Changes in execute.sh

1. Added few input parameters which can be passed onto deploy.sh for creation of new variables
2. gcs\_root – path of google cloud storage bucket and folder where the code is located (gs://bucketname/../foldername), foldername is equivalent to modelname
3. cluster\_name – Dataproc cluster where the oozie job is launched
4. hdfs\_dir – path of hdfs location where the xml’s are supposed to be pushed. (hdfs://…….:8020/{hdfs\_dir})

Changes in deploy.sh

1. Few parameters are hardcoded as this file doesn’t when associated with a model
2. Data\_project – name of the GCP project where the BQ datasets reside
3. Current\_project – Name of the current GCP project where the dataproc cluster resides
4. nameNode – is parameterized by taking the input of cluster\_name from the execute.sh
   1. hdfs://${cluster\_name}:8020 – as the hdfs and port remain same
5. hdfs\_proj – path to hdfs directory attached with cluster where the xmls are stored
   1. hdfs://${cluster\_name}:8020/${hdfs\_dir}
6. In the if else loop, we are testing for the existence of files, if yes then we are deleting the xmls and then we are reuploading the xml’s from local(runtime) to ${hdfs\_proj}
7. workflowRoot is assigned the value attached to hdfs\_proj as both are same.
8. While launching oozie job parameterized inputs are used
   1. cluster\_name
   2. nameNode
   3. workflowRoot
   4. gcs\_root
   5. current\_project
   6. data\_project
9. New/Changed parameters while launching oozie job
   1. Gcs\_root – folder location in a bucket where code resides
   2. Workflowroot – to provide hdfs location of the xml files
   3. Current\_project – Project name where the oozie job is running
   4. Data\_project – project name where the Big Query datasets reside

Alert: Apart from the changes described above rest of the inputs or the code structure remains same as the older one.