Mobile Phone Price Prediction with Feature Reduction

Menghan Chen *

Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada

* Corresponding Author Email: menghan4@ualberta.ca

Abstract. Feature reduction can reduce data dimensionality and streamline model size, which focuses on the high relevance data and inferences the output faster. This paper aims to explore the performance and effectiveness of feature reduction methods that accompany the Multilayer Perceptron classifier in predicting the mobile phone price range. Pearson's Correlation and Principal Components Analysis are chosen as the feature reduction techniques in the research. The experiment sorts the features in significant order with two distinct methods. The three experimental groups reduce 5 features each time and the control group has no feature selection. Then all the groups use the open dataset to train and test the accuracy and loss through MLP. The result indicates that the feature selected by the correlation coefficient facilitates the accuracy of the classification model. When PCA is implemented and only a few features get reduced, the performance improves a little bit, but when more features are eliminated there are huge negative influences. Pearson's correlation has a better performance than PCA in this experiment, which achieves 95.8% accuracy and validate the effectiveness of the feature reduction method.

Keywords: Machine Learning, Classification, Feature Reduction, Correlation, PCA.

1. Introduction

In the past decade, the market for mobile phones has experienced explosive growth due to their indispensable role in communication, social media, work, making payment, and entertainment. More and more businesses are feasible to be done through mobile phones. The penetration rate of the mobile phone has shown a rapid increase; in 2016 the penetration rate was less than 50%, but by 2020 it had reached 78.05% [1]. However, market saturation is also something to consider. The high quality of phones leads to high sustainability and less need for replacement. In addition, there has been a reduction in the number of mobile phones sold due to the impact of COVID-19 recession [1]. Not to mention, customers have become more rational in terms of choosing which mobile phones to purchase; most people best prefer the product that is worth its price. The price of a mobile phone is affected by its brand, features, and specification. Pricing the newly released mobile phone with a reasonable price compared with the rest of the market can help it to launch successfully [2].

Prediction of the price of mobile phones, which is done by training the model with given data from the current market, is a supervised learning problem. In more detail, this is a multi-class classification problem designed for labeling mobile phones of different price ranges based on their features. A large dataset is used in the prediction in order to cover the variety of features that the mobile phone offers. However, the irrelative dimensions are redundant and may cause negative consequences in the building model [3]. Feature selection has the ability to select the features that have high relevance from the initial datasets, which helps reduce the complexity of the dataset and also simplifies the model [2, 3].

In Keval Pipalia and Rahul Bhadja's research article, they compared the performance of five distinct commonly used supervised learning algorithms in the mobile price classification problem. The experiment follows the classification methodology and comes up with a rank of precision that involves different algorithms as a conclusion; of these algorithms, the performance of logistic regression is not outstanding [4]. In another aspect, the logistic regression shares "common roots in statistical pattern recognition" with the artificial neural network [5]. Stephan Dreiseitl and Lucila Ohno-Machado sampled 72 papers in their journal for comparing these two methodologies. They found out that even though none of the algorithms has absolute advantages, the neural network has no worse performance because the neural network is seen as a nonlinear generalization of the logistic regression. Multilayer perceptron (MLP) is one of the artificial neural network methods that is used in multiclass classification, which is available to add layers for applying feature selection.

The paper aims to explore the effectiveness of feature reduction of dimension. The method introduces two mechanisms to reduce the feature dimension, including Principal Component Analysis (PCA) and feature correlation factors. The experiments are conducted using the public dataset, which shows that the Pearson's correlation can obtain more satisfying performance than the PCA method.

2. Method

2.1. Data Description

The data used in this research comes from the train.csv file from Mobile Price Classification, which was uploaded by Abhishek Sharma and made available on Kaggle Datasets [6].

Normally, pre-processing is a necessary step for data analysis after collecting the data from the real world. Proper pre-processing could eliminate the negative effect of variations in data for building parsimonious models [7]. With data inspection, there are no missing attribute values or noisy data found in the given dataset. The author has already done partial preprocessing for uploading the clean dataset. The chosen dataset contains 2000 instances with 21 columns of features, including 20 distinct features of mobile phones and the price range for the phone that has those features. The detail of feature names, descriptions, and data types are shown in Table 1 below.

Table 1. Data Description, data from [6]

Name	Description	Type	Category
battery_power	Battery capacity (mAh)	int64	Discrete
blue	Bluetooth: 1 for yes, 0 for no.	int64	Nominal
clock_speed	Clock speed (GHz)	float64	Discrete
dual_sim	Dual SIM: 1 for yes, 0 for no.	int64	Nominal
fc	Camera pixel (MP)	int64	Discrete
four_g	4G: 1 for yes, 0 for no.	int64	Nominal
int_memory	Memory size (GB)	int64	Discrete
m_dep	Depth of mobile (cm)	float64	Discrete
mobile_wt	Weight (g)	int64	Discrete
n_cores	Number of cores: 1 to 8.	int64	Discrete
pc	Primary camera (MP)	int64	Discrete
px_height	Height of resolution (px)	int64	Discrete
px_width	Width of resolution (px)	int64	Discrete
ram	RAM size (MB)	int64	Discrete
sc_h	Height of screen (cm)	int64	Discrete
sc_w	Width of screen (cm)	int64	Discrete
talk_time	Time that a battery can continuous running without charging (sec).	int64	Discrete
three_g	3G: 1 for yes, 0 for no.	int64	Nominal
touch_screen	Touch screen: 1 for yes, 0 for no.	int64	Nominal
wifi	Wi-Fi: 1 for yes, 0 for no.	int64	Nominal
price_range	Price range: 0 to 3, ascending order in the amount.	int64	Ordinal

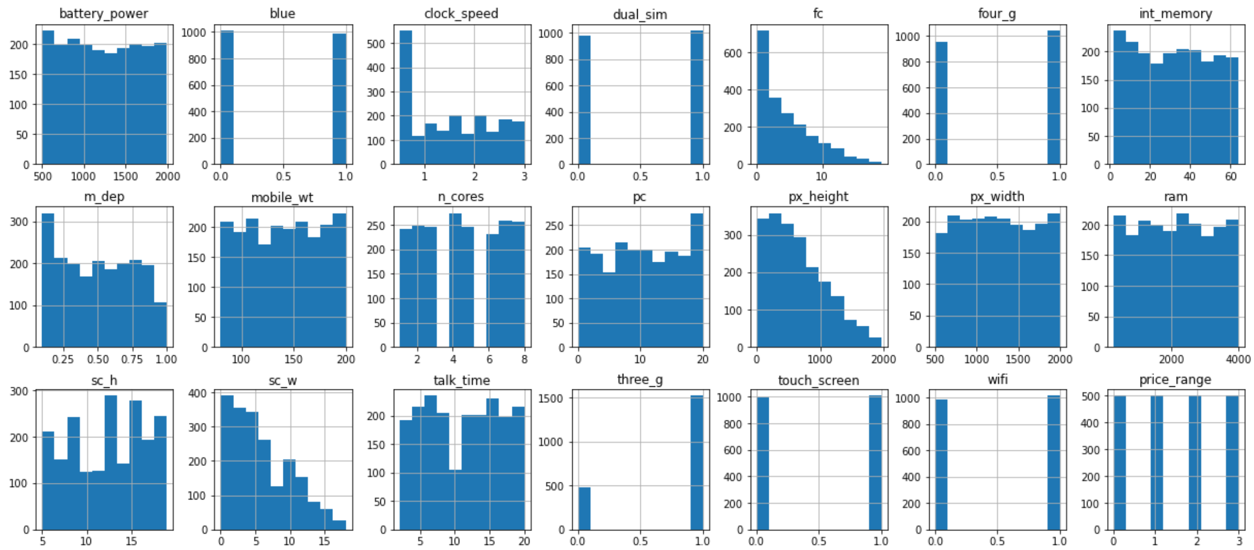


Figure 1. Histogram of Features, data from [6]

Figure 1 above is the histogram collection of 21 features, which indicates the data distribution and the range of each feature. Combining Table 1 and Figure 1, distinct features have both quantitative and qualitative in varying scales, and the range of each feature also shows clear differences. Standardizing the dataset to scale is necessary for applying the data to machine learning, otherwise when there is a feature with a large scale for example, it will affect the model detrimentally. The standardization follows the z-score formula that is shown as follows:

$$z = \frac{\mathrm{x}-\mu}{\sigma} \tag{1}$$

Where x is the origin data value, μ is the mean value and σ is the standard deviation. After scaling to unit variance, each feature in the dataset has a zero-mean with standard deviation 1 [8].

To check the performance after implementing methodologies, the dataset is split into two datasets. The 2000 instances have been randomly split into training dataset and testing dataset with a ratio of 4:1, i.e., there are 1600 pieces data for training and 400 pieces data for testing. For both sets, the 20 feature columns are stored as variable x for predictor variable and the price range column is the target value to be stored as y.

2.2. Feature Engineering

Figure 2 below shows the correlation heat-map in which all the features are compared pairwise. The correlation coefficients are annotated in the respective squares. A square with stronger color in red indicates a stronger positive correlation between the two corresponding variables. The squares in the last line show the correlation of each feature with the price range. According to the figure, 4 features have a correlation that is larger than 0.1 with the mobile price range, and only 1 feature in 4 reaches the strong correlation requirement, i.e., $r > 0.7$. Also, there are 3 features that have negative correlation exist.

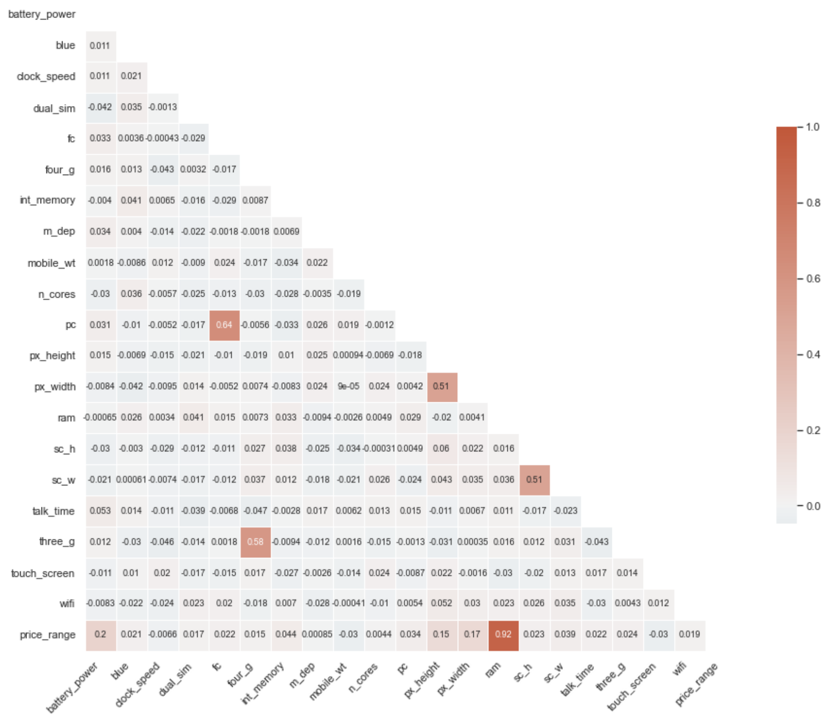


Figure 2. Correlation Heatmap, data from [6]

2.3. Multiple Layer Perceptron

Generally, Multilayer Perceptron (MLP) is a feedforward artificial neural network, which can also be seen as a perceptron's network. The MLP consists of an input layer, one or more hidden layers, and an output layer in a directed graph [9]. Figure 3 is shown in *Sharing Data and Models in Software Engineering* to present a simple scheme of a three-layer MLP, which shows the connections between neurons in adjacent layers with 3 input features and 1 output [10]. A layer of perceptron in MLP is a set of neurons which are fully connected with all the neurons in the subsequent layer [9]. All the black lines are showing the connection between nodes in adjacent layers, which have different weights that will keep on optimizing during the learning process.

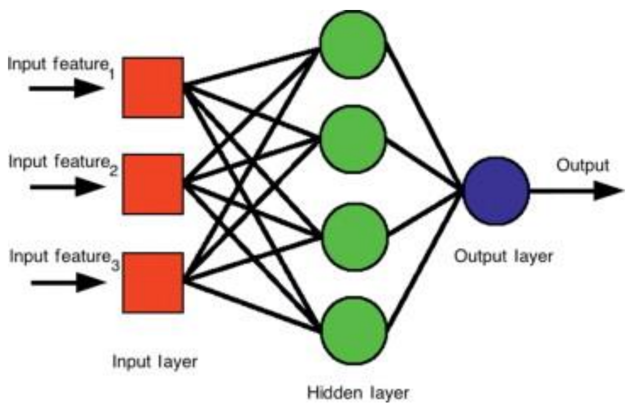


Figure 3. Scheme for three-layer MLP, resource from [10]

The value of the perceptron is calculated by the formula with input feature x and corresponding weight vector w and then passed to an activation function f [10]. The activation function is a step to help process the output based on the input through a certain function, which gives the neural network a chance to become nonlinear, or else the output of the model would be a linear composition due to formula 2 below.

$$u(x) = \sum_{i=1}^{n} x_i w_i \tag{2}$$

Backpropagation is a commonly used algorithm for training the model. With the loss function, backpropagation can update the weight to minimize the error to optimize the model [10].

The dataset used in the research only has 20 features, which is not a huge dataset. In order to avoid overfitting, only 1 hidden layer with 10 neurons will be applied to the MLP classifier in the following sections. The hidden layer uses Rectified Linear Unit as the activation function that is shown in formula 3:

$$g(x) = \begin{cases} 0, & x \leq 0 \\ x, & x > 0 \end{cases} \tag{3}$$

for the output layer is the Softmax function [11]. The models are training 20 epochs with batch size 100. Also, Adam, the Adaptive Moment Estimation Optimizer, is used in a 0.01 learning rate.

The loss value and accuracy value are important measurements for evaluating the model that applies to the testing datasets. The accuracy of the model corresponds to the percentage of which the instances are classified correctly. To calculate the loss, the categorical cross entropy is needed where $\hat{y}_i$ indicates the output of model, and y_i is the target category value with the i-th instance.

$$Loss = -\sum_{i=1}^{n} y_i \cdot \log \hat{y}_i \tag{4}$$

2.4. MLP with feature selection

2.4.1 Correlation

Correlation, as shown in the heatmap in Figure 2, is a measurement that indicates the strength and direction of the relationship among the pairwise features. When correlation coefficient has a value of 1, features have a strong positive correlation, and -1 for strong negative correlation [12]. In addition, all the correlation coefficient should between range from -1 to 1 [12]. A positive correlation is when the variable increases while another one increases, and a negative correlation is when one variable increases and another one decreases [12]. Pearson's correlation coefficient formula is popular for use. In this formula, x and y represent the two variables respectively:

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}} \tag{5}$$

2.4.2 Principal Component Analysis

Principal Component Analysis (PCA) is commonly used for dimension reduction with the orthogonal transformation of the dataset. As its name suggests, PCA can maintain the significant features in the dataset and reduce the number of input variables. When implementing PCA on the dataset, the objective is to try to change the basis in the space and find a projection that could preserve as many features that can represent the dataset as possible. Furthermore, eliminating the size of the input layer can help the model run faster.

With applying the PCA, the dataset has to be centered and standardized to unit variance for calculating the covariance matrix as pre-treatment [13]. Then the features with the larger eigenvalues, which bring more information, will be transformed to the principal components [13].

The order results after implementing the two feature reduction methods, correlation and PCA, are shown below in Table 2. Sorting for correlation is based on the correlation coefficient, and PCA is based on the eigenvalue. The features are sorted in descending order according to the significance based on different methods, which will be used in the experiment as different groups. Since the order relates to the time that the feature is reduced, the order could be the main factor that influences the performance of feature reduction methods.

When comparing Correlation and PCA columns, the feature "battery_power" is in the 2nd and 1st place respectively. Feature "battery_power" is also the only feature that appears in top-5 for both columns, which indicates its importance in both methods. Another notable feature is "ram", which is in first place in the correlation column and in last place in the PCA column. Feature "ram" has the largest correlation coefficient, which indicates it has the strongest linear association with "price_range". However, PCA does not simply eliminate the unrelated features, it transfers the features into a new basis that has the features' projection mapping with large variance. Feature "ram" disappears in this process, since others bring more information.

Table 2. Feature order sorted by correlation and PCA

Significant Order	Correlation	PCA
1	ram	battery_power
2	battery_power	blue
3	px_width	clock_speed
4	px_height	dual_sim
5	int_memory	fc
6	sc_w	four_g
7	pc	int_memory
8	three_g	m_dep
9	sc_h	mobile_wt
10	fc	n_score
11	talk_time	pc
12	blue	px_height
13	wifi	px_width
14	dual_sim	sc_w
15	four_g	talk_time
16	n_cores	touch_screen
17	m_dep	wifi
18	clock_speed	three_g
19	mobile_wt	sc_h
20	touch_screen	ram

3. Results

3.1. Experimental results

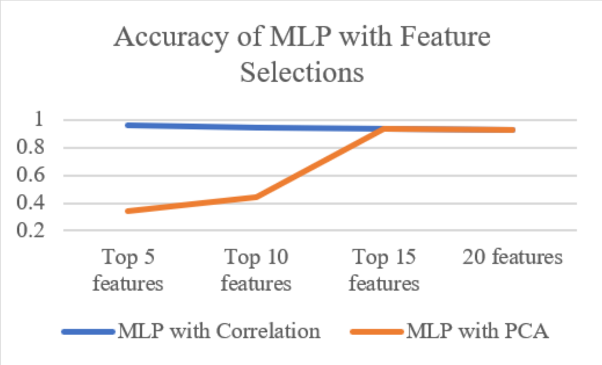
The weights in the model are initially assigned randomly. For reducing the affection and error, the results are averaging 100 runs. Table 3 and Table 4 show the Accuracy and Loss of testing the MLP with different feature selection methods. Figure 4 shows the corresponding 2D line graph.

Table 3. Accuracy comparison between MLP with and without feature selection.

Accuracy	MLP with Correlation	MLP with PCA
Top 5 features	0.9584999799728393	0.3406249988079071
Top 10 features	0.9465249913930893	0.43974999874830245
Top 15 features	0.9353500020503998	0.9321750038862229
20 features	0.9284250074625016	0.9284250074625016

Table 4. Loss comparison between MLP with and without feature selection.

Loss	MLP with Correlation	MLP with PCA
Top 5 features	0.12039686642587184	1.3481214666366577
Top 10 features	0.1389427600055933	1.1876236557960511
Top 15 features	0.1606126058101654	0.16126121111214162
20 features	0.17376982241868974	0.17376982241868974



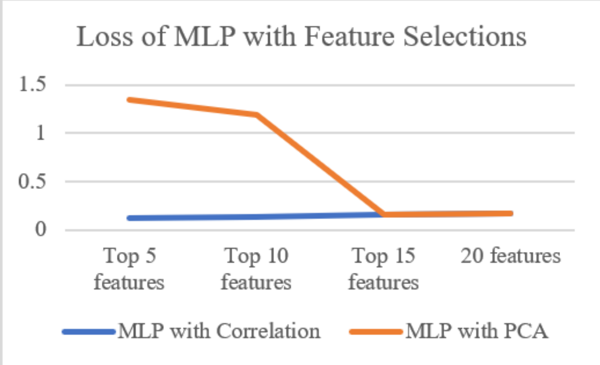


Figure 4. Accuracy and loss of MLP with feature selection.

3.2. Analysis

With the experiment result shown in Section 3.1, implementing Pearson's correlation as the feature selection will result in higher accuracy and a lower loss compared to no feature selection at all. The accuracy keeps increasing as the number of features decreases. By eliminating negative and zero correlation coefficients, the accuracy of the model will increase.

On the other side, the principal component analysis shows poor performance in both top-5 and top-10 features. The top-15 features group has a better performance than no feature selection, but it is not that significant. The reason that the failure took place may be the lack of information on the remaining features. After the PCA process, more features are needed to help build the model properly.

4. Conclusion

This paper focuses on how feature reduction techniques influence the performance of the Multilayer Perceptron classifier in the mobile phone price classification problem. The two feature reduction methods are Pearson's correlation and PCA. Without any feature reduction method, MLP is 92.84% accurate. When there are less features, the performance of the correlation group is improving. When there are only top-5 features that have a strong correlation coefficient with the target value, correlation methods could reach 95.85%. Data reflects that the accuracy of PCA only gets better when the data accounts for up to top-15 features, but when limited to only top-5 features the accuracy plummets; 93.22% for top-15 features and 34.06% for top-5 features. In conclusion, these two feature reduction methods are helpful in improving the performance accompanied by multilayer perceptron classification. However, choosing an appropriate component value n based on the data is what determines if the result is valid. In this paper, only 2 feature reduction methods and 3 experimental groups have been used for completing the task. In the future, there are other feature reduction features that can be explored. Furthermore, there is the possibility of discovering a solution that produces better accuracy when experimental groups are subdivided into smaller ranges.

References

[1] Resource from https://www.statista.com/statistics/263437/global-smartphone-sales-to-end-users-since-2007/

[2] Kumuda S, Vishal Karur, Karthick Balaje S E. (2021). Prediction of Mobile Model Price using Machine Learning Techniques. International Journal of Engineering and Advanced Technology, 11(1), 273-275.

[3] Miao, J., & Niu, L. (2016). A survey on feature selection. Procedia Computer Science, 91, 919-926.

[4] Pipalia, K., & Bhadja, R. (2020). Performance Evaluation of Different Supervised Learning Algorithms for Mobile Price Classification. In International Journal for Research in Applied Science & Engineering Technology, 8(6), 1841-1848.

[5] Dreiseitl, S., & Ohno-Machado, L. (2002). Logistic regression and artificial neural network classification models: a methodology review. Journal of biomedical informatics, 35(5-6), 352-359.

[6] Resource from https://www.kaggle.com/datasets/iabhishekofficial/mobile-price-classification

[7] Famili, A., Shen, W. M., Weber, R., & Simoudis, E. (1997). Data preprocessing and intelligent data analysis. Intelligent data analysis, 1(1), 3-23.

[8] Ali, P. J. M., Faraj, R. H., Koya, E., Ali, P. J. M., & Faraj, R. H. (2014). Data normalization and standardization: a technical report. Mach Learn Tech Rep, 1(1), 1-6.

[9] Murtagh, F. (1991). Multilayer perceptrons for classification and regression. Neurocomputing, 2(5-6), 183-197.

[10] Menzies, T., Kocaguneli, E., Turhan, B., Minku, L., & Peters, F. (2014). Sharing data and models in software engineering. Morgan Kaufmann

[11] Nair, V., & Hinton, G. E. (2010, January). Rectified linear units improve restricted boltzmann machines. In International Conference on Machine Learning.

[12] Schober, P., Boer, C., & Schwarte, L. A. (2018). Correlation coefficients: appropriate use and interpretation. Anesthesia & Analgesia, 126(5), 1763-1768.

[13] Wold, S., Esbensen, K., & Geladi, P. (1987). Principal component analysis. Chemometrics and intelligent laboratory systems, 2(1-3), 37-52.