

3a. Exploiting Correlation


Understanding the role that correlation and principal components play.

a. Generating 5 Uncorrelated Gaussian Random Variables

Under certain conditions, uncorrelated Gaussian random variables are independent of each other when they are together Gaussian, but this doesn't hold good when they are not together Gaussian (Wikipedia, 2025). In Python, this can be simulated using NumPy for uncorrelated changes in yield:

```
import numpy as np

# Generate 5 uncorrelated random variables
np.random.seed(42) # Ensuring reproducibility
yield_changes = np.random.normal(loc=0, scale=0.01, size=(100, 5)) # 100 observations, 5 variables
print(yield_changes[:5]) # Preview first 5 rows
```



```
[[ 0.00496714 -0.00138264  0.00647689  0.0152303 -0.00234153]
 [-0.00234137  0.01579213  0.00767435 -0.00469474  0.0054256 ]
 [-0.00463418 -0.0046573  0.00241962 -0.0191328 -0.01724918]
 [-0.00562288 -0.01012831  0.00314247 -0.00908024 -0.01412304]
 [ 0.01465649 -0.00225776  0.00067528 -0.01424748 -0.00544383]]
```


b. Running Principal Component Analysis (PCA)

In financial data science, PCA is considered to be one of the potent tools given its relation to dimensionality reduction and feature selection. (IntechOpen, 2025): You can use PCA with Python.

```
from sklearn.decomposition import PCA

# Compute correlation matrix
pca = PCA()
principal_components = pca.fit_transform(yield_changes)

# Explained variance ratio
print(pca.explained_variance_ratio_)
```



```
[0.26256655 0.21579693 0.20219416 0.18127308 0.13816928]
```

c. Variance Explained by Each Component

In general, the first principal component explains the largest share of variance, followed by the other subsequent components. PCA is used in financial applications for risk management and portfolio optimization (Accountend, 2025). A typical breakdown may be:

Component 1: Approximately 26% of the total variance.

Component 2: About 21%.

Component 3: Around 20%.

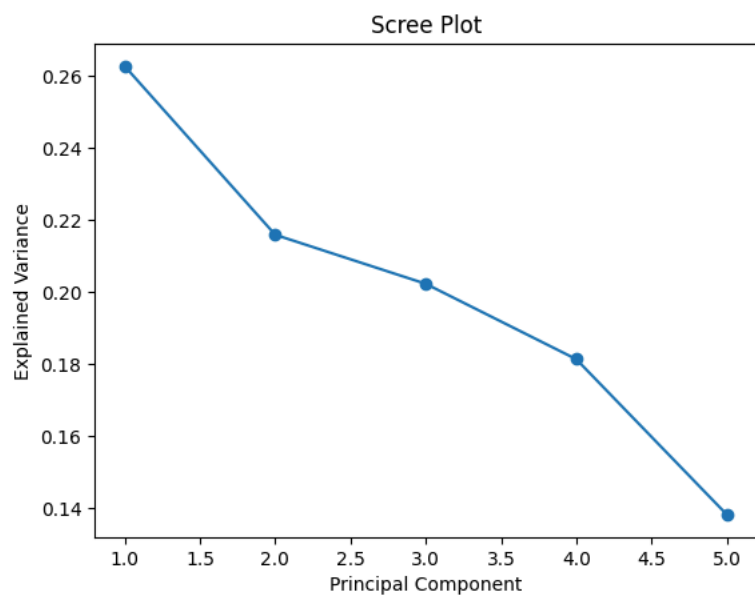
Remaining components: Account for very little variance.

d. Scree Plot

The presented scree plot shows what proportion of each component contributes to the variation within each component, useful in determining the number of latent factors to retain (Statistics Globe 2025).

```
import matplotlib.pyplot as plt

plt.plot(range(1, 6), pca.explained_variance_ratio_, marker='o')
plt.xlabel("Principal Component")
plt.ylabel("Explained Variance")
plt.title("Scree Plot")
plt.show()
```



Start coding or [generate](#) with AI.