# ELABORATION OF SOME DECISION MODELS FOR THE NUTRI-SCORE LABEL

Ali ABUSALEH
Rishika GUPTA

# Timeline

## 01

### Dataset

Extraction &
Preprocessing

## 02

### Additive Model

Study correlations &
determining Weights

## 03

### ELECTRE-Tri Model

- Pessimistic
- Optimistic

## 04

### Machine Learning Models

Results and confusion matrices

## 05

### Let's Compare!

Compare **our additive model with another group!**

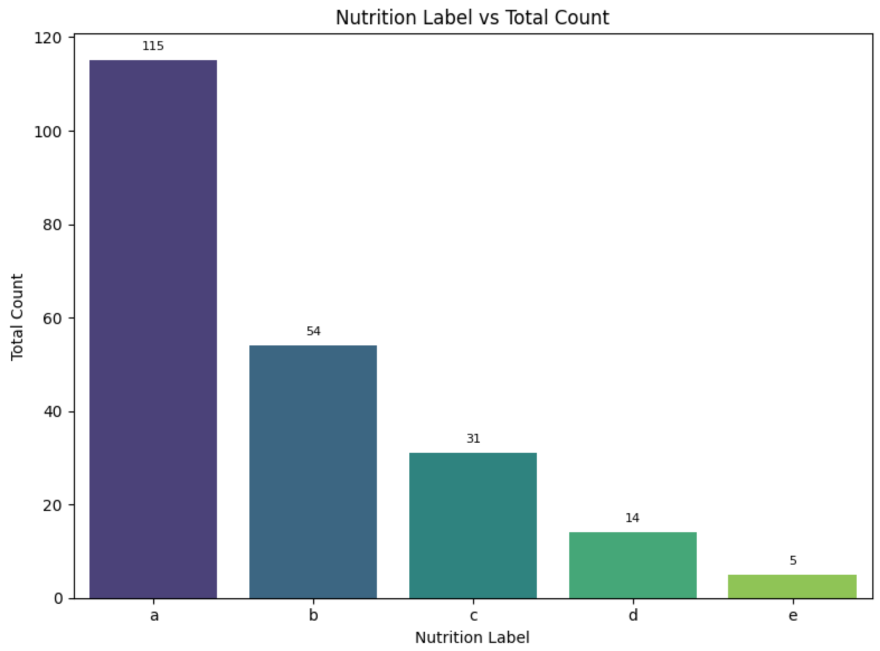
# Dataset

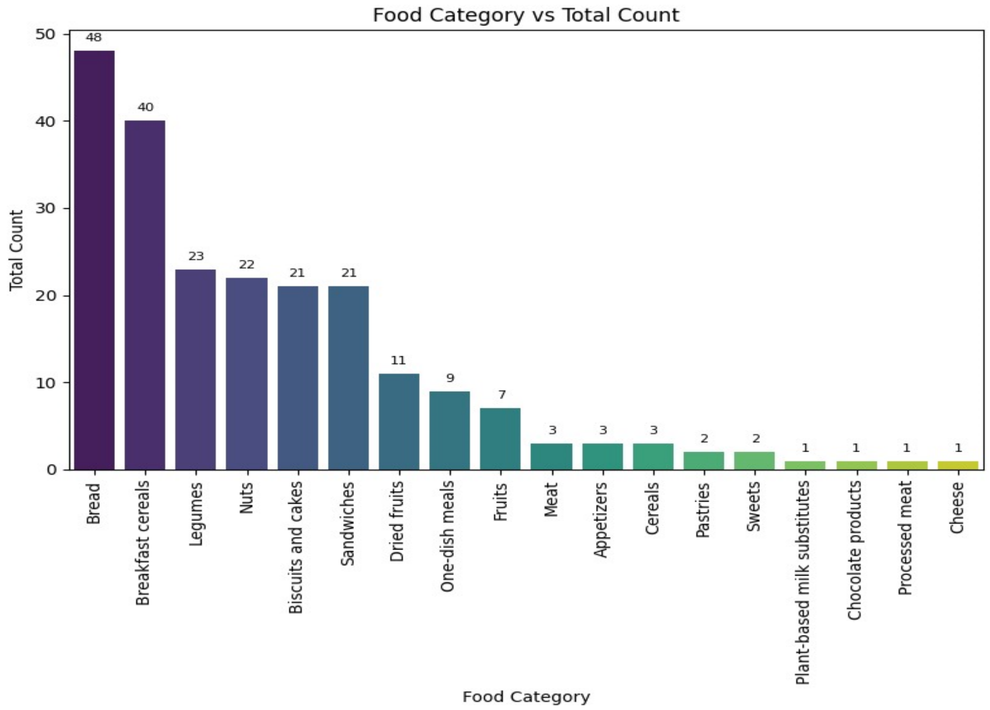
## Dataset Overview:

- **Data Source**: Open Food Facts API
- **Categories Extracted**: Biscuits, Breads, Nuts, Sandwiches, Snacks, Meat alternatives, Chocolate candies, Breakfast cereals, Fruits.
  - Excluded Beverages to have one nutriscore matrix.
  - Exclude products with **Negative points > 11 → Nutriscore equation**

## Data Preprocessing:

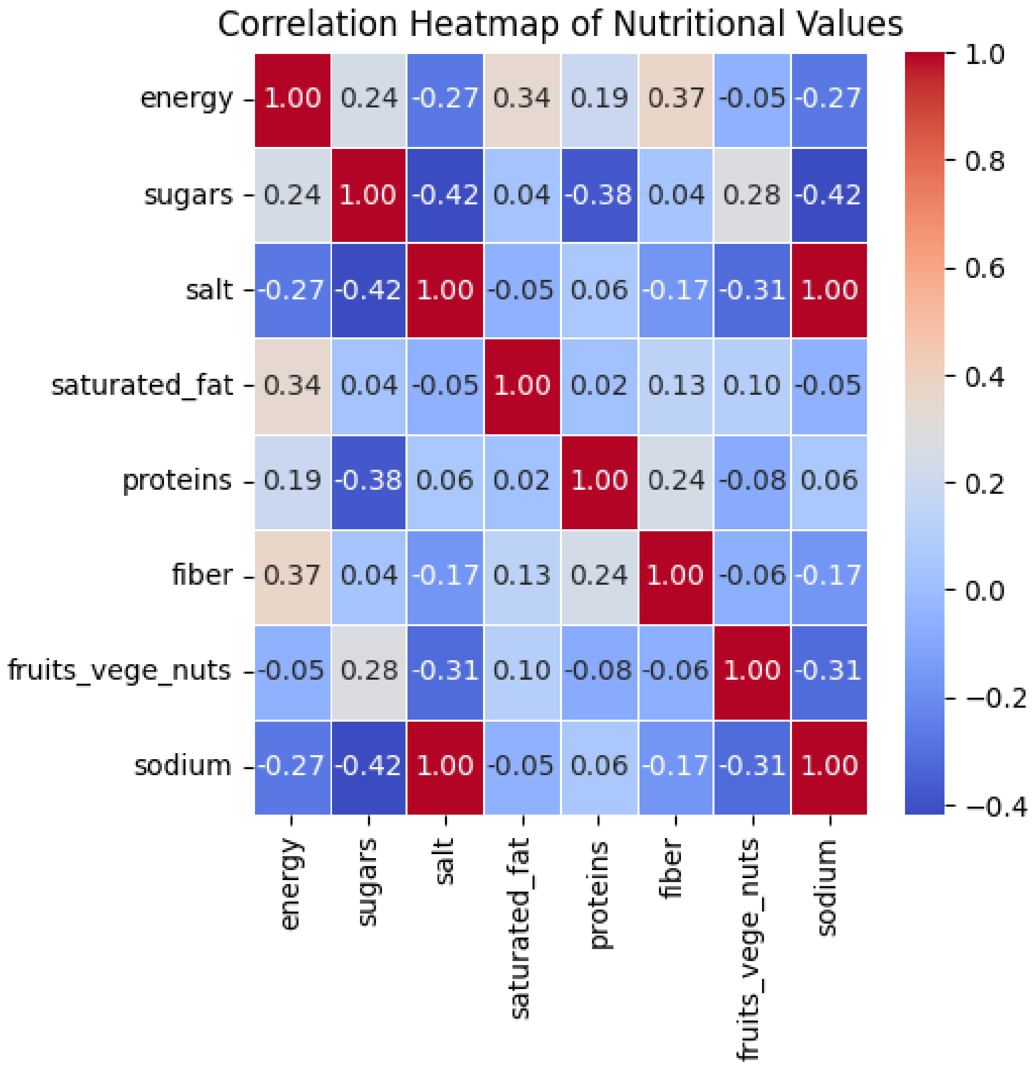
- Preprocessing Steps:
  - **Selected Key Columns**: _id, image_url, brands, pnns_groups_2, nutriments, nutriscore_data, nutrition_grade_fr
  - **Handled Missing Values**: Dropped rows with missing 'nutriments'
  - **JSON Data Handling**: Flattened 'nutriments' for analysis, validated 'nutriscore_data'
  - **Final Dataset**: Filtered relevant columns for analysis, created a final preprocessed dataset with 300 randomly sampled items.



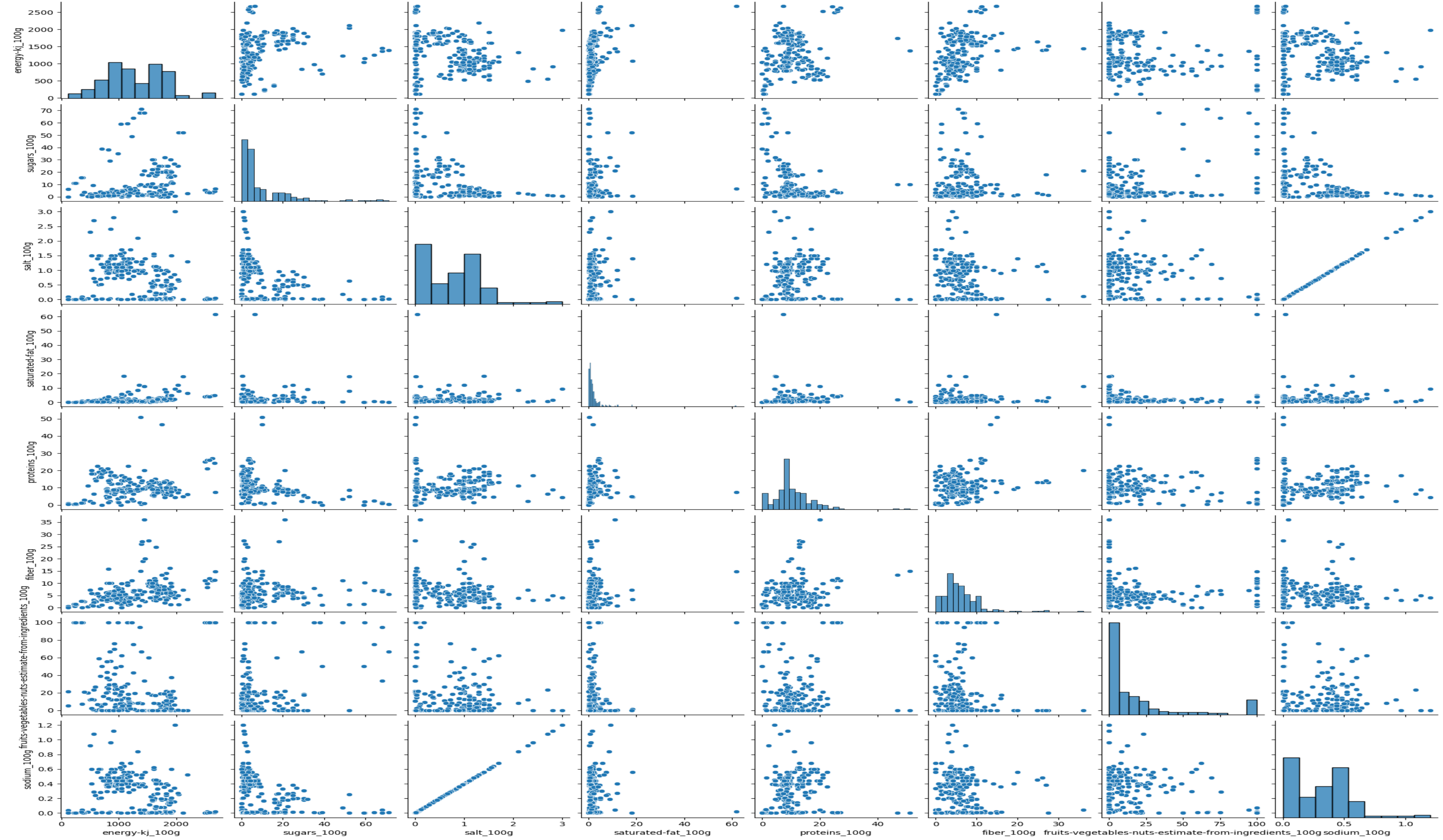



# Additive Model (1) - Study correlations -

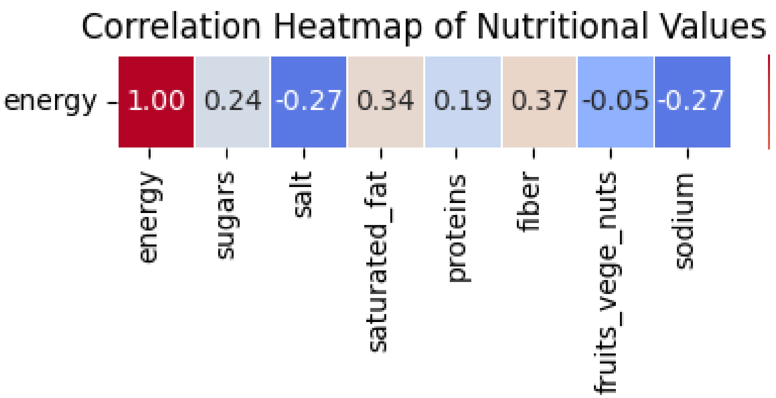
- In order to exclude the correlations in the criteria
  - Physical equation
    - $Energy = (9 \times fat) + {\color{red}(7 \times alcohol)} + (4 \times protein) + (4 \times Sugar) + {\color{red}(2.4 \times Organic\ acid)} + (2 \times Fibers)$
    - We are missing 2 items from the equation
      → **we can't exclude the energy totally.**
  - Values Correlations
    - Some observations:
      - Either salt or sodium can be omitted (as it is fully correlated (100%))
      - Energy is highly correlated with fat and fiber
      - Negative correlated with fruit !



# Study correlations



# Additive Model (2) - determining Weights -




- Model weights
  - $P(x) = 15.75 - \Big(\big(0.5 \times f(Energy)\big) + \big(0.1 \times f(Sugar)\big) + \big(0.4 \times f(Fat)\big) + \big(0.35 \times f(Salt)\big) + \big(0.2 \times f(Proteins)\big) + \big(0.3 \times f(Fiber)\big) - \big(0.05 \times f(Fruit)\big)\Big)$
  - Max Value of the points = 15.75.
  - Where $F(x) = Points\ for\ X\ criteria\ in\ the\ marginal\ utility\ function$
  - Assign Grade $G\big(P(x)\big)$ = Grade based on $P(X)$

$G(x) =$

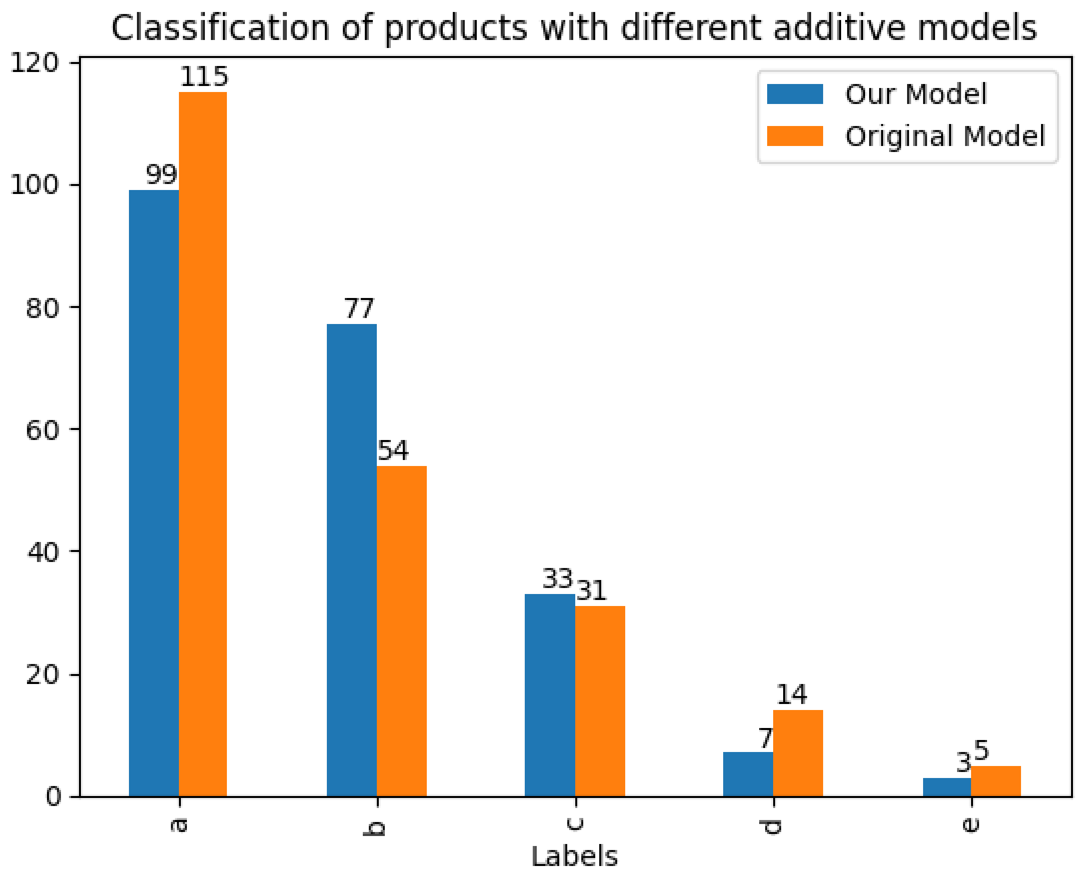
| |
|---|
| $F(X)$ <0 |
| $F(X)$ <=2 |
| $F(X)$ < 4.578 |
| $F(X)$ < 6.679 |
| $F(X)$ < 15.75 |

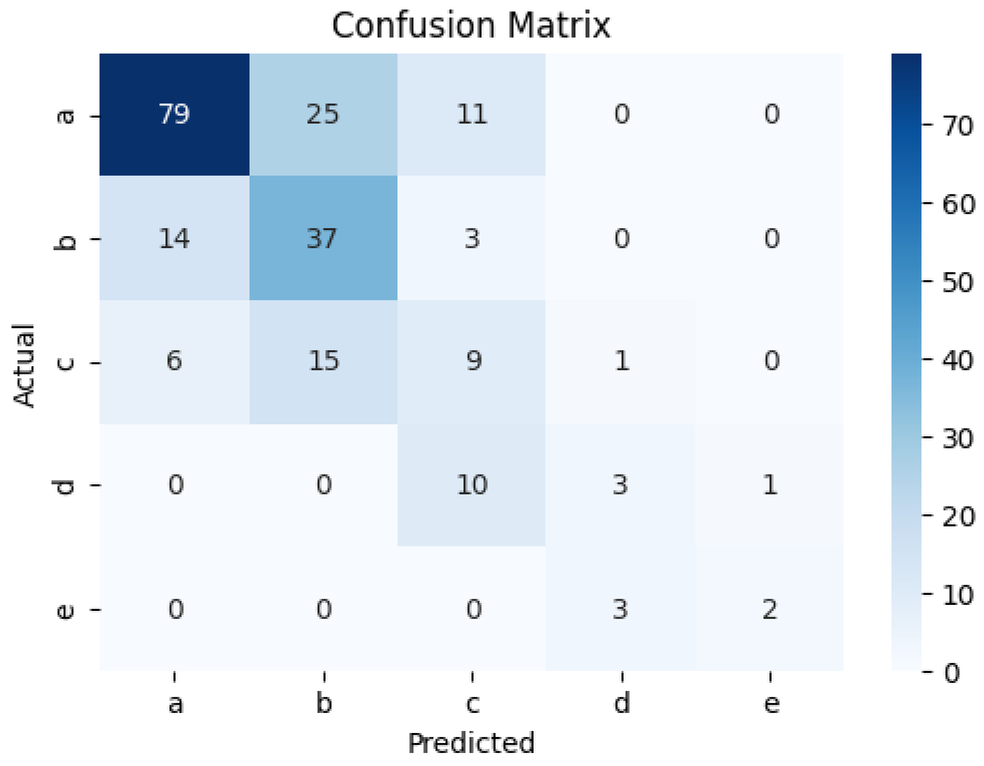

$F(x) =$

| Points | Fruits Légumes Fruits à coque (%) | Fibres (g/100g) NSP | ou AOAC | Protéines g/100g) |
|---|---|---|---|---|
| 0 | ≤40 | ≤0,7 | ≤0,9 | ≤1,6 |
| 1 | >40 | >0,7 | >0,9 | >1,6 |
| 2 | >60 | >1,4 | >1,9 | >3,2 |
| 3 | - | >2,1 | >2,8 | >4,8 |
| 4 | - | >2,8 | >3,7 | >6,4 |
| 5 | >80 | >3,5 | >4,7 | >8,0 |

| Points | Valeur énergétique (kJ/100g) | Acides gras saturés (g/100g) | Sucres (g/100g) | Sodium (mg/100g) |
|---|---|---|---|---|
| 0 | ≤335 | ≤1 | ≤4,5 | ≤90 |
| 1 | >335 | >1 | >4,5 | >90 |
| 2 | >670 | >2 | >9 | >180 |
| 3 | >1005 | >3 | >13,5 | >270 |
| 4 | >1340 | >4 | >18 | >360 |
| 5 | >1675 | >5 | >22,5 | >450 |
| 6 | >2010 | >6 | >27 | >540 |
| 7 | >2345 | >7 | >31 | >630 |
| 8 | >2680 | >8 | >36 | >720 |
| 9 | >3015 | >9 | >40 | >810 |
| 10 | >3350 | >10 | >45 | >900 |

# Additive Model (3) - Results-




- Our Model is more restrictive in the A class
- **But counts** isn't all we care about.

- We can see that in categories A and B our model predicted most of the data same as the original model.
- But what about the others? Let's see together

# Additive Model (3)
## Samples




| | Gerblé - Sugar Free Sesame Vanilla Cookie, 132g | Tartines de pain - blé complet - Bio - Pasquier - 240g | Amandes 100Cal la poignée - Carrefour - 200 g |
|---|---|---|---|
| Fat | Moderate | Moderate | **High** |
| Saturated fat | Moderate | Low | Moderate |
| Sugar | Low | Low | Low |
| Salt | Moderate | Moderate | Low |
| Energy | Moderate | Low | **High** |

Help!

Can you give estimated Grades A/B/C?

# Additive Model (3)

## Samples




| | Gerblé - Sugar Free Sesame Vanilla Cookie, 132g | Tartines de pain - blé complet - Bio - Pasquier - 240g | Amandes 100Cal la poignée - Carrefour - 200 g |
|---|---|---|---|
| Fat | Moderate | Moderate | **High** |
| Saturated fat | Moderate | Low | Moderate |
| Sugar | Low | Low | Low |
| Salt | Moderate | Moderate | Low |
| Energy | Moderate | Low | **High** |
| Original Model Grade | A | B | A |
| Our Model | B | A | C |

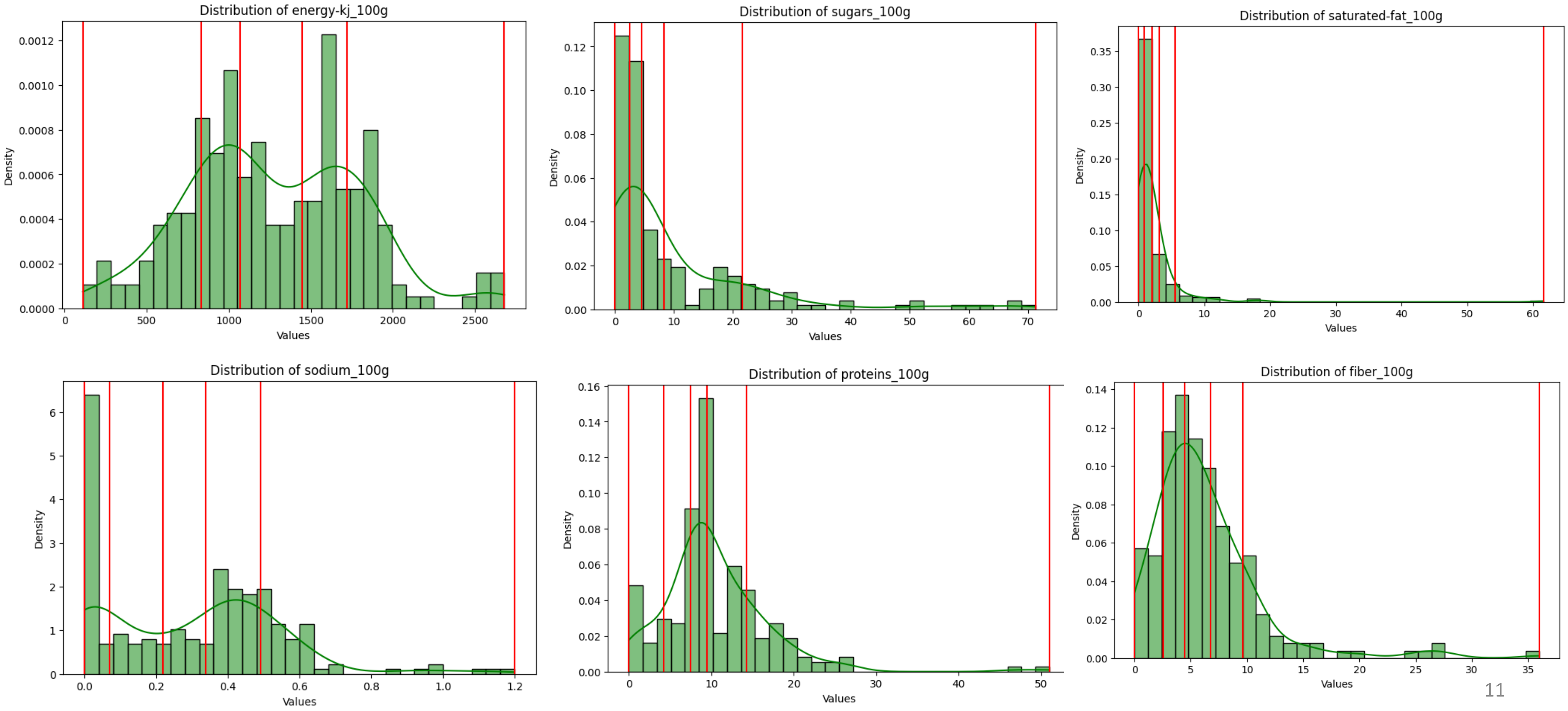
# ELECTRE-Tri Model

- Profiles built based on the percentiles (20% intervals) of the data.
- Some criteria are right-skewed (like saturated fat, and proteins).

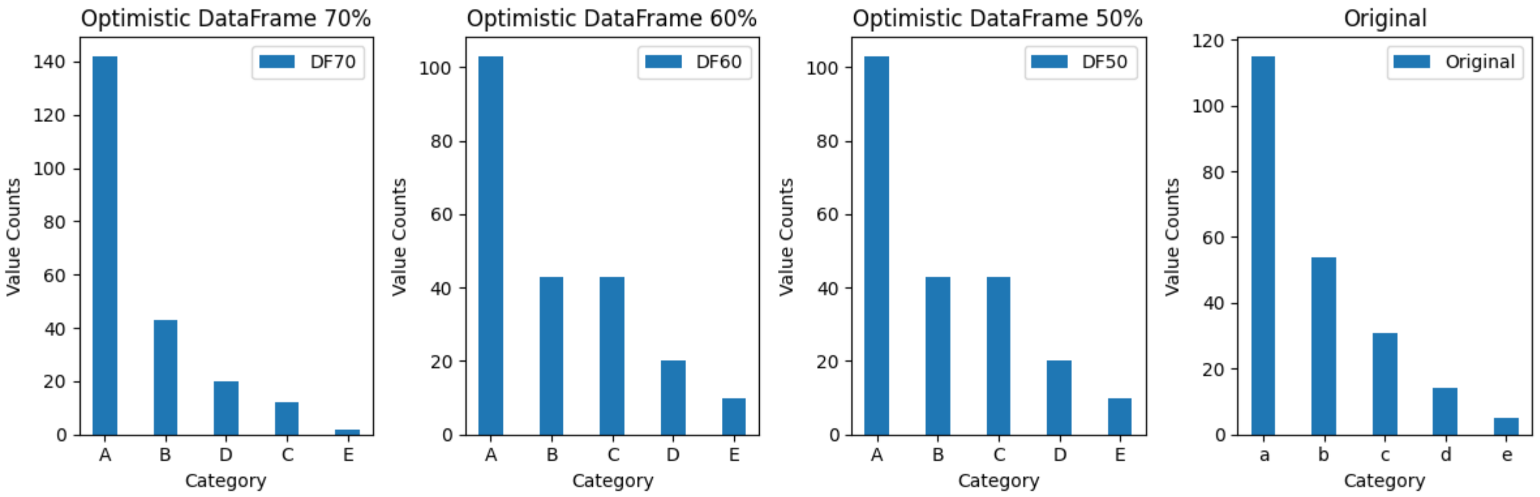
| | P-Limit #1 | P-Limit #2 | P-Limit #3 | P-Limit #4 | P-Limit #5 | P-Limit #6 |
|---|---|---|---|---|---|---|
| Energy | 111.0 | 832.2 | 1070.4 | 1446.4 | 1720.6 | 2679.0 |
| Sugar | 0.0 | 2.54 | 4.58 | 8.34 | 21.6 | 71.3 |
| Saturated Fat | 0.0 | 0.8 | 2.0 | 3.2 | 5.6 | 61.6 |
| Sodium | 0.0 | 0.0704 | 0.21760 | 0.3376 | 0.4904 | 1.2 |
| Proteins | 0.0 | 4.2 | 7.5 | 9.5 | 14.3 | 51.0 |
| Fiber | 0.0 | 2.546 | 4.48 | 6.7399 | 9.620 | 36.0 |
| Fruits – Nuts | 0.0 | 3.90 | 8.455 | 18.211 | 28.286 | 100 |

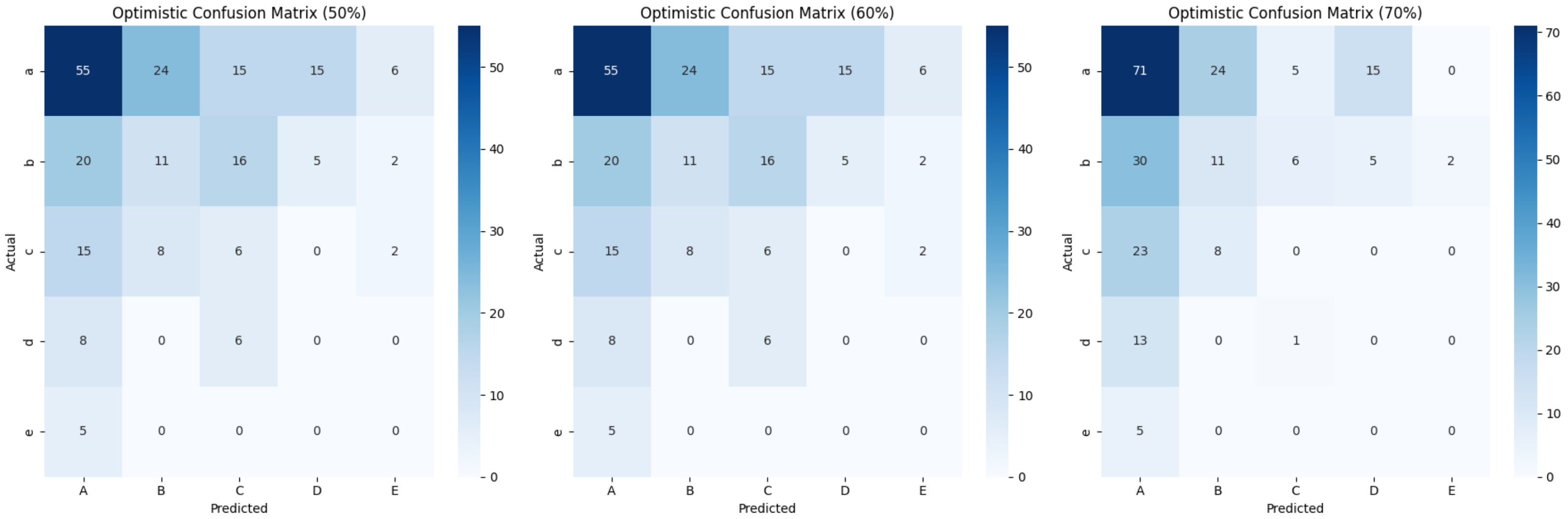
- 5 Profiles in our Model (A, B, C, D, E)

# ELECTRE-Tri Model Profiles



# Optimistic ELECTRE-Tri Model Results (provided weights)

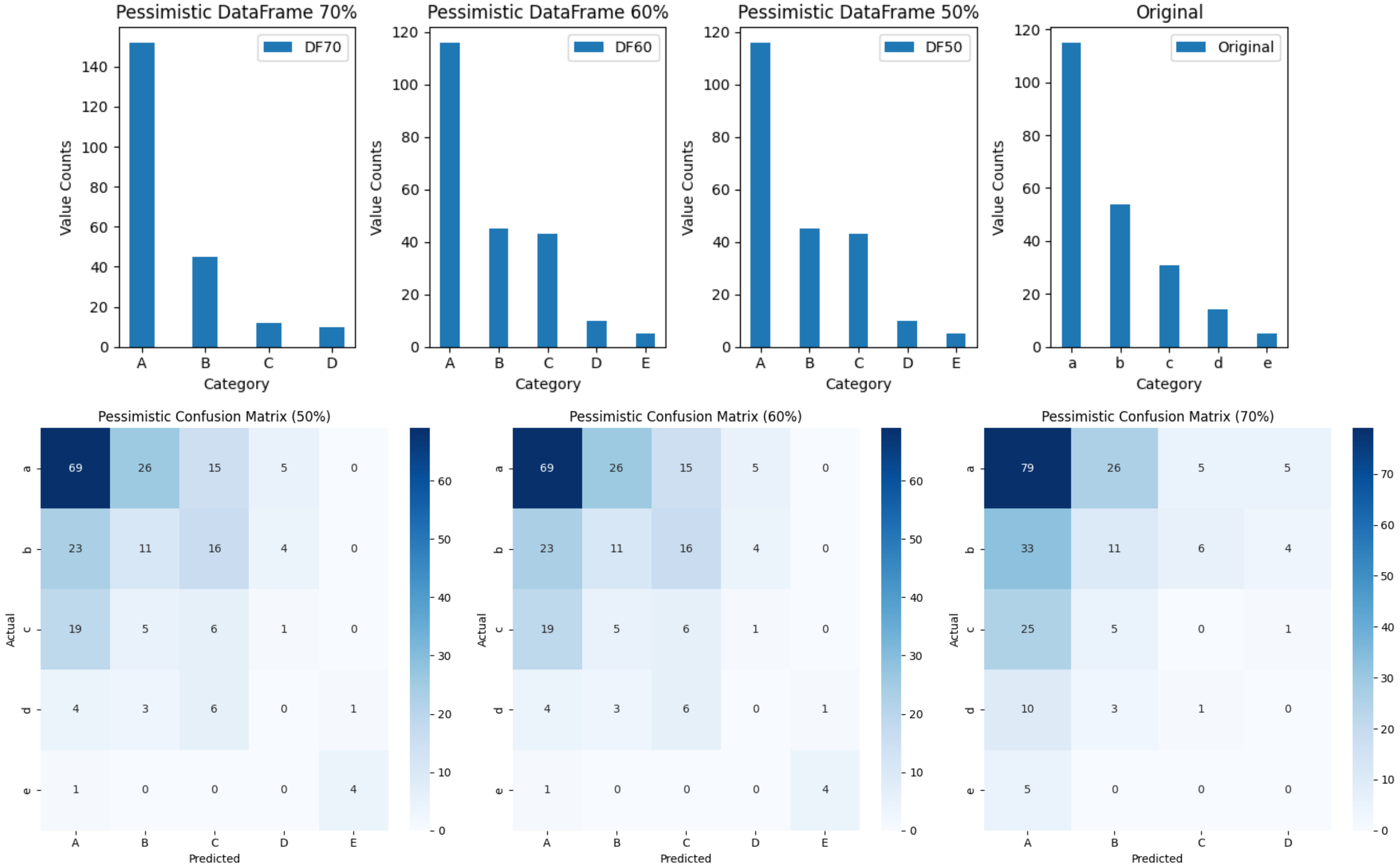



Insights:

- Threshold 50% and 60% has the same results.
- The skewed data affected the system to be very optimistic in assigning the grades toward higher class (A,B)

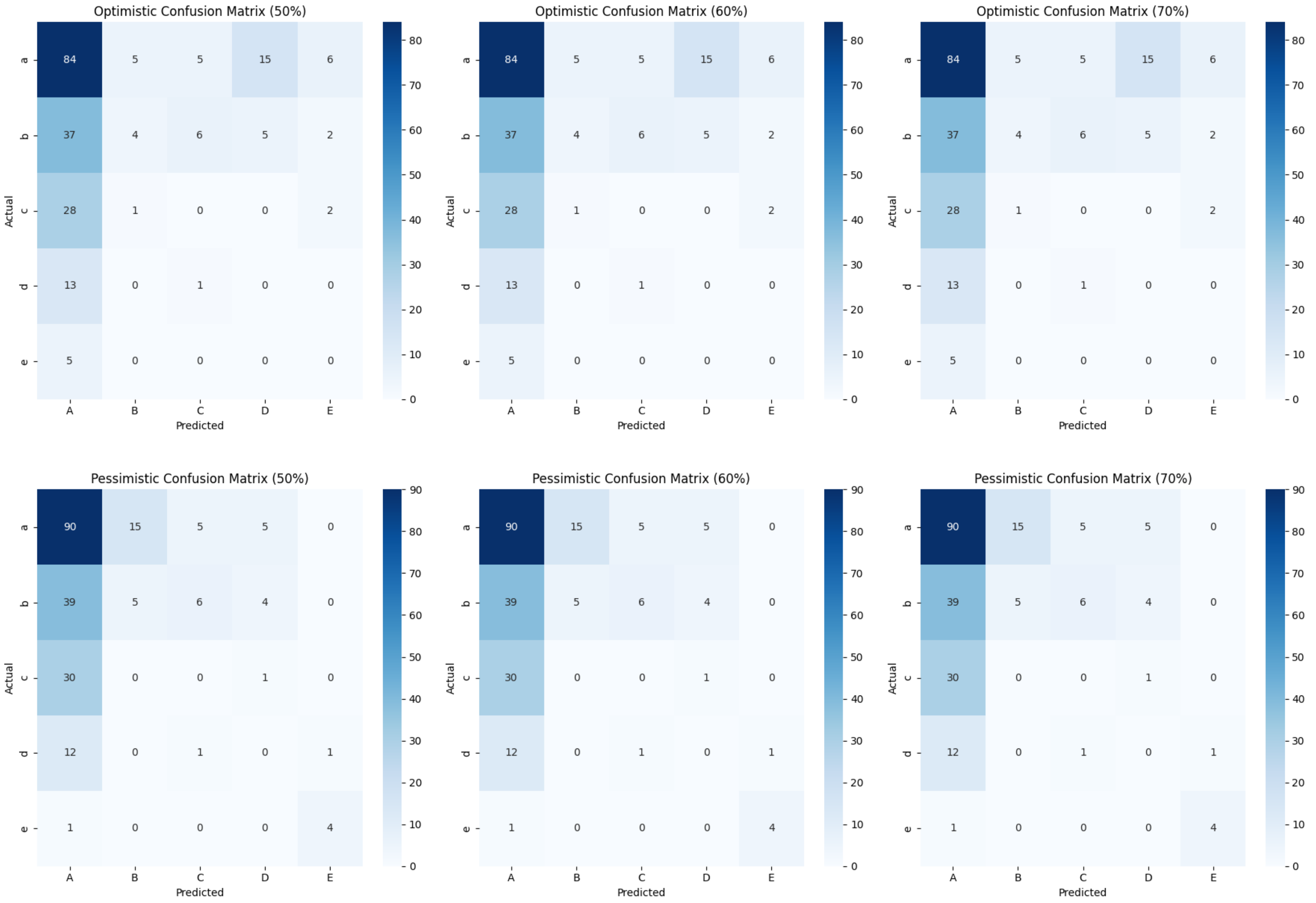
# Pessimistic ELECTRE-Tri Model Results (provided weights)




Insights:

- Threshold 50% and 60% has the same results.
- The Pessimistic with 70% threshold didn't assign any item to grade E!
- The Pessimistic with higher threshold skewed the grades to class A.

# ELECTRE-Tri Model Results (Our weights)




- **Weights are chosen based on our additive model + exclude energy.**

as follow:

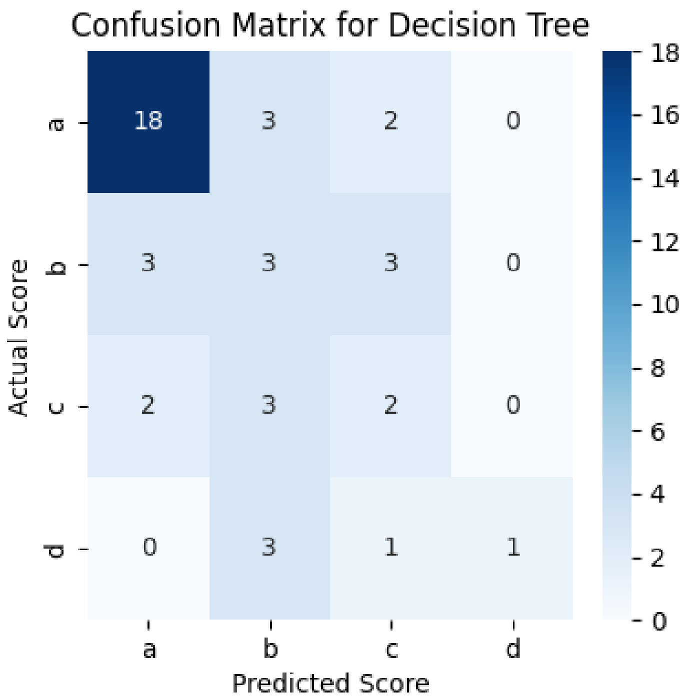
{en: 0, su: 2, fa: 4, sa: 0.8, pr: 2, fi: 1, fr: 0.5}

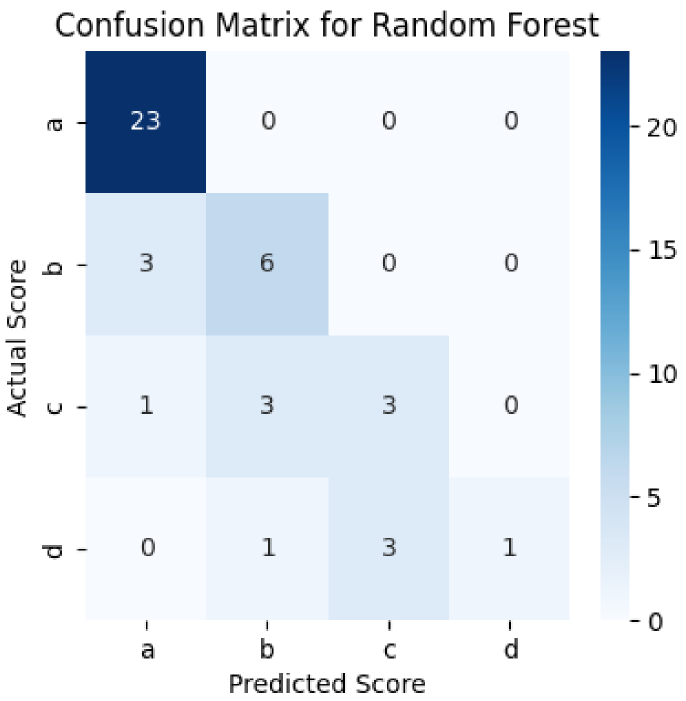
Insights:

- Regardless of the threshold the class assigned to the items are the same!
- the Difference between the optimistic and pessimistic are very small.
- Both models are providing similar results with 95% of the product.

# Machine Learning Models

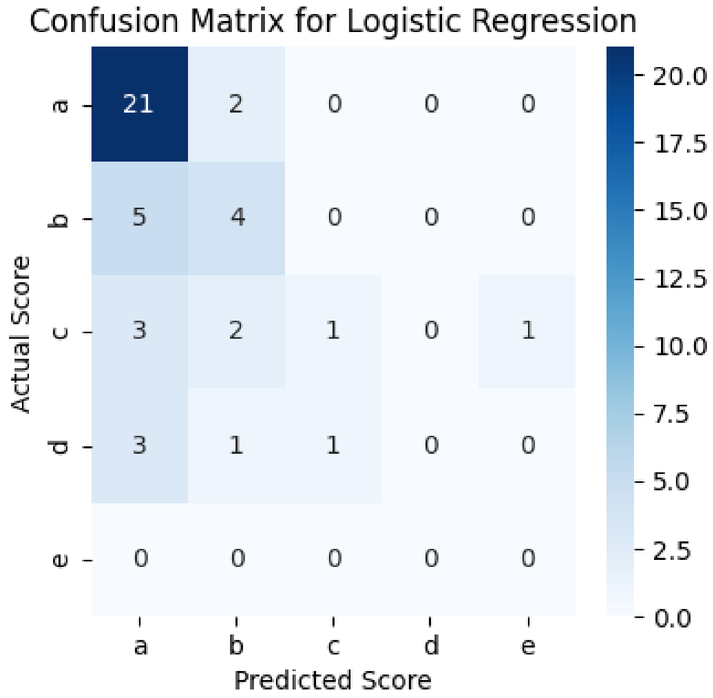
- **Decision Tree**
  - Setup:
    - Criterion ➔ Entropy.
    - Max Depth ➔ 7.
  - Insights:
    - Results biased toward class occurrences → since our data has a lot of A > B > C  > D class products.
    - **Unable to predict class E.**
    - Accuracy of the Model is 54.5%.
- **Random Forest**
  - Setup:
    - Max depth ➔ 5.
  - Insights:
    - Less biased against the data compared with the decision tree.
    - Product's classes **is either same class or lower**, but not higher class.
    - **Unable to predict class E.**
    - Accuracy of the model is 75%.

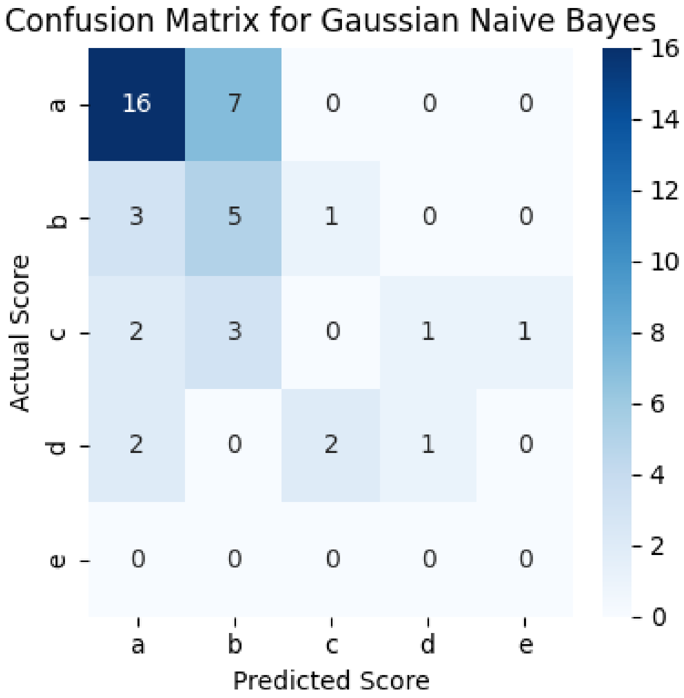


# Machine Learning Models

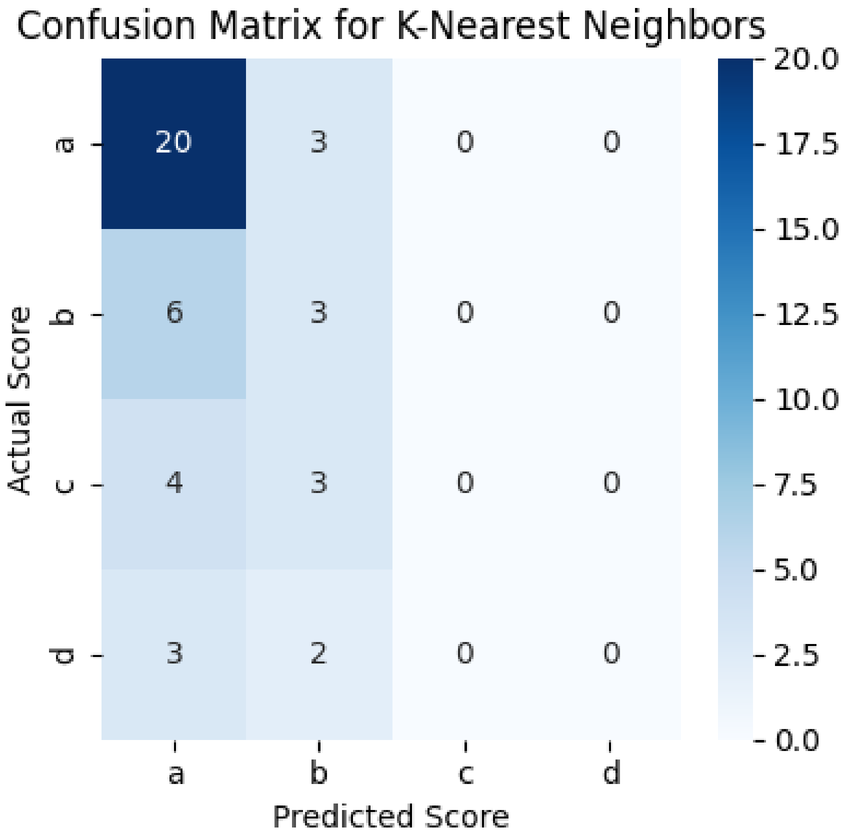
- **Logistic Regression**
  - Setup:
    - default.
  - Insights:
    - **Very similar to decision tree**.
    - Results biased toward A class → since our data has a lot of A class products.
    - Predicted Class E but wrong!
    - Accuracy of the Model is 59.1%.
- **Gaussian Naïve Bayes.**
  - Setup:
    - default.
  - Insights:
    - Less biased against the data compared with the decision tree.
    - Product's classes **is either same class or lower**, but not higher class.
    - Predicted Class E but wrong!
    - Accuracy of the model is 75%.



# Machine Learning Models

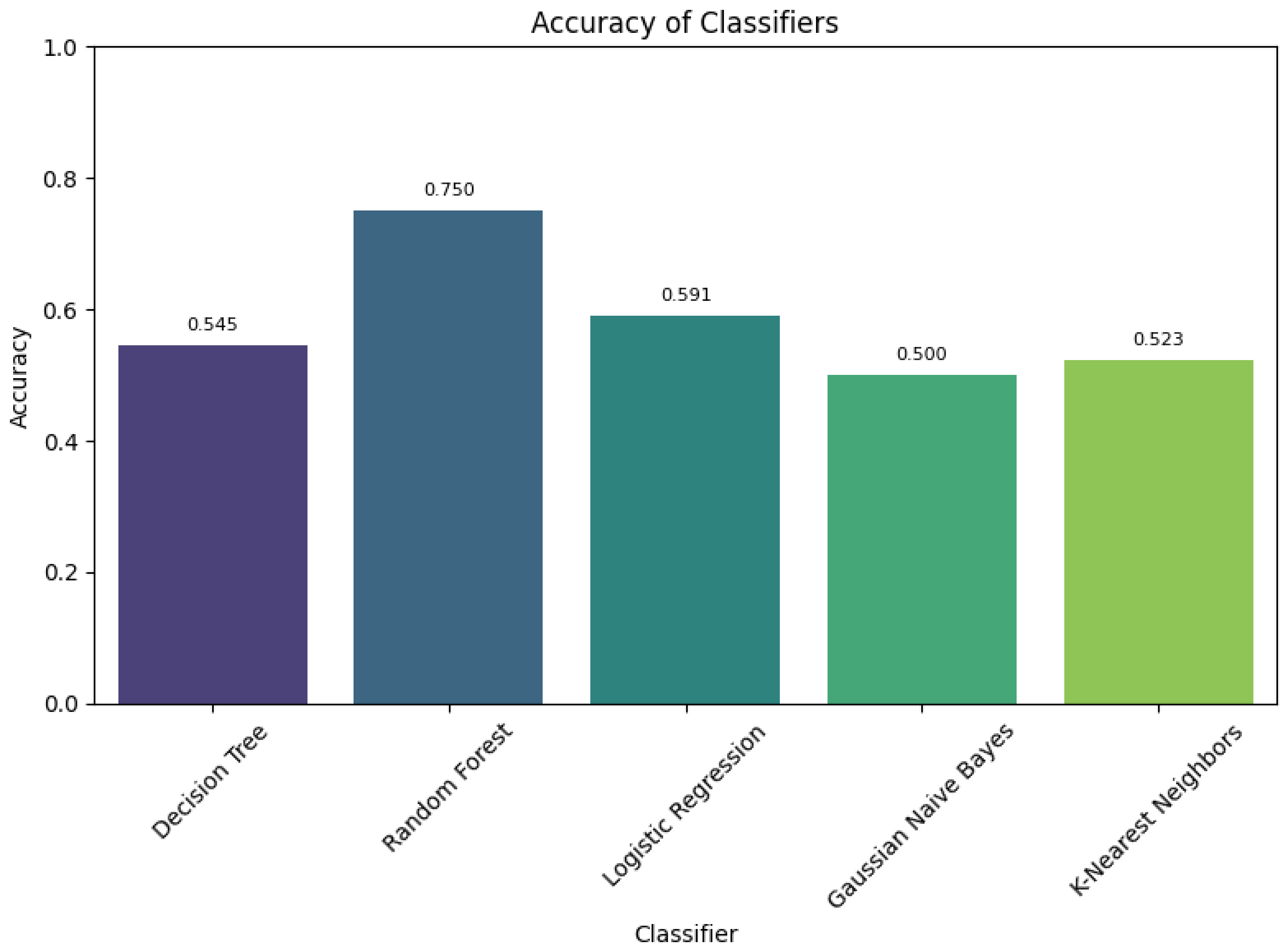
- **K-Nearest Neighbors**
  - Setup:
    - Number of neighbors ➔ 15.
  - Insights:
    - **Very similar to decision tree**.
    - Results biased toward A class → since our data has a lot of A class products.
    - Since the # neighbors > E present in the data base, E is treated as outlier.
    - Accuracy of the Model is 52.3%.



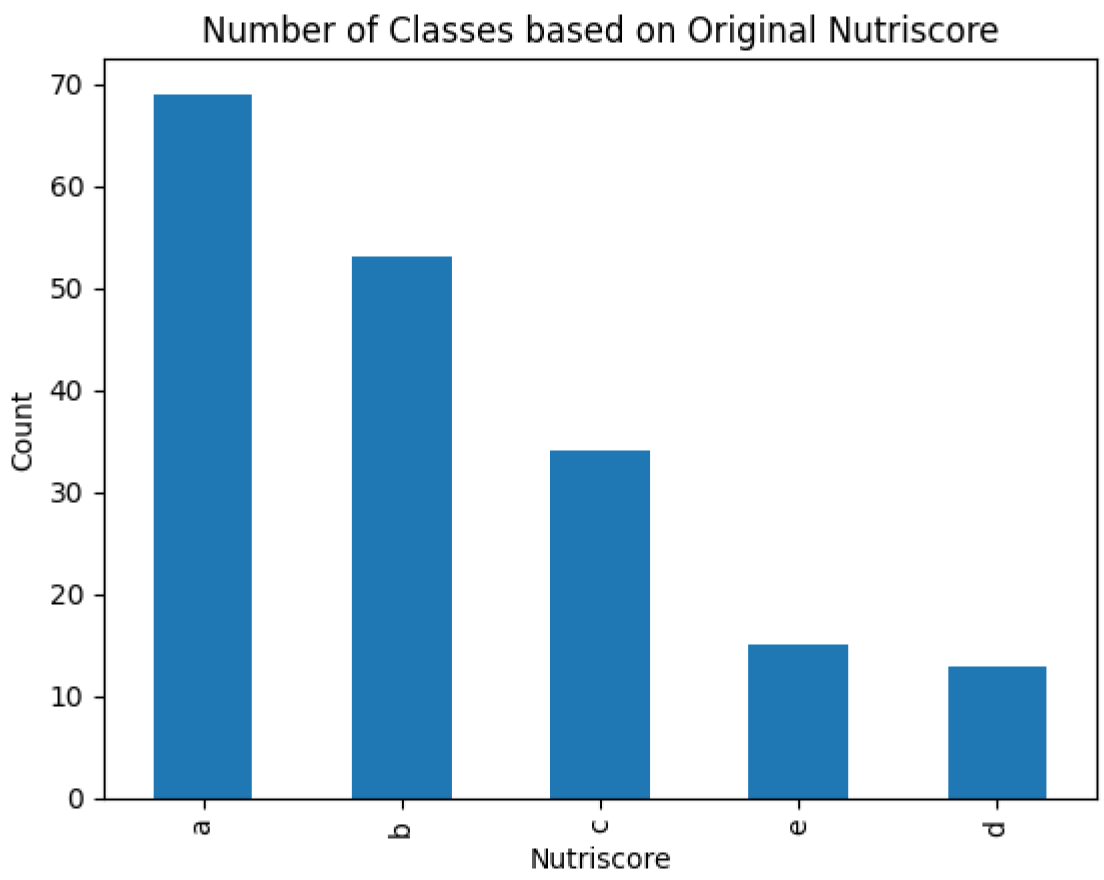

Final insights and results…….(next) ➔➔➔

# Machine Learning Models Final results

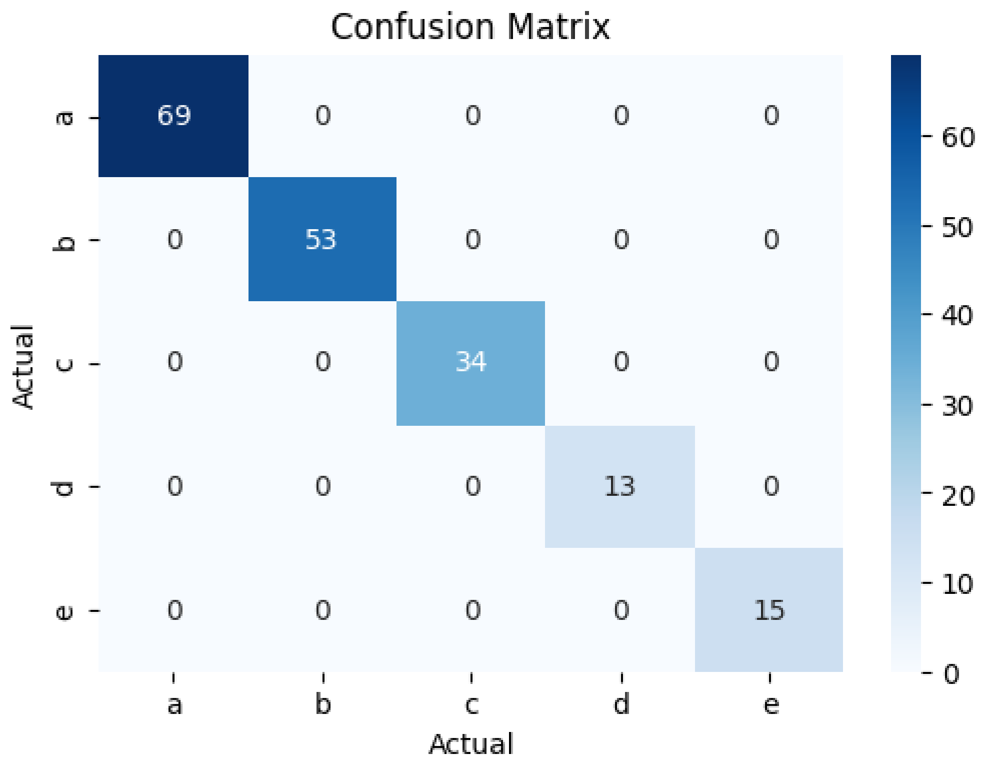
- Machine learning is **data hungry**, and the data **needs** to be **unbiased** in the number of each class.
- Train-test-split: 80:20 for all.
- These models learn from the already assigned data ➔ **the correlations** in the “true” data calculation **persists**.
- **Random Forest** has the highest accuracy.
- Product’s classes **is either same class or lower**, but not higher class.
- More data will result in better true positive and less false negative.



# Challenge Time! Let's compare



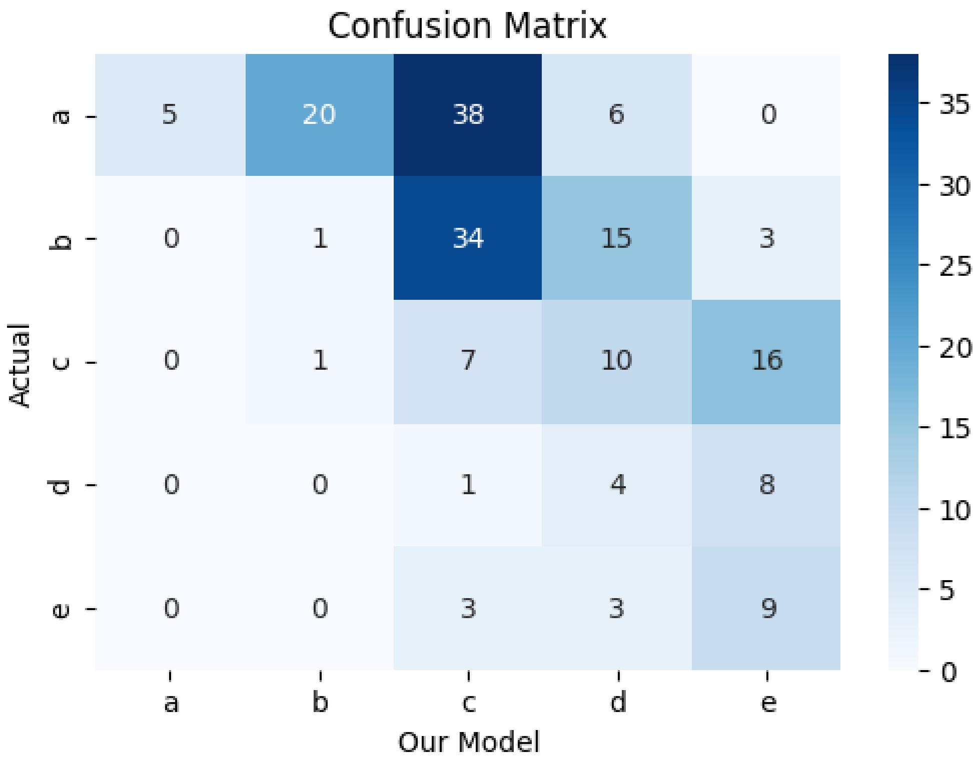

**Original Data**



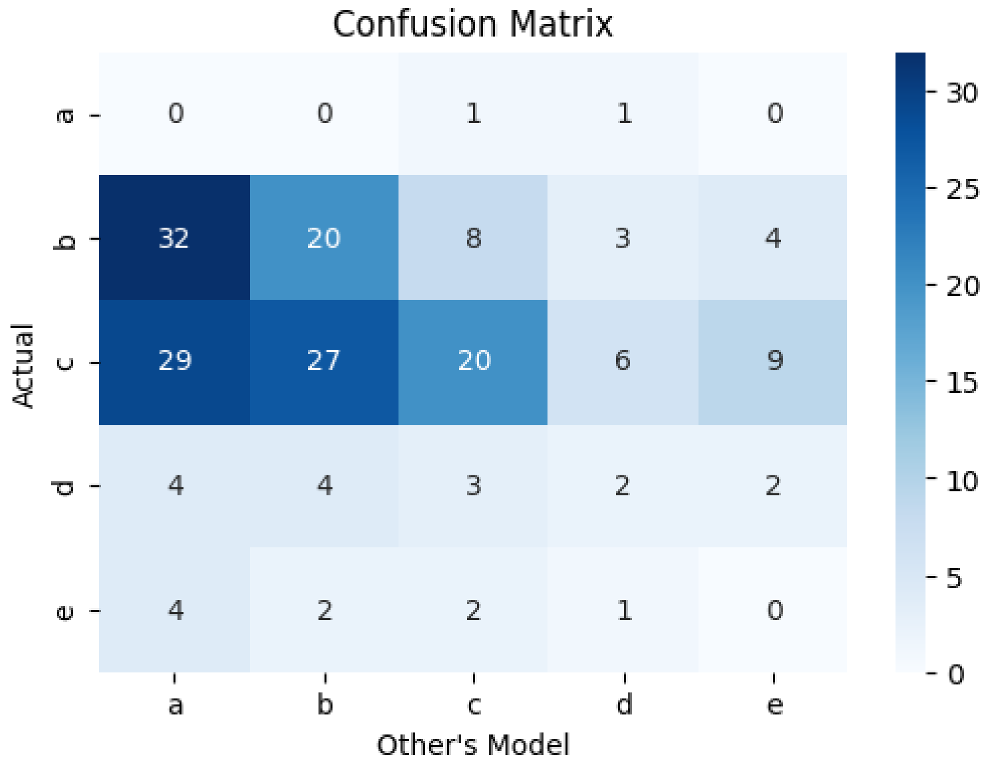

Note: Their dataset includes Beverages, it was hard to generalize to our model's score

# Challenge Time! Let's compare

**Our additive Model compared to original data**




**Other's Model compared to original data**

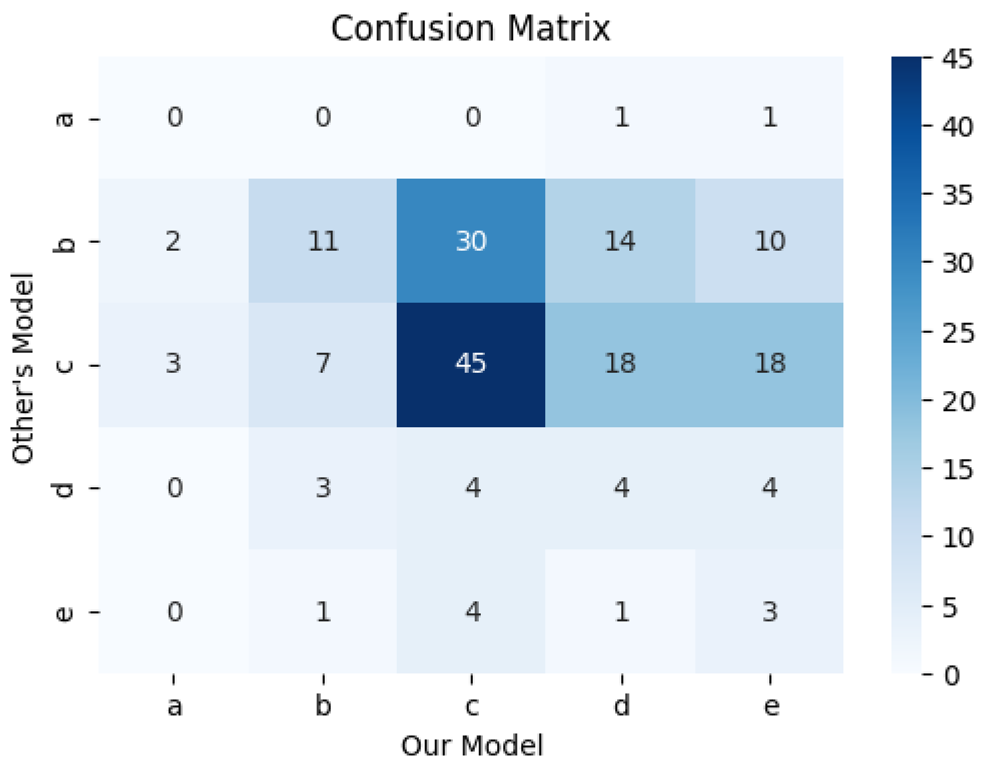



- It got accuracy of TP as follows:
  - Class A: 7.2% === Class B: 1.8%
  - Class C: 20.5% === Class D: 30.8%
  - Class E: 60%
- Overall : 24.06%

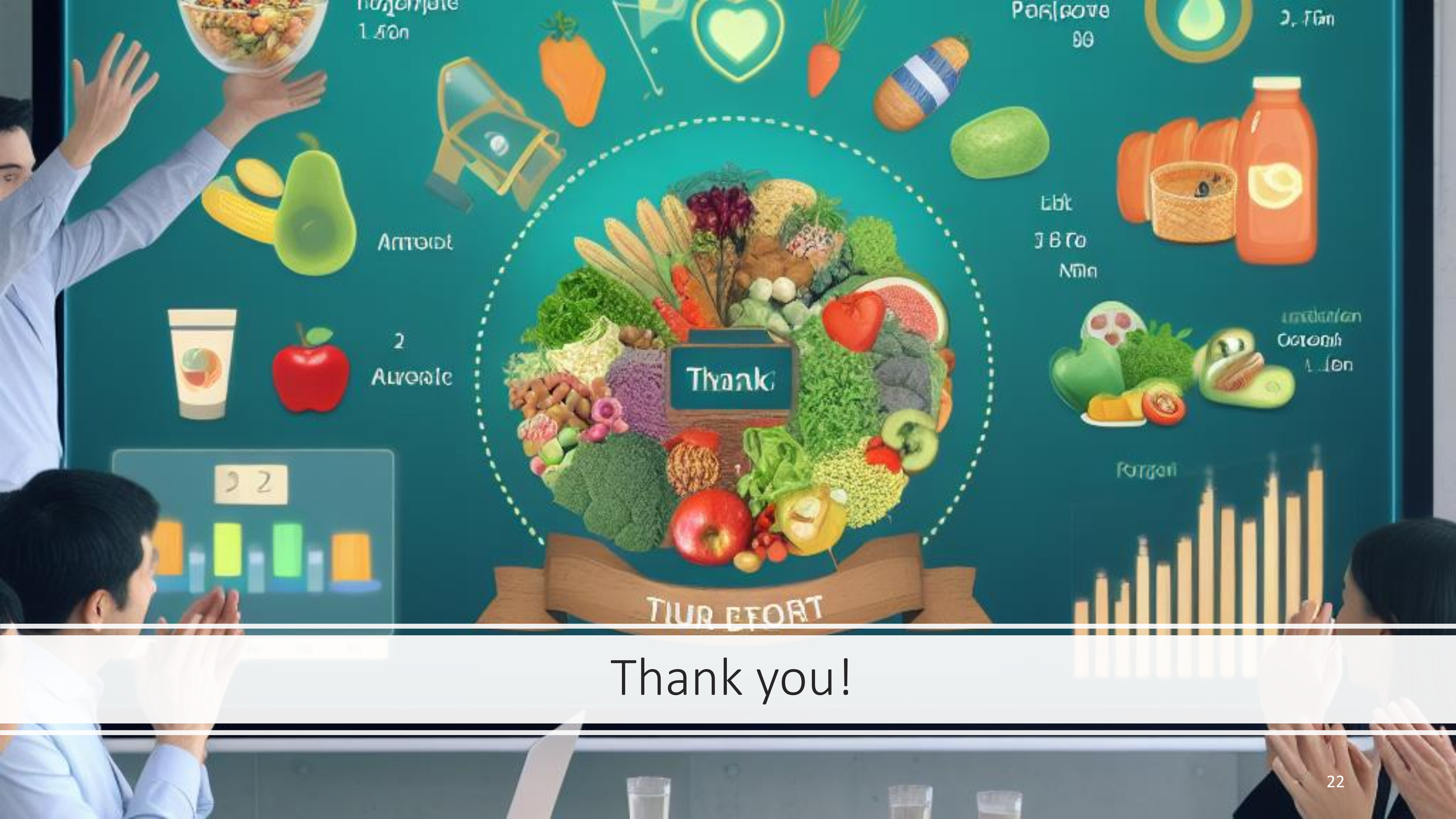
- It got accuracy of TP as follows:
  - Class A: 0.0% === Class B: 37.74%
  - Class C: 58.9% === Class D: 15.4%
  - Class E: 0.0%
- Overall: 22.408%

# Challenge Time! Let's compare

**Our additive Model compared to Other's results**




- Other's model works better in the middle classes (B&C).
  - **GOOD JOB GUYS!**
- Our Model has better coverage over all classes and doesn't neglect any class.
- We need to consider removing the Beverages to better evaluate against our model.




# Thank you!