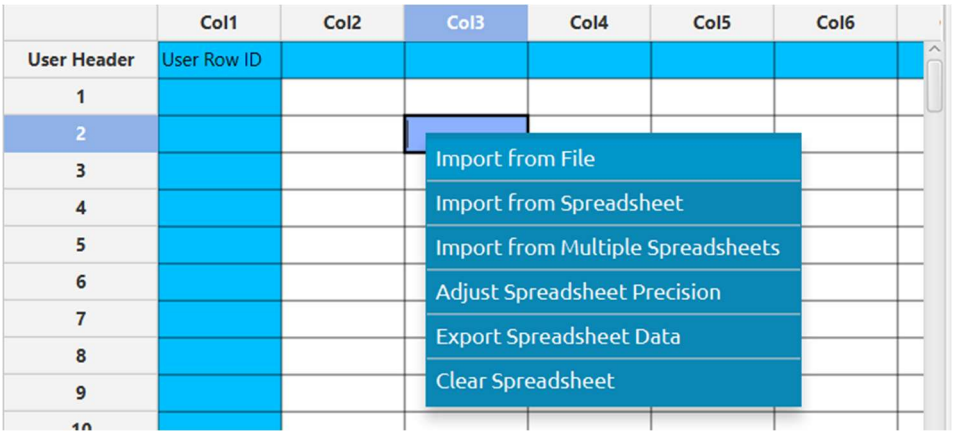
Credit card Details Binary Classification Problem

This dataset, which can be found in https://www.kaggle.com/datasets/rohitudageri/credit-card-details?select=Credit_card_label.csv, contains information about credit card applicants. It consists of 53 features and 1548 samples and it is used to predict whether an application for a credit card will be approved or denied. The dataset contains some categorical features which are encoded outside of Isalos using Python code.

Isalos version used: 2.0.6

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 credit card data.

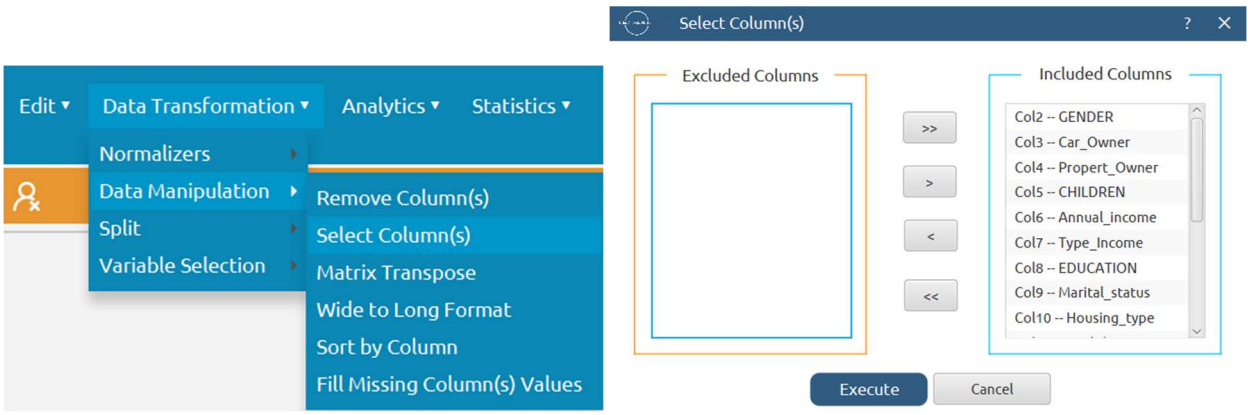


The data will appear on the left spreadsheet.

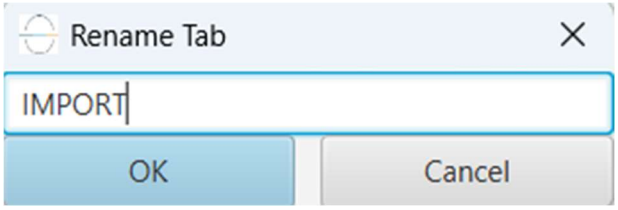
	Col1	Col2 (S)	Col3 (S)	Col4 (S)	Col5 (I)	Col6 (D)	Col7 (S)	Col8 (S)	Col9 (S)	Col10 (S)	Col11 (D)	Col12 (I)	Col13 (I)	Col14 (I)	Col15 (I)	Col16 (I)	Col17 (S)	Col18 (I)	Col19 (I)
User Header	User Row ID	GENDER	Car_Owner	Propert_Owner	CHILDREN	Annual_income	Type_Income	EDUCATION	Marital_status	Housing_type	Birthday_count	Employed_days	Mobile_phone	Work_Phone	Phone	EMAIL_ID	Type_Occupation	Family_Members	label
1	5008827	M	Y	Y	0	180000.0	Pensioner	Higher education	Married	House / apartment	-18772.0	365243	1	0	0	0		2	1
2	5009744	F	Y	N	0	315000.0	Commercial associate	Higher education	Married	House / apartment	-13557.0	-586	1	1	1	0		2	1
3	5009746	F	Y	N	0	315000.0	Commercial associate	Higher education	Married	House / apartment		-586	1	1	1	0		2	1
4	5009749	F	Y	N	0		Commercial associate	Higher education	Married	House / apartment	-13557.0	-586	1	1	1	0		2	1
5	5009752	F	Y	N	0	315000.0	Commercial associate	Higher education	Married	House / apartment	-13557.0	-586	1	1	1	0		2	1
6	5009753		Y	N	0	315000.0	Pensioner	Higher education	Married	House / apartment	-13557.0	-586	1	1	1	0		2	1
7	5009754	F	Y	N	0	315000.0	Commercial associate	Higher education	Married	House / apartment	-13557.0	-586	1	1	1	0		2	1
8	5009894	F	N	N	0	180000.0	Pensioner	Secondary / secondary special	Married	House / apartment	-22134.0	365243	1	0	0	0		2	1
9	5010864	M	Y	Y	1	450000.0	Commercial associate	Secondary / secondary special	Married	House / apartment	-18173.0	-678	1	0	1	1	Core staff	3	1
10	5010868	M	Y	Y	1	450000.0	Pensioner	Secondary / secondary special	Married	House / apartment	-18173.0	-678	1	0	1	1	Core staff	3	1
11	5010869	M	Y	Y	1	450000.0	Commercial associate	Secondary / secondary special	Single / not married	House / apartment	-18173.0	-678	1	0	1	1	Core staff	1	1
12	5018498	F	Y	Y	0	90000.0	Working	Secondary / secondary special	Married	House / apartment	-18950.0	-1002	1	1	1	0	Cooking staff	2	1
13	5018501	F	Y	Y	0		Working	Secondary / secondary special	Married	House / apartment	-18950.0	-1002	1	1	1	0	Cooking staff	2	1
14	5018503	F	Y	Y	0	90000.0	Working	Secondary / secondary special	Married	House / apartment	-18950.0	-1002	1	1	1	0	Cooking staff	2	1
15	5021303	M	N	N	1	472500.0	Pensioner	Higher education	Married	With parents	-8907.0	-913	1	0	0	1		3	1

Step 2: Manipulate data

We can select all the columns to be used. On the menu click on *Data Transformation → Data Manipulation → Select Column(s)* and select all columns.



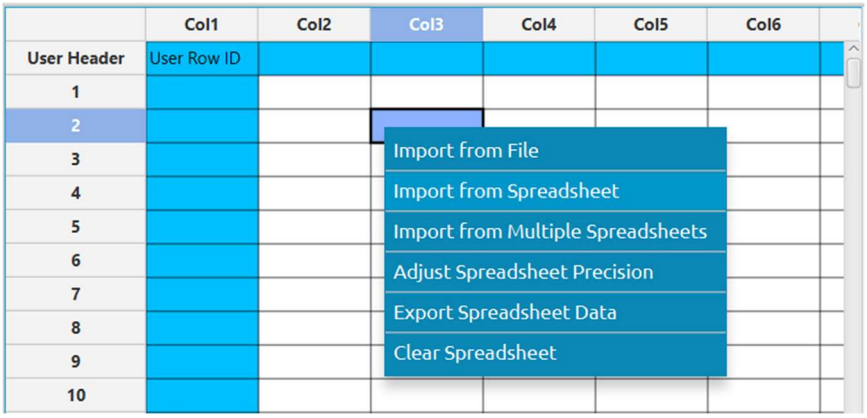
All of the data will appear in the output (right) spreadsheet. This tab can be renamed “IMPORT” by right-clicking on it and choosing the “Rename” option.



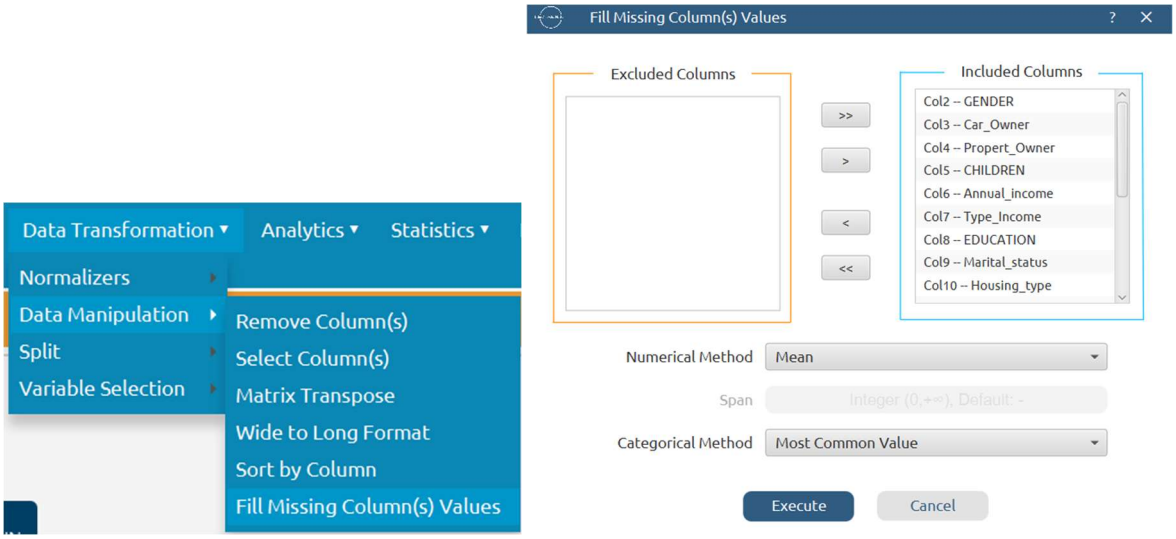
Step 3: Fill missing data

This dataset includes some missing values therefore we need to fill them appropriately. Create a new tab by pressing the "+" button on the bottom of the page with the name "FILL_MISSING" which we will use for filling the missing values.

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

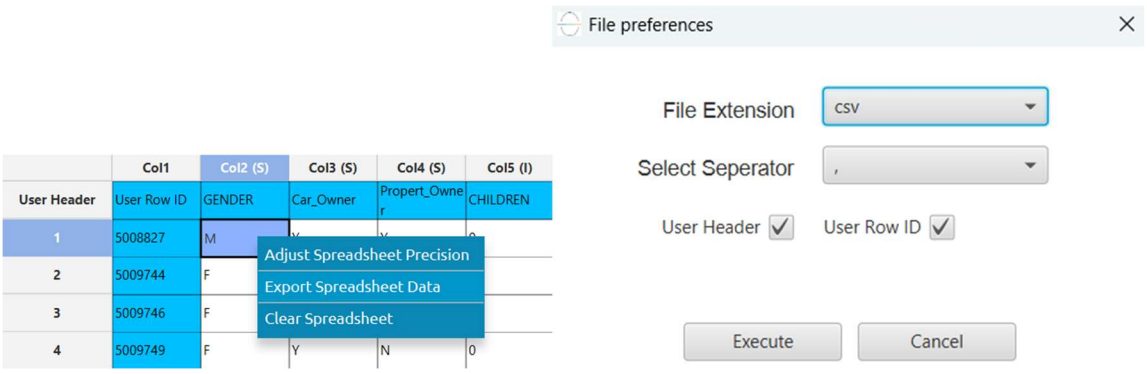


To fill the missing data, choose: *Data Transformation → Data Manipulation → Fill Missing Column(s) Values*. Choose to fill the numerical features with their mean values and the categorical features with their most common values.



The results will appear on the output spreadsheet.

Export the results in a csv file by right-clicking on the output spreadsheet and selecting "Export Spreadsheet Data".



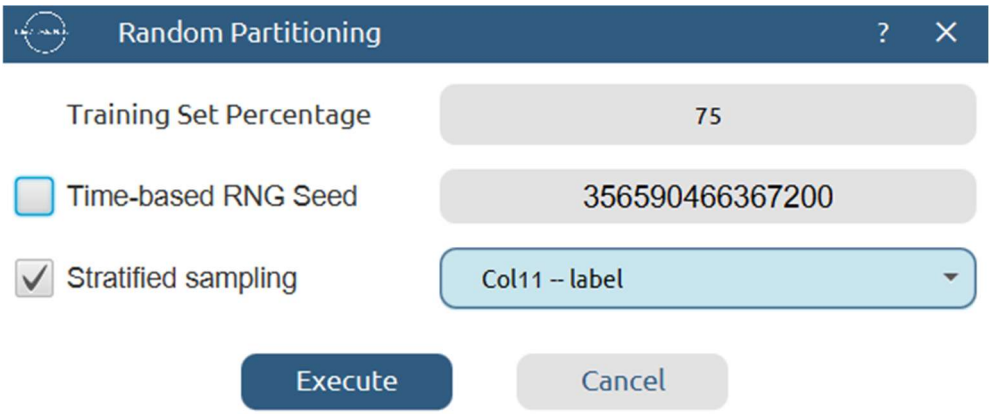
Afterwards, apply one-hot encoding on the categorical features of the dataset, which are: "GENDER", "Car_Owner", "Propert_Owner", "Type_Income", "EDUCATION", "Marital_status", "Housing_type" and "Type_Occupation".

Step 4: Split data

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

Import data into the input spreadsheet of the "TRAIN_TEST_SPLIT" tab from the one-hot encoded dataset by right-clicking on the input spreadsheet and then choosing "Import from File". Then navigate through your files to load the one with the encoded dataset.

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

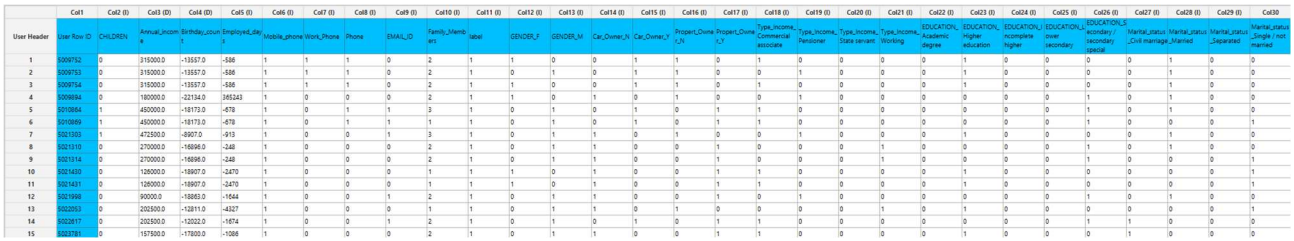


The results will be two separate spreadsheets, "TRAIN_TEST_SPLIT: Training Set" and "TRAIN_TEST_SPLIT: Test Set", which will be available to import into the next tabs.

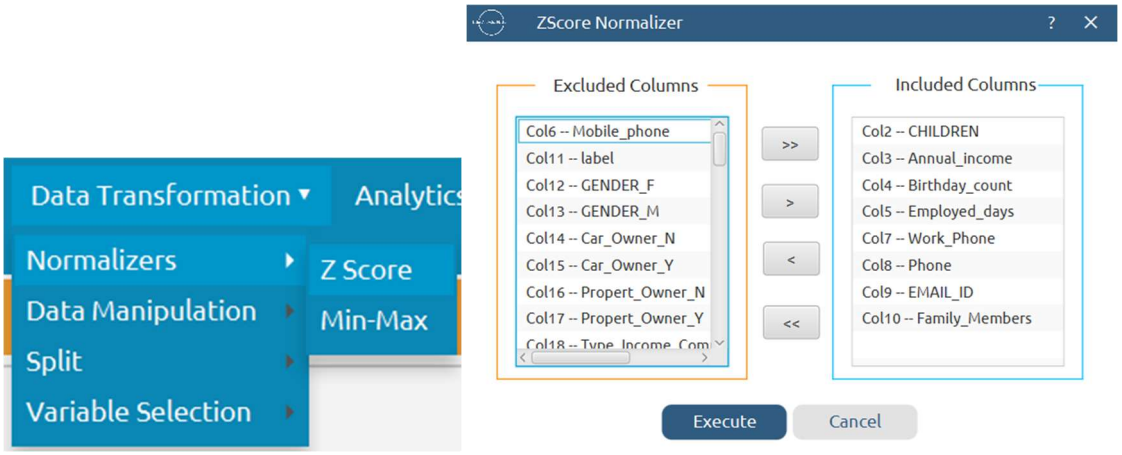
Step 5: Normalize the training set

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

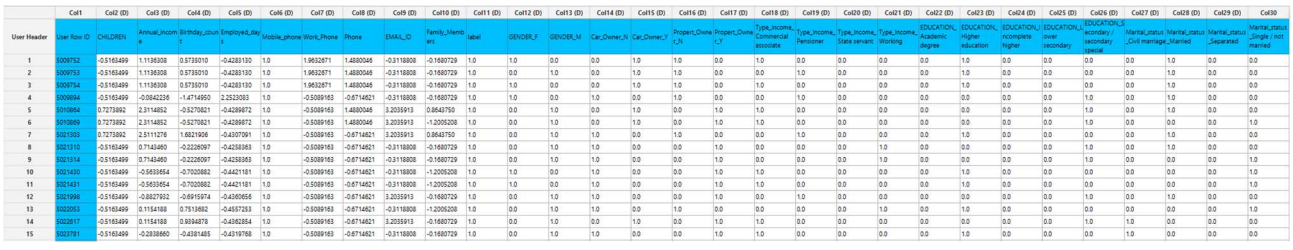
Import into the input spreadsheet of the "NORMALIZE_TRAIN_SET" tab the train set from the output of the "TRAIN_TEST_SPLIT" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet". From the available Select input tab options choose "TRAIN_TEST_SPLIT: Training Set".



Normalize the data using Z-score: *Data Transformation → Normalizers → Z Score* and select all the numerical feature columns except "Mobile_phone".



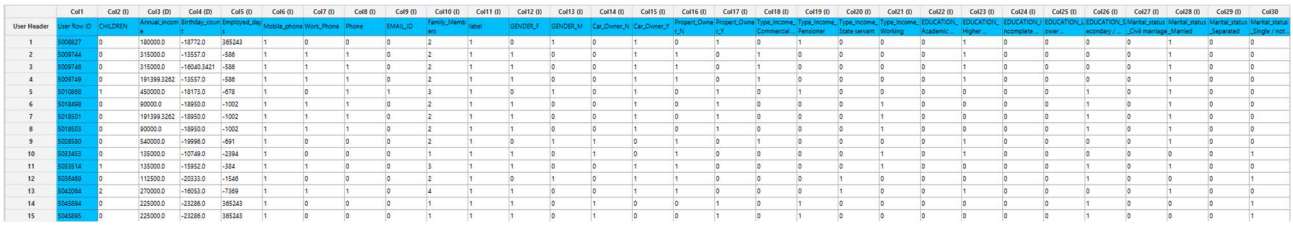
The results will appear on the output spreadsheet.



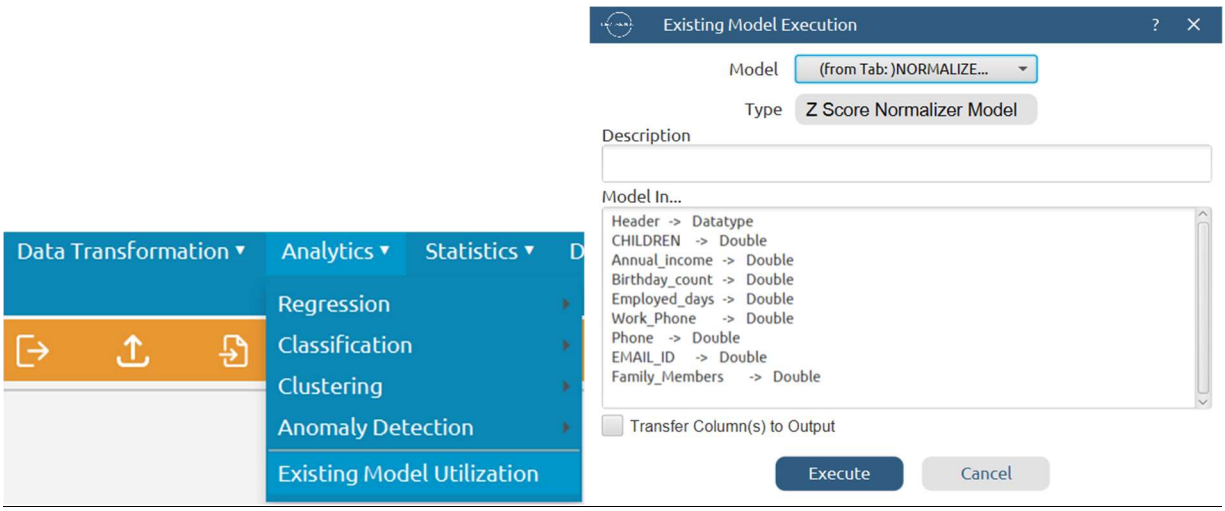
Step 6: Normalize the test set

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

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



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



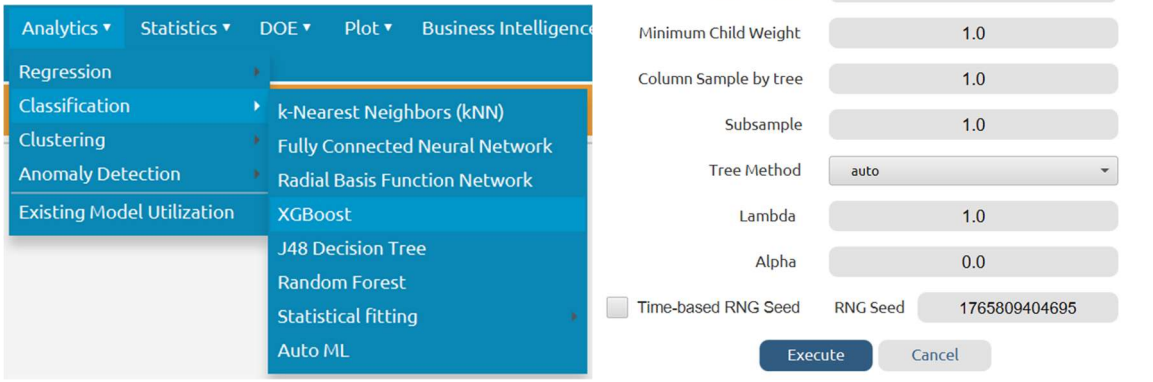
The results will appear on the output spreadsheet.

Step 7: Train the model

Create a new tab by pressing the "+" button on the bottom of the page with the name "TRAIN_MODEL(.fit)".

Import data into the input spreadsheet of the "TRAIN_MODEL(.fit)" tab from the output of the "NORMALIZE_TRAIN_SET" tab by right-clicking on the input spreadsheet and then choosing "Import from Spreadsheet".

Use the XGBoost method to train and fit the model: *Analytics → Classification → XGBoost*



The predictions will appear on the output spreadsheet.

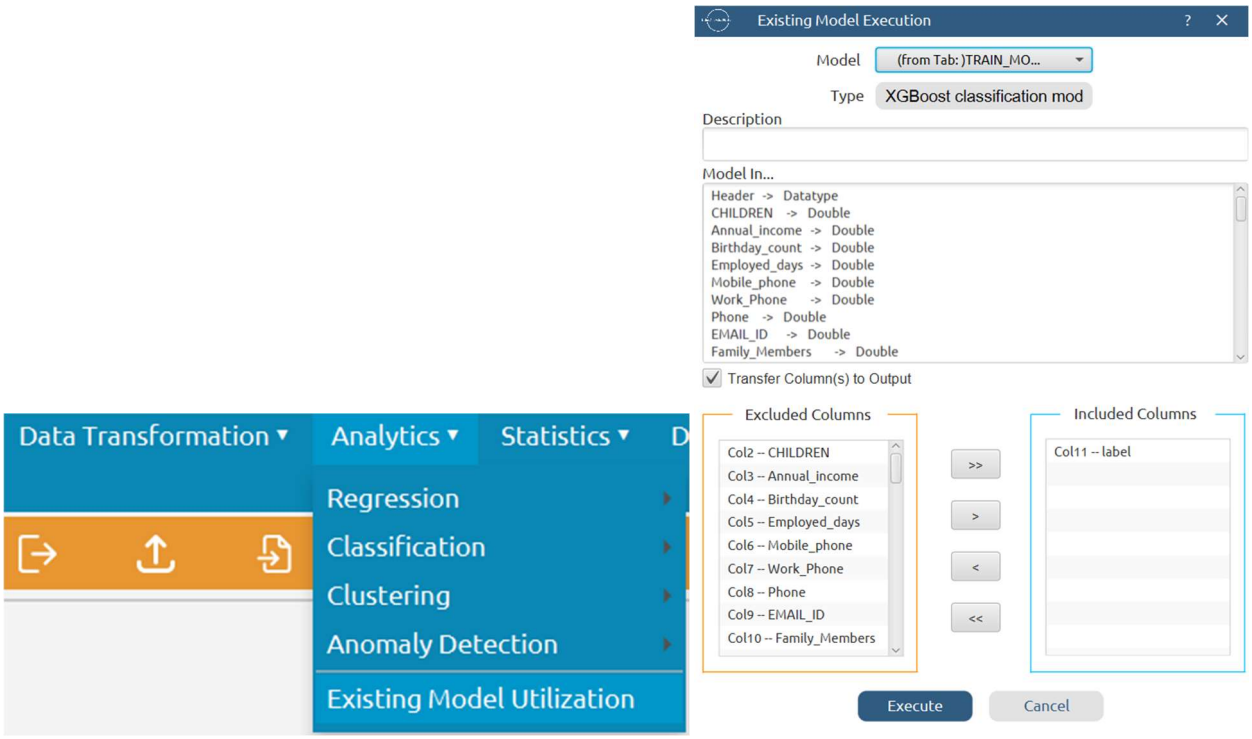
	Col1	Col2 (D)	Col3 (D)
User Header	User Row ID	label	Prediction
1	5009752	1.0	1.0
2	5009753	1.0	1.0
3	5009754	1.0	1.0
4	5009894	1.0	1.0
5	5010864	1.0	0.0
6	5010869	1.0	1.0
7	5021303	1.0	1.0
8	5021310	1.0	1.0
9	5021314	1.0	1.0
10	5021430	1.0	1.0
11	5021431	1.0	1.0
12	5021998	1.0	1.0
13	5022053	1.0	1.0
14	5022617	1.0	1.0
15	5023781	1.0	1.0

Step 8: Validate the model

Create a new tab by pressing the "+" button on the bottom of the page with the name "VALIDATE_MODEL(.predict)".

Import data into the input spreadsheet of the "VALIDATE_MODEL(.predict)" 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:) TRAIN_MODEL(.fit)*. Choose the column "label" to be transferred to the output spreadsheet.



The predictions will appear on the output spreadsheet.

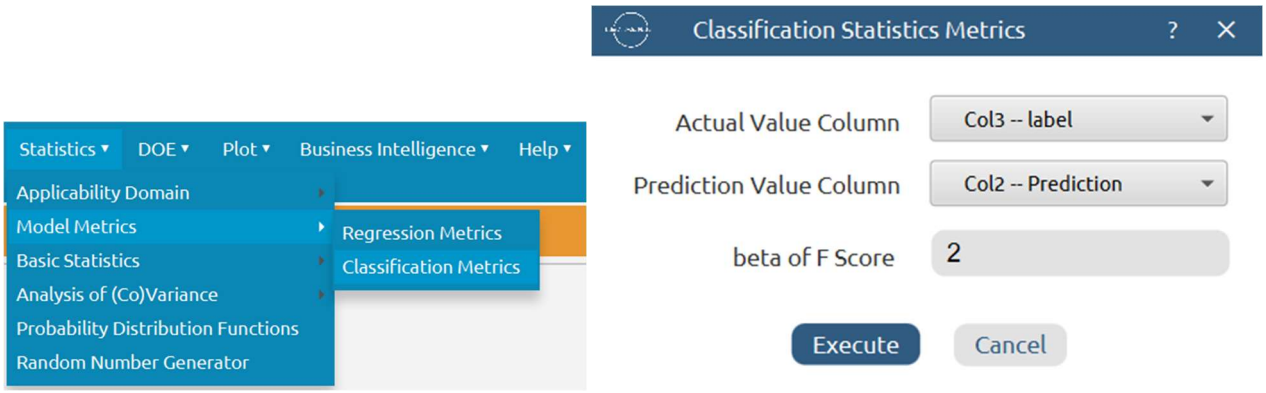
	Col1	Col2 (D)	Col3 (D)
User Header	User Row ID	Prediction	label
1	5008827	0.0	1.0
2	5009744	1.0	1.0
3	5009746	0.0	1.0
4	5009749	0.0	1.0
5	5010868	1.0	1.0
6	5018498	0.0	1.0
7	5018501	0.0	1.0
8	5018503	0.0	1.0
9	5028580	0.0	1.0
10	5033453	0.0	1.0
11	5033514	0.0	1.0
12	5036469	0.0	1.0
13	5042064	0.0	1.0
14	5045894	1.0	1.0
15	5045895	1.0	1.0

Step 9: Statistics calculation

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

Import data into the input spreadsheet of the "STATISTICS_ACCURACIES" tab from the output of the "VALIDATE_MODEL(.predict)" 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 (D)	Col3 (S)	Col4 (S)
User Header	User Row ID			
1			Predicted Class	Predicted Class
2			1.0	0.0
3	Actual Class	1.0	16	28
4	Actual Class	0.0	13	330
5				
6				
7	Classification Accuracy	0.8940568		
8				
9	Precision		0.5517241	0.9217877
10				
11	Recall/Sensitivity		0.3636364	0.9620991
12				
13	Specificity		0.9620991	0.3636364
14				
15	F1 Score		0.4383562	0.9415121
16				
17	F (beta=2)		0.3902439	0.9537572
18				
19	MCC	0.3927335		

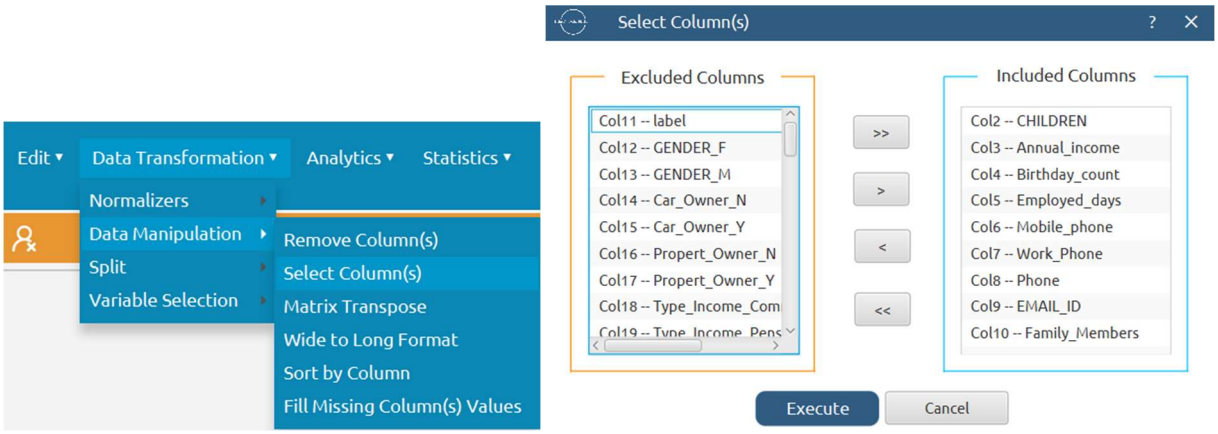
Step 10: Reliability check for each record of the test set

Step 10.a: Create the domain

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

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

Manipulate the data to exclude the target column "label" and the one-hot encoded columns: *Data Transformation → Data Manipulation → Select Column(s)*

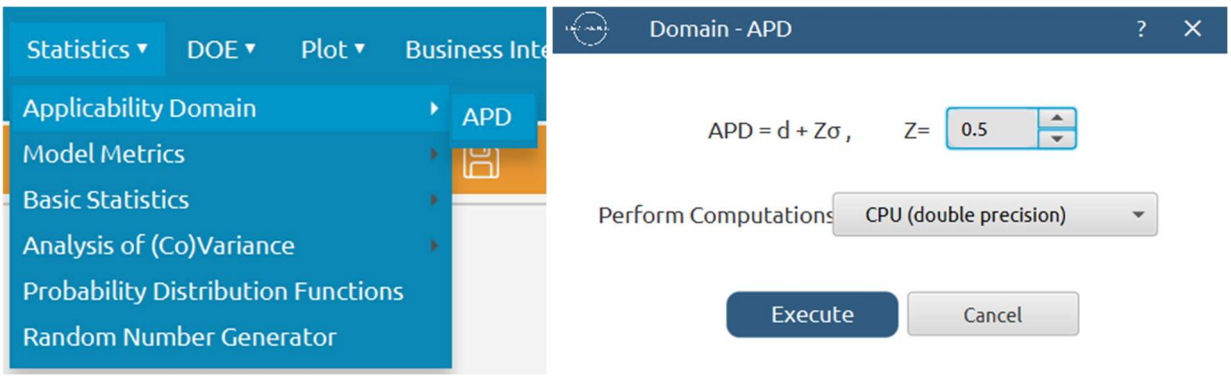


The results will appear on the output spreadsheet.

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

Import data into the input spreadsheet of the "DOMAIN" tab from the output of the "EXCLUDE_LABEL" 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.

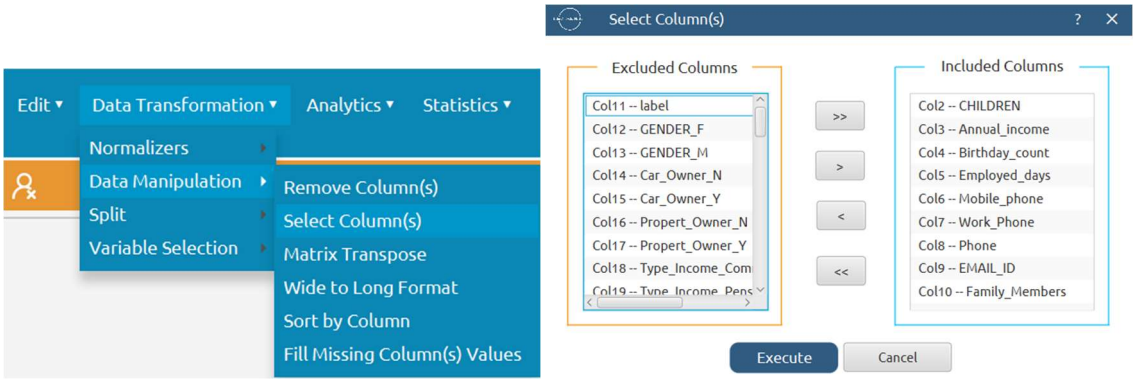
	Col1	Col2 (D)	Col3 (D)	Col4 (S)
User Header	User Row ID	Domain	APD	Prediction
1	5009752	0.0	2.9798623	reliable
2	5009753	0.0	2.9798623	reliable
3	5009754	0.0	2.9798623	reliable
4	5009894	0.0	2.9798623	reliable
5	5010864	0.0	2.9798623	reliable
6	5010869	0.0	2.9798623	reliable
7	5021303	0.0	2.9798623	reliable
8	5021310	0.0	2.9798623	reliable
9	5021314	0.0	2.9798623	reliable
10	5021430	0.0	2.9798623	reliable
11	5021431	0.0	2.9798623	reliable
12	5021998	0.0	2.9798623	reliable
13	5022053	0.0	2.9798623	reliable
14	5022617	0.0	2.9798623	reliable
15	5023781	0.0	2.9798623	reliable

Step 10.b: Check the test set reliability

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

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

Manipulate the data to exclude the target column "label" and the one-hot encoded columns: *Data Transformation → Data Manipulation → Select Column(s)*

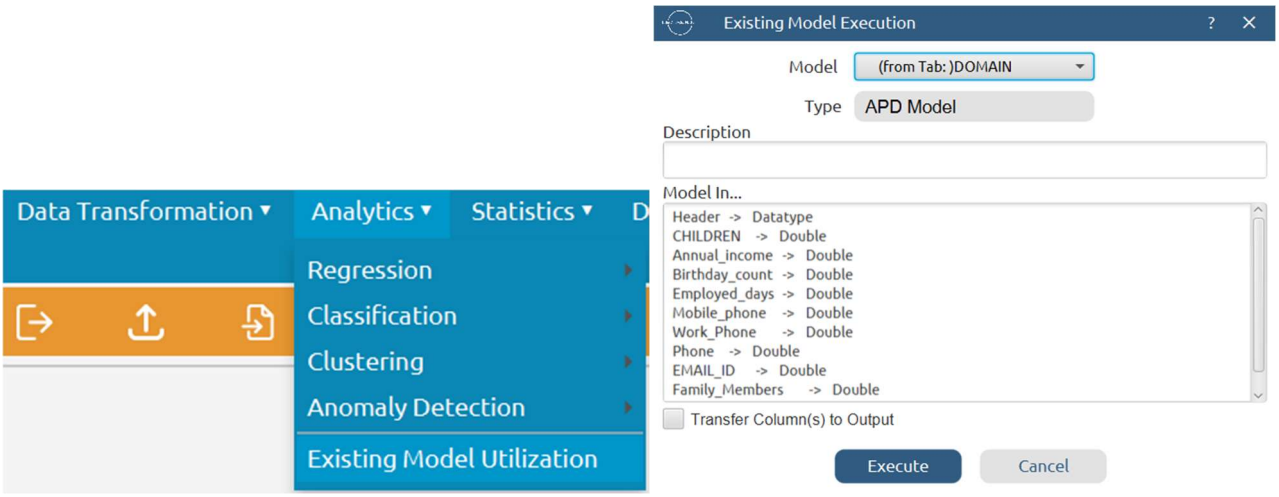


The results will appear on the output spreadsheet.

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 "EXCLUDE_LABEL_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:) DOMAIN*



The results will appear on the output spreadsheet.

	Col1	Col2 (D)	Col3 (D)	Col4 (S)
User Header	User Row ID	Domain	APD	Prediction
1	5008827	0.2945236	2.9798623	reliable
2	5009744	0.0	2.9798623	reliable
3	5009746	0.0562452	2.9798623	reliable
4	5009749	0.1035060	2.9798623	reliable
5	5010868	0.0	2.9798623	reliable
6	5018498	0.1401839	2.9798623	reliable
7	5018501	0.0472154	2.9798623	reliable
8	5018503	0.1401839	2.9798623	reliable
9	5028580	1.3404827	2.9798623	reliable
10	5033453	0.0214850	2.9798623	reliable
11	5033514	0.1765022	2.9798623	reliable
12	5036469	0.0553171	2.9798623	reliable
13	5042064	0.4014907	2.9798623	reliable
14	5045894	0.0	2.9798623	reliable
15	5045895	0.0	2.9798623	reliable

Final Isalos Workflow

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