

EIF Performance

Extended Isolation Forest (extension_level = 0)

		Normal Function (Predicted)	Subnormal Function (Predicted)
Normal Function (Truth)	Normal Function (Truth)	3636 (96.39%)	43 (1.139%)
	Subnormal Function (Truth)	41 (1.086%)	52 (1.378%)

Extended Isolation Forest (extension_level = 1)

		Normal Function (Predicted)	Subnormal Function (Predicted)
Normal Function (Truth)	Normal Function (Truth)	3620 (95.97%)	59 (1.564%)
	Subnormal Function (Truth)	57 (1.511%)	36 (0.954%)

Extended Isolation Forest (extension_level = 2)

		Normal Function (Predicted)	Subnormal Function (Predicted)
Normal Function (Truth)	Normal Function (Truth)	3609 (95.67%)	70 (1.855%)
	Subnormal Function (Truth)	68 (1.802%)	25 (0.662%)

Extended Isolation Forest (extension_level = 3)

		Normal Function (Predicted)	Subnormal Function (Predicted)
Normal Function (Truth)	Normal Function (Truth)	3606 (95.59%)	73 (1.935%)
	Subnormal Function (Truth)	71 (1.882%)	22 (0.583%)

Extended Isolation Forest (extension_level = 4)

		Normal Function (Predicted)	Subnormal Function (Predicted)
Normal Function (Truth)	Normal Function (Truth)	3607 (95.62%)	72 (1.908%)
	Subnormal Function (Truth)	70 (1.855%)	23 (0.609%)

Extended Isolation Forest (extension_level = 5)

		Normal Function (Predicted)	Subnormal Function (Predicted)
Normal Function (Truth)	Normal Function (Truth)	3610 (95.70%)	69 (1.829%)
	Subnormal Function (Truth)	67 (1.776%)	26 (0.689%)

Ummmmmm.... ?

The performance starts roughly the same when extension = 0, which is expected as this is just standard IF.

However, as the extension parameter increases...

i.e. we draw more complex, higher-dimension, hyperplanes

... we see a deterioration of outlier detection performance?

I'm going to do the noble thing and blame it on the data.

What about those pesky
scoring regions?