Reconstructing long-term (1980-2022) daily ground particulate matter datasets in India (LongPMInd)

Shuai Wang[1], Mengyuan Zhang[1], Hui Zhao[2], Peng Wang[3,4], Sri Harsha Kota[5], Qingyan Fu[6], Cong Liu[7], Hongliang Zhang[1,4,8*]

[1]Department of Environmental Science and Engineering, Fudan University, Shanghai 200438, China

[2]School of Resources and Environmental Engineering, Jiangsu University of Technology, Changzhou 213001, China

[3]Department of Atmospheric and Oceanic Sciences, and Institute of Atmospheric Sciences, Fudan University, Shanghai, 200438, China

[4]IRDR ICoE on Risk Interconnectivity and Governance on Weather/Climate Extremes Impact and Public Health, Fudan University, Shanghai, China

[5]Department of Civil Engineering, Indian Institute of Technology, Delhi, 110016, India

[6] Shanghai Academy of Environmental Sciences, Shanghai 200003, China

[7]School of Public Health, Fudan University, Shanghai, 200032, China

[8]Institute of Eco-Chongming (IEC), Shanghai 200062, China

Correspondence to: Hongliang Zhang (zhanghl@fudan.edu.cn)

Figure

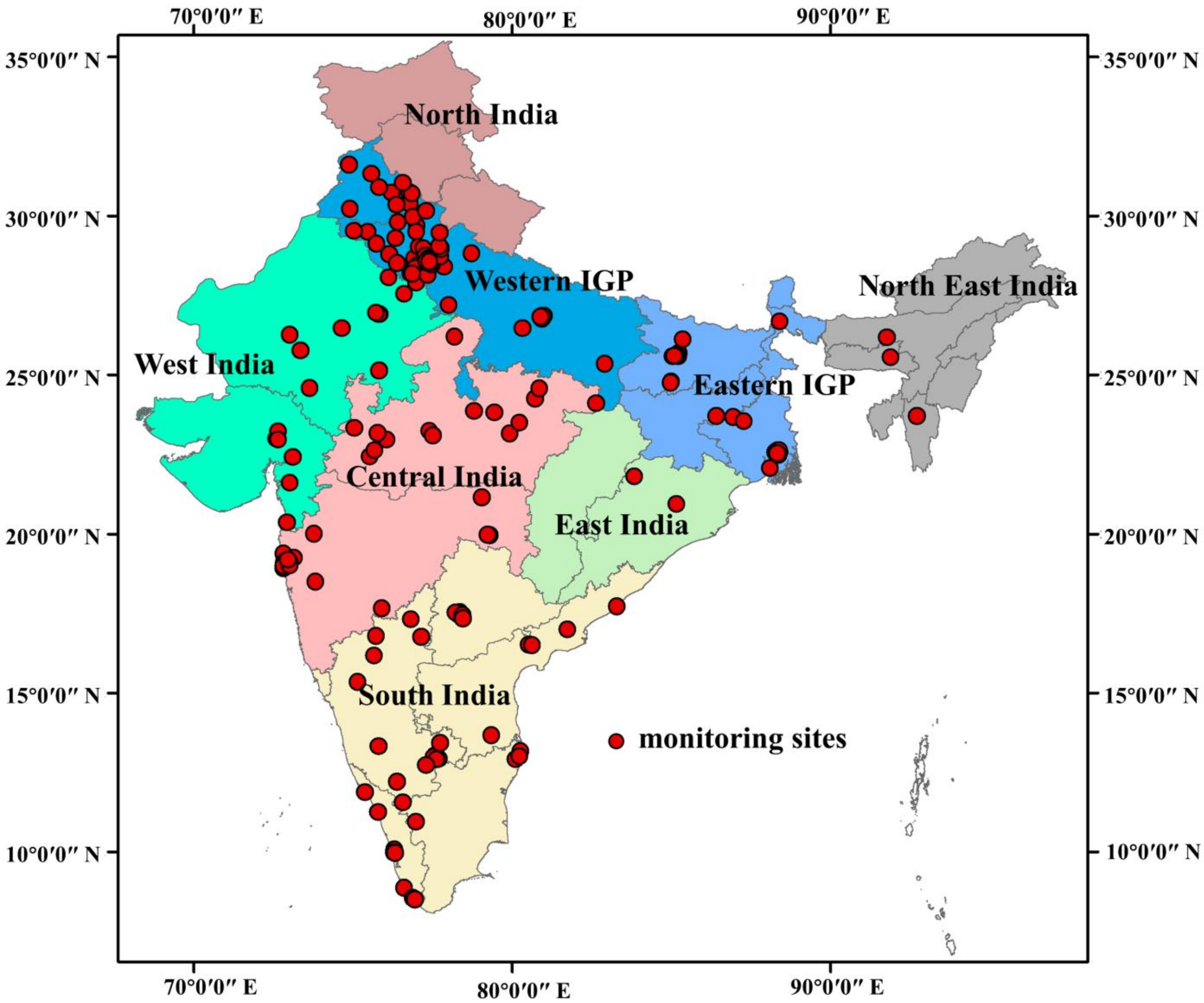

Figure S1: Research domain and location of monitoring sites in India.

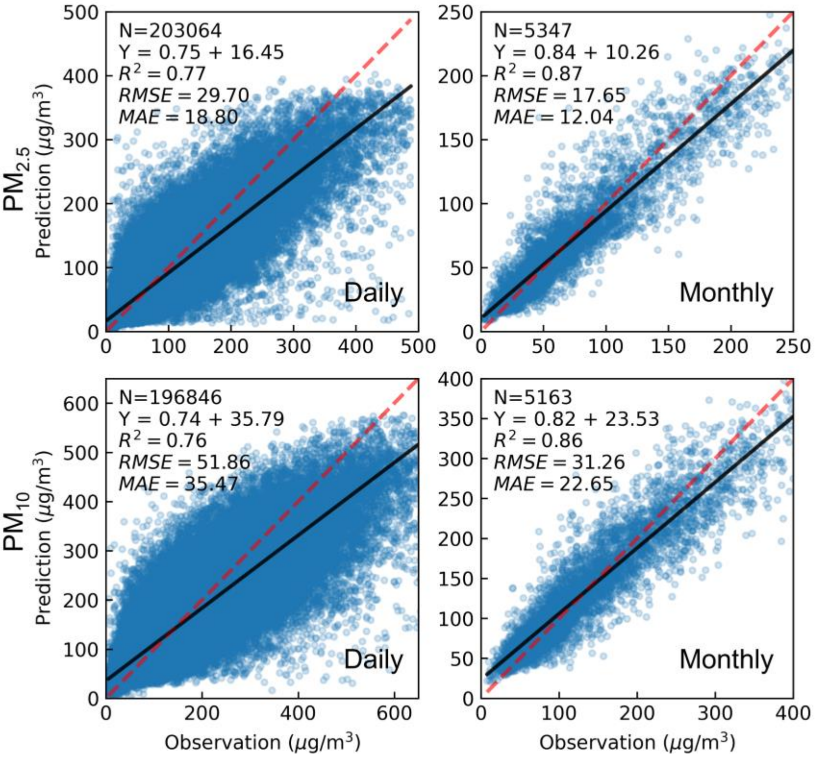

Figure S2: Comparison between observations and predictions of out-of-sample CV for daily and monthly PM2.5 and PM10. Dashed lines denote 1:1 line. Solid lines denote linear regression fitting. The sample numbers (N), linear regression function, R2, RMSE, and MAE are also shown. Units of RMSE and MAE are μg m^{-3}.

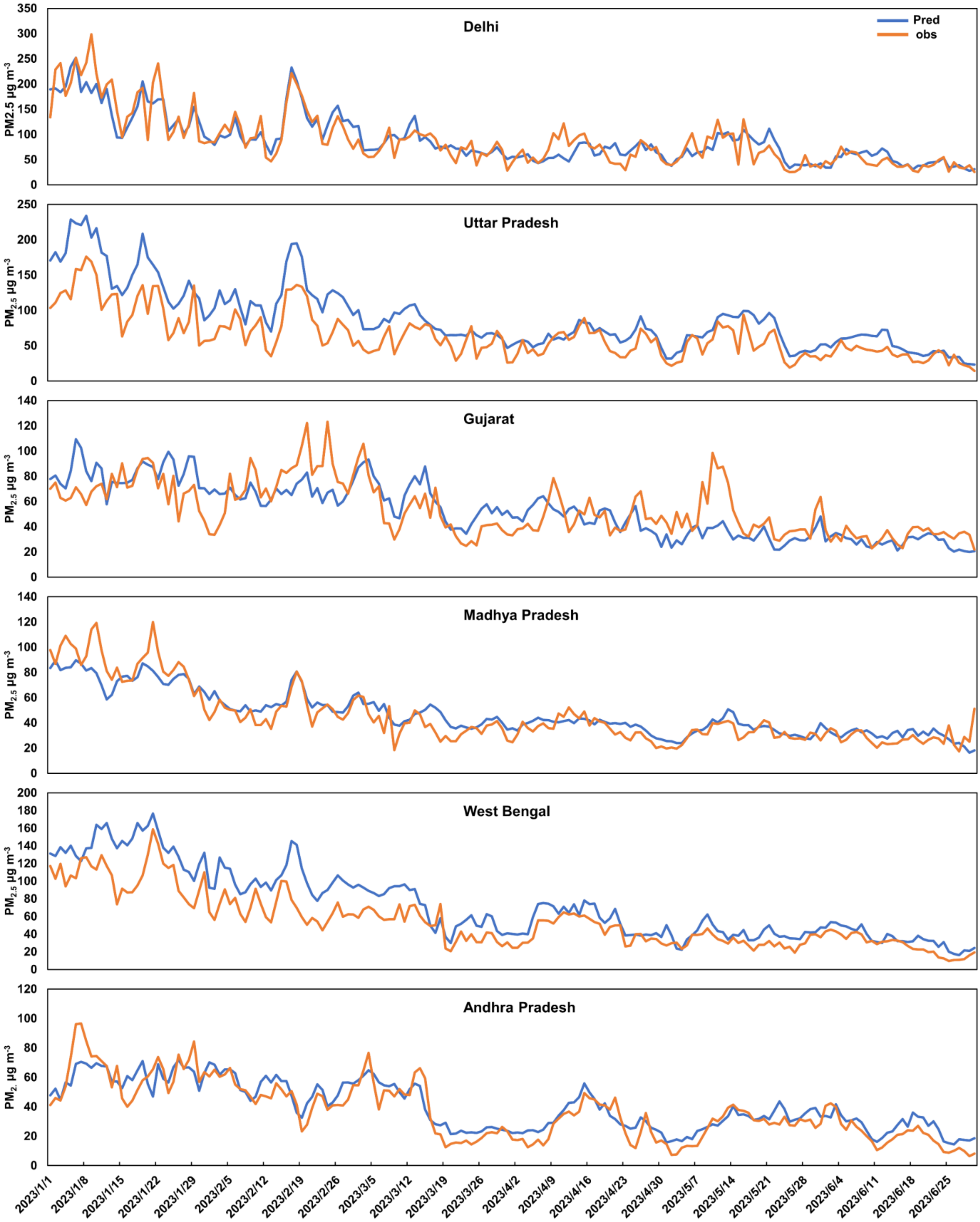

Figure S3: Comparison between observations and predictions of daily $PM_{2.5}$ for January - June in 2023.

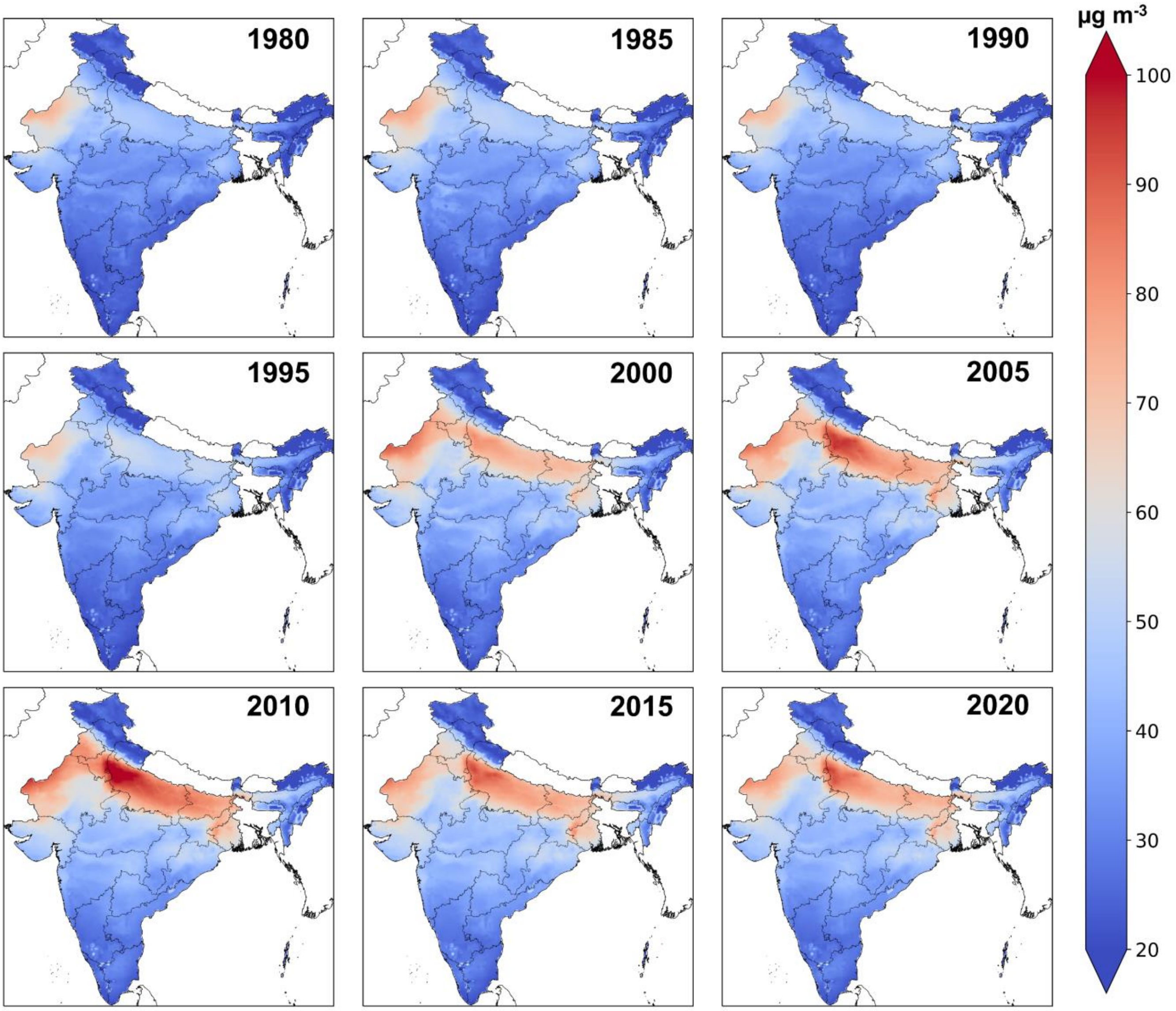

Figure S4: The spatial patterns of annual $PM_{2.5}$ in India during 1980-2022.

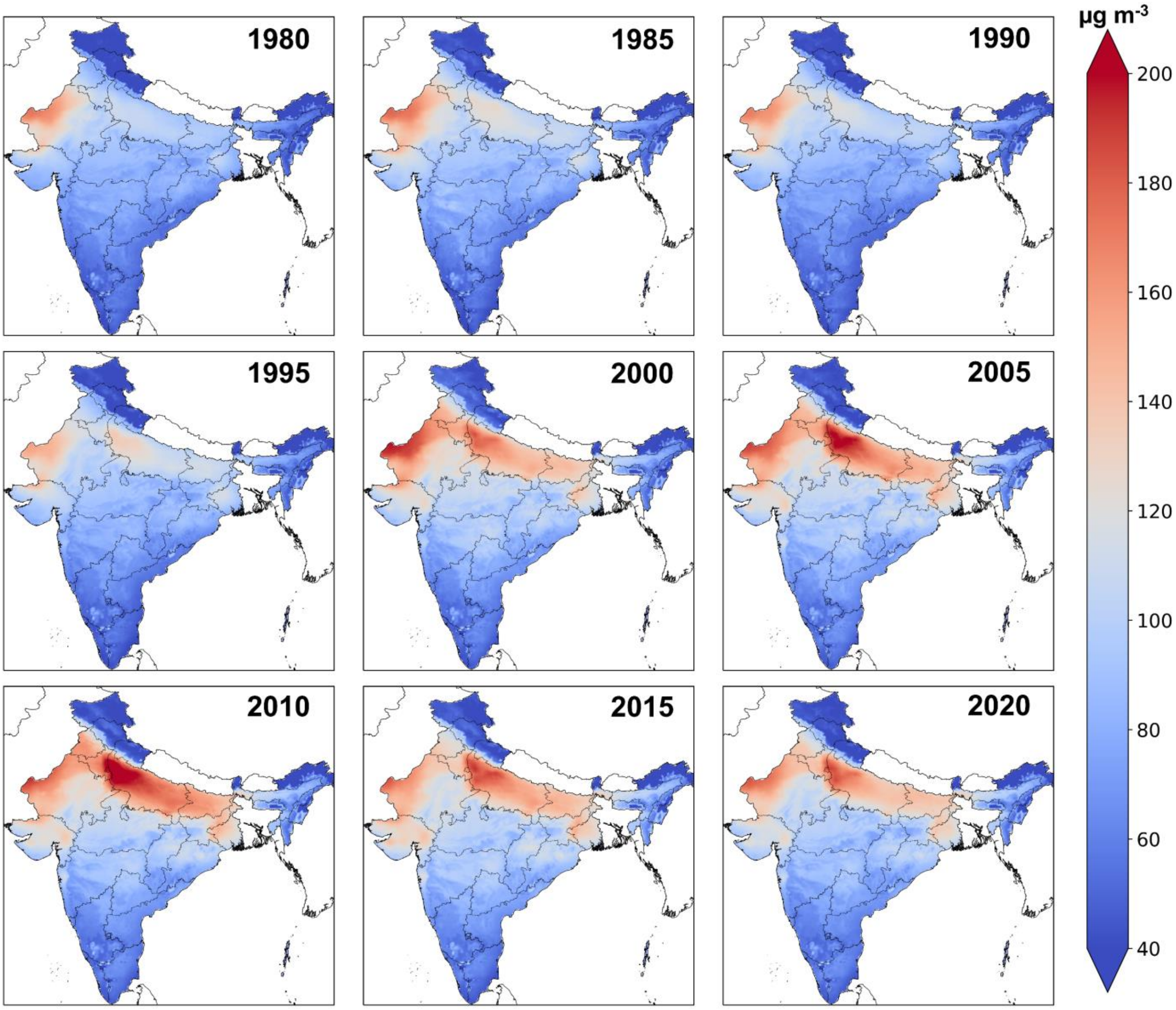

Figure S5: The spatial patterns of annual PM_{10} in India during 1980-2022.

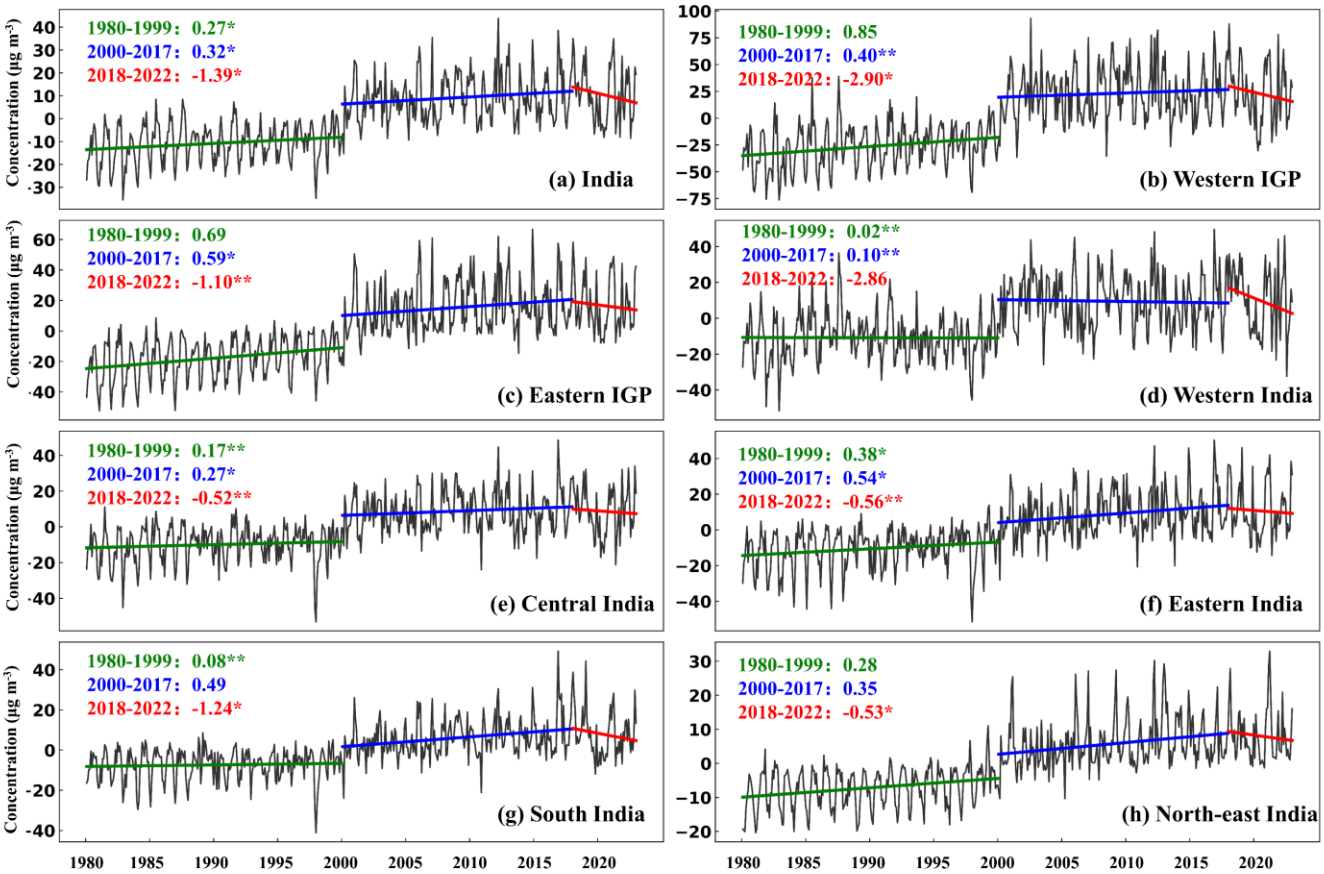

Figure S6: Time series of monthly PM_{10} anomaly from 1980 to 2022 in India and typical regions. The colored straight lines are the linear regression trend (μg/m^3/yr) for different period in China, and * represent the significance of the trends (*mean p < 0.05 and ** mean p < 0.01).

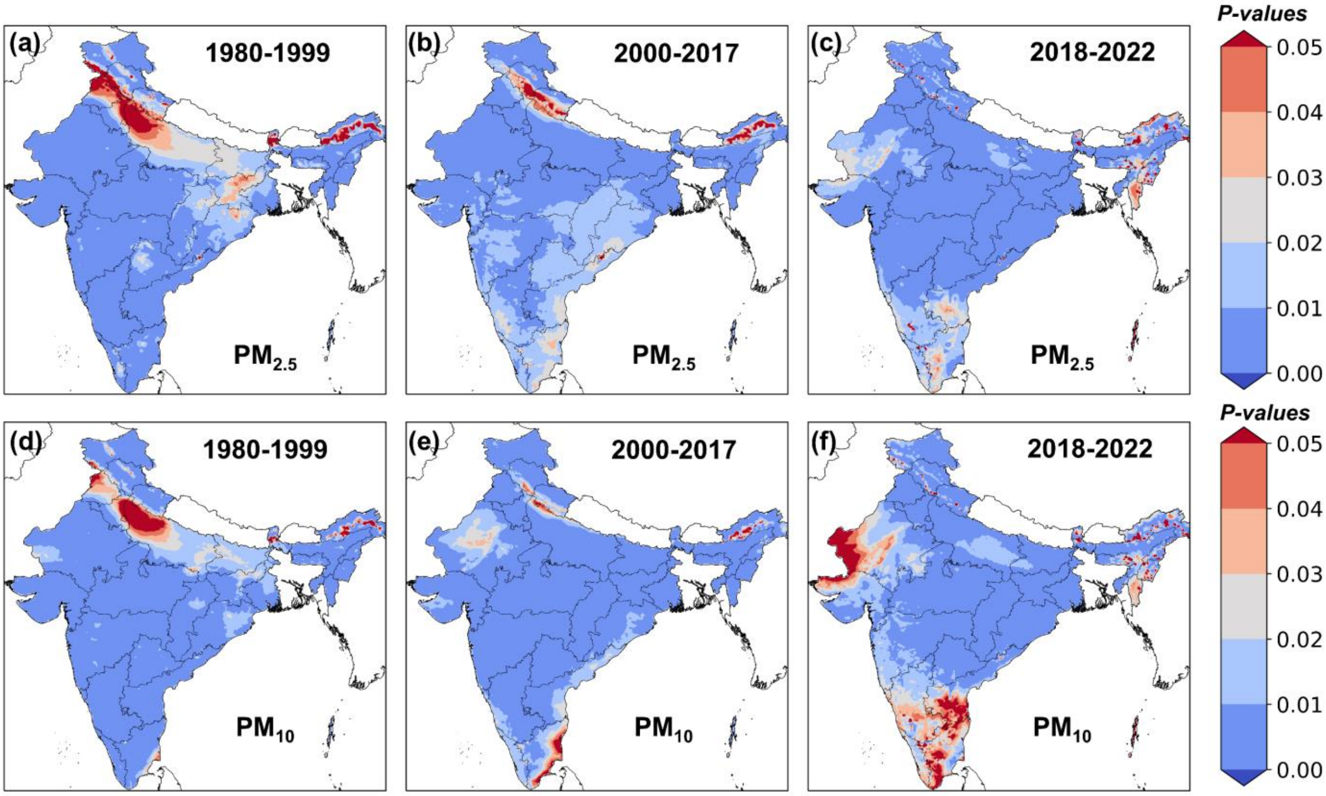

Figure S7: Spatial patterns of the significance of the trends for $PM_{2.5}$ and PM_{10} ($\mu g/m^3/yr$) during different period (1980-1999, 2000-2017, and 2018-2022).

Table

Table S1: Daily $PM_{2.5}$ and PM_{10} test results (not involved in training) for January - June in 2023. RSME and MAE unit: μg/m^3.

Species	Region	R	RMSE	MAE
$PM_{2.5}$	India	0.77	33.58	23.96
	Delhi	0.90	24.04	17.15
	Uttar Pradesh	0.91	32.72	25.67
	Gujarat	0.69	16.95	13.06
	Madhya Pradesh	0.92	10.03	7.28
	West Bengal	0.94	24.25	19.02
	Andhra Pradesh	0.92	8.34	6.71
PM_{10}	India	0.72	64.25	46.69
	Delhi	0.80	50.79	37.26
	Uttar Pradesh	0.85	54.05	44.15
	Gujarat	0.60	37.05	28.87
	Madhya Pradesh	0.88	19.31	14.40
	West Bengal	0.94	40.42	33.08
	Andhra Pradesh	0.87	16.62	13.36

Table S2: Uncertainty of estimated annual mortalities due to $PM_{2.5}$-induced diseases for India and typical regions during 2000-2019.

year	cvd_ihd	cvd_stroke	lri	neo_lung	resp_copd	t2_dm	all
2000	0.24 (0.22, 0.26)	0.17 (0.15, 0.19)	0.13 (0.12, 0.14)	0.01 (0.01, 0.01)	0.15 (0.12, 0.17)	0.03 (0.02, 0.03)	0.73 (0.65, 0.8)
2001	0.26 (0.23, 0.28)	0.18 (0.16, 0.19)	0.13 (0.12, 0.14)	0.01 (0.01, 0.01)	0.16 (0.13, 0.18)	0.03 (0.02, 0.03)	0.76 (0.68, 0.84)
2002	0.27 (0.25, 0.29)	0.18 (0.17, 0.2)	0.13 (0.12, 0.15)	0.01 (0.01, 0.01)	0.17 (0.14, 0.2)	0.03 (0.03, 0.03)	0.8 (0.72, 0.88)
2003	0.26 (0.24, 0.29)	0.18 (0.16, 0.2)	0.13 (0.11, 0.14)	0.01 (0.01, 0.01)	0.16 (0.14, 0.19)	0.03 (0.03, 0.03)	0.77 (0.69, 0.85)
2004	0.26 (0.24, 0.28)	0.17 (0.16, 0.19)	0.12 (0.11, 0.14)	0.01 (0.01, 0.01)	0.16 (0.13, 0.18)	0.03 (0.03, 0.03)	0.75 (0.68, 0.83)
2005	0.27 (0.25, 0.29)	0.18 (0.16, 0.2)	0.12 (0.11, 0.14)	0.01 (0.01, 0.01)	0.17 (0.14, 0.19)	0.03 (0.03, 0.03)	0.78 (0.7, 0.87)
2006	0.29 (0.27, 0.31)	0.19 (0.17, 0.2)	0.12 (0.11, 0.14)	0.01 (0.01, 0.01)	0.18 (0.15, 0.2)	0.03 (0.03, 0.04)	0.82 (0.74, 0.9)
2007	0.31 (0.29, 0.34)	0.2 (0.18, 0.21)	0.12 (0.11, 0.14)	0.01 (0.01, 0.01)	0.19 (0.16, 0.21)	0.04 (0.03, 0.04)	0.87 (0.78, 0.95)
2008	0.34 (0.31, 0.37)	0.21 (0.19, 0.23)	0.13 (0.12, 0.14)	0.01 (0.01, 0.02)	0.21 (0.17, 0.24)	0.04 (0.03, 0.04)	0.94 (0.84, 1.03)
2009	0.34 (0.31, 0.36)	0.2 (0.19, 0.22)	0.12 (0.11, 0.13)	0.01 (0.01, 0.02)	0.2 (0.17, 0.22)	0.04 (0.03, 0.04)	0.91 (0.82, 0.99)
2010	0.35 (0.32, 0.37)	0.2 (0.18, 0.22)	0.12 (0.11, 0.13)	0.02 (0.01, 0.02)	0.2 (0.16, 0.22)	0.04 (0.03, 0.04)	0.91 (0.82, 1)
2011	0.37 (0.34, 0.4)	0.21 (0.19, 0.23)	0.11 (0.1, 0.13)	0.02 (0.01, 0.02)	0.2 (0.17, 0.23)	0.04 (0.04, 0.04)	0.95 (0.85, 1.05)
2012	0.39 (0.36, 0.42)	0.22 (0.2, 0.24)	0.12 (0.11, 0.13)	0.02 (0.02, 0.02)	0.22 (0.18, 0.25)	0.04 (0.04, 0.05)	1.01 (0.91, 1.1)
2013	0.4 (0.37, 0.43)	0.22 (0.2, 0.24)	0.11 (0.1, 0.13)	0.02 (0.02, 0.02)	0.23 (0.19, 0.25)	0.05 (0.04, 0.05)	1.03 (0.92, 1.12)
2014	0.41 (0.38, 0.44)	0.23 (0.21, 0.25)	0.11 (0.1, 0.13)	0.02 (0.02, 0.02)	0.24 (0.19, 0.27)	0.05 (0.04, 0.05)	1.06 (0.95, 1.16)

2015	0.42 (0.38, 0.45)	0.23 (0.21, 0.25)	0.11 (0.1, 0.12)	0.02 (0.02, 0.02)	0.24 (0.19, 0.27)	0.05 (0.05, 0.05)	1.06 (0.95, 1.17)
2016	0.44 (0.4, 0.48)	0.25 (0.22, 0.27)	0.11 (0.1, 0.12)	0.02 (0.02, 0.02)	0.25 (0.2, 0.29)	0.05 (0.05, 0.06)	1.12 (0.99, 1.24)
2017	0.46 (0.42, 0.51)	0.25 (0.23, 0.29)	0.11 (0.1, 0.12)	0.02 (0.02, 0.02)	0.26 (0.21, 0.3)	0.05 (0.05, 0.06)	1.16 (1.01, 1.31)
2018	0.48 (0.42, 0.55)	0.27 (0.23, 0.31)	0.11 (0.1, 0.13)	0.02 (0.02, 0.03)	0.28 (0.22, 0.33)	0.06 (0.05, 0.06)	1.23 (1.05, 1.41)
2019	0.49 (0.42, 0.56)	0.27 (0.23, 0.31)	0.11 (0.09, 0.12)	0.02 (0.02, 0.03)	0.28 (0.21, 0.33)	0.06 (0.05, 0.07)	1.22 (1.03, 1.41)
mean	0.35 (0.32, 0.38)	0.21 (0.19, 0.23)	0.12 (0.11, 0.13)	0.02 (0.01, 0.02)	0.21 (0.17, 0.24)	0.04 (0.04, 0.04)	0.94 (0.84, 1.05)

Algorithm

Algorithm 1. HyperparametersSelect

```
param_range = hyperparameters range
delta_rmse0 = 9999
rmse _test0 = 9999
model = lightGBM(param)
for param in param_range do
    model.train(train_x_data, train_y_data)
    rmse_train = rmse (model.predict(train_x_data), train_y_data)
    rmse _test = rmse (model.predict(test_x_data), test_y_data)
    delta_rmse = rmse_train - rmse_test
    if rmse_test0 / rmse_test > 1.01 then
        rmse_test0 = rmse_test
    else then
        return param
    if delta_rmse / delta_rmse0 < 1.05 then
        delta_rmse0 = delta_rmse
    else then
        return param
end for
```