# DiscountMate – Smart Substitution

## **Sprint 1 – Data Preparation & Feature Engineering (Short Doc)**

### 1) Goal (what Sprint 1 delivers)

Prepare a **clean, model-ready dataset** so Sprint 2 can build substitution logic. Focus on:

* Reliable **brand identity** (brand\_clean, brand\_confidence, brand\_tier)
* Consistent **unit pricing** (price\_per\_unit)
* **Size compatibility** within categories (size\_band)

### **2) Inputs (expected columns)**

product\_code, name, category, subcategory, brand, price, item\_size, unit\_type (g/ml/each), price\_per\_unit, tags

If tags arrive as a stringified list, we parse them back to lists.

### **3) How to run (quick)**

1. Open the Sprint-1 notebook.
2. Set paths: INPUT\_PATH (e.g., Bailey\_df\_V2.csv) and EXPORT\_DIR.
3. Run all cells top-to-bottom.
4. You’ll get one CSV at the end: **smart\_substitution\_working\_v4\_features\_neat.csv**.

### **4) What the pipeline does (high level)**

* **Basic cleaning & types:** numeric coercion for price/item\_size/price\_per\_unit; fix zeros/∞.
* **Brand extraction (generic):**
  + Learn product **headwords** per subcategory (from tags/names).
  + Take the **prefix before the first headword** as brand (1–3 tokens).
  + Small rules: detect “Coles …” as **store**; safe frequency fallback for brand\_raw.
  + Outputs: brand\_clean, **why** (brand\_confidence), and brand\_tier (store/branded/unbranded).
* **Size bands:** within **(subcategory × unit\_type)**
  + g/ml: tertiles ⇒ small/medium/large (fallback mixed if too few/low-variety).
  + each: mixed (pack parsing deferred to Sprint 2).
* **Tags:** ensured usable (list of strings) for later similarity.

### **5) Output schema (V4)**

product\_code, name, category, subcategory, unit\_type, price, item\_size, price\_per\_unit, tags, brand\_clean, brand\_confidence, brand\_tier, size\_band

We **drop** helper/duplicate columns (e.g., brand, brand\_raw, name\_norm, etc.) in the final CSV.

### **6) Light checks you’ll see**

* **Brand sanity:** % rows labelled unbranded (target much lower after fix).
* **Unit sanity:** boxplot of price\_per\_unit by g/ml/each.
* **Size coverage:** counts of size\_band by unit\_type.

### **7) Known limits (Sprint 1)**

* each items aren’t pack-parsed yet (default size = 1; size\_band = mixed).
* A small share of niche brands may remain unbranded.
* Outlier handling is minimal (kept simple for Sprint 1).

### **8) Hand-off to Sprint 2 (where to start)**

Use V4 to:

* Generate candidates within **(subcategory × unit\_type)**,
* Rank by **savings**, with **brand\_tier** and **size\_band** as guards,
* Optionally use **tags** for similarity.