Symphony AYASDI

TDA and Machine Learning: Better Together

WHITEPAPER

Table of Contents

Overview

Ayasdi's award-winning artificial intelligence platform powers the design, development and deployment of enterprise-scale, intelligent applications. Our approach, our underlying technology, and our products are expressly crafted to deliver against an enterprise's requirements in this area – with the goal of delivering extraordinary business value. This paper details our underlying technology, Topological Data Analysis (TDA), and how it interacts with and enhances other machine learning technologies from unsupervised approaches through supervised approaches.

As a technology, TDA is important due to its ability to distill business value from large, complex datasets. Global enterprises increasingly look to their data to make decisions that can affect millions of lives and billions of dollars of revenue. Embedded in the challenge is an explosion of data – log, sensor, social, health and financial. The quantity of possible insights in a given dataset is an exponential function of the number of data points. On top of this, aggregate data growth is an exponential function with time. Unfortunately, we cannot train enough data scientists to meet this runaway, double-exponential demand curve. This is driving scientists and mathematicians to examine new approaches, such as TDA to improve both the quality and the speed of their analytics engines. High-performance machines and algorithms can examine complex data far faster and seek insights more comprehensively than ever before. However, we need to find exponential improvements in analysis techniques to meet the growing demand of exploding data volumes. Topological Data Analysis is one such technique.

Introducing Topology, Topological Data Analysis, and the Ayasdi Machine Intelligence Platform

Topology is a mathematical discipline that studies shape. Topological Data Analysis (TDA) refers to the adaptation of this discipline to analyzing highly complex data. It draws on the philosophy that all data has an underlying shape and that shape has meaning. Ayasdi's approach to TDA is embodied in the Ayasdi Machine Intelligence Platform. This Platform draws together a broad range of machine learning, statistical, and geometric algorithms to create a summary or compressed representation of all the data points in a large data set and thus to rapidly uncover critical patterns and relationships in that data set. By identifying the geometric relationships that exist between data points, TDA offers an extremely simple and efficient way of partitioning data to understand the underlying properties that characterize the segments and sub-segments that lie within that data.

Ayasdi's Machine Intelligence Platform is the only commercially available implementation of TDA. The marriage of TDA and machine learning in the Ayasdi Machine Intelligence Platform provides customers with an infrastructure that greatly enhances their ability to process and draw meaning from data that is highly resistant to other methods of analysis and interpretation.

The Platform supports a suite of application technologies designed to facilitate the design, development, and deployment of intelligent applications. These applications serve specific vertical markets such as healthcare, financial services, and the public sector.

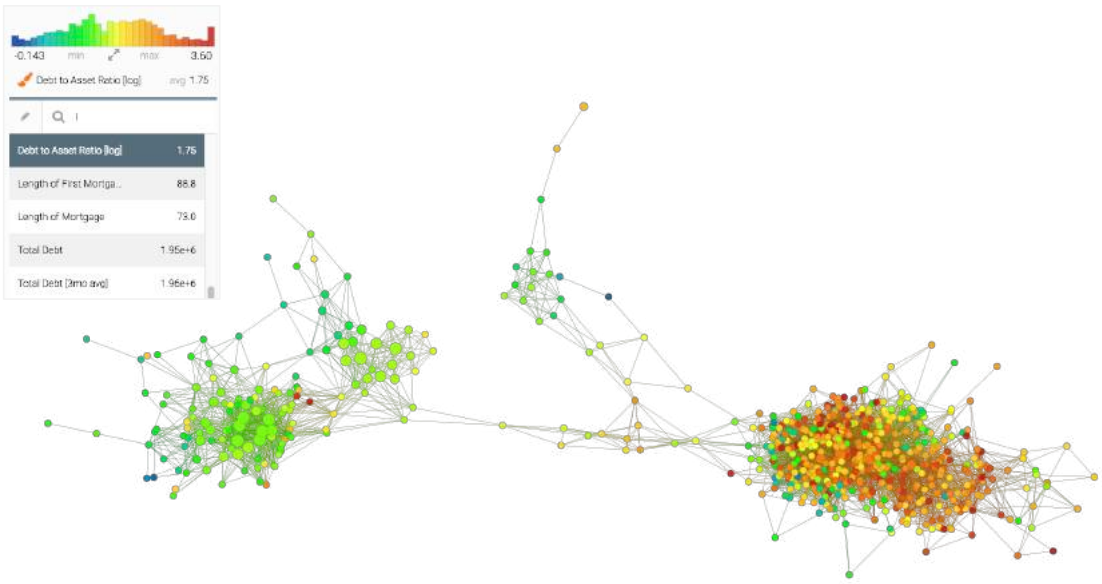


Figure 1: Creating a Compressed Representation of Data to Uncover Patterns and Subgroups of Interest

The Promise of Machine Learning

Machine learning is a class of algorithms that adjust and learn from data that then take or suggest actions. This field helps companies:

1. Effectively segment existing data into meaningful groupings.
2. Identify the key attributes and features that drive segmentation.
3. Find patterns and anomalies in a data set.
4. Precisely classify new data points as they arrive.

There are two classes of machine learning techniques – unsupervised and supervised.

Unsupervised learning helps discover the hidden structure in data; it utilizes only the content of input data and has no knowledge of the expected output.

Supervised learning constructs predictive models by using a training data set to create a function that can accurately infer outputs when presented with new input data.

Innovations in machine learning techniques promise to help drive new revenue streams, forge stronger customer relationships, predict risk, improve medical outcomes, and prevent fraud. However, analyzing complex data using these methods is constrained by certain intrinsic issues as well as a dependency on scarce machine learning expertise.

Machine Learning – Mind the Gap

Unsupervised Learning – An Overview

Unsupervised learning algorithms, which have no knowledge of expected results, fall into the following two categories:

1. Clustering – These algorithms discover the underlying sub-segments within data by grouping sets of data points in such a way that those in the same group (called a cluster) are more similar to each other than to those in other clusters.
2. Dimensionality Reduction - These algorithms are especially useful for reducing the number of properties or attributes (represented by data columns) required for describing each data point while retaining the inherent structure of the data.

Unsupervised Learning – Clustering

Clustering methods divide or segment a dataset into smaller datasets. Different clustering algorithms rely on different techniques to cluster data. Take the example of a hierarchical clustering algorithm such as single-linkage clustering. With single-linkage clustering each data point starts out as its own cluster. The single-point clusters are combined into larger clusters of points by sequentially fusing data point pairs that are the most similar to each other by some measure. The process continues until all data points have been fused into a cluster. The resulting cluster hierarchy can be visualized as a dendrogram (tree diagram) that shows which clusters were fused together to produce new clusters. Knowing the sequence and distance at which cluster fusion took place can help determine the optimal scale for clustering.

In general, there are three key issues with clustering algorithms:

1. The Number of Clusters – Some clustering algorithms require that the number of clusters be determined in advance. While a machine learning expert might use some informed criteria (such as a "Bayesian information score") to make an educated guess at the number of clusters, typically this is an arbitrary choice that can greatly impact conclusions drawn from the data.
2. Continuous Data Sets and Multi-modality - Clustering methods work well when data sets decompose cleanly into distinct groups that are well separated. However, many data sets are continuous and exhibit progressions rather than sharp divisions. Clustering methods can create spurious divisions in such data sets, thereby obscuring the true underlying structure of the data. Moreover, most algorithms implicitly assume that each cluster that we are looking for is equally dense. This is rarely the case in real datasets.
3. Shape Assumptions – Many algorithms have implicit or explicit constraints on the shape of the clusters. As an example, the k-means family of algorithms implicitly assumes clusters to be 'spherical'. Similarly, model based algorithms explicitly assume a model of the shape of the cluster. Imagine you find clusters using either of these methods – how might you discover the fact that the clusters are not faithful to the underlying distribution?

Unsupervised Learning – Dimensionality Reduction

Dimensionality Reduction methods make it easier to visualize data sets that have a large number of data columns. For example, consider credit card transactions that have thousands of attributes that are each represented as a data column. Visualizing these transactions can be extremely difficult given that we cannot see more than three dimensions at a time. Principal Component Analysis (PCA) is a good example of a Dimensionality Reduction algorithm. Other Dimensionality Reduction methods include Multi-dimensional Scaling, Isomap, t-Distributed Stochastic Neighbor Embedding, UMAP embedding, and Google's PageRank algorithm.

Dimensionality Reduction methods are extremely powerful since they can reduce the number of dimensions required to describe data while still revealing some inherent structure in that data.

However, there are two issues with Dimensionality Reduction methods:

1. Projection Loss – Dimensionality Reduction methods compress a large number of attributes down to a few. As a result, data points that are well separated in the high-dimensional space might appear as neighbors in the projection. Distinct clusters might overlap. This increases the chances of missing out on important insights.
2. Inconsistent Results - Distinct Dimensionality Reduction algorithms produce dissimilar projections because they encode different assumptions. None of the results are wrong; they are simply unlike one another because different algorithms accentuate different aspects of the data. Relying on a single algorithm might result in missed critical insights.

Supervised Learning - Regression and Classification

Supervised learning algorithms are used for producing predictive models. There are two types of supervised learning algorithms:

1. Regression algorithms.
2. Classification algorithms.

Regressors predict real-valued variables (e.g., profit margins, stock prices). Classifiers predict discrete variables (such as fraud or customer churn). Examples include Linear and Logistic Regression, Support Vector Machine, and Artificial Neural Networks.

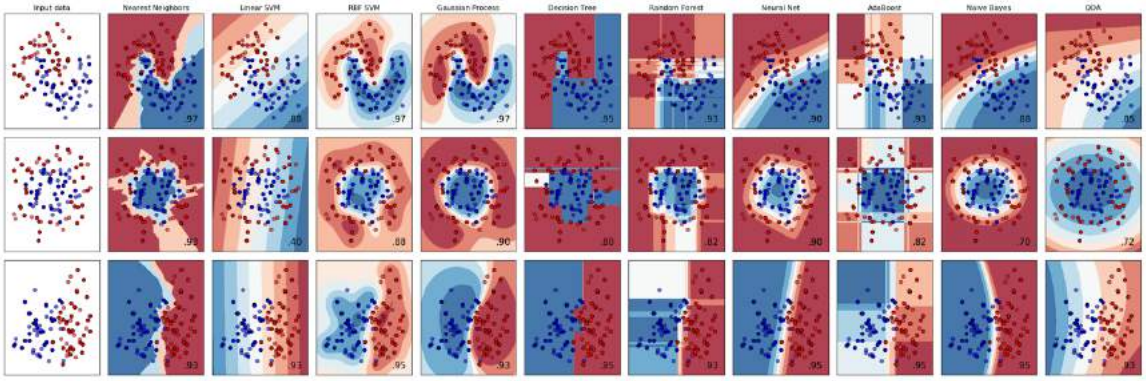
There are two phases in supervised learning:

1. Training - In this phase, the algorithm analyzes a training data set to produce parameters for a function that will infer results when presented with new input data. This phase needs historical data, i.e. empirical evidence known to be correct ("ground truth"), against which to verify the accuracy of the function. By definition, the training set contains associated ground truth information that is used to modify or tweak the function as needed to improve its ability to accurately infer data values.
2. Prediction - In this phase, the function that was produced in the training phase is used to predict the values for new input data points.

Supervised learning algorithms have certain inherent issues that need to be taken into consideration:

1. Assumptions - The choice of algorithm entails an assumption about the shape of the underlying data. For example, linear regression assumes that the data is planar (though possibly higher dimensional) and tries to find the best plane that fits the data. If the actual shape of the underlying data is not planar, then the analysis will produce incorrect results.

   The chart below shows decision boundaries of standard machine learning classifiers. Notice that each algorithm has a distinctive shape to its decision boundary. Without first knowing the shape of the underlying data, the chosen algorithm might produce poor results. These approaches rely heavily on a machine learning expert knowing which algorithm to choose.



https://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html

2. Global Optimization - All supervised learning algorithms try to find parameters for a function that best approximate all of the data. However, data is rarely homogeneous. In other words, it is unlikely that there is a single shape that fits the entire dataset.
3. Generalization - A model may perform well with test data, but produce inaccurate results with new data. This is known as a generalization error and it occurs because the model has more parameters than are actually required. This issue is also known as overfitting.

Supervised Learning - What is Missing

Despite its promise, supervised learning has the following limitations:

1. Successful implementations require experts in machine learning, an increasingly scarce resource to find.
2. It is easy to miss important insights in data by choosing the wrong algorithm or by not trying enough algorithms or hyperparameters.
3. Each class of machine learning algorithms has its own set of intrinsic issues that need to be taken into account.
4. Most supervised machine learning algorithms require large quantities of high quality labeled data that can be difficult, expensive, or impossible to attain.

The next section details how TDA enhances standard machine learning methods.

How TDA Improves the Application of Machine Learning Algorithms

All machine learning methods produce functions or maps. For example:

1. Clustering maps each input data point to a cluster.
2. Dimensionality Reduction maps each input data point to a lower dimensional data point.
3. Supervised Learning algorithms map each input data point to a predicted value.

The Ayasdi Machine Intelligence Platform dramatically increases the effectiveness of machine learning by using all of these maps or functions simultaneously to process input and thereby produce superior quality output. In addition, not-yet-invented machine learning algorithms can easily be added to the Machine Intelligence Platform infrastructure to further enhance its ability to streamline analysis of large and/or complex data sets.
Let us explore how TDA and each of the previously mentioned machine learning approaches interact.

Unsupervised Learning - Clustering

TDA uses clustering as an integral step in building a network representation of data. As opposed to trying to find disjoint groups, TDA applies clustering to small portions of data. It then combines these "partial clusters" into a network representation that gives an overview of the similarity between the data points. This makes TDA more appropriate for constructing a connected representation of either continuous data sets or data with heterogeneous densities.

Most all clustering algorithms rely on global optimization techniques (including HDBSCAN), which are susceptible to the noise since they consider all of the data during the optimization. By comparison, TDA splits data into multiple independent parts using lens functions and runs clustering algorithms within each part of the data independently. The multiple local optimizations executed by TDA dramatically reduce the effect of noise on the final results.

Unsupervised Learning - Dimensionality Reduction

TDA supports the automatic execution and synthesis of Dimensionality Reduction algorithms. The key benefits of this approach include the following:

1. Elimination of the projection loss issue typical of Dimensionality Reduction methods wherein data points that were well separated in higher dimensions end up overlapping in a lower dimensional projection. This is achieved by clustering the data in the original high dimensional space. As a result, data points that were well separated in the original space will typically still be well separated in the output. This enables the easy identification of distinct segments and sub-segments within data that might have been missed using other Dimensionality Reduction methods.

2. Automatic synthesis of the results of different Dimensionality Reduction algorithms into a single output. This eliminates the need to know or guess the correct sets of assumptions for a Dimensionality Reduction method.

Supervised Learning - Regression and Classification

TDA augments supervised learning algorithms in the following ways:

1. Eliminates systematic errors - Most supervised learning algorithms are based on global optimization. As such they assume a shape for the underlying data and then try to discover the parameters that best approximate all the data. The result can be mistakes in some data regions. TDA uses the output of these supervised algorithms as an input to discover areas of the underlying data where errors are being made systematically.
2. Optimized for local data sets - TDA effectively constructs a collection or ensemble of models rather than making global assumptions regarding all the underlying data. Each model is responsible for a different segment of the data. This eliminates the need to create a single model that works well on all of the data, which is a difficult if not impossible mission. This "collection of models" method generates more accurate results and can incorporate any supervised algorithm.

Additionally, TDA utilized on the feature space (that is the transpose of the original data) can be used to substantially improve the convergence capabilities of advanced machine learning techniques such as neural networks. More details on this can be found on the Ayasdi blog.

Summary

TDA reduces the possibility of missing critical insights by reducing the dependency on machine learning experts choosing the right algorithms. It uses current machine learning techniques as input to find subtle patterns and insights in local data. In general, Ayasdi's approach enhances any algorithm with which it is paired.

Creating Topological Networks with TDA by Using Ayasdi's Machine Intelligence Platform

TDA as a discipline identifies data points that are related to each other. It then pieces these regions of data together to build a global, compressed summary of the data in the form of a network. This network can be executed programmatically, or it can be visualized for further investigation.

In the case of a visual network, the Ayasdi Machine Intelligence Platform consistently applies a function (call it *f*) to the data while using a measure of similarity to generate a compressed representation of the data. The resulting visual network consists of nodes wherein each node represents data points having similar function values that have been clustered together based on a measure of that similarity.

Consider two simple examples to illustrate how the Ayasdi Machine Intelligence Platform creates networks. The first example steps through the general methodology and the second example demonstrates how the Ayasdi Platform enhances machine learning.

For the first example, consider a data set that is represented by a circle in the *xy*-plane as depicted in Figure 2. Apply a function *f* that maps each point in the data set to its *y*-coordinate value, as shown on the right side of Figure 2.

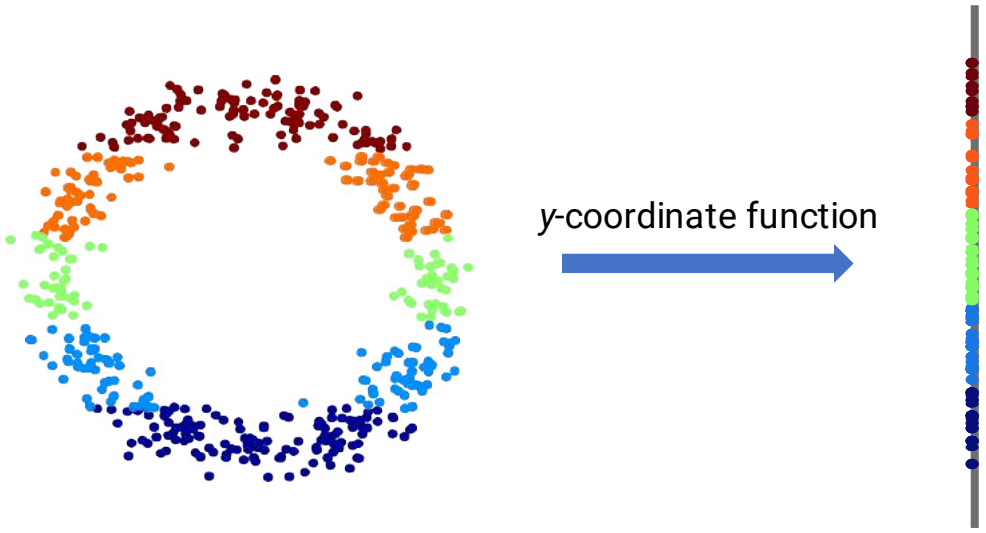


Figure 2: Using a Function to Map Data Points in the Shape of a Circle to their y-Coordinate Values

The Ayasdi Machine Intelligence Platform then subdivides the image of the function into overlapping sets of nearby values. In this example, the points are divided into four overlapping groups that have similar y-coordinate values as illustrated in Figure 3.

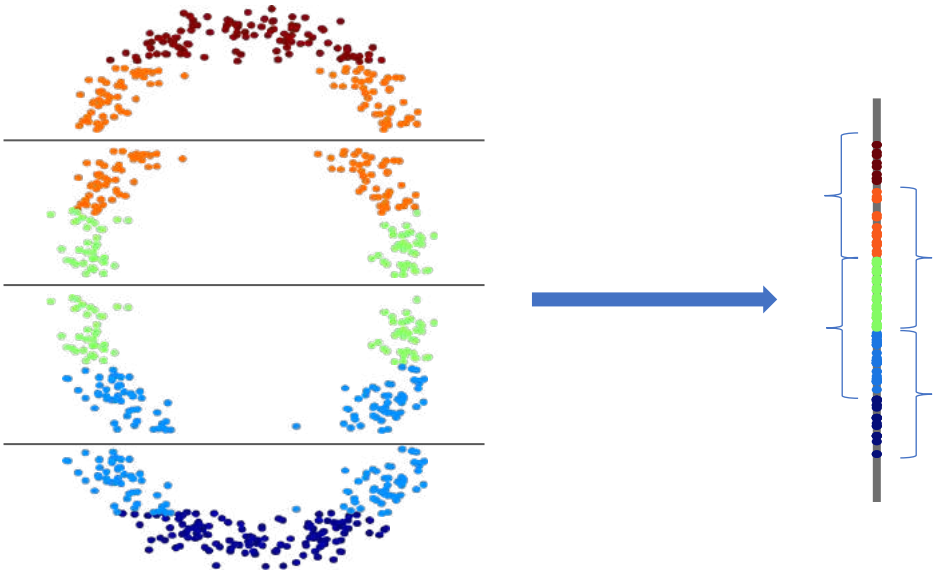


Figure 3: Dividing Data Points into Overlapping Sets with Similar y-Coordinate Values

Next, the Ayasdi Platform clusters each group of data points independently using a measure of similarity. In this example, similarity is defined using the standard Euclidean (straight line) distance. Each cluster is represented as a node. A node represents a set of data points that have a measure of similarity (Euclidean distance) and the function value (y-coordinate) in common. The size of each node reflects the number of data points within. Notice that as shown in Figure 4 the top node represents both red and orange data points while the second set of data points from the top in the original circular pattern contains two distinct regions of data points that produce two separate nodes in the topological image at right in Figure 4.

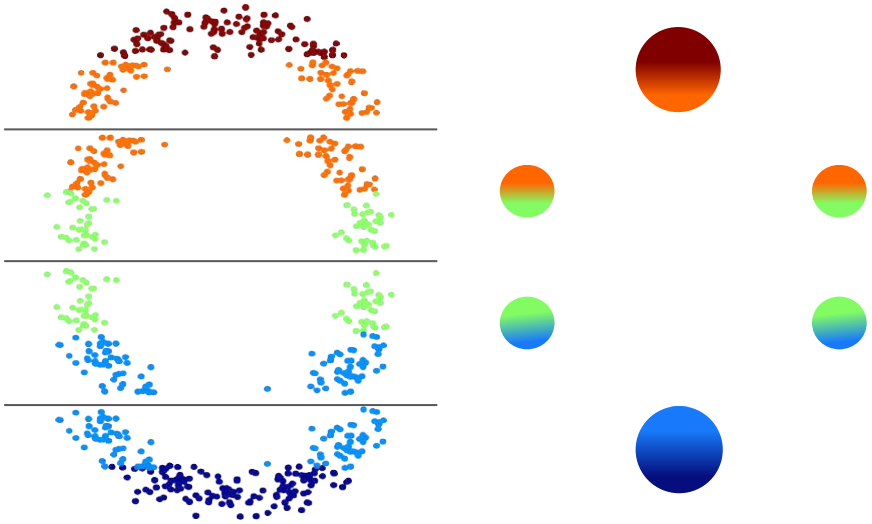


Figure 4: Nodes Represent Clusters of Data Points with Similar Function Values and Measures of Similarity

Nodes with data points in common are connected by edges in the Ayasdi-generated network. Since the data set was divided into overlapping sets, a data point can be represented in multiple nodes. In this example, as can be seen in Figure 5, the orange data points on the left are represented in both the top red node as well as in the orange node on the left. These nodes are connected by an edge because they contain data points in common.

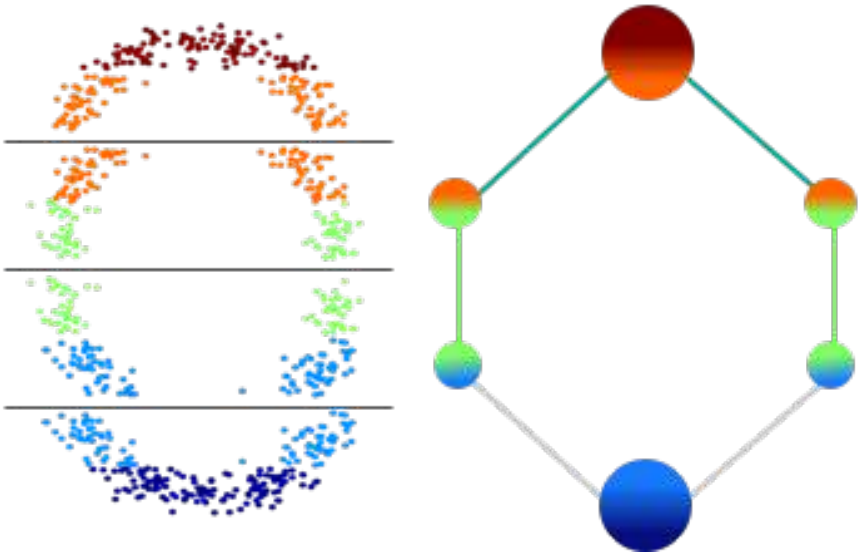


Figure 5: Nodes with Data Points in Common are Connected by Edges to Form a Network

- The resulting network is a compressed representation of the original data set that retains its fundamental circular shape. The network is much simpler to visualize and work with than the original data, yet it captures the essential behavior of the data.

Let's turn now to a second example, in which a data set is sampled in the two-dimensional Euclidean plane from four Gaussian distributions. In Figure 6, we color the data points by the values of the density estimator function. The Ayasdi Machine Intelligence Platform then divides the data set into overlapping groups with similar function values (in this case, density estimations). Each subset of the data is clustered to create nodes that represent data points with similar function values.

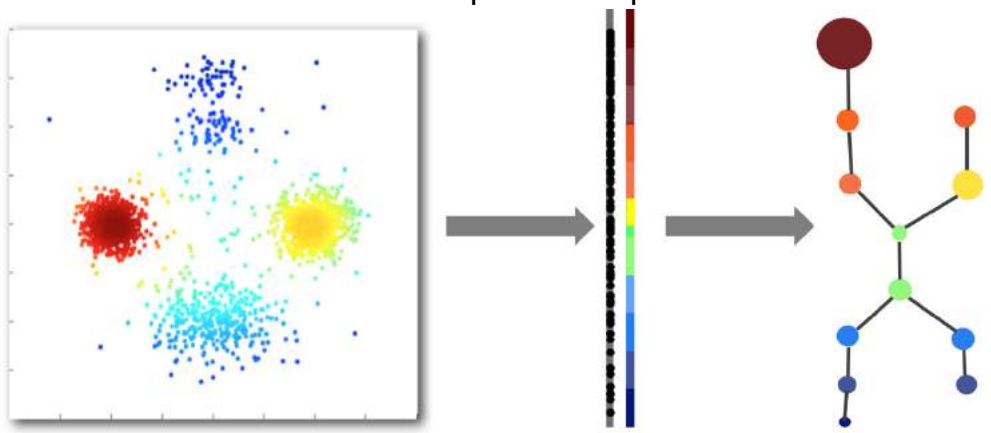


Figure 6: TDA Enhances Machine Learning by Capturing the Overall Structure and Fine-Grained Behavior in a Data Set

The resulting network depicted in Figure 6 captures both the overall structure of the data as well as its fine-grained behavior. The four flares in the network correspond to the four regions of varying densities. The flares in the network connect to each other because these contain common data points having varying degrees of density.

Standard machine learning techniques would have identified the four regions but they would have lost the continuous transitions between them. The Ayasdi Machine Intelligence Platform captures both the differences and the similarities in the data.

Complex data holds useful information that could go undetected when using standard machine learning and statistical techniques. By contrast the Ayasdi Machine Intelligence Platform begins by understanding data at a small scale. It then stitches together these pieces of information to create a topological summary or compressed representation of the entire data set. This is another instance of the Platform's ability to create networks that display subtle insights in the data while also showing the global behavior of the data.

The Ayasdi Machine Learning Platform can incorporate virtually any machine learning, statistical, or geometric technique (or combination thereof) and apply these techniques as a function *f* to display as a visual network a compressed facsimile of a data set. Principal component analysis, autoencoders,

random forests, and density estimators are just some examples of functions that the Ayasdi Platform can use to derive insights from large, complex data sets.

Exploring and Using the Ayasdi Machine Intelligence Platform to Understand your Data

As discussed above, the Ayasdi Machine Intelligence Platform uses TDA to create a visual representation of data in the form of a topological network. A network is comprised of the following:

- Nodes that represent collections of similar data points.
- Edges that connect nodes that share data points in common.

TDA helps automatically discover these networks that reveal the underlying structure of a data set. The networks produced by TDA are simple, yet extremely powerful representations of the data and the information encoded in them feed discrete business applications – including fraud detection, money laundering, clinical variation, credit risk or program performance.

How the Ayasdi Machine Intelligence Platform Uses TDA to Make Complex Data More Intelligible

The Ayasdi Machine Intelligence Platform has a number of features that support discovery of otherwise hidden information in data sets. These include segmentation, feature discovery, classification, model creation, model validation, and anomaly detection. Each is discussed below.

Segmentation

As stated above, data segmentation involves grouping data points that are more similar to each other by some metric than to the remainder of the data. The most common segmentation approaches involve either a data scientist manually generating and testing hypotheses or the use of clustering algorithms. Manual testing of hypotheses can be a huge undertaking even when dealing with small data sets. This approach typically consists of a domain expert choosing a logical attribute of the data in order to create segments. This approach may have limited utility because key information is not taken into account. Consider, for example, segmenting customers by the amount of money spent; this might seem like a good idea but it ignores the impact of key factors such as demographics.

By comparison, standard clustering methods for segmentation produce better results. However, these methods still suffer from the limitations described earlier including: 1. A need to know the number of clusters in advance of applying the algorithm; 2. The unsuitability for tackling continuous data sets; and 3. The assumption of spherical shape or uniform densities.

As a more detailed example of clustering, consider a financial institution that segments its clients by their investment behavior under specific market conditions and then precisely targets them at the right

time with tailored recommendations. Such an approach relies on macro-trends to explain client-buying behavior, yet it can miss subtle trends hidden in the data that are tied, for example, to specific regional events. Such an event might result in a particular group of clients trading in a specific class of products that diverges from the general trend and is never seen because of the segmentation approach applied. The invisibility of such subtle trends is more likely if the number of clients exhibiting a particular behavior is small compared to the total number of clients in the data set. Complex human behavior such as financial markets typically has numerous instances of overlapping trends both large and small; teasing out the "small" is challenging with traditional methodologies.

The Ayasdi Machine Intelligence Platform would, on the other hand, discover that while these regional investors are similar to the majority of clients in the data, they are more similar to each other than to the majority. This subtle signal would be captured in the Ayasdi Machine Intelligence Platform output as a flare in the visual network and its presence encoded in an application designed to alert the bank's sales force of this unique subpopulation. Moreover, the application would direct the sales force to prioritize and target these highly responsive clients with tailored recommendations ahead of those that are more similar to the majority of the clients in the data set.

Anomaly Detection

The Model Validation description above relies on the availability of predicted outcome and ground truth information for a data set. There are, however, cases in which this information is not readily available. The Ayasdi Platform offers an alternative approach, anomaly detection, that does not require the existence of predicted outcome and ground truth information. The workflow within the Ayasdi Platform for anomaly detection is as follows:

1. Construct a data set of transactions. Ground truth or other information from current models is not required.
2. Segment the data set based on all data columns to generate a topological network.
3. Explore onscreen regions of the network that represent low density points or points far away from the central core of the data set. This "anomalous" data has less in common with data points in network nodes, which is the reason it was not displayed as part of a node.

Feature Discovery

Understanding the underlying features or attributes of the data that drive segmentation can be invaluable when pinpointing the factors that impact business outcomes. Ayasdi's software helps with feature discovery by automatically producing a list of the attributes (data columns) that drive segmentation, ranked in order of statistical significance.

Take the example of using the Ayasdi Platform to understand the reasons for customer churn. While the ability to predict churn is useful, spotting the root causes for churn is significantly more important as it often brings systemic issues to the surface.

Identifying the attributes of departing customers with the Ayasdi Platform involves the following steps:

1. Construct a data set with columns – each column is a customer attribute of interest. Optionally create an outcome data column by which data can be segmented. In this example, an output data column tracks whether a customer departed or remained.
2. Segment the data set using all data columns. For the present example, the output data column that tracks customer churn serves as an additional data lens through which data is viewed.
3. Create clusters of data points or node groups that form a visual network by using the outcome data column for tracking churn.
4. Use the "Explain" operation in the Ayasdi Platform to get a tabular listing of the underlying features or attributes of the node groups that represent departing customers. This listing will be arranged in statistical order of importance.
5. Encode this information into the workflow of the customer care teams responsible for intercepting potential churners.

Recommendation Engines

Recommendation engines are designed to help organizations generate more revenue by precisely targeting customers with sales efforts for products and services purchased in the past by other customers with similar profiles. The Ayasdi Machine Intelligence Platform serves as an ideal foundation for a recommendation engine. The steps to build a recommendation application with the Ayasdi Platform is as follows:

1. Create precise sub-segments of a customer base by correlating and analyzing a wide range of client-related data including demographics, buying behavior, market, CRM, and social media information. Figure 7 presents an example of using the Ayasdi Platform to distill this information into a topology diagram that can be subjected to further analysis.
3. Assign all newly arriving customer data points to a specific node or group of nodes (sub-segments) in the topology diagram created in the previous step.
4. Look up the buying behavior of the other customers that are represented in the sub-segments to which the new customers were assigned.
5. Develop tailored recommendations based on what similar customers (those in the same sub-segments identified in the previous step) have bought in the past.
6. Use the information encoded in the network to feed the existing CRM application or to build a client facing application.

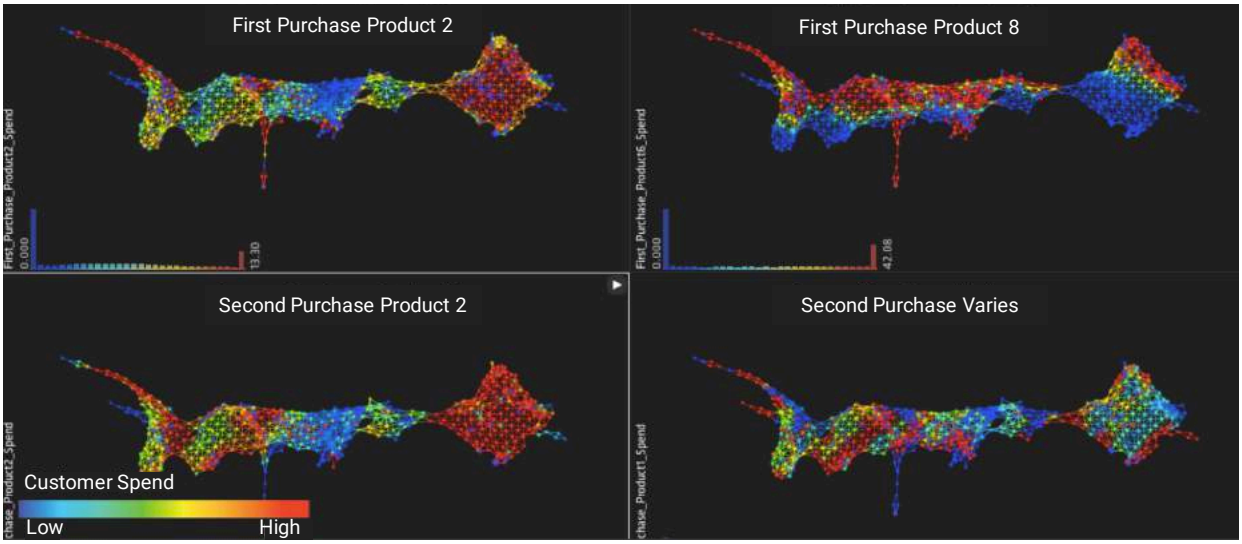


Figure 7: Analyzing Returning Customers by Buying Patterns and Spend

Model Creation

Supervised learning methods are typically employed to create models that can predict future actions or behavior. The Ayasdi Machine Intelligence Platform supports the application of multiple different algorithms to a data set when creating a topological model. The result is a group or collection of models, referred to as piecewise models or an ensemble of models, that together best represent all of the data. This ensemble of models tends to provide a far more accurate representation of the data since each is optimized for one or more different segments of the data.

A sample workflow for using the Ayasdi Platform to create a topological model contains the following steps:

1. Construct a data set with columns (attributes) of interest as well as an outcome data column.
2. Segment the data set without using the outcome data column.
3. Create node groups within the topological network.
4. Create a simple, distinct model for each node group using standard supervised learning methods like linear regression.
5. Use the model associated with each node group to accurately predict the placement of newly arriving data points within this node group.

Model Validation

Most organizations rely on a plethora of automated models to help with fraud detection, compliance, regulatory risk management, network security, and client relationship management. These models range from simple rule-based systems to those that are the result of supervised learning algorithms. One of the primary steps involved in validation or auditing exercises is the discovery of systematic errors or biases in a model. Typically, models created by supervised learning algorithms produce

systematic errors as a result of incorrect assumptions about the shape of the underlying data. The Ayasdi Platform uses TDA to uncover these errors in models.
Consider the process of validating models used to detect fraud in credit card transactions. Identifying issues in these models using the Ayasdi Platform involves the following steps:

1. Construct a data set where each data point is a transaction. Create and populate two additional data columns within this data set as follows:
    a. The predicted outcome from the model for each transaction.
    b. The actual ground truth – was each transaction fraudulent or not?
2. Segment the transaction data using all columns except those that track predicted outcomes and ground truth. The result will be a topological network.
3. Use the color scheme capability of the Ayasdi Platform to create two separate continuums of hues whose color values are controlled by the predicted outcome and ground truth data columns respectively. Apply each of these color schemes to the topological network created earlier to highlight nodes and/or data points that are outliers, i.e. that do not conform to the predicted outcome or the ground truth values.
4. Focus on the subgroups of transactions in the topological network where the model made mistakes.
5. Use the "Explain" operation in the Ayasdi Platform to retrieve a list of the data columns (features or attributes) associated with these subgroups of transactions that were not handled properly by the model. This approach can identify the attributes of questionable transactions that were not discovered by the existing fraud detection model.

AyasdiAI Vs. Open Source TDA

The AyasdiAI platform provides unique machine learning and TDA capabilities to solve some of the world's most challenging problems. The AyasdiAI Platform has been at the forefront of innovation in Topological Data Analysis and Machine Learning for the past decade. We have invested over $50 million in R&D to build an enterprise-ready product for the most highly regulated industries on the planet. A data scientist may claim that they can perform the functions and capabilities that AyasdiAI is providing but they will always fall short of this claim in a real-world setting. Areas of data science where AyasdiAI's platform sets itself apart from open source are:

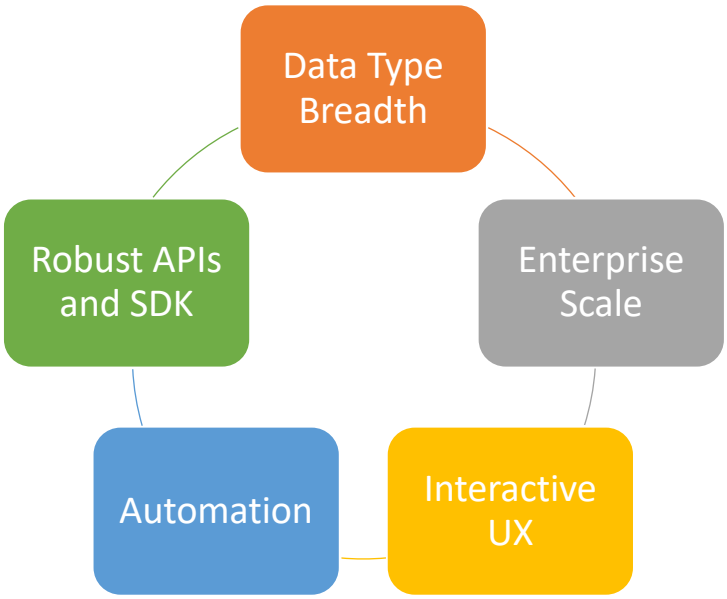
Data Type Breadth: The Platform can handle both text and numeric data with sparsity; integrating large amounts of heterogeneous and diverse data sets into common data and modelling environment at enterprise scale.

Enterprise Scale: AyasdiAI's Platform is optimized for computations on large datasets, millions of rows and hundreds of thousands of columns. Open source TDA cannot handle large datasets with consistency.

Interactive UX: Ayasdi AI provides an interactive UI, Workbench, that allows unparalleled discovery. Users are able to interact with their data, rapidly understand subpopulations, and compare them for insights. All open source code bases lack this rapid discovery interface.

Automation: AyasdiAI's Platform has a suite of auto topological model generation algorithms that intelligently evaluates a large space of metrics and filters to suggest the most optimum models. These automation techniques are very useful for both supervised and unsupervised scenarios.

Robust APIs and SDK for Productionizing Solutions: AyasdiAI's APIs and SDK give users access to a large code base that is extensively tested for fast prototyping and easy integration into their existing codebase.



There is a whole other set of criteria when it comes to enterprise suitability and readiness. Our years of experience working along the US government, financial and health institutes have tested our data security infrastructure and scale readiness time and time again. AyasdiAI's platform is built as an enterprise-grade software with high availability guarantee, user management, data security, team collaboration, multi-tenancy (can accommodate up to 1000 or more users), capabilities open source can never provide. These requirements can be summed up in the following 4 criteria:

Regulated Market	•Fully compliant with regulatory reporting and transparency requirements across multiple jurisdictions. •Full explanability and consistent documentation throughout the entire data management, model creation and results continuum.
Enterpise Readiness	•Both enterprise ready and globally deployed ensuring enterprise deployment readiness against any alternatives. •Tested and approved against enterprise class volume, resiliency, security, audit and reporting requirements out of the box.
Team Productivity	•Fully deployed multi-user workbench enabling global teams to ideate, prototype, test and deploy as a single unit in a globally consistent workflow and process to drive team productivity an order of magnitude higher than individual focused projects.
Multiple Business Challenges	•Consistently leverageable against any number of business problems as an Enterprise class platform - includes AML, human and wildlife trafficking, cyber crime, fraud and tax evasion, churn and multiple other customer predictive and inferred behaviors – driving a cost improvement vs ad hoc projects of over 60%.

A head-to-head comparison between the Ayasdi AI platform and open source data science tools always results in the same conclusion: open source does not stack up. It will always fall short in many necessary areas for enterprise-scale data science.

Summary

While organizations have successfully tackled the challenge of storing and querying vast amounts of data, they continue to lack the necessary tools and techniques for extracting useful information from highly complex data sets. Topology and TDA are well suited for analyzing complex data with potentially millions of attributes. The Ayasdi Machine Intelligence Platform uses TDA to bring together a broad range of machine learning, statistical, and geometric algorithms to create compressed representations of data. This advanced analytics software creates highly interactive visual networks that make possible rapid exploration and understanding of critical patterns and relationships in data.

Ayasdi's use of TDA augments current machine-learning techniques by ameliorating some of their intrinsic issues and reducing the dependency on increasingly scarce human expertise. Innovative companies are using TDA and the Ayasdi Machine Intelligence Platform to:

1. Precisely segment their data
2. Identify the underlying features that drive segmentation
3. Create more effective predictive models and tailor product recommendations
4. Develop, validate and improve models
5. Detect subtle anomalies in data sets

About Symphony AyasdiAI

Symphony AyasdiAI, part of the SymphonyAI Group, is the world's most advanced artificial intelligence software company. Symphony AyasdiAI helps organizations discover new and valuable insights in enterprise data. With unprecedented accuracy, transparency, and speed. Built upon over a decade of research and experience, Symphony AyasdiAI delivers insights to Fortune 500 companies and public sector organizations to capture growth, avoid risks and manage inefficiencies.
www.ayasdi.com

A Symphony Group Company

The SymphonyAI Group is the fastest growing and most successful group of B2B AI companies, backed by a $1 billion commitment to build advanced AI and machine learning applications that transform the enterprise. Symphony AI is a unique operating group of over 1,600 skilled technologist and data scientists, successful and proven entrepreneurs, and accomplished professionals, under the leadership of one of Silicon Valley's most successful serial entrepreneurs, Dr. Romesh Wadhwani.

Symphony AYASDI
555 Twin Dolphin Dr, Suite 370
Redwood City, CA 94065 USA
+1 650.704.3395
sales@ayasdi.com
ayasdi.com | @ayasdi