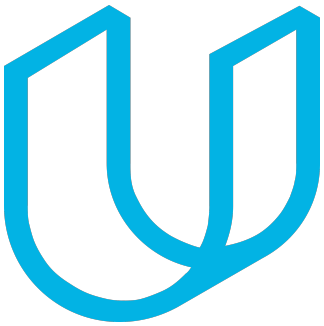


UDACITY

Machine Learning Engineer Nanodegree

REPORTE NO. 01

Autor:
Díaz Medina Jesús Kaimorts
Escuela:
Escuela Superior de Cómputo - Instituto Politécnico Nacional

29 de abril del 2019

Índice

1. Lesson 3: Introductory Practice Project

1.1. Titanic Survival Exploration

Description: In this practice project, I start to create decision **functions that attempt to predict survival outcomes from the 1912 Titanic disaster based on each passenger's features**, such as sex and age. Starting with a simple algorithm and increase its complexity until you are able to accurately predict the outcomes for at least 80 % of the passengers in the provided data.

In the first instance, it is necessary to install Python and its modules and packages that are required during the course, as they are:

1. Numpy: pip install –user numpy
2. Pandas: pip install –user pandas
3. Jupyter: pip install –user jupyter
4. Matplotlib: pip install –user matplotlib

Ïnstalación"

```
pip install --user numpy pandas matplotlib jupyter
```

For this assignment, we can find the **titanic_survival_exploration** folder containing the necessary project files on the Machine Learning projects GitHub.

This project contains three main files:

1. **titanic_survival_exploration.ipynb**: This is the main file where you will be performing your work on the project (in Jupyter Notebook.)
2. **titanic_data.csv**: The project dataset. We'll load this data in the notebook (Jupyter Notebook).
3. **visuals.py**: This Python script provides supplementary visualizations for the project. Do not modify.

In the Terminal or Command Prompt, navigate to the folder containing the project files, and then use the command (Figura 1) to open up a browser window or tab to work with your notebook.

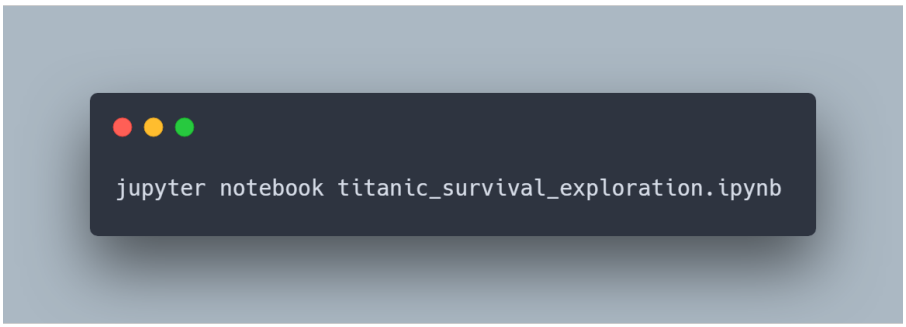


Figura 1: Open the work station in Jupyter Notebook.

1.2. Development

To begin working with the RMS Titanic passenger data, we'll first need to *import* the functionality we need, and load our data into a **pandas** DataFrame.

"Load of DataFrame"

```
# Import libraries necessary for this project
import numpy as np
import pandas as pd
from IPython.display import display # Allows the use of display() for DataFrames

# Import supplementary visualizations code visuals.py
import visuals as vs

# Pretty display for notebooks
%matplotlib inline

# Load the dataset
in_file = 'titanic_data.csv'
full_data = pd.read_csv(in_file)

# Print the first few entries of the RMS Titanic data
display(full_data.head())
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

Figura 2: DataFrame shown with a few values.

We're interested in the outcome of survival for each passenger or crew member, **we can remove the Survived feature from this dataset and store it as its own separate variable outcomes.**

"DataFrame without 'Survived' field"

```
# Store the 'Survived' feature in a new variable and remove it from the dataset
outcomes = full_data['Survived']
data = full_data.drop('Survived', axis = 1)

# Show the new dataset with 'Survived' removed
display(data.head())
```

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

Figura 3: DataFrame shown with a few values.

To measure the performance of our predictions, **we need a metric to score our predictions against the true outcomes of survival**. Since we are interested in how accurate our predictions are, we will ***calculate the proportion of passengers where our prediction of their survival is correct***.

"Test a prediction on the first five passengers."

```
def accuracy_score(truth, pred):
    """ Returns accuracy score for input truth and predictions. """

    # Ensure that the number of predictions matches number of outcomes
    if len(truth) == len(pred):

        # Calculate and return the accuracy as a percent
        return "Predictions have an accuracy of {:.2f}%.".format(
                                                    (truth == pred).mean()*100
                                                    )

    else:
        return "Number of predictions does not match number of outcomes!"

# Test the 'accuracy_score' function
# Generate an numpy array with 1's: [1,1,1,1,1] Nobody dead.
predictions = pd.Series(np.ones(5, dtype = int))

"""
print("Prediction:\n{}".format(predictions))
print("Outcomes:\n{}".format(outcomes[:5]))
"""

print(accuracy_score(outcomes[:5], predictions))
```

Predictions have an accuracy of 60.00 %.

If we were asked to make a prediction about any passenger aboard the RMS Titanic whom we knew nothing about, then the best prediction we could make would be that they did not survive: how accurate would a prediction be that none of the passengers survived?, well we can use this code to calculate it.

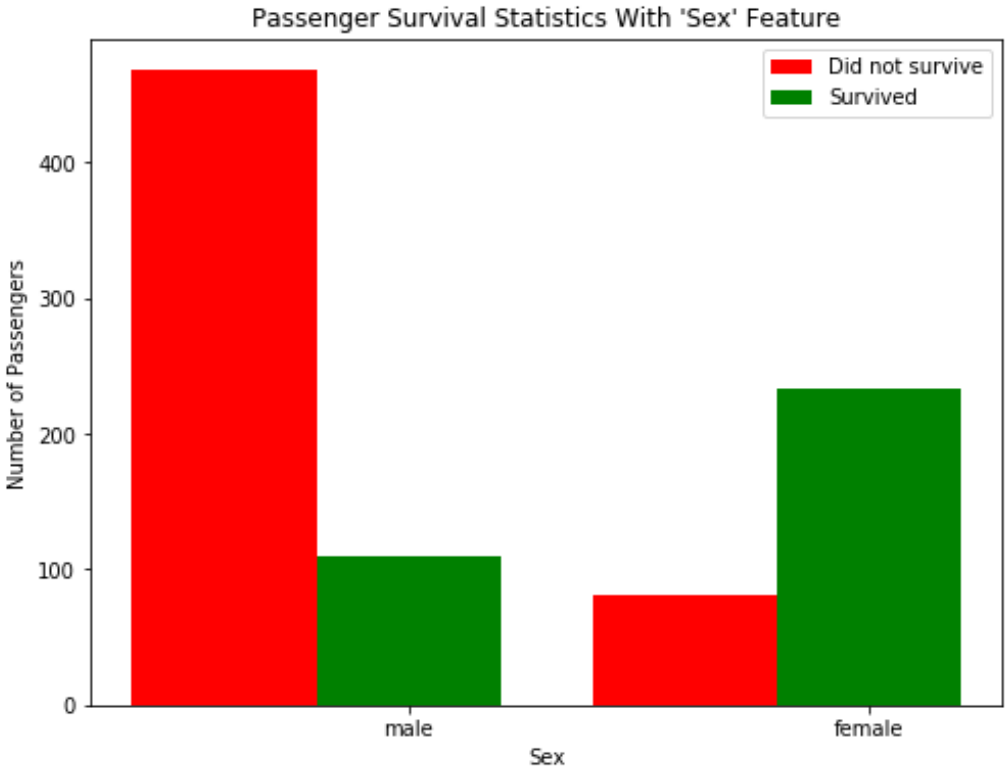
"Percentage of people killed during the Titanic disaster."

```
def predictions_0(data):
    """ Model with no features. Always predicts a passenger did not survive. """

    predictions = []
    for _, passenger in data.iterrows():

        # Predict the survival of 'passenger'
        predictions.append(0)

    # Return our predictions
    return pd.Series(predictions)

# Make the predictions
predictions = predictions_0(data)
#print(predictions)
print(accuracy_score(outcomes, predictions))
```

Predictions have an accuracy of 61.62 %.

Let's take a look at whether the feature Sex has any indication of survival rates among passengers.

```
vs.survival_stats(data, outcomes, 'Sex')
```



1.3. Process

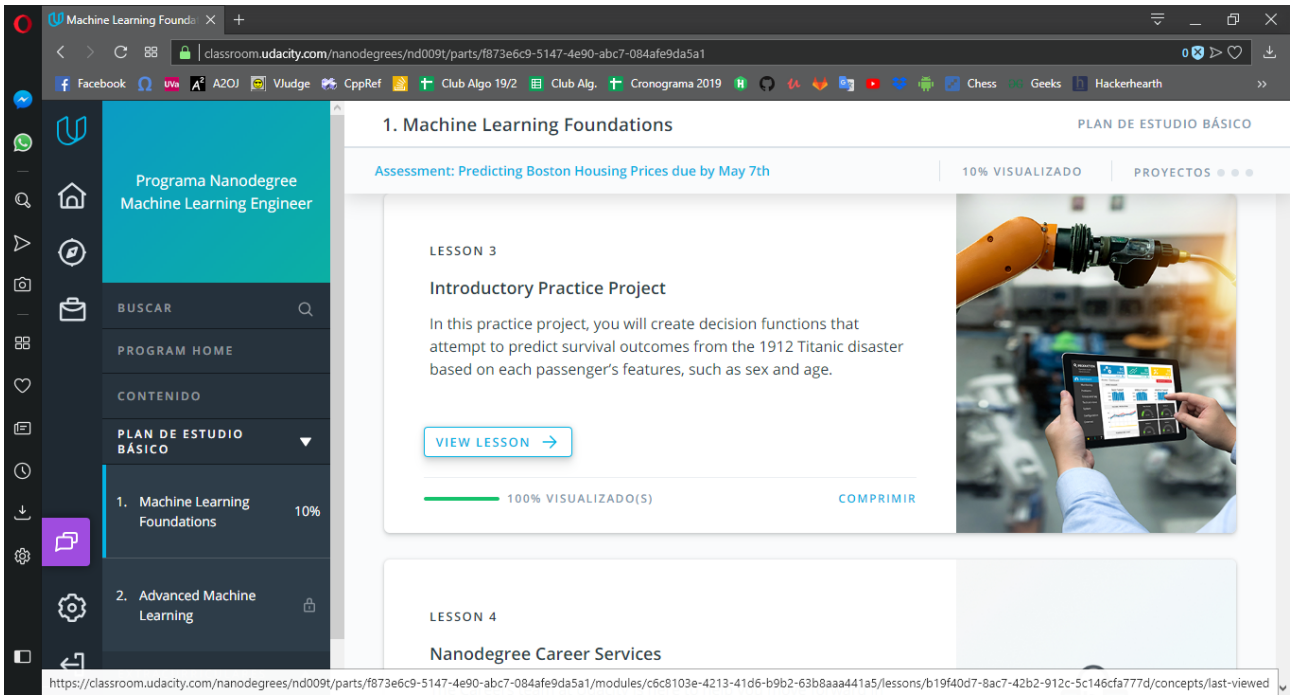


Figura 4: Current process at April 29th, 2019