知能情報実験 III グループ２最終発表

学籍番号・氏名　　195732F 大城悠翔　　195735A 林彦男

195754G 塩月海都　　195763F 大城璃功

2021 年 8 月 11 日

目次

1　概要

本グループではフリマアプリであるメルカリの出品データに対して、販売者が投稿した情報をもとに、適正な販売価格を予測することを対象問題とした。商品の価格というものは量や質、その他の要素によって大きく変化する。例として衣類のように季節やブランド名が大きく関係するものや、電子機器のように純粋なスペックが大きく関係するものがある [3]。私たちグループは様々な商品に対し適正である販売価格を予測し提案することにより、円滑な売買取引の一助となるのではと考え今回のテーマを設定した

1.1

2　はじめに

2.1　実験の目的と達成目標

知能情報実験 III は情報工学分野のより専門的な知識を理解習得を目的として実施される。そこで私たちの班はメルカリの価格予想チャレンジを通して機械学習の外観について理解し、特徴量抽出等の前処理、グループによる開発方法などの技術の習得を目指す。

2.2　テーマについて

本グループではメルカリの出品データに対して、販売者が投稿した情報を基に「適正な販売価格」を予測するすることを対象問題として設定した。機械学習で大量のデータから法則性を見つけ、正確性の高い予測をするためには、パラメータの調整や欠損しているデータの補完、データを特徴量へ変換する等の前処理、適切なモデルの選択などが必要不可欠である。これらの手順を理解するにはデータ数が適切な量で、ある程度データが欠損している等の前処理が行われていなかったり、一部の特徴が特徴量になっていないデータが用意されているデータセットであることが望ましい。そのため、これらの条件に該当するメルカリの出品データから価格予想をすることは機械学習の基本的な考え方や scikit-learn を用いた機械学習の理解に適していると判断した。

3　実験方法

本実験においては macbook air を用いて行った。実行環境としては python を用いたほかライブラリとして numpy、scikit-learn、pandas、matplotlib、nltk を使用した。データセットに関しては kaggle[3] において用意されたものを利用した。

3.1 実験目的

フリマアプリ「メルカリ」において販売者が投稿した情報を基に「適正な販売価格」を予測する。用意されたデータとして、ユーザー ID、商品の状態、実際の価格など 8 項目が与えられており、これらを元に価格を予想するモデルを作成する。

train_id or test_id	ユーザー投稿の ID
name	投稿のタイトル (リークを防ぐために値段の記述は消されている)
item_condition_id	販売者が提供した商品の状態
category_name	投稿のカテゴリー
brand_name	ブランドの名前
price	実際に売られた値段
shipping	送料のフラグ。1 ならば販売者負担、0 ならば購入者負担
item_description	投稿された説明の全文 (値段の記述は消されている)

3.2 データセット構築

kaggle の Mercari Price Suggestion Challenge に公開されていたものを利用した。ダウンロードしたメルカリ販売データの商品説明文などの文字列の特徴量に対して CountVectorizer を使用した BoW 化や TF-IDF などの処理を行いデータセットを整形した。また、BoW 化する際にはステミングを行い、語幹が同じ単語を一つの特徴量として扱うようにした。その際に、動詞や助詞、副詞などを省き、名詞や固有名詞のみで特徴ベクトルを生成しようとした。

3.3 モデル選定

scikit-learn のモデル選択チートシートを参考にして、回帰モデルを一通り試し、最もスコアが高かったものを利用することにした。サンプル数は 1 万個で, 学習後の決定係数を比較に利用した (表 1).

モデル	決定係数
RandomForestRegressor	0.48
Gradient Boosting Regressor	0.35
Ridge	0.01

表 1　モデルの比較

3.4 パラメータ調整

- CountVectorizer
  - min_df = 0.01
  - max_df = 0.99
  - stopword = text.ENGLISH_STOP_WORDS

min_df、max_df はそれぞれ出現頻度の低すぎる・高すぎる単語を省くためにあり、全文書中の割合で示す (%)。極端に少ない頻度で出現している単語を学習から省くために min_df を 0.01 に設定した。また、同じように極端に多い頻度で出現する単語も学習の際には有効ではないと判断したため max_df を 0.99 に設定して除外した。

- RandomForestRegressor
  - n_jobs=-1
  - min_samples_leaf=5
  - n_estimators=200

参考サイトから引用。BoW の調整に時間がかかったためあまり触れることができなかった。パラメータはそれぞれ、n_jobs は使用 cpu 数、min_samples_leaf は葉ノードのサンプル数制限、n_estimators は構築する決定木の数を示す。

4 実験結果

以下の表 2 は、NLP の効果を確かめるために文字列特徴量に対して単純なダミー変数化の場合と BoW との比較を表している (表 2)。評価には score 関数から得られた決定係数を用いた。

	訓練用データ	テスト用データ
BoW	0.33	0.26
ダミー変数化	0.73	0.48

表 2　比較結果

ダミー変数化の方が BoW より 2 倍近く高い決定係数を得られた。これはダミー変数化の方が精度が高いということを示している。一方で、訓練用とテスト用の差は広がってしまっている。それぞれの予測値と実際の値のグラフを以下に示す (図 1, 2).

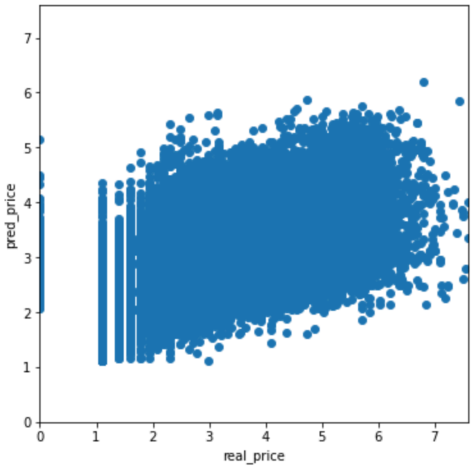

図 1　Bag of Words

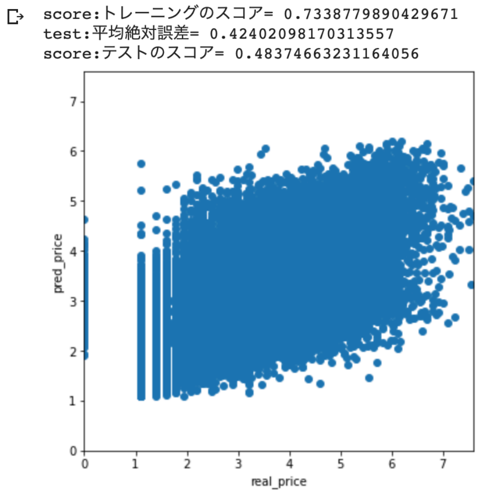

図 2　ダミー変数化

5　考察

文字列特徴量に対して BoW 化を行ったが、そのうちの Category の特徴に関しては単語にうまく分割することができなく、beauty/makeup/face など、/を取り除くことができずにいくつかの単語が結合してしまっている状態で学習しているので、うまくいってないと思われる。前処理後の特徴量数も少なくなっているため、CountVectorizer 適用範囲の下限をもっと下げる必要があった。また、図 3 にあるような”new” や ”free” などの比較的出現回数が高いが目的変数に対して意味のありそうな単語の影響を考慮して、TF-IDF で重み調整したり、”size”や”rm”(元は値段だったところをメルカリ側が変換) などのあまり予測に関係のないと思われる単語に対して stopwords に追加するという方法もあった。

図 3　stopwords 除去後の wordcloud

6　意図していた実験計画との違い

今回の実験ではできる限り精度を高めることを目標としていたが、モデルの選定や自然言語処理の難易度が想定より高く、計画より時間がかかったため以上のような結果となった。本来の実験計画では他のモデルとの比較や使用モデルのパラメータチューニング、特徴量ごとに最適な処理を行う予定であり、途中まで行ったものもあったが満足な結果には至らなかった。また、ソースコード等を関数レベルや処理毎に役割分担して進める予定だったが、どこをどう分担して良いのかうまく判断できず結局は個人での成果を持ち合い検討していくという形となった。計画を立てる段階では、なるべく実行できやすい事だけを並べて、発展させる所は後から追加していくことでこのような事態は避けられると思われる。

7　まとめ

今回の実験ではメルカリの価格予想という題材を設定しグループで機械学習用いて取り組んだ。その中で回帰におけるモデルごとの特徴や必要な特徴量の選択とその処理方法について知見を深めることができた。モデルの選定や自然言語処理を行っていく中で、いくつかの候補を挙げて試すことができた。候補に挙がったモデルやその他の処理においても一つずつ勉強しながら実験していくような状態であったため、ある程度の時間を伴ったが最終的には形にすることができた。今後は本実験の経験を活かし、機械学習等の実験を行っていきたい。

参考文献

[1] レポートテンプレート, https://github.com/naltoma/info3dm-report-template/blob/master/template.pdf , 2021/07/29

[2] Kaggle メルカリ価格予想チャレンジの初心者チュートリアル、https://www.codexa.net/kaggle-mercari-price-suggestion-challenge/

[3] Mercari Price Suggestion Challenge , https://www.kaggle.com/c/mercari-price-suggestion-challenge/data