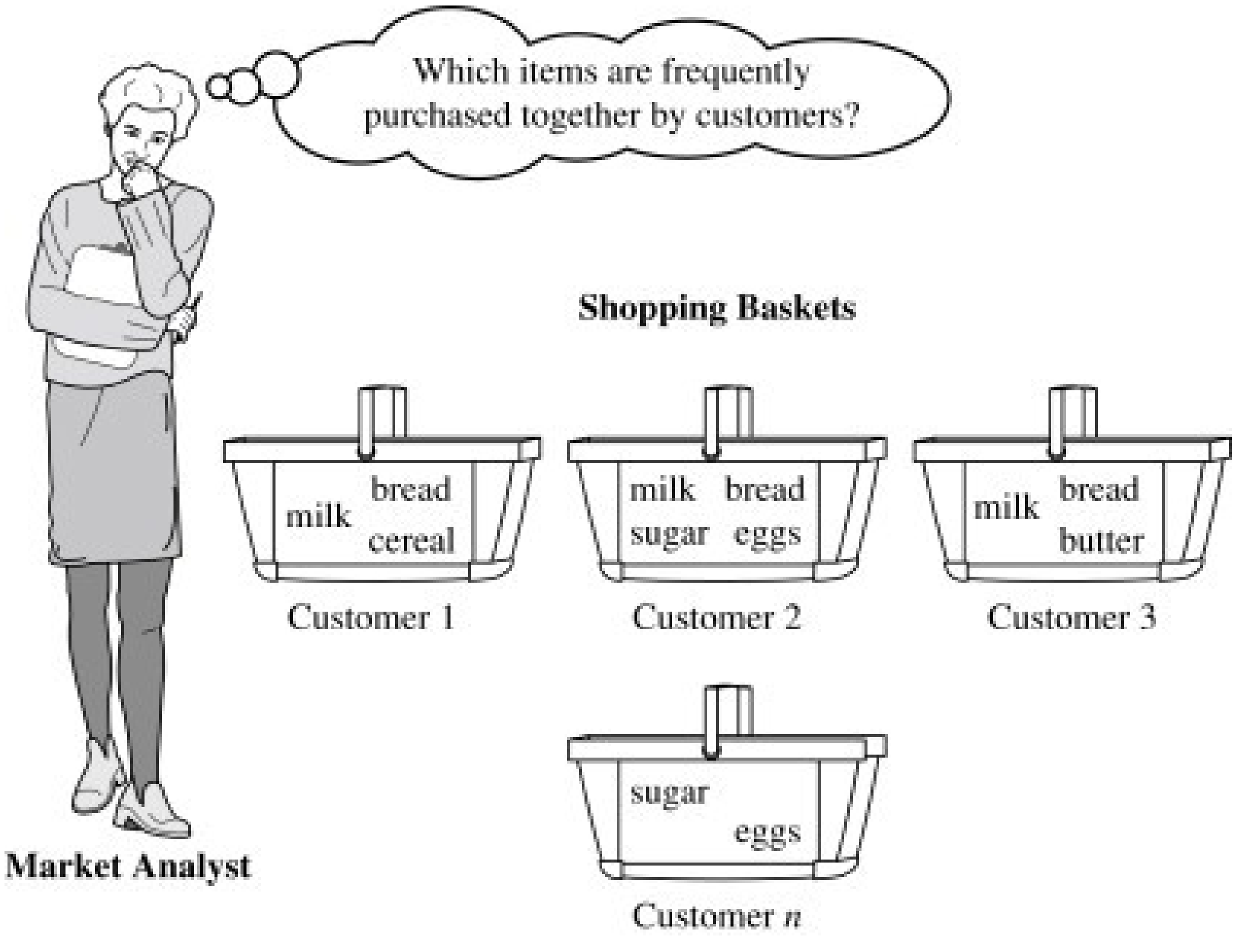
MARKET BASKET ANALYSIS

PHASE 4 SUBMISSION DOCUMENT

DEVELOPMENT PART 2

TOPIC : continue building the project by performing different activities like feature engineering, model training, evaluation etc



Introduction :

Market Basket Analysis (MBA), also known as Association Rule Mining or Affinity Analysis, is a data mining and analytical technique used in retail and e-commerce to discover patterns, relationships, and associations among products that customers frequently purchase together.

The primary goal of MBA is to uncover insights into customer buying behavior, enhance marketing strategies, and optimize product placement in stores.

Given detaset :

	BillNo	Itemname	Quantity	Date	Price	CustomerID	Country
0	536365	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...
195	536389	CHRISTMAS LIGHTS 10 REINDEER	6	2010-12-01 10:03:00	8.50	12431.0	Australia
196	536389	VINTAGE UNION JACK CUSHION COVER	8	2010-12-01 10:03:00	4.95	12431.0	Australia
197	536389	VINTAGE HEADS AND TAILS CARD GAME	12	2010-12-01 10:03:00	1.25	12431.0	Australia
198	536389	SET OF 3 COLOURED FLYING DUCKS	6	2010-12-01 10:03:00	5.45	12431.0	Australia
199	536389	SET OF 3 GOLD FLYING DUCKS	4	2010-12-01 10:03:00	6.35	12431.0	Australia

200 rows × 7 columns

Feature Engineering:

In a Market Basket Analysis project, feature engineering is mainly about transforming your data into a suitable format for association rule mining. We've already done some of this during the preprocessing step. However, you might consider adding features like the number of items in a transaction, the total amount spent, or the time of the transaction if applicable to your dataset.

Feature Selection:

Identify the most relevant features (product attributes) for your analysis. You may consider factors like product category, price, brand, and customer demographics.

Creating Association Rules:

Implement the chosen association rule mining algorithm (e.g., Apriori or FP-growth) to generate frequent itemsets and association rules.

In [75]:
```
# Split the 'Itemname' column into individual items
items_df = transaction_data['Itemname'].str.split(', ', expand=True)

# Concatenate the original DataFrame with the new items DataFrame
transaction_data = pd.concat([transaction_data, items_df], axis=1)

# Drop the original 'Itemname' column
transaction_data = transaction_data.drop('Itemname', axis=1)

# Display the resulting DataFrame
print(transaction_data.head())
```

Association Rules:

Once you've identified frequent itemsets using Apriori, you can generate association rules. This code will generate rules with a minimum confidence of 0.5 (adjust as needed):

rules = association_rules(frequent_itemsets, metric='lift', min_threshold=1.0)

The 'lift' metric measures the strength of association between items. You can use other metrics like 'confidence' or 'support' depending on your needs.

In [82]:

```python
# Load transaction data into a DataFrame
df_encoded = pd.read_csv('transaction_data_encoded.csv')

# Association Rule Mining
frequent_itemsets = apriori(df_encoded, min_support=0.007, use_colnames=True)
rules = association_rules(frequent_itemsets, metric="confidence", min_threshold=0.5)

# Display information of the rules
print("Association Rules:")
print(rules.head())
```

```
Association Rules:
                          antecedents                          consequents  \
0              (CHOCOLATE BOX RIBBONS)           (6 RIBBONS RUSTIC CHARM)
1   (60 CAKE CASES DOLLY GIRL DESIGN)  (PACK OF 72 RETROSPOT CAKE CASES)
2       (60 TEATIME FAIRY CAKE CASES)  (PACK OF 72 RETROSPOT CAKE CASES)
3   (ALARM CLOCK BAKELIKE CHOCOLATE)        (ALARM CLOCK BAKELIKE GREEN)
4   (ALARM CLOCK BAKELIKE CHOCOLATE)         (ALARM CLOCK BAKELIKE PINK)

   antecedent support  consequent support   support  confidence       lift  \
0            0.012368            0.039193  0.007036    0.568889  14.515044
1            0.018525            0.054529  0.010059    0.543027   9.958409
2            0.034631            0.054529  0.017315    0.500000   9.169355
3            0.017150            0.042931  0.011379    0.663462  15.454151
4            0.017150            0.032652  0.009125    0.532051  16.294742

   leverage  conviction  zhangs_metric
0  0.006551    2.228676       0.942766
1  0.009049    2.068984       0.916561
2  0.015427    1.890941       0.922902
3  0.010642    2.843862       0.951613
4  0.008565    2.067210       0.955009
```

Model Training - Apriori Algorithm:

- **Algorithm Selection**: Discuss the choice of the association rule mining algorithm. Explain why you selected a particular algorithm and how it aligns with the project goals.

- **Data Split**: Split the dataset into training and testing sets (if applicable). Describe your approach to data partitioning.
- **Model Training**: Train the association rule mining model on the training data.

The Apriori algorithm is commonly used for association rule mining. You can use the mlxtend library to apply this algorithm:

```
from mlxtend.frequent_patterns import apriori
from mlxtend.frequent_patterns import association_rules

# Perform market basket analysis using Apriori
frequent_itemsets = apriori(basket.drop('TransactionID', axis=1),
min_support=0.01, use_colnames=True)
```

In this code, min_support specifies the minimum support threshold for itemsets. Adjust this value according to your dataset and analysis requirements.

Model Evaluation:

To evaluate the quality of the association rules, you can consider various metrics like lift, confidence, and support. You might also filter the rules based on these metrics to retain only those that are most interesting or relevant

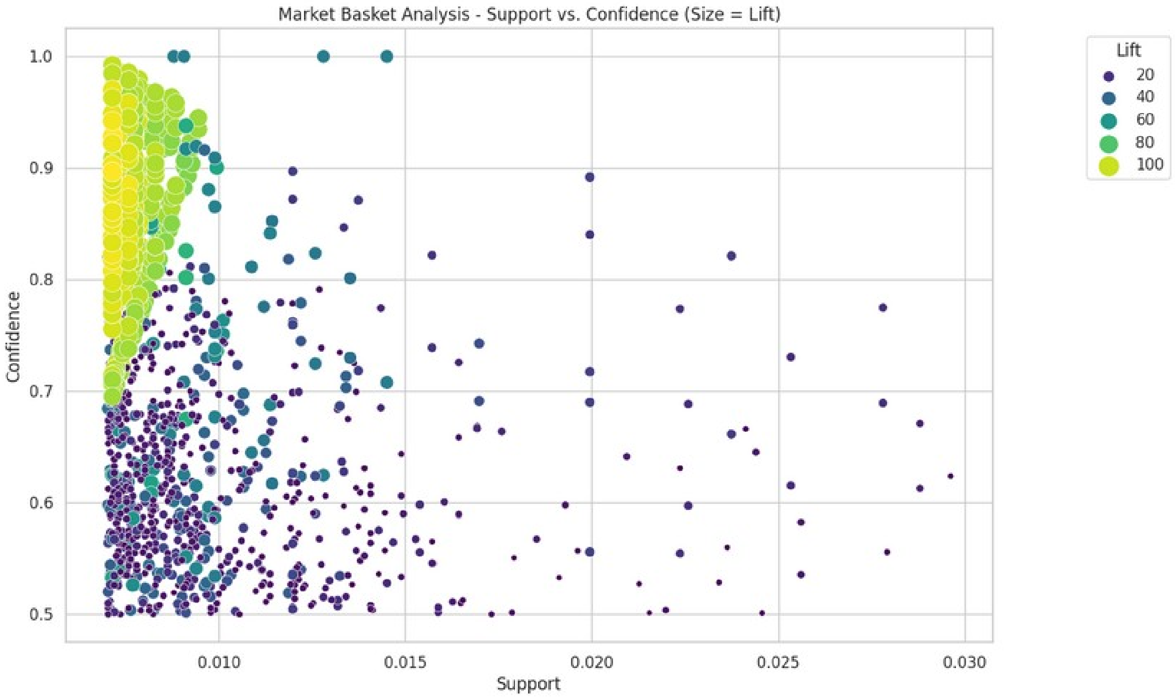
Performance Metrics: Report the model's performance based on the selected metrics. Present numerical results and provide interpretations. Discuss the implications of high or low values for support, confidence, and lift.
Quality of Rules: Evaluate the quality and relevance of the generated association rules. Discuss the significance of the rules, such as identifying strong associations or discovering unexpected patterns. Consider the potential for actionable insights.

Visualizations: If appropriate, create visualizations to support your findings. Examples include bar charts showing rule metrics, network graphs illustrating item associations, or heatmaps depicting rule relationships.

Visualization:

Create visualizations to represent the discovered association rules. For instance, you can visualize support vs. confidence or lift vs. support to identify significant rules.

In [78]:
```
import matplotlib.pyplot as plt
import seaborn as sns

# Plot scatterplot for Support vs. Confidence
plt.figure(figsize=(12, 8))
sns.scatterplot(x="support", y="confidence", size="lift", data=rules, hue="lift", palette="viridis", sizes=(20, 200))
plt.title('Market Basket Analysis - Support vs. Confidence (Size = Lift)')
plt.xlabel('Support')
plt.ylabel('Confidence')
plt.legend(title='Lift', loc='upper right', bbox_to_anchor=(1.2, 1))
plt.show()
```

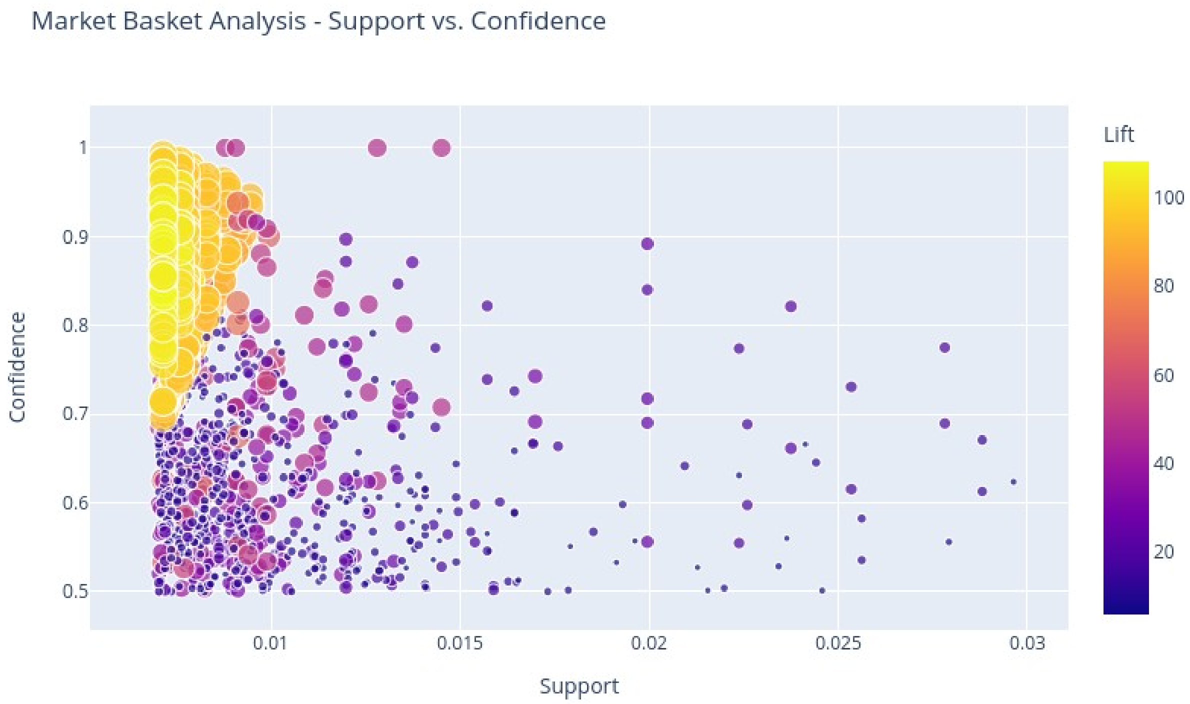


Interactive Market Basket Analysis Visualization

We leverage the Plotly Express library to create an interactive scatter plot visualizing the results of the market basket analysis. This plot provides an interactive exploration of the relationship between support, confidence, and lift for the generated association rules.

In [11]:

```
import plotly.express as px

# Convert frozensets to lists for serialization
rules['antecedents'] = rules['antecedents'].apply(list)
rules['consequents'] = rules['consequents'].apply(list)

# Create an interactive scatter plot using plotly express
fig = px.scatter(rules, x="support", y="confidence", size="lift",
                 color="lift", hover_name="consequents",
                 title='Market Basket Analysis - Support vs. Confidence',
                 labels={'support': 'Support', 'confidence': 'Confidence'})

# Customize the layout
fig.update_layout(
    xaxis_title='Support',
    yaxis_title='Confidence',
    coloraxis_colorbar_title='Lift',
    showlegend=True
)

# Show the interactive plot
fig.show()
```



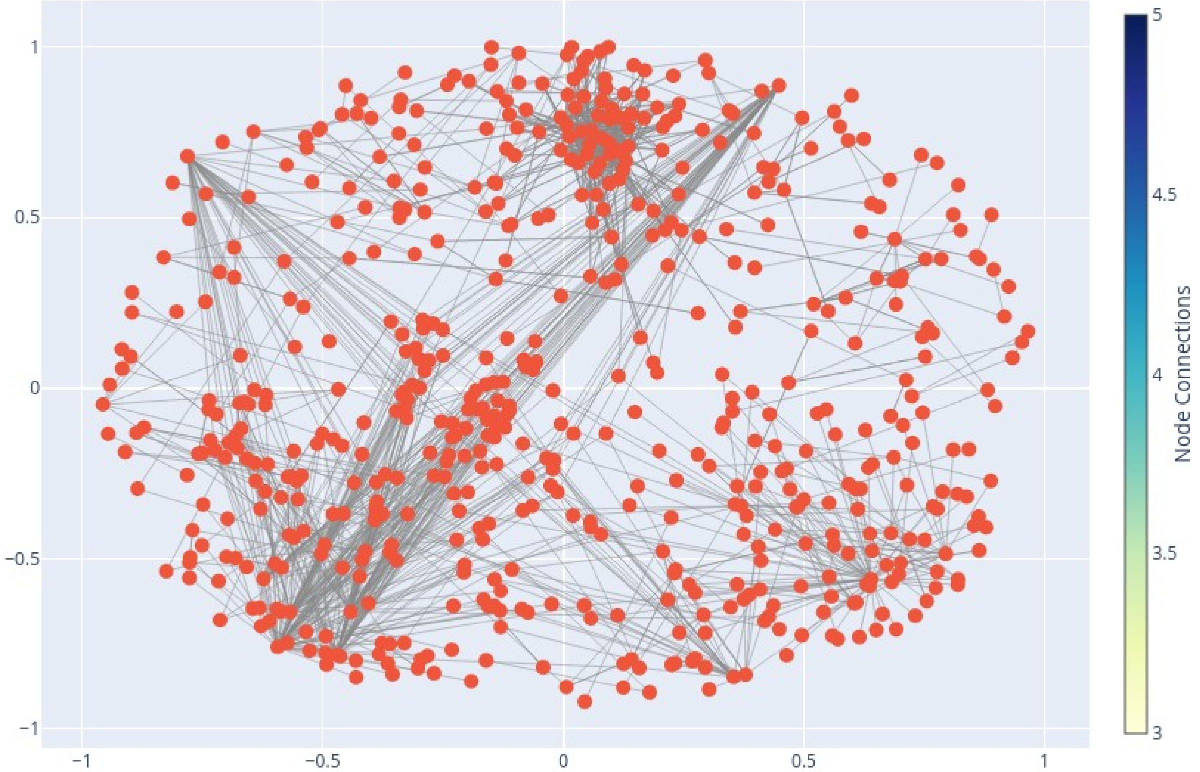
Interactive Network Visualization for Association Rules

We utilize the NetworkX and Plotly libraries to create an interactive network graph visualizing the association rules. This graph represents relationships between antecedent and consequent items, showcasing support as edge weights.

In [12]:

```
import networkx as nx
import matplotlib.pyplot as plt
import plotly.graph_objects as go

# Create a directed graph
G = nx.DiGraph()

# Add nodes and edges from association rules
for idx, row in rules.iterrows():
    G.add_node(tuple(row['antecedents']), color='skyblue')
    G.add_node(tuple(row['consequents']), color='orange')
    G.add_edge(tuple(row['antecedents']), tuple(row['consequents']), weight=row['support'])

# Set node positions using a spring layout
pos = nx.spring_layout(G)

# Create an interactive plot using plotly
edge_x = []
edge_y = []
for edge in G.edges(data=True):
    x0, y0 = pos[edge[0]]
    x1, y1 = pos[edge[1]]
    edge_x.append(x0)
    edge_x.append(x1)
    edge_x.append(None)
    edge_y.append(y0)
    edge_y.append(y1)
    edge_y.append(None)

edge_trace = go.Scatter(
    x=edge_x, y=edge_y,
    line=dict(width=0.5, color='#888'),
    hoverinfo='none',
    mode='lines')

node_x = []
node_y = []
for node in G.nodes():
    x, y = pos[node]
    node_x.append(x)
    node_y.append(y)

node_trace = go.Scatter(
    x=node_x, y=node_y,
    mode='markers',
    hoverinfo='text',
    marker=dict(
        showscale=True,
```

```
        colorscale='YlGnBu',
        size=10,
        colorbar=dict(
            thickness=15,
            title='Node Connections',
            xanchor='left',
            titleside='right'
        )
    )
)

# Customize the layout
layout = go.Layout(
    showlegend=False,
    hovermode='closest',
    margin=dict(b=0, l=0, r=0, t=0),
)

# Create the figure
fig = go.Figure(data=[edge_trace, node_trace], layout=layout)

# Show the interactive graph
fig.show()
```

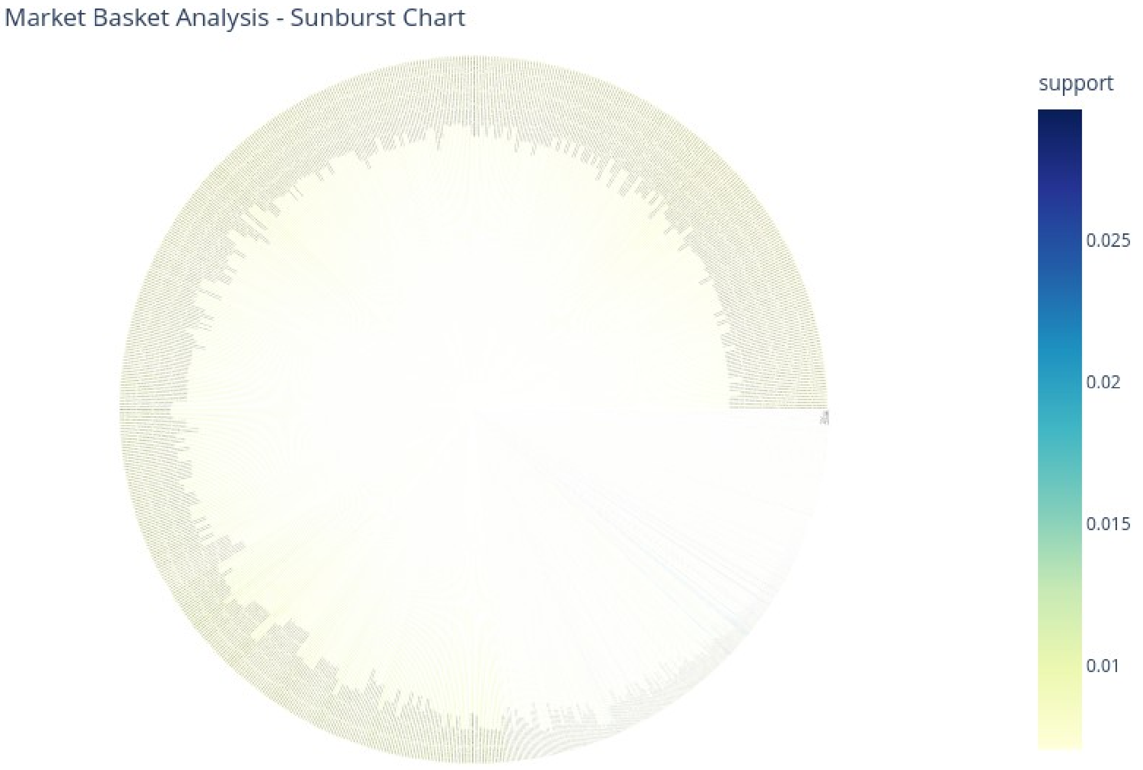


Interactive Sunburst Chart for Association Rules

We use Plotly Express to create an interactive sunburst chart visualizing association rules. This chart represents the relationships between antecedent and consequent items, showcasing lift as well as support through color intensity.

In [13]:

```
import plotly.express as px

# Combine antecedents and consequents into a single column for each rule
rules['rule'] = rules['antecedents'].astype(str) + ' -> ' + rules['consequents'].astype(str)

# Create a sunburst chart
fig = px.sunburst(rules, path=['rule'], values='lift',
                  title='Market Basket Analysis - Sunburst Chart',
                  color='support', color_continuous_scale='YlGnBu')

# Customize the layout
fig.update_layout(
    margin=dict(l=0, r=0, b=0, t=40),
)

# Show the interactive plot
fig.show()
```



Actionability and Recommendations:

- **Interpretation**: Interpret the findings from your analysis in a business context. Explain what the association rules reveal about customer purchasing behavior. Offer insights into common purchasing patterns, complementary products, or cross-selling opportunities.
- **Business Impact**: Discuss how the results can impact the business. Highlight potential improvements in marketing strategies, inventory management, or customer experience. Quantify the expected benefits or changes.
- **Recommendations**: Provide actionable recommendations based on the associations discovered. For example, if certain products frequently co-occur in transactions, recommend bundling or promoting them together. Suggest specific strategies for product placement or cross-selling.

Conclusion

In the conclusion phase of your Market Basket Analysis project, you should aim to provide a comprehensive and insightful summary of the work you've done and the significance of your findings. This is where you tie together all the pieces of your analysis and explain the implications for the business.