

Supplementary information

Proof of ATW reliability

ATW can be associated with non-warping distances. Whatever two series A and B, there is a set of parameters noted Π_{1-norm} that allows $ATW(A, B, \Pi_{1-norm}) = 1-norm(A, B)$ following Equation S1. On the other hand, ATW is related to a DTW using a warping window δ in Equation S2.

$$\forall \pi_{ij} \in \Pi_{1-norm}, \pi_{ij} = \begin{cases} 1, & \text{if } i = j \\ \infty, & \text{else} \end{cases} \Rightarrow ATW(A, B, \Pi_{1-norm}) = 1-norm(A, B) \quad (\text{Eq.S1})$$

$$\forall \pi_{ij} \in \Pi_{DTW}, \pi_{ij} = \begin{cases} 1, & \text{if } |i-j| \leq \delta \\ \infty, & \text{else} \end{cases} \Rightarrow \forall i, j \in [1, t], \gamma_A(A'_i, B'_j, \pi_{ij}) = \gamma_D(A'_i, B'_j) \\ \Rightarrow ATW(A, B, \Pi_{DTW}) = DTW(A, B) \quad (\text{Eq.S2})$$

The proof of Equation S1 is the following. Note that the proof for Equation S2 follows the same reasoning; consequently it is not integrated in the text. Let $P_1, P_2, \dots, P_m \dots$ be an infinite sequence of statements defined by Equation S3.

$$P_m : t = m, \forall \pi_{ij} \in \Pi_{1-norm}, \pi_{ij} = \begin{cases} 1, & \text{if } i = j \\ \infty, & \text{else} \end{cases} \Rightarrow ATW(A, B, \Pi_{1-norm}) = 1-norm(A, B) \quad (\text{Eq.S3})$$

Equations S4 and S5 show the statements hold when $m = 1$, and $m = 2$.

if $t=1$ and $\Pi_{1-norm} = (1)$ then

- $m=1$: $ATW(A, B, \Pi_{1-norm}) = \gamma_A(A'_1, B'_1, \pi_{11}) = |a_1 - b_1| \times 1 + 0 = \sum_{k=1}^1 |a_k - b_k|$ (Eq.S4)
 $= 1-norm(A, B) \Rightarrow P_1 \text{ true}$

if $t = 2$ and $\Pi_{1-norm} = \begin{pmatrix} 1 & \infty \\ \infty & 1 \end{pmatrix}$ then

$$ATW(A, B, \Pi_{1-norm}) = \gamma_A(A'_2, B'_2, \pi_{22}) = |a_2 - b_2| \times 1 + \min \left\{ \begin{array}{l} \gamma_A(A'_1, B'_2, \pi_{12}) \\ \gamma_A(A'_2, B'_1, \pi_{21}) \\ \gamma_A(A'_1, B'_1, \pi_{11}) \end{array} \right\} \quad (\text{Eq.S5})$$

• $m=2$:

$$= |a_2 - b_2| \times 1 + \min \left\{ \begin{array}{l} \infty \\ \infty \\ |a_1 - b_1| \times 1 + 0 \end{array} \right\} = |a_2 - b_2| + |a_1 - b_1| = \sum_{k=1}^2 |a_k - b_k|$$

$$= 1-norm(A, B) \Rightarrow P_2 \text{ true}$$

Then the inductive step, *i.e.* $\forall m \geq 2, P_m \Rightarrow P_{m+1}$ must be proved to be true. Assuming P_m is true for $m \geq 2$,

$$ATW(A, B, \Pi_{1-norm}) = 1-norm(A, B) \Rightarrow \gamma_A(A'_m, B'_m, \pi_{mm}) = \sum_{k=1}^m |a_k - b_k|, \text{ and } P_{m+1} \text{ follows}$$

Equation S6.

if $t = (m+1)$ and $\forall \pi_{ij} \in \Pi_{1-norm}, \pi_{ij} = \begin{cases} 1 & \text{if } i = j \\ \infty & \text{else} \end{cases}$ then

$$ATW(A, B, \Pi_{1-norm}) = \gamma_A(A'_{m+1}, B'_{m+1}, \pi_{(m+1)(m+1)})$$

$$= |a_{m+1} - b_{m+1}| \times 1 + \min \left\{ \begin{array}{l} \gamma_A(A'_m, B'_{m+1}, \pi_{m(m+1)}) \\ \gamma_A(A'_{m+1}, B'_m, \pi_{(m+1)m}) \\ \gamma_A(A'_m, B'_m, \pi_{mm}) \end{array} \right\} \quad (\text{Eq.S6})$$

$$= |a_{m+1} - b_{m+1}| \times 1 + \min \left\{ \begin{array}{l} \infty \\ \infty \\ \sum_{k=1}^m |a_k - b_k| \end{array} \right\} = |a_{m+1} - b_{m+1}| + \sum_{k=1}^m |a_k - b_k| = \sum_{k=1}^{m+1} |a_k - b_k|$$

$$= \sum_{k=1}^t |a_k - b_k| = 1-norm(A, B)$$

$\forall m \in \mathbb{N}^*, m \geq 2, P_m \Rightarrow P_{m+1}$, consequently the statements $P_1, P_2, \dots, P_m \dots$ are true, and thus Equation S1 is true.

Warping path and ATW formula

The sequence of pairs of points that are matched together is the warping path $W(A,B)=w_1,\dots,w_k,\dots,w_K$ with $w_k=(a_i, b_j)_k$ and $i, j \in [1..t]$. w_k is the k^{th} pair of the warping path, and is composed of the i^{th} point of the series A and the j^{th} point of B . K , i.e. the length of the warping path, can be superior or equal to the length t of the series involved. Due to the meaning of the matching, a warping window δ is employed. A point that occurs at instant t can be matched with points from the other series that occur in $[(i - \delta)..(i + \delta)]$. Let us defined $A'_i=(a_1,\dots,a_i)$ and $B'_j=(b_1,\dots,b_j)$ with $1 \leq i, j \leq t$, two subsequences of A and B starting at 1 and finishing respectively at i and j . Let Π a $t \times t$ matrix, be the set of parameters required to compute ATW, $\Pi=[\pi_{ij}] \in \mathbb{R}^+$, $\forall i, j \in [1, t]$. If $\pi_{ij}=\infty$, a_i is not allowed to be matched with b_j . $\gamma(A'_i, B'_j, \pi_{ij})$ is the recursive function that computes the distance between A'_i and B'_j . ATW is defined as it follows (Eq.S7), (Eq.S8) corresponding to DTW is given for comparison.

$$ATW(A, B, \Pi) = \gamma_A(A'_t, B'_t, \pi_{tt})$$

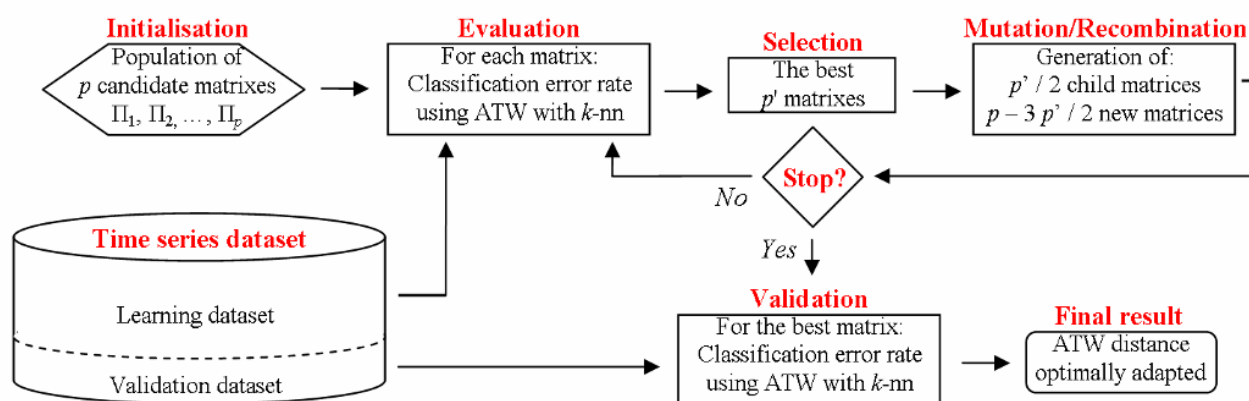
$$\text{with } \gamma_A(A'_i, B'_j, \pi_{ij}) = \begin{cases} \infty & \text{if } \pi_{ij} = \infty \\ \left. \begin{array}{l} 0, \text{ if } i = j = 1 \\ \gamma_A(A'_i, B'_{j-1}, \pi_{i(j-1)}), \text{ if } i = 1 \text{ and } j > 1 \\ \gamma_A(A'_{i-1}, B'_j, \pi_{(i-1)j}), \text{ if } i > 1 \text{ and } j = 1 \\ \min \left\{ \begin{array}{l} \gamma_A(A'_{i-1}, B'_j, \pi_{(i-1)j}) \\ \gamma_A(A'_i, B'_{j-1}, \pi_{i(j-1)}) \\ \gamma_A(A'_{i-1}, B'_{j-1}, \pi_{(i-1)(j-1)}) \end{array} \right\}, \text{ else} \end{array} \right\} \times \pi_{ij} + \end{cases} \quad (\text{Eq.S7})$$

$$DTW(A, B) = \gamma_D(A'_t, B'_t)$$

$$\gamma_D(A'_i, B'_j) = |a_i - b_j| + \begin{cases} 0, \text{ if } i = j = 1 \\ \gamma_D(A'_i, B'_{j-1}), \text{ if } i = 1 \text{ and } j > 1 \\ \gamma_D(A'_{i-1}, B'_j), \text{ if } i > 1 \text{ and } j = 1 \\ \min \left\{ \begin{array}{l} \gamma_D(A'_{i-1}, B'_j) \\ \gamma_D(A'_i, B'_{j-1}) \\ \gamma_D(A'_{i-1}, B'_{j-1}) \end{array} \right\}, \text{ else} \end{cases} \quad (\text{Eq.S8})$$

Optimization with genetic algorithm

The “brute force” solution would perform the classification with all the possible values of Π , and finally keeps the one that corresponds to the best error rate. Each possible matrix Π must be tested for obtaining the error rate. However, the resulting complexity in time of the exhaustive search is intractable. This implies the use of a heuristic which aims at quickly finding a solution, *i.e.* obtaining a locally optimal matrix Π . Genetic algorithms (GAs) are selected because there is a body of theory providing theoretical evidence to complement the empirical proof of their robustness. Such techniques are now consistently used. The reader is referred to [1] for an introduction and advanced discussions.



Let X be a set of n time series, $X = \{x_1, \dots, x_n\}$, and Π a population of p individuals $\{\Pi_1, \dots, \Pi_p\}$. Each individual is a candidate matrix Π_i that owns t^2 variables. The terms “individuals”, “population”, etc... are used following the GA terminology. Each cell of the matrix is a variable that the GA optimizes. The GA (*i.e.* all individuals Π_i) is initialized by assigning random values to every cell π_{ij} . The evaluation, then the selection, and finally the mutation/recombination are applied to each generation. The evaluation of Π_i computes the classification error rate with a k' -cross validation (k' -CV), using ATW distance and k -nearest neighbors (k -nn) on the learning dataset. The classification error rate is employed as fitness function, *i.e.* indicating the relevance of a given matrix for the classification task. The selection operator picks up the p' best individuals of the current generation. Matching parents randomly, $p'/2$ pairs are created for reproduction. The same number of “children”

matrices is produced using a special crossover due to the matrix chromosomal representation. For crossover, Π is cut into q peaces noted π^b with $b \in \mathbb{N}$, $b \in [1..q]$, $\pi^b = \{\pi_{ij}\}$, $((b-1)n/q) < (i, j) < (bn/q)$. For example, if $q=3$ then $\pi^1 = \pi_{ij}$ with $(i, j) < (n/3)$, $\pi^2 = \pi_{ij}$ with $(n/3) < (i, j) < (2n/3)$, and $\pi^3 = \pi_{ij}$, $(2n/3) < (i, j) < n$. Then, each peace of the children is randomly taken from the parents with equiprobability. In order to prevent from stagnation, children are mutated; each π_{ij} is modified with a probability p_m following the adapted mutation described by Equation S10. Finally, $(p-3p'/2)$ new individuals are randomly generated for maintaining a fixed population size. The loop is repeated until a termination condition is reached such as a classification error rate threshold or an *a priori* number of iterations. As such, ATW distance is adapted for the classification task in hand. Over fitting is prevented by testing the best candidate matrix on the validation dataset.

The GA process enables us to considerably reduce the complexity of the whole process. Rather than exploring all the possible values of Π , p matrixes are evaluated at each iteration. Another reason GA is particularly fit is that the multi-dimensional continuous search space defined by Π is transformed in a discrete one which speed up ATW calculations by order of magnitude. π_{ij} is fixed to 0, 1, 2 or ∞ , $\forall i, j \in [0, t]$. After numerous tests such a set of values appeared sufficient for a nearly-optimal efficiency of the algorithm which can be explained by the following:

- ∞ : The matching is not allowed between points i and j
- 0 : There is no influence on the global distance
- 1 : Corresponds to a “normal” influence of the comparison between the two points
- 2 : The influence of the comparison is high (*i.e.* important). The “weight” of this distance is two times bigger than in 1 considering the global distance.

Consequently, the initialization is changed to Equation S9 and an adapted mutation Equation S10 is employed.

$$\forall \pi_{ij} \in \Pi_i, \pi_{ij} = \begin{cases} \text{Random } \{0,1,2\} & \text{if } i = j \\ \infty & \text{otherwise} \end{cases} \quad (\text{Eq. S9})$$

$$\begin{aligned} \text{mutate1}(\pi_{ij}) &= \begin{cases} 1 & \text{if } \pi_{ij} = \{0,2\}, i = j \\ \text{Random } \{0,2\} & \text{if } \pi_{ij} = 1, i = j \\ \pi_{ij} & \text{otherwise} \end{cases} \\ \text{mutate2}(\pi_{ij}) &= \begin{cases} 1 & \text{if } \pi_{ij} = 2 \\ \text{Random } \{0,2\} & \text{if } \pi_{ij} = 1 \\ \text{Random } \{1,\infty\} & \text{if } \pi_{ij} = 0, i \neq j, \pi_{(i-1)j} > 0, \pi_{i(j-1)} > 0 \end{cases} \end{aligned} \quad (\text{Eq.S10})$$

The learning is first applied using “mutate1” and the Euclidian distance until the diagonal values of the matrix are stabilized which allows discriminating the importance of the different zones. The remaining of the matrix is filled using “mutate2”. The first phase is the optimization of the matrix diagonal with the Euclidian distance, and the second one allows modifying the cells which are accessible from the diagonal element with value 1 or 2.

Synthesis of “hexamethonium study”

Hexamethonium is used as structure directing agent (SDA). An initial experimental factorial design (3×4^3) is selected. Si/Ge, T^{III}/(Si+Ge), OH⁻/(Si+Ge), and H₂O/(Si+Ge) are the synthesis variables. This design considers the following four molar ratios (level): Si/Ge (**4**) ranging from 2 to 30; B/(Si+Ge) (**4**) from 0 to 0.05; OH⁻/(Si+Ge) (**3**) from 0.1 to 0.5; and H₂O/(Si+Ge) (**4**) from 5 to 30. The total number of samples synthesized and characterized is 192. The flexibility of the hexamethonium allows different conformations that stabilize diverse competing structures, as EU-1, ITQ-17, ITQ-22, ITQ-24, SSZ-31, a lamellar phase, and the new structure ITQ-33.

Figure S1. Mean relative gain on error rate using ATW when compared to DTW with best delta value and Euclidian distances on both “ITQ21-30” and “zeolite beta” real datasets. The indicated percentages are the average values according to the *k*-fold cross-validation.

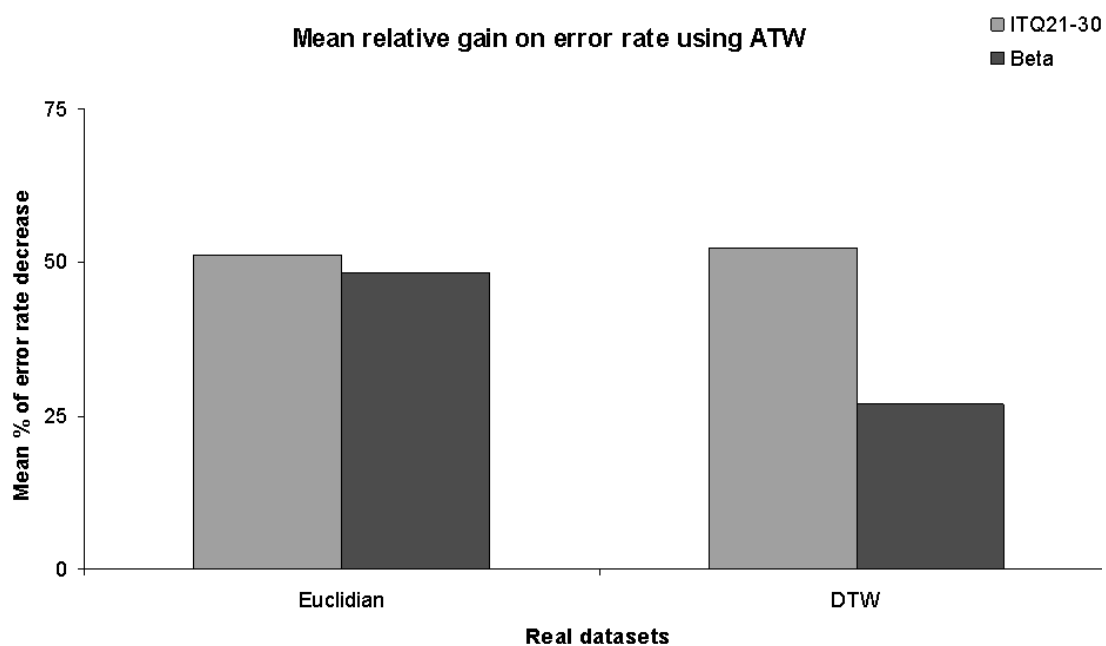
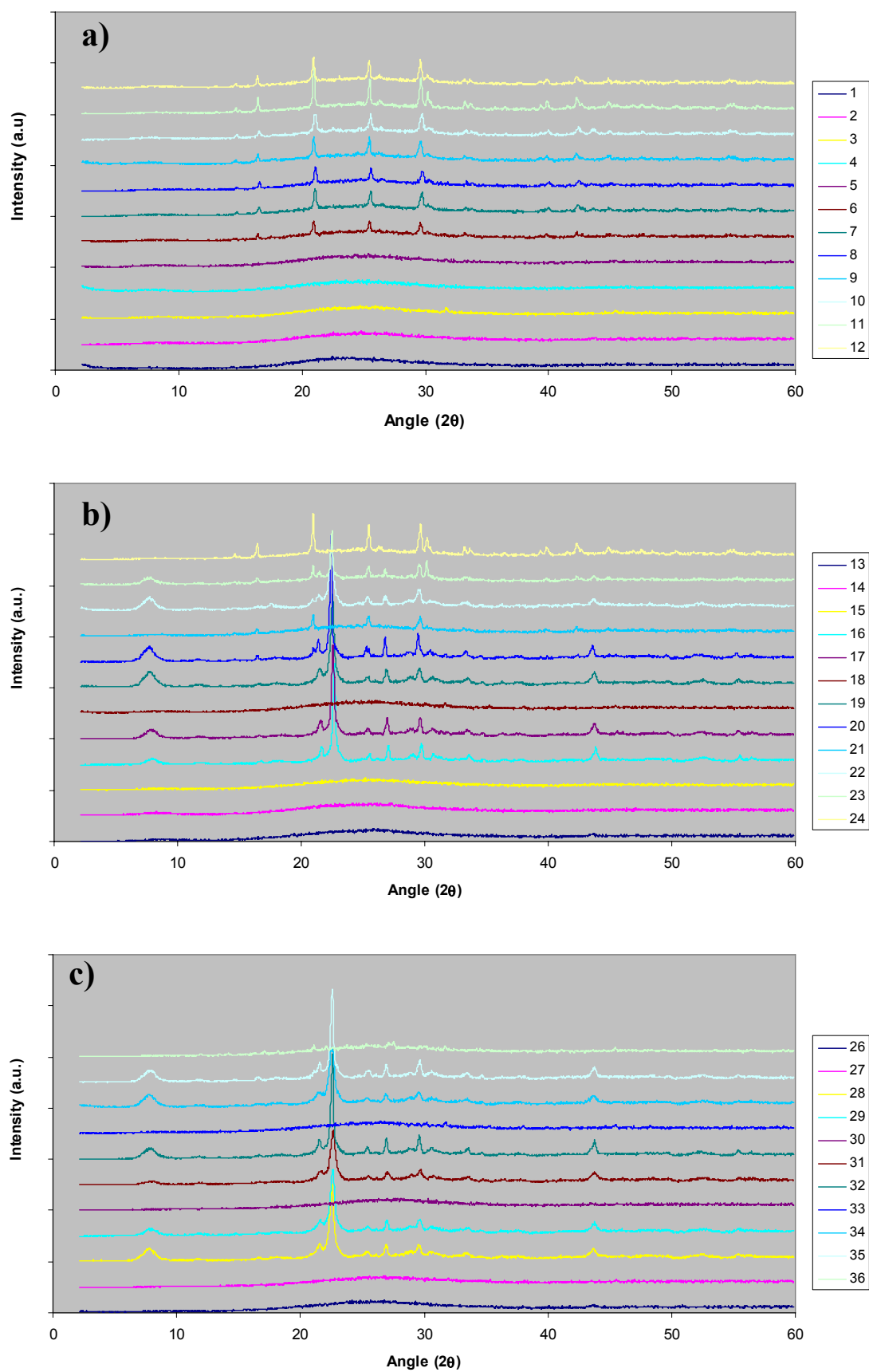
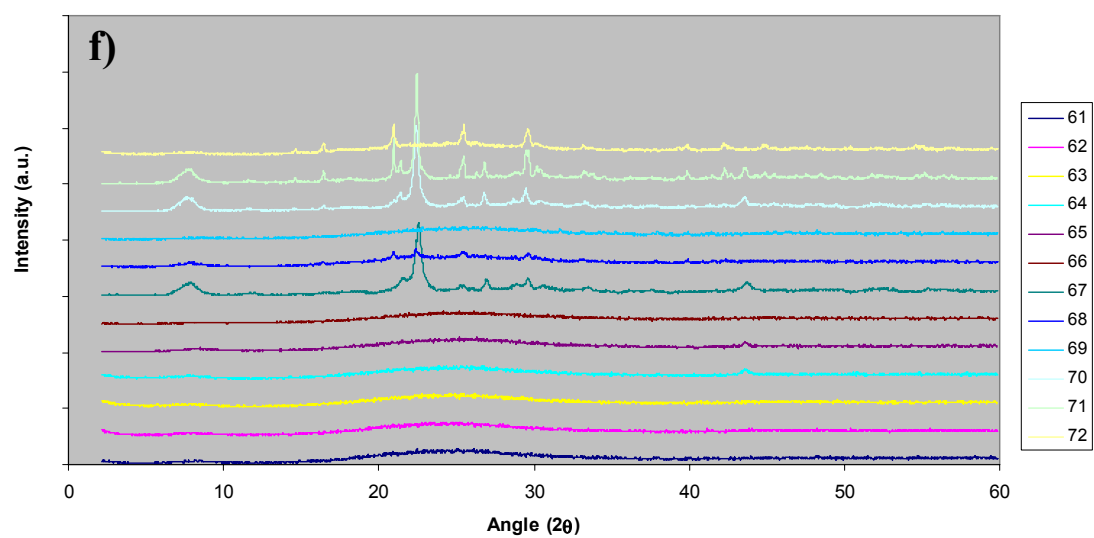
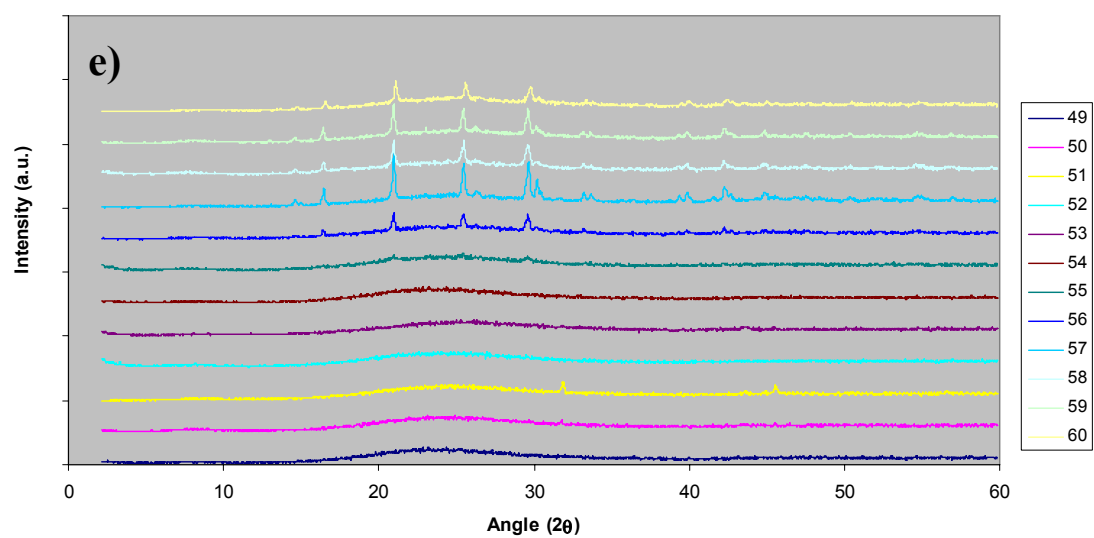
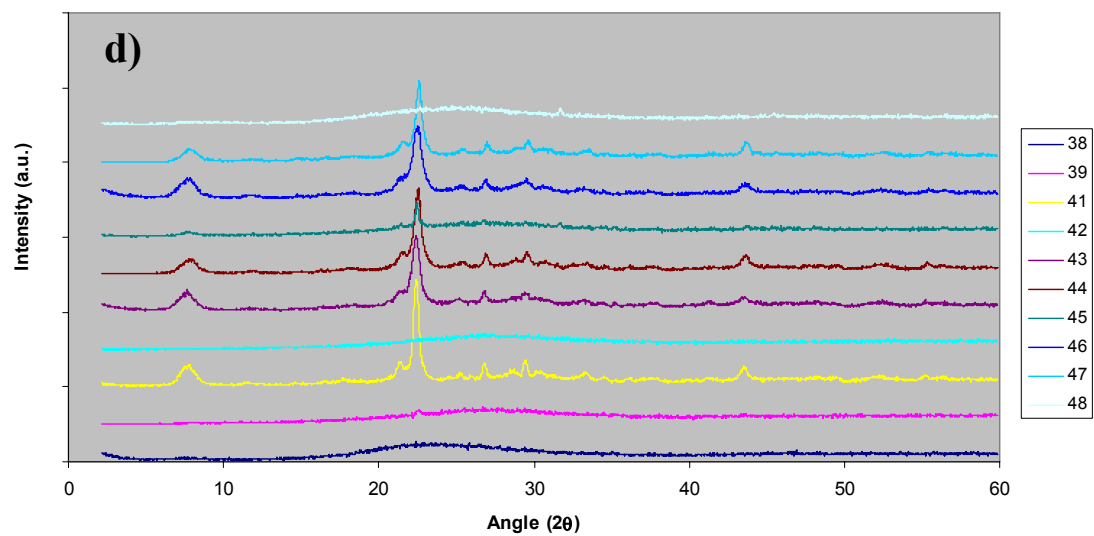


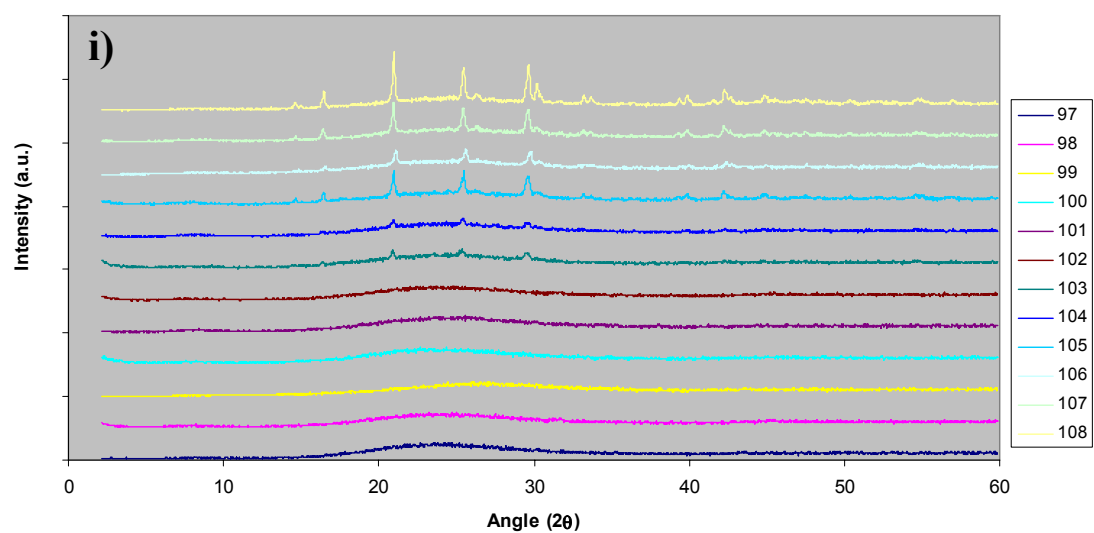
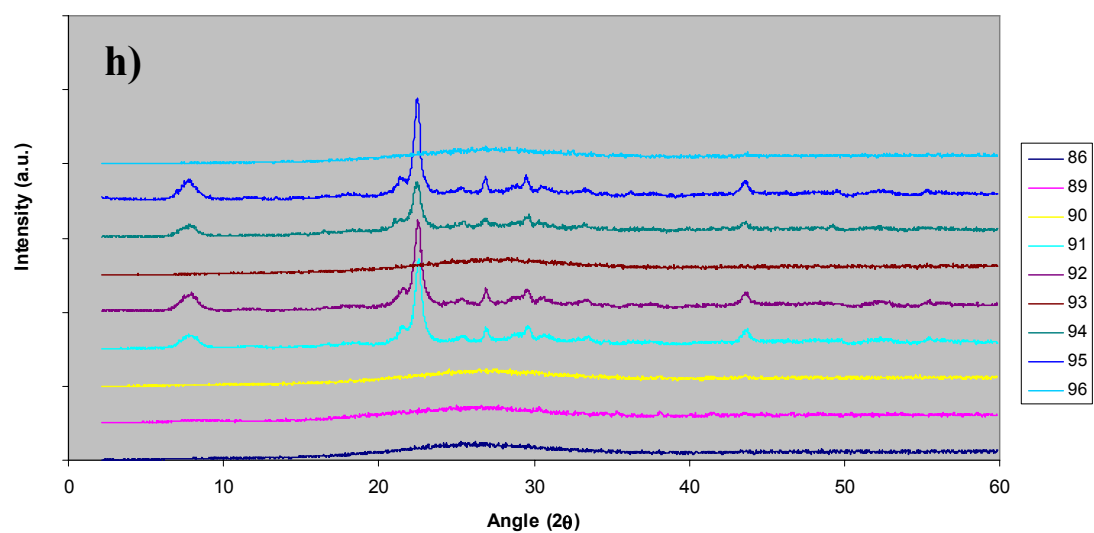
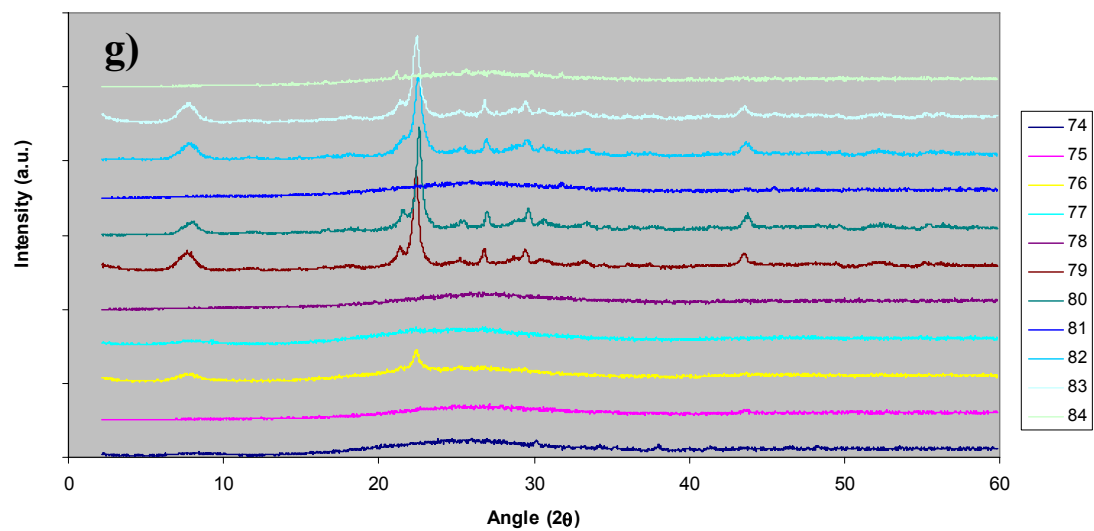
Table S2. Performance (error rate) comparison on benchmarks

Benchmark Name	Euclide	DTW _δ (best δ value)	DTW (no warping window)	ATW
Synthetic Control	0.12	0.017 (6)	0.007	0.007
Gun-Point	0.087	0.087 (0)	0.093	0.033
CBF	0.148	0.004 (11)	0.003	0.003
Face (all)	0.286	0.192 (3)	0.192	0.192
OSU Leaf	0.483	0.384 (7)	0.409	0.384
Swedish Leaf	0.213	0.157 (2)	0.210	0.157
50Words	0.369	0.242 (6)	0.310	0.242
Trace	0.24	0.01 (3)	0.0	0.0
Two Patterns	0.09	0.0015 (4)	0.0	0.0
Wafer	0.005	0.005 (1)	0.020	0.004
Face (four)	0.216	0.114 (2)	0.170	0.068
Lightning-2	0.246	0.131 (6)	0.131	0.131
Lightning-7	0.425	0.288 (5)	0.274	0.274
ECG	0.12	0.12 (0)	0.23	0.10
Adiac	0.389	0.391 (3)	0.396	0.389
Yoga	0.170	0.155 (2)	0.164	0.155
Fish	0.217	0.160 (4)	0.167	0.160
Beef	0.467	0.467 (3)	0.5	0.433
Coffee	0.25	0.179 (8)	0.179	0.071
OliveOil	0.133	0.167 (1)	0.133	0.100

Figures S3 (a-l): X-ray diffractograms for all synthesized samples in the Beta study







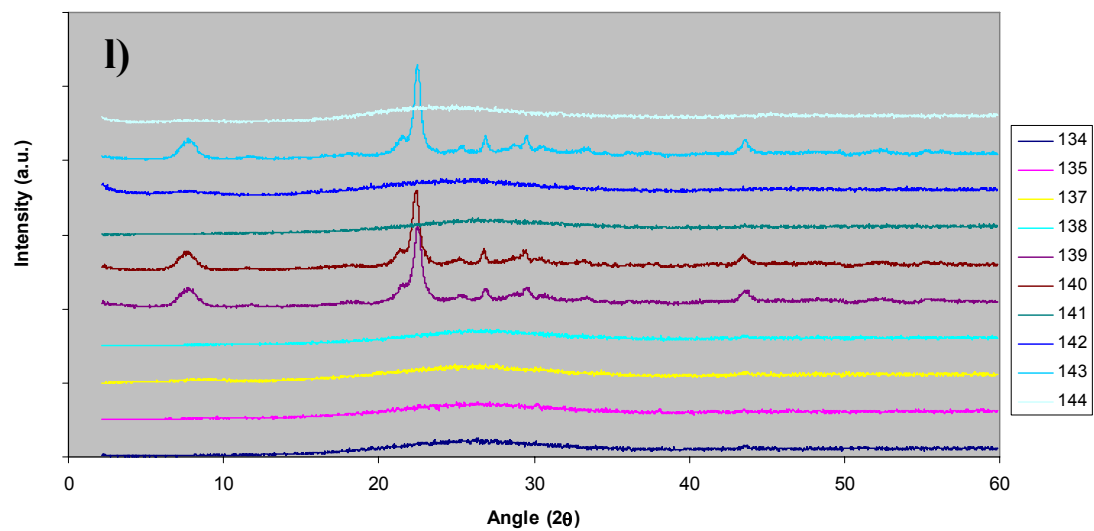
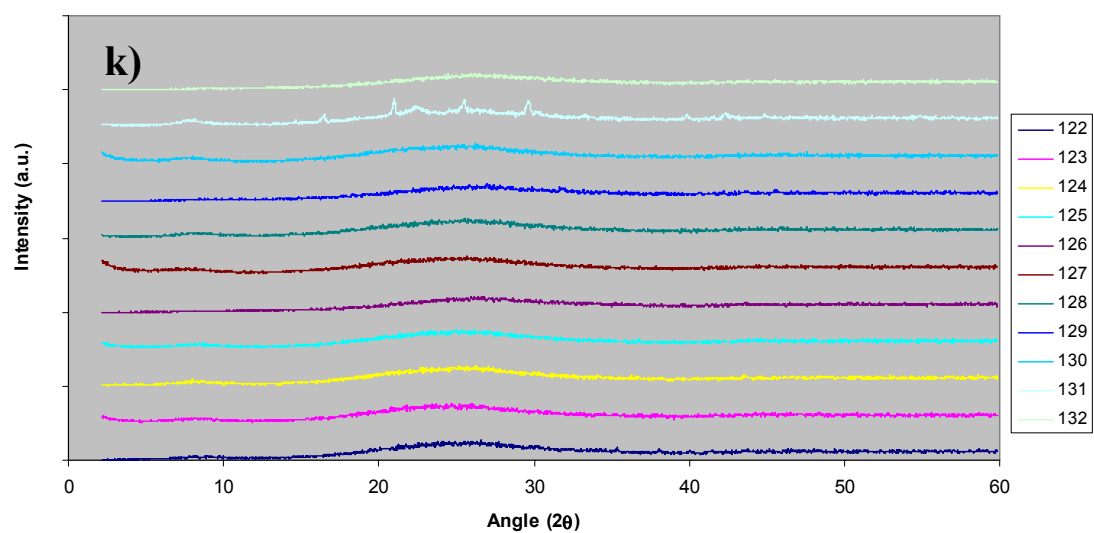
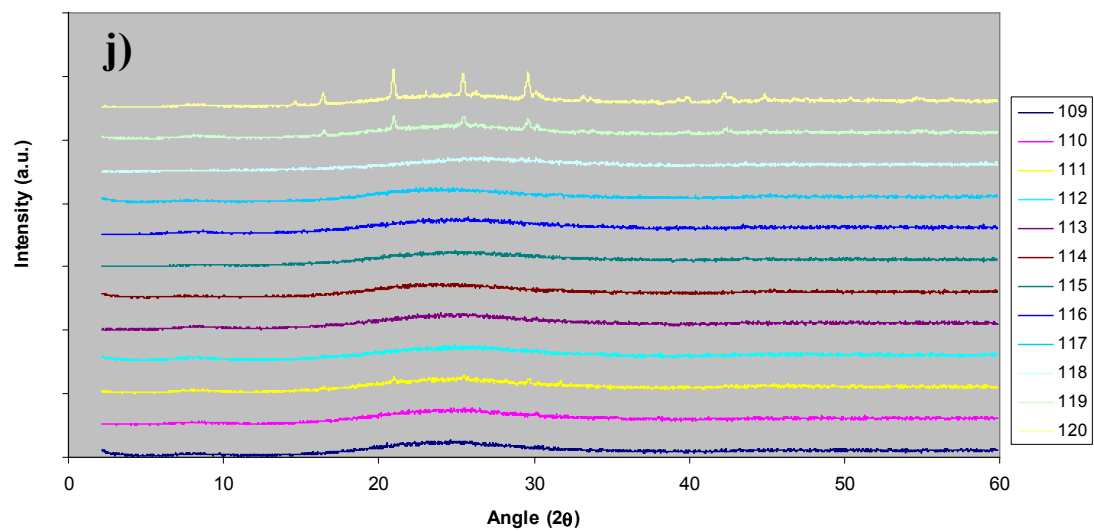
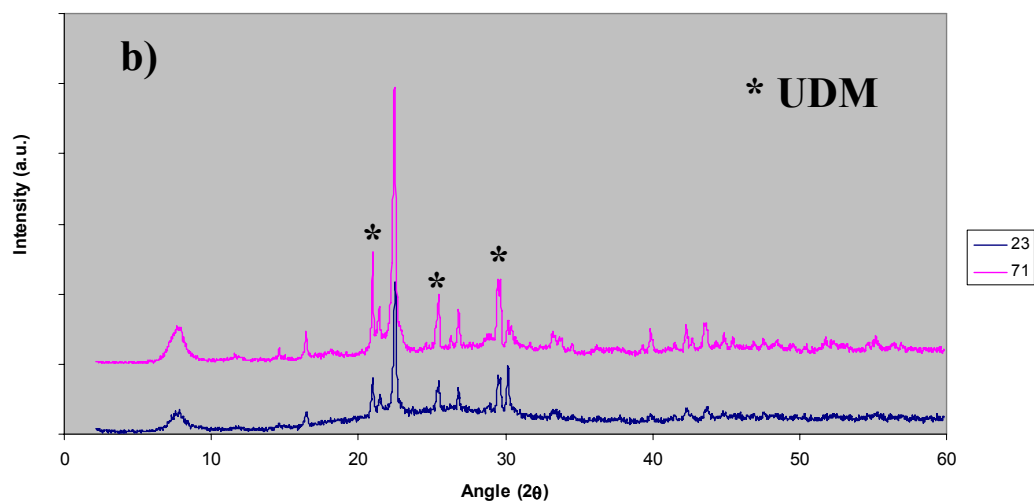
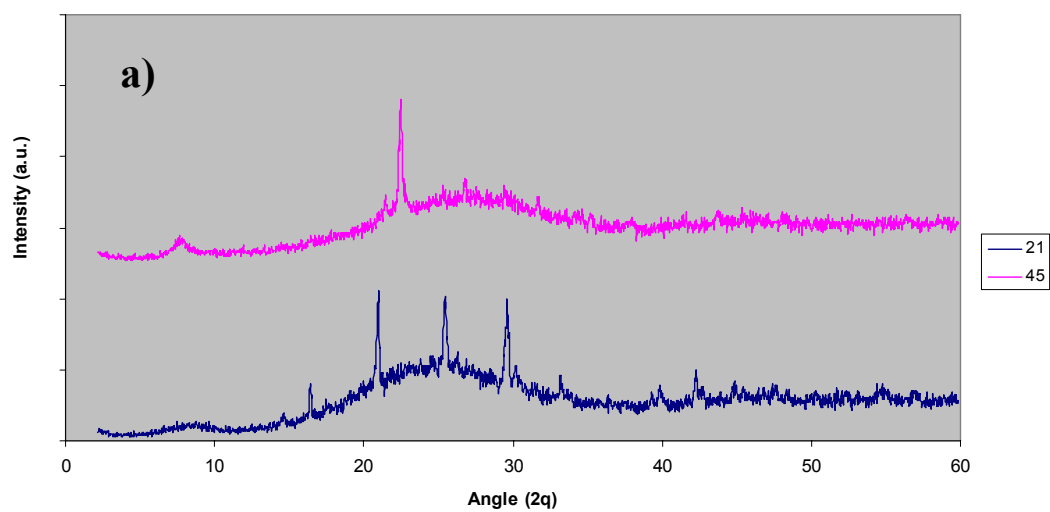


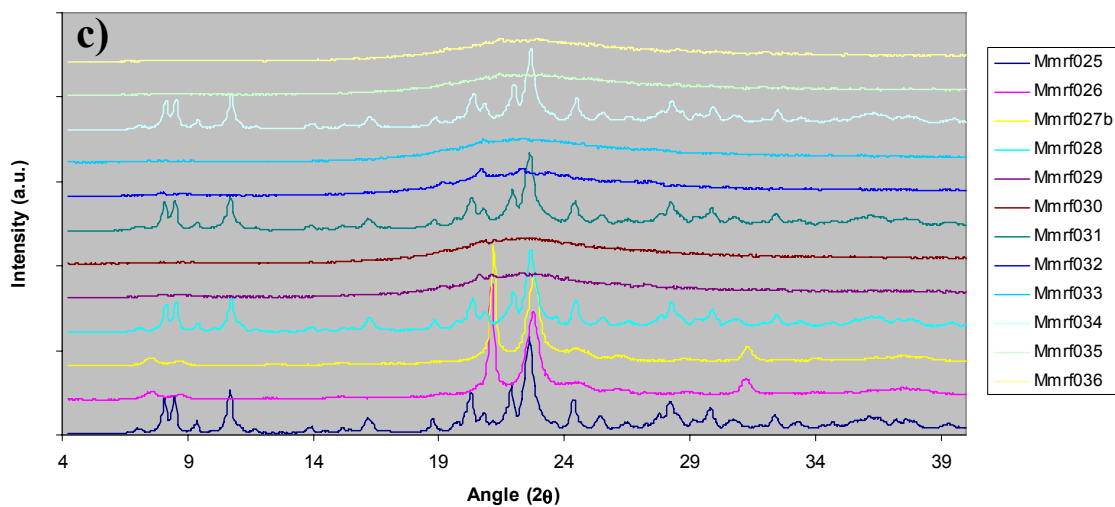
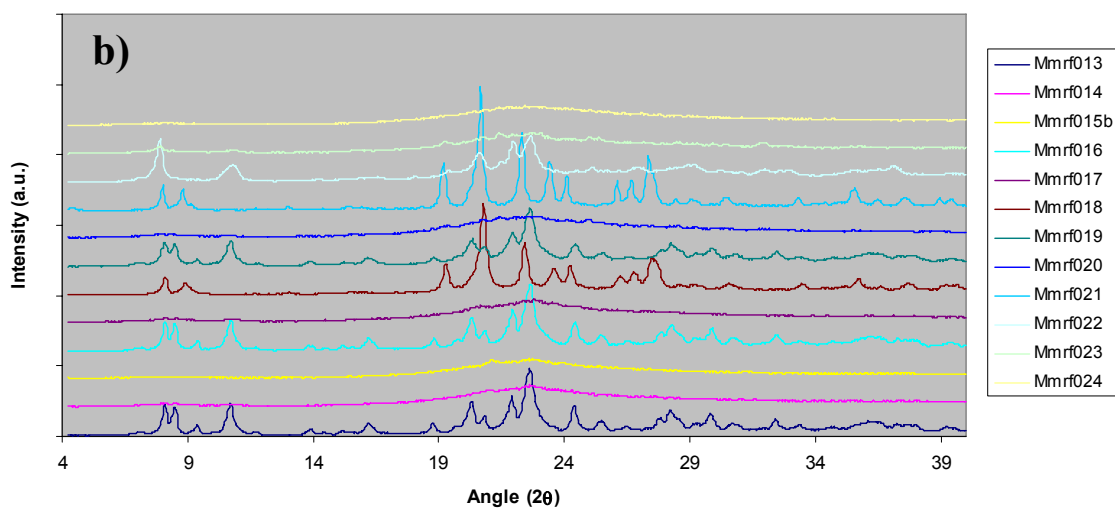
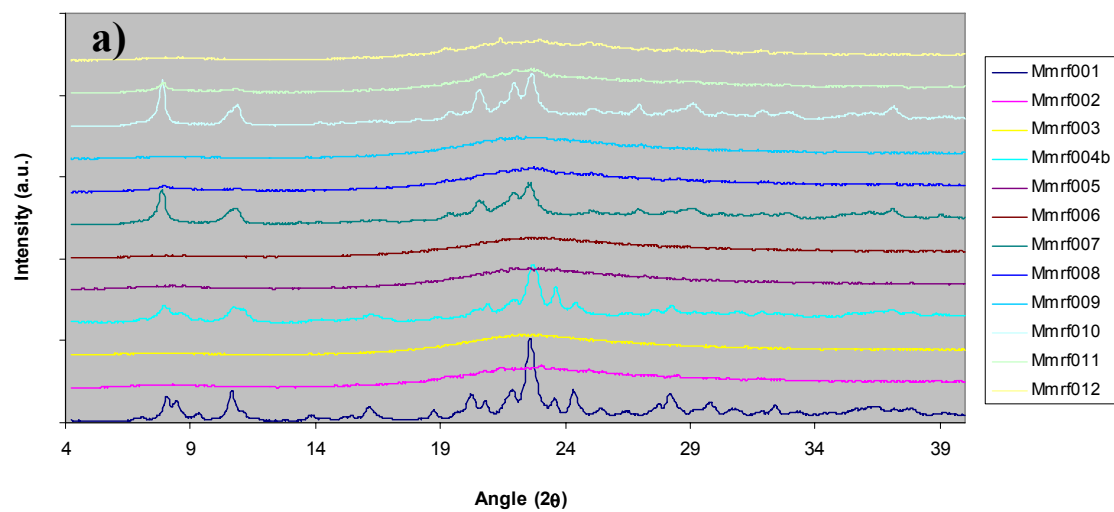
Table S4: Confusion matrix with the real phases in the experimental design “Beta study” versus predicted classes obtained with PolySnap2®.

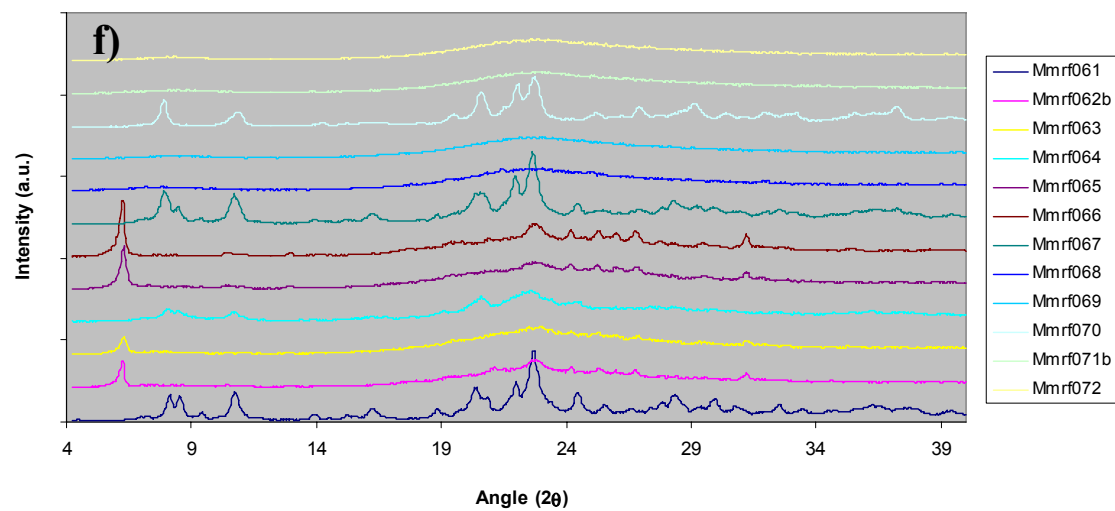
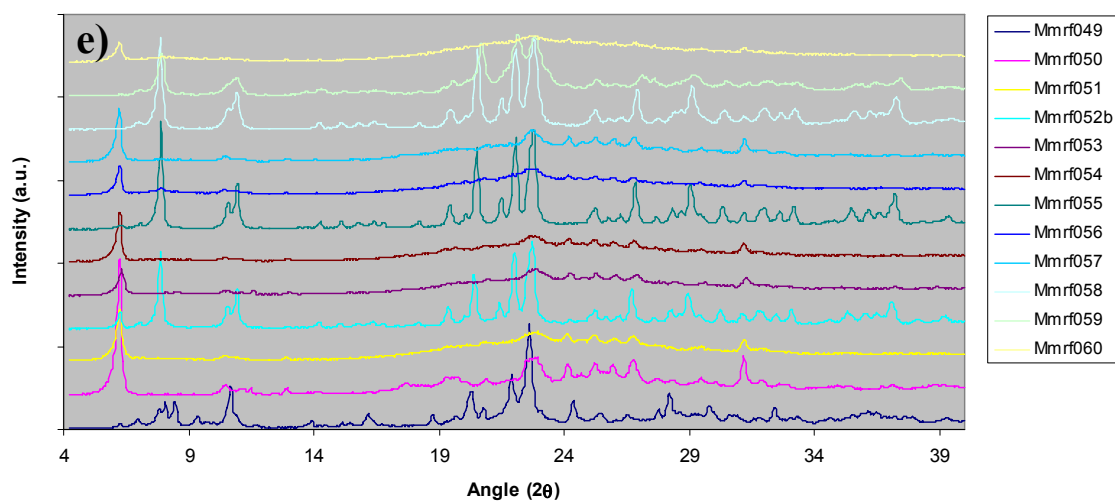
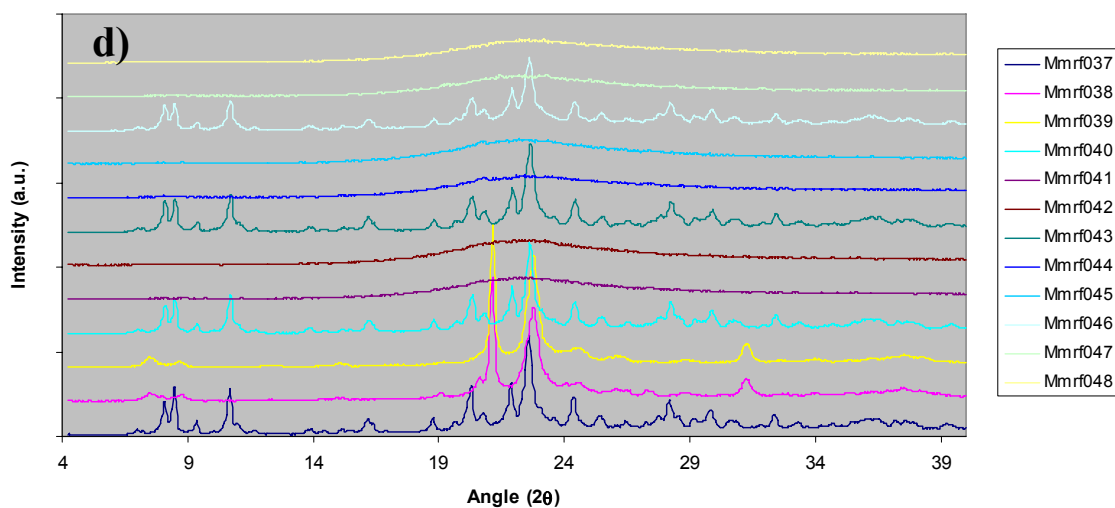
		REAL PHASES							TOTAL
		amorphous	UDM	Growing UDM	Beta	Growing Beta	Mixture	Growing Mixture	
PREDICTED PHASES	amorphous	75	1	9		2		2	89
	UDM		11	2					13
	Grow UDM			0					0
	Beta				27		2		29
	Grow Beta					0			0
	Mixture	1			2		0		3
	Grow Mixture							0	0
	TOTAL	76	12	11	29	2	2	2	113/134 (85%)

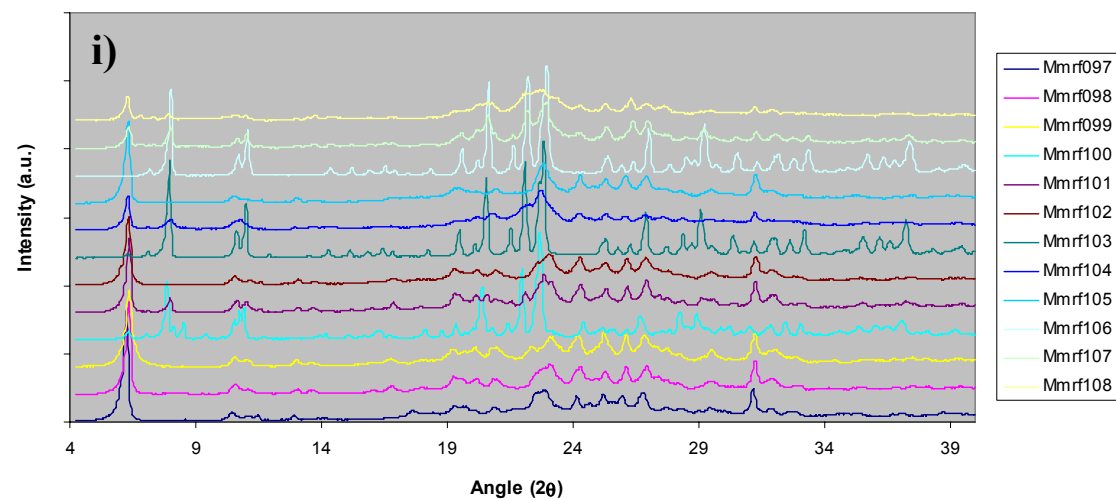
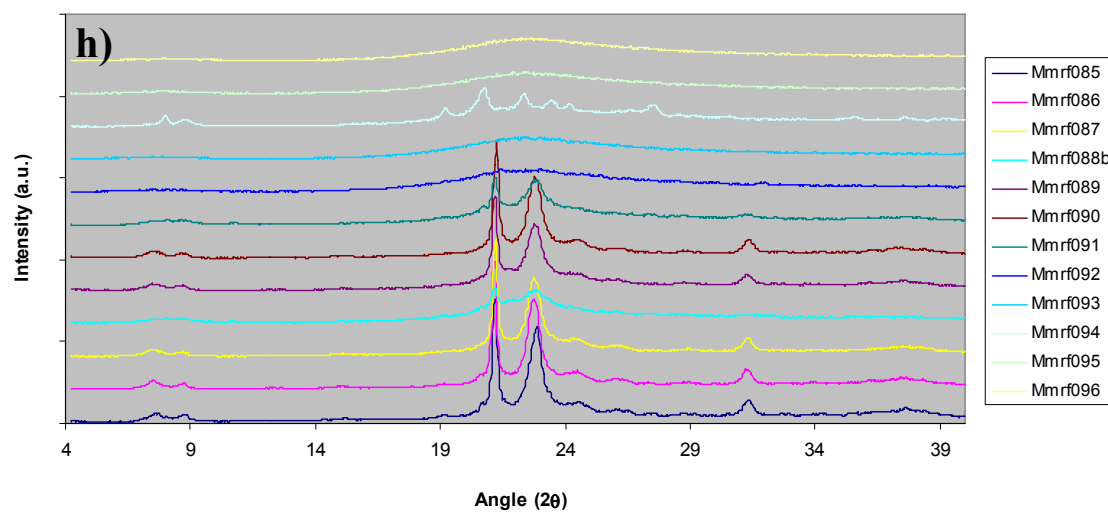
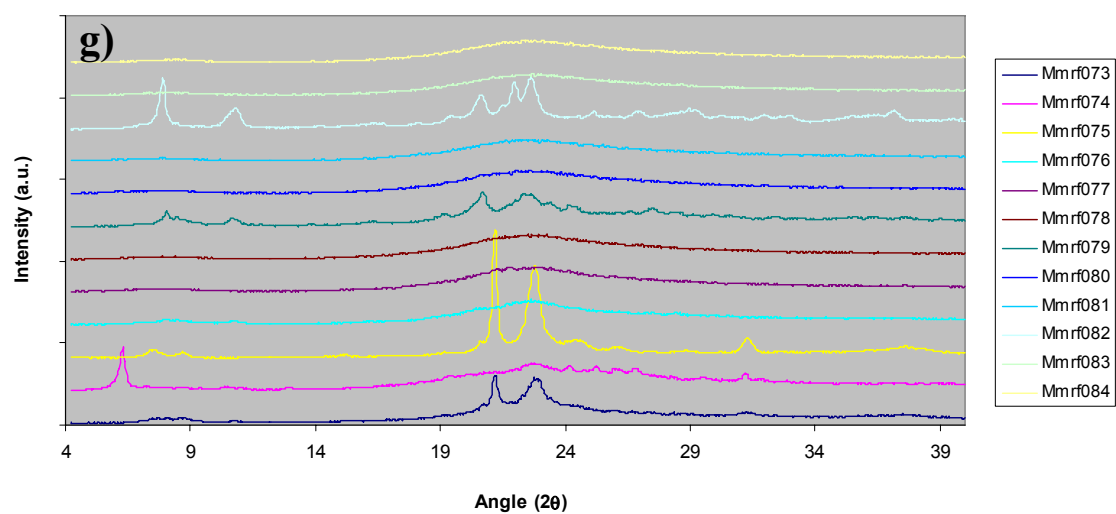
Figures S5: Predicted errors by using PolySnap2[®]. a) Growing phases, UDM (21) and Beta (45), being the software prediction “amorphous”. b) Mixtures between Beta and UDM, being the software prediction “Beta” phase.

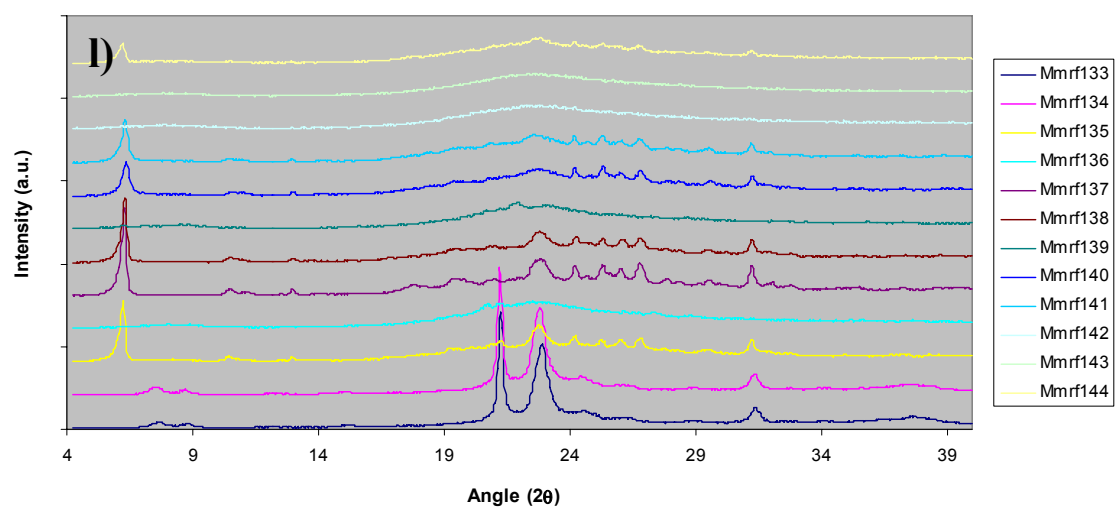
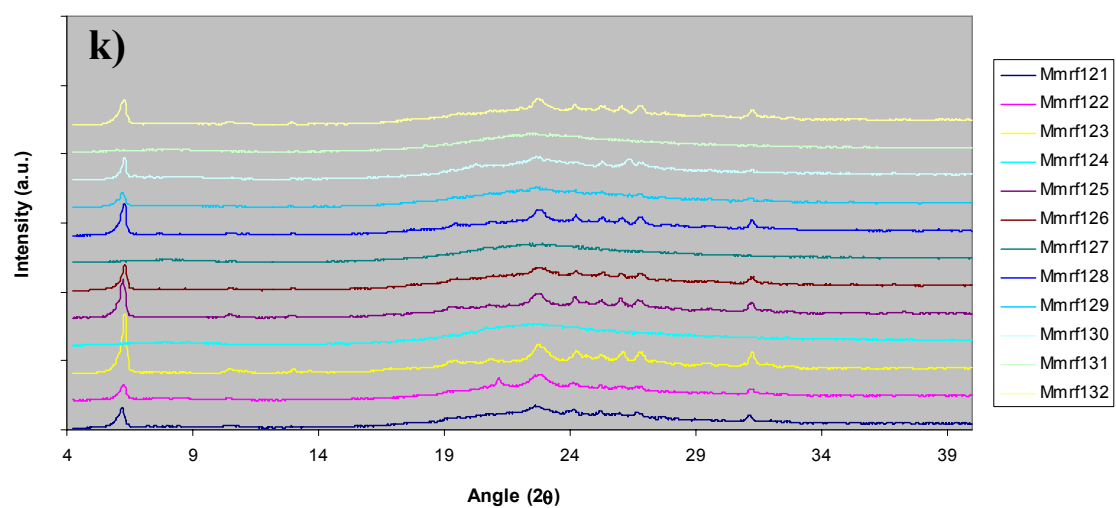
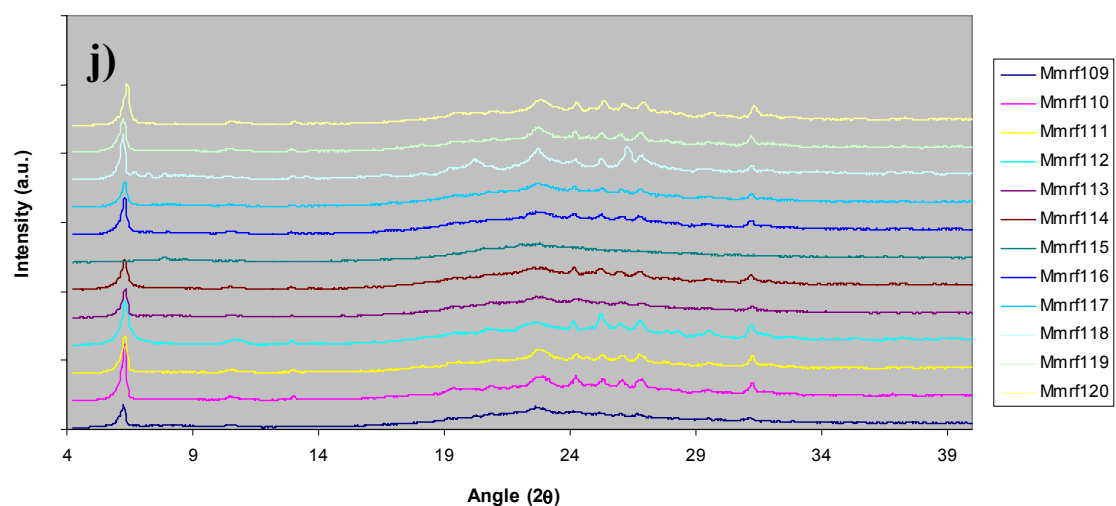


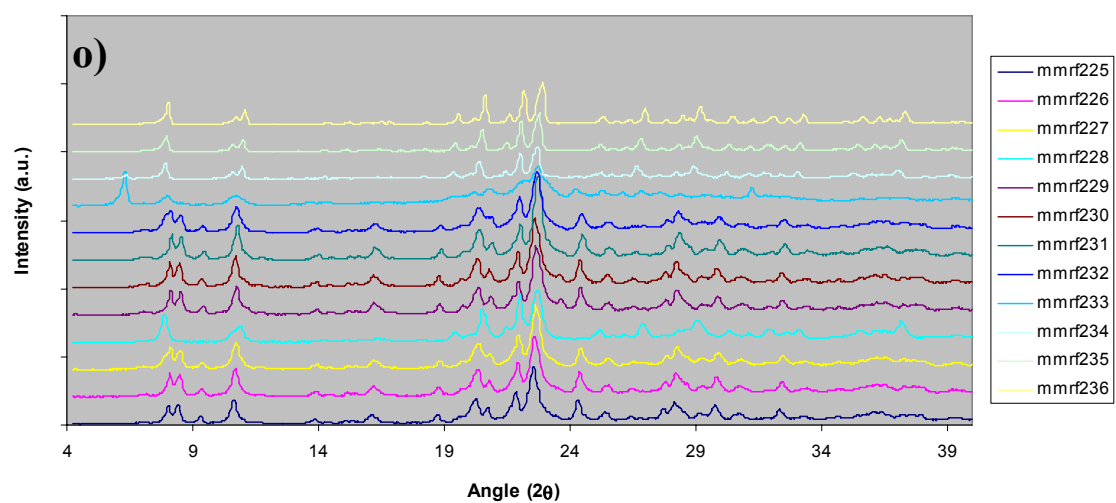
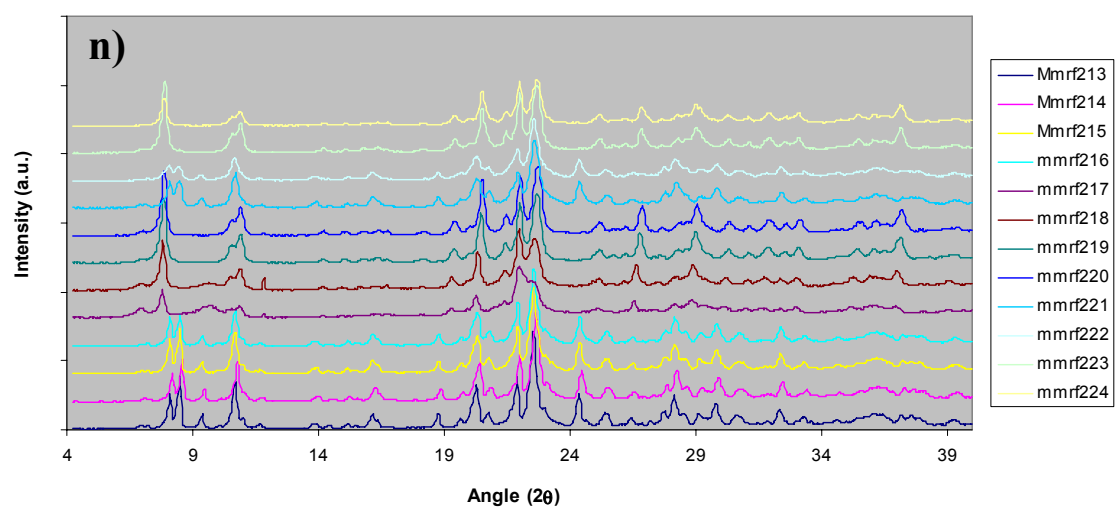
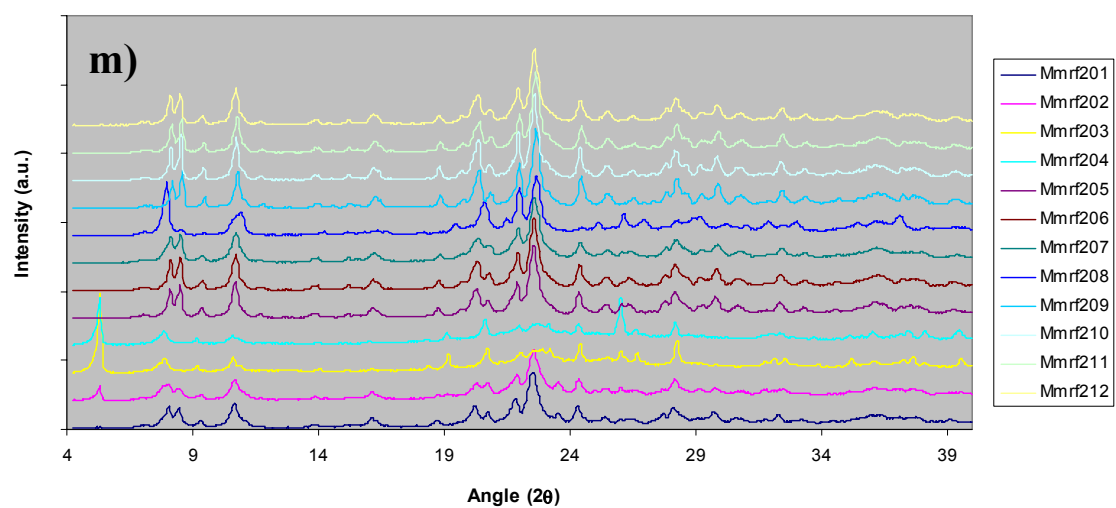
Figures S6 (a-p): X-ray diffractograms for all synthesized samples in the “Hexamethonium study”











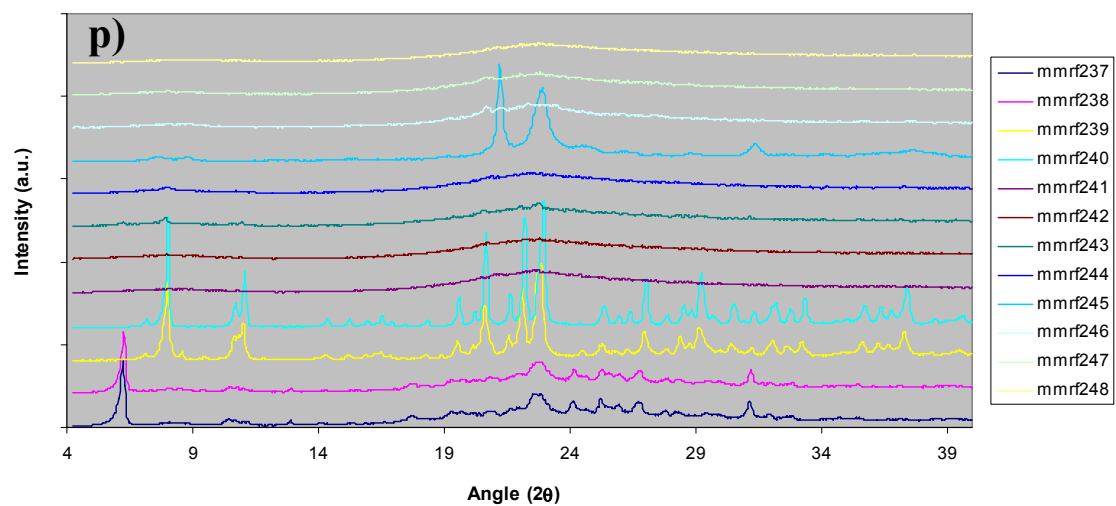
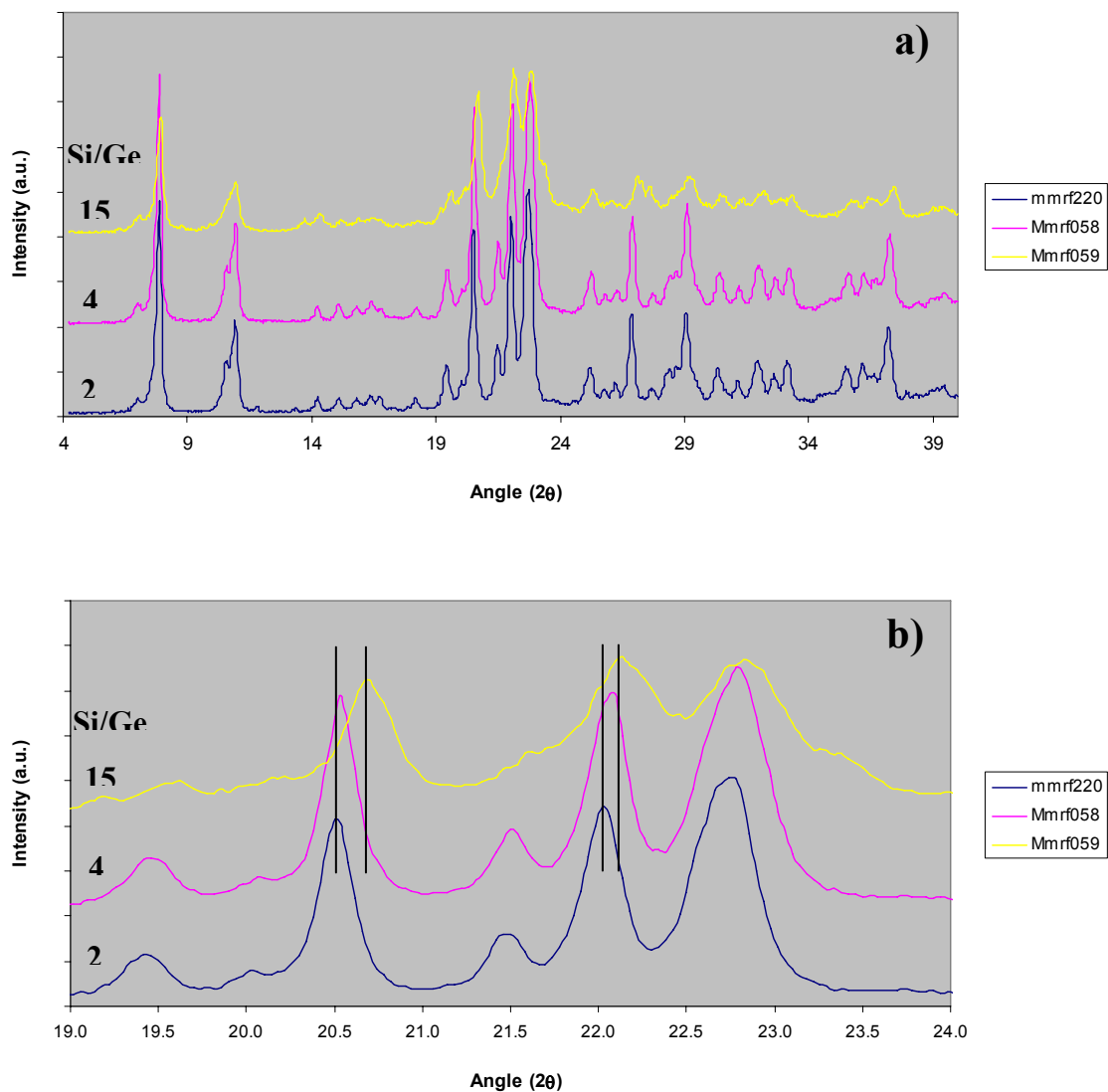


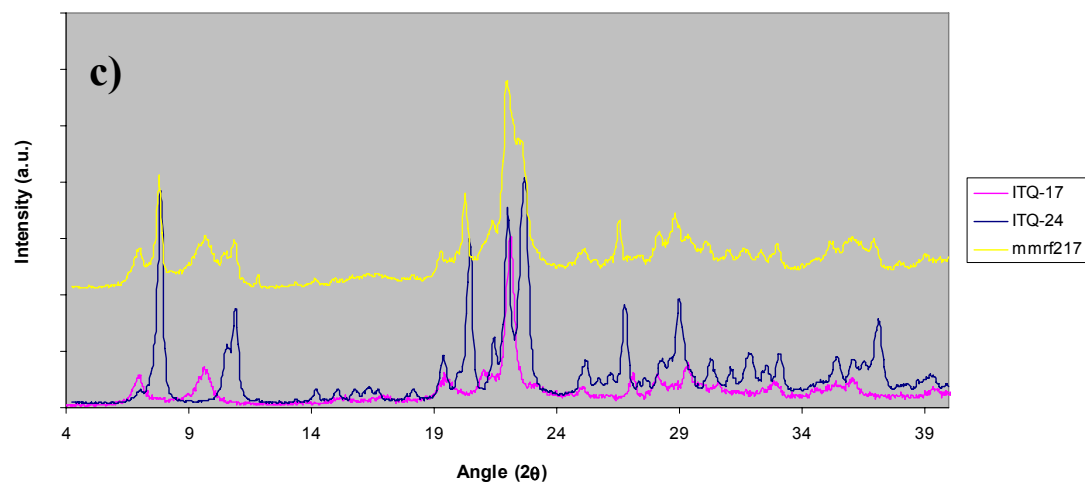
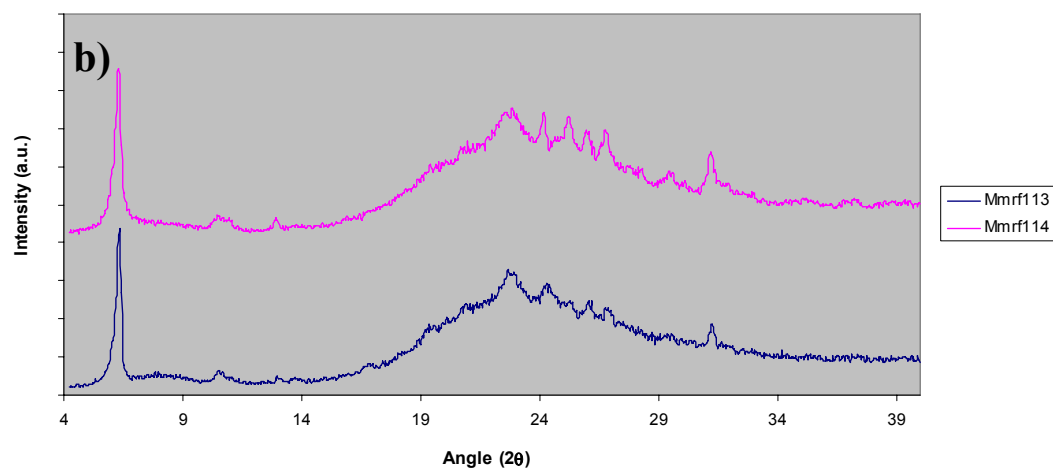
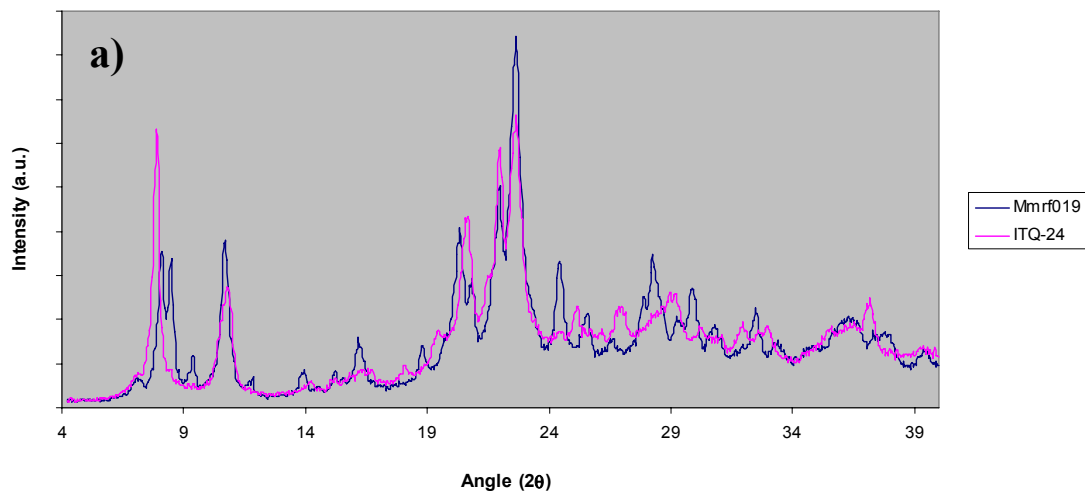
Table S7: Confusion matrix with the real phases in the experimental design “Hexamethonium study” versus predicted classes obtained with PolySnap2[®].

Predicted	Real												
	Amorphous	ITQ-22	ITQ-24	EU-1	SSZ-31	Lamellar	ITQ-17	Lamellar + ITQ-24	ITQ-24 + ITQ-33	ITQ-17 + ITQ-24	ITQ-22 + ITQ-24		ITQ-22 + ITQ-24 + ITQ-33
Amorphous	55	2	3	2	3	28	2						95
ITQ-22		36	2								1	1	40
ITQ-24			8				2						10
EU-1				1									1
SSZ-31					13								13
Lamellar						18	1						19
ITQ-17							0			1			1
Lamellar/ITQ-24								0					0
ITQ-24/ITQ-33									0				0
ITQ17/ITQ-24										0			0
ITQ-22/ITQ-24											0		0
ITQ-22/ITQ-24/ITQ-33												0	0
Mixtures (¿?)			6				3	2	1	1			13
	55	38	19	3	16	46	0	8	2	2	2	1	131/192 =68%

Figures S8: XRD patterns for ITQ-24 zeolites obtained in the “hexamethonium study” with different Si/Ge ratio (a) full pattern, and (b) small angle range (19-24°).



Figures S9: Predicted errors by using PolySnap2[®]. a) The real phase is ITQ-22 (MMRF019), being the software prediction ITQ-24, b) it is growing the Lamellar phase, being the software prediction “amorphous”, and c) it predicts pure ITQ-17 (MMRF217), being a mixture between ITQ-17 and ITQ-24.



- ¹ a) D. E. Goldberg. *The Design of Innovation: Lessons from and for Competent Genetic Algorithms*, Addison-Wesley, Reading, MA. **2002**. b) L. M. Schmitt. *Theory of Genetic Algorithms*, *Theoretical Computer Science* (259), pp. 1-61. **2001**. c) M. D. Vose. *The Simple Genetic Algorithm: Foundations and Theory*, MIT Press, Cambridge, MA. **1999**. d) D. Whitley. *A genetic algorithm tutorial*. *Statistics and Computing* 4, 65–85. **1994**