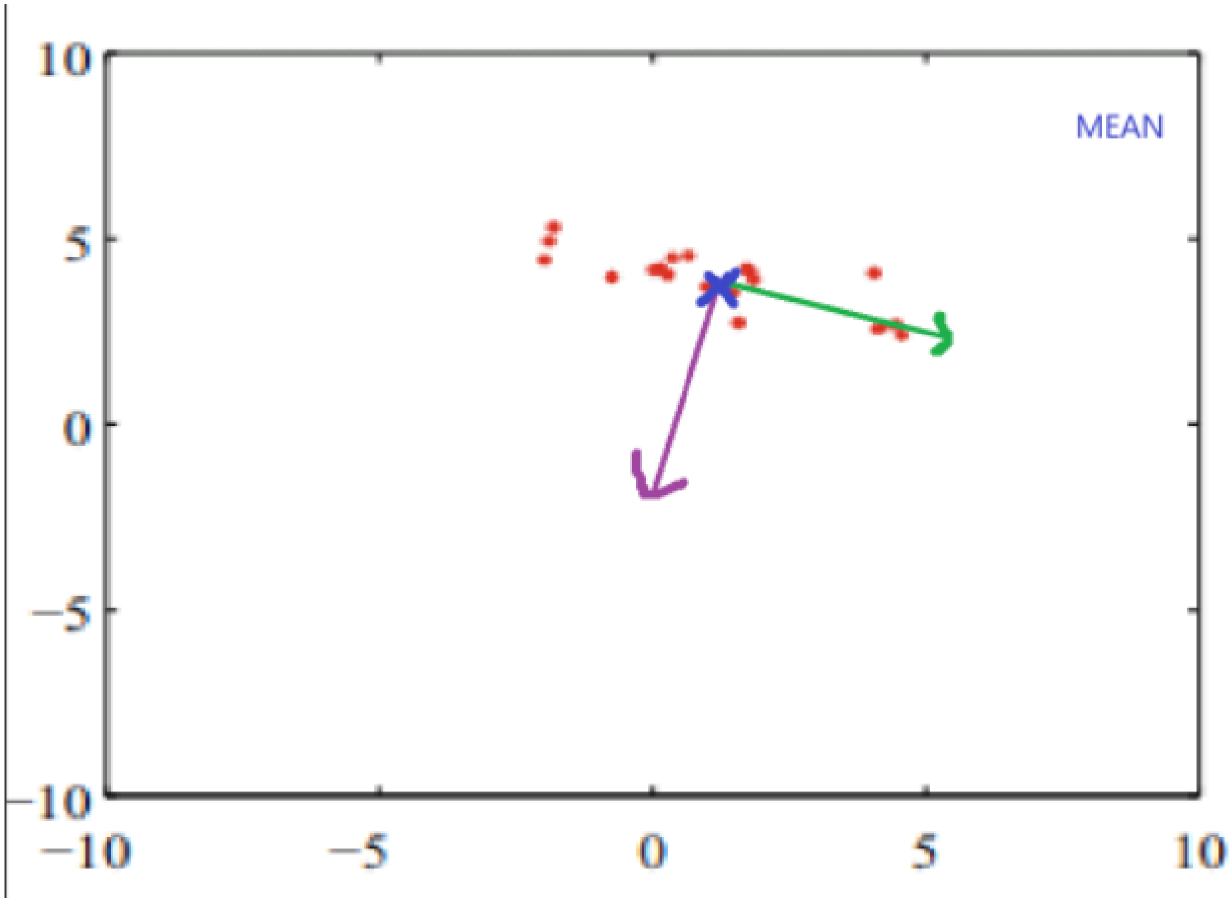
Homework 9

1. Textbook Problem 10.2



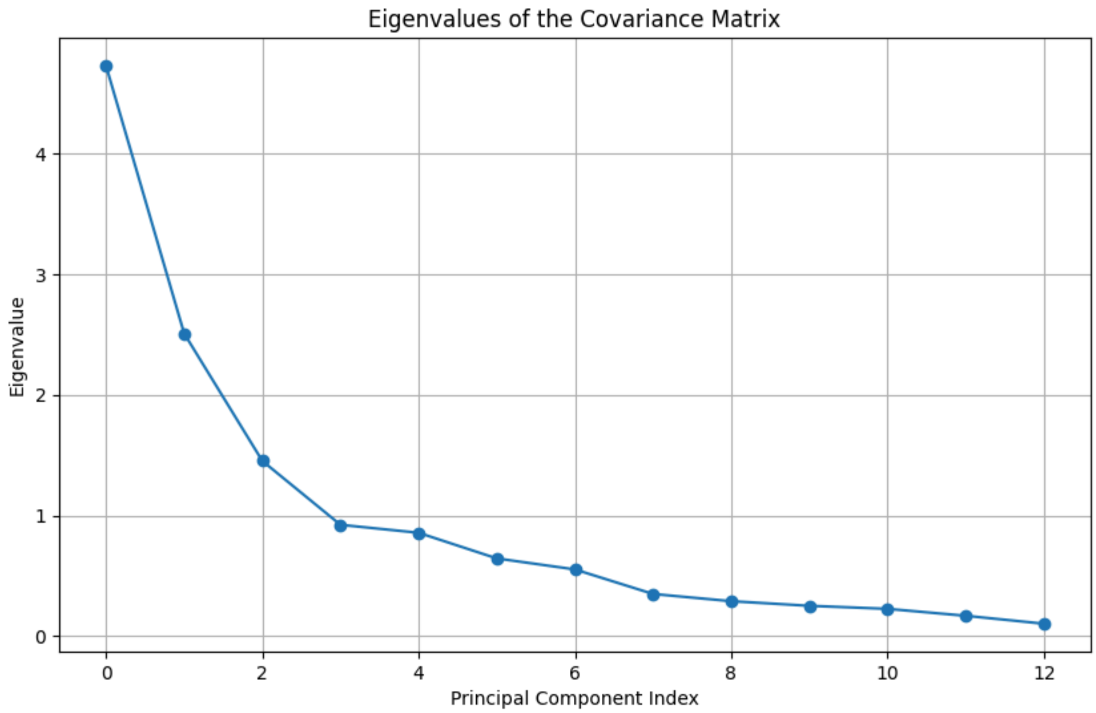
2. Textbook Problem 10.5

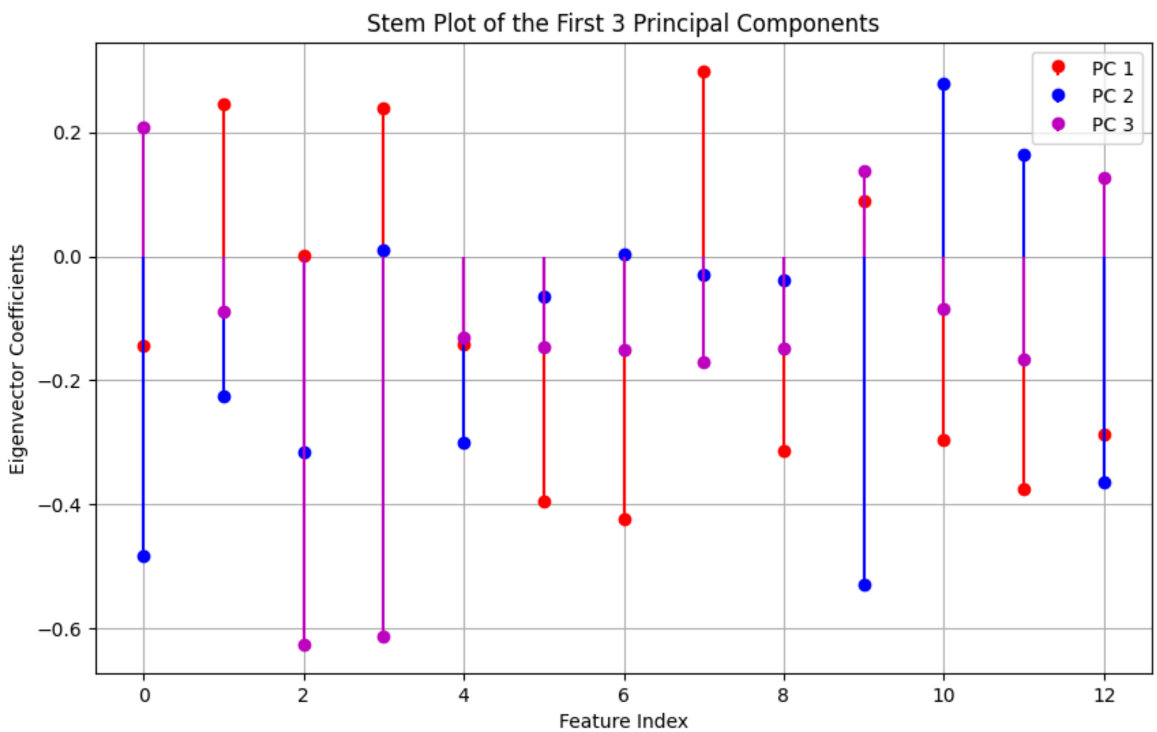
Here is my code and output for this question:

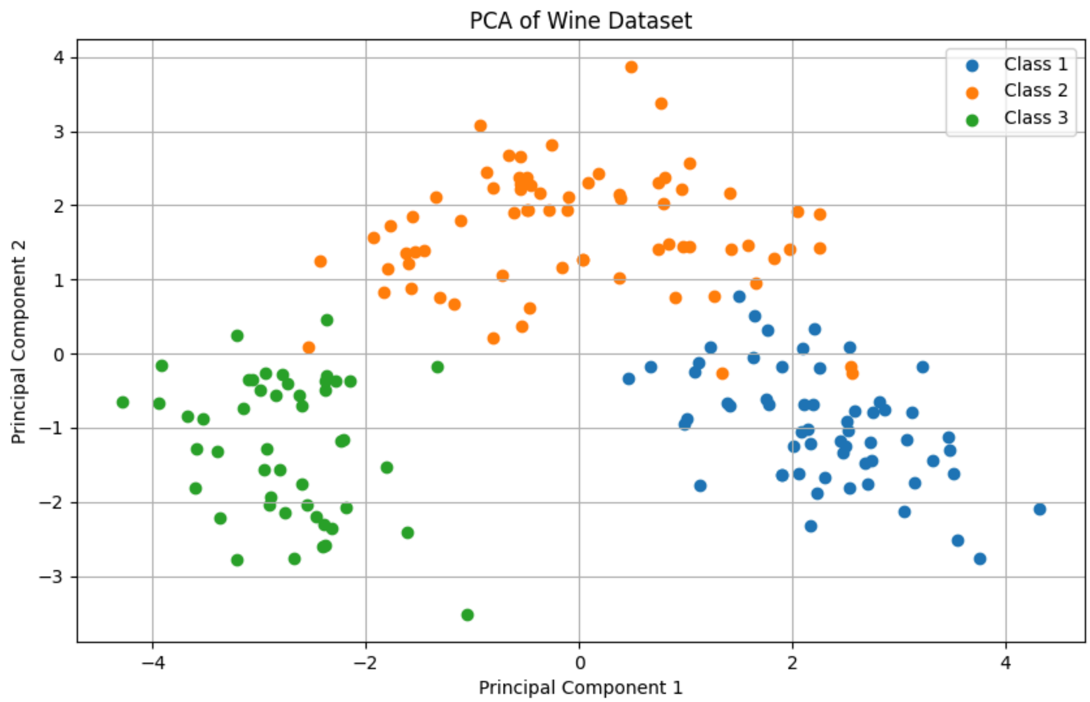
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler

wine_data_path = 'wine.data'

column_names = ['Class', 'Alcohol', 'Malic acid',
'Ash', 'Alcalinity of ash', 'Magnesium',
                'Total phenols', 'Flavanoids',
'Nonflavanoid phenols', 'Proanthocyanins',
                'Color intensity', 'Hue',
'OD280/OD315', 'Proline']

wine_df = pd.read_csv(wine_data_path, header=None,
names=column_names)

X = wine_df.drop('Class', axis=1)
y = wine_df['Class'].values

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

cov_matrix = np.cov(X_scaled, rowvar=False)
```

```python
eigenvalues, eigenvectors = np.linalg.eig(cov_matrix)

sorted_eigenvalues = np.sort(eigenvalues)[::-1]

plt.figure(figsize=(10, 6))
plt.plot(sorted_eigenvalues, marker='o')
plt.title('Eigenvalues of the Covariance Matrix')
plt.xlabel('Principal Component Index')
plt.ylabel('Eigenvalue')
plt.grid()
plt.show()

first_three_eigenvectors = eigenvectors[:,
np.argsort(eigenvalues)[::-1][:3]]

plt.figure(figsize=(10, 6))
colors = ['r', 'b', 'm']

for i in range(3):
    plt.stem(first_three_eigenvectors[:, i],
             label=f'PC {i + 1}',
             basefmt=" ",
             linefmt=colors[i],
             markerfmt=f'o{colors[i][0]}')

plt.title('Stem Plot of the First 3 Principal
Components')
plt.xlabel('Feature Index')
plt.ylabel('Eigenvector Coefficients')
plt.legend()
```

```python
plt.grid()
plt.show()

pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)

plt.figure(figsize=(10, 6))
for class_label in np.unique(y):
    plt.scatter(X_pca[y == class_label, 0], X_pca[y ==
class_label, 1], label=f'Class {class_label}')

plt.title('PCA of Wine Dataset')
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')
plt.legend()
plt.grid()
plt.show()

num_components = np.sum(sorted_eigenvalues > 1)
num_components
```


Eigenvalues of the Covariance Matrix
Eigenvalue
Principal Component Index


Stem Plot of the First 3 Principal Components
PC 1
PC 2
PC 3
Eigenvector Coefficients
Feature Index



a. A

   We should use three principal components for this dataset. Because if we use three components then all of the eigenvalues will be larger than 1. Therefore preserving the same amount of variance as the original dataset.

b. B

c. c

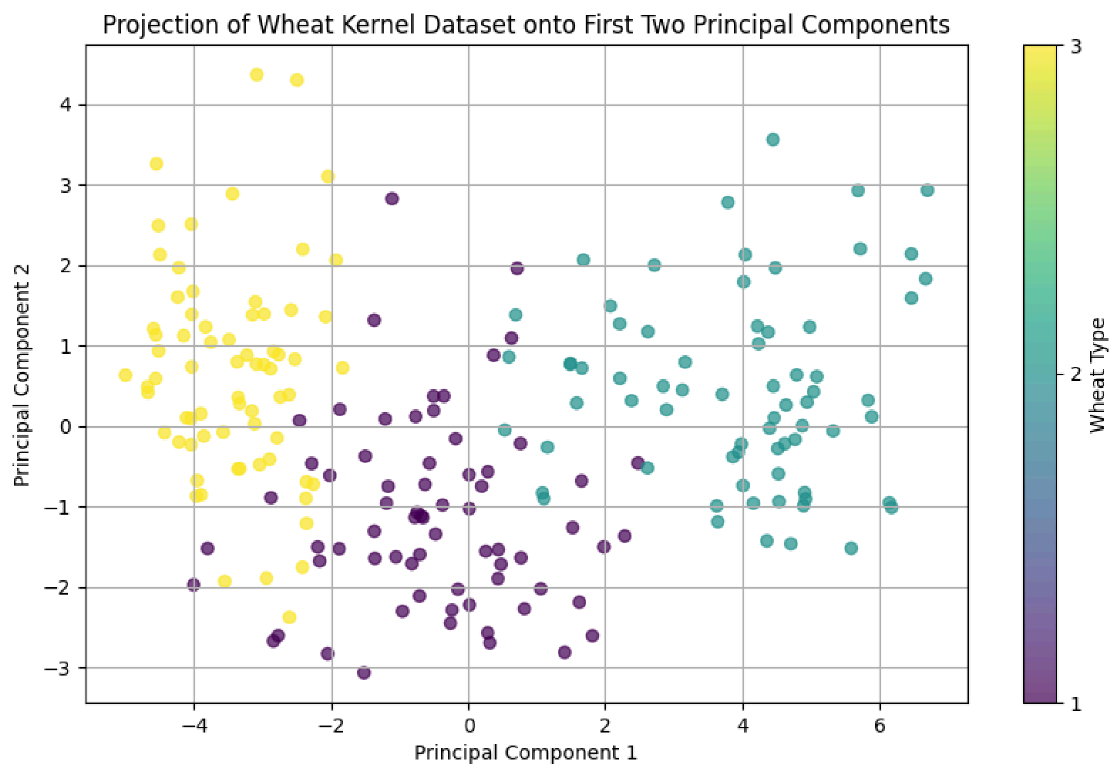
   Showed in the output a the top of this question.

3. Textbook Problem 10.6

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA

file_path = 'seeds_cleaned.csv'
data = pd.read_csv(file_path)

data = data.apply(pd.to_numeric, errors='coerce')

features = data.drop(columns=['label'])
labels = data['label']

pca = PCA(n_components=2)
principal_components = pca.fit_transform(features)

pc_df = pd.DataFrame(data=principal_components, columns=['PC1',
'PC2'])
pc_df['label'] = labels

plt.figure(figsize=(10, 6))
scatter = plt.scatter(pc_df['PC1'], pc_df['PC2'], c=pc_df['label'],
cmap='viridis', alpha=0.7)
plt.title('Projection of Wheat Kernel Dataset onto First Two
Principal Components')
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')

cbar = plt.colorbar(scatter, ticks=[1, 2, 3])
cbar.set_label('Wheat Type')
cbar.ax.set_yticklabels(['1', '2', '3'])

plt.grid()
plt.show()

eigenvalues = pca.explained_variance_

plt.figure(figsize=(8, 5))
plt.bar(range(1, len(eigenvalues) + 1), eigenvalues, alpha=0.7,
color='blue')
```

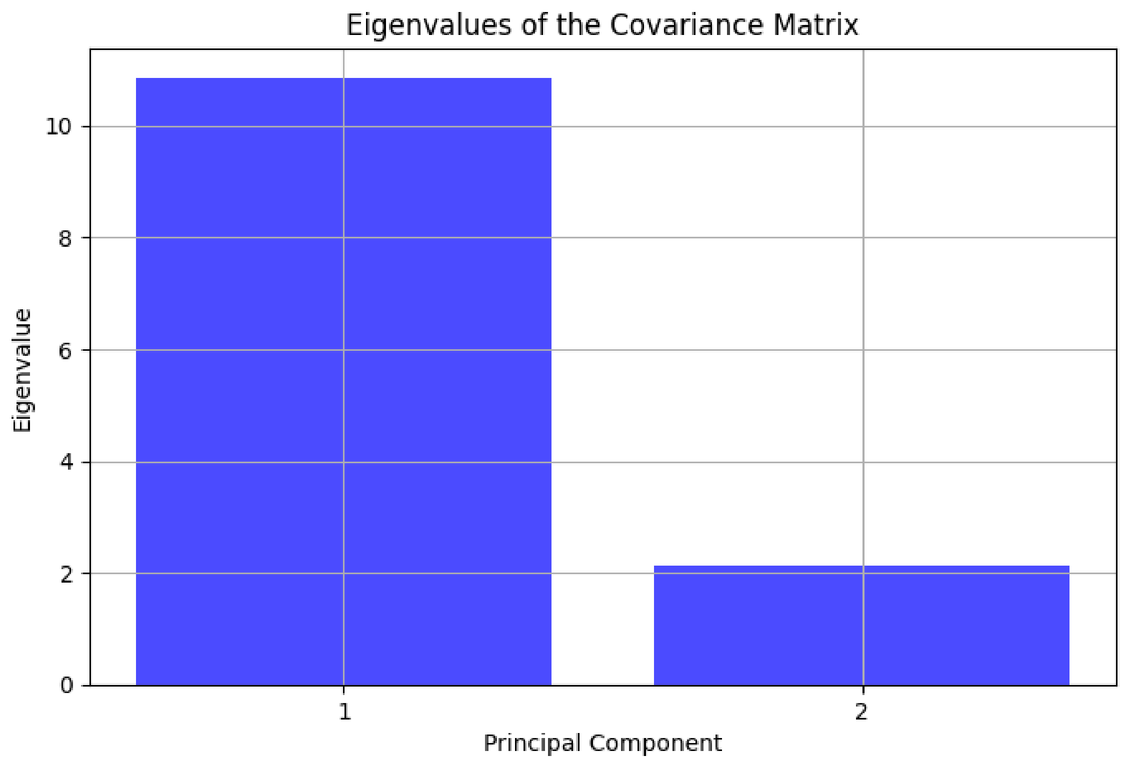
```
plt.title('Eigenvalues of the Covariance Matrix')
plt.xlabel('Principal Component')
plt.ylabel('Eigenvalue')
plt.xticks(range(1, len(eigenvalues) + 1))
plt.grid()
plt.show()

pc_df.to_csv('seeds_principal_components.csv', index=False)
```

a. A



Yes, the different typers of wheat have almost categorized themselves according to the first principal component.

b. b



We should use at least three principal components because the eigenvalues are too high for just two principal components. We want the eigenvalues to be over 1 but this much over is too much.

4. 4
   a. Covmat{(x)}
      The covariance matrix is:

$$\begin{bmatrix} var(x^{(1)}) & cov(x^{(2)}, x^{(1)}) \\ cov(x^{(2)}, x^{(1)}) & var(x^{(2)}) \end{bmatrix}$$

      Now lets substitute for the known values:

$$\begin{bmatrix} 9 & 6 \\ 6 & 4 \end{bmatrix} = covmat\{(x)\}$$

   b. B
      To solve for the eigenvalues we can use this equation:

$$det(covmat\{(x)\}) - \lambda I = 0$$

      Which then turns into:

$$covmat\{(x)\} - \lambda I = \begin{bmatrix} 9-\lambda & 6 \\ 6 & 4-\lambda \end{bmatrix}$$

      Now we set the determinant to 0:

$$(9 - \lambda)(4 - \lambda) - 6 \times 6 = 0$$

$$\lambda^2 - 13\lambda + 36 - 36 = 0$$

$$\lambda^2 - 13\lambda = 0$$

      So we get:

$$\lambda = 0 \qquad and \qquad \lambda = 13$$

      Therefore the two eigenvalues are 0 and 13.
   c. c
      Since we have a positive eigenvalue and an eigenvalue that is 0, we know that the shape of the blop is linear and wide. A positive eigen value means there is a lot of spread in one direction, and a 0 eigenvalue means there is no variance in the other direction.

5. 5

first , lets center the data pointst by subtracting the mean:

$$\begin{bmatrix} 3 \\ 3 \\ 5 \\ 7 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 4 \\ 6 \end{bmatrix}$$

Now lets calculate the projection using this formula:

$$proj_{u_1} = \frac{(x_1 - mean(\{x\}))^T u_1}{u_1^T u_1} u_1$$

For the first projectino, lets calculate the numerator first:

$$\begin{bmatrix} 2 \\ 2 \\ 4 \\ 6 \end{bmatrix}^T \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{bmatrix} =$$

$$2 * 0.5 + 2 * 0.5 + 4 * 0.5 + 6 * 0.5 = 7$$

Now the numerator:

$$\begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{bmatrix}^T \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{bmatrix} =$$

$$0.5^2 + 0.5^2 + 0.5^2 + 0.5^2 = 1$$

So the fraction is $\frac{7}{1}$ or just 7. Now we need to multiply it with u_1

So,

$$7 * \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 3.5 \\ 3.5 \\ 3.5 \\ 3.5 \end{bmatrix} = proj_{u_1}$$

Now lets do the second projection

$$\begin{bmatrix} 2 \\ 2 \\ 4 \\ 6 \end{bmatrix}^T \begin{bmatrix} 0.5 \\ -0.5 \\ -0.5 \\ -0.5 \end{bmatrix} =$$

$2 * 0.5 + 2 * (-0.5) + 4 * (-0.5) + 6 * (-0.5) = -5$

Now the numerator:

$$\begin{bmatrix} 0.5 \\ -0.5 \\ -0.5 \\ -0.5 \end{bmatrix}^T \begin{bmatrix} 0.5 \\ -0.5 \\ -0.5 \\ -0.5 \end{bmatrix} =$$

$0.5^2 + (-0.5)^2 + (-0.5)^2 + (-0.5)^2 = 1$

So the fraction is $\frac{-5}{1}$ or just $-\ 5$. Now we need to multiply it with u_2

So,

$$-\ 5\ * \begin{bmatrix} 0.5 \\ -0.5 \\ -0.5 \\ -0.5 \end{bmatrix} = \begin{bmatrix} -2.5 \\ 2.5 \\ 2.5 \\ 2.5 \end{bmatrix} = proj_{u_2}$$

Since the first projection fraction gave us 7 and the second gave us 5 the projected coordinates are (7, -5)