Marie Curie Initial Training Network

Environmental Chemoinformatics (ECO)

Final Report

20 July 2012

Applying QSAR/QSPR approaches to nanoparticles

Early stage researcher:

Ehret Jacques

Project supervisor:

Prof. Willie Peijnenburg

Research Institution:

Universiteit Leiden

Introduction

QSAR/QSPR approaches:

Quantitative Structure Activity Relationship (QSAR) approaches are based on the Structure-Activity Relationship (SAR) assumption, which states that there is a correlation between the structure of a molecule and its activity. This seems fair, because if it is not the structure of a molecule that invoke an effect, so what does it? We have then to identify what in the structure plays a significant role for its activity. For example, whether a specific activity is due to steric or electronic effects. It is indeed not straightforward to identify which are the features of interest related to the target activity or property.

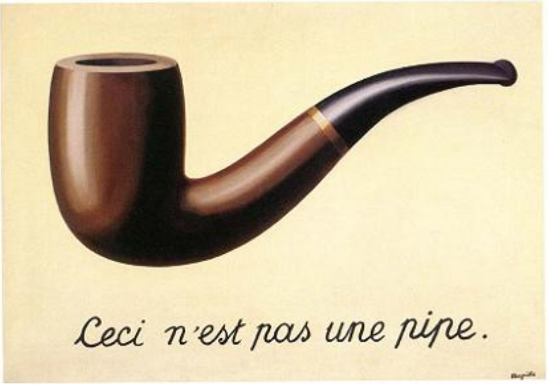


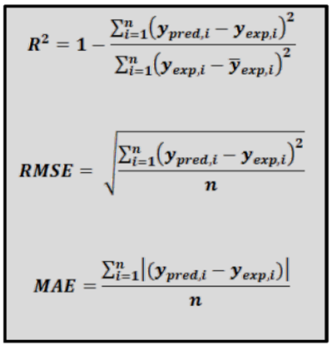
Figure 1: *the treachery of images*, René Magritte, 1929.

In a QSAR approach, one cannot directly link the structure and the corresponding activity. Indeed, neither humans nor computers understand really what a molecule is. Representations of molecules can be understood, not molecules by themselves. And depending on what kind of representation is made from a molecule, other features will be highlighted. For instance, using a Lewis representation does not provide information about chirality of molecules, but Cram representation does. This issue about representations was nicely underlined by the artist René Magritte (see **Figure 1**). On that drawing one can see a pipe with a French sentence underneath: “Ceci n’est pas une pipe”. Its English translation is “this is not a pipe”. The first reaction of people watching this illustration would be a misunderstanding, because this is clearly a pipe. But Magritte infers by this sentence that this is not a pipe but its representation. The difference is the use you will have with both objects. You obviously cannot take the representation of the pipe and smoke with it. The representation is then just the reflection of a part of the reality, but not the reality by itself. One can extend this consideration to the chemical world. Molecules are physicochemical entities and the different representation (Lewis, Cram, 3D…) we are making out of them are just a reflection of some parts of this chemical reality. The important point is that a false, biased, or incomplete representation will lead undoubtedly to false SAR. Moreover, a representation cannot take every single feature from the chemical reality into account. This is why different representations of the structural information have to be tested to screen the widest range of possible effects induced by the molecule’s features.

Then, a link between the representation and the computers world has to be introduced. This is why chemometricians developed the so-called “descriptors”. According to Prof. Todeschini and Dr. Consonni (2003), *“A molecular descriptor is the final result of a logic and mathematical procedure which transforms chemical information encoded within a symbolic representation of a molecule into a useful number or the result of some standardized experiment.”* So basically a descriptor is a number, and a molecule representation is described by a vector of descriptors.

Of course for each representation you can calculate plenty of different descriptors, some strongly correlated to each other.

To describe a whole dataset of molecules, for each molecule is calculated one vector of descriptor and all those vectors are merged a matrix of descriptors. This matrix is the description of the initial molecular dataset. Then, data mining algorithms are used on this matrix in order to extract latent knowledge in order to build prediction models. It is usual to use different algorithms because all have upsides and drawbacks. Some examples of machine learning algorithms are Artificial Neural Networks (ANN), Support Vector Machine (SVM), k-Nearest Neighbor (k-NN), Partial Least Square (PLS), Multi Linear Regression (MLR), Decisions Tree. To analyze model's accuracy, statistical parameters are calculated, comparing measured value and predicted value. Those parameters are usually determination coefficient (usually called R^2 or Q^2), Mean Average Error (MAE), Root Mean Square Error (RMSE). One can see on **Figure 2** their formulas. Determination coefficient is used to overview the correlation of either the points between each others (for Pearson's coefficient) or the points and the line following the equation predicted value = experimental value (which is the one in our case). MAE and RMSE provide information about the average error of predictions, with the difference that RMSE is more sensitive to outliers (e.g. points with a big difference between prediction and measurement).



$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_{pred,i} - y_{exp,i}\right)^2}{\sum_{i=1}^{n}\left(y_{exp,i} - \bar{y}_{exp,i}\right)^2}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(y_{pred,i} - y_{exp,i}\right)^2}{n}}$$

$$MAE = \frac{\sum_{i=1}^{n}\left|\left(y_{pred,i} - y_{exp,i}\right)\right|}{n}$$

Figure 2: Statistical parameters usually used to assess the accuracy of QSAR models.

Some issues inherent to data mining have to be considered: Frequently, experimental data contains noise, due to errors or deviations produced in the data collection phase, as a consequence of human error in translating information or due to limitations of the measurement procedures. Learning the noise will decrease the model's ability to predict true signals for new data. Since machine-learning methods use training data and try to minimize differences between experimental measurements and predicted values, they could also be influenced by the noise component and start to learn also noise. The prediction's accuracy of a model which learned the noise will decrease and this is called overfitting. To reduce such effect, validation set and test sets are usually used. One of the most common validations is the n-cross validation. The training set is split in n parts, n-1 are used to train a model and 1 part to validate it. Then another part is selected to be used for validation, and so on and so forth until all parts were used once for validation. The accuracy of the model is then calculated only using predictions for the validation part.

QNAR – Applying QSAR on nanoparticles:

In a QSAR approach applied on nanoparticles (Quantitative Nanostructure Activity Relationship QNAR), the important steps to take care of are representations and descriptions of features relevant to the targeted activity. Indeed there are no reasons why data mining methods would not work with matrices built on nanoparticles information, according to the fact that those methods are used on any kind of data (geographic, economic…), whenever there is latent information to be found. But nanoparticles do have many more specific features than usual molecules, and it is a must to take those into consideration. What are the specificities of nanoparticles?

According to the ASTM International terminology for Nanotechnology standard E2456-06, Nanoparticles are defined as "a sub-classification of ultrafine particle with lengths in two or three dimensions greater than 0.001 micrometer (1 nanometer) and smaller than about 0.1 micrometer (100 nanometers)". Nanomaterials have the structural features in between of those of atoms and the bulk materials. This is mainly due to the nanometer size of the materials which render them:

- Large fraction of surface atoms.
- High surface energy.
- Spatial confinement.
- Reduced imperfections.

Those structural features do not exist in the corresponding bulk materials. Spatial confinement effects on the materials bring quantum effects, leading to novel optical, electrical, and magnetic behaviors. In daily products, to modify or optimize surface properties (stability in solution, reactivity, selectivity), it is usual to coat them with atoms, molecules, or particles (organic functions, metal oxides, polymers ...).

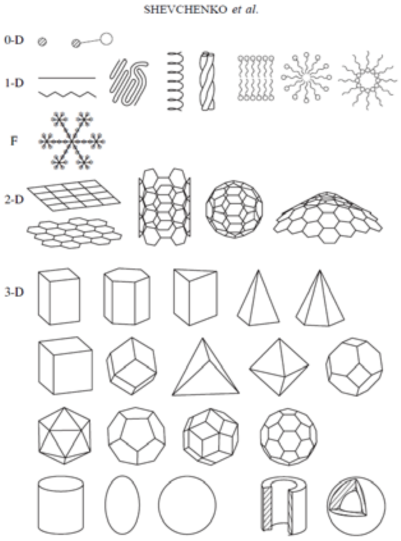


Figure 3: Diversity of the nanoworld, Shevchenko et al., 2003

Moreover, as stated by SHEVCHENKO ET AL. (2003), "*an ensemble of nanoparticles is a strongly nonequilibrium nonlinear multivariant system. There are no grounds to believe that, in the course of the evolution, this ensemble should tend to homogenization rather than to a new hierarchic order according to the self-organization principle. This suggests that the structural inhomogeneity is a fundamental property of the nanostate.*" In other terms: a system of nanoparticle is always evolving. This instability infers that there is an uncertainty whether an observed effect is due to the nanoparticle or its evolution (i.e. agglomerates or aggregates, see **Figure 4**).

Then, it was shown by OBERDORSTER ET AL. (2005) that size distribution, agglomeration state, shape, porosity, surface area, chemical composition, structure-dependent electronic configuration, surface chemistry, surface charge, and crystal structure play significant role in the properties of nanoparticles. As displayed on **Figure 3**, nanoparticles can have a lot of different shape. And for each shape of each specific composition one can have different size of particles which will have different effects.

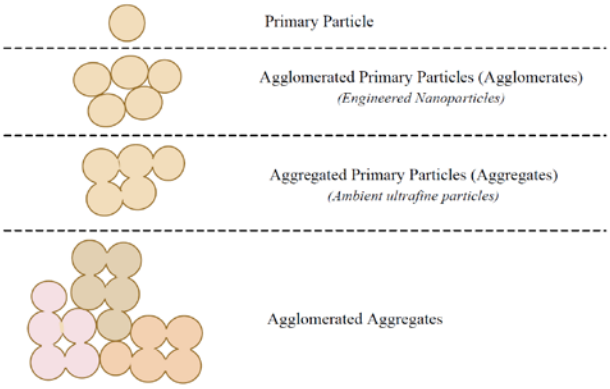

Primary Particle
Agglomerated Primary Particles (Agglomerates)
(Engineered Nanoparticles)
Aggregated Primary Particles (Aggregates)
(Ambient ultrafine particles)
Agglomerated Aggregates

Figure 4: Illustration about the possible evolution of nanoparticles.

THOMAS ET AL. (2011) developed a NanoParticle Ontology (NPO, see **Figure 5**) to meet the terminological and informatics needs in cancer nanotechnology research, such as:

- Facilitating interdisciplinary discourse among diverse research groups.
- Enabling semantic interoperability among applications and resources that store and exchange nanomaterial data.
- Providing knowledge support for data annotation in order to facilitate semantic integration, knowledge-based searching, unambiguous interpretation, mining and inference of data.

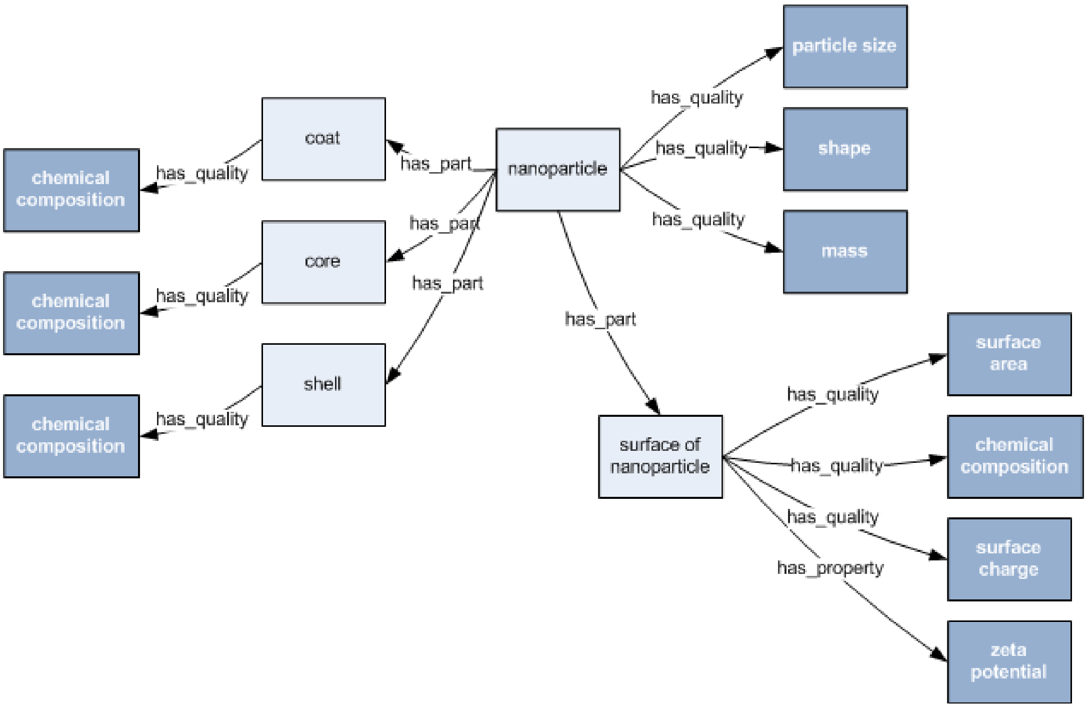


Figure 5: NanoParticle Ontology from Thomas et al., 2011

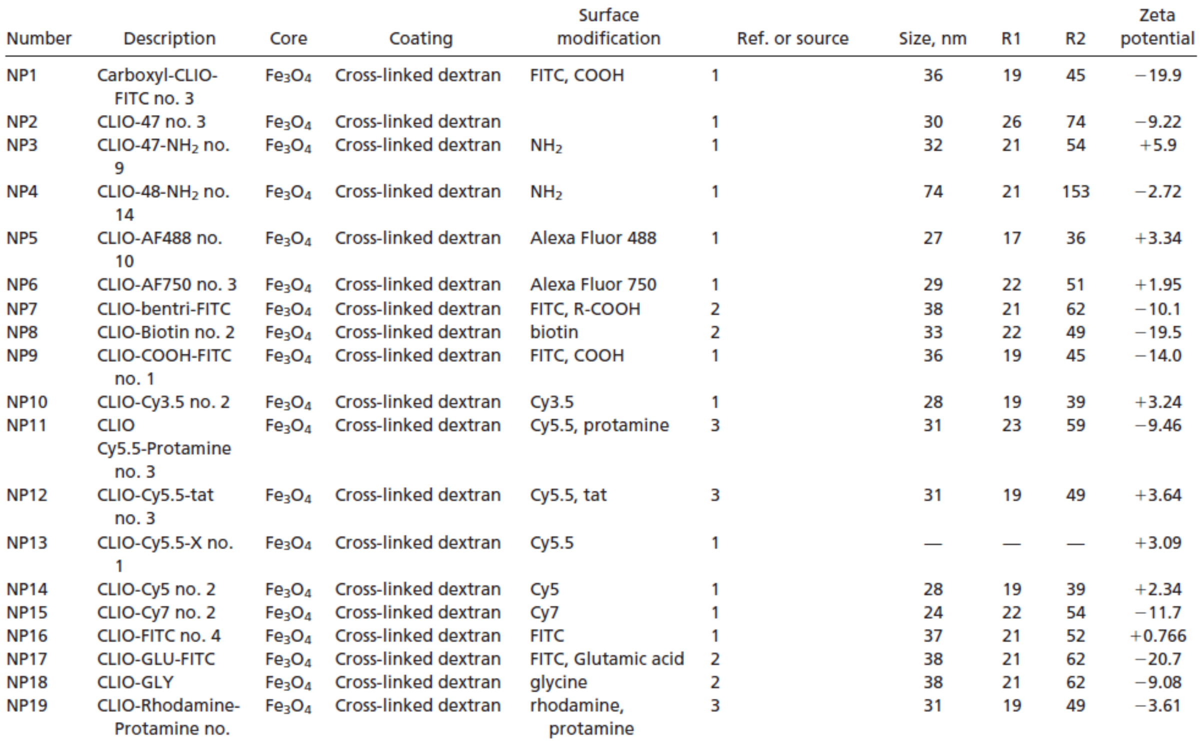
An accurate representation of nanoparticles based on this ontology require information about core's, coating's, shell's and surface's chemical composition for each nanoparticles. Besides, nanoparticles being a meta-stable equilibrium, information about the interface medium/particle and about the evolution of the particle should be provided as well. However, it is not realistic to analyze the evolution of the whole system during an experiment, which is why stable nanoparticles or the most stable form of a NP (as with isotopes for instance) are usually considered. Unfortunately, considering only stable particles is an important approximation of the reality because stability in the medium does not mean stability during the measurements, i.e. a contact between NPs and cells could modify part of the surface composition which will lead undoubtedly to a modification of interface medium/nanoparticle's surface stability.

In this NPO, an extensive description is done on NPs' surface because surface is by definition at the interface with the medium. One have to consider surface area, surface charge, and zeta potential. Zeta potential is a measure of the stability of a particle in a medium. An empirical rule is that particles within the interval [-30 mV; 30 mV] are considered as stable in the measured medium.

Moreover, specific features about the nanoparticle by itself should be considered as well, such as particle size, shape and mass. There is important issues about particle size. Indeed, size of nanoparticles can modify tremendously their properties. Usually, to provide information about size of particles, the Average Particle Size (APS) value is used. Unfortunately, one can have a mixture of nanoparticles, that are either different (in order to combine 2 specific properties) or just at different aggregation states. In such cases, APS is irrelevant, because if 2 different sizes of particles are in one medium, information about the average size is not representative of the solution's reality.

Furthermore, as NP research is in its infancies still, finding a dataset combining all those parameters is a hard task. The set with the more information we found is a set about biological activity, made by Shaw and all in 2008. Composed by 50 nanoparticles, for each is provided chemical information about core, coating and surface modification, as well as 4 experimental values: Zeta potential, relaxivities R1 and R2, and Average Particle Size (APS).

Methods



Number	Description	Core	Coating	Surface modification	Ref. or source	Size, nm	R1	R2	Zeta potential
NP1	Carboxyl-CLIO-FITC no. 3	Fe_3O_4	Cross-linked dextran	FITC, COOH	1	36	19	45	−19.9
NP2	CLIO-47 no. 3	Fe_3O_4	Cross-linked dextran		1	30	26	74	−9.22
NP3	CLIO-47-NH_2 no. 9	Fe_3O_4	Cross-linked dextran	NH_2	1	32	21	54	+5.9
NP4	CLIO-48-NH_2 no. 14	Fe_3O_4	Cross-linked dextran	NH_2	1	74	21	153	−2.72
NP5	CLIO-AF488 no. 10	Fe_3O_4	Cross-linked dextran	Alexa Fluor 488	1	27	17	36	+3.34
NP6	CLIO-AF750 no. 3	Fe_3O_4	Cross-linked dextran	Alexa Fluor 750	1	29	22	51	+1.95
NP7	CLIO-bentri-FITC	Fe_3O_4	Cross-linked dextran	FITC, R-COOH	2	38	21	62	−10.1
NP8	CLIO-Biotin no. 2	Fe_3O_4	Cross-linked dextran	biotin	2	33	22	49	−19.5
NP9	CLIO-COOH-FITC no. 1	Fe_3O_4	Cross-linked dextran	FITC, COOH	1	36	19	45	−14.0
NP10	CLIO-Cy3.5 no. 2	Fe_3O_4	Cross-linked dextran	Cy3.5	1	28	19	39	+3.24
NP11	CLIO Cy5.5-Protamine no. 3	Fe_3O_4	Cross-linked dextran	Cy5.5, protamine	3	31	23	59	−9.46
NP12	CLIO-Cy5.5-tat no. 3	Fe_3O_4	Cross-linked dextran	Cy5.5, tat	3	31	19	49	+3.64
NP13	CLIO-Cy5.5-X no. 1	Fe_3O_4	Cross-linked dextran	Cy5.5	1	—	—	—	+3.09
NP14	CLIO-Cy5 no. 2	Fe_3O_4	Cross-linked dextran	Cy5	1	28	19	39	+2.34
NP15	CLIO-Cy7 no. 2	Fe_3O_4	Cross-linked dextran	Cy7	1	24	22	54	−11.7
NP16	CLIO-FITC no. 4	Fe_3O_4	Cross-linked dextran	FITC	1	37	21	52	+0.766
NP17	CLIO-GLU-FITC	Fe_3O_4	Cross-linked dextran	FITC, Glutamic acid	2	38	21	62	−20.7
NP18	CLIO-GLY	Fe_3O_4	Cross-linked dextran	glycine	2	38	21	62	−9.08
NP19	CLIO-Rhodamine-Protamine no.	Fe_3O_4	Cross-linked dextran	rhodamine, protamine	3	31	19	49	−3.61

Figure 6: Part of the data table from Shaw and all (2008), which includes information about cores, coating and surface modification

Work approach

Denis Fourches proved in 2010 on Shaw's dataset that QNAR is a valid approach. However, he is not using the intrinsic structure of nanoparticles, but only 4 experimental measurements: APS, longitudinal relaxivity R1, transverse relaxivity R2, and zeta potential (see **Figure 6**) after clustering the set in 3. So the basic idea of my work about QNAR is to prove that describing nanoparticles using the approach of the NanoParticle Ontology would provide results that are significantly better than those without using it. Besides, identifying which features are relevant for which part of the particle could be a plus. Only few papers apply QSAR approach on nanoparticles, and almost none using complex nanoparticles (e.g. with cores, coatings, and surface modifications).

Out of that I formulated my aim being to use the same dataset, represent it according to the NPO (see **Figure 5**), derive descriptors from such representation, and prove that such approach improves significantly the accuracy of prediction compared to using only experimental values. Also allowing for extrapolation, as semi-process or mechanistic knowledge is the basis for formulating the descriptors. The limits are that the set we are using which is the only one allowing our approach is a set of 50 nanoparticles which is few, but the scientific "world" of nanoparticles being recent there will most likely be more and more data available in the future to come. Indeed, many tests are currently done which will increase the knowledge we have about nanoparticles.

Describing separately the core, coating, and surface modification with usual descriptors on Shaw's dataset will lead to an unbalanced matrix (50 rows x more than 300 columns). Such situations usually facilitate overfitting issues. Moreover, splitting such set into 3 (training, test and validation sets) is not doable either, because too few information would be within the training set. This is why all calculated models are cross validated using a 5-Cross Validation and not tested afterwards. This is a good compromise between overfitting data and having too few knowledge to train a model.

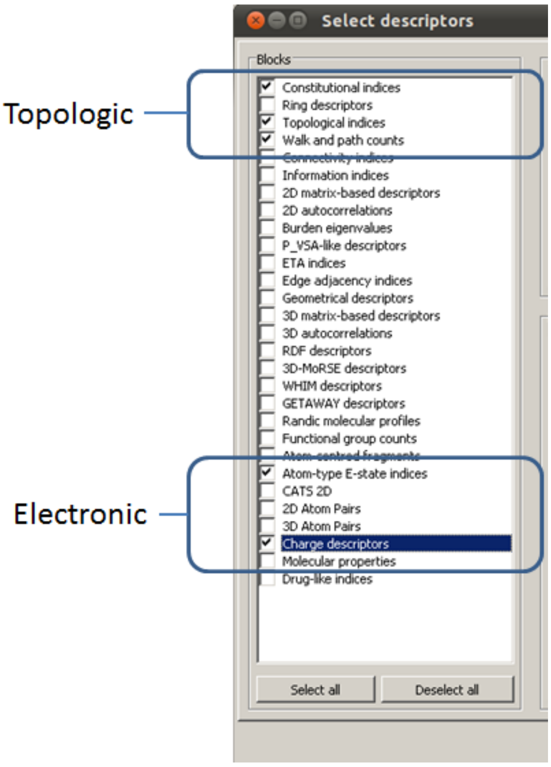
Databases

To describe accurately all the inherent parts of a nanoparticle, information about intern constitution of NPs is necessary. The only set found reaching our expectations (see **Figure 6**) was the one used by SHAW ET AL. (2008) and FOURCHES ET AL. (2010). There was indeed information about core, coating, surface modification, and experimental data. Then I translated manually all those information into SMILES. For that both websites of chemical producers and chemspider (http://www.chemspider.com/) were used. Some surface modification (i.e. Alexa Fluor 750) did not have a known structure, so instead of writing SMILES code, blank was let. For such particles, the descriptors calculated on such missing parts would have value of 0. I decided to build models using and not using particles with such missing information, to identify whether a lack of information could still allow building valid models.

This data set includes 50 nanoparticles with 2 different cores, 5 different coatings, 17 surface modifications, and 4 different experimental values. Its endpoint is biological activity of a nanomaterial assessed by multiple physiologic cell-based assays in multiple cell types, and at multiple doses. The biological activity values are used by FOURCHES AND ALL (2010).

Description

Different ways of describing each part of nanoparticles were tested: using atomic and molecular descriptors (electronical, topological, and those from the Chemistry Development Kit CDK), energy values, and random numbers (to check whether the data mining algorithm needs real information about the composition or only need to differentiate categories). Moreover, each description was not tested on each part of the particle. For instance, after preliminary calculations I noticed that information about the core can improve a model but any kind of description can be used. Indeed, even a random description performs as good as accurate description. This is probably due to the fact that there are only 2 different cores and data mining methods only need to differentiate them. This is why I considered that it was not necessary to calculate models with 3 different kind of molecular descriptors. Moreover, we did not calculate any 3D

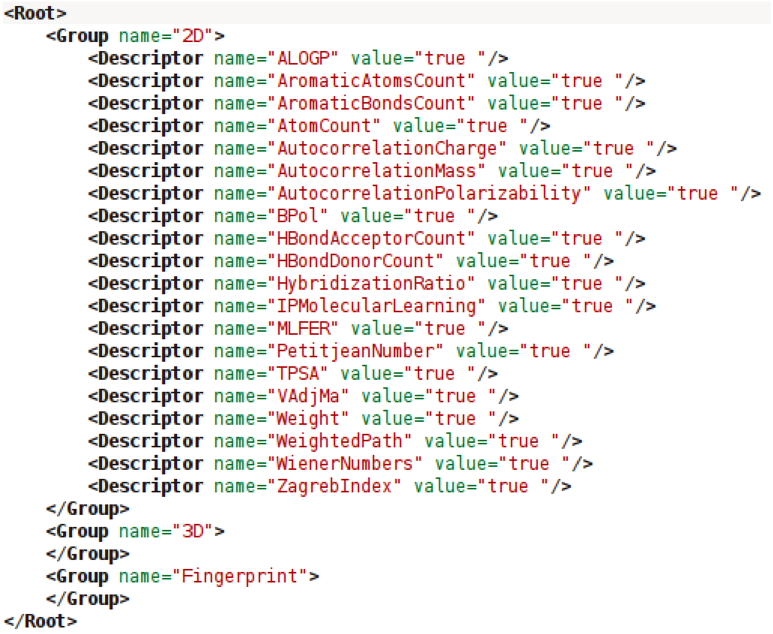


Blocks

☑ Constitutional indices
☐ Ring descriptors
☑ Topological indices
☑ Walk and path counts
☐ Connectivity indices
☐ Information indices
☐ 2D matrix-based descriptors
☐ 2D autocorrelations
☐ Burden eigenvalues
☐ P_VSA-like descriptors
☐ ETA indices
☐ Edge adjacency indices
☐ Geometrical descriptors
☐ 3D matrix-based descriptors
☐ 3D autocorrelations
☐ RDF descriptors
☐ 3D-MoRSE descriptors
☐ WHIM descriptors
☐ GETAWAY descriptors
☐ Randic molecular profiles
☐ Functional group counts
☐ Atom-centred fragments
☑ Atom-type E-state indices
☐ CATS 2D
☐ 2D Atom Pairs
☐ 3D Atom Pairs
☑ Charge descriptors
☐ Molecular properties
☐ Drug-like indices

Figure 7: Used Dragon's descriptors

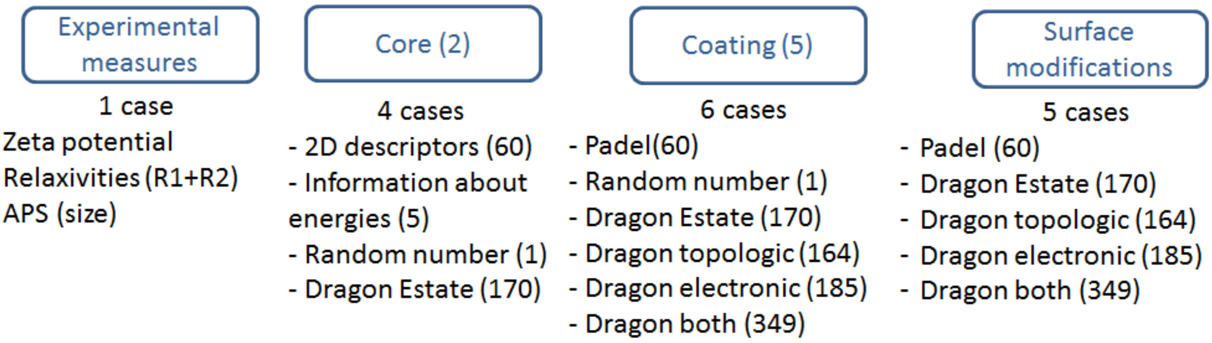
descriptors. Those descriptors need obviously a 3D structure to be calculated. To obtain such structure one can use experimental data like X-ray spectrum derived structure (that are not available for the particles of interest) or to do a 3D optimization using molecular modeling methods. However, molecular structures optimized using force fields methods should not be used here because they would not be related to the structural reality about the nanoparticle.



```
<Root>
    <Group name="2D">
        <Descriptor name="ALOGP" value="true "/>
        <Descriptor name="AromaticAtomsCount" value="true "/>
        <Descriptor name="AromaticBondsCount" value="true "/>
        <Descriptor name="AtomCount" value="true "/>
        <Descriptor name="AutocorrelationCharge" value="true "/>
        <Descriptor name="AutocorrelationMass" value="true "/>
        <Descriptor name="AutocorrelationPolarizability" value="true "/>
        <Descriptor name="BPol" value="true "/>
        <Descriptor name="HBondAcceptorCount" value="true "/>
        <Descriptor name="HBondDonorCount" value="true "/>
        <Descriptor name="HybridizationRatio" value="true "/>
        <Descriptor name="IPMolecularLearning" value="true "/>
        <Descriptor name="MLFER" value="true "/>
        <Descriptor name="PetitjeanNumber" value="true "/>
        <Descriptor name="TPSA" value="true "/>
        <Descriptor name="VAdjMa" value="true "/>
        <Descriptor name="Weight" value="true "/>
        <Descriptor name="WeightedPath" value="true "/>
        <Descriptor name="WienerNumbers" value="true "/>
        <Descriptor name="ZagrebIndex" value="true "/>
    </Group>
    <Group name="3D">
    </Group>
    <Group name="Fingerprint">
    </Group>
</Root>
```

Figure 8: Padel descriptors that were used

Dragon and Padel descriptor software were used to calculate descriptors. For topological descriptors from Dragon were included all the constitution, topological and walking path (see **Figure 7**) descriptors, which includes an amount of 164. About electronical descriptors, Estate and charges descriptors (see **Figure 7**) were integrated, which leads to 185 different ones. Hence, both descriptions (electronical and topological) were merged to obtain a 349 descriptors matrix in order to check whether both information were needed to represent accurately the molecule.



Experimental measures
1 case
Zeta potential
Relaxivities (R1+R2)
APS (size)

Core (2)
4 cases
- 2D descriptors (60)
- Information about energies (5)
- Random number (1)
- Dragon Estate (170)

Coating (5)
6 cases
- Padel(60)
- Random number (1)
- Dragon Estate (170)
- Dragon topologic (164)
- Dragon electronic (185)
- Dragon both (349)

Surface modifications
5 cases
- Padel (60)
- Dragon Estate (170)
- Dragon topologic (164)
- Dragon electronic (185)
- Dragon both (349)

Figure 9: Scheme representing which description was performed on which part of a nanoparticle.

With Padel software 61 descriptors in 2 dimensions from the Chemistry Development Kit (CDK) were calculated (see **Figure 8**). 3D descriptors were not calculated because of the reasons explained before. Then, on cores and coatings were tested a random description (core A = 1, core B = 2). Moreover, on cores some energy data were used as descriptors. Those are Dipole momentum, Electron energy, Nuclease repulsion, and binding energy. You can see on **Figure 9** which descriptors were calculated and used for which part of each nanoparticle. Out of those descriptors, several combinations were tested. With and without experimental data, core, coating, and surface modification. And for each combination, all the different ways of describing data were tested, which leads to 149 different sets of descriptors. The different combinations are illustrated on **Figure 10**.

Data Mining Method

To mine all the 149 descriptor sets, the open source software Weka developed by the university of Waikato (http://www.cs.waikato.ac.nz/ml/weka/) was used. It was considered accurate (open source) and handy because of the possibility to launch calculations using bash scripts.

Before any kind of calculations, the data were filtered by removing nominal values (as name of particles), removing useless values (the same value for each particle), and normalizing the descriptors between 0 and 1. All this filtering is unsupervised. Using supervised methods are not fair because this is a first influence of the property about the dataset. It already bring some knowledge about the property and can introduce overfitting that would not be probed by cross validation techniques but only by an external validation set, which we do not have.

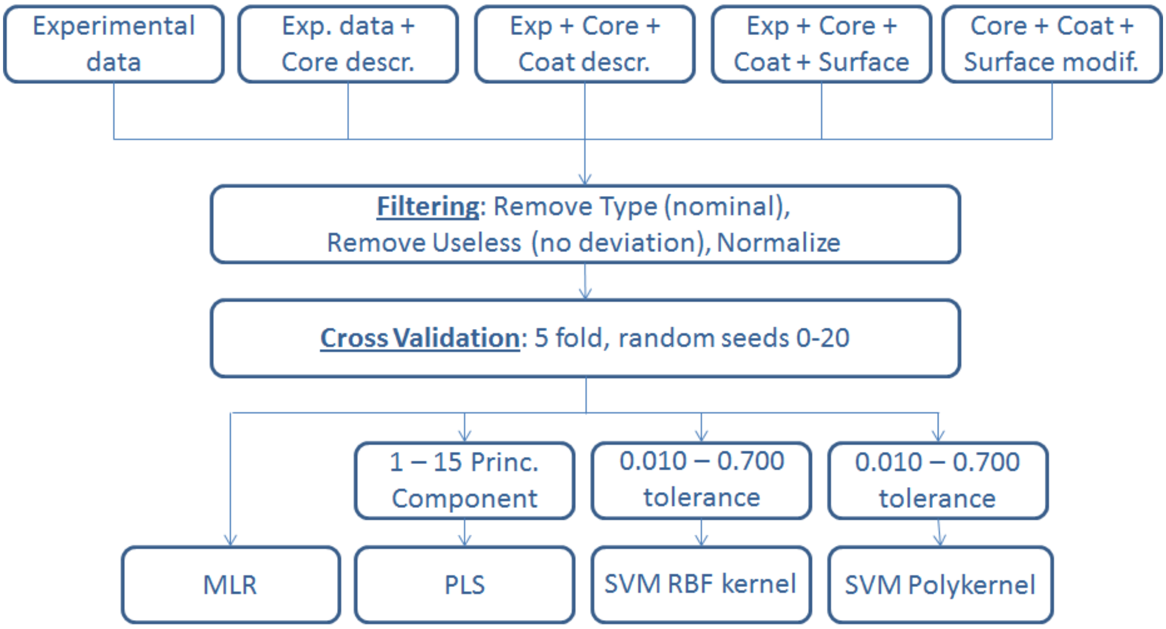

Experimental data
Exp. data + Core descr.
Exp + Core + Coat descr.
Exp + Core + Coat + Surface
Core + Coat + Surface modif.
Filtering: Remove Type (nominal), Remove Useless (no deviation), Normalize
Cross Validation: 5 fold, random seeds 0-20
1 – 15 Princ. Component
0.010 – 0.700 tolerance
0.010 – 0.700 tolerance
MLR
PLS
SVM RBF kernel
SVM Polykernel

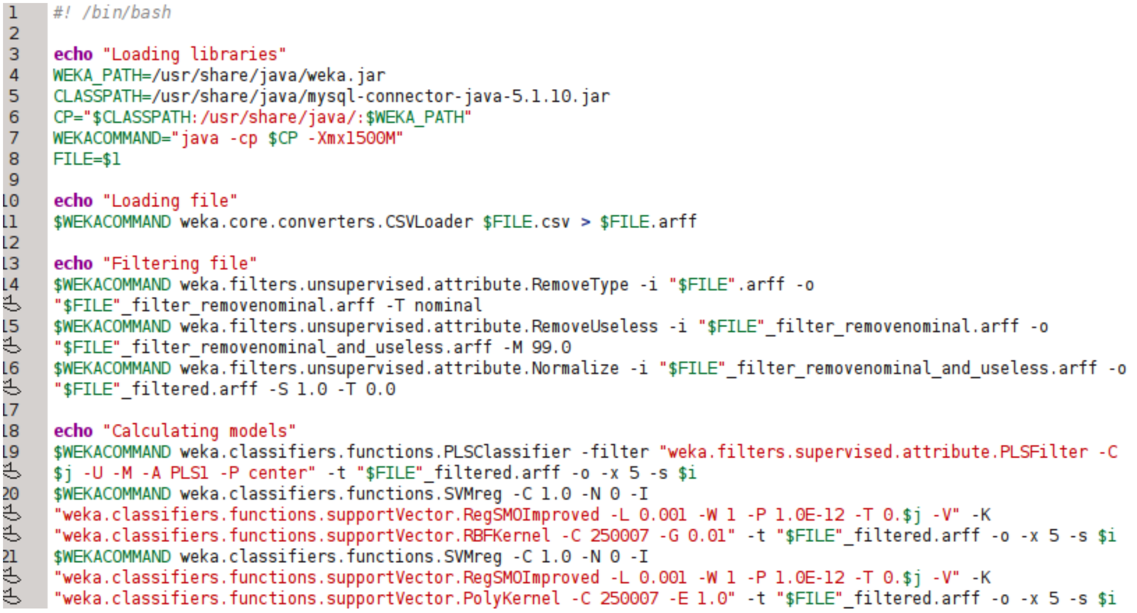
Figure 10: Scheme of the data mining workflow that was used to build all the prediction model using weka.

Several data mining methods were evaluated such as Artificial Neural Networks (ANN), Multi Linear Regression (MLR), Partial Least square (PLS), and Support Vector Machine (SVM). The dataset being too small, ANN failed dramatically. Indeed, ANN is highly prone to overfitting, so usually half of a data set is used to train the model and half to identify when the learning should be stopped. But if there are few data, molecules from the training set will not be representative enough of the covered chemical space, the validation set will be too different from the training set and this will bias the early stopping point in the learning process. This is why using neural networks on small dataset usually fails during the validation procedure (here the cross fold validation). Then, MLR was not accurate enough (probably due to the high correlation between some descriptors, and/or the unbalanced ratio lines/columns). On the contrary, PLS is an improved MLR that deals better with inter-correlated descriptors. Indeed, first a Principal Component Analysis (PCA) is operated, which groups correlated descriptors according to the amount of Principal Components that was chosen and then a MLR is operated on those Principal Components. That is why a focus on PLS and SVM was chosen. Besides, 2 different kernels for the SVM were tested: Radial Based Function (RBF) and the PolyKernel.

For each data mining method, specific parameters were modified. For PLS, the amount of principal components was modified from 1 to 15. For both SVMs, the tolerance parameter was modified from 0.010 to 0.700 (because a convergence was observed around 650) by step of 0.012. Moreover, each model cross validated using 5 folds. To lower the chance impact on the selection of each fold, different seeds for random numbers were used. Seeds were used for each data mining method from 0 to 20 by a step of 1 (e.g. 21 different seeds per method). An illustration of all this parameters incrementing is shown on **Figure 10**.

Which means that on each of the 149 descriptors sets, 3 data mining methods were tested each with respectively 15, 66 and 66 different parameters, and each of those were validated by 21 different cross validations. Which leads us to more than 1 billion calculated models.

Hereafter can be found a screenshot of the bash script summoning Weka. “$i” is for the different seeds and “$j” for the different parameters, such as amount of principal components for PLS.



```
#! /bin/bash

echo "Loading libraries"
WEKA_PATH=/usr/share/java/weka.jar
CLASSPATH=/usr/share/java/mysql-connector-java-5.1.10.jar
CP="$CLASSPATH:/usr/share/java/:$WEKA_PATH"
WEKACOMMAND="java -cp $CP -Xmx1500M"
FILE=$1

echo "Loading file"
$WEKACOMMAND weka.core.converters.CSVLoader $FILE.csv > $FILE.arff

echo "Filtering file"
$WEKACOMMAND weka.filters.unsupervised.attribute.RemoveType -i "$FILE".arff -o
"$FILE"_filter_removenominal.arff -T nominal
$WEKACOMMAND weka.filters.unsupervised.attribute.RemoveUseless -i "$FILE"_filter_removenominal.arff -o
"$FILE"_filter_removenominal_and_useless.arff -M 99.0
$WEKACOMMAND weka.filters.unsupervised.attribute.Normalize -i "$FILE"_filter_removenominal_and_useless.arff -o
"$FILE"_filtered.arff -S 1.0 -T 0.0

echo "Calculating models"
$WEKACOMMAND weka.classifiers.functions.PLSClassifier -filter "weka.filters.supervised.attribute.PLSFilter -C
$j -U -M -A PLS1 -P center" -t "$FILE"_filtered.arff -o -x 5 -s $i
$WEKACOMMAND weka.classifiers.functions.SVMreg -C 1.0 -N 0 -I
"weka.classifiers.functions.supportVector.RegSMOImproved -L 0.001 -W 1 -P 1.0E-12 -T 0.$j -V" -K
"weka.classifiers.functions.supportVector.RBFKernel -C 250007 -G 0.01" -t "$FILE"_filtered.arff -o -x 5 -s $i
$WEKACOMMAND weka.classifiers.functions.SVMreg -C 1.0 -N 0 -I
"weka.classifiers.functions.supportVector.RegSMOImproved -L 0.001 -W 1 -P 1.0E-12 -T 0.$j -V" -K
"weka.classifiers.functions.supportVector.PolyKernel -C 250007 -E 1.0" -t "$FILE"_filtered.arff -o -x 5 -s $i
```

Results

Analyzing the results

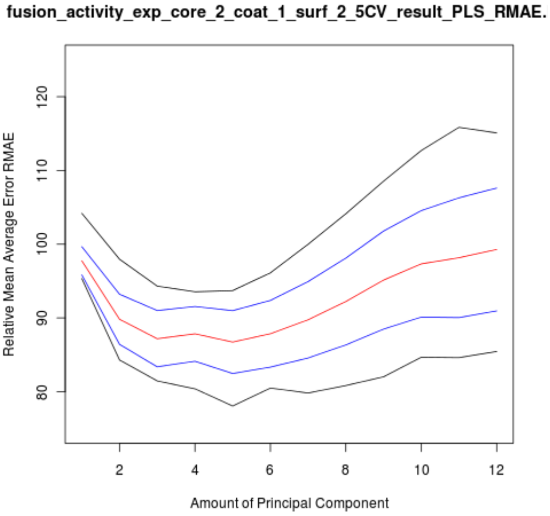
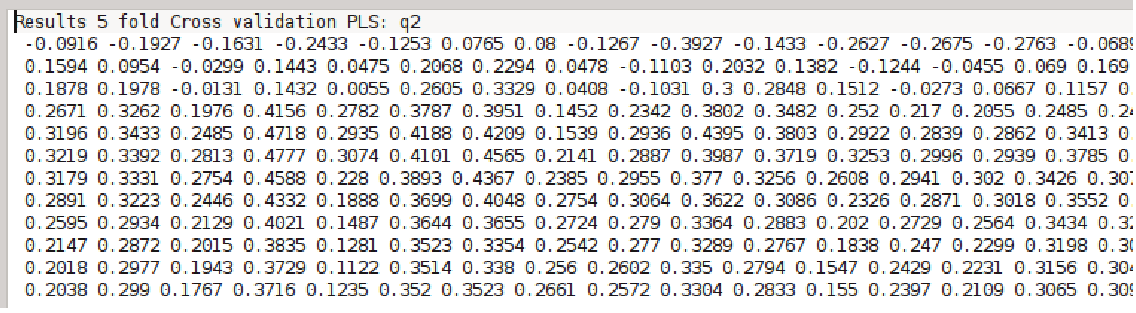


Figure 11: Representative graph of the data mining method PLS. As abscissa the amount of principal components, as ordinate the Relative MAE (MAE in %). Red curve is the mean of results using the same PC but different seeds. Blue curves are mean + and – the standard deviation of those results. Black curves are the minimum and maximum of those results.

The aim of our study is to identify whether one description is better than another. For instance, should the surface modification be described by topologic or electronic descriptors. To answer this question we have to compare prediction's accuracies of both approaches and check whether there is significant difference or not. Furthermore, this difference should be for one general way of describing particles, not for particular cases. Indeed it is not representative to say that topological descriptors are better to describe nanoparticles than electronical descriptors in the case of PLS algorithm using 5 principal components and splitting the validations fold with a seed of value 16. To prove that one description is in average better than another we compare two files containing the values of the same statistical parameter (Q2, MAE, or RMSE) and using the same method (PLS, SVM RBF or SVM polykernel). On **Figure 11** is a representative graph for PLS. One can then easily understand that to compare two descriptions using the same data mining methods (such as PLS or SVM) one has to pair the similar parameters (for instance the amount of principal components or the tolerance value). Undeniably there is no point for comparing the accuracy of PLS using 14 components with the PLS using 2 components. On **Figure 12** one can see a table with the Q2 values of PLS method for one specific description. The lines represent the different components (line 3 = 2 principal components, line 6 = 5 PC) and the columns the different seeds (from 0 to 20). Each value is the value of the Q2 of the model calculated with the corresponding parameters. We compared files to files, pairing the mining parameters.



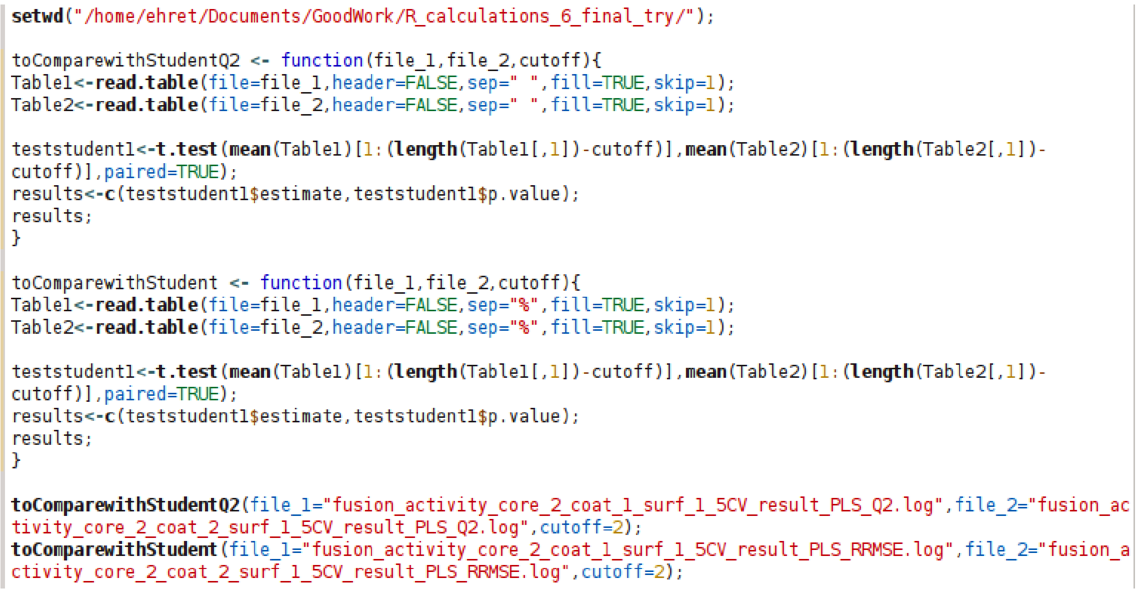
```
Results 5 fold Cross validation PLS: q2
 -0.0916 -0.1927 -0.1631 -0.2433 -0.1253 0.0765 0.08 -0.1267 -0.3927 -0.1433 -0.2627 -0.2675 -0.2763 -0.068
 0.1594 0.0954 -0.0299 0.1443 0.0475 0.2068 0.2294 0.0478 -0.1103 0.2032 0.1382 -0.1244 -0.0455 0.069 0.169
 0.1878 0.1978 -0.0131 0.1432 0.0055 0.2605 0.3329 0.0408 -0.1031 0.3 0.2848 0.1512 -0.0273 0.0667 0.1157 0
 0.2671 0.3262 0.1976 0.4156 0.2782 0.3787 0.3951 0.1452 0.2342 0.3802 0.3482 0.252 0.217 0.2055 0.2485 0.2
 0.3196 0.3433 0.2485 0.4718 0.2935 0.4188 0.4209 0.1539 0.2936 0.4395 0.3803 0.2922 0.2839 0.2862 0.3413 0
 0.3219 0.3392 0.2813 0.4777 0.3074 0.4101 0.4565 0.2141 0.2887 0.3987 0.3719 0.3253 0.2996 0.2939 0.3785 0
 0.3179 0.3331 0.2754 0.4588 0.228 0.3893 0.4367 0.2385 0.2955 0.377 0.3256 0.2608 0.2941 0.302 0.3426 0.30
 0.2891 0.3223 0.2446 0.4332 0.1888 0.3699 0.4048 0.2754 0.3064 0.3622 0.3086 0.2326 0.2871 0.3018 0.3552 0
 0.2595 0.2934 0.2129 0.4021 0.1487 0.3644 0.3655 0.2724 0.279 0.3364 0.2883 0.202 0.2729 0.2564 0.3434 0.3
 0.2147 0.2872 0.2015 0.3835 0.1281 0.3523 0.3354 0.2542 0.277 0.3289 0.2767 0.1838 0.247 0.2299 0.3198 0.3
 0.2018 0.2977 0.1943 0.3729 0.1122 0.3514 0.338 0.256 0.2602 0.335 0.2794 0.1547 0.2429 0.2231 0.3156 0.30
 0.2038 0.299 0.1767 0.3716 0.1235 0.352 0.3523 0.2661 0.2572 0.3304 0.2833 0.155 0.2397 0.2109 0.3065 0.30
```

Figure 12: Table grouping Q2 values for models built for one description and one method (PLS), using different mining parameters (lines) and seeds (columns).

Regarding the amount of models, analyzing each of them by hand is not possible. So the power of informatics had to be used. The open source software R (http://www.r-project.org/) is perfectly designed to perform the analysis we intended to do. Besides the possibility to run bash scripts calling R is really handy to treat large amount of data.

To identify whether one description is better than another, a first overview can be done by a glimpse to both graphs drawn using the statistical parameters values. For instance comparing the Q2 of description's regression using core 1 and core 2. But then, to assess that there is a statistically significant difference (which means that the difference cannot happen by chance), more advanced tests should be done. In our case, we compare the average values for each method parameters (average of each difference seeds for 1 method parameter, such as the amount of principal components) paired to the equivalent average for another description. For instance comparing the average of the 21 values calculated using different seeds with the method parameter "3 Principal Components" with dataset 1, paired to the average of values calculated with method parameter "3 Principal Components" with dataset 2.

Those averages values being the reflection of tries to predict one reality, one can assess that they are following a normal distribution. Besides, there are less than 30 values that are treated. Those two reasons explain why the significancy test we used is the Student t-test. Two values from this test are of interest: the p-value (if p < 0.05, there is a significant difference) and the difference of statistical mean (to identify which method is better). Moreover, this difference is tested using as much results as possible. It is not because description 1 is better than description 2 once that it will be the case every time. This is why for instance to analyze the usefulness of experimental values we compared 12 different methods, each with the 3 different statistical parameters.



```
setwd("/home/ehret/Documents/GoodWork/R_calculations_6_final_try/");

toComparewithStudentQ2 <- function(file_1,file_2,cutoff){
Table1<-read.table(file=file_1,header=FALSE,sep=" ",fill=TRUE,skip=1);
Table2<-read.table(file=file_2,header=FALSE,sep=" ",fill=TRUE,skip=1);

teststudent1<-t.test(mean(Table1)[1:(length(Table1[,1])-cutoff)],mean(Table2)[1:(length(Table2[,1])-
cutoff)],paired=TRUE);
results<-c(teststudent1$estimate,teststudent1$p.value);
results;
}

toComparewithStudent <- function(file_1,file_2,cutoff){
Table1<-read.table(file=file_1,header=FALSE,sep="%",fill=TRUE,skip=1);
Table2<-read.table(file=file_2,header=FALSE,sep="%",fill=TRUE,skip=1);

teststudent1<-t.test(mean(Table1)[1:(length(Table1[,1])-cutoff)],mean(Table2)[1:(length(Table2[,1])-
cutoff)],paired=TRUE);
results<-c(teststudent1$estimate,teststudent1$p.value);
results;
}

toComparewithStudentQ2(file_1="fusion_activity_core_2_coat_1_surf_1_5CV_result_PLS_Q2.log",file_2="fusion_ac
tivity_core_2_coat_2_surf_1_5CV_result_PLS_Q2.log",cutoff=2);
toComparewithStudent(file_1="fusion_activity_core_2_coat_1_surf_1_5CV_result_PLS_RRMSE.log",file_2="fusion_a
ctivity_core_2_coat_2_surf_1_5CV_result_PLS_RRMSE.log",cutoff=2);
```

Figure 13: R-code to operate Student t-tests. As parameters for the comparing functions are the two files to compare, and the cutoff to use to take of outliers results (when data mining parameters came close to the extremums). Cutoff were the same for all PLS (2 last components out of 15) and all SVMs (20 last tolerances out of 100)

Moreover, when getting close to high values for parameters (14-15 principal components for PLS, after 0.550 of tolerance for SVM), specific behaviors were observed. For PLS, big amount of principal components prone models to overfitting with which leads to a tremendous decreasing of the predictive accuracy. Besides, for both SVMs a convergence was observed. To avoid that such outlier results would be taken into consideration for the Student test, some cutoff was introduced. The R code of this function is available on **Figure 13**.

Issue with the dataset

As already mentioned, some structure of surface modifications were not available (not known or not published by the company which is making them). So we calculated models using all particles and using only particles we fully knew. For instance models using or not NP6, because the surface modification is the dye Alexa Fluor 750 and its structure is not known. Excluded particles because of their lack of information were: NP6, NP22, NP23, NP33, NP37. (number corresponding to the IDs in Fourches data set)

On the contrary of our expectations, models using particles with structures those were not fully available (i.e with Alexa Fluor 750 as surface modification) performed better than models with less particles but full information. We first thought it would be because the model overfits more when learning from a set including only particles which structures were fully known, but after looking at the training results (models not cross validated), the set with missing information was still reaching higher accuracy. We then postulated that it was due to the lack of particles, and if we had more particles accuracy of models would be far better. Somehow the loss of information resulting by taking of not fully describe particles is higher than the perturbation of the models due to lack of information.

Experimental data

With - without		Is Significant out of 18	Sign if significant difference
PLS	Q2	18	-
	MAE	18	+
	RMSE	18	+
SVM	Q2	6	
	MAE	6	
	RMSE	6	
SVMK2	Q2	18	-
	MAE	17	+
	RMSE	12	+

Table 1: Comparing accuracies of models using or not experimental data.

Experimental data were Average Particle Size (APS), longitudinal relaxivity, transverse relaxivity, and zeta potential. We aimed to verify whether there is a significant difference between using or not experimental data.

For such purposes, we first calculated the accuracy of prediction using only experimental data. Such models performed really badly (Q2 stable around 0.1 for PLS and 0 for SVM, RMAE stable around 95% for PLS, 100% for SVM Polykernel and 120% for SVM RBF). Whenever some information was added, the prediction's accuracy increased (for instance using coating and surface description, RMAE for PLS drops to 85% for 5 components and RMAE for SVM RBF is stable with an RMAE of 95%).

Then we compared different models with as only difference in the description set was using or not experimental values. For instance dataset 1 being using “core 3” (random numbers) + “coat 1” (CDK descriptors) + “surface modification 3” (topologic and electronic descriptors) and dataset 2 being using “experimental value” + “core 3” + “coat 1” + “surface modification 3”. On **Table 1**, one can see the results of such test. In red the cases were the amount of significant (according to paired Student t-tests) differences was judged as significant using binomial test (e.g. when more than 13 out of 18 differences were significant). To resume, this table groups the amount of models having a significant difference between using or not experimental data, and in red the cases were this amount was significant according to the total number of tested cases.

There is a significant difference between using and not using experimental data for PLS and SVM Polykernel (SVMK2) methods. For Q2, RMAE and RRMSE, there are 16 to 18 out of 18 significant differences. SVM RBF did not have enough significant differences (6 out of 18 for each statistical parameters), so we cannot state about the importance of experimental data for this method.

We expected to have better results while using experimental values, but the contrary was observed. Not using experimental measurements provides better results. That can be due to 2 things: Either it is not necessary to use it (even if it is hard to believe) or experimental values are somehow wrong. We think that the second option caused that result. Indeed, APS is not relevant because as explained in the introduction: if there are not only 1 size of particle but two, the average would not describe the particle correctly. For instance, an APS of 50nm could describe a suspension of 50nm size particles or a mixture ratio 1:1 of particles having a size of 30nm and 70nm. Indeed for NP26-30 the size (according to Shaw’s paper) is in between 20 and 60 nm. Fourches used the average (40nm). Moreover, for NP from 26 to 44, there are imprecision about the relaxivities (value is literally <0.5 in Shaw’s set, and Fourches used 0.5 as value for his models).

This is why we observed better results for some data mining methods while not using experimental data.

Core

Core 2 – core 3		Is Significant out of 18	sign
PLS	Q2	17	-
	MAE	15	+
	RMSE	12	+
SVM	Q2	10	-
	MAE	6	+
	RMSE	6	+
SVMK2	Q2	11	-
	MAE	11	+

	RMSE	8	+

Table 2: Comparing accuracies of models based on different cores' description.

The different cores are: core 1 with CDK 2D descriptors, core 2 with energies, core 3 with random numbers (1 and 2), core 4 with Estate descriptors calculated using Dragon.

First we identified that it was necessary to describe the core of nanoparticles (significant improvement if accurate description). Then, depending on the data mining method, different descriptions performed better than others. But it was observed that neither describing using lot or few descriptors nor describing using random numbers (1 for core 1 and 2 for core 2) provided consistently significant differences. **Table 2** groups the comparison between describing with energies and with random numbers. In red the cases were the amount of significant (according to paired Student t-tests) differences was judged as significant using binomial test (e.g. when more than 13 out of 18 differences were significant).

One can see from this table that describing cores does not worsen accuracy, and does improve it in some cases, but only random description seems already sufficient to have the effects of it. It is also highlighted that for some method (here PLS) using more than random numbers can improve some (Q2 and RMAE) statistical parameters.

Coating

Coat 4, 5, 6.		Significance (amount of significative success out 12)			If significant, sign		
		4 - 5	4 - 6	5 - 6	4 - 5	4 - 6	5 - 6
PLS	Q2	0	2	1			
	MAE	2	8	4			
	RMSE	12	11	12	-	+	+
SVM	Q2	12	0	12	+		-
	MAE	12	1	12	-		+
	RMSE	9	0	11			+
SVMK2	Q2	5	1	2			
	MAE	8	0	10			-
	RMSE	3	1	4			

Table 3: Comparing accuracies of models based on different coatings' description

The different coatings are: coat 1 with CDK 2D descriptors, coat 2 with random numbers (1 to 5), coat 3 with Estate descriptors calculated using Dragon, coat 4 with topologic descriptors, coat 5 with electronic descriptors, coat 6 with both electronic and topologic descriptors.

By early calculations we saw that coatings should better be described using proper descriptors and not only random numbers as we did for cores. **Table 3** groups comparisons between different coatings' descriptions: Using topological (4), electronical (5), and both topological and electronical descriptors (6). In red the cases were the amount of significant differences was judged as significant using binomial test (e.g. when more than 10 out of 12 differences were significant).

According to that table, there is no significant difference between using topological only or all descriptors. Then, there is a notable difference when using only electronical descriptors. This table shows that adding electronical description improve models accuracy but the most important features are to describe topologically the different coatings.

Surface modification

Surf 3, 4, 5.		Significance (amount of significative success out 12)			If significant, sign		
		3 - 4	3 - 5	4 - 5	4 - 5	4 - 6	5 - 6
PLS	Q2	0	0	0			
	MAE	0	6	0			
	RMSE	0	1	0			
SVM	Q2	12	12	0	-	-	
	MAE	12	12	0	+	+	
	RMSE	12	12	3	+	+	
SVMK2	Q2	0	2	3			
	MAE	0	0	0			
	RMSE	0	0	0			

Table 4: Comparing accuracies of models based on different surface modifications' description.

The different surface modifications that were used are: surf1 with CDK 2D descriptors, surf 2 with Estate descriptors calculated using Dragon, surf 3 with topologic descriptors, surf 4 with electronic descriptors, coat 6 with both electronic and topologic descriptors.

Early calculations revealed that surface modifications was the most important feature that should be described. The most significant improvement were achieved using data about surface modification description. Then we tried to identify whether a specific feature for this should be described. For that extent we compared topological only, electronical only and both ways of description. The **Table 4** groups the results for that comparison. In red the cases were the amount of significant differences was judged as significant using binomial test (e.g. when more than 10 out of 12 differences were significant).

One can see that description does matter only using RBF based SVM, and that it should include electronical descriptors. Which means that description about surface modifications is necessary, but we cannot yet conclude about which kind of descriptors should be used.

Conclusion

In this work, our aim was to show that QNAR is indeed a valid approach for nanoparticles' activity predictions, and that using a description as shown in the NPO provides better results than already tested QNAR (so just using experimental measurements). Nanoparticles research is in its infancies still, and large amount of data are rare which is a problem for QNAR. But we can already mention that despite the amount of data we showed that QNPR is a valid approach for NPs and describing NPs as in the NPO provides significantly better results than the other attempts.

During our study, we showed that experimentally measured property should be relevant to be used in QNAR but accuracy of those data is of real importance. Uncertainty about size for instance lowers tremendously accuracy of prediction. A system has to be correctly represented by such values (for instance APS is not relevant enough, but size distribution could be). Furthermore, describing all the constituent (core, coating, surface modification) of nanoparticles increases drastically the prediction's accuracy of models. Even though we cannot say yet which kind of description should be done, we proved that molecular descriptors can describe accurately coatings and surface modification. Our dataset grouping only 2 kind of cores, we cannot state whether molecular descriptors, simple energies, or other description on cores should be used. Cores being often metallic with crystalline organization, we thought it could be relevant to use group theory based descriptors (providing information about molecular symmetry). Besides, we confirmed that the NanoParticle Ontology approach is a valid approach and further researches should be based on it.

It was believed that new descriptors applied to nanoparticles should be developed (like derived from TEM images). We showed that the important point about nanoparticles is more about the approach of describing rather than the concept of new descriptors. We now believe that cores, coatings, and surface modifications should be described with descriptors reflecting the specificities of each part (using orbital and symmetry based descriptors for metals, topological descriptors for organic groups, …), and this description should be combined with accurate measurements of the interaction of the particle within the medium, and it's big scale specific parameters such as size, porosity,…

Further work should be done on larger datasets including more accurate experimental data, with a perfect knowledge of each part of the used nanoparticles (not some surface modifications with molecules with unknown structures)

Dictionnary:

NP: NanoParticle
APS: Average Size of Particles
NPO: NanoParticle Ontology
MAE: Mean Average Error
RMAE: Relative MAE (in %)
RMSE: Root Mean Square Error
RRMSE: Relative RMSE (in %)
RBF: Radial Based Function
PCA: Principal Component Analysis
SAR: Structure Activity Relationship
QSAR: Quantitative Structure Activity Relationship
QSPR: Quantitative Structure Property Relationship
QNPR: Quantitative Nanostructure Property Relationship
The difference between QSAR and QSPR is just that the target is not a molecular activity but a molecular property (such as pK_A, boiling temperature,…).

Bibliography

Handbook of Chemoinformatics ; Todeschini, R. and Consonni, V.; 2003; Vol. 3 (ed. J. Gasteiger), WILEY-VCH, Weinheim (GER), pp. 1004-1033.

The Structural Diversity of the Nanoworld; V. Ya. Shevchenko, A. E. Madison, and V. E. Shudegov; **Glass Physics and Chemistry**, Vol. 29, No. 6, 2003, 577–582.

Principles for characterizing the potential human health effects from exposure to nanomaterials: elements of a screening strategy; Günter Oberdörster et al.; **Particle and Fibre Toxicology**; 2005, 2:8;

Perturbational profiling of nanomaterial biologic activity; Stanley Y. Shaw and all; **Applied Biological Science**; May 27, 2008 vol. 105 no. 21 7387–7392

Quantitative Nanostructure-Activity relationship Modeling; Denis Fourches, D. Pu, C. Tassa, R. Weissleder, S. Y. Shaw, R. Mumper, A. Tropsha; **ACS nano** ; 2010, vol. 4, no.10, 5703-5712.

NanoParticle Ontology for cancer nanotechnology research; Dennis G. Thomas, Rohit V. Pappu, Nathan A. Baker; **Journal of Biomedical Informatics**, 2011; 44: 59-74;

Exploring Quantitative Nanostructure-Activity Relationships (QNAR) Modeling as a Tool for Predicting Biological Effects of Manufactured Nanoparticles; Denis Fourches, Dongqiuye Pu and Alexander Tropsha; **Combinatorial Chemistry & High Throughput Screening**; 2011, 14, 217-225