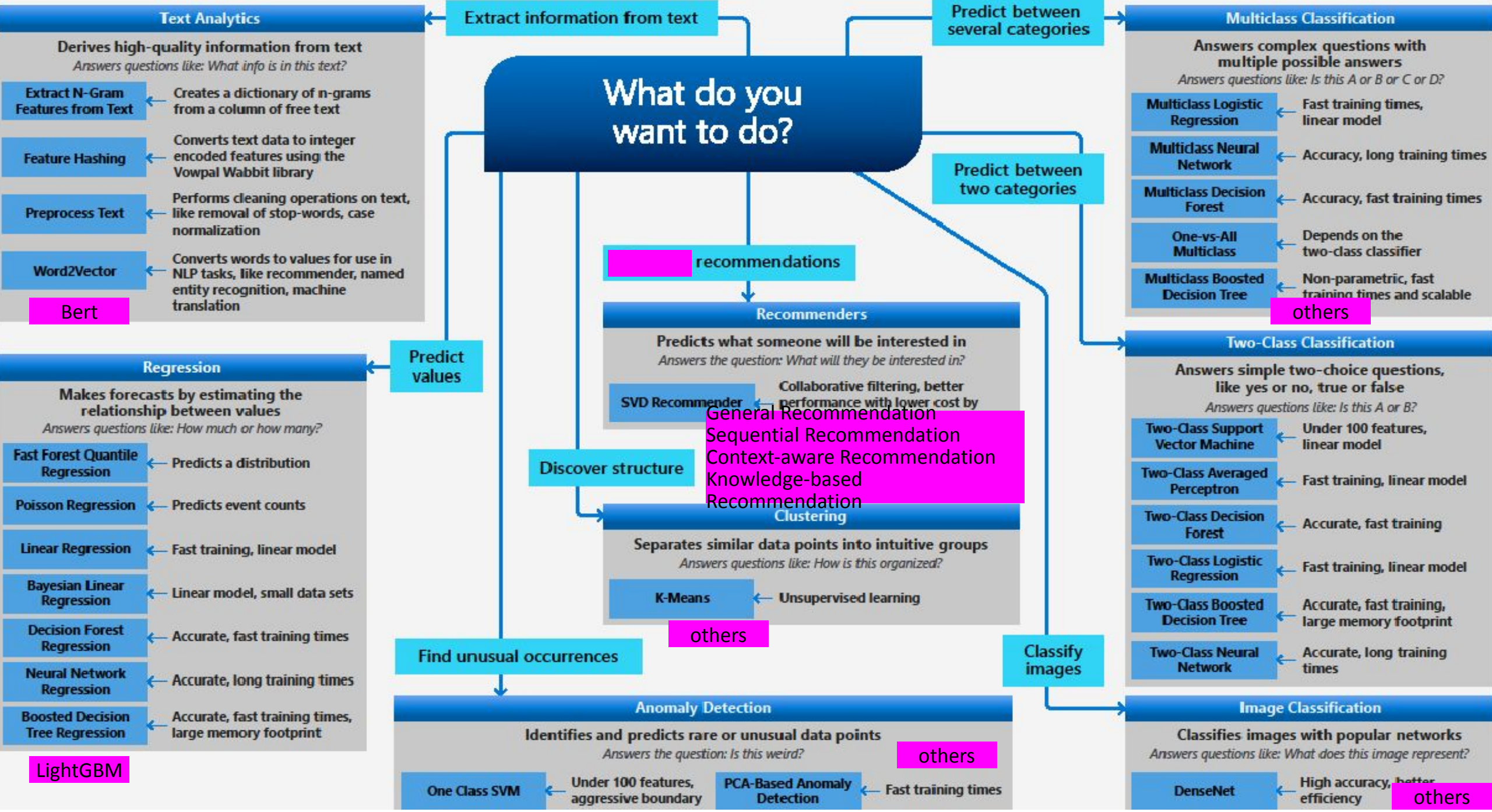
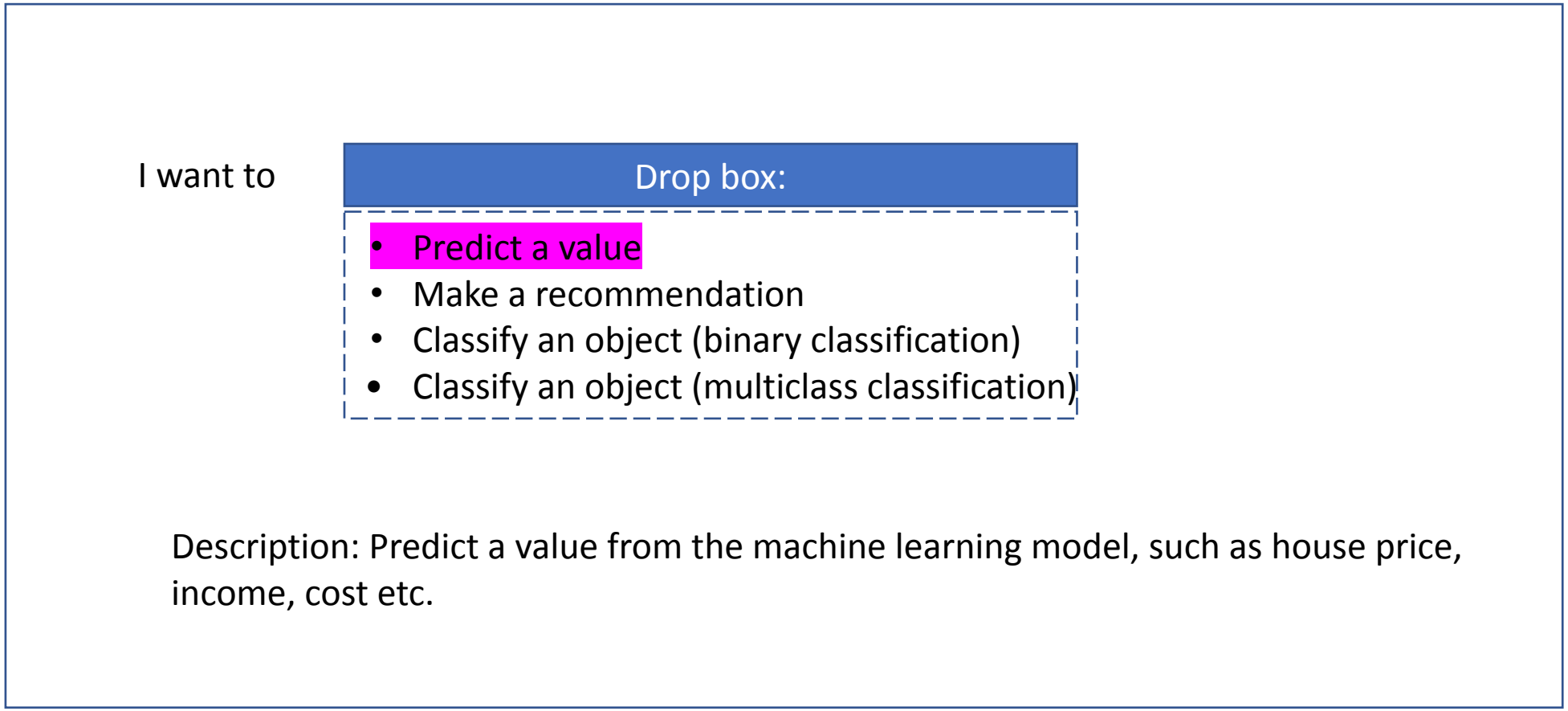
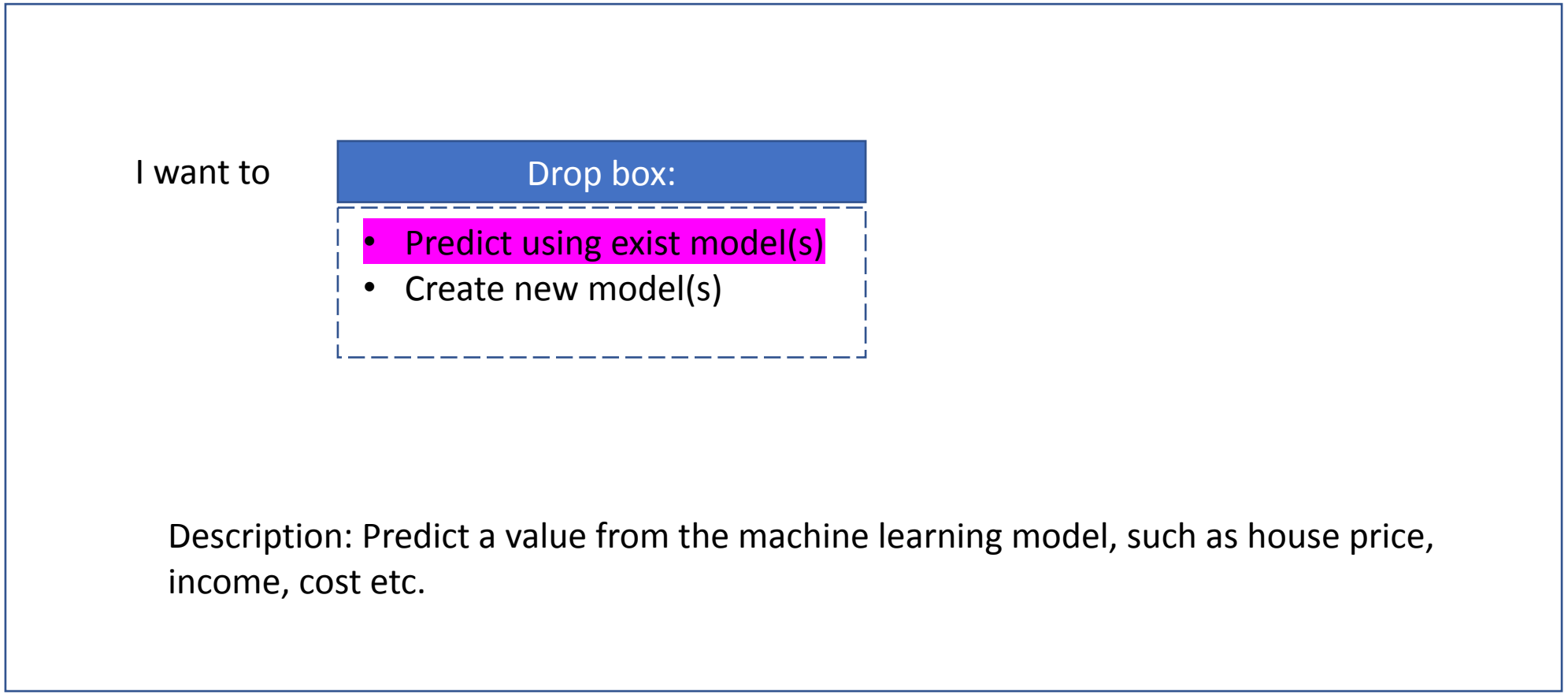
Machine Learning Use case


What do you want to do?
Extract information from text
Predict between several categories
Predict between two categories
Predict values
Discover structure
Find unusual occurrences
Classify images
recommendations
Text Analytics
Derives high-quality information from text
Answers questions like: What info is in this text?
Extract N-Gram Features from Text
Creates a dictionary of n-grams from a column of free text
Feature Hashing
Converts text data to integer encoded features using the Vowpal Wabbit library
Preprocess Text
Performs cleaning operations on text, like removal of stop-words, case normalization
Word2Vector
Converts words to values for use in NLP tasks, like recommender, named entity recognition, machine translation
Bert
Regression
Makes forecasts by estimating the relationship between values
Answers questions like: How much or how many?
Fast Forest Quantile Regression
Predicts a distribution
Poisson Regression
Predicts event counts
Linear Regression
Fast training, linear model
Bayesian Linear Regression
Linear model, small data sets
Decision Forest Regression
Accurate, fast training times
Neural Network Regression
Accurate, long training times
Boosted Decision Tree Regression
Accurate, fast training times, large memory footprint
LightGBM
Recommenders
Predicts what someone will be interested in
Answers the question: What will they be interested in?
SVD Recommender
Collaborative filtering, better performance with lower cost by
General Recommendation
Sequential Recommendation
Context-aware Recommendation
Knowledge-based Recommendation
Clustering
Separates similar data points into intuitive groups
Answers questions like: How is this organized?
K-Means
Unsupervised learning
others
Anomaly Detection
Identifies and predicts rare or unusual data points
Answers the question: Is this weird?
others
One Class SVM
Under 100 features, aggressive boundary
PCA-Based Anomaly Detection
Fast training times
Multiclass Classification
Answers complex questions with multiple possible answers
Answers questions like: Is this A or B or C or D?
Multiclass Logistic Regression
Fast training times, linear model
Multiclass Neural Network
Accuracy, long training times
Multiclass Decision Forest
Accuracy, fast training times
One-vs-All Multiclass
Depends on the two-class classifier
Multiclass Boosted Decision Tree
Non-parametric, fast training times and scalable
others
Two-Class Classification
Answers simple two-choice questions, like yes or no, true or false
Answers questions like: Is this A or B?
Two-Class Support Vector Machine
Under 100 features, linear model
Two-Class Averaged Perceptron
Fast training, linear model
Two-Class Decision Forest
Accurate, fast training
Two-Class Logistic Regression
Fast training, linear model
Two-Class Boosted Decision Tree
Accurate, fast training, large memory footprint
Two-Class Neural Network
Accurate, long training times
Image Classification
Classifies images with popular networks
Answers questions like: What does this image represent?
DenseNet
High accuracy, better efficiency
others

Main UI


I want to
Drop box:
Predict a value
Make a recommendation
Classify an object (binary classification)
Classify an object (multiclass classification)
Description: Predict a value from the machine learning model, such as house price, income, cost etc.

Predict a value UI



Predict using exist model(s) UI

Button: Confirm

Select one or more exist models

user ID	project ID	create time	model type	predict colume	model name	model_accuracy	recomended stars	Please select
0001	001	2021-12-06	value prediction	price	Linear Regression	mean_absolute_error=125636 root_mean_squared_error=195381 median_absolute_error=87607 r2=0.987	**	☑
0001	001	2021-12-06	value prediction	price	Decision Tree Regression	mean_absolute_error=96879 root_mean_squared_error=172768 median_absolute_error=55897 r2=0.89	***	☑
0001	001	2021-12-06	value prediction	price	Boosted Decision Tree Regression	mean_absolute_error=73303 root_mean_squared_error=125287 median_absolute_error=47525.0 r2=0.76	*****	☑
0001	002	2021-12-06	value prediction	price	Linear Regression	mean_absolute_error=185625 root_mean_squared_error=165354 median_absolute_error=97501 r2=1.257	*	☐
0001	002	2021-12-06	value prediction	price	Decision Tree Regression	mean_absolute_error=106567 root_mean_squared_error=199788 median_absolute_error=86546 r2=0.957	**	☐
0001	002	2021-12-06	value prediction	price	Boosted Decision Tree Regression	mean_absolute_error=88624 root_mean_squared_error=240258 median_absolute_error=65524 r2=0.92	**	☐

* table location: s3://dev-demo-land-area/autoML/test_prepare/regression_exist_models.csv

Predict using exist model(s) UI

House data view

TO use the selected model(s), your source should look like:

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	grade
1	7129300520	20141013T000000	221900	3	1	1180	5650	1	0	0	3	7
2	6414100192	20141209T000000	538000	3	2.25	2570	7242	2	0	0	3	7
3	5631500400	20150225T000000	180000	2	1	770	10000	1	0	0	3	6
4	2487200875	20141209T000000	604000	4		1960	5000	1	0	0	5	7
5	1954400510	20150218T000000	510000	3	2	1680	8080	1	0	0	3	8
6	7237550310	20140512T000000	1.23E+06	4	4.5	5420	101930	1	0	0	3	11
7	1321400060	20140627T000000	257500		2.25	1715	6819	2	0	0	3	7
8	2008000270	20150115T000000	291850	3	1.5	1060	9711	1	0	0	3	7
9	2414600126	20150415T000000	229500	3	1	1780	7470	1	0	0	3	7
10	3793500160	20150312T000000	323000	3	2.5	1890	6560	2	0	0	3	7
11	1736800520	20150403T000000	662500	3	2.5	3560	9796	1	0	0	3	8
12	9212900260	20140527T000000		2	1	1160	6000	1	0	0	4	7
13	114101516	20140528T000000	310000	3	1	1430	19901	1.5	0	0	4	7
14	6054650070	20141007T000000	400000	3	1.75	1370	9680	1	0	0	4	7
15	1175000570	20150312T000000	530000	5	2	1810	4850	1.5	0	0	3	7

Select or Create a Source: Drop box: Button: Confirm

*House data: s3://dev-demo-land-area/autoML/test_prepare/kc_house_data.csv

Predict using exist model(s) UI

Your source preview (1000 rows):

Button: Confirm

House data view

Column1	column2	column3	column4	column5	column6	column7	.....	columnN	Predict Value

Button: Submit job

Click to create json file

*House data(show different rows): s3://dev-demo-land-area/autoML/test_prepare/kc_house_data.csv
*json file saved: s3://dev-demo-land-area/autoML/requests/history/prediction_0001_001_Scoring_20211206.json
* copy json file to: s3://dev-demo-land-area/autoML/requests/queue/

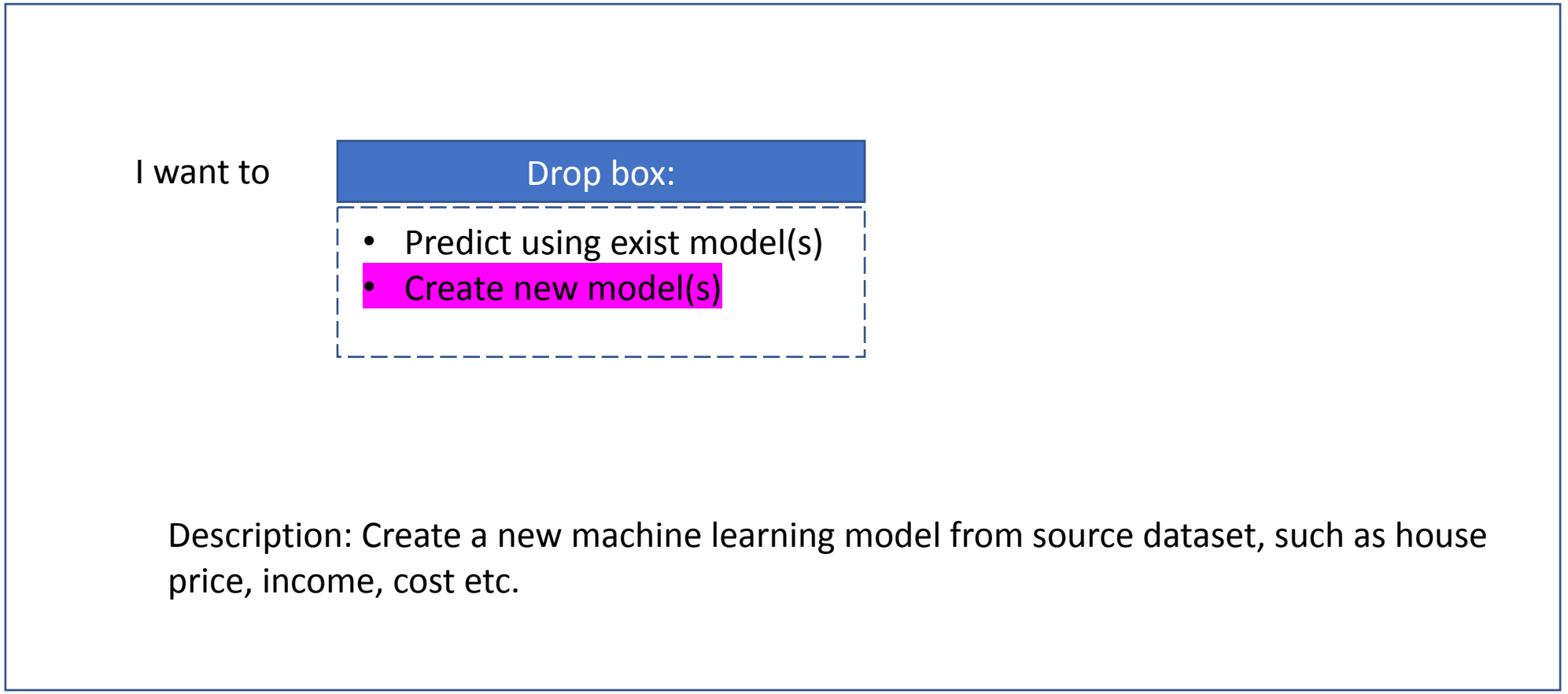
Predict using exist model(s) UI

Your prediction preview (1000 rows):

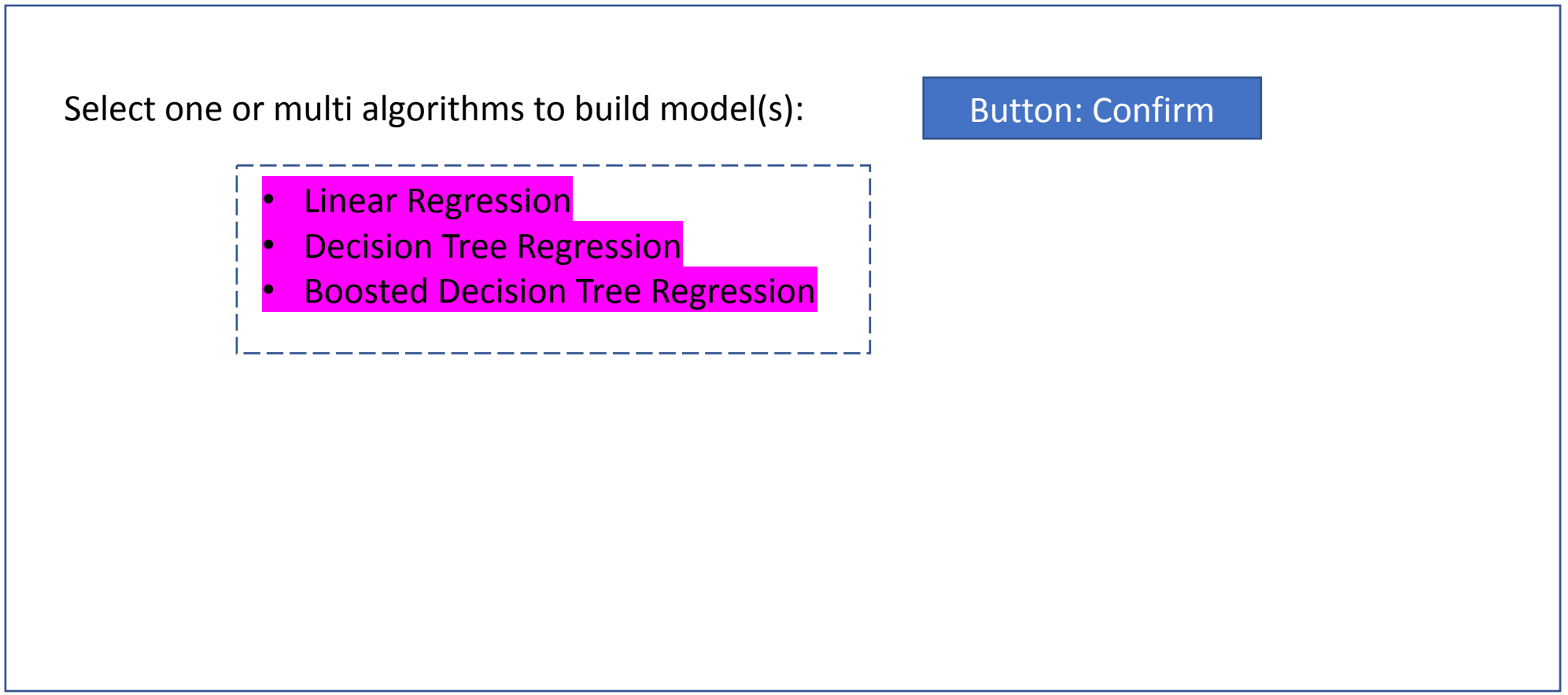
Column1	column2	column3	column4	column5	column6	column7	……	columnN	Predict Value

*Data saved: s3://dev-demo-land-area/autoML/users/0001/Regression/Scoring/001/score/score_output_Boosted Decision Tree Regression.csv
Data saved: s3://dev-demo-land-area/autoML/users/0001/Regression/Scoring/001/score/score_output_Decision Tree Regression.csv
Data saved: s3://dev-demo-land-area/autoML/users/0001/Regression/Scoring/001/score/score_output_Linear Regression.csv

Predict a value UI



Predict a value UI



Create new model(s)

Select or Create a Source: Drop box: Button: Confirm

Your source should look like:

Column1	column2	column3	column4	column5	column6	column7	……	columnN	Predict Value

Create new model(s)

Your source preview (1000 rows):

Button: Confirm

House data view

Column1	column2	column3	column4	column5	column6	column7	……	columnN	Predict Value

Button: Submit job

Click to create json file

*House data(show different rows): s3://dev-demo-land-area/autoML/test_prepare/kc_house_data.csv
*json file saved: s3://dev-demo-land-area/autoML/requests/history/prediction_0001_001_Training_20211206.json
* copy json file to: s3://dev-demo-land-area/autoML/requests/queue/

Create new model(s)

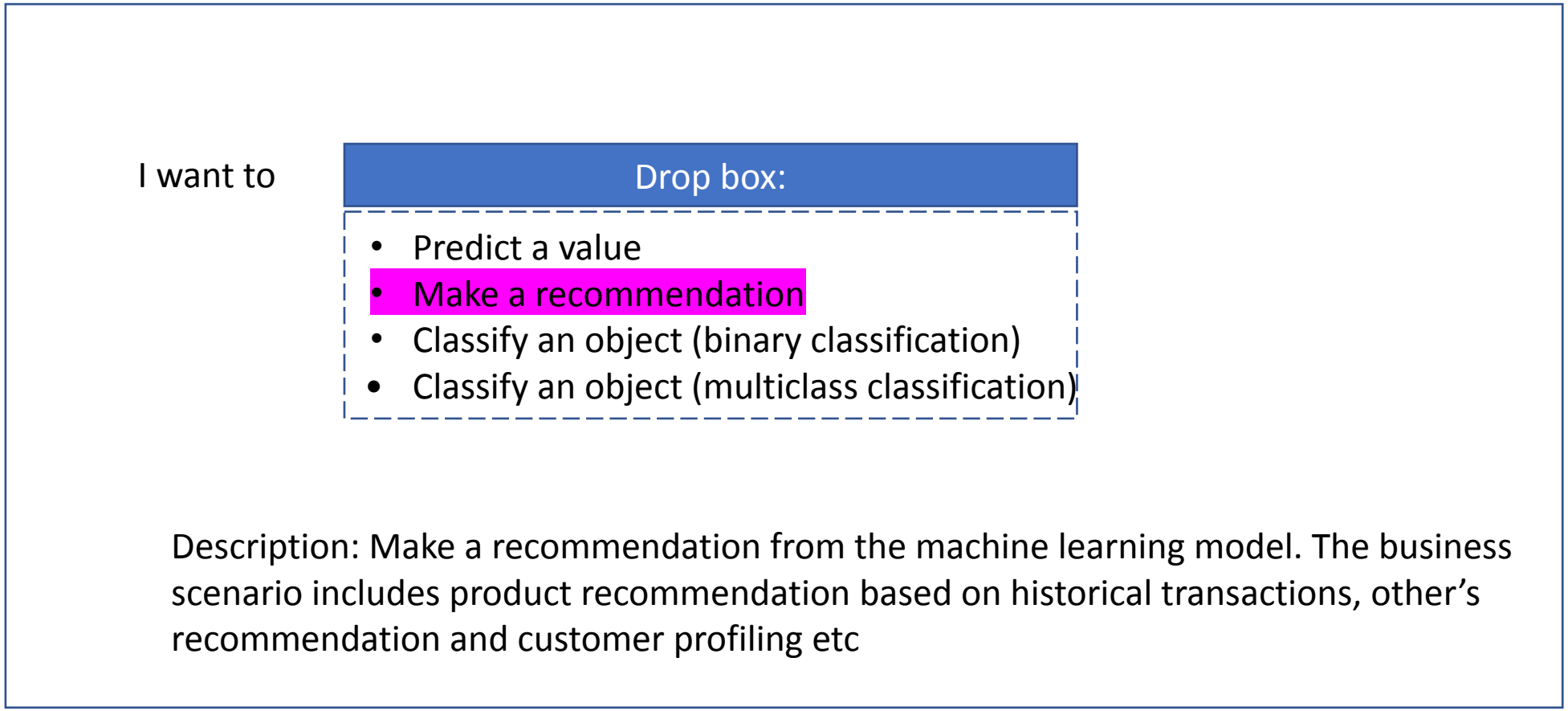
Button: Predict values

Created models overview

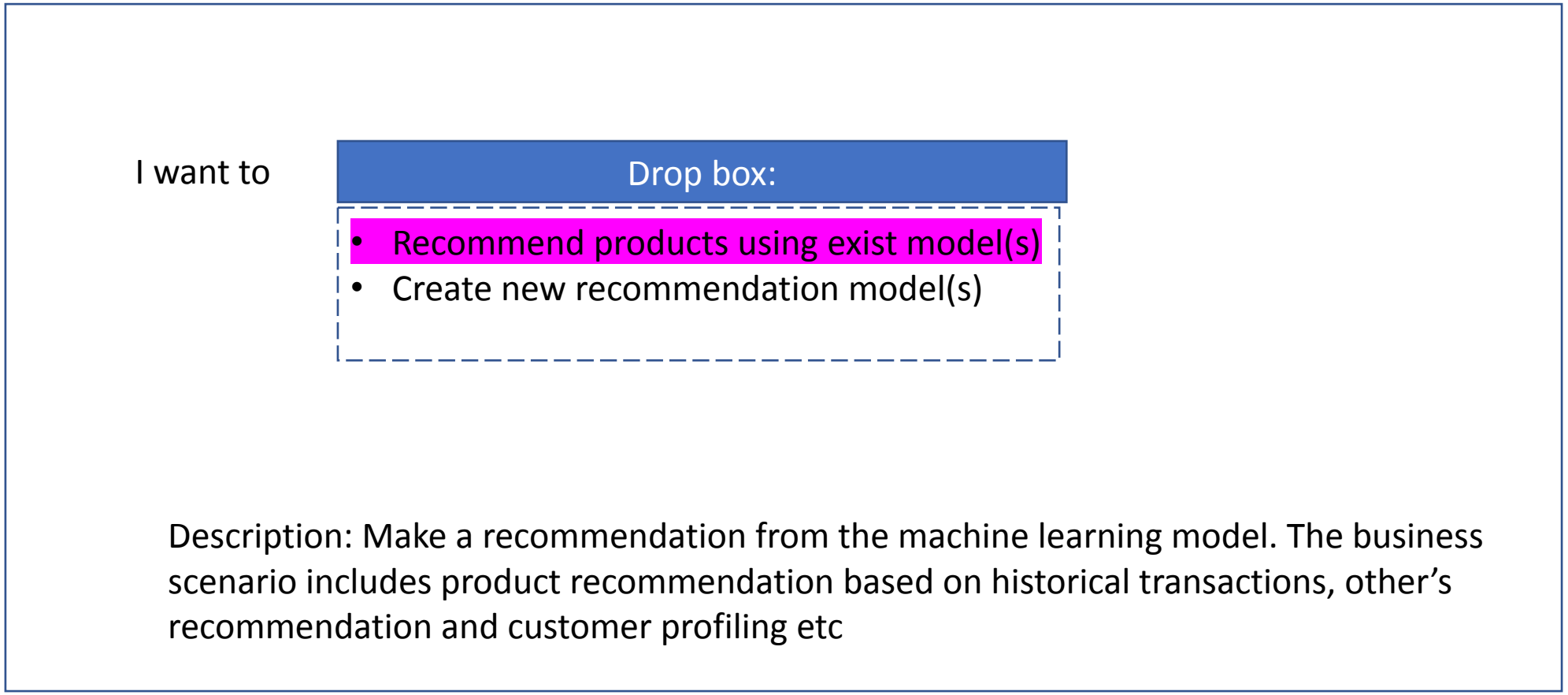
user ID	project ID	create time	model type	predict colume	model name	model_accuracy	recomended stars
0001	001	2021-12-06	value prediction	price	Linear Regression	mean_absolute_error=125636 root_mean_squared_error=195381 median_absolute_error=87607 r2=0.987	**
0001	001	2021-12-06	value prediction	price	Decision Tree Regression	mean_absolute_error=96879 root_mean_squared_error=172768 median_absolute_error=55897 r2=0.89	***
0001	001	2021-12-06	value prediction	price	Boosted Decision Tree Regression	mean_absolute_error=73303 root_mean_squared_error=125287 median_absolute_error=47525.0 r2=0.76	*****

* table location: s3://dev-demo-land-area/autoML/test_prepare/regression_created_model_overview.csv

Main UI


I want to
Drop box:
• Predict a value
• Make a recommendation
• Classify an object (binary classification)
• Classify an object (multiclass classification)
Description: Make a recommendation from the machine learning model. The business scenario includes product recommendation based on historical transactions, other's recommendation and customer profiling etc

Recommendation UI



Recommend products using exist model(s) UI

Button: Confirm

Select one or more exist models

user ID	project ID	create time	model type	predict colume	model name	model_accuracy	recomended stars	please select
0001	001	2021-12-06	Recommendation	ProductKey	Factorization Machine Model	auc=0.82	*****	☑
0001	001	2021-12-06	Recommendation	ProductKey	Logistic Regression Model	auc=0.75	***	☐
0001	001	2021-12-06	Recommendation	ProductKey	Automatic Feature Interaction Model	auc=0.72	***	☐
0001	002	2021-12-06	Recommendation	ProductKey	Factorization Machine Model	auc=0.78	*	☐
0001	002	2021-12-06	Recommendation	ProductKey	Logistic Regression Model	auc=0.65	**	☐
0001	002	2021-12-06	Recommendation	ProductKey	Automatic Feature Interaction Model	auc=0.70	**	☐

* table location: s3://dev-demo-land-area/autoML/test_prepare/recsys_exist_models.csv

Recommend products using exist model(s) UI

TO use the selected model(s), your source should look like:

	CustomerKey	GeographyKey	MaritalStatus	Gender	YearlyIncome	TotalChildren	NumberChildrenAtHome	HouseOwnerFlag	NumberCarsOwned
1	21626	648	M	F	70000	5	5	1	2
2	21627	302	M	F	70000	5	5	1	2
3	21628	612	S	F	40000	0	0	0	1
4	21629	52	S	M	50000	3	3	0	1
5	21630	627	M	F	50000	3	3	1	2
6	21631	49	M	M	50000	3	3	1	3
7	21632	312	M	F	50000	3	3	1	3
8	21633	633	M	M	50000	3	3	1	3
9	21634	614	M	F	50000	3	3	1	3
10	21635	352	M	M	50000	3	3	1	3
11	21636	385	M	F	50000	3	3	1	3

Select or Create a Source: Drop box: Button: Confirm

*recommendation data: s3://dev-demo-land-area/autoML/test_prepare/user_data.csv

Recommend products using exist model(s) UI

Your source preview (100 rows):

Button: Confirm

	CustomerKey	GeographyKey	MaritalStatus	Gender	YearlyIncome	TotalChildren	NumberChildrenAtHome	HouseOwnerFlag	NumberCarsOwned
1	21626	648	M	F	70000	5	5	1	2
2	21627	302	M	F	70000	5	5	1	2
3	21628	612	S	F	40000	0	0	0	1
4	21629	52	S	M	50000	3	3	0	1
5	21630	627	M	F	50000	3	3	1	2
6	21631	49	M	M	50000	3	3	1	3
7	21632	312	M	F	50000	3	3	1	3
8	21633	633	M	M	50000	3	3	1	3

Button: Submit job

Click to create json file

*recommendation data(show different rows): s3://dev-demo-land-area/autoML/test_prepare/user_data.csv
*json file saved: s3://dev-demo-land-area/autoML/requests/history/recommendation_0001_001_Scoring_20211206.json
* copy json file to: s3://dev-demo-land-area/autoML/requests/queue/

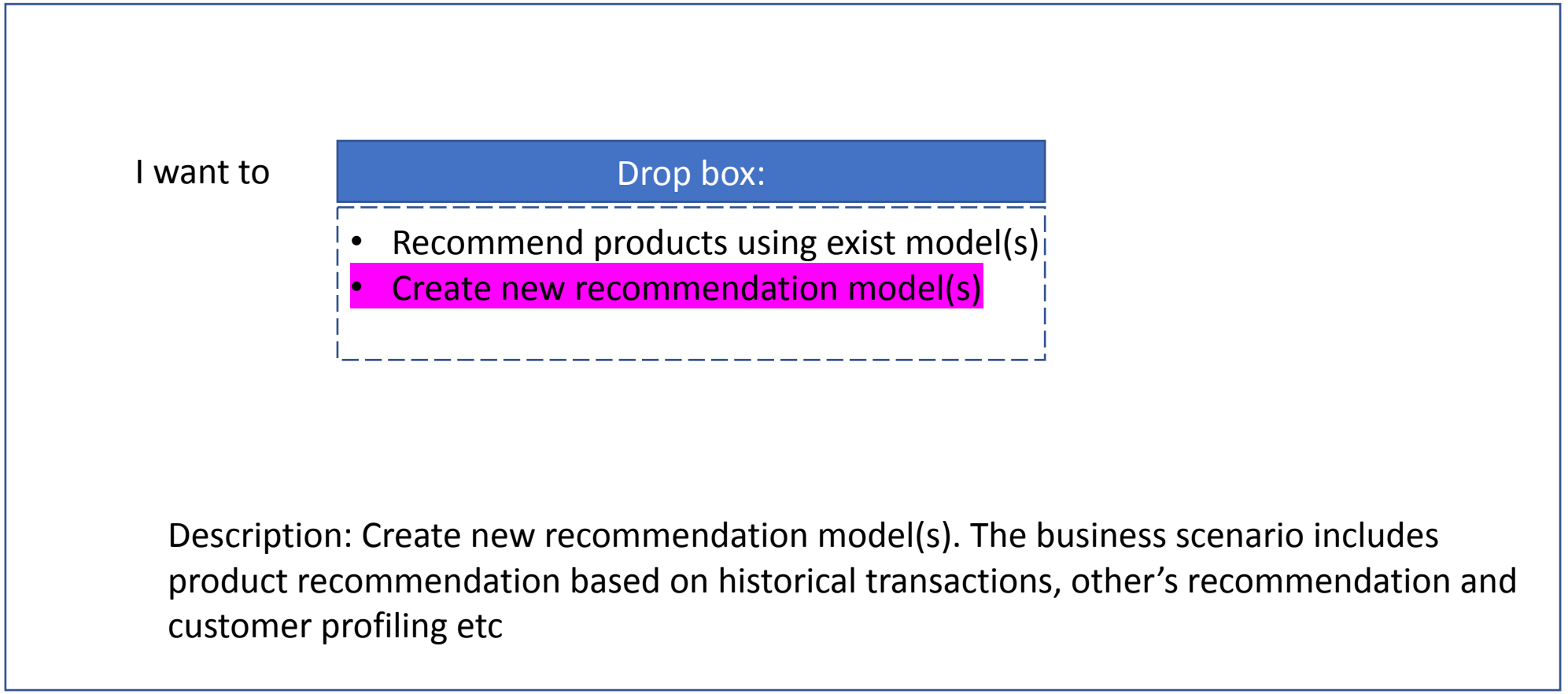
Recommend products using exist model(s) UI

Your prediction preview (1000 rows):

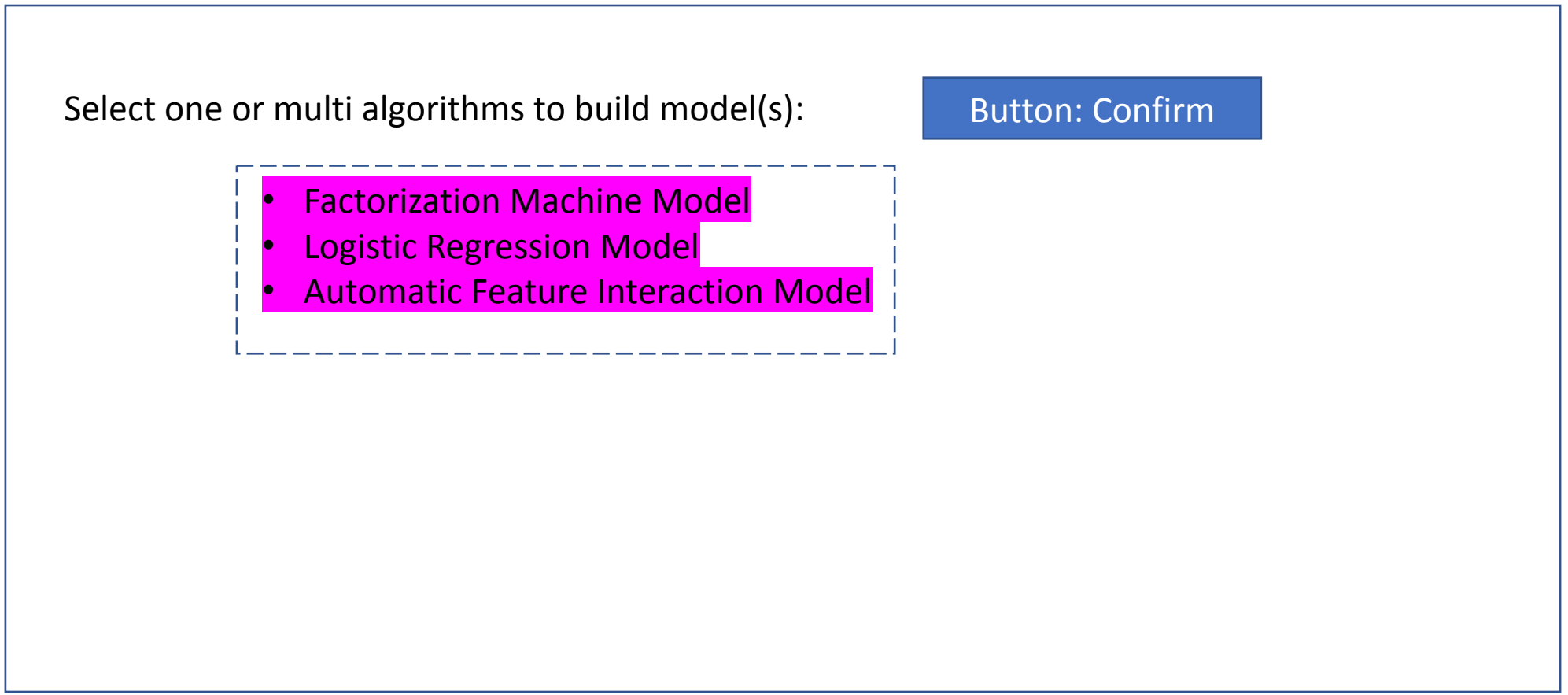
CustomerKey	ProductKey	Score	Order	GeographyKey	MaritalStatus	Gender	YearlyIncome	TotalChildren	ChildrenAtHome	HouseOwnerFlag	CarsOwned
16031	35	0.9861404895782471	1	27	M	M	30000	2	0	0	2
16031	79	0.9861404895782471	2	27	M	M	30000	2	0	0	2
16031	169	0.9861404895782471	3	27	M	M	30000	2	0	0	2
16031	187	0.9861404895782471	4	27	M	M	30000	2	0	0	2
16031	190	0.9861404895782471	5	27	M	M	30000	2	0	0	2
16032	9	0.9587547183036804	1	2	M	M	10000	2	1	1	2
16032	10	0.9587547183036804	2	2	M	M	10000	2	1	1	2
16032	21	0.9587547183036804	3	2	M	M	10000	2	1	1	2
16032	32	0.9587547183036804	4	2	M	M	10000	2	1	1	2
16032	33	0.9587547183036804	5	2	M	M	10000	2	1	1	2
16033	35	0.9885239005088806	1	248	M	F	10000	0	0	1	0
16033	79	0.9885239005088806	2	248	M	F	10000	0	0	1	0
16033	169	0.9885239005088806	3	248	M	F	10000	0	0	1	0
16033	187	0.9885239005088806	4	248	M	F	10000	0	0	1	0

*Data saved: s3://dev-demo-land-area/autoML/users/0001/Recommendation/Scoring/001/score/score_output.csv

Recommendation UI



Recommendation UI



Create new recommendation model(s) UI

TO create recommendation models, your need user, item and user-item interaction datasets.

Select or create a **user data:**

Drop box:

Your source preview (100 rows):

Button: Confirm

	CustomerKey	GeographyKey	MaritalStatus	Gender	YearlyIncome	TotalChildren	NumberChildrenAtHome	HouseOwnerFlag	NumberCarsOwned
1	21626	648	M	F	70000	5	5	1	2
2	21627	302	M	F	70000	5	5	1	2
3	21628	612	S	F	40000	0	0	0	1
4	21629	52	S	M	50000	3	3	0	1

Select or create an **item data**:

Drop box:

Button: Confirm

Select or create a **user-item** interaction data:

Drop box:

Button: Confirm

*recommendation data: s3://dev-demo-land-area/autoML/test_prepare/user_data.csv
*recommendation data: s3://dev-demo-land-area/autoML/test_prepare/item_data.csv
*recommendation data: s3://dev-demo-land-area/autoML/test_prepare/user_item_interaction.csv

Create new recommendation model(s) UI

Select or create a user-item interaction data:

Drop box:

Button: Confirm

CustomerKey	ProductKey	SalesTerritoryKey	OrderQuantity	SalesAmount
13917	536	4	1	29.99
13917	528	4	1	4.99
13917	217	4	1	34.99
12164	359	4	1	2294.99
12164	537	4	1	35
11538	363	4	1	2294.99
11538	485	4	1	21.98
28948	561	9	1	2384.07
28948	214	9	1	34.99

Button: Submit job

Click to create json file

*json file saved: s3://dev-demo-land-area/autoML/requests/history/recommendation_0001_001_Training_20211206.json
* copy json file to: s3://dev-demo-land-area/autoML/requests/queue/

Main UI

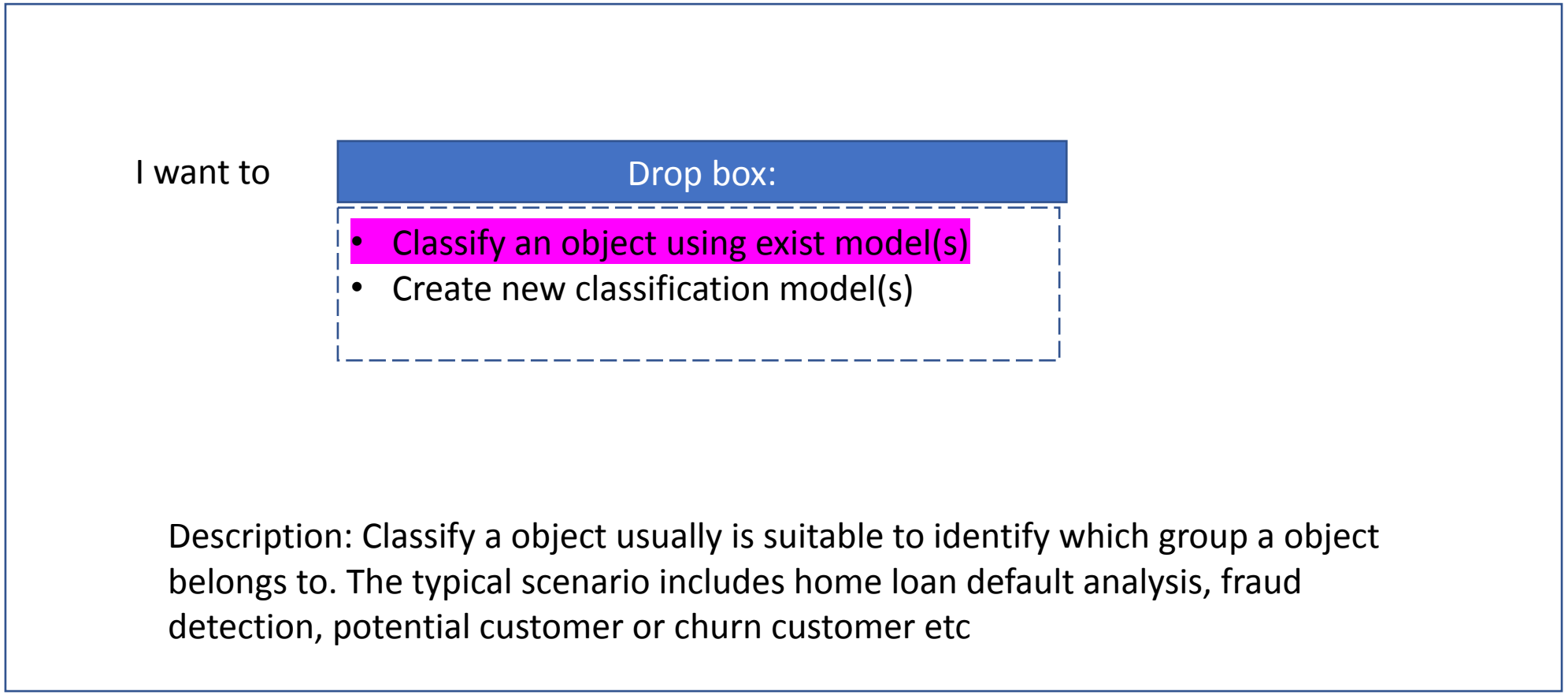
I want to

Drop box:

- Predict a value
- Make a recommendation
- Classify an object (binary classification)
- Classify an object (multiclass classification)

Description: Classify a object usually is suitable to identify which group a object belongs to. The typical scenario includes home loan default analysis, fraud detection, potential customer or churn customer etc

Classification UI



Classify an object using exist model(s) UI

Button: Confirm

Select one or more exist models

user ID	project ID	create time	model type	predict colume	model name	model_accuracy	recomended stars	please select
0001	001	2021-12-06	Binary_Classification	healthy	Random Forest Classifier	accuracy_score:0.970 f1_score:0.971 precision_score:0.956 recall_score:0.986 roc_auc_score:0.969	*****	☑
0001	001	2021-12-06	Binary_Classification	healthy	Gradient Boosting Classifier	accuracy_score:0.968 f1_score:0.969 precision_score:0.955 recall_score:0.984 roc_auc_score:0.967	*****	☑
0001	001	2021-12-06	Binary_Classification	healthy	Logistic Regression Classifier	accuracy_score:0.955 f1_score:0.957 precision_score:0.936 recall_score:0.979 roc_auc_score:0.955	****	☑
0001	001	2021-12-06	Binary_Classification	healthy	Support Vector Machine Classifier	accuracy_score:0.962 f1_score:0.963 precision_score:0.941 recall_score:0.987 roc_auc_score:0.961	****	☑
0001	001	2021-12-06	Binary_Classification	healthy	KNeighbors Classifier	accuracy_score:0.941 f1_score:0.945 precision_score:0.901 recall_score:0.994 roc_auc_score:0.940	***	☐
0001	001	2021-12-06	Binary_Classification	healthy	Naive Bayes Classifier	accuracy_score:0.912 f1_score:0.914 precision_score:0.904 recall_score:0.925 roc_auc_score:0.911	***	☐
0001	001	2021-12-06	Binary_Classification	healthy	Decision Tree Classifier	accuracy_score:0.910 f1_score:0.9178 precision_score:0.863 recall_score:0.980 roc_auc_score:0.909	***	☐

* table location: s3://dev-demo-land-area/autoML/test_prepare/binary_classification_exist_models.csv

Classify an object using exist model(s) UI

TO use the selected model(s), your source should look like:

	age	gender	height_cm	weight_kg	body fat	diastolic	systolic	gripForce	sit and bend forward_cm	sit-ups counts	broad jump_cm	healthy
1	25.0	M	165.0	55.8	15.7	77.0	126.0	36.4	16.3	53.0	229.0	1
2	42.0	F	164.5	63.7	32.2	72.0	135.0	22.7	0.8	18.0	146.0	0
3	42.0	M	169.2	65.4	19.3	63.0	110.0	43.5	16.0	68.0	211.0	1
4	57.0	F	153.0	49.0	20.9	69.0	106.0	21.5	30.0	0.0	90.0	0
5	45.0	F	159.0	63.1	30.9	93.0	144.0	34.1	19.0	30.0	155.0	1
6	26.0	M	179.9	71.5	9.7	64.0	135.0	59.6	17.8	61.0	239.0	1
7	59.0	F	155.9	62.7	30.2	76.0	143.0	36.8	29.1	25.0	122.0	1
8	38.0	M	166.7	67.3	23.2	70.0	111.0	26.1	19.7	62.0	215.0	0
9	47.0	F	158.3	53.5	29.2	70.0	117.0	25.9	8.1	32.0	151.0	0
10	48.0	M	175.8	84.5	31.4	83.0	125.0	33.8	3.7	24.0	211.0	0
11	50.0	F	159.8	57.1	24.4	63.0	103.0	30.8	24.4	30.0	143.0	1
12	28.0	F	159.5	51.54	24.5	82.0	123.0	37.2	23.0	37.0	145.0	1
13	30.0	M	172.1	79.5	26.7	91.0	148.0	34.7	-2.0	40.0	192.0	0
14	21.0	M	172.5	66.4	12.5	82.0	130.0	51.2	6.3	46.0	178.0	0
15	31.0	M	177.5	79.5	23.0	90.0	148.0	51.2	18.4	62.0	208.0	1
16	32.0	M	178.0	84.5	21.2	68.0	130.0	52.3	16.2	62.0	232.0	1

Select or Create a Source: Drop box: Button: Confirm

*recommendation data: s3://dev-demo-land-area/autoML/test_prepare/body_performance_healthy.csv

Classify an object using exist model(s) UI

Your source preview (100 rows):

Button: Confirm

	age	gender	height_cm	weight_kg	body fat	diastolic	systolic	gripForce	sit and bend forward_cm	sit-ups counts	broad jump_cm	healthy
1	25.0	M	165.0	55.8	15.7	77.0	126.0	36.4	16.3	53.0	229.0	1
2	42.0	F	164.5	63.7	32.2	72.0	135.0	22.7	0.8	18.0	146.0	0
3	42.0	M	169.2	65.4	19.3	63.0	110.0	43.5	16.0	68.0	211.0	1
4	57.0	F	153.0	49.0	20.9	69.0	106.0	21.5	30.0	0.0	90.0	0
5	45.0	F	159.0	63.1	30.9	93.0	144.0	34.1	19.0	30.0	155.0	1
6	26.0	M	179.9	71.5	9.7	64.0	135.0	59.6	17.8	61.0	239.0	1
7	59.0	F	155.9	62.7	30.2	76.0	143.0	36.8	29.1	25.0	122.0	1
8	38.0	M	166.7	67.3	23.2	70.0	111.0	26.1	19.7	62.0	215.0	0
9	47.0	F	158.3	53.5	29.2	70.0	117.0	25.9	8.1	32.0	151.0	0
10	48.0	M	175.8	84.5	31.4	83.0	125.0	33.8	3.7	24.0	211.0	0

Button: Submit job

Click to create json file

*recommendation data(show different rows): s3://dev-demo-land-area/autoML/test_prepare/body_performance_healthy.csv
*json file saved: s3://dev-demo-land-area/autoML/requests/history/binary_classification_0001_001_Scoring_20211206.json
* copy json file to: s3://dev-demo-land-area/autoML/requests/queue/

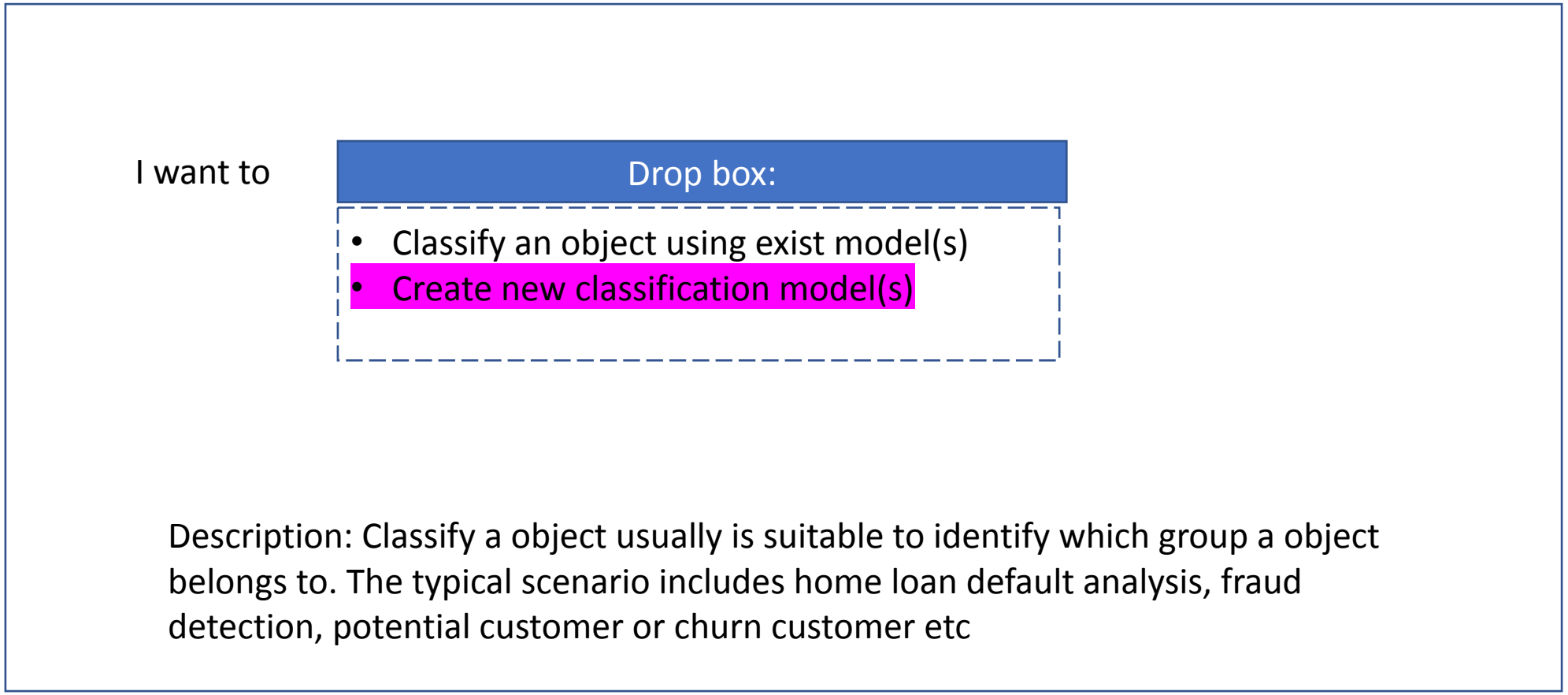
Classify an object using exist model(s) UI

Your prediction preview (1000 rows):

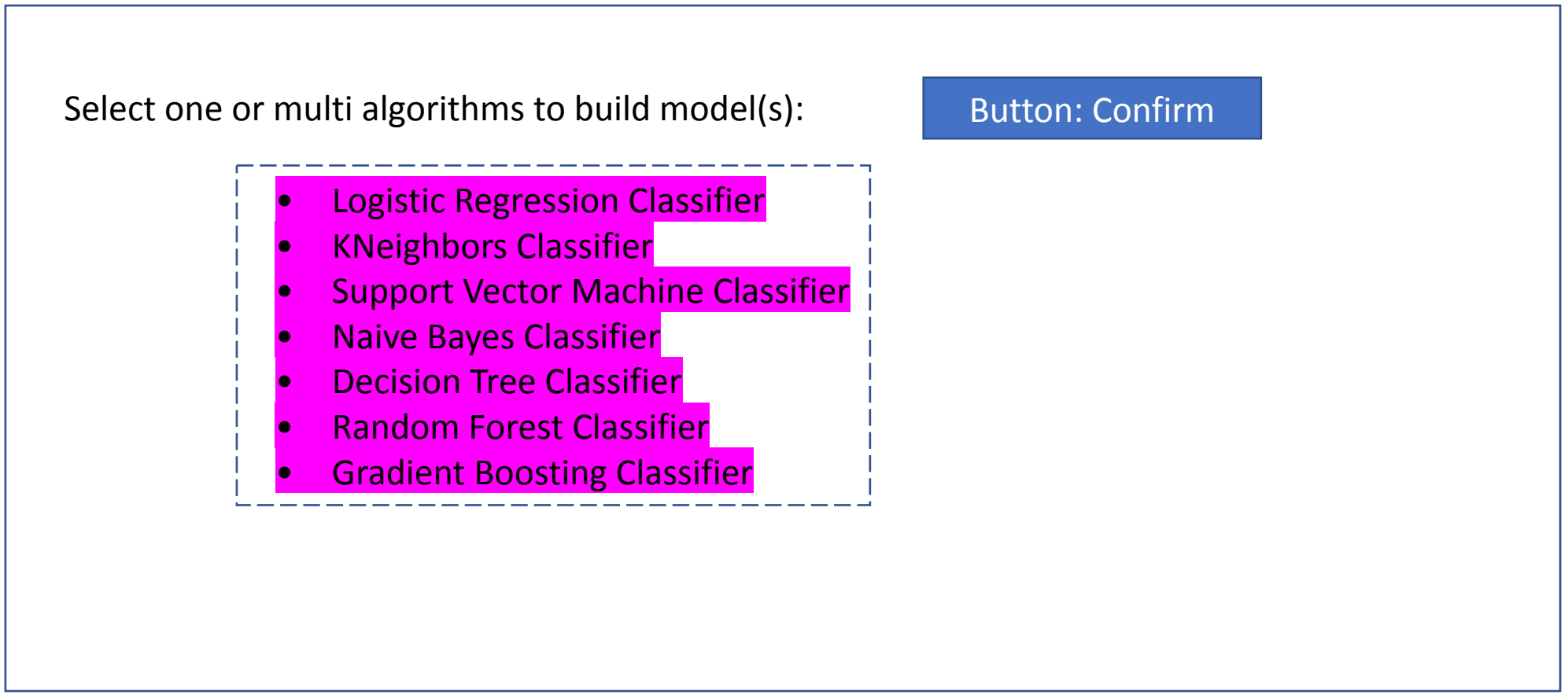
	age	gender	height_cm	weight_kg	body fat	diastolic	systolic	gripForce	sit and bend forward_cm	sit-ups counts	broad jump_cm	pred_healthy
1	25.0	M	165.0	55.8	15.7	77.0	126.0	36.4	16.3	53.0	229.0	1
2	42.0	F	164.5	63.7	32.2	72.0	135.0	22.7	0.8	18.0	146.0	0
3	42.0	M	169.2	65.4	19.3	63.0	110.0	43.5	16.0	68.0	211.0	1
4	57.0	F	153.0	49.0	20.9	69.0	106.0	21.5	30.0	0.0	90.0	0
5	45.0	F	159.0	63.1	30.9	93.0	144.0	34.1	19.0	30.0	155.0	0
6	26.0	M	179.9	71.5	9.7	64.0	135.0	59.6	17.8	61.0	239.0	1
7	59.0	F	155.9	62.7	30.2	76.0	143.0	36.8	29.1	25.0	122.0	1
8	38.0	M	166.7	67.3	23.2	70.0	111.0	26.1	19.7	62.0	215.0	0
9	47.0	F	158.3	53.5	29.2	70.0	117.0	25.9	8.1	32.0	151.0	0
10	48.0	M	175.8	84.5	31.4	83.0	125.0	33.8	3.7	24.0	211.0	0
11	50.0	F	159.8	57.1	24.4	63.0	103.0	30.8	24.4	30.0	143.0	1
12	28.0	F	159.5	51.54	24.5	82.0	123.0	37.2	23.0	37.0	145.0	1
13	30.0	M	172.1	79.5	26.7	91.0	148.0	34.7	-2.0	40.0	192.0	0
14	21.0	M	172.5	66.4	12.5	82.0	130.0	51.2	6.3	46.0	178.0	0
15	31.0	M	177.5	79.5	23.0	90.0	148.0	51.2	18.4	62.0	208.0	1
16	32.0	M	178.0	84.5	21.2	68.0	130.0	52.3	16.2	62.0	232.0	1

*Data saved: s3://dev-demo-land-area/autoML/users/0001/Binary_Classification/Scoring/001/score/score_output_Random Forest Classifier.csv

Classification UI



Classification UI



Create new model(s)

Select or Create a Source: Drop box: Button: Confirm

Your source should look like:

Column1	column2	column3	column4	column5	column6	column7	……	columnN	column_class

Create new model(s)

Your source preview (1000 rows):

Button: Confirm

Column1	column2	column3	column4	column5	column6	column7	……	columnN	Predict Value

Button: Submit job

Click to create json file

*House data(show different rows): s3://dev-demo-land-area/autoML/test_prepare/body_performance_healthy.csv
*json file saved: s3://dev-demo-land-area/autoML/requests/history/binary_classification_0001_001_Training_20211206.json
* copy json file to: s3://dev-demo-land-area/autoML/requests/queue/

Create new model(s)

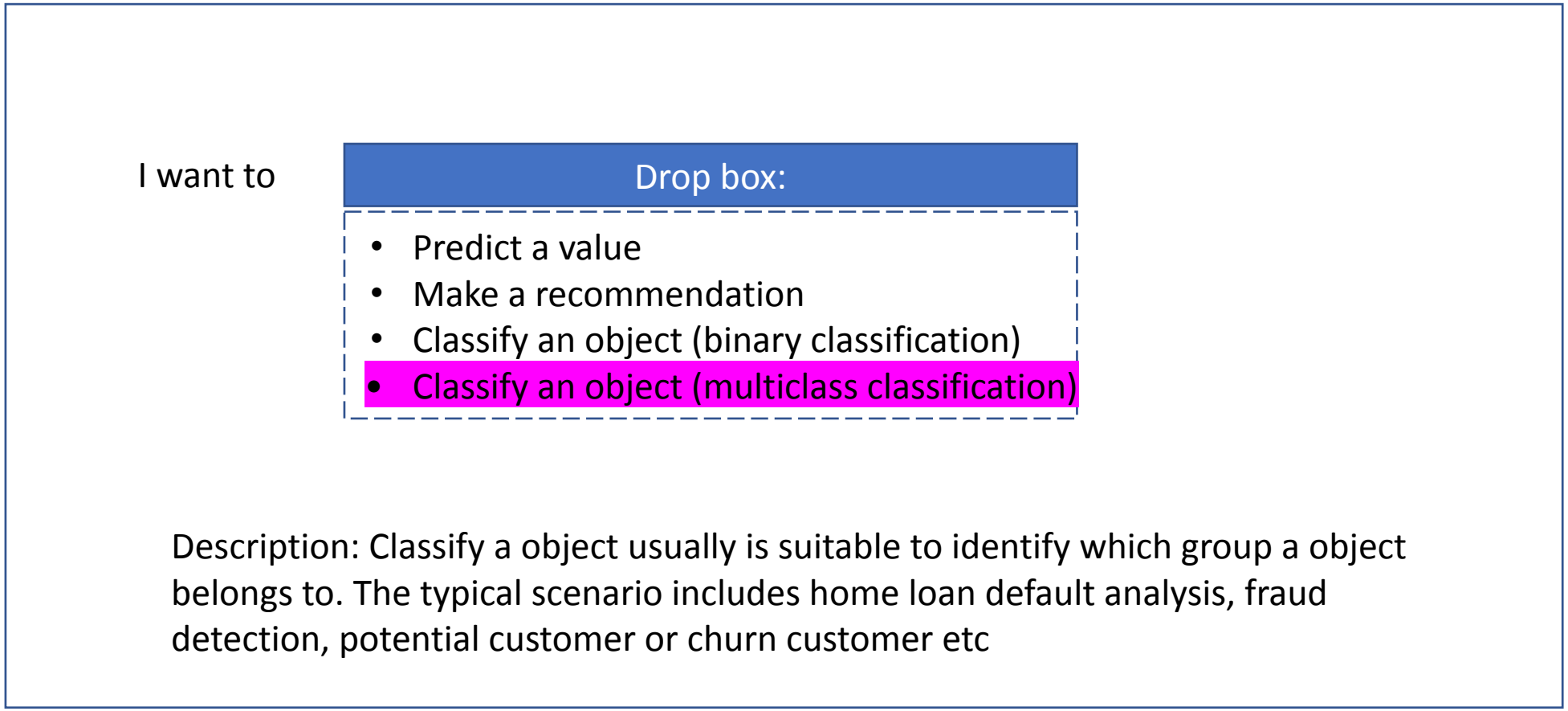
Button: Predict values

Created models overview

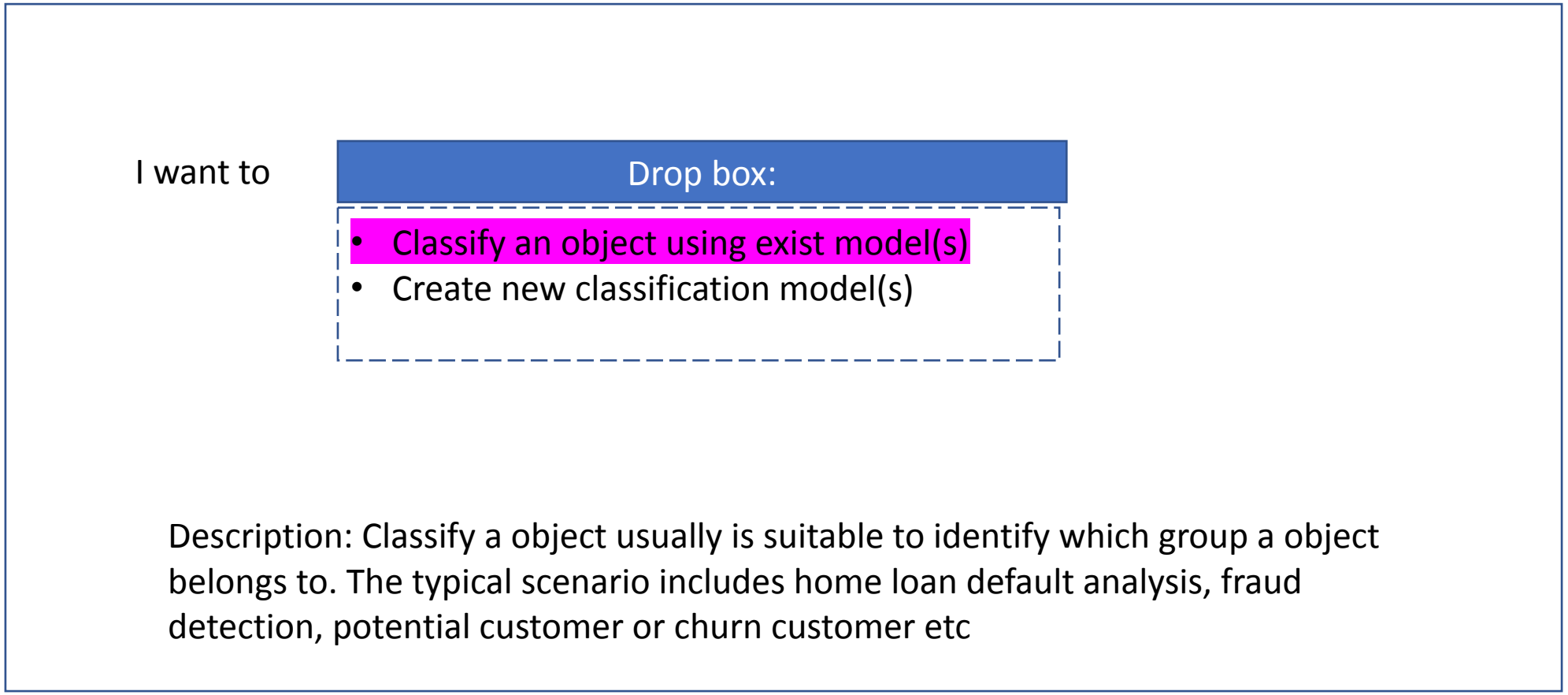
user ID	project ID	create time	model type	predict colume	model name	model_accuracy	recomended stars
0001	001	2021-12-06	Binary_Classification	healthy	Random Forest Classifier	accuracy_score:0.970 f1_score:0.971 precision_score:0.956 recall_score:0.986 roc_auc_score:0.969	*****
0001	001	2021-12-06	Binary_Classification	healthy	Gradient Boosting Classifier	accuracy_score:0.968 f1_score:0.969 precision_score:0.955 recall_score:0.984 roc_auc_score:0.967	*****
0001	001	2021-12-06	Binary_Classification	healthy	Logistic Regression Classifier	accuracy_score:0.955 f1_score:0.957 precision_score:0.936 recall_score:0.979 roc_auc_score:0.955	****
0001	001	2021-12-06	Binary_Classification	healthy	Support Vector Machine Classifier	accuracy_score:0.962 f1_score:0.963 precision_score:0.941 recall_score:0.987 roc_auc_score:0.961	****
0001	001	2021-12-06	Binary_Classification	healthy	KNeighbors Classifier	accuracy_score:0.941 f1_score:0.945 precision_score:0.901 recall_score:0.994 roc_auc_score:0.940	***
0001	001	2021-12-06	Binary_Classification	healthy	Naive Bayes Classifier	accuracy_score:0.912 f1_score:0.914 precision_score:0.904 recall_score:0.925 roc_auc_score:0.911	***
0001	001	2021-12-06	Binary_Classification	healthy	Decision Tree Classifier	accuracy_score:0.910 f1_score:0.9178 precision_score:0.863 recall_score:0.980 roc_auc_score:0.909	***

* table location: s3://dev-demo-land-area/autoML/test_prepare/binary_classification_exist_models.csv

Main UI


I want to
Drop box:
• Predict a value
• Make a recommendation
• Classify an object (binary classification)
• Classify an object (multiclass classification)
Description: Classify a object usually is suitable to identify which group a object belongs to. The typical scenario includes home loan default analysis, fraud detection, potential customer or churn customer etc

Classification UI



Classify an object using exist model(s) UI

Button: Confirm

Select one or more exist models

user ID	project ID	create time	model type	predict colume	model name	model_accuracy	recomended stars	please select
0001	001	2021-12-06	Multiclass_Classification	class	Logistic Regression Classifier	accuracy_score:0.592 f1_score:0.584 precision_score:0.581 recall_score:0.592 cohen_kappa_score:0.456 matthews_corrcoef:0.457	*****	☑
0001	001	2021-12-06	Multiclass_Classification	class	KNeighbors Classifier	accuracy_score:0.537 f1_score:0.538 precision_score:0.563 recall_score:0.537 cohen_kappa_score:0.383 matthews_corrcoef:0.389	****	☐
0001	001	2021-12-06	Multiclass_Classification	class	Support Vector Machine Classifier	accuracy_score:0.230 f1_score:0.185 precision_score:0.208 recall_score:0.230 cohen_kappa_score:-0.026 matthews_corrcoef:-0.031	*	☐
0001	001	2021-12-06	Multiclass_Classification	class	Naive Bayes Classifier	accuracy_score:0.530 f1_score:0.520 precision_score:0.517 recall_score:0.530 cohen_kappa_score:0.374 matthews_corrcoef:0.376	***	☐
0001	001	2021-12-06	Multiclass_Classification	class	Decision Tree Classifier	accuracy_score:0.531 f1_score:0.499 precision_score:0.535 recall_score:0.531 cohen_kappa_score:0.375 matthews_corrcoef:0.392	***	☐

* table location: s3://dev-demo-land-area/autoML/test_prepare/multiclass_classification_exist_models.csv

Classify an object using exist model(s) UI

TO use the selected model(s), your source should look like:

	age	gender	height_cm	weight_kg	body fat	diastolic	systolic	gripForce	sit and bend forward_cm	sit-ups counts	broad jump_cm	class
1	27.0	M	172.3	75.24	21.3	80.0	130.0	54.9	18.4	60.0	217.0	C
2	25.0	M	165.0	55.8	15.7	77.0	126.0	36.4	16.3	53.0	229.0	A
3	31.0	M	179.6	78.0	20.1	92.0	152.0	44.8	12.0	49.0	181.0	C
4	32.0	M	174.5	71.1	18.4	76.0	147.0	41.4	15.2	53.0	219.0	B
5	28.0	M	173.8	67.7	17.1	70.0	127.0	43.5	27.1	45.0	217.0	B
6	36.0	F	165.4	55.4	22.0	64.0	119.0	23.8	21.0	27.0	153.0	B
7	42.0	F	164.5	63.7	32.2	72.0	135.0	22.7	0.8	18.0	146.0	D
8	33.0	M	174.9	77.2	36.9	84.0	137.0	45.9	12.3	42.0	234.0	B
9	54.0	M	166.8	67.5	27.6	85.0	165.0	40.4	18.6	34.0	148.0	C
10	28.0	M	185.0	84.6	14.4	81.0	156.0	57.9	12.1	55.0	213.0	B
11	42.0	M	169.2	65.4	19.3	63.0	110.0	43.5	16.0	68.0	211.0	A
12	57.0	F	153.0	49.0	20.9	69.0	106.0	21.5	30.0	0.0	90.0	D
13	27.0	F	156.0	53.9	35.5	69.0	116.0	23.1	13.1	28.0	144.0	C
14	22.0	M	175.7	67.9	11.3	71.0	103.0	52.5	19.2	55.0	232.0	C
15	24.0	M	181.0	84.4	20.4	80.0	120.0	48.9	7.2	54.0	213.0	C

Select or Create a Source: Drop box: Button: Confirm

*recommendation data: s3://dev-demo-land-area/autoML/test_prepare/body_performance_class.csv

Classify an object using exist model(s) UI

Your source preview (100 rows):

Button: Confirm

	age	gender	height_cm	weight_kg	body fat	diastolic	systolic	gripForce	sit and bend forward_cm	sit-ups counts	broad jump_cm	class
1	27.0	M	172.3	75.24	21.3	80.0	130.0	54.9	18.4	60.0	217.0	C
2	25.0	M	165.0	55.8	15.7	77.0	126.0	36.4	16.3	53.0	229.0	A
3	31.0	M	179.6	78.0	20.1	92.0	152.0	44.8	12.0	49.0	181.0	C
4	32.0	M	174.5	71.1	18.4	76.0	147.0	41.4	15.2	53.0	219.0	B
5	28.0	M	173.8	67.7	17.1	70.0	127.0	43.5	27.1	45.0	217.0	B
6	36.0	F	165.4	55.4	22.0	64.0	119.0	23.8	21.0	27.0	153.0	B
7	42.0	F	164.5	63.7	32.2	72.0	135.0	22.7	0.8	18.0	146.0	D
8	33.0	M	174.9	77.2	36.9	84.0	137.0	45.9	12.3	42.0	234.0	B
9	54.0	M	166.8	67.5	27.6	85.0	165.0	40.4	18.6	34.0	148.0	C
10	28.0	M	185.0	84.6	14.4	81.0	156.0	57.9	12.1	55.0	213.0	B
11	42.0	M	169.2	65.4	19.3	63.0	110.0	43.5	16.0	68.0	211.0	A
12	57.0	F	153.0	49.0	20.9	69.0	106.0	21.5	30.0	0.0	90.0	D

Button: Submit job

Click to create json file

*recommendation data(show different rows): s3://dev-demo-land-area/autoML/test_prepare/body_performance_class.csv
*json file saved: s3://dev-demo-land-area/autoML/requests/history/multiclass_classification_0001_001_Scoring_20211206.json
* copy json file to: s3://dev-demo-land-area/autoML/requests/queue/

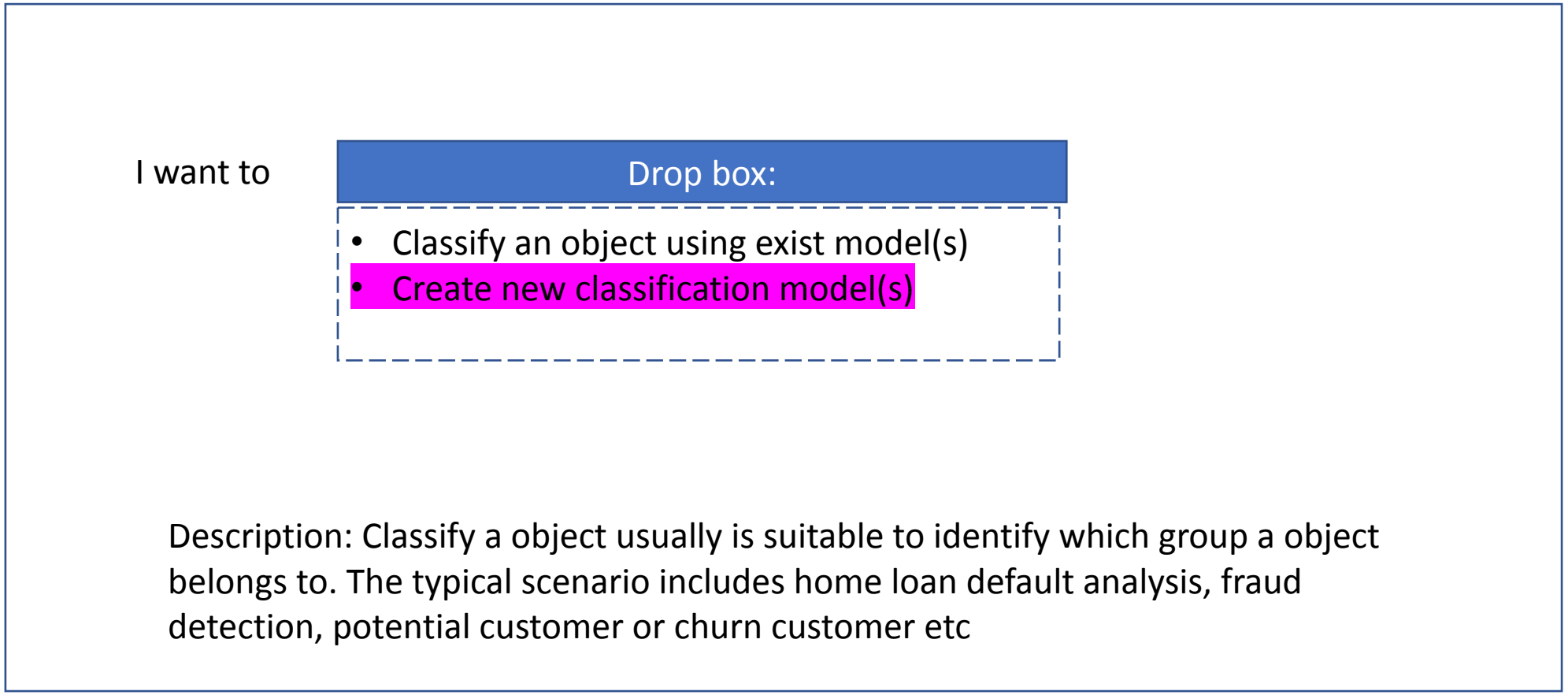
Classify an object using exist model(s) UI

Your prediction preview (1000 rows):

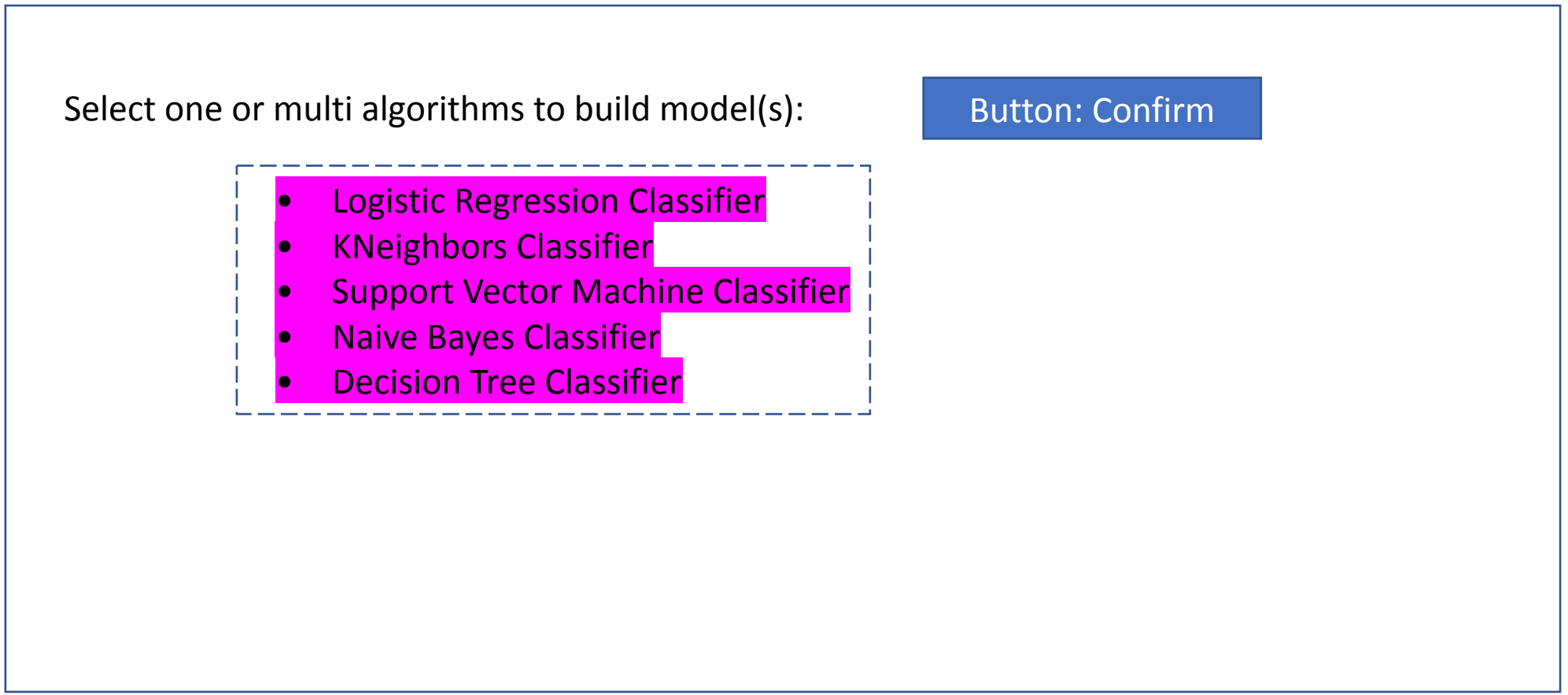
	age	gender	height_cm	weight_kg	body fat	diastolic	systolic	gripForce	sit and bend forward_cm	sit-ups counts	broad jump_cm	pred_class
1	27.0	M	172.3	75.24	21.3	80.0	130.0	54.9	18.4	60.0	217.0	A
2	25.0	M	165.0	55.8	15.7	77.0	126.0	36.4	16.3	53.0	229.0	B
3	31.0	M	179.6	78.0	20.1	92.0	152.0	44.8	12.0	49.0	181.0	C
4	32.0	M	174.5	71.1	18.4	76.0	147.0	41.4	15.2	53.0	219.0	B
5	28.0	M	173.8	67.7	17.1	70.0	127.0	43.5	27.1	45.0	217.0	B
6	36.0	F	165.4	55.4	22.0	64.0	119.0	23.8	21.0	27.0	153.0	C
7	42.0	F	164.5	63.7	32.2	72.0	135.0	22.7	0.8	18.0	146.0	D
8	33.0	M	174.9	77.2	36.9	84.0	137.0	45.9	12.3	42.0	234.0	D
9	54.0	M	166.8	67.5	27.6	85.0	165.0	40.4	18.6	34.0	148.0	C
10	28.0	M	185.0	84.6	14.4	81.0	156.0	57.9	12.1	55.0	213.0	B
11	42.0	M	169.2	65.4	19.3	63.0	110.0	43.5	16.0	68.0	211.0	A
12	57.0	F	153.0	49.0	20.9	69.0	106.0	21.5	30.0	0.0	90.0	C
13	27.0	F	156.0	53.9	35.5	69.0	116.0	23.1	13.1	28.0	144.0	D
14	22.0	M	175.7	67.9	11.3	71.0	103.0	52.5	19.2	55.0	232.0	A

*Data saved: s3://dev-demo-land-area/autoML/users/0001/Multiclass_Classification/Scoring/001/score/score_output_Logistic Regression Classifier.csv

Classification UI



Classification UI



Create new model(s)

Select or Create a Source: Drop box: Button: Confirm

Your source should look like:

Column1	column2	column3	column4	column5	column6	column7	……	columnN	column_class

Create new model(s)

Your source preview (1000 rows):

Button: Confirm

Column1	column2	column3	column4	column5	column6	column7	……	columnN	Predict Value

Button: Submit job

Click to create json file

*House data(show different rows): s3://dev-demo-land-area/autoML/test_prepare/body_performance_class.csv
*json file saved: s3://dev-demo-land-area/autoML/requests/history/multiclass_classification_0001_001_Training_20211206.json
* copy json file to: s3://dev-demo-land-area/autoML/requests/queue/

Create new model(s)

Button: Predict values

Created models overview

user ID	project ID	create time	model type	predict colume	model name	model_accuracy	recomended stars
0001	001	2021-12-06	Multiclass_Classification	class	Logistic Regression Classifier	accuracy_score:0.592 f1_score:0.584 precision_score:0.581 recall_score:0.592 cohen_kappa_score:0.456 matthews_corrcoef:0.457	*****
0001	001	2021-12-06	Multiclass_Classification	class	KNeighbors Classifier	accuracy_score:0.537 f1_score:0.538 precision_score:0.563 recall_score:0.537 cohen_kappa_score:0.383 matthews_corrcoef:0.389	****
0001	001	2021-12-06	Multiclass_Classification	class	Support Vector Machine Classifier	accuracy_score:0.230 f1_score:0.185 precision_score:0.208 recall_score:0.230 cohen_kappa_score:-0.026 matthews_corrcoef:-0.031	*
0001	001	2021-12-06	Multiclass_Classification	class	Naive Bayes Classifier	accuracy_score:0.530 f1_score:0.520 precision_score:0.517 recall_score:0.530 cohen_kappa_score:0.374 matthews_corrcoef:0.376	***
0001	001	2021-12-06	Multiclass_Classification	class	Decision Tree Classifier	accuracy_score:0.531 f1_score:0.499 precision_score:0.535 recall_score:0.531 cohen_kappa_score:0.375 matthews_corrcoef:0.392	***

* table location: s3://dev-demo-land-area/autoML/test_prepare/multiclass_classification_exist_models.csv

Microsoft Azure
Machine Learning Algorithm Cheat Sheet

This cheat sheet helps you choose the best machine learning algorithm for your predictive analytics solution. Your decision is driven by both the nature of your data and the goal you want to achieve with your data.

What do you want to do?

Extract information from text

Text Analytics

Derives high-quality information from text

Answers questions like: What info is in this text?

- Extract N-Gram Features from Text ← Creates a dictionary of n-grams from a column of free text
- Feature Hashing ← Converts text data to integer encoded features using the Vowpal Wabbit library
- Preprocess Text ← Performs cleaning operations on text, like removal of stop-words, case normalization
- Word2Vector ← Converts words to values for use in NLP tasks, like recommender, named entity recognition, machine translation

Predict between several categories

Multiclass Classification

Answers complex questions with multiple possible answers

Answers questions like: Is this A or B or C or D?

- Multiclass Logistic Regression ← Fast training times, linear model
- Multiclass Neural Network ← Accuracy, long training times
- Multiclass Decision Forest ← Accuracy, fast training times
- One-vs-All Multiclass ← Depends on the two-class classifier
- Multiclass Boosted Decision Tree ← Non-parametric, fast training times and scalable

Predict between two categories

Two-Class Classification

Answers simple two-choice questions, like yes or no, true or false

Answers questions like: Is this A or B?

- Two-Class Support Vector Machine ← Under 100 features, linear model
- Two-Class Averaged Perceptron ← Fast training, linear model
- Two-Class Decision Forest ← Accurate, fast training
- Two-Class Logistic Regression ← Fast training, linear model
- Two-Class Boosted Decision Tree ← Accurate, fast training, large memory footprint
- Two-Class Neural Network ← Accurate, long training times

Predict values

Regression

Makes forecasts by estimating the relationship between values

Answers questions like: How much or how many?

- Fast Forest Quantile Regression ← Predicts a distribution
- Poisson Regression ← Predicts event counts
- Linear Regression ← Fast training, linear model
- Bayesian Linear Regression ← Linear model, small data sets
- Decision Forest Regression ← Accurate, fast training times
- Neural Network Regression ← Accurate, long training times
- Boosted Decision Tree Regression ← Accurate, fast training times, large memory footprint

Generate recommendations

Recommenders

Predicts what someone will be interested in

Answers the question: What will they be interested in?

- SVD Recommender ← Collaborative filtering, better performance with lower cost by reducing dimensionality

Discover structure

Clustering

Separates similar data points into intuitive groups

Answers questions like: How is this organized?

- K-Means ← Unsupervised learning

Find unusual occurrences

Anomaly Detection

Identifies and predicts rare or unusual data points

Answers the question: Is this weird?

- One Class SVM ← Under 100 features, aggressive boundary
- PCA-Based Anomaly Detection ← Fast training times

Classify images

Image Classification

Classifies images with popular networks

Answers questions like: What does this image represent?

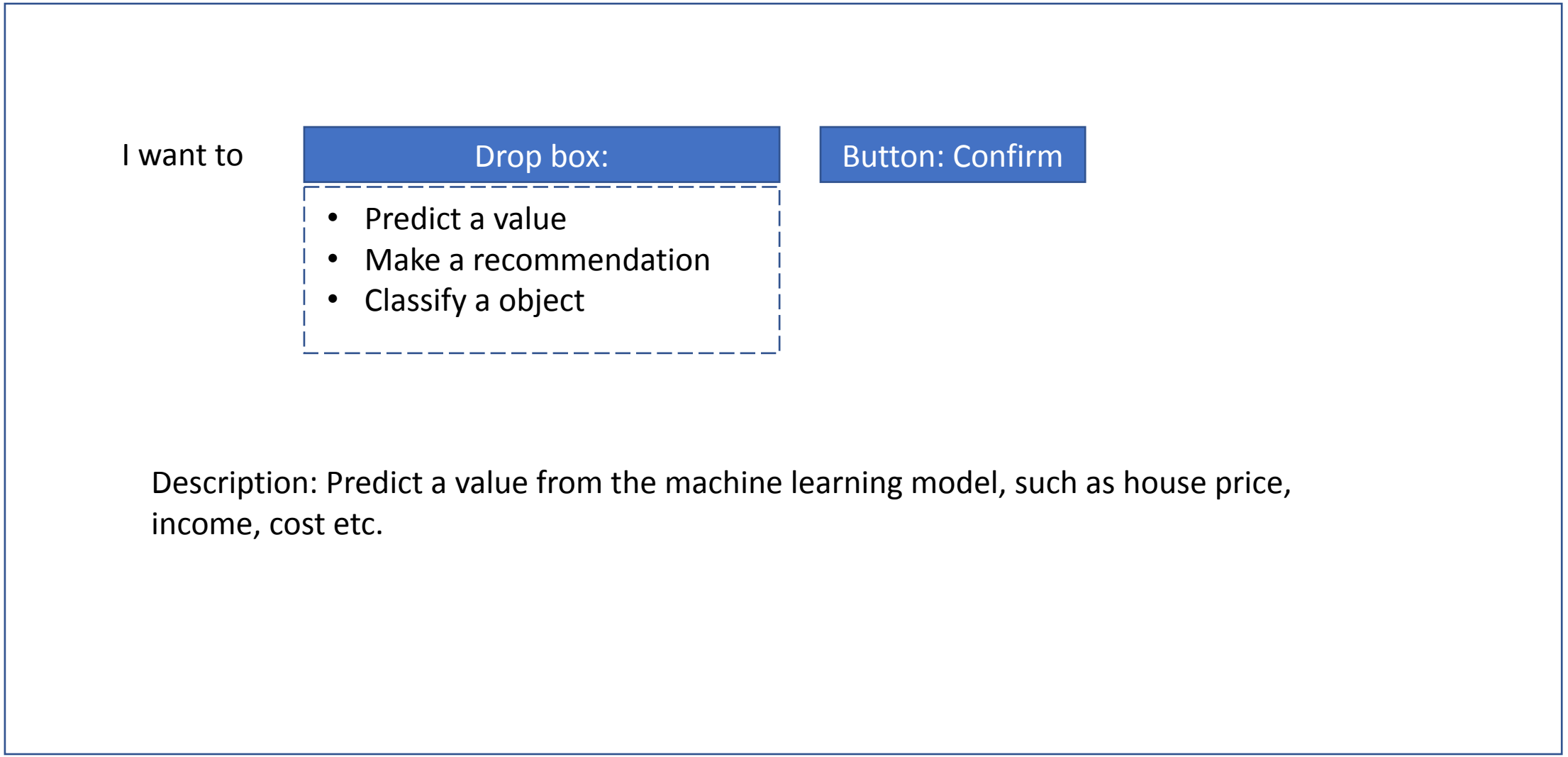
- DenseNet ← High accuracy, better efficiency

© 2019 Microsoft Corporation. All rights reserved.

Share this poster: aka.ms/mlcheatsheet

Microsoft

Main UI


I want to
Drop box:
Button: Confirm
• Predict a value
• Make a recommendation
• Classify a object
Description: Predict a value from the machine learning model, such as house price, income, cost etc.

Predict a value UI

Select or Create a Source: Drop box: Button: Confirm

Your source should look like:

Column1	column2	column3	column4	column5	column6	column7	……	columnN	Predict Value

Or create a dataset via our data integration tool Button: Confirm

Or use notebook to create your own experiment Button: Confirm

Predict a value UI

Your source preview (1000 rows):

Button: Confirm

Column1	column2	column3	column4	column5	column6	column7	……	columnN	Predict Value

Predict a value UI

We have done:

Algorithm 1,2,3

Algorithm 1 feature, validation scores, curves…

Algorithm 2 feature, validation scores, curves…

We would recommend:

Algorithm X because the following reason….

Click here to download your trained model or deploy

Button: Confirm

Predict a value UI - test your model

Select or Create a Source:
Or fill the form

Drop box:

Button: Confirm

Column1	column2	column3	column4	column5	column6	column7	……	columnN	Predict Value
input	input	input	input	input	input	input		input	
input	input	input	input	input	input	input		input	
input	input	input	input	input	input	input		input	

Test result

Column1	column2	column3	column4	column5	column6	column7	……	columnN	Predict Value
input	input	input	input	input	input	input		input	X
input	input	input	input	input	input	input		input	X
input	input	input	input	input	input	input		input	X

Predict a value UI

We have done:

Algorithm 1,2,3

Algorithm 1 feature, validation scores, curves…

Algorithm 2 feature, validation scores, curves…

We would recommend:

Algorithm X because the following reason….

Click here to download your trained model or deploy

Button: Confirm