

Meeting #9

11-4-21

Michael Lee

Deliverables

- Implement callbacks between the slider and graphs

Methodology and Learnings

- Callbacks

- Placeholder Graphs

- Dataframes

Results

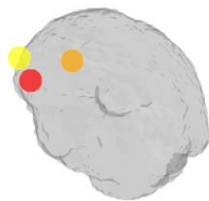
```
53 # function that returns a figure with the biometric graph for a selected epoch
54 # INPUTS - the dataframe and biometric to be graphed - string, i.e HR or SpO2, and epochbm dict
55 #         and epoch as an int
56 # OUTPUTS - the figure
57 def graph_biometric_selection(df, biometric, epochbm_dict, epoch):
58     epoch = str(epoch)
59     df_hr = df[biometric]
60     df_hr_epoch_x = df_hr[epochbm_dict[epoch][0][0]:epochbm_dict[epoch][0][1]]
61     hr_epoch = df_hr_epoch_x.to_numpy()
62
63     fig = go.Figure(data=[go.Scatter(x=list(range(len(hr_epoch))), y=hr_epoch)])
64     fig.update_layout(
65         title="%s for Epoch %s" % (biometric, epoch),
66         title_x=0.5,
67         xaxis_title="Time",
68         yaxis_title=biometric
69     )
70     return fig
```

```
146 # Updates hr plot when the slider is changed
147 @app.callback(
148     dash.dependencies.Output('hr-graph1', 'figure'),
149     dash.dependencies.Input(slider.id, 'value')
150 )
151 def update_graph1(selected_value):
152     fig = graph_biometric_selection(df1, 'HR', epochbm_dict, selected_value)
153     return fig
154
155 # Updates SpO2 plot when the slider is changed
156 @app.callback(
157     dash.dependencies.Output('SpO2-graph1', 'figure'),
158     dash.dependencies.Input(slider.id, 'value')
159 )
160 def update_graph1(selected_value):
161     fig = graph_biometric_selection(df1, 'SpO2', epochbm_dict, selected_value)
162     return fig
163
```

```
108 # main application
109 app = dash.Dash(__name__, external_stylesheets=external_stylesheets)
110 app.layout = html.Div(children=[
111     html.Div([ # upper row with brain and hr plot
112         html.Div([
113             html.Div([
114                 mycheckbox,
115             ]),
116             html.Div([
117                 dcc.Graph(
118                     id="image-display-graph-3d",
119                     config=dict(displayModeBar=False),
120                     # figure=fig
121                 ),
122             ]),
123         ], className='six columns'),
124         html.Div([
125             dcc.Graph(id='hr-graph1')
126         ], className='six columns'),
127     ], className='row'),
128     html.Div([ # bottom row with just spo2 plot for now
129         html.Div([
130             dcc.Graph(id='SpO2-graph1')
131         ], className='six columns'),
132     ], className='row'),
133     slider
134 ])
135
```

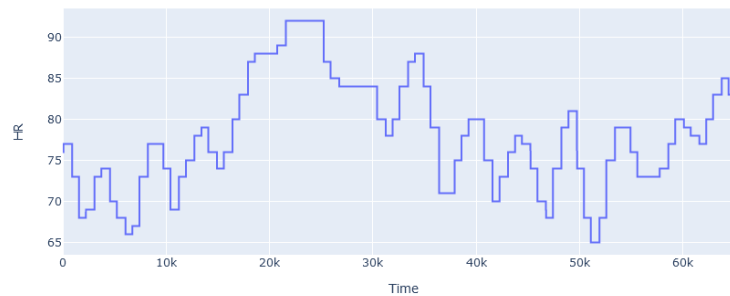
Results

Fp1
Fp2
F3

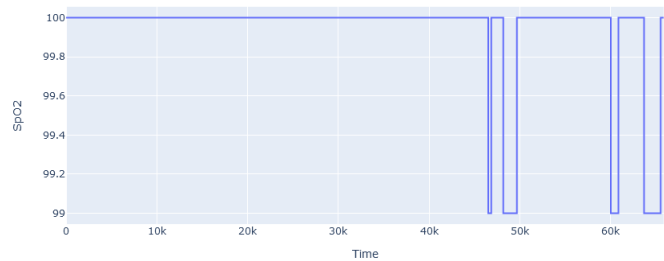


Fp1
Fp2
F3

HR for Epoch 2



SpO2 for Epoch 2



epoch 1

epoch 2

epoch 3

epoch 4

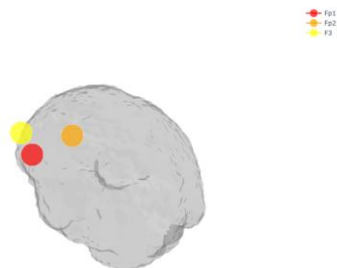


epoch 5

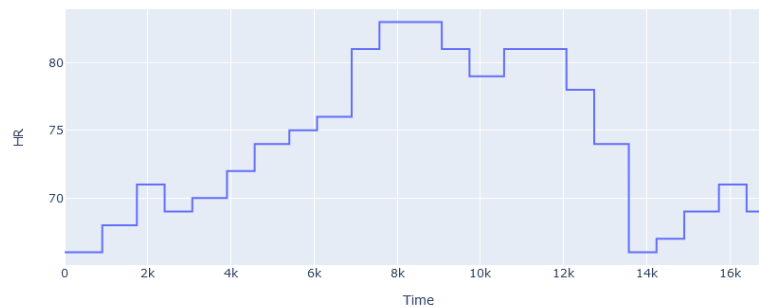
5

Results

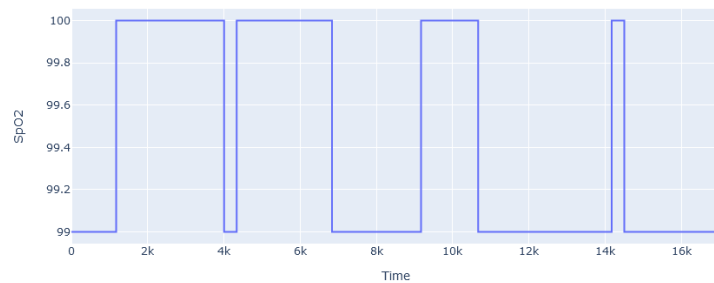
☒ Fp1
☒ Fp2
☒ F3



HR for Epoch 4



SpO2 for Epoch 4



epoch 1

epoch 2

epoch 3

epoch 4

epoch 5