What are advantages and disadvantages of K-means?  
How do you select the optimum value of k(i.e., the number of clusters) in kmeans() function?

Avantages:

1. K-means is a popular clustering algorithm when the data is all numeric and the distance metric is squared Euclidean.
2. It’s easy to implement
3. Can be faster than hierarchical clustering on large datasets.
4. It works best on data that looks like a mixture of Gaussians

Disadvantages:

1. Must pick *k* in advance.
2. This algorithm isn’t guaranteed to have a unique stopping point.
3. K-means can be fairly unstable, in that the final clusters depend on the initial cluster centers.

**THE KMEANS () FUNCTION:**

Its good practice to run k-means several times with different random starts, and then select the clustering with the lowest total WSS. The kmeans() function can do this automatically, though it defaults to only using one random start.

**THE KMEANSRUNS() FUNCTION FOR PICKING K:**

To run kmeans(), you must know *k*. The fpc package (the same package that has clusterboot()) has a function called kmeansruns() that calls kmeans() over a range of *k* and estimates the best *k*. It then returns its pick for the best value of *k*, the output of kmeans() for that value, and a vector of criterion values as a function of *k*. Currently, kmeansruns() has two criteria: the *Calinski-Harabasz Index* ("ch"), and the *average* *silhouette*. It’s a good idea to plot the criterion values over the entire range of *k*, since you may see evidence for a *k* that the algorithm didn’t automatically