Una aproximación eficiente y flexible al modelado estadístico usando Template Model Builder (TMB)

Joaquin Cavieres

Doctor (c) en Estadística
Universidad de Valparaíso

Motivación

Para crear un modelo matemático/estadístico primero necesitamos tener claro el fenómeno a modelar, y segundo, debemos tener en consideración los siguientes puntos:

- Formulación del modelo
- Implementación del modelo
- Evaluación del modelo

Para crear un modelo matemático/estadístico primero necesitamos tener claro el fenómeno a modelar, y segundo, debemos tener en consideración los siguientes puntos:

- Formulación del modelo
- Implementación del modelo
- Evaluación del modelo

Para crear un modelo matemático/estadístico primero necesitamos tener claro el fenómeno a modelar, y segundo, debemos tener en consideración los siguientes puntos:

- Formulación del modelo
- Implementación del modelo
- Evaluación del modelo

Para formular un modelo **matemático/estadístico** necesitamos:

- Conocer las características inherentes del problema a resolver
- Conocer la función de distribución de la variable respuesta

Para implementar un modelo **matemático/estadístico** necesitamos:

- Ajustar el modelo utilizando nuestros datos (observaciones)
- Proponer una función de verosimilitud (likelihood) para la variable respuesta

Para evaluar un modelo **matemático/estadístico** necesitamos:

- Aplicar análisis de diagnósticos del modelo ajustado a los datos
- Evaluar la incertidumbre en las estimaciones de los parámetros

¿Como podemos hacer todo esto? → Template Model Builder (`TMB`)

Para evaluar un modelo **matemático/estadístico** necesitamos:

- Aplicar análisis de diagnósticos del modelo ajustado a los datos
- Evaluar la incertidumbre en las estimaciones de los parámetros

¿Como podemos hacer todo esto? → Template Model Builder (`TMB`)

Para evaluar un modelo **matemático/estadístico** necesitamos:

- Aplicar análisis de diagnósticos del modelo ajustado a los datos
- Evaluar la incertidumbre en las estimaciones de los parámetros

¿Como podemos hacer todo esto? $\rightarrow$ Template Model Builder (`TMB`)

¿Que es TMB?

- Template Model Builder (TMB) es una librería "open source" de R que permite una rápida implementación de modelos estadísticos, p. ej; efectos aleatorios, jerárquicos, splines, etc (Kristensen et al. (2015)).

- TMB usa la metodología frecuentista y utliza la diferenciación automática (AD en inglés) para obtener las primeras y segundas derivadas de una función (p. ej log-verosimilitud, Skaug and Fournier (2006))

- Ofrece de manera simple el cálculo en paralelo en donde el usuario puede definir la log-verosimilitud para los datos y efectos aleatorios en un arhivo (template) de C++, mientras todas las demás operaciones son realizadas en R

¿Qué es TMB?

- TMB está diseñado para modelos estadísticos de estructura compleja
- Usa la verorsimilitud marginal
- La incertidumbre del modelo es calculada mediante la estimación asintótica de las matrices de varianza-covarianza y calcula los perfiles de verosimilitud para los parámetros estimados.
- Compatible con Stan para hacer inferencia Bayesiana (tmbstan)

¿Por que usar `TMB`?

- La función objetivo (y sus derivadas) pueden ser llamadas desde `R`, por lo tanto, la optimización de los parámetros se pueden hacer vía cualquier optimizador (p. ej `nlminb()`)
- El usuario puede usar la aproximación de Laplace para obtener la verosimilitud marginal de los efectos aleatorios
- Estima las desviaciones estándar de los parámetros por el método Delta.

¿Por que usar `TMB`?

- La función objetivo (y sus derivadas) pueden ser llamadas desde `R`, por lo tanto, la optimización de los parámetros se pueden hacer vía cualquier optimizador (p. ej `nlminb()`)
- El usuario puede usar la aproximación de Laplace para obtener la verosimilitud marginal de los efectos aleatorios
- Estima las desviaciones estándar de los parámetros por el método Delta.

Instalando `TMB`

- Desde `github`: https://github.com/kaskr/adcomp

  git clone https://github.com/kaskr/adcomp

- Desde `R`:

```
install.packages("TMB")

library(TMB)

# Para testear que la instalación es correcta:

runExample(all=TRUE)
```

Nota: Quizás necesite instalar en primera instancia `Rtools`

- https://cran.r-project.org/bin/windows/Rtools/

Reiniciar `R` despues de la intalación de `Rtools` y ejecutar:

```
library(devtools)
```

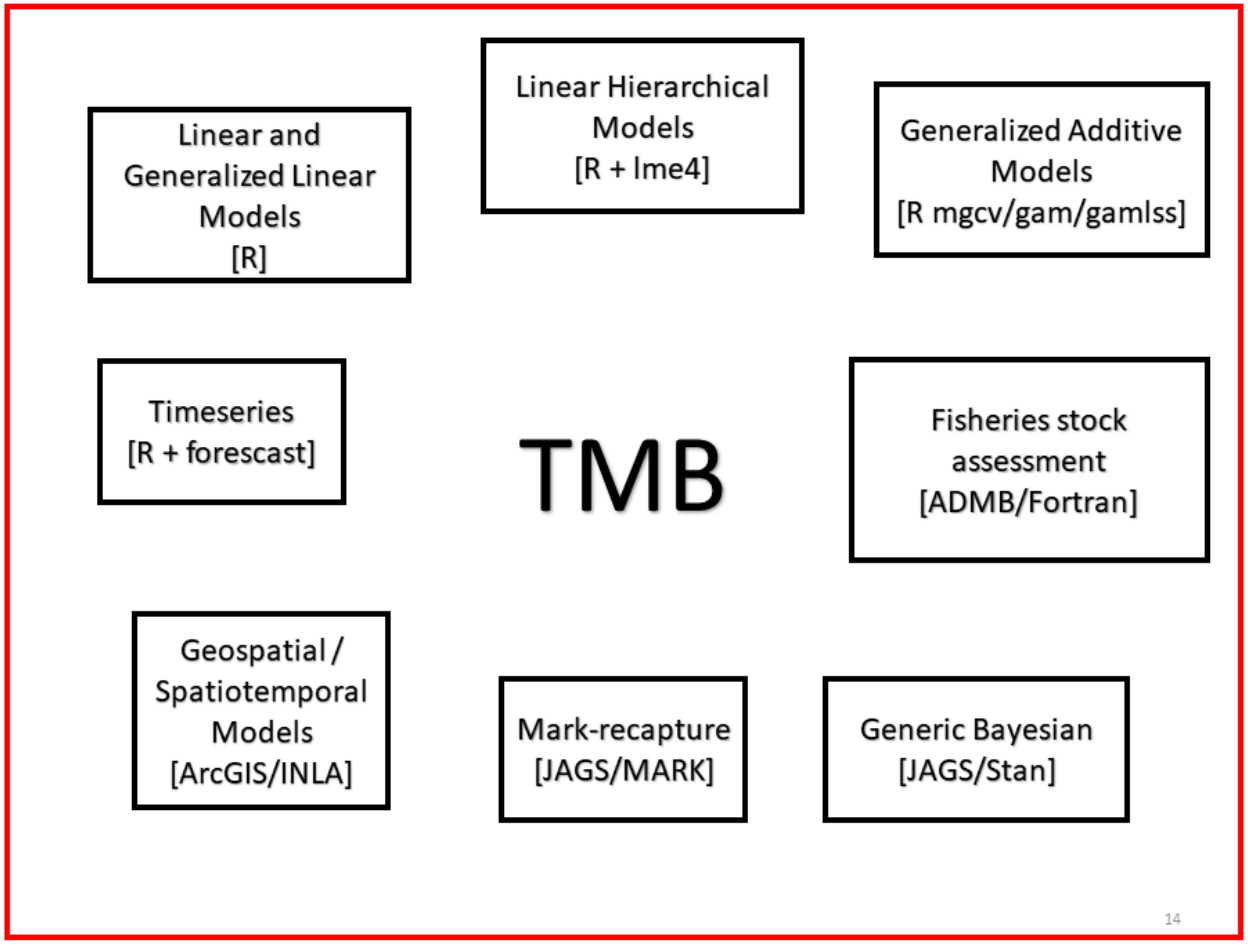


Figura 1: Resumen de los modelos que se pueden crear con `TMB` (por Cole. C Monnahan)

TMB workflow

- Proponer un modelo estadístico
- Escribir un template en C++ (en R) para calcular la log-verosimilitud negativa dada los paramétros
- Compilar el modelo y "cargarlo" en R
- Declarar que parámetros son aleatorios
- Ajustar el modelo usando R y minimizar la función retornada por TMB
- Hacer inferencia desde el modelo ajustado

Template Model Builder (TMB)

Secciones en TMB

El modelo escrito en C++ tiene la siguiente estructura:

- Leer los datos desde R
- Setear los parámetros
- Calculos:
  1. Media del modelo dado los parámetros
  2. Negative log-likelihood (NLL)
- Reportar los resultados de vuelta a R
- Retornar la NLL

DATA section

Importando los datos desde R

TMB syntax	C++ syntax	R syntax
DATA_VECTOR(x)	vector<Type>	vector()
DATA_MATRIX(x)	matrix<Type>	matrix()
DATA_SCALAR(x)	Type	numeric()
DATA_INTEGER(x)	int	integer()
DATA_FACTOR(x)	vector<int>	factor()
DATA_ARRAY(x)	array<Type>	array()
DATA_SPARSE_MATRIX(x)	Eigen::SparseMatrix<Type>	dgTMatrix()

PARAMETER section

TMB syntax	C++ syntax	R syntax
`PARAMETER_MATRIX(x)`	`matrix<Type>`	`matrix()`
`PARAMETER_VECTOR(x)`	`vector<Type>`	`vector()`
`PARAMETER_ARRAY(x)`	`array<Type>`	`array()`
`PARAMETER(x)`	`Type`	`numeric()`

REPORT section

Retornar los resultados a `R`

- Retornar el objeto via `REPORT()`, por ejemplo:

```
REPORT(predict);
```

- En `R` se debe hacer:

```
obj$report()
```

- Reportar los parámetros desde el modelo ajustado

```
obj$report(par)
```

Calculando la -log-likelihood

- Calcular la función de verosimilitud (likelihood), e.g,. `dnorm()`

```
nll= -dnorm(y(i), mu(i), sigma, true);
```

- Para la clase de vectores, se necesita hacer:

```
nll= -dnorm(y, mu, sigma, true).sum();
```

- Se necesita retornar el valor negativo de la NLL
- En la última parte del template C++ se debe poner `return nll;`
- The `nll` debe ser una variable del tipo escalar

Calculando las varianzas

`TMB` retorna las varianzas asintóticas para los parámetros mediante la función `sdreport(obj)`

- Calcular las desviaciones estándar desde las cantidades derivadas

```
Type nu = exp(eta);
ADREPORT(nu);
```

Modelado estadístico en TMB

Considere el siguiente modelo con efectos mixtos:

$$\begin{aligned} \boldsymbol{y} \mid \boldsymbol{\beta}, \boldsymbol{u}, \theta_1 &\sim \pi_1(\boldsymbol{y} \mid \boldsymbol{\beta}, \boldsymbol{u}, \theta_1) \\ \boldsymbol{u} \mid \theta_2 &\sim \pi_2(\boldsymbol{u} \mid \theta_2) \end{aligned}$$

donde $\boldsymbol{y}$ representan los datos, π_1 y π_2 es la verosimilitud y la distribución de los efectos aleatorios respectivamente, $\boldsymbol{\beta}$ representan los parámetros asociados a los efectos fijos, $\boldsymbol{u}$ son los efectos aleatorios y $\boldsymbol{\theta} = (\theta_1, \theta_2)$ los parametros de variance-covarianza en donde θ_1 parece en la verosimilitud y θ_2 aparece en la prior de los efectos aleatorios.

Laplace approximation

- Definir la log-verosimilitud

$$f(\beta, \boldsymbol{u}, \boldsymbol{\theta}) = \log \pi_1 \left(\boldsymbol{y} \mid \beta, \boldsymbol{u}, \theta_1\right) + \log \pi_2 \left(\boldsymbol{u} \mid \theta_2\right),$$

- así, la expresión para maximizar la verosimilitud marginal es:

$$\mathcal{L}(\beta, \boldsymbol{\theta}) = \int \exp[f(\beta, \boldsymbol{u}, \boldsymbol{\theta})] d\boldsymbol{u} \tag{1}$$

Laplace approximation

- Mediante la moda condicional de

$$\hat{\boldsymbol{u}}(\beta, \boldsymbol{\theta}) = \underset{u}{\text{argmax}}\, f(\beta, \boldsymbol{\theta}, \boldsymbol{u})$$

- `TMB` aproxima (1) usando la aproximación de Laplace para marginalizar los efectos aleatorios:

$$\mathcal{L}(\beta, \boldsymbol{\theta}) = \tilde{\mathcal{L}}(\beta, \boldsymbol{\theta}) = (2\pi)^{n/2} |\boldsymbol{\mathcal{H}}(\beta, \boldsymbol{\theta})|^{1/2} \exp[-f(\beta, \hat{\boldsymbol{u}}(\beta, \boldsymbol{\theta}), \boldsymbol{\theta})] \tag{2}$$

donde $\boldsymbol{\mathcal{H}}(\beta, \boldsymbol{\theta})$ es la Hessiana de $f(\beta, \hat{\boldsymbol{u}}(\beta, \boldsymbol{\theta}), \boldsymbol{\theta})$

El punto crítico aquí es obtener la Hessiana ($\boldsymbol{\mathcal{H}}(\boldsymbol{\beta}, \boldsymbol{\theta})$) pero este computo podemos realizarlo de forma sencilla mediante diferenciación automática.

Aproximación de Laplace en `TMB`

- Declarar la log-verosimilitud en el template de C++

$$f(\boldsymbol{\theta}, \boldsymbol{u}) = \log \pi_1 \left(\boldsymbol{y} \mid \theta_1), \boldsymbol{u}\right) + \log \pi_2 \left(\boldsymbol{u} \mid \theta_2\right)$$

- Dar valores iniciales a los parámetros fijos del modelo $\boldsymbol{\theta}_0$ y para los efectos aleatorios $\boldsymbol{u}_0$
- "Inner optimization"

$$\hat{\boldsymbol{u}} = \underset{\boldsymbol{u}}{\operatorname{argmax}}\, f((\boldsymbol{\theta}_0, \boldsymbol{u}))$$

- Calcular la aproximación de Laplace para la verosimilitud marginal de los efectos fijos

$$\log \tilde{\mathcal{L}}(\boldsymbol{\theta}_0; \boldsymbol{y}) \approx f(\boldsymbol{\theta}_0, \hat{\boldsymbol{u}}) - \frac{1}{2}\log(|\boldsymbol{H}|)$$

- "Outer optimization" (Repetir pasos 2 - 3). Aquí la "Outer optimization" es realizada en `R` usando la función y la gradiente proporcionada por `TMB`

¿Como crear en R un modelo con efecto aleatorio espacial (o espacio-temporal)?

- R-INLA $\Longrightarrow$ Rue et al. (2009), Blangiardo et al. (2013), Lindgren et al. (2015), Bakka et al. (2018), Cavieres and Nicolis (2018)
- TMB (Kristensen et al. (2015))

¿Como crear en R un modelo con efecto aleatorio espacial (o espacio-temporal)?

- R-INLA $\Longrightarrow$ Rue et al. (2009), Blangiardo et al. (2013), Lindgren et al. (2015), Bakka et al. (2018), Cavieres and Nicolis (2018)
- TMB (Kristensen et al. (2015))

¿Como crear en R un modelo con efecto aleatorio espacial (o espacio-temporal)?

- R-INLA $\Longrightarrow$ Rue et al. (2009), Blangiardo et al. (2013), Lindgren et al. (2015), Bakka et al. (2018), Cavieres and Nicolis (2018)
- TMB (Kristensen et al. (2015))

Caso de estudio

El erizo (*Loxechinus albus*) es uno de los recursos bentónicos más importantes en Chile (Guisado (1987), Moreno et al. (2011)). Dada su estructura de metapoblación espacial a gran escala, las subpoblaciones de erizo están interconectadas por dispersión larval, así que la recuperación de la abundancia local depende de la distancia y de característica propias de la pesquería dentro de ese dominio espacial.

Figura 2: Estructura común del erizo en “parches”. Reference: http://cocinafuturo.net/

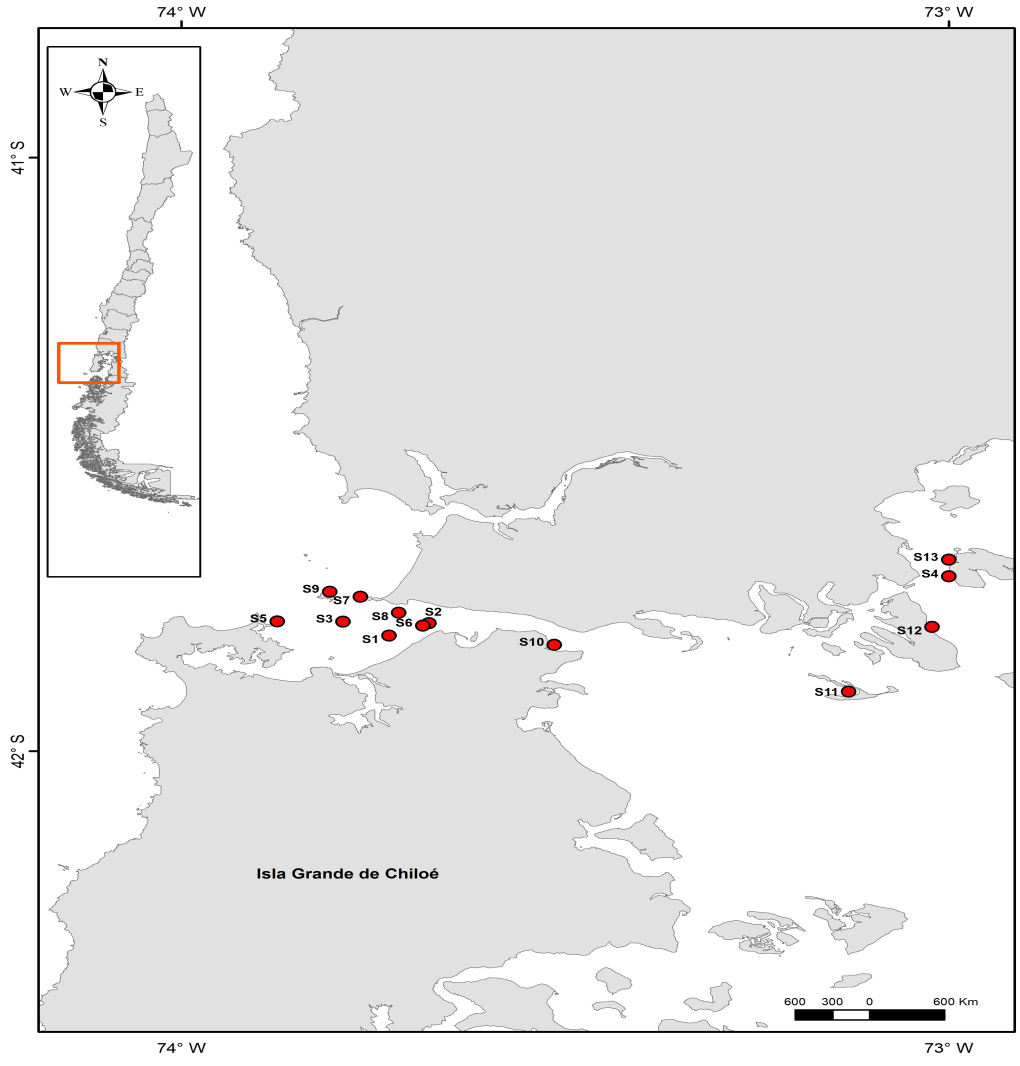


Figura 3: Sitios de pesca seleccionados en la pesquería del erizo al norte de Ancud

¿Cuál es el problema?

Actualmente este recurso es evaluado con un clásico modelo de stock assessment, el cual utiliza la CPUE como pieza clave de información en la obtención de un índice relativo de abundancia. Sin embargo, el índice estimado, no incorpora la dependencia espacial entre sitios de pesca.

¿Cuál es el problema?

Actualmente este recurso es evaluado con un clásico modelo de stock assessment, el cual utiliza la CPUE como pieza clave de información en la obtención de un índice relativo de abundancia. Sin embargo, el índice estimado, no incorpora la dependencia espacial entre sitios de pesca.

Datos disponibles

- Observaciones temporales: desde 1996 a 2016 ("Year"tratada como "factor").
- Sitios espaciales: 13 sitios de pesca ("sites"tratada como spatial random effects)
- Covariables: "Depth"(Profundidad), "Quarter"(temporada del año), y la variable "Market"(recurso vendido en fresco o a la industria (dummy 1 o 2)).

Modelos

Cuadro 1: Modelos propuestos

Modelos	Estructura
Lognormal	$(y_i \mid \boldsymbol{\theta}) \sim p(y_i \mid \eta_i, \boldsymbol{\theta})$
Spatial Lognormal	$(y_i \mid \boldsymbol{u}, \boldsymbol{\theta}) \sim p(y_i \mid \eta_i, \boldsymbol{\theta})$
Gamma	$(y_i \mid \boldsymbol{\theta}) \sim p(y_i \mid \eta_i, \boldsymbol{\theta})$
Spatial Gamma	$(y_i \mid \boldsymbol{u}, \boldsymbol{\theta}) \sim p(y_i \mid \eta_i, \boldsymbol{\theta})$

Resultados

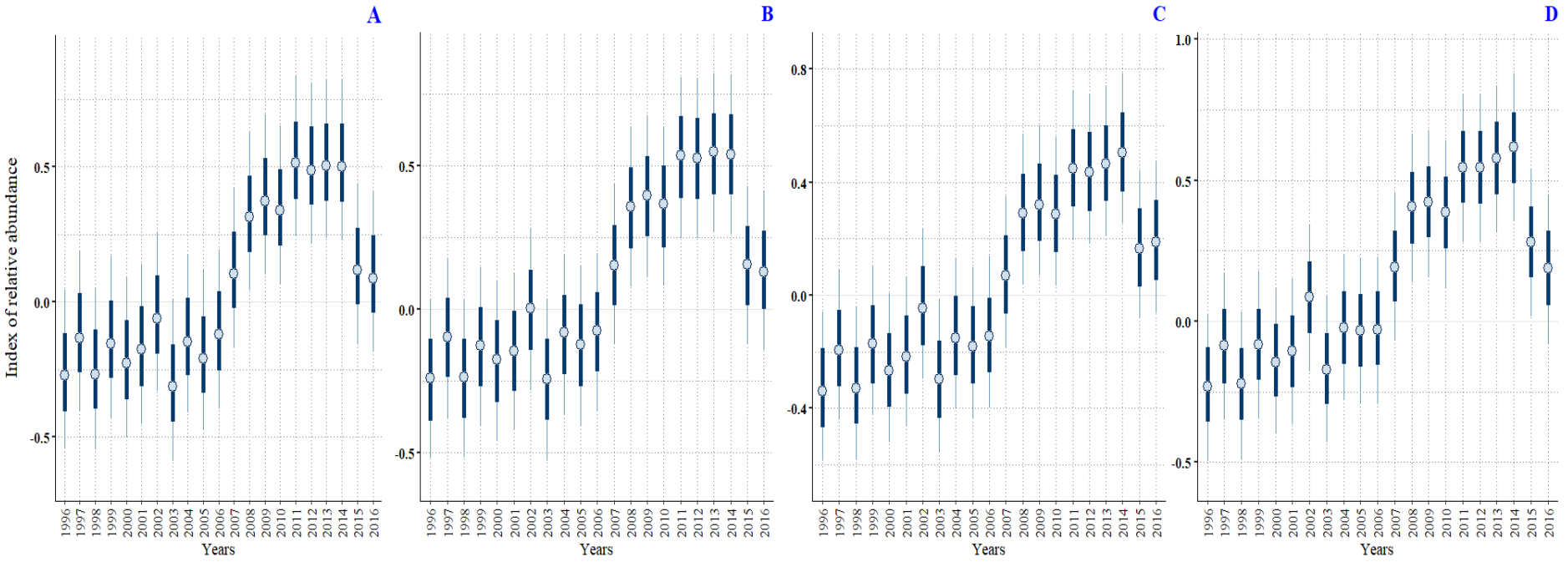


Figura 4: Comparación de los coeficientes (índices de abundancia relativa) estimados con modelo Lognormal, modelo Lognormal espacial, modelo Gamma y modelo Gamma espacial. Los puntos son los valores de los coeficientes y las barras gruesas son intervalos de incertidumbre calculados a partir de las posteriores obtenidas del método MCMC (90 % de intervalo creíble).

Cuadro 2: Comparación con el criterio LOO para cada modelo propuesto. `elpd_diff` mide la diferencia entre cada modelo en relación con el mejor $\widehat{elpd}_n$ (el modelo en la primera fila) y `se_diff` es el error estándar de la diferencia en `elpd_diff`.

Models	`elpd_diff`	`se_diff`
Spatial Gamma	0	0
Spatial Lognormal	-42	23
Lognormal	-88	18
Gamma	-125	28

Para evaluar los efectos potenciales de incluir sitios con solo un año de observaciones, realizamos dos comparaciones adicionales.

Cuadro 3: Comparaciones adicionales con LOO

Comparación excluyendo site 1		
Models	elpd_diff	se_diff
Spatial Gamma	0	0
Spatial Lognormal	-15	23
Gamma	-89	18
Lognormal	-95	28

Comparación excluyendo sites 1, 2, 3 y 8		
Models	elpd_diff	se_diff
Spatial Lognormal	0	0
Spatial Gamma	-5	22
Lognormal	-70	16
Gamma	-81	28

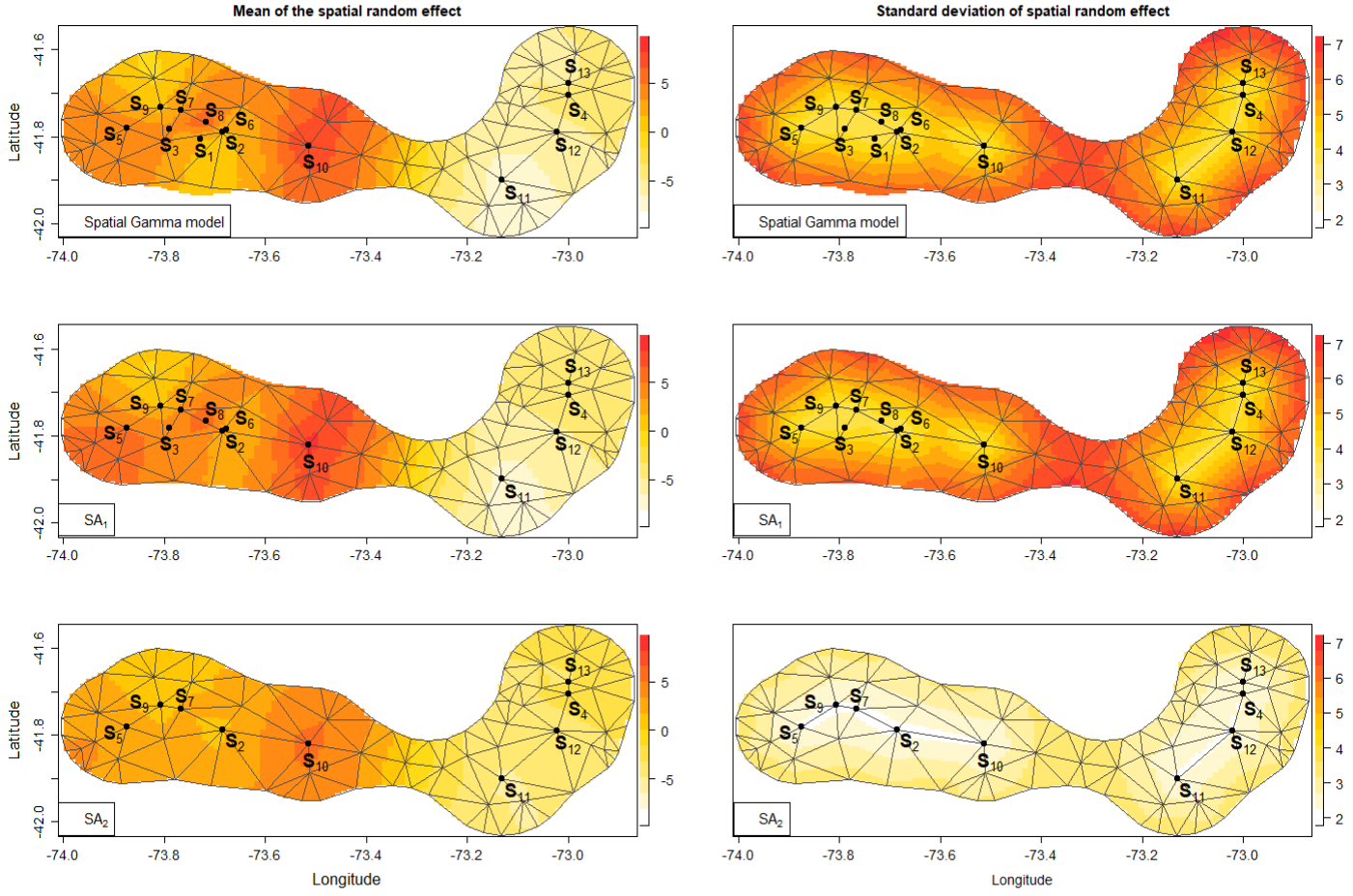


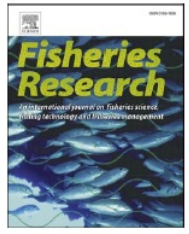
Figura 5: Media (izquierda) y desviación estándar (derecha) del campo espacial estimado por el modelo espacial Gamma, para el primer (SA_1) y segundo (SA_2) análisis de sensibilidad respectivamente.

- Esta investigación fue realizada en la Universidad de Aalto en Finlandia (pasantía), supervisada por Aki Vehtari.
- Cole C Monnahan (NOAA)

Contents lists available at ScienceDirect

Fisheries Research

journal homepage: www.elsevier.com/locate/fishres



Accounting for spatial dependence improves relative abundance estimates in a benthic marine species structured as a metapopulation

Joaquin Cavieres [a,*], Cole C. Monnahan [b], Aki Vehtari [c]

[a] Instituto de Estadística, Facultad de Ciencias, Universidad de Valparaíso, Valparaíso, Chile
[b] Resource Ecology and Fisheries Management, National Marine Fisheries Service (NOAA), Seattle, WA, United States
[c] Department of Computer Science, Aalto University, Finland

Cavieres et al. (2021)

Conclusiones

- `TMB` es un ambiente para crear modelos estadísticos de estructuras complejas
- Podemos integrar efectos aleatorios fácilmente y así maximizar la verosimilitud marginal
- Es muy flexible y computacionalmente eficiente
- Puede ser difícil de comprender dada la escritura en C++ del modelo
- Se pueden crear distintos tipos de modelos estadísticos, ya que si se puede escribir la verosimilitud, entonces `TMB` probablemente puede ajustar el modelo!

Gracias!

References I

Bakka, H., Rue, H., Fuglstad, G.-A., Riebler, A., Bolin, D., Illian, J., Krainski, E., Simpson, D., and Lindgren, F. (2018). Spatial modeling with r-inla: A review. *Wiley Interdisciplinary Reviews: Computational Statistics*, 10(6):e1443.

Blangiardo, M., Cameletti, M., Baio, G., and Rue, H. (2013). Spatial and spatio-temporal models with r-inla. *Spatial and spatio-temporal epidemiology*, 4:33–49.

Cavieres, J., Monnahan, C. C., and Vehtari, A. (2021). Accounting for spatial dependence improves relative abundance estimates in a benthic marine species structured as a metapopulation. *Fisheries Research*, 240:105960.

References II

Cavieres, J. and Nicolis, O. (2018). Using a spatio-temporal bayesian approach to estimate the relative abundance index of yellow squat lobster (cervimunida johni) off chile. *Fisheries research*, 208:97–104.

Guisado, C. (1987). Historia de vida, reproducción y avances en el cultivo del erizo comestible chileno l. albus (molina, 1782)(echinoidea; echinidae). *Manejo y desarrollo pesquero*, pages 59–68.

Kristensen, K., Nielsen, A., Berg, C. W., Skaug, H., and Bell, B. (2015). Tmb: automatic differentiation and laplace approximation. *arXiv preprint arXiv:1509.00660*.

Lindgren, F., Rue, H., et al. (2015). Bayesian spatial modelling with r-inla. *Journal of Statistical Software*, 63(19):1–25.

References III

Moreno, C. A., Molinet, C., Díaz, P., Díaz, M., Codjambassis, J., and Arévalo, A. (2011). Bathymetric distribution of the chilean red sea urchin (loxechinus albus, molina) in the inner seas of northwest patagonia: Implications for management. *Fisheries Research*, 110(2):305–311.

Rue, H., Martino, S., and Chopin, N. (2009). Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations. *Journal of the royal statistical society: Series b (statistical methodology)*, 71(2):319–392.

Skaug, H. J. and Fournier, D. A. (2006). Automatic approximation of the marginal likelihood in non-gaussian hierarchical models. *Computational Statistics & Data Analysis*, 51(2):699–709.