Segmentation and territorial smoothing in property & casualty insurance ratemaking

The Ninth Montreal Industrial Problem Solving Workshop

Problem presented by: Desjardins Insurance

August 23, 2019

Review of the problem

- In property and casualty insurance ratemaking, the **geospatial** location is one of the most critical factors for **risk segmentation** purposes
- The fundamental goal of the workshop was to develop a statistical model allowing to incorporate a geospatial component, among other rating variables, for risk segmentation purposes, with the important business constraint that the model produce a **smoothed map of risk levels across territories**

Review of the problem

- The current method consists of a 4 step modelling process:
    - GLM $\rightarrow$ XGBOOST $\rightarrow$ MRF $\rightarrow$ GLM
- This approach is complex and lacks robustness
- Work was needed to find alternative approaches

Overview of approaches

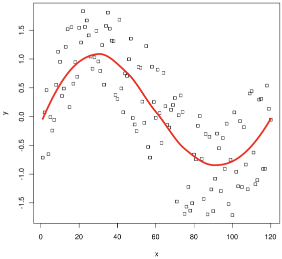
- There are numerous other ways in which geographic information can be incorporated into a model
- We focused on three methods:
  1. Geographically Weighted Poisson Regression
  2. Poisson Kriging
  3. Fused Lasso for Poisson GLMs

Disclaimer

- Due to the size of the data and its format (i.e. categorical), various processing were involved in rendering the data usable

Geographically Weighted Poisson Regression

- Wish to predict number of claims Y of a given client
- Develop local GLM model for $Y \sim \text{Poiss}(\lambda)$ at each point x
- Points nearby x have a greater influence in describing λ
- Appropriate choices may yield smooth parameters



Geographically Weighted Poisson Regression

- Model via

$$\log \lambda(x) = \hat{\beta}(x)^T \Phi(x)$$

  where $\hat{\beta}(x)$ is a local parameter estimator and $\Phi(x)$ are the features at x

- Estimators $\hat{\beta}(x)$ are found from a minimization problem

$$\hat{\beta}(x) = \text{argmin}_\beta - \sum_{i \in \text{Training}} w(x, x_i) \log \Pr(Y_i = y_i | x_i, \beta(x))$$

- Choose $w(x, y) = \exp(-\frac{||x-y||^2}{2\alpha^2})$ where α is a hyperparameter

Geographically Weighted Poisson Regression

- $\lambda(x)$ depends on α: select α by cross-validation
- For some penalty, seek to minimize

$$G(\alpha) = \sum_{i \in \text{Validation}} P_i$$

- Tried

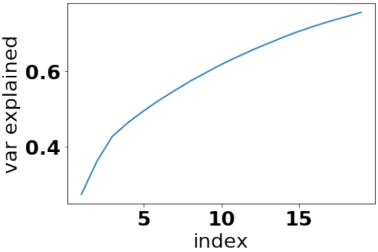
$$P_i = -\log L_i = \lambda_i - y_i \log \lambda_i + \log(y_i!),$$

$$P_i = D_i = y_i \log(y_i/\lambda_i) - (y_i - \lambda_i)$$

(negative log likelihood and deviance)

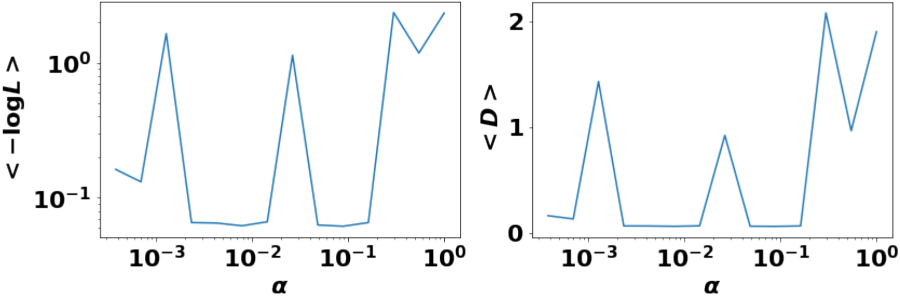
Geographically Weighted Poisson Regression

- Use one-hot encoding: categorical data become binary
- Use first 10 Principal Components to speed up computation ($\approx$ 52% of variance explained)



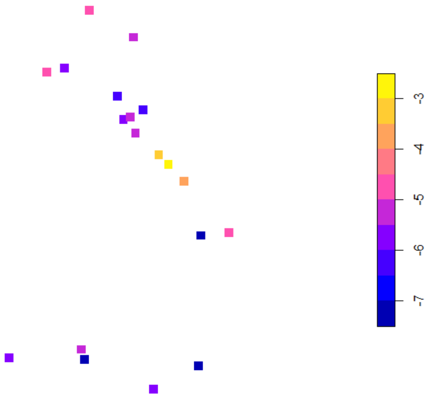
Geographically Weighted Poisson Regression

Negative log likelihood and deviance give similar results



Geographically Weighted Poisson Regression

Plot of $\log \lambda$ over space:



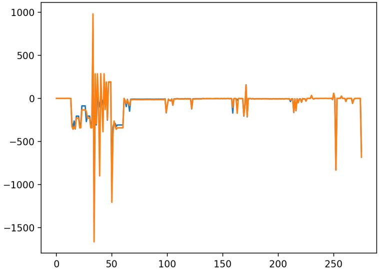
Geographically Weighted Poisson Regression

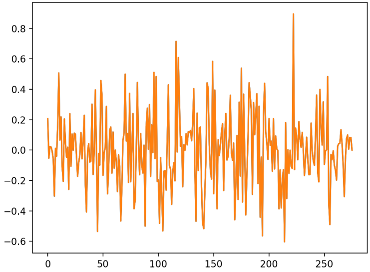
- Can consider model selection and possible further smoothing of regression parameters e.g.

$$\hat{\beta}(x) = \text{argmin}_{\beta} - \sum_{i \in \text{Training}} w(x, x_i) \log \Pr(Y_i = y_i | x_i, \beta(x)) + \xi ||\beta(x)||_2$$

Geographically Weighted Poisson Regression

- Compare non-regularized vs L^2-regularized $\hat{\beta}$'s:





Geographically Weighted Poisson Regression

Pros:

- Global: everything done within a single model
- Spatial smoothness
- Flexibility

Cons:

- Timing: to cross-validate over 20 α-values, with 10 PC, averaging 10 random batches of 0.1% of data takes 8600 s!
- Forces spatial dependence across all rating factors
- Flexibility

1. Introduction
2. Overview of Methods
   - Geographically Weighted Poisson Regression
   - Poisson Kriging
   - Fused Lasso Poisson Regression
3. Conclusion

Poisson Kriging

- Kriging is used to interpolate spatial attributes and predict the number of claims made per exposure unit at unsampled locations.
- Data aggregation based on location
- Assume given $S(\cdot), Y_i$, $i = 1, 2, \ldots n$,follows poisson distribution, where $S(\cdot) = \{S(x) : x \in D\}$ is a Gaussian random field

- Model via

$$\log \lambda(x) = S(x) \tag{1}$$

where

$S(x)$=$\mu(x) + \epsilon(x)$

$\mu(x)$=$\beta^T f(x)$ is a deterministic function of location x

$\epsilon(\cdot)$ is Gaussian random field with mean $-C_\epsilon(0)/2$ and covariance function $C_\epsilon(s-u)$=$\sigma^2 \exp(-\frac{||s-u||^2}{\phi})$

- Estimation

  β —- OLS by GLM

  σ^2, ϕ —- semivariogram model

- Semivariogram function

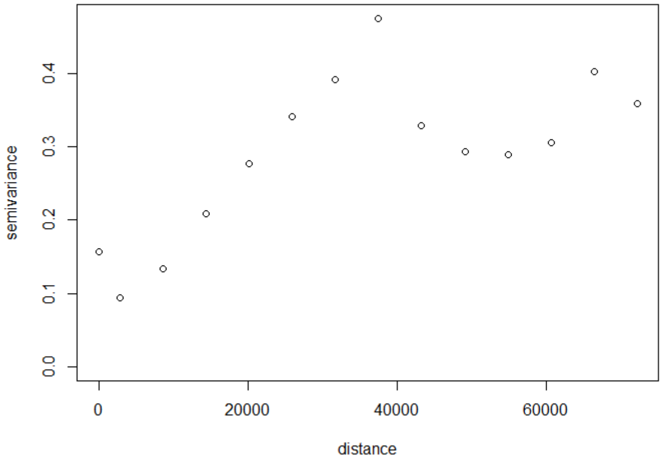
$$\gamma(s-u) = \frac{1}{2} var(Y(s) - Y(u)) \tag{2}$$

which describes the spatial covariance as function of distance. Observations that are geographically closer are more similar than observations that are further apart.

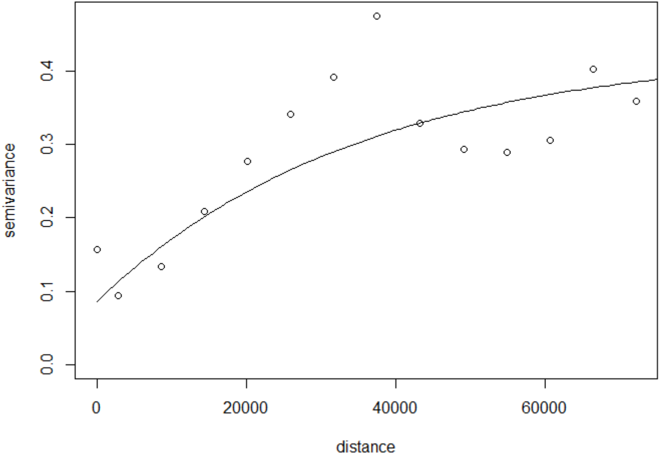
- Empirical semivariogram function

$$\hat{\gamma}(h) = \frac{1}{2N_h} \sum_{i=1}^{N_h} [Y(x_i + h) - Y(x_i)]^2 \tag{3}$$

N_h represents the number of pairs of data points that are h away.


semivariance
0.0
0.1
0.2
0.3
0.4
0
20000
40000
60000
distance

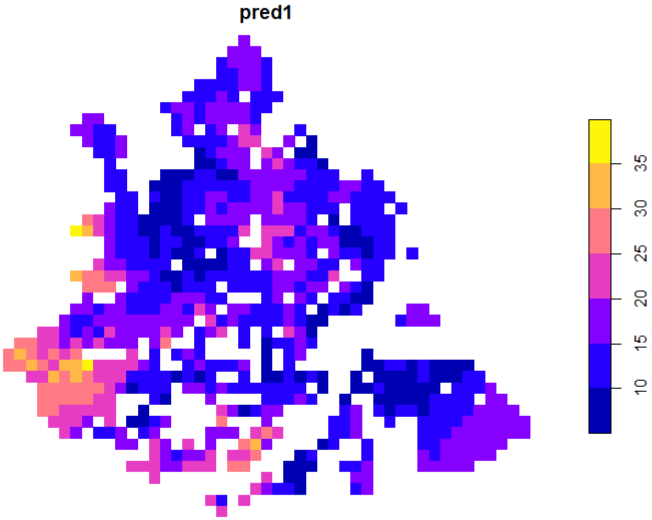
Next we use weighted least squares to fit through the variogram to get the spatial correlation for unsampled locations.


semivariance
0.0
0.1
0.2
0.3
0.4
0
20000
40000
60000
distance

- Prediction

$$Y(u_\alpha) = \sum_{i=1}^{k} \rho_i(u_\alpha) Y(u_i) \tag{4}$$

where $Y(u_i)$ is the count observed at location u_i and $\rho_i(u_\alpha)$ are the krige weights to be found. The krige weights ρ are found using the parameters obtained from the empirical variogram model.


pred1
35
30
25
20
15
10

Poisson Kriging

Pros:

- Simplification of the original method
- Parameter κ available for modifying the level of smoothing

Cons:

- Hard to apply to Poisson distribution, even harder for Gamma and Tweedie. Need large data for a good model.
- Very slow and RAM demanding, 2 hours for 5000 observations and 25gb of RAM. (Almost impossible higher than that)
- Hard to give user-specified weights to the observations.

Where did we Drew our Inspiration?

- Fused Lasso Linear Regression.
  - $y_1, \ldots, y_n \in \mathbb{R}$: data points from a dependent variable.
  - $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$: data points from covariates.
  - Objective function:

$$\min_{\boldsymbol{\beta}} \left\{ \frac{1}{n} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq t_1 \quad \text{and} \quad \sum_{j=2}^{p} |\beta_j - \beta_{j-1}| \leq t_2,$$

where $\boldsymbol{\beta} := (\beta_1, \ldots, \beta_p)$.

Two Considerations

1. Poisson regression instead:
   - replace $(y_i - \mathbf{x}_i^T\beta)^2$ by $-\log \mathbb{P}(Y_i = y_i)$,
   - where $Y_i \sim \mathcal{P}(\lambda_i)$ with $\lambda_i := \exp(\mathbf{x}_i^T\beta)$.
2. Intuitive when there is a natural order in the categorical location variable:

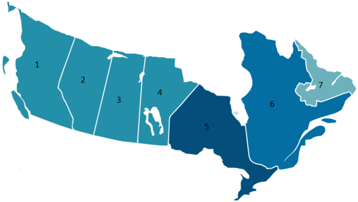


Figure: A modified version of Canada to illustrate the idea

Two Considerations (Cont'd)

2. For each location, we thus identify the "nearest" (with respect to the other geo-demographic covariates) neighbour:
   - replace $\sum_{j=2}^{p} |\beta_j - \beta_{j-1}| \leq t_2$ by $\sum_{j=1}^{p} |\beta_j - \beta_{j^*}| \leq t_2$.

Results

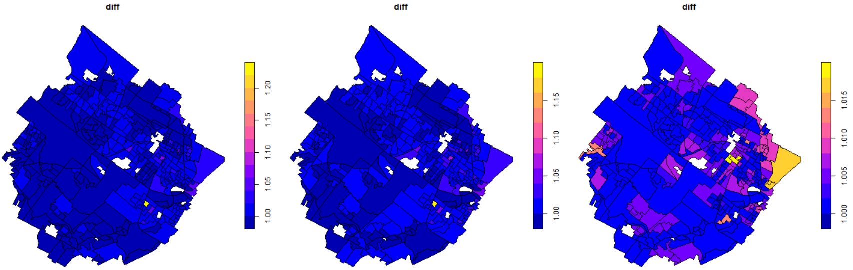


Figure: Maps with different values for t_1 and t_2

Fused Lasso Poisson Regression

Pros

- Fits the business needs.
- Intuitive.
- Simple and easy to interpret.
- Global: does everything at once.

Fused Lasso Poisson Regression

Cons and Room for Improvements

- Location variable with a lot of values (e.g. 200,000) implies that we need to create 200,000 dummy variables!
  - Solution: aggregate nearest neighbours and find new nearest neighbours (and repeat until the size is reasonable).
- Room for improvements:
  - Use Mahalanobis distance instead of euclidean distance to identify nearest neighbours (some correlations are strong).
  - No time to tune t_1 and t_2 (should be done using cross-validation).

Conclusion

- All methods provide interesting alternatives to the current methods
- All methods were difficult to scale to large data

The Team

Delphine Boursicot, Maxime Comeau, Philippe Gagnon, Charles Gauvin, Rachel Han, Bastien Ferland-Raymond, Michael Lindstrom, Nassim Razaaly, Juliana Schulz, Junwei Shen, Tony Wong, Ramin Eghbalzadeh

The Team

Thank you

Thank you to all the sponsors who helped organize this workshop!