

VAST

James T. Thorson

```
library(tinyVAST)
library(fmesher)
set.seed(101)
```

tinyVAST is an R package for fitting vector autoregressive spatio-temporal (VAST) models. We here explore the capacity to specify the vector-autoregressive spatio-temporal component.

Spatio-temporal autoregressive model

We first explore the ability to specify a first-order autoregressive spatio-temporal process:

```
# Simulate settings
theta_xy = 0.4
n_x = n_y = 10
n_t = 10
rho = 0.8
spatial_sd = 0.5

# Simulate GMRFs
R = exp(-theta_xy * abs(outer(1:n_x, 1:n_y, FUN="-")))
d = mvtnorm::rmvnorm(n_t, sigma=spatial_sd^2*kronecker(R,R) )

# Project through time and add mean
for( t in seq_len(n_t) ){
  if(t>1) d[t,] = rho*d[t-1,] + d[t,]
}
d = d + 2

# Shape into longform data-frame and add error
Data = data.frame( expand.grid(time=1:n_t, x=1:n_x, y=1:n_y), "var"="logn", z=as.vector(d))
Data$n = Data$z + rnorm(nrow(Data), sd=0.2)

# make mesh
mesh = fm_mesh_2d( Data[,c('x','y')] )

# fit model
out = fit( sem = "logn -> logn, -1, rho",
           data = Data,
           formula = n ~ 0 + factor(time),
           spatial_graph = mesh,
           quiet = TRUE )

out
```

```

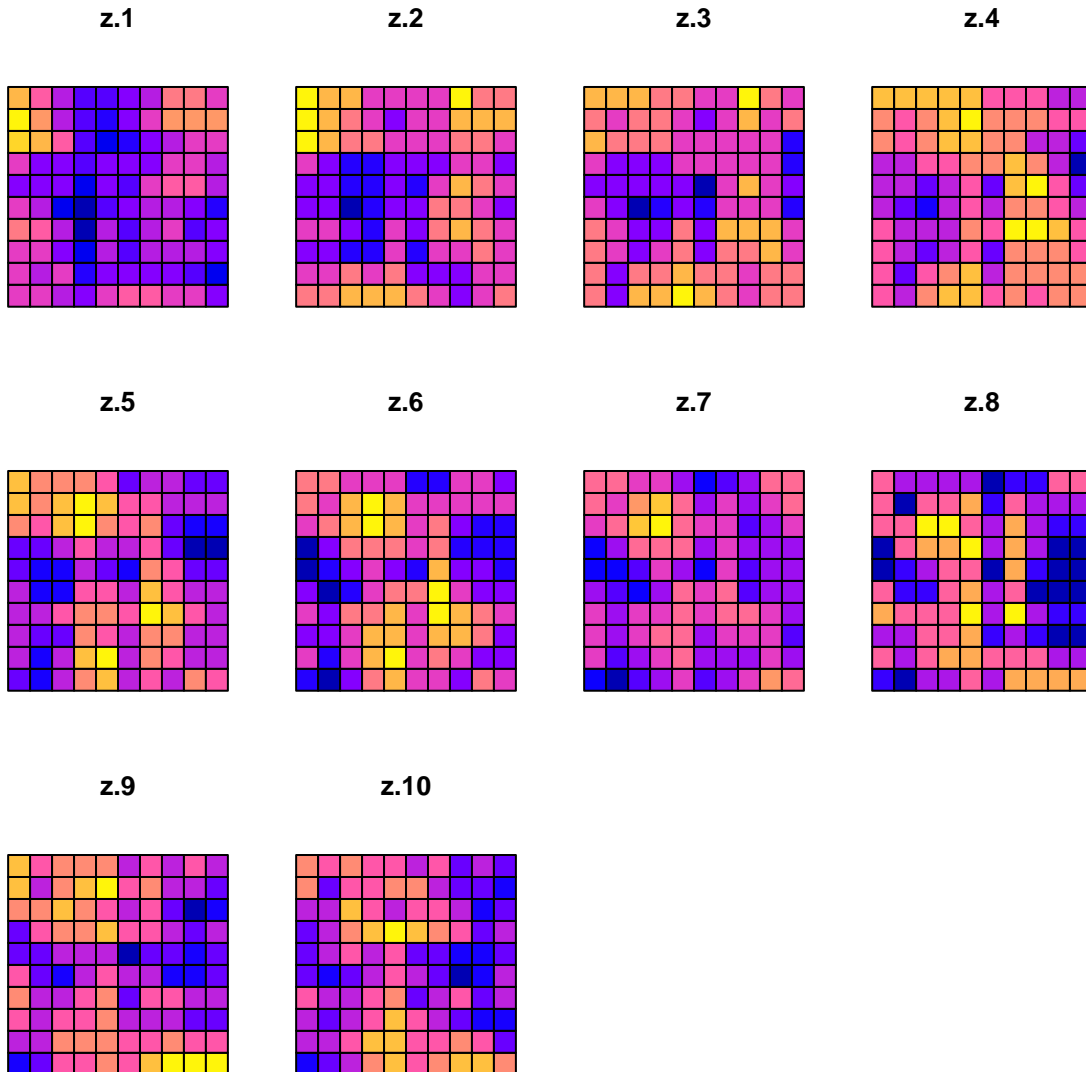
#> $call
#> fit(data = Data, formula = n ~ 0 + factor(time), sem = "logn -> logn, -1, rho",
#>     spatial_graph = mesh, quiet = TRUE)
#>
#> $opt
#> $opt$par
#>   log_kappa      alpha      alpha      alpha      alpha      alpha      alpha      alpha      alpha
#> -0.0242655  1.8698540  1.8358180  1.8750198  1.8515331  1.5253819  1.7166222  1.5876683  1.3815611
#>   beta_z  log_sigma
#>  0.3874778 -1.6006247
#>
#> $opt$objective
#> [1] 562.4018
#>
#> $opt$convergence
#> [1] 0
#>
#> $opt$iterations
#> [1] 52
#>
#> $opt$evaluations
#> function gradient
#>      76      53
#>
#> $opt$message
#> [1] "relative convergence (4)"
#>
#>
#> $sdrep
#> sdreport(.) result
#>           Estimate Std. Error
#> log_kappa -0.0242655 0.08572333
#> alpha      1.8698540 0.13882963
#> alpha      1.8358180 0.17847092
#> alpha      1.8750198 0.20059657
#> alpha      1.8515331 0.21408285
#> alpha      1.5253819 0.22262214
#> alpha      1.7166222 0.22813880
#> alpha      1.5876683 0.23174392
#> alpha      1.3815611 0.23411717
#> alpha      1.5879173 0.23568658
#> alpha      1.8828027 0.23672744
#> beta_z      0.8165793 0.03015656
#> beta_z      0.3874778 0.01720828
#> log_sigma -1.6006247 0.07505542
#> Maximum gradient component: 0.003510379
#>
#> $run_time
#> Time difference of 5.24138 secs

```

The estimated values for **beta_z** then correspond to the simulated value for **rho** and **spatial_sd**.

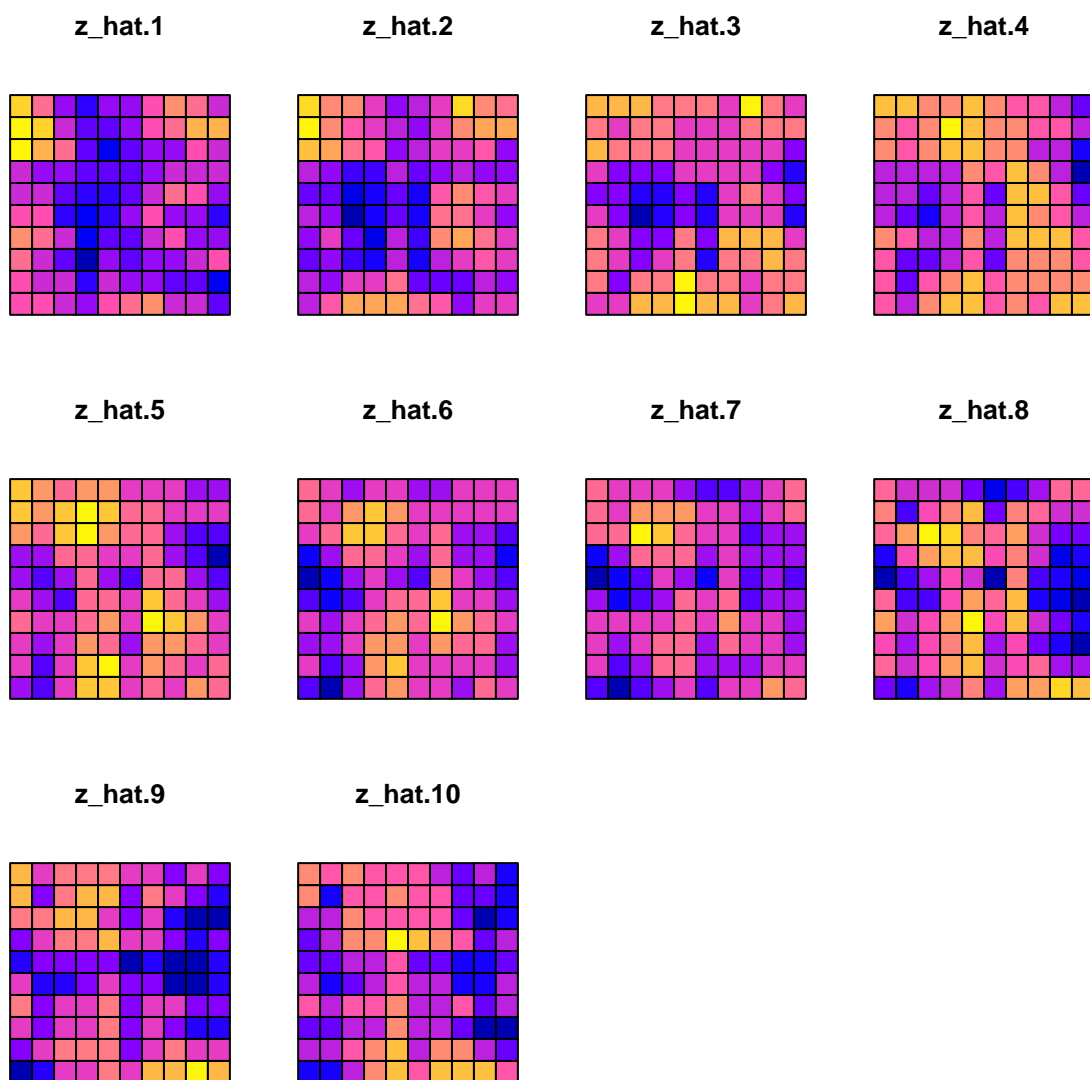
We can compare the true densities:

```
library(sf)
#> Warning: package 'sf' was built under R version 4.3.1
data_wide = reshape( Data[,c('x','y','time','z')],
                      direction = "wide", idvar = c('x','y'), timevar = "time")
sf_data = st_as_sf( data_wide, coords=c("x","y"))
sf_grid = sf::st_make_grid( sf_data )
sf_plot = st_sf(sf_grid, st_drop_geometry(sf_data) )
plot(sf_plot, max.plot=n_t )
```



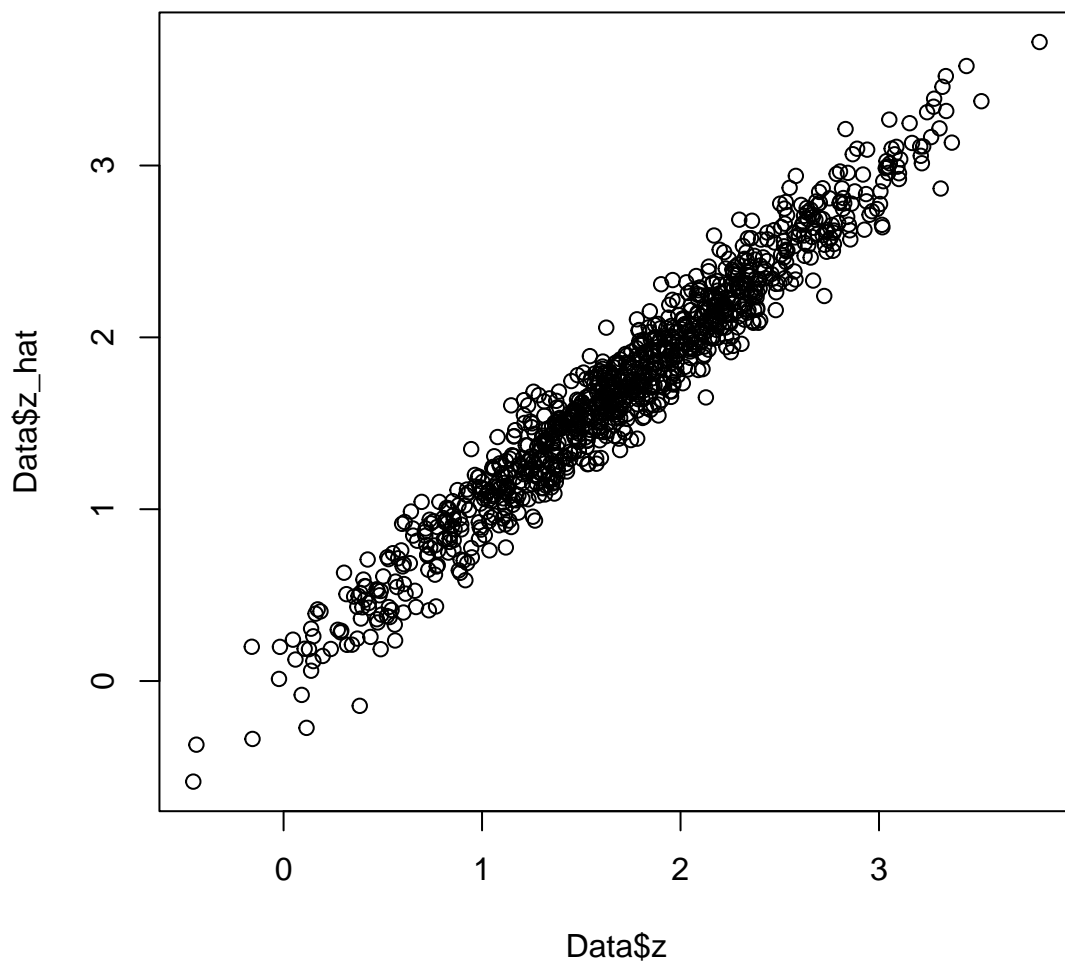
with the estimated densities:

```
Data$z_hat = predict(out)
data_wide = reshape( Data[,c('x','y','time','z_hat')],
                      direction = "wide", idvar = c('x','y'), timevar = "time")
sf_data = st_as_sf( data_wide, coords=c("x","y"))
sf_plot = st_sf(sf_grid, st_drop_geometry(sf_data) )
plot(sf_plot, max.plot=n_t )
```



where a scatterplot shows that they are highly correlated:

```
plot( x=Data$z, y=Data$z_hat )
```



Next, we compare this against the current version of VAST

```
library(INLA)
#> Warning: package 'INLA' was built under R version 4.3.1
#> Warning: package 'sp' was built under R version 4.3.1
library(VAST)
#> Warning: package 'units' was built under R version 4.3.1
#> Warning: package 'marginaleffects' was built under R version 4.3.1
settings = make_settings( purpose="index3",
                          n_x = n_x*n_y,
                          Region = "Other",
                          bias.correct = FALSE,
                          use_anisotropy = FALSE )
settings$FieldConfig['Epsilon','Component_2'] = 0
settings$FieldConfig['Omega',] = 0
settings$RhoConfig['Epsilon1'] = 4
```

```

myVAST = fit_model( settings=settings,
                    Lat_i = Data[, 'y'],
                    Lon_i = Data[, 'x'],
                    t_i = Data[, 'time'],
                    b_i = exp(Data[, 'n']),
                    a_i = rep(1, nrow(Data)),
                    observations_LL = cbind(Lat=Data[, 'y'], Lon=Data[, 'x']),
                    grid_dim_km = c(100, 100),
                    ObsModel = c(1, 4),
                    newtonsteps = 0,
                    loopnum = 0,
                    control = list(eval.max=100, iter.max=100, trace=0) )

#> Warning: The `returnclass` argument of `ne_download()` sp as of rnaturalearth 1.0.0.
#> i Please use `sf` objects with {rnaturalearth}, support for Spatial objects (sp) will be removed in
#> i The deprecated feature was likely used in the FishStatsUtils package.
#> Please report the issue to the authors.
#> This warning is displayed once every 8 hours.
#> Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
#> Warning in FishStatsUtils::inla.barrier.fem.copy(mesh = anisotropic_mesh, : Please install the `INL
#> which contains an implementation that runs faster!
#>      Component_1 Component_2
#> Omega          -1          -1
#> Epsilon         -2          -1
#> Beta            -2          -2
#> Epsilon_time    -3          -3
#> Eta1 Eta2
#>  -1  -1
#>      Coefficient_name Number_of_coefficients Type
#> 1      beta1_ft          10 Fixed
#> 2  Epsilon_rho1_f          1 Fixed
#> 3    L_epsilon1_z          1 Fixed
#> 4      logkappa1          1 Fixed
#> 5      logSigmaM          1 Fixed
#> 6 Epsiloninput1_sf      1360 Random
#> 0: 4894.2283: 20.0000 20.0000 20.0000 20.0000 20.0000 20.0000 20.0000 20.0000 20.0000
#> 1: 4438.1813: 19.9995 19.9995 19.9995 19.9995 19.9995 19.9995 19.9995 19.9994 19.9995
#> 2: 4425.9216: 19.9871 19.9864 19.9871 19.9869 19.9812 19.9849 19.9824 19.9787 19.9820
#> 3: 4410.5566: 19.5801 19.5608 19.5799 19.5742 19.4176 19.5182 19.4501 19.3473 19.4398
#> 4: 4259.7142: 14.6347 14.4725 14.6561 14.6065 13.1418 14.0869 13.4506 12.4874 13.3478
#> 5: 4184.8409: 12.6473 12.4300 12.6777 12.6109 10.6372 11.9104 11.0528 9.75485 10.9142
#> 6: 3883.8869: 8.51254 8.23137 8.51344 8.39364 5.91481 7.45055 6.36317 4.74708 6.19091
#> Warning in nlminb(start = startpar, objective = fn, gradient = gr, control = nlminb.control, : NA/Na
#> 7: 3429.2064: 4.31122 3.99694 4.30956 4.13994 1.25063 2.97992 1.70061 -0.171716 1.5001
#> Warning in nlminb(start = startpar, objective = fn, gradient = gr, control = nlminb.control, : NA/Na
#> 8: 3181.2403: 2.61824 2.33640 2.63385 2.40572 -0.513021 1.13494 -0.177412 -2.05427 -0.343
#> 9: 2752.0904: 2.60860 2.34357 2.62177 2.36867 -0.438463 1.09902 -0.173796 -1.98127 -0.314
#> 10: 2692.9981: 2.58078 2.35240 2.60481 2.30909 -0.330328 1.03964 -0.173797 -1.88290 -0.279
#> 11: 2524.1742: 2.38827 2.19480 2.42028 2.07078 -0.397943 0.790873 -0.364955 -1.97958 -0.428
#> 12: 2484.1287: 2.38740 2.19524 2.41958 2.06840 -0.393764 0.788209 -0.365340 -1.97588 -0.427
#> 13: 2475.8920: 2.38549 2.19625 2.41803 2.06305 -0.384490 0.782159 -0.366342 -1.96778 -0.424
#> 14: 2473.3779: 2.42816 2.24524 2.46366 2.10372 -0.323073 0.821311 -0.320157 -1.90742 -0.372
#> 15: 2466.5210: 2.47704 2.33184 2.52336 2.10678 -0.157839 0.804363 -0.285000 -1.77883 -0.289
#> 16: 2465.0350: 2.52992 2.42097 2.58177 2.12244 -0.000242209 0.794713 -0.251257 -1.64502 -0.

```

#> 17:	2457.4230:	2.63922	2.72944	2.67644	1.98309	0.670643	0.443700	-0.372721	-1.11277	0.03439
#> 18:	2451.9494:	2.79709	3.05871	2.83630	2.05927	1.25362	0.380820	-0.255511	-0.470920	0.4149
#> 19:	2445.8974:	2.56599	2.91929	2.59635	2.15139	1.15237	0.443918	-0.0960517	0.115605	0.6342
#> 20:	2444.2949:	2.56505	2.91648	2.59612	2.15053	1.15140	0.445779	-0.0928384	0.115063	0.6328
#> 21:	2443.6387:	2.55895	2.90409	2.59251	2.14266	1.14890	0.447399	-0.0835520	0.114173	0.6257
#> 22:	2442.7980:	2.54661	2.88288	2.58347	2.12596	1.14561	0.444389	-0.0726652	0.113963	0.6125
#> 23:	2442.1636:	2.54153	2.87158	2.58035	2.12055	1.14214	0.448763	-0.0627556	0.112418	0.6063
#> 24:	2441.3384:	2.52861	2.84968	2.56967	2.10314	1.13857	0.446703	-0.0518165	0.111325	0.5922
#> 25:	2440.7351:	2.52504	2.84002	2.56782	2.10073	1.13450	0.454232	-0.0408404	0.109338	0.5875
#> 26:	2439.9154:	2.51202	2.81755	2.55622	2.08290	1.13096	0.453936	-0.0289248	0.107046	0.5727
#> 27:	2439.3645:	2.50735	2.80561	2.55335	2.08005	1.12582	0.463669	-0.0154226	0.104759	0.5670
#> 28:	2438.5410:	2.49420	2.78289	2.54081	2.06213	1.12227	0.464776	-0.00308151	0.101762	0.552
#> 29:	2437.9948:	2.48815	2.76828	2.53652	2.05880	1.11589	0.477212	0.0133159	0.0992179	0.5452
#> 30:	2437.1793:	2.47473	2.74538	2.52289	2.04087	1.11242	0.479625	0.0259217	0.0959573	0.5303
#> 31:	2436.7024:	2.46759	2.72914	2.51716	2.03733	1.10527	0.494030	0.0439490	0.0934880	0.5231
#> 32:	2435.8541:	2.45398	2.70632	2.50250	2.01955	1.10221	0.497747	0.0567641	0.0903360	0.5086
#> 33:	2435.3782:	2.44576	2.68865	2.49520	2.01581	1.09450	0.513951	0.0760981	0.0881433	0.5013
#> 34:	2434.5439:	2.43172	2.66580	2.47923	1.99817	1.09168	0.518539	0.0887000	0.0855052	0.4874
#> 35:	2434.1089:	2.42259	2.64719	2.47046	1.99428	1.08377	0.536030	0.108716	0.0837924	0.48027
#> 36:	2433.2524:	2.40827	2.62453	2.45335	1.97693	1.08131	0.541218	0.120860	0.0819129	0.46728
#> 37:	2432.8130:	2.39831	2.60512	2.44313	1.97284	1.07341	0.559741	0.141330	0.0807968	0.46051
#> 38:	2431.9676:	2.38364	2.58248	2.42489	1.95574	1.07121	0.565317	0.152976	0.0798196	0.44846
#> 39:	2431.5464:	2.37281	2.56234	2.41307	1.95097	1.06371	0.584149	0.173385	0.0793822	0.44204
#> 40:	2430.6918:	2.35793	2.53981	2.39394	1.93418	1.06186	0.590020	0.184578	0.0793688	0.43106
#> 41:	2430.2783:	2.34661	2.51930	2.38102	1.92923	1.05473	0.609285	0.204952	0.0796499	0.42536
#> 42:	2429.4233:	2.33157	2.49681	2.36111	1.91273	1.05319	0.615365	0.215770	0.0806210	0.41545
#> 43:	2429.0040:	2.31945	2.47560	2.34661	1.90677	1.04676	0.634304	0.235717	0.0816798	0.41017
#> 44:	2428.1586:	2.30428	2.45305	2.32602	1.89050	1.04559	0.640655	0.246378	0.0836546	0.40136
#> 45:	2427.7273:	2.29145	2.43120	2.30999	1.88341	1.03997	0.658935	0.265676	0.0855313	0.39655
#> 46:	2426.8986:	2.27620	2.40855	2.28878	1.86736	1.03918	0.665604	0.276324	0.0885506	0.38889
#> 47:	2426.4512:	2.26268	2.38603	2.27118	1.85898	1.03440	0.682898	0.294778	0.0913029	0.38456
#> 48:	2425.6438:	2.24744	2.36322	2.24942	1.84316	1.03413	0.690119	0.305732	0.0954615	0.37822
#> 49:	2425.1785:	2.23344	2.34023	2.23050	1.83367	1.03022	0.706473	0.323417	0.0991740	0.37459
#> 50:	2424.3960:	2.21829	2.31725	2.20826	1.81815	1.03056	0.714501	0.334996	0.104639	0.369846
#> 51:	2423.9112:	2.20397	2.29389	2.18819	1.80768	1.02761	0.729978	0.352053	0.109516	0.367195
#> 52:	2423.1555:	2.18908	2.27082	2.16567	1.79265	1.02883	0.739352	0.364855	0.116626	0.364576
#> 53:	2422.6498:	2.17477	2.24741	2.14489	1.78164	1.02712	0.754543	0.381898	0.123092	0.363607
#> 54:	2416.9743:	1.80216	1.65897	1.55225	1.38986	1.12153	1.01591	0.742850	0.383206	0.378278
#> 55:	2413.9521:	1.83467	1.63004	1.55186	1.44221	1.20890	1.25707	0.974291	0.499839	0.572938
#> 56:	2410.5992:	1.75624	1.64408	1.60392	1.57803	1.21778	1.36744	1.13168	0.825915	0.879241
#> 57:	2410.3696:	1.75861	1.64413	1.60611	1.57788	1.21645	1.36595	1.13255	0.823480	0.880902
#> 58:	2410.2605:	1.76204	1.64438	1.60929	1.57769	1.21447	1.36376	1.13363	0.820051	0.883390
#> 59:	2410.0962:	1.76471	1.64517	1.61297	1.57916	1.21553	1.36490	1.13902	0.822284	0.889381
#> 60:	2409.9854:	1.76854	1.64623	1.61685	1.57964	1.21410	1.36356	1.14130	0.820445	0.893727
#> 61:	2409.8548:	1.77101	1.64782	1.62018	1.58150	1.21571	1.36580	1.14659	0.823830	0.899884
#> 62:	2409.7557:	1.77445	1.64980	1.62399	1.58289	1.21573	1.36646	1.15010	0.824862	0.905651
#> 63:	2409.6493:	1.77680	1.65197	1.62707	1.58504	1.21774	1.36948	1.15515	0.828971	0.911899
#> 64:	2409.5606:	1.77991	1.65445	1.63062	1.58685	1.21849	1.37112	1.15880	0.831275	0.918068
#> 65:	2409.4675:	1.78214	1.65689	1.63355	1.58922	1.22079	1.37457	1.16363	0.835783	0.924405
#> 66:	2409.3868:	1.78499	1.65958	1.63695	1.59131	1.22203	1.37674	1.16731	0.838820	0.930845
#> 67:	2409.3027:	1.78711	1.66212	1.63982	1.59385	1.22457	1.38042	1.17206	0.843589	0.937310
#> 68:	2409.2283:	1.78975	1.66488	1.64316	1.59616	1.22617	1.38289	1.17579	0.847123	0.943969
#> 69:	2409.1510:	1.79176	1.66742	1.64603	1.59884	1.22891	1.38670	1.18052	0.852093	0.950556

#> 70:	2409.0816:	1.79423	1.67019	1.64933	1.60133	1.23081	1.38937	1.18432	0.856003	0.957380
#> 71:	2409.0097:	1.79613	1.67271	1.65221	1.60414	1.23372	1.39325	1.18907	0.861140	0.964061
#> 72:	2408.9443:	1.79845	1.67545	1.65551	1.60677	1.23585	1.39607	1.19295	0.865349	0.971003
#> 73:	2408.8768:	1.80026	1.67794	1.65840	1.60968	1.23889	1.40001	1.19773	0.870629	0.977750
#> 74:	2408.8149:	1.80244	1.68066	1.66169	1.61243	1.24122	1.40294	1.20168	0.875085	0.984772
#> 75:	2408.7510:	1.80416	1.68311	1.66459	1.61543	1.24438	1.40693	1.20649	0.880490	0.991564
#> 76:	2408.6921:	1.80622	1.68581	1.66787	1.61828	1.24687	1.40996	1.21052	0.885153	0.998643
#> 77:	2408.6313:	1.80787	1.68822	1.67079	1.62135	1.25013	1.41400	1.21537	0.890665	1.00547
#> 78:	2408.5749:	1.80981	1.69089	1.67405	1.62428	1.25276	1.41712	1.21946	0.895504	1.01258
#> 79:	2408.5169:	1.81139	1.69327	1.67697	1.62741	1.25611	1.42120	1.22435	0.901111	1.01942
#> 80:	2408.4628:	1.81324	1.69590	1.68023	1.63041	1.25887	1.42441	1.22850	0.906101	1.02657
#> 81:	2408.4072:	1.81476	1.69825	1.68315	1.63360	1.26229	1.42854	1.23343	0.911790	1.03342
#> 82:	2408.3553:	1.81652	1.70085	1.68638	1.63665	1.26517	1.43184	1.23766	0.916913	1.04058
#> 83:	2408.3020:	1.81798	1.70317	1.68930	1.63988	1.26866	1.43602	1.24263	0.922675	1.04744
#> 84:	2408.2520:	1.81966	1.70573	1.69251	1.64299	1.27163	1.43939	1.24692	0.927915	1.05461
#> 85:	2408.2007:	1.82106	1.70802	1.69542	1.64625	1.27519	1.44362	1.25193	0.933742	1.06148
#> 86:	2408.1525:	1.82268	1.71054	1.69861	1.64940	1.27825	1.44708	1.25629	0.939084	1.06865
#> 87:	2408.1032:	1.82403	1.71280	1.70151	1.65269	1.28186	1.45135	1.26134	0.944970	1.07552
#> 88:	2408.0567:	1.82558	1.71528	1.70467	1.65587	1.28501	1.45488	1.26576	0.950404	1.08269
#> 89:	2408.0092:	1.82688	1.71750	1.70755	1.65919	1.28868	1.45921	1.27085	0.956344	1.08957
#> 90:	2407.9644:	1.82836	1.71994	1.71069	1.66240	1.29190	1.46281	1.27534	0.961861	1.09674
#> 91:	2407.9185:	1.82961	1.72213	1.71355	1.66573	1.29561	1.46718	1.28047	0.967849	1.10361
#> 92:	2407.8753:	1.83104	1.72452	1.71666	1.66896	1.29890	1.47086	1.28501	0.973441	1.11078
#> 93:	2407.8311:	1.83224	1.72667	1.71949	1.67230	1.30265	1.47527	1.29018	0.979475	1.11766
#> 94:	2407.7893:	1.83361	1.72902	1.72256	1.67554	1.30600	1.47901	1.29478	0.985136	1.12483
#> 95:	2407.7467:	1.83477	1.73112	1.72537	1.67890	1.30979	1.48346	1.29998	0.991213	1.13171
#> 96:	2407.7063:	1.83609	1.73342	1.72840	1.68214	1.31319	1.48727	1.30464	0.996937	1.13888
#> 97:	2407.6652:	1.83720	1.73548	1.73117	1.68549	1.31701	1.49175	1.30988	1.00305	1.14577
#> 98:	2407.6261:	1.83846	1.73773	1.73416	1.68874	1.32045	1.49561	1.31459	1.00884	1.15294
#> 99:	2407.5865:	1.83952	1.73974	1.73690	1.69208	1.32430	1.50013	1.31985	1.01499	1.15984
#> 100:	2407.5487:	1.84074	1.74195	1.73984	1.69533	1.32778	1.50404	1.32461	1.02083	1.16701
#> 101:	2407.5104:	1.84176	1.74391	1.74253	1.69865	1.33164	1.50857	1.32990	1.02703	1.17393
#> 102:	2407.4739:	1.84292	1.74606	1.74543	1.70188	1.33516	1.51253	1.33470	1.03292	1.18112
#> 103:	2407.4368:	1.84390	1.74797	1.74807	1.70518	1.33903	1.51708	1.34002	1.03915	1.18805
#> 104:	2407.4015:	1.84502	1.75006	1.75091	1.70838	1.34256	1.52106	1.34485	1.04509	1.19525
#> 105:	2407.3657:	1.84595	1.75193	1.75351	1.71165	1.34643	1.52562	1.35019	1.05136	1.20221
#> 106:	2407.3314:	1.84703	1.75397	1.75630	1.71483	1.34997	1.52962	1.35505	1.05734	1.20943
#> 107:	2407.2968:	1.84792	1.75579	1.75884	1.71806	1.35384	1.53417	1.36039	1.06365	1.21642
#> 108:	2407.2635:	1.84897	1.75779	1.76158	1.72120	1.35737	1.53817	1.36527	1.06966	1.22367
#> 109:	2407.2299:	1.84983	1.75957	1.76407	1.72439	1.36123	1.54272	1.37061	1.07601	1.23070
#> 110:	2407.1976:	1.85086	1.76154	1.76677	1.72749	1.36475	1.54670	1.37550	1.08205	1.23798
#> 111:	2407.1650:	1.85170	1.76329	1.76922	1.73064	1.36859	1.55122	1.38083	1.08843	1.24505
#> 112:	2407.1336:	1.85272	1.76524	1.77189	1.73370	1.37208	1.55517	1.38570	1.09449	1.25237
#> 113:	2407.1017:	1.85355	1.76698	1.77432	1.73681	1.37590	1.55966	1.39102	1.10088	1.25947
#> 114:	2407.0711:	1.85459	1.76894	1.77699	1.73985	1.37936	1.56356	1.39586	1.10694	1.26683
#> 115:	2407.0401:	1.85544	1.77070	1.77943	1.74294	1.38316	1.56800	1.40115	1.11334	1.27396
#> 116:	2407.0101:	1.85651	1.77270	1.78213	1.74597	1.38658	1.57184	1.40594	1.11936	1.28136
#> 117:	2406.9798:	1.85740	1.77452	1.78460	1.74907	1.39035	1.57623	1.41119	1.12575	1.28851
#> 118:	2406.9505:	1.85855	1.77660	1.78739	1.75212	1.39372	1.57998	1.41590	1.13171	1.29593
#> 119:	2406.5926:	1.87626	1.81852	1.84483	1.83551	1.50456	1.71286	1.56765	1.30924	1.47333
#> 120:	2406.5687:	1.86111	1.80902	1.84847	1.83163	1.49772	1.69550	1.55719	1.29873	1.48966
#> 121:	2406.4527:	1.86820	1.79611	1.83532	1.82193	1.48461	1.67822	1.53650	1.29643	1.49251
#> 122:	2406.4334:	1.88364	1.83032	1.83555	1.81021	1.47608	1.67715	1.55159	1.28854	1.50931


```

#> 123: 2406.4113: 1.89525 1.83715 1.84756 1.81080 1.47304 1.68099 1.55100 1.29767 1.50550
#> 124: 2406.3846: 1.89454 1.84096 1.86088 1.82100 1.48031 1.68445 1.55917 1.30247 1.50745
#> 125: 2406.3670: 1.89541 1.84300 1.86202 1.83788 1.49141 1.68867 1.55906 1.30151 1.51100
#> 126: 2406.3635: 1.89346 1.84326 1.87075 1.84263 1.50011 1.70094 1.55330 1.30020 1.52176
#> 127: 2406.3424: 1.89179 1.84892 1.88073 1.84941 1.50245 1.69834 1.56362 1.31017 1.52339
#> 128: 2406.3406: 1.89273 1.84873 1.88075 1.84995 1.50185 1.69835 1.56309 1.31078 1.52340
#> 129: 2406.3373: 1.89400 1.84880 1.88069 1.85067 1.50169 1.69803 1.56327 1.31155 1.52381
#> 130: 2406.3352: 1.89537 1.84890 1.88075 1.85138 1.50142 1.69777 1.56337 1.31247 1.52436
#> 131: 2406.3331: 1.89664 1.84937 1.88079 1.85178 1.50118 1.69762 1.56356 1.31347 1.52497
#> 132: 2406.3313: 1.89755 1.85012 1.88096 1.85169 1.50075 1.69779 1.56348 1.31447 1.52558
#> 133: 2406.3297: 1.89827 1.85028 1.88159 1.85164 1.50051 1.69797 1.56379 1.31544 1.52675
#> 134: 2406.3283: 1.89886 1.85035 1.88193 1.85210 1.50051 1.69806 1.56417 1.31636 1.52794
#> 135: 2406.3269: 1.89890 1.85125 1.88153 1.85252 1.50047 1.69846 1.56386 1.31698 1.52823
#> 136: 2406.3257: 1.89951 1.85106 1.88239 1.85301 1.50062 1.69861 1.56431 1.31777 1.52949
#> 137: 2406.3249: 1.90010 1.85082 1.88330 1.85350 1.50090 1.69875 1.56493 1.31855 1.53077
#> 138: 2406.3236: 1.90032 1.85148 1.88382 1.85389 1.50127 1.69937 1.56522 1.31940 1.53164
#> 139: 2406.3225: 1.90045 1.85224 1.88401 1.85449 1.50156 1.69995 1.56531 1.32017 1.53233
#> 140: 2406.3215: 1.90089 1.85220 1.88453 1.85532 1.50208 1.70017 1.56601 1.32096 1.53352
#> 141: 2406.3207: 1.90133 1.85234 1.88525 1.85587 1.50231 1.70049 1.56649 1.32182 1.53468
#> 142: 2406.3196: 1.90144 1.85291 1.88597 1.85602 1.50268 1.70129 1.56692 1.32267 1.53563
#> 143: 2406.3188: 1.90181 1.85342 1.88615 1.85685 1.50290 1.70153 1.56719 1.32347 1.53648
#> 144: 2406.3178: 1.90216 1.85358 1.88648 1.85776 1.50358 1.70185 1.56794 1.32427 1.53753
#> 145: 2406.3148: 1.90303 1.85505 1.89117 1.85916 1.50636 1.70593 1.57171 1.32915 1.54393
#> 146: 2406.3075: 1.90591 1.85974 1.89509 1.86788 1.51421 1.71293 1.58044 1.33958 1.55543
#> 147: 2406.3060: 1.90422 1.86154 1.89492 1.86777 1.51434 1.71675 1.58198 1.34261 1.55822
#> 148: 2406.3041: 1.90541 1.86110 1.89548 1.86996 1.51583 1.71825 1.58586 1.34631 1.56277
#> 149: 2406.3025: 1.90848 1.86366 1.89801 1.87059 1.51816 1.72033 1.58909 1.35030 1.56559
#> 150: 2406.3007: 1.90619 1.86223 1.90056 1.87464 1.52265 1.72422 1.59433 1.35608 1.57448
#> 151: 2406.2993: 1.90783 1.86719 1.90588 1.88064 1.52966 1.73039 1.59939 1.36256 1.58032
#> 152: 2406.2984: 1.91072 1.86963 1.90878 1.88500 1.53166 1.73632 1.60776 1.37042 1.58790
#> 153: 2406.2980: 1.91045 1.86881 1.90668 1.88218 1.53016 1.73500 1.60752 1.37193 1.59037
#> 154: 2406.2979: 1.91161 1.87018 1.90857 1.88441 1.53364 1.73715 1.60891 1.37253 1.59141
#> 155: 2406.2978: 1.91161 1.87029 1.90895 1.88450 1.53360 1.73765 1.60937 1.37275 1.59156
#> 156: 2406.2978: 1.91139 1.87041 1.90889 1.88450 1.53359 1.73720 1.60914 1.37333 1.59207
#> 157: 2406.2978: 1.91129 1.87065 1.90922 1.88470 1.53361 1.73758 1.60951 1.37383 1.59242
#> 158: 2406.2978: 1.91125 1.87030 1.90888 1.88416 1.53319 1.73730 1.60944 1.37375 1.59254
#> 159: 2406.2978: 1.91129 1.87034 1.90884 1.88428 1.53326 1.73725 1.60933 1.37358 1.59242
#> 160: 2406.2978: 1.91128 1.87036 1.90889 1.88431 1.53331 1.73731 1.60938 1.37365 1.59242

```

```
myVAST
```

```
#> fit_model(.) result
```

```
#> $par
```

```

#>      beta1_ft      beta1_ft      beta1_ft      beta1_ft      beta1_ft      beta1_ft      beta
#>      1.9112804      1.8703625      1.9088857      1.8843138      1.5333074      1.7373072      1.609
#>      beta1_ft      L_epsilon1_z      logkappa1      Epsilon_rho1_f      logSigmaM
#>      1.9129480      -0.4256334      -4.5141119      0.9396206      -1.1819730
#>

```

```
#> $objective
```

```
#> [1] 2406.298
```

```
#>
```

```
#> $iterations
```

```
#> [1] 5
```

```
#>
```

```
#> $evaluations
```

```

#> function gradient
#>      9      6
#>
#> $time_for_MLE
#> Time difference of 0.4639349 secs
#>
#> $max_gradient
#> [1] 0.0009569113
#>
#> $Convergence_check
#> [1] "The model is likely not converged"
#>
#> $number_of_coefficients
#> Total Fixed Random
#> 1374      14    1360
#>
#> $AIC
#> [1] 4840.596
#>
#> $diagnostics
#>

```

	Param	starting_value	Lower	MLE	Upper	final_gradient
#> 1	beta1_ft	1.9112762	-Inf	1.9112804	Inf	-1.601276e-04
#> 2	beta1_ft	1.8703611	-Inf	1.8703625	Inf	-1.014944e-04
#> 3	beta1_ft	1.9088872	-Inf	1.9088857	Inf	2.265398e-05
#> 4	beta1_ft	1.8843134	-Inf	1.8843138	Inf	2.843907e-05
#> 5	beta1_ft	1.5333092	-Inf	1.5333074	Inf	9.096140e-05
#> 6	beta1_ft	1.7373075	-Inf	1.7373072	Inf	8.865084e-05
#> 7	beta1_ft	1.6093756	-Inf	1.6093746	Inf	1.162509e-04
#> 8	beta1_ft	1.3736500	-Inf	1.3736440	Inf	1.717719e-04
#> 9	beta1_ft	1.5924244	-Inf	1.5924303	Inf	-1.646457e-04
#> 10	beta1_ft	1.9129472	-Inf	1.9129480	Inf	-1.281410e-04
#> 11	L_epsilon1_z	-0.4256342	-Inf	-0.4256334	Inf	5.249374e-04
#> 12	logkappa1	-4.5141158	-6.214608	-4.5141119	-3.565449	-1.934060e-04
#> 13	Epsilon_rho1_f	0.9396210	-0.990000	0.9396206	0.990000	-9.569113e-04
#> 14	logSigmaM	-1.1819727	-Inf	-1.1819730	10.000000	-4.258738e-05

```

#>
#> $SD
#> sdreport(.) result
#>

```

	Estimate	Std. Error
#> beta1_ft	1.9112804	0.12533621
#> beta1_ft	1.8703625	0.16951598
#> beta1_ft	1.9088857	0.20052381
#> beta1_ft	1.8843138	0.22442668
#> beta1_ft	1.5333074	0.24390978
#> beta1_ft	1.7373072	0.25938468
#> beta1_ft	1.6093746	0.27247405
#> beta1_ft	1.3736440	0.28383066
#> beta1_ft	1.5924303	0.29303093
#> beta1_ft	1.9129480	0.30089411
#> L_epsilon1_z	-0.4256334	0.02168763
#> logkappa1	-4.5141119	0.10217800
#> Epsilon_rho1_f	0.9396206	0.02503805
#> logSigmaM	-1.1819730	0.03921188

```
#> Maximum gradient component: 0.0009569113
#>
#> $time_for_sdreport
#> Time difference of 1.938026 secs
#>
#> $time_for_run
#> Time difference of 19.85776 secs
```

Or with sdmTMB

```
library(INLA)
library(sdmTMB)
#> Warning: package 'sdmTMB' was built under R version 4.3.1
mesh = make_mesh(Data, c("x","y"), n_knots=n_x*n_y )

start_time = Sys.time()
mysdmTMB = sdmTMB(
  formula = n ~ factor(time),
  data = Data,
  mesh = mesh,
  spatial = "off",
  spatiotemporal = "ar1",
  time = "time"
)
Sys.time() - start_time
#> Time difference of 4.568448 secs
```

Bivariate spatio-temporal autoregressive model

We next highlight how to specify a bivariate spatio-temporal model with a cross-lagged (vector autoregressive) interaction.

```
# Simulate settings
theta_xy = 0.2
n_x = n_y = 10
n_t = 20
B = rbind( c( 0.5, -0.25),
            c(-0.1,  0.50) )

# Simulate GMRFs
R = exp(-theta_xy * abs(outer(1:n_x, 1:n_y, FUN="-"))) )
d1 = mvtnorm::rmvnorm(n_t, sigma=0.2*kronecker(R,R) )
d2 = mvtnorm::rmvnorm(n_t, sigma=0.2*kronecker(R,R) )
d = abind::abind( d1, d2, along=3 )

# Project through time and add mean
for( t in seq_len(n_t) ){
  if(t>1) d[t,,] = t(B%*%t(d[t-1,,])) + d[t,,]
}
d[, ,1] = d[, ,1] + 2
d[, ,2] = d[, ,2] + 3
```

```

# Shape into longform data-frame and add error
Data = data.frame( expand.grid(time=1:n_t, x=1:n_x, y=1:n_y, "var"=c("d1","d2")), z=as.vector(d))
Data$z = Data$z + rnorm(nrow(Data), sd=0.2)

# make mesh
mesh = fm_mesh_2d( Data[,c('x','y')] )

# Define sem
sem = "
  d1 -> d1, -1, b11
  d2 -> d2, -1, b22
  d2 -> d1, -1, b21
  d1 -> d2, -1, b12
  d1 <-> d1, 0, var1
  d2 <-> d2, 0, var1
"

# fit model
out = fit( sem = sem,
           data = Data,
           formula = n ~ 0 + var,
           spatial_graph = mesh,
           quiet = TRUE )

#> Warning in Ops.factor(variables, 0): '>' not meaningful for factors
out
#> $call
#> fit(data = Data, formula = n ~ 0 + var, sem = sem, spatial_graph = mesh,
#>      quiet = TRUE)
#>
#> $opt
#> $opt$par
#>      log_kappa      alpha      alpha      beta_z      beta_z      beta_z      beta_z      beta_z      lo
#> -0.37119583  1.93011292  2.92522181  0.40288548  0.41412627 -0.19531475 -0.08333711  0.34839337 -1.7
#>
#> $opt$objective
#> [1] 1232.655
#>
#> $opt$convergence
#> [1] 0
#>
#> $opt$iterations
#> [1] 54
#>
#> $opt$evaluations
#> function gradient
#>      76      55
#>
#> $opt$message
#> [1] "relative convergence (4)"
#>
#>
#> $sdrep
#> sdreport(.) result

```

```

#>           Estimate Std. Error
#> log_kappa -0.37119583 0.046417959
#> alpha      1.93011292 0.067580370
#> alpha      2.92522181 0.066140302
#> beta_z     0.40288548 0.031927107
#> beta_z     0.41412627 0.031447063
#> beta_z    -0.19531475 0.031078105
#> beta_z    -0.08333711 0.029533567
#> beta_z     0.34839337 0.009009959
#> log_sigma -1.72787248 0.040607053
#> Maximum gradient component: 0.01495924
#>
#> $run_time
#> Time difference of 1.862839 mins

```

The values for `beta_z` again correspond to the specified value for interaction-matrix B