CAV_explorative

Data

The dataset employed regards the counts of accesses to gender-based violence support centers in the Apulia region by residence municipality of the women victims of violence during 2022. `R` codes to generate the dataset are in the R script (posted here) which this report is based on.

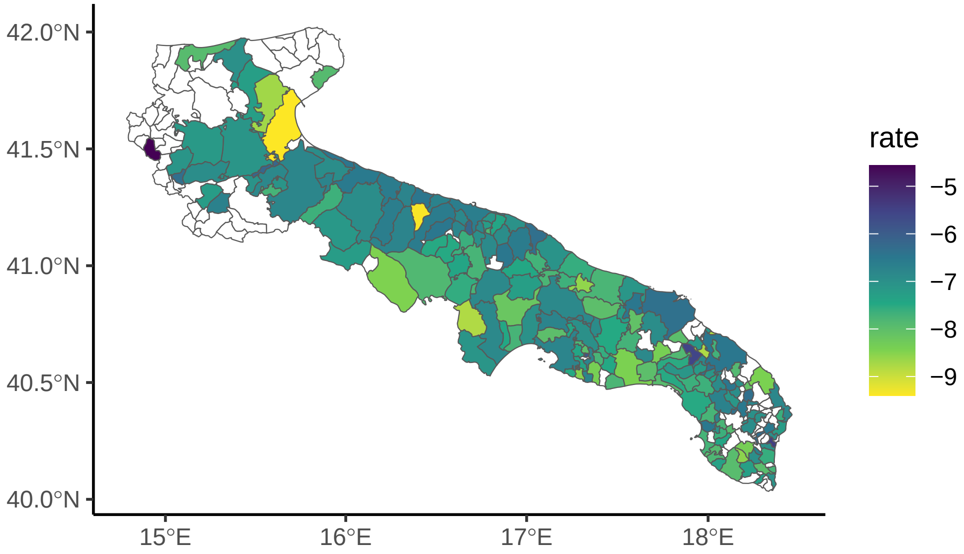
Here, we only take into account the violence reports which support centers actually take charge of, at the risk of underestimating the counts of gender-based violence cases. This choice is driven by the need of avoiding duplicated records, since e.g. it may happen that a support center redirects a victim to another support center.

In order to avoid singletons in the spatial structure of the dataset, we removed the Tremiti Islands from the list of municipalities included (0 accesses to support centers in 2022).

Therefore, the municipality-level dataset in scope consists of 256 observations.

We can only take into account the accesses to support centers for which the origin municipality of victims is reported. Therefore, the total count of accesses in scope is 2259. Among these accesses, 1516 were taken charge of.

Here, we plot the log-access rate per residence municipality, i.e. the logarithm of the ratio between access counts and female population. Blank areas correspond to municipalities from which zero women accessed support centers (82 municipalities).



Explanatory variables

Our target is explaining the number of accesses to support centers, y, defined at the municipality level, on the basis of a set of candidate known variables.

We model y via a simple Poisson GLM.

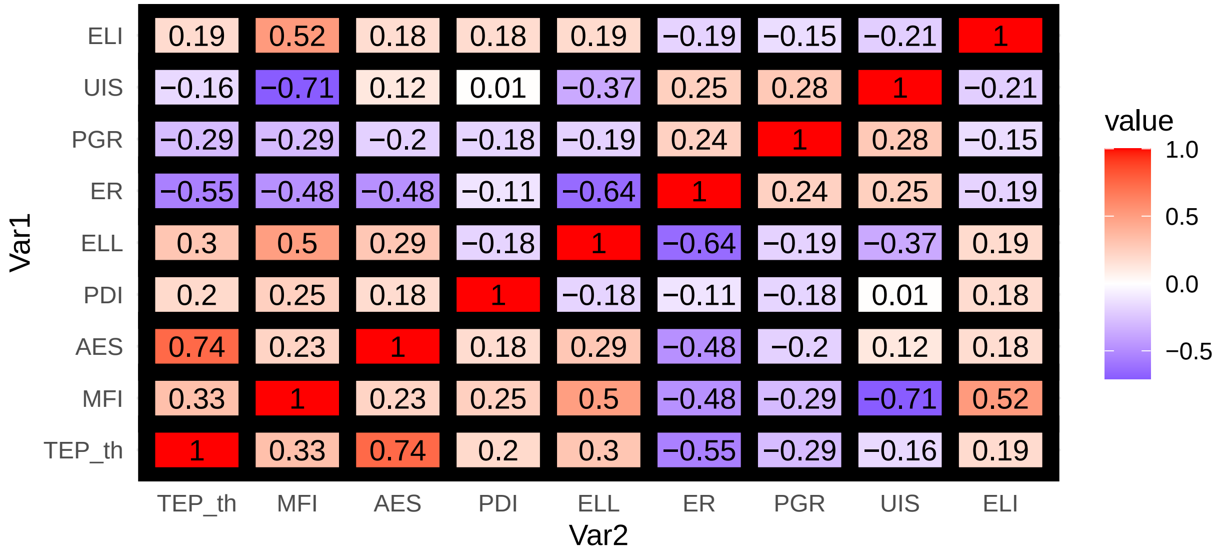
We have at disposal a number of candidate explanatory variables, which include the distance of a municipality from the closest support center and a set of variables measuring social vulnerability under different dimensions; these latter covariates are provided by the ISTAT.A more detailed description of these covariates is in this excel metadata file.

All covariates are scaled to have null mean and unit variance.

- d, i.e. the theoretical distance of each municipality from the municipality hosting the support center competent for that municipality. Distance is measured by road travel time in minutes. Fo r instance, the support center designated for the municipality of Adelfia (province of Bari, 3rd municipality in the dataset) is located in Triggiano (BA). Then, $d_3^{(T)}$ denotes the travel time between Adelfia and Triggiano (17 minutes).

  In ‘R‘, d is encoded as `TEP_th`.
- AES, the distance from the closest infrastructural pole, always measured in travel time. It is encoded as ‘AES‘
- ‘MFI‘, i.e. the decile of municipality vulnerability index.
- ‘PDI‘, i.e. the dependency index, i.e. population either ≤ 20 or ≥ 65 years over population in $[20 - 64]$ years.
- ‘ELL‘, i.e. the proportion of people aged $[25 - 54]$ with low education.
- ‘ER‘, i.e. employment rate among people aged $[20 - 64]$.
- ‘PGR‘, i.e. population growth rate with respect to 2011.
- ‘UIS‘, i.e. the ventile of the density of local units of industry and services (where density is defined as the ratio between the counts of industrial units and population).
- ‘ELI‘, i.e. the ventile of employees in low productivity local units by sector for industry and services.

First, we visualise the correlations among these explanatory variables:



Then, we implement a very simple forward selection algorithm. At each iteration, we add to the model the covariate allowing for the lowest BIC, until adding an additional covariate does not allow to reduce it anymore:

```
covariates <- colnames(X)[-1]
covs.in <- c()
```

```
BIC.min <- c()
while(length(covs.in) < length(covariates)){
  covs.out <- covariates[which(!covariates %in% covs.in)]

  BICs <- c()
  for(j in c(1:length(covs.out))) {
    formula.temp <- paste0("N_ACC ~ 1 + offset(log(nn)) +",
                          paste(covs.in, collapse = "+"),
                          "+", covs.out[j])
    mod.tmp <- glm(formula.temp, data = dd_con, family = "poisson")
    BICs[j] <- stats::BIC(mod.tmp)
  }
  BIC.min <- c(BIC.min, min(BICs))
  if(length(BIC.min)>1 && BIC.min[length(BIC.min)] >= BIC.min[length(BIC.min)-1]){
    break
  } else{
    covs.in <- c(covs.in, covs.out[which.min(BICs)])
  }
}

# Only two, yet quite correlated.
covs.in
```

```
## [1] "TEP_th" "AES"
```

As we see, the only two relevant covariates appear the distance indicators: distance from the closest support center and distance. These two variables have a high correlation (0.74).

OLS regression

We regress the counts of accesses y to support centers on the distance from the former. Both covariates are scaled to zero mean an unit variance.

$$\frac{y_i}{P_i} \mid \eta_i \sim \text{Poisson}(e^{\eta_i}) \quad \text{where} \quad \eta_i = \beta_0 + \beta_d d_i + \beta_{AES}\text{AES}_i \tag{1}$$

Where P_i is the female population of municipality i, and d_i can be either $d_i^{(T)}$ or $d_i^{(A)}$.

```
##
## Call:
## glm(formula = N_ACC ~ 1 + TEP_th + AES, family = "poisson", data = dd_con,
##     offset = log(nn))
##
## Coefficients:
##             Estimate Std. Error  z value Pr(>|z|)
## (Intercept) -7.49029    0.04374 -171.232  < 2e-16 ***
## TEP_th      -0.38633    0.04969   -7.775 7.55e-15 ***
## AES         -0.13625    0.03193   -4.267 1.98e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 729.12  on 255  degrees of freedom
## Residual deviance: 485.64  on 253  degrees of freedom
## AIC: 1058.2
##
## Number of Fisher Scoring iterations: 5
```

How do we interpret the regression coefficients? Keeping in mind we are working on the logarithm of the access rate, the standard deviations of explantory variables, both expressed in minutes, are the following:

```
# Distance from closest support center
attr(scale(dists_th$TEP_th), "scaled:scale")
```

```
## [1] 15.74582
```

```
# Distance from closest infrastructural pole
attr(scale(Indicators$AES), "scaled:scale")
```

```
## [1] 15.47489
```

We plot the log-residuals ε of the regression model in equation 1, defined as $\varepsilon := \ln y_i - \ln P_i - \ln \hat{y}_i$ being $\hat{y}_i$ the fitted value. Explanatory variable is theorical distance.

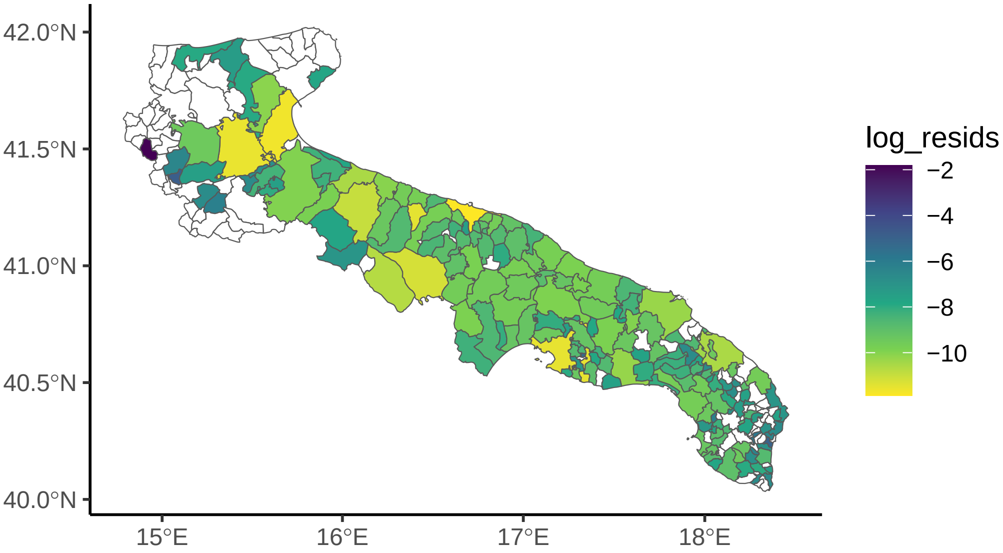


Figure 1: Log-residuals of glm regression using theorical distance as explanatory variable

```
## Warning in spdep::poly2nb(dd_con[nonzero_con, ]): neighbour object has 2 sub-graphs;
## if this sub-graph count seems unexpected, try increasing the snap argument.
```

Residuals may exhibit spatial structure. To assess it, we employ the Moran and Geary tests. Since

Please notice that log-residuals only take finite values across the 175 municipalities whose female citizens have reported at least one case of violence in 2022.

Additionally, this set of municipalities includes 2 singletons, which we remove to assess the value of the Moran and Geary statistics. Thus, we have defined the indexes set `nonzero_con` as the set of municipalities from which at least one case of gender-based violence has been reported, *and* which have at least one neighbouring municipalities from which at least one case of gender-based violence was reported.

```
spdep::moran.test(resids_glm_th[nonzero_con],
                  listw = spdep::nb2listw(nb_con_nonzero))
```

```
##
##  Moran I test under randomisation
```

```
##
## data:  resids_glm_th[nonzero_con]
## weights: spdep::nb2listw(nb_con_nonzero)
##
## Moran I statistic standard deviate = 5.6902, p-value = 6.344e-09
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##       0.302415738      -0.005813953       0.002934223
```

```
spdep::geary.test(resids_glm_th[nonzero_con],
                  listw = spdep::nb2listw(nb_con_nonzero))
```

```
##
##  Geary C test under randomisation
##
## data:  resids_glm_th[nonzero_con]
## weights: spdep::nb2listw(nb_con_nonzero)
##
## Geary C statistic standard deviate = 4.0358, p-value = 2.721e-05
## alternative hypothesis: Expectation greater than statistic
## sample estimates:
## Geary C statistic       Expectation          Variance
##       0.751958417       1.000000000       0.003777415
```

In both cases, we find evidence for spatial autocorrelation. However, we must stress out this result does not refer to all the regional territory, but only to a subset of all municipalities (173 over 257)

Spatial regression

Based on the autocorrelation evidence, though it has only been assessed for a subset of all municipalities, we try implementing a simple spatial model, say the intrinsic autoregressive:

$$\eta_i = \beta_0 + \beta_d d_i + \beta_{AES}\mathrm{AES}_i + z_i \tag{2}$$

Where

$$z_i \mid z_{-i} \sim N\left(\sum_{j=1}^{n} \frac{w_{ij}}{d_i} z_j \, , \frac{\sigma^2}{d_i}\right) \tag{3}$$

Here w_{ij} is a generic element of the neighbourhood matrix of Apulia's municipalities, tanking value 1 if two municipalities i and j are neighbours and zero if either i and j are not neighbours or $i = j$; d_i is the number of neighbours of municipality i, and σ^2 is the scale parameter of the spatial effect. $n = 256$ and the remainder of the model follows the same notation as eq. 1.

On a first stance, we will let the Laplace Approximation approximate posterior distributions of the quantities of interest (instead of e.g. relying on MCMC)

We set a PC-prior on the random effect to avoid overfitting. The `f()` argument sets the prior model on the spatial latent effect. Arguments `num.threads` and `internal.opt` are used for reproducibility. We use a PC-prior with rate parameter $\lambda = 1.5$, such that $\mathrm{Prob}(\sigma > \lambda) = 0.01$

```
cav_icar_INLA_th <- inla(
  N_ACC ~ 1 + TEP_th + AES + f(ID, model = "besag", graph = W_con,
                           scale.model = T, prior = "pc.prec", param = c(1.5, 0.01)),
  family = "poisson", offset = log(nn), data =dd_con,
  num.threads = 1, control.compute = list(internal.opt = F, cpo = T, waic = T),
  inla.mode = "classic", control.inla = list(strategy = "laplace"),
  verbose = F) # better
```

We compare the spatial and the nonspatial model (for coparability it has been re-run with R-INLA) through the Watanabe-Akaike information criterion

```
cav_nosp_INLA_th$waic$waic # Higher WAIC
```

```
## [1] 1064.348
```

```
cav_nosp_INLA_th$waic$p.eff # Low complexity
```

```
## [1] 8.605647
```

```
cav_icar_INLA_th$waic$waic # Lower WAIC
```

```
## [1] 958.8593
```

```
cav_icar_INLA_th$waic$p.eff # High complexity
```

```
## [1] 72.69854
```

Which suggests us the spatial model is preferable; model complexity increases (`p.eff` denotes the number of free/unconstrained parameters) but is still outweighter by better fitting. Here we show some summaries for the spatial model:

```
cav_icar_INLA_th$summary.fixed
```

```
##                   mean         sd 0.025quant   0.5quant  0.975quant       mode
## (Intercept) -7.6230980 0.06223487 -7.7493003 -7.6217593 -7.50479148 -7.6216469
## TEP_th      -0.3311004 0.08771165 -0.5044677 -0.3307689 -0.15964244 -0.3307749
## AES         -0.1760451 0.07353991 -0.3223210 -0.1754928 -0.03291102 -0.1754966
##                      kld
## (Intercept) 2.061461e-05
## TEP_th      5.250088e-08
## AES         5.990678e-08
```

We still see distance from the closest support center has a negative effect on the accesses. A tentative interpretation we may draw from this result is that the distance from support centers may tend to discourage women from reporting gender-based violence, and the occurrence of gender-based violence may be underestimated in areas distant from support centers.

The spatial effect is to be interpreted as the spatial variation not explained by model components. Here we show the posterior summaries of the marginal variance parameter (σ^2) of the spatial effect

```
inla.zmarginal(inla.tmarginal(
  fun = function(X) 1/X,
  marginal = cav_icar_INLA_th$marginals.hyperpar[[1]]))
```

```
## Mean            0.657288
## Stdev           0.182537
## Quantile  0.025 0.350284
## Quantile  0.25  0.526114
## Quantile  0.5   0.639436
## Quantile  0.75  0.76874
## Quantile  0.975 1.06491
```

And we plot the posterior expectations of the latent effect.

Now, the intrinsic autoregressive model is relatively simple to interpret and to implement, while also requiring the minimum number of additional parameter (either the scale or the precision). The drawback, however, is

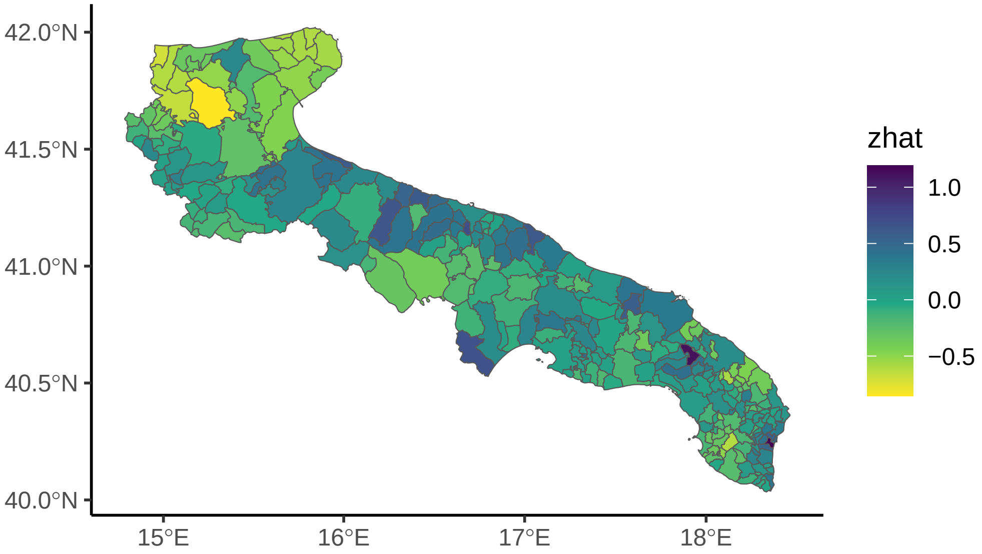


Figure 2: Posterior expectation of the spatial latent effect

that we assume a spatial autocorrelation coefficient equal to 1. When the autocorrelation is weak, setting an ICAR prior may be a form of misspecification.

A generalisation of this model is the PCAR (proper CAR), which introduces an autocorrelation parameter α:

$$z_i \mid z_{-i} \sim N\left(\sum_{j=1}^{n} \alpha \frac{w_{ij}}{d_i} z_j \, , \, \frac{\sigma^2}{d_i}\right) \tag{4}$$

Although it is not readily implemented in `R-INLA` (the `"besagproper"` effect is actually the Leroux model!) we may base the R code on the `INLAMSM` package:

```
inla.rgeneric.PCAR.model <-
  function (cmd = c("graph", "Q", "mu", "initial", "log.norm.const",
                                              "log.prior", "quit"), theta = NULL) {
  interpret.theta <- function() {
    alpha <- 1/(1 + exp(-theta[1L])) # alpha modelled in logit scale
    mprec <- sapply(theta[2L], function(x) {
      exp(x)
    })
    PREC <- mprec
    return(list(alpha = alpha, mprec = mprec, PREC = PREC))
  }
  graph <- function() {
    G <- Matrix::Diagonal(nrow(W), 1) + W
    return(G)
  }
  Q <- function() {
    param <- interpret.theta()
    Q <- param$PREC *
      (Matrix::Diagonal(nrow(W), apply(W, 1, sum)) - param$alpha * W)
    return(Q)
  }
  mu <- function() {
    return(numeric(0))
  }
  log.norm.const <- function() {
    val <- numeric(0)
    return(val)
  }
  log.prior <- function() {
    param <- interpret.theta()
    val <- -theta[1L] - 2 * log(1 + exp(-theta[1L]))
    # # PC prior
    val <- val + log(lambda/2) - theta[2L]/2 - (lambda * exp(-theta[2L]/2))
    # # Gamma(1, 5e-5), default prior:
    #val <- val + dgamma(exp(theta[2L]), shape = 1, rate = 5e-5, log = T) + theta[2L]
    # # Uniform prior on the standard deviation
    #val <- val - sum(theta[2L])/2 - k * log(2)
    return(val)
  }
  initial <- function() {
    return(c(0, 4))
  }
  quit <- function() {
    return(invisible())
  }
  if (as.integer(R.version$major) > 3) {
    if (!length(theta))
      theta = initial()
  }
  else {
    if (is.null(theta)) {
      theta <- initial()
    }
  }
  val <- do.call(match.arg(cmd), args = list())
  return(val)
}
PCAR.model <- function(...) INLA::inla.rgeneric.define(inla.rgeneric.PCAR.model, ...)
```

Then the model is run

```
cav_pcar_INLA_th <- inla(N_ACC ~ 1 + TEP_th + AES +
                           f(ID, model = PCAR.model(W = W_con, k = 1, lambda = 1.5)),
                         family = "poisson", offset = log(nn), data =dd_con,
                         num.threads = 1, control.compute =
                           list(internal.opt = F, cpo = T, waic = T),
                         inla.mode = "classic", control.inla = list(strategy = "laplace"),
                         control.predictor = list(compute = T),
                         verbose = T)
```

We compare the WAICs of the two spatial models

```
cav_icar_INLA_th$waic$waic # Higher WAIC
```

```
## [1] 958.8593
```

```
cav_icar_INLA_th$waic$p.eff
```

```
## [1] 72.69854
```

```
cav_pcar_INLA_th$waic$waic # Lower WAIC
```

```
## [1] 940.7756
```

```
cav_pcar_INLA_th$waic$p.eff # Complexity is not even raised
```

```
## [1] 71.39892
```

The lower WAIC suggests us that the PCAR, which does not even appear to be more complex although the additional parameter α. We show the posterior summaries of the regression coefficients

```
cav_pcar_INLA_th$summary.fixed
```

```
##                   mean         sd 0.025quant   0.5quant  0.975quant       mode
## (Intercept) -7.6187902 0.07249439 -7.7656954 -7.6173246 -7.48032377 -7.6172846
## TEP_th      -0.3634936 0.08235752 -0.5261945 -0.3631800 -0.20256749 -0.3631796
## AES         -0.1978873 0.06704965 -0.3307792 -0.1975221 -0.06703311 -0.1975143
##                      kld
## (Intercept) 4.183002e-06
## TEP_th      5.357745e-08
## AES         5.227461e-08
```

however, the spatial autocorrelation does not appear particularly strong. Additionally, the credible interval appears uncannily wide:

```
## Mean            0.470016
## Stdev           0.222425
## Quantile  0.025 0.0809099
## Quantile  0.25  0.290043
## Quantile  0.5   0.47005
## Quantile  0.75  0.646321
## Quantile  0.975 0.869732
```