**SIVIC Scripting Tutorial**

HMTRC Workshop - March 23-24, 2017

Department of Radiology and Biomedical Imaging, UCSF

Supported by NIBIB P41EB013598

**Goal:** The purpose of this tutorial is to introduce you to the SIVIC command line tools for batch processing HP 13C data. Additional tutorials from this workshop will cover performing similar analysis using the SIVIC GUI. The SIVIC GUI and other visualization tools will be used to display intermediate results.

**Introduction:** In this example you will process a 2D 13C dynamic spectroscopy rat data set acquired on a GE 3T scanner. This data was acquired with an EPSI sequence using a dual-band variable flip angle pulse scheme. You will reconstruct the data, combine the EPSI lobes, combine the coils and then correct for the dual-band and variable-flip pulse profiles using a Data Acquisition Description (DAD) file produced for the sequence. Then you will generate metabolite maps for pyruvate and lactate at each time point and also manually create an ROI mask of the animal. These results will be saved as DICOM images and used as input to a kinetic modeling program that will fit the observed dynamic signal changes to a 2-site exchange model within the ROI and generate 3D maps of Kpl and T1all.

1. **Downloading SIVIC and sample data:**

Open a terminal and run the following commands:

***LINUX/OSX Terminal:>***

mkdir hmtrc\_2017

cd hmtrc\_2017

From a web browser download the sample data from the following URL and save the zip file into your hmtrc\_2017 folder:

<http://sourceforge.net/projects/sivic/files/sample_data/HMTRC_2016/variable_flip_c13_dynamic.zip/download>

***LINUX/OSX Terminal:>***

uunzip variable\_flip\_c13\_dynamic.zip

**MAC OSX USERS**:

Mac OSX users should use a web browser to download and install the SIVIC.dmg from the following URL:

<https://sourceforge.net/projects/sivic/files/0.9.55/SIVIC.dmg/download>

**LINUX USERS**:

***LINUX Terminal:>***

wget <https://sourceforge.net/projects/sivic/files/0.9.55/sivic_0.9.55_Linux_x86_64.tar.gz/download> -O sivic.tar.gz

gunzip sivic.tar.gz

tar –xvf sivic.tar

1. **Preparing your environment:**

The SIVIC package includes a number of command line tools that will be used. These can be found in the bin directory. This resides in different locations depending on your platform:

***LINUX Terminal:>***

ls sivic\_0.9.55\_Linux\_x86\_64/local/bin/

***Mac OSX Terminal:>***

ls /Applications/SIVIC.app/Contents/sivic/local/bin/

***Output:>***

sivic

svk\_apodize

svk\_auto\_phase

svk\_average\_spec

svk\_combine\_spec

svk\_dcmdump

svk\_extract\_spec

svk\_fft

svk\_file\_convert

svk\_gepfile\_anon

svk\_gepfile\_reader

svk\_hsvd

svk\_image\_mathematics

svk\_image\_pipeline

svk\_image\_stats

svk\_image\_threshold

svk\_integrate\_dynamic

svk\_interpolate\_spectra

svk\_lcmodel\_reader

svk\_lcmodel\_writer

svk\_met\_kinetics

svk\_mrs\_combine

svk\_multi\_view

svk\_noise

svk\_peak\_pick

svk\_phase\_spec

svk\_point\_selector

svk\_psd\_prescription\_convert

svk\_quantify

svk\_quick\_view

svk\_reorder\_epsi

svk\_reslice

svk\_scale\_image

svk\_spec\_diff

svk\_transform

svk\_variable\_flip\_scaler

svk\_volume\_diff

svk\_zerofill

svk\_zscore

To make these tools available you will need to add the install location to your PATH variable.

***LINUX Terminal:>***

export PATH=/$PWD/sivic\_0.9.55\_Linux\_x86\_64/local/bin/:$PATH

***Mac OSX Terminal:>***

export PATH=/Applications/SIVIC.app/Contents/sivic/local/bin/:$PATH

alias sivic=/Applications/SIVIC.app/Contents/Resources/sivic

1. **Reorder raw EPSI Data:**

First step is to reorder the EPSI data using the command line tool svk\_gepfile\_reader. This tool is used to interpet pfiles acquired on GE scanners. Note in the output that all pfiles in the dynamic series are found and loaded.

***LINUX/OSX Terminal:>***

cd c13\_rat

svk\_gepfile\_reader -i variable\_flip/raw/dyn01 -o dyn\_reordered -t 4 --epsi\_flip 2

1. **Visualizing MRS data from the command line:**

Visualizing intermediate steps while processing via scripting can be very usefully for evaluating the correctness and the quality of your data. For this tutorial cookbook we will be using the SIVIC GUI purely for visualization of the data and all processing will be via the command line tools. With a LINUX installation use the “sivic -h” to see command line options:

***LINUX Terminal:>***

sivic -h

***Output:>***

Version 0.9.55

sivic [-a anatomy] [-h]

-a anatomy Anatomy preferences

brain (default)

prostate

-i fileName Input file name, may specify multiple,

instances of -i, e.g MRS, MRI (ref and

overlay). For series, only specify one

image from the series.

--is fileName Only load the single explicit file name,

no globbing of associated files names.

--id fileName Load as dynamic traces

-h Print help mesage.

SIVIC GUI.

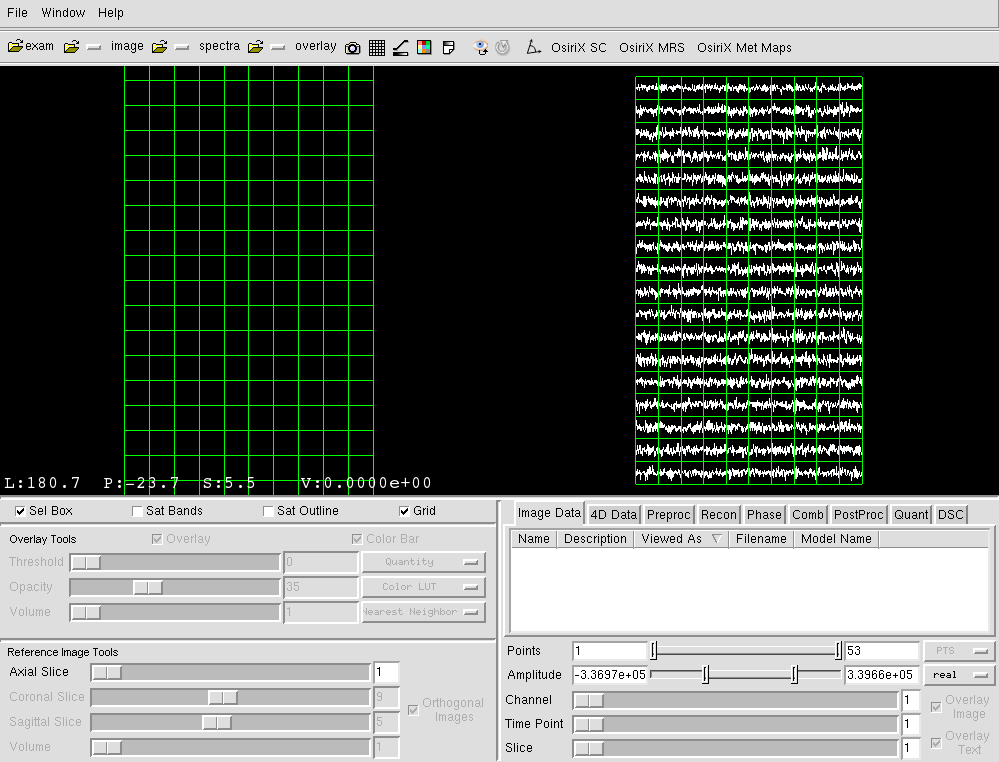
Example:

sivic -i P1234 -i E1234S1I1.DCM -i E1234S2I1.DCM

***LINUX/OSX Terminal:>***

sivic -i dyn\_reordered.dcm

The sivic gui will load as seen below:



1. **Inspecting meta-data from the command line:**

All data in the svk framework is stored internally as DICOM objects. The command line tool “svk\_dcmdump” can be used to dump the DICOM header as interpreted by svk:

***LINUX/OSX Terminal:>***

svk\_dcmdump dyn\_reordered.dcm

Displayed below are some fields from the DICOM header that are critical for analyzing the data. Note at the end of the output below the “DimensionIndexSequence” section which identifies the dimensionality of the data. In this case we can see that this data has Slice, Time, Channel, and EPSI\_ACQ (lobes):

***Output:>***

|  |  |
| --- | --- |
| SVK DICOM Header:  # Dicom-File-Format  # Dicom-Meta-Information-Header  # Used TransferSyntax: Little Endian Explicit  …  (0002,0002) UI =MRSpectroscopyStorage  …  (0010,0010) PN [SC03 rat]  (0010,0020) LO [trash]  (0010,0030) DA (no value available)  (0010,0040) CS [O]  (0018,0087) DS [3]  …  (0018,9004) CS [RESEARCH]  (0018,9005) SH [fidcsi\_ucsf\_multipulse]  …  (0018,9032) CS [RECTILINEAR]  (0018,9033) CS [SINGLE]  (0018,9034) CS [LINEAR]  (0018,9052) FD 581  (0018,9053) FD 178  …  (0018,9093) US 1  (0018,9094) CS [FULL]  (0018,9098) FD 32.136220799999997  (0018,9100) CS [13C]  …  (0020,9222) SQ (Sequence with explicit length #=4)  (fffe,e000) na (Item with explicit length #=3)  (0020,9165) AT (0020,0032)  (0020,9167) AT (0020,9113)  (0020,9421) LO [SLICE]  (fffe,e00d) na (ItemDelimitationItem for re-encoding)  (fffe,e000) na (Item with explicit length #=3)  (0020,9165) AT (0020,9128)  (0020,9167) AT (0020,9111)  (0020,9421) LO [TIME]  (fffe,e00d) na (ItemDelimitationItem for re-encoding)  (fffe,e000) na (Item with explicit length #=3)  (0020,9165) AT (0018,9047)  (0020,9167) AT (0018,9045)  (0020,9421) LO [CHANNEL]  (fffe,e00d) na (ItemDelimitationItem for re-encoding)  (fffe,e000) na (Item with explicit length #=3)  (0020,9165) AT (2000,025a)  (0020,9167) AT (21d0,025a)  (0020,9421) LO [EPSI\_ACQ]  (fffe,e00d) na (ItemDelimitationItem for re-encoding)  (fffe,e0dd) na (SequenceDelimitationItem for re-encod.) | …  #28, 1 MediaStorageSOPClassUID  …  #8, 1 PatientName  #6, 1 PatientID  #0, 0 PatientBirthDate  #2, 1 PatientSex  #2, 1 MagneticFieldStrength  …  #8, 1 ContentQualification  #22, 1 PulseSequenceName  …  #12, 1 GeometryOfKSpaceTraversal  #6, 1 SegmentedKSpaceTraversal  #6, 1 RectilinearPhaseEncodeReordering  #8, 1 SpectralWidth  #8, 1 ChemicalShiftReference  …  #2, 1 NumberOfKSpaceTrajectories  #4, 1 CoverageOfKSpace  #8, 1 TransmitterFrequency  #4, 1 ResonantNucleus  …  #0, 1 DimensionIndexSequence  #0, 1 Item  #4, 1 DimensionIndexPointer  #4, 1 FunctionalGroupPointer  #6, 1 DimensionDescriptionLabel  #0, 0 ItemDelimitationItem  #0, 1 Item  #4, 1 DimensionIndexPointer  #4, 1 FunctionalGroupPointer  #4, 1 DimensionDescriptionLabel  #0, 0 ItemDelimitationItem  #0, 1 Item  #4, 1 DimensionIndexPointer  #4, 1 FunctionalGroupPointer  #8, 1 DimensionDescriptionLabel  #0, 0 ItemDelimitationItem  #0, 1 Item  #4, 1 DimensionIndexPointer  #4, 1 FunctionalGroupPointer  #8, 1 DimensionDescriptionLabel  #0, 0 ItemDelimitationItem  #0, 0 SequenceDelimitationItem |

1. **Preprocessing data:**

Here we will perform two pre-processing steps: apodization and zero filling:

***LINUX/OSX Terminal:>***

svk\_zerofill -i dyn\_reordered.dcm -o dyn\_zf --custom 256

svk\_apodize -i dyn\_zf.dcm -o dyn\_apodized -f 1 --width 10

Now we can use svk\_dcmdump to see the application of the zero filling:

***LINUX/OSX Terminal:>***

svk\_dcmdump dyn\_zf.dcm | grep DataPointColumns

***Output:>***

(0028,9002) UL 256 #4, 1 DataPointColumns

1. **Reconstruct data:**

Again using svk\_dcmdump we can see that the data is currently in K-Space and needs to be transformed:

***LINUX/OSX Terminal:>***

svk\_dcmdump dyn\_zf.dcm | grep Domain$

***Output:>***

(7777,1001) CS [KSPACE] # 6, 1 SVK\_ColumnsDomain

(7777,1002) CS [KSPACE] # 6, 1 SVK\_RowsDomain

(7777,1003) CS [KSPACE] # 6, 1 SVK\_SliceDomain

Note: These fields are not currently part of the DICOM standard and have been recommended as additions to the DICOM standard. Next we will use the command line tool svk\_fft to reconstruct the data.

***LINUX/OSX Terminal:>***

svk\_fft -i dyn\_apodized.dcm -o dyn\_recon2lobes

svk\_dcmdump dyn\_recon2lobes.dcm | grep Domain$

***Output:>***

(7777,1001) CS [SPACE] # 6, 1 SVK\_ColumnsDomain

(7777,1002) CS [SPACE] # 6, 1 SVK\_RowsDomain

(7777,1003) CS [SPACE] # 6, 1 SVK\_SliceDomain

1. **Combine EPSI lobes and Coils:**

Currently our data still contains two EPSI lobes and 8 coils. We will combine the ESPI lobes using svk\_reorder\_epsi and combine the coils using svk\_mrs\_combine.

***LINUX/OSX Terminal:>***

svk\_reorder\_epsi -i dyn\_recon2lobes.dcm -o dyn\_recon\_combinedlobes --combine

svk\_mrs\_combine -i dyn\_recon\_combinedlobes.dcm -o combined -a 3 -t 4

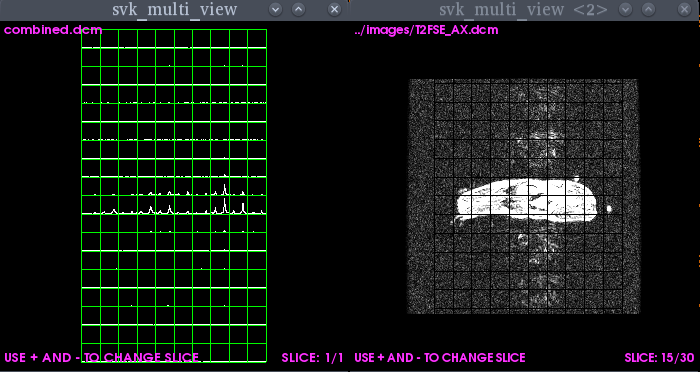
1. **Load corrected dynamic 13C MRS Data using svk\_multi\_view:**

As an alternative to the “SIVIC” GUI the svk\_multi\_view command can be used to quickly render both images and spectra:

***LINUX/OSX Terminal***

svk\_multi\_view -s combined.dcm images/T2FSE\_AX.dcm

Traces can be selected by clicking and dragging a box around the voxels of interest in the trace view (left side) and the image can be window-leveled by clicking and dragging as in the image view (right side).

**

1. **Correct for dual-band variable flip angle pulses:**

This dataset contains a variable flip angle acquisition. We have a “dad” file that was

acquired with the data that contains the weights to correct for this variance. We will

apply this correction using svk\_variable\_flip\_scaler:

***LINUX/OSX Terminal:>***

svk\_variable\_flip\_scaler -o corrected\_mrs -m 0.02 -i combined.dcm --dad variable\_flip/vfa\_profile.dad

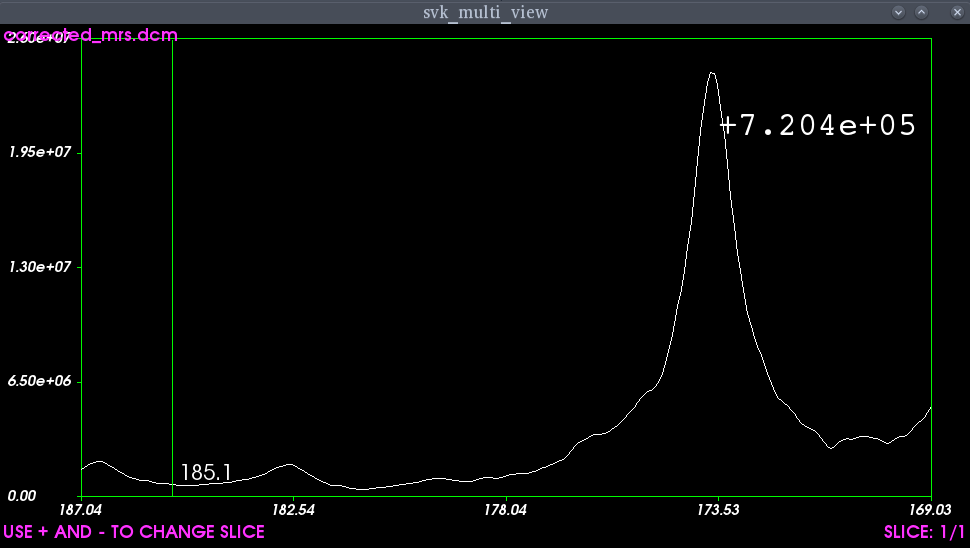
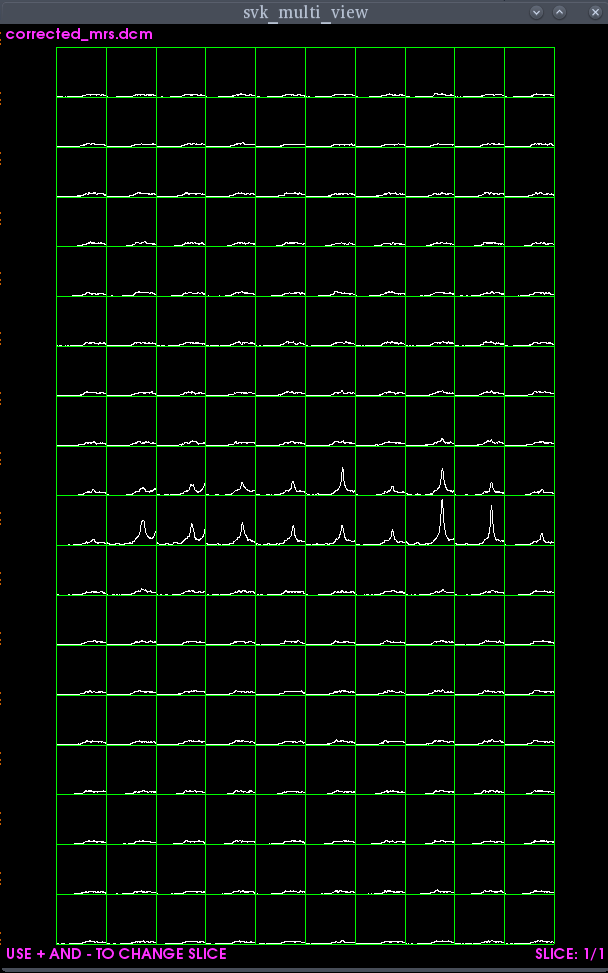
1. **Quantify the metabolites at each time point:**

In this step the magnitude peak height of pyruvate and lactate at each time point will be quantified. These quantified values will be written to dynamic metabolite maps in DICOM Enhanced MRI format and used as inputs for fitting the pyruvate to lactate conversion kinetics below. This process can be done via the SIVIC GUI, but here we will use svk\_multi\_view to show some of its extended capabilities. We know the lactate signal will peak around timepoint 5 so we will load up traces at that timepoint, and adjust the Y-axis range to be between 0 and 2.6\*107:

***LINUX/OSX Terminal:>***

svk\_multi\_view -s corrected\_mrs.dcm -t5 -l0 -u26000000

Resize this window to an easy to view size and then click on a single voxel that shows the largest lactate signal as show below. Again this window can be resized for a better aspect ratio.



Cursor

Pyruvate

Lactate

Click!

Cursor

* When a single voxel is selected a cursor will appear that can be used to determine PPM ranges used for generating metabolite maps.
* Determine the upper and lower frequency for the quantifying of lactate and pyruvate. We will use a peak of 172ppm and a width of 6ppm for pyruvate and a peak of 185.75ppm with a width of 2.5ppm for lactate.
* We will now have to open a configuration file use by SIVIC. This can be done with any text editor. It can be found here: ~/.SIVICQuantrc.xml
* Edit the following 13C section of the text file using your preferable text editor:

<APPLICATION nucleus="13C"

<REGION id="0" name="LACTATE" peak\_ppm="**186.6**" width\_ppm="**2.5**">

</REGION>

<REGION id="1" name="ALANINE" peak\_ppm="178" width\_ppm="2">

</REGION>

<REGION id="2" name="PYRUVATE" peak\_ppm="**173.7**" width\_ppm="**6**">

</REGION>

<REGION id="3" name="UREA" peak\_ppm="164" width\_ppm="2">

</REGION>

* Next we will use this configuration file to quantify the data using svk\_quantify.

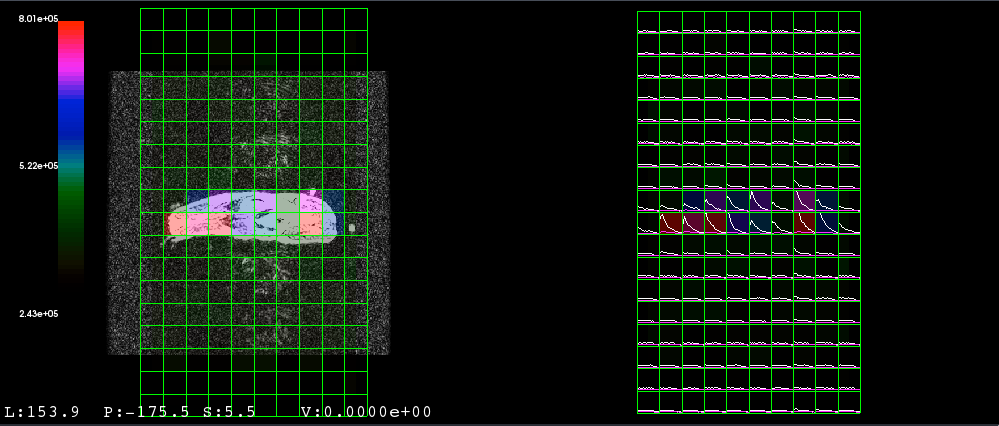
***LINUX/OSX Terminal:>***

svk\_quantify --xml ~/.SIVICQuantrc.xml -i corrected\_mrs.dcm -o mets -t6

We can display the lactate and pyruvate traces using sivic on the command line.

***LINUX/OSX Terminal:***

sivic --id metsPYRUVATE\_MAG\_PEAK\_HT.dcm --id metsLACTATE\_MAG\_PEAK\_HT.dcm -i images/T2FSE\_AX.dcm



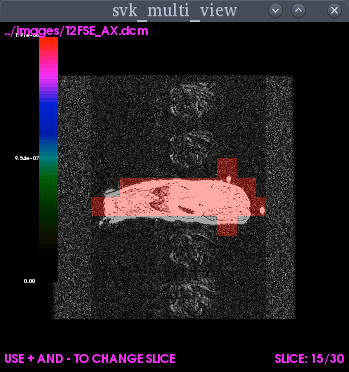
1. **Generate an ROI mask for kinetic modeling:**

We are going to use the svk\_image\_threshold command line tool to try and generate a rough mask by theresholding the pyruvate image. We’ll view the results using svk\_multi\_view.

***LINUX/OSX Terminal:>***

svk\_image\_threshold -i metsPYRUVATE\_MAG\_PEAK\_HT.dcm -l5300000 -o mask -t6 -b -v1

svk\_multi\_view -o mask.dcm images/T2FSE\_AX.dcm



**13. Run command line program to fit dynamic data to a 2-site exchange model and generate a Kpl parameter map:**

For this step we will use svk\_met\_kinetics to fit kinetic data at each voxel within the mask ROI:

***LINUX/OSX Terminal***

svk\_met\_kinetics --i1 metsPYRUVATE\_MAG\_PEAK\_HT.dcm --i2 metsLACTATE\_MAG\_PEAK\_HT.dcm --mask mask.dcm -o rat\_kinetics -t 6 --tr 3

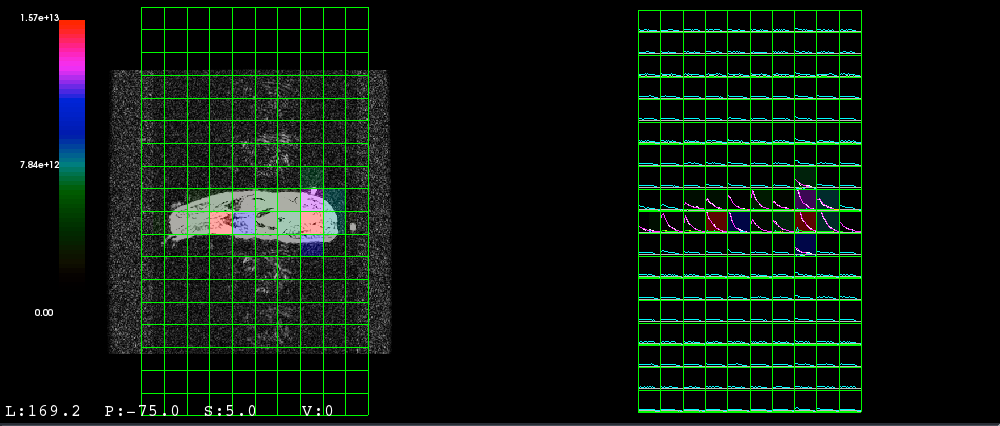
* Once the program completes a set of files named rat\_kinetics\*.dcm will have been created.
  + \_pyr\_fit calculated pyruvate signal based on fitted parameters (4D) points
  + \_lac\_fit calculated lactate signal based on fitted parameters (4D)
  + \_Kpl 3D Kpl parameter map (3D)
  + \_T1all T1all parameter map (3D)

**14. Load Model Results into the SIVIC GUI:**

Load results from the kinetic modeling into the SIVIC GUI.

***LINUX/OSX Terminal:***

sivic --id metsPYRUVATE\_MAG\_PEAK\_HT.dcm --id rat\_kinetics\_pyr\_fit.dcm --id rat\_kinetics\_pyr\_residual.dcm -i mask.dcm --id metsLACTATE\_MAG\_PEAK\_HT.dcm --id rat\_kinetics\_lac\_fit.dcm --id rat\_kinetics\_lac\_residual.dcm -i images/T2FSE\_AX.dcm -i rat\_kinetics\_T1all.dcm -i rat\_kinetics\_Kpl.dcm -i rat\_kinetics\_dcoffset.dcm -i rat\_kinetics\_rss.dcm



* All dynamic data sets will be displayed in the right SIVIC panel.
* Select voxels within the rat body.
* Explore the data using the SIVIC GUI. See our GUI Cookbook for instructions for adjusting the display as describe below.
  + Compare quality of fitted pyruvate and lactate signal to input signal
  + Window level anatomic image
  + Adjust color window level of overlay
  + Interpolate overlay
  + Adjust opacity of overlay

