

SUCK-SESSFUL IMPROVEMENTS ON A DATA-ANALYZING INTERFACE



Authors: Zachary Bradshaw, Matthew Cavell, Murad Chowdhury, Jeannie Grodner, Joe Wilkerson

Project Manager: Richard Frnka

Introduction

Pennington Biomedical Research is conducting a study where they are determining the relationship between infant suck patterns and childhood obesity. A previous project was done to help create an interface that would help the researchers analyze their data and test their hypotheses. The interface presents the data graphically and detects “sucks” and trends in the data using specific prototypes that represent a typical suck.

The goal of the project this semester was to improve upon the interface, including the methods in which the data was analyzed and the results were presented, and additional features were added to allow the user more flexibility in handling different kinds of testing data.

Prototypes

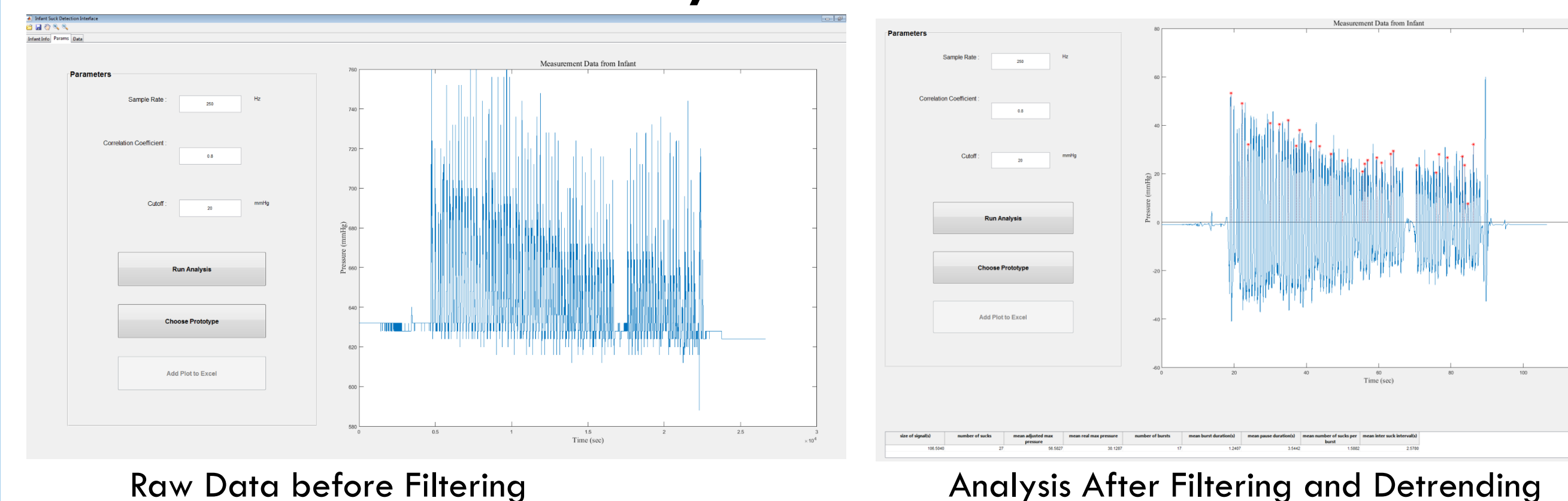
To determine if a peak in a data set is a suck, we convolute it with a prototype of what a typical suck looks like. If the peak correlates within a certain range with the prototype, it is registered as a suck.

The original paper and methodology that laid the foundation for this project used a simple upside-down parabola as the prototype. This had limited applications and could not deal with irregular data sets. With the use of the POD GUI, we can create our own prototypes by compiling a list of sucks that represent a data set.

The Interfaces

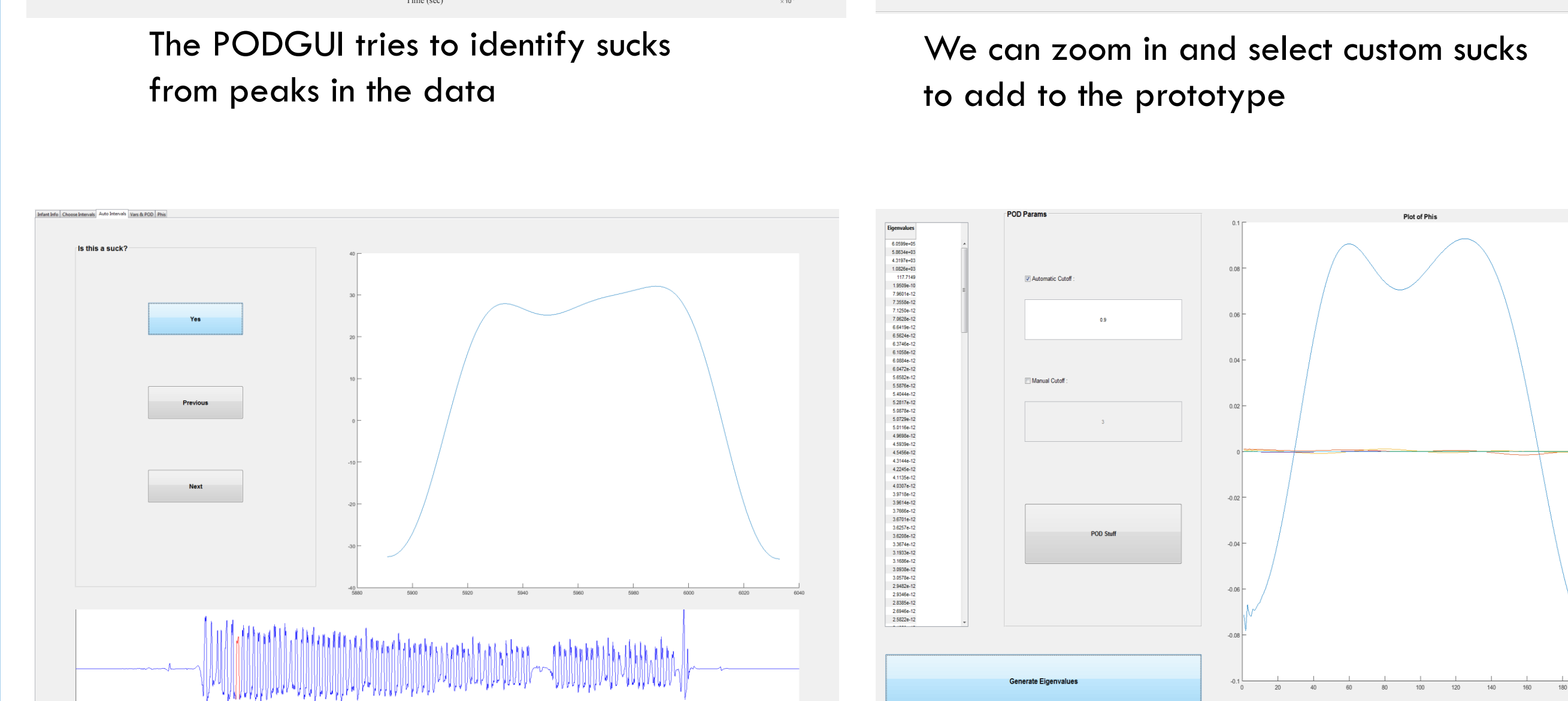
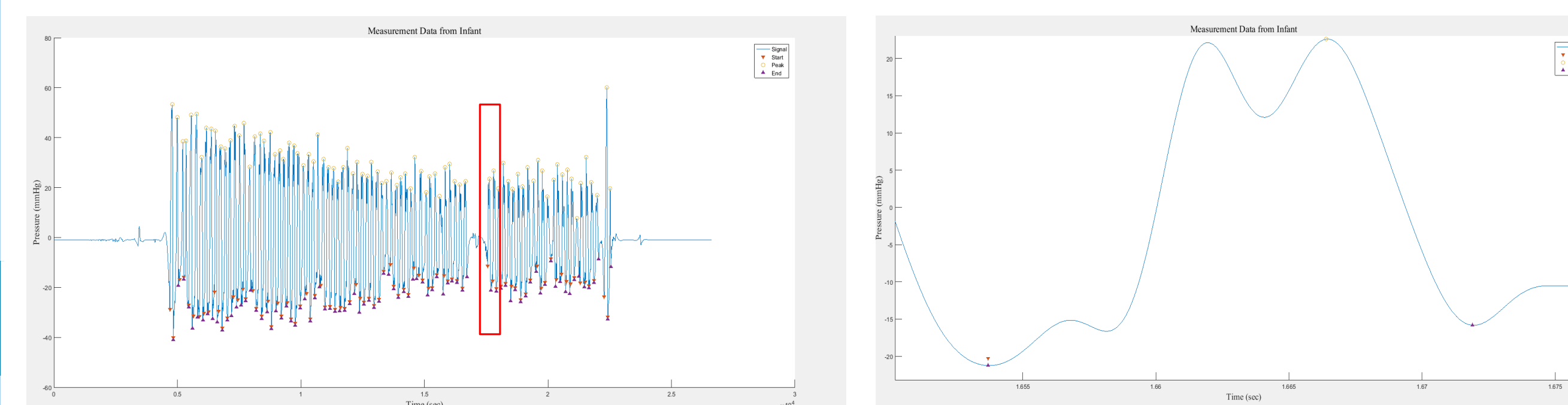
The SuckUI

The SuckUI does the actual analysis of the data set, with existing choices for prototypes to use in the analysis



The POD GUI

The POD Gui allows the user to create custom prototypes by adding snapshots of existing sucks to a list, which are then put into a matrix that is decomposed using Principal Component Analysis, creating a prototype from the most significant principal components.



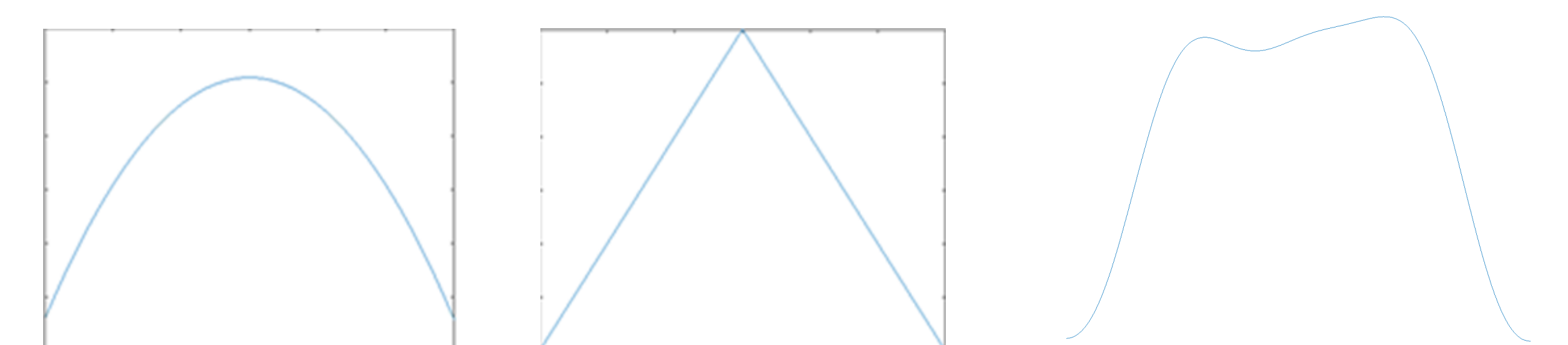
Improvements over the Previous Interface

-Filtering/Preprocessing - We fixed issues with the filtering and preprocessing of the data sets. It now uses a low-pass/high-pass filter, which “de-trends” the data set.

-Optimization of the GUIs - Various backend improvements have been made that give the user a smoother testing experience.

-Creation of Custom Suck Feature - We have implemented a user focused tool for the creation of new prototypes.

-New Prototype Creation - We have enabled the user to have a more robust catalog of testable prototypes by adding new default prototypes.



Acknowledgements

We would like to thank Pennington Biomedical Center for providing this project. Also, we would like to thank Dr. Peter Wolenski and the LSU Math Department for their assistance and support.

CAPSTONE FALL 2016