Final Projects for Data Science with Python - Course 1

***The projects are rated by \* , \*\* or \*\*\* corresponding to its difficulty.***

**Project 1: Predicting Boston Housing Prices (2 students) \*\***

**1.1 Project Overview**

In this project, you will apply basic machine learning concepts on data collected for housing prices in the Boston, Massachusetts area to predict the selling price of a new home. You will first explore the data to obtain important features and descriptive statistics about the dataset. Next, you will properly split the data into testing and training subsets, and determine a suitable performance metric for this problem. You will then analyze performance graphs for a learning algorithm with varying parameters and training set sizes. This will enable you to pick the optimal model that best generalizes for unseen data. Finally, you will test this optimal model on a new sample and compare the predicted selling price to your statistics.

**1.2 Project Highlights**

This project is designed to get you acquainted to working with datasets in Python and applying basic machine learning techniques using NumPy and Scikit-Learn. Before being expected to use many of the available algorithms in the sklearn library, it will be helpful to first practice analyzing and interpreting the performance of your model.

Things you will learn by completing this project:

- How to use NumPy to investigate the latent features of a dataset.

- How to analyze various learning performance plots for variance and bias.

- How to determine the best-guess model for predictions from unseen data.

- How to evaluate a model's performance on unseen data using previous data.

**1.3 Description**

The Boston housing market is highly competitive, and you want to be the best real estate agent in the area. To compete with your peers, you decide to leverage a few basic machine learning concepts to assist you and a client with finding the best selling price for their home. Luckily, you've come across the Boston Housing dataset which contains aggregated data on various features for houses in Greater Boston communities, including the median value of homes for each of those areas. Your task is to build an optimal model based on a statistical analysis with the tools available. This model will then be used to estimate the best selling price for your clients' homes.

**1.4 Data**

The modified Boston housing dataset consists of 489 data points, with each datapoint having 3 features. This dataset is a modified version of the Boston Housing dataset found on the [UCI Machine Learning Repository](https://archive.ics.uci.edu/ml/datasets/Housing).

\*\*Features\*\*

1. `RM`: average number of rooms per dwelling

2. `LSTAT`: percentage of population considered lower status

3. `PTRATIO`: pupil-teacher ratio by town

\*\*Target Variable\*\*

4. `MEDV`: median value of owner-occupied homes

**Dataset link** [**https://drive.google.com/open?id=14igQCG4GrblzEUUYDGbt5F-ViKTRXd\_e**](https://drive.google.com/open?id=14igQCG4GrblzEUUYDGbt5F-ViKTRXd_e)

**Project 2: Creating Customer Segments (2 students) \*\***

**2.1 Project Overview**

In this project you will apply unsupervised learning techniques on product spending data collected for customers of a wholesale distributor in Lisbon, Portugal to identify customer segments hidden in the data. You will first explore the data by selecting a small subset to sample and determine if any product categories highly correlate with one another. Afterwards, you will preprocess the data by scaling each product category and then identifying (and removing) unwanted outliers. With the good, clean customer spending data, you will apply PCA transformations to the data and implement clustering algorithms to segment the transformed customer data. Finally, you will compare the segmentation found with an additional labeling and consider ways this information could assist the wholesale distributor with future service changes.

**2.2 Project Highlights**

This project is designed to give you a hands-on experience with unsupervised learning and work towards developing conclusions for a potential client on a real-world dataset. Many companies today collect vast amounts of data on customers and clientele, and have a strong desire to understand the meaningful relationships hidden in their customer base. Being equipped with this information can assist a company engineer future products and services that best satisfy the demands or needs of their customers.

Things you will learn by completing this project:

- How to apply preprocessing techniques such as feature scaling and outlier detection.

- How to interpret data points that have been scaled, transformed, or reduced from PCA.

- How to analyze PCA dimensions and construct a new feature space.

- How to optimally cluster a set of data to find hidden patterns in a dataset.

- How to assess information given by cluster data and use it in a meaningful way.

**2.3 Description**

A wholesale distributor recently tested a change to their delivery method for some customers, by moving from a morning delivery service five days a week to a cheaper evening delivery service three days a week. Initial testing did not discover any significant unsatisfactory results, so they implemented the cheaper option for all customers. Almost immediately, the distributor began getting complaints about the delivery service change and customers were canceling deliveries, losing the distributor more money than what was being saved. You've been hired by the wholesale distributor to find what types of customers they have to help them make better, more informed business decisions in the future. Your task is to use unsupervised learning techniques to see if any similarities exist between customers, and how to best segment customers into distinct categories.

**2.4 Data**

The customer segments data is included as a selection of 440 data points collected on data found from clients of a wholesale distributor in Lisbon, Portugal. More information can be found on the [UCI Machine Learning Repository](https://archive.ics.uci.edu/ml/datasets/Wholesale+customers).

Note (m.u.) is shorthand for \*monetary units\*.

\*\*Features\*\*

1) `Fresh`: annual spending (m.u.) on fresh products (Continuous);

2) `Milk`: annual spending (m.u.) on milk products (Continuous);

3) `Grocery`: annual spending (m.u.) on grocery products (Continuous);

4) `Frozen`: annual spending (m.u.) on frozen products (Continuous);

5) `Detergents\_Paper`: annual spending (m.u.) on detergents and paper products (Continuous);

6) `Delicatessen`: annual spending (m.u.) on and delicatessen products (Continuous);

7) `Channel`: {Hotel/Restaurant/Cafe - 1, Retail - 2} (Nominal)

8) `Region`: {Lisbon - 1, Oporto - 2, or Other - 3} (Nominal)

link of dataset: <https://drive.google.com/file/d/1F_VpOT7WZU3uGb4SnUxfBwS8GDXEMiz2/view?usp=sharing>

**Project 3: Finding Donors for CharityML (2 students) \*\***

**3.1 Project Overview**

In this project, you will apply supervised learning techniques and an analytical mind on data collected for the U.S. census to help CharityML (a fictitious charity organization) identify people most likely to donate to their cause. You will first explore the data to learn how the census data is recorded. Next, you will apply a series of transformations and preprocessing techniques to manipulate the data into a workable format. You will then evaluate several supervised learners of your choice on the data, and consider which is best suited for the solution. Afterwards, you will optimize the model you've selected and present it as your solution to CharityML. Finally, you will explore the chosen model and its predictions under the hood, to see just how well it's performing when considering the data it's given.

**3.2 Project Highlights**

This project is designed to get you acquainted with the many supervised learning algorithms available in sklearn, and to also provide for a method of evaluating just how each model works and performs on a certain type of data. It is important in machine learning to understand exactly when and where a certain algorithm should be used, and when one should be avoided.

Things you will learn by completing this project:

- How to identify when preprocessing is needed, and how to apply it.

- How to establish a benchmark for a solution to the problem.

- What each of several supervised learning algorithms accomplishes given a specific dataset.

- How to investigate whether a candidate solution model is adequate for the problem.

**3.3 Description**

In this project, you will employ several supervised algorithms of your choice to accurately model individuals' income using data collected from the 1994 U.S. Census. You will then choose the best candidate algorithm from preliminary results and further optimize this algorithm to best model the data. Your goal with this implementation is to construct a model that accurately predicts whether an individual makes more than $50,000. This sort of task can arise in a non-profit setting, where organizations survive on donations. Understanding an individual's income can help a non-profit better understand how large of a donation to request, or whether or not they should reach out to begin with. While it can be difficult to determine an individual's general income bracket directly from public sources, we can (as we will see) infer this value from other publically available features.

**3.4 Data**

The modified census dataset consists of approximately 32,000 data points, with each datapoint having 13 features. This dataset is a modified version of the dataset published in the paper \*"Scaling Up the Accuracy of Naive-Bayes Classifiers: a Decision-Tree Hybrid",\* by Ron Kohavi. You may find this paper [online](https://www.aaai.org/Papers/KDD/1996/KDD96-033.pdf), with the original dataset hosted on [UCI](https://archive.ics.uci.edu/ml/datasets/Census+Income).

\*\*Features\*\*

- `age`: Age

- `workclass`: Working Class (Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked)

- `education\_level`: Level of Education (Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool)

- `education-num`: Number of educational years completed

- `marital-status`: Marital status (Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse)

- `occupation`: Work Occupation (Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces)

- `relationship`: Relationship Status (Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried)

- `race`: Race (White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black)

- `sex`: Sex (Female, Male)

- `capital-gain`: Monetary Capital Gains

- `capital-loss`: Monetary Capital Losses

- `hours-per-week`: Average Hours Per Week Worked

- `native-country`: Native Country (United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands)

\*\*Target Variable\*\*

- `income`: Income Class (<=50K, >50K)

Link: <https://drive.google.com/file/d/1L0Dwd4ERW9r-Qnweuu1bcWyd8F-1et1M/view?usp=sharing>

**Context of project 4 and 5**

Collaborative filtering and content-based filtering are two kinds of *recommender systems* that provide users with information to help them find and choose anything from books, to movies, to restaurants, to courses based on their own preferences compared to the preferences of others.



In 2009 Netflix awarded one million dollars to a group that had developed a better-recommender system than the Netflix, in-house system. This [NY Times Magazine](http://www.nytimes.com/2008/11/23/magazine/23Netflix-t.html?pagewanted=all) article describes the competition, the winning teams, and how the movie *Napolean Dynamite* caused problems for the algorithms and ranking/rating systems developed by contest participants. You can watch this short YouTube video about the prize and contest: <https://www.youtube.com/watch?v=ImpV70uLxyw>

**Project 4: User clustering (2 students)\*\***

**Description**

The most important key of collaborative filtering is to find a set of similar users. There are many ways and based on different features to cluster users into a set. For this project you have a"[movies.txt](https://drive.google.com/open?id=12DSuY7EW0bfx6WThD3Txy-6slRhgGZ2q)" file that contains 22,930 lines in the format shown below (these are the first three and last three lines)

**user1367,Star Trek Beyond,3**

**user1367,Rogue One,3**

**user1367,Moana,1**

**…**

**user1460,Pirates of the Caribbean: On Stranger Tides,1**

**user1460,The Dark Knight,5**

**user1460,Avatar,5**

In general the format is that on each line the file has: "student-id/name",Movie,Rating

**Requirements**

Your mission in this project are:

1. Load this data file into a matrix (called rating matrix) in which: row is user list, column is the movie list and value of matrix is the value that user rates to movie.
2. Calculate distance matrix among users to measure how they are similar. The distance of 2 users is computed by sum of multiplication of 2 correspondent rows. Example: user1 and user2 have correspondent rating vector like: u1 = [0, 2, 5, -3, 0, -2, 0] and u2 = [3, 2, 0, -4, 3, 3, 1] then d = 0\*3 + 2\*2 + 5\*0 + (-3) \*(-4) + 0\*3 + (-2)\*3 + 0 \* 1 = 10.
3. Visualize the distance matrix to have a first recognition of similar user.
4. Using the distance matrix to cluster user into similar group.

**Project 5: Topic classification (2 students)\*\***

**Description**

The second method of a recommender system is content-based where similar items is recommended to those like it. In this project, you have a [list of topics](https://drive.google.com/open?id=1F2bEibuH_HxYlooZ23DJs97DBVqnw84g) here. Your need to classify them into different groups that will be used by recommender engine whenever it needs a similar topic. Detail requirements are:

1. Build a Term frequency–Inverse document frequency matrix that can measure how 2 topic are similar. Use [TfidfVectorizer](http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html) class for this purpose.
2. Visualize this matrix to see how topics are clustered.
3. Classify topics into different groups.

**Project 6: Clustering stock (2 students) \***

**Description**

Classifying stocks into clusters is an important task of any investor that helps to optimize risk portfolio management. In the real application, classifying the stocks is using many information such as: asset, brand, sale price, recent activities, etc.

However, in this project, we consider a simplest case where the stock is classified using only its sale price.

**Requirements**:

1. Download the nasdaq stock market data here: <https://assets.datacamp.com/production/course_3882/datasets/nasdaq-listings.csv>
2. Visualize this data to see the last sale of stocks.
3. Find the stock that has max sale by sector
4. Use a cluster/classify algorithm to put the stocks into different groups.
5. Show statistical description of each group

**Project 7: Wine Quality Classification (2 student) \*\***

**Data Set Information**

[The two datasets](https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/) are related to red and white variants of the Portuguese "Vinho Verde" wine. Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available (e.g. there is no data about grape types, wine brand, wine selling price, etc.) like this:

Input variables (based on physicochemical tests):

1 - fixed acidity   
2 - volatile acidity   
3 - citric acid   
4 - residual sugar   
5 - chlorides   
6 - free sulfur dioxide   
7 - total sulfur dioxide   
8 - density   
9 - pH   
10 - sulphates   
11 - alcohol

Output variable (based on sensory data):

12 - quality (score between 0 and 10)

These datasets can be viewed as classification or regression tasks. The classes are ordered and not balanced (e.g. there are munch more normal wines than excellent or poor ones). Outlier detection algorithms could be used to detect the few excellent or poor wines. Also, we are not sure if all input variables are relevant. So it could be interesting to test feature selection methods.

Requirements

1. Classification type of wine with 4800 first of lines;
2. Use the last 98 of lines to test.

**Project 8: Potential Bus stops (2 or 3 students) \*\*\***

See here: <https://drive.google.com/file/d/1bP4jxtLYBd-Dl30M8XEvDO_EA67Ou-vj/view>

Datasets: <https://drive.google.com/open?id=1csP3O_dv4nct8k814dGAbE_fYfAZ5sE4>

**Project 9: Simple fraud detection (2 or 3 students) \*\*\***

See here: <https://drive.google.com/file/d/1ALMuxzQz7kbr7EkELpQcTbWeMxLs88ai/view>

Datasets: <https://drive.google.com/file/d/1a04XKxiMfPCKfX5eSsCpWwxBwSC3dYYn/view?usp=sharing>