1_Logistic_Regression_with_sklearn

May 27, 2019

```
In [ ]: import pandas as pd

In [ ]: data_url= "./uva.txt"

        dataframe = pd.read_table(data_url)

In [ ]: dataframe[:5]

In [ ]: dataframe.shape

In [ ]: dataframe.dtypes

In [ ]: dataframe["Marital Status"].unique()

In [ ]: dataframe.pop('who') # dataframe = dataframe.ix[:, 1:]

        dataframe.dtypes

In [ ]: dataframe.pop('Country')

        dataframe.dtypes

In [ ]: dataframe.pop('Years on Internet')

        dataframe.dtypes

In [ ]: dataframe["Gender"] = dataframe["Gender"].astype("category")
        dataframe.dtypes

In [ ]: for col in ['Household Income', 'Sexual Preference', 'Education Attainment', 'Major Oc
            dataframe[col] = dataframe[col].astype('category')
        dataframe.dtypes

In [ ]: dataframe.head()

In [ ]: dataframe_modified = pd.get_dummies(dataframe)
        dataframe_modified[:5]

In [ ]: dataframe_modified.shape
In [ ]: dataframe_modified.isnull().sum()

In [ ]: pd.isnull(dataframe_modified['Age']).head()

In [ ]: dataframe_modified[pd.isnull(dataframe_modified['Age'])].head()

In [ ]: dataframe_modified.loc[pd.isnull(dataframe_modified['Age'])].head()

In [ ]: dataframe_modified.loc[pd.isnull(dataframe_modified['Age'])].head().index

In [ ]: dataframe_modified.loc[pd.isnull(dataframe_modified['Age']), "Age"] = dataframe_modifi

In [ ]: dataframe_modified['Age'][pd.isnull(dataframe_modified['Age'])]

In [ ]: dataframe_modified.head()

In [ ]: dataframe_modified.info()

In [ ]: dataframe_modified['Age'] = dataframe_modified['Age'].astype('int')
        dataframe_modified.info()

In [ ]: x_data = dataframe_modified.iloc[:, 1:].as_matrix()
        y_data = dataframe_modified.iloc[:, 0].as_matrix().reshape(-1, 1)
        y_data.shape, x_data.shape

In [ ]: from sklearn import preprocessing # Min-Max Standardzation

        min_max_scaler = preprocessing.MinMaxScaler()

        x_data = min_max_scaler.fit_transform(x_data)
        x_data[:5]

In [ ]: y_data.shape

In [ ]: # list(pd.Series(y_data[:, 0]).index)

In [ ]: import numpy as np

        training_idx = np.random.randint(y_data.shape[0], size=int(y_data.shape[0] * 0.8))
        test_idx = np.random.randint(y_data.shape[0], size=int(y_data.shape[0] * 0.2))

        x_training, x_test = x_data[training_idx,:], x_data[test_idx,:]
        y_training, y_test = y_data[training_idx,:], y_data[test_idx,:]

        x_training.shape, x_test.shape

In [ ]: from sklearn import linear_model, datasets

        logreg = linear_model.LogisticRegression(fit_intercept=True)
        logreg.fit(x_training, y_training.flatten())
```

```
In [ ]: logreg.predict(x_test[:10])

In [ ]: logreg.predict_proba(x_test[:10])

In [ ]: x_test.shape

In [ ]: y_test.shape

In [ ]: logreg.predict(x_test) == y_test.flatten()

In [ ]: sum(logreg.predict(x_test) == y_test.flatten()) / len(y_test)

In [ ]: logreg.predict_proba(x_test[:10])

In [ ]: logreg.decision_function(x_test[:10])

In [ ]: logreg.predict(x_test[:10])

In [ ]:
```