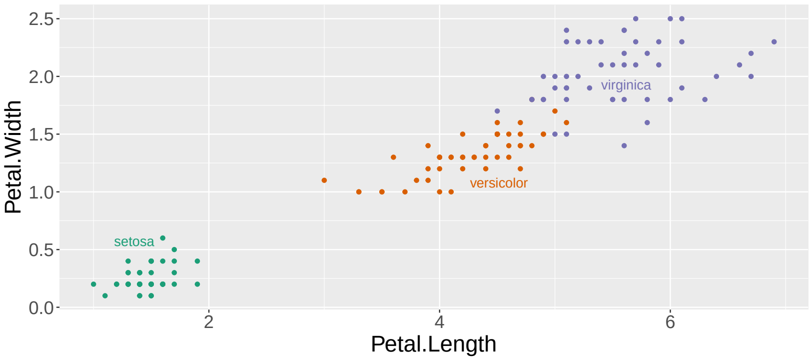
# Gaussian mixture models

Toby Dylan Hocking

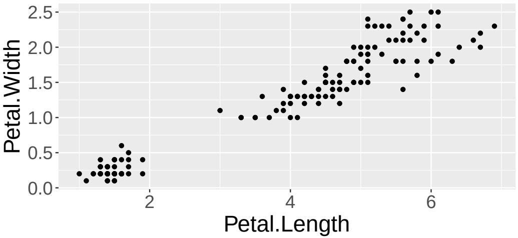
# Visualize iris data with labels



# Visualize iris data without labels

- Let $X \in \mathbb{R}^{n \times p}$ be the data matrix (input for clustering).
- Example iris $n = 150$ observations, $p = 2$ dimensions.

```
##      Petal.Width Petal.Length
## [1,]         0.2          1.4
## [2,]         0.2          1.4
## [3,]         0.2          1.3
## [4,]         0.2          1.5
```



# Gaussian mixture model parameters

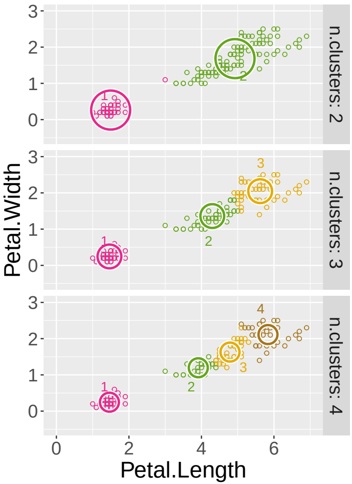
Need to fix number of clusters $K$, then for every $k \in \{1, \dots, K\}$ we have cluster-specific parameters $\theta_k = [\mu_k, S_k, \pi_k]$,

- mean vector $\mu_k \in \mathbb{R}^p$,
- covariance matrix $S_k \in \mathbb{R}^{p \times p}$, (must be symmetric, positive definite)
- prior weight $\pi_k \in [0, 1]$ (sum over all clusters $k$ must equal one).

There can be additional constraints on the covariance matrix (next slides).

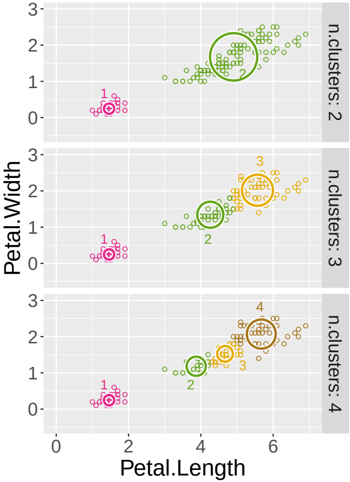
# spherical, equal volume

```
##             c1     c1     c2     c2     c3     c3
## width  0.1077 0.0000 0.1077 0.0000 0.1077 0.0000
## length 0.0000 0.1077 0.0000 0.1077 0.0000 0.1077
```



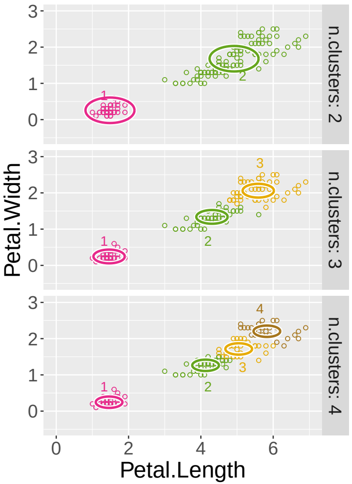
# spherical, unequal volume

```
##              c1     c1     c2     c2     c3     c3
## width  0.0202 0.0000 0.1298 0.0000 0.1837 0.0000
## length 0.0000 0.0202 0.0000 0.1298 0.0000 0.1837
```



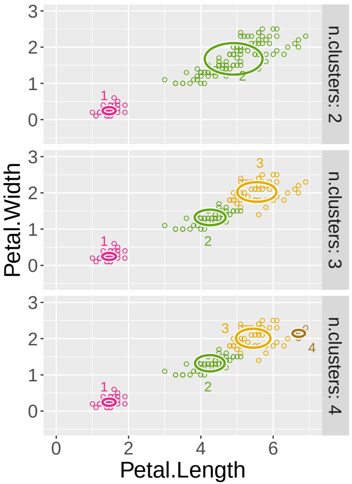
# diagonal, equal volume and shape

```
##              c1     c1    c2     c2    c3     c3
## width  0.036 0.0000 0.036 0.0000 0.036 0.0000
## length 0.000 0.1878 0.000 0.1878 0.000 0.1878
```



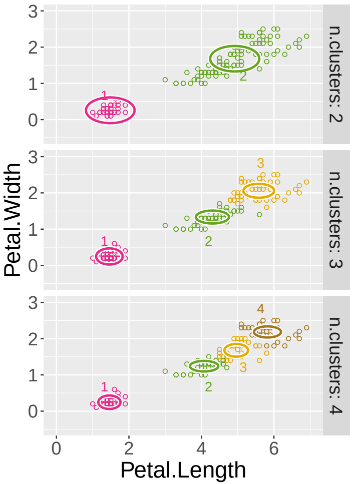
# diagonal, varying volume, equal shape

```
##            c1     c1     c2     c2     c3     c3
## width  0.0091 0.0000 0.0457 0.0000 0.0732 0.0000
## length 0.0000 0.0367 0.0000 0.1837 0.0000 0.2944
```



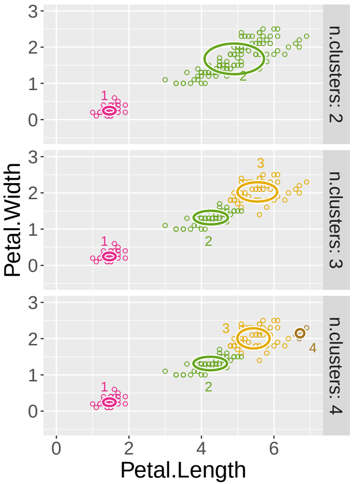
# diagonal, equal volume, varying shape

```
##                 c1     c1     c2     c2     c3     c3
## width  0.0494 0.0000 0.0317 0.0000 0.0368 0.0000
## length 0.0000 0.1341 0.0000 0.2089 0.0000 0.1802
```



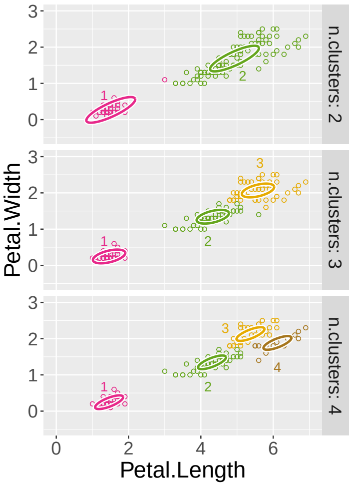
# diagonal, varying volume and shape

```
##              c1     c1     c2     c2     c3     c3
## width  0.0109 0.0000 0.0352 0.0000 0.0709 0.0000
## length 0.0000 0.0296 0.0000 0.2243 0.0000 0.3008
```



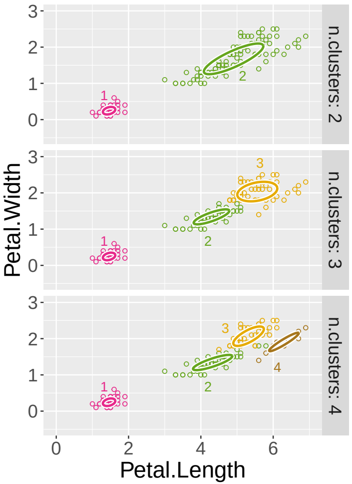
# ellipsoidal, equal volume, shape, and orientation

```
##               c1     c1     c2     c2     c3     c3
## width  0.0358 0.0425 0.0358 0.0425 0.0358 0.0425
## length 0.0425 0.2005 0.0425 0.2005 0.0425 0.2005
```

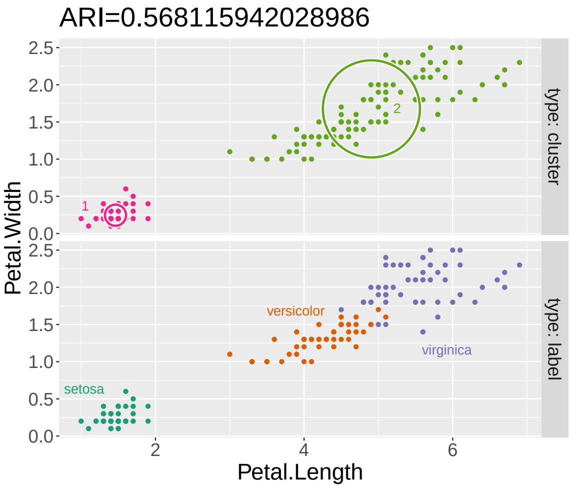


# ellipsoidal, varying volume, shape, and orientation

```
##              c1     c1     c2     c2     c3     c3
## width  0.0109 0.0059 0.0428 0.0813 0.0727 0.0482
## length 0.0059 0.0296 0.0813 0.2438 0.0482 0.3065
```

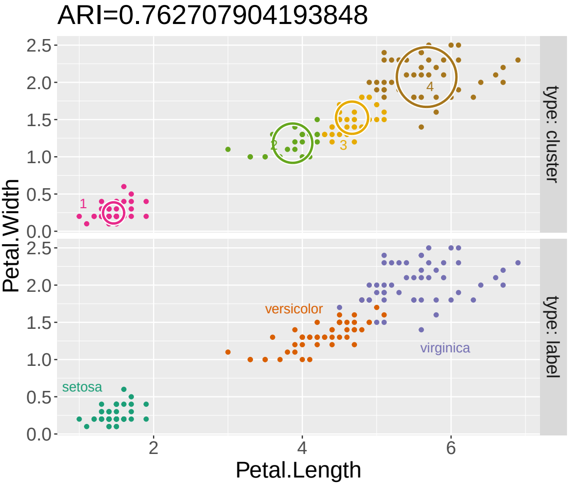


# Compare two clusters to labels

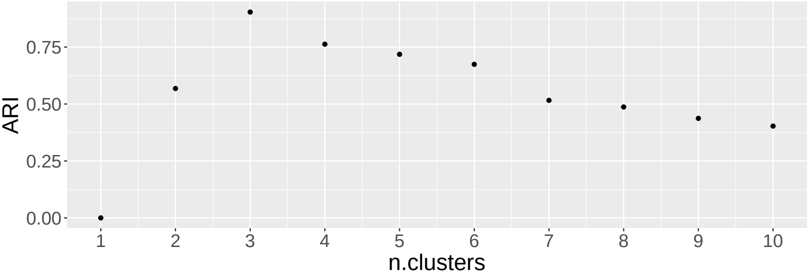


# Compare three clusters to labels

# Compare four clusters to labels

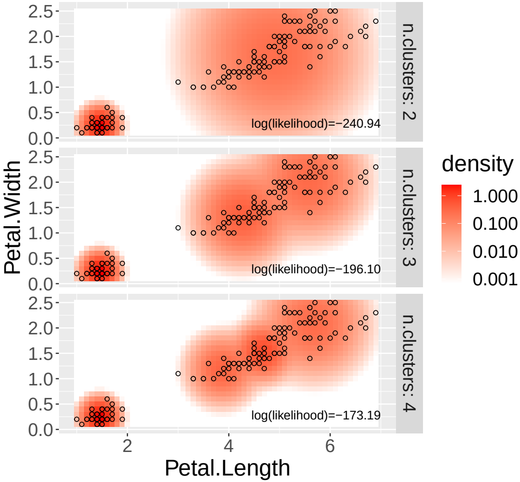


# Compute ARI for several clusterings



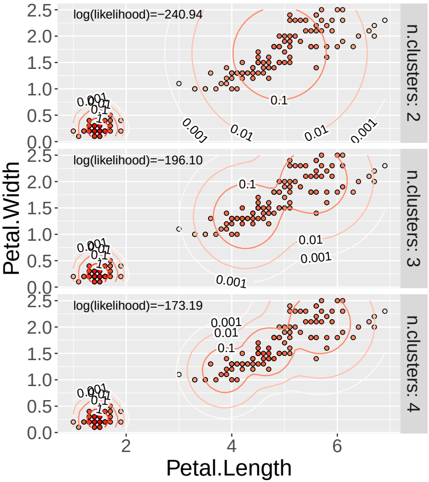

- Which K is best? Clear peak at 3 clusters, which makes sense since there are three species in these data.

# Visualization of log likelihood

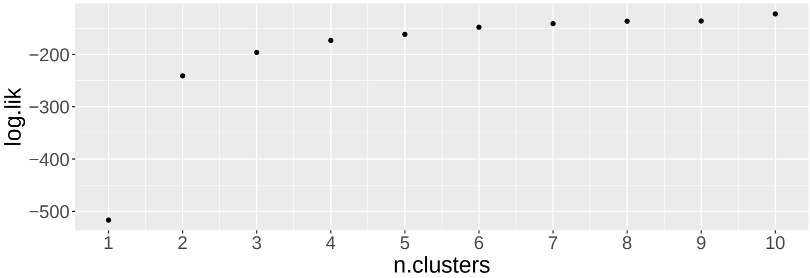



- Darker red means larger density value from learned model.
- The total redness in the data points represents the log likelihood, which is what the EM algorithm attempts to maximize.

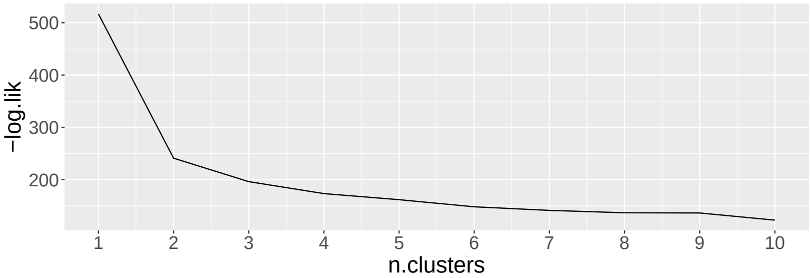
# Visualize density using level curves



# Compute log likelihood for several clusterings

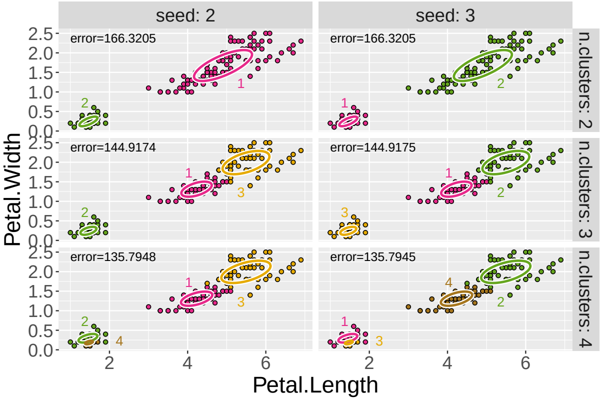


# Model selection via error curve analysis (negative log likelihood)



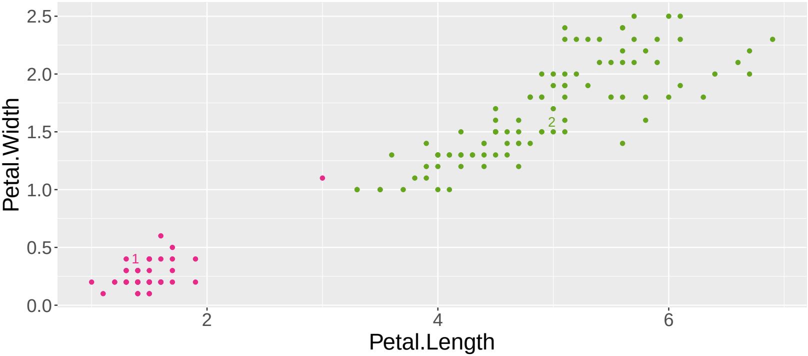

- These error values can be computed using only the input data (labels/outputs are not required).
- In general, for any problem/data set, making this plot and then locating the "kink in the curve" is a good rule of thumb for selecting the number of clusters.

# Visualize clusters using two random seeds

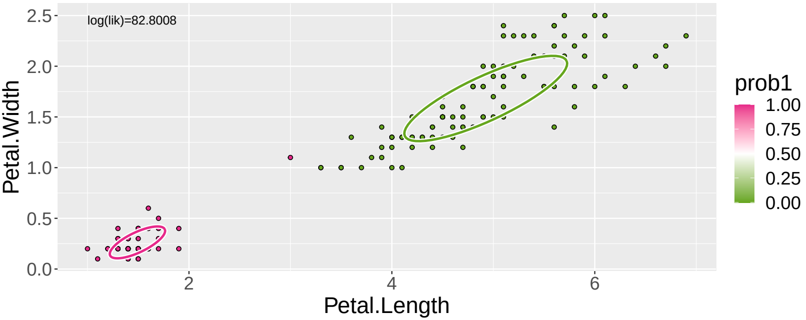



- Different seeds used for initial assignment based on K-means.
- EM solution quality depends on random seed (not much variation in these simple data though).

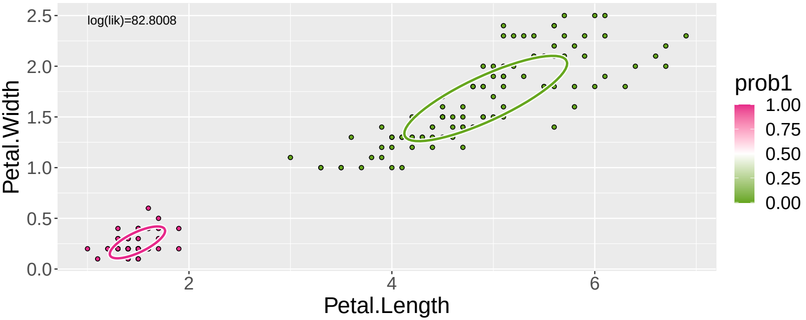
# EM algo starting with K-means assignments



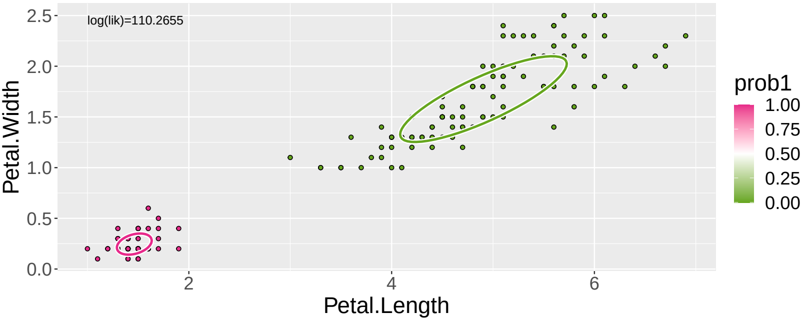
# Compute weights, means, covariance matrices



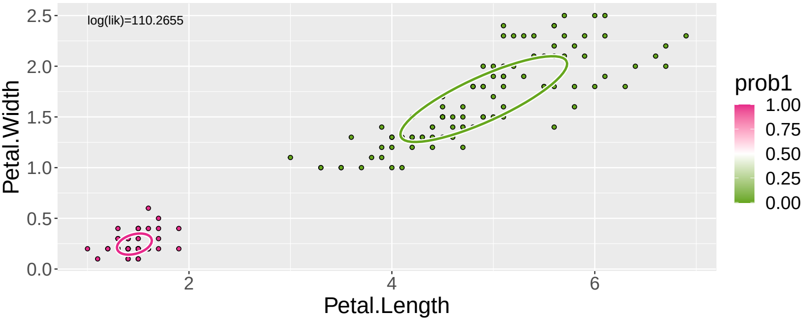
# Cluster probabilities updated



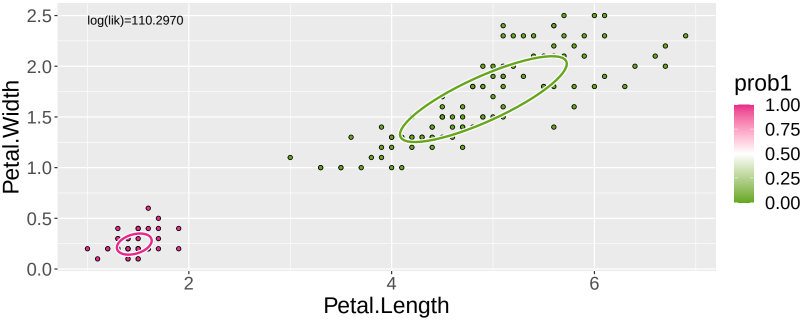
# Compute new cluster parameters



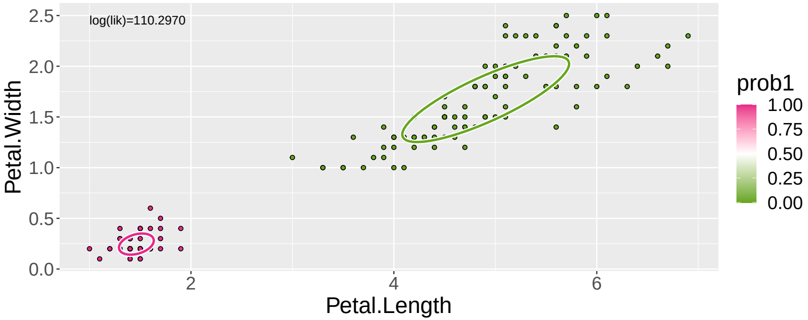
# Compute new cluster/data probabilities



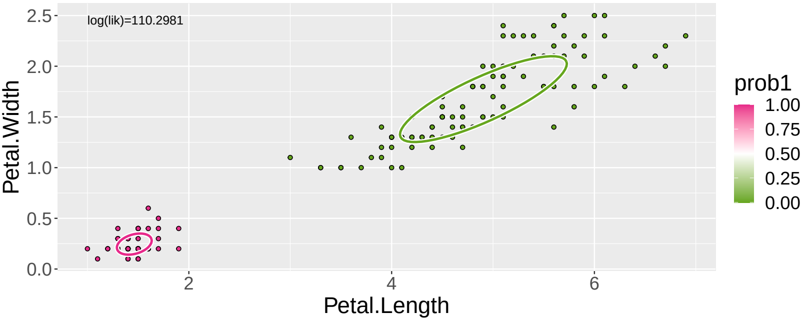
# Compute cluster parameters iteration 3



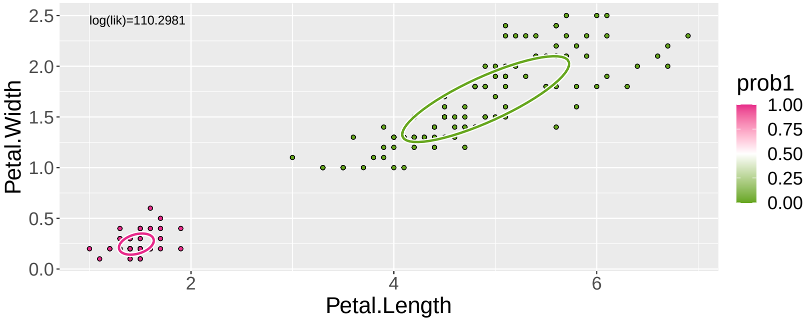
# Compute probabilities iteration 3



# Compute cluster parameters iteration 4



# Compute probabilities iteration 4



# Compute cluster parameters iteration 5 (no change = stop)