Posterior Approximation for Case-Crossover Models with Structured Additive Predictors

Alex Stringer

2019-04-05

Summary and state of project

1 Introduction

1.1 Motivation and proposed work

When studying the association of mortality with short-term exposures to risk factors, for example air pollution or extreme temperature, retrospectively-sampled observational data are common. It would usually be unethical to perform a randomized experiment in such a setting and performing a prospective sample would under-sample the case group, as deaths attributable to exposure to any single risk factor are often rare. The most practical method of data collection is therefore to sample from those known to have died due to exposure during some fixed time period and compare their exposure at time of death to exposure at one or more "control times", when they did not die. This is an especially attractive sampling scheme if the risk factors under consideration are the types of things that are already being recorded as part of large administrative databases available to the researcher, like weather or air pollution data.

The case-crossover design allows for modelling the effect of exposure on mortality for data sampled in this way. In such a design, each subject's contribution to the likelihood consists of the probability of them dying when they did, and not dying at any of the control times considered. This gives the associated probability model an explicit connection to stratified Cox Proportional Hazards regression. Current methods of fitting case-crossover models make use of this equivalence and as a result, there is no single, dedicated, documented software package for fitting these models with linear and smooth terms, and random effects. Since these models are being used in medical and epidemiological research, a single source of methodology and application of these models represents an overdue contribution to the literature.

In this paper, we will consider the case-crossover model in a Bayesian context, and develop an approximation-based method of inference for the posterior distribution of the exposure effects. This will allow a single comprehensive interface for fitting case-crossover models with any and all of linear and smooth terms and random effects, with multiple different covariance structures allowing for structured dependence between observations; time series models, and random-walk smoothing; and principled uncertainty estimates in all cases. A well-documented software package will accompany this research, the result being a one-stop shop for practitioners who want to fit case-crossover models to their data.

1.2 Examples

Maclure (1991) introduces the case-crossover design as a method for studying the hypothesis that myocardial infarctions (essentially heart attacks) do not occur at random for individuals, but are influenced by short-term exposure to risk factors in the time immediately before the event. For example, are short-term bursts of anger, physical exertion, or over-eating associated with a positive increase in risk of heart attacks? The key point here is that while the risk of *long-term* exposure to these risk factors is well-established, the specific motivation for this design is to determine the association of the event of interest with *short-term* exposure. This allows the practitioner to assess whether, all else equal between them, an individual who over-eats or experiences a bout of intense anger has a *temporarily* increased risk of experiencing a heart attack over an individual who does not experience these events.

Redelmeier and Tibshirani (1997) utilized the case-crossover design for determining the relative risk of drivers having a serious accident while using a cell phone compared to while not. They sampled their data by interviewing subjects at motor vehicle collision centres in the Greater Toronto Area, and comparing their cell phone use ("exposure") in the time leading up to their accidents to their cell phone use on a previous day when they undertook a similar amount of driving, but when they did not experience an accident. Compare this to how else these data could have been obtained—by asking some people to talk on the phone while driving and some to not and seeing who crashed (randomized experiment; highly unethical), or by randomly sampling individuals and asking if they crashed on two chosen days and comparing their cell phone usage (prospective; will result in very sparse data on crashes). The case-crossover design provides a method for answering this research question using a relatively cost-effective and ethical method of data collection.

More recently, Fu et al. (2018) consider the effect of short-term exposure to both extreme and moderate temperatures on mortality in India. Using a large, nationally-representitive dataset that was not collected for this specific purpose, the authors were able to associate short term exposure to extremely hot, but also moderately cold, temperatures to mortality. This illustrates an advantage of the case-crossover design; modern data is often collected on a large scale with no *specific* purpose with the expectation that researchers will anlayse it in support of numerous different goals. Whereas in the previous examples, there was choice about how to collect the data, with the type of sampling amenable to analysis using the case-crossover design being the most feasible, here there is no choice, as the data is already collected. The case-crossover design here enabled the authors to associate mortality with exposure to moderately cold temperatures, a novel finding with mitigation policy implications.

1.3 Existing methods of fitting

There are several methods available in the `R` language for fitting case-crossover models. To reiterate, the purpose of this work is to provide a *unified, principled*, and *documented* framework for fitting these models with a combination of linear and smooth terms, and different random effects structures. To provide context, here we illustrate how case-crossover methods may currently be fit to data, and the drawbacks of each. One global drawback that can't be overstated is that none of the following approaches are well documented; with the exception of `survival::clogit`, they all exist as examples buried away in documentation for other functions or online support lists. All make use of the equivalent functional forms of the likelihoods for the case-crossover model and Cox's stratified proportional hazards regression.

In all of the below one-line code snippets, it is assumed the analyst is working with a dataframe `dat` containing multiple rows per subject with subject identifier `id`; an indicator `case` which equals 1 for exactly one row per subject and 0 else, and covariates `x1` and `x2`.

1.3.1 `survival::clogit`

The `survival` package provides a `clogit` function for fitting case-crossover (or more generally, conditional logistic) models. All this function does is prepare the data in an appropriate way and call `survival::coxph`. The drawbacks are

- **Restricted model class**: only linear terms are permitted.
- **Documentation** There is limited documentation, and no fully-worked examples.

To fit the model with linear terms, the analyst would run

```
clogit(case ~ x1 + x2 + strata(id),data = dat)
```

and receive back a `coxph` object with the usual available methods.

1.3.2 `mgcv::gam`

The `gam` function in the `mgcv` package, used for fitting generalized additive models, may be used to fit case-crossover models with smooth terms for the covariates, using the same trick as `survival::clogit`. Some drawbacks are:

- **Ridgidness of smoothing methodology**. There are a lot of options available within `mgcv::gam`, but each one requires separate knowledge and has parameters and options that need to be tuned. The resulting interface is intimidating and requires expert knowledge to apply. While it is easy to switch between different smoothing methods in the software interface, a non-expert user will lack the ability to make a principled choice between them.
- **Computational efficiency**. Running `mgcv::gam` with three smooth terms on a dataset with approximately $40,000$ rows takes around 11 minutes on a modern laptop. This is not prohibitive, but it's not convenient, and such a dataset and model are not at all unrepresentitive of the scale at which these models are fit in real world applications.
- **Documentation**. The only available documentation for fitting case-crossover methods in this way is found at the end of the help page for the `cox.ph` family, `?mgcv::cox.ph`. The `mgcv::gam` function itself is documented excellently, but in order to make use of this, the user has to have detailed knowledge of GAMs and the associated theory. In the common case where GAMs are being investigated because a model with only linear terms reults in unsatisfactory fit, this might not be the case, and might serve as a deterrent.

To fit a model with smooth terms, the analyst would run the somewhat confusing

```
mgcv::gam(
    cbind(rep(1,nrow(dat)),dat$id) ~ s(x1) + s(x2),
    data = dat,
    family = mgcv::cox.ph,
    weights = case
  )
```

and receive back an object of class `gam` with the usual available methods.

1.3.3 `R-INLA`

To fit the case-crossover models in a Bayesian framework with a structured additive predictor, the analyst can make use of the `R-INLA` software. INLA is described in detail in sections 2 and 3 as it will be the basis of our approximations. The software allows for fitting of linear and smooth terms and random effects, with a variety of structures. The drawbacks of this approach are:

- **Computationally intensive**. The speed and memory requirements to use INLA for Cox-PH models in general are prohibitive when there are many strata.
- **Documentation**. The INLA package is documented largely through an FAQ and extensive online support group containing user questions and developer responses. This makes using INLA at all a daunting task for a practitioner; to use it for a "trick" model like this is nearly impossible to figure out for someone without expert knowledge of both INLA and case-crossover designs.
- **Incompatability**. In raw form, the Cox Proportional Hazards regression model is not compatable with the INLA framework, because the likelihood for each death time depends on more than one element of the underlying "latent field". INLA uses tricks to get around this, which brings us back to the documentation drawback.

To fit a model with linear terms, the analyst would run the *very* confusing

```
survobj <- inla.surv(
  time = rep(1,nrow(dat)),
  event = dat$case
)

system.time({
  inla_clogit_mod <- inla(
    survobj ~ x1 + x2
    data = dat,
    family = "coxph",
    verbose = TRUE,
    control.hazard=list(model="rw1", n.intervals=2,strata.name = "id")
  )
})
```

All of INLA's flexibility could be used here to include models with smooth terms and random effects with different structures, *in theory*. In practice, when run on a dataset with around $40,000$ observations and the above three linear terms only, the model crashes; on a cloud virtual machine with 8 cores and 64 GB of RAM, there is not enough memory. The problem is that INLA uses a smoothing model to estimate the hazard function, which works well when there are no strata but fails when there are many strata (and hence many hazard functions) with a small number of observations in each. It is documented in the INLA support groups online that there is no way to turn this off, and that it is not a priority for future releases.

1.4 Outline of paper

The discussion in section 1.3 suggests that INLA would be the ideal framework in which to set our general treatment of case-crossover models with structured additive predictors. In section 2, we will develop background theory on case-crossover models and INLA. In section 3, we develop the approximation to the posterior distribution of the latent variables that we will use to fit the case-crossover model to data. Section 4 presents worked examples and section 5 concludes with a discussion.

2 Methodology

2.1 Case-Crossover model

In this paper, we parametrize the case-crossover model as a structured additive model. In this section, we explain the details of this parametrization and derive the likelihood for the model.

2.1.1 Structured additive models: the linear predictor

The case-crossover model is parametrized as a **structured additive model**. We start with a linear predictor:

$$\eta_{it} = \alpha_i(t) + \sum_{q=1}^{M_1} \beta_q z_{qi}(t) + \sum_{q=1}^{M_2} f_q(u_{qi}(t)) \tag{1}$$

This is really a *structured additive predictor*; we use the term "linear predictor" throughout to improve readability. This is the most general form allowable, containing unstructured smooth, structured smooth, and linear terms. Our notation is as follows:

1. Indices:
   (a) $i = 1, \ldots, n$ for subjects,
   (b) $t = 1, \ldots, T_i$ for time points.
2. Each subject is observed at times $t \in S_i \equiv \{1, \ldots, T_i\}$, referred to as the **referent frame**. A key feature of the case-crossover study is that S_i is *chosen by the experimenter* for each subject,
3. $\alpha_i(t)$ represents the subject-specific mortality propensity at time t. This is an unstructured random effect included to capture the haphazard variation in subjects' individual mortality propensities. It will be demonstrated that the experimenter must choose S_i for each subject such that these terms cancel out of the likelihood,
4. $z_{qi}(t)$ is the value of the q^{th} covariate, representing exposures to risk factors, $q = 1, \ldots, M_1$, on the i^{th} subject at the t^{th} observation time, which we choose to model **linearly** with cofficients β_q,
5. $u_{qi}(t)$ is the value of the q^{th} covariate, $q = 1, \ldots, M_2$, which we choose to model via **arbitrary smooth functions** f_q. These represent structured random effects, included to capture the structural changes in mortality propensity associated with exposure to these risk factors that are not well modelled by linear terms. In the Bayesian framework, we don't need to specify what they are, rather we put a prior on the joint distributions of their *realizations* at the observed covariates.
   - *Note:* the f_q terms may also be taken to be classical random effects; in the Bayesian structured additive model framework, no mathematical distinction is made between these and smooth terms.

The choice of referent frame S_i is highly important in order to identify the effects in the model. Specifically, the control days must be chosen so that they resemble the death day in all non exposure-related mortality risks, allowing us to consider $\alpha_i(t)$ to be approximately constant with respect to t for each subject. This will allow $\alpha_i \equiv \alpha_i(t)$ to cancel out of likelihood calculations, and corresponds to each subject "serving as their own control". This isn't as esoteric or unachievable as it sounds; examples of how this is done can be found in the three studies discussed in section 1. For example, for Redelmeier and Tibshirani (1997), "exposure" was "using a cell phone while driving", so the authors chose control days where the subjects drove a similar amount and for similar reasons as on their crash days. It wouldn't make sense to compare a day of heavy commuting into the city with a day of light errand-running in the suburbs in terms of overall crash risk, regardless of cell phone use. Other potential factors in this context could include the weather (snowy days may have more crashes) and so on.

2.1.2 The probability models

With a probability model linking the linear predictor to the mean response, inference is based on a partial likelihood formed by conditioning on the fact that one and only one death occurred for each subject. Because of this, there are two probability models that lead to identical inferences. The first is a binary logistic model. Denote by Y_{it} the indicator of whether subject i died at time t. With the linear predictor η_{it} defined in (1), our probability model is

$$\begin{aligned} Y_{it} &\sim \text{Bernoulli}(p_{it}) \\ \log\left(\frac{p_{it}}{1-p_{it}}\right) &= \eta_{it} \end{aligned} \tag{2}$$

We may also consider a Poisson model, where we model the count of deaths per subject directly:

$$\begin{aligned} Y_{it} &\sim \text{Poisson}(\lambda_{it}) \\ \log \lambda_{it} &= \eta_{it} \end{aligned} \tag{3}$$

In what follows, these two models will be seen to lead to the same partial likelihood for η_{it}, when conditioning is done in the appropriate way. The informal intuition for this is as follows. When conditioning on one and only one event occurring per subject, the binary logistic model is really a multinomial model, where subjects' contribution to the likelihood consists of the probability of them dying when they did, and *not* dying at any time before that. Hence the vector of indicators $Y_i = (Y_{i0}, \ldots, Y_{iT_i})$ is modelled as Multinomial$(1, p_{i0}, \ldots, p_{iT_i})$ and this is used to form the likelihood. Under the Poisson model, the same conditioning amounts to modelling the joint distribution of $(Y_{i0}, \ldots, Y_{iT_i}) | \sum_{t \in S_i} Y_{it} = 1$, which is known to be multinomial with parameters $(\lambda_{i0}, \ldots, \lambda_{iT_i}) / \sum_{t \in S_i} \lambda_{it}$. The result is (2) and (3) leading to the same inferences.

We now derive the conditional likelihood for the case crossover model. To provide context and understanding, the derivation is presented in detail for the Bernoulli model with a single control day and the math is then extended to multiple control days.

2.1.3 Binary logistic model: single control day

With a single control day, subjects' exposures are measured once at death, and then at one additional pre-death time. This makes these models nearly the same as the Matched Pairs models described by Cox (1970). Without loss of generality, let $t = 0$ represent the control day and $t = 1$ the case day (when they died) for each subject. Assume as well that the experimenter has chosen the control day to be "like" the case-day in terms of other unmeasured mortality risk, so that we may take $\alpha_i(t) \equiv \alpha_i$. We start with:

$$\begin{aligned} p_{it} &= P\left(Y_i = t | z\right) \\ \log\left(\frac{p_{it}}{1 - p_{it}}\right) &= \alpha_i + \eta_{it} \\ \implies p_{it} &= \frac{\exp(\alpha_i + \eta_{it})}{1 + \exp(\alpha_i + \eta_{it})} \\ \implies 1 - p_{it} &= \frac{1}{1 + \exp(\alpha_i + \eta_{it})} \end{aligned} \tag{4}$$

With $\alpha_i \equiv \alpha_i(t)$ taken to be constant in time, η_{it} represents the additional mortality risk over a subject's baseline when exposure is at $(z_i(t), u_i(t))$. Cox (1970) starts with something similar, taking the control observation to have logit-probability equal to α_i and the case observation having logit-probability equal to $\alpha_i + \Delta$, with Δ representing the treatment effect. This results in probability modelled as

$$p_{i1} = \frac{e^{\Delta}}{1 + e^{\Delta}} \tag{5}$$

In the derivation here, Δ corresponds to $\eta_{i1} - \eta_{i0}$, the difference in exposure risks at case vs control times. If η_{it} contained only linear terms, our construction here would be equivalent to performing a logistic regression on the differences in exposure on the case vs control days. Fundamental to our approach is its ability to handle smooth terms and linear terms simultaneously, which cannot be achieved by simply differencing the data prior to analysis.

The i^{th} subject's contribution to the likelihood, L_i, is the probability that they die when they did, given that they died exactly once. This is enforced by the retrospective sampling and is compactly written as $\{Y_{i0} = 0, Y_{i1} = 1 | Y_{i0} + Y_{i1} = 1\}$. To get the likelihood, first note that the left-hand event is nested within the right-hand event,

$$\{Y_{i0} = 0, Y_{i1} = 1\} \subset \{Y_{i0} + Y_{i1} = 1\} \tag{6}$$

so that the joint probability of these two events equals the marginal probability of the former. It then follows from the rules of conditional probability that

$$
\begin{aligned}
L_i &= P\left(Y_{i0}=0, Y_{i1}=1 \middle| Y_{i0}+Y_{i1}=1\right) \\
&= \frac{P\left(Y_{i0}=0, Y_{i1}=1, Y_{i0}+Y_{i1}=1\right)}{P\left(Y_{i0}+Y_{i1}=1\right)} \\
&= \frac{\left(\frac{\exp(\alpha_i+\eta_{i1})}{1+\exp(\alpha_i+\eta_{i1})}\right) \times \left(\frac{1}{1+\exp(\alpha_i+\eta_{i0})}\right)}{\left(\frac{\exp(\alpha_i+\eta_{i1})}{1+\exp(\alpha_i+\eta_{i1})}\right) \times \left(\frac{1}{1+\exp(\alpha_i+\eta_{i0})}\right) + \left(\frac{\exp(\alpha_i+\eta_{i0})}{1+\exp(\alpha_i+\eta_{i0})}\right) \times \left(\frac{1}{1+\exp(\alpha_i+\eta_{i1})}\right)} \\
&= \frac{\exp(\alpha_i+\eta_{i1})}{\exp(\alpha_i+\eta_{i0})+\exp(\alpha_i+\eta_{i1})} \\
&= \frac{\exp(\eta_{i1})}{\exp(\eta_{i0})+\exp(\eta_{i1})}
\end{aligned}
\tag{7}
$$

The subject-specific mortality risk α_i cancels, because subjects serve as their own controls. The form shown in (7) is the form that the likelihood is commonly presented in; multiplying through by $\exp(-\eta_{i0})$ on the top and bottom explicitly exposes the connection to Cox's Δ:

$$
\begin{aligned}
\frac{\exp(\eta_{i1})}{\exp(\eta_{i0})+\exp(\eta_{i1})} &= \frac{\exp(\eta_{i1}-\eta_{i0})}{1+\exp(\eta_{i1}-\eta_{i0})} \\
&= \frac{e^{\Delta}}{1+e^{\Delta}}
\end{aligned}
\tag{8}
$$

The interested reader may verify that the same conditional likelihood is obtained by starting with a Poisson model in which $\log \mathbb{E}(Y_{it}) = \eta_{it}$.

2.1.4 Multiple control days

The derivation for multiple control days is the same but less illuminating, simply due to the messier notation. Define for each subject the *referent scheme* $S_i = \{0, 1, \ldots, T_i\}$ as the labelling of all the days on which subject i was observed, with T_i indicating the death time (again, without loss of generality—these are just labels). The probability model and likelihood are constructed in exactly the same way:

$$
\begin{aligned}
L_i &= P\left(Y_{i0}=0, Y_{i1}=0, \ldots, Y_{iT_i}=1 \middle| Y_{i0}+Y_{i1}+\cdots+Y_{iT_i}=1\right) \\
&= \frac{P\left(Y_{i0}=0, Y_{i1}=0, \ldots, Y_{iT_i}=1, Y_{i0}+Y_{i1}+\cdots+Y_{iT_i}=1\right)}{P\left(Y_{i0}+Y_{i1}+\cdots+Y_{iT_i}=1\right)} \\
&= \frac{\frac{\exp(\alpha_i+\eta_{iT_i})}{1+\exp(\alpha_i+\eta iT_i)} \times \prod_{t=0}^{T_i-1} \frac{1}{1+\exp(\alpha_i+\eta_{it})}}{\sum_{t=0}^{T_i}\left(\frac{\exp(\alpha_i+\eta_{it})}{1+\exp(\alpha_i+\eta_{it})} \prod_{j=0, j\neq t}^{T_i} \frac{1}{1+\exp(\alpha_i+\eta_{it})}\right)} \\
&= \cdots \\
&= \frac{\exp(\eta_{iT_i})}{\sum_{t\in S_i}\exp(\eta_{it})}
\end{aligned}
\tag{9}
$$

Note the similarity in form to the likelihood for a Cox proportional hazards regression. The interpretation is similar too: the probability of dying on the day that you did, conditional on *not* dying on the other days that you could have. The full log-likelihood for the case crossover model is then, assuming independence between subjects:

$$
\begin{aligned}
\ell(\theta) &= \sum_{i=1}^{n}\left(\eta_{iT_i} - \log\left(\sum_{t\in S_i}\exp(\eta_{it})\right)\right) \\
&= -\sum_{i=1}^{n}\log\left(\exp(-(\eta_{iT_i}-\eta_{i0}) + \cdots + \exp(-(\eta_{iT_i}-\eta_{iT_{i-1}})\right)
\end{aligned}
\tag{10}
$$

The second statement, involving the differences, will be useful in dealing with the constraint that $\sum_{t=1}^{T_i} p_{it} = 1$ for each i.

2.2 Approximation methodology

2.2.1 INLA

Integrated Nested Laplace Approximations (INLA; Rue, Martino, and Chopin (2009)) is a deterministic method for approximating the marginal posterior distribution of a large number of latent effects in a Bayesian heirarchical model. INLA considers heirarchical Bayesian models of the following form:

$$\begin{aligned} Y_i|W_i, \theta_2 &\sim \pi(y_i|w_i, \theta_2), i = 1 \dots n \\ W|\theta_1 &\sim \text{Normal}\left(0, Q^{-1}(\theta_1)\right) \\ (\theta_1, \theta_2) \equiv \theta &\sim \pi(\theta) \end{aligned} \tag{11}$$

Y_i is the observed response, W_i is an element of the (unobserved) *latent field*, which could represent each subject's *risk* towards whatever event Y_i represents, and $W = (W_1, \dots, W_n)$. θ collects all hyperparameters, which include those upon which the precision matrix Q depends, and those upon which the likelihood depends. In the implementation, little distinction is made between the two sources of hyperparameters, but a major distinction is drawn between θ and W.

The objects of primary interest here are the marginal posteriors of the elements of the latent field given the data, $\pi(w_i|y_i)$, for which INLA provides fast, scalable approximations. In a structured additive model, W is the collection of all the terms in the linear predictor, plus the linear predictors themselves. Specifically, INLA adds a noise variable onto the linear predictor itself,

$$\eta_{it} = \sum_{q=1}^{M_1} \beta_q z_{qi}(t) + \sum_{q=1}^{M_2} f_q(u_{qi}(t)) + Z_{it} \tag{12}$$

with $Z_{it} \overset{i.i.d.}{\sim} \text{Normal}(0, \tau^{-1})$, with τ set to some large constant (the default in INLA is e^{15}). The latent field is the ordered collection

$$W = (\eta_{11}, \dots, \eta_{nT_n}, f_1(u_{11}(1)), \dots, f_{M_2}(u_{M_2 n}(T_n)), \beta_1, \dots, \beta_{M_1}) \tag{13}$$

of all linear terms and evaluations of smooth terms. This is given a joint Gaussian prior, with precision matrix Q. The key factor that makes the computations feasible is that Q is a very sparse matrix. The "full" INLA proceeds by finding a set of integration points $\theta_1, \dots, \theta_K$ that interpolate the posterior $\pi(\theta|y)$, computing an efficient approximation to $\pi(W_i|\theta_k, y)$, and then computing

$$\pi(W_i|y) = \int \pi(W_i, \theta|y) d\theta \approx \sum_{k=1}^{n} \pi(W_i|\theta_k, y)\pi(\theta_k|y)\Delta_k \tag{14}$$

for each i, where Δ_k are weights chosen so that the resulting approximation integrates (sums) to 1.

The full INLA is quite complex. Much of this complexity is added in order to make the resulting approximations more accurate in smaller samples. Since the primary motivation of the case-crossover model is to analyze large administrative databases, "higher-order" accuracy isn't a main concern in this work. In the next section, we develop an INLA-inspired approximation that retains the computational efficiency and generality of the INLA approach to approximating the posterior of interest in (11), but is simple enough for the end-user to understand.

3 Proposed Approach

3.1 Overview

We first note that the case-crossover model with a structured additive predictor can be written in a heirarchical form *nearly* compatible with (11):

$$\begin{aligned} \pi(y_i|W, S_i) &= \frac{\exp(\eta_{iy_i})}{\sum_{t\in S_i}\exp(\eta_{it})} \\ W|\theta &\sim N(0, Q^{-1}(\theta)) \\ \theta &\sim \pi(\theta) \end{aligned} \tag{15}$$

Comparing with (11), there are two differences, the first trivial and the second not:

1. The likelihood doesn't have any hyperparameters θ, which simplifies the notation,
2. Importantly: $\pi(y_i|W, S_i, z)$ depends on *more than one* element of W.

Notwithstanding the drawbacks discussed in section 1.3.3, this latter difference explains why we can't just fit the model with the existing INLA software; it's not directly compatible. The key quantities of interest for inference are

1. $\pi(W|y)$, the *posterior distribution of the latent field*, and
2. $\pi(\theta|y)$, the *posterior distribution of the hyperparameters.*

Our approach for obtaining these quantities is to obtain a Gaussian approximation for $\pi(W|y)$, centred at the posterior mode $\hat{W} = \text{argmax}_W \pi(W, \theta|y) = \text{argmax}_W \pi(y|W, \theta)\pi(W|\theta)$, which is itself a function of θ. We use a plug-in estimate of θ for this purpose, obtained as $\hat{\theta} = \text{argmax}_\theta \pi(\theta|y)$. This itself requires an approximation for $\pi(\theta|y)$, which we then also use for inference. To readers familiar with the R-INLA software, our approach is analagous to `strategy='gaussian'` and `int.strategy='eb'` in `INLA::inla`. We now elaborate on this approach.

3.2 Explicit expressions for the precision matrix

So far in the description of the method, the precision matrix $Q = \text{var}(W|\theta)^{-1}$ of the Gaussian prior on the latent field is a mysterious object. Here we construct an explicit expression for it. Begin by writing the linear predictor η in m-dimensional vector form, $\eta = (\eta_{11}, \ldots, \eta_{n,T_n})$, giving

$$\eta = X\beta + AU + Z \tag{16}$$

Here,

1. X is a (dense) design matrix as in a usual linear regression,
2. β is a vector of regression coefficients, given a $\beta \sim \text{Normal}(0, \Sigma_\beta)$ prior,
3. A is a *sparse* random effect design matrix, assigning random effects U_k to observations, as in a usual mixed model,
4. $U \sim \text{Normal}(0, \Sigma_U)$ is the vector of random effects, and

5. $Z \sim \text{Normal}(0, \tau^{-1} I)$ is this aformentioned extra noise term.

In the notation of section 2, U is the vector of $f_q(u_{qi}(t))$ "smooth" terms. This is an artifact of the Bayesian structured additive regression paradigm; no mathematical distrinction is made between classical "random effects" and smooth terms. Note the flexibility of this construction: Σ_U can be specified to include random-walk smoothing, time series structures, correlated random effects, and so on.

To actually get Q, now, directly use properties of jointly normal random variables. We want to characterize

$$W = \begin{pmatrix} \eta \\ U \\ \beta \end{pmatrix} \sim \text{Normal}\left(0, Q^{-1}\right) \tag{17}$$

The relation (16) specifies the joint (unconditional) distribution of (U, β), and the conditional distribution of $\eta | U, \beta$, as follows:

$$\begin{aligned} \eta | U, \beta &\sim \text{Normal}\left(AU + X\beta, \tau^{-1} I\right) \\ \begin{pmatrix} U \\ \beta \end{pmatrix} &\sim \text{Normal}\left(0, \begin{pmatrix} \Sigma_U & 0 \\ 0 & \Sigma_\beta \end{pmatrix}\right) \end{aligned} \tag{18}$$

from which we obtain:

$$\begin{aligned} \begin{pmatrix} U \\ \beta \\ Z \end{pmatrix} &\sim \text{Normal}\left(0, \begin{pmatrix} \Sigma_U & 0 & 0 \\ 0 & \Sigma_\beta & 0 \\ 0 & 0 & \tau^{-1} I \end{pmatrix}\right) \\ \begin{pmatrix} \eta \\ U \\ \beta \end{pmatrix} &\stackrel{d}{=} \begin{pmatrix} A & X & I \\ I & 0 & 0 \\ 0 & I & 0 \end{pmatrix} \begin{pmatrix} U \\ \beta \\ Z \end{pmatrix} \equiv V \begin{pmatrix} U \\ \beta \\ Z \end{pmatrix} \\ &\sim \text{Normal}\left(0, V \begin{pmatrix} \Sigma_U & 0 & 0 \\ 0 & \Sigma_\beta & 0 \\ 0 & 0 & \tau^{-1} I \end{pmatrix} V^T\right) \\ \sim \text{Normal}\left(0, \begin{pmatrix} A\Sigma_U A^T + X\Sigma_\beta X^T + \tau^{-1} I & A\Sigma_U & X\Sigma_\beta \\ \Sigma_U A^T & \Sigma_U & 0 \\ \Sigma_\beta X^T & 0 & \Sigma_\beta \end{pmatrix}\right) &\equiv \text{Normal}\left(0, \Sigma\right) \end{aligned} \tag{19}$$

The precision matrix is then obtained as

$$Q = \Sigma^{-1} = \begin{pmatrix} \tau I & -\tau A & -\tau X \\ -\tau A^T & \Sigma_U^{-1} + \tau A^T A & \tau A^T X \\ -\tau X^T & \tau X^T A & \Sigma_\beta^{-1} + \tau X^T X \end{pmatrix} \tag{20}$$

Though this expression looks messy, we note a few things:

1. A is sparse and X is dense; X has a small number of columns, one for each linear term in the model. Combining these observations, we note that the dense matrices X, X^T, $A^T X$, $X^T A$, and $X^T X$ are of very small dimension, and the *sparse* matrices A, A^T, $A^T A$ and I are all of large dimension. The result is that Q is very large and very sparse. Of course if the linear terms are mostly categorical variables then X may also be sparse.
2. This is the whole story. Models with different linear and random effects/smooth term structures may have different X, A, Σ_β and Σ_U, but the form for Q remains the same across models in this framework. The exception is when there are no linear, or no smooth terms in the linear predictor. In this case, the form of Q is the same, with the appropriate rows/columns removed.

We will use this Q in the approximations to the posteriors of the latent field and the hyperparameters.

3.3 Gaussian approximation to the posterior of the latent field

To begin, we seek a Gaussian approximation to $\pi(W|y,\theta) = \pi(y|W,\theta)\pi(W|\theta)$, $\tilde{\pi}_G(W|y,\theta)$. Since $\pi(W|\theta)$ is already Gaussian, this will amount to a quadratic approximation to the log-likelihood. We have

$$\begin{aligned}\pi(W|y,\theta) &\propto \exp\left(-\frac{1}{2}W^TQW + \log\pi(y|W,\theta)\right)\\ &\approx \exp\left(-\frac{1}{2}W^TQW + -\frac{1}{2}(W-\hat{W})^TC(W-\hat{W})\right)\\ &\propto \exp\left(-\frac{1}{2}(W-\hat{W})^T(Q+C)(W-\hat{W})\right)\end{aligned} \tag{21}$$

where $\log\pi(y|W,\theta) \approx \exp(-\frac{1}{2}(W-\hat{W})^TC(W-\hat{W}))$ is a quadratic approximation to the log-likelihood centred at an appropriate point $\hat{W}$; we may add a $\hat{W}Q\hat{W}$ term into the exponent, as this is constant with respect to W and so gets absorbed into the integrating constant. $C = -\frac{\partial^2}{\partial W\partial W^T}\log\pi(y|W,\theta)|_{W=\hat{W}}$ is minus the Hessian of the log-likelihood evaluated at $\hat{W}$. The mode of $\pi(W|y,\theta)$ is this point $\hat{W}$, which we have to find. Note that $\hat{W} \equiv \hat{W}(\theta)$ is a function of θ, as it is the mode of $\pi(W|y,\theta)$ for a particular θ.

As the latent field W is usually very large-dimensional, optimizing (21) is potentially computationally prohibitive. The key is the sparsity of Q, which ensures a single Newton step requires only solving one sparse system. To see this, take a quadratic approximation to $\log\pi(y|W,\theta) = \sum_{i=1}^n \log\pi(y_i|W,\theta)$ at some initial value W_0,

$$\log\pi(y|W,\theta) \approx \log\pi(y|W_0,\theta) + (W-W_0)^T\nabla\log\pi(y|W_0,\theta) - \frac{1}{2}(W-W_0)^TC_0(W-W_0), \tag{22}$$

where $\nabla\log\pi(y|W_0,\theta)$ is the gradient evaluated at $W = W_0$ and $C_0 = -\frac{\partial^2}{\partial W\partial W^T}|_{W=W_0}$ is the negative Hessian evaluated at the same point. Now, note that the quantity $-W_0^TQW_0$ doesn't depend on W, so we may add/subtract it in the constant, rearranging to obtain:

$$\log\pi(W|y,\theta) \approx -\frac{1}{2}(W-W_0)^T(Q+C_0)(W-W_0) + (W-W_0)^T\nabla\log\pi(y|W_0,\theta) \tag{23}$$

The gradient of this equation:

$$\nabla_W\log\pi(W|y,\theta) = -(Q+C_0)(W-W_0) + \nabla\log\pi(y|W_0,\theta) \tag{24}$$

is set to 0 to obtain the update equation:

$$(Q+C_0)(W_1-W_0) = \nabla\log\pi(y|W_0,\theta) \tag{25}$$

The matrix on the LHS is sparse and analytic expressions for Q, C, and $\nabla\log\pi(y|W_0,\theta)$ are available. Hence, the only computational cost of an iteration is solving a sparse linear system. The ability to do this quickly is precisely why INLA works.

An important point here is that while this algorithm is *efficient* by construction, there is absolutely nothing that guarantees that it is *stable*. Instability is a well-known reason not to naively apply Newton's method in practice, especially in high-dimensions. In the implementation, we make use of the highly efficient and robust `IPOPT` software of Wachter and Biegler (2006), which utilizes sparse matrix algorithms internally, retaining the problem-specific efficiency present here. However, the construction (25) will be useful during the *hyperparameter* optimization, to be discussed next.

The final Gaussian approximation to $\pi(W|y,\theta)$ is

$$\pi_G(W|y,\hat{\theta}) = \text{Normal}\left(\hat{W}(\hat{\theta}), Q(\hat{\theta}) + \hat{C}\right) \tag{26}$$

where $\hat{\theta} = \text{argmax}_\theta \tilde{\pi}(\theta|y)$ and $\hat{W}(\hat{\theta}) = \text{argmax}_W \tilde{\pi}_G(W|y,\hat{\theta})$. We now turn to the task of finding $\hat{\theta}$.

3.4 Approximation to the posterior of the hyperparameters

To begin, we describe the approximation to $\pi(\theta|y)$, $\tilde{\pi}(\theta|y)$. Taking the entire vector of latent variables (θ, W), the desired quantity can be viewed as the *marginal* posterior of θ. We use the method of Tierney and Kadane (1986). This is based off of Laplace approximations; see Appendix A for a brief review.

3.4.1 Marginal posteriors: the method of Tierney and Kadane

In our present situation, we have data y, a large latent field W, and a small number of additional latent parameters θ, and our object of interest is $\pi(\theta|y) = \int \pi(\theta, W|y)dW$. Tierney and Kadane (1986) describe a method of obtaining highly accurate approximations to this density using Laplace approximations. We have

$$\pi(\theta|y) = \int_{\mathbb{R}^m} \pi(\theta, W|y)dW = \frac{\int_{\mathbb{R}^m} \pi(\theta, W, y)dW}{\int_{\mathbb{R}^k}\int_{\mathbb{R}^m} \pi(\theta, W, y)dW d\theta} \tag{27}$$

where k is the (small) dimension of θ and m is the (large) dimension of W. Tierney and Kadane (1986) apply a Laplace approximation to both the numerator and denominator separately. Their motivation in doing this is accuracy; they argue that while the error in naively applying this method is still relative $O(n^{-1})$, broken down it is actually the product of two terms, one for the integration constant and one for the "functional form". When renormalized via numerical integration, the relative error is $O(n^{-3/2})$, which is quite accurate. It is also imporant for the reader who is not an expert on Laplace approximations to observe that because the errors are *relative*, not *absolute*, the approximation here does not depend on the size of the quantity being approximated. When estimating tail probabilities and other small quantities this presents a substantial advantage over sampling-based methods, which typically have *additive* error rates like $O(n^{-1/2})$.

To actually get this approximation, do Laplace twice. For the numerator, we have

$$\begin{aligned}\int_{\mathbb{R}^m} \pi(\theta, W, y)dW &= \pi(\theta)\int_{\mathbb{R}^m} \pi(W, y|\theta)dW \\ &\approx \pi(\theta)(2\pi)^{m/2}|Q(\theta) + \hat{C}(\theta)|^{-1/2}\pi(\hat{W}(\theta), y|\theta)\end{aligned} \tag{28}$$

where Q and C are the usual precision and hessian matrices, the latter being evaluated at the conditional mode $\hat{W}(\theta) = \text{argmax}\pi(W, y|\theta)$ already found. It bears repeating that while this expression is complicated, every quantity needed here has already been computed in finding the Gaussian approximation to $\pi(W, y|\theta)$.

For the denominator, it's a bit more complicated. We're now interested in $(\hat{W}, \hat{\theta})$, the conditional mode of $\pi(\theta, W, y)$ with respect to both W and θ jointly. Similarily, we need the $m + k$-dimensional Hessian matrix D of (the log of) this function, evaluated at this mode. The approximation is then

$$\int_{\mathbb{R}^k}\int_{\mathbb{R}^m} \pi(\theta, W, y)dW d\theta \approx (2\pi)^{(m+k)/2}|\hat{D}|^{-1/2}\pi(\hat{\theta}, \hat{W}, y) \tag{29}$$

The full approximation of Tierney and Kadane (1986) is then

$$\pi(\theta|y) \approx \pi(\theta)(2\pi)^{m/2}|Q(\theta) + \hat{C}(\theta)|^{-1/2}\pi(\hat{W}(\theta), y|\theta) \times (2\pi)^{-(m+k)/2}|\hat{D}|^{1/2}\pi(\hat{\theta}, \hat{W}, y)^{-1} \tag{30}$$

3.4.2 The INLA approach

Equation (30) looks very complicated, because it is very complicated. We immediately note two things:

1. Most of the terms in (30) are constant with respect to θ.
2. All of the terms which are *not* constant with respect to θ are already available as a result of the Gaussian approximation.

With these points in mind, INLA uses the approximation

$$\pi(\theta|y) \approx \frac{\pi(y|\hat{W}(\theta),\theta)\pi(\hat{W}(\theta)|\theta)\pi(\theta)}{\pi_G(W|y,\theta)}|_{W=\hat{W}(\theta)} \tag{31}$$

Rue, Martino, and Chopin (2009) claim that this is equal to the TK approximation, "after renormalization", which means "up to constants". For purposes of optimization, the constants don't actually matter, and for inference, the low dimension of θ allows renormalization via numerical integration.

This claim of equivalence is true, but it's not obvious as stated. To see it, note the following:

1. The $\pi(\hat{W}(\theta), y|\theta)$ term is common to (31) and (30), because $\pi(\hat{W}(\theta), y|\theta) = \pi(y|\hat{W}(\theta),\theta)\pi(\hat{W}(\theta)|\theta)$.
2. The Gaussian approximation in the denominator of (31) is a Gaussian approximation *with mean* $\hat{W}(\theta)$ *evaluated at* this same point, $\hat{W}(\theta)$. *So the exponential term cancels*—and the result is just $\pi_G(W|y,\theta)|_{W=\hat{W}(\theta)} = (2\pi)^{-m/2}|Q(\theta)+\hat{C}(\theta)|^{1/2}$.
3. Everything else in both expressions is constant.

With these observations, (31) evaluates to

$$\begin{aligned}\log\tilde{\pi}(\theta|y) &\approx c + \log\pi(y|W,\theta) + \log\pi(\hat{W}|\theta) + \log\pi(\theta) - \frac{1}{2}\log|Q(\theta)+\hat{C}(\theta)| \\ &= c + \log\pi(y|W,\theta) + \log\pi(\theta) + \frac{1}{2}\log|Q(\theta)| - \frac{1}{2}\hat{W}(\theta)^T Q(\theta)\hat{W}(\theta) - \frac{1}{2}\log|Q(\theta)+\hat{C}(\theta)|\end{aligned} \tag{32}$$

We use the approximation (32) for $\tilde{\pi}(\theta|y)$ in our approach.

There is a subtle detail here that equation (32) glosses over: the conditional mode $\hat{W}(\theta)$ depends on θ. This is extremely inconvenient; the objective $\pi(\theta|y)$ will be evaluated many times during the optimization, so having each evaluation itself depend on a very high-dimensional optimization (directly and through $\hat{C}$) sounds like a computational deal-breaker. This is where (25) comes in: for a *small change* in θ, $\hat{W}(\theta)$ is not expected to change too much. Hence in moving from point θ_1 to point θ_2, performing a single or small number of Newton steps to find $\hat{W}(\theta_2)$ *starting from* $\hat{W}(\theta_1)$ is efficient, and likely to be stable. Hence in the implementation, each evaluation of $\pi(\theta|y)$ involves a Newton-based optimization, but by making use of a lookup table of pre-computed $\theta \mapsto \hat{W}(\theta)$ mappings, stability and efficiency can be ensured.

3.5 Marginal variances

With the approximation (26) in hand, posterior marginal means, standard deviations, and quantiles can be obtained from $\hat{W}$ and $Q(\hat{\theta}) + \hat{C}$. Specifically,

$$W_i|y \dot{\sim} \text{Normal}\left(\hat{W}, \left(Q(\hat{\theta})+\hat{C}\right)^{-1}_{ii}\right) \tag{33}$$

gives the user everything they need. However, we don't want to compute the entire inverse of $Q(\hat{\theta}) + \hat{C}$, as this matrix is quite large in general. All we need is the diagonals of its inverse. To compute these efficiently, we again use sparsity. Compute the cholesky decomposition of $Q(\hat{\theta}) + \hat{C}$:

$$Q(\hat{\theta}) + \hat{C} = \hat{L}\hat{L}^T \tag{34}$$

where $\hat{L}$ is sparse and upper-triangular. Sparsity of the LHS means this operation can be performed in $O(n)$ time. It's much easier to invert $\hat{L}$ than $Q(\hat{\theta}) + \hat{C}$, because its upper triangular structure allows solving by back-substitution. Doing this yields the diagonal elements of the inverse of the precision matrix as

$$\ldots \tag{35}$$

Rue, Martino, and Chopin (2009) specify a recursive equation for computing the desired diagonal elements from $\hat{L}$, which bypasses the need to invert $\hat{L}$ explicitly. Since computing marginal variances only needs to be done once, (35) is sufficient for our purposes.

With (33) in hand, we have our full approximation to the marginal posteriors of $W_i|y$, and hence for all regression coefficients, random effects, smooth terms, and of course the linear predictor itself. We now turn back to the case-crossover model.

3.6 Calculations for the case-crossover model

3.6.1 The likelihood

With the approximation methodology worked out in general for models of the form (11), all that we would need to specify for a given implementation is the likelihood and its Hessian, and a prior for the hyperparameters. The latter is application-specific, since what θ contains depends on the structure of η. The likelihood, on the other hand, is model-specific. For the case-crossover, however, we have the added complication of having the non-linear constraint that $\sum_{t=1}^{T_i} p_{it} = \sum_{t=1}^{T_i} \frac{e^{\eta_{it}}}{1+e^{\eta_{it}}} = 1$ for each i. We call this "non-linear" in reference to it being non-linear in η_{it}, not p_{it}. From (15), we have

$$\begin{aligned} \log \pi(y|W) &= -\sum_{i=1}^{n} \log\left(\exp(-(\eta_{iT_i} - \eta_{i0}) + \cdots + \exp(-(\eta_{iT_i} - \eta_{iT_{i-1}})\right) \\ &= -\sum_{i=1}^{n} \log\left(\exp(-(\Delta_{i0}) + \cdots + \exp(-(\Delta_{i(T_i-1)})\right) \end{aligned} \tag{36}$$

where we have defined $\Delta_{it} = \eta_{iT_i} - \eta_{it}$ for $i = 0, \ldots, T_i - 1$ as the *difference* in log-odds of exposure for control day t compared to the case day T_i for subject i. Inference is done on the Δ_{it} as a manner for dealing with the non-linear constraint. We may compute η_{it} from $\Delta_i = (\Delta_{i1}, \ldots, \Delta_{iT_i})$ by solving the non-linear equation

$$\frac{e^{\eta_{iT_i}}}{1 + e^{\eta_{iT_i}}} = 1 - \sum_{t=0}^{T-1} \frac{e^{\eta_{iT_i}}}{e^{\Delta_{it}} + e^{\eta_{iT_i}}} \tag{37}$$

numerically for η_{iT_i}, and then subbing in $\eta_{it} = \eta_{iT_i} - \Delta_{it}$. Note though that in given applications, the Δ_{it} or a simple function of them are often the objects of primary inferential interest anyways, used to make statements such as "covariate xyz is associated with an increase of odds of mortality by a factor of $e^{\Delta_{it}}$", and so on.

3.6.2 The precision matrix under a non-linear constraint

This does, however, change the precision matrix and Hessian. Convenient to our re-parameterization is that, while our *constraint* is non-linear, Δ_i is a linear transformation of η_i,

$$\begin{aligned}\Delta_i &= \begin{pmatrix} \eta_{iT_i} - \eta_{i0} \\ \vdots \\ \eta_{iT_i} - \eta_{iT_{i-1}} \end{pmatrix} = D_i \eta_i \\ D_i &= \begin{pmatrix} -1 & 0 & \cdots & 0 & 1 \\ 0 & -1 & \cdots & 0 & 1 \\ & & \ddots & & \\ & & & -1 & 1 \end{pmatrix}\end{aligned} \tag{38}$$

where the differencing matrix $D_i \in \mathbb{R}^{(T_i-1)\times T_i}$ has rank $T_i - 1$. The functional form of each D_i is exactly the same; all that changes is the dimension. This is applied to the whole vector η, which gives

$$\Delta = \begin{pmatrix} \Delta_1 \\ \vdots \\ \Delta_n \end{pmatrix} = \begin{pmatrix} D_1\eta_1 \\ \vdots \\ D_n\eta_n \end{pmatrix} = \begin{pmatrix} D_1 & & \\ & \ddots & \\ & & D_n \end{pmatrix} \begin{pmatrix} \eta_1 \\ \vdots \\ \eta_n \end{pmatrix} \equiv D\eta \tag{39}$$

D is the block-diagonal matrix of differencing matrices D_i, with dimension $(\sum_{i=1}^n T_i - n) \times \sum_{i=1}^n T_i$ and full row rank. It remains to reconstruct the latent field, from the joint distribution of (Δ, U, β). Let

$$W_c = \begin{pmatrix} \Delta \\ U \\ \beta \end{pmatrix} \tag{40}$$

be the latent field under our non-linear constraint on η. Let $X_c = DX$ and $A_c = DA$. Following the construction of the original W, we have

$$\begin{aligned}\Delta | U, \beta = D\eta | U, \beta &\sim \text{Normal}\left(A_c U + X_c \beta, \tau^{-1} D D^T\right) \\ \begin{pmatrix} U \\ \beta \end{pmatrix} &\sim \text{Normal}\left(0, \begin{pmatrix} \Sigma_U & 0 \\ 0 & \Sigma_\beta \end{pmatrix}\right)\end{aligned} \tag{41}$$

from which we obtain

$$\begin{aligned}\begin{pmatrix} U \\ \beta \\ Z \end{pmatrix} &\sim \text{Normal}\left(0, \begin{pmatrix} \Sigma_U & 0 & 0 \\ 0 & \Sigma_\beta & 0 \\ 0 & 0 & \tau^{-1} I \end{pmatrix}\right) \\ \begin{pmatrix} \Delta \\ U \\ \beta \end{pmatrix} &\stackrel{d}{=} \begin{pmatrix} A_c & X_c & D \\ I & 0 & 0 \\ 0 & I & 0 \end{pmatrix} \begin{pmatrix} U \\ \beta \\ Z \end{pmatrix} \equiv V \begin{pmatrix} U \\ \beta \\ Z \end{pmatrix} \\ &\sim \text{Normal}\left(0, V \begin{pmatrix} \Sigma_U & 0 & 0 \\ 0 & \Sigma_\beta & 0 \\ 0 & 0 & \tau^{-1} I \end{pmatrix} V^T\right) \\ \sim \text{Normal}\left(0, \begin{pmatrix} A_c \Sigma_U A_c^T + X_c \Sigma_\beta X_c^T + \tau^{-1} D D^T & A_c \Sigma_U & X_c \Sigma_\beta \\ \Sigma_U A_c^T & \Sigma_U & 0 \\ \Sigma_\beta X_c^T & 0 & \Sigma_\beta \end{pmatrix}\right) &\equiv \text{Normal}\left(0, \Sigma\right)\end{aligned} \tag{42}$$

The precision matrix is then obtained as

$$Q = \Sigma^{-1} = \begin{pmatrix} \tau \left(DD^T\right)^{-1} & -\tau \left(DD^T\right)^{-1} A_c & -\tau \left(DD^T\right)^{-1} X_c \\ -\tau A_c^T \left(DD^T\right)^{-1} & \Sigma_U^{-1} + \tau A_c^T \left(DD^T\right)^{-1} A_c & \tau A_c^T \left(DD^T\right)^{-1} X_c \\ -\tau X_c^T \left(DD^T\right)^{-1} & \tau X_c^T \left(DD^T\right)^{-1} A_c & \Sigma_\beta^{-1} + \tau X_c^T \left(DD^T\right)^{-1} X_c \end{pmatrix} \tag{43}$$

This has a very similar form to (20), with the replacement of A and X with their differences versions $A_c = DA$, $X_c = DX$, and the addition of the $\left(DD^T\right)^{-1}$ terms. In order to understand if and how this affects the sparsity of Q, we need to examine the structure of D, DD^T, and $(DD^T)^{-1}$. D is a large block-diagonal matrices with D_i on the blocks. This is quite sparse, as each D_i has the form (38). The "transpose-crossproduct" DD^T of D is the block-diagonal matrix of transpose-crossproducts of the D_i:

$$DD^T = \begin{pmatrix} D_1 D_1^T & & \\ & \ddots & \\ & & D_n D_n^T \end{pmatrix}; (DD^T)^{-1} = \begin{pmatrix} (D_1 D_1^T)^{-1} & & \\ & \ddots & \\ & & (D_n D_n^T)^{-1} \end{pmatrix} \tag{44}$$

Each $D_i D_i^T$ is dense with dimension $T_i - 1$, equal to the number of control days. However, simple computation shows that its inverse has a particular form,

$$D_i D_i^T = J + I \tag{45}$$

where I is the identity and $J = 11^T$ is a matrix of ones of the appropriate dimension. Since $\text{rank}(J) = 1$ regardless of its dimension, DD^T is a low-rank update of the identity and its inverse can be computed via a rank-one update to the identity. Specifically, using the Woodbury (or Sherman-Morrison) formula,

$$(D_i D_i^T)^{-1} = \left(I_{T_i - 1} + 11^T\right)^{-1} = I - 1(1 + 1^T 1)^{-1} 1^T = I - \frac{1}{T_i} J \tag{46}$$

This matrix is cheap to compute. The resulting $(DD^T)^{-1}$ is still quite sparse, as each dense block is of dimension equal to the number of control days (i.e. $T_i - 1$). In the uncommon event that the number of control days is so large as to render computation of $(DD^T)^{-1}$ impractical or unfeasible, note that the factors $\frac{1}{T_i} J$ will then be very small, and a sparse approximation can be obtained simply by removing this term, approximating $(DD^T)^{-1} \approx I$. In common applications, it is not expected that the number of control days is so large as for this to be necessary.

3.6.3 The Hessian

The last "custom" ingredient for this model is (minus) the Hessian of the log-likelihood. With the difference-based construction (40), we compute the Hessian with respect to Δ_i, not η_i. Noting the form of the log-likelihood as a sum-of-sums,

$$\log \pi(y|W) = -\sum_{i=1}^{n} \log \left(\exp(-(\Delta_{i0}) + \cdots + \exp(-(\Delta_{i(T_i - 1)})\right) \tag{47}$$

we see that C will be a block-diagonal matrix with each block depending on $\Delta_{i0}, \ldots, \Delta_{i(T_i - 1)}$. We have

$$\begin{aligned} \frac{\partial}{\partial \Delta_{it}} \log \pi(y_i|W) &= \frac{e^{-\Delta_{it}}}{e^{-\Delta_{i0}} + \cdots + e^{-\Delta_{i(T_i - 1)}}} \\ \frac{\partial^2}{\partial \Delta_{it}^2} \log \pi(y_i|W) &= -\frac{e^{-\Delta_{it}}}{e^{-\Delta_{i0}} + \cdots + e^{-\Delta_{i(T_i - 1)}}} \left(1 - \frac{e^{-\Delta_{it}}}{e^{-\Delta_{i0}} + \cdots + e^{-\Delta_{i(T_i - 1)}}}\right) \\ \frac{\partial^2}{\partial \Delta_{it} \Delta_{is}} \log \pi(y_i|W) &= -\frac{e^{-(\Delta_{it} + \Delta_{is})}}{e^{-\Delta_{i0}} + \cdots + e^{-\Delta_{i(T_i - 1)}}} \end{aligned} \tag{48}$$

With

$$C_i = \begin{pmatrix} \frac{\partial^2}{\partial \Delta_{i0}^2} \log \pi(y_i|W) & \frac{\partial^2}{\partial \Delta_{i0}\Delta_{i1}} \log \pi(y_i|W) & \cdots & \frac{\partial^2}{\partial \Delta_{i0}\Delta_{i(T_i-1)}} \log \pi(y_i|W) \\ \frac{\partial^2}{\partial \Delta_{i0}\Delta_{i1}} \log \pi(y_i|W) & \frac{\partial^2}{\partial \Delta_{i1}^2} \log \pi(y_i|W) & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2}{\partial \Delta_{i0}\Delta_{i(T_i-1)}} \log \pi(y_i|W) & \cdots & \cdots & \frac{\partial^2}{\partial \Delta_{i(T_i-1)}^2} \log \pi(y_i|W) \end{pmatrix} \tag{49}$$

as the negative Hessian of the i^{th} subject's contribution to the log-likelihood with respect to Δ_i, the full C-matrix is

$$C = \begin{pmatrix} C_1 & & & & & \\ & \ddots & & & & \\ & & C_n & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{pmatrix} \tag{50}$$

The zeroes at the end are the derivatives with respect to the U_i and β. This may seem peculiar, but it is an artifact of the structure of η_i as a linear predictor "plus noise". The latent field is a function of *both* η_{it} and U_i, β; changing these latter two terms while holding η_{it} fixed then does not change the likelihood, resulting in zero derivatives. This is consistent with the construction of the equivalent matrix in INLA. Each C_i is dense, but of small dimension (again, equal to the number of control days). Further, comparing (50) and (44), we see that the additional denseness of Q and C occur *in the same pattern*, so the added inefficiency does not compound when adding these two matrices.

Appendix A: review of Laplace approximations

Common in statistics are real-value functions $f_n : \mathbb{R}^k \to \mathbb{R}$ depending on some parameter n (usually a sample size), with the following properties:

1. f_n has a single global maximum, or if many local extrema, one local max which "dominates" (is bigger than the rest).
2. As $n \to \infty$, f_n becomes more and more peaked around this dominating mode, and hence becomes better and better approximated by a quadratic centred at this point.

The best example is the log-likelihood function $f_n(x) \equiv \sum_{i=1}^n \ell(x_i)$; as n gets larger, the CLT ensures that this function becomes more and more peaked about its mean. Approximating a log-likelihood using a quadratic is equivalent to approximating a density using a Gaussian, which partly explains why this latter approach is so popular.

Laplace approximation is closely related to this idea, in the context of approximating *integrals* of such functions. Suppose for such a function we want to estimate

$$I = \int_{\mathbb{R}^k} e^{f_n(x)} dx \tag{51}$$

Laplace's method works as follows:

1. Find the mode of f_n, call it $\hat{x}$.
2. Taylor-expand f_n about $\hat{x}$, obtaining $f_n(x) \approx f_n(\hat{x}) - \frac{1}{2}(x-\hat{x})^T H(\hat{x})(x-\hat{x})$. H is minus the Hessian, $H = -\partial^2/\partial x \partial x^T f_n(x)$, evaluated at $\hat{x}$.
3. The exponential of this is a Gaussian density, which we know how to integrate.

The result is the *Laplace approximation to the integral*, given by

$$\hat{I} \approx (2\pi)^{k/2} |H(\hat{x})|^{1/2} e^{f_n(\hat{x})} \tag{52}$$

A good question to ask is: why? Why Taylor expand inside the expoential, why not the whole thing? An obvious reason is computational: typically the kinds of functions that concentrate around their modes are made up of sums of things, and these are much easier to work with than their exponentials, which are products. But a better, and subtler, reason is how the approximation errors work. When taking a raw Taylor expansion of a function, the error is *additive*, of the form $|f(x) - \hat{f}(x)| = O((x-\hat{x})^3)$, for example. But when taking the *exponential* of a Taylor approximation, the error becomes *relative*, which can offer huge advantages when e.g. using an approximate density to approximate a tail probability, or some other really small number. It can be shown through tedious calculation that the error in the Laplace approximation is $\hat{I} = I(1 + O(n^{-1}))$. Compared to the *additive* $O(n^{-1/2})$ error rate attained by most sampling procedures, Laplace approximations are much better suited to approximating quantities which are likely to be small, like tail probabilities.

References

Cox, David. 1970. *Analysis of Binary Data.* Methuen; Co.

Fu, Sze Hang, Antonio Gasparrini, Peter S. Rodriguez1, and Prabhat Jha. 2018. "Mortality Attributable to Hot and Cold Ambient Temperatures in India: A Nationally Representative Case-Crossover Study." *PLoS Med* 15 (7). doi:https://doi.org/10.1371/journal. pmed.1002619.

Maclure, Malcolm. 1991. "The Case-Crossover Design: A Method for Studying Transient Effects on the Risk of Acute Events." *American Journal of Epidemiology* 185 (11): 144–53. doi:10.1093/aje/kwx105.

Redelmeier, Donald A., and Robert Tibshirani. 1997. "Association Between Cellular Telephone Calls and Motor Vehicle Collisions." *The New England Journal of Medicine* 336 (7): 453–58.

Rue, Håvard, Sara Martino, and Nicolas Chopin. 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): 319–92.

Tierney, Luke, and Joseph B. Kadane. 1986. "Accurate Approximations to Posterior Moments and Marginal Densities." *Journal of the American Statistical Association* 81 (393).

Wachter, A., and L. T. Biegler. 2006. "On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming." *Mathematical Programming* 106 (1): 25–57.