Dimensionality reduction: feature extraction

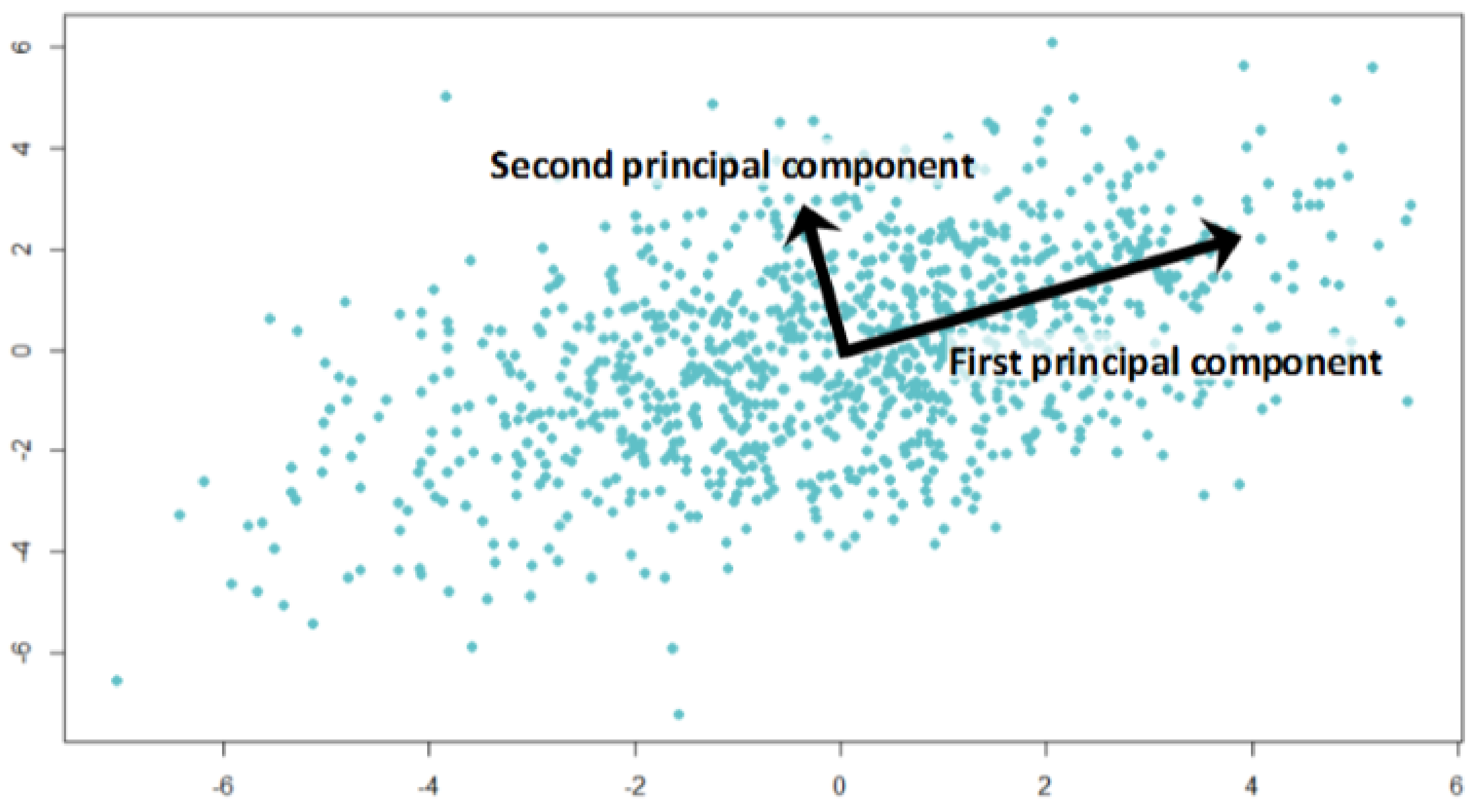
PREPARING FOR MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

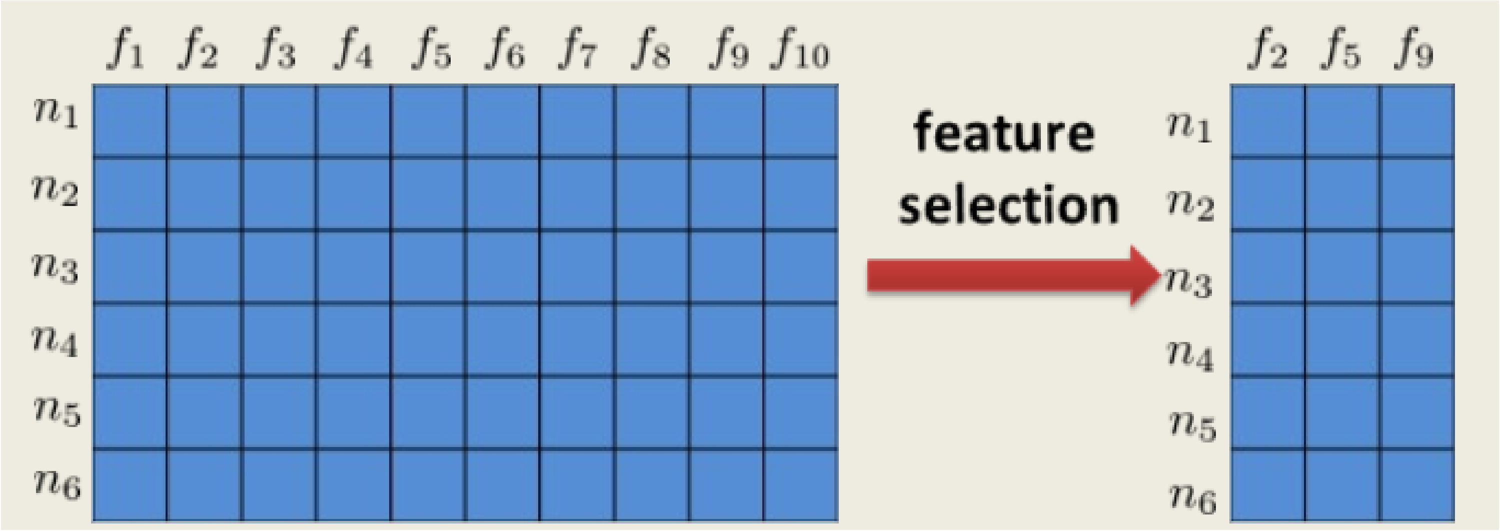
Lisa Stuart
Data Scientist

Unsupervised learning methods

- Principal component analysis (PCA) --> Lesson 3.1
- Singular value decomposition (SVD) --> Lesson 3.1
- Clustering/grouping --> Lesson 3.3
- Exploratory data mining

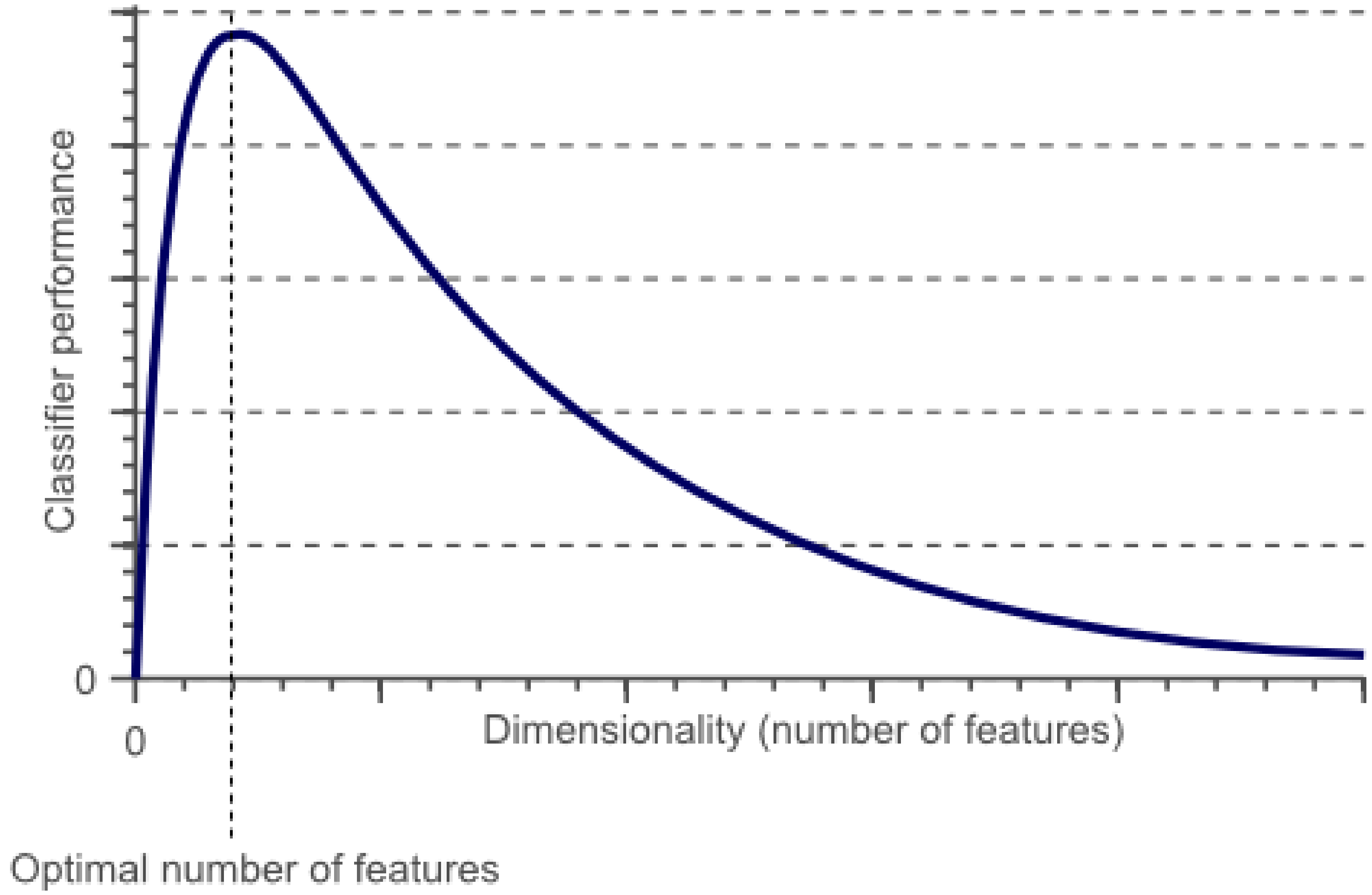
Dimensionality reduction != feature selection





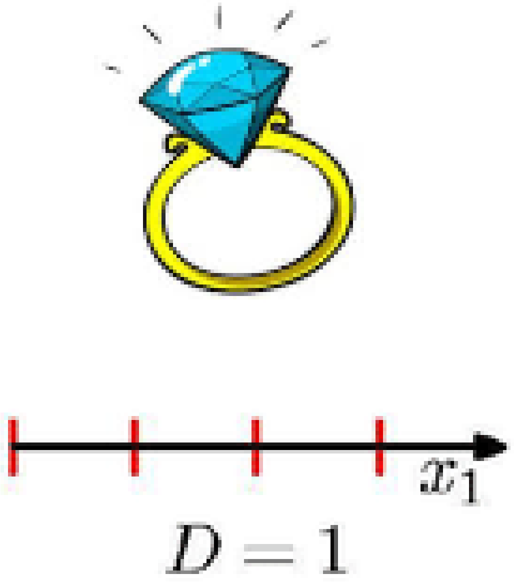
[1] https://slideplayer.com/slide/9699240/ [2] https://www.analyticsvidhya.com/blog/2016/03/practical [3] guide [4] principal [5] component [6] analysis [7] python/

Curse of dimensionality

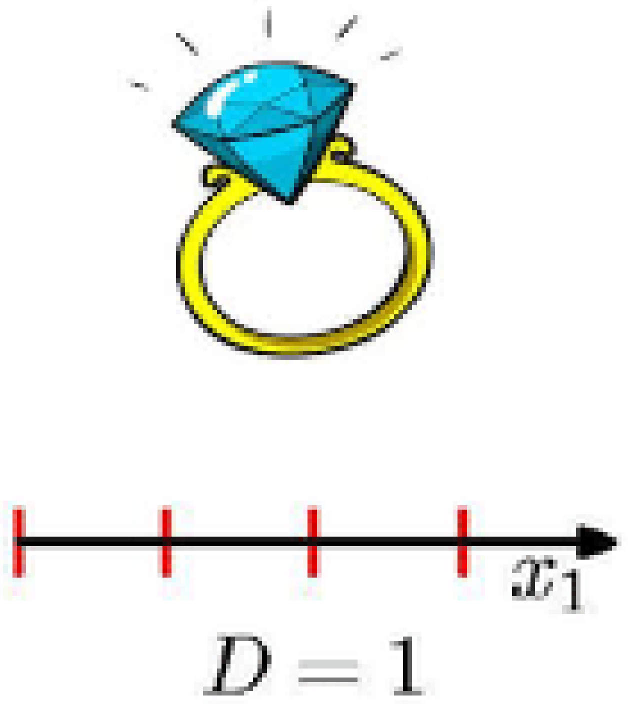


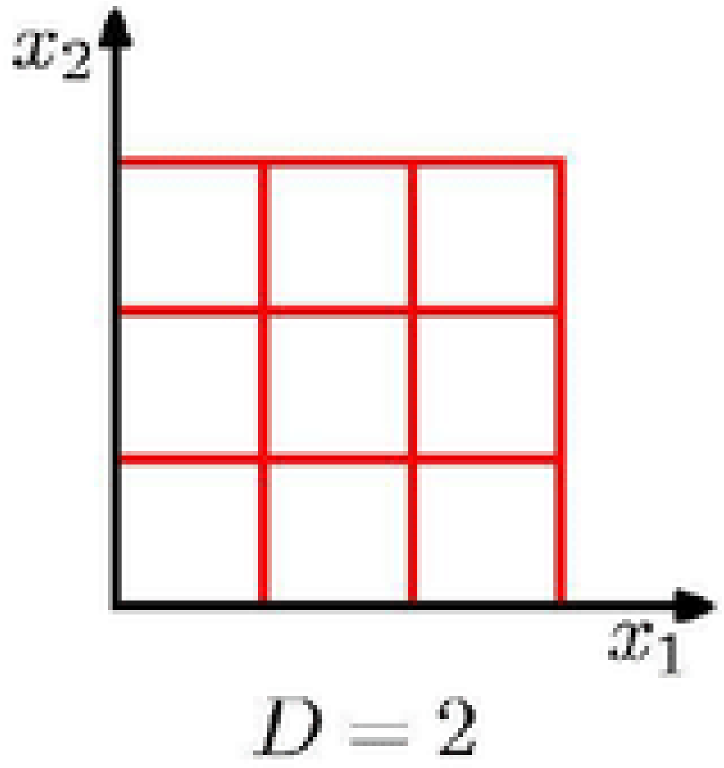
[1] https://www.visiondummy.com/2014/04/curse [2] dimensionality [3] affect [4] classification/

1-D search

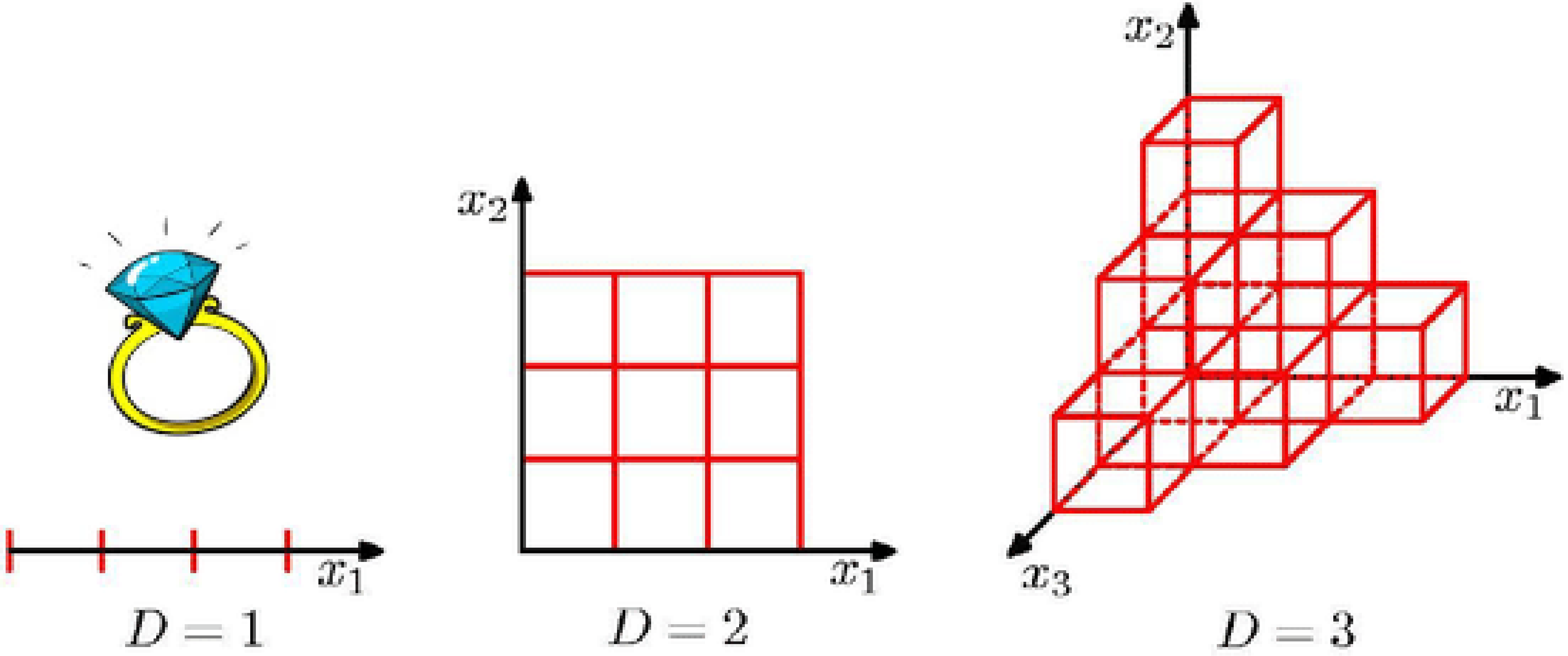


2-D search





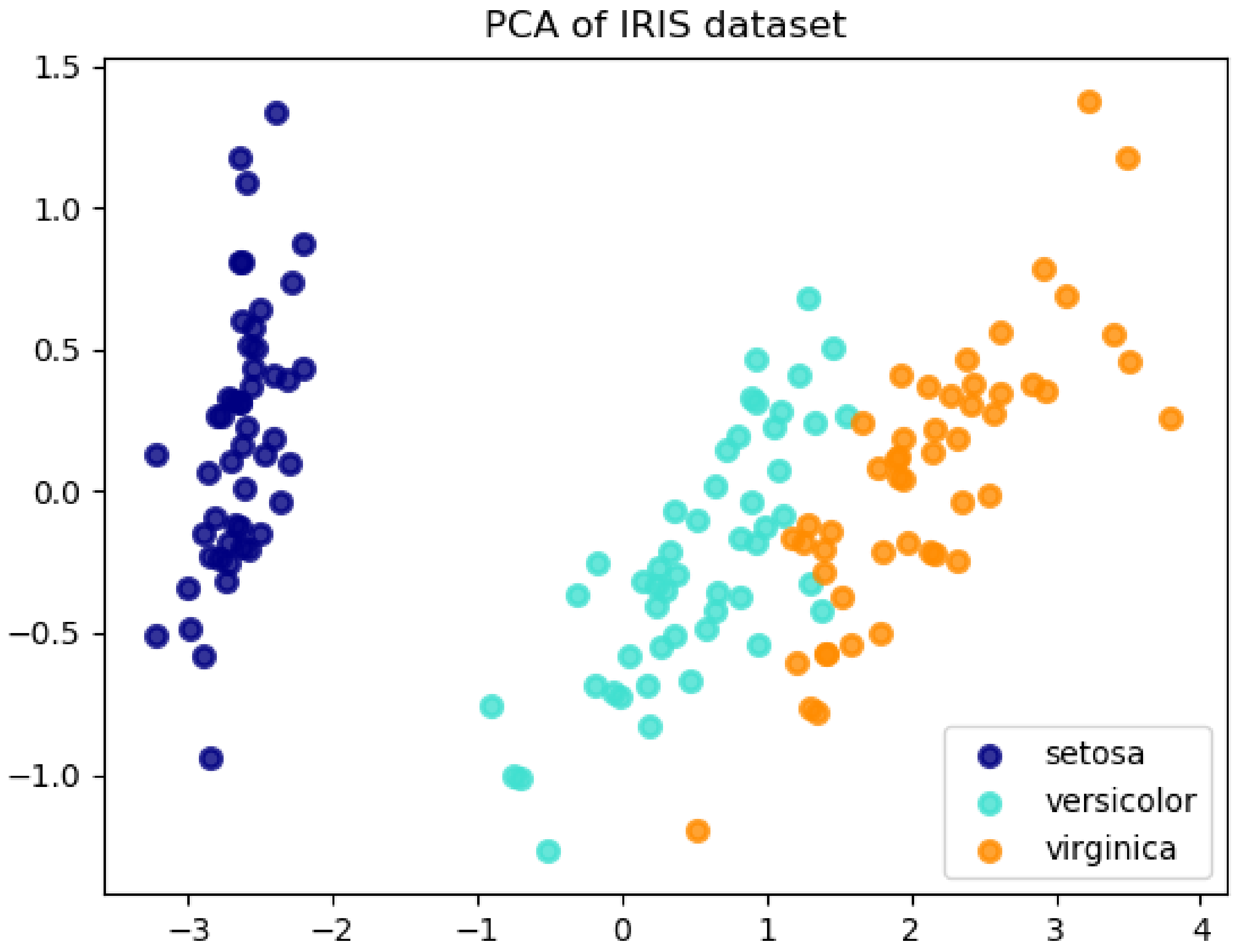
3-D search



Dimensionality reduction methods

- PCA
- SVD

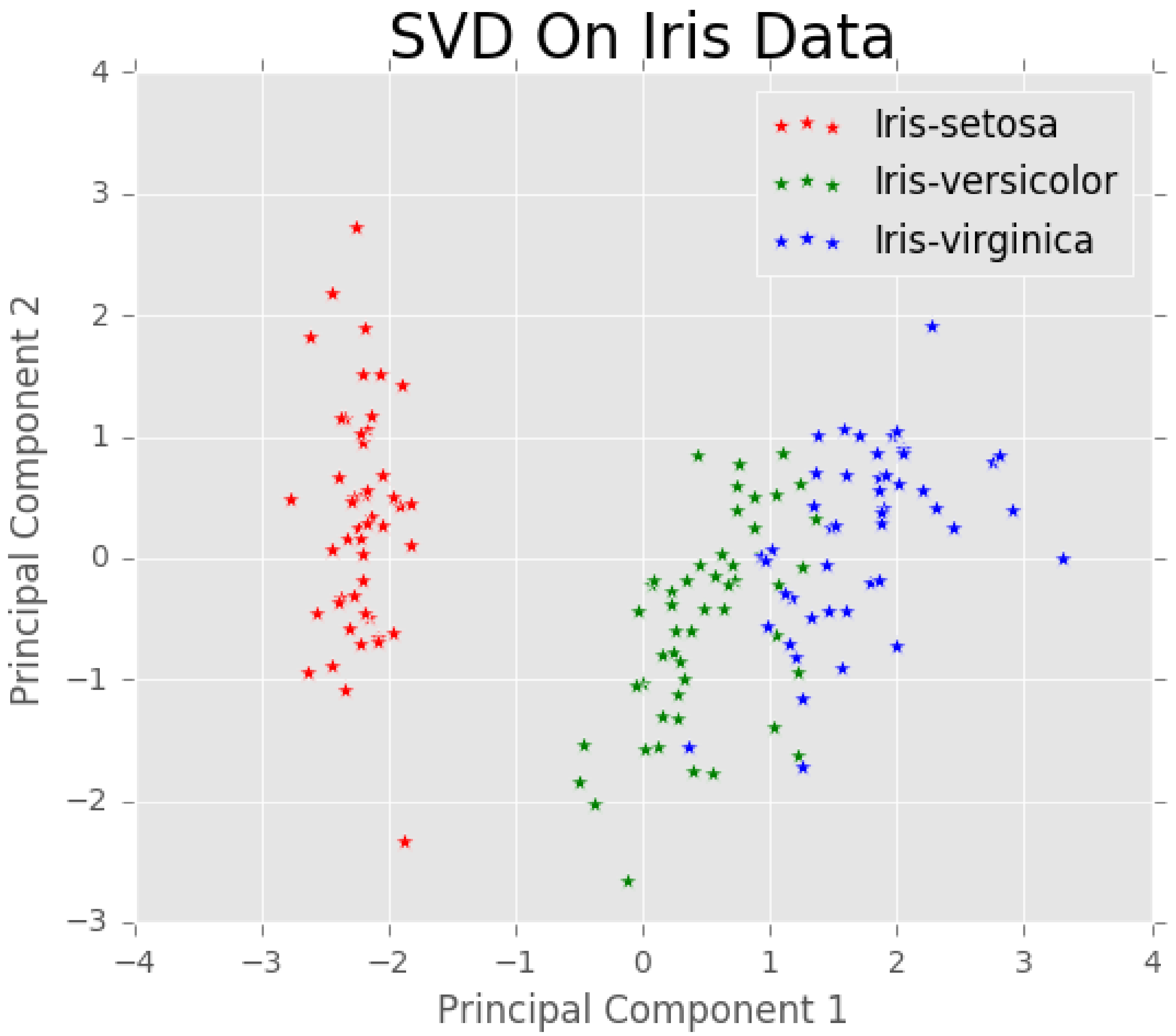
PCA



- PCA
  - Relationship between X and y
  - Calculated by finding principal axes
  - Translates, rotates and scales
  - Lower-dimensional projection of the data

[1] https://scikit [2] learn.org/stable/modules/decomposition.html

SVD



- SVD
  - Linear algebra and vector calculus
  - Decomposes data matrix into three matrices
  - Results in 'singular' values
  - Variance in data approximately equals SS of singular values

[1] https://galaxydatatech.com/2018/07/15/singular [2] value [3] decomposition/

Dimension reduction functions

Function/method	returns
`sklearn.decomposition.PCA`	principal component analysis
`sklearn.decomposition.TruncatedSVD`	singular value decomposition
`PCA/SVD.fit_transform(X)`	fits and transforms data
`PCA/SVD.explained_variance_ratio_`	variance explained by PCs

- Other matrix decomposition algorithms

Let's practice!

PREPARING FOR MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

Dimensionality reduction: visualization techniques

PREPARING FOR MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

Lisa Stuart
Data Scientist

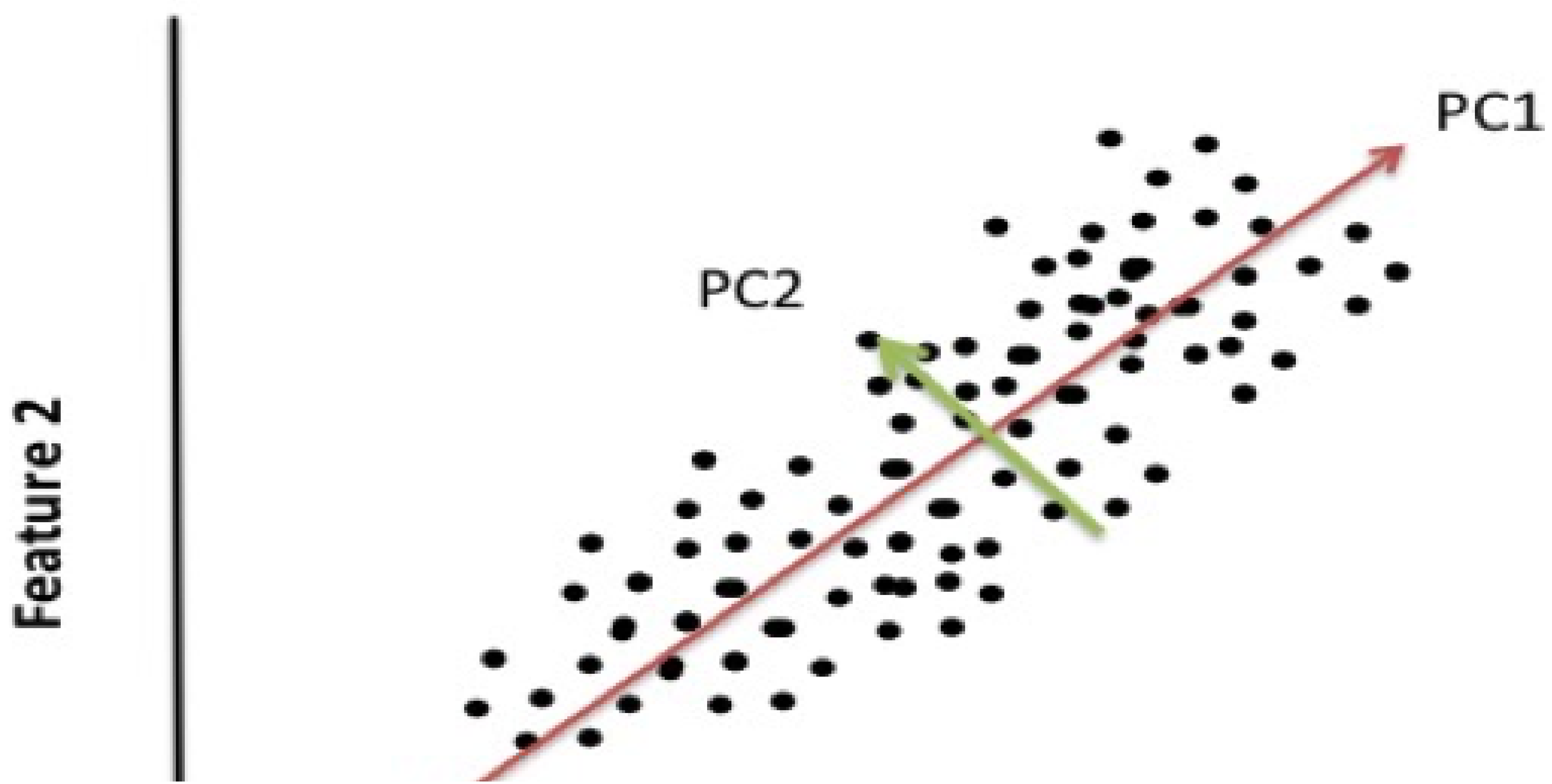
Why dimensionality reduction?

1. Speed up ML training
2. Visualization
3. Improves accuracy

Visualization techniques

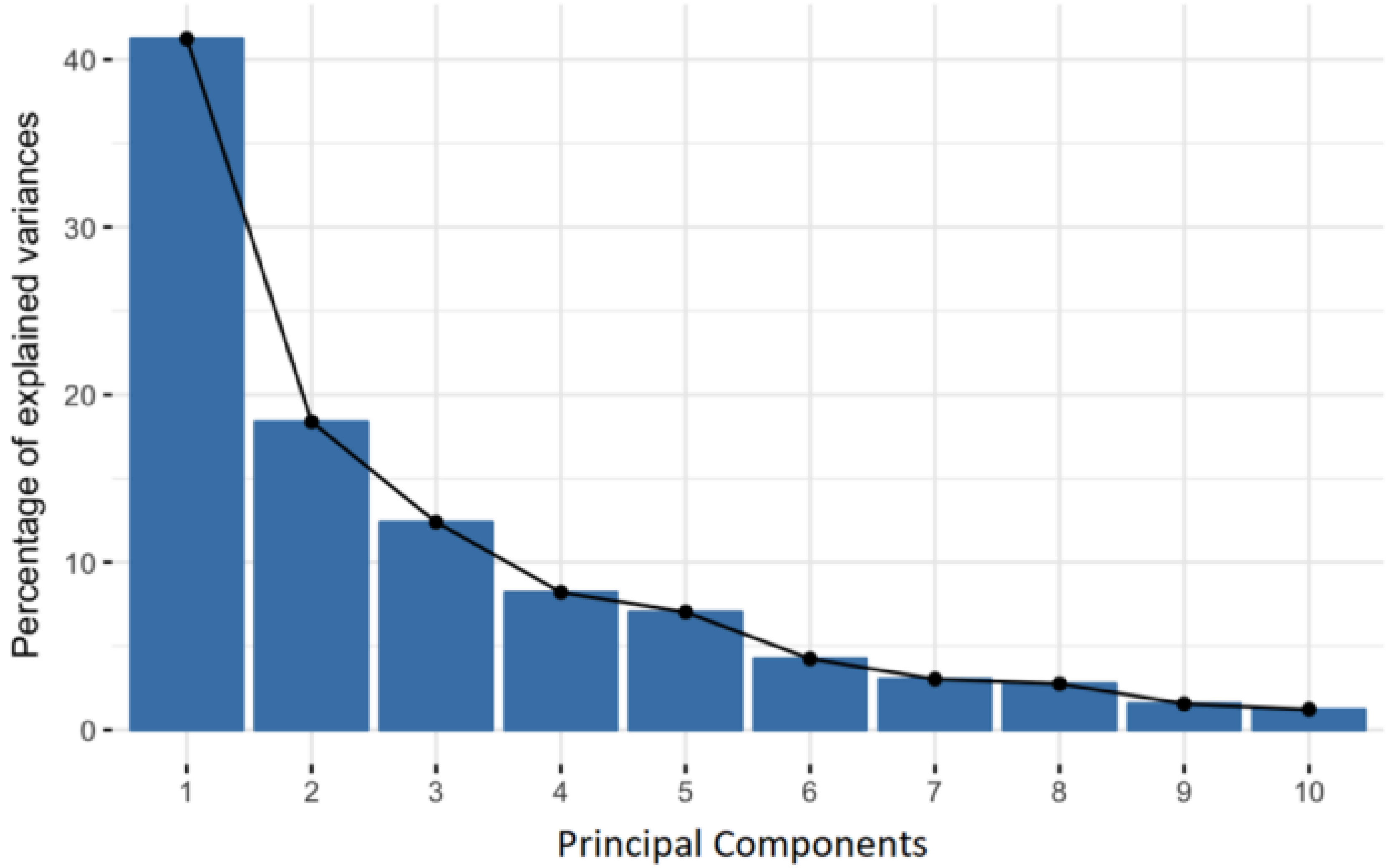
- PCA
- t-SNE

Visualizing with PCA



[1] https://districtdatalabs.silvrback.com/principal [2] component [3] analysis [4] with [5] python

Scree plot



[1] https://towardsdatascience.com/a [2] step [3] by [4] step [5] explanation [6] of [7] principal [8] component [9] analysis [10] b836fb9c97e2

t-SNE

- Probabilistic
- Pairs of data points
- Low-dimensional embedding
- Plot embeddings

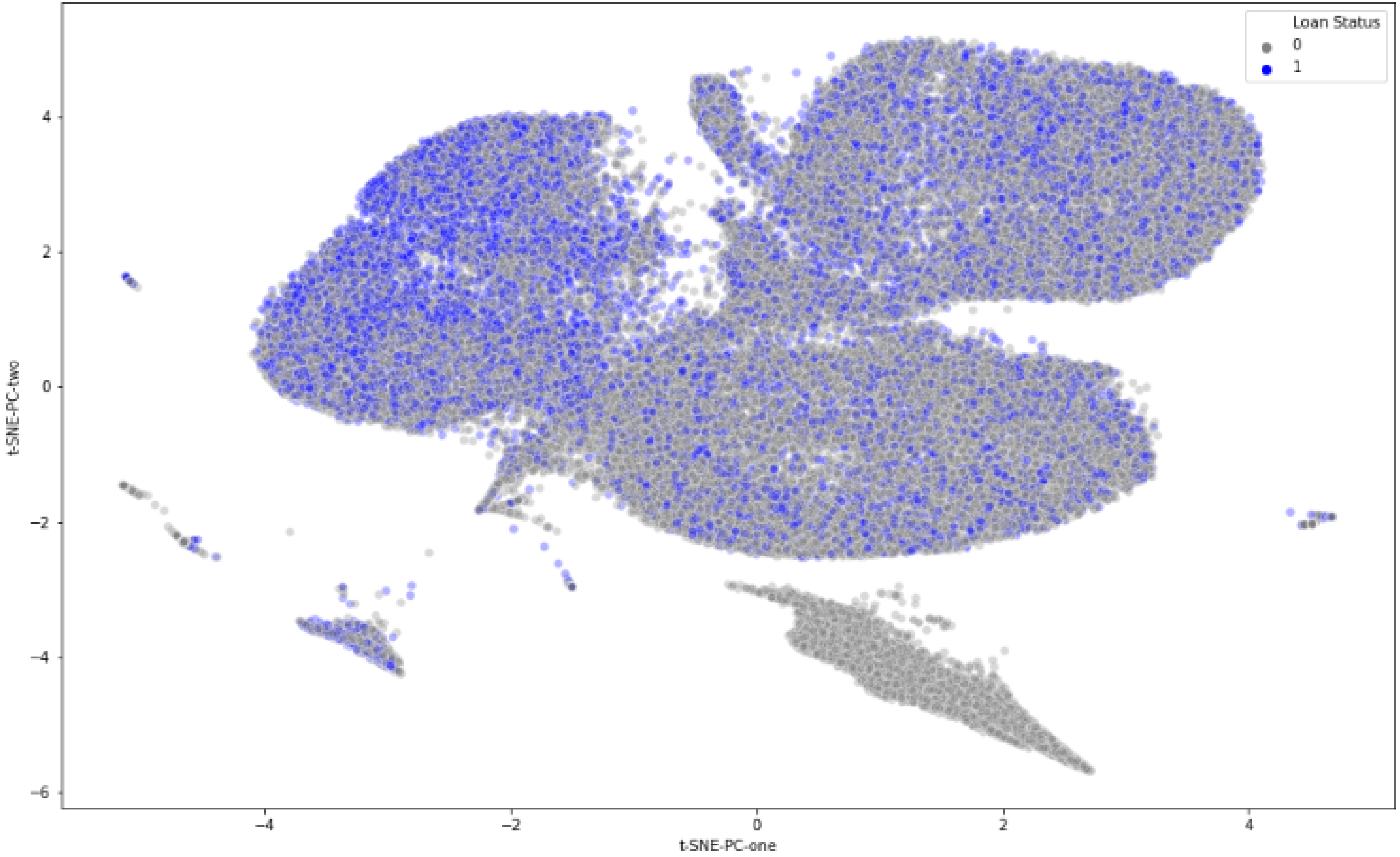
Visualizing with t-SNE

```
# t-sne with loan data
from sklearn.manifold import TSNE
import seaborn as sns

loans =  pd.read_csv('loans_dataset.csv')

# Feature matrix
X = loans.drop('Loan Status', axis=1)

tsne = TSNE(n_components=2, verbose=1, perplexity=40)
tsne_results = tsne.fit_transform(X)

loans['t-SNE-PC-one'] = tsne_results[:,0]
loans['t-SNE-PC-two'] = tsne_results[:,1]
```

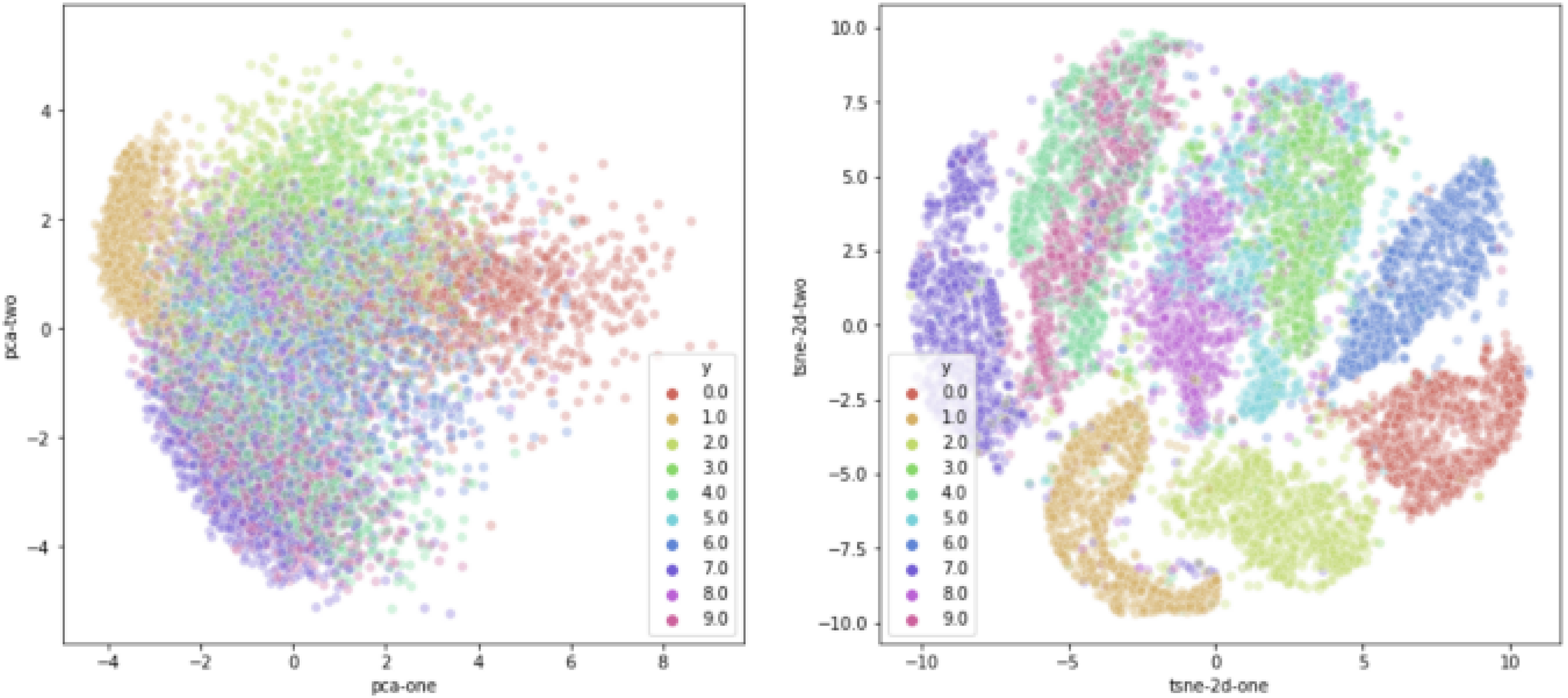
```
# t-sne viz
plt.figure(figsize=(16,10))
sns.scatterplot(
    x="t-SNE-PC-one", y="t-SNE-PC-two",
    hue="Loan Status",
    palette=sns.color_palette(["grey","blue"]),
    data=loans,
    legend="full",
    alpha=0.3
)
```

[1] https://scikit [2] learn.org/stable/modules/generated/sklearn.manifold.TSNE.html

Visualizing with t-SNE



PCA vs t-SNE digits data



[1] https://towardsdatascience.com/visualising [2] high [3] dimensional [4] datasets [5] using [6] pca [7] and [8] t [9] sne [10] in [11] python [12] 8ef87e7915b

Let's practice!

PREPARING FOR MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

Clustering analysis: selecting the right clustering algorithm

PREPARING FOR MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

Lisa Stuart
Data Scientist

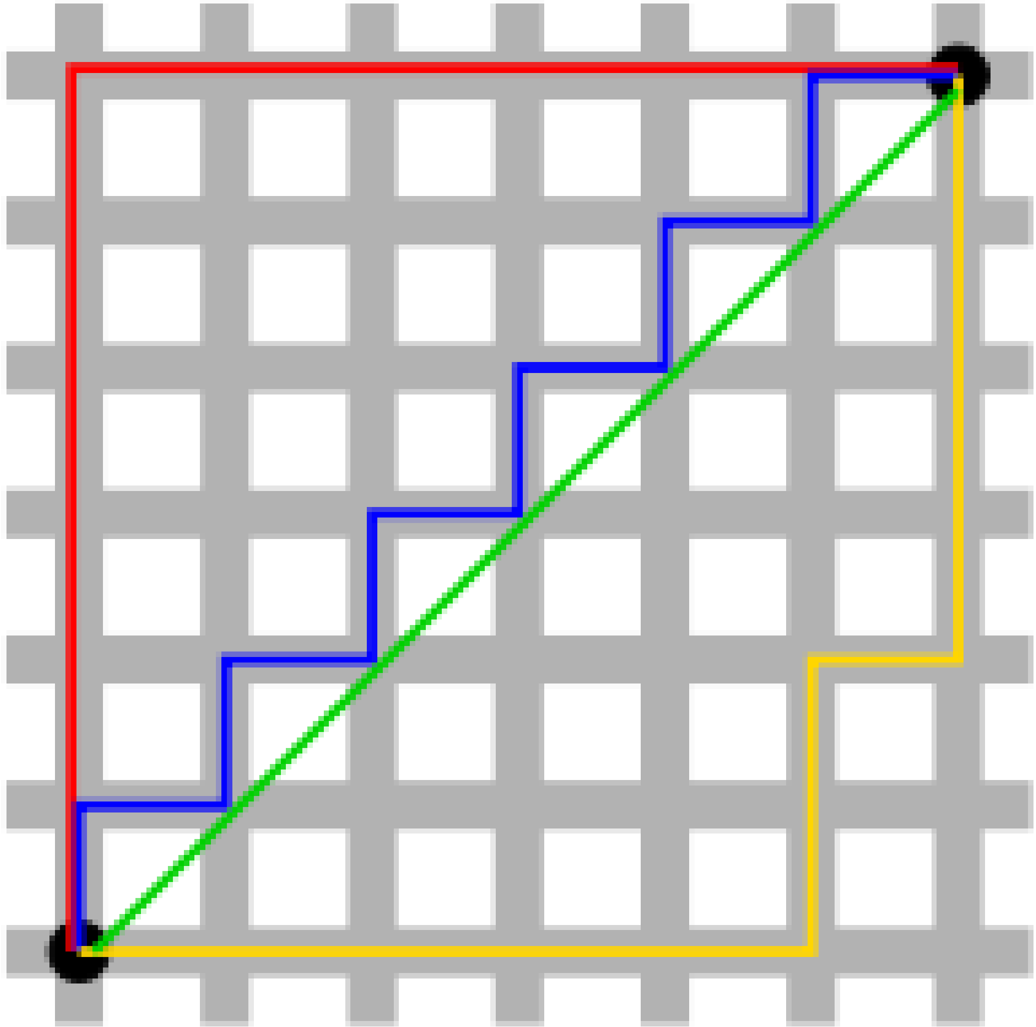
Clustering algorithms

- Features >> Observations
- Model training more challenging
- Rely on distance calculations
- Most commonly used unsupervised technique

Practical applications of clustering

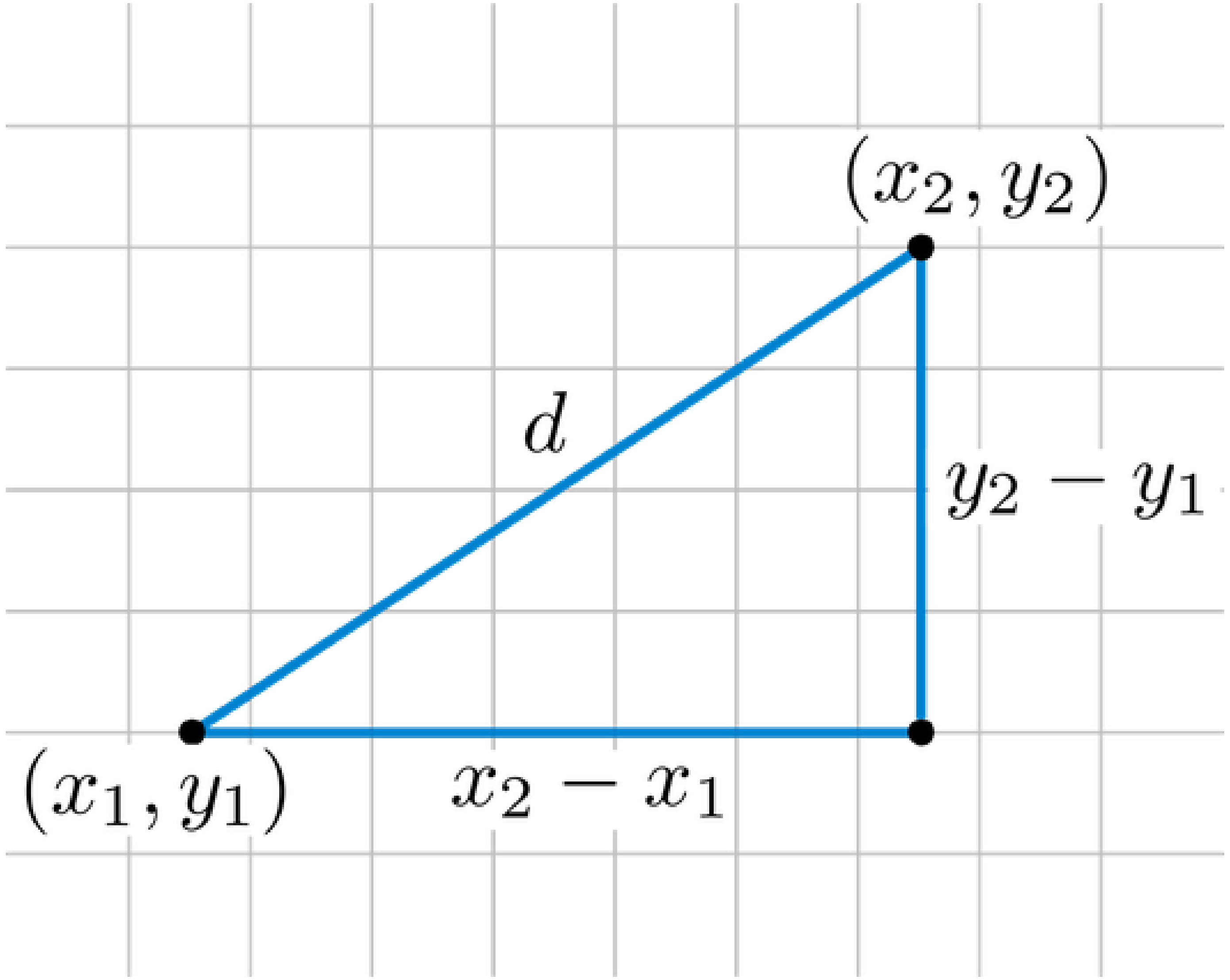
- Customer segmentation
- Document classification
- Insurance/transaction fraud detection
- Image segmentation
- Anomaly detection
- Many more...

Distance metrics: Manhattan (taxicab) distance



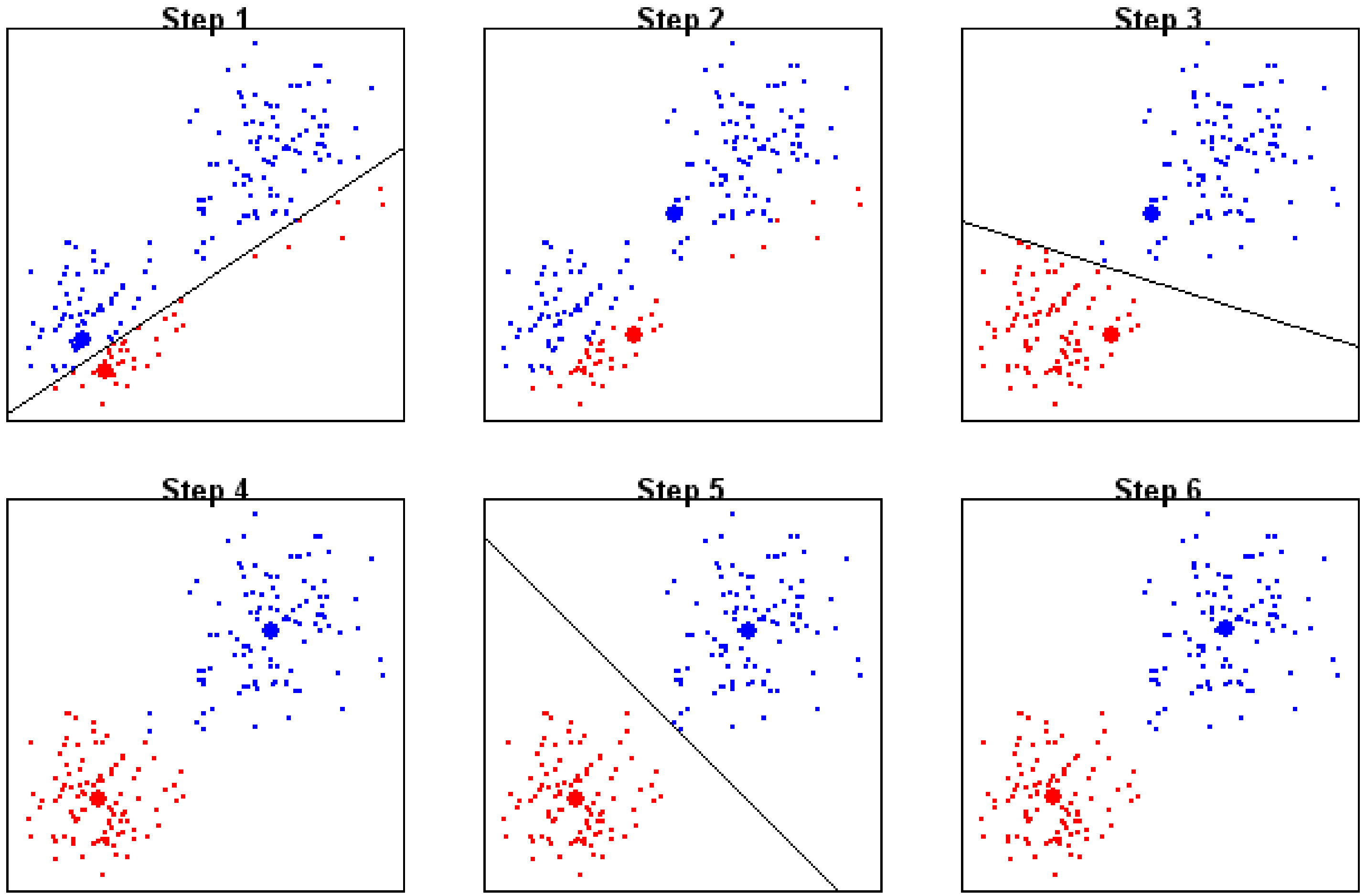
[1] https://en.wikipedia.org/wiki/Taxicab_geometry

Distance metrics: Euclidian distance



[1] http://rosalind.info/glossary/euclidean [2] distance/

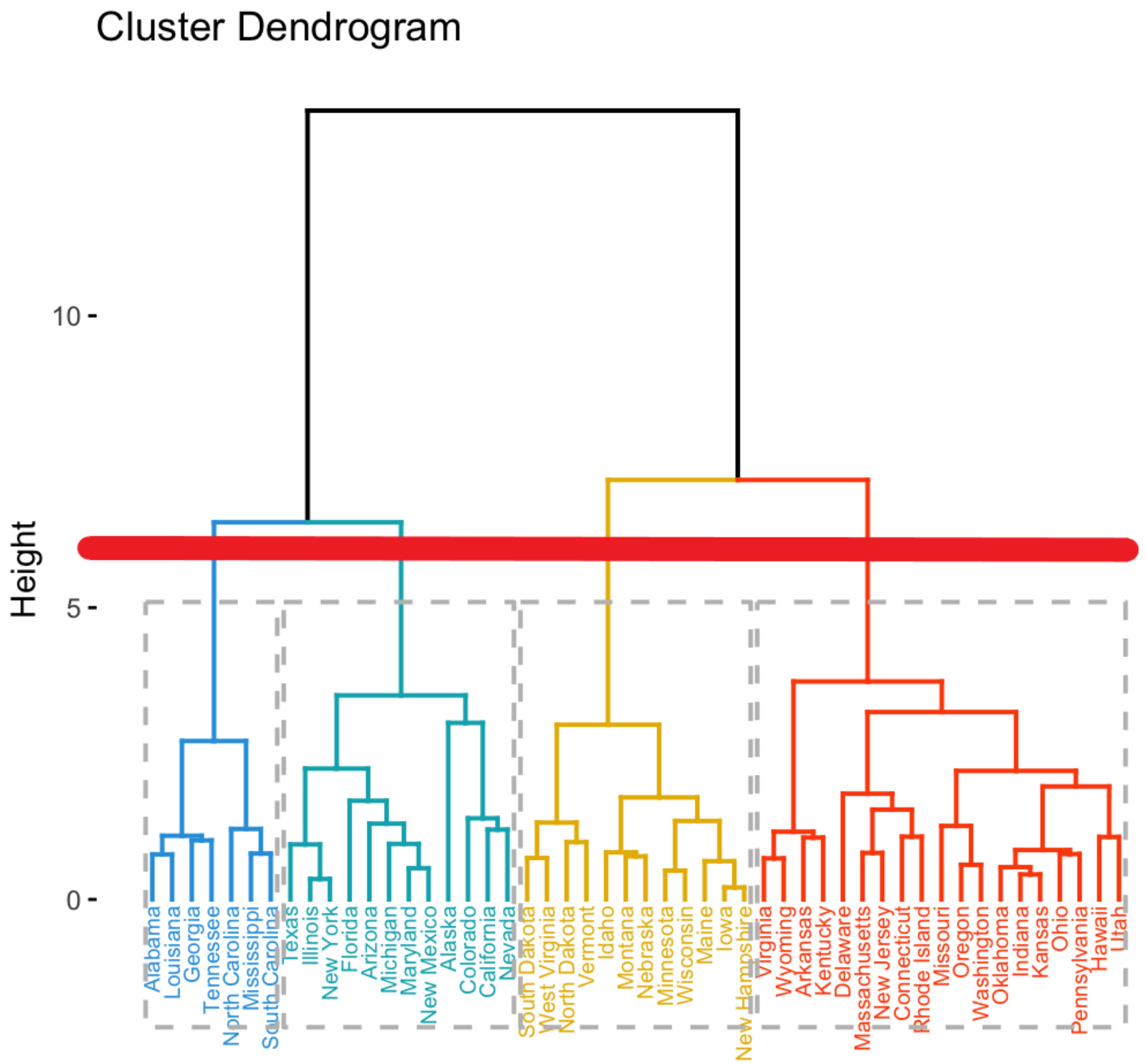
K-means



1. Initial centroids
2. Assign each observation to nearest centroid
3. Create new centroids
4. Repeat steps 2 and 3

[1] http://sherrytowers.com/2013/10/24/k [2] means [3] clustering/

Hierarchical agglomerative clustering

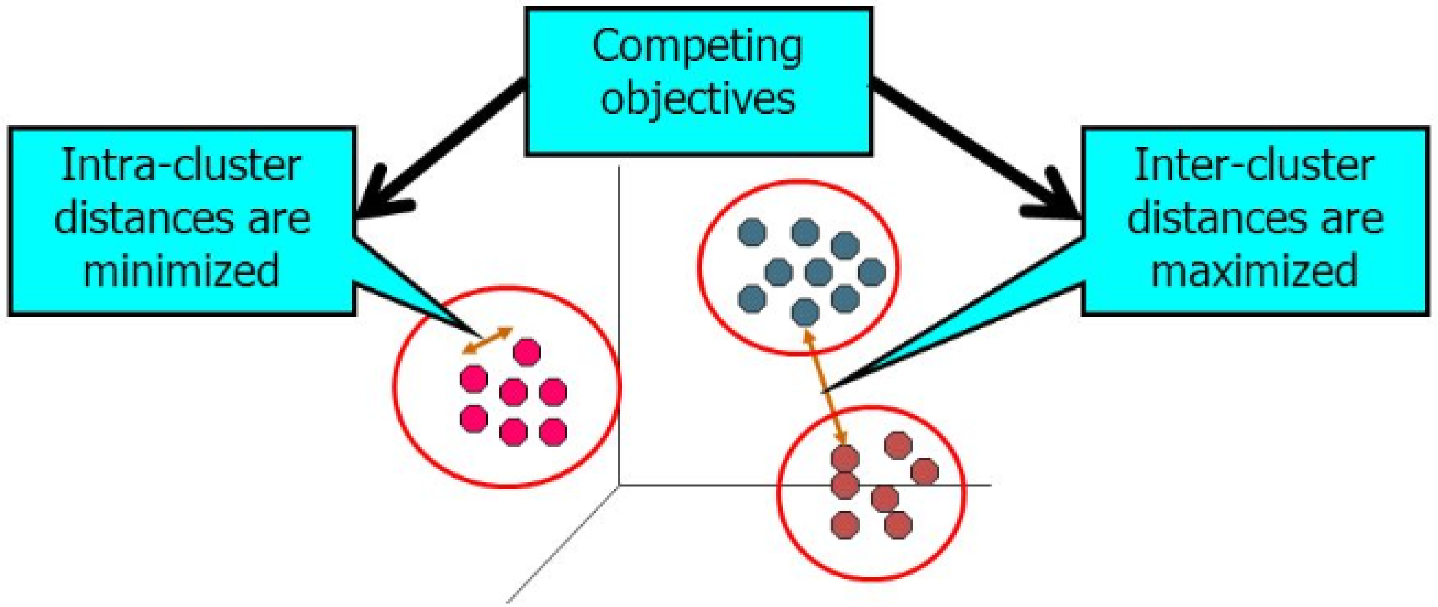


[1] https://www.datanovia.com/en/lessons/agglomerative [2] hierarchical [3] clustering/

Agglomerative clustering linkage

- Ward linkage
- Maximum/complete linkage
- Average linkage
- Single linkage

Selecting a clustering algorithm



- Cluster stability assessment
- K-means and HC use Euclidian distance
- Inter- and intra-cluster distances

> "An appropriate dissimilarity measure is far more important in obtaining success with clustering than choice of clustering algorithm." - from **Elements of Statistical Learning**

[1] https://slideplayer.com/slide/8363774/

Clustering functions

Function/method	returns
`sklearn.cluster.Kmeans`	K-Means clustering algorithm
`sklearn.cluster.AgglomerativeClustering`	Agglomerative clustering algorithm
`kmeans.inertia_`	SS distances of observations to closest cluster center
`scipy.cluster.hierarchy` as `sch`	Hierachical clustering for dendrograms
`sch.dendrogram()`	Dendrogram function

Let's practice!

PREPARING FOR MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

Clustering analysis: choosing the optimal number of clusters

PREPARING FOR MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

Lisa Stuart
Data Scientist

Methods for optimal k

- Silhouette method
- Elbow method

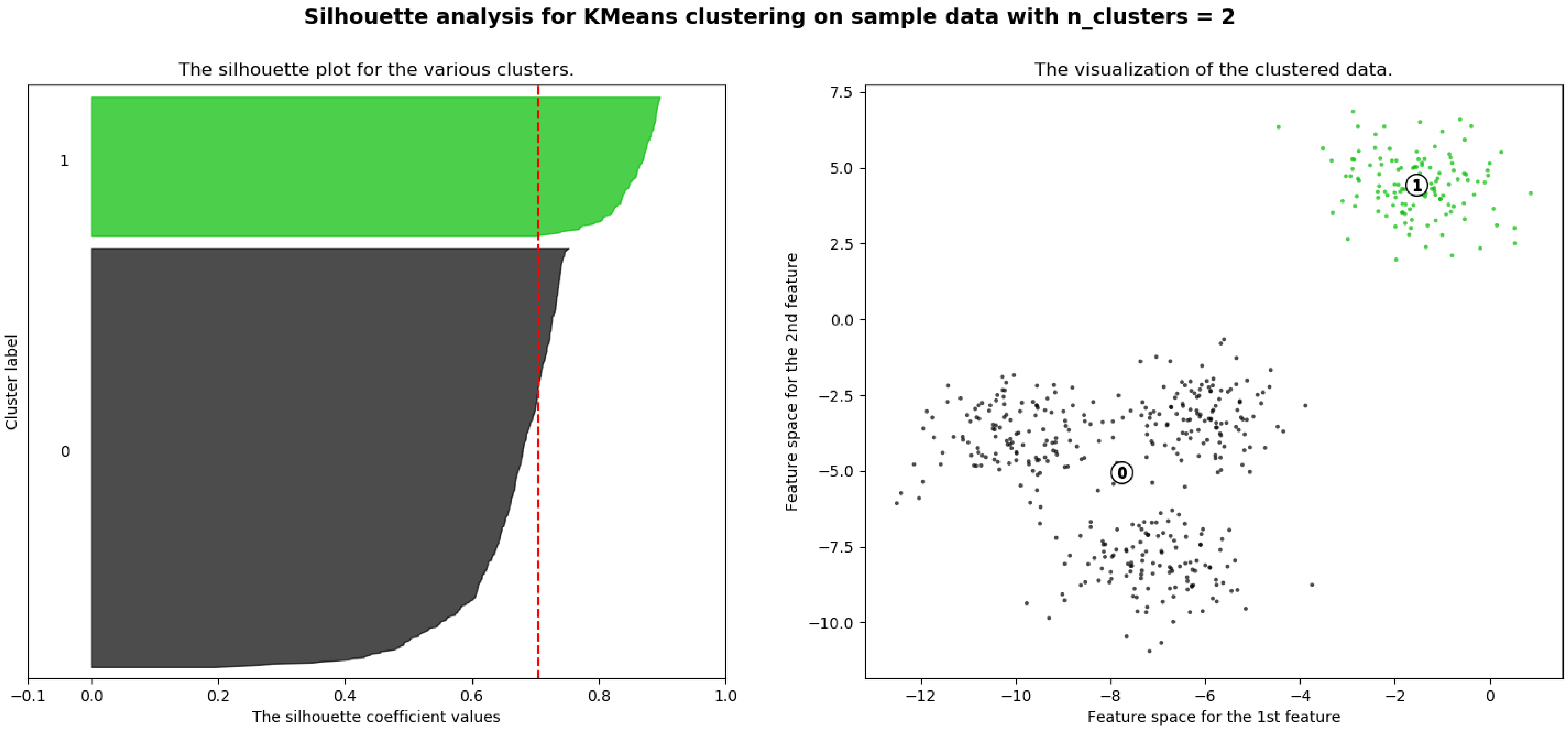
Silhouette coefficient

- Composed of 2 scores
  - Mean distance between each observation and all others:
    - in the same cluster
    - in the nearest cluster

Silhouette coefficient values

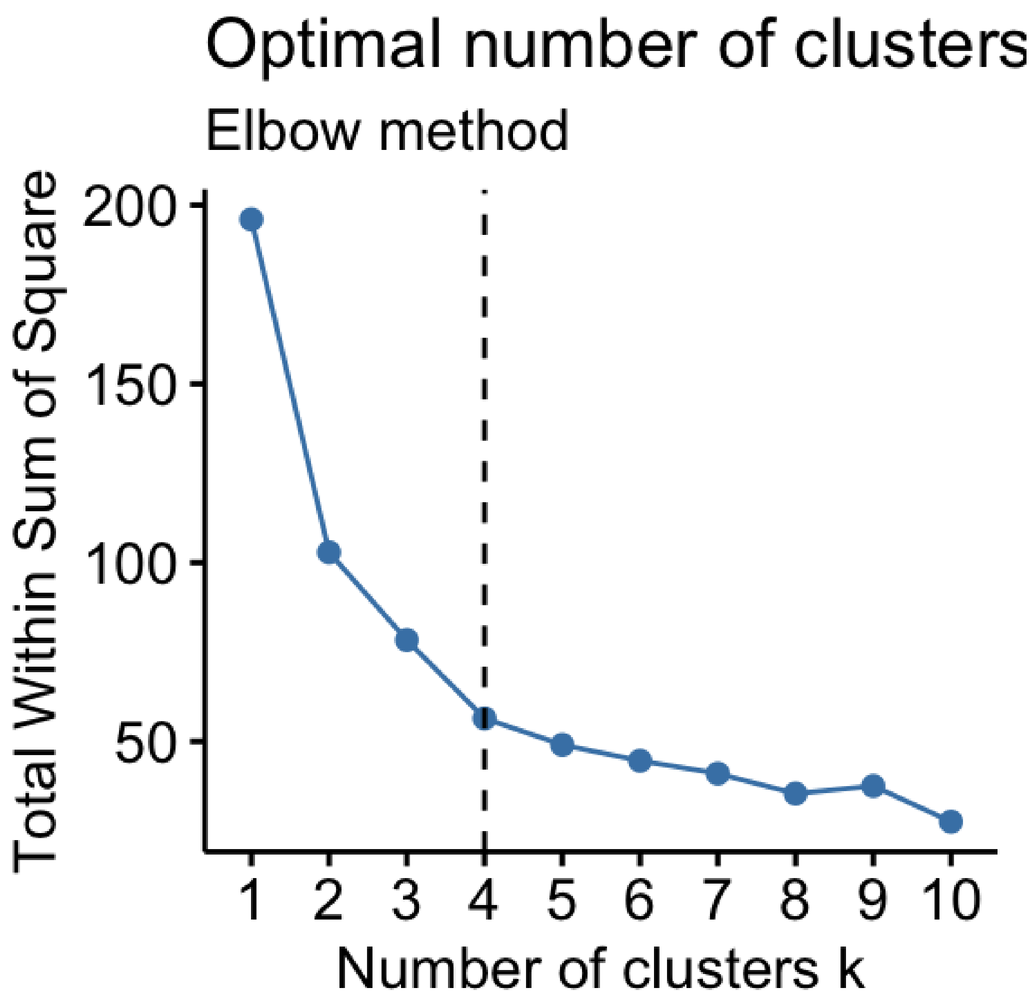
- Between -1 and 1
  - 1
    - near others in same cluster
    - very far from others in other clusters
  - -1
    - not near others in same cluster
    - close to others in other clusters
  - 0
    - denotes overlapping clusters

Silhouette score



[1] https://scikit [2] learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html

Elbow method



[1] https://www.datanovia.com/en/lessons/determining [2] the [3] optimal [4] number [5] of [6] clusters [7] 3 [8] must [9] know [10] methods/

Optimal k selection functions

Function/method	returns
`sklearn.cluster.KMeans`	K-Means clustering algorithm
`sklearn.metrics.silhouette_score`	score between -1 and 1 as measure of cluster stability
`kmeans.inertia_`	SS distances of observations to closest cluster center
`range(start, stop)`	list of values beginning with start, up to but not including stop
`list.append(kmeans.inertia_)`	appends inertia value to list

Let's practice!

PREPARING FOR MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON