Mobile Price Classification

Group 1

Yunuo Ma, Tianle Zhu, Xinyi Wei, Citina Liang, Wannian Lou, Jiaqi Yuan

Problem Description & Goal

Bob has started his own mobile company. He wants to compete against big companies like Apple,Samsung etc. He does not know how to estimate price of mobiles his company creates. In this competitive mobile phone market you cannot simply assume things.

To help him solve this problem, we collect sales data of mobile phones of various companies. We want to find out some relation between features of a mobile phone(eg:- RAM,Internal Memory etc) and its selling price. We use the price range indicating how high the price is.

Data Description

Numerical Variables:

Battery power: Total energy a battery can store in one time measured
Clock_speed: Speed at which microprocessor executes instructions
Fc: Front camera megapixels
Int_memory: Internal memory in Gigabytes
M_dep: Mobile depth in cm
Mobile_wt: Mobile weight of mobile phone
N_cores: Number of cores of processor
Pc: Primary Camera megapixels
Px_height: Pixel Resolution Height
Px_width: Pixel Resolution Width
ram : Random Access Memory in Megabytes
Sc_h : Screen Height of mobile in cm
Sc_w : Screen weight of mobile in cm
Talk_time: longest time a single battery charge will last when you are

Categorical variables:

Bluetooth: Has bluetooth or not
Dual_sim: Has dual sim support or not
Four_g: Has 4G or not
Three_g: Has 3G or not
Touch_screen: Has touch screen or not
Wifi: Has wifi or not

Predicted variable:

Price_range: 4 levels of price range from low to high

Sample size:

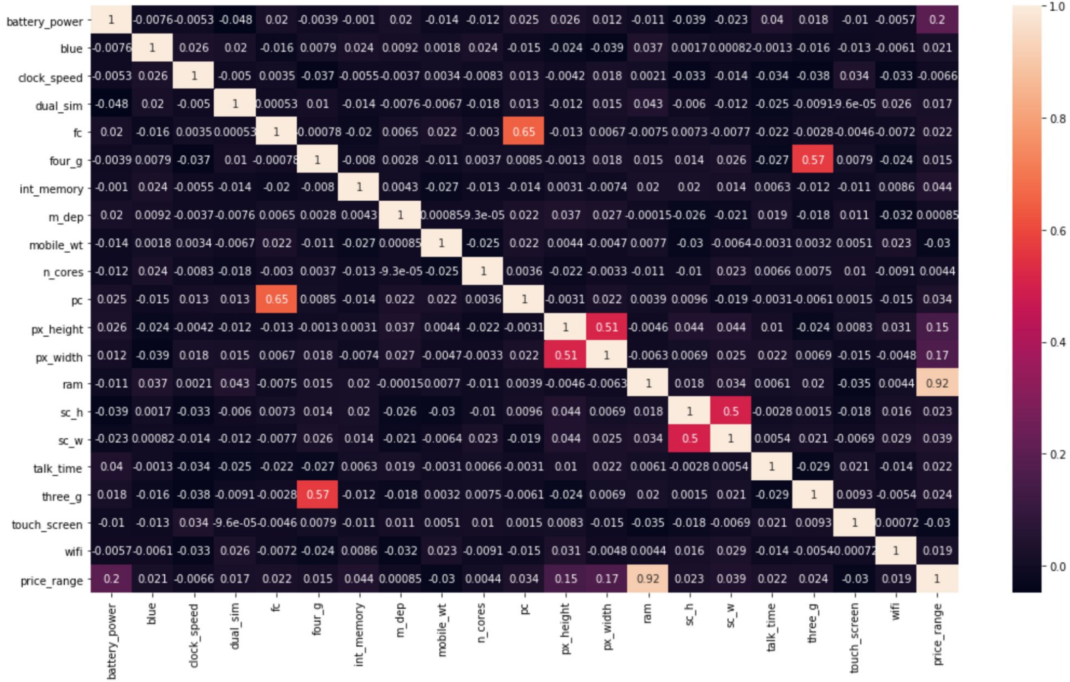
dataset: 2000 x 21
We split it into 70% training 30% testing

Simple Outline

- Exploratory Data Analysis(correlation, multicollinearity, data description)
- Data Cleaning
- Data Scaling
- Checking Data Distribution
- Future Selection and Comparison
- Modeling and Evaluation
- Conclusion and Application

EDA

Correlation between numerical feature:

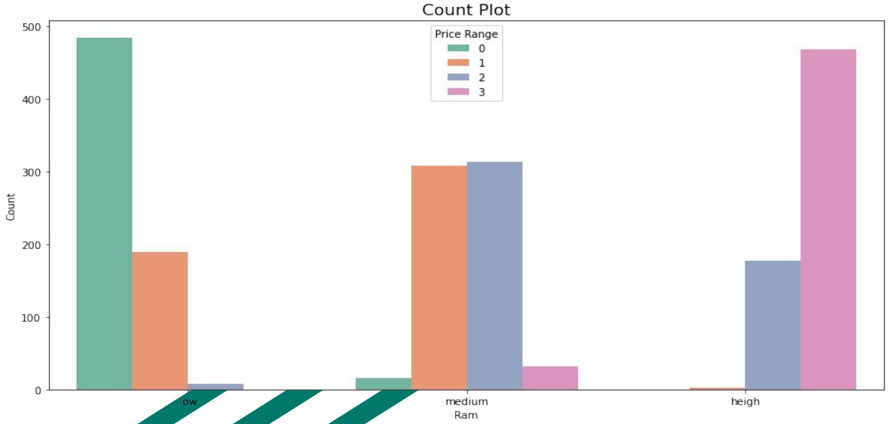


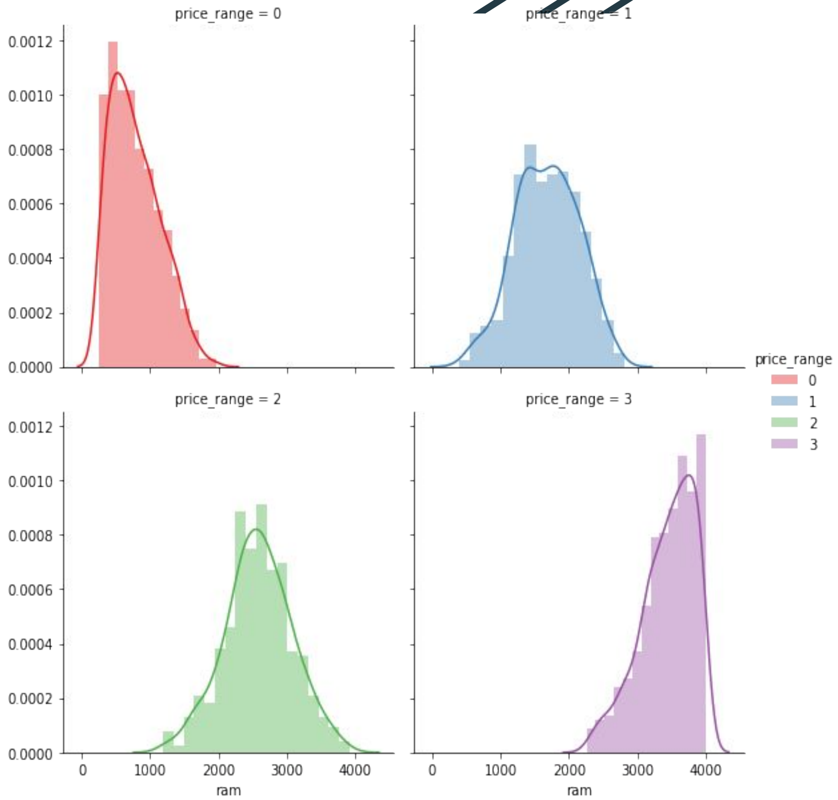
- The primary camera megapixels and the front camera megapixels are highly correlated
- The pixel resolution height and pixel resolution width are highly correlated
- The screen height and screen width are highly correlated
- Battery is uncorrelated with all other numerical variables in general
- The VIF is 20 and obviously there is no serious multicollinearity

EDA continued

price_range vs. ram:

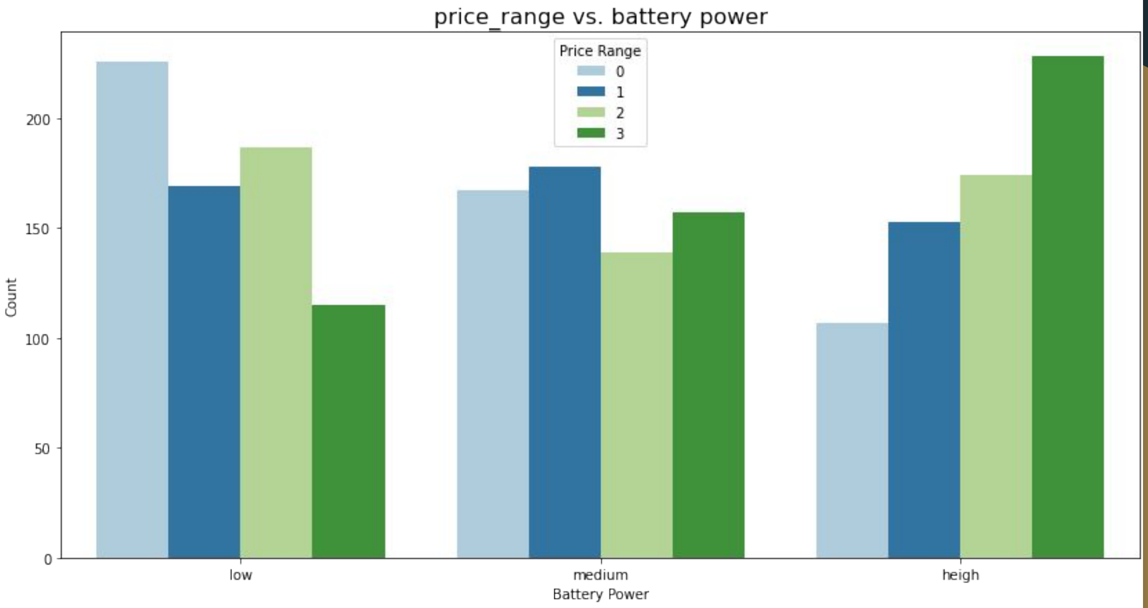
The distribution curve are changing form right skew to left skew as the price range getting higher, which means higher price range corresponding to higher RAMThe distribution curve are changing form right skew to left skew as the price range getting higher, which means higher price range corresponding to higher RAM.





EDA continued

We can find in the high group of battery power, the quantity of high price(price range=3) is much more bigger than others. therefore , we can infer the high battery power can determine higher price of cell phone.



Feature Selection

1. Variance filtering
2. Mutual Information method
3. Aoava filtering
4. Chi-square filtering
5. Embedded
6. Wrapper

Feature engineering is a very important part if we want to get good accuracy and save the time of computing. We tried all the methods above and find embedded method is best(just 4 features kept). However, it still worse than the original data because every feature contains the useful information. GIve the number of features are not large, we use the original features to get better accuracy.

Decision tree & Random forest

Random forest is an bagging ensemble algorithm. It can deal with the regression and classification problems. Its weak classifier is decision tree and therefore it can get better results than a single decision tree. In addition, we do not need to prepossessing the data, because it use the information entropy difference to classifier which is very convenient.

	Before feature selection	After feature
Accuracy: Decision tree	0.82	0.8433333333333334
Random forest	0.8666666666666667	0.9016666666666666

Logistic Regression

What: Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary and multiple dependent variable, although many more complex extensions exist.

Implementation: lrl2=LR(penalty="l2",solver="liblinear",C=0.5,max_iter=1000)

-	Before feature selection	After feature
Accuracy:	0.7533333333333333	0.7383333333333333

Dense Neural Network

What: Dense layer is the regular deeply connected neural network layer. We try the 2,3,4 layers and each layers' neuron are: n_neurons={1:512,2:256,3:256,4:128} and we give them a random weights and a random bias first. Then we add a dropout layer and a Relu layer. We compare the model of DNN+dropout+Relu with the DNN +relu and find it is almost the same. Therefore we omit the dropout layer and get the final result.

Implementation: tensorflow

Accuracy: 0.738

Naive Bayes

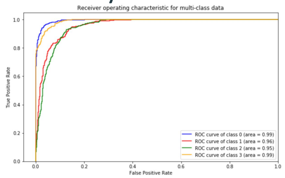
What: Naive Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naïve) independence assumptions between the features.

Implementation: GaussianNB()

Accuracy:

- Original accuracy: 0.8116

CatBoost & XGBoost

	CatBoost	XGBoost
Implement	CatBoostClassifier()	XGBClassifier()
Accuracy	- Accuracy: 0.845 - 	- Original accuracy: 0.91 - After GridSearch: 0.922 (parameters tuned: Parameters Tuned: learning_rate, max_depth, Min_child_weight, subsample, colsample_bytree, n_estimators)

LDA & QDA

	LDA	QDA
Implement	The latent Dirichlet allocation (LDA) is a generative statistical model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar.	Quaker Digital Academy (QDA) - A virtual academy providing online education to Ohio students.
Accuracy	0.956	0.932

KNN

- **k-nearest neighbors algorithm (k-NN)** is a non-parametric and non model building method proposed by Thomas Cover used for classification and regression. Therefore it is very fast and can deal with a lot of problems with a high accuracy.
- **Implementation:** KNeighborsClassifier()
- **Accuracy:**
    - Original accuracy: 0.95

SVM

- **Support Vector Machine (SVM)** is a supervised machine learning model that uses classification algorithms for classification.
- **Implementation:** SVC()
- **Accuracy:**
  - Original accuracy: 0.955
  - After GridSearch: 0.978
  - Parameters Tuned: C: 15, Kernel: linear, Degree: 2

Conclusion

Model	Random Forest	CatBoost	XGBoost	LDA	QDA	Logistic Regression	Naive Bayes	SVM	KNN	NN
Accuracy	0.88	0.845	0.922	0.956	0.932	0.783	0.812	0.978	0.95	0.738

Through trying different models, we figured that the SVM model gives the highest accuracy in price range prediction. We suggest Bob to use the SVM model on his data to decide on the price range. However, when we consider recall rate, CatBoost is the best model for this problem.

Application

This project aims at pricing the cell phone for the manufacturer and they can make a reasonable price for customers to purchase.
Also, customers can consider if it is worth buying given all of the features of one cell phone.

THANK YOU FOR YOUR LISTENING