I. Supplementary Information: Methods

A. Multi-Rate State Space Formulation

We put Equations 1-3 into a state space form and derive the discrete analog of the system. In this formulation, the unknowns include τ_1 and τ_2, q_i and Δ_i (for $i = 1, 2, \cdots, N$). Let, $x(t) = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^\top$, $\mathcal{A}_c = \begin{bmatrix} -\frac{1}{\tau_1} & 0 \\ \frac{1}{\tau_1} & -\frac{1}{\tau_2} \end{bmatrix}$, $\mathcal{B}_c = \begin{bmatrix} \frac{1}{\tau_1} \\ 0 \end{bmatrix}$ and $C_c = \begin{bmatrix} 0 & 1 \end{bmatrix}$. Hence, the state space model can be written as:

$$\dot{x}(t) = \mathcal{A}_c x(t) + \mathcal{B}_c u(t)$$
$$y(t) = C_c x(t) + \nu(t)$$

where $y(t)$ is the observed skin conductance (SC) and $\nu(t)$ is the measurement noise at time t. Assuming that the input and the states are constant over T_u, by letting $\Lambda = e^{AT_u}$, and $\Gamma = \int_0^{T_u} e^{A(T_u-\rho)}d\rho$, we can write the discrete state space form as:

$$x[k+1] = \Lambda x[k] + \Gamma u[k]$$
$$y[k] = C_c x[k] + \nu[k].$$

This can be extended to a multi-rate formulation, i.e., for the cases where neural stimuli and SC measurement have different sampling frequencies. We let the SC measurement sampling frequency $T_y = LT_u$, where L is an integer. We can also represent L as the ratio of number of samples in neural stimuli and SC signal, i.e. $L = \frac{N}{M}$ where M is the number of available data points. By letting $\mathcal{A}_d = \Lambda^L$, $\mathcal{B}_d = \begin{bmatrix} \Lambda^{L-1}\Gamma & \Lambda^{L-2}\Gamma & \cdots & \Gamma \end{bmatrix}$, $u_d[k] = \begin{bmatrix} u[Lk] & u[Lk+1] & \cdots & u[Lk+L-1] \end{bmatrix}^\top$, $\nu_d[k] = \nu[Lk]$ and $x_d[k] = x_d[Lk]$, we can represent the multi-rate system as:

$$x_d[k+1] = \mathcal{A}_d x_d[k] + \mathcal{B}_d u_d[k]$$
$$y[k] = C_c x_d[k] + \nu_d[k]$$

where $\mathcal{A}_d$ and $\mathcal{B}_d$ are functions of $\boldsymbol{\tau} = \begin{bmatrix} \tau_1 & \tau_2 \end{bmatrix}$. Then, using the state transition matrix, and considering that the system is causal, we can write the system equation as:

$$y[k] = \mathcal{F}[k] x_d[0] + \mathcal{D}[k]\mathbf{u} + \nu_d[k]$$

where $\mathcal{F}[k] = C_c\mathcal{A}_d^k$, $\mathcal{D}[k] = C_c \begin{bmatrix} \mathcal{A}_d^{k-1}\mathcal{B}_d & \mathcal{A}_d^{k-2}\mathcal{B}_d & \cdots & \mathcal{B}_d & \underbrace{0 \cdots 0}_{N-kL} \end{bmatrix}$, and $\mathbf{u} = \begin{bmatrix} u_d[0] & u_d[1] & \cdots & u_d[k-1] & \cdots & u_d[M-1] \end{bmatrix}^\top$. $\mathbf{u}$ represents the entire input over the duration of the study. Considering the initial condition $x_1(0) = 0$ and $y(0) = x_2(0) = y_0$, we can let $x_d[0] = \begin{bmatrix} 0 & y_0 \end{bmatrix}^\top$. Then, let $\mathbf{y} = \begin{bmatrix} y[1] & y[2] & \cdots & y[M] \end{bmatrix}^\top_{M\times 1}$, where $\mathbf{y}$ represents all the data points. Moreover, let $\mathbf{F}_\tau = \begin{bmatrix} \mathcal{F}[0] & \mathcal{F}[1] & \cdots & \mathcal{F}[M-1] \end{bmatrix}^\top_{M\times 2}$, $\mathbf{D}_\tau = \begin{bmatrix} \mathcal{D}[0] & \mathcal{D}[1] & \cdots & \mathcal{D}[M-1] \end{bmatrix}^\top_{M\times N}$, and $\boldsymbol{\nu} = \begin{bmatrix} \nu[1] & \nu[2] & \cdots & \nu[M] \end{bmatrix}^\top_{M\times 1}$. Hence, we can represent this system as:

$$\mathbf{y} = \mathbf{F}_\tau x_d[0] + \mathbf{D}_\tau \mathbf{u} + \nu.$$

This solution is equivalent to Equation 5 by considering $\mathbf{F}_\tau x_d[0] = \mathbf{A}_\tau y_0$ and $\mathbf{D}_\tau = \mathbf{B}_\tau$.

B. Discretization of Neural Impulse Train

As both discrete and continuous representation of neural stimuli has been carried out with impulse functions, a careful conversion between these representations is necessary. $u(t)$ is defined as a summation of weighted delta functions, i.e., $u(t) = \sum_{i=1}^N q_i\delta(t-\Delta_i)$ where $\Delta_i = iT_u$ is the arrival time of the corresponding impulse. Each delta function has an area of 1 under the curve. For discretization, we first take an approximation of the Dirac delta function with a rectangular function of width T_u and height $\frac{1}{T_u}$ to have the area of the rectangle 1 where T_u is the sampling interval of neural stimuli. Then, we sample the neural stimuli with the sampling interval T_u. For example, a continuous time neural stimuli $u_i(t) = q_i\delta(t-\Delta_i)$ with only one weighted impulse (Figure 1 (a)) can be written as a scaled and shifted rectangular $\tilde{u}_i(t) = \frac{q_i}{T_u}\Pi(\frac{t-\Delta_i-T_u/2}{T_u})$ (Figure 1 (b)). It has been time scaled to have a bin size equal to the sampling frequency. Amplitude has been scaled with the reciprocal of the sampling frequency to keep the area under the curve same as in $u_i(t)$. Finally, the approximation can be sampled to obtain discrete sequence $u_i[k]$ (Figure 1 (c)).

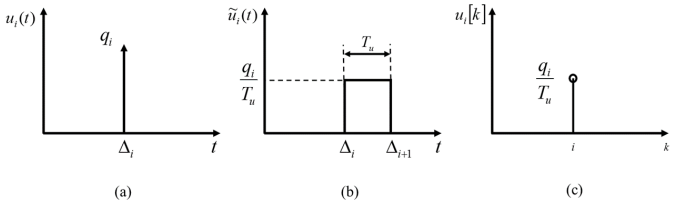

Fig. 1: **Discretization of Neural Stimuli.** Each panel shows the steps for discretization of the neural stimuli represented with weighted impulse train. (a) An example of continuous time neural stimuli $u_i(t)$, represented with with an weighted and shifted Dirac delta function, (b) the approximation to the $u_i(t)$ function with a rectangular function and (c) the equivalent discrete neural stimuli represented with Kronecker delta function.

C. Subjects Information

The following table shows subject ID, age, gender and BMI of each participant.

Participant No.	Subject ID	Age	Gender	BMI $[\frac{kg}{m^2}]$
1	01	30	M	30.00
2	05	30	M	24.75
3	08	27	M	19.32
4	09	25	M	21.70
5	12	32	F	20.20
6	16	24	M	16.66

D. Derivation of Equation 11

For any signal $\zeta(t)$, the κth order spectral component of lth derivative of the signal can be modified as below [1]:

$$\bar{H}_\zeta^{(l)}(m\omega_0) = \int_0^{T_d} \phi_m(t)\frac{\mathrm{d}^l\zeta(t)}{\mathrm{d}t^l}dt$$

$$
\begin{aligned}
&= (-1)^l \int_0^{T_d} \zeta(t) \frac{\mathrm{d}^l \phi_m(t)}{\mathrm{d}t^l} dt \\
&= (-1)^l \sum_{j=0}^{\kappa} (-1)^j \binom{\kappa}{j} (\kappa + m - j)^l \omega_0^l (-1)^l \mathrm{cas}(-\frac{l\pi}{2}) \\
&\cdot \int_0^{T_d} \zeta(t) \mathrm{cas}((-1)^l(\kappa + m - j)\omega_0 t) dt \\
&= \sum_{j=0}^{k} (-1)^j \binom{\kappa}{j} (\kappa + m - j)^l \omega_0^l \mathrm{cas}(-\frac{l\pi}{2}) \\
&\cdot H_\zeta((-1)^l(\kappa + m - j)\omega_0).
\end{aligned}
$$

E. Choice of HMF dependent Time Domain Optimization for Estimating β

We can rewrite the cost function in the Equation 17 as follows,

$$J(\boldsymbol{\Theta}, \mathbf{u}, \beta) = \frac{1}{2} \sum_{m=-\mathcal{M}}^{-\mathcal{M}} w_\beta[m + M] \varepsilon^2(m\omega_0).$$

Each of the error terms is multiplied by a window coefficient and smaller values of window function coefficients will lead to smaller value of cost function. Figure 2 shows how the shape of the window function changes with the value of β. A higher value of β leads to a very narrow function which minimizes the cost function. However, a higher value of β will discard most of the information in the HMF spectral components. This way of solving the optimization problem in the HMF domain for β tends to discard the signal components along with the noise components. For example, the minimum value for the HMF domain cost function can be found with all zeros in the window function. However, zero window function clearly discarding all the information of the signal. To prevent discarding the key information in the HMF spectral components, we can instead solve the time domain equivalent of the optimization formulation in 17, which is presented in the optimization formulation in 19.

F. FOCUSS+ Algorithm

FOCUSS+ [2] solves for nonnegative $\mathbf{u}$ such that $\mathbf{u}$ has a certain maximum sparsity n_u while minimizing the following optimization problem,

$$\underset{\mathbf{u} \geq 0}{\text{minimize}} \quad \frac{1}{2} ||\mathbf{y} - \mathbf{A}_{\boldsymbol{\tau}} y_0 - \mathbf{B}_{\boldsymbol{\tau}} \mathbf{u}||_2^2 + \lambda ||\mathbf{u}||_p^p.$$

(a) $\mathbf{P}_{\mathbf{u}}^{(r)} = \mathrm{diag}(|\mathbf{u}_i^{(r)}|^{2-p})$

(b) $\lambda^{(r)} = \left(1 - \frac{||\mathbf{y} - \mathbf{A}_{\boldsymbol{\tau}} y_0 - \mathbf{B}_{\boldsymbol{\tau}} \mathbf{u}||_2}{||\mathbf{y} - \mathbf{A}_{\boldsymbol{\tau}} y_0||_2}\right) \lambda_{\max}, \quad \lambda > 0$

(c) $\mathbf{u}^{(r+1)} = \mathbf{P}_{\mathbf{u}} \mathbf{B}_{\boldsymbol{\tau}}^\top (\mathbf{B}_{\boldsymbol{\tau}} \mathbf{P}_{\mathbf{u}} \mathbf{B}_{\boldsymbol{\tau}}^\top + \lambda \mathbf{I})^{-1} (\mathbf{y} - \mathbf{A}_{\boldsymbol{\tau}} y_0)$

(d) $\mathbf{u}_i^{(r+1)} \leq 0 \rightarrow \mathbf{u}_i^{(r+1)} = 0$

(e) After half of the selected number of iterations, search for the peaks with distances less than the minimum peak to peak distance Δ_{min}. Keep the largest peak among the adjacent peaks within Δ_{min} window.

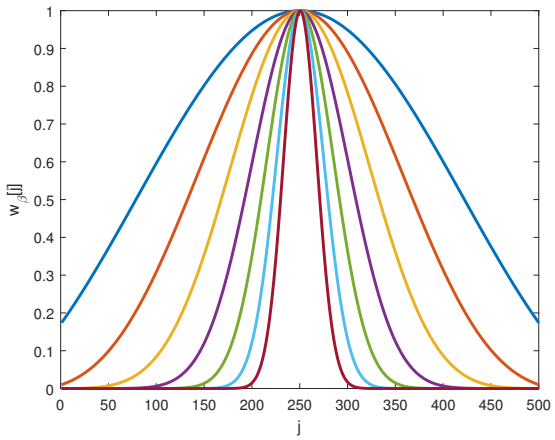

Fig. 2: **Kaiser Windows With Different Shape Parameters.** Each curve represents different $w_\beta[j]$ window functions with different shape parameters β. For example the narrowest window represented with red curve is $w_\beta[j]$ with $\beta = 204.8$ and the widest window represented with blue curve is $w_\beta[j]$ with $\beta = 3.2$.

(f) After about half of the selected number of iterations, if $||\mathbf{u}^{(r+1)}||_0 > n_{\mathbf{u}}$, select $n_{\mathbf{u}}$ the largest values of elements of $\mathbf{u}^{(r+1)}$ and set all other elements to zero.

(g) Iterate

Note that we used $\Delta_{min} = 0.5$ s in this study.

G. GCV-FOCUSS+ Algorithm

The sparse identification problem in the HMF domain is as follows,

$$
\begin{aligned}
\underset{\boldsymbol{\Theta}, \mathbf{u}, \beta}{\text{minimize}} \quad & J(\boldsymbol{\Theta}, \mathbf{u}, \beta) = \frac{1}{2} \boldsymbol{\varepsilon}^\top(\mathcal{M}\omega_0) \mathbf{W}(\beta) \boldsymbol{\varepsilon}(\mathcal{M}\omega_0) + \lambda ||\mathbf{u}||_p^p \\
\text{subject to} \quad & \\
& G(\boldsymbol{\Theta}) \leq \mathbf{0} \\
& \mathbf{u} \geq \mathbf{0}
\end{aligned}
$$

where $\boldsymbol{\varepsilon}(\mathcal{M}\omega_0) = \mathbf{Z}(\mathcal{M}\omega_0) - \boldsymbol{\Phi}(\mathcal{M}\omega_0)\boldsymbol{\Theta} - \boldsymbol{B}_\phi^\top(\mathcal{M}\omega_0)\mathbf{u}$. Let $\mathbf{Z}_{\boldsymbol{\Theta},\beta} = \sqrt{\mathbf{W}(\beta)}(\mathbf{Z}(\mathcal{M}\omega_0) - \boldsymbol{\Phi}(\mathcal{M}\omega_0)\boldsymbol{\Theta})$ and $\mathbf{B}_{\phi,\beta} = \sqrt{\mathbf{W}(\beta)}\mathbf{B}_\phi^\top(\mathcal{M}\omega_0)$. Given β and $\boldsymbol{\Theta}$, the optimization problem can be solved for $\mathbf{u}$ using the FOCUSS+ algorithm. We use a GCV based method for choosing a regularization parameter λ that balances between capturing noise and the sparsity level. Zdunek *et al.* [3] used the GCV technique for finding the value of λ for the FOCUSS+ algorithm incorporating singular value decomposition:

$$G(\lambda) = \frac{\mathcal{L} \sum_{i=1}^{\mathcal{L}} \gamma_i^2 \left(\frac{\lambda}{\sigma_i^2 + \lambda}\right)^2}{\sum_{i=1}^{\mathcal{L}} \left(\frac{\lambda}{\sigma_i^2 + \lambda}\right)^2}$$

where $\boldsymbol{\gamma} = \mathbf{R}^\top \mathbf{Z}_{\boldsymbol{\Theta},\beta} = \begin{bmatrix} \gamma_1 & \gamma_2 & \cdots & \gamma_{\mathcal{L}} \end{bmatrix}^\top$ and $\boldsymbol{B}_{\phi,\beta}\mathbf{P}_{\mathbf{u}}^{\frac{1}{2}} = \mathbf{R}\boldsymbol{\Sigma}\mathbf{Q}^\top$ with $\boldsymbol{\Sigma} = \mathrm{diag}\{\sigma_i\}$; $\mathbf{R}$ and $\mathbf{Q}$ are unitary matrices and σ_i's are the singular values of $\boldsymbol{B}_{\phi,\beta}\mathbf{P}_{\mathbf{u}}^{\frac{1}{2}}$ [3]. Moreover, $\mathcal{L}$ is the number of data points. In this study, we use a range of

zero to 0.1 for λ. For $r = 0, 1, 2, \cdots$, GCV-FOCUSS+ works as follows [4]:

(a) $\mathbf{P}_u^{(r)} = \text{diag}(|\mathbf{u}_i^{(r)}|^{2-p})$
(b) $\mathbf{u}^{(r+1)} = \mathbf{P}_u \boldsymbol{B}_{\phi,\beta}^{\top}(\boldsymbol{B}_{\phi,\beta}\mathbf{P}_u\boldsymbol{B}_{\phi,\beta}^{\top} + \lambda\mathbf{I})^{-1}\boldsymbol{Z}_{\boldsymbol{\Theta},\beta}$
(c) $\mathbf{u}_i^{(r+1)} \leq 0 \rightarrow \mathbf{u}_i^{(r+1)} = 0$
(d) $\lambda^{(r+1)} = \underset{0\leq\lambda\leq 0.1}{\text{argmin}}\, G(\lambda)$
(e) Iterate until convergence

H. Initialization Algorithm

The initialization is performed in the time domain (similar to [4], [5], [6]). A summary of the algorithm to obtain good initial conditions for $\boldsymbol{\tau}$, $\mathbf{u}$ and β is as follows:

(a) Initialize $\tilde{\boldsymbol{\tau}}^0$ by sampling a uniform random variable on $\begin{bmatrix} 0.10 & 1.4 \end{bmatrix}$ for $\tilde{\tau}_1^{(0)}$ and $\begin{bmatrix} 1.5 & 6 \end{bmatrix}$ for $\tilde{\tau}_2^{(0)}$ and let $j = 1$.
(b) Set $\boldsymbol{\tau}$ equal to $\tilde{\boldsymbol{\tau}}^{(j-1)}$ and use FOCUSS+ to solve the inverse problem to find the stimuli $\tilde{\mathbf{u}}^{(j)}$ by initializing $\tilde{\mathbf{u}}^{(0)}$ at a vector with all ones.
(c) Set $\mathbf{u}$ equal to $\tilde{\mathbf{u}}^{(j)}$; use the interior point method and minimize error $||\mathbf{y} - \mathbf{A}_{\boldsymbol{\tau}} y_0 - \mathbf{B}_{\boldsymbol{\tau}}\mathbf{u}||_2$ to solve the time domain system parameter identification problem for obtaining $\tilde{\boldsymbol{\tau}}^{(j)}$.
(d) Repeat between steps (b)-(c) for $j = 1, 2, 3, \cdots, 30$.
(e) We set $\boldsymbol{\tau}^0 = \tilde{\boldsymbol{\tau}}^{(j)}$ and $\mathbf{u}^0 = \tilde{\mathbf{u}}^{(j)}$
(f) We calculate $\boldsymbol{\Theta}^0$ by plugging in $\boldsymbol{\tau}^0$ in $\boldsymbol{\Theta} = \begin{bmatrix} \tau_1 + \tau_2 \\ \tau_1\tau_2 \end{bmatrix}$.
(g) Using $\mathbf{u}^0$ and $\boldsymbol{\tau}^0$, we first take $\beta_i = 0.1 \times 2^i$ for $i = 0, 1, 2, 3, \cdots, 10$ and set $\beta^0 = \beta_{\min}$ such that $\beta_{\min}$ minimizes $||\mathbf{y} - \mathbf{A}_{h(\beta)}y_0 - \mathbf{B}_{h(\beta)}\mathbf{u}||_2^2$.

II. Supplementary Information: Results

Figure 3 shows the quantile-quantile plots of the phasic SC model residual errors for the 6 participants suggesting that the model captures the SC dynamics, and that the phasic SC residual errors have a Gaussian structure and are white. Figure 4 shows the reconstructed signal that includes both tonic and phasic components. The values R^2 for this case is higher than 0.95 all participants.

References

[1] H. Unbehauen and P. Rao, "Identification of continuous-time systems: A tutorial," *IFAC Proceedings Volumes*, vol. 30, no. 11, pp. 973–999, 1997.
[2] J. F. Murray, "Visual recognition, inference and coding using learned sparse overcomplete representations," Ph.D. dissertation, University of California, San Diego, 2005.
[3] R. Zdunek and A. Cichocki, "Improved m-focuss algorithm with overlapping blocks for locally smooth sparse signals," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 4752–4761, 2008.
[4] R. T. Faghih *et al.*, "Deconvolution of serum cortisol levels by using compressed sensing," *Plos one*, vol. 9, no. 1, p. e85204, 2014.
[5] R. T. Faghih *et al.*, "Characterization of fear conditioning and fear extinction by analysis of electrodermal activity," in *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*. IEEE, 2015, pp. 7814–7818.
[6] R. T. Faghih *et al.*, "Quantifying pituitary-adrenal dynamics and deconvolution of concurrent cortisol and adrenocorticotropic hormone data by compressed sensing," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 10, pp. 2379–2388, 2015.

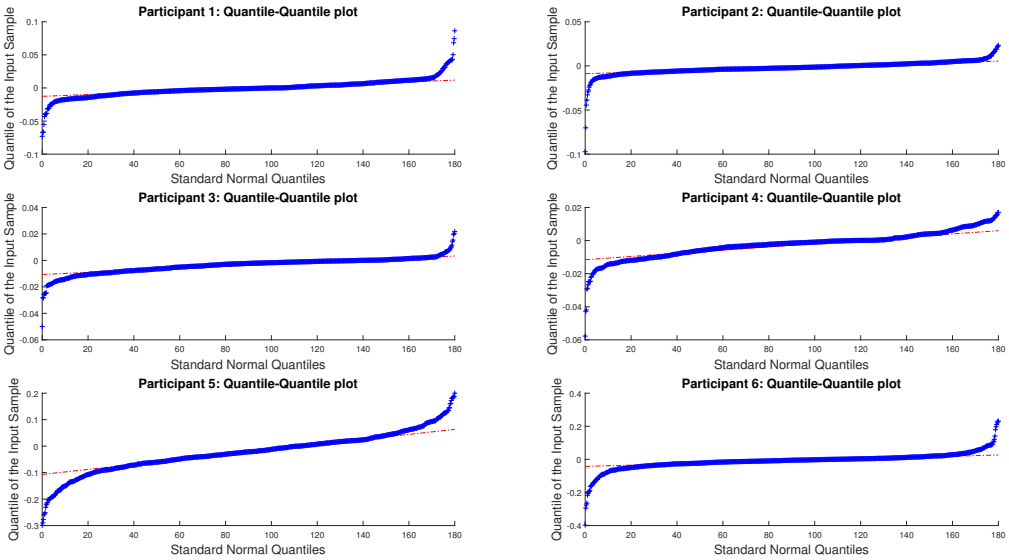

Fig. 3: **White Gaussian Structure in the Model Residual Errors of Phasic SC data of 6 Participants.** Each of the panels displays the quantile-quantile plot of the SC model residual errors for each of the 6 participants; the graph shows that the residual errors are Gaussian.

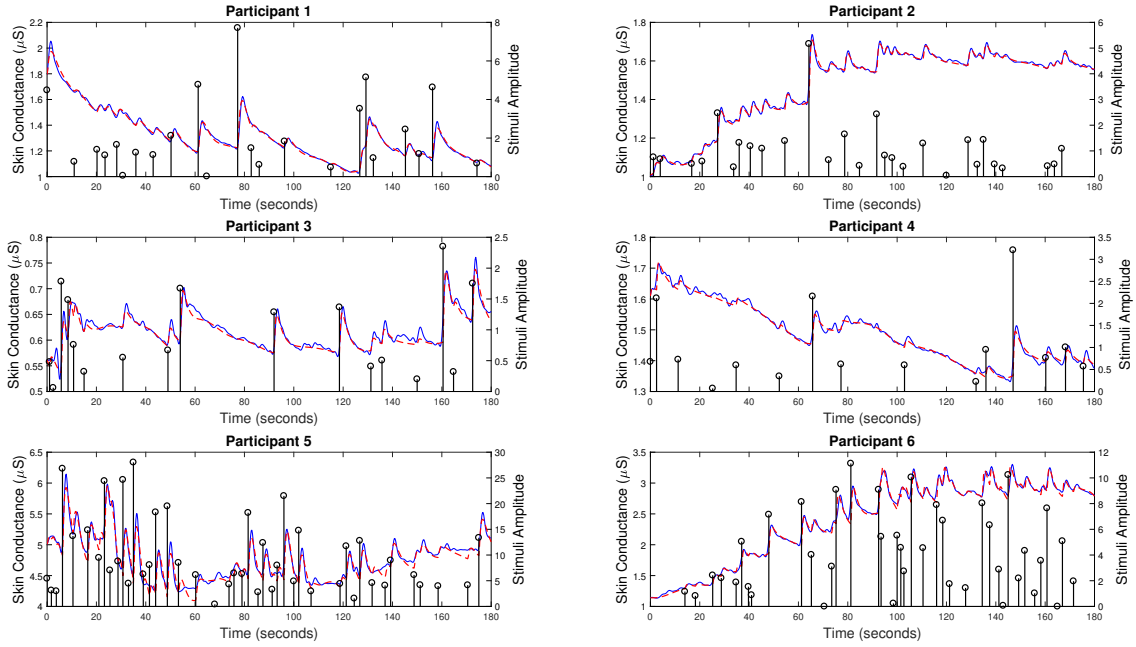

Fig. 4: **Estimated Neural Stimuli and Reconstructed Signals of the Experimental SC Data in 6 Participants.** Each panel shows the SC signal (sum of phasic and tonic components) (blue curve), the reconstructed SC signal (sum of phasic and tonic components) (red dashed), the estimated neural stimuli timings and amplitudes (black vertical lines with circle on top) for each of the participants. The estimation is done on phasic components using the proposed method; then, the previously separated tonic components are added to the estimated phasic components.