Mobile Price Range Prediction (mobile-prediction.ipynb)

- **Problem Statement**: The project aims to predict the price range of mobile phones (categorized into classes like 0, 1, 2, 3). This is a multi-class classification problem.
- **Dataset Used**
  :
  - Training data was loaded from train.csv and test data from test.csv.
  - The target variable in train.csv is 'price_range'.
  - Features include 'battery_power', 'blue' (Bluetooth), 'clock_speed', 'dual_sim', 'fc' (front camera megapixels), 'four_g', 'int_memory' (internal memory), 'm_dep' (mobile depth), 'mobile_wt' (mobile weight), 'n_cores', 'pc' (primary camera megapixels), 'px_height', 'px_width', 'ram', 'sc_h' (screen height), 'sc_w' (screen width), 'talk_time', 'three_g', 'touch_screen', 'wifi'.

- **Methodology and Approach**
  :
  - **Data Preprocessing**
    :
    - Checked for missing values; none were found in the training data.
    - Features were scaled using StandardScaler.

  - **Model Training**
    :
    - The training data was used to train the models.
    - Several classification models were trained and evaluated:
      - K-Nearest Neighbors (KNeighborsClassifier)
      - Logistic Regression (LogisticRegression)
      - Support Vector Classifier (SVC)
      - Decision Tree Classifier (DecisionTreeClassifier)
      - Random Forest Classifier (RandomForestClassifier)

  - **Model Evaluation**

:

    - Models were evaluated based on their accuracy scores on the training data itself (as a preliminary check) and then predictions were made on the separate test.csv (after dropping its 'id' column and scaling its features).
    - Confusion matrices and classification reports were generated for each model's performance on the training data.

- **Results and Conclusion**

  :

  - Accuracy on training data:
    - KNN: 95.4%
    - Logistic Regression: 97.0%
    - SVC: 97.6%
    - Decision Tree: 100% (indicates overfitting)
    - Random Forest: 100% (indicates overfitting)

  - The notebook proceeds to make predictions on the test.csv using the trained Logistic Regression model, chosen perhaps due to its high accuracy on training data without being 100% (less overfitting than Decision Tree/Random Forest on training).
  - The final predictions for the test.csv data were generated using the Logistic Regression model.
  - The Logistic Regression model was saved to mobile_price_model.pkl.