Second exercise assignment

Lucas Nogueira Ribeiro

The Python source code written to solve this assignment is available at `https://github.com/lnribeiro/patternrecognition`.

Exercise #01

Consider a Bernoulli random variable $x \in \{0, 1\}$ parametrized in μ. Its probability density function is given by $p(x|\mu) = \mu^x(1-\mu)^{1-x}$.

- The mean value of x is given by

$$\mathbb{E}[x] = \sum_{x \in \{0,1\}} xp(x|\mu) = 0.(1-\mu) + 1.\mu = \mu$$

- The variance of x is given by

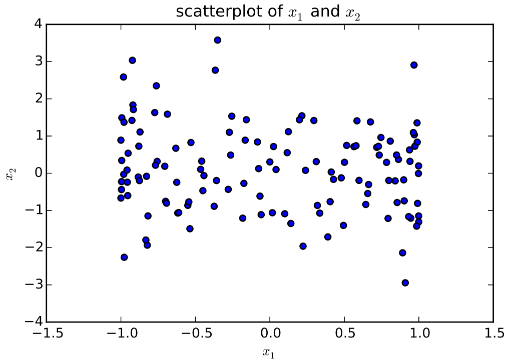
$$var[x] = \mathbb{E}[x^2] - \mathbb{E}[x]^2 = \sum_{x \in \{0,1\}} xp(x|\mu) = 0^2.(1-\mu) + 1^2.\mu - \mu^2 = \mu(1-\mu)$$

Exercise #02

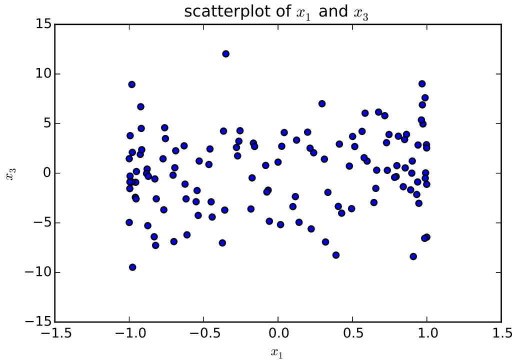
The covariance matrix of $\mathbf{X}$ is:

```
[[  0.49866646  -0.10245845   0.4109965 ]
 [ -0.10245845   1.33957189   3.89705712]
 [  0.4109965    3.89705712  16.13133976]]
```

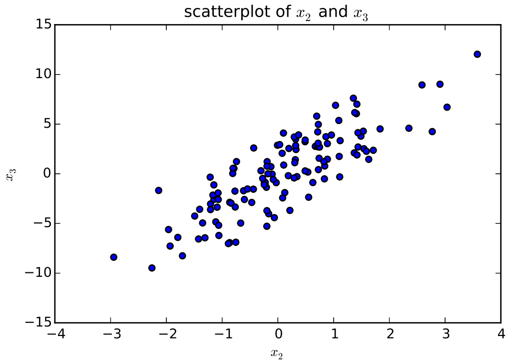
By inspection, $var[x_1] = 0.498$, $var[x_2] = 1.339$, and $var[x_3] = 16.13$. The variables pair (x_1, x_2) is negatively correlated since $cov[x_1, x_2] = -0.10$. On the other hand, the pairs (x_1, x_3) and (x_2, x_3) are positively correlated since $cov[x_1, x_3] = 0.41$, and $cov[x_2, x_3] = 3.89$. The scatterplot diagrams of the random variables are depicted in Fig. 1. The mean values of the random variables are:



(a) Scatterplot of x_1 and x_2.



(b) Scatterplot of x_1 and x_3.

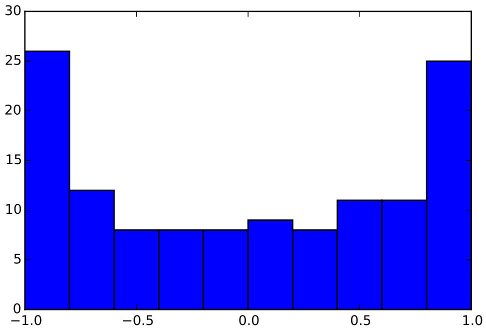


(c) Scatterplot of x_2 and x_3.

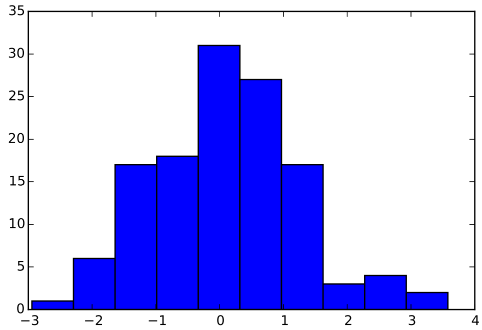
Figure 1: Scatterplots of the generated data

```
mean(x_1): -0.000089
mean(x_2): 0.111424
mean(x_3): 0.160293
```

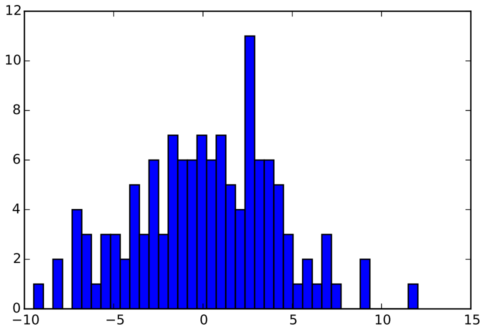
It is not possible to infer the data probability distribution function (pdf) just by calculating its mean value and plotting its values. The histograms of the three random variables were calculated (using the builtin `matplotlib.hist` Python module) to give us an idea of how they are distributed. Histograms 3b and 3c resembles Gaussian pdfs, whereas histogram 3a presents spikes at ± 1, similar to a arcsine pdf.



(a) Histogram of x_1.



(b) Histogram of x_2.

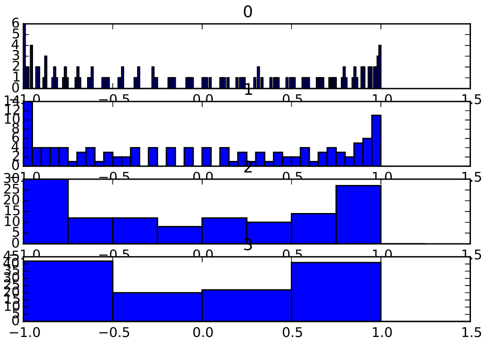


(c) Histogram of x_3.

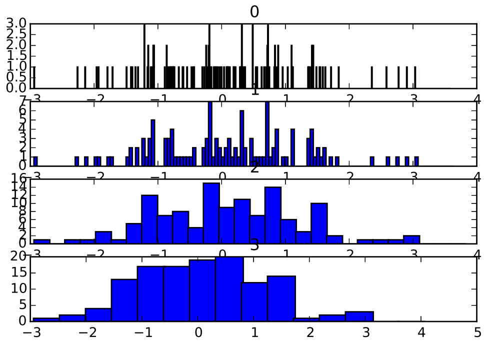
Figure 2: Histograms of the generated data

Exercise #03

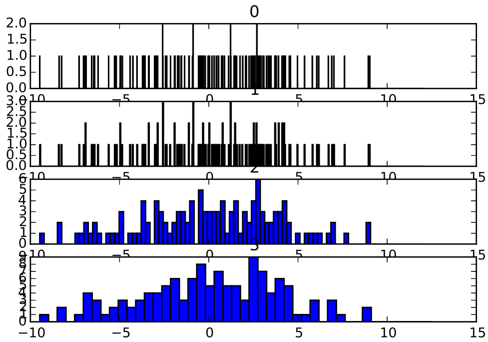
The following histograms were generated using our own Python implementation. An intermediate value for Δ, say 0.25, gives good results.



(a) Histogram of x_1.



(b) Histogram of x_2.

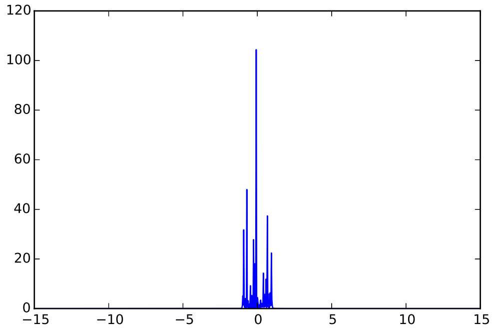


(c) Histogram of x_3.

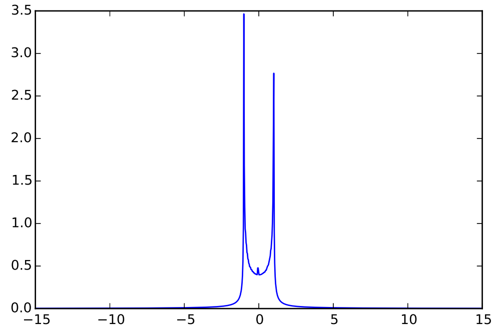
Figure 3: Histograms of the generated data using $\Delta = 0.01, 0.05, 0.25, 0.5$.

Exercise #04

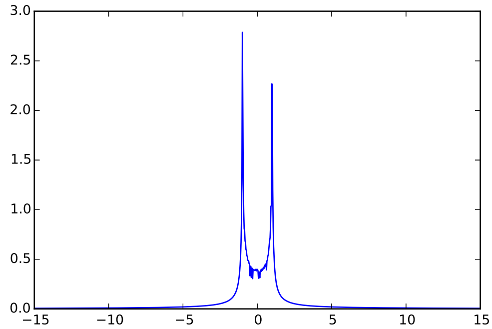
In this exercise, the K nearest neighbors (kNN) was used to estimate the density function of x_1, x_2, and x_3. It consists of a nonparametric estimation method based on a neighborhood averaging. The estimated densities are depicted below.



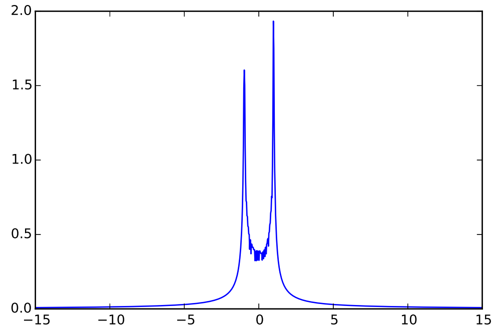
(a) $K = 1$



(b) $K = 5$

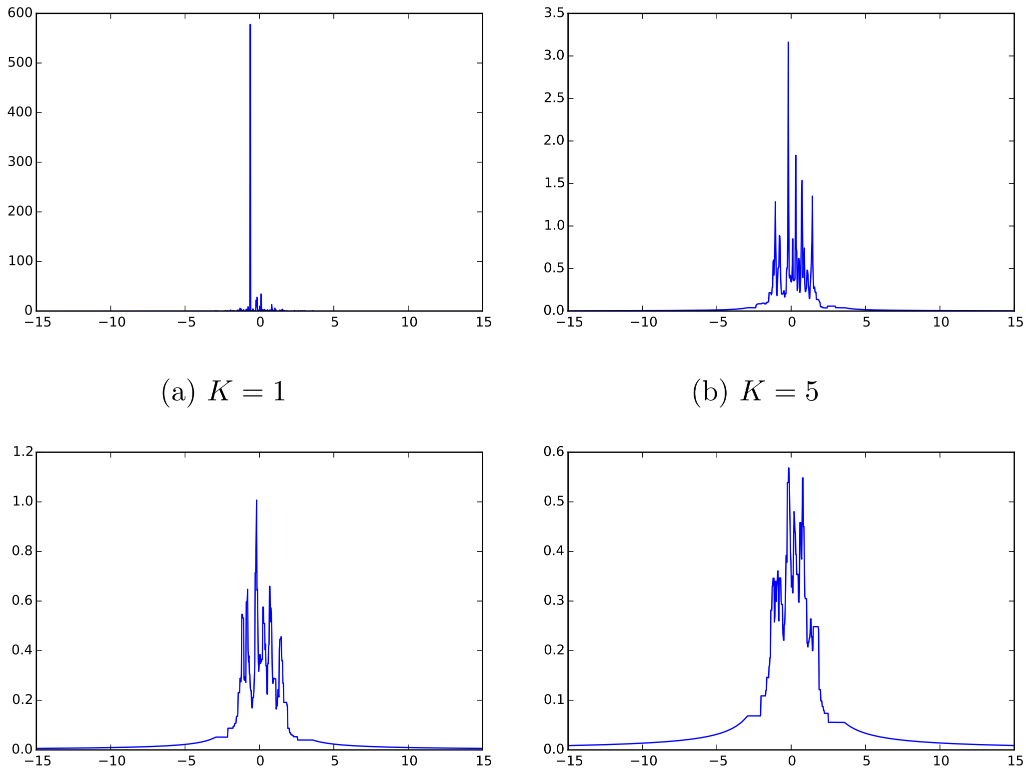


(c) $K = 10$



(d) $K = 15$

Figure 4: Estimated pdf of x_1 for different K.



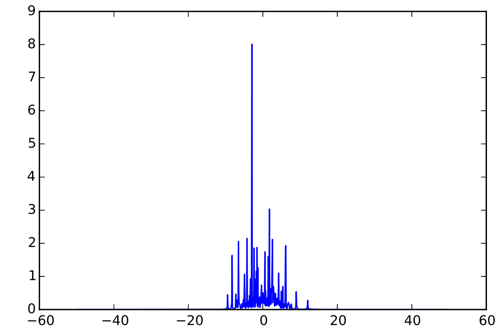
(a) $K = 1$

(b) $K = 5$

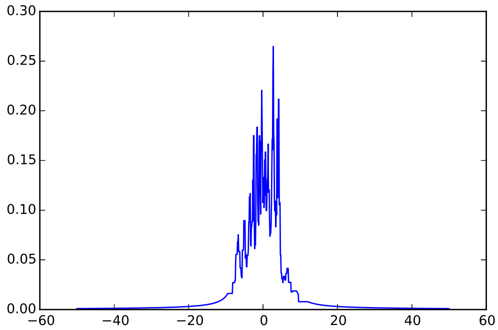
(c) $K = 10$

(d) $K = 15$

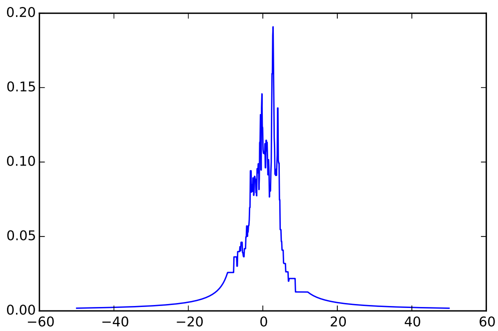
Figure 5: Estimated pdf of x_2 for different K.



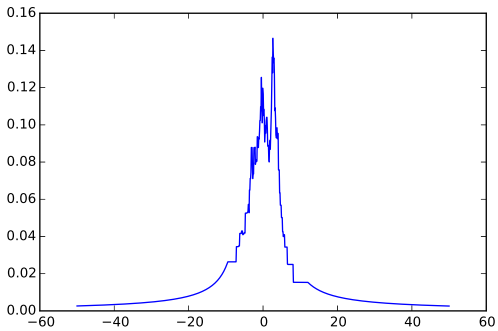
(a) $K = 1$



(b) $K = 5$



(c) $K = 10$



(d) $K = 15$

Figure 6: Estimated pdf of x_3 for different K.