Read-across structural analysis of PFAS acute oral toxicity in rats powered by the Isalos Analytics Platform's Automated Machine Learning

The purpose of this research paper is to develop a robust modelling framework for predicting PFAS acute oral toxicity class in rats, leveraging the enhanced capabilities of the in-house Isalos Analytics Platform. The original dataset contains 777 molecular descriptors, from which 6 are selected for the model training, and a target column named "Class". The target column consists of two toxicity level indications, "high" and "low," according to the EPA categorization. Optimization of the machine learning algorithms was conducted using the Automated Machine Learning (Auto ML) functionality of the Isalos software, which indicated kNN as the best method for this application. The final predicting model was validated with statistical classification metrics, and its domain of applicability was calculated.

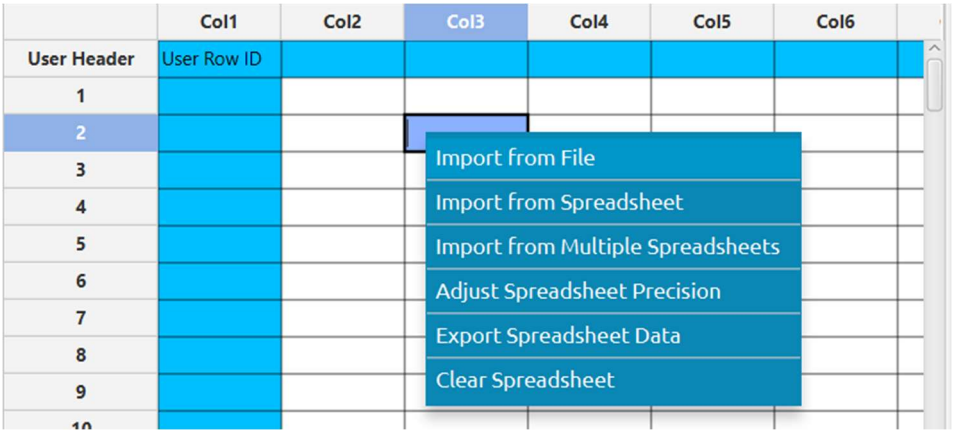
Scientific article: https://www.mdpi.com/2305-6304/14/2/152

Dataset access: https://db.chempharos.eu/datasets/Datasets.zul?datasetID=ds17

Isalos version used: 2.0.0

Step 1: Import data from file

Right click on the input spreadsheet (left) and choose the option "Import from File." Then navigate through your files to load the one with the PFAS toxicity data.

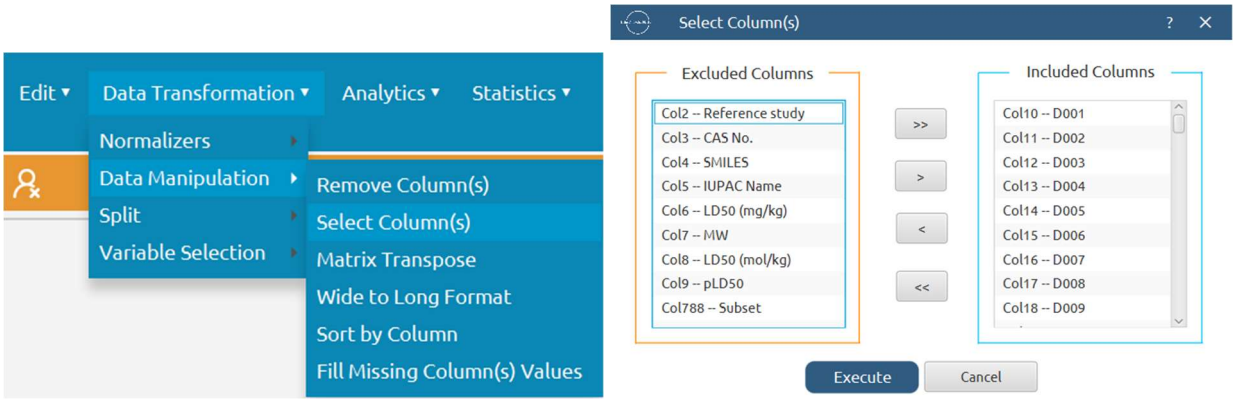


The data will appear on the left spreadsheet.

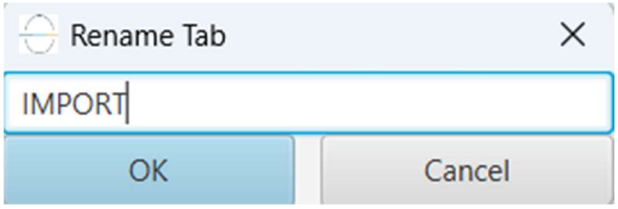
	Col1	Col2 (S)	Col3 (S)	Col4 (S)	Col5 (S)	Col6 (D)	Col7 (D)	Col8 (D)	Col9 (D)	Col10 (D)	Col11 (D)	Col12 (D)	Col13 (D)	Col14 (D)	Col15 (D)	Col16 (D)
User Header	User Row ID	Reference study	CAS No.	SMILES	IUPAC Name	LD50 (mg/kg)	MW	LD50 (mol/kg)	pLD50	D001	D002	D003	D004	D005	D006	D007
1		10.1016/j.jhazmat.2024.136071	458-24-2	CCNC(C)CC1=CC(=CC=C1)C(F)(F)F	N-ethyl-1-[3-(trifluoromethyl)phenyl]propan-2-amine	130.0	231.26	5.621378534982227E-4	3.25	1.0	0.0	0.0	0.0	1.0	0.0	0.0
2		10.1016/j.jhazmat.2024.136071	54910-89-3	CNCCC(C1=CC=CC=C1)OC2=CC=C(C=C2)C(F)(F)F	N-methyl-3-phenyl-3-[4-(trifluoromethyl)phenoxy]propan-1-amine	825.0	309.33	0.002667054601888149	2.57	2.0	0.0	0.0	0.0	2.0	0.0	0.0
3		10.1016/j.jhazmat.2024.136071	5002-47-1	CCCCCCCCCC(=O)OCCN1CCN(CC1)CCCN2C3=CC=CC=C3SC4=C2C=C(C=C4)C(F)(F)F	2-[4-[3-[2-(trifluoromethyl)phenothiazin-10-yl]propyl]piperazin-1-yl]ethyl decanoate	19.0	591.8	3.210544102737341E-5	4.49	2.0	0.0	0.0	0.0	4.0	0.0	0.0
4		10.1016/j.jhazmat.2024.136071	2746-81-8	CCCCCCC(=O)OCCN1CCN(CC1)CCCN2C3=CC=CC=C3SC4=C2C=C(C=C4)C(F)(F)F	2-[4-[3-[2-(trifluoromethyl)phenothiazin-10-yl]propyl]piperazin-1-yl]ethyl heptanoate	230.0	549.7	4.184100418410004E-4	3.38	2.0	0.0	0.0	0.0	4.0	0.0	0.0
5		10.1016/j.jhazmat.2024.136071	75706-12-6	CC1=C(C=NO1)C(=O)NC2=CC=C(C=C2)C(F)(F)F	5-methyl-N-[4-(trifluoromethyl)phenyl]-1,2-oxazole-4-carboxamide	235.0	270.21	8.696939417490 1E-4	3.06	1.0	0.0	0.0	1.0	1.0	0.0	0.0
6		10.1016/j.jhazmat.2024.136071	76-38-0	COC(C(Cl)Cl)(F)F	2,2-dichloro-1,1-difluoro-1-methoxyethane	3600.0	164.96	0.021823472356935	1.66	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Step 2: Manipulate data

Some columns contain metadata, so we will exclude them to retain only the molecular descriptor features and the class variable. On the menu click on *Data Transformation → Data Manipulation → Select Column(s)* and select all columns except Col2-9 and Col788.



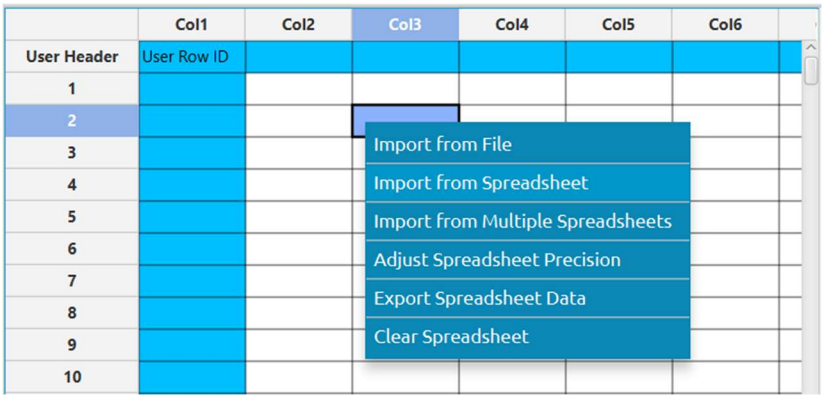
All the data will appear in the output (right) spreadsheet. This tab can be renamed “Dataset” by right-clicking on it and choosing the “Rename” option.



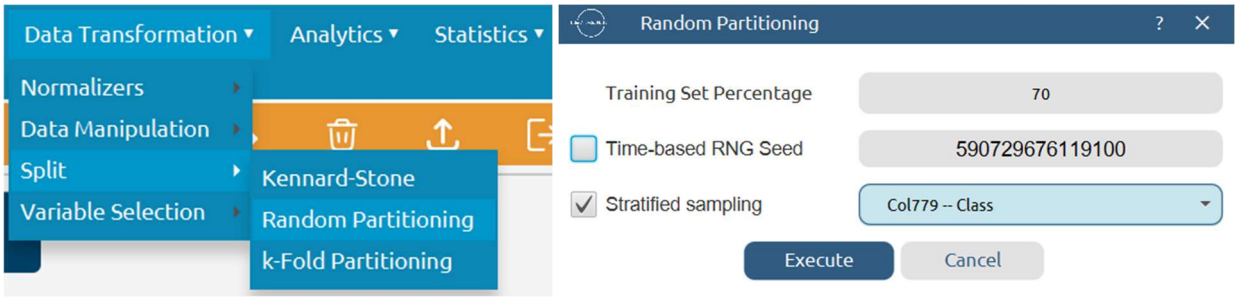
Step 3: Split data

Create a new tab by pressing the "+" button on the bottom of the page with the name "Splitting" which we will use for splitting the train and test set with stratified random partitioning.

Import data into the input spreadsheet of the "Splitting" tab from the output of the "IMPORT" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."



Split the dataset by choosing *Data Transformation → Split → Random Partitioning*. Then choose the "Training set percentage" and the column for the stratified sampling as shown below:



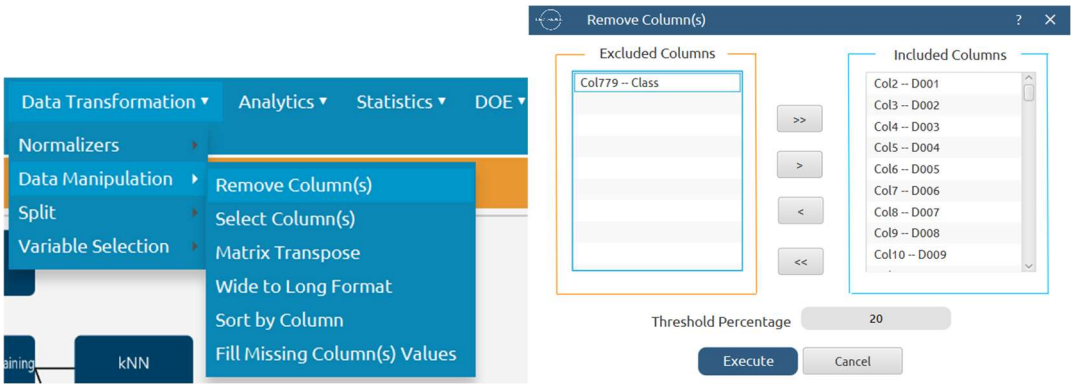
The results will be two separate spreadsheets, "Splitting: Training Set" and "Splitting: Test Set," which will be available to import into the next tabs.

Step 4: Filter columns

Create a new tab by pressing the "+" button on the bottom of the page with the name "Column filtering." We will use this tab to remove the redundant columns. A column is removed if it contains an instance whose percentage is above the defined threshold in the specific column.

Import into the input spreadsheet of the "Column filtering" tab the train set from the output of the "Splitting" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet." From the available Select input tab options choose "Splitting: Training Set."

On the menu click on *Data Transformation → Data Manipulation → Remove Column(s)*, select all columns except "Class," and set the "Threshold Percentage" to 20%.



The results will appear on the output spreadsheet.

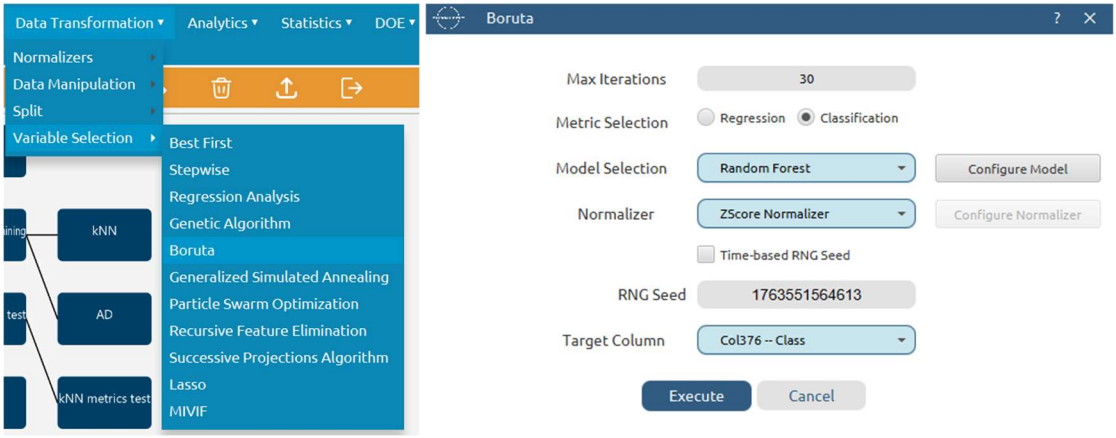
	Col1	Col2 (D)	Col3 (D)	Col4 (D)	Col5 (D)	Col6 (D)	Col7 (D)	Col8 (D)	Col9 (D)	Col10 (D)	Col11 (D)	Col12 (D)	Col13 (D)	Col14 (D)	Col15 (D)
User Header	User Row ID	D014	D015	D018	D019	D122	D123	D124	D125	D126	D127	D131	D132	D133	D134
1		18.0	0.439024	48.0	38.0	549.707	7.23299	76.0	38.0	79.0	41.0	474.842	123.03	1.61882	11.0
2		8.0	0.380952	28.0	9.0	282.225	9.7319	29.0	20.0	30.0	21.0	140.881	55.4915	1.9135	7.0
3		6.0	0.375	20.0	5.0	234.202	11.7101	20.0	15.0	21.0	16.0	86.4386	44.074	2.2037	6.0
4		11.0	1.0	11.0	3.0	200.057	13.3371	15.0	12.0	14.0	11.0	58.6034	26.197	1.74647	5.0
5		10.0	0.384615	32.0	19.0	352.426	8.19596	43.0	24.0	45.0	26.0	233.329	70.8052	1.64663	9.0
6		15.0	0.652174	28.0	16.0	335.286	8.59709	39.0	23.0	39.0	23.0	206.131	76.5152	1.96193	9.0
7		5.0	1.0	5.0	3.0	100.041	11.1157	9.0	6.0	8.0	5.0	28.5293	17.0195	1.89106	2.0
8		7.0	1.0	7.0	0.0	187.377	23.4221	8.0	8.0	7.0	7.0	24.0	12.4902	1.56128	4.0
9		7.0	0.5	18.0	3.0	225.556	13.268	17.0	14.0	17.0	14.0	69.4869	38.3256	2.25445	6.0
10		6.0	0.5	15.0	3.0	215.004	14.3336	15.0	12.0	15.0	12.0	58.6034	27.4421	1.82947	5.0
11		29.0	0.966667	31.0	1.0	514.088	16.0653	32.0	31.0	31.0	30.0	160.0	44.0701	1.37719	19.0
12		5.0	0.454545	14.0	1.0	168.067	14.0056	12.0	11.0	12.0	11.0	43.0195	15.9001	1.32501	5.0
13		25.0	1.0	25.0	4.0	432.107	14.4036	30.0	26.0	29.0	25.0	147.207	46.6774	1.55591	15.0
14		8.0	0.444444	22.0	4.0	254.137	12.1017	21.0	17.0	22.0	18.0	92.2387	38.1996	1.81903	8.0
15		7.0	0.538462	16.0	3.0	198.094	12.3809	16.0	13.0	16.0	13.0	64.0	27.984	1.749	6.0

Step 5: Select features

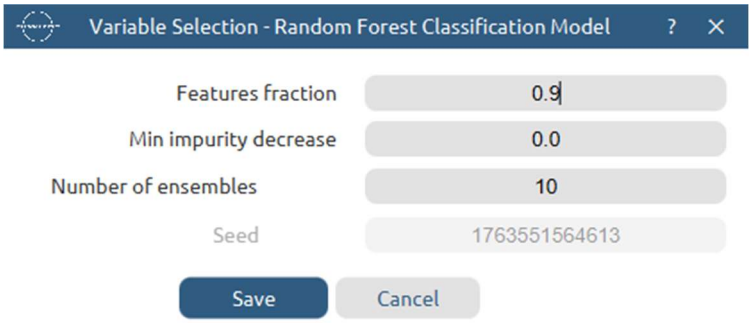
We want to determine the features that will be the most useful for predicting the toxicity outcome. Create a new tab by pressing the "+" button on the bottom of the page with the name "Variable Selection."

Import into the input spreadsheet of the "Variable Selection" tab the train set from the output of the "Column filtering" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."

Use the Boruta method for the feature selection by choosing: *Data Transformation → Variable Selection → Boruta*. Choose the "Random Forest" model and the Z-score normalizer in the configuration box. Select the column "Class" as the target column and a maximum of 30 iterations.



Afterwards, choose "Configurate Model" for the Random Forest model to specify its configuration parameters.



The results will appear on the output spreadsheet.

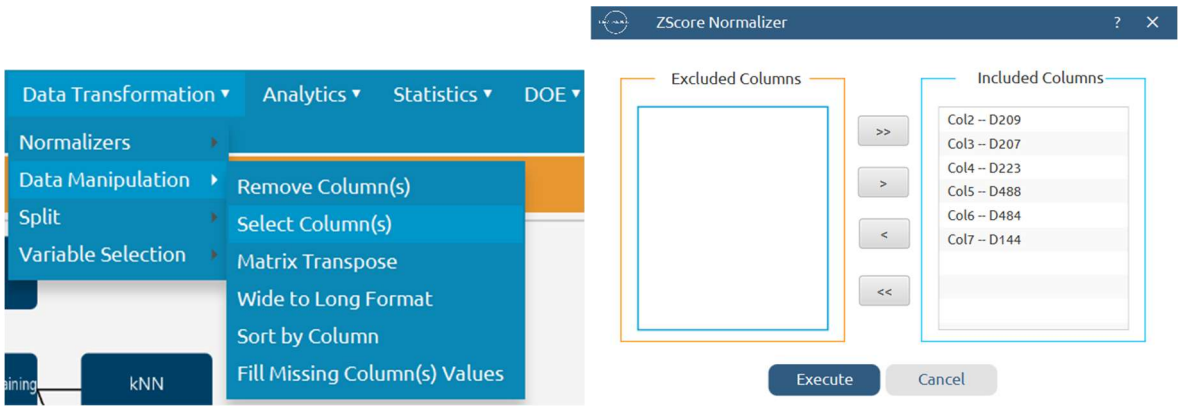
	Col1	Col2 (D)	Col3 (D)	Col4 (D)	Col5 (D)	Col6 (D)	Col7 (D)	Col8 (S)
User Header	User Row ID	D209	D207	D223	D488	D484	D144	Class
1		-0.0353844	0.2507846	0.7863221	-0.4405343	-0.7505183	-1.5236702	high
2		-0.0260365	0.1898634	-0.1893803	-0.8769640	-0.8308732	-0.4167087	high
3		-0.4776107	-0.5898311	0.2457132	-0.1867812	0.6501662	0.0047608	high
4		-0.3577662	1.7454829	-0.6256126	1.0043197	-1.3712213	-0.9731268	low
5		-0.6303886	-0.3929619	0.5268425	-0.9442624	-0.4761701	-1.0351843	high
6		-0.6121243	1.1400401	-0.1160083	-0.8722781	-1.1412004	-1.3592496	low
7		8.0638542	4.1816028	-1.2562375	2.9257729	-1.3712213	-1.7055035	high
8		3.0997491	1.2725834	-1.2562375	2.0490788	-1.3712213	2.0577571	low
9		0.1590994	0.4925240	-0.1281125	-0.1015263	0.8855670	0.3566358	low
10		0.0566563	0.4531866	0.2318662	-0.6003198	2.0869392	0.7932217	low
11		-0.0935329	-0.7240767	-0.6831829	3.0567082	0.2798080	-0.0479845	high
12		1.1760044	0.8825537	-0.5160500	2.6409245	-1.3712213	0.6092189	low
13		0.1550247	-0.1942079	-0.6266562	1.9187085	-0.1600983	-0.5622796	low
14		-0.5117425	-1.0258861	-0.3677910	-0.2915596	0.8170065	-0.0419971	high
15		0.5912588	1.1153555	-0.5097251	0.3125322	-1.1504466	-0.0745685	low

Step 6: Select the train set columns

We want to choose the columns of the non-normalized dataset that will be included in the training set, as indicated by the Boruta method. Create a new tab by pressing the "+" button on the bottom of the page with the name "Select columns - Train."

Import into the input spreadsheet of the "Select columns - Train" tab the train set from the output of the "Column filtering" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."

On the menu click on *Data Transformation → Data Manipulation → Select Column(s)* and select the columns "D209", "D207", "D223", "D488", "D484", "D144", and "Class".



The results will appear on the output spreadsheet.

Step 7: Automated ML optimization

We will compare four machine learning algorithms commonly employed for classification applications: the kNN, XGBoost, Random Forest, and Fully Connected Neural Network models. This procedure can be performed automatically with the Auto ML option of Isalos Analytics Platform and it is beneficial when optimizing a predictive model.

Create a new tab by pressing the "+" button on the bottom of the page with the name "AutoML."

Import into the input spreadsheet of the "AutoML" tab the output of the "Select columns - Train" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."

Perform the algorithm optimisation by choosing: *Analytics → Classification → Auto ML*.

Select the kNN, XGBoost, Random Forest, and Fully Connected Neural Network models to be used inside the Auto ML configuration. Define the search range of all hyperparameters for each algorithm with the values written in Table 1, by double-clicking on them inside the "Selected Models" box. It should be noted that several parameters are kept constant to decrease computational costs.

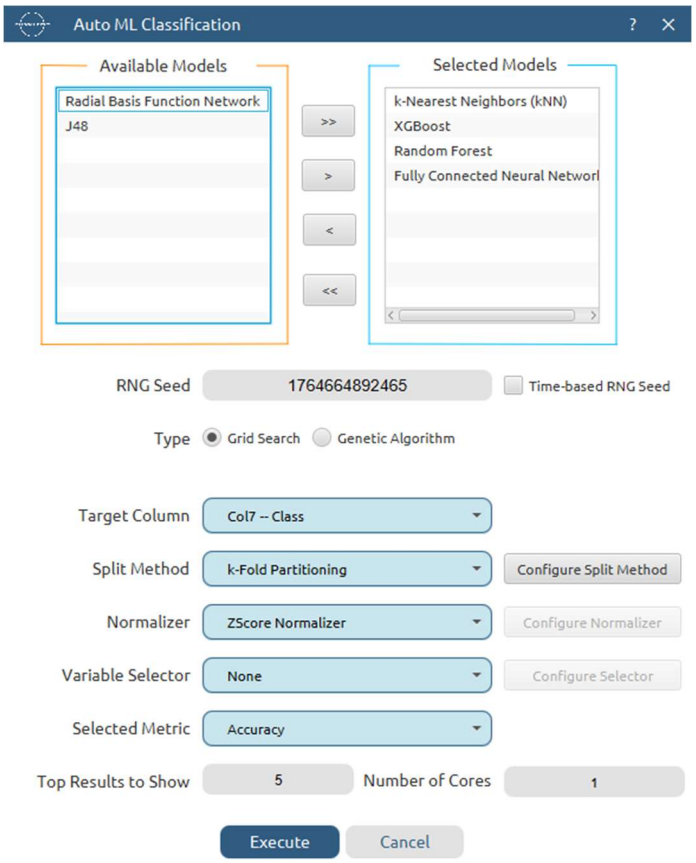
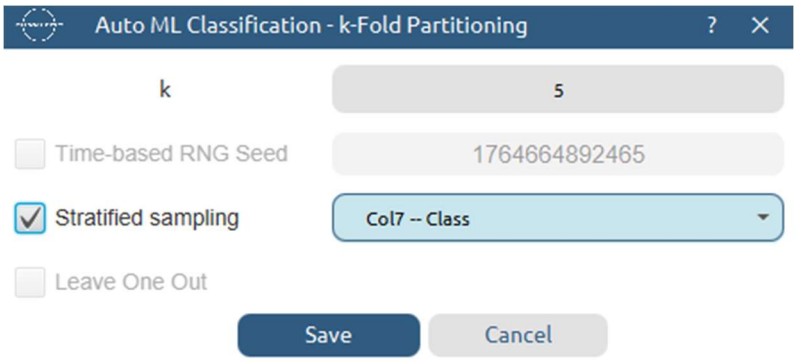


Table 1: Hyperparameter search ranges for model optimization inside the Isalos AutoML scheme

ML method	Hyperparameter	Search range [min, max; step]
kNN	Number of nearest neighbours, k	[3, 9; 1]
Random Forest	Feature fracture	[0.5, 0.9; 0.2]
	Min impurity decrease (constant)	[0, 0; 1]
	Number of ensembles	[100, 200; 50]
XGBoost	Number of trees	[5, 20; 1]
	Learning rate	[0.1, 0.3; 0.1]
	Gamma (constant)	[0, 0; 1]
	Max tree depth (constant)	[6, 6; 1]
	Minimum child weight (constant)	[1, 1; 1]
	Column sample by tree (constant)	[1, 1; 1]
	Subsample (constant)	[1, 1; 1]
	Lambda	[0.8, 1; 0.1]
	Alpha	[0.8, 1; 0.1]
Fully Connected Neural Network	Number of hidden layers (constant)	[2, 2; 1]
	Number of neurons/layer	[50, 100; 50]
	Activation function	RELU
	Batch size (constant)	[128, 128; 1]
	Number of epochs	[50, 150; 50]
	Learning rate	[0.001, 0.01; 0.009]
	Momentum	[0.8, 0.9; 0.1]

Afterwards, employ the grid search method for the exploration of the hyperparameter space, and define the target column, “Class.” Choose the split method “k-Fold Partitioning,” and click on the “Configure Split Method” button to select 5 folds and stratified sampling for the column “Class.”



Choose the z-score normalizer to maintain consistency with the previous preprocessing steps. There is no need for variable selection, so choose the option “None.” Finally, for the selected metric choose “Accuracy.” The fine-tuned model with the hyperparameters yielding the highest average accuracy across the five folds are selected as optimal.

The results will appear on the output spreadsheet. The algorithm indicated as optimal is kNN with 4 nearest neighbours.

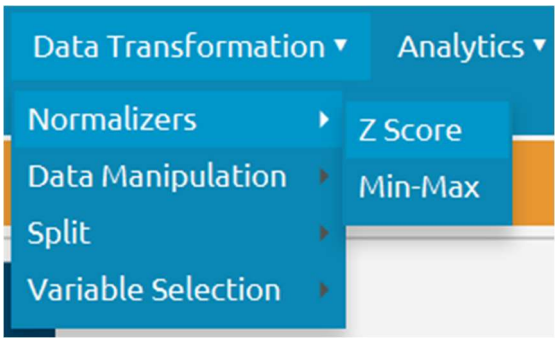
	Col1	Col2 (I)	Col3 (S)	Col4 (S)	Col5 (S)	Col6 (D)	Col7 (D)	Col8 (D)	Col9 (D)	Col10 (D)	Col11 (D)	Col12 (D)	Col13 (D)	Col14 (D)	Col15 (D)
User Header	User Row ID	Rank	Model	Description	Value	Hamming Loss	Weighted Precision	Macro Average ROC AUC	Accuracy	Weighted Recall	Micro Recall	Weighted F1	Micro Youden's J	Micro Precision	Kappa
1		1	kNN model	Selected Metric	Accuracy = 0.8511627906976743	0.0279070	0.1720930	0.1712821	0.8511628	0.1719022	0.1720930	0.1715775	-0.0279070	0.1720930	0.1366093
2				Feature Headers	D144, D207, D209, D484, D488										
3				k	4										
4				Confusion Matrix											
5				Class 0: [23.8, 2.8]											
6				Class 1: [3.6, 12.8]											
7				Precision (Per Class)											
8				Class 0	0.893137										
9				Class 1	0.796923										
10				Recall (Per Class)											
11				Class 0	0.860642										
12				Class 1	0.820049										
13				F1Score (Per Class)											
14				Class 0	0.872613										
15				Class 1	0.803434										

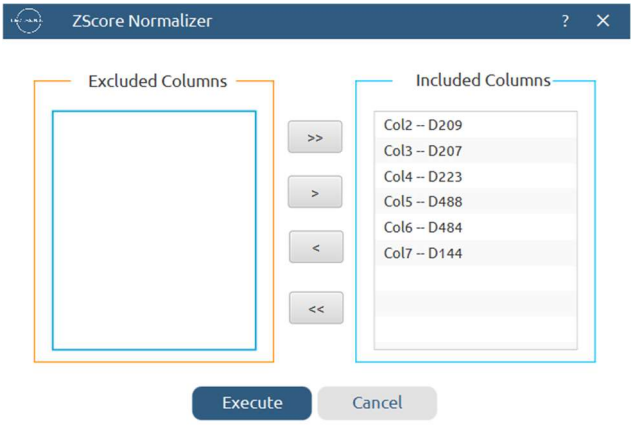
Step 8: Normalize the training set

Create a new tab by pressing the "+" button on the bottom of the page with the name "ZScore."

Import into the input spreadsheet of the "ZScore" tab the train set from the output of the "Select columns - Train" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."

Normalize the data using z-score: *Data Transformation → Normalizers → Z Score* and select all columns.





The results will appear on the output spreadsheet.

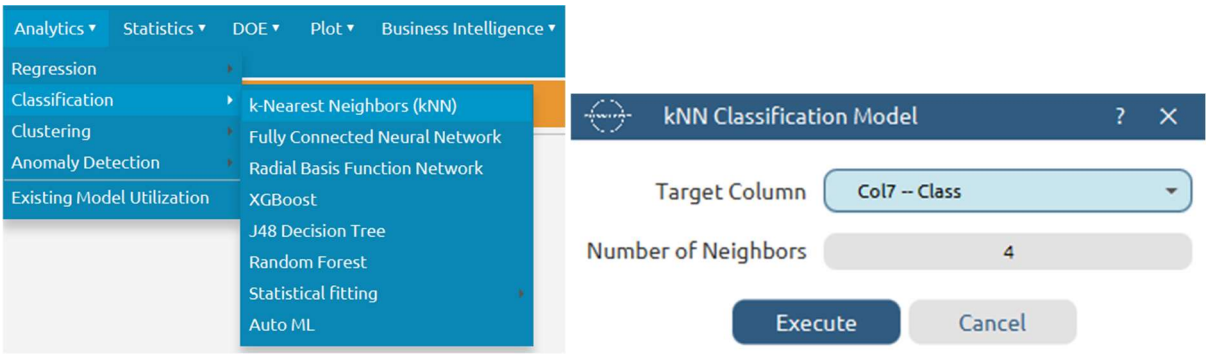
	Col1	Col2 (D)	Col3 (D)	Col4 (D)	Col5 (D)	Col6 (D)	Col7 (D)	Col8 (S)
User Header	User Row ID	D144	D207	D209	D223	D484	D488	Class
1		-1.5236702	0.2507846	-0.0353844	0.7863221	-0.7505183	-0.4405343	high
2		-0.4167087	0.1898634	-0.0260365	-0.1893803	-0.8308732	-0.8769640	high
3		0.0047608	-0.5898311	-0.4776107	0.2457132	0.6501662	-0.1867812	high
4		-0.9731268	1.7454829	-0.3577662	-0.6256126	-1.3712213	1.0043197	low
5		-1.0351843	-0.3929619	-0.6303886	0.5268425	-0.4761701	-0.9442624	high
6		-1.3592496	1.1400401	-0.6121243	-0.1160083	-1.1412004	-0.8722781	low
7		-1.7055035	4.1816028	8.0638542	-1.2562375	-1.3712213	2.9257729	high
8		2.0577571	1.2725834	3.0997491	-1.2562375	-1.3712213	2.0490788	low
9		0.3566358	0.4925240	0.1590994	-0.1281125	0.8855670	-0.1015263	low
10		0.7932217	0.4531866	0.0566563	0.2318662	2.0869392	-0.6003198	low
11		-0.0479845	-0.7240767	-0.0935329	-0.6831829	0.2798080	3.0567082	high
12		0.6092189	0.8825537	1.1760044	-0.5160500	-1.3712213	2.6409245	low
13		-0.5622796	-0.1942079	0.1550247	-0.6266562	-0.1600983	1.9187085	low
14		-0.0419971	-1.0258861	-0.5117425	-0.3677910	0.8170065	-0.2915596	high
15		-0.0745685	1.1153555	0.5912588	-0.5097251	-1.1504466	0.3125322	low

Step 9: Train the model

Create a new tab by pressing the "+" button on the bottom of the page with the name "kNN."

Import data into the input spreadsheet of the "kNN" tab from the output of the "ZScore" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."

Use the kNN method with 4 neighbours to train and fit the model: *Analytics → Classification → k-Nearest Neighbors (kNN)*



The predictions will appear on the output spreadsheet.

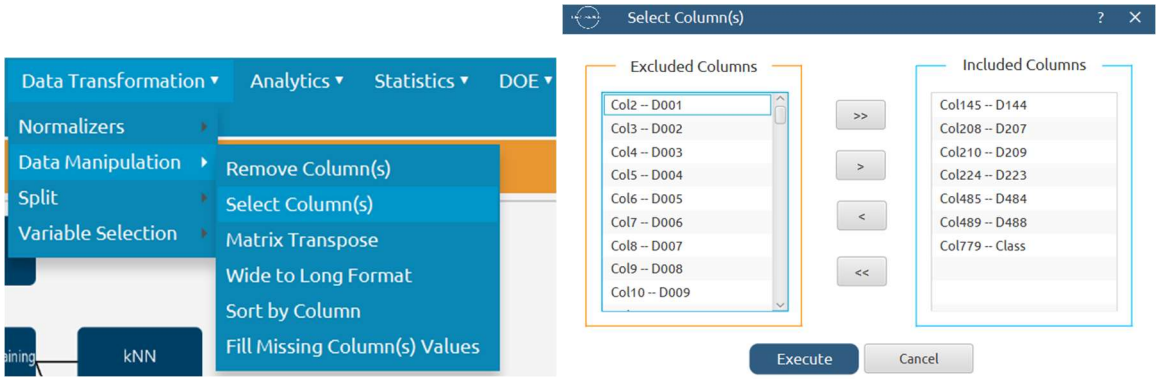
	Col1	Col2 (S)	Col3 (S)	Col4 (S)	Col5 (D)	Col6 (S)	Col7 (D)	Col8 (S)	Col9 (D)	Col10 (S)	Col11 (D)
User Header	User Row ID	Class	kNN Prediction	Closest NN1	Distance from NN1	Closest NN2	Distance from NN2	Closest NN3	Distance from NN3	Closest NN4	Distance from NN4
1		high	high	Entry 1	0.0	Entry 70	0.1018647	Entry 139	0.1303730	Entry 197	0.1382681
2		high	high	Entry 2	0.0	Entry 98	0.0715729	Entry 182	0.0778756	Entry 196	0.0871878
3		high	high	Entry 3	0.0	Entry 168	0.0646056	Entry 57	0.0704717	Entry 95	0.0752556
4		low	low	Entry 4	0.0	Entry 93	0.2182752	Entry 144	0.2291413	Entry 211	0.2479454
5		high	high	Entry 5	0.0	Entry 127	0.0863026	Entry 154	0.1139500	Entry 24	0.1280309
6		low	low	Entry 6	0.0	Entry 91	0.0772513	Entry 42	0.0793264	Entry 36	0.1060843
7		high	high	Entry 7	0.0	Entry 199	0.6811341	Entry 51	0.6873395	Entry 160	0.7737180
8		low	low	Entry 8	0.0	Entry 205	0.2842967	Entry 134	0.3794264	Entry 138	0.3867656
9		low	low	Entry 9	0.0	Entry 41	0.1398754	Entry 130	0.1426352	Entry 44	0.1510365
10		low	low	Entry 10	0.0	Entry 28	0.2121486	Entry 27	0.2159452	Entry 203	0.2282322
11		high	high	Entry 11	0.0	Entry 171	0.0689094	Entry 133	0.0734026	Entry 13	0.2598155
12		low	low	Entry 12	0.0	Entry 175	0.1503823	Entry 17	0.1781120	Entry 142	0.2294134
13		low	low	Entry 13	0.0	Entry 171	0.1952589	Entry 92	0.2099183	Entry 88	0.2276716
14		high	high	Entry 14	0.0	Entry 97	0.0621231	Entry 189	0.0784473	Entry 95	0.0813766
15		low	low	Entry 15	0.0	Entry 211	0.1231385	Entry 126	0.1778211	Entry 20	0.1942033

Step 10: Select the columns of the test set

Create a new tab by pressing the "+" button on the bottom of the page with the name "Select Test set."

Import data into the input spreadsheet of the "Select Test set" tab from the output of the "Splitting" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet." From the available input tab options select "Splitting: Test Set."

Select the columns "D144", "D207", "D209", "D223", "D484", "D488" and the target column "Class": *Data Transformation → Data Manipulation → Select Column(s)*



The results will appear on the output spreadsheet.

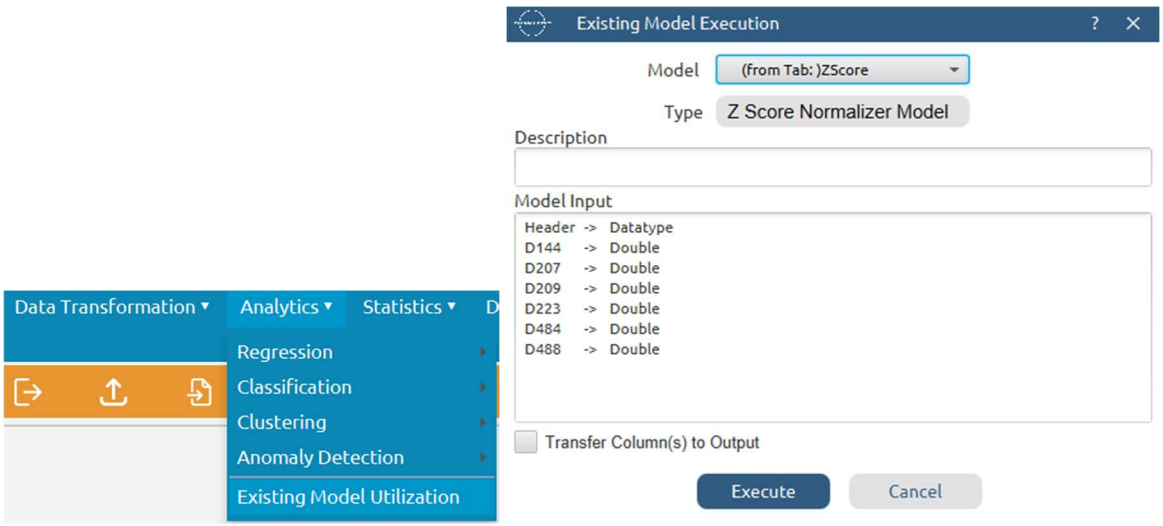
	Col1	Col2 (D)	Col3 (D)	Col4 (D)	Col5 (D)	Col6 (D)	Col7 (D)	Col8 (S)
User Header	User Row ID	D144	D207	D209	D223	D484	D488	Class
1		0.635161	0.464429	0.196727	0.0307312	0.223063	0.268892	high
2		0.668585	0.456667	0.193126	0.0279066	0.0545424	0.298983	low
3		0.638893	0.451061	0.187288	0.0406846	0.324102	0.377843	high
4		0.716095	0.443801	0.189275	0.0199492	0.155046	0.546474	high
5		0.716375	0.500577	0.274909	0.0	0.0	0.544148	low
6		0.740618	0.437952	0.187249	0.0111683	0.843315	1.21319	low
7		0.713218	0.460517	0.202137	0.0255082	0.0	0.196187	low
8		0.694585	0.454456	0.197571	0.0220726	0.0	0.17172	high
9		0.7373	0.454592	0.194116	0.0190874	0.290474	0.419639	low
10		0.719204	0.453778	0.181349	0.0179999	0.637907	0.707714	high
11		0.754077	0.464386	0.194973	0.020037	1.11599	0.410514	low
12		0.66399	0.470153	0.198029	0.0166708	0.240419	0.484788	low
13		0.734421	0.444338	0.192363	0.0112969	0.697074	1.16032	low
14		0.692545	0.429752	0.179547	0.0201117	0.840291	0.410653	high
15		0.816248	0.429752	0.172812	0.0204447	1.36333	0.375124	high

Step 11: Normalize the test set

Create a new tab by pressing the "+" button on the bottom of the page with the name "Normalize Test set."

Import data into the input spreadsheet of the "Test normalization" tab from the output of the "Test column selection" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."

Normalize the test set using the existing normalizer of the training set: *Analytics → Existing Model Utilization → Model (from Tab:) ZScore*



The results will appear on the output spreadsheet.

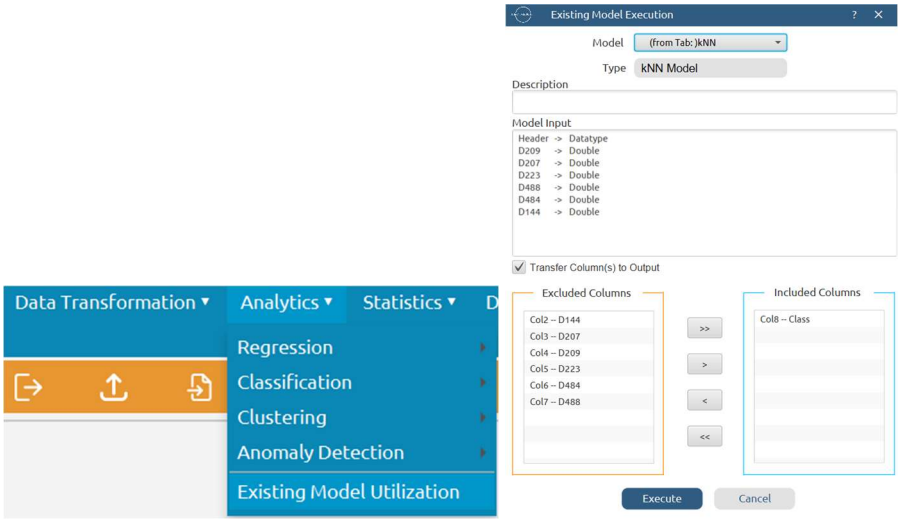
	Col1	Col2 (D)	Col3 (D)	Col4 (D)	Col5 (D)	Col6 (D)	Col7 (D)	Col8 (S)
User Header	User Row ID	D144	D207	D209	D223	D484	D488	Class
1		-1.6948248	1.2812777	0.5459576	0.3716654	-0.9706627	-0.9958756	high
2		-1.2239488	0.8093509	0.3733335	0.2220398	-1.2732785	-0.8825276	low
3		-1.6422486	0.4685082	0.0934725	0.8989201	-0.7892250	-0.5854747	high
4		-0.5546298	0.0271028	0.1887250	-0.1994821	-1.0928022	0.0497311	high
5		-0.5506852	3.4790630	4.2938323	-1.2562375	-1.3712213	0.0409694	low
6		-0.2091508	-0.3285143	0.0916029	-0.6646267	0.1431358	2.5611429	low
7		-0.5951609	1.0434295	0.8053012	0.0949910	-1.3712213	-1.2697436	low
8		-0.8576619	0.6749229	0.5864171	-0.0870007	-1.3712213	-1.3619069	high
9		-0.2558946	0.6831917	0.4207919	-0.2451336	-0.8496115	-0.4280359	low
10		-0.5108304	0.6337008	-0.1912302	-0.3027410	-0.2257193	0.6570962	high
11		-0.0195409	1.2786633	0.4618747	-0.1948311	0.6327836	-0.4624083	low
12		-1.2886830	1.6292948	0.6083726	-0.3731465	-0.9394962	-0.1826301	low
13		-0.2964539	0.0597522	0.3367569	-0.6578145	-0.1194720	2.3619901	low
14		-0.8864013	-0.8270713	-0.2776142	-0.1908741	0.1377056	-0.4618847	high
15		0.8563217	-0.8270713	-0.6004754	-0.1732343	1.0769368	-0.5957167	high

Step 12: Validate the model

Create a new tab by pressing the "+" button on the bottom of the page with the name "kNN validation."

Import data into the input spreadsheet of the "kNN validation" tab from the output of the "Normalize Test set" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet".

To validate the model: *Analytics → Existing Model Utilization→ Model (from Tab:) kNN*. Choose the column "Class" to be transferred to the output spreadsheet.



The predictions will appear on the output spreadsheet.

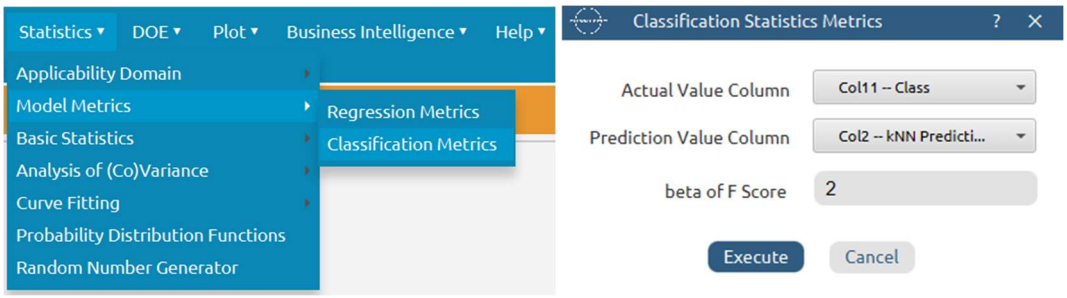
	Col1	Col2 (S)	Col3 (S)	Col4 (D)	Col5 (S)	Col6 (D)	Col7 (S)	Col8 (D)	Col9 (S)	Col10 (D)	Col11 (S)
User Header	User Row ID	kNN Prediction	Closest NN1	Distance from NN1	Closest NN2	Distance from NN2	Closest NN3	Distance from NN3	Closest NN4	Distance from NN4	Class
1		high	Entry 214	0.0	Entry 213	0.0	Entry 212	0.0	Entry 19	0.1482415	high
2		low	Entry 140	0.0759320	Entry 197	0.0835042	Entry 155	0.0866350	Entry 198	0.1029921	low
3		high	Entry 1	0.0524842	Entry 70	0.1236874	Entry 197	0.1380271	Entry 212	0.1612330	high
4		low	Entry 84	0.1149125	Entry 45	0.1440514	Entry 126	0.1458870	Entry 121	0.1476533	high
5		low	Entry 206	0.3113938	Entry 136	0.3385773	Entry 79	0.4187363	Entry 138	0.4652877	low
6		low	Entry 171	0.0534886	Entry 11	0.1157152	Entry 133	0.1262472	Entry 13	0.1517970	low
7		low	Entry 140	0.1458117	Entry 182	0.1497604	Entry 198	0.1502957	Entry 71	0.1717558	low
8		low	Entry 71	0.0882067	Entry 140	0.0943846	Entry 198	0.1108212	Entry 200	0.1515502	high
9		low	Entry 126	0.0666384	Entry 182	0.1002847	Entry 98	0.1018697	Entry 2	0.1228770	low
10		low	Entry 50	0.0808327	Entry 86	0.1208032	Entry 47	0.1243311	Entry 156	0.1369498	high
11		low	Entry 37	0.1640727	Entry 9	0.1691188	Entry 81	0.1996671	Entry 210	0.2090286	low
12		low	Entry 179	0.1323244	Entry 155	0.1328903	Entry 215	0.1388080	Entry 48	0.1388080	low
13		low	Entry 13	0.1066223	Entry 171	0.1450451	Entry 133	0.1842520	Entry 11	0.2004614	low
14		high	Entry 167	0.0849276	Entry 181	0.0941675	Entry 201	0.1006921	Entry 78	0.1086432	high
15		high	Entry 187	0.0341200	Entry 151	0.0430686	Entry 150	0.0681960	Entry 59	0.0831426	high

Step 13: Calculate statistics

Create a new tab by pressing the "+" button on the bottom of the page with the name "Metrics."

Import data into the input spreadsheet of the "Metrics" tab from the output of the "kNN validation" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet".

Calculate the statistical metrics for the classification: *Statistics → Model Metrics → Classification Metrics*



The results will appear on the output spreadsheet.

	Col1 (S)	Col2 (S)	Col3 (S)	Col4 (S)
User Header	User Row ID			
1			Predicted Class	Predicted Class
2			high	low
3	Actual Class	high	50	8
4	Actual Class	low	9	25
5				
6				
7	Classification Accuracy	0.8152174		
8				
9	Precision		0.8474576	0.7575758
10				
11	Recall/Sensitivity		0.8620690	0.7352941
12				
13	Specificity		0.7352941	0.8620690
14				
15	F1 Score		0.8547009	0.7462687
16				
17	F (beta=2)		0.8591065	0.7396450
18				
19	MCC	0.6011860		

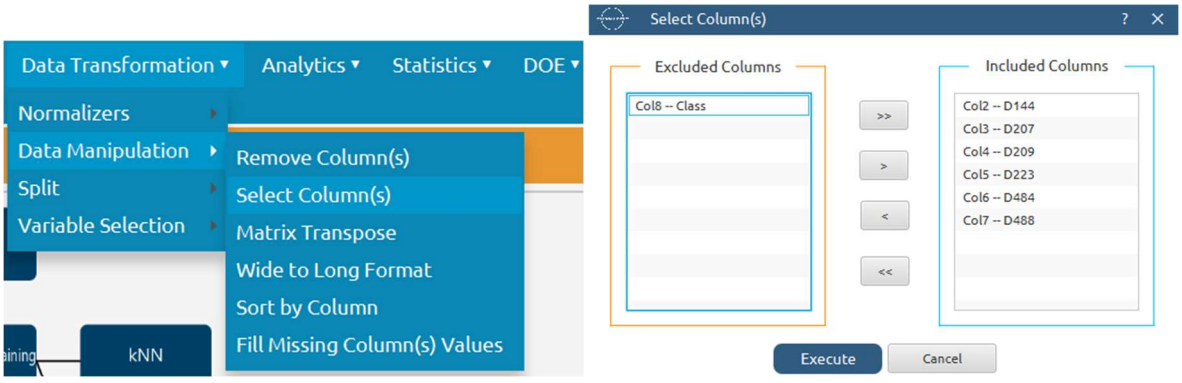
Step 14: Reliability check for each record of the test set

Step 14.a: Create the domain

Create a new tab by pressing the "+" button on the bottom of the page with the name "Remove column - Train."

Import into the input spreadsheet of the "Remove column - Train" tab the train set from the output of the "ZScore" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."

On the menu click on *Data Transformation → Data Manipulation → Select Column(s)* and select the columns all columns except "Class."

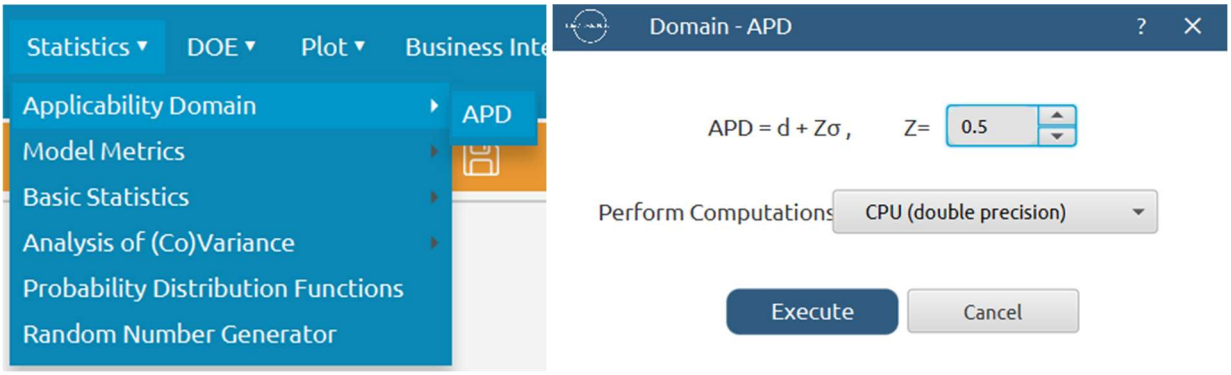


The results will appear on the output spreadsheet.

Afterwards, create a new tab by pressing the "+" button on the bottom of the page with the name "AD."

Import data into the input spreadsheet of the "AD" tab from the output of the "Remove column - Train" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."

Create the domain: *Statistics → Applicability Domain → APD*



The results will appear on the output spreadsheet.

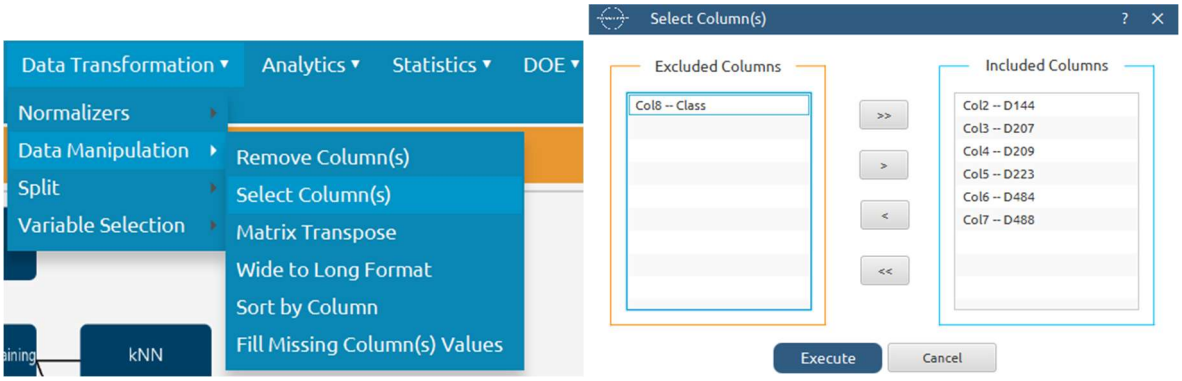
	Col1	Col2 (D)	Col3 (D)	Col4 (S)
User Header	User Row ID	Domain	APD	Prediction
1		0.0	2.1509897	reliable
2		0.0	2.1509897	reliable
3		0.0	2.1509897	reliable
4		0.0	2.1509897	reliable
5		0.0	2.1509897	reliable
6		0.0	2.1509897	reliable
7		0.0	2.1509897	reliable
8		0.0	2.1509897	reliable
9		0.0	2.1509897	reliable
10		0.0	2.1509897	reliable
11		0.0	2.1509897	reliable
12		0.0	2.1509897	reliable
13		0.0	2.1509897	reliable
14		0.0	2.1509897	reliable
15		0.0	2.1509897	reliable

Step 14.b: Check the test set reliability

Create a new tab by pressing the "+" button on the bottom of the page with the name "Remove column - Test."

Import into the input spreadsheet of the "Remove column - Test" tab the train set from the output of the "Normalize Test set" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."

On the menu click on *Data Transformation → Data Manipulation → Select Column(s)* and select the columns all columns except "Class."

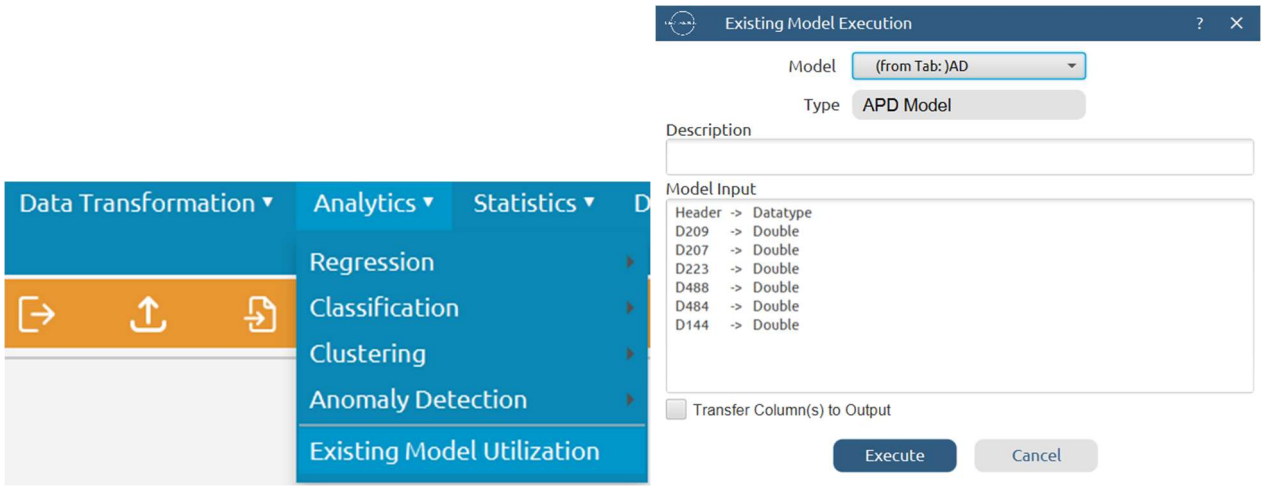


The results will appear on the output spreadsheet.

Afterwards, create a new tab by pressing the "+" button on the bottom of the page with the name "Reliability."

Import data into the input spreadsheet of the "Reliability" tab from the output of the "Normalize Test set" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet."

Check the Reliability: *Analytics → Existing Model Utilization → Model (from Tab:) AD*

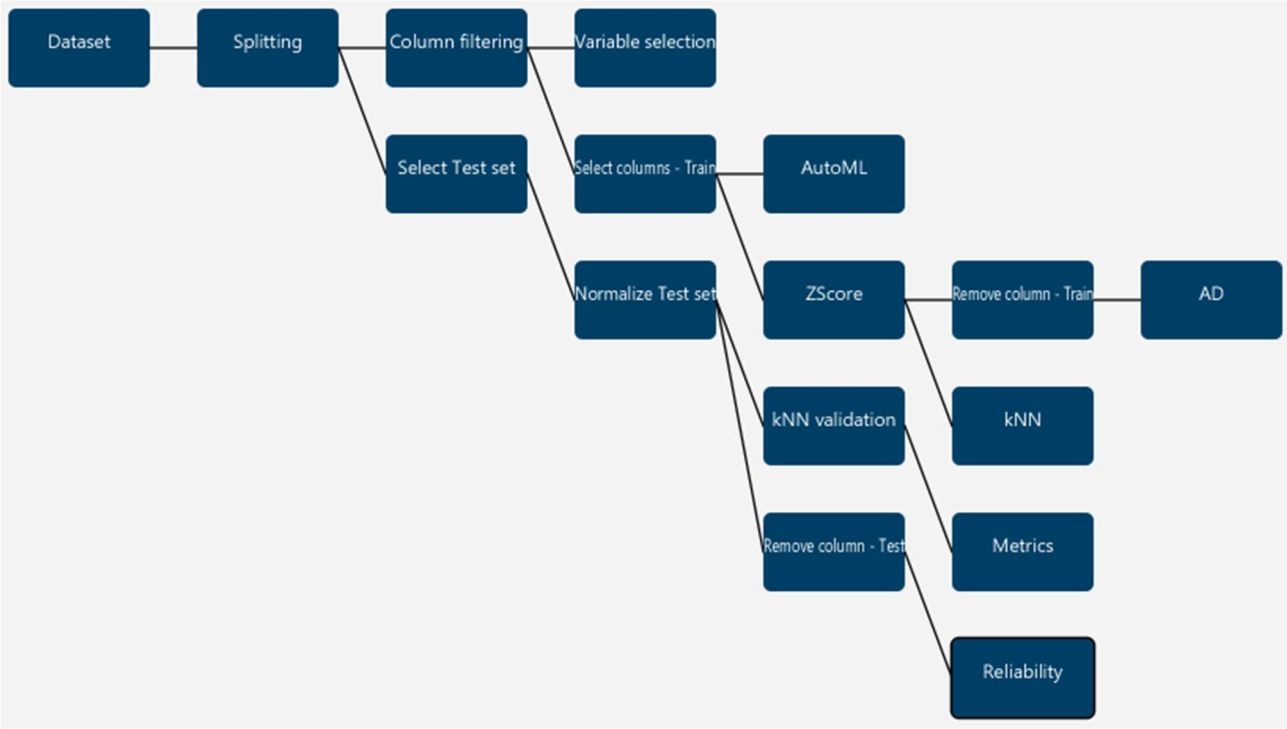


The results will appear on the output spreadsheet.

	Col1	Col2 (D)	Col3 (D)	Col4 (S)
User Header	User Row ID	Domain	APD	Prediction
1		0.0	2.1509897	reliable
2		0.4738005	2.1509897	reliable
3		0.3365304	2.1509897	reliable
4		0.5782019	2.1509897	reliable
5		2.3932166	2.1509897	unreliable
6		0.3400835	2.1509897	reliable
7		0.8605908	2.1509897	reliable
8		0.5598000	2.1509897	reliable
9		0.4046202	2.1509897	reliable
10		0.5145065	2.1509897	reliable
11		1.0245894	2.1509897	reliable
12		0.8729767	2.1509897	reliable
13		0.6060574	2.1509897	reliable
14		0.5035996	2.1509897	reliable
15		0.2786530	2.1509897	reliable

Final Isalos Workflow

Following the above-described steps, the final workflow on Isalos will look like this:



References

(1) Theodori, A.; Papavasileiou, K. D.; Tsoumanis, A.; Melagraki, G.; Afantitis, A. Read-Across Structural Analysis of PFAS Acute Oral Toxicity in Rats Powered by the Isalos Analytics Platform's Automated Machine Learning. Toxics 2026, 14 (2), 152. https://doi.org/10.3390/toxics14020152.