ITSCA2 · Scientific Computing with Python

# Project 2

*A Vehicle Valuation & Decision-Support System for Mutuka Automotive*

***Drive Value, Drive Trust.***

**Prepared by:** Jacobus Erasmus · eduv8821832
**Module:** ITSCA2-12 — Scientific Computing with Python
**Lecturer:** Ms KP Thubisi **Campus:** Pretoria

# The Brief & Today's Talk

- Goal: a specification-based first-level valuation + decision-support tool (no mileage / age / condition data)
- Data: 205 → 201 clean vehicles; price is driven by engine size, weight and power
- Segments: four economic tiers, plus use-case groups that cross-cut price
- Model: Random Forest - about R34k error (11%), R-squared 0.82, correct band 78% of the time

**Takeaway — A dependable first-level estimator for mainstream stock, with clear limits at the premium end.**

# Drive Value, Drive Trust.

- Mission — Accurate Valuations, Confident Decisions
- Vision — Leading the Future of Car Reselling
- Data-Driven Valuations
- Trusted Automotive Experts
- Smart Selling Solutions

# Objective & What Success Looks Like

- Business problem: fast, consistent, defensible first-level valuations
- Analytical task: predict price and categorise vehicles from specifications
- Success = accurate within an agreed error band + clear manual-review rules
- Measured by MAE / RMSE / $R^2$ and the share safely auto-valued

MAE (Mean Absolute Error) — average rand difference between predicted and actual price, easy to explain to non-technical stakeholders. RMSE (Root Mean Squared Error) — similar to MAE but squares errors before averaging, so large misses are penalised more heavily. $R^2$ (R-squared) — share of total price variation explained by the model; 1.0 means perfect prediction, 0 means no better than always guessing the average price.

# The Data — and Its Limits

- 205 vehicles × 26 specification attributes; target = price
- Quality work: missing values, duplicates, inconsistent categories, outliers
- Key limitation: no mileage, age, condition, service or accident history
- → a specification-based first-level tool, not a full market valuation

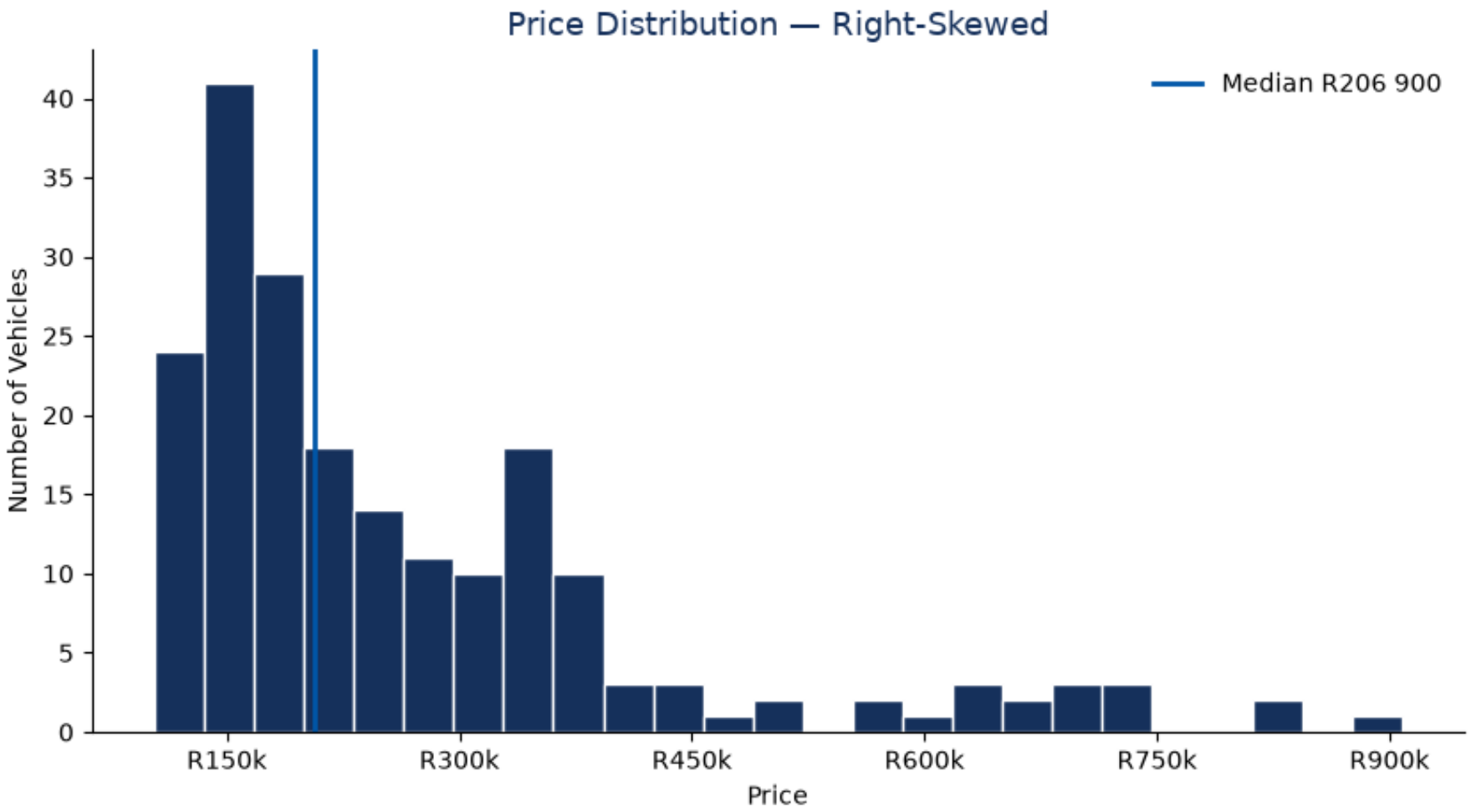
**Takeaway — Missing grouping fields are inferred from similar specifications where evidence is clear; unresolved rows are removed and the system remains decision support.**

# What the Data Tells Us

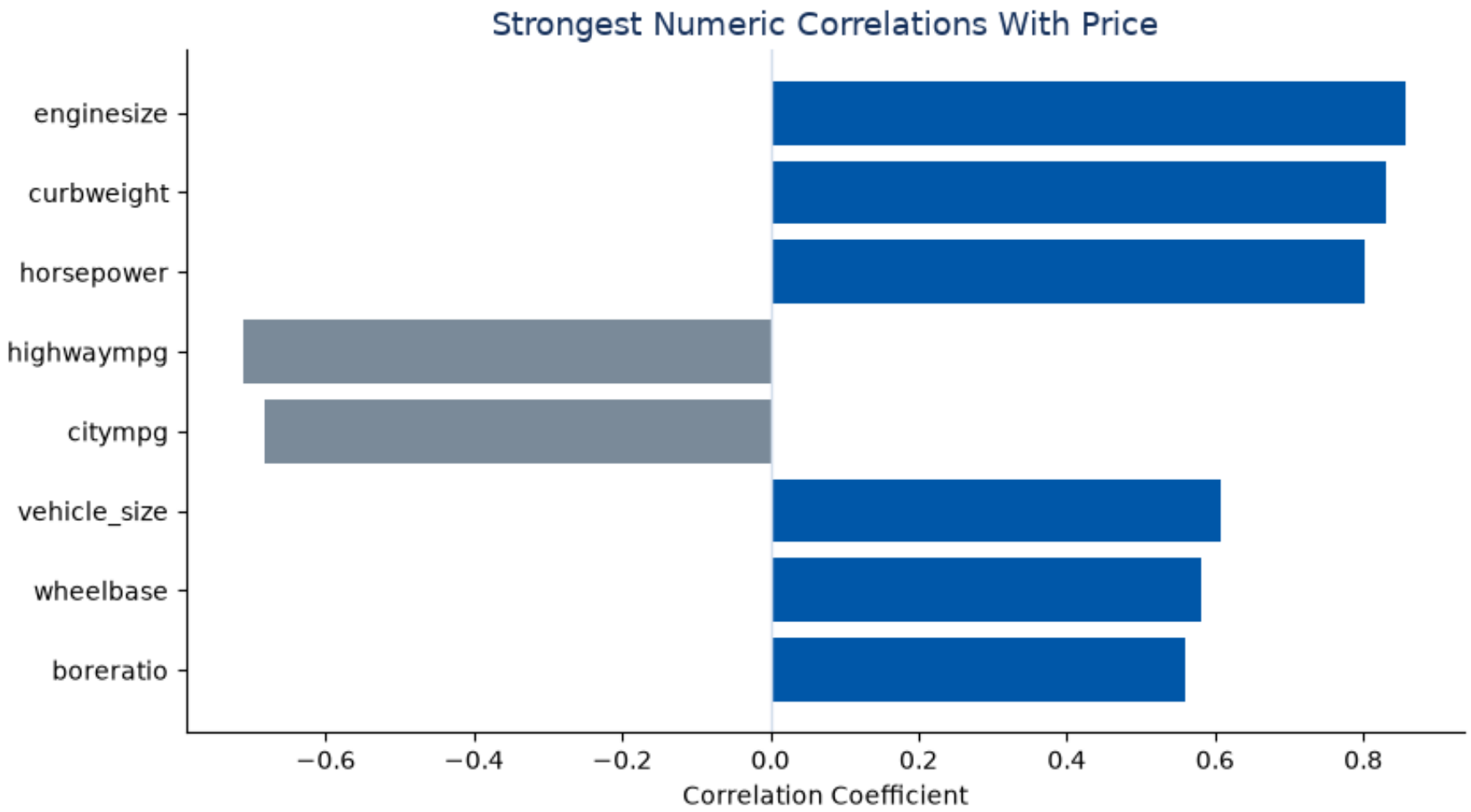
- Price is strongly right-skewed — most vehicles are affordable, a few premium
- Engine power, size and curb weight track price most closely
- Body style, fuel type and drive wheel shift price systematically

**Takeaway — Price is driven mainly by specification variables; model names are too sparse and make groups are too broad for reliable prediction by themselves.**

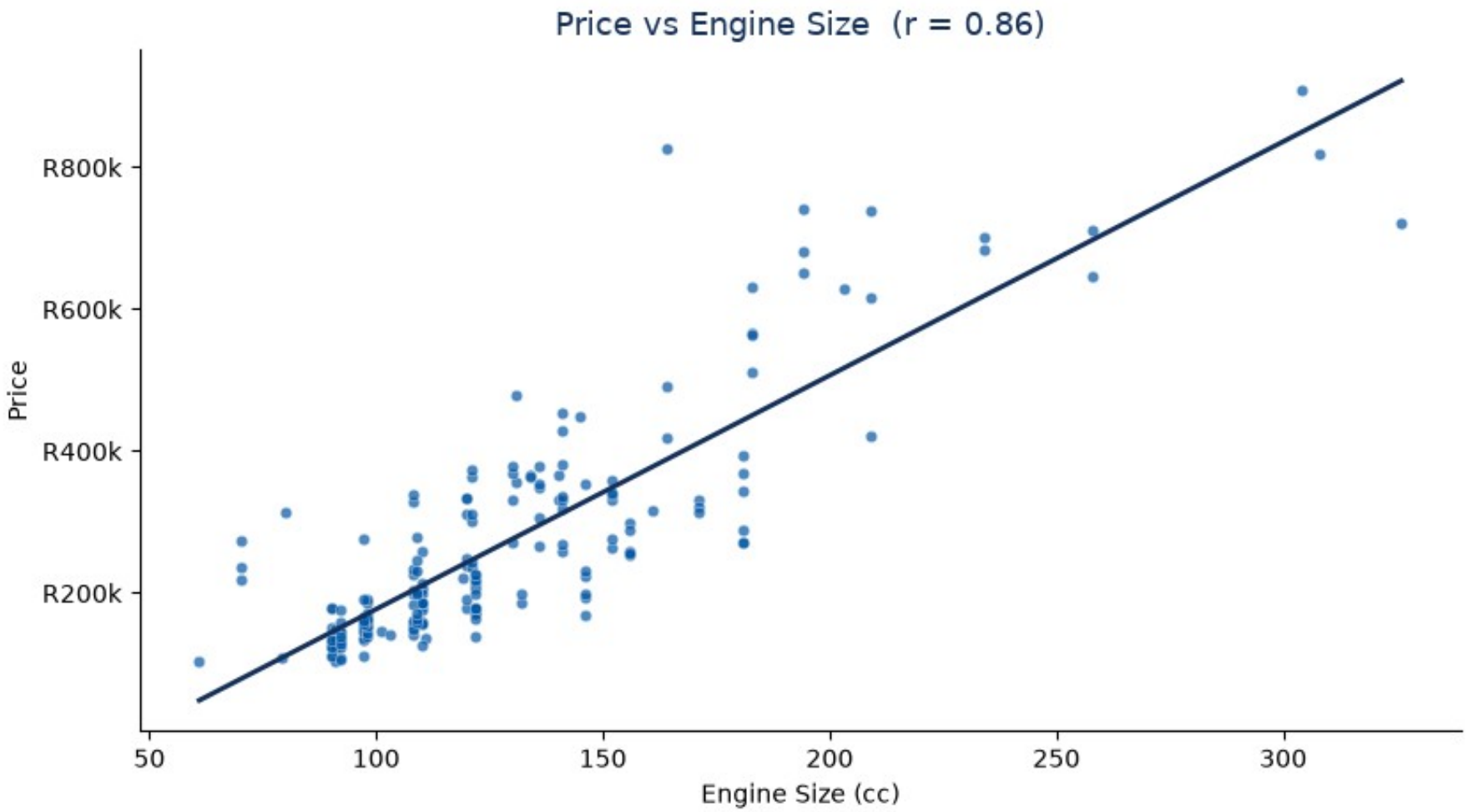
# Price Distribution



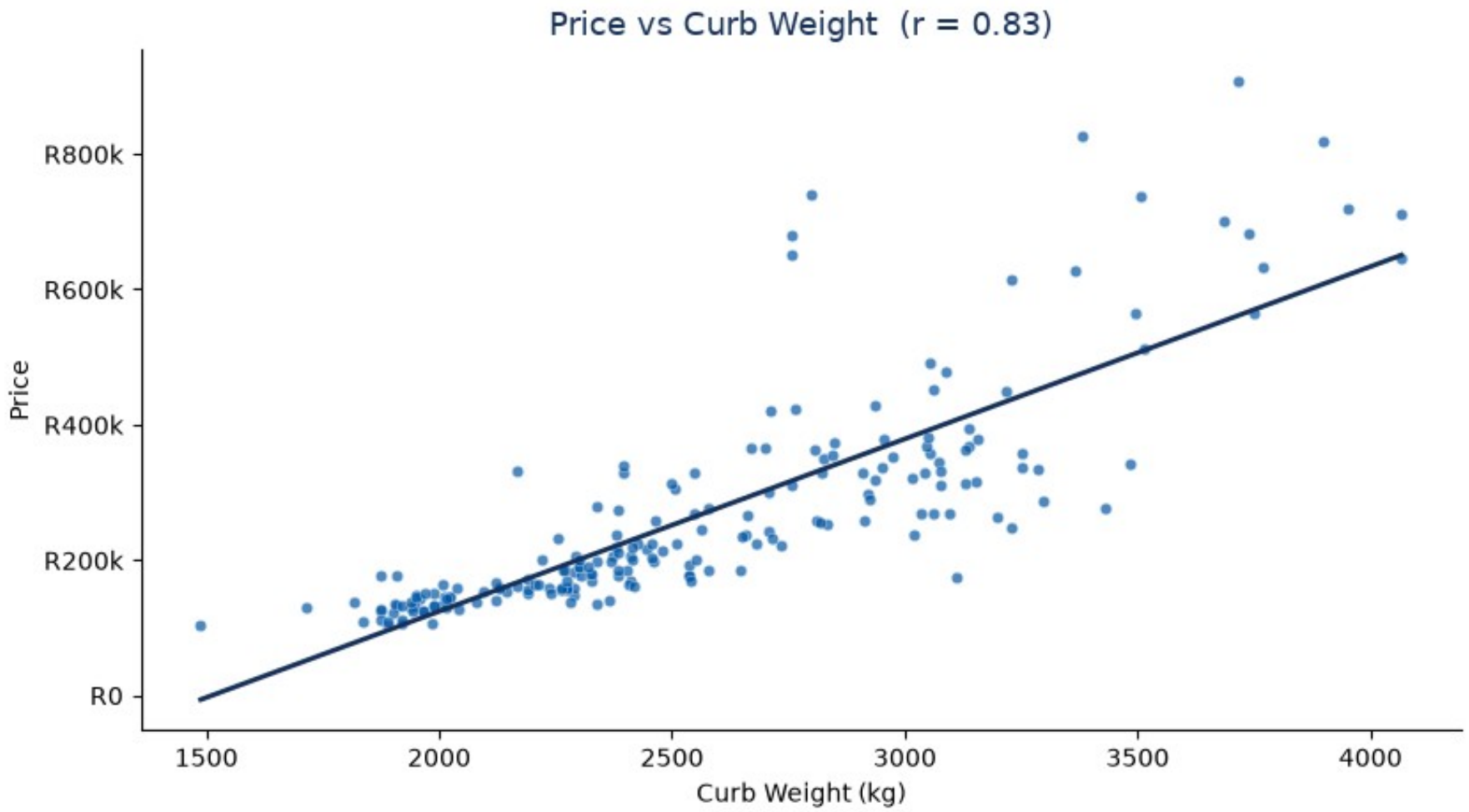
# Correlations With Price



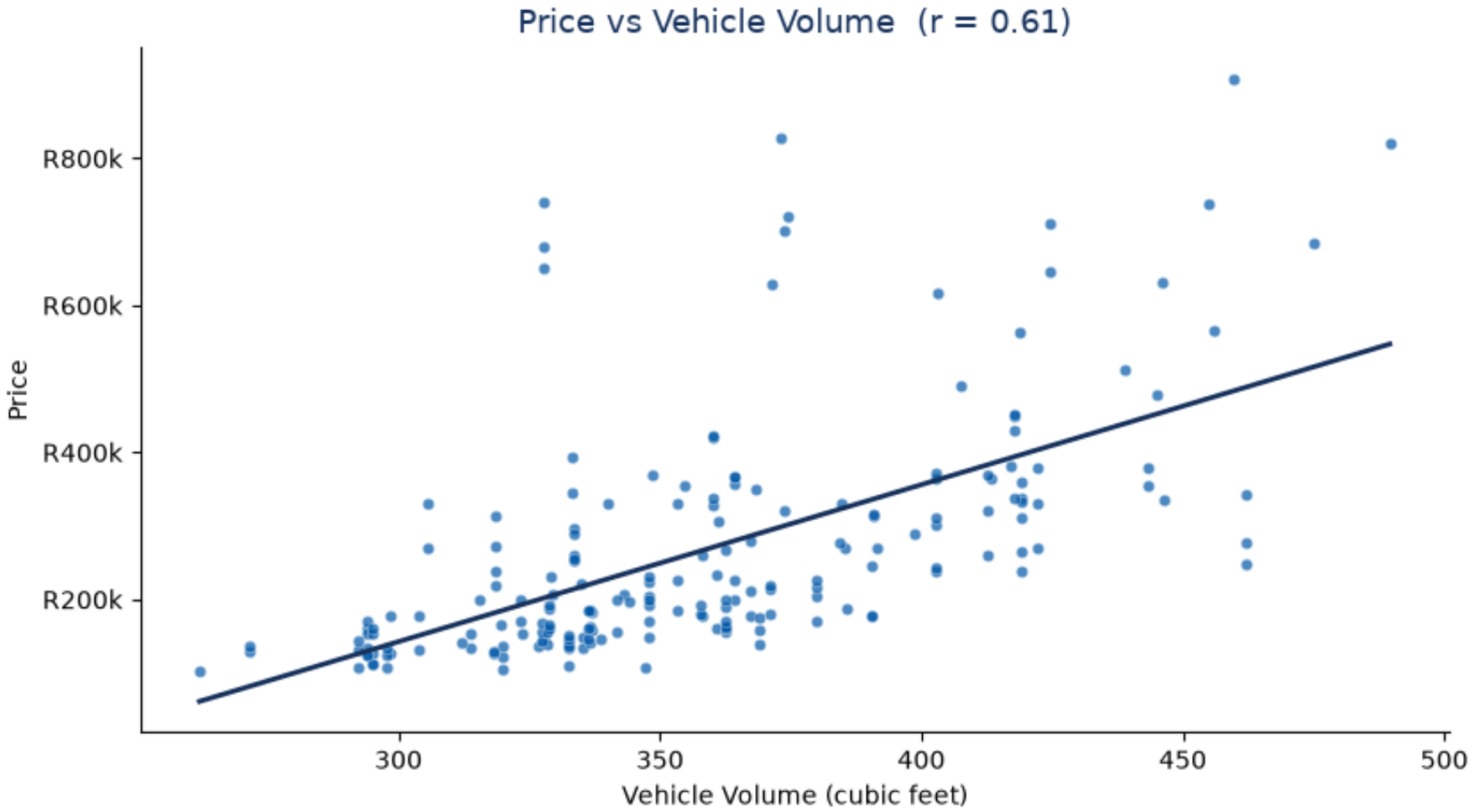
# Price vs Engine Size



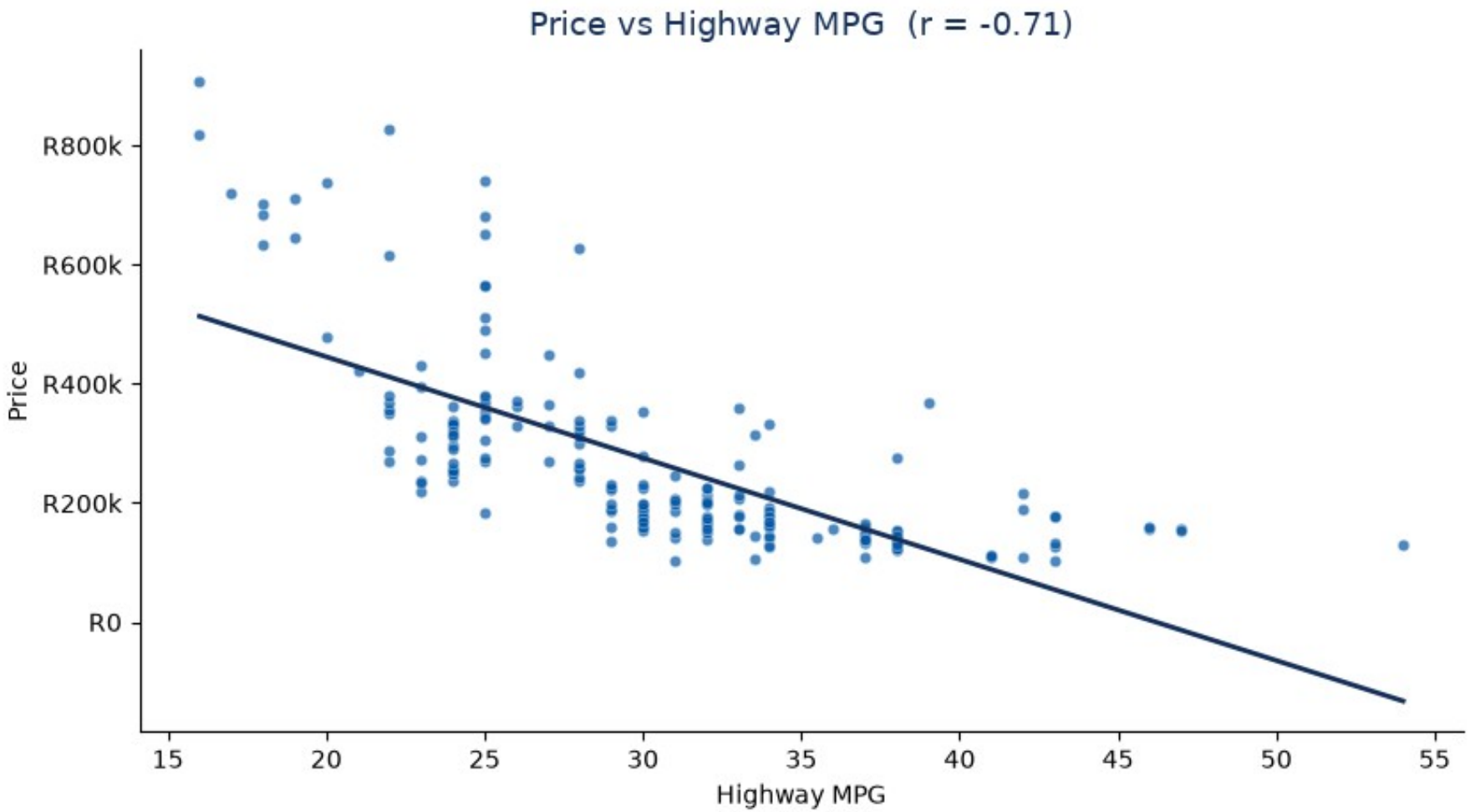
# Price vs Curb Weight



# Price vs Vehicle Volume



# Price vs Fuel Economy

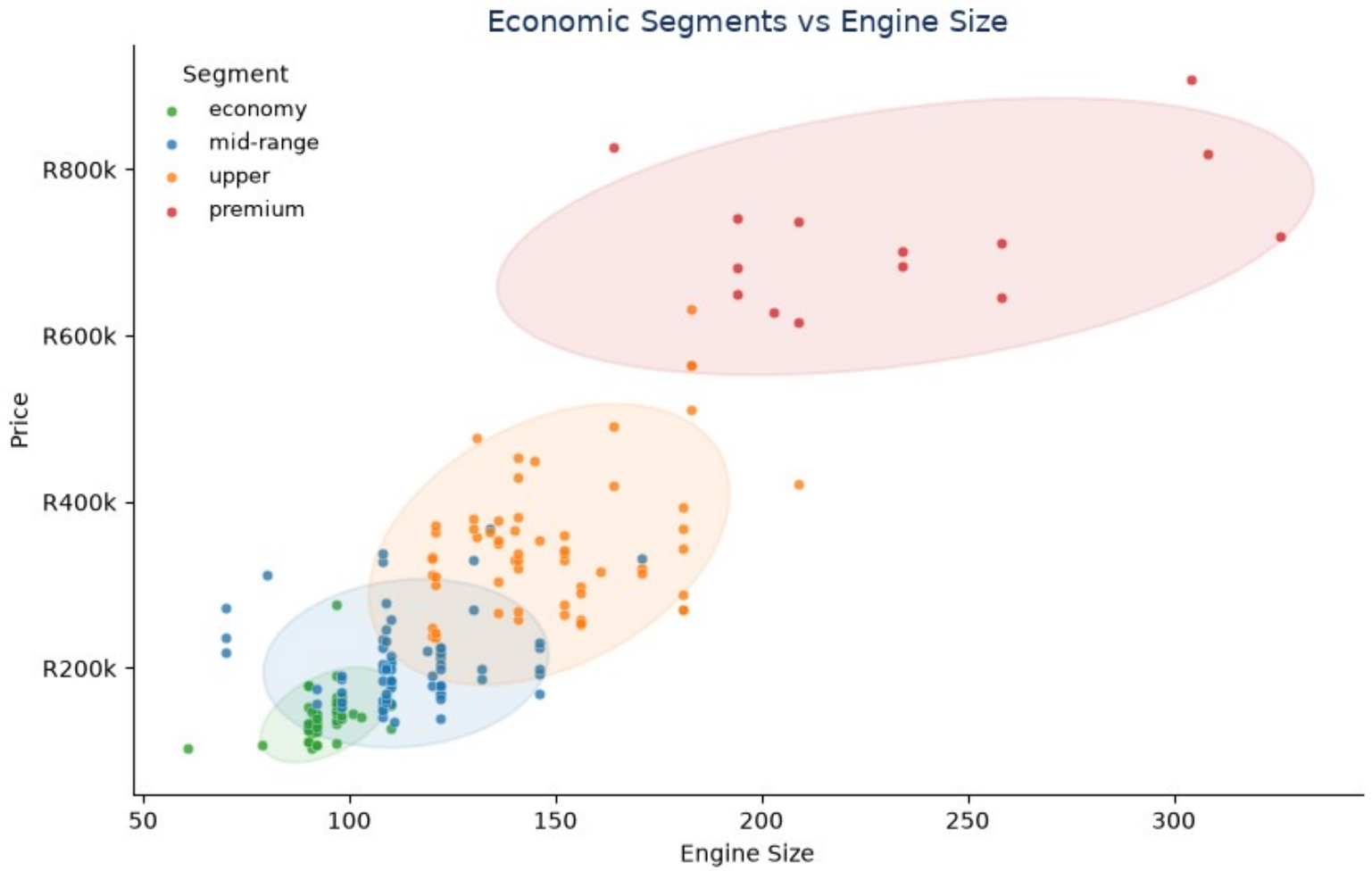


# Meaningful Vehicle Segments

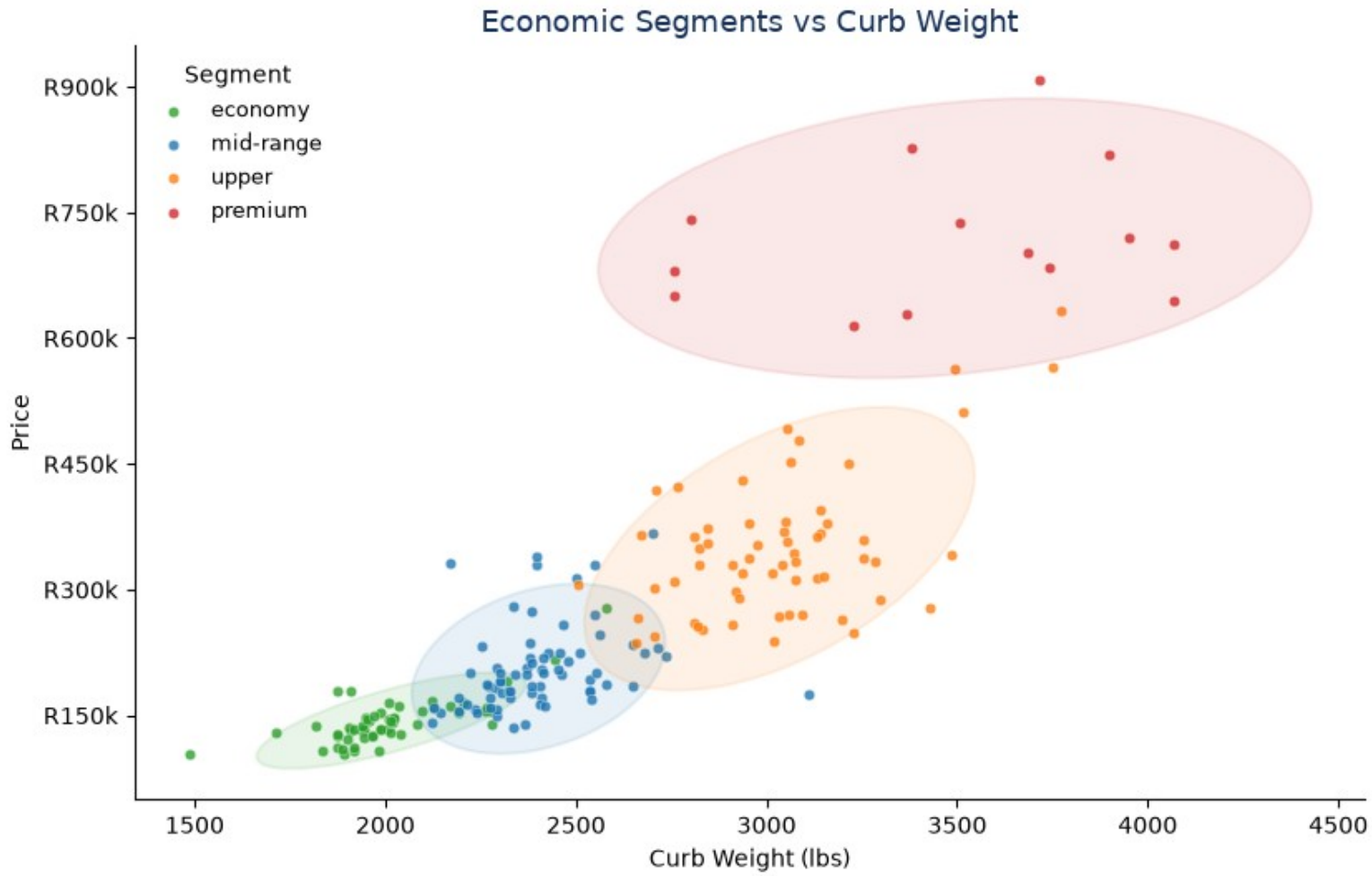
- Clustering on specification features reveals natural vehicle groups
- e.g. economy compacts, mainstream sedans, performance / premium
- Segments justified by the features that separate them

**Takeaway — Vehicles cluster into four price-ordered bands — economy, mid-range, upper and premium — driven by a distinct combination of price, engine size, power and fuel efficiency.**

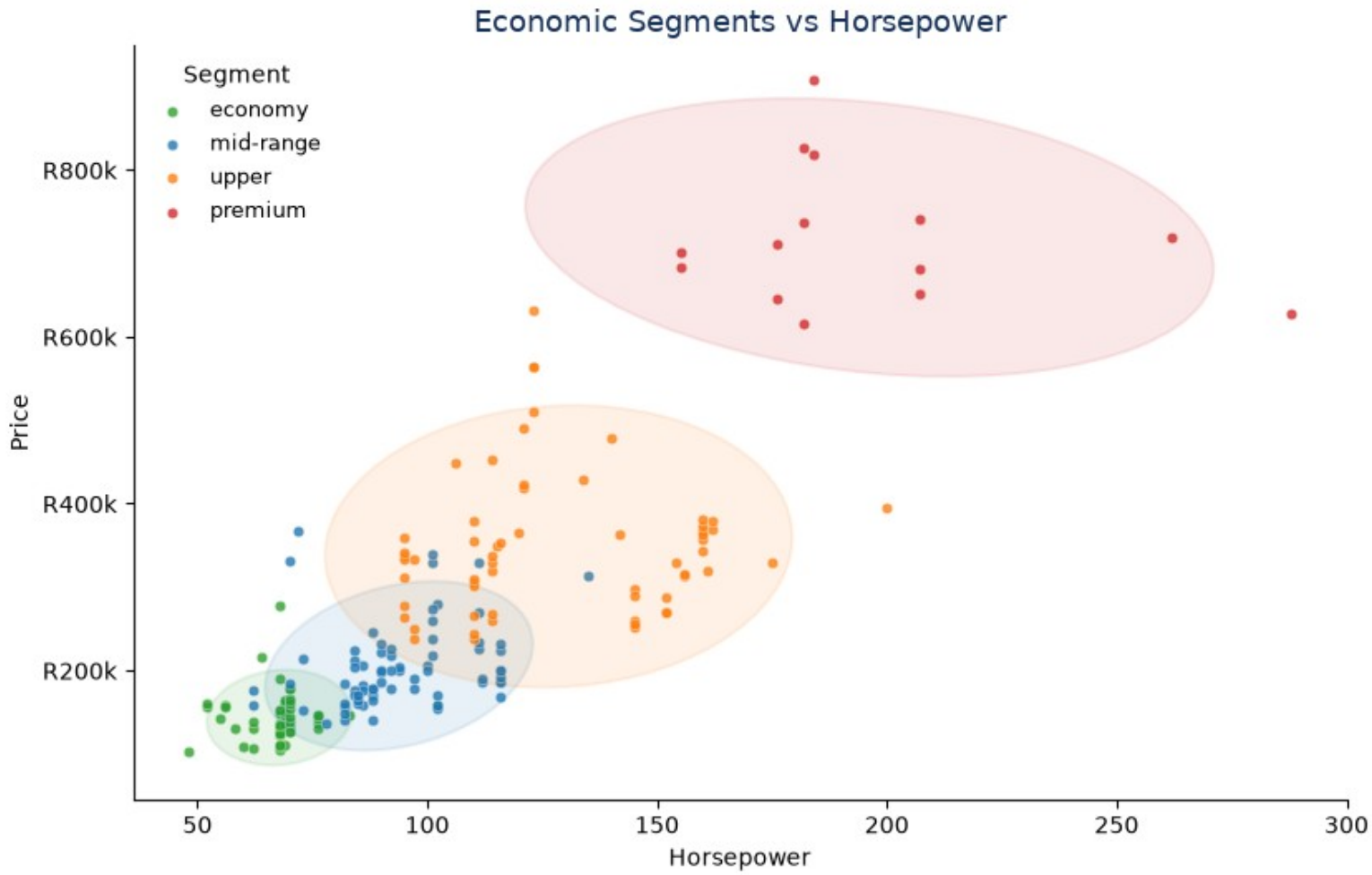
# Segments vs Engine Size



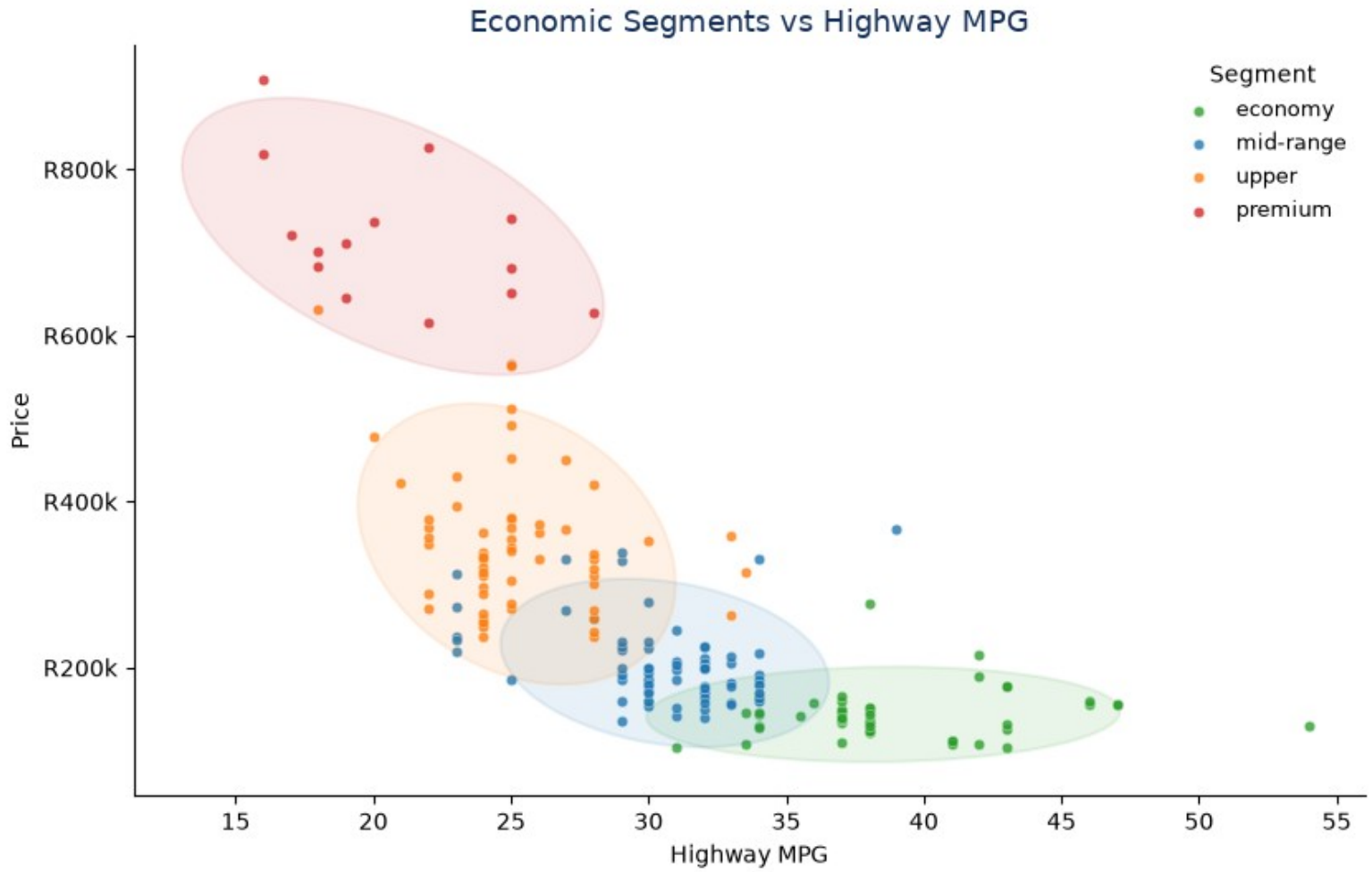
# Segments vs Curb Weight



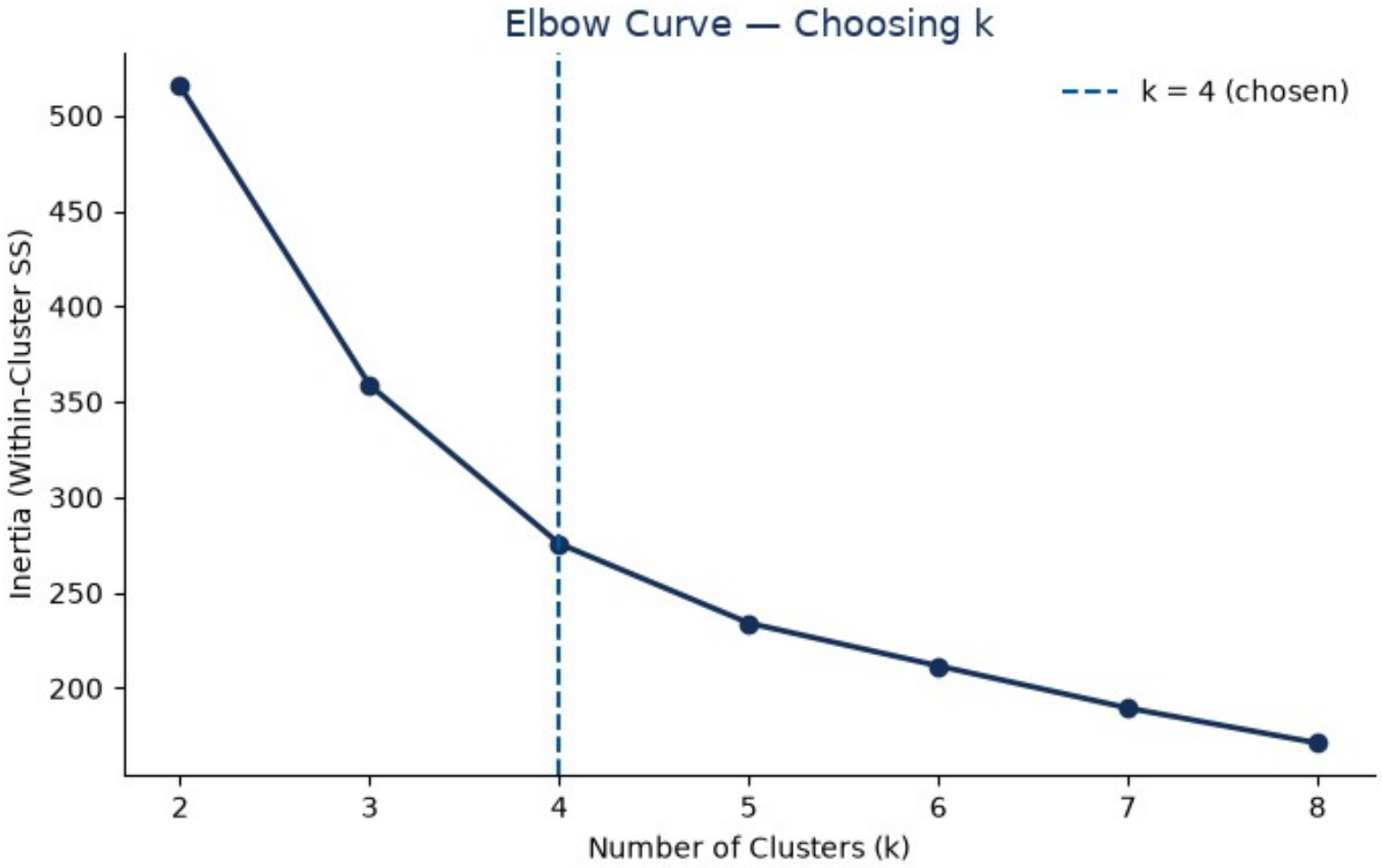
# Segments vs Horsepower



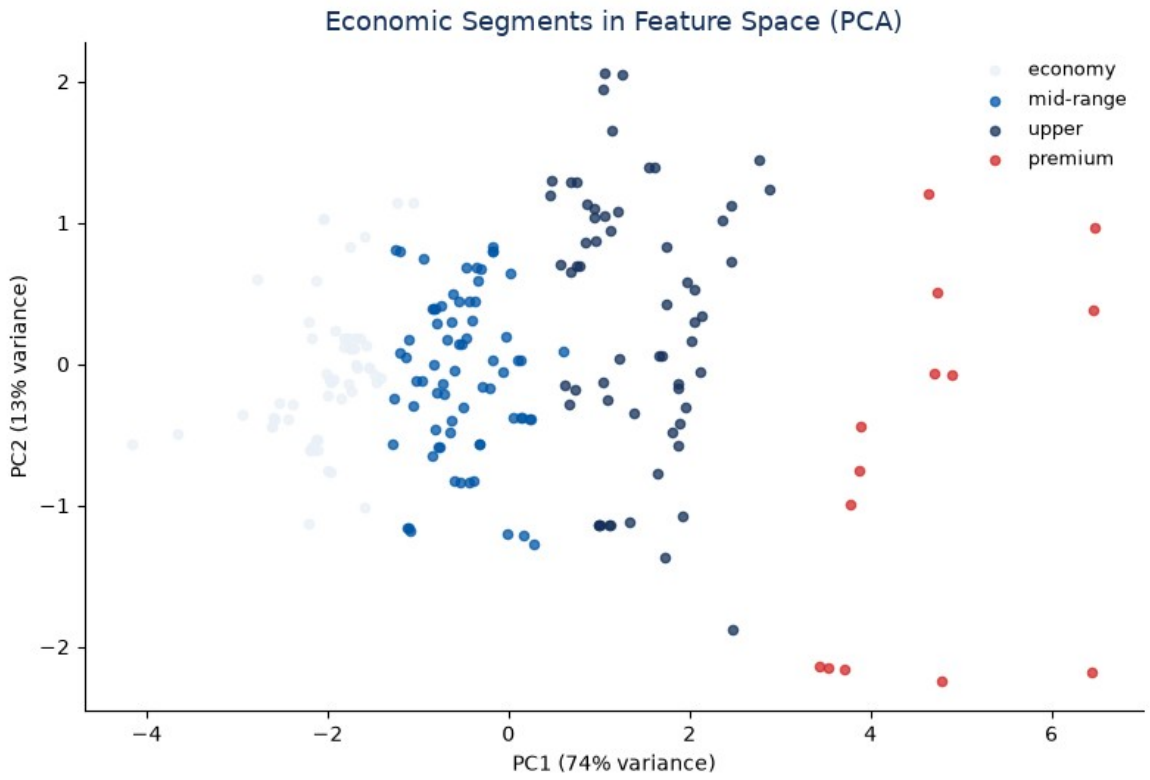
# Segments vs Highway MPG



# Choosing the Number of Clusters

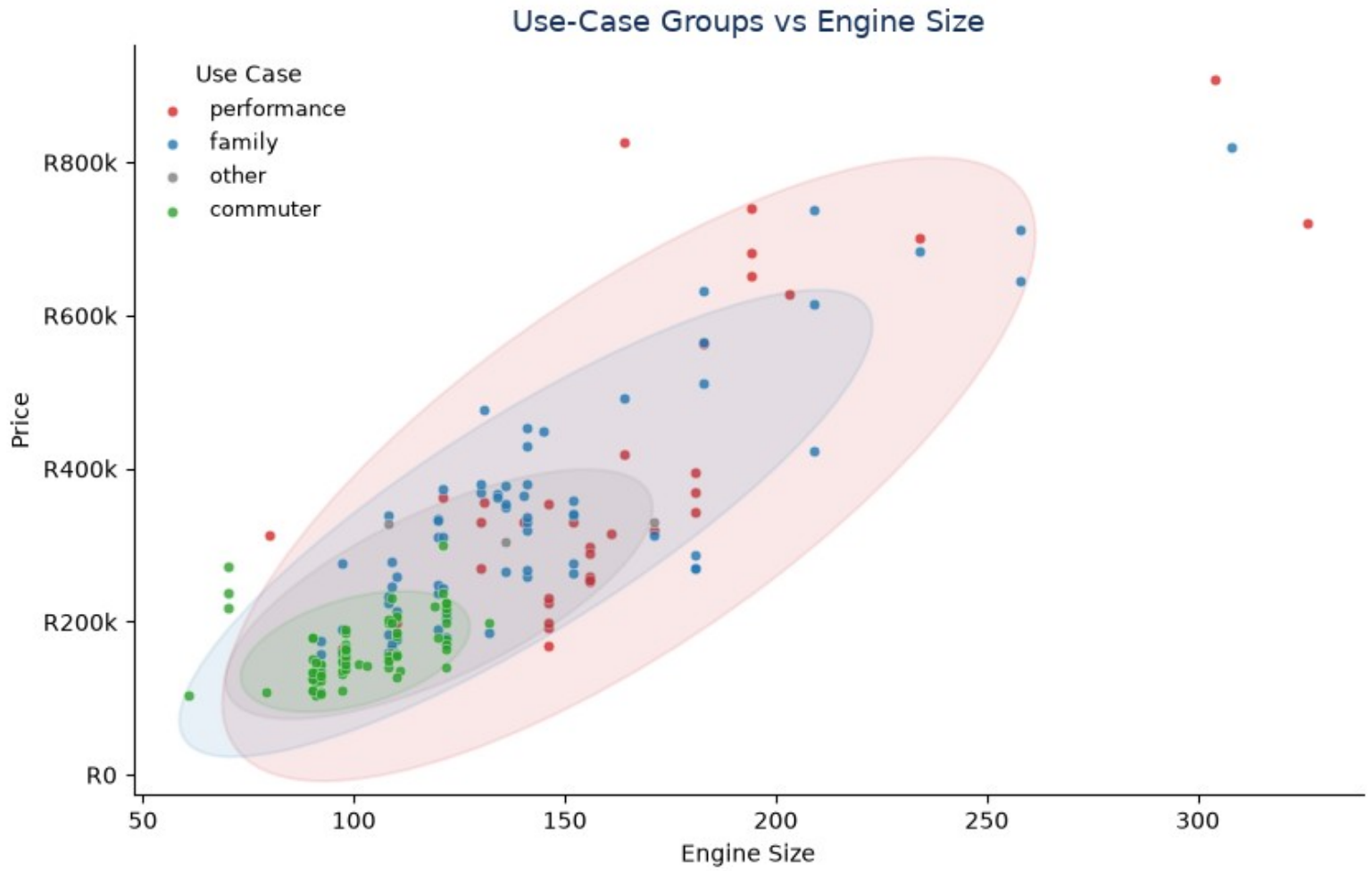


# Segment Structure in Feature Space (PCA)

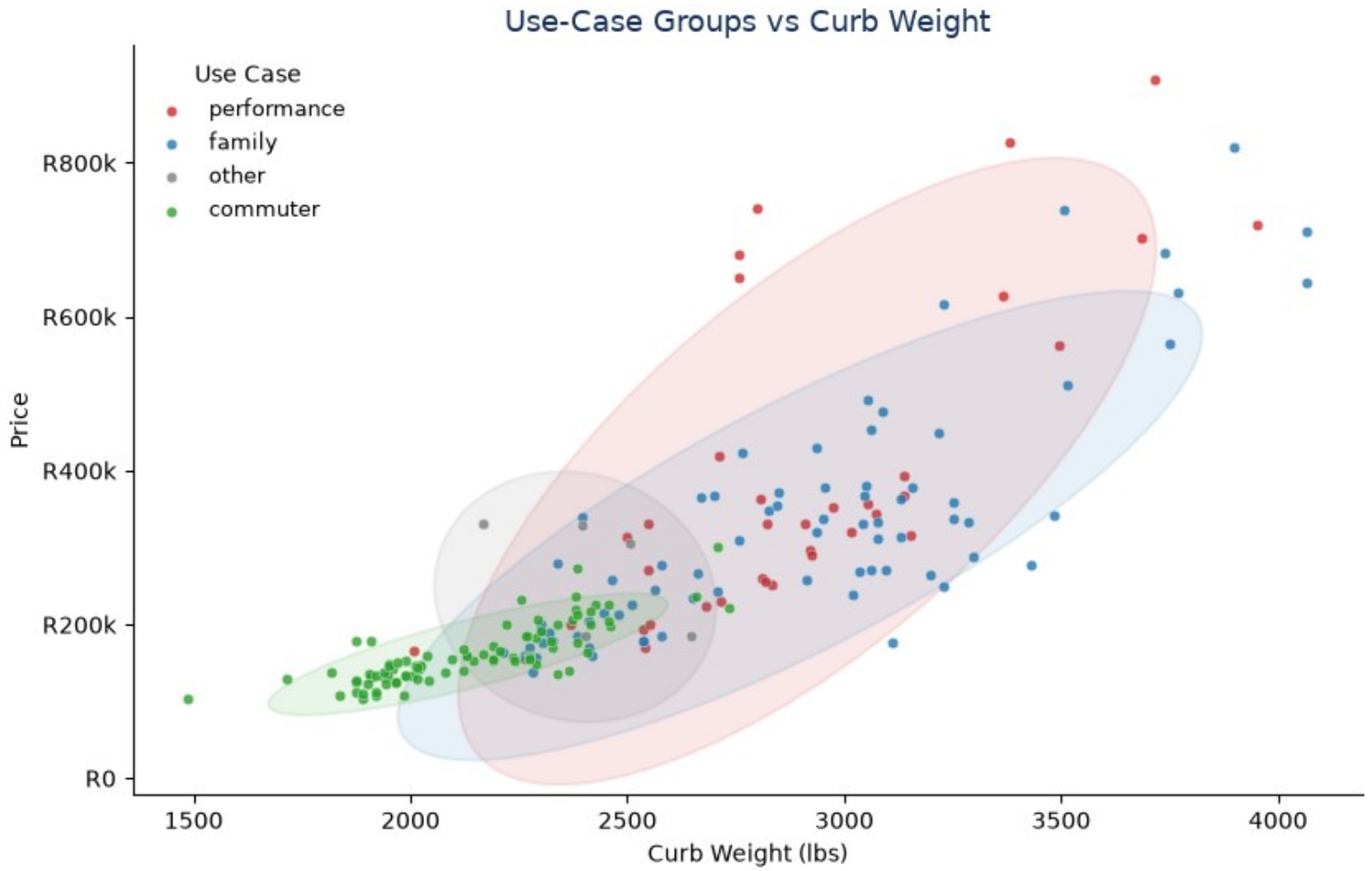



PCA (Principal Component Analysis) — reduces the five-dimensional feature space (price, horsepower, engine size, vehicle size, city MPG) to two orthogonal axes (PC1, PC2) that together capture the most variance. Points close together share similar feature profiles; well-separated clusters confirm the segments are genuinely distinct across all five dimensions, not just in price alone.

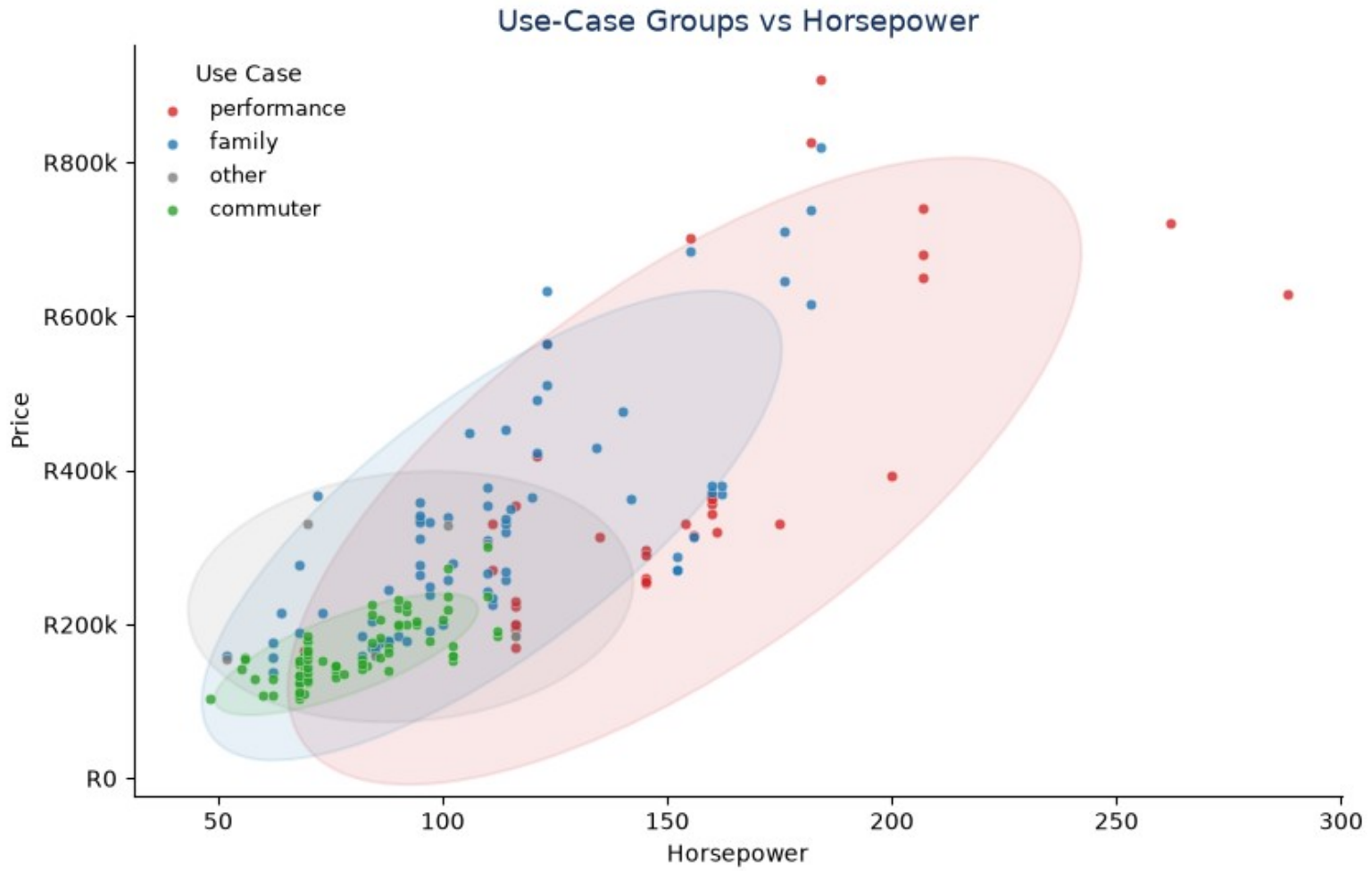
# Use-Case Groups vs Engine Size



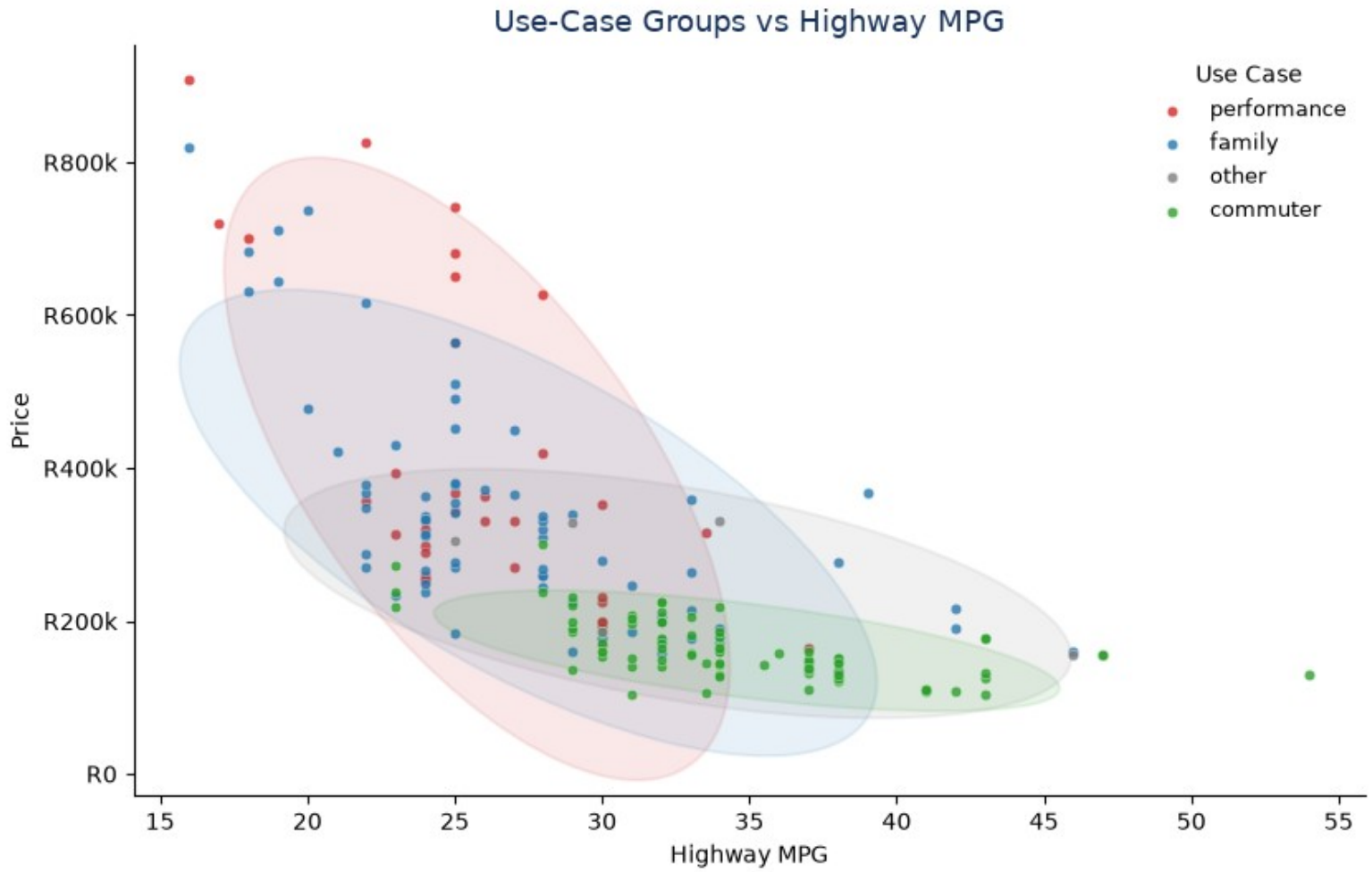
# Use-Case Groups vs Curb Weight



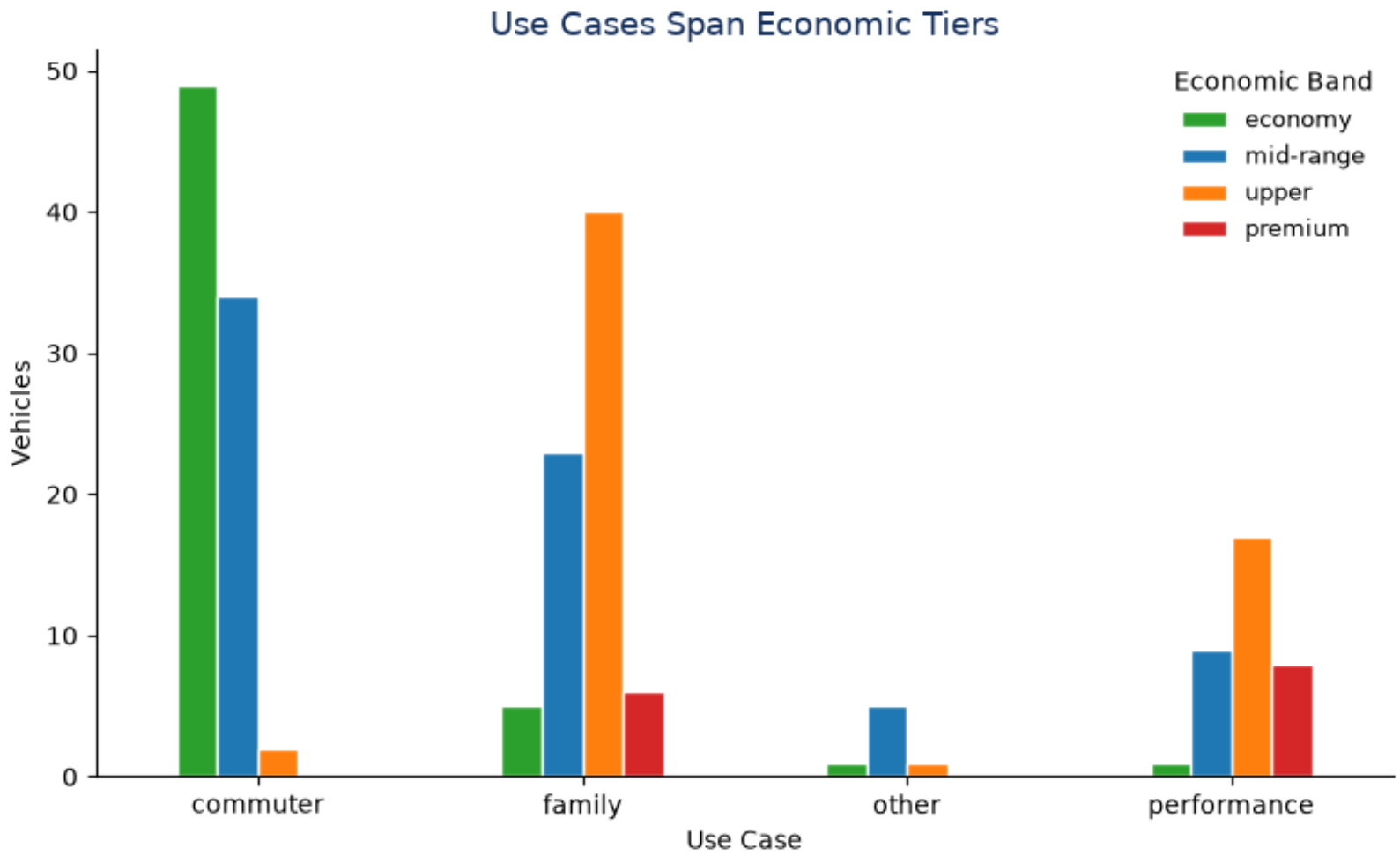
# Use-Case Groups vs Horsepower



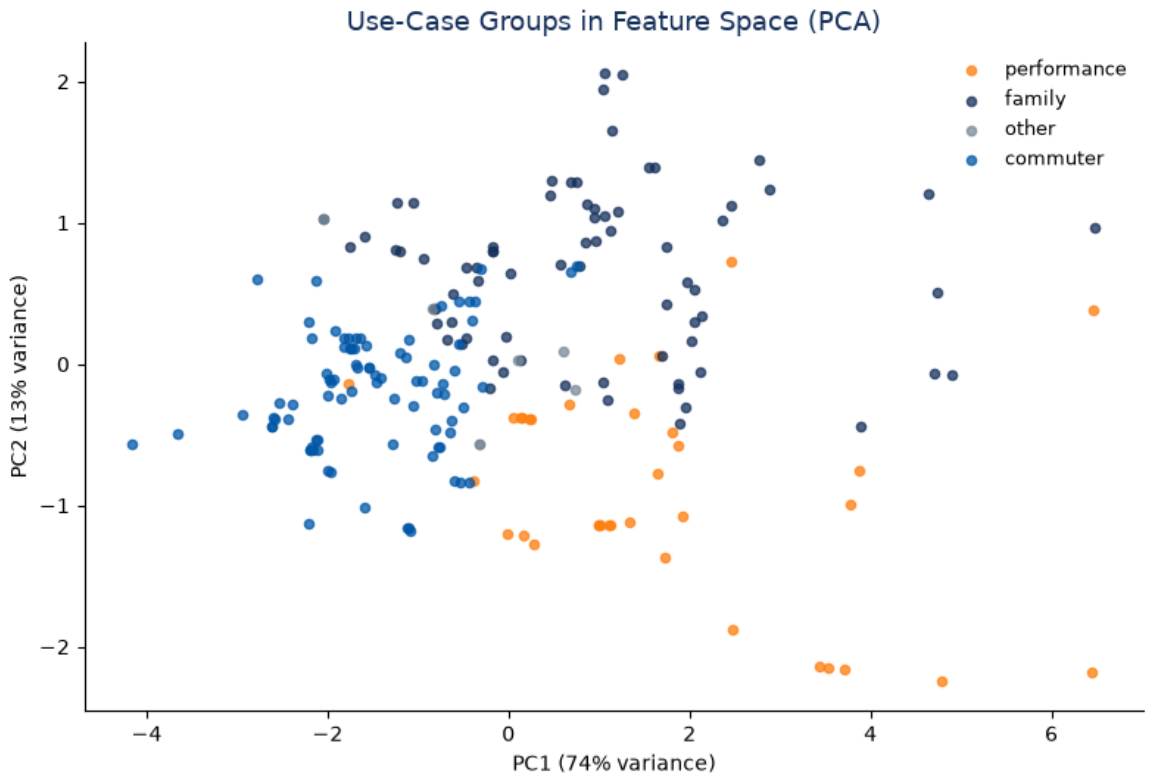
# Use-Case Groups vs Highway MPG



# Use Cases Span Economic Tiers



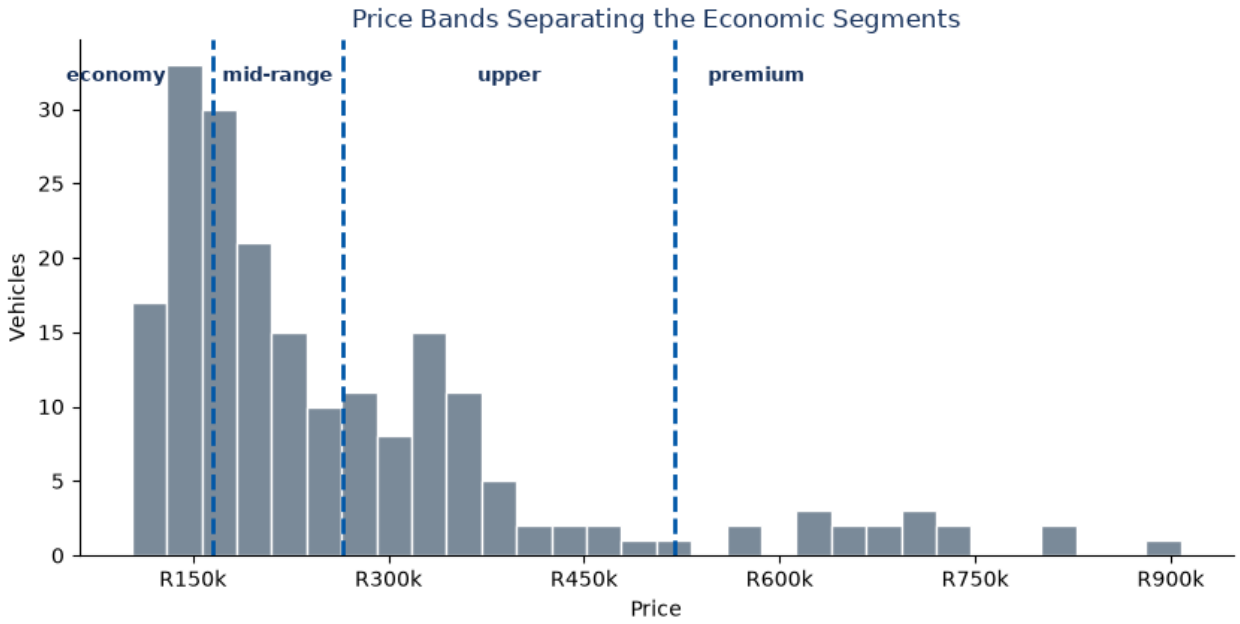
# Use-Case Structure in Feature Space




PCA (Principal Component Analysis) — the axes here are the same economic feature space (price, horsepower, engine size, vehicle size, city MPG) used for the segment clustering. Projecting use-case groups onto this space shows how far apart they are in value terms: overlapping clouds mean two use cases occupy similar price–spec territory; separation means they do not. PC1 and PC2 are linear combinations of all five features that maximise the explained variance.

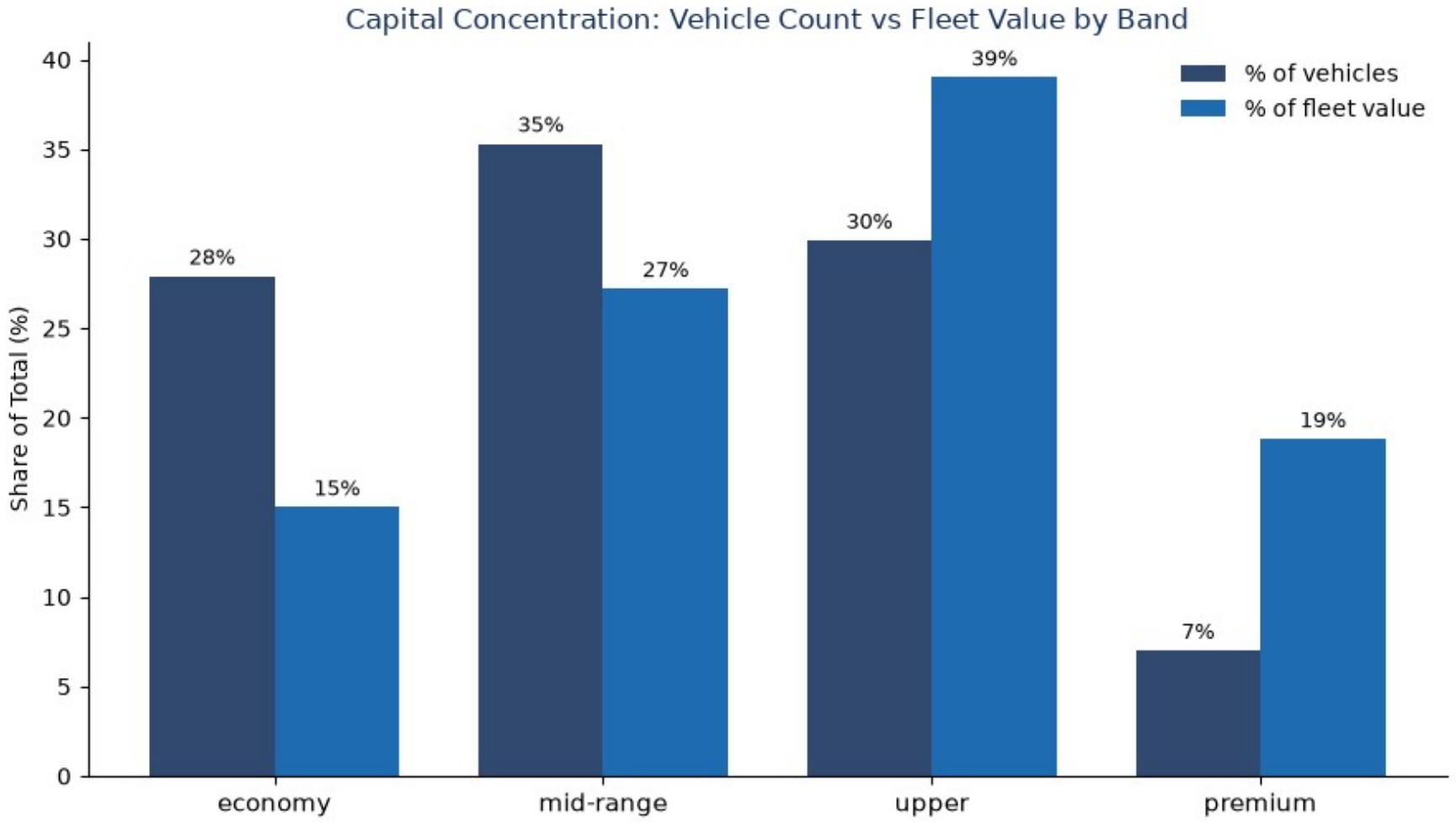
# Valuation & Decision Categories

- Price bands: budget · mid-range · high-value · premium
- Decision bands: automatic valuation · manual review · high-risk review
- Checked for balance, usefulness and realism — supports pricing & negotiation




**Takeaway — Four price bands (economy → premium) drawn from the clustering give an instant price-only classification; sell each vehicle on its top price-driving specs - engine size, weight and power.**

# Capital Concentration by Band

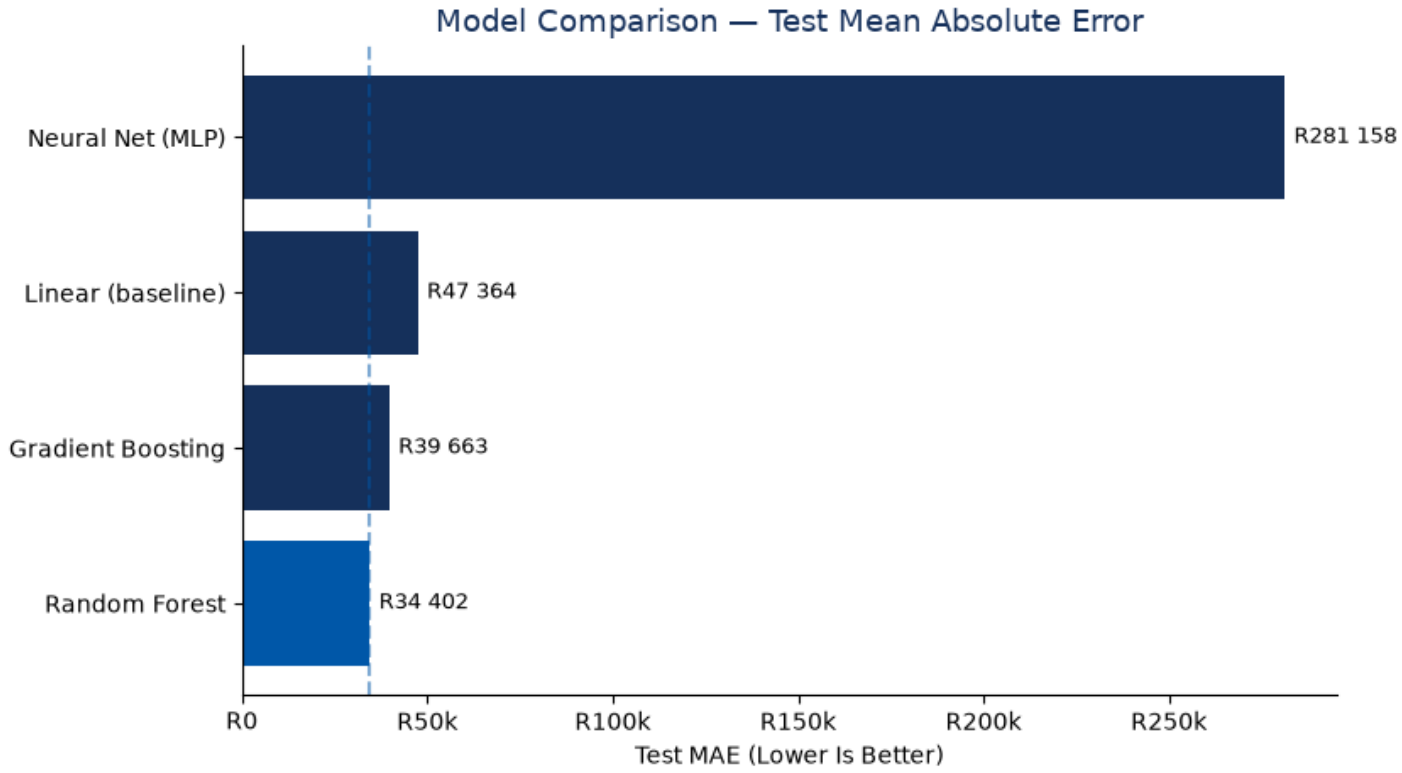


# Modelling Approach

- Framed as a regression problem (with optional band classification)
- Features: encoded categoricals, scaled numerics, make extracted from CarName
- Compared ≥3 models: a simple baseline vs flexible non-linear models
- Chosen on accuracy, interpretability, generalisation and business fit

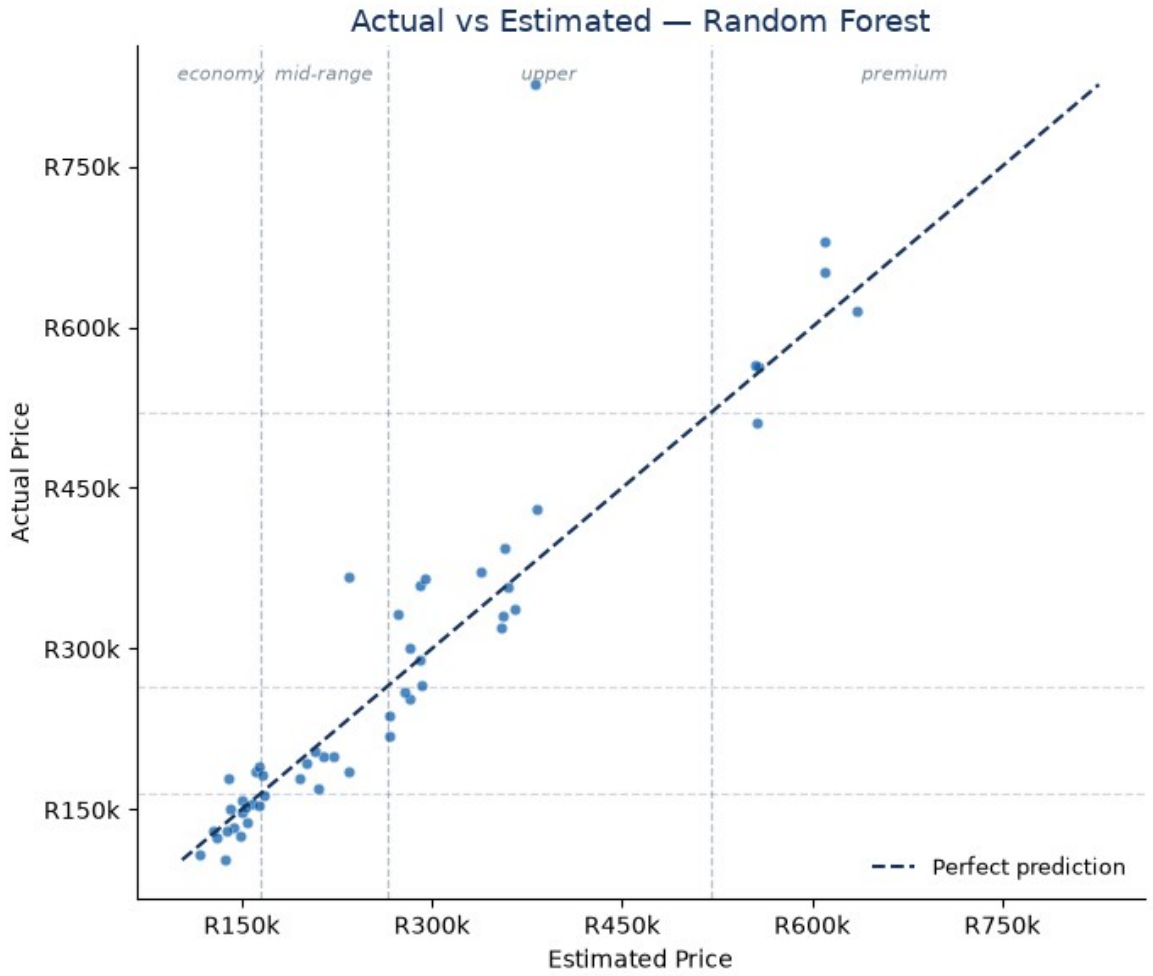
**Takeaway — Random Forest gives the lowest error (MAE ~R34k, R2 0.82); the neural network fails on so little data, and a linear model is nearly as good and the simplest.**

# Model Comparison — Test MAE




MLP (Multi-Layer Perceptron) — a feedforward neural network that learns non-linear relationships by stacking layers of weighted connections with activation functions; trained here with two hidden layers (64 → 32 units) and early stopping to limit over-fitting on the small (~150 row) training set.

# Predicted vs Actual — Best Model

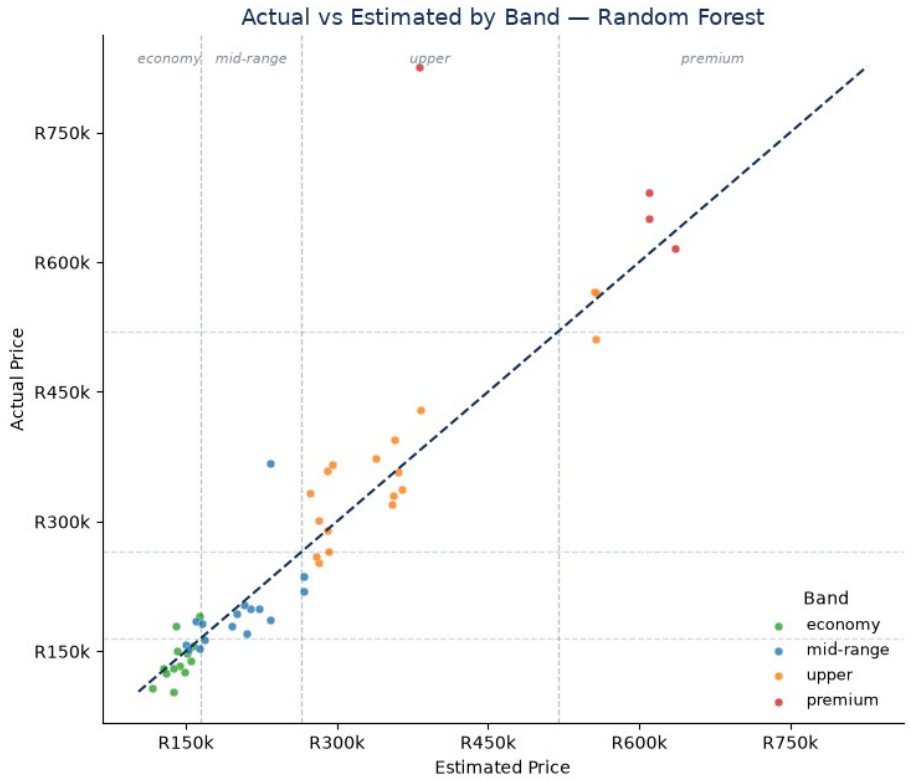


# How Accurate Is It?

- Typical error about R34 000 (~11% of price); R-squared 0.82
- 49% of cars within R20 000; correct decision band 78% of the time
- Acceptable threshold: within +/-10% of the true price (percentage, not fixed rand)
- Accurate enough for a first estimate, not for a final unaided price

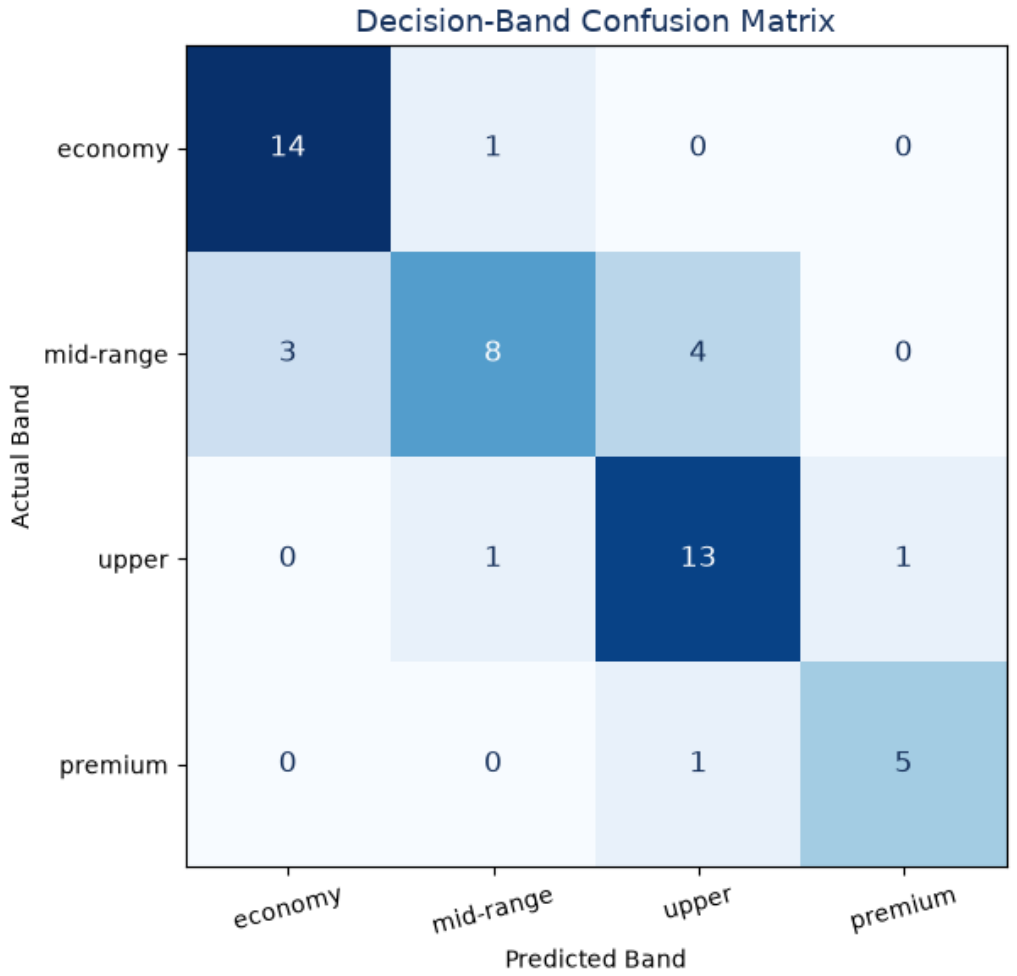
**Takeaway — Good enough to start the negotiation, not to end it.**

# Predicted vs Actual Prices




MAE (Mean Absolute Error) — average absolute rand difference between predicted and actual price; directly interpretable in the same currency units as the target. RMSE (Root Mean Squared Error) — like MAE but squares errors before averaging, so large misses are penalised more heavily than small ones. MAPE (Mean Absolute Percentage Error) — average error expressed as a percentage of the actual price, useful for comparing accuracy across different price levels.

# Decision-Band Assignment Accuracy

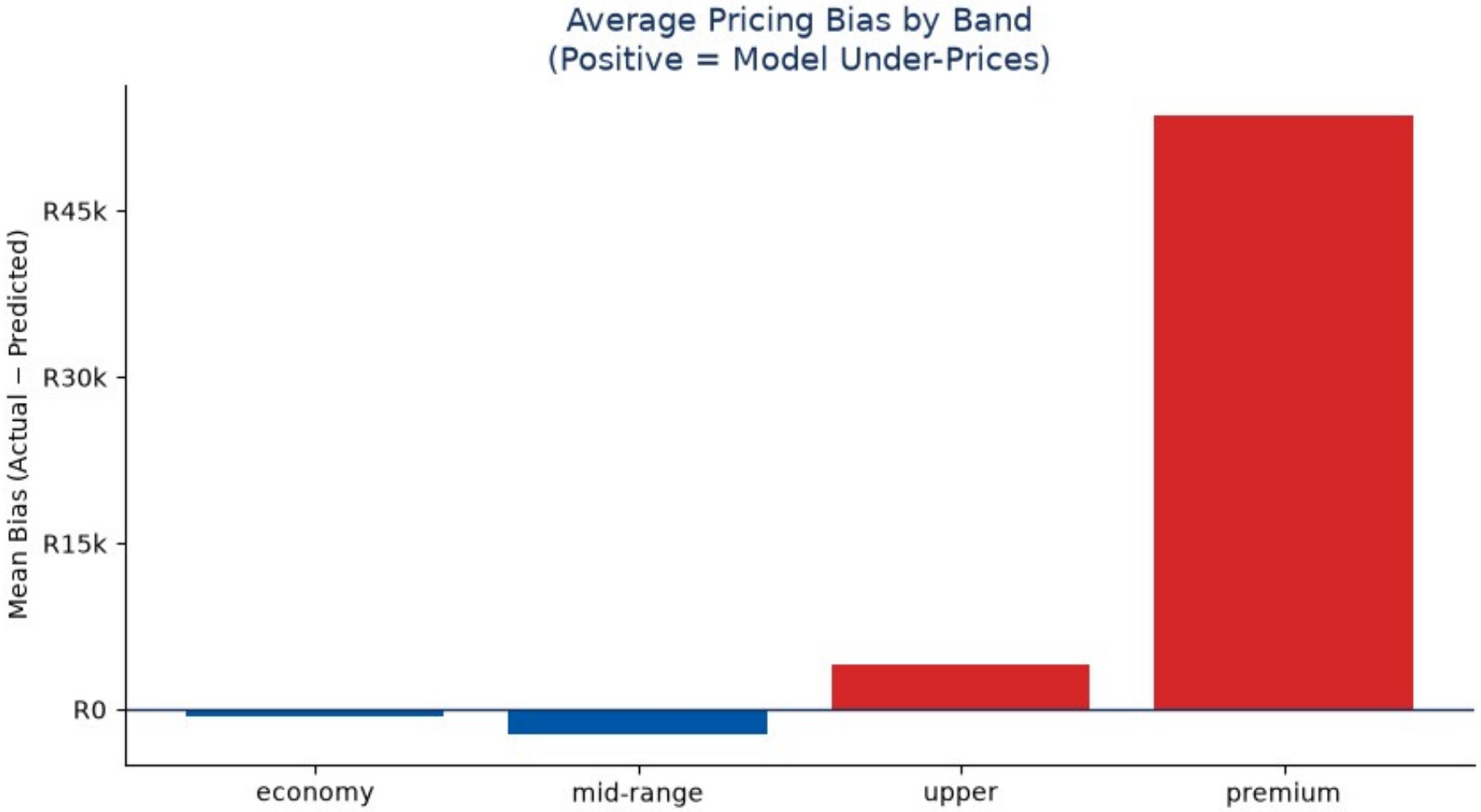


# Where It Works — and Where It Doesn't

- Under-pricing loses margin - premium is short about R92k on average
- Over-pricing leaves stock sitting and ties up capital
- ~22% of cars land in an adjacent band → mis-set strategy
- No mileage / age / condition: never auto-price premium, rare or out-of-range cars

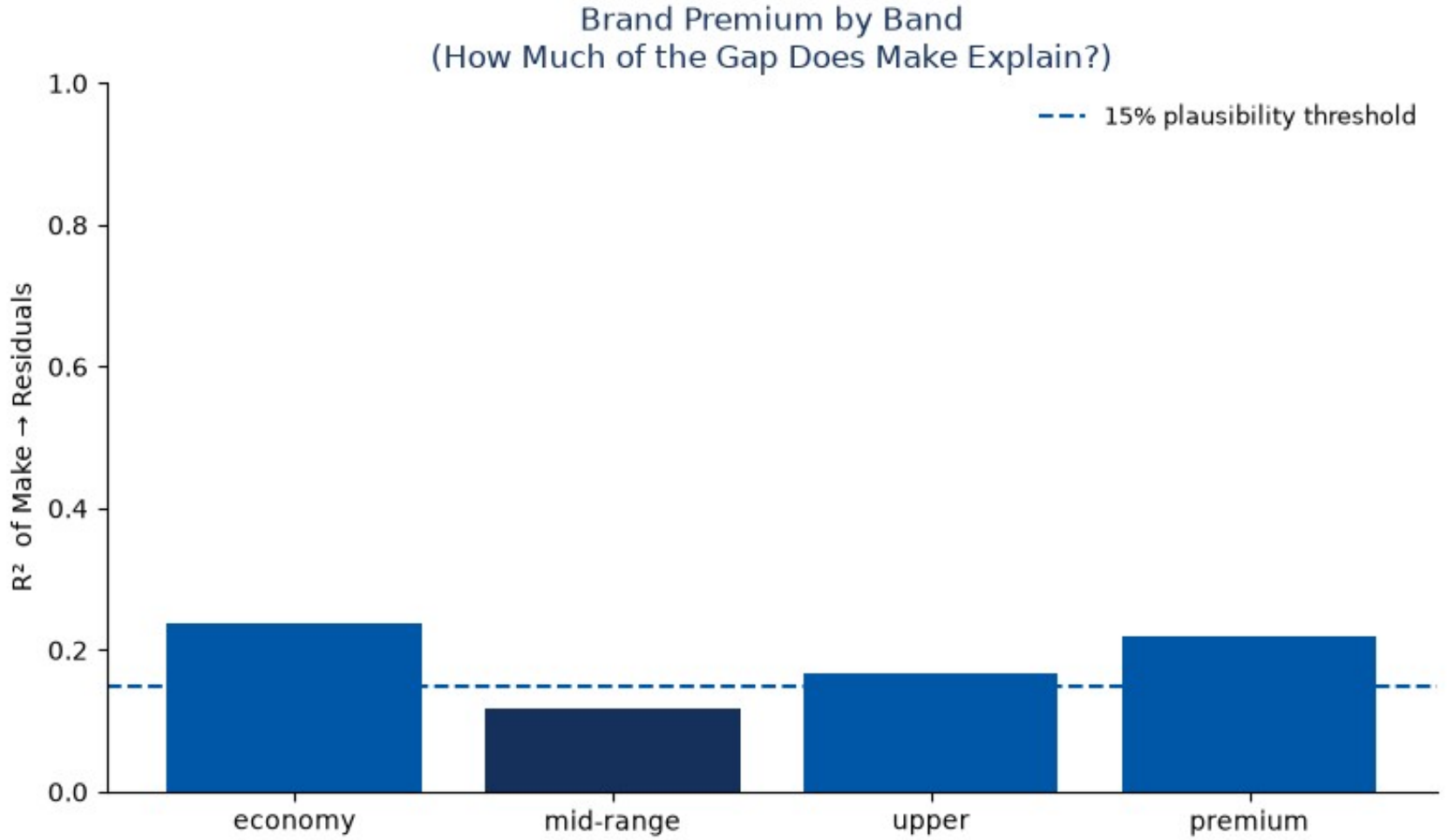
**Takeaway — The model prices the specification, not the individual car - keep humans on the high-stakes cases.**

# Average Pricing Bias by Band

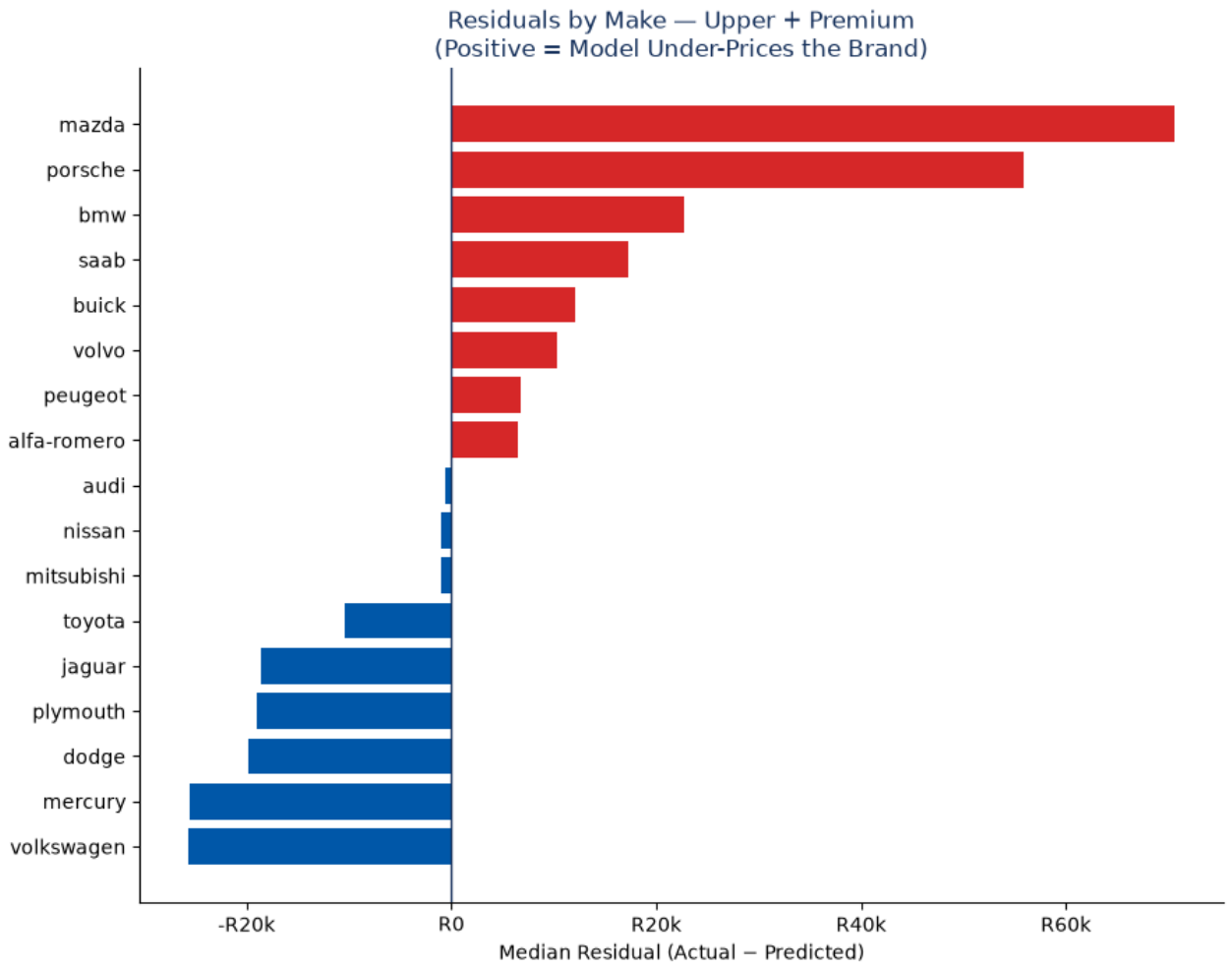


# Residuals Across the Price Range

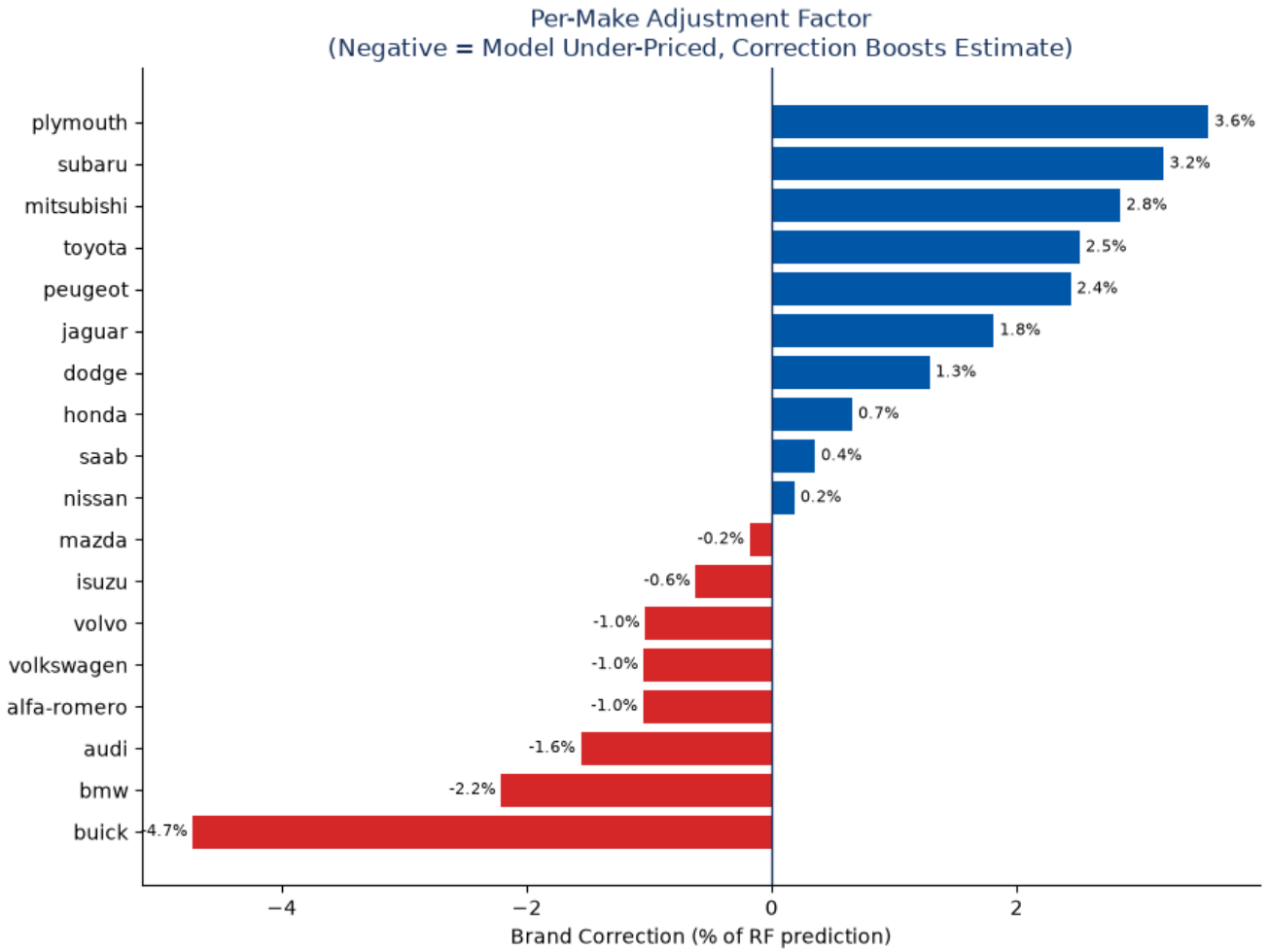
# Brand Premium by Segment



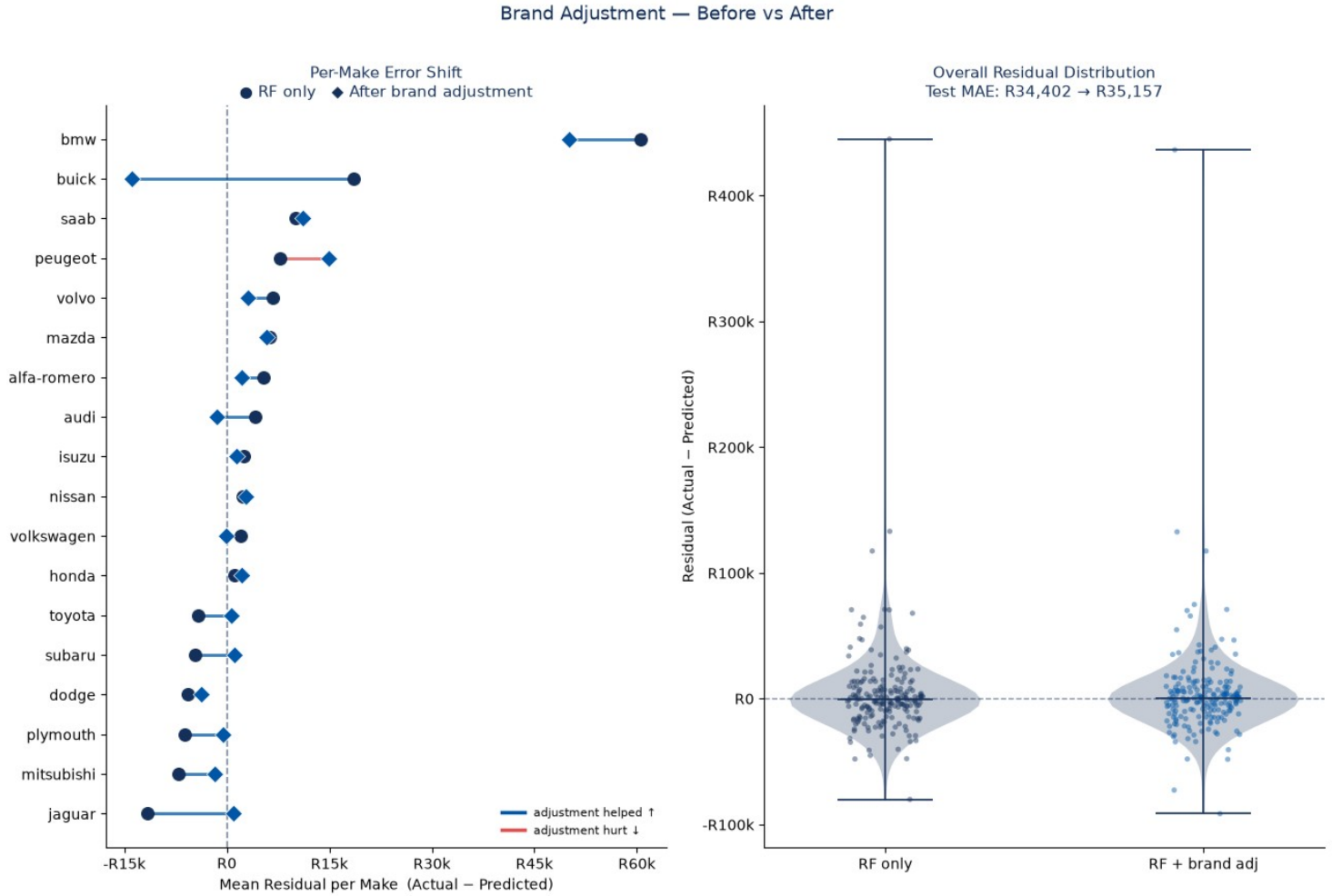
# Residuals by Make — Upper & Premium



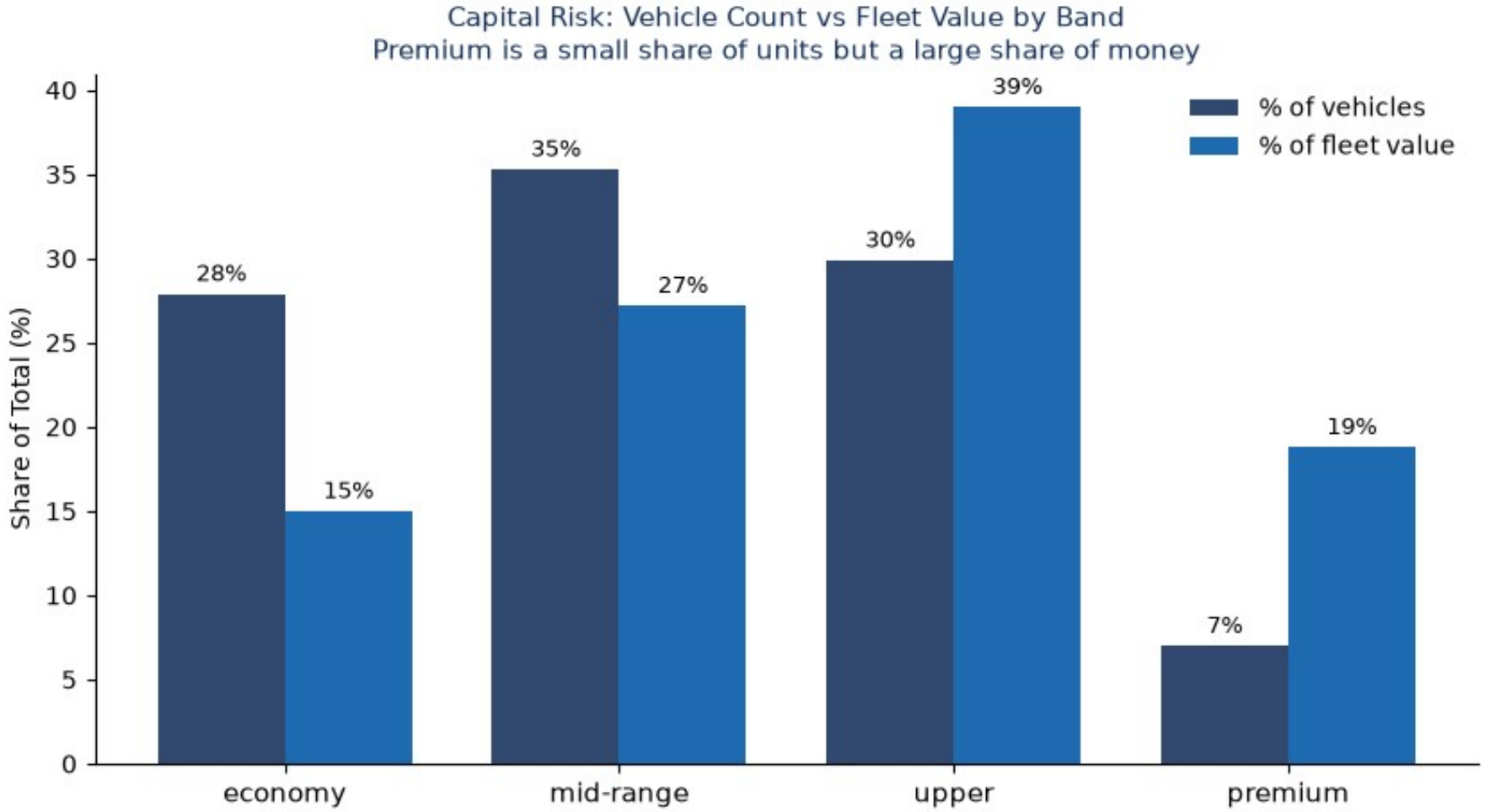
# Per-Make Percentage Correction



# Brand-Adjusted Model — Before vs After



# Capital Concentration — The Risk Behind the Rules



# How Mutuka Should Use the Model

- Reliable (economy / mid / upper, in-range) → automated preliminary estimate
- Unusual / rare / high predicted error / out-of-range → manual review
- Premium & high-value → physical inspection before a final offer

**Takeaway — Automate where the model is accurate; escalate where it is not.**

# Recommendation

- Adopt as decision support / partial automation - not full automation yet
- Auto-value the reliable majority; route premium / rare / out-of-range to review
- Next: add mileage, age, condition & history; log time-to-sell; then re-evaluate

**Takeaway — Partial automation now; full automation once the missing real-world data is collected.**


Thank you

# Questions?

*Mutuka Automotive · Vehicle Valuation & Decision Support*