

Supplementary Information for “Variational decomposition autoencoding improves disentanglement of latent representations”

Contents

1	Method comparison	2
2	Simulated vowels dataset and generation algorithm	2
3	Hyperparameters	5
4	Self-supervised contrastive decomposition loss derivation and algorithm	5
5	Derivation of variational decomposition evidence lower bound	9
6	Decomposition generalizes disentanglement	12
7	Variational decomposition autoencoding pursues orthogonal components directions (by design)	13
8	Decomposition model algorithm	14
9	Disentanglement metrics	16
9.1	Mutual Information	16
9.2	Gaussian Correlation Norm	16
9.3	Disentanglement, Completeness, Informativeness	16
9.4	Modularity	17
10	Supplementary experimental results	17
10.1	SimVowels analysis	17
10.2	Real speech datasets analysis	22
11	Supplementary Figures	25

1 Method comparison

In Supplementary Table 1 we provide a qualitative comparison of the proposed DecVAE with other state-of-the-art variational and non-variational disentangled representation learning models, in terms of structure and functionality. In Supplementary Table 2 we further outline the variational autoencoding mathematical entities in the standard VAE and contrast them with our proposed variational decomposition autoencoding entities.

Supplementary Table 1: Qualitative Comparison with State-of-the-Art Disentangled Representation Models. VAE: Variational Autoencoder, MFL-VAE: Multiple Filtered Latent VAE, FHVAE: Factorized Hierarchical VAE, DSVAE: Disentangled VAE, PCA: Principal Component Analysis, ICA: Independent Component Analysis.

	DecVAE	VAE	β -VAE	MFL-VAE	FHVAE	DSVAE	PCA	ICA
Spectral/Frequency Components Disentanglement	✓							
Latent subspace representations	✓							
Self-Supervised Model	✓							
Time Series Sequence Modeling	✓			✓	✓	✓		
Generative Factors Disentanglement	✓	✓	✓	✓	✓	✓		
Model Type								
Encoder-only Autoencoder	✓							
Encoder-Decoder Autoencoder		✓	✓	✓	✓	✓	✓	
Eigenanalysis							✓	✓

Supplementary Table 2: Comparison of Variational Autoencoding and Variational Decomposition Autoencoding

	Variational Autoencoding	Variational Decomposition Autoencoding
Prior Distribution(s)	$p(z)$	$p(z) \leftarrow p(z_1), p(z_2), \dots, p(z_C)$
Posterior Distribution (unknown & intractable)	$p(z x)$	$\prod_{i=1}^C p(z_i x, \dots, z_{i-1}, z_{i+1}, \dots)$
Variational Approximation of the Posterior - Encoder	$q_\phi(z x)$	$\prod_{i=1}^C q_\phi(z_i x, \dots, z_{i-1}, z_{i+1}, \dots)$
Reconstruction	$\mathbb{E}_{q_\phi(z x)} [\log p_\theta(x z)]$	$\sum_{i=1}^C \mathbb{E}_{q_\phi(h_i x^{(i)}, \{h_k\}_{k \in \{1, \dots, C\} \setminus \{i\}})} [\log q_\phi(h x)]$
Disentanglement dynamics	L_{prior}	$L_{recon}, L_{ortho}, L_{prior}$

2 Simulated vowels dataset and generation algorithm

The motivation for creating the SimVowels dataset comes from the work of MFL-VAE, where the authors introduced the concept of creating a simulated vowels dataset to evaluate the disentanglement properties of their model in a two-generative factor speech dataset [1]. Here, we create a simulated vowels dataset according to Alg.1. We model a vowel as a combination of three distinct formant frequencies and a speaker as unique values of the fundamental frequency or pitch $F0$. We select a $F0 = 120Hz$ and subsequently sample a vocal tract factor v , which comes from an interval of values $v \in [0.7, 1.3]$. Each speaker has a unique vocal tract factor and their pitch is given by $f0 = F0/v$. The speaker information is encoded in the vowels’ power spectral density as a periodic excitation that modulates the vowels’ formants. We design each formant as a narrow-band component with a bandwidth of $200Hz$, and assign energies of $[1, 0.5, 0.3]$ to the 1st, 2nd and 3rd formants respectively.

To obtain a dataset of time domain speech signals given the above considerations, for each sample, we first randomly select a vocal tract factor v to model the speaker of a sequence. Then, the algorithm iteratively generates 1-second time domain signals by sampling a vowel from the available vowels and keeping the speaker factor v constant within the sequence. White noise of $SNR = 15dB$ is added to the resulting signals for more realistic results. The time domain generation happens by IIR filtering of a white noise $\mathcal{N}(0, 1)$ process, around the specified vowel’s

49 formants". The three formants are then summed and their FFT is calculated. A periodic excitation
50 based on the specific speaker's pitch f_0 is also generated as a time domain signal of periodic white
51 noise pulses. The periodic excitation signal is then convolved (multiplied in the frequency domain)
52 with the summation of the formants; the real part of the inverse FFT is then calculated to obtain
53 a time domain signal. Our approach aims to create a time domain signal whose frequency domain
54 properties are closely resembling real speech, rather than produce a realistic speech segment.

Algorithm 1 SimVowels Dataset Generation (pseudo-code)

Require: *sequence_length*, *dataset_size*, *duration*

```
1: vowels  $\leftarrow$  [ /a/, /e/, /I/, /aw/, /u/ ]
2: formants  $\leftarrow$  [[710, 1100, 2540], [550, 1770, 2490], [400, 1920, 2560], [590, 880, 2540], [310, 870, 2250]] # in Hz
3: F0base  $\leftarrow$  120
4: amplitudes  $\leftarrow$  [1.0, 0.5, 0.3] # [1st, 2nd, 3rd] formant
5: bandwidths  $\leftarrow$  [200, 200, 200] # in Hz
6: snr_db  $\leftarrow$  15 # contaminate waveforms with white noise
7: Fs  $\leftarrow$  16kHz
8: NFFT  $\leftarrow$  2048 # number of bins in FFT calculation

9: procedure GENERATEPERIODICEXCITATION(f0, duration)
10:   pulse_train  $\leftarrow$  [0, 0, ..., 0]duration*Fs
11:   pulse_interval  $\leftarrow$  Fs / f0
12:   for i  $\leftarrow$  0 to duration, step = pulse_interval do
13:     pulse_train[i : i + pulse_interval]  $\leftarrow$  sample( $\mathcal{N}(0, 1)$ , length = pulse_interval)
14:   end for
15:   pulse_train  $\leftarrow$  Butterworthfilter(pulse_train, lowcut = 65, highcut = Fs/2 - 1, Fs)
16:   excitation_spectrum  $\leftarrow$  FFT(pulse_train, NFFT)
17:   return excitation_spectrum
18: end procedure

19: procedure GENERATETIMEDOMAIN SIGNAL(vowel, vt_factor, duration, snr_db)
20:   scaled_freqs  $\leftarrow$  scale_formants(vowel, vt_factor) # divide each formant frequency with the vocal tract factor of the
   current speaker
21:   f0  $\leftarrow$  F0base / vt_factor
22:   signal  $\leftarrow$  -[0, 0, ..., 0]duration*Fs
23:   white_noise  $\leftarrow$  sample( $\mathcal{N}(0, 1)$ , length = duration * Fs)
24:   for all (freq, amp, bw)  $\in$  (scaled_freqs, amplitudes, bandwidths) do
25:     formant_wave  $\leftarrow$  amp * Butterworthfilter(white_noise, lowcut = freq - bw/2, highcut = freq + bw/2, Fs)
26:     signal  $\leftarrow$  signal + formant_wave
27:   end for
28:   formant_spectrum  $\leftarrow$  FFT(signal)
29:   excitation_spectrum  $\leftarrow$  GENERATEPERIODICEXCITATION(f0, duration, Fs)
30:   convolved_spectrum  $\leftarrow$  formant_spectrum  $\otimes$  excitation_spectrum
31:   time_domain_signal  $\leftarrow$  real(IFFT(convolved_spectrum)) # Real part of the complex spectrum
32:   if snr_db  $\neq$  None then
33:     return add_gaussian_noise(time_domain_signal, snr_db)
34:   else
35:     return time_domain_signal
36:   end if
37: end procedure

38: audio  $\leftarrow$  {}
39: vowel_list  $\leftarrow$  {}
40: speaker_vocal_tract_factor  $\leftarrow$  {}
41: for i  $\leftarrow$  0 to dataset_size - 1 do
42:   vt_factor  $\leftarrow$  sample( $\mathcal{U}[0.7, 1.3]$ ) # sample from the uniform distribution that starts from 0.7 and ends to 1.3
43:   audio_utt  $\leftarrow$  {}, vowels_utt  $\leftarrow$  {}, vt_factors_utt  $\leftarrow$  {}
44:   for j  $\leftarrow$  0 to sequence_length - 1 do
45:     vowel_label  $\leftarrow$  sample( $\mathcal{U}[\text{vowels}]$ )
46:     time_domain_signal  $\leftarrow$  GENERATETIMEDOMAIN SIGNAL(vowel, vt_factor, duration, snr_db)
47:     Append time_domain_signal to audio_utt
48:     Append vowel_label to vowels_utt
49:   end for
50:   Append vt_factor to speaker_vocal_tract_factor
51:   Append audio_utt to audio
52:   Append vowels_utt to vowels
53: end for
54: return {audio, vowels, speaker_vocal_tract_factor}
```

55 3 Hyperparameters

Supplementary Table 3: Decomposition variational autoencoder hyperparameters. Values are given in parentheses (,) whenever they differ for the frame and sequence variables.

Model Part	Parameter	Value (Simulations)	Value (Real data)	Description
Input	input	mel filterbanks	mel filterbanks	type of features used
	max_duration	4 seconds	7.15 seconds	maximum duration of the input utterances - shorter or longer utterances are clipped or zero-padded
	min_duration	0 seconds	1 second	minimum duration of the input utterances - shorter utterances are discarded
	n_mels	80	80	number of mel bands to use
DecVAE Encoder	mel_hops	4	4	divides the segment length, corresponds to the overlap between consecutive frames in mel extraction
	mel_norm	global	global	normalization of the mel spectrum in a batch
	NoC	(3,3)	(4,5)	number of components in the decomposition
	receptive_field	25 ms	25ms	encoder receptive field
DecVAE Latent Projections	stride	20ms	20ms	encoder stride
	use_learnable_OC_embeddings	true	true	assigns a cluster average OC, that accumulates knowledge from prior batches, if a sample was missing an OC in the current batch
	feature_norm	layer	layer	type of normalization in encoder layers
	layer_norm_eps	10^{-5}	10^{-5}	eps parameter of LayerNorm layer
DecVAE Encoder	dropout	0	0	dropout to use in fully connected layers in the network
	conv_bias	true	true	whether to use bias in the convolution layers
	conv_layers	7	7	number of convolutional layers
	conv_channels	(512,192)	(512,192)	number of channels in the convolution layers
DecVAE Encoder	conv_kernel	((10,3,3,3,3,2,2],[5,2,2,2,2,2,2])	((10,3,3,3,3,2,2],[5,2,2,2,2,2,2])	kernel sizes for each convolution layer
	skip_connections	false	false	whether to use residual skip connections
	enc_activation	gelu	gelu	type of activation function to use in the encoder
	hidden_dim	(512,192)	(512,192)	output of convolutional layers dimension
DecVAE Encoder	init	He	He	weight initialization strategy
	initializer_range	0.02	0.02	standard deviation of the weights in the initialized layers
DecVAE Latent Projections	proj_codevector_dim	(192,192)	(192,192)	dimension of the projecting H and aggregated OCs spaces Z
	proj_layers	(2,2)	(2,2)	number of fully-connected hidden projection layers
	proj_activation	gelu	gelu	type of activation function to use in the hidden projection layers
	proj_intermediate_dim	(512,512)	(512,512)	intermediate dimension in the hidden projection layers
DecVAE Latent Projections	self_attention	(false,true)	(false,true)	whether to use self-attention as a first pooling operation
	attention_heads	(None,6)	(None,12)	number of attention heads in self attention
	pooling_type	(None, avg)	(None, avg)	type of pooling used after self-attention
	agg_norm	batch	batch	type of normalization in the aggregation function layers
DecVAE Latent Projections	hidden_activation	gelu	gelu	type of activation function to use in the hidden projection layers
	z_latent_dim	48	64	dimensionality of the Z frame latent variable
	s_latent_dim	24	32	dimensionality of the S sequence latent variable
Optimization and training	batch_size	16	8	number of sequences in a training step
	ssl_frames_used_percent	50	50	percentage of frames used in the loss calculation
	decomp_loss_reduction	sum	sum	how to aggregate the loss over a batch (sum or avg)
	div_pos_weight	$1/\sqrt{N_{OC}}$	$1/\sqrt{N_{OC}}$	weight of each component-original divergence term
Optimization and training	div_neg_weight	$1/(\sum_{i=1}^{N_{OC}-1} 1)$	$1/(\sum_{i=1}^{N_{OC}-1} 1)$	weight of each component-component divergence term
	decomp_loss_S_weight	(None,10)	(None,10)	scaling of L_{recon}, L_{ortho} to ensure same order of magnitude of the loss terms
	prior_reg_weight	(10,400)	(10,400)	scaling of L_{prior} to ensure same order of magnitude of the loss terms
	β	0.1	0.1	best overall β value for prior approximation
Optimization and training	learning_rate	$(8 \times 10^{-5}, 10^{-4})$	$(8 \times 10^{-5}, 10^{-4})$	peak learning rate for the optimizer
	learning_schedule	(constant-with-warmup,linear)	(constant-with-warmup,linear)	learning schedule for adjusting the learning rate
	warmup_steps	5000	8650	steps during which the learning rate is increased according to the schedule
	training_steps	37500	65000	total steps to train for
Optimization and training	early_stop_warmup_steps	28000	41568	steps before early stop criterion takes effect
	optimizer	Adam	Adam	optimizer type used in training
	adam_b1	0.5	0.5	b1 parameter of the Adam optimizer
	adam_b2	0.999	0.999	b2 parameter of the Adam optimizer
Optimization and training	weight_decay	10^{-4}	10^{-4}	L2 regularization penalty on large weights
	adam_epsilon	10^{-6}	10^{-6}	epsilon parameter of the Adam optimizer
	clip_grad_value	2	2	clip gradients exceeding a maximum value
Hardware	GPU type	one NVIDIA RTX 6000 Ada Generation		

Supplementary Table 4: Decomposition methods hyperparameters and their values for simulations and real data experiments. Values are given in parentheses (,) whenever they differ for the frame and sequence variables.

Method	Parameter	Value (Simulations)	Value (Real data)	Description
Filter Decomposition (FD)	NoC	(3,3)	(4,5)	number of components assumed in all methods(frame,sequence)
	grouping_strategy	(equally dist., equally dist.)	(high frequencies, equally dist.)	strategy to group components if more than NoC detected (frame,sequence)
	peel_tolerance	10^{-5}	$(5 \times 10^{-6}, 10^{-6})$	delete a detected peak if amplitude less than this (frame, sequence)
	detection_intervals	5	6	split the spectrum in intervals and run peak detection in each of them
	power_law	0.8	1.4	power law exponent to split spectrum in uneven intervals
	global_thres	10^{-3}	10^{-3}	minimum absolute amplitude for a peak to be considered (max. is 1)
	nfft	512	512	number of FFT bins
	min_distance	150	50	minimum distance in Hz for peaks to be considered separate
	peak_bandwidth	200	400	bandwidth that a peak should have to qualify as separate
	prominence_thres	10^{-2}	10^{-2}	relative amplitude to the highest peak in a neighborhood that a peak must have to be considered as a separate peak
Empirical Mode Decomposition (EMD)	N_peaks	6	2	number of peaks to select inside each spectrum interval
	bw_order	2	2	order of the IIR Butterworth filter
	spline_kind	pcchip	pcchip	interpolation method to connect extrema
	max_iter	1000	1000	iterations after which to stop if no convergence, per single sifting
	energy_ratio_thr	5×10^{-3}	5×10^{-3}	energy ratio threshold, per component
	std_thr	0.1	0.1	standard deviation threshold, per component
	svar_thr	10^{-2}	10^{-2}	scaled variance threshold, per component
	total_power_thr	10^{-4}	10^{-4}	total power threshold per decomposition
	range_thr	10^{-3}	10^{-3}	amplitude range threshold after scaling, per decomposition
	extrema_detection	simple	simple	method used to find extrema
Variational Mode Decomposition (VMD)	alpha	1500	1500	balancing parameter of the data-fidelity constraint
	tau	0	0	time-step of the dual ascent
	DC	false	false	whether to keep the DC as a component
	init	random	random	position initialization for the modes in the frequency domain
Empirical Wavelet Transform (EWT)	tol	10^{-8}	10^{-8}	convergence criterion tolerance
	completion	false	false	whether to complete the modes if less than specified found
	filter	gaussian	gaussian	type of filter to use for regularizing the spectrum
	filter_length	10	10	width of the filter
	filter_sigma	2	2	standard deviation of the filter
	log_spectrum	false	false	whether to work on the log spectrum
Empirical Wavelet Transform (EWT)	detect	locmax	locmax	method to segment the spectrum

56 4 Self-supervised contrastive decomposition loss derivation and algorithm

57 We design a self-supervised contrastive latent decomposition optimization algorithm that leverages
58 the structural biases of orthogonality and reconstruction enforced by $\mathcal{D}_w^C$ in the input domain X ,

Supplementary Table 5: Supporting methods hyperparameters.

Model Part	Parameter	Value (Simulations)	Value (Real data)	Description
Input	input	mel filterbanks	mel filterbanks	type of features used
	max_duration	4 seconds	7.15 seconds	maximum duration of the input utterances - shorter or longer utterances are clipped or zero-padded
	min_duration	0 seconds	1 second	minimum duration of the input utterances - shorter utterances are discarded
	n_mels	80	80	number of mel bands to use
	mel_hop	4	4	divides the segment length, corresponds to the overlap between consecutive frames in mel extraction
	mel_norm	global	global	normalization of the mel spectrum in a batch
ICA	components	20	20	number of independent components in the latent space
	whitening	unit-variance	unit-variance	whitening matrix scaling
	tol	10^{-4}	10^{-4}	tolerance for un-mixing matrix convergence
	solver	svd	svd	solver to use for the whitening
	iter	200	200	iterations during fit
PCA	components	20	20	number of principal components in the latent space
	whiten	false	false	whether to whiten the data
	solver	covariance_eigh	auto	solver to use for the singular value decomposition
	kernel_gamma	0.1	0.1	gamma parameter of kernel-PCA (when a kernel is used)
VAE-type models	receptive_field	25	25	encoder receptive field (milliseconds)
	stride	20	20	encoder stride (milliseconds)
	enc_norm	batch	batch	type of normalization layer used in the encoder
	dec_norm	none	none	type of normalization layer used in the decoder
	dropout	0	0	dropout to use in fully connected layers in the network
	fc_layers	7	7	number of fully-connected layers in both encoder and decoder sides
	fc_dims	[2048,1024,512,512,512,256,256]	[2048,1024,512,512,512,256,256]	number of neurons in each fully-connected layers
	fc_bias	true	true	bias in the fully-connected layers
	skip_connections	false	false	whether to use residual skip connections
	enc_activation	gelu	gelu	type of activation function to use in the encoder
	hidden_dim	192	192	output of convolutional layers dimension
	init	He	He	weight initialization strategy
	initializer_range	0.02	0.02	standard deviation of the weights in the initialized layers
	z_latent_dim	48	64	dimensionality of the Z latent variable
VAE Optimization and training	batch_size	16	8	number of sequences in a training step
	vae_loss_frames_percent	100	100	percentage of frames used in the loss calculation
	β	0.1	0.1	best overall β value for prior approximation
	learning_rate	$(8 * 10^{-5}, 10^{-4})$	$(8 * 10^{-5}, 10^{-4})$	peak learning rate for the optimizer
	learning_schedule	(constant_with_warmup,linear)	(constant_with_warmup,linear)	learning schedule for adjusting the learning rate
	warmup_steps	5000	8650	steps during which the learning rate is increased according to the schedule
	training_steps	37500	65000	total steps to train for
	early_stop_warmup_steps	28000	41568	steps before early stop criterion takes effect
	early_stop_delta	0.2	0.2	minimum difference percentage in subsequent steps loss calculation to not trigger early stop
	early_stop_patience	5	5	number of consecutive epochs with no improvement that triggers early stop
	optimizer	Adam	Adam	optimizer type used in training
	adam_b1	0.5	0.5	b1 parameter of the Adam optimizer
	adam_b2	0.999	0.999	b2 parameter of the Adam optimizer
	weight_decay	10^{-4}	10^{-4}	L2 regularization penalty on large weights
	adam_epsilon	10^{-6}	10^{-6}	epsilon parameter of the Adam optimizer
	clip_grad_value	2	2	clip gradients exceeding a maximum value
Hardware	GPU type	one NVIDIA RTX 6000 Ada Generation		

and aims to retain these structural dynamics in subsequent latent spaces H . Specifically, the following contrastive objective is formulated and minimized:

$$\mathcal{L}_{SSLD_{ec}} = \mathcal{L}_{recon} + \mathcal{L}_{ortho} \quad (1)$$

We define the following distributions of original signal and its components $\{X^{(0)}, X^{(1)}, X^{(2)}, \dots, X^{(C)}\}$ after they are propagated through a DecVAE in a latent space H :

$$\begin{aligned} P &= \text{softmax}(G(f_L(f_C(X^{(0)})))) \\ R_i &= \text{softmax}(G(f_L(f_C(X^{(i)})))) \\ \text{softmax}(z_i) &= \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \end{aligned} \quad (2)$$

where G is an activation function, f_L is a neural network projector, f_C is an neural network encoder. The Jensen-Shannon divergence [2] between two distributions X, Y is defined as:

$$D_{JS}(X||Y) = \frac{1}{2}(D_{KL}(M || X) + D_{KL}(M || Y)) \quad (3)$$

where $M = \frac{1}{2}(X + Y)$ and D_{KL} is the Kullback-Leibler divergence.

We define separate D_{JS} terms for “positive” (component-original) and “negative” (component-component) pairs:

$$\begin{aligned}
D_{JSpos}^{(i)} &= \frac{1}{2}(D_{KL}(M_i \parallel P + \epsilon) + D_{KL}(M_i \parallel R_i + \epsilon)), i \in \{1, \dots, C\} \\
D_{JSneg}^{(i,j)} &= \frac{1}{2}(D_{KL}(M_{ij} \parallel R_i + \epsilon) + D_{KL}(M_{ij} \parallel R_j + \epsilon)), i \neq j, i, j \in \{1, \dots, C\} \\
M_i &= \frac{1}{2}(P + R_i) \\
M_{ij} &= \frac{1}{2}(R_i + R_j)
\end{aligned} \tag{4}$$

68 where ϵ is a small constant for numerical stability. Using D_{JSpos} and D_{JSneg} we can formulate a
69 cross-entropy objective for N batched latent samples of dimensionality d in $H^{d \times N}$:

$$\begin{aligned}
CE_{pos}^{(i)} &= D_{KL}(D_{JSpos}^{(i)} \parallel p), i \in \{1, \dots, C\}, \\
&\text{where } p(y) = \epsilon, \forall y \in \{1, \dots, d\}, H^{d \times N} \\
CE_{neg}^{(i,j)} &= D_{KL}(D_{JSneg}^{(i,j)} \parallel n), i \neq j, i, j \in \{1, \dots, C\}, \\
&\text{where } n(y) = 1, \forall y \in \{1, \dots, d\}, H^{d \times N}
\end{aligned} \tag{5}$$

70 In practice, we use a numerical stabilization heuristic, limiting JS divergences to constrain their
71 values in the $[0, 1]$ range. The final self-supervised learning decomposition loss is:

$$\mathcal{L}_{SSLD_{ec}} = \left(\sum_{i=1}^C w_{pos}^i CE_{pos}^{(i)} \right) + \left(\sum_{i=1}^C \sum_{j=i+1}^C w_{neg}^{(i,j)} CE_{neg}^{(i,j)} \right) \tag{6}$$

72 where right-hand side terms are L_{recon} and L_{ortho} respectively.

73 The use of $\mathcal{L}_{SSLD_{ec}}$ for training neural networks with gradient-based optimization, is described
74 in PyTorch pseudo-code for a single training step in Algorithm 2.

Algorithm 2 Self-Supervised Contrastive Decomposition Loss (PyTorch Pseudo-code)

```

1: Require inputs: Batched multi-scale decomposed (masked) time-series utterance embeddings of original signal  $\mathbf{h}^0 \in \mathbb{R}^{[1, B, F, H]}$ , components  $\mathbf{h}^C \in \mathbb{R}^{[C, B, F, H]}$ , number of components  $C$ , batch size  $B$ , number of frames in utterance  $F$ , hidden dimension  $H$ , number  $N$  of masked frames to use, mask  $L \in \mathbb{R}^{[B, F]}$  that contains indices to be used (not flagged as empty due to zero padding / missing component), positive and negative weights  $w_{pos}, w_{neg}$  for weighting loss terms

2: begin
3:  $\epsilon \leftarrow 10^{-8}$ 
4:  $D \leftarrow [h^0, h^{1:C}], D \in \mathbb{R}^{[C+1, B, F, H]}$  # decomposition matrix
5: for utterance embedding  $\{h\}_{b=1}^B$  do
6:   Sample  $D$  masked frames from the same utterance
7:    $l^{(b)} \leftarrow L[L == b]$  # keep indices of the current batch entry
8:    $l^{(f)} \leftarrow \text{randperm}(l^{(b)})[1 : N]$  # shuffle and select  $N$  frames
9:    $D^{(b)} \leftarrow D[:, b, l^{(f)}, :]$  # shape(C+1, N, H)
10:   $h_b^{(0)} = D^{(b)}[0]$ 
11:   $h_b^{(1:C)} = D^{(b)}[1 : C]$ 
12:   $p = \exp(h_b^{(0)} / \sum_{i=1}^H h_b^{(0)}[i])$  # softmax over the hidden dimension to get probabilities  $\in (0, 1)$ 
13:  for component  $\{c\}_{j=1}^C$  do
14:     $q = \exp(h_b^{(j)} / \sum_{i=1}^H h_b^{(j)}[i])$  # softmax - component distribution
15:     $m = \log(0.5 * (p + q))$ 
16:     $D_{JS_{pos}}^{(b,j)} = \text{clamp}(0.5 * (D_{KL}(m, p + \epsilon) + D_{KL}(m, q + \epsilon)), \min = 0, \max = 1)$ 
17:    for component  $\{c\}_{k=j+1}^C$  do
18:       $r = \exp(h_b^{(k)} / \sum_{i=1}^H h_b^{(k)}[i])$  # softmax - other component distribution
19:       $m = \log(0.5 * (r + q))$ 
20:       $D_{JS_{neg}}^{(b,j,k)} = \text{clamp}(0.5 * (D_{KL}(m, q + \epsilon) + D_{KL}(m, r + \epsilon)), \min = 0, \max = 1)$ 
21:    end for
22:  end for
23: end for

24:  $\text{target}_{pos} = [0, 0, \dots, 0]^{(N)}$  # targets for the cross-entropies
25:  $\text{target}_{neg} = [1, 1, \dots, 1]^{(N)}$ 
26: for component  $\{j\}_1^C$  do
27:   $jsd_{pos} = D_{JS_{pos}}^{(:,j)}$  # across all batch entries
28:   $CE_{pos}^{(j)} = -\frac{1}{N} \sum_{b=1}^B [\log(1 - jsd_{pos}^{(b)})]$ 
29:  for pair in  $\{k\}_{j+1}^C$  do
30:     $jsd_{neg} = D_{JS_{neg}}^{(:,j,k)}$  # across all batch entries
31:     $CE_{neg}^{(j,k)} = -\frac{1}{N} \sum_{b=1}^B [\log(1 - jsd_{neg}^{(b)})]$ 
32:  end for
33: end for

34:  $L_{recon} = \sum_{i=1}^C w_{pos}^{(i)} CE_{pos}^{(i)}$ 
35:  $L_{ortho} = \sum_{j=1}^C \sum_{k=j+1}^C w_{neg}^{(j,k)} CE_{neg}^{(j,k)}$ 
36:  $L_{SSLD_{Dec}} = L_{recon} + L_{ortho}$ 
37: return

```

5 Derivation of variational decomposition evidence lower bound

We assume a latent generative model from $Z^k \rightarrow H^d$ with parameters $\psi \subset \phi$, where ϕ is the encoder inference model from X^t to Z , and for latents z, h , $h = \mathbf{w}^{-1}(z)$ holds, where t is the dimensionality of the input X' to the encoder, d is the dimensionality of latent representations distributions h_i , and k is the dimensionality of latent space Z . The latent generative model aims to maximize the likelihood of h_i , $i \in \{1, C\}$ given z . Below we derive a lower bound on the collective log-likelihood of individual subspaces likelihoods $r_\psi(h_i)$ that incorporates all disentanglement mechanics of variational decomposition autoencoding, i.e., latent reconstruction, orthogonality of subspaces, and Gaussian prior approximation.

Since $h = \mathbf{w}^{-1}(z)$ holds and we want to tie the Gaussian prior approximation in the z space, to a latent generative model $r_\psi(h|z)$, the invertibility of $\mathbf{w}$ is important; given that ψ is parameterized as a linear neural layer with a weight matrix $W^{d \times k}$ that maps inputs H^d to outputs in Z^k , with $d > k$, we condition our lower bound derivation on the assumption that data in h live in lower-dimensional manifold M .

Assumption 1 (Invertibility of $\mathbf{w}$). *Latent representations h lie on an k -dimensional manifold $M \subset \mathbb{R}^d$ and the encoder $\mathbf{w}$ restricted to M is injective. Therefore, there exists an inverse map $\mathbf{w}^{-1} : \mathbf{w}(M) \rightarrow M$, such that $\mathbf{w}^{-1}(\mathbf{w}(h)) = h$.*

The marginal probabilities of subspaces h_i over h are:

$$\sum_i r_\psi(h_i) = \sum_i \int_h r_\psi(h_i|h)p(h)dh. \quad (7)$$

If we change the integration variable from h to z , and considering $h = \mathbf{w}^{-1}(z)$, then:

$$\begin{aligned} \sum_i \int_h r_\psi(h_i|h)p(h)dh &= \sum_i \int_z r_\psi(h_i|\mathbf{w}^{-1}(z))p(\mathbf{w}^{-1}(z))\left|\frac{\partial h}{\partial z}\right|dz \\ &= \sum_i \int_z r_\psi(h_i|z)p(z), \end{aligned} \quad (8)$$

where $r_\psi(h_i|z) := r_\psi(h_i|\mathbf{w}^{-1}(z))$, and $p(z) := p(\mathbf{w}^{-1}(z))\left|\frac{\partial h}{\partial z}\right|$. Note that the Jacobian $\left|\frac{\partial h}{\partial z}\right|$ is constant in z as $\mathbf{w}$ is a linear layer $z = \mathbf{w}(h) = Ah + b$.

Then, the variational lower bound for the marginal probability of the latent projection h_n^d of an input observation x_n , can be derived as follows, by using the convexity of the log function and applying the Jensen inequality:

$$\begin{aligned}
& \sum_i \log(r_\psi(h_i)) \\
&= \sum_i \log \int_z q_\phi(z|x, h_i) \frac{r_\psi(h_i|z)p(z)}{q_\phi(z|x, h_i)} dz \\
&\geq \mathcal{L}_{ELBO}(\psi, \phi; x, z) \\
&= \sum_i \mathbb{E}_{z \sim q_\phi(z|x, h_i)} [\log(r_\psi(h_i|z))] + \sum_i \mathbb{E}_{z \sim q_\phi(z|x, h_i)} [\log(\frac{p(z)}{q_\phi(z|x, h_i)})] \\
&= \sum_i \mathbb{E}_{z \sim q_\phi(z|x, h_i)} [\log(r_\psi(h_i|z))] + \sum_i \int_z \log(\frac{p(z)}{q_\phi(z|x, h_i)}) \\
&= \sum_i \mathbb{E}_{z \sim q_\phi(z|x, h_i)} [\log(r_\psi(h_i|z))] - \sum_i D_{KL}(q_\phi(z|x, h_i)||p(z)) + \sum_i \log(|\det A^{-1}|)
\end{aligned} \tag{9}$$

99 Because $\mathbf{w}$ is the weight matrix of a linear layer, the relevant Jacobian determinant is the volume
 100 form induced by the restriction $A|_M$. This determinant is constant for fixed A , so it can be absorbed
 101 as a constant term in the ELBO derivation and, hence $\mathcal{L}(\psi, \phi; x_n)$ is still a lower bound on the
 102 log-likelihood of h_i . Then, the above equation gives us the variational lower bound for the assumed
 103 latent generative model ψ in subspaces h_i . To get the full decomposition evidence lower bound
 104 (DELBO), we augment the above ELBO with the orthogonality term L_{ortho} of subspaces h_i (see
 105 Supplementary Section ‘Self-supervised contrastive decomposition loss derivation and algorithm’):

$$\begin{aligned}
& \sum_i \log(r_\psi(h_i)) \\
&\geq \mathcal{L}_{DELBO}(\phi; x, z) \\
&= \mathcal{L}_{ELBO}(\psi, \phi; x, z) - \sum_i \sum_j D_{KL}(h_i||h_j) \\
&= \sum_i \mathbb{E}_{z \sim q_\phi(z|x, h_i)} [\log(r_\psi(h_i|z))] - \sum_i \sum_j D_{KL}(h_i||h_j) \\
&\quad - \beta \sum_i D_{KL}(q_\phi(z|x, h_i)||p(z)) + const. \\
&= \sum_i \mathbb{E}_{z \sim q_\phi(z|x, h_i)} [\log(r_\psi(h_i|z))] - L_{ortho} - \beta L_{prior} + const.
\end{aligned} \tag{10}$$

106 which is still a lower bound on $\sum_i \log(r_\psi(h_i))$, as D_{KL} is lower-bound by 0 and $\beta > 0$, and since
 107 $\psi \subset \phi$, we can express the DELBO only through ϕ . We reach the final form of the DELBO by
 108 also adding the weighting terms on the latent reconstruction and the L_{ortho} terms; because these
 109 weights are positive and < 1 , the DELBO is still a lower bound of the log-likelihood:

$$\begin{aligned}
& \sum_i \log(r_\psi(h_i)) \\
&\geq \mathcal{L}_{DELBO}(\phi; x, z, w_{pos}, w_{neg}, \beta) \\
&= \sum_i w_{pos}^{(i)} \mathbb{E}_{z \sim q_\phi(z|x, h_i)} [\log(r_\phi(h_i|z))] - \sum_i \sum_j w_{neg}^{(i,j)} D_{KL}(h_i||h_j) \\
&\quad - \beta \sum_i D_{KL}(q_\phi(z|x, h_i)||p(z)) + const. \\
&= \sum_i w_{pos}^{(i)} \mathbb{E}_{z \sim q_\phi(z|x, h_i)} [\log(r_\phi(h_i|z))] - L_{ortho} - \beta L_{prior} + const.
\end{aligned} \tag{11}$$

DELBO is a “looser” lower bound to the marginal log-likelihood of the observation h_n compared to the classic variational autoencoding ELBO [3], due to the extra orthogonality constraint that we add in a similar manner to how FactorVAE inserts the total correlation term in the ELBO [4], and the constant terms that are a product of our assumptions on the latent generative model and the connection of spaces h , z .

In practice, to avoid using a latent generative model we perform the latent reconstruction SSL task to maximize likelihood of individual h_i subspaces w.r.t h (or z) (see Supplementary Information ‘Self-supervised contrastive decomposition loss derivation and algorithm’) and optimize the below DELBO-inspired objective:

$$\mathcal{L}_{DELBO_i}(\phi; x, z, w_{pos}, w_{neg}, \beta) = L_{recon} + L_{ortho} - \beta L_{prior} \quad (12)$$

Note that the sign of L_{ortho} has changed; this is because in practice, L_{recon} and L_{ortho} are competing terms forming an adversarial objective; in ‘Self-supervised contrastive decomposition loss derivation and algorithm’ we show how we use the Jensen-Shannon and cross-entropy quantities to transform the adversarial objective into a typical stable minimization task.

Eq.(12) indeed connects to our derived DELBO; the positive cross-entropy term CE_{pos} that we calculate in Eq.(5) can be expressed as a log-likelihood term over the Jensen-Shannon divergences distributions with support $|d|$ in space H^d :

$$\begin{aligned} CE_{pos} &= \sum_i w_{pos}^{(i)} D_{KL}(D_{JS_{pos}}^{(i)} || p) \\ &= \sum_i w_{pos}^{(i)} \mathbb{E}_{D_{JS_{pos}}^{(i)}} [\log(D_{JS_{pos}}^{(i)}) - \log(p)] \\ &= \sum_i w_{pos}^{(i)} \mathbb{E}_{D_{JS_{pos}}^{(i)}} [\log(D_{JS_{pos}}^{(i)})] + \sum_i w_{pos}^{(i)} \log|d| \end{aligned} \quad (13)$$

In other words, the latent generative model is expressed as the Jensen-Shannon divergence between subspaces h_i and subspace h (or h_0), expressing a distribution-level reconstruction up to a constant, whereas the log-likelihood is a point-level reconstruction. Although the algebraic connection between the quantities CE_{pos} and $\mathbb{E}_{z \sim q_\phi(z|x,h)} [\log(r_\phi(h_i|z))]$ is not direct, they both express latent information reconstruction dynamics within the same latent space; having a single model ϕ that acts both as the encoder and the decoder, strengthens this connection and allows us to view $D_{JS_{pos}}$ as a target distribution whose likelihood we want to maximize w.r.t. a reference distribution p .

6 Decomposition generalizes disentanglement

In [5], Mathieu *et al.* formulate a generalized disentanglement problem for VAEs as one where the goal is to decompose the latent representations; they pose the decomposition as arising from the fulfillment of two factors: (a) the latent encodings have an appropriate level of overlap, (b) the aggregate encoding of the data conforms to a desired structure (represented through the prior) [5]. Although our approach differs significantly from this hypothesis, the high level outcome that we also demonstrate to have achieved in this work is the same. Decomposition can be viewed as a generalization of disentanglement: rather than requiring strict statistical independence of latent factors, it allows us to impose a specific, desired structure on the latent representations learned. By designing structural constraints, such as orthogonality, decomposition promotes disentanglement while extending it to richer forms of representation learning. Mathieu *et al.* posit that this desired structure can arise as a result of a sufficient overlap in the latent representation in conjunction to the quality of regularisation of the prior distribution approximation [5]. Without balance of these two factors or effect of only one of them, the latent space is under no constraint to exhibit the desired structure.

Decomposition is fundamentally subjective as the nature of the features that are learned heavily depend on how we use the inductive biases that structure the latent space (in our case by selecting the signal decomposition-inspired dynamics and the prior approximation). β -VAE [6] for instance, encourages disentanglement by directly controlling the level of overlap (factor (a) [5]) through β in the latent space, without other constraints. They reformulate the variational evidence lower bound (ELBO) in the β -VAE model to also satisfy the second factor, i.e., conforming to the desired prior structure:

$$\mathcal{L}_{ELBO} = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \beta D_{KL}(q_\phi(z|x) \parallel p(z)) - \alpha \mathbb{D}(q_\phi(z), p(z)) \quad (14)$$

where the divergence $\alpha \mathbb{D}(q_\phi(z), p(z))$ is a term used in Mathieu *et al.* [5] to control the regularization. Our signal decomposition approach actually makes the third term in Eq. (14) more explicit to conform to a decomposed latent subspaces structure through the first and second terms in the below DELBO formulation (Supplementary Eq.(11)):

$$\begin{aligned} \mathcal{L}_{DELBO}(\phi; x, z, w_{pos}, w_{neg}, \beta) = & \sum_i w_{pos}^{(i)} \mathbb{E}_{z \sim q_\phi(z|x, h)}[\log(r_\phi(h_i|z))] \\ & - \sum_i \sum_j w_{neg}^{(i,j)} D_{KL}(D_{JS}(h_i, h_j) || n) \\ & - \beta \sum_i D_{KL}(q_\phi(z|x, h_i) || p(z)) \end{aligned} \quad (15)$$

Here, we have expressed the divergence term $\alpha \mathbb{D}(q_\phi(z), p(z))$ as divergence terms between the latent subspaces h_i . The Jensen-Shannon divergences weights $w_{pos}^{(i)}, w_{neg}^{(i,j)}$ that we impose on the reconstruction and orthogonality terms in Eq.(15), essentially correspond to α and control the exact structure that we desire over the posterior. In Supplementary Section 4 we have derived the first and second terms exclusively from KL divergence terms between latent subspaces, hence our DELBO first and second terms can be expressed as a divergence term. Our DELBO formulation enforces a much more specific structure on the latents, that might be more fit for time-series given the frequency domain properties that influence the generative factors in time series.

7 Variational decomposition autoencoding pursues orthogonal components directions (by design)

In addition to the above, the structural properties of orthogonality and reconstruction (Methods, Eq.(2)) that the second term in Eq.(15) expresses, hints a connection of DecVAEs to PCA[7, 8] and ICA [9]. The connection of VAEs to these methods has been previously analyzed in [10], where Rolinek *et al.* show that VAE exhibits local orthogonality of the decoder and along with the reconstruction mechanism, this behavior matches the way that PCA embeddings are chosen. The prior approximation term in both β -VAEs and DecVAEs i.e., $\beta \sum_i D_{KL}(q_\phi(z|x, h_i)||p(z))$, enforces the covariance matrix to be diagonal [10].

A common denominator in PCA, ICA and VAEs, is that the setting is unsupervised, generative factors are unknown, and thus these methods rely on statistical properties to learn a latent representation of the prior distributions. PCA isolates sources of variance and projects the data into a space of principal components [7]; ICA recovers a factorized representation [11].

PCA projects input space X to latent space Z by applying a linear mapping function through multiplication with a matrix W and apply the inverse W^{-1} to recover X from Z , and formulates a reconstruction objective [11]:

$$\begin{aligned} z &= xW \\ \tilde{x} &= zW^{-1} \\ \mathcal{L}_{PCA} &= \|\tilde{x} - x\|_2^2 \end{aligned} \tag{16}$$

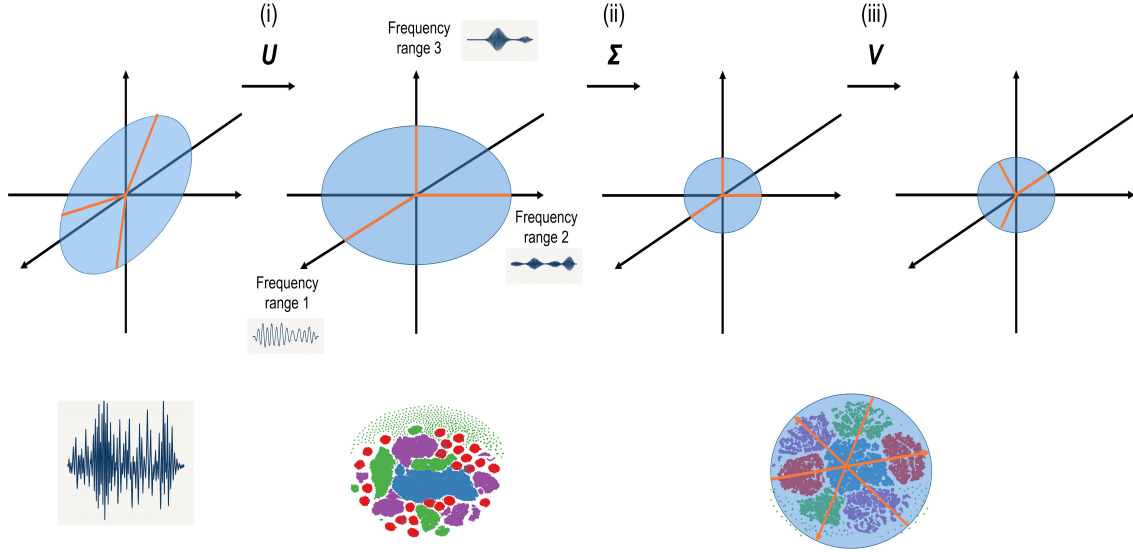
PCA also enforces dimensions in Z to be uncorrelated, by orthogonal columns in W . Specifically, variables that are jointly Gaussian are independent if they are uncorrelated [12]. In our variational decomposition autoencoding problem formulation (Methods "Problem formulation"), we have assumed latent subspaces z_i that create the subspace z , and have the orthogonality and reconstruction properties. In our experiments, we demonstrate that the learned approximations to the true latent subspaces are indeed uncorrelated (low MI, GCN in Supplementary Fig.7, Supplementary Tables 6, 9, 12, 10). This decorrelation or orthogonality leads to independent components in the latent space, given that the subspaces are Gaussian. While PCA components correspond to the variance that they can explain in the input, DecVAE learns components that correspond to the variance in the time- and frequency-domain inside of each subspace. Our problem formulation and the structure of the DELBO, ensures that these components are orthogonal and independent.

If we were to express DecVAE in the terms of a singular value decomposition (SVD), DecVAE can be said to embed data in latent subspaces, where every subspace z_i is designed to have maximal variance w.r.t. a specific frequency range and be orthogonal to other subspaces z_j , while reconstructing z . Defining an SVD linear transformation matrix $M = V\Sigma U^T$, where $U : \mathbb{R}^n \rightarrow \mathbb{R}^n$ a rotation matrix of the input space X , $\Sigma : \mathbb{R}^n \rightarrow \mathbb{R}^d$ a scale-and-embed diagonal matrix, and $V : \mathbb{R}^d \rightarrow \mathbb{R}^d$ a rotation matrix of the latent space, PCA can be expressed in terms of these matrices [10]. For DecVAE, we can define:

$$\begin{aligned} \mathcal{D}_w^C : \mathbb{R}^n &\rightarrow \mathbb{R}^n, \quad \text{the decomposition model (analogous to } U) \\ g(f_L(f_H(f_W()))) : \mathbb{R}^n &\rightarrow \mathbb{R}^d, \quad \text{the DecVAE encoder (analogous to } \Sigma) \end{aligned} \tag{17}$$

The difference with PCA is that in our case, U, Σ are non-linear operators; U takes the form of the decomposition model that transforms the input space into a space of orthogonal components according to variance in frequency, Σ takes the form of the DecVAE architecture (Methods "Decomposition variational autoencoder"), and V corresponds to the loadings or the directions of maximum

frequency-variance. Our results suggest that DecVAE frequency disentanglement is invariant to the exact rotation applied by V in the latent space Z . As seen in Supplementary Figures 5,8, although different rotation matrices U can be used in the input space X (FD, EMD[13],VMD[14],EWT[15]), the disentanglement of the frequency variables is not affected, and independent axes of variation exist in the representation. A geometrical intuition of this parallelism between DecVAE, SVD and PCA is given in Supplementary Fig. 1, adapted for DecVAE from [10]. As discussed by Rolinek *et al.*, this behavior is also observed in VAEs that pursue principal directions, although this effect arises not by design of the VAEs. In contrast, our variational decomposition autoencoding has been designed to pursue directions of variation as dictated by a decomposition model.



Supplementary Figure 1: Geometric interpretation of the DecVAE transformation from input space X to latent space Z , by expressing DecVAE components as singular value decomposition (SVD) equivalent matrices. Three steps are illustrated where SVD matrices are sequentially applied on the data. Step (i) is expressed by matrix U that in DecVAE is expressed by the decomposition model $\mathcal{D}_w^C$ and is connected to the orthogonal space learned by separating different time-frequency regions. Step (ii) expressed by a matrix Σ corresponds to the DecVAE neural network that scales and embeds the representation by retaining the independent and orthogonal components direction in the data. Step (iii) accounts for a rotation matrix that may be part of the DecVAE application, that though has not an effect on the disentanglement of the latent space, making the representation rotation invariant.

8 Decomposition model algorithm

The decomposition model was described in “Methods-Decomposition model and self-supervised learning masking”. The decomposition model $\mathcal{D}_w^C$ is a key component of DecVAE (Supplementary Fig.2) and enables DecVAE to facilitate disentanglement in accordance to VDA. $\mathcal{D}_w^C$ can support any existing decomposition method; first a peak detection is applied inside pre-defined detection intervals to find modes or components of interest. Our custom filter decomposition (FD) uses the detected peaks and filters around them to extract oscillatory components (*OCs*). The latter need to satisfy a minimal power spectral density amplitude in order to be kept as meaningful activity and not be discarded as noise. The modes are also used in cases where fewer components than specified are found, to understand which frequency ranges might be missing and fill with zeros the correct *OC*. In case the found *OCs* are more than the number C specified, a merging strategy has to be followed to end up with exactly C components. For that, one of three grouping strategies

can be used, i.e., grouping first the higher frequencies in a single component, grouping first the low frequencies in a single component, or group uniformly neighboring components to end up with C components. The repetition of this process for every frame extends the input batch X with shape $(1, B, F, S)$ to a batch X'' with shape $(C + 1, B, F, S)$, where C the number of components, B the batch size, F the number of frames in the sequence, S the frame length.

Algorithm 3 Decomposition model algorithm

```

1: Require inputs: Batched time-series utterance  $\mathbf{X} \in \mathbb{R}^{[B, F, S]}$ , batch size  $B$ , number of frames in utterance  $F$ , segment length  $S$ , sampling
   rate  $F_s$ , decomposition method  $\mathcal{D} \in \{FD, EWT, VMD, EMD\}$ , number of oscillatory components  $C$ , parameters  $w$  of each decomposition,
   length of the Fast Fourier Transform to use in spectral operations  $NFFT$ 
2: if  $\mathcal{D} == FD$  then
3:   Also require: list of peak detection intervals  $intervals_{det}$ , grouping strategy  $G$ , detected components tolerance  $psd\_tolerance$ 
4: end if
5: procedure FILTERDECOMPOSITION( $x, modes$ )
6:    $OCs \leftarrow []$  # initialize oscillatory components
7:   for  $i \leftarrow iterate(modes)$  do
8:      $x_{filtered} \leftarrow \text{Butterworth}_{filter}(x, lowcut = i[0], highcut = i[1], F_s)$  # apply bandpass filter
9:     Append  $x_{filtered}$  to  $OCs$ 
10:   end for
11: end procedure
12:  $X'' \leftarrow \text{zeros}(C + 1, B, F, S)$  # tensor of zeros with shape
13: for  $\{x\}^{(b, f, s)} \leftarrow iterate(X)$  do
14:    $modes \leftarrow \text{peakDetection}(x^{(b, f, s)}, intervals_{det})$ 
15:   if  $modes == []$  then
16:     continue
17:   end if
18:   if  $\mathcal{D} == FD$  then
19:      $OCs \leftarrow \text{FILTERDECOMPOSITION}(x^{(b, f, s)}, modes)$ 
20:   else
21:      $OCs \leftarrow \mathcal{D}(x^{(b, f, s)}, w)$ 
22:   end if
23:    $OCs\_PSD \leftarrow \text{WelchPSD}(OCs, NFFT, F_s)$ 
24:   for  $OC, o \leftarrow iterate(OCs\_PSD)$  do
25:     if  $\max(OC) < psd\_tolerance$  then
26:        $OC \leftarrow []$  # delete found OCs with a pre-defined tolerance threshold
27:     end if
28:   end for
29:   if  $\text{shape}(OCs)[0] > C$  then
30:     if  $G \leftarrow \text{low\_frequencies\_first}$  then
31:        $OCs[0] \leftarrow \text{sum}(OCs[: \text{shape}(OCs)[0] - (C - 1)])$  # superpose low frequencies first to reach desirable number of components  $C$ 
32:        $OCs[1 : \text{shape}(OCs)[0] - (C - 1)] \leftarrow []$ 
33:     else if  $G \leftarrow \text{high\_frequencies\_first}$  then # superpose high frequencies first
34:        $OCs[C - 1] \leftarrow \text{sum}(OCs[C - 1 :])$ 
35:        $OCs = OCs[: C]$ 
36:     else if  $G \leftarrow \text{equally\_distribute}$  then # superpose frequencies uniformly with nearest neighbors
37:        $new\_OCs \leftarrow []$ 
38:        $step \leftarrow \text{round}(\text{shape}(OCs)[0] / C)$ 
39:       for  $i \leftarrow iterate(C)$  do
40:          $start \leftarrow i * step$ 
41:         if  $i \neq C - 1$  then
42:            $end \leftarrow (i + 1) * step$ 
43:         else
44:            $end \leftarrow \text{shape}(OCs)[0]$ 
45:         end if
46:          $new\_OCs \leftarrow \text{sum}(OCs[start : end])$ 
47:       end for
48:        $OCs \leftarrow new\_OCs$ 
49:     end if
50:   else if  $\text{shape}(OCs)[0] < C$  then # In this case we have to pad the missing components with zeros
51:     if  $OCs == []$  then
52:        $OCs \leftarrow \text{zeros}(C, S)$ 
53:     else
54:        $groups \leftarrow \text{classify\_frequencies}(modes)$  # find to which interval each mode belongs to understand which OC is missing
55:       for  $g \leftarrow iterate(C)$  do
56:         if  $g \notin groups$  then
57:            $OCs \leftarrow \text{insert}(g, \text{zeros}(S))$ 
58:         end if
59:       end for
60:     end if
61:   end if
62:    $X''[:, b, f, s] \leftarrow [X[0, b, f, s], OCs]$ 
63: end for
64: return  $X''$ 

```

9 Disentanglement metrics

Here we provide the mathematical expressions for the disentanglement evaluation metrics used.

9.1 Mutual Information

After obtaining the representation for all samples in the evaluation datasets, we calculate MI:

$$MI = \sum_{i=1}^{B_z-1} \sum_{j=i+1}^{B_z} P(i, j) \log\left(\frac{P(i, j)}{P(i)P(j)}\right) \quad (18)$$

where B_z are the number of bins used in the discretization of the latent dimensions, $P(i), P(j)$ are the proportions of samples assigned to bins i, j and $P(i, j)$ the proportion of samples that have been assigned both to i and j [16]. The discretization is done based on a histogram with 30 bins. MI is lower bounded by 0 and representation dimensions z_j have less mutual information as their MI approaches 0.

9.2 Gaussian Correlation Norm

After obtaining the representation Z for all samples in the dataset, we compute the mean vector μ_Z and covariance matrix Σ_Z and then compute the total correlation of a Gaussian with mean μ_Z and covariance matrix Σ_Z [17]:

$$GCN = \frac{D_{KL}(\mathcal{N}(\mu_Z, \Sigma_Z) \parallel \prod_j^{|\mathcal{Z}|} \mathcal{N}(\mu_{z_j}, \Sigma_{z_{jj}})}{|\mathcal{Z}| * |\log(\frac{1}{N} \sum_j^{|\mathcal{Z}|} \Sigma_{jj} + \epsilon)|} \quad (19)$$

where j denotes the latent dimension and $|\mathcal{Z}|$ denotes the number of latent dimensions. GCN is lower bounded by 0 and a representation Z is less correlated as its GCN approaches 0.

9.3 Disentanglement, Completeness, Informativeness

The DCI framework [18] trains predictors on the latent representation to predict the generative factors (or labels) from the latent code Z . We train a gradient boosting tree classifier on the latent codes, thus examining for non-linear factor-code relations. For disentanglement and completeness, importance weights R_{ij} are calculated by inspecting the Gini feature importance of each code dimension [18]. Completeness for generative factor u_i is given by $C_i = 1 + \sum_{j=1}^d p_{ij} \log_d p_{ij}$, where p_{ij} is the probability that code dimension z_j is important to predict factor u_i . p_{ij} is calculated by normalizing the importance weight R_{ij} over all importance weights for the factor u_i , $p_{ij} = \frac{R_{ij}}{\sum_{k=1}^d R_{ik}}$.

To obtain a single completeness score C , scores C_i are averaged over all factors.

Disentanglement or modularity is calculated in a similar manner for every dimension j : $D_j = 1 + \sum_{i=1}^M p_{ij} \log_M p_{ij}$. This time p_{ij} refers to the probability that dimension z_j is important to predict only factor u_i . Normalization of the factors also happens with respect to the dimension: $p_{ij} = \frac{R_{ij}}{\sum_{k=1}^M R_{kj}}$. The total disentanglement for the whole representation is a weighted average

$\sum_{j=1}^d \rho_j D_j$, where $\rho_j = \frac{\sum_{i=1}^M R_{ij}}{\sum_{k=1}^d \sum_{i=1}^M R_{ik}}$ is an importance factor for dimension j to account for less important dimensions in prediction. Finally, informativeness is calculated as the prediction error of the classifier on an unseen test set. All three metrics D, C, I are evaluated in the $[0, 1]$ interval.

9.4 Modularity

The formula to calculate modularity [12] is expressed as: $\mathcal{M}_j = 1 - \frac{\sum_{i \in \mathcal{V}_{\neq*}} I(i, z_j)^2}{I(u_*, z_j)^2 (M-1)}$. u_* is the factor that shares the maximum mutual information for each code dimension z_j , $\mathcal{V}_{\neq*}$ is the set of all factors except u_* and $M > 1$ is the number of factors. Modularity takes values in the range $[0, 1]$.

10 Supplementary experimental results

10.1 SimVowels analysis

Supplementary Table 6 presents a comprehensive sweep over a wide array of different inputs, architectures and other important DecVAE and supporting methods parameters in terms of disentanglement performance on the SimVowels dataset. These results exemplify the use of mel filterbanks as a compact speech representation that promotes disentanglement and informativeness of representations, compared to raw waveform data (Supplementary Table 6, Category - Raw inputs). Using different eigenanalysis methods such as ICA [9], PCA [7] and kernel-PCA methods [8] on raw waveforms and mel filterbank features reveals that there are no consistent performance gains by the decomposition of information before an eigenanalysis algorithm (Supplementary Table 6, Categories - Eigenanalysis of Waveform Input/Decomposed Waveform Input/Mel Fbank Input/Decomposed Mel Fbank Input). We also ablate the β value, input type and architecture of the β -VAE models, experimenting with convolutional and fully-connected architectures, as well as raw waveform and mel filterbank features (Supplementary Table 6, Categories VAEs - Untrained/Waveform input/Mel fbank input). The results indicate the use of mel filterbanks and the fully-connected VAE as superior to the alternatives. For DecVAEs, we ablate the β value, decomposition model, input type, and number of variables (frame, sequence or both) (Supplementary Table 6, Categories VAEs - Untrained/Frame DecVAEs waveform input/Frame DecVAEs mel fbank input/Multivariate DecVAEs waveform input/Multivariate DecVAEs mel fbank input). We find out that the optimal value of $\beta = 0.1$ works best when the input is mel filterbank features, in contrast to raw waveform features that require a higher prior approximation pressure of $\beta = 5$. We establish the utility of training simultaneously for two time scales (frame and sequence) as the multivariate DecVAE model achieves the best disentanglement scores of all comparisons, showcasing increased disentanglement performance compared to the model that uses only the frame variable (fDecVAE).

Supplementary Tables 7,8 present a post-training supervised evaluation of the SimVowels dataset. The evaluation is performed by fitting classifiers on the latent representations obtained by DecVAE and supporting models. We experiment with Logistic Regression (linear) and RandomForest (non-linear) classifiers. We report the best performing of the two, which in the majority of the cases is RandomForest. We ablate the same parameters as in Supplementary Table 6, this time for the tasks of vowel recognition and speaker identification. We follow an unsupervised and a supervised path to examine the predictive capability of the representations. For the unsupervised case, we fit a KMeans classifier on the representations in the development and test set of SimVowels, and measure accuracy by assigning ground truth labels to clusters through the unsupervised optimal transport optimization algorithm of Munkres [19]. We observe the best unsupervised performance comes from autoencoder models ($\beta = 0$), indicating that the organization of the representation into linearly separable clusters does not guarantee enhanced informativeness or disentanglement as previously evidenced in Supplementary Table 6. We also notice that untrained DecVAE models can have a quite satisfactory supervised performance both in the vowel and speaker recognition tasks; this indicates the importance of the decomposition model in providing an initial disentanglement of the data. The same applies to VAE mel filterbank models for the vowel recognition task. DecVAE

models score the highest vowel recognition performance in the supervised case albeit with a small margin to other methods VAEs and ICA,PCA, yet they demonstrate a significant improvement in the speaker recognition task with an increase of more than 15% compared to the best performing supporting method (ICA). It is interesting to note that both DecVAE branches (frame and sequence) have the ability to disentangle the speaker identity, a task that is considered difficult in the short time-scale of 25ms that is the frame length. This exemplifies the ability of DecVAEs to detect subtle frequency alterations in the spectral contours of uttered vowels that are attributed to speaker identity. Moreover, the highest overall performance in vowel recognition is scored by the raw mel filterbank features; this is expected as they constitute a very close approximation to the true prior.

Our experiment here aims to find a model that after all the processing can “mirror” this prior from the input space to the latent space. We can say that DecVAEs do this most successfully by also significantly improving the recovery of the speaker generative factor in addition to the vowel factor. Finally, it can be seen that the sequence branch has very good performance even without training the network (Supplementary Table 7 DecVAEs - Untrained). On the contrary, the frame representation needs refinement through training in order to reach its peak performance.

Supplementary Table 6: SimVowels disentanglement evaluation on Z and S latent spaces (frame and sequence variables). All models (DecVAEs and VAEs) train for 150 epochs with early stopping (occurs around 110 epochs) and the last checkpoint for which the validation loss minimizes is selected. Single branch DecVAE models train for 100 epochs. Results reported for DecVAEs come from the aggregated joint representation ($X + \text{OCs}$). AE and DecAE imply $\beta = 0$. VAE and DecVAE imply $\beta = 1$. OCs: oscillatory components, Conv: convolutional, FC: fully-connected, fDecVAE: frame DecVAE, MDecVAE: multivariate DecVAE.

Category	Method	Info-based		Gaussianity		Predictor-based				Intervention-based	
		Mutual Inform. ↓	Total Corr. Norm. ↓	Disent. ↑	Compleat. ↑	Inform. ↑	Explic. ↑	Modul. ↑	Interv. Robustness Score ↑		
Raw Inputs	Waveform	0.004	31.940	0.251	0.060	0.381	0.003	0.700	0.303		
	Mel Filterbank Features	0.019	1.082	0.444	0.204	0.908	0.844	0.849	0.527		
	Mel Filterbank Decomposed	0.100	6.415	0.538	0.338	0.840	0.950	0.819	0.382		
	Mel Filterbank Decomposed + Original	0.062	4.924	0.582	0.314	0.875	0.963	0.826	0.422		
Eigenanalysis - Waveform Input	PCA	0.014	< 0.001	0.010	0.065	0.525	0.009	0.817	0.611		
	k-PCA radial	0.333	0.101	0.328	0.206	0.169	0.037	0.671	0.999		
	k-PCA sigmoid	0.007	0.001	0.064	0.147	0.529	0.168	0.923	0.503		
	k-PCA poly	0.031	0.010	0.147	0.190	0.382	0.175	0.790	0.794		
	ICA	0.023	0.052	0.127	0.066	0.578	0.009	0.968	0.703		
Eigenanalysis of Decomposed Waveform Input	PCA - OCs	0.031	0.006	0.142	0.172	0.494	0.015	0.776	0.654		
	PCA - X + OCs	0.012	0.002	0.252	0.124	0.568	0.011	0.729	0.631		
	k-PCA radial - OCs	0.095	0.023	0.088	0.055	0.193	0.233	0.931	0.597		
	k-PCA radial - X + OCs	0.3889	0.174	< 0.001	1.0	0.169	0.002	0.741	0.999		
	k-PCA sigmoid - OCs	0.017	0.003	0.157	0.128	0.516	0.267	0.805	0.525		
	k-PCA sigmoid - X + OCs	0.006	0.002	0.131	0.184	0.590	0.259	0.784	0.527		
	k-PCA poly - OCs	0.041	0.007	0.259	0.297	0.367	0.334	0.875	0.836		
	k-PCA poly - X + OCs	0.035	0.007	0.188	0.303	0.369	0.062	0.857	0.829		
	ICA - OCs	0.036	0.118	0.254	0.104	0.564	0.016	0.880	0.777		
	ICA - X + OCs	0.022	0.017	0.263	0.080	0.665	0.010	0.884	0.748		
Eigenanalysis - Mel Fbank Input	PCA	0.004	0.035	0.230	0.239	0.801	0.894	0.645	0.477		
	k-PCA radial	0.043	0.008	0.288	0.114	0.791	0.935	0.561	0.580		
	k-PCA sigmoid	0.061	0.004	0.098	0.173	0.688	0.801	0.698	0.710		
	k-PCA poly	0.014	0.010	0.164	0.120	0.807	0.922	0.519	0.586		
Eigenanalysis of Decomposed Mel Fbank Input	ICA	0.005	4.85	0.086	0.156	0.812	0.894	0.605	0.538		
	PCA - OCs	0.011	0.047	0.357	0.204	0.817	0.940	0.641	0.520		
	PCA - X + OCs	0.005	0.031	0.258	0.310	0.830	0.947	0.682	0.491		
	k-PCA radial - OCs	0.131	0.046	0.355	0.161	0.646	0.851	0.641	0.787		
	k-PCA radial - X + OCs	0.097	0.035	0.253	0.145	0.610	0.822	0.495	0.916		
	k-PCA sigmoid - OCs	0.037	0.006	0.377	0.259	0.498	0.681	0.599	0.518		
	k-PCA sigmoid - X + OCs	0.051	0.006	0.136	0.156	0.526	0.713	0.555	0.570		
	k-PCA poly - OCs	0.034	0.009	0.246	0.236	0.741	0.914	0.750	0.767		
	k-PCA poly - X + OCs	0.025	0.013	0.308	0.219	0.793	0.934	0.591	0.748		
	ICA - OCs	0.019	1.180	0.176	0.189	0.834	0.940	0.589	0.660		
VAEs - Untrained	ICA - X + OCs	0.006	0.717	0.042	0.198	0.846	0.947	0.561	0.611		
	Conv-VAE Waveform	0.006	0.015	0.027	0.017	0.186	0.226	0.920	0.501		
	FC-VAE Waveform	0.006	0.015	0.055	0.022	0.248	0.159	0.890	0.507		
	FC-VAE Mel Fbank	0.022	0.039	0.128	0.077	0.565	0.906	0.731	0.514		
VAEs - Waveform Input	Conv-AE	0.070	0.486	0.081	0.212	0.690	0.664	0.963	0.526		
	FC-AE	0.027	3.171	0.183	0.214	0.433	0.615	0.966	0.520		
	Conv-VAE	0.004	0.264	0.017	0.773	0.755	0.690	0.759	0.449		
	FC-VAE	0.097	0.316	0.106	0.363	0.361	0.403	0.960	0.535		
	Conv- β -VAE ($\beta = 0.1$)	0.020	0.072	0.020	0.538	0.605	0.631	0.742	0.458		
	FC- β -VAE Time Dom. ($\beta = 0.1$)	0.011	0.256	0.039	0.615	0.425	0.488	0.834	0.482		
VAEs - Mel Fbank Input	FC-AE	0.044	1.715	0.329	0.147	0.794	0.902	0.779	0.570		
	FC-VAE	0.011	0.044	0.030	0.578	0.781	0.789	0.450	0.517		
DecVAEs - Untrained	FC- β -VAE Mel Fbank ($\beta = 0.1$)	0.007	0.061	0.062	0.615	0.805	0.772	0.563	0.552		
	fDecVAE + FD - Waveform	0.007	0.023	0.509	0.1928	0.439	0.522	0.803	0.671		
	fDecVAE + EWT - Waveform	0.004	0.021	0.339	0.152	0.373	0.456	0.896	0.655		
	fDecVAE + VMD - Waveform	0.008	0.023	0.554	0.183	0.354	0.394	0.776	0.643		
	fDecVAE + EMD - Waveform	0.008	0.024	0.177	0.209	0.362	0.433	0.830	0.636		
	fDecVAE + FD - Mel Fbank	0.006	0.032	0.565	0.250	0.816	0.949	0.834	0.563		
	fDecVAE + EWT - Mel Fbank	0.003	0.028	0.531	0.197	0.7367	0.929	0.742	0.573		
	fDecVAE + VMD - Mel Fbank	0.005	0.026	0.492	0.184	0.654	0.854	0.777	0.563		
	fDecVAE + EMD - Mel Fbank	0.002	0.023	0.359	0.191	0.597	0.841	0.794	0.622		
	MDecVAE + FD - Waveform	0.008	0.023	0.560	0.204	0.442	0.528	0.824	0.690		
Frame DecVAEs - Waveform Input	MDecVAE + EWT - Waveform	0.004	0.021	0.366	0.159	0.375	0.456	0.909	0.672		
	MDecVAE + VMD - Waveform	0.009	0.023	0.506	0.169	0.354	0.379	0.818	0.656		
	MDecVAE + EMD - Waveform	0.008	0.023	0.212	0.168	0.357	0.416	0.855	0.643		
	MDecVAE + FD - Mel Fbank	0.006	0.030	0.544	0.251	0.808	0.947	0.797	0.573		
	MDecVAE + EWT - Mel Fbank	0.003	0.026	0.515	0.220	0.731	0.930	0.756	0.589		
	MDecVAE + VMD - Mel Fbank	0.005	0.026	0.454	0.184	0.661	0.854	0.772	0.572		
	MDecVAE + EMD - Mel Fbank	0.002	0.023	0.321	0.172	0.604	0.841	0.771	0.639		
	fDecAE + FD	0.138	1.006	0.508	0.239	0.434	0.477	0.918	0.629		
	fDecAE + EWT	0.102	0.657	0.395	0.255	0.527	0.654	0.925	0.600		
	fDecAE + VMD	0.159	1.603	0.424	0.224	0.492	0.518	0.929	0.388		
Frame DecVAEs - Mel Fbank Input	fDecAE + EMD	0.089	0.475	0.531	0.264	0.451	0.637	0.736	0.523		
	fDecVAE + FD	0.008	0.198	0.221	0.29	0.586	0.781	0.923	0.676		
	fDecVAE + EWT	0.054	0.135	0.480	0.263	0.590	0.778	0.923	0.620		
	fDecVAE + VMD	0.068	0.152	0.442	0.194	0.424	0.538	0.911	0.602		
	fDecVAE + EMD	0.038	0.115	0.276	0.210	0.406	0.547	0.899	0.666		
	β -fDecVAE + FD ($\beta = 5$)	0.093	0.178	0.502	0.257	0.508	0.729	0.979	0.532		
	β -fDecVAE + EWT ($\beta = 5$)	0.071	0.135	0.399	0.246	0.435	0.611	0.885	0.619		
	β -fDecVAE + VMD ($\beta = 5$)	0.048	0.124	0.367	0.233	0.431	0.601	0.853	0.606		
	β -fDecVAE + EMD ($\beta = 5$)	0.075	0.143	0.300	0.242	0.388	0.519	0.858	0.482		
	Frame DecVAEs - Mel Fbank Input	fDecAE + FD	0.099	0.589	0.475	0.241	0.800	0.933	0.782	0.566	
fDecAE + EWT		0.101	0.623	0.531	0.267	0.764	0.911	0.855	0.583		
fDecAE + VMD		0.104	0.656	0.448	0.230	0.647	0.764	0.757	0.630		
fDecAE + EMD		0.088	0.585	0.538	0.338	0.594	0.860	0.803	0.507		
fDecVAE + FD		0.048	0.142	0.574	0.228	0.738	0.918	0.812	0.728		
fDecVAE + EWT		0.051	0.137	0.564	0.246	0.725	0.921	0.813	0.680		
fDecVAE + VMD		0.041	0.103	0.374	0.214	0.575	0.815	0.716	0.714		
fDecVAE + EMD		0.062	0.153	0.405	0.250	0.623	0.843	0.729	0.692		
β -fDecVAE + FD ($\beta = 0.1$)		0.061	0.187	0.577	0.197	0.838	0.949	0.833	0.668		
β -fDecVAE + EWT ($\beta = 0.1$)		0.063	0.189	0.563	0.244	0.762	0.923	0.869	0.578		
Multivariate DecVAEs - Waveform Input	β -fDecVAE + VMD ($\beta = 0.1$)	0.040	0.176	0.404	0.169	0.605	0.800	0.619	0.403		
	β -fDecVAE + EMD ($\beta = 0.1$)	0.057	0.199	0.581	0.306	0.633	0.854	0.851	0.615		
	MDecAE + FD	0.078	0.460	0.416	0.211	0.525	0.725	0.870	0.568		
	MDecAE + EWT	0.089	0.524	0.427	0.184	0.544	0.652	0.801	0.623		
	MDecAE + VMD	0.137	1.366	0.386	0.262	0.440	0.485	0.935	0.322		
	MDecAE + EMD	0.085	0.457	0.523	0.218	0.444	0.608	0.850	0.549		
	MDecVAE + FD	0.111	0.206	0.514	0.269	0.548	0.703	0.958	0.696		
	MDecVAE + EWT	0.099	0.198	0.478	0.257	0.561	0.772	0.906	0.707		
	MDecVAE + VMD	0.090	0.192	0.464	0.227	0.406	0.503	0.899	0.594		
	MDecVAE + EMD	0.062	0.154	0.435	0.267	0.421	0.571	0.891	0.551		
Multivariate DecVAEs - Mel Fbank Input	β -MDecVAE + FD ($\beta = 5$)	0.098	0.182	0.462	0.265	0.550	0.758	0.891	0.581		
	β -MDecVAE + EWT ($\beta = 5$)	0.033	0.118	0.438	0.205	0.416	0.629	0.789	0.564		
	β -MDecVAE + VMD ($\beta = 5$)	0.104	0.191	0.231	0.200	0.377	0.483	0.862	0.830		
	β -MDecVAE + EMD ($\beta = 5$)	0.075	0.171	0.396	0.242	0.422	0.585	0.839	0.540		
	MDecAE + FD	0.064	0.459	0.526	0.219	0.805	0.947	0.817	0.521		
	MDecAE + EWT	0.089	0.565	0.529	0.262	0.761	0.913	0.871	0.619		
	MDecAE + VMD	0.116	0.717	0.366	0.210	0.632	0.754	0.866	0.492		
	MDecAE + EMD	0.075	0.475	0.525	0.227	0.725	0.837	0.729	0.602		
	MDecVAE + FD	0.031	0.102	0.618	0.220	0.761	0.921	0.828	0.655		
	MDecVAE + EWT	0.040	0.114	0.553	0.233	0.717	0.918	0.779	0.626		
Multivariate DecVAEs - Mel Fbank Input	MDecVAE + VMD	0.031	0.122	0.541	0.227	0.767	0.947	0.645	0.425		
	MDecVAE + EMD	0.043	0.116	0.556	0.261	0.620	0.854	0.747	0.739		
	β -MDecVAE + FD ($\beta = 0.1$)	0.051	0.173	0.526	0.196	0.809	0.929	0.809	0.600		
	β -MDecVAE + EWT ($\beta = 0.1$)	0.036	0.147	0.582	0.222	0.742	0.915	0.905	0.523		
	β -MDecVAE + VMD ($\beta = 0.1$)	0.041	0.147	0.513	0.193	0.653	0.703	0.801	0.437		
	β -MDec										

Supplementary Table 7: SimVowels dataset classification evaluation - Unsupervised (Kmeans) and Supervised Classification. All models (DecVAEs and VAEs) train for 150 epochs with early stopping (occurs around 110 epochs) and the last checkpoint for which the validation loss minimizes is selected. Single branch DecVAE models train for 100 epochs. Results reported for DecVAEs come from the aggregated joint representation ($X + \text{OCs}$). AE and DecAE imply $\beta = 0$. VAE and DecVAE imply $\beta = 1$. OCs: oscillatory components, Conv: convolutional, FC: fully-connected, sDecVAE: sequence DecVAE, fDecVAE: frame DecVAE, MDecVAE: multivariate DecVAE. **bold** numbers denote best performance across trained models. underline denotes best overall performance.

Category	Method	Vowel Recognition			Speaker Identification		
		KMeans Dev. $\uparrow$	KMeans Test $\uparrow$	Test Accuracy $\uparrow$	KMeans Dev. $\uparrow$	KMeans Test $\uparrow$	Test Accuracy $\uparrow$
Dummy Classifiers	Random Classifier	NA	NA	19.94	NA	NA	10.09
	Majority Classifier	NA	NA	20.22	NA	NA	13.98
Raw Inputs	Waveform	26.16	26.66	84.15	21.79	22.68	46.78
	Mel Filterbank Features	52.47	52.25	<u>98.09</u>	30.14	30.22	76.65
	Mel Filterbank Decomposed	35.53	30.64	94.65	30.43	21.96	76.11
	Mel Filterbank Decomposed + Original	39.04	39.41	97.08	30.82	29.49	79.15
Eigenanalysis - Waveform	PCA	25.95	25.77	79.37	21.82	21.97	42.06
	k-PCA radial	20.21	20.28	25.28	22.39	22.91	14.14
	k-PCA sigmoid	26.00	25.57	83.25	23.64	22.79	40.95
	k-PCA poly	25.74	24.97	56.30	22.11	23.21	24.69
	ICA	26.40	25.57	82.39	22.02	21.94	47.07
Eigenanalysis - Decomposed Waveform	PCA - OCs	24.18	23.51	72.42	22.02	22.32	37.89
	PCA - X + OCs	22.81	22.30	86.39	22.14	22.73	49.25
	k-PCA radial - OCs	20.21	20.30	43.68	22.40	22.90	16.87
	k-PCA radial - X + OCs	20.20	20.28	20.42	22.38	22.91	13.98
	k-PCA sigmoid - OCs	26.75	25.50	73.78	24.25	22.44	38.46
	k-PCA sigmoid - X + OCs	28.15	30.16	88.52	23.16	23.30	51.49
	k-PCA poly - OCs	26.32	26.80	50.91	22.46	23.51	21.80
	k-PCA poly - X + OCs	24.41	24.29	52.41	21.96	22.74	23.49
	ICA - OCs	24.57	24.70	78.44	22.37	22.91	47.23
	ICA - X + OCs	23.18	24.69	92.35	22.26	22.39	58.82
Eigenanalysis - Mel Fbank	PCA	47.78	38.78	97.36	50.29	49.59	74.10
	k-PCA radial	42.95	44.87	96.33	35.33	32.28	69.08
	k-PCA sigmoid	34.73	36.93	91.11	30.84	30.27	57.01
	k-PCA poly	51.09	50.07	97.17	41.58	40.69	72.79
	ICA	45.47	45.60	97.36	50.31	50.81	74.13
Eigenanalysis - Decomposed Mel Fbank	PCA - OCs	36.69	33.50	94.74	67.86	55.64	75.55
	PCA - X + OCs	36.81	36.76	97.23	64.00	58.80	78.22
	k-PCA radial - OCs	25.40	27.16	79.28	25.96	26.82	59.82
	k-PCA radial - X + OCs	23.23	23.47	86.32	24.81	23.92	58.89
	k-PCA sigmoid - OCs	26.59	26.65	72.46	25.18	25.98	46.33
	k-PCA sigmoid - X + OCs	28.87	28.14	78.59	23.58	24.05	42.42
	k-PCA poly - OCs	33.91	35.54	92.58	38.33	35.45	68.02
	k-PCA poly - X + OCs	37.26	38.14	95.57	45.46	38.69	73.08
	ICA - OCs	32.65	30.89	94.67	52.81	54.30	75.47
	ICA - X + OCs	38.14	35.11	97.27	57.03	57.59	78.22
VAEs - Untrained	Conv-VAE Waveform	22.57	23.10	64.83	21.43	21.40	29.20
	FC-VAE Waveform	22.20	22.55	61.42	20.68	20.26	26.93
	FC-VAE Mel Fbank	58.55	56.31	96.03	35.15	32.89	70.63
VAEs - Waveform	Conv-AE	56.17	57.29	95.75	31.94	34.66	71.95
	FC-AE	45.34	46.31	64.98	27.32	26.74	35.27
	Conv-VAE	34.58	35.21	90.30	26.09	25.45	57.27
	FC-VAE	22.83	22.66	52.27	20.85	20.89	22.61
	Conv- β -VAE ($\beta = 0.1$)	53.09	49.57	90.84	24.25	25.72	56.17
	FC- β -VAE Time Dom. (vowel $\beta = 0.1$, speaker $\beta = 10$)	24.64	24.57	58.86	22.19	22.73	29.85
VAEs - Mel Fbank	FC-AE	52.71	59.4	97.61	33.53	29.73	74.40
	FC-VAE	34.99	39.55	93.90	35.70	30.64	73.84
	FC- β -VAE Mel Fbank (vowel $\beta = 0.1$, speaker $\beta = 5$)	36.82	36.67	96.30	26.51	26.55	73.2
DecVAEs - Untrained	fDecVAE + FD - Waveform	24.80	25.67	61.36	22.71	20.56	36.55
	fDecVAE + EWT - Waveform	23.84	25.46	58.20	22.21	20.91	27.10
	fDecVAE + VMD - Waveform	26.46	25.72	54.64	21.99	21.55	26.48
	fDecVAE + EMD - Waveform	31.19	32.19	55.74	26.68	23.41	23.32
	fDecVAE + FD - Mel Fbank	44.42	42.71	93.41	39.55	39.76	86.50
	fDecVAE + EWT - Mel Fbank	48.95	45.49	92.91	34.96	36.56	82.21
	fDecVAE + VMD - Mel Fbank	34.43	31.30	88.68	29.94	29.93	72.34
	fDecVAE + EMD - Mel Fbank	44.02	39.18	81.85	29.83	31.20	67.75
	sDecVAE + FD - Waveform	NA	NA	NA	78.98	79.27	91.25
	sDecVAE + EWT - Waveform	NA	NA	NA	49.52	49.83	57.50
	sDecVAE + VMD - Waveform	NA	NA	NA	46.24	49.97	62.50
	sDecVAE + EMD - Waveform	NA	NA	NA	76.02	71.00	65.00
	sDecVAE + FD - Mel Fbank	NA	NA	NA	77.38	82.27	94.38
	sDecVAE + EWT - Mel Fbank	NA	NA	NA	74.16	79.00	96.88
	sDecVAE + VMD - Mel Fbank	NA	NA	NA	76.94	73.93	96.25
	sDecVAE + EMD - Mel Fbank	NA	NA	NA	85.48	89.27	92.50
	MDecVAE + FD - Waveform	24.59	25.92	61.17	72.88 (s)	77.47 (s)	91.25 (s)
	MDecVAE + EWT - Waveform	24.88	25.63	57.45	46.26 (s)	47.03 (s)	53.75 (s)
	MDecVAE + VMD - Waveform	25.77	25.60	54.05	49.04 (s)	59.70 (s)	65.00 (s)
	MDecVAE + EMD - Waveform	30.68	30.83	54.50	77.44 (s)	67.70 (s)	61.25 (s)
	MDecVAE + FD - Mel Fbank	43.76	39.06	93.73	84.24 (s)	75.87 (s)	96.88 (s)
	MDecVAE + EWT - Mel Fbank	47.78	47.56	93.22	71.54 (s)	82.27 (s)	95.00 (s)
	MDecVAE + VMD - Mel Fbank	31.53	34.53	89.73	68.28 (s)	64.10 (s)	95.63 (s)
	MDecVAE + EMD - Mel Fbank	48.66	47.55	83.61	<u>96.00</u> (s)	<u>90.23</u> (s)	94.38 (s)

Continued on Supplementary Tables

Supplementary Table 8: Continued from Supplementary Table 7

Category	Method	Vowel Recognition			Speaker Identification		
		KMeans Dev. $\uparrow$	KMeans Test $\uparrow$	Test Accuracy $\uparrow$	KMeans Dev. $\uparrow$	KMeans Test $\uparrow$	Test Accuracy $\uparrow$
Frame DecVAE - Waveform	fDecAE + FD	25.27	26.85	65.52	21.88	23.22	32.89
	fDecAE + EWT	36.03	34.72	70.78	29.25	27.85	40.01
	fDecAE + VMD	34.05	32.69	70.47	25.13	26.19	30.59
	fDecAE + EMD	26.77	26.22	64.86	34.05	33.87	34.99
	fDecVAE + FD	38.97	37.38	79.66	35.39	34.88	58.24
	fDecVAE + EWT	36.68	35.44	85.39	36.64	36.93	50.57
	fDecVAE + VMD	26.05	25.87	72.26	22.24	20.77	36.72
	fDecVAE + EMD	32.67	31.99	66.75	23.73	23.63	27.86
	β -fDecVAE + FD ($\beta = 5$)	29.60	28.83	77.12	30.28	30.93	42.62
	β -fDecVAE + EWT ($\beta = 5$)	29.74	29.62	69.75	26.12	26.31	34.43
	β -fDecVAE + VMD ($\beta = 5$)	40.54	38.61	73.79	26.24	27.21	41.80
	β -fDecVAE + EMD($\beta = 5$)	32.98	33.09	67.36	25.33	25.54	30.21
Frame DecVAE - Mel Fbank	fDecAE + FD	36.84	42.16	94.28	46.98	42.16	79.49
	fDecAE + EWT	42.26	35.68	97.55	36.59	35.68	64.59
	fDecAE + VMD	31.08	25.62	86.14	25.04	25.62	62.81
	fDecAE + EMD	31.38	35.17	93.68	34.68	35.17	51.52
	fDecVAE + FD	38.79	36.56	89.91	34.54	36.56	84.80
	fDecVAE + EWT	44.25	33.75	97.90	34.53	33.75	73.09
	fDecVAE + VMD	32.01	30.45	92.32	29.29	30.45	64.88
	fDecVAE + EMD	32.42	30.53	93.60	29.72	30.53	48.83
	β -fDecVAE + FD($\beta = 0.1$)	36.42	42.50	93.46	45.46	42.50	91.84
	β -fDecVAE + EWT ($\beta = 0.1$)	44.71	35.72	97.71	35.88	35.72	71.43
	β -fDecVAE + VMD ($\beta = 0.1$)	29.34	30.48	92.34	29.39	30.48	68.53
	β -fDecVAE + EMD($\beta = 0.1$)	39.19	36.47	92.58	36.88	36.47	51.55
Sequence DecVAEs - Waveform	sDecAE + FD	NA	NA	NA	68.76	75.97	85.63
	sDecAE + EWT	NA	NA	NA	40.60	50.43	49.38
	sDecAE + VMD	NA	NA	NA	49.20	53.73	64.38
	sDecAE + EMD	NA	NA	NA	56.48	55.33	56.25
	sDecVAE + FD	NA	NA	NA	65.54	68.37	68.13
	sDecVAE + EWT	NA	NA	NA	24.70	27.83	35.63
	sDecVAE + VMD	NA	NA	NA	25.88	29.33	53.13
	sDecVAE + EMD	NA	NA	NA	40.20	34.90	59.38
	β -sDecVAE + FD ($\beta = 5$)	NA	NA	NA	60.22	64.93	80.00
	β -sDecVAE + EWT ($\beta = 5$)	NA	NA	NA	37.48	39.50	34.38
	β -sDecVAE + VMD ($\beta = 5$)	NA	NA	NA	25.90	27.10	44.38
	β -sDecVAE + EMD($\beta = 5$)	NA	NA	NA	42.76	40.00	54.38
Sequence DecVAEs - Mel Fbank	sDecAE + FD	NA	NA	NA	56.90	52.7	86.88
	sDecAE + EWT	NA	NA	NA	55.92	49.03	83.13
	sDecAE + VMD	NA	NA	NA	52.62	55.37	89.38
	sDecAE + EMD	NA	NA	NA	78.58	73.97	91.25
	sDecVAE + FD	NA	NA	NA	70.66	60.27	81.25
	sDecVAE + EWT	NA	NA	NA	51.02	58.07	60.00
	sDecVAE + VMD	NA	NA	NA	39.18	51.37	71.88
	sDecVAE + EMD	NA	NA	NA	42.42	35.40	69.38
	β -sDecVAE + FD ($\beta = 0.1$)	NA	NA	NA	71.88	70.23	76.25
	β -sDecVAE + EWT ($\beta = 0.1$)	NA	NA	NA	55.46	46.50	76.88
	β -sDecVAE + VMD ($\beta = 0.1$)	NA	NA	NA	57.56	46.53	76.88
	β -sDecVAE + EMD($\beta = 0.1$)	NA	NA	NA	58.32	53.00	65.00
Multivariate DecVAE - Waveform	MDecAE + FD	33.04	31.25	69.78	65.56 (s)	66.37 (s)	87.50 (s)
	MDecAE + EWT	31.61	33.50	70.83	46.56 (s)	44.40 (s)	48.13 (s)
	MDecAE + VMD	32.97	32.80	65.68	55.36 (s)	57.73 (s)	68.75 (s)
	MDecAE + EMD	32.04	31.17	61.26	55.02 (s)	61.20 (s)	60.00 (s)
	MDecVAE + FD	30.32	30.17	78.48	52.24 (s)	61.60 (s)	71.25 (s)
	MDecVAE + EWT	25.24	25.76	87.70	29.12 (s)	29.79 (f)	50.15 (f)
	MDecVAE + VMD	28.19	27.34	68.96	29.10 (s)	31.47 (s)	41.88 (s)
	MDecVAE + EMD	36.93	35.67	64.52	49.38 (s)	40.43 (s)	57.50 (s)
	β -MDecVAE + FD ($\beta = 5$)	37.65	37.61	81.55	55.28 (s)	60.90 (s)	78.75 (s)
	β -MDecVAE + EWT ($\beta = 5$)	37.33	36.54	73.59	29.72 (f)	31.11 (f)	39.51 (f)
	β -MDecVAE + VMD ($\beta = 5$)	27.60	25.26	60.15	30.34 (s)	32.10 (s)	47.50 (s)
	β -MDecVAE + EMD($\beta = 5$)	32.62	32.76	66.51	45.74 (s)	53.67 (s)	55.63 (s)
Multivariate DecVAE - Mel Fbank	MDecAE + FD	35.78	35.47	94.37	65.06 (s)	65.03 (s)	93.75 (s)
	MDecAE + EWT	39.52	38.01	97.36	53.90 (s)	55.97 (s)	95.63 (s)
	MDecAE + VMD	32.92	32.48	86.88	60.84 (s)	61.80 (s)	83.75 (s)
	MDecAE + EMD	30.58	30.85	87.45	64.24 (s)	75.70 (s)	91.88 (s)
	MDecVAE + FD	29.10	30.08	91.61	59.32 (s)	56.80 (s)	86.14 (f)
	MDecVAE + EWT	40.48	41.98	97.40	39.24 (s)	36.29 (f)	73.47 (f)
	MDecVAE + VMD	27.85	27.43	90.55	61.32 (s)	62.90 (s)	71.25 (s)
	MDecVAE + EMD	33.41	32.13	92.76	76.00 (s)	62.57 (s)	68.75 (s)
	β -MDecVAE + FD($\beta = 0.1$)	34.50	33.55	92.43	73.32 (s)	70.37 (s)	87.84 (f)
	β -MDecVAE + EWT ($\beta = 0.1$)	37.69	37.32	97.66	64.74 (s)	62.87 (s)	75.85 (f)
	β -MDecVAE + VMD ($\beta = 0.1$)	27.34	30.71	92.8	45.16 (s)	57.87 (s)	72.50 (s)
	β -MDecVAE + EMD($\beta = 0.1$)	37.01	36.23	88.79	67.62 (s)	66.20 (s)	65.63 (s)

10.2 Real speech datasets analysis

Supplementary Tables 9,10,12 present the disentanglement results for the datasets TIMIT[20], VOC-ALS[21], IEMOCAP[22], respectively. Supplementary Tables 11,13 present the supervised classification tasks of zero-shot dysarthria severity evaluation and emotion recognition in the datasets VOC-ALS[21], IEMOCAP[22], respectively.

Supplementary Table 9: TIMIT [20] Disentanglement evaluation on Z and S latent spaces (frame and sequence variables). All models (DecVAEs and VAEs) train for 100 epochs. Results reported for DecVAEs come from the aggregated joint representation (X + OCs). MI: Mutual Information, TGCN: Total Correlation Normalized, D: Disentanglement, C: Completeness, I: Informativeness, EX: Explicitness, M: Modularity, IRS: Interventional Robustness Score. 95% confidence intervals are reported for metrics D, C, I, EX, M. Values smaller than 0.001 are displayed as 0.001.

Category	Method	Info-based	Gaussianity	Predictor-based					Intervention-based
		MI ↓	TGCN ↓	D ↑	C ↑	I ↑	EX ↑	M ↑	IRS ↑
Raw Inputs	Mel Filterbank Features	0.1905	4.021	0.109 ± 0.001	0.036 ± 0.001	0.265 ± 0.001	0.679 ± 0.001	0.980 ± 0.001	0.557
Eigenanalysis - Mel Fbank Input	PCA	0.002	< 0.001	0.229 ± 0.001	0.156 ± 0.001	0.258 ± 0.001	0.648 ± 0.001	0.729 ± 0.001	0.551
	ICA	0.003	0.212	0.139 ± 0.001	0.085 ± 0.001	0.262 ± 0.001	0.648 ± 0.001	0.714 ± 0.001	0.554
VAEs - Mel Fbank Input	FC-AE	0.093	2.378	0.153 ± 0.001	0.063 ± 0.001	0.191 ± 0.001	0.579 ± 0.001	0.957 ± 0.001	0.544
	FC-VAE	0.041	0.084	0.191 ± 0.001	0.117 ± 0.001	0.188 ± 0.001	0.492 ± 0.001	0.980 ± 0.001	0.683
	FC- β -VAE Mel Fbank ($\beta = 0.1$)	0.022	0.074	0.202 ± 0.001	0.164 ± 0.001	0.184 ± 0.001	0.486 ± 0.001	0.933 ± 0.001	0.669
Multivariate DecVAEs - Mel Fbank Input - Z branch (Frame)	MDecAE + FD	0.036	0.217	0.230 ± 0.001	0.071 ± 0.001	0.408 ± 0.001	0.472 ± 0.001	0.657 ± 0.001	0.625
	MDecAE + EWT	0.034	0.236	0.254 ± 0.001	0.086 ± 0.001	0.383 ± 0.001	0.475 ± 0.001	0.650 ± 0.001	0.675
	MDecVAE + FD	0.061	0.184	0.139 ± 0.001	0.065 ± 0.001	0.491 ± 0.001	0.667 ± 0.001	0.569 ± 0.001	0.772
	MDecVAE + EWT	0.065	0.185	0.233 ± 0.001	0.072 ± 0.001	0.472 ± 0.001	0.673 ± 0.001	0.541 ± 0.001	0.706
	β -MDecVAE + FD ($\beta = 0.1$)	0.052	0.148	0.172 ± 0.001	0.067 ± 0.001	0.394 ± 0.001	0.467 ± 0.001	0.739 ± 0.001	0.872
	β -MDecVAE + EWT ($\beta = 0.1$)	0.051	0.155	0.256 ± 0.001	0.087 ± 0.001	0.376 ± 0.001	0.504 ± 0.001	0.575 ± 0.001	0.875

Supplementary Table 10: VOC-ALS [21] disentanglement zero-shot evaluation on Z and S latent spaces (frame and sequence variables). All models are transferred from SimVowels or TIMIT [20] and zero-shot evaluated on VOC-ALS without any fine-tuning. Results reported for DecVAEs come from the aggregated joint representation (X + OCs). MI: Mutual Information, TGCN: Total Correlation Normalized, D: Disentanglement, C: Completeness, I: Informativeness, EX: Explicitness, M: Modularity, IRS: Interventional Robustness Score. 95% confidence intervals are reported for metrics D, C, I, EX, M. Values smaller than 0.001 are displayed as 0.001.

Category	Method	Info-based	Gaussianity	Predictor-based					Intervention-based
		MI ↓	TGCN ↓	D ↑	C ↑	I ↑	EX ↑	M ↑	IRS ↑
Raw Inputs	Mel Filterbank Features	0.179	2.884	0.123 ± 0.001	0.083 ± 0.001	0.506 ± 0.001	0.624 ± 0.001	0.845 ± 0.001	0.652
Eigenanalysis - Mel Fbank Input	PCA	0.004	<< 0.001	0.130 ± 0.001	0.154 ± 0.001	0.434 ± 0.001	0.590 ± 0.001	0.863 ± 0.001	0.620
	ICA	0.004	<< 0.001	0.082 ± 0.001	0.069 ± 0.001	0.439 ± 0.001	0.590 ± 0.001	0.842 ± 0.001	0.611
Transfer from Sim Vowels - VAEs - Mel Fbank	FC-AE	0.091	2.985	0.107 ± 0.001	0.070 ± 0.001	0.374 ± 0.001	0.570 ± 0.001	0.866 ± 0.001	0.553
	FC-VAE	0.036	0.134	0.075 ± 0.001	0.069 ± 0.001	0.350 ± 0.001	0.516 ± 0.001	0.853 ± 0.001	0.468
	FC- β -VAE Mel Fbank ($\beta = 0.1$)	0.043	0.219	0.074 ± 0.001	0.065 ± 0.001	0.365 ± 0.001	0.543 ± 0.001	0.775 ± 0.001	0.481
Transfer from TIMIT - VAEs - Mel Fbank	FC-AE	0.092	3.014	0.103 ± 0.001	0.068 ± 0.001	0.369 ± 0.001	0.574 ± 0.001	0.792 ± 0.001	0.743
	FC-VAE	0.122	0.341	0.077 ± 0.001	0.047 ± 0.001	0.340 ± 0.001	0.546 ± 0.001	0.888 ± 0.001	0.500
	FC- β -VAE Mel Fbank ($\beta = 0.1$)	0.067	0.332	0.086 ± 0.001	0.067 ± 0.001	0.350 ± 0.001	0.555 ± 0.001	0.863 ± 0.001	0.489
Transfer from SimVowels - Multivariate DecVAE - Mel Fbank - Z branch (Frame)	β -MDecVAE + FD($\beta = 0.1$) - NoC = 3	0.044	0.155	0.097 ± 0.001	0.085 ± 0.001	0.653 ± 0.001	0.626 ± 0.001	0.750 ± 0.001	0.729
	β -MDecVAE + EWT ($\beta = 0.1$) - NoC = 3	0.036	0.150	0.099 ± 0.001	0.081 ± 0.001	0.659 ± 0.001	0.662 ± 0.001	0.750 ± 0.001	0.650
	β -MDecVAE + FD($\beta = 0.1$) - NoC = 4	0.039	0.144	0.115 ± 0.001	0.084 ± 0.001	0.679 ± 0.001	0.703 ± 0.001	0.696 ± 0.001	0.834
	β -MDecVAE + EWT ($\beta = 0.1$) - NoC = 4	0.060	0.140	0.125 ± 0.001	0.090 ± 0.001	0.675 ± 0.001	0.696 ± 0.001	0.732 ± 0.001	0.654
Transfer from TIMIT - Multivariate DecVAE - Mel Fbank - Z branch (Frame)	β -MDecVAE + FD($\beta = 0.1$) - NoC = 4	0.039	0.204	0.172 ± 0.001	0.101 ± 0.001	0.612 ± 0.001	0.670 ± 0.001	0.767 ± 0.001	0.624
	β -MDecVAE + EWT ($\beta = 0.1$) - NoC = 4	0.046	0.229	0.178 ± 0.001	0.093 ± 0.001	0.571 ± 0.001	0.654 ± 0.001	0.776 ± 0.001	0.611
Transfer from SimVowels - Multivariate DecVAE - Mel Fbank - S branch (Sequence)	β -MDecVAE + FD($\beta = 0.1$) - NoC = 3	0.0634	0.273	0.116 ± 0.001	0.053 ± 0.001	0.308 ± 0.001	0.484 ± 0.001	0.846 ± 0.001	0.560
	β -MDecVAE + EWT ($\beta = 0.1$) - NoC = 3	0.060	0.292	0.096 ± 0.001	0.048 ± 0.001	0.260 ± 0.001	0.503 ± 0.001	0.816 ± 0.001	0.644
	β -MDecVAE + FD($\beta = 0.1$) - NoC = 4	0.0652	0.231	0.106 ± 0.001	0.043 ± 0.001	0.297 ± 0.001	0.481 ± 0.001	0.881 ± 0.001	0.775
	β -MDecVAE + EWT ($\beta = 0.1$) - NoC = 4	0.034	0.242	0.126 ± 0.001	0.055 ± 0.001	0.254 ± 0.001	0.493 ± 0.001	0.811 ± 0.001	0.571
Transfer from TIMIT - Multivariate DecVAE - Mel Fbank - S branch (Sequence)	β -MDecVAE + FD($\beta = 0.1$) - NoC = 4	0.060	0.272	0.114 ± 0.001	0.044 ± 0.001	0.295 ± 0.001	0.514 ± 0.001	0.937 ± 0.001	0.564
	β -MDecVAE + EWT ($\beta = 0.1$) - NoC = 4	0.0289	0.214	0.071 ± 0.001	0.025 ± 0.001	0.227 ± 0.001	0.324 ± 0.001	0.784 ± 0.001	0.645

Supplementary Table 11: VOC-ALS [21] classification evaluation on zero-shot transferred representations. Leave-one-group-out cross-validation supervised classification with five random states. All models are transferred from SimVowels or TIMIT [20] and zero-shot evaluated on VOC-ALS without any fine-tuning - Results reported for DecVAEs come from the aggregated joint representation (X + OCs). The average value across random states and folds is presented accompanied by 95% confidence intervals ($\pm$). KCS D: King’s Clinical Stage Detection, DDD: Disease Duration Detection, PR: Phoneme Recognition, SI: Speaker Identification, WA: Weighted Accuracy, WF1: Weighted F1-score.

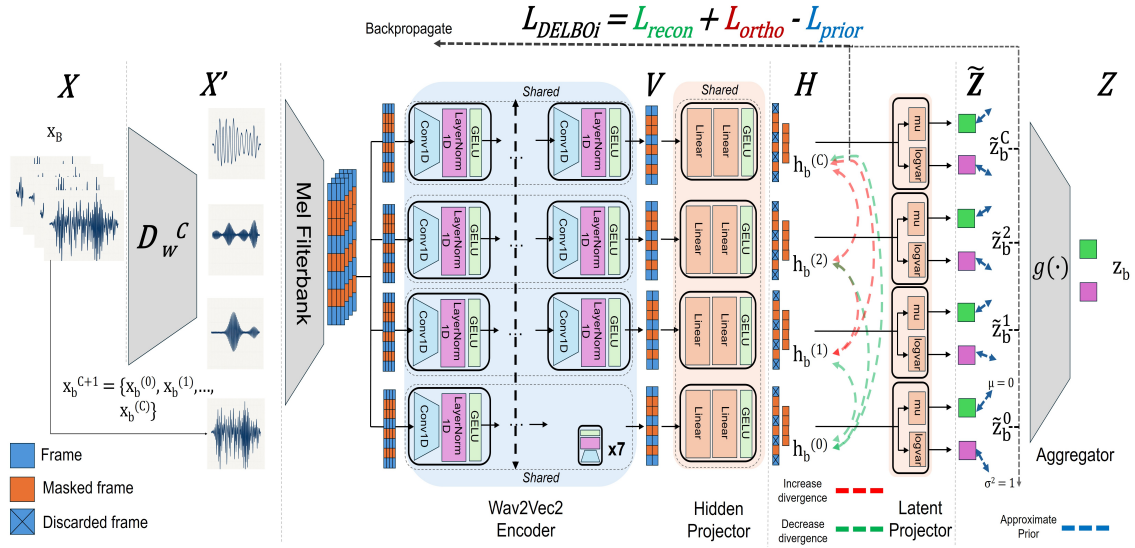
Category	Method	KCS D			DDD			PR			SI		
		WA $\uparrow$	WF1 $\uparrow$	F1-macro $\uparrow$	WA $\uparrow$	WF1 $\uparrow$	F1-macro $\uparrow$	WA $\uparrow$	WF1 $\uparrow$	F1-macro $\uparrow$	WA $\uparrow$	WF1 $\uparrow$	F1-macro $\uparrow$
Dummy Classifiers	Random Classifier	0.167 $\pm$ 0.001	0.212 $\pm$ 0.001	0.189 $\pm$ 0.001	0.200 $\pm$ 0.001	0.212 $\pm$ 0.001	0.189 $\pm$ 0.001	0.125 $\pm$	0.125 $\pm$ 0.001	0.125 $\pm$ 0.001	0.007 $\pm$ 0.001	0.007 $\pm$ 0.001	0.006 $\pm$ 0.001
	Majority Classifier	0.341 $\pm$ 0.001	0.174 $\pm$ 0.001	0.102 $\pm$ 0.001	0.342 $\pm$ 0.001	0.174 $\pm$ 0.001	0.102 $\pm$ 0.001	0.128 $\pm$ 0.001	0.029 $\pm$ 0.001	0.028 $\pm$ 0.001	0.007 $\pm$ 0.001	0.001 $\pm$ 0.001	0.001 $\pm$ 0.001
Raw Inputs		Mel Filterbank Features											
Eigenanalysis - Mel Fbank	PCA	0.709 $\pm$ 0.001	0.094 $\pm$ 0.001	0.020 $\pm$ 0.000	0.706 $\pm$ 0.000	0.097 $\pm$ 0.001	0.072 $\pm$ 0.001	0.777 $\pm$ 0.000	0.777 $\pm$ 0.001	0.779 $\pm$ 0.001	0.705 $\pm$ 0.001	0.704 $\pm$ 0.001	0.699 $\pm$ 0.001
	ICA	0.584 $\pm$ 0.001	0.564 $\pm$ 0.001	0.462 $\pm$ 0.002	0.587 $\pm$ 0.001	0.562 $\pm$ 0.001	0.514 $\pm$ 0.001	0.711 $\pm$ 0.001	0.710 $\pm$ 0.001	0.713 $\pm$ 0.001	0.550 $\pm$ 0.001	0.548 $\pm$ 0.001	0.542 $\pm$ 0.001
	PCA	0.575 $\pm$ 0.001	0.554 $\pm$ 0.001	0.453 $\pm$ 0.002	0.577 $\pm$ 0.001	0.551 $\pm$ 0.001	0.501 $\pm$ 0.001	0.700 $\pm$ 0.001	0.700 $\pm$ 0.001	0.703 $\pm$ 0.001	0.523 $\pm$ 0.001	0.520 $\pm$ 0.001	0.515 $\pm$ 0.001
Transfer from SimVowels - VAEs - Mel Fbank	FC-AE	0.481 $\pm$ 0.001	0.450 $\pm$ 0.001	0.360 $\pm$ 0.003	0.489 $\pm$ 0.001	0.454 $\pm$ 0.001	0.391 $\pm$ 0.001	0.648 $\pm$ 0.001	0.647 $\pm$ 0.001	0.649 $\pm$ 0.001	0.322 $\pm$ 0.001	0.318 $\pm$ 0.001	0.315 $\pm$ 0.001
	FC-VAE	0.403 $\pm$ 0.001	0.424 $\pm$ 0.001	0.326 $\pm$ 0.002	0.474 $\pm$ 0.001	0.432 $\pm$ 0.001	0.364 $\pm$ 0.001	0.632 $\pm$ 0.001	0.630 $\pm$ 0.001	0.632 $\pm$ 0.001	0.300 $\pm$ 0.001	0.297 $\pm$ 0.001	0.295 $\pm$ 0.001
	FC- β -VAE Mel Fbank ($\beta = 0.1$)	0.476 $\pm$ 0.001	0.437 $\pm$ 0.001	0.342 $\pm$ 0.002	0.485 $\pm$ 0.001	0.442 $\pm$ 0.001	0.372 $\pm$ 0.001	0.648 $\pm$ 0.001	0.645 $\pm$ 0.001	0.647 $\pm$ 0.001	0.341 $\pm$ 0.001	0.338 $\pm$ 0.001	0.336 $\pm$ 0.001
Transfer from TIMIT - VAEs - Mel Fbank	FC-AE	0.475 $\pm$ 0.001	0.440 $\pm$ 0.001	0.331 $\pm$ 0.001	0.483 $\pm$ 0.001	0.443 $\pm$ 0.001	0.376 $\pm$ 0.001	0.648 $\pm$ 0.001	0.647 $\pm$ 0.001	0.650 $\pm$ 0.001	0.332 $\pm$ 0.001	0.329 $\pm$ 0.001	0.324 $\pm$ 0.001
	FC-VAE	0.415 $\pm$ 0.001	0.385 $\pm$ 0.001	0.285 $\pm$ 0.002	0.425 $\pm$ 0.001	0.387 $\pm$ 0.001	0.318 $\pm$ 0.001	0.622 $\pm$ 0.001	0.622 $\pm$ 0.001	0.624 $\pm$ 0.001	0.291 $\pm$ 0.001	0.287 $\pm$ 0.001	0.289 $\pm$ 0.001
	FC- β -VAE Mel Fbank ($\beta = 0.1$)	0.439 $\pm$ 0.001	0.403 $\pm$ 0.001	0.300 $\pm$ 0.001	0.447 $\pm$ 0.001	0.407 $\pm$ 0.001	0.337 $\pm$ 0.001	0.635 $\pm$ 0.001	0.636 $\pm$ 0.001	0.639 $\pm$ 0.001	0.324 $\pm$ 0.001	0.320 $\pm$ 0.001	0.320 $\pm$ 0.001
Transfer from SimVowels - Multivariate DecVAE - Mel Fbank - Z branch (Frame)	β -MDecVAE + FD ($\beta = 0.1$), $Nac = 3$	0.853 $\pm$ 0.001	0.855 $\pm$ 0.001	0.816 $\pm$ 0.002	0.850 $\pm$ 0.001	0.848 $\pm$ 0.001	0.837 $\pm$ 0.001	0.888 $\pm$ 0.001	0.888 $\pm$ 0.001	0.888 $\pm$ 0.001	0.918 $\pm$ 0.001	0.918 $\pm$ 0.001	0.916 $\pm$ 0.001
	β -MDecVAE + EWT ($\beta = 0.1$), $Nac = 3$	0.879 $\pm$ 0.001	0.880 $\pm$ 0.001	0.852 $\pm$ 0.001	0.882 $\pm$ 0.001	0.881 $\pm$ 0.001	0.878 $\pm$ 0.001	0.918 $\pm$ 0.001	0.918 $\pm$ 0.001	0.918 $\pm$ 0.001	0.938 $\pm$ 0.001	0.938 $\pm$ 0.001	0.937 $\pm$ 0.001
	β -MDecVAE + FD ($\beta = 0.1$), $Nac = 4$	0.864 $\pm$ 0.001	0.866 $\pm$ 0.001	0.846 $\pm$ 0.001	0.864 $\pm$ 0.001	0.863 $\pm$ 0.001	0.853 $\pm$ 0.001	0.913 $\pm$ 0.001	0.913 $\pm$ 0.001	0.913 $\pm$ 0.001	0.958 $\pm$ 0.001	0.958 $\pm$ 0.001	0.952 $\pm$ 0.001
	β -MDecVAE + EWT ($\beta = 0.1$), $Nac = 4$	0.829 $\pm$ 0.001	0.831 $\pm$ 0.001	0.819 $\pm$ 0.001	0.834 $\pm$ 0.001	0.832 $\pm$ 0.001	0.818 $\pm$ 0.001	0.904 $\pm$ 0.001	0.904 $\pm$ 0.001	0.904 $\pm$ 0.001	0.936 $\pm$ 0.001	0.936 $\pm$ 0.001	0.935 $\pm$ 0.001
Transfer from TIMIT - Multivariate DecVAE - Mel Fbank - Z branch (Frame)	β -MDecVAE + FD ($\beta = 0.1$), $Nac = 3$	0.710 $\pm$ 0.001	0.733 $\pm$ 0.001	0.738 $\pm$ 0.002	0.762 $\pm$ 0.001	0.760 $\pm$ 0.001	0.749 $\pm$ 0.001	0.791 $\pm$ 0.001	0.794 $\pm$ 0.001	0.794 $\pm$ 0.001	0.787 $\pm$ 0.001	0.790 $\pm$ 0.001	0.791 $\pm$ 0.001
	β -MDecVAE + EWT ($\beta = 0.1$), $Nac = 4$	0.700 $\pm$ 0.001	0.698 $\pm$ 0.001	0.683 $\pm$ 0.001	0.711 $\pm$ 0.001	0.704 $\pm$ 0.001	0.685 $\pm$ 0.001	0.778 $\pm$ 0.001	0.778 $\pm$ 0.001	0.780 $\pm$ 0.001	0.712 $\pm$ 0.001	0.715 $\pm$ 0.001	0.714 $\pm$ 0.001
	β -MDecVAE + FD ($\beta = 0.1$), $Nac = 3$	0.367 $\pm$ 0.007	0.394 $\pm$ 0.007	0.262 $\pm$ 0.006	0.352 $\pm$ 0.007	0.286 $\pm$ 0.007	0.202 $\pm$ 0.007	0.525 $\pm$ 0.008	0.523 $\pm$ 0.008	0.523 $\pm$ 0.008	NA	NA	NA
	β -MDecVAE + EWT ($\beta = 0.1$), $Nac = 3$	0.363 $\pm$ 0.006	0.316 $\pm$ 0.006	0.222 $\pm$ 0.011	0.346 $\pm$ 0.007	0.287 $\pm$ 0.006	0.204 $\pm$ 0.006	0.426 $\pm$ 0.009	0.418 $\pm$ 0.010	0.417 $\pm$ 0.010	NA	NA	NA
Transfer from SimVowels - Multivariate DecVAE - Mel Fbank - S branch (Sequence)	β -MDecVAE + FD ($\beta = 0.1$), $Nac = 3$	0.380 $\pm$ 0.007	0.323 $\pm$ 0.008	0.221 $\pm$ 0.009	0.367 $\pm$ 0.007	0.304 $\pm$ 0.006	0.220 $\pm$ 0.006	0.537 $\pm$ 0.010	0.534 $\pm$ 0.010	0.534 $\pm$ 0.010	NA	NA	NA
	β -MDecVAE + EWT ($\beta = 0.1$), $Nac = 4$	0.376 $\pm$ 0.007	0.310 $\pm$ 0.007	0.205 $\pm$ 0.006	0.363 $\pm$ 0.008	0.295 $\pm$ 0.007	0.209 $\pm$ 0.006	0.394 $\pm$ 0.010	0.384 $\pm$ 0.011	0.384 $\pm$ 0.011	NA	NA	NA
	β -MDecVAE + FD ($\beta = 0.1$), $Nac = 4$	0.369 $\pm$ 0.007	0.312 $\pm$ 0.007	0.207 $\pm$ 0.006	0.356 $\pm$ 0.006	0.293 $\pm$ 0.006	0.208 $\pm$ 0.006	0.594 $\pm$ 0.008	0.590 $\pm$ 0.008	0.590 $\pm$ 0.008	NA	NA	NA
	β -MDecVAE + EWT ($\beta = 0.1$), $Nac = 4$	0.370 $\pm$ 0.005	0.290 $\pm$ 0.006	0.196 $\pm$ 0.005	0.371 $\pm$ 0.007	0.288 $\pm$ 0.006	0.193 $\pm$ 0.005	0.522 $\pm$ 0.008	0.524 $\pm$ 0.008	0.524 $\pm$ 0.008	NA	NA	NA

Supplementary Table 12: IEMOCAP [22] disentanglement evaluation on Z and S latent spaces (frame and sequence variables). All models are transferred from SimVowels and TIMIT [20] and fine-tuned with SSL and VAE losses on IEMOCAP [22], before being evaluated. Results reported for DecVAEs come from the aggregated joint representation (X + OCs). MI: Mutual Information, TGCN: Total Correlation Normalized, D: Disentanglement, C: Completeness, I: Informativeness, EX: Explicitness, M: Modularity, IRS: Interventional Robustness Score. 95% confidence intervals are reported for metrics D, C, I, EX, M. Values smaller than 0.001 are displayed as 0.001.

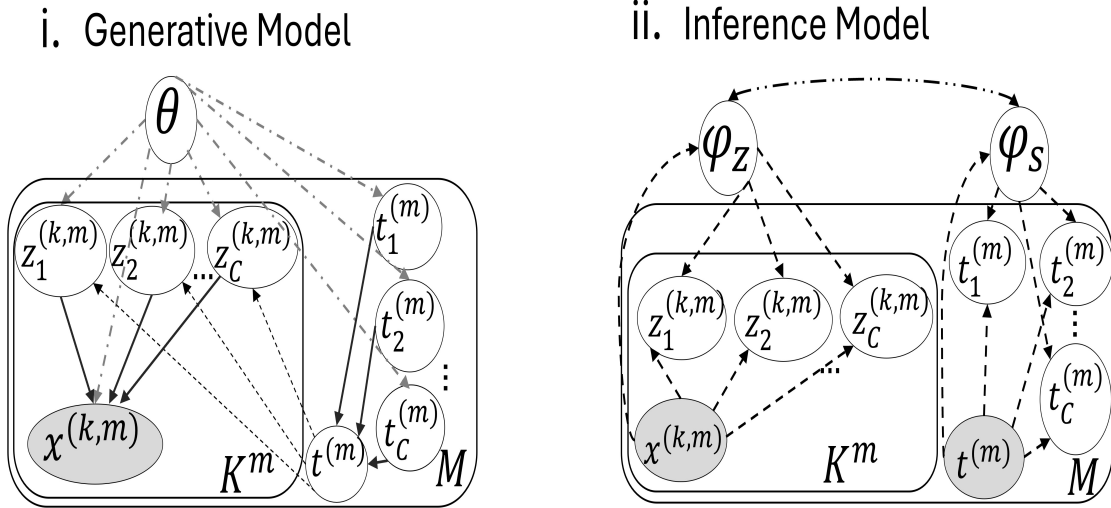
Category	Method	Info-based		Gaussianity	Predictor-based					Intervention-based	
		MI ↓	TGCN ↓	D ↑	C ↑	I ↑	EX ↑	M ↑	IRS ↑		
Raw Inputs	Mel Filterbank Features	0.226	4.571	0.254 ± 0.001	0.224 ± 0.001	0.365 ± 0.001	0.398 ± 0.001	0.981 ± 0.001	0.635		
Eigenanalysis - Mel Fbank Input	PCA	0.003	<< 0.001	0.120 ± 0.001	0.215 ± 0.001	0.358 ± 0.001	0.375 ± 0.001	0.907 ± 0.001	0.610		
	ICA	0.003	<< 0.001	0.128 ± 0.001	0.132 ± 0.001	0.358 ± 0.001	0.375 ± 0.001	0.895 ± 0.001	0.610		
Transfer from SimVowels - VAEs - Mel Fbank	FC-AE	0.110	4.408	0.208 ± 0.001	0.115 ± 0.001	0.341 ± 0.001	0.338 ± 0.001	0.978 ± 0.001	0.620		
	FC-VAE	0.034	0.0699	0.090 ± 0.001	0.221 ± 0.001	0.327 ± 0.001	0.226 ± 0.001	0.963 ± 0.001	0.567		
	FC-β-VAE Mel Fbank (β = 0.1)	0.019	0.084	0.071 ± 0.001	0.238 ± 0.001	0.332 ± 0.001	0.266 ± 0.001	0.907 ± 0.001	0.556		
Transfer from TIMIT - VAEs - Mel Fbank	FC-AE	0.134	10.415	0.240 ± 0.001	0.153 ± 0.001	0.339 ± 0.001	0.325 ± 0.001	0.978 ± 0.001	0.668		
	FC-VAE	0.049	0.104	0.155 ± 0.001	0.129 ± 0.001	0.330 ± 0.001	0.290 ± 0.001	0.971 ± 0.001	0.617		
	FC-β-VAE Mel Fbank (β = 0.1)	0.024	0.078	0.118 ± 0.001	0.199 ± 0.001	0.330 ± 0.001	0.265 ± 0.001	0.966 ± 0.001	0.607		
Transfer from SimVowels - Multivariate DecVAE - Mel Fbank - Z branch (Frame)	MDecAE + FD	0.072	0.365	0.250 ± 0.001	0.127 ± 0.001	0.445 ± 0.001	0.287 ± 0.001	0.925 ± 0.001	0.645		
	MDecAE + EWT	0.086	0.334	0.253 ± 0.001	0.159 ± 0.001	0.439 ± 0.001	0.253 ± 0.001	0.921 ± 0.001	0.731		
	MDecVAE + FD	0.017	0.091	0.233 ± 0.001	0.141 ± 0.001	0.341 ± 0.001	0.277 ± 0.001	0.940 ± 0.001	0.809		
	MDecVAE + EWT	0.017	0.111	0.240 ± 0.001	0.153 ± 0.001	0.341 ± 0.001	0.278 ± 0.001	0.909 ± 0.001	0.576		
	β-MDecVAE + FD (β = 0.1)	0.019	0.116	0.451 ± 0.001	0.293 ± 0.001	0.367 ± 0.001	0.277 ± 0.001	0.755 ± 0.001	0.683		
	β-MDecVAE + EWT (β = 0.1)	0.019	0.053	0.280 ± 0.001	0.160 ± 0.001	0.359 ± 0.001	0.290 ± 0.001	0.889 ± 0.001	0.799		
	MDecAE + FD	0.057	0.341	0.445 ± 0.001	0.309 ± 0.001	0.443 ± 0.001	0.278 ± 0.001	0.876 ± 0.001	0.658		
Transfer from TIMIT - Multivariate DecVAE - Mel Fbank - Z branch (Frame)	MDecAE + EWT	0.059	0.373	0.283 ± 0.001	0.210 ± 0.001	0.442 ± 0.001	0.269 ± 0.001	0.801 ± 0.001	0.673		
	MDecVAE + FD	0.017	0.134	0.289 ± 0.001	0.209 ± 0.001	0.341 ± 0.001	0.285 ± 0.001	0.831 ± 0.001	0.778		
	MDecVAE + EWT	0.019	0.130	0.238 ± 0.001	0.158 ± 0.001	0.335 ± 0.001	0.297 ± 0.001	0.849 ± 0.001	0.808		
	β-MDecVAE + FD (β = 0.1)	0.025	0.127	0.231 ± 0.001	0.118 ± 0.001	0.316 ± 0.001	0.243 ± 0.001	0.788 ± 0.001	0.713		
β-MDecVAE + EWT (β = 0.1)	0.022	0.136	0.254 ± 0.001	0.149 ± 0.001	0.317 ± 0.001	0.242 ± 0.001	0.879 ± 0.001	0.755			
Transfer from SimVowels - Multivariate DecVAE - Mel Fbank - S branch (Sequence)	MDecAE + FD	0.023	0.370	0.105 ± 0.001	0.064 ± 0.001	0.371 ± 0.001	0.524 ± 0.001	0.608 ± 0.001	0.343		
	MDecAE + EWT	0.025	0.464	0.129 ± 0.001	0.065 ± 0.001	0.369 ± 0.001	0.529 ± 0.001	0.697 ± 0.001	0.324		
	MDecVAE + FD	0.010	0.017	0.088 ± 0.001	0.038 ± 0.001	0.333 ± 0.001	0.429 ± 0.001	0.669 ± 0.001	0.540		
	MDecVAE + EWT	0.012	0.061	0.082 ± 0.001	0.030 ± 0.001	0.321 ± 0.001	0.363 ± 0.001	0.553 ± 0.001	0.533		
	β-MDecVAE + FD (β = 0.1)	0.010	0.063	0.083 ± 0.001	0.035 ± 0.001	0.336 ± 0.001	0.457 ± 0.001	0.711 ± 0.001	0.600		
	β-MDecVAE + EWT (β = 0.1)	0.009	0.064	0.059 ± 0.001	0.033 ± 0.001	0.306 ± 0.001	0.432 ± 0.001	0.705 ± 0.001	0.521		
	MDecAE + FD	0.056	0.581	0.135 ± 0.001	0.084 ± 0.001	0.369 ± 0.001	0.514 ± 0.001	0.619 ± 0.001	0.252		
Transfer from TIMIT - Multivariate DecVAE - Mel Fbank - S branch (Sequence)	MDecAE + EWT	0.012	0.515	0.126 ± 0.001	0.063 ± 0.001	0.314 ± 0.001	0.389 ± 0.001	0.778 ± 0.001	0.271		
	MDecVAE + FD	0.015	0.017	0.089 ± 0.001	0.045 ± 0.001	0.326 ± 0.001	0.405 ± 0.001	0.685 ± 0.001	0.780		
	MDecVAE + EWT	0.003	0.023	0.095 ± 0.001	0.045 ± 0.001	0.306 ± 0.001	0.314 ± 0.001	0.686 ± 0.001	0.593		
	β-MDecVAE + FD (β = 0.1)	0.013	0.059	0.089 ± 0.001	0.049 ± 0.001	0.325 ± 0.001	0.416 ± 0.001	0.618 ± 0.001	0.593		
	β-MDecVAE + EWT (β = 0.1)	0.009	0.055	0.057 ± 0.001	0.024 ± 0.001	0.237 ± 0.001	0.256 ± 0.001	0.789 ± 0.001	0.776		

Supplementary Table 13: IEMOCAP [22] dataset classification evaluation. All models are transferred from SimVowels and TIMIT [20] and fine-tuned with SSL and VAE losses on IEMOCAP [22], before being evaluated. Results reported for DecVAEs come from the aggregated joint representation ($X + OCs$). Confidence intervals are displayed for Leave-One-Speaker-Out cross-validation in the case of ER, and 5-fold cross-validation in the case of PR and SI. All cross-validations are repeated for 10 random states and 95% confidence intervals ($\pm$) are calculated taking into account all random states. ER: Emotion Recognition, PR: Phoneme Recognition, SI: Speaker Identification, WA: Weighted Accuracy, UWA: Unweighted Accuracy, WF1: Weighted F1-score.

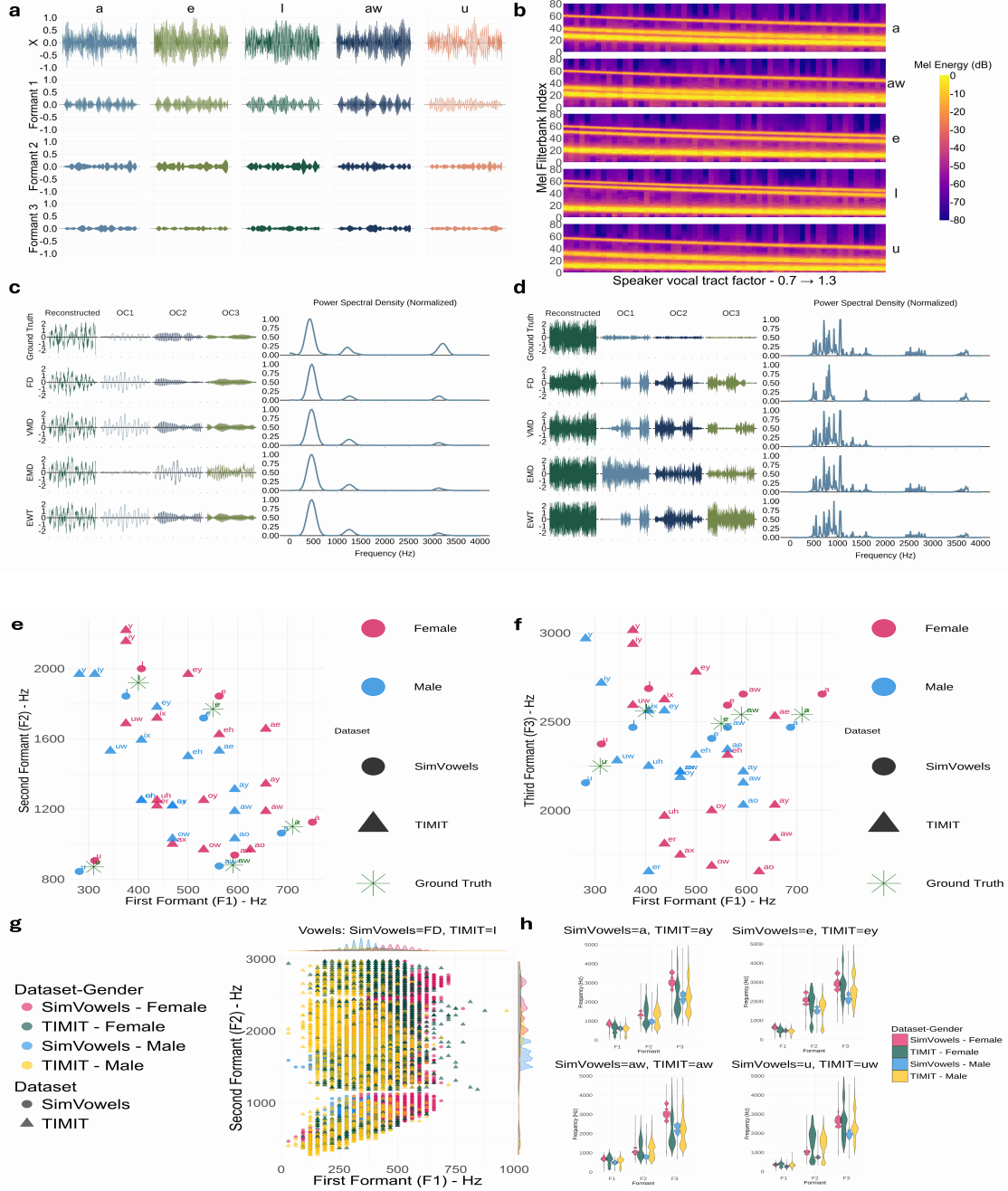
Category	Method	ER			PR		SI		
		WA $\uparrow$	UWA $\uparrow$	WF1 $\uparrow$	WA $\uparrow$	WF1 $\uparrow$	WA $\uparrow$	WF1 $\uparrow$	
Dummy Classifiers	Random Classifier	0.249 $\pm$ 0.002	0.250 $\pm$ 0.001	0.254 $\pm$ 0.004	0.020 $\pm$ 0.001	0.0268 $\pm$ 0.001	0.100 $\pm$ 0.003	0.024 $\pm$ 0.003	
	Majority Classifier	0.300 $\pm$ 0.035	0.250 $\pm$ 0.000	0.140 $\pm$ 0.028	0.453 $\pm$ 0.001	0.282 $\pm$ 0.001	0.116 $\pm$ 0.001	0.100 $\pm$ 0.001	
Raw Inputs	Mel Filterbank Features	0.405 $\pm$ 0.008	0.403 $\pm$ 0.010	0.405 $\pm$ 0.008	0.508 $\pm$ 0.001	0.401 $\pm$ 0.001	0.247 $\pm$ 0.001	0.242 $\pm$ 0.001	
Eigenanalysis - Mel Fbank	PCA	0.395 $\pm$ 0.008	0.391 $\pm$ 0.009	0.394 $\pm$ 0.008	0.504 $\pm$ 0.001	0.399 $\pm$ 0.001	0.211 $\pm$ 0.001	0.206 $\pm$ 0.001	
	ICA	0.388 $\pm$ 0.008	0.381 $\pm$ 0.009	0.385 $\pm$ 0.008	0.499 $\pm$ 0.001	0.383 $\pm$ 0.001	0.206 $\pm$ 0.001	0.201 $\pm$ 0.001	
Transfer from SimVowels - VAEs - Mel Fbank	FC-AE	0.368 $\pm$ 0.007	0.364 $\pm$ 0.010	0.365 $\pm$ 0.007	0.489 $\pm$ 0.001	0.376 $\pm$ 0.001	0.182 $\pm$ 0.001	0.179 $\pm$ 0.001	
	FC-VAE	0.367 $\pm$ 0.007	0.360 $\pm$ 0.008	0.363 $\pm$ 0.008	0.482 $\pm$ 0.001	0.359 $\pm$ 0.001	0.163 $\pm$ 0.001	0.158 $\pm$ 0.001	
	FC- β -VAE Mel Fbank ($\beta = 0.1$)	0.372 $\pm$ 0.007	0.366 $\pm$ 0.008	0.369 $\pm$ 0.008	0.489 $\pm$ 0.001	0.371 $\pm$ 0.001	0.170 $\pm$ 0.001	0.166 $\pm$ 0.001	
Transfer from TIMIT - VAEs - Mel Fbank	FC-AE	0.364 $\pm$ 0.007	0.358 $\pm$ 0.008	0.361 $\pm$ 0.008	0.484 $\pm$ 0.001	0.360 $\pm$ 0.001	0.184 $\pm$ 0.001	0.180 $\pm$ 0.001	
	FC-VAE	0.363 $\pm$ 0.006	0.360 $\pm$ 0.007	0.362 $\pm$ 0.006	0.481 $\pm$ 0.001	0.376 $\pm$ 0.001	0.154 $\pm$ 0.001	0.151 $\pm$ 0.001	
	FC- β -VAE Mel Fbank ($\beta = 0.1$)	0.366 $\pm$ 0.008	0.359 $\pm$ 0.008	0.362 $\pm$ 0.008	0.484 $\pm$ 0.001	0.365 $\pm$ 0.001	0.166 $\pm$ 0.001	0.163 $\pm$ 0.001	
Transfer from SimVowels - Multivariate DecVAE - Mel Fbank - Z branch (Frame)	MDecAE + FD	0.369 $\pm$ 0.019	0.365 $\pm$ 0.023	0.366 $\pm$ 0.019	0.540 $\pm$ 0.001	0.458 $\pm$ 0.001	0.741 $\pm$ 0.001	0.744 $\pm$ 0.001	
	MDecAE + EWT	0.373 $\pm$ 0.017	0.371 $\pm$ 0.020	0.371 $\pm$ 0.016	0.541 $\pm$ 0.001	0.458 $\pm$ 0.001	0.712 $\pm$ 0.001	0.713 $\pm$ 0.001	
	MDecVAE + FD	0.357 $\pm$ 0.018	0.350 $\pm$ 0.020	0.353 $\pm$ 0.017	0.536 $\pm$ 0.001	0.451 $\pm$ 0.001	0.737 $\pm$ 0.001	0.741 $\pm$ 0.001	
	MDecVAE + EWT	0.359 $\pm$ 0.023	0.359 $\pm$ 0.025	0.357 $\pm$ 0.021	0.537 $\pm$ 0.001	0.453 $\pm$ 0.001	0.766 $\pm$ 0.001	0.768 $\pm$ 0.001	
	β -MDecVAE + FD($\beta = 0.1$)	0.354 $\pm$ 0.024	0.332 $\pm$ 0.022	0.351 $\pm$ 0.026	0.533 $\pm$ 0.001	0.447 $\pm$ 0.001	0.737 $\pm$ 0.001	0.740 $\pm$ 0.001	
	β -MDecVAE + EWT ($\beta = 0.1$)	0.360 $\pm$ 0.016	0.357 $\pm$ 0.017	0.357 $\pm$ 0.015	0.533 $\pm$ 0.001	0.443 $\pm$ 0.001	0.766 $\pm$ 0.001	0.714 $\pm$ 0.001	
Transfer from TIMIT - Multivariate DecVAE - Mel Fbank - Z branch (Frame)	MDecAE + FD	0.361 $\pm$ 0.016	0.354 $\pm$ 0.021	0.357 $\pm$ 0.016	0.534 $\pm$ 0.001	0.447 $\pm$ 0.001	0.727 $\pm$ 0.002	0.731 $\pm$ 0.002	
	MDecAE + EWT	0.358 $\pm$ 0.016	0.351 $\pm$ 0.022	0.354 $\pm$ 0.017	0.534 $\pm$ 0.001	0.444 $\pm$ 0.001	0.684 $\pm$ 0.001	0.688 $\pm$ 0.001	
	MDecVAE + FD	0.365 $\pm$ 0.020	0.361 $\pm$ 0.023	0.354 $\pm$ 0.017	0.534 $\pm$ 0.001	0.446 $\pm$ 0.001	0.718 $\pm$ 0.001	0.722 $\pm$ 0.001	
	MDecVAE + EWT	0.367 $\pm$ 0.023	0.365 $\pm$ 0.027	0.366 $\pm$ 0.023	0.533 $\pm$ 0.001	0.444 $\pm$ 0.001	0.715 $\pm$ 0.001	0.718 $\pm$ 0.001	
	β -MDecVAE + FD($\beta = 0.1$)	0.365 $\pm$ 0.020	0.339 $\pm$ 0.018	0.363 $\pm$ 0.019	0.528 $\pm$ 0.001	0.437 $\pm$ 0.001	0.677 $\pm$ 0.001	0.685 $\pm$ 0.001	
	β -MDecVAE + EWT ($\beta = 0.1$)	0.341 $\pm$ 0.014	0.332 $\pm$ 0.018	0.338 $\pm$ 0.015	0.534 $\pm$ 0.001	0.436 $\pm$ 0.001	0.672 $\pm$ 0.001	0.678 $\pm$ 0.001	
Transfer from SimVowels - Multivariate DecVAE - Mel Fbank - S branch (Sequence)	MDecAE + FD	0.460 $\pm$ 0.020	0.459 $\pm$ 0.026	0.459 $\pm$ 0.019	NA	NA	0.300 $\pm$ 0.011	0.296 $\pm$ 0.012	
	MDecAE + EWT	0.459 $\pm$ 0.024	0.461 $\pm$ 0.024	0.457 $\pm$ 0.024	NA	NA	0.321 $\pm$ 0.015	0.318 $\pm$ 0.015	
	MDecVAE + FD	0.440 $\pm$ 0.021	0.440 $\pm$ 0.029	0.439 $\pm$ 0.019	NA	NA	0.224 $\pm$ 0.026	0.222 $\pm$ 0.027	
	MDecVAE + EWT	0.420 $\pm$ 0.033	0.446 $\pm$ 0.025	0.414 $\pm$ 0.033	NA	NA	0.210 $\pm$ 0.015	0.204 $\pm$ 0.015	
	β -MDecVAE + FD($\beta = 0.1$)	0.436 $\pm$ 0.018	0.432 $\pm$ 0.027	0.436 $\pm$ 0.017	NA	NA	0.228 $\pm$ 0.026	0.223 $\pm$ 0.027	
	β -MDecVAE + EWT ($\beta = 0.1$)	0.448 $\pm$ 0.021	0.446 $\pm$ 0.025	0.447 $\pm$ 0.022	NA	NA	0.230 $\pm$ 0.009	0.222 $\pm$ 0.010	
Transfer from TIMIT - Multivariate DecVAE - Mel Fbank - S branch (Sequence)	MDecAE + FD	0.462 $\pm$ 0.029	0.461 $\pm$ 0.034	0.462 $\pm$ 0.027	NA	NA	0.308 $\pm$ 0.010	0.305 $\pm$ 0.010	
	MDecAE + EWT	0.419 $\pm$ 0.015	0.416 $\pm$ 0.026	0.417 $\pm$ 0.015	NA	NA	0.206 $\pm$ 0.008	0.203 $\pm$ 0.019	
	MDecVAE + FD	0.429 $\pm$ 0.024	0.425 $\pm$ 0.024	0.430 $\pm$ 0.023	NA	NA	0.208 $\pm$ 0.018	0.206 $\pm$ 0.019	
	MDecVAE + EWT	0.411 $\pm$ 0.025	0.361 $\pm$ 0.024	0.411 $\pm$ 0.024	NA	NA	0.164 $\pm$ 0.007	0.158 $\pm$ 0.001	
	β -MDecVAE + FD($\beta = 0.1$)	0.396 $\pm$ 0.017	0.389 $\pm$ 0.019	0.396 $\pm$ 0.016	NA	NA	0.220 $\pm$ 0.016	0.250 $\pm$ 0.016	
	β -MDecVAE + EWT ($\beta = 0.1$)	0.366 $\pm$ 0.021	0.361 $\pm$ 0.024	0.365 $\pm$ 0.021	NA	NA	0.156 $\pm$ 0.012	0.148 $\pm$ 0.012	



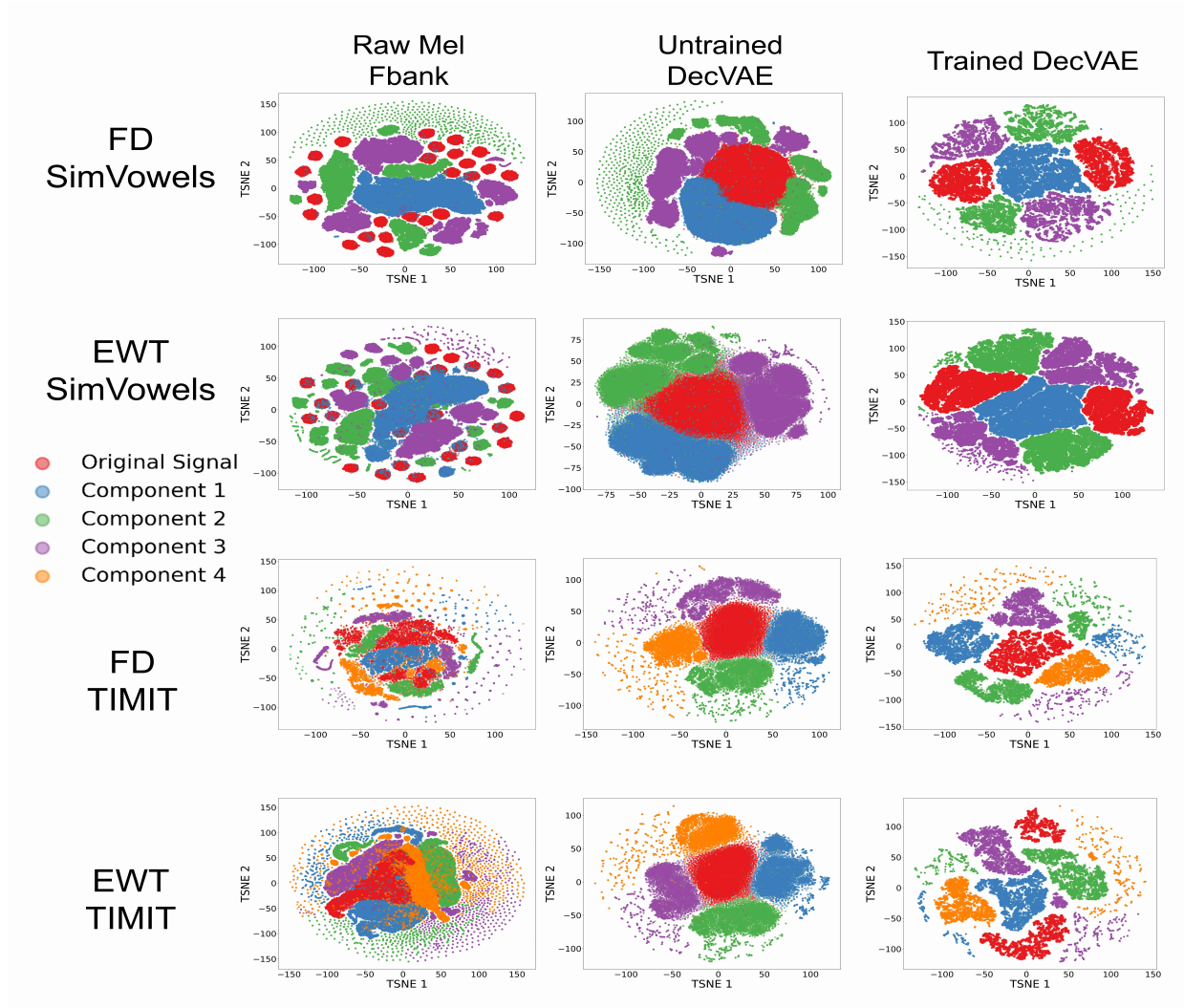
Supplementary Figure 2: Decomposition VAE architecture for a single latent variable Z . The decomposition model D_w^C yields a decomposed input space X' of C components and the original signal. D_w^C also yields a mask for using specific frames of the input to calculate losses. Mel filterbank features are fed into a Wav2Vec2 encoder [23]. Wav2Vec2 features are projected in an intermediate space H where the L_{recon} and L_{ortho} parts of the DELBO are calculated. A further projection from single-layered linear latent projectors yields the subspaces $\tilde{z}_b$. The approximation loss is calculated on the subspaces followed by an aggregation function $g(\cdot)$ that leads to the disentangled latent space Z .



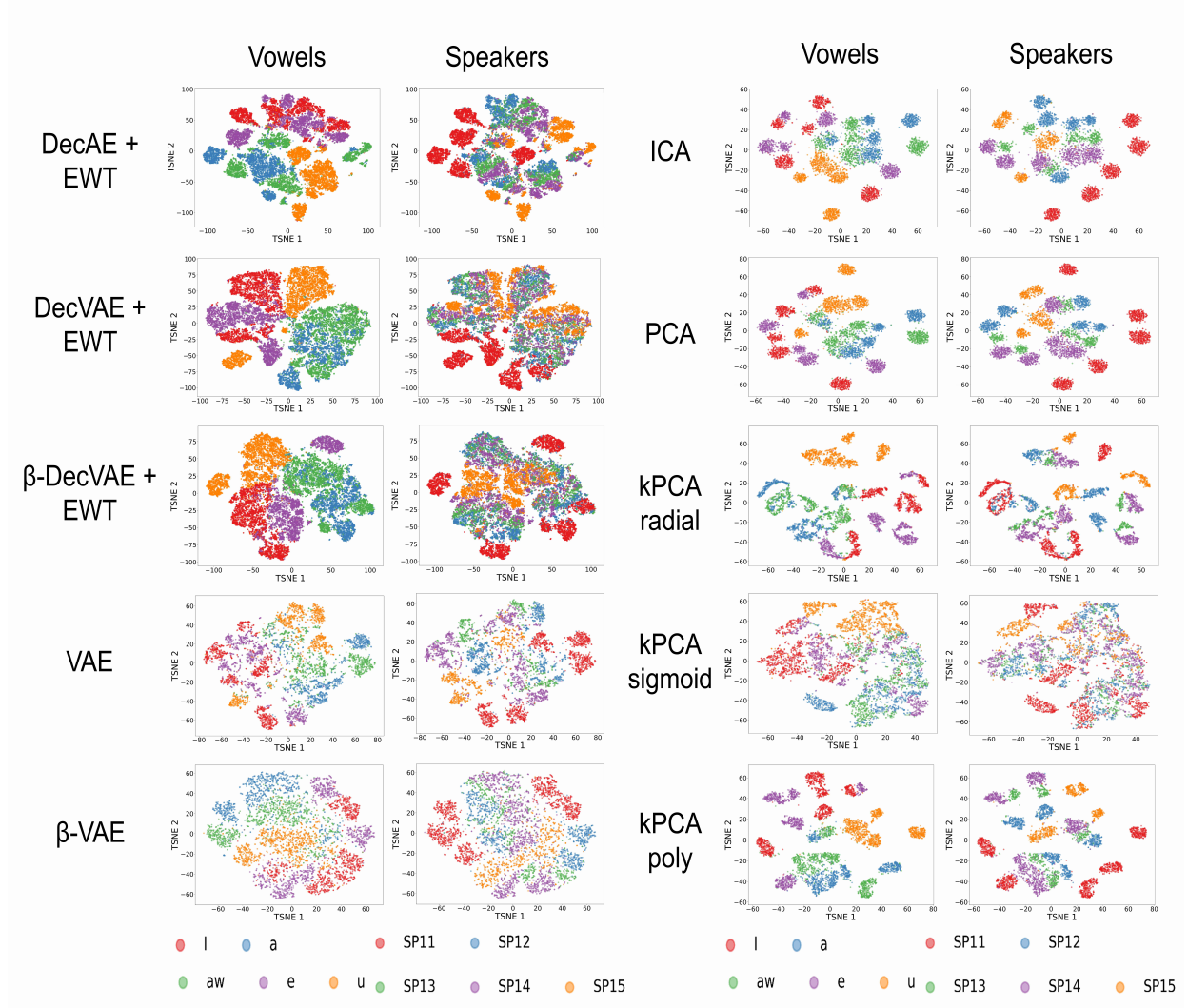
Supplementary Figure 3: Multiple branches directed factor graph of the generative and inference models in variational decomposition autoencoding with multiple latent variables. (i) The generative model that accounts for multiple subspaces $z_1^{(k,m)}, \dots, z_C^{(k,m)}$ and $t_1^{(m)}, \dots, t_C^{(m)}$ in two different time scales. The long-term variable t has no variation inside the m -th sequence and influences fast-varying variables $z^{k,m}$ that change within a time-frame K^m . (ii) Inference model that consists of two separate models ϕ_z and ϕ_s that perform inference on the frame and sequence level variables.



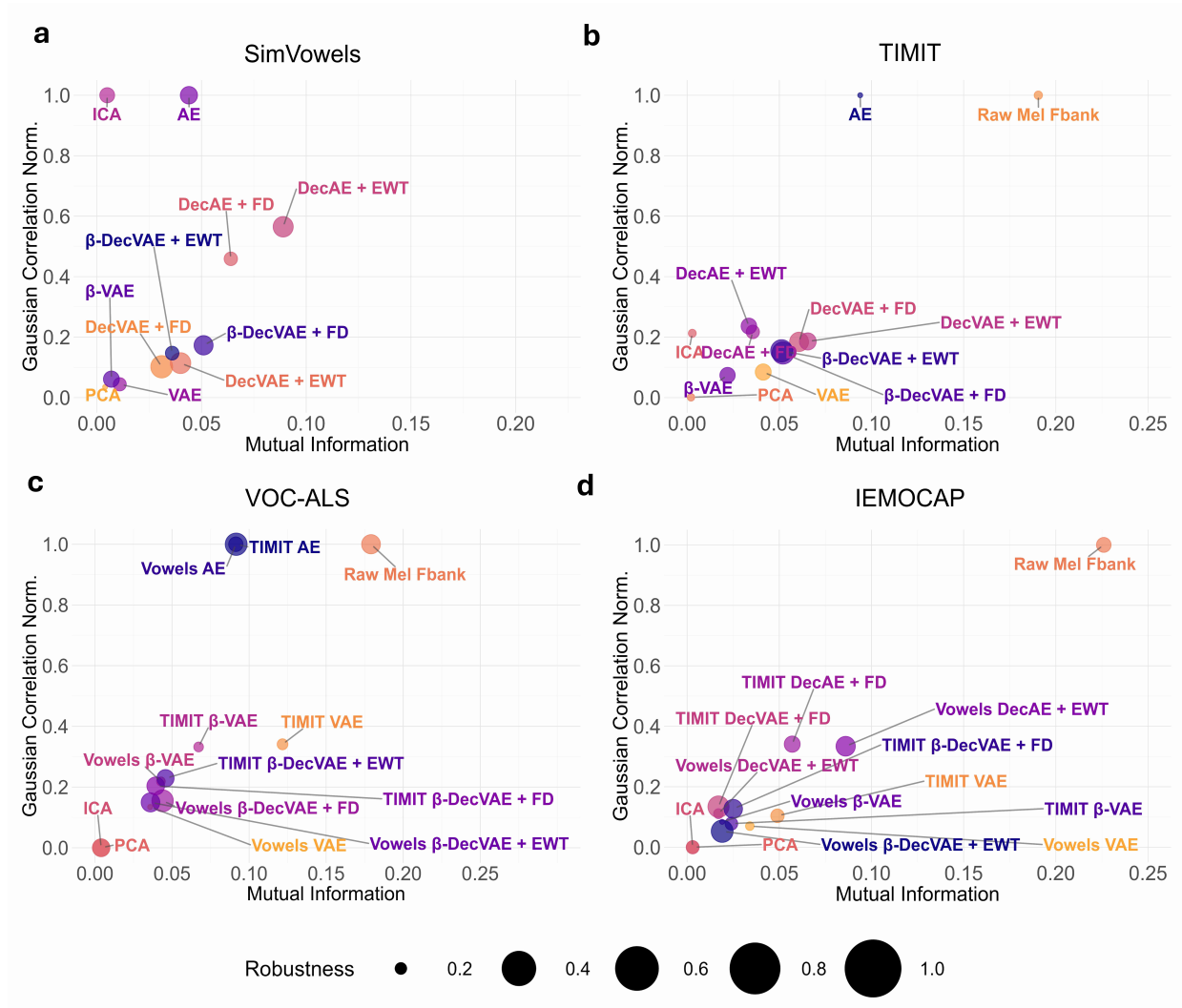
Supplementary Figure 4: Decomposition as a generative and recognition-front-end model, and simulated vowels frequency analysis and comparison with real data from TIMIT [20]. (a): generative process of five simulated vowels and their formants, for a single speaker time domain. (b): generative process of five vowels in the mel filterbank domain, for all speakers. Speakers are modeled through a vocal tract factor, a value ranging in $[0.7, 1.3]$ that divides the F_0 to simulate different speakers. (c): recognition front-end $\mathcal{D}_w^C$ for different methods [14, 13, 15] in the time (left) and frequency domains (right), at the frame (25ms) level. (d): recognition front-end $\mathcal{D}_w^C$ at the sequence level (4s). (e,f): Median formant scatter plot for the SimVowels and TIMIT [20] datasets, as detected by the filter decomposition (FD). FD can detect the vowels' formants successfully (w.r.t. the ground truth (in green)), and the detected simulated vowels are also close to the real vowels from TIMIT [20]. (g): phoneme /I/, scattered detected formants per sample with FD; distributions of real and simulated data have similar properties. (h): violin plots of formant distributions for four vowels, SimVowels and TIMIT [20] datasets.



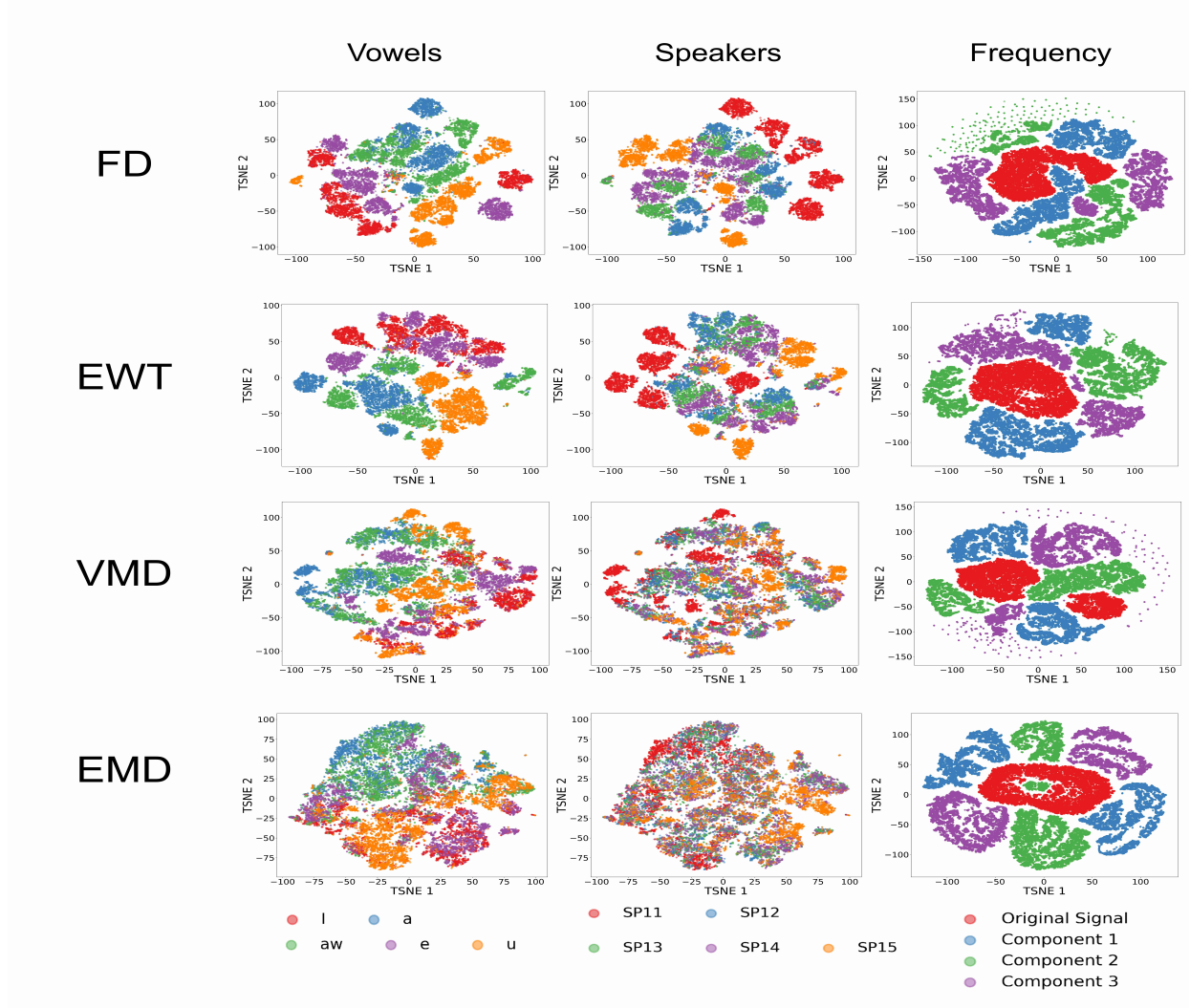
Supplementary Figure 5: Comparison of frequency correspondence of DecVAE latent space $\tilde{Z}$ TSNE [24] projection (after aggregation of subspaces), on SimVowels and TIMIT [20] development sets latent embeddings for two decomposition methods (FD,EWT). First column: raw input features X'' . Second column: projection onto $\tilde{Z}$ through an untrained DecVAE. Third column: projection onto $\tilde{Z}$ through a trained DecVAE. The space $\tilde{Z}$ demonstrates the frequency-correspondent subspaces learned by DecVAE on the complete development set.



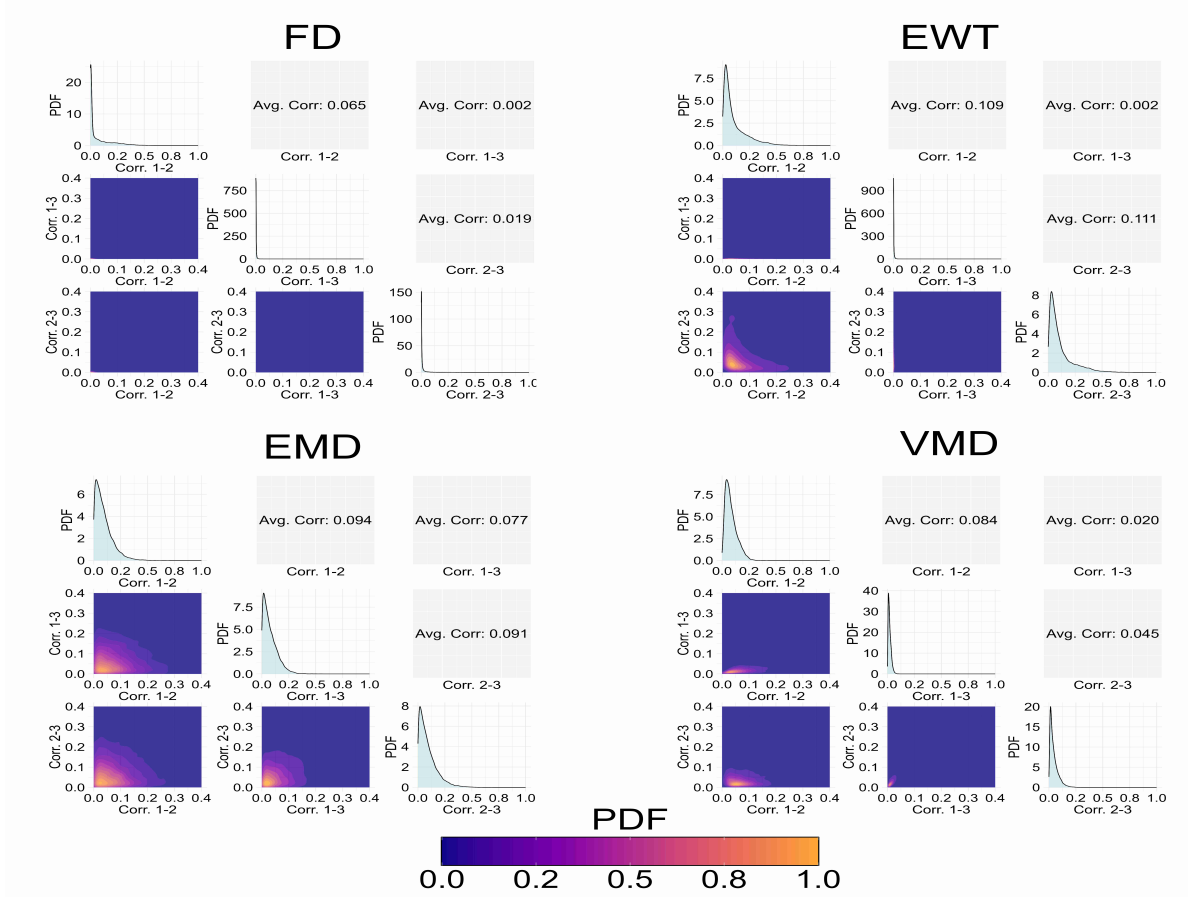
Supplementary Figure 6: Comparison of DecVAEs and supporting methods [25, 6, 9, 7, 8, 15] in vowel and speaker correspondence of latent space Z TSNE [24] manifold, on SimVowels DecVAE. We visualize around two batches (30 utterances) from the development set, with all five vowels and five speakers.



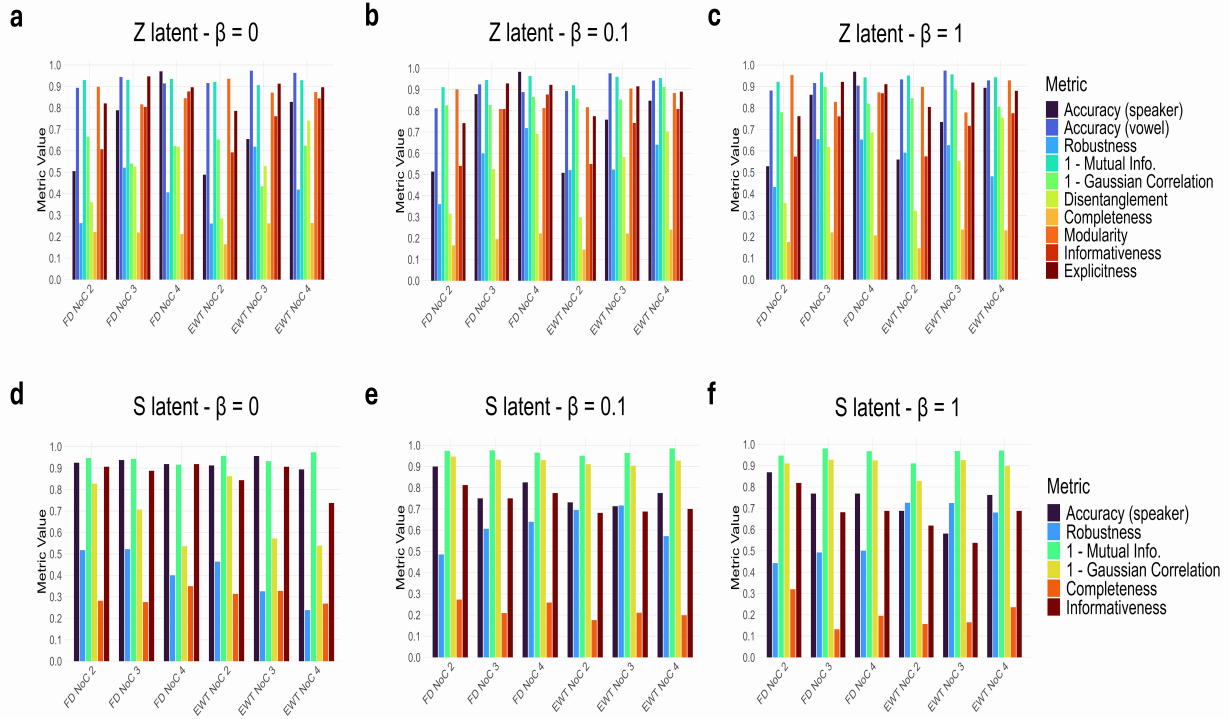
Supplementary Figure 7: Unsupervised disentanglement evaluation (mutual information, gaussian correlation norm. and robustness (IRS)) [26] on all datasets for DecVAE models and supporting methods.



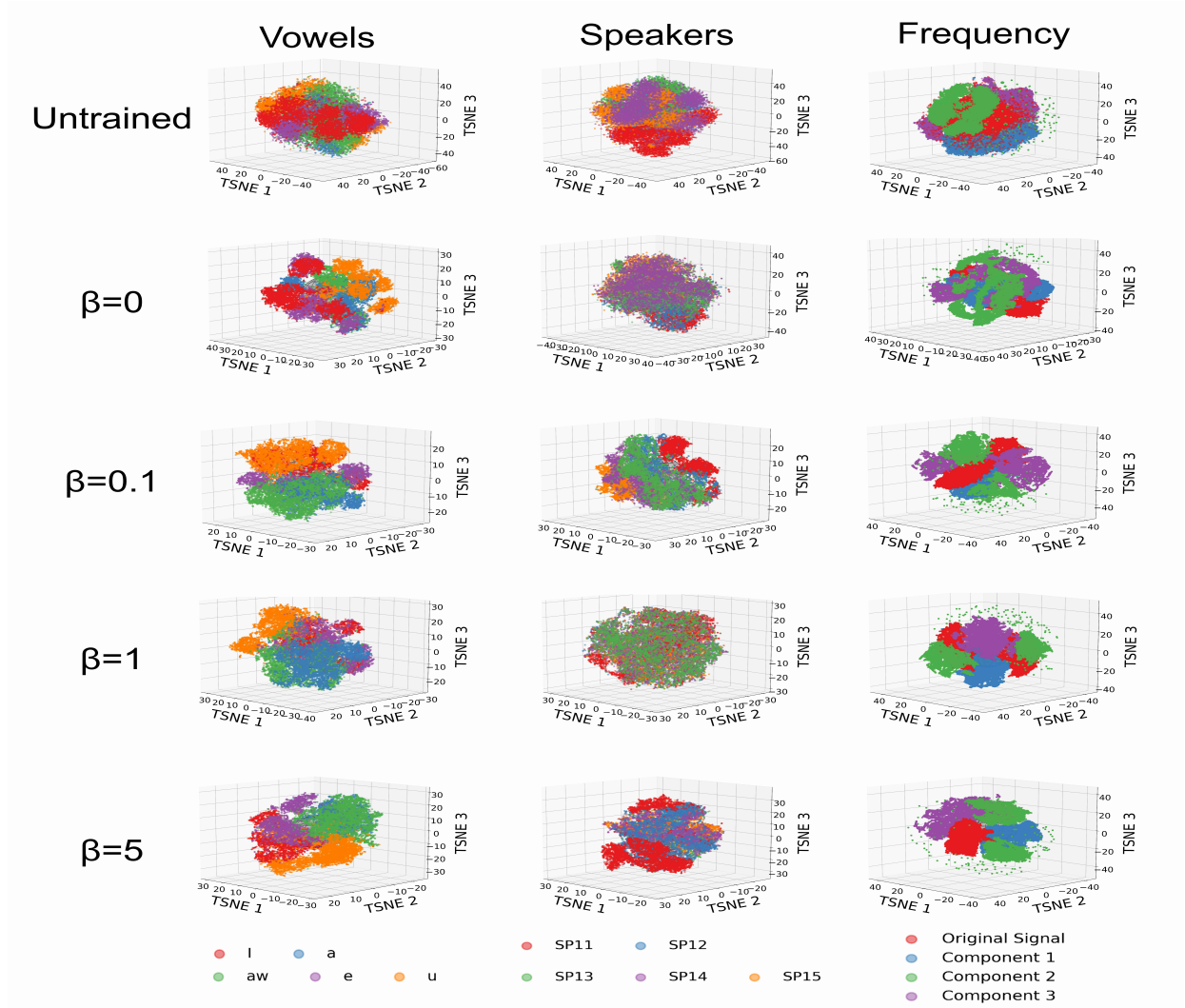
Supplementary Figure 8: Comparison of vowel, speaker and frequency correspondence of latent space Z (first and second column) and $\tilde{Z}$ (third column) TSNE [24] projections, on SimVowels DecVAE latent embeddings for four decomposition methods (FD,EWT[15],VMD[14],EMD[13]). We visualize around two batches (30 utterances) from the development set, with all five vowels and five speakers.



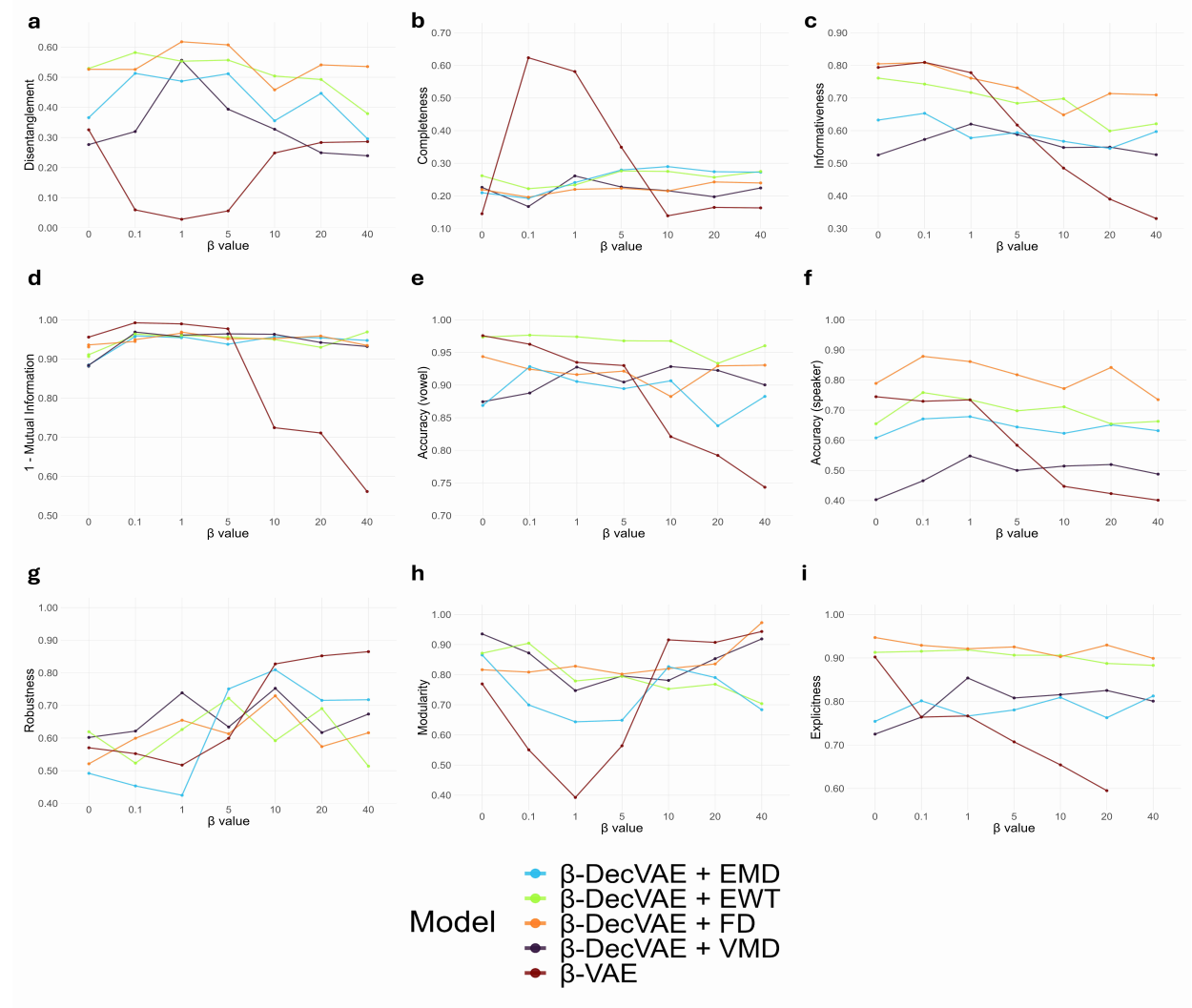
Supplementary Figure 9: Correlation distributions between components and correlation between the correlations of component pairs. Comparison between four decomposition methods (FD, EWT[15],VMD [14], EMD[13])for the detection of phoneme /ax/ from the TIMIT [20] dataset. FD presents the lowest correlation between components across the four methods. EWT performs generally well but allows information leakage between the first and second components. EMD and VMD allow greater mode mixing evident from correlations between all pairs of components.



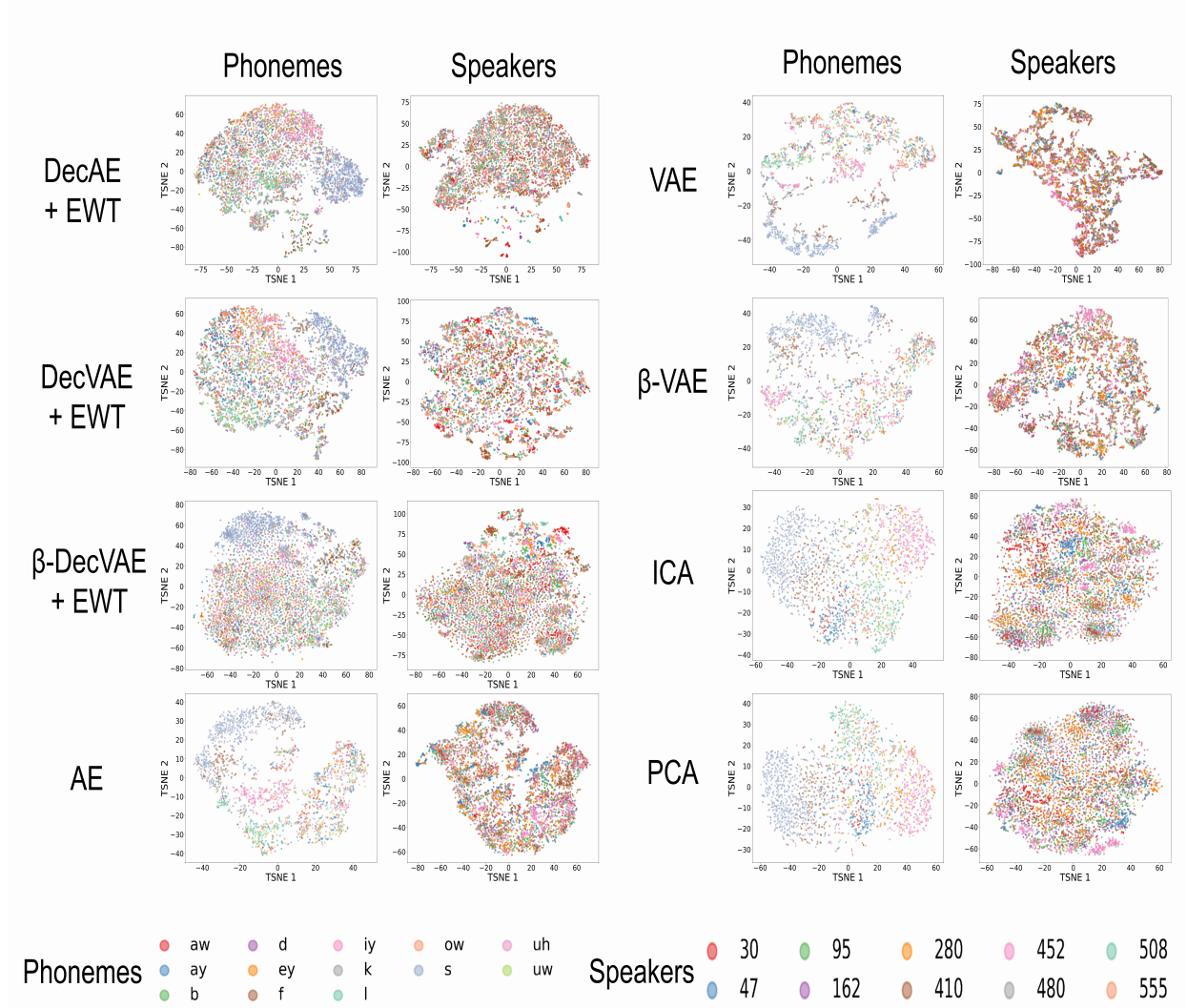
Supplementary Figure 10: Number of components (NoC) C ablation on SimVowels metrics, for the frame variable Z (a,b,c) and the sequence variable S (d,e,f), two options for the decomposition model $\mathcal{D}_w^C$ (EWT [15],FD) and different β values.



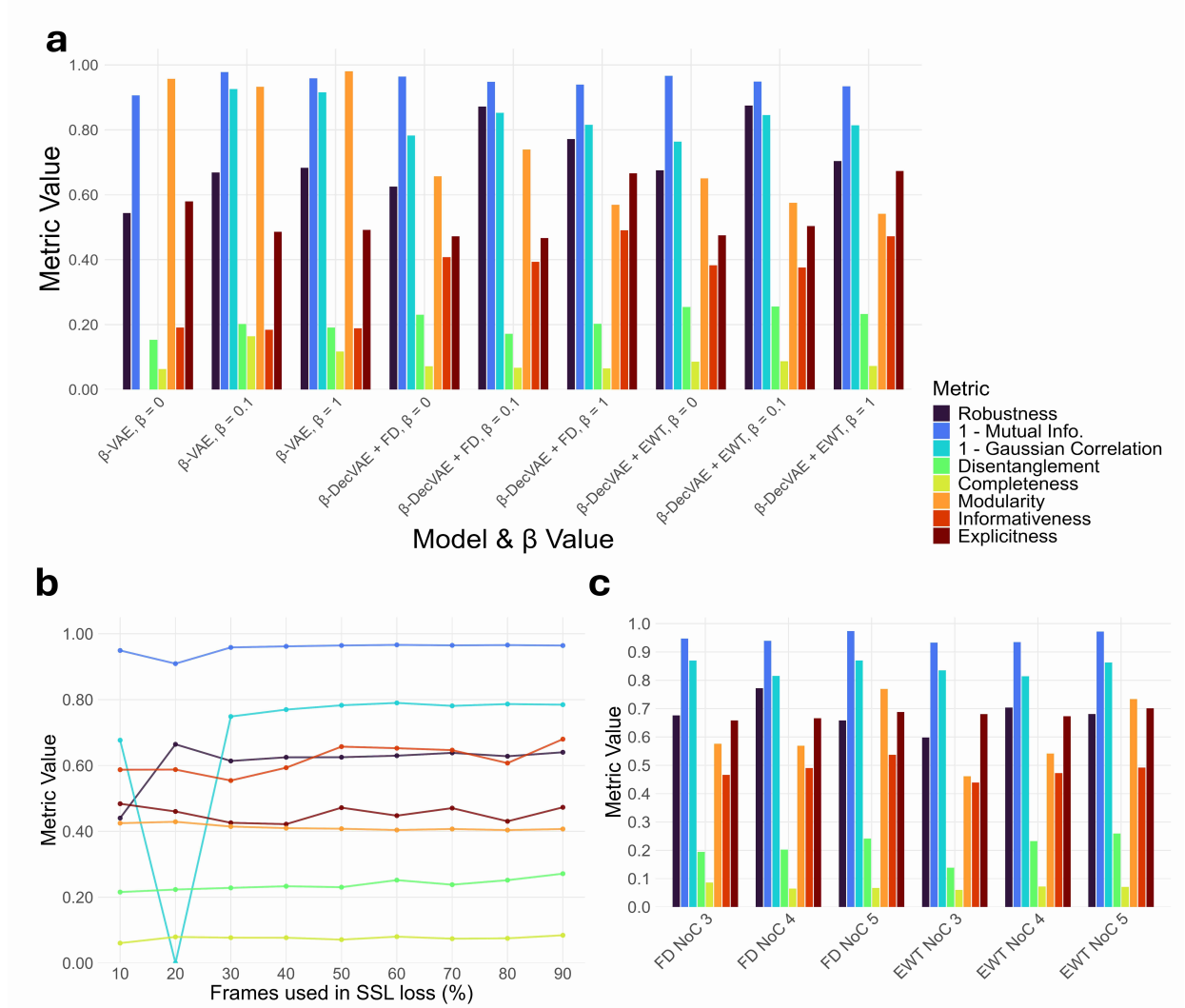
Supplementary Figure 11: Comparison of vowel, speaker and frequency correspondence of latent space Z (first and second column) and $\hat{Z}$ (third column) TSNE [24] projection on a three-dimensional manifold, on SimVowels DecVAE latent embeddings for FD. The untrained DecVAE and different β values are presented in each row. We visualize around two batches (30 utterances) from the development set, with all five vowels and five speakers.



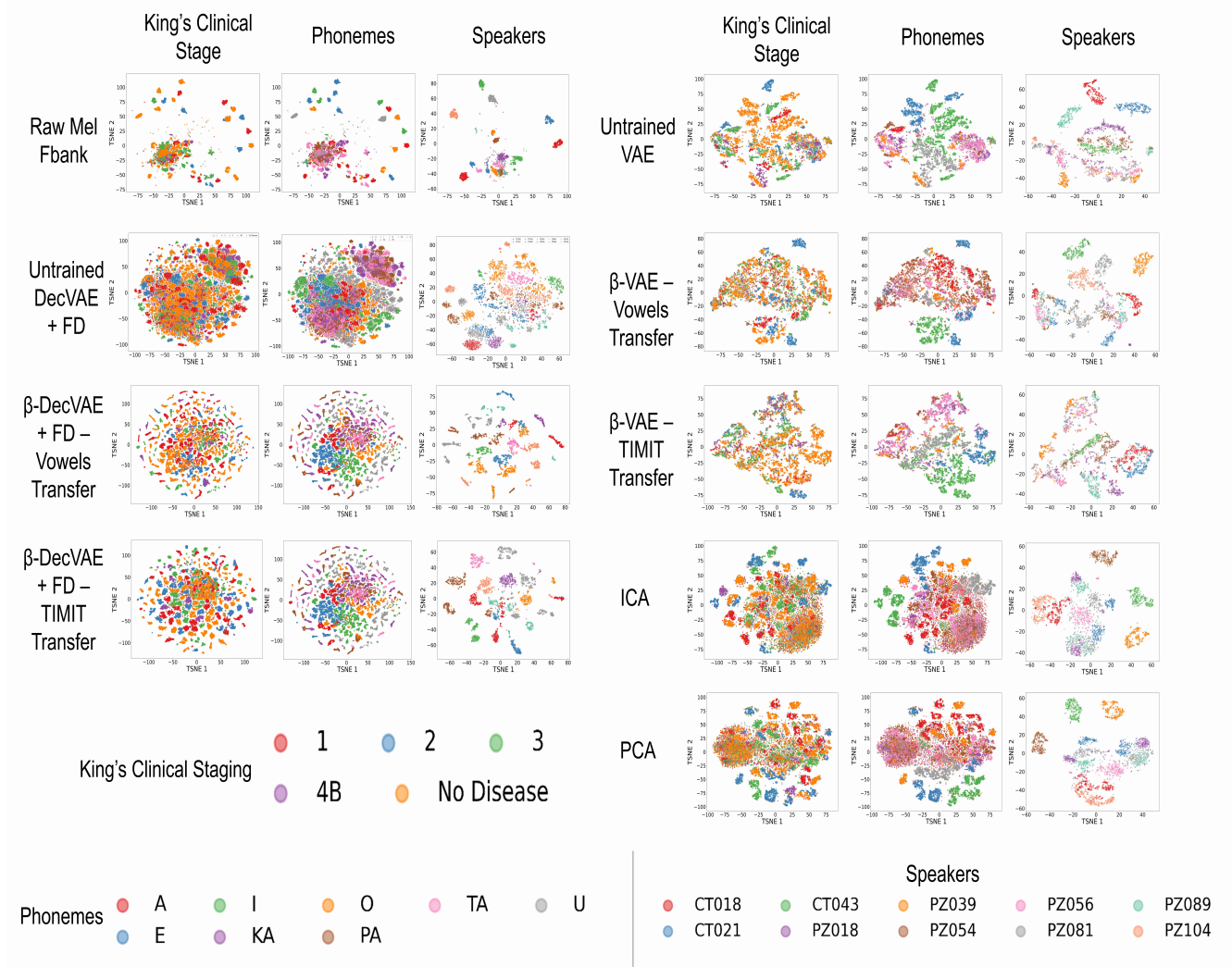
Supplementary Figure 12: Post-training beta ablation on evaluation metrics, SimVowels dataset. (a): disentanglement[18][18], (b): completeness[18], (c): informativeness[18], (d): 1 - mutual information, (e): (weighted) vowel recognition accuracy, (f): (weighted) speaker identification accuracy, (g): robustness (IRS)[26], (h): modularity [18], (i): explicitness[18].



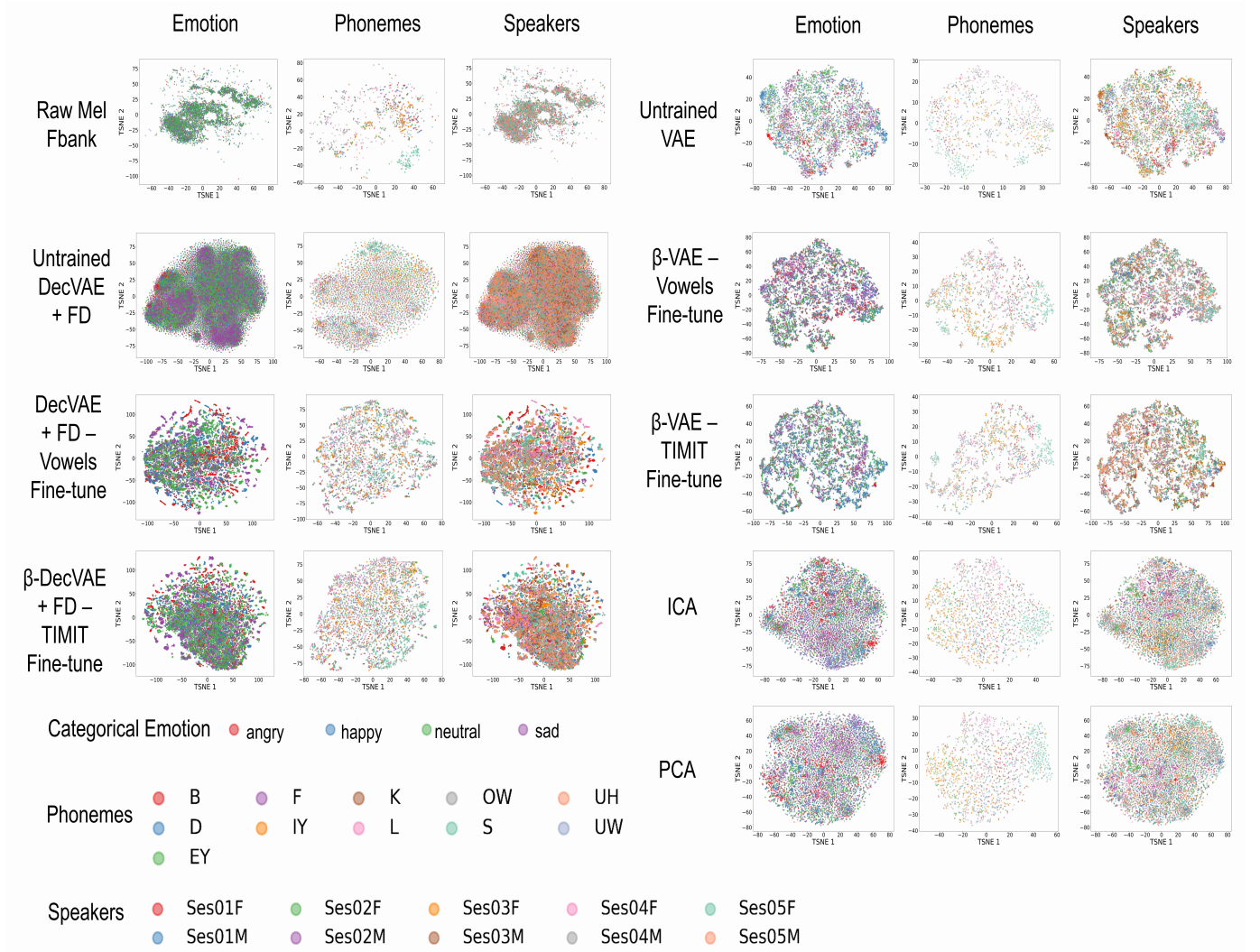
Supplementary Figure 13: Comparison of DecVAEs and supporting methods [25, 6, 9, 7, 15] in phoneme and speaker correspondence of latent space Z TSNE [24] manifold, on TIMIT [20]. We visualize a subset of the available phonemes (thirteen) and a subset of the available speakers (ten), using around 6 batches (50 utterances) from the development set.



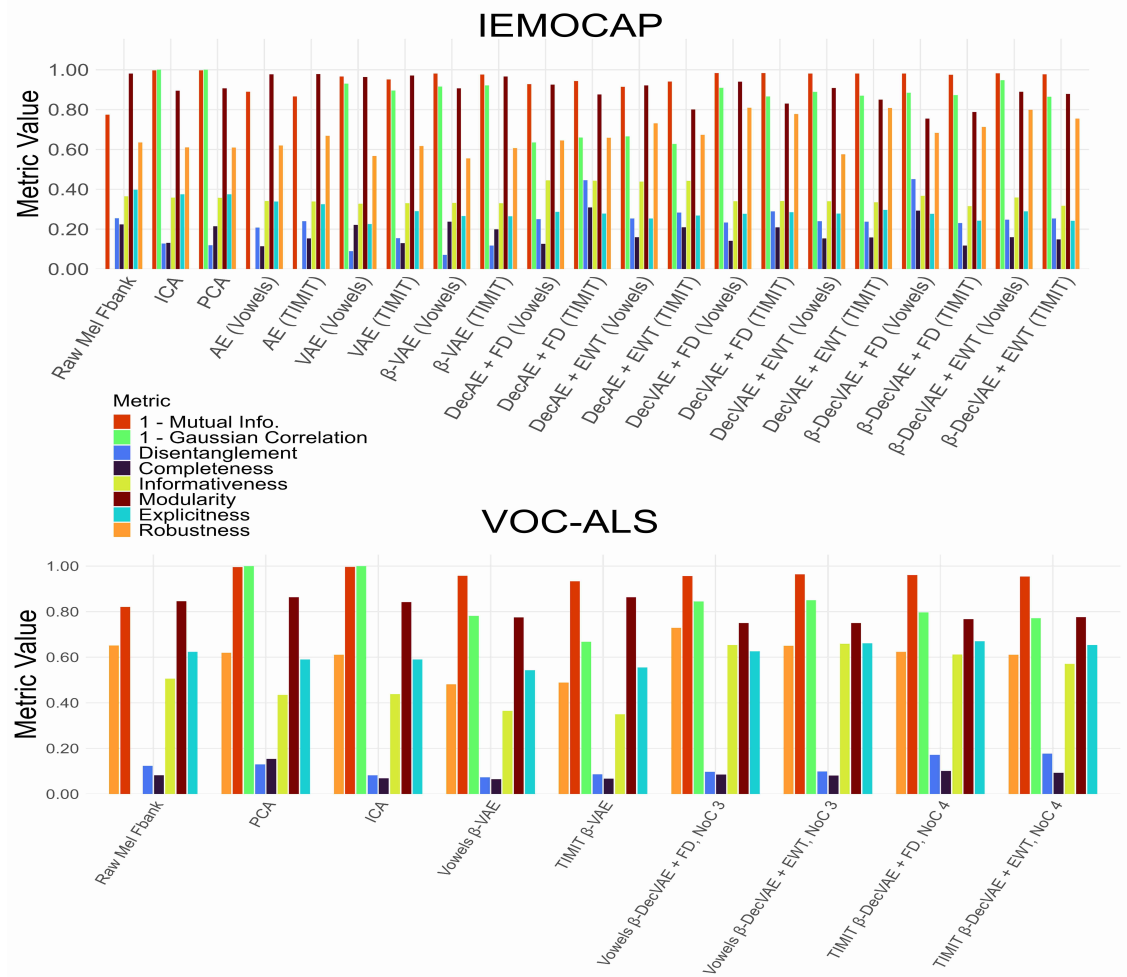
Supplementary Figure 14: Post-training ablations on TIMIT [20] disentanglement metrics. (a) β ablation for β -VAEs [6] and β -DecVAEs. (b) used frame percentage in loss ablation, for FD and $\beta = 0$. (c) number of components C ablation, for FD, EWT[15] and $\beta = 1$.



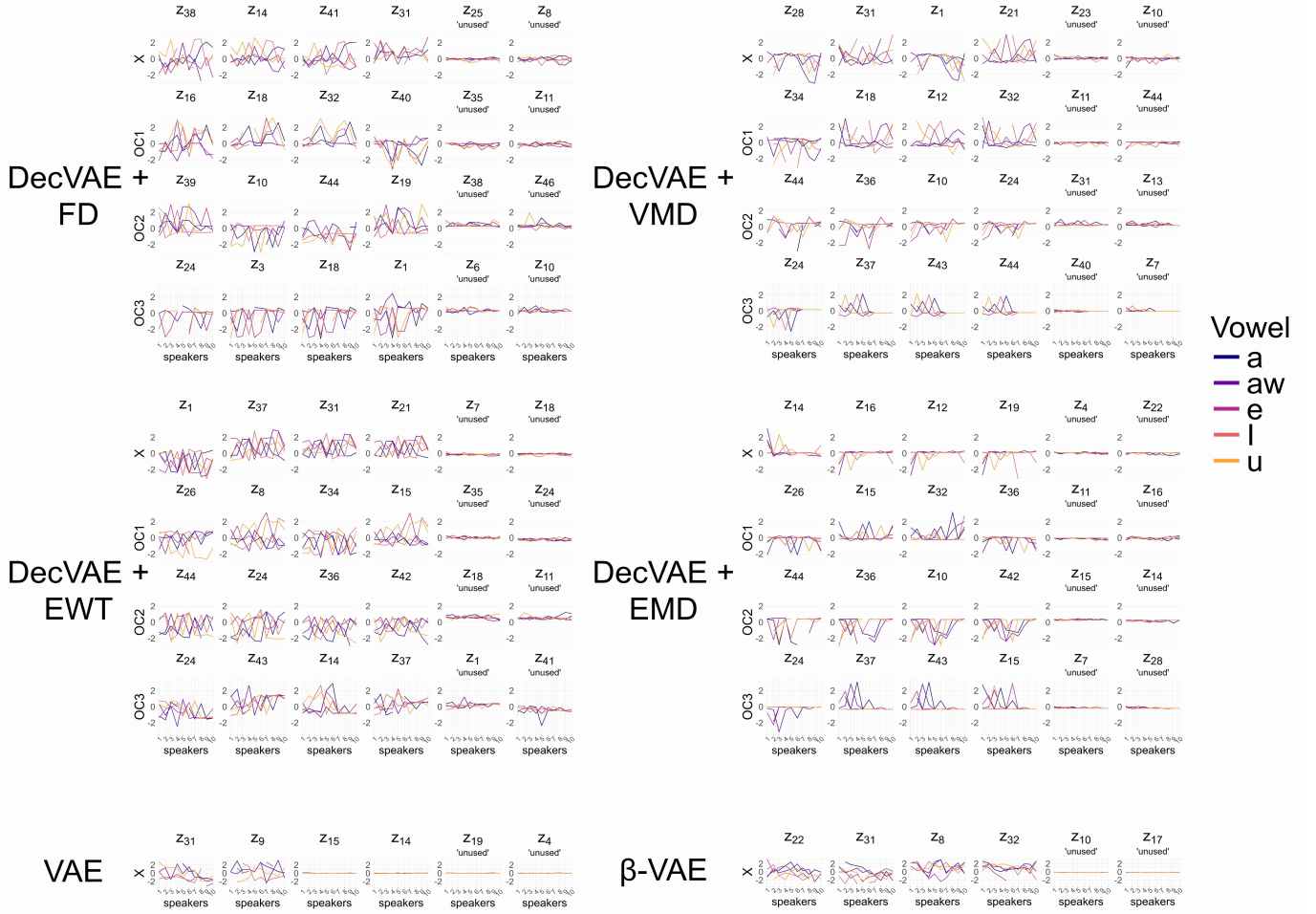
Supplementary Figure 15: Comparison of DecVAEs and supporting methods [25, 6, 9, 7] in King's clinical stage, phoneme and speaker correspondence of latent space $\mathcal{Z}$ TSNE [24] manifold, on VOC-ALS [21]. We visualize a subset of the available phonemes (thirteen) and a subset of the available speakers (ten), using around 4 batches (30 utterances). For DecVAE and VAE models we demonstrate embeddings where a model trained on SimVowels or TIMIT[20] is transferred and zero-shot evaluated on VOC-ALS [21].



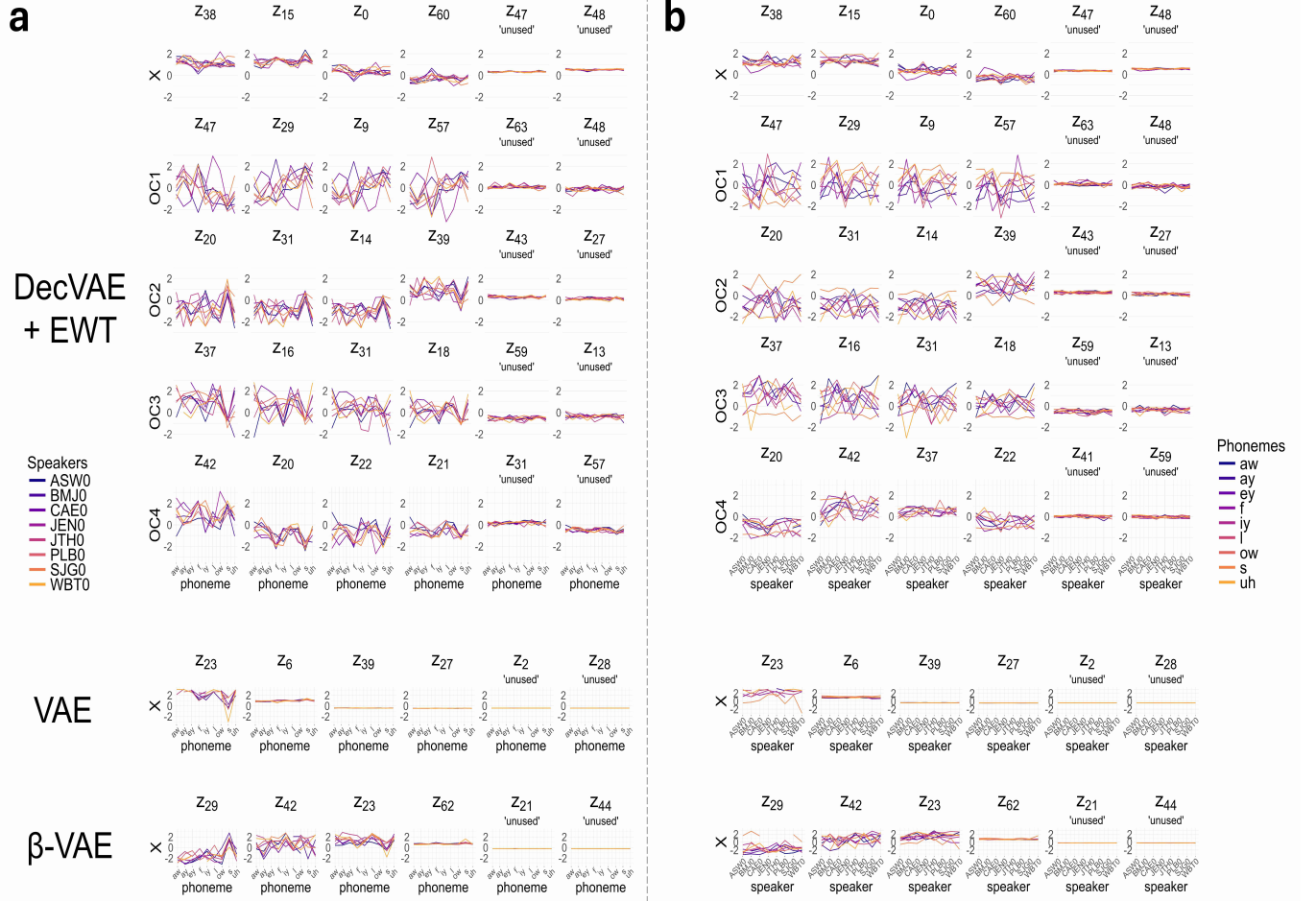
Supplementary Figure 16: Comparison of DecVAEs ($\beta = 0$) and supporting methods [25, 6, 9, 7] in categorical emotion, phoneme and speaker correspondence of latent space Z TSNE [24] manifold, on IEMOCAP [22]. We visualize a subset of the available phonemes (eleven) and all available speakers (ten), using around 4 batches (30 utterances). For DecVAE and VAE models we demonstrate embeddings where a model trained on SimVowels or TIMIT[20] is transferred, fine-tuned and then evaluated on IEMOCAP [21].



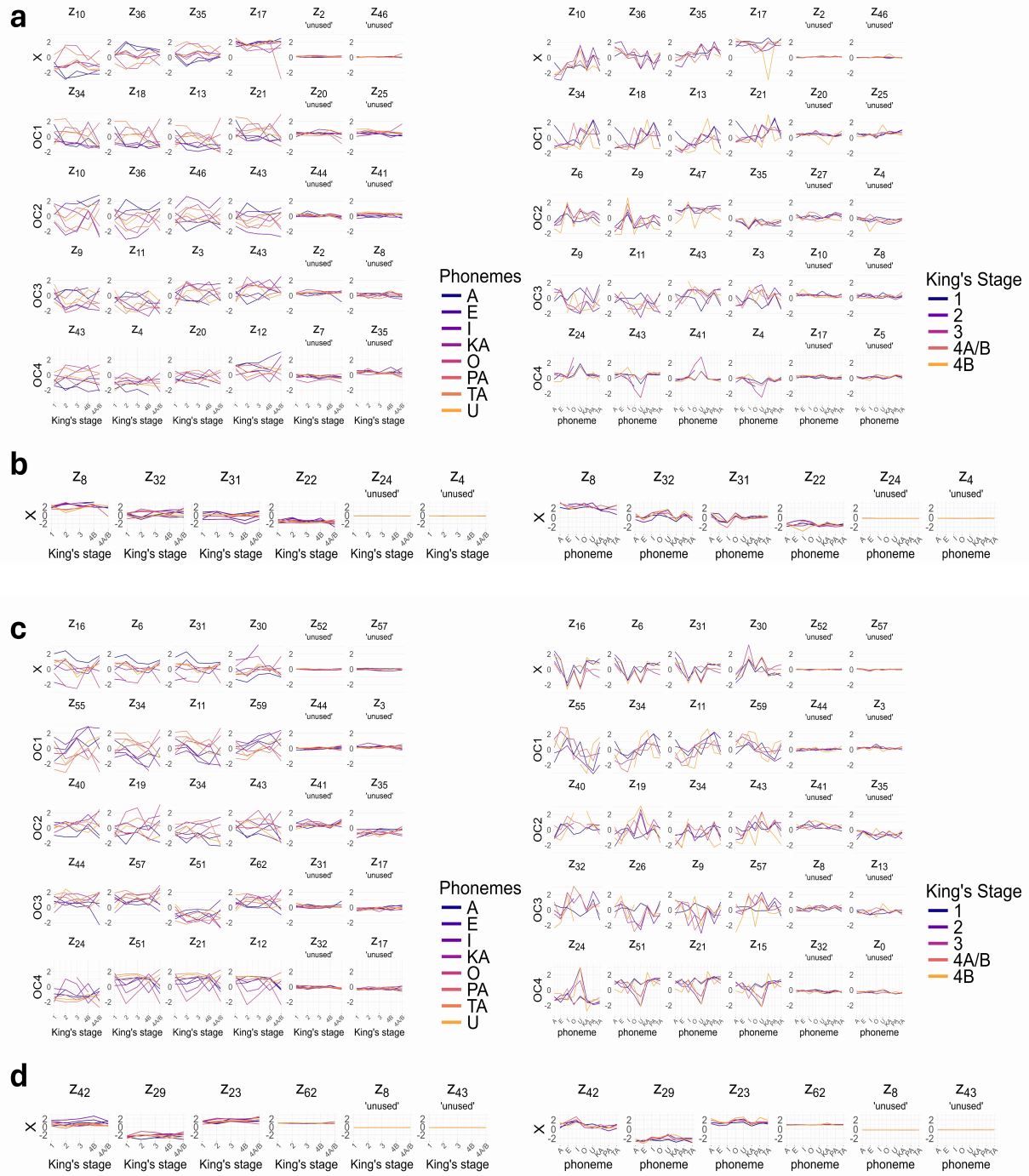
Supplementary Figure 17: Disentanglement evaluation results on IEMOCAP [22] (upper) and VOC-ALS [21] (lower) datasets for DecVAE and supporting benchmarks. Transferred model information is given in parenthesis wherever applicable.



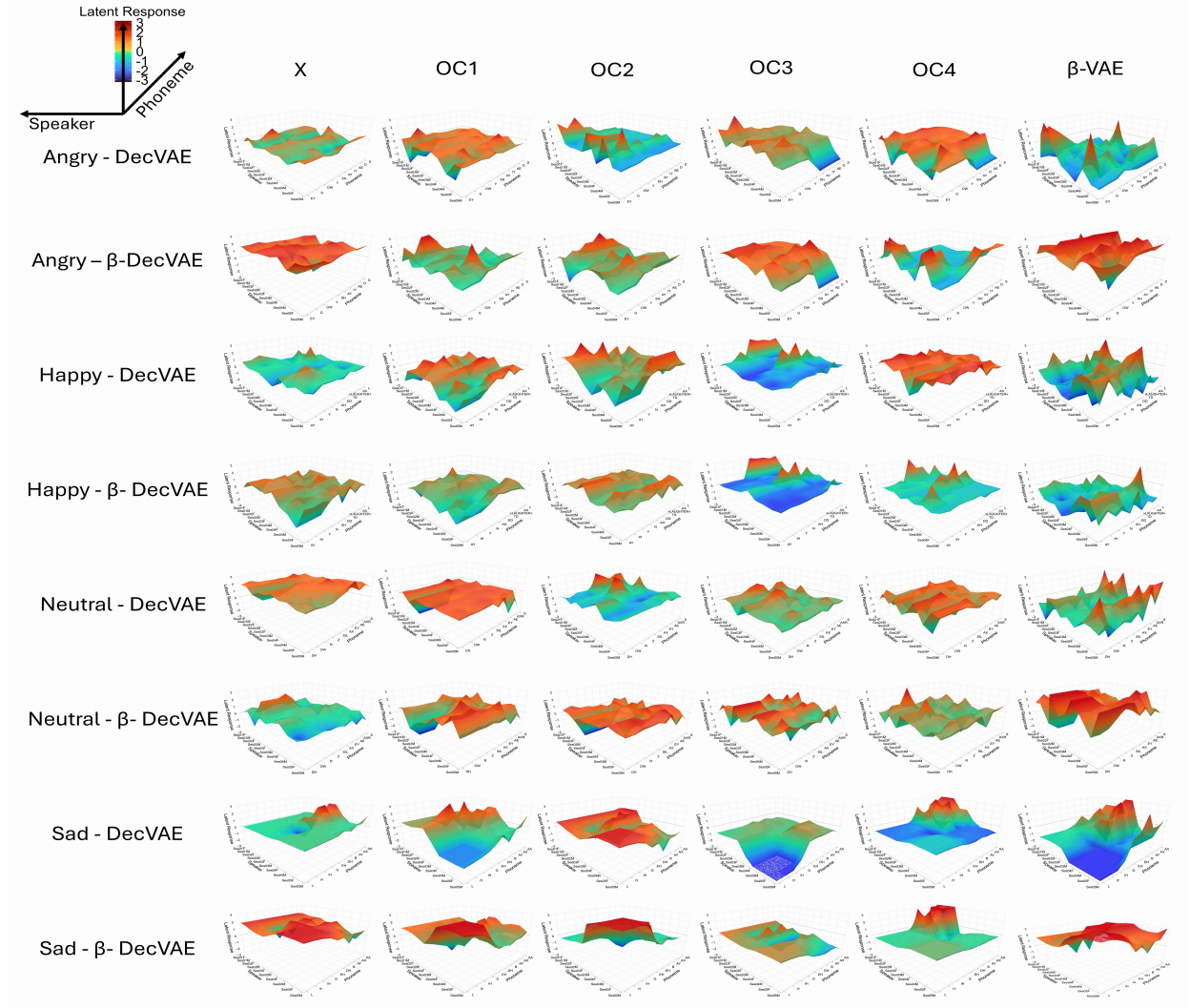
Supplementary Figure 18: Comparison of DecVAEs ($\beta = 0$) with four decomposition methods (FD, EWT[15], VMD[14], EMD[13]) and VAEs [25, 6] in the latent response and interpretability analysis of SimVowels. Speaker values are kept fixed (x-axis) while vowel values are left free to vary (y-axis). We collect responses for each traversal of the vowel variable across all speakers, for the four most variant dimensions (first four columns) and the two less variant dimensions (last two columns). DecVAEs have four latent subspaces $\tilde{Z}$ with number of components $C = 3$, whereas VAEs have a single latent space Z .



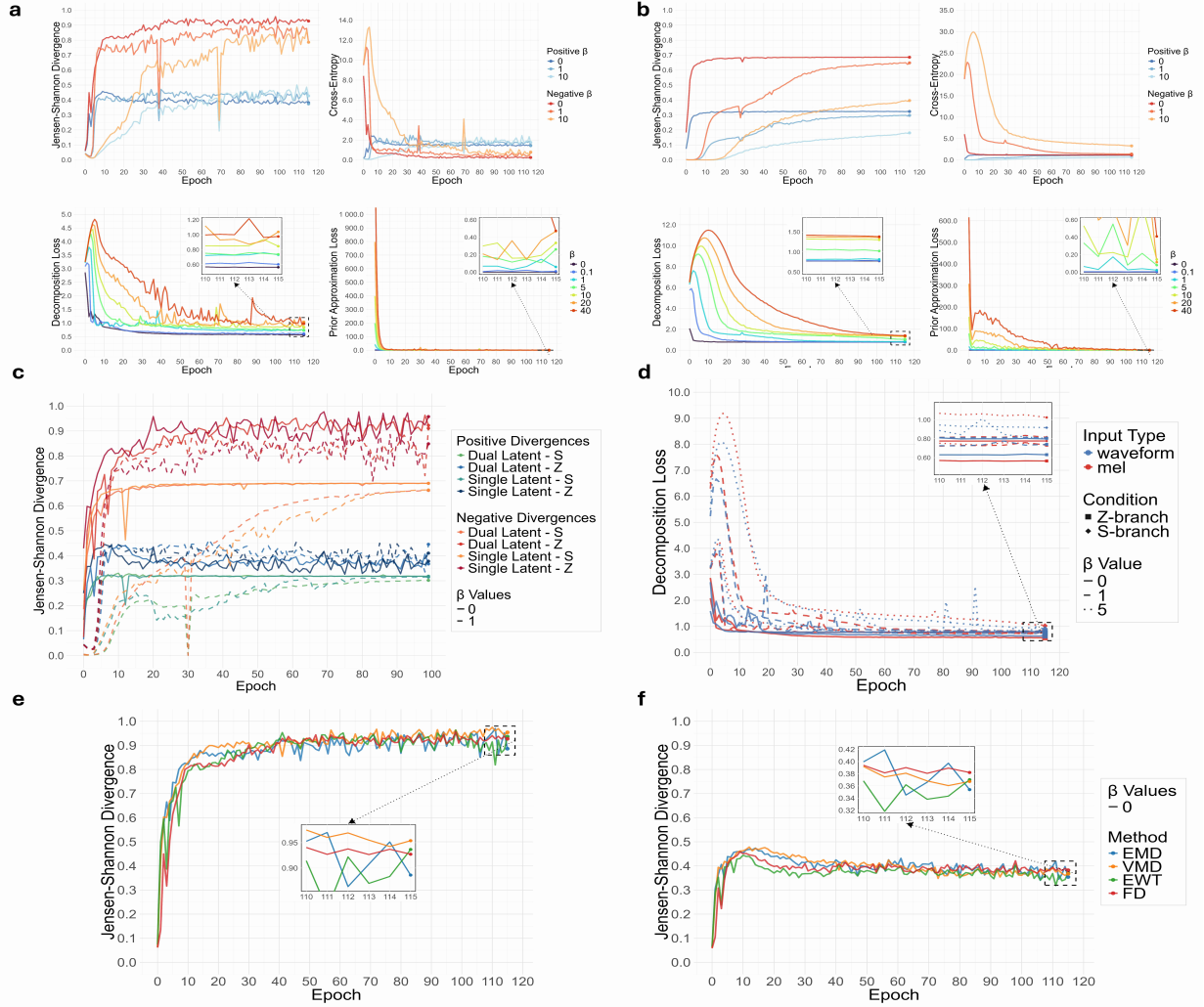
Supplementary Figure 19: Comparison of DecVAEs ($\beta = 1$) using EWT[15] and VAEs [25, 6] in the latent response and interpretability analysis of TIMIT [20]. (a) Phoneme values are kept fixed (x-axis) while speaker values are left free to vary (y-axis). (b) Speaker values are kept fixed (x-axis) while phoneme values are left free to vary (y-axis). We collect responses for each traversal, for the four most variant dimensions (first four columns) and the two less variant dimensions (last two columns). DecVAEs have four latent subspaces $\tilde{Z}$ with number of components $C = 3$, whereas VAEs have a single latent space Z .



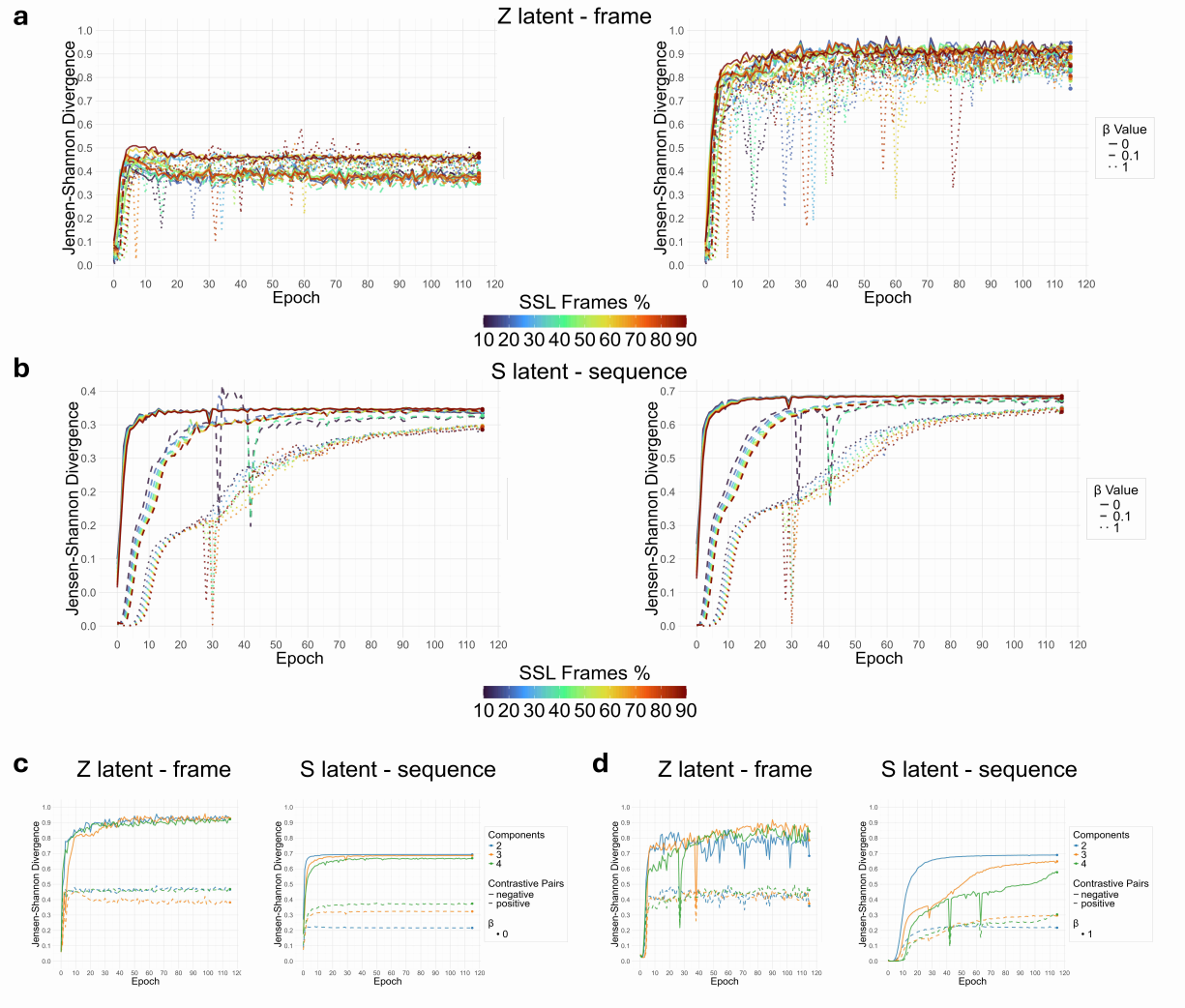
Supplementary Figure 20: Comparison of DecVAEs ($\beta = 0.1$) using EWT[15] and β -VAE [25, 6] in the zero-shot latent response and interpretability analysis of VOC-ALS [21], with models transferred from SimVowels (a,b) and TIMIT [20] (c,d). (a,b,c,d-left column): King's clinical stage values are kept fixed (x-axis) while phoneme values are left free to vary (y-axis) for SimVowels transfer zero-shot evaluation. (a,b,c,d-right column): Phoneme values are kept fixed (x-axis) while King's clinical stage values are left free to vary (y-axis). We collect responses for each traversal, for the four most variant dimensions (first four columns) and the two less variant dimensions (last two columns). DecVAEs have five latent subspaces $\tilde{Z}$ with number of components $C = 4$, whereas VAEs have a single latent space Z .



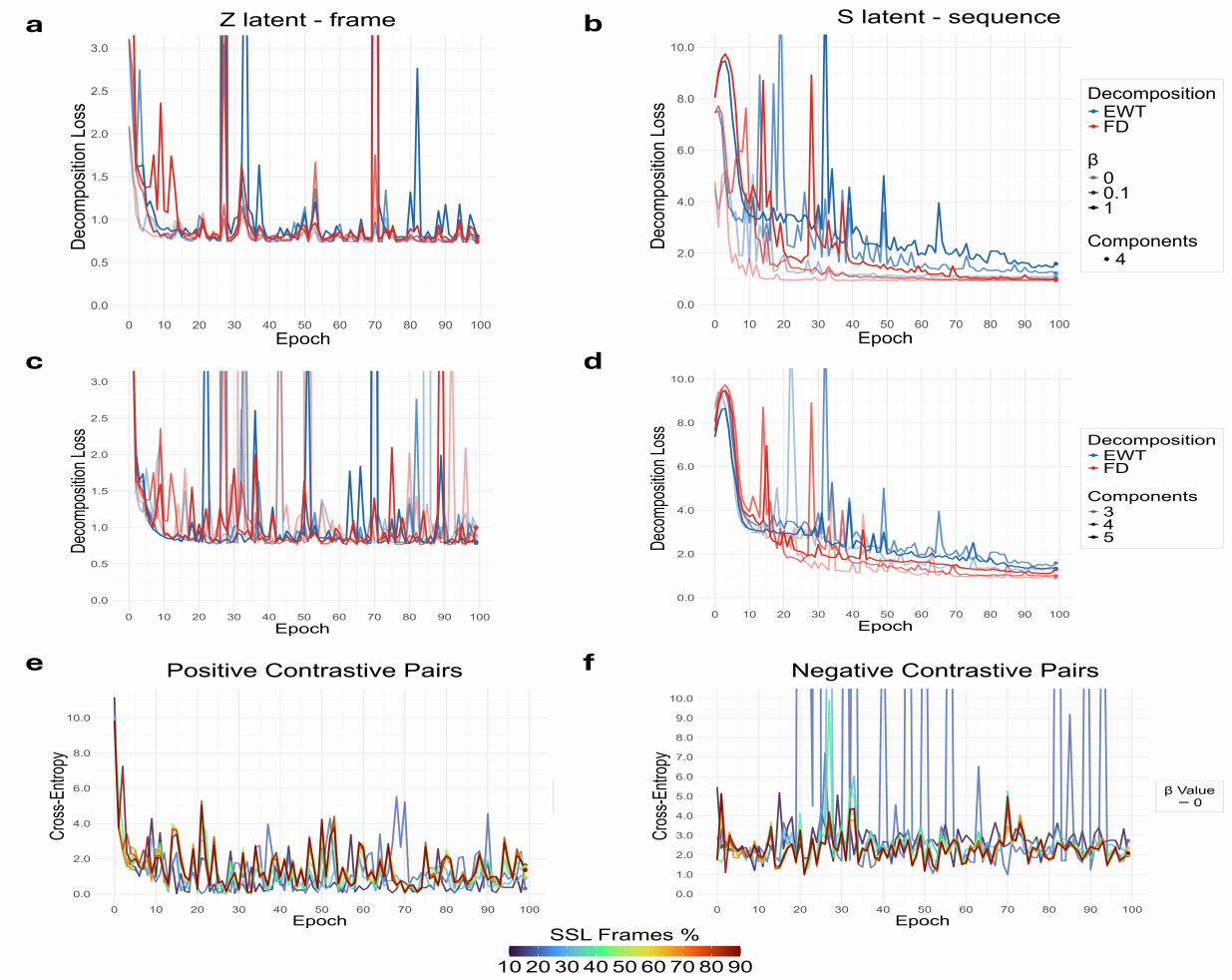
Supplementary Figure 21: Comparison of DecVAEs ($\beta = 1$) using EWT[15] and β -VAE [25, 6] in the latent response and interpretability analysis of IEMOCAP [22], with models transferred from TIMIT [20], for the most informative (highest variance) latent dimension. Latent subspaces $\tilde{Z}$ and β -VAE's Z are displayed in columns, whereas each row denotes a single emotion. By fixing the emotion and freeing the phoneme and speaker variables in the x,y axes, we traverse the learned spaces and obtain the models latent response which we plot as a contour. DecVAEs have five latent subspaces $\tilde{Z}$ with number of components $C = 4$, whereas β -VAE have a single latent space Z .



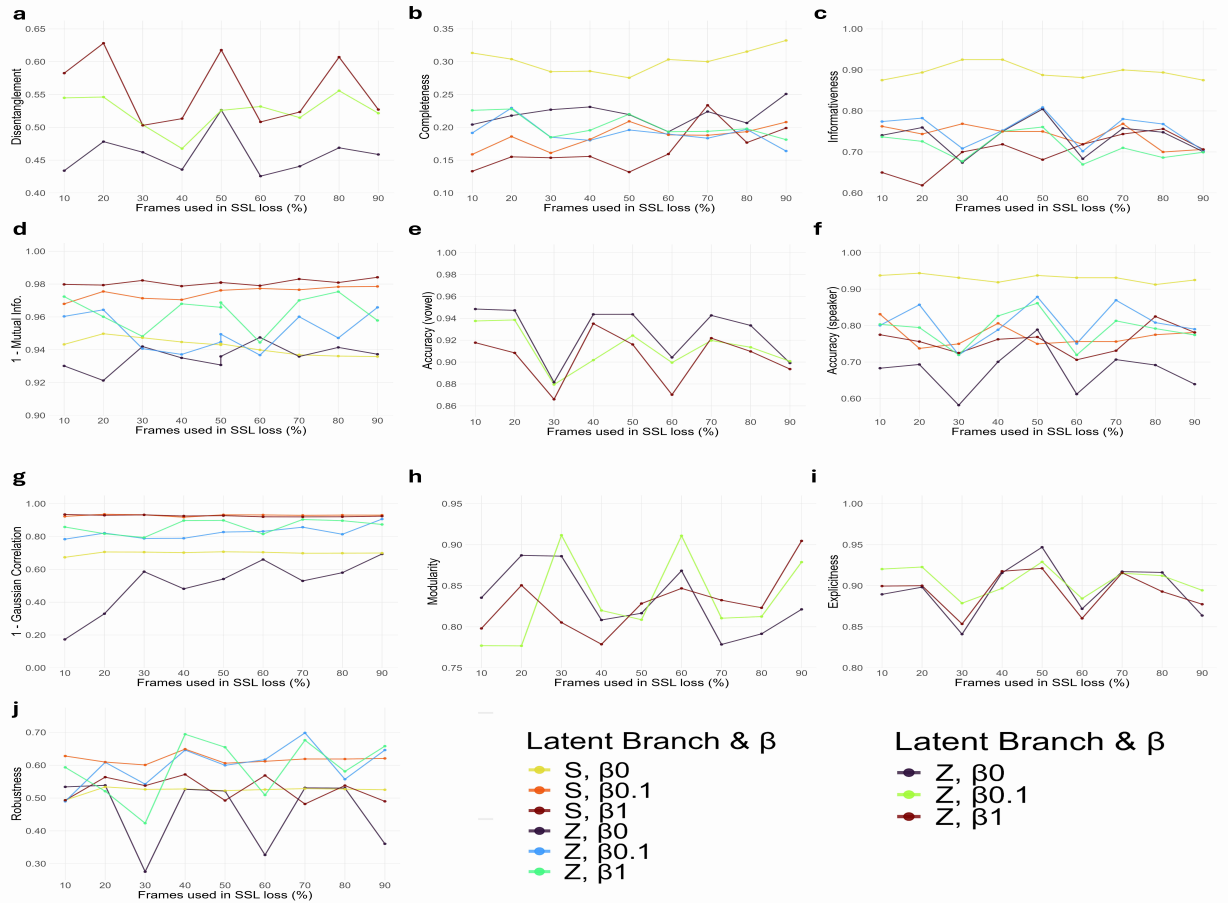
Supplementary Figure 22: Pre-training phase ablations on SimVowels: β , latent branch, input type and decomposition type. (a): frame-level β ablation for positive and negative divergences (upper left), cross-entropies (upper right), $L_{recon} + L_{ortho}$ (lower left), prior approximation loss L_{prior} for various values of β . (b): sentence-level β ablation. (c): comparison of divergences for frame and segment variables, for the dual and single branch cases. (d): $L_{ortho} + L_{recon}$ for input type ablation, raw waveform or mel filterbank. (e,f): negative (e) and positive (f) divergences for four different decomposition methods, $\beta = 0$.



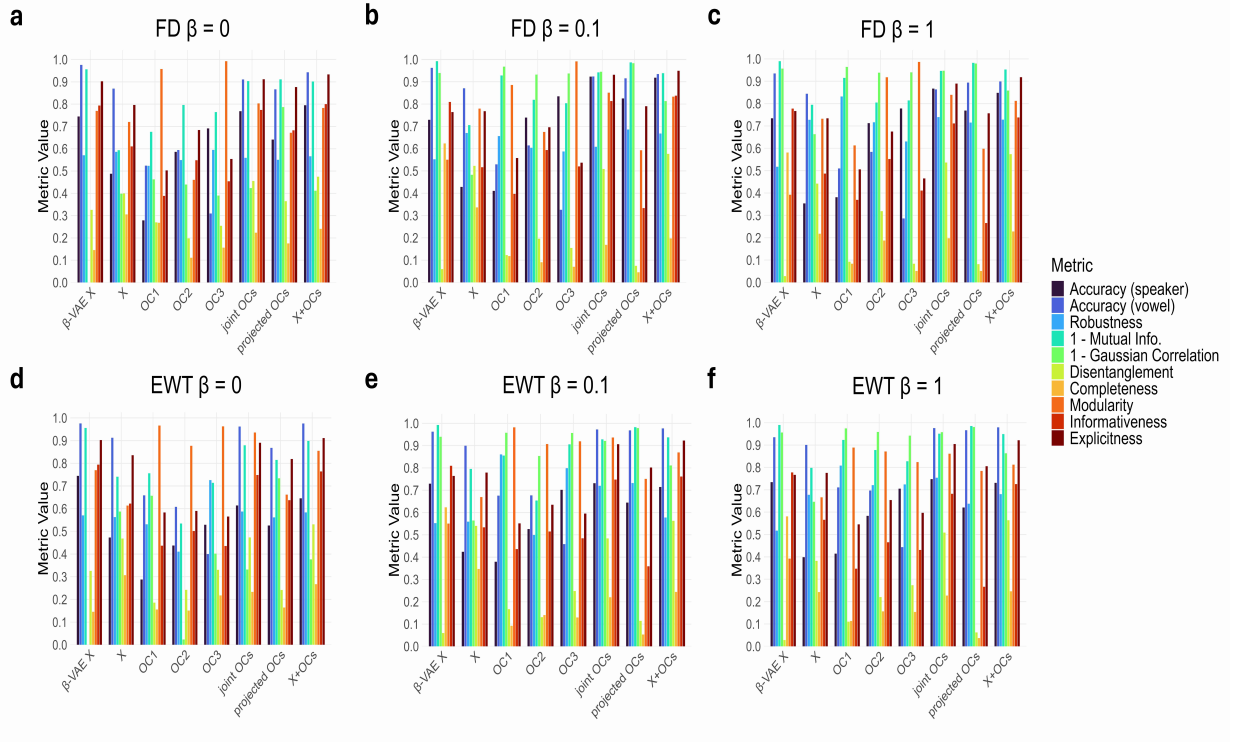
Supplementary Figure 23: Pre-training phase ablations on SimVowels on the percentage of frames used in loss calculations and number of components C . (a): frame variable positive (left) and negative (right) divergences for various frames percentages used in loss calculation. (b): sequence variable positive (left) and negative (right) divergences. (c): positive and negative divergences for the frame (left) and sequence (right) variables for 2,3,4 components, $\beta = 0$. (d): positive and negative divergences for the frame (left) and sequence (right) variables for 2,3,4 components, $\beta = 1$.



Supplementary Figure 24: Pre-training phase ablations on TIMIT [20] on β values, number of components C , and the percentage of frames used in loss calculations. Frame-level (a,c,e) and sequence-level (b,d,f). (a,b): β ablation on decomposition loss $L_{recon} + L_{ortho}$ for FD and EWT [15] for $C = 4$. (c,d): C ablation on decomposition loss $L_{recon} + L_{ortho}$ for FD and EWT[15], $\beta = 1$. (e,f): used frames percentage ablation on decomposition loss $L_{recon} + L_{ortho}$ for FD and EWT[15], $\beta = 0$.



Supplementary Figure 25: Post-training frame percentage in loss ablation on evaluation metrics, SimVowels dataset, for different β values and latent branch. (a): disentanglement [18], (b): completeness [18], (c): informativeness[18], (d): 1 - mutual information, (e): (weighted) vowel recognition accuracy, (f): (weighted) speaker identification accuracy, (g): 1 - gaussian correlation norm. [17], (h): modularity [12], (i): explicitness [12], (j): robustness (IRS) [26]. Frame percentage in SSL loss ablation on metrics. Post-training evaluation of SimVowels.



Supplementary Figure 26: Latent subspace aggregation function $g(\cdot)$ ablation on SimVowels metrics, for FD (a,b,c) and EWT (d,e,f) as the decomposition model $\mathcal{D}_w^C$ and different β values, for the frame variable Z . We compare different options of aggregation functions including the β -VAE embedding: X , individual components ($OC1, OC2, OC3$), $g(\cdot) = \text{concatenate}(\cdot)$, $g(\cdot) = f_{FC_{project}}(\cdot)$, where FC a learned fully-connected projector.

References

- [1] Boulianne G. A Study of Inductive Biases for Unsupervised Speech Representation Learning. *IEEE/ACM Transactions on Audio Speech and Language Processing*, **28**:2781–2795, (2020).
- [2] Menéndez M.L., Pardo J.A., Pardo L., and Pardo M.C. The Jensen-Shannon divergence. *Journal of the Franklin Institute*, **334**(2):307–318, (1997).
- [3] Kingma D.P. and Ba J.L. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*, (ICLR, 2015).
- [4] Kim H. and Mnih A. Disentangling by factorising. *International Conference on Machine Learning*, **80**:2649–2658, (PMLR, 2018).
- [5] Mathieu E., Rainforth T., Siddharth N., and Teh Y.W. Disentangling Disentanglement in Variational Autoencoders. *International Conference on Machine Learning*, pages 7744–7754, (PMLR, 2019).
- [6] Higgins I. et al. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *International conference on learning representations*, (ICLR, 2017).
- [7] Greenacre M., Groenen P.J.F., Hastie T., D’Enza A.I., Markos A., and Tuzhilina E. Principal component analysis. *Nature Reviews Methods Primers*, **2**:1–21, (2022).
- [8] Schölkopf B., Smola A., and Müller K.R. Kernel principal component analysis. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, **1327**:583–588, (1997).
- [9] A. Hyvärinen, Hurri J., and Hoyer P.O. Independent component analysis. In *Natural Image Statistics: A Probabilistic Approach to Early Computational Vision*, pages 151–175. Springer, (2001).
- [10] Rolinek M., Zietlow D., and Martius G. Variational Autoencoders Pursue PCA Directions (by Accident). *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 12398–12407, (2018).
- [11] Ridgeway K. A Survey of Inductive Biases for Factorial Representation-Learning. *Preprint at <https://arxiv.org/pdf/1612.05299>*, (2016).
- [12] Ridgeway K. and Mozer M.C. Learning Deep Disentangled Embeddings With the F-Statistic Loss. *Advances in Neural Information Processing Systems*, **31**, (2018).
- [13] Huang N.E. et al. The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences*, **454**(1971):903–995, (1998).
- [14] Dragomiretskiy K. and Zosso D. Variational mode decomposition. *IEEE Transactions on Signal Processing*, **62**(3):531–544, (2014).
- [15] Gilles J. Empirical wavelet transform. *IEEE Transactions on Signal Processing*, **61**(16):3999–4010, (2013).

- [16] Carbonneau M.A., Zaidi J., Boilard J., and Gagnon G. Measuring Disentanglement: A Review of Metrics. *IEEE Transactions on Neural Networks and Learning Systems*, **35**:8747–8761, (2022).
- [17] Locatello F., Bauer S., Lucic M., Raetsch G., Gelly S., Schölkopf B., and Bachem O. Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations. *Proceedings of the 36th International Conference on Machine Learning*, **97**:4114–4124, (PMLR,2019).
- [18] Eastwood C. and Williams C.K.I. A Framework for the Quantitative Evaluation of Disentangled Representations. In *International Conference on Learning Representations*, (ICLR, 2018).
- [19] Munkres J. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, **5**:32–38, (1957).
- [20] Garofolo J.S. et al. Timit acoustic-phonetic continuous speech corpus. *LDC93S1 Philadelphia: Linguistic Data Consortium*, (1993).
- [21] Dubbioso R. et al. Voice signals database of ALS patients with different dysarthria severity and healthy controls. *Scientific Data*, **11**:1–14, (2024).
- [22] Busso C. et al. IEMOCAP: Interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, **42**:335–359, (2008).
- [23] Baevski A., Zhou Y., Mohamed A., and Auli M. wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. *Advances in Neural Information Processing Systems*, **33**:12449–12460, (2020).
- [24] van der Maaten L. and Hinton G. Visualizing data using t-sne. *Journal of machine learning research*, **9**:2579–2605, (2008).
- [25] Kingma D. P. and Welling M. Auto-Encoding Variational Bayes. In *International Conference on Learning Representations*, (ICLR,2014).
- [26] Suter R., Miladinovic D., Schölkopf B., and Bauer S. Robustly Disentangled Causal Mechanisms: Validating Deep Representations for Interventional Robustness. *Proceedings of the 36th International Conference on Machine Learning*, **97**:6056–6065, (PMLR,2019).