

1000 **A Description of benchmarks in AIM 1**

1001 The benchmarks examples used in AIM 1 of the paper are shown in this section along with the models  
 1002 extracted by *GenEval*. All coefficients are rounded up to three decimal points after most significant  
 1003 digit.

1004 **Lotka Volterra**: It has two variables  $x_1$  and  $x_2$  given by the following equations:

$$\dot{x}_1 = ax_1 - bx_1x_2, \dot{x}_2 = -cx_2 + dx_1x_2 + u$$

1005  $a = 0.5, b = 0.025, c = 0.5, \text{ and } d = 0.005$

1006 **Recovered model with all variables measured:**

$$\dot{x}_1 = 0.52x_1 - 0.026x_1x_2, \dot{x}_2 = -0.501x_2 + 0.005x_1x_2 + 0.999u$$

1007 **Chaotic Lorenz**: The chaotic lorenz system is described in the following equations:

$$\dot{x}_1 = \sigma(x_2 - x_1) + u, \dot{x}_2 = x_1(\rho - x_3) - x_2, \dot{x}_3 = x_1x_2 - \beta x_3,$$

1008  $\sigma = 10, \beta = 8/3, \rho = 28.$

1009 **Recovered model with all variables measured:**

$$\dot{x}_1 = 10.000(x_2 - x_1) + 0.999u, \dot{x}_2 = 27.992x_1 - 1.002x_1x_3 - 0.998x_2, \dot{x}_3 = 1.000x_1x_2 - 2.7x_3$$

1010 **F8 Cruiser**: The F8 Cruiser system is given by:

$$\begin{aligned} \dot{x}_1 &= -0.9x_1 + x_3 - 0.09x_1x_3 + 0.47x_1^2 - 0.02x_2^2 - x_1^2x_3 + 3.85x_1^3 - 0.21u + 0.28x_1^2u + 0.47x_1u^2 + 0.6u^3 \\ \dot{x}_2 &= x_3, \dot{x}_3 = -4.208x_1 - 0.396x_3 - 0.47x_1^2 - 3.564x_1^3 - 20.967u + 6.265x_1^2u + 46x_1u^2 + 61.1u^3 \end{aligned}$$

1011 **Recovered model with all variables measured:**

$$\begin{aligned} \dot{x}_1 &= -0.872x_1 + 0.998x_3 - 0.088x_1x_3 + 0.476x_1^2 - 0.0186x_2^2 - 0.970x_1^2x_3 \\ &\quad + 3.849x_1^3 - 0.22u + 0.265x_1^2u + 0.472x_1u^2 + 0.63u^3, \dot{x}_2 = 1.000x_3 \\ \dot{x}_3 &= -4.210x_1 - 0.399x_3 - 0.465x_1^2 - 3.565x_1^3 - 20.978u + 6.267x_1^2u + 45.711x_1u^2 + 62.002u^3 \end{aligned}$$

1012 **Pathogenics attack model**: The pathogenic attack system is given by:

$$\begin{aligned} \dot{x}_1 &= \lambda - dx_1 - \beta(1 - \eta u)x_1x_2, \dot{x}_2 = \beta(1 - \eta u)x_1x_2 - ax_2 - p_1x_4x_2 - p_2x_5x_2 \\ \dot{x}_3 &= c_2x_1x_2x_3 - c_2qx_2x_3 - b_2x_3, \dot{x}_4 = c_1x_2x_4 - b_1x_4, \dot{x}_5 = c_2qx_2x_3 - hx_5, \end{aligned}$$

1013 with  $\lambda = 1, d = 0.1, \beta = 1, a = 0.2, p_1 = 1, p_2 = 1, c_1 = 0.03, c_2 = 0.06, b_1 = 0.1,$   
 1014  $b_2 = 0.01, q = 0.5, h = 0.1, \text{ and } \eta = 0.9799.$

1015 **Recovered model with all variables measured:**

$$\begin{aligned} \dot{x}_1 &= 0.939 - 0.1x_1 - 0.982x_1x_2 + 0.98u x_1x_2, \dot{x}_2 = 0.982x_1x_2 - 0.98u x_1x_2 - 0.18x_2 - 1x_4x_2 - 1.001x_5x_2 \\ \dot{x}_3 &= 0.059x_1x_2x_3 - 0.03x_2x_3 - 0.009x_3, \dot{x}_4 = 0.029x_2x_4 - 0.1x_4, \dot{x}_5 = 0.059x_2x_3 - 0.1x_5 \end{aligned}$$

1017 **B Architecture in Figure 2 is universal dynamics approximator**

1018 The forward pass of liquid time constant neural network (LTC-NN) is given by [27]:

$$\frac{dg(t)}{dt} = -g(t)/\tau + f_{NN}(g(t), I(t), t, \omega)(A - g(t)), \quad (12)$$

1019 where  $g(t)$  is one hidden state of the LTC-NN,  $\tau$  is a time constant parameter, required to assist any  
 1020 autonomous system to reach equilibrium state. As such existence of the  $-g(t)/\tau$  is an important  
 1021 stability criteria as it ensures that the unperturbed plant settles in time.  $f_{NN}$  is the forward pass and  
 1022 is a function of the hidden states,  $I(t)$  is the input to the LTC-NN,  $\omega$  and  $A$  are the parameters of the  
 1023 LTC-NN architecture.

1024 **Theorem 3** *The forward pass of an LTC-NN architecture generates a set of implicit physical dynam-*  
 1025 *ics that are equivalent to a bilinear approximations of the control affine autonomous system in Eqn.*  
 1026 *8.*

1027 **Proof:** Algebraic manipulation of the forward pass of LTC-NN architecture gives the structure of  
 1028 Eqn. 13 which allows an input dependent time constant  $\frac{\tau}{1+\tau f_{NN}(g(t), I(t), t, \omega)}$ .

$$\frac{dg(t)}{dt} = -\frac{g(t)}{\frac{\tau}{1+\tau f_{NN}(g(t), I(t), t, \omega)}} + f_{NN}(g(t), I(t), t, \omega)(A). \quad (13)$$

1029 The stability criteria for any autonomous system requires the control affine model to have a time  
 1030 constant term as shown in Eqn. 14

$$\frac{dX}{dt} = -X/\tau + \mathcal{G}_{f-\tau}(X) + \mathcal{G}_u(X)U_T, \quad (14)$$

1031 where  $\tau$  is the time constant of the system and  $\mathcal{G}_{f-\tau}(\cdot)$  is the unperturbed dynamics obtained by  
 1032 removing the time constant component from  $\mathcal{G}_f(\cdot)$ .

1033 Assuming that the autonomous system is a dynamic causal system, the bilinear approximation [21] of  
 1034 the control affine system in Eqn. 14 results in Eqn. 15.

$$\frac{dX}{dt} \approx -X/\tau + \mathcal{G}_{f-\tau}(X) + BX + CU_T + \sum_j u_T^j D^j X + H, \quad (15)$$

1035 where  $B = \frac{\partial(\mathcal{G}_u(X)U_T)}{\partial X}$ ,  $C = \frac{\partial(\mathcal{G}_u(X)U_T)}{\partial U_T}$ , and  $D^j = \frac{\partial^2(\mathcal{G}_u(X)U_T)}{\partial X \partial u_T^j}$ ,  $H$  is a constant. Rearranging  
 1036 Eqn. 15, we have the similar form as the LTC-NN forward pass in Eqn. 16.

$$\frac{dX}{dt} \approx -\frac{X}{\frac{\tau}{1+\tau(B+\sum_j u_T^j D^j)}} + (\mathcal{G}_{f-\tau}(X) + CU_T + H). \quad (16)$$

1037 We observe that Eqn. 16 is the same form as Eqn. 13 if the input to the LTC-NN  $I(t)$  is a concatenation  
 1038 of  $Y$  and  $U_T$ . The hidden layers of the LTC-NN model an inflated set of implicit dynamics which  
 1039 may include the unmeasured system variables of the physics model.

1040 **Theorem 4** *The inflated set of implicit dynamics modeled by LTC-NN induces an over-determined*  
 1041 *set of equations in the coefficients of the bilinear approximation of any control affine model.*

1042 **Proof:** The training process of LTC-NN fixes weights and instantiates the hidden layer outputs.  
 1043 The values of the unmeasured variables in  $X$  is estimated by the hidden state in each training step  
 1044 utilizing the forward pass and learned LTC-NN weights  $\omega$ . Hence each forward pass provides an  
 1045 over-determined set of linear equations in the coefficients  $B$ ,  $C$ , and  $D^j$ .

1046 The original control affine model coefficients  $\Theta$  are non-linear functions of the coefficients  $B$ ,  $C$ , and  
 1047  $D^j$ s. The dense layer is best suited for exploring a large set of possible non-linear combinations of  
 1048  $B$ ,  $C$ , and  $D^j$  that express  $\Theta$ . An overdetermined system of equations is inconsistent and may be  
 1049 unsolvable. The dense layer guided by the ODE solver induced loss function (ODE Loss) learns a  
 1050 consistent set of linear equations in  $B$ ,  $C$ , and  $D^j$  and also learns their non-linear combination to  
 1051 determine  $\Theta$ .

## 1052 C Proof of Theorem 1

1053 For any data point  $X, Y \in D_J$ ,  $Pr(\rho(\mathcal{K}_\Psi(X_i, Y_i), D_I) \in C = [-d, d]) \geq 1 - \alpha$ ,  $\alpha > 0$  if and only  
 1054 if  $Pr(\mathcal{K}_\Psi(X, Y) | \{X, Y\} \in D_J) = Pr(\mathcal{K}_\Psi(X, Y) | \{X, Y\} \in D_I)$ , where  $d$  is given by Algorithm  
 1055 1.

1056 **Proof:** To prove the if part of Theorem 1 lets assume that  $Pr(\rho(\mathcal{K}_\Psi(X_i, Y_i), D_I) \in C = [-d, d]) \geq$   
 1057  $1 - \alpha$ ,  $\alpha = 0.05$ . This means that  $[-d, d]$  is the 95% confidence interval of the robustness value  
 1058  $\rho(\mathcal{K}_\Psi(X_i, Y_i), D_I)$ . Now  $[-d, d]$  is derived from the training data of  $D_I$ . Which means that  $[-d, d]$   
 1059 is also the 95% confidence interval of  $\rho(\mathcal{K}_\Psi(X, Y), D_I)$  for  $\forall X, Y \in D_I$ . This entails that the 95%  
 1060 confidence intervals of  $D_I$  and  $D_J$  are the same. The interval is a zero mean distribution, hence any  
 1061 memory-less stochastic process can uniquely define the interval with only one parameter,  $\alpha$ . Since  $\alpha$   
 1062 only defines the probability density and  $\alpha$  is same for both  $X_i, Y_i \in D_J$  and  $X, Y \in D_I$  we prove  
 1063 that  $Pr(\mathcal{K}_\Psi(X, Y) | \{X, Y\} \in D_J) = Pr(\mathcal{K}_\Psi(X, Y) | \{X, Y\} \in D_I)$ .

1064 Note that the interval was defined by  $\alpha$  where  $\alpha$  can be chosen arbitrarily. This implies that if  $\forall \alpha$  the  
 1065 intervals are same, then even for non memoryless distributions, the two distribution have to be same.

1066 The only if part is simpler since if the distributions are same, the  $1 - \alpha$  confidence intervals will be  
 1067 same by definition.

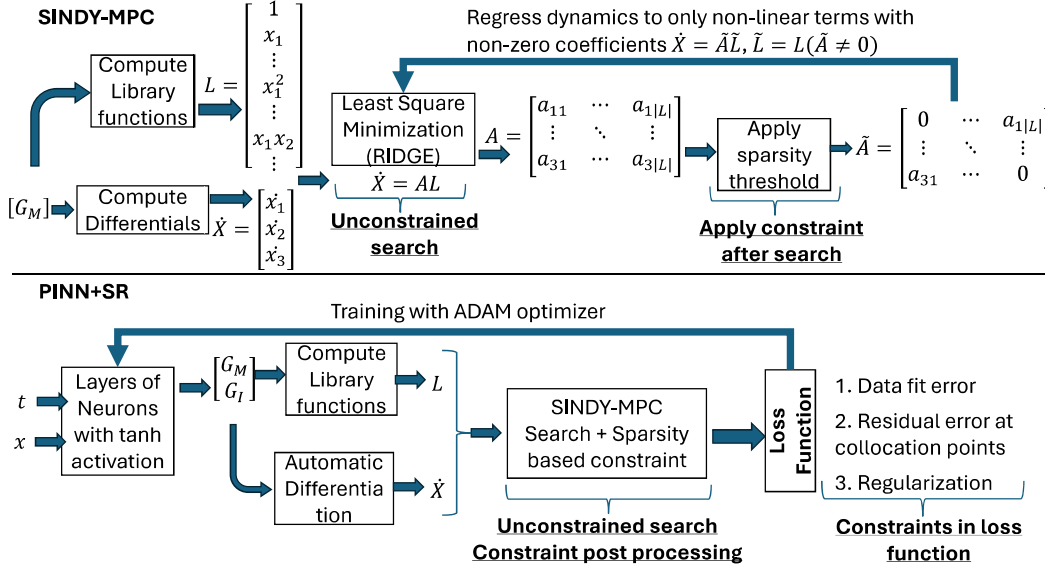


Figure 3: Learning architectures of SINDy-MPC and PINN+SR.

## D Proof of Theorem 2

For any data point  $X, Y \in D_J$ , and hypothesis  $h = f \circ g$  learned on data from source domain  $D_I$ ,  $Pr(\rho(\mathcal{K}_g(X, Y), D_I) \in C = [-d, d]) \geq 1 - \alpha$ ,  $\alpha > 0$  if and only if  $Pr(\mathcal{K}_g(X, Y) | \{X, Y\} \in D_J) = Pr(\mathcal{K}_g(X, Y) | \{X, Y\} \in D_I)$ , where  $d$  is given by Algorithm 1 with  $X$  replaced by  $g(X)$ .

**Proof:** This theorem is only arrived at if the two domains  $D_I$  and  $D_J$  have the same set of causal factors, i.e. Theorem 1 is satisfied. In that case the proof of this theorem is straightforward.

If  $X, Y \in D_I$  and  $X_i, Y_i \in D_J$  are random variables with  $Pr(\mathcal{K}_\Psi(X, Y) | \{X, Y\} \in D_J) = Pr(\mathcal{K}_\Psi(X, Y) | \{X, Y\} \in D_I)$ , then for any measurable function  $g: \mathbb{R}^{|X|} \rightarrow \mathbb{R}^M$ , where  $M$  is the dimension of the representation  $g(X, Y)$  and  $g(X_i, Y_i)$  have the same distribution.  $\mathcal{K}_g(\cdot)$  is the potentially infinite dimensional koopman operator and hence can represent the function  $g(\cdot)$  with arbitrary precision. Hence properties on  $g(\cdot)$  will also satisfy on  $\mathcal{K}_g$ . Hence, as a consequence of measurability and satisfaction of Theorem 1, Theorem 2 is proved.

## E Implementation details of SINDy-MPC and PINN+SR

**SINDy-MPC:** Significant breakthrough was achieved through introduction of sparse identification of non-linear dynamics (SINDy). Subsequently SINDy has been extended to tackle control inputs in SINDy-MPC [33], however, as shown in this manuscript, it does not generalize well for low sampling frequencies. Given  $N$  samples of data, at sampling frequency  $f_r = \frac{1}{\tau}$  the recovery problem can be reduced to solving the following set of linear equations:

$$\begin{bmatrix} \frac{dX}{dt}(\tau) \\ \vdots \\ \frac{dX}{dt}(N\tau) \end{bmatrix} = \begin{bmatrix} \zeta(X(\tau), 0) & \cdots & \zeta(X(\tau), W) \\ \vdots & \ddots & \vdots \\ \zeta(X(N\tau), 0) & \cdots & \zeta(X(N\tau), W) \end{bmatrix} \times \begin{bmatrix} \theta_0 \\ \vdots \\ \theta_W \end{bmatrix}, \quad (17)$$

where  $W = \binom{M+n}{n}$ , and  $\zeta(X(k\tau), i)$  is the  $i^{th}$  term in the bilinear expansion of  $f(X)$  at the time sample  $t = k\tau$ . For sparse identification, majority of the  $\theta_j \approx 0, j \in \{1 \dots W\}$  with only  $p$  significant elements  $\Theta = [\theta_1 \dots \theta_p]$ . As such  $N \gg p$ , making Eqn 17 an over-determined set of linear equations with no consistent solution. A solution method is least squares minimization, to recover  $\Theta_{est}$  that minimizes  $e_T = \|X_{est} - X\|^2$ . SINDy-MPC solves the unconstrained least square minimization problem using the sequential threshold ridge regression (STRidge) algorithm (Figure 3). This method iteratively selects dominant candidate from a library of high dimensional nonlinear functions [53]. The sparsity was achieved through iteratively removing non-linear components utilizing hard thresholds on the derived model coefficients.

1095 **PINN+SR:** As shown in Figure 3, the PINN+SR approach starts with generating a valuation of the  
 1096 multi-variate data  $X$  as a function of time using a feedforward network [17]. The derivatives of  $X$   
 1097 with respect to time are then derived by another layer of nodes with automatic differentiation. The  
 1098 derivatives and valuations of  $X$  are used to evaluate a library of nonlinear functions. These non-linear  
 1099 functions are provided as input to the SINDy method that outputs a sparse matrix using the STRidge  
 1100 method. The sparse matrix is used to re-evaluate the solution of the ODE and compute residual  
 1101 with respect to the ground truth, the physics loss, and regularization loss. The overall loss is used to  
 1102 optimize and train the weights of the entire network using backpropagation.

## 1103 F AIM 2 benchmark details

1104 **F8Cruiser:** This is an aircraft pitch control system using a model predictive control for trajectory  
 1105 tracking. The first domain  $D_1$  is normal pitch control operation. The second domain  $D_2$  is a  
 1106 **hardware failure** where the elevator gets jammed and maintains a constant position overriding the  
 1107 controller (*F8Stuck*). We created another domain *F8Slow* where the elevator speed is reduced by 3  
 1108 times the normal speed.

1109 **Automated Insulin Delivery System** This is an hybrid close loop autonomous system that au-  
 1110 tonomously decides on insulin delivery for the most part, but requires **human intervention** with  
 1111 extra insulin delivery to manage meal intake. The human may trick the system to deliver a high  
 1112 dosage of insulin by announcing to the system that a large meal has been ingested without actually  
 1113 consuming the meal. Domain  $D_1$  is normal operation of the AID. The domain  $D_2$  has data from  
 1114 individuals who injected rescue meal of carbohydrate content of 15 g whenever glucose is less than 15  
 1115 g (*AIDRescue*). The underlying causal factor model of the AID system is governed by the bergman  
 1116 minimal model (BMM) described in Equation 18. We created another domain  $D_3$  with insulin  
 1117 cartridge failure *AIDCartridge*. In cartridge failure, whenever the insulin system needs to inject  
 1118 bolus, it does not inject. Instead it builds up at the cartridge. Finally, when the pressure goes beyond  
 1119 a threshold, a large dosage of insulin rushes through into the blood stream.

$$\delta \dot{i}(t) = -n\delta i(t) + p_4 u_1(t) \quad (18)$$

$$\delta \dot{i}_s(t) = -p_1 \delta i_s(t) + p_2 (\delta i(t) - i_b) \quad (19)$$

$$\delta \dot{G}(t) = -\delta i_s(t) G_b - p_3 (\delta G(t)) + u_2(t)/VoI, \quad (20)$$

1120 The input vector  $U(t)$  consists of the input insulin level  $u_1(t)$  and the glucose appearance rate in the  
 1121 body  $u_2$ . The output vector  $Y(t)$  comprises the blood insulin level  $i$ , the interstitial insulin level  $i_s$ ,  
 1122 and the BG level  $G$ . In AP, only  $G$  is a measurable output.  $i_s$  and  $i$  are un-measurable.  $p_1, p_2, i_b, p_3$ ,  
 1123  $p_4, n$ , and  $1/VoI$  are all patient specific coefficients.

1124 **Unmanned Aerial Vehicle:** This is a quadcopter, whose altitude is controlled by four proportional  
 1125 integrative and derivative (PID) controllers. These controllers provide balanced thrusts in each  
 1126 propeller so that the UAV maintains a given height. The first domain  $D_1$  is normal altitude control  
 1127 operation of the UAV. The second domain  $D_2$  is a **software failure** that changes the gravity parameter  
 1128  $g$  in the controller software (*UAVSimG*). The third domain  $D_3$  is an **electromagnetic attack** on  
 1129 the UAV gyroscope sensor (*UAVEMA*).

1130 **Diabetic Retinopathy:** It is an important image classification application in the medial domain.  
 1131 There have been several works in DR that attempts to classify state 0 i.e. no DR to stage 5 i.e.  
 1132 proliferative DR. We create two domains *DRNormal* for stage 0 DR and *DRStage5* with stage 5  
 1133 DR.

## 1134 G AIM 2 Comprehensive Results

1135 For the time series case studies, in each of the causal support violation case study, we train the  
 1136 baseline technique as an autoencoder on 80% of the normal data. Then we extract the intermediate  
 1137 representation from the 20% normal data and domain  $D_2$  or  $D_3$  data. We then extract an anomaly  
 1138 score from the internal representation and use extreme value theory to determine if the datapoint  
 1139 is normal or from a different domain. We then report the precision recall and F1 score for each  
 1140 technique in Table 6.

1141 For the image domain datasets, the baseline techniques were already configured to classify five classes  
 1142 of DR in [22]. We re-ran the baseline techniques to determine the precision recall and accuracy for  
 1143 Stage 5 DR and Stage 0 DR classification only. This is reported in Table 6. We then extract the  
 1144 DCB% from each domain for each case study with respect to the normal data  $D_1$  for the given case  
 1145 study. This is reported in the DCB row of Table 6. As seen the maximum F1 score obtained in Table  
 1146 6 is highly correlated with the DCB % (correlation coefficient 0.91 p = 0.043). This shows that

Table 6: Comparison of **SPIE-AD** against baseline techniques for U2 benchmark examples (R is real world, S is synthetic). **SPIE-ADS** uses SINDY-MPC, while **SPIE-ADL** uses LTC-NN. <sup>+</sup> denotes with point adjustment (PA) and absence of <sup>+</sup> is without PA.

Approach	F8Stuck S			F8Slow S			UAVSimG S			UAVEMA S			AIDPhantom S			AIDCartridge S			DRStage5		
	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
Omni [56]	41	26.8	32.4	65	28.1	39.2	32	19.7	24.4	29	16.8	21.3	19.1	16.5	17.7	65	31.9	43			
AT [67]	85.5	75.8	80.3	34.2	32.8	33.5	35	33.5	34.2	33.9	32.4	33	34	32	33	34.3	33.8	34			
iForest [42]	14	33	19.6	9.8	8.2	8.9	10.6	8.5	9.4	8.6	7.6	8.1	9.5	8.1	8.7	9.5	7.9	8.6			
LODA [51]	88	70	78	60.7	13.7	22.4	50.7	11	18	35	8.6	13.8	35.8	9.4	14.9	36.4	9.7	15.3			
LSTM [28]	77	85	80	61	35.8	45.2	59.4	13.2	21.6	60.8	14.2	23	58.6	12.6	20.7	54.7	12.1	19.9			
USAD [6]	81	67.7	74	55.3	14.2	22.6	51.2	12.3	19.8	49.2	12.1	19.4	52.6	12.1	19.7	58	8.8	15.2			
GANF [76]	61	79	68.8	3.2	4.3	3.7	51.4	85	64.3	0.9	24.7	1.8	3.2	4.5	3.8	2.1	2.7	2.4			
GAT [80]	71.4	80.5	75.7	58.9	34.5	43.5	59.2	32.3	41.8	50.4	28	36	54.5	28.9	37.8	57.2	30.3	39.7			
OFA [79]	21.4	4.5	7.4	21.9	9.7	13.4	37.5	22.1	27.2	20.3	8.5	12	31.3	18.3	23.1	21.7	10.1	13.8			
FITS [68]	21.4	8.6	12.3	48.1	14.3	22.05	17.3	21.9	19.3	80.4	2.4	4.7	24.5	18.4	21.0	14.7	40.1	21.5			
TFAD [72]	11.2	30.4	16.4	9.8	21.7	13.5	29.5	12.4	17.5	21.9	8.7	12.4	14.7	31.8	19.9	17.7	21.4	19.4			
ERMViT [59]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	13.5	78.7	23
DRGen [5]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	27.1	65.2	38.3
SD-ViT [22]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	34.2	71.4	46.2
SPSD-ViT [22]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	34.1	77	47.3
Mean	52.1 (29.3)	51.0 (29.2)	49.5 (29.8)	38.9 (22.7)	19.8 (10.8)	24.4 (13.6)	39.4 (15.7)	24.7 (20.7)	27.0 (14.4)	35.5 (22.3)	14.9 (9.1)	16.9 (10.4)	30.7 (17.8)	17.5 (9.2)	20.0 (9.1)	33.8 (21.1)	19.0 (12.3)	21.2 (12.2)	27.2 (9.7)	73 (6.1)	38.7 (11.2)
Max F1		80.3 (AT)			45.2 (LSTM)			64.3 (GANF)		36 (GAT)			37.8 (GAT)			43 (Omni)			47.3 (SPSD-ViT)		
DCB (%)		74%			57%			61%		38%			33%			52%			59%		

the DCB is a good indicator of the necessary conditions for DG, that requires that a domain should not introduce new causal factor. If it does introduce new causal factor then DCB % decreases as a consequence, accuracy (F1 in this case) of machines also decrease.

## H Code organization with SINDY-MPC and DR example

### H.1 Theta Data Generation and Preprocessing

In this study, we utilized four well-established ophthalmic imaging datasets—Messidor-1 (M1), Messidor-2 (M2), Aptos 2019 (Aptos), and EyePACS—which are prominently used in diabetic retinopathy (DR) research. To extract meaningful latent representations, we implemented an advanced preprocessing and data-generation pipeline leveraging the PySINDy framework, designed for sparse identification of nonlinear dynamical systems.

Initially, each retinal fundus image was standardized by resizing to a uniform resolution of  $512 \times 512$  pixels, followed by applying a circular mask to isolate the fundus region and reduce irrelevant peripheral noise. Image contrast was further enhanced using Contrast Limited Adaptive Histogram Equalization (CLAHE) within the LAB color space, significantly improving the visibility of subtle retinal features critical for downstream analysis.

Next, a unique radial sampling strategy was executed, wherein each preprocessed image was sampled in concentric circles starting from the central region outward, with sampling points placed at consistent angular intervals (every  $10^\circ$ ). This method generated a structured radial time series of RGB values, capturing how the retinal characteristics evolve from the image center toward the periphery.

The resulting radial trajectories served as input time-series data for the PySINDy model, enabling the discovery of underlying dynamical relationships inherent within retinal structures. For each sampled trajectory, the following computational steps were performed:

- 1169 1. Construct a time-series matrix  $X$  from the radial RGB trajectories.
- 1170 2. Compute the time derivative  $dX$  using forward finite differences.
- 1171 3. Apply the PySINDy framework using a third-degree polynomial feature library and a sparse
- 1172 thresholded least squares optimizer.
- 1173 4. Extract sparse model coefficients (referred to as  $\theta$ -values), representing latent dynamics
- 1174 describing radial RGB transitions.

1175 Robustness and scalability were integral to our implementation, demonstrated by processing images  
 1176 in batches of 1000, incorporating intermediate result checkpointing, systematic memory management,  
 1177 detailed error logging, and the ability to resume processing seamlessly after interruptions. The  
 1178 generated  $\theta$ -values, which succinctly encapsulate significant retinal features and underlying image  
 1179 dynamics, were stored alongside metadata—image identifiers, error metrics, and model performance  
 1180 measures—in organized output directories, providing a solid foundation for subsequent analyses.

## 1181 H.2 Domain Conformal Boundary Computation

1182 After collecting the  $\theta$ -vectors from PySINDy, we applied a specialized “RHO” analysis to each  
 1183 dataset (M1, M2, Aptos, EyePACS) to compute a conformal boundary quantifying cross-domain  
 1184 alignment. The procedure was:

- 1185 1. **Data loading and split.** Load all  $\theta$ -vectors (size  $N \times 128$ ) and split into training ( $Q_{\text{train}}$ ,
- 1186 60%) and validation ( $Q_{\text{val}}$ , 40%) sets.
- 1187 2. **Leave-one-out  $\rho$ -value (training).** For each  $\theta_k \in Q_{\text{train}}$ :
- 1188 (a) Define  $Q_{\text{train}}^{-k} = Q_{\text{train}} \setminus \{\theta_k\}$ .
- 1189 (b) For every  $\theta_j \in Q_{\text{train}}^{-k}$  compute

$$\rho_{kj} = \frac{\theta_k \cdot \theta_j}{\|\theta_k\| \|\theta_j\|}.$$

- 1190 (c) Average over  $j$ :

$$\rho_k^{(-k)} = \frac{1}{|Q_{\text{train}}^{-k}|} \sum_j \rho_{kj}.$$

1191 Then compute the overall reference mean:

$$\rho_{\text{avg}} = \frac{1}{|Q_{\text{train}}|} \sum_k \rho_k^{(-k)}.$$

- 1192 3. **Validation  $\rho$ -value and deviation.** For each  $\theta_m \in Q_{\text{val}}$ :

$$\rho_{mj} = \frac{\theta_m \cdot \theta_j}{\|\theta_m\| \|\theta_j\|} \quad \forall \theta_j \in Q_{\text{train}},$$

$$\rho_m = \frac{1}{|Q_{\text{train}}|} \sum_j \rho_{mj}, \quad \sigma_m = |\rho_m - \rho_{\text{avg}}|.$$

- 1194 4. **Conformal interval via order statistic.**

- 1195 • Collect the  $N_{\text{val}}$  deviations  $\{\sigma_m\}$  and sort ascending.
- 1196 • Set  $\alpha = 0.05$  for a 95% interval.
- 1197 • Compute cutoff index

$$\text{idx} = \lfloor \frac{N_{\text{val}}+1}{2} (1 - \alpha) \rfloor,$$

1198 and let  $\sigma^*$  be the deviation at that position.

- 1199 • The conformal interval is

$$\text{CI} = [-\sigma^*, +\sigma^*].$$

- 1200 5. **Implementation notes.**

- 1201 • GPU acceleration via CuPy/PyTorch, with CPU fallback.
- 1202 • Batching (1 000 images), checkpointing, and garbage collection.
- 1203 • Robust error and shape-validation checks.

### 1204 H.3 Single-Domain Generalization (SDG) Analysis

1205 To evaluate how well the learned  $\theta$ -features generalize across domains, we implemented a statistical  
1206 domain generalization (SDG). First,  $\theta$ -vectors and their image IDs are loaded for both the source and  
1207 target datasets and flattened for analysis. For each target image, we compute the cosine similarity  
1208 between its  $\theta$ -vector and all source  $\theta$ -vectors, yielding a similarity matrix. We then calculate the  
1209 average similarity ( $\rho_{\text{avg}}$ ) per test image, subtract a precomputed source reference average, and check  
1210 whether the result falls within the source confidence interval (CI). Each image is labeled “inside” or  
1211 “outside” the CI, and the final report records the percentage of target images deemed typical of the  
1212 source distribution.

#### 1213 H.3.1 Coverage Results

##### 1214 M1 On

- 1215 • **reference\_avg** = 0.04047960306748797
- 1216 • **conf\_interval** = (−0.03824305970024752, 0.03824305970024752)

##### 1217 M2 On

- 1218 • **reference\_avg** = 0.041717564380119414
- 1219 • **conf\_interval** = (−0.043998791641485506, 0.043998791641485506)

##### 1220 Aptos On

- 1221 • **reference\_avg** = 0.018978290495836176
- 1222 • **conf\_interval** = (−0.0596098734708454, 0.0596098734708454)

##### 1223 EyePACS On

- 1224 • **reference\_avg** = 0.01288287374567551
- 1225 • **conf\_interval** = (−0.04568944923551152, 0.04568944923551152)

### 1226 References

- 1227 [1] Karan Ahuja, Jia Wang, Amit Dhurandhar, Karthikeyan Shanmugam, and Kush R. Varshney.  
1228 Empirical or invariant risk minimization? a sample complexity perspective. arXiv preprint  
1229 arXiv:2010.16412, 2020.
- 1230 [2] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk mini-  
1231 mization. arXiv preprint arXiv:1907.02893, 2020.
- 1232 [3] Sanjeev Arora, Zhiyuan Li, and Abhishek Panigrahi. Understanding gradient descent on the  
1233 edge of stability in deep learning. In *International Conference on Machine Learning*, pages  
1234 948–1024. PMLR, 2022.
- 1235 [4] Devansh Arpit, Huan Wang, Yingbo Zhou, and Caiming Xiong. Ensemble of averages: Improv-  
1236 ing model selection and boosting performance in domain generalization. *Advances in Neural  
1237 Information Processing Systems*, 35:8265–8277, 2022.
- 1238 [5] Mohammad Atwany and Mohammad Yaqub. Drgen: domain generalization in diabetic retinopa-  
1239 thy classification. In *International Conference on Medical Image Computing and Computer-  
1240 Assisted Intervention*, pages 635–644. Springer, 2022.
- 1241 [6] Jean-Yves Audibert, Michal Michiardi, and Younès Boujemaa. Usad: Unsupervised anomaly  
1242 detection on multivariate time series. In *Proceedings of the 26th ACM SIGKDD International  
1243 Conference on Knowledge Discovery & Data Mining (KDD)*, pages 3395–3404. ACM, 2020.
- 1244 [7] Ayan Banerjee and Sandeep K.S. Gupta. Recovering implicit physics model under real-world  
1245 constraints. In *Proceedings of the 26th European Conference on Artificial Intelligence (ECAI  
1246 2024)*, volume 392 of *Frontiers in Artificial Intelligence and Applications*, pages 737–744. IOS  
1247 Press, 2024.
- 1248 [8] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer W.  
1249 Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175,  
1250 2010.

- 1251 [9] Asa Ben-Hur, Daniel Horn, Hava T. Siegelmann, and Vladimir Vapnik. Support vector clustering.  
1252 *Journal of Machine Learning Research*, 2(Dec):125–137, 2001.
- 1253 [10] C. Bennett and R. Thompson. Hybrid systems and state-space transformations for discontinuous  
1254 systems. *Systems & Control Letters*, 54(3):263–273, 2005.
- 1255 [11] Vance W Berger and YanYan Zhou. Kolmogorov–smirnov test: Overview. *Wiley statsref:*  
1256 *Statistics reference online*, 2014.
- 1257 [12] E. Camacho and S. Bordes. Hybrid systems in state-space representations for system design  
1258 and control. *Automatica*, 43(9):1462–1473, 2007.
- 1259 [13] Fabio M. Carlucci, Antonio D’Innocente, Sebastiano Bucci, Barbara Caputo, and Tatiana  
1260 Tommasi. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE*  
1261 *Conference on Computer Vision and Pattern Recognition*, pages 2229–2238, 2019.
- 1262 [14] Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee,  
1263 and Sungrae Park. Swad: Domain generalization by seeking flat minima. *Advances in Neural*  
1264 *Information Processing Systems*, 34:22405–22418, 2021.
- 1265 [15] Tony F Chan, Jianhong Shen, and Luminita Vese. Variational pde models in image processing.  
1266 *Notices AMS*, 50(1):14–26, 2003.
- 1267 [16] Kevin K Chen, Jonathan H Tu, and Clarence W Rowley. Variants of dynamic mode decom-  
1268 position: boundary condition, koopman, and fourier analyses. *Journal of nonlinear science*,  
1269 22:887–915, 2012.
- 1270 [17] Zhao Chen, Yang Liu, and Hao Sun. Physics-informed learning of governing equations from  
1271 scarce data. *Nature communications*, 12(1):6136, 2021.
- 1272 [18] M. David, S. Robert, and F. Christian. Hybrid systems: Computation and control (hscc). In  
1273 *Proceedings of the 14th International Conference on Hybrid Systems: Computation and Control*,  
1274 pages 56–68. Springer, 2011.
- 1275 [19] Z. Ding and Y. Fu. Deep domain generalization with structured low-rank constraint. *IEEE*  
1276 *Transactions on Image Processing*, 27(1):304–313, 2017.
- 1277 [20] Urban Fasel, J Nathan Kutz, Bingni W Brunton, and Steven L Brunton. Ensemble-sindy:  
1278 Robust sparse model discovery in the low-data, high-noise limit, with active learning and  
1279 control. *Proceedings of the Royal Society A*, 478(2260):20210904, 2022.
- 1280 [21] K.J. Friston, L. Harrison, and W. Penny. Dynamic causal modelling. *NeuroImage*, 19(4):1273–  
1281 1302, 2003.
- 1282 [22] Chamuditha Jayanga Galappaththige, Gayal Kuruppu, and Muhammad Haris Khan. Generaliz-  
1283 ing to unseen domains in diabetic retinopathy classification. In *Proceedings of the IEEE/CVF*  
1284 *Winter Conference on Applications of Computer Vision*, pages 7685–7695, 2024.
- 1285 [23] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François  
1286 Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural  
1287 networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.
- 1288 [24] Alejandro Garnung Menéndez. Physics meets pixels: Pde models in image processing. *arXiv*  
1289 *e-prints*, pages arXiv–2412, 2024.
- 1290 [25] M Grewal and Keith Glover. Identifiability of linear and nonlinear dynamical systems. *IEEE*  
1291 *Transactions on automatic control*, 21(6):833–837, 2003.
- 1292 [26] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint*  
1293 *arXiv:2007.01434*, 2020.
- 1294 [27] Ramin Hasani, Mathias Lechner, Alexander Amini, Daniela Rus, and Radu Grosu. Liquid  
1295 time-constant networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*,  
1296 volume 35, pages 7657–7666, 2021.
- 1297 [28] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Söderström.  
1298 Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In  
1299 *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery &*  
1300 *Data Mining (KDD)*, pages 387–395. ACM, 2018.



- [29] JAEB center. JAEB center dataset. <https://public.jaeb.org/datasets/diabetes>, 2023.
- [30] C. Jayanga, G. Kuruppu, and M. H. Khan. Generalizing to unseen domains in diabetic retinopathy classification. *arXiv preprint arXiv:2311.01673*, 2023.
- [31] Fredrik D. Johansson, David Sontag, and Rajesh Ranganath. Support and invertibility in domain-invariant representations. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, pages 527–536. PMLR, 2019.
- [32] Kadierdan Kaheman, J Nathan Kutz, and Steven L Brunton. Sindy-pi: a robust algorithm for parallel implicit sparse identification of nonlinear dynamics. *Proceedings of the Royal Society A*, 476(2242):20200279, 2020.
- [33] Eurika Kaiser, J Nathan Kutz, and Steven L Brunton. Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. *Proceedings of the Royal Society A*, 474(2219), 2018.
- [34] Payal Kamboj, Ayan Banerjee, Bin Xu, and Sandeep Gupta. Generating customized prompts for zero-shot rare event medical image classification using llm. *arXiv preprint arXiv:2501.16481*, 2025.
- [35] Siwon Kim, Kukjin Choi, Hyun-Soo Choi, Byunghan Lee, and Sungroh Yoon. Towards a rigorous evaluation of time-series anomaly detection, 2022.
- [36] Moez Krichen and Stavros Tripakis. Black-box conformance testing for real-time systems. In *International SPIN Workshop on Model Checking of Software*, pages 109–126. Springer, 2004.
- [37] David Krueger, Eduardo Caballero, Jakob H. Jacobsen, Alexander Zhang, Jonas Binas, Daniel Zhang, Raphaël Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *Proceedings of the International Conference on Machine Learning*, pages 5815–5826. PMLR, 2021.
- [38] Baochen Li, Yi Shen, Yuxin Wang, Weidi Zhu, Dong Li, Kurt Keutzer, and Huanru Zhao. Invariant information bottleneck for domain generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 7399–7407, 2022.
- [39] H. Li, S. J. Pan, S. Wang, and A. C. Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [40] Y. Li, X. Tian, M. Gong, Y. Liu, T. Liu, K. Zhang, and D. Tao. Deep domain generalization via conditional invariant adversarial networks. In *European Conference on Computer Vision (ECCV)*, 2018.
- [41] Yen Ting Lin, Yifeng Tian, Daniel Livescu, and Marian Anghel. Data-driven learning for the mori–zwanzig formalism: A generalization of the koopman learning framework. *SIAM Journal on Applied Dynamical Systems*, 20(4):2558–2601, 2021.
- [42] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM)*, pages 413–422. IEEE Computer Society, 2008.
- [43] Jiaqi Liu, Guoyang Xie, Jinbao Wang, Shangnian Li, Chengjie Wang, Feng Zheng, and Yaochu Jin. Deep industrial image anomaly detection: A survey. *Machine Intelligence Research*, 21(1):104–135, 2024.
- [44] Jinyang Liu, Wenwei Gu, Zhuangbin Chen, Yichen Li, Yuxin Su, and Michael R. Lyu. Mtad: Tools and benchmarks for multivariate time series anomaly detection, 2024.
- [45] Qinghua Liu and John Paparrizos. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 108231–108261. Curran Associates, Inc., 2024.
- [46] Yang Liu, Yao Zhang, Yixin Wang, Feng Hou, Jin Yuan, Jiang Tian, Yang Zhang, Zhongchao Shi, Jianping Fan, and Zhiqiang He. A survey of visual transformers. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

- [47] M. Mancini, S. R. Buló, B. Caputo, and E. Ricci. Best sources forward: Domain generalization through source-specific nets. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 1353–1357. IEEE, 2018.
- [48] M. Mathews and L. Carlson. Nonlinear state space modeling with discontinuous transitions. *Mathematical Methods in the Applied Sciences*, 29(8):1101–1122, 2006.
- [49] Emmanuel Menier, Sebastian Kaltenbach, Mouadh Yagoubi, Marc Schoenauer, and Petros Koumoutsakos. Interpretable learning of effective dynamics for multiscale systems. In *Proceedings A*, volume 481, page 20240167. The Royal Society, 2025.
- [50] J. Mitrovic, B. McWilliams, J. Walker, L. Buesing, and C. Blundell. Representation learning via invariant causal mechanisms. arXiv preprint arXiv:2010.07922, 2020.
- [51] Tomáš Pevný. Loda: Lightweight on-line detector of anomalies. *Machine Learning*, 102(2):275–304, 2016.
- [52] Thien Phung, Thanh Le, Tri-Lam Vuong, Truyen Tran, Anh Tran, Hung Bui, and Dorina Phung. On learning domain-invariant representations for transfer learning with multiple sources. In *Advances in Neural Information Processing Systems*, volume 34, 2021.
- [53] Markus Quade, Markus Abel, J Nathan Kutz, and Steven L Brunton. Sparse identification of nonlinear dynamics for rapid model recovery. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(6), 2018.
- [54] A. Slightly and J. Gadsen. A hybrid model approach for systems with discontinuities. *IEEE Transactions on Automatic Control*, 43(3):420–428, 1998.
- [55] R. Smith and Q. Zhang. Modeling hybrid systems with discontinuities in state-space representations. In *Proceedings of the IEEE Conference on Decision and Control*, pages 1859–1864. IEEE, 2008.
- [56] Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '19*, page 2828–2837, New York, NY, USA, 2019. Association for Computing Machinery.
- [57] M. Sultana, M. Naseer, M. H. Khan, S. Khan, and F. S. Khan. Self-distilled vision transformer for domain generalization. In *Asian Conference on Computer Vision (ACCV)*, 2022.
- [58] Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. Conformal prediction under covariate shift. *Advances in neural information processing systems*, 32, 2019.
- [59] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer Science & Business Media, 1999.
- [60] Daniele Venturi and Xiantao Li. The mori–zwanzig formulation of deep learning. *Research in the Mathematical Sciences*, 10(2):23, 2023.
- [61] Roberto Visentin, Enrique Campos-Náñez, Michele Schiavon, Dayu Lv, Martina Vettoretti, Marc Breton, Boris P Kovatchev, Chiara Dalla Man, and Claudio Cobelli. The uva/padova type 1 diabetes simulator goes from single meal to single day. *Journal of diabetes science and technology*, 12(2):273–281, 2018.
- [62] Long-Tung Vuong, Vy Vo, Hien Dang, Van-Anh Nguyen, Thanh-Toan Do, Mehrtash Harandi, Trung Le, and Dinh Phung. Why domain generalization fail? a view of necessity and sufficiency, 2025.
- [63] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip S Yu. Generalizing to unseen domains: A survey on domain generalization. *IEEE transactions on knowledge and data engineering*, 35(8):8052–8072, 2022.
- [64] Rui Wang, Meng Yi, Zhi Chen, and Song Zhu. Out-of-distribution generalization with causal invariant transformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 375–385, 2022.
- [65] S. Wang, L. Yu, K. Li, X. Yang, C.-W. Fu, and P.-A. Heng. Dofe: Domain-oriented feature embedding for generalizable fundus image segmentation on unseen datasets. *IEEE Transactions on Medical Imaging*, 39(12):4237–4248, 2020.

- [66] Yichao Wang, Hong Li, and Alex C. Kot. Heterogeneous domain generalization via domain mixup. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3622–3626. IEEE, 2020.
- [67] Xuefeng Xu, Jingcheng Xu, Zhaoxi Chen, Bo Sun, Hongtu Zhu, Chaochao Chen, and Jianwei Yin. Anomaly transformer: Time series anomaly detection with association discrepancy. In *International Conference on Learning Representations (ICLR)*, 2022.
- [68] Zhijian Xu, Ailing Zeng, and Qiang Xu. Fits: Modeling time series with  $10k$  parameters. *ICLR Spotlight*, 2024.
- [69] Haichao Yao, Yilin Wang, Shijie Li, Liang Zhang, Wen Liang, Jun Zou, and Chelsea Finn. Improving out-of-distribution robustness via selective augmentation. In *International Conference on Machine Learning*, pages 25407–25437. PMLR, 2022.
- [70] Xiang Yao, Yang Bai, Xin Zhang, Yi Zhang, Qi Sun, Rong Chen, Rui Li, and Bu Yu. Pcl: Proxy-based contrastive learning for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7097–7107, 2022.
- [71] Wei Yu, Kuiyuan Yang, Yalong Bai, Tianjun Xiao, Hongxun Yao, and Yong Rui. Visualizing and comparing alexnet and vgg using deconvolutional layers. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 3, pages 43–76, 2016.
- [72] Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. Tfad: A decomposition time series anomaly detection architecture with time-frequency analysis. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, CIKM '22*, page 2497–2507, New York, NY, USA, 2022. Association for Computing Machinery.
- [73] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. 2017.
- [74] N. L. Zhang, K. Li, H. Gao, W. Xie, Z. Lin, Z. Li, L. Wang, and Y. Huang. A causal framework to unify common domain generalization approaches. arXiv preprint arXiv:2307.06825, 2023.
- [75] Lichao Zhao, Tao Liu, Xinchao Peng, and Dimitris Metaxas. Maximum-entropy adversarial data augmentation for improved generalization and robustness. *Advances in Neural Information Processing Systems*, 33:14435–14447, 2020.
- [76] Ziwei Zhao, Yuhang Lu, Quanming Yao, and Yong Li. Graph-augmented normalizing flows for anomaly detection of multiple time series. In *International Conference on Learning Representations (ICLR)*, 2022.
- [77] Ke Zhou, Yuhong Yang, Tim M. Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *AAAI*, pages 13025–13032, 2020.
- [78] Ke Zhou, Yuhong Yang, Yufei Qiao, and Tao Xiang. Domain adaptive ensemble learning. *IEEE Transactions on Image Processing*, 30:8008–8018, 2021.
- [79] Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis by pretrained lm. *Advances in neural information processing systems*, 36:43322–43355, 2023.
- [80] Yujing Zhou, Cao Xiao, and Yan Liu. Multivariate time-series anomaly detection via graph attention networks. In *Proceedings of the 20th IEEE International Conference on Data Mining (ICDM)*, pages 841–850. IEEE, 2020.