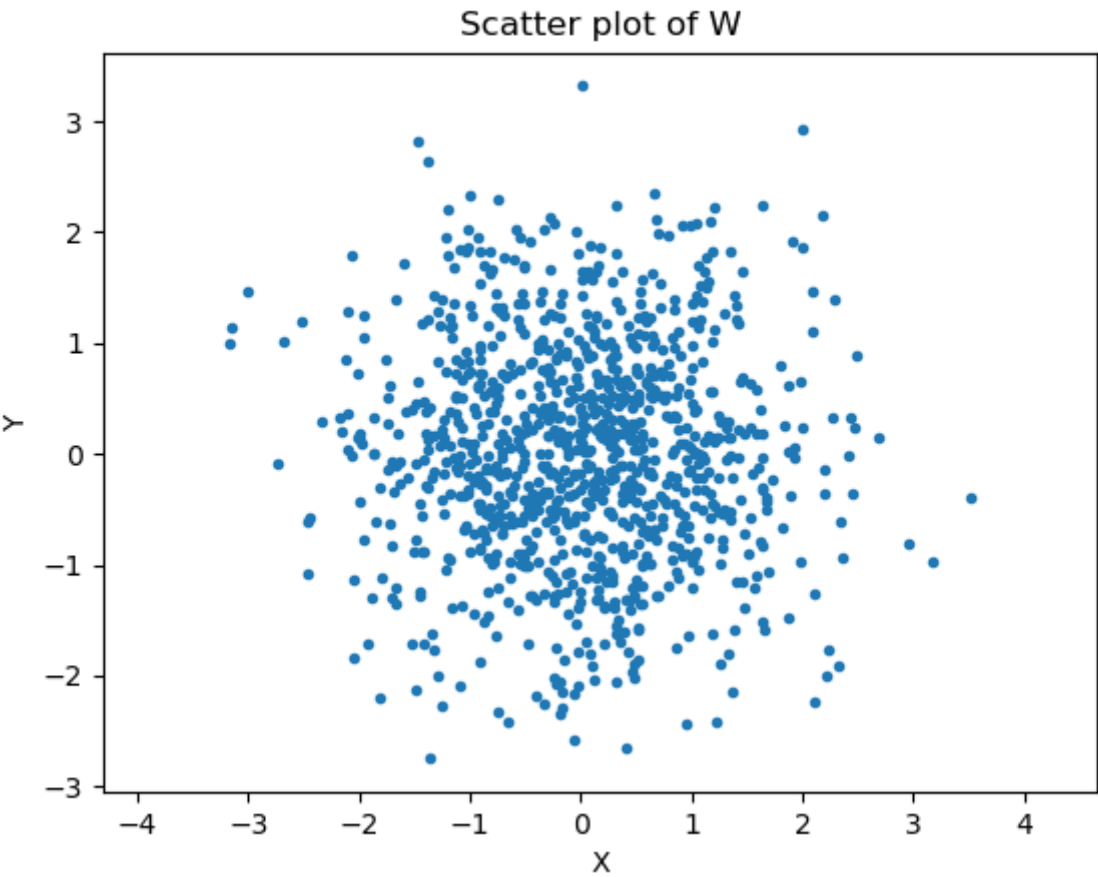
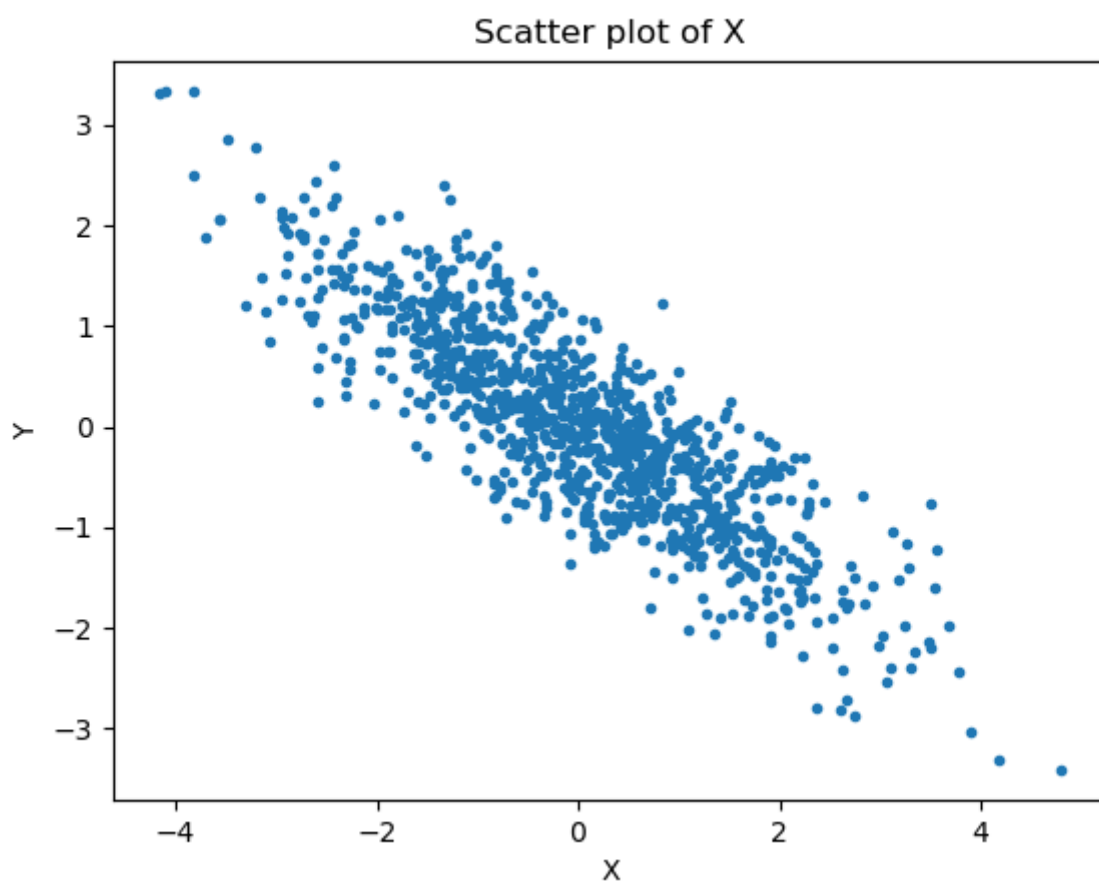
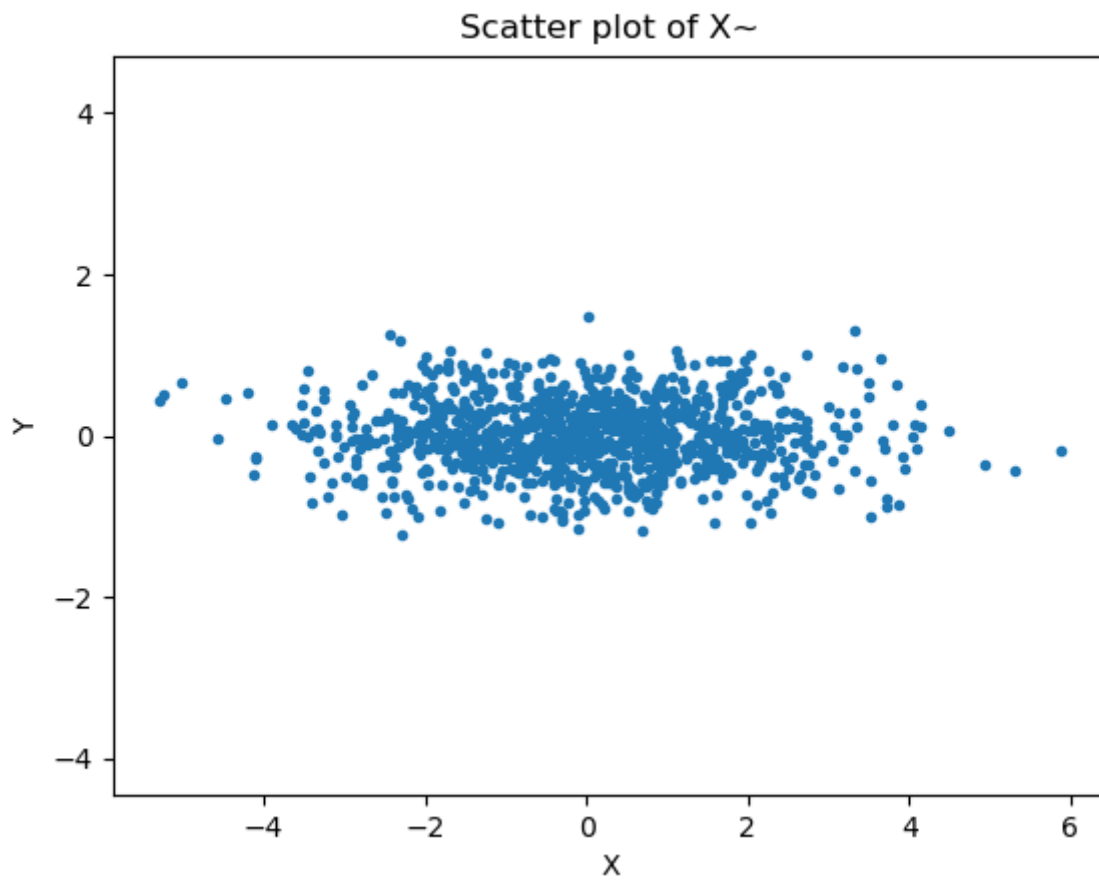


Eigen Image Analysis

Generating Gaussian random vectors





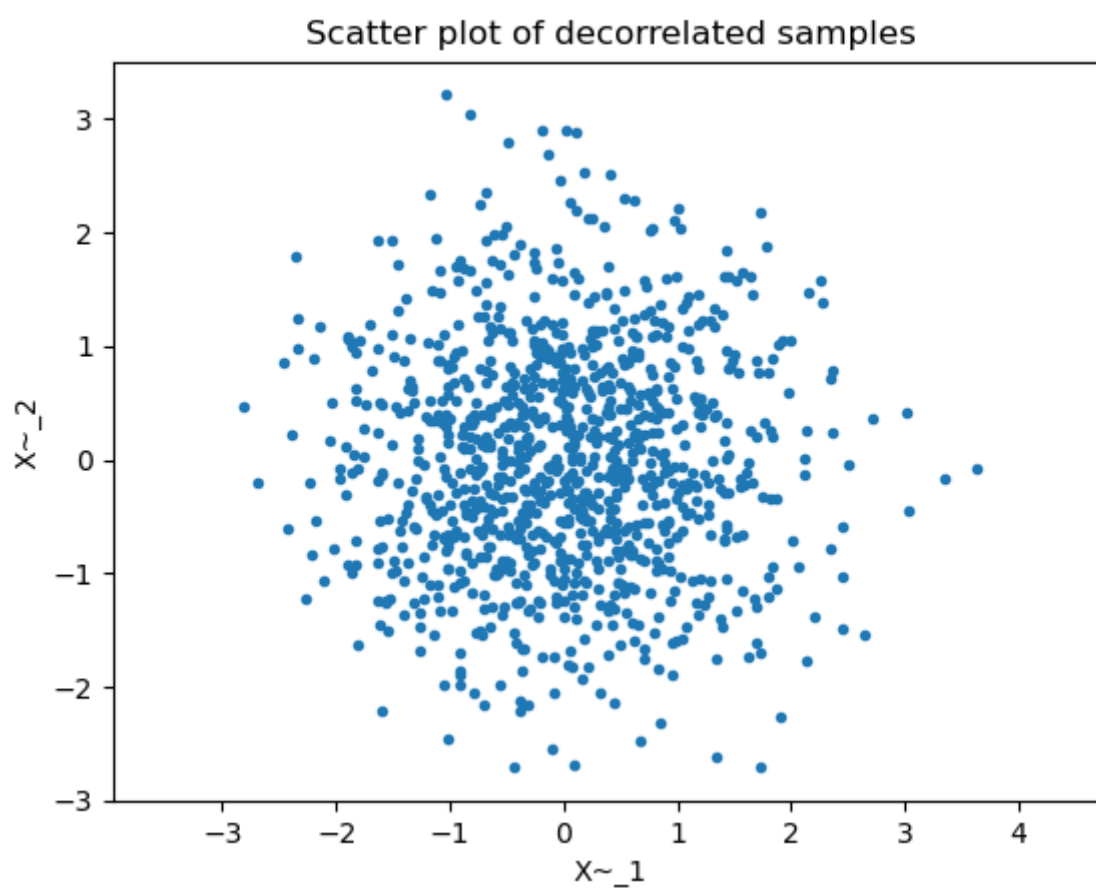
Theoretical Covariance

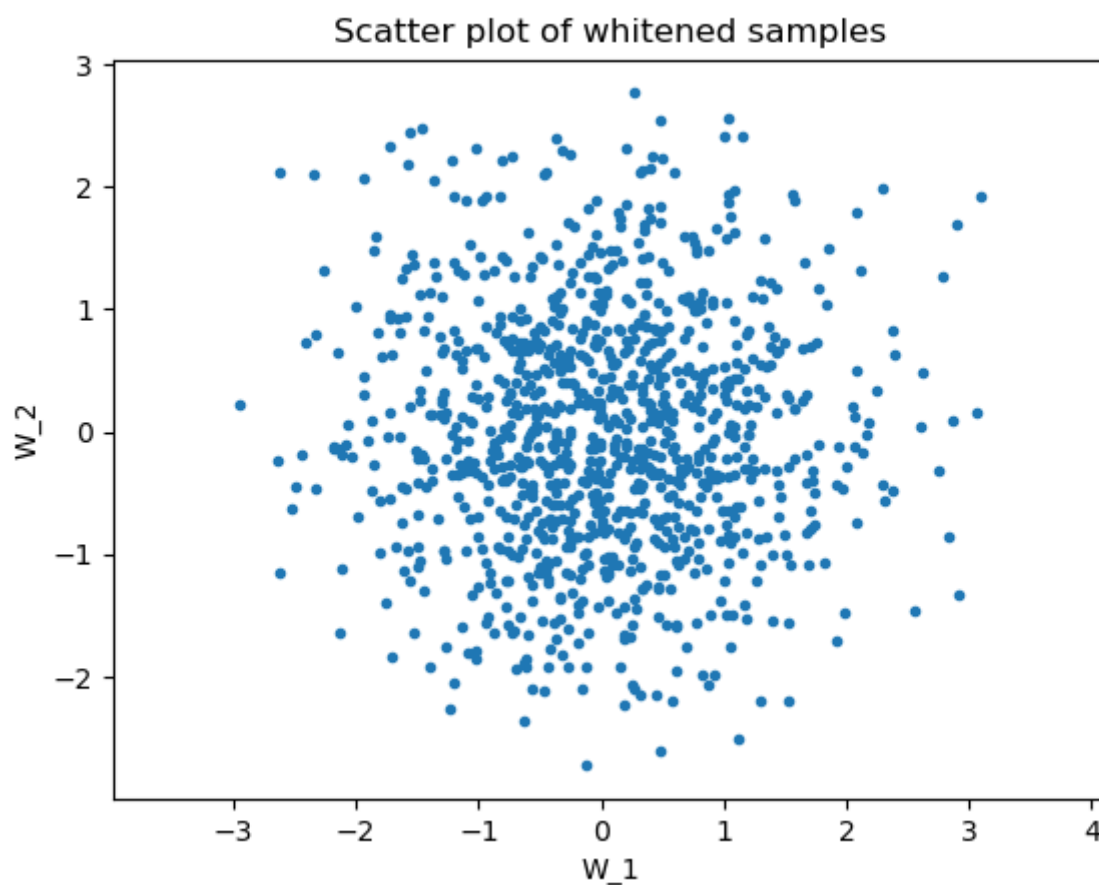
```
RX = [[2, -1.2],  
      [-1.2, 1]]
```

Estimated Covariance

```
R_hat = [[ 1.97972664 -1.16673842]  
         [-1.16673842  0.94524608]]
```

Scatter Plots for \tilde{X}_i and W_i

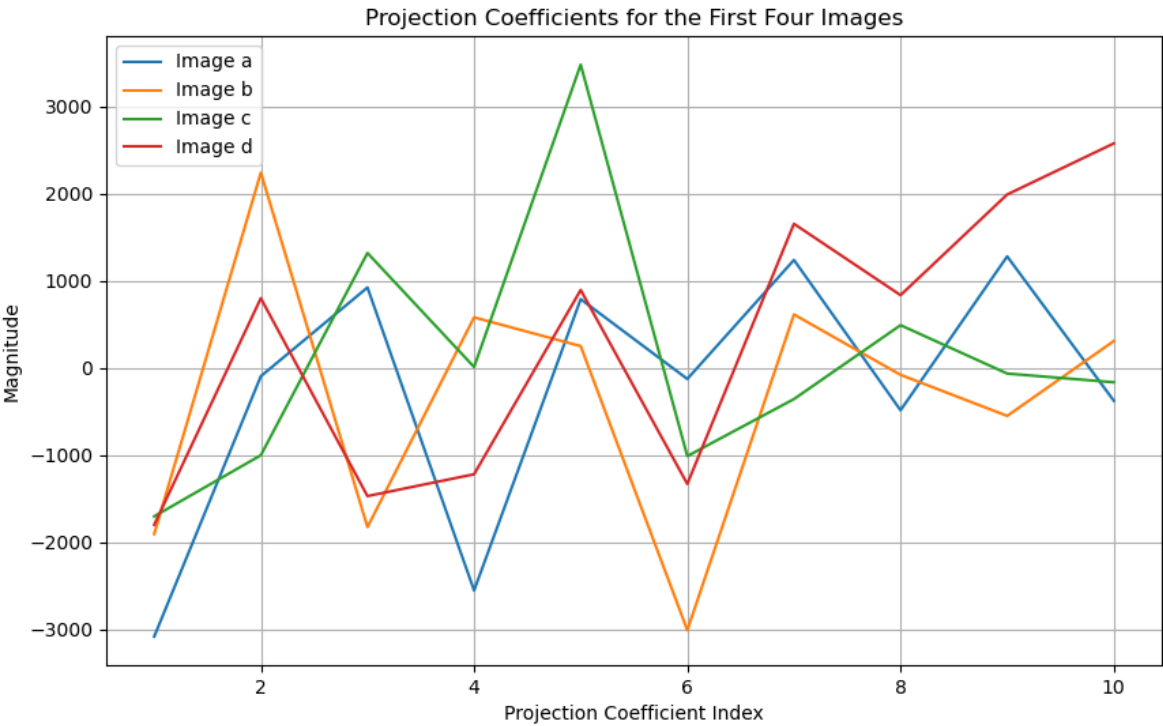
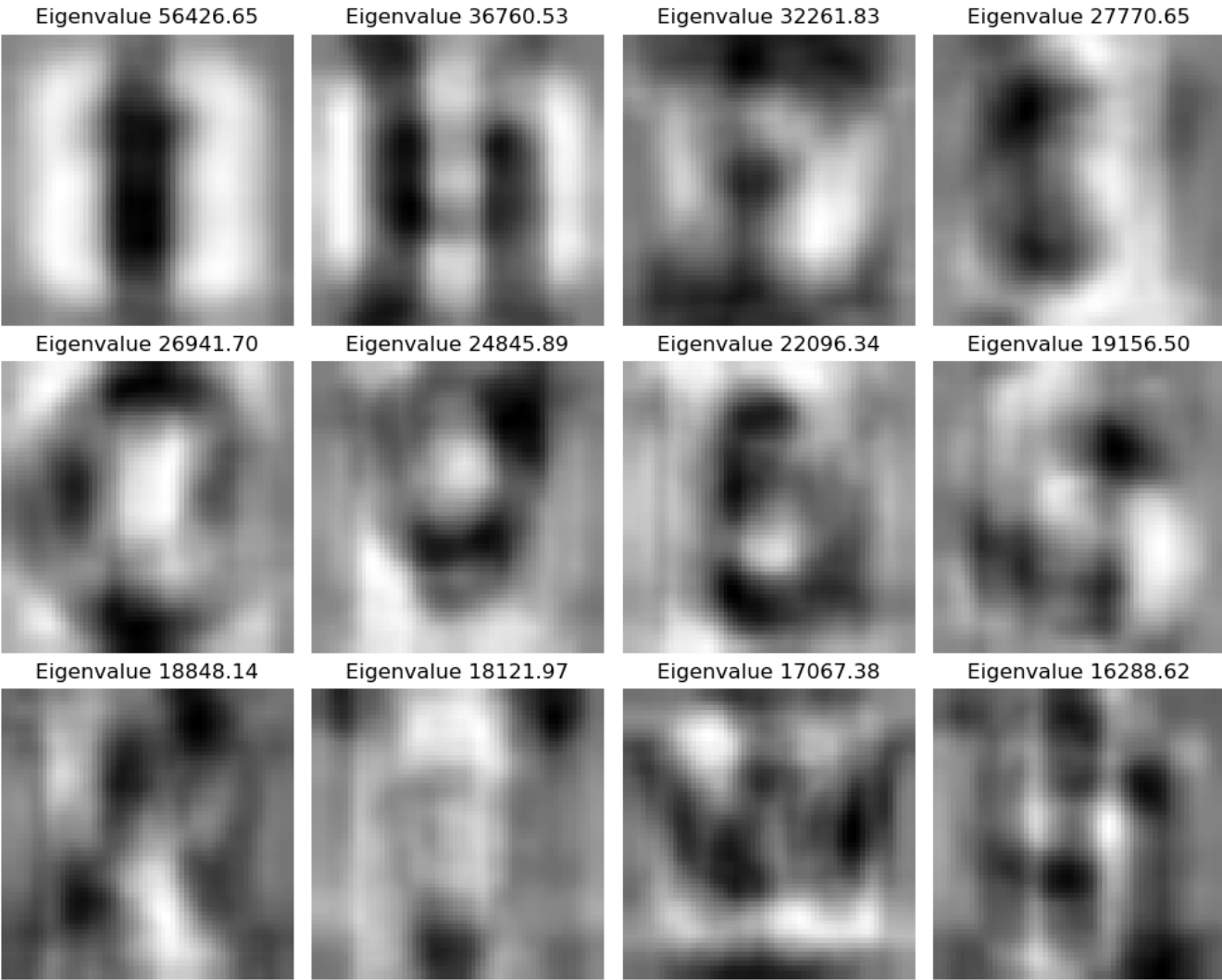




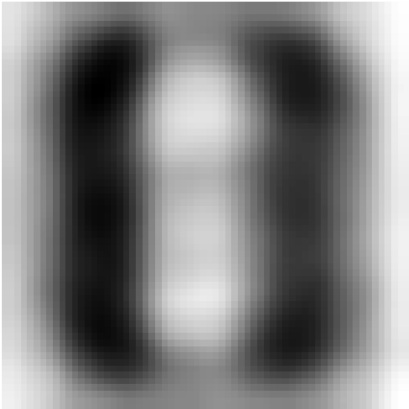
Covariance estimation R_{hat}_W

```
R_hat_whitened =  
[[ 1.00000000e+00 -1.70700958e-16]  
 [-1.70700958e-16 1.00000000e+00]]
```

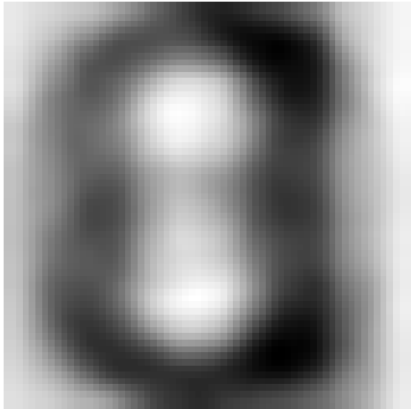
Eigenimages, PCA, and Data Reduction



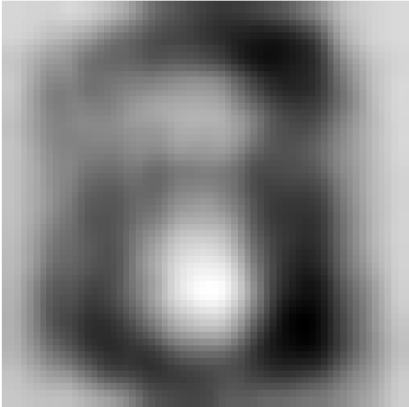
Synthesized Image (m = 1)



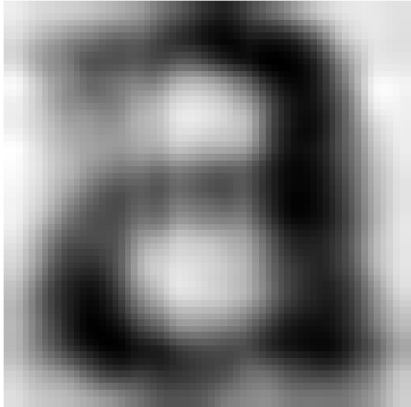
Synthesized Image (m = 5)



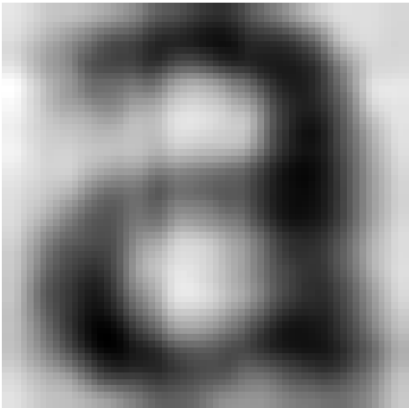
Synthesized Image (m = 10)



Synthesized Image (m = 15)



Synthesized Image (m = 20)



Synthesized Image (m = 30)

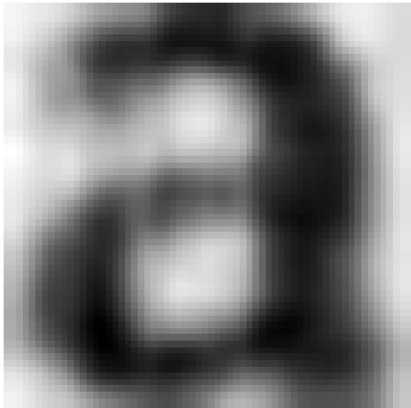




Image Classification

Misclassification using Rk

Input Character	Classifier Output
d	a
j	y
l	i
n	v
q	a
u	a
y	v

Misclassification of Bk variations

Bk = lambda_k

Input Character	Classifier Output
i	l

Input Character	Classifier Output
y	v

Bk = Rwc

Input Character	Classifier Output
g	q
y	v

Bk = Diagonal of Rwc

Input Character	Classifier Output
f	t
y	v

Bk = I

Input Character	Classifier Output
f	t
g	q
y	v

Tradeoff of data model accuracy vs estimate accuracy

The more complex the covariance model, the better it may fit the training data. But this may also lead to overfitting and poor generalization for unseen data. Simplifying the covariance model can reduce variance, but may introduce bias. The trade-off involves finding the right balance between model complexity and available data, aiming to achieve a model that captures the data distribution while remaining general to unseen data.