

Lecture1_Clustering

February 1, 2021

1 MATH310 - Lecture 1 (k -means clustering, theory and examples)

In the beginning of this course we will mainly refer to available material for the book “[Introduction to Applied Linear Algebra – Vectors, Matrices, and Least Squares](#)” (VMLS) with corresponding available resources.

Linear Algebra requirements and geometric considerations required to understand the k -means Clustering is covered in [VMLS, ch. 1-4](#) and the corresponding [VMLS-slides, ch. 1-4](#).

2 The required key concepts are:

- The Euclidean (standard) inner product and norm in \mathbf{R}^n
- Distance in \mathbf{R}^n .
- Cauchy-Schwartz inequality.
- Definition of the angle between two vectors in \mathbf{R}^n .
- The triangle inequality.
- **Clustering:** The k -means algorithm for grouping unlabelled datapoints (Ch 4.1-4.3 in [VMLS](#))

3 To get started with Julia you should

- [Read the Julia language companion, ch. 1-4](#)
- [Explore the VMLS Julia resource links here.](#)

4 The standard inner product and norm in \mathbf{R}^n ([chapter 1.4](#) in VMLS)

The the standard [inner product \(scalar product\)](#) between two vectors

$$\mathbf{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix}, \mathbf{v} = \begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix} \in \mathbf{R}^n$$

is defined as the number

$$\mathbf{u} \cdot \mathbf{v} = \mathbf{u}^t \mathbf{v} = \sum_{i=1}^n u_i v_i.$$

The **norm** of \mathbf{v} in \mathbf{R}^n is defined as

$$\|\mathbf{v}\| = \sqrt{\mathbf{v} \cdot \mathbf{v}} = \sqrt{\sum_{i=1}^n v_i v_i},$$

i.e. the square root of the inner product of \mathbf{v} with itself.

5 The distance between vectors in Euclidean space (**chapter 3.2 in VMLS**)

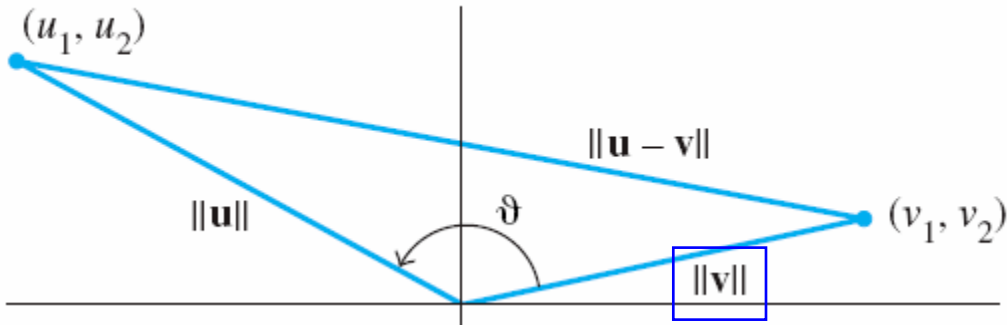
The distance between two vectors $\mathbf{u}, \mathbf{v} \in \mathbf{R}^n$ is defined as the norm of their difference

$$dist(\mathbf{u}, \mathbf{v}) = \|\mathbf{u} - \mathbf{v}\| = \sqrt{(\mathbf{u} - \mathbf{v})^t (\mathbf{u} - \mathbf{v})} = \sqrt{\sum_{i=1}^n (u_i - v_i)^2}.$$

6 Angles between vectors

For any pair of vectors \mathbf{u}, \mathbf{v} in \mathbf{R}^2 or \mathbf{R}^3 in angle ϑ , there is an important relationship given by their norms and the inner product:

$$\begin{aligned} \mathbf{u} \cdot \mathbf{v} &= \|\mathbf{u}\| \|\mathbf{v}\| \cos(\vartheta) \\ \Updownarrow \\ \cos(\vartheta) &= \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}. \end{aligned}$$



The angle ϑ between two vectors \mathbf{u} and \mathbf{v} .

The cosine-formula can be extended to a general definition of the angle between two vectors \mathbf{u}, \mathbf{v} in \mathbf{R}^n :

7 The Cauchy-Schwartz (CS) inequality ([chapter 3.4](#) in VMLS)

Theorem (Cauchy-Schwartz inequality)

If $\mathbf{u}, \mathbf{v} \in \mathbf{R}^n$, then

$$|\mathbf{u} \cdot \mathbf{v}| \leq \|\mathbf{u}\| \|\mathbf{v}\|,$$

and equality only applies when either \mathbf{u} or \mathbf{v} is equal to $\mathbf{0}$, or when $\mathbf{u} = k\mathbf{v}$ for $k \in \mathbf{R}$.

Proof:

If \mathbf{u} or \mathbf{v} is equal to $\mathbf{0}$ equality in the above formula holds by inspection. Therefore we assume that $\mathbf{u}, \mathbf{v} \neq \mathbf{0}$. Let $\mathbf{p} = \mathbf{v}(\mathbf{v}^t \mathbf{v})^{-1} \mathbf{v}^t \mathbf{u} = \left(\frac{\mathbf{v} \cdot \mathbf{u}}{\mathbf{v} \cdot \mathbf{v}}\right) \mathbf{v}$ be the orthogonal projection of \mathbf{u} onto \mathbf{v} . Because \mathbf{p} is orthogonal to the residual vector $(\mathbf{u} - \mathbf{p})$, [Pythagoras theorem](#) implies

$$\|\mathbf{p}\|^2 + \|\mathbf{u} - \mathbf{p}\|^2 = \|\mathbf{u}\|^2 \Rightarrow \|\mathbf{p}\|^2 \leq \|\mathbf{u}\|^2.$$

Because $\mathbf{p} = \left(\frac{\mathbf{v} \cdot \mathbf{u}}{\mathbf{v} \cdot \mathbf{v}}\right) \mathbf{v} = \frac{(\mathbf{v} \cdot \mathbf{u})}{\|\mathbf{v}\|^2} \mathbf{v}$,

$$\|\mathbf{p}\|^2 = \mathbf{p} \cdot \mathbf{p} = \frac{(\mathbf{v} \cdot \mathbf{u})^2}{\|\mathbf{v}\|^4} \|\mathbf{v}\|^2 = \frac{(\mathbf{v} \cdot \mathbf{u})^2}{\|\mathbf{v}\|^2} \leq \|\mathbf{u}\|^2$$

↓

$$(\mathbf{v} \cdot \mathbf{u})^2 \leq \|\mathbf{u}\|^2 \|\mathbf{v}\|^2,$$

and the CS inequality follows by taking the square root of both sides of this inequality ■

An alternative proof of CS is given [on page 57](#) in VMLS.

8 Definition: Angle between vectors in \mathbf{R}^n

We define the cosine of the angle θ between two vectors $\mathbf{u}, \mathbf{v} \in \mathbf{R}^n$ by the formula

$$\cos(\theta) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}.$$

This definition is sound because of the CS-inequality:

$$|\mathbf{u} \cdot \mathbf{v}| \leq \|\mathbf{u}\| \|\mathbf{v}\| \Leftrightarrow \frac{|\mathbf{u} \cdot \mathbf{v}|}{\|\mathbf{u}\| \|\mathbf{v}\|} \leq 1 \Leftrightarrow -1 \leq \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|} \leq 1 \Rightarrow -1 \leq \cos(\theta) \leq 1.$$

Finally we define the angle θ between \mathbf{u} and \mathbf{v} as

$$\theta = \arccos\left(\frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}\right)$$

where arccos denotes the inverse cosine, normalized to lie in the interval $[0, \pi]$.

9 The triangle inequality

Theorem (the triangle inequality)

For all $\mathbf{u}, \mathbf{v} \in \mathbf{R}^n$ the following inequality holds:

$$\|\mathbf{u} + \mathbf{v}\| \leq \|\mathbf{u}\| + \|\mathbf{v}\|$$

Proof:

$$\begin{aligned}\|\mathbf{u} + \mathbf{v}\|^2 &= (\mathbf{u} + \mathbf{v}) \cdot (\mathbf{u} + \mathbf{v}) = \mathbf{u} \cdot \mathbf{u} + 2\mathbf{u} \cdot \mathbf{v} + \mathbf{v} \cdot \mathbf{v} \\ &\leq \|\mathbf{u}\|^2 + 2|\mathbf{v} \cdot \mathbf{u}| + \|\mathbf{v}\|^2 \\ &\quad \text{(use CS here)} \\ &\leq \|\mathbf{u}\|^2 + 2\|\mathbf{u}\|\|\mathbf{v}\| + \|\mathbf{v}\|^2 \\ &= (\|\mathbf{u}\| + \|\mathbf{v}\|)^2.\end{aligned}$$

The 2nd inequality holds due to the Cauchy-Schwartz inequality. The result follows by taking the square root of both sides of this inequality ■

10 k -means clustering material

- [The VMLS lecture slides on clustering \(page 73\)](#) and [Chapter 4 on clustering in VMLS](#).
- [A visualization tool for \$k\$ -Means Clustering](#).
- [VMLS-slides](#) on image compression by k -means clustering.

11 The computational basics of k -means clustering

Cluster analysis by the [k-means clustering](#) is a popular method for grouping high-dimensional unlabelled data.

In the following 5 videos from the [Coursera-course on Machine learning](#), Stanford-professor [Andrew Ng](#) explains the key aspects of k -means clustering:

- [Clustering - Unsupervised Learning](#) - the basic ideas (3min18sec)
- [Clustering - the k-means algorithm](#) - how the algorithm works (12min33sec)
- [Clustering - the optimization objective](#) - for measuring and comparing the goodness of solution candidates (7min05sec)
- [Clustering - random initialization](#) - heuristics on how to initialize (start) the clustering process (7min50sec)
- [Clustering - choosing the number of clusters](#) - heuristics on choosing the number of clusters (8min23sec)

Understanding the principle of k -Means Clustering idea becomes easy when playing with an interactive tool for [Visualizing the k-means clustering process](#).

12 Examples with simulated data (Julia code)

In Julia we can do the k-means clustering either by using the [VMLS-library](#) function `kmeans`, or by implementing our own version of the k-means algorithm, see `mykmeans` below.

12.1 Lets start by generating a random “artificial” dataset X containing 3 point-clouds in 2 dimensions:

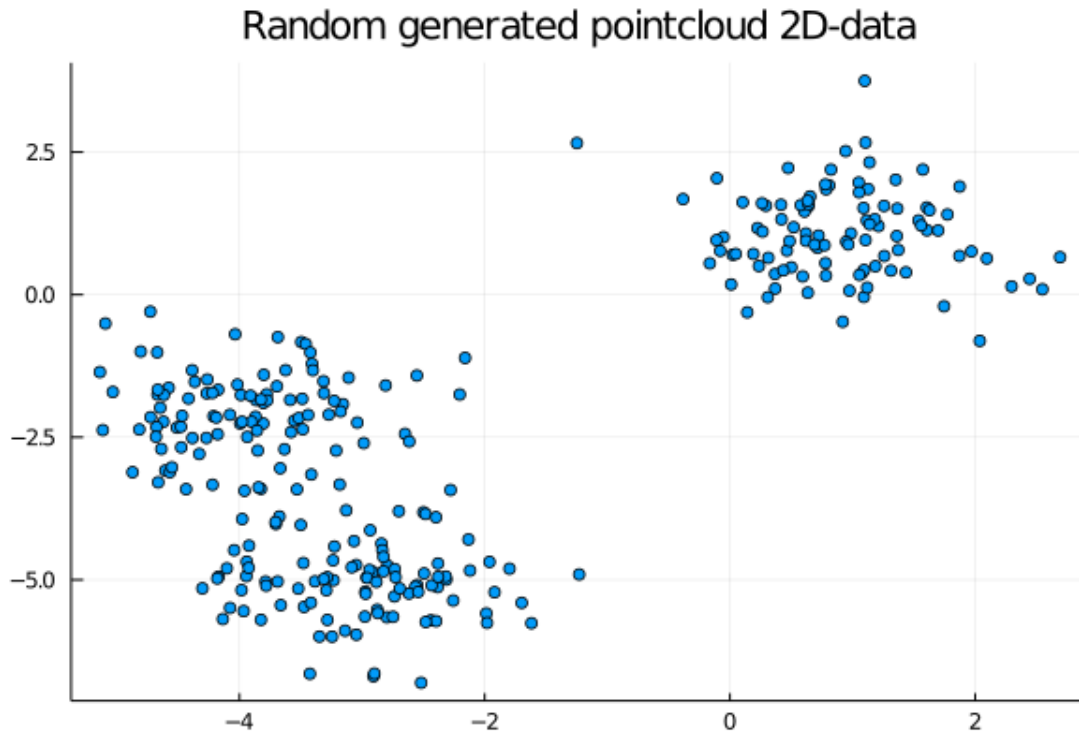
```
[1]: using Random
Random.seed!(1234) # This line assures the same random dataset to be generated
                  ↪ each time.

nn = 100;        # Generate 3 random "clouds" of datapoints (samples), each of
                  ↪ size nn.
X = [randn((nn,2))*0.7 .+ [ 1  1];
     randn((nn,2))*0.7 .+ [-3 -5];
     randn((nn,2))*0.7 .+ [-4 -2]];
# Each row of X corresponds to a datapoint
```

12.1.1 A scatterplot shows the generated dataset:

```
[2]: using Plots #b Precompiles on every startup (~20 seconds)
gr() # Needs modules Plots and GR to be installed, may need a rebuild of GR
     ↪ with ']'build GR'
default(size=(600, 400), fmt = :png) # Default plot size, change output format
     ↪ to png
```

```
[3]: # Define and display a plot of the raw random data
sp = scatter(X[:,1],X[:,2], title = "Random generated pointcloud 2D-data",
             ↪ legend = false)
display(sp)
```

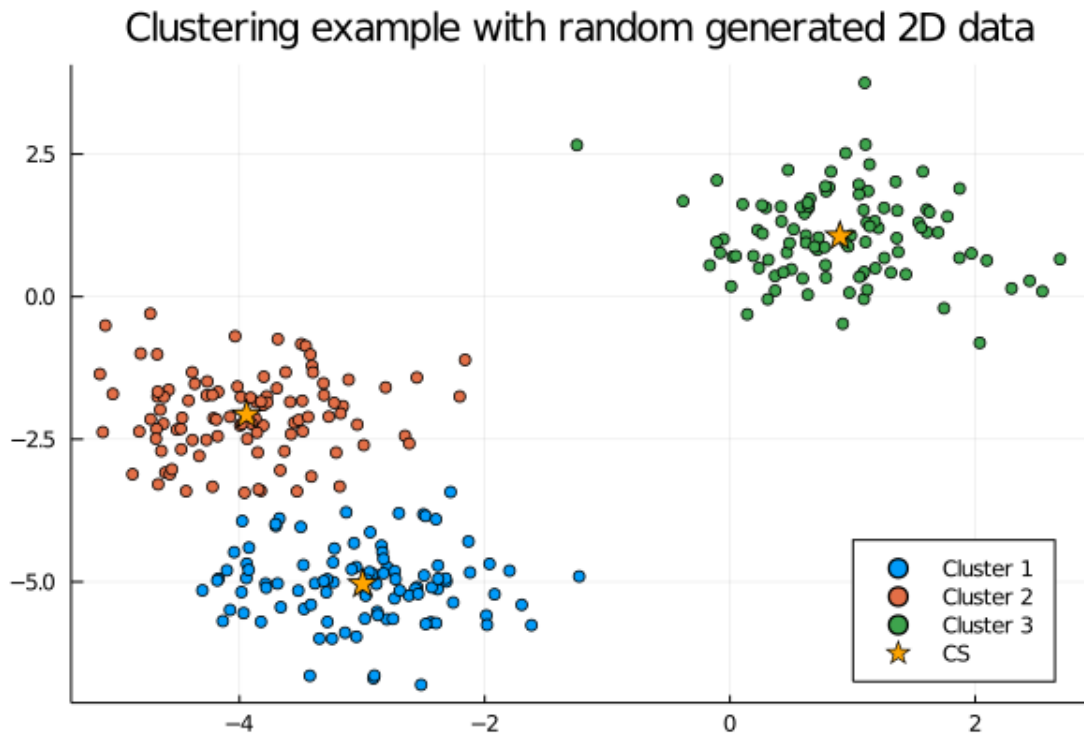


12.1.2 Apply the k-means clustering algorithm (mykmeans.jl) to search for clusters in the generated dataset X

```
[4]: include("mykmeans.jl")
k = 3;      # The suggested number of clusters (you should also repeatedly try k
    ↪ = 2, 4 and 5)
Cid, CS, J = mykmeans(X, k);
```

12.1.3 We plot the solution with coloring of each identified cluster

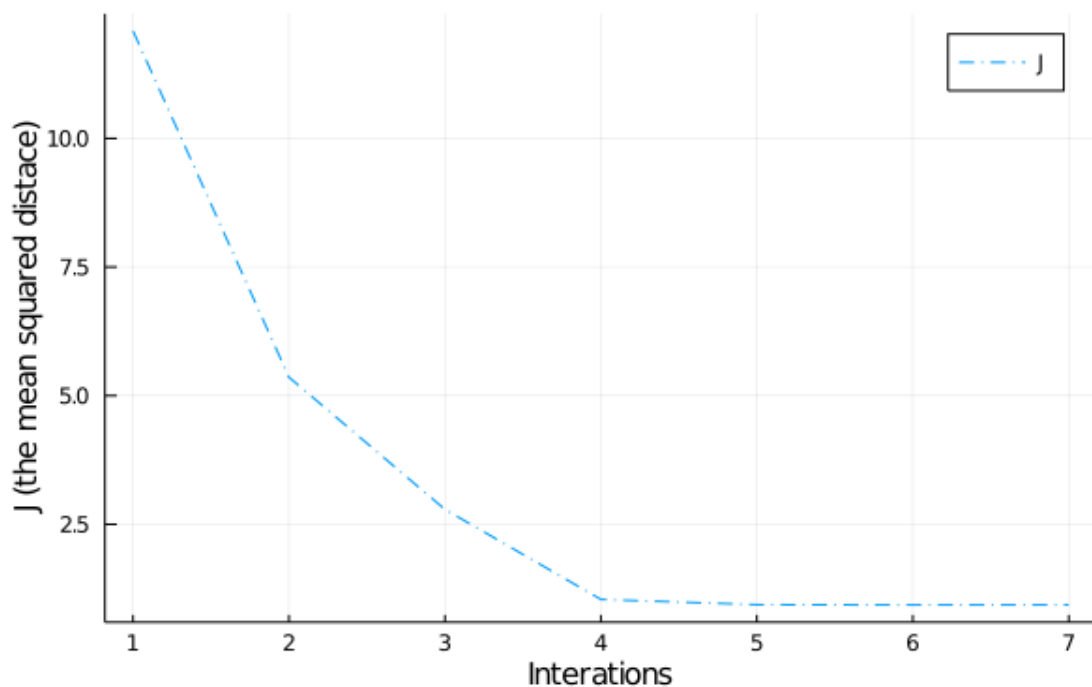
```
[6]: # Define and display a plot of the clustered data and the cluster centers(CS):
p = plot(title = "Clustering example with random generated 2D data",
    label = " ", legend = :bottomright, size = (600, 400))
for i=1:k
    snr = vec(Cid.==i) # the sample numbers of the j-th cluster
    scatter!(p, X[snr,1], X[snr,2], label = string("Cluster ",i))
end
scatter!(p, CS[:,1],CS[:,2], marker = :star, markersize = 8, color = :orange,
    ↪ label = "CS")
display(p)
```



12.1.4 Plotting the objective function values for the iterative clustering process

```
[7]: # Define and display a plot of the objective function values reflecting the
      ↪ clustering process
Jp = plot(J, linestyle = :dashdot, title = "Objective function (J) values -
      ↪ monitoring the clustering process",
          ylabel = "J (the mean squared distace)", xlabel = "Interations", label =
      ↪ "J")
display(Jp)
```

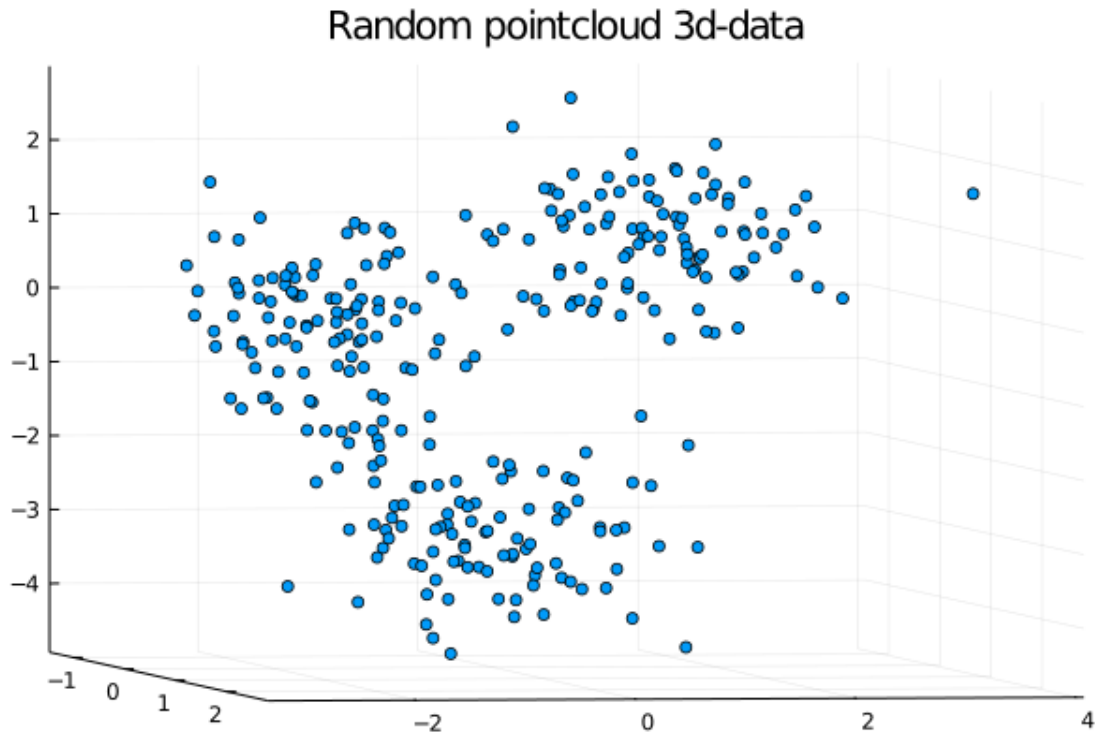
Objective function (J) values - monitoring the clustering proce



12.2 A random “artificial” dataset X containing 3 point-clouds in 3 dimensions:

```
[8]: ## Visualization for 3D data
# Here is a corresponding 3-dimensional dataset:
Random.seed!(1234) # This line assures the same random dataset to be generated
↳ each time.
nn = 100;
X = [randn(nn,3)*0.7 .+ [1 1 1];
     randn(nn,3)*0.7 .+ [1 -2 -0];
     randn(nn,3)*0.7 .+ [0 0 -3]];
```

```
[9]: # Define and display a plot of the raw random data
sp = scatter(X[:,1],X[:,2],X[:,3], title = "Random pointcloud 3d-data", legend_
↳ = false, camera = (75,10))
display(sp)
```

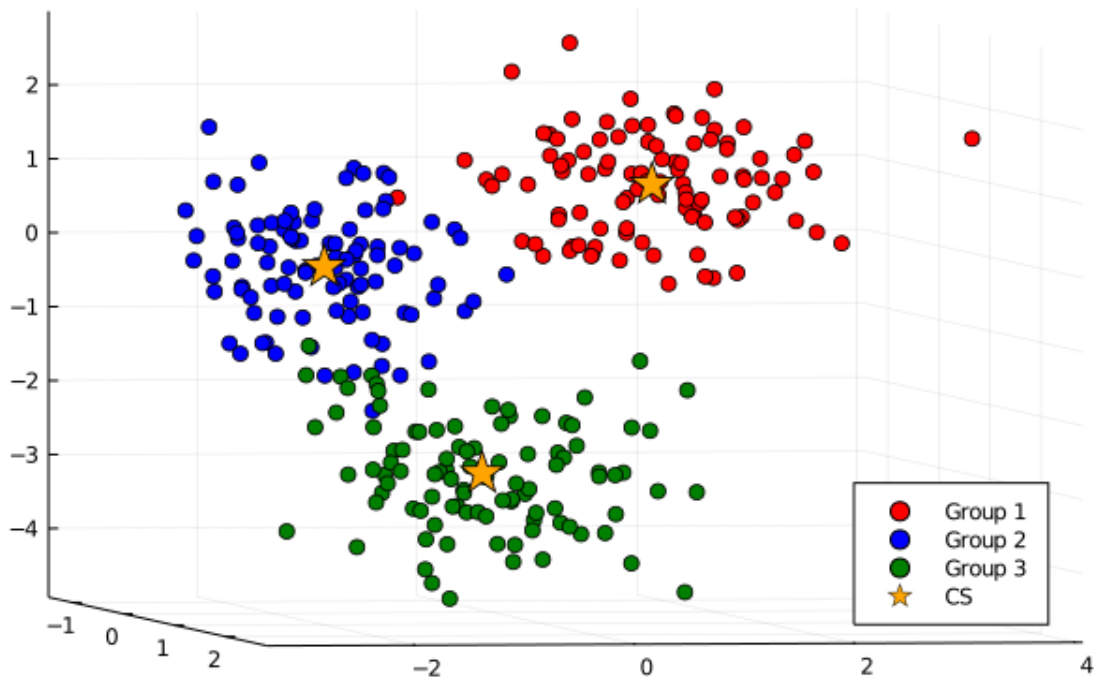
12.2.1 We use our k-means algorithm to cluster the 3d-data.

```
[10]: k = 3;      # The suggested number of clusters (you should also repeatedly try k
      ↪= 3, 4 and 5)
      Cid, CS, J = mykmeans(X, k);
```

```
[11]: ##
      # Plotting the clustered data and the cluster-centers:
      colors = [:red, :blue, :green, :cyan, :magenta, :black];

      # Define and display a plot of the clustered data and the cluster centers(CS):
      p = plot(legend = :bottomright, title = "Clustering example with random_
      ↪generated 3D data", size = (600,400))
      for j = 1:k
          snr = vec(Cid.==j); # the sample numbers of the j-th cluster
          # Plot only this group:
          scatter!(p, X[snr,1],X[snr,2],X[snr,3],color = colors[j], markersize = 5,
          ↪label = string("Group ", j))
      end
      scatter!(p, CS[:,1],CS[:,2], CS[:,3], marker = :star, markersize = 12, color = :
      ↪orange, label = "CS", camera = (75,10)) # Plotting the cluster centers
      display(p)
```

Clustering example with random generated 3D data



12.2.2 Plotting the objective function values for the iterative clustering process

```
[12]: # Define and display a plot of the objective function values reflecting the
      ↪ clustering process
Jp = plot(J, linestyle = :dashdot, title = "Objective function (J) values -
      ↪ monitoring the clustering process",
          ylabel = "J (the mean squared distance)", xlabel="Iteration", label="J")
display(Jp)
```



13 Exercises

- Make an extension of the **mykmeans** algorithm by considering ($r \geq 2$) repeated random initializations of the cluster centers. The new algorithm should return the cluster centers of the best among the R solutions in terms of J-value.

An alternative measure of similarity between vectors is obtained by considering angles rather than distances.

- Figure out, and implement a modification of mykmeans that measure similarity by the angle between vectors.

14 Other clustering techniques:

See Wikipedia-review on [Cluster analysis](#).