Information Retention via Learning Supplemental Features

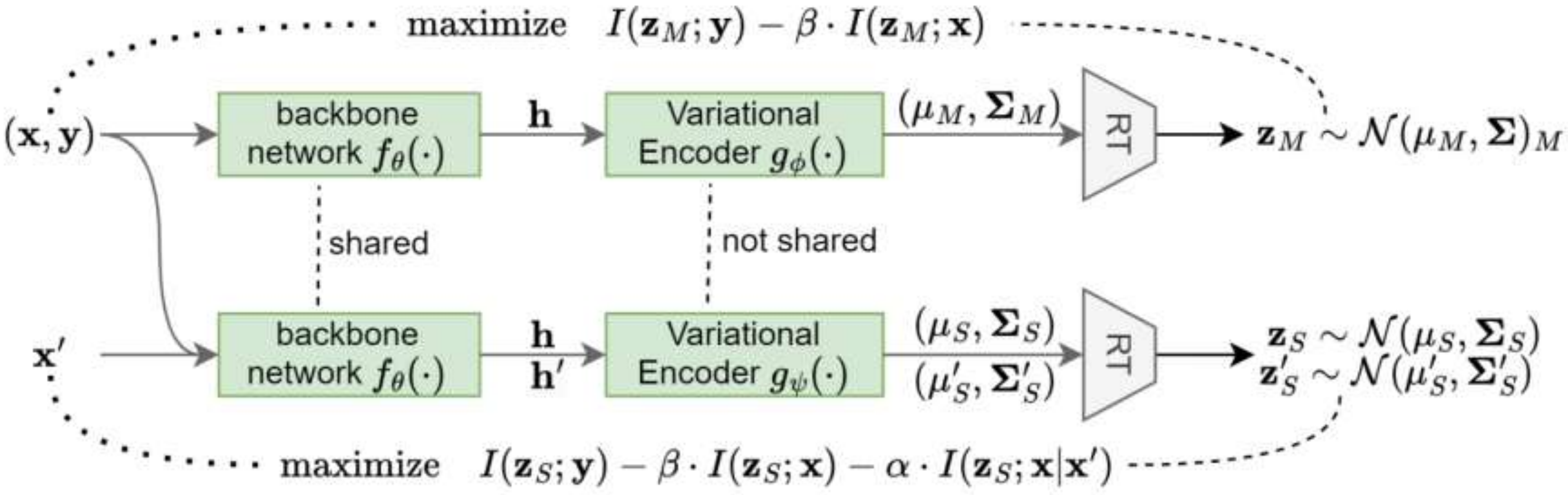
Zhipeng Xie and Yahe Li

School of Computer Science, Fudan University

ICLR, May 2024

Overview of Contributions

- We propose the information retention principle that favors using as much relevant information as possible in supervised learning
- We develop a three-stage framework named InfoR-LSF for information retention via learning supplemental features



Motivation

In contrast to the information bottleneck(IB) principle that ignores as many details of the input, we propose ***Information Retention***: it is preferable to keep as much relevant information as possible in use when making predictions.

Information Bottleneck

- suppress relevant but redundant features

Information Retention

- keep as much relevant information as possible

Motivation

We use a simple example to illustrate the motivation.

- For training, the label y can be perfectly predicted by using the feature $f_1 = x_1 + x_2$, partially predicted by $f_2 = x_3$ and $f_3 = x_4$.
- However, taking f_2 or f_3 into consideration will not bring any lifting in predictive ability.
- For a test data $[x_1 = 1, x_2 = 3, x_3 = 1, x_4 = 2]$, $f_1 = 4$ is unseen, however, f_2 and f_3 can deal with this situation.

x_1	x_2	x_3	x_4	y
1	1	1	2	True
1	1	1	2	True
0	2	2	2	True
0	3	2	2	False
1	2	2	1	False
0	3	2	1	False

The Proposed Method - InfoR-LSF

InfoR-LSF contains three stages:

- ➢ The first stage: initial training of mainline features
- ➢ The second stage: saliency erasing from inputs
- ➢ The third stage: joint training of mainline and supplemental features

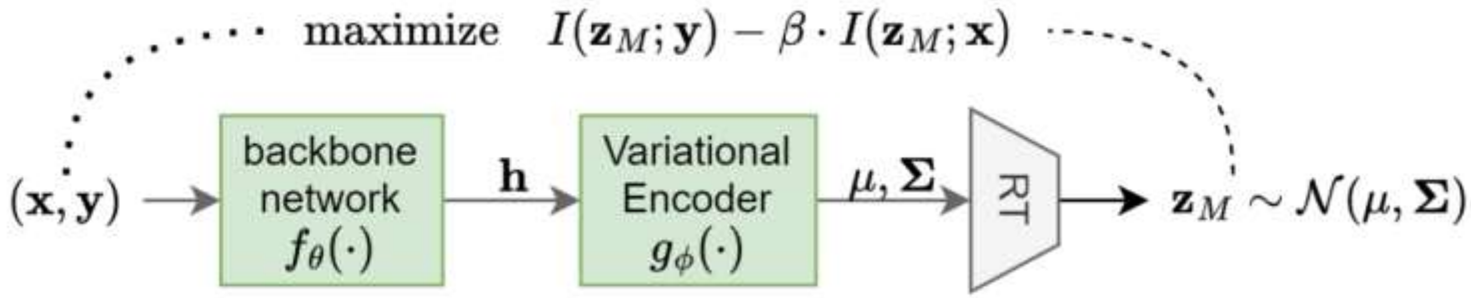
Training of Mainline Features

At the first stage, the task is to train an initial mainline features $\mathbf{z}_M$

Training of Mainline Features

At the first stage, the task is to train an initial mainline features $\mathbf{z}_M$

- Maximize the mutual information between $\mathbf{z}_M$ and the label $\mathbf{y}$
- Minimize the mutual information between $\mathbf{z}_M$ and input $\mathbf{x}$ (the term is optional)



Saliency Erasing

The objective of the second stage is to find and erase salient input features with respect to mainline features $\mathbf{z}_M$ from input $\mathbf{x}$

Saliency Erasing

The objective of the second stage is to find and erase salient input features with respect to mainline features $\mathbf{z}_M$ from input $\mathbf{x}$

- Here, we use the magnitude of the gradient of the loss with respect to the input to determine the importance level of input features.

$$\mathbf{x}_{\mathrm{sf}} = \underset{x \in \mathbf{x}}{\mathrm{topK}} \, ||\nabla_x \mathcal{L}(g_\phi(f_\theta(\mathbf{x})), \mathbf{y})||$$

Saliency Erasing

The objective of the second stage is to find and erase salient input features with respect to mainline features $\mathbf{z}_M$ from input $\mathbf{x}$

- Here, we use the magnitude of the gradient of the loss with respect to the input to determine the importance level of input features.

$$\mathbf{x}_{\text{sf}} = \underset{x \in \mathbf{x}}{\text{topK}} \, ||\nabla_x \mathcal{L}(g_\phi(f_\theta(\mathbf{x})), \mathbf{y})||$$

- the next step is to perform $\text{MASK}(\cdot)$ operation on the raw input $\mathbf{x}$ to get a modified input $\mathbf{x}'$

$$\mathbf{x}' = \text{MASK}(\mathbf{x}) = \mathbf{x}/_{\mathbf{x}_{\text{sf}}}$$

- Replace token with $[\text{MASK}]$ for text data and delete image patches for image data.

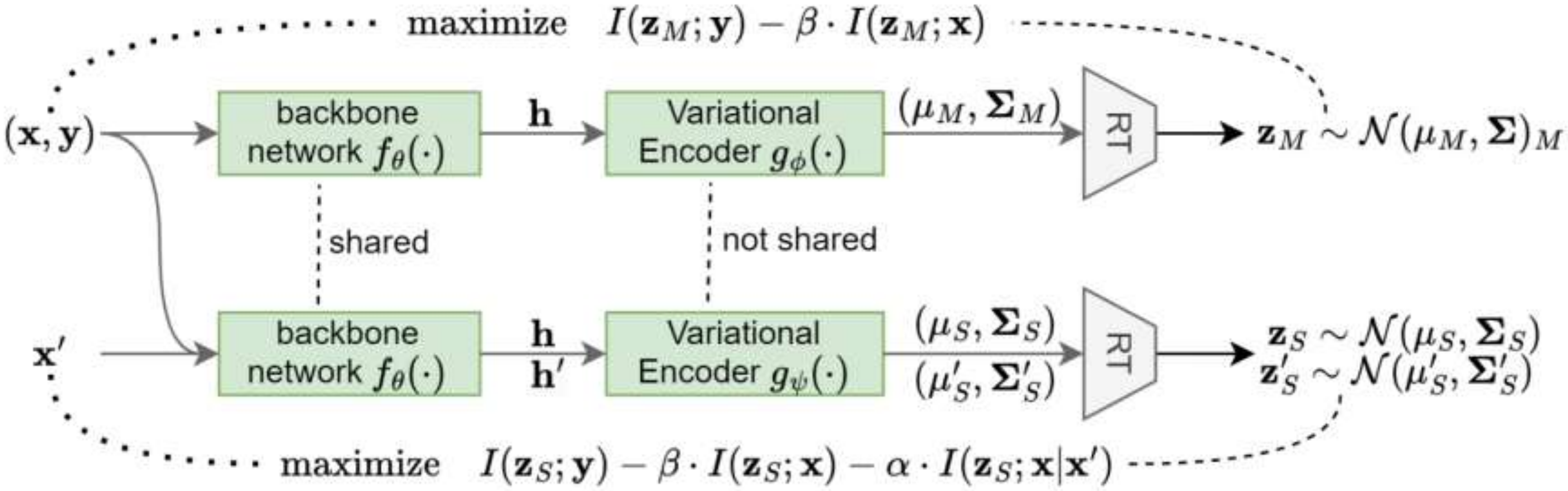
Joint-training of Mainline and Supplemental Features

The objective of the third stage is to simultaneously learn the mainline features $\mathbf{z}_M$ and the supplemental features $\mathbf{z}_S$

Joint-training of Mainline and Supplemental Features

The objective of the third stage is to simultaneously learn the mainline features $\mathbf{z}_M$ and the supplemental features $\mathbf{z}_S$

➢ Overall framework

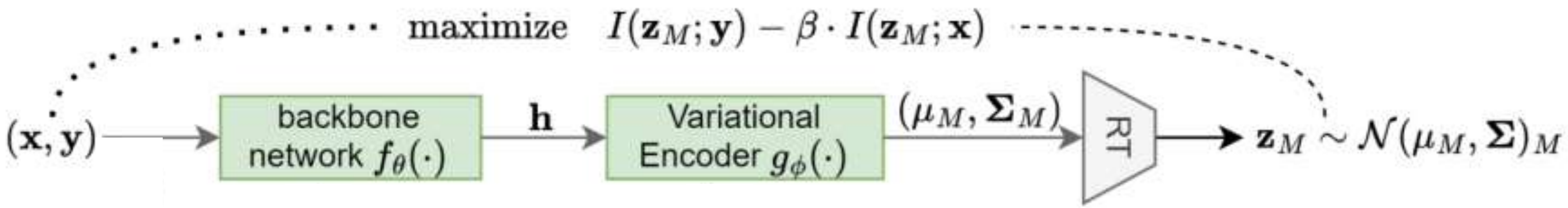


Joint-training of Mainline and Supplemental Features

The objective of the third stage is to simultaneously learn the mainline features $\mathbf{z}_M$ and the supplemental features $\mathbf{z}_S$

- Mainline $\mathbf{z}_M$ training objective: (as same as the first stage)

$$\text{maximize} \quad I(\mathbf{z}_M; \mathbf{y}) - \beta \cdot I(\mathbf{z}_M; \mathbf{x})$$

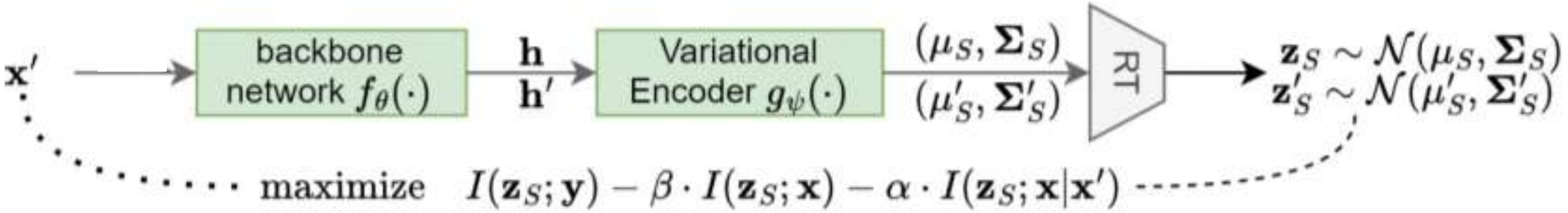


Joint-training of Mainline and Supplemental Features

The objective of the third stage is to simultaneously learn the mainline features $\mathbf{z}_M$ and the supplemental features $\mathbf{z}_S$

➢ Supplemental $\mathbf{z}_S$ training objective:

$$\text{maximize} \quad I(\mathbf{z}_S; \mathbf{y}) - \beta \cdot I(\mathbf{z}_S; \mathbf{x}) - \alpha \cdot I(\mathbf{z}_S; \mathbf{x}|\mathbf{x}')$$



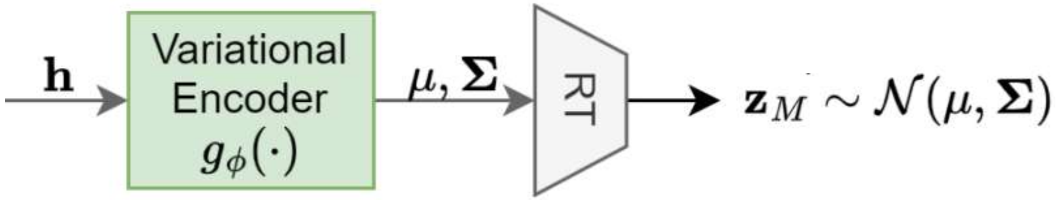
- $I(\mathbf{z}_S; \mathbf{x}|\mathbf{x}')$ represents the information $\mathbf{z}_S$ contains which is unique to $\mathbf{x}$ and is not predictable by observing $\mathbf{x}'$ and we tend to suppress the term

MI-based Loss Function

To compute the aforementioned optimization objective in practice, we employ a variational encoding network to encode $\mathbf{z}_M$ and $\mathbf{z}_S$

MI-based Loss Function

To compute the aforementioned optimization objective in practice, we employ a variational encoding network to encode $\mathbf{z}_M$ and $\mathbf{z}_S$



- $\mathbf{z}$ follows a parameterized Gaussian distribution so we can compute the Kullback-Leibler (KL) divergence of $\mathbf{z}$
- RT means reparameterization trick

MI-based Loss Function

We further estimate the upper and lower bounds of mutual information based on the Gaussian distribution

[1] Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. In ICLR, 2017.

MI-based Loss Function

We further estimate the upper and lower bounds of mutual information based on the Gaussian distribution

- Variational estimate of IB objective[1](maximize $I(\boldsymbol{z}_M;\mathbf{y}) - \beta \cdot I(\mathbf{z}_M;\mathbf{x})$):

$$\mathcal{L}_{\mathrm{VIB}}(\mathbf{x}, \mathbf{z}_M, \theta, \phi) = \mathbb{E}_{\mathbf{x}}[\mathbb{E}_{\mathbf{z}_M \sim p_{\theta,\phi}(\mathbf{z}_M|\mathbf{x})}[-\log q(\mathbf{y}|\mathbf{z}_M)] + \beta \cdot D_{\mathrm{KL}}[p_{\theta,\phi}(\mathbf{z}_M|\mathbf{x})||r_\phi(\mathbf{z}_M)]]$$

- where $r_\phi(\mathbf{z}_M) \sim N(\mu_\phi, \Sigma_\phi)$ is prior distribution of $\mathbf{z}_M$

[1] Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. In ICLR, 2017.

MI-based Loss Function

We further estimate the upper and lower bounds of mutual information based on the Gaussian distribution

- Upper bound of $I(\mathbf{z}_S; \mathbf{x}|\mathbf{x}')$:

$$\mathcal{L}_{\text{IS}} = \mathbb{E}_{\mathbf{x},\mathbf{x}'}[D_{\text{KL}}[p_{\theta,\psi}(\mathbf{z}_S|\mathbf{x})||p_{\theta,\psi}(\mathbf{z}'_S|\mathbf{x}')]]$$

- Note that the modified inputs $\mathbf{x}'$ are only used for the calculation of above loss term

MI-based Loss Function

We further estimate the upper and lower bounds of mutual information based on the Gaussian distribution

- Total loss of mainline features $\mathbf{z}_M$ and supplemental features $\mathbf{z}_S$:

$$\mathcal{L} = \mathcal{L}_{\text{VIB}}(\mathbf{x}, \mathbf{z}_M, \theta, \phi) + \mathcal{L}_{\text{VIB}}(\mathbf{x}, \mathbf{z}_S, \theta, \psi) + \alpha \cdot \mathcal{L}_{\text{IS}}$$

Experiments

Benchmarks

➢ Dataset

Dataset	#Lables	Train	Valid	Test
Image Classification				
CIFAR10	10	50K	-	10K
CIFAR100	10	50K	-	10K
Sentiment Classification				
IMDB	2	20K	5K	25K
YELP	5	62.5K	7.8K	8.7K
YELP-2	2	560K	-	38K
SST-2	2	6.9K	0.9K	1.8K
SST-5	5	8.5K	1.1K	2.2K
MR	2	8.7K	-	2K
Amazon-2	2	3600K	-	400k
Amazon-5	5	3000K	-	650K
Semantic Textual Similarity				
STS-B	1	5.8K	1.5K	1.4K
Regression				
Appliance Energy Prediction	1	15.8K	-	3.9K

➢ Baselines

- **IFM** a method which avoids shortcut solutions by implicit feature modification
- **FGSM** a classic adversarial training method in computer vision
- **VIB** a variational approximation to the information bottleneck by leveraging the reparameterization trick
- **VIBERT** a method implementing the variational information bottleneck on the pretrained BERT

In-domain Generalization on Supervised Tasks

We conduct experiments on both image and text classification tasks, as well as text regression and tabular regression.

In-domain Generalization on Supervised Tasks

We conduct experiments on both image and text classification tasks, as well as text regression and tabular regression.

- InfoR-LSF surpasses all competitors under all settings of training data sizes on image classification tasks.

- InfoR-LSF exhibits much notable improvements in low resource conditions

Table 1: CIFAR10 classification task accuracy under different train data size.

Model	Train Data Size							
	50	**100**	**200**	**500**	**1000**	**2000**	**3000**	**50000**
ResNet-18	17.2	22.6	31.1	40.4	48.9	63.3	74.2	95.1
IFM	17.1	22.4	31.5	42.1	51.8	65.8	75.1	94.6
FGSM	20.1	23.7	31.4	40.3	47.7	58.1	65.5	91.8
VIB	18.6	22.4	31.0	39.7	49.9	64.8	74.7	95.1
InfoR-LSF	**20.3**	**24.5**	**32.1**	**42.1**	**52.8**	**67.3**	**76.2**	**95.2**
Δ	+3.1	+1.9	+1.0	+1.7	+3.9	+4.0	+2.0	+0.1

Table 8: CIFAR100 classification task accuracy under different train data size.

Model	Train Data Size						
	1000	**2000**	**3000**	**5000**	**10000**	**20000**	**50000**
ResNet-18	13.90	20.65	27.10	38.08	55.52	67.14	77.85
IFM	14.04	21.71	28.46	39.46	56.72	67.19	77.53
FGSM	14.19	20.56	26.21	34.80	48.46	59.60	71.66
VIB	13.94	21.17	27.85	39.46	56.30	67.30	77.54
InfoR-LSF	**15.51**	**22.61**	**30.43**	**43.32**	**58.79**	**68.85**	**78.44**
Δ	+1.61	+1.96	+3.33	+5.24	+3.27	+1.71	+0.59

In-domain Generalization on Supervised Tasks

We conduct experiments on both image and text classification tasks, as well as text regression and tabular regression.

- InfoR-LSF also works for text classification tasks.

Table 2: Text classification task accuracy under different train data size.

Dataset	Model	Train Data Size				
		50	**100**	**200**	**500**	**1000**
IMDB	BERT	66.6 (2.2)	77.9 (2.3)	85.6 (0.5)	87.1 (0.6)	88.7 (0.3)
	IFM	66.1 (2.2)	78.2 (2.4)	85.6 (0.7)	87.4 (0.7)	88.7 (0.4)
	VIBERT	68.9 (2.5)	80.8 (1.7)	86.1 (0.6)	87.8 (0.7)	88.8 (0.4)
	InfoR-LSF	**75.5 (2.3)**	**83.0 (2.9)**	**86.9 (0.4)**	**88.3 (0.5)**	**89.4 (0.4)**
	Δ	+8.9	+5.1	+1.3	+1.2	+0.7
YELP	BERT	35.1 (1.8)	39.6 (2.1)	43.1 (1.7)	51.9 (0.9)	55.6 (0.7)
	IFM	35.7 (2.5)	40.1 (1.8)	43.4 (1.0)	50.9 (1.0)	55.5 (0.7)
	VIBERT	37.7 (1.2)	40.8 (2.3)	44.8 (2.2)	53.1 (2.2)	55.4 (0.6)
	InfoR-LSF	**39.6 (1.1)**	**41.4 (1.4)**	**44.9 (2.4)**	**53.6 (0.6)**	**55.9 (0.3)**
	Δ	+4.5	+1.8	+1.8	+1.7	+0.3

In-domain Generalization on Supervised Tasks

We conduct experiments on both image and text classification tasks, as well as text regression and tabular regression.

- InfoR-LSF can also be applied to regression tasks.

Table 3: STS-B test set Pearson correlation coefficient under different train data sizes.

Dataset	Model	Train Data Size				
		50	100	200	500	1000
STS-B	BERT	72.2 (3.2)	79.1 (1.9)	83.8 (0.8)	86.4 (1.0)	87.5 (0.2)
	IFM	72.3 (3.1)	79.2 (1.9)	84.0 (0.9)	86.8 (0.7)	87.6 (0.2)
	VIBERT	74.4 (2.8)	81.9 (1.8)	85.0 (0.4)	87.1 (0.3)	88.4 (0.3)
	InfoR-LSF	**75.0 (3.1)**	**82.4 (2.0)**	**85.4 (0.5)**	**87.5 (0.6)**	**88.7 (0.3)**
	Δ	+2.8	+3.3	+1.6	+1.1	+1.2

Table 4: Coefficient of determination(R^2) of AEP under different train data sizes.

Model	Train Data Size			
	10%	20%	50%	100%
MLP	0.338	0.456	0.597	0.684
IFM	0.373	0.469	0.605	0.680
VIB	0.347	0.471	0.602	0.679
InfoR-LSF	**0.376**	**0.483**	**0.618**	**0.691**
Δ	+0.038	+0.027	+0.021	+0.007

Out-of-domain Performance

We conduct experiments on text classification tasks to evaluate out-of-domain performance of InfoR-LSF

Table 5: Test accuracy of models transferring to new target datasets. All models are trained on YELP and evaluated linear readout on the target datasets. **Δ** are the absolute differences with BERT.

Model	Target Dataset							
	YELP	YELP-2	IMDB	SST-2	SST-5	MR	Amazon-2	Amazon-5
BERT	65.81	94.95	88.24	86.54	44.88	80.70	81.59	54.53
VIBERT	66.00	95.87	88.05	83.90	44.75	81.20	81.81	56.05
InfoR-LSF	**66.31**	**95.89**	**88.55**	**88.19**	**46.28**	**82.00**	**83.03**	**57.43**
Δ	+0.5	+0.94	+0.31	+1.65	+1.4	+1.3	+1.44	+2.9

- On all target tasks, InfoR-LSF consistently achieves the highest improvement

Conclusion

- We introduce the principle of information retention.
- We design a three-stage supervised learning framework named InfoR-LSF for information retention by jointly learning the mainline and supplemental features.
- InfoR-LSF performs well on tasks involving multiple different data types, including both classification and regression.