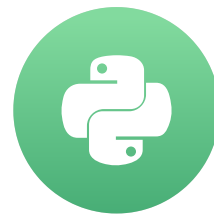


Dimensionality reduction: feature extraction

PRACTICING MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

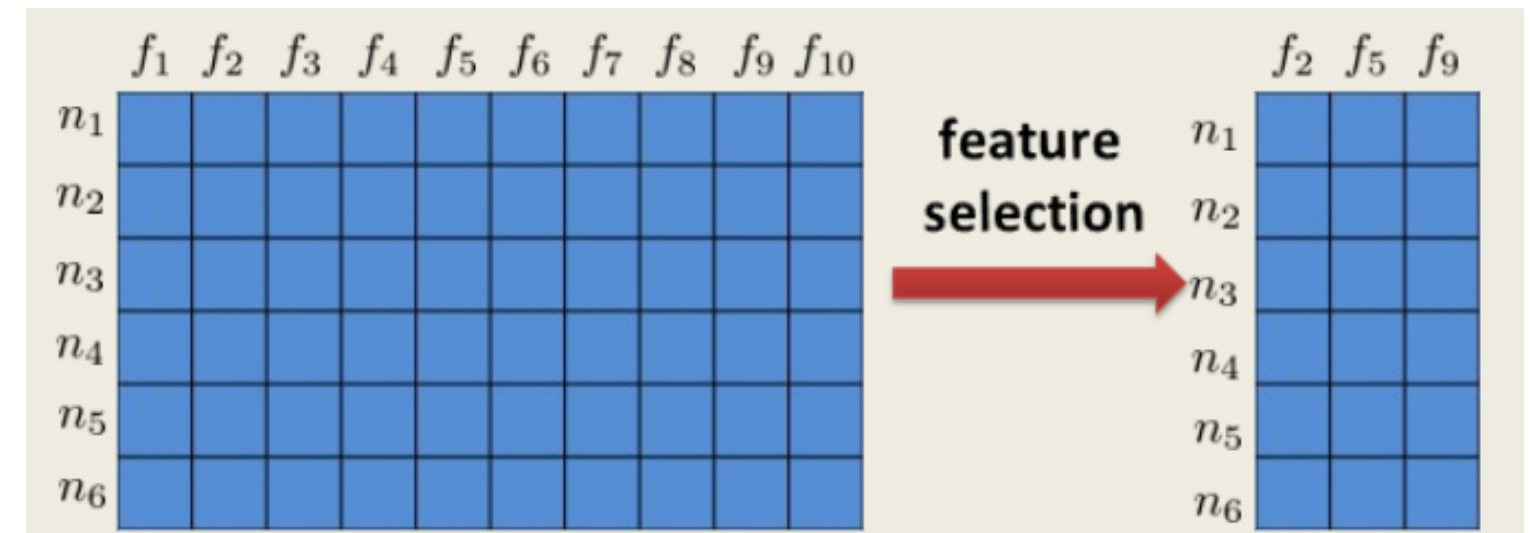
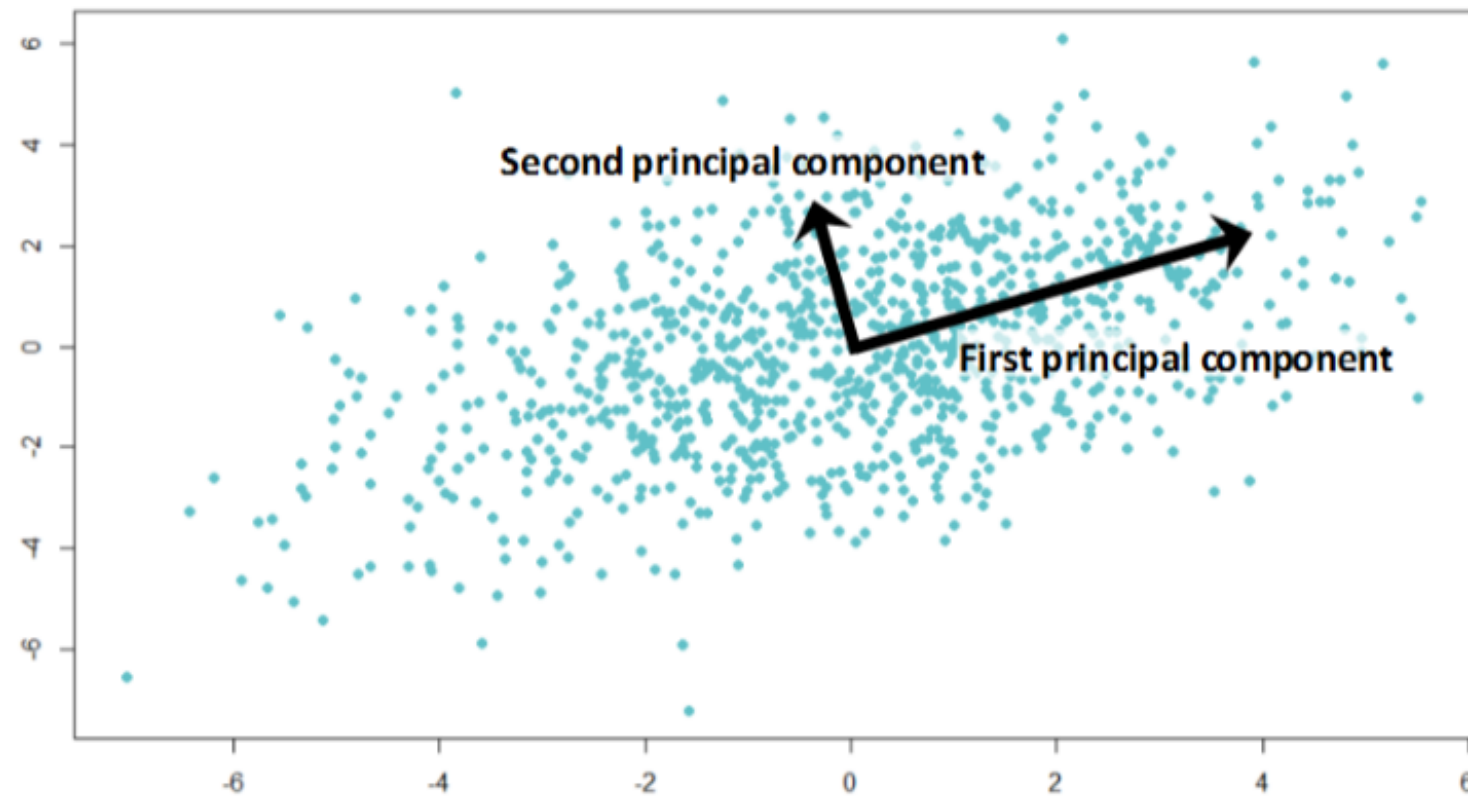
Lisa Stuart
Data Scientist



Unsupervised learning methods

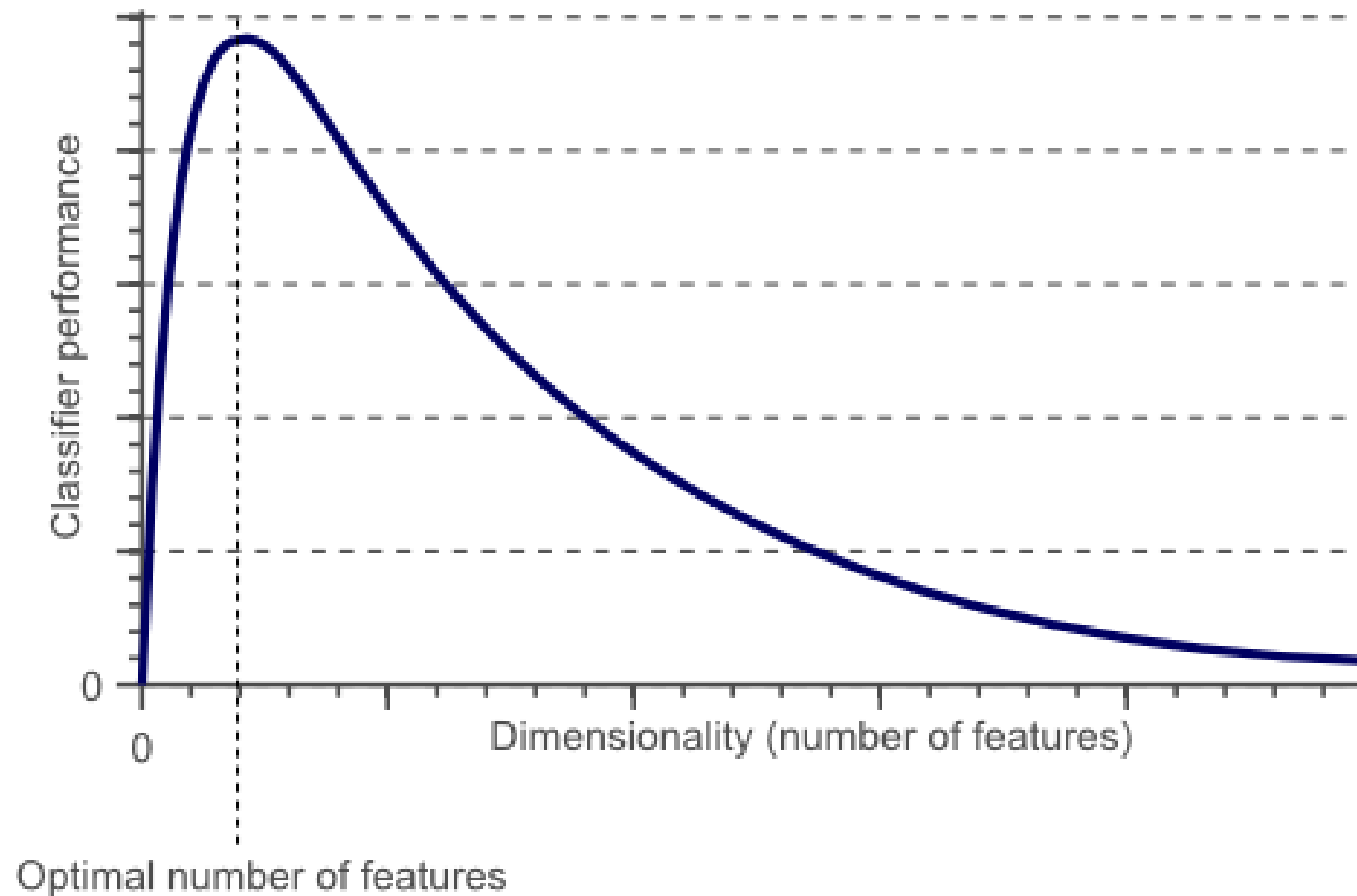
- Principal component analysis (PCA) --> Lesson 3.1
- Singular value decomposition (SVD) --> Lesson 3.1
- Clustering/grouping --> Lesson 3.3
- Exploratory data mining

Dimensionality reduction != feature selection



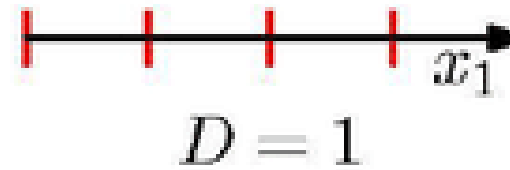
¹ <https://slideplayer.com/slide/9699240/> ² <https://www.analyticsvidhya.com/blog/2016/03/practical-guide-principal-component-analysis-python/>

Curse of dimensionality

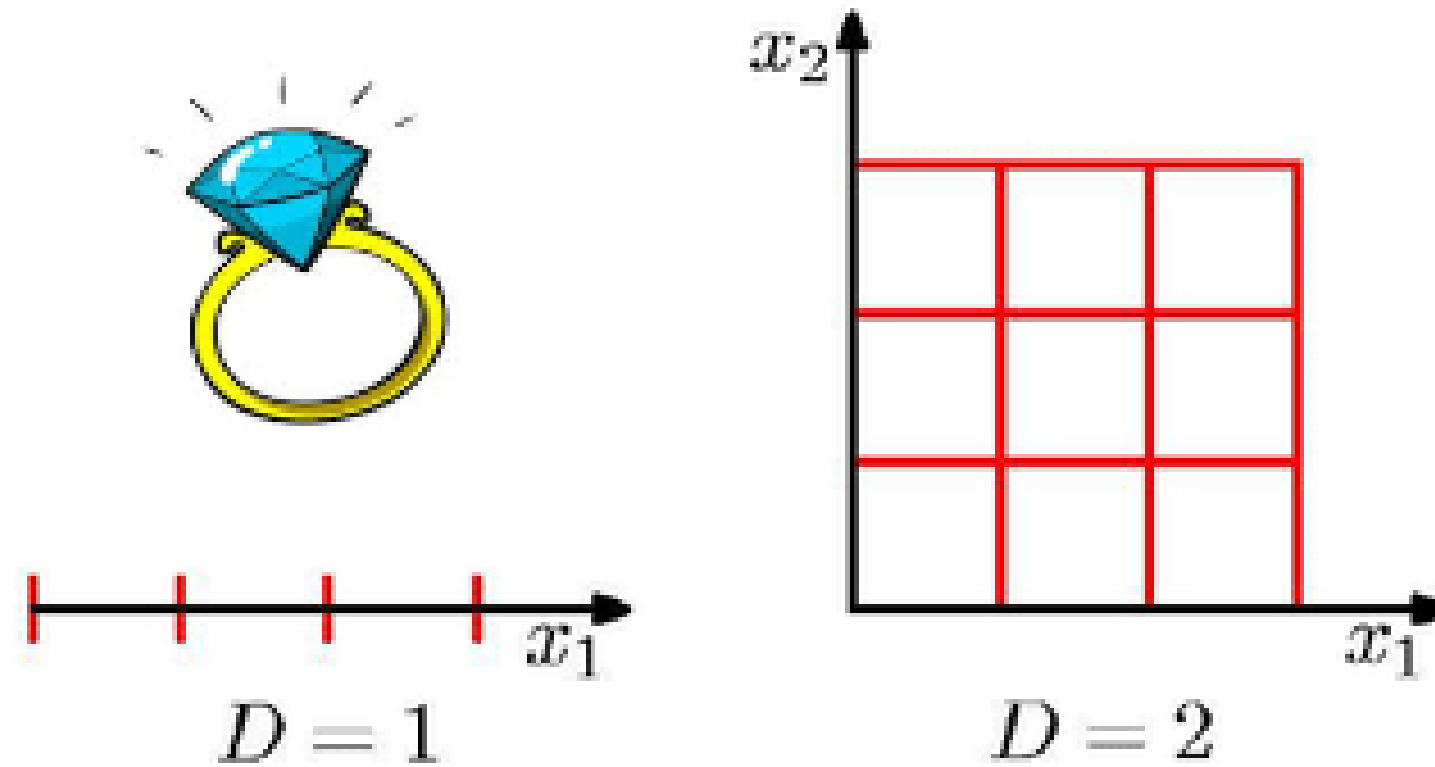


¹ <https://www.visiondummy.com/2014/04/curse-of-dimensionality-affect-classification/>

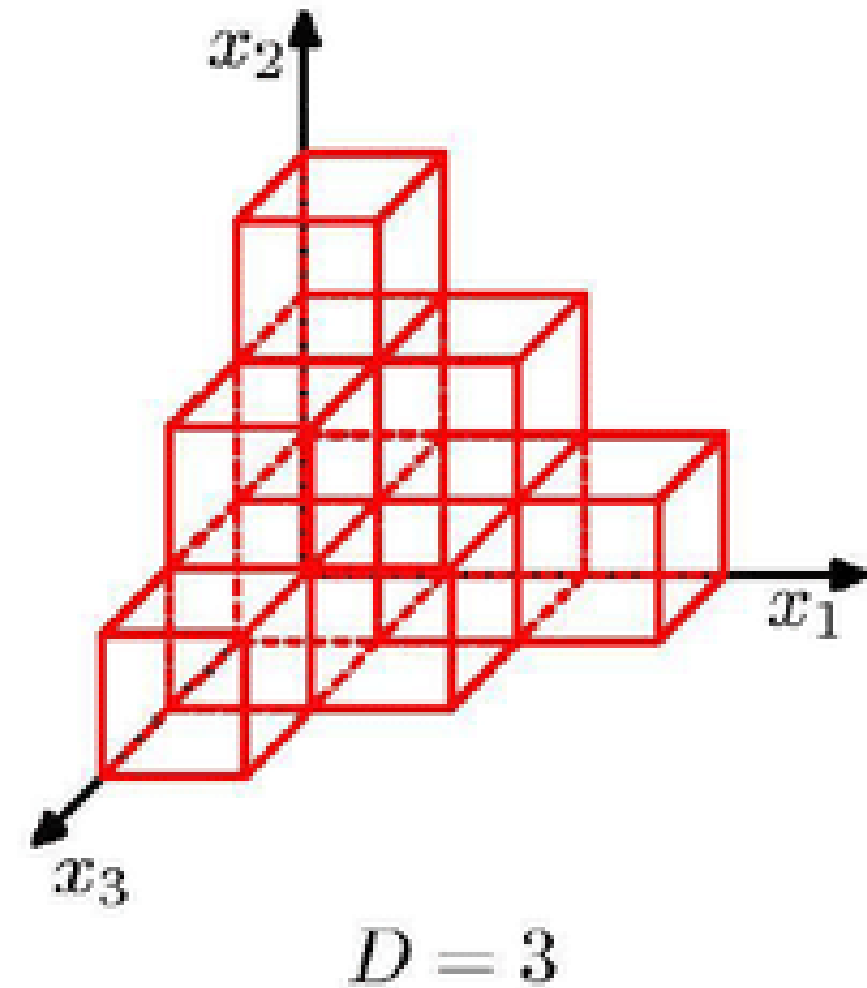
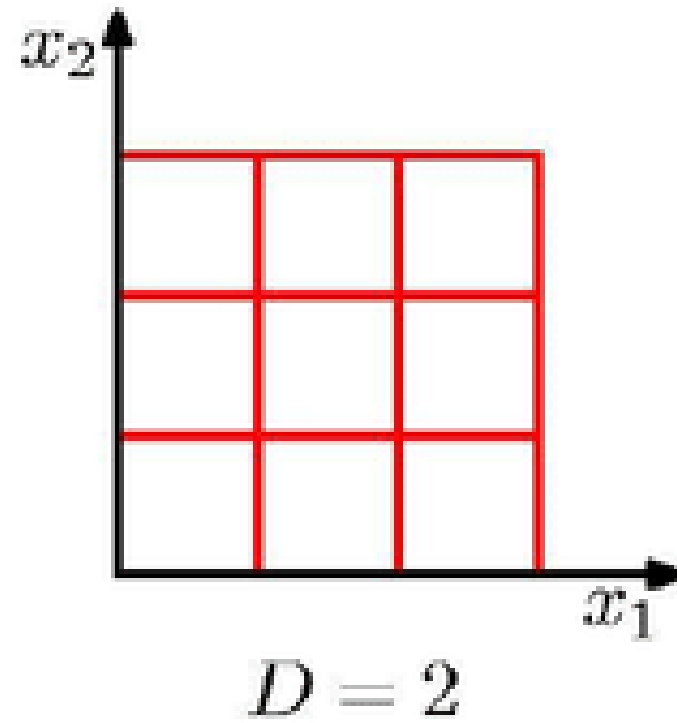
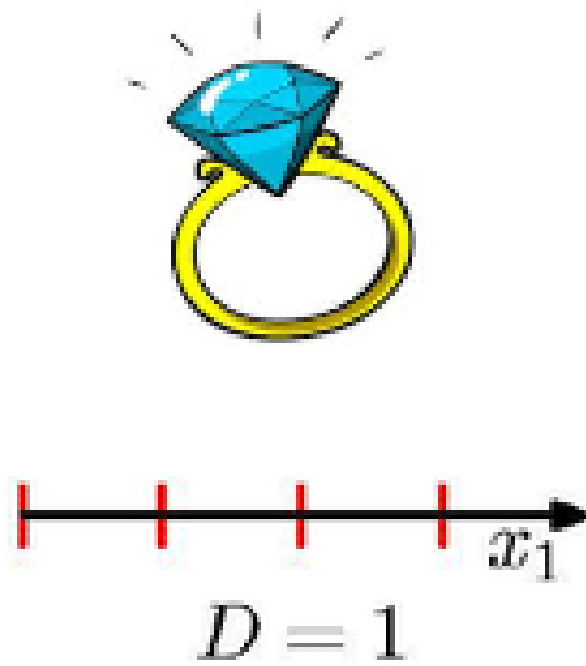
1-D search



2-D search



3-D search

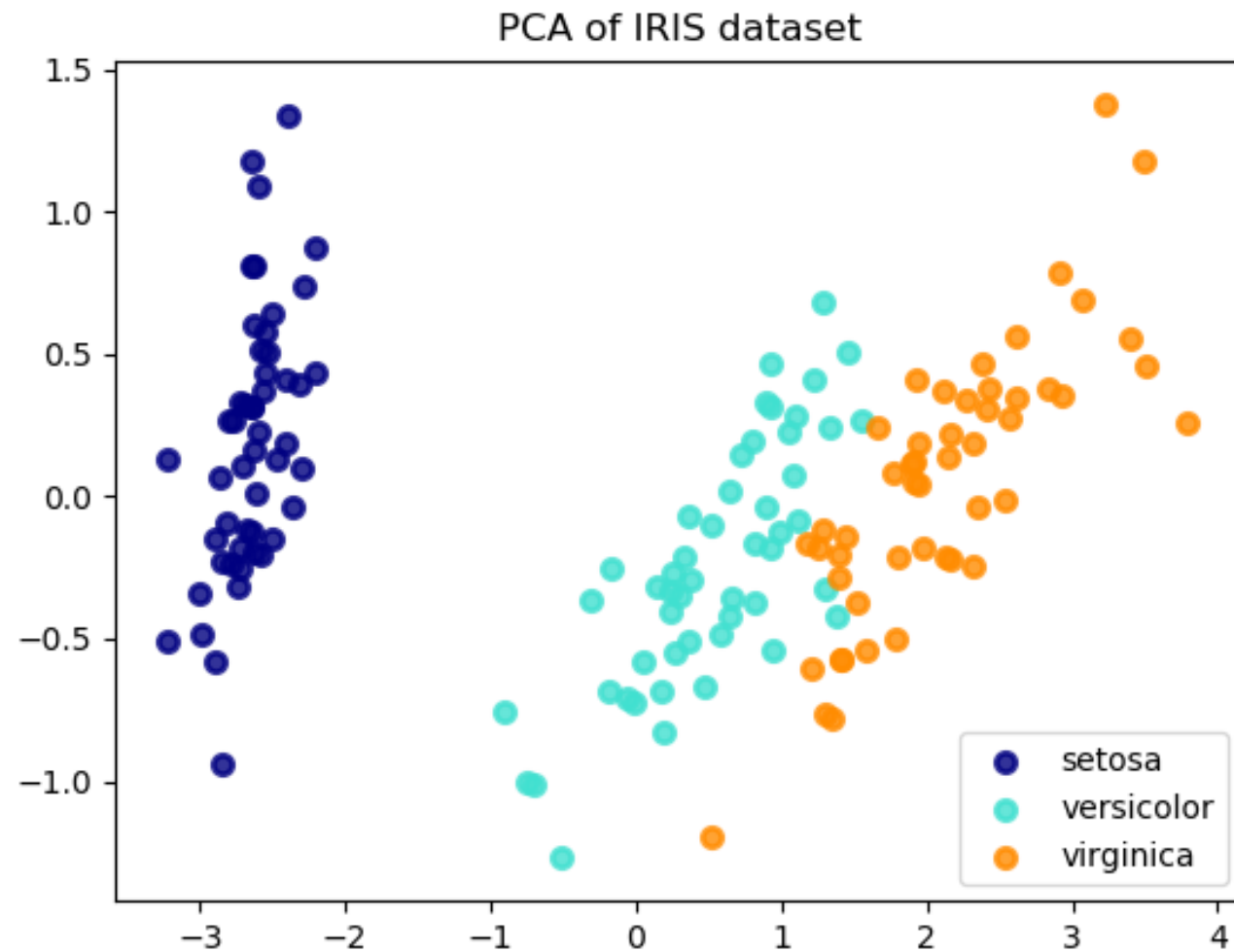


Dimensionality reduction methods

- PCA
- SVD

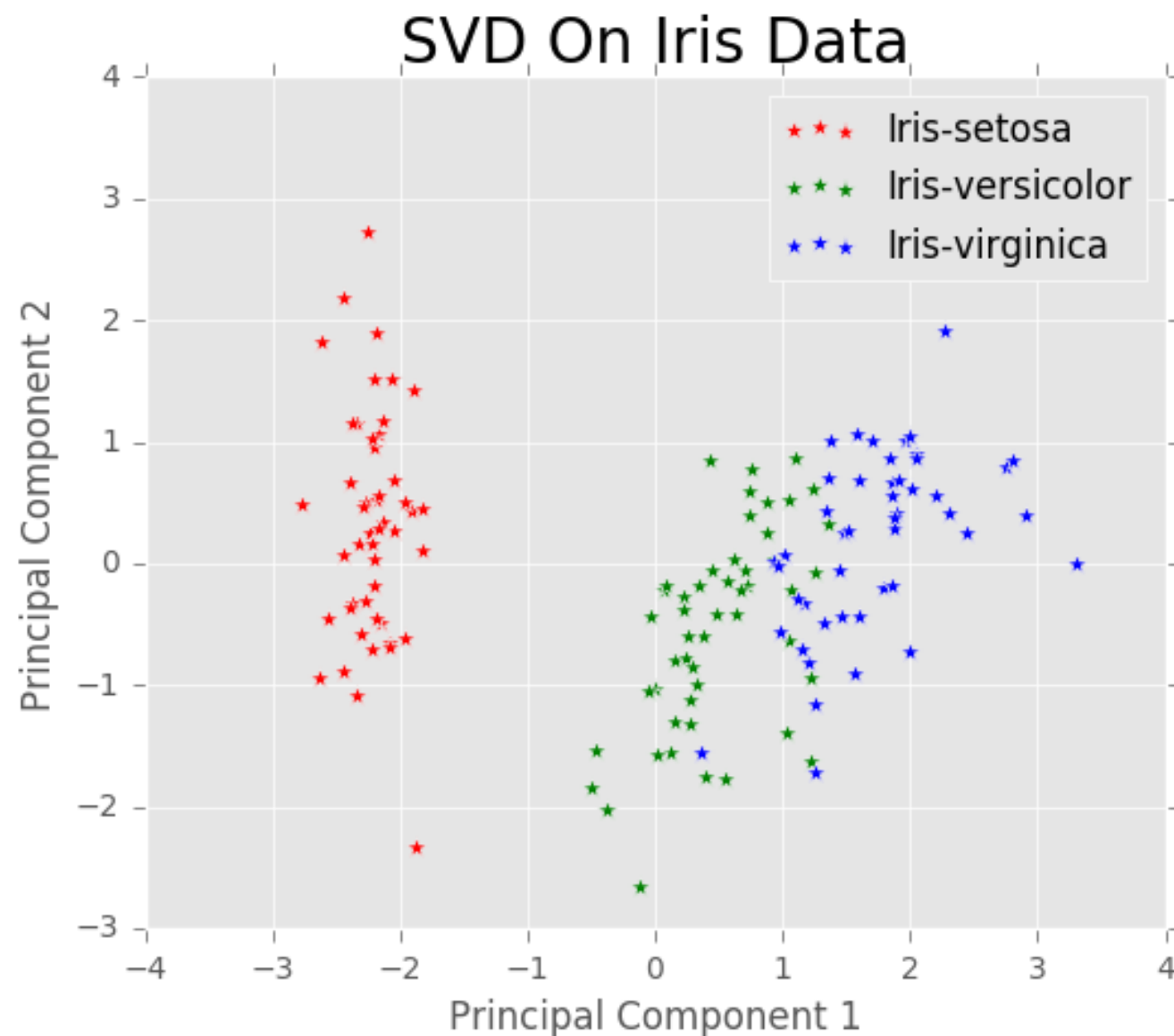
PCA

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



¹ <https://scikit-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-value-decomposition/>

Dimension reduction functions

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

- **Other matrix decomposition algorithms**

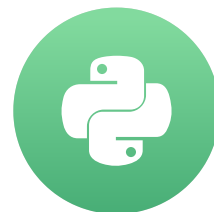
Let's practice!

PRACTICING MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

Dimensionality reduction: visualization techniques

PRACTICING MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

Lisa Stuart
Data Scientist



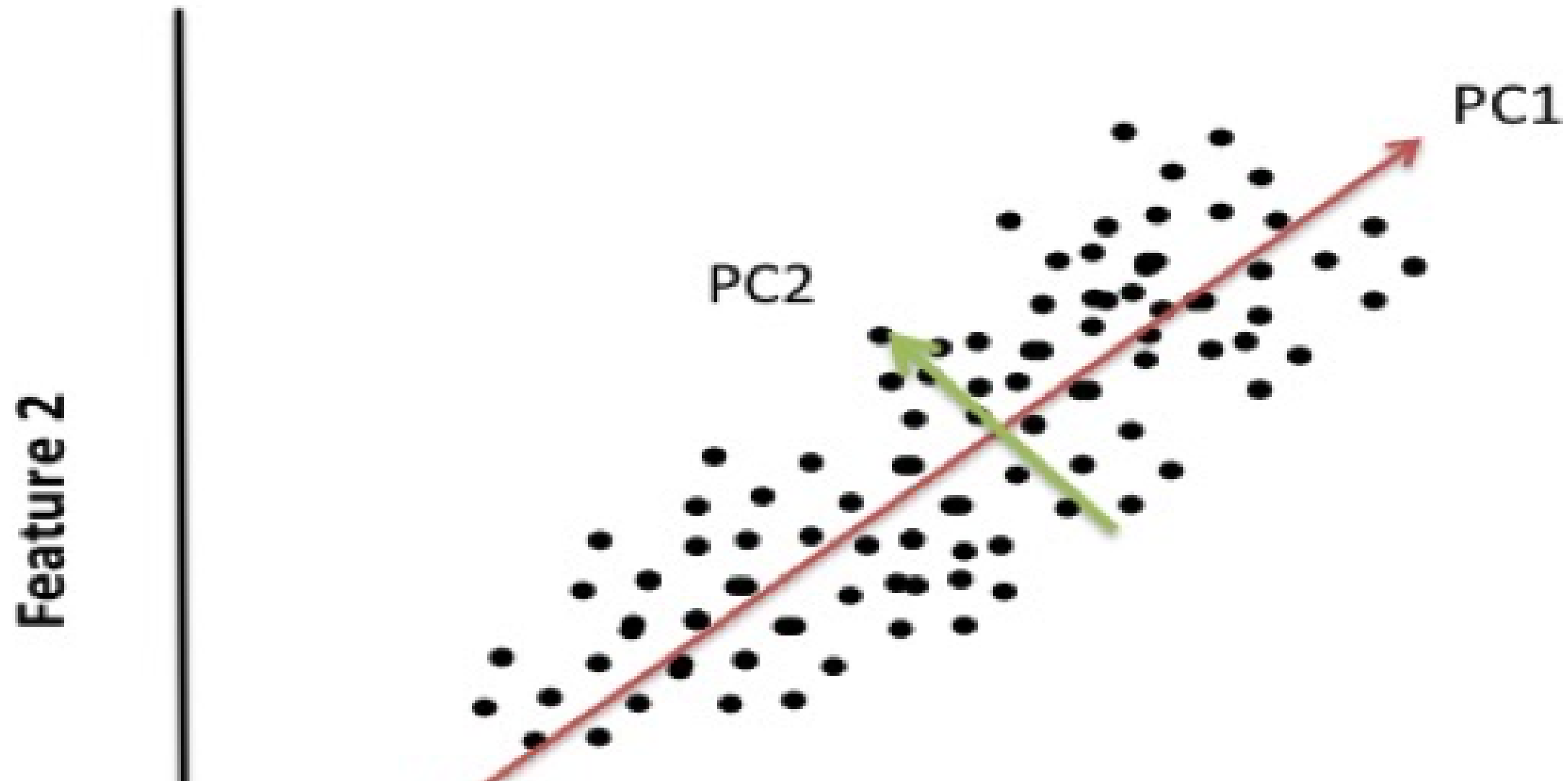
Why dimensionality reduction?

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

Visualization techniques

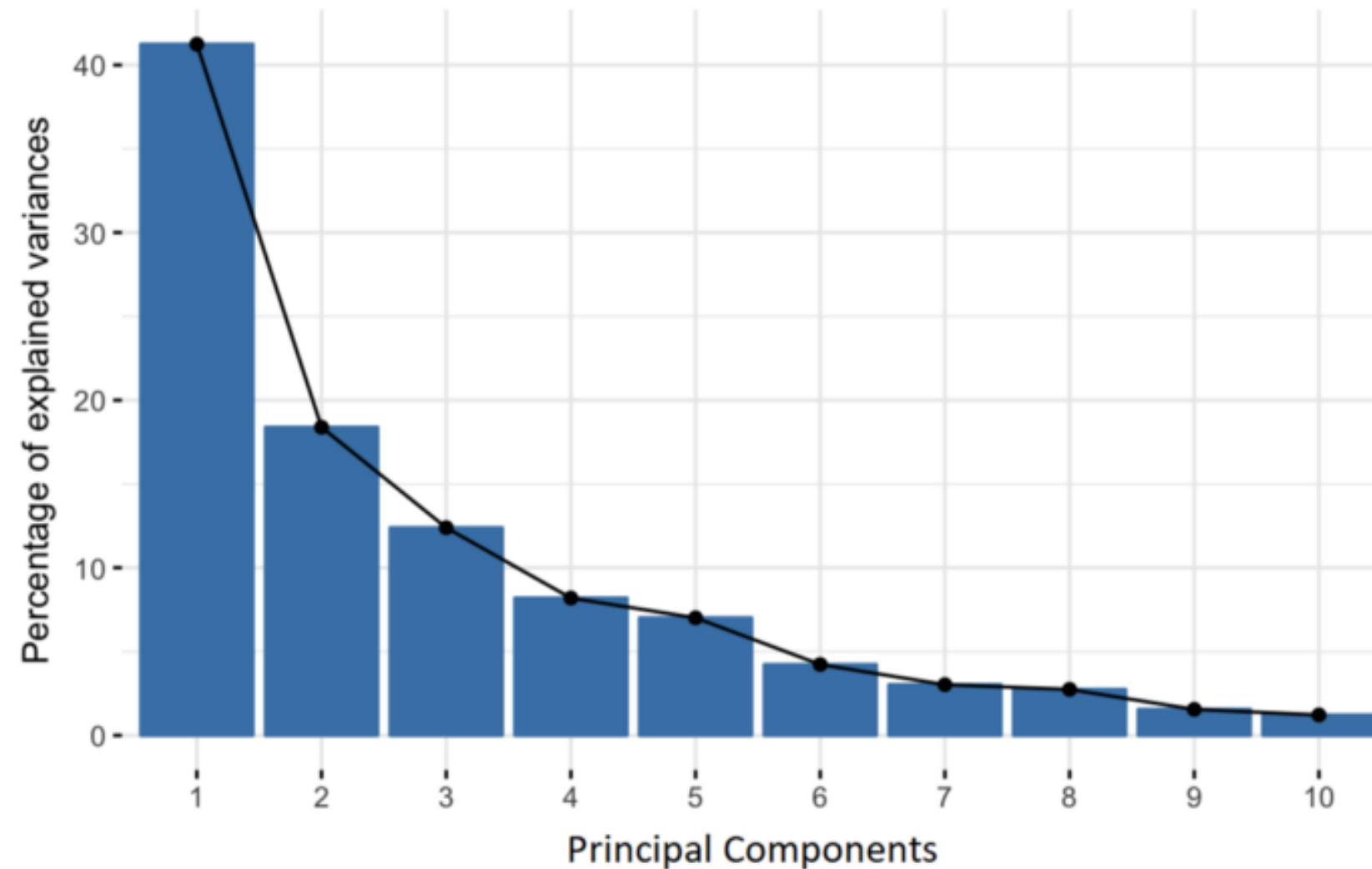
- PCA
- t-SNE

Visualizing with PCA



¹ <https://districtdatalabs.silvrback.com/principal-component-analysis-with-python>

Scree plot



¹ <https://towardsdatascience.com/a-step-by-step-explanation-of-principal-component-analysis-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)

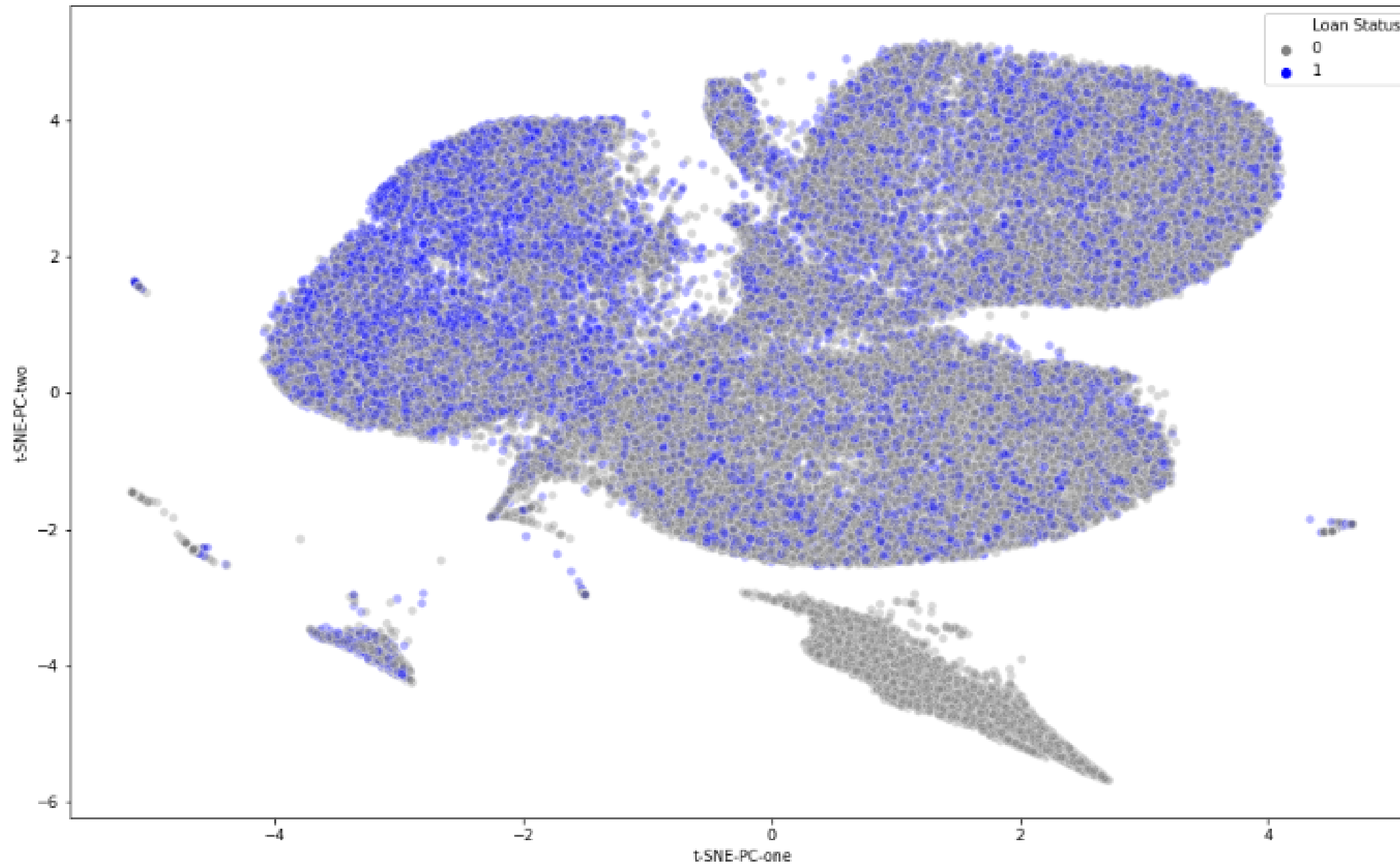
tsne = TSNE(n_components=2, verbose=1, perplexity=40)
tsne_results = tsne.fit_transform(X)

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

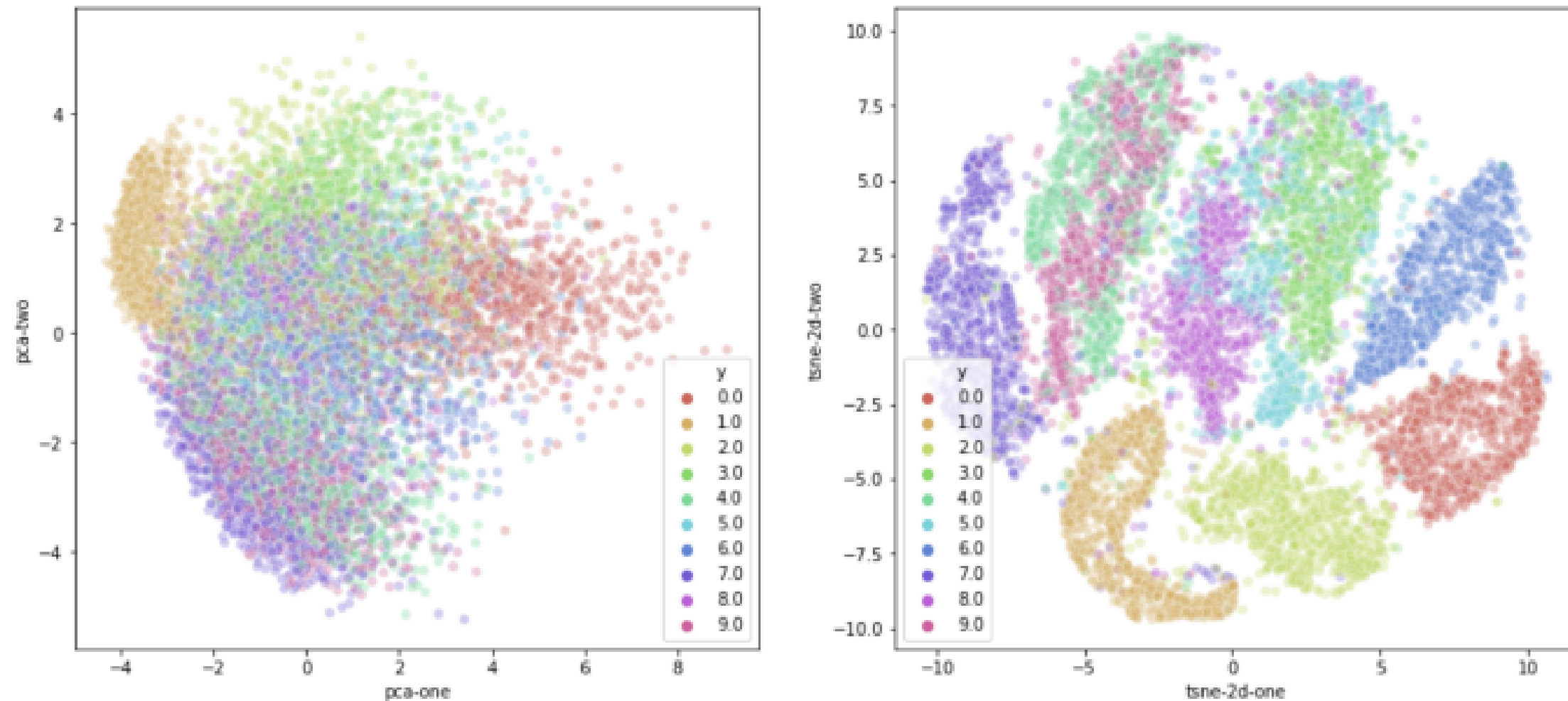
```
# t-sne viz
plt.figure(figsize=(16,10))
sns.scatterplot(
    x="t-SNE-PC-one", y="t-SNE-PC-two",
    hue="Loan Status",
    palette=sns.color_palette(["grey", "blue"]),
    data=loans,
    legend="full",
    alpha=0.3
)
```

¹ <https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html>

Visualizing with t-SNE



PCA vs t-SNE digits data



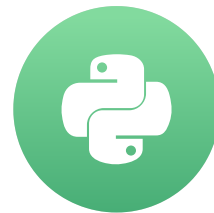
¹ <https://towardsdatascience.com/visualising-high-dimensional-datasets-using-pca-and-t-sne-in-python-8ef87e7915b>

Let's practice!

PRACTICING MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

Clustering analysis: selecting the right clustering algorithm

PRACTICING MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON



Lisa Stuart
Data Scientist

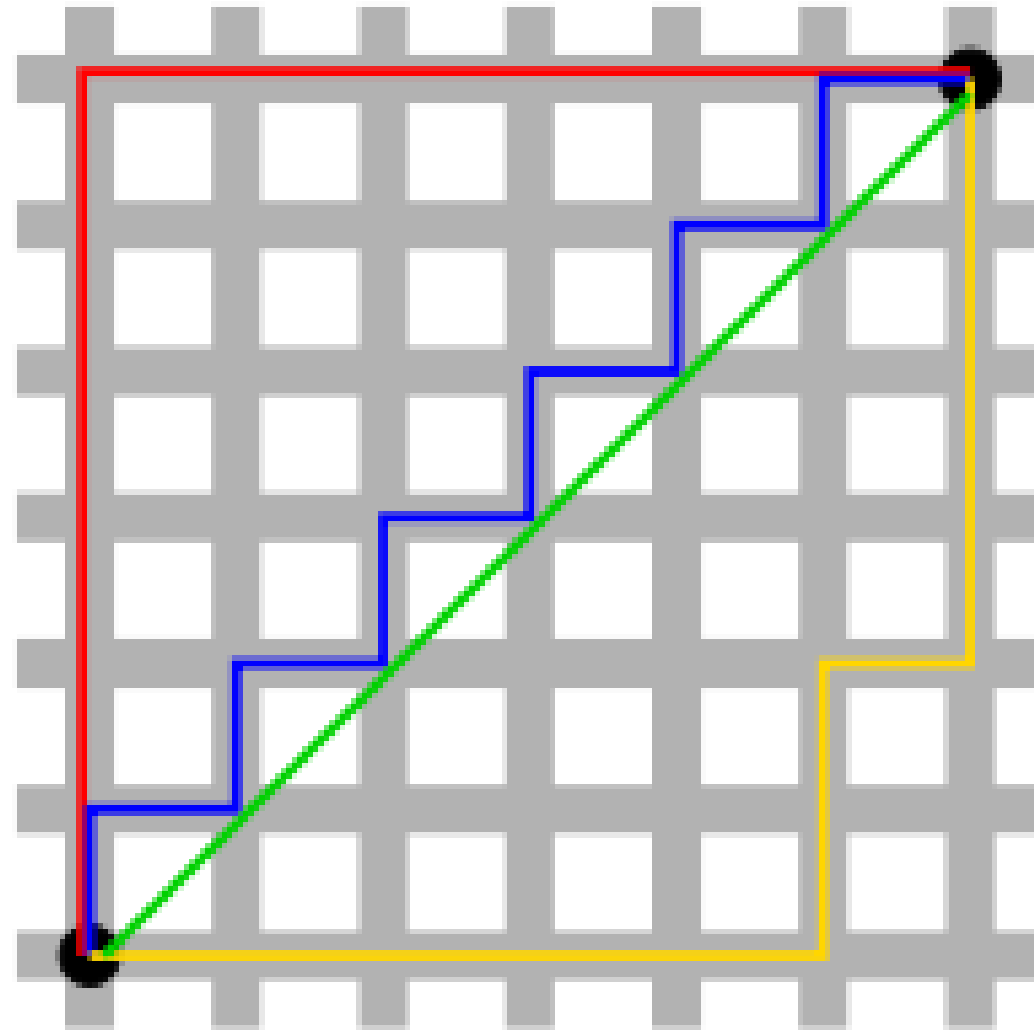
Clustering algorithms

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

Practical applications of clustering

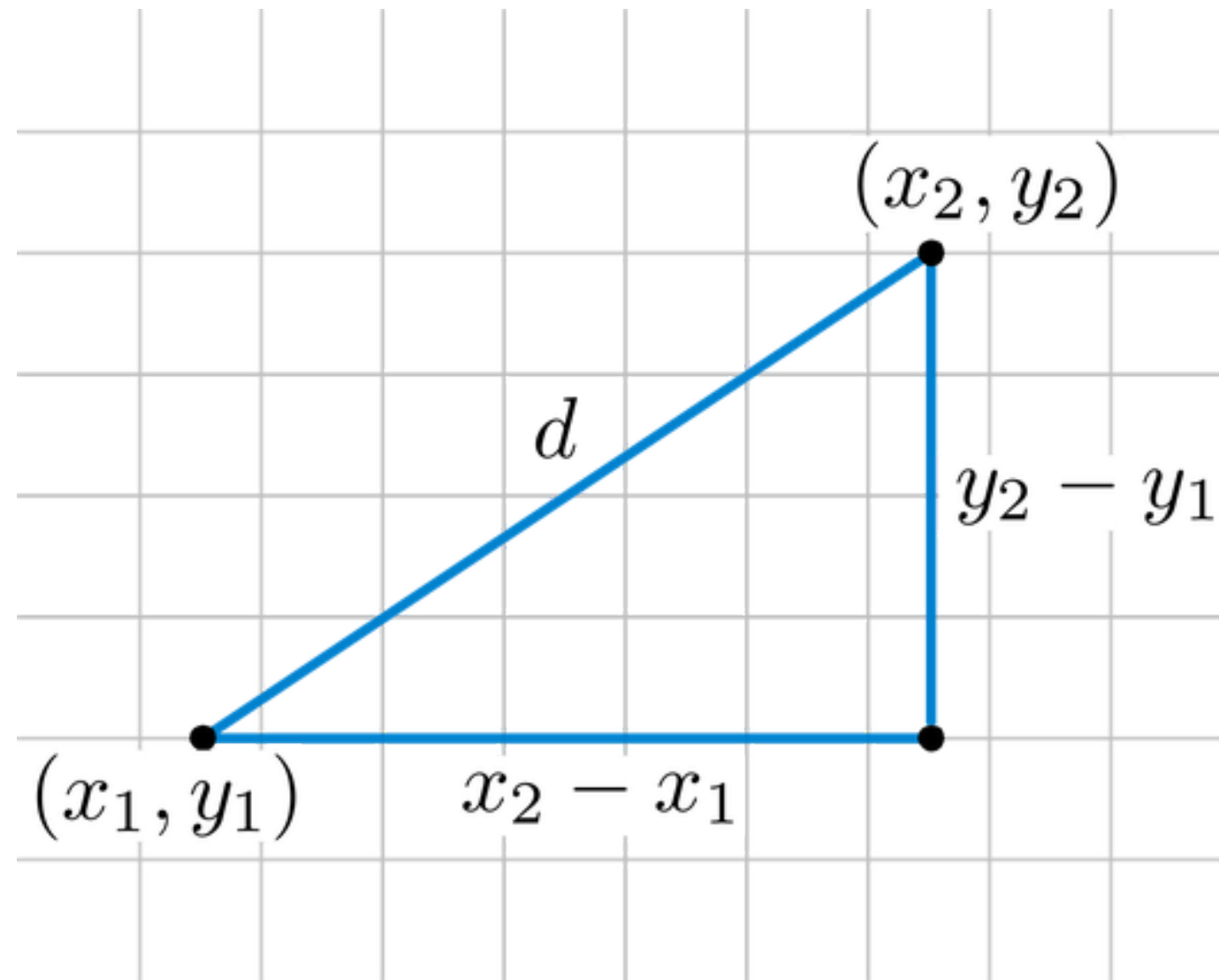
- Customer segmentation
- Document classification
- Insurance/transaction fraud detection
- Image segmentation
- Anomaly detection
- Many more...

Distance metrics: Manhattan (taxicab) distance



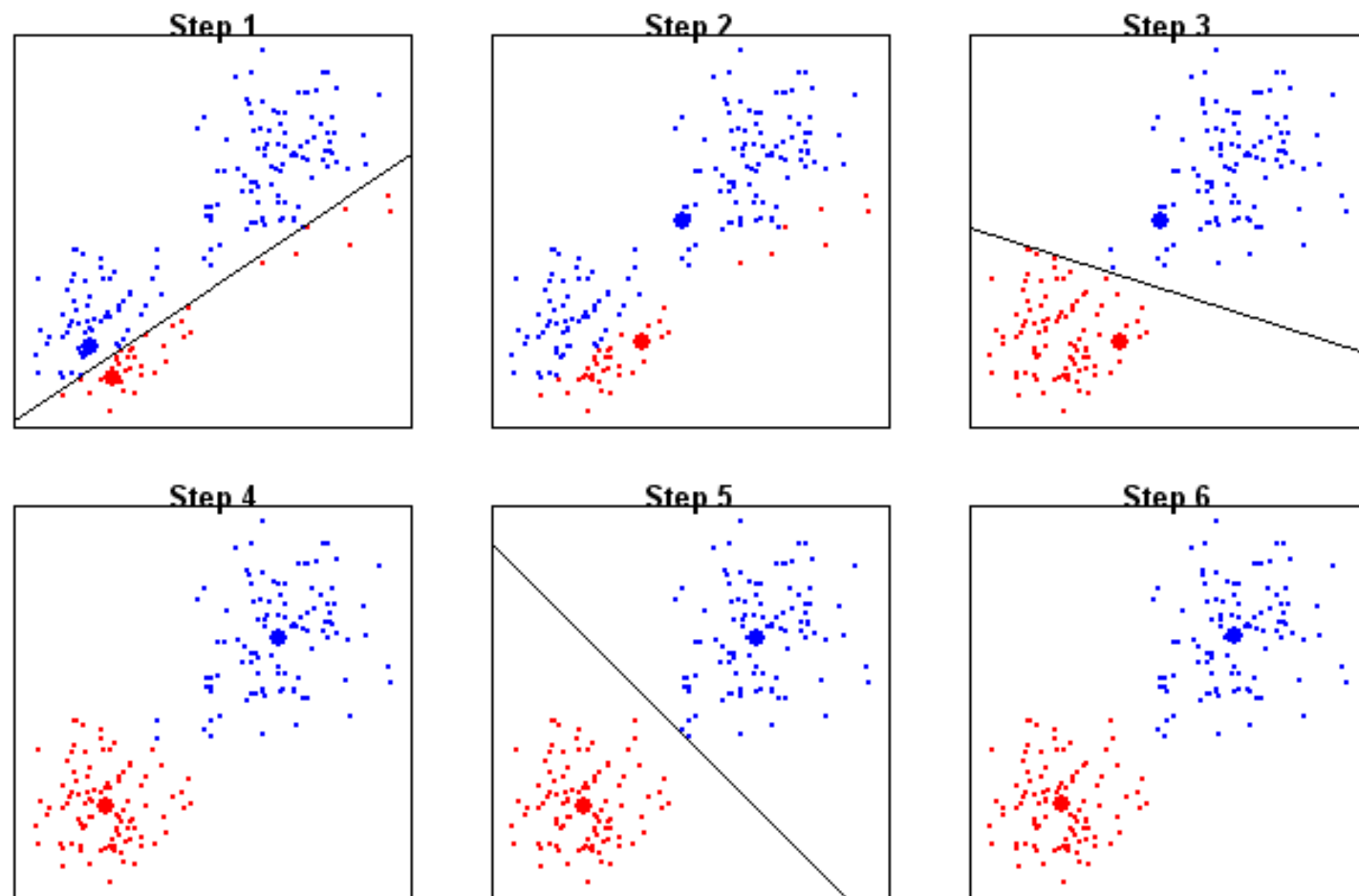
¹ https://en.wikipedia.org/wiki/Taxicab_geometry

Distance metrics: Euclidian distance



¹ <http://rosalind.info/glossary/euclidean> ² distance/

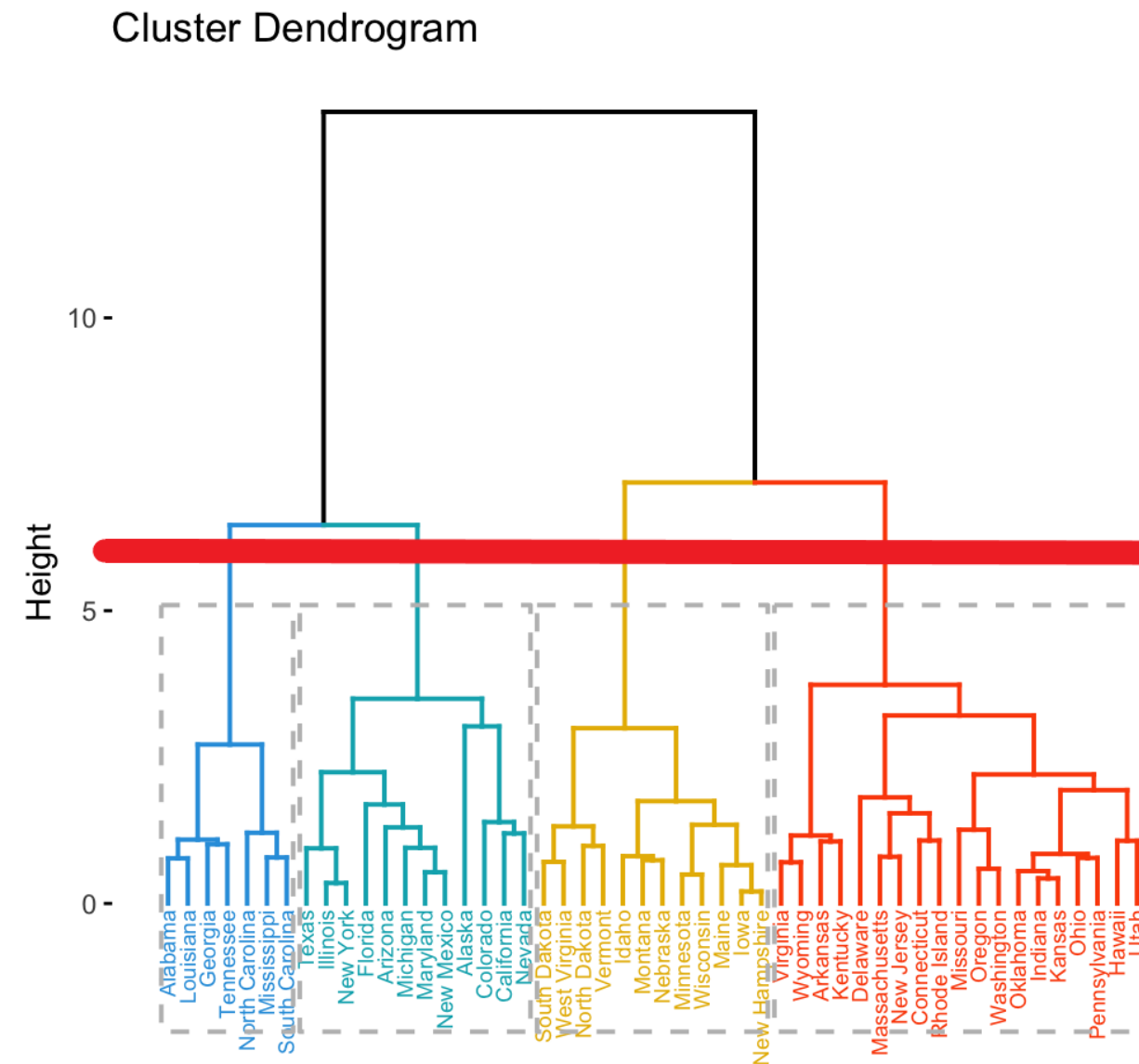
K-means



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-means-clustering/>

Hierarchical agglomerative clustering

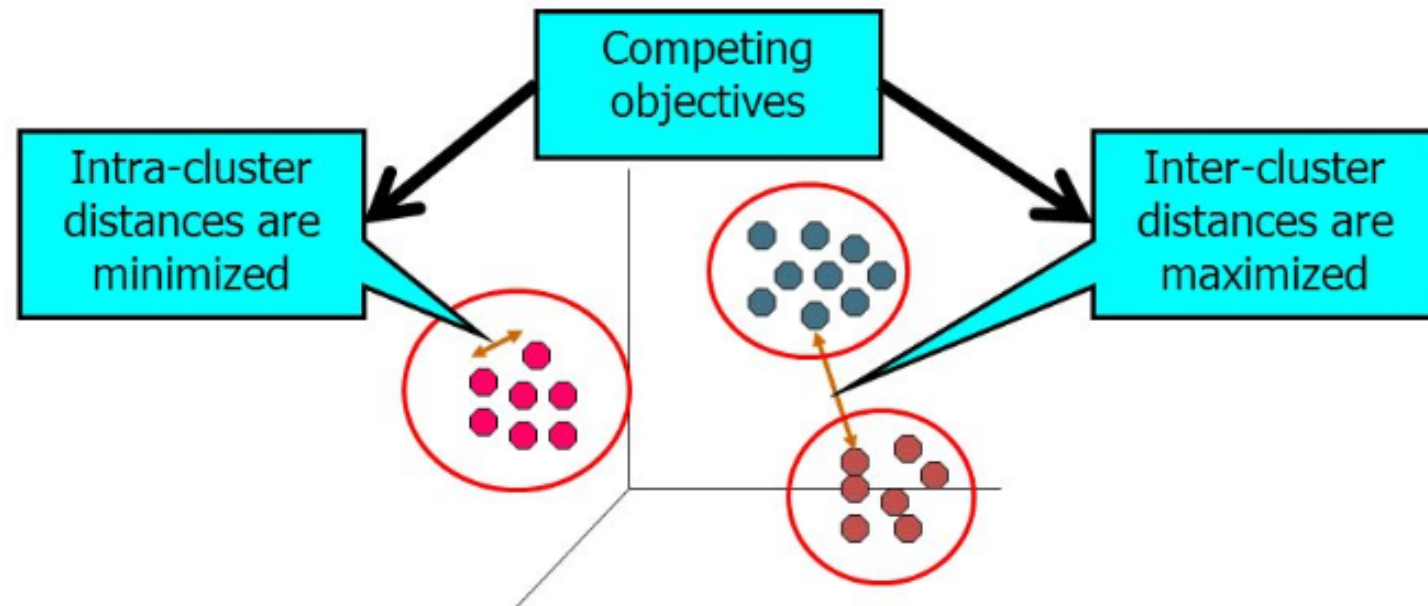


¹ <https://www.datanovia.com/en/lessons/agglomerative> ² hierarchical ³ clustering/

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

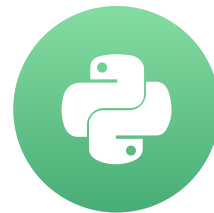
Function/method	returns
<code>sklearn.cluster.Kmeans</code>	K-Means clustering algorithm
<code>sklearn.cluster.AgglomerativeClustering</code>	Agglomerative clustering algorithm
<code>kmeans.inertia_</code>	SS distances of observations to closest cluster center
<code>scipy.cluster.hierarchy</code> as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

Let's practice!

PRACTICING MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON

Clustering analysis: choosing the optimal number of clusters

PRACTICING MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON



Lisa Stuart
Data Scientist

Methods for optimal k

- Silhouette method
- Elbow method

Silhouette coefficient

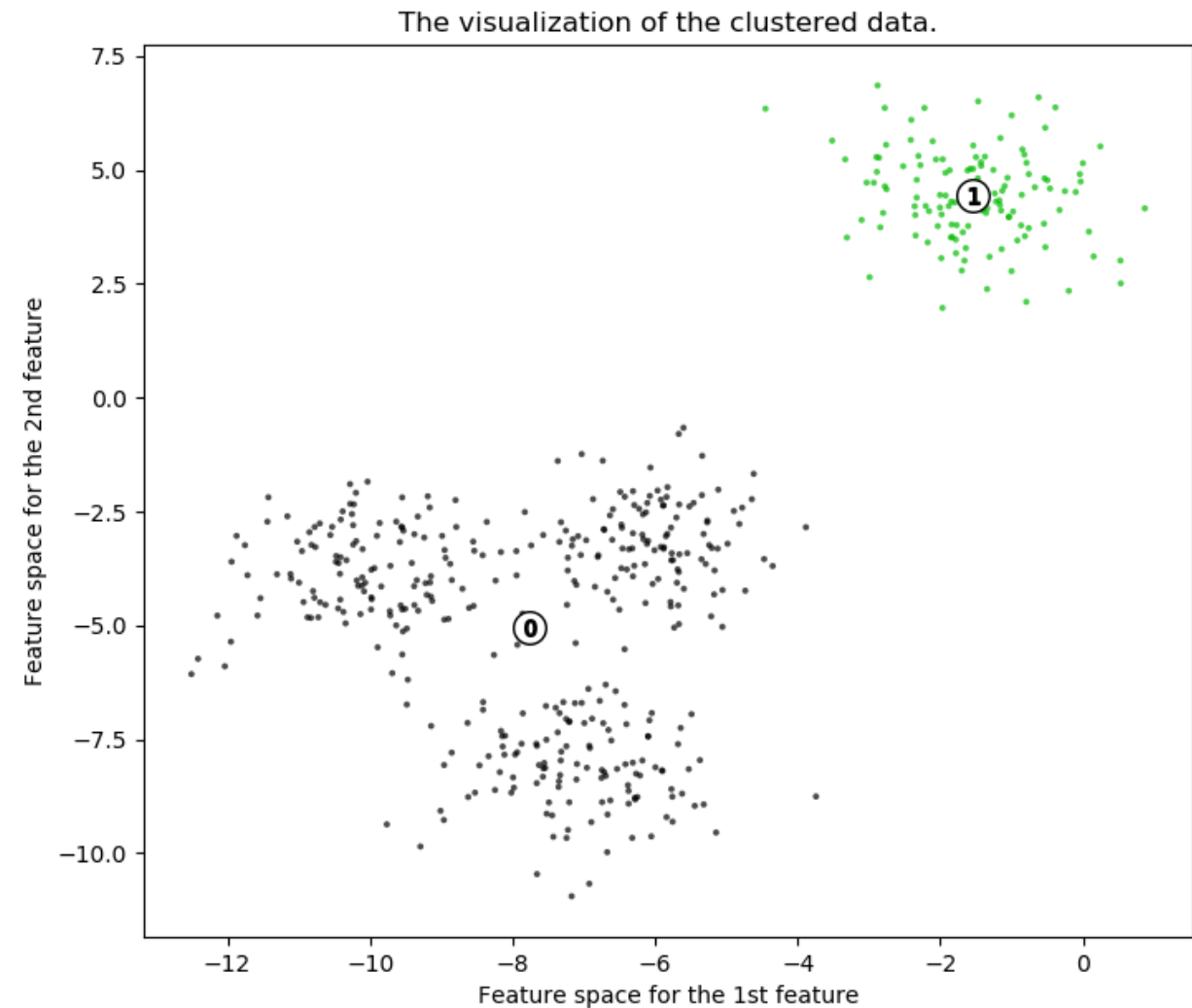
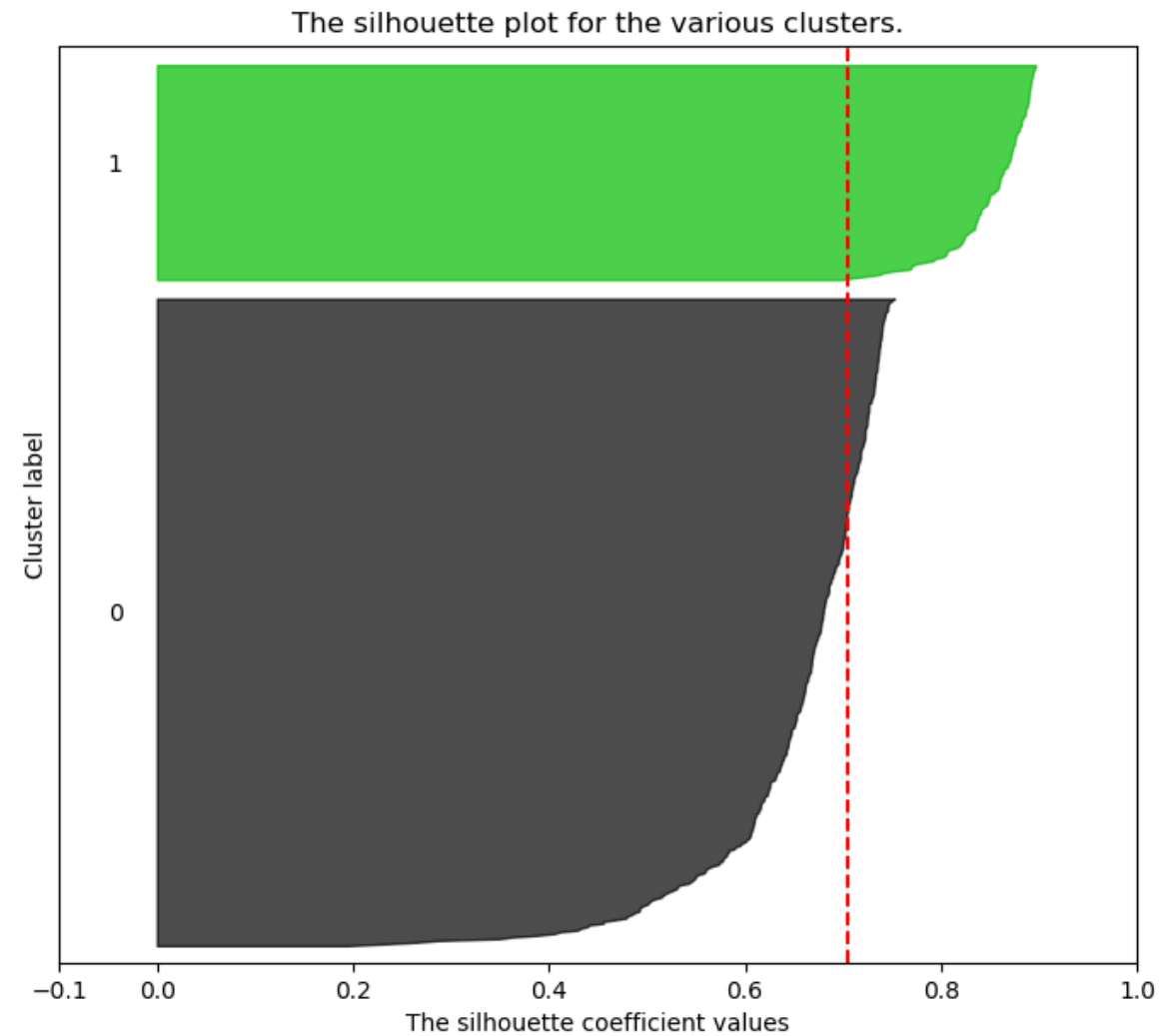
- Composed of 2 scores
 - Mean distance between each observation and all others:
 - in the same cluster
 - in the nearest cluster

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

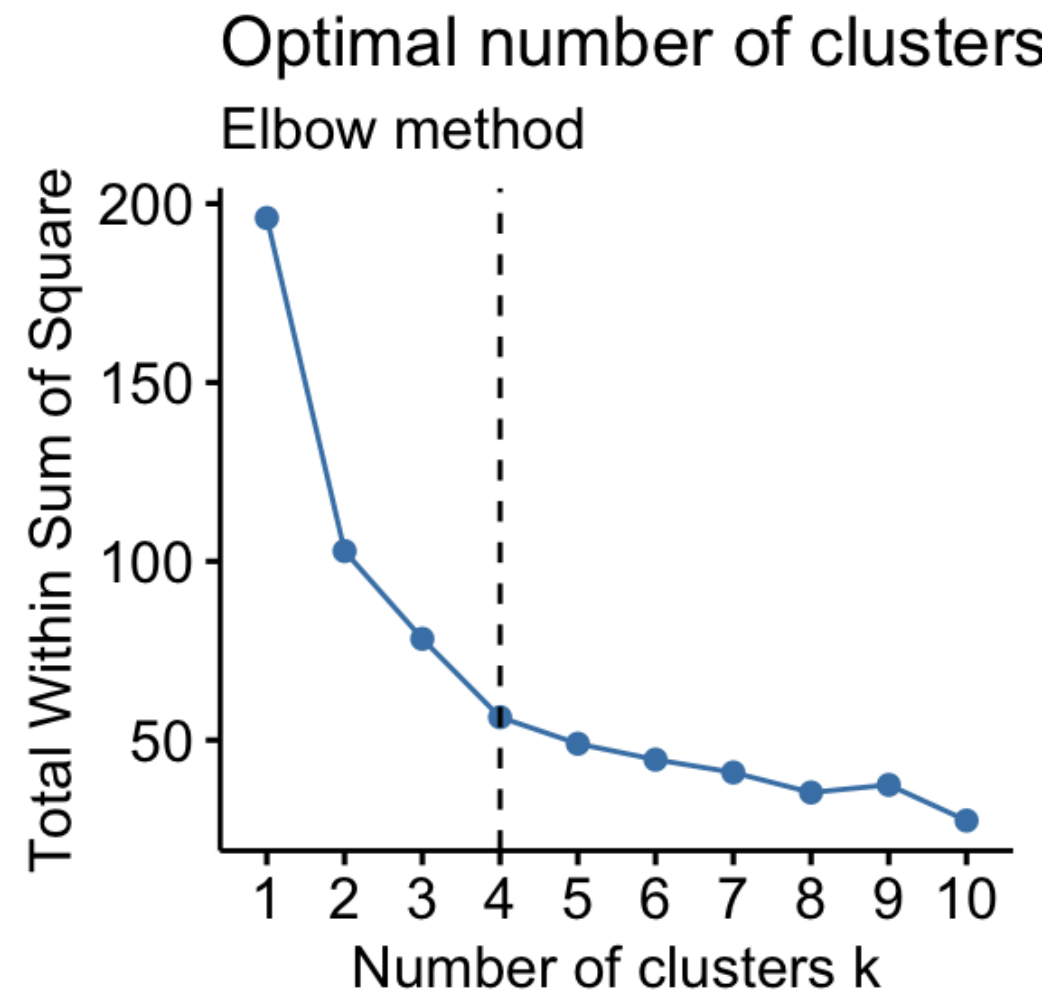
Silhouette score

Silhouette analysis for KMeans clustering on sample data with `n_clusters = 2`



¹ https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html

Elbow method



¹ <https://www.datanovia.com/en/lessons/determining-the-optimal-number-of-clusters-3-must-know-methods/>

Optimal k selection functions

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

Let's practice!

PRACTICING MACHINE LEARNING INTERVIEW QUESTIONS IN PYTHON