

LimeTabularRegression

November 30, 2025

1 Imports

```
[3]: import numpy as np
import pandas as pd
import ast
import matplotlib.pyplot as plt

import sklearn
import sklearn.datasets
import sklearn.ensemble
from sklearn.svm import SVC
from sklearn.pipeline import make_pipeline
from sklearn.metrics import f1_score, accuracy_score
from sklearn import svm
```

```
[4]: import lime
import lime.lime_tabular
```

```
[5]: from xaicsv import xai_csv_utils
```

2 Model Training

```
[8]: diabetes = sklearn.datasets.load_diabetes()

train, test, labels_train, labels_test = sklearn.model_selection.
↳ train_test_split(diabetes.data, diabetes.target, train_size=0.80)
```

```
[9]: regr = svm.SVR()
```

```
[10]: regr = regr.fit(train, labels_train)
y_pred = regr.predict(test)
```

```
[11]: print('SVR MSEError', np.mean((y_pred - labels_test) ** 2))
```

SVR MSEError 5196.631132382582

```
[12]: categorical_features = np.argwhere(np.array([len(set(diabetes.data[:,x])) for x in range(diabetes.data.shape[1])) <= 10).flatten())
```

3 LIME

```
[13]: explainer_lime = lime.lime_tabular.LimeTabularExplainer(train,
                                                                feature_names=diabetes.
                                                                feature_names,
                                                                class_names=['target'],
                                                                categorical_features=categorical_features,
                                                                verbose=False,
                                                                mode='regression',
                                                                random_state = 42)
```

```
[14]: df_test = pd.DataFrame(data=test, columns=diabetes.feature_names)
```

```
[15]: list_labels = ["target"]
list_index_labels= list(range(0, len(list_labels)))
features = diabetes.feature_names

lime_weights = []
for i in range(len(df_test)):
    exp = explainer_lime.explain_instance(df_test.iloc[0].values,
                                          regr.predict,
                                          num_features=len(diabetes.
                                          feature_names),
                                          labels=list_index_labels)

    weight_val = xai_csv_utils.lime_values_to_weights_dict(
        lime_explanation=exp,
        class_names=list_labels)

    lime_weights.append(weight_val)
```

```
[16]: df_test['label'] = labels_test
df_test["predict_label"] = y_pred
df_test['lime_weights'] = lime_weights
```

```
[17]: df_test.head()
```

```
[17]:
```

	age	sex	bmi	bp	s1	s2	s3 \
0	-0.070900	-0.044642	-0.057941	-0.081413	-0.045599	-0.028871	-0.043401
1	0.019913	-0.044642	0.004572	0.045972	-0.018080	-0.054549	0.063367
2	0.027178	-0.044642	-0.007284	-0.050427	0.075484	0.056619	0.033914
3	0.016281	-0.044642	-0.024529	0.035644	-0.007073	-0.003193	-0.013948
4	0.048974	0.050680	0.088642	0.087287	0.035582	0.021546	-0.024993

	s4	s5	s6	label	predict_label	\
0	-0.002592	0.001148	-0.005220	162.0	130.952091	
1	-0.039493	0.028658	0.061054	191.0	138.732086	
2	-0.002592	0.043444	0.015491	95.0	137.450863	
3	-0.002592	0.015568	0.015491	66.0	141.756378	
4	0.034309	0.066051	0.131470	310.0	156.148098	

	lime_weights
0	{'target': [('bmi <= -0.03', -5.77496528024560...]
1	{'target': [('bmi <= -0.03', -6.06342800890212...]
2	{'target': [('bmi <= -0.03', -6.00607707399959...]
3	{'target': [('bmi <= -0.03', -5.98516584553083...]
4	{'target': [('bmi <= -0.03', -5.95316590861652...]

```
[18]: df_test.to_csv("diabetes_results_lime.csv", sep=';', index=False)
```

4 Read LIME CSV

```
[19]: df = pd.read_csv("diabetes_results_lime.csv", sep=";")
```

```
[20]: sample_weights = df.iloc[25]["lime_weights"]
for label in (list_labels):
    fig = xai_csv_utils.weights_dict_to_pyplot(sample_weights, label)
```



