Contents

Progetto 6: I mammiferi depongono uova?

Il dataset Zoo fornisce una serie di dati relativi a diverse specie animali al fine di classificarle in 7 diverse classi, ovvero:

0. mammiferi
1. uccelli
2. rettili
3. pesci
4. anfibi
5. insetti
6. invertebrati

Seguendo un approccio non supervisionato, ovvero senza osservare la classe di ogni specie animale, il progetto mira a confrontare le diverse specie e raggrupparle utilizzando diversi algoritmi di clustering. Confrontando poi il risultato di ogni algoritmo, si intende mostrare quale algoritmo di clustering approssimi meglio le classi fornite dal dataset. Si richiede pertanto non solo di definire una metodologia per confrontare i risultati del clustering con la classificazione attesa, ma anche di descrivere in modo sintetico le caratteristiche distintive di ciascun cluster di specie prodotto dall'algoritmo oggetto della valutazione.

Esplorazione del dataset

Procediamo con l'importazione del dataset (salvato in formato .csv) importandolo sotto forma di dataframe per poi successivamente analizzare per ogni feature:

- il tipo (category, float, etc)
- presenza di dati mancanti
- valori presenti nel dataset per ogni features
- La correlazione tra le features e ogni singola classe (specie) target

Nella cartella dataset ci sono:

- zoo.data: dataset contenente in ogni riga la descrizione di ogni animale come 18 attributi espressi sotto forma di colonne e nella colonna "type" la specie di appartenenza.
  Sono per facilità le classi "originali" sono state ridotte di uno nella loro corrispondenza numerica così da poter far partire la numerazione da 0 come nel linguaggio di programmazione python
- class.csv: il dataset delle classi o file dei risultati, contiene al suo interno la suddivisione nelle 7 diverse classi dei vari elementi del dataset "zoo.data". Di ogni classe espressa sotto forma di riga sono riportate le seguenti informazioni:

- Class_Number

- Number_Of_Animal_Species_In_Class

- Class_Type

-Animal_Names

```
import support_function as sf # importiamo la libbreria di supporto
import pandas as pd
import os
colnames = ['animal name', 'hair','feathers','eggs','milk','airborne','aquatic','predator','toothed','b
zoo_data = pd.read_csv(os.getcwd()+'/Dataset/zoo.data', delimiter=',', names=colnames, header=None)
zoo_data['type']=zoo_data['type']-1
print(zoo_data)
```

```
# Stampiamo le informazioni di tipo, presenza di dati nulli, nome e numero della colonna del dataframe
print(zoo_data.info())
# Valutiamo per ogni colonna del dataframe quanti elementi unici sono presenti
print(zoo_data.nunique())
# Verifichiamo che non ci siano elementi mancanti all'interno delle singole colonne del dataframe
# e facciamo un bar plot perentuale
zoo_data.isnull().sum(axis=0)/zoo_data.shape 0
```

Il dataset è composto dalle seguenti features:

- animal name: unico per ogni elemento
- hair: Boolean
- feathers: Boolean
- eggs: Boolean
- milk: Boolean
- airborne: Boolean
- aquatic: Boolean
- predator: Boolean
- toothed: Boolean
- backbone: Boolean
- breathes: Boolean
- venomous: Boolean
- fins: Boolean
- legs: Numeric (insieme di valori: {0,2,4,5,6,8})
- tail: Boolean
- domestic: Boolean
- catsize: Boolean
- type: Numeric (variabile obbiettivo, intero nel range [0,6])

Procediamo quindi alla verifica di eventuali valori mancanti all'interno del dataset e del numero di possibili valori che può assumere ogni variabile:

```
# Valutiamo per ogni colonna del dataframe quanti elementi unici sono presenti
print(zoo_data.nunique())
# Verifichiamo che non ci siano elementi mancanti all'interno delle singole colonne del dataframe
# e facciamo un bar plot perentuale
zoo_data.isnull().sum(axis=0)/zoo_data.shape[0]
```

Casiatica per ogni Feature

```
animal name    100
hair             2
feathers         2
eggs             2
milk             2
airborne         2
aquatic          2
predator         2
toothed          2
backbone         2
breathes         2
venomous         2
fins             2
legs             6
tail             2
domestic         2
catsize          2
type             7
dtype: int64
```

Valori mancanti

```
animal name    0.0
hair           0.0
feathers       0.0
eggs           0.0
milk           0.0
airborne       0.0
aquatic        0.0
predator       0.0
toothed        0.0
backbone       0.0
breathes       0.0
venomous       0.0
fins           0.0
legs           0.0
tail           0.0
domestic       0.0
catsize        0.0
type           0.0
dtype: float64
```

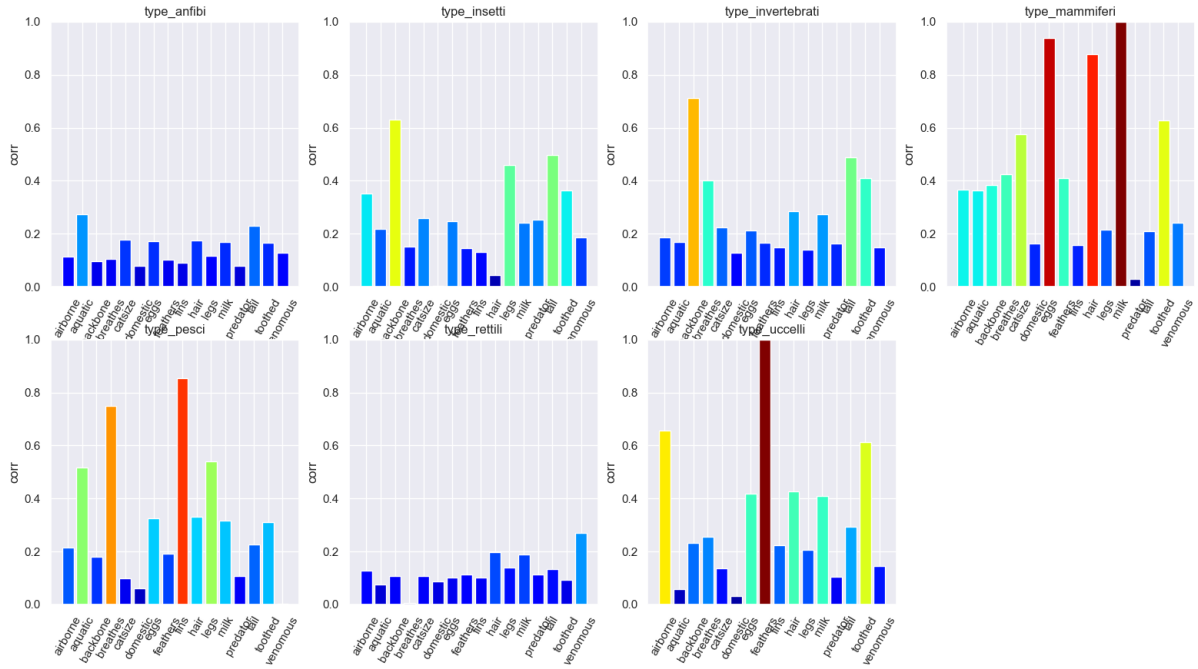
Procediamo quindi a costruire manualmente (in italiano) il dizionario che ci permetterà di identificare ogni specie tramite il suo numero di riferimento [0,6]

species_dict =dict(list(enumerate(['mammiferi', 'uccelli', 'rettili', 'pesci', 'anfibi', 'insetti','invertebrati'])))

print(species_dict)

Analisi del Dataset

Per capire se ci sono feature che presentano una maggiore correlazione con alcune specie costruiamo una matrice di correlazione e la grafichiamo tramite heatmap (che omettiamo in questo pdf ma è presente nel file main.jpyter) e barplot.



Possiamo osservare come alcune features abbiano una elevata correlazione con determinate classi:

0. mammiferi

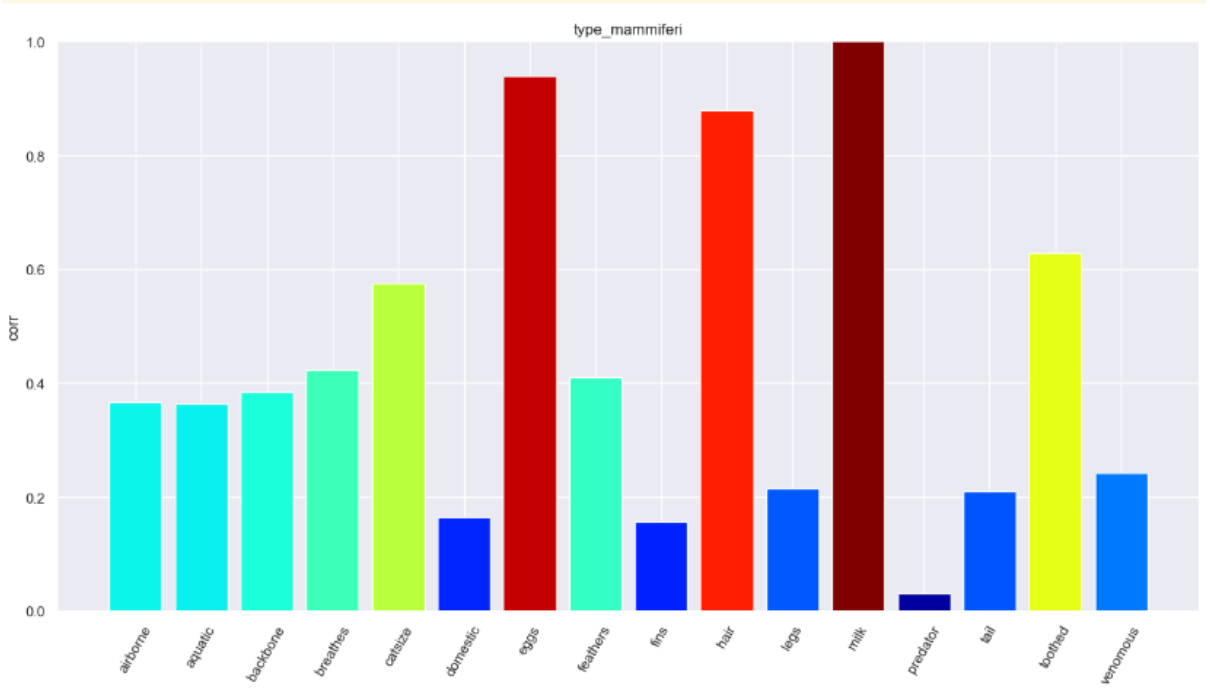
La classe presenta la maggiore correlazione per le feature:

- milk (Latte)

- hair (pelo)

- eggs (uova)

Tutti i mammiferi sono provvisti di ghiandole mammarie, di peli (nei mammiferi marini essi sono solo accennati e solo durante lo sviluppo embrionale) e solo due specie tra loro depongono uova (ornitorinco e echidna)



1. Uccelli

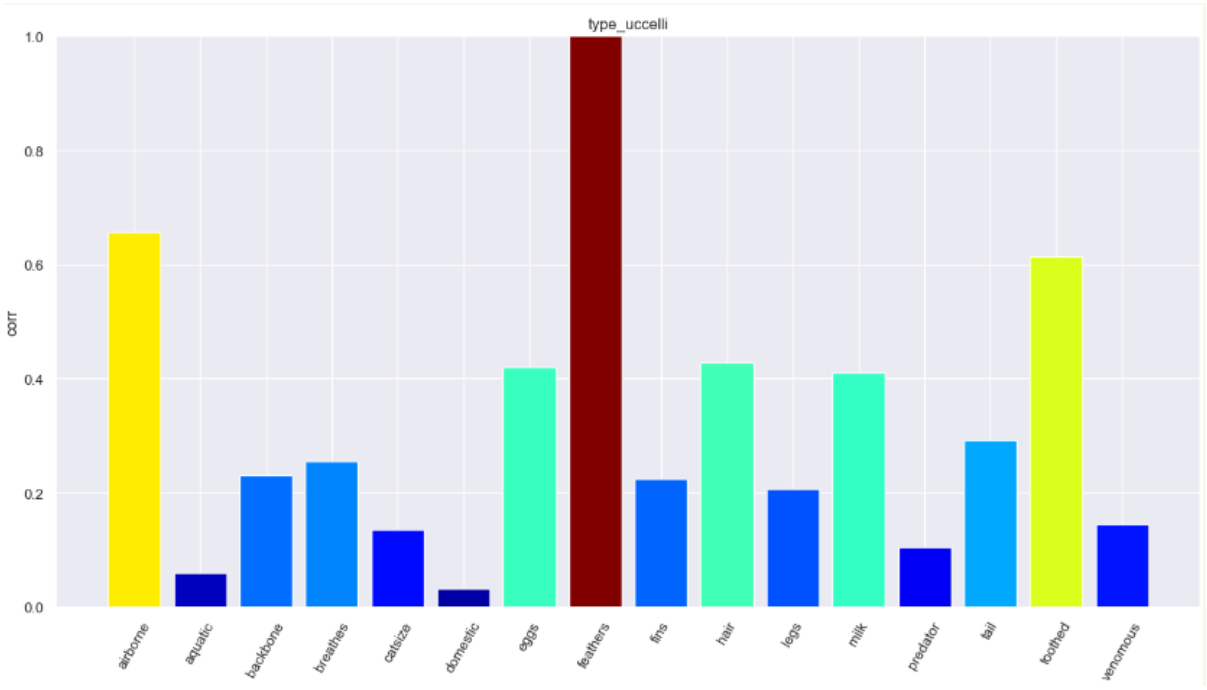
La classe presenta la maggiore correlazione per le feature:

- feathers (piume)

- airbone (volante)

- toothed (dentati)

La maggiorparte degli uccelli hanno piume, volano e presentano una dentatura più o meno marcata nel becco



3. Pesci

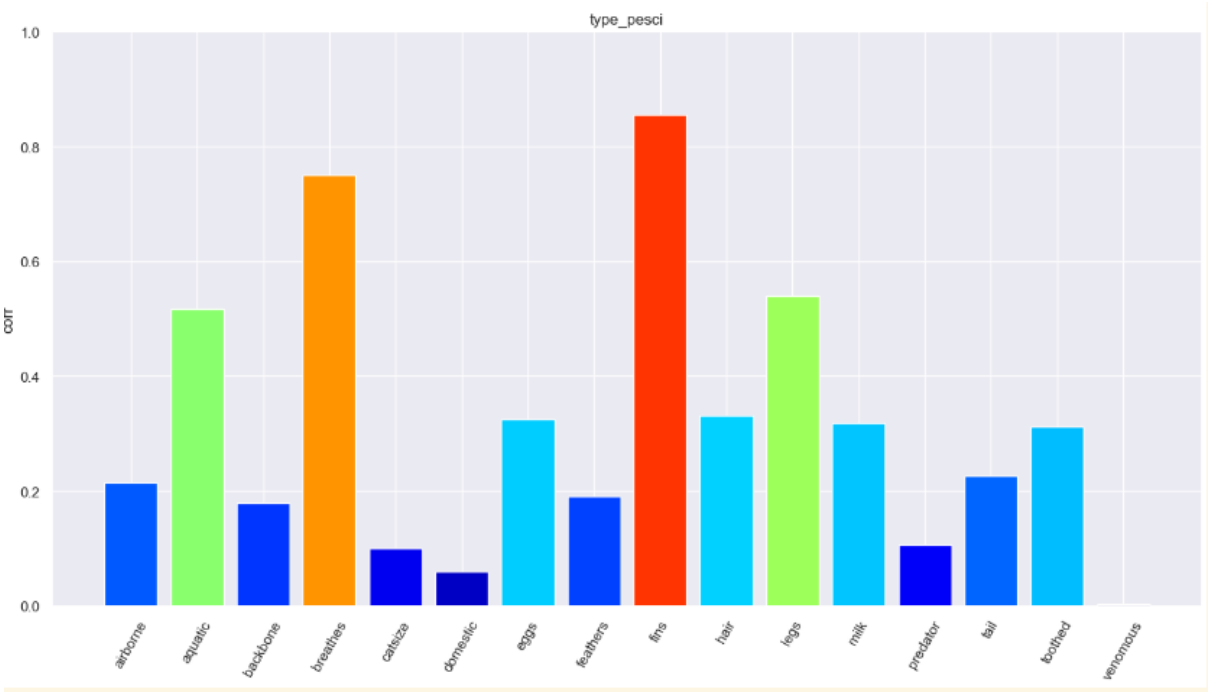
La classe presenta la maggiore correlazione per le feature:

- fins (pinne)

- bresthes (respira)

- legs (gambe)

I pesci sono un gruppo eterogeneo di organismi vertebrati fondamentalmente acquatici, dotati di pinne, che respirano attraverso le branchie.



5. Insetti

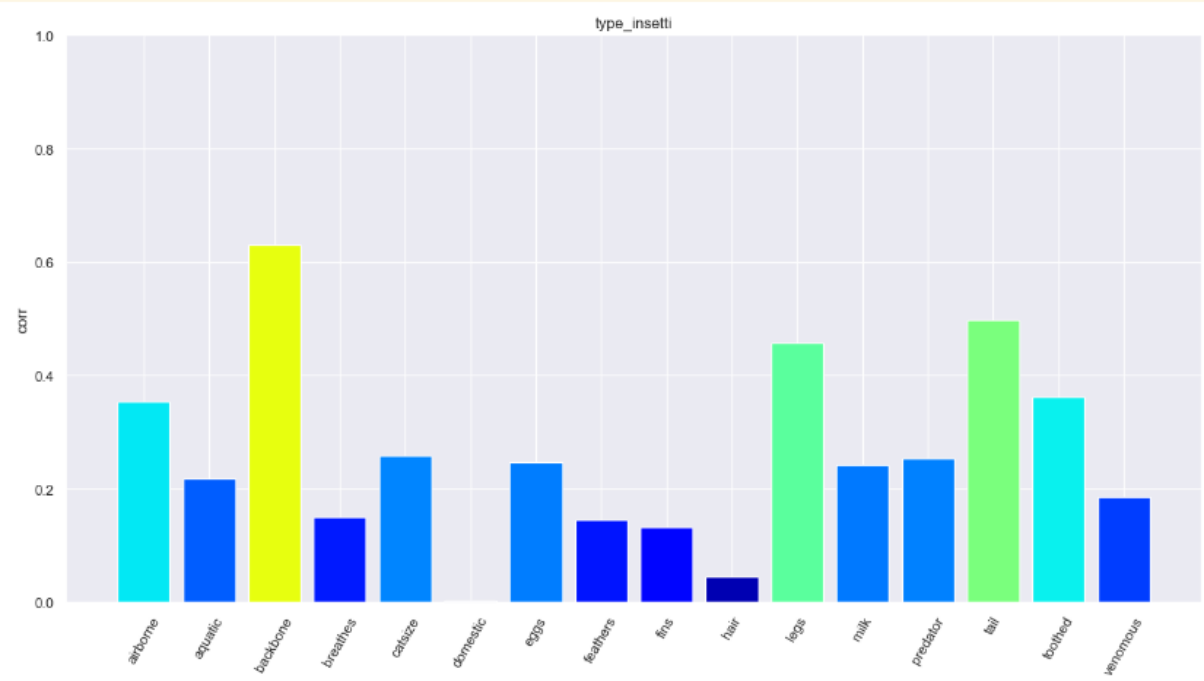
La classe insetti presenta la maggiore correlazione per le feature

- backbone (spina dorsale)

- tail (coda)

- legs (gambe)

Essa è la classe più ampia e popolosa nel mondo reale, ritroviamo tale correlazione con alcune caratteristiche descrittive della specie in quanto essi hanno uno scheletro esterno (esoscheletro), più zampe e molti di loro presentano un allungamento dell'esoscheletro descrivibile come coda



6. Invertebrati

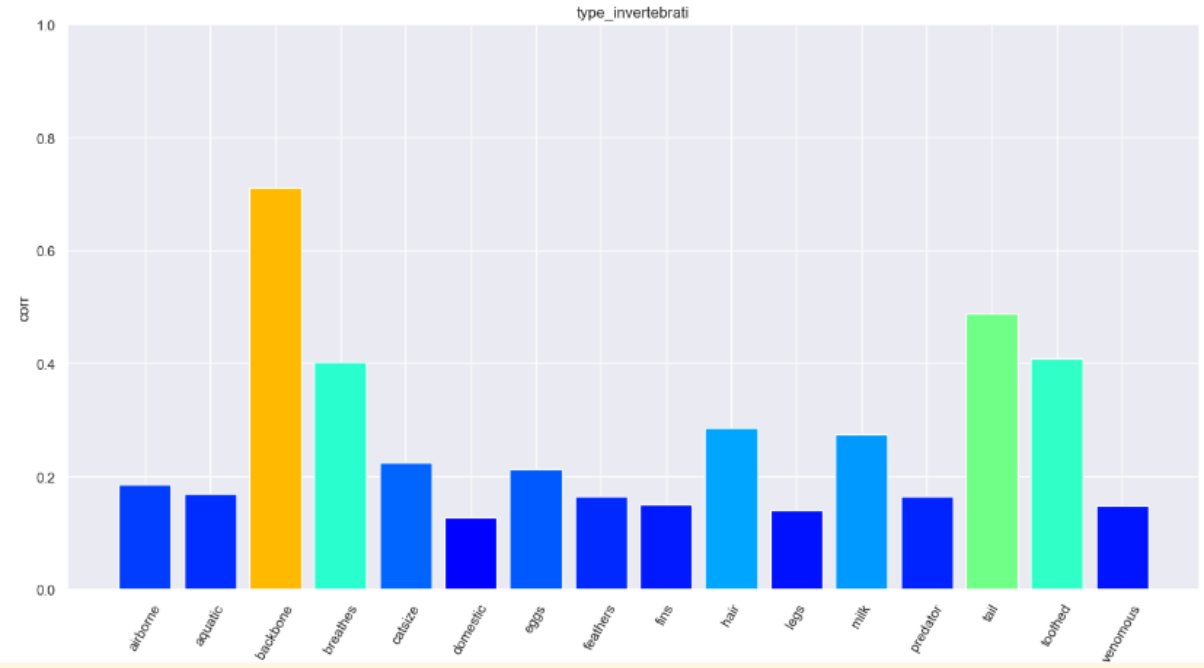
La classe presenta la maggiore correlazione per le feature:

- backbone (spina dorsale)

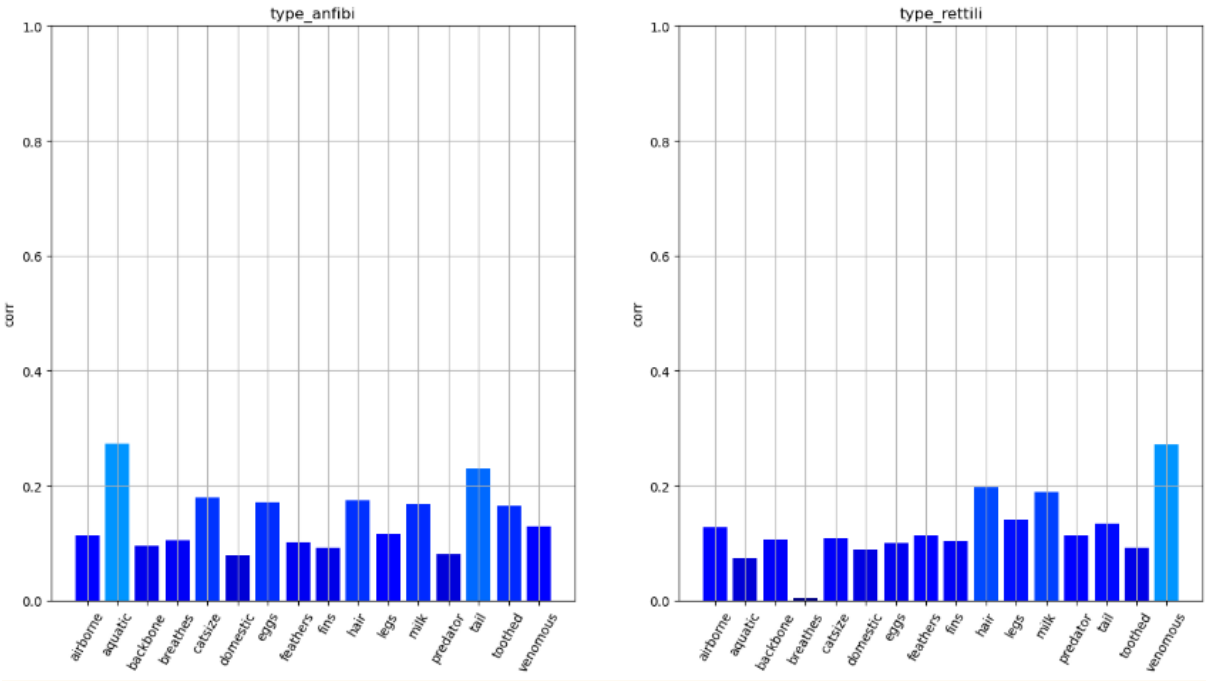
- tail (coda)

- toothed (dentati)

Gli animali invertebrati sono gli animali privi di colonna vertebrale e di scheletro interno, molti di loro presentano un allungamento dell'esoscheletro descrivibile come coda e non presentano denti



Le altre due classi non presentano una spiccata correlazione tra le features, esso può essere dovuto al basso numero di campioni



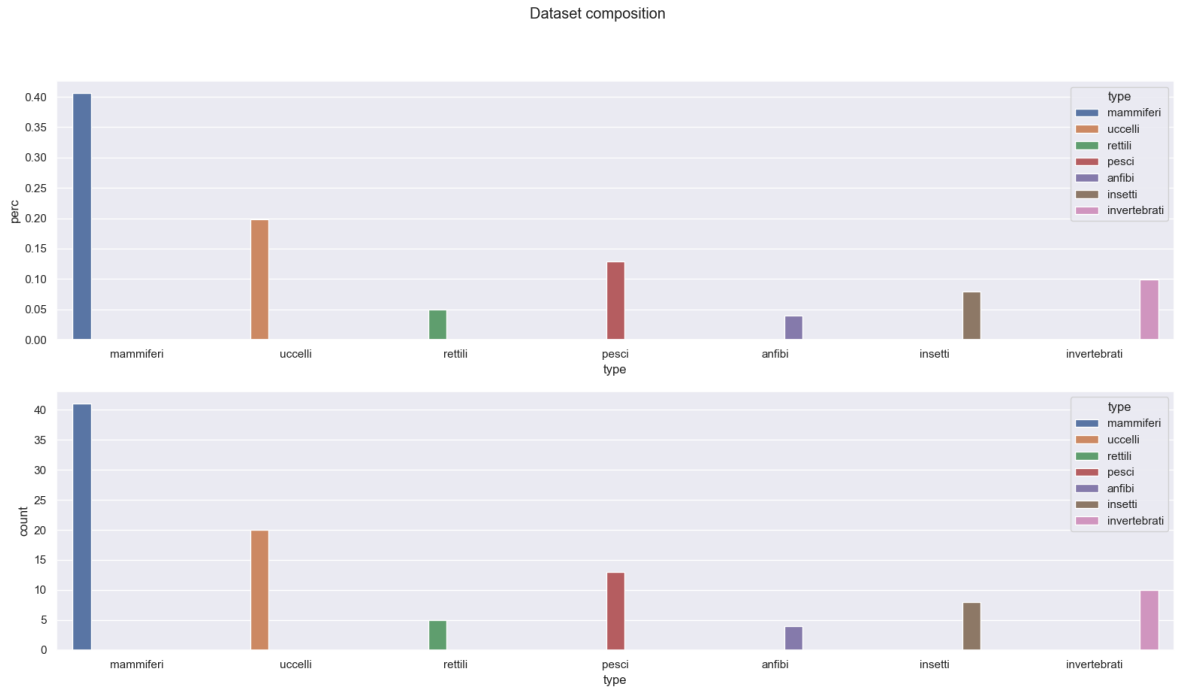
Dataset Composition

Valutiamo ora la composizione del dataset espressa come:

- composizione percentuale delle varie specie

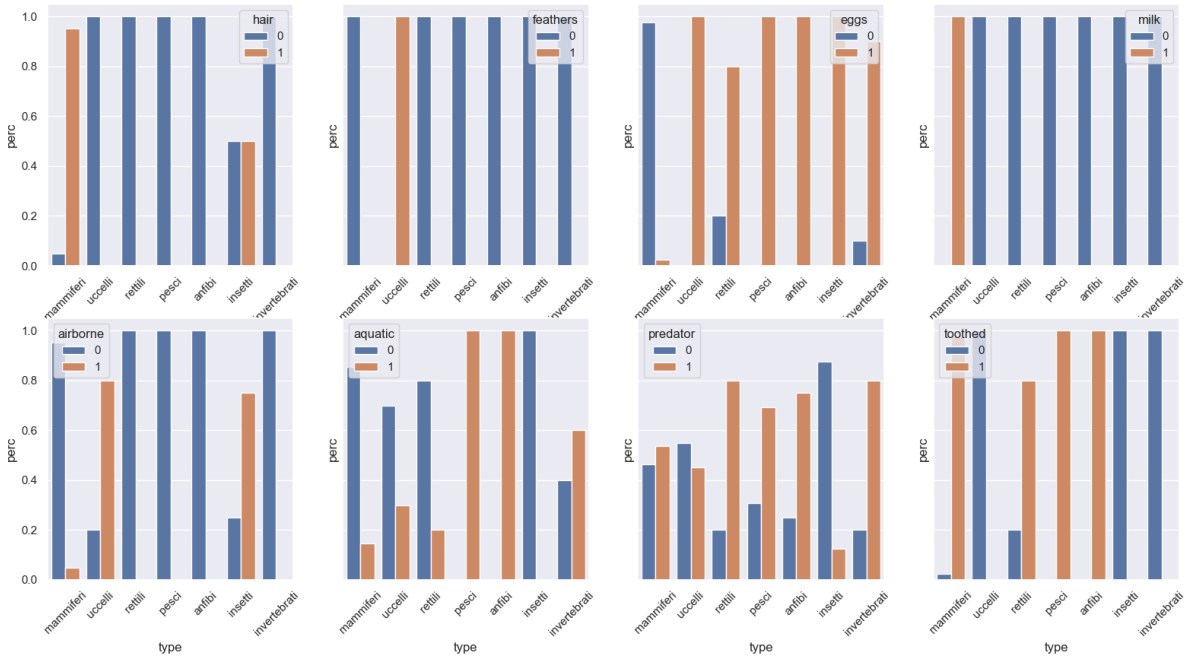
- numero campioni per ogni specie

Questi due numeri sono molto vicini in quanto abbiamo un dataset composto da 101 elementi

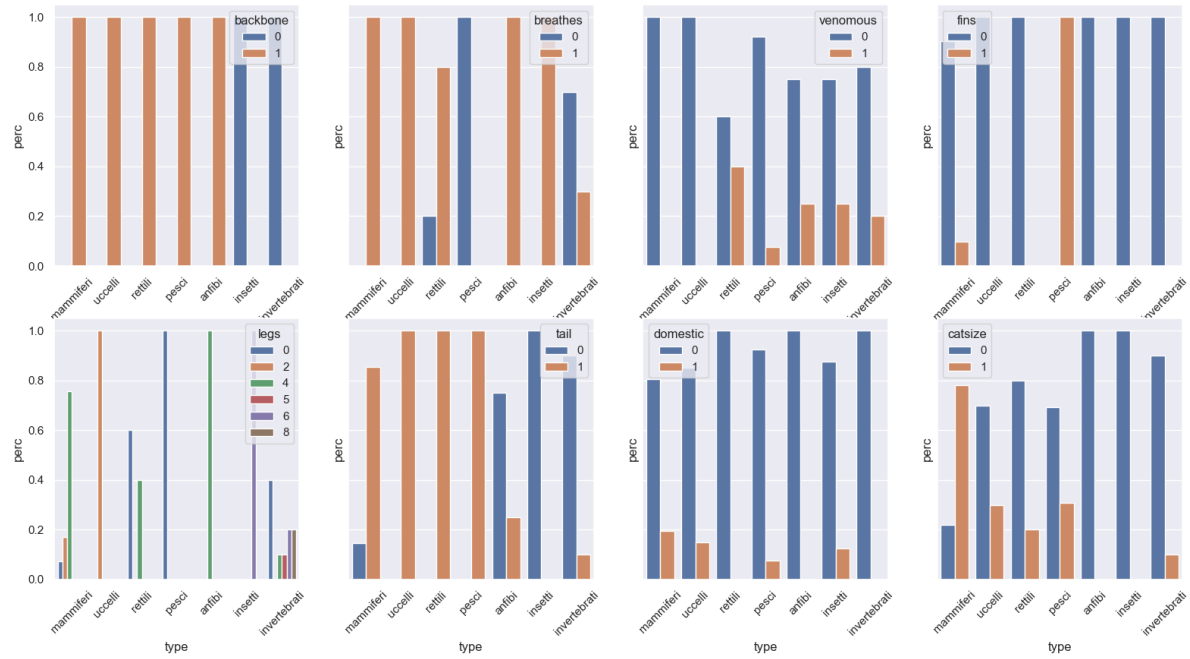


Per ogni classe osserviamo anche la distribuzione delle varie feature

Class features composition



Class features composition



Evaluation Metrics for Clustering Algorithms

Al fine di poter valutare e confrontare i vari modelli di clustering useremo differenti indici di performance.

Confusion Matrix

Il confronto tra la classi a cui appartengono realmente i singoli elementi del dataset e quella loro attribuita dall'algoritmo, la cui label è assegnata tramite majority voting, può essere riassunto tramite una matrice di confondimento:

- essa ci mostra per ogni item la sua classe originale e la classe assegnata dal modello

- sulla diagonale principale troviamo gli elementi "clusterizzati" correttamente

Faremo uso di questa matrice per riassumere in modo grafico insieme allo scatterplot (specie originale vs specie assegnata) il comportamento dell'algorimo di clusterizzazione

Rand index (RI)

Statistica che quantifica la tendenza al raggruppamento di un certo dataset a valle di una procedura di clustering, esso si basa sui principi:

- Ogni elemento appartiene ad uno ed un solo cluster

- Ogni cluster è definito sia dagli elementi che contiene che da quelli che non contiene

- ogni elemento ha la stessa importanza nel definire la qualità di una partizione (pari peso tra gli elementi)

Quindi se una coppia di elementi è inclusa in uno stesso cluster da due distinte clusterizzazioni (label originali e label assegnate dall'algoritmo di clustering) vuol dire che c'è similarità altrimenti se la coppia viene "mappata" in due cluster differenti c'è dissimilarità (non considera le coppie che risultino scompagnate in entrambe le classificazioni)

$$Rand = \frac{\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} c_{ij}}{\binom{n}{2}} \quad con \quad c_{ij} = \begin{cases} 1 \ se\ U_i\ e\ U_j\ sono\ nello\ stesso\ cluster \\ \quad sia\ nella\ teorica\ che\ nella\ stimata \\ 1\ se\ U_i\ e\ U_j\ sono\ in\ cluster\ diversi \\ \quad sia\ nella\ teorica\ che\ nella\ stimata \\ 0\ otherwise \end{cases}$$

Il rand index varia tra 0 e 1 (clustering perfettamente coincidenti)

Mutual Information (MI, NMI)

Le mutual information (MI, NMI, AMI) misurano l'"accordo" tra le assegnazioni del cluster. Un punteggio più alto indica una somiglianza maggiore.

Il grado di accordo tra i cluster è calcolato da probabilità congiunte e marginali, espresse tramite gli indici:

- informazioni reciproche (MI)

- informazioni reciproche normalizzate (NMI), è la MI divisa per le entropie medie dei cluster

V-Measure

Misura la correttezza delle assegnazioni del cluster usando l'analisi dell'entropia condizionale. Un punteggio più alto indica una somiglianza maggiore.

Si possono usare due metriche per valutare la correttezza delle assegnazioni del cluster:

- Omogeneità (HS): ogni cluster contiene solo membri di una singola classe (un po'come la "precision")

$$\text{Homogeneity } (h) = 1 - \frac{\text{Conditional entropy given cluster assignments}}{\text{Entropy of (predicted) class}}$$
$$= 1 - \frac{H(C|K)}{H(C)}$$

- Completezza (CS):tutti i membri di una determinata classe sono assegnati allo stesso cluster (un po' come la "recall")

$$\text{Completeness } (c) = 1 - \frac{\text{Conditional entropy given cluster assignments}}{\text{Entropy of (actual) class}}$$
$$= 1 - \frac{H(K|C)}{H(K)}$$

- La V-Measure è la media armonica di omogeneità e misura di completezza, simile al f-score è una media armonica di precision e recall.

$$\text{V-measure } (v) = 2 \cdot \frac{h \cdot c}{h + c}$$

Silhouette Coefficient

Il coefficiente di silhouette misura la distanza between-cluster vs la distanza within-cluster. Un punteggio più alto significa cluster meglio definiti.

per ogni singolo campione il coefficiente di silhouette misura la distanza media del campione da tutti gli altri elementi dei cluster più vicini (between-cluster) contro tutti gli altri elementi del suo stesso cluster (within-cluster).

Se il risultato di tale rapporto è "alto" significa che il singolo cluster è "lontano" dagli altri cluster lui vicini e che il cluster è ben definito.

Il coefficiente di silhouette di un dataset prende il coefficiente di silhouette medio per ciascun elemento

$$\text{a} = \text{Average distance between sample and all other points in same cluster}$$
$$\text{b} = \text{Average distance between sample and all other points in next nearest cluster}$$
$$\text{Silhouette Coefficient } (s) = \frac{b - a}{\max(a, b)}$$

Il coefficiente di Silhouette va da -1 (modello molto scarso) a 1 (modello eccellente), al fine di poterlo usare successivamente con i restanti indici che vivono tra [0,1] applichiamo la funzione normalizzazione MIN-MAX con minimo scala a 0 e massimo scala a 1.

$$normalize_{value} = min_{scale} + \frac{x - min([0,1])}{max([0,1]) - min([0,1])} * (max_{scale} - min_{scale})$$

ovvero:

$$normalize_{value} = \frac{x+1}{2}$$

Model_comparison dataframe

Procederemo quindi alla valutazione dei vari algoritmi di clustering tramite gli indici selezionati.

Per ogni algoritmo di clustering saranno valutate le performace per diversi iperparametri, tutti i valori saranno inseriti nel "model_comparison" dataframe il quale avrà la seguente struttura:

```
model_comparison =[model_type, model, ri, MI, NMI, HS, CS, V, siluetteN]
```

Algoritmi di Clustering

Di seguito una veloce panoramica sui i vari algoritmi di clustering presi in esame, omettiamo il codice utilizzato per valutarli al variare dei specifici iperparametri per esso rimandiamo al file jupiter main.

Affinity Propagation (AP)

Algoritmo di clustering basato sul concetto di “passaggio di messaggi” tra i campioni.

Esso non richiede che siano stabili a priori il numero dei cluster ma cerca tramite la misura di somiglianza Standar (negativo delle distanze euclidee al quadrato) i campioni "rappresentativi" per l’intero set di dati.

L'iperparametro che andremo a valutare è il “damping”(“smorzamento”) tra 0,5 e 1 .

Agglomerative Clustering

Il clustering agglomerativo appartiene agli algoritmi di “clustering gerarchico”, ovvero partendo da dall'associazione di un ckluster ad ogni elemento del dataset esegue la "fusione" di dei vari cluster fino a raggiungere il numero desiderato di k-gruppi.

Si è allenato il modello Agglomerative Clustering con l'iperparametro k con valore in [4,10].

BIRCH

Balanced Iterative Reducing and Clustering using Hierarchies, algoritmo di clustering che visualizza i dati in una struttura ad albero da cui vengono estratti i centroidi dei cluster.

Tale algoritmo integra il clustering gerarchico (utilizzato nello stadio iniziale di microclustering) con altri metodi di clustering quali il partizionamento iterativo (utilizzato nello stadio di macroclustering successivo).

Raggruppare in modo incrementale e dinamico i punti di dati metrici multidimensionali, con l’obiettivo di ottenere un clustering migliore in condizioni limitate, come una scarsa memoria o un tempo ridotto. Supera così le due principali difficoltà dei metodi di clustering agglomerativo (scalabilità e l’incapacità di annullare ciò che è stato fatto nei passi precedenti).

Si è allenato il modello BIRCH con l'iperparametro th (threshold) e k (n_clusters) usati per fornire una stima del numero di cluster.

DBSCAN

Density-Based Spatial Clustering of Applications with Noise, ricerca aree ad alta densità di campioni nello spazio multidimensionale e l'espansione di quelle aree come cluster, basandosi sulle loro caratteristiche.

Tale algoritmo è stato specificamente ideato per scoprire cluster di forma arbitraria. Basta avere un parametro di input e DBSCAN è in grado di aiutarti a determinare il suo esatto valore.

Si è allenato il modello DBSCAN con l'iperparametro "eps" costante (0.75) e n (min_samples) che varia da 1 a 10 con passo 2.

OPTICS

Ordering Points To Identify the Clustering Structure è una versione modificata del DBSCAN. Tale algoritmo differentemente dai precedenti non produce l'analisi di cluster di un set di dati, ma mostra la struttura di clustering basata sulla densità di un database.

Si è allenato il modello OPTICS con l'iperparametro "eps" costante (0.75) e n (min_samples) che varia

K-means

Tale algoritmo richiede di conoscere a priori il numero di cluster totali, classifica i punti dell'input nei vari cluster.

Aggiusta iterativamente la dimensione dei cluster, spostando i punti nei cluster più vicini a loro ad ogni iterazione, aggiorna i centroidi dei cluster finché non si raggiunge un criterio di ottimalità (minimizzare la varianza totale intra-gruppo).

Si è allenato il modello k-means con l'iperparametro k con valore in [4,10].

Mini-Batch K-Means

Mini-Batch K-Means è una versione modificata di k-means che aggiorna i centroidi dei cluster usando dei piccoli batch dei campioni invece dell'intero set di dati, tale escamotage elimina il rumore e riduce i tempo di addestramento sui grandi dataset.

Si è allenato il modello Mini-Batch K-Means con l'iperparametro k con valore in [4,10].

Mean shift

Il clustering Mean shift adatta i centroidi in base alla densità dei campioni nello spazio delle caratteristiche. È utilissimo per rilevare le varie modalità di densità per i dati discreti, perché attua uno spostamento medio ricorsivo al punto stazionario più vicino all'interno di una funzione di densità.

Spectral Clustering

Algoritmo di clustering di spettro, prevede di usare i primi autovettori di una matrice derivata dalla distanza tra i punti.

Si è allenato il modello Spectral Clustering con l'iperparametro k (n_clusters) con valore in [4,10].

Gaussian Mixture

Il modello a miscela gaussiana, usa il principio della funzione di densità di probabilità multivariata con una miscela di distribuzioni di probabilità gaussiane.

Si è allenato il modello Gaussian Mixture con l'iperparametro k (n_clusters) con valore in [4,10].

Best Model Selection

Avendo esplorato diversi algoritmi di clustering e per quasi ogni uno diversi possibili iperparametri si è scelto di estrarre per ogni algoritmo di clustering il best-model basato sull'indice ottenuto dalla media degli score index:

- rand index
- normalized mutual nfo
- v_measure il quale racchiude in se le informazioni degli indici (homogeneitye completeness)
- siluetteN

La scelta dell'uso dell'indice medio è dovuta all'osservazione che ogni indice osserva un aspetto specifico del risultato di clustering, siccome tutti vivono in un range tra [0,1] essi sono già normalizzati e quindi nessuno di essi "pesa" di più degli altri.

A livello di codice si è proceduto con il calcolo del valore medio degli indici osservati (resume_index) per ogni rida del dataframe model_comparison e a raggruppare gli elementi per tipologia di modello per poter estrarre per ogni gruppo i "migliori" rappresentanti

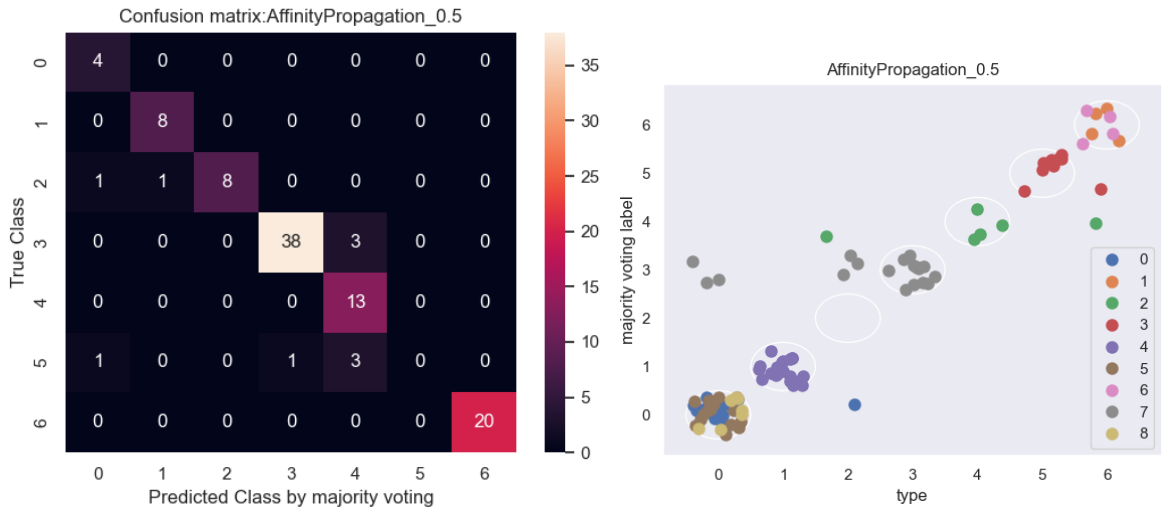
model_type	model	ri	MI	NMI	HS	CS	V	siluetteN	resume_index
AffinityPropagation	AffinityPropagation_0.6	0.858416	1.4106	0.761383	0.851291	0.688652	0.761383	0.703699	0.77122
AgglomerativeClustering	AgglomerativeClustering_9	0.908119	1.46699	0.816925	0.885327	0.758335	0.816925	0.720019	0.815497
birch	birchMlustering_k:_9_th_0.01	0.90099	1.43227	0.805376	0.864373	0.753918	0.805376	0.724538	0.80907
birch	birchMlustering_k:_9_th_0.03	0.90099	1.43227	0.805376	0.864373	0.753918	0.805376	0.724538	0.80907
dbscan	dbscan_eps_0.75_min_samples_1	0.787525	1.65701	0.604699	1	0.433382	0.604699	0.80198	0.699726
gaussian_mixture	gaussian_mixture8	0.907129	1.42961	0.817077	0.862768	0.775982	0.817077	0.720069	0.815338
kmeans	kmeans_8	0.905941	1.37504	0.786414	0.829832	0.747313	0.786414	0.722218	0.800247
mb_kmeans	mb_kmeans_7	0.922574	1.30026	0.786528	0.7847	0.788363	0.786528	0.710817	0.801612
mean_shift	mean_shift	0.708713	0.693742	0.542629	0.418671	0.770863	0.542629	0.689337	0.620827
optics	optics_m_eps_0.75_min_samples_8	0.810693	1.02642	0.651478	0.619442	0.687009	0.651478	0.633657	0.686827
spectral_clustering	spectral_clustering9	0.913663	1.51173	0.847284	0.912324	0.7909	0.847284	0.721322	0.832388

Per ogni "campione" di ogni algoritmo di clustering andremo ad analizzare le performance tramite:

- matrice di confusione
- scatterplot che presenta:
    - asse x: la specie originale dell'elemento
    - asse y: la specie assegnata all'elemento tramite il metodo del majority voting (ovvero per ogni cluster assegnamo l'etichetta degli elementi a massima rappresentanza all'interno di essa)
    - colore diverso per ogni cluster generato dall'algoritmo (riportato il leggenda)
- BarPlot della correlazione tra le features e il cluster assegnato dall'algoritmo comparato nel grafico speculare sottostante) con la correlazione tra le features e la specie (proveniente dal majority voting del cluster) all'interno del dataset originale

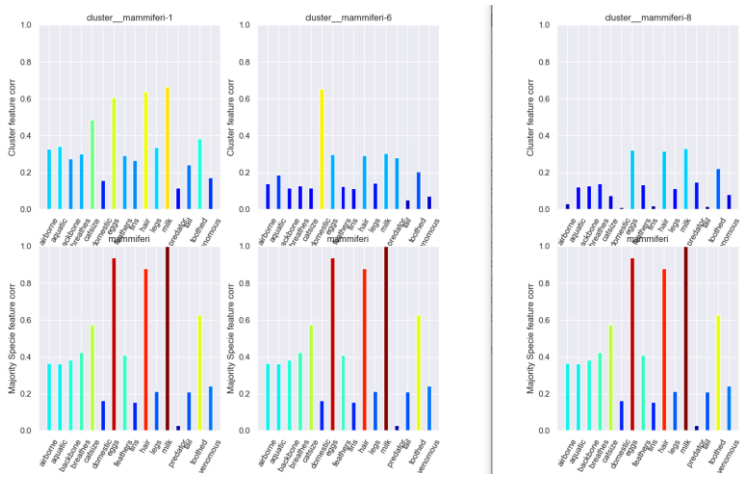
Siccome l'analisi di 11 algoritmi potrebbe risultare tediosa al lettore cercheremo di riportare solo le informazioni principali (per una versione più estesa rimandiamo al notebook main.ipynb.

AffinityPropagation k=0.5



Possiamo osservare che l'algoritmo clusterizza il dataset in 9 gruppi contro i 7 reali, in particolare:

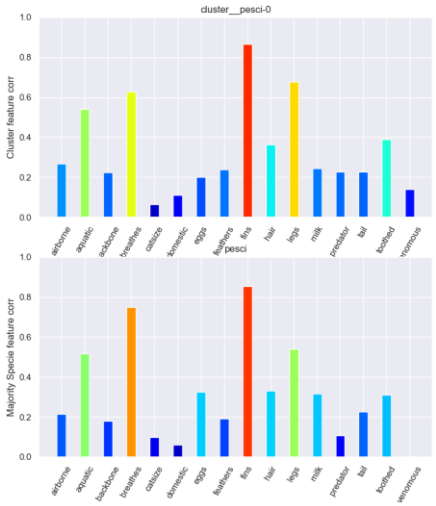
mammiferi (0): sono stati clusterizzati in tre gruppi (1,2,8) più alcuni elementi che sono stati clusterizzari nel cluster 0 dei "pesci" (majority voting label d'ora in poi MV). Osservando le matrici di correlazione dei gruppi 1,2,8 possimao notare come le feature a maggior correlazione con il cluster sono le stesse della specie-cluster "reale" dei mammiferi, in particolare il cluster 2 presenta una alta correlazione con la feature "domestic" a differenza del cluster 8 in cui la correlazione con tale feature è a zero. Il cluster 1 è quello con valori di correlazione rispetto alle features "eggs", "hair" e "milk" più simili se non addirittura maggiori della specie originale



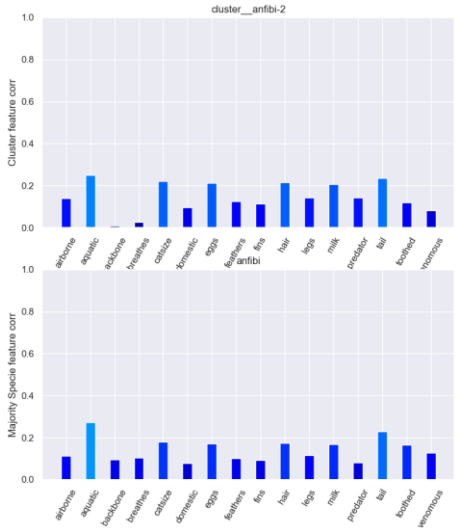
uccelli (1): sono stati clusterizzati in tutti in un unico gruppo (7)

rettili(2): sono stati "assorbiti" da tre gruppi che tramite MV sono associati alle etichette mammiferi, pesci e anfibi.

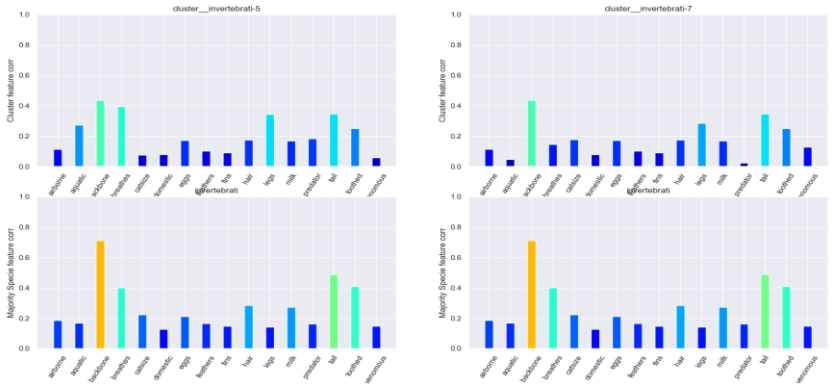
pesci(3): sono stati clusterizzati in tutti in un unico gruppo (0) al cui interno troviamo anche elementi esterni di altre specie. Osservando la matrici di correlazione si nota che le feature a maggior correlazione con il cluster sono le stesse della specie-cluster "reale":



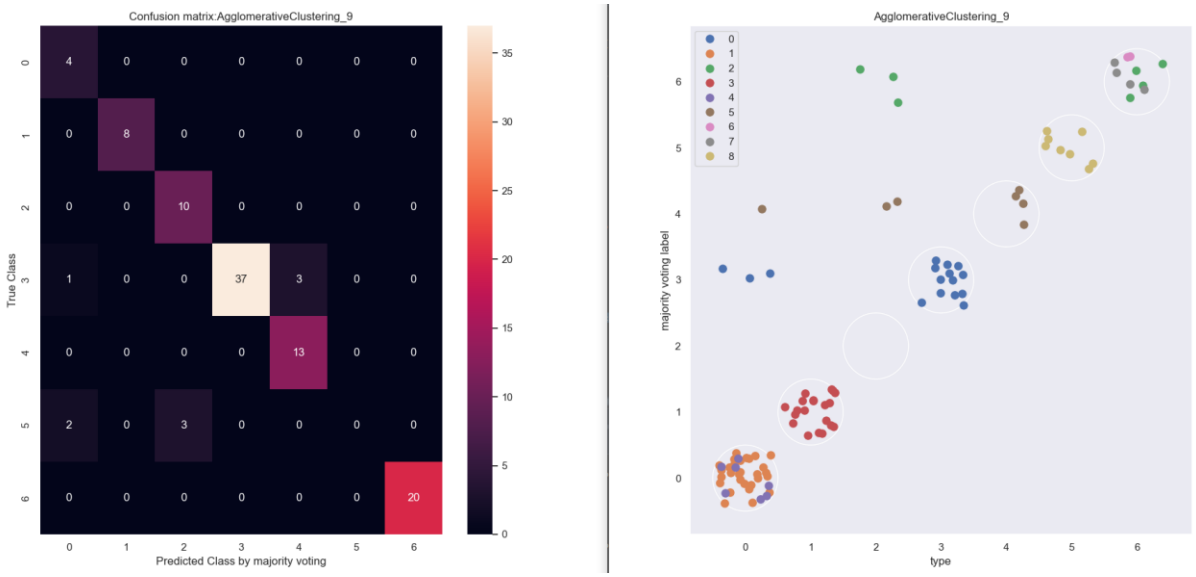
anfibi (4): sono stati clusterizzati in tutti in un unico gruppo (5) degli anfibi (MV),con un elemento proveniente dai rettili ed uno dagli invertebrati Il profilo di correlazione tra features e gruppo è simile a quello della specie originale



invertebrati (6): sono stati clusterizzati in due gruppi (3,4) degli "invertebrati" (MV) ,più un outsider finito negli "anfibi". Possiamo osservare come le matrici di correlazione delle features di ambo i gruppi (1,2) sono molto simili alle matrici di correlazione della specie "originale", la distinzione maggiore tra i due gruppi la associamo alla features "legs" combinata a "breathers"

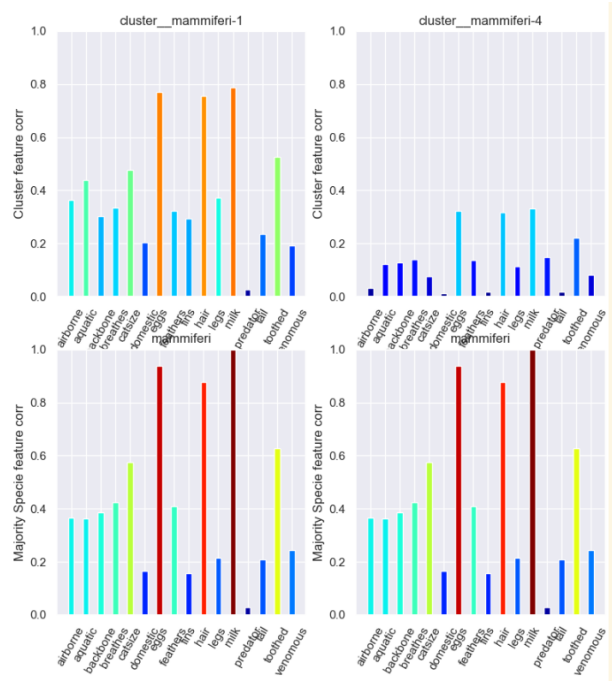


Agglomerative Clustering k=9



Possiamo osservare che l'algoritmo clusterizza il dataset in 9 gruppi contro i 7 reali, in particolare:

mammiferi (0): sono stati clusterizzati in due gruppi (4,1) più alcuni elementi che sono stati clusterizzari nel cluster 5 degli "anfibi" e 0 dei "pesci" (MV). Osservando le matrici di correlazione dei gruppi 4,1 possiamo notare come le feature a maggior correlazione con il cluster sono le stesse della specie-cluster "reale" dei mammiferi, tra i due cluster si vede una maggiore correlazione con la feature "fins" nel cluster (1):



uccelli (1): sono stati clusterizzati in tutti in un unico gruppo (3)

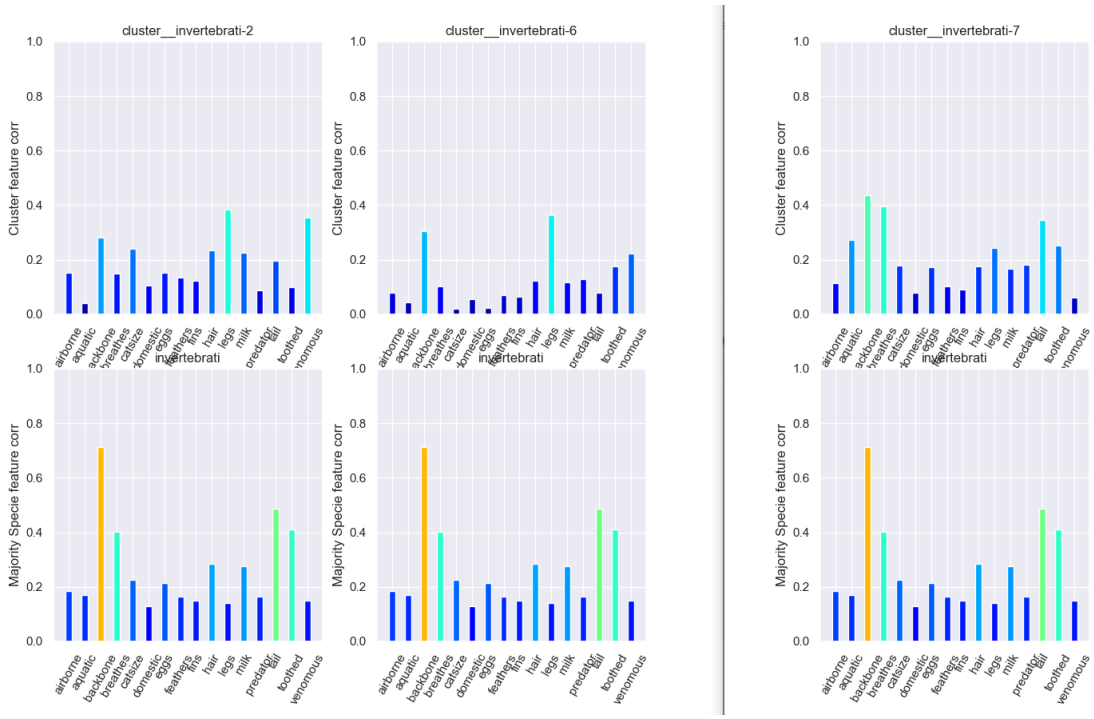
rettili(2): sono stati clusterizzati in due gruppi, il gruppo (4) degli "anfibi" (MV-lab) e il gruppo 2 degli invertebrati (MV-lab).

pesci(3):sono stati clusterizzati in tutti in un unico gruppo (0) al cui interno troviamo anche elementi esterni (mammiferi). Osservando la matrici di correlazione si nota che le feature a maggior correlazione con il cluster sono le stesse della specie-cluster "reale".

anfibi (4): sono stati clusterizzati in tutti in un unico gruppo (5) degli "anfibi" (MV), con alcuni elementi provenienti da rettili e mammiferi. Anche qui il profilo di correlazione tra features e gruppo è simile a quello della specie originale

insetti(5): sono stati clusterizzati in tutti in un unico gruppo (8)

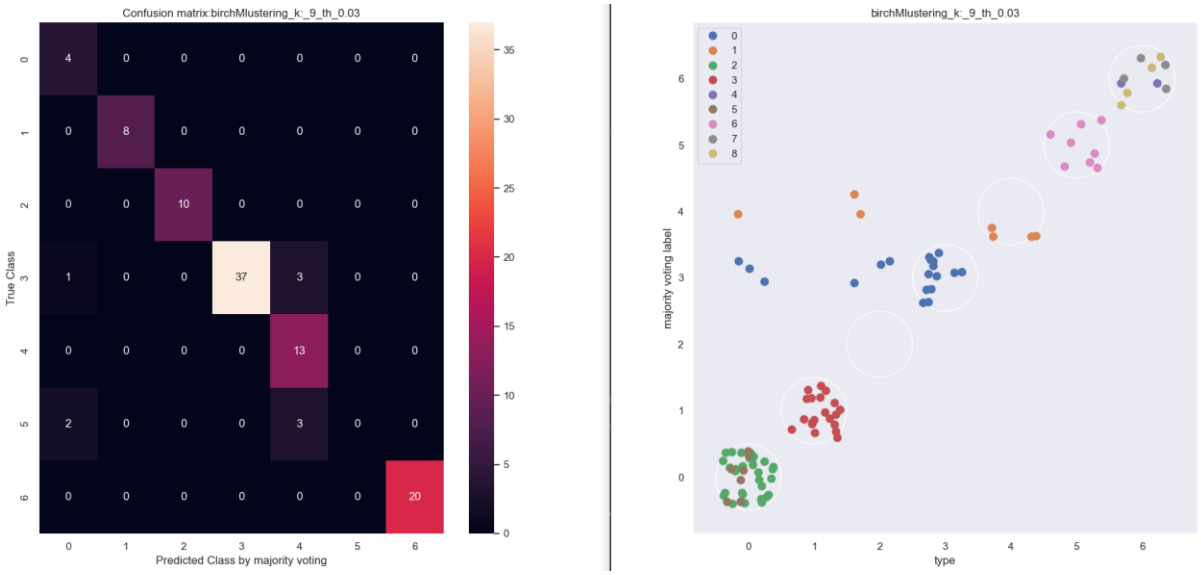
invertebrati (6): sono stati clusterizzati in tre gruppi (6,7) degli "invertebrati" (MV-lab) ,più alcuni outsider provenienti dai "rettili". Possiamo osservare come le matrici di correlazione delle features di tutti e tre i gruppi sono molto simili alle matrici di correlazione della specie "originale" per quello che riguarda le features che spiccano per maggior correlazione con il gruppo. Il gruppo 2 spicca per la feature "venonomous", il gruppo 6 per avere bassissima correlazione con "breathes" e "eggs" (a differenza della specie originale) il gruppo 7 è quello con il profilo più simile alla specie originale.



BIRCH

Di questo modello sono state valutate due varianti perché entrambe hanno avuto lo stesso Resume_index finale BIRCH con k=9 e th=0.01 e BIRCH con k=9 e th=0.03

Siccome le considerazioni sono simili per entrambi i modelli commenteremo solo il primo (th=0.01)



Possiamo osservare che l'algoritmo clusterizza il dataset in 9 gruppi contro i 7 reali, in particolare:

mammiferi (0): sono stati clusterizzati in due gruppi (2,5) più alcuni elementi che sono stati clusterizzari nel cluster 5 degli "anfibi" e 0 dei "pesci" (MV). Osservando le matrici di correlazione dei gruppi 2,5 possiamo notare come le feature a maggior correlazione con il cluster sono le stesse della specie-cluster "reale" dei mammiferi anche se il gruppo 2 ha un profilo con valori più vicini al profilo della specie originale.

uccelli (1): sono stati clusterizzati in tutti in un unico gruppo (3)

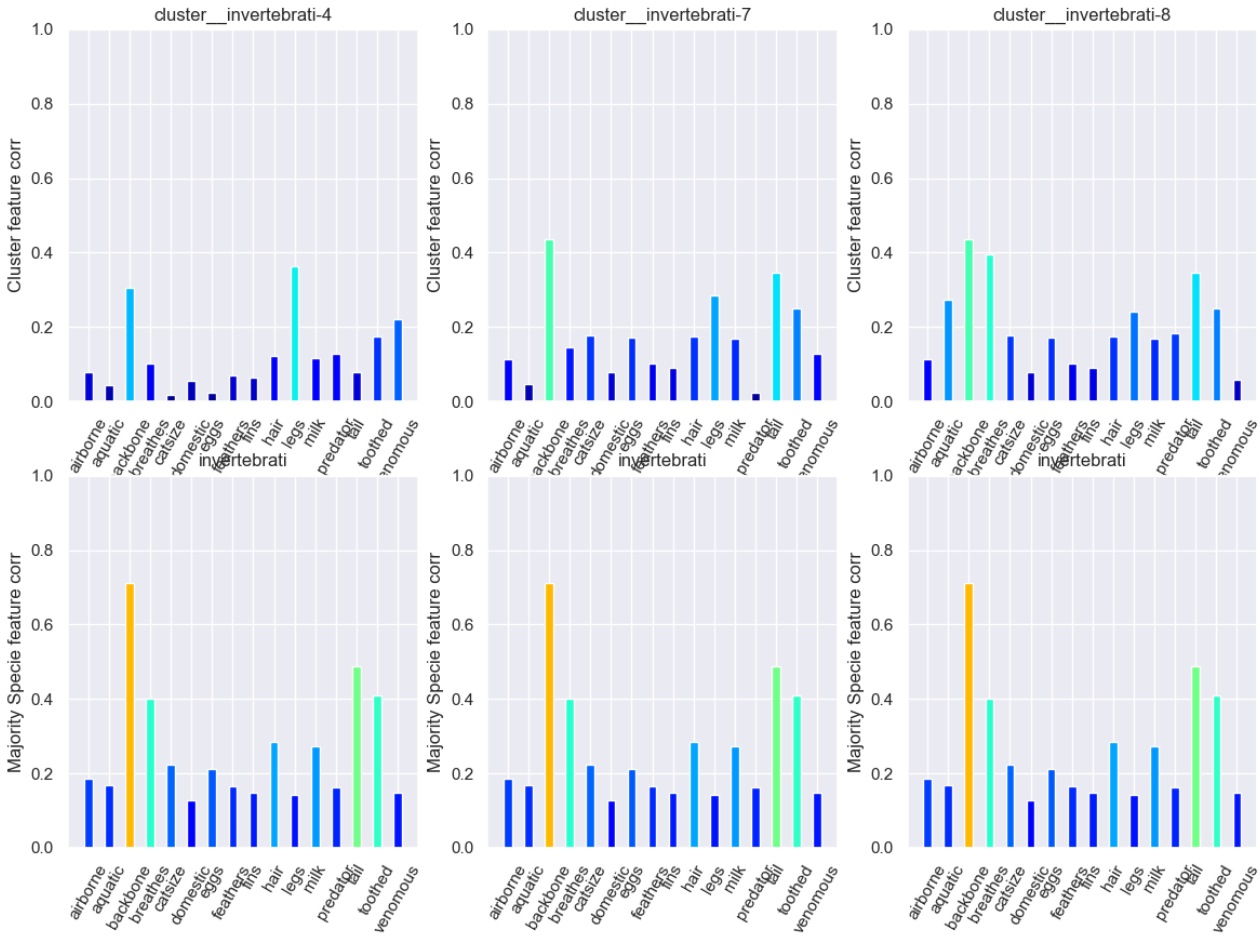
rettili(2): sono stati clusterizzati in due gruppi, il gruppo (1) degli "anfibi" (MV-lab) e il gruppo 0 dei pesci (MV-lab). Per le matrici di correlazione stesse osservazioni dei modelli precedente.

pesci(3): sono stati clusterizzati in tutti in un unico gruppo (0) al cui interno troviamo anche elementi esterni (mammiferi e rettili). Osservando la matrici di correlazione si nota che le feature a maggior correlazione con il cluster sono le stesse della specie-cluster "reale".

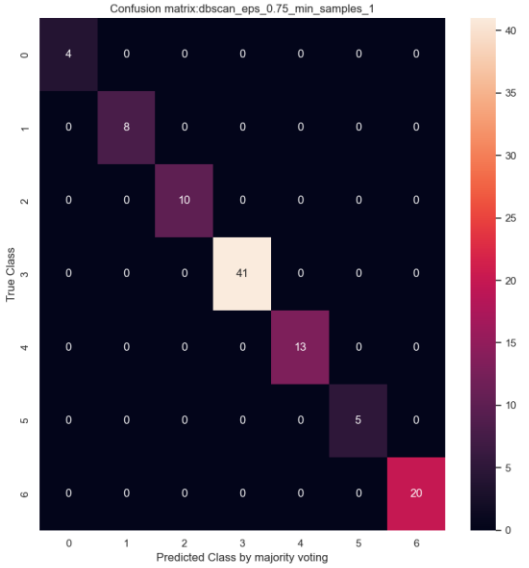
anfibi (4): sono stati clusterizzati in tutti in un unico gruppo (1) degli "anfibi" (MV-lab), per le matrici di correlazione stesse osservazioni dei modello precedente.

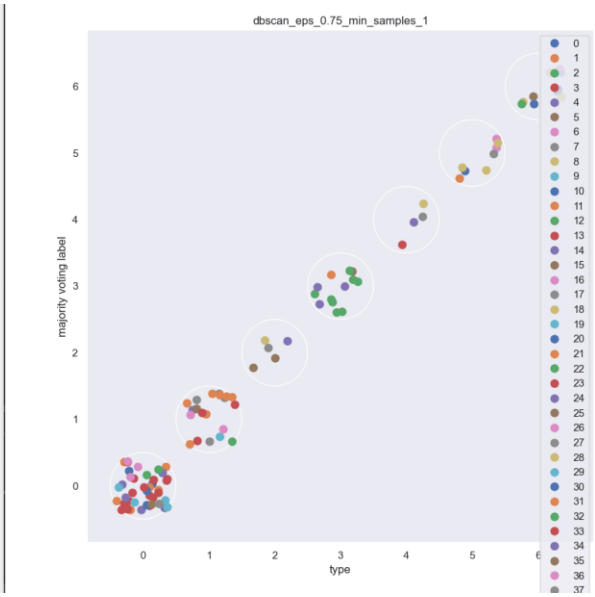
insetti(5): sono stati clusterizzati in tutti in un unico gruppo (6)

invertebrati (6): sono stati clusterizzati in tre gruppi (4,7,8) degli "invertebrati" (MV-lab) ,più un outsider finito nei "rettili". Possiamo osservare come le matrici di correlazione dei gruppi siano simili alla specie originale e variano tra loro principalmente nelle features "legs", "tail" e "toothed"


cluster__invertebrati-4
cluster__invertebrati-7
cluster__invertebrati-8
Cluster feature corr
Majority Specie feature corr
1.0
0.8
0.6
0.4
0.2
0.0
airborne
aquatic
backbone
breathes
catsize
domestic
eggs
feathers
fins
hair
legs
milk
predator
tail
toothed
venomous
invertebrati

DBSCAN n=1





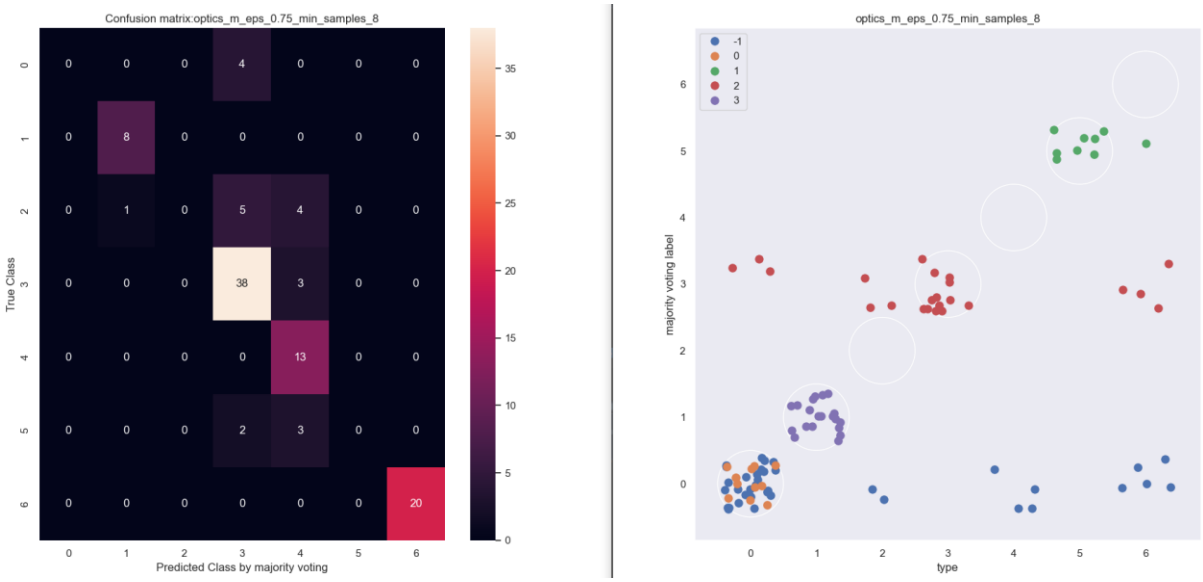
L'algoritmo clusterizza il dataset in 59 gruppi contro i 7 reali, se associassimo tramite il majority voting i gruppi questo è il gruppo che commette meno errori (nessuna etichetta errata). A livello di performance è tra quelli con performance intese come numero di cluster sul target "specie" peggiore. Osservando per esempio i cluster che sono associati tramite il majority voting ai pesci possiamo osservare che ha suddiviso la specie originale in tanti sotto cluster quante sono le possibili "varianti" sulle combinazioni di features.

animal name	hair	feathers	eggs	milk	airborne	aquatic	predator	toothed	backbone	breathes	venomou	fins	legs	tail	domestic	catsize	predi
bass	0	0	1	0	0	1	1	1	1	0	0	1	0	1	0	0	2
catfish	0	0	1	0	0	1	1	1	1	0	0	1	0	1	0	0	2
chub	0	0	1	0	0	1	1	1	1	0	0	1	0	1	0	0	2
herring	0	0	1	0	0	1	1	1	1	0	0	1	0	1	0	0	2
piranha	0	0	1	0	0	1	1	1	1	0	0	1	0	1	0	0	2
carp	0	0	1	0	0	1	0	1	1	0	0	1	0	1	1	0	5
dogfish	0	0	1	0	0	1	1	1	1	0	0	1	0	1	0	1	12
pike	0	0	1	0	0	1	1	1	1	0	0	1	0	1	0	1	12
tuna	0	0	1	0	0	1	1	1	1	0	0	1	0	1	0	1	12
haddock	0	0	1	0	0	1	0	1	1	0	0	1	0	1	0	0	24
seahorse	0	0	1	0	0	1	0	1	1	0	0	1	0	1	0	0	24
sole	0	0	1	0	0	1	0	1	1	0	0	1	0	1	0	0	24
stingray	0	0	1	0	0	1	1	1	1	0	1	1	0	1	0	1	51

Sfruttiamo tale algoritmo per fare una considerazione, negli algoritmi in cui l'iperparametro valutato era il numero di cluster da decidere a priori si sono valutati sempre cluster nell'intorno [4,10] limitando in parte il problema delle eccessive partizioni del dataset da parte dei vari modelli.

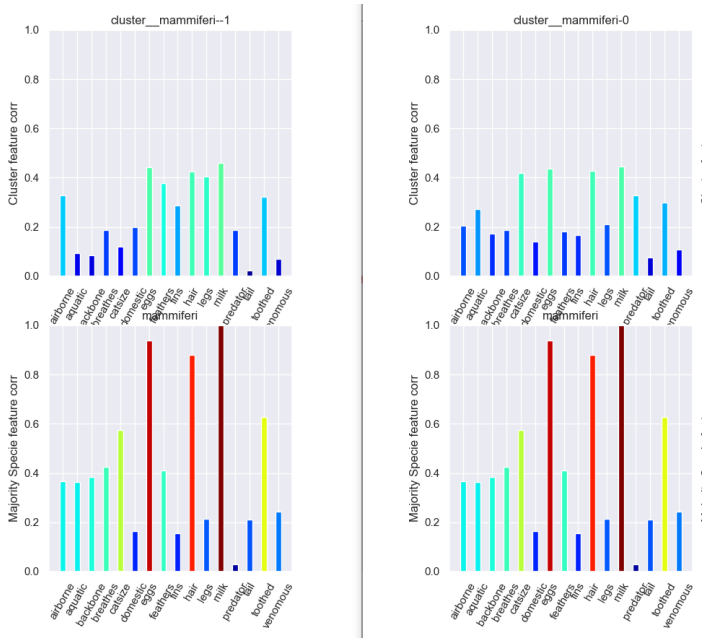
C'è da dire che quasi sempre essi hanno restituito un best model con $k \in (7,8)$ però se si riflette sulla dimensionalità del dataset 18 features per 7 cluster su 101 elementi (sbilanciato) è interessante notare come le i vari modelli abbiano partizionato sempre abbastanza bene il dataset.

OPTICS n=8



Possiamo osservare che l'algoritmo clusterizza il dataset in 5 gruppi contro i 7 reali, in particolare:

mammiferi (0): sono stati clusterizzati nei gruppi (-1) che però contiene anche campioni di altre specie ed il gruppo (0), più alcuni elementi che sono stati clusterizzari nel cluster 2 dei pesci (MV). Osservando le matrici di correlazione dei gruppi -1 e 0 possiamo notare come le feature a maggior correlazione con il cluster sono molto vicine della specie-cluster "reale" dei mammiferi, nel cluster (-1) possiamo notare una correlazione maggiore nella features "legs" dovuta alla presenza di elementi provenienti principalmente delle specie di anfibi ed invertebrati.

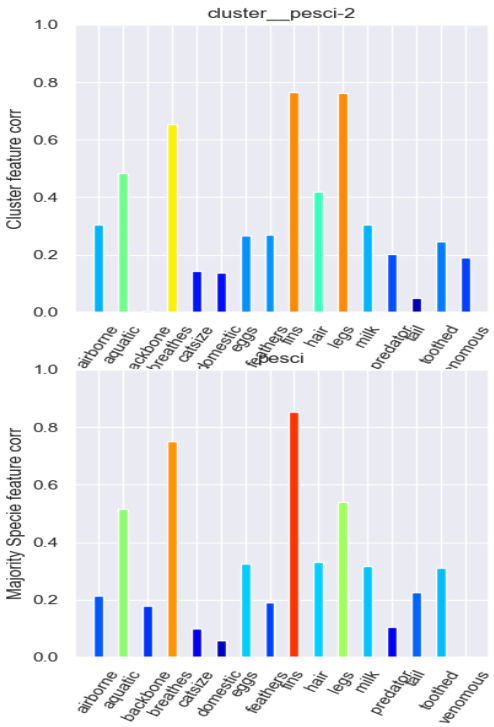


uccelli (1): sono stati clusterizzati in tutti in un unico gruppo (3)

rettili(2): sono stati clusterizzati in due gruppi, il gruppo (-1) dei "mammiferi" (MV) e il gruppo 2 dei pesci (MV-lab). Questo cluster è stato completamente assorbito dalle altre specie

- pesci(3):sono stati clusterizzati tutti nel cluster dei pesci (3) al cui interno troviamo anche elementi esterni (mammiferi, rettili ed invertebrati). Possiamo notare come le features "fins", "breathes" e

"acquatic" abbiano mantenuto una alta correlazione come nella specie originale ed invece siano cresciute le features "hair" e "legs" da associare al rumore introdotto dagli elementi esterni al cluster originale

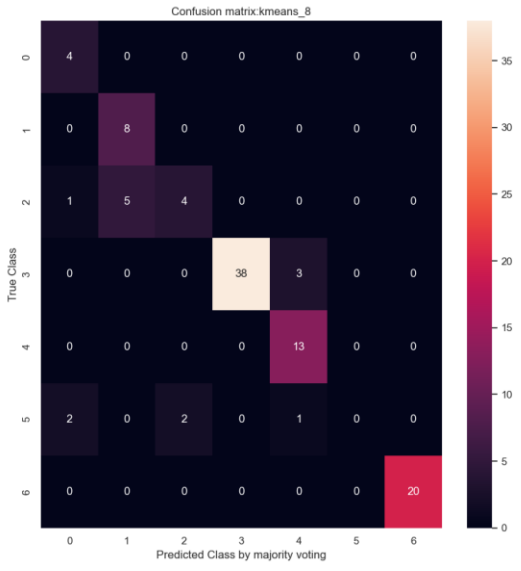


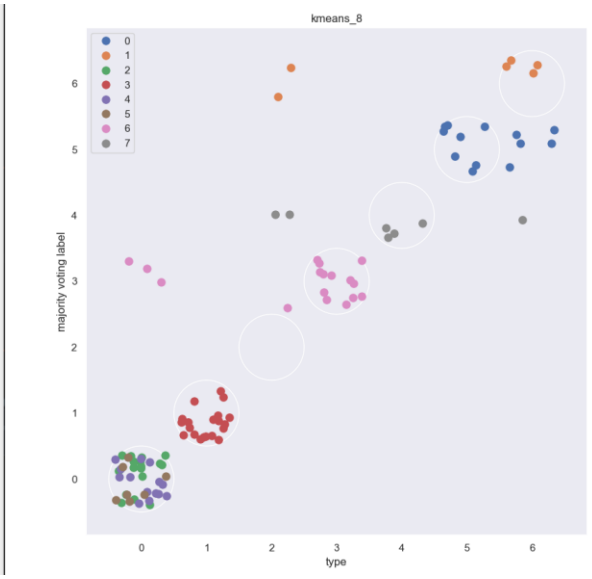
- anfibi (4): sono stati clusterizzati in tutti in un unico gruppo (-1) dei "mammiferi" (MV), come per i modelli precedenti tale specie forse anche dovuto all'esiguo numero di rappresentanti è stata completamente assorbita in cluster a maggioranza di oltra specie

- insetti(5): sono stati clusterizzati tutti in un unico gruppo (1) con un singolo elemento proveniente dalla classe invertebrati, per questo motivo la matrice di correlazione delle features è vicinissima a quella della specie originale

- invertebrati (6): sono stati clusterizzati in due gruppi (-1,2) dei pesci e dei mammiferi (MV) ,più un outsider finito negli "insetti".

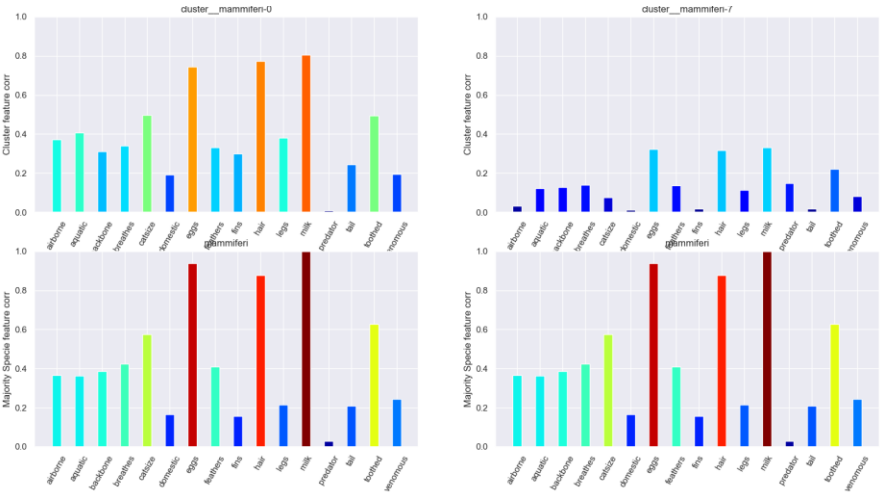
k-means algorithm k=8





Nell'insieme dei modelli allenati tramite k-mean, il modello che ha ottenuto sulla base dei nostri indici di valutazione le performance migliori è stato quello con l'iperparametro k numero cluster) pari a 8 contro i 7 reali, interessante è osservare che nonostante fosse stato imposto un cluster "in più" osservando i vari raggruppamenti tramite majority voting si ha che la specie dei rettili come cluster a se stante "scompare" in quanto viene completamente assorbita da altri cluster. In particolare:

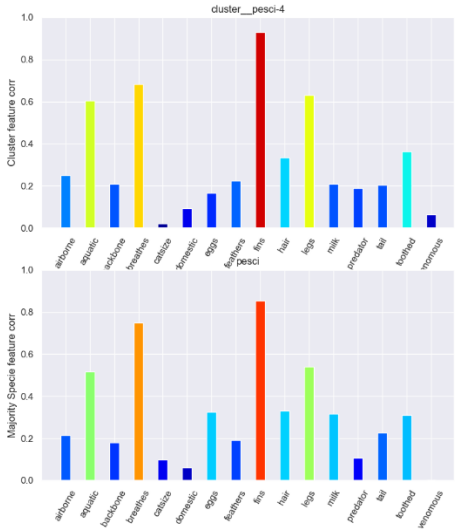
mammiferi (0): sono stati clusterizzati nei gruppi (0,7) con però alcuni elementi che sono stati clusterizzari nel cluster 3 dei pesci (majority voting label). Osservando le matrici di correlazione dei gruppi 0 e 7 possiamo notare come le feature a maggior correlazione con il cluster sono molto vicine della specie-cluster "reale" dei mammiferi, nel cluster (0) possiamo notare una correlazione maggiore nella features "catsize" rispetto al cluster 7 tale features potrebbe essere stata quella che ha maggiormente contribuito alla suddivisione in due sottocluster il cluster originale.



uccelli (1): sono stati clusterizzati in tutti in un unico gruppo (2)

rettili(2): sono stati clusterizzati in tre gruppi (1,4,5) venendo completamente assorbito dalle altre specie

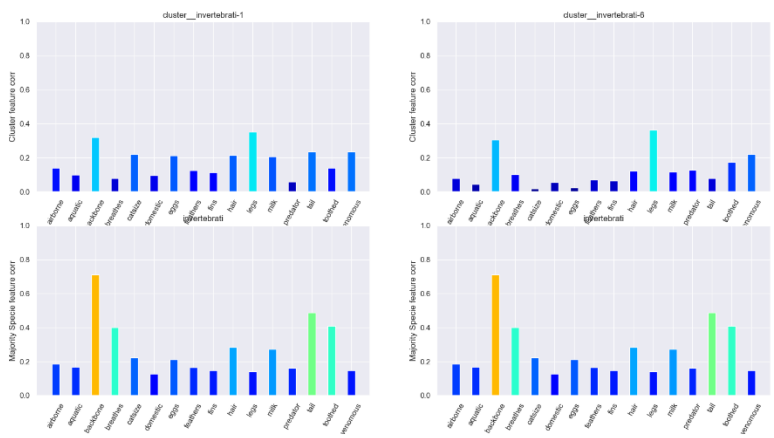
pesci(3): sono stati clusterizzati tutti nel cluster dei pesci (3) al cui interno troviamo anche elementi esterni (mammiferi e rettili). Possiamo notare come le features "fins", "breathes" e "acquatic" abbiano mantenuto una alta correlazione come nella specie originale ed invece siano cresciute le features "hair" e "eggs" da associare al rumore introdotto dagli elementi estreni quali rettili e mammiferi al cluster originale



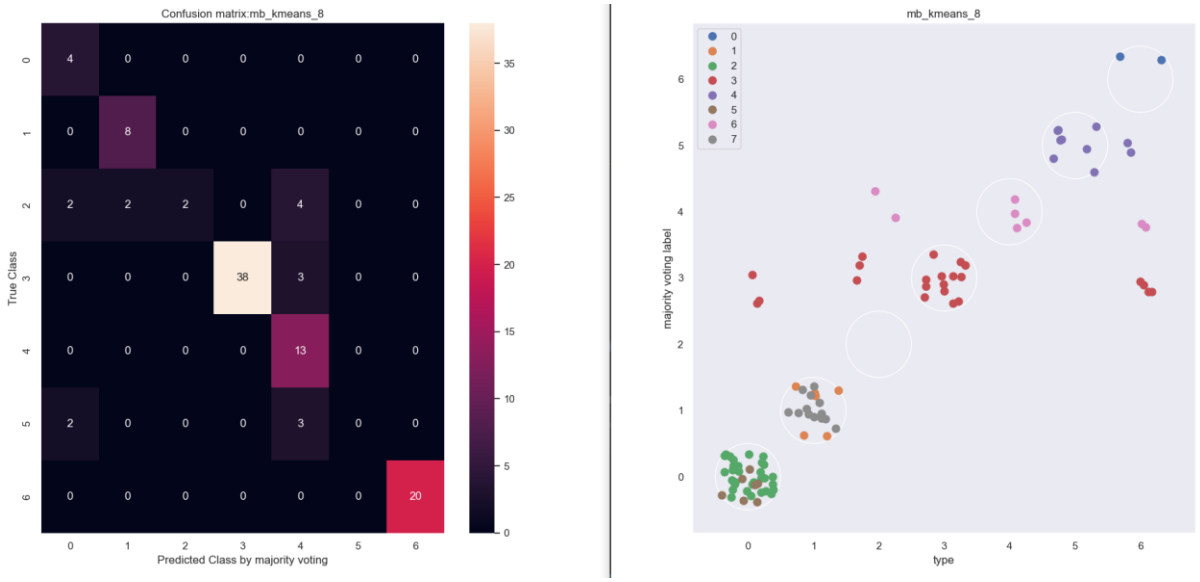
- anfibi (4): sono stati clusterizzati in tutti in un unico gruppo (5) degli "anfibi" (MV-lab),abbiamo al suo interno anche elementi provenienti dalle specie originali direttili e invertebrati, la matrice di correlazione che mantiene elementi a bassa correlazione con tutte le feature mantenendo un trend di valori simile all'originale.

- insetti(5): sono stati clusterizzati tutti in un unico gruppo (3) con alcuni elemento proveniente dalla classe invertebrati, per questo motivo la matrice di correlazione delle features è vicinissima a quella della specie originale tranne che per la feature "domestic" che è aumenta. tale aumento potrebbe essere dovuto all'errata clusterizzazione degli invertebrati come "honeybee" che essendo state "addomesticate" per il miele hanno aumentato la correlazione con tale feature rispetto al cluster

- invertebrati (6): sono stati clusterizzati in due gruppi (6) che contiene solo invertebrati e (1) che contiene al suo interno anche alcuni rettili, altri invertebrati sono stati classificati erroneamente tra insetti e anfibi. Osservando le correlazioni delle features con i cluster notiamo che la features "backbone" è rimasta caratteristica della specie, sono invece cresciute le features "legs" e "venomous" probabilmente dovute alla suddivisione in sue sotto cluster e alla presenza di due elementi esterni provenienti dalla classe rettili



Mini-Batch k-means algorithm k=8

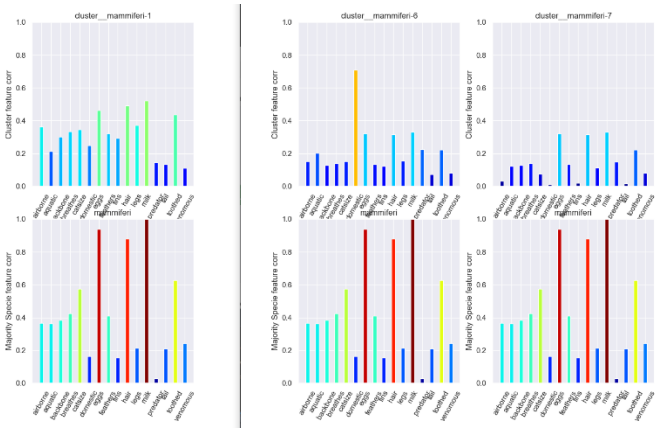


Nell'insieme dei modelli allenati tramite Mini-Batch k-means, il modello che ha ottenuto sulla base dei nostri indici di valutazione le performance migliori è stato quello con l'iperparametro k numero cluster) pari a 8 contro i 7 reali come k-mean, e proprio come il modello precedente interessante è osservare che nonostante fosse stato imposto un cluster "in più" osservando i vari raggruppamenti tramite majority voting si ha che le specie dei rettili e anfibi come cluster a se stanti "scompaiono" in quanto viengomo completamente assorbita da altri cluster (mammiferi e pesci).

In particolare:

- mammiferi (0): sono stati clusterizzati nei gruppi (6,7) di soli mammiferi, nel cluster (1) che presenta alcuni elementi provenienti da rettili e dalla totalità degli anfibi. In più alcuni elementi sono stati clusterizzari nel cluster 3 dei pesci (MV).

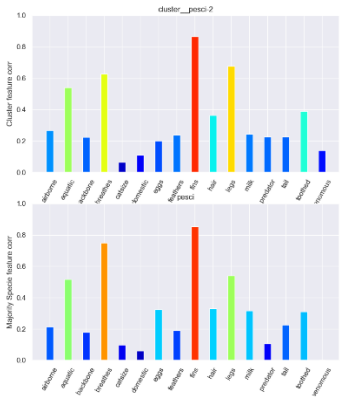
Osservando le matrici di correlazione dei gruppi (1,6,7) possiamo notare come le feature a maggior correlazione con il cluster sono molto vicine della specie-cluster "reale" dei mammiferi, nel cluster (6) possimao notare una correlazione maggiore nella features "domestic" mentre nel cluster (7) tale feature possiamo a correlazione bassissima .



- uccelli (1): sono stati clusterizzati in tutti in un unico gruppo (0)

- rettili(2): sono stati clusterizzati in tre gruppi (1 e 2) di mammiferi e uccelli venendo completamente assorbito dalle altre specie

- pesci(3):sono stati clusterizzati tutti nel cluster dei pesci (2) al cui interno troviamo anche elementi esterni (mammiferi e rettili). Possiamo notare come le features "fins", "breathes" e "acquatic" abbiano mantenuto una alta correlazione come nella specie originale ed invece siano cresciute le features "toothed" e "eggs" da associare al rumore introdotto dagli lementi estreni quali rettili e mammiferi al cluster originale

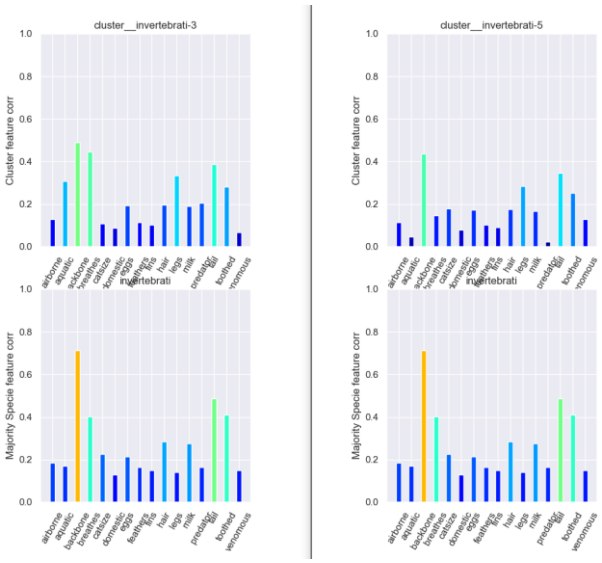


- anfibi (4): sono stati clusterizzati tutti nel gruppo (1) dei mammiferi venendo completamente "assorbiti" come cluster a voto maggioritario

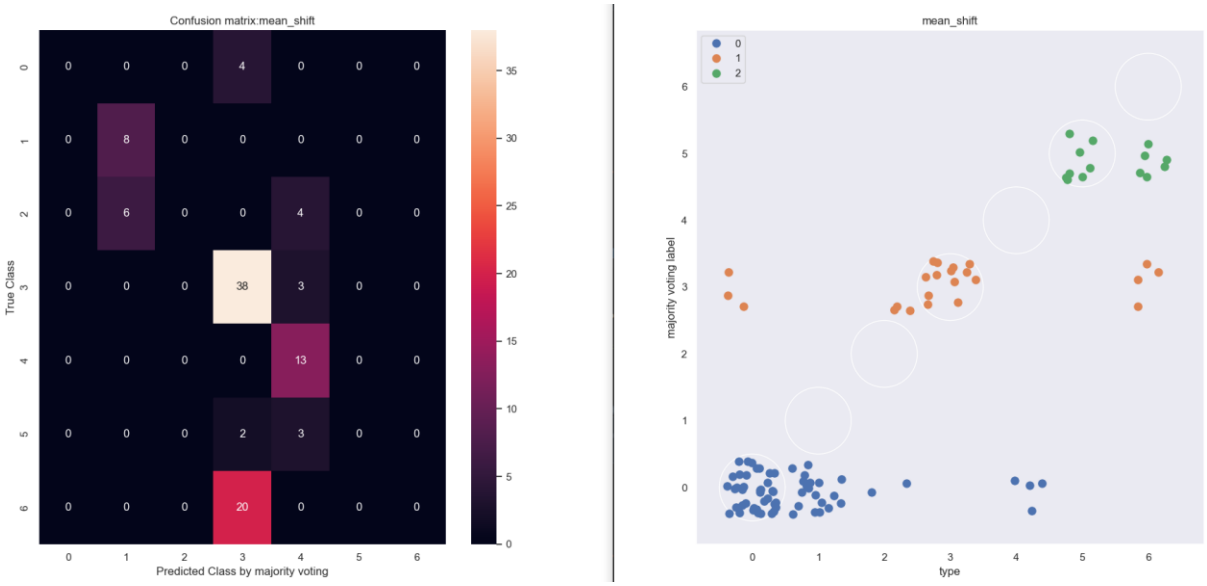
- insetti(5): sono stati clusterizzati tutti nel gruppo (4) degli insetti con un elemento proveniente dalla classe invertebrati, per questo motivo la matrice di correlazione delle features è vicinissima a quella della specie originale.

- invertebrati (6): sono stati clusterizzati in due gruppi (3,5)

Osservando le correlazioni delle features con i cluster notiamo che la features "backbone" è rimasta caratteristica della specie, si due sottogruppi hanno correlazioni differenti con la feature "predator" che potrebbe essere elemento di distinzione tra i due cluster

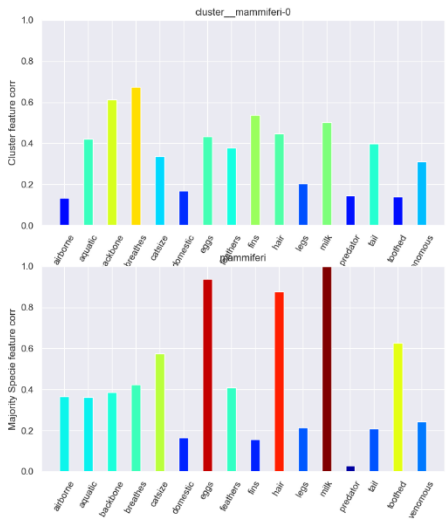


Mean shift algorithm

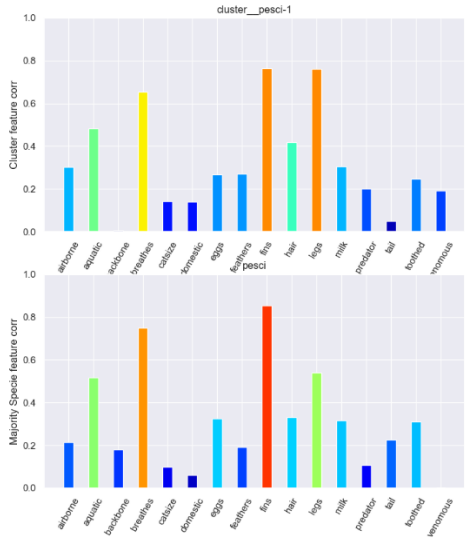


Possiamo osservare che l'algoritmo clusterizza il dataset in 3 gruppi contro i 7 reali, in particolare:

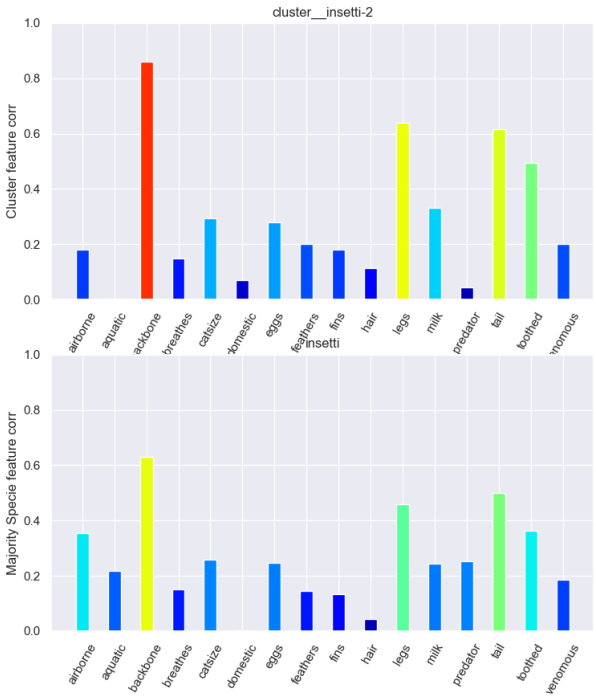
mammiferi (0): sono stati clusterizzati nei gruppi (0) con qualche elemento clusterizzato nel gruppo dei pesci, tale cluster però contiene anche la totalità degli uccelli e degli anfibi Osservando le matrici di correlazione del gruppo 0 possimao notare come le feature a maggior correlazione con il cluster dei mammimerisi sono ancora marcate ma sono presenti le feature "acquatic", "backbone" e "breathes" dovute agli elementi esterni che potrebbero far descrivere più questa specie animali tutto ciò che non è un pesce o un insetto/invertebrato.



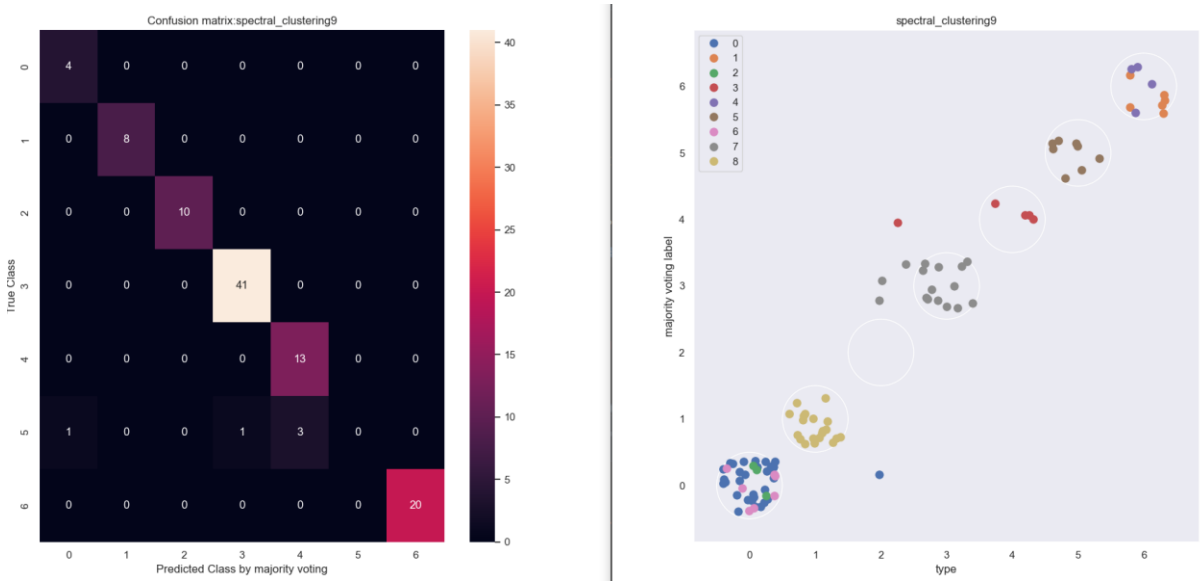
pesci(3):sono stati clusterizzati tutti nel cluster dei pesci (1) al cui interno troviamo anche elementi esterni (mammiferi, rettili ed invertebrati). Possiamo notare come le features "fins", "breathes" e "acquatic" abbiano mantenuto una alta correlazione come nella specie originale ed invece siano cresciute le features "hair" e "legs" da associare al rumore introdotto dagli elementi esterni al cluster originale



- insetti(5): sono stati clusterizzati tutti in un unico gruppo (2) insieme a parte degli invertebrati, per questo motivo la matrice di correlazione delle features è vicinissima a quella della specie originale accentuando ancora di più la correlazione con "legs", "backbone" e "tail"

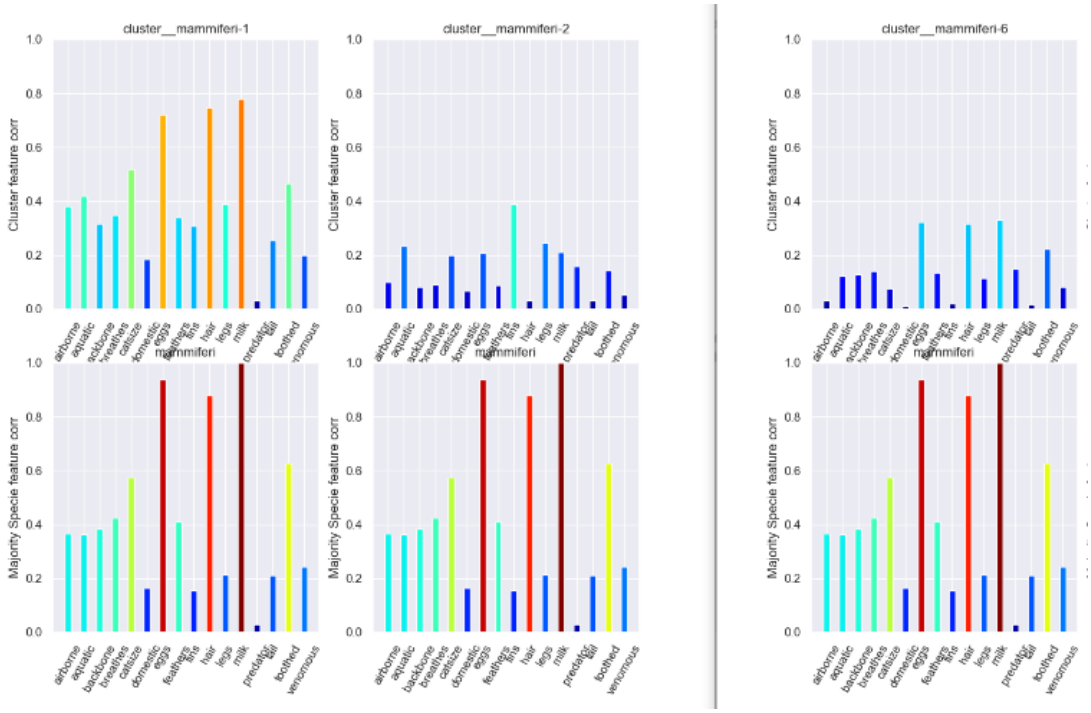


Spectral Clustering k=9



Nell'insieme dei modeli allenati il modello che ha ottenuto sulla base dei nostri indici di valutazione le performance migliori è stato quello con l'iperparametro k (numero cluster) pari a 9 contro i 7 reali, anche in questo caso nonostante le due classi in più osservando i vari raggruppamenti tramite majority voting si ha che la specie dei rettili come cluster a se stante "scompare" in quanto viene completamente assorbita da altri cluster. In particolare:

mammiferi (0): sono stati clusterizzati nei gruppi (1,2,6) con un elemento proveniente dai rettili. Osservando le matrici di correlazione del gruppo 1 possiamo notare come le feature a maggior correlazione con il cluster sono molto vicine della specie-cluster "reale" dei mammiferi, nei cluster (2 e 6) possiamo notare un profilo di correlazione simili alla specie originale ma meno marcato. Tra questui due gruppi notiamo come nel gruppo 6 la correlazione con la feature domestic sia a 0 a differenza del gruppo 2.



- uccelli (1): sono stati clusterizzati in tutti in un unico gruppo (2)

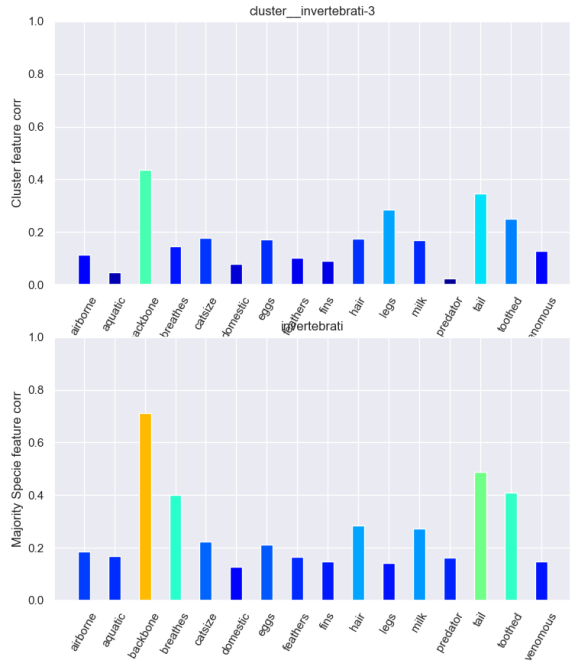
- rettili(2): sono stati clusterizzati in tre gruppi (1,5,7) venendo completamente assorbito principalmente dalla specie pesci (MV)

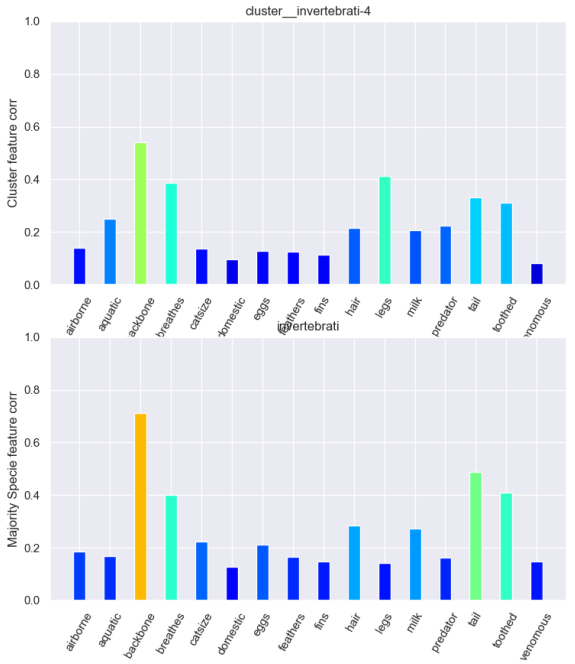
- pesci(3):sono stati clusterizzati tutti nel cluster dei pesci (7) al cui interno troviamo anche elementi esterni (rettili). Possiamo notare come le features il profilo del grafico a barre delle correlazioni tra le features e la classe è rimasto simile al profilo della specie originale.

- anfibi (4): sono stati clusterizzati in tutti in un unico gruppo (5) degli "anfibi" (MV-lab), abbiamo al suo interno anche un elemento provenienti dai rettili.

- insetti(5): sono stati clusterizzati tutti in un unico gruppo (0)

- invertebrati (6): sono stati clusterizzati in due gruppi (3,4), osservando le correlazioni delle features con i cluster notiamo che le features "predator" e "breathes" sono quelle che potrebbe distinguere di più i due sottocluster tra loro

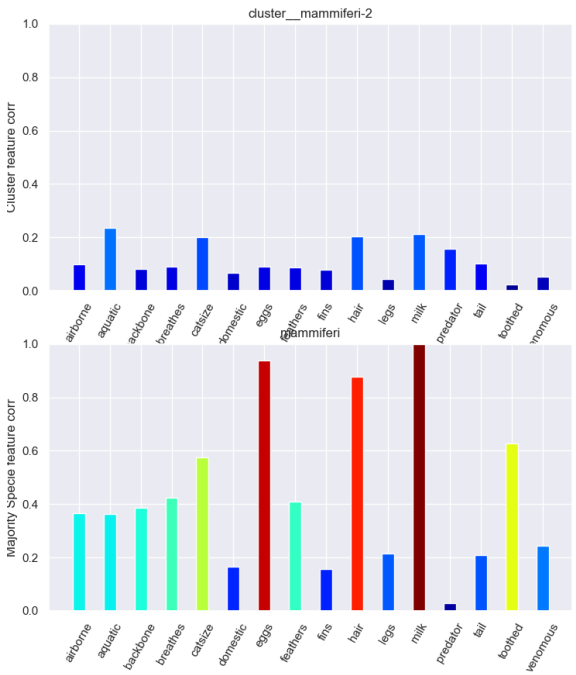


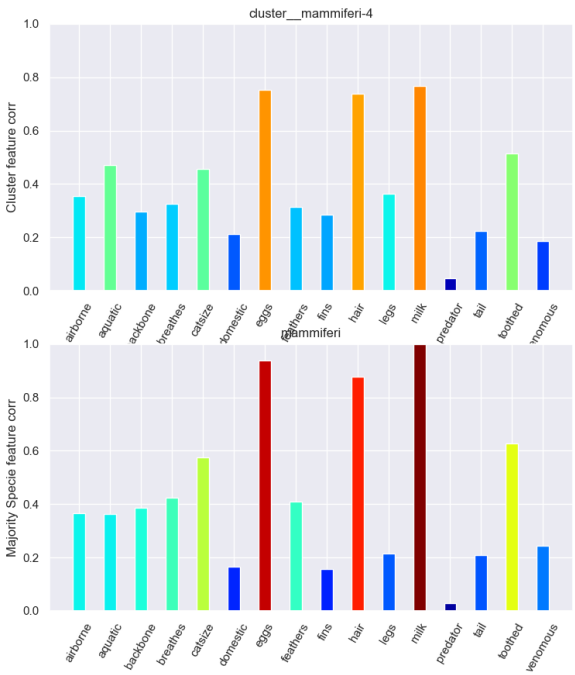


Gaussian Mixture k=7

Nell'insieme dei modeli allenati il modello che ha ottenuto sulla base dei nostri indici di valutazione le performance migliori è stato quello con l'iperparametro k (numero cluster) pari a 7, è il primo ed unico che ha suddiviso in un pari numero di cluster come quello del dataset originale i dati.Notiamo anche qui che tramite il majority voting si ha che le specie dei rettili e degli invertebrati come cluster a se stante "scompaiono" in quanto vengono completamente assorbita da altri cluster. In particolare:

mammiferi (0): sono stati clusterizzati nei gruppi (4,5) con alcuni elementi finiti nei cluster associato ai pesci. Osservando le matrici di correlazione del gruppo 4 possiamo notare come le feature a maggior correlazione con il cluster sono molto vicine della specie-cluster "reale" dei mammiferi, nei cluster (2) possiamo notare un profilo di correlazione non diverso dalla specie originale ma meno marcato.

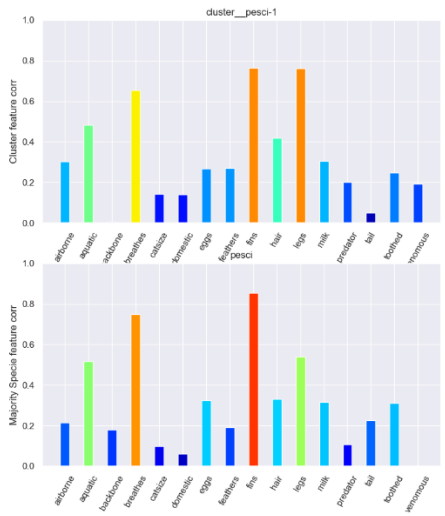




- uccelli (1): sono stati clusterizzati in tutti in un unico gruppo (3)

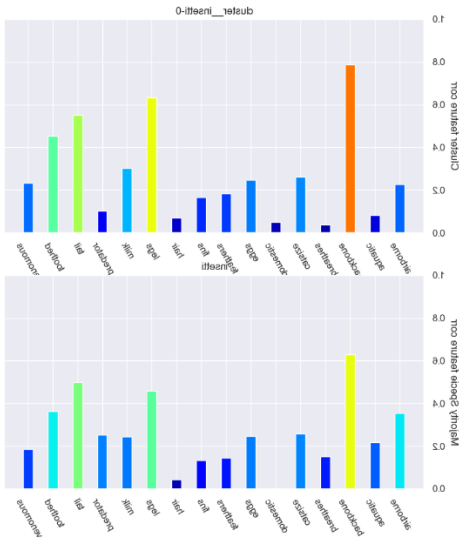
- rettili(2): sono stati clusterizzati in tre gruppi (1,6) venendo completamente assorbito principalmente dalla specie pesci e poi dagli anfibi(MV)

- pesci(3):sono stati clusterizzati tutti nel cluster dei pesci (1) al cui interno troviamo anche elementi esterni (rettili, anfibi, invertenrati). Possiamo notare come le features il profilo del grafico a barre delle correlazioni tra le features e la classe è rimasto simile al profilo della specie originale aumentando per quanto riguarda "acquatic" e "venomous" dovuto all'influenza di elementi esterni alla specie originale.



- anfibi (4): sono stati clusterizzati in tutti in un unico gruppo (6) degli "anfibi" (MV-lab),abbiamo al suo interno anche un elemento provenienti dai rettili ed invertebrati.

- insetti(5): sono stati clusterizzati tutti in un unico gruppo (0) che contiene al suo interno anche diversi invertebrati mantenendo però lo stesso profilo del grafico a barre tra correlazione tra features e cluster

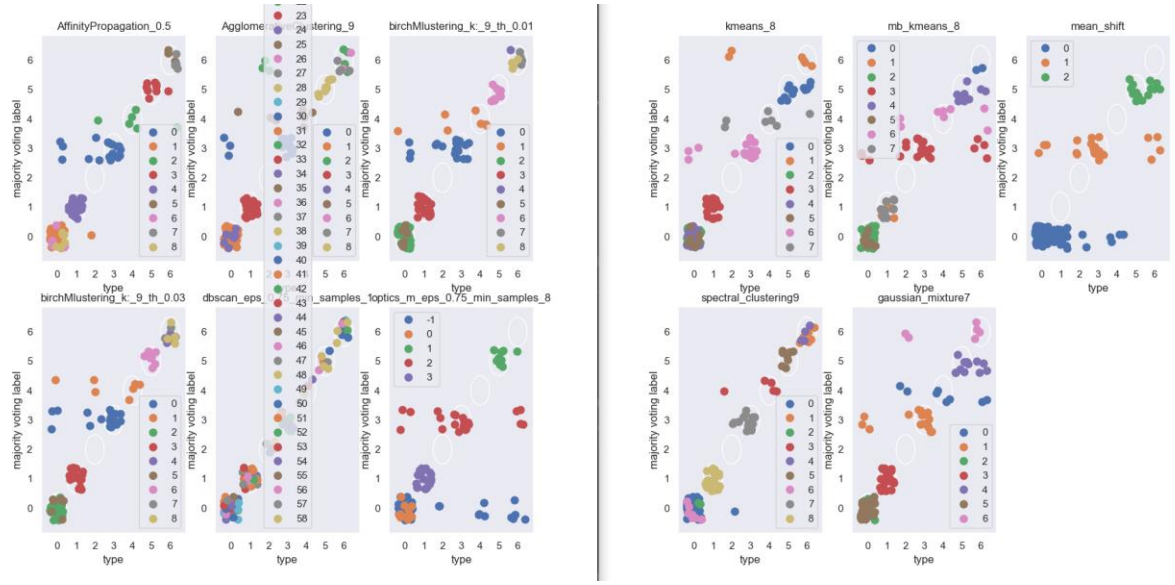


- invertebrati (6): sono stati clusterizzati nel due gruppi (0,1,6) venendo completamente assorbiti dagli altri cluster

Final Consideration

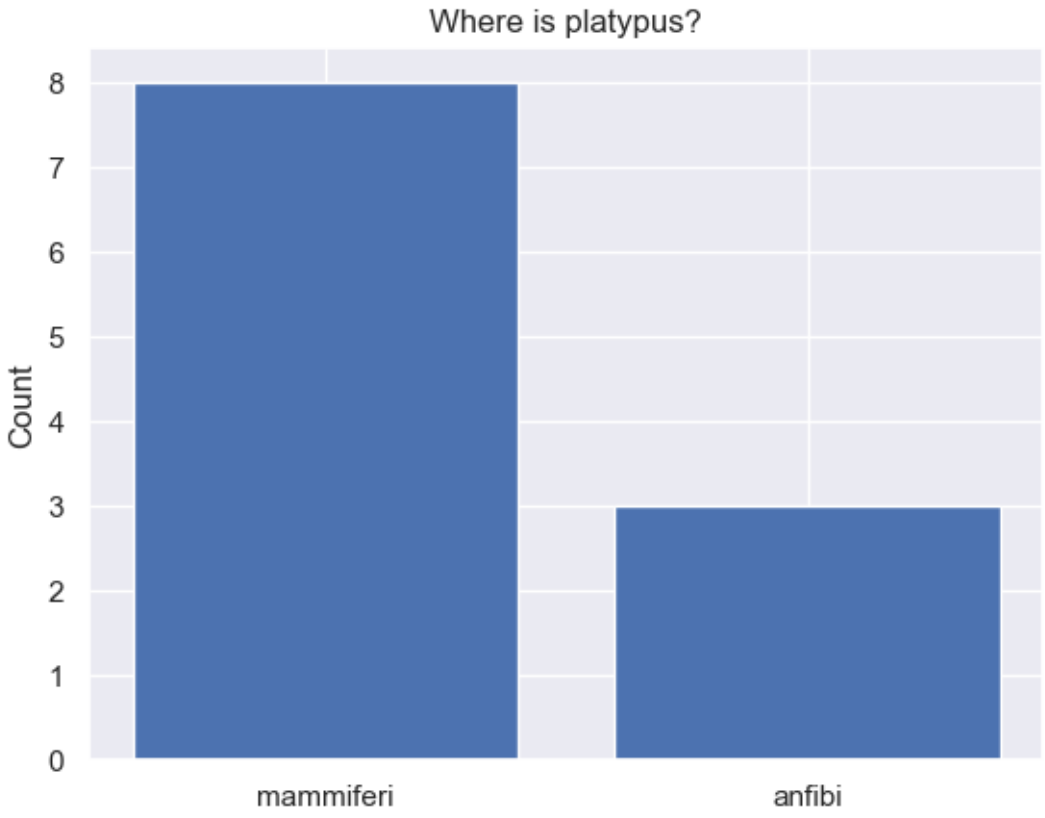
Tra i migliori rappresentanti dei vari algoritmi di clustering vogliamo provare ad identificare il modello più performante. Se ci basiamo sull'indice di performance utilizzato per selezionare il miglio rappresentante di ogni algoritmo il migliore è lo spectral_clustering con k=9. Esso suddivide il dataset in 9 gruppi in cui anche osservando lo scatter plo si osserva che la specie dei rettili è l'unica non rappresentanta e "assorbita" dagli altri cluster mentre le altri specie sono tutte raggruppate in un gruppo unico o in sotto gruppi (mammiferi e invertebrati) che li differenziano al loro interno senza però clusterizzare un elemento di una specie al di fuori della specie "rettili" in cluster con elementi di altre specie.

Se ci basassimo sull'algoritmo che è andato più "vicino" alla clusterizzazione originale intesa come numero di cluster e appartenenza di ogni elemento al cluster più rappresentativo di quella specie (MV) il più adatto lo scegliamo confrontando gli scatterplot dei vari "best model" il spectral_clustering con k=9 risulta ancora il più performante. Infatti per esempio DBSCAN dentro ogni gruppo ha elementi di una sola specie, però suddivide il dataset in troppi sottogruppi, è anche l'unico a non far "assorbire" gli elementi appartenenti alla specie rettili da gruppi di altre specie (tramite MV). I restanti metodi suddividono in più cluster il dataset e associano a cluster meno omogenei alcuni elementi (elementi appartenenti alla specie mammiferi si trovano nello stesso cluster degli anfibi, ecc)



Where is platypus?

Per concludere vogliamo rispondere alla domanda "I mammiferi fanno le uova?" valutando come l'ornitorinco (mammifero che fa le uova) sia stato clusterizzato nei vari modelli tramite il metodo del majority voting



Osserviamo che su 11 best model 8 volte è stato clusterizzato correttamente nonostante sia velenoso, faccia le uova ed abbia il becco.