Assignment 1: (individual): Deadline Monday 1/28/2019 11:59pm

"The data set consists of customer default records for a credit card company. We are interested in predicting whether an individual will default on his credit card payment, on the basis of annual income and monthly credit card balance"

Setup

```
# To support both python 2 and python 3
from __future__ import division, print_function, unicode_literals

# Common imports
import numpy as np
import os

# to make this notebook's output stable across runs
np.random.seed(42)

# To plot pretty figures
%matplotlib inline
import matplotlib
import matplotlib.pyplot as plt
plt.rcParams['axes.labelsize'] = 14
plt.rcParams['xtick.labelsize'] = 12
plt.rcParams['ytick.labelsize'] = 12

# Where to save the figures
PROJECT_ROOT_DIR = "."
CHAPTER_ID = "Assignment2"

def save_fig(fig_id, tight_layout=True):
    path = os.path.join(PROJECT_ROOT_DIR, "images", CHAPTER_ID, fig_id + ".png")
    print("Saving figure", fig_id)
    if tight_layout:
        plt.tight_layout()
    plt.savefig(path, format='png', dpi=300)

# Ignore useless warnings (see SciPy issue #5998)
import warnings
warnings.filterwarnings(action="ignore", module="scipy", message="^internal
gelsd")
```

Load and prepare data

```
import pandas as pd
DATA_PATH = os.path.join("datasets", "data_assignments")
def load_data(data_path=DATA_PATH):
csv_path = os.path.join(data_path, "Default.csv")
return pd.read_csv(csv_path)

data = load_data()
data.head()
```

	recondNum	default	student	balance	income
0	1	No	No	729.526495	44361.62507
1	2	No	Yes	817.180407	12106.13470
2	3	No	No	1073.549164	31767.13895
3	4	No	No	529.250605	35704.49394
4	5	No	No	785.655883	38463.49588

```
data = data.drop("recondNum",axis=1)
data.head()
```

1. Split the data into training and test subsets with test size 25%
2. Choose Logistic Regression Model
3. Train the Logistic Regression Model on training data (show the accuracy on training data)
4. Predict on the testing data (show the accuracy on testing data)