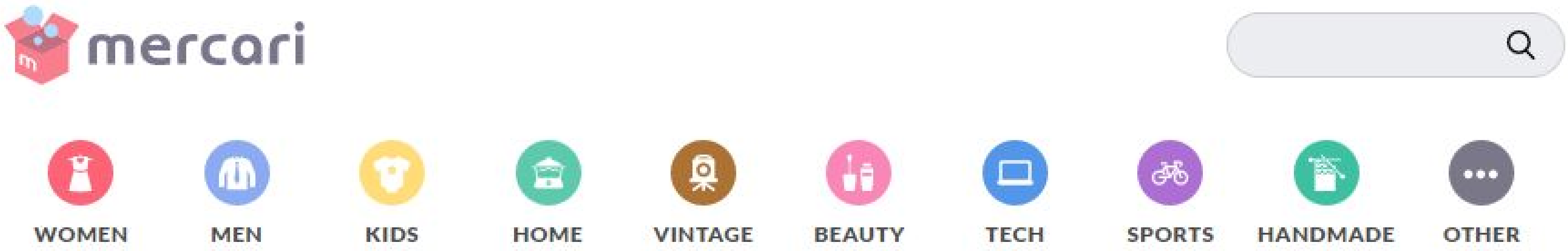


Mercari Price Suggestion Challenge

第一組 - Benchmark

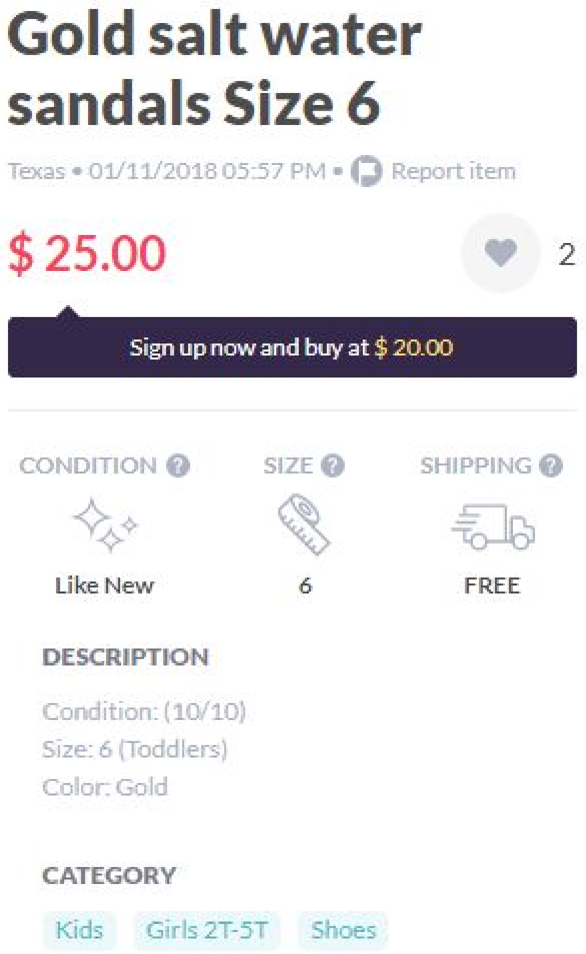
王選仲(法科碩二)、王韋勝(地政四)、張銘仁(資科三)、林建甫(資科三)、張為淳(資管五)

Project Introduction

題目說明

Dataset: Kaggle - Mercari Price Suggestion Challenge

Macari為一個網路二手交易平台，而這次的dataset便是其提供於kaggle上的資料，資料包含了商品名稱、商品狀態、貨運、拍賣者描述、商品類別等...
其中許多attributes都是文字，所以這次的project主要考驗的是如何處裡文字，已達到準確的價格預測。



Evaluation

Root Mean Squared Logarithmic Error

$$\epsilon = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\log(p_i + 1) - \log(a_i + 1))^2}$$

Where:

ϵ is the RMSLE value (score)
n is the total number of observations in the (public/private) data set,
p_i is your prediction of price, and
a_i is the actual sale price for i.
$\log(x)$ is the natural logarithm of x

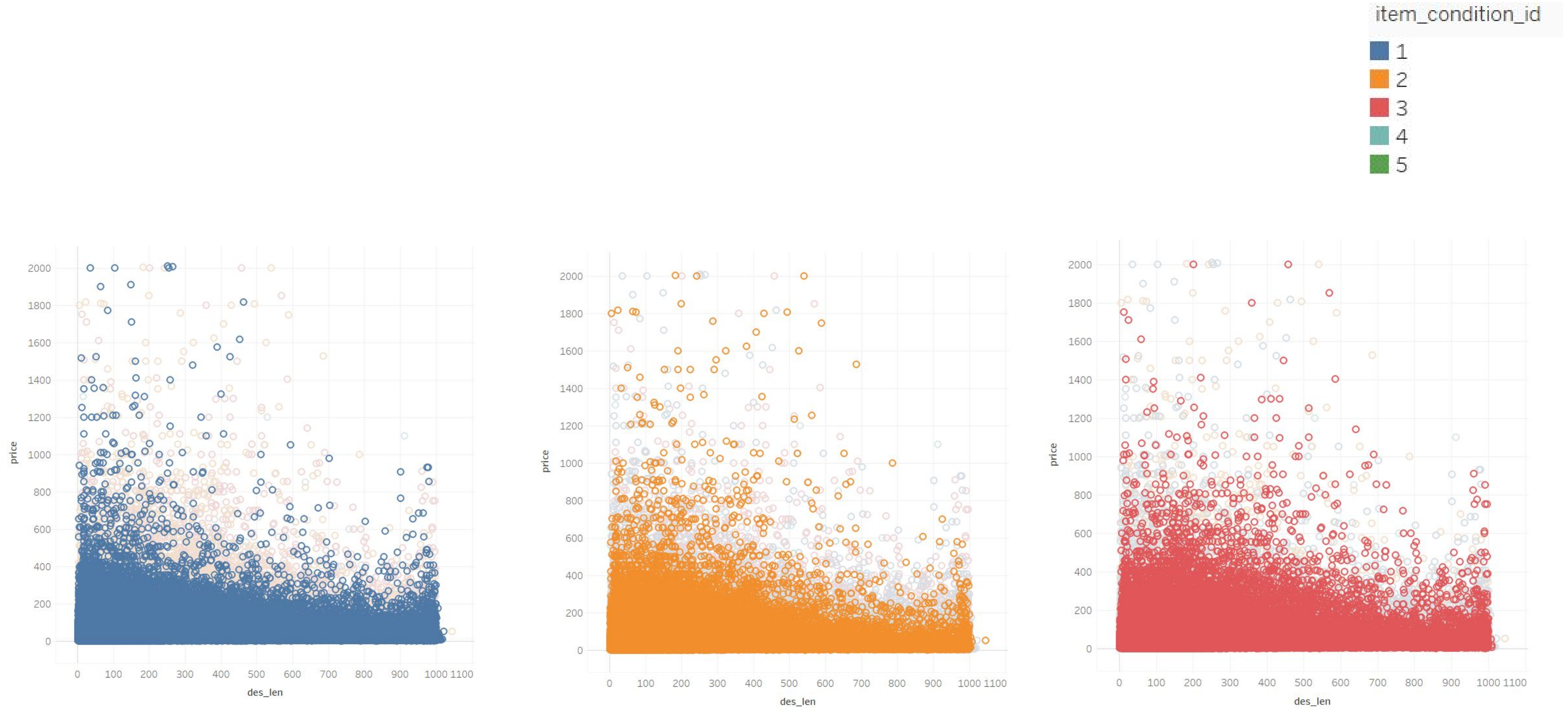
Features Overview

欄位概觀

資料中,欄位不多,只有八欄,基本上就是購物網站上會看到的那些資訊 ,如產品名稱、分類、品牌、產品新舊、價格、是否含運費及商品敘述 。產品新舊的部分是從 1-5分等級的;而產品分類欄位 內包括了三個分類由斜線分開 ;是否含運費的部分由 1代表價格含運費 ,0代表運費自付 。絕大部分的欄位都是文字,包括了產品名稱、分類、品牌及商品敘述 。資料中,training set跟testing set都只有分類及品牌有null值

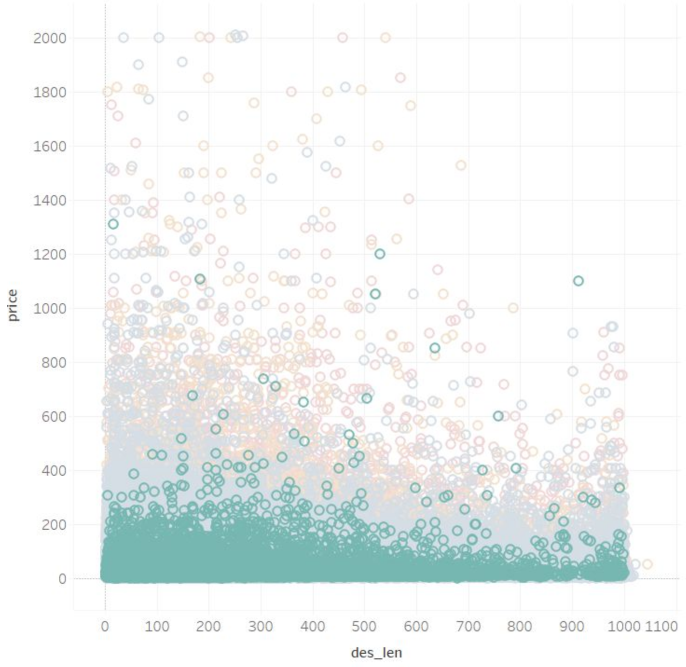
Column	id	name	Item_condition_id	Category_name	brand_name	price	shipping	Item_description
Type	string	string	int64	string	string	numeric	binary	string

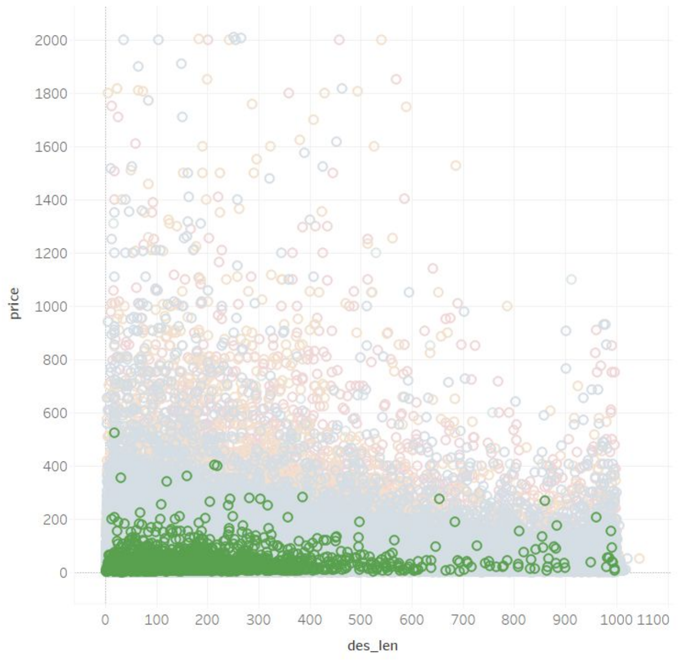
價錢與敘述長度Scatter plot



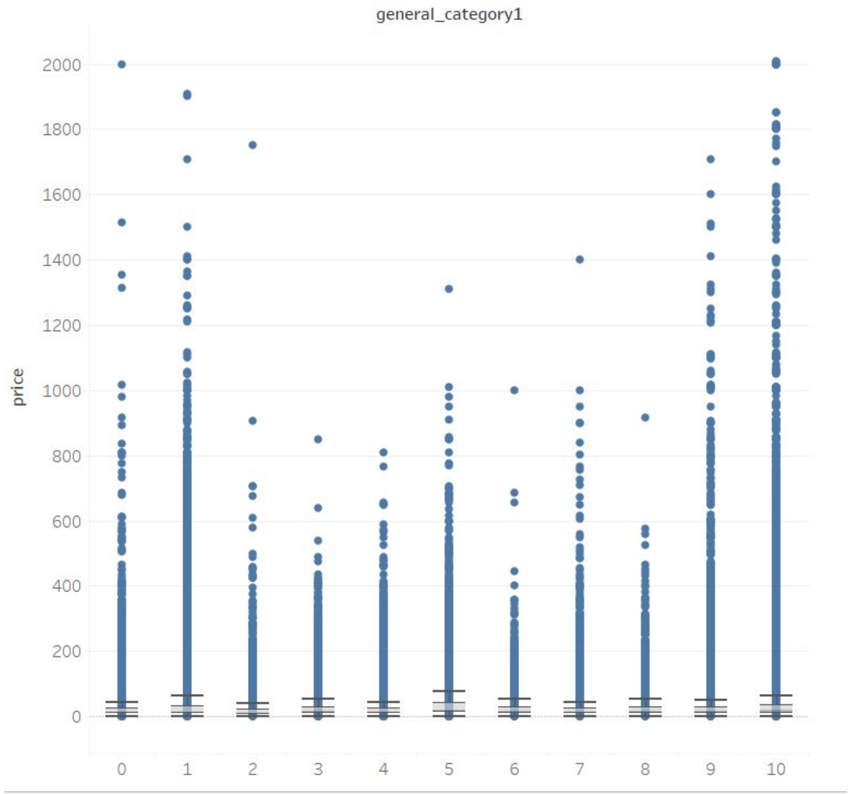
價錢與敘述長度Scatter plot

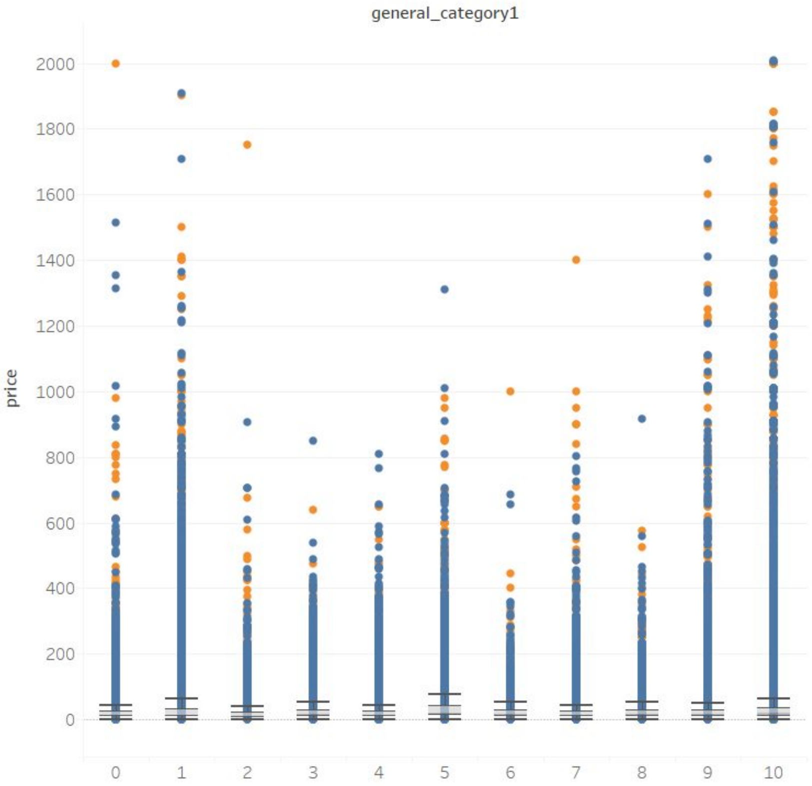
這裡可以看到,其實商品新舊1-3的商品價格分布差異並不大 ,只有新舊為4-5的商品明顯價格較低;而關於商品敘述的長度的話 ,基本上長度超過500之後的商品就不太出現高價品了





General category 價錢分布

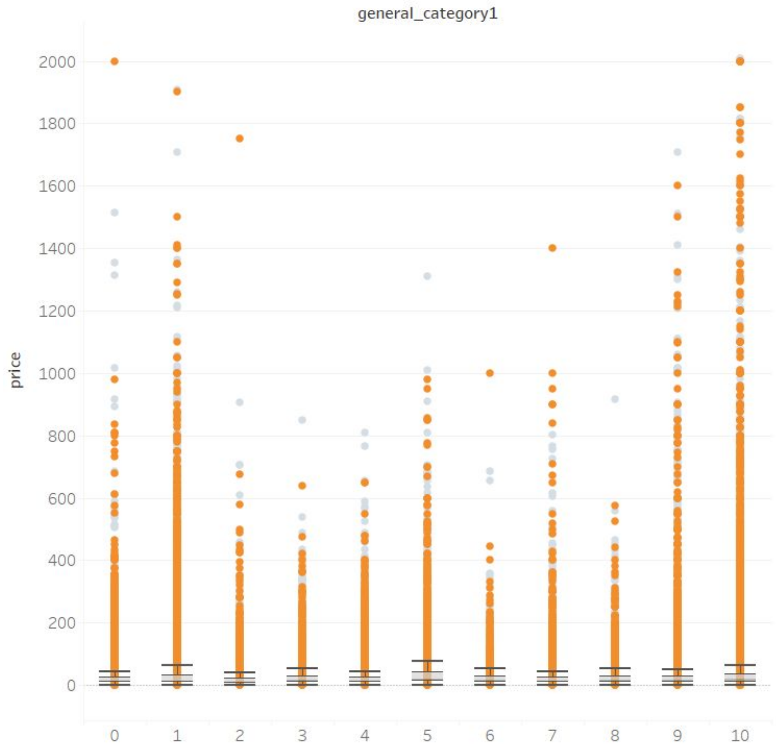




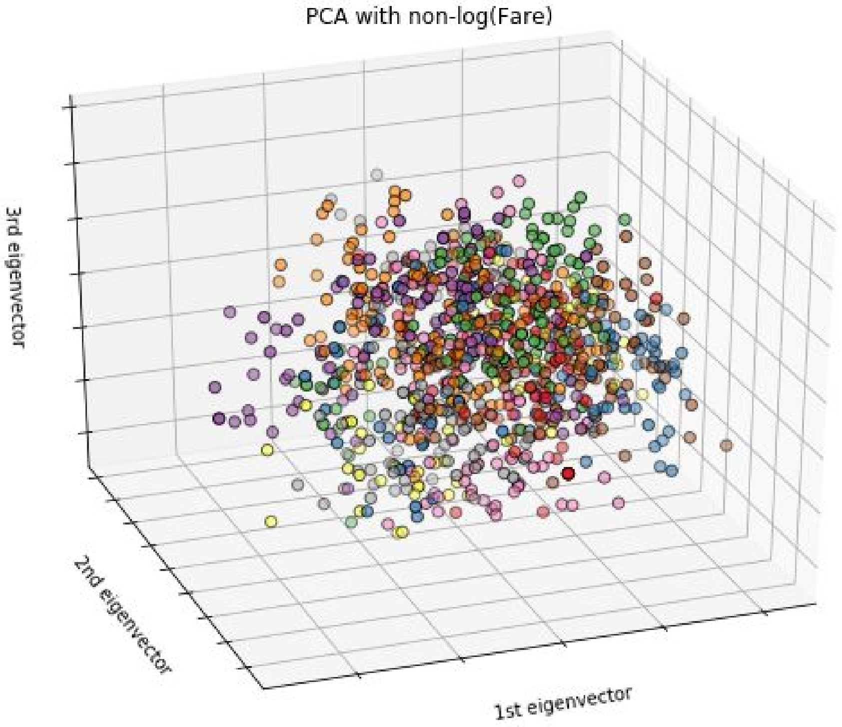
General category 價錢分布

0 : Beauty
1 : Eletronics
2 : Handmade
3 : Home
4 : Kids
5 : Men
6 : No category
7 : Others
8 : Sports & outdoors
9 : Vintage & Collectibles
10 : Women

以美妝、電子產品及古著還有女裝最多高價商品,但是主要分類的商品平均價格相差不多,而橘色為含運的商品,藍色為不含運費的商品,基本上並沒有除了上述的四種分類 ,其他的品項有含運的商品價格偏高



Category cluster 結果



分為30群

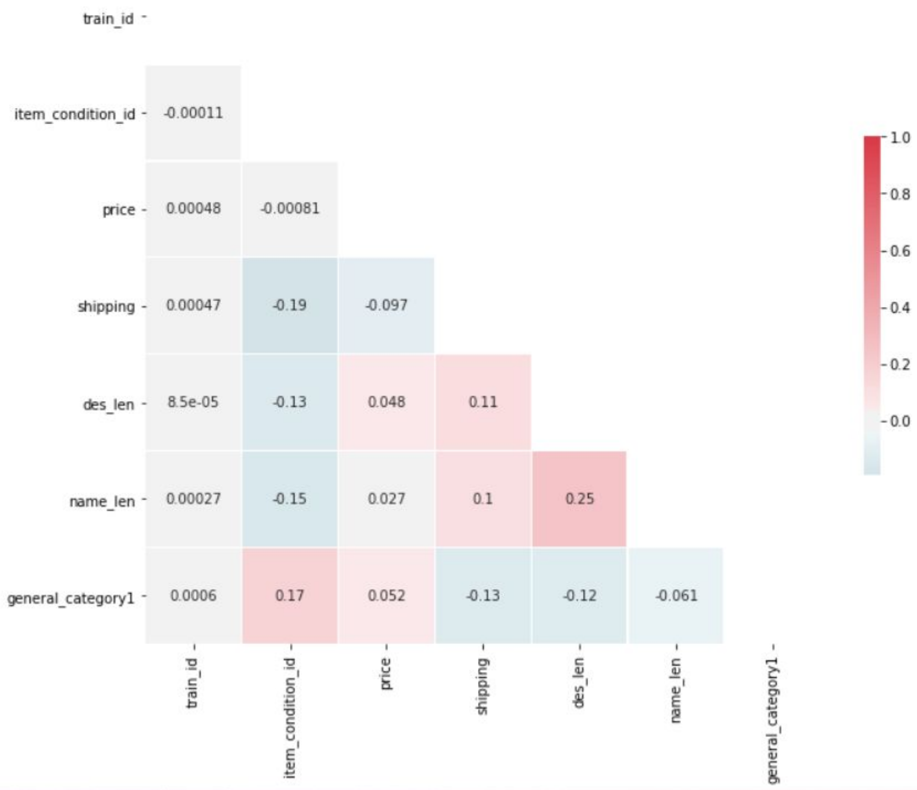
1	24	11	31	21	18
2	28	12	17	22	24
3	25	13	57	23	48
4	38	14	43	24	33
5	55	15	29	25	17
6	23	16	42	26	33
7	21	17	34	27	10
8	19	18	21	28	49
9	38	19	8	29	31
10	49	20	44	30	31

Category cluster 結果

其中個數比較少的群其實群 內的分類都十分相似 ,如第12群['Outdoors', 'Artwork', 'Posters & Prints', 'Painting', 'Paintings', 'Drawings', 'Magazines', 'Patterns', 'Sculptures', 'Magnets', 'Bookmark', 'Photographs', 'Postcard', 'Illustration', 'Frames', 'Collages', 'Portraits'] 及第19群 ['NFL', 'MLB', 'NCAA', 'NBA', 'Bowl', 'NHL', 'Pitcher', 'Draft Stoppers']

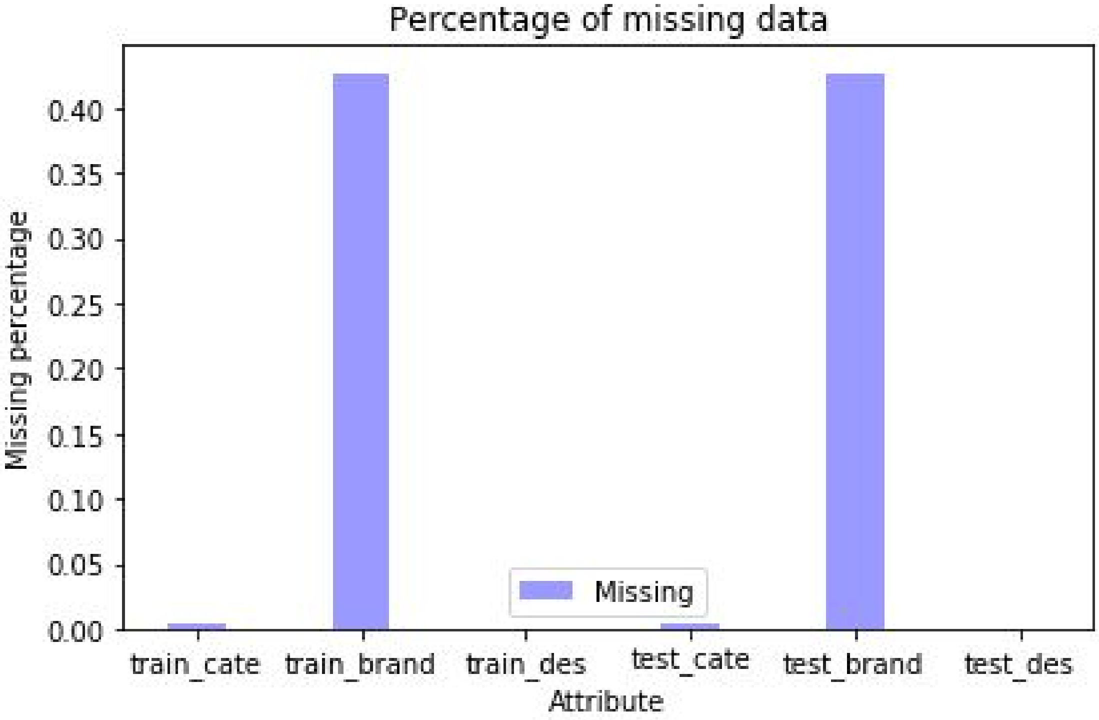
Correlation plot

這裡可以看到,General category與價格關係的正相關最大,而商品名稱及敘述長度為負相關，而有含運費的商品事實上價格並沒有比較高



Preprocessing

Missing Value



Missing Value

由於miss的data都是文字，故我們處裡的方法為將nan的欄位填上"missing"的字串

```
dataset.category_name.fillna(value="missing", inplace=True)
dataset.brand_name.fillna(value="missing", inplace=True)
dataset.item_description.fillna(value="missing", inplace=True)
```

LabelEncoding

doing labelencoding onto brand_name & category_name

unique category name: 1311

encode category to a number between [0, 1310]

unique brand name: 5290

encode brand to a number between [0, 5289]

Spliting category

A category name has multiple categories (e.g. Women/Beauty/Handmade)

1. Split this kind of category name into names

2. Labelencoding these names again

One_hot encoding

Target: category_name, item_description, name

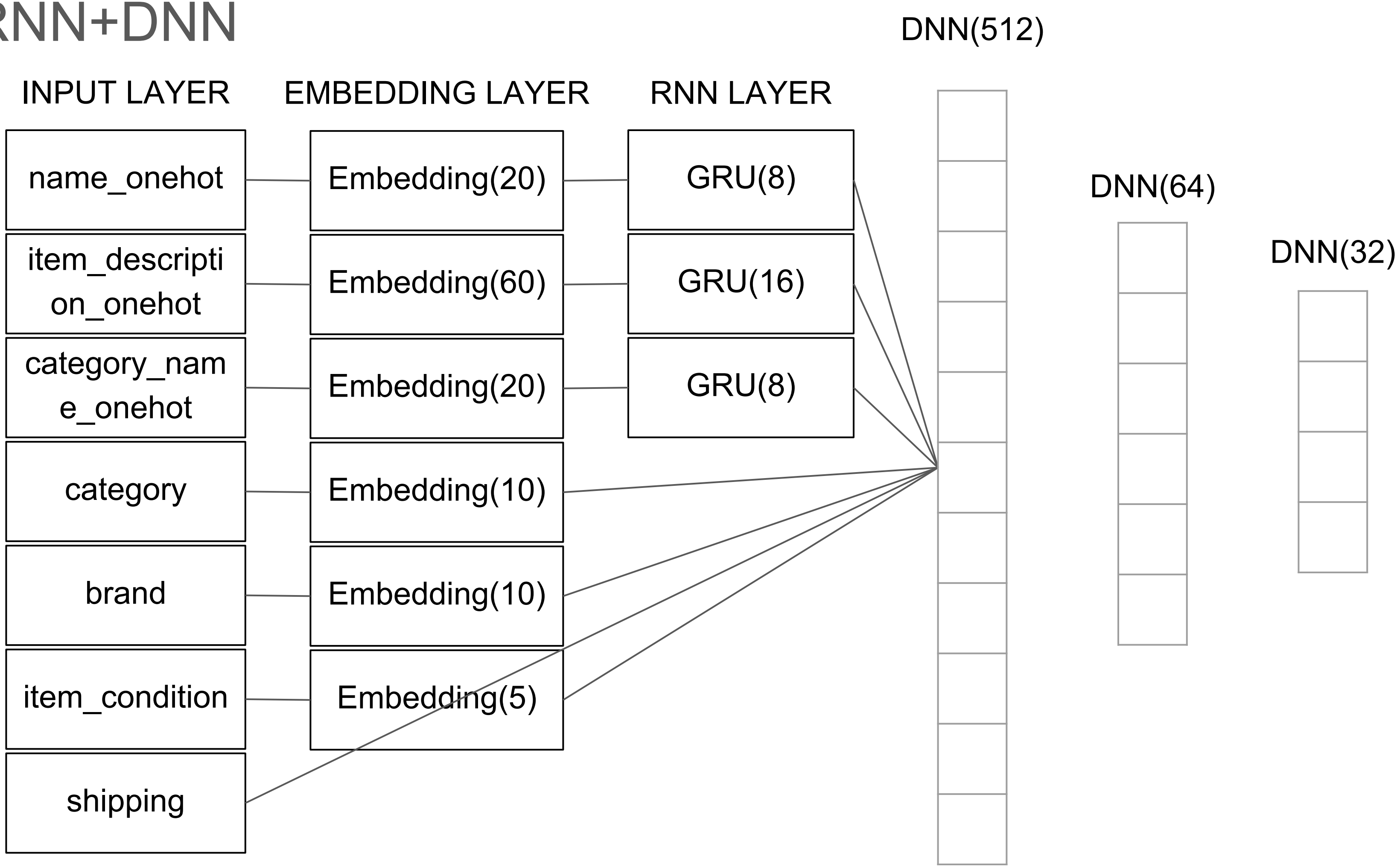
Apply Models

XGBoost, RNN+DNN

XGBoost

- Performance: 0.61
- Reason:
  - Memory: X should be calculated in advance. One hot for each term => word2vec
  - XGBoost: not good at Linear model.

RNN+DNN



INPUT LAYER
EMBEDDING LAYER
RNN LAYER
DNN(512)
DNN(64)
DNN(32)
name_onehot
item_descripti
on_onehot
category_nam
e_onehot
category
brand
item_condition
shipping
Embedding(20)
Embedding(60)
Embedding(20)
Embedding(10)
Embedding(10)
Embedding(5)
GRU(8)
GRU(16)
GRU(8)

RNN+DNN

- Performance: 0.44
- Reason:
  - Embedding layer help to save memory.
  - RNN helps to memorize all words in the sentence.
  - DNN helps to predict precise price.