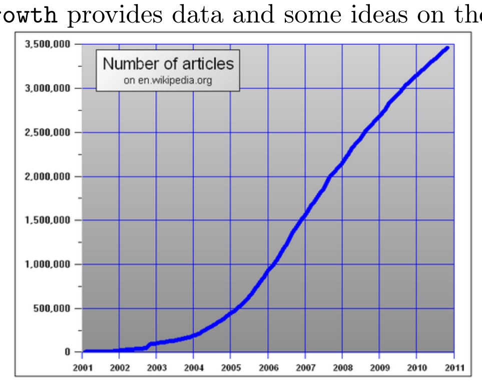
Modelling the size of Wikipedia

The wikipedia entries `http://en.wikipedia.org/wiki/Size_of_wikipedia` and `http://en.wikipedia.org/wiki/Wikipedia:Modelling_Wikipedia%27s_growth` provides data and some ideas on the growth of wikipedia.



The number of articles are growing, but the growth rate is slowing down for various reasons.

The models used in the entries are deterministic models, leading to poor fit (the growth is clearly stochastic!). The project is therefore to replace the deterministic model with a stochastic model (e.g. a SDE model) that captures the features discussed in the entries.

Remark: This type of models can be used to model almost any growing system. Have a look at the statistics at `www.gapminder.org` to see how different quanities are related, e.g. health vs GNP or GNP vs electricity consumption. It has been observed that electricity consumption and economic growth are firmly related, and it was pointed out at some point by Paul Krugman during the financial crises that even the Chinese economy probably slowed down `http://krugman.blogs.nytimes.com/2009/05/29/what-you-dont-know/`. Feel free to find your own example where this type of 'limited growth' models can be applied.

Navigation on a lake

Assume you are out fishing on a lake (or in an archipelago). After an intense morning of battling the beasts of the deep you take a nap in the boat. You wake up several hours later, completely disoriented. You can safely assume you have drifted around quite a lot. What's even more troubling is that a think fog has settled over the water. You approximate the visibility to about five meters. Looking around in your boat, you find the following objects:

- One topological map of the lake/archipelago.
- One laptop running MatLab, R or similar.
- One depth gauge of lower quality. The instructions on the back reveal "Measurement error $\pm 15\%$".
- (Optional) One compass.

Sitting in the boat you try to asses the surrounding conditions. The rippling of the water surface reveals some wind. However, you know that there are some currents as well. You assume that the effects of wind and current are of the same size in average, i.e. the drift has zero expectation. However, the variance can be assumed to be proportional to time.

The topological maps are given in the files "Bolmen", "Ringsjon", and "Vombsjon" for the lakes, and "Oresund" and "Skagerack" for the sea. Each lake (sea) contains a small, blue bar. This is home. Your task is, starting anywhere on the lake, to find your way back. Accordingly, pick a position in the lake for reference. Next, using only the equipment found in the boat, try to locate yourself. At this stage you of course assume that the position is unknown. Use it *only* to read off the map. If you are so inclined, you can also try to sail back home.

You can choose to use the compass or not. Naturally, a solution without using the compass is more advanced than one with compass and is *not* expected for a pass. However for the higher grades, creativity will be rewarded. Feel free to add extra complications of your own design.

We encourage you to supplement the report with executable program code and/or visual illustrations. To draw topological maps in MatLab one can use the following:

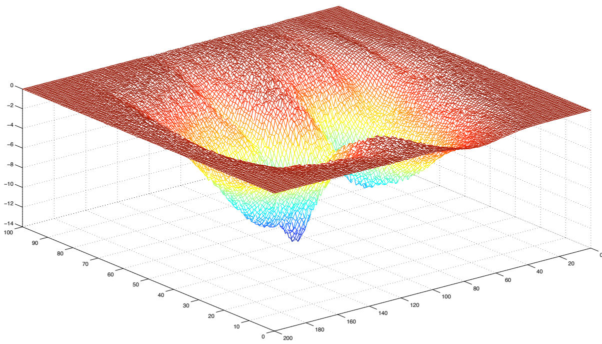


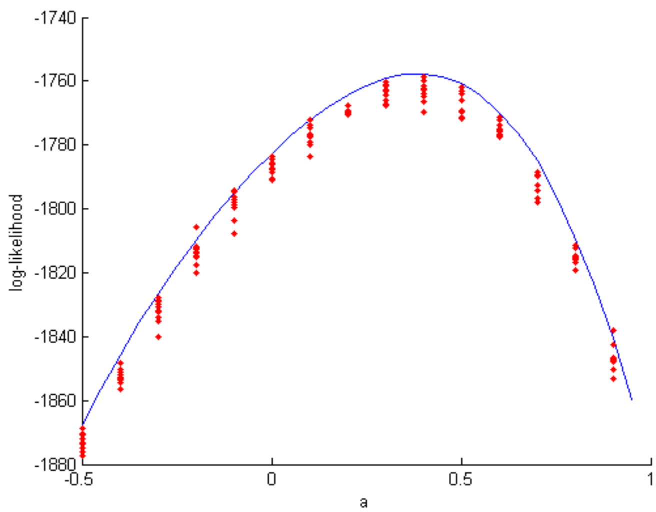
Figure 1: Vombsjön topological map.

```
>> contour(xscale,yscale,Depth,10)
>> hold on
>> line([harbour(1)-4 harbour(1)],[harbour(2) harbour(2)],'LineWidth',4)
```

for a jetty in the x-direction. Also the routine `plotlake.m`, found on the course home page, can make nice 3D plots. Navigation of this type is usually referred to as *inertia navigation*, see e.g. `http://en.wikipedia.org/wiki/Inertial_navigation_system`.

Estimating parameters using particle filters and SPSA

One of the most severe limitations of particle filters is that the standard likelihood approximation is non-smooth in the parameter space, making optimization difficult. Ten independent likelihood approximations for a linear Gaussian model ($a = 0.4$) computed using a particle filter are presented below where the true likelihood also is included.



Maximization of the likelihood can be done using stochastic approximation. The purpose of the project is to implement a trivial (i.e. scalar, linear Gaussian model) and a non-trivial model of your choice and estimate the parameters using SPSA, see `http://www.jhuapl.edu/SPSA/` for details and a review paper on the method. Use the simple model to calibrate the SPSA by comparing it to the optimizing the likelihood from the Kalman filter, and proceed with the non-trivial model of your choice when you feel ready.

MatLab code can be found on the SPSA home page, code for the particle filter was provided during the lectures.

Recursive, adaptive estimation of GARCH parameters

One of the problems with GARCH models is that structural breaks causes unrealistic parameter estimates, an effect known as the IGACRH effect. It has even been showed empirically that estimating GARCH parameters on sufficiently long data series leads with few exceptions to IGARCH models.

The purpose of this project is to implement a recursive, adaptive estimator for GARCH processes according to the techniques presented during the lectures.