DATA STORIES
Data Science Lab. @ University of Piraeus

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
UNIVERSITY OF PIRAEUS

Mobility Data Analytics

Yannis Theodoridis
Data Science Lab.*, Univ. Piraeus

*** Credits**: Eva Chondrodima, Christos Doulkeridis, Harris Georgiou, Yannis Kontoulis, Nikos Pelekis, Panagiotis Tampakis, George S. Theodoropoulos, Andreas Tritsarolis

v.2022.06

The Data Science Lab @ UniPi.GR

Our research agenda:

- Extreme-scale data management
- Mobility data analytics at the computing continuum (edge / fog / cloud)
- Time series analytics & forecasting
- Semantic integration
- etc.

https://www.datastories.org

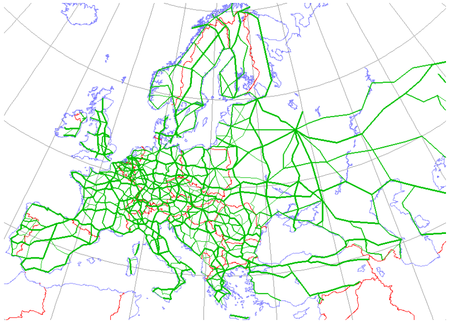
Outline

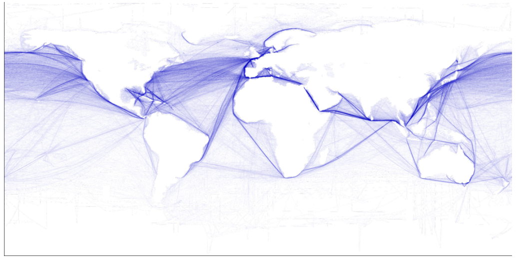
1. **Introduction - Getting to know mobility data**
2. **Pre-processing mobility data**
   - Cleansing, Simplification, Enrichment, Sampling, etc.
3. **Analyzing mobility data**
   - Point / trajectory clustering
   - Group behavior discovery
   - Future location prediction
4. **Real-world use case**
5. **Summary**

1.
Introduction –
Getting to know mobility data

Application domains

- **Land movement**: Find shortest path from location A to location B; Which points of interest (POIs) are found in a range of 5 km from A? etc.
- **Sea / Air movement**: Find the routes from (sea/air) port A to port B with direct connection (or at most 1 intermediate stop)? Which is the anticipated movement of vessel / aircraft X during the next Δt? etc.





All images source: Wikipedia.org

Examples of datasets @ land

- **GeoLife** (source: Microsoft Research Asia)
  - 182 user movements (under various transportation means) organized in 17,621 trajectories;
  - 68 Km in 2,7 hrs. per trajectory, avg.;
  - dense sampling (1 sample every ~5 sec)
- **T-Drive** (source: Microsoft Research Asia):
  - 2,357 taxis in Beijing for 1 week (15 million points, in total);
  - 869 Km per taxi, avg.;
  - sparse sampling (1 sample every ~3 min)

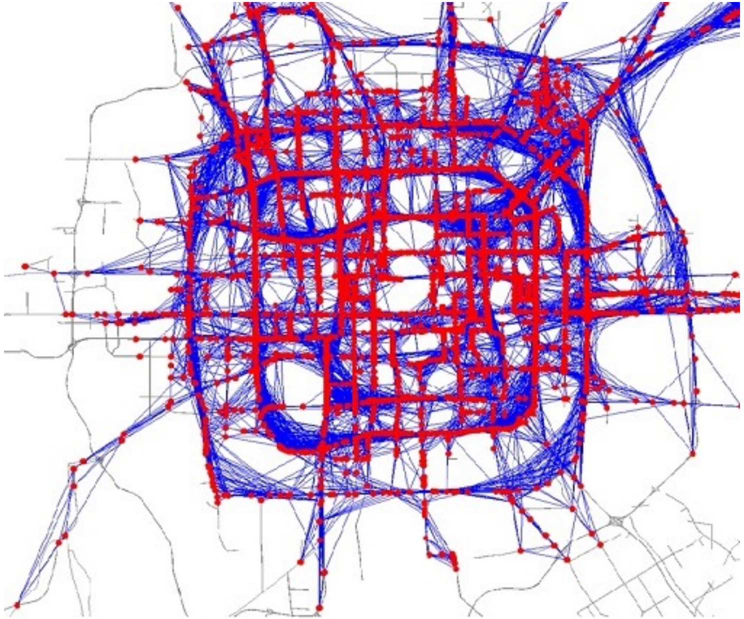
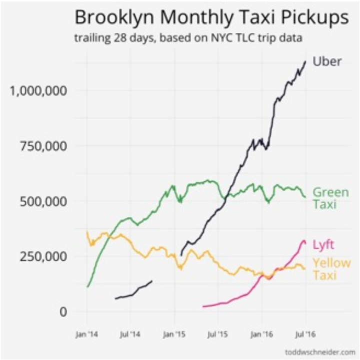


image source: research.microsoft.com

Examples of datasets @ land (cont.)

- **NYC taxis** (source: NYC Taxi & Limousine Commission): 1.4 billion trips, Jan. 09 – Dec.17.
  - **Ride-hailing apps** data are also provided
  - Attention: pickup – drop-off locations are only available



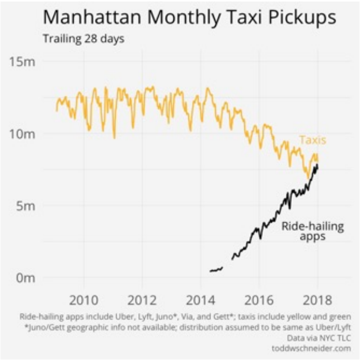
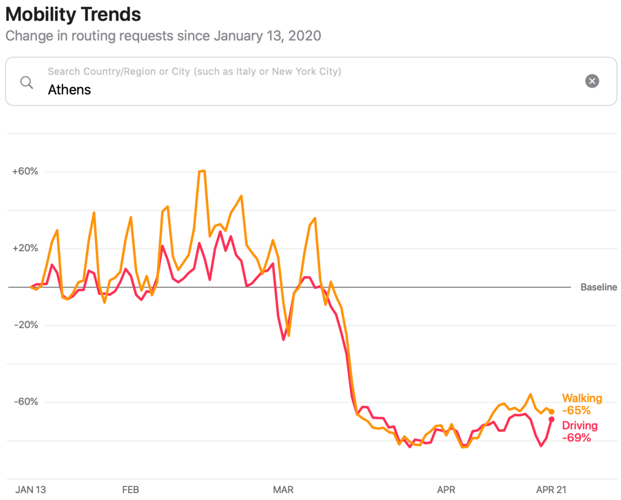
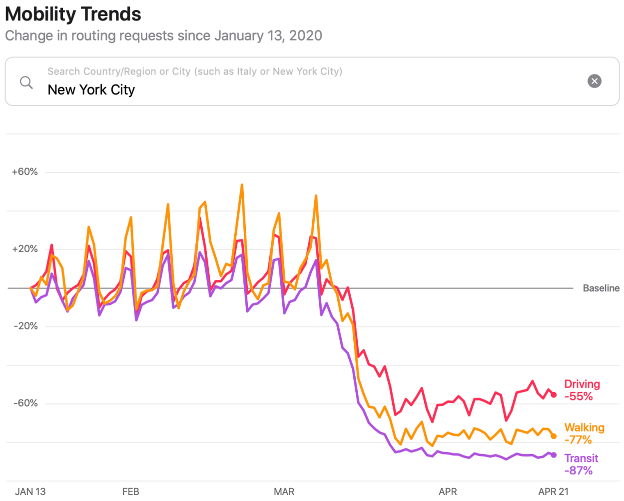


image source: toddwschneider.com

Examples of datasets @ land (cont.)

- Mobility trends during COVID-19 pandemic
  - e.g., search for correlations (Theodoridis & Theodoridis, 2021; Georgiou et al. 2022)

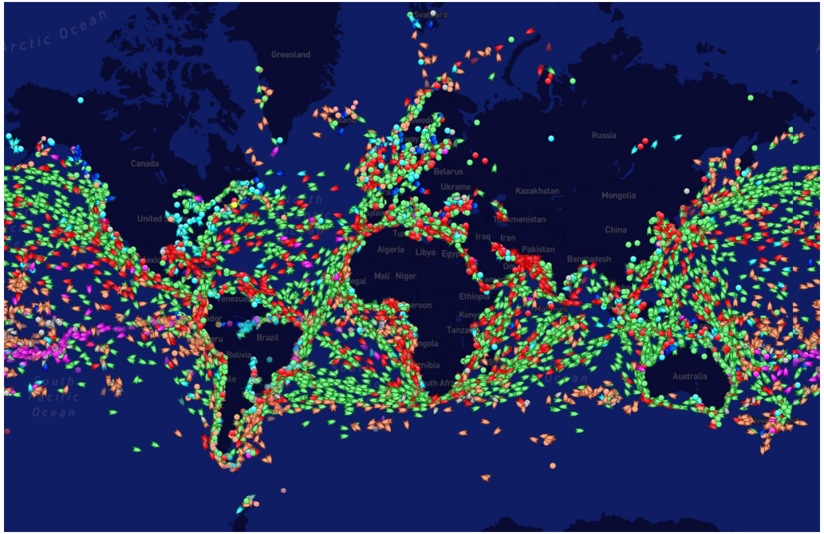




Data source: www.apple.com/covid19/mobility

Examples of datasets @ sea

- **AIS** (Automatic Identification System)
  - >250,000 vessels tracked daily (source: marinetraffic.com)
  - AIS signal transmitted: every 2 to 10 sec depending on speed while underway; every 3 min while at anchor



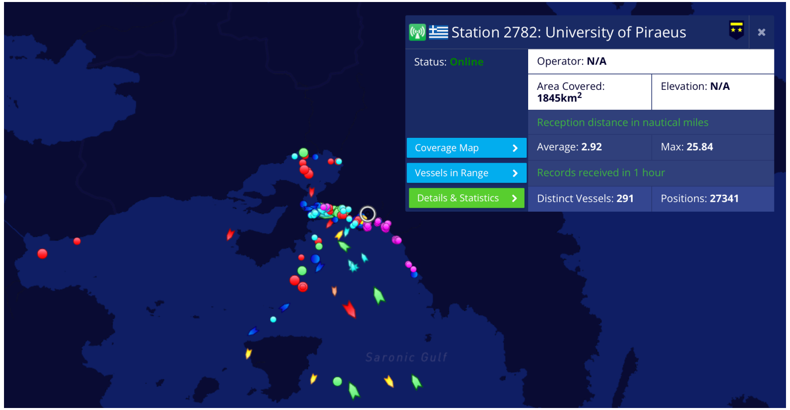


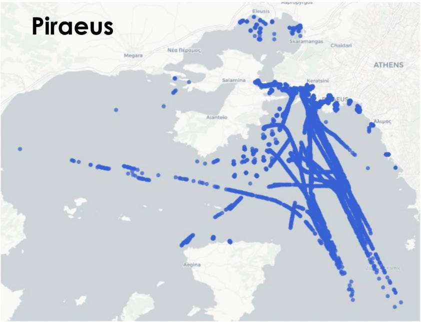
image source: marinetraffic.com

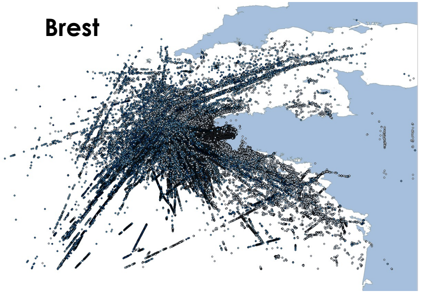
- top: global snapshot on May 26th, 2022; vessel colors correspond to different vessel types (e.g., cargo is green, tanker is red)
- left: vessels tracked by the Univ. Piraeus' AIS station

Examples of datasets @ sea (cont.)

- **Piraeus (GR)** provided by Univ. Piraeus *
- **Brest (FR)** provided by French Naval Academy **

Dataset	**Piraeus**	**Brest**
time frame	~32 months (9/5/2017-26/12/2019)	6 months (01/10/2015–31/03/2016)
# of records	~244M	~16M
# of distinct vessels	~6K (anonymized)	~5K
sampling rate (avg.)	~5 min	< 1 min
complementary data	ports, coastline, weather, areas of interest, etc.	ports, coastline, weather, trajectory synopses, etc.
Zenodo downloads	~1K (since 2021)	~14K (since 2018)





* https://doi.org/10.5281/zenodo.5562629
** https://doi.org/10.5281/zenodo.1167594

Examples of datasets @ air

- **ADS-B** (Automatic Detection System - Broadcast)
  - >15,000 aircrafts flying at the same time worldwide (source: flightradar24.com)
  - ADS-B signal transmitted: every 1 sec while on air; not transmitted while on the ground

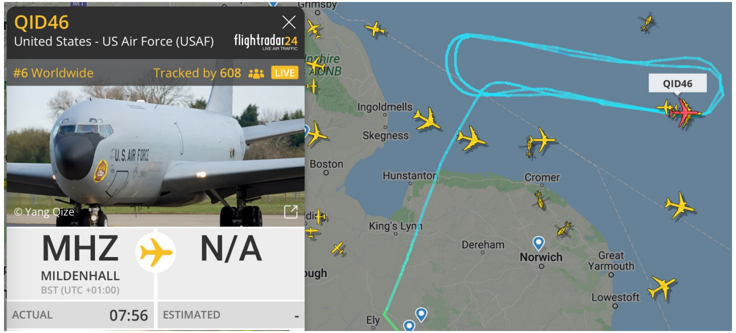
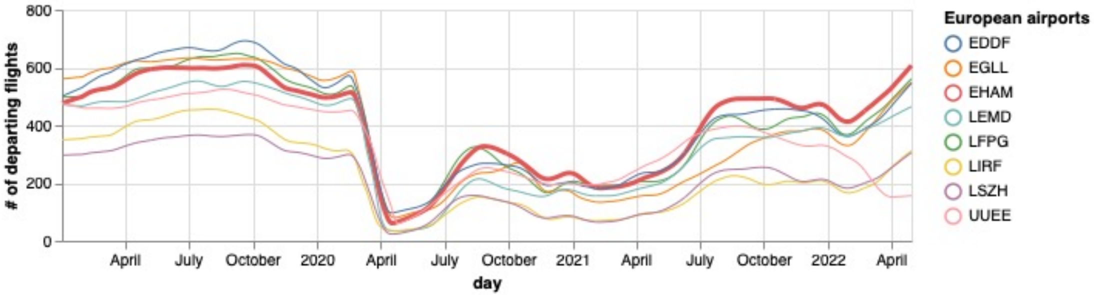


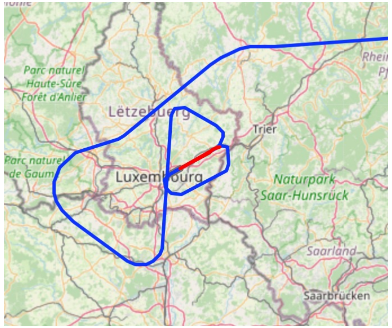
image source: flightradar24.com

- top: global snapshot on May 25th, 2022; yellow vs. blue planes if located by terrestrial vs. satellite stations
- left: the route of a military aircraft

Examples of datasets @ air (cont.)

- **Air traffic** provided by OpenSky Network*
  - For each flight, origin-destination airports and respective timestamps
  - Timeframe: Jan 1st, 2019 – Jan. 31st, 2022 (ongoing)
    - high vs. low peak: Aug. 2019 (2.3M records) vs. Apr. 2020 (843K records)
  - Related dataset: in-flight emergency situations **
  - More analytics examples at https://traffic-viz.github.io





Flight A319 near Luxembourg on Aug 20th 2019, "hot brakes" alarm

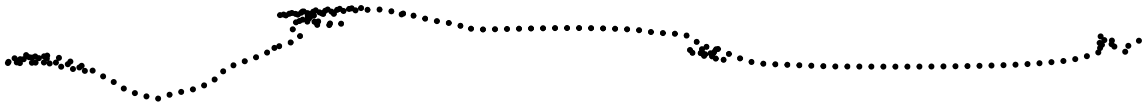
* https://doi.org/10.5281/zenodo.3737101
** https://doi.org/10.5281/zenodo.3937482

2.
Pre-processing mobility data

Data pre-processing

- Definition: **preparing data for analytics purposes**

$$T = \{ <p_1, t_1>, <p_2, t_2>, \ldots, <p_n, t_n> \}$$

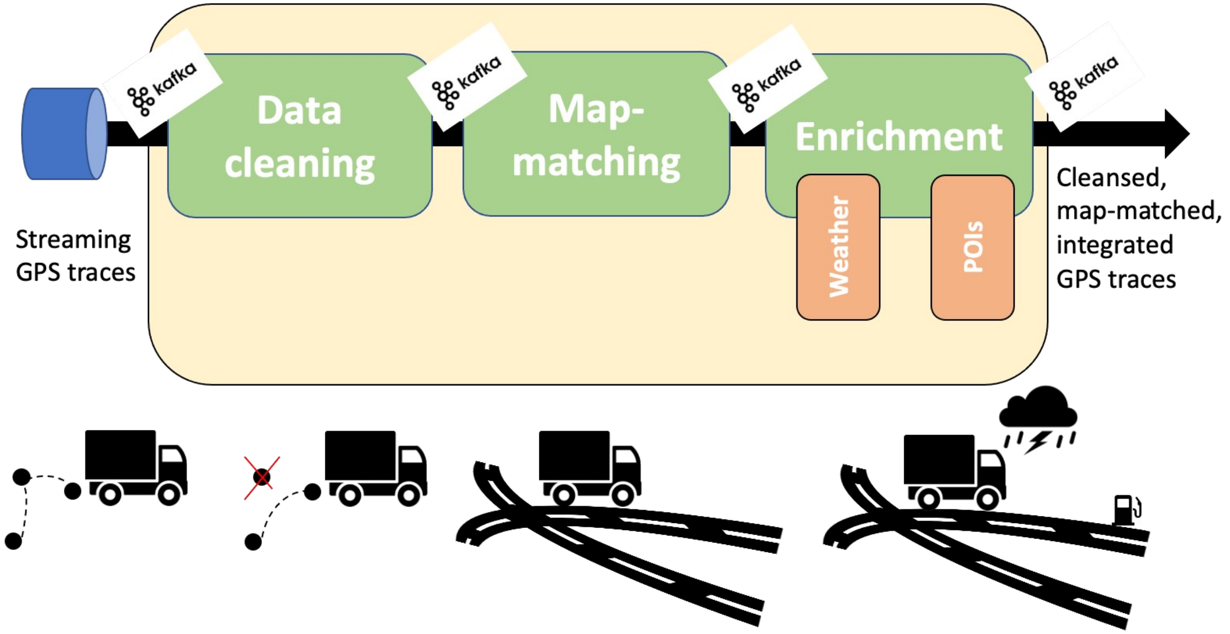


- Data pre-processing includes:
  - **Cleansing** (noise removal, smoothing, map matching, etc.)
  - **Transformation** (trajectory segmentation, simplification, etc.)
  - **Enrichment** (semantic annotation, data fusion, etc.)
  - **Sampling** (over the entire dataset)

  etc.

Data pre-processing (cont.)

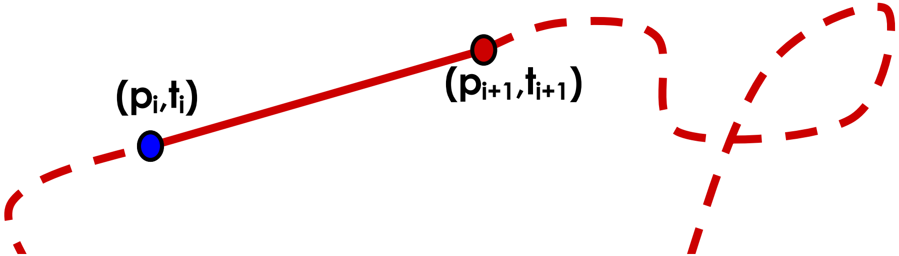
- An example: **data pre-processing pipeline (urban traffic)**



Source: Track & Know project

From GPS locations to trajectories

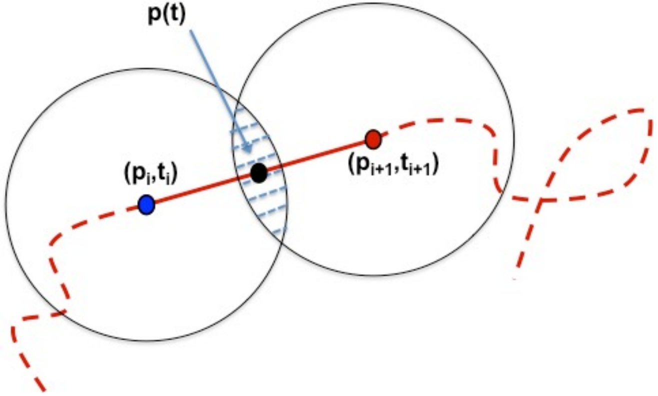
- GPS records correspond to **samples** (p_i, t_i) of our movement – inferring 'continuous' movement is not trivial.
- A trajectory is represented by a **3D[4D] polyline** (x-, y-, [z-,] t-); vertices correspond to (p_i, t_i)
  - alternative: a 2D[3D] polyline consisting of p_i's along with an array of t_i's
- Typically, **linear interpolation** is assumed between (p_i, t_i) and (p_{i+1}, t_{i+1})

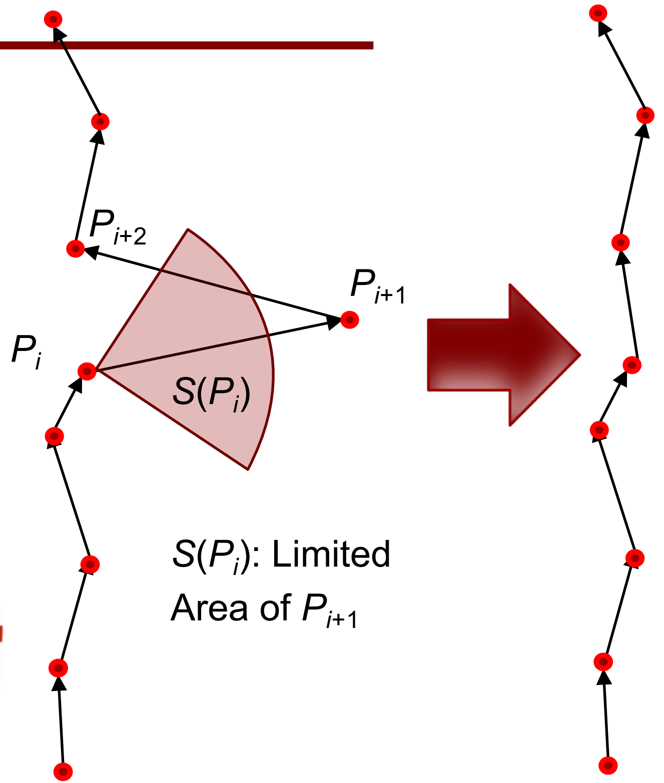


$$p(t) = \left(x_i + \frac{t - t_i}{t_{i+1} - t_i}(x_{i+1} - x_i), y_i + \frac{t - t_i}{t_{i+1} - t_i}(y_{i+1} - y_i) \right)$$

GPS Data Cleansing

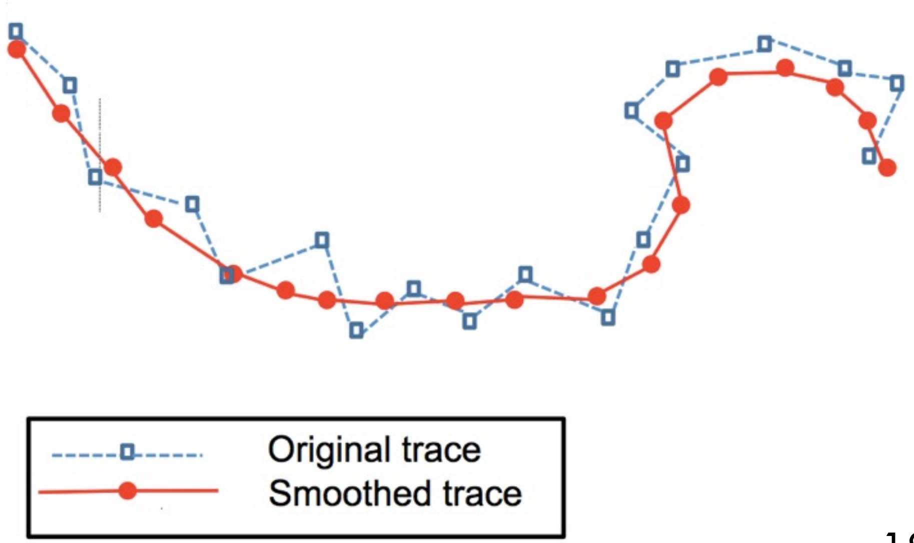
- Erroneous recordings: noise vs. random errors
- **Noise** corresponds to values that are 'impossible' to appear
- Can be detected and removed using appropriate filters
  - e.g., maximum speed
- **Potential Area of Activity** (PAA)





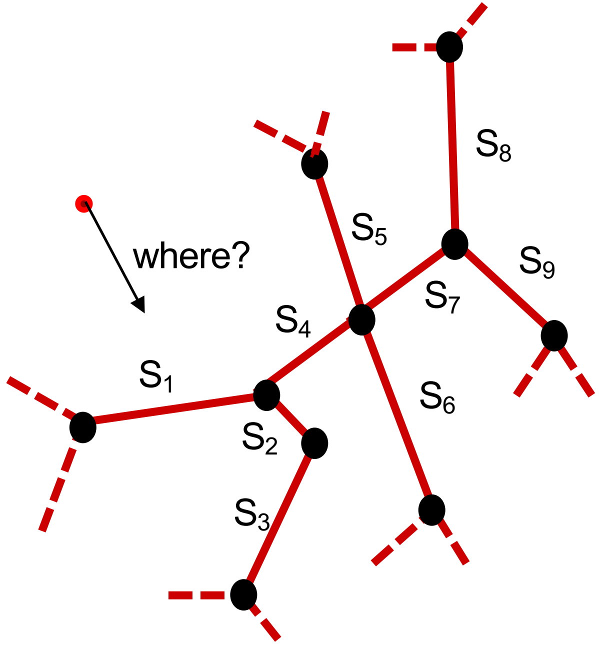
GPS Data Cleansing (cont.)

- Erroneous recordings: noise vs. random errors
- **Random errors** correspond to ‘possible’ values that appear to be small deviations from actual ones
- Can be smoothed using a plethora of statistical methods
  - e.g., least squares spline approximation (de Boor, 1978)



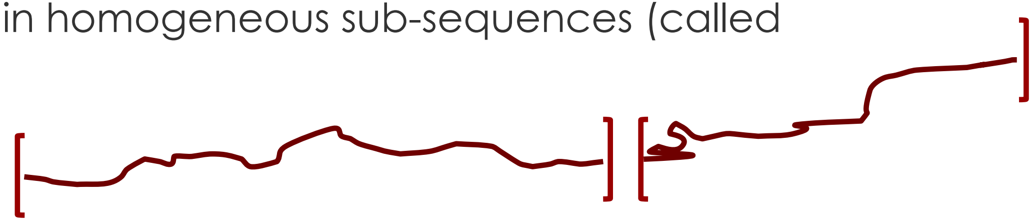
GPS Data Cleansing (cont.)

- Special case: network-constrained movement
- Requires an additional step: **map-matching**
- Several techniques (Quddus et al. 2003; 2007):
  - Geometric map-matching
  - Topological map-matching
  - Probabilistic map-matching
  - Hybrid map-matching

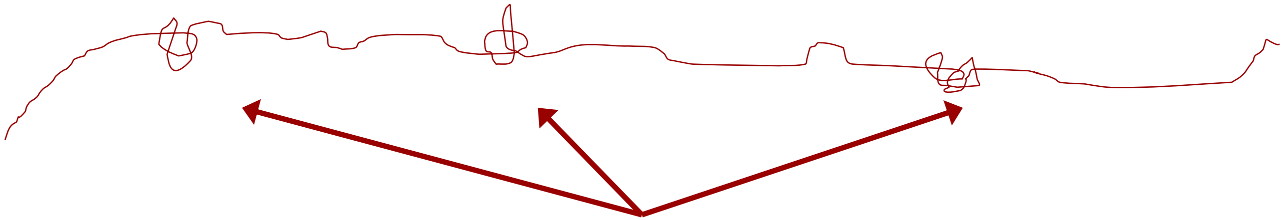


Trajectory segmentation

- Goal: **Segment sequences of points** in homogeneous sub-sequences (called **trajectories**)



- Various approaches:
  - Segmentation via raw (spatial / temporal) gap
  - Segmentation via stop discovery
  - Segmentation via prior knowledge (e.g., office / sleeping hours, arrival at ports)
  - etc.



Trajectory simplification

- The need for simplification: efficiency in storage, processing time, etc.
  - Actually, a form of data compression
- Goal: maintain the original ‘signature’ as much as possible by keeping a set of **critical points** only
- Approaches
  - Offline, i.e., multi-pass, vs.
  - Online, i.e., 1-pass

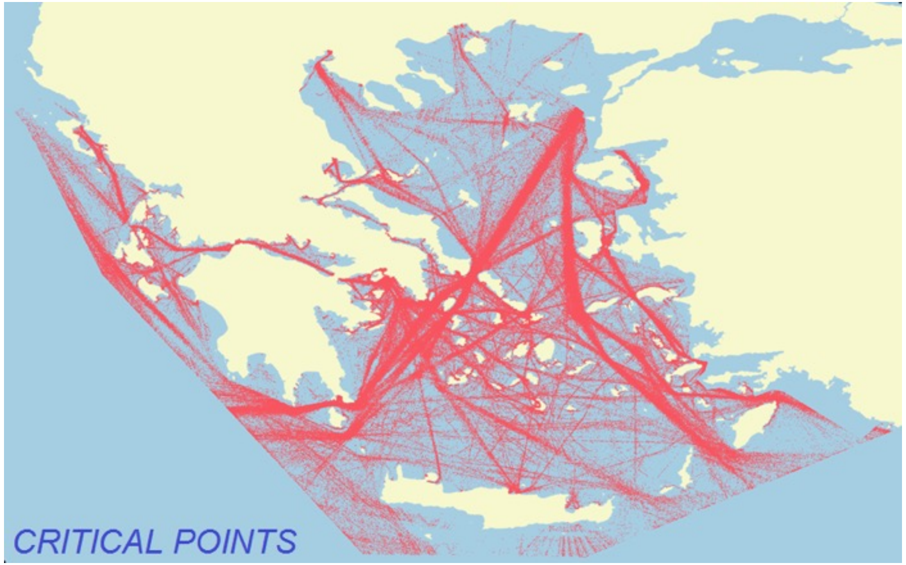


image source: aminess.eu

Trajectory simplification (cont.)

- Offline approaches:
  - top-down vs. bottom-up vs. sliding window vs. opening window
- e.g., **Synchronous Euclidean Distance – SED** (Meratnia & de By, 2004)
  - Adapts the popular Douglas & Peucker polyline simplification (1973) to the mobility domain

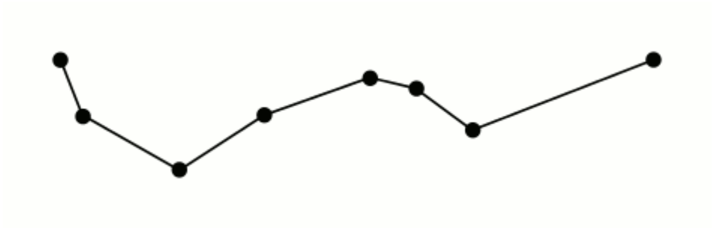
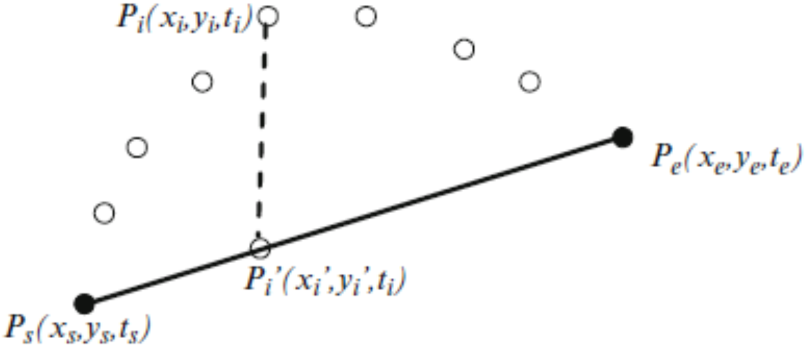
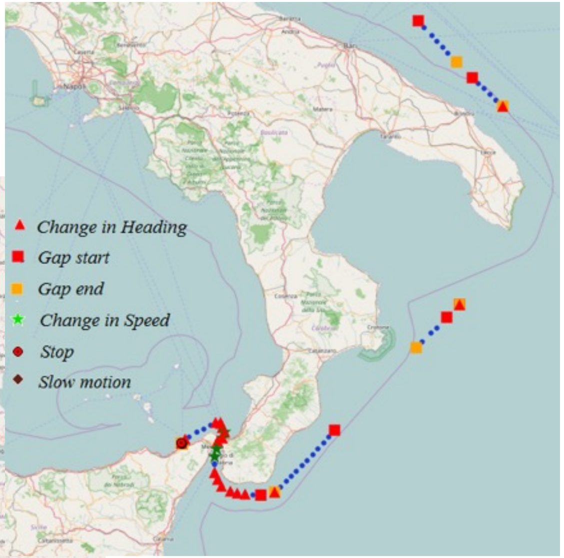


image source: https://commons.wikimedia.org/wiki/File:Douglas-Peucker_animated.gif



Trajectory simplification (cont.)

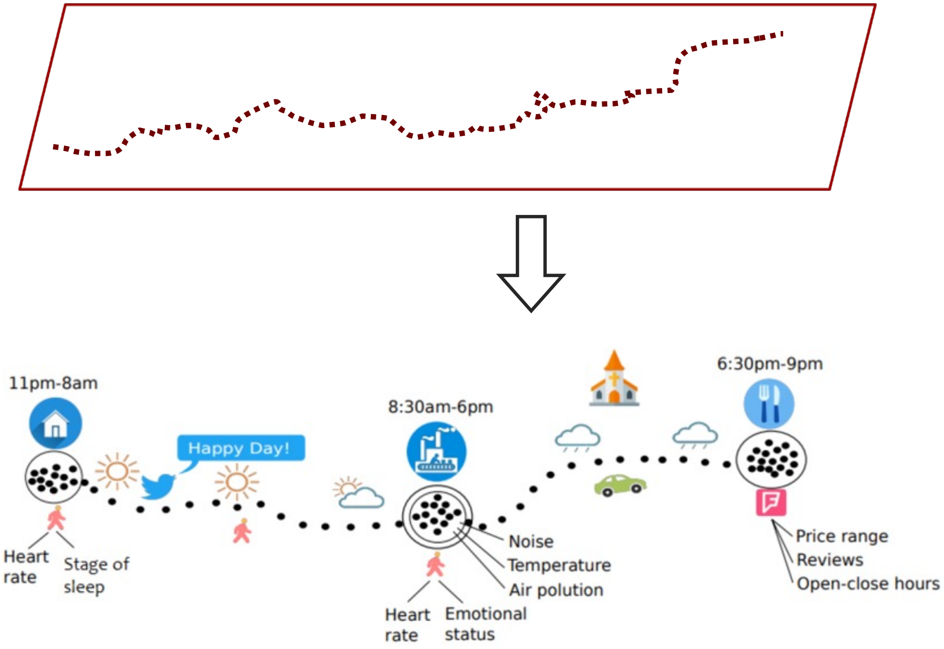
- Online approaches, e.g., **Trajectory Synopses** (Patroumpas et al. 2015; 2017)
- Maintains a **velocity vector** per moving object in order to detect **instantaneous events**
  - stop; change in velocity vector; etc.
- Tradeoff: degree of compression vs. quality of approximation



images source: datacron-project.eu

Trajectory enrichment

- From "raw" sequences (p,t) of time-stamped locations
- … to meaningful mobility tuples <where, when, what/how/why>
- **Semantic trajectory** (Yan et al. 2011; 2012, Parent et al. 2015)
  - semantically-annotated representation of the motion path of a moving object
  - **sequence of episodes (stops/moves)** along with appropriate **tags**



Source: MASTER project

3. Analyzing mobility data

Types of mobility data analytics

- Discovering **groups** and **outliers**
- Discovering **frequent routes** (hot paths) and **frequent locations** (hot spots)
- **Route prediction** tasks, etc.

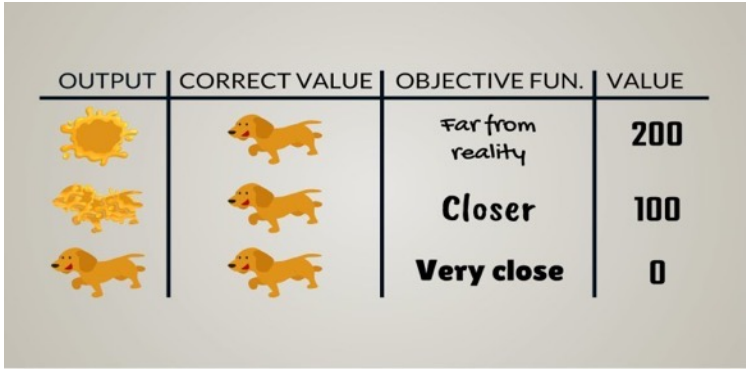
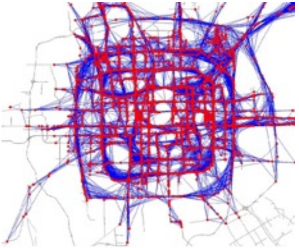
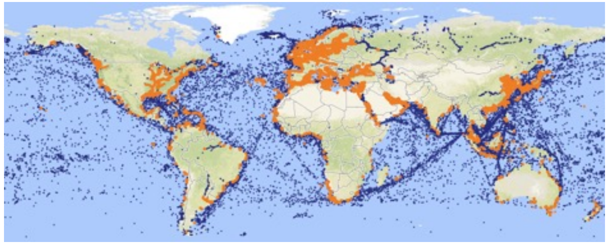


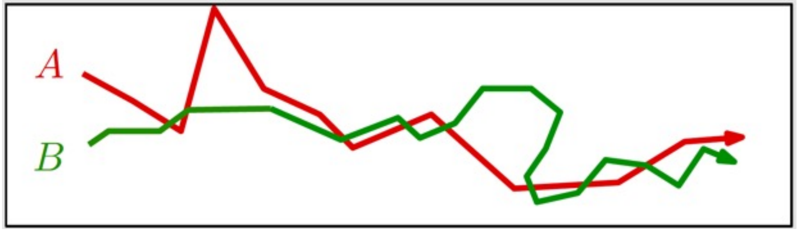
image source: kdnuggets.com





Orthogonal issue: Trajectory similarity

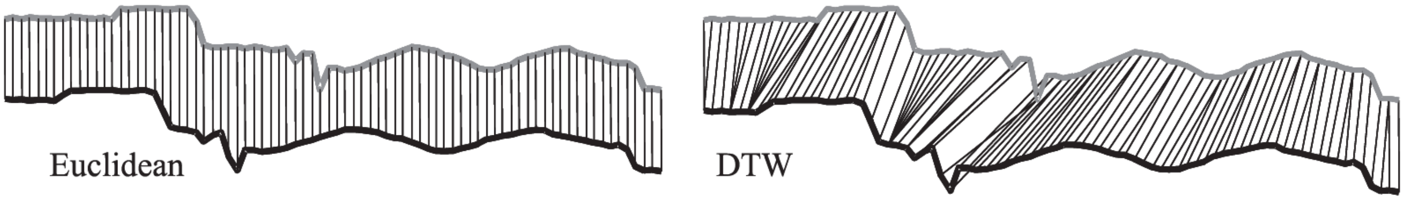
- How do we measure **similarity** between two trajectories A, B?
  - not so trivial as it sounds



- Alternative approaches:
  - Trajectory as a 2D time-series
  - Trajectory as a 2D polyline
  - Trajectory as a movement function

Trajectory as a time series

- Time series similarity has been studied extensively (e.g., Vlachos et al. 2002; Chen et al. 2005). Examples:
  - Euclidean distance, Chebyshev distance, Dynamic Time Warping (DTW),
  - Longest Common SubSequence (LCSS),
  - Edit Distance on Real sequences (EDR),
  - Edit distance with Real Penalty (ERP), etc.



Trajectory as a polyline

- **DISSIM** (Nanni & Pedreschi, 2006; Frentzos et al. 2007)
  - Extension of Euclidean distance:

$$DISSIM(R,S) = \int_{t_1}^{t_n} L_2\big(R(t), S(t)\big)dt$$

$$DISSIM(R,S) \approx \frac{1}{2}\sum_{k=1}^{n-1}\Big(\Big(L_2\big(R(t_k), S(t_k)\big) + L_2\big(R(t_{k+1}), S(t_{k+1})\big)\Big) \cdot (t_{k+1} - t_k)\Big)$$

  - DISSIM function is a metric
    - Conditions: (1) non-negativity; (2) identity of indiscernibles; (3) symmetry; (4) triangle inequality

1. $d(x,y) \geq 0$
2. $d(x,y) = 0 \Leftrightarrow x = y$
3. $d(x,y) = d(y,x)$
4. $d(x,z) \leq d(x,y) + d(y,z)$

Trajectory as a movement function

- Trajectory similarity using **Fréchet distance**, e.g. (Buchin et al. 2009; Gudmundsson et al. 2019)
  - a measure of similarity between curves that takes into account the location and ordering of the points along the curves
  - continuous mapping $\mu : A \rightarrow B$
  - distance $\max_{\alpha \in A} d(\alpha, \mu(\alpha))$

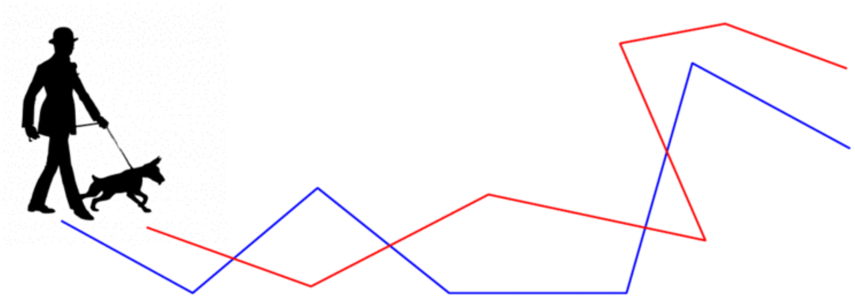


image source: https://omrit.filtser.com

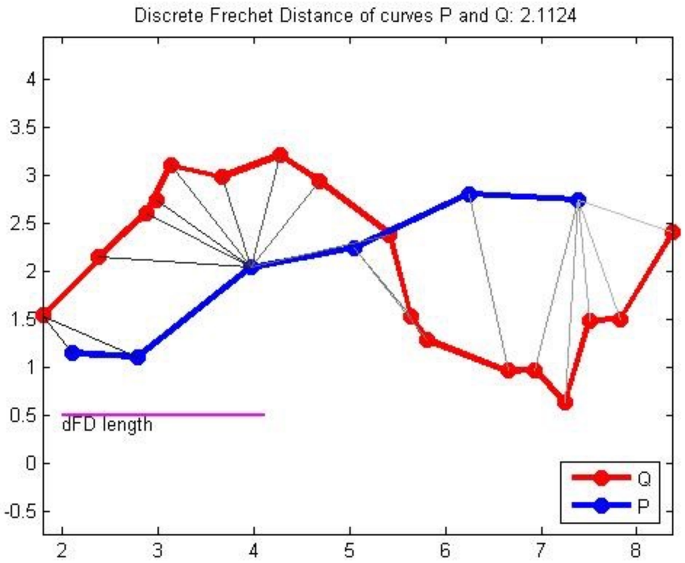
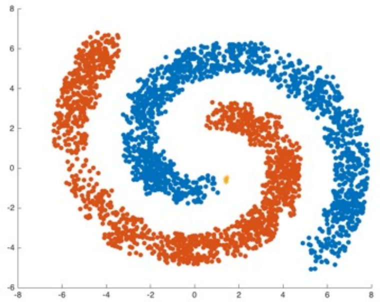


image source: mathworks.com

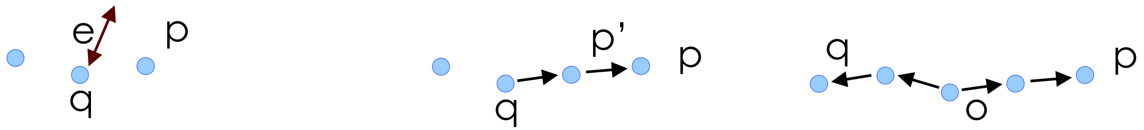
Point clustering



- **DBSCAN** (Ester et al. 1996): A density-based algorithm for discovering clusters in large spatial databases with noise
- Method parameters:
  - radius of an object's neighborhood (e)
  - minimum population within an object's neighborhood (m)
- Cores (build clusters) vs. Borders (assigned to their cores' clusters) vs. Noise

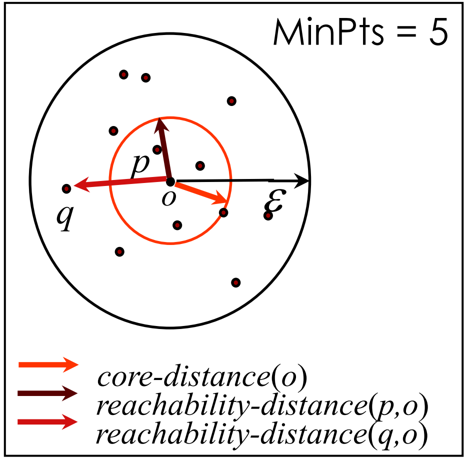
- The notion of **density reachability**
  - Directly Density-Reachable vs. Density-Reachable vs. Density Connected

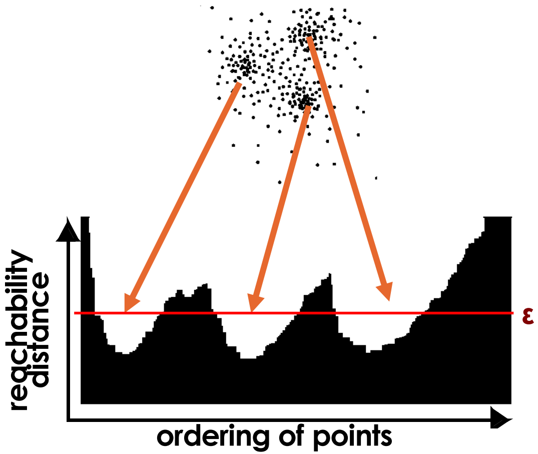
m = 3

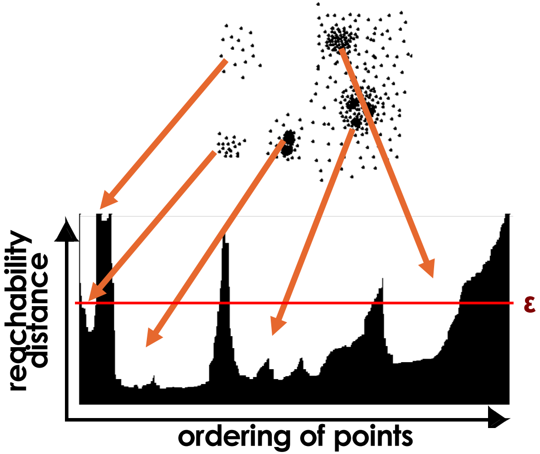


Point clustering (cont.)

- **OPTICS** (Ankerst et al. 1996): ordering points to identify the clustering structure
- The notions of **core distance** and **reachability distance**
- **Reachability plot**: partitions the dataset in a sequence of '**valleys**' (==> clusters) and '**hills**' (==> outliers)







Trajectory clustering

- Objectives:
  - Cluster trajectories w.r.t. similarity
  - Eventually, detect outliers

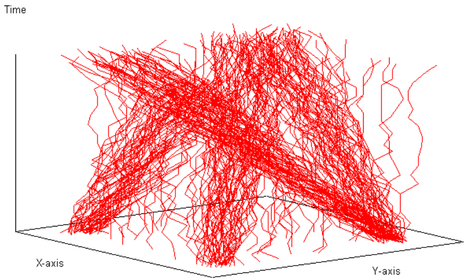
- Issues:
  - Which similarity function?
  - Upon the entire trajectories or portions (sub-trajectories?

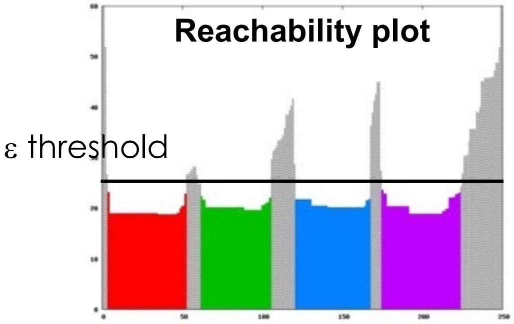
Trajectory clustering (cont.)

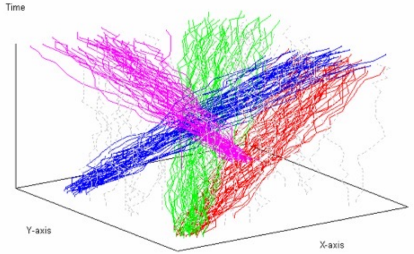
- T-OPTICS (Trajectory OPTICS) (Nanni & Pedreschi, 2006)
  - Builds upon OPTICS (Ankerst et al, 1999) and DISSIM distance function

$$DISSIM(R,S) = \int_{t_1}^{t_n} L_2(R(t), S(t))dt$$

  - The **reachability plot** produces “valleys” and “hills”
    - Valleys → clusters; Hills → outliers (noise)
    - Recall that DISSIM is a metric → indexing is allowed



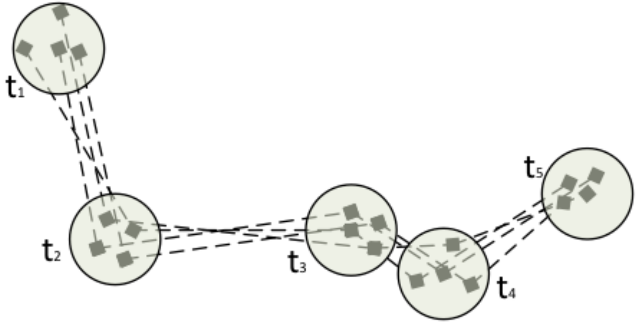


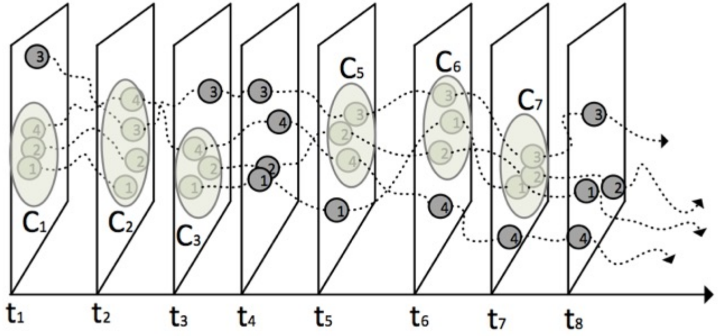


Discovering collective mobility behavior

- Detecting a large enough subset of objects moving along paths close to each other for a certain time. Main approaches:
  - Spherical-like clustering: **Flocks** (Laube et al. 2005; Gudmundsson & van Kreveld, 2006) vs.
  - Density-based clustering: **Convoys** (Jeung et al. 2008); **Swarms** (Li et al. 2010), etc.
  - Hybrid: **Evolving Clusters** (Tritsarolis et al. 2021)

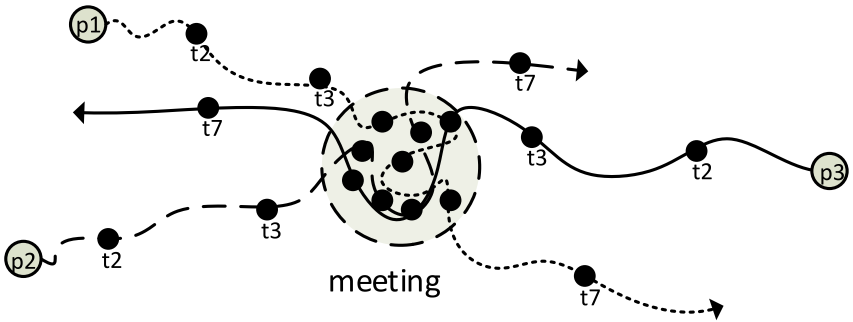
- Note: these methods work on time-aligned location sequences → need for fixed re-sampling



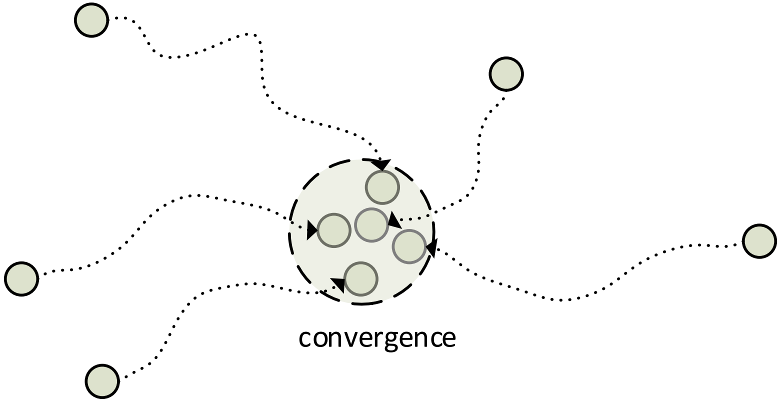


Flocks and variants

- Interesting applications of the flock/convoy pattern discovery:
  - Identify long flock patterns (**top-k longest flock pattern discovery**)
  - Discover **meetings** (fixed- vs. varying- versions)
  - Discover **convergences**
  - Discover **leaders** and **followers**

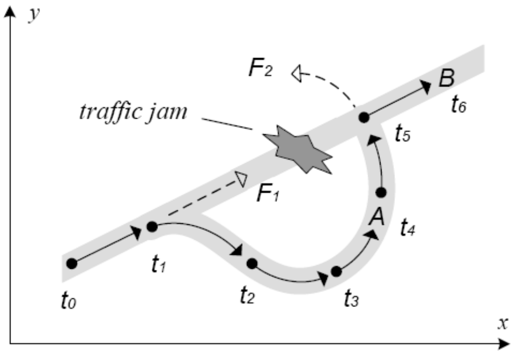


meeting

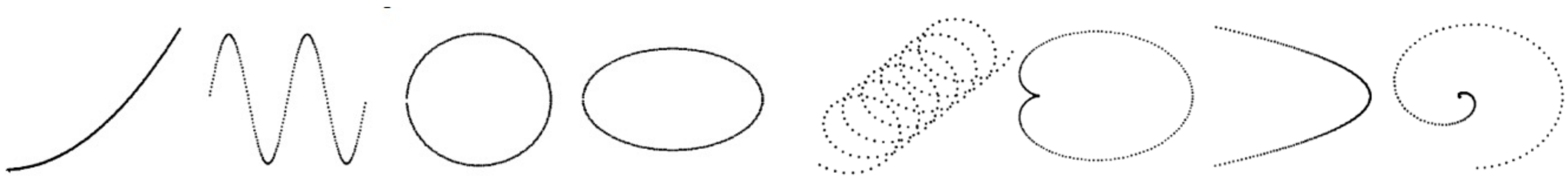


convergence

Location / Trajectory prediction



- **Prediction** aims to predict the future location(s) of (or even the entire trajectory to be followed by) a moving object.
- Two main approaches: **Formula-** vs. **Pattern-based** prediction
  - Motion function models, e.g., RMF (Tao et al. 2004)
  - vs. patterns built upon the history, e.g., Personal profiles (Trasarti et al. 2017)
  - A survey of 50+ methods: (Georgiou et al. 2018)



Location / Trajectory prediction (cont.)

- **MyWay** (Trasarti et al. 2017) maintains a Personal Mobility Data Store (PMDS) per participating person
  - How is a person moving?
    - According to his/her past movement patterns
  - What if the personal datastore is not adequate?
    - Look into the collective knowledge base
- 3 predictors: personal (red), collective (blue), hybrid (green)

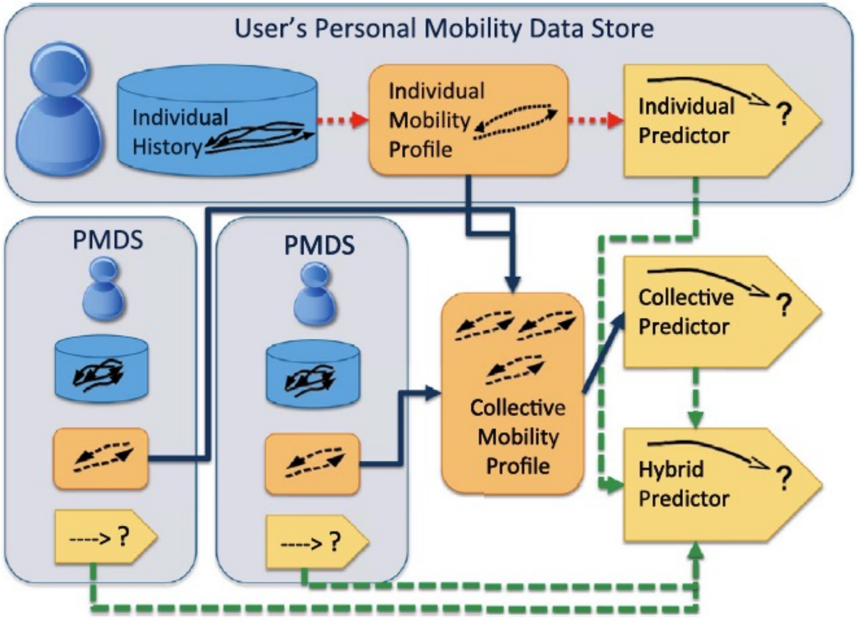


image source: kdd.isti.cnr.it

4.
Real-world use case

MDA in the maritime domain

- Vessel Route Forecasting (VRF)
- Vessel Traffic Flow Forecasting (VTFF)
- Vessel Collision Risk Assessment (VCRA)

Material based on:

- Chondrodima E., Mandalis P., Pelekis N., Theodoridis Y. (2022) **Machine Learning Models for Vessel Route Forecasting: An Experimental Comparison**. Proc. 23rd IEEE Int. Conf. MDM.
- Mandalis P., Chondrodima E., Kontoulis I., Pelekis N., Theodoridis Y. (2022) **Machine Learning Models for Vessel Traffic Flow Forecasting: An Experimental Comparison**. Proc. 3rd IEEE Int. Workshop MBDW.
- Tritsarolis A., Chondrodima E., Pelekis N., Theodoridis Y. (2022) **Vessel Collision Risk Assessment using AIS Data: A Machine Learning Approach**. Proc. 3rd IEEE Int. Workshop MBDW.

Motivation

- Vast spread of AIS-enabled maritime fleet
- Emergence of Unmanned Surface Vessels (USVs), etc.

Motivation for several analytics & forecasting tasks

- Topics of interest:
  - **Vessel Route Forecasting** (**VRF**) has a wide range of applications, such as accurate ETA calculation, collision / traffic jam assessment, etc.
  - **Vessel Traffic Flow Forecasting** (**VTFF**) is vital for maritime authorities to alleviate congestion (operational level); assists route planning purposes (strategic level)
  - **Vessel Collision Risk Assessment** (**VCRA**) is critical for maritime safety

- All the above are quite challenging due to complex and dynamic maritime traffic conditions

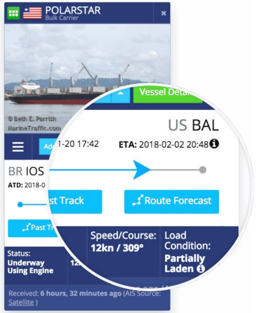


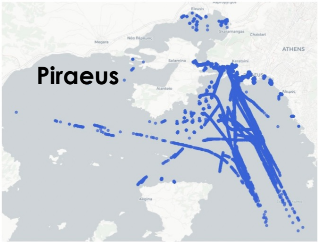
image source: marinetraffic.com

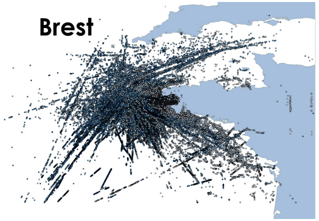
image source: www.ntnu.edu

Datasets at hand

- **Piraeus (GR)** provided by Univ. Piraeus [1]
- **Aegean-Cyclades (GR)** provided by MarineTraffic
- **Brest (FR)** provided by French Naval Academy [2]

Dataset	Piraeus	Aegean-Cyclades	Brest
Time frame	1 day (3/7/2018)	1 month (01–30/11/2018)	6 months (01/10/2015–31/03/2016)
# of records	455,145	1,720,368	16,311,185
# of distinct vessels	361	2645	5041
Sampling rate (avg.)	~ 5 min	~ 2.5 min	< 1 min
Used in	VCRA	VRF, VTFF	VRF

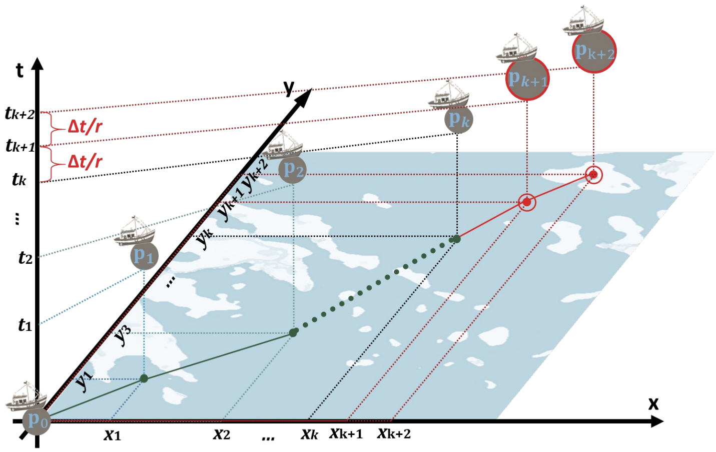




[1] https://doi.org/10.5281/zenodo.5562629; [2] https://doi.org/10.5281/zenodo.1167594

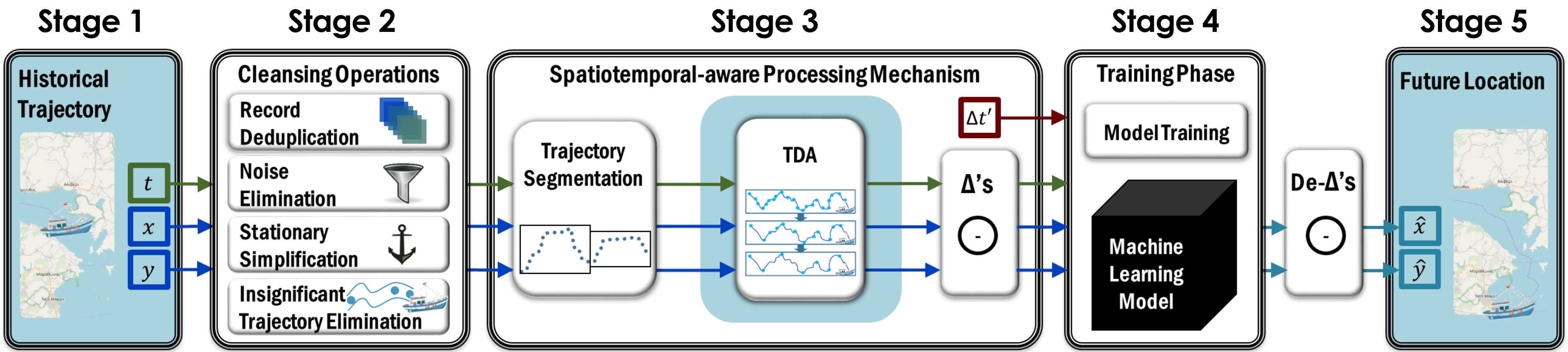
VRF – Problem formulation

- Given:
  - a vessel's trajectory [($\mathbf{p}_0$,$\mathbf{t}_0$), …, ($\mathbf{p}_k$, $\mathbf{t}_k$)] consisting of ***k*** transitions at (irregular) timepoints,
  - a number of transitions ***r***, and
  - a time duration (prediction horizon) ***Δt***



- Predict:
  - the vessel's future trajectory [($\mathbf{p}_{k+1}$,$\mathbf{t}_{k+1}$), …, ($\mathbf{p}_{k+r}$, $\mathbf{t}_{k+r}$)] consisting of ***r*** transitions at (fixed) timepoints, i.e., with sampling rate equal to ***Δt/r***

VRF – Proposed framework



- Input: a historical AIS database
- Intermediate phases: data cleansing; trajectory preprocessing; model training
- Output: a trained VRF model
  - Different ML models validated: Linear, SVMr, CART, RFT, AdaBoost, MLP, GRU, LSTM

VRF – Experimental results

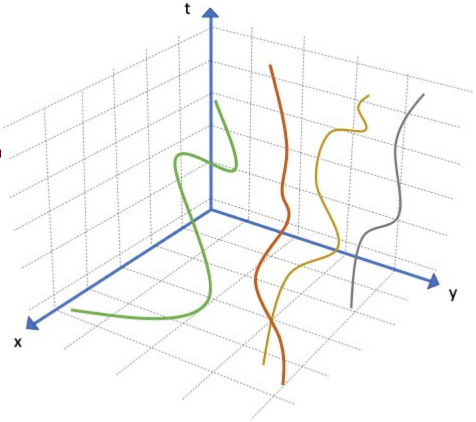
- Quality measures:
  - **Average displacement error (ADE)** – the average distance error for all predicted time steps
  - **Final displacement error (FDE)** – the distance error at the final predicted time step
- Output:
  - **LSTM** clearly outperforms all competitors

PREDICTION RESULTS FOR Δt UP TO 30 MIN. AND r UP TO 6 TRANSITIONS (UNIT: METERS)

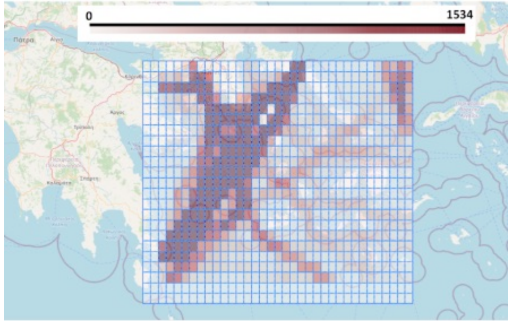
Data	Method	ADE per Δt in min. for r=6						FDE(30 min)
		5	10	15	20	25	30	
Aegean-Cyclades	Linear	867	1717	2569	3420	4271	5121	9371
	CART	340	889	1481	1916	2335	2796	5102
	RFT	221	654	1114	1506	1911	2377	4709
	AdaBoost	230	640	984	1374	1785	2217	4376
	SVMr	638	1335	2223	2938	3706	4310	7328
	MLP	180	735	1290	1782	2264	2765	5270
	GRU	79	195	337	511	727	977	2229
	LSTM	**76**	**184**	**317**	**481**	**684**	**920**	**2097**
Brest	Linear	1158	1788	2412	3030	3642	4312	7666
	CART	571	1091	1679	2218	2708	3247	5945
	RFT	286	641	1016	1445	1852	2226	4094
	AdaBoost	252	610	983	1387	1782	2159	4041
	SVMr	697	1388	2008	2668	3276	3828	6591
	MLP	677	1067	1482	1936	2403	2894	5344
	GRU	241	466	710	959	1215	1485	2832
	LSTM	**239**	**440**	**663**	**899**	**1146**	**1408**	**2719**

VTFF – Problem formulation

- Given:
  - a set of vessel trajectories ***D*** spanning in $\boldsymbol{D_s}$ (minimum bounding box of locations) in space and $\boldsymbol{D_T}$ in time,
  - a time duration (prediction horizon) ***Δt***,
  - a number of temporal transitions ***r***
  - a spatiotemporal (3D) grid that partitions $\boldsymbol{D_s}$ into grid cells of resolution ***G***×***G***, and $\boldsymbol{D_T} \cup \boldsymbol{\Delta t}$ into ***r*** time frames
- Predict:
  - The expected number of vessels (presence) in each grid cell related to ***Δt***.



Example grid: 4 x 4 x 5 space-time frames

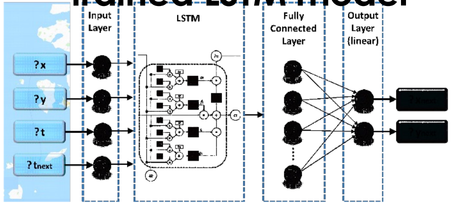


Traffic flow (Nov. 2018; G = 10km). Darker color indicates higher traffic flow.

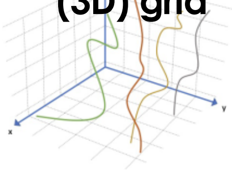
VTFF – Proposed approaches

VRF-based VTFF

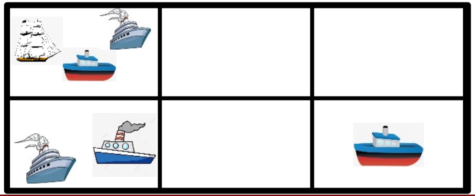
Perform VRF using a trained LSTM model



Allocate future locations into the spatio-temporal (3D) grid



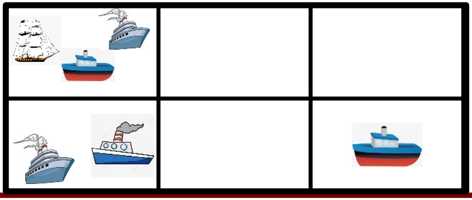
Calculate presence within each 3D grid cell



vs.

Sequence-based VTFF

Produce a sequence of vessels' presence per spatial (2D) grid cell



Train an ML model to predict the temporal evolution of each sequence

VTFF – Experimental results

$$SMAPE = \frac{1}{B}\sum_{b=1}^{B}\frac{1}{F}\sum_{t=1}^{F} 2\frac{|y_{b,t} - \hat{y}_{b,t}|}{|y_{b,t}| + |\hat{y}_{b,t}|} \qquad Jaccard = \frac{1}{B}\sum_{b=1}^{B}\frac{1}{F}\sum_{t=1}^{F}\frac{|Y_{b,t} \cap \hat{Y}_{b,t}|}{|Y_{b,t} \cup \hat{Y}_{b,t}|}$$

- **Quality measures**:
  - Symmetric Mean Absolute Percentage Error (SMAPE);
  - Jaccard similarity coefficient
- **Experiments**:
  - comparing the two approaches (Table I);
  - a closer look at the VRF-based approach (Table II)

TABLE I.
PREDICTION RESULTS (SMAPE) IN THE TESTING SET (20 BUSIEST GRID CELLS), $G = 10$KM .

VTFF strategy	Method	Time prediction horizon (min)		
		5	10	15
Flow sequence-based	XgBoost	17.72	30.41	27.43
	ARIMA	46.94	37.75	48.73
VRF-based	LSTM	6.35	16.76	28.71

TABLE II.
PREDICTION RESULTS (SMAPE, JACCARD) FOR THE VRF-BASED VTFF STRATEGY IN THE TESTING SET (ALL GRID CELLS) .

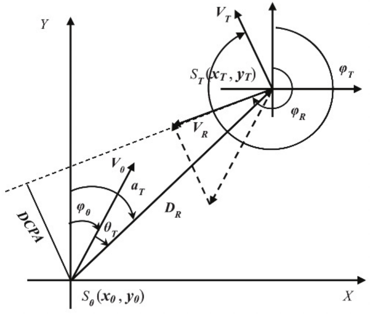
Grid cell (km)	Time frame (min)	SMAPE	Jaccard
5	5	9.57	0.95
	10	26.20	0.87
	15	44.00	0.78
10	5	4.97	0.97
	10	14.23	0.93
	15	24.90	0.87
15	5	3.52	0.98
	10	10.08	0.95
	15	18.04	0.91

VCRA – Problem formulation

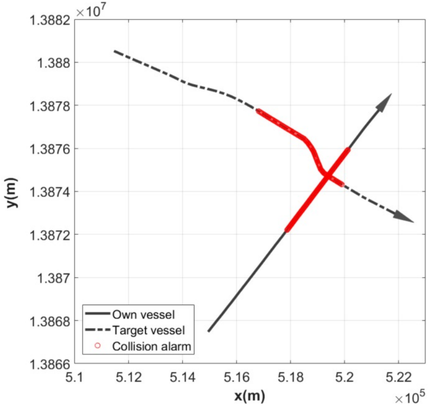
$$CRI = WU = W_{DCPA} * U_{DCPA} + W_{TCPA} * U_{TCPA} + W_D * U_D + W_B * U_B + W_K * U_K$$

$$W = [W_{DCPA}, W_{TCPA}, W_D, W_B, W_K] = [0.4457, 0.2258, 0.1408, 0.1321, 0.0556]$$

- (train a ML model in order to) estimate CRI(v_o,v_t), i.e., the collision risk index of an own vessel v_o w.r.t. a target vessel v_t that are in an encountering process, at real-time
  - Two vessels are in an ***encountering process*** during a time period, when their distance decreases along this time period and increases right after

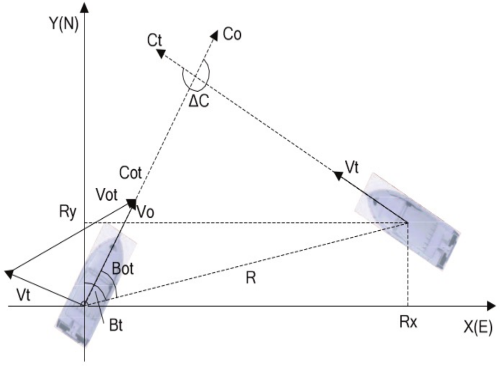


Vessel collision geometry



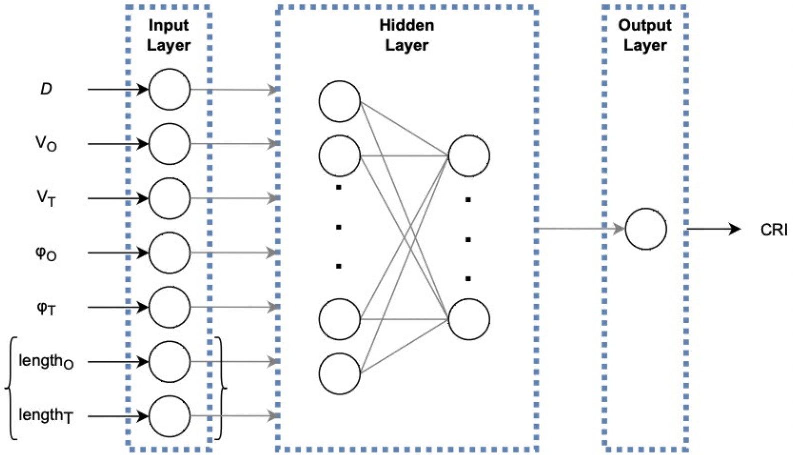
(left) Trajectories of encountering vessels in the case of crossing situation – image source: Park & Jeong 2021 [21]

(right) The moving vector diagram of encounter ships – image source: Chen et al. 2015 [7]



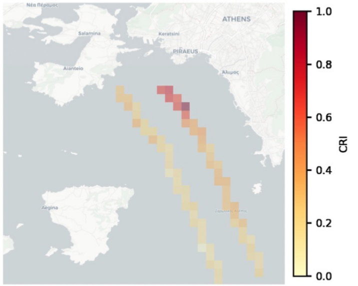
VCRA – Proposed methodology

- Given the following features for each pair (v_O,v_T) of vessels in an encountering process:
  - location (x, y), length, course φ, speed V
- Create a dataset with 5+2 features:
  - distance D, speed V_O and V_T, course $φ_O$ and $φ_T$
  - (optionally) lengthO and lengthT
- Train an MLP model with
  - two hidden layers (of 256 and 32 neurons, resp.)
  - one output: CRI(v_O,v_T)



(top) the proposed MLP-VCRA architecture

(right) the estimated CRI over cargo vessels as they approach the port of Piraeus



VCRA – Experimental results

- In terms of quality, our MLP-VCRA approach
  - Reaches 87.5% accuracy after training
  - Outperforms its competitors by a large margin
- In terms of latency* (i.e., response time)
  - Outperforms competitors and the kinematic equations (ground truth)
- Regarding the features used
  - Vessels' length is optional. Nevertheless, it marginally improves quality and latency

Method	MAE	RMSE	Response Time (msec.)
Kinematic Eq.	-	-	329 ± 11.7
SVM-VCRA [19]	0.0572	0.0945	351 ± 1.45
AFNN-VCRA [20]	0.0476	0.0934	314 ± 2.16
RVM-VCRA [21]	0.0359	0.0802	$322 \pm .744$
MLP-VCRA	**0.0179**	**0.0485**	**311 ± 1.05**

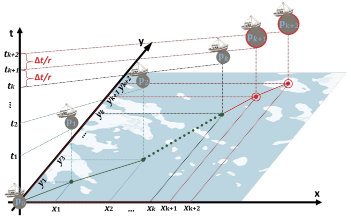
	Accuracy (%)	MAE	RMSE	response time (msec.) (min.; med.; max.;)
MLP-VCRA ($length_O$)	86.827	0.0179	0.0485	196; 354; 680
MLP-VCRA ($length_T$)	87.134	0.0167	0.0480	201; 360; 684
MLP-VCRA ($length_{O,T}$)	**87.514**	**0.0165**	**0.0472**	**192; 332; 638**
MLP-VCRA (w/out $length_{O,T}$)	87.207	0.0189	0.0478	197; 369; 695

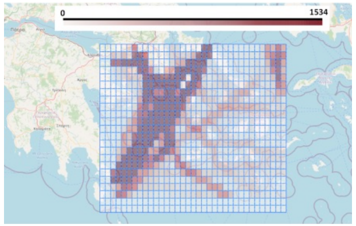
* Machine used: a single node with 8 CPU cores and 16 GB of RAM

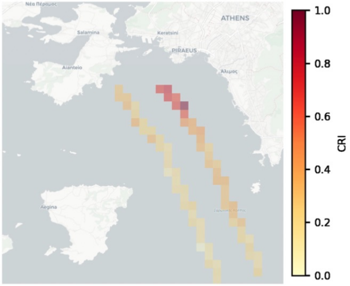
Conclusions

- Taking advantage of the wealth of AIS data, we studied several popular ML methods w.r.t. their prediction accuracy on three maritime analytics problems.
- Our experimental results show that
  - VRF: LSTM outperforms competition
  - VTFF: the VRF-based solution is quite promising
  - VCRA: the MLP-VCRA approach avoids CRI calculations and outperforms competition

- As such, the proposed VRF/VTFF/VCRA models are strong candidates to be used as references for MTS purposes







5. Summary

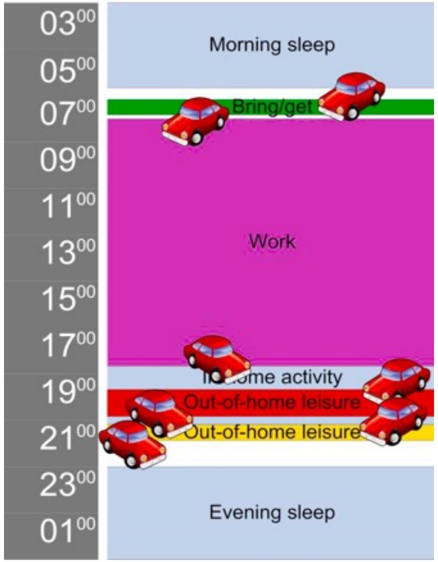
Summary

- The field of **MDA** has many success stories to narrate on*:
  - **Data management** - access methods, query processing techniques, DBMS extensions (the so-called, Moving Object Databases)
  - **Data exploration** – data mining techniques (clusters, flocks, convoys, T-patterns, hot spots, etc.)
  - … mostly based on the sampled spatio-temporal coordinates (x-, y-, z-, t-) of moving objects

* see e.g. (Pelekis & Theodoridis 2014)

Summary (cont.)

- The new era that emerges is around two keywords:
  - **Semantically-annotated trajectories*** – information about when, where, what, how, why
  - **Extreme-scale mobility data**** – voluminous, streaming, disperse information about objects' movement



* Parent C, et al. (2013): Semantic trajectories modeling and analysis. ACM Computing Surveys, 45(4).

** Vouros GA, et al. (2018) Big data analytics for time critical mobility forecasting: recent progress and research challenges. In Proceedings of EDBT.

Acknowledgments

Ack EU- and Greek-level support through a series of grants:

- **VesselAI**– Enabling Maritime Digitalization by Extreme-scale Analytics, AI and Digital Twins . 2021-23 [vessel-ai.eu]
- **i4Sea**– Big Data in Monitoring and Analyzing Sea Area Traffic: innovative ICT and Analysis Models. 2018-21 [i4sea.eu]
- **Track & Know** – Big Data for Mobility Tracking Knowledge Extraction in Urban Areas. 2018-20 [trackandknowproject.eu]
- **MASTER** – Multiple Aspect Trajectory Management and Analysis, 2018-22 [master-project-h2020.eu]
- **datAcron** – Big Data Analytics for Time Critical Mobility Forecasting, 2016-18 [datacron-project.eu]
- **DART** – Data-Driven Aircraft Trajectory Prediction Research. 2016-18 [dart-research.eu]

References (1/4)

- Alvares LO, et al (2007) A model for enriching trajectories with semantic geographical information. In Proceedings of GIS.
- Ankerst M, et al (1999) OPTICS: Ordering points to identify the clustering structure. In Proceedings of SIGMOD.
- de Boor C (1978) A practical guide to splines. Springer-Verlag.
- Buchin K, et al (2009) Finding long and similar parts of trajectories. In Proceedings of SIGSPATIAL-GIS.
- Cao H, et al (2007) Discovery of periodic patterns in spatiotemporal sequences. IEEE Transactions on Knowledge and Data Engineering, 19(4).
- Chen L, et al (2005) Robust and fast similarity search for moving object trajectories. In Proceedings of SIGMOD.
- Claramunt C, et al (2017) Maritime data integration and analysis: recent progress and research challenges. In Proceedings of EDBT.
- Douglas D, Peucker T (1973) Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. The Canadian Cartographer, 10(2).
- Ester M, et al (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of KDD.
- Frentzos E, et al (2007) Index-based most similar trajectory search. In Proceedings of ICDE.

References (2/4)

- Georgiou H, et al (2018) Moving objects analytics: survey on future location & trajectory prediction methods. Technical Report. arXiv:1807.04639.
- Georgiou H, et al (2019) Semantic-aware aircraft trajectory prediction using flight plans. Int. J. Data Sci. and Analytics.
- Georgiou H, et al (2021) Driver behaviour profiling based on trajectory analytics. Technical Report. DOI: 10.5281/zenodo.5708676
- Giannotti F, et al (2007) Trajectory pattern mining. In Proceedings of KDD.
- Gudmundsson J, van Kreveld MJ (2006) Computing longest duration flocks in trajectory data. In Proceedings of GIS.
- Gudmundsson J, et al. (2019) Fast Fréchet distance between curves with long edges. Int. J. Comput. Geom. & Applications, 29(02).
- Jeung H, et al (2008) Discovery of convoys in trajectory databases. In Proceedings of VLDB.
- Laube P, et al (2005) Discovering relative motion patterns in groups of moving point objects. Int. J. Geo, Info. Sci., 19(6).
- Lee JG, et al (2008) Trajectory outlier detection: A partition-and-detect framework. In Proceedings of ICDE.
- Lee JG, et al (2007) Trajectory clustering: a partition-and-group framework. In Proceedings of SIGMOD.
- Li Z, et al (2010) Swarm: Mining relaxed temporal moving object clusters. Proceedings of VLDB, 3(1).
- Lin N, et al (2014) An overview on study of identification of driver behavior characteristics for automotive control. Math. Probl. in Eng.

References (3/4)

- Meratnia N, de By RA (2004) Spatiotemporal compression techniques for moving point objects. In Proceedings of EDBT.
- Monreale A, et al (2009) WhereNext: a location predictor on trajectory pattern mining. In Proceedings of KDD.
- Nanni M, Pedreschi D (2006) Time-focused clustering of trajectories of moving objects. J. Intelli. Info. Sys., 27(3).
- Palma AT, et al (2008) A clustering-based approach for discovering interesting places in trajectories. In Proceedings of ACM-SAC.
- Panagiotakis C, et al (2012) Segmentation and sampling of moving object trajectories based on representativeness. IEEE Trans. Knowl. and Data Eng., 24(7).
- Parent C, et al (2013) Semantic trajectories modeling and analysis. ACM Computing Surveys, 45(4), Article no. 42.
- Patroumpas K, et al (2017) Online event recognition from moving vessel trajectories. GeoInformatica, 21(2).
- Patroumpas K, et al (2015): Event Recognition for Maritime Surveillance. In Proceedings of EDBT.
- Pelekis N, et al (2010) Unsupervised trajectory sampling. In Proceedings of ECML-PKDD.
- Pelekis N, et al (2017a) In-DBMS sampling-based sub-trajectory clustering. In Proceedings of EDBT.
- Pelekis N, et al (2017b) On temporal-constrained sub-trajectory cluster analysis. Data Mining and Knowl. Disc., 31(5).
- Pelekis N, Theodoridis Y (2014) Mobility data management and exploration. Springer.
- Quddus MA, et al (2007) Current map-matching algorithms for transport applications: state-of-the-art and future research directions. Transp. Res. Part C: Emerging Technologies, 15(5).
- Quddus MA, et al (2003) A general map matching algorithm for transport telematics applications. GPS Solutions, 7(3).

References (4/4)

- Spiliopoulou M, et al (2006) MONIC: Modeling and monitoring cluster transitions. In Proceedings of KDD.
- Tampakis P, et al. (2019) Scalable distributed sub-trajectory clustering. In Proceedings of IEEE Big Data.
- Tampakis P, et al. (2020) Distributed subtrajectory join on massive datasets. ACM Trans. Spatial Algorithms & Systems, 6(2), article no. 8.
- Tao Y, et al (2004) Prediction and indexing of moving objects with unknown motion patterns. In Proceedings of SIGMOD.
- Theodoridis C, Theodoridis Y (2021) Sustainable Urban Mobility in the Post-Pandemic Era. Technical Report. arXiv:2109.12982
- Trasarti R, et al (2017) MyWay: location prediction via mobility profiling. Inf. Syst. 64, pp. 350-367.
- Tritsarolis A, et al (2021) Online discovery of co-movement patterns in mobility data. Int. J. Geogr. Inf. Sci. 35(4).
- Vlachos M, et al (2002) Discovering similar multidimensional trajectories. In Proceedings of ICDE.
- Vouros GA, et al (2018) Big data analytics for time critical mobility forecasting: recent progress and research challenges. In Proceedings of EDBT.
- Wang W, et al (2019) Driving style analysis using primitive driving patterns with Bayesian nonparametric approaches. IEEE Trans Int. Transp. Sys. 20(8).
- Yan Z, et al (2011) SeMiTri: A Framework for Semantic Annotation of Heterogeneous Trajectories. In Proceedings of EDBT.
- Yan Z, et al (2012) Semantic trajectories: Mobility data computation and annotation. ACM Trans. Intelligent Systems and Technology, 9(4), Article no. 49.