

POWER BI ALL MEASURES

BASIC KPI MEASURES

// Total Products

Total Products = COUNTROWS('amazon_cleaned')

// Total Profit (in Lakhs)

Total Profit Lakhs = SUM('amazon_cleaned'[estimated_profit]) / 100000

// Total Profit (in Crores)

Total Profit Cr = SUM('amazon_cleaned'[estimated_profit]) / 10000000

// Average Margin %

Avg Margin % = AVERAGE('amazon_cleaned'[margin_percent])

// Average Discount %

Avg Discount % = AVERAGE('amazon_cleaned'[discount_percentage])

// Average Rating

Avg Rating = AVERAGE('amazon_cleaned'[rating])

// Total Revenue (in Lakhs)

Total Revenue Lakhs = SUM('amazon_cleaned'[estimated_revenue]) / 100000

✅ CALCULATED COLUMNS

// Price Segment Column

Price Segment =

```
SWITCH(  
    TRUE(),  
    'amazon_cleaned'[actual_price] < 500, "Budget (<₹500)",  
    'amazon_cleaned'[actual_price] < 2000, "Mid-Range (₹500-2000)",  
    'amazon_cleaned'[actual_price] < 5000, "Premium (₹2000-5000)",  
    "Luxury (>₹5000)"  
)
```

// Rating Category Column

Rating Category =

```
SWITCH(  
    TRUE(),  
    'amazon_cleaned'[rating] < 3, "Poor (1-3)",  
    'amazon_cleaned'[rating] < 4, "Average (3-4)",  
    "Good (4-5)"  
)
```

// Margin Tier Column

Margin Tier =

```
SWITCH(  
    TRUE(),  
    'amazon_cleaned'[margin_percent] < 0, "Loss Making",  
    'amazon_cleaned'[margin_percent] < 10, "Low Margin (0-10%)",  
    'amazon_cleaned'[margin_percent] < 20, "Medium Margin (10-20%)",  
    'amazon_cleaned'[margin_percent] < 30, "High Margin (20-30%)",  
    "Super High Margin (>30%)"  
)
```

// Profit Status Column

Profit Status =

IF('amazon_cleaned'[estimated_profit] > 0, "Profitable", "Loss Making")

COUNT MEASURES

// Loss Making Products Count

Loss Making Count =

**COUNTROWS(
 FILTER(
 'amazon_cleaned',
 'amazon_cleaned'[estimated_profit] < 0
)
)**

// Loss Making Percentage

Loss Making % =

**DIVIDE(
 [Loss Making Count],
 [Total Products],
 0
) * 100**

// Profitable Products Count

Profitable Products Count =

**COUNTROWS(
 FILTER(
 'amazon_cleaned',
 'amazon_cleaned'[estimated_profit] > 0
)
)**

)

// Profitable Products %

Profitable % =

DIVIDE(

[Profitable Products Count],

[Total Products],

0

) * 100

// High Margin Count (>20%)

High Margin Count =

COUNTROWS(

FILTER(

'amazon_cleaned',

'amazon_cleaned'[margin_percent] > 20

)

)

// Low Margin Count (0-10%)

Low Margin Count =

COUNTROWS(

FILTER(

'amazon_cleaned',

'amazon_cleaned'[margin_percent] > 0 &&

'amazon_cleaned'[margin_percent] < 10

)

)

// Critical Margin Count (<0%)

Critical Margin Count =

```
COUNTROWS(  
  FILTER(  
    'amazon_cleaned',  
    'amazon_cleaned'[margin_percent] < 0  
  )  
)
```

✅ LOSS PRODUCTS ANALYSIS MEASURES

// Total Loss Value (in Lakhs)

Total Loss Lakhs =

```
CALCULATE(  
  SUM('amazon_cleaned'[estimated_profit]),  
  'amazon_cleaned'[estimated_profit] < 0  
) / -100000
```

// Loss Products - Avg Discount

Loss Products Avg Discount =

```
CALCULATE(  
  AVERAGE('amazon_cleaned'[discount_percentage]),  
  'amazon_cleaned'[estimated_profit] < 0  
)
```

// Loss Products - Avg Price

Loss Products Avg Price =

```
CALCULATE(  
  AVERAGE('amazon_cleaned'[actual_price]),  
  'amazon_cleaned'[estimated_profit] < 0  
)
```

// Loss Products - Avg Rating

Loss Products Avg Rating =

```
CALCULATE(  
    AVERAGE('amazon_cleaned'[rating]),  
    'amazon_cleaned'[estimated_profit] < 0  
)
```

// Loss Products - Avg Cost %

Loss Products Cost % =

```
AVERAGEX(  
    FILTER('amazon_cleaned', 'amazon_cleaned'[estimated_profit] < 0),  
    DIVIDE('amazon_cleaned'[estimated_cost], 'amazon_cleaned'[actual_price], 0) * 100  
)
```

PROFITABLE PRODUCTS ANALYSIS MEASURES

// Profitable Products - Avg Discount

Profitable Products Avg Discount =

```
CALCULATE(  
    AVERAGE('amazon_cleaned'[discount_percentage]),  
    'amazon_cleaned'[estimated_profit] > 0  
)
```

// Profitable Products - Avg Price

Profitable Products Avg Price =

```
CALCULATE(  
    AVERAGE('amazon_cleaned'[actual_price]),  
    'amazon_cleaned'[estimated_profit] > 0  
)
```

// Profitable Products - Avg Rating

Profitable Products Avg Rating =

```
CALCULATE(  
    AVERAGE('amazon_cleaned'[rating]),
```

```
'amazon_cleaned'[estimated_profit] > 0
)
```

```
// Profitable Products - Avg Cost %
```

```
Profitable Products Cost % =
```

```
AVERAGEX(
    FILTER('amazon_cleaned', 'amazon_cleaned'[estimated_profit] > 0),
    DIVIDE('amazon_cleaned'[estimated_cost], 'amazon_cleaned'[actual_price], 0) * 100
)
```

```
// Profitable Products Value
```

```
Profitable Products Value =
```

```
CALCULATE(
    SUM('amazon_cleaned'[estimated_profit]),
    'amazon_cleaned'[estimated_profit] > 0
) / 100000
```

SOLUTION MEASURES (4 RECOMMENDATIONS)

```
// SOLUTION 1: PRICE INCREASE
```

```
// Products for price increase (margin > -10%)
```

```
Price Increase Products Count =
```

```
COUNTROWS(
    FILTER(
        'amazon_cleaned',
        'amazon_cleaned'[estimated_profit] < 0 &&
        'amazon_cleaned'[margin_percent] > -10
    )
)
```

```
Price Increase Current Price =
```

```
CALCULATE(
```

```

    AVERAGE('amazon_cleaned'[actual_price]),
    'amazon_cleaned'[estimated_profit] < 0 &&
    'amazon_cleaned'[margin_percent] > -10
)

```

Price Increase New Price = [Price Increase Current Price] * 1.2

Price Increase Expected Recovery =

[Price Increase Products Count] * [Price Increase Current Price] * 0.2 * 0.5 / 100000

// SOLUTION 2: DISCOUNT REDUCTION

// Products for discount reduction (discount > 25%)

Discount Reduce Products Count =

```

COUNTROWS(
    FILTER(
        'amazon_cleaned',
        'amazon_cleaned'[estimated_profit] < 0 &&
        'amazon_cleaned'[discount_percentage] > 25
    )
)

```

Discount Reduce Current Discount =

```

CALCULATE(
    AVERAGE('amazon_cleaned'[discount_percentage]),
    'amazon_cleaned'[estimated_profit] < 0 &&
    'amazon_cleaned'[discount_percentage] > 25
)

```

Discount Reduce New Discount = 15

Discount Reduce Expected Recovery = [Discount Reduce Products Count] * 5000 / 100000

// SOLUTION 3: DISCONTINUE

// Products to discontinue (margin < -10% AND rating < 2.5)

Discontinue Products Count =

```
COUNTROWS(  
  FILTER(  
    'amazon_cleaned',  
    'amazon_cleaned'[estimated_profit] < 0 &&  
    'amazon_cleaned'[margin_percent] < -10 &&  
    'amazon_cleaned'[rating] < 2.5  
  )  
)
```

Discontinue Loss Saved =

```
CALCULATE(  
  SUM('amazon_cleaned'[estimated_profit]),  
  'amazon_cleaned'[estimated_profit] < 0 &&  
  'amazon_cleaned'[margin_percent] < -10 &&  
  'amazon_cleaned'[rating] < 2.5  
) / -100000
```

// SOLUTION 4: BUNDLE

// Products for bundle (margin between -10% and 0%)

Bundle Products Count =

```
COUNTROWS(  
  FILTER(  
    'amazon_cleaned',  
    'amazon_cleaned'[estimated_profit] < 0 &&
```

```

        'amazon_cleaned'[margin_percent] >= -10 &&
        'amazon_cleaned'[margin_percent] <= 0
    )
)

```

Bundle Products Loss =

```

CALCULATE(
    SUM('amazon_cleaned'[estimated_profit]),
    'amazon_cleaned'[estimated_profit] < 0 &&
    'amazon_cleaned'[margin_percent] >= -10 &&
    'amazon_cleaned'[margin_percent] <= 0
) / -100000

```

Bundle Expected Recovery = [Bundle Products Loss] * 0.4

Bundle Avg Discount =

```

CALCULATE(
    AVERAGE('amazon_cleaned'[discount_percentage]),
    'amazon_cleaned'[estimated_profit] < 0 &&
    'amazon_cleaned'[margin_percent] >= -10 &&
    'amazon_cleaned'[margin_percent] <= 0
)

```

Bundle Avg Rating =

```

CALCULATE(
    AVERAGE('amazon_cleaned'[rating]),
    'amazon_cleaned'[estimated_profit] < 0 &&
    'amazon_cleaned'[margin_percent] >= -10 &&
    'amazon_cleaned'[margin_percent] <= 0
)

```

// SOLUTION 1: PRICE INCREASE

// Products for price increase (margin > -10%)

Price Increase Products Count =

```
COUNTROWS(  
    FILTER(  
        'amazon_cleaned',  
        'amazon_cleaned'[estimated_profit] < 0 &&  
        'amazon_cleaned'[margin_percent] > -10  
    )  
)
```

Price Increase Current Price =

```
CALCULATE(  
    AVERAGE('amazon_cleaned'[actual_price]),  
    'amazon_cleaned'[estimated_profit] < 0 &&  
    'amazon_cleaned'[margin_percent] > -10  
)
```

Price Increase New Price = [Price Increase Current Price] * 1.2

Price Increase Expected Recovery =

[Price Increase Products Count] * [Price Increase Current Price] * 0.2 * 0.5 / 100000

// SOLUTION 2: DISCOUNT REDUCTION

// Products for discount reduction (discount > 25%)

Discount Reduce Products Count =

```
COUNTROWS(  
    FILTER(  
        'amazon_cleaned',  
        'amazon_cleaned'[estimated_profit] < 0 &&
```

```

        'amazon_cleaned'[discount_percentage] > 25
    )
)

```

Discount Reduce Current Discount =

```

CALCULATE(
    AVERAGE('amazon_cleaned'[discount_percentage]),
    'amazon_cleaned'[estimated_profit] < 0 &&
    'amazon_cleaned'[discount_percentage] > 25
)

```

Discount Reduce New Discount = 15

Discount Reduce Expected Recovery = [Discount Reduce Products Count] * 5000 / 100000

// SOLUTION 3: DISCONTINUE

// Products to discontinue (margin < -10% AND rating < 2.5)

Discontinue Products Count =

```

COUNTROWS(
    FILTER(
        'amazon_cleaned',
        'amazon_cleaned'[estimated_profit] < 0 &&
        'amazon_cleaned'[margin_percent] < -10 &&
        'amazon_cleaned'[rating] < 2.5
    )
)

```

Discontinue Loss Saved =

```

CALCULATE(
    SUM('amazon_cleaned'[estimated_profit]),

```

```
'amazon_cleaned'[estimated_profit] < 0 &&  
'amazon_cleaned'[margin_percent] < -10 &&  
'amazon_cleaned'[rating] < 2.5  
) / -100000
```

// SOLUTION 4: BUNDLE

// Products for bundle (margin between -10% and 0%)

Bundle Products Count =

```
COUNTROWS(  
  FILTER(  
    'amazon_cleaned',  
    'amazon_cleaned'[estimated_profit] < 0 &&  
    'amazon_cleaned'[margin_percent] >= -10 &&  
    'amazon_cleaned'[margin_percent] <= 0  
  )  
)
```

Bundle Products Loss =

```
CALCULATE(  
  SUM('amazon_cleaned'[estimated_profit]),  
  'amazon_cleaned'[estimated_profit] < 0 &&  
  'amazon_cleaned'[margin_percent] >= -10 &&  
  'amazon_cleaned'[margin_percent] <= 0  
) / -100000
```

Bundle Expected Recovery = [Bundle Products Loss] * 0.4

Bundle Avg Discount =

```
CALCULATE(  
  AVERAGE('amazon_cleaned'[discount_percentage]),
```

```
'amazon_cleaned'[estimated_profit] < 0 &&  
'amazon_cleaned'[margin_percent] >= -10 &&  
'amazon_cleaned'[margin_percent] <= 0  
)
```

Bundle Avg Rating =

```
CALCULATE(  
    AVERAGE('amazon_cleaned'[rating]),  
    'amazon_cleaned'[estimated_profit] < 0 &&  
    'amazon_cleaned'[margin_percent] >= -10 &&  
    'amazon_cleaned'[margin_percent] <= 0  
)
```

ADVANCED ANALYSIS MEASURES

// Weighted Average Margin

Weighted Avg Margin % =

```
DIVIDE(  
    SUMX('amazon_cleaned', 'amazon_cleaned'[margin_percent] *  
'amazon_cleaned'[estimated_revenue]),  
    SUM('amazon_cleaned'[estimated_revenue]),  
    0  
)
```

// Profit per Product

Avg Profit per Product =

```
DIVIDE(  
    SUM('amazon_cleaned'[estimated_profit]),  
    [Total Products],  
    0  
)
```

// Margin Health Score (0-100)

Margin Health Score =

VAR AvgMarg = [Avg Margin %]

VAR TargetMarg = 20

VAR Score = (AvgMarg / TargetMarg) * 100

RETURN IF(Score > 100, 100, Score)

// Profit by Price Segment

Profit by Price Segment =

SUMMARIZE(

'amazon_cleaned',

'amazon_cleaned'[Price Segment],

"Profit", SUM('amazon_cleaned'[estimated_profit])/100000

)

// Profit by Rating Category

Profit by Rating =

SUMMARIZE(

'amazon_cleaned',

'amazon_cleaned'[Rating Category],

"Profit", SUM('amazon_cleaned'[estimated_profit])/100000

)

// Profit by Margin Tier

Profit by Margin Tier =

SUMMARIZE(

'amazon_cleaned',

'amazon_cleaned'[Margin Tier],

"Profit", SUM('amazon_cleaned'[estimated_profit])/100000

)

