MDM-2025 Homework 2

Mert Dagli (100830277)
Tuomas Mäkinen (103382573)
Maja Sellmer (103396682)

October 13, 2025

1 Methods

The task was to explore clustering bird species by their physical characteristics using agglomerative hierarchical clustering. To this end, we extracted features from the given dataset, clustered the data in different ways, and evaluated the results by comparing them to the real biological groupings. We completed the assignment in Python, using the following libraries:

- `matlplotlib` for plotting
- `pandas` to handle the given data as a dataframe
- `math` for calculations
- `scipy` for distances and clustering
- `sklearn.metrics` for NMI score

1.1 Feature extraction

The data had to be preprocessed first to extract useful features. Some of the features were given as numerical ranges. We used the midpoints of these ranges to be able to use them for calculation. Since the features are in different units and scales, to avoid one of them dominating the distance we need to scale. We tried multiple scaling and normalization methods separately such as min-max scaling, mean normalization and z-score normalization, with z-score ultimately giving the best results.

We created two new features: Body Mass Index (BMI) and Wing Span Index (WSI). BMI combines body length and mass to represent overall body composition, and WSI measures the ratio between wingspan and body length.

1.2 Combined distance measure

For the colours of the back and belly, we define the distance as 0 if the colour is identical, 0.8 if the main hue is the same (for example dark grey vs light grey) and 1 if it is completely different.

For the numerical features, we use Euclidean distance.

The combined distance is given by

$$dist_{Combined} = \lambda * dist_{Numerical} + (1 - \lambda) * dist_{Categorical}$$

A first approach as discussed in the lecture is to choose λ as the proportion of numerical values, which in this case is 7/10. Indeed, this seems to be a good choice since the categorical distance is less discriminating - many birds of different families share the same colours.

1.3 Feature selection

Since the length, weight and wingspan are encoded in the BMI and WSI, these features can be dropped. Indeed, we found that this leads to better results in the clustering since these features have high variance, and there can be a lot of variance even within one group - for example, naurulokki and isolokki are closely related, while one weighs only 200-350 grams and the other 1000-1800 grams. The ratios BMI and WSI seem to be a better indicator of relatedness between species than the raw values.

1.4 Clustering

First we experimented with the Ward linkage, but this only works for numerical distance. For the combined distance, we tested three different linkage metrics (single, complete and average) and numbers of clusters ranging from 5 to 24. The clustering was done using methods from `scipy.cluster.hierarchy`.

Table 1 shows the resulting Normalized Mutual Information score, using the combined distance after dropping some features.

Linkage	**K=5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Single	0.25	0.27	0.29	0.33	0.43	0.45	0.47	0.46	0.53	0.54	0.56
Complete	0.52	0.54	0.58	0.61	0.66	0.70	0.72	0.73	0.73	0.73	0.75
Average	0.48	0.58	0.59	0.62	0.66	0.68	0.74	0.73	0.73	0.73	0.73

Table 1: NMI values for different linkage methods across K=5–15 clusters

1.5 Evaluation

We can see that the NMI increases with higher K, but there is a tradeoff to be found since we don't want many singleton clusters. For lower numbers of clusters, the average linkage metric performs the best. It reaches a local maximum at $k = 11$ with a NMI score of 0.735, so this seems to be a good clustering.

Cluster	Species	Group	Length	Wspan	Weight	AR	Wload	Back	Belly	Ftype	BMI	WSI
1	nokikana	rallidae	37.0	75.0	750	5.71	1.16	greyish black	greyish black	C	0.55	2.03
2	merikotka	accipitridae	89.0	215.0	5500	5.18	0.68	brown	brown	A	0.69	2.42
3	punasotka	diving ducks	45.5	71.0	870	7.27	1.46	beige	beige	C	0.42	1.56
3	tukkasotka	diving ducks	43.5	68.5	700	7.12	1.52	dark brown	dark brown	C	0.37	1.57
3	härkälintu	podicipedidae	45.0	81.0	800	6.48	1.81	blackish grey	reddish brown	C	0.40	1.80
3	mustakurkku-uikku	podicipedidae	34.5	50.5	560	7.60	1.56	black	reddish brown	C	0.47	1.46
3	silkkiuikku	podicipedidae	52.5	66.0	1000	7.87	1.34	blackish grey	white	C	0.36	1.26
4	kyhmyjoutsen	anserinae	152.5	220.0	11000	7.69	2.15	white	white	C	0.47	1.44
4	kaakkuri	gaviidae	61.0	100.5	1650	9.26	1.64	greyish brown	white	C	0.44	1.65
4	kuikka	gaviidae	65.5	111.0	2500	8.94	2.34	black-white	white	C	0.58	1.69
5	ruskosuohaukka	accipitridae	49.0	127.5	615	6.37	0.34	dark brown	brown	A	0.26	2.60
5	haarahaukka	accipitridae	53.0	142.5	750	7.14	0.33	dark brown	dark brown	A	0.27	2.69
5	keräkurmitsa	charadriidae	22.0	60.5	113	7.43	0.45	dappled brown	black	C	0.23	2.75
6	naurulokki	laridae	36.0	92.5	275	8.13	0.31	light grey	white	B	0.21	2.57
6	isolokki	laridae	65.5	148.0	1400	8.24	0.66	bluish grey	white	B	0.33	2.26
6	harmaalokki	laridae	60.0	135.5	1050	8.24	0.64	bluish grey	white	B	0.29	2.26
7	lapintiira	sternidae	35.0	71.5	110	8.97	0.20	grey	white	B	0.09	2.04
7	kalatiira	sternidae	39.0	75.0	123	9.14	0.24	grey	white	B	0.08	1.92
8	liro	scolopacidae	22.0	37.0	60	7.53	0.23	dappled brown	white	C	0.12	1.68
8	metsäviklo	scolopacidae	22.5	41.5	80	7.20	0.29	brown	white	C	0.16	1.84
8	suosirri	scolopacidae	19.0	34.0	49	6.95	0.31	dappled brown	black-white	C	0.13	1.79
8	pikkusirri	scolopacidae	15.0	28.5	25	6.63	0.18	dappled brown	white	C	0.11	1.90
8	karikukko	scolopacidae	22.5	46.0	110	7.21	0.45	black-brown	white	C	0.22	2.04
8	suokukko	scolopacidae	25.5	47.5	110	6.73	0.36	dappled brown	white	C	0.17	1.86
8	tylli	charadriidae	18.5	38.0	58	7.43	0.32	greyish brown	white	C	0.17	2.05
8	pikkutylli	charadriidae	15.5	33.5	40	6.93	0.24	greyish brown	white	C	0.17	2.16
9	töyhtöhyyppä	charadriidae	32.0	69.5	200	5.45	0.34	black	white	C	0.20	2.17
9	varpushaukka	accipitridae	35.0	69.0	238	5.35	0.24	greyish brown	black-white	A	0.19	1.97
9	hiirihaukka	accipitridae	52.0	121.0	775	5.58	0.40	dark brown	brown-white	A	0.29	2.33
9	kanahaukka	accipitridae	56.5	110.0	1225	5.38	0.66	grey	black-white	A	0.38	1.95
9	sinisuohaukka	accipitridae	48.5	107.5	458	5.81	0.21	brown	brown-white	A	0.19	2.22
9	mehiläishaukka	accipitridae	55.5	121.0	835	5.48	0.38	greyish brown	brown-white	A	0.27	2.18
10	liejukana	rallidae	34.0	52.5	330	5.33	0.74	brownish black	dark grey	C	0.29	1.54
10	luhtakana	rallidae	26.0	41.5	133	4.50	0.45	dappled brown	grey	C	0.20	1.60
10	ruisrääkkä	rallidae	28.5	47.5	170	5.18	0.48	dappled brown	beige	C	0.21	1.67
10	taivaanvuohi	scolopacidae	26.0	42.0	100	5.91	0.40	dappled brown	white	C	0.15	1.62
10	lehtokurppa	scolopacidae	35.0	60.0	305	5.68	0.54	dappled brown	dappled beige	C	0.25	1.71
10	kaulushaikara	ardeidae	75.0	115.0	1350	5.58	0.60	dappled brown	dappled brown	B	0.24	1.53
10	harmaahaikara	ardeidae	95.0	165.0	1040	5.88	0.40	light grey	greyish white	B	0.12	1.74
10	luhtahuitti	rallidae	21.0	39.5	100	5.44	0.38	dappled brown	dappled brown	C	0.23	1.88
11	merihanhi	anserinae	82.5	158.5	3000	6.57	1.09	dappled brown	light grey	C	0.44	1.92
11	kurki	gruifores	115.0	202.5	5000	6.30	1.00	grey	grey	C	0.38	1.76
11	lapasorsa	dabbling ducks	48.0	77.5	600	7.47	1.06	dappled brown	dark brown	C	0.26	1.61
11	haapana	dabbling ducks	46.0	75.5	700	7.70	1.09	dappled brown	white	C	0.33	1.64
11	sinisorsa	dabbling ducks	55.0	83.0	1100	6.19	1.05	dappled brown	beige	C	0.36	1.51
11	tavi	dabbling ducks	36.0	56.0	350	6.51	0.75	dappled brown	beige	C	0.27	1.56
11	heinätavi	dabbling ducks	39.0	63.0	425	5.80	0.76	dappled brown	beige	C	0.28	1.62
11	merimetso	phalacrocoracidae	95.0	135.0	2900	6.54	1.02	black	black	C	0.32	1.42
11	kattohaikara	ardeidae	107.5	197.5	3250	6.41	0.86	black-white	white	A	0.28	1.84
11	pikku-uikku	podicipedidae	26.0	42.5	175	6.37	0.92	black	beige	C	0.26	1.63

Table 2: Tabular results of the most informative clustering

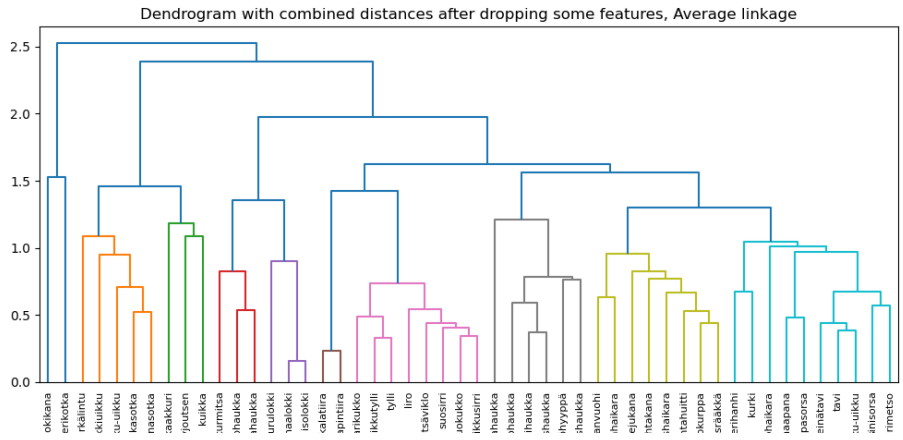

Figure 1: Dendrogram for most informative clustering

2 Results

Figure 1 shows the dendrogram for this clustering and Table 2 the data.

- The first cluster is a singleton, containing the species nokikana of the group rallidae.
- The second is also a singleton, containing the species merikotka of the group accipitridae.
- The third cluster contains birds with relatively high BMI and wing load. These fall into the families of the diving ducks and podicipedidae.
- The fourth cluster, consisting of anserinae and gaviidae, have even higher BMI and wing load, being among the heaviest of all birds in the dataset.
- The fifth cluster is likely grouped together because they have very similar values for BMI and WSI, even though two belong to the accipitridae family and one to charadriidae. This cluster is also one of the few not homogenous with respect to flying type, so one could consider this a mistake.
- The seagulls of the family laridae in the sixth cluster fit together since they have very high aspect ratio and similar colours.
- Likewise, the two sternidae in the seventh cluster have very similar values across all categories, and we can see in the dendrogram that they are very close to each other.
- Cluster 8 consists of some scolopacidae and some charadriidae, which are actually both subfamilies of the Charadrii, so it is understandable that they are grouped together. They are quite small with low BMI, and all have flying type C.
- In the ninth cluster, we find the accipitridae, which are medium-sized but with rather long wings.
- The tenth cluster is quite large with birds from three different groups: rallidae, scolopacidae and ardeidae. What they have in common is low aspect ratio and short wings.
- The eleventh cluster is the biggest of the bunch, containing all the dabbling ducks, but also the gruifores and phalacrocoracidae (of which there was only one species each in the dataset), as well as three other species which were likely misclassified since their BMI and WSI is similar to the dabbling ducks.

3 Conclusions

To summarize, we calculated distances between the bird species by introducing two new numerical columns (BMI and WSI), using appropriate scaling for the numerical features, defining a suitable distance measure for the categorical features, and then combining these two distances. We experimented with different linkage metrics and different subsets of features for the clustering until we found one with a good NMI score, but not too many singleton clusters. Finally, we analysed the results of this clustering by inspecting the clusters and identifying which characteristics the species in them have in common.

4 Appendix – Code

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import math
from scipy.spatial.distance import cdist, pdist, squareform
from scipy.cluster.hierarchy import linkage, dendrogram, fcluster
from sklearn.metrics import normalized_mutual_info_score

df = pd.read_csv("birds2025.csv", sep=';')

def midpoint(value):
        if isinstance(value, str) and '-' in value:
            try:
                a, b = map(float, value.split('-'))
                return (a + b) / 2
            except ValueError:
                return value
        try:
            return float(value)
        except ValueError:
            return value

def midpoint_column(df, features=['length', 'wspan', 'weight']):
    df_mid = df.copy()
    for feature in features:
        df_mid[feature] = df_mid[feature].apply(midpoint)
    return df_mid

def add_bmi(df):
    """
    Adds a new column 'bmi' = weight / length^2
    """
    new_df = df.copy()
    new_df['bmi'] = new_df["weight"] / (new_df["length"] ** 2)
    return new_df

def add_wsi(df):
    """
    Adds a new column 'wsi' = wspan / length
    """
    new_df = df.copy()
```

```
    new_df['wsi'] = new_df["wspan"] / new_df["length"]
    return new_df

df = midpoint_column(df)
df = add_bmi(df)
df = add_wsi(df)

features_list = ['length', 'wspan', 'weight', 'AR', 'wload', 'bmi', 'wsi']
for f in features_list:
    print(f, df[f].min(), df[f].max(), df[f].mean(), df[f].std())

features_list = ['length', 'wspan', 'weight', 'AR', 'wload', 'bmi', 'wsi']
def minmax_scaling(df, features=features_list):
    minmax_df = df.copy()
    for feature in features:
        column = minmax_df[feature]
        min_val = column.min()
        max_val = column.max()
        minmax_df[feature] = (column - min_val) / (max_val - min_val)
    return minmax_df

def mean_scaling(df, features=features_list):
    scaled_df = df.copy()
    for feature in features:
        column = scaled_df[feature]
        min_val = column.min()
        max_val = column.max()
        mean_val = column.mean()
        scaled_df[feature] = (column - mean_val) / (max_val - min_val)
    return scaled_df

def zscore_scaling(df, features=features_list):
    scaled_df = df.copy()
    for feature in features:
        column = scaled_df[feature]
        scaled_df[feature] = (column - column.mean()) / column.std()
    return scaled_df

df_minmax = minmax_scaling(df)
df_meanscaled = mean_scaling(df)
df_zscore = zscore_scaling(df)

def pwnum_dists(df, features):
    X = df[features].to_numpy()
    dist_vector = pdist(X, metric='euclidean')
    return squareform(dist_vector)

dists_meanscaled = pwnum_dists(df_meanscaled, features_list)
dists_minmax = pwnum_dists(df_minmax, features_list)
dists_zscore = pwnum_dists(df_zscore, features_list)
dists_dropped = pwnum_dists(df_zscore, ['AR', 'wload', 'bmi', 'wsi'])
# dists_minmax and dists_meanscaled should be equal since it is only a translation
print(dists_meanscaled)
print(dists_minmax)
print(dists_zscore)
print(dists_dropped)

Z_num1 = linkage(squareform(dists_minmax), method='ward')
Z_num2 = linkage(squareform(dists_meanscaled), method='ward')
Z_num3 = linkage(squareform(dists_zscore), method='ward')
Z_num4 = linkage(squareform(dists_dropped), method='ward')
```

```
labels = df['species'].values

plt.figure(figsize=(12, 5))
plt.title("Dendrogram with numerical distances after Min-max scaling, Ward linkage")
dendrogram(Z_num1, labels=labels, leaf_rotation=90, leaf_font_size=8, color_threshold=0.9)
plt.show()

plt.figure(figsize=(12, 5))
plt.title("Dendrogram with numerical distances after Z-score normalization, Ward linkage")
dendrogram(Z_num3, labels=labels, leaf_rotation=90, leaf_font_size=8, color_threshold=2.1)
plt.show()

plt.figure(figsize=(12, 5))
plt.title("Dendrogram with numerical distances after dropping some features, Ward linkage")
dendrogram(Z_num4, labels=labels, leaf_rotation=90, leaf_font_size=8, color_threshold=1.5)
plt.show()

def main_colour(c):
    try:
        pre , main  = c.split(' ')
    except ValueError:
        main = c
    return main

def col_distance(c1, c2):
    if c1 == c2:
        return 0.0
    m1, m2 = main_colour(c1), main_colour(c2)
    if m1 == m2:
        return 0.8  #0.9 - 0.7 seems to have best results
    return 1

def ftype_distance(f1, f2):
    if f1 == f2:
        return 0.0
    return 1

def cat_dists(df, feature, func):
    n = len(df)
    mat = np.zeros((n, n))
    for i in range(n):
        for j in range(i+1, n):
            d = func(df.iloc[i][feature], df.iloc[j][feature])
            mat[i, j] = mat[j, i] = d
    return mat

back_dists = cat_dists(df, 'back', col_distance)
belly_dists = cat_dists(df, 'belly', col_distance)
ftype_dists = cat_dists(df, 'ftype', ftype_distance)
dists_cat = (back_dists+ belly_dists+ftype_dists)/3
print(dists_cat)
combined_dists = dists_zscore * 0.7 + dists_cat * 0.3
combined_dists_dropped = dists_dropped * 0.7 + dists_cat * 0.3
print(combined_dists)
print(combined_dists_dropped)

print("Normalized Mutual Information between clustering and biological groups \n")
for k in range(5, 25):
    Z_single = linkage(squareform(combined_dists), method="single")
    clusters_single = fcluster(Z_single, k, criterion='maxclust')
    score_single = normalized_mutual_info_score(df['group'], clusters_single)
    Z_complete = linkage(squareform(combined_dists), method="complete")
```

```
    clusters_complete = fcluster(Z_complete, k, criterion='maxclust')
    score_complete = normalized_mutual_info_score(df['group'], clusters_complete)
    Z_average = linkage(squareform(combined_dists), method="average")
    clusters_average = fcluster(Z_average, k, criterion='maxclust')
    score_average = normalized_mutual_info_score(df['group'], clusters_average)
    print(f"K={k}: NMI_single={score_single:.3f}, NMI_complete={score_complete:.3f}, NMI_average={sc

print("Normalized Mutual Information between clustering and biological groups (after dropping some f
for k in range(5, 25):
    Z_single = linkage(squareform(combined_dists_dropped), method="single")
    clusters_single = fcluster(Z_single, k, criterion='maxclust')
    score_single = normalized_mutual_info_score(df['group'], clusters_single)
    Z_complete = linkage(squareform(combined_dists_dropped), method="complete")
    clusters_complete = fcluster(Z_complete, k, criterion='maxclust')
    score_complete = normalized_mutual_info_score(df['group'], clusters_complete)
    Z_average = linkage(squareform(combined_dists_dropped), method="average")
    clusters_average = fcluster(Z_average, k, criterion='maxclust')
    score_average = normalized_mutual_info_score(df['group'], clusters_average)
    print(f"K={k}: NMI_single={score_single:.3f}, NMI_complete={score_complete:.3f}, NMI_average={sc

Z_comb1 = linkage(squareform(combined_dists), method='average')
Z_comb2 = linkage(squareform(combined_dists_dropped), method='average')

k = 11

# plt.figure(figsize=(12, 5))
# plt.title("Dendrogram with combined distances, Average linkage")
# ct = Z_comb1[-(k-1),2]
# dendrogram(Z_comb1, labels=labels, leaf_rotation=90, leaf_font_size=8, color_threshold=ct)
# plt.show()

plt.figure(figsize=(12, 5))
plt.title("Dendrogram with combined distances after dropping some features, Average linkage")
ct = Z_comb2[-(k-1),2]
dendrogram(Z_comb2, labels=labels, leaf_rotation=90, leaf_font_size=8, color_threshold=ct)
plt.savefig("dendrogram_final.png")
plt.show()

clusters = fcluster(Z_comb2, k, criterion='maxclust')
df_results = df.copy()
df_results["Cluster"] = clusters
df_results.sort_values("Cluster")
```

5 Bibliography

References

[1] Charu C. Aggarwal. *Data Mining - The Textbook*. Springer, 2015.