SIT384 Cyber security analytics

Distinction Task 6.2D: LogisticRegression and Decision tree

Task description:

You are given one dataset “payment_fraud.csv”.

It has the following information:

accountAgeDays: how long the account has been created (in days)

numItems: number of items bought

localTime: when the payment was made (it has been converted to a float number)

paymentMethod: how the payment was made (by paypal, storecredit or creditcard)

paymentMethodAgeDays: how long the payment was completed (in days)

label: fraud payment (1) or not (0)

Sample data:

accountAgeDays	numItems	localTime	paymentMethod	paymentMethodAgeDays	label
29	1	4.745402	paypal	28.20486111	0
725	1	4.742303	storecredit	0	0
845	1	4.921318	creditcard	0	0

(The above data is for demonstration purposes only. Please download the full version of payment_fraud.csv.)

In the data set, there is one column stands out because it is of non-numerical type: paymentMethod, called a categorical variable because it takes on a value indicating the category it belongs to. Many machine learning algorithms require all features to be numeric. We can use pandas.get_dummies() to convert variables from categorical to numeric:

```
# Convert categorical feature into dummy variables with one-hot encoding
df = pd.get_dummies(df, columns=['paymentMethod'])
```

You will find that three new columns have been added to the table: paymentMethod_creditcard, paymentMethod_paypal, and paymentMethod_storecredit. Each of these features is a binary feature (0 or 1), and each row has exactly one of these features set to 1, hence the name of this method of categorical variable encoding: one-hot encoding. These variables are called dummy variables in statistics terminology.

You are asked to:

- split the datasets by setting test_size=0.33 in train_test_split();
- apply a supervised learning algorithm logistic regression to this data;

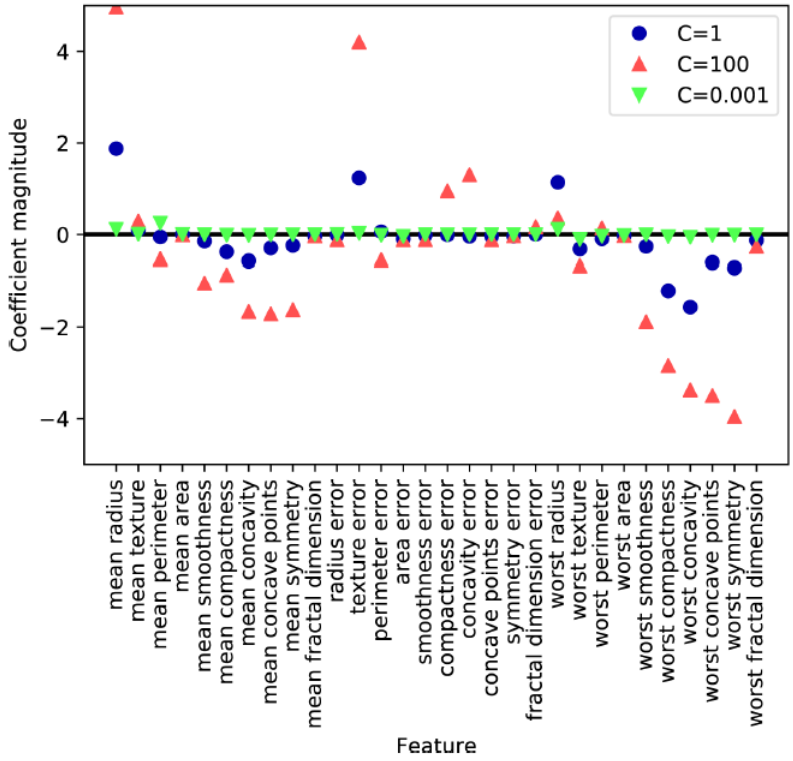
- build logistic regression models with C parameter set to default (C=1), C=100, C=10, C=0.01 and C=0.001, respectively;
- create a plot to visualize the coefficients of the above models;
- print the training set score and test set score of these models with different C parameters;
- apply a supervised learning algorithm decision tree to this data;
- print the training set score and test set score of the decision tree model;
- print the decision tree depth and feature importances;
- create a plot to visualize the decision tree feature importances.

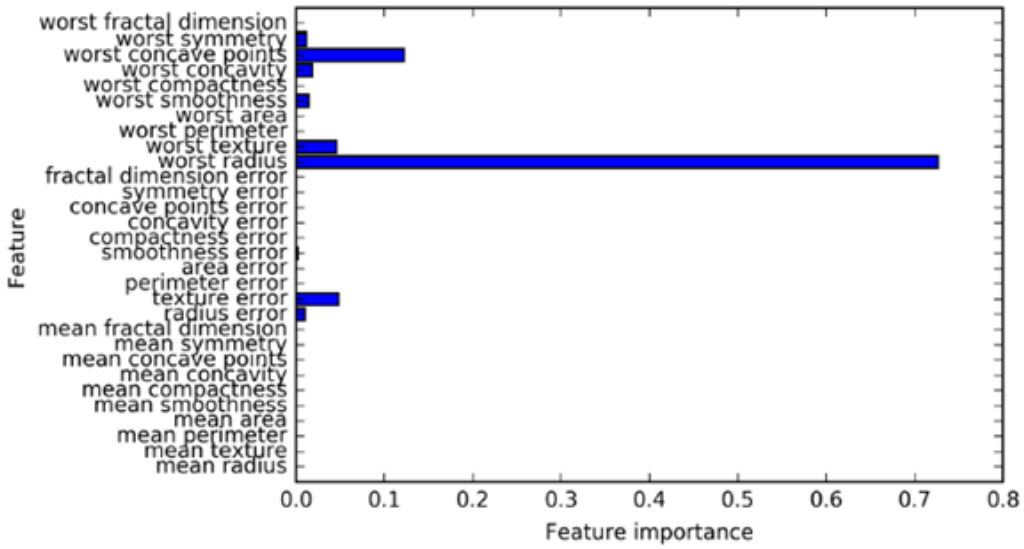
Use the following settings:

- figsize=(7,7), dpi=100
- Plot colors and markers of your choice

The github website of the prescribed textbook has quite some useful supplemental material (**code examples**, IPython notebooks, etc.), available at https://github.com/amueller/introduction_to_ml_with_python, especially chapter 2.

Sample output as shown in the following figure is **for demonstration purposes only as they are plots for different datasets, which just show you what your plots might look like.**





Submission:

Submit the following files to OnTrack:

1. Your program source code (e.g. task6_2.py)
2. A screen shot of your program running

Check the following things before submitting:

1. Add proper comments to your code