**HeatmapDataGenerator PDF Processing**

* The PDF process in the HeatmapDataGenerator (HMDG) is driven off of the –PDF runtime parameter. If the param is present, a PDF will be generated.
* The first step in the PDF generation process is to load an array, (s data tiles are being written. HMDG.writeTileFile. The array, pdfMatrices , is stored on the ImportData object. One matrix for each flick layer. Currently, this process loads SUMMARY data into the matrix. If we want to display a detail image, the matrix will need to be filled with detail values by changing the conditional in this method.
* Also during the processing of matrix data, a 3 points per leaf dendrogram matrix is filled and stored on the RowColData object. This matrix is loaded everytime and does not require the –PDF parameter because it is also used to generate thumbnail images. Later this will be used in the PDF process to create dendrogram images for the PDF.
* Processing of the PDF itself begins after the tiles have been created and is called in the main HMDG method, processHeatmap.
* First the method, createSummaryImage, is called and this will construct a buffered RGB image using the pdfMatrices matrix for each layer. Colors are blended and the image is constructed. Once done, the image is added to the ImportData object.
* For every covariate bar on the map, the InputClass.createClassSummaryImg will be called. This method creates a matrix of values for the covariate bar that is the length of the bar and the height specified by the map config. This is done by calling the method, InputClass.generateClassMatrix and the resultant matrix is stored on the InputClass object in the variable classMatrix. Then a buffered image RGB image is created for the covariate bar and stored on the InputClass object as the variable classImage. Again, this is done using SUMMARY level data.
* For every covariate bar on the map, the InputClass.createClassLegendImg will be called. A buffered image RGB image is created for the covariate bar legend and stored on the InputClass object as the variable classLegend.
* For both axes (where applicable), the HMDG methods createRowDendroImg and createColDendroImg methods. These methods will utilize the dendrogramMatrix stored on the RowColData object to generate buffered RGB images for each matrix. These images are stored on their respective RowColData objects in the variable, dendroImage.
* After all of the above “preparatory” steps have been performed, the createHeatmapPdf method in the class pdfGenerator is finally called. This class will assemble the PDF from all of the images that have been created in prior steps.
* The process is broken into 2 primary steps that generate the PDF page creatPDFHeatmapPage and the PDF legend pages createPDFLegendPage.
* In the createPDFHeatmapPage method, the page is assembled by calling the drawing of component parts in a VERY specific order of execution:
  + Column Dendrogram (drawColumnDendrogram)
  + Column Covariates (drawColumnCovariates)
  + Row Dendrogram (drawRowDendrogram)
  + Row Covariates (drawRowCovariates)
  + Heatmap Image (done within createPDFHeatmapPage)
  + Row Top Items (drawTopRowItems)
  + Column Top Items (drawTopColumnItems)
* It is important to note that the methods drawRowCovariates and drawColumnCovariates ALSO draw labels for the covariate bars as they place the covariate bars on the page.
* In the createPDFLegendPage method, again elements are drawn in specific order. First row covariates, then columnCovariates. There is a lot of extra logic in these methods to handle page breaks and the application of legend text to the legend page.