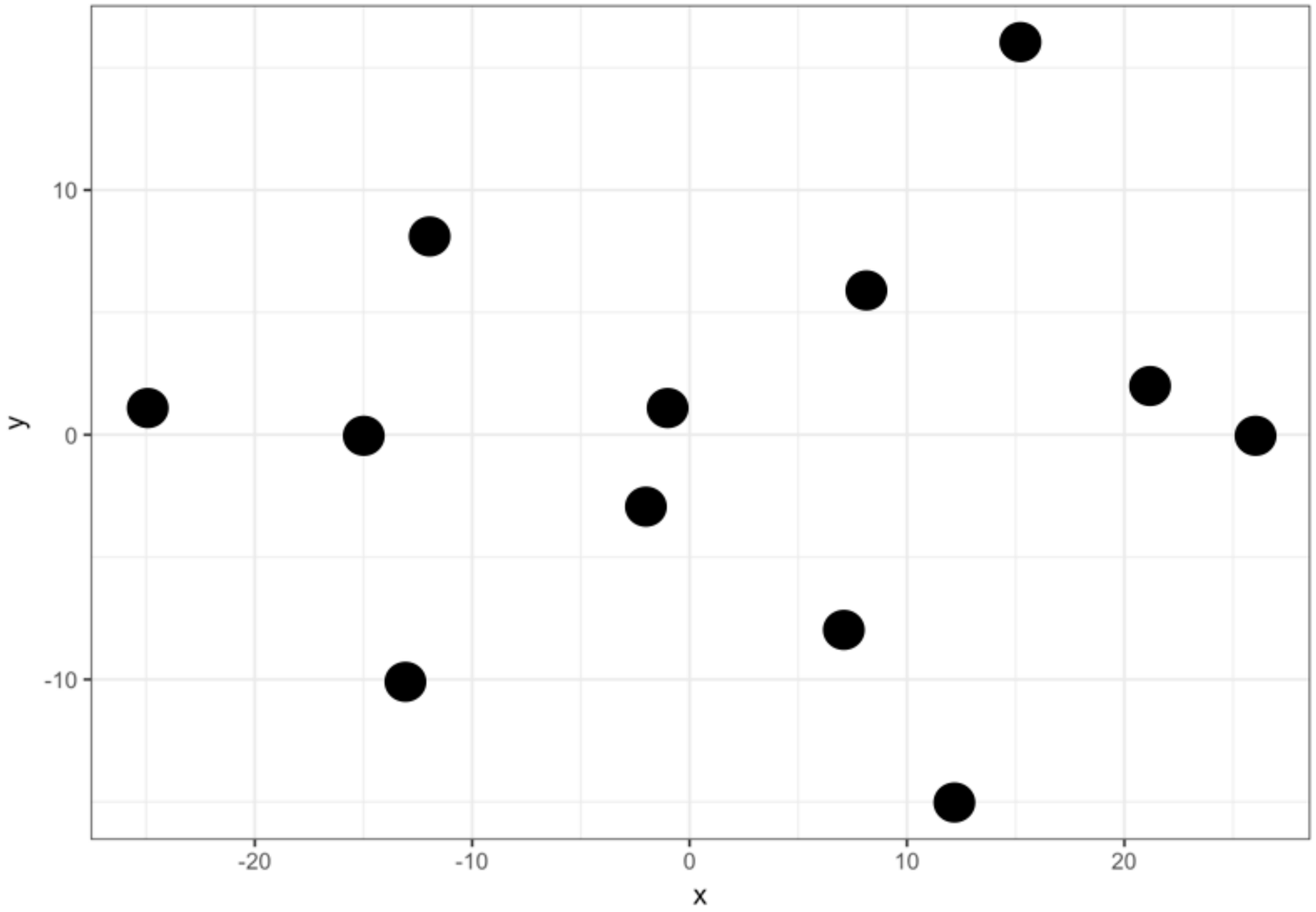
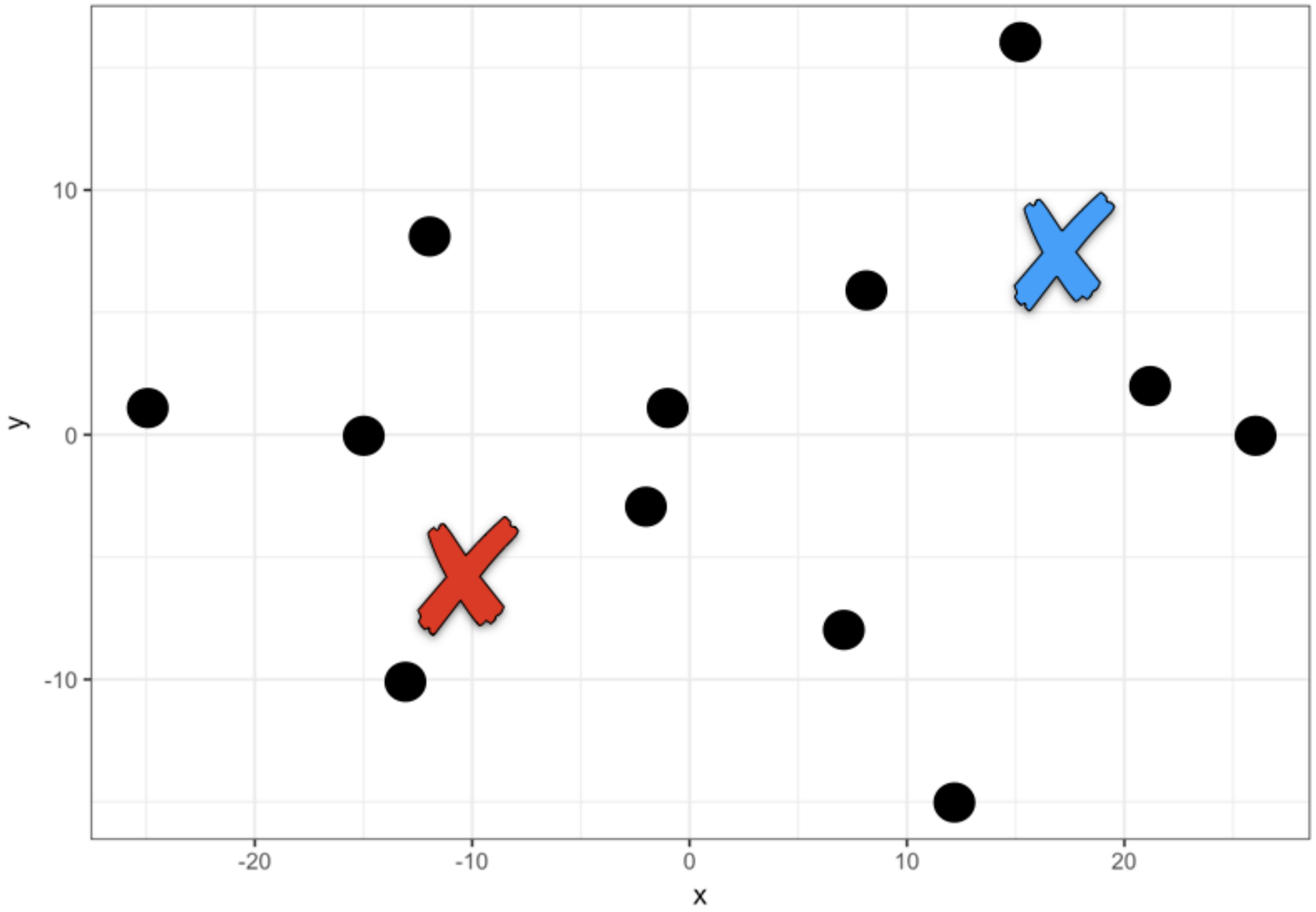
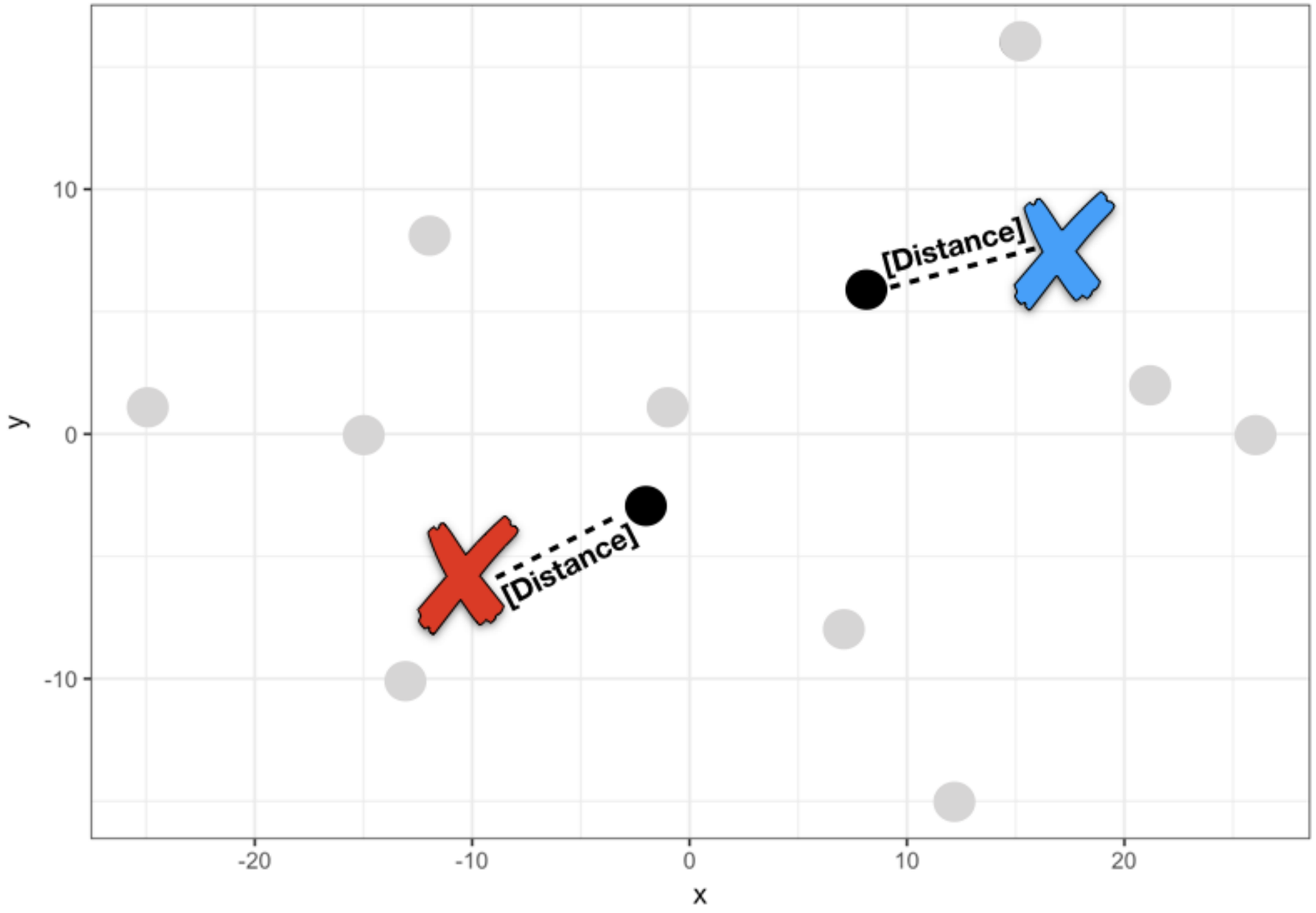
Introduction to K-means

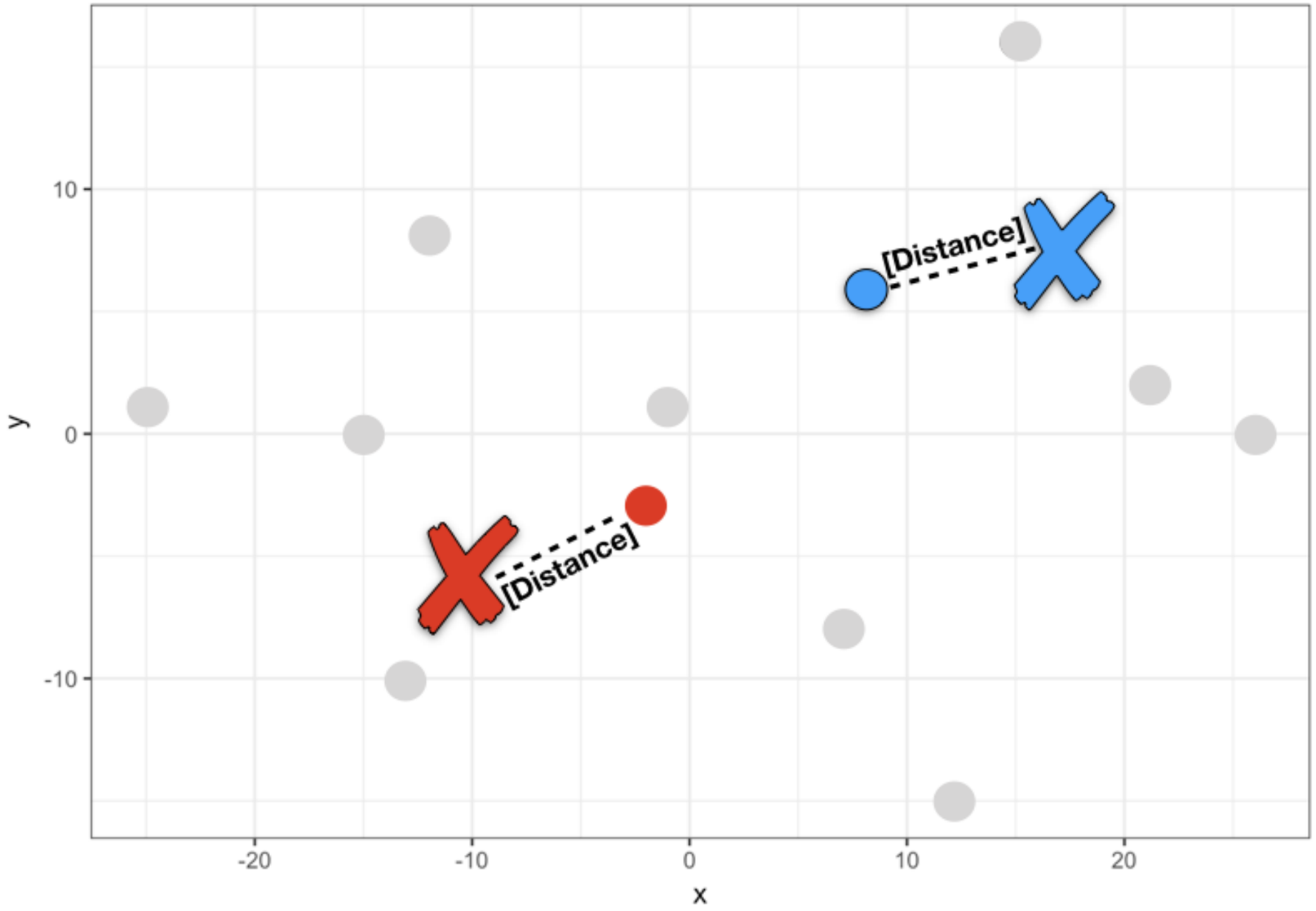
CLUSTER ANALYSIS IN R

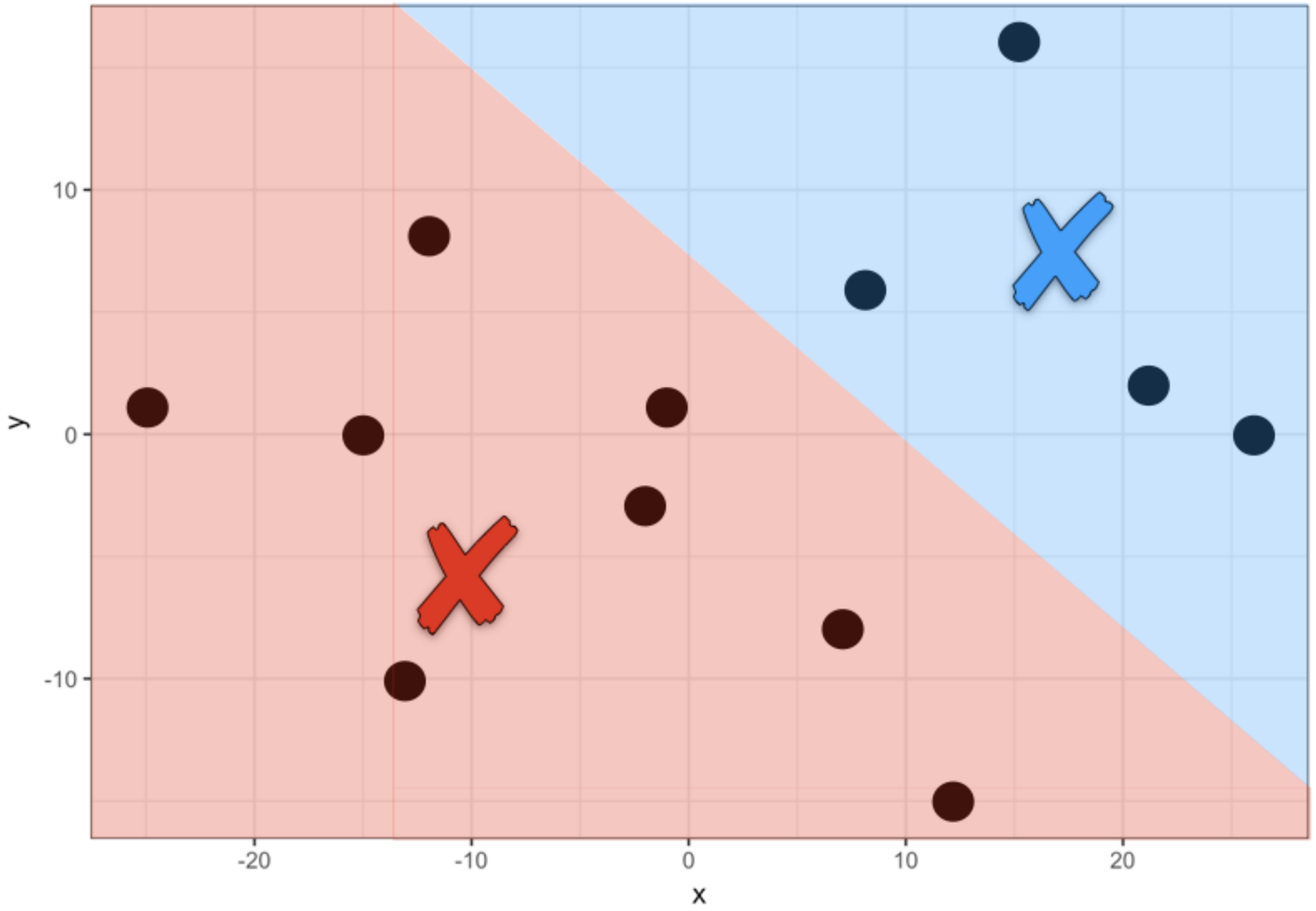
Dmitriy (Dima) Gorenshteyn
Lead Data Scientist, Memorial Sloan Kettering Cancer Center

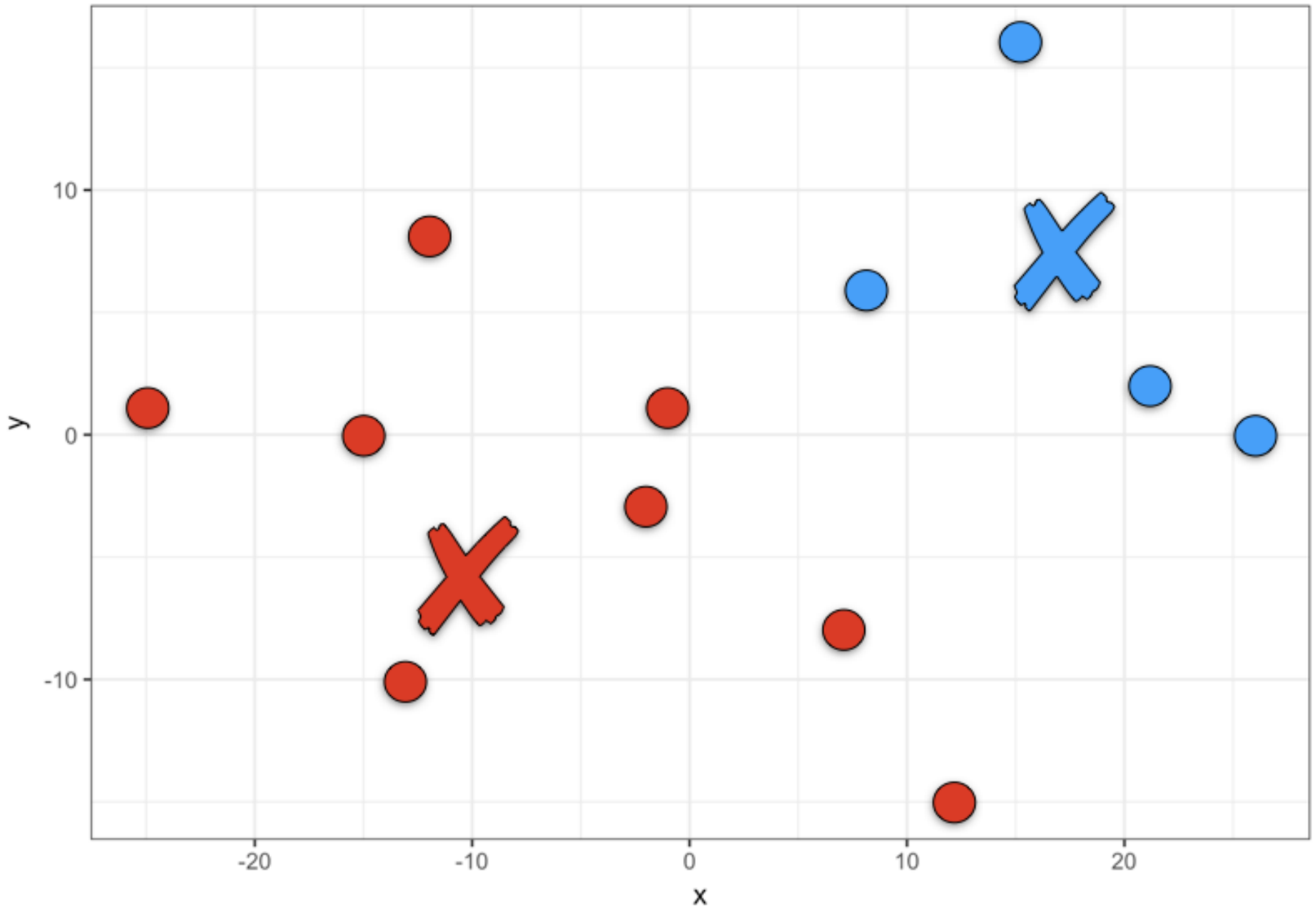

10
0
-10
y
-20
-10
0
10
20
x

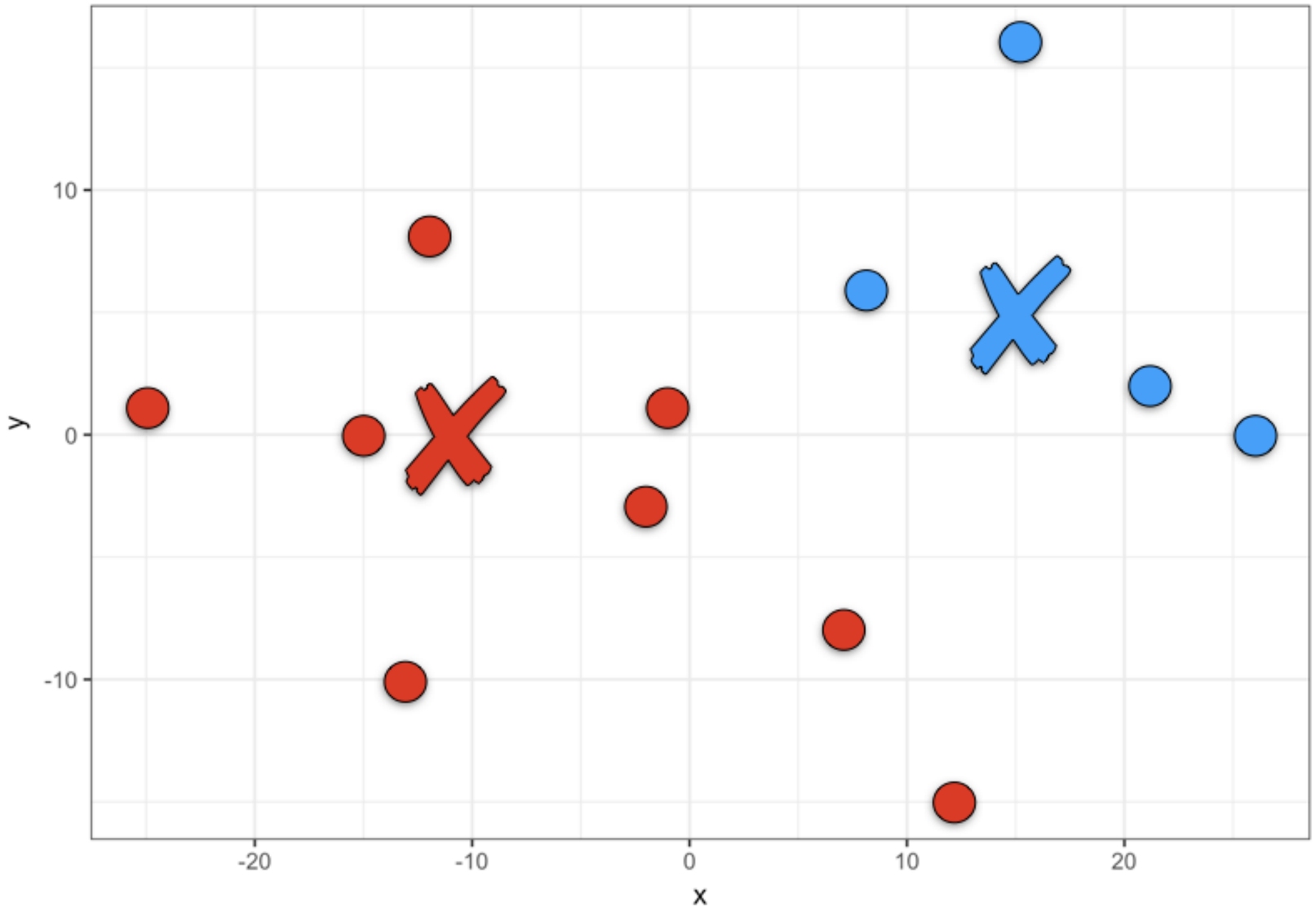

10
0
-10
y
-20
-10
0
10
20
x

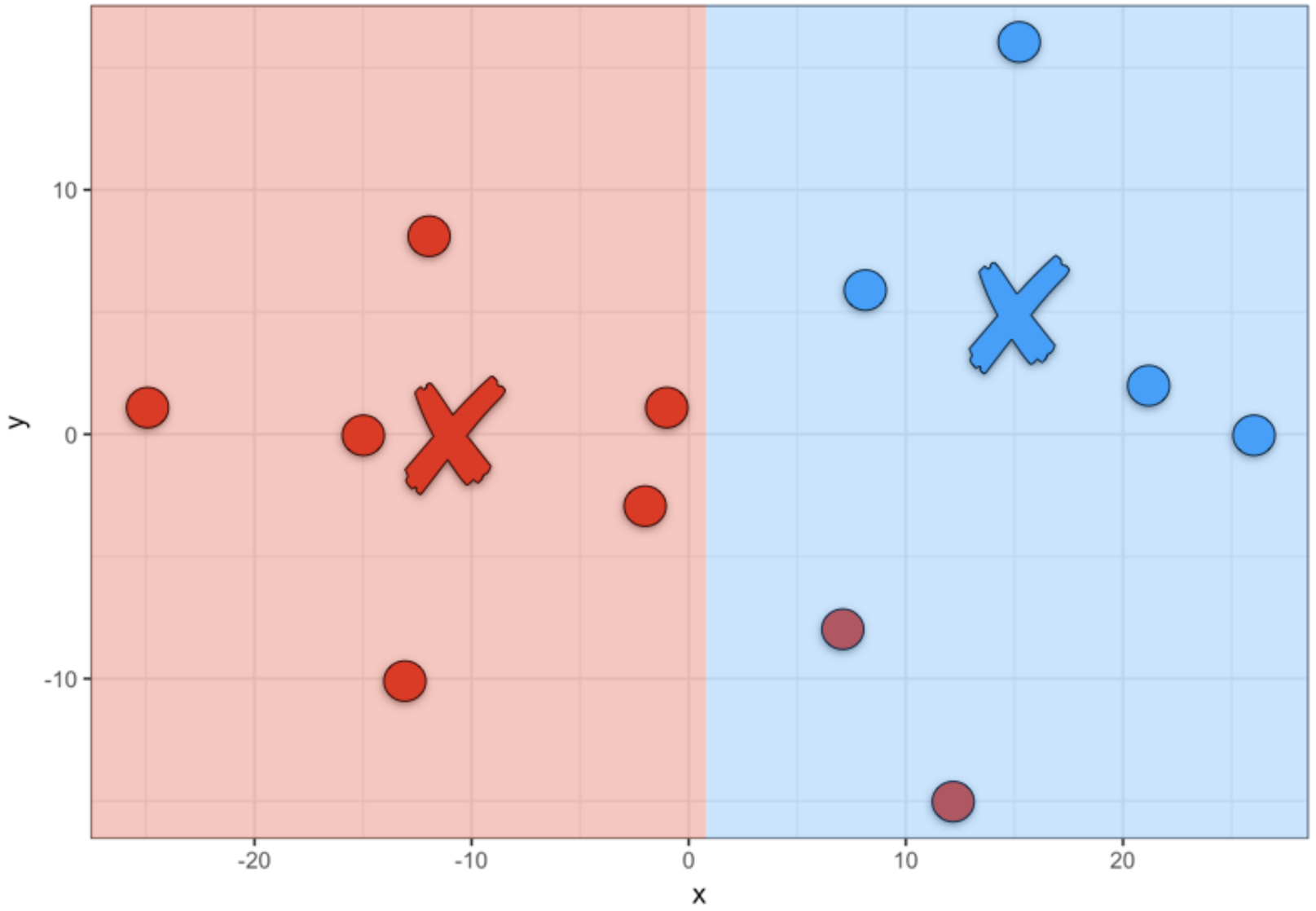

[Distance]
[Distance]
10
0
-10
y
-20
-10
0
10
20
x

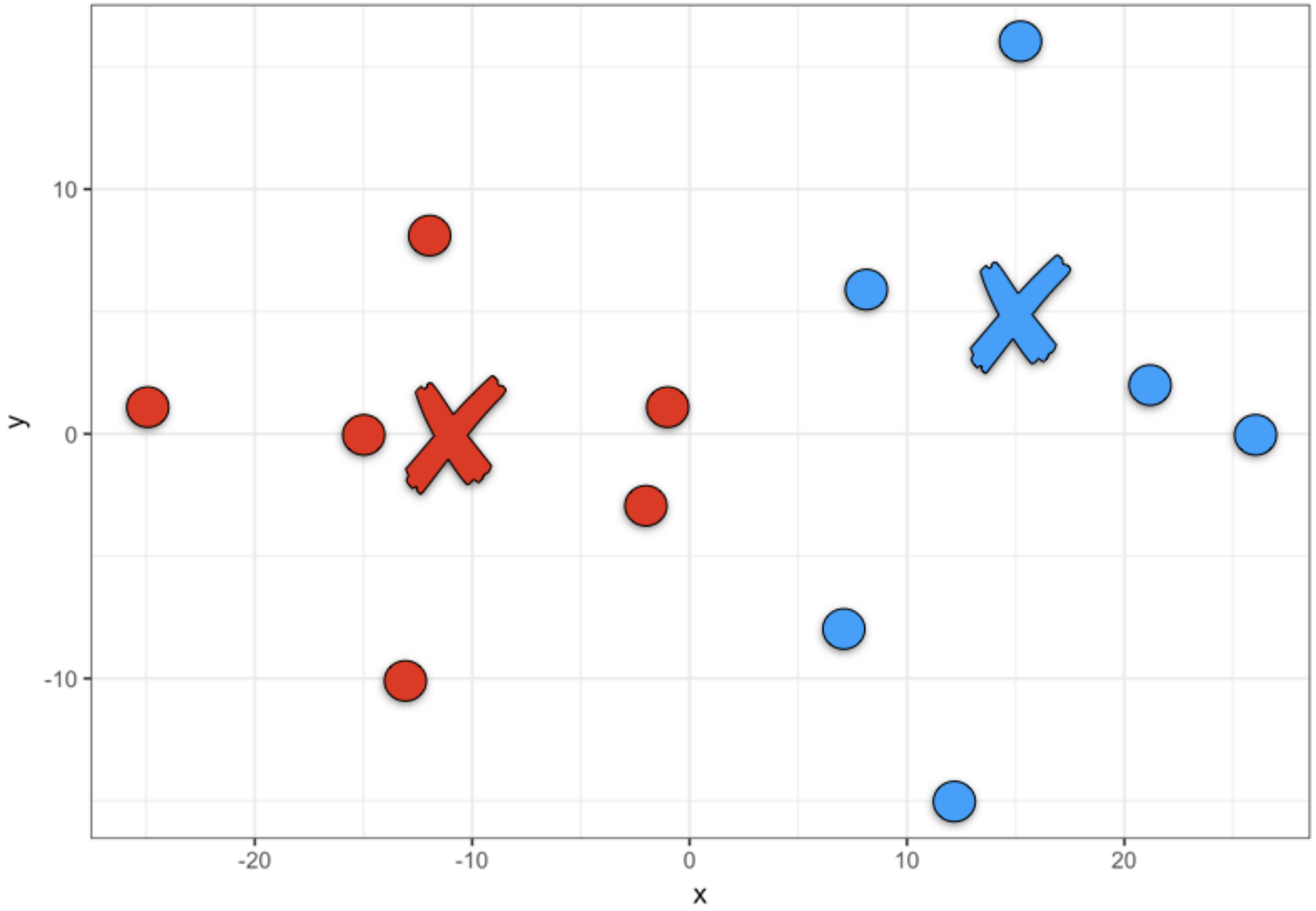

10
0
-10
y
-20
-10
0
10
20
x
[Distance]
[Distance]

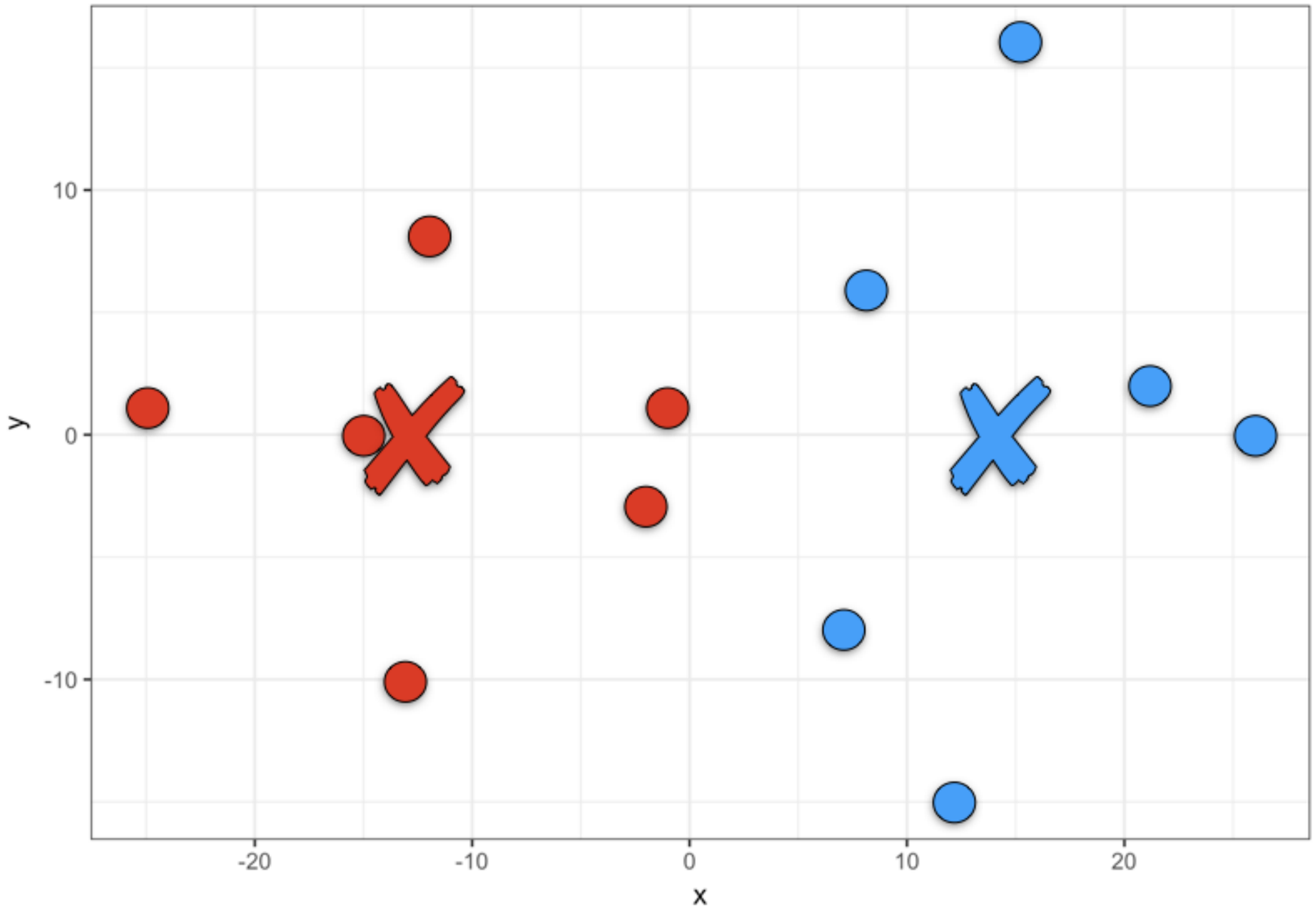

10
0
-10
y
-20
-10
0
10
20
x


10
0
-10
y
-20
-10
0
10
20
x


10
0
-10
y
-20
-10
0
10
20
x


10
0
-10
y
-20
-10
0
10
20
x


10
0
-10
y
-20
-10
0
10
20
x


10
0
-10
y
-20
-10
0
10
20
x

kmeans()

```
print(lineup)
```

```
      x     y
1    -1     1
2    -2    -3
3     8     6
4     7    -8
...  ...   ...
```

```
model <- kmeans(lineup, centers = 2)
```

Assigning clusters

```
print(model$cluster)
```

```
1 1 2 2 1 1 1 2 2 2 1 2
```

```
lineup_clustered <- mutate(lineup, cluster = model$cluster)
print(lineup_clustered)
```

```
       x      y cluster
   <dbl> <dbl>   <int>
 1    -1     1       1
 2    -2    -3       1
 3     8     6       2
 4     7    -8       2
 ...   ...   ...     ...
```

Let's practice!

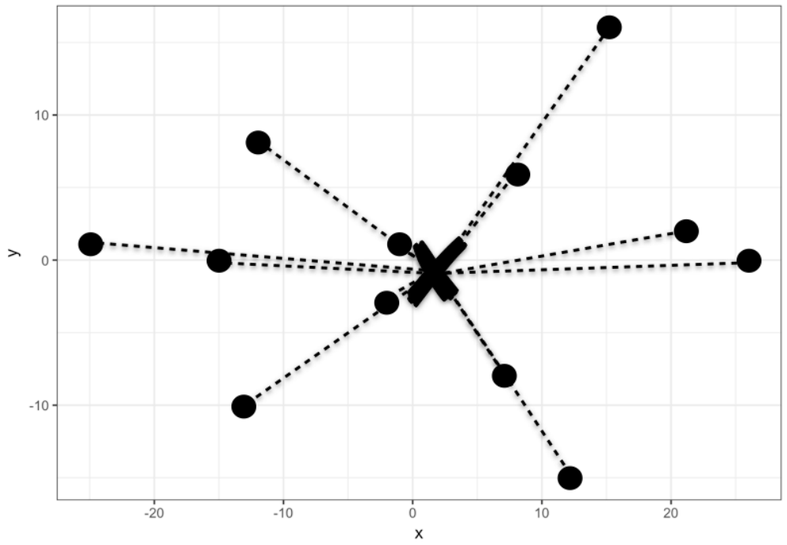
CLUSTER ANALYSIS IN R

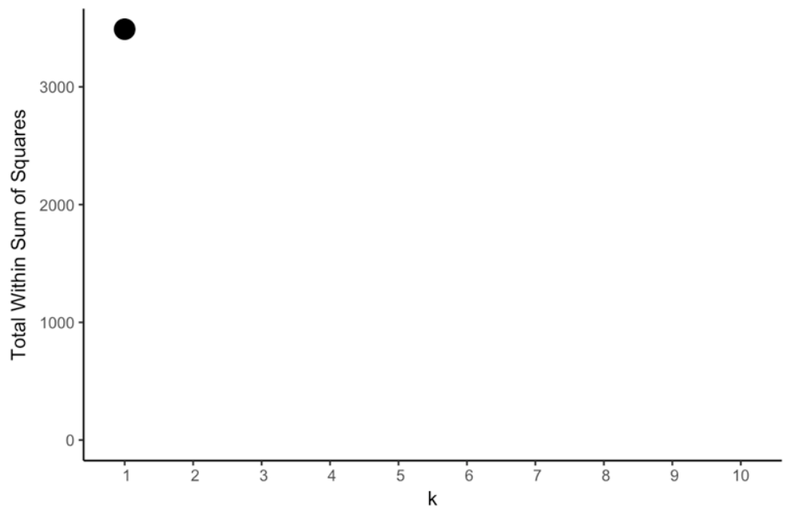
Evaluating different values of K by eye

CLUSTER ANALYSIS IN R

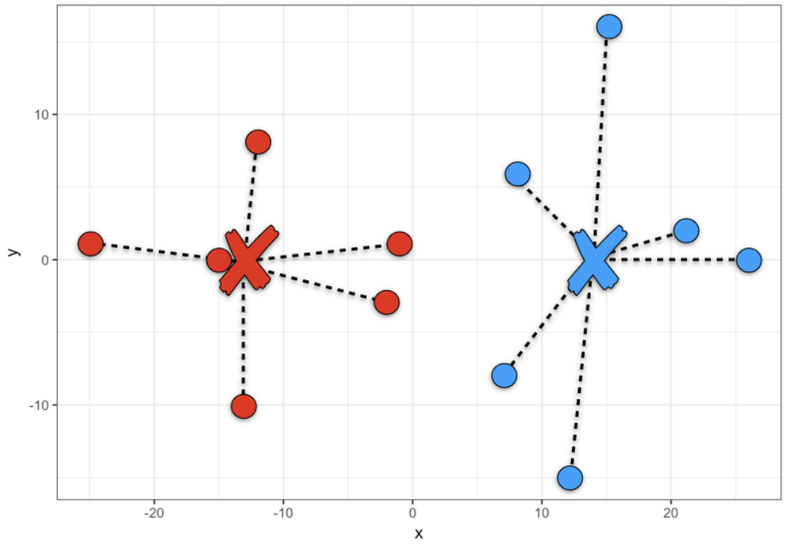
Dmitriy (Dima) Gorenshteyn
Lead Data Scientist, Memorial Sloan Kettering Cancer Center

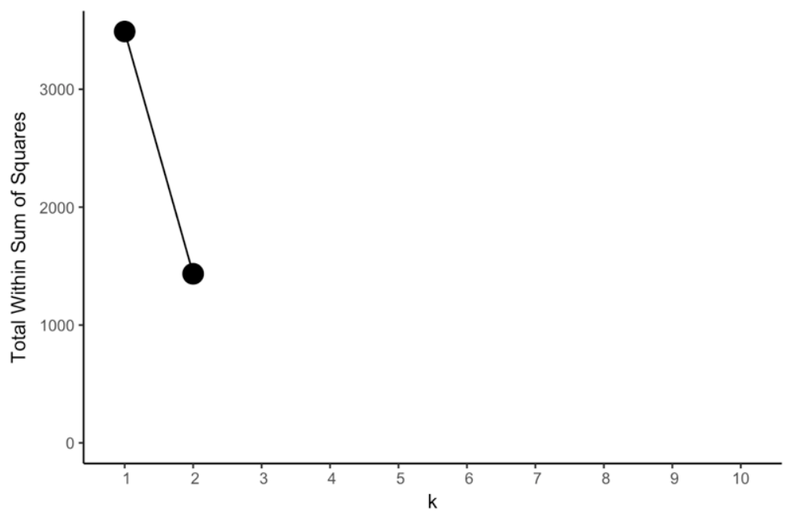
Total within-cluster sum of squares: k = 1



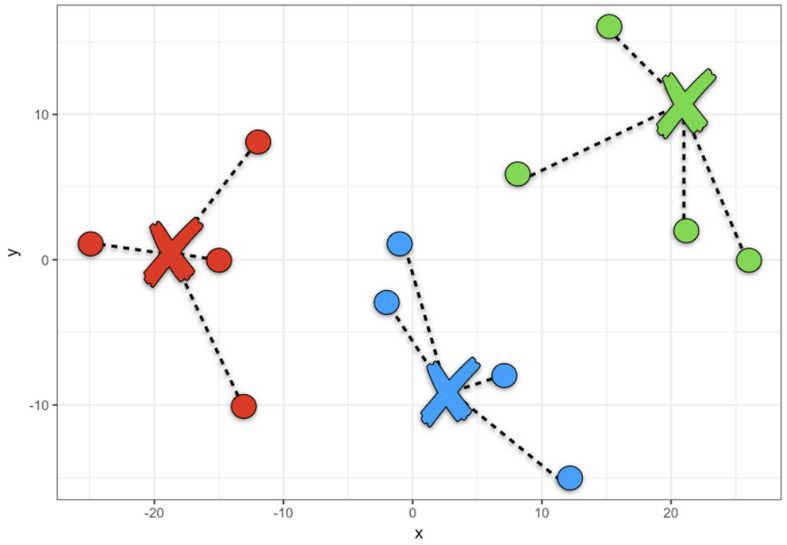


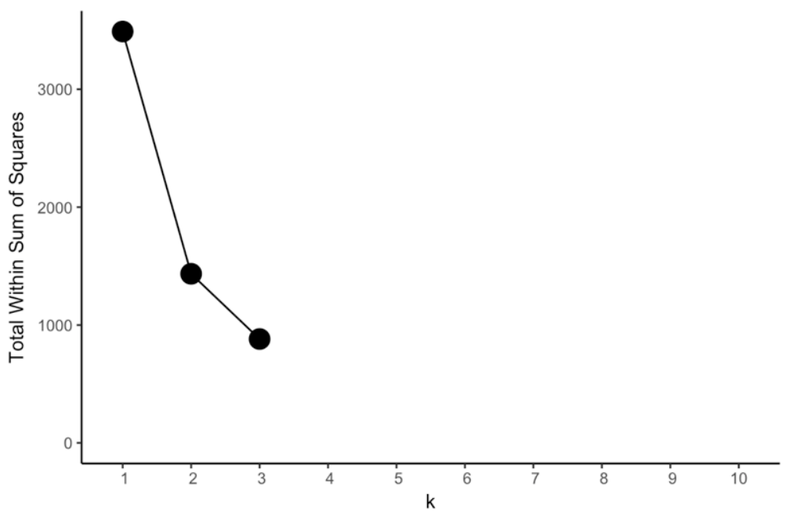
Total within-cluster sum of squares: k = 2



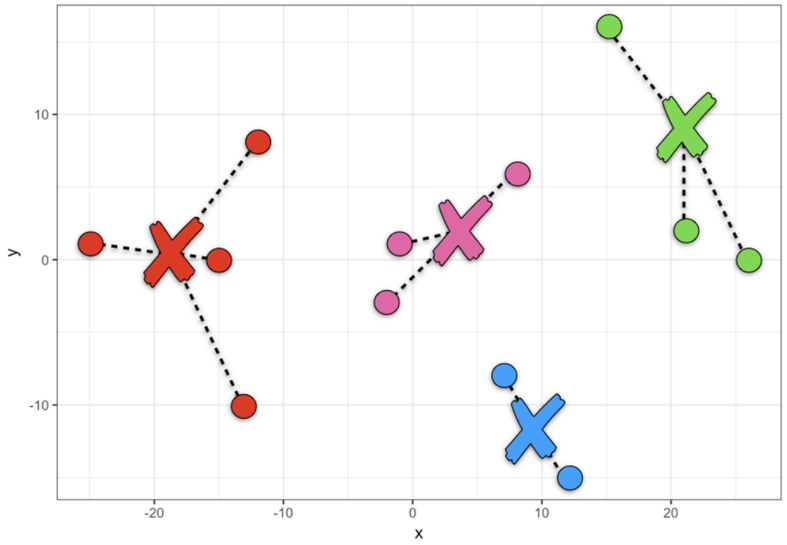


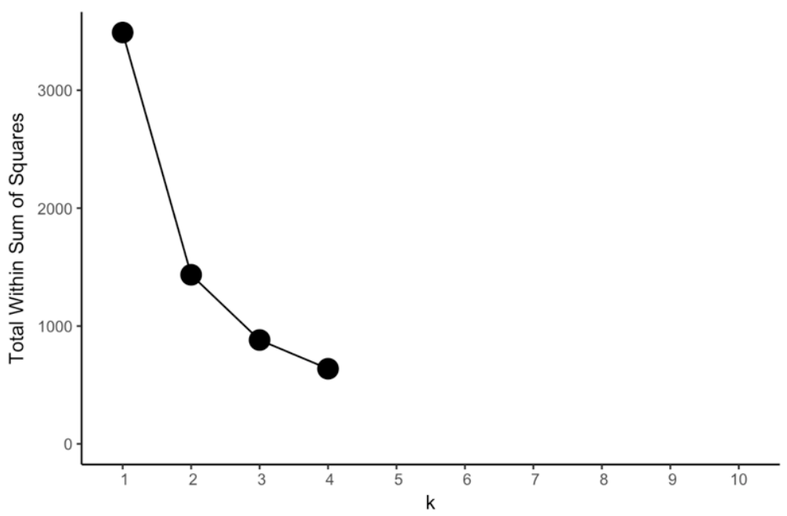
Total within-cluster sum of squares: k = 3



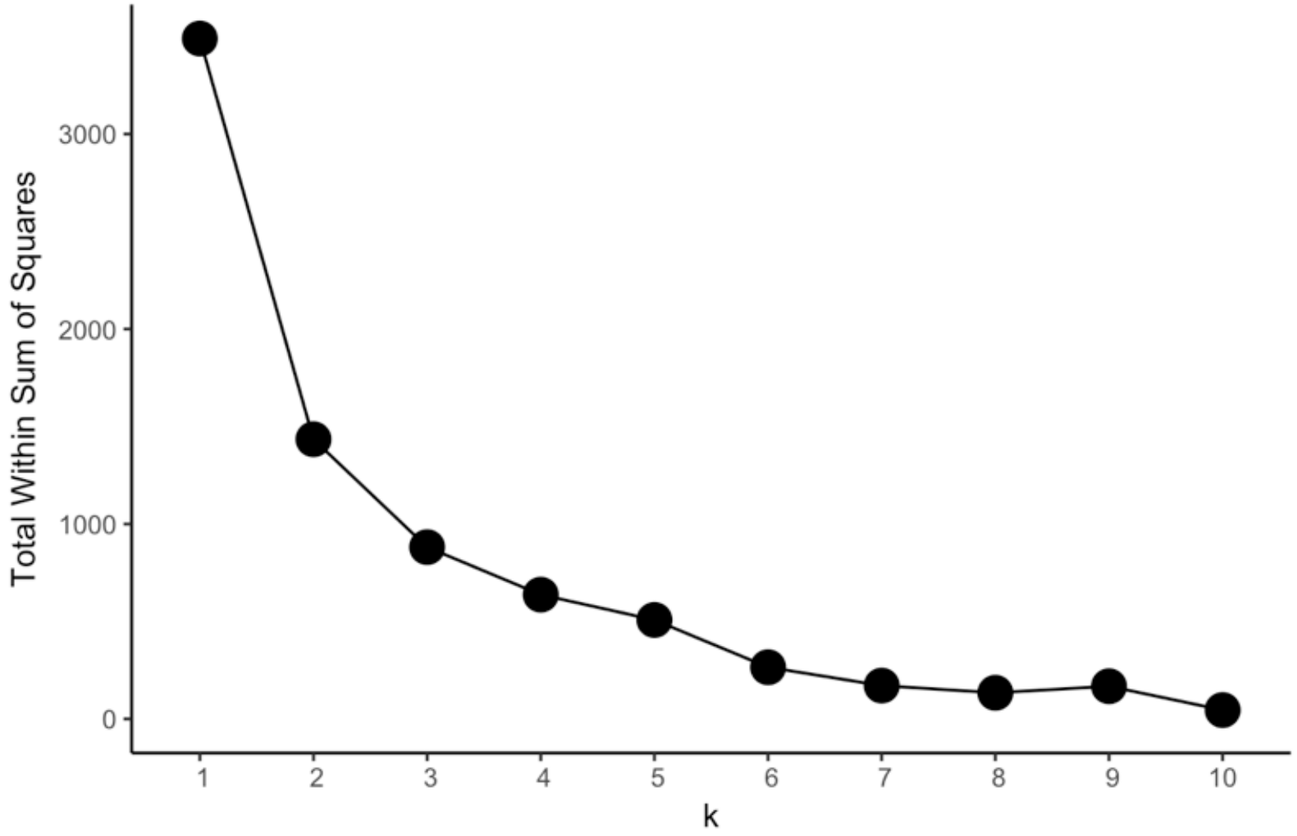


Total within-cluster sum of squares: k = 4

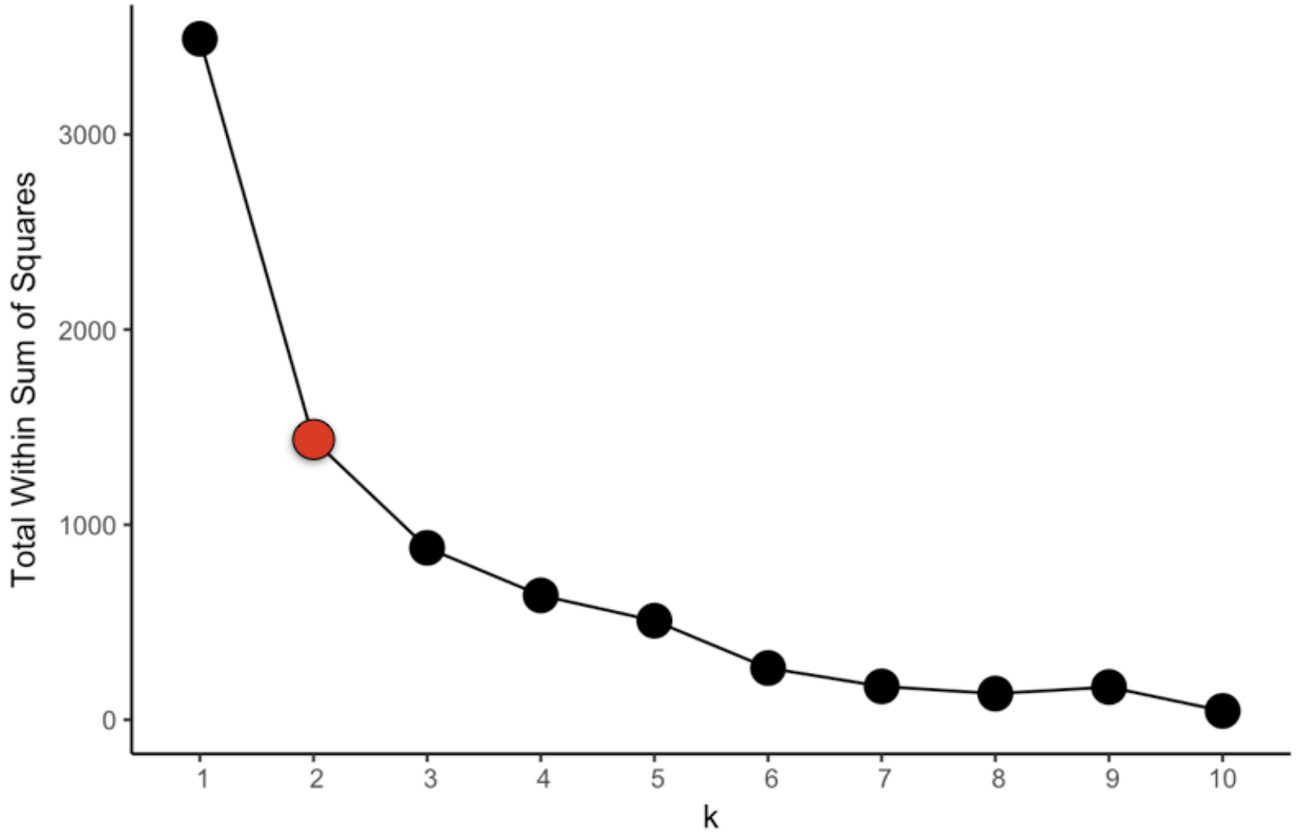




Elbow plot



Elbow plot



Generating the elbow plot

```
model <- kmeans(x = lineup, centers = 2)
model$tot.withinss
```

```
[1] 1434.5
```

Generating the elbow plot

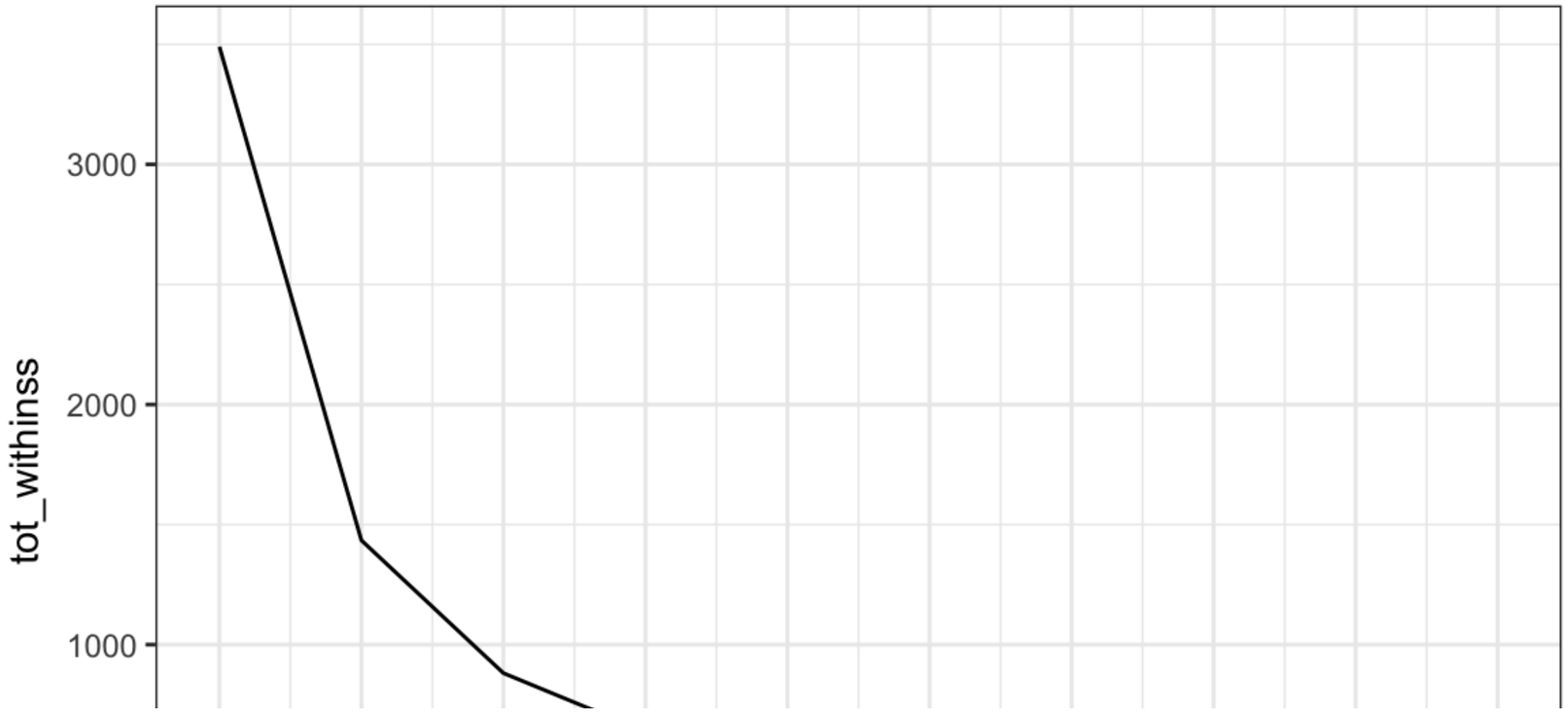
```
library(purrr)

tot_withinss <- map_dbl(1:10,  function(k){
  model <- kmeans(x = lineup, centers = k)
  model$tot.withinss
})
elbow_df <- data.frame(
  k = 1:10,
  tot_withinss = tot_withinss
)
print(elbow_df)
```

```
    k tot_withinss
1   1    3489.9167
2   2    1434.5000
3   3     881.2500
4   4     637.2500
... ...    ...
```

Generating the elbow plot

```
ggplot(elbow_df, aes(x = k, y = tot_withinss)) +
  geom_line() +
  scale_x_continuous(breaks = 1:10)
```



Let's practice!

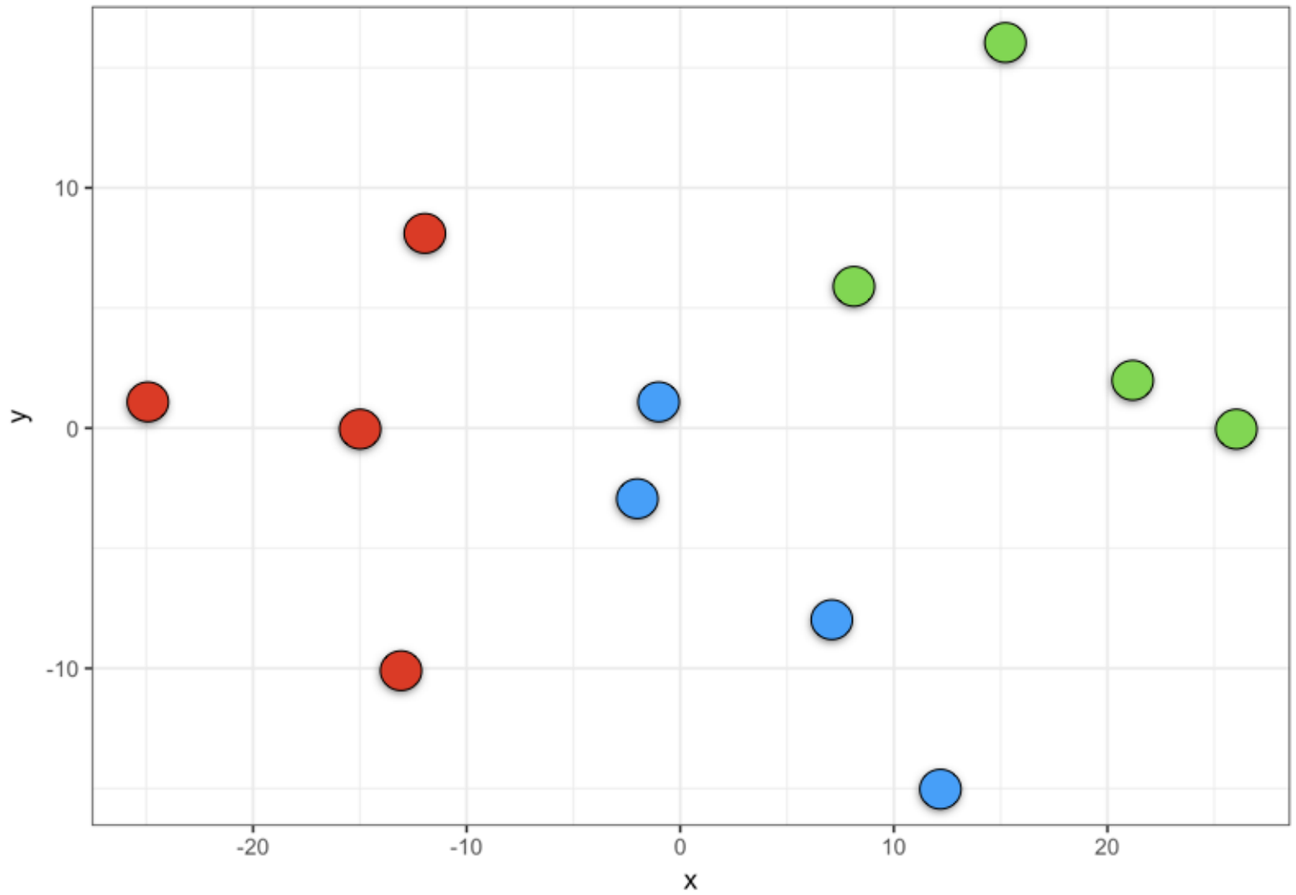
CLUSTER ANALYSIS IN R

Silhouette analysis: observation level performance

CLUSTER ANALYSIS IN R

Dmitriy (Dima) Gorenshteyn
Lead Data Scientist, Memorial Sloan Kettering Cancer Center

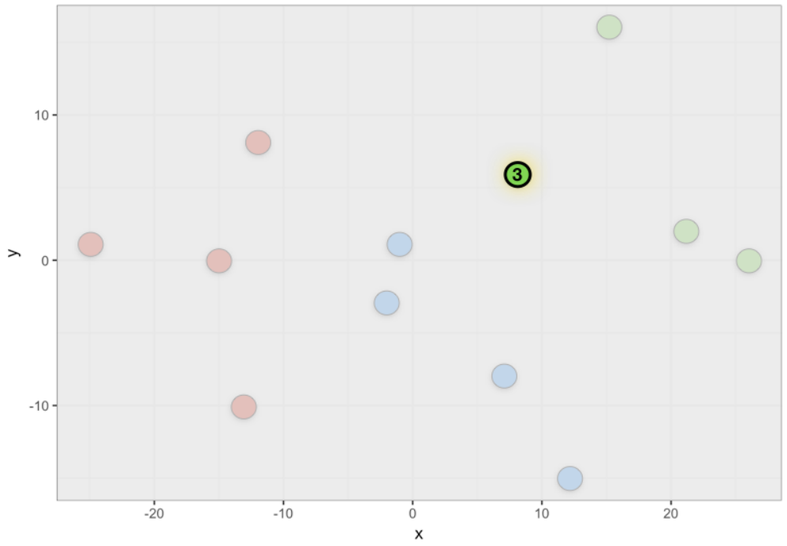
Soccer lineup with K = 3



Silhouette width

Within Cluster Distance: C(i)

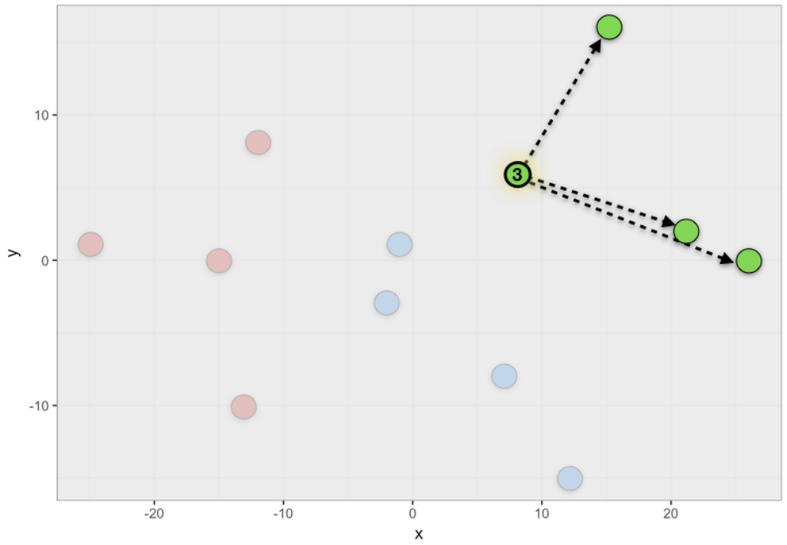
Closest Neighbor Distance: N(i)



Silhouette width

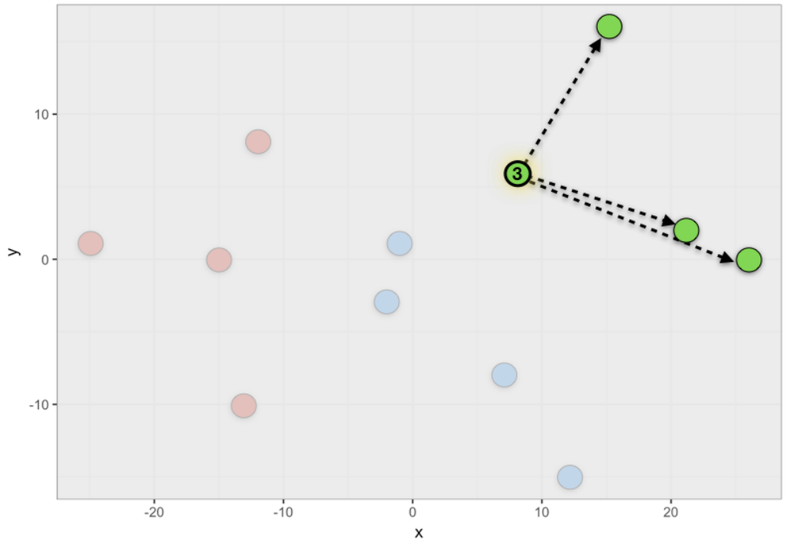
Within Cluster Distance: C(i)

Closest Neighbor Distance: N(i)

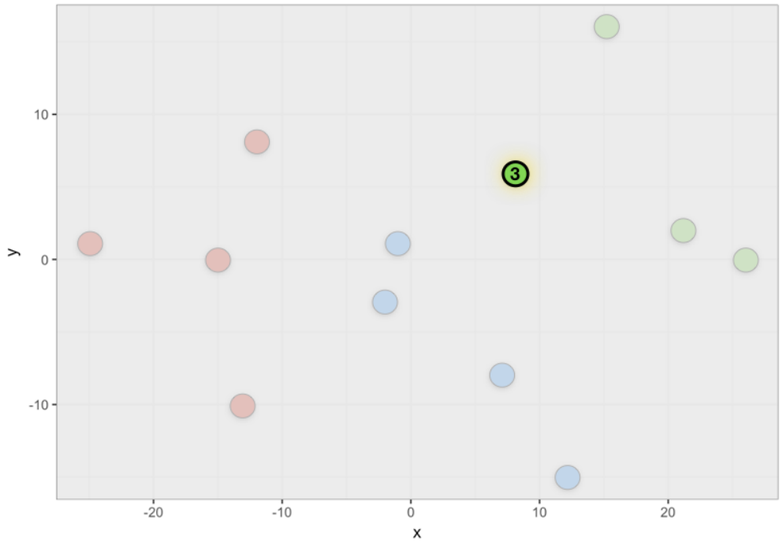


Silhouette width

Within Cluster Distance: C(i)

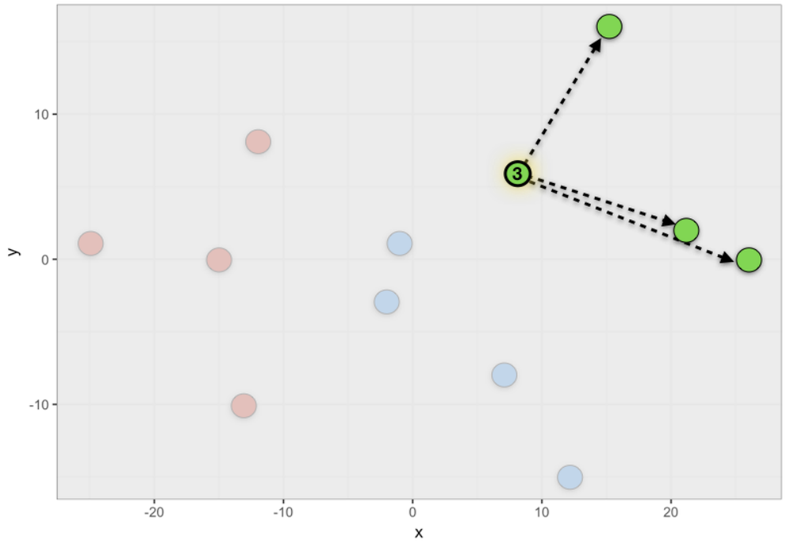


Closest Neighbor Distance: N(i)

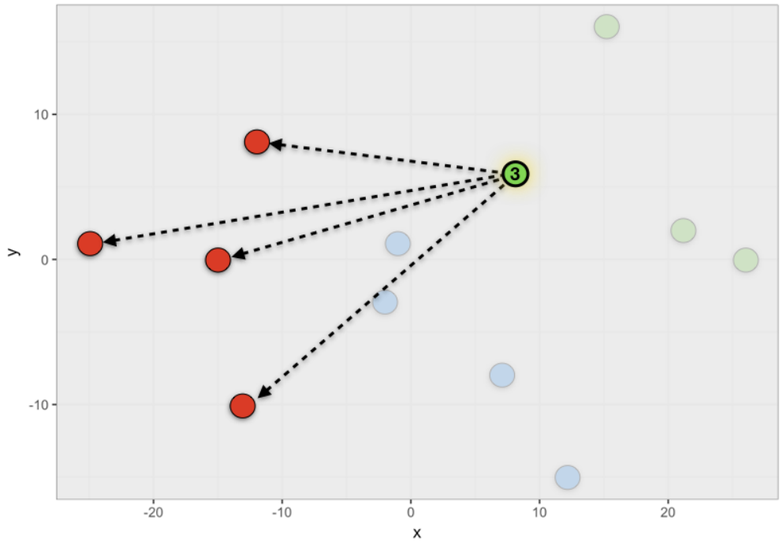


Silhouette width

Within Cluster Distance: C(i)

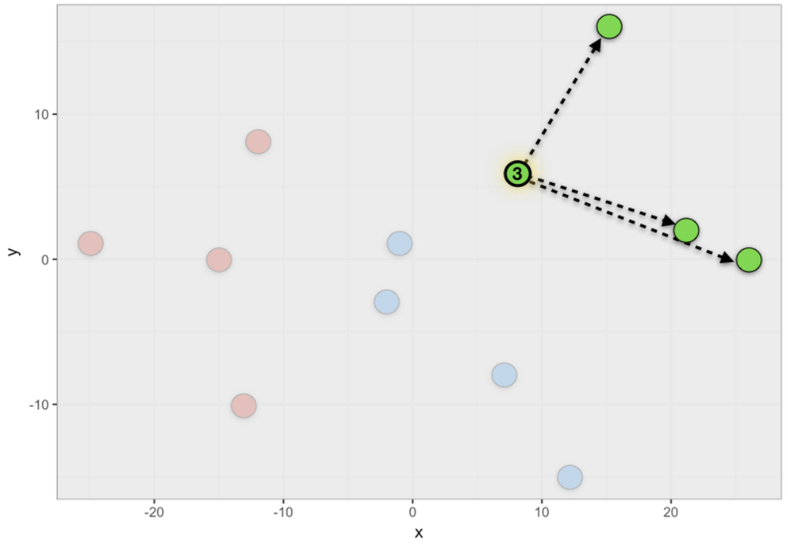


Closest Neighbor Distance: N(i)

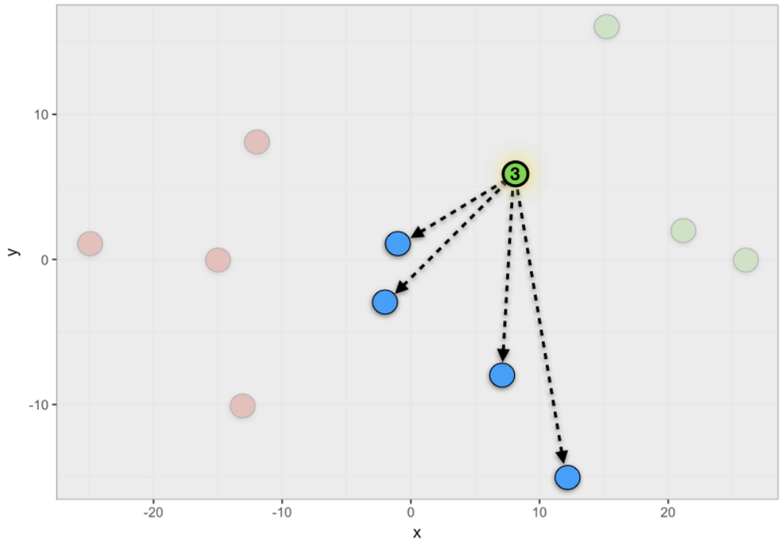


Silhouette width

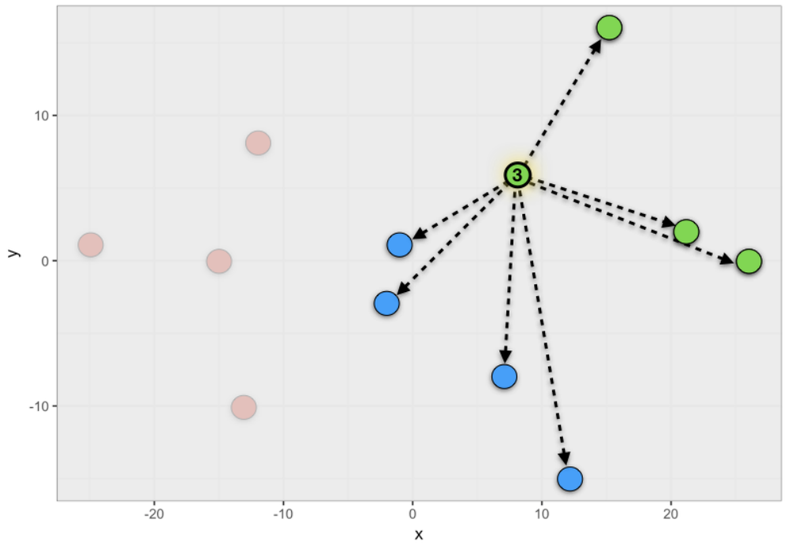
Within Cluster Distance: C(i)



Closest Neighbor Distance: N(i)

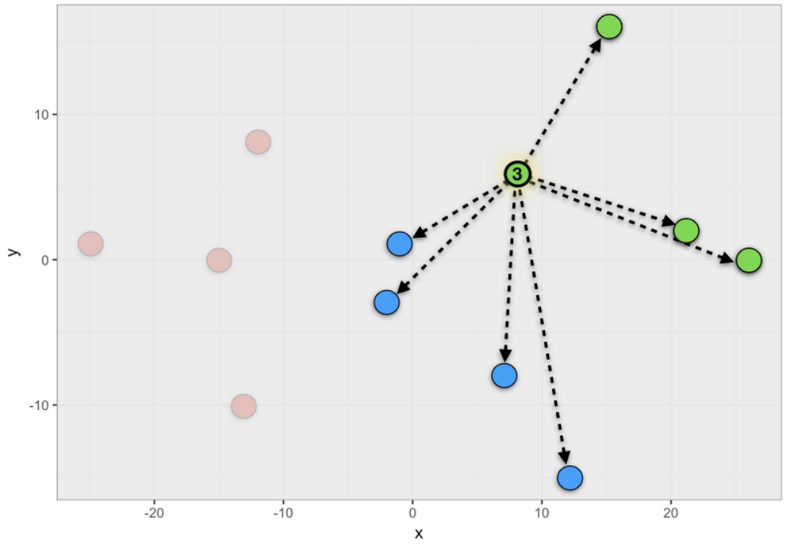


Silhouette width: S(i)



$$s(i) = \begin{cases} 1 - C(i)/N(i), & \text{if } C(i) < N(i) \\ 0, & \text{if } C(i) = N(i) \\ N(i)/C(i) - 1, & \text{if } C(i) > N(i) \end{cases}$$

Silhouette width: S(i)



- **1:** Well matched to cluster
- **0:** On border between two clusters
- **-1:** Better fit in neighboring cluster

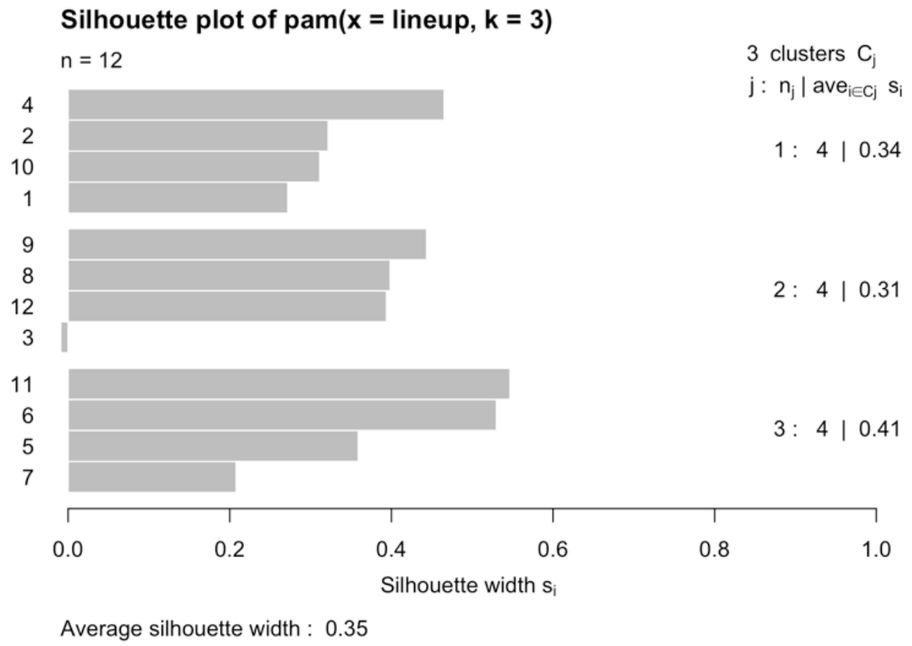
Calculating S(i)

```
library(cluster)
pam_k3 <- pam(lineup, k = 3)
pam_k3$silinfo$widths
```

```
   cluster neighbor    sil_width
4        1        2  0.465320054
2        1        3  0.321729341
10       1        2  0.311385893
1        1        3  0.271890169
9        2        1  0.443606497
...    ...      ...          ...
```

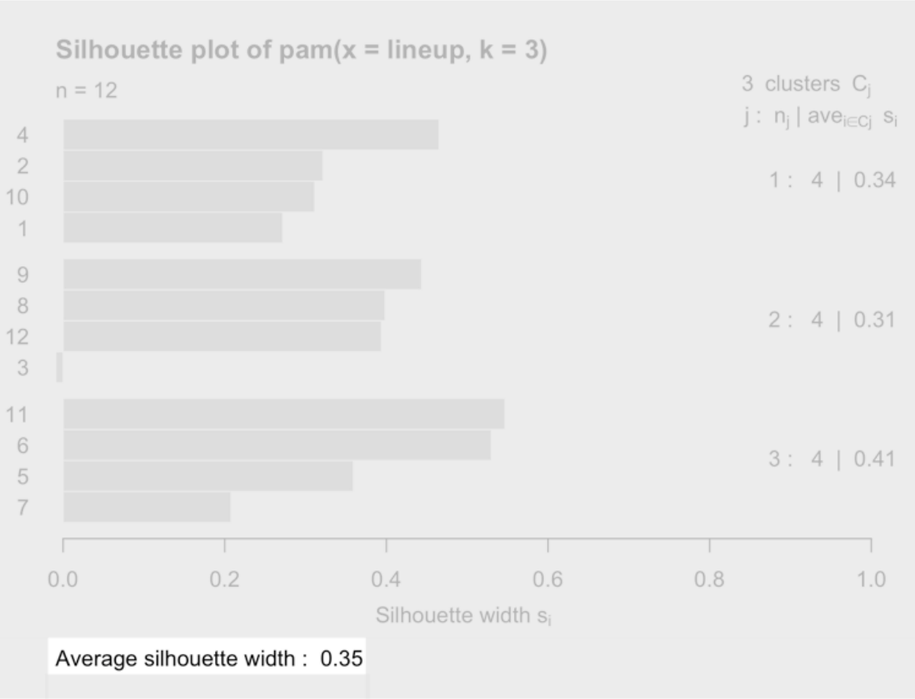
Silhouette plot

```
sil_plot <- silhouette(pam_k3)
plot(sil_plot)
```



Silhouette plot

```
sil_plot <- silhouette(pam_k3)
plot(sil_plot)
```



Average silhouette width

```
pam_k3$silinfo$avg.width
```

```
[1] 0.353414
```

- **1:** Well matched to each cluster
- **0:** On border between clusters
- **-1:** Poorly matched to each cluster

Highest average silhouette width

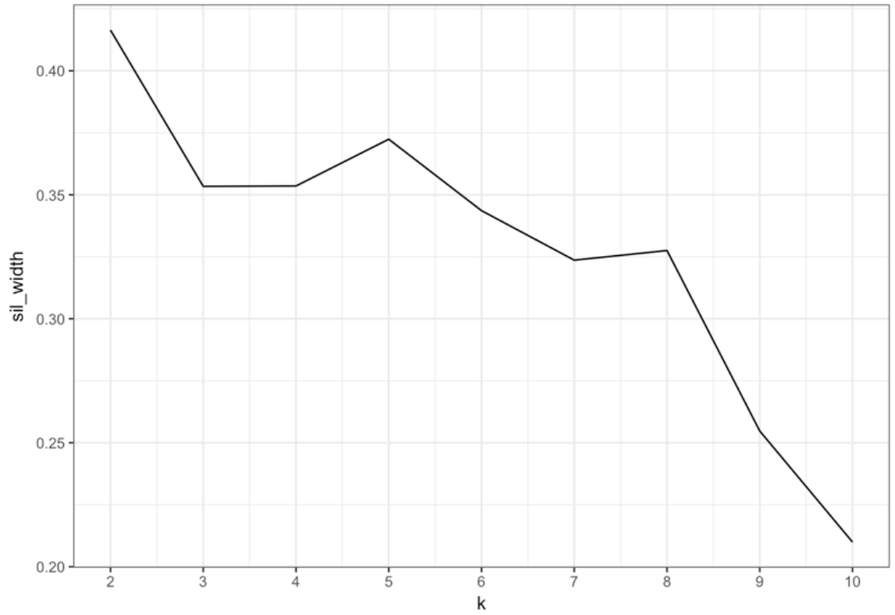
```
library(purrr)

sil_width <- map_dbl(2:10,  function(k){
  model <- pam(x = lineup, k = k)
  model$silinfo$avg.width
})
sil_df <- data.frame(
  k = 2:10,
  sil_width = sil_width
)
print(sil_df)
```

```
    k    sil_width
1   2    0.4164141
2   3    0.3534140
3   4    0.3535534
4   5    0.3724115
...  ...        ...
```

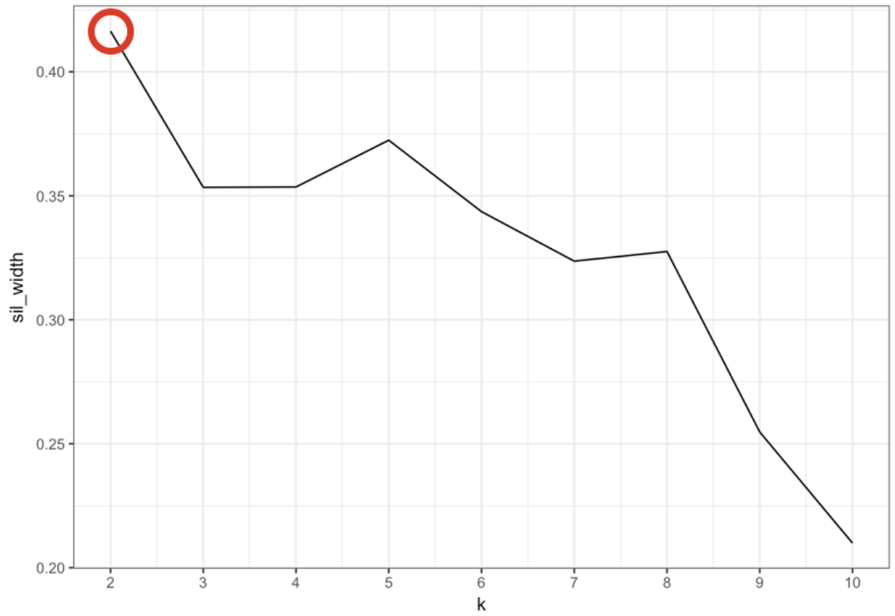
Choosing K using average silhouette width

```
ggplot(sil_df, aes(x = k, y = sil_width)) +
  geom_line() +
  scale_x_continuous(breaks = 2:10)
```



Choosing K using average silhouette width

```
ggplot(sil_df, aes(x = k, y = sil_width)) +
  geom_line() +
  scale_x_continuous(breaks = 2:10)
```



Let's practice!

CLUSTER ANALYSIS IN R

Making sense of the K-means clusters

CLUSTER ANALYSIS IN R

Dmitriy (Dima) Gorenshteyn
Lead Data Scientist, Memorial Sloan Kettering Cancer Center

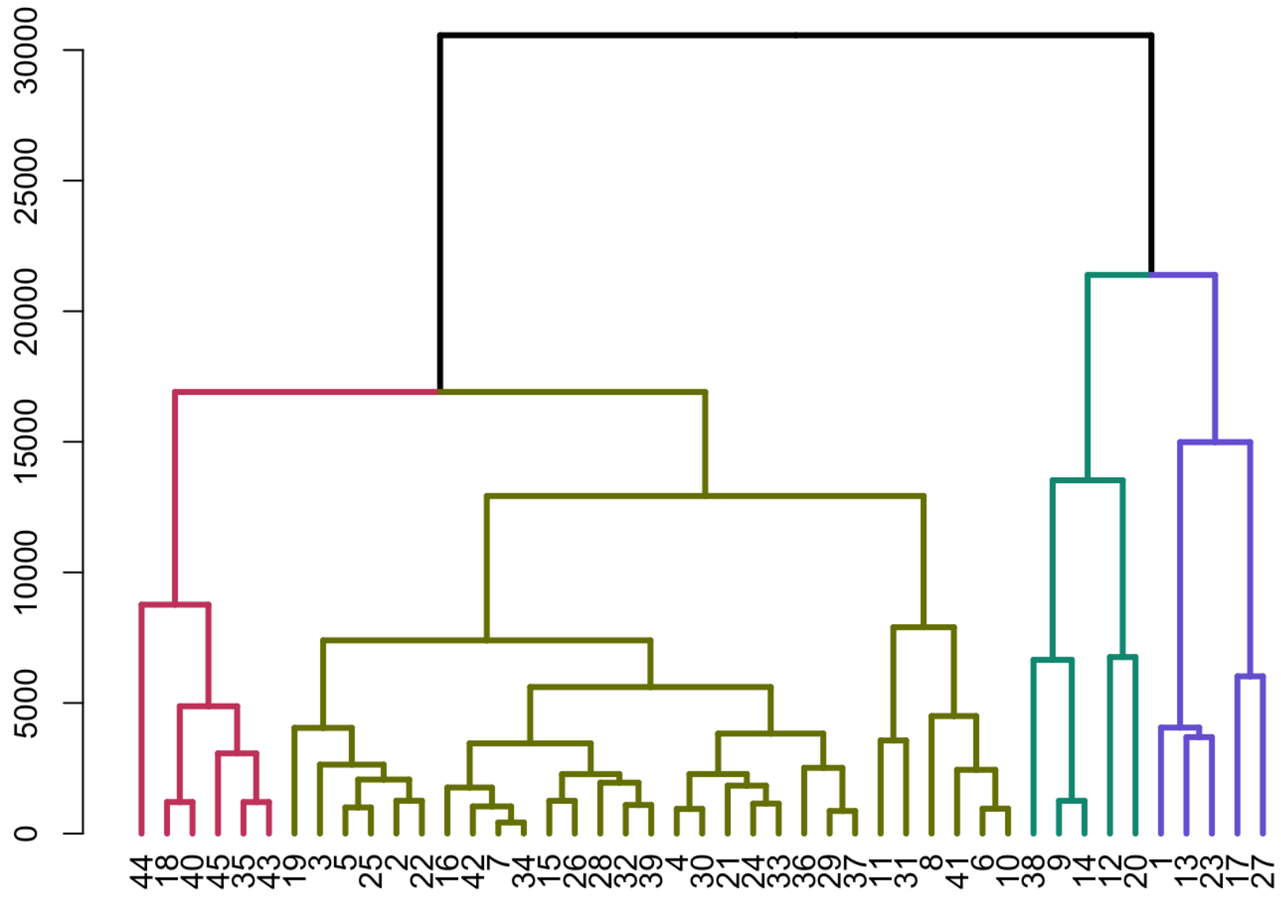
Wholesale dataset

- 45 observations
- 3 features:
  - Milk Spending
  - Grocery Spending
  - Frozen Food Spending

```
print(customers_spend)
```

```
     Milk Grocery Frozen
1   11103   12469    902
2    2013    6550    909
3    1897    5234    417
4    1304    3643   3045
5    3199    6986   1455
...   ...     ...    ...
```

Segmenting with hierarchical clustering



Segmenting with hierarchical clustering

cluster	Milk	Grocery	Frozen	cluster size
1	**16950**	12891	991	5
2	2512	5228	1795	29
3	10452	**22550**	1354	5
4	1249	3916	**10888**	6

Segmenting with K-means

- Estimate the "best" k using average silhouette width
- Run k-means with the suggested k
- Characterize the spending habits of these clusters of customers

Let's cluster!

CLUSTER ANALYSIS IN R