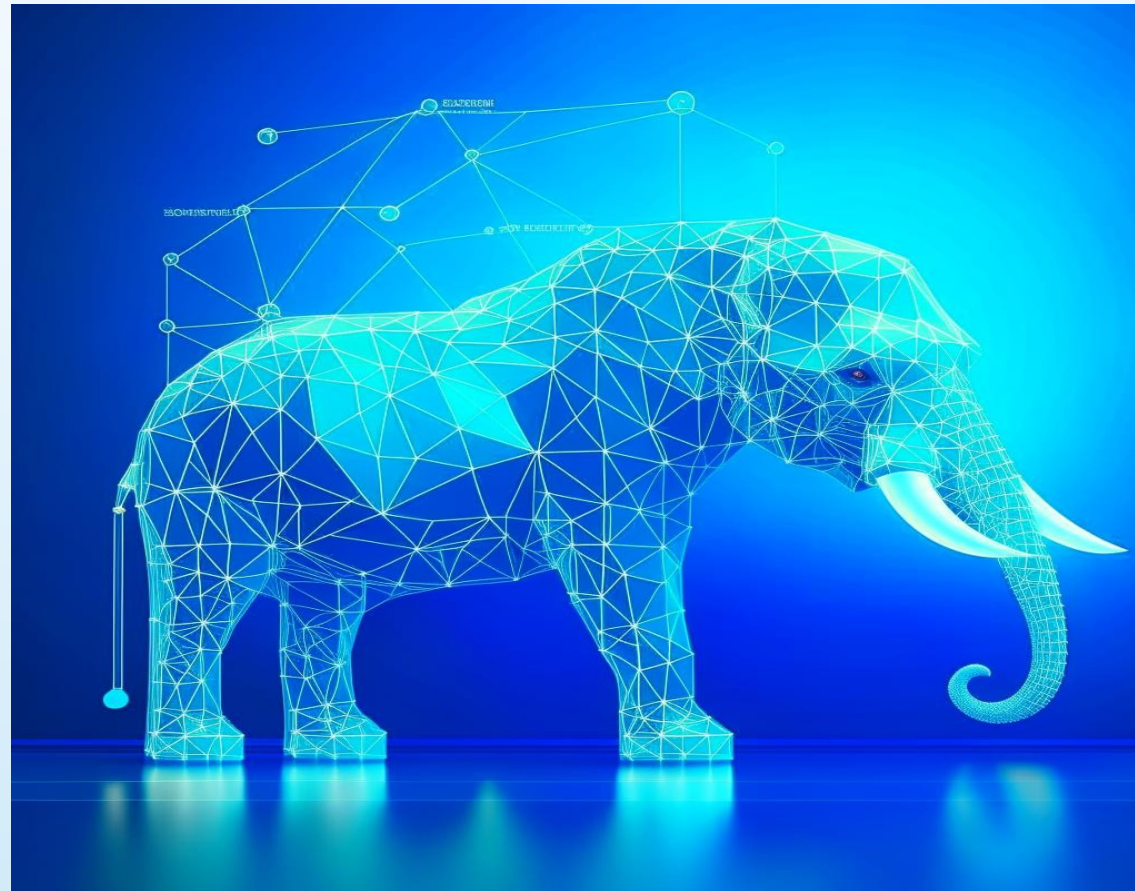


# ML in PostgreSQL



# Using models:

- Binary Classification
- Multi Classification
- Regression

Using framework:

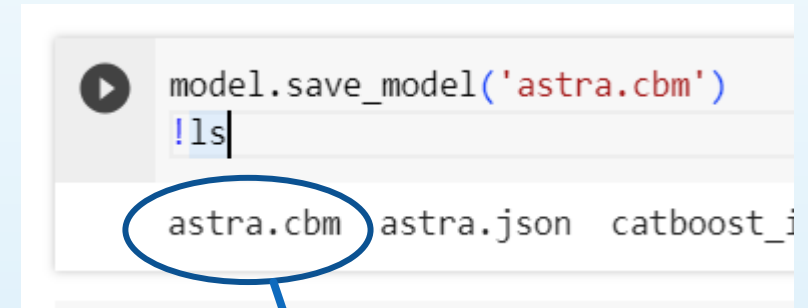


CatBoost: <https://catboost.ai/>

only prediction

# ML process:

1. Training model
2. Save model to database server



3. prediction

```
adult=# SELECT ml_predict('astra3.cbm', 'astra3');  
WARNING: field run_id not used  
WARNING: field field_id not used  
WARNING: field spec_obj_id not used  
WARNING: field predict not used  
      ml_predict  
-----  
public.astra3_predict  
(1 row)
```

# Installation

- ❑ git clone [https://github.com/akalend/pg\\_ml.git](https://github.com/akalend/pg_ml.git)
- ❑ export PG\_HOME=/usr/local/pgsql //where is main postgres folder
- ❑ wget https://github.com/catboost/catboost/releases/download/v1.2.2/libcatboostmodel.so
- ❑ mv libcatboostmodel.so \$PG\_HOME/lib
- ❑ cd pg\_ml
- ❑ export PG\_CONFIG=\$PG\_HOME/bin/pg\_config
- ❑ export LD\_LIBRARY\_PATH=\$PG\_HOME/lib
- ❑ USE\_PGXS=1 make
- ❑ sudo su
- ❑ export PATH=\$PATH:\$PG\_HOME/bin
- ❑ USE\_PGXS=1 make install
- ❑ chown postgres model.cbm
- ❑ [optional] cp model.cbm \$PG\_HOME/data



# Configuration

postgresql.conf:

# path to model folder

ml.model\_path = /usr/local/model

# DataFrame columns convert to PostgreSQL fields



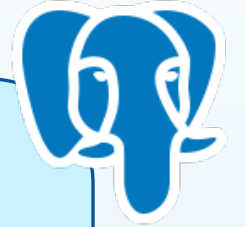
ID, Id, id

FieldID, Field\_ID

Import-port

Port, PORT

<=>



id

field\_id

import\_port

port

# DataFrame columns and PostgreSQL fields

```
[14] df = pd.read_csv('star_classification.csv')
```

```
[16] for it in df.columns:  
      print(it)
```

```
obj_ID  
alpha  
delta  
u  
g  
r  
i  
z  
run_ID  
rerun_ID  
cam_col  
field_ID  
spec_obj_ID  
class  
redshift  
plate  
MJD  
fiber_ID
```



```
adult=# \d astra3
```

Column	Type	Collation
alpha	double precision	
delta	double precision	
u	double precision	
g	double precision	
r	double precision	
i	double precision	
z	double precision	
run_id	bigint	
cam_col	bigint	
field_id	bigint	
spec_obj_id	double precision	
redshift	double precision	
plate	bigint	
mjd	bigint	
fiber_id	bigint	

```
adult=#
```





# DataFrame columns and PostgreSQL fields



```
df.head()
```

	obj_ID	alpha	delta	u	g	r	i	z	run_ID	rerun_ID	cam_col	field_ID	spec_obj_ID
0	1.237661e+18	135.689107	32.494632	23.87882	22.27530	20.39501	19.16573	18.79371	3606	301	2	79	6.543777e+18
1	1.237665e+18	144.826101	31.274185	24.77759	22.83188	22.58444	21.16812	21.61427	4518	301	5	119	1.176014e+19
2	1.237661e+18	142.188790	35.582444	25.26307	22.66389	20.60976	19.34857	18.94827	3606	301	2	120	5.152200e+18
3	1.237663e+18	338.741038	-0.402828	22.13682	23.77656	21.61162	20.50454	19.25010	4192	301	3	214	1.030107e+19
4	1.237680e+18	345.282593	21.183866	19.43718	17.58028	16.49747	15.97711	15.54461	8102	301	3	137	6.891865e+18

```
adult=# select * from astra3 limit 3;
```

alpha	delta	u	g	r	i	z	run_id	cam_col	field_id	spec_obj_id
16.9568897845004	3.64613008870454	23.33542	21.95143	20.48149	19.603	19.13094	7712	6	442	4.855016555329904e+18
240.063240247767	6.13413059813973	17.86033	16.79228	16.43001	16.30923	16.25873	3894	1	243	2.4489280322708705e+18
30.887222067625	1.18870964120799	18.18911	16.89469	16.42161	16.24627	16.18549	7717	1	536	8.255357438959835e+18

```
(3 rows)
```



# Information about model

```
adult=# SELECT ml_info ('astra3.cbm');
              ml_info
-----
dimension:3 numeric features:12 categorical features:0 modelType "MultiClass"+
fieldName:alpha,delta,u,g,r,i,z,can_col,redshift,plate,MJD,fiber_ID
(1 row)
adult=#
```

- Dimension result (How many classes)
- Feature count (categorical and float)
- Type of model
- Fields name

# Information about model

```
adult=# SELECT ml_info ('astra3.cbm');
              ml_info
-----
dimension:3 numeric features:12 categorical features:0 modelType "MultiClass"+
fieldName:alpha,delta,u,g,r,i,z,can_col,redshift,plate,MJD,fiber_ID
(1 row)

adult=#
```

```
adult=# SELECT ml_info ('titanic.cbm');
              ml_info
-----
dimension:1 numeric features:2 categorical features:9 modelType "Accuracy" +
fieldName:PassengerId,Pclass,Name,Sex,Age,SibSp,Parch,Ticket,Fare,Cabin,Embarked
(1 row)
```

# Model information

```
akalend@notebook-sasha: ~/stars
adult=# select ml_info('boston.cbm');
          ml_info
-----
dimension:1 numeric features:13 categorical features:0 modelType "RMSE" +
fieldName:crim,zn,indus,chas,nox,rm,age,dis,rad,tax,ptratio,black,lstat
(1 row)
adult=#
```

## More information

```
model.save_model('astra.json', format='json')  
!ls  
astra.cbm astra.json catboost_info sample_data
```

```
adult=# SELECT * from ml_json_info('astra.json');  
ml_json_info  
-----  
float feature:alpha,delta,u,g,r,i,z,cam_col,redshift,plate,MJD,fiber_ID,+  
categorical feature:  
(1 row)
```

Categorical Feature list  
Float Feature count list



# Prediction of model (recordset)

**Path to model file**

**Table name**

**Categorical field list**

```
adult=# SELECT * from ml_cat_predict ('titanic.cbm',  
'titanic', '{name,passenger_id,pclass,sex,sibsp,parch,ticket,cabin,embarked }');
```

row_num	predict	class
0	-1.7937342449233795	0
1	-0.7958399022225136	0
2	-2.392873216013247	0
3	-1.942976624899004	0
4	-0.41747860726736713	0
5	-2.0608914711097546	0
6	0.5914467057444344	1
7	-1.0786526230973736	0
8	0.6757411102494171	1
9	-3.250956928980716	0
10	-2.274725588104562	0
11	-1.3228896775643357	0
12	2.70931909246417	1
13	-2.4233542239140187	0

# Prediction of model (result table)

**Path to model file**

**Table name**

```
adult=# SELECT * from ml_predict_table('astra3.cbm','astra3');
ml_predict_table
-----
public.astra3_predict (1 row)
```

**Create  
the new  
table**

# Prediction of model (table)

**Path to model file**

**Table name**

**List of categorical fields**

```
adult=# SELECT ml_predict ('adult.cbm', 'adult2',  
adult(# '{workclass,education,marital_status, occupation,relationship,race,sex,native_country}');  
ml_predict  
-----  
public.adult2_predict  
(1 row)
```

**Create  
the new  
table**



# Prediction results

SELECT \* FROM {table}\_predict;

adult=# SELECT * from astra3_predict;																	
row	alpha	delta	u	g	r	i	z	run_id	cam_col	field_id	spec_obj_id	redshift	plate	mjd	fiber_id	predict	class
1	16.9568897845004	3.64613008870454	23.33542	21.95143	20.48149	19.603	19.13094	7712	6	442	4.855016555329904e+18	0.5062369	4312	55511	495	0.98686	GALAXY
2	240.063240247767	6.13413059813973	17.86033	16.79228	16.43001	16.30923	16.25873	3894	1	243	2.4489280322708705e+18	0.0003448142	2175	54612	348	0.990419	STAR
3	30.887222067625	1.18870964120799	18.18911	16.89469	16.42161	16.24627	16.18549	7717	1	536	8.255357438959835e+18	4.085216e-06	7332	56683	943	0.997588	STAR
4	247.594400505002	10.8877797153666	24.99961	21.71203	21.47148	21.30532	21.29109	5323	1	134	4.577998722756271e+18	-0.0002914838	4066	55444	326	0.997667	STAR
5	18.8964507920807	-5.26133022886992	23.76648	21.79737	20.69543	20.23403	19.97464	7881	3	148	8.91047176642785e+18	-0.0001361561	7914	57331	363	0.996044	STAR
6	182.713733094955	51.3758050594777	22.44608	21.68444	20.24292	19.41423	19.08227	2830	1	411	7.516725588574623e+18	0.5026683	6676	56389	792	0.984373	GALAXY
7	150.089423193165	39.4670880748061	18.96441	17.82906	17.31429	16.99891	16.85583	3560	4	278	1.5267956411104236e+18	0.06366445	1356	53033	274	0.996164	GALAXY
8	189.510984338851	58.7411197772507	21.37376	20.80187	20.84925	21.13449	20.34689	2243	1	353	7.696817897528907e+18	0.7936153	6836	56443	604	0.957787	QSO
9	37.7138728560977	-0.525138228146508	20.77988	19.54618	19.16687	18.89438	18.64286	2700	2	117	1.7553283123029217e+18	0.1060118	1559	53271	183	0.993892	GALAXY
10	201.074980072746	28.7699058867715	25.05349	22.23362	20.8122	19.69488	19.28336	4649	3	120	7.306035245308205e+18	0.567082	6489	56329	257	0.993856	GALAXY
11	151.83091832672	19.8108624669417	24.04443	22.48608	20.59701	19.50985	19.00457	5183	5	142	6.622787444780849e+18	0.5475619	5882	56029	888	0.998885	GALAXY

SELECT \* FROM  
ml\_predict(...);

adult=# SELECT * from ml_predict('astra3.cbm' 'astra3');		
id	predict	class
0	0.9868595777513302	GALAXY
1	0.9904188657285139	STAR
2	0.9975875623929414	STAR
3	0.9976669380943318	STAR
4	0.9960439244920889	STAR
5	0.9843734017027631	GALAXY
6	0.9961635567874662	GALAXY
7	0.9577871819302538	QSO
8	0.9938922568658763	GALAXY
9	0.9938564131331261	GALAXY

# Binary classification

postgres@notebook-sasha: /usr/local/pgsql

```
adult=# select * from titanic_predict;
```

row	id	passenger_id	pclass	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked	res	predict	clas
1	0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	-999	Q	f	0.142616	0
2	1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47	1	0	363272	7	-999	S	f	0.310916	0
3	2	894	2	Myles, Mr. Thomas Francis	male	62	0	0	240276	9.6875	-999	Q	f	0.083718	0
4	3	895	3	Wirz, Mr. Albert	male	27	0	0	315154	8.6625	-999	S	f	0.125321	0
5	4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22	1	1	3101298	12.2875	-999	S	f	0.39712	0
6	5	897	3	Svensson, Mr. Johan Cervin	male	14	0	0	7538	9.225	-999	S	f	0.112956	0
7	6	898	3	Connolly, Miss. Kate	female	30	0	0	330972	7.6292	-999	Q	t	0.643697	1
8	7	899	2	Caldwell, Mr. Albert Francis	male	26	1	1	248738	29	-999	S	f	0.253761	0
9	8	900	3	Abraham, Mrs. Joseph (Sophie Halaut Easu)	female	18	0	0	2657	7.2292	-999	C	t	0.662787	1
10	9	901	3	Davies, Mr. John Samuel	male	21	2	0	A/4 48871	24.15	-999	S	f	0.037293	0
11	10	902	3	Iliffe, Mr. Yllo	male	-999	0	0	349220	7.8958	-999	S	f	0.093238	0
12	11	903	1	Jones, Mr. Charles Cresson	male	46	0	0	694	26	-999	S	f	0.210338	0

```
adult=# SELECT * from ml_cat_predict ('titanic.cbm', 'titanic', '{name,passenger_id,pclass,sex,sibsp,parch,ticket,cabin,embarked}');
```

row_num	predict	class
0	-1.7937342449233795	0
1	-0.7958399022225136	0
2	-2.392873216013247	0
3	-1.942976624899004	0
4	-0.41747860726736713	0
5	-2.0608914711097546	0
6	0.5914467057444344	1
7	-1.9769579739773739	0

# Binary classification

```
adult=# SELECT * FROM ml_cat_predict ('adult.cbm', 'adult2','{workclass,  
education,marital_status, occupation,relationship,race,sex,native_country}  
' );
```

row_num	predict	class
0	-5.926338548423682	<=50K
1	-1.225876230403332	<=50K
2	-0.7485117670534811	<=50K
3	3.6351647093731705	>50K
4	-4.644606242153101	<=50K
5	-5.342578732065899	<=50K
6	-3.9224526779262296	<=50K

postgres@notebook-sasha: /usr/local/pgsql															
row	age	workclass	fnlwgt	education	education_num	marital_status	occupation	relationship	race	sex	capital_gain	capital_loss	hours_per_week	native_country	predict   class
1	25	Private	226802	11th	7	Never-married	Machine-op-inspct	Own-child	Black	Male	0	0	40	United-States	0.002661   <=50K
2	38	Private	89814	HS-grad	9	Married-civ-spouse	Farming-fishing	Husband	White	Male	0	0	50	United-States	0.226904   <=50K
3	28	Local-gov	336951	Assoc-acdm	12	Married-civ-spouse	Protective-serv	Husband	White	Male	0	0	40	United-States	0.321146   <=50K
4	44	Private	160323	Some-college	10	Married-civ-spouse	Machine-op-inspct	Husband	Black	Male	7688	0	40	United-States	0.974298   >50K
5	18	nan	103497	Some-college	10	Never-married	nan	Own-child	White	Female	0	0	30	United-States	0.009522   <=50K
6	34	Private	198693	10th	6	Never-married	Other-service	Not-in-family	White	Male	0	0	30	United-States	0.004761   <=50K
7	29	nan	227026	HS-grad	9	Never-married	nan	Unmarried	Black	Male	0	0	40	United-States	0.019408   <=50K
8	63	Self-emp-not-inc	104626	Prof-school	15	Married-civ-spouse	Prof-specialty	Husband	White	Male	3103	0	32	United-States	0.560734   >50K
9	24	Private	369667	Some-college	10	Never-married	Other-service	Unmarried	White	Female	0	0	40	United-States	0.003572   <=50K
10	55	Private	104996	7th-8th	4	Married-civ-spouse	Craft-repair	Husband	White	Male	0	0	10	United-States	0.092036   <=50K
11	65	Private	104454	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	Male	6418	0	40	United-States	0.759614   >50K
12	36	Federal-gov	212465	Bachelors	13	Married-civ-spouse	Adm-clerical	Husband	White	Male	0	0	40	United-States	0.64466   >50K
13	26	Private	82091	HS-grad	9	Never-married	Adm-clerical	Not-in-family	White	Female	0	0	39	United-States	0.007208   <=50K
14	58	nan	299831	HS-grad	9	Married-civ-spouse	nan	Husband	White	Male	0	0	35	United-States	0.432042   <=50K



# Regression

akalend@notebook-sasha: ~/stars

row	index	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	black	lstat	medv	predict
1	0	0.00632	18	2.31	0	0.538	6.575	65.2	4.09	1	296	15.3	396.9	4.98	24	24.99982
2	1	0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6	20.664359
3	2	0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7	33.677379
4	3	0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4	34.289002
5	4	0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.9	5.33	36.2	34.615708
6	5	0.02985	0	2.18	0	0.458	6.43	58.7	6.0622	3	222	18.7	394.12	5.21	28.7	27.968317
7	6	0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9	21.682186
8	7	0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311	15.2	396.9	19.15	27.1	22.853984
9	8	0.21124	12.5	7.87	0	0.524	5.631	100	6.0821	5	311	15.2	386.63	29.93	16.5	17.011092
10	9	0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.1	18.9	18.24062
11	10	0.22489	12.5	7.87	0	0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	15	17.543837
12	11	0.11747	12.5	7.87	0	0.524	6.009	82.9	6.2267	5	311	15.2	396.9	13.27	18.9	19.974764
13	12	0.09378	12.5	7.87	0	0.524	5.889	39	5.4509	5	311	15.2	390.5	15.71	21.7	20.70866
14	13	0.62976	0	8.14	0	0.538	5.949	61.8	4.7075	4	307	21	396.9	8.26	20.4	20.202922
15	14	0.63796	0	8.14	0	0.538	6.096	84.5	4.4619	4	307	21	380.02	10.26	18.2	18.175456
16	15	0.62739	0	8.14	0	0.538	5.834	56.5	4.4986	4	307	21	395.62	8.47	19.9	19.698336
17	16	1.05393	0	8.14	0	0.538	5.935	29.3	4.4986	4	307	21	386.85	6.58	23.1	22.304514
18	17	0.7842	0	8.14	0	0.538	5.99	81.7	4.2579	4	307	21	386.75	14.67	17.5	17.160698
19	18	0.80271	0	8.14	0	0.538	5.456	36.6	3.7965	4	307	21	288.99	11.69	20.2	18.706903
20	19	0.7258	0	8.14	0	0.538	5.727	69.5	3.7965	4	307	21	390.95	11.28	18.2	18.769804
21	20	1.25179	0	8.14	0	0.538	5.57	98.1	3.7979	4	307	21	376.57	21.02	13.6	13.985032

--Далее--

```
adult=# SELECT * from ml_cat_predict ('boston.cbm', 'boston2');
row_num predict class
-----
0 24.99982028068538
1 20.664358727562394
2 33.67737911788664
3 34.28900239364565
4 34.61570849423551
5 27.968317495475695
6 21.68218578618033
```

# Multi classification

```
adult=# SELECT * from astra3_predict;
```

row	alpha	delta	u	g	r	i	z	run_id	cam_col	field_id	spec_obj_id	redshift	plate	mjd	fiber_id	predict	class
1	16.9568897845004	3.64613008870454	23.33542	21.95143	20.48149	19.603	19.13094	7712	6	442	4.855016555329904e+18	0.5062369	4312	55511	495	0.98686	GALAXY
2	240.063240247767	6.13413059813973	17.86033	16.79228	16.43001	16.30923	16.25873	3894	1	243	2.4489280322708705e+18	0.0003448142	2175	54612	348	0.990419	STAR
3	30.887222067625	1.18870964120799	18.18911	16.89469	16.42161	16.24627	16.18549	7717	1	536	8.255357438959835e+18	4.085216e-06	7332	56683	943	0.997588	STAR
4	247.594400505002	10.8877797153666	24.99961	21.71203	21.47148	21.30532	21.29109	5323	1	134	4.577998722756271e+18	-0.0002914838	4066	55444	326	0.997667	STAR
5	18.8964507920807	-5.26133022886992	23.76648	21.79737	20.69543	20.23403	19.97464	7881	3	148	8.91047176642785e+18	-0.0001361561	7914	57331	363	0.996044	STAR
6	182.713733094955	51.3750050594777	22.44608	21.68444	20.24292	19.41423	19.08227	2830	1	411	7.516725588574623e+18	0.5026683	6676	56389	792	0.984373	GALAXY
7	150.089423193165	39.4670880748061	18.96441	17.82906	17.31429	16.99891	16.85583	3560	4	278	1.5267956411104236e+18	0.06366445	1356	53033	274	0.996164	GALAXY
8	189.510984338851	58.7411197772507	21.37376	20.80187	20.84925	21.13449	20.34689	2243	1	353	7.696817897528907e+18	0.7936153	6836	56443	604	0.957787	QSO
9	37.7138728560977	-0.525138228146508	20.77988	19.54618	19.16687	18.89438	18.64286	2700	2	117	1.7553283123029217e+18	0.1060118	1559	53271	183	0.993892	GALAXY
10	201.074980072746	28.7699058867715	25.05349	22.23362	20.8122	19.69488	19.28336	4649	3	120	7.306035245308205e+18	0.567082	6489	56329	257	0.993856	GALAXY
11	151.83091832672	19.8108624669417	24.04443	22.48608	20.59701	19.50985	19.00457	5183	5	142	6.622787444780849e+18	0.5475619	5882	56029	888	0.998885	GALAXY

```
adult=# SELECT * from ml_predict('astra3.cbm'  
'astra3');
```

id	predict	class
0	0.9868595777513302	GALAXY
1	0.9904188657285139	STAR
2	0.9975875623929414	STAR
3	0.9976669380943318	STAR
4	0.9960439244920889	STAR
5	0.9843734017027631	GALAXY
6	0.9961635567874662	GALAXY
7	0.9577871819302538	QSO
8	0.9938922568658763	GALAXY
9	0.9938564131331261	GALAXY

# Inner data model

```
adult=#
adult=# select name, j #> '{data_processing_options,cla
  name      |                class                |  loss_func
-----+-----+-----+-----
  astra     | ["GALAXY", "QSO", "STAR"] | "MultiClass"
  titanic   | [0, 1]                     | "Logloss"
  titanic   | [0, 1]                     | "Logloss"
  boston     | []                         | "RMSE"
  adult     | ["<=50K", ">50K"]         | "Logloss"
(5 rows)
```

# PostgreSQL vs ClickHouse

```
adult=# select * from ml_predict('amazon.cbm' , 'amazon','{
RESOURCE,
MGR_ID,
ROLE_ROLLUP_1,
ROLE_ROLLUP_2,
ROLE_DEPTNAME,
ROLE_TITLE,
ROLE_FAMILY_DESC,
ROLE_FAMILY,
ROLE_CODE}') )
LIMIT 10;
```

index	predict	class
0	5.075591747501174	1
1	4.677445251644691	1
2	3.4881006705946156	1
3	4.654735526605757	1
4	4.546219076437382	1
5	-0.7881046669169504	0
6	5.249330192285552	1
7	4.5361327711227215	1
8	4.542787758485275	1
9	3.9540183530568065	1

(10 rows)

```
catboostEvaluate('/tmp/amazon.cbm',
RESOURCE,
MGR_ID,
ROLE_ROLLUP_1,
ROLE_ROLLUP_2,
ROLE_DEPTNAME,
ROLE_TITLE,
ROLE_FAMILY_DESC,
ROLE_FAMILY,
ROLE_CODE) AS prediction,
^I ACTION AS target
FROM amazon_train LIMIT 10;


SELECT
catboostEvaluate('/tmp/amazon.cbm', RESOURCE, MGR
ACTION AS target
FROM amazon_train
LIMIT 10


Query id: c4975c2f-9380-4619-8b07-2dc9e3886470
```

prediction	target
5.075591747501174	1
4.677445251644691	1
3.4881006705946156	1
4.654735526605757	1
4.546219076437382	1
-0.7881046669169504	0
5.249330192285552	1
4.5361327711227215	1
4.542787758485275	1
3.9540183530568065	1


10 rows in set. Elapsed: 0.111 sec. Processed 8.19 th


# PostgresML


 PostgresML


Search 


<<


PostgresML 


Status 


Manage 


**Notebooks **

Projects 

Models 

Snapshots 

Upload Data 

New Database 

Dashboard | Notebooks | test

▶ Run All


↺ Clear All Output

+ Create New Cell

▶ Run

□ Stop

🗑 Delete

SQL 

1	25139	311198	91261	118026	122392	121143	173805	249618	121145
1	34924	28805	117961	118327	120299	124922	152038	118612	124924
1	80574	55643	118256	118257	117945	280788	280788	292795	119082
1	14354	59575	117916	118150	117920	118568	122142	19721	118570

93.228ms

4

```
1 SELECT pgml.transform(  
2   inputs => ARRAY[  
3     'I am Omar and I live in New York City.'  
4   ],  
5   task => 'token-classification'  
6 ) as ner;
```

**ner**  
[[{"end":9,"entity":"I-PER","index":3,"score":0.9971067309379578,"start":5,"word":"Omar"},{"end":27,"entity":"I-LOC","index":8,"score":0.9993748068809508,"start":24,"word":"New"},{"end":32,"entity":"I-LOC","index":9,"score":0.9993545413017272,"start":28,"word":"York"},{"end":37,"entity":"I-LOC","index":10,"score":0.9994328618049622,"start":33,"word":"City"}]]