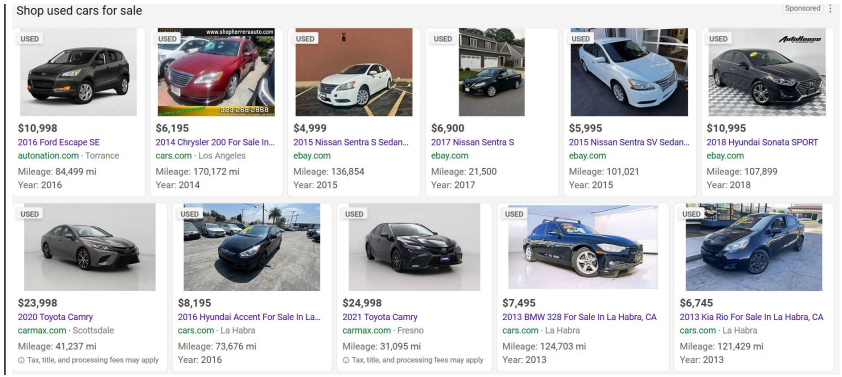
CS4210 Used Car Price Predictor

By: David Carbajal, Ethan Owusu-Bour, Noah Ojeda, David Malone

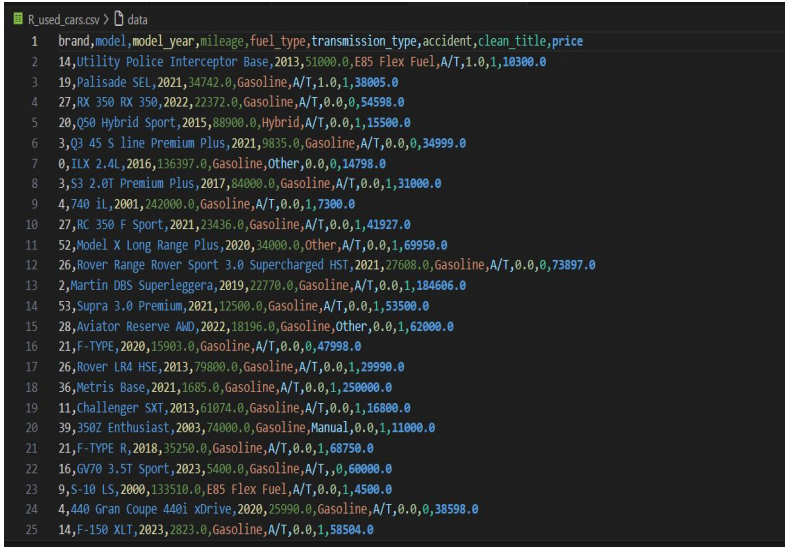
Purpose / Objective

- Predict used car prices using machine learning
- Improve prediction accuracy through feature engineering and preprocessing
- Compare baseline and more advanced ML models
- Evaluate model performance in real dollar values



Data Set

- Data source ↠ Kaggle ‘used_cars.csv’
- Total Amount of Records:
  - 4,009 car listings | 12 feature columns
- Features include: brand, model, mileage, fuel_type, transmission_type, model_year, price, etc.
- Includes both car characteristics and condition
- Contains a mix of categorical, numerical, and descriptive fields
- Target Class ↠ Price



R_used_cars.csv > data

```
brand,model,model_year,mileage,fuel_type,transmission_type,accident,clean_title,price
14,Utility Police Interceptor Base,2013,51000.0,E85 Flex Fuel,A/T,1.0,1,10300.0
19,Palisade SEL,2021,34742.0,Gasoline,A/T,1.0,1,38005.0
27,RX 350 RX 350,2022,22372.0,Gasoline,A/T,0.0,0,54598.0
20,Q50 Hybrid Sport,2015,88900.0,Hybrid,A/T,0.0,1,15500.0
3,Q3 45 S line Premium Plus,2021,9835.0,Gasoline,A/T,0.0,0,34999.0
0,ILX 2.4L,2016,136397.0,Gasoline,Other,0.0,0,14798.0
3,S3 2.0T Premium Plus,2017,84000.0,Gasoline,A/T,0.0,1,31000.0
4,740 iL,2001,242000.0,Gasoline,A/T,0.0,1,7300.0
27,RC 350 F Sport,2021,23436.0,Gasoline,A/T,0.0,1,41927.0
52,Model X Long Range Plus,2020,34000.0,Other,A/T,0.0,1,69950.0
26,Rover Range Rover Sport 3.0 Supercharged HST,2021,27608.0,Gasoline,A/T,0.0,0,73897.0
2,Martin DBS Superleggera,2019,22770.0,Gasoline,A/T,0.0,1,184606.0
53,Supra 3.0 Premium,2021,12500.0,Gasoline,A/T,0.0,1,53500.0
28,Aviator Reserve AWD,2022,18196.0,Gasoline,Other,0.0,1,62000.0
21,F-TYPE,2020,15903.0,Gasoline,A/T,0.0,0,47998.0
26,Rover LR4 HSE,2013,79800.0,Gasoline,A/T,0.0,1,29990.0
36,Metris Base,2021,1685.0,Gasoline,A/T,0.0,1,250000.0
11,Challenger SXT,2013,61074.0,Gasoline,A/T,0.0,1,16800.0
39,350Z Enthusiast,2003,74000.0,Gasoline,Manual,0.0,1,11000.0
21,F-TYPE R,2018,35250.0,Gasoline,A/T,0.0,1,68750.0
16,GV70 3.5T Sport,2023,5400.0,Gasoline,A/T,,0,60000.0
9,S-10 LS,2000,133510.0,E85 Flex Fuel,A/T,0.0,1,4500.0
4,440 Gran Coupe 440i xDrive,2020,25990.0,Gasoline,A/T,0.0,0,38598.0
14,F-150 XLT,2023,2823.0,Gasoline,A/T,0.0,1,58504.0
```

Preprocessing

- Data Cleaning
  - removing outliers using IQR to reduce noise (41 rows / instances removed)
  - removed instances with missing values
  - Standardized categorical fields and converts strings “$15,000” into numeric values
- Feature Engineering
  - Converted flags (clean_title, accident) to 0 or 1
- Handling Categorical Data
  - Applied One Hot Encoding to features like brand, model, fuel_type, and transmission

Methodology

Trained three models:

- Random Forest Generator (Python)
- Linear regression model (R/RStudio)
- Deep Neural Network (Python)

Used a 90% / 10% training split

- Split occurred before encoding to prevent a data leak (training data doesn't influence the testing data)

All three models used log transformation to improve model's ability

Post Processing

R^2 (Coefficient of Determination) - how well the model explains variability in the target.

1. Take your model's total squared error.
2. Take the total squared error from always predicting the average.
3. Divide the model's error by the average-only error.
4. Subtract that result from 1.

That final number is R^2.

Meaning: R^2 tells you how much of the changes in price your model can explain compared to just guessing the average. The closer to 1 the better.

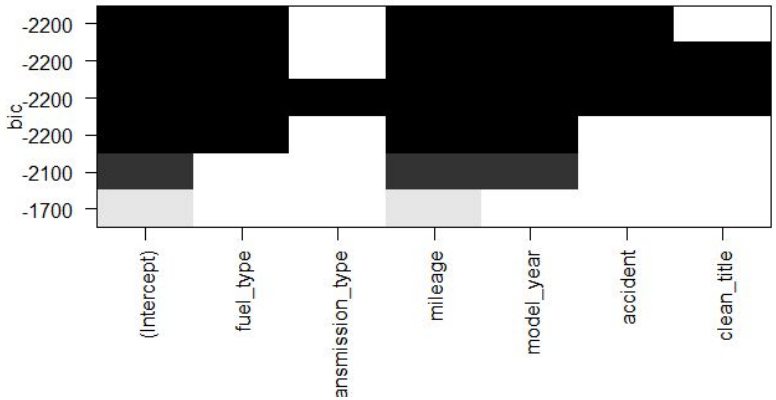
RSME (Root Mean Square Error) - average error between predicted and actual prices.

1. Find each error: (actual − predicted)
2. Square each error
3. Average the squared errors
4. Take the square root

Meaning**:**
Average size of prediction mistakes (in dollars), big mistakes punished more.

Post Processing

- Regression model
  - Original RMSE: $104,468.85
  - RMSE: $15,539.39
  - R^2: 0.6198
- Random Forest
  - Original RMSE: $176,940.23
  - RMSE: $12,850.14
  - R^2: 0.6383
- Deep Neural Network
  - RMSE: $69,034.94
  - R^2: 0.2662
- Random Forest outperformed the Linear Regression Model and the DNN



Takeaways / Future Improvements

Takeaways

- Random Forest delivered the best predictive performance
- Log-transforming price was essential due to skewed distribution
- Encoding after the split prevented data leakage
- Cleaned dataset of 4,009 cars provides strong modeling potential

Future Improvements

- Try more advanced models:
  - XGBoost
  - LightGBM
  - Gradient Boosting Regressor
- Add additional features:
  - Location / region
  - Car condition scoring
  - Number of owners
  - Accident severity
  - More balance between car brands