ch02. 의사결정트리 시각화

학습 내용

- 1. 의사결정트리 모델을 시각화를 통해 이해해 봅니다.
- 2. 다양한 데이터 셋을 시각화 해 봅니다.
    - boston data - 집값 데이터 예측 모델 시각화
    - IRIS 꽃의 종류 예측 모델 시각화
    - 유방암 예측 모델 시각화
    - 당뇨병 진행도 예측 모델 시각화
    - 웹에서 파일 저장해서 확인하기

목차

01 시각화 라이브러리 설치

목차로 이동하기

In [3]:

```
import sys
import os

if 'google.colab' in sys.modules:
  !pip install -q dtreeviz
```

라이브러리 설치

In [8]:

```
import sys
import os

# add library module to PYTHONPATH
print(os.getcwd())
sys.path.append(f"{os.getcwd()}/../")
```

```
/content
```

In [9]:

```
from sklearn.datasets import *
from dtreeviz.trees import *
from IPython.display import Image, display_svg, SVG
```

02 보스턴 집값 예측 모델 시각화

목차로 이동하기

회귀 트리(Regression tree)

- 데이터 셋 : boston data
- url : boston house-prices dataset (https://archive.ics.uci.edu/ml/machine-learning-databases/housing/) (regression).

In [10]:

```
model = tree.DecisionTreeRegressor(max_depth=3)
boston = load_boston()

# 데이터 나누기
X_train = boston.data
y_train = boston.target

# 모델 학습
model.fit(X_train, y_train)

# 모델 시각화
viz = dtreeviz(model,
               X_train,
               y_train,
               target_name='price',  # leaf node에 보여지는 target 표시
               feature_names=boston.feature_names,
               title="Boston data set regression",   # 제목
               fontname="Arial",  # 글씨 폰트
               title_fontsize=16,  # 타이틀 폰트 사이즈
               colors = {"title":"purple"}
              )
viz
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:87: FutureWarnin
g: Function load_boston is deprecated; `load_boston` is deprecated in 1.0 and will b
e removed in 1.2.

    The Boston housing prices dataset has an ethical problem. You can refer to
    the documentation of this function for further details.

    The scikit-learn maintainers therefore strongly discourage the use of this
    dataset unless the purpose of the code is to study and educate about
    ethical issues in data science and machine learning.

    In this special case, you can fetch the dataset from the original
    source::

        import pandas as pd
        import numpy as np


        data_url = "http://lib.stat.cmu.edu/datasets/boston"
        raw_df = pd.read_csv(data_url, sep="₩s+", skiprows=22, header=None)
        data = np.hstack([raw_df.values[::2, :], raw_df.values[1::2, :2]])
        target = raw_df.values[1::2, 2]

    Alternative datasets include the California housing dataset (i.e.
    :func:`~sklearn.datasets.fetch_california_housing`) and the Ames housing
    dataset. You can load the datasets as follows::

        from sklearn.datasets import fetch_california_housing
        housing = fetch_california_housing()

    for the California housing dataset and::

        from sklearn.datasets import fetch_openml
        housing = fetch_openml(name="house_prices", as_frame=True)
    for the Ames housing dataset.

  warnings.warn(msg, category=FutureWarning)
WARNING:matplotlib.font_manager:findfont: Font family ['Arial'] not found. Falling b
ack to DejaVu Sans.
```

Out[10]:

```
<dtreeviz.trees.DTreeViz at 0x7f7cac997b50>
```

이미지 스케일 조정

In [11]:

```
dtreeviz(model,
       X_train,
       y_train,
       target_name='price',  # leaf node에 보여지는 target 표시
       feature_names=boston.feature_names,
       scale=0.7              # scale를 통해 이미지의 크기를 조절
      )
```

Out[11]:

```
<dtreeviz.trees.DTreeViz at 0x7f7cac997d10>
```

03 IRIS 꽃의 종류 예측 모델 시각화

목차로 이동하기

분류 트리(Classification tree) - 다항분류

In [12]:

```
model = tree.DecisionTreeClassifier(max_depth=2)


# 데이터 나누기
iris = load_iris()
X_train = iris.data
y_train = iris.target

# 모델 학습
model.fit(X_train, y_train)

# 모델 시각화
viz = dtreeviz(model,
               X_train,
               y_train,
               target_name='iris type',
               fontname="Arial",
               feature_names=iris.feature_names,
               class_names=["setosa", "versicolor", "virginica"],
               histtype= 'barstacked')  # 히스토 그램 형태 : 기본(barstacked)
viz
```

```
WARNING:matplotlib.font_manager:findfont: Font family ['Arial'] not found. Falling b
ack to DejaVu Sans.
/usr/local/lib/python3.7/dist-packages/numpy/core/fromnumeric.py:3208: VisibleDeprec
ationWarning: Creating an ndarray from ragged nested sequences (which is a list-or-t
uple of lists-or-tuples-or ndarrays with different lengths or shapes) is deprecated.
If you meant to do this, you must specify 'dtype=object' when creating the ndarray.
  return asarray(a).size
/usr/local/lib/python3.7/dist-packages/matplotlib/cbook/__init__.py:1376: VisibleDep
recationWarning: Creating an ndarray from ragged nested sequences (which is a list-o
r-tuple of lists-or-tuples-or ndarrays with different lengths or shapes) is deprecat
ed. If you meant to do this, you must specify 'dtype=object' when creating the ndarr
ay.
  X = np.atleast_1d(X.T if isinstance(X, np.ndarray) else np.asarray(X))
WARNING:matplotlib.font_manager:findfont: Font family ['Arial'] not found. Falling b
ack to DejaVu Sans.
WARNING:matplotlib.font_manager:findfont: Font family ['Arial'] not found. Falling b
ack to DejaVu Sans.
```

Out[12]:

```
<dtreeviz.trees.DTreeViz at 0x7f7cac1c2310>
```

04 유방암 예측 모델 시각화

목차로 이동하기

Breast Cancer Wisconsin Dataset
(http://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+%28diagnostic%29)

In [ ]:

```
model = tree.DecisionTreeClassifier(max_depth=2)
cancer = load_breast_cancer()

X_train = cancer.data
y_train = cancer.target
model.fit(X_train, y_train)

viz = dtreeviz(model,
               X_train,
               y_train,
               target_name='cancer',
               feature_names=cancer.feature_names,
               class_names=["malignant", "benign"],
               orientation='LR')
viz
```

```
/usr/local/lib/python3.7/dist-packages/numpy/core/_asarray.py:83: VisibleDeprecation
Warning: Creating an ndarray from ragged nested sequences (which is a list-or-tuple
of lists-or-tuples-or ndarrays with different lengths or shapes) is deprecated. If y
ou meant to do this, you must specify 'dtype=object' when creating the ndarray
  return array(a, dtype, copy=False, order=order)
```

Out[61]:

```
<dtreeviz.trees.DTreeViz at 0x7f9fcf97a090>
```

05 당뇨병 진행도 예측 모델 시각화

목차로 이동하기

컬럼명	설명	데이터 유형
age	나이	숫자
sex	성별	명목형
bmi	체질량 지수	숫자
bp	평균 혈압	숫자
s1	혈청 측정값1	숫자
s2	혈청 측정값2	숫자
s3	혈청 측정값3	숫자
s4	혈청 측정값4	숫자
s5	혈청 측정값5	숫자
s6	혈청 측정값6	숫자
Y	10개변수 측정 후, 당뇨병 진행도	숫자

In [13]:

```
model = tree.DecisionTreeRegressor(max_depth=3)

# 데이터 나누기
diabetes = load_diabetes()
X_train = diabetes.data
y_train = diabetes.target

# 모델 학습
model.fit(X_train, y_train)

X = diabetes.data[np.random.randint(0, len(diabetes.data)),:]

viz = dtreeviz(model,
               X_train,
               y_train,
               target_name='progress',
               feature_names=diabetes.feature_names,
               X=X,
               show_node_labels = True,
               scale=.7
              )
viz
```

Out[13]:

```
<dtreeviz.trees.DTreeViz at 0x7f7cac019710>
```

06 웹에서 확인(파일 저장 및 다운로드)

목차로 이동하기

In [14]:

```
viz.save("decision_tree_diabetes.svg")

from google.colab import files
files.download("decision_tree_diabetes.svg")
```

```
<IPython.core.display.Javascript object>

<IPython.core.display.Javascript object>
```

REF

- https://colab.research.google.com/github/parrt/dtreeviz/blob/master/notebooks/examples.ipynb (https://colab.research.google.com/github/parrt/dtreeviz/blob/master/notebooks/examples.ipynb)
- https://towardsdatascience.com/4-ways-to-visualize-individual-decision-trees-in-a-random-forest-7a9beda1d1b7 (https://towardsdatascience.com/4-ways-to-visualize-individual-decision-trees-in-a-random-forest-7a9beda1d1b7)