Summary

Name Jack Rutherford

	Assigned	Possible
C level	31.5	33.5
B level	5.75	30
A level	0	19
Total	37.25	82.5

Score 68

C-Level

Item	Assigned	Possible	Comment
Statement to dereference $datapointsArrayRef and store in the variable @datapoints is correct in kmeans-C.pl	1	1	
Statement to dereference $clusterCentersArrayRef and store in the variable @clusterCenters is correct in kmeans-C.pl	1	1	
readLabels			
Argument to function is correctly recognized as $_[0]	0	1	Instead of using the file handle passed as a parameter to the subroutine, the program opens the file associated with $ARGV[0] within readLabels
readLabels reads **exactly one line** from the file passed as an argument	1	1	
Line that was read is chomped to eliminate the newline character	1	1	
Double quote characters are correctly eliminated from the line	1	1	
Line correctly split into parts using comma	1	1	
Array returned from split is returned by the function	1	1	
readObservations			
First argument is referenced as $_[0] correctly	1	1	
That argument is dereferenced and stored in an array	1	1	
Loop exists to process each entry of the array	1	1	
Each line of the array is split on the , character	1	1	
Hash is created containing keys **label** and **values**	1	1	
Value associated with key **label** is set to be the last element in the array created by split	1	1	
First n-1 values of array created by split referenced somehow	1	1	
Value associated with key **values** is set to be a reference to the list containing the first n-1 values in array created by split	1	1	
Array of **references** to these hashes is correctly maintained	1	1	
readObservations returns a reference to the array that has been created	1	1	
printObservations			
Arguments are referenced as $_[0] and $_[1] somehow	1	1	
2nd argument is dereferenced as a hash	1	1	
Observation number is printed	0.5	0.5	

C-Level

Tab character printed after observation number	0.5	0.5	
Label printed after tab	0.5	0.5	
Values are printed out in parentheses	0	0.5	The code assumes there will be 4 values in the @valArr array, instead of being generalizable to any data file.
Commas separate each of the values from each other	0.5	0.5	
readAndValidateClusterCenters			
Arguments are referenced as $_[0] and $_[1] somehow	1	1	
First argument is split on comma to create array of index values	1	1	
Second argument is dereferenced and stored as an array	1	1	
Loop exists to process each entry of the array created by split	1	1	
Check exists to make sure that entry is at least 0, and less than the number of observations	1	1	
Invalid index causes return with 1st element of array given a value of 0	0.5	0.5	
Invalid index causes return with 2nd element of array having an error message	0.5	0.5	
Error message indicates the value that was invalid	0	0.5	This would work, except that += is used rather than .= += is a numeric operator, and so $EM += "error message" assigns a numeric value of 0 to $EM.
Error message indicates the range of legal values	0.5	0.5	
Invalid index causes return with 3rd element of array being a reference to an empty array	0.5	0.5	
When all values are valid, 1st element of returned list is 1	0.5	0.5	
When all values are valid, 2nd element of returned list is the empty string	0.5	0.5	
When all values are valid, 3rd element of returned list reference to the list of cluster center indices	1	1	
If an index is specified more than once, the function returns false in the first element of the returned list	1	1	

If an index is specified more than once, the function returns error message that includes the index that was specified more than once	0.5	0.5
Total	31.5	33.5

Other comments

B-Level

Item	Assigned	Possible	Comment
Command line argument handling			
Call to GetOptions it made with correct options specified	1	2	The help option should not be negatable (leave out the ! At the end of the argument name) centers, groups, and datapoints arguments are specified twice, once as integers, and once as negatable options.
References to scalar variables are passed to GetOptions to allow it to store values into the appropriate variables	1	1	
Variable to hold --datapoints argument value is initialized to 0	0.5	0.5	
Variable to hold --centers argument value is initialized to 0	0.5	0.5	
Variable to hold --groups argument value is initialized to 1	0	0.5	It's initialized to -1, so it's required that a value be specified in order for groups to be shown.
Correct help message is printed when --help argument is present	0.25	0.5	Yes, although there's also a message "help called" that should not be printed
Program exits with a status of 0 if --help argument is present	0.5	0.5	
Program enforces restriction that at least one of --datapoints, --center, and --groups must be specified	1	1	Variables named $flag are confusing to a reader unfamiliar with the program. Try to come up with something more descriptive of what the variable means.
printGroups			
Reference to hash in $_[0] is dereferenced	1	1	
Loop exists to iterate over the elements of the hash	0	1	
Label printed correctly	0	0.5	
Observations in the group accessed correctly	0	0.5	
Observation reference dereferenced to get list of observations	0	1	
Number of observations printed after the observation label	0	1	

B-Level

Observations in the group iterated over correctly	0	1
printObservation called on each observation in the group	0	0.5
Correct arguments are passed to printObservation	0	1
Call to computeAverages made, passing the reference to the array of observations	0	1
Results of calling computeAverages printed correctly	0	1
computeAverages		
Argument deferenced as an array correctly	0	1
Running total by index created correctly	0	2
Running totals divided by number of observations correctly	0	1
Reference to array containing totals is returned	0	1
createGroupsByLabel		
Argument deferenced as an array correctly	0	1
Empty hash created	0	1
Observations in the array iterated over correctly	0	1
Label associated with observation referenced correctly	0	0.5
Array reference containing only the current observation created if this label doesn't exist in the hash	0	1
Current observation added to existing array if this label has been seen before	0	1
Reference to the hash is returned by the subroutine	0	0.5
kmeans-B.pl		
if statement added to only display the data points if variable maintained by --datapoints has non-zero value	0	1
if statement added to only display the cluster centers if variable maintained by --centers has non-zero value	0	1
if statement added to only display the groups if variable maintained by --groups has non-zero value	0	1
Total	5.75	30

Other comments

A-level

Item	Assigned	Possible	Comment
Command line argument handling			
--iterations option correctly added to list of possible options	0	1	
Value of variable associated with --iterations is initialized to 100	0	1	
Error message printed if value given to --iterations argument is 0	0	0.5	
Error message printed if value given to --iterations argument is negative	0	0.5	
Default value of --groups changed to 0	0	0.5	
Requirement for at least 1 of --datapoints, --centers, and --groups removed	0	0.5	
Cluster center initialization			
Data structure to hold cluster centers initialized (could be array, could be hash)	0	0.5	
Array containing initial cluster center indices iterated over	0	1	
Observation at current index correctly accessed	0	0.5	
Observation set as the center of a cluster	0	1	
Iterations			
Loop set to run the correctly number of times	0	1	
Each observation iterated over within the loop	0	1	
Distance to each cluster center computed correctly	0	1	
Minimum distance computed correctly	0	1	
Observation placed into correct group based on distance	0	1	
Cluster centers correctly recomputed at the end of the iteration	0	1	
Output			
Loop exists to iterate over the groups	0	1	
Cluster and number is printed for each group	0	1	
For each group, a count of the number of observations for each label is computed	0	2	
Labels / counts are sorted in **descending order** based on the number of observations in the group with the associated label	0	1	
Each label / count pair is printed out correctly	0	1	
Total	0	19	

Other comments