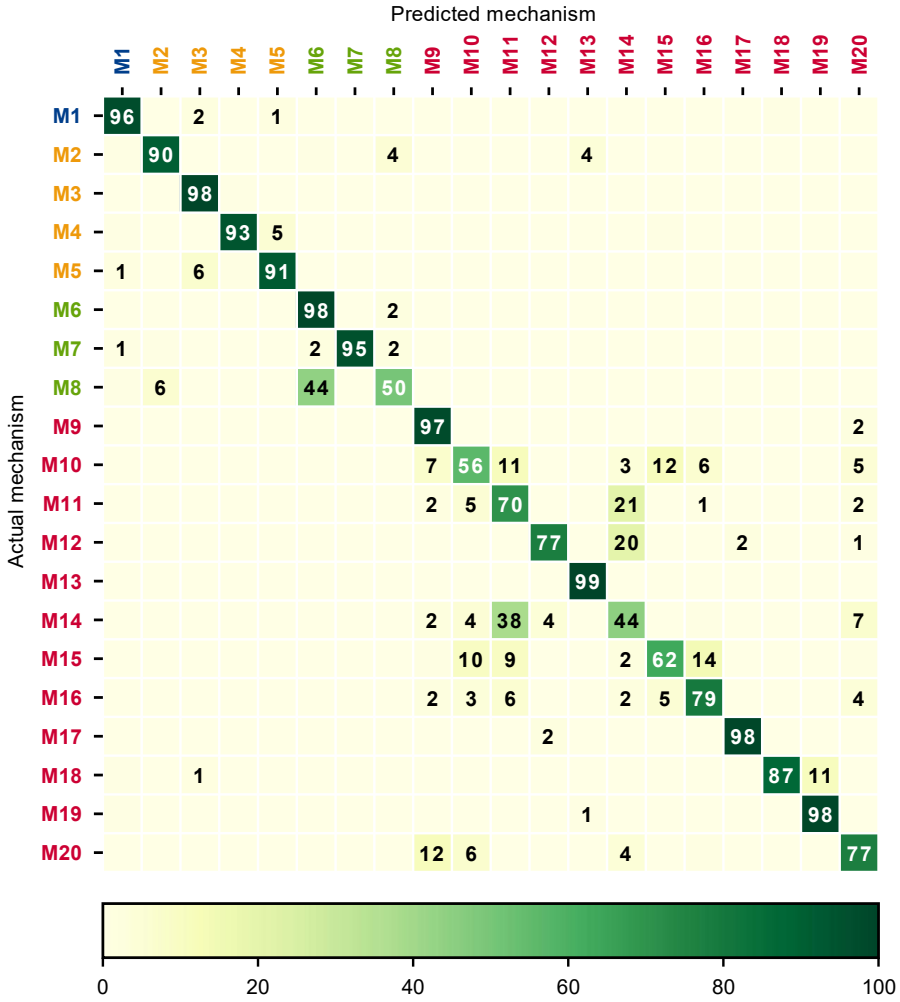


Fig. S19. Confusion matrix of top predicted mechanism versus actual mechanism (in percentage) with reduced model using only [P] and a no-product added same-excess experiment (0.1-5 mol%).

When allowing the model to group answers together precision increases to 99.8%, with the grouping distribution shown in Fig. S20.

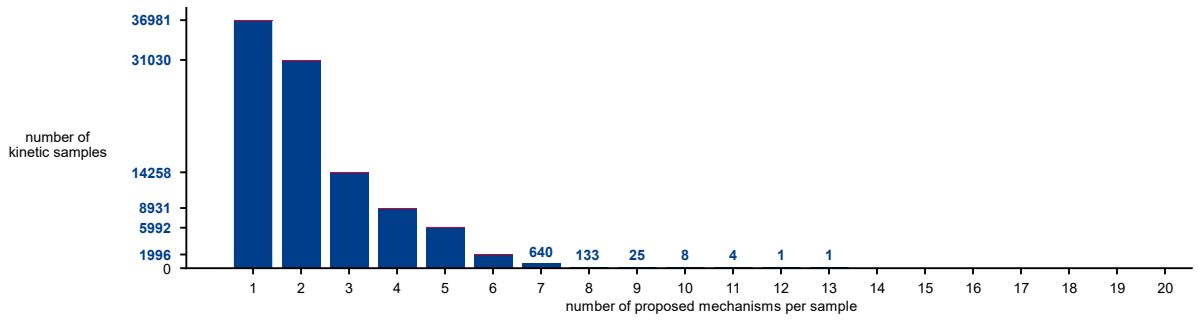


Fig. S20. Grouping distribution when using the reduced model using only [P] and a no-product added same-excess experiment (0.1-5 mol%).

6.2.5. Model training with reduced input data: only [P] and a no-product added same-excess experiment (1-10 mol%)

This model was trained as described in section 4.2, but with using a training set with the following differences to the standard training set:

1) Kinetic samples contain accurate $[\mathbf{P}]_t$ whereas $[\mathbf{S}]_t$ is calculated as $[\mathbf{S}]_0 - [\mathbf{P}]_t$.
2) The fourth kinetic sample is a same excess run without product added. That is, a kinetic run with a reduced $[\mathbf{S}]_0$.

During training, data augmentation was carried out as described in section 4.2.

The model was trained for a total of 980 epochs.

This model can be retrieved from the provided files as: M1_20_model_P_noPXS.h5

Test set evaluation

This model was evaluated with a test set made of 20 time points per sample (100,000 samples) resulting in a categorical accuracy of 81.9% and a CEN of 0.16. The resulting confusion matrix is shown in Fig. S21.

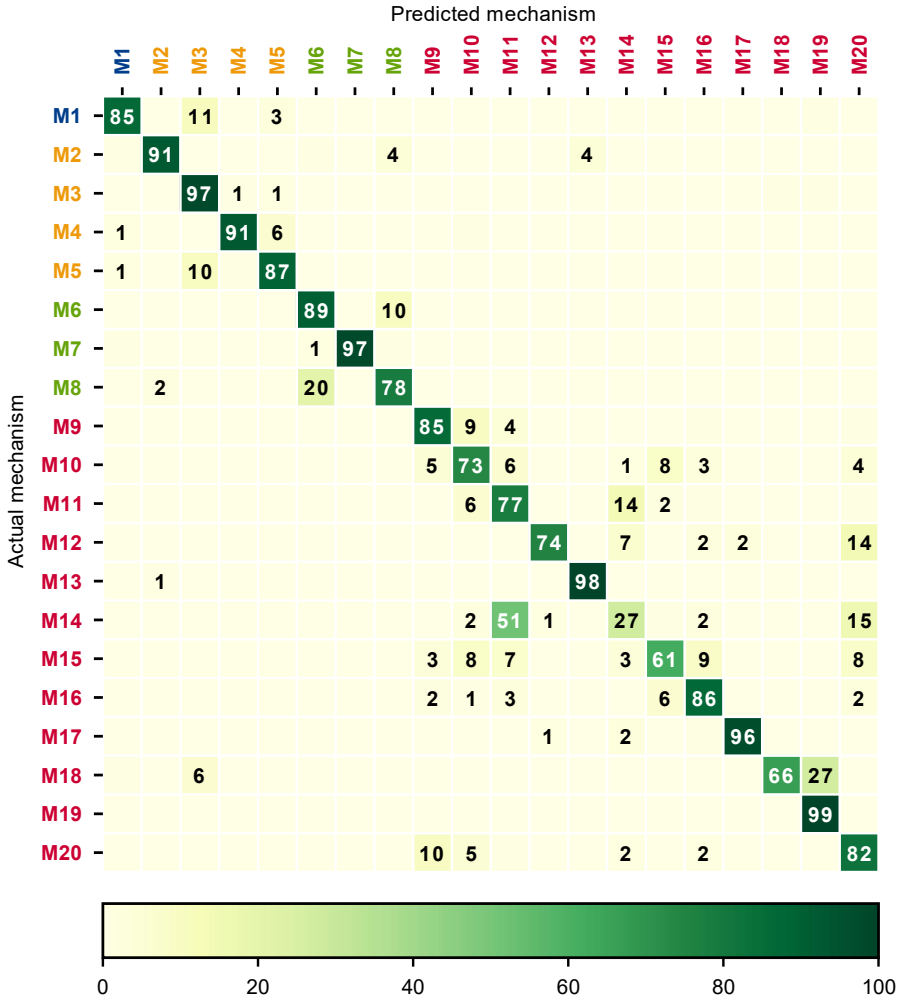


Fig. S21. Confusion matrix of top predicted mechanism versus actual mechanism (in percentage) with reduced model using only [P] and a no-product added same-excess experiment (1-10 mol%).

When allowing the model to group answers together precision increases to 99.5%, with the grouping distribution shown in Fig. S22.

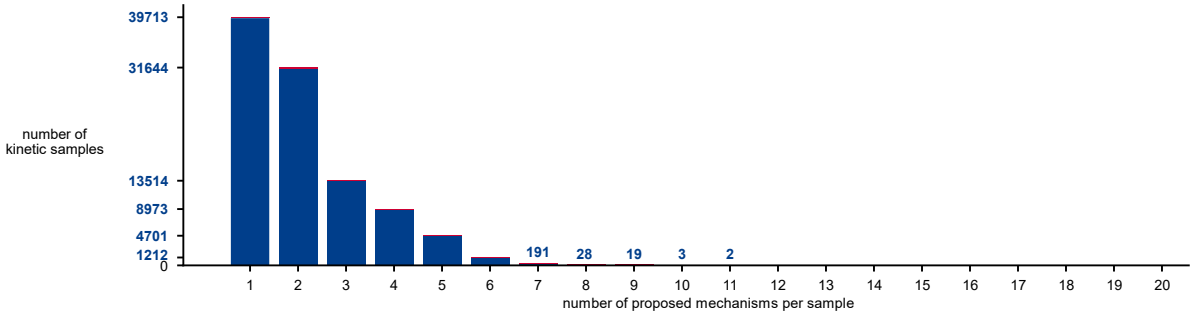


Fig. S22. Grouping distribution when using the reduced model using only [P] and a no-product added same-excess experiment (1-10 mol%).

6.2.6. Model training with reduced input data: only [S] and a no-product added same-excess experiment (0.1-5 mol%)

This model was trained as described in section 4.2, but with using a training set with the following differences to the standard training set:

1) Kinetic runs were recorded using 0.1 to 5 mol% $[\mathbf{cat}]_0$, instead of 1 to 10 mol%.
2) Kinetic samples contain only $[\mathbf{S}]_t$.
3) The fourth kinetic sample is a same excess run without product added. That is, a kinetic run with a reduced $[\mathbf{S}]_0$.

During training, data augmentation was carried out as described in section 4.2.

The model was trained for a total of 985 epochs.

This model can be retrieved from the provided files as: M1_20_model_S_noPXS_01to5.h5

Test set evaluation

This model was evaluated with a test set made of 20 time points per sample (100,000 samples) resulting in a categorical accuracy of 87.0% and a CEN of 0.11. The resulting confusion matrix is shown in Fig. S23.

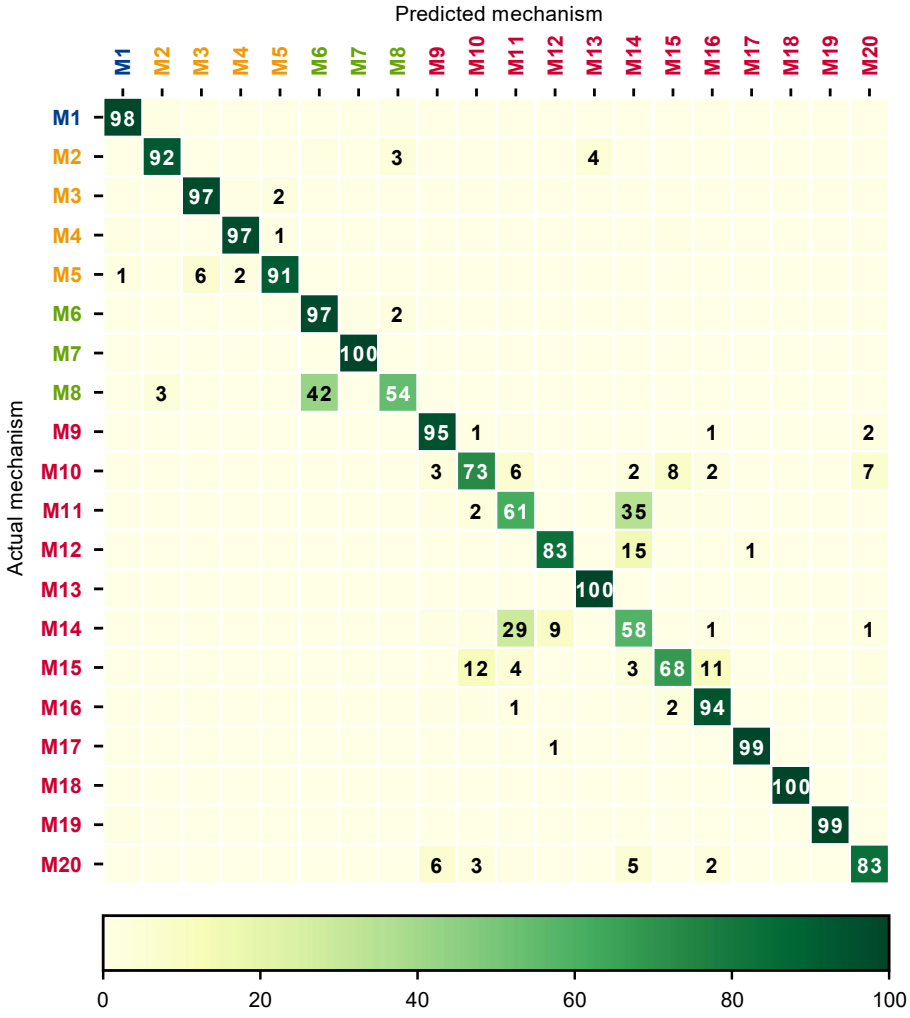


Fig. S23. Confusion matrix of top predicted mechanism versus actual mechanism (in percentage) with reduced model using only [S] and a no-product added same-excess experiment (0.1-5 mol%).

When allowing the model to group answers together precision increases to 99.9%, with the grouping distribution shown in Fig. S24.

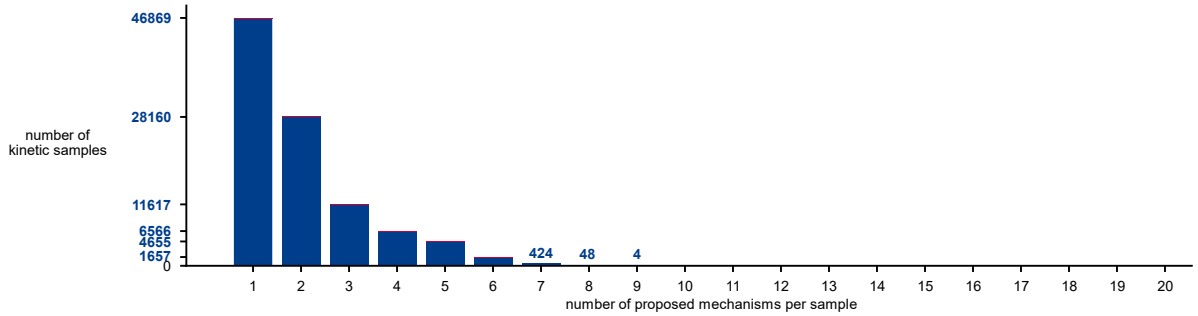
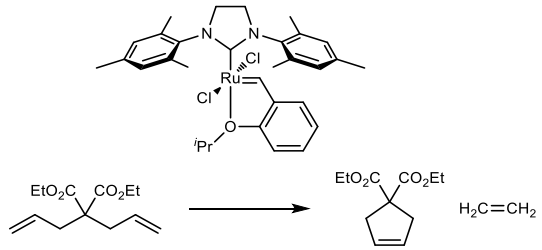


Fig. S24. Grouping distribution when using the reduced model using only [P] and a no-product added same-excess experiment (0.1-5 mol%).

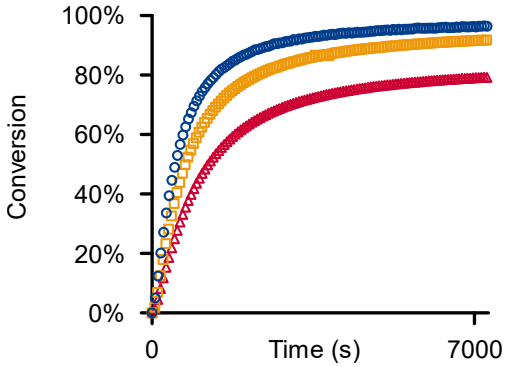
6.3. Case studies

6.3.1. Case Study 1. Thiel, V., Wannowius, K. J., Wolff, C., Thiele, C. M. and Herbert Plenio, H. Ring-Closing Metathesis Reactions: Interpretation of Conversion–Time Data. *Chem. Eur. J.* **19**, 16403 – 16414 (2013)

Reaction Scheme:



Data provided to the AI:



Red triangles: lowest catalyst loading (0.1 mol%); yellow squares: medium catalyst loading (0.2 mol%); blue circles: largest catalyst loading (0.3 mol%).

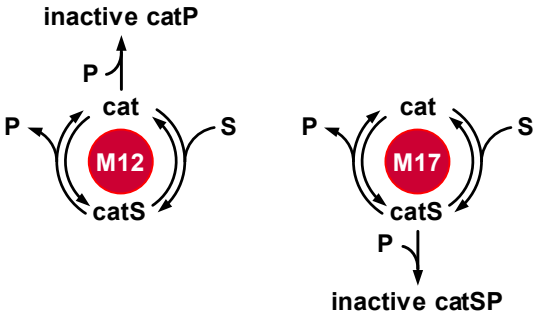
Data contained in Fig. 1 of the original publication. Numerical data provided by the authors.

AI-model used: only [P] and no same-excess experiment

File name: M1_20_model_P_noXS.h5;

Command to run prediction: python predict.py "Experiments\Kinetic_data_Case_study_1.txt" --S0 100 --model Data\M1_20_model_P_noXS.h5

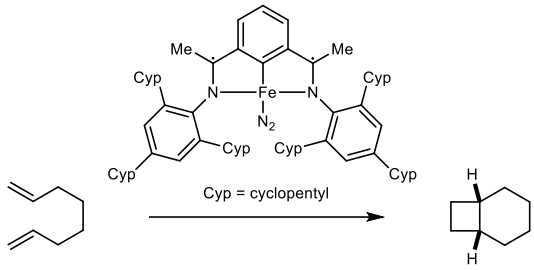
AI-model outcome:



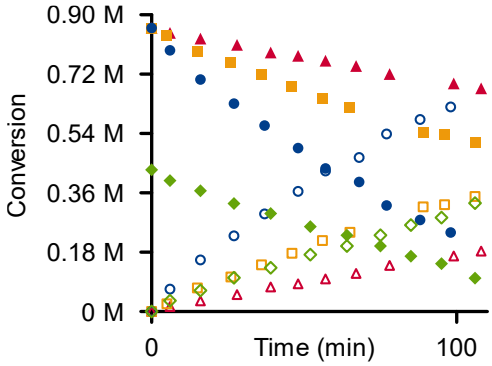
Notes: Using classic kinetic analyses, the authors were able to identify catalyst deactivation, but could not pinpoint the exact deactivation pathway. Our AI-model predicts M12 and M17 as closest mechanistic representations of the kinetic data. Thus, not only the model identified the presence of catalyst decomposition but also proposed that the main decomposition under the reaction conditions is mediated by the product of the reaction. This prediction agrees with previous work carried out on stoichiometric Ru-complexes and DFT calculations, which indeed showed the possibility of the ethene product inducing catalyst decomposition.[11,12]

6.3.2. Case Study 2. Joannou, M. V., Hoyt, J. M. and Chirik, P. J. Investigations into the Mechanism of Inter- and Intramolecular Iron-Catalyzed [2 + 2] Cycloaddition of Alkenes. *J. Am. Chem. Soc.* **142**, 5314−5330 (2020).

Reaction Scheme:



Data provided to the AI:

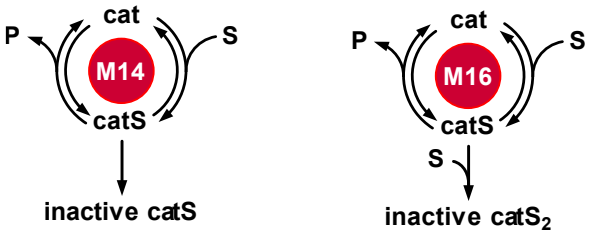


Filled symbols: substrate concentration; hollow symbols: product concentration. Red triangles: lowest catalyst loading (2.5 mol%); yellow squares: medium catalyst loading (5 mol%); blue circles: largest catalyst loading (10 mol%); green diamonds (10 mol% of 0.43 M or 5 mol% of 0.86 M).

Product data contained in Fig. 9B left and Fig 9C left of the original publication. Numerical data provided by the authors. Substrate concentration data is the calculated mass balance.

AI-model used: only [P] and a no-product added same-excess experiment (1-10 mol%)
File name: M1_20_model_P_noPXS.h5
Command to run prediction: python predict.py "Experiments\Kinetic_data_Case_study_2.txt" --model Data\M1_20_model_P_noPXS.h5

AI-model outcome:

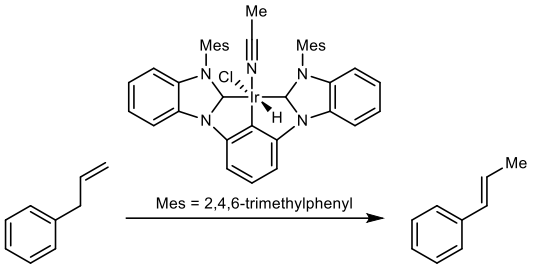


Notes: The authors of the original publication could not detect the deactivation of the catalyst in the kinetics of this reaction due to the intrinsic limitations of standard kinetic analysis. However, they noticed the deactivation in a similar reaction, intermolecular instead of intramolecular, run with the same catalyst. Also, the authors found through additional experiments, such as stoichiometric organometallic studies, that the deactivation arise from the dehydrogenation of a cyclopentyl group in the ligand of the catalyst by reaction with the substrate. This observation is perfectly aligned with

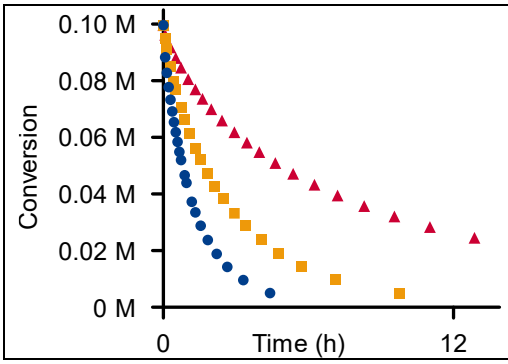
the mechanisms proposed by the AI-model, which involves one molecule of substrate in the deactivation processes.

6.3.3. Case Study 3. Melody et al. Mechanistic Studies of Alkene Isomerization Catalyzed by CCC-Pincer Complexes of Iridium. *Organometallics* **33**, 473−484 (2014).

Reaction Scheme:



Data provided to the AI:

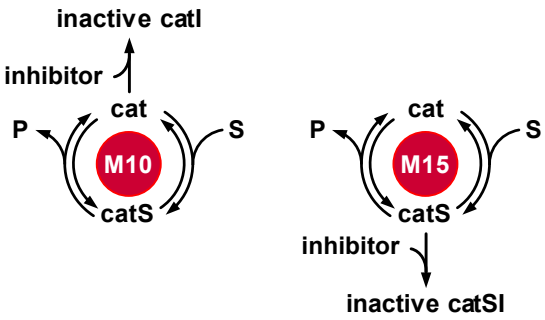


Red triangles: lowest catalyst loading (0.25 mol%); yellow squares: medium catalyst loading (0.5 mol%); blue circles: largest catalyst loading (1 mol%).

Data contained in Fig. 8 of the original publication. Numerical data provided by the authors.

AI-model used: only [S] and no same-excess experiment (0.1-5 mol%)
File name: M1_20_model_S_noXS_01to5.h5
Command to run prediction:
python predict.py "Experiments\Kinetic_data_Case_study_3.txt" --model Data\M1_20_model_S_noXS_01to5.h5

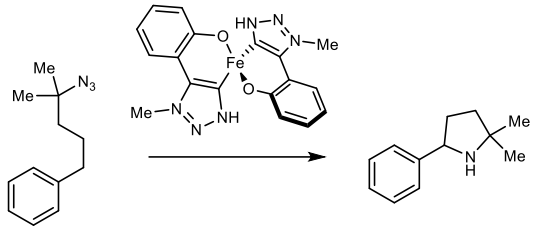
AI-model outcome:



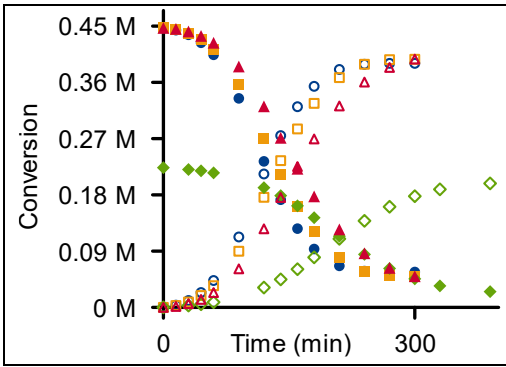
Notes: The AI-model has identified a deactivation of the catalyst with an unknown inhibitor. In the original publication, there is no mention of this possibility as it is very challenging to detect by standard kinetic analysis.

6.3.4. Case Study 4. Stroek, W., Keilwerth, M., Pividori, D. M., Meyer, K., and Albrecht M. An Iron−Mesoionic Carbene Complex for Catalytic Intramolecular C−H Amination Utilizing Organic Azides. *J. Am. Chem. Soc.* **143**, 20157−20165 (2021).

Reaction Scheme:

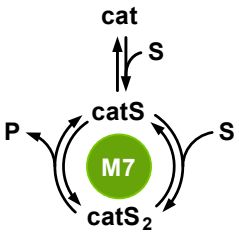


Data provided to the AI:

	Filled symbols: substrate concentration; hollow symbols: product concentration. Red triangles: lowest catalyst loading (1 mol%); yellow squares: medium catalyst loading (1.5 mol%); blue circles: largest catalyst loading (2 mol%); green diamonds (2 mol% of 0.2236 M or 1 mol% of 0.4472 M). Data contained in Fig. S8 and Fig. S11 of the original publication.

AI-model used: only [P] and a no-product added same-excess experiment (0.1-5 mol%)
File name: M1_20_model_P_noPXS_01to5.h5
Command to run prediction:
python predict.py "Experiments\Kinetic_data_Case_study_4.txt" --model Data\M1_20_model_P_noPXS_01to5.h5

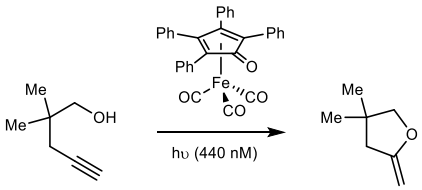
AI-model outcome:



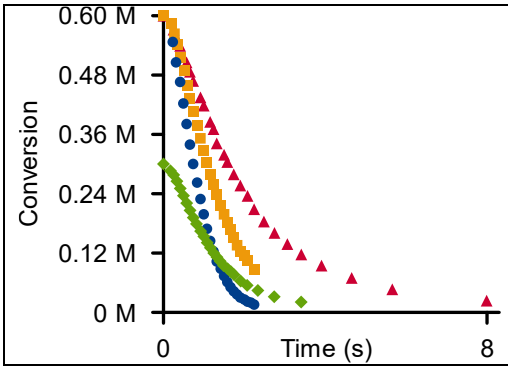
Notes: The AI-model proposed exclusively the mechanism with an activation of the catalyst by the substrate. This relatively unusual type of catalyst activation was also proposed by the authors of this study based on a thorough analysis of 10 kinetic runs varying [S] and [cat]. Remarkably, the model was able to reach this conclusion from just four runs, highlighting the advantages for AI-assisted mechanistic elucidation.

6.3.5. Case Study 5. Lehnherr, D. et al. Discovery of a Photoinduced Dark Catalytic Cycle Using in Situ LED-NMR Spectroscopy. *J. Am. Chem. Soc.* **140**, 13843−13853 (2018).

Reaction Scheme:

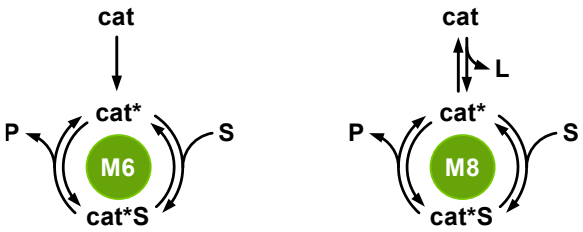


Data provided to the AI:

	Red triangles: lowest catalyst loading (0.375 mol%); yellow squares: medium catalyst loading (1.5 mol%); blue circles: largest catalyst loading (3 mol%); green diamonds (0.75 mol% of 0.3 M or 0.375 mol% of 0.6 M). The data was extracted graphically from Fig. S175-right and Fig. S177-left in the original manuscript, because the authors refused to provide us with the numerical data.

AI-model used: only [S] and a no-product added same-excess experiment (0.1-5 mol%)
File name: Data\M1_20_model_S_noPXS_01to5.h5
Command to run prediction:
python predict.py "Experiments\Kinetic_data_Case_study_5.txt" --model Data\M1_20_model_S_noPXS_01to5.h5

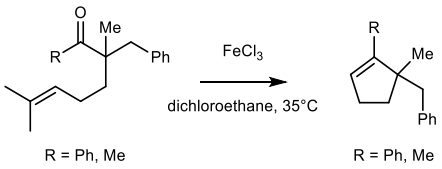
AI-model outcome:



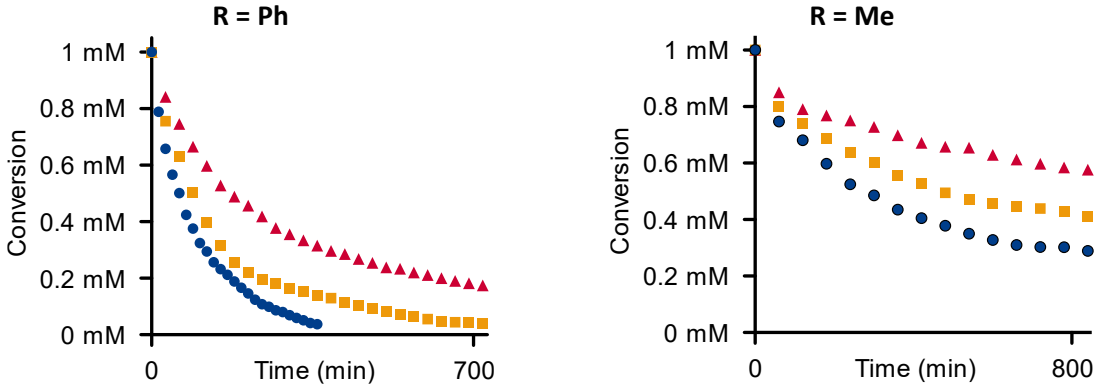
Notes: The AI-model identified that the catalyst requires a pre-activation step, either direct or by releasing a ligand. This is indeed consistent with the authors proposed mechanism, where light activation is proposed to reversibly dissociate a CO ligand in the catalyst.

6.3.6. Case Study 6. Albright, H. et al. Catalytic Carbonyl-Olefin Metathesis of Aliphatic Ketones: Iron(III) Homo-Dimers as Lewis Acidic Superelectrophiles. *J. Am. Chem. Soc.* **141**, 1690−1700 (2019).

Reaction Scheme:



Data provided to the AI:



R = Ph: Red triangles: lowest catalyst loading (5 mol%); yellow squares: medium catalyst loading (7.5 mol%); blue circles: largest catalyst loading (10 mol%). **R = Me**: Red triangles: lowest catalyst loading (5 mol%); yellow squares: medium catalyst loading (7.5 mol%); blue circles: largest catalyst loading (10 mol%). Numerical data was provided by the authors within the original paper.

AI-model used: only [S] and no same-excess experiment

File name: M1_20_model_S_noXS.h5

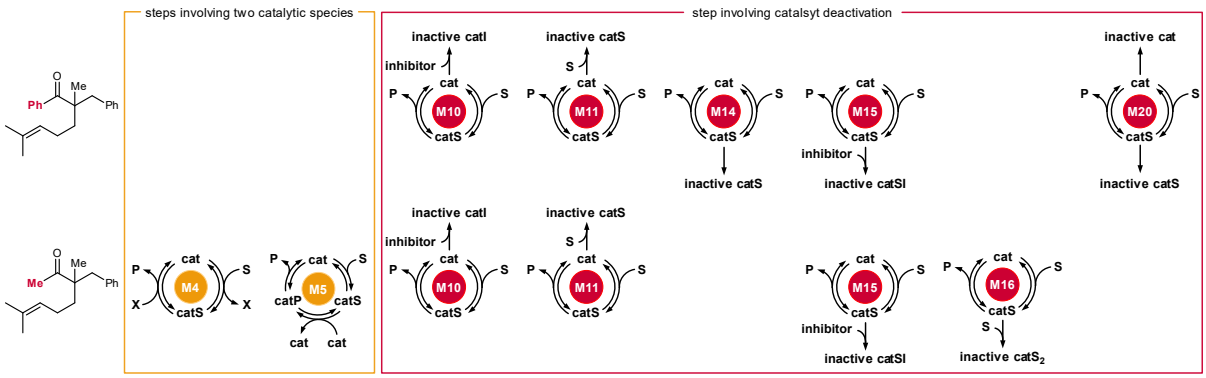
Commands to run prediction for R = Ph:

python predict.py "Experiments\Kinetic_data_Case_study_6_Ph.txt" --model Data\M1_20_model_S_noXS.h5

Commands to run prediction for R = Me:

python predict.py "Experiments\Kinetic_data_Case_study_6_Me.txt" --model Data\M1_20_model_S_noXS.h5

AI-model outcome:



Notes:

Substrate with Ph: A variety of deactivation pathways are proposed by the AI-model, including the presence of an inhibitor, deactivation by substrate and direct deactivation of the catalyst. In the original publication, there is no mention of the possibility of catalyst deactivation as it is very challenging to detect by standard kinetic analysis. The use of the AI-model in this case highlights the need for additional kinetic experiments, such as a same-excess experiment, which would narrow down the origin of decomposition, leading to hypotheses that could be tested via other experiments, such as stoichiometric organometallic studies.

Substrate with Me: In addition to a variety of deactivation pathways as for R = Ph, two mechanisms with a step involving two catalytic species reacting together are proposed. For this substrate (R = Me), the authors proposed a type-M5 mechanism based in a second-order rate dependence on catalyst. The result of the AI-model in this case would assist the authors to design additional kinetic experiments to rule out (or confirm) the deactivation of the catalyst, which was also suggested by the AI-model for the substrate with R = Ph.

7. Instructions to install and run the provided python files

7.1. Setting up working environment

1) Create a working folder with any name

2) Download AI_model_and_files.zip from (https://doi.org/10.48420/16965271) and unpack it in the working folder

3) If you are going to be retraining the model, download and unpack trainvaltest.zip (https://doi.org/10.48420/16965292) and save the unpacked files into the Data folder.

4) Install Anaconda (https://www.anaconda.com/products/individual)

5) Open Anaconda Prompt

6) Type: conda env create -f environment.yml

7.2. Prediction with own data on full model ([S] and [P], 1-10 mol% $[cat]_0$)

1) In Anaconda Prompt, activate the environment by typing: conda activate kinetics

2) Input your data on template 'kinetics.xlsx' within the following rules:

- Time for all four runs must be in the same units (eg minutes)
- Concentration values for **S**, **P** and $\mathbf{cat_T}$ must be all on the same units (eg M)

- First three kinetic runs must be ordered from smallest to highest **[cat]$_0$**
- Fourth kinetic run must have same [**cat**]$_0$ as one of the first three
- Fourth kinetic run must have less [**S**] than the first three, with an amount of added product equal to the difference
- All [**cat**]$_0$ should be between 1 and 10 mol%
- All runs must have the same number of time points

3) Save file as a txt with TAB delimitations in the Data folder

4) In the working folder type: python predict.py *path/filename_with_data (for example: python predict.py Data/kinetics.txt)*

7.3. Prediction with own data on other models

1) In Anaconda Prompt, activate the environment by typing: conda activate kinetics

2) Input your data on a suitable template from the Data folder within the requirements for the model to be used. Pay attention to:

- Time for all kinetic runs must be in the same units (eg minutes)
- Concentration values for **S**, **P** and **cat$_T$** (when present) must be all on the same units (eg M)
- The kinetic runs must be ordered from smallest to highest [**cat**]$_0$, except when a same excess experiment is present, which always has to be fourth.
- All [**cat**]$_0$ should be between in the correct range for the trained model

3) Save file as a txt with TAB delimitations in the Data folder

4) In the working folder type:

python predict.py *path/filename_with_data --model path/model_file*

eg: python predict.py Data/kinetics_S_only.txt --model Data\M1_20_model_S_noXS.h5

5) If using a P_only model, the initial concentration of substrate must be provided with the command --S0, in the same units as the other concentrations in the data file. For example:

python predict.py Data/kinetics_P_only.txt --S0 1.0 --model Data\M1_20_model_P_noXS.h5

7.4. Training the model

1) In Anaconda Prompt, activate the environment by typing: conda activate kinetics

2) In the working folder type: python train.py

3) If necessary, edit train.py to change the number of time points, error range, epochs and batch_size, as required

7.5. Provided Files

M1_20_model.h5	Full model and trained weights (with [S], [P], and product added same excess, for cat loadings between 1 and 10 mol%)
M1_20_model_P_noXS.h5	Model and trained weights (with only [P] and without same excess, for cat loadings between 0.1 and 1 mol%)
M1_20_model_S_noXS_01to5.h5	Model and trained weights (with only [S] and without same excess, for cat loadings between 0.1 and 5 mol%)
M1_20_model_S_noXS.h5	Model and trained weights (with only [S] and without same excess, for cat loadings between 1 and 10 mol%)
M1_20_model_P_noPXS_01to5.h5	Model and trained weights (with only [P], mass balance of [S] and with a no-product-added same excess, for cat loadings between 0.1 and 5 mol%)
M1_20_model_P_noPXS.h5	Model and trained weights (with only [P], mass balance of [S] and with a no-product-added same excess, for cat loadings between 1 and 10 mol%)
M1_20_model_S_noPXS.h5	Model and trained weights (with only [S] and with a no-product-added same excess, for cat loadings between 1 and 10 mol%)
kinetics.xlsx kinetics_P_only.xlsx	Templates for inputting kinetic data for predictions
kinetics.txt kinetics_P_only.txt	Examples of raw input file for predictions
predict.py	Script that takes data_file and returns a prediction using the full model (by default Data/kinetics.txt)
predict_reduced.py	Script that takes data_file and returns a prediction using the reduced model (only [P] and without same-excess run) (by default Data/kinetics_P_only.txt)
train.py	Script that trains the model using a provided training set
utils.py	Functions and classes used by the scripts “predict.py” and “train.py”
TrainValTest.zip	Raw data used for training, validation and testing
environment.yml	Recommended Anaconda environment for running provided files (requires GPU)
Kinetic_data_Case_study_1.txt Kinetic_data_Case_study_2.txt Kinetic_data_Case_study_3.txt Kinetic_data_Case_study_4.txt Kinetic_data_Case_study_5.txt Kinetic_data_Case_study_6_Ph.txt Kinetic_data_Case_study_6_Me.txt	Files with input data provided to the AI for the case studies 1 to 6

Supplementary Table 3. List and description of the provided files.

8. Supplementary Tables

Data for Figure 3A in the manuscript:

actual mechanism \ predicted mechanism	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20
M1	**4989**	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M2	1	**4734**	0	0	0	7	1	63	0	0	0	0	194	0	0	0	0	0	0	0
M3	0	0	**4990**	1	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M4	25	0	0	**4956**	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M5	108	0	247	23	**4620**	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0
M6	0	3	0	0	0	**4196**	6	795	0	0	0	0	0	0	0	0	0	0	0	0
M7	0	0	0	0	0	1	**4998**	1	0	0	0	0	0	0	0	0	0	0	0	0
M8	0	41	0	0	0	917	1	**4041**	0	0	0	0	0	0	0	0	0	0	0	0
M9	0	0	0	0	0	0	0	0	**4973**	27	0	0	0	0	0	0	0	0	0	0
M10	0	0	10	0	0	0	0	0	150	**4812**	0	0	0	0	5	0	0	0	0	23
M11	0	0	0	0	0	0	0	0	0	0	**3708**	0	0	1252	40	0	0	0	0	0
M12	0	0	0	0	0	0	0	0	0	0	0	**4591**	0	396	13	0	0	0	0	0
M13	0	61	0	0	0	0	2	0	0	0	0	0	**4937**	0	0	0	0	0	0	0
M14	0	0	0	0	0	0	0	0	0	0	2011	506	0	**2438**	45	0	0	0	0	0
M15	0	0	0	0	0	0	0	0	0	0	158	35	0	113	**4690**	4	0	0	0	0
M16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**4999**	1	0	0	0
M17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**5000**	0	0	0
M18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**5000**	0	0
M19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	**4996**	0
M20	0	0	0	0	0	0	0	0	12	1	4	0	0	12	0	0	0	0	0	**4971**

Supplementary Table 4. Confusion matrix with 6 time point test set. Absolute values.

Data for Figure 3A in the manuscript:

actual mechanism \ predicted mechanism	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20
M1	**99.8**	0	0	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M2	0	**94.7**	0	0	0	0.1	0	1.3	0	0	0	0	3.9	0	0	0	0	0	0	0
M3	0	0	**99.8**	0	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M4	0.5	0	0	**99.1**	0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M5	2.2	0	4.9	0.5	**92.4**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M6	0	0.1	0	0	0	**83.9**	0.1	15.9	0	0	0	0	0	0	0	0	0	0	0	0
M7	0	0	0	0	0	0	**100**	0	0	0	0	0	0	0	0	0	0	0	0	0
M8	0	0.8	0	0	0	18.3	0	**80.8**	0	0	0	0	0	0	0	0	0	0	0	0
M9	0	0	0	0	0	0	0	0	**99.5**	0.5	0	0	0	0	0	0	0	0	0	0
M10	0	0	0.2	0	0	0	0	0	3	**96.2**	0	0	0	0	0.1	0	0	0	0	0.5
M11	0	0	0	0	0	0	0	0	0	0	**74.2**	0	0	25	0.8	0	0	0	0	0
M12	0	0	0	0	0	0	0	0	0	0	0	**91.8**	0	7.9	0.3	0	0	0	0	0
M13	0	1.2	0	0	0	0	0	0	0	0	0	0	**98.7**	0	0	0	0	0	0	0
M14	0	0	0	0	0	0	0	0	0	0	40.2	10.1	0	48.8	0.9	0	0	0	0	0
M15	0	0	0	0	0	0	0	0	0	0	3.2	0.7	0	2.3	**93.8**	0.1	0	0	0	0
M16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**100**	0	0	0	0
M17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**100**	0	0	0
M18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**100**	0	0
M19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	**99.9**	0
M20	0	0	0	0	0	0	0	0	0.2	0	0.1	0	0	0.2	0	0	0	0	0	**99.4**

Supplementary Table 5. Confusion matrix with 6 time point test set. Percentage values.

Data for Figure 3C in the manuscript:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Correct	69708	23761	6067	405	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wrong	32	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Supplementary Table 6. Number of grouped predicted mechanisms and whether the correct mechanism is contained within the group.

Data for Figure 3D in the manuscript (circos plot):

Each value between different mechanisms represents the number of times 2 mechanisms have been grouped together in the predictions with a 99% confidence threshold. Only connections worth >1% of the total are displayed in the plot of the Figure 3D in the manuscript (ie absolute value of connections >168).

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20
M1		25	92	**174**	**2240**	0	20	0	0	0	0	0	0	0	1	0	0	0	0	0
M2			2	0	1	111	101	**1176**	0	0	0	0	**3116**	0	0	0	0	0	14	0
M3				**187**	**1829**	0	0	0	0	13	0	0	0	0	12	0	0	0	0	0
M4					**691**	0	0	0	0	1	0	0	0	0	8	0	0	0	0	0
M5						0	0	0	0	37	0	0	0	0	49	0	0	0	0	0
M6							**185**	**6588**	0	0	0	0	0	0	0	0	0	0	0	0
M7								122	0	0	0	0	8	0	0	0	0	1	0	0
M8									0	0	0	0	22	0	0	0	0	0	0	0
M9										**2309**	13	1	0	17	8	0	0	0	0	**723**
M10											21	0	0	10	67	0	0	0	0	**702**
M11												9	0	**7886**	**4049**	89	0	0	0	**279**
M12													0	**2680**	**1757**	5	1	0	0	**251**
M13														0	0	0	0	0	14	0
M14															**5509**	81	0	0	0	**653**
M15																123	0	0	0	**375**
M16																	39	0	0	7
M17																		0	0	0
M18																			104	0
M19																				0
M20																				

Supplementary Table 7. Number of times that each pair of mechanisms are proposed together by the model using 6 time points.

Data for Figure 4B in the manuscript in the manuscript (top left):

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Correct	69708	23761	6067	405	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wrong	32	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Supplementary Table 8. Prediction accuracy for the test set containing 6 time points and different Gaussian error.

Data for Figure 4B in the manuscript (top right):

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
σ = 0	69740	23767	6067	405	21	0	0	0	0	0	0	0	0	0	0	0	0
σ = 1	43643	30844	10360	7631	4691	2057	624	121	26	2	1	0	0	0	0	0	0
σ = 5	22686	31530	17962	10474	7327	4829	2465	1375	658	344	197	95	34	20	1	3	0

Supplementary Table 9. Number of grouped predicted mechanisms (correct and incorrect predictions are grouped together), for the test set containing 6 time points and different Gaussian error.

Data for Figure 4B in the manuscript (bottom left):

Time points in data	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**15**	**20**
Accuracy	0.6053	0.9941	0.9983	0.9983	0.9985	0.9986	0.9984	0.9986	0.9984	0.9983	0.9983	0.9983

Supplementary Table 10. Prediction accuracy for test sets with varying number of time points (in addition to time zero point), with data containing Gaussian error with a σ = 1%.

Data for Figure 4B in the manuscript (bottom right):

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2 time points	14660	31011	17599	12991	9432	7200	4096	1815	736	267	117	50	15	8	3	0	0
6 time points	43643	30844	10360	7631	4691	2057	624	121	26	2	1	0	0	0	0	0	0
20 time points	58483	25463	8802	5295	1514	371	60	12	0	0	0	0	0	0	0	0	0

Supplementary Table 11. Number of grouped predicted mechanisms (correct and incorrect predictions are grouped together), for the test sets made of 2, 6 and 20 time points, and with data containing Gaussian error with a σ = 1%.

Data for Fig. S8 in the manuscript:

Circos plot for grouping of mechanisms in the predictions of the test set with 6 time points points and Gaussian error with σ = 1%. Each value between different mechanisms represents the number of times 2 mechanisms have been grouped together in the predictions. Only connections worth >1% of the total are displayed in Fig. S8 (ie absolute value of connections >595).

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20
M1		463	**612**	531	**3266**	9	204	1	13	33	39	11	41	30	44	38	8	49	64	10
M2			31	45	120	296	535	**2584**	0	0	3	0	**3971**	1	0	1	1	125	**733**	0
M3				**1224**	**3598**	0	21	0	9	250	40	0	5	27	212	47	2	1	4	11
M4					**2807**	0	0	0	18	185	73	1	0	59	245	74	5	0	0	28
M5						0	45	0	32	470	103	5	3	69	437	107	7	14	8	36
M6							**1021**	**6889**	0	0	0	0	1	0	0	0	0	0	0	0
M7								**814**	0	0	0	1	192	1	1	0	2	34	106	0
M8									0	0	0	0	31	0	0	0	0	0	0	0
M9										**6975**	**3496**	492	0	**3354**	**2997**	**729**	278	52	50	**7463**
M10											**6486**	223	0	**5648**	**6986**	**1845**	233	42	38	**8443**
M11												381	0	**13347**	**11329**	**5218**	349	44	34	**9333**
M12													0	**4382**	**3331**	**1208**	**1568**	33	11	**3396**
M13														0	0	0	0	157	**1064**	0
M14															**13296**	**5852**	**967**	54	23	**12415**
M15																**6483**	492	66	22	**10214**
M16																	**809**	35	12	**3081**
M17																		12	3	532
M18																			**4292**	59
M19																				27
M20																				

Supplementary Table 12. Number of times that each pair of mechanisms are proposed together by the model using 6 time points and Gaussian error with σ = 1%.

Data for Fig. S9:

Circos plot for grouping of mechanisms in the predictions of the test set with 6 time points and Gaussian error with σ = 5%. Each value between different mechanisms represents the number of times 2 mechanisms have been grouped together in the predictions. Only connections worth >1% of the total are displayed in Fig. S9 in the manuscript (ie absolute value of connections >932).

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20
M1		**2498**	**1985**	801	**3578**	323	**1024**	33	586	710	718	662	922	842	720	663	790	**1098**	**1413**	460
M2			749	337	**1129**	**1691**	**2652**	**4155**	513	372	311	348	**7715**	324	218	191	370	**1718**	**4184**	257
M3				**2971**	**7131**	196	565	30	**1320**	**3337**	**1153**	455	445	**1161**	**2207**	833	349	274	607	**1017**
M4					**6047**	11	76	0	495	**1775**	756	176	93	706	**1851**	634	217	140	180	448
M5						137	487	9	**1257**	**4395**	**1698**	582	290	**1565**	**3867**	**1423**	732	703	679	**1066**
M6							**3036**	**7209**	13	10	7	45	118	25	17	14	64	23	21	7
M7								**2438**	285	227	133	278	**1318**	196	149	176	322	596	881	163
M8									0	1	0	3	266	2	1	0	12	9	12	0
M9										**13500**	**8258**	**3957**	779	**8488**	**7344**	**3225**	**2956**	**1081**	**1750**	**14308**
M10											**11188**	**2315**	461	**10073**	**14654**	**5088**	**2097**	772	**1247**	**13945**
M11												**1982**	340	**15425**	**13405**	**9478**	**2809**	**1210**	**1165**	**12570**
M12													424	**6312**	**4213**	**2952**	**5767**	852	904	**5600**
M13														365	194	177	324	**1824**	**5223**	415
M14															**14524**	**10095**	**5223**	**1354**	**1208**	**15268**
M15																**10265**	**3010**	**1045**	847	**12560**
M16																	**3550**	**1123**	792	**6482**
M17																		916	703	**4059**
M18																			**7022**	**1048**
M19																				**1247**
M20																				

Supplementary Table 13. Number of times that each pair of mechanisms are proposed together by the model using 6 time points and Gaussian error with σ = 5%.

References

1 Ashish, A. M. A. et al. TensorFlow: large-scale machine learning on heterogeneous systems. Preprint at https://arxiv.org/abs/1603.04467 (2015).

2 Harris, C. R. et al. Array programming with NumPy. *Nature* **585**, 357-362, (2020).

3 Virtanen, P. et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat Methods* **17**, 261-272, (2020).

4 Hunter, J. D. Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering* **9**, 90-95, (2007).

5 The pandas development team Pandas 1.2.3. https://zenodo.org/record/4572994 (Zenodo, 2021).

6 Pedregosa *et al.* Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **12**, 2825-2830, (2011).

7 Haghighi, S., Jasemi, M., Hessabi, S. and Zolanvari, A. (2018). PyCM: Multiclass confusion matrix library in Python. *J. Open Source Softw.*, **3**, 729, (2018).

8 Blackmond, D. G. Reaction progress kinetic analysis: a powerful methodology for mechanistic studies of complex catalytic reactions. *Angew. Chem., Int. Ed.* **44**, 4302-4320 (2005).

9 Baxter, R. D., Sale, D., Engle, K. M., Yu, J.-Q. and Blackmond, D. G. Mechanistic Rationalization of Unusual Kinetics in Pd-Catalyzed C–H Olefination. *J. Am. Chem. Soc.* **134**, 4600-4606 (2012).

10 Nielsen, C. D.-T. and Burés, J. Visual kinetic analysis. *Chem. Sci.* **10**, 348-353 (2019).

11 Janse van Rensburg, W., Steynberg, P. J., Meyer, W. H., Kirk, M. M. & Forman, G. S. DFT prediction and experimental observation of substrate-induced catalyst decomposition in ruthenium-catalyzed olefin metathesis. *J. Am. Chem. Soc.* **126**, 14332-14333, (2004).

12 van der Eide, E. F. & Piers, W. E. Mechanistic insights into the ruthenium-catalysed diene ring-closing metathesis reaction. *Nat. Chem.* **2**, 571-576, (2010).