**Submission by: Melinda Gomez Tellez**

**Business context:**

Robert has started his own mobile company. He wants to give tough fight to 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 solve this problem, he collects sales data of mobile phones of various companies.**

Robert wants to find out some relation between features of a mobile phone (e.g.: - RAM, Internal Memory etc.) and its selling price. But he is not so good at Machine Learning. So, he needs your help to solve this problem.

**In this problem you do not have to predict actual price but a price range indicating how high the price is**

Use SAS, Python or R to solve this problem

**Training and test dataset:** Data dictionary in page 2



**Model results template: This must be shared with the DAC team along with the self-nomination**

1. Is this a regression or classification problem? Reason for selection

**This is a classification problem as the target variable price\_range is a label, not an actual price. It appears most likely the actual prices were binned into 4 categories. What’s more, because there are more than 2 categories for the target, this is a multiclass classification problem.**

1. What are the top 3 insights derived from the exploratory data analysis?

* **The feature RAM by far is the dominant feature for predicting the price\_range. If you reference the histogram plot, you see a linear trend upwards of RAM as the price\_range increases.**
* **The remaining features (both numeric and Boolean) do not have a strong trend in having correlation with the price\_range. If we take a look at the cross validation score for each individual variable we see that the next most influential features are the pixel resolution (both height and width)**
* **This was a balanced multi-classification problem, we did not have to sample our set to create even sized price\_range groups.**
* **Correlation between the numeric features was practically non-existent except for pixel res height and width (which is expected) and pc and fc for camera pixels (again, in context this makes sense).**

1. How many features should be included as part of the model and what is the measure of selection?

**The 4 most influential features when comparing individually to the price\_range were ram, battery\_power, px\_height, px\_width. However, ram was the most important to include by far as on its own it roughly predicted accurately 75% of the time.**

1. Does any of the given features require transformations and why?

**The first model I created was kNN and usually the features should be normalized to avoid amplification of the distance from features on a larger scale. This however did not make sense for this dataset when utilizing this model because ram was on the largest scale and was the most dominant feature. I did normalize all of the numeric variables and then applied my own hardcoded weights to the important features and ended up getting the same accuracy percentage as when the variables were not normalized. The dataset did not require any transformations as the Boolean values were already convert to 1/0 and the remaining values were numeric. There also weren’t any missing values.**

1. Reason to choose a specific algorithm over others? Other algorithms which can be used in this scenario.

**My primary algorithm for this analysis was kNN. The primary reason was that kNN is simple to understand as it utilizes distance and supports multi-classification. This also was a small dataset in terms of the number of features and this tends to work better for kNN. There weren’t any missing values and the training set was balanced. Though kNN is simple, this was an ideal dataset to practice this algorithm on. I did use a decision tree (just ran it through with basis parameters) as a gut check to ensure kNN was the way to go. Because of the simplicity of kNN, it would be easy to understand through data exploration, what features would be considered important and how they influence the model.**

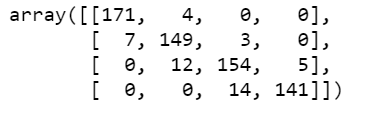
1. How will you fine tune the hyperparameters?

**For kNN, the only part to tune is to find the optimal number of neighbors. I therefore incorporated a grid search to find the optimal number of neighbors.**

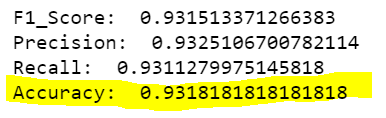
1. Measure of the model performance and the value

**Measure of Performance (split of training into train/test):**

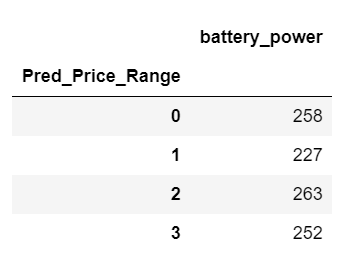
**Confusion Matrix:**



**Scoring:**



Prediction from Test Dataset:



1. How would you improve the model performance?

**I think the best approach, if I stick with kNN, is to find an ideal way to weight the most important features based on the results of cross validation. If that doesn’t work, it may be worth approaching a different algorithm.**

1. Top 5 insights derived out of the model parameters?
2. Top 3 items you have learned out of this exercise?

**Data dictionary:**

|  |  |  |
| --- | --- | --- |
| Column name | Description | Type |
| Id | ID | Numeric |
| battery\_power | Total energy a battery can store in one time measured in mAh | Numeric |
| blue | Has Bluetooth or not | Boolean |
| clock\_speed | speed at which microprocessor executes instructions | Numeric |
| dual\_sim | Has dual sim support or not | Boolean |
| Fc | Front Camera mega pixels | Numeric |
| four\_g | Has 4G or not | Boolean |
| int\_memory | Internal Memory in Gigabytes | Numeric |
| m\_dep | Mobile Depth in cm | Numeric |
| mobile\_wt | Weight of mobile phone | Numeric |
| n\_cores | Number of cores of processor | Numeric |
| Pc | Primary Camera mega pixels | Numeric |
| px\_height | Pixel Resolution Height | Numeric |
| px\_width | Pixel Resolution Width | Numeric |
| ram | Random Access Memory in Megabytes | Numeric |
| sc\_h | Screen Height of mobile in cm | Numeric |
| sc\_w | Screen Width of mobile in cm | Numeric |
| talk\_time | longest time that a single battery charge will last when you are | Numeric |
| three\_g | Has 3g or not | Boolean |
| touch\_screen | Has touch screen or not | Boolean |
| Wifi | Has wifi or not | Boolean |
| Price\_range | Target variable: This is the target variable with value of 0(low cost), 1(medium cost), 2(high cost) and 3(very high cost). Only available in Train and this has to be predicted for the test records | Numeric |