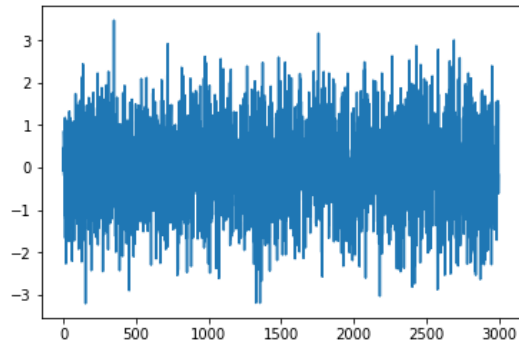


Práctica 2

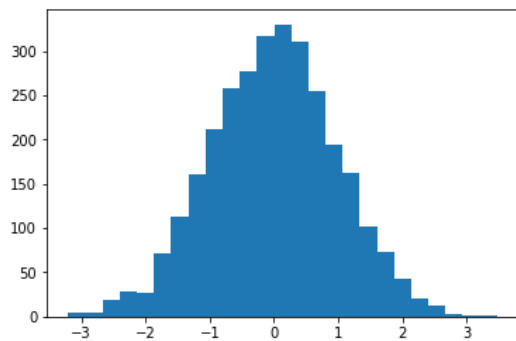
Vector de 3000 muestras con distribución normal.

```
In [2]: y_randn = np.random.randn(3000)
...: plt.plot(y_randn)
...: plt.show()
```



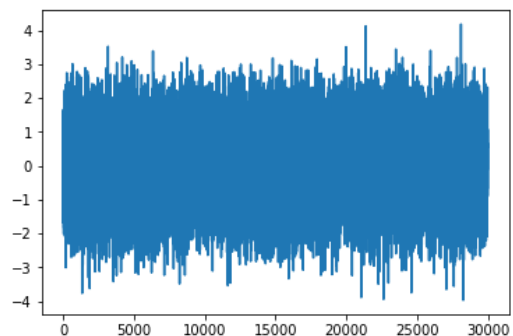
Histograma de dicho vector.

```
In [3]: plt.hist(y_randn, 25)
...: plt.show()
```



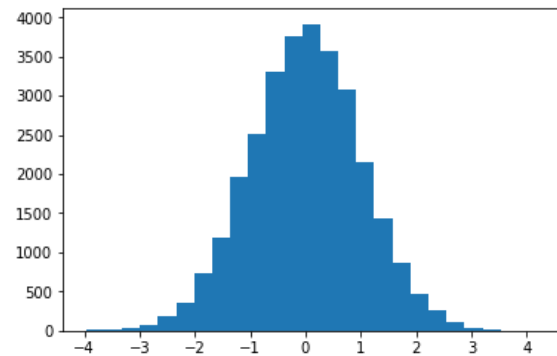
Vector con 30,000 muestras con distribución normal.

```
In [4]: y = np.random.randn(30000)
...: plt.plot(y)
...: plt.show()
```



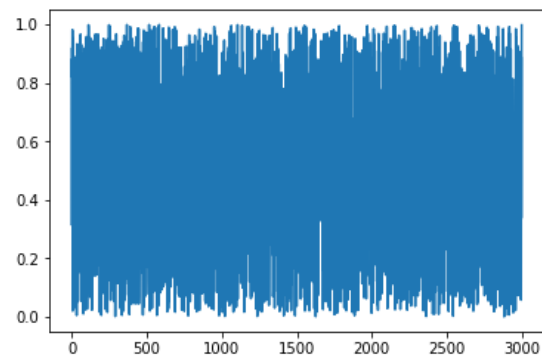
Histograma del vector de 30,000 muestras.

```
In [5]: plt.hist(y, 25)
...: plt.show()
```



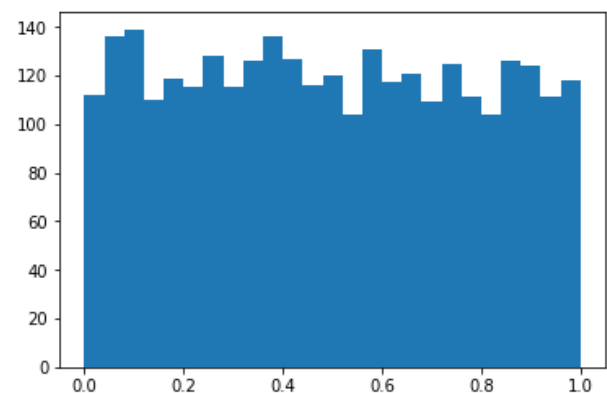
Vector con 3,000 muestras con distribución uniforme.

```
In [6]: y_randu = np.random.rand(3000)
...: plt.plot(y_randu)
...: plt.show()
```



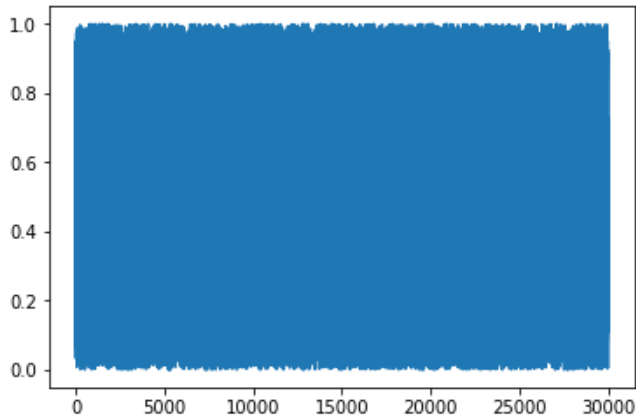
Histograma del vector anterior.

```
In [7]: plt.hist(y_randu, 25)
...: plt.show()
```



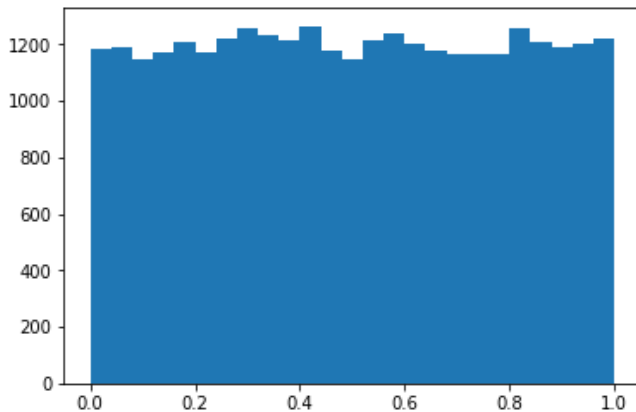
Ahora un vector con 30,000 muestras con la misma distribución.

```
In [8]: y = np.random.rand(30000)
...: plt.plot(y)
...: plt.show()
```



Y su histograma.

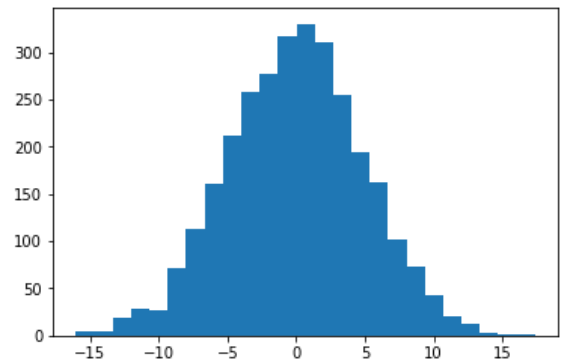
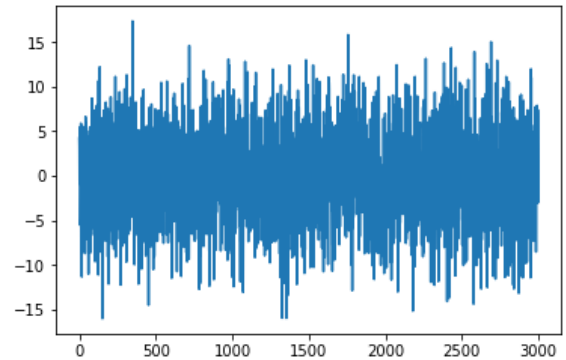
```
In [9]: plt.hist(y, 25)
...: plt.show()
```



Al multiplicar un vector por un escalar, modificamos su desviación estándar y covarianza.

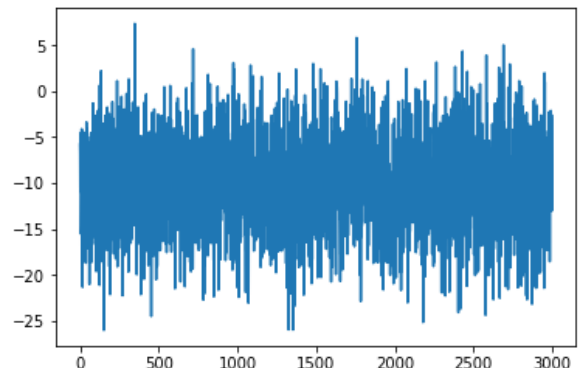
Introducción a las DNNs

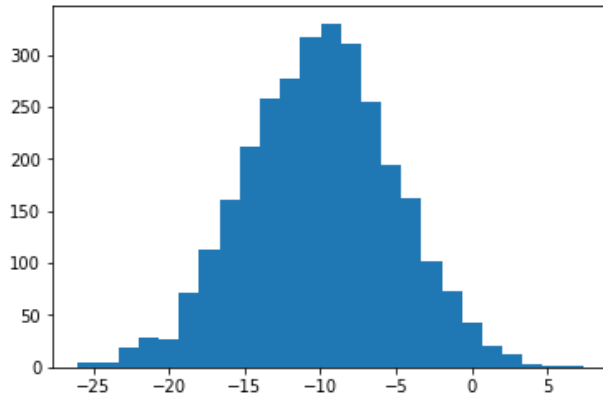
```
In [10]: y = 5*y_randn
...: plt.plot(y)
...: plt.show()
...: z = plt.hist(y, 25)
```



Cuando le restamos o sumamos un escalar a cada uno de los valores del vector, le modificamos su media.

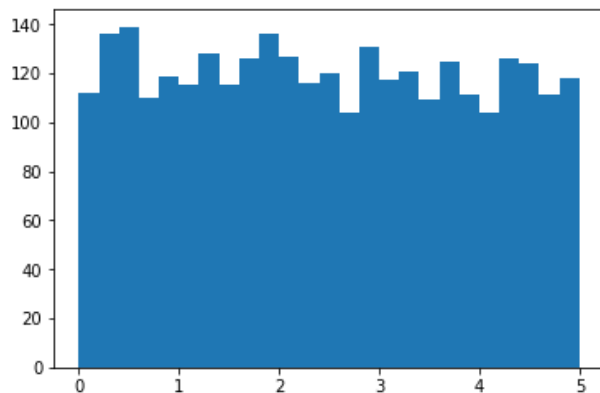
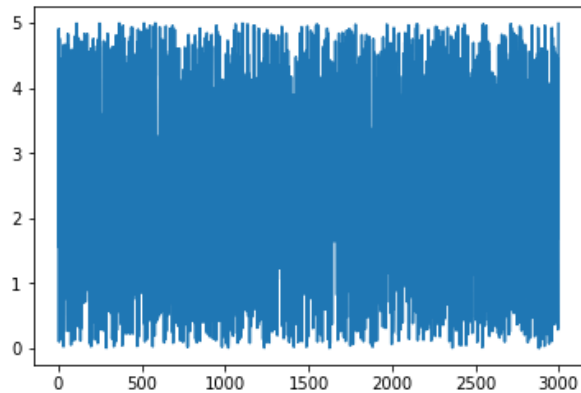
```
In [11]: y = (5*y_randn-10)
...: plt.plot(y)
...: plt.show()
...: z=plt.hist(y, 25)
```





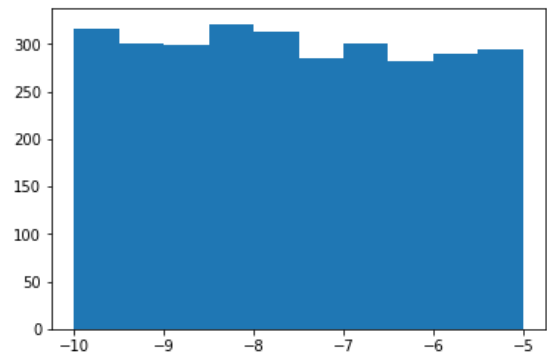
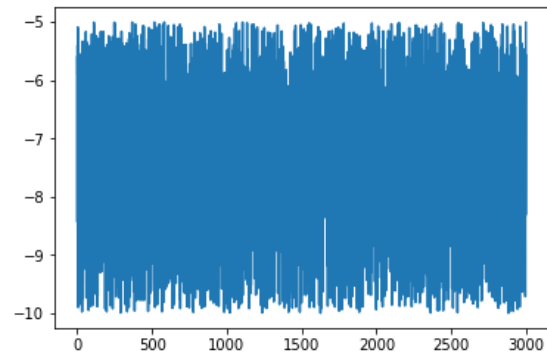
Ahora le aplicamos las mismas operaciones a un vector de 3,000 muestras con distribución uniforme.

```
In [12]: y = 5*y_randu
...: plt.plot(y)
...: plt.show()
...: z = plt.hist(y, 25)
```



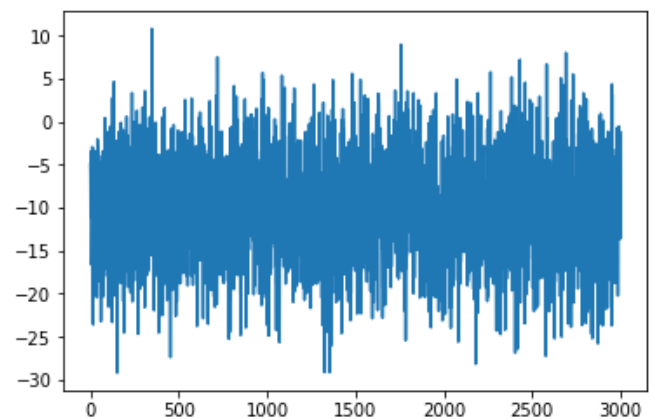
Luego le modificamos su media

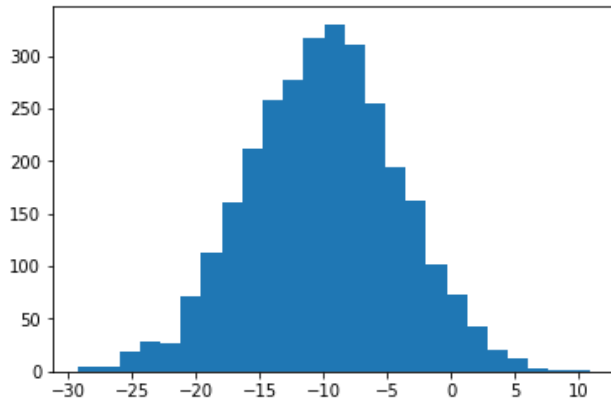
```
In [13]: y = (5*y_randu-10)
...: plt.plot(y)
...: plt.show()
...: z=plt.hist(y)
```



Al sumar dos vectores con distribuciones normales de diferente media y desviación estándar, el resultado no tiene distribución normal.

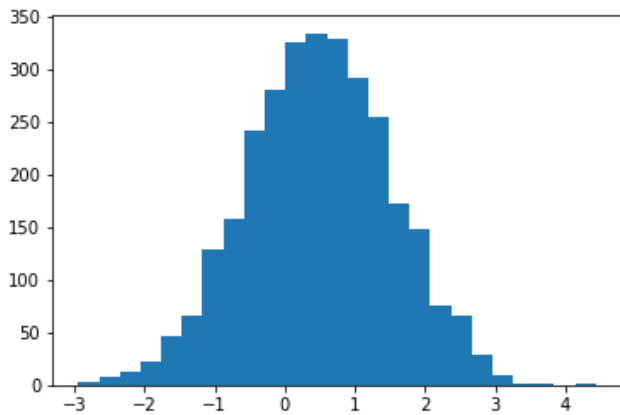
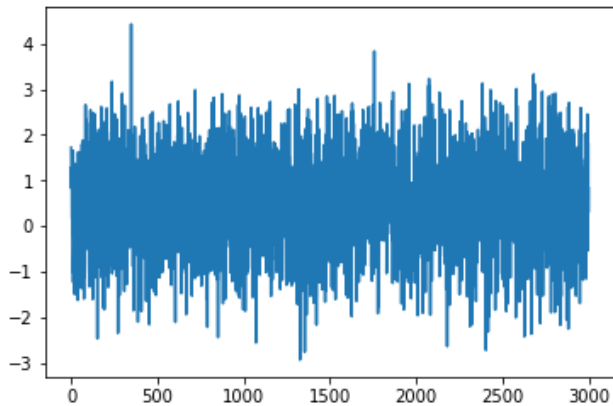
```
In [16]: y = y_randn+(5*y_randn-10)
...: plt.plot(y)
...: plt.show()
...: plt.hist(y,25)
```





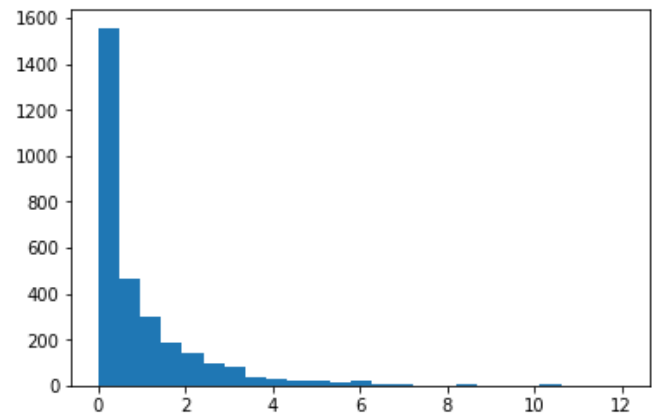
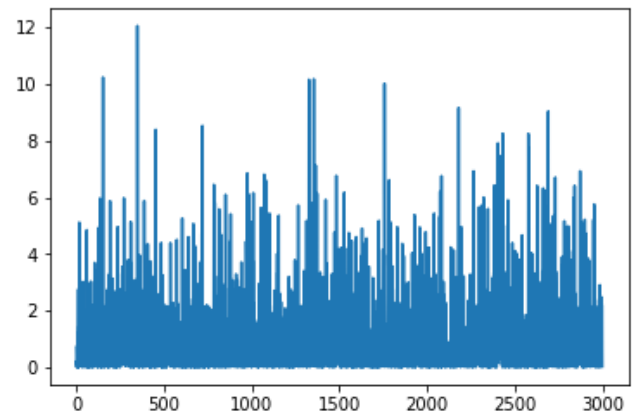
Ahora observamos los resultados de sumar muestras aleatorias con distribución normal, con muestras con distribución uniforme.

```
In [17]: y = y_randn + y_randu
...: plt.plot(y)
...: plt.show()
...: plt.hist(y, 25)
...: plt.show()
```



Luego, multiplicamos dos vectores con distribuciones normales. El resultado no tiene distribución normal.

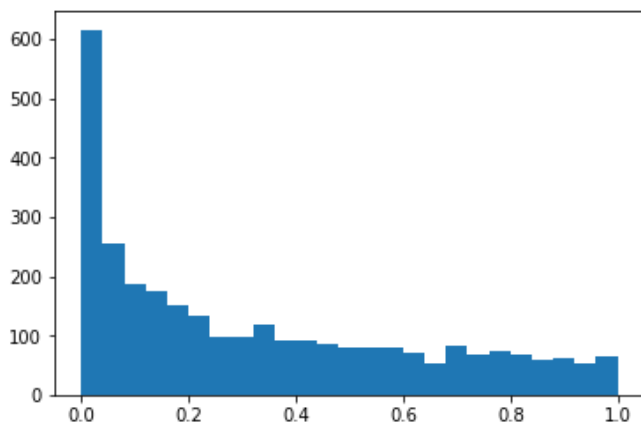
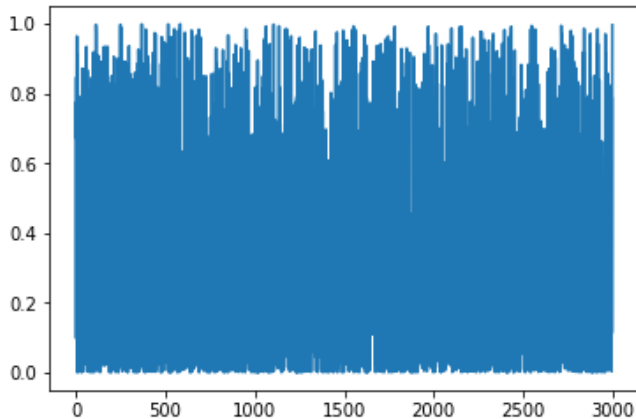
```
In [18]: y = y_randn * y_randn
...: plt.plot(y)
...: plt.show()
...: plt.hist(y, 25)
...: plt.show()
```



González Flores Andrés

Ahora hacemos lo mismo para dos vectores cuyos valores están distribuidos uniformemente.

```
In [19]: y = y_randu * y_randu
...: plt.plot(y)
...: plt.show()
...: plt.hist(y, 25)
...: plt.show()
```



El resultado no es una distribución uniforme.

Ahora se calculan la media, desviación estándar y varianza de la distribución normal.

```
In [20]: media = np.mean(y_randn)
...: print('Media = ', media)
...: desv_est = np.std(y_randn)
...: print('Std = ', desv_est)
...: var = desv_est**2
...: print('Varianza = ', var)
Media = -0.004773356343996198
Std = 0.9874373716424611
Varianza = 0.9750325629161719
```

Introducción a las DNNs

Y, por último, hacemos lo mismo para la distribución uniforme.

```
In [21]: media = np.mean(y_randu)
...: print('Media = ', media)
...: desv_est = np.std(y_randu)
...: print('Std = ', desv_est)
...: var = desv_est**2
...: print('Varianza = ', var)
Media = 0.49329141982179664
Std = 0.2882555138980179
Varianza = 0.08309124129261039
```