Fitting data into probability distributions

Tasos Alexandridis

analexan@csd.uoc.gr

Problem statement

- Consider a vector of N values that are the results of an experiment.
- We want to find if there is a probability distribution that can describe the outcome of the experiment.
- In other words we want to find the model that our experiment follows.

Probability distributions: *The Gaussian distribution*

Probability density function: $f(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$

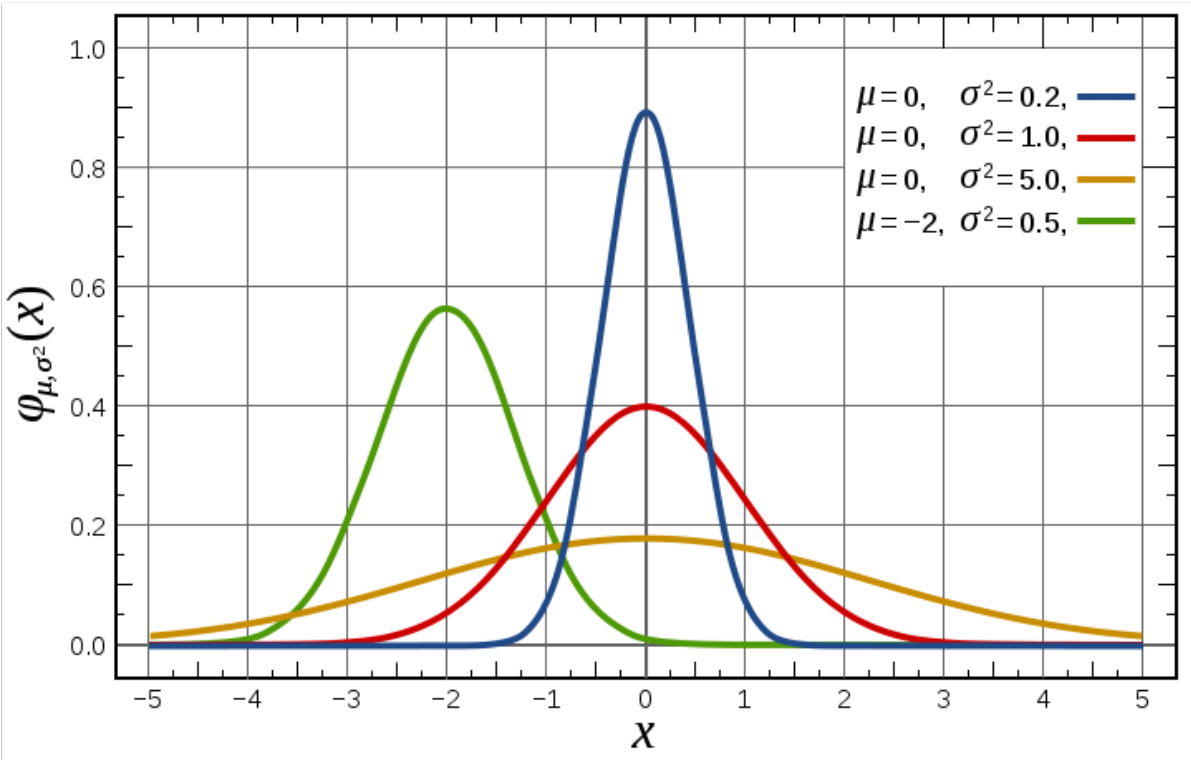

Figure: The Gaussian distribution

The red line is the *standard normal distribution*

Probability distributions: *The exponential distribution*

Probability density function: $f(x;\lambda) = \begin{cases} \lambda e^{-\lambda x}, x \geq 0 \\ 0, x < 0 \end{cases}$

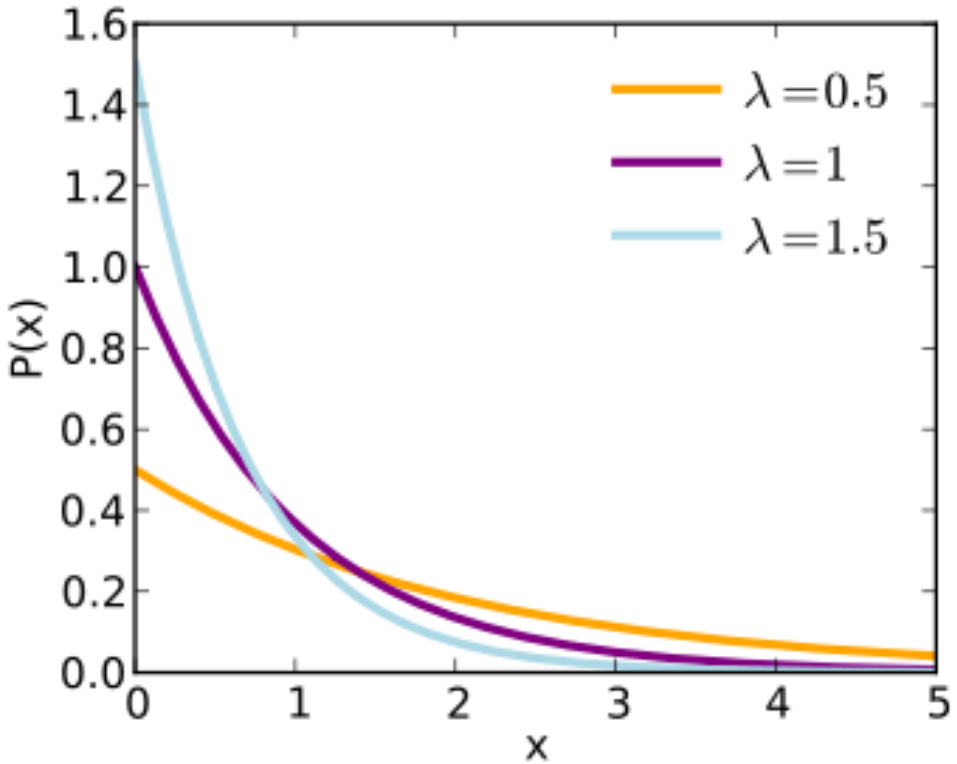

Figure: The exponential distribution

Fitting procedure: Overview

- Fit your real data into a distribution (i.e. determine the parameters of a probability distribution that best fit your data)
- Determine the goodness of fit (i.e. how well does your data fit a specific distribution)
  - qqplots
  - simulation envelope
  - Kullback-Leibler divergence

Example: Fitting in MATLAB

Generate data that follow an exponential distribution with $\mu = 4$

```
values = exprnd(5,100,1);
```

Generate random Gaussian noise N(0,1)

```
noise = randn(100,1);
```

Add noise to the exponential distributed data so as to look more realistic

```
real_data = values + abs(noise);
```

Consider `real_data` to be the outcome of the experiment

Example: Fitting in MATLAB
Test goodness of fit using qqplot

Generate synthetic data from the initial probability distribution and plot the real versus the sythetic data

The closer the points are to the y=x line, the better the fit is.

```
syntheticData = exprnd(5,100,1);
qqplot(real_data,syntheticData);
```

Example: Fitting in MATLAB
Test goodness of fit using qqplot

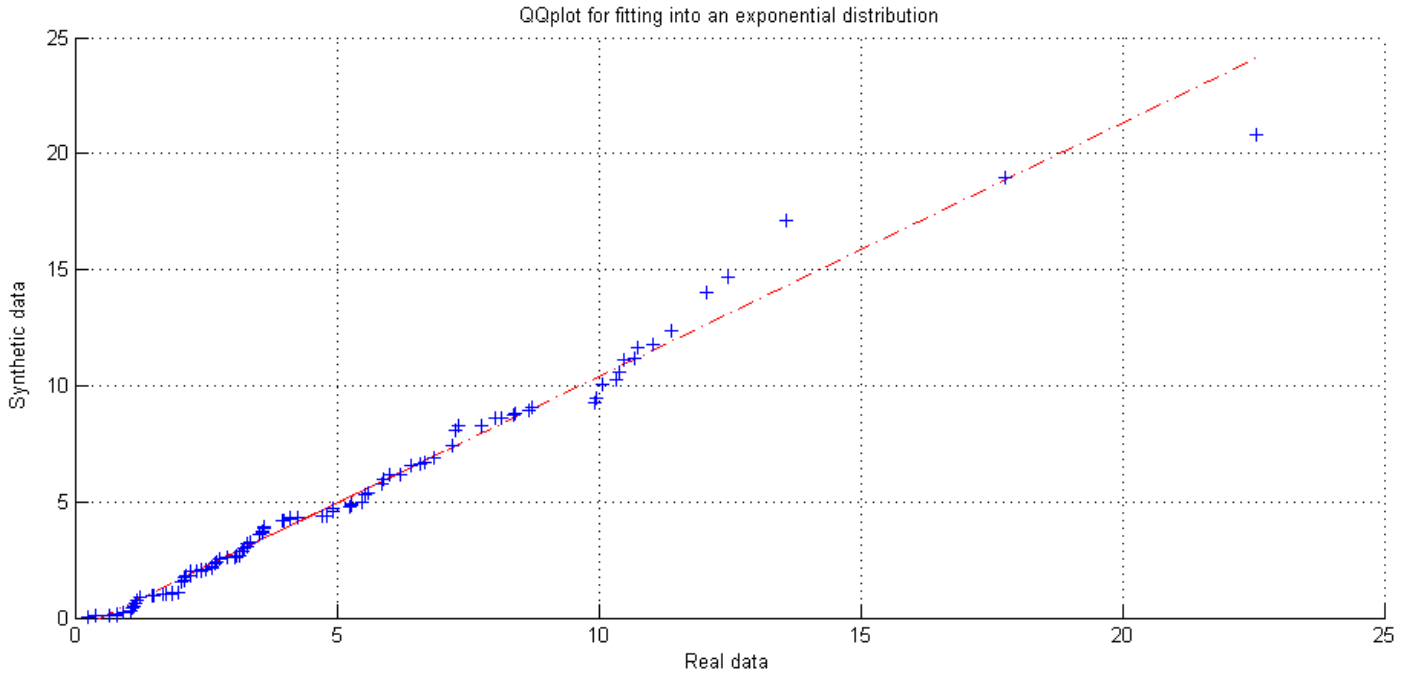

Figure: QQplot for fitting into an exponential distribution

Example: Fitting in MATLAB
Test goodness of fit using qqplot

Now generate samples from a Gaussian distribution

```
synthetic_data2 = normrnd(0,1,100,1);
```

Make the qqplot again:

```
qqplot(real_data,synthetic_data2
```

Fix axes and draw y=x line

```
xlim( [min([a;b]) max([a;b])]);
ylim( [min([a;b]) max([a;b])]);
plot( min([a;b]):max([a;b]), min([a;b]):max([a;b]),
'r');
```

Test goodness of fit using qqplot