Data Visualization Techniques

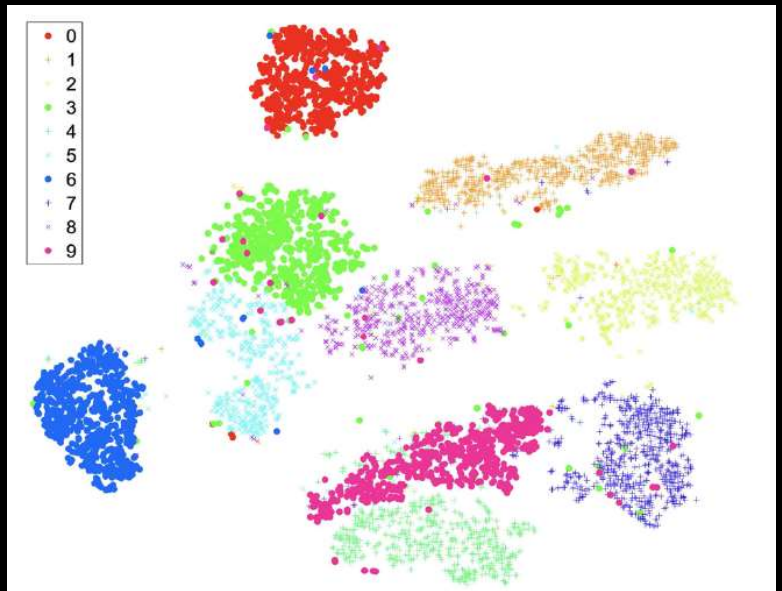
1. **t-Distributed Stochastic Neighbor Embedding**
2. **Scatter and Violin Plots**
3. **Scanpy Plots**

t-Distributed Stochastic Neighbor Embedding (t-SNE)

- Non-linear dimensionality reduction for embedding high dimensional data in low dimensional space i.e., 2D or 3D
- Visualize high dimensional data – give each point a location in 2D or 3D map
- Randomized approach to reduce dimensionality non-linearly
- Retains local structure of data in lower dimension
- Algorithm – find patterns in data based on similarity of datapoints with features
- Convert high dimension Euclidean distances between data points into joint probabilities
- Similarity – calculate conditional probability that point A chooses point B as neighbour
- Minimize distance between conditional probabilities in higher dimensional and lower dimensional space to represent data points in lower dimension

t-Distributed Stochastic Neighbor Embedding (t-SNE)

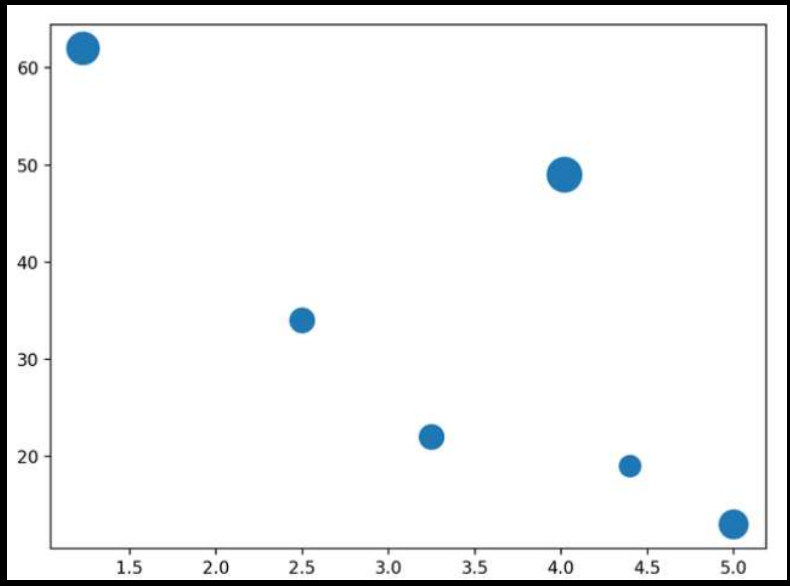
- Reduce dimensionality, preserve most information in dataset
- Unsupervised algorithm
- Increase interpretability of data in lower dimension
- Minimize information loss due to dimensionality reduction



A two-dimensional t-SNE map of the MNIST data set, a collection of 60,000 images of handwritten digits,

Scatter Plots and Violin Plots

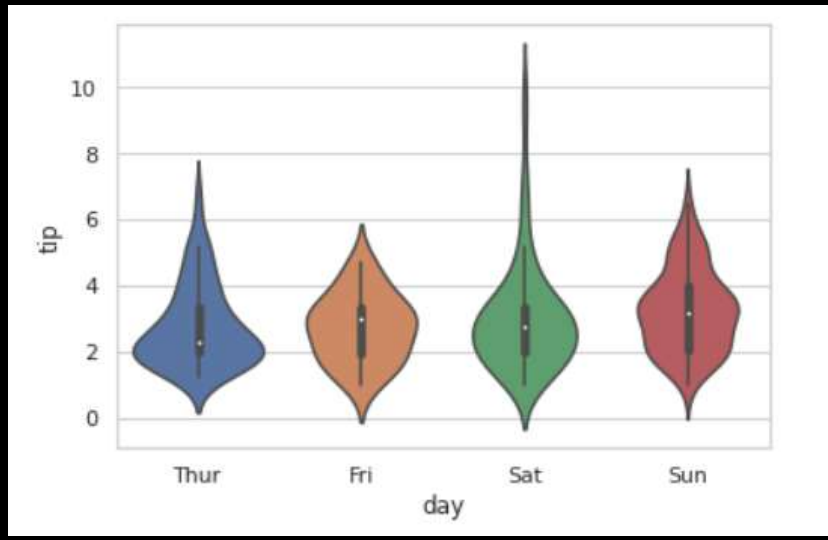
- **Scatter Plot**
  - Represent relationship among variables
  - Dots used to represent the relationship
  - Find correlation between variables
  - Demonstrate how change in one variable affects the other
  - Scatter() method in matplotlib library to draw scatter plots



Price vs Sales per day

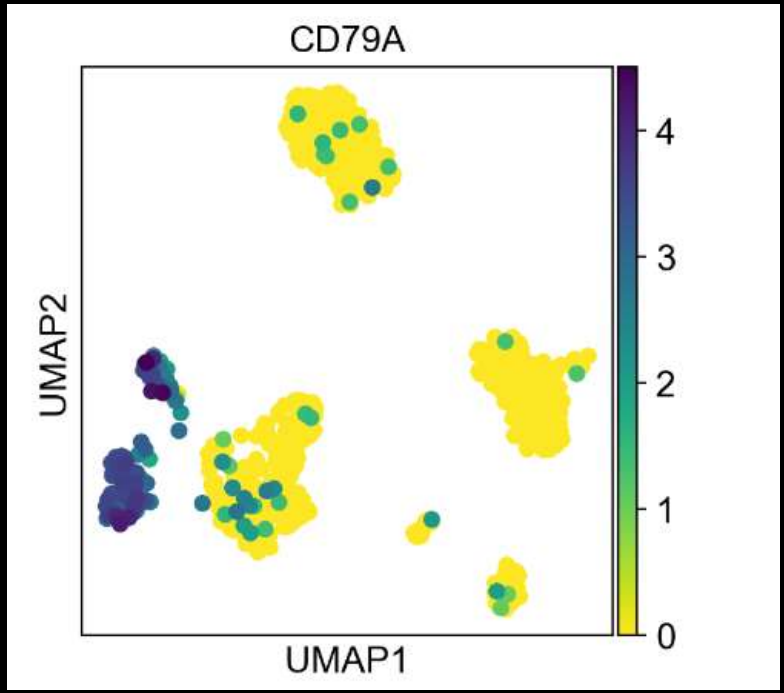
Scatter Plots and Violin Plots

- **Violin Plot**
  - Statistical representation of numerical data
  - Shows quantitative data across one or more categorical variables
  - Distribution of data points after grouping by one or more variables
  - Effective and attractive to show multiple data at several units
  - violinplot() method in seaborn library to draw violin plots



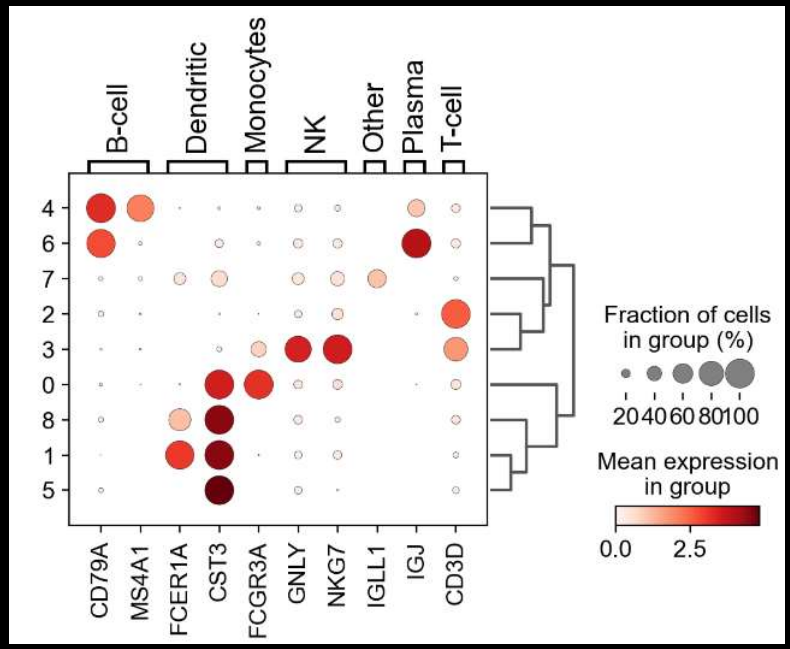
Scanpy Plots

- Scanpy – scalable toolkit for analyzing single-cell gene expression data built jointly with anndata
- **Scatter plots for embeddings**
- In scanpy, scatter plots for t-SNE, UMAP and several other embeddings readily available using ‘sc.pl.tsne’, ‘sc.pl.umap’ etc. functions
- These functions access data stored in ‘adata.obsm’



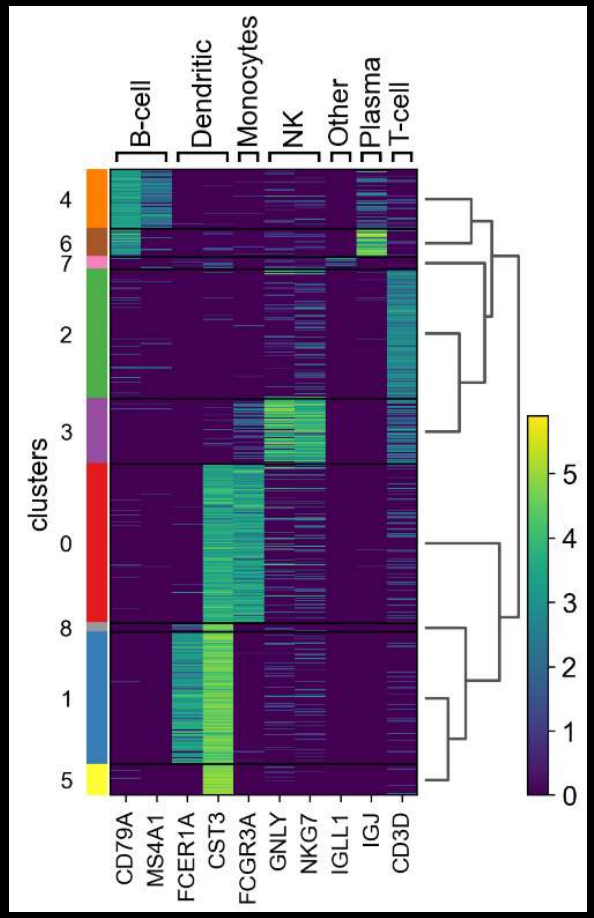
Scanpy Plots

- **Identification of clusters based on known marker genes**
- Clusters need to be labelled using well known marker genes
- **Dotplot –** check expressions of genes per cluster
- Color represents mean expression within each category(cluster)
- Dot represents fraction of cells in categories expressing a gene



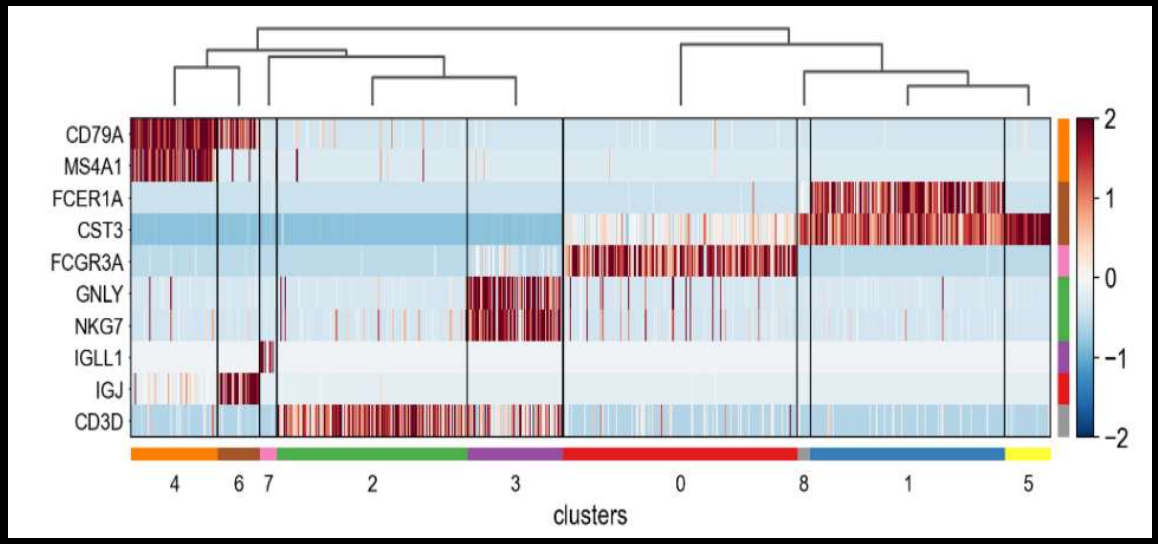
Scanpy Plots

- **Heatmaps**
- Find collinearity of data
- Each value in matrix is represented as a color
- Each cell is shown in a row
- Groupby information can be added



Scanpy Plots

- **Tracksplot**
- Same information as heatmap
- Instead of color scale, gene expression is represented as height



Scanpy Plots

- **Dendrogram**
- Tree like structure to visualize relationship among clusters
- Illustrates how each cluster is composed by drawing U-shaped link between non-singleton cluster and its children
- To arrange categories