Multivariate Analysis

Clustering

In [345…
```
import pandas as pd
import seaborn as sns
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
```

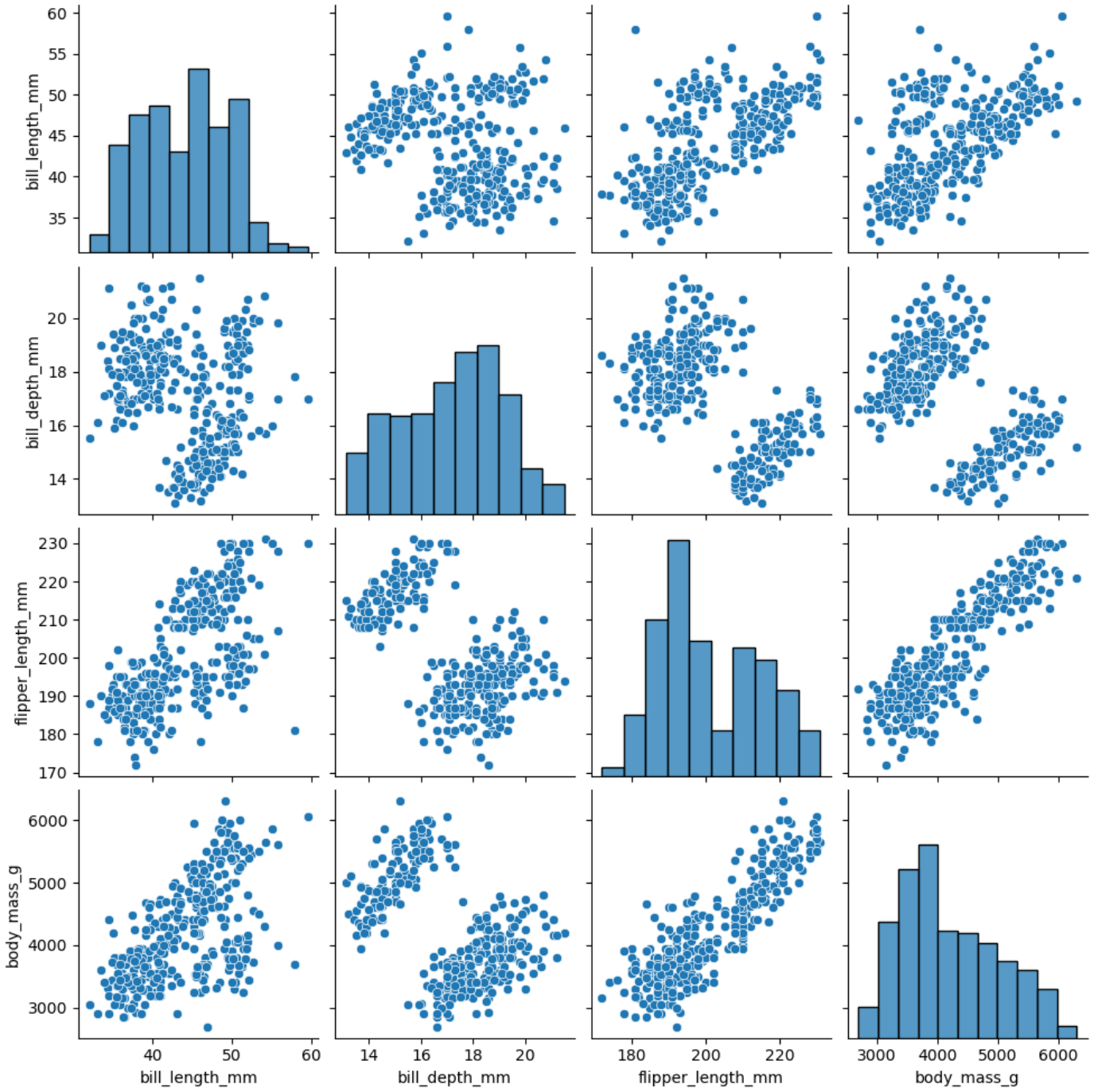
In [346…
```
from sklearn import metrics
from sklearn.metrics import pairwise_distances
```

In [347…
```
penguins = sns.load_dataset("penguins")
penguins = penguins.dropna()
penguins.head()
```

Out[347]:

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	Male
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	Female
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	Female
4	Adelie	Torgersen	36.7	19.3	193.0	3450.0	Female
5	Adelie	Torgersen	39.3	20.6	190.0	3650.0	Male

In [348…
```
sns.pairplot(penguins)
plt.show()
```

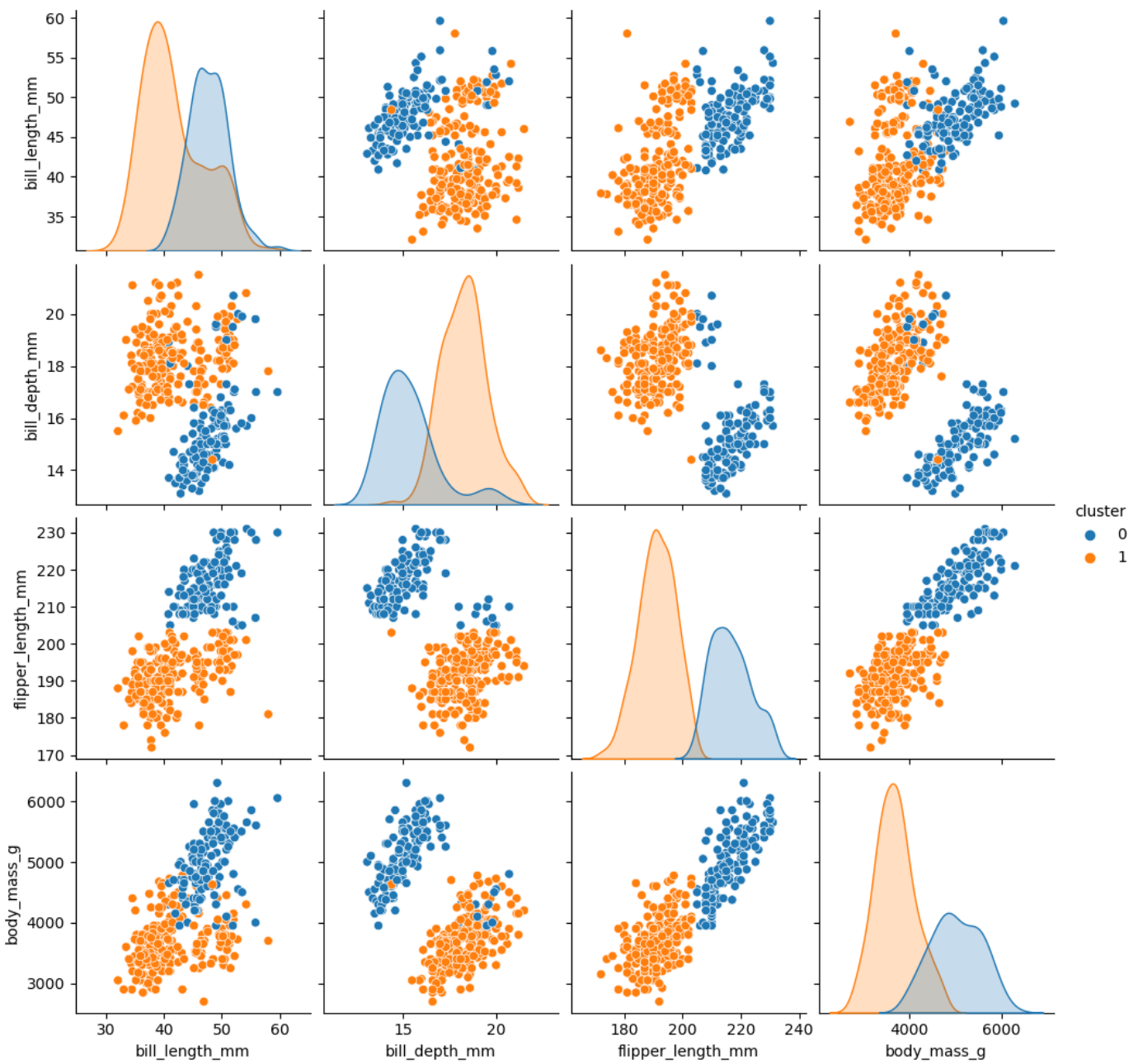


Kijkend naar dit overzicht lijken er 2 duidelijke clusters te zijn voor body mass en flipper length. Verder lijkt het ook dat er 3 clusters zijn tussen bijvoorbeeld body mass en bill length en flipper length en bill length.

In [349…
```
features = ['flipper_length_mm']
km = KMeans(n_clusters=2, random_state=43).fit(penguins[features])
```

In [350…
```
penguins['cluster'] = km.predict(penguins[features])
```

In [351…
```
sns.pairplot(penguins, hue="cluster")
plt.show()
```

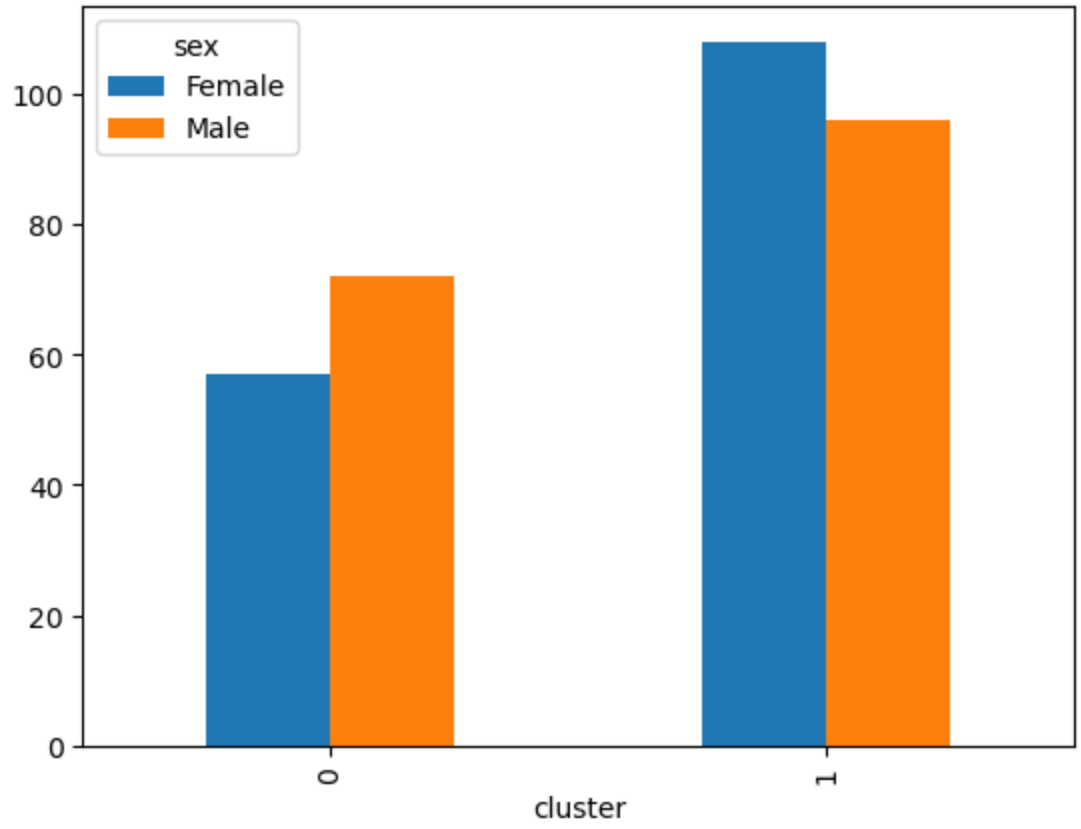


In [352…
```
metrics.silhouette_score(penguins[features], km.labels_, metric='euclidean')
```

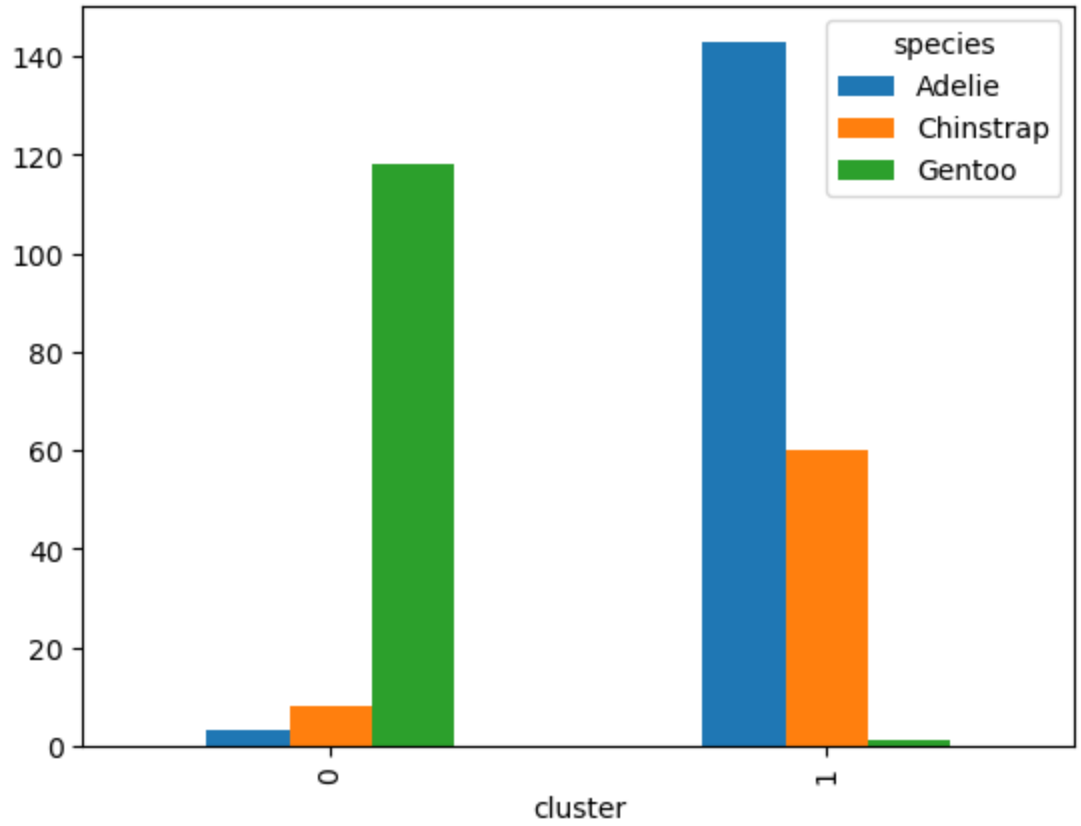
Out[352]:
```
0.6892852079650291
```

Na wat uitproberen blijkt dat twee clusters zoeken met flipper length de beste silhouette score geeft.

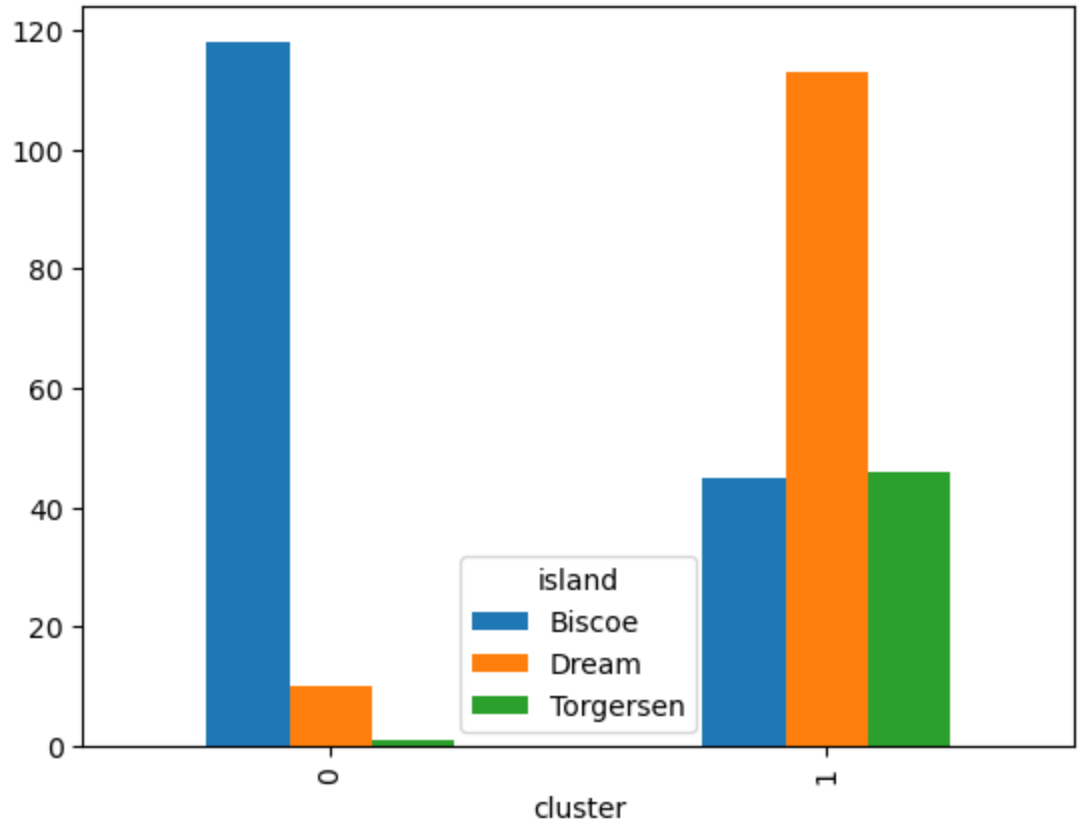
In [353…
```
contingency_table = penguins.groupby(['sex','cluster']).size().unstack('sex', fill_value
contingency_table.plot(kind='bar')
plt.show()
```



In [354…
```
contingency_table = penguins.groupby(['species','cluster']).size().unstack('species', fi
contingency_table.plot(kind='bar')
plt.show()
```

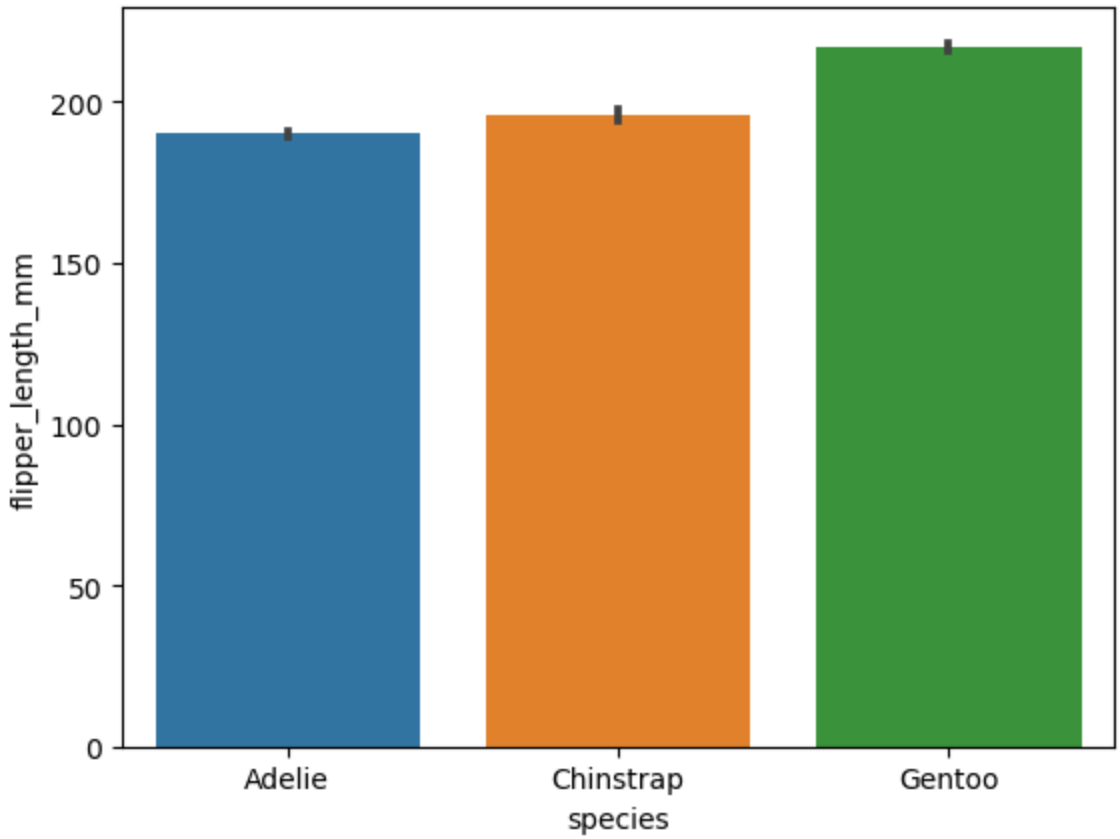


In [355…
```
contingency_table = penguins.groupby(['island','cluster']).size().unstack('island', fill
contingency_table.plot(kind='bar')
plt.show()
```

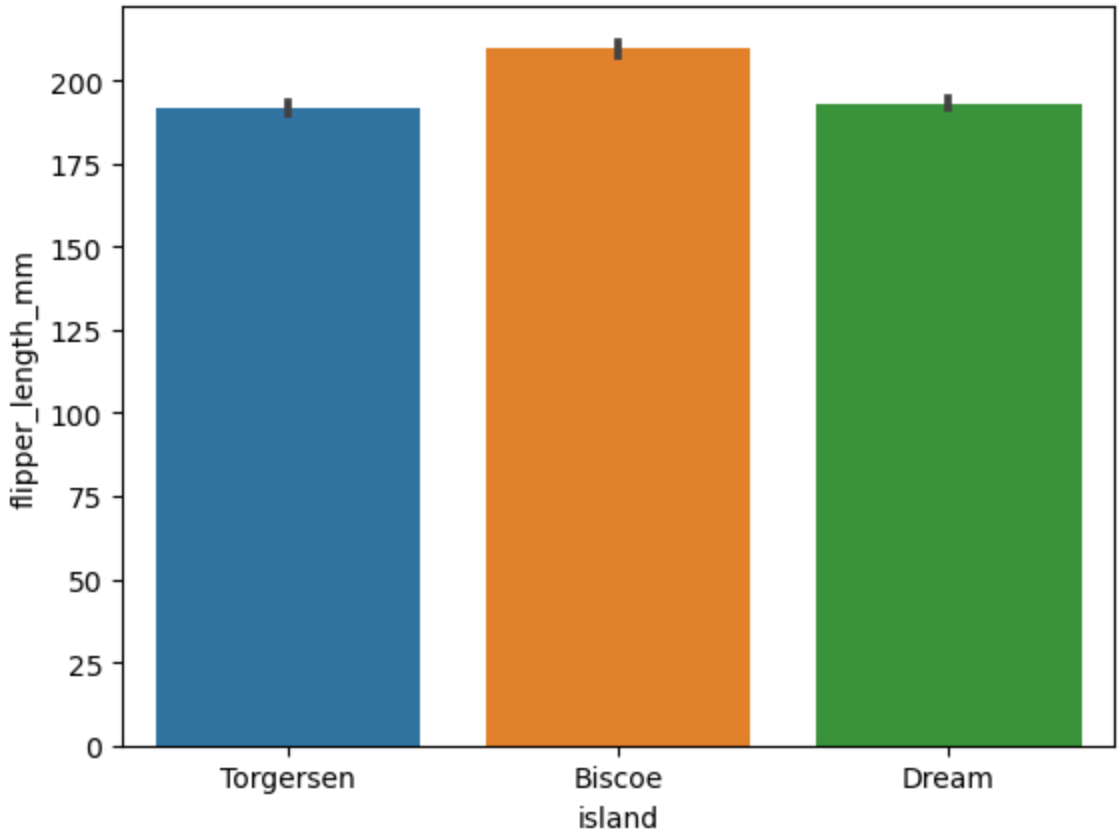


Van deze clusters zien we een scheiding tussen de species Gentoo en de andere species. Het verschil in flipper length tussen Adelie en Chinstrap net zoals Dream en Torgersen is dus blijkbaar niet zo groot.

In [356…
```
sns.barplot(y="flipper_length_mm", x='species', data=penguins)
plt.show()
```



In [357…
```
sns.barplot(y="flipper_length_mm", x='island', data=penguins)
plt.show()
```



Het is het wel waard om nog te kijken of er een combinatie is van features die ofwel island ofwel species beter kan voorspellen met drie clusters.

In [358…
```
features_3 = ['flipper_length_mm', 'bill_length_mm']
km_3 = KMeans(n_clusters=3, random_state=43).fit(penguins[features_3])
```

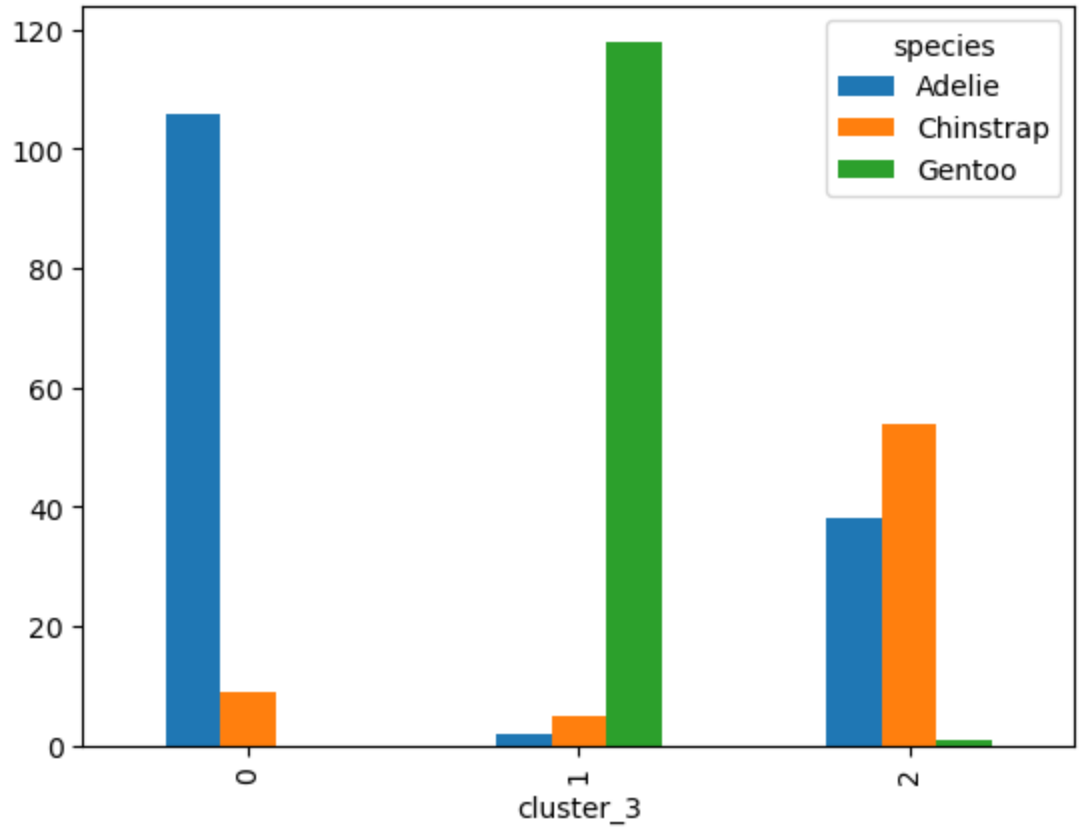
In [359…
```
penguins['cluster_3'] = km_3.predict(penguins[features_3])
```

In [360…
```
metrics.silhouette_score(penguins[features_3], km_3.labels_, metric='euclidean')
```

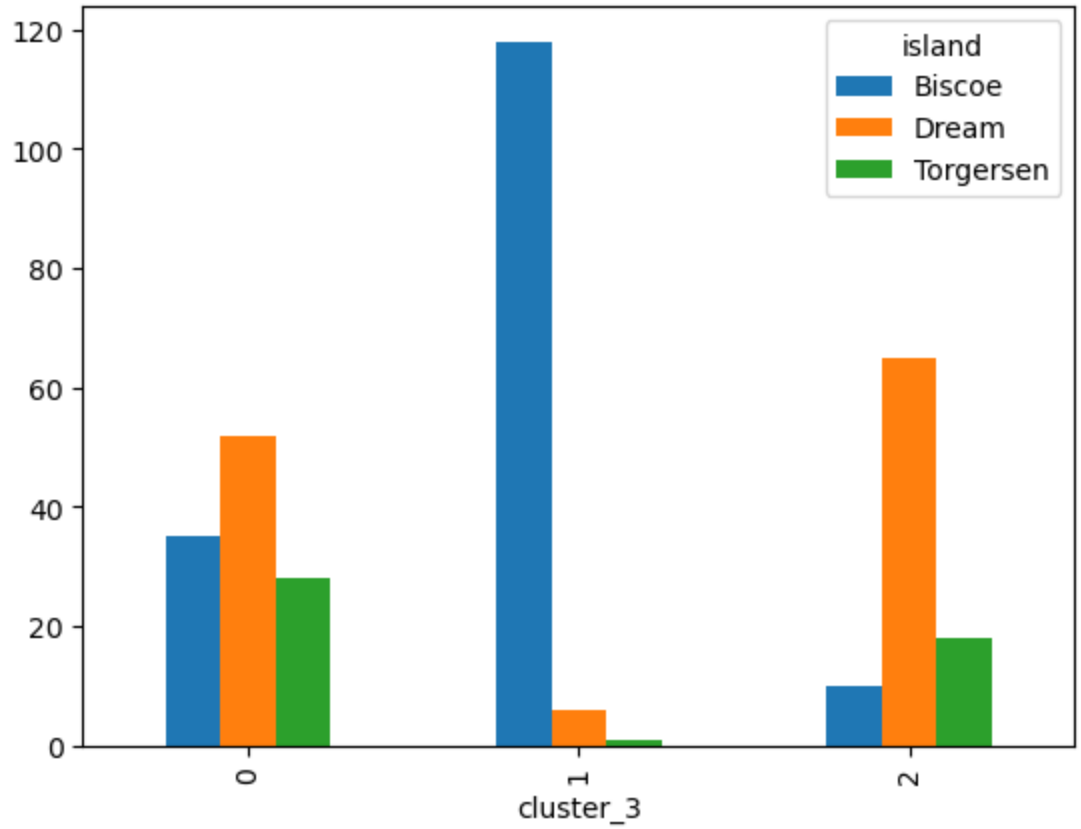
Out[360]:
```
0.48062618684371117
```

Je krijg met alleen massa de hoogste silhouette score. Als je echter gaat kijken naar hoe het over species en eilanden is verdeeld, zie je dat een combinatie van flipper lenght en bill length zorgt dat Adelie beter gescheiden is van Chinstrap.

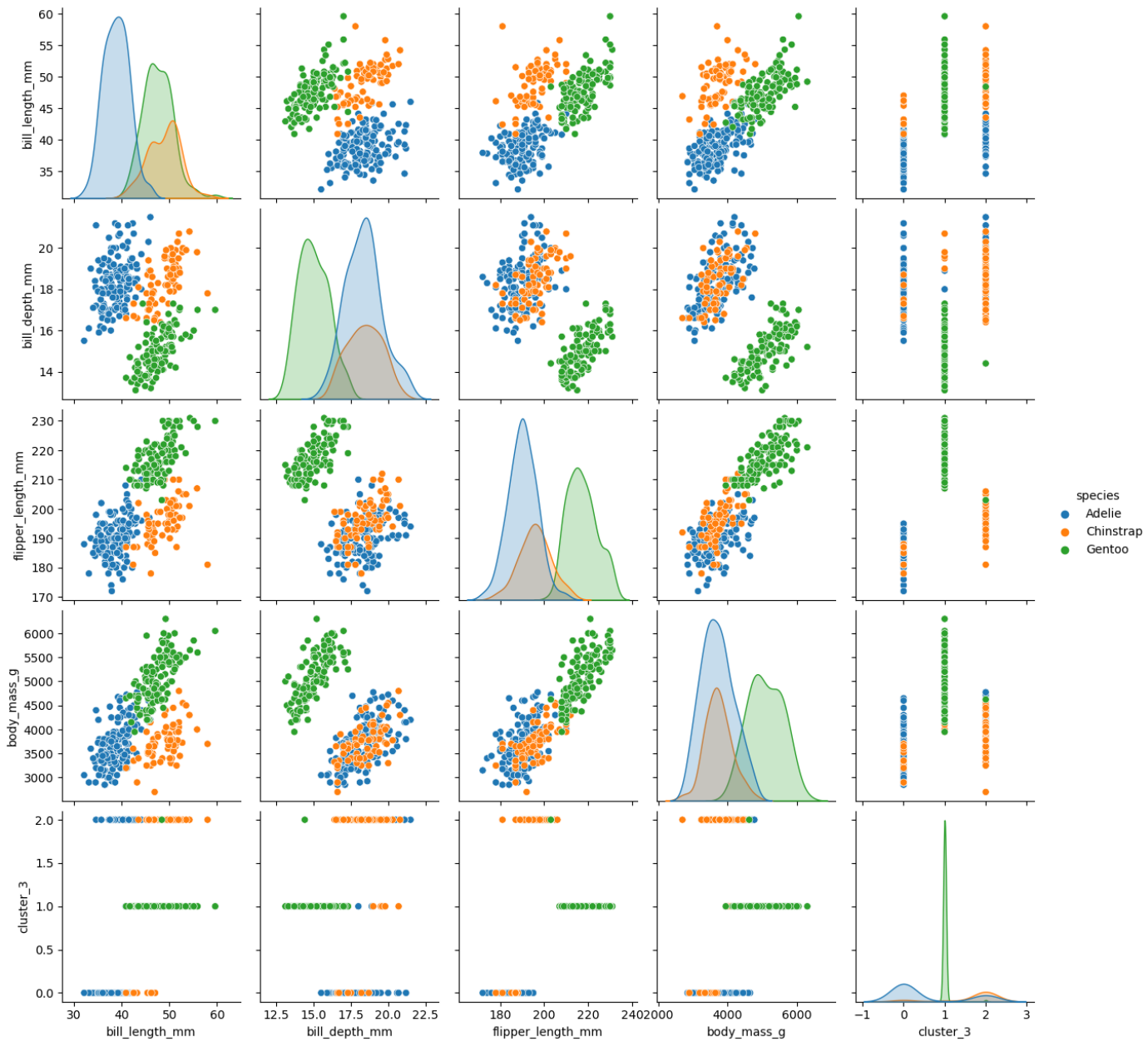
In [361…
```
contingency_table = penguins.groupby(['cluster_3', 'species']).size().unstack('species',
contingency_table.plot(kind='bar')
plt.show()
```



In [362…
```
contingency_table = penguins.groupby(['cluster_3','island']).size().unstack('island', fi
contingency_table.plot(kind='bar')
plt.show()
```



In [366…
```
sns.pairplot(penguins.drop(['cluster'], axis=1), hue="species")
plt.show()
```


bill_length_mm
bill_depth_mm
flipper_length_mm
body_mass_g
cluster_3
species
Adelie
Chinstrap
Gentoo