

```
In [3]: !pip install pygltflib pandas

Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pygltflib in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (1.16.3)
Requirement already satisfied: pandas in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (1.5.3)
Requirement already satisfied: dataclasses-json<0.6.25 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pygltflib) (0.6.7)
Requirement already satisfied: deprecated in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pygltflib) (1.2.18)
Requirement already satisfied: python-dateutil<=2.8.1 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz<=2020.1 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pandas) (2024.2)
Requirement already satisfied: numpy<=1.20.3 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pandas) (1.24.4)
Requirement already satisfied: marshmallow<4.0.0,>=3.18.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from dataclasses-json<=0.6.25->pygltflib) (3.26.1)
Requirement already satisfied: typing-inspect<1,>=0.4.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from dataclasses-json<=0.6.25->pygltflib) (0.9.0)
Requirement already satisfied: six<=1.5 in /Library/Developer/CommandLineTools/Library/Frameworks/Python3.framework/Versions/3.9/lib/python3.9/site-packages (from python-dateutil<=2.8.1->pandas) (1.15.0)
Requirement already satisfied: wrapt<2,>=1.10 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from deprecated->pygltflib) (1.17.2)
Requirement already satisfied: packaging<=17.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from marshmallow<4.0.0,>=3.18.0->dataclasses-json<=0.6.25->pygltflib) (24.2)
Requirement already satisfied: numpy<=1.20.3 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from typing-inspect<1,>=0.4.0->dataclasses-json<=0.6.25->pygltflib) (1.0.0)
Requirement already satisfied: typing-extensions<=3.7.4 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from typing-inspect<1,>=0.4.0->dataclasses-json<=0.6.25->pygltflib) (4.12.2)
```

```
In [4]: import os
from pygltflib import GLTF2
import pandas as pd

def extract_glb_info(glb_file):

    gltf = GLTF2().load(glb_file)

    asset = gltf.asset
    info = {
        "Filename": os.path.basename(glb_file),
        "Asset Version": asset.version,
        "Asset Generator": asset.generator,
        "Number of Scenes": len(gltf.scenes) if gltf.scenes else 0,
        "Number of Nodes": len(gltf.nodes) if gltf.nodes else 0,
        "Number of Meshes": len(gltf.meshes) if gltf.meshes else 0,
        "Number of Materials": len(gltf.materials) if gltf.materials else 0,
        "Number of Animations": len(gltf.animations) if gltf.animations else 0,
        "Number of Textures": len(gltf.textures) if gltf.textures else 0,
        "Number of Images": len(gltf.images) if gltf.images else 0,
        "Number of Buffers": len(gltf.buffers) if gltf.buffers else 0,
        "Extensions Used": gltf.extensionsUsed if gltf.extensionsUsed else [],
        "Extensions Required": gltf.extensionsRequired if gltf.extensionsRequired else []
    }

    return info
```

```
In [5]: model_folder = "My3DModelsEDA"
```

```
glb_files = [os.path.join(model_folder, f) for f in os.listdir(model_folder) if f.lower().endswith(".glb")]
print(f"Found {len(glb_files)} GLB models in '{model_folder}'.")
```

```
metadata_list = []
for file in glb_files:
    try:
        metadata = extract_glb_info(file)
        metadata_list.append(metadata)
    except Exception as e:
        print(f"Error processing {file}: {e}")
```

```
df = pd.DataFrame(metadata_list)
df.head()
```

Found 205 GLB models in 'My3DModelsEDA'.

Out[5]:	Filename	Asset Version	Asset Generator	Number of Scenes	Number of Nodes	Number of Meshes	Number of Materials	Number of Animations	Number of Textures	Number of Images	Number of Buffers	Extensions Used	Extensions Required
0	The_Morning_Room.glb	2.0	pygltflib@v1.16.3	1	30	13	1	0	1	1	1	[]	[]
1	angkor_wat_temple_cambodia.glb	2.0	Sketchfab-12.68.0	1	27	23	1	0	1	1	1	[]	[]
2	terracotta_warrior.glb	2.0	Sketchfab-19.93.0	1	11	7	1	0	3	3	1	[]	[]
3	parthenon_-_acropolis_athens_greece.glb	2.0	Sketchfab-12.68.0	1	13	8	2	0	2	2	1	[]	[]
4	Cathedral.glb	2.0	pygltflib@v1.16.3	1	70	33	2	0	4	4	1	[]	[]

```
In [6]: import matplotlib.pyplot as plt
import seaborn as sns
```

```
sns.set(style="whitegrid")
```

```
print("Summary Statistics:")
print(df.describe(include='all'))
```

```
plt.figure(figsize=(10, 6))
sns.histplot(df['Number of Meshes'], bins=10, kde=True, color='skyblue')
plt.title("Distribution of Number of Meshes in 3D Models")
plt.xlabel("Number of Meshes")
plt.ylabel("Frequency")
plt.show()
```

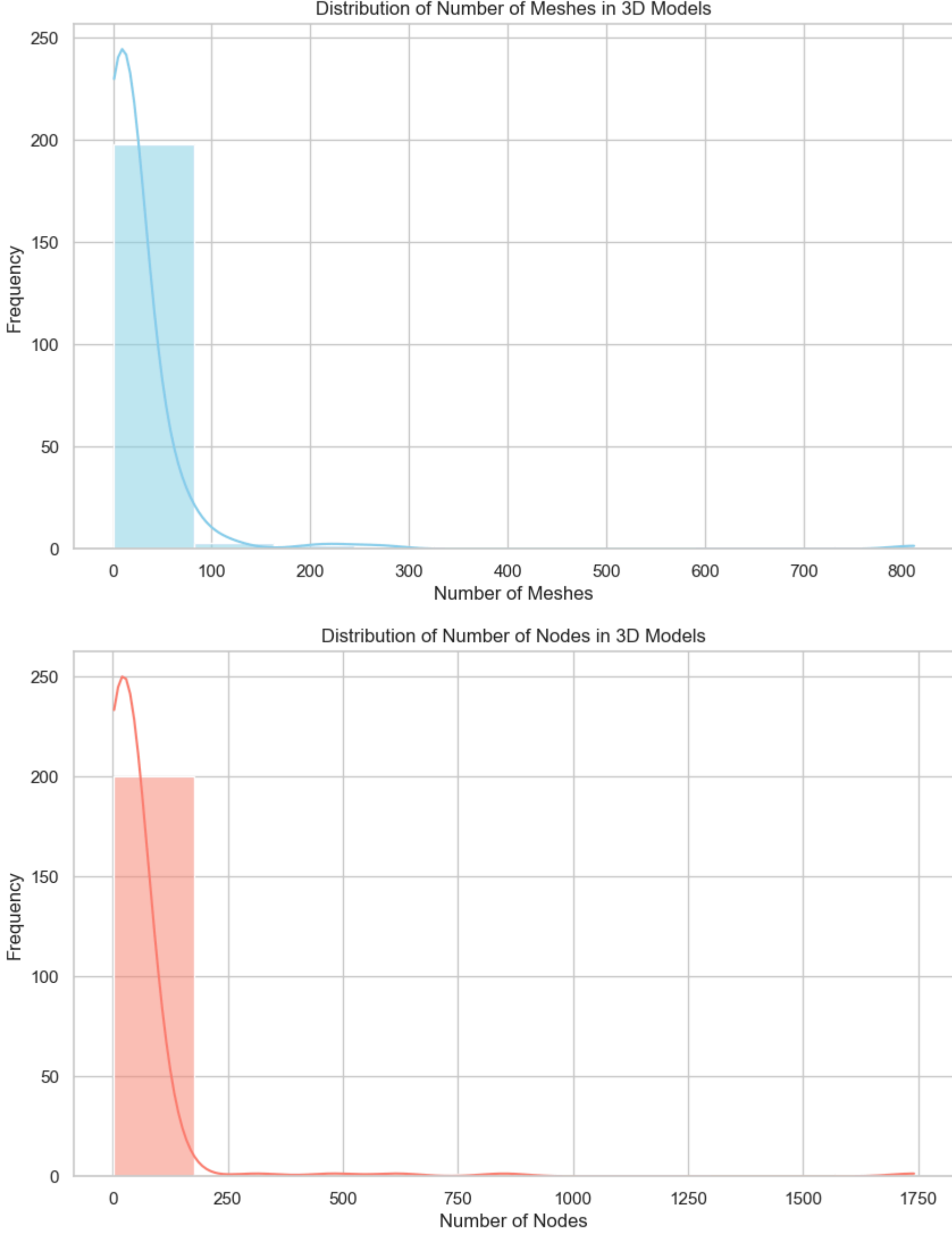
```
plt.figure(figsize=(10, 6))
sns.histplot(df['Number of Nodes'], bins=10, kde=True, color='salmon')
plt.title("Distribution of Number of Nodes in 3D Models")
plt.xlabel("Number of Nodes")
plt.ylabel("Frequency")
plt.show()
```

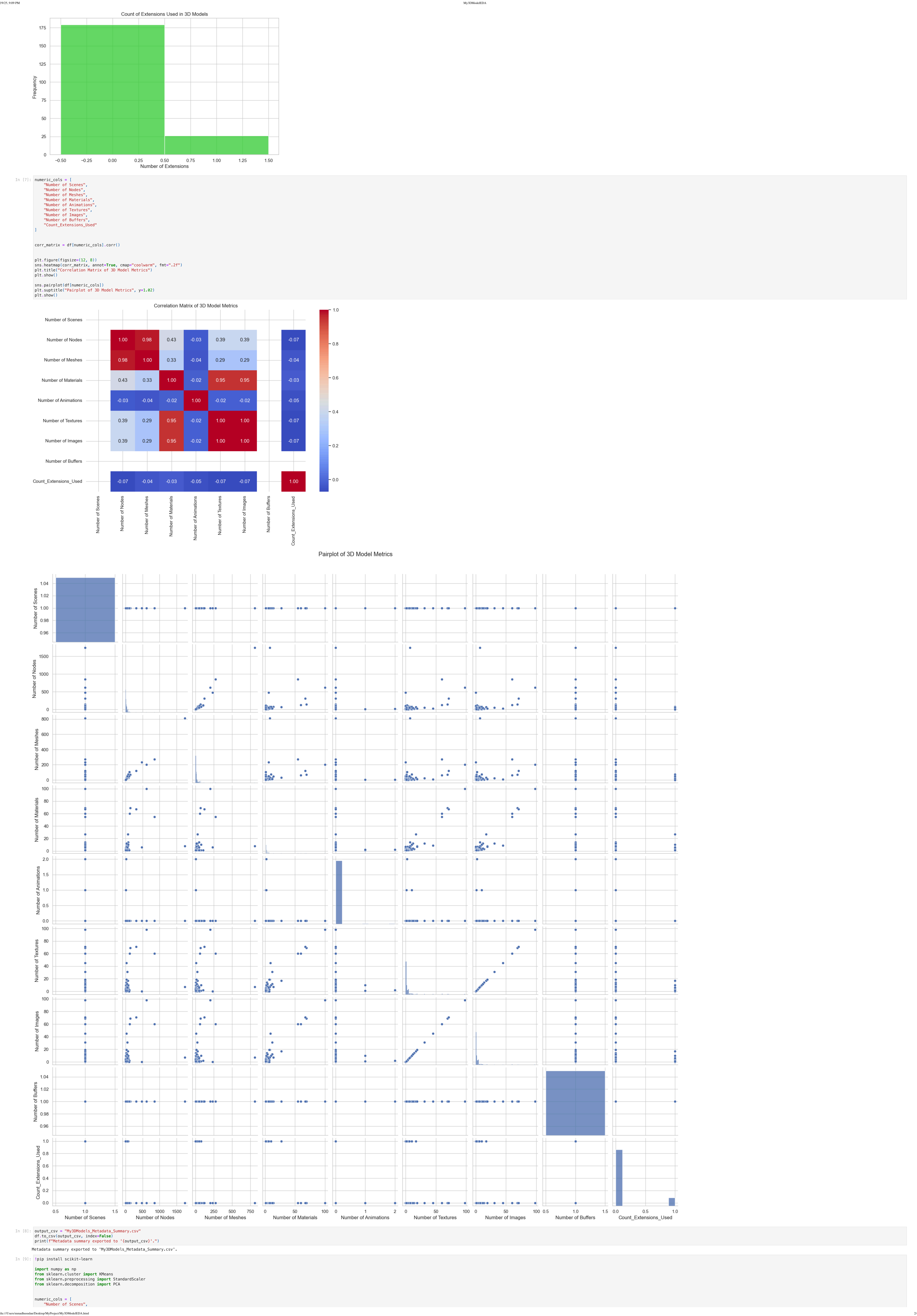
```
df['Count_Extensions_Used'] = df['Extensions Used'].apply(lambda x: len(x) if isinstance(x, list) else 0)
```

```
plt.figure(figsize=(10, 6))
sns.histplot(df['Count_Extensions_Used'], bins=range(0, df['Count_Extensions_Used'].max() + 2), discrete=True, color='limegreen')
plt.title("Count of Extensions Used in 3D Models")
plt.xlabel("Number of Extensions")
plt.ylabel("Frequency")
plt.show()
```

Summary Statistics:

	Filename	Asset Version	Asset Generator	\
count	205	205	205	
unique	205	1	30	
top	The_Morning_Room.glb	2.0	pygltflib@v1.16.3	NaN
freq	1	205	130	
mean	NaN	NaN	NaN	
std	NaN	NaN	NaN	
min	NaN	NaN	NaN	
25%	NaN	NaN	NaN	
50%	NaN	NaN	NaN	
75%	NaN	NaN	NaN	
max	NaN	NaN	NaN	
	Number of Scenes	Number of Nodes	Number of Meshes	\
count	205.0	205.000000	205.000000	
unique	NaN	NaN	NaN	
top	NaN	NaN	NaN	
freq	NaN	NaN	NaN	
mean	1.0	46.497561	23.048780	
std	0.0	145.326833	64.788476	
min	1.0	3.000000	1.000000	
25%	1.0	10.000000	3.000000	
50%	1.0	18.000000	9.000000	
75%	1.0	38.000000	19.000000	
max	1.0	1741.000000	812.000000	
	Number of Materials	Number of Animations	Number of Textures	\
count	205.000000	205.000000	205.000000	
unique	NaN	NaN	NaN	
top	NaN	NaN	NaN	
freq	NaN	NaN	NaN	
mean	3.785366	0.029268	4.707317	
std	11.181777	0.219451	11.811518	
min	1.000000	0.000000	0.000000	
25%	1.000000	0.000000	1.000000	
50%	1.000000	0.000000	1.000000	
75%	2.000000	0.000000	4.000000	
max	100.000000	2.000000	98.000000	
	Number of Images	Number of Buffers	Extensions Used	\
count	205.000000	205.0	205	
unique	NaN	NaN	4	
top	NaN	NaN	[]	
freq	NaN	NaN	179	
mean	4.707317	1.0	NaN	
std	11.811518	0.0	NaN	
min	0.000000	1.0	NaN	
25%	1.000000	1.0	NaN	
50%	1.000000	1.0	NaN	
75%	4.000000	1.0	NaN	
max	98.000000	1.0	NaN	
	Extensions Required			
count	205			
unique	2			
top	[]			
freq	203			
mean	NaN			
std	NaN			
min	NaN			
25%	NaN			
50%	NaN			
75%	NaN			
max	NaN			






```

    "Number of Nodes",
    "Number of Meshes",
    "Number of Materials",
    "Number of Animations",
    "Number of Textures",
    "Number of Images",
    "Number of Buffers",
    "Count_Extensions_Used"
]

features = df[numeric_cols].fillna(0)
scaler = StandardScaler()
features_scaled = scaler.fit_transform(features)

inertia = []
K = range(1, 10)
for k in K:
    kmeans = KMeans(n_clusters=k, random_state=42)
    kmeans.fit(features_scaled)
    inertia.append(kmeans.inertia_)

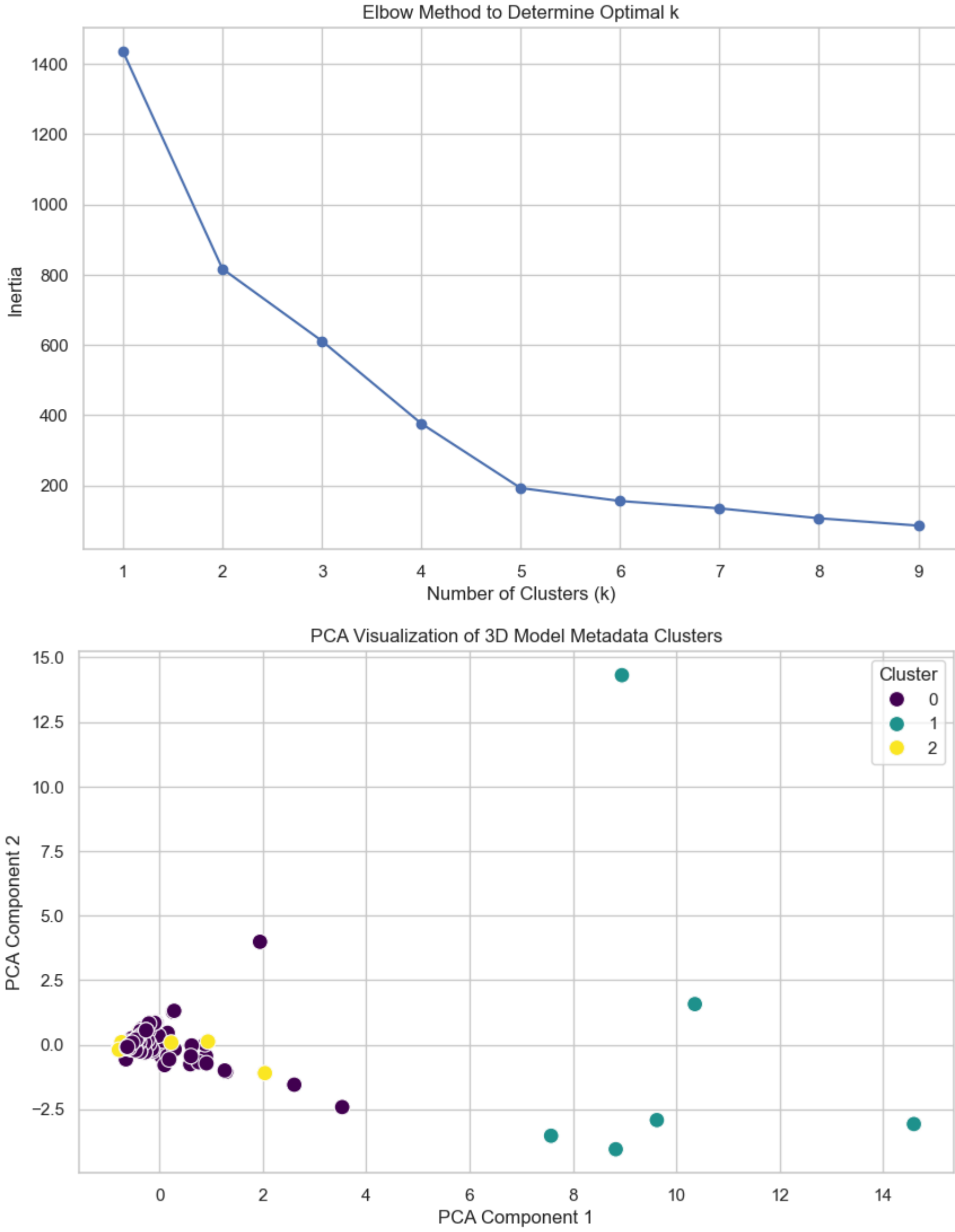
plt.figure(figsize=(10, 6))
plt.plot(K, inertia, 'bo-')
plt.xlabel('Number of Clusters (k)')
plt.ylabel('Inertia')
plt.title('Elbow Method to Determine Optimal K')
plt.show()

k_opt = 3
kmeans = KMeans(n_clusters=k_opt, random_state=42)
df['Cluster'] = kmeans.fit_predict(features_scaled)

pca = PCA(n_components=2)
pca_components = pca.fit_transform(features_scaled)
df['PCA1'] = pca_components[:, 0]
df['PCA2'] = pca_components[:, 1]

plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='PCA1', y='PCA2', hue='Cluster', palette='viridis', s=100)
plt.title("PCA Visualization of 3D Model Metadata Clusters")
plt.xlabel("PCA Component 1")
plt.ylabel("PCA Component 2")
plt.show()
```

Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: scikit-learn in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (1.6.1)
Requirement already satisfied: numpy<=1.19.5 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from scikit-learn) (1.24.4)
Requirement already satisfied: scipy<=1.6.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from scikit-learn) (1.10.1)
Requirement already satisfied: joblib<=1.2.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl<=3.1.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from scikit-learn) (3.5.0)



In [21]: !pip install trimesh pyvista ipywidgets

Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: trimesh in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (4.6.1)
Requirement already satisfied: pyvista in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (0.44.2)
Requirement already satisfied: ipywidgets in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (8.1.5)
Requirement already satisfied: numpy<=1.20 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from trimesh) (1.24.4)
Requirement already satisfied: matplotlib<=3.6.1 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pyvista) (3.9.4)
Requirement already satisfied: pillow in /Users/Python/3.9/lib/python/site-packages (from pyvista) (11.1.0)
Requirement already satisfied: pooch in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pyvista) (1.8.2)
Requirement already satisfied: scooby<=0.5.1 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pyvista) (0.10.0)
Requirement already satisfied: vtk<=9.4.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pyvista) (9.3.1)
Requirement already satisfied: typing-extensions in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pyvista) (4.12.2)
Requirement already satisfied: comm<0.1.3 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipywidgets) (0.2.2)
Requirement already satisfied: ipython<=6.1.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipywidgets) (8.18.1)
Requirement already satisfied: decorator in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (5.14.3)
Requirement already satisfied: widgetsnbextension<=4.0.12 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipywidgets) (4.0.13)
Requirement already satisfied: jupyterlab-widgets<=3.0.12 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipywidgets) (3.0.13)
Requirement already satisfied: ipykernel in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (5.1.1)
Requirement already satisfied: jedi<=0.16 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (0.19.2)
Requirement already satisfied: matplotlib-inline in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (0.1.7)
Requirement already satisfied: prompt-toolkit<3.1.0,=>3.0.41 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (3.0.50)
Requirement already satisfied: pygments<=2.4.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (2.19.1)
Requirement already satisfied: stack-data in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (0.6.3)
Requirement already satisfied: exceptiongroup in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (1.2.2)
Requirement already satisfied: pexpect<=4.3 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (4.9.0)
Requirement already satisfied: contourpy<=1.0.1 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from matplotlib<=3.6.1->pyvista) (1.3.0)
Requirement already satisfied: cycler<=0.10 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from matplotlib<=3.6.1->pyvista) (0.12.1)
Requirement already satisfied: fonttools<=4.22.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from matplotlib<=3.6.1->pyvista) (4.55.3)
Requirement already satisfied: kiwisolver<=1.3.1 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from matplotlib<=3.6.1->pyvista) (1.4.7)
Requirement already satisfied: packaging<=20.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from matplotlib<=3.6.1->pyvista) (24.2)
Requirement already satisfied: pyparsing<=2.3.1 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from matplotlib<=3.6.1->pyvista) (3.2.1)
Requirement already satisfied: python-dateutil<=2.7 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from matplotlib<=3.6.1->pyvista) (2.9.0.post0)
Requirement already satisfied: importlib-resources<=3.2.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from matplotlib<=3.6.1->pyvista) (6.5.2)
Requirement already satisfied: platformdirs<=2.5.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pooch->pyvista) (4.3.6)
Requirement already satisfied: requests<=2.19.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from pooch->pyvista) (2.32.3)
Requirement already satisfied: zipp<=3.1.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from importlib-resources<=3.2.0->matplotlib<=3.6.1->pyvista) (3.21.0)
Requirement already satisfied: parso<0.9.0,=>0.8.4 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from jedi<=0.16->ipython<=6.1.0->ipywidgets) (0.8.4)
Requirement already satisfied: wcwidth in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from prompt-toolkit<3.1.0,=>3.0.41->ipython<=6.1.0->ipywidgets) (0.2.13)
Requirement already satisfied: six<=1.5 in /Library/Developer/CommandLineTools/Library/Frameworks/Python3.framework/Versions/3.9/lib/python3.9/site-packages (from python-dateutil<=2.7->matplotlib<=3.6.1->pyvista) (1.15.0)
Requirement already satisfied: charset-normalizer<4,=>2 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from requests<=2.19.0->pooch->pyvista) (3.4.1)
Requirement already satisfied: idna<=4,=>2.5 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from requests<=2.19.0->pooch->pyvista) (3.10)
Requirement already satisfied: urllib3<3,=>1.21.1 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from requests<=2.19.0->pooch->pyvista) (2.3.0)
Requirement already satisfied: certifi<=2017.4.17 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from requests<=2.19.0->pooch->pyvista) (2024.12.14)
Requirement already satisfied: executing<=1.2.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from stack-data->ipython<=6.1.0->ipywidgets) (2.1.0)
Requirement already satisfied: asttokens<=2.1.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from stack-data->ipython<=6.1.0->ipywidgets) (3.0.0)
Requirement already satisfied: pure-eval in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from stack-data->ipython<=6.1.0->ipywidgets) (0.2.3)

In [11]: !pip install ipywidgets

Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: ipywidgets in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (8.1.5)
Requirement already satisfied: comm<0.1.3 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipywidgets) (0.2.2)
Requirement already satisfied: ipython<=6.1.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipywidgets) (8.18.1)
Requirement already satisfied: traitlets<=4.3.1 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipywidgets) (5.14.3)
Requirement already satisfied: widgetsnbextension<=4.0.12 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipywidgets) (4.0.13)
Requirement already satisfied: jupyterlab-widgets<=3.0.12 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipywidgets) (3.0.13)
Requirement already satisfied: decorator in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (5.1.1)
Requirement already satisfied: jedi<=0.16 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (0.19.2)
Requirement already satisfied: matplotlib-inline in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (0.1.7)
Requirement already satisfied: prompt-toolkit<3.1.0,=>3.0.41 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (3.0.50)
Requirement already satisfied: pygments<=2.4.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (2.19.1)
Requirement already satisfied: stack-data in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (0.6.3)
Requirement already satisfied: typing-extensions in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (4.12.2)
Requirement already satisfied: exceptiongroup in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (1.2.2)
Requirement already satisfied: pexpect<=4.3 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from ipython<=6.1.0->ipywidgets) (4.9.0)
Requirement already satisfied: parso<0.9.0,=>0.8.4 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from jedi<=0.16->ipython<=6.1.0->ipywidgets) (0.8.4)
Requirement already satisfied: wcwidth in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from prompt-toolkit<3.1.0,=>3.0.41->ipython<=6.1.0->ipywidgets) (0.2.13)
Requirement already satisfied: executing<=1.2.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from stack-data->ipython<=6.1.0->ipywidgets) (2.1.0)
Requirement already satisfied: asttokens<=2.1.0 in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from stack-data->ipython<=6.1.0->ipywidgets) (3.0.0)
Requirement already satisfied: pure-eval in /Users/mmadhusudan/Library/Python/3.9/lib/python/site-packages (from stack-data->ipython<=6.1.0->ipywidgets) (0.2.3)

In [35]: import ipywidgets as widgets
from IPython.display import display, clear_output

```
unique_clusters = sorted(df['Cluster'].unique())
cluster_dropdown = widgets.Dropdown(
    options=[All] + unique_clusters,
    value=All,
    description='Cluster:',
)

model_dropdown = widgets.Dropdown(
    options=df['Filename'].tolist(),
    description='Model:',
)

dashboard_output = widgets.Output()

def update_model_dropdown(args):
    selected_cluster = cluster_dropdown.value
    if selected_cluster == 'All':
        filtered_df = df
    else:
        filtered_df = df[df['Cluster'] == selected_cluster]

    model_dropdown.options = filtered_df['Filename'].tolist()

def display_dashboard(change):
    with dashboard_output:
        clear_output() # clear previous output
        selected_model = model_dropdown.value

        model_meta = df[df['Filename'] == selected_model]
        print("Metadata for the selected model:")
        display(model_meta.T) # Transpose for easier viewing.

        print("\nLaunching Interactive 3D Viewer...")

        model_path = os.path.join(model_folder, selected_model)
        display_model(model_path)

cluster_dropdown.observe(update_model_dropdown, names='value')
model_dropdown.observe(display_dashboard, names='value')

print("Interactive Dashboard for 3D Model Exploration")
display(cluster_dropdown, model_dropdown, dashboard_output)

Interactive Dashboard for 3D Model Exploration
Dropdown(description='Cluster:', options=['All', 0, 1, 2], value='All')
Dropdown(description='Model:', options=['The_Morning_Room.glb', 'angkor_wat_temple_cambodia.glb', 'terracotta_...
Output()
```

In [36]: output_csv = "My3DModels_Metadata_Summary.csv"
df.to_csv(output_csv, index=False)
print(f"Metadata summary exported to '{output_csv}'.")

```
html_report = """
<html>
<head>
<title>3D Model Metadata Summary Report</title></head>
<body>
<h1>3D Model Metadata Summary Report</h1>
```



```
<p>This report summarizes key metrics extracted from the 3D models in the 'My3DModels' folder.</p>
<h2>Summary Statistics</h2>
(df.describe(include='all')).to_html()
<h2>Clustering Analysis</h2>
<p>Based on the selected numerical features, the models were grouped into clusters. These clusters help identify groups of models with similar complexity or scanning characteristics. Such groupings can inform quality assessment and digital reconstruction strategies.</p>
</body>
</html>
"""
```

```
with open("Metadata_Summary_Report.html", "w") as f:
    f.write(html_report)
print("HTML summary report saved as 'Metadata_Summary_Report.html'.")

Metadata summary exported to 'My3DModels_Metadata_Summary.csv'.
HTML summary report saved as 'Metadata_Summary_Report.html'.
```

```
In [37]: if all(col in df.columns for col in ["Number of Meshes", "Number of Nodes", "Count_Extensions_Used"]):
df["Quality Score"] = (df["Number of Meshes"] + df["Number of Nodes"]) / (df["Count_Extensions_Used"] + 1)
# Normalize the quality score (optional)
df["Quality Score Normalized"] = (df["Quality Score"] - df["Quality Score"].min()) / (df["Quality Score"].max() - df["Quality Score"].min())

df_sorted = df.sort_values(by="Quality Score Normalized", ascending=False)
display(df_sorted[["Filename", "Number of Meshes", "Number of Nodes", "Count_Extensions_Used", "Quality Score", "Quality Score Normalized"]])
else:
    print("Required columns for quality score are missing.")
```

	Filename	Number of Meshes	Number of Nodes	Count_Extensions_Used	Quality Score	Quality Score Normalized
184	candi_prambanan_low_poly.glb	812	1741	0	2553.0	1.000000
131	Fire_Truck.glb	272	850	0	1122.0	0.439044
97	City_RTX.glb	203	618	0	821.0	0.321051
96	forbidden_city_model_.glb	232	478	0	710.0	0.277538
182	Model_of_the_Golden_Temple_Wan_fa_Gui_yi_Hall.glb	120	313	0	433.0	0.168953
...
30	notre_dame_cathedral_work_in_progress.glb	2	4	1	3.0	0.000392
13	chichen_itza_pyramid_3d_reconstruction.glb	1	4	1	2.5	0.000196
43	cliff_palace_mesa_verde_colorado.glb	1	3	1	2.0	0.000000
89	model_at_basilis_cathedral_red_square.glb	1	3	1	2.0	0.000000
35	lascaux.glb	1	3	1	2.0	0.000000

205 rows x 6 columns

```
In [38]: import ipywidgets as widgets
from IPython.display import display, clear_output

quality_slider = widgets.FloatSlider(
    value=0.5,
    min=0,
    max=1.0,
    step=0.01,
    description='Min Quality:',
    continuous_update=False,
)

quality_output = widgets.Output()

def filter_quality(change):
    with quality_output:
        clear_output()
        threshold = quality_slider.value
        filtered_df = df[df["Quality Score Normalized"] >= threshold]
        if filtered_df.empty:
            print("No models meet the current quality threshold.")
        else:
            display(filtered_df[["Filename", "Quality Score", "Quality Score Normalized"]])

quality_slider.observe(filter_quality, names='value')

print("Adjust the slider to filter models based on their normalized quality score:")
display(quality_slider)
display(quality_output)

filter_quality({'type': 'change', 'name': 'value', 'new': quality_slider.value})

Adjust the slider to filter models based on their normalized quality score:
FloatSlider(value=0.5, continuous_update=False, description='Min Quality:', max=1.0, step=0.01)
Output()
```

```
In [ ]: def prepare_model_for_reconstruction(filename, quality_threshold=0.7):

    try:
        model = trimesh.load(filename, force='scene', skip_materials=True)
    except Exception as e:
        print(f"Error loading model {filename}: {e}")
        return None

    if isinstance(model, trimesh.Scene):
        meshes = [geom for geom in model.geometry.values() if hasattr(geom, 'vertices') and len(geom.vertices) > 0]
        if not meshes:
            print(f"No valid meshes in {filename}")
            return None
        mesh = meshes[0]
    else:
        mesh = model

    vertex_count = len(mesh.vertices)

    print(f"{filename} has {vertex_count} vertices.")

    if vertex_count < 1000:
        print(f"{filename} does not meet the quality threshold for reconstruction.")
        return None

    processed_mesh = mesh
    output_filename = filename.replace(".glb", "_processed.obj")
    processed_mesh.export(output_filename)
    print(f"Processed model saved as {output_filename}")
    return output_filename
```

```
In [43]: import trimesh

def prepare_model_for_reconstruction(filename, quality_threshold=0.000):
    try:
        # Force load as a scene, skipping material info.
        model = trimesh.load(filename, force='scene', skip_materials=True)
        if model is None:
            print(f"Model {filename} could not be loaded.")
            return None

        # If the model is a scene, merge valid meshes.
        if isinstance(model, trimesh.Scene):
            meshes = [geom for geom in model.geometry.values() if hasattr(geom, 'vertices') and len(geom.vertices) > 0]
            if not meshes:
                print(f"No valid meshes in {filename}")
                return None
            merged = trimesh.util.concatenate(meshes) if len(meshes) > 1 else meshes[0]
        else:
            merged = model

        vertex_count = len(merged.vertices)
        print(f"{filename} has {vertex_count} vertices.")
        if vertex_count < quality_threshold:
            print(f"Model {filename} does not meet the quality threshold ({quality_threshold} vertices).")
            return None

        output_filename = filename.replace(".glb", "_processed.obj")
        merged.export(output_filename)
        print(f"Processed model saved as {output_filename}")
        return output_filename
    except Exception as e:
        print(f"Error processing {filename}: {e}")
        return None
```

```
In [45]: def extend_metadata_with_vertex_count(file_list):
import trimesh
import pandas as pd
metadata = []
for file in file_list:
    try:
        model = trimesh.load(file, force='scene', skip_materials=True)
        if isinstance(model, trimesh.Scene):
            meshes = [geom for geom in model.geometry.values() if hasattr(geom, 'vertices') and len(geom.vertices) > 0]
            if meshes:
                merged = trimesh.util.concatenate(meshes) if len(meshes) > 1 else meshes[0]
                vertex_count = len(merged.vertices)
            else:
                vertex_count = 0
        else:
            vertex_count = len(model.vertices)
        metadata.append({"Filename": file, "Vertex Count": vertex_count})
    except Exception as e:
        print(f"Error processing {file}: {e}")
        metadata.append({"Filename": file, "Vertex Count": None})
return pd.DataFrame(metadata)

# Example usage:
vertex_df = extend_metadata_with_vertex_count(glb_files)
print(vertex_df.head())
# Optionally merge with your existing DataFrame 'df'
df = df.merge(vertex_df, on="Filename", how="left")
df.head()
```

Error processing My3DModelsEDA/lincoln_memorial.glb: 'list' object has no attribute 'vertices'

Error processing My3DModelsEDA/casa_dello_scheletro_at_herculaneum.glb: 'list' object has no attribute 'vertices'

	My3DModelsEDA/The_Morning_Room.glb	2999994.0
0	My3DModelsEDA/angkor_wat_temple_cambodia.glb	1452600.0
1	My3DModelsEDA/terracotta_warrior.glb	442892.0
2	My3DModelsEDA/parthenon_-_acropolis_athens_gre...	484483.0
3	My3DModelsEDA/Cathedral.glb	3041195.0

	Filename	Asset Version	Asset Generator	Number of Scenes	Number of Nodes	Number of Meshes	Number of Materials	Number of Animations	Number of Textures	Number of Images	...	Extensions Used	Extensions Required	Count_Extensions_Used	Cluster	PCA1	PCA2	Quality Score	Quality Score Normalized	Vertex Count_x	Vertex Count_y
0	The_Morning_Room.glb	2.0	pygltflib@v1.16.3	1	30	13	1	0	1	1 ...				0	0	-0.509978	0.112773	43.0	0.016072	NaN	NaN
1	angkor_wat_temple_cambodia.glb	2.0	Sketchfab-12.68.0	1	27	23	1	0	1	1 ...				0	0	-0.465713	0.199260	50.0	0.018816	NaN	NaN
2	terracotta_warrior.glb	2.0	Sketchfab-13.93.0	1	11	7	1	0	3	3 ...				0	0	-0.422523	-0.126662	18.0	0.006272	NaN	NaN
3	parthenon_-_acropolis_athens_greece.glb	2.0	Sketchfab-12.68.0	1	13	8	2	0	2	2 ...				0	0	-0.451730	-0.079912	21.0	0.007448	NaN	NaN
4	Cathedral.glb	2.0	pygltflib@v1.16.3	1	70	33	2	0	4	4 ...				0	0	-0.003810	0.288469	103.0	0.039592	NaN	NaN

5 rows x 21 columns

```
In [46]: html_report = """
<html>
<head>
<title>Final 3D Model Metadata Report</title>
<style>
body {{ font-family: Arial, sans-serif; margin: 40px; }}
h1 {{ color: #2c3e50; }}
table {{ border-collapse: collapse; width: 100%; }}
th, td {{ border: 1px solid #ddd; padding: 8px; }}
th {{ background-color: #f2f2f2; text-align: left; }}
</style>
</head>
<body>
<h1>Final 3D Model Metadata Report</h1>
<p>This report summarizes key metrics and quality assessments for the 3D models in the My3DModels folder.</p>
<h2>Summary Statistics</h2>
(df.describe(include='all')).to_html()
<h2>Detailed Model Data</h2>
(df.to_html(index=False))
</body>
</html>
"""

with open("Final_Metadata_Report.html", "w") as f:
    f.write(html_report)
print("Final HTML report saved as 'Final_Metadata_Report.html'.")

Final HTML report saved as 'Final_Metadata_Report.html'.
```