EagleEye Analytics

Territorial Ratemaking

Eliade Micu, PhD, FCAS
emicu@eeanalytics.com

SCCAC Spring Meeting, June 2nd 2011

Description of the Problem

- ✓ Territorial ratemaking (and highly dimensional predictors in general) has been an area of active actuarial research lately

- ✓ Compare and contrast possible approaches:
  - GLM
  - GLM + spatial smoothing + clustering
  - Machine learning (rule induction)

- ✓ Newer approaches try to incorporate some domain knowledge in solving the problem, such as distance, spatial adjacency or other similarity measures

- ✓ Challenges:
  - Choice of building block (zip code, census tract)
  - Data credibility and volume
  - Ease of explanation

Evaluating Model Performance

- ✓ Fundamental predictive modeling questions:
  - How well would the model perform when applied to new risks (generalization power)?
  - How well does the model fit training data (goodness of fit)?
  - Selected model is always a “compromise” between these two criteria
- ✓ Analysis setup:
  - Split the data into training and validation datasets (60 – 40 split)
  - Derive new model using only the training data
  - Validate by applying the model to the validation data
- ✓ Model performance metrics:
  - *Correlation*: measure of predictive stability (generalization power), computed as the correlation coefficient of pure premium by territory between training and validation datasets
  - *Goodness-of-fit statistics* (deviances):
    - ➢ Derive relativities on training data, then apply them to validation data to compute new model fitted premiums
    - ➢ Compare new model fitted premiums to the observed incurred losses

EagleEye Analytics

Spatial Smoothing

- ✓ Compute better estimators for zip code loss propensity by incorporating the experience of other zips

- ✓ Requirements:
  - *Credibility*: zips with higher volume should receive less smoothing than zips with sparse experience
  - *Distance*: incorporate the experience of other zips based on some measure of "closeness" to a given zip
  - *Smoothing amount*: determined based on data, possibly adjusted due to pragmatic considerations

- ✓ Data needed:
  - "Zip code variables": demographic, crime, weather, etc
  - Location: latitude, longitude of zip centroid
  - List of neighbors for each zip

Spatial Smoothing – General Approach

- ✓ Fit GLM to multistate data:

  Observed Pure Premium ~ class plan variables + zip code variables

- ✓ Compute *Residual Pure Premium*:

  ResPP = Observed PP / GLM Fitted PP

- ✓ Adjust model weights:

  AdjEEXP = EEXP * GLM fitted PP

- ✓ Residual PP enters the smoothing algorithm, Adjusted EEXP are the model weights

- ✓ Choose:
  - distance measure between zips d_{ik}:
    - ➢ Distance between centroids
    - ➢ Adjacency distance: number of zips that need to be traversed to get from Zip_i to Zip_k
  - Neighborhood N_i

Inverse Distance Weighted Smoothing

- ✓ Aggregate AdjEEXP and ResPP at the zip code level
- ✓ Compute Smoothed Residual PP for each Zip$_i$:

$$SmResPP_i = Z_i \cdot ResPP_i + (1 - Z_i) \cdot \frac{\sum_{k \in N_i} AdjEEXP_k \cdot f(d_{ik}) \cdot ResPP_k}{\sum_{k \in N_i} AdjEEXP_k \cdot f(d_{ik})}$$

- ✓ Where:

$$Z_i = \frac{AdjEEXP_i}{AdkEEXP_i + K}$$

$$f(x) = \frac{1}{x^p}$$

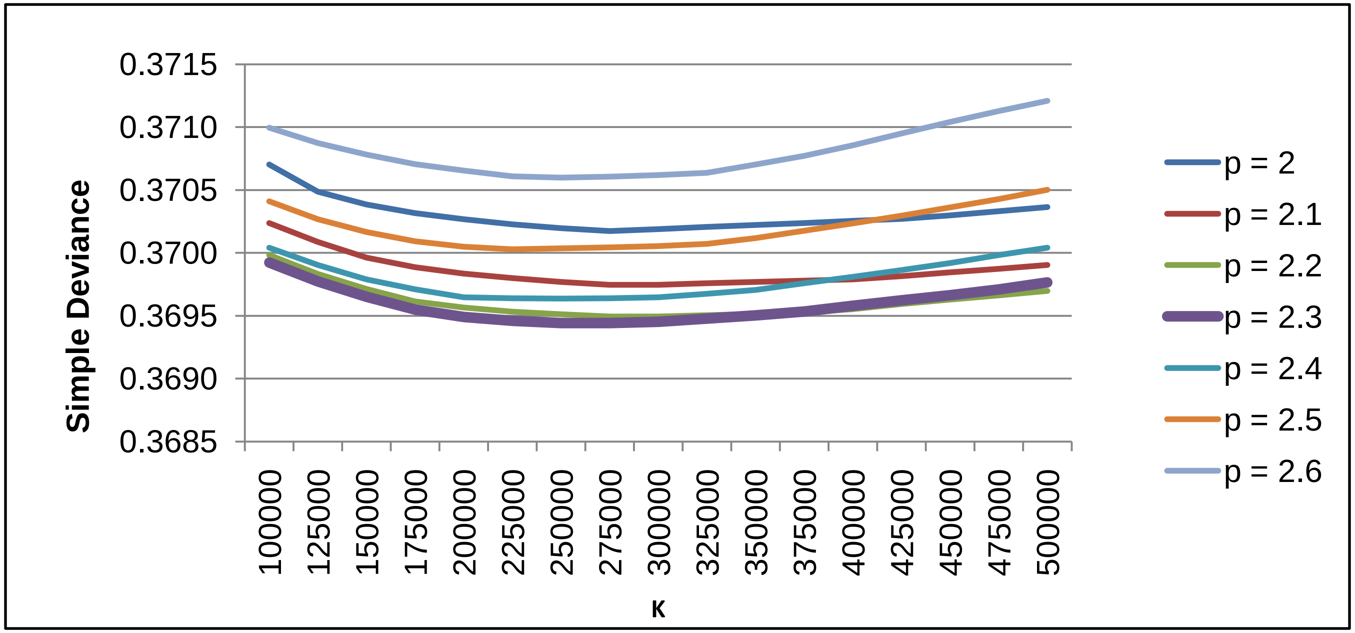
- ✓ Compute Fitted Geographical PP for each zip:

Fitted Geo PP$_i$ = SmResPP$_i$ · Zip Code Variables GLM relativities

EagleEye Analytics

Estimating K and p

- ✓ K and p need to be estimated from the *training* data by cross-validation
- ✓ Split the training data 70 – 30 at random
- ✓ Apply the smoothing algorithm on 70% of the data and compute Residual fitted pure premiums for each zip
- ✓ Compute a deviance measure on the remaining 30% and choose K and p that minimize deviance:



Clustering

- ✓ Type of *unsupervised* learning: no training examples
- ✓ Cluster: collection of objects similar to each other within cluster and dissimilar to objects in other clusters
- ✓ Form of data compression: all objects in a cluster are represented by the cluster (mean)
- ✓ Objects: individual zip codes, described by Fitted Geo PP_i
- ✓ Types of clustering algorithms:
  - *Hierarchical*: agglomerative or divisive - HCLUST
  - *Partitioning*: create an initial partition (possibly at random), then use iterative relocation to improve partitioning by switching objects between clusters – k-Means
  - *Density-based*: grow a cluster as long as the number of data points in the "neighborhood" exceeds some density threshold - DBSCAN
  - *Grid-based*: quantize space into a grid, then use some transform (FFT or similar) to identify structure - WaveCluster

How Many Clusters?

- ✓ Most algorithms have the number of desired clusters p as an input

- ✓ Between sum of squares (SS_b), within sum of squares(SS_w):
  - SS_b increases as the number of clusters increase, highest when each object is assigned to its own cluster, opposite for SS_w
  - Plot SS_b, SS_w vs. the number of clusters p and judgmentally select p such that the improvement appears "insignificant"

- ✓ Use F-test:
  - $F_w = SS_w(p) / SS_w(q)$ has a $F_{n-p,n-q}$ distribution
  - $F_b = SS_b(p) / SS_b(q)$ has a $F_{p-1,q-1}$ distribution
  - Select p based on a given significance level

- ✓ Clustering is unsupervised learning, so need better metrics to assess quality of results

Cluster Validity Index

- ✓ p clusters $C_1, \ldots, C_p$, with means $m_1, \ldots, m_p$
- ✓ Each object r described by a given metric x_r
- ✓ Define *Dunn Index*:

$$r(C_j) = \frac{1}{|C_j|} \sum_{r \in C_j} |x_r - m_j| \text{ (cluster radius)}$$

$$d(C_i, C_j) = \frac{1}{|C_i| \cdot |C_j|} \sum_{r \in C_i, s \in C_j} |x_r - x_s| \text{ (inter - cluster distance)}$$

$$D = \frac{\min\limits_{1 \le i < j \le p} d(C_i, C_j)}{\max\limits_{1 \le j \le p} r(C_j)} \text{ (Dunn Index)}$$

- ✓ Higher values for D indicate better clustering, so choose p that maximizes D
- ✓ Used k-Means with p=22 based on SS_b, SS_w and D

Alternative Approach

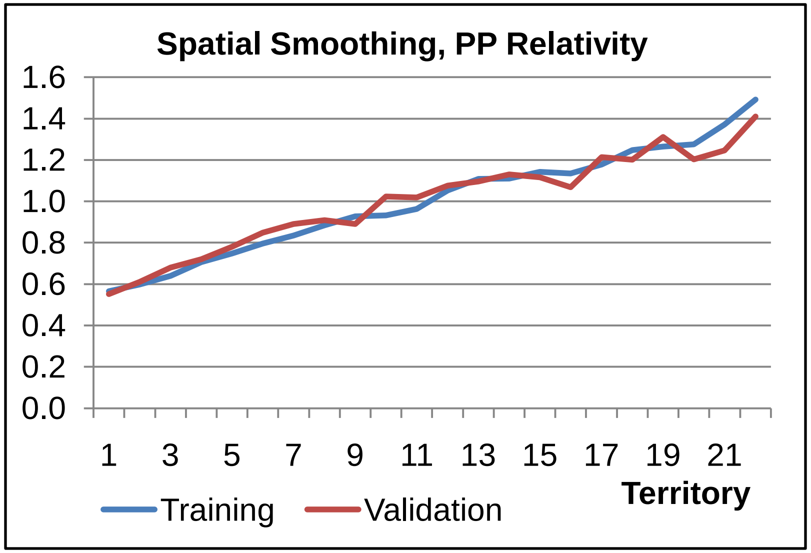
- ✓ *Machine Learning* methods:
  - Non-parametric: no explicit assumptions about the functional form of the distribution of the data
  - Computer does the "heavy lifting", no human intervention required in the search process
- ✓ *Rule Induction*:
  - Partitions the whole universe into "segments" described by combinations of significant attributes: *compound variables*
  - Risks in each segment are homogeneous with respect to chosen model response
  - Risks in different segments show a significant difference in expected value for the response
- ✓ The only predictors used are zip code variables, the segments will become the new territories
- ✓ Response: ResPP = Observed PP / Class Plan Variables GLM relativities
- ✓ Model weights: AdjEEXP = EEXP * Class Plan Variables GLM relativities

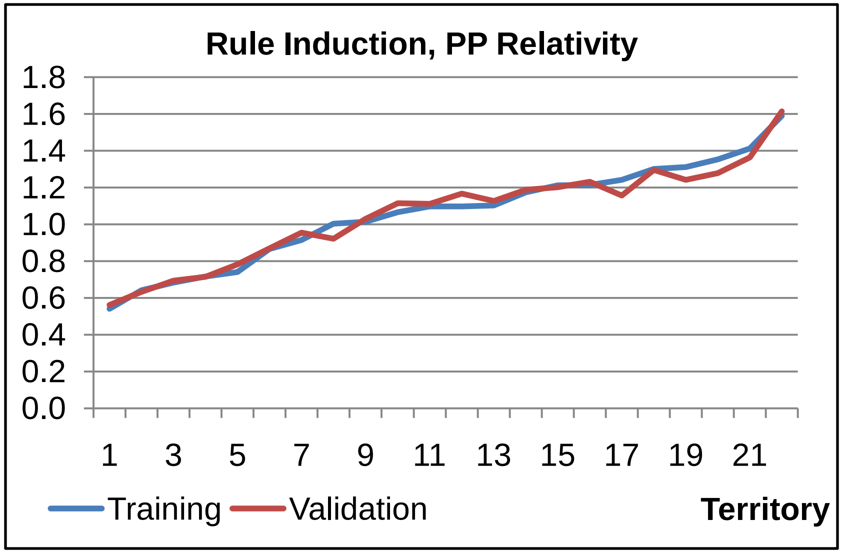
Segment Description – Illustrative Output

Segment	Description
1	Population=[-1 or 0 to 13119] TransportationCommuteToWorkGreaterThan60min=[-1 or 9 or more] CostofLivingFood=[95 to 122]
2	EconomyHouseholdIncome=[-1 or 53663 or more] TransportationCommuteToWorkGreaterThan60min=[-1 or 9 or more] PopulationByOccupationConstructionExtractionAndMaintenance=[-1 or 0 to 7] EducationStudentsPerCounselor=[27 to 535] HousingUnitsByYearStructureBuilt1999To2008=[-1 or 0 to 5]
…	...
20	TransportationCommuteToWorkGreaterThan60min=[-1 or 9 or more] Population=[-1 or 0 to 28784] HousingUnitsByYearStructureBuilt1990To1994=[0 to 2] CostofLivingFood=[-1 or 123 or more]
21	TransportationCommuteToWorkGreaterThan60min=[-1 or 9 or more] PopulationByOccupationSalesAndOffice=[0 to 28] EconomyHouseholdIncome=[-1 or 53663 or more] HousingUnitsByYearStructureBuilt1999To2008=[6 or more]
22	EconomyHouseholdIncome=[-1 or 53663 or more] TransportationCommuteToWorkGreaterThan60min=[-1 or 9 or more] PopulationByOccupationConstructionExtractionAndMaintenance=[8 or more] EducationStudentsPerCounselor=[27 to 535] HousingUnitsByYearStructureBuilt1999To2008=[-1 or 0 to 5]

Model Validation

- ✓ Each approach produced 22 territories using training data only
- ✓ Apply each set of territory definitions to the “unseen” validation data





Statistic	Spatial Smoothing	Rule Induction
Lift Training	2.64	2.95
Lift Validation	2.56	2.87
Correlation	98.09%	98.76%

EagleEye Analytics

Goodness of Fit Measures on Validation Data

$$\text{Simple Dev} = \sum_{i=1}^{n} \text{EEXP}_i \cdot \left| \text{Hist PP}_i - \text{Fitted PP}_i \right|$$

$$\text{Sum of Squares Dev} = \sum_{i=1}^{n} \text{EEXP}_i \cdot \left(\text{Hist PP}_i - \text{Fitted PP}_i \right)^2$$

$$\text{Chi Sq Dev} = \sum_{i=1}^{n} \text{EEXP}_i \frac{\left(\text{Hist PP}_i - \text{Fitted PP}_i \right)^2}{\text{Fitted PP}_i}$$

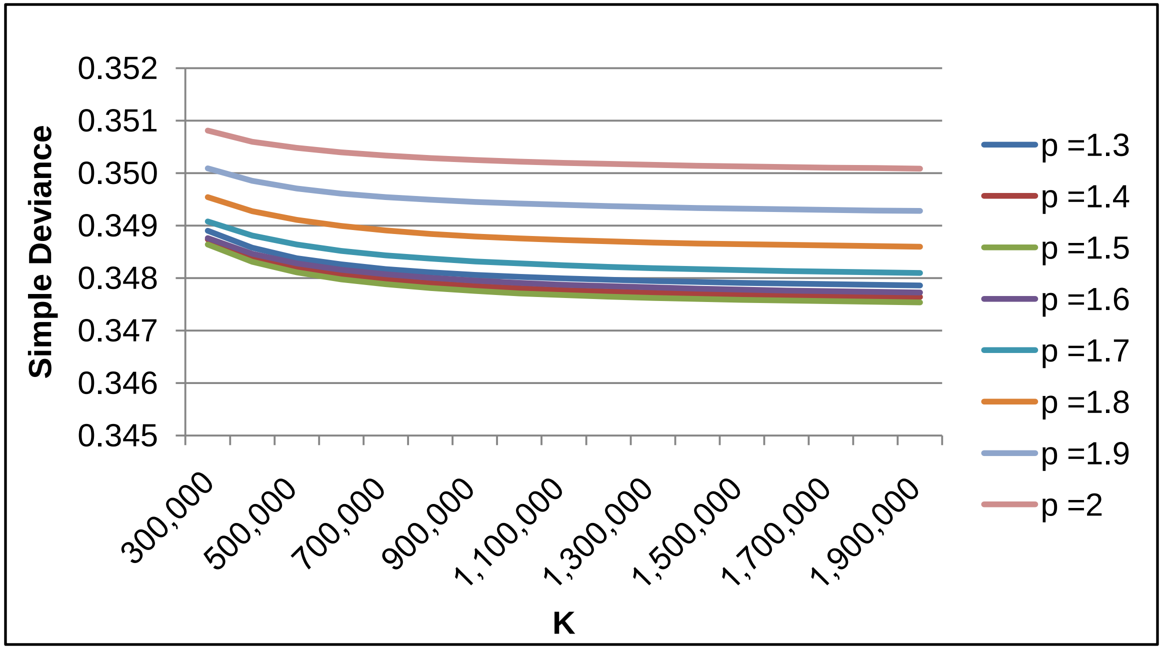
	Simple Dev	SS Dev	Chi Sq Dev
Spatial Smoothing	0.3084	0.2235	0.3201
Rule Induction	0.2984	0.2199	0.3155
Improvement	3.26%	1.63%	1.43%

Agreement on Predicted Values

Spatial Smoothing Territory	Rule Induction Territory																					
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	4.3%	0.1%	0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
2	1.4%	2.4%	0.3%	0.2%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
3	0.3%	1.6%	1.3%	0.6%	0.7%	0.0%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
4	0.0%	0.2%	1.2%	1.2%	1.7%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
5	0.0%	0.7%	1.3%	1.0%	1.4%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
6	0.0%	0.1%	0.5%	1.3%	1.2%	1.0%	0.4%	0.0%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
7	0.0%	0.0%	0.1%	0.3%	0.3%	2.0%	1.6%	0.0%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
8	0.0%	0.0%	0.0%	0.0%	0.2%	1.6%	1.9%	0.4%	0.4%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
9	0.0%	0.0%	0.0%	0.0%	0.3%	0.3%	0.2%	2.1%	1.4%	0.1%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
10	0.0%	0.0%	0.0%	0.0%	0.1%	0.0%	0.1%	1.6%	1.2%	0.8%	0.4%	0.0%	0.0%	0.1%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
11	0.0%	0.0%	0.0%	0.0%	0.1%	0.0%	0.0%	0.7%	0.5%	0.8%	1.9%	0.2%	0.0%	0.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
12	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	1.9%	1.7%	0.3%	0.1%	0.2%	0.2%	0.0%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%
13	0.0%	0.0%	0.0%	0.0%	0.4%	0.0%	0.0%	0.1%	0.6%	0.6%	0.7%	1.5%	0.2%	0.0%	0.4%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
14	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.1%	0.5%	0.5%	0.6%	0.9%	1.1%	0.5%	0.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
15	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.3%	0.5%	1.2%	0.7%	0.5%	0.2%	0.5%	0.3%	0.0%	0.0%	0.0%	0.0%
16	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.3%	0.0%	0.0%	0.1%	0.4%	0.6%	0.5%	0.9%	0.0%	0.9%	0.9%	0.0%	0.0%	0.1%	0.0%
17	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	1.0%	1.4%	0.4%	0.6%	0.8%	0.0%	0.1%	0.3%	0.0%
18	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.1%	0.8%	1.7%	0.1%	0.7%	0.0%	0.3%	0.8%	0.0%
19	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.1%	0.0%	0.0%	0.4%	0.9%	0.5%	1.7%	0.3%	0.3%	0.0%
20	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.1%	0.1%	0.0%	0.0%	0.3%	1.8%	0.6%	1.9%	0.0%
21	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.6%	2.8%	1.0%	0.0%
22	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	1.1%	1.0%	2.6%

Spatial Smoothing + Rule Induction

- ✓ Try to combine both methods, any potential gain?
- ✓ Remove the signal accounted for by rule induction, apply spatial smoothing on the residuals
- ✓ Determine K and p using the same approach: the implied value for K is very large, which suggest that there is no signal left in the residuals



Conclusions

- ✓ Both models validated well when applied to unseen data

- ✓ Rule Induction:
  - Provides more lift and better fit
  - Plain English description for the territories
  - Less information required
  - May be applied to other states with sparser data
  - Easy to extend to other highly dimensional problems (symbols)

- ✓ Spatial Smoothing:
  - Makes intuitive sense for PPA (driving patterns)
  - Requires user selection for distance measure, neighborhood, clustering algorithm and number of clusters
  - Less transparent, harder to explain
  - Challenging to extend to other problems: distance, neighborhood

EagleEye Analytics