1

AN INTRODUCTION TO BIG DATA ANALYTICS

Erik Hoel

Environmental Systems Research Institute, Redlands, California, USA

Big data analytics, in the context of geospatial data, employs distributed computing using advanced tools that support spatiotemporal analysis, spatial statistics, and machine learning algorithms and techniques (e.g., classification, clustering, and prediction) on very large spatiotemporal data sets to visualize, detect patterns, gain deeper understandings, and answer questions. In this chapter, the key definitions, domain specific problems, analysis concepts, current technologies and tools, and remaining challenges are discussed.

1.1. Overview

Big data analytics involves analyzing large volumes of varied data, or big data, to identify and understand patterns, correlations, and trends that ordinarily are invisible due to the volumes involved in order to allow users and organizations to make better decisions. These analytics, in the context of geospatial data, commonly involve spatial processing, sophisticated spatial statistical algorithms, and predictive modeling. Big data can be obtained from a wide variety of sources; this includes sensors (both

Big Data Analytics in Earth, Atmospheric, and Ocean Sciences, Special Publications 77, First Edition. Edited by Thomas Huang, Tiffany C. Vance, and Christopher Lynnes.

DOI: 10.1002/9781119467557.ch1

stationary and moving), aerial and satellite imagery, Lidar, videos, social networks, website activity, sales transaction records, and real-time stock trading transactions. Users and data scientists apply big data analytics to evaluate these large collections of data, data with volumes that traditional analytical systems are unable to accommodate (Miller & Goodchild, 2014). This is particularly the case with unstructured or semistructured data (such data types are problematic with data warehouses, which often utilize relational database concepts and work with structured data).

To address these complex demands, many new analytic environments and technologies have been developed. This includes distributed processing infrastructures such as Spark and MapReduce (Dean & Ghemawat, 2008; Garillot &Maas, 2018; Zaharia et al., 2010), distributed file stores, and NoSQL databases (Alexander & Copeland, 1988; DeWitt & Gray, 1992; Klein et al., 2016; NoSQL, 2022; Pavlo & Aslett, 2016). Many of these technologies are available in open-source software frameworks, such as Apache Hadoop (2018), that can be used to process huge data sets with clustered systems.

When working with big data, there is a collection of objectives that users have when performing big data analytics (Marz & Warren, 2013; Mysore et al., 2013). These include

1. *Discovering value from big data*. Visualize and analyze big data in a way that reveals patterns, trends, and relationships that traditional reports and spatial processing do not. Data may exist in many disparate places, streams, or web logs.
2. *Exploiting streaming data*. Filter and convert raw streaming data from various sources, which contain geographical elements, into geographic layers of information. The geographical layers can then be used to create new, more useful maps and dashboards for decision making.
3. *Exposing geographic patterns*. Use maps and visualization to see the story behind the data. Examples of identifying geographical patterns include retailers seeing where promotions are most effective and where the competition is, banks understanding why loans are defaulting and where there is an underserved market, climate-change scientists determining the impact of shifting weather patterns.
4. *Finding spatial relationships*. Seeing spatially enabled big data on a map allows you to answer questions and ask new ones. Where are disease outbreaks occurring? Where is insurance risk greatest given recently updated population shifts? Geographic thinking adds a new dimension to big data problem solving and helps you make sense of big data.

5. *Performing predictive modeling*. Predictive modeling using spatially enabled big data helps you develop strategies from if/then scenarios. Governments can use it to design disaster response plans. Natural resource managers can analyze recovery of wetlands after a disaster. Health service organizations can identify the spread of disease and ways to contain it.

1.1.1. What Differentiates Spatial Big Data

Spatial big data are differentiated from standard (nonspatial) big data by the presence of spatial relationships, geostatistical correlations, and spatial semantic relations (this can be generalized to include the temporal domain (Hägerstrand, 1970). Spatial big data offer additional challenges beyond what is encountered with more traditional big data. Spatial big data are characterized by the following (Barwick, 2011):

- *Volume*. The quantity of data. Spatial big data also include global satellite imagery, mobile sensors (smart phones, GPS trackers, and fitness monitors), and georeferenced digital camera imagery.
- *Variety*. Spatial data are composed of 2D or 3D vector or raster imagery. Spatial data are more complex and subsume the types found with conventional big data.
- *Velocity*. Velocity of spatial data is significant given the rapid collection of satellite imagery in addition to mobile sensors.
- *Veracity*. For vector data (points, lines, and polygons), the quality and accuracy vary. Quality is dependent upon whether the points have been GPS determined, determined by unknown origins, or determined manually. Resolution and projection issues can also alter veracity. For geocoded points, there may be errors in the address tables and in the point location algorithms associated with addresses. For raster data, veracity depends on accuracy of recording instruments in satellites or aerial devices, and on timeliness.
- *Value*. For real-time spatial big data, decisions can be enhanced through visualization of dynamic change in such spatial phenomena as climate, traffic, social-media-based attitudes, and massive inventory locations. Exploration of data trends can include spatial proximities and relationships.

Once spatial big data are structured, formal spatial analytics can be applied, such as spatial autocorrelation, overlays, buffering, spatial cluster techniques, and location quotients.

1.2. Definitions

The terms in Table 1.1 are referenced in this chapter and are included here to facilitate a more rapid understanding of the general concepts discussed later.

Table 1.1 Terms for understanding general concepts

Amazon Web Services	(AWS) A secure, on-demand, cloud computing platform where users pay for the computing resources that they consume (e.g., computing, database storage, and content delivery).
Artificial Intelligence	Computer systems or machines that are able to perform tasks and mimic behavior that normally requires human intelligence, such as visual perception, speech recognition, and language translation.
Big Data as a Service (BDaaS)	Cloud-based hardware and software services that support the analysis of large or complex data sets. These services can provide data, analytical tools, event-driven processing, visualization, and management capabilities.
Cloudera	A software company that provides a software platform that can run either in the cloud or on-prem, supporting data warehousing, machine learning, and big data analytics. The company is a major contributor to the Apache Hadoop platform (e.g., Avro, HBase, Hive, and Spark).
Computer Vision	A scientific discipline that focuses on the acquisition, extraction, analysis, and understanding of information obtained from either single or multidimensional image or video data.
Data as a Service (DaaS)	Built on top of software as a service, data are provided to users on demand for further processing and analysis. The centralization of the data enables higher quality curated data at a lower cost to the client.
Databricks	A company that provides a cloud-based platform for working with Apache Spark. Databricks traces it origins to the AMPLab project at Berkeley that evolved into an open-source distributed computing framework for working with big data.
Data Mining	The process of discovering and extracting hidden patterns and knowledge found in big data using methods and techniques that are commonly associated with database management, machine learning, and statistics.

Table 1.1 (*continued*)

Deep Learning	A subfield of machine learning that focuses on algorithms and computational architectures that mimic the structure of the brain (commonly termed artificial neural networks). Recent advances in large-scale distributed processing have enabled the development and use of very large neural networks.
Elastic Compute Cloud (EC2)	Infrastructure within Amazon Web Services (AWS) that provides scalable computing capacity; clients can develop, deploy, and run their own applications. EC2 is elastic and allows clients to scale their compute and storage up or down as necessary.
Hadoop	An open-source framework and set of software modules that enable users to solve problems on big data sets using a distributed cluster of hardware resources. This includes distributed data storage and computation using the MapReduce programming model. Apache Hadoop was originally inspired by Google's work in the distributed processing domain.
HDFS	A distributed and scalable file system and data store that is part of Apache Hadoop. HDFS stores big data files across a cluster of machines and supports high reliability by replication of the data across different nodes in the cluster.
Hive	Data warehouse software module in Apache Hadoop that facilitates querying and analyzing big data stored in HDFS in a distributed and replicated manner using a SQL-like language termed HiveQL.
IBM Cloud	A set of cloud computing capabilities and services that provides capabilities including Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS).
Infrastructure as a Service (IaaS)	A type of cloud computing infrastructure that virtualizes computing resources, storage, data partitioning, scaling, and networking. Unlike Software as a Service (SaaS) or Platform as a Service (PaaS), IaaS clients must maintain the applications, data, middleware, and operating system.
Machine Learning	A subset of artificial intelligence where software systems can automatically learn and improve without any explicit programming, relying upon statistical methods for pattern detection and inference. Machine learning software creates statistical models using sample data in order to make decisions or predictions.

(Continued)

Table 1.1 (*continued*)

MapReduce	A programming model, originally developed at Google, that is often used when processing big data sets in a distributed manner. MapReduce programs contain a map procedure where data can be sorted and filtered, and a reduce procedure where summary operations are performed. MapReduce systems, such as Apache Hadoop, are responsible for managing communications and data transfer among the collection of distributed processing nodes.
Microsoft Azure	A cloud computing service from Microsoft for creating, deploying, and managing applications using data centers managed by Microsoft. Hundreds of services are available that provide functionality related to compute, data management, messaging, mobile, and storage capabilities.
Natural Language Processing (NLP)	A portion of artificial intelligence that focuses on enabling computers to understand and communicate (including language translation) through human language, both written and spoken.
NoSQL data stores	A non-SQL or non-relational database that provides a mechanism for storage and retrieval of data. NoSQL data stores often trade consistency in favor of availability, speed, horizontal scalability, and partitionability.
Oracle Cloud	A collection of cloud computing services from Oracle providing servers, storage, network, applications, and services using Oracle-managed data centers. The Oracle Clouse provides Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS), and Data as a Service (DaaS).
Pig	An Apache platform to develop programs for analyzing big data sets that run on Apache Hadoop using a high-level language (Pig Latin). Pig can be used to develop functionality that runs as MapReduce, Tez, or Spark jobs.
Platform as a Service (PaaS)	A category of cloud computing service that allows clients to develop, deploy, run, and manage applications without needing to build or maintain the cloud computing infrastructure. Unlike software as a service (SaaS), the client is responsible for maintaining the applications and data.

Table 1.1 (*continued*)

Predictive Analytics	A group of statistical and machine learning algorithms that are used to predict the likelihood of future or other unknown events based upon existing historical data.
Real-time Data Processing	A collection of software and hardware that processes data on-the-fly and is subjected to a constraint where responses must be provided within a short interval of time (e.g., fractions of a second), independent of system or event data load.
Redshift	A column-oriented, fully managed, data warehouse for big data. Redshift is similar to other columnar NoSQL databases as it is intended to scale out with distributed clusters of low-cost hardware.
Simple Storage Service (S3)	An object storage service offered by Amazon Web Services (AWS); it is intended to store any type of data (objects) that can later be used for big data analytic processing.
Software as a Service (SaaS)	A category of cloud computing service that allows clients to license applications, web-based software, on-demand software, and hosted software. The delivery model is on a subscription basis and is centrally hosted. Differing from Platform as a Service (PaaS), SaaS does not require client to manage either data or software.
Spark	An analytic engine and cluster-computing framework, part of Apache Hadoop, that supports applications that run across a distributed cluster. Originally developed at Berkeley in 2009, it provides a framework for programming clusters of machines with data parallelism.
Speech Recognition	A collection of methodologies and techniques that enables the recognition and transformation of spoken language into text for further computational processing.
Storm	A real-time, distributed, high-volume, stream-processing framework for big data. It is part of the Apache Hadoop open-source framework.
Stream Processing	A computer programming paradigm (similar to dataflow programming), where given a sequence of data (a stream), a series of pipelined operations (or kernel functions) is applied to each element in the stream.

1.3. Example Problems

There are a significant number of industries and application domains that benefit from spatiotemporal big data analytics (Hey et al., 2009). As the sheer number of processes and technologies that are collecting spatial data grows, the ubiquity and significance of the data have grown. Spatial big data analytics has wide applicability and value across numerous domains; a few of these are the following.

1.3.1. Agriculture

Farmers can use spatial big data analytics to detect and analyze patterns in weather data, correlated with historical crop yields, surface topography, and soil characteristics. This helps farmers determine the best seed varieties to use and times and places to plant crops in order to maximize yields. In addition, the distribution of fertilizer can be optimized based upon historical information. Tractor and heavy equipment movement can also be tracked via GPS and incorporated into the logistic optimization analytics, and the areas of usable and productive land within a field can be identified.

1.3.2. Commerce

Commercial retailers have always used local shopping patterns and demographics to drive marketing strategies and site selection. However, retailers can now use spatial big data analytics to analyze the locations and characteristics of customers along with social media conversations and browsing behavior in order to better understand customers' needs. Retailers can essentially build a richer and more useful understanding and relationship with their customer base. New store site selection on regional or national levels can be optimized based on the locations of customers, competitors, and other nontraditional data.

1.3.3. Connected Cars

Developers of systems for connected cars and autonomous vehicles can use spatial big data analytics to provide accurate situational awareness to drivers and vehicles about their surrounding environment.

Systems can apply analytics capabilities such as road snapping, predictive road snapping, change detection of objects sensed by the vehicle but not on the map, and accident prediction. This is all under the topic of improved vehicle reliability and passenger safety.

1.3.4. Environment

Environmental organizations can employ spatial big data analytics to answer a number of important questions including whether there are spatiotemporal correlations between species observations (this can be by geographic area or species).

1.3.5. Financial Services

In the financial services/insurance industry, spatial big data analytics are used to overlay weather data with claim data to assist companies in detecting possible instances of fraud. In other contexts, non-traditional data sources like satellite imagery are combined with traditional topographic data sources to identify the potential risk of offering flood insurance. Insurers can also assess spatial relationships between their insurance portfolios and past hazards to balance risk exposure. Finally, banks can use spatiotemporal historical transaction data to help them detect evidence of fraud.

1.3.6. Government Agencies

National and regional government agencies would like to use spatial big data analytics to process and overlay nationwide data sets containing land use; parcels; planning information; geological informational, and environmental data in order to create information products that can be used by analysts, scientists, and policy makers to make better policy decisions.

1.3.7. Health Care

Public health agencies can use spatial big data analytics to see how far patients are from health facilities helping them evaluate access to care. Hospital networks can determine the density of hospitals in certain areas

to identify gaps and opportunities. They can also measure the prevalence of certain habits and illnesses in the community using demographic data. Public health agencies can also utilize tracking data to perform contact tracing of infected individuals to identify who they have been in contact with in the past. The contact information can then be utilized to help reduce the infections in the general population. Proximity tracing is a variant in which contact is specified using a proximity-based filtering criteria (e.g., spatial and temporal range) in order to identify potential contact events.

1.3.8. Marketing

Geospatial big data analytics is frequently used in corporate marketing for prospect and customer segmentation. Data from body sensors (e.g., smart phones, smart watches, fitness monitors) can be used to segment the customer base according to physical activity or behavioral patterns and deliver advertising in a targeted manner. Companies also want to be able to identify where their customers are in relation to their competitors' customers. This allows them to identify areas where they are losing the market and help determine where they need to focus their marketing efforts.

1.3.9. Mining

Mining companies can apply spatial big data analytics to perform complex vehicle tracking analysis to find ways to better manage equipment moves. For example, they can analyze patterns of equipment locations when braking, and they can review shock absorption, RPM changes, and other telematics information. They can also analyze geochemical sample results.

1.3.10. Petroleum

Spatial big data analytics enable petroleum companies to identify suitable areas for exploration based upon historical production, geographic composition, and competitor activity (including leasing activity). Spatial big data analytics can also be used to review historical production data to assess reservoir production over time. Vehicle tracking data can be analyzed to determine time spent on both commercial and noncommercial roads. They can also review vessel tracks over offshore blocks using AIS vessel tracking information.

1.3.11. Retail

Retailers can use spatial big data analytics to model retail networks and help them select the best sites to optimize their store network. Analytic results can be used to create customer profile maps, allowing retailers to better understand customer behavior and the factors that influence their behavior. Retailers also want to spatially analyze the types of products that consumers are buying based upon seasonal and weather-related stimuli. This often incorporates promotions and sale activity. The spatiotemporal analysis can extend to a very fine-grained level, for example, hourly sales activity on Black Friday.

1.3.12. Telecommunications

Telecommunications companies can use spatial big data analytics to review spatial trends in bandwidth usage over time to help plan new network deployments. They can analyze spatial patterns in consumer habits, spending patterns, demographics, and service purchases to improve marketing, define new products, and help plan network expansions. Customer service departments can correlate network problems and trouble tickets with customer complaints or cancellations to determine where and when service issues have led to customer dissatisfaction. Call detail records can be used to identify areas where cellular service is problematic (quality, speed, coverage), both temporally and spatially.

1.3.13. Transportation

With spatial big data analytics, commercial delivery companies can reconstruct vehicle routes from millions of individual position reports to check for routing inefficiencies and identify incidents of unsafe speeding and braking. This level of visibility into past trips helps them develop strategies to improve efficiency and safety. Transportation planners can also use spatial big data analytics to aggregate, visualize, and analyze historical crash data for metropolitan areas, helping them identify unsafe road conditions. State and regional transportation agencies can analyze and model traffic slowdowns and congestion in order to optimize future road construction and rapid transit planning activities. City mobility planning (encompassing buses, ride sharing, and public bike systems) makes heavy use of spatiotemporal big data analytics in optimizing route

planning and resource deployments in order to maximize throughputs and minimize congestion delays.

1.3.14. Utilities

Geospatial big data analytics is used by utility companies to summarize and analyze customer usage patterns across a service area. They can assess customer usage through time and correlate usage to weather patterns, helping them anticipate future demand. Utilities can also use spatial big data analytics to analyze Supervisory Control and Data Acquisition (SCADA), smart meter, and other sensor data to detect and quantify potential problems in the distribution network, such as when and where outages occur, whether they correlate with weather events, and how many customers are affected. They can use this information to prioritize maintenance activities and prevent or mitigate future problems. Public utility commissions consume raw energy data from utilities and prepare future forecasts of energy consumption. Energy efficiency can also be studied to determine what the seasonal impacts are and what can be done to guide consumers toward smarter energy usage (Fig. 1.1).

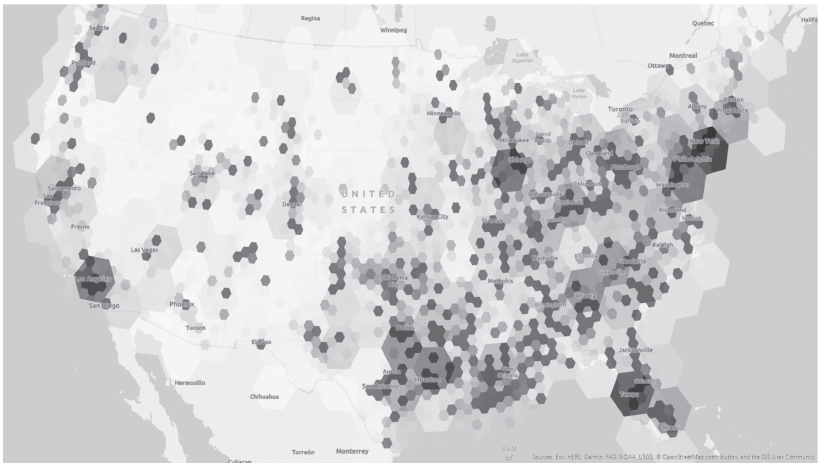

Figure 1.1 Leveraging feature binning technology to see geographic trends between industrial emission activity in 2014 (small hexes) as reported in the EPA Toxic Release Inventory and total U.S. electrical generation by load (large hexes) in 2018 as published by the Homeland Infrastructure Foundation-Level Data.

1.4. Big Data Analysis Concepts

The type of analysis that may be performed against spatial big data often parallels that which is typically done with traditional spatial data (Longley et al., 2015). However, when working with big data, it is oftentimes necessary to identify the key or most significant subsets of data in the larger collection. Once the interesting data are identified, further detailed analysis using the full breadth of spatiotemporal analysis tools and techniques can then be applied. This is particularly common when working with spatial big data that are obtained from sensors.

1.4.1. Summarizing Data

Summarizing data encompasses operations that calculate total counts, lengths, areas, and basic descriptive statistics of features and their attributes within areas or near other features (Fig. 1.2). Common operations that summarize data include the following.

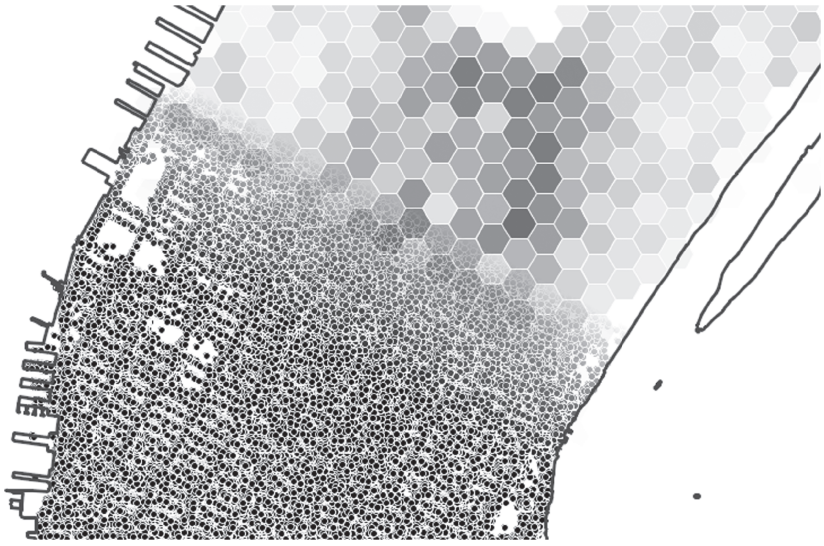

Figure 1.2 Ridesharing pick up locations in midtown Manhattan. In the southern portion of the figure, the raw data are shown. The northern region shows the data aggregated into 250 m height hexagon cells.

- *Aggregations* aggregate points into polygon features or bins. At all locations where points exist, a polygon is returned with a count of points as well as optional statistics.
- *Joins* matches two data sets based upon their spatial, temporal, or attribute relationships (Abel et al., 1995). *Spatial* joins match features based upon their spatial relationships (e.g., overlapping, intersecting, within distance, etc.); *temporal* joins match features based upon their temporal relationships; and *attribute* joins match features based upon their attribute values.
- *Track reconstruction* creates line tracks from temporally enabled, moving point features (e.g., positions of cars, aircraft, ships, or animals).
- *Summarization* overlays one data set on another and calculates summary statistics representing these relationships. For example, one set of polygons may be overlaid on another data set in order to summarize the number of polygons, their area, or attribute statistics.

1.4.2. Identify Locations

Location identification involves identifying areas that meet a number of different specified criteria. The criteria can be based on attribute queries (for example, parcels that are vacant) and spatial queries (for example, within 1 km of a river). The areas that are found can be selected from existing features (such as existing land parcels), or new features can be created where all the requirements are met. Common operations that are used to identify locations include (1) *incident detection*, which detects all features that meet a specified criteria (e.g., lightning strikes exceeding a given intensity), and (2) *similarity*, which identifies the features that are either the most similar or least similar to another set of features based upon attribution.

1.4.3. Pattern Analysis

Pattern analysis involves identifying, quantifying, and visualizing spatial patterns in spatial data (Bonham-Carter, 1994; Golledge & Stimson, 1997). Identifying geographic patterns is important for understanding how geographic phenomena behave.

Although it is possible to understand the overall pattern of features and their associated values through traditional mapping, calculating a statistic quantifies the pattern (Vapnik, 2000). Statistical quantification facilitates the comparison of patterns with different distributions or across different time periods. Pattern analysis tools are often used as a starting point for

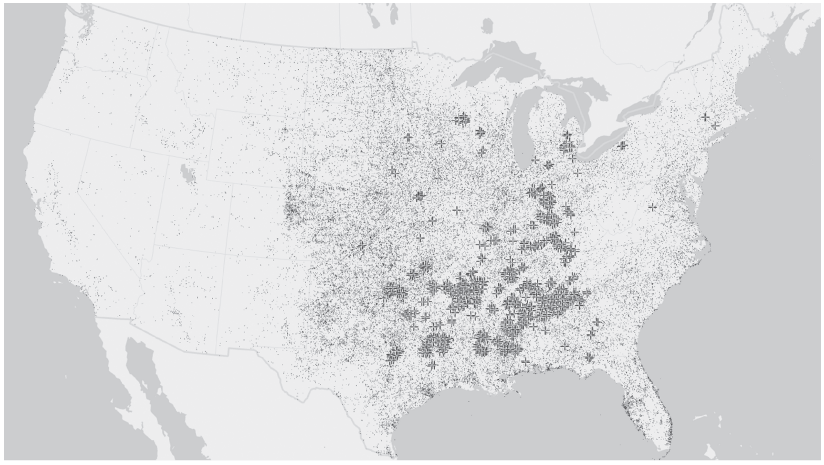

Figure 1.3 Tornado hotspots (+) and reported start points across the United States from 1950 to 2018. Hotspots are calculated using the Getis-Ord Gi* statistic on tornado geographic frequency and weighted by severity (Fugita Scale 0–5) to determine locations with a higher risk of damage based upon reported historical events (p-value < 0.05; z-score > 3). Tornado data from the NOAA Storm Prediction Center for Severe Weather.

more in-depth analyses. For example, spatial autocorrelation can be used to identify distances where the processes promoting spatial clustering are most pronounced. This might help the user to select an appropriate distance (scale of analysis) to use for investigating hot spots (hot spot analysis using the Getis-Ord Gi* statistic) (Fig. 1.3).

Pattern analysis tools are used for inferential statistics; they start with the null hypothesis that features, or the values associated with the features, exhibit a spatially random pattern. They then compute a p-value representing the probability that the null hypothesis is correct (that the observed pattern is simply one of many possible versions of complete spatial randomness). Calculating a probability may be important if you need to have a high level of confidence in decision making. If there are public safety or legal implications associated with your decision, for example, you may need to justify your decision using statistical evidence.

1.4.4. Cluster Analysis

Cluster analysis is used to identify the locations of statistically significant hot spots, spatial outliers, and similar features (Ester et al., 1996)

Figure 1.4 Spatiotemporal clustering (DBSCAN – Density-Based Spatial Clustering of Applications) of ridesharing drop-off locations in midtown Manhattan. This identified clusters (darker points in the figure) where many drop offs occurred in a similar place and time and the minimal cluster size is 15 events.

(Fig. 1.4). Cluster analysis is particularly useful when action is needed based on the location of one or more clusters. An example would be the assignment of additional police officers to deal with a cluster of burglaries. Pinpointing the location of spatial clusters is also important when looking for potential causes of clustering; where a disease outbreak occurs can often provide clues about what might be causing it. Unlike pattern analysis (which as used answer the questions such as, "Is there spatial clustering?") cluster analysis supports the visualization of the cluster locations and extent. Cluster analysis can be used to answer the questions such as, "Where are the clusters (hot spots and cold spots)?", "Where are incidents

most dense?", "Where are the spatial outliers?", and "Which features are most alike?"

1.4.5. Proximity Analysis

Proximity analysis allows people to answer one of the most common questions posed in spatial analysis: "What is near what?" This type of analysis supports the determination of proximal features within one or more data sets; for example, identify features that are closest to one another or calculate the distances between or around them. Common analysis methods include the following:

1. Distance calculation: The Euclidean distance from a single source or set of sources.
2. Travel cost calculation: The least accumulative cost distance from or to the least-cost source, while accounting for surface distance along with horizontal and vertical cost factors.
3. Optimal travel cost calculation: The optimum cost network from a set of input regions. One example application of this tool is finding the best network for emergency vehicles.

1.4.6. Predictive Modeling

Predictive analytics builds models to forecast behavior and other future developments. It encompasses techniques from spatial statistics, data mining, machine learning, and artificial intelligence (Minsky, 1986; Newell et al., 1959; Pedregosa et al., 2011). Patterns are identified in historical data and are used when creating models for future events.

Machine learning uses algorithms and statistical models to analyze large data sets without using explicit sequences of instructions. Machine learning algorithms create a model of training data that is used to make optimized predictions and decisions. Machine learning is considered to be a subset of artificial intelligence.

Deep learning is a subset of artificial intelligence where models resembling biological nervous systems are arrayed in multiple layers where each layer uses the output of the preceding as input to create a more abstract and composite representation of the data (LeCun et al., 2015). Deep learning architectures include deep neural networks, belief networks, and recurrent neural networks. Deep learning is commonly used in the domains of natural language processing, computer vision, and speech recognition.

1.5. Technology and Tools

There are several key technologies that are commonly employed to process large volumes of spatial data. These technologies are frequently distributed in nature, allowing collections of computing resources to work collaboratively toward solutions. This collection includes distributed processing frameworks and distributed data stores. At the most basic level, distributed systems are collections of networked computers that work in a coordinated manner; this is sometimes termed *concurrent computing*, *parallel computing*, or *distributed computing*. In a distributed computing environment, the processors run concurrently and each processor has its own private memory (distributed memory). Information is exchanged between processors through messages between the processors. In parallel computing, all processors in the cluster usually have access to shared memory that is used to exchange information between processors.

A distributed data store is a computer cluster where data are persisted on more than one node, often in a replicated fashion. Distributed databases are commonly nonrelational databases that are optimized to support rapid or parallel access of data across a large number of nodes. Distributed databases usually expose rich query capabilities; some however are limited to key-value store semantics. Examples of distributed databases are Google's Bigtable (Chang et al., 2008), Amazon's DynamoDB, and Microsoft's Azure Storage.

1.5.1. Available Tools

There are a number of tools and technologies that are used to support big data analytics. Some of the tools are open source, while others are more traditional commercial offerings. This collection encompasses distributed file systems, distributed processing frameworks, NoSQL and columnar data stores, as well as cloud-based computational platforms (Sena et al., 2017).

Distributed Processing Frameworks

Apache Hadoop is an open-source software framework that allows a cluster of commodity computers to solve problems involving large amounts of data and/or computation. Hadoop was motivated by work at Google on the MapReduce programming model and the Google File System (GFS) (Ghemawat et al., 2003). Hadoop supports a distributed storage and processing framework using the MapReduce programming model

(Sakr et al., 2013). Apache Spark is a more modern open-source, distributed processing framework that is optimized to support running large-scale data analytics applications across clustered systems; it differs from MapReduce as it better supports in-memory and memory pipelined applications.

Hadoop was designed for computer clusters built from commodity hardware; this follows the original Google GFS model. Hadoop is designed with the assumption that hardware failures are common and should be automatically handled by the framework. The core of Hadoop is the Hadoop Distributed File System (HDFS) (Shvachko et al., 2010), a resource manager, and a distributed processing framework that supports the MapReduce programming model. HDFS splits large files into shards (or blocks) that are distributed across multiple nodes in a cluster. Reliability is achieved through data replication, copying or replicating the blocks across multiple nodes in the cluster. Hadoop distributes software across the collection of nodes in the cluster to enable the processing of data in parallel. Taking the compute to the data allows large data sets to be processed faster and more efficiently than in systems that rely on parallel file systems where the compute and the data are distributed via high-speed communication architectures.

Hadoop can be deployed in on-premise datacenters as well as in the cloud; this allows organizations to deploy Hadoop without acquiring expensive hardware or having setup and operational (DevOps) expertise. Hadoop-compatible cloud offerings are available from Amazon, Microsoft, IBM, Google, and Oracle (among others).

Datastores

NoSQL databases (originally referencing "non-SQL" or "nonrelational") store and retrieve data differently from standard relational databases. NoSQL databases (sometimes considered next generation databases) are designed to address some of the limitations of traditional relational databases such as being distributable, simpler in design, often open-source, and horizontally scalable. Many databases supporting these characteristics originated in the late 1960s; the NoSQL description was employed beginning in the late 1990s with the requirements imposed by companies such as Facebook, Google, and Amazon. NoSQL databases are commonly used with big data applications. NoSQL systems are also sometimes called "Not only SQL" to emphasize that they may support SQL-like query languages.

In order to achieve increased performance and scalability, NoSQL databases commonly used data structures (e.g., key-value, columnar,

document, or graph) that are different from those used in relational databases. NoSQL databases vary in terms of applicability to particular problem domains. NoSQL databases are often classified by their primal data structures; examples include the following:

- Key-value: Apache Ignite, Couchbase, Amazon DynamoDB, Oracle NoSQL Database, Redis, Riak
- Columnar: Apache Accumulo, Apache Cassandra, Druid, Apache HBase, Vertica
- Document: Apache CouchDB, Azure Cosmos DB, IBM Domino, MarkLogic, MongoDB
- Graph: AllegroGraph, Apache Giraph, MarkLogic, Neo4j, Spark GraphX
- Multimodel: Apache Ignite, Couchbase, MarkLogic

Cloud Platforms

Big data analytic systems were often deployed on premises; however, cloud platform vendors have made it easier to deploy big data systems in the cloud. Cloud-based services enable organizations to create cloud-based clusters and run analytical processes as long as necessary. These clusters can then be taken offline when they are no longer needed (Chang et al., 2010). Cloud platforms commonly support scaling horizontally (also termed *scale-out*, e.g., adding more nodes to a system), as well as vertically (*scale-up*, e.g., adding resources to a node, such as CPU cores, memory, or storage).

Platform as a Service (PaaS) is a category of cloud computing services that provides a platform allowing organizations to run and manage distributed applications without the complexity of building and maintaining the infrastructure usually associated with developing and launching an application. PaaS is commonly delivered in one of three ways: (1) as a public cloud service from a provider, (2) as a private service (on-premises) inside the firewall, or (3) as software deployed on a public infrastructure as a service.

Big Data as a Service (BDaaS) is a new concept that combines Software as a Service (SaaS), Platform as a Service (PaaS), and Data as a Service (DaaS) to address the requirements of working with massively large data sets. BDaaS offerings commonly incorporate the Hadoop stack (e.g., HDFS, Hive, MapReduce, Pig, Storm, and Spark), NoSQL data stores, and stream processing capabilities.

Microsoft Azure is a cloud computing service utilizing Microsoft-managed data centers that supports both Software as a Service (SaaS) and Platform as a Service (PaaS). Visualization of the differences between

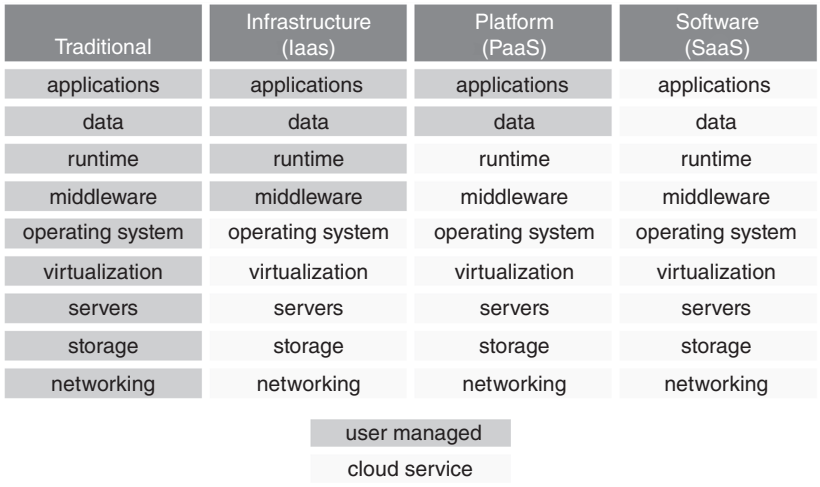

Figure 1.5 Cloud service models (IaaS, PaaS, and SaaS) (Chou, 2018).

the main cloud service models is provided in Figure 1.5. It provides data storage capabilities including Azure Cosmos DB (a NoSQL database), the Azure Data Lake, and SQL server-based databases. Azure supports a scalable event processing engine and a machine learning service that supports predictive analytics and data science applications.

The Google Cloud is a PaaS offering that supports big data with data warehousing, batch and stream processing, data exploration, and support for the Hadoop/Spark framework. Key components include BigQuery, a managed data warehouse supporting analytics at scale; Cloud Dataflow, which supports both stream and batch processing; and Cloud Dataproc, a framework for running Apache MapReduce and Spark processes.

Amazon AWS, though commonly considered an Infrastructure as a Service (IaaS) where the user is responsible for configuration, also provides PaaS functionality. Amazon supports Elastic MapReduce (EMR), which works in conjunction with EC2 (Elastic Compute Cloud) and S3 (Simple Storage Service). Data storage is provided through DynamoDB (NoSQL), Redshift (columnar), and RDS (Relational Data Store). Machine learning and real-time data processing infrastructures are also supported.

Other significant examples of BDaaS providers include the IBM Cloud and the Oracle Data Cloud. Big data Infrastructure as a Service (IaaS) offerings (that work with other clouds such as AWS, Azure, and Oracle) are also available from Cloudera and Databricks.

GIS: Hadoop-GIS, SpatialHadoop, Esri GeoAnalytics Server

In the academic and commercial realms, there are several systems of note. Hadoop-GIS (Aji et al., 2013) is an academic distributed spatial data warehousing and query processing system that utilizes the Hadoop along with the MapReduce programming model. Hadoop-GIS supports spatial partitioning and exposes a customizable spatial query engine (RESQUE) along with the ability to perform 2D and 3D spatial joins. Declarative queries are supported via an integration with Hive. A successor was implemented on Spark, called SparkGIS. SparkGIS also supports spatially aware management of partitions loaded into memory rather than arbitrary spilling to disk. It was benchmarked using medical pathology images as well as OpenStreetMap data.

SpatialHadoop is another academic research system (Eldawy & Mokbel, 2015); it is an open-source MapReduce extension to Hadoop that is focused on big spatial data. It has a custom spatial high-level language along with support for native spatial data types, spatial indexes (grid files, R-trees, and R+-trees), and spatial query operations (e.g., range queries, kNN (k-Nearest Neighbor), and spatial joins) on HDFS.

Other interesting academic research systems include GeoSpark, a framework for performing spatial joins, range queries, and kNN queries. GeoSpark supports both quadtree and R-tree based indexing of spatial data (Yu et al., 2013). GeoSpark uses a regular grid for global partitioning, with local spatial indexes. Simba is another system that provides range, distance, and kNN queries and joins. It uses two-level indexing and can support custom partitioning of data (Xie et al., 2016). Simba does not support spatiotemporal queries. Magellan is open-source software for spatial analytics based upon Spark (Sriharsha 2017). It supports Spark SQL for traditional SQL processing as well as a custom broadcast join. It uses the Java API provided by GIS Tools for Hadoop (Whitman et al., 2014). LocationSpark is another Spark-based library that supports range queries, spatial joining, and kNN queries (Tang et al., 2016). Spatial data are stored in key-value pairs with a geometry key. GeoMesa is a framework built upon Accumulo that provides geohash-based spatial indexing and query capabilities (Hughes et al., 2015). Finally, STARK (Spatio-Temporal Data Analytics on Spark) is another Spark-based framework that supports range queries, kNN queries, and range queries on both spatial and spatiotemporal data. STARK also supports density-based spatial clustering (DBSCAN) (Ester et al., 1996; Hagedon et al., 2017).

The Esri GeoAnalytics Server is a big spatial and temporal data processing and analysis capability of the ArcGIS Enterprise platform. It utilizes the Spark distributed-processing framework to support aggregation, regression, clustering, and analysis of big spatial data

(Whitman et al., 2019). It works with distributed file shares, HDFS, cloud storage, and Hive. It provides a large collection of tools that can be accessed through the ArcGIS desktop, the enterprise portal Map Viewer, a REST API, or via Python directly.

1.6. Challenges

The huge volume of spatial data coupled with its variety causes significant data management challenges with data quality, consistency, and governance (Hilbert, 2015). Building and maintaining the diverse collection of commercial and open-source big data processing tools and architectures (e.g., Apache Hadoop, HDFS, and Spark) in an accessible and cohesive architecture is a challenging proposition for most organizations. Other common problems when organizations initiate big data analytics initiatives include a lack of analytics skills among existing personnel coupled with the high cost of hiring new data scientists.

Recently, the proliferation and advancement of AI and machine learning technologies have enabled vendors to produce software for big data analysis that is easier to use, particularly for the growing citizen data scientist population. Some of the leading vendors in this field include Alteryx, IBM, and Microsoft.

The major challenges of spatial big data and analytics are less about the hardware and more about identifying individuals that are capable of working with and managing large volumes of data and being able analyze it and identify information that is valuable to their organizations.

The complexities related to the relationship between hardware, software, and expertise have evolved with time. The cost of hardware (CPUs and storage) was an original big data challenge. During the past decade, the cost per gigabyte for computer storage has dropped by a factor of five. Similar trends in processing power, memory, and communication infrastructures have also been observed.

Most organizations can afford big data processing hardware that will support the storage and analytic processing; smaller organizations can alternatively employ highly scalable cloud solutions that will support their spatial big data analytic requirements.

1.7. Summary

Big data analytics on spatial data (e.g., moving sensors, aerial and satellite imagery, Lidar, social networks, etc.) commonly involves spatial

processing, sophisticated spatial statistical algorithms, and predictive modeling. GIS users and data scientists apply big data analytics to evaluate these large collections of data, data with volumes that traditional analytical systems are unable to accommodate. Spatial big data are differentiated from standard big data by the presence of spatial relationships, geostatistical correlations, and spatial semantic relations; these additional challenges are beyond what is usually encountered with traditional big data. When one works with big data, one seeks common analytic objectives and workflows, including the visualization and identification of patterns and trends, filtering and converting streaming data containing geographical elements into geographic layers of information, cluster and proximity analysis, and predictive modeling.

To address these demands, new analytic environments and technologies have been developed; this includes distributed processing infrastructures (e.g., Hadoop and Spark), distributed file stores (e.g., HDFS), NoSQL databases (e.g., Accumulo, Cassandra, DynamoDB, HBase, and MongoDB), and big data–enabled cloud platforms (e.g., Azure, Amazon, Google, SAP, and Oracle).

The combination of recent developments in advanced analytical processing and sophisticated distributed processing technologies and infrastructures enables both modest and large organizations to take advantage of spatial big data and obtain new insights and understanding of their problem domains and communities.

References

Abel, D. J., Ooi, B. C., Tan, K. L., Power, R., & Yu, J. X. (1995). Spatial join strategies in distributed spatial DBMS. In M. J. Egenhofer & J. R. Herring (eds.), *Advances in spatial databases*. SSD 1995: Lecture Notes in Computer Science, vol. 951. Berlin, Heidelberg: Springer. https://doi.org/10.1007/3-540-60159-7_21

Aji, A., Wang, F., Vo, H., Lee, R., Liu, Q., Zhang, X., & Saltz, J. (2013). Hadoop-GIS: A high performance spatial data warehousing system over MapReduce. *Proceedings of the VLDB Endowment*, *6*(11), 1009–1020.

Alexander, W., & Copeland, G. (1988). Process and dataflow control in distributed data-intensive systems. *ACM SIGMOD Record*, 17(3), 90–98. https://doi.org/10.1145/50202.50212

Apache Hadoop (2018). Apache: Welcome to Apache Hadoop!, hadoop.apache.org

Barwick, H. (2011). *IIIS: The "four Vs" of big data*. Computerworld. At www.computerworld.com.au/article/396198/iiis_four_vs_big_data

Bonham-Carter, G. (1994). *Geographic information systems for geoscientists: Modeling with GIS*. New York: Pergamon.

Chang, F., Dean, J., Ghemawat, S., Hsieh, W., Wallach, D., Burrows, M., et al. (2008). Bigtable: A distributed storage system for structured data. *ACM Transactions on Computer Systems, 26*(2). https://doi.org/10.1145/50202.50212

Chang, W., Abu-Amara, H., & Sanford, J. (2010). *Transforming enterprise cloud services*. London: Springer.

Chou, D. (2018). *Cloud service models (IaaS, PaaS, SaaS) diagram*. https://dachou.github.io/2018/09/28/cloud-service-models.html

Dean, J., & Ghemawat, S. (2008). *MapReduce: Simplified data processing on large clusters, communications of the ACM*, 51(1), 107–113. https://doi.org/10.1145/1327452.1327492

DeWitt, D., & Gray, J. (1992). Parallel database systems: The future of high performance database systems. *Communications of the ACM, 35* (6). https://doi.org/10.1145/129888.129894

Eldawy, A., & Mokbel, M. (2015). SpatialHadoop: A MapReduce framework for spatial data. *Proceedings of the IEEE 31st International Conference on Data Engineering (ICDE)*, 1352–1363.

Ester, M., Kriegel, H., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD 1996)*, 226–231.

Garillot, F., & Maas, G. (2018). *Stream processing with Apache Spark: Best practices for scaling and optimizing Apache Spark*. O'Reilly Media.

Ghemawat, S., Gobioff, H., & Leung, S. (2003): The Google file system. *Proceedings of the 19th ACM Symposium on Operating Systems Principles (2003)*. 29–43. https://doi.org/10.1145/945445.945450

Golledge, R., & Stimson, R. (1997): *Spatial behavior: A geographic perspective*. New York: Guilford Press.

Hagedorn, S., Guotze, P., & Sattler, K.(2017). The STARK framework for spatio-temporal data analytics on Spark. *Proceedings of the 17th Conference on Database Systems for Business, Technology, and the Web (BTW 2017)* Stuttgart, *Germany.*

Hägerstrand, T. (1970). What about people in regional science? *Papers of the Regional Science Association*, *24* (1), 7–24. https://doi.org/10.1111/j.1435-5597.1970.tb01464.x

Hey, A., Tansley, S., & Tolle, K. (2009). *The fourth paradigm: Data-intensive scientific discovery*. Microsoft Research.

Hilbert, M. (2015). Big data for development: A review of promises and challenges. *Development Policy Review*, *34* (1), 135–174.

Hughes, J., Annex, A., Eichelberger, C., Fox, A., Hulbert, A., & Ronquest, M. (2015). Geomesa: A distributed architecture for spatio-temporal fusion. Proceedings of SPIE Defense and Security *(2015)*. https://doi.org/10.1117/12.2177233

Klein, J., Buglak, R., Blockow, D., Wuttke, T., & Cooper, B. (2016). A reference architecture for big data systems in the national security domain. *Proceedings of the 2nd International Workshop on BIG Data Software Engineering (BIGDSE 2016)*. https://doi.org/10.1145/2896825.2896834

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521* (7553), 436–444. https://doi.org/10.1038/nature14539

Longley, P., Goodchild, M., Maguire, D., & Rhind, D. (2015). *Geographic information systems and science, 2nd. ed*. Hoboken, NJ: Wiley.

Marz, N., & Warren, J. (2013). *Big data: Principles and best practices of scalable realtime data systems*. Greenwich, CT: Manning Publications.

Miller, H., & Goodchild, M. (2014). Data-driven geography. *GeoJournal*, *80*(4), 449–461. https://doi.org/10.1007/s10708-014-9602-6

Minsky, M. (1986). *The society of mind*. New York: Simon & Schuster.

Mysore, D., Khupat, S., & Jain, S. (2013). *Big data architecture and patterns*. IBM White Paper 2013. www.ibm.com/developerworks/library/bdarchpatterns1

Newell, A., Shaw, J., & Simon, H. (1959). Report on a general problem-solving program. *Communications of the ACM*, *2* (7), 256–264.

NoSQL (2022). *NoSQL definition*. *At* www.nosql-database.org.

Pavlo, A., & Aslett, M. (2016). What's really new with NewSQL? *SIGMOD Record*, *45*(2), 45–55. https://doi.org/10.1145/3003665.3003674

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Machine learning in Python. *Journal of Machine Learning Research*, 2825–2830.

Sakr, S., Liu, A., & Fayoumi, A. (2013). The family of MapReduce and large-scale data processing systems. *ACM Computing Surveys, 46* (1). https://doi.org/10.1145/2522968.2522979

Sena, B., Allian, A., & Nakagawa, E. (2017). Characterizing big data software architectures: A systematic mapping study. *Proceedings of the 11th Brazilian Symposium on Software Components, Architectures, and Reuse (SBCARS 2017)*. https://doi.org/10.1145/3132498.3132510

Shvachko, K., Kuang, H., Radia, S., & Chansler, R. (2010). The Hadoop distributed file system. *Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST 2010)*. https://doi.org/10.1109/MSST.2010.5496972

Sriharsha, R. (2017). Magellan: Geospatial analytics on Spark. www.hortonworks.com/blog/magellan-geospatial-analytics-in-spark

Tang, M., Yu, Y., Malluhi, Q., Ouzzani, M., & Aref, W. (2016). LocationSpark: A distributed in-memory data management system for big spatial data. *Proceedings of the VLDB Endowment*, *9* (13), 1565–1568. https://doi.org/10.14778/3007263.3007310

Vapnik, V. (2000). *The nature of statistical learning theory*. Berlin: Springer. https://doi.org/10.1007/978-1-4757-3264-1

Whitman, R., Park, M., Ambrose, S., & Hoel, E. (2014). Spatial indexing and analytics on Hadoop. *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL 2014)*, 73–82. https://doi.org/10.1145/2666310.2666387

Whitman, R., Park, M., Marsh, B., & Hoel, E. (2019). Distributed spatial and spatiotemporal join on Apache Spark. *ACM Transactions on Spatial Algorithms and Systems (TSAS), 5* (1). https://doi.org/10.1145/3325135

Xie, D., Li, F., Yao, B., Li, G., Zhou, L., & Guo, M. (2016). Simba: Efficient in-memory spatial analytics. *Proceedings of the 2016 International Conference on Management of Data (SIGMOD 2016)*, 1071–1085. https://doi.org/10.1145/2882903.2915237

Yu, J., Wu, J., & Sarwat, M. (2013). Geospark: A cluster computing framework for processing large-scale spatial data. *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL 2013)*. https://doi.org/10.1145/2820783.2820860

Zaharia, M., Chowdhury, M., Franklin, M., Shenker, S., & Stoica, I. (2010). Spark: Cluster computing with working sets. *Proceedings of the 2nd USENIX conference on hot topics in cloud computing (HotCloud 2010)*.