Outline of Ice Project to Help with Write-Up

Alison Kleffner

Last Updated: 10/10/2021

Ice Features

Introduction to the Problem

- Why We Care
  - Ice is a barrier between the warm air of the ocean and the atmosphere (ask for paper on this?)
  - Transportation Routes
- Introduction to the Data
  - Number of Days of Data: 22 Days
  - From Satellite that passed overhead
    - Explain issues with this method of data collection
  - gpid - identity (how many trying to track)
  - xmap - Location that is region-bounded - latitude
  - ymap - location that is region-bounded - longitude
  - observation time (and t) - One is rounded the other is not. differ by 1 integer - 1 day apart
  - k - image index (if more than one image at same t)
  - Some duplicate data and some missing data
- What we are hoping to do with the data
  - Figure out where Ice Cracks would form using clustering methods and the trajectories of the gpids, with the idea that the boundary between two clusters would be where ice cracks would form (different trajectories)
    - Moving away from each other - pulling ice apart
  - Do this with all of the weeks information used in the clustering, and also by breaking it down by week (subtrajectories)
- Put Plot of trajectories
  - Plot vector path of each gpid excluding the missing data. Has arrow to show direction that is is moving.
  - Explain the plot and what it is showing us
  - What to do with animations in written report?
  - Motivation for our bounding box idea

Literature Review (Beginnings - Needs a lot more work)

- Overview of spatial-temporal clustering methods:
  - Overview Papers
    - A Review of Moving Object Trajectory Clustering
      - Each point in a trajectory represents a position in space in a certain time.
      - Trajectory clustering aims at finding out trajectories that are of the same or similar patterns
      - 5 Categories (Han et. al 2011)
      - 1. Partition Based Methods - kmeans and kmedoids
      - 2. Hierarchy Based Methods - Agglomerative and divisive
      - 3. Density Based - The idea is adding the area to the cluster which is closer to it, while

the density of points in the area is greater than the threshold.
            - 4. Grid-Based Clustering - adobt a multi-resolutoin grid structure where the data space is quantized to a limited number of units which form a grid structure. All clustering operations are carried out on the grid
            - 5. Model-Based Clustering
            - Most similarity measures must have trajectories of the same length (not our case)
    - Density-Based Methods
        * ST-DBSCAN: An Algorithm for Clustering Spatial-Temporal Data
            - Looks to be used in traffic analysis
            - Cluster based on a density threshold
    - Model-Based
        * Discovering Clusters in Motion Time-Series
            - Independence Issue
    - Partition and Group Based Clustering - since we are also looking at sub-trajectories -Lee et al. (2007) provide a partition-and-group framework for trajectory clustering. Cluster based on these trajectories.Works well when lengths of trajectories are quite different, but easily affected by the partition criterion of the trajectory
- Give papers on examples of spatial-temporal clusterings?
- Literature on determining number of clusters (to be upfront with how we determined ours)
- Paper: Analyzing Nonstationary Spatial Data Using Piecewise Gaussian Processes (Interest in trying this method on our data)
    - Useful if have sharp changes in correlation structure at boundaries
        * What we would imagine we would have with different movement in the ice
    - Use of separate Gaussian model for each of the regions which are created using a Voronoi Tesselation
    - Use of Bayesian models (also way to figure out how many regions are necessary, instead of just using gap statistic, etc)
    - Doesn't have a temporal component.
- Need a section on Interpolation Methods
    - Introduce the standard methods
    - Talk about difficulty since want locations, not a y value.
    - Interpolation of spatial neighbors and trajectory interpolations
- Why won't these work for us?
    - How are data is (no true response variable)
    - Observations not independent through time
    - Missing Data/Unequal lengths of trajectories

Challenges

- How gpids are laid out (can't use density-based clustering)
- missing chunks of data
- only motion data is observed
- Typical interpolation methods aren't suitable
    - Non-smooth spatial process
    - nonstationarity due to ice moving as patches

Our Method

- Bounding Box Idea: Put a bounding box around the trajectories from before, with the idea that can find regions that shift against each other.
    - length of x traveled
    - length of y traveled
    - angle of movement
    - when done by week, also used prior weeks bounding box information
    - average x location (try and get contiguous clusters)
    - average y location (try and get contiguous clusters)

- Standardized Data
  - All on different scales, so that way average x and y didn't get more weight than others just because it was bigger
- Determining number of Clusters
  - With Kmeans has to be determined a priori, and don't really have any information beforehand to help determine this
  - Used Silhouette Statistic (also should try gap statistic and elbow method)
  - Dynamic process, so would expect when clustering by weeks to have different number of clusters for each week.
- Looking at total and broken out for the three weeks
- Tried Different Combinations of Bounding Box Features for the Clustering algorithm to visually determine which one is best.
- How Dealing With missing Data for now
  - down/up imputation (for plotting) - Use again some later
  - Explain why the bounding box might be helpful when dealing with this missing data.

Results

- Maps showing clusters
- What we see in these clusterings
- Explain the different pieces of it -> the different clusterings, imputed information
- Why won't go smaller than a week
  - Too much change (not stable)
- Maybe talk about other things that we have tried
  - Used both kmeans and agglomerative hierarchical clustering

Interpolation Method

- Want to interpolate the missing x/y gpid information
- Prior Methods:
  - IDW - Weighted Average of the data points, giving the closest locations more weights
  - Regression (Trend-Surface) Estimation (Things Attempted)
    - SpDynLM - univariate spatio-temporal
    - SpMvLM - multivariate interpolation (spatial interpolation at one time point)
- Issue with prior methods
  - We don't have covariates (so can't use the regresssion methods)
    - For spatial - temporal interpolation, in order to calculate the distance matrix, need latitude and longitude
  - Missing chunks of data at each time point due to where satellite passes, which makes IDW difficult because don't have nearest neighbors
- Our method - Polygon intersection method
  - Trying to find spatial and temporal neighbors and use these to interpolate onto a grid
  - Create Polygons for each of the weeks, where a polygon is created for each cluster
    - Process: Create polygons one by one. Start with polygon, then remove all of the gpids found within the boundaries of this polygon from the data set. Then using this reduced data set, create the next cluster polygon. Repeat process until have a polygon for each of the gpids. Have some issues with overlapping but this is due to the polygons being convex, so not super much we can do about that at the moment.
    - Need to talk about order of creating the polygons and how tried to smooth out the clusters so more continguous (Having a hard time explaining my order process - would pictures help?).
  - Find the intersections of each of these polygons with polygons from a different time point. All of the gpids in this area become it's spatial/temporal neighbors.
- Create a grid to estimate and using centroid of grid cell to give an estimate of the missing gpid location in order to use a model for interpolation

- Using fit_model in gpgp package with a exponential_spacetime covariance function to create our model for interpolation.
  - GpGp package does fast gaussian process computation using Vecchia's Approximation (How much detail do we need?)
    - Vecchia's Approximation is a technique of approximating the joint likelihood function of several variables
      - How into detail do we need to do with this?
      - Leads to computation savings
      - Has emerged as a leader in Gaussian process approximations
    - fit_model returns the maximum Vecchia likelihood estimates, obtained with a Fisher's Scoring Algorithm -Choices made in function to potentially talk about?
      - reorder = True
      - group = True
      - m_seq (sequence of values for number of neighbors) - computational savings when this is small - part of more recent generalization of Vecchia's approximation
- To test to see how well this is working, cross-validation on gpids where we have data for the location at that time.
- Once we have this, can apply numerical algorithm to derive ice cracks from this idea in comparison to just using linear interpolation.

Where we are going from here

- Finish Interpolation
- Comparison of Methods
- Create a pipeline to make this process a little more automated.
- Future Work -> Something to make clusters continguous? (Paper)

Citations (of what have in lit review so far)

- J. Alon, S. Sclaroff, G. Kollios and V. Pavlovic, "Discovering clusters in motion time-series data," 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings., 2003, pp. I-I, doi: 10.1109/CVPR.2003.1211378
- Kim, H.-M., Mallick, B. K., & Holmes, C. C. (2005). Analyzing Nonstationary Spatial Data Using Piecewise Gaussian Processes. Journal of the American Statistical Association, 100(470), 653–668. http://www.jstor.org/stable/27590585
- Lee, J. G., Han, J., & Whang, K. Y. (2007). Trajectory clustering: A partition-and-group framework. In SIGMOD 2007: Proceedings of the ACM SIGMOD International Conference on Management of Data (pp. 593-604). (Proceedings of the ACM SIGMOD International Conference on Management of Data). https://doi.org/10.1145/1247480.1247546
- Yuan, Guan & Sun, Penghui & Zhao, Jie & Li, Daxing & Wang, Canwei. (2017). A review of moving object trajectory clustering algorithms. Artificial Intelligence Review. 47. 10.1007/s10462-016-9477

Interpolation Method Description (some really rough writing - needs lots of work)

Motivation for wanting to Interpolate the Missing Data

Our proposed method for interpolating missing locations for gpids at certain times involves finding and using the spatial and temporal neighbors of the missing gpid. As a new bounding box is created for each each week, the clusterings for each of these weeks are used to find these spatial-temporal neighbors, with the intersection of each of the weekly clusterings being a group of spatial and temporal neighbors for that week. For the first week, the cluster intersections of Week 1 and Week 2 formed the different spatial-temporal neighbor groups. The spatial-temporal neighbor groups for Week 2 and Week 3 were created in a likewise fashion with Week 2's being created from the intersection of Week 1, Week 2, and Week 3's clusters, and Week 3's being created from the intersection of Week 2 and Week 3's clusters.

In order to find these intersections, for each week polygons needed to be created for each of the clusters.

A polygon is created by finding the coordinates of each of the clusters. In order to reduce the amount of overlapping between the cluster polygons, they were created in a strategic order. This was done by first creating a polygon from a cluster on the top edge of the ice chunk. After this polygon is created, then all of the gpids that are located within this polygon is then removed from the data set, even if the gpid belonged to a different cluster. This was done to reduce overlapping. This process is then repeated with the next cluster and continues until all of the clusters have then been put into a polygon. For Week 1 and Week 2, one of the clusters is very distinctly split into two locations. For this particular cluster, a different polygon was created for each of the two different locations.

How get intersections and what these intersections mean - picture!

Grid Creation: In order to use our intersection method, a spatial grid encompassing the gpids was created for each time to give an initial starting point for the estimation of the missing gpids. These would then be used in our model used in our created model for the estimation of the locations of the missing gpids. The size of our grid cells used was 10 km by 10 km, in order for a maximum of four gpids to be located in the cell. Then as the estimate for the gpids, the centroid of the gpids was used, so each of the gpids located in that cell would have the same initial estimate.

Model (GpGp package - learn more about this!)

Cross - Validation Results

Comparison to linear interpolation??

Description of Trajectory Plot

Description of the Data used