CSP31A Big Data Analytics

School of Computer Engineering and Technology

Big Data Analytics(PE-1)

(Computer Engineering and Technology)
(TYB.Tech)
UNIT V
Big Data Visualization Techniques

UNIT V

Big Data Visualization Techniques:

Introduction to data visualization, Data visualization factors, Challenges in Data Visualization, Analytics Techniques: Basic charts scatter plots, Histogram, advanced visualization Techniques Tree Map Circle packing, Sunburst Circular Network Diagram Parallel Coordinates Streamgraph, Plots, Graphs, Networks, Hierarchies, Reports, Introduction to D3.js

Case study: Google Analytics /Twitter Analytics

PowerBI

Introduction to Big Data Visualization

- Big data visualization is a crucial aspect of working with large and complex datasets.
- It involves representing data in visual forms like charts, graphs, maps, and other visual elements to help users understand and extract insights from the data more effectively.
- As the volume, variety, and velocity of data continue to increase, visualization techniques become essential for making sense of the information contained within big data sets.

Introduction to Big Data Visualization

Well, you might wonder why data visualization is important?

- Big data often contains vast amounts of information that can be challenging to comprehend through traditional methods.
- Visualization helps in identifying patterns, trends, outliers, and relationships within the data.
- It simplifies complex data for decision-makers, making it easier to draw meaningful conclusions.

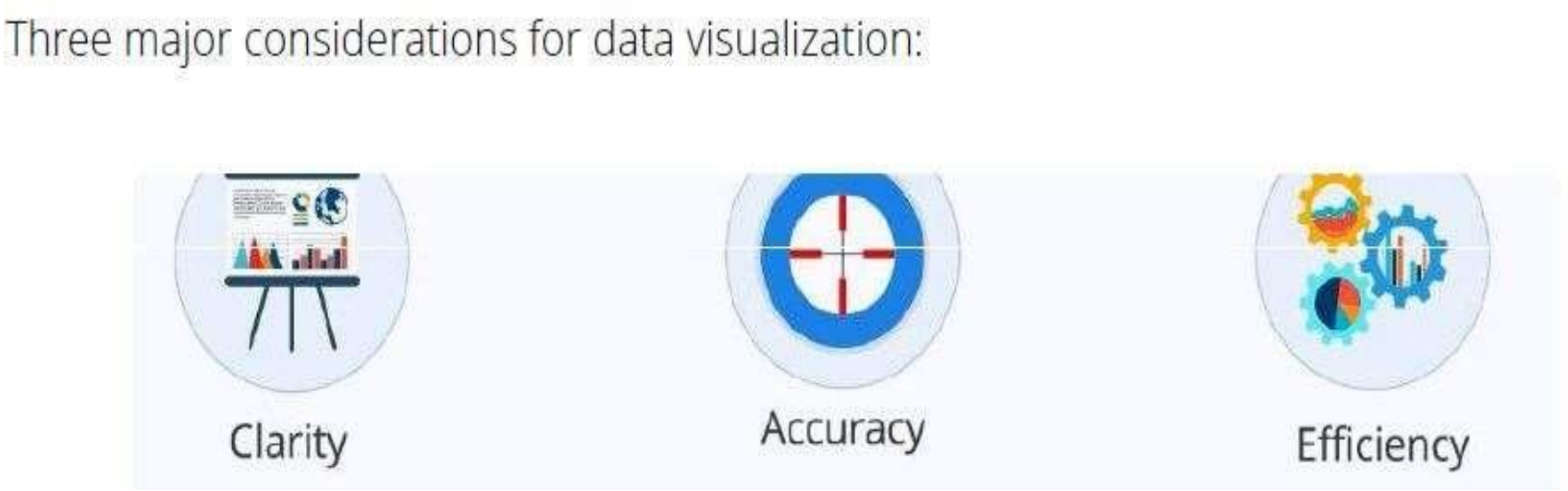
Why is big data visualization important?

The short answer is that humans don't have the capability to quickly make sense of large volumes of raw statistical information. Our eyes are not drawn to numbers, but to colors and patterns, so if we see a chart, we can quickly identify trends and patterns, and understand the meanings behind them.

The longer answer is that it enables us to:

1. Review large amounts of data
   - -The graphical form enables us to quickly make sense of large amounts of data — much faster than going over raw numbers
2. Spot trends
   - -Spotting trends within data is extremely complex, but techniques for big data visualization can make it much easier and faster — that's important because a trend that is spotted early is an opportunity that can be acted upon
3. Identify correlations
   - -Big data visualization enables us to explore entire data sets to gather insights — identifying patterns and relationships in data can provide businesses with huge competitive advantages
4. Present the data to others
   - -Visualization techniques are also an effective way to communicate any insights to others — they convey meaning very quickly and in a way that it is easy to understand

Data visualization factors



- Clarity Ensure the dataset is complete and relevant. This enables the Data Scientist to use the new patterns obtained from the data in the relevant places
- Accuracy Ensure you use appropriate graphical representation to convey the intended message
- Efficiency Use efficient visualization techniques that highlight all the data points

There are some basic factors that one needs to be aware of before visualizing the data:

- The visual effect includes the usage of appropriate shapes, colors, and sizes to represent the analyzed data
- The coordinate system helps organize the data points within the provided coordinates
- The data types and scale choose the type of data, for example, numeric or categorical
- The informative interpretation helps create visuals in an effective and easily interpretable manner using labels, title, legends, and pointers
- Diversity and heterogeneity in big data creates a big problem while visualizing that data
- Analysis speed is most challenging factor in Big data Analysis
- Handling Big data scalability the cloud computing and advanced GUI are combined with the big data

- Usually data is unstructured, to visualize Tables, texts, trees, graphs, and other meta data is used
- Providing huge Parallelization is a challenge in big data Visualization
- High Complexity and High dimensionality during Discovery process due to huge amount of data
- It is difficult to design new big data visualization tool which results efficiency
- Due to the large size and dimensions of big data the visualization becomes more challenging

Challenges to Big Data visualization

- **Data Volume:**
  - Big data sets can be massive, containing billions or even trillions of data points. Visualizing such large volumes of data can strain the computational resources and slow down rendering times.
- **Data Variety:**
  - Big data often includes a wide variety of data types, including structured and unstructured data, text, images, sensor data, and more. Visualizing this diverse data requires flexibility in visualization techniques.
- **Data Velocity:**
  - Real-time or near-real-time data streams are common in big data applications. Visualizations need to handle constant updates and present information as it arrives, which can be challenging for traditional visualization tools.
- **Data Quality:**
  - Big data can suffer from data quality issues such as missing values, outliers, noise, and inconsistencies. Visualizations need to address and possibly filter out these issues while providing accurate insights.
- **Scalability:**
  - Traditional data visualization tools may not scale well to accommodate large and growing data sets. Scalable visualization solutions are required to handle big data effectively.
- **Interactivity:**
  - Users often expect interactive features in big data visualizations to explore data from different perspectives. Building interactive features that perform well with large data sets can be complex.

- **Comprehension and Cognitive Load:**
  - With large and complex data sets, there's a risk of overwhelming users with too much information. Effective big data visualizations must strike a balance between showing detailed insights and avoiding cognitive overload.
- **Performance and Rendering:**
  - Rendering large data sets in real-time can strain both hardware and software resources. Achieving acceptable performance while visualizing big data is a constant challenge.
- **Data Integration:**
  - Big data often comes from diverse sources and platforms. Integrating these data sources for meaningful visualization can be complex and may require data preprocessing and transformation.
- **Security and Privacy:**
  - Big data may contain sensitive or confidential information. Protecting data security and privacy while visualizing data is crucial, especially in regulated industries.
- **Dimensionality Reduction:**
  - Visualizing high-dimensional data can be challenging. Techniques for reducing the dimensionality while preserving important information are necessary for effective big data visualization.
- **Tool Selection:**
  - Choosing the right visualization tools and software for big data can be a challenge. Not all tools are well-suited for handling large and complex data sets, so making the right selection is crucial.
- **User Expertise:**
  - Users may lack the expertise to interpret complex visualizations, especially in domains with specialized knowledge requirements. Designing visualizations that are understandable to the intended audience is essential.

Analytical techniques used in Big Data visualization

- Analytical techniques play a crucial role in extracting meaningful insights from big data visualizations. These techniques help analysts and data scientists uncover patterns, relationships, and trends within large and complex datasets.
  - **Descriptive Analytics:**
    - Descriptive analytics involve summarizing and aggregating data to provide an overview of its characteristics. This includes basic statistics like mean, median, mode, range, and measures of central tendency.
  - **Exploratory Data Analysis (EDA):**
    - EDA techniques help in uncovering patterns and trends within the data. Common EDA methods include data profiling, scatter plots, box plots, and histograms to gain initial insights into the data distribution.
  - **Correlation Analysis:**
    - Correlation analysis helps identify relationships between variables. Techniques like Pearson's correlation coefficient and Spearman's rank correlation are used to measure the strength and direction of relationships.
- **Regression Analysis:**
  - Regression analysis is used to model and understand the relationship between a dependent variable and one or more independent variables. It's valuable for predicting outcomes and understanding causality.

- **Cluster Analysis:**
  - Cluster analysis is used to group data points into clusters based on their similarities. Techniques like k-means clustering and hierarchical clustering can be applied to find hidden patterns in the data.
- **Principal Component Analysis (PCA):**
  - PCA is a dimensionality reduction technique that helps reduce the complexity of high-dimensional data while retaining as much relevant information as possible.
- **Time Series Analysis:**
  - Time series analysis is used for data that varies with time. Techniques like moving averages, exponential smoothing, and autoregressive integrated moving average (ARIMA) are employed to model and forecast time-series data.
- **Text Analysis and Natural Language Processing (NLP):**
  - When dealing with textual data, NLP techniques are used to extract insights. These techniques include sentiment analysis, topic modeling, and text classification.
- **Machine Learning and Predictive Analytics:**
  - Machine learning algorithms are employed for predictive analytics, classification, regression, and anomaly detection. Techniques such as decision trees, random forests, support vector machines, and neural networks are used.
- **Graph Analysis:**
  - Graph analysis techniques are applied to data with complex relationships, such as social networks or network data. Graph algorithms like centrality, community detection, and network connectivity analysis are used.
- **Geospatial Analysis:**
  - Geospatial analytics is crucial when working with location-based data. It includes techniques for spatial interpolation, hotspot analysis, and spatial clustering.

- **Time Series Forecasting:**
  - Time series forecasting techniques like exponential smoothing, ARIMA, and Prophet are used to make predictions based on historical time-series data.
- **Anomaly Detection:**
  - Anomaly detection techniques identify outliers or irregularities in the data, which can be indicative of errors, fraud, or other significant events.
- **Simulation and Monte Carlo Analysis:**
  - Monte Carlo simulations are used to model complex systems, analyze risks, and estimate probabilities by generating multiple random samples of data.
- **Data Mining:**
  - Data mining techniques, including association rule mining and frequent pattern mining, help discover hidden patterns and relationships in large datasets.

Big Data Visualization Methods

Big Data Visualization Methods include

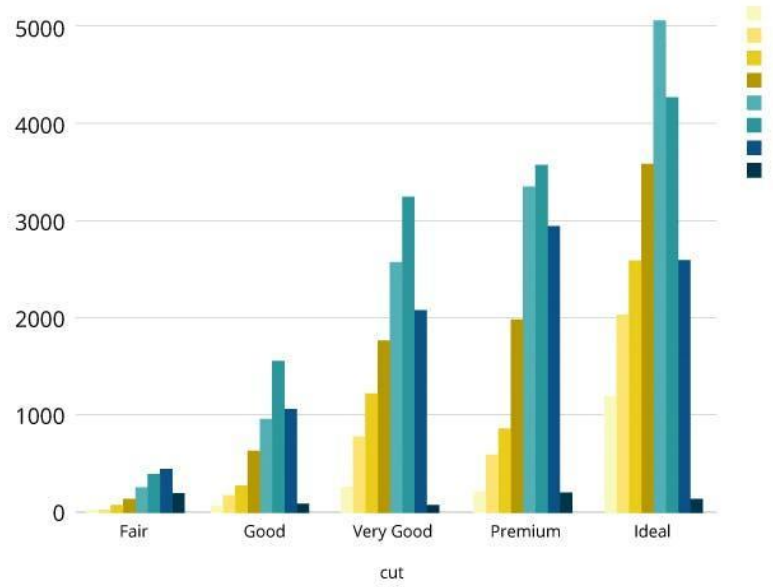
- **Bar Charts and Histograms:**
  - Bar charts are used to represent categorical data, while histograms display the distribution of continuous data by dividing it into bins or intervals. They are effective for summarizing data and identifying patterns.
- **Line Charts:**
  - Line charts are ideal for visualizing trends over time. They connect data points with lines, making it easy to see how values change continuously.
- **Scatter Plots:**
  - Scatter plots display individual data points as dots on a two-dimensional graph. They are useful for identifying relationships, correlations, and outliers in the data.
- **Heatmaps:**
  - Heatmaps use color-coding to represent data density and relationships. They are commonly used in fields like biology and finance to visualize large datasets.
- **Tree Maps:**
  - Tree maps are hierarchical visualizations that display data in nested rectangles. They are effective for showing the structure and distribution of data within categories and subcategories.

- **Choropleth Maps:**
  - Choropleth maps use color shading to represent data values on geographic maps. They are frequently used in applications involving regional or spatial data.
- **Network Graphs:**
  - Network graphs visualize relationships between data points in a network or graph structure. Nodes represent entities, and edges represent connections or interactions between them.
- **Parallel Coordinates Plots:**
  - Parallel coordinates plots are used for visualizing high-dimensional data. Each axis represents a different variable, and lines connecting points across the axes reveal patterns and relationships.
- **Sankey Diagrams:**
  - Sankey diagrams show the flow of data or resources from one category to another. They are often used to depict energy flows, financial transactions, or process analysis.
- **Word Clouds:**
  - Word clouds display text data, with word size and color indicating word frequency. They are commonly used to summarize and highlight significant terms in textual data.
- **Dendrogram:**
  - Dendrograms are hierarchical tree-like visualizations used in clustering and taxonomy analysis. They show the grouping and relationships between data points.

- **Streamgraphs:**
  - Streamgraphs are used to display data over time, showing the evolution of multiple variables as stacked, flowing streams. They provide insights into the changing composition of data.
- **3D Visualizations:**
  - 3D visualizations add an extra dimension to data representation, allowing for more complex and spatial data exploration. They are used for various applications, such as geospatial data analysis.
- **Virtual Reality (VR) and Augmented Reality (AR):**
  - Emerging technologies like VR and AR enable immersive data exploration, providing new ways to interact with and visualize big data in three dimensions.
- **Dashboard and Data Storytelling:**
  - Dashboards combine multiple visualizations to provide an integrated view of key metrics and data. Data storytelling involves creating a narrative around visualizations to convey insights effectively.
- **Interactive Visualizations:**
  - Interactive visualizations allow users to explore data by interacting with charts and graphs, changing parameters, and filtering data dynamically.
- **Real-time Visualization:**
  - Real-time visualizations continuously update data as it streams in, providing immediate insights and allowing for monitoring and decision-making in real-time.

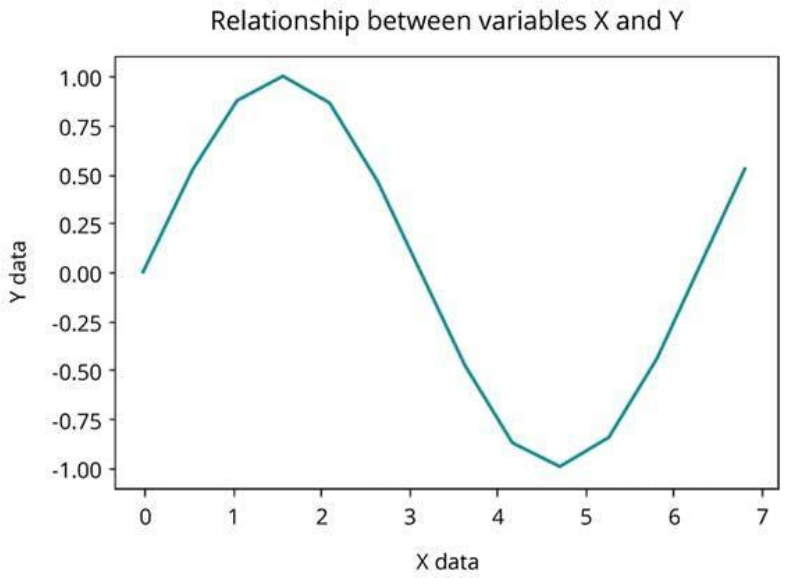
Bar chart

- Bar charts are similar to column charts — compared to them, bar charts have reversed axes and the number of bars can be much larger



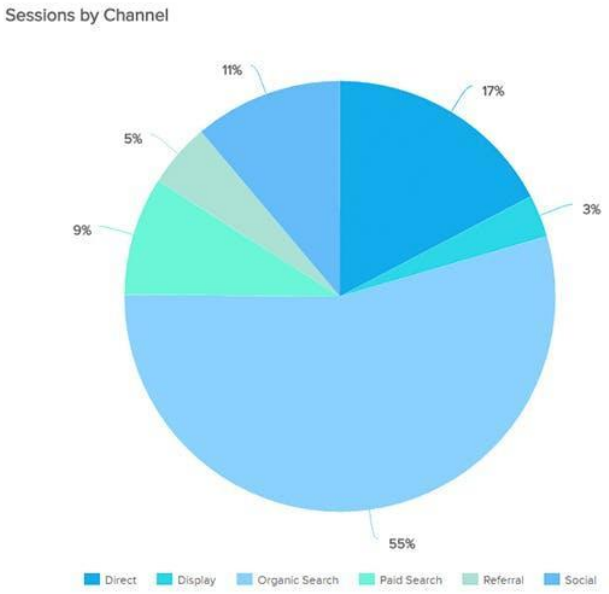
Line chart

- Line charts are used to show resulting data relative to a continuous variable — in most cases either time or money.



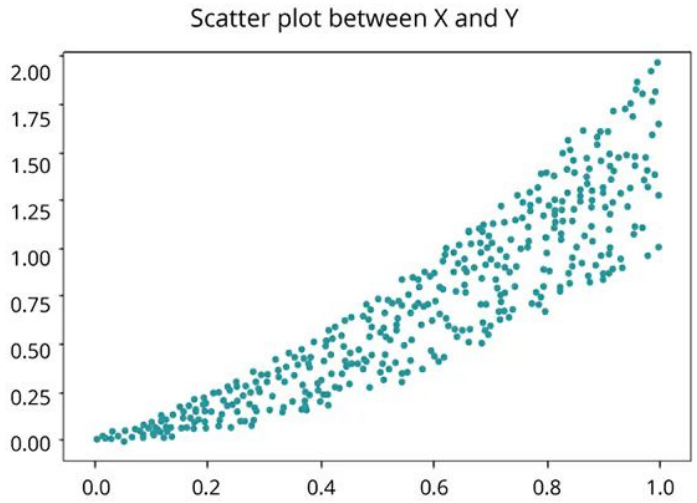
Pie chart

- Pie charts are used to compare the parts of a whole with the angle and the arc being proportional to the value represented — they are most effective when combined with text and percentages to descri



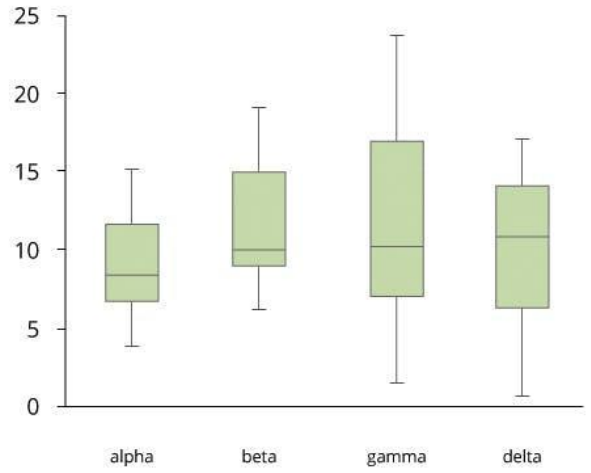
Scatter plot

- Scatter plots show two variables in the form of points on a coordinate system — by observing the distribution of the data points, we can deduct the correlation between the variables.



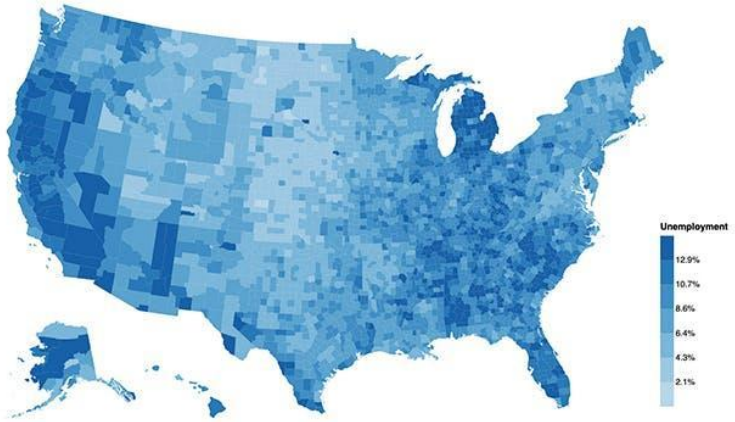
Box plot

- Box plots display a distribution of data across groups based on a five-number summary — minimum, first quartile, median, third quartile and maximum



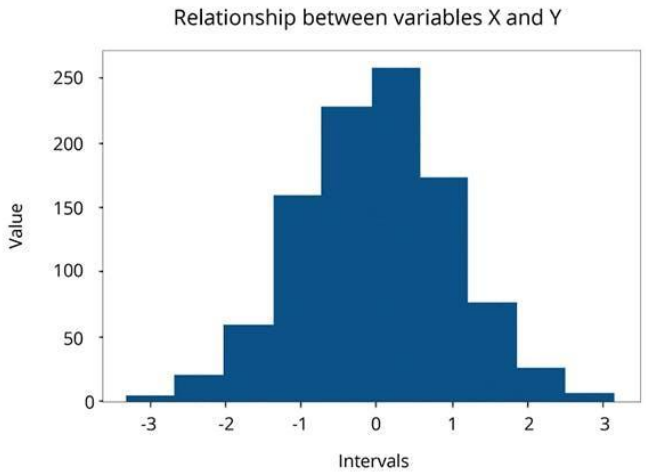
Heatmap

- Heatmaps or choropleths show the relationship between two variables and provide a rating — this is displayed through various colors or color saturation.



Histogram

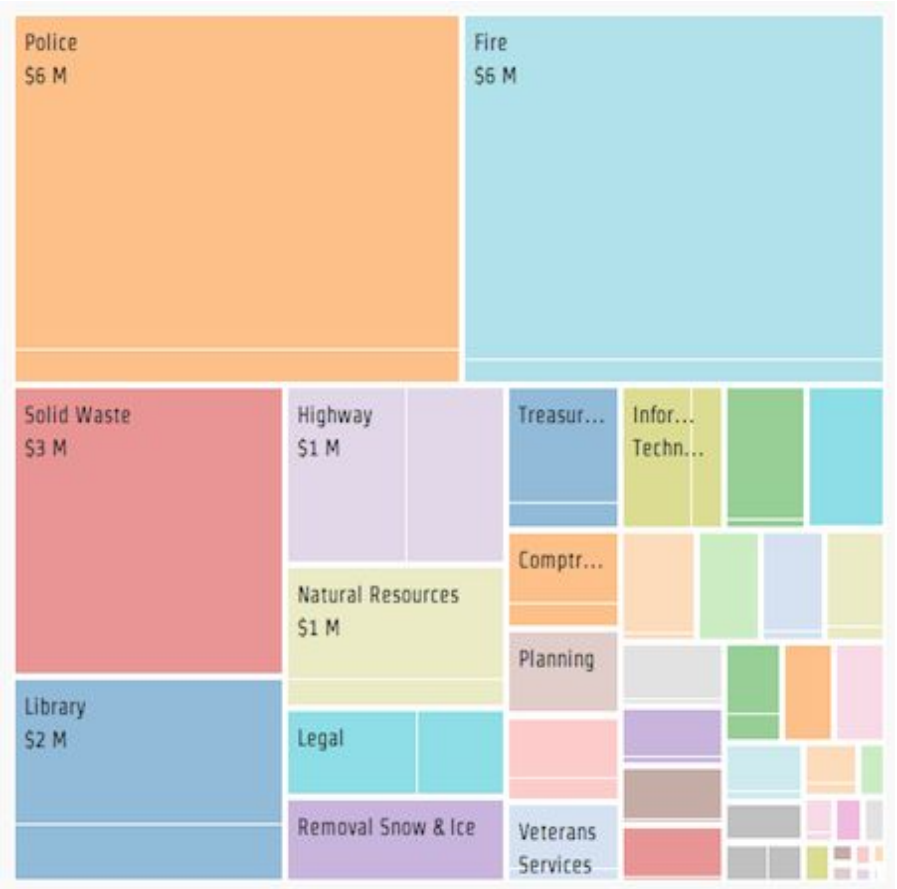
- Histograms represent the distribution of a continuous variable over a given period of time — they give an estimate as to where the values are concentrated, what are the extremes and whether there are any gaps or unusual values.



Advanced visualization Techniques:

Tree Map

- A Tree Map is a hierarchical visualization that represents data as nested rectangles, with each level of the hierarchy corresponding to a different level of nesting.
- The size and color of the rectangles can be used to convey additional information about the data within each hierarchy.
- TreeMap is represented by a root rectangle, divided into groups, also represented by the smaller rectangles which correspond to data objects from a set.
- This method of visualization is used for hierarchical data two-dimensional.
- The treemap method can be applied to large data volumes; iteratively representing data layers for each level of thehierarchy.
- This method satisfies the large data volume criterion. However, the method can only show two data dimensions presented by size and color shapes. And the data representation appears at one moment in time. So the criterion data variety and dynamicity are not met in this method



Tree Map:

- Advantages:

(i) Hierarchical grouping clearly shows data relations;

(Ii)Extreme outliers are immediately visible using special color.

- Disadvantages:

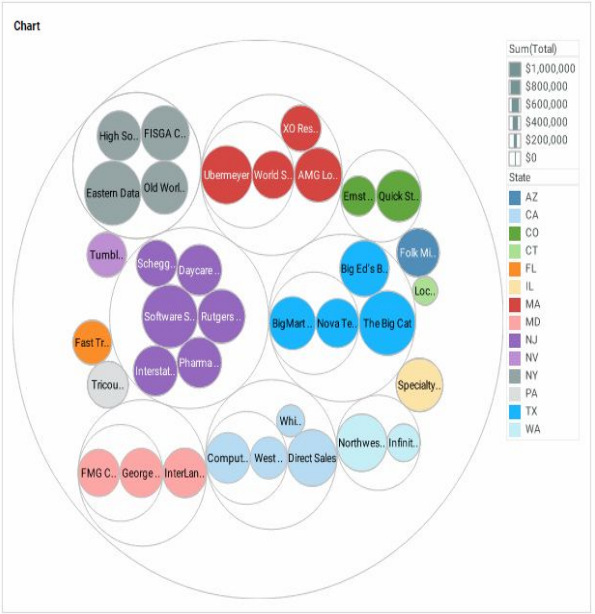
(i) Data must be hierarchical and, even more, Tree Maps are better for analyzing data sets where there is at least one important quantitative dimension with wide variations;

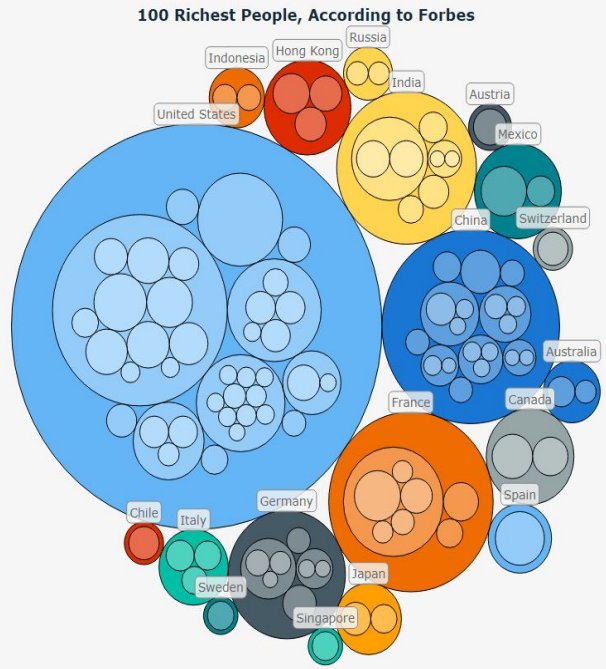
(ii) Not suitable for examining historical trends and time patterns;

(iii) The factor used for size calculation cannot have negative values

Circle Packing

- Circle packing is a visualization technique that represents hierarchical data as a series of nested circles, where each circle's size and position indicate the hierarchy and relationships among data elements.
- The sizes of circles are proportional to the values of the data points they represent.
- it is an alternative to Treemap that uses circles instead of rectangles.
- The Primitive shape is a circle; which can include circles as presented in the figure.
- The most advantage of this method is the possibility to place and percept a lot of objects with many levels of hierarchy.
- The area of each circle presents an attribute such as quantity. Color may be used to present the second fact.
- This method looks more beautiful, but it is not as space-efficient as a Treemap, as there is a lot of empty spaces within the circles.





Circle Packing:

- Advantages:

(i) space-efficient visualization method compared to Tree map

- Disadvantages:

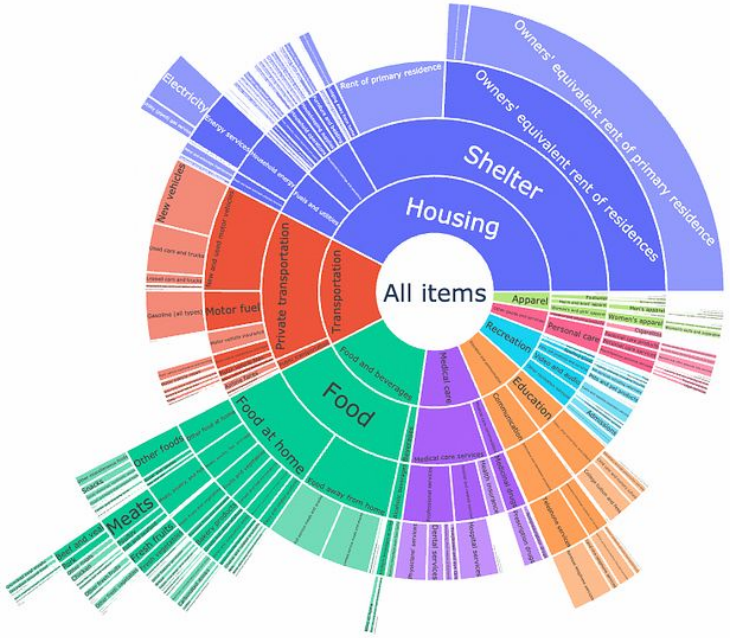
(i) Data must be hierarchical and, even more, tree maps are better for analyzing data sets where there is at least one important quantitative dimension with wide variations;

(Ii) Not suitable for examining historical trends and time patterns;

(Iii) The factor used for size calculation cannot have negative values

Sunburst:

- A Sunburst diagram is a radial representation of hierarchical data.
- It's useful for displaying data with multiple levels of categorization.
- Each ring represents a level in the hierarchy, and segments within the rings show data distribution.
- This method is a directive of treemap: it converted to a polar coordinate system.
- It is more flexible and allows repaint the whole diagram by changing the radius and arc length.
- It allows understanding large amounts of data using efficient and intuitive graphic.



Sun Burst:

- Advantages:

i) easily perceptible by most humans

- Disadvantages:

(i) Data must be hierarchical and, even more, Tree Maps are better for analyzing data sets where there is at least one important quantitative dimension with wide variations;

(ii) Not suitable for examining historical trends and time patterns;

(iii) The factor used for size calculation cannot have negative values

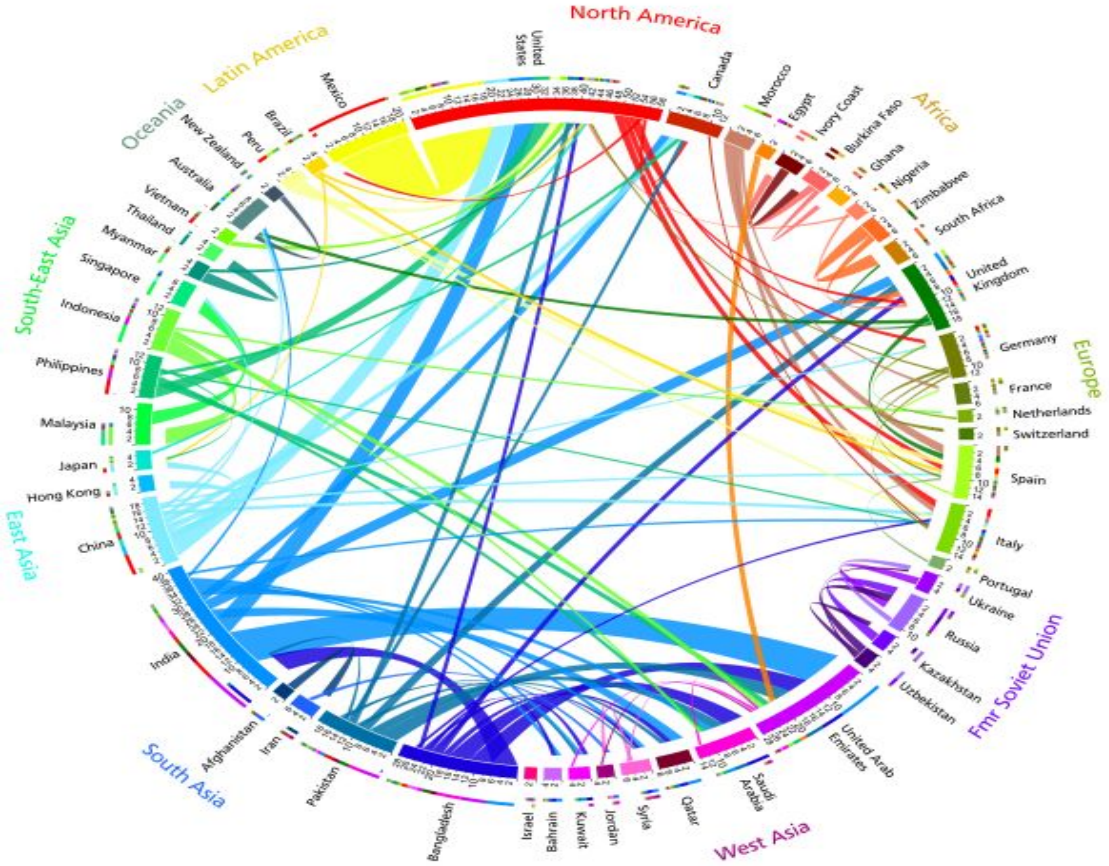
Let's visualize the tabular data using *Sunburst Charts in ChartExpo.*

Category	Months	Country	Sales Orders
Sales	June	USA	75
Sales	June	United Kingdom	67
Sales	June	Germany	90
Sales	July	USA	75
Sales	July	United Kingdom	67
Sales	July	Germany	90
Sales	April	USA	67
Sales	April	United Kingdom	90
Sales	April	Germany	75
Sales	May	USA	67
Sales	May	United Kingdom	90
Sales	May	Germany	75

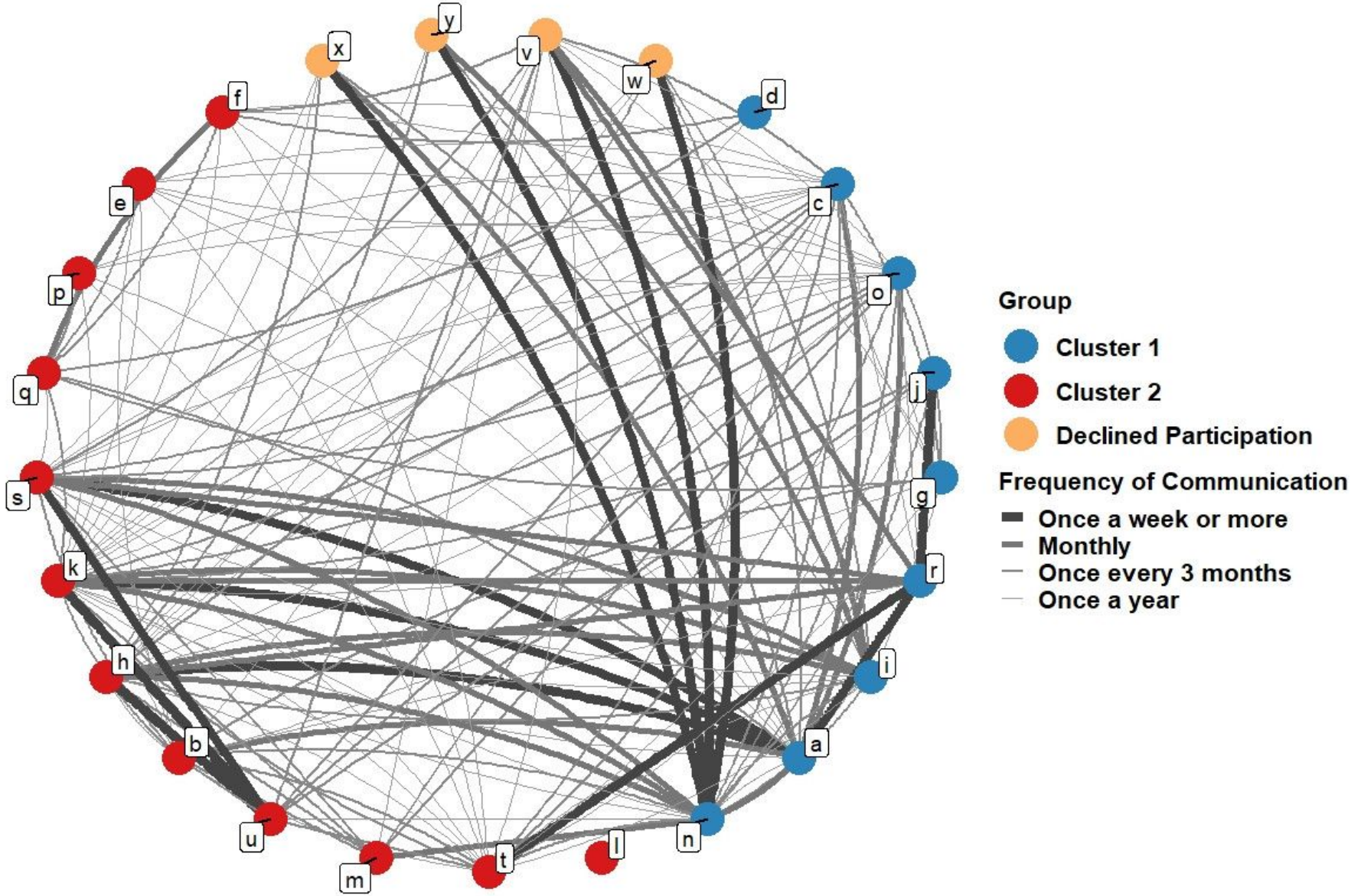
- To visualize the data (above) with Sunburst (one of the hierarchical data visualization charts), copy the table above into your Google Sheets.
- Click the **Add-on button>ChartExpo — Best Data Visualization Tool button>Open.**
- Click the **Create New Chart button** to access your fully stocked library of charts.
- Click **the Search Box** and type "**Sunburst Charts**." It should pop up together with other charts.
- Select the sheet holding your data and select the **Metrics option.** Fill in the numerical numbers (in our scenario, we'll fill in **sales Order**)
- Select the **Dimensions button** and fill in the dimensional data (in our example, we'll use **category, months, and country**)
- Visualizing your data with hierarchy charts does not have to be complex. ChartExpo makes the whole process **seamless and easy.**
- Finish the simple process by clicking the **Create Chart button**.

Circular Network Diagram:

- This chart visualizes the inter-relationships between entities.
- Data object are placed around a circle and linked by curves based on the rate of their relativeness.
- Color can be used to group the data into different categories, which aids in making comparisons and distinguishing groups.
- So, this method directly links several objects and shows how relative it is.
- It is an elegant and compact way to show networks of relations between items such as products, individuals or groups.



Migration flows among regions of the world for four five-year periods between 1990 and 2010. Migrated to are hanging and migrated from are originating from a specific color.


y
x
v
w
d
f
c
e
o
p
j
q
g
s
r
k
i
h
a
b
u
n
l
t
m
Group
Cluster 1
Cluster 2
Declined Participation
Frequency of Communication
Once a week or more
Monthly
Once every 3 months
Once a year

Circular Network Diagram :

- **advantages:**

(i) allows us to make relative data representation, which can be easily percepted

(ii) within the circle, the resolution varies linearly, increasing with radial position. This makes the center of the circle ideal for compactly displaying summary statistics or indicating points of interest.

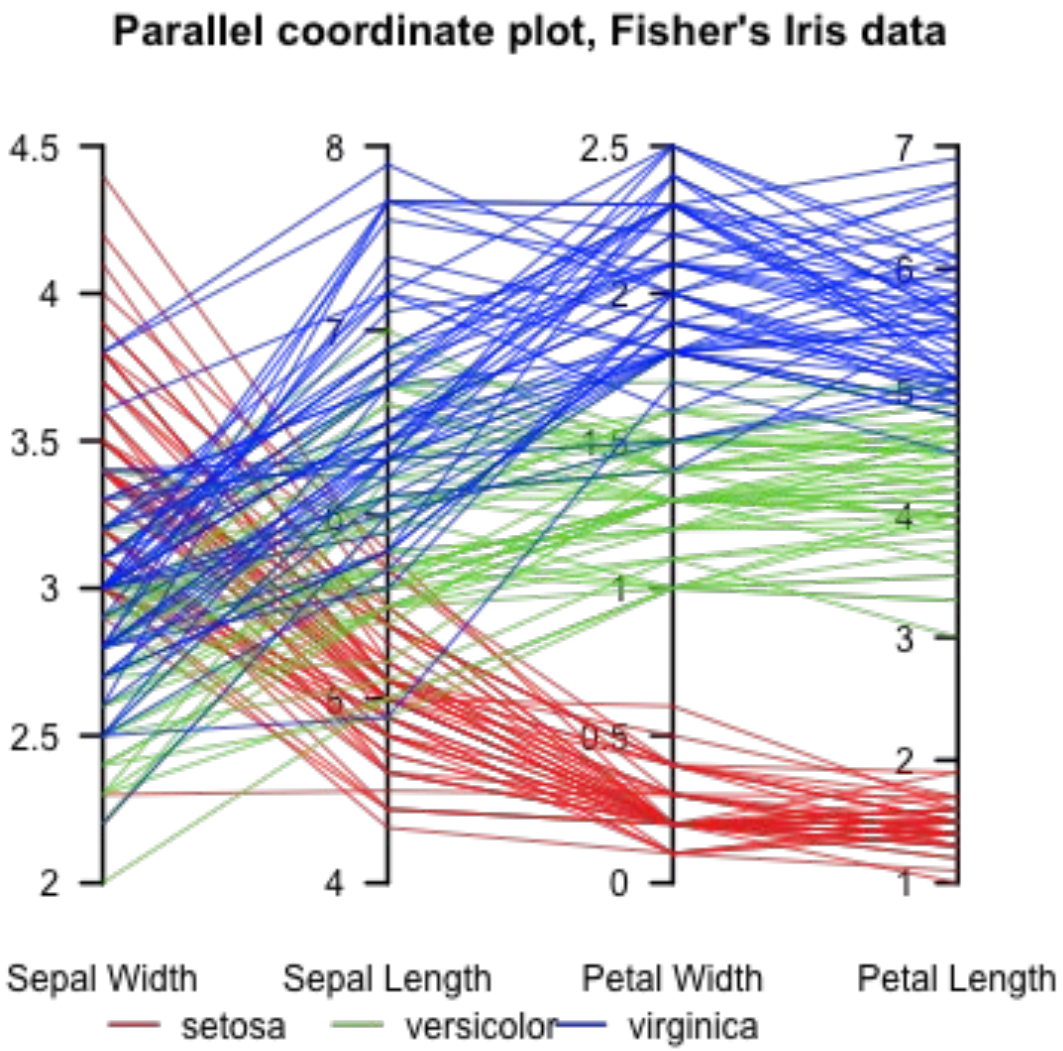
- **disadvantages:**

(i) method may end in imperceptible representation formandmay need regrouping of data objects on the screen

(ii) objects with the smallest parameter weight can be suppressed by larger ones, ending up in total mess onto the diagram

Parallel Coordinates:

- “It is a widely used visualization technique for multivariate data and high-dimensional geometry.“
- Parallel Coordinates is a technique for visualizing high-dimensional data by drawing a series of parallel axes, each representing a different data dimension
- Data points are connected by lines to reveal relationships or patterns in the data.



Parallel Coordinates :

- **advantages:**

(i) factors ordering does not influence total diagram perceptions

(ii) method allows us to analyze both whole data set of objects at once and individual data objects

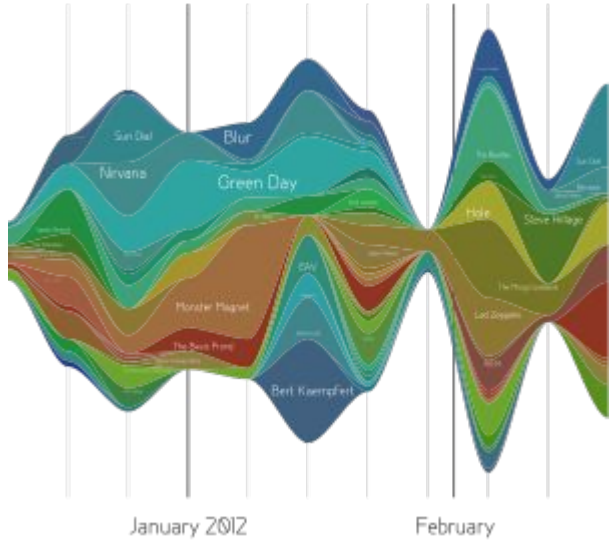
- **disadvantages:**

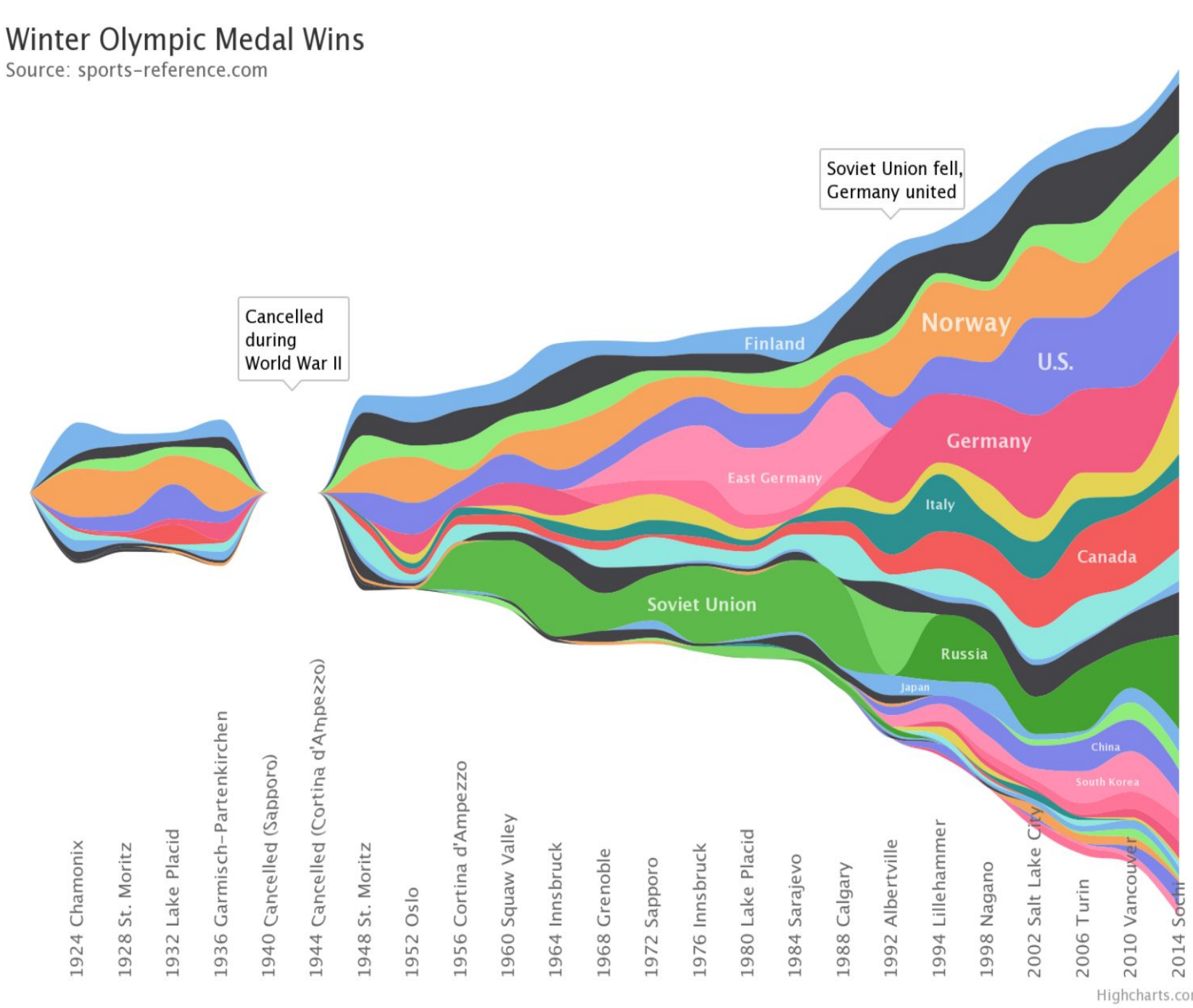
(i) method has limitation to the number of factors, shown at once

(ii) visualization dynamic data end up in changing whole data representation

Streamgraph:

- A Streamgraph is a stacked area chart that visualizes the change in composition of data over time.
- It shows how different categories contribute to a whole over a period, with smooth flowing areas.
- It is a “type of a stacked area graph, which is displaced around a central axis, resulting in flowing and organic shape.”
- Series of similar events are displayed in the timeline.
- Unstructured text is supported by this method.
- This method supports one data dimension, but it can be applied to large data.





This beautiful streamgraph, created by data journalist [Talia Bronshtein], plots the nationality of different immigrants to the United States over 200 years (1820 to 2015). And its findings jump right out. For instance, we can immediately see that during the wartime period (1939-1945) immigration to the US almost stopped.

We can also see that while most immigrants before WW2 came from countries like Austria-Hungary, Italy, and Russia; by the late 2000s, the bulk of immigration was coming from Asian and South American countries.

Comparison of Big Data Visualization Methods based on various data, large volumes data, and handles changes in time data

	Large data volume	Data variety	Data dynamics
Treemap	+	−	−
Circle packing	+	−	−
Sunburst	+	−	+
Circular network diagram	+	+	−
Parallel coordinates	+	+	+
Streamgraph	+	−	+

+ indicates satisfying the said criteria

Plots:

Plots are graphical representations of data, often used to visualize trends, relationships, and distributions. Common types of plots include scatter plots, line plots, bar charts, and histograms.

- **Example:** A scatter plot can be used to visualize the relationship between two variables, such as plotting student scores on a math test (x-axis) against scores on a science test (y-axis). Each point on the plot represents a student, and the pattern of points can reveal the correlation between the two subjects.

Graphs:

Graphs are visual representations of data that emphasize the relationships and connections between data points. They are used to model and analyze networks, dependencies, and interactions.

- **Example:** A social network graph illustrates how people are connected on social media. Each individual is a node, and connections (friendships or interactions) between individuals are represented as edges or links. Graphs can help analyze network structures and identify influential nodes.

Networks:

- Networks are specialized graphs used to model complex relationships, such as social networks, transportation systems, and the World Wide Web. They can also represent biological, economic, or communication networks.
- **Example:** A transportation network graph represents a city's roads, highways, and public transportation routes. Nodes can represent intersections, bus stops, or train stations, while edges denote the connections between them. Network analysis helps optimize transportation routes and infrastructure.

Reports:

- Reports are textual or visual summaries of data that provide detailed insights, analysis, and recommendations. Reports often include tables, charts, and visualizations to convey information effectively.
- **Example:** A financial report for a company may include tables showing revenue, expenses, and profit figures, along with line charts depicting revenue growth over time. Reports help stakeholders make data-driven decisions by presenting information in a structured and informative manner.

- Each of these components plays a unique role in data visualization, and the choice of which to use depends on the nature of the data and the objectives of the analysis.

- They can be combined and integrated to create comprehensive and insightful data presentations, aiding in data exploration, analysis, and decision-making.

Introduction to D3.js

- **D3.js**, or Data-Driven Documents, is a JavaScript library commonly used for creating dynamic and interactive data visualizations in web browsers.
- D3.js provides a powerful set of tools for binding data to DOM (Document Object Model) elements and creating visualizations that can be easily updated as the data changes.
- It is a JavaScript library to manipulate documents based on data, which-
    - Draws chart
    - Visualizes data
    - can be used to develop real time dashboards and
    - Does not provide pre-defined charts

Features of D3.js

- **Extremely flexible**
  - You can draw any data driven shape for visualization
- **Very fast and easy to use**
  - It makes use of existing web technologies like HTML, SVG and CSS
- **Works with large datasets**
- **Declarative programming**
  - Supports data driven transformations to the document
  - e.g. you can use to generate HTML table from array of numbers or you can generate bar charts applying different transformations
- **Promotes Code reusability**
- **Supports wide variety of curve generating functions**
- **Allows manipulation of Document Object Model(DOM)**
  with easy to use APIs

Benefits of D3.js

- **Data-Driven Approach:**
  - D3.js is built around a data-driven approach. This means visual elements are directly tied to data, making it easy to bind data to DOM elements and create dynamic visualizations. When the data changes, the visualization can be updated seamlessly.
- **Flexibility and Customization:**
  - D3.js provides a high level of flexibility and customization. Developers have full control over the appearance and behavior of visual elements, allowing for the creation of unique and tailored visualizations. This flexibility is crucial for handling diverse datasets and accommodating specific design requirements.
- **Versatility of Visualizations:**
  - D3.js supports a wide range of visualization types, from basic charts like bar charts and line charts to more complex visualizations like hierarchical visualizations, choropleth maps, and force-directed graphs. This versatility makes D3.js suitable for various data representation needs.

- **Active Community and Resources:**
  - D3.js has a large and active community of developers. This community contributes to a wealth of resources, tutorials, examples, and plugins that are available online. The active community also means that developers can find support and solutions to common challenges.
- **Open Source and Extensible:**
  - D3.js is an open-source library, which means it is freely available and can be modified to suit specific project requirements. The open nature of D3.js encourages collaboration and allows developers to contribute to its development.
- **Data Transformation and Manipulation:**
  - D3.js includes a range of functions for transforming and manipulating data. These functions can be used to scale data, create axes, calculate layouts, and perform other data-related tasks, providing a comprehensive toolkit for data manipulation.
- **Compatibility with Web Standards:**
  - D3.js is designed to work seamlessly with web standards, making it compatible with modern web browsers. It can be integrated into web applications and used alongside other web technologies.

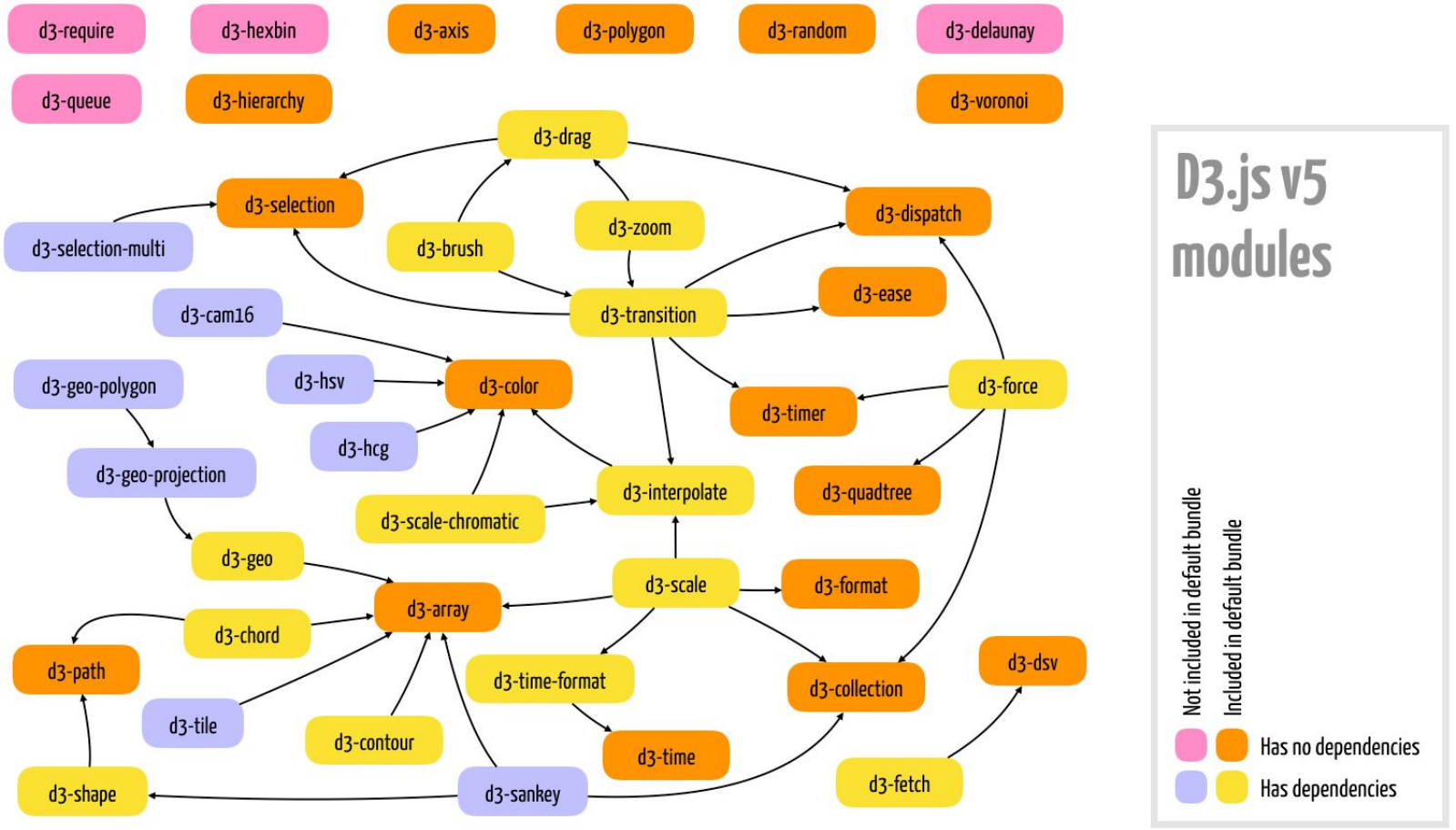
- **Interactive Capabilities:**

  D3.js makes it easy to add interactivity to visualizations. Developers can respond to user interactions, such as mouse clicks or hovers, to provide additional information or enable exploration of the data. This interactivity enhances user engagement and understanding of the data.

- **Scalable Vector Graphics (SVG):**
  - D3.js leverages SVG for drawing visual elements. SVG is a web-standard format for vector graphics, providing a flexible and scalable way to create shapes, lines, and other graphical elements directly in the browser. SVG graphics are resolution-independent and can be easily styled with CSS.

- **Transition Animations:**
  1. D3.js facilitates smooth transitions between different states in visualizations. Transitions can be used to create animated effects when data changes, enhancing the user experience and making visualizations more engaging and informative.

Modules of D3.js



1.Selections (d3-selection):

- The **d3-selection** module is fundamental to D3.js, enabling the selection of DOM elements and their manipulation. It includes methods for selecting, modifying, appending, and removing elements in the DOM.

2.Scale (d3-scale):

- The **d3-scale** module provides functions for creating scales, which are used to map data values to visual representation attributes, such as position and color. It includes linear, logarithmic, and ordinal scales.

3.Axes (d3-axis):

- The **d3-axis** module facilitates the creation of axes for visualizations. It includes functions to generate axes based on scale configurations and supports various orientations (top, bottom, left, right).

4.Shapes (d3-shape):

- The **d3-shape** module assists in creating common shapes for visualizations, such as lines, areas, curves, and symbols. It includes generators for these shapes, providing an easy way to represent data graphically.

5.Transition (d3-transition):

- The **d3-transition** module handles transitions between different states in visualizations. It provides methods for creating smooth animations when elements are added, updated, or removed.

6.Hierarchy (d3-hierarchy):

- The **d3-hierarchy** module is useful for working with hierarchical data structures. It includes functions for creating tree layouts, partition layouts, and other hierarchical visualizations.

7.Force Simulation (d3-force):

- The **d3-force** module is used for creating force-directed graphs. It provides a physics simulation engine that models forces such as gravity, charge, and links to arrange nodes in a visually pleasing way.

8.Color (d3-color and d3-scale-chromatic):

- The **d3-color** module provides utilities for working with colors, including color interpolation and conversion functions. The **d3-scale-chromatic** module includes pre-built color scales for creating visually appealing color mappings.

9.Time (d3-time):

- The **d3-time** module includes functions for working with time-based data. It provides time scales, formats, and utilities for handling dates and times.

10. **Geo (d3-geo and d3-geo-projection):**
    - The **d3-geo** module supports the creation of geographic visualizations, including map projections, path generators, and utilities for working with GeoJSON data. The **d3-geo-projection** module extends this functionality with additional map projections.

11. **Voronoi Diagram (d3-voronoi):**
    - The **d3-voronoi** module helps in creating Voronoi diagrams, which partition a plane into regions based on proximity to a set of points. This is useful for spatial data analysis.

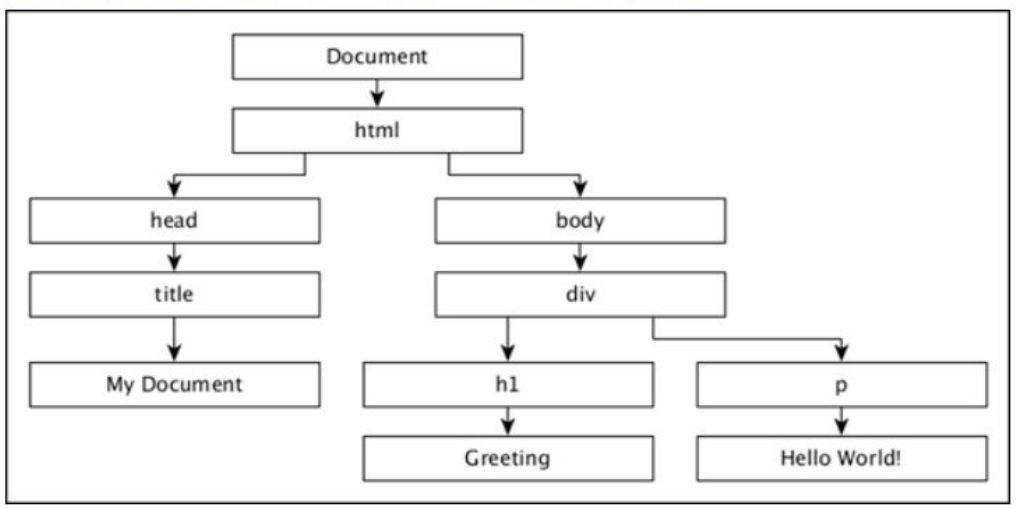
12. **Brush (d3-brush):**
    - The **d3-brush** module provides support for creating brushable areas, allowing users to interactively select regions within a visualization. It's commonly used for creating interactive charts with zooming and panning.

- HTML
  - Used to structure the content of the web page
- Document Object Model (DOM)
  - After reading HTML, its converted into hierarchical structure
- Cascading Style Sheets (CSS)
  - CSS styles make the web pages pleasant with colors, sizes, fonts etc.
  - Style sheet language describes the presentation of the HTML/XML document
- Scalable Vector Graphics (SVG)
  - Is a way to render images on web
  - Its not an image but a way to create images using text input
  - Images created with SVG don't distort on resizing browsers
  - e.g. *<rect x = "100" y = "50" width = "300" height = "200" fill="red"></rect>*
- *JavaScript*
  - *Loosely typed client side scripting language which executes in local browser*
  - *Provides interactivity to the web user interface*
  - *Implements ECMSScript Standards*

```
<!DOCTYPE html>
<html lang = "en">
   <head>
      <title>My Document</title>
   </head>

   <body>
      <div>
         <h1>Greeting</h1>
         <p>Hello World!</p>
      </div>
   </body>
</html>
```

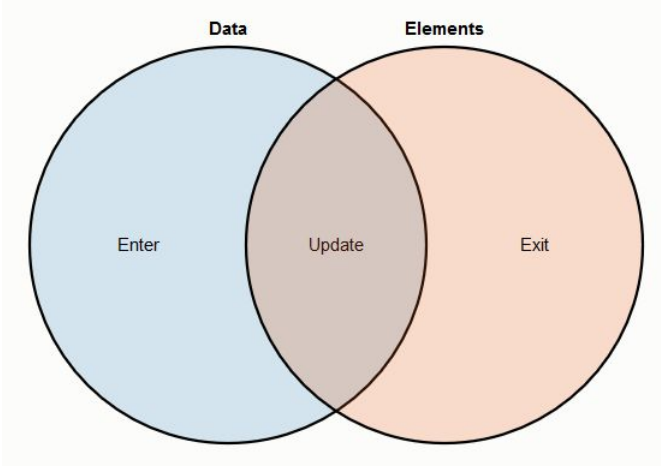
The document object model of the above HTML document is as follows,



- D3.js Selections
  - We can select elements and apply various transformations on them.
- Data Binding
  - we can populate or manipulate DOM elements in real-time.
- Creating SVG Elements
  - Scalable Vector Graphics (SVG) is a way to render graphical elements and images in the DOM.
  - As SVG is vector-based, it's both lightweight and scalable.
  - D3 uses SVG to create all its visuals, and therefore it is a core building block of the library.
- Event Handling
  - D3 also supports built-in and custom events which we can bind to any DOM element with its listener.

- It is based on CSS selectors
- Allows us to select DOM elements (i.e.paragraph, div, head, body, attributes, classes etc.) within the web page
  - select()
    - Selects one DOM element based on the CSS selector
      - e.g. d3.select("body") => *Selects the "body" element from the DOM*
    - Different manipulations can be done to the selected DOM elements
      - e.g. d3.select("body").style("background-color", "black");
      - d3.select("div.myclass").append("span");
  - selectAll()
    - Selects all elements based on CSS selector
      - e.g. d3.selectAll("p") => Selects all paragraphs
    - Different manipulations can be done on the selected DOM elements
      - e.g. d3.selectAll("p").style("color", "blue"); => Applies style to all the selected paragraphs
    - Elements may be selected using a variety of predicates, including containment, attribute values, class and ID

- It is a way of joining DOM elements to the data
- e.g. If we want to join circle with each data element
  - So instead of telling D3 to create circles, we are telling D3 that selection circle should correspond to data, this concept is called data join
- Data points joined with existing elements produce “update” section in the above image (inner overlapping section)
- Leftover unbound data produce the “enter” selection
- Any remaining unbound elements produce “exit” selection (which represents remove elements)



```
var circle = svg.selectAll("circle")
  .data(data);

circle.exit().remove();

circle.enter().append("circle")
    .attr("r", 2.5)
  .merge(circle)
    .attr("cx", function(d) { return d.x; })
    .attr("cy", function(d) { return d.y; });
```

```
svg.selectAll("circle")
  .data(data)
  .enter().append("circle")
    .attr("cx", function(d) { return d.x; })
    .attr("cy", function(d) { return d.y; })
    .attr("r", 2.5);
```

- Below code recomputes the data join and maintains the desired correspondence between elements and data.
- If the new dataset is smaller than the old one, the surplus elements end up in the exit selection and get removed.
- If the new dataset is larger, the surplus data ends up in the enter selection and new nodes are added.
- If the new dataset is exactly the same size, then all the elements are simply updated with new positions, and no elements are added or removed.
- If a given enter, update or exit selection happens to be empty, the corresponding code is a no-op

```
var circle = svg.selectAll("circle")
  .data(data);

circle.exit().remove();

circle.enter().append("circle")
    .attr("r", 2.5)
  .merge(circle)
    .attr("cx", function(d) { return d.x; })
    .attr("cy", function(d) { return d.y; });
```

- SVG stands for Scalable Vector Graphics
- XML based vector graphics format which provides ways to draw shapes like line, circle, eclipse etc.
- Features of SVG
  - Vector based image format and is text based
  - Similar in structure to HTML
  - SVG properties can be specified as attributes
  - Works in browsers

- The code with the output is as follows

```
<!DOCTYPE html>
<html lang = "en">
   <head>
      <script src = "d3.min.js"></script>
   </head>

   <body>
   <h1>This is my first D3.js</h1>
         <svg width = "500" height = "500">
            <rect x = "100" y = "50" width = "300" height = "200" fill="red"></rect>
         </svg>
      <script>
         // write your d3 code here..

      </script>
   </body>
</html>
```

This is my first D3.js

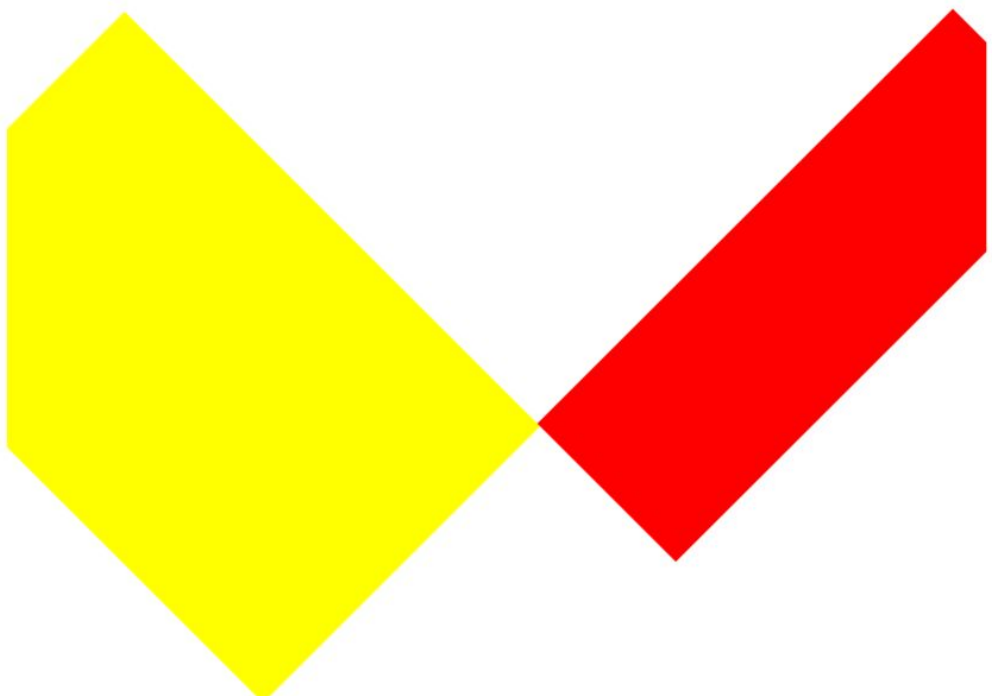
- SVG supports transformation with the help of attribute “transform”
- Following transformations are supported
  - **Translate** : takes 2 values tx and ty which refers to translation along x and y axis respectively e.g. translate(20 20)
  - **Rotate**: takes 3 options cx, cy and angle. cx and cy specifies the center of rotation in x and y direction and angle specifies the angle of rotation. e.g rotate(60) rotates he figure by 60 degrees with cx=cy=0 (origin)
  - **Scale**: takes sx and sy as input which refers to scaling factor along x and y axis respectively
  - **Skew**(SkewX and SkewY): It takes single option; skew angle refers to the angle along x-axis for skewX e.g. skewX(20)

```
<!DOCTYPE html>
<html lang = "en">
   <head>
      <script src = "d3.min.js"></script>
   </head>

   <body>
   <h1>This is my first D3.js</h1>
         <svg width = "500" height = "500">
            <g transform = "translate(60,60) rotate(45)">
                <rect x = "0" y = "0" width = "300" height = "200" fill="yellow" ></rect>
            </g>
            <rect x = "0" y = "100" width = "300" height = "100" fill="red" transform = "translate(200,200) rotate(-45)"></rect>
         </svg>
      <script>
         // write your d3 code here..

      </script>
   </body>
</html>
```

This is my first D3.js

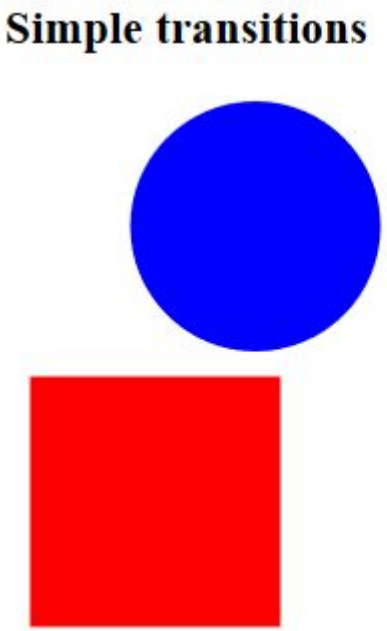


- Transition is process of changing from one state to another state
- This method supports most of the selection methods like attr(), style() etc.
- Doesn’t support append and data methods, so need to be called before transition

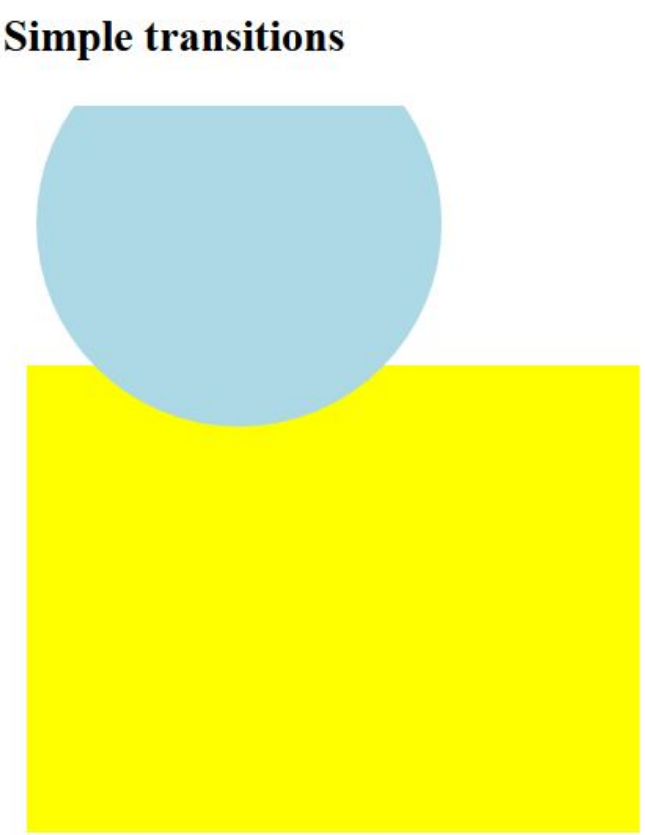
```
<!DOCTYPE html>
<html>
   <head>
      <script type = "text/javascript" src = "d3.min.js"></script>
   </head>

   <body>
      <h3>Simple transitions</h3>
      <svg width = "500" height = "500">
        <rect x = "0" y = "100" width = "100" height = "100" fill="red" transform = "translate(10,10)"></rect>
        <circle cx = "100" cy = "50" r = "50" fill="blue"</circle>
      </svg>
      <script>
         d3.select("circle").transition()
            .style("fill", "lightblue")
            .delay(10)
            .attr("r", function(d) { return Math.floor(Math.random()*100); });
        d3.select("rect").transition()
            .delay(10)
            .style("fill", "yellow")
            .attr("height", function(d) { return Math.floor(Math.random()*200); })
            .attr("width", function(d) { return Math.floor(Math.random()*300); });
      </script>
   </body>
</html>
```

Before Transition



After Transition



- Transitions are limited form of Key Frame Animation with only 2 key frames start and end
- duration() method
  - duration method allows the transition to occur over a period of duration specified in duration method
  - e.g. Animation.html
  - The values from start to end are interpolated using internal interpolate methods
  - D3 also supports following interpolate methods
    - interpolateNumber for numbers
    - interpolateRgb for colors (rgb)
    - interpolateString for strings
- delay() method
  - delay method delays the transition by the amount specified in delay method

- D3.js can be used to draw following charts
  - Bar Chart
  - Circle Chart
  - Pie Chart
  - Donut Chart
  - Line Chart
  - Bubble Chart etc.

- Bar Chart
  - Bar chart is used to show the comparison of the values, frequency or measure of something for different descrete categories/groups
  - e.g. Quarterly sales figures of any organization
  - Bar charts can be drawn horizontally or vertically

- Pie Chart
  - Important functions
  - d3.pie() – creates pie chart
  - d3.arc() – to draw the arc of the pie chart
  - d3.csv(filename, callbckfunction()) – used to load csv file data

- Line Chart- Imp functions
  - d3.scaleTime() - used for scaling time especially on x axis)
  - d3.scaleLinear() – used for linear scaling (used on y axis)
  - range() – specifies the range of the available canvas for drawing
  - d3.line() – used to define line chart
  - Uses .x and .y accessors to access the x and y data
  - d3.csv(filename, callbckfunction()) – used to load csv file data
  - x.domain – specifies range of data values on x axis
  - y.domain – specifies range of data values on y axis
  - d3.path – used to actually draw the lines on the canvas (takes data and line chart as input)
  - d3.axisBottom(x) – add x axis at the bottom
  - d3.axisLeft(y) – add y axis at the bottom

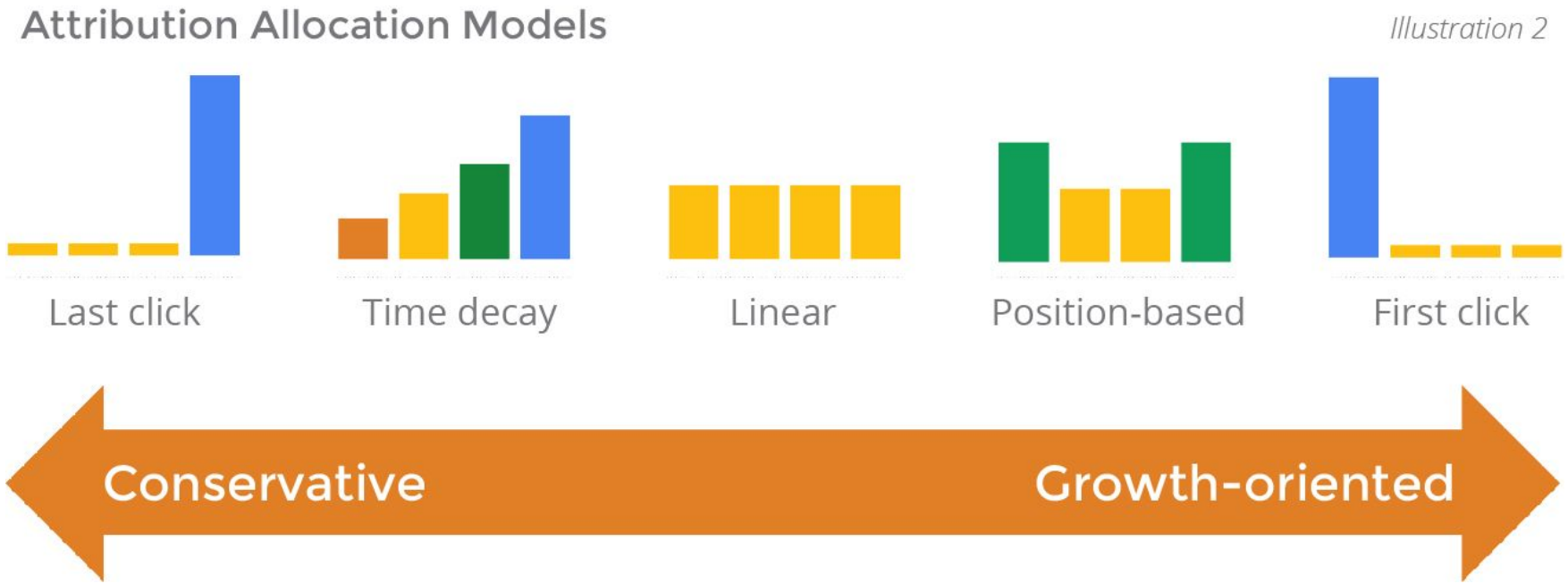
- There are numerous web analytics tools available today. Most of these web analytics tools are part of hosted web analytics services offered by handful of companies such as Google, IBM, and Mint etc.
- While most features of these web analytics tools are offered as free services, some feature are offered in premium subscription

- **Features of Google Analytics-** Google Analytics offers varied features for businesses to get the hold over user behavior on web site, which helps businesses, formulate their web strategies.
  - Advertising and Campaign Performance
  - Analysis and Testing
  - Audience Characteristics and Behavior
  - Cross-device and cross-platform measurement
  - Product Integrations
  - Sales and Conversions
  - Site and App Performance

- Google Analytics has various products under its umbrella such as:
  - Google Analytics
  - Google Analytics 360
  - Google Tag Manager
  - Google Big Query etc.
- These products have assisted many big brands to achieve their milestones with their new and innovative approach.

- Example: Dominos Google Analytics
  - Realizes an immediate 6% increase in monthly revenue
  - Saves 80% YOY in ad serving and operations costs
  - Increases agility with streamlined tag management
  - Obtains easy access to powerful reporting and customized dashboards
- Achieved this by making use of Google Analytics Premium, Google Tag Manager, and BigQuery to integrate digital data sources and CRM data

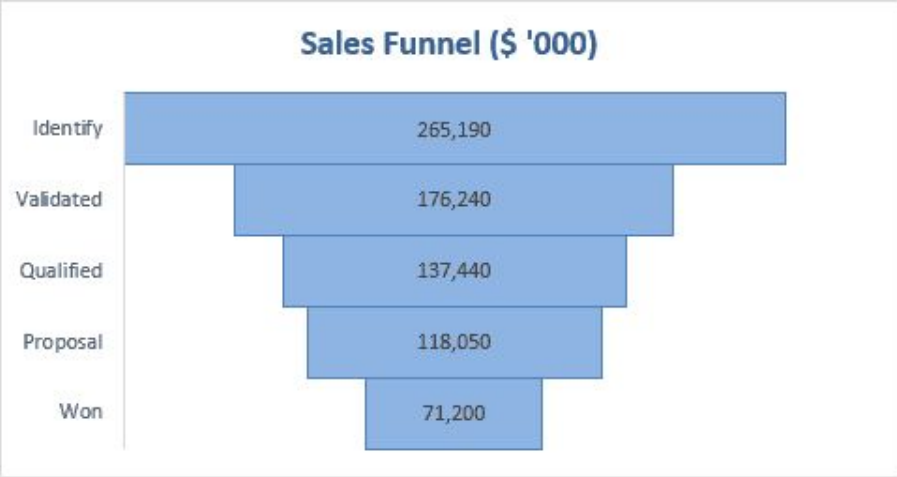
- Attribution models are used to assign credit to touchpoints in the customer journey.
- Attribution aims to help marketers get a better picture of when and how various marketing channels play contribute to conversion events. That information can then be used to inform future budget allocations.


Attribution Allocation Models
Illustration 2
Last click
Time decay
Linear
Position-based
First click
Conservative
Growth-oriented

- Last-click attribution.
  - With this model, all the credit goes to the customer's last touchpoint before converting.
  - This model doesn't take into consideration any other engagements the user may with the company's marketing efforts leading up to that last engagement.
- First-click attribution.
  - It gives 100 percent of the credit to the first action the customer took on their conversion journey.
  - It ignores any subsequent engagements the customer may have had with other marketing efforts before converting.
- Linear attribution.
  - This multi-touch attribution model gives equal credit to each touchpoint along the user's path.

- Time decay attribution.
  - This model gives the **touchpoints that occurred closer to the time of the conversion** more credit than touchpoints further back in time.
  - The closer in time to the event, the more credit a touchpoint receives.
- U-shaped attribution.
  - The first and last engagement get the most credit and the rest is assigned equally to the touchpoints that occurred in between.
  - In Google Analytics, the first and last engagements are each given 40 percent of the credit and the other 20 percent is distributed equally across the middle interactions.

- Funnel Reporting



Example: Trump Excel sales funnel report (Funnel Leakage

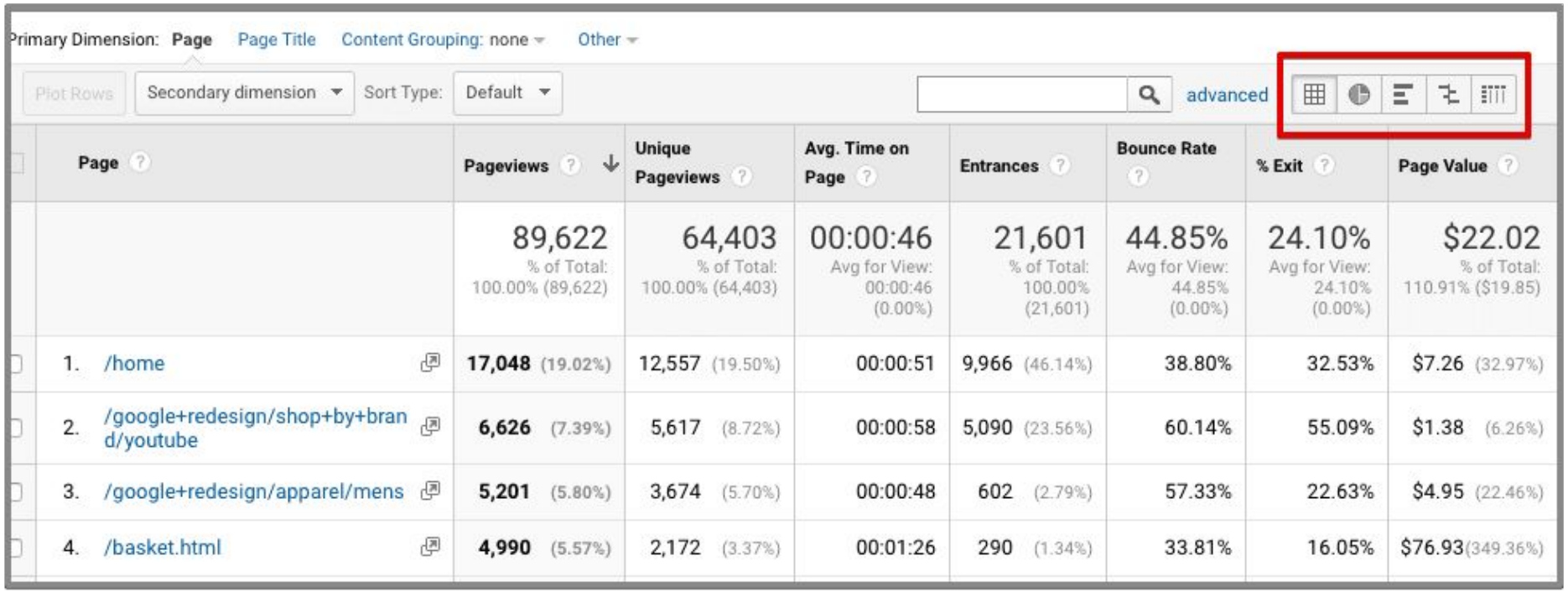
Funnel Stage	Value (in $ 000s)	Funnel Visualization	Conv %
Initial contact	$ 13,210		
Application of Initial Fit Criteria	$ 8,250		62%
Sales lead	$ 7,400		90%
Need identification	$ 6,200		84%
Qualified prospect	$ 4,847		78%
Proposal	$ 3,215		66%
Negotiation	$ 2,020		63%
Closing	$ 963		48%
Deal Transaction	$ 904		94%

Example: Chandoo ,using the two major elements to every sales funnel report (i.e.the value of opportunities/opportunities count, and the phases)

- Google Analytics offers five different visualizations to analyze your data in every report by default.
  - Tabular Reports
  - Pie Charts
  - Performance
  - Comparison
  - Pivot tables

- Tabular Reports
  - Core reporting APIs gives you access to most of the report data in Google Analytics.
  - With the Core Reporting API you can:
    - Build custom dashboards to display Google Analytics data.
    - Save time by automating complex reporting tasks.
    - Integrate your Google Analytics data with other business applications.
  - There are 3 fundamental concepts underlying the Core Reporting API:
    - How reports relate to users and views (profiles).
    - The structure of a report and how to build queries
    - Working with the API response

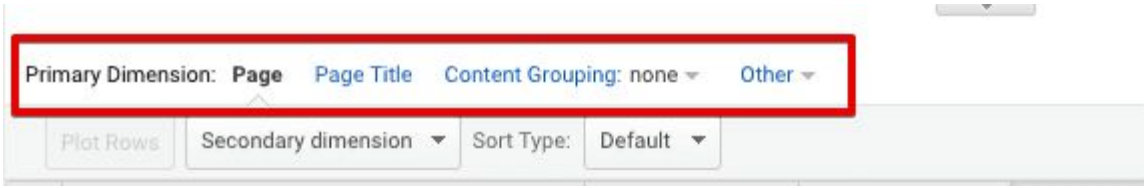
- Tabular Reports



	Page	Pageviews	Unique Pageviews	Avg. Time on Page	Entrances	Bounce Rate	% Exit	Page Value
		89,622 % of Total: 100.00% (89,622)	64,403 % of Total: 100.00% (64,403)	00:00:46 Avg for View: 00:00:46 (0.00%)	21,601 % of Total: 100.00% (21,601)	44.85% Avg for View: 44.85% (0.00%)	24.10% Avg for View: 24.10% (0.00%)	$22.02 % of Total: 110.91% ($19.85)
1.	/home	17,048 (19.02%)	12,557 (19.50%)	00:00:51	9,966 (46.14%)	38.80%	32.53%	$7.26 (32.97%)
2.	/google+redesign/shop+by+brand/youtube	6,626 (7.39%)	5,617 (8.72%)	00:00:58	5,090 (23.56%)	60.14%	55.09%	$1.38 (6.26%)
3.	/google+redesign/apparel/mens	5,201 (5.80%)	3,674 (5.70%)	00:00:48	602 (2.79%)	57.33%	22.63%	$4.95 (22.46%)
4.	/basket.html	4,990 (5.57%)	2,172 (3.37%)	00:01:26	290 (1.34%)	33.81%	16.05%	$76.93 (349.36%)

- # Tabular Reports

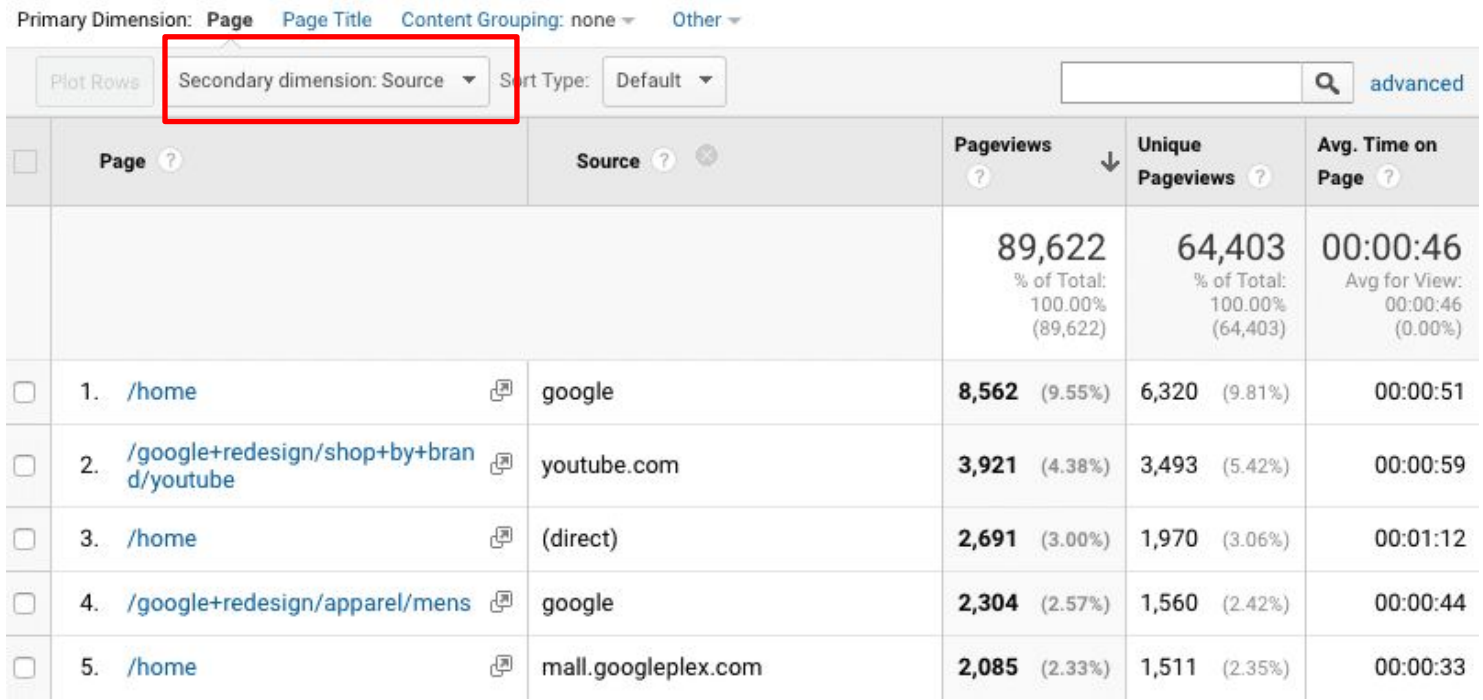
Tables in Google Analytics are selected as the default way to display data and is easy to understand at first glance.
The first column displays a dimension and the rest shows the metrics.



- # Tabular Reports

Adding a secondary dimension to the report.

Example: Add ‘source’ to see traffic source as shown below:

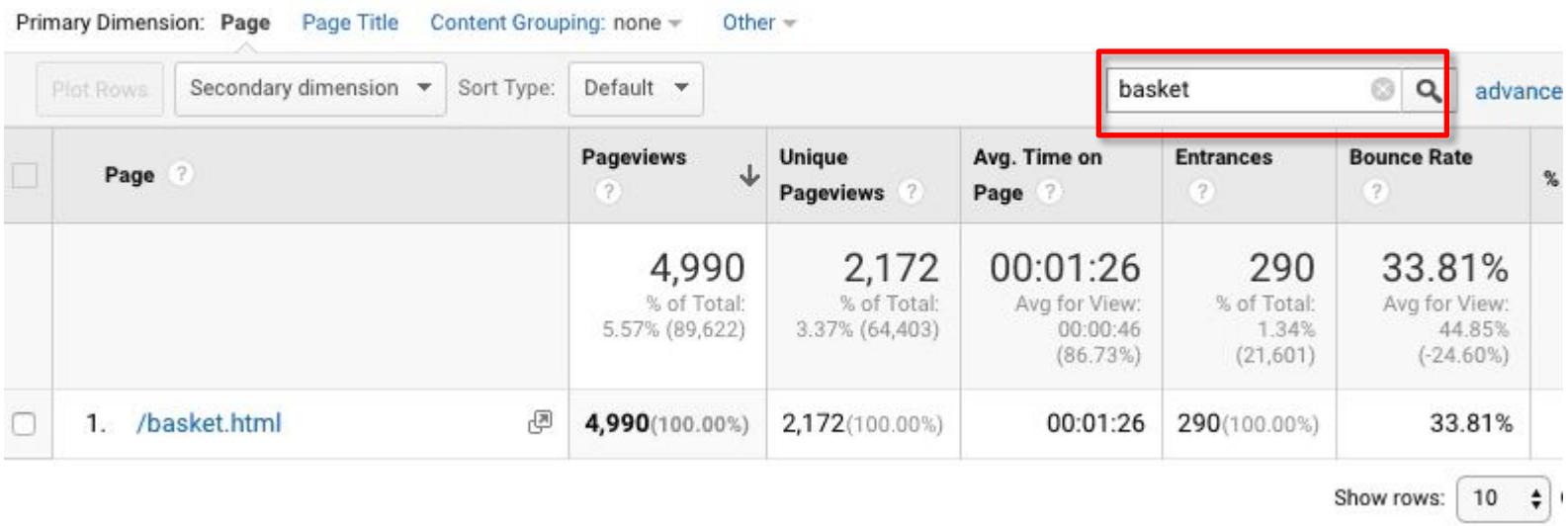


	Page	Source	Pageviews	Unique Pageviews	Avg. Time on Page
			89,622 % of Total: 100.00% (89,622)	64,403 % of Total: 100.00% (64,403)	00:00:46 Avg for View: 00:00:46 (0.00%)
1.	/home	google	8,562 (9.55%)	6,320 (9.81%)	00:00:51
2.	/google+redesign/shop+by+brand/youtube	youtube.com	3,921 (4.38%)	3,493 (5.42%)	00:00:59
3.	/home	(direct)	2,691 (3.00%)	1,970 (3.06%)	00:01:12
4.	/google+redesign/apparel/mens	google	2,304 (2.57%)	1,560 (2.42%)	00:00:44
5.	/home	mall.googleplex.com	2,085 (2.33%)	1,511 (2.35%)	00:00:33

- Tabular Reports

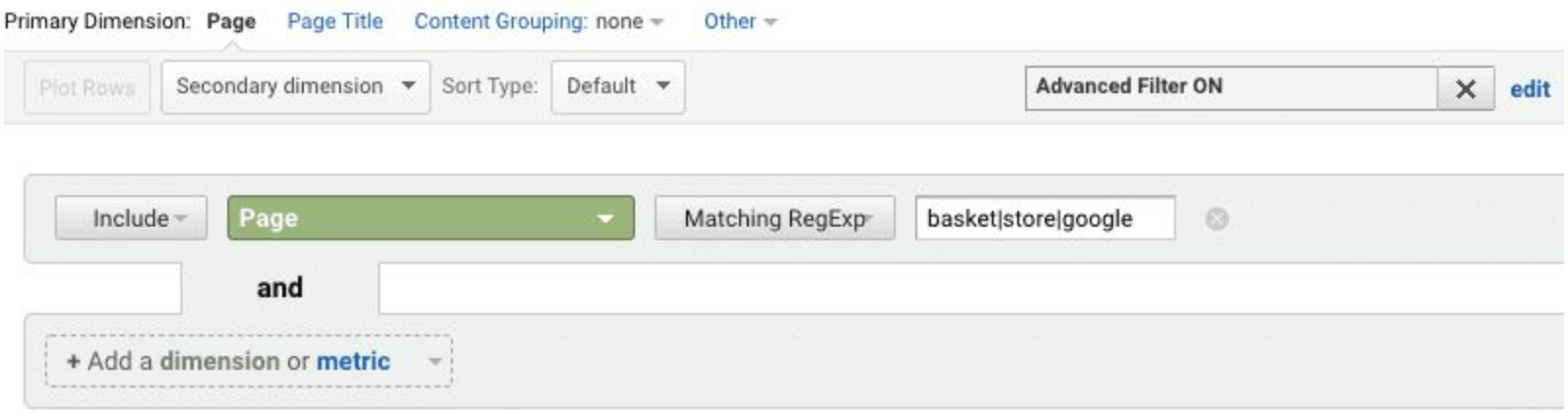
Viewing specific page from the site (Filtering)

Example: Add ‘basket’ keyword from the URL and place it into search box as shown below:



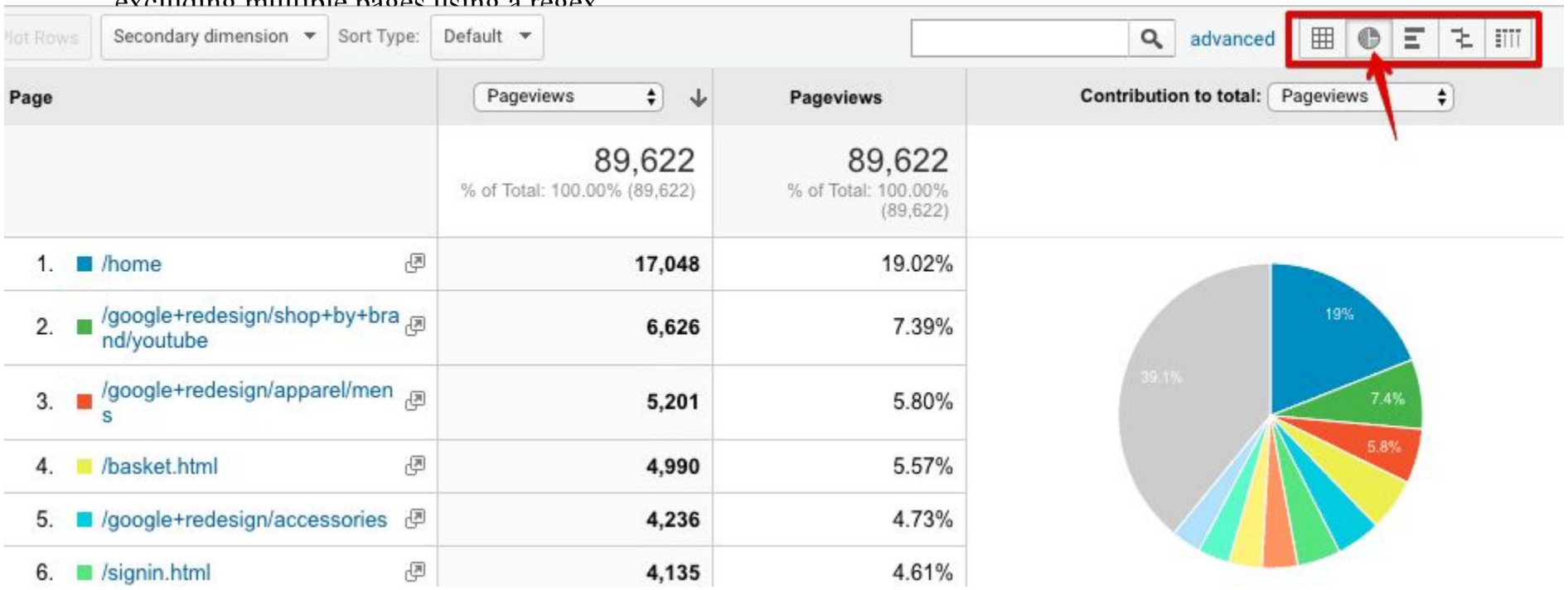
- # Tabular Reports

Using a regex expression to filter multiple pages, as well as, applying an advanced filter for including or excluding multiple pages using a regex.



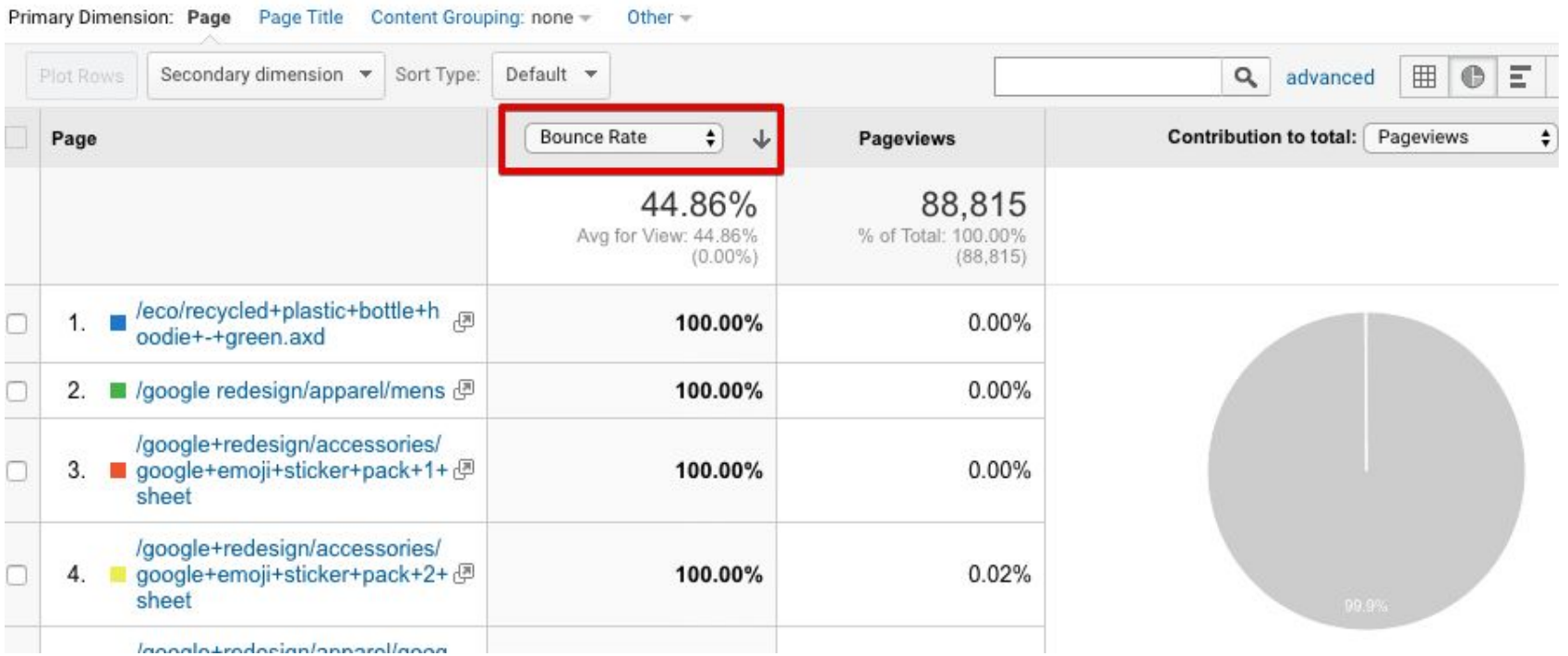
- # Pie Charts

Using a regex expression to filter multiple pages, as well as, applying an advanced filter for including or excluding multiple pages using a regex



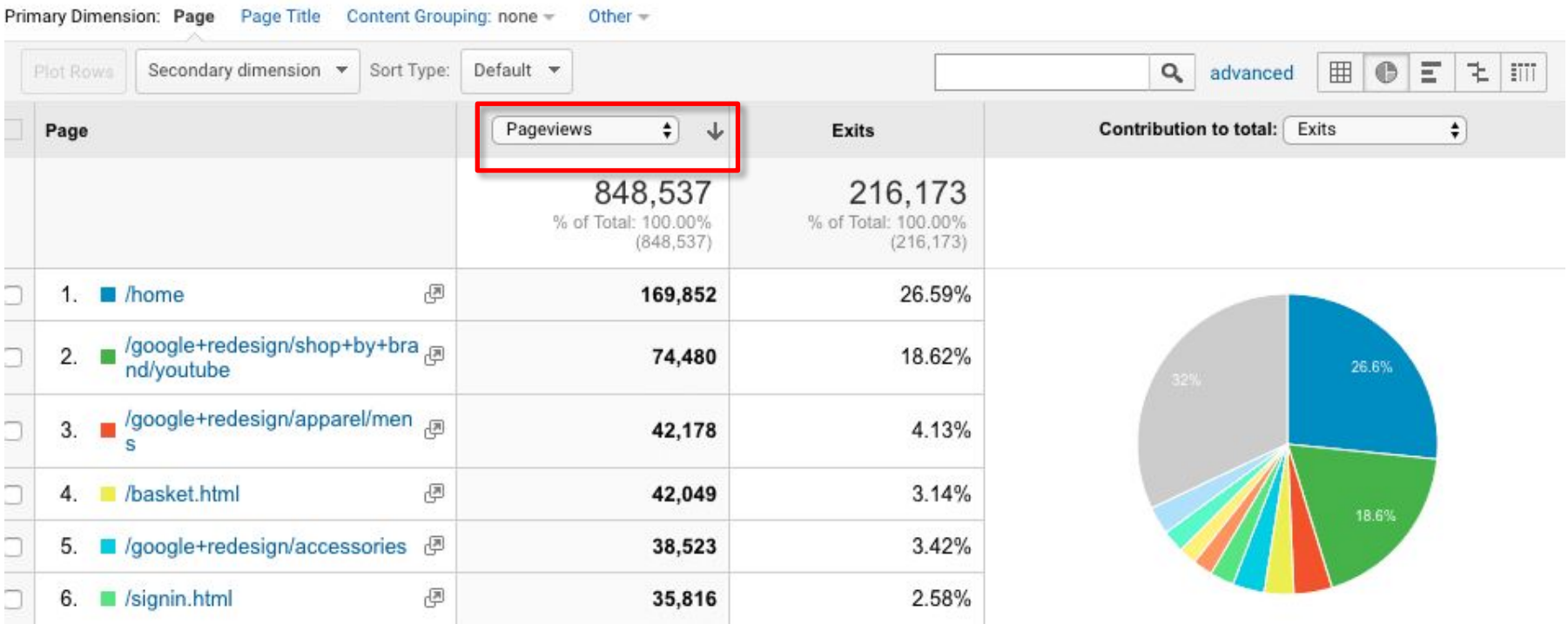
- Pie Charts

The pie chart report only consists of a single metric which makes it easier to consume and digest.
For example, To see the bounce rate for each page on website, select the option from the drop-down on the pie type report, and the resulting pie chart will show you the bounce rate for all pages respectively in a single display.



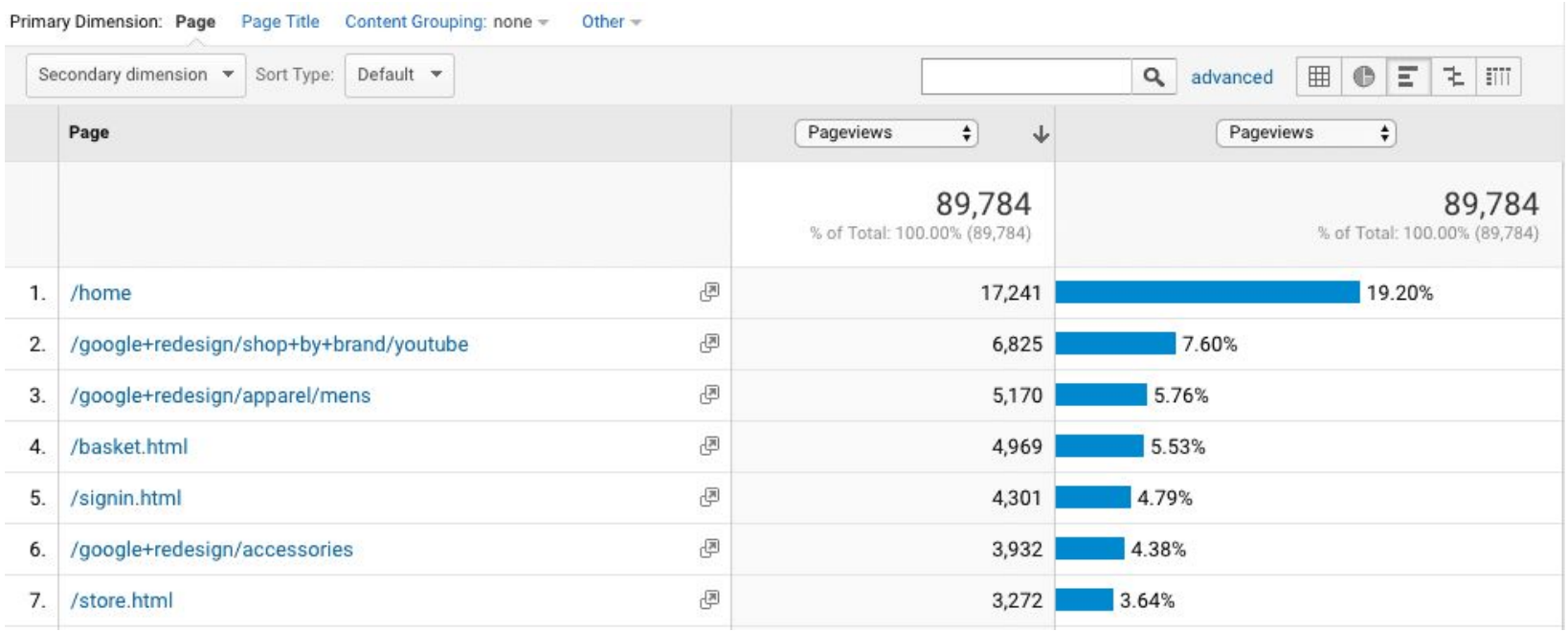
- Pie Charts

To see which page resulted in a high exit rate



- Performance

The performance report displays the total percentage of pageviews for each URL. The first column shows the total number and the second column shows the percentage contribution of a page to the total.



- Comparison

How bounce rate is changing for each specific page, this report will help. The image below shows how the bounce rate is changing for pages with most pageviews.

Page	Pageviews	Bounce Rate (compared to site average)
	89,784 % of Total: 100.00% (89,784)	44.93% Avg for View: 44.93% (0.00%)
/home	17,241	-10.56%
/google+redesign/shop+by+brand/youtube	6,825	30.62%
/google+redesign/apparel/mens	5,170	29.05%
/basket.html	4,969	-21.21%
/signin.html	4,301	-54.37%
/google+redesign/accessories	3,932	5.64%
/store.html	3,272	29.77%
/google+redesign/nest/nest-usa	3,134	-35.43%
/google+redesign/bags	2,968	-35.32%

- Pivot Table

Primary Dimension: **Page** Page Title Content Grouping: none Other

Secondary dimension | Sort Type: Default | advanced

Pivot by: Page | Pivot metrics: Pageviews | Bounce Rate | Columns: 1 - 5 of 274

		Total		1. /home		2. /google+redesign/shop+by+brand/youtube		3. /google+redesign/apparel/mens		4. /basket.html		5. /signin.html	
	Page	Pageviews	Bounce Rate	Pageviews	Bounce Rate	Pageviews	Bounce Rate	Pageviews	Bounce Rate	Pageviews	Bounce Rate	Pageviews	Bounce Rate
1.	/home	**17,663**	41.20%	17,663	41.20%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
2.	/google+redesign/shop+by+brand/youtube	**6,898**	59.06%	0	0.00%	6,898	59.06%	0	0.00%	0	0.00%	0	0.00%
3.	/google+redesign/apparel/mens	**5,210**	57.05%	0	0.00%	0	0.00%	5,210	57.05%	0	0.00%	0	0.00%
4.	/basket.html	**4,904**	36.18%	0	0.00%	0	0.00%	0	0.00%	4,904	36.18%	0	0.00%

- **Pivot Table** To see the performance of your online campaigns by each country.

Primary Dimension: Page | Page Title | Content Grouping: none | Campaign

Secondary dimension | Sort Type: Default | advanced

Pivot by: Country | Pivot metrics: Unique Pageviews | Bounce Rate | Columns: 1 - 5 of 160

		Total		1. United States		2. India		3. Canada		4. United Kingdom		5. Japan	
	Campaign	Unique Pageviews	Bounce Rate	Unique Pageviews	Bounce Rate	Unique Pageviews	Bounce Rate	Unique Pageviews	Bounce Rate	Unique Pageviews	Bounce Rate	Unique Pageviews	Bounce Rate
1.	(not set)	**63,401**	45.05%	39,023	32.22%	2,980	53.90%	2,154	38.21%	1,733	60.57%	1,090	61.55%
2.	Data Share Promo	**1,302**	55.20%	244	61.86%	105	52.94%	52	42.11%	123	31.71%	47	61.54%
3.	Smart Display Campaign	**113**	77.89%	17	46.15%	23	54.55%	0	0.00%	1	100.00%	0	0.00%
4.	AW - Accessories	**70**	33.33%	69	28.57%	0	0.00%	1	100.00%	0	0.00%	0	0.00%
5.	AW - Dynamic Search Ads Whole Site	**41**	33.33%	41	33.33%	0	0.00%	0	0.00%	0	0.00%	0	0.00%

- In marketing, we have the concept of a purchase funnel. There are different stages within the funnel
- that describe customer interactions. A basic purchase funnel includes the following steps:
- **Acquisition** involves building awareness and acquiring user interest
- **Behavior** is when users engage with your business
- **Conversion** is when a user becomes a customer and transacts with your business

- We can track what online behavior led to purchases and use that data to make informed decisions about how to reach new and existing customers.

- Different kinds of businesses can benefit from digital analytics:

✔ **Publishers** can use it to create a loyal, highly-engaged audience and to better align on-site advertising with user interests.

✔ **Ecommerce** businesses can use digital analytics to understand customers' online purchasing behavior and better market their products and services.

✔ **Lead generation** sites can collect user information for sales teams to connect with potential leads.

- **How Google Analytics work?**

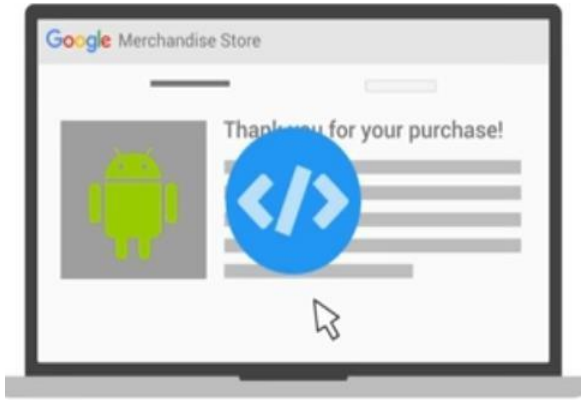
- Tracking a Website

  - To track a website, you first have to create a Google Analytics account. Then you need to add a small piece of **Javascript tracking code** to each page on your site.
  - Every time a user visits a web page, the tracking code will collect anonymous information about how that user interacted with the page.
  - The tracking code could show how many users visited a page or how many users bought an item by tracking whether they made it to the purchase confirmation page.

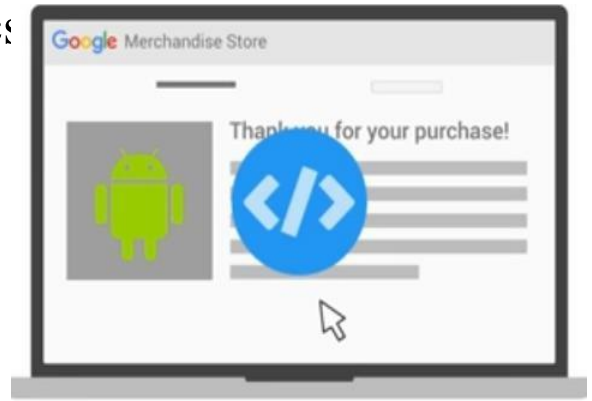
- **Javascript Tracking Code**
- Tracking a Website

  - The tracking code will also collect information from the browser like:

  - **Language**: the browser is set to.
  - **Type of Browser**: like chrome, explorer, etc.
  - **Device**
  - **Operating System**
  - **Traffic Source**: what brought users to the site in the first place This might be a search engine, an advertisement they clicked on, or an email marketing campaign.

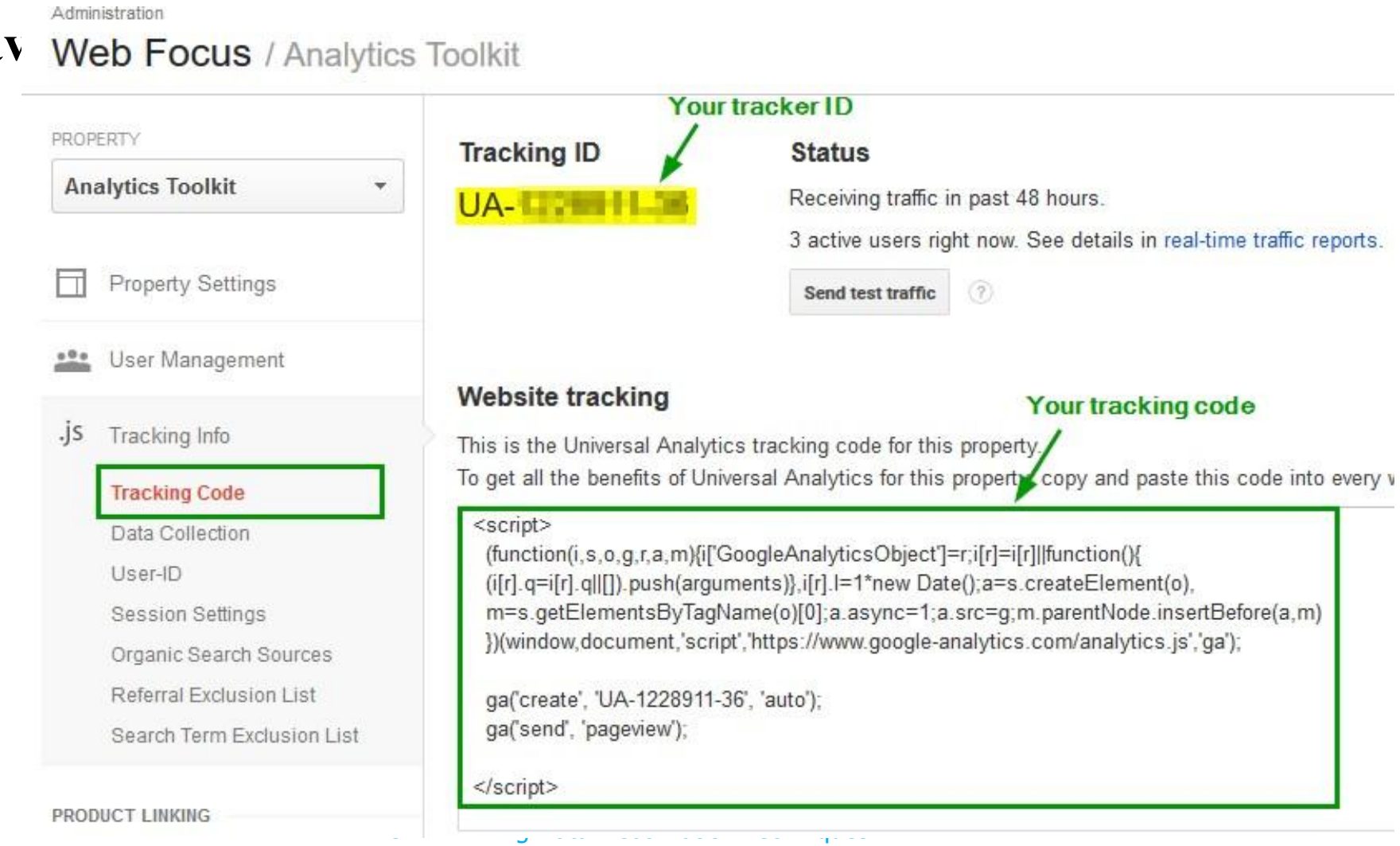


- **Javascript Tracking Code**
- Establish a User ID and password at google.com/analytics
- To find the tracking ID and code snippet:
  - Sign in to your Analytics account.
  - Click Admin.
  - Select an account from the menu in the ACCOUNT column.
  - Select a property from the menu in the PROPERTY column.
  - Under PROPERTY, click Tracking Info > Tracking Code.



- Once you have successfully installed the Analytics tracking code, it can take up to 24 hours for data such as traffic-referral information, user characteristics, and browsing information to appear in your reports.

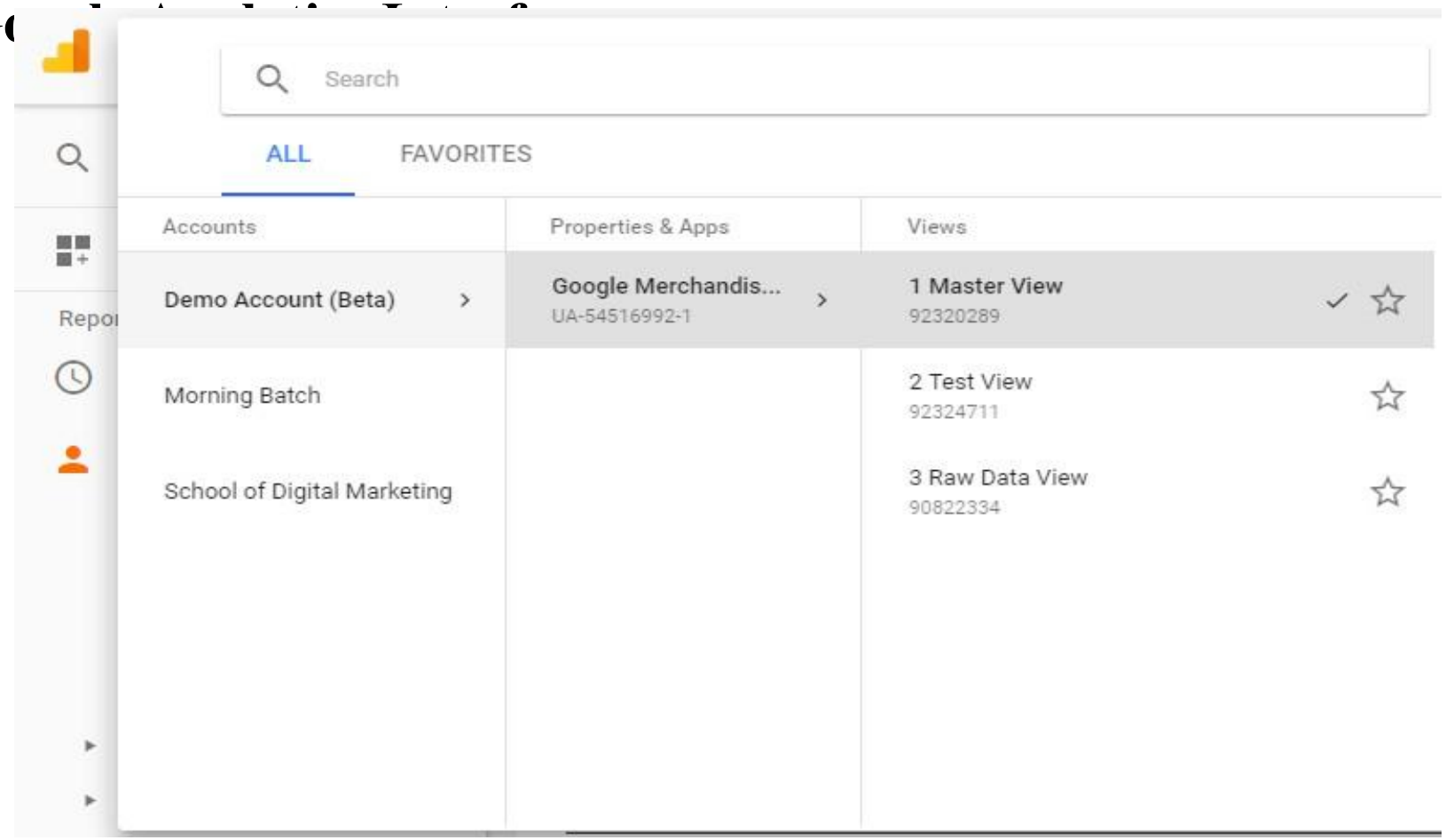
- **Jav**


Administration
Web Focus / Analytics Toolkit
PROPERTY
Analytics Toolkit
Property Settings
User Management
.js Tracking Info
Tracking Code
Data Collection
User-ID
Session Settings
Organic Search Sources
Referral Exclusion List
Search Term Exclusion List
PRODUCT LINKING
Your tracker ID
Tracking ID
UA-
Status
Receiving traffic in past 48 hours.
3 active users right now. See details in real-time traffic reports.
Send test traffic
Website tracking
Your tracking code
This is the Universal Analytics tracking code for this property.
To get all the benefits of Universal Analytics for this property, copy and paste this code into every
<script>
(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){
(i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o),
m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m)
})(window,document,'script','https://www.google-analytics.com/analytics.js','ga');
ga('create', 'UA-1228911-36', 'auto');
ga('send', 'pageview');
</script>

- **Processing and Reporting**
- When the tracking code collects data, it packages that information up and sends it to Google Analytics to be processed into reports.
- When Analytics processes data, it aggregates and organizes the data based on particular criteria like whether a user's device is mobile or desktop, or which browser they're using.
- Once Analytics processes the data, it's stored in a database where it can't be changed.

- **Google Analytics Setup**

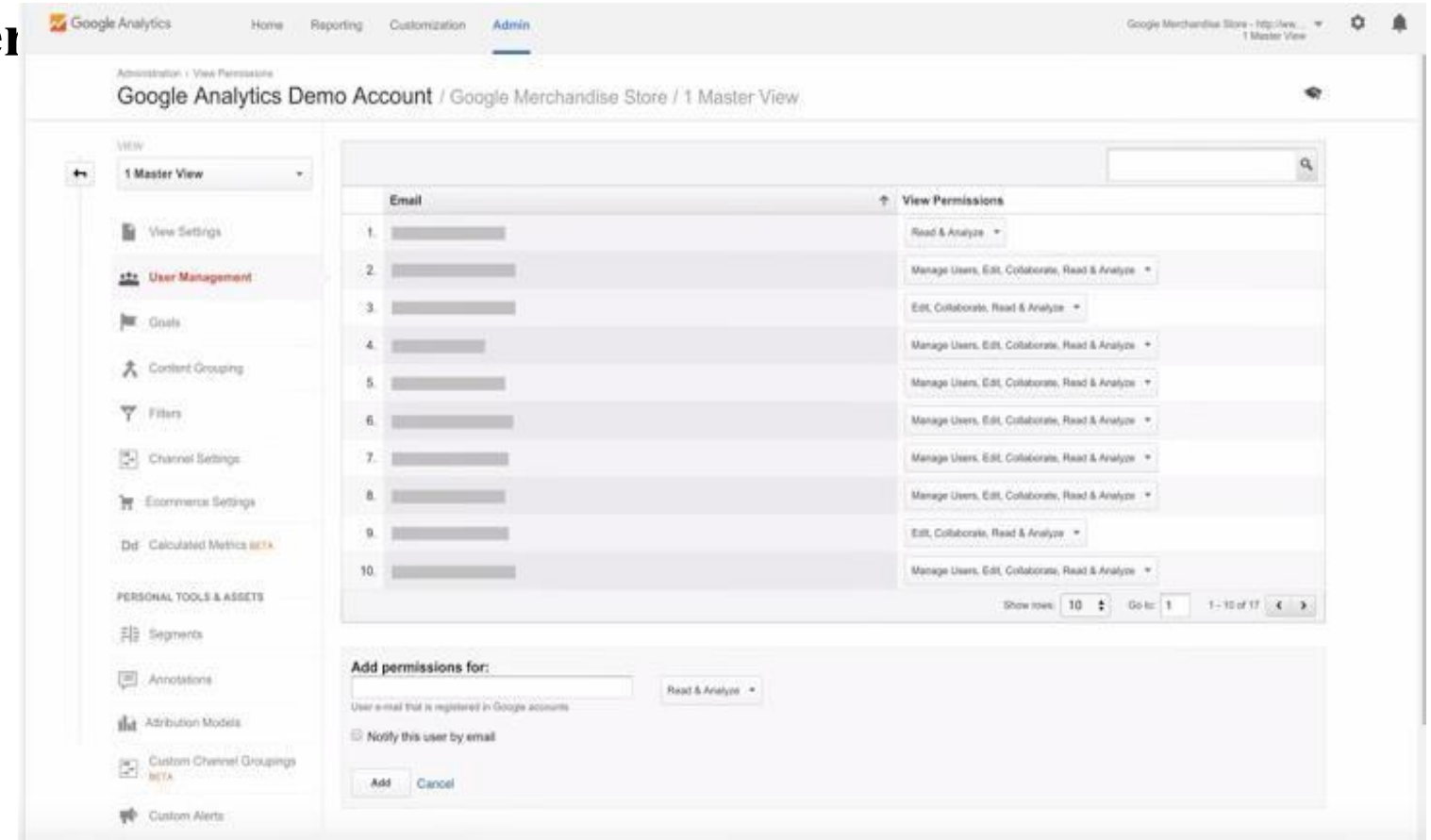
- Go


Search
ALL
FAVORITES
Accounts
Properties & Apps
Views
Demo Account (Beta)
Google Merchandis...
UA-54516992-1
1 Master View
92320289
Morning Batch
2 Test View
92324711
School of Digital Marketing
3 Raw Data View
90822334

- **Goals**

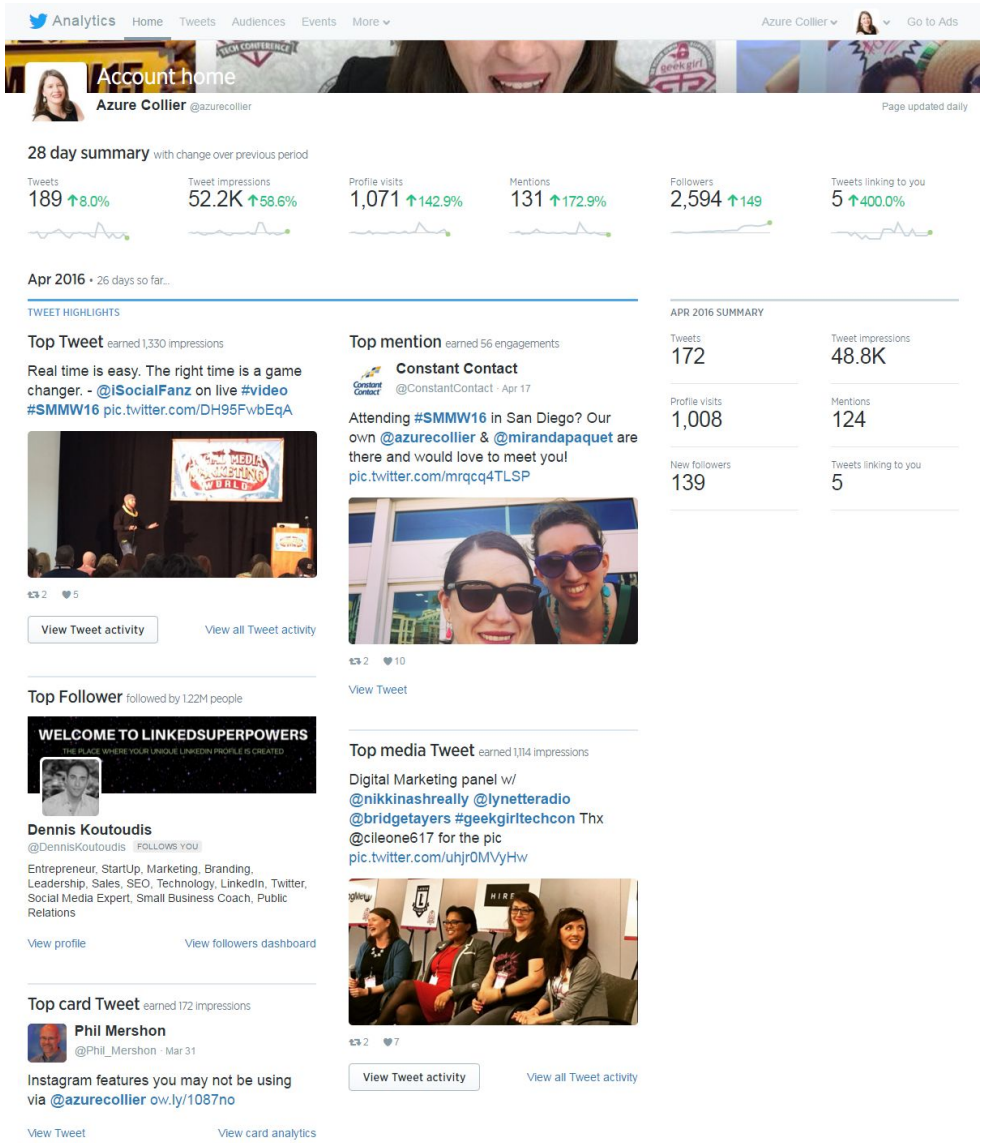
- At the View level, you can set Google Analytics “Goals.”
- Goals are a simple way to track conversions (or business objectives) from your website.
- For example: A goal could be how many users signed up for an email newsletter, or how many users purchased a product.

- **Per**

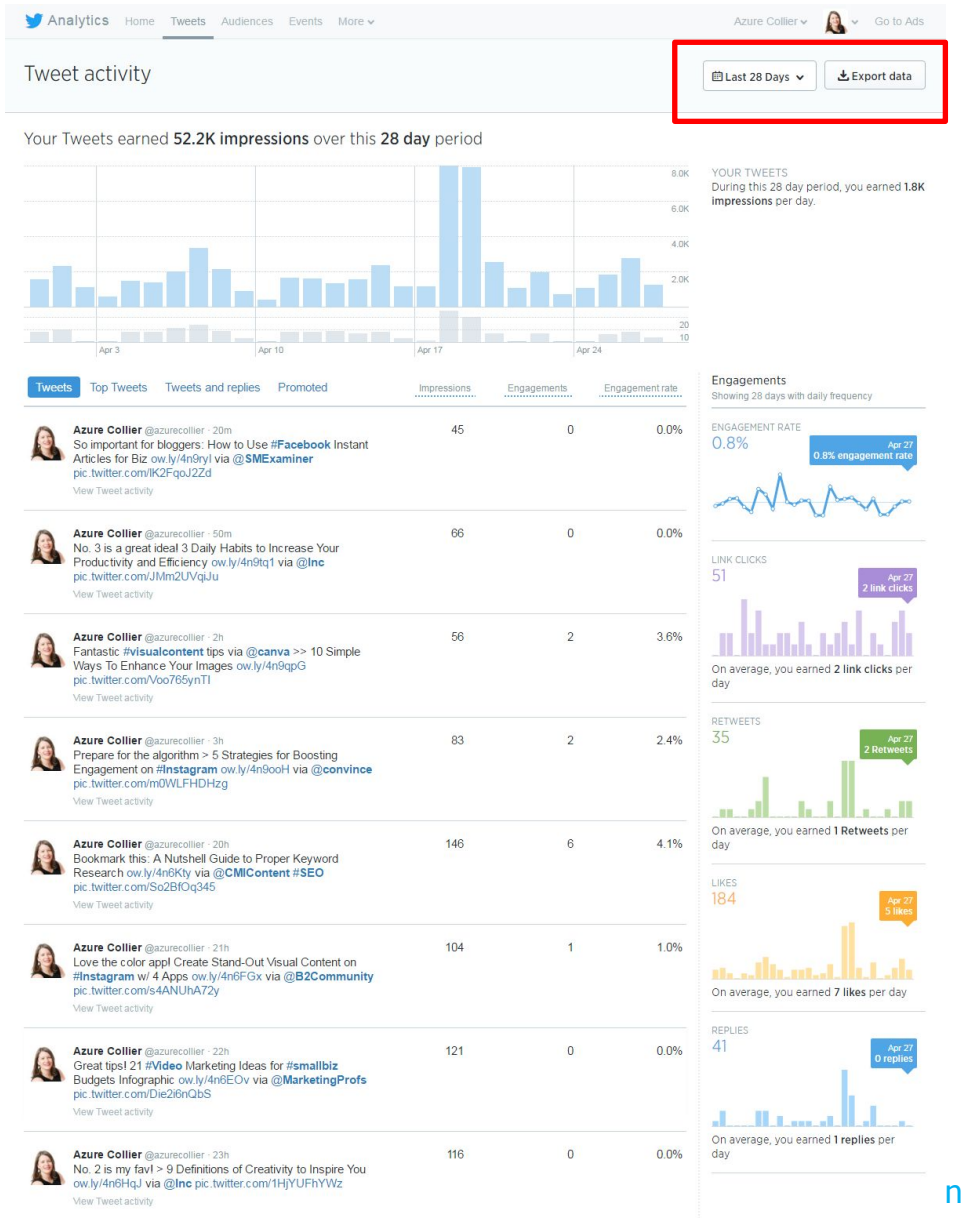

Google Analytics
Home
Reporting
Customization
Admin
Google Analytics Demo Account / Google Merchandise Store / 1 Master View
1 Master View
View Settings
User Management
Goals
Content Grouping
Filters
Channel Settings
Ecommerce Settings
Calculated Metrics
PERSONAL TOOLS & ASSETS
Segments
Annotations
Attribution Models
Custom Channel Groupings
Custom Alerts
Email
View Permissions
Add permissions for:
Notify this user by email
Add
Cancel

- Twitter is now the third most popular social network, behind Facebook and MySpace.
- Twitter has over 100 million active users and about 50 million of them log in every day.
- In Several opinion Twitter is not a much of social network and it is rather a place for marketers.

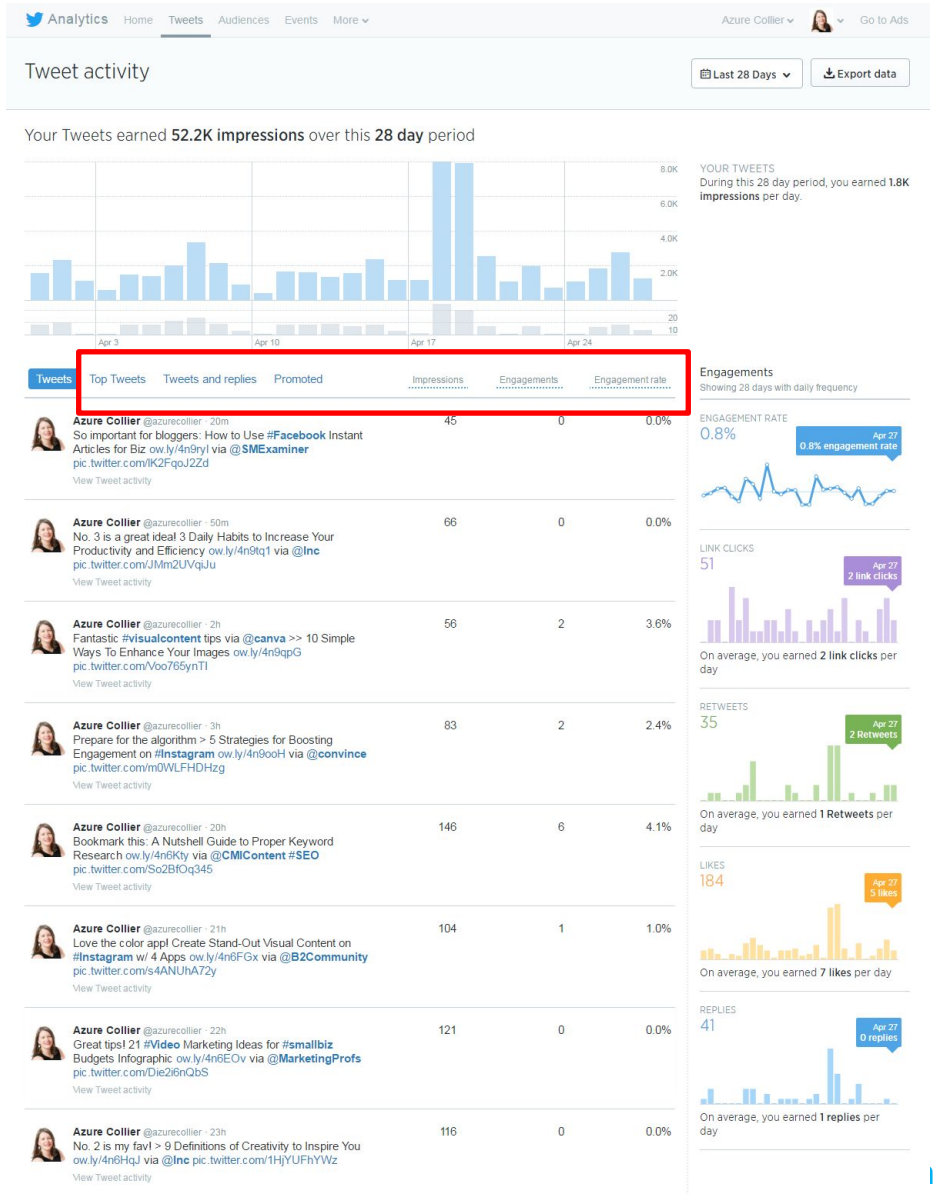
- Twitter Analytics Tools:
  - Twitter Analytics Tools are one of the best way for measuring your online presence on twitter.
  - Twitter Analytics Tools help website owners understand how much traffic they receive from Twitter and the effectiveness of Twitter integration on their sites.
  - Twitter tools are designed to add value by presenting a different way to visualize or analyze tweets, the people in the network, and the tweets from the people in users' network.



highlights for each month which includes your top tweet, mention, follower, media tweet and card tweet, as well as summaries.



Twitter displays analytics for your tweets over the last 28 days
but you can select a different time period by clicking the Last 28 Days button on the top right side of the page.



In the graph, you can click through a menu of your tweets arranged by four categories:

- **Tweets** – your tweets in reverse-chronological order
- **Top tweets** – the tweets that got the most impressions
- **Tweets and replies** – your tweets and replies by other Twitter users
- **Promoted** – any promoted tweets that you published

- Twitter Analytics Metrics:
  - **Impressions** – the number of times a user is served a Tweet in timeline or search results
  - **Engagements** – the total number of times a user has interacted with a tweet, includes clicks, retweets, replies, follows, likes, links, Twitter cards, hashtags, embedded media, Twitter username, profile photo or expanding the tweet
  - **Engagement rate** – the number of engagements divided by the number of impressions

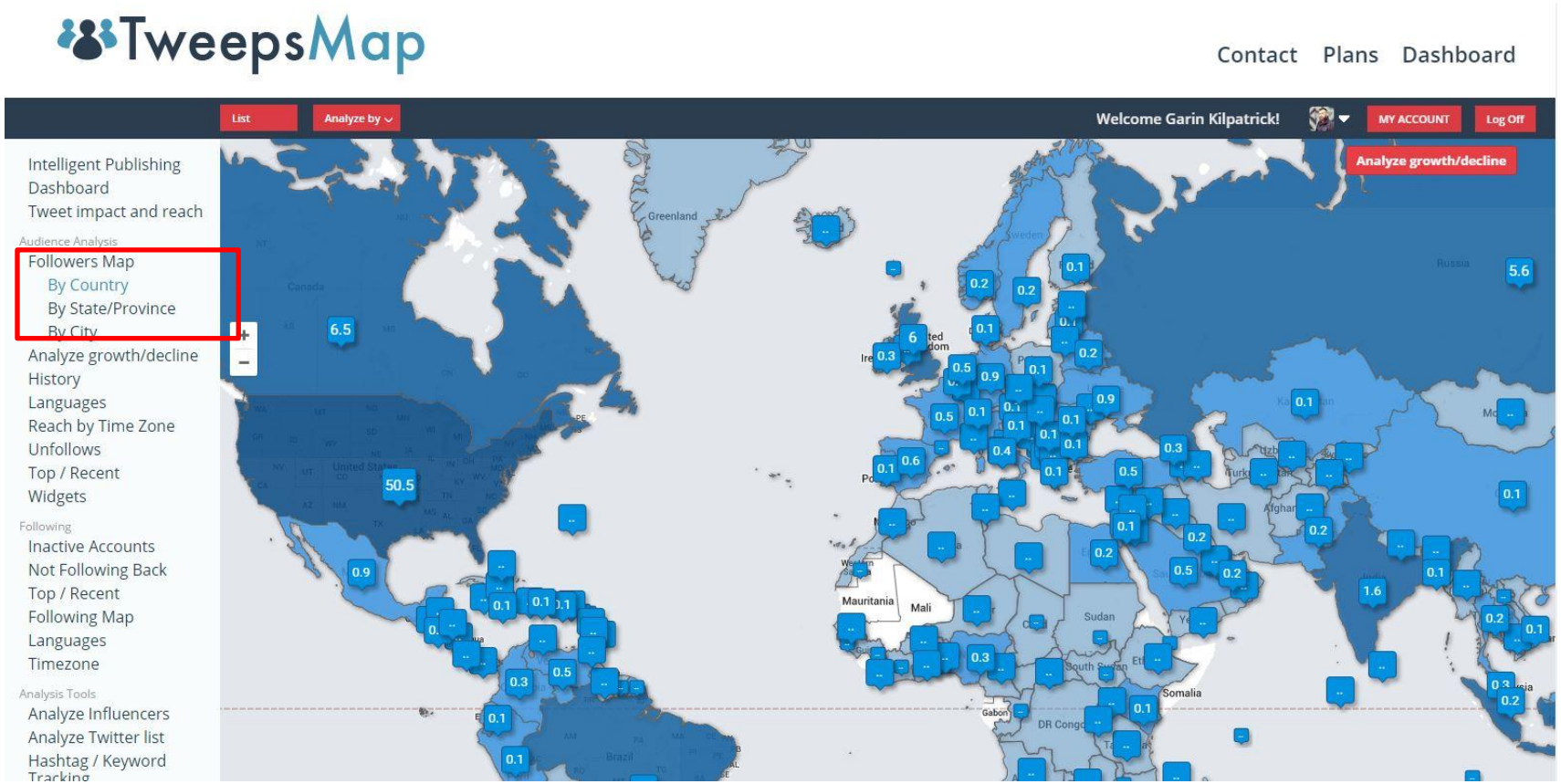
- 7 insights you can get from Twitter Analytics:
  - Tweet Impressions
    - What did you do differently in a month with higher impressions?
    - Did you Tweet more frequently?
    - cumulative overview to compare monthly activity
  - Tweet engagements and engagement rate
    - If your tweets are getting more or little tweets
    - Engagement Rate- Engagements/Impressions
  - Top Tweets
    - aggregate the learnings and see what they have in common.
    - Are they all adopting the same brand voice?
    - Do they all have an emoji in them?

- 7 insights you can get from Twitter Analytics:
  - Follower Growth
    - Did your followers are increasing or decreasing
    - How many new followers you received
    - On what day you had many followers
  - Profile visits
    - No of visit to your profile
    - Graph includes comparison w.r.t. 28days
  - Mentions
    - View @mentions
  - Video content performance
    - How people are responding to your videos.

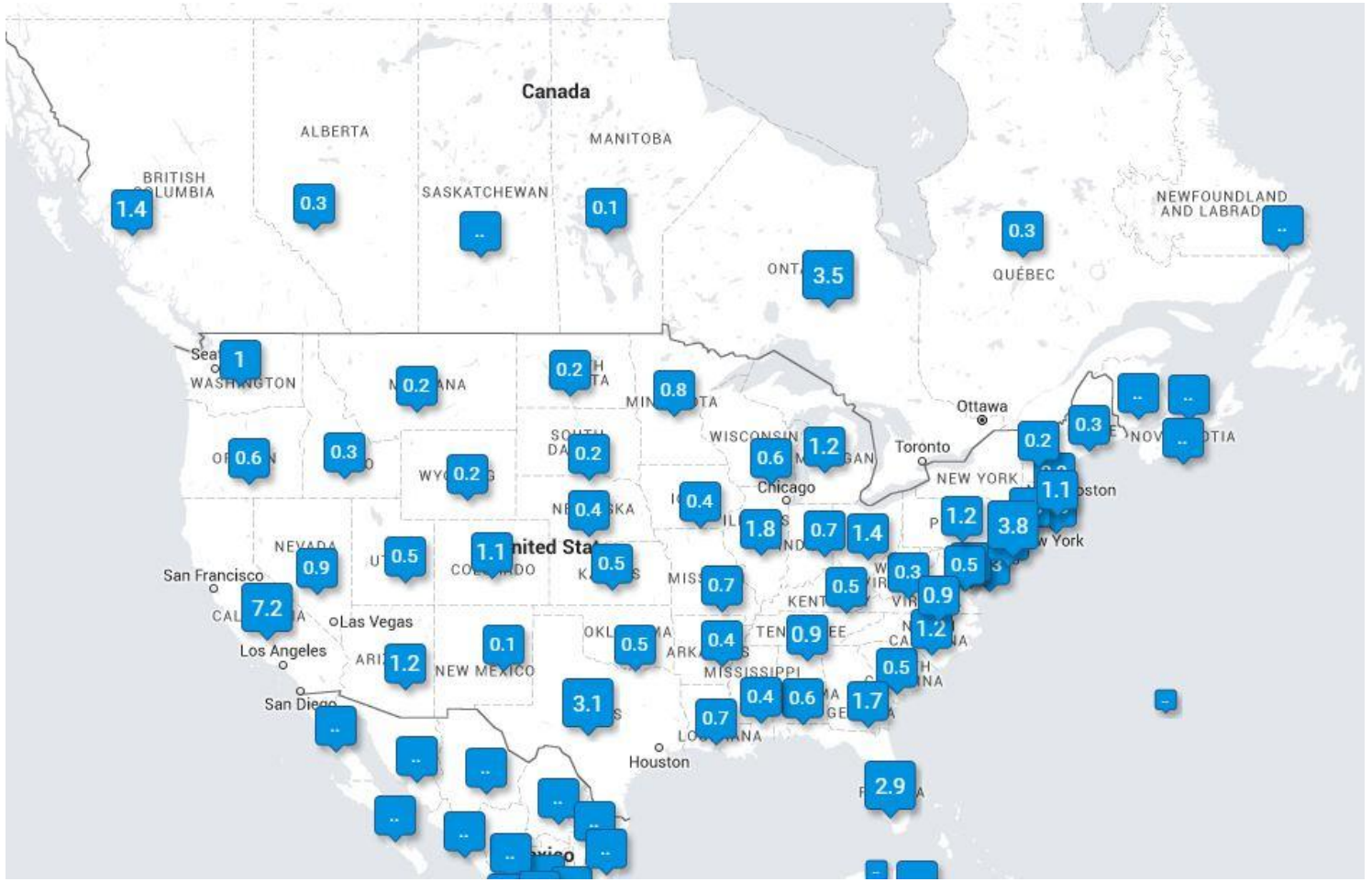
- Few Twitter tools listed down here as:
  - TweepsMap
  - Audiense
  - Keyhole
  - Twitter Counter
  - Twenty Feet

- TweepsMap
  - It is a Twitter tool for analyzing and visualizing your Twitter network
  - Useful in showing how your followers are distributed on a map, in terms of percentages.

- TweepsMap

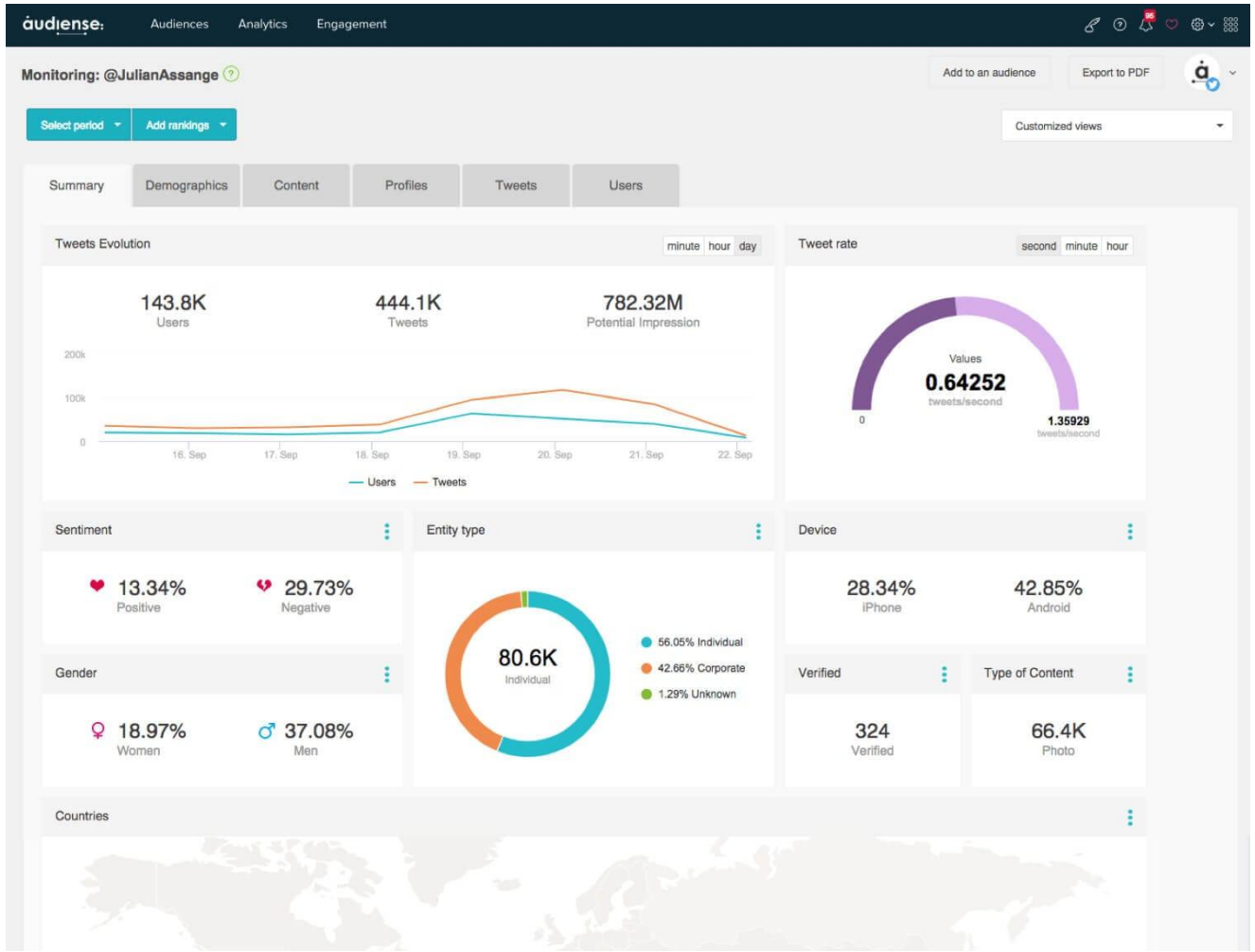

TweepsMap
Contact
Plans
Dashboard
List
Analyze by
Welcome Garin Kilpatrick!
MY ACCOUNT
Log Off
Analyze growth/decline
Intelligent Publishing
Dashboard
Tweet impact and reach
Audience Analysis
Followers Map
By Country
By State/Province
By City
Analyze growth/decline
History
Languages
Reach by Time Zone
Unfollows
Top / Recent
Widgets
Following
Inactive Accounts
Not Following Back
Top / Recent
Following Map
Languages
Timezone
Analysis Tools
Analyze Influencers
Analyze Twitter list
Hashtag / Keyword
Greenland
Canada
Russia
Mauritania
Mali
Sudan
Somalia
Brazil

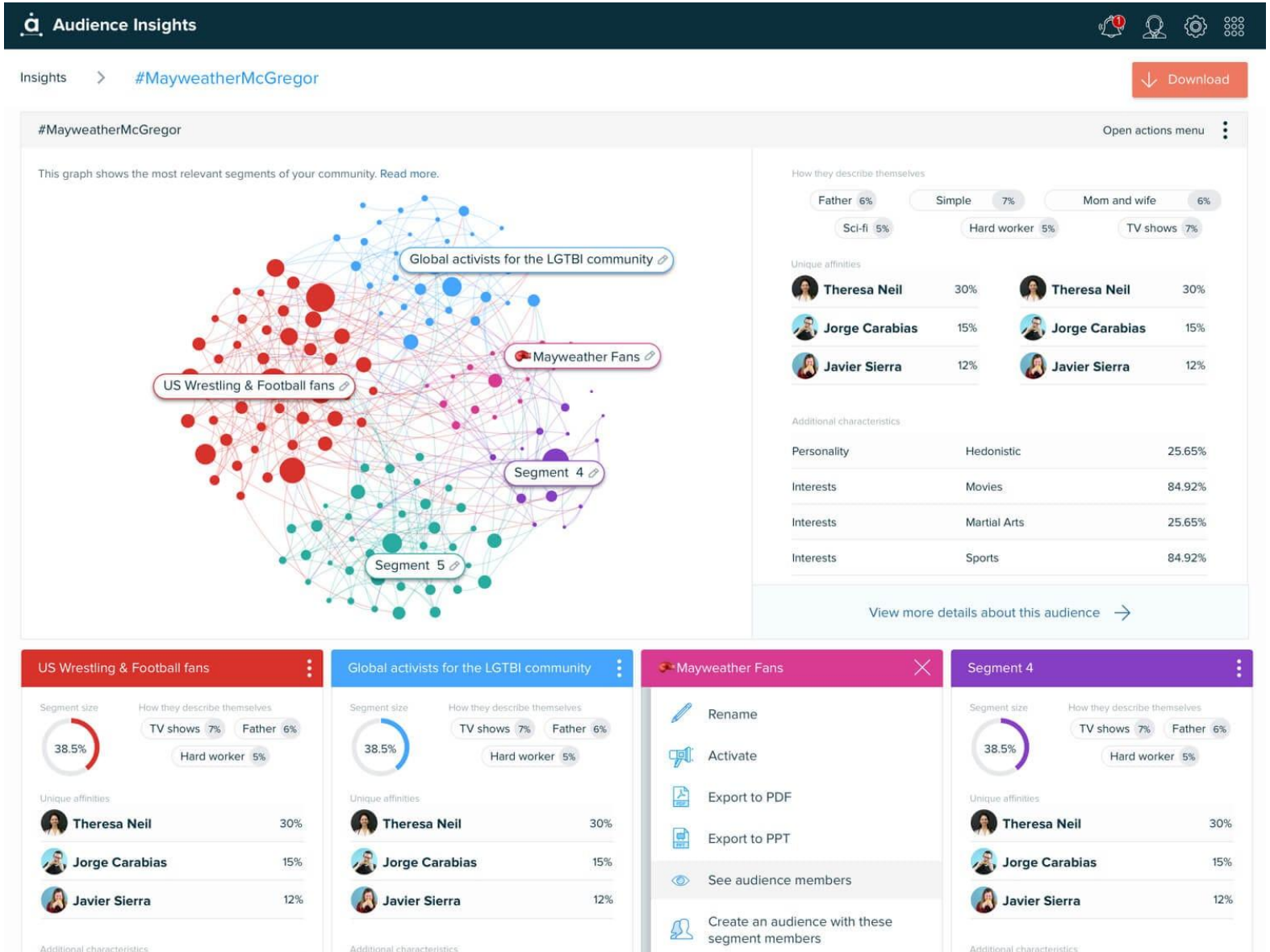
- This data can be viewed at state level or even city level



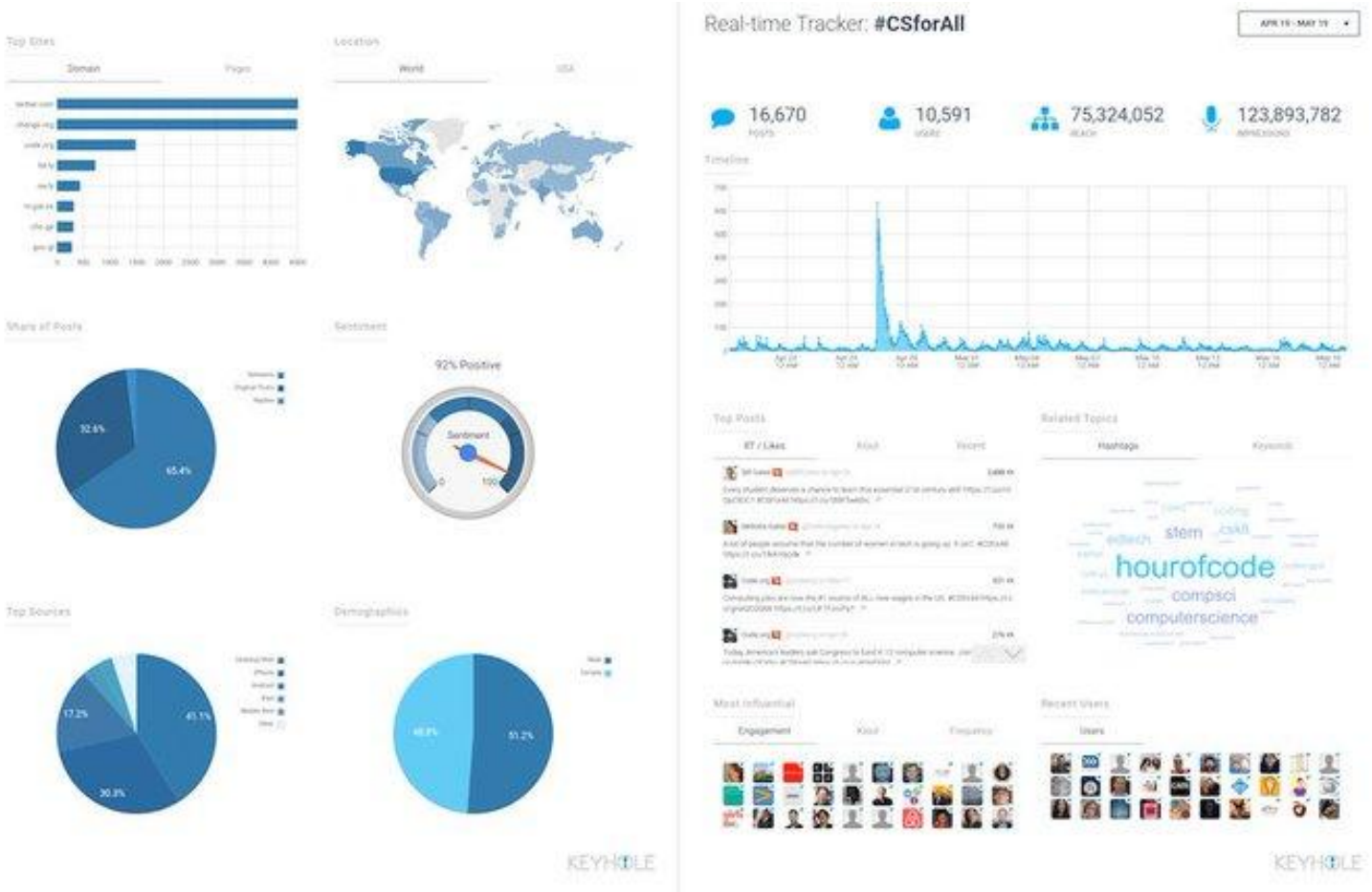
- Audiense has some of the features as:
  - Explore Your Twitter Community
  - Determine the Best Time to Tweet
  - Identify Influencers
  - Target Your Audience with Precision
  - Discover and Easily Follow Targeted Twitter Users

- Audiense:


audiense
Audiences
Analytics
Engagement
Monitoring: @JulianAssange
Add to an audience
Export to PDF
Select period
Add rankings
Customized views
Summary
Demographics
Content
Profiles
Tweets
Users
Tweets Evolution
minute hour day
143.8K
Users
444.1K
Tweets
782.32M
Potential Impression
200k
100k
0
16. Sep
17. Sep
18. Sep
19. Sep
20. Sep
21. Sep
22. Sep
Users
Tweets
Tweet rate
second minute hour
Values
0.64252
tweets/second
0
1.35929
tweets/second
Sentiment
13.34%
Positive
29.73%
Negative
Entity type
80.6K
Individual
56.05% Individual
42.66% Corporate
1.29% Unknown
Device
28.34%
iPhone
42.85%
Android
Gender
18.97%
Women
37.08%
Men
Verified
324
Verified
Type of Content
66.4K
Photo
Countries

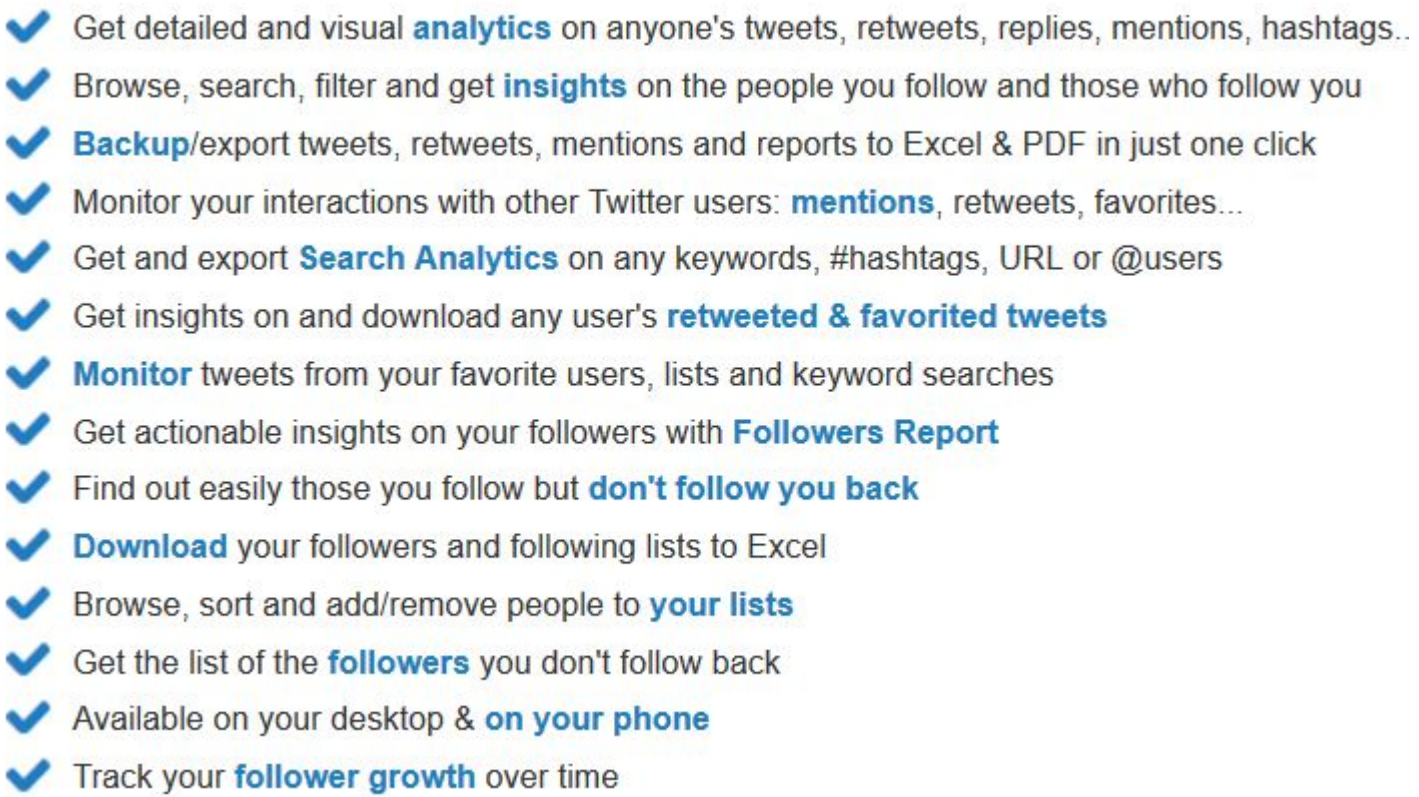

Audience Insights
Insights
#MayweatherMcGregor
Download
#MayweatherMcGregor
Open actions menu
This graph shows the most relevant segments of your community. Read more.
Global activists for the LGTBI community
Mayweather Fans
US Wrestling & Football fans
Segment 4
Segment 5
How they describe themselves
Father 6%
Simple 7%
Mom and wife 6%
Sci-fi 5%
Hard worker 5%
TV shows 7%
Unique affinities
Theresa Neil 30%
Jorge Carabias 15%
Javier Sierra 12%
Additional characteristics
Personality Hedonistic 25.65%
Interests Movies 84.92%
Interests Martial Arts 25.65%
Interests Sports 84.92%
View more details about this audience
US Wrestling & Football fans
Global activists for the LGTBI community
Mayweather Fans
Segment 4
Segment size
38.5%
Rename
Activate
Export to PDF
Export to PPT
See audience members
Create an audience with these segment members

- Keyhole
  - Keyhole is a Twitter Analytics tool that enables you to tap into Instagram data as well.
  - With Keyhole you can track hashtags, influencers, high impact data, and more.

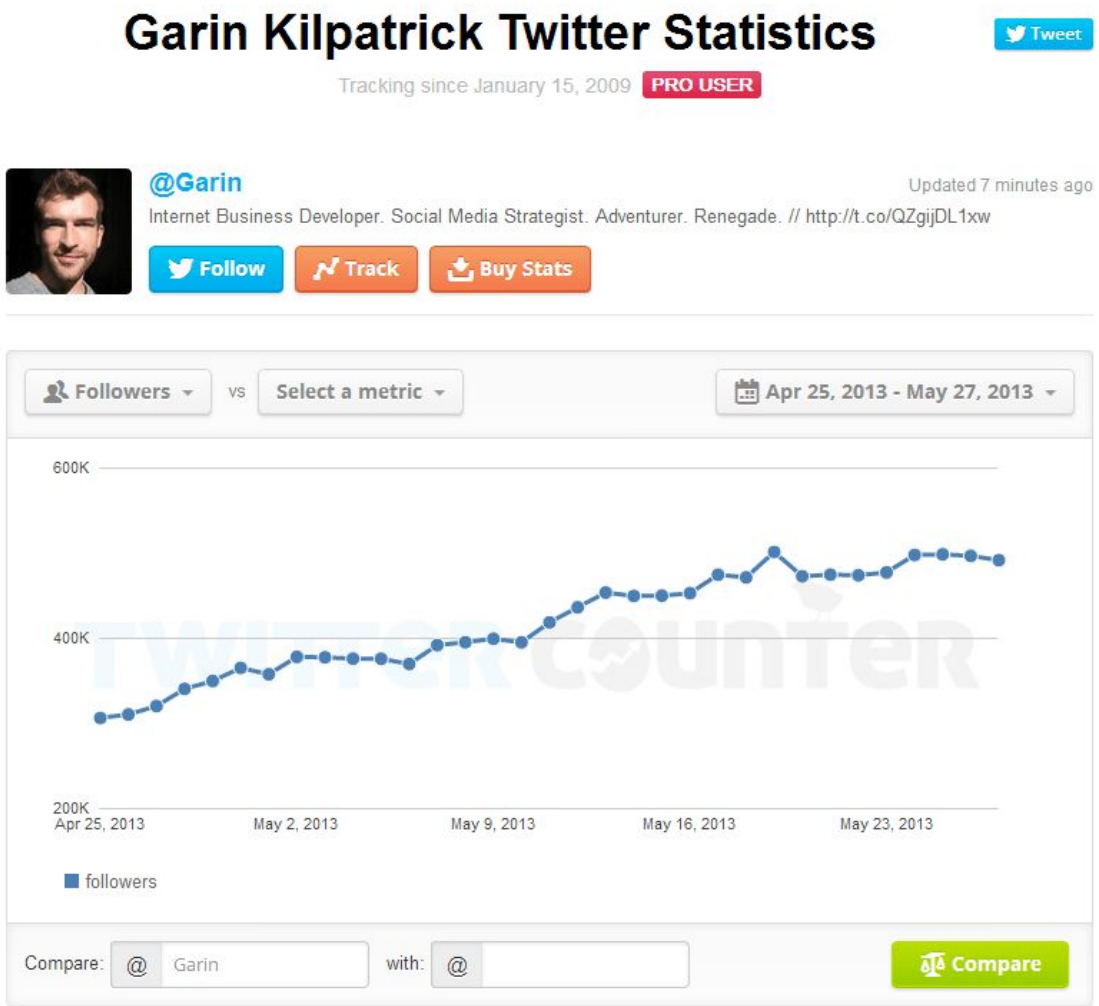

Real-time Tracker: #CSforAll
16,670
10,591
75,324,052
123,893,782
92% Positive
Sentiment
hourofcode
compsci
computerscience
stem
KEYHOLE
KEYHOLE

- Twitonomy
  - It is a powerful Twitter analytics platform
  - This free service is actually very robust

- Twitonomy

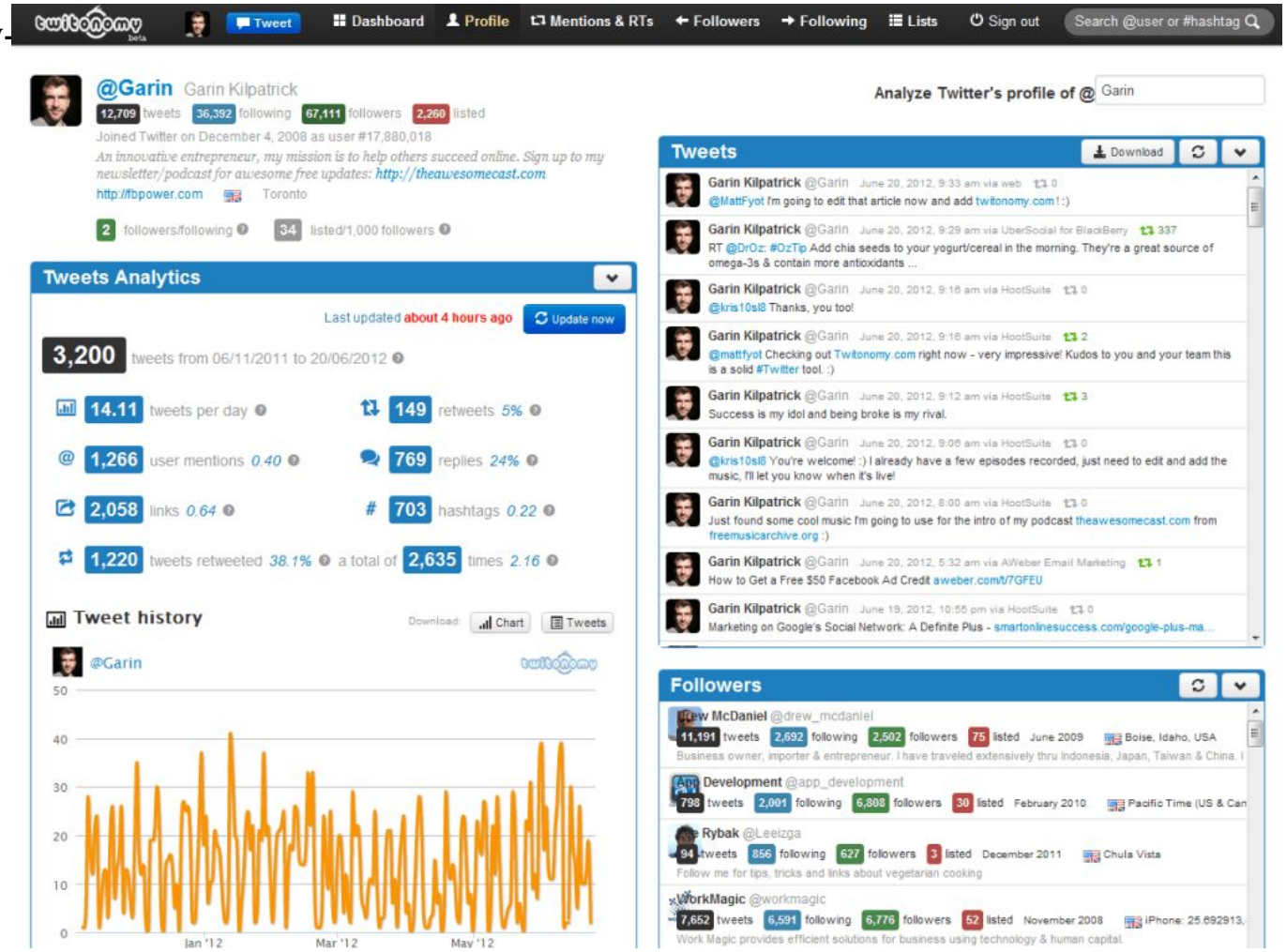

Get detailed and visual analytics on anyone's tweets, retweets, replies, mentions, hashtags..
Browse, search, filter and get insights on the people you follow and those who follow you
Backup/export tweets, retweets, mentions and reports to Excel & PDF in just one click
Monitor your interactions with other Twitter users: mentions, retweets, favorites...
Get and export Search Analytics on any keywords, #hashtags, URL or @users
Get insights on and download any user's retweeted & favorited tweets
Monitor tweets from your favorite users, lists and keyword searches
Get actionable insights on your followers with Followers Report
Find out easily those you follow but don't follow you back
Download your followers and following lists to Excel
Browse, sort and add/remove people to your lists
Get the list of the followers you don't follow back
Available on your desktop & on your phone
Track your follower growth over time

- Twitter Counter
  - It is a way to visualize and track the growth of your own followers, and even compare your growth to the growth of other users

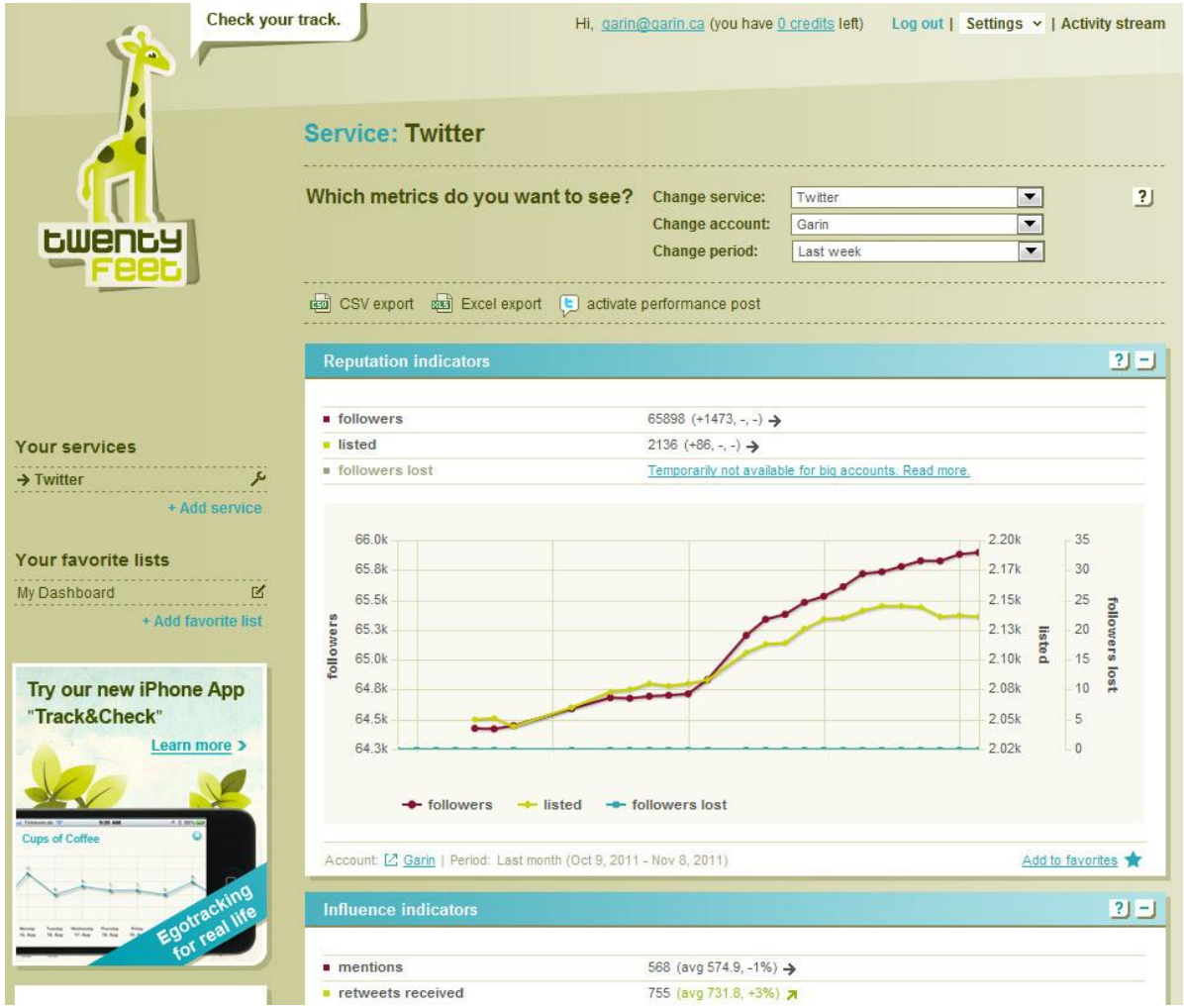


Graph showing follower growth of the user during the last month.

Profile on Twitonomy-



- Twenty Feet
  - Twenty Feet is a powerful analytics platform that tracks and graphs stats like Twitter mentions, followers, retweets, and more.
  - Twenty feet also integrates with other services like Facebook, bitly, Google Analytics, YouTube, and more.


Check your track.
Hi, garin@garin.ca (you have 0 credits left)
Log out | Settings | Activity stream
twenty feet
Service: Twitter
Which metrics do you want to see?
Change service: Twitter
Change account: Garin
Change period: Last week
CSV export
Excel export
activate performance post
Reputation indicators
followers 65898 (+1473, -, -)
listed 2136 (+86, -, -)
followers lost
Temporarily not available for big accounts. Read more.
66.0k
65.8k
65.5k
65.3k
65.0k
64.8k
64.5k
64.3k
followers
2.20k
2.17k
2.15k
2.13k
2.10k
2.08k
2.05k
2.02k
listed
35
30
25
20
15
10
5
0
followers lost
followers
listed
followers lost
Account: Garin | Period: Last month (Oct 9, 2011 - Nov 8, 2011)
Add to favorites
Influence indicators
mentions 568 (avg 574.9, -1%)
retweets received 755 (avg 731.8, +3%)
Your services
Twitter
+ Add service
Your favorite lists
My Dashboard
+ Add favorite list
Try our new iPhone App "Track&Check"
Learn more
Cups of Coffee
Egotracking for real life

References

URLs:

1. https://www.scnsoft.com/blog/big-data-visualization-techniques
2. https://www.klipfolio.com/resources/articles/what-is-data-visualization
3. https://www.import.io/post/9-ways-make-big-data-visual/
4. https://chezvoila.com/blog/parallel/
5. https://datavizcatalogue.com/methods/parallel_coordinates.html
6. https://www.data-to-viz.com/graph/streamgraph.html
7. https://marketlytics.com/blog/google-analytics-data-visualizations/
8. https://twittertoolsbook.com/10-awesome-twitter-analytics-visualization-tools/
9. https://twittertoolsbook.com/10-awesome-twitter-analytics-visualization-tools/
10. https://mumbaiunivercity.academia.edu/MCTA
11. https://business.twitter.com/en/blog/7-useful-insights-Twitter-analytics.html

Technical Papers:

1. Analytical Review of Data Visualization Methods in Application to Big Data, Hindawi Publishing Corporation, Journal of Electrical and Computer Engineering, Volume 2013, Article ID 969458, http://dx.doi.org/10.1155/2013/969458
2. Big Data and Visualization: Methods, Challenges and Technology Progress, Digital Technologies, 2015, Vol. 1, No. 1, 33-38, DOI:10.12691/dt-1-1-7
3. Google Analytics - Case study by Suraj Chande