

```
In [23]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
In [24]: from sklearn import model_selection
from sklearn import preprocessing
from sklearn import metrics
from sklearn import tree
```

```
In [25]: data = pd.read_csv('diabetes.csv')
```

```
In [26]: y = data['Outcome']
X = data.drop(columns=['Outcome'], axis=1)
```

Trening, validacioni i test skup čemo u ovom primeru podeliti u odnosu 56:14:30.

```
In [27]: X_train_and_val, X_test, y_train_and_val, y_test = model_selection.train_test_split(
X, y, test_size = 0.3, random_state = 7, stratify = y)
```

```
In [28]: X_train, X_val, y_train, y_val = model_selection.train_test_split(
X_train_and_val, y_train_and_val, train_size = 0.8,
random_state = 7, stratify = y_train_and_val)
```

```
In [29]: scaler = preprocessing.StandardScaler()
scaler.fit(X_train)
X_train = scaler.transform(X_train)
X_val = scaler.transform(X_val)
X_test = scaler.transform(X_test)
```

```
In [30]: max_depths = [2,3,4,5,6,7]
```

```
In [31]: criterions = ['gini', 'entropy', 'log_loss']
```

```
In [32]: max_features = [0.75, 0.8, 0.85, 0.9, 0.95]
```

```
In [33]: best_f1_score = 0
best_max_depth = None
best_criterion = None
best_max_feature = None
```

```
In [34]: for max_depth in max_depths:
    for criterion in criterions:
        for max_features in max_features:
            model = tree.DecisionTreeClassifier(criterion=criterion,
                                                max_features=max_features,
                                                max_depth=max_depth, random_state=7)

            model.fit(X_train, y_train)
            f1_score = metrics.f1_score(y_val, model.predict(X_val))
            if f1_score > best_f1_score:
                best_f1_score = f1_score
                best_max_depth = max_depth
                best_criterion = criterion
                best_max_features = max_features
```

```
In [35]: print('Najbolji f1-score na validacionom skupu je: ', best_f1_score)

Najbolji f1-score na validacionom skupu je: 0.5952380952380952
```

```
In [36]: print('Najbolji hiperparametri modela su: ', best_max_depth,
```

```
best_criterion, best_max_features)
```

Najbolji hiperparametri modela su: 2 gini 0.75

Kreiramo konačni model, koristeći trening skup:

```
In [37]: best_model = tree.DecisionTreeClassifier(criterion=best_criterion,
                                                max_features=best_max_features,
                                                max_depth=best_max_depth,
                                                random_state=7)

best_model.fit(X_train, y_train)
```

```
Out[37]: ▼ DecisionTreeClassifier
DecisionTreeClassifier(max_depth=2, max_features=0.75, random_state=7)
```

```
In [38]: f1_score = metrics.f1_score(y_test, best_model.predict(X_test))
```

```
In [39]: print('f1-score na test skupu je: ', f1_score)
```

f1-score na test skupu je: 0.6040268456375839