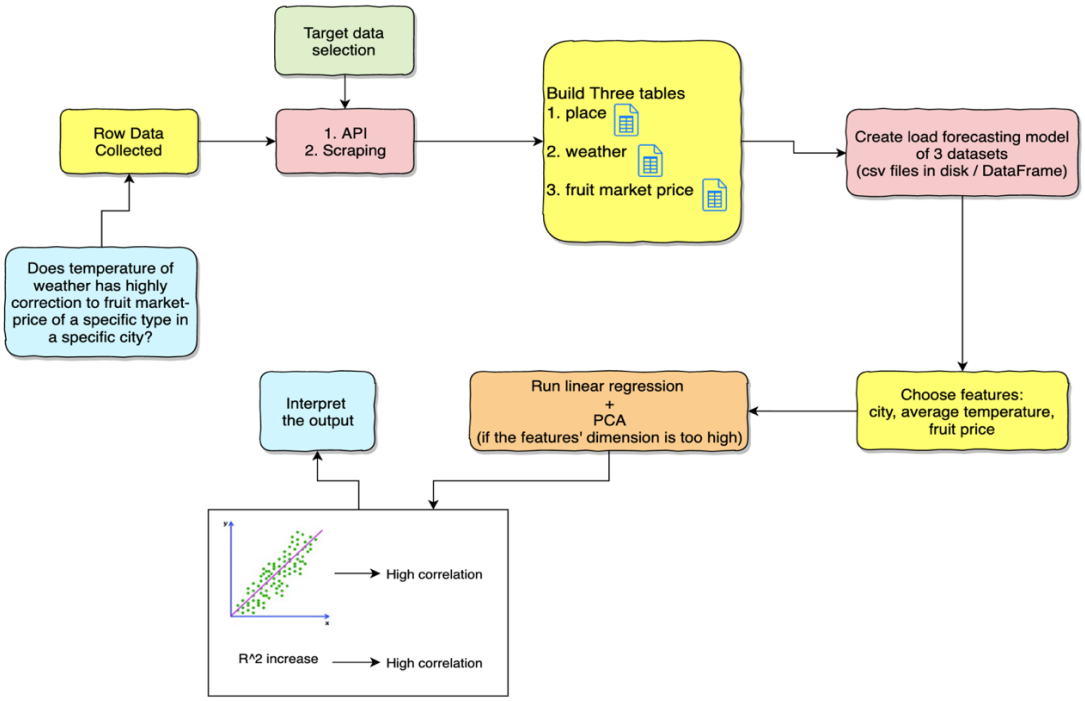
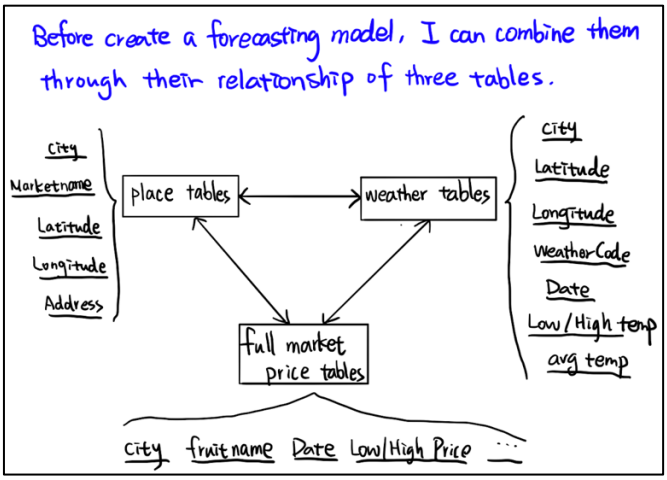
HW4 Documentation (INF510)

Kuan-Hui Lin

Section 1: Motivation (Information about how you want to use your data and what you want to achieve with this data)

I would like to know whether or not the fruit market price of different kinds of fruit has high correlation to the weather in different city. For example, my assumption is that the tropical fruits that are abundant in tropical cities should be cheap. For my research, I need to the datasets which contains the information of some attributes, like weather, city, fruit price and fruit type. Thus, I use these data to build the linear regression model, to analyze the output result and to find the answer of this problem.





Section 2: Data Source Info. (Information about the data sources i.e., their links, did you do scraping/API, etc...)

There are three data sources that I can get the related information.

First data source:

1. Weatherstack API endpoint:
   https://api.weatherstack.com/historical?access_key={YOUR_ACCESS_KEY}&query={city}&historical_date_start=2019-01-01&historical_date_end=2019-12-31
   API documentation: https://weatherstack.com/documentation
2. The output from Weatherstack API is json format which contains the data about city name, low/high temperature, average temperature, latitude and longitude of each day in past year/recent.

Second data source:

1. Place API endpoint: (need API KEY)
   https://maps.googleapis.com/maps/api/place/findplacefromtext/json?input=market%20LosAngeles&inputtype=textquery&fields=formatted_address,name,geometry&key={YOUR_ACCESS_KEY}.
2. API documentation: https://developers.google.com/places/web-service/intro?hl=en_US
3. The output from Place API is json format which contains some information about city name, market name, address, latitude and longitude of the specific place.

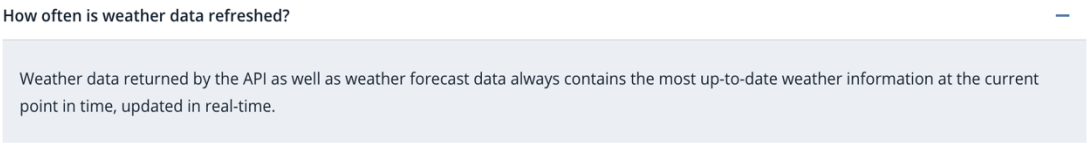
Third data source:

1. URL: https://www.freshfruitportal.com/usda-prices/

2. This is a scraped dataset. The data set on this website that includes the price of different kinds of fruit, name of fruit, location, high market price, low market price, date and origin.

Section 3: Frequency of data update of the data source (Approximately) (You are required to provide information as to how this data source is live and not static)

(1). First data source (Weatherstack API): Live data, update data on every day.



(2). Second data source (Place API from Google Maps Platform): The update frequency depends on what part of the world. In small, highly populated portions of the

continental United States, updates can happen as often as every week. For places more isolated, the frequency could be as slow as every couple of years or longer.

(3). Third data source (freshfruitportal.com): Live data, update data on every day.

Section 4: Procedure of extraction and description of your code.

1) Main flow of code

Start get_three_dataframe.py→ Call to __main__ → __main__ calling main_task() → main_task() calling multiple functions to get all fruit price and type data → after getting the data, main_task() calling write_to_csv() to store data in csv file. → __main__ will store fruit data in DataFrame form (we can obtain the first dataset) → __main__ keep calling get_weather_place_data() → get_weather_place_data() return two DataFrame, weather_df and place_df → All process are done.

2) How I get my data?

I. Fruit price data: First, I observe the parameter's change of url and I view page source code and then use re.compile for pattern matching. Before scraping, I create fake user agent header to disguise my python script as normal browser to access the website if the website has anti-reptile strategy until getting success status. Then, I prepare parameters, like dates, locations, fruit name, for scraping the content from freshfruitportal.com, and then there is "/" in the date, it will need to be coded and directly replaced. If the city name contains spaces, it needs to be converted to + in main_task(). After all tasks finished, it calls write_to_csv() to store the fruit price data in csv format in the disk. Final, In the __main__, fruit_price.csv will be stored by Dataframe format.

Note1: Because sometimes there exist some dates cannot be worked in freshfruitportal.com, if we choose 01/01/2019 and the date is broken, the page will jump directly to today's date. Thus, according to this situation, if the date has this problem, I use the data of today's date to replace the data on 01/01/2019.

Note2: It will take about 10 minutes to get all fruit price data because the total number of data is more than 60,000.

II. Place data and Weather data: These two datasets are obtained from external API, Google Place API and Weatherstack API. In the __main__, it calls get_weather_place_data() which is the function in weather_place_data.py. In get_weather_place_data(), there are two functions, grab_data_from_place_api()

which can be invoked to get place data and grab_data_from_weather_api() which can be invoked to get weather data. Each API has its endpoint and its key that we can use them to get the target data. In grab_data_from_weather_api(), I prepare the start date list(dateliat_start) and end date list(datelist_end) to be date parameter. In addition, the limit of query is 60 days, and each query result only show city name, latitude, longitude, weather code once, and the remaining temperature data is based on the date range. Inside the funtion, it calls create_date() which needs to input two arguments, start date(dateliat_start[a]) and end date(datelist_end[a]), a is an index of len(datelist), and it will return a date list which includes every day within the specified range of dates. In grab_data_from_place_api(), I also prepare the parameters, input, inputtype, fields and request url and get the specific data I want. The format of output query result of two API is JSON format. I create a dictionary to store query result of each API. At the end of each function, it will return weather DataFrame/place DataFrame and store these two datasets as csv files.

3) Where my data is stored finally?
   I use three Dataframes--fruit_price_df, place_df, weather_df--to store my data, and in Pandas, I also use ".to_csv" of Pandas and csv.writer() to store my data in disk as csv file--fruit_price.csv, place.csv and weather.csv.
4) Special procedure
   Before running my code, there are two packages which should be installed in your computer first.

   pip3 install fake_useragent

   pip3 install pandas

   Besides, there is the only one entry file, get_three_dataframe.py, and it will call all functions and then get all data. I also create the document(readme.txt) to list each .py file and briefly describe what they do in this project.

Section 5: Now, I only have weather data with latitude, longitude, temperature. If I have other data source that relates to my existing data, like typhoon data, earthquake data, fruit appearance, which may be other factors will influence fruit price. I can add these data to existing models as different columns. After expanding my dimension of the features, I can use them as predictors and get more the coefficient of the predictors. Through linear regression model, I can use regression to make predictions based on the values of the

predictors. In the process of analyzation, I just need to extract the corresponding rows and columns with their name to analyze their relationship with corresponding data.

1. earthquake data source: https://www.aerisweather.com/support/docs/api/reference/endpoints/earthquakes/#response
2. typhoon data source: https://www.meteomatics.com/en/api/request/
3. fruit appearance data: It can be gotten from freshfruitportal.com which already is used in this project.