

```
# Load the diabetes dataset

from sklearn import tree
from pandas import read_csv
import os
import numpy as np
df = read_csv("diabetes.csv")

# Separate the target variable (Outcome) from the features
x = np.array(df.drop(["Outcome"], 1))
y = np.array(df["Outcome"])

<ipython-input-8-2379300d97f2>:10: FutureWarning: In a future version of pandas all arguments of DataFrame.drop except for the argument
x = np.array(df.drop(["Outcome"], 1))
```

```
# Display the first few rows of the dataset to inspect the data
df.head()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFu
0	6	148	72	35	0	33.6	
1	7	85	66	29	0	26.6	
2	8	183	64	0	0	23.3	
3	1	89	66	23	94	28.1	
4	0	137	40	35	168	43.1	

```
# Feature Selection Techniques
from sklearn.feature_selection import SelectKBest, chi2, RFE
```

```
# 1. Univariate Feature Selection (SelectKBest)
```

```
k_best = SelectKBest(score_func=chi2, k=5)
x_k_best = k_best.fit_transform(x,y)
x_k_best

array([[ 6. , 148. ,  0. , 33.6, 50. ],
       [ 7. ,  85. ,  0. , 26.6, 31. ],
       [ 8. , 183. ,  0. , 23.3, 32. ],
       ...,
       [ 5. , 121. , 112. , 26.2, 30. ],
       [ 1. , 126. ,  0. , 30.1, 47. ],
       [ 1. ,  93. ,  0. , 30.4, 23. ]])
```

```
# 2. Recursive Feature Elimination (RFE)
```

```
from sklearn.ensemble import RandomForestClassifier
estimator = RandomForestClassifier(random_state=42) # You can change the estimator as needed
rfe = RFE(estimator, n_features_to_select=4, step=1)
x_rfe = rfe.fit_transform(x, y)
x_rfe
```

```
array([[148. , 33.6 , 0.627, 50. ],
       [ 85. , 26.6 , 0.351, 31. ],
       [183. , 23.3 , 0.672, 32. ],
       ...,
       [121. , 26.2 , 0.245, 30. ],
       [126. , 30.1 , 0.349, 47. ],
       [ 93. , 30.4 , 0.315, 23. ]])
```

```
# 3. Feature Importance from Tree-based Model (Random Forest)
```

```
rf_model = RandomForestClassifier(random_state=42)
rf_model.fit(x, y)
feature_importances = rf_model.feature_importances_
sorted_indices = np.argsort(feature_importances)[::-1]
top_features_indices = sorted_indices[:4]
x_rf_importance = x[:, top_features_indices]
x_rf_importance
```

```
array([[148. , 33.6 , 50. , 0.627],
       [ 85. , 26.6 , 31. , 0.351],
       [183. , 23.3 , 32. , 0.672],
       ...,
       [121. , 26.2 , 30. , 0.245],
```

```
[126. , 30.1 , 47. , 0.349],  
[ 93. , 30.4 , 23. , 0.315]])
```