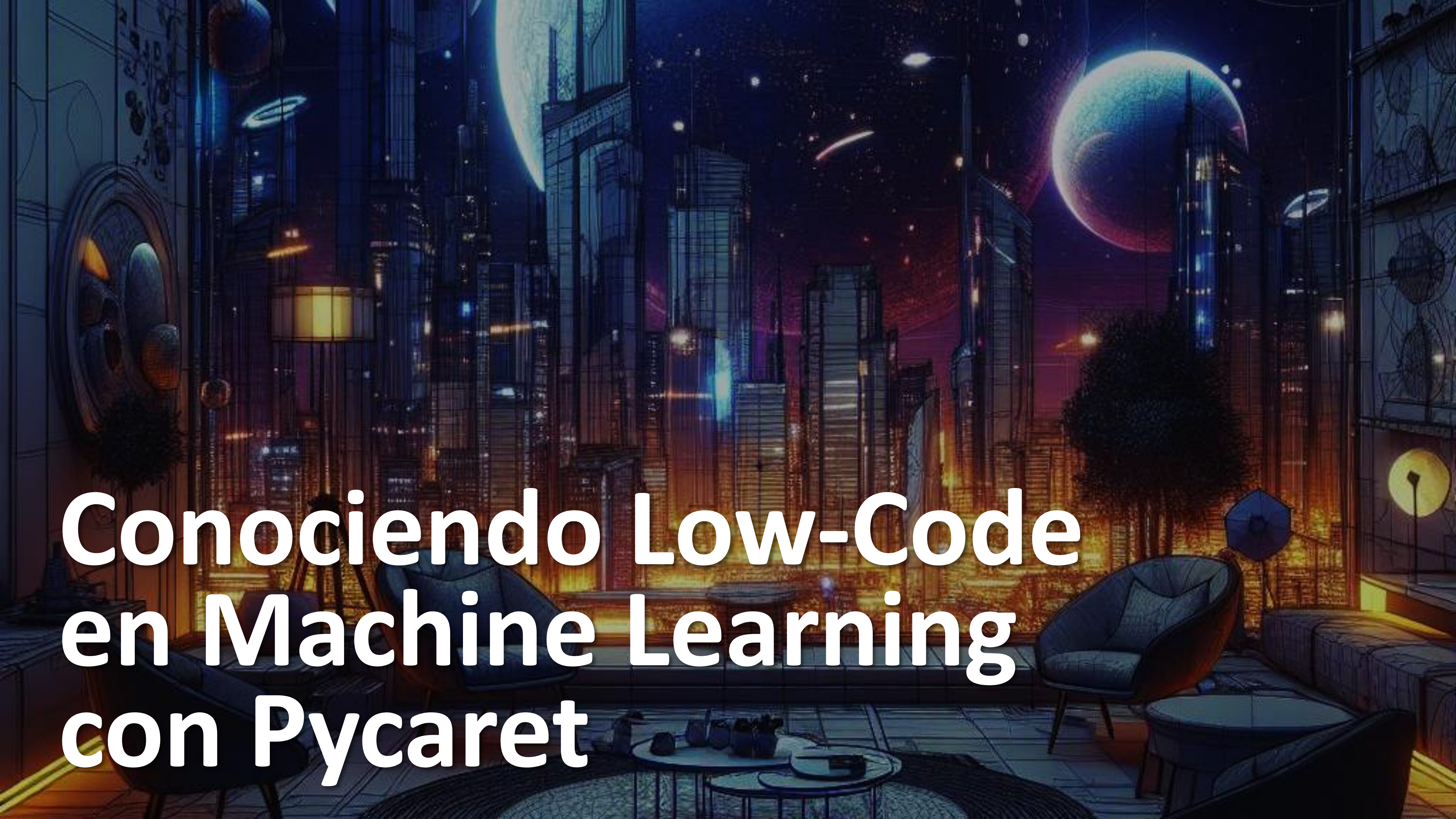


Conociendo Low-Code en Machine Learning con Pycaret

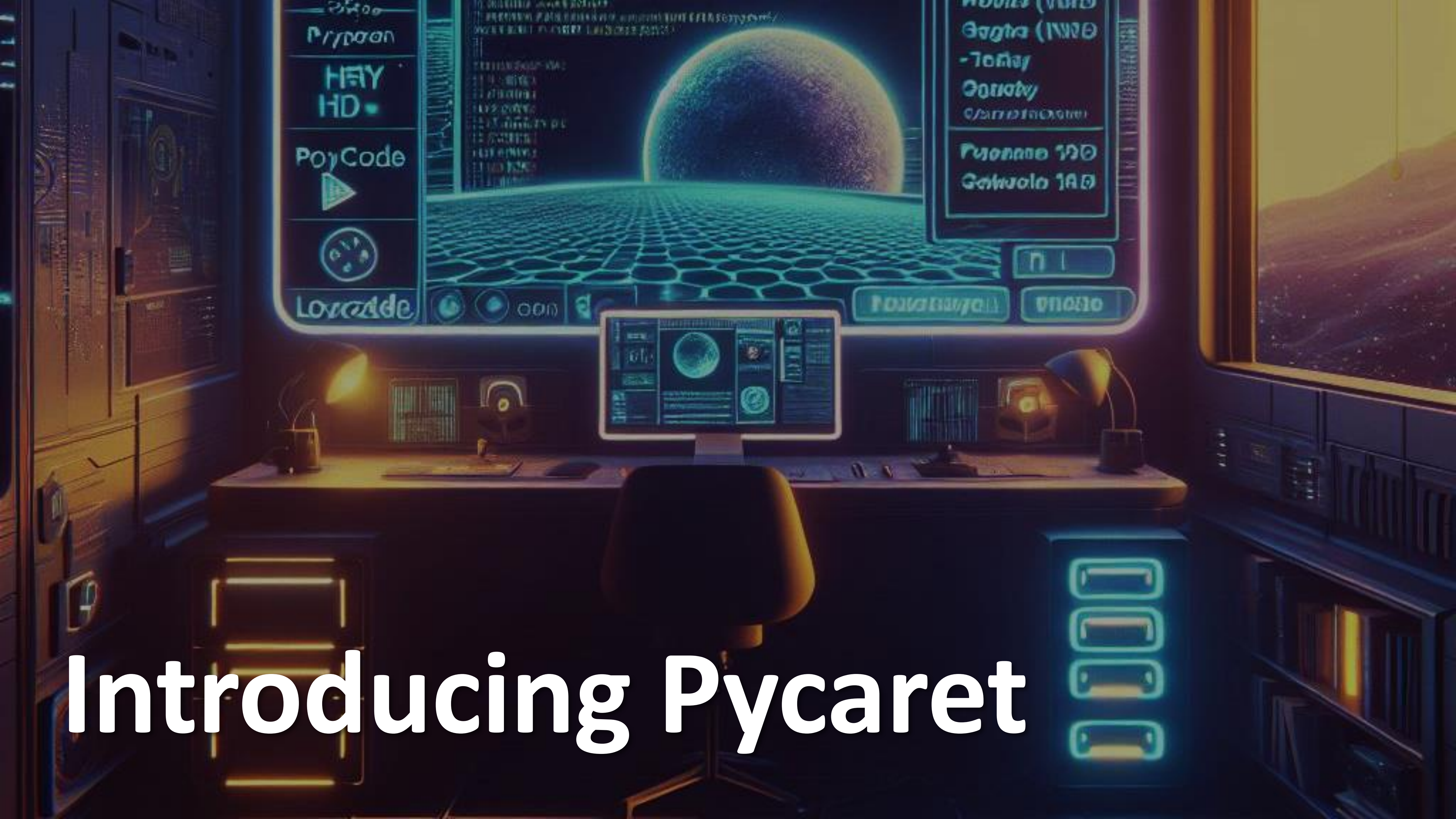
Ana Muñoz Maquera

Data Analyst with experience in fintech, people analytics and business intelligence

Business Management Engineer

Studying a master's degree in quantitative techniques

Data enthusiast y self-taught



Introducing Pycaret

PYCARET

Data Preparation

Model Training

Hyperparameter Tuning

Analysis & Interpretability

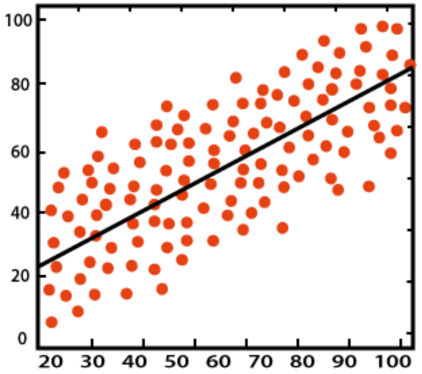
Model Selection

Experiment Logging

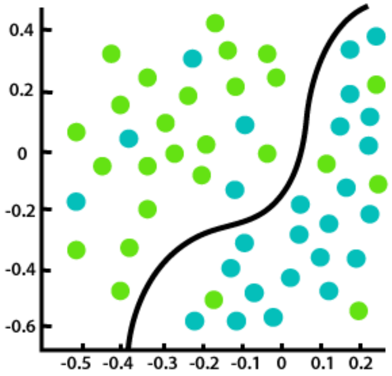
Version: Pycaret 3.0

Model Training

Supervised ML

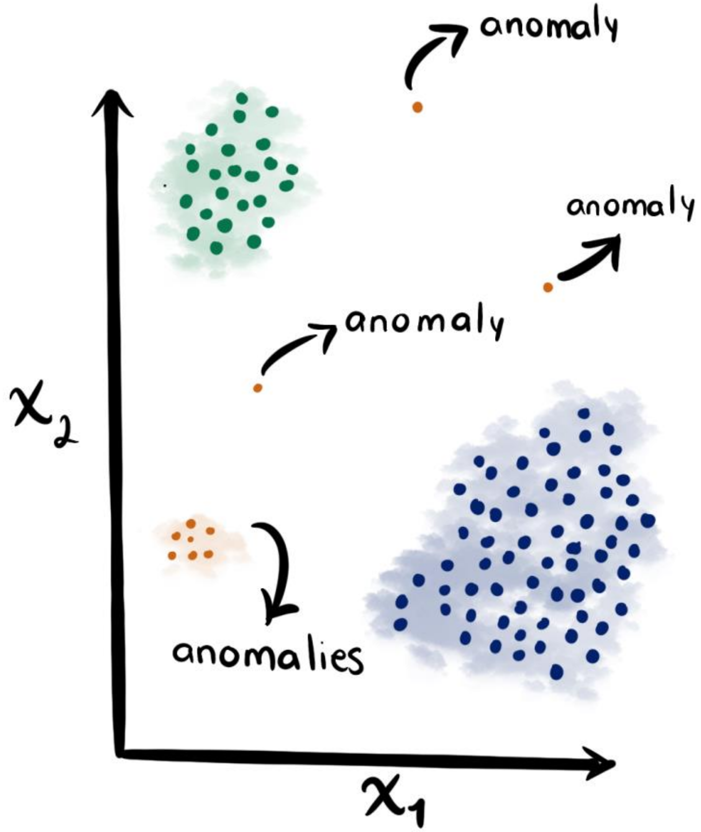


Regression

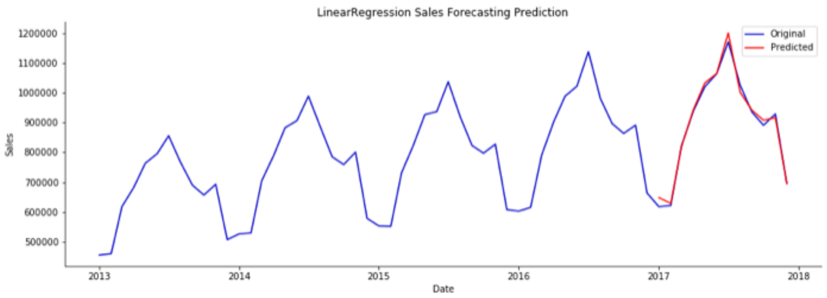


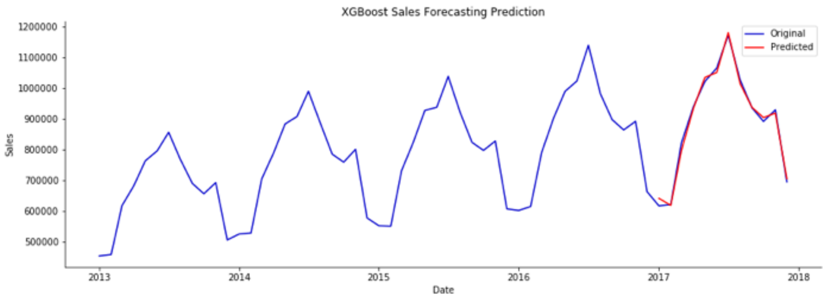
Classification

Unsupervised ML



Time Series *





Demonstration

Installation

Installers

noarch v3.2.0

conda install

To install this package run one of the following:

```
conda install conda-forge::pycaret
```

```
!pip install pycaret
import pycaret
pycaret.__version__
```

```
'3.3.0'
```

Exploring the data frame

Dependent variable

	age	sex	bmi	children	smoker	region	charges
1126	55	male	29.90	0	no	southwest	10214.6360
940	18	male	23.21	0	no	southeast	1121.8739
295	18	male	22.99	0	no	northeast	1704.5681

- **6 columns**
  - **3 numerical variables**
  - **3 categorical variables**
- **1338 records**

Regression with pycaret

```python
from pycaret.regression import *
```

First experiment

```python
r1 = setup(df,
           target = 'charges',
           train_size = 0.8,
           numeric_features = ['age', 'bmi', 'children'],
           categorical_features = ['sex', 'smoker', 'region'],
           preprocess = False,
           session_id = 2024)

best_r1 = compare_models(sort = 'R2')
```

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
lightgbm	Light Gradient Boosting Machine	2727.8776	22559755.4892	4724.6931	0.8404	0.5213	0.3334	0.2070
dummy	Dummy Regressor	9125.5168	145937033.6000	12046.2177	-0.0069	0.9894	1.4886	0.0110

Second experiment

```python
r2 = setup(df,
           target = 'charges',
           train_size = 0.8,
           numeric_features = ['age', 'bmi', 'children'],
           categorical_features = ['sex', 'smoker', 'region'],
           preprocess = True,
           remove_outliers = True,
           outliers_threshold = 0.05,
           remove_multicollinearity = True,
           multicollinearity_threshold = 0.8,
           session_id = 2024)

best_r2 = compare_models(sort = 'R2')
```

Second experiment

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
gbr	Gradient Boosting Regressor	2514.8645	20720291.6177	4523.0500	0.8544	0.4269	0.2936	0.3400
rf	Random Forest Regressor	2602.7697	22216376.6023	4686.6698	0.8440	0.4464	0.3031	0.5450
lightgbm	Light Gradient Boosting Machine	2831.2084	23102869.3708	4786.1431	0.8370	0.5151	0.3481	0.7580
ada	AdaBoost Regressor	3733.9224	23802499.2638	4862.6149	0.8324	0.5772	0.6220	0.2680
et	Extra Trees Regressor	2581.7033	25356295.5460	5009.5272	0.8215	0.4454	0.2813	0.6300
xgboost	Extreme Gradient Boosting	2995.9290	26719500.0000	5143.6417	0.8124	0.5510	0.3760	0.2980
llar	Lasso Least Angle Regression	4049.6758	35752049.6806	5951.0191	0.7506	0.5080	0.3717	0.2510
lr	Linear Regression	4049.3056	35750380.6416	5950.8942	0.7506	0.5091	0.3716	0.7240
lar	Least Angle Regression	4049.3056	35750380.6416	5950.8942	0.7506	0.5091	0.3716	0.2490
lasso	Lasso Regression	4049.6753	35752048.0431	5951.0189	0.7506	0.5080	0.3717	0.3190
ridge	Ridge Regression	4063.6383	35772524.4822	5953.0942	0.7505	0.5051	0.3739	0.3250
br	Bayesian Ridge	4056.2375	35761863.5877	5952.0244	0.7505	0.5063	0.3727	0.2700
dt	Decision Tree Regressor	2987.6097	39816475.4170	6287.4680	0.7219	0.5366	0.3576	0.3400

Second experiment

```
1  get_config('X_transformed').sample(3)
```

	age	sex	bmi	children	smoker	region_northwest	region_southwest	region_northeast	region_southeast
669	40.0	0.0	29.809999	1.0	0.0	0.0	0.0	0.0	1.0
1284	61.0	1.0	36.299999	1.0	1.0	0.0	1.0	0.0	0.0
1075	32.0	0.0	29.590000	1.0	0.0	0.0	0.0	0.0	1.0

```
1  get_config('X_transformed').shape
```

(1284, 9)

Second experiment

```
gbr_r2 = create_model(best_r2)
```

```
print(gbr_r2.get_params())
```

Fold	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	2376.2411	18737288.5832	4328.6590	0.8277	0.4400	0.3070
1	2589.0236	21213281.8676	4605.7879	0.8406	0.4331	0.2676
2	2693.8757	23977304.7522	4896.6626	0.8495	0.4093	0.3289
3	2732.6124	26530314.0278	5150.7586	0.8186	0.4275	0.2655
4	2545.6306	18811735.3286	4337.2497	0.8603	0.4056	0.2571
5	2761.9706	26096497.6475	5108.4731	0.8331	0.4535	0.3497
6	2716.1136	24248251.5989	4924.2514	0.8340	0.4865	0.3122
7	2181.2434	13830209.9235	3718.8990	0.8845	0.4124	0.2794
8	2063.5829	12921919.4990	3594.7072	0.9339	0.3900	0.2929
9	2496.9310	20889919.2045	4570.5491	0.8615	0.4107	0.2757
Mean	2515.7225	20725672.2433	4523.5998	0.8544	0.4269	0.2936
Std	228.4637	4492758.9837	512.5597	0.0322	0.0266	0.0287

Second experiment - Tuned

```
tuned_gbr = tune_model(gbr_r2,
                       n_iter = 15)
```

Fold	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	2418.5004	18771086.2187	4332.5612	0.8274	0.4269	0.2980
1	2500.7349	20562903.2565	4534.6338	0.8455	0.4229	0.2768
2	2478.9521	20265215.6086	4501.6903	0.8728	0.3901	0.3166
3	2753.6853	26421805.2122	5140.2145	0.8193	0.4407	0.2818
4	2435.9650	17114176.0868	4136.9283	0.8729	0.3958	0.2622
5	2708.6977	25258827.7649	5025.8161	0.8384	0.4297	0.3221
6	2720.5099	23890277.4603	4887.7681	0.8365	0.4567	0.3066
7	2137.5130	12832816.4050	3582.2921	0.8928	0.3817	0.2717
8	2036.6591	11874976.8839	3446.0088	0.9393	0.3850	0.3030
9	2334.4798	19292435.6038	4392.3155	0.8721	0.3900	0.2604
Mean	2452.5697	19628452.0501	4398.0229	0.8617	0.4120	0.2899
Std	227.7507	4590851.8643	534.6465	0.0343	0.0252	0.0211

Second experiment - Tuned

```
1  print(tuned_gbr)
```

```
GradientBoostingRegressor(max_features=1.0, min_impurity_decrease=0.3,
                          min_samples_leaf=2, min_samples_split=5,
                          n_estimators=60, random_state=2024, subsample=0.85)
```

Ensemble Model

```
gbr = create_model('gbr', verbose = False)
rf = create_model('rf', verbose = False)
lgbm = create_model('lightgbm', verbose = False)
blender = blend_models([gbr, rf, lgbm])
```

```
VotingRegressor(estimators=[('Gradient Boosting Regressor',
                             GradientBoostingRegressor(random_state=2024)),
                            ('Random Forest Regressor',
                             RandomForestRegressor(n_jobs=-1,
                                                   random_state=2024)),
                            ('Light Gradient Boosting Machine',
                             LGBMRegressor(n_jobs=-1, random_state=2024))],
                n_jobs=-1)
```

Fold	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	2493.7736	19479031.8340	4413.5056	0.8209	0.4828	0.3277
1	2534.7189	21127801.4169	4596.4988	0.8412	0.4359	0.2585
2	2698.5386	23465639.7454	4844.1346	0.8527	0.4140	0.3244
3	2851.1322	28121283.6864	5302.9505	0.8077	0.4578	0.2848
4	2528.7715	18475932.3880	4298.3639	0.8628	0.4341	0.2694
5	2753.9593	26117179.7922	5110.4970	0.8329	0.4449	0.3261
6	2747.4093	24216569.4502	4921.0334	0.8342	0.4971	0.3130
7	2212.0705	14660002.2846	3828.8382	0.8775	0.4294	0.2878
8	2035.1977	13162846.2755	3628.0637	0.9327	0.4011	0.3037
9	2550.3483	20760596.6892	4556.3798	0.8623	0.4274	0.2979
Mean	2540.5920	20958688.3562	4550.0266	0.8525	0.4425	0.2993
Std	239.8942	4513106.4732	505.9118	0.0333	0.0281	0.0229

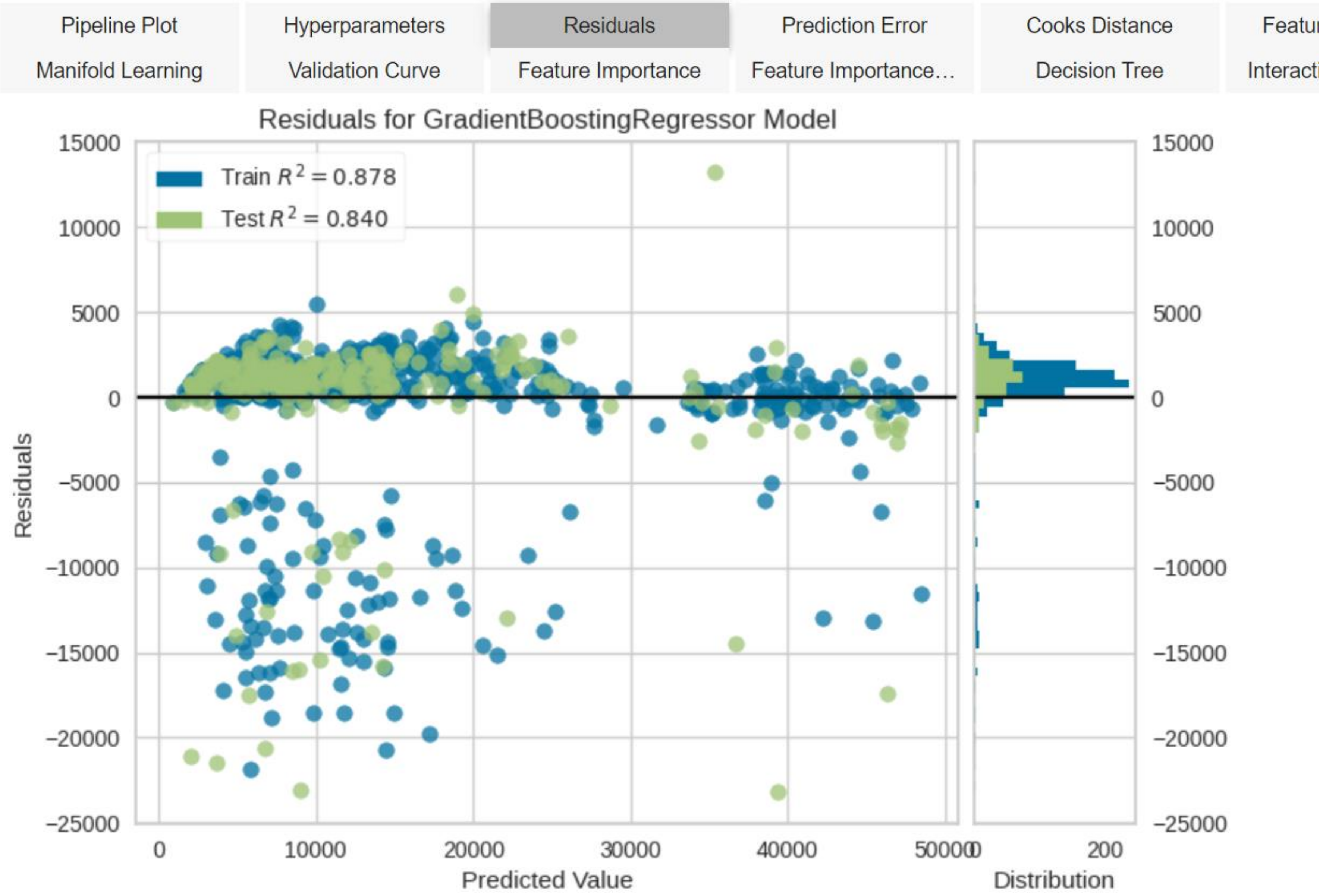
Evaluate Model

```python
evaluate_model(tuned_gbr)
```

Plot Type:

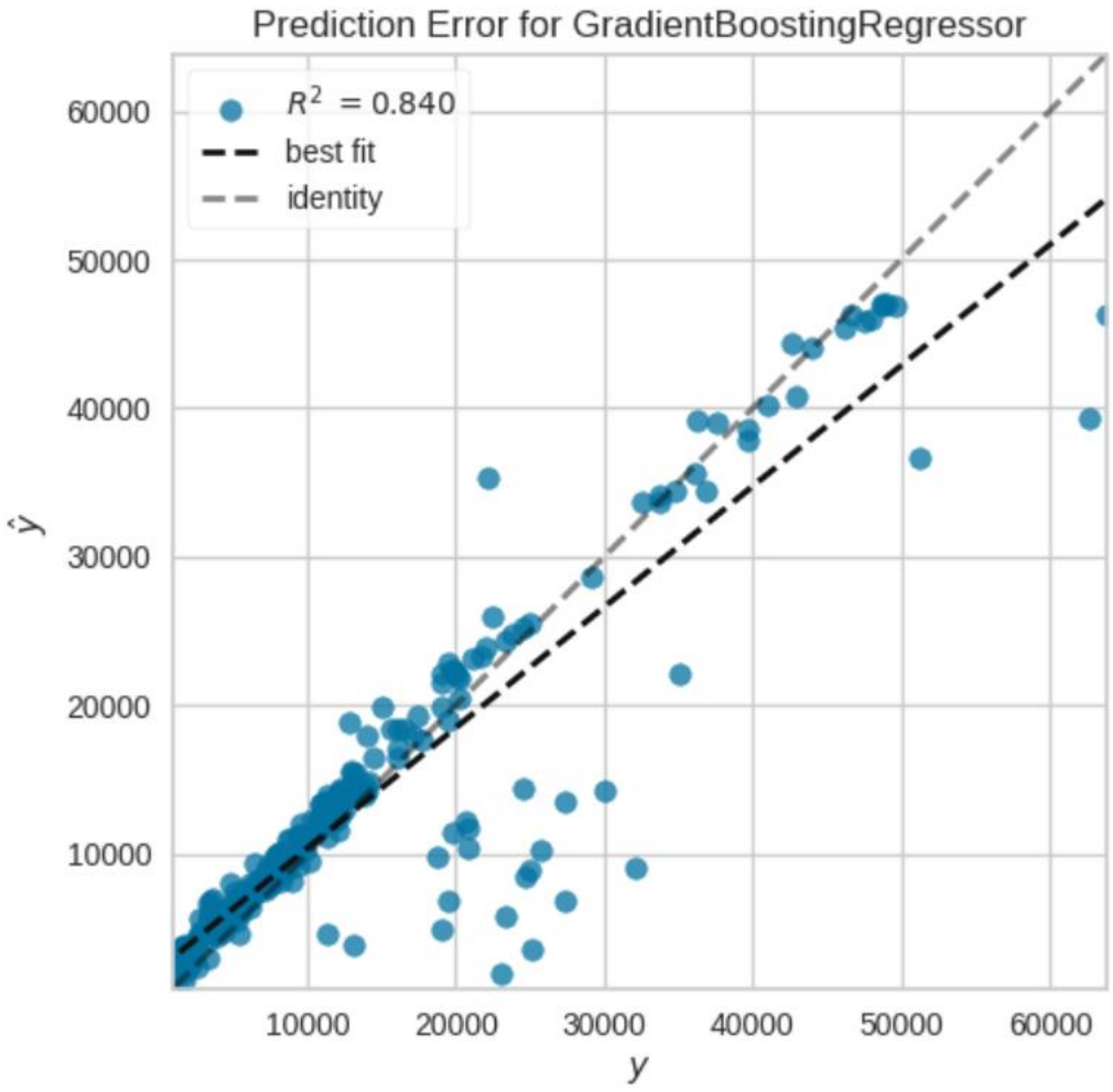
Pipeline Plot	Hyperparameters	Residuals	Prediction Error	Cooks Distance	Feature Selection	Learning Curve
Manifold Learning	Validation Curve	Feature Importance	Feature Importance…	Decision Tree	Interactive Residuals	

Evaluate Model

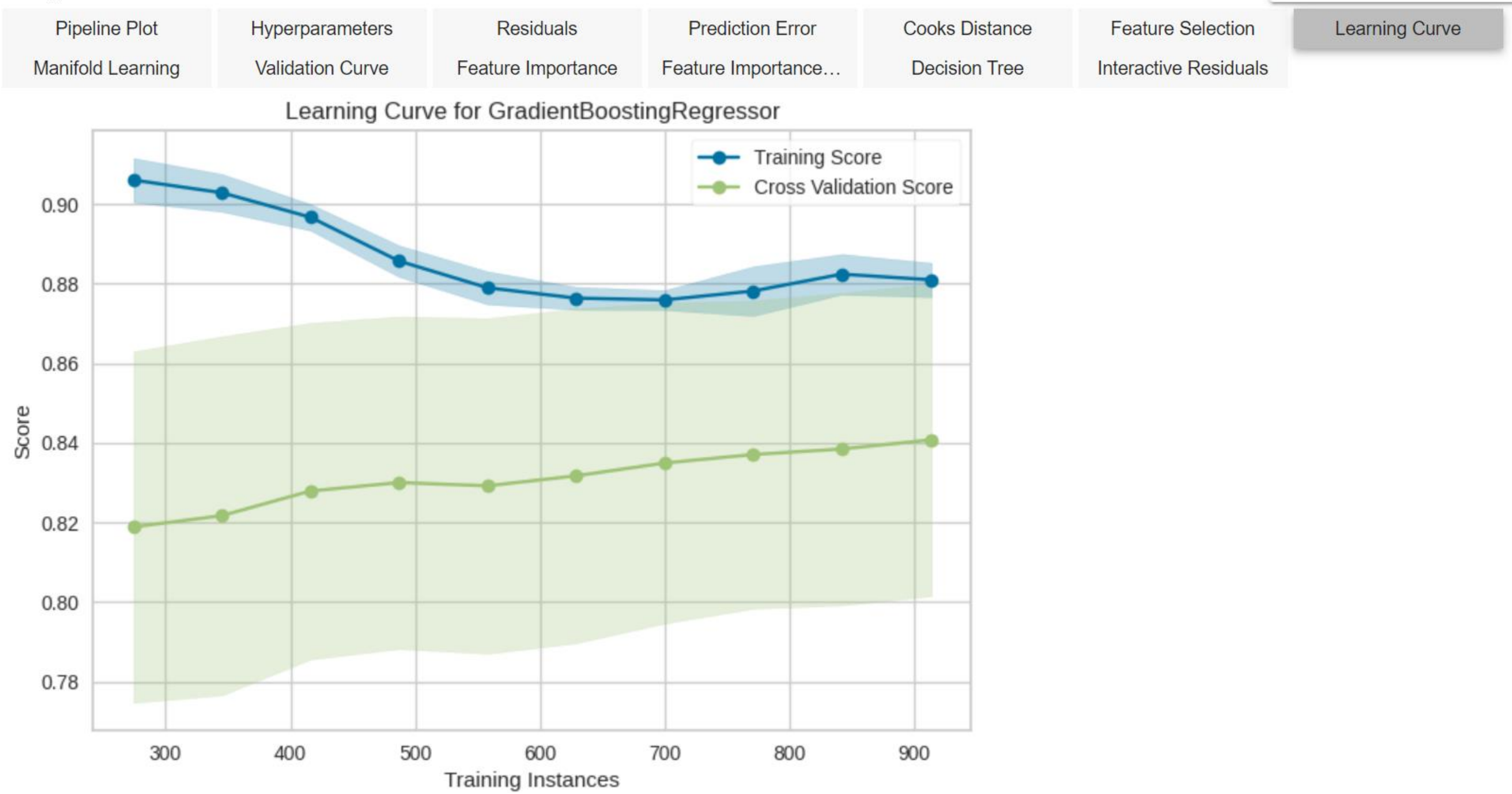


Evaluate Model

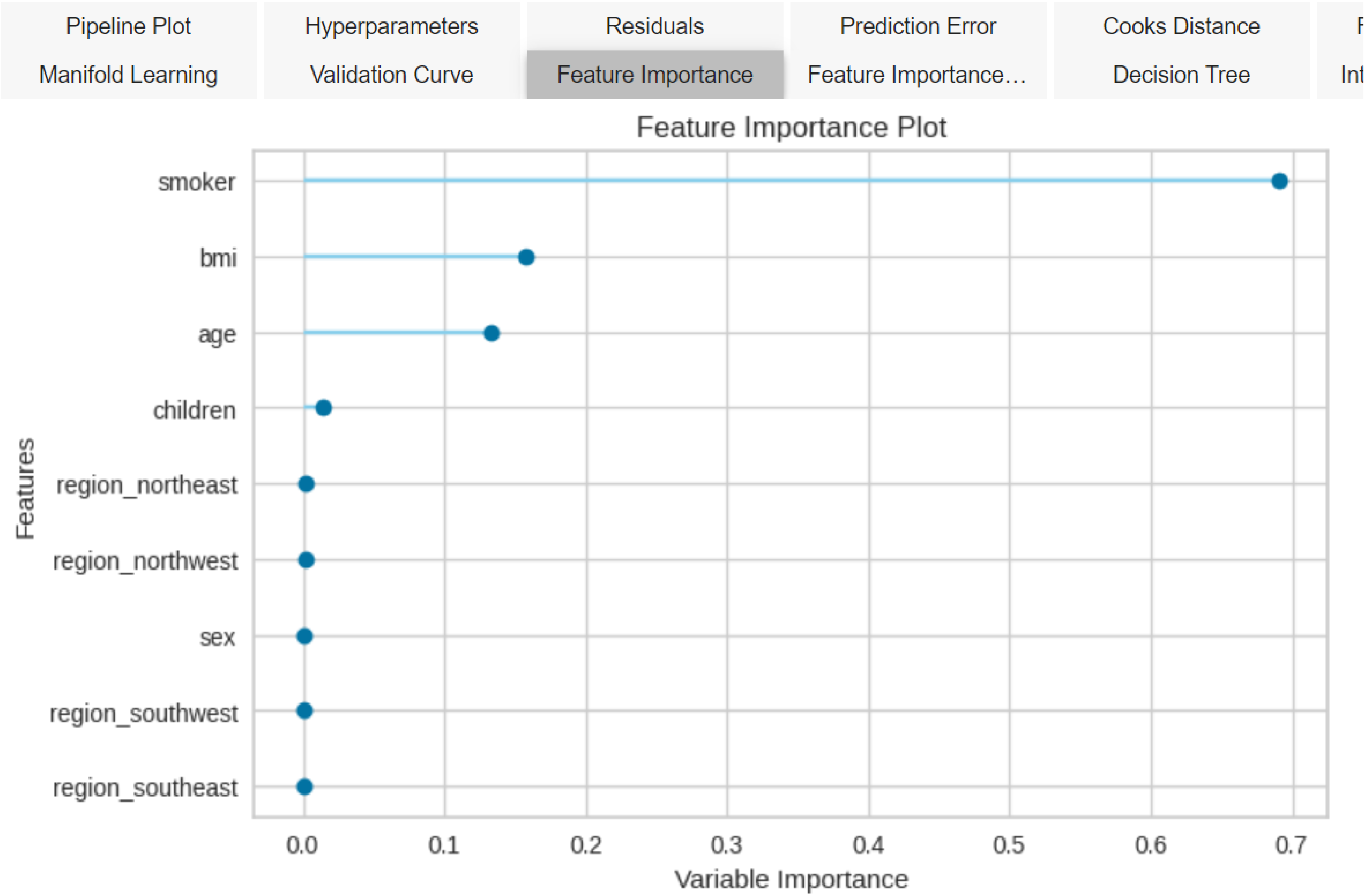
Pipeline Plot	Hyperparameters	Residuals	Prediction Error	Cooks Distance
Manifold Learning	Validation Curve	Feature Importance	Feature Importance…	Decision Tree



Evaluate Model

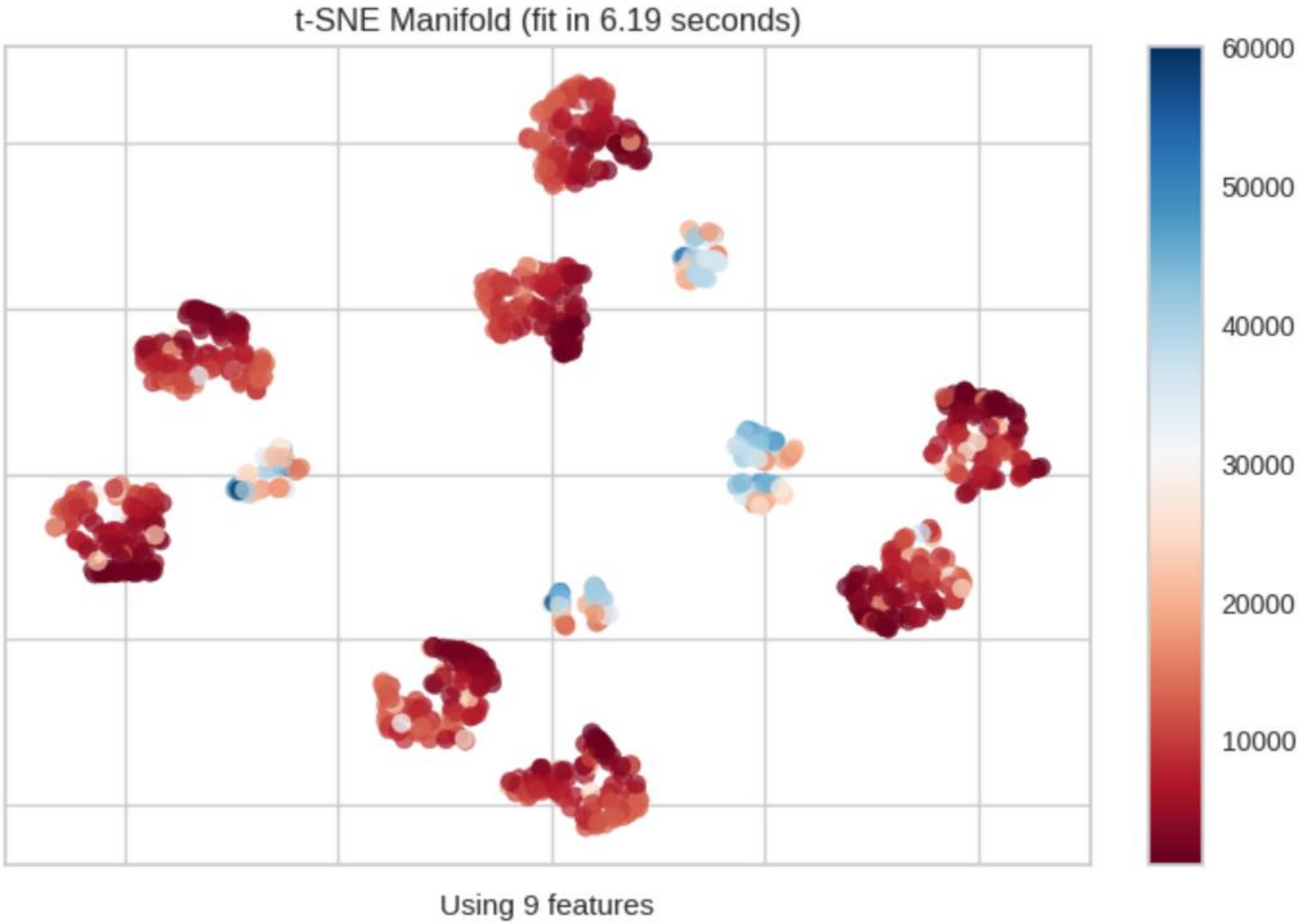


Evaluate Model



Evaluate Model

Pipeline Plot	Hyperparameters	Residuals	Prediction Error	Cooks Distance
Manifold Learning	Validation Curve	Feature Importance	Feature Importance…	Decision Tree



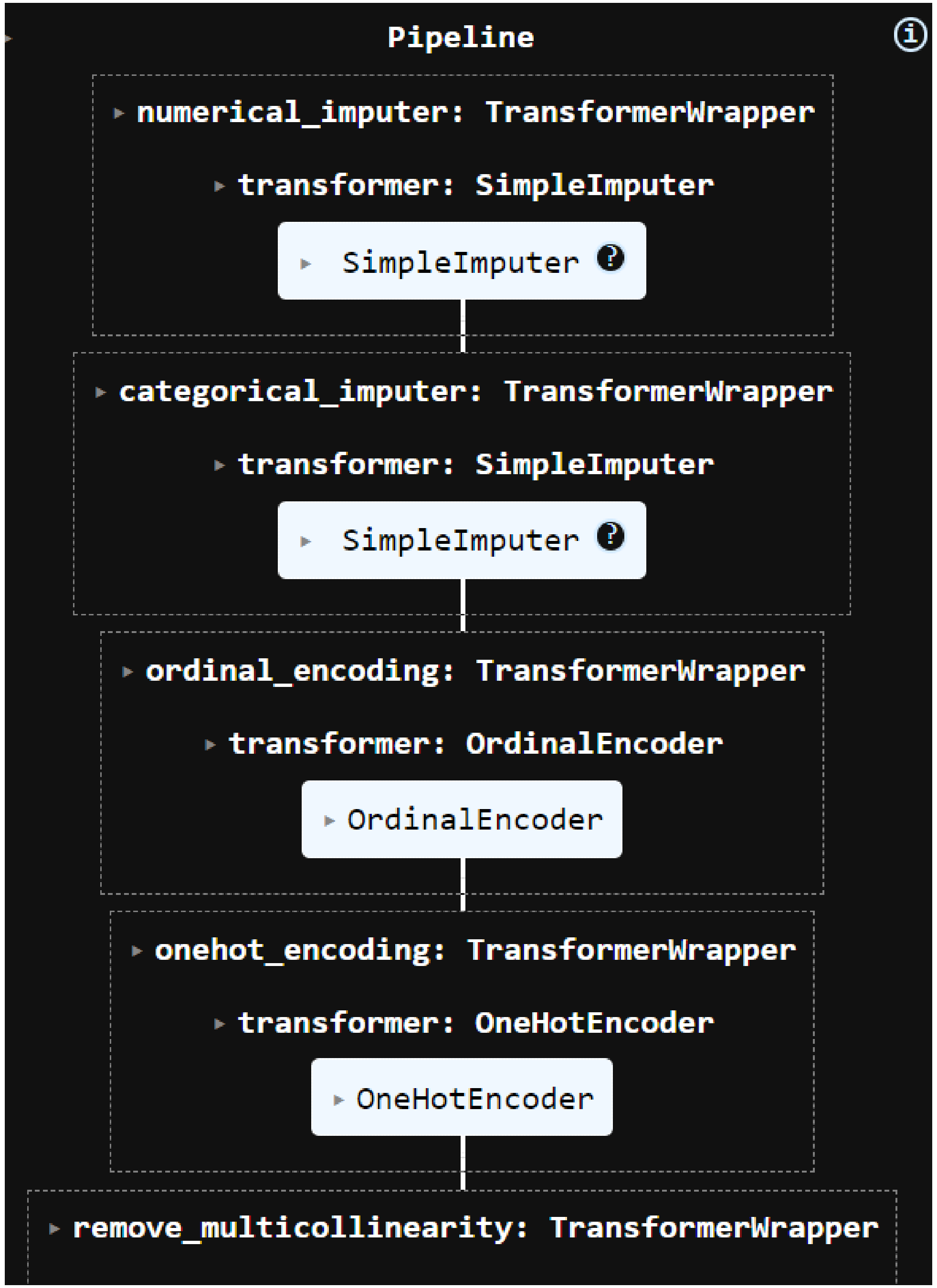
Predictions

```
pred_holdout = predict_model(tuned_gbr)
pred_holdout.sample(3)
```

	age	sex	bmi	children	smoker	region	charges	prediction_label
356	46	male	43.889999	3	no	southeast	8944.115234	8165.572510
816	24	female	24.225000	0	no	northwest	2842.760742	4588.523558
723	19	male	35.400002	0	no	southwest	1263.249023	2049.101873

Model Pipeline

```python
finalize_model(tuned_gbr)
```



Model Pipeline

ordinal_encoding: TransformerWrapper

transformer: OrdinalEncoder

OrdinalEncoder

```
OrdinalEncoder(cols=['sex', 'smoker'], handle_missing='return_nan',
               mapping=[{'col': 'sex', 'data_type': dtype('O'),
                         'mapping': female    0
male      1
NaN      -1
dtype: int64},
                        {'col': 'smoker', 'data_type': dtype('O'),
                         'mapping': no     0
yes    1
NaN   -1
dtype: int64}])
```

onehot_encoding: TransformerWrapper

transformer: OneHotEncoder

OneHotEncoder

```
OneHotEncoder(cols=['region'], handle_missing='return_nan', use_cat_names=True)
```

GradientBoostingRegressor

```
GradientBoostingRegressor(max_features=1.0, min_impurity_decrease=0.3,
                          min_samples_leaf=2, min_samples_split=5,
                          n_estimators=60, random_state=2024, subsample=0.85)
```

More information

Docs

Ask or Search Ctrl + K

PyCaret 3.0

GET STARTED

- Installation
- Quickstart
- Tutorials
- Modules
- Data Preprocessing
- Functions

LEARN PYCARET

- Blog
- Videos
- Cheat sheet
- FAQs
- Examples

Powered by GitBook

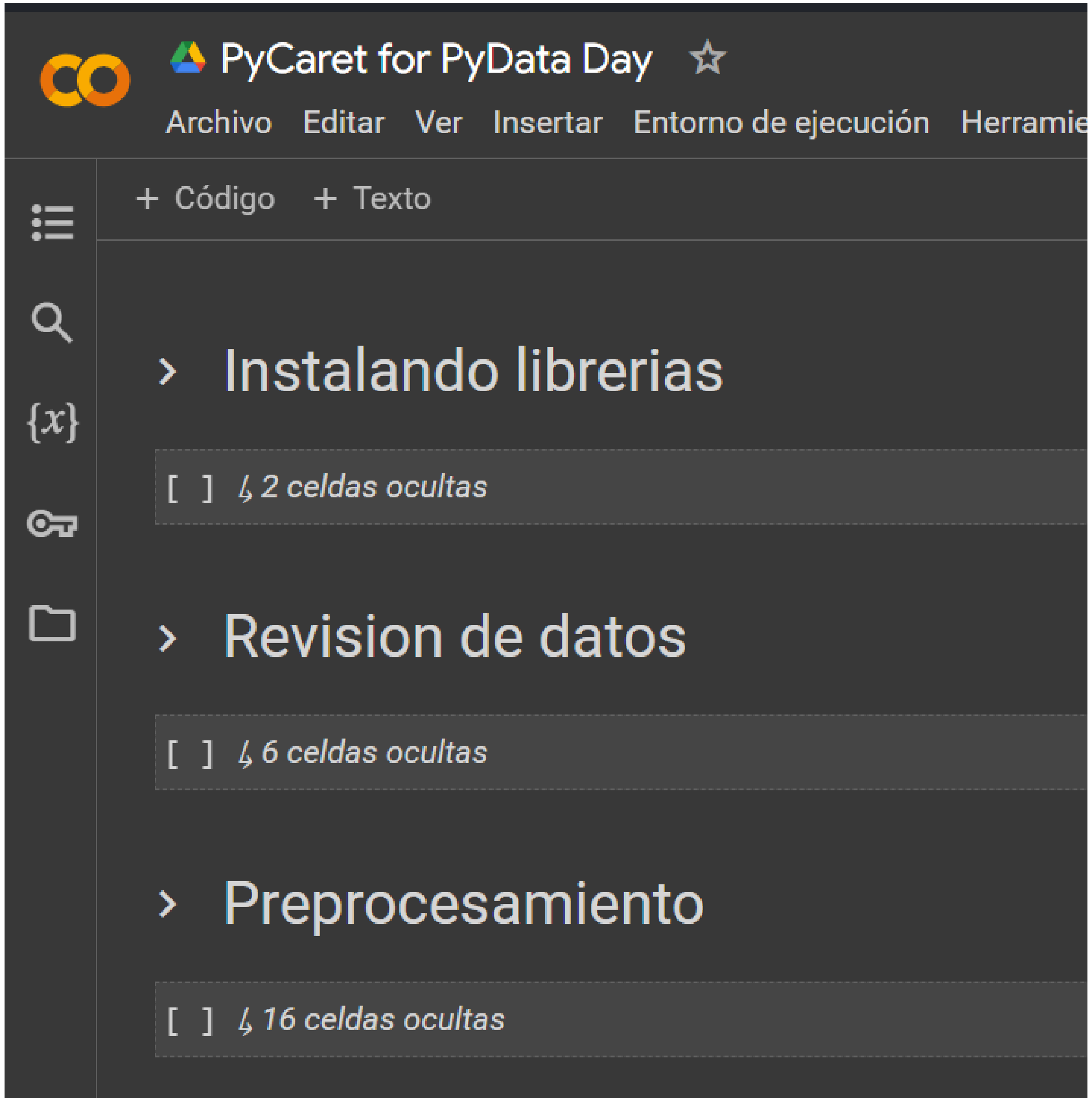
PyCaret 3.0

An open-source, low-code machine learning library in Python

PyCaret is an open-source, low-code machine learning library in Python that automates machine learning workflows. It is an end-to-end machine learning and model management tool that exponentially speeds up the experiment cycle and makes you more productive.

Compared with the other open-source machine learning libraries, PyCaret is an alternate low-code library that can be used to replace hundreds of lines of

https://pycaret.gitbook.io/docs


PyCaret for PyData Day
Archivo Editar Ver Insertar Entorno de ejecución Herramie
+ Código + Texto
Instalando librerias
[ ] 2 celdas ocultas
Revision de datos
[ ] 6 celdas ocultas
Preprocesamiento
[ ] 16 celdas ocultas