IAAP-CLI

Overview

IAAP-CLI is a command line interface application intended for cross-platform data analysis of Illumina microarray data. Currently it only supports the genotyping (gencall) command which converts raw intensity files (IDATs) to multiple formats with genotypes (e.g. GTC).

Prerequisites

There may be additional shared libraries that need to be installed on your system related to the .NET core runtime for Linux. For any installation problems see the **Troubleshooting** section or contact Illumina support.

Installation

1. The un-tarred package should look something like this:

```
1 |-- iaap-cli
2 |    |-- lots of .dll files
3 |    |-- iaap-cli <-- executable
4 |-- licenses.txt
5 |-- README.md
6 |-- README.pdf
```

2. **(Optional)** Add the installation directory to your $PATH environmental variable. Example for bash:

```
1 echo "export PATH=$PATH:/path/to/iaap-cli-linux-x64-${VERSION}" >> ~/.bash_profile
2 source ~/.bash_profile
```

Examples

- **View help for the `gencall` command**

```
1 iaap-cli gencall -h
```

- **Convert some GSA v1 raw intensity samples (IDAT files) to GTC files:**

```
1 iaap-cli gencall /path/to/GSA-24v1-0_A1.bpm /path/to/GSA-24v1-0_A1_ClusterFile.egt
    /path/to/my_output -f /path/to/GSA_idats -g
```

- **Convert some GSA v1 raw intensity samples (IDAT files) to GTC files using a Sample Sheet:**

```
1 iaap-cli gencall /path/to/GSA-24v1-0_A1.bpm /path/to/GSA-24v1-0_A1_ClusterFile.egt
    /path/to/my_output -s /path/to/GSA_SampleSheet.csv -g
```

- **Convert some GSA v1 raw intensity samples (IDAT files) to GTC files using 4 parallel threads:**

```
1 iaap-cli gencall /path/to/GSA-24v1-0_A1.bpm /path/to/GSA-24v1-0_A1_ClusterFile.egt
    /path/to/my_output -f /path/to/GSA_idats -g -t 4
```

- **Convert some GSA v1 raw intensity samples (IDAT files) to PED files:**

```
1 iaap-cli gencall /path/to/GSA-24v1-0_A1.bpm /path/to/GSA-24v1-0_A1_ClusterFile.egt
    /path/to/my_output -f /path/to/GSA_idats -p
```

- **Convert some GSA v1 raw intensity samples (IDAT files) to GTC files using an include list of loci:**

```
1 iaap-cli gencall /path/to/GSA-24v1-0_A1.bpm /path/to/GSA-24v1-0_A1_ClusterFile.egt
    /path/to/my_output -f /path/to/GSA_idats -g -inc /path/to/include_list.csv
```

Sample Sheet

IAAP v1.1 adds the "sampleSheetPath" parameter as an alternative input to the "idatFolder" parameter. This parameter should be the full path location to a sample sheet file. The sample sheet should be a comma-separated (csv) file and is intended to be backwards-compatible with existing GenomeStudio-based sample sheets. However, only the [Data] section is used in IAAP and the [Header] and [Manifests] sections will be ignored. An example sample sheet is shown below:

```
1 [Data]
2 Sample_ID,SentrixBarcode_A,SentrixPosition_A,Path
3 SampleA,9348171001,R01C02,/full/path/to/idats/A/9348171001
4 SampleB,9348171002,R01C01,/full/path/to/idats/B/9348171002
```

Currently the `Sample_ID,SentrixBarcode_A,SentrixPosition_A,Path` columns are required. Extra columns such as `Aux` or `Gender` will not result in a parsing failure. If the sample sheet was generated in GenomeStudio, the `Path` must be mapped from the Windows paths to their Unix paths.

Include File

IAAP v1.1 adds the "includeFilePath" optional parameter to enable analysis on only the "Markers of Interest" from a product. It will allow filtering the SNPs/loci reported in output files (e.g GTC and PED) to only be those that appear in the "include file". the file path should be the full path location to a csv file where each row lists a unique LociName that appears in the product's manifest as shown below: e.g.

```
1 rs1234
2 rs1235
3 rs1236
```

There are a couple of additional outcomes possible when passing an include file parameter compared to when doing a regular analysis. - If the file includes loci that do not appear in the manifest, the analysis will continue but the application will output a file called badLoci.txt. The output file contains any loci not found. - The application will output an extra manifest (bpm) that is a subset/filtered version of the original manifest based on the loci in the include list (e.g. GSA-24v1-0_A1_filtered_read_only.bpm). This filtered manifest is what should be used during secondary analysis of the filtered GTC file (e.g with the [BeadArrayFiles] (https://github.com/Illumina/BeadArrayFiles) or GTCtoVCF libraries), instead of the full manifest.

A MAJOR CAVEAT FOR THE "FILTERED" MANIFEST: This manifest is NOT intended to be re-used for primary analysis (ie, generation of GTC files, loading projects in GenomeStudio, etc.). The filtered manifest is to be used only for secondary analysis! Because of some details

associated with the GenTrain algorithm, GTC files produced using the full manifest vs. the filtered manifest could result in discordance between base calls for the same samples.

Troubleshooting

Missing icu library

```
1 $ ./iaap-cli
2 FailFast: Couldn't find a valid ICU package installed on the system. Set the
     configuration flag System.Globalization.Invariant to true if you want to run with no
     globalization support.
3
4    at System.Environment.FailFast(System.String)
5    at System.Globalization.GlobalizationMode.GetGlobalizationInvariantMode()
6    at System.Globalization.GlobalizationMode..cctor()
7    at System.Globalization.CultureData.CreateCultureWithInvariantData()
8    at System.Globalization.CultureData.get_Invariant()
9    at System.Globalization.CultureData.GetCultureData(System.String, Boolean)
10   at System.Globalization.CultureInfo.InitializeFromName(System.String, Boolean)
11   at System.Globalization.CultureInfo.Init()
12   at System.Globalization.CultureInfo..cctor()
13   at System.StringComparer..cctor()
14   at
       System.AppDomainSetup.SetCompatibilitySwitches(System.Collections.Generic.IEnumerable'1<System.St
15   at System.AppDomain.PrepareDataForSetup(System.String, System.AppDomainSetup,
       System.String[], System.String[])
16 Aborted (core dumped)
```

Solution Install `icu` examples: - For Ubuntu run, `apt-get install icu-devtools` - For CentOS/RHEL run, `yum install libicu`

Or, to avoid installation of `icu`, you can disable globalization support for .NET core via setting the evironmental variable `DOTNET_SYSTEM_GLOBALIZATION_INVARIANT=true`.

Out of memory exception

```
1 Killed.
```

Solution This error is the result of the "OOM killer" of the linux kernel killing iaap-cli because it requested too much memory from the system. This can occur if `--num-threads` is set to high for your system or the system does not have enough memory to analyze the product. Reduce the `--num-threads` or remove the option to default to 1 thread.