Assignment 2

Probability, Statistics and Discrete Mathematics
16.4.2019
Salla Vesterinen
Helsinki Metropolia University of Applied Sciences

In [ ]:

```
%pylab inline
import numpy.random as random
from scipy.stats import mode, describe
```

```
Populating the interactive namespace from numpy and matplotlib
```

Problem 1

Create an array of 10,000 random integer numbers in range from 1 to 20 and make a histogram plot of the distribution. Pay special attention for selecting proper bins for the histogram.

In [ ]:

```
from numpy.random import randint
x1 = numpy.random.randint(low=1, high=21, size=10000)
```

In [ ]:

```
count, bins, ignored = plt.hist(x1, bins = range(1,22), align = "left", edgecolor = "black")
plt.title("Histogram of values")
```

Problem 2

Create an array of 10,000 normally distributed random numbers having a mean value of 10.0 and standard deviation of 2.50.

- Observe and write down your comments how this distribution differs from the distribution in problem 1. Explain why.
- Write a code that calculates the mean value and the standard deviation from the array and prints them out.
- Compare the calculated values to the given values. Observe how much do the calculated values differ from the given values and try to explain the amount of the difference.

In [ ]:

```
from numpy.random import randn

x2 = numpy.random.normal(loc=10.0, scale=2.50, size=10000)

count, bins, ignored = plt.hist(x2, bins = range(1,22), align = "left", edgecolor = "black")
plt.title("Histogram of values")
```

This distribution differs from the problem 1 distribution because
in problem 1 the probabilities are theoretically equal between all values, whereas
in problem 2 with the mean being 10 the probability is highest at that point and close around it

In [ ]:

```
print(np.mean(x2))
x2.std()
```

The mean and standard deviation are very close to the given ones
because the values were given, but the sample size restricts the accuracy

Problem 3

We want to simulate a random experiment where we toss a coin several times and count how many times we get 'heads', e.g. the probability of $P(x)$, where x is the number of heads $x \in \{0, 1, 2, 3, \ldots, N\}$, and N is how many times we repeated the tossing. <img src="https://upload.wikimedia.org/wikipedia/commons/thumb/1/15/Coin_Toss_%283635981474%29.jpg/1024px-Coin_Toss_%283635981474%29.jpg (https://upload.wikimedia.org/wikipedia/commons/thumb/1/15/Coin_Toss_%283635981474%29.jpg/1024px-Coin_Toss_%283635981474%29.jpg)" alt="Tossing a coin", width = "200">

Write a code that uses a random number generator and counts the number of heads when we toss the coin 100 times. Write a loop around the counting and repeat the experiment 1,000 times. Plot the histogram of the number of heads.

Based on the histogram (**no calculations or coding!!!**), answer to the following questions:

- What was the most common value for the number of heads?
- What were the smallest and largest values for the number of heads?
- In which range of the values of the number of heads are typically? (e.g. try to estimate in which range is 90 % of the values)

Then use `mode()` function found from `scipy.stats` package and `mean()` and `median()` from `numpy` package to calculate the most common value, mean value and the median value of the experiment.

- How much these values differ from each other? Can you explain the differences?

Tip: See the comparison of mean, median and mode (https://en.wikipedia.org/wiki/Mode_(statistics)#Comparison_of_mean,_median_and_mode) in Wikipedia.

In [ ]:

```
H = np.zeros(1000)
i=0
for n in range(1000):
    n = numpy.random.randint(low=0, high=2, size=100)
    y = (n==0).sum()
    H[i] = y
    i=i+1
    None
print(H)
```

In [ ]:

```
count, bins, ignored = plt.hist(H, bins = range(30,70), align = "left", edgecolor = "black")
plt.title("Histogram of amount of heads")
```

Most common value: 50 Smallest number of heads: 34 Largest number of heads: 65 I'd estimate 90% of the values to be in the range of 43-57

In [ ]:

```
from scipy import stats
from numpy import mean, median

print("Mode:", stats.mode(H))
print("Mean:", numpy.mean(H))
print("Median:", numpy.median(H))
```

Mode is the most frequent value (a particular value that appears the most)
Mean is the middle value (when aligned in order with 50% on both sides)
Median is the sum of values in the set divided by the number of values
Therefore these values are similar in this context, but still different since they mean different things