Exploratory Analysis of HAM10000

July 12, 2025

1 Exploratory Analysis of HAM10000

1.1 Setup

[17]:
```
import pandas as pd
import matplotlib.pyplot as plt
from pathlib import Path
```

[18]:
```
# basic plot style setup
plt.style.use("default")
plt.rcParams.update({"figure.figsize": (8, 5), "axes.titlesize": 14, "axes.
 ↪labelsize": 12})
```

[19]:
```
# paths relative to exploratory_analysis/
root_dir = Path("..")
meta_csv = root_dir / "HAM10000_metadata.csv"
images_dir_1 = root_dir / "HAM10000_images_part_1"
images_dir_2 = root_dir / "HAM10000_images_part_2"
```

[40]:
```
# load metadata
meta = pd.read_csv(meta_csv)
print("dataframe shape:", meta.shape)
meta.head()
```

```
dataframe shape: (10015, 7)
```

[40]:

	lesion_id	image_id	dx	dx_type	age	sex	localization
0	HAM_0000118	ISIC_0027419	bkl	histo	80.0	male	scalp
1	HAM_0000118	ISIC_0025030	bkl	histo	80.0	male	scalp
2	HAM_0002730	ISIC_0026769	bkl	histo	80.0	male	scalp
3	HAM_0002730	ISIC_0025661	bkl	histo	80.0	male	scalp
4	HAM_0001466	ISIC_0031633	bkl	histo	75.0	male	ear

[41]:
```
# show basic info
print("\n===== info =====")
meta.info()
print("\n===== describe (numerical only) =====")
print(meta.describe(include=["number"]))
```

```
===== info =====
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10015 entries, 0 to 10014
Data columns (total 7 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   lesion_id     10015 non-null  object
 1   image_id      10015 non-null  object
 2   dx            10015 non-null  object
 3   dx_type       10015 non-null  object
 4   age           9958 non-null   float64
 5   sex           10015 non-null  object
 6   localization  10015 non-null  object
dtypes: float64(1), object(6)
memory usage: 547.8+ KB

===== describe (numerical only) =====
                age
count  9958.000000
mean     51.863828
std      16.968614
min       0.000000
25%      40.000000
50%      50.000000
75%      65.000000
max      85.000000
```

1.2 Distributions

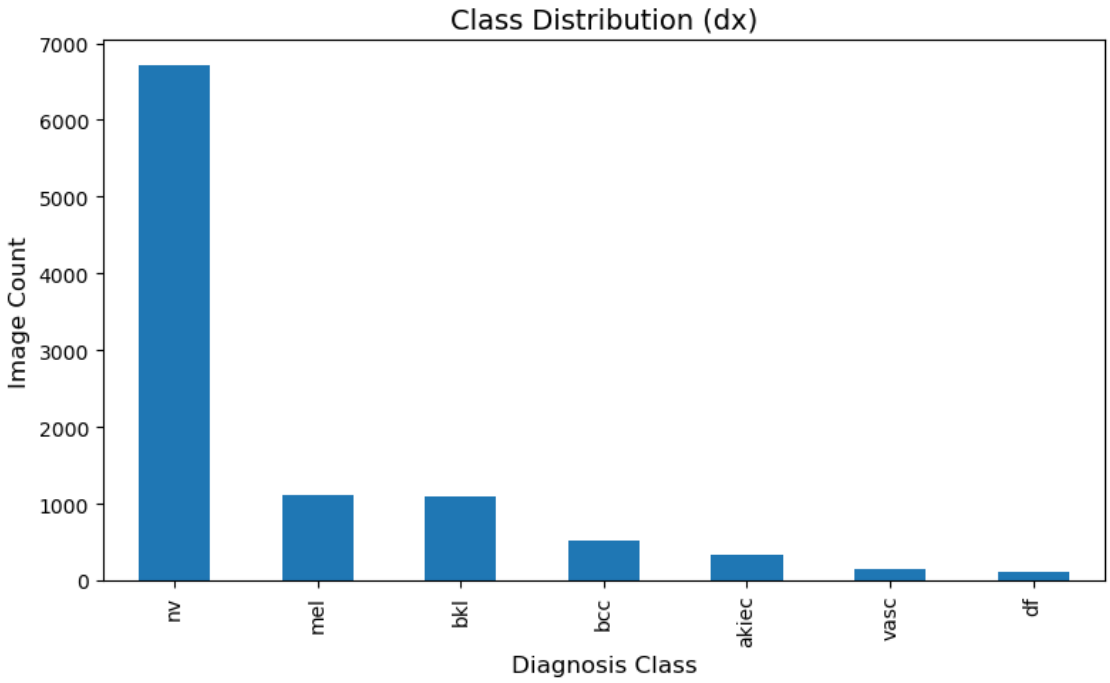
1.2.1 Class Distribution (dx)

[23]:
```
class_counts = meta["dx"].value_counts().sort_values(ascending=False)
print(class_counts)

class_counts.plot(kind="bar")
plt.title("Class Distribution (dx)")
plt.xlabel("Diagnosis Class")
plt.ylabel("Image Count")
plt.tight_layout()
plt.show()
```

```
dx
nv       6705
mel      1113
bkl      1099
bcc       514
akiec     327
```

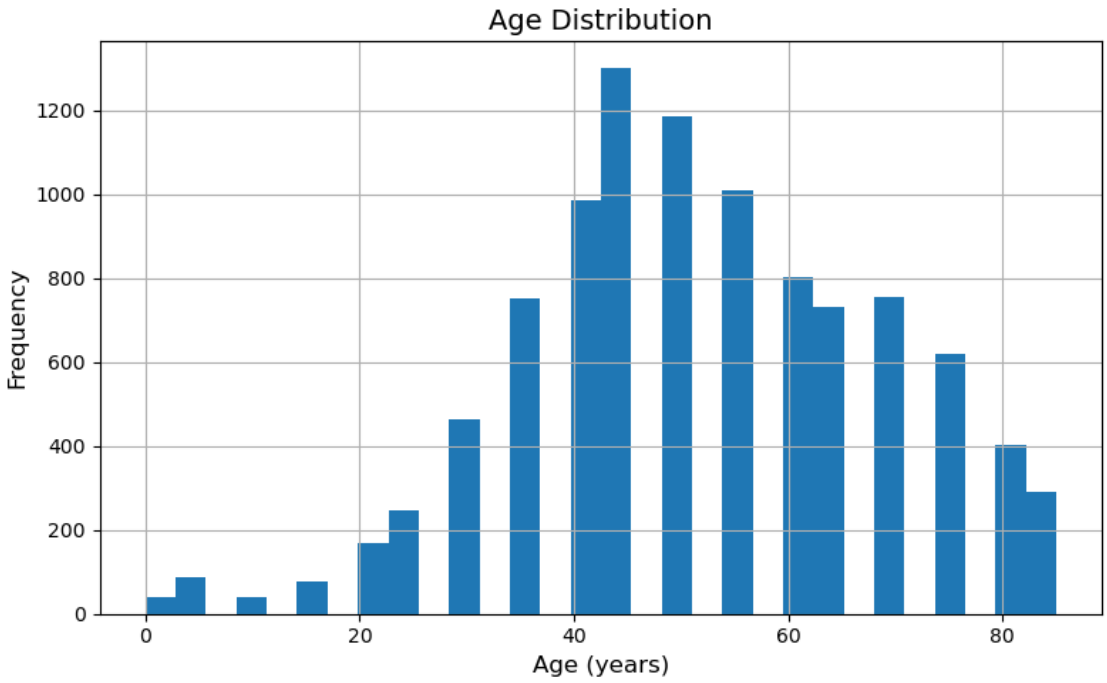
```
vasc      142
df        115
Name: count, dtype: int64
```

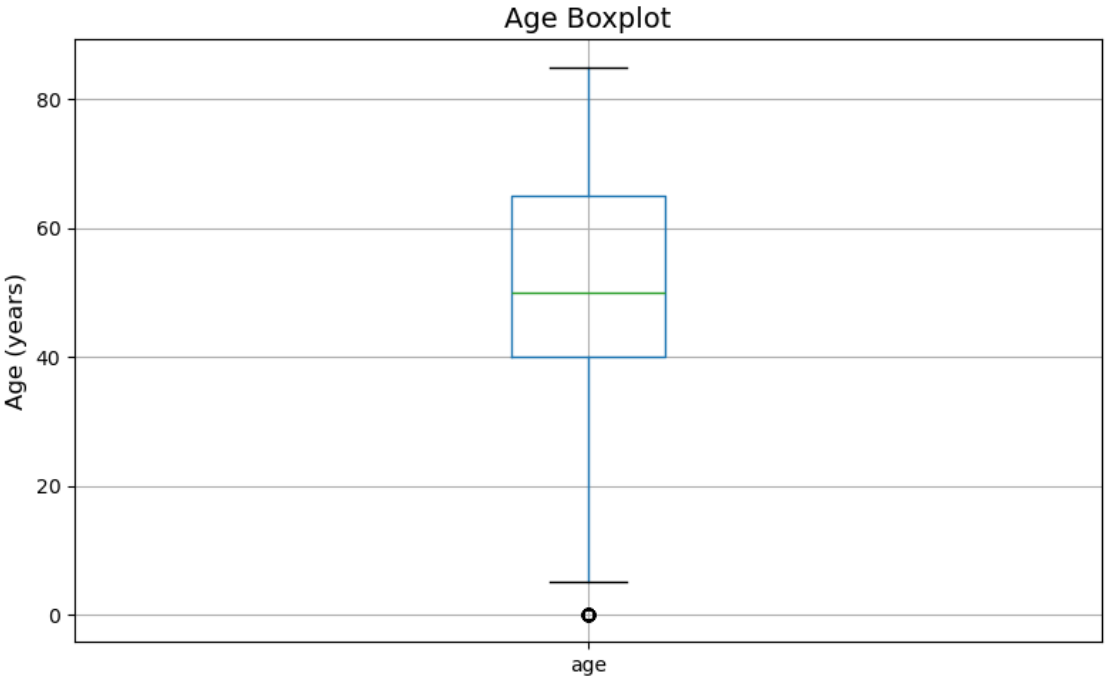


1.2.2 Age Distribution (age)

[22]:
```
meta["age"].hist(bins=30)
plt.title("Age Distribution")
plt.xlabel("Age (years)")
plt.ylabel("Frequency")
plt.tight_layout()
plt.show()

meta.boxplot(column="age")
plt.title("Age Boxplot")
plt.ylabel("Age (years)")
plt.tight_layout()
plt.show()
```


Age Distribution
Frequency
Age (years)
0
200
400
600
800
1000
1200
0
20
40
60
80

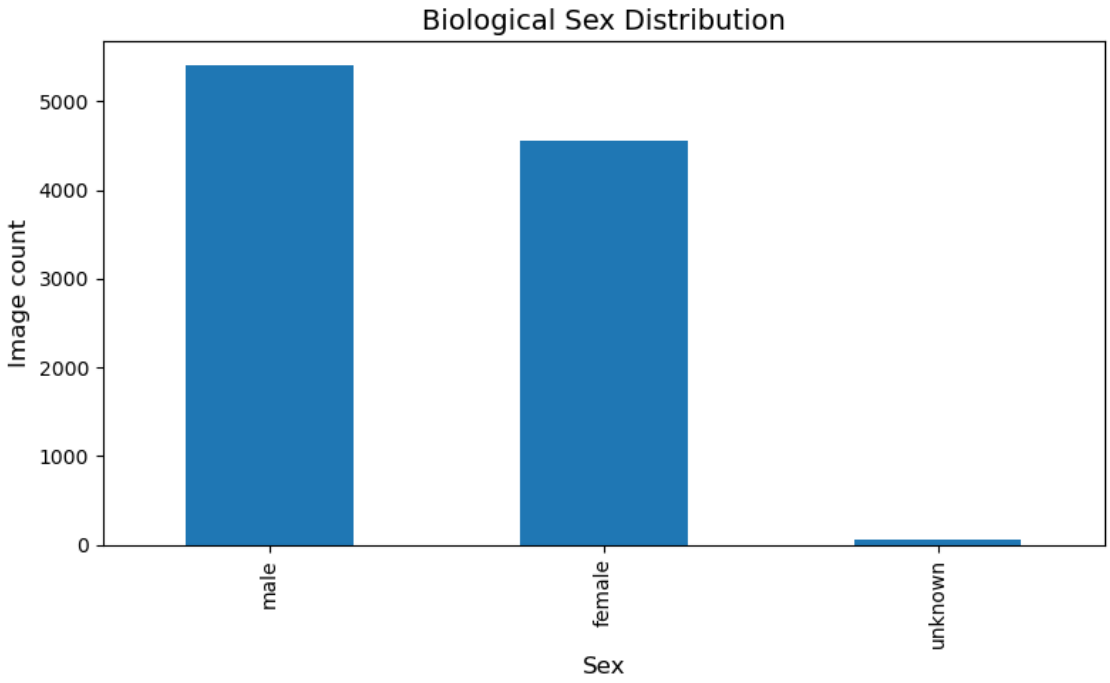

Age Boxplot
Age (years)
0
20
40
60
80
age

1.2.3 Biological Sex Distribution (sex)

[24]:
```
sex_counts = meta["sex"].value_counts()
print(sex_counts)

sex_counts.plot(kind="bar")
plt.title("Biological Sex Distribution")
plt.xlabel("Sex")
plt.ylabel("Image count")
plt.tight_layout()
plt.show()
```

```
sex
male       5406
female     4552
unknown      57
Name: count, dtype: int64
```



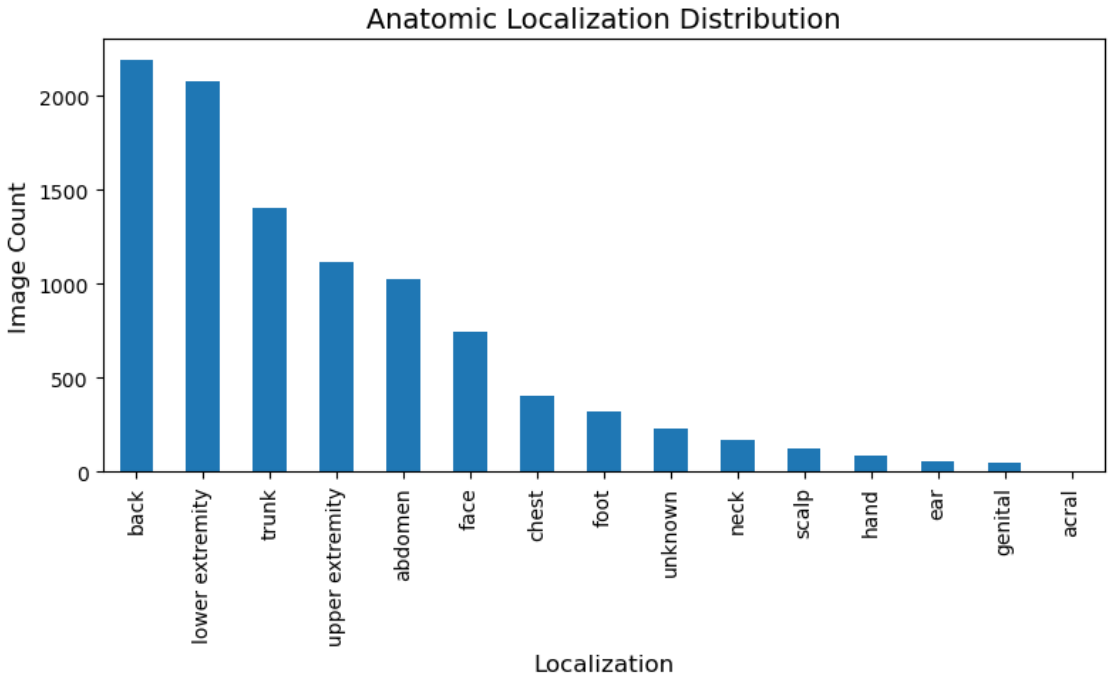
1.2.4 Anatomic Localization Distribution (localization)

[26]:
```
loc_counts = meta["localization"].value_counts().sort_values(ascending=False)
print(loc_counts.head(10))

loc_counts.plot(kind="bar")
plt.title("Anatomic Localization Distribution")
```

```
plt.xlabel("Localization")
plt.ylabel("Image Count")
plt.tight_layout()
plt.show()
```

```
localization
back                2192
lower extremity     2077
trunk               1404
upper extremity     1118
abdomen             1022
face                 745
chest                407
foot                 319
unknown              234
neck                 168
Name: count, dtype: int64
```



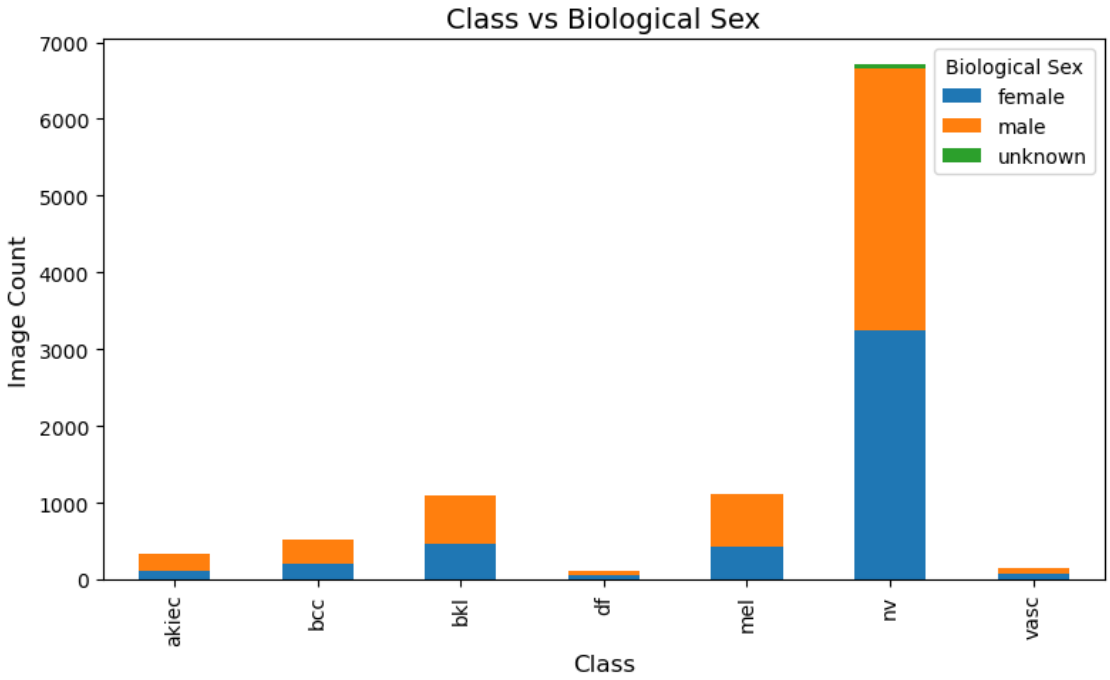
1.3 Categorical Variable Relationship

1.3.1 Class x Biological Sex

[29]:
```
ct_dx_sex = pd.crosstab(meta["dx"], meta["sex"])
print(ct_dx_sex)
```

```
ct_dx_sex.plot(kind="bar", stacked=True)
plt.title("Class vs Biological Sex")
plt.xlabel("Class")
plt.ylabel("Image Count")
plt.legend(title="Biological Sex")
plt.tight_layout()
plt.show()
```

```
sex     female  male  unknown
dx
akiec      106   221        0
bcc        197   317        0
bkl        463   626       10
df          52    63        0
mel        424   689        0
nv        3237  3421       47
vasc        73    69        0
```

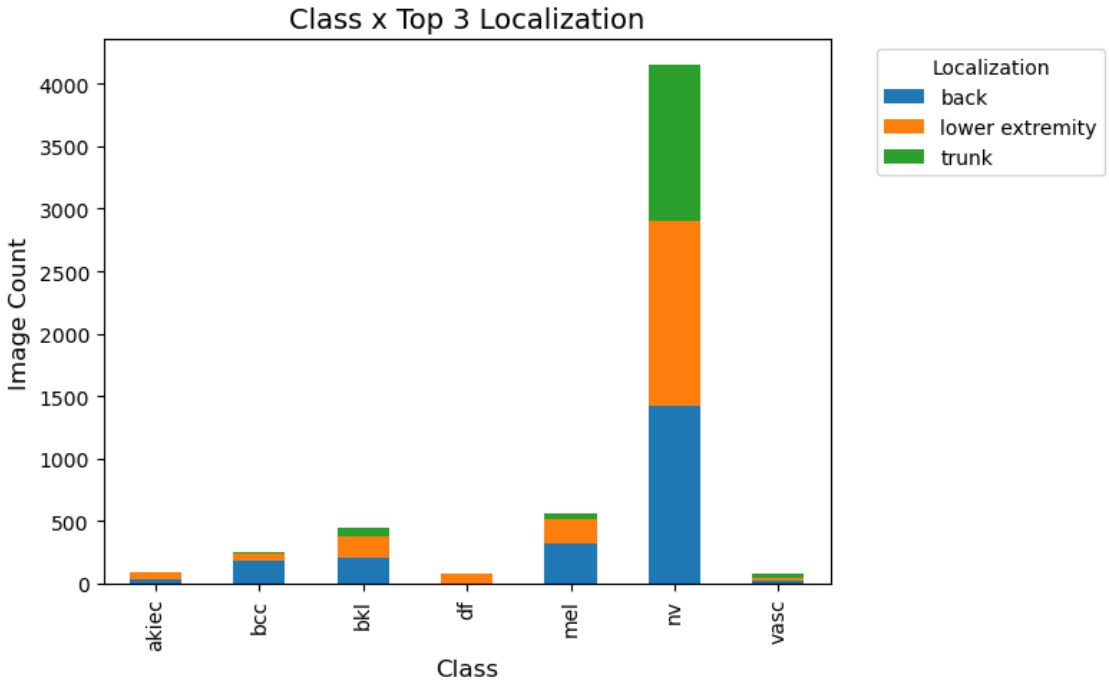


1.3.2 Class x Top 3 Localization

[37]:
```
top_n = 3
top_localizations = loc_counts.head(top_n).index
subset = meta[meta["localization"].isin(top_localizations)]
ct_dx_loc = pd.crosstab(subset["dx"], subset["localization"])
print(ct_dx_loc)
```

```
ct_dx_loc.plot(kind="bar", stacked=True)
plt.title(f"Class x Top {top_n} Localization")
plt.xlabel("Class")
plt.ylabel("Image Count")
plt.legend(title="Localization", bbox_to_anchor=(1.05, 1), loc="upper left")
plt.tight_layout()
plt.show()
```

localization	back	lower extremity	trunk
dx			
akiec	29	65	1
bcc	186	58	11
bkl	202	174	73
df	2	82	0
mel	324	192	47
nv	1427	1479	1241
vasc	22	27	31



1.4 Missing Values

[39]:
```
missing = meta.isnull().sum()
print(missing[missing > 0])

missing.plot(kind="bar")
```

```
plt.title("Missing Values per Column")
plt.xlabel("Column")
plt.ylabel("NA Count")
plt.tight_layout()
plt.show()
```

```
age     57
dtype: int64
```