Principal Component Analysis for Dimension Reduction

Ryan Miller

Grinnell College
Statistics

Introduction

Last time we covered several common pre-processing steps:

- Scaling
- Transformation
- One-hot encoding

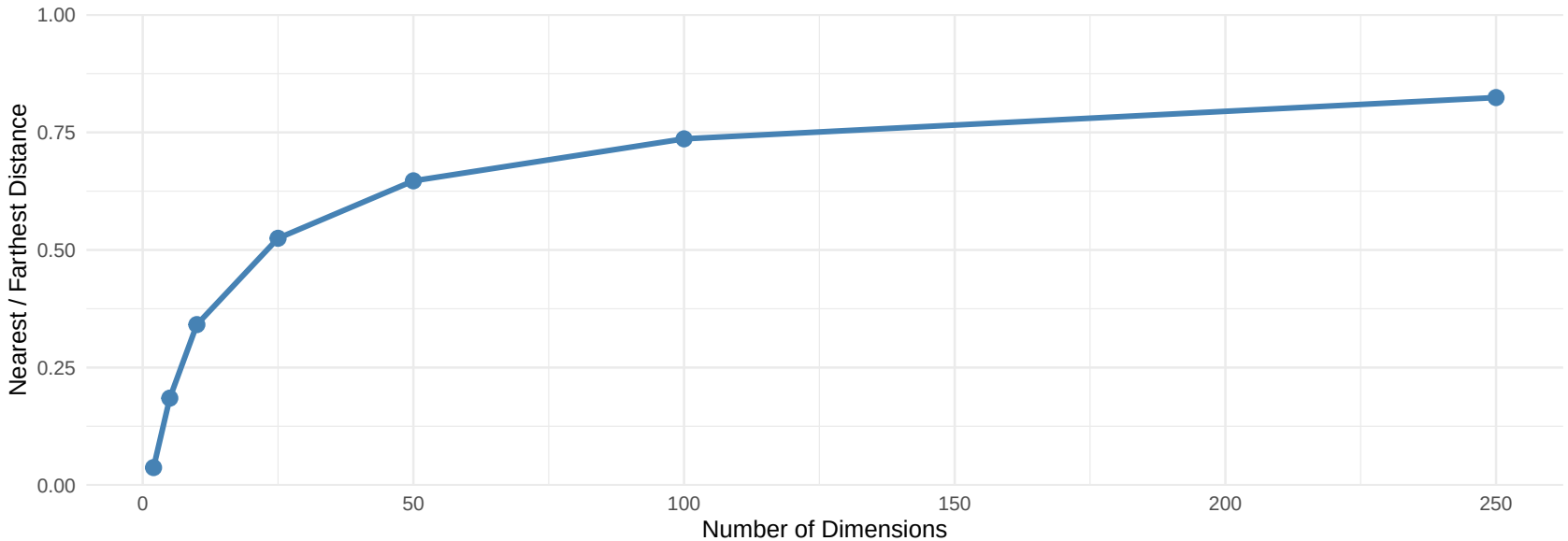
Another type of pre-processing step is *dimension reduction*, or reducing the number of features given to a machine learning model.

Why Worry About Dimensionality?

- Suppose we collect enough training samples to cover 10% of the distributional range of each feature we record
  - In 1-dimension we're covering 10% of space, but in 2-dimensions we're only covering 1% of space ($0.1 \cdot 0.1 = 0.01$)
  - If we extend this to 10-dimensions, 0.1^10 = 0.0000000001, or a minuscule percentage of the feature space

Why Worry about Dimensionality?

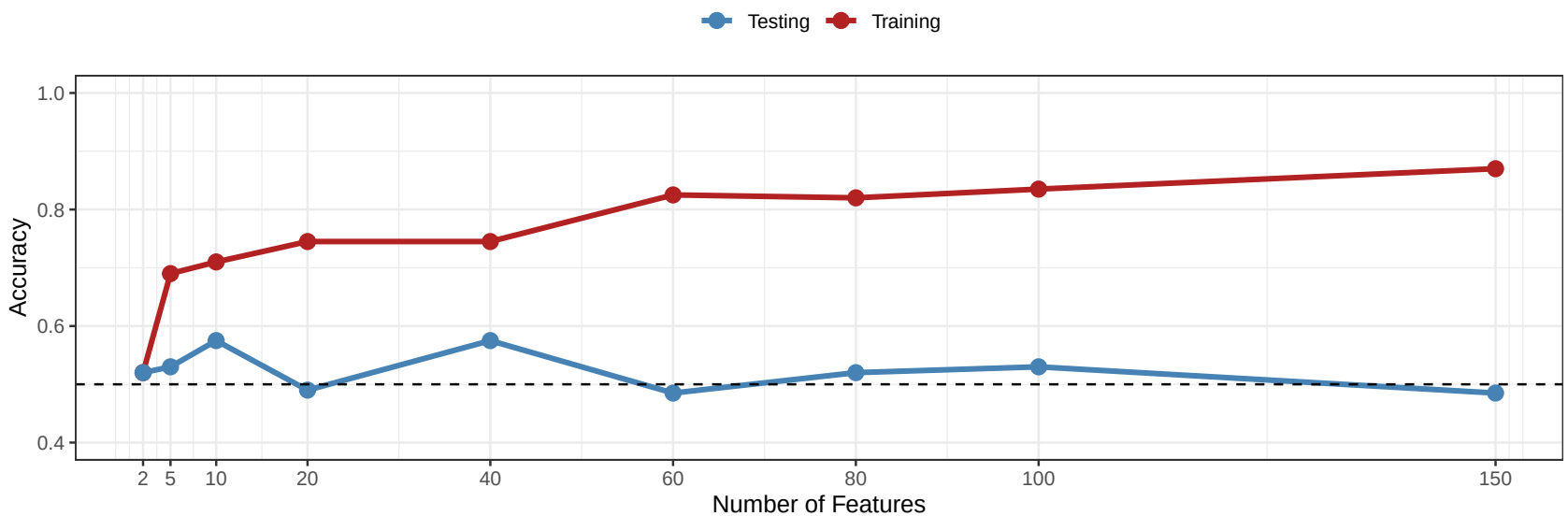
Consider $n = 200$ data-points with each feature value being randomly drawn from a Uniform(0,1) distribution.



As dimensionality increases, the distance between the nearest and furthest neighbor become more similar, reducing the effectiveness of algorithms like KNN.

Why Worry about Dimensionality?

Alternatively, consider features that are all unrelated to a binary outcome but used to train a decision tree:



As the number of noninformative features increases, the decision tree algorithm can find more and more splits that increase purity purely by chance.

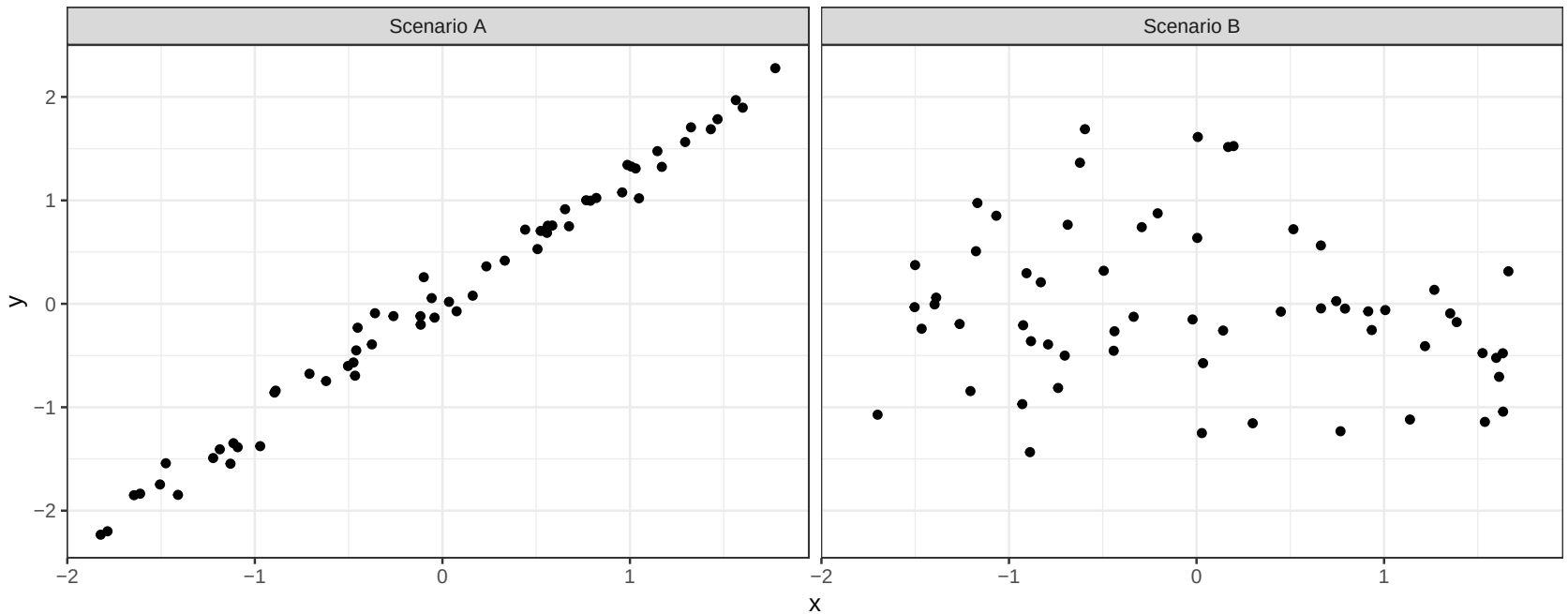
Principal Component Analysis

Principal Component Analysis (PCA) is a dimension reduction approach that creates a new representation of the data using linear transformations of the original features

- Principal components are sequentially constructed
- The first principal component defined as the *direction of maximal variation* within the data
  - The second is the direction of maximal variation that is *uncorrelated* with the first
  - This continues, with the total number of components being the minimum of the n, the sample size, and p, the number of predictors

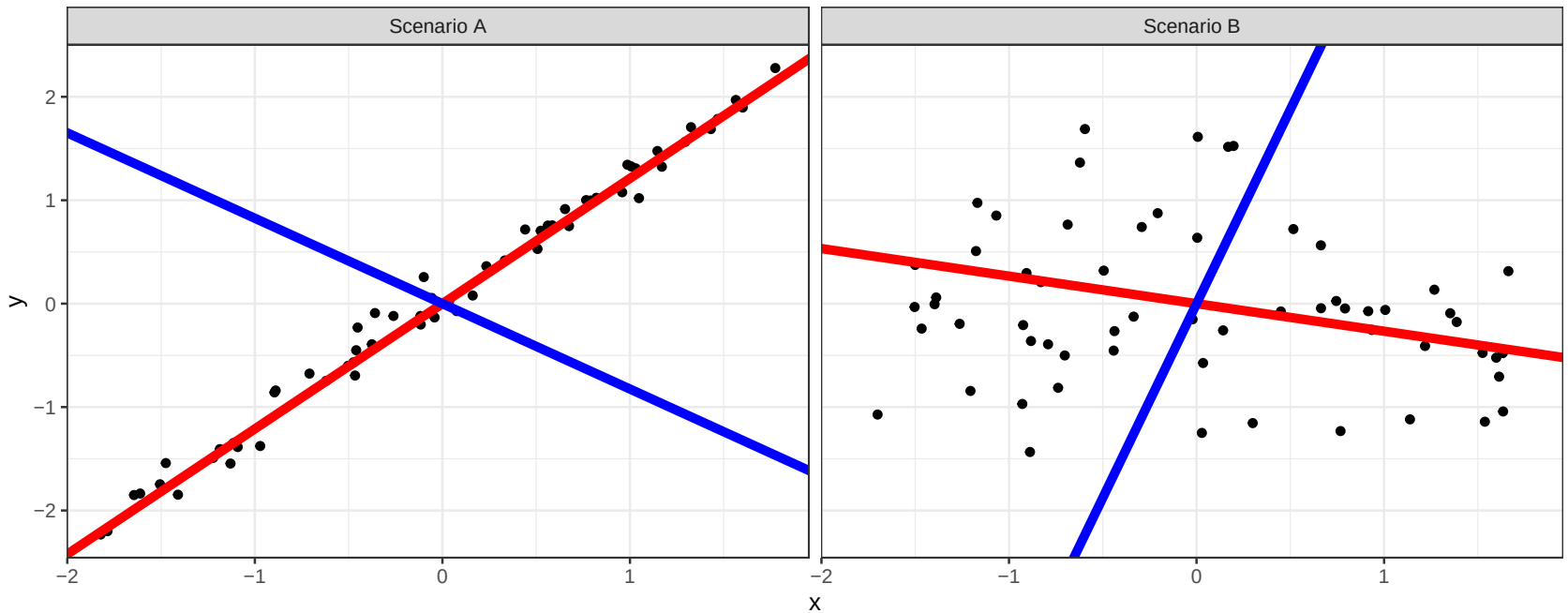
Principal Component Analysis (visual example)

What is the "direction of maximal variation" in each scenario?



Principal Component Analysis (visual example)

Shown below are the first principal component (red) and second principal component (blue)



Principal Component Analysis

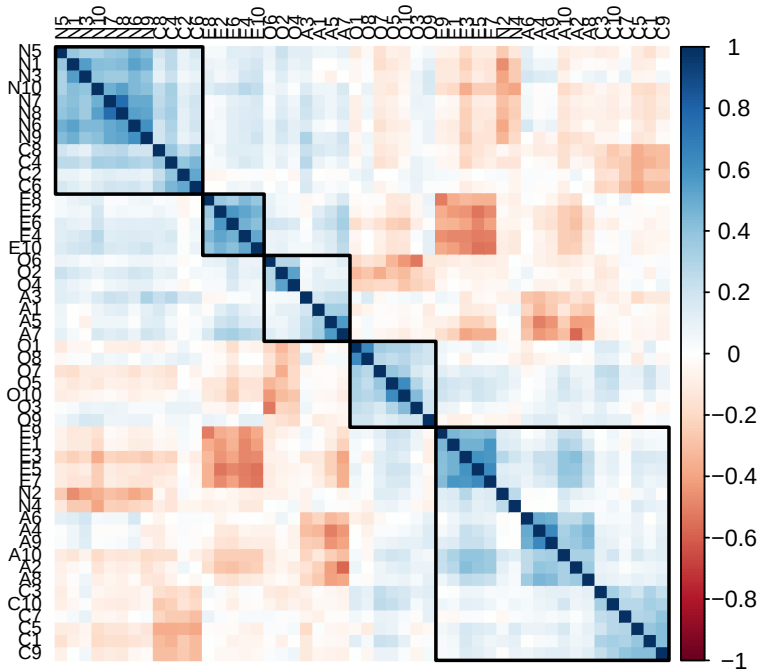
Principal components are linear combinations of the original variables:

$$PC_1 = \phi_{11}X_1 + \phi_{21}X_2 + \ldots + \phi_{p1}X_p$$
$$PC_2 = \phi_{12}X_1 + \phi_{22}X_2 + \ldots + \phi_{p2}X_p$$
$$\ldots$$

- The coefficients, $\phi_{11}, \phi_{21}, \ldots$, are called **loadings**. They describe the contribution of a variable to a principal component.
  - The matrix containing these loadings is sometimes called the **rotation matrix**.

Example - Big 5 Personality Questionnaire

Open Psychometrics hosts a data set containing over 20,000 responses to a 50-item Likert scale (1-5) questionnaire. As you might expect, many items tend to be correlated:



Link: https://openpsychometrics.org/tests/IPIP-BFFM/

Example - Big 5 Personality Questionnaire

Below are the top 10 loadings for PC1 (by absolute magnitude). We can see that the PC1 direction is essentially measuring Extroversion:

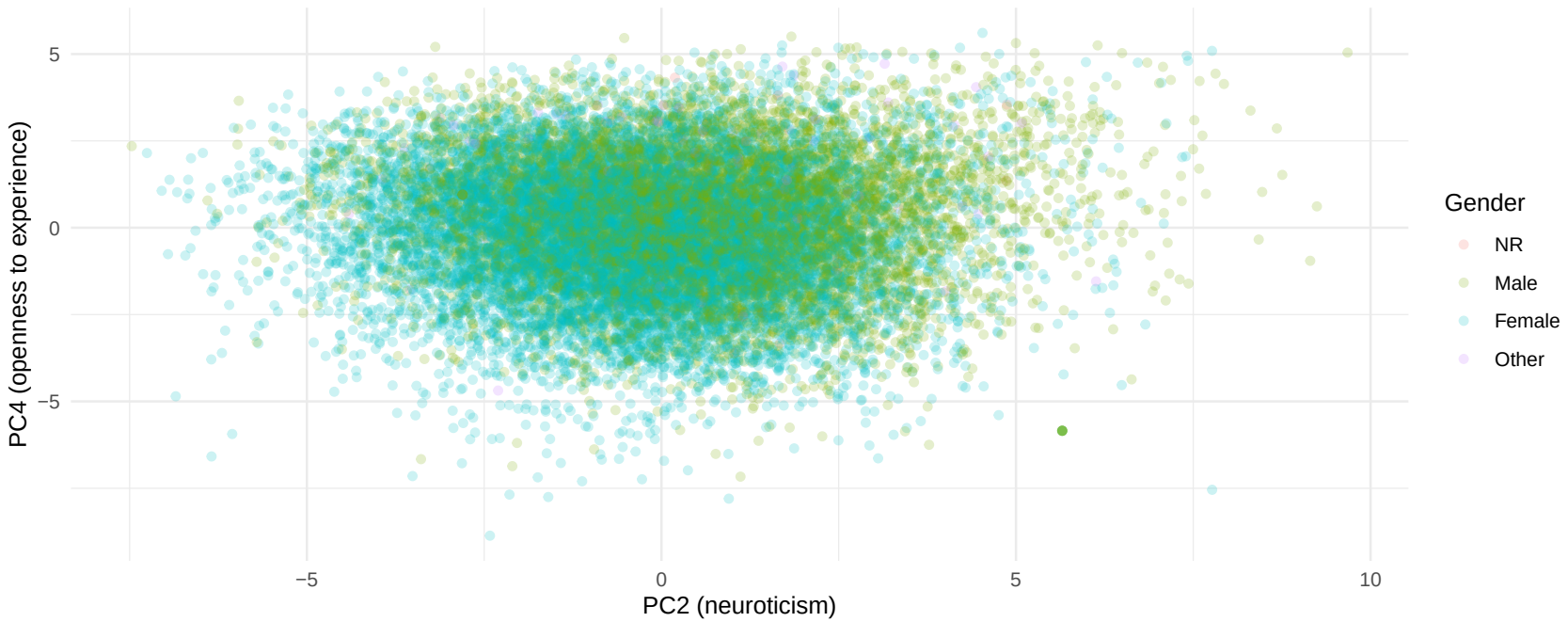
	PC1
E3	-0.2502747
E5	-0.2317793
E7	-0.2206402
E4	0.2057281
E10	0.2017135
E6	0.1985275
N10	0.1945291
A7	0.1875035
A10	-0.1873902
E1	-0.1835065

Example - Big 5 Personality Questionnaire

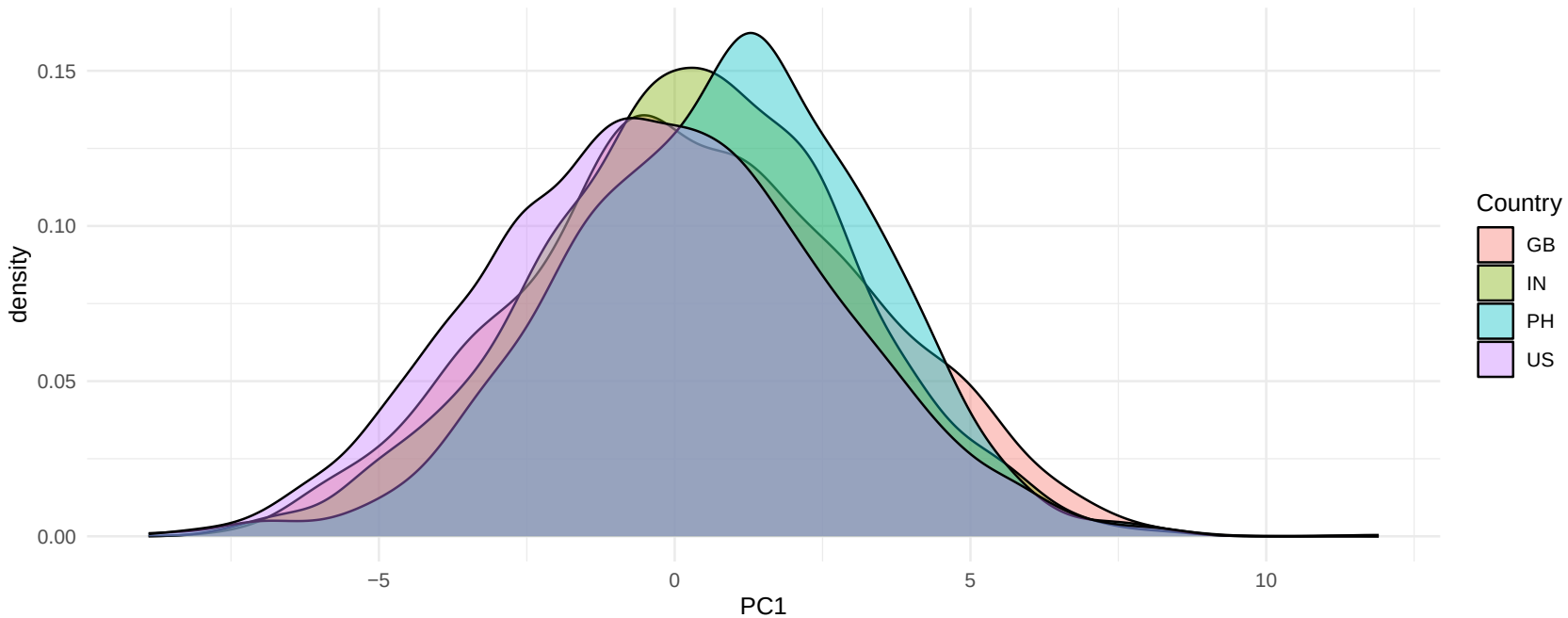
The number of principal components equals the rank of the data matrix. So, we can solve for up to 50 of them for the Big 5 dataset. Shown below is PC4:

	PC4
O1	0.3286096
O10	0.3274160
O8	0.3236670
O5	0.2965985
O2	-0.2911891
O3	0.2881393
O7	0.2692051
O6	-0.2546234
O4	-0.2510374
O9	0.2135816

PCA Visualizations



PCA Visualizations



How Many Components?

PCA is a factorization of the covariance matrix, $\mathbf{C} = \frac{1}{n-1}\mathbf{X}^T\mathbf{X}$, into:

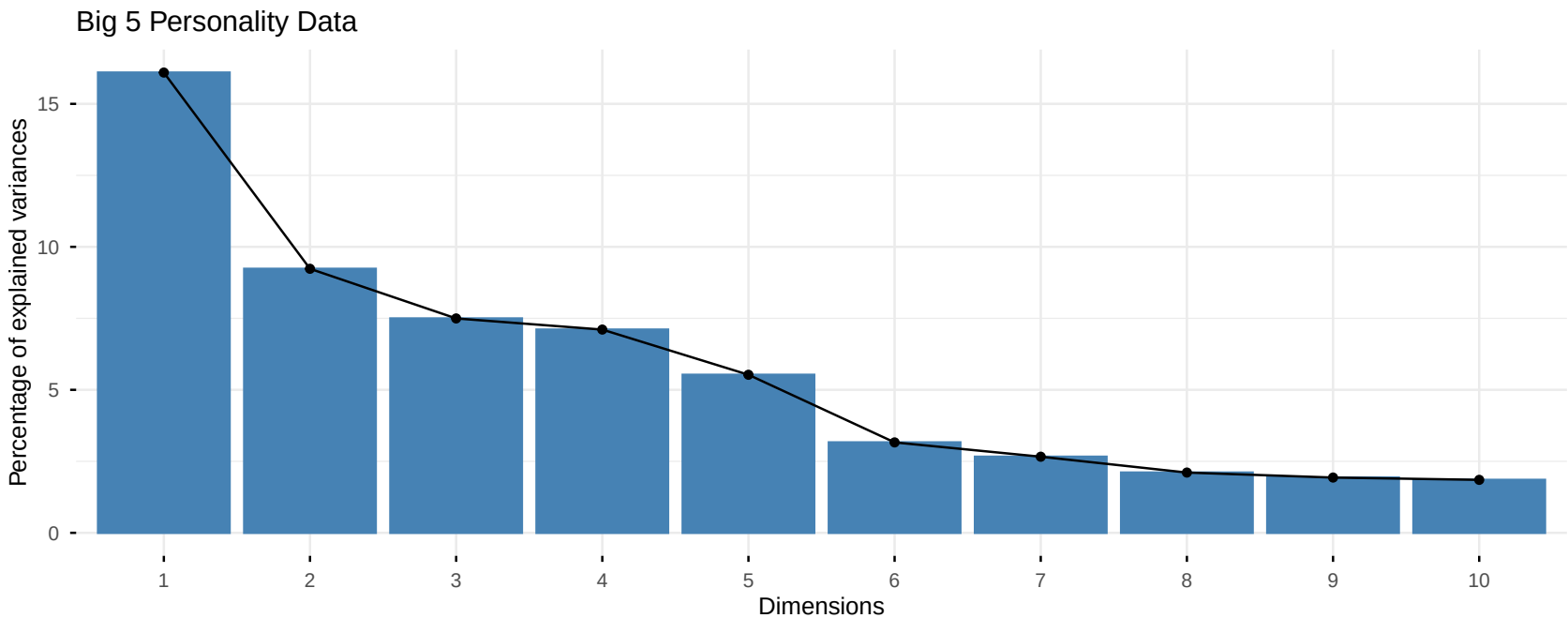
$$\mathbf{C} = \mathbf{V}\mathbf{L}\mathbf{V}^T$$

- $\mathbf{V}$ is a matrix of **eigenvectors**, while $\mathbf{L}$ is a diagonal matrix of **eigenvalues**
- The *columns* of $\mathbf{V}$ contain the *loadings* for each principal component
- The elements of $\mathbf{L}$ relate to the "variance explained" (importance) of each principal component

Note: If the data are standardized, $\mathbf{C}$ is the *correlation matrix*.

How Many Components?

We can use the elements in **L** (the eigenvalues of **C**) to assess the variation captured by each component:



Advice on When to Consider PCA

PCA is *most likely* to benefit model performance when:

- Many features are highly correlated
- The number of features, p, is large relative to the sample size, n
- The modeling approach is sensitive to dimensionality (KNN and others)

PCA is *least likely* to benefit performance when:

- Features are largely independent
- The modeling approach already handles high dimensionality well

What to know for the next quiz?

1. What are principal components? How do they relate to the original features in the data?
2. Under what circumstances is PCA as a pre-processing step most likely to improve machine learning performance?