These are the basic steps for preparing your own meta-analysis. See our meta-analysis of lung cancer subtypes for a template.

1. Create table of study metadata, with a row for each study. See luca_main_study_metadata.csv for an example. At a minimum, the following columns must exist: study, studyDataType, and platformInfo.
   - study: Name of the study, which must be unique.
   - studyDataType: Indicates how the expression data is stored. See below for details.
   - platformInfo: Microarray platform, used for mapping probes to genes. See below for details.

   There are currently five options for studyDataType: affy_geo, affy_custom, affy_series_matrix, series_matrix, and eset_rds.
   - affy_geo: Raw Affymetrix data from a GEO study.
   - affy_custom: Raw Affymetrix data from a non-GEO study.
   - affy_series_matrix: Normalized but untransformed, probe-level Affymetrix data in a GEO series matrix file.
   - series_matrix: Normalized, log-transformed (or equivalent) data in a GEO series matrix file.
   - eset_rds: Normalized, log-transformed (or equivalent) data, already mapped to Entrez Gene IDs, saved as an RDS file.

   The options for platformInfo depend on the studyDataType.
   - If studyDataType is affy_geo, affy_custom, or affy_series_matrix, then platformInfo should be the name of the corresponding custom CDF from BrainArray.
   - If studyDataType is series_matrix, then platformInfo should be the corresponding GPL identifier.
   - If studyDataType is eset_rds, then platformInfo should be "ready".

   The table of study metadata can also contain columns indicating which studies should be used for discovery and which for validation. There are many ways to do this.

2. For each study, download the expression data. The form of the expression data will depend on the studyDataType.
   - All the expression data should go in the same folder.
   - If studyDataType is affy_geo or affy_custom, the expression data should be a folder containing cel or cel.gz files. The name of the folder should match the name of the study.
   - If studyDataType is affy_series_matrix or series_matrix, the expression data should be a file ending in "_series_matrix.txt". The part of the file name prior to "_series_matrix.txt" should match the name of the study.

- If studyDataType is eset_rds, the expression data should be an RDS file containing a Bioconductor ExpressionSet. The part of the file name prior to ".rds" should match the name of the study.

3. Download all files and install all packages necessary for mapping probes to genes and for conducting a meta-analysis.
   - Open R and set the working directory to the folder that contains the code for meta-analysis. Execute the following command.
   - > source('metaAnalyzeInstall.R')

   For each study with studyDataType of affy_geo or affy_custom, download the corresponding custom CDF package.
   - Go to http://brainarray.mbni.med.umich.edu/Brainarray/Database/CustomCDF/CDF_download.asp.
   - Click on the link for the latest version of "ENTREZG".
   - Download the appropriate R source package(s), which should end in "cdf_nn.0.0.tar.gz", where nn refers to the version number.
   - Install the custom CDF packages.
   - Open R and set the working directory to the folder containing the downloaded custom CDF packages. Execute the following command for each custom CDF.
   - > install.packages('file name of custom CDF', repos=NULL, type='source')
   - Make sure that the part of the file name prior to "nn.0.0.tar.gz" matches the information in the platformInfo column.

   For each study wth studyDataType of affy_series_matrix, download the corresponding custom CDF zip file.
   - Go to http://brainarray.mbni.med.umich.edu/Brainarray/Database/CustomCDF/CDF_download.asp.
   - Click on the link for the latest version of "ENTREZG".
   - Download the appropriate zip file(s) (under column "CDF Seq, Map, Desc").
   - Unzip each zip file.
   - Move the file that ends in "_mapping.txt" to the folder that contains the downloaded expression data.
   - Make sure that the part of the file name that occurs prior to "_mapping.txt" matches the information in the platformInfo column.

4. If necessary, extend the getStudyData function to support all the microarray platforms in your meta-analysis.
   - Open R and set the working directory to the folder that contains the code for meta-analysis. Execute the following commands.
   - > source('metaAnalyze.R')
   - > getSupportedPlatforms()

- For each study that has a studyDataType of series_matrix, check that the corresponding platformInfo is in the list of supported platforms.
- If any of your platforms are not supported, you will need to edit the getStudyData function in metaAnalyze.R.
    - You need to add another "else if" statement (see line 130 or so) that lets the function map probes to genes for your platform.
    - Look at the "else if" statements for supported platforms to see examples of how this is done, but the specifics will depend on the platform.
    - Finally, add the microarray platform to the character vector of supported platforms in the getSupportedPlatforms function (line 80 in metaAnalyze.R).

5. Create table of sample metadata, with a row for each sample. See luca_sample_metadata.csv for an example.
    - At a minimum, the following columns must exist: study, sample, and a column for outcome or class.
    - The values in the study column should match the values in the study column in the table of study metadata.
    - For studies with a studyDataType of affy_geo or affy_custom, the sample names should be the names of the cel or cel.gz files (excluding the file extension).
    - For studies with a studyDataType of affy_series_matrix or series_matrix, the sample names should be the GSM identifiers.
    - For studies with a studyDataType of eset_rds, the samples names should be the colnames of the corresponding ExpressionSet.
    - The name of each sample must be unique across all the samples.

6. Adapt the code in luca_main_metaAnalyze.R to perform your meta-analysis. The details will depend on your meta-analysis. The main steps are:
    - Load the study and sample metadata.
    - Load the expression data and perform intra-study normalization.
    - Merge discovery datasets and perform cross-study normalization.
    - Run cross-validation.
    - Predict outcome/class for samples in validation datasets.
    - Generate plots and tables.