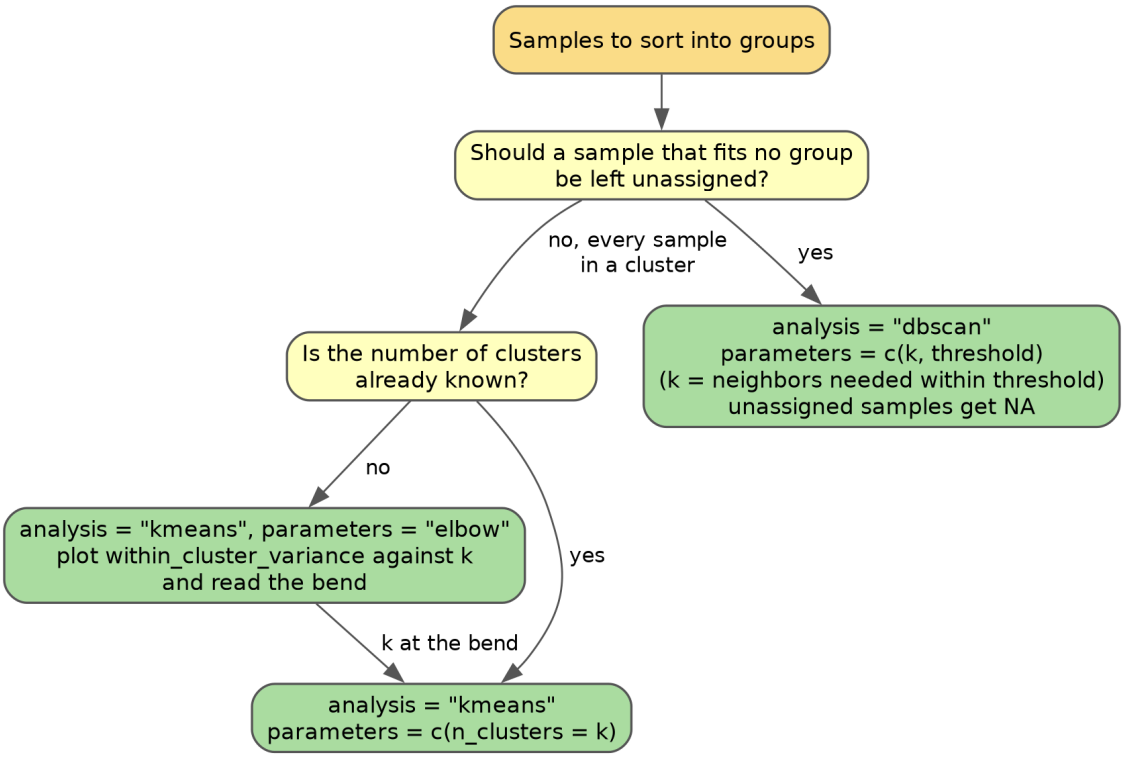
flat clustering: summary

Integrated Bioanalytical Chemistry · thebustalab.github.io/integrated_bioanalytics



Flat clustering sorts samples straight into groups. **K-means** places every sample in the cluster whose center it sits closest to, so it needs the number of clusters determined ahead of time and can never leave a sample out of a cluster. **Dbscan** grows clusters outwards from points with enough close neighbors and can leave a sample that fits no cluster as `NA`. For k-means, plot the within-cluster variance against k and examine whether there is a bend in the plot (the elbow). A sharp elbow means the number of groups is a property of the data, and a gentle one means that the data are not obviously structured into clusters. Both methods scale the data first by default. Once samples have clusters, the `cluster` column works like any other grouping column.

Function or setting	What it does	What to vary
`runMatrixAnalysis(analysis = "kmeans", parameters = c(n_clusters = 3))`	sorts every sample into the number of clusters asked for	`n_clusters`
`analysis = "kmeans", parameters = "elbow"`	returns k and `within_cluster_variance` for one to ten clusters, for an elbow plot	`c("elbow", 6)` sets a different maximum
`analysis = "dbscan", parameters = c(4, 0.3)`	grows clusters from samples with at least k neighbors within a threshold distance, and leaves the rest as NA	k (the number of neighbors), then the threshold (measured on the scaled columns)
`columns_w_values_for_single_analyte`	names the columns to cluster on	any number of columns
`scale_variance`	scales each column before clustering (TRUE by default)	leave on when columns are in different units
`group_by(cluster) %>% summarize(n = n())`	counts the samples in each cluster (the NA row counts unassigned samples, not a cluster)	other summary statistics
`left_join(x, y, by = "fragment")`	brings back columns that were not clustered on	`by`, and `select()` the needed columns of y first
`geom_mark_ellipse(aes(fill = cluster, label = cluster))`	draws an ellipse around each cluster	use `drop_na()` on dbscan output to skip the NA samples